

UNIVERSIDAD DE COSTA RICA
SISTEMA DE ESTUDIOS DE POSGRADO

CREACIÓN DE UNA HERRAMIENTA COLABORATIVA
PARA EL APRENDIZAJE DE LA PROGRAMACIÓN QUE
INCENTIVE LA COLABORACIÓN EN NIÑOS CON EDADES
COMPRENDIDAS ENTRE 4 Y 6 AÑOS

Tesis sometida a la consideración de la Comisión del Programa de
Estudios de Posgrado en Computación e Informática para optar al
grado y título de Doctorado Académico en Computación e Informática

KRYSCIA DAVIANA RAMÍREZ BENAVIDES

Ciudad Universitaria Rodrigo Facio, Costa Rica

2016

DEDICATORIA

Este trabajo de investigación, principalmente, se lo dedico a las dos personas más importantes de mi vida: mi compañero, amigo y esposo Franklin García Marín, y a mi hija (mi mayor bendición) Luz Daviana García Ramírez. Le agradezco a mi esposo por su amor, paciencia, su apoyo y ayuda incondicional, y por darme valor para seguir adelante en todo momento. A mi niña Luz Daviana, por ser mi inspiración, mi motivación y mi usuario número uno en esta investigación.

A Dios, ya que sin él no somos nadie ni nada podemos hacer. Dios es quien nos concede el privilegio de la vida y nos ofrece lo necesario para lograr nuestras metas, le agradezco por otorgarme la sabiduría y la salud para hacerlo.

A mis padres: María Eugenia Benavides Vargas y Pedro Ramírez Hernández, porque ellos siempre han estado junto a mí en las buenas y en las malas; me educaron, me impartieron valores para conducirme correctamente y me ofrecieron el sabio consejo en el momento oportuno.

Y finalmente; a mi amada e inolvidable abuelita: Luzmilda Vargas (mi otra mamá), gracias por guiarme y protegerme, estés donde estés. Aunque no estés físicamente a mi lado, tu presencia cada día crece más en mi corazón, ya que a ti te debo lo que soy.

AGRADECIMIENTOS

A mi director de tesis: Dr. Luis A. Guerrero Blanco, que siempre creyó y apoyó este trabajo desde un inicio. A mis asesoras: Dra. Gabriela Marín Raventós y Dra. Susan Francis Salazar, quienes, con su apoyo y sus valiosos comentarios y observaciones, dieron grandes aportes en la realización de esta investigación.

A los profesores del Doctorado en Computación e Informática de la Universidad de Costa Rica (UCR), por sus enseñanzas y su guía durante el periodo de estudios del Programa de Posgrado. En especial a los profesores: Dr. Vladimir Lara Villagrán, Dra. Gabriela Barrantes Sliesarieva y Dra. Ileana Castillo Arias, por su ayuda y guía durante todo proceso de investigación doctoral.

A los investigadores de la Fundación Omar Dengo (FOD), de la Unidad de Aprendizaje lógico, científico y robótica (Instituto Innova / Desarrollos Educativos), bajo la coordinación de la Máster Ana Lourdes Acuña Zúñiga; por su valiosa ayuda en la planeación y desarrollo de los diferentes pilotos llevados a cabo como parte de la evaluación de esta investigación. Además, por creer en mí para llevar a cabo la investigación y ser mis aliados desde el primer día de esta aventura.

A los investigadores del RExLab de la Universidade Federal de Santa Catarina (UFSC), Campus Araranguá, Santa Catarina, Brasil, bajo la coordinación del Dr. Juarez Bento da Silva; por recibirme y hacerme parte de su equipo de investigación durante mi periodo de pasantía, y por su valiosa colaboración en el desarrollo de la herramienta colaborativa de programación y la realización del pilotaje realizado en una escuela pública de Araranguá, Santa Catalina, Brasil.

Al equipo de investigación USING (*USer Interaction Group*), por su apoyo y ayuda durante el proceso de la investigación, colaborando en las diferentes evaluaciones realizadas en este trabajo de investigación.

A los docentes y los niños, tanto de Costa Rica como de Brasil, participantes en las diferentes validaciones, pilotos y pruebas realizadas como parte de esta investigación; por su valiosa ayuda en la evaluación de esta investigación. A las diferentes Instituciones Educativas y su personal administrativo, por su colaboración en la realización del proceso de evaluación de este trabajo.

Un reconocimiento también a todas aquellas personas que ayudaron de alguna u otra forma en la realización de este trabajo.

“Esta tesis fue aceptada por la Comisión del Programa de Estudios de Posgrado en Computación e Informática de la Universidad de Costa Rica, como requisito parcial para optar al grado y título de Doctorado Académico en Computación e Informática.”

Dr. Arturo Camacho Lozano
**Representante de la Decana
Sistema de Estudios de Posgrado**

Dr. Luis Guerrero Blanco
Director de Tesis

Dra. Gabriela Marín Raventós
Asesora

Dra. Susan Francis Salazar
Asesora

Dr. Vladimir Lara Villagrán
**Director
Programa de Posgrado en Computación e Informática**

Kryscia Ramírez Benavides
Candidata

TABLA DE CONTENIDOS

DEDICATORIA.....	II
AGRADECIMIENTOS.....	III
TABLA DE CONTENIDOS	VI
RESUMEN	XI
ABSTRACT.....	XII
LISTA DE FIGURAS.....	XIII
LISTA DE TABLAS.....	XVII
CAPÍTULO 1. INTRODUCCIÓN	1
1.1 Antecedentes	2
1.2 Uso de Dispositivos Móviles en Niños	13
1.3 Pensamiento Computacional	25
1.4 Importancia de la Programación y la Robótica en la Educación.....	28
1.5 Importancia del Aprendizaje Colaborativo en la Educación.....	30
1.6 Colaboración en Ambientes de Aprendizaje Apoyados por Computadora.....	33
1.7 Problema.....	37
1.8 Preguntas de Investigación.....	39
1.9 Objetivos de la Investigación	40
1.9.1 Objetivo General	40
1.9.2 Objetivos Específicos.....	40
1.10 Alcances y Limitaciones	41
1.11 Estructura de la Tesis	43
CAPÍTULO 2. REVISIÓN SISTEMÁTICA DE LITERATURA	44
2.1 Preguntas de la Revisión Sistemática de Literatura	44
2.2 Descripción General de la Revisión Sistemática de Literatura	44
2.2.1 Planificación de la Revisión.....	45
2.2.2 Proceso de la Revisión	46
2.3 Herramientas de Programación Basadas en Robots Orientadas a Niños	48

CAPÍTULO 3. APRENDIZAJE COLABORATIVO.....	63
3.1 Teorías Psicosociales de la Educación	68
3.1.1 Teoría Psicogenética de Piaget.....	68
3.1.2 Teoría Sociocultural de Vygotsky	71
3.1.3 Comparación de las Teorías de Piaget y Vygotsky	74
3.2 Desarrollo de la Niñez entre 4 y 6 años.....	76
3.3 Aprendizaje Colaborativo Apoyado por Computadora.....	80
3.3.1 Aprendizaje Colaborativo Móvil.....	82
3.3.2 Automatización de Actividades del Aprendizaje Colaborativo	86
CAPÍTULO 4. USABILIDAD	91
4.1 Diseño Centrado en el Usuario.....	93
4.2 Evaluación con Expertos	96
4.3 Test de Usabilidad.....	100
4.4 Usabilidad Móvil.....	106
4.4.1 Dificultades en la Evaluación de Usabilidad Móvil.....	107
4.4.2 Evaluación con Expertos	108
4.4.3 Test de Usabilidad.....	109
4.4.4 Tests en Laboratorio versus Tests de Campo.....	111
CAPÍTULO 5. METODOLOGÍA.....	114
5.1 Fases de la Investigación.....	115
5.2 Usabilidad de TITIBOTS Colab en la Primera Infancia	116
5.2.1 Fase 1. Evaluación de TITIBOTS	125
5.2.2 Fase 2. Evaluación de TITIBOTS Colab.....	128
5.3 Aporte de TITIBOTS Colab en el Proceso de Colaboración de la Primera Infancia	131
5.3.1 Participantes	135
5.3.2 Espacio Físico y Materiales.....	135
5.3.3 Procedimiento.....	135
5.3.4 Análisis de Datos.....	144
5.4 Proceso de Diseño Centrado en el Usuario	145

CAPÍTULO 6. TITIBOTS	152
6.1 Descripción General	152
6.2 Guía de Lineamientos.....	154
6.3 TITIBOTS – Herramienta Individual de Programación.....	171
6.3.1 Arquitectura.....	174
6.3.2 Interfaz Móvil de Usuario	177
6.3.3 Robots.....	179
6.3.4 Protocolo de Comunicación	179
6.4 TITIBOTS Colab – Herramienta Colaborativa de Programación.....	180
6.4.1 Arquitectura.....	183
6.4.2 Interfaz Móvil de Usuario de la Docente	186
6.4.3 Interfaz Móvil de Usuario del Estudiante.....	188
6.4.4 Robots.....	191
6.4.5 Protocolo de Comunicación	192
CAPÍTULO 7. RESULTADOS Y DISCUSIÓN	201
7.1 Usabilidad de la Herramienta en la Primera Infancia.....	201
7.1.1 Usabilidad de TITIBOTS	201
7.1.2 Usabilidad de TITIBOTS Colab.....	212
7.2 Aporte en el Proceso de Aprendizaje de la Programación de la Primera Infancia....	218
7.3 Aporte en el Proceso de Colaboración de la Primera Infancia.....	222
7.3.1 Perfil de los Participantes	223
7.3.2 Medición del Desempeño de los Grupos Colaborativos	226
7.3.3 Medición del Desempeño de las Actividades Colaborativas	237
CAPÍTULO 8. CONCLUSIONES	239
8.1 Aportes Adicionales	243
8.2 Trabajo Futuro.....	246
REFERENCIAS.....	250
APÉNDICES	265
APÉNDICE 1. DIVULGACIÓN DE LA INVESTIGACIÓN	265

APÉNDICE 2. REFERENCIAS DE LA REVISIÓN SISTEMÁTICA DE LITERATURA.....	269
APÉNDICE 3. GUÍA DE VALIDACIÓN DE TITIBOTS.....	273
APÉNDICE 4. GUÍA DE OBSERVACIÓN DE TITIBOTS	274
APÉNDICE 5. ENTREVISTA SEMIESTRUCTURADA DE TITIBOTS – PERFIL DE USUARIO	276
APÉNDICE 6. CUESTIONARIO DE SATISFACCIÓN DE TITIBOTS.....	277
APÉNDICE 7. ENTREVISTA SEMIESTRUCTURADA DE TITIBOTS COLAB – PERFIL DE USUARIO.....	278
APÉNDICE 8. PRUEBA DE MEMORIA DE TITIBOTS Y TITIBOTS COLAB....	279
APÉNDICE 9. CUESTIONARIO DE SATISFACCIÓN DE TITIBOTS COLAB....	280
APÉNDICE 10. ENTREVISTA SEMIESTRUCTRADA DE LAS DOCENTES – PERFIL ACADÉMICO Y LABORAL.....	281
APÉNDICE 11. LISTA DE COTEJO PARA LA EVALUACIÓN DEL APRENDIZAJE COLABORATIVO.....	282
APÉNDICE 12. EVALUACIÓN DE LOS DESAFÍOS.....	284
APÉNDICE 13. VALIDACIÓN DE LA ICONOGRAFÍA Y LA INTERACCIÓN CON NIÑOS DE PREESCOLAR.....	285
APÉNDICE 14. PROCESO DE PRUEBAS CON USUARIOS FINALES – TITIBOTS	289
APÉNDICE 15. DESARROLLO Y USO EN UN ENTORNO REAL – TITIBOTS..	292
APÉNDICE 16. PROCESO DE PRUEBAS CON USUARIOS FINALES – TITIBOTS COLAB	296
APÉNDICE 17. DESARROLLO Y USO EN UN ENTORNO REAL – TITIBOTS COLAB	301
APÉNDICE 18. DATOS DE LOS DESAFÍOS DEL TALLER CON TITIBOTS.....	306
APÉNDICE 19. DATOS DE LOS DESAFÍOS DEL TALLER CON TITIBOTS COLAB EN BRASIL.....	310
APÉNDICE 20. DATOS DE LOS DESAFÍOS DEL TALLER CON TITIBOTS COLAB EN COSTA RICA.....	314

APÉNDICE 21. DATOS DEL CUASI-EXPERIMENTO.....	318
APÉNDICE 22. DIBUJO DE UNA NIÑA SOBRE EL TALLER.....	327
APÉNDICE 23. OBSERVACIONES FINALES DE LOS EVALUADORES CON RESPECTO AL PROCESO DE COLABORACIÓN.....	328

RESUMEN

La presente investigación se enfoca en la creación de una herramienta colaborativa para el aprendizaje de la programación, llamada TITIBOTS Colab, que incentiva la colaboración en niños de la primera infancia, con edades entre 4 y 6 años. Se realizaron dos evaluaciones: una para valorar la usabilidad y la funcionalidad de la herramienta, y la otra para valorar el aporte de la herramienta en el proceso de colaboración en niños con edades entre 4 y 6 años.

La justificación del estudio radica en su valor teórico, utilidad práctica y relevancia social, debido a que fomenta la solución de problemas, el desarrollo del pensamiento lógico, la abstracción, la creatividad y el aprendizaje, a través de exploraciones lúdicas. También propicia el desarrollo de habilidades blandas, proporcionado por el proceso de colaboración, desde edades tempranas. El estudio se sustenta, principalmente, en las teorías psicosociales de la educación de Piaget y Vygotsky, y en estudios realizados por diferentes investigadores, reconocidos a nivel mundial, sobre la importancia de incorporar la programación y la robótica en las actividades curriculares de los niños de preescolar. Según estos estudios, los niños aprenden a ver la tecnología, la robótica y la programación como herramientas creativas, y experimentan las recompensas de comprensión y realización.

En esta investigación se contó con expertos en el área de programación y robótica educativa, así como docentes de preescolar. En total participaron 10 expertos, 17 docentes de preescolar y 11 investigadores. Además, participaron en las diferentes evaluaciones, aproximadamente 100 niños con edades entre 4 y 6 años.

Metodológicamente, el trabajo investigativo se abordó en cuatro etapas. La primera etapa consistió en identificar de las actividades del aprendizaje colaborativo que pueden ser automatizadas para desarrollar una herramienta que apoye el proceso de aprendizaje de la programación en niños de la primera infancia. En la segunda etapa se diseñó y construyó el prototipo de una herramienta colaborativa de programación orientada a esta población. En la tercera etapa se creó una guía de lineamientos para diseñar y construir la herramienta. Finalmente, en la cuarta etapa, se evaluó la usabilidad y la funcionalidad de la herramienta, y se valoró su aporte en el proceso de colaboración.

Los resultados obtenidos muestran que la herramienta colaborativa incentiva la colaboración en niños con edades entre 4 y 6 años, ya que los resultados obtenidos fueron estadísticamente significativos. En las poblaciones de niños donde se utilizó la herramienta colaborativa, se pudo apreciar que los componentes esenciales del aprendizaje colaborativo se evidenciaron, casi en su totalidad, en los niños. Por otra parte, al evaluar la usabilidad y la funcionalidad de la herramienta, se obtuvieron buenos resultados, demostrando que la herramienta colaborativa es intuitiva y fácil de usar para los niños de la primera infancia y se garantiza que hace lo que debe, no falla y satisface los requisitos.

Frases clave: Herramienta Colaborativa de Programación, Niños de la Primera Infancia, Aprendizaje colaborativo, Programación, Robótica, Interfaces Móviles, Colaboración, Usabilidad.

ABSTRACT

This research is focused on creating a collaborative tool to help programming education, called TITIBOTS Colab, that promotes collaboration on children with ages between 4 and 6 years. Two evaluations were made: one to verify the input of the tool during collaboration process with children between 4 and 6 years, and the other to measure usability and functionality of the tool.

The study is justified based on its practical utility, theory value and social relevance, since it supports skills on kids such as problem solving, logical thinking, abstraction, creativity, and learning, all through playful explorations. It also helps developing soft skills, providing a collaborative process on early ages. The research is sustained mainly in psychosocial theories on education from Piaget and Vygotsky and also on research on robotics and curricular activities made by several researchers. That research says that children see technology, robotics and programming as creative tools and experience development and comprehension rewards.

This research work was carried out with experts in the field of software development and educational robotics, as well as preschool teachers. In all, ten experts, seventeen preschool teachers and eleven researchers. They also participated in various evaluations, approximately 100 children with ages between 4 and 6 years.

Methodologically, the research work was addressed in four stages. The first stage consisted in identifying collaborative learning activities that could be automated to develop a tool that supports the learning process of programming for children in early childhood. In the second stage, a prototype of a collaborative tool was designed and built. In the third stage, guidelines were created for the design and construction of a generated collaborative programming tool for children in early childhood. Finally, in the fourth stage, the usability and functionality of the tool was evaluated, and the contribution was assessed in the collaborative process.

Results show that the collaborative tool encourages collaboration in children aged between 4 and 6 years, since the quasi-experiment results were statistically significant. In populations of children where the collaborative tool was used, it was observed that the essential components of learning collaborative were evident, almost entirely in children. Moreover, in assessing the usability and functionality of the tool, very good results were obtained, showing that the collaborative tool is intuitive and easy to use by children in early childhood and is guaranteed to do what it should, does not fail and complies with what has been requested (meets requirements).

Keyphrases: Collaborative Programming Tool, Early Childhood, Collaborative Learning, Programming, Robotics, Mobile Interfaces, Collaboration, Usability.

LISTA DE FIGURAS

Figura 1. Porcentaje de plataformas móviles que tienen en el hogar niños entre 0 y 8 años (Common Sense Media Research Study, 2013).	18
Figura 2. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles (Common Sense Media Research Study, 2013).....	19
Figura 3. Tiempo promedio por día que usan medios de pantalla los niños entre 0 y 8 años (Common Sense Media Research Study, 2013).	19
Figura 4. Porcentaje de niños entre 0 y 8 años de familias de bajos ingresos o de minorías que usan dispositivos móviles (Common Sense Media Research Study, 2013).	20
Figura 5. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles según el tipo de ingresos de sus familias (Common Sense Media Research Study, 2013).	20
Figura 6. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles para actividades específicas (Common Sense Media Research Study, 2013).....	21
Figura 7. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles para actividades específicas por plataforma (Common Sense Media Research Study, 2013).	21
Figura 8. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles para diferentes tipos de aplicaciones móviles por rango de edades (Common Sense Media Research Study, 2013).	22
Figura 9. Tiempo en minutos del uso de medios (Carmioli & Pérez, 2015b).....	23
Figura 10. Preferencia del uso de aplicaciones mediante tableta o celular (Carmioli & Pérez, 2015b). 23	
Figura 11. Clasificación de las herramientas de programación por tipo de interfaz.	61
Figura 12. Clasificación de las herramientas de programación por rangos de edad.	61
Figura 13. Proceso del DCU (International Organization for Standardization, 1999)...	95
Figura 14. Componentes a definir en un test de usabilidad (Jakob Nielsen & Budiu, 2012). 110	
Figura 15. Fases y técnicas del DCU utilizadas.	116
Figura 16. Método de evaluación de TITIBOTS.	126

Figura 17.	Método de evaluación de TITIBOTS Colab.....	130
Figura 18.	Diseño de dos series cronológicas realizado en la investigación.....	134
Figura 19.	Materiales utilizados en el pre-test y pos-test.....	136
Figura 20.	Actividades del pre-test y pos-test.....	137
Figura 21.	Acciones esperadas en cada componente esencial del aprendizaje colaborativo.	139
Figura 22.	Prueba- <i>t</i> : Diferencia entre grupos independientes para puntuaciones diferenciales.	141
Figura 23.	Prueba <i>U</i> de Mann-Whitney: Diferencia entre grupos independientes para puntuaciones diferenciales.....	141
Figura 24.	Diagrama del proceso metodológico de la investigación.....	146
Figura 25.	Proceso de DCU.....	147
Figura 26.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño de Interacción”.	159
Figura 27.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño de Pantalla”.	162
Figura 28.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Texto en la Pantalla”.	163
Figura 29.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño Visual”.....	165
Figura 30.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Composición Visual”.	166
Figura 31.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño de Audio”...	167

Figura 32.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Intencionalidad”.....	168
Figura 33.	Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Otras Notas de Diseño”. 169	
Figura 34.	Evolución de TITIBOTS.	173
Figura 35.	Herramienta de programación TITIBOTS.....	174
Figura 36.	Arquitectura de TITIBOTS.....	176
Figura 37.	Diagrama con la arquitectura de TITIBOTS.	177
Figura 38.	Robots utilizados en TITIBOTS.....	179
Figura 39.	Herramienta colaborativa de programación TITIBOTS Colab.	182
Figura 40.	Arquitectura de TITIBOTS Colab.	185
Figura 41.	Diagrama de TITIBOTS Colab – Interfaz de Usuario de la Docente.....	187
Figura 42.	Diagrama de TITIBOTS Colab – Interfaz de Usuario del Estudiante.....	189
Figura 43.	Interfaces de TITIBOTS Colab de solicitud del sexo y la edad.	189
Figura 44.	Interdependencia positiva de distribución de recursos.	190
Figura 45.	Robots utilizados en TITIBOTS Colab.	192
Figura 46.	Información del perfil del usuario de TITIBOTS.....	202
Figura 47.	Información del perfil del usuario de TITIBOTS Colab.	213
Figura 48.	Información del perfil del usuario de TITIBOTS Colab – Pregunta 5.	214
Figura 49.	Información del perfil del usuario de TITIBOTS Colab – Pregunta 7.	214
Figura 50.	Resultados del cuestionario de satisfacción de TITIBOTS Colab.....	217
Figura 51.	Categorías de los dibujos realizados por los niños.	217
Figura 52.	Información del perfil del usuario del cuasi-experimento.	224
Figura 53.	Información del perfil del usuario del cuasi-experimento – Pregunta 5.....	225
Figura 54.	Información del perfil del usuario del cuasi-experimento – Pregunta 7.....	225

Figura 55.	Promedios de los porcentajes de acciones logradas de cada componente esencial del aprendizaje colaborativo de los dos test para la muestra de grupos colaborativos.	229
Figura 56.	Promedios de los porcentajes de acciones logradas de cada componente esencial del aprendizaje colaborativo de los dos test para la muestra de estudiantes.....	229
Figura 57.	Tamaño del efecto para el contraste de medias de la muestra de los grupos colaborativos.	235
Figura 58.	Tamaño del efecto para el contraste de medias de la muestra de los estudiantes.	235
Figura 59.	Tamaño del efecto de las diferencias para los porcentajes de acciones logradas entre el pre-test y el pos-test de la muestra de los grupos colaborativos.	236
Figura 60.	Tamaño del efecto de las diferencias para los porcentajes de acciones logradas entre el pre-test y el pos-test de la muestra de los estudiantes.....	236
Figura 61.	Metodología de enseñanza-aprendizaje de los talleres.	244
Figura 62.	Procedimiento de Solución de Problemas.	245
Figura 63.	Espacio físico, materiales y proceso de la validación.....	286
Figura 64.	Patrones de interacción y opciones de secuencia.....	287
Figura 65.	Niños trabajando en el respectivo espacio de trabajo.....	290
Figura 66.	Actividades realizadas en el taller de TITIBOTS.....	293
Figura 67.	Actividades realizadas en el taller de TITIBOTS Colab en Brasil.....	297
Figura 68.	Actividades realizadas en el taller de TITIBOTS Colab en Costa Rica.	302

LISTA DE TABLAS

Tabla 1.	Resumen de herramientas de programación orientadas a la niñez.	14
Tabla 2.	Datos del proceso de mapeo de la RSL.	47
Tabla 3.	Lista de herramientas de programación basadas en robots y orientadas a la niñez. 49	
Tabla 4.	Características de las herramientas de programación basadas en robots y orientadas a la niñez.....	56
Tabla 5.	Desarrollo de la niñez entre 4 y 6 años según las áreas de desarrollo.	78
Tabla 6.	Fases y actividades del proceso del AC (Collazos, 2003).	88
Tabla 7.	Actividades del proceso del AC que se pueden automatizar.	90
Tabla 8.	Descripción de las técnicas utilizadas en cada fase del DCU (Norman, 1983). 117	
Tabla 9.	Atributos de usabilidad, métricas y metas asociadas.	128
Tabla 10.	Características de las herramientas de programación.	153
Tabla 11.	Conjunto abierto de comandos de TTIBOTS.	180
Tabla 12.	Actividades del proceso del AC que se automatizaron en TITIBOTS Colab.	182
Tabla 13.	Mensajes del protocolo de comunicación de TITIBOTS Colab.....	199
Tabla 14.	Participantes de la evaluación de TITIBOTS.	202
Tabla 15.	Resultados de la validación de íconos, interfaz e interacción.....	203
Tabla 16.	Resultados de las métricas de usabilidad en TITIBOTS.	205
Tabla 17.	Cantidad de niños que realizaron con éxito los desafíos.	206
Tabla 18.	Tiempo promedio y cantidad de intentos promedio de los desafíos exitosos.	207
Tabla 19.	Resultados de las métricas de usabilidad en TITIBOTS.	210
Tabla 20.	Participantes de la evaluación de TITIBOTS Colab.....	212
Tabla 21.	Resultados de las métricas de usabilidad en Brasil.....	215
Tabla 22.	Resultados de las métricas de usabilidad en Costa Rica.....	216

Tabla 23.	Cantidad de intentos y tiempo de los desafíos por niño en el taller con TITIBOTS.	219
Tabla 24.	Cantidad de intentos de los desafíos por grupo colaborativo en el taller de Brasil con TITIBOTS Colab.	220
Tabla 25.	Tiempo de los desafíos por grupo colaborativo en el taller de Brasil con TITIBOTS Colab.	220
Tabla 26.	Cantidad de intentos de los desafíos por grupo colaborativo en el taller de Costa Rica con TITIBOTS Colab.	221
Tabla 27.	Tiempo de los desafíos por grupo colaborativo en el taller de Costa Rica con TITIBOTS Colab.	221
Tabla 28.	Niños participantes del cuasi-experimento.	223
Tabla 29.	Docentes participantes del cuasi-experimento.	224
Tabla 30.	Perfil de las docentes participantes del cuasi-experimento.	226
Tabla 31.	Porcentaje de acciones logradas de los grupos colaborativos en el pre-test.	227
Tabla 32.	Porcentaje de acciones logradas de los grupos colaborativos en el pos-test.	227
Tabla 33.	Porcentaje de acciones logradas de los niños de preescolar en el pre-test.	228
Tabla 34.	Porcentaje de acciones logradas de los niños de preescolar en el pos-test.	228
Tabla 35.	Resumen de pruebas de hipótesis para el contraste de medias de la muestra de los grupos colaborativos.	232
Tabla 36.	Resumen de pruebas de hipótesis para el contraste de medias de la muestra de los estudiantes.	233
Tabla 37.	Tamaño del efecto y coeficiente de correlación para el contraste de medias de la muestra de los grupos colaborativos.	234
Tabla 38.	Tamaño del efecto y coeficiente de correlación para el contraste de medias de la muestra de los estudiantes.	234
Tabla 39.	Medidas de desempeño de las actividades colaborativas en el pre-test.	237
Tabla 40.	Medidas de desempeño de las actividades colaborativas en el pos-test.	238
Tabla 41.	Resumen de las variables de los desafíos de evaluación de los talleres.	249

Tabla 42.	Métrica de usabilidad “Facilidad de aprendizaje” con TITIBOTS.....	306
Tabla 43.	Métrica de usabilidad “Eficiencia”, porcentaje de desafíos exitosos, con TITIBOTS. 306	
Tabla 44.	Métrica de usabilidad “Eficiencia”, tiempo promedio para completar un desafío, con TITIBOTS.	307
Tabla 45.	Métrica de usabilidad “Cualidad de ser recordado” con TITIBOTS.....	307
Tabla 46.	Métrica de usabilidad “Efectividad”, promedio de errores, con TITIBOTS. .	308
Tabla 47.	Métrica de usabilidad “Efectividad”, promedio de errores recuperados, con TITIBOTS. 308	
Tabla 48.	Métrica de usabilidad “Satisfacción” con TITIBOTS.	308
Tabla 49.	Preguntas sobre conceptos básicos de programación y robótica con TITIBOTS. 309	
Tabla 50.	Métrica de usabilidad “Facilidad de aprendizaje” con TITIBOTS Colab en Brasil. 310	
Tabla 51.	Métrica de usabilidad “Eficiencia”, porcentaje de desafíos exitosos, con TITIBOTS Colab en Brasil.....	310
Tabla 52.	Métrica de usabilidad “Eficiencia”, tiempo promedio para completar un desafío, con TITIBOTS Colab en Brasil.	311
Tabla 53.	Métrica de usabilidad “Cualidad de ser recordado” con TITIBOTS Colab en Brasil. 311	
Tabla 54.	Métrica de usabilidad “Efectividad”, promedio de errores, con TITIBOTS Colab en Brasil.....	312
Tabla 55.	Métrica de usabilidad “Efectividad”, promedio de errores recuperados, con TITIBOTS Colab en Brasil.....	312
Tabla 56.	Métrica de usabilidad “Satisfacción” con TITIBOTS Colab en Brasil.	312
Tabla 57.	Preguntas sobre conceptos básicos de programación y robótica con TITIBOTS Colab en Brasil.....	313

Tabla 58.	Métrica de usabilidad “Facilidad de aprendizaje” con TITIBOTS Colab en Costa Rica.	314
Tabla 59.	Métrica de usabilidad “Eficiencia”, porcentaje de desafíos exitosos, con TITIBOTS Colab en Costa Rica.	314
Tabla 60.	Métrica de usabilidad “Eficiencia”, tiempo promedio para completar un desafío, con TITIBOTS Colab en Costa Rica.	315
Tabla 61.	Métrica de usabilidad “Cualidad de ser recordado” con TITIBOTS Colab en Costa Rica.	315
Tabla 62.	Métrica de usabilidad “Efectividad”, promedio de errores, con TITIBOTS Colab en Costa Rica.	316
Tabla 63.	Métrica de usabilidad “Efectividad”, promedio de errores recuperados, con TITIBOTS Colab en Costa Rica.	316
Tabla 64.	Métrica de usabilidad “Satisfacción” con TITIBOTS Colab en Costa Rica.	316
Tabla 65.	Preguntas sobre conceptos básicos de programación y robótica con TITIBOTS Colab en Costa Rica.	317
Tabla 66.	Puntuaciones diferenciales de la muestra de los grupos colaborativos.	319
Tabla 67.	Puntuaciones diferenciales de la muestra de los estudiantes.	319
Tabla 68.	Estadística descriptiva de los datos de los grupos colaborativos del GC.	321
Tabla 69.	Estadística descriptiva de los datos de los grupos colaborativos del GE.	321
Tabla 70.	Estadística descriptiva de los datos de los estudiantes del GC.	321
Tabla 71.	Estadística descriptiva de los datos de los estudiantes del GE.	322
Tabla 72.	Estadística descriptiva de las puntuaciones diferenciales de la muestra de los grupos colaborativos.	322
Tabla 73.	Estadística descriptiva de las puntuaciones diferenciales de la muestra de los estudiantes.	322
Tabla 74.	Pruebas- <i>t</i> para las VD en el pre-test de la muestra de grupos colaborativos.	323
Tabla 75.	Pruebas <i>U</i> de Mann-Whitney para las VD en el pre-test de la muestra de grupos colaborativos.	323

Tabla 76.	Pruebas- <i>t</i> para las VD en el pre-test de la muestra de estudiantes.....	323
Tabla 77.	Pruebas <i>U</i> de Mann-Whitney para las VD en el pre-test de la muestra de estudiantes. 323	
Tabla 78.	Comprobación de suposiciones para la muestra de grupos colaborativos.....	324
Tabla 79.	Comprobación de suposiciones para la muestra de estudiantes.	324
Tabla 80.	Pruebas- <i>t</i> para las puntuaciones diferenciales de las VD de la muestra de grupos colaborativos.....	325
Tabla 81.	Pruebas <i>U</i> de Mann-Whitney para las puntuaciones diferenciales de las VD de la muestra de grupos colaborativos.....	325
Tabla 82.	Pruebas- <i>t</i> para las puntuaciones diferenciales de las VD de la muestra de estudiantes. 325	
Tabla 83.	Pruebas <i>U</i> de Mann-Whitney para las puntuaciones diferenciales de las VD de la muestra de estudiantes.	326

CAPÍTULO 1. INTRODUCCIÓN

El advenimiento de la era digital ha provocado que las recientes generaciones sean consideradas como nativas digitales, ya que han crecido junto con las nuevas tecnologías, rodeados de todas las herramientas de la era digital. Esto los ha convertido en usuarios expertos de las nuevas tecnologías (Prensky, 2001).

Por otra parte, la programación y la robótica han permitido el aprendizaje de conceptos complejos de una manera concreta y divertida, facilitando el aprendizaje en áreas como ciencia, tecnología, ingeniería y matemática, de gran importancia para lograr el desarrollo tecnológico y educativo de los países (Campe, Werner, & Denner, 2012). El aprendizaje de estas habilidades permite que los niños y los jóvenes puedan desarrollar competencias claves para alcanzar el éxito en la vida profesional y personal, según el modelo de las habilidades más importantes para el siglo XXI (Binkley et al., 2012).

En esta investigación se desarrolla y evalúa el impacto de la herramienta colaborativa de programación TITIBOTS Colab (TITI por el mono tití, BOTS por robots y Colab por colaborativo), en el proceso de colaboración en niños con edades entre 4 y 6 años. Con el fin de cumplir con este objetivo se identificaron actividades automatizables del aprendizaje colaborativo, se creó una herramienta para apoyar el proceso de aprendizaje colaborativo de la programación en niños de edad preescolar, se generó una guía de lineamientos para la construcción de herramientas de aprendizaje de la programación orientadas a preescolares, y se valoró el aporte de dicha herramienta en el proceso de colaboración en niños de la primera infancia.

La herramienta fue implementada para dispositivos móviles y con ella se puede programar robots, con el fin de que se promueva en los niños el desarrollo del pensamiento computacional desde edades tempranas a través de la programación de robots de forma colaborativa, mediante TITIBOTS Colab, se incentiva en los niños el desarrollo de habilidades, tanto blandas como técnicas.

A continuación, se presentan algunos antecedentes relacionados con la evolución de las herramientas de programación orientadas a la niñez, y la justificación de esta investigación. Posteriormente, se presenta una descripción detallada del problema en cuestión. Finalmente se presentan los objetivos, los alcances y las limitaciones de la presente investigación.

1.1 Antecedentes

En la década de 1970, Papert y sus estudiantes del MIT iniciaron la investigación de métodos para introducir conceptos de programación en la niñez. Esta investigación inicial llevó a la creación del lenguaje de programación Logo y una tortuga que se movía por el suelo en respuesta a los comandos simples de este lenguaje, como adelante, atrás, izquierda y derecha. Papert y sus discípulos lograron que Logo quedara relativamente libre de las reglas sintácticas que hacen difícil el aprendizaje de la programación en niños (McNerney, 2004; Tempel, 2012).

Como parte del lenguaje Logo, se tenía una tortuga de piso (robot), que era una estructura con una pluma para dibujar figuras en una hoja grande de papel colocada en el suelo. Los niños debían enseñarle a la tortuga a dibujar formas básicas, como triángulos, cuadrados, círculos y letras. La tortuga también podría recibir instrucciones para dibujar formas complejas girando varias veces y formas simples girando un poco antes de cada repetición (McNerney, 2004; Papert, 1980).

Logo adquirió gran popularidad a principios de 1980, cuando las computadoras personales (especialmente el Apple II) se hicieron comunes en las escuelas. Sin embargo, pocas tortugas robóticas llegaron a la escuela, ya que los robots eran caros y poco fiables. Los investigadores del MIT temporalmente se apartaron de este enfoque y desarrollaron una tortuga virtual: una representación de la tortuga de suelo, pero en la pantalla de la computadora (McNerney, 2004; Tempel, 2012; Umaschi Bers & Horn, 2010). Logo era un lenguaje basado en escritura con instrucciones fáciles de aprender. El resultado de cada instrucción se veía plasmado en el recorrido efectuado por un “robot virtual” (popularmente una tortuga virtual).

En 1985, Logo Computer Systems Inc. (LCSI) introdujo el programa LogoWriter, que incluía la capacidad de procesar textos (de allí su nombre). La interfaz de usuario era simple y más intuitiva (Tempel, 2012). Dada la popularidad que alcanzó LogoWrite en el mundo, se implementó en muchos idiomas

Otra innovación en la década de los ochenta fue LEGO Logo. Mitchel Resnick y Steve Ocko, trabajando para el MIT Media Lab, desarrollaron un sistema que interconectaba Logo con motores, luces y sensores que se incorporaron en máquinas construidas con ladrillos LEGO y otros elementos. Los sistemas de robótica con Logo no eran nuevos, pero LEGO TC Logo fue un éxito comercial que llegó a miles de docentes y estudiantes (M. Resnick, Martin, Sargent, & Silverman, 1996).

Durante esa misma época (1988), la Fundación Omar Dengo (FOD) inició en Costa Rica el Programa de Informática Educativa, apoyado por el Ministerio de Educación Pública de Costa Rica e IBM de América Latina. Este proyecto puso a Logo en manos de la mayoría de los estudiantes y docentes de primaria en Costa Rica, mientras que un proyecto posterior hizo lo mismo con los estudiantes de secundaria (Zúñiga, 2003).

Una nueva versión de Logo, llamado MicroMundos, fue lanzado en 1993 por LCSI. Esta versión incorporó muchos cambios en la herramienta y en el lenguaje Logo. Se incluyeron muchas características adicionales como herramientas de dibujo, editores, fabricación de melodías, y la posibilidad de importar gráficos y sonidos. Estas nuevas características, junto con Logo, permitieron apoyar la creación de proyectos multimediales, juegos y simulaciones (Einhorn, 2011). Desde entonces, MicroMundos se ha actualizado varias veces. Actualmente LCSI ofrece MicroMundos EX, MicroMundos JR y MicroMundos EX Robotics, que incorpora la posibilidad de programación para el robot Cricket y el ladrillo LEGO RCX.

Otra innovación de la época fue el ladrillo programable LEGO, un proyecto de investigación del MIT encabezado por Fred Martin. Este ladrillo programable tenía un pequeño computador en su interior, al que se le descargaban programas escritos en un computador de escritorio. Traía su propia herramienta de programación, y los programas se ejecutaban de

forma autónoma para crear un robot inteligente (M. Resnick et al., 1996). LEGO comercializó en 1994 el ladrillo programable RCX, que se programaba por medio del software RoboLab. Esta herramienta estaba basada en íconos y permitía crear programas que controlaban el RCX. Ofrecía modos diferentes de programación, como Pilot e Inventor, adaptados al nivel de aprendizaje de los alumnos. Además, ofrecía el modo Investigator orientado a su uso en el laboratorio de ciencias (M. Resnick et al., 1996).

En el año 2006 se comenzó a comercializar el ladrillo NXT de LEGO Mindstorms. El LEGO Mindstorms es atractivo para todas las edades, y se puede usar para diferentes propósitos dependiendo de las habilidades de sus usuarios. Permite comunicarse con diferentes dispositivos a través de Bluetooth. Además, su ambiente de programación visual permite a un programador principiante crear fácilmente un nuevo programa. El software de NXT se basa en National Instruments' LabVIEW, el cual permite arrastrar íconos y soltarlos en la pantalla principal del programa (Kim & Jeon, 2007).

También se han desarrollado versiones más pequeñas del ladrillo programable, llamados *crickets*. Estos ladrillos se desarrollaron comercialmente con el nombre Pico-Cricket. Tenían su propia herramienta de programación PicoBlocks (Nam & Lee, 2011). Sin embargo, este ladrillo se discontinuó en el año 2010.

Como parte del proyecto del ladrillo programable, el Media Lab del MIT creó una nueva versión de Logo llamada LogoBlocks. En lugar de escribir líneas de código, en este lenguaje los programas se construyen mediante el ajuste de piezas, como si fuera un rompecabezas (M. Resnick et al., 1996).

Un Logo radicalmente diferente llamado StarLogo se introdujo en 1994. Fue una versión paralela desarrollada por Resnick en el MIT. StarLogo era una herramienta de programación de videojuegos en tres dimensiones, cuya sintaxis se basa en enlazar virtualmente varias cajas o bloques creando diferentes secuencias que definen el comportamiento de los objetos en el juego. Este tipo de lenguajes tienen la ventaja de que están libres de errores de sintaxis y semántica, ya que las diferentes cajas representan las instrucciones (sintaxis) y cada caja solo

se puede enlazar con un tipo de cajas predeterminado (metáfora de rompecabezas) evitando así errores semánticos (Mitchel Resnick, 1996).

Aproximadamente 10 años después, apareció StarLogo TNG, la siguiente generación de StarLogo. Entre sus mejoras estaban los gráficos 3D, el sonido y una interfaz de programación basada en bloques. Permitía la entrada de datos por teclado, lo que lo convirtió en una gran herramienta para la programación de videojuegos educativos (Begel & Klopfer, 2007).

En 1999 apareció la herramienta de programación Alice, la cual es innovadora y libre, permitiendo crear fácilmente una animación para contar una historia jugando un juego interactivo, o un video para compartir en la Web (Campe et al., 2012; Mullins, Whitfield, & Conlon, 2009). Alice está programada en Java y usa una herramienta sencilla basada en “arrastrar y soltar” para crear animaciones mediante modelos 3D (Cooper, Dann, & Pausch, 2000; Pausch et al., 1995). Este software fue desarrollado por los investigadores de la Carnegie Mellon University, entre los que destaca Randy Pausch. Fue creado para aprender programación orientada a objetos (Conway, Pausch, Gossweiler, & Burnette, 1994).

Durante el año 2001, surge el proyecto Electronic Blocks, de los investigadores Peta Wyeth y Helen C. Purchase. Este proyecto incluye elementos de programación tangible, bloques con circuitos electrónicos que se pueden apilar y formar estructuras que interactúan con el mundo físico. Estaba orientado a niños entre 4 y 8 años. Hay tres tipos: bloques de sensores, bloques de acción y bloques lógicos (Wyeth & Purchase, 2002). Los bloques de sensores detectan la luz, el sonido y el tacto, mientras que los bloques de acción son capaces de producir luz, sonido y movimiento. Mediante la conexión de bloques de sensores con los bloques de acción los niños pueden programar sus propias estructuras. La inclusión de bloques lógicos (negar, esperar y conmutar señales entre bloques) añade una dimensión adicional a las capacidades de las creaciones de los niños. Con el uso de los bloques, los niños pueden crear robots que chocan entre sí, luces que se encienden cuando se aplaude, y animales que susurran uno al otro. Mediante la colocación de los bloques electrónicos unos sobre otros, los niños

construyen programas, donde cada pila de bloques electrónicos es capaz de una función diferente.

En el 2004 apareció Etoys, una herramienta informática amigable para niños, basada en un lenguaje de programación orientado a objetos, el cual es utilizado para la educación. Crea medios con un modelo de objetos de secuencias de comandos para los diferentes objetos en que se ejecuta (James, 2012; Lee, 2011). Es código libre y abierto.

En el año 2005, LEGO saca al mercado el set LEGO WeDo, que consistía en un conjunto de piezas y partes mecánicas usadas para construir y diseñar modelos de LEGO. Traía un software fácil de usar, basado en íconos, que proporcionaba una herramienta de programación intuitiva con instrucciones de montaje, ejemplos de programación y sugerencias de actividades (Coxon, 2010; Coxon & Chandler, 2010; Mayerová, 2012).

En el verano del 2007 aparece un actor importante: Scratch. Scratch es una herramienta de programación que permite a principiantes obtener resultados sin tener que aprender a escribir de manera sintácticamente correcta, facilitando así el aprendizaje autónomo. Fue desarrollado por un equipo del Media Lab del MIT dirigido por Mitchel Resnick. Actualmente, Scratch se usa en casi todo el mundo en muchos entornos diferentes: escuelas, museos, centros comunitarios y hogares. Scratch está diseñado especialmente para personas entre 8 y 16 años de edad y se caracteriza por su simplicidad. Está constituido por símbolos iconográficos denominados “bloques” (J. H. Maloney, Pepler, Kafai, Resnick, & Rusk, 2008; J. Maloney, Resnick, Rusk, Silverman, & Eastmond, 2010). Esta herramienta aprovecha los avances en diseño de interfaces para hacer que la programación sea más atractiva y más accesible para todo aquel que se enfrente por primera vez al reto de aprender a programar. Debido a la popularidad de Scratch, la programación por bloques se ha generalizado y se usa actualmente en muchas aplicaciones.

En ese mismo año (2007) comienza la creación del software libre de código abierto miniBloq, una herramienta de programación gráfica para plataformas Arduino, Multiplo y otras (Tomoyose, 2014). Creada por Julián da Silva Gillig, esta herramienta es utilizada con los

kits educativos de robótica RobotGroup. La adopción de miniBloq en los robots educativos de RobotGroup comenzó en 2008, cuando se sumó como socio Da Silva Gillig, junto a Paves y el ingeniero Lucio Mercado. Este software, basado en C++, usa símbolos, permite visualizar en tiempo real los posibles errores de código y permite agregar código textual o incluso programar sin los bloques (pasa un proyecto de programación gráfica a textual, y viceversa con el proyecto ya comenzado) (Tomoyose, 2014).

También, en el 2007, apareció Tern, desarrollado por Michael S. Horn. Tern es un lenguaje computacional tangible diseñado para proporcionar una introducción a la programación. Está orientado a niños de 5 años en adelante para usarse en entornos educativos (Horn, Solovey, & Jacob, 2008). Se pueden crear programas para robots como el LEGO Mindstorms RCX o el iRobot Create. Tern está en exposición permanente en el Museo de Ciencias de Boston, con el nombre Robot Park. También se está usando para el proyecto Tangible Kindergarten del grupo Tufts University Developmental Technologies. Esta herramienta permite crear programas usando bloques de madera entrelazados que representan las acciones que debe llevar a cabo el robot. Los bloques de madera no contienen electrónica incorporada o fuentes de alimentación, lo que usa es una cámara web estándar conectada a una computadora para tomar la imagen del programa, que luego se convierte en código digital usando la librería de visión por computador TopCodes.

En el año 2010 aparece Enchanting, desarrollado por Clinton Blackmore, quien pertenece a la asociación Southern Alberta Robotics Enthusiasts (SABRE) de Canadá, junto con un grupo de colaboradores. Enchanting es una herramienta de programación para LEGO Mindstorms NXT que permite a los niños programar robots LEGO Mindstorms NXT. Se basa en Scratch y BYOB Snap!, y se puede usar con leJOS NXJ (Java para el NXT) (Yera-Gil, 2010). Es libre y de código abierto.

Durante el año 2011 comienza el proyecto ScratchJr, liderado por Marina Umaschi Bers, su grupo de investigación DevTech en la Tufts University y Mitch Resnick del Lifelong Kindergarten del Media Lab del MIT. Este proyecto pretende desarrollar y estudiar la próxima generación de tecnologías innovadoras y materiales curriculares para la educación,

orientados a niños de 5 años en adelante (Flannery et al., 2013). A finales de agosto del 2014 publicaron la primera versión.

En el transcurso del 2012 surge mOway, un pequeño robot autónomo, rápido y provisto de varios sensores. Cuenta con su propio lenguaje de programación llamado MowayWorld, el cual es adecuado para niños mayores de 8 años (Romero, Angulo, Ruíz, & Angulo, 2008). Desafortunadamente, este robot no puede cambiar su forma, lo que lo hace poco flexible.

En abril del 2012 se ofrece en el App Store de Apple la aplicación Move The Turtle de la empresa Next is Great. Es una aplicación para iPhone y iPad que enseña a los niños los fundamentos de la creación de programas informáticos, usando comandos gráficos intuitivos (Next is Great, 2012). Esta aplicación es una versión de LOGO para dispositivos móviles de Apple. Está orientada a niños de 5 en adelante para la programación básica, y a niños entre 9 y 12 años para la programación avanzada.

En junio del 2012 se lanza Kodable, una aplicación para iPad que enseña un nuevo lenguaje de programación a niños de 5 años en adelante (endshelf, 2013; Orsini, 2013). Los creadores de esta aplicación son Grechen Huebner y Jon Mattingly de la empresa SurfScore, Smeeborgian Central Intelligence. Esta aplicación trata de enseñar los conceptos fundamentales de programación y lógica, mediante comandos de arrastrar y colocar para llevar a un personaje de un punto inicial a un punto final (endshelf, 2013; Huebner & Mattingly, 2012).

En agosto del 2013 se ofrece en el AppStore de Apple la aplicación KineScript, es una aplicación para iPad que proporciona un escenario vacío, y actores que se arrastran a la pantalla, a quienes se les dan comportamientos y lógica con el guión, para que luego disfruten de los resultados (Thomas, 2013). Es un lenguaje de programación visual en que los niños pueden aprender a programar y luego compartir sus resultados. KineScript está inspirado en Scratch del MIT.

En setiembre del 2013 se comercializó el ladrillo EV3 de LEGO Mindstorms. El LEGO Mindstorms EV3 combina la versatilidad del sistema de construcción LEGO con la más avanzada tecnología, ofreciendo una interfaz de programación fácil, intuitiva y basada en íconos (LEGO, 2013; Rollins, 2014). Además, en esta versión se puede controlar el ladrillo mediante un control remoto (incluido en el kit) y la aplicación móvil Robot Commander (disponible para dispositivos iOS y Android) (LEGO, 2013).

En ese mismo mes, Dan Shapiro dio a conocer un juego de mesa (inspirado en Logo) que enseña de forma lúdica los fundamentos de programación a niños entre 4 y 8 años (Shapiro, 2013). Este juego trata, mediante tarjetas de comandos y un tablero, de llevar a las tortugas a través de un laberinto. Es decir, los niños dan las órdenes a los adultos (Shapiro, 2013). Este juego estuvo disponible a finales de junio del 2014.

En octubre del 2013, aparece Play-i, proyecto cuyos fundadores y creadores del proyecto son Vikas Gupta, Saurabh Gupta y Mikal Greaves. En septiembre del 2014 cambiaron el nombre a Wonder Workshop. El objetivo de Play-i es que sus dos pequeños robots, Dash y Dot (inicialmente se llamaban Bo y Yana) (Wonder Workshop, 2014a), enseñen a los niños de más de 5 años el lenguaje y los conceptos básicos de la programación a través de juegos interactivos (Gupta, Gupta, & Greaves, 2013). Los dos robots actúan y se ven como juguetes, pero son máquinas de aprendizaje, que guían a los niños a través de códigos básicos. Los niños simplemente deben coreografiar una secuencia de acciones desde su iPad o tableta Android, y los robots los imitan (Wonder Workshop, 2014b).

En noviembre del 2013 aparece el proyecto llamado Primo, de Filippo Yacob, Matteo Loglio, Danilo Di Cuia, Valeria Leonardi, Lucia Rabago y Josh Valman. Primo es una interfaz de programación tangible, para niños entre 3 y 7 años, diseñada para enseñar a programar (Yacob et al., 2013). Cuenta con un robot que usa la tecnología de Arduino.

A finales de enero del 2014, Linda Liukas propone el proyecto llamado Hello Ruby. La metáfora de Hello Ruby es un libro de cuentos (físico y digital) que mediante dibujos y textos sencillos realizados por niños entre 5 y 7 años, les permite aprender a programar (Liukas,

2014). Además, Linda escribió una guía para padres y madres donde puedan aprender a programar junto con sus hijos, y con ello les puedan orientar

En junio del 2014 se propone el proyecto KIBO. KIBO es un kit robótico diseñado específicamente para niños de 4 a 7 años. Los niños construyen sus propios robots con KIBO, los programan para que hagan lo que ellos quieren, y los decoran. KIBO es una interfaz de programación tangible, donde los niños juegan con bloques de madera que traen imágenes de comandos de programación, estos bloques pueden ser compartidos de una forma fácil (Marina Umaschi Bers, 2012b; Lentz, 2014). KIBO es el resultado de investigación de Marina Umaschi Bers y su grupo de investigación DevTech de la Tufts University's Eliot Pearson Department of Child Development. Esta investigación es la unión del software CHERP (2009) (Marina U. Bers, 2010; Marina Umaschi Bers, 2010) y el robot KIWI (2012) (Marina Umaschi Bers, Seddighin, & Sullivan, 2013; Lentz, 2014), ambos son proyectos del grupo investigación DevTech.

En setiembre del 2014 codeSpark da a conocer The Foos, una aplicación para que los niños aprendan a programar, inspirada en lenguajes de programación visual tales como Scratch de MIT, Alicia de CMU y otros lenguajes de programación para principiantes como Logo, SmallTalk y Squeak. Esta aplicación se puede usar en dispositivos móviles y computadoras. Está dirigida a niños de 5 años en adelante. La interfaz es gráfica, y permite arrastrar y soltar bloques en la pantalla para programar. La aplicación móvil está para Android y iOS, y la aplicación Web es para computadoras (codeSpark, 2014).

En ese mismo año (2014), en el mes de agosto, aparece SoRo (Social Robot) Toolkit, proyecto de los investigadores Michal Gordon, Eileen Rivera, Edith Ackermann y Cynthia Breazeal. Social Robot (SoRo) Toolkit ofrece un diseño novedoso que incorpora robots sociales y medios tangibles para que los niños de preescolar (entre 4 y 8 años) exploren conceptos computacionales en el contexto del diseño de reglas de interacción que pueden “enseñar” a un robot social (Gordon, Rivera, Ackermann, & Breazeal, 2015). Para la plataforma robótica, se usa DragonBot (Setapen, 2012), un robot que se controla con una aplicación Android con una interfaz de programación grafica intuitiva, arrastrando y soltando

bloques en la pantalla. El robot aparece como una criatura suave, peluda y de fantasía, diseñada para involucrar a los niños. DragonBot es una plataforma expresiva que tiene un gran repertorio de expresiones faciales, sonidos y acciones. La aplicación móvil le permite al experimentador colocar las reglas creadas por los niños con los componentes tangibles para desencadenar eventos, insertar nuevas reglas y mostrar las reglas aprendidas, y así tele operar al robot.

También, en el año 2014, se da a conocer el proyecto TanProRobot, de los investigadores Danli Wang, Yunfeng Qi y Lan Zhang. TanProRobot es un sistema tangible diseñado para niños entre 6 y 8 años, con el objetivo de que aprendan conceptos de programación. Se compone de tres partes, bloques de programación tangibles (bloques de poder, bloques de sensores, y los bloques de acción), el carro robot (Arduino Mega 2560) y los manipuladores (modelos de casa, túnel, papel de color y baches de velocidad) (Wang, Qi, & Zhang, 2015). Los niños pueden programar el robot para realizar acciones organizando los bloques de programación. El Arduino Mega en el robot actúa como la unidad de procesamiento central. Hay un botón digital en el robot que actúa como interruptor de la fase de ejecución. En fase de implementación, el robot recibe la señal inalámbrica de los bloques de programación. Luego, el Arduino decodifica la señal, controla el mini reproductor MP3 para proporcionar información y, mantiene y almacena las secuencias digitales de las reglas. Después, al entrar en la fase de ejecución, el robot ejecuta las reglas que la función de gestión de eventos. El Arduino hace que los sensores detecten el estado actual y controlen las acciones del robot según el programa construido con los bloques de programación.

A principios de marzo del 2015 se da a conocer el proyecto Teebot, un robot que permite enseñar programación y fundamentos de electrónica y robótica a niños, de los 4 a los 12 años, mientras se divierten jugando (EGM Robotics & Clear Minds, 2015). Teebot fue desarrollado por las empresas EGM Robotics y Clear Minds Consultores. Es un producto diseñado y ensamblado 100% en Ecuador. El niño programa el robot en la computadora a través de un software intuitivo y didáctico, con programación gráfica en la que arrastra bloques que se juntan de forma secuencial para darle instrucciones al robot. El programa creado por el niño

se descarga en el robot, y este ejecuta las órdenes generadas por el niño (EGM Robotics & Clear Minds, 2015). Su pre-venta se inició desde inicios de abril del 2015.

En septiembre del 2015 se da a conocer Robo Wunderkind, un kit robótico orientado a niños de 5 años en adelante, que les permite construir y programar su propio robot. Este kit posee una aplicación móvil para iPad que tiene interfaz de programación gráfica intuitiva, arrastrando y soltando bloques en la pantalla para programar el robot (Robo Wunderkind, 2015).

En la Tabla 1 se resumen las herramientas de programación descritas en cuanto a las edades a las que están dirigidas y las características que poseen, tales como el paradigma de programación, los componentes de programación, si usa o no robots, si es código abierto, y la fuente de enlace, entre otras.

También se han creado juegos que permiten programar. Un ejemplo de esto es el juego Little Big Planet (juego de Playstation 3), un potente emulador de física donde los jugadores pueden crear sus propios escenarios mediante la definición de características físicas de materiales programando de esta manera diferentes objetos.

Estos lenguajes de programación, orientados a los niños, han adquirido gran importancia en las últimas décadas en el proceso de enseñanza-aprendizaje. Esto se debe a la era digital, según Prensky, ya que esta era ha provocado la creación de dos categorías generacionales: los Nativos Digitales y los Inmigrantes Digitales. Los primeros han nacido y crecido en esta era, rodeados de todas las herramientas tecnológicas; mientras que los segundos no nacieron en la era digital, pero adoptan las nuevas tecnologías a sus vidas (Prensky, 2001). Esto ha provocado que los patrones de pensamiento de las nuevas generaciones hayan cambiado y el sistema educativo tradicional no sea el más adecuado para ellos (Prensky, 2001, 2003).

Por su parte, la era digital ha llevado a crear juguetes más útiles y más funcionales gracias a la tecnología. LEGO Mindstorms NXT y EV3 son ejemplo de ello, ya que son juguetes que tienen un ladrillo inteligente que se puede programar desde una computadora, con su propia

herramienta de programación, para crear un robot inteligente. El LEGO Mindstorms es atractivo para todas las edades, y se puede usar para diferentes propósitos dependiendo de las habilidades de los usuarios. También puede comunicarse con diferentes dispositivos a través de Bluetooth. Además, su herramienta de programación visual permite a un programador principiante crear fácilmente un programa. El software de NXT y EV3 se basa en National Instruments' Lab VIEW, que consiste en arrastrar íconos y soltarlos en la pantalla principal del programa (Hamada & Sato, 2010; Kim & Jeon, 2007).

La mayoría de las herramientas de programación citadas, incluido el ambiente de programación del NXT o EV3, tienen la desventaja de que no pueden ser utilizadas por niños que todavía no saben leer, o sea, están diseñadas para niños mayores de 6 años, dejando por fuera a los niños que no han adquirido esta habilidad. Son pocas las herramientas que se han centrado en niños de preescolar. Por otra parte, ninguna de las herramientas de programación mencionadas apoya el proceso de colaboración.

A continuación, se discute un estudio realizado por Common Sense Media sobre el uso de las tecnologías en niños entre 0 y 8 años, donde se muestra el uso de dispositivos móviles en estos niños.

1.2 Uso de Dispositivos Móviles en Niños

La era digital también ha provocado que los dispositivos móviles se encuentren en todas partes. Estos dispositivos son pequeños, accesibles, potentes y personalizables. Pueden comunicarse entre sí por diversas infraestructuras de red y proporcionar aplicaciones integradas utilizadas en la vida cotidiana (Frohberg, Göth, & Schwabe, 2009).

Tabla 1. Resumen de herramientas de programación orientadas a la niñez.

Año	Herramienta de Programación	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Usa Robots	Robots	Código Abierto	Fuente de Enlace	Comunicación (Herramienta y Robot)
1970	Logo	7+	Por procedimientos	Expresiones en procedimientos o funciones	Escritura de líneas de código	Sí	Tortuga robot	No	Computadora	Cable serial
1985	LogoWriter	7+	Por procedimientos	Expresiones en procedimientos o funciones	Escritura de líneas de código	No	---	No	Computadora	---
1988	LEGO TC Logo	7+	Por procedimientos	Expresiones en procedimientos o funciones	Escritura de líneas de código	Sí	LEGO Mindstorms RCX	No	Computadora	Puerto infrarrojo
1993	MicroMundos EX	10+	Por procedimientos	Expresiones en procedimientos o funciones	Escritura de líneas de código	No	---	No	Computadora	---
1994	RoboLab (LabView) y Software LEGO Mindstorms RCX	7+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	Sí	LEGO Mindstorms RCX	No	Computadora	Puerto infrarrojo
1994	LogoBlocks	7+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	No	---	No	Computadora	---
1994	MicroMundos EX Robotics	10+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	Sí	Pico-Cricket kit y LEGO Mindstorms RCX	No	Computadora	Puerto infrarrojo y Cable USB
1994	StarLogo	7+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	No	---	Sí	Computadora	---
1999	Alice	8+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	No	---	Sí	Computadora	---
2000	MicroMundos JR	5-7	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	No	---	No	Computadora	---
2001	Electronic Blocks	4-8	Tangible	Bloques (circuitos electrónicos)	Tangible	Sí	Bloques electrónicos	No	Interfaz tangible	Circuitos electrónicos

Tabla 1. Resumen de herramientas de programación orientadas a la niñez. (cont.)

Año	Herramienta de Programación	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Usa Robots	Robots	Código Abierto	Fuente de Enlace	Comunicación (Herramienta y Robot)
2004	Etoys	8+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	No	---	Sí	Computadora	---
2004	StarLogo TNG	7+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	No	---	Sí	Computadora	---
2005	Software LEGO WeDo	6+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	Sí	LEGO WeDo	No	Computadora	Cable USB
2006	PicoBlocks	7+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	Sí	Pico-Cricket kit	Sí	Computadora	Cable USB
2006	Software LEGO Mindstorms NXT	7+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	Sí	LEGO Mindstorms NXT	No	Computadora	Bluetooth y Cable USB
2007	Scratch	8+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	No	---	Sí	Computadora	---
2007	Tern	5+	Tangible	Piezas de rompecabezas (bloques físicos)	Tangible	Sí	LEGO Mindstorms RCX y iRobot Create	No	Computadora – Interfaz tangible	Técnicas de visión por computadora y Bluetooth
2008	miniBloq	8+	Gráfica (visual) y textual	Bloques y Código textual	<i>Drag and Drop</i> y Código textual	Sí	Kits educativos de robótica RobotGroup	Sí	Computadora	Cable serial y USB
2010	Enchanting	8+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	Sí	LEGO Mindstorms NXT	Sí	Computadora	Bluetooth y Cable USB
2012	Kodable	5-7	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	No	---	No	Computadora (PC y Mac), Web y Móvil (Android y iOS)	---
2012	Move The Turtle	5+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	No	---	No	Móvil (iOS)	---

Tabla 1. Resumen de herramientas de programación orientadas a la niñez. (cont.)

Año	Herramienta de Programación	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Usa Robots	Robots	Código Abierto	Fuente de Enlace	Comunicación (Herramienta y Robot)
2012	Software MowayWorld	7+	Por procedimientos	Escritura de instrucciones / Diagramas de flujo	Escritura de líneas de código	Sí	Robot mOway	No	Computadora	Cable USB, Radiofrecuencia y WiFi
2013	KineScript	8+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	No	---	No	Móvil (iPad)	---
2013	PRIMO	3-7	Tangible	Circuitos electrónicos (Arduino)	Tangible	Sí	Cubetto (Arduino)	No	Interfaz tangible	Wireless
2013	Robot Turtles	4-8	Tangible	Tablero y tarjetas	Tangible	No	---	No	Juego de mesa	---
2013	ScratchJr	5-8	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	No	---	Sí	Computadora – Móvil (Android y iOS)	---
2013	Software LEGO Mindstorms EV3	7+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	Sí	LEGO Mindstorms EV3	No	Computadora	Bluetooth, WiFi y Cable USB
2013	Wonder Workshop	5+	Gráfica (visual)	Íconos	<i>Drag and Drop</i>	Sí	Dash y Dot	No	Móvil (Android y iOS)	Bluetooth
2014	Hello Ruby	5-7	Gráfica (visual)	Libro de imágenes y actividades	Imágenes y actividades	No	---	No	Libro (físico y digital)	---
2014	KIBO (KIWI y CHERP)	4-7	Tangible	Bloques y códigos de barras	Tangible	Sí	KIBO	No	Interfaz tangible	Escáner (lector cód. barras)
2014	SoRo (Social Robot) Toolkit	4-8	Basada en eventos – Tangible	Postales de vinilo, creando reglas mediante plantillas con <i>triggers</i> y acciones	Tangible	Sí	DragonBot	No	Móvil (Android) – Interfaz tangible	Bluetooth

Tabla 1. Resumen de herramientas de programación orientadas a la niñez. (cont.)

Año	Herramienta de Programación	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Usa Robots	Robots	Código Abierto	Fuente de Enlace	Comunicación (Herramienta y Robot)
2014	TanProRobot	6-8	Tangible	Bloques	Tangible	Sí	Robot Car (Arduino)	No	Interfaz tangible	Wireless
2014	The Foos	5+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	No	---	No	PC (Web) – Móvil (Android y iOS)	---
2015	Robo Wunderkind	5+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	Sí	Robot modular	No	Móvil (Android y iOS)	Bluetooth
2015	Teebot	4+	Gráfica (visual)	Bloques / Rompecabezas	<i>Drag and Drop</i>	Sí	Teebot (carro robótico)	No	Computadora	Cable de Red

El estudio realizado por Common Sense Media sobre el uso de las tecnologías en niños entre 0 y 8 años de Estados Unidos presenta entre sus principales conclusiones lo siguiente (Common Sense Media Research Study, 2013):

- El uso de dispositivos móviles por parte de niños entre 0 y 8 años se ha incrementado en los últimos dos años (ver Figura 1).
- Casi el doble de los niños ha utilizado dispositivos móviles en comparación con hace dos años (ver Figura 2). Además, la cantidad de tiempo dedicado al uso de dispositivos móviles en un día típico se ha triplicado, pasando de un promedio de 5 minutos al día en 2011, hasta 15 minutos al día en 2013 (ver Figura 3).
- El tiempo dedicado a los dispositivos “tradicionales” (medios no móviles), como la televisión, DVD, videojuegos y computadoras se ha reducido sustancialmente, por más de media hora al día (ver Figura 3).
- El acceso a los dispositivos y aplicaciones móviles entre los niños de familias de bajos ingresos y de minorías es mucho mayor de lo que era hace dos años (ver Figura 4), pero aún persiste una gran brecha entre ricos y pobres (ver Figura 5).

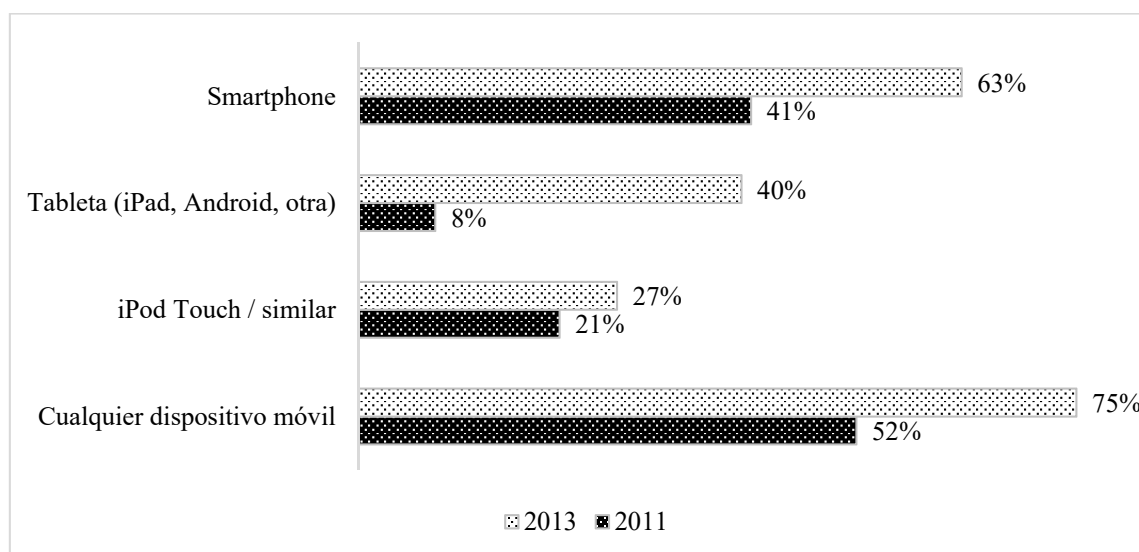


Figura 1. Porcentaje de plataformas móviles que tienen en el hogar niños entre 0 y 8 años (Common Sense Media Research Study, 2013).

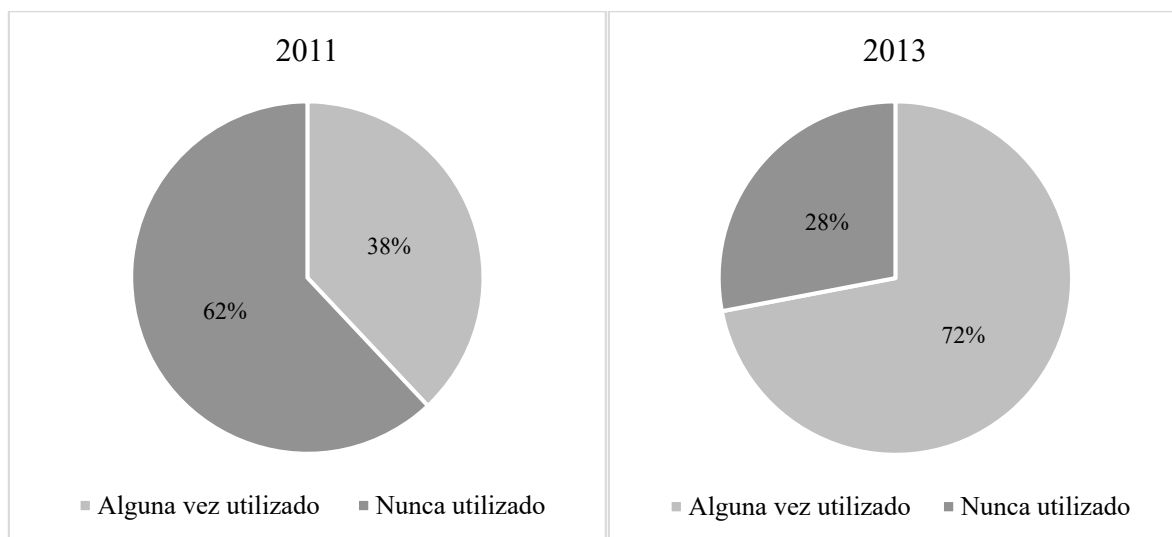


Figura 2. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles (Common Sense Media Research Study, 2013).

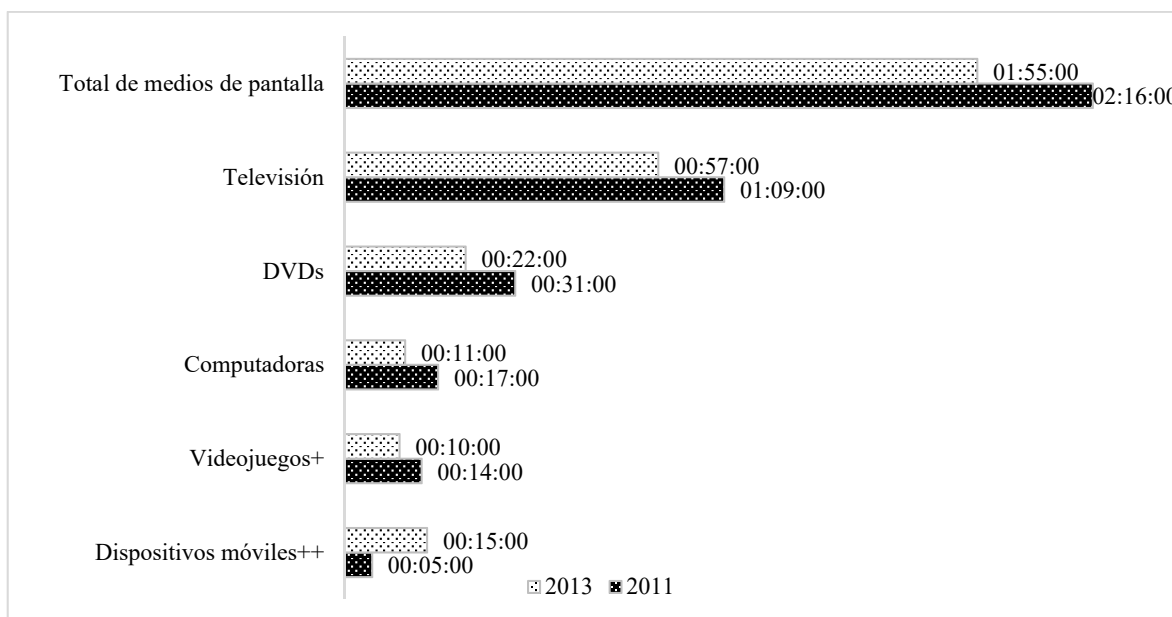


Figura 3. Tiempo promedio por día que usan medios de pantalla los niños entre 0 y 8 años (Common Sense Media Research Study, 2013).

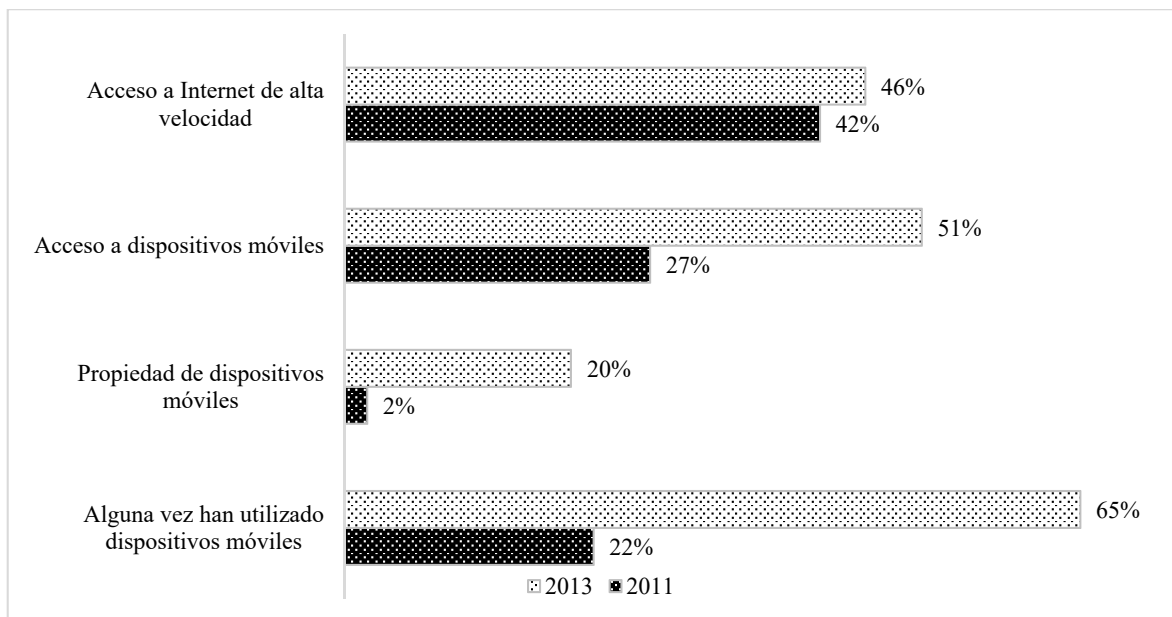


Figura 4. Porcentaje de niños entre 0 y 8 años de familias de bajos ingresos o de minorías que usan dispositivos móviles (Common Sense Media Research Study, 2013).

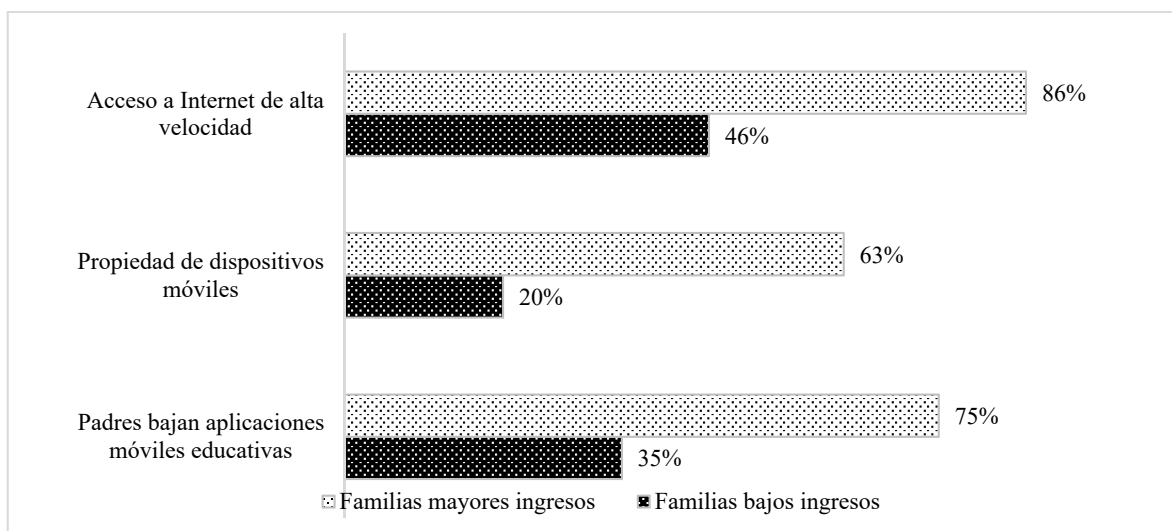


Figura 5. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles según el tipo de ingresos de sus familias (Common Sense Media Research Study, 2013).

Por otro lado, este estudio muestra las actividades en que los niños entre 0 y 8 años usan los dispositivos móviles. En la Figura 6 se muestra las actividades que se usan en 2011 y 2013. En la Figura 7 se muestra las actividades según la plataforma móvil, y en la Figura 8 el tipo de aplicaciones móviles usadas según el rango de edades.

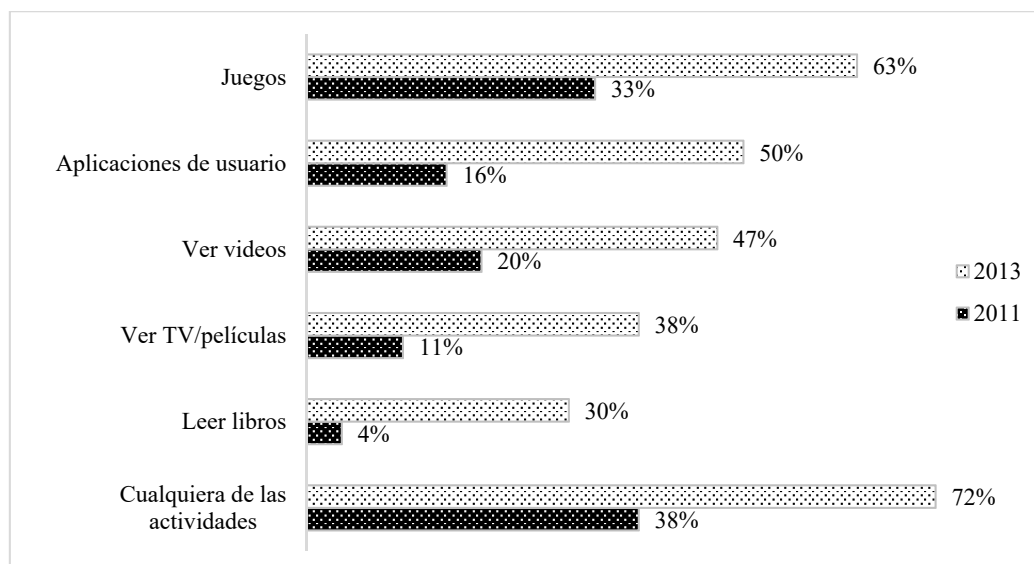


Figura 6. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles para actividades específicas (Common Sense Media Research Study, 2013).

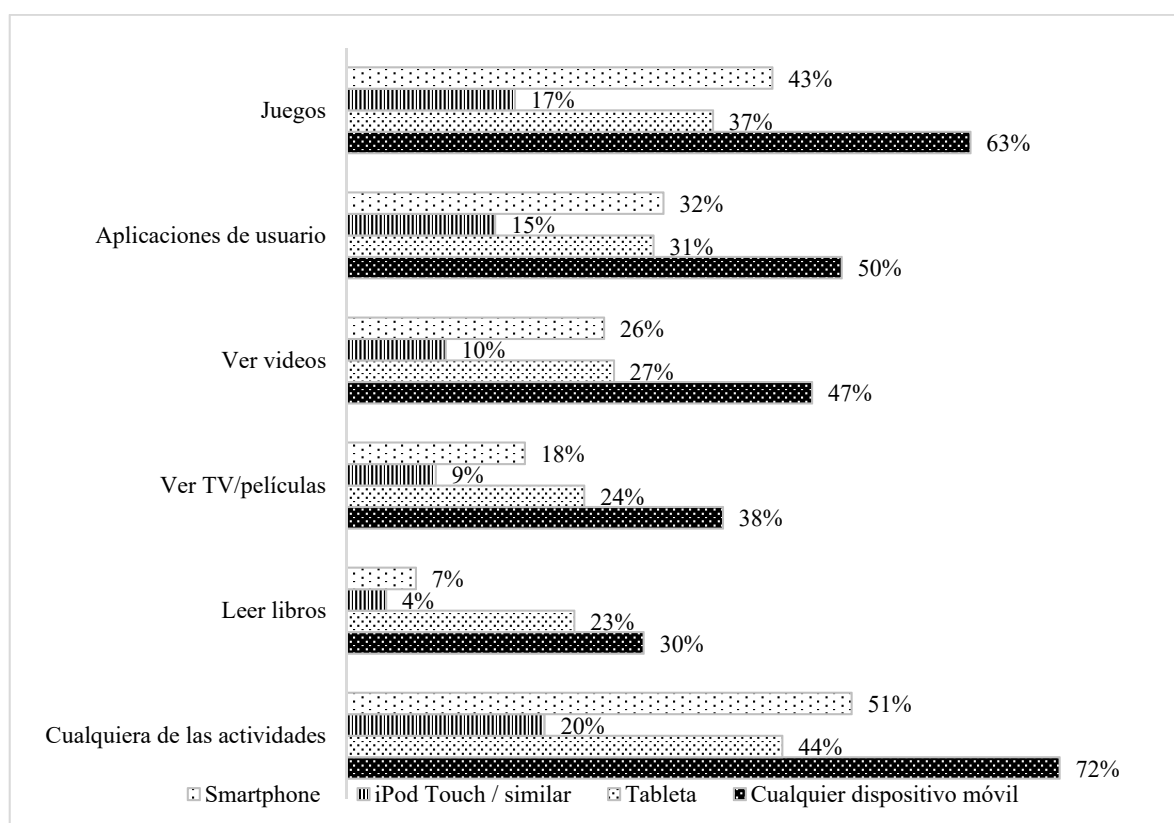


Figura 7. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles para actividades específicas por plataforma (Common Sense Media Research Study, 2013).

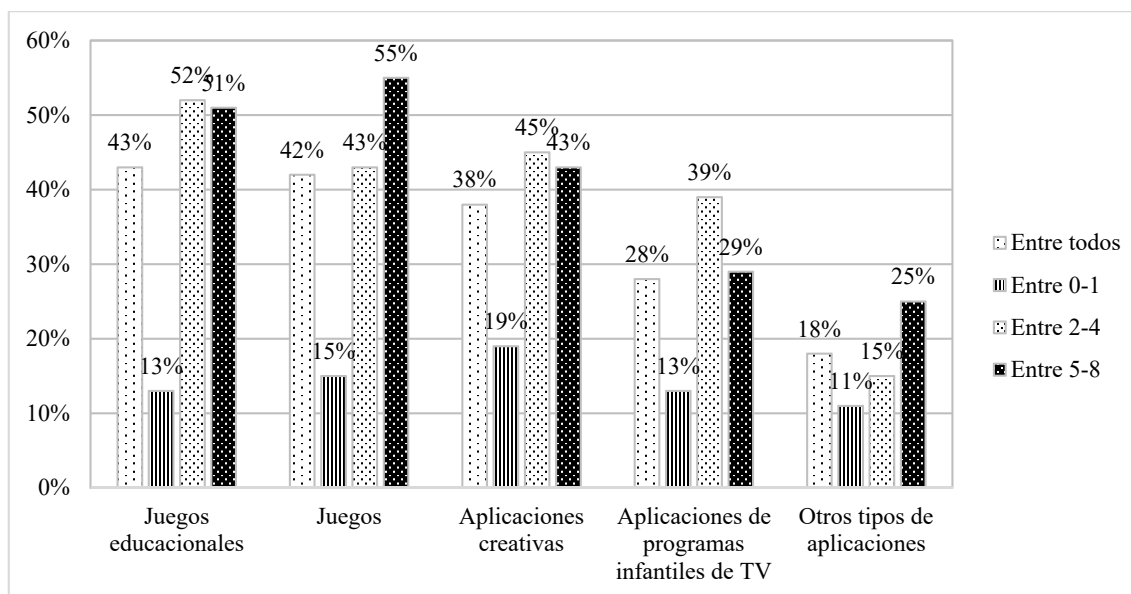


Figura 8. Porcentaje de niños entre 0 y 8 años que usan dispositivos móviles para diferentes tipos de aplicaciones móviles por rango de edades (Common Sense Media Research Study, 2013).

En Costa Rica, los investigadores Ana María Carmiol y Rolando Pérez, del Instituto de Investigaciones Psicológicas de la Universidad de Costa Rica, están llevando a cabo el proyecto: “El andamiaje parental durante el visionado televisivo conjunto entre madres y sus hijos/as en edad preescolar: Caracterización y efecto en la comprensión del contenido televisivo de los niños y las niñas” (Carmiol & Pérez, 2015a).

Como parte de este proyecto se realizó el estudio: “Uso de TICs en niñas y niños preescolares”, para conocer cómo los niños y las niñas entre 3 y 5 años usan la televisión, las tabletas, el teléfono celular o la computadora. Para esto, realizaron un cuestionario dirigido a los padres de familia de niños entre 2 y 5 años entre mayo y junio del 2015 (URL: <http://cuestionarios.iip.ucr.ac.cr/index.php/248573/lang-es>).

Este estudio tuvo una muestra de 290 individuos, de los cuales 130 (45%) eran niñas y 148 (51%) niños con edades entre 2 años y 2 meses, y 8 años y 1 mes (la edad promedio fue de 5 años y 1 mes) (Carmiol & Pérez, 2015b). Entre los resultados preliminares de este estudio se muestra el tiempo promedio diario (en minutos) que una niña o un niño ve televisión, usa la

tableta o celular, o usa la computadora (ver Figura 9). Además, la preferencia en el uso de cierto tipo de aplicaciones mediante la tableta o el celular (se utilizó una escala Likert de 5 para medir la preferencia) (ver Figura 10).

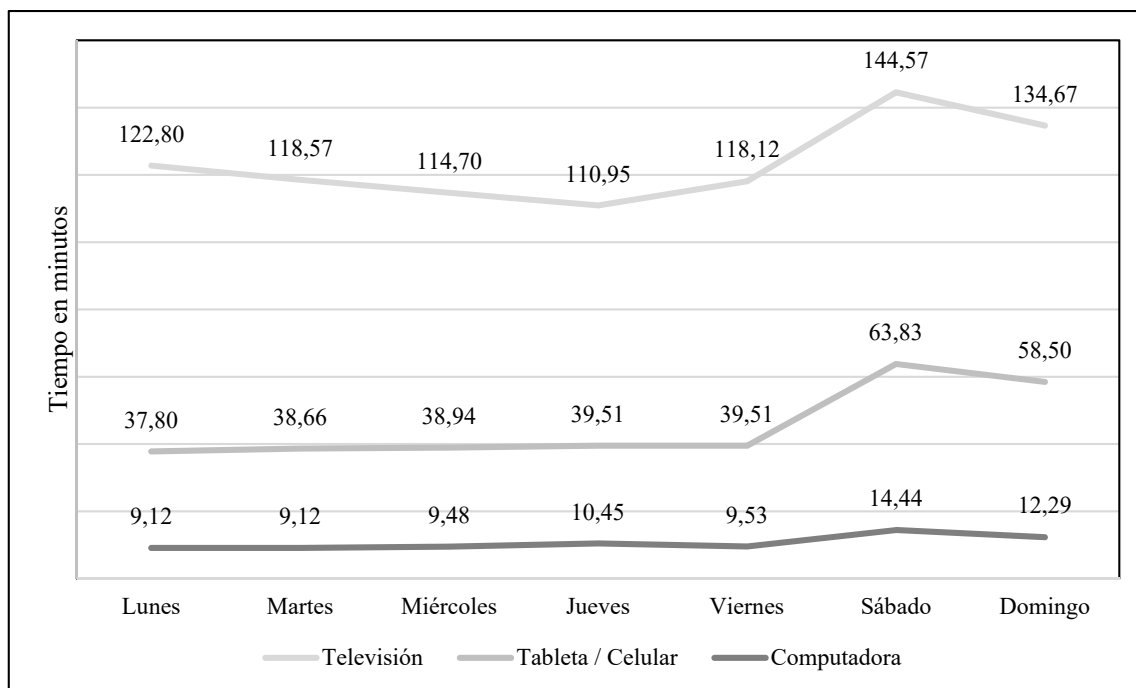


Figura 9. Tiempo en minutos del uso de medios (Carniol & Pérez, 2015b).

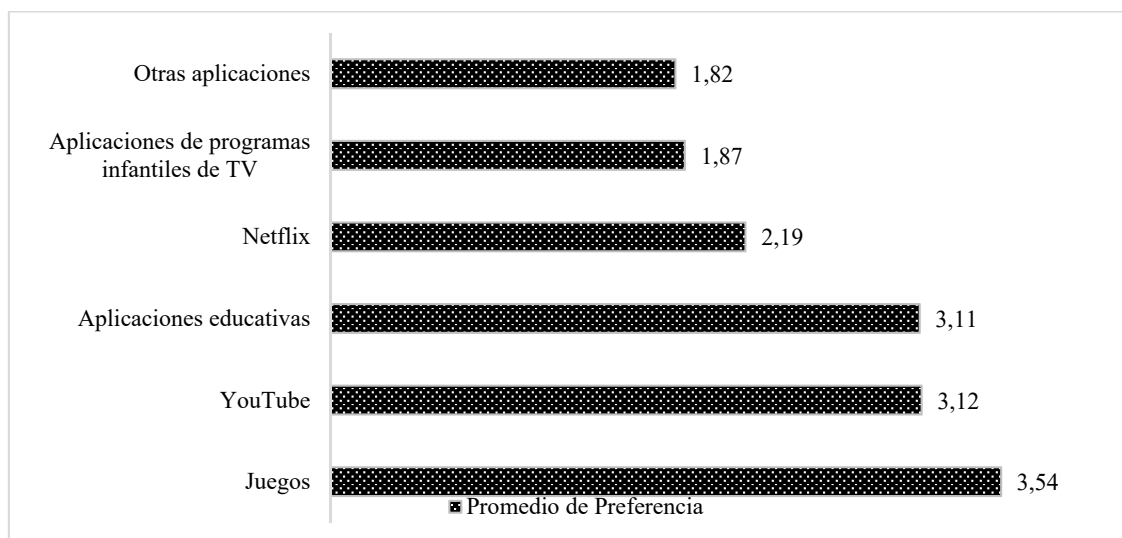


Figura 10. Preferencia del uso de aplicaciones mediante tableta o celular (Carniol & Pérez, 2015b).

Todo esto ha llevado a diferentes investigadores estudiar la forma en que los dispositivos móviles pueden ser integrados en los sistemas de educación y sus efectos sobre los procesos de aprendizaje. Esto porque la actitud de los nativos digitales hacia estas tecnologías móviles es interesada, competitiva, cooperativa, y orientada a los resultados (Prensky, 2001, 2003). Esta es, precisamente, la actitud que muchos docentes desearían en los salones de clases.

Lo que atrae a los nativos digitales a estas tecnologías, en opinión de Prensky, es el aprendizaje que ofrecen los juegos de vídeo y computador, ya que a los niños les encanta aprender constantemente y sobre todo cuando se les imponen retos. Las herramientas de la era digital ofrecen oportunidades de aprendizaje en cada instante (Prensky, 2003).

Prensky propone cambiar el proceso de enseñanza-aprendizaje reconsiderando cambiar la metodología y el contenido. En primer lugar, la metodología debe cambiar en la manera de interactuar con los estudiantes y de transmitir los conocimientos: ir más rápido, menos paso a paso, en paralelo, con acceso aleatorio, entre otras cosas. En segundo lugar, qué contenido se debe enseñar, existiendo dos tipos: contenido antiguo (de siempre: matemática, español, ciencias, entre otros) y el contenido futuro (lógico, digital y tecnológico, como robótica) (Prensky, 2001, 2003). Además, se debe pensar en cómo enseñar ambos tipos de contenido a los nativos digitales, donde el primero implica un cambio de la metodología, y el segundo consiste en agregar nuevos contenidos (Prensky, 2001).

A medida que se avanza hacia un mundo donde la tecnología controla cada vez más nuestras vidas, el incorporar la habilidad de programar en el proceso de enseñanza-aprendizaje desde temprana edad se vuelve importante (Prensky, 2001; Mitchel Resnick, 1998). Esto ayuda a expandir la mente y a pensar de una forma más ordenada. La programación es una plataforma para mostrar creatividad, especialmente en la resolución de problemas, y forma parte importante del pensamiento computacional, tema a tratar en la siguiente sección.

1.3 Pensamiento Computacional

En el año 2006, Jeannette Wing publicó un artículo donde define pensamiento computacional (*computational thinking*). En este artículo explica que esta nueva competencia debería ser incluida en la formación de los niños, ya que representa un ingrediente vital del aprendizaje de la ciencia, la tecnología, la ingeniería y las matemáticas (STEM) (Wing, 2006).

En palabras de la propia Wing: “el pensamiento computacional implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, haciendo uso de los conceptos fundamentales de la informática” (Wing, 2006, 2008). Es decir, la esencia del pensamiento computacional es pensar como lo haría un científico computacional cuando se enfrenta a un problema de la vida real, tiene aplicaciones más allá de la computación (Denning, 2009; Hambruch, Hoffmann, Korb, Haugan, & Hosking, 2009). El pensamiento computacional es considerado como una habilidad fundamental de la vida (Wing, 2006).

Otras definiciones de pensamiento computacional han surgido en la literatura científica desde entonces. Entre las más aceptadas se encuentran las siguientes:

- El pensamiento computacional es el proceso que permite formular problemas de forma que sus soluciones pueden ser representadas como secuencias de instrucciones y algoritmos (Aho, 2012).
- El pensamiento computacional es el proceso de reconocimiento de aspectos de la computación en el mundo que nos rodea, y aplicar herramientas y técnicas computacionales para comprender y razonar sobre los sistemas y procesos tanto naturales como artificiales (Henderson, Cortina, Hazzan, & Wing, 2007).

Aunque el término está abierto a cualquier interpretación, los principios fundamentales generalmente aceptados del pensamiento computacional son los siguientes (Henderson et al., 2007):

- **Abstracción.** Las soluciones a menudo no son sencillas y requieren múltiples niveles de pensamiento y aplicación.
- **Análisis lógico.** Usa la mayor parte de la información que se tenga para deducir soluciones.

- **Pensamiento algorítmico.** La solución a un problema a menudo tiene varios pasos, incluso repeticiones, requiere estrategia y formulación de un conjunto de reglas.
- **Eficiencia.** Los recursos con los que se cuentan, a menudo se refieren al tiempo necesario para resolver un problema.
- **Innovación.** La observación del mundo y ver que las cosas se pueden mejorar, pueden dar lugar a importantes avances.

Una iniciativa interesante en relación con la definición del pensamiento computacional es promovida por la International Society for Technology in Education (ISTE) y la Computer Science Teacher Association (CSTA). Estas instituciones han colaborado con líderes del mundo de la investigación y la educación superior, la industria y la educación primaria y secundaria para desarrollar una definición operativa que describa con precisión las características esenciales y ofrezca un marco de trabajo y un vocabulario común con el que los profesionales de la educación puedan trabajar (International Society for Technology in Education & Computer Science Teachers Association, 2011). Según esta definición operativa, el pensamiento computacional es un proceso de resolución de problemas que incluye las siguientes características (International Society for Technology in Education & Computer Science Teachers Association, 2011):

- Formular problemas de forma que se permita el uso de una computadora y otras herramientas para ayudar a resolverlos.
- Organizar y analizar lógicamente la información.
- Representar la información a través de abstracciones como los modelos y las simulaciones.
- Automatizar soluciones haciendo uso del pensamiento algorítmico (estableciendo una serie de pasos ordenados para llegar a la solución).
- Identificar, analizar e implementar posibles soluciones para lograr la combinación más efectiva y eficiente de pasos y recursos.
- Generalizar y transferir este proceso de resolución de problemas para ser capaz de resolver una gran variedad de familias de problemas.

Estas habilidades son apoyadas y reforzadas por una serie de disposiciones o actitudes que son dimensiones esenciales del pensamiento computacional, las cuales incluyen (International Society for Technology in Education & Computer Science Teachers Association, 2011):

- Confianza al tratar con la complejidad.
- Persistencia en el trabajo con problemas difíciles.
- Tolerancia para la ambigüedad.
- Capacidad de hacer frente a los problemas de composición abierta.
- Capacidad de comunicarse y trabajar con otros para lograr un objetivo común o solución.

Cuando se piensa en el pensamiento computacional, se piensa en el contexto del aprendizaje de la programación. Los lenguajes de programación son una combinación formal de estructuras y comandos que permiten crear algoritmos (cualquier conjunto de instrucciones paso a paso) que se ejecutan. Pero el pensamiento computacional se refiere más a la capacidad para dividir de manera efectiva un problema y diseñar una solución. La construcción de un programa de computadora conlleva la dificultad en cómo se debe dividir en tareas individuales el problema dado y diseñar para cada una su propia solución (Wing, 2008).

Por los argumentos anteriores, la razón para centrarse en la habilidad del pensamiento computacional en este trabajo radica en su importancia en el campo de la computación y su aplicación a literalmente en cualquier otro campo (Hambruch et al., 2009; Landau et al., 2013).

A continuación, se discute sobre la importancia de la programación y la robótica en la educación, como recursos para desarrollar en los niños el pensamiento computacional.

1.4 Importancia de la Programación y la Robótica en la Educación

En la actualidad la tecnología es parte de casi de todas las actividades del hombre en su vida cotidiana. Los conceptos de áreas específicas, tales como la ingeniería y la ciencia, ahora están siendo utilizados al realizar actividades cotidianas. La incorporación de actividades que promueven estas habilidades en el proceso de aprendizaje desde la primera infancia podría ayudar a los niños a desarrollar el pensamiento lógico-matemático, la abstracción y el pensar de una forma más ordenada (Papert, 1980; Verner & Ahlgren, 2004). Papert describe la importancia de la programación como una herramienta para reflexionar sobre el propio pensamiento, ya que desarrolla un amplio conjunto de capacidades interconectadas como articulación de problemas, trabajo en equipo, persistencia y otras habilidades esenciales en la vida (Papert, 1980). Explica además cómo las computadoras pueden ayudar a los niños a aprender activamente y a crear conocimiento.

Papert introduce el concepto de construccionismo, el cual define como una extensión del concepto de constructivismo de Jean Piaget. El construccionismo es una teoría del aprendizaje atribuido principalmente a la investigación psicológica del desarrollo, y se centra en cómo las interacciones con las tecnologías pueden promover el desarrollo social y cognitivo, proponiendo que la tecnología se combine con el constructivismo. Esto permite a los estudiantes oportunidades para diseñar, construir y programar proyectos de significado personal, lo que añade un componente afectivo (Papert, 1980).

Por otro lado, Bers afirma que el uso de la robótica puede ser una poderosa herramienta práctica en niños pequeños para el aprendizaje de conceptos STEM. Bers hace un paralelismo entre el uso de la robótica durante los primeros años de escolaridad y el desarrollo de las habilidades que los estudiantes necesitan en el siglo XXI (Marina Umaschi Bers, 2008). Estas habilidades serán críticas para el éxito en el futuro de los nativos digitales: pensamiento creativo, claridad en la comunicación, análisis sistemático, colaboración efectiva, diseño reiterativo y aprendizaje continuo.

El uso de la robótica permite a los niños explorar conceptos complejos de una manera concreta y divertida, involucra habilidades sensomotoras y socioemocionales fundamentales para el sano desarrollo de los niños pequeños, proporciona el ambiente de aprendizaje libre de ideales y el uso principal de la curiosidad natural de los niños para aprender a construir y programar (Marina Umaschi Bers, 2008, 2012a).

Por otra parte, diversos autores coinciden en que las habilidades básicas que se desarrollan con la participación en cursos o talleres de robótica, tales como el diseño, la construcción, y optimización de modelos robóticos, permiten adquirir habilidades fundamentales de Ingeniería, que a su vez inciden en el logro de habilidades en otras áreas como lenguajes matemáticos, análisis de datos y trabajo en equipo (Verner & Ahlgren, 2004).

Asimismo, en diversas latitudes, particularmente en países desarrollados, se ha utilizado la robótica y la programación para el logro de estándares de aprendizaje en ciencia, ingeniería y matemática, áreas identificadas como claves para lograr el desarrollo tecnológico y educativo de los países (Caron, 2010; Colelli, 2009; Ivey & Quam, 2009). En ese sentido, también se ha evidenciado que muchos estudiantes que terminan programas en robótica continúan con una carrera en ingeniería (Caron, 2010).

Algunas de las habilidades que promueve la participación en programas de robótica se refieren a pensamiento crítico y resolución de problemas, colaboración y liderazgo, agilidad y adaptabilidad, iniciativa y emprendedurismo, acceso y análisis de información, curiosidad e imaginación (Ivey & Quam, 2009).

Todo lo anterior redundaría en la posibilidad de que los niños y los jóvenes puedan desarrollar las competencias claves para alcanzar el éxito en la vida profesional y personal. Según el modelo de las habilidades para el siglo XXI, esas habilidades se refieren al pensamiento superior o sofisticado, la resolución flexible de problemas, y habilidades de comunicación y colaboración, todo ello a través del uso y la apropiación de las tecnologías (Binkley et al., 2012).

En resumen, la codificación, es decir, la creación de programas, es considerada la nueva alfabetización (Marina Umaschi Bers, Flannery, Kazakoff, & Sullivan, 2014). Además, la robótica combina las posibilidades físicas y creativas de construir y manipular objetos, con las experiencias de resolución de problemas y la colaboración, para aprender a programar los objetos y moverlos como se desee.

En Costa Rica, se ha observado que niños, niñas y adolescentes que participan en programas de robótica adquieren habilidades referidas a la resolución de problemas, construcción de sistemas mecánicos y la construcción y control de prototipos robóticos de invención propia (Acuña-Zúñiga, 2006b). Asimismo, desarrollan habilidades en áreas sociales como el trabajo en equipo, la comunicación de procesos de pensamiento y la divulgación de resultados, todas habilidades que buscan superar la brecha digital y generar intereses e inquietudes que en el mediano y largo plazo puedan materializarse en el desarrollo profesional de las nuevas generaciones (Castro-Rojas & Acuña-Zúñiga, 2012).

A continuación, se discute sobre la importancia del aprendizaje colaborativo en las actividades de enseñanza-aprendizaje desde edades tempranas.

1.5 Importancia del Aprendizaje Colaborativo en la Educación

La colaboración es esencial en la educación, puesto que se trata de que los hombres y las mujeres, los niños y las niñas, aprendan a trabajar conjuntamente para un mismo fin (Vidanes-Díez, 2007; Zurbano Díaz de Cerio, 1998).

Para sentar la base del aprendizaje de la colaboración en los niños, es necesario enseñarlos a realizar muchas y variadas acciones, por ejemplo: socorrer a otro, saber coordinar acciones para realizar un trabajo conjunto, prestar ayuda a otros para realizar un fin común, sentir alegría colectiva por el logro de un resultado, defender a los compañeros y ayudarlos cuando tengan dificultades (Zurbano Díaz de Cerio, 1998).

La colaboración y ayuda mutua ha de enseñarse desde las primeras edades, porque el niño, por su propio egocentrismo, tiende a quererlo todo para sí y no compartir con nadie, pero cuando empieza a realizar acciones a favor de los demás, su egocentrismo inicial cede el paso a la conducta gregaria y cooperadora (Zurbano Díaz de Cerio, 1998).

A partir de aquí nace la importancia de incorporar el aprendizaje colaborativo en el proceso de enseñanza-aprendizaje, y de hacerlo desde edades tempranas. Además, el aprendizaje colaborativo ofrece muchas ventajas, entre las que se mencionan las siguientes (D. W. Johnson & Johnson, 1997; McConnell, 1999):

- Promueve el proceso enseñanza-aprendizaje, concebido como un proceso activo, producto de la interrelación del aprendiz con su medio. Se da la construcción de conocimiento, ya que obliga a activar el pensamiento individual, a buscar formas de investigar, sea en forma independiente o en grupo, y promueve valores en forma semiconsciente, como la cooperación, la responsabilidad, la comunicación, el trabajo en equipo, la autoevaluación y la evaluación de los compañeros.
- Propicia un ambiente para la comunicación y discusión productiva cuando se logra una interdependencia óptima entre propósitos, sistemas y equipos disponibles. Se genera un lenguaje común, pues se establecen normas de funcionamiento grupal y se disminuye el temor a la crítica y a la retroalimentación. Con esto disminuyen también los sentimientos de aislamiento y gracias a ello puede darse una mejora de las relaciones interpersonales entre diferentes culturas, profesiones y etnias, entre otros.
- Aumenta la motivación por el trabajo, al propiciarse una mayor cercanía y apertura entre los miembros del grupo, incrementando la satisfacción y el aumento de la productividad.
- Impulsa el desarrollo de habilidades sociales, al exigir la aceptación de otra persona como cooperante en la labor común de construir conocimientos y al valorar a los demás como fuente para evaluar y desarrollar nuevas estrategias de aprendizaje.
- Genera una interdependencia positiva, abarcando las condiciones organizacionales y de funcionamiento que deben darse en el interior del grupo. Los miembros del equipo se necesitan unos a otros y confían en el entendimiento y éxito de cada u.

- Promueve la interacción de las formas y del intercambio verbal entre las personas del grupo, lo que afecta finalmente los resultados del aprendizaje. En la medida en que se posean diferentes medios de interacción, el grupo podrá enriquecerse, aumentar sus refuerzos y retroalimentarse.
- Valora la contribución individual, dado que cada miembro del grupo asume íntegramente su responsabilidad en la tarea, a la vez que al socializar recibe las contribuciones del grupo.
- Estimula habilidades personales y de grupo al permitir que cada miembro participante, desarrolle y potencie las habilidades personales y grupales como escuchar, participar, liderar, coordinar actividades, realizar seguimiento y evaluar.
- Propicia que el estudiante se mantenga activo y atento a lo que se discute o comenta dentro del equipo; ya que crea sinergia al aprovechar el conocimiento y experiencia de los miembros, según su área de especialización y diversos enfoques o puntos de vista. Logrando una visión completa del estudio a realizar mejorando la calidad de las decisiones y de los productos obtenidos.
- Presenta la posibilidad de realizar procesos en paralelo en función de la división del equipo en comisiones o subgrupos, que permiten aligerar la presentación de la solución, producto u opinión, según la situación para la cual trabajan.
- Asegura la calidad, confiabilidad y exactitud de las ideas y soluciones planteadas, al extraer el máximo provecho de las capacidades individuales para beneficio del grupo.
- Permite el logro de objetivos cualitativamente más ricos en contenidos (en relación al conocimiento). Esto se debe a que, al conocer diferentes temas y adquirir nueva información, se reúnen propuestas y soluciones de personas con diferentes puntos de vista, lo que permite valorar las distintas maneras de abordar y solucionar un problema, las diversas formas de comprenderlo y las diferentes estrategias para manejar información proveniente de una amplia gama de fuentes.
- Obliga a la autoevaluación del grupo, ya que exige evaluar lo realizado por sus integrantes en la consecución de los objetivos.

Cabe destacar que las ventajas del aprendizaje colaborativo se ubican en dos áreas: la cognitiva y la socio-afectiva. En sus inicios, muchas investigaciones reportaron solo ventajas socio-afectivas, por ejemplo: la mejora de las relaciones sociales, el aumento de la tolerancia, de la integración, la cohesión grupal, y del control individual derivadas de la interacción social (D. W. Johnson & Johnson, 1997, 1999; D. W. Johnson, Johnson, & Holubec, 1998). En recientes estudios, se encuentran los siguientes beneficios en el dominio cognitivo (McConnell, 1994):

- Ayuda a clarificar las ideas.
- Proporciona oportunidades para que los estudiantes adquieran información e ideas.
- Desarrolla destrezas de comunicación.
- Provee de un contexto en el que el estudiante toma el control de su propio aprendizaje en un contexto social.
- Valida las ideas individuales.

Por último, el aprendizaje nunca se produce en aislamiento, sino que se produce a partir del inter-juego dinámico de individuos (Williams & Burden, 1999). Es en esta perspectiva pedagógica donde el concepto de aprendizaje colaborativo toma real importancia, ya que este implica el trabajo conjunto de individuos para alcanzar objetivos de aprendizaje (Nunan, 1993).

En la sección siguiente se discute sobre los beneficios que la tecnología puede brindar al proceso de colaboración.

1.6 Colaboración en Ambientes de Aprendizaje Apoyados por Computadora

El aprendizaje colaborativo apoyado por computadora (ACAC, en inglés *Computer Supported for Collaborative Learning* (CSCL)) es un área emergente de las ciencias del aprendizaje referente a estudiar como las personas pueden aprender de manera conjunta con la ayuda de los computadores (Collazos, Muñoz, & Hernández, 2014, p. 23).

Kaye resume los seis elementos más importantes para el aprendizaje colaborativo (Kaye, 1992):

- El aprendizaje es inherentemente un proceso individual, no colectivo, y está influido por una variedad de factores externos, entre ellos el grupo y las interacciones interpersonales.
- Las interacciones interpersonales y grupales están relacionadas con el uso de un lenguaje en la reorganización y modificación de acuerdos comunes, y para la construcción de estructuras de conocimiento individuales. Por tanto, el aprendizaje es un fenómeno tanto privado como social.
- El aprendizaje colaborativo implica compartir conocimientos, interactuar e intercambiar roles.
- La colaboración consiste en la sinergia y se supone que de alguna manera “el todo es mayor que la suma de las partes individuales”, esto significa que el aprendizaje colaborativo puede producir ganancias superiores al aprendizaje individual.
- No todos los intentos de aprendizaje de manera colaborativa han tenido éxito. Bajo ciertas circunstancias, el aprendizaje colaborativo significa pérdida del proceso, falta de iniciativa, malentendidos, conflictos y descrédito: los beneficios potenciales no siempre se alcanzan.
- El aprendizaje colaborativo no significa necesariamente aprender dentro de un grupo. Implica la posibilidad de contar con el apoyo de otras personas, y dar retroalimentación en un ambiente no competitivo.

Las propuestas para el uso de técnicas de aprendizaje colaborativo apoyado por computadora argumentan que, bajo la práctica de actividades grupales, los estudiantes tienen la oportunidad de aprender en los procesos de grupo, aprender estrategias personales en contraste con otras personas, múltiples perspectivas de un mismo tema, liderazgo, gestión de los grupos de tareas y comunicación. El apoyo de una computadora en este proceso es compatible con la comunicación y la colaboración (proporcionando asistencia al proceso de coordinación, y permitiendo la creación de situaciones imposibles en el mundo real), y el seguimiento de las acciones de los miembros del grupo (que se puede usar para la mejora de

las estrategias de solución de problemas y la estimulación de los procesos mentales de mejora de la adquisición de conocimientos). A pesar de esto, algunos autores mencionan problemas o resultados inesperados de experiencias con el uso de entornos de aprendizaje colaborativo apoyado por computadora.

Como Santoro y otros mencionan, el problema de la falta o del bajo nivel de la colaboración dentro de los ambientes puede ocurrir por las siguientes razones o una combinación de ellos (Santoro, Borges, & Santos, 1999):

- **Cultura.** Una de las dificultades que determinan los resultados negativos en el uso de entornos de aprendizaje colaborativos apoyados por computadora, es que la gente no está acostumbrada a trabajar en grupos. Por esta razón, uno de los objetivos de este tipo de ambiente es el desarrollo de esta capacidad, además de la adquisición de algún contenido. Además, aspectos relacionados con la cultura de las personas, tales como factores de comportamiento, la edad, el sexo y los antecedentes, podrían influir en la forma en cómo va a ser utilizado un entorno. Por ejemplo, en una cultura donde la disciplina es floja, no se puede esperar que los estudiantes hagan sus tareas sin necesidad de usar una cierta presión para obtener resultados positivos.
- **Estímulo.** Algunos ambientes de aprendizaje colaborativo ofrecen exclusivamente herramientas para dar apoyo a la ejecución de una tarea por un grupo de personas. El fracaso de estos ambientes se justifica por la ausencia de mecanismos que favorecen las funciones de un grupo como actividad cognitiva, el apoyo a las personas y el bienestar del grupo. Se propone un objetivo, se brindan herramientas colaborativas y se supone que los estudiantes interactúan colaborativamente para alcanzar el objetivo, a pesar de que el medio ambiente no induce a esto, lo que significa que no hay flujo o proceso definido y la única manera de lograr resultados es a través del trabajo en equipo.
- **Contexto.** Los ambientes de aprendizaje colaborativo apoyados por computadora suelen tener objetivos educativos específicos o metas de formación práctica de organización, pero además de ellos, deben ser integrados dentro de otras actividades en las que los estudiantes participen. Por ejemplo, seguramente será difícil usar un

ambiente de colaboración en un contexto donde la competencia sustenta las relaciones entre las personas. El medio ambiente debe motivar los intereses comunes dentro del grupo, por lo general relacionados con su vida cotidiana, sus necesidades y preferencias.

- **Tecnología.** No hay integración de herramientas dentro de los entornos. En general, la gente tiene dificultades en el uso de varias herramientas diferentes. El problema de la integración podría resolverse mediante la recopilación de los objetivos de aprendizaje y las acciones en el medio ambiente. Además, las interfaces todavía no son cómodas, y no exploran mecanismos específicos para el trabajo colaborativo.

En los últimos años algunas investigaciones se han hecho sobre el estudio de los procesos de colaboración y cómo mejorarlos (Barros & Verdejo, 1999; Brna & Burton, 1997). Algunos de estos estudios se centran en la forma de enseñar-aprender a colaborar, poniendo énfasis en “aprender a colaborar” en lugar de “colaborar para aprender”.

La mayoría de las investigaciones sobre el proceso de colaboración han tratado de medir sus efectos, en general, a través de algún pre-test o pos-test ganancia individuo con respecto a la ejecución de tareas. Algunos efectos específicos se han descrito en términos de cambio conceptual (Amigues & Caillot, 1990) o el aumento de la autorregulación. La elección de estas variables dependientes lleva a dos preguntas metodológicas. La primera pregunta podría ser declarada como “¿efectos de qué?”. Una situación de aprendizaje colaborativo incluye una variedad de contextos e interacciones. Hablar sobre el efecto de tal término en sentido amplio tendría tan poco sentido como hablar sobre el beneficio de tomar un medicamento, sin especificar cuál. No hay que hablar de los efectos de aprendizaje colaborativo en general, sino sobre los efectos de determinadas categorías de interacción (Dillenbourg et al., 1995). La segunda pregunta se refiere a la modalidad de la evaluación. Los efectos de aprendizaje colaborativo a menudo son evaluados por las distintas medidas de desempeño de tareas. Se ha objetado que una evaluación válida sería medir el desempeño del grupo (Collazos, 2003). Esta validez puede ser entendida en términos prácticos: más profesionales tienen que colaborar y es un objetivo importante para cualquier institución educativa para mejorar el rendimiento de los estudiantes en situaciones de colaboración.

A continuación, se describe el problema de esta investigación.

1.7 Problema

A medida que se avanza hacia un mundo donde la tecnología se inserta cada vez más en nuestras vidas, el incorporar la habilidad de programar en el proceso de enseñanza-aprendizaje desde temprana edad se vuelve importante, porque ayuda a desarrollar el pensamiento lógico, la abstracción y el pensar de una forma ordenada (Papert, 1980; Verner & Ahlgren, 2004). La programación se convierte en una plataforma para mostrar la creatividad, especialmente en la solución de problemas, lo que ayuda a los aprendices, consecuentemente, a desarrollar otras habilidades.

Entre más temprano los niños aprendan a programar, más rápido desarrollan competencias para diseñar y resolver problemas fundamentales para el éxito académico futuro, y comienzan a usar conceptos matemáticos y destrezas del lenguaje en un contexto significativo.

Resnick dice que la programación es la nueva alfabetización, y agrega que, al igual que la escritura, la programación ayuda a organizar el pensamiento y a expresar las ideas. En el pasado se pensaba que la programación era difícil para la mayoría de las personas, pero gracias al avance de la tecnología es posible crear herramientas de programación accesibles para todas las personas (Mitchel Resnick, 1998).

Por otra parte, existen varios investigadores (Acuña-Zúñiga, 2006a, 2006b; M. Bers et al., 2006; Marina Umaschi Bers, 2012a, 2012b; Castro-Rojas & Acuña-Zúñiga, 2012; J. Johnson, 2003; Kazakoff, Sullivan, & Bers, 2012) que han estudiado la utilización de la robótica en el proceso de enseñanza-aprendizaje de los niños, logrando que los niños programen y construyan robots para alcanzar contenidos curriculares específicos. Dichos estudios han generado buenos resultados.

Áreas como Ciencia, Tecnología, Ingeniería y Matemática (conceptos STEM) pueden resultar complejos. Sin embargo, el aprendizaje de estas áreas durante la primera infancia

(periodo que va del nacimiento hasta los ocho años de edad) puede ser interesante y gratificante para los niños, especialmente si los temas se abordan a través de la robótica y la programación básica (Marina Umaschi Bers, 2008). Además, las habilidades obtenidas con la programación y la robótica son un aspecto clave en el desarrollo de los niños y su futuro profesional (Binkley et al., 2012).

Asimismo, el aprendizaje en los niños de edad preescolar debería darse de forma concreta y divertida para obtener su atención. Es claro que la robótica es una buena herramienta de aprendizaje para niños pequeños, porque los robots les permiten a los niños tener un aprendizaje partiendo desde lo concreto hacia lo abstracto, de forma lúdica.

Por otro lado, es importante incorporar la habilidad de colaborar desde edades tempranas. Los niños, por la etapa de desarrollo de las primeras edades, son egocéntricos, pero cuando empiezan a realizar acciones a favor de los demás, su egocentrismo inicial cede el paso a la conducta cooperadora (Zurbano Díaz de Cerio, 1998). Esto se puede lograr incorporando actividades de aprendizaje colaborativo en el salón de clases. Con ello se consiguen ventajas en dos importantes áreas: la cognitiva y la socio-afectiva (citadas en la sección anterior). Además, favorece la capacidad de resolver problemas de forma creativa, a partir de estrategias de negociación, la comunicación, la mediación y la búsqueda cooperativa de alternativas.

A nivel mundial existe un déficit alto de profesionales en STEM. Esta realidad no es ajena a Costa Rica, donde se tiene una demanda de estos profesionales mucho mayor a la oferta actual. El país se enfrenta al reto de despertar el interés de los niños por la tecnología desde edades tempranas y una forma de alcanzarlo es a través de la enseñanza de la programación y robótica de manera lúdica. Sin embargo, las herramientas disponibles en el mercado para este propósito son costosas y, además, muchos docentes carecen de estos conocimientos. Por otro lado, no existe ninguna herramienta dirigida a niños de edades entre 4 y 6 años que integren el aprendizaje de la programación (o el desarrollo del pensamiento abstracto), el fomento del pensamiento lógico, la robótica, los dispositivos móviles y el aprendizaje colaborativo.

Por estas razones, en este trabajo de investigación se realizó el diseño y la construcción de la herramienta colaborativa de programación TITIBOTS Colab. También se evaluó su impacto en el proceso de colaboración en niños con edades entre 4 y 6 años. TITIBOTS Colab es una herramienta colaborativa móvil que permite el aprendizaje de la programación en niños de preescolar mientras se divierten jugando, aprenden fundamentos de robótica y de programación.

Con esta investigación se desea promover el desarrollo del pensamiento computacional desde edades tempranas, a través de la programación de robots de forma colaborativa. Así, mediante TITIBOTS Colab, se incentiva en los niños el desarrollo de habilidades blandas y técnicas, necesarias para asegurar un futuro lleno de oportunidades y próspero para la niñez costarricense.

Con la descripción del problema de investigación, se plantea la hipótesis de la investigación.

1.8 Preguntas de Investigación

En esta investigación se busca responder las siguientes preguntas:

1. ¿Qué actividades del aprendizaje colaborativo se pueden automatizar para ser incorporadas en una herramienta colaborativa de programación orientada a niños con edades entre 4 y 6 años?
2. ¿Qué lineamientos se deben tomar en cuenta al diseñar y construir una herramienta colaborativa de programación orientada a niños con edades entre 4 y 6 años?
3. ¿Es posible que niños con edades entre 4 y 6 años puedan usar una herramienta colaborativa basada en interfaces móviles para programar robots?
4. ¿Es posible que una herramienta colaborativa de programación pueda incentivar la colaboración en niños con edades entre 4 y 6 años?

A partir de estas preguntas de investigación, se definió el objetivo general y los objetivos específicos del trabajo propuesto, que se detallan a continuación.

1.9 Objetivos de la Investigación

De acuerdo con las preguntas de investigación planteadas, a continuación, se establecen el objetivo general y los objetivos específicos de la investigación.

1.9.1 Objetivo General

El objetivo general de la investigación es evaluar el impacto de una herramienta colaborativa de programación, basada en interfaces móviles y robots, en el proceso de colaboración en niños con edades entre 4 y 6 años.

1.9.2 Objetivos Específicos

Los objetivos específicos de la investigación son los siguientes:

1. Identificar las actividades del aprendizaje colaborativo que pueden ser automatizadas para desarrollar una herramienta que apoya el proceso de aprendizaje de la programación en niños con edades entre 4 y 6 años.
2. Diseñar e implementar una herramienta colaborativa de programación orientada a niños con edades entre 4 y 6 años.
3. Generar una guía de lineamientos para el diseño y la construcción de herramientas de aprendizaje de la programación orientadas a niños con edades entre 4 y 6 años.
4. Valorar el aporte de la herramienta colaborativa de programación en el proceso de colaboración en niños con edades entre 4 y 6 años.

Con los objetivos definidos, se procede a discutir los alcances y las limitaciones de la presente investigación.

1.10 Alcances y Limitaciones

La investigación contempla la evaluación del impacto de una herramienta colaborativa de programación, basada en interfaces móviles y robots, en el proceso de colaboración en niños con edades entre 4 y 6 años. Con el fin de cumplir con las preguntas de investigación y los objetivos se identificaron las actividades del aprendizaje colaborativo que se puedan automatizar. Para esto se creó una herramienta colaborativa para apoyar el proceso de programación en niños de edad preescolar. Se generó una guía de lineamientos para el diseño y la construcción de una herramienta colaborativa orientada a niños entre 4 y 6 años, y se valoró el aporte de dicha herramienta en el proceso de colaboración de los niños de la primera infancia. Esta herramienta está implementada en dispositivos móviles para programar robots, con el fin de que los niños, desde temprana edad, desarrollen la parte cognitiva lógica para que puedan resolver problemas simples de forma organizada.

El objetivo de la creación de esta herramienta es apoyar a los docentes para que puedan enseñar conceptos básicos de programación y robótica a los niños de edad preescolar en un contexto colaborativo, antes de que incluso hayan aprendido a leer y escribir, y con ello favorecer el desarrollo del pensamiento lógico, la abstracción, la solución de problemas y la colaboración.

La audiencia a la que está dirigida la herramienta es niños entre 4 y 6 años. La herramienta fue diseñada para ser utilizada en un contexto educativo (salón de clases de preescolar), debido a que los niños de estas edades necesitan de mediación para aprender y para colaborar.

El enfoque de esta investigación recae en la importancia de que los niños programen de forma colaborativa, dejando de lado que los niños construyan los robots, debido a las edades que se contemplan. Es decir, es posible que se trabaje con robots previamente diseñados y construidos, excluyendo de la investigación la construcción de los robots.

Asimismo, la investigación se realizó en dispositivos móviles para dar mayor comodidad y flexibilidad a los niños. Se seleccionó el Sistema Operativo Móvil (SOM) Android, debido a

los requisitos de la herramienta y el proyecto de desarrollo, ya que este SOM ofrece una buena experiencia de usuario y cumple con los requisitos necesarios de la audiencia. Además, debido a que la investigación fue de carácter académico, se procuró aprovechar al máximo las posibilidades que ofrecen las herramientas gratuitas (*open source*).

La evaluación de la herramienta de programación estuvo a cargo de la FOD, ya que ellos serán los encargados de llevar a cabo la ejecución de las evaluaciones con los niños y solicitar los permisos correspondientes a los padres o encargados. Asimismo, la FOD brindó varios investigadores que fueron los encargados de evaluar las pruebas realizadas mediante las observaciones realizadas. La investigadora se encargó del diseño y construcción de la herramienta, del diseño experimental y de analizar el proceso de evaluación para obtener los resultados relevantes de esta investigación. Los docentes también fueron evaluadores de la herramienta, y se tuvo contacto directo con ellos.

Tal y como se discutió en los antecedentes, no existe al día de hoy, una herramienta similar a la propuesta, se construyó y se evaluó en este proyecto de investigación. A pesar de que hay algunos proyectos orientados a los niños de 4 años, tales como Robot Turtles, Hello Ruby, Tern, PRIMO, SoRo, KIBO y Teebot (aunque los cinco últimos usan robots), estos no fueron diseñados para dispositivos móviles ni integran el aprendizaje colaborativo. Debido a esto, no será posible comparar el ambiente diseñado con otros ambientes previamente construidos.

La evaluación de la herramienta colaborativa de programación se abordará de dos maneras. La primera evaluación comprueba que la herramienta cumpla con los requerimientos especificados por los expertos y que sea intuitiva y fácil de usar para los niños; se realizará mediante pruebas de funcionalidad y usabilidad tradicionales. La segunda evaluación está orientada a valorar el aporte de la herramienta en el proceso de colaboración en niños con edades entre 4 y 6 años. Para esta última evaluación se realizó un cuasi-experimento con dos grupos de preescolar de instituciones educativas diferentes (grupos naturales).

Por último, se usaron recursos de software y hardware *open-source* con el fin de favorecer la replicación de la investigación, y su integración en la enseñanza y el aprendizaje de las disciplinas STEM en la Educación Preescolar de las Escuelas Públicas de Costa Rica. Con el objetivo de contribuir al desarrollo tecnológico del país y a la inclusión social.

1.11 Estructura de la Tesis

En el Capítulo 2 se presenta el proceso y los resultados obtenidos de la revisión sistemática de literatura llevada a cabo. En el Capítulo 3 se describe el proceso de aprendizaje colaborativo, una de las áreas más notables y fértiles de la teoría, la investigación y la práctica en la educación. En el Capítulo 4 se discute sobre usabilidad, diferentes metodologías y técnicas. En el Capítulo 5 se describe la base metodológica llevada a cabo para el desarrollo y el logro de los objetivos de la investigación. En el Capítulo 6 se describe el proceso de creación de la herramienta colaborativa de programación TITIBOTS Colab. En el Capítulo 7 se muestran los resultados y la discusión de la investigación. En el Capítulo 8 se muestran las conclusiones de la investigación y el trabajo futuro que se puede desprender de ella.

CAPÍTULO 2. REVISIÓN SISTEMÁTICA DE LITERATURA

En este capítulo se describe el proceso llevado a cabo para obtener información sobre otras herramientas de programación, basadas en robots y orientadas a niños (trabajo relacionado con la investigación), mediante una revisión sistemática de literatura (RSL).

2.1 Preguntas de la Revisión Sistemática de Literatura

En la realización de esta RSL se buscó responder las siguientes preguntas:

1. ¿Qué tipo de interfaces se usan más en las herramientas de programación basadas en robots para niños? (PI-RSL_1)
2. ¿Existe alguna herramienta de programación basada en robots que fomente la colaboración y el aprendizaje de programación en niños? (PI-RSL_2)
3. ¿Qué efectos tiene la incorporación de los robots en el aprendizaje de la programación en niños? (PI-RSL_3)

2.2 Descripción General de la Revisión Sistemática de Literatura

La RSL se usa para encontrar estudios relacionados con un tema de investigación. El proceso de revisión en el presente estudio se hizo con base en las etapas de la revisión sistemática de literatura propuesta por Kitchenham (2004). El proceso de mapeo propuesto por Kitchenham está basado en las siguientes tres etapas (Kitchenham, 2004):

1. Planeación de la revisión.
 - a. Identificación de la necesidad para la revisión.
 - b. Desarrollo del protocolo de la revisión.
2. Realización de la revisión.
 - a. Identificación de la investigación.
 - b. Selección de estudios primarios.
 - c. Evaluación de la calidad del estudio.
 - d. Extracción de los datos y monitoreo.

- e. Síntesis de los datos.
3. Reporte de la revisión.

La etapa final, el reporte de la revisión, abarca la revisión de todo el proceso de mapeo, el análisis de todos los resultados y las publicaciones revisadas en el trabajo.

Aunque esta investigación se orienta a niños de preescolar (entre 4 y 6 años), en la revisión sistemática de literatura se tomaron en cuenta todas las herramientas para programar robots orientadas a niños, sin un rango de edades definido.

En las secciones siguientes se presentan las etapas de planificación y realización de la revisión.

2.2.1 Planificación de la Revisión

En esta etapa se define el protocolo de mapeo de la RSL, incluyendo los métodos realizados. Los aspectos del protocolo de mapeo de la revisión de esta investigación son los siguientes:

- **Objetivo.** El estudio de mapeo tiene como objetivo principal encontrar las herramientas para aprender a programar robots orientadas a niños.
- **Preguntas de investigación.** Las preguntas de investigación de la revisión están definidas en la Sección 2.1.
- **Fuente del método de búsqueda.** Las fuentes de búsqueda serán las bases de datos electrónicas usando una consulta formal.
- **Palabras claves.** Las palabras claves que se usaron para la revisión fueron las siguientes: “*programming assistance tool*”, “*programming environment*”, “*programming kit*”, “*programming interface*”, “*programming language*”, “*programming framework*”, “*robots*”, “*robotics*”, “*children*”, “*kid*” e “*infant*”.
- **Lista de fuentes.** Todos los tipos de publicaciones indexadas en ACM Digital Library, IEEE Xplore Digital Library y Springer Link.

- **Criterios de inclusión y exclusión.** Las publicaciones que fueron consideradas en este mapeo deben estar disponibles en formato electrónico, ser un artículo de una revista o conferencia internacional, estar escritas en inglés, y estar en el rango de 1980 a la actualidad. Además, las publicaciones deben tener al menos las siguientes palabras claves: “*programming*”, “*robot*” y “*children*” en el título y el resumen.
- **Criterio de evaluación de calidad.** Las publicaciones deben tener información sobre herramientas para el aprendizaje de la programación de robots orientadas a niños. Cada publicación debe tener información de la herramienta que usan para el aprendizaje de la programación orientada a niños: descripción, funcionamiento, características, robots que utiliza, entre otros.

Con la definición del protocolo de mapeo de la revisión, se procede a describir la realización de la revisión.

2.2.2 Proceso de la Revisión

Con el protocolo de mapeo de la revisión completo, el proceso del mapeo fue realizado por la investigadora de este trabajo y es supervisado por tres investigadores del grupo de investigación USING (*USer INteraction Group*) de la Escuela de Ciencias de la Computación e Informática de la Universidad de Costa Rica.

La etapa de identificación de la investigación se realizó del 30 de octubre al 13 de noviembre del 2015. Para ejecutar las búsquedas, se eligió una lista de bases de datos electrónicas y se utilizó la siguiente consulta: “*programming AND (tool OR interface OR language OR environment OR framework) AND (robot OR robotics) AND (children OR kid OR infant)*”. Un total de 14.514 resultados fueron devueltos por las bases de datos electrónicas consultadas. De ellos, 1.117 se recuperaron de ACM Digital Library (7,7%), 43 de IEEE Xplore Digital Library (0,3%) y 13.354 de Springer Link (92%) (ver Tabla 2).

Del conjunto inicial de 14.514 resultados de la búsqueda, se realizó la selección de los estudios primarios (descrito anteriormente) para ejecutar un primer filtro. Este filtro reduce el número de publicaciones que no tiene relevancia en el proceso de mapeo de forma rápida. En la selección de los estudios primarios se obtuvieron 73 estudios aceptados, 25 de ACM Digital Library, 20 de IEEE Xplore Digital Library y 28 de Springer Link (ver Tabla 2). Luego, se realizó un filtro final, donde el criterio de evaluación de calidad fue aplicado. El objetivo de esta etapa fue analizar todos los resultados que muestran evidencia de herramientas de programación basadas en robots para niños, con el fin de conformar el conjunto de los resultados de esta revisión sistemática. Después de la etapa del criterio de evaluación de calidad, 46 estudios fueron incluidos en el proceso de mapeo, 16 de ACM Digital Library, 15 de IEEE Xplore Digital Library y 15 de Springer Link (ver Tabla 2).

Tabla 2. Datos del proceso de mapeo de la RSL.

Base de Datos Electrónica	Documentos Recuperados	Relevantes 1° Filtro	Relevantes 2° Filtro
ACM Digital Library	1.117 (7,7%)	25 (34%)	16 (35%)
IEEE Xplore Digital Library	43 (0,3%)	20 (28%)	15 (33%)
Springer Link	13.354 (92%)	28 (38%)	15 (33%)
TOTAL	14.514 (100%)	73 (100%)	46 (100%)

En esta RSL se obtuvieron 14.514 publicaciones sobre el tema, y después de aplicar dos filtros, se obtuvieron 46 estudios. Estos estudios fueron revisados para realizar la etapa de extracción de datos y monitoreo, obteniendo 27 herramientas para aprender a programar robots orientadas a niños (ver la lista en la Tabla 3).

Finalmente, en la etapa de síntesis de datos del mapeo de esta revisión, se analizaron los principales aspectos observados en la literatura sobre las herramientas para programar robots orientadas a niños. La Tabla 4 muestra la información detallada de cada una de las herramientas.

A continuación, se muestran los resultados obtenidos en estas etapas.

2.3 Herramientas de Programación Basadas en Robots Orientadas a Niños

En la revisión sistemática se obtuvieron 46 estudios relevantes, de los cuales 33 eran publicaciones en conferencias (72%) y 13 en revistas (28%). Además, 4 eran publicaciones tipo “demo” (9%), 1 póster (2%), 10 *short paper* (22 %) y 31 *full paper* (67%).

En la Tabla 3 se muestra las herramientas para aprender a programar robots orientadas a niños utilizadas en los estudios relevantes de la revisión sistemática de literatura. Se puede observar una descripción general de cada una y los estudios donde se mencionan (ver la lista de referencias de la revisión sistemática de literatura en el Apéndice 2). Además, en la Tabla 4 se muestra un resumen de diferentes características de cada una de las herramientas de programación.

En la Figura 11 se muestra un gráfico de burbuja, con dos cuadrantes, donde se clasifican las herramientas de programación basadas en robots orientadas a niños según el tipo de interfaz. En el primer cuadrante (izquierda), se encuentra la información sobre los grupos (clasificación) y la cantidad de herramientas que son o no colaborativas (automatizan actividades del aprendizaje colaborativo). En el segundo cuadrante (derecha), se encuentra la información sobre los grupos (clasificación) y la cantidad de herramientas orientadas a niños en diferentes rangos de edad. En aquellas herramientas de programación donde se indica una edad mínima, pero no una máxima, la herramienta se contó para cada rango de edad incluido en su rango general (por lo que se intersecan las burbujas). Por ejemplo: si una herramienta dice que su rango de edad es “6+”, se incluyen en los intervalos de 6-8, 8-10, 10-12 y 12+.

Por otro lado, en la Figura 12 se muestra un gráfico de burbuja donde se clasifican las herramientas dependiendo si es o no colaborativa (automatizan actividades del aprendizaje colaborativo), clasificadas por rangos de edad.

Tabla 3. Lista de herramientas de programación basadas en robots y orientadas a la niñez.

Herramienta	Descripción	Artículos
A-Bricks	A-Bricks (<i>Algorithmic Bricks</i>) fue diseñado para programar mediante la manipulación de los bloques de dos maneras, ya sea mediante la conexión horizontal o por apilarlos. A través de estas dos actividades básicas de manipulación, son posibles dos tipos de actividades de programación de robots: 1) la programación procedimental través de la conexión horizontal; y 2) el establecimiento de parámetros de bloque a través de una actividad de apilamiento. A-Bricks puede expresar los conceptos de programación por procedimiento, como la secuencia, la repetición, el estado, la función y los parámetros a través de actividades prácticas.	[1]
AmbiProg	Arco es una arquitectura para gestionar un entorno digital en un contexto de ambiente inteligente, puede integrar todo tipo de objetos y es totalmente configurable. Propone una interfaz de programación visual: AmbiProg, que permite controlar cada objeto digital que compone un ambiente. AmbiProg se compone de dos partes principales, en la izquierda se muestra el área de programación gráfica con los elementos <i>start</i> y <i>stop</i> ; y a la derecha, un árbol que contiene los datos de programación (percepciones, acciones y estructuras de programas). AmbiProg está diseñado para controlar un entorno completo mediante la creación de escenarios de interacción.	[2]
Backpacks y Topobo	Backpacks son componentes físicos modulares que los niños pueden incorporar en creaciones robóticas para modular la frecuencia, amplitud, fase y la orientación de las grabaciones de movimiento. Backpacks permite a los niños investigar los principios cinemáticos básicos que subyacen a los comportamientos que exhiben sus creaciones y que se observan. Usa el sistema Topobo, el cual es un sistema de montaje constructivo con memoria cinética (la posibilidad de grabar y reproducir el movimiento físico). Topobo se compone de componentes “pasivos” (piezas de plástico estáticas), y “Activos” (módulos robóticos motorizados con memoria y comunicaciones). Al usar Topobo, un niño hace un modelo, grabar un movimiento, y ve como se reproduce. Aunque hay flexibilidad que un niño pueda editar la forma de su modelo físico (el movimiento), no puede editar la forma de su grabación (el programa).	[3], [4]
Card Programming	Card Programming se basa en un robot móvil que puede ser programado por medio de comandos y cartas de instrucciones. Todo lo que se debe hacer es organizar las cartas sobre la mesa y en la fase de programación por una cámara. El sistema de programación de cartas reconoce las cartas de instrucciones y traduce a comandos icónicos en el programa controlador del robot.	[5]
CHERP	CHERP (<i>Creative Hybrid Environment for Robotic Programming</i>) se compone de instrucciones de programación con código de color. El código CHERP puede escribirse usando una interfaz gráfica en la computadora o bloques de madera tangibles. La GUI posee íconos para que los niños hagan clic y arrastren las instrucciones en una lista, o secuencia. El TUI permite a los niños poner juntos bloques de madera, con imágenes de los íconos en los laterales. Ambas interfaces envían el programa al robot para que lo ejecute, se utilizó el kit robótico WeDo en el estudio. CHERP es un descendiente directo del lenguaje Tern y el antecesor de KIBO.	[6]

Tabla 3. Lista de herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Descripción	Artículos
Electronic Blocks	Electronic Blocks son elementos de programación tangible, con bloques con circuitos electrónicos dentro de ellos que se pueden apilar y formar estructuras que interactúan con el mundo físico. Hay tres tipos: bloques de sensores, bloques de acción y bloques lógicos. Los bloques de sensores detectan la luz, el sonido y el tacto, mientras que los bloques de acción son capaces de producir luz, sonido y movimiento. Mediante la conexión de bloques de sensores con los bloques de acción los niños pueden programar sus propias estructuras. La inclusión de bloques lógicos (negar, esperar y conmutar señales entre bloques) añade una dimensión adicional a las capacidades de las creaciones de los niños. Con el uso de los bloques, los niños pueden crear robots que chocan entre sí, luces que se encienden cuando se aplaude, y animales que susurran el uno al otro. Mediante la colocación de los bloques electrónicos unos sobre otros, los niños construyen programas, donde cada pila de bloques electrónicos es capaz de una función diferente.	[7], [8], [9]
GameBlocks	GameBlocks es un entorno de programación física para niños. Se hace uso de elementos de sintaxis físicos en forma de bloques para construir una secuencia de programación. Un carro de juguete se controla cuando se ejecuta la secuencia. Esto se realiza mediante la colocación de los bloques en una secuencia lineal en un mat de programación y al encender el subsistema electrónico asociado. Cada bloque es puesto en secuencia y el comando de infrarrojos, asociado con ese bloque, se envía al carro de juguete para su ejecución. Hay cuatro tipos de bloques de acción (avanzar, retroceder, girar a la izquierda y girar a la derecha) y dos tipos de bloques de tono (disminución de tono y el aumento de tono). GameBlocks usa el lenguaje Logo simplificado, usa símbolos en los cubos en lugar de texto en una pantalla de computadora. Cuando el niño está feliz con su programa, se activa el sistema y el programa es ejecutado por el circuito electrónico incorporado. Los resultados son visibles a través de la actuación de los motores en el carro robot de juguete.	[10], [11]
JBrick	JBrick es una aplicación para iOS que se ejecuta en un iPad o iPad Mini. Los programas se construyen usando bloques. Los bloques representan construcciones, variables, sensores, motores, etc., los comandos secuenciales se disponen de arriba a abajo en una línea vertical, mientras que las construcciones de condicionales y ciclos tienen bloques que se apilan dentro de ellos para indicar anidación y secuencia. Cada tipo de bloque tiene un símbolo y color para indicar el tipo.	[12]
KIBO	KIBO es un kit robótico diseñado específicamente para niños de 4 a 7 años. Los niños construyen sus propios robots con KIBO, los programan para que hagan lo que ellos quieren, y los decoran. KIBO es una interfaz de programación tangible, donde los niños juegan con bloques de madera que traen imágenes de comandos de programación, estos bloques pueden ser compartidos de una forma fácil. Esta investigación es la unión del software CHERP (2009) y el robot KIWI (2012).	[13]

Tabla 3. Lista de herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Descripción	Artículos
lasm2aibo	El programa lasm2aibo convierte comandos LASM (RoboLab) en comandos Aibo (R-CODE). Se usa LEGO Mindstorms NXT para construir un robot con ruedas y el robot Sony Aibo (un perro robot de cuatro patas). RoboLab fue implementado por National Instruments' LabVIEW. Los íconos individuales y programas completos se guardan como "Vis" (instrumentos virtuales). Cada ícono o programa puede ser visto como la imitación de un instrumento real. Se usa RoboLab incorporado "subVIs" (VI módulos o subprogramas) para construir los comportamientos personalizados. R-CDE se usa para implementar comandos Aibo en el sistema propuesto, porque R-CODE es fácil de usar, no requiere compilación, y ofrece un mapeo más natural hacia RoboLab.	[14]
LEGOsheets	LEGOsheets es un entorno de programación basado en reglas que permite a los niños simular y manipular el ladrillo programable de LEGO RCX.	[15]
MicroWorlds EX Robotics	MicroWorlds EX Robotics es una nueva versión de Logo que incorpora la posibilidad de programación para el robot Cricket y el ladrillo LEGO RCX. Esta versión incorporó muchos cambios en la herramienta y en el lenguaje de Logo. Se incluyeron muchas características adicionales como herramientas de dibujo, editores, fabricación de melodías, y posibilidad de importar gráficos y sonidos. Estas nuevas características, junto con Logo, permitieron apoyar la creación de proyectos multimediales, juegos y simulaciones.	[16]
MODEBOTS	MODEBOTS (<i>Mobile Development Environment for Robots</i>) es una herramienta de programación orientado a niños de preescolar. El primer prototipo implementado es simple y se desarrolló con el objetivo de resolver los problemas técnicos relacionados con la comunicación Bluetooth entre el dispositivo móvil y el ladrillo NXT. Por esta razón, se implementó un protocolo de comunicación con los comandos, así como un lenguaje para interpretar dichos comandos, del dispositivo móvil al ladrillo NXT, se estableció una conexión uno a uno.	[17]
NQCBaby y NXCJunior	Usando NQCBaby o NXCJunior (basados en LOGO), los niños crean programas que contienen los tres componentes básicos de la algoritmia: secuencia, selección e iteración. En sus actividades con pequeños robots, los niños manipulan los componentes tangibles de los robots. Después de la especificación del programa que describe una trayectoria, corren el código y comprueban su solución mientras el robot se mueve (ladrillo RCX). NQCBaby, un lenguaje textual basado en el idioma materno de base y tiene primitivas del lenguaje de los niños. También se diseñó el lenguaje de programación NXCJunior, texto, íconos e idioma materno de base, para ser utilizado con el robot NXT. NXCJunior corresponde a un subconjunto del lenguaje de programación NXC. Al ser NQCBaby y NXCJunior basado en sus ideas, comparten un subconjunto común de primitivas.	[18], [19], [20]
P-CUBE	P-CUBE consiste en un robot móvil, bloques cúbicos, un mat de programación y una computadora. El usuario crea un programa al colocar los bloques en los huecos de la superficie del mat de programación basado en la estructura del algoritmo. Los lectores RFID están puestos en el mat de programación y la información de las etiquetas RFID en los bloques, ambas se leen con el fin de especificar la ubicación de cada bloque. Luego, por cable USB se transfieren los RFID del programa creado a la computadora, quien los interpreta al lenguaje del robot móvil, y guarda el programa interpretado en una memoria MicroSD, la cual se le pone al robot para que ejecute el programa. Usando P-CUBE, los usuarios pueden crear dos tipos de programas: programas secuenciales y programas de seguimiento de líneas.	[21]

Tabla 3. Lista de herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Descripción	Artículos
Playte	<p>Playte es una interfaz tangible, diseñada para que los niños animen robots interactivos, da acceso fácil del control del robot basado en comportamientos. El sistema permite la manipulación física de comportamientos de los ladrillos programables de LEGO y, permite grabar y entrenar nuevos comportamientos. Playte está conformado por un <i>shell</i> de programación (caja cuadrada construida con el Sistema de Ladrillos de LEGO) y ladrillos de comportamiento. En el lado superior del <i>shell</i> de programación hay diez ranuras donde el usuario puede colocar los ladrillos de comportamiento. Una ranura consiste de un ladrillo LEGO especial con conexiones eléctricas y un diodo LED para retroalimentación de los usuarios. La fila superior de cinco ranuras selecciona qué comportamientos están activos en el robot. La última fila de cinco ranuras, permite al usuario eliminar un comportamiento, registrar un comportamiento, entrenar un comportamiento, y combinar varios comportamientos en uno. Los comportamientos están representados por ladrillos LEGO especiales, que están modificados para contener una resistencia para la identificación del comportamiento. Las resistencias eléctricas se leen usando convertidores AD por un microcontrolador dentro de la <i>shell</i> Playte. El microcontrolador envía periódicamente el estado de las ranuras de Playte a la computadora usando comunicación serial. La computadora envía el estado de Playte al robot junto con cualquier entrada del usuario desde un <i>gamepad</i>.</p>	[22]
PROTEAS kit	<p>PROTEAS (PROgramming Tangible Activity System) kit fue diseñado para funcionar como una herramienta de programación introductoria y fácil uso. Con la herramienta de programación tangible los niños pueden desarrollar su código mediante la organización de bloques que representan comandos de programación sencillos. La plataforma consta de dos lenguajes de programación tangibles y un lenguaje gráfico. Los lenguajes tangibles son T_Butterfly y T_ProRob, ambos lenguajes se pueden usar de forma complementaria. Los usuarios, interactuando con el subsistema T_Butterfly pueden programar con el fin de llevar una mariposa virtual en un laberinto. Además, usando el subsistema T_ProRob, los usuarios pueden programar un robot Lego NXT. Por último, el kit contiene el subsistema V_ProRob, el equivalente gráfico del subsistema T_ProRob. T_ProRob funciona de la siguiente manera: el usuario conecta los comandos en la caja maestra, los cuales representan su programa. Para ejecutar el programa se debe oprimir el botón de ejecución que se encuentra en la caja maestra. La caja maestra se comunica con los bloques y lee el programa creado. Luego, la caja maestra se comunica con una computadora a través de Bluetooth o RS 232. Esta información queda en registros en una base de datos sobre el programa. Una vez que la computadora finalice el proceso de grabación, envía el programa al robot NXT a través de Bluetooth con el fin de ejecutarlo. Todas las comunicaciones son bidireccionales, de esta manera a la caja maestra y a los bloques se les informa acerca de la ejecución del programa con el robot NXT. V_ProRob tiene los mismos comandos y parámetros que ofrece T_ProRob. El subsistema es el equivalente a T_ProRob, pero gráfico, y ofrece a los usuarios una alternativa para programar el NXT con un entorno gráfico simple y fácil.</p>	[23], [24], [25]

Tabla 3. Lista de herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Descripción	Artículos
RoboEduc	RoboEduc es un software para la programación, el diseño de robots y el estudio de las interacciones entre agentes. Es decir, RoboEduc apoya la enseñanza de los conceptos de robótica y programación de robots y la comunicación, la asistencia en la construcción, el control y la programación de diversos modelos de robots prototipos, y permite el aprendizaje de conductas que se usan para construir robots. RoboEduc tiene cinco niveles de programación, cada nivel es un el grado de complejidad mayor, desde programar con íconos hasta escribir líneas de código.	[26]
RoboGan	RoboGan es un entorno desarrollado para que los niños programen un robot físico, incluye interfaces que van progresando adaptándose al crecimiento de su comprensión. Los niños construyen comportamientos de robots mediante el entorno. Este entorno incluye una interfaz de computadora, un robot físico (sistema LEGO) y “paisajes” modificables para la navegación del robot. Un componente clave del entorno es una interfaz icónica sencilla e intuitiva para la definición de las reglas de control. El panel izquierdo muestra las entradas al sistema, la información que los sensores pueden sentir y transmitir. El panel de la derecha presenta las posibles acciones que el robot puede realizar. La sección central está dedicada al “tablero de construcción” en forma de matriz. La configuración de esta sección cambia con tareas de avance: comenzando con una pareja de condición-acción (dos reglas completas) o cuatro parejas de condición-acción. Cada cuadro muestra una acción, que ejecutará el robot al cumplir las dos condiciones (fila y columna).	[27], [28], [29]
RoboLab (LabVIEW)	RoboLab es una herramienta basada en íconos y permite crear programas que controla el ladrillo RCX. Ofrece modos diferentes de programación adaptados al nivel de aprendizaje de los alumnos, como <i>Pilot e Inventor</i> . Además, ofrece el modo <i>Investigator</i> orientado a su uso en el laboratorio de ciencias.	[16], [30], [31], [32], [33] [34]
Software LEGO Mindstorms NXT	El LEGO Mindstorms NXT (NXT-G) tiene un ambiente de programación visual que permite crear fácilmente un nuevo programa y comunicarse con diferentes dispositivos a través de bluetooth. El software de NXT se basa en National Instruments’ LabVIEW, que permite arrastrar íconos y soltarlos en la pantalla principal del programa para usarlo.	[16], [35], [36]
SoRo toolkit	SoRo (<i>Social Robot</i>) toolkit ofrece un diseño novedoso que incorpora robots sociales y medios tangibles para que los niños de preescolar exploren conceptos computacionales en el contexto del diseño de reglas de interacción que pueden “enseñar” a un robot social. Para la plataforma robótica, se usa DragonBot, un robot que se controla con una aplicación Android con una interfaz de programación gráfica intuitiva, arrastrando y soltando bloques en la pantalla. El robot aparece como una criatura suave, peluda y de fantasía, diseñada para involucrar a los niños. DragonBot es una plataforma expresiva que tiene un gran repertorio de expresiones faciales, sonidos y acciones. La aplicación móvil le permite al experimentador colocar las reglas creadas por los niños con los componentes tangibles para desencadenar eventos, insertar nuevas reglas y mostrar las reglas aprendidas, y así teleoperar al robot.	[37]

Tabla 3. Lista de herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Descripción	Artículos
Tangible cubes	<p>El entorno de programación consta de un número de bloques acrílicos, cada uno integrado con imanes (están montados en la superficie interior de la parte inferior del cubo). En esta implementación los cubos se colocan en bandejas acrílicas que contienen circuitos para detectar y que están conectados a un controlador central. El controlador detecta que el cubo está colocado en cada posición permisible y envía los comandos apropiados hacia un robot de juguete. Las posiciones permitidas se limitan a las indicadas por las bandejas. Los interruptores de alarma magnéticos de la puerta/ventana se utilizan, es una combinación de un imán de barra y un interruptor de láminas, y ambos están encapsulados en plástico. Una vez que el sistema ha sido activado, la interpretación del código comienza en la primera bandeja y procede de forma secuencial a lo largo de las bandejas hasta que todas se han consultado. Hay dos maneras de proceder. El primer modo de ejecución espera a que un bloque sea colocado en una bandeja. La ejecución se detiene en la siguiente posición de bandeja vacía hasta que un cubo sea colocado allí. La ejecución continúa de esta manera, haciendo una pausa donde no hay cubo. Un modo alternativo de ejecución es detener la ejecución antes de que se pregunte a la primera bandeja y se coloque todos los bloques en secuencia y en las posiciones deseadas. Cuando el programa creado se ha completado, se elimina la interrupción y se realiza la interpretación del código completo.</p>	[38]
TanProRobot	<p>TanProRobot es un sistema tangible diseñado para niños de 1ero y 2do grado, con el objetivo que aprendan conceptos de programación. Se compone de tres partes, bloques de programación tangibles (bloques de poder, bloques de sensores, y los bloques de acción), el carro robot (Arduino Mega 2560) y los manipuladores (modelos de casa, túnel, papel de color y baches de velocidad). Los niños pueden programar el carro robot para realizar acciones mediante la organización de los bloques de programación. El Arduino Mega en el robot actúa como la unidad de procesamiento central del sistema. Hay un botón digital en el robot que actúa como interruptor de la fase de ejecución. En fase de implementación, el robot recibe la señal inalámbrica de los bloques de programación. Luego, el Arduino decodifica la señal, controla el mini reproductor MP3 para proporcionar información, mantiene y almacena las secuencias digitales de las reglas. Después, al entrar en la fase de ejecución, el robot ejecutará las reglas que la función de gestión de eventos. El Arduino hará que los sensores detecten el estado actual y controlen las acciones del robot según el programa construido con los bloques de programación.</p>	[39], [40]
Tern	<p>Tern es un lenguaje computacional tangible diseñado para proporcionar una introducción a la programación orientado a niños en entornos educativos. Se pueden crear programas para robots como el LEGO Mindstorms RCX o el iRobot Create. Tern se presenta en una exposición permanente en el Museo de Ciencias de Boston llamada Robot Park. También se está usando para el proyecto Tangible Kindergarten del grupo Tufts University Developmental Technologies. Tern permite crear programas usando bloques de madera entrelazados que representan las acciones que debe llevar a cabo el robot. Los bloques de madera no contienen electrónica incorporada o fuentes de alimentación, lo que usa es una cámara web estándar conectada a una computadora para tomar la imagen del programa, que luego se convierte en código digital usando la librería de visión por computador TopCodes.</p>	[41], [42], [43], [44]

Tabla 3. Lista de herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Descripción	Artículos
TITIBOTS Colab	TITIBOTS Colab es una herramienta colaborativa de programación desarrollado para los niños de preescolar en entornos educativos. TITIBOTS Colab permite a los niños desarrollar un programa trabajando colaborativamente y enviar el programa creado a un robot para su ejecución. La herramienta ofrece una sencilla interfaz gráfica usando un enfoque iconográfico, es intuitiva y fácil de usar para los niños. TITIBOTS Colab fue diseñado para permitir el uso de cualquier robot con un conjunto abierto de comandos. La comunicación entre las tabletas y el robot es a través de WiFi Peer-to-Peer; el docente puede monitorear cada grupo y evaluar el trabajo realizado. Además, el docente es responsable de enviar al robot cada programa creado por los equipos de colaboración para su ejecución. Esta versión corre en los dispositivos Android. La tableta del docente está conectada al robot, los niños pueden crear un programa usando arrastrar y soltar los comandos, y luego lo envían al docente para ejecutarlo en el robot.	[45]
TOTA kit	TOTA (<i>Toys of the Apocalypse</i>) es un kit de construcción con sensores de bajo costo que permite a los niños crear sus propios juguetes con componentes simples y polímeros conductivos caseros. Se da a los niños una introducción a la electrónica y la programación que permite un completo control creativo de la estética y la mecánica. El objetivo de TOTA es introducir a los niños a los conceptos básicos de circuitos, comportamiento programado, y la ingeniería mecánica. TOTA busca alcanzar un nivel de simplicidad ofreciendo componentes que son de fácil comprensión e interactivos. La interfaz de TOTA permite que los niños comienzan la programación básica: declaraciones “si/entonces”, con la retroalimentación de los sensores conectados para crear comportamientos e interacciones únicas.	[46]

Tabla 4. Características de las herramientas de programación basadas en robots y orientadas a la niñez.

Herramienta	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Robots	Fuente de Enlace	Comunicación (Herramienta y Robot)	Herramienta Colaborativa
A-Bricks	6+	Tangible	Bloques (circuitos electrónicos)	Tangible	Robot Car (similar al Robot <i>Line-Tracers</i>)	Interfaz tangible	Cable y conectores USB	No
AmbiProg	10+	Basada en eventos	Árbol con datos de programación (percepciones, acciones y estructuras de programas)	<i>Drag and Drop</i>	Robots Nabaztag	Computadora	Wireless	No
Backpacks y Topobo	8+	<i>Record and play</i>	Backpacks y Topobo	Tangible	Backpacks	Interfaz tangible – Computadora	Circuitos electrónicos	No
Card Programming	7+	Tangible	Cartas de instrucciones	Tangible	Robot móvil	Computadora y Cámara	RS-232C	No
CHERP	5-7	Tangible y visual (gráfica)	Bloques (códigos de barras) e íconos	Tangible y <i>Drag and Drop</i>	LEGO WeDo kit	Interfaz tangible – Computadora	Escáner (lector de código de barras) y Cable USB	No
Electronic Blocks	4-8	Tangible	Bloques (circuitos electrónicos)	Tangible	Bloques electrónicos	Interfaz tangible	Circuitos electrónicos	No
GameBlocks	5+	Tangible	Bloques (material EVA con circuitos electrónicos)	Tangible	LEGO Mindstorms RCX	Interfaz tangible	Circuitos electrónicos y Puerto infrarrojo	No
JBrick	7+	Visual (gráfica)	Bloques / Rompecabezas	<i>Drag and Drop</i>	LEGO Mindstorms NXT	Móvil (iOS)	Bluetooth	No

Tabla 4. Características de las herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Robots	Fuente de Enlace	Comunicación (Herramienta y Robot)	Herramienta Colaborativa
KIBO	4-7	Tangible	Bloques (códigos de barras)	Tangible	KIWI	Interfaz tangible	Escáner (lector de código de barras)	No
lasm2aibo	8+	Por procedimientos	Expresiones en procedimientos o funciones	Escritura de líneas de código	LEGO Mindstorms NXT y Robot Sony Aibo	Computadora	Bluetooth y Cable USB	No
LEGOsheets	10+	Basada en reglas	Reglas e Íconos	Escritura de reglas y <i>Drag and Drop</i>	LEGOpet (LEGO Mindstorms RCX)	Computadora	Puerto infrarrojo	No
MicroWorlds EX Robotics	10+	Visual (gráfica)	Bloques / Rompecabezas	<i>Drag and Drop</i>	Pico-Cricket kit y LEGO Mindstorms RCX	Computadora	Puerto infrarrojo y Cable USB	No
MODEBOTS	4-6	Visual (gráfica)	Íconos y Sonidos	<i>Drag and Drop</i>	Mono Tití (LEGO Mindstorms NXT)	Móvil (Android)	Bluetooth	No
NQCBaby and NXCJunior	7+	Visual (gráfica)	Íconos y Sonidos	<i>Drag and Drop</i>	Chariot (LEGO Mindstorms RCX y NXT) – Bee-Bot, Parallax Scribbler, LEGO Mindstorms RCX y LEGO Mindstorms NXT	Computadora	Puerto infrarrojo, Bluetooth y Cable USB	No
P-CUBE	5+	Tangible	Mat de programación (lectores RFID) y bloques (etiquetas RFID)	Tangible	Robot móvil (Arduino UNO)	Computadora	Cable USB, lectores RFID y MicroSD	No

Tabla 4. Características de las herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Robots	Fuente de Enlace	Comunicación (Herramienta y Robot)	Herramienta Colaborativa
Playte	6+	Tangible basada en comportamientos	Shell de programación (microcontrolador con convertidores AD) y ladrillos de comportamientos (resistencias eléctricas)	Tangible	Robot abeja (Robotis Bioid kit y LEGO Mindstorms NXT)	Interfaz tangible – Computadora – Gamepad	Cable serial y Bluetooth	No
PROTEAS kit	5+	Tangible y visual (gráfica)	Caja maestra (colocar los bloques), bloques (circuitos electrónicos) e íconos	Tangible y <i>Drag and Drop</i>	LEGO Mindstorms NXT	Interfaz tangible – Computadora	Bluetooth y RS 232	No
RoboEduc	10+	Visual (gráfica) y por procedimientos	Íconos y expresiones en procedimientos	<i>Drag and Drop</i> y escritura de líneas de código	LEGO Mindstorms RCX	Computadora	Puerto infrarrojo	No
RoboGan	5-6	Basada en reglas	Íconos	<i>Drag and Drop</i>	LEGOpet (LEGO Mindstorms NXT)	Computadora	Bluetooth y Cable USB	No
RoboLab (LabVIEW)	7+	Visual (gráfica)	Íconos y Sonidos	<i>Drag and Drop</i>	LEGO Mindstorms RCX	Computadora	Puerto infrarrojo	No
Software LEGO Mindstorms NXT	7+	Visual (gráfica)	Íconos y Sonidos	<i>Drag and Drop</i>	LEGO Mindstorms NXT	Computadora	Bluetooth y Cable USB	No

Tabla 4. Características de las herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Robots	Fuente de Enlace	Comunicación (Herramienta y Robot)	Herramienta Colaborativa
SoRo <i>toolkit</i>	4-8	Tangible y basada en eventos	Postales de vinilo de colores y reutilizables, creando reglas mediante plantillas con <i>triggers</i> y acciones	Tangible	DragonBot	Interfaz tangible – Móvil (Android)	Bluetooth	No
Tangible cubes	10+	Tangible	Bandejas acrílicas (circuitos eléctricos) y bloques acrílicos (imanes)	Tangible	Robot de juguete	Interfaz tangible	Bluetooth	No
TanProRobot	6-8	Tangible	Bloques	Tangible	Robot Car (Arduino)	Interfaz tangible	Wireless	No
Tern	5+	Tangible	Piezas de rompecabezas (bloques físicos)	Tangible	LEGO Mindstorms RCX y iRobot Create	Interfaz tangible – Computadora	Técnicas de visión por computadora, Puerto infrarrojo y Bluetooth	No
TITIBOTS Colab	4-6	Visual (gráfica)	Íconos y Sonidos	<i>Drag and Drop</i>	Mono Tití (LEGO Mindstorms NXT-EV3 y Arduino), Serpiente Cora (Raspberry PI)	Móvil (Android)	WiFi	Sí

Tabla 4. Características de las herramientas de programación basadas en robots y orientadas a la niñez. (cont.)

Herramienta	Rango de Edad	Paradigma de Programación	Componentes de Programación	Tipo de Interacción	Robots	Fuente de Enlace	Comunicación (Herramienta y Robot)	Herramienta Colaborativa
TOTA	8+	Programación basada en comportamientos	Íconos	<i>Drag and Drop</i>	Construcciones de juguetes: kits de sensores (ofrecidos por TOTA), componentes simples y polímeros conductivos caseros	Móvil (iPhone)	Bluetooth	No

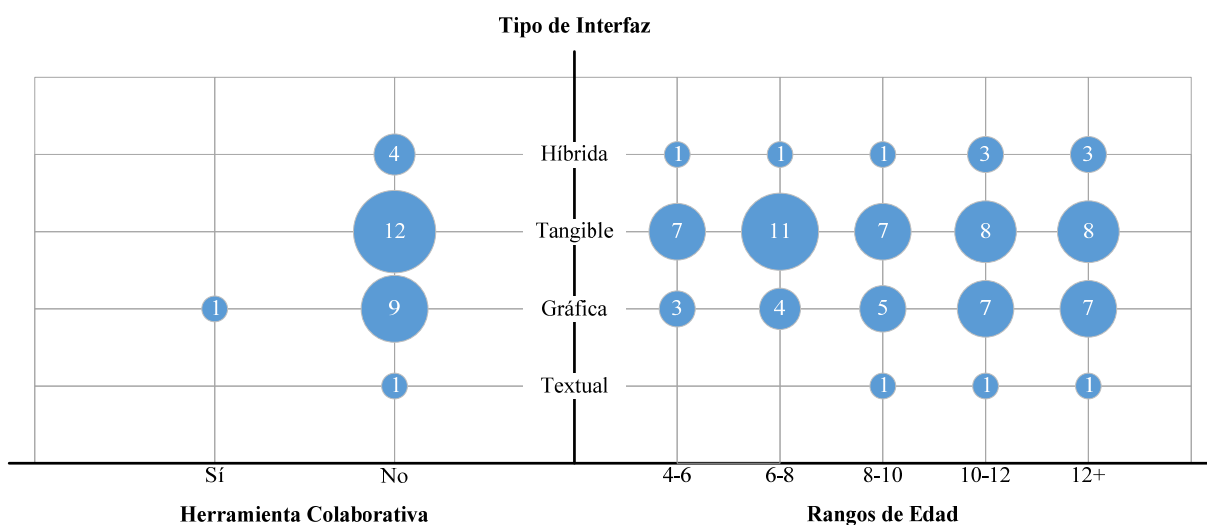


Figura 11. Clasificación de las herramientas de programación por tipo de interfaz.

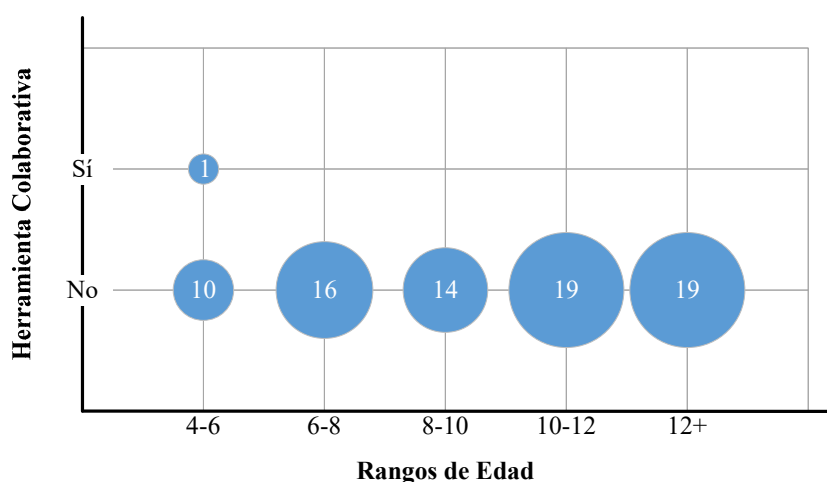


Figura 12. Clasificación de las herramientas de programación por rangos de edad.

Con la información obtenida en los estudios de la RSL, se pueden responder las preguntas planteadas en este proceso de mapeo y presentadas en la Sección 2.1. Con base en el gráfico de la Figura 11, se pueden responder las dos primeras preguntas.

Existen cuatro tipos de interfaces utilizadas en las herramientas de programación basadas en robots para niños: textual, gráfica, tangible e híbrida. De estas interfaces, las más utilizadas son, en primer lugar, las interfaces tangibles (12 herramientas) y, en segundo lugar, las interfaces gráficas (9 herramientas) (PI-RSL_1). Dado que esta investigación está dirigida a

niños entre 4 y 6 años, es importante tomar en cuenta el tipo de interfaces que se usan en este rango de edad. Como se puede observar en la Figura 11, existen 7 interfaces tangibles, 3 interfaces gráficas y 1 híbrida. Dos de las tres interfaces gráficas, aparecen en publicaciones realizadas como parte de esta investigación.

Por otro lado, se puede observar que la única herramienta colaborativa de programación, basada en robots que fomente la colaboración y el aprendizaje de programación en niños de preescolar, es TITIBOTS Colab, la cual es la herramienta construida en esta investigación (PI-RSL_2).

En relación con la tercera pregunta de la revisión (PI-RSL_3), los autores de los estudios afirman que la incorporación de robots es una poderosa herramienta práctica en el aprendizaje de la programación en niños, ya que motivan y mejoran el aprendizaje de conceptos STEM. Los autores explican un paralelismo entre el uso de la robótica durante los primeros años de escolaridad y el desarrollo de las habilidades que los estudiantes necesitan en el siglo XXI. Además, mencionan que el uso de la robótica permite a los niños explorar conceptos complejos de una manera concreta y divertida, involucra habilidades sensomotoras y socioemocionales fundamentales para el sano desarrollo de los niños pequeños, proporciona el ambiente de aprendizaje libre de ideales y el uso principal de la curiosidad natural de los niños para aprender a construir y programar. Algunas de las habilidades que promueve la robótica se refieren al pensamiento crítico y la resolución de problemas, la colaboración y el liderazgo, la agilidad y la adaptabilidad, la iniciativa y el emprendedurismo, el acceso y el análisis de información, la curiosidad y la imaginación. En síntesis, la programación es la nueva alfabetización, y al complementarla con la robótica combina las posibilidades físicas y creativas de construir y manipular objetos.

En el siguiente capítulo, se describe el proceso del aprendizaje colaborativo y la importancia que tiene en las actividades de enseñanza-aprendizaje desde edades tempranas.

CAPÍTULO 3. APRENDIZAJE COLABORATIVO

El aprendizaje colaborativo representa una teoría y un conjunto de estrategias metodológicas que surgen de un nuevo enfoque de la educación, donde el trabajo cooperativo en grupo es un componente importante en las actividades de enseñanza-aprendizaje.

Según Johnson y Johnson, el aprendizaje colaborativo existe cuando los estudiantes trabajan juntos para lograr objetivos de aprendizaje compartidos (D. W. Johnson et al., 1998; D. W. Johnson, Johnson, & Stanne, 2000). El aprendizaje colaborativo tiene muchas formas y definiciones. La mayoría de los enfoques de colaboración usan equipos pequeños y heterogéneos, por lo general de cuatro o cinco miembros, trabajando juntos hacia una tarea de grupo en la que cada miembro es responsable de un resultado que no se puede completar a menos que los miembros trabajen juntos. En otras palabras, los miembros del grupo son positivamente interdependientes.

Cuando la colaboración es exitosa, tiene lugar la sinergia, y el todo se vuelve mayor que la suma de sus partes. Para que los grupos de colaboración sean eficaces, los miembros deben participar en actividades de trabajo en equipo y otras tareas que tienen que ver explícitamente con el desarrollo de las habilidades sociales necesarias para el trabajo en equipo eficaz. Los miembros también deben participar en actividades de procesamiento de grupo en las que se discuten las habilidades interpersonales que influyen en su eficacia en el trabajo conjunto (Collazos, 2003).

La base del trabajo colaborativo es la cooperación, y es por ello que frecuentemente se solapan los términos de aprendizaje colaborativo y aprendizaje cooperativo. Cooperar significa trabajar juntos para alcanzar objetivos compartidos, y está directamente relacionado con el producto final. Colaborar se refiere a la forma de trabajo, y se relaciona más con el proceso que con el producto final. No obstante, para efectos del presente trabajo, ambas definiciones son acertadas y, los términos “colaboración” y “cooperación” se consideran sinónimos.

Sin embargo, varios autores han criticado sobre el aprendizaje cooperativo y están a favor del aprendizaje colaborativo. Ambos paradigmas están fundados en la epistemología constructivista y por tanto son muchos más los aspectos que comparten que los que los diferencian. No obstante, Panitz señala algunas pequeñas distinciones entre ellos (Panitz, 1998):

1. En el aprendizaje cooperativo el proceso de enseñanza-aprendizaje está más estructurado deliberadamente por el docente. Hay una estructura predefinida de la actividad, mientras que en el aprendizaje colaborativo se le atribuye mayor responsabilidad individual al estudiante y, por tanto, la estructura de la actividad tiende a ser más libre.
2. En el aprendizaje cooperativo se enfatiza más en la tarea; en cambio, en el colaborativo parece haber mayor énfasis en el proceso. La premisa básica del aprendizaje colaborativo es la construcción del consenso a través de la cooperación de los miembros del grupo.

En las situaciones colaborativas, las personas buscan resultados beneficiosos para sí mismas y para los otros integrantes de sus grupos. El aprendizaje colaborativo en la educación usa grupos pequeños, donde los alumnos trabajan juntos para mejorar su propio aprendizaje y el de los demás (D. W. Johnson & Johnson, 1999). Los alumnos además sienten que pueden alcanzar sus objetivos de aprendizaje si y solo si los demás integrantes de su grupo también lo alcanzan.

Los esfuerzos colaborativos dan como resultado que los participantes reconozcan que todos los integrantes del grupo comparten un objetivo común (“nos salvamos juntos o nos hundimos juntos”), para que todos obtengan crédito del esfuerzo de cada uno (“tus esfuerzos me benefician y mi esfuerzo te beneficia”), reconozcan que el desempeño de cada uno es provocado mutuamente por uno mismo y sus compañeros (“en la unión está la fuerza”) y, se sientan felices y orgullosos por los logros de cualquier integrante del grupo (D. W. Johnson & Johnson, 2000).

Johnson, Johnson y Holubec señalan los componentes esenciales del aprendizaje colaborativo (D. W. Johnson et al., 1998), los cuales son:

1. **Interdependencia positiva.** Existe cuando los estudiantes perciben un vínculo con sus compañeros de grupo, de forma tal que no pueden lograr el éxito sin ellos (y viceversa), coordinando sus esfuerzos con los de sus compañeros para poder completar una tarea, compartiendo recursos, proporcionando apoyo mutuo y celebrando juntos el éxito.
2. **Interacción estimuladora (cara a cara).** La interacción cara a cara es importante porque existe un conjunto de actividades cognitivas y dinámicas interpersonales que solo ocurre cuando los estudiantes interactúan entre sí en relación con los materiales y actividades, permitiendo la posibilidad de ayudar y asistir a los demás. Esta interacción permite que los estudiantes obtengan retroalimentación de los demás y ejerzan presión social sobre los miembros poco motivados para trabajar.
3. **Responsabilidad individual y grupal.** El propósito de los grupos de aprendizaje cooperativo es fortalecer académica y latitudinalmente a sus integrantes. Por tanto, se requiere de la existencia de una evaluación del avance personal, la cual va dirigida hacia el individuo y hacia el grupo. Para asegurar que cada individuo sea valorado convenientemente se recomienda lo siguiente:
 - Evaluar cuánto del esfuerzo que realiza cada miembro contribuye al trabajo de grupo.
 - Proporcionar retroalimentación a nivel individual y grupal.
 - Auxiliar a los grupos para evitar esfuerzos redundantes por parte de sus miembros
 - Asegurar que cada miembro sea responsable del resultado final.
4. **Habilidades interpersonales y grupales.** Debe enseñarse a los alumnos las habilidades sociales requeridas para lograr una colaboración de alto nivel y para que estén motivados a emplearlas. En particular debe enseñarse a los alumnos a:
 - Conocerse y confiar unos en otros.
 - Comunicarse de manera precisa, sin ambigüedades.
 - Aceptarse y apoyarse mutuamente.

- Resolver conflictos constructivamente.
5. **Evaluación grupal.** Los miembros del grupo necesitan reflexionar y discutir entre sí el nivel de logro de sus metas y mantenimiento efectivo de relaciones de trabajo, identificando cuáles de las acciones de los miembros son útiles y cuáles no, y tomar decisiones acerca de las acciones que deben continuar o cambiar.

Por otra parte, existen tres tipos de grupos para el aprendizaje colaborativo (D. W. Johnson et al., 1998):

- **Grupos de aprendizaje cooperativo formal.** Son grupos que duran entre una clase y varias semanas. Se puede estructurar cualquier actividad académica para el aprendizaje colaborativo formal. Los grupos de aprendizaje cooperativo formal aseguran que los alumnos se involucren de manera activa en el trabajo intelectual de organizar el material, explicarlo, resumirlo e integrarlo en las estructuras conceptuales existentes.
- **Grupos de aprendizaje colaborativo informal.** Son grupos *ad hoc* que duran unos minutos dentro de una clase. Se usan durante la enseñanza directa (explicaciones, demostraciones, ejercicios grupales) para concentrar la atención de los alumnos en los materiales que deben aprender, crear un clima favorable para el aprendizaje, ayudar a establecer expectativas sobre lo que abarcará la actividad, asegurar que los alumnos procesen cognitivamente el material que se está enseñando y proporcionar el cierre a una situación educativa.
- **Grupos colaborativos de base.** Son grupos heterogéneos, de largo plazo (duran al menos un semestre) y con miembros estables. Su objetivo principal es hacer que los integrantes del grupo se ofrezcan entre sí el apoyo, la ayuda y el aliento necesarios para progresar académicamente. Los grupos de base ofrecen a los alumnos relaciones comprometidas a largo plazo.

No existe un tipo de grupo ideal, ya que la productividad de un grupo no está determinada por quiénes lo integran, sino por la forma en que trabajan juntos. En ciertos casos, en cada tipo de grupo, se puede recurrir a formar grupos homogéneos para enseñar determinadas

habilidades o para alcanzar determinados objetivos de enseñanza. Pero, en general, es conveniente recurrir a grupos heterogéneos, en los que sus integrantes proceden de diferentes ámbitos, tienen diversas aptitudes, experiencias e intereses, por los siguientes motivos (D. W. Johnson et al., 1998):

- Se exponen a ideas variadas, múltiples perspectivas y diferentes métodos de resolución de problemas.
- Se generan desequilibrios cognitivos, que estimulan el aprendizaje, la creatividad y el desarrollo cognitivo y social.
- Se involucran en un pensamiento más elaborado, dan y reciben más explicaciones y adoptan con más frecuencia puntos de vista personales para discutir el material.
- Se aumenta la profundidad de la comprensión, la calidad del razonamiento y la precisión de retención a largo plazo.

En lo referente al arreglo del aula, se sugiere acomodar círculos de trabajo con roles complementarios interconectados y rotativos entre los miembros del grupo. Estos roles dependen de la actividad y del tamaño del grupo. El docente puede seleccionar los más apropiados según la situación de aprendizaje (D. W. Johnson et al., 1998):

- Un compendiador, que se encargará de resumir las principales conclusiones o respuestas generadas por el grupo.
- Un inspector, que se asegurará que todos los miembros del grupo puedan intervenir y decir explícitamente como arribaron a una conclusión o respuesta.
- Un entrenador, que corrige los errores de las explicaciones o resúmenes de los otros miembros.
- Un elaborador, que tiene como tarea pedir a los integrantes del equipo relacionar los nuevos conceptos y estrategias con el material aprendido previamente.
- Un investigador-mensajero, que consigue los materiales que el grupo necesita y se comunica con los otros grupos y con el docente.
- Un registrador, cuya función es escribir las decisiones del grupo y editar el reporte del trabajo.
- Un animador, que refuerza las contribuciones de los miembros del equipo.

- Un observador, que cuida que el grupo esté colaborando de manera adecuada.

Es importante indicar que el aprendizaje colaborativo se sustenta en teorías cognoscitivas y constructivistas. Entre los autores que han realizado un aporte importante en estas teorías están Jean Piaget y Lev Vygotsky. Para Piaget hay cuatro factores que inciden e intervienen en la modificación de estructuras cognoscitivas: la maduración, la experiencia, el equilibrio y la transmisión social (Piaget, 1985). Estos factores se pueden propiciar a través de ambientes colaborativos. En la teoría constructivista de Vygotsky, el aprendiz requiere un agente mediador para acceder a la zona de desarrollo próximo. Este será responsable de tenderle un andamiaje que proporcione le seguridad y le permita que se apropie del conocimiento y lo transfiera a su propio entorno (Vygotsky, 1978). En la siguiente sección se explica con más detalle las teorías psicosociales de la educación de Piaget y Vygotsky.

3.1 Teorías Psicosociales de la Educación

En esta sección se discute los diversos enfoques teóricos del estudio del desarrollo infantil de Jean Piaget y Lev Vygotsky, con el fin de describir las habilidades y limitaciones que tienen los niños entre 4 y 6 años. Estos autores han dado un aporte importante al desarrollo cognitivo en la psicología educativa. Ambos autores contribuyeron con sus teorías a explicar el desarrollo humano desde el punto de vista de dos perspectivas, ayudando a entender los procesos o fases del desarrollo: la teoría psicogenética (Epistemología Genética: concepción genética-cognitiva del aprendizaje) de Piaget, la teoría sociocultural del desarrollo y del aprendizaje (escuela histórico-cultural) de Vygotsky, y sus continuadores. Estas teorías se presentan en las siguientes secciones.

3.1.1 Teoría Psicogenética de Piaget

La idea central de Piaget (1985) es que se debe comprender el desarrollo mental del niño para conocer su naturaleza y su funcionamiento como adulto. El desarrollo mental consiste en una evolución hacia una forma de equilibrio final representada por el adulto.

El desarrollo debe ser un equilibrio progresivo: mental, emocional, física, sentimental y social, donde el desarrollo mental es una continua construcción. Piaget argumenta que "...las funciones superiores de la inteligencia y de la afectividad tienden hacia un 'equilibrio móvil', cuanto mayor es la movilidad más sano es..." (Piaget, 1985, p. 12), por lo que las estructuras variables serán las formas de organización de la actividad mental.

La teoría psicogenética de Piaget argumenta que la inteligencia proviene de la acción, es decir, los niños aprenden a través de la interacción con el medio ambiente físico únicamente. Para Piaget, el ser humano al nacer es un individuo biológico, pero reconoce que en el desarrollo del ser humano hay un proceso de socialización (Piaget, 1985).

Piaget argumenta que el aprendizaje tiene lugar después del desarrollo, por lo que hay un vínculo de dependencia: el aprendizaje depende en gran medida del desarrollo. Además, Piaget afirma que el pensamiento es la base del lenguaje, el cual depende del desarrollo cognitivo (Labinowicz, 1982).

Piaget representa una perspectiva constructivista cognitiva, donde el niño no es un simple recipiente al cual se llena de información o de conocimientos, sino que deben usarse estrategias educativas en función del desarrollo cognitivo del niño. Además, para obtener efectividad en el proceso de enseñanza-aprendizaje debe existir motivación (Labinowicz, 1982; Piaget, 1985).

Piaget describe la evolución del niño y del adolescente sobre la base del concepto de equilibrio; establece cuatro estadios del desarrollo cognitivo (Piaget, 1985): sensoriomotor (0-2 años), preoperacional (2-7 años), operaciones concretas (7-12 años) y operaciones formales (12-20 años).

Cada etapa se caracteriza por la aparición de estructuras originales, y cada una constituye una forma particular de equilibrio, buscando siempre un equilibrio cada vez mejor (Piaget, 1985), donde toda acción, pensamiento o sentimiento responde a una necesidad.

Los intereses de un niño dependen del conjunto de sus nociones adquiridas y de sus disposiciones afectivas, ya que se tiende a complementarlas para un mejor equilibrio. Piaget argumenta que “...toda necesidad tiende: primero a incorporar las cosas y las personas a la actividad propia del sujeto, y por tanto a asimilar el mundo exterior a las estructuras ya construidas, y segundo a reajustar estas en función de las transformaciones experimentadas, y por tanto a “acomodarlas” a los objetos externos.” (Piaget, 1985, p. 16).

Los estadios de desarrollo que establece Piaget son los siguientes (Piaget, 1985):

1. **Sensorio motor.** En este primer estadio los niños comprenden el mundo básicamente por las acciones que realizan en él. Es el periodo que inicia con el nacimiento y termina con la adquisición del lenguaje (0 a 2 años), donde ocurre un extraordinario desarrollo mental. Este periodo consiste en una conquista, mediante las percepciones y los movimientos, del universo que lo rodea (Labinowicz, 1982; Piaget, 1985). El recién nacido refiere todo a sí mismo, a su propio cuerpo. Las tres etapas entre el origen y el final de este periodo son:
 - Los reflejos innatos, primeras tendencias instintivas y emociones.
 - La organización de las percepciones y primeros hábitos motores, primeros sentimientos diferenciados.
 - La inteligencia sensorio-motriz, de las primeras fijaciones exteriores de la afectividad.
2. **Preoperatorio.** Este es el estadio de la inteligencia intuitiva, del desarrollo de los sentimientos interindividuales espontáneos y de las relaciones sociales de sumisión al adulto. Este periodo comprende de los 2 a los 7 años. El niño ahora puede usar representaciones (pensamiento) en lugar de acciones abiertas para solucionar problemas. Con la aparición del lenguaje las conductas se modifican en su aspecto afectivo e intelectual. Según Piaget: “...el niño es capaz, mediante el lenguaje, de reconstituir sus acciones pasadas bajo la forma de relato y de anticipar sus acciones futuras mediante la representación verbal” (Piaget, 1985, p. 28). Se derivan tres consecuencias esenciales para el desarrollo mental: principio de socialización, aparición del pensamiento, e interiorización de la acción (perceptiva y motriz).

3. **Operaciones concretas.** En este estadio se dan las operaciones intelectuales concretas y los sentimientos morales y sociales de cooperación. Comprende el periodo de escolaridad del infante (entre 7 y 12 años), que señala un giro decisivo en el desarrollo mental del niño. Con la llegada de las operaciones el niño adquiere un sistema de acciones internas mentales necesarias para la resolución lógica de problemas.
4. **Operaciones formales.** Estadio de las operaciones intelectuales abstractas, de la formación de la personalidad y de la inserción afectiva e intelectual a la sociedad de los adultos. Este estadio empieza a los 12 años y termina a los 20 años, y reconoce como adolescencia. El nuevo desarrollo de operaciones conduce a la capacidad para el razonamiento hipotético deductivo.

Con el enfoque teórico del estudio del desarrollo infantil de Piaget, se procede a mostrar el enfoque de Vygotsky.

3.1.2 Teoría Sociocultural de Vygotsky

Vygotsky argumenta que el conocimiento es un proceso de interacción entre el sujeto y el medio, entendido físico, lógico y socialmente, representando una perspectiva constructivista sociocultural. Al nacer el ser humano es un individuo social, por lo que en el desarrollo del ser humano hay un proceso de diferenciación social.

A diferencia de Piaget, Vygotsky argumenta que el aprendizaje ocurre antes del desarrollo. Se aprende por medio de la historia y el simbolismo; es decir, el aprendizaje promueve y posibilita el desarrollo, por lo cual son dos procesos distintos que interactúan (son paralelos). Vygotsky argumenta que "... el proceso de maduración prepara y posibilita un determinado proceso de aprendizaje, mientras que el proceso de aprendizaje estimula, por así decirlo, el proceso de maduración y lo hace avanzar hasta cierto grado." (Vygotsky, Leontiev, & Luria, 2011, p. 27).

Vygotsky argumenta que no hay etapas (o estadios) preestablecidas, lo que se da es un desarrollo continuo. Además, afirma que el lenguaje es un medio para planear y resolver problemas, y es la base del pensamiento. Mediante el lenguaje se interactúa socialmente desarrollándose así el pensamiento.

Vygotsky introduce el concepto de zona de desarrollo próximo (ZDP), distancia entre el nivel real de desarrollo y el nivel de desarrollo potencial. Para determinar la ZDP hay que tener presentes dos aspectos: la importancia del contexto social y la capacidad de imitación (Vygotsky et al., 2011).

El enfoque de la ZDP es hacia al niño que será, no al niño que es. La ZDP no tiene límites. Las implicaciones de la ZDP en la enseñanza-aprendizaje son ayudar al niño a alcanzar sus objetivos, determinar lo más adecuado para el desarrollo infantil, y evaluar de forma dinámica a los niños. Otros conceptos que define Vygotsky en relación con la ZDP son (Vygotsky et al., 2011):

- **Zona de desarrollo real (efectivo).** Zona donde el niño es capaz de hacer y aprender por sí mismo, según los conocimientos y las habilidades que posee.
- **Zona de desarrollo potencial.** Zona donde el niño alcanza nuevos aprendizajes, con el apoyo y la intervención de otros.

Además, Vygotsky introduce los procesos psicológicos elementales y superiores, y las funciones mentales inferiores y superiores (Vygotsky et al., 2011).

Vygotsky realiza una clasificación de procesos psicológicos (PP) en donde señala que el desarrollo intelectual del individuo no puede entenderse como independiente del medio social en el que está inmersa la persona (Vygotsky et al., 2011). Los procesos psicológicos se dividen en (Vygotsky et al., 2011):

- **Elementales.** Los procesos psicológicos elementales (PPE) son comunes al hombre y a otros animales superiores, y se corresponden con la línea natural de desarrollo. Ejemplos de PPE son la memoria y la atención (Vygotsky et al., 2011).

- **Superiores.** Los procesos psicológicos superiores (PPS), que se caracterizan por ser específicamente humanos, se desarrollan en los niños a partir de la incorporación de la cultura. Desde este punto de vista, las interacciones sociales y las formas de mediación semiótica son las unidades de análisis de base sobre las cuales se explican los procesos de subjetivación individual. Diferentes experiencias culturales pueden producir diversos procesos de desarrollo.

Finalmente, para Vygotsky existen los siguientes dos tipos de funciones mentales (Vygotsky et al., 2011):

- **Inferiores.** Son las funciones naturales y están determinadas genéticamente, con ellas nacemos, por ejemplo: sensación, atención reactiva, memoria espontánea, inteligencia sensomotora. El comportamiento derivado de las funciones mentales inferiores es limitado. Está condicionado por lo que podemos hacer y limitan nuestro comportamiento a una reacción o respuesta al ambiente.
- **Superiores.** Son funciones mentales que se adquieren y se desarrollan a través de la interacción social. Se pueden mencionar las conductas: deliberadas, mediadas e interiorizadas. Puesto que el individuo se encuentra en una sociedad específica con una cultura concreta, estas funciones mentales están determinadas por la forma de la sociedad: son mediadas culturalmente.

De acuerdo con esta perspectiva, el ser humano es, ante todo, un ser cultural y esto es lo que establece la diferencia entre el ser humano y otro tipo de seres vivos.

El punto central de esta distinción entre funciones mentales inferiores y superiores es que el individuo no se relaciona únicamente en forma directa con su ambiente, sino también a través de y mediante la interacción con los demás individuos (Vigotsky et al., 2011).

Sintetizando, a continuación, se presenta una comparación de las teorías de Piaget y Vygotsky.

3.1.3 Comparación de las Teorías de Piaget y Vygotsky

Las teorías de Piaget y Vygotsky describen: ¿cómo es la concepción de los alumnos con respecto al aprendizaje? Desde sus distintas acepciones, este modelo considera que la construcción se produce:

- Cuando el sujeto interactúa con el objeto del conocimiento (Piaget & Cook, 1954).
- Cuando esto lo realiza en interacción con otros (Vygotsky, 2010).

No se puede dogmatizar que uno de estos postulados sea correcto y el otro no. Se piensa que los dos pueden ser correctos, y que estas premisas son complementarias y enriquecen el método de la enseñanza si se llevan a cabo simultáneamente. También se cree que esto se puede lograr sin ningún problema pues no son principios aislados sino suplementarios (Díaz Barriga & Hernández Rojas, 2002).

La teoría de Piaget resalta los conceptos fundamentales: adaptación, organización, asimilación, acomodación y equilibrio, mientras que la teoría de Vygotsky resalta las funciones mentales (inferiores y superiores), las habilidades psicológicas, la conversación privada, la zona de desarrollo próximo, las herramientas del pensamiento y la mediación.

Una diferencia importante entre Piaget y Vygotsky es la relación entre desarrollo y aprendizaje, y el papel del lenguaje. El concepto de aprendizaje apunta a los procesos mediante los cuales las personas incorporamos nuevos conocimientos, valores y habilidades que son propios de la cultura y la sociedad en que vivimos (León Sáenz, 2010). Cuando se habla de desarrollo se trata explícitamente de la formación progresiva de las funciones propiamente humanas: lenguaje, razonamiento, memoria, atención, entre otras. Se trata del proceso mediante el cual se ponen en marcha las potencialidades de los seres humanos (León Sáenz, 2010).

Piaget apunta una dependencia entre desarrollo y aprendizaje. Afirma que el aprendizaje tiene lugar después del desarrollo, y que el pensamiento es la base del lenguaje, el cual depende del desarrollo cognitivo. Por otro lado, Vygotsky argumenta que el aprendizaje ocurre antes

del desarrollo, y que los dos procesos que interactúan son paralelos. Afirma que el lenguaje es un medio para planear y resolver problemas, la base del pensamiento, y que el lenguaje provoca la interacción social.

Otra diferencia es que Piaget divide en cuatro estadios el desarrollo humano, y argumenta que se debe enseñar a los niños en formas asimilables a su desarrollo, desde un punto de vista del individuo. Mientras que Vygotsky argumenta que no hay etapas (o estadios) preestablecidas, lo que se da es un desarrollo continuo. Recalca que el aprendizaje debe ser congruente con el nivel de desarrollo del niño, aunque puede adelantarse al desarrollo, produciéndose más fácilmente en situaciones colectivas.

Se puede decir que la postura de Piaget se enfoca simplemente en lo biológico. Cree que el ser humano cuando nace es un ser meramente biológico que se irá desarrollando de manera precisa de acuerdo a los estadios que se han mencionado, y que este será el factor determinante de su progreso cognitivo. Si bien la teoría piagetiana de los estadios fue un gran avance para la ciencia y la educación, no se puede dejar de lado las relaciones sociales.

Fue precisamente Vygotsky quien puso especial énfasis en este último aspecto. Declaró que la potencialidad cognoscitiva del sujeto depende de la calidad de su interacción social y de su ZDP. Pero no hay que dejar de lado que la edad del sujeto tiene serias implicaciones en su desarrollo intelectual. Por tanto, se puede concluir que Piaget y Vygotsky tienen razón parcialmente, pero hay que unificar ambos postulados para obtener un efecto óptimo.

Se puede deducir que cada uno hizo buenas aseveraciones, pero no son del todo acertadas y deben integrarse. Se quiere aquí hacer nuevamente hincapié en que estas dos teorías son útiles por igual, pero deben complementarse.

Por último, tanto Piaget como Vygotsky piensan que el aprendizaje es una construcción mental, donde los niños aprenden como resultado de aumentar o integrar al conocimiento actual la nueva información. Por tanto, ambos autores explicaron el desarrollo infantil y la importancia del aprendizaje en la primera infancia, ayudando a entender los procesos o fases

del desarrollo. A continuación, se describe brevemente el desarrollo de los niños entre 4 y 6 años.

3.2 Desarrollo de la Niñez entre 4 y 6 años

El desarrollo humano considera todos los procesos y el cambio que se dan en el individuo a partir de su concepción y hasta su muerte, y que progresivamente van formando al ser humano en sus diversas etapas (León Sáenz, 2010).

Los primeros seis años de vida constituyen una etapa importante caracterizada por el rápido crecimiento físico, mental, social y emocional. Es por ello, que el abordaje integral de los niños en estas edades, debe visualizarse desde una perspectiva no solo asistencial, sino con un fuerte componente pedagógico, que involucre las diferentes áreas del desarrollo infantil, los derechos de los niños, la satisfacción de las necesidades básicas, así como el respeto por el contexto social, cultural e histórico de pertenencia (Arce-Ulloa, Blanco-García, Cerdas-González, & Zúñiga-León, 1995; Cerdas-González & Mata-Solano, 2001).

En el proceso de desarrollo del niño, intervienen una gran cantidad de factores que pueden ser agrupados en dos categorías básicas: los que se consideran fundamentalmente biológicos y los que se perciben principalmente como sociales. Además, existe un tercer componente que se debe tomar en cuenta al estudiar el desarrollo infantil: la participación activa y no solamente pasiva que tiene el niño. El niño conforme acumula sus experiencias, va formando su propia estructura mental y física, única y que constituye su componente “individual” de desarrollo (León Sáenz, 2010).

El desarrollo humano es complejo debido a la multiplicidad de sucesos simultáneos que se dan, como los factores que afectan este proceso (León Sáenz, 2010). Apunta Piaget (1985) que el desarrollo psíquico se inicia al nacer y concluye en la adultez, y consiste básicamente, en una evolución hacia una forma de equilibrio final. La importancia de conocerlo y estudiarlo radica en establecer las condiciones necesarias para lograr un desarrollo óptimo, que le permita realizar sus potencialidades en cada fase de su vida (León Sáenz, 2010).

Así, el desarrollo humano y en particular el desarrollo infantil (especialmente de 4 a 6 años) se puede ver como "...la construcción progresiva de una edificación, en la cual la estructura y las bases se establecen al principio..." (León Sáenz, 2010, p. 19), como sucede en los 6 primeros años de vida del ser humano. Sobre estos cimientos se asientan el desarrollo humano. Por esto es importante conocer, estudiar y fomentar las condiciones indispensables para el óptimo desarrollo de cada niño (Ginsburg & Opper, 1988; León Sáenz, 2010; Piaget & Cook, 1954; Vygotsky et al., 2011).

En la Tabla 5 se muestra un resumen del desarrollo de los niños entre 4 y 6 años de acuerdo a las tres áreas de desarrollo humano infantil: cognoscitiva-lingüística, socioemocional y psicomotriz (Arce-Ulloa et al., 1995; Cerdas-González & Mata-Solano, 2001). Las argumentaciones mostradas en la Tabla 5, respecto al desarrollo de los niños entre los 4 y 6 años, revelan, de manera didáctica, cómo los seres humanos evolucionan con el tiempo. Las edades cronológicas son únicamente un punto de referencia para explicar acontecimientos críticos que suceden a lo largo de estos años.

Es importante destacar, durante este periodo de desarrollo, que los niños son egocéntricos. Según Piaget, el egocentrismo es el estado natural de los niños pequeños, que no pueden imaginar el mundo desde otro punto de vista más que del suyo (Piaget, 1985). Es por eso que los niños en edad preescolar tienen dificultades para distinguir sus propias necesidades de las de los demás. No obstante, si el niño de preescolar empieza a realizar acciones a favor de los demás, su egocentrismo inicial cede el paso a la conducta cooperadora (Zurbano Díaz de Cerio, 1998). Esta conducta se puede potencializar incorporando actividades de aprendizaje colaborativo en la vida cotidiana del niño.

Tabla 5. Desarrollo de la niñez entre 4 y 6 años según las áreas de desarrollo.

Área de Desarrollo	Niños entre 4 y 5 años	Niños entre 5 y 6 años
Cognoscitiva-Lingüística	<ul style="list-style-type: none"> • No es capaz de dar explicaciones sobre las cosas que pasan. Lo que hacen es unir cosas sin entender la causa real: “las nubes se mueven porque yo me muevo”. • Piensa únicamente en lo que percibe y esto, por ejemplo: si se llena de agua un vaso y se echa la misma cantidad en una botella, dirá que hay más agua en el vaso que en la botella, porque el vaso está más lleno y la botella apenas tiene agua. • La articulación del niño experimenta una notable mejoría. Van desapareciendo, aquellas incorrecciones en la pronunciación que hacía el niño en un intento de imitar palabras largas que había oído al adulto o los sonidos que se presentaban complicados. Respeta el orden de las sílabas al pronunciar y no modifica los sonidos. Empieza a dominar la pronunciación, aunque a veces cometa errores. • La utilización de otras partes de la oración (artículos, pronombres, preposiciones y adverbios), que dan nuevas posibilidades a su forma de hablar, se van consolidando y los usa de la misma forma que el adulto. • Abandona las imperfecciones que tenía, usando bastantes verbos en el tiempo correcto, y no comete fallos en la conjugación. Alarga las frases y expresa en alguna de ellas relaciones de causa y consecuencia: “gana porque va deprisa”, “es malo, por eso le pego”. • Su lenguaje le afirmar, exclamar, negar; pero con una mayor precisión. Es capaz de incluir ya la negación dentro del enunciado de la frase. Ha dejado de decir “sopa no” para decir “no quiero sopa”. • Su vocabulario es amplio y le gusta explorar el lenguaje de los demás aprendiendo palabras nuevas, algunas de ellas por su peculiar sonido y reacción del adulto, le hacen gracia y le resultan divertidas. Disfruta con su repertorio y quiere enriquecerlo más. 	<ul style="list-style-type: none"> • Progresivamente el niño va abandonando el tipo de pensamiento que utilizaba antes y descubriendo las causas de lo que sucede a su alrededor. A esto le ayuda el hecho de estar en contacto con niños de su edad y con los adultos. • Se apoya mucho en el lenguaje y gracias a él busca explicaciones a las cosas. No obstante, le cuesta comprender las cosas que pasan y cómo suceden en el tiempo. • El niño tiene que pronunciar correctamente todos los sonidos de la lengua y sus errores habrán ido desapareciendo. No se debe olvidar que en el curso siguiente empieza de forma sistemática la lectura y la escritura, y los errores que tenga en estos aspectos, le pueden perjudicar notablemente. • No obstante, no alarmarse en el caso de que algunos niños tengan dificultades en la pronunciación de la “r”, la “s”, la “z” o “ch”; el ritmo de todos los niños no es el mismo y habrá algunos a los que cueste más pronunciar estos sonidos y necesitarán más tiempo para conseguirlo. • En la utilización de artículos, pronombres, preposiciones y adverbios, así como en los verbos y su conjugación, el dominio debería ser total y no existir ningún tipo de problemas. • En el tipo de frases que usa aparecen ya construcciones que expresan tiempo: “juego un poco con mis coches antes de ir a comer”, “voy a jugar después de los dibujos”. • El aumento de vocabulario se sigue produciendo, pero no de forma tan vertiginosa como en los años anteriores. No obstante, el número de palabras que el niño usa en esta edad se sitúa alrededor de las dos mil quinientas, que corresponden a las preocupaciones e intereses del niño.

Tabla 5. Desarrollo de la niñez entre 4 y 6 años según las áreas de desarrollo. (cont.)

Área de Desarrollo	Niños entre 4 y 5 años	Niños entre 5 y 6 años
Socioemocional	<ul style="list-style-type: none"> • Gracias a los avances en el desarrollo de su pensamiento, el niño comienza a comprender el mundo que le rodea (conoce a los miembros de la familia, el nombre de sus compañeros de clase) y el lugar que ocupa en él (el nombre de la localidad o barrio y su dirección). • Sigue dando muestras de su incipiente “personalidad”: <ul style="list-style-type: none"> ○ Se siente orgulloso de sus propias creaciones ○ Llama la atención sobre lo que hace, le gusta exhibirse y se manifiesta seductor frente a los adultos. • Se identifica con los adultos y tiende a imitarlos, le gusta hacer las cosas de los “mayores”, por ejemplo, ponerse los zapatos de mamá. • Es capaz de mantener por breves instantes un juego de reglas sencillas, aunque al final surge la anarquía y cada cual juega a lo suyo. • Siente interés por las diferencias anatómicas de los sexos. • Hábitos de autonomía: <ul style="list-style-type: none"> ○ Puede comer completamente solo usando mejor los cubiertos. ○ Se viste y desviste sin ayuda si es una ropa fácil de poner. ○ Se lava las manos, la cara y los dientes. ○ Aprende a atarse los zapatos. ○ Duerme toda la noche de un tirón, aunque dilata el momento de ir a la cama. Pide que le narren cuentos. ○ Puede recoger y ordenar los juguetes. 	<ul style="list-style-type: none"> • La relación familiar sigue siendo positiva, centrada especialmente en la madre, aunque el padre adquiere cada vez mayor importancia para el niño. • Se muestra servicial, tiende a agradar porque es dependiente. Además, es capaz de mantener un diálogo con cualquier persona adulta. • Con sus hermanos pequeños adopta una actitud proteccionista, con los otros niños juega y empieza a aceptar las normas y reglas de los juegos. • Siente fascinación por la televisión, especialmente por los programas infantiles. • Hábitos de autonomía: <ul style="list-style-type: none"> ○ Mejora el comportamiento en la mesa (no derrama la comida, ni se mancha como antes). ○ Es capaz de manejar dinero y realizar pequeñas compras. ○ Cruza la calle de forma adecuada. ○ Usa correctamente los materiales de clase y los comparte con sus compañeros.
Psicomotriz	<ul style="list-style-type: none"> • En esta edad el niño perfecciona sus movimientos, se cae menos veces. • Mejora mucho su habilidad manual y puede realizar actividades escolares como: recortar, puntear y colorear con cierta precisión de movimientos. 	<ul style="list-style-type: none"> • El dominio del cuerpo se consigue casi totalmente: salta, sube rampas, trepa, corre por las escaleras y se cansa poco porque sus movimientos son más ajustados. • Usa preferentemente la misma mano para realizar actividades: recortar, dibujar, dar cartas, pintar, entre otros. • Aquellas actividades que exigen habilidad manual son cada vez más precisas (dibujar, puntear, recortar, entre otros). Estos avances son imprescindibles para el inicio del aprendizaje de la escritura.

En el desarrollo infantil y la educación de los niños de la primera infancia, Prensky apunta que la tecnología es una aliada, ya que podría desempeñar un papel importante en la reestructuración de procesos de enseñanza-aprendizaje para estar mejor preparados para los retos del futuro. Ya que se vive en un mundo donde la tecnología controla cada vez más nuestras vidas, el incorporar el uso de tecnología en la educación desde temprana edad se vuelve importante (Prensky, 2001). Por otra parte, la tecnología apoya y beneficia el aprendizaje colaborativo, ya que permite desarrollar, extender y profundizar las habilidades interpersonales y asimila las barreras culturales a medida que estudiantes y docentes aprenden a comunicarse mediante las nuevas formas que propone este medio. Por ello, a continuación, se discute sobre el aprendizaje colaborativo apoyado por computadora.

3.3 Aprendizaje Colaborativo Apoyado por Computadora

Uno de los requisitos básicos de la educación para el futuro es preparar a los alumnos para participar en una sociedad de la información en red, en la que el conocimiento será el recurso más importante para el desarrollo social y económico. Las instituciones educativas se ven obligadas a encontrar mejores métodos pedagógicos para hacer frente a estos nuevos desafíos. En este desarrollo se espera que las computadoras desempeñen un papel importante en la reestructuración de procesos de enseñanza-aprendizaje.

El aprendizaje colaborativo apoyado por computadora (ACAC) es una de las ideas más prometedoras para mejorar el proceso de enseñanza-aprendizaje con la ayuda de la información y la tecnología de comunicación (Gifford & Enyedy, 1999; Kolodner & Nagel, 1999; Muukkonen, Hakkarainen, & Lakkala, 1999; G. Stahl, 1999).

El ACAC se da cuando el uso de la tecnología se suma al uso de grupos de aprendizaje colaborativo. Desde el punto de vista pedagógico, la tecnología brinda ventajas para el proceso de aprendizaje colaborativo (Alvarez, Alarcon, & Nussbaum, 2011):

- **Estimular la comunicación interpersonal.** Posibilita el intercambio de información y el diálogo y discusión entre todas las personas implicadas en el proceso.

- **Facilitar el trabajo colaborativo.** Permite que los estudiantes compartan información, trabajen con documentos conjuntos y faciliten la solución de problemas y toma de decisiones.
- **Seguir el progreso del grupo a nivel individual y colectivo.** Permite que los docentes pueden monitorear a los estudiantes para ver el progreso como grupo e individual.
- **Acceder a información y contenidos de aprendizaje.** Posibilita obtener información y contenidos de aprendizaje a través de diferentes medios digitales.
- **Administrar información de estudiantes.** Permite el acceso a toda la información vinculada con el estudiante, que le pueda ser útil al docente en un momento dado, para integrar grupos o para facilitar su desarrollo y consolidación.
- **Crear ejercicios de evaluación y autoevaluación.** El docente podrá conocer el nivel de logro y rediseñar la experiencia según su ritmo y nivel. Al estudiante le ofrecerá retroalimentación sobre el nivel de desempeño.

El aprendizaje colaborativo describe una situación donde se espera la interacción entre las personas, lo que debe dar lugar a mecanismos de aprendizaje, pero no hay garantía de que la interacción que se esperaba ocurra realmente (Dillenbourg, 1999). Por tanto, una preocupación general consiste en desarrollar formas de aumentar la probabilidad de que se produzcan algunos tipos de interacción. También es necesario que los docentes y los estudiantes asuman nuevos roles en una actividad de colaboración (Collazos, Guerrero, & Vergara, 2001), para definir nuevos esquemas que permitan estructurar las actividades que involucran elementos de colaboración. Por lo cual, la tecnología se vuelve una aliada estratégica en el proceso del aprendizaje colaborativo.

Dillenbourg y otros afirman que, desde hace muchos años, las teorías del aprendizaje colaborativo se han centrado en cómo los individuos trabajan en grupo, y solo recientemente, se han centrado en el propio grupo, tratando de establecer cuándo y bajo qué circunstancias el aprendizaje colaborativo es más efectivo que el aprendizaje individual (Dillenbourg et al., 1995). En el contexto del ACAC, algunas de las variables independientes que se han

identificado y estudiado ampliamente son el tamaño y la composición del grupo, la naturaleza y los objetivos de la tarea, los medios de comunicación y los canales de comunicación, la interacción entre pares, el sistema de recompensas y las diferencias de sexo, entre otros (Adams & Hamm, 1996; Collazos, Guerrero, Llaña, & Oetzel, 2001; Dillenbourg et al., 1995; Slavin, 1991; Underwood, Mc.Caffrey, & Underwood, 1990).

En la sección siguiente, se describe una forma específica del aprendizaje colaborativo apoyado por computadora: el aprendizaje colaborativo móvil. Finalmente, para concluir el capítulo, se describe la automatización de actividades del aprendizaje colaborativo para crear herramientas colaborativas.

3.3.1 Aprendizaje Colaborativo Móvil

Los avances en las tecnologías y dispositivos móviles en la última década, han incrementado las oportunidades de realizar investigación en *m-learning* (*mobile learning*). Esto ha provocado un incremento en el número de centros educativos que adoptan herramientas móviles como apoyo al proceso de aprendizaje (Fetaji & Fetaji, 2011).

Entre las ventajas del aprendizaje móvil, Fetaji y Fetaji apuntan lo siguiente (Fetaji & Fetaji, 2011):

- Muchos estudios han dado resultados alentadores del uso de tecnologías móviles para apoyar el proceso de enseñanza-aprendizaje.
- Las tecnologías móviles ayudan a optimizar los procesos y servicios de aprendizaje a través de una mayor flexibilidad. El aprendizaje puede estar disponible de forma inmediata en cualquier momento y en cualquier lugar.
- Las tecnologías móviles permiten un aprendizaje ubicuo y móvil.
- Las tecnologías móviles ofrecen un modelo “lo justo, justo a tiempo, justo para mí”, o sea, un aprendizaje flexible.

- Las características de las tecnologías móviles e inalámbricas de portabilidad, flexibilidad, accesibilidad inmediata, personalización y accesibilidad son apropiadas, y mejoran el proceso de aprendizaje.
- Las tecnologías móviles apoyan una concepción de enseñanza centrada en el docente y en el contenido.
- Los dispositivos móviles realizan muchas de las funciones de las computadoras, con las ventajas que da su simplicidad y el acceso mejorado (en cualquier lugar y en cualquier momento), a excepción de las capacidades de ancho de banda, el costo y capacidades de entrada, que en muchos casos son las limitaciones documentadas.
- Los dispositivos móviles proveen mayor interacción con los materiales educativos.
- Los dispositivos móviles, además de funciones de texto, pueden tener acceso a Internet y capacidades de audio y video.

Estos mismos autores también identificaron problemas e inconvenientes del aprendizaje móvil (Fetaji & Fetaji, 2011):

- La diversidad de plataformas y sistemas operativos y las características (limitaciones) utilizados por diferentes modelos, influyen directamente y hacen que el uso de dispositivos móviles en el proceso de enseñanza-aprendizaje sea difícil. Sin embargo, la tecnología avanza rápidamente, y las limitaciones documentadas no serán obstáculo en un futuro próximo.
- Las limitaciones de hardware y software de los dispositivos móviles existentes provocan que no se utilicen en el proceso de enseñanza-aprendizaje.
- Muchos navegadores móviles no admiten secuencias de comandos o *plugins*, y no tienen memoria disponible para mostrar algunas páginas de computadoras con gráficos.

La mayoría de los problemas planteados hacen referencia a problemas de usabilidad de los dispositivos móviles, pero la mayoría de ellos no serán obstáculos en un futuro próximo, debido a la rápida evolución de la tecnología.

Aun así, estos problemas deben considerarse y examinarse con cuidado durante las pruebas de usabilidad de una aplicación móvil, con el fin de seleccionar una metodología de investigación adecuada y reducir el efecto de los factores contextuales en los resultados de las pruebas (Coursaris & Kim, 2007; Fetaji & Fetaji, 2011).

Hamdan y Schaper exponen que el aprendizaje móvil se convierte en interactivo cuando se usa en una herramienta de colaboración. Esto conlleva a la unión del aprendizaje colaborativo y el aprendizaje móvil (Hamdan & Schaper, 2012)

En la actualidad, los niños son usuarios frecuentes y experimentados de la tecnología (Druin, 1999; Prensky, 2001). Es por esto que Druin piensa que los niños deben tomarse en cuenta en todo el proceso de desarrollo de la tecnología actual, por lo cual describe técnicas de investigación colaborativa que desarrolla nuevas tecnologías para y con los niños (Druin, 1999; Guha et al., 2004).

La investigación colaborativa es un enfoque de la investigación que incluye tres aspectos fundamentales de la literatura de interfaces humano-computador (Druin, 1999):

1. Asociación multidisciplinaria con niños.
2. Investigación de campo, dando importancia a la comprensión de contexto, las actividades y las herramientas.
3. Prototipos iterativos de baja tecnología y de alta tecnología.

Este enfoque de la investigación colaborativa se ha convertido en una realidad para realizar actividades de colaboración con los niños. El objetivo del desarrollo de la investigación colaborativa es encontrar técnicas intergeneracionales que apoyen a los equipos de diseño en la comprensión de lo que los niños como usuarios de la tecnología hacen ahora, lo que podrían hacer mañana, y lo que se imaginan en un futuro (Druin, 1999).

Druin también define el concepto de diseño centrado en el aprendizaje como los resultados del aprendizaje que pueden estar relacionados con el proceso de investigación colaborativa.

El diseño centrado en el aprendizaje se puede producir en niños y adultos, principiantes y expertos en tecnología y técnicos, y profesionales no técnicos (Druin, 1999).

En resumen, el uso de las técnicas de investigación colaborativa se puede trabajar con los niños durante todo el proceso de desarrollo de tecnologías, como socios de diseño, usuarios, informantes o evaluadores de lo que se va creando junto con los adultos encargados del proyecto. Entre las técnicas que se pueden usar están (Guha et al., 2004):

- **Diseño participativo.** Los niños dibujan ideas para crear prototipos durante la lluvia de ideas.
- **Investigación de observación.** Los niños y los adultos del equipo de diseño observan a otros niños, que no están relacionados con el proceso, usando tecnología y, luego los niños que si están relacionados con el proceso de diseño realizan dibujos acerca de lo que vieron y los adultos redactan lo que observaron.

La investigación ha mostrado que estos métodos funcionan, solo si son modificados según el rango de edad de los participantes (Guha et al., 2004).

Por otra parte, Fails enfatiza que la interacción social y la colaboración son esenciales en el desarrollo cognitivo y emocional de los niños pequeños, y estas capacidades pueden ser logradas gracias a los dispositivos móviles, ya que pueden apoyar al aprendizaje haciendo que los niños construyan cosas en distintos contextos (Fails, 2007). Además, el uso creativo, constructivo y de colaboración de los dispositivos móviles puede mejorar el aprendizaje en contextos bien diseñados.

A continuación, se describe la automatización de actividades del aprendizaje colaborativo para crear herramientas colaborativas.

3.3.2 Automatización de Actividades del Aprendizaje Colaborativo

La colaboración y la ayuda mutua se deben enseñar desde edades tempranas, ya que los niños de la primera infancia, debido a su característico egoísmo de la edad, lo quieren todo para sí mismos evitando compartir. Pero cuando empiezan a realizar acciones a favor de los demás, su egoísmo disminuye, permitiendo que crezca el comportamiento colaborativo (Zurbano Díaz de Cerio, 1998). Además, la utilización de tecnología en el proceso logra facilitar el trabajo colaborativo, ya que puede permitir que la mediación sea efectiva en los roles que se establecen, y asegura que cada integrante cumpla con el rol y las tareas asignadas (Verdugo, Barros, Albornoz, Nussbaum, & McFarlane, 2014). Aquí radica la importancia de la incorporación del aprendizaje colaborativo apoyado por computadora (ACAC) en el proceso de enseñanza y aprendizaje, y hacerlo desde edades tempranas.

En contextos ACAC, las interacciones del grupo entre los individuos son mediadas por ambientes computacionales. El lado tecnológico del ACAC se enfoca en el diseño y estudio de tecnologías fundamentalmente sociales. Esto significa que la tecnología está diseñada específicamente para mediar y fomentar actos sociales que constituyen el aprendizaje del grupo y lleva a un aprendizaje individual (Gerry Stahl, Koschmann, & Suthers, 2006). Se debe recordar que el aprendizaje colaborativo es constituido de las interacciones entre los participantes.

El proceso del aprendizaje colaborativo (AC) se divide en tres fases de acuerdo con su ejecución temporal: pre-proceso, en-proceso y pos-proceso. Las tareas de pre-proceso son principalmente actividades de coordinación y definición de estrategia y las tareas de pos-proceso se refieren principalmente a las actividades de evaluación. Ambas fases, pre-proceso y pos-proceso, se cumplirán en su totalidad por el mediador. Los miembros del grupo se encargarán de las tareas relativas a la fase en-proceso, en gran medida. Con el fin de especificar esta división, las actividades que se llevan a cabo en el aprendizaje colaborativo y la clasificación de cada actividad según la fase se presentan a continuación (resumen en la Tabla 6) (Adams & Hamm, 1996; Collazos, 2003):

1. Diseñar el contenido y las tareas principales para alcanzar los objetivos por los grupos colaborativos (pre-proceso).
2. Especificar el tamaño de los grupos, se sugiere de 3 a 6 personas, dependiendo de la naturaleza de la tarea y el tiempo disponible (pre-proceso).
3. Organizar los grupos, es decir, designar a los estudiantes para conformar cada grupo o permitir que se formen a voluntad de los estudiantes (pre-proceso).
4. Organizar el espacio físico donde se realizarán las actividades del aprendizaje colaborativo. El mediador debe estar seguro que las actividades deben ser alcanzables por todos los grupos. Los colaboradores deben sentarse juntos sin interrupciones de otros grupos (pre-proceso).
5. Distribuir el material de instrucción (pre-proceso).
6. Especificar los roles de los miembros de cada grupo (pre-proceso).
7. Especificar las directivas de la tarea, es decir, definir las reglas a seguir (pre-proceso).
8. Aplicar estrategias como la interdependencia positiva de la meta, la motivación entre pares y el apoyo para el aprendizaje. Crear tareas con un objetivo común donde las recompensas se basen en los resultados individuales y de grupo (se definen en la fase pre-proceso, pero se evalúan en la fase en-proceso).
9. Organizar la cooperación intra-grupo, es decir, definir las estrategias de colaboración que van a ser utilizadas por los miembros del grupo (pre-proceso, la definición de las estrategias de colaboración ocurre en la fase en-proceso).
10. Probar los criterios de éxito que explican las pautas, los límites y los roles. Los criterios de éxito se deben definir en el inicio de la actividad, ser revisados durante la actividad para comprobar si se está alcanzando el objetivo común y, después de la actividad, para comprobar si se ha alcanzado el objetivo común (pre-proceso, en-proceso y pos-proceso).
11. Determinar el comportamiento deseado (pre-proceso, la definición del comportamiento deseado se produce en la fase en-proceso).
12. Monitorear los grupos, por ejemplo: comprobar que el punto anterior se cumple (en-proceso).

13. Proporcionar asistencia cuando alguien lo solicita: se proporciona a todo el grupo por el mediador o pares (en-proceso).
14. Intervenir cuando los grupos tienen problemas para colaborar (en-proceso).
15. Terminar una actividad (pos-proceso).
16. Evaluar la calidad del aprendizaje logrado por los estudiantes (pos-proceso).
17. Alentar a los estudiantes para llevar a cabo una evaluación del trabajo en conjunto de cada grupo (al final de la fase en-proceso).
18. Proporcionar y fomentar la retroalimentación, es decir, discutir cómo las actividades podrían mejorarse (al final de la fase en-proceso).

Tabla 6. Fases y actividades del proceso del AC (Collazos, 2003).

Pre-Proceso	En-Proceso	Pos-Proceso
<ul style="list-style-type: none"> • Diseñar los contenidos • Especificar el tamaño de los grupos • Organizar los grupos • Organizar el espacio físico • Distribuir el material • Especificar los roles • Especificar las reglas a seguir • Definir los criterios de éxito • Determinar el comportamiento deseado 	<ul style="list-style-type: none"> • Aplicar estrategias (interdependencia positiva de la meta, la motivación entre pares, la ayuda para aprender) • Organizar la cooperación intra-grupo • Probar los criterios de éxito • Monitorear a los grupos • Proporcionar ayuda (mediador y pares) • Intervenir en caso de problemas • Verificar el reporte de los grupos • Proporcionar y fomentar la retroalimentación 	<ul style="list-style-type: none"> • Inspeccionar los criterios de éxito • Presentar el cierre de la actividad • Evaluar la calidad del aprendizaje

También, en el proceso de aprendizaje colaborativo, las tareas colaborativas deben cumplir los siguientes aspectos (Alvarez et al., 2011; A. Echeverría et al., 2011):

- Tener un objetivo en común.
- Permitir comunicación entre pares.
- Proveer coordinación entre pares.
- Proveer interdependencia entre pares (mutua).
- Proveer mecanismos para apoyo entre pares.
- Clarificar la responsabilidad en las acciones.

- Proveer conciencia del trabajo de los otros miembros del grupo.
- Proveer retroalimentación común.

El papel que juega el uso de tecnología en el proceso del aprendizaje colaborativo es importante, ya que en relación con los aspectos que una tarea colaborativa debe cumplir, la tecnología aporta (Alvarez et al., 2011; A. Echeverría et al., 2011):

- **Objetivo común.** La tecnología no lo establece, ya que se establece en la tarea a realizar.
- **Comunicación entre pares.** Aunque se da por los miembros del equipo, la tecnología puede ser usada por ellos para comunicar o mostrar una idea.
- **Coordinación entre pares.** La tecnología puede mediar permitiendo la coordinación y la sincronización del trabajo colaborativo de la manera en que se lleva a cabo la tarea.
- **Interdependencia entre pares.** La tecnología permite que la mediación sea efectiva en los roles que se establecen y cómo se establecen.
- **Apoyo entre pares.** La tecnología debe corregir cuando algunos de los integrantes realicen algo mal, esto impulsará a los otros pares a apoyar en la corrección, produce el espacio para que los pares se puedan apoyar. Se debe apoyar porque en los equipos hay algunos que saben más que otros y los que sabe más van a reproducir la construcción del conocimiento en aquellos que saben menos.
- **Responsabilidad en las acciones.** La tecnología permite que todo el equipo se dé cuenta de las acciones que realiza cada integrante del equipo, por lo que cada integrante asume su responsabilidad en las acciones que realiza.
- **Conciencia del trabajo de los otros miembros del grupo.** La tecnología permite que todo el equipo se dé cuenta de las acciones que realiza cada integrante del equipo y la importancia del trabajo que realiza para lograr el objetivo común.
- **Retroalimentación común.** Aunque se refleja en el éxito o el fracaso en común del equipo (se logró o no el objetivo común), la tecnología puede asegurar que se dé trabajo en equipo y que los roles estén entrelazados de tal manera que el éxito o el fracaso sea común.

Dado lo anterior, las actividades de cada fase que se pueden automatizar, permitiendo que la tecnología apoye el proceso del aprendizaje colaborativo, se muestran en la Tabla 7.

Tabla 7. Actividades del proceso del AC que se pueden automatizar.

Pre-Proceso	En-Proceso	Pos-Proceso
<ul style="list-style-type: none"> • Especificar el tamaño de los grupos • Organizar los grupos • Especificar los roles • Definir los criterios de éxito 	<ul style="list-style-type: none"> • Aplicar estrategias (interdependencia positiva de la meta, la motivación entre pares, la ayuda para aprender) • Organizar la cooperación intra-grupo • Probar los criterios de éxito • Monitorear a los grupos • Proporcionar ayuda (mediador y pares) • Verificar el reporte de los grupos • Proporcionar y fomentar la retroalimentación 	<ul style="list-style-type: none"> • Inspeccionar los criterios de éxito • Presentar el cierre de la actividad (en una pequeña parte)

En el capítulo siguiente se describen metodologías y técnicas de usabilidad, las cuales serán utilizadas en este trabajo para llevar a cabo el proceso de desarrollo de la herramienta creada como parte de esta investigación.

CAPÍTULO 4. USABILIDAD

Usabilidad es un término que se refiere a la facilidad con que las personas pueden usar una herramienta particular o cualquier otro producto tecnológico con el fin de alcanzar un objetivo concreto¹. La usabilidad también puede referirse al estudio de los principios que hay tras la eficacia percibida de un objeto.

La usabilidad ha sido en incontables ocasiones foco de discusión y un término descrito a través de diversas definiciones en la academia y en la industria durante mucho tiempo (International Organization for Standardization, 2001; Jakob Nielsen, 1993; Brian Shackel, 1991). El cambio de siglo supuso un mayor enfoque en los estudios de usabilidad para su investigación en el campo de la Interacción Humano-Computador (IHC, en inglés HCI: *Human-Computer Interaction*).

En IHC el término usabilidad hace referencia a la claridad y facilidad con que se diseña la interacción en una aplicación informática, ya sea un programa, o bien un sitio web. Nielsen definió la usabilidad como el atributo de calidad que mide lo fáciles que son de usar las interfaces de las aplicaciones (Jakob Nielsen, 1993).

El grado de usabilidad de un sistema interactivo es, por su parte, una medida empírica y relativa de su usabilidad, es decir, de su sencillez de uso y facilidad de aprendizaje (Jakob Nielsen, 1993). Es empírica porque no se basa solamente en opiniones o sensaciones, denominados datos cualitativos, sino también en pruebas de usabilidad realizadas en laboratorio u observadas mediante tests de campo, donde también se pueden obtener datos cuantitativos (por ejemplo: qué zonas observa el usuario y cuántos clics realiza para llevar a cabo una acción, entre otros). A la vez es relativa porque el resultado no es ni bueno ni malo, sino que depende de las metas y objetivos planteados (por ejemplo: al menos el 80% de los usuarios de un determinado grupo pueden realizar una tarea sin más ayuda que una guía rápida), o de una comparación con otros sistemas similares.

¹ Sitio web de Usability.gov: <http://www.usability.gov/basic/index.html>.

El modelo conceptual de la usabilidad, proveniente del Diseño Centrado en el Usuario (DCU), no está completo sin la idea de utilidad (International Organization for Standardization, 2001; Brian Shackel, 1991). En un esfuerzo de homogeneizar las diferentes premisas y definiciones de los diferentes autores y entidades. Los principios básicos de la usabilidad se pueden resumir en (International Organization for Standardization, 2001; Jakob Nielsen, 1993; Brian Shackel, 1991):

- **Facilidad de Aprendizaje.** Facilidad con la que nuevos usuarios desarrollan una interacción efectiva con el sistema o producto. Está relacionada con la predictibilidad, la capacidad de síntesis, la familiaridad, la generalización de los conocimientos previos y la consistencia.
- **Facilidad de uso.** Facilidad con la que se usa la herramienta, con menos pasos o más naturales a su formación específica. Tiene que ver con la eficacia y eficiencia de la herramienta.
- **Flexibilidad.** Relativa a la variedad de posibilidades con las que el usuario y el sistema pueden intercambiar información. También abarca la posibilidad de diálogo, la multiplicidad de vías para realizar la tarea, similitud con tareas anteriores y la optimización entre el usuario y el sistema.
- **Robustez.** Es el nivel de apoyo al usuario que facilita el cumplimiento de sus objetivos. Está relacionada con la capacidad de observación del usuario, de recuperación de información y de ajuste de la tarea al usuario.

Todos estos beneficios implican una reducción y optimización general de los costos de producción, así como un aumento en la productividad. La usabilidad permite mayor rapidez en la realización de tareas y reduce posibles pérdidas de tiempo a los usuarios.

Existen diferentes metodologías para para alcanzar y mejorar la usabilidad en una aplicación o un sistema. En la siguiente sección se explica la metodología diseño centrado en el usuario, la cual fue la metodología de trabajo de la investigación.

4.1 Diseño Centrado en el Usuario

El diseño centrado en el usuario (DCU, en inglés *User Centered Design* (UCD)) es definido por la Usability Professionals Association² (UPA) como un enfoque de diseño cuyo proceso está dirigido por información sobre las personas que van a usar el producto; es decir, situar al usuario en el centro del diseño de un dispositivo o aplicación.

Usabilidad no es lo mismo que DCU. La usabilidad es un atributo de la calidad del diseño final del producto, mientras que el DCU es el camino para alcanzar y mejorar su usabilidad; o sea, la usabilidad es el *qué*, mientras que el DCU es el *cómo*. Diseñar algo usable es bueno pero no significa necesariamente que se haya conseguido aplicando DCU (Hassan-Montero & Ortega-Santamaría, 2009).

El concepto de DCU se utilizó como marco de trabajo, investigación y desarrollo de principios del diseño de interfaces de usuario. Era el momento de observar cómo la gente usaba los sistemas y creaba sus propios modelos mentales a partir de los procesos de interacción. Tres fueron los términos que debían ser valorados para entender estos procesos (Norman, 1983):

- **Modelo conceptual.** Ofrecido por el diseñador del sistema.
- **Interfaz.** La imagen que el sistema presenta al usuario.
- **Modelo mental.** Desarrollado por el usuario a partir de la imagen.

Como señala Norman, la imagen del sistema guía al usuario en la adquisición y construcción de un modelo mental ajustado al modelo conceptual creado por el diseñador (Norman, 1983). De este modo, el enfoque del DCU persigue asegurar la consecución de un producto con la funcionalidad adecuada para usuarios concretos. El objetivo de esta filosofía es ofrecer respuesta a preguntas como: quién usará el sistema, qué va a hacer con él y qué información necesita para alcanzar sus objetivos.

² Sitio web de Usability Professionals Association: <http://www.upassoc.org/>.

El DCU es un proceso cíclico en el que las decisiones de diseño están dirigidas por y hacia el usuario y los objetivos que pretende satisfacer el producto, y donde la usabilidad del diseño es evaluada de forma iterativa y mejorada incrementalmente (ver Figura 13) (International Organization for Standardization, 1999).

De acuerdo con la norma ISO 13407, este proceso consta de cuatro fases (ver Figura 13) (International Organization for Standardization, 1999):

- **Entender y especificar el contexto de uso.** En esta fase se identifican a los usuarios a las que se dirige el producto, para qué lo usarán y en qué condiciones.
- **Especificar requisitos.** En esta fase se identifican los objetivos del usuario y del proveedor del producto deberán satisfacer, es decir, los propósitos, las características o los requerimientos que se deben satisfacer.
- **Producir soluciones de diseño.** Esta fase se puede subdividir en diferentes etapas secuenciales, desde las primeras soluciones conceptuales hasta la solución final de diseño:
 - **Arquitectura de información.** La actividad y el resultado de organizar, clasificar, ordenar, estructurar y describir los contenidos de una aplicación, con el fin de que sus usuarios puedan satisfacer sus necesidades informativas con el menor esfuerzo posible. Para referirse al grado en que una arquitectura de información satisface a sus usuarios, suele emplearse el término encontrabilidad (*findability*). La arquitectura de información es el esqueleto de la aplicación.
 - **Diseño conceptual.** Las posibles opciones que se deben tomar en cuenta para el diseño final de la aplicación, identificando los objetivos y los requerimientos de la aplicación. Este diseño está compuesto por la creación, la exploración y la representación de una idea.
 - **Diseño de interacción.** La actividad y el resultado de definir el comportamiento interactivo de la aplicación; es decir, qué acciones se ofrecerán al usuario en cada momento, y cómo responderá la aplicación a las acciones que realice. Es el sistema nervioso de la aplicación.

- **Diseño gráfico.** Aspecto gráfico o visual, esta dimensión gráfica del producto será la que, desde el primer momento de la interacción, condicione su usabilidad. La comunicación visual no solo está supeditada a la facilidad con la que el usuario pueda localizar visualmente el contenido de su interés, o sea capaz de discriminar visualmente el sentido y significado de los elementos de la página. Un diseño gráfico usable se convierte en una forma más de lograr la expresividad del producto y de potenciar, entre otras cosas, la identidad visual como parte del proceso de comunicación.
- **Evaluación.** Esta es la fase más importante del proceso. En ella se validan las soluciones de diseño (el sistema satisface los requisitos) o se detectan problemas de usabilidad, normalmente a través de test con usuarios.

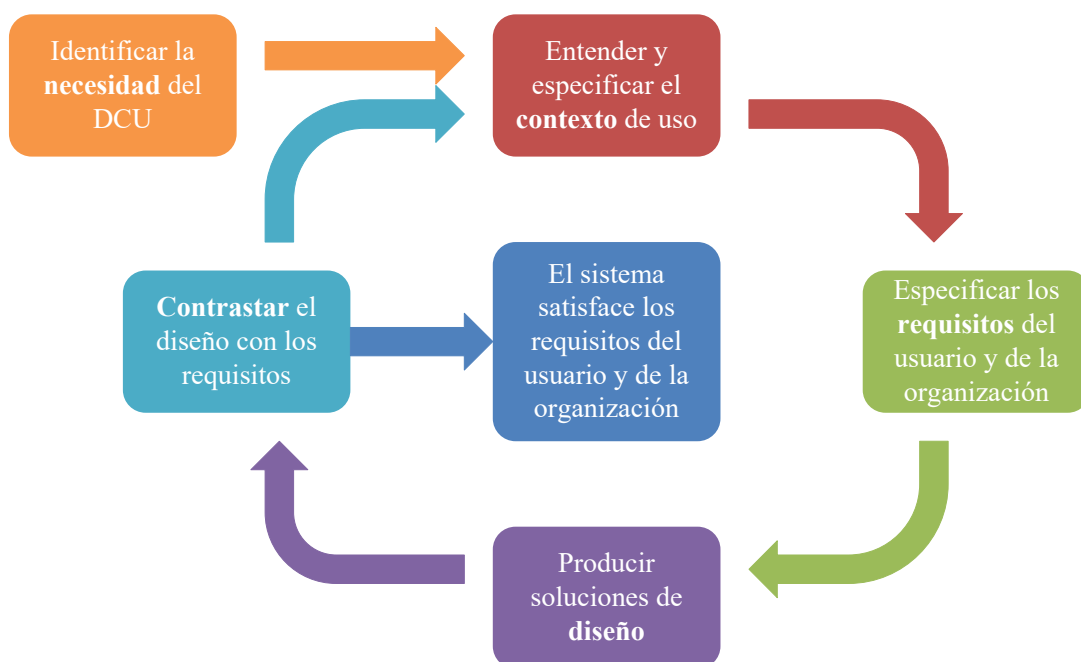


Figura 13. Proceso del DCU (International Organization for Standardization, 1999).

La norma ISO 13407 ha sido revisada por la norma ISO 9241-210, en donde se describen los seis principios clave que caracterizan al DCU (International Organization for Standardization, 2010):

1. El diseño está basado en una comprensión explícita de usuarios, tareas y entornos.

2. Los usuarios están involucrados durante el diseño y el desarrollo.
3. El diseño está dirigido y refinado por evaluaciones centradas en usuarios.
4. El proceso es iterativo.
5. El diseño está dirigido a toda la experiencia del usuario.
6. El equipo de diseño incluye habilidades y perspectivas multidisciplinares.

Los principales beneficios que se obtienen empleando los conceptos y técnicas de usabilidad tomando en cuenta el DCU al desarrollar una aplicación o un sistema se pueden resumir en³:

1. Incremento en la productividad del usuario.
2. Incremento en la satisfacción del usuario final.
3. Reducción de los costes en desarrollo y mantenimiento de las aplicaciones.
4. Disminución de los costes de capacitación y apoyo a los usuarios.

A continuación, se muestran diferentes metodologías y técnicas que se usan para evaluar la usabilidad de una aplicación o un sistema.

4.2 Evaluación con Expertos

Las evaluaciones con expertos son análisis detallados y sistemáticos de la interfaz del producto, realizados por expertos, teniendo en cuenta unos principios de usabilidad reconocidos (Brian Shackel, 1991). Los especialistas realizan una evaluación aplicando una lista de criterios y sus conocimientos sobre la experiencia de los usuarios y las mejores prácticas.

Las metas de las evaluaciones realizadas con expertos son bastante similares a las realizadas con usuarios, ya que ambas se focalizan en encontrar y priorizar problemas de usabilidad, evaluando diseños en el contexto de ciertas tareas. La diferencia radica en dónde se pone el foco de atención. En los análisis y evaluaciones con expertos se hace incidencia en la

³ Sitio web de Usability.gov: <http://www.usability.gov/basics/usasaves/index.html>.

identificación de problemas en lo que se refiere a pautas de diseño y consistencia. En cambio, en las pruebas de usabilidad se encuentran problemas relacionados con áreas especiales y flujos de tarea, así como la obtención de retroalimentación, es decir, la respuesta por parte de los usuarios al usar el sistema o aplicación (Jakob Nielsen, 1994b; Brian Shackel, 1991).

A menudo, en los análisis realizados con expertos se escapan problemas de usabilidad debido a su grado de experiencia, los cuales emergen durante las pruebas de usabilidad, o bien, encuentran problemas que los usuarios no detectan (falsas alarmas o problemas inexistentes) (Jakob Nielsen, 1994b).

A pesar de ello, las evaluaciones con expertos en usabilidad son utilizadas porque son mucho más baratas y rápidas de realizar que las pruebas de usabilidad. Normalmente, solo conlleva unos pocos días inspeccionar un producto o aplicación y escribir un informe. De esta manera no es necesario contratar laboratorios ni reclutar usuarios, minimizando los gastos. Debido a esto, las evaluaciones con expertos son realizadas más frecuentemente sobre todo cuando el presupuesto o el tiempo son limitados. Además, suelen ser útiles para poder descartar problemas graves de usabilidad, relacionados con estándares o incluso de funcionalidad.

El proceso a seguir en una evaluación es el siguiente⁴:

1. Estudiar la audiencia objetivo y las tareas típicas.
2. Navegar de forma exploratoria por las páginas de la interfaz.
3. Examinar la interfaz de las páginas, siguiendo una lista de criterios establecidos.
4. Agrupar la lista de problemas de usabilidad y asignar la gravedad según su importancia y frecuencia.
5. Analizar los resultados y buscar soluciones.

Existen diferentes tipos de pruebas para realizar en la evaluación con expertos. Algunas de ellas son las siguientes:

⁴ Sitio web de Usolab: Análisis y test de usabilidad: <http://www.usolab.com/analisis-usabilidad/>.

- **Evaluación heurística.** Técnica en la que los especialistas en usabilidad juzgan la conformidad de la interfaz de usuario con ciertos principios de usabilidad universalmente conocidos. Una evaluación heurística conlleva que un grupo de expertos inspeccionan la interfaz y evalúan el grado de cumplimiento de cada uno de esos principios, denominadas comúnmente heurísticas (J. Nielsen & Molich, 1990; J. Nielsen, 1994c).
- **Recorridos cognitivos.** El especialista que realiza la sesión recorre un escenario de tareas determinado como habría de hacerlo un usuario (Barnum, 2001), estos recorridos derivan de los análisis cognitivos.
- **Inspección Formal de Usabilidad.** Es una metodología de inspección de código. Es la que se lleva a la práctica con más frecuencia después de las evaluaciones heurísticas (J. Nielsen, 1994c). Ello se debe a que contempla con mayor amplitud los conceptos más técnicos.
- **Revisión de guías de estilo.** Los expertos revisan que el sitio cumpla con las directrices fijadas por una institución o plataforma o entorno particular (Rubin & Chisnell, 2008).
- **Evaluación cooperativa**⁵. Técnica mucho más participativa por parte del usuario. Los dos métodos definidos en esta evaluación son:
 - **Métodos de Diario.** Se requiere que los usuarios registren las actividades que desarrollan en su entorno de trabajo durante un día normal.
 - **Modelado por Empatía.** Se trata de un método desarrollado para usuarios con discapacidades. En este caso el diseñador o desarrollador trata de ponerse en la situación del usuario, simulando su discapacidad.

En la evaluación con expertos, se usan una serie de heurísticas creadas por Nielsen para evaluar el diseño de la interfaz de usuario (Nielsen, 1994a, 2005):

1. **Visibilidad del estado del sistema.** El sistema debe informar a los usuarios del estado del sistema, dando una retroalimentación apropiada en un tiempo razonable.

⁵ Sitio web de European Usability Support Center: Co-operative evaluation:
<http://www.ucc.ie/hfrg/projects/respect/urmethods/coop.htm>.

2. **Usar el lenguaje de los usuarios.** El sistema debe usar el lenguaje de los usuarios, con palabras o frases que le sean conocidas, en lugar de los términos que se usan en el sistema, para que al usuario no se le dificulte usar el sistema.
3. **Control y libertad para el usuario.** En casos en los que los usuarios elijan una opción del sistema por error, este debe contar con las opciones de deshacer y rehacer para proveer al usuario de una salida fácil sin tener que usar diálogo extendido.
4. **Consistencia y estándares.** El usuario debe seguir las normas y convenciones de la plataforma sobre la que está implementando el sistema, para que no se tenga que preguntar el significado de las palabras, situaciones o acciones del sistema.
5. **Prevención de errores.** Es más importante prevenir la aparición de errores que generar buenos mensajes de error. Hay que eliminar acciones predispuestas al error o, en todo caso, localizarlas y preguntar al usuario si está seguro de realizarlas.
6. **Minimizar la carga de la memoria del usuario.** El sistema debe minimizar la información que el usuario debe recordar mostrándola a través de objetos, acciones u opciones. El usuario no tiene por qué recordar la información que recibió anteriormente. Las instrucciones para el uso del sistema deberían ser visibles o estar al alcance del usuario cuando se requieran.
7. **Flexibilidad y eficiencia de uso.** Los aceleradores permiten aumentar la velocidad de interacción para el usuario experto tal que el sistema pueda atraer a usuarios principiantes y experimentados. Es importante que el sistema permita personalizar acciones frecuentes para así acelerar el uso de este.
8. **Diálogos estéticos y diseño minimalista.** La interfaz no debe contener información que no sea relevante o se utilice raramente, pues cada unidad adicional de información en un diálogo compite con las unidades relevantes de la información y disminuye su visibilidad relativa.
9. **Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores.** Los mensajes de error deben expresarse en un lenguaje claro, indicar exactamente el problema y ser constructivos.

10. **Ayuda y documentación.** Aunque es mejor un sistema que no necesite documentación, a veces puede ser necesario disponer de esta. Si es así, la documentación tiene que ser fácil de encontrar, estar centrada en las tareas del usuario, tener información de las etapas a realizar y no ser extensa.

A continuación, se describe sobre la metodología de test de usabilidad.

4.3 Test de Usabilidad

Los tests o pruebas de usabilidad son una forma habitual de estimar la facilidad con la que una persona puede usar un objeto o una aplicación, como por ejemplo una página web, una interfaz de usuario o un dispositivo. Un test de usabilidad es el proceso que permite aprender de los usuarios cuán usable es un producto observándolos mientras lo utilizan, anotando los problemas de uso con los que se encuentran para poder solucionarlos posteriormente (Barnum, 2001).

Dichas pruebas consisten primordialmente en realizar una experimentación, generalmente en un laboratorio, formalizando una serie de tareas que deben realizar los usuarios mientras son monitoreados. Se selecciona a un grupo de usuarios, que pueden o no ser usuarios habituales de dicha aplicación, dependiendo de los objetivos de la prueba, y se les solicita que lleven a cabo las tareas más representativas para las cuales fue diseñada (Rubin & Chisnell, 2008). El equipo de diseño, desarrollo y otros involucrados toman nota de la interacción, particularmente en lo que respecta a los errores y dificultades con las que se encuentran los usuarios. En la mayoría de casos se aplica sobre aplicaciones no finalizadas, pudiendo tratarse de un prototipo.

Existen ciertas variantes en los tests de usabilidad:

- **Tests de usabilidad (pruebas con usuarios).** Pruebas llevadas a cabo con los usuarios finales del sistema.

- **Tests remotos.** Forma de llevar a cabo los tests de usabilidad sin la necesidad de que el evaluador se encuentre en el mismo espacio físico que el participante (Rubin & Chisnell, 2008).
- **Tests mediante técnicas de *eyetracking*.** El seguimiento del movimiento de los ojos (*eye tracking*) es el proceso de evaluar el punto donde se fija la mirada (dónde se encuentra mirando el usuario) o el movimiento del ojo en relación con la cabeza, dependiendo del equipo utilizado (J. Nielsen & Pernice, 2010a). Este proceso es usado en campos como la investigación de sistemas visuales, la psicología, la lingüística cognitiva y el diseño de nuevos productos.

Los tests remotos y el uso de las técnicas de *eyetracking* pueden considerarse una especialización de los tests de usabilidad (Rubin & Chisnell, 2008).

Todas estas pruebas no son excluyentes entre sí. En ocasiones se usan varias pruebas con iguales o diferentes perfiles de usuario (tanto de usuarios novatos como expertos) a lo largo del ciclo de vida del producto, para realizar un estudio de la usabilidad en las diversas etapas del desarrollo. Se explica con más detalle las pruebas con usuarios, porque es el método usado en esta investigación.

La prueba con usuarios es la principal herramienta del DCU, ya que representa la mejor forma de evaluar la usabilidad de un diseño y es una prueba destinada a medir tanto la usabilidad objetiva (qué y cómo actúa el usuario), como la usabilidad subjetiva (cómo de fácil ha percibido la tarea).

Las características de todo test de usabilidad (también conocido como pruebas con usuarios) son (Rubin & Chisnell, 2008):

- El objetivo principal es mejorar la usabilidad de un producto.
- Los participantes son una representación de los usuarios reales.
- Los participantes realizan tareas reales.
- Se observa y se registra lo que los participantes hacen y dicen.

- Se analizan los datos, se diagnostican problemas reales y se recomiendan cambios para resolver dichos problemas.

Según Kuniavsky, algunos requisitos que deben cumplir las tareas encomendadas al participante durante las pruebas son (Kuniavsky, 2003):

- **Estar descritas en términos de objetivos finales.** La tarea debe contextualizarse bajo un objetivo o motivación mayor.
- **Ser razonables.** Las tareas típicas que un usuario real llevaría a cabo.
- **Ser específicas.** La tarea no puede ser demasiado genérica, sino que debe describir objetivos concretos, con el fin de poder comparar los problemas encontrados con los del resto de participantes.
- **Ser factibles.** Encomendar al usuario tareas irrealizables no aporta información útil sobre los problemas reales de usabilidad de la aplicación. En estas pruebas lo que se debe evaluar es el diseño a través de los usuarios, no al contrario.
- **Duración razonable (adecuada).** Si la tarea requiere demasiado tiempo para ser completada, sería recomendable descomponerla en subtareas.

Las principales métricas que se tienen en cuenta a la hora de realizar un test de usabilidad son (Jakob Nielsen, 1993, 2003):

- **Facilidad de Aprendizaje (*Learnability*).** Facilidad con la que nuevos usuarios desarrollan una interacción efectiva con el sistema o producto. Cuando el usuario no ha utilizado dicha aplicación, mediante una serie de tareas se puede observar si la interacción con esta es sencilla, intuitiva y fácil de realizar, es decir, si es fácil aprender a usarla. Está relacionada con la predictibilidad, la capacidad de síntesis, la familiaridad, la generalización de los conocimientos previos y la consistencia.
- **Eficiencia (*Efficiency*).** Una vez que los usuarios han aprendido el funcionamiento básico del diseño, se contabiliza: (1) el número de errores cometidos por los usuarios durante la prueba y si dichos errores fueron recuperables, es decir, si el usuario, puede continuar la prueba y realizar el procedimiento adecuado y (2) el tiempo empleado por los usuarios en la realización de las tareas.

- **Cualidad de ser recordado (*Memorability*).** Cuando se trata de usuarios que son habituales o que han trabajado con una aplicación, el recuerdo sería la memoria en el uso de dicha aplicación, es decir, qué tanto recuerda el usuario después de un lapso sin usar la aplicación.
- **Efectividad (*Errors*).** Esta métrica se valora en función de los éxitos y fracasos en la consecución de las tareas. Esta medida está relacionada con el recuerdo y el aprendizaje.
- **Satisfacción (*Satisfaction*).** Esta métrica está compuesta por medidas subjetivas, provenientes de comentarios, opiniones, o bien a través de cuestionarios. Nos da la respuesta emocional del usuario de tal manera que permite conocer cómo se siente el usuario al terminar la tarea o la prueba. En algunos casos es el propio usuario el que lo transmite (por ejemplo: si se ha sentido cómodo en el uso de la aplicación, si le ha resultado fácil realizar la tarea, si le gusta la aplicación, si la encuentra útil, entre otros) comentándolo en voz alta mientras realiza la prueba o bien, mediante cuestionarios post-tarea/post-test. En otras ocasiones, son las personas que realizan la prueba (los mediadores o los observadores presentes, o bien mediante la grabación de dicha prueba) los que denotaran el comportamiento del usuario.

El número de usuarios depende de cómo se formalice la prueba y cuáles son los perfiles a tener en cuenta pero, según Hwang y Salvendy, empleando entre 8 y 12 usuarios es suficiente para poder detectar el 80% de los problemas de usabilidad (Hwang & Salvendy, 2010). Otros autores, como Nielsen y Landauer, opinan que con 5 usuarios es suficiente (Jakob Nielsen & Landauer, 1993). En general, la comunidad internacional acepta que para pruebas informales es suficiente con 5 usuarios⁶. En cambio, para pruebas más rigurosos se requiere más usuarios, dependiendo de los casos y los objetivos de la prueba.

Normalmente, es común que las pruebas con usuarios se realicen en laboratorios equipados donde se dispone de al menos dos salas o habitaciones: una es la sala de observación (*control room*), para los observadores o evaluadores, y la otra es la sala de pruebas (*test room*), donde

⁶ Jakob Nielsen's Alertbox - Usability Testing with 5 Users: <http://www.useit.com/alertbox/20000319.html>.

los usuarios llevan a cabo la prueba. Ambas salas suelen estar separadas por un cristal con visibilidad en un único sentido, de esta manera se permite que se pueda a ver a los usuarios desde la sala de observación, sin que los usuarios puedan ver a través de cristal, haciendo que se sientan más cómodos durante la realización de la prueba.

El equipamiento básico para la sala de test implica todo el material necesario para reproducir el entorno de trabajo en el que el usuario tendrá que usar el sistema o aplicación. Normalmente, esto conlleva una computadora u otro dispositivo homólogo, preparado con el sistema operativo y el software necesario para realizar la prueba: la aplicación a testear y, en el caso de que se disponga, el software que realice el monitoreo. Además, también se debe contar con una cámara (puede ser la propia cámara web del dispositivo, o en su defecto una cámara externa) y un micrófono para grabar los comentarios del usuario.

El equipamiento básico para la sala de observación será una computadora con el programa específico para registrar y monitorear la interacción del usuario, que posteriormente también permitirá analizar la información recogida. Se suele usar también una mesa de control para manejar los dispositivos externos, como cámaras y micrófonos.

La metodología que se debe seguir en una prueba con usuarios es la siguiente (Jakob Nielsen, 1993):

1. Preparación del test con usuarios.
2. Montaje del ambiente de trabajo (laboratorio de usabilidad).
3. Realización del test con usuarios.
4. Exploración de los datos del test y extracción de conclusiones.
5. Generación del informe.

El primer punto de la metodología es importante, por lo que la planificación para la realización de una prueba de usuarios debe realizarse de la siguiente manera (Jakob Nielsen, 1993):

1. Establecer el equipo de trabajo.

2. Definir las características del producto y la audiencia.
3. Establecer los objetivos y las unidades de medida para la prueba.
4. Definir el perfil de usuario.
5. Reclutar los usuarios necesarios para la realización de la prueba.
6. Seleccionar las tareas que se incluirán en la prueba.
7. Determinar cómo categorizar los resultados.
8. Escribir el plan de la prueba.
9. Confeccionar la documentación necesaria: carta de bienvenida, documento de conformidad para grabar al usuario, entre otros.

Todas estas tareas deben ser discutidas ampliamente por todo el equipo. La mayoría de ellas pueden trabajarse en paralelo, no hay necesidad de seguir un proceso secuencial. Una vez establecido el conjunto de miembros del equipo, debe decidirse el rol que desarrollará cada uno de ellos. Los tres roles fundamentales a desempeñar son (Jakob Nielsen, 1993):

- **Mediador.** Es la persona que interacciona directamente con los participantes, por lo que debe ser el miembro del equipo más afín al trato con los usuarios. La principal tarea del mediador consiste en la introducción del participante al test y a cada una de las tareas que se van a testear. También suele entrevistarse con el participante al finalizar el test. Este papel es quizás el más difícil de desempeñar, puesto que en la medida de lo posible se deben evitar sesgos durante la interacción que el mediador tenga con los diferentes usuarios.
- **Observador.** Si el equipo usa algún tipo de software para capturar y registrar las observaciones, así como los tiempos de cada tarea mediante una computadora, entonces el observador debe tener conocimiento de dicho software, además del que se está testeando para poder entender lo que sucede, y poder marcar los momentos en que el usuario tiene dificultades. Si no existe posibilidad de usar herramientas informáticas, se debe determinar el conjunto de personas necesarias para poder tomar notas, controlar el tiempo por tarea, entre otros.

- **Personal de servicio técnico.** Es la persona responsable de la puesta en marcha de todos los equipos del laboratorio (tanto la sala de pruebas como la sala de control). Debe establecer la posición de las cámaras y asegurarse que registran correctamente.

Una vez realizada la prueba, todo lo que se haya observado, grabado, monitoreado y anotado debe ser resumido y sintetizado en un informe final. El informe debe incluir los problemas de usabilidad que se han encontrado en el producto y algunas indicaciones o sugerencias para solucionarlos (Jakob Nielsen, 1993).

En esta sección se han discutido metodologías y técnicas para evaluar la usabilidad de una aplicación o un sistema para dispositivos no móviles. A continuación, se detallan las metodologías y las técnicas para evaluar la usabilidad de una aplicación móvil.

4.4 Usabilidad Móvil

En los últimos años, los esfuerzos por hallar metodologías apropiadas para evaluar la usabilidad móvil se han incrementado enormemente (Jesper Kjeldskov & Graham, 2003). Son muchos los grupos de investigación que han intentado encontrar solución a las diversas problemáticas y retos asociados a los dispositivos móviles.

A pesar de ello, aunque existe un esfuerzo de investigación considerable en el campo de la usabilidad, en términos generales, debido a que la irrupción de la tecnología móvil es todavía reciente, relativamente pocos estudios en usabilidad móvil han sido llevados a cabo hasta el momento (Dutson, 2014; J. Kjeldskov et al., 2005).

A continuación, se mencionan las principales dificultades al realizar una evaluación de usabilidad en dispositivos móviles.

4.4.1 Dificultades en la Evaluación de Usabilidad Móvil

Aunque los métodos de evaluación de usabilidad existentes en el campo de la IHC tienen una amplia trayectoria y pueden también aplicarse en este tipo de entornos, no son especialmente adecuados a los entornos móviles puesto que no contemplan sus particularidades propias. Es por ello que las técnicas y métodos existentes deben ser especializados (formulación de nuevos principios heurísticos acordes a las peculiaridades de las interfaces móviles, adecuación de cuestionarios a este tipo de consideraciones, análisis de la conveniencia de los tests de campo, de tal forma que el factor de movilidad sea una característica intrínseca de las pruebas, entre otros), con objeto de que el contexto de uso pase a ser un parámetro destacado en la evaluación. Por otro lado, debe dedicarse especial atención al software y hardware que les dé soporte y tests de usabilidad específicos para entornos móviles.

Existen varias razones que hacen que la evaluación de usabilidad móvil, y en especial el registro de datos cuantitativos, sea un reto, tal y como explica Nielsen⁷. Las técnicas de usabilidad móvil deben abordar tres dificultades clave (Jakob Nielsen & Budiu, 2012):

1. El limitado tamaño de pantalla de los dispositivos móviles.
2. La falta de herramientas de software específicas.
3. Las dificultades adicionales derivadas del uso de un contexto móvil.

En primer lugar, las pequeñas pantallas de estos dispositivos dificultan la interacción del usuario, en relación con una pantalla de escritorio tradicional, de diversas formas. Especialmente notable es el hecho de que las limitadas capacidades tecnológicas de estos dispositivos restringen el proceso de recogida de datos cualitativos y cuantitativos durante los tests de usabilidad. Esta restricción es especialmente importante en la recogida de datos precisos relativos a la mirada y movimiento de los ojos del usuario, por medio de técnicas de *eyetracking*. Esto es debido a que el ojo apenas se mueve cuando la mirada de la persona se

⁷ Sitio web de Alertbox de Nielsen, Mobile Usability First Findings: <http://www.useit.com/alertbox/mobile-usability-study-1.html>.

desplaza entre elementos relativamente próximos, como ocurre en las pantallas de los dispositivos móviles (Cheng, 2011).

En segundo lugar, se carece de software exhaustivo y completamente desarrollado para el estudio de la usabilidad, específicamente orientado a dispositivos móviles. Finalmente, la característica inherente de movilidad asociada a los dispositivos móviles es un problema, ya que el usuario y el teléfono están en constante movimiento y, por tanto, el entorno es cambiante. Por otra parte, las tareas del usuario están sujetas a las interrupciones, por ejemplo: la pérdida de la cobertura, llamadas entrantes o distracciones.

A continuación, se explican las metodologías discutidas en las secciones anteriores, pero en relación con usabilidad móvil.

4.4.2 Evaluación con Expertos

La evaluación con expertos, como se comentó anteriormente, es una de las principales herramientas utilizadas en la evaluación de usabilidad, debido a su facilidad para realizarla teniendo en cuenta tanto costos temporales como costos de recursos humanos.

En el caso de los dispositivos móviles, es también una de las metodologías más populares por los mismos motivos. Los expertos o especialistas realizan una evaluación aplicando una lista de criterios y sus conocimientos sobre la experiencia de los usuarios, pero en este caso, dichos criterios son definidos especialmente teniendo en cuenta las características propias de este tipo de dispositivos. Es por ello que en algunas evaluaciones se incluye el cambio de escenario y de contexto de uso para que el evaluador pueda realizar una evaluación más precisa (Jakob Nielsen & Budiu, 2012).

No obstante, existen distintos conjuntos de heurísticos definidos en la literatura, los cuales difieren según la tipología del dispositivo (por ejemplo: si es un teléfono inteligente o una tableta) e incluso, según la plataforma (sistema operativo), de manera que no hay una lista de heurísticos universalmente reconocida y consensuada. Son muchos los fabricantes de

dispositivos y proveedores que no permiten la publicación de aplicaciones que cumplan un mínimo de criterios en usabilidad, como es el caso de Apple y Nokia, que ofrecen guías de estilo para los desarrolladores. Además, como paso previo a su publicación, los expertos de estas marcas realizan diferentes evaluaciones heurísticas y otras evaluaciones similares, para poder evaluar la usabilidad de las aplicaciones y solo en caso de superarlas, son publicadas en las tiendas en línea.

Todas estas guías y criterios tienden más a confundir que a clarificar, puesto que resulta difícil establecer el conjunto de heurísticos a considerar, de tal forma que implica un mayor esfuerzo para los desarrolladores.

Las evaluaciones más típicas que se realizan son las heurísticas, los recorridos cognitivos y la revisión de guías de estilo. Cabe destacar también algunas variaciones de la evaluación heurística, como el recorrido heurístico, ideado para evaluar dispositivos móviles incorporando escenarios de uso a la aplicación práctica de la evaluación heurística, o bien el recorrido contextual, para conducir un recorrido heurístico mediante el trabajo de campo (Po, Howard, Vetere, & Skov, 2004).

A continuación, se describe la metodología de test de usabilidad aplicada a dispositivos móviles.

4.4.3 Test de Usabilidad

Como se ha comentado anteriormente, un test de usabilidad es el proceso que permite aprender de los usuarios cuán usable es un producto o una aplicación, observando o monitoreando a los usuarios mientras lo utilizan. En estos tests se intenta recabar la máxima información de la interacción del usuario con la aplicación y el dispositivo, observando sus reacciones.

En los tests de usabilidad para dispositivos móviles se siguen sus pesquisas y métricas que en los tests de usabilidad convencionales, pero con las dificultades inherentes derivadas de

sus particularidades: sus limitaciones físicas, la pantalla pequeña con la que interactúan el factor de movilidad, entre otros (Jakob Nielsen & Budiu, 2012). A pesar de dichas limitaciones, se intenta recabar el máximo de información. Ello dependerá, principalmente, de la prueba de usuario que se vaya a realizar.

Antes de proceder a medir la usabilidad de las aplicaciones móviles, para poder verificar si son o no usables por medio de esta metodología, es necesario tener presente la siguiente información (ver el diagrama de la Figura 14) (Jakob Nielsen & Budiu, 2012):

- **Objetivos.** Descripción de los objetivos esperados con las pruebas de usuarios.
- **Contexto de uso.** Descripción de los componentes de contexto, como son: usuarios, tareas a realizar, equipo necesario y entorno.
- **Medidas de usabilidad.** La efectividad, la eficiencia y la satisfacción para el contexto de uso concreto.

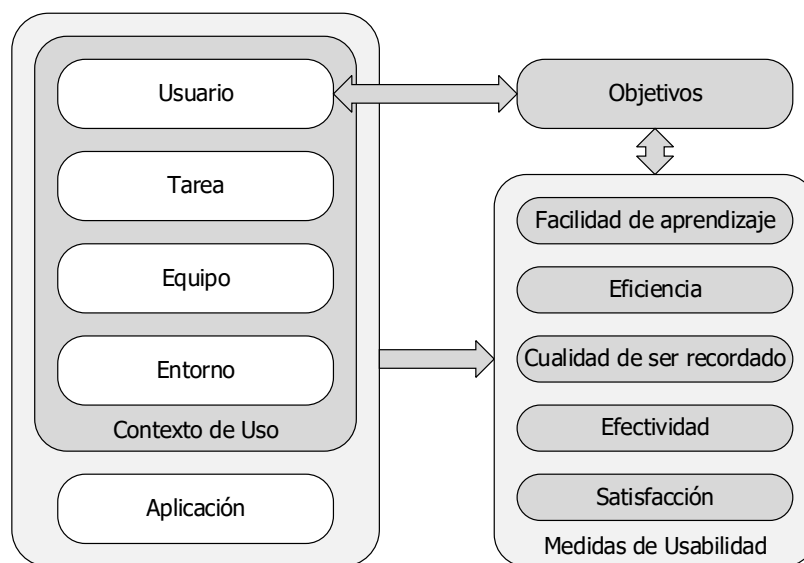


Figura 14. Componentes a definir en un test de usabilidad (Jakob Nielsen & Budiu, 2012).

A continuación, se explican las diferentes adaptaciones de los tests de usabilidad según la técnica que se vaya a realizar y, teniendo en cuenta el factor de movilidad: dónde se deben realizar.

4.4.4 Tests en Laboratorio versus Tests de Campo

La característica de movilidad, propia de los dispositivos móviles, afecta completamente a la interacción del usuario con el dispositivo. A su vez, aumenta enormemente la variabilidad de contextos de uso en que se puede interactuar con un dispositivo móvil. Por tanto, afecta a la medición de la usabilidad de un dispositivo y sus aplicaciones. Por estos motivos, una de las discusiones más comunes en la investigación de la usabilidad móvil es la necesidad de realizar las experimentaciones en laboratorio o en diferentes escenarios y contextos de uso reales, más allá de las técnicas y métodos de evaluación que se realizan comúnmente en un ambiente controlado.

Esta controversia es un aspecto que ha motivado a diferentes grupos de investigación a llevar a cabo estudios comparativos de tests de usabilidad en laboratorio y en campo (Duh, Tan, & Chen, 2006; Kallio & Kaikkonen, 2005; J. Kjeldskov et al., 2005; C. M. Nielsen, Overgaard, Pedersen, Stage, & Stenild, 2006). Las conclusiones de dichos trabajos apuntan a que en la mayoría de casos se obtienen datos más relevantes cuando los tests se realizan en el laboratorio, en relación con los tests de campo, llevados a cabo en contextos reales.

Los tests de campo están sujetos a múltiples interrupciones y distracciones por parte del usuario, conllevan a un consumo mucho mayor de tiempo efectivo para su evaluación, por lo que no llega a compensar todo el esfuerzo que supone su puesta en escena. Cabe remarcar que la dificultad de organizar y realizar un test de campo es mayor debido a que conlleva un costo elevado tanto de recursos humanos como temporal (puesto que hay más necesidad de técnicos, observadores y mediador durante la realización de la prueba para recabar datos y ayudar al usuario en caso de que sea necesario), puesto que la organización y la realización del test implica también más planificación y está sujeto a más interrupciones (Duh et al., 2006; J. Kjeldskov et al., 2005). Así pues, los tests de campo conllevan a un consumo mucho mayor de tiempo efectivo para su evaluación, por lo que no llega a compensar todo el esfuerzo que supone su puesta en escena.

Por otro lado, la mayoría de estas investigaciones llegan a la conclusión de que en la mayoría de casos en los tests de campo se obtienen únicamente datos cualitativos. Otra de las conclusiones obtenidas es que este tipo de estudios se limitan a la detección de los problemas de usabilidad más evidentes (Duh et al., 2006; J. Kjeldskov et al., 2005), así como aquellos problemas directamente relacionados con el dispositivo utilizado, por ejemplo: falta de visibilidad de la pantalla debido a la exposición a la luz natural o la dificultad de poder manejar con una sola mano el dispositivo. Este tipo de problemas, más relacionadas con el diseño y ergonomía del propio dispositivo, en la mayoría de casos, no pueden ser detectados mediante un test controlado convencional (Duh et al., 2006). A pesar de ello, en la mayoría de estudios se comprueba la carencia de datos cualitativos sólidos en las experimentaciones.

Por el contrario, los test realizados en ambientes controlados o laboratorios, ofrecen la posibilidad de obtener datos tanto cualitativos como cuantitativos. Por otra parte, no permiten evaluar la usabilidad en un contexto real de uso del dispositivo.

Debido a estas premisas, se aconseja posponer los tests de campo para las fases finales del desarrollo, y siempre habiendo realizado previamente, como mínimo, un test en laboratorio en las fases de desarrollo anteriores o antes de los tests de campo, para poder descartar el máximo número de problemas y haber podido obtener datos cuantitativos sólidos sobre la aplicación o el dispositivo (C. M. Nielsen et al., 2006).

Cabe destacar que aunque se pierde completamente el factor de movilidad, se puede tener en cuenta que se pueden ofrecer diferentes contextos de uso, como ya se han hecho en diferentes estudios dentro de laboratorio (J. Kjeldskov et al., 2005).

En resumen, los estudios de usabilidad en dispositivos y en aplicaciones móviles realizados hasta el momento, así como la exploración de los distintos métodos de evaluación se centran en averiguar cómo identificar mejor los problemas de usabilidad. El propósito principal es dotar adecuadamente de usabilidad a las aplicaciones móviles, teniendo en cuenta las dificultades que ello conlleva, sobre todo en lo que se refiere a la obtención de datos tanto cualitativos como cuantitativos relevantes. Para ello, en la mayoría de casos, se ha optado

por la adaptación de las metodologías existentes de evaluación de usabilidad, tratando de optimizar los problemas que presentan estas técnicas en dichos dispositivos.

En el capítulo siguiente, se describe la base metodológica que se llevó a cabo para el desarrollo de este trabajo y el logro de los objetivos de la investigación.

CAPÍTULO 5. METODOLOGÍA

A diferencia de las herramientas de programación orientadas a los niños de edad preescolar discutidas en la Sección 1.1 y el Capítulo 2, se plantea realizar una herramienta que apoye el aprendizaje colaborativo de la programación en niños entre 4 y 6 años con la presencia de mediación docente.

En este capítulo se explica la metodología realizada para contestar cada una de esas preguntas de investigación, logrando cada objetivo específico respectivo, y con ello comprobar el objetivo general de esta investigación. En la Sección 5.2 se explica la metodología llevada a cabo para lograr los tres primeros objetivos específicos y en la Sección 5.3 se explica la metodología llevada a cabo para lograr el cuarto objetivo específico.

Cabe destacar que este trabajo de investigación contó con la colaboración de los expertos de la Unidad de Aprendizaje Lógico, Científico y Robótica de la FOD, quienes coordinan y llevan a cabo, junto con el MEP, el Programa Nacional de Informática Educativa MEP-FOD⁸ (PRONIE MEP-FOD), una alianza que durante 29 años ha apoyado a la educación costarricense. También se cuenta con el apoyo del Programa de Tecnologías Educativas Avanzadas⁹ (PROTEA) de la Universidad de Costa Rica.

Se entrevistó a los expertos de la FOD para realizar la especificación de requerimientos y tareas específicas de la herramienta. Además, los expertos de la FOD fueron los responsables en la realización de las evaluaciones con los niños. Los expertos de PROTEA brindaron el contacto con docentes y estudiantes de la carrera de Educación Preescolar de la Escuela de Formación Docente de la Universidad de Costa Rica, contando así con la experticia necesaria en la investigación sobre el área de Educación Preescolar.

⁸ La FOD ha desarrollado esta área de experticia desde su creación en 1987 y la ha llevado a la práctica desde el Programa Nacional de Informática Educativa que se ejecuta en coordinación con el Ministerio de Educación Pública (PRONIE MEP-FOD).

⁹ Programa de Tecnologías Educativas Avanzadas (PROTEA), promueve la apropiación pedagógica de los recursos que ofrecen las tecnologías digitales de la información y la comunicación (TIC), en la formación de docentes y estudiantes.

En las siguientes secciones se describen las etapas de la metodología de trabajo, con los pasos a seguir para conseguir el objetivo planteado en esta investigación.

5.1 Fases de la Investigación

La metodología de la investigación se diseñó para crear y evaluar la herramienta colaborativa de programación en dos ejes diferentes:

- Creación y evaluación de usabilidad de la herramienta colaborativa de programación orientada a niños entre 4 y 6 años, y con ello lograr los tres primeros objetivos específicos.
- Evaluación del aporte de la herramienta colaborativa de programación orientada a niños entre 4 y 6 años en el proceso de colaboración de la primera infancia, y con ello lograr el cuarto y último objetivo específico.

Con el fin de alcanzar los objetivos específicos planteados en esta investigación, se utilizó el proceso de diseño centrado en el usuario (DCU) del estándar ISO 13407 como metodología de trabajo, siguiendo las fases de este proceso y los principios clave que lo caracterizan (comentados en la Sección 4.1). En la Figura 15 se muestra el proceso de DCU con las principales técnicas utilizadas en cada fase del DCU, y en la Tabla 8 se describen dichas técnicas. Las técnicas utilizadas en la investigación fueron tomadas del estándar ISO TR-18529 (International Organization for Standardization, 2000).

En las secciones siguientes se describe en detalle cada una de las fases y los tipos de evaluaciones que se realizaron en la investigación, de acuerdo con los ejes planteados.

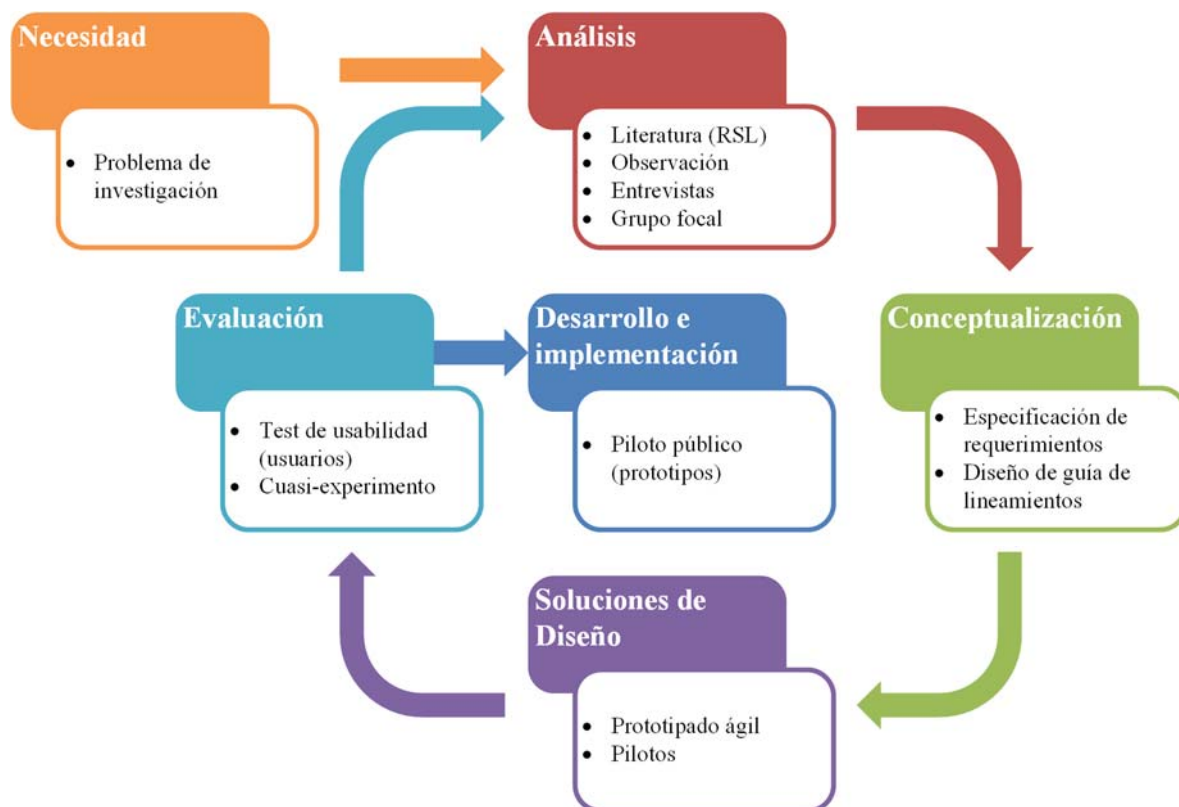


Figura 15. Fases y técnicas del DCU utilizadas.

5.2 Usabilidad de TITIBOTS Colab en la Primera Infancia

Siguiendo la metodología de trabajo del DCU, en primer lugar, fue importante entender y especificar el contexto de uso de la herramienta colaborativa de programación, donde se identificaron las características y las limitaciones de los usuarios finales, el propósito de crear la herramienta, y bajo qué condiciones se debe crear. Esto se logró mediante un estudio sistemático de literatura y entrevistas a docentes de preescolar y expertos de la FOD. Además, se realizaron visitas de campo a grupos de preescolar y a la FOD, con el fin de observar la interacción y las actividades de aprendizaje de los grupos de preescolar, así como los cursos de robótica y programación que la FOD ofrece para niños entre 6 y 8 años, registrando la interacción de los niños en relación al proceso de enseñanza-aprendizaje de la programación y la robótica.

Tabla 8. Descripción de las técnicas utilizadas en cada fase del DCU (Norman, 1983).

Fase del Proceso	Técnicas Utilizadas
Análisis	<ul style="list-style-type: none"> • Literatura. Revisión de literatura para entender y especificar el contexto de uso, donde se debe identificar las características y las limitaciones de los usuarios finales, el propósito de crear la herramienta, y bajo qué condiciones se debe crear. • Método de observación. Implica que un investigador vea usuarios trabajando en el campo de estudio, y tomar notas sobre la actividad que se lleva a cabo. La observación puede ser directa, investigador presente durante la tarea, o indirecta, donde la tarea es vista por otros medios: grabaciones. El método es útil para el estudio de las tareas y los procesos actualmente ejecutados. • Entrevistas. Preguntas realizadas directamente a expertos y usuarios finales para conocer de primera mano información que ayude a mejorar el diseño, así se conocen las opiniones, las motivaciones y las experiencias de los usuarios. • Grupo focal. Reunión con un conjunto de expertos y consultores cuyas opiniones son solicitada acerca de un tema específico. La meta es obtener las percepciones y las ideas de los participantes sobre las soluciones de diseño, llegar a un consenso.
Conceptualización	<ul style="list-style-type: none"> • Especificación de requerimientos. Descripción de características, requisitos y restricciones de la herramienta y su contexto de uso, que se prueban en el proceso de desarrollo. • Diseño de guía de lineamientos. Los lineamientos se utilizan para proporcionar una apariencia uniforme y agradable al usuario de la herramienta. Forma parte de los requisitos de usabilidad y su diseño debe ser monitoreado durante el desarrollo.
Prototipado	<ul style="list-style-type: none"> • Prototipado ágil. Validación preliminar con el cliente y las diferentes instituciones implicadas en los contenidos, de forma que se vaya consensuando el prototipo, junto con las recomendaciones de los consultores. Propuestas de interfaz de la aplicación, para ser evaluadas y proceder a su aceptación, mejora o rechazo • Pilotos. Realización de pruebas con los usuarios finales usando prototipos funcionales para su evaluación. Se realizan varios pilotos en distintos momentos del diseño para ir mejorándolo, usando pocos usuarios (se recomienda 5 usuarios).
Evaluación	<ul style="list-style-type: none"> • Test de usabilidad. Pruebas que se basan en la observación de cómo un grupo de usuarios llevan a cabo una serie de tareas encomendadas por el evaluador, analizando los problemas de usabilidad con los que se encuentran. • Cuasi-experimento. Diseño de investigación experimental en el cual los sujetos o grupos de sujetos de estudio no están asignados aleatoriamente, se trabaja con grupos naturales. El diseño cuasi-experimental es una forma de investigación experimental utilizado ampliamente en las ciencias sociales y la psicología.
Desarrollo e implementación	<ul style="list-style-type: none"> • Piloto público. Prueba de un prototipo funcional final con los usuarios finales en un entorno real.

La primera fase de la investigación se centró en desarrollar una herramienta de programación (individual, sin colaboración) fácil e intuitiva de usar para niños en edad preescolar, y establecer una comunicación efectiva entre el dispositivo móvil y el robot. En esta fase se especificaron los requisitos de la herramienta de programación, identificando los objetivos del usuario y de los expertos a satisfacer. Esto se realizó mediante entrevistas a los expertos de la FOD y a los docentes de preescolar para definir las tareas específicas a llevar a cabo con la herramienta de programación y los requerimientos.

Paralelamente, debido a que la investigación pretende independizarse de las computadoras de escritorio y usar dispositivos móviles, se realizó una revisión bibliográfica sobre los sistemas operativos móviles y dispositivos móviles existentes del mercado, y se creó una tabla comparativa de ventajas y desventajas de estos de acuerdo con sus características. El estudio se realizó con al menos 3 sistemas operativos y 5 dispositivos móviles. De sus resultados se escogió el sistema operativo móvil y el dispositivo móvil más adecuado a usar para desarrollar la herramienta de programación.

Luego, se estudiaron diversos diseños de interfaces orientadas a niños entre 4 y 6 años. También se tomó en cuenta la literatura, las visitas de campo y los criterios de los expertos entrevistados. Además, se especificaron las heurísticas de usabilidad para aplicaciones móviles. A partir de esto se comenzó a extraer características de la interfaz de una herramienta de programación adecuada para los niños de preescolar.

Posteriormente, se diseñaron los elementos de las interfaces de la herramienta basados en las heurísticas de usabilidad, las características del dispositivo seleccionado, la especificación de requerimientos, las tareas especificadas y las características extraídas del estudio de las interfaces de las aplicaciones orientadas a niños entre 4 y 6 años en dispositivos móviles.

También se definió y diseñó el protocolo de comunicación para realizar la conectividad entre el dispositivo móvil y el robot. Como parte de este proceso se analizaron varias opciones de tipos de robots y kits robóticos para ser utilizados con la herramienta colaborativa de programación, según la especificación de requerimientos, las tareas especificadas y los elementos colaborativos. Luego, según el análisis, se seleccionaron algunos tipos de robots y kits robóticos. Al estar trabajando conjuntamente con la FOD, uno de los kits robóticos que se utilizó fue el LEGO Mindstorms NXT.

En esta etapa se realizaron soluciones de diseño acordes a la herramienta. Como resultado se debió especificar la arquitectura de información, el diseño conceptual, el diseño de interacción y el diseño gráfico. Estos diseños se validaron con expertos de la FOD, docentes de preescolar y niños entre 4 y 6 años, hasta obtener una propuesta final.

Con las soluciones de diseño se llevó a cabo la validación de cada una. Las primeras soluciones de diseño fueron validadas por expertos de la FOD y las docentes de preescolar. Cuando se tuvo una solución final validada por los expertos, se llevó a cabo una validación con niños entre 4 y 6 años, cuyos resultados de esta validación fueron analizados por los expertos de la FOD, las docentes de preescolar y la investigadora para obtener una propuesta final de diseño.

En este proceso de análisis de los resultados de la validación, se comenzó a generar una guía de lineamientos a tomar en cuenta al diseñar y construir una herramienta de aprendizaje orientada a niños con edades entre 4 y 6 años, con los comentarios y las recomendaciones de los expertos y las docentes de preescolar. Posteriormente, se realizó una búsqueda de estudios que proporcionaran lineamientos para el diseño de herramientas de aprendizaje orientadas a niños de preescolar. Como resultado, se agregaron a la guía los lineamientos pertinentes a la investigación obtenidos de los estudios de varios autores: Chau, Chaudron, Díaz, Harari, Amadeo, Leavitt, Shneiderman, Nielsen, Pane, Resnick y Sesame Workshop.

Debido a que la investigación es de carácter académico, se procuró aprovechar al máximo las posibilidades que ofrecen las herramientas gratuitas disponibles a través de Internet para llevar a cabo la implementación del prototipo de la herramienta de programación.

En esta etapa se usaron las siguientes técnicas:

- Prototipado ágil de soluciones de diseño de la herramienta de programación, donde se realizó con las personas implicadas (expertos de la FOD) un consenso de la herramienta, tomando en cuenta las recomendaciones de los consultores (docentes de preescolar).
- Pilotos de la herramienta de programación con niños entre 4 y 6 años, donde se evaluó su usabilidad y su funcionalidad, capturando las métricas respectivas comentadas en la Sección 4.3.1.

Con los resultados obtenidos de las técnicas anteriores, se procedió a la implementación de una versión final de la herramienta, con el consenso de las personas implicadas, las

recomendaciones de los consultores y los cambios resultantes de las evaluaciones de los pilotos.

Cuando se tuvo un diseño final y un prototipo implementado, se comprobó que la herramienta cumpliera con los requerimientos especificados por los expertos, y que fuera fácil e intuitiva de usar por los niños. Esta fase de evaluación fue la más importante del proceso del DCU, ya se validan las soluciones de diseño (el sistema satisface los requisitos) o, por el contrario, se detectan problemas de usabilidad.

En las etapas de diseño e implementación se crearon documentos que describen la funcionalidad y los requisitos especificados, y se diseñó un plan de pruebas que cubriera todos los casos de uso de la aplicación, la simulación de entornos de pruebas, la ejecución de los casos de prueba y la gestión de los defectos encontrados. Dichos documentos fueron utilizados en esta etapa de evaluación por los distintos actores involucrados.

La evaluación de usabilidad se llevó a cabo siguiendo la técnica “test de usabilidad” usando el método de pruebas con usuarios comentado en la Sección 4.3.1 (Jakob Nielsen, 1993).

La evaluación de funcionalidad estuvo contemplada en la evaluación de usabilidad, ya que en estas pruebas el usuario debe ejecutar las tareas especificadas en los requisitos de la herramienta (casos de uso), garantizando que hace lo que debe (casos positivos), que no falla (casos negativos) y que cumple con lo que se ha solicitado (satisface los requisitos). Cada usuario realizó las acciones indicadas en cada caso de prueba, comprobando que se cumpliera el resultado esperado. Si el resultado era distinto al esperado, se reportaba el defecto con sus detalles: descripción, datos utilizados y capturas de pantalla, entre otros, para facilitar su solución.

Aunque las pruebas con usuarios son pruebas de evaluación, no se deben dejar para el final del proceso de diseño e implementación de la aplicación, ya que el DCU es una filosofía de diseño iterativa, basada en la mejora incremental del producto. Por lo que, en esta investigación se realizaron varias evaluaciones durante el desarrollo de la herramienta.

Cada evaluación realizada se contextualizó bajo un objetivo o motivación. Además, como se mencionó en la Sección 4.3.1, cada evaluación fue razonable, específica, factible y tuvo una duración adecuada (al trabajar con niños de estas edades, se debe trabajar en la planificación de diversas actividades en lapsos de 90 minutos como máximo).

Los expertos de FOD, los docentes de preescolar y los niños usaron y evaluaron los diferentes prototipos de la herramienta para probar y validar lo que se especificó con ellos, buscando problemas de usabilidad y funcionalidad que fueron corregidos. Además, todas las pruebas piloto fueron evaluadas con niños.

Para cada uno de estas pruebas piloto se confeccionaron guías didácticas de trabajo, guías de observación y guías de evaluación, para los miembros del equipo de trabajo. Además, cada piloto fue grabado y contó con un grupo de mediadores y observadores que tomaron notas de todo el proceso.

En general, los primeros pilotos que ejecutaron fueron pruebas informales de la herramienta sin colaboración, para detectar posibles errores de usabilidad y funcionalidad, y evaluar que la herramienta realmente apoyara el aprendizaje de la programación (que no interfiera negativamente en el proceso) mediante la definición de variables como cantidad de intentos y tiempo promedio para completar con éxito un desafío, el promedio de errores cometidos y el promedio de errores que se recuperan. Cada prueba fue realizada con al menos 5 niños, tal como recomienda Nielsen (Jakob Nielsen, 1993), y en sesiones de aproximadamente 2 horas.

Posteriormente, se trabajó en la fase 2, donde se identificaron mecanismos para apoyar la colaboración usando la herramienta, identificando e implementando actividades del proceso de aprendizaje colaborativo que podrían ser automatizadas, y planteando estrategias didácticas colaborativas para niños de edad preescolar (tomando en cuenta las actividades del proceso de aprendizaje colaborativo). Estas estrategias se pusieron en práctica en los dos últimos experimentos que se llevaron a cabo con niños entre 4 y 6 años, contando con la ayuda de expertos de la FOD y docentes de preescolar para la realización, la revisión y la validación de dichas estrategias didácticas.

Con la identificación de las actividades del aprendizaje colaborativo que se pueden automatizar, se procedió a diseñar e implementar la herramienta colaborativa de programación. Al mismo tiempo, se definió y diseñó el protocolo de comunicación para realizar la conectividad entre los dispositivos móviles y el robot, para apoyar el proceso del aprendizaje colaborativo.

Como resultado de este proceso, se agregaron a la guía de lineamientos, los factores que se deben tomar en cuenta al diseñar y construir una herramienta colaborativa de programación orientada a niños con edades entre 4 y 6 años.

Una vez desarrollada la herramienta colaborativa de programación, se realizaron los dos últimos pilotos para evaluar la usabilidad de la herramienta colaborativa. Estos pilotos se realizaron en un contexto real, llevando a cabo una prueba más rigurosa al usar más usuarios (aproximadamente 15), y realizando talleres de programación con sesiones de aproximadamente 2 horas.

Las métricas que se usaron para llevar a cabo la evaluación de usabilidad de cada piloto realizado son las que establece Nielsen (1993). Estas métricas se usaron de la siguiente manera:

- **Facilidad de aprendizaje.** Cuando un niño utilizó por primera vez la herramienta, se observó si la interacción con esta fue sencilla, intuitiva y fácil de realizar, es decir, si fue fácil aprender a usarla.
- **Eficiencia.** Cuando los niños aprendieron el funcionamiento básico de la herramienta, se contabilizó: (1) el número de errores cometidos por los niños al realizar las tareas durante la prueba y si dichos errores fueron recuperables, es decir, si el niño pudo continuar la prueba y realizar el procedimiento adecuado; y (2) el tiempo empleado por los niños para hacer las tareas.
- **Cualidad de ser recordado.** Cuando los niños usaron varias veces la herramienta, se recurrió al recuerdo (la memoria en el uso de dicha aplicación), es decir, qué tanto recuerda de la herramienta el niño después de un período sin usarla. Se realizó

entrevistando a los niños, preguntándoles aspectos relacionados con la herramienta y pidiéndoles que dibujaran lo que recuerden o lo que más les gustó.

- **Efectividad.** Se midió en función de los éxitos y fracasos en la ejecución de las tareas asignadas durante la prueba: cuántas tareas logró realizar cada niño o cada equipo colaborativo.
- **Satisfacción.** Medidas subjetivas, provenientes de comentarios, opiniones, o bien a través de encuestas a los niños, obteniendo respuestas emocionales de ellos, de tal manera que permiten conocer cómo se sienten al terminar la tarea o la prueba (es decir, si le gustó la herramienta, si le ha resultado fácil usar la herramienta, si le fue fácil realizar las tareas, entre otros). Se realizó esta evaluación escuchando lo que comentan en voz alta mientras realizan la prueba, mediante encuestas pos-prueba, las observaciones del personal que realiza la prueba (los mediadores o los observadores presentes) y la grabación de la actividad.

Las tareas están definidas como desafíos (retos) de programación que se les da a los niños para que los solucionen mediante las herramientas de programación propuestas en esta investigación, tanto la individual como la colaborativa. Durante cada taller se realizaron desafíos guiados por las docentes, desafíos que se resolvían entre todos y desafíos que se dejaban para resolver por cada niño o grupo (la docente podría guiar). La solución de estos desafíos se colocó en la pizarra para que todos los niños pudieran observar la solución y colocarla en sus pizarras y luego pasarla en la herramienta de programación. En el taller del último piloto con la herramienta individual, y los talleres de Brasil y Costa Rica con la herramienta colaborativa, los últimos dos días eran de evaluación sobre los aprendizajes logrados y se realizaban desafíos donde cada niño o grupo debía solucionarlos sin ayuda o guía. En relación con las métricas de usabilidad, se toman en cuenta todos los desafíos realizados durante el taller, y los desafíos de evaluación se toman en cuenta para obtener datos sobre el aprendizaje de la programación de robots en la primera infancia.

Durante cada prueba se registró toda la información relevante (análisis de métricas, encuestas, observaciones, evaluaciones) para analizar el comportamiento y la usabilidad de

la herramienta. En esta parte de la evaluación, se usaron datos cuantitativos y cualitativos para poder responder la tercera pregunta de investigación.

Para los datos cuantitativos se usaron los siguientes métodos:

- **Análisis de métricas.** La herramienta obtiene y guarda de forma automática en una base de datos un conjunto de métricas que se definieron de acuerdo con los objetivos planteados, por ejemplo: tiempo promedio para realizar una tarea, cantidad promedio de comandos utilizados para realizarla y cantidad de errores, entre otros. Estas métricas fueron usadas para evaluar algunas de las pruebas de usabilidad.
- **Evaluaciones.** En las últimas dos sesiones de cada piloto se establecieron desafíos para que los equipos colaborativos los resolvieran, y se contabilizó la cantidad de desafíos realizados y el tiempo promedio de solución por cada equipo. Esto se hizo para evaluar algunas de las métricas de usabilidad.

Para los datos cualitativos se usaron los siguientes métodos:

- **Observaciones.** La investigadora, los mediadores y los observadores estuvieron atentos a todo lo que sucedió en cada piloto, registrando toda la información, y generando informes diarios. Además, se usaron las grabaciones para realizar un análisis más minucioso. Las observaciones sirvieron para evaluar métricas de usabilidad.
- **Entrevistas.** La investigadora diseñó dos tipos de entrevistas, una dirigida a los padres de familia para obtener información del perfil tecnológico de los niños, y otra dirigida a las mediadoras (docentes) para evaluar su opinión de la herramienta.
- **Cuestionarios.** Las mediadoras (docentes) realizaron dos cuestionarios a los niños sobre aspectos relacionados con la satisfacción al usar herramienta y cuánto recordaban los niños de ella. Los cuestionarios sirvieron para evaluar algunas de las métricas de usabilidad. Las preguntas de opción única se realizaron con una escala de 2 opciones, ya que de Leeuw recomienda que para niños entre 7 y 10 años la escala viable a usar es de dos o tres opciones, debido a sus características (de Leeuw, 2011)

(ver Apéndice 6 y Apéndice 9). Se decidió, que para niños de la primera infancia (entre 4 y 6 años) es más viable usar solamente dos opciones.

Una vez que los participantes finalizaron cada prueba y fue registrada toda la información pertinente, se procedió a analizar los resultados y sintetizarlos en un informe final, concluyendo qué mejoras necesita el diseño y la aplicación.

En las secciones siguientes, se describe cada una de las fases y los tipos de evaluaciones que se realizaron en la investigación para evaluar la usabilidad de la herramienta de programación TITIBOTS.

5.2.1 Fase 1. Evaluación de TITIBOTS

La primera fase de la investigación se centró en el desarrollo de una aplicación fácil e intuitiva de usar, que permitiera a los niños de preescolar programar robots (de forma individual). Esto con el fin de que aprendieran conceptos básicos de programación y robótica, para promover el desarrollo del pensamiento computacional desde edades tempranas.

Esta versión de la herramienta no incluía automatización de ninguna de las actividades del proceso del aprendizaje colaborativo. Era una versión para ser utilizada de forma individual: cada niño tenía una tableta y un robot asignados.

En la Figura 16 se muestra el proceso de evaluación llevado a cabo en esta fase, el cual consistió de cuatro etapas.

En la primera etapa se llevó a cabo una actividad de esbozado grupal¹⁰ (*group sketching*) (Greenberg & Bohnet, 1991) para diseñar una versión preliminar de TITIBOTS. La primera

¹⁰ Método ágil, rápido y económico para el desarrollo y explicación de las ideas de forma simultánea. Se utiliza durante las sesiones de co-diseño con el fin de compartir los puntos de vista en el interior del equipo: ofrece una base común para la discusión de los participantes. Se basa en dibujos básicos y sencillos con el fin de fomentar la participación de todos en el diseño de una versión preliminar. Sitio web de Service Design Tools: <http://www.servicedesigntools.org>.

etapa del proceso de evaluación fue una validación de concepto con expertos y docentes de preescolar.

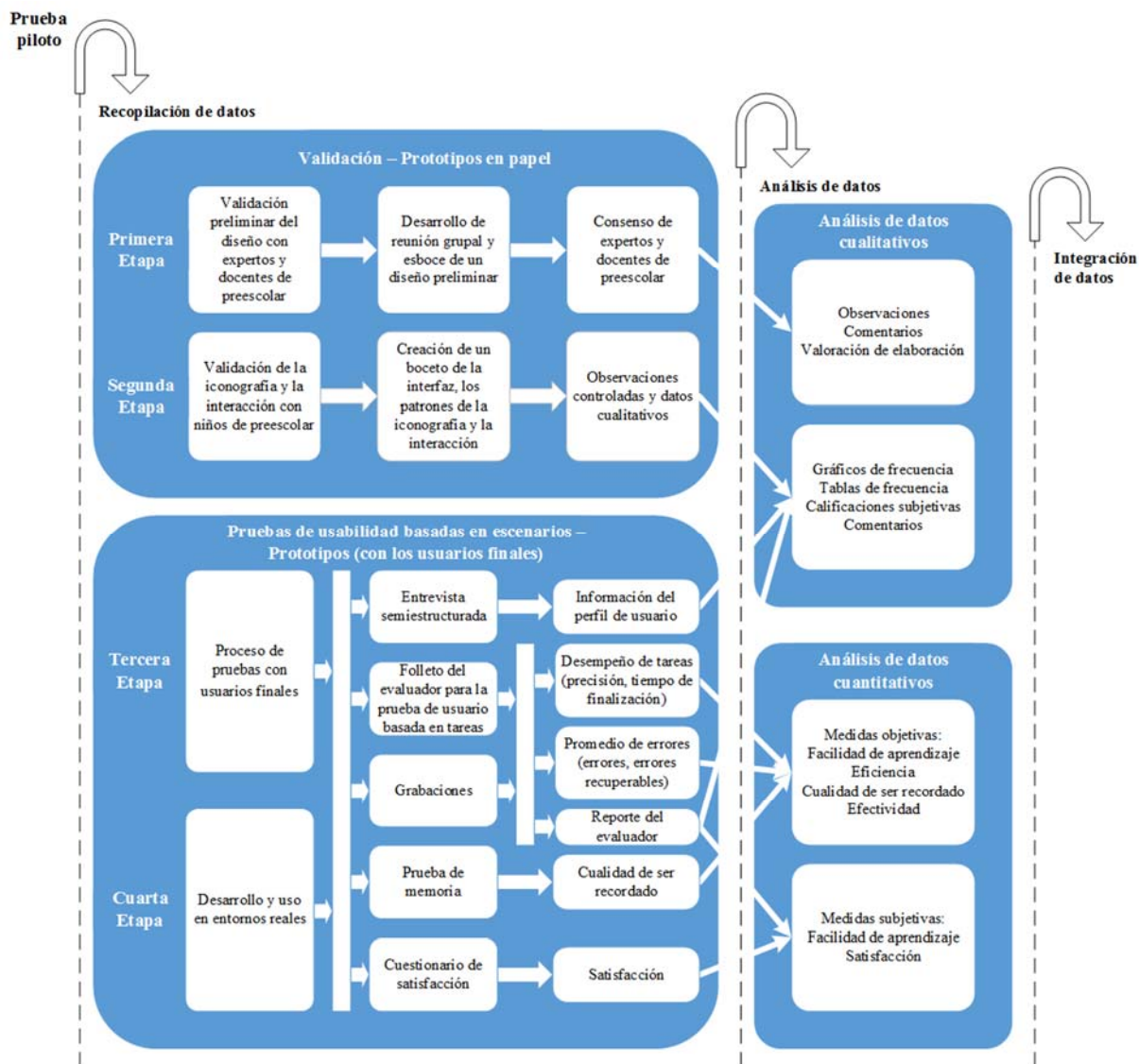


Figura 16. Método de evaluación de TITIBOTS.

La segunda etapa fue la validación de la iconografía y la interacción con los niños, donde se usaron las mejores propuestas del conceso de la primera etapa (ver Apéndice 13 para más detalles). En esta etapa se creó un esquema que representa los patrones de la interfaz, la iconografía y la interacción de la herramienta de programación individual, con el objetivo de validarlos. El fin de esta validación fue generar una guía de lineamientos de diseño y

construcción de herramientas de aprendizaje de la programación orientadas a niños entre 4 y 6 años y así, obtener una propuesta final de diseño y desarrollar el primer prototipo funcional.

En la tercera etapa, se evaluó la usabilidad y la funcionalidad de la herramienta de programación con niños de preescolar (ver Apéndice 14 para más detalles). Para ello, se crearon una serie de desafíos para los niños y se realizaron observaciones para evaluar su comportamiento. El objetivo principal de esta evaluación fue observar la reacción de los niños con la herramienta y, encontrar problemas de usabilidad y funcionalidad.

Finalmente, el prototipo se evaluó en un escenario real, por lo que se organizó un taller en la FOD con niños de preescolar (ver Apéndice 15 para más detalles). El objetivo principal de esta evaluación fue usar la herramienta de programación para introducir conceptos básicos de programación y robótica, y observar si los niños pueden aprender a programar robots mediante el uso de dicha herramienta.

Las dos últimas etapas se realizaron con niños entre 4 y 5 años de edad, porque se quería probar el prototipo con los usuarios finales más jóvenes. En estas etapas se siguió la metodología propuesta por Nielsen para llevar a cabo un test de usabilidad con usuarios (Jakob Nielsen, 1993).

La Tabla 9 muestra las métricas asociadas con los atributos de usabilidad y los criterios de aceptación para cada atributo (Granic & Cukusic, 2011; Jakob Nielsen, 1993). Para recoger los datos cuantitativos y cualitativos, se usaron los siguientes instrumentos de medición (Granic & Cukusic, 2011; Jakob Nielsen, 1993):

- **Entrevista semiestructurada.** Obtención de la información del perfil de usuario.
- **Folleto del evaluador.** Folleto donde el investigador realiza la evaluación, toma notas, describe los problemas identificados y completa la información sobre el promedio de tareas completadas y el tiempo promedio empleado.
- **Grabaciones.** Grabaciones de audio y video de los pilotos.

- **Prueba de memoria.** Cuestionario para medir el número de funciones y características de la aplicación que son recordadas correctamente por los niños.
- **Cuestionario de satisfacción.** Cuestionario para medir la satisfacción de los niños con la aplicación.

Tabla 9. Atributos de usabilidad, métricas y metas asociadas.

Atributo de Usabilidad	Métricas	Meta de Usabilidad
Facilidad de aprendizaje	<ul style="list-style-type: none"> • Tiempo promedio empleado para completar un desafío por primera vez. • Tiempo promedio de entrenamiento. 	<ul style="list-style-type: none"> • El tiempo promedio para completar un desafío por los niños debe estar entre 10 y 30 minutos. • El tiempo promedio de entrenamiento debe estar entre 30 y 60 minutos.
Eficiencia	<ul style="list-style-type: none"> • Total y porcentaje de desafíos exitosos. • Tiempo promedio para completar un desafío. 	<ul style="list-style-type: none"> • El porcentaje de desafíos exitosos completados debe ser mayor al 70%. • El tiempo promedio para completar un desafío por los niños debe estar entre 10 y 20 minutos.
Cualidad de ser recordado	<ul style="list-style-type: none"> • Total y porcentaje de respuestas correctas sobre la aplicación. 	<ul style="list-style-type: none"> • Los usuarios principiantes deben recordar al menos el 50% de las funcionalidades de la aplicación, mientras que los usuarios experimentados deben memorizar más del 80%.
Efectividad	<ul style="list-style-type: none"> • Total y promedio de errores. • Total y promedio de errores recuperados (error = intento fallido al llevar a cabo un desafío) 	<ul style="list-style-type: none"> • El promedio de errores debe estar entre 5 y 10 errores, si es menor a 5 mejor. • El promedio de errores recuperados por el usuario debe ser mayor al 60% del total de errores.
Satisfacción	<ul style="list-style-type: none"> • Gusta o no. • Nivel de dificultad. 	<ul style="list-style-type: none"> • El porcentaje de satisfacción, usualmente, debe ser mayor al 65%.

En las secciones siguientes, se describe en detalle cada una de las fases y los tipos de evaluaciones que se realizaron en la investigación para evaluar la usabilidad de la herramienta colaborativa de programación TITIBOTS Colab.

5.2.2 Fase 2. Evaluación de TITIBOTS Colab

En la primera fase se creó una herramienta de programación orientada a niños entre 4 y 6 años, la cual es fácil e intuitiva de usar, y con el diseño de la interfaz, la iconografía y la interacción validados.

A partir de este punto, se comenzó la segunda fase de la investigación, la cual se centró en identificar y automatizar actividades del aprendizaje colaborativo para agregarlas a la herramienta, con el fin de que aprendieran conceptos básicos de programación y robótica de forma colaborativa, con el fin de promover el desarrollo del pensamiento computacional y el proceso de colaboración desde edades tempranas.

El desarrollo de esta herramienta colaborativa se realizó durante la pasantía realizada en el RExLab de la Universidade Federal de Santa Catarina (UFSC), Campus Araranguá, Santa Catarina, Brasil, bajo la coordinación del Dr. Juarez Bento da Silva.

En esta versión de la herramienta se incluyó la automatización de varias actividades del proceso del aprendizaje colaborativo. Es una versión para ser utilizada de forma colaborativa en el contexto de un salón de clases, donde la docente es la mediadora del proceso. Al ser una herramienta colaborativa, se forman grupos de dos o tres niños, donde cada niño tiene una tableta, pero con una parte de la totalidad de los comandos. Los comandos se reparten de forma automática por la herramienta entre los miembros del grupo. Con ello se pretende incentivar el proceso de colaboración mediante la distribución de los recursos. Además, se tiene solo un robot para todos los grupos colaborativos que se forman. En la Figura 17 se muestra el proceso de evaluación llevado a cabo en esta fase, el cual consistió de dos etapas.

En la primera etapa, proceso de prueba con usuarios finales, se evaluó la usabilidad y la funcionalidad de la herramienta colaborativa de programación con los niños de preescolar de la Escuela de Educación Básica Municipal Otávio Manoel Anastácio, en Araranguá – Santa Catarina – Brasil (ver Apéndice 16 para más detalles). Para ello, se crearon una serie de actividades lúdicas y desafíos para los niños, y se realizaron observaciones para evaluar su comportamiento. El objetivo principal de esta evaluación fue encontrar problemas de usabilidad y funcionalidad. Además, se quiso observar el comportamiento de los niños al trabajar de forma colaborativa en las actividades y los desafíos, para ver si con el uso de la herramienta se incentiva la colaboración.

Finalmente, el prototipo TITIBOTS Colab se evaluó en un entorno real: la Escuela Santa Mónica, en Montelimar, Guadalupe, Costa Rica, con el grupo de kínder de la Institución (ver Apéndice 17 para más detalles). Para ello, se crearon una serie de actividades lúdicas y desafíos para los niños, y se realizaron observaciones para evaluar su comportamiento. El objetivo principal de esta evaluación fue valorar el aporte de la herramienta en el proceso de colaboración en niños de preescolar. Se observó el comportamiento de los niños al trabajar de forma colaborativa en las actividades y los desafíos, para ver si con el uso de la herramienta se incentiva la colaboración.

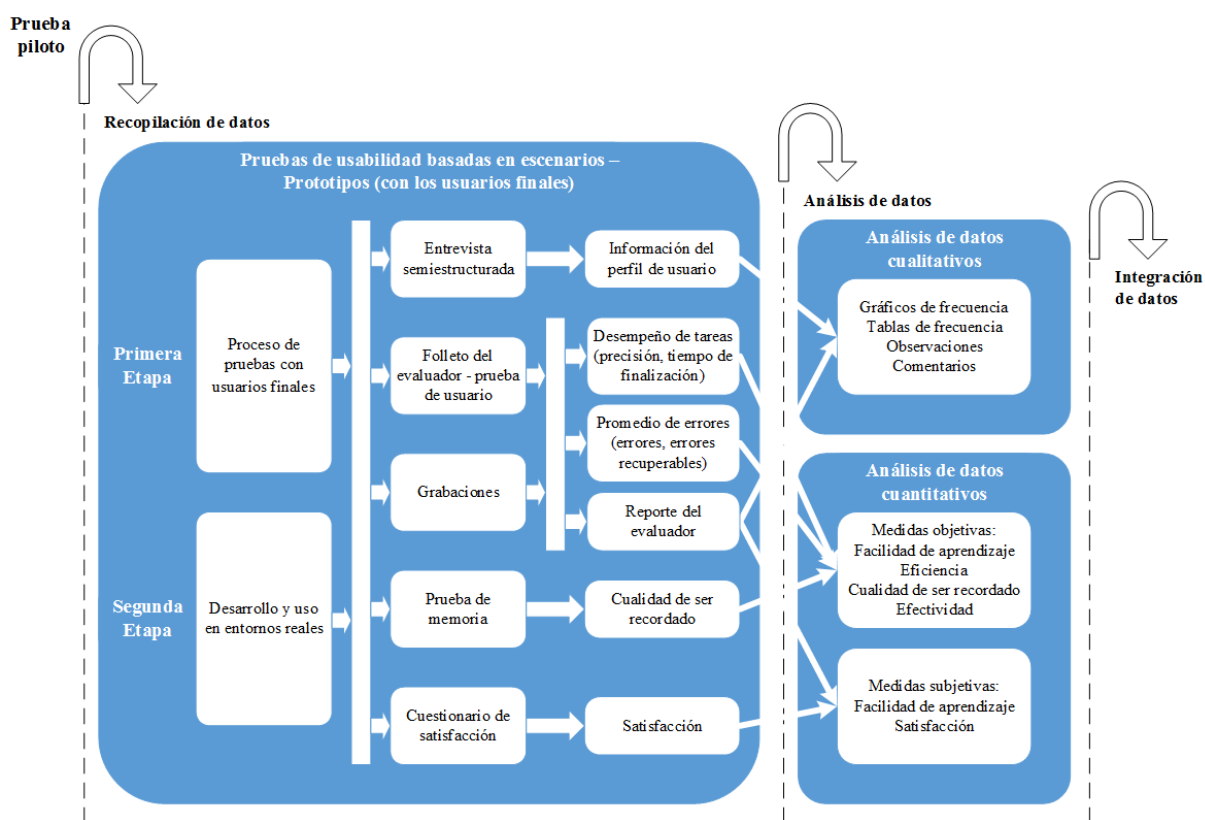


Figura 17. Método de evaluación de TITIBOTS Colab.

En ambas etapas se siguió la metodología propuesta por Nielsen para llevar a cabo un test de usabilidad con usuarios (Jakob Nielsen, 1993). En la Tabla 9, Sección 5.2.1, se mostraron las métricas asociadas con los atributos de usabilidad y los criterios de aceptación para cada atributo, y se explicaron los instrumentos de medición para recoger los datos cuantitativos y cualitativos.

En la siguiente sección se describe la metodología llevada a cabo para evaluar el aporte de TITIBOTS Colab en el proceso de colaboración con niños de preescolar.

5.3 Aporte de TITIBOTS Colab en el Proceso de Colaboración de la Primera Infancia

En Sección 3.2 se describió brevemente el desarrollo de los niños entre 4 y 6 años, donde se resalta que los niños en este rango de edad, por la etapa de desarrollo en que se encuentran, son egocéntricos, pero cuando empiezan a realizar acciones a favor de los demás, su egocentrismo inicial cede el paso a la conducta cooperadora (Zurbano Díaz de Cerio, 1998). Esto se puede potencializar incorporando actividades de aprendizaje colaborativo en el salón de clases.

Por este motivo, en esta investigación, el objetivo principal es evaluar el impacto de una herramienta colaborativa de programación en el proceso de colaboración en niños con edades entre 4 y 6 años. Cuando se desea evaluar el impacto, se debe medir usando una línea base, la cual se consigue mediante un cuasi-experimento con un diseño de series cronológicas con pre y post-test con dos grupos (Aedo, 2005). Así se puede valorar el aporte de la herramienta colaborativa de programación TITIBOTS Colab, que se propone en esta investigación, en el proceso de colaboración en niños con edades entre 4 y 6 años.

En la Sección 1.6 se dijo que la mayoría de las investigaciones sobre el proceso de colaboración han tratado de medir sus efectos, en general, a través de algún pre-test o post-test ganancia individuo con respecto a la ejecución de tareas (mediante medidas de desempeño). En lugar de medir el desempeño del trabajo de grupo (acciones determinadas que evidencian la interacción, la comunicación, el apoyo, la responsabilidad y el trabajar juntos por un fin mutuo). Por esto, se decidió realizar la evaluación mediante la medición del desempeño del grupo, en términos de las acciones esperadas para alcanzar cada componente esencial del aprendizaje colaborativo, y complementarla con medidas de desempeño de la ejecución de las tareas (por ejemplo: se logró, cantidad de intentos, tiempo promedio)

(Alvarez et al., 2011; A. Echeverría et al., 2011; Guerrero, Alarcón, & Collazos, 2000; Laboratorio de Innovación Educativa, 2012).

Por otra parte, cuando se realiza investigación en el área de Educación, frecuentemente, se trabaja con grupos naturales, y no hay asignación aleatoria de los sujetos. Estos diseños entran en la categoría de cuasi-experimentales. En relación con el tamaño de las muestras, el tamaño del grupo natural es el tamaño de la muestra. Por esto, se suele realizar estudios cuasi-experimentales con muestras pequeñas (Morales Vallejo, 2013, p. 28).

El cuasi-experimento que se realizó en esta investigación fue un diseño de dos series cronológicas, donde se trabajó con dos grupos naturales¹¹: el grupo experimental, donde se aplicó el tratamiento o variable independiente, y el grupo de control, no se le aplicó el tratamiento; con pre y post-test en los dos grupos (H. D. Echeverría, 2005, pp. 67-69; Morales Vallejo, 2013) (ver Figura 18). Se procuró que los grupos naturales fueran grupos equivalentes y comparables en todo, excepto en estar o no estar sometidos a la variable independiente.

Cada grupo natural estuvo conformado por 12 niños entre 4 y 6 años de edad. Los grupos fueron de dos instituciones diferentes, pero ambas instituciones tenían un perfil parecido. En cada grupo natural, se pudieron conformar cuatro grupos colaborativos formados por tres niños cada uno. La selección de los grupos se realizó de forma aleatoria. Además, ambos grupos estuvieron mediados por dos docentes: la docente a cargo del grupo y la docente de otro grupo de preescolar, ambas con perfil profesional parecido. La selección de la escuela donde se aplicaría el tratamiento fue realizada aleatoriamente, y la seleccionada fue la Escuela Santa Mónica.

La FOD se encargó de seleccionar las instituciones, poniendo en contacto a la investigadora con las directoras correspondientes para explicarles sobre la investigación y coordinar las actividades con las docentes. Además, brindó a las personas encargadas de observar y evaluar

¹¹ Conformado por sujetos que, en los diseños cuasi-experimentales, no se asignan de manera aleatoria ni se emparejan, sino que estaban formados antes del experimento (H. D. Echeverría, 2005).

las observaciones del pre-test y pos-test, y prestó algunos de los recursos utilizados en el cuasi-experimento.

Con este cuasi-experimento se desea contestar la cuarta pregunta de investigación:

¿Es posible que una herramienta colaborativa de programación pueda incentivar la colaboración en niños con edades entre 4 y 6 años?

En el cuasi-experimento se desea observar si hay un cambio en el grupo experimental, comprobar si la utilización de la herramienta colaborativa incentiva el proceso de colaboración en niños de preescolar. Con base en esto y a la pregunta de investigación se establecieron, de forma general, las siguientes hipótesis nula y alternativa del cuasi-experimento:

H_0 : No hay diferencia significativa en el porcentaje de acciones logradas del componente esencial X, al realizar un pre-test y un pos-test, en los dos grupos naturales

H_1 : Hay diferencia significativa en el porcentaje de acciones logradas del componente esencial X, al realizar un pre-test y un pos-test, en los dos grupos naturales

A continuación, se describen los elementos del diseño del cuasi-experimento (ver Figura 18):

- **Grupos naturales:** En cada grupo natural se tuvieron 12 niños entre 4 y 5 años, y se formaron 4 grupos colaborativos. Se cuenta con un grupo experimental y uno de control.
- **Variable Independiente (VI):** Herramienta colaborativa de programación TITIBOTS Colab.
- **Variables Dependientes (VD):** Porcentaje de acciones logradas del componente esencial del aprendizaje colaborativo:
 1. Independencia positiva.
 2. Interacción estimuladora (comunicación cara a cara).
 3. Responsabilidad individual y grupal.

4. Habilidades interpersonales y grupales (trabajo en equipo).
 5. Evaluación grupal (proceso de grupo).
- **Tratamiento experimental:** Taller de TITIBOTS Colab (utilización de la herramienta colaborativa de programación, ver Apéndice 17 para más detalles del taller realizado con la herramienta).
 - **Observaciones o mediciones:** Realización de tres actividades colaborativas en el aula.

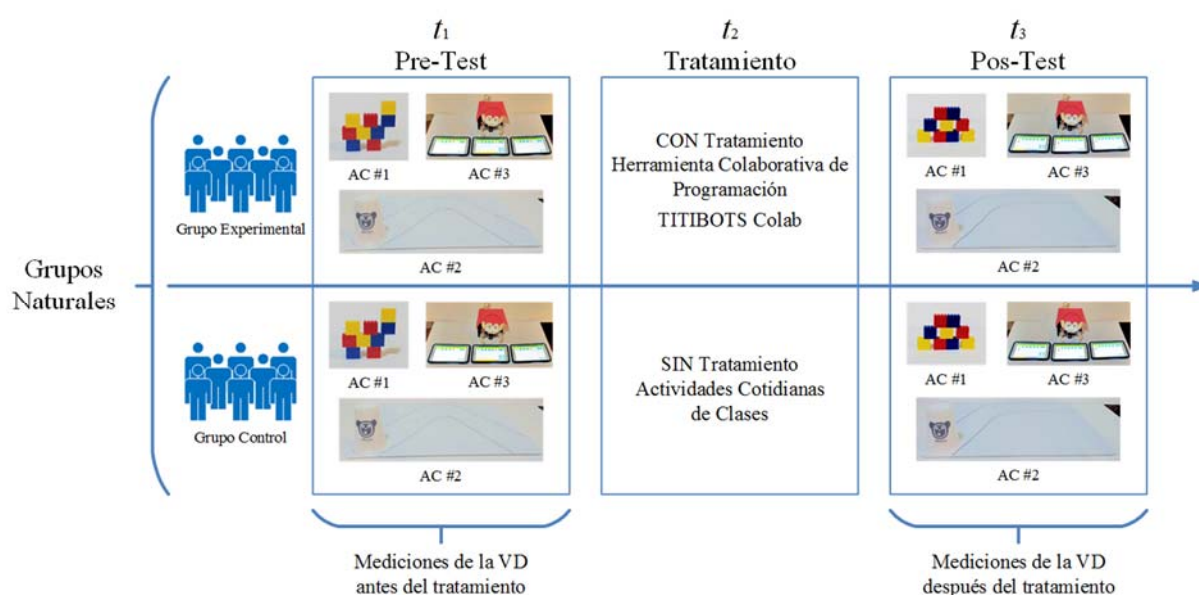


Figura 18. Diseño de dos series cronológicas realizado en la investigación.

Como parte de este proceso, se capacitaron las docentes de la escuela del grupo experimental sobre los contenidos del taller, en las instalaciones de la Escuela Santa Mónica, con un tiempo presencial de 4 horas. La capacitación sobre las actividades colaborativas a realizar en el pre-test y pos-test se realizó en instalaciones de ambas escuelas, con un tiempo presencial de 2 horas. En esta capacitación, se realizó la entrevista semiestructurada a las docentes sobre su perfil académico y laboral (ver Apéndice 10) con el fin de establecer que las instituciones y las docentes tuvieron un perfil parecido.

En las secciones siguientes se describe el proceso de experimentación del cuasi-experimento.

5.3.1 Participantes

Este experimento se realizó en la Escuela Santa Mónica (en Montelimar, Guadalupe, Costa Rica) con el grupo de kínder (12 niños, entre 4 y 5 años), y en la Escuela El Castillo (en Alajuela, Costa Rica) con el grupo de kínder (12 niños, entre 4 y 5 años) La evaluación fue realizada por un total de 12 personas, entre ellas: ocho observadores, cuatro mediadoras, un experto de soporte técnico y un camarógrafo.

5.3.2 Espacio Físico y Materiales

La evaluación se llevó a cabo en el salón de clases respectivo de cada grupo en la institución correspondiente. En cada salón se acomodaron cinco mesas (una por grupo colaborativo, con tres niños), cada una en forma de media luna, y con un espacio en el centro para que las docentes explicaran las actividades colaborativas a realizar. Cada grupo tenía asignado un observador al lado del grupo, que no interfería con el trabajo de los niños. Además, había tres evaluadores más en áreas estratégicas para observar el trabajo de todos los grupos.

En las actividades colaborativas realizadas en ambos test (pre y pos) se usaron bloques de construcción LEGO, cartulinas, vasos plásticos, tabletas (con la aplicación) y un robot (ver la Figura 19). En el grupo de kínder donde se realizó el tratamiento, se usaron tabletas con la aplicación (una por niño), y un robot compartido por todos los grupos colaborativos. Se definieron guías didácticas de trabajo, recursos para el taller (tratamiento), guías de trabajo y listas de cotejo (ver Apéndice 11) para la evaluación de cada actividad colaborativa.

5.3.3 Procedimiento

En el proceso de evaluación se realizó un pre-test y un pos-test en ambos grupos naturales. En el grupo experimental se aplicó el tratamiento: un taller conformado por siete sesiones (una hora y treinta minutos cada una, una sesión por día). Como parte del proceso se realizaron capacitaciones en ambas escuelas. En todo el proceso los investigadores tomaron notas, fotos y videos.



Figura 19. Materiales utilizados en el pre-test y pos-test.

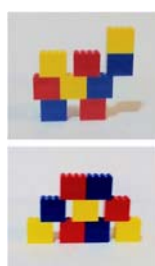
Por otra parte, la capacitación sobre las actividades colaborativas a realizar en el pre-test y pos-test se realizó en las instalaciones de ambas escuelas, con un tiempo presencial de 2 horas. En esta capacitación se explicó cada actividad colaborativa a realizar, brindando una guía didáctica. Además, las docentes realizaron todas las actividades colaborativas para entender a fondo cada una. En relación con la capacitación sobre el taller (tratamiento), se realizó en la escuela del grupo experimental (escogida aleatoriamente), en las instalaciones de la Escuela Santa Mónica con un tiempo presencial de 4 horas. Para la capacitación se diseñaron un plan de trabajo y un taller con contenidos sobre conceptos de colaboración, programación y robótica. Además, se diseñaron estrategias didácticas, material didáctico orientado a docentes y estudiantes, y actividades lúdicas. Todos los recursos diseñados fueron revisados y avalados por las docentes. La capacitación de la Escuela Santa Mónica se llevó a cabo el 21 y 22 de octubre del 2015 de 1:00p.m. a 4:00p.m., y en la Escuela El Castillo el 19 de octubre del 2015 de 1:00p.m. a 3:00p.m.

El pre-test en el grupo experimental se realizó el jueves 22 de octubre del 2015 de 8:00a.m. a 9:00a.m. y en el grupo de control el viernes 23 de octubre del 2015 de 8:00a.m. a 9:00a.m. En este test se realizaron tres actividades colaborativas con una duración de 20 minutos cada una. Antes de iniciar el test, se conformaron los grupos colaborativos, la cual fue de forma aleatoria. Los grupos siempre estuvieron conformados por los mismos niños durante todo el taller, es decir, fueron grupos formales (D. W. Johnson & Johnson, 1997). El pos-test en el grupo experimental se realizó el martes 10 de noviembre del 2015 de 8:00a.m. a 9:00a.m. y en el grupo de control el miércoles 11 de noviembre del 2015 de 8:00a.m. a 9:00a.m. En este

test se realizaron las mismas tres actividades colaborativas, pero con un objetivo diferente, con una duración de 20 minutos cada una.

En ambos tests (pre y pos), los grupos naturales realizaron las siguientes tres actividades colaborativas (dichas actividades eran iguales en ambos test, pero con un objetivo diferente) por ambos grupos naturales:

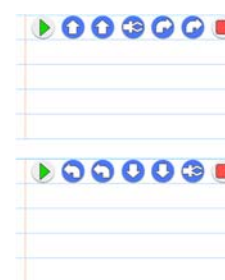
1. **Realización de un patrón con bloques de construcción.** Los grupos colaborativos armaron con los bloques de construcción el patrón establecido por la docente. En la Figura 20a se observan los dos patrones que los grupos debían realizar, uno para cada test.
2. **Balance de un objeto en una cartulina.** Los grupos colaborativos balancearon un objeto (un vaso con la figura de un animal) en una cartulina rectangular, en la cual se dibuja un camino que debe seguir el objeto de inicio a fin. En la Figura 20b se observan los dos caminos que los grupos debieron realizar, uno para cada test.
3. **Realización de un desafío usando TITIBOTS Colab.** Los grupos colaborativos debieron usar TITIBOTS Colab para realizar un desafío dado por la docente. En la Figura 20c se observan los dos desafíos que los grupos debían realizar, uno para cada test.



a) Objetivos de cada test de la Act.Colab. 1



b) Objetivos de cada test de la Act.Colab. 2



c) Objetivos de cada test de la Act.Colab. 3

Figura 20. Actividades del pre-test y pos-test.

En cada escuela se formaron un total de 5 grupos, cada grupo conformado por tres niños (solo un grupo de la Escuela Santa Mónica quedó con dos niños). A cada grupo se le asignó una imagen de un animal que los representaba como grupo: Mono, Tortuga, Pájaro, Delfín y

Cocodrilo. En la Escuela Santa Mónica (grupo experimental), el día del pre-test solo llegaron 12 niños, formándose solo 4 grupos, por lo que, para el pre-test y el pos-test, se usaron los datos de los 12 niños y los 4 grupos formados. Sin embargo, en los días del taller estuvieron presentes todos los niños y se trabajó con los 5 grupos. Por otro lado, en la Escuela El Castillo (grupo de control), uno de los grupos que se formó tenía un niño autista y en el pos-test un niño de este grupo faltó. Por estos motivos, se decidió tomar los datos de 12 niños de los otros 4 grupos formados. Así se tuvo igual cantidad de niños y grupos colaborativos en cada grupo natural.

En el pre-test y pos-test, cada grupo conformado por tres niños fue observado y evaluado por un investigador. El mismo investigador observó y evaluó al mismo grupo en ambos test. Participaron cinco investigadores encargados de los grupos colaborativos. Cada investigador se hizo cargo del mismo grupo en ambas escuelas, es decir, si en la Escuela Santa Mónica observó y evaluó a los Monos, en la Escuela El Castillo también observó y evaluó a los Monos. Los investigadores usaron una lista de cotejo para registrar cada actividad colaborativa realizada en los test (ver Apéndice 11).

La lista de cotejo registra las acciones de un grupo que trabaja de forma colaborativa debe realizar por componente esencial del aprendizaje colaborativo (ver Capítulo 3 y Figura 21), las cuales fueron definidas por Johnson, Johnson y Holubec (D. W. Johnson et al., 1998). Cada investigador debía indicar si la acción fue lograda o no por los niños para cada actividad colaborativa.

Al ser cinco los componentes esenciales del aprendizaje colaborativo, se tienen cinco variables dependientes (VD) en el cuasi-experimento:

1. Interdependencia positiva.
2. Interacción estimuladora (comunicación cara a cara).
3. Responsabilidad individual y grupal.
4. Habilidades interpersonales y grupales (trabajo en equipo).
5. Evaluación grupal (proceso de grupo).

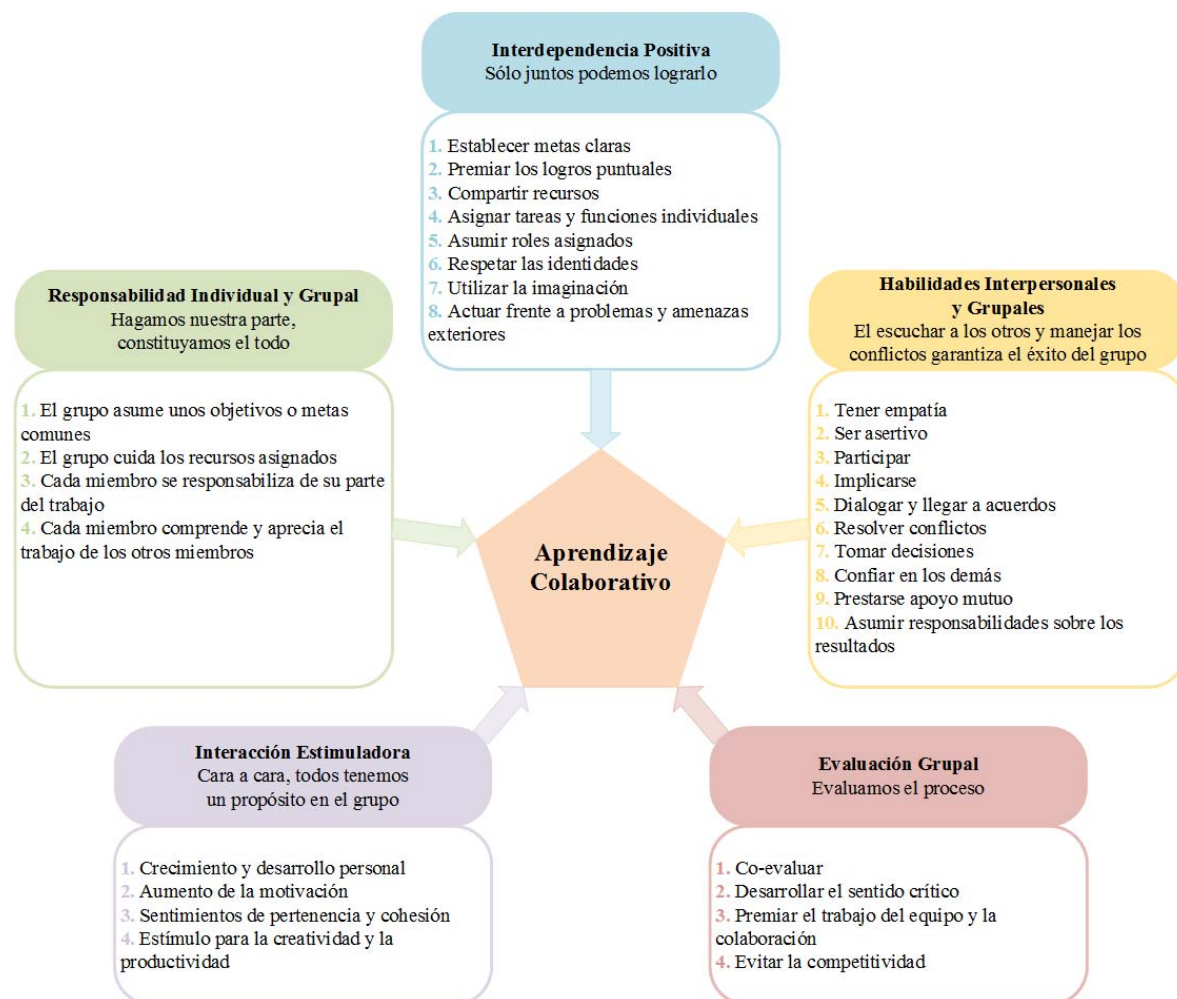


Figura 21. Acciones esperadas en cada componente esencial del aprendizaje colaborativo.

Con los resultados de la evaluación realizada en las actividades colaborativas para cada grupo en ambos test, se procedió a tabularlos. A cada niño se le calculó el porcentaje de acciones logradas en cada componente esencial (las cinco VD) por actividad colaborativa en los dos tests, y se calculó el promedio de los porcentajes de las tres actividades colaborativas para cada variable dependiente. Luego, si todos los miembros de cada grupo lograban la acción, al grupo se le ponía lograda la acción. Seguidamente, a cada grupo se le calculó el porcentaje de acciones logradas en cada componente esencial (las cinco VD) por actividad colaborativa en los dos tests, y se calculó el promedio de los porcentajes de las tres actividades colaborativas para cada variable dependiente.

Posteriormente, con cada variable dependiente (componente esencial) se procedió a realizar un análisis de contraste de medias en las puntuaciones diferenciales de las muestras independientes (Pérez Juste, Galán González, & Quintanal Díaz, 2012, p. 65), tanto para la muestra de los individuos como para la muestra de los grupos colaborativos. Por lo que, se calculó para cada observación una puntuación de cambio o puntuación diferencial, obteniendo para cada observación una única puntuación para analizar (esto se realizó en ambas muestras: grupos colaborativos e individuos).

Luego, con las puntuaciones diferenciales, se procedió a realizar la prueba-*t* con muestras independientes, para comprobar que, al aplicar el tratamiento en el grupo experimental, existieron diferencias estadísticamente significativas en cada una de las variables dependientes (ver Figura 22). Además, se comprobó que no hay diferencias significativas entre los grupos naturales en el pre-test; es decir, existe comparabilidad inicial de ambos grupos naturales.

Al mismo tiempo, se realizó la prueba *U* de Mann-Whitney (prueba no paramétrica alternativa de la prueba-*t* con dos muestras independientes) para comprobar la heterogeneidad de las dos muestras ordinales de puntuaciones diferenciales: la distribución de partida de ambos grupos naturales es la misma o los valores de una de las muestras tienden a exceder a los de la otra (Downie & Heath, 1973). Este tipo de prueba se realizó para los estudiantes (datos individuales) y los grupos colaborativos (datos grupales) con ambos grupos naturales (las dos instituciones) (ver Figura 23). La prueba *U* de Mann-Whitney se realizó porque se tuvieron muestras pequeñas¹² y, al realizar la prueba-*t* se puede incurrir en el error tipo II, el cual se comete cuando el investigador no rechaza una hipótesis nula falsa (Downie & Heath, 1973), por lo que en los resultados, no se ve diferencia, pero puede haberla.

¹² Las pruebas no paramétricas se utilizan cuando no se cumplen las condiciones exigidas para la aplicación de las paramétricas y, cuando las muestras son pequeñas y falta información respecto de la densidad de probabilidad. (Downie & Heath, 1973)

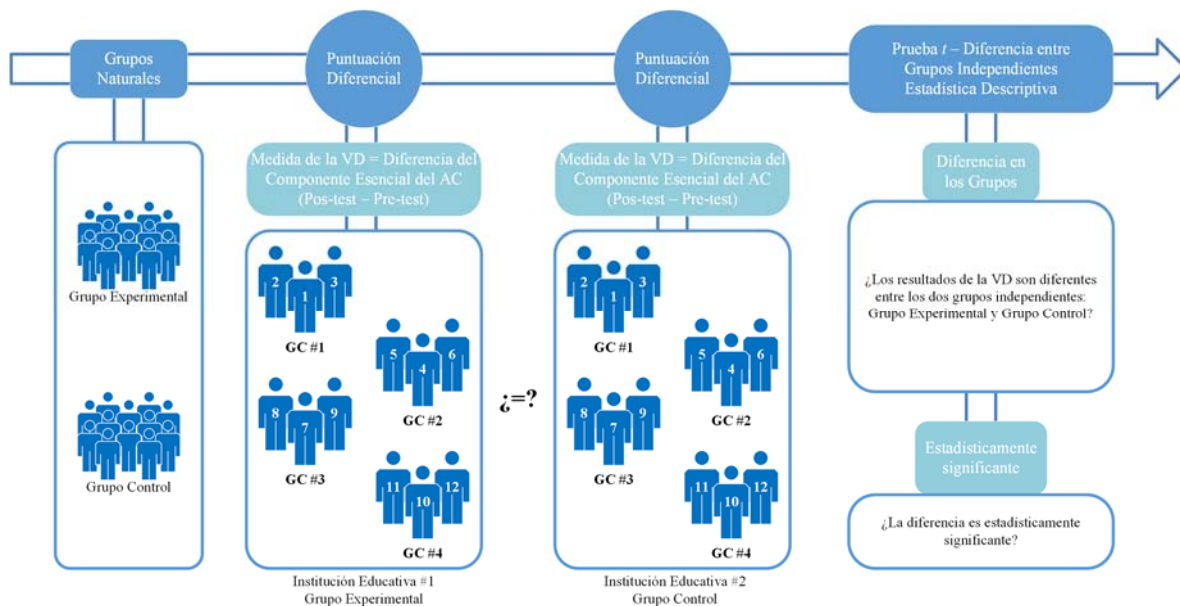


Figura 22. Prueba *t*: Diferencia entre grupos independientes para puntuaciones diferenciales.

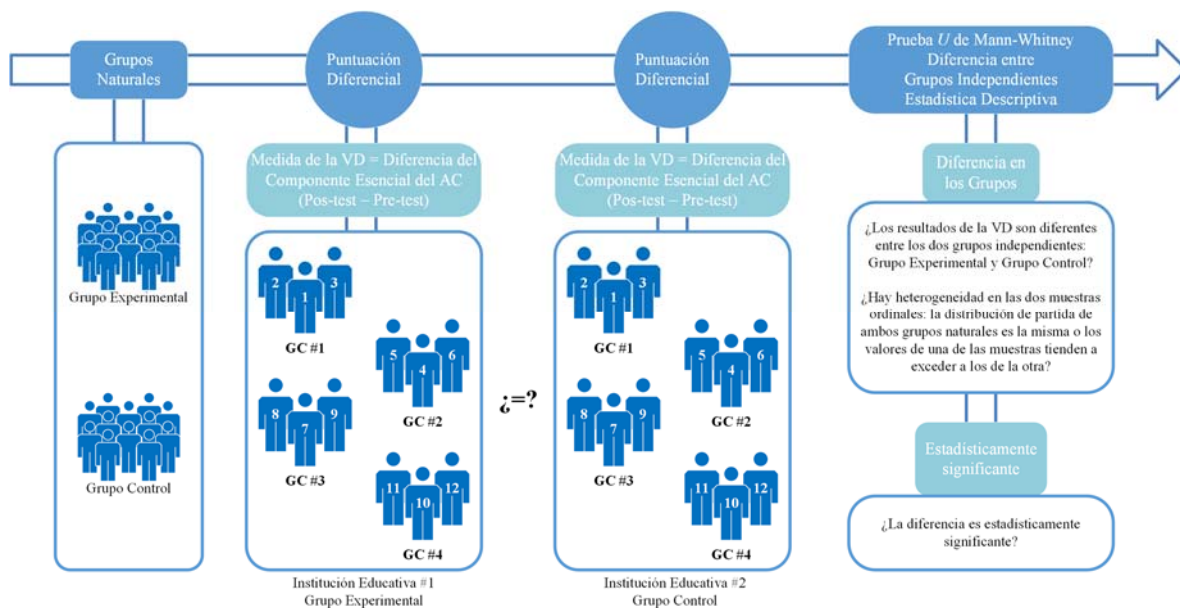


Figura 23. Prueba *U* de Mann-Whitney: Diferencia entre grupos independientes para puntuaciones diferenciales.

Finalmente, se calculó el tamaño del efecto¹³, complemento del contraste de medias tradicional. Debido a que el tamaño del efecto muestra una diferencia tipificada (diferencia de medias estandarizada) que informa cuántas desviaciones típicas de diferencia hay entre los resultados de los dos grupos comparados (Glass, McGaw, & Smith, 1981; Morales Vallejo, 2012; Pérez Juste et al., 2012).

El procedimiento general realizado en el grupo experimental fue el siguiente:

- Capacitación de las docentes y asistentes para la aplicación de la estrategia, las actividades del taller y el uso de la herramienta colaborativa de programación.
- Entrega de circular a los padres de familia para informar del taller, solicitar la autorización respectiva y enviarles la entrevista semiestructurada del perfil de usuario.
- Realización de las actividades colaborativas del pre-test.
- Realización del taller para la utilización de TITIBOTS en la solución de problemas de forma colaborativa.
- Realización de las actividades colaborativas del pos-test.
- Realización de cuestionarios a los niños participantes
- Realización de cuestionarios a las docentes participantes.

El procedimiento general realizado en el grupo de control fue el siguiente:

- Capacitación de las docentes y asistentes para la aplicación de la estrategia.
- Entrega de circular a los padres de familia para informar del taller, solicitar la autorización respectiva y enviarles la entrevista semiestructurada del perfil de usuario.
- Realización de las actividades colaborativas del pre-test.
- Realización de las actividades colaborativas del pos-test.

¹³ El efecto es la medida de la fuerza de un fenómeno, por ejemplo: cambio en el resultado después de una intervención experimental. El cálculo del tamaño del efecto va más allá de la significancia estadística para conocer la relevancia práctica; ya que una diferencia estadística significativa no dice si tal diferencia es importante, sino tan solo que existe diferencia en la población de referencia (Pérez Juste et al., 2012).

El taller tuvo una duración de siete días (una hora y media cada día: de 8:00a.m. a 9:30a.m.), donde se realizaron una serie de actividades lúdicas para usar la herramienta colaborativa con el objetivo de realizar desafíos de forma colaborativa (ver Sección 5.2.2). En las últimas dos sesiones del taller se realizaron desafíos para realizar la evaluación de los conocimientos adquiridos, así como la usabilidad y funcionalidad de la herramienta. En esta parte, se contabilizó la cantidad de desafíos realizados, la cantidad de intentos y el tiempo de duración de cada grupo colaborativo. En total se realizaron seis desafíos, divididos en tres desafíos por día.

Por cuestiones éticas, nos comprometimos con la escuela del grupo de control a brindarles el taller en la última semana de febrero del 2016, y la capacitación correspondiente a las docentes en la segunda semana de febrero del 2016.

Entre los factores de validez interna se controlaron en el cuasi-experimento los siguientes (Cambell & Stanley, 1995):

- **Maduración.** Las actividades realizadas en ambos test y en el taller (tratamiento) se planearon para que fueran atractivas y cortas. En un día no se sobrepasó de 1 hora y 30 minutos de trabajo con los niños (este tiempo es el ideal para trabajar con niños de preescolar, según se ha verificado en diferentes pilotos).
- **Administración del test (o pruebas).** Un aspecto importante en este factor es que el efecto de la docente no se puede controlar; ya que la docente sea buena, mala o que no participe, siempre afectará el experimento (de buena o mala forma). A pesar de esto, sí se controló la calidad de la mediación, mediante la guía desarrollada y la capacitación brindada a las docentes, donde se les indicaron a las docentes el tipo de mediación que debían llevar a cabo y cómo realizar las diferentes actividades.
- **Instrumentación.** La medición de cada variable dependiente se realizó mediante listas de cotejo, que fueron utilizadas por los investigadores. Un investigador evaluaba dos grupos colaborativos, uno en cada escuela.

- **Mortalidad experimental o diferencial.** En los test hubo dos grupos (uno en cada Escuela) que no fueron tomados en cuenta para el análisis de los resultados, ya que se presentaron situaciones que se comentaron anteriormente.

Por otra parte, entre los factores de validez interna que no se controlaron en el cuasi-experimento se encuentran (Cambell & Stanley, 1995):

- **Historia.** La historia asociada a los niños que participan en el experimento no se puede controlar, si tienen o no contacto con dispositivos móviles, aplicaciones colaborativas, entre otros.
- **Distribución.** Los grupos que se usan son naturales, por lo que no se realizó asignación al azar de los sujetos al grupo experimental o de control.

Los factores de validez externa, por el tipo de diseño utilizado en este cuasi-experimento no suceden o no son pertinentes (Cambell & Stanley, 1995).

Además, se complementó la evaluación con las medidas de desempeño de la ejecución de las actividades colaborativas y las observaciones finales solicitadas a los evaluadores. Las medidas de desempeño fueron el logro de la actividad colaborativa, la cantidad de intentos y el tiempo por grupo. En las observaciones finales, se les solicitó a los evaluadores, comentar su apreciación cuanto a los roles: si trabajaron juntos, si hubo comunicación, si hubo cohesión del grupo; es decir, aspectos importantes del proceso de colaboración e información que pudieran brindar con respecto a la evaluación realizada en el pre-test y el pos-test.

5.3.4 Análisis de Datos

En el análisis de los datos se utilizó estadística descriptiva. Se realizaron diferentes pruebas estadísticas usando el paquete estadístico SPSS.

Las pruebas-*t* y las pruebas *U* de Mann-Whitney se realizaron siguiendo las guías de (Laerd Statistics, 2015a) y (Laerd Statistics, 2015b). En todas las pruebas estadísticas realizadas en esta investigación se utilizó un nivel de significancia $\alpha = 0,05$.

En el capítulo siguiente se describe el proceso de creación de la herramienta colaborativa de programación TITIBOTS orientada a niños entre 4 y 6 años.

5.4 Proceso de Diseño Centrado en el Usuario

En esta investigación participaron 10 expertos de la Unidad de Aprendizaje lógico, científico y robótica (bajo la coordinación de la Máster Ana Lourdes Acuña Zúñiga), 2 expertas de PROTEA, 6 investigadores del RExLab-UFSC (bajo la coordinación del Dr. Juarez Bento da Silva), 15 docentes de preescolar y 5 investigadores de USING. Además, participaron en las diferentes evaluaciones, aproximadamente 100 niños con edades entre 4 y 6 años.

La metodología de trabajo utilizada en la investigación fue el proceso de DCU, con el fin de lograr cada uno de los objetivos específicos, y con ello comprobar el objetivo general de esta investigación. En la Figura 24 se muestra un diagrama que resume el proceso metodológico de la investigación para cada objetivo específico, el cual se describió en este capítulo.

El diagrama del proceso metodológico de esta investigación se completa con la Figura 25, que muestra el proceso iterativo de DCU del estándar ISO 13407 (International Organization for Standardization, 1999). El DCU permite evaluar la usabilidad del diseño de forma iterativa y mejorarlo incrementalmente utilizando en cada fase una serie de técnicas. Las técnicas utilizadas en la investigación fueron tomadas del estándar ISO TR-18529 (International Organization for Standardization, 2000).

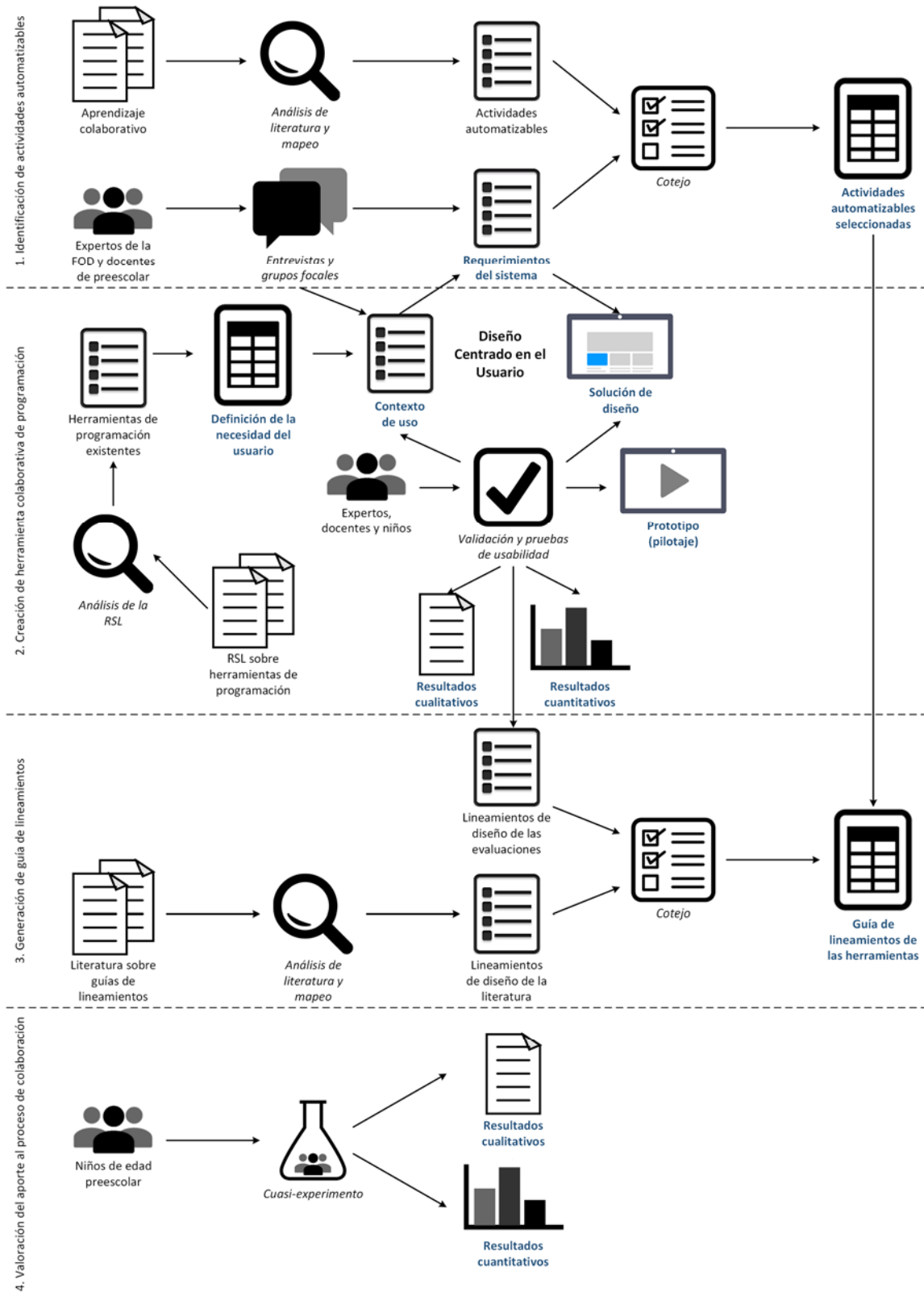


Figura 24. Diagrama del proceso metodológico de la investigación.

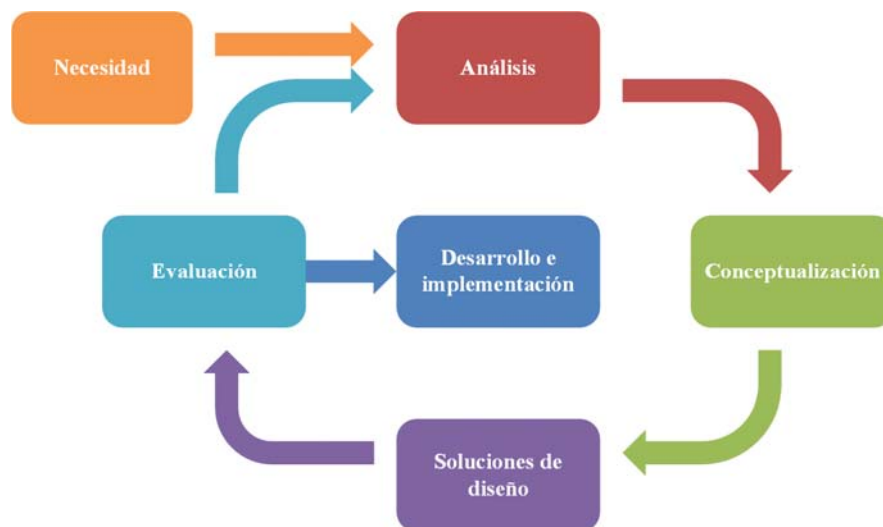


Figura 25. Proceso de DCU.

Dado que el proceso es iterativo, en esta investigación se realizaron siete iteraciones del proceso de DCU para llegar a tener dos herramientas de programación simples, intuitivas y fáciles de usar para los niños entre 4 y 6 años. De las siete iteraciones, una fue para ver la factibilidad técnica de la investigación, cuatro fueron para crear la herramienta individual de programación TITIBOTS y dos para crear la herramienta colaborativa de programación TITIBOTS Colab. Cada iteración tuvo la necesidad de resolver el problema de investigación (planteado en la Sección 1.7) y, se desarrollaron e implementaron dos pilotos públicos con dos prototipos funcionales y usables. Cada una de estas iteraciones se describieron en este capítulo y se resumen a continuación.

Iteración 0

- **Necesidad.** Problema de investigación, enfocado en comprobar la factibilidad técnica de las herramientas de programación que se crearon.
- **Análisis.** Técnicas utilizadas en esta fase: revisión de literatura y especificación del contexto de uso.
- **Conceptualización.** Técnicas utilizadas en esta fase: identificación y documentación del entorno técnico.
- **Soluciones de diseño.** Técnica utilizada en esta fase: desarrollo de prototipo.

- **Evaluación.** Técnica utilizada en esta fase: test de funcionalidad.
- **Desarrollo e implementación.** El resultado de esta iteración fue generar soluciones de diseño al comprobar la factibilidad técnica, en busca de realizar pilotos públicos.

Iteración 1

- **Necesidad.** Problema de investigación, enfocado en generar soluciones de diseño para validarlas con los expertos y consultores.
- **Análisis.** Técnicas utilizadas en esta fase: revisión sistemática de literatura, visitas de campo, método de observación y, entrevistas con expertos de la FOD y consultores (docentes de preescolar).
- **Conceptualización.** Técnica utilizada en esta fase: especificación de requerimientos.
- **Soluciones de diseño.** Técnicas utilizadas en esta fase: esbozado grupal y desarrollo de prototipos en papel.
- **Evaluación.** Técnica utilizada en esta fase: validación con expertos de la FOD y consultores (docentes de preescolar).
- **Desarrollo e implementación.** El resultado de esta iteración fue desarrollar un prototipo en papel (concepto, interfaz e interacción) para validarlo con niños de edad preescolar, en busca de realizar pilotos públicos.

Iteración 2

- **Necesidad.** Problema de investigación, enfocado en validar el prototipo en papel desarrollado en la iteración anterior con niños de edad preescolar.
- **Análisis.** Técnicas utilizadas en esta fase: lluvias de ideas y conceso de expertos de la FOD y consultores (docentes de preescolar).
- **Conceptualización.** Técnica utilizada en esta fase: especificación de requerimientos.
- **Soluciones de diseño.** Técnica utilizada en esta fase: desarrollo de prototipo en papel (concepto, interfaz e interacción).

- **Evaluación.** Técnica utilizada en esta fase: validación con usuarios finales (niños de edad preescolar).
- **Desarrollo e implementación.** El resultado de esta iteración fue obtener una solución de diseño (concepto, interfaz e interacción) para comenzar con la implementación, en busca de realizar pilotos públicos.

Iteración 3

- **Necesidad.** Problema de investigación, enfocado en implementar la solución de diseño resultante de la iteración anterior.
- **Análisis.** Técnicas utilizadas en esta fase: grupo focal con consultores de PROTEA (docentes de preescolar) y revisión de literatura enfocada en lineamientos para herramientas de aprendizaje orientadas a niños de edad preescolar.
- **Conceptualización.** Técnica utilizada en esta fase: diseño de guía de lineamientos.
- **Soluciones de diseño.** Técnicas utilizadas en esta fase: desarrollo de prototipo y piloto.
- **Evaluación.** Técnica utilizada en esta fase: test de usabilidad con usuarios finales (niños de edad preescolar).
- **Desarrollo e implementación.** El resultado de esta iteración fue obtener un prototipo funcional y usable preliminar para continuar con más evaluaciones con usuarios finales, en busca de realizar pilotos públicos.

Iteración 4

- **Necesidad.** Problema de investigación, enfocado en mejorar el prototipo de la iteración anterior.
- **Análisis.** Técnicas utilizadas en esta fase: método de observación y grupo focal con expertos de la FOD y consultores de PROTEA (docentes de preescolar).
- **Conceptualización.** Técnicas utilizadas en esta fase: diseño de guía de lineamientos y proceso de refinamiento.

- **Soluciones de diseño.** Técnicas utilizadas en esta fase: desarrollo de prototipo y piloto.
- **Evaluación.** Técnica utilizada en esta fase: test de usabilidad con usuarios finales (niños de edad preescolar).
- **Desarrollo e implementación.** El resultado de esta iteración fue obtener un prototipo funcional y usable de la herramienta de programación TITIBOTS para niños entre 4 y 6 años, en busca de realizar pilotos públicos. Con la herramienta de programación individual TITIBOTS se llevó a cabo un piloto público con niños de edad preescolar.

Iteración 5

- **Necesidad.** Problema de investigación, enfocado en crear la versión colaborativa del prototipo resultante de la iteración anterior.
- **Análisis.** Técnicas utilizadas en esta fase: revisión sistemática de literatura, método de observación y grupo focal con expertos de la FOD y consultores del RExLab.
- **Conceptualización.** Técnicas utilizadas en esta fase: diseño de guía de lineamientos y especificación de requerimientos.
- **Soluciones de diseño.** Técnicas utilizadas en esta fase: desarrollo de prototipo y piloto.
- **Evaluación.** Técnica utilizada en esta fase: test de usabilidad con usuarios finales (niños de edad preescolar).
- **Desarrollo e implementación.** El resultado de esta iteración fue obtener un prototipo funcional y usable de la herramienta colaborativa de programación TITIBOTS Colab para niños entre 4 y 6 años, en busca de realizar pilotos públicos.

Iteración 6

- **Necesidad.** Problema de investigación, enfocado en mejorar el prototipo de la iteración anterior.

- **Análisis.** Técnicas utilizadas en esta fase: método de observación y grupo focal con expertos de la FOD y consultores de PROTEA (docentes de preescolar).
- **Conceptualización.** Técnica utilizada en esta fase: proceso de refinamiento.
- **Soluciones de diseño.** Técnicas utilizadas en esta fase: desarrollo de prototipo y piloto.
- **Evaluación.** Técnicas utilizadas en esta fase: test de usabilidad con usuarios finales (niños de edad preescolar) y cuasi-experimento (para evaluar el impacto de la herramienta colaborativa de programación en el proceso de colaboración en niños con edades entre 4 y 6 años).
- **Desarrollo e implementación.** El resultado de esta iteración fue obtener un prototipo funcional y usable de la herramienta colaborativa de programación TITIBOTS Colab para niños entre 4 y 6 años, que incentiva el proceso de colaboración, en busca de realizar pilotos públicos. Con la herramienta colaborativa de programación TITIBOTS Colab se llevaron a cabo pilotos públicos con niños de edad preescolar.

CAPÍTULO 6. TITIBOTS

En este capítulo se describe las herramientas de programación desarrolladas como parte de esta investigación: TITIBOTS (herramienta individual) y TITIBOTS Colab (herramienta colaborativa).

6.1 Descripción General

A nivel mundial existe un déficit alto de profesionales con buenas habilidades en las áreas de STEM. Esta realidad no es ajena a Costa Rica, donde se tiene una demanda de estos profesionales mucho mayor a la oferta actual. El país se enfrenta al reto de despertar el interés de niños por la tecnología desde tempranas edades. Para cumplirlo, la mejor forma es el aprendizaje de la programación y la robótica de manera lúdica. Sin embargo, las herramientas que existen en el mercado para este propósito son costosas y los docentes carecen de los conocimientos requeridos. Esta ha sido la razón fundamental para desarrollar la investigación presentada en este trabajo.

Como parte de esta investigación, se desarrollaron dos herramientas de programación orientadas a niños entre 4 y 6 años. Una de ellas apoya el aprendizaje colaborativo de la programación y puede ser utilizada en el salón de clases con presencia de mediación docente.

La investigación inició con la creación de la herramienta de programación TITIBOTS, la cual es una herramienta móvil de uso individual que permite a los niños de preescolar programar robots. Mientras se divierten jugando aprenden fundamentos de programación y robótica.

Esta herramienta fue la base en la creación de la herramienta colaborativa de programación TITIBOTS Colab. Esta herramienta móvil es uno de los principales aportes de la investigación, ya que incentiva el desarrollo de habilidades blandas y técnicas en niños de edad preescolar, necesarias para asegurarles un futuro lleno de oportunidades.

Ambas herramientas de programación fueron diseñadas con un conjunto abierto de comandos, que forma parte del protocolo de comunicación creado para ellas, para permitir el uso de cualquier robot, al implementar en ellos un interpretador de los comandos. Además, se implementaron en tres idiomas: español, inglés y portugués. La selección del idioma se realiza de forma automática según la configuración del idioma de la tableta. En la Tabla 10 se resumen las características principales de las dos herramientas construidas.

Tabla 10. Características de las herramientas de programación.

Herramienta	TITIBOTS	TITIBOTS Colab
Descripción	Herramienta de programación desarrollada para los niños de la primera infancia; la cual les permite crear un programa y enviarlo a un robot con el fin de ejecutarlo. Los niños de entre 4 y 6 años son nuestra meta de usuario. La herramienta ofrece una sencilla interfaz gráfica usando un enfoque iconográfico, es intuitiva y fácil de usar para los niños. La herramienta es para dispositivos móviles Android y fue diseñada para permitir el uso de cualquier robot con un conjunto abierto de comandos. TITIBOTS ha sido utilizado con el kit robótico LEGO Mindstorms NXT, y robots que usan tecnología Arduino.	Herramienta colaborativa de programación desarrollada para los niños de preescolar en entornos educativos. TITIBOTS Colab permite a los niños desarrollar un programa trabajando colaborativamente y enviar el programa creado a un robot para su ejecución. La herramienta ofrece una sencilla interfaz gráfica usando un enfoque iconográfico, es intuitiva y fácil de usar para los niños. TITIBOTS Colab fue diseñado para permitir el uso de cualquier robot con un conjunto abierto de comandos con una arquitectura centralizada. La comunicación entre las tabletas y el robot es a través de WiFi Peer-to-Peer; el docente puede monitorear cada grupo y evaluar el trabajo realizado. Esta versión corre en los dispositivos Android.
Rango de Edad	4-6	4-6
Paradigma de Programación	Programación visual (gráfica)	Programación visual (gráfica)
Componentes de Programación	Íconos y Sonidos	Íconos y Sonidos
Tipo de Interacción	<i>Drag and Drop</i>	<i>Drag and Drop</i>
Robots	Mono Tití (LEGO Mindstorms NXT-EV3 y Arduino)	Mono Tití (LEGO Mindstorms NXT-EV3 y Arduino), Serpiente Cora (Raspberry PI)
Fuente de Enlace	Móvil (Android)	Móvil (Android)
Comunicación (Herramienta y Robot)	Bluetooth y WiFi	WiFi
Herramienta Colaborativa	No	Sí

El personaje de ambas herramientas es un mono, específicamente, un mono tití (mono ardilla). Se escogió debido a que es un animal representativo de la fauna de Costa Rica, y en la investigación se quería usar su belleza natural característica. Además, a los niños les gustan mucho los monos, un animal divertido y gracioso. Se tiene un personaje virtual en la herramienta y los robots utilizados se disfrazan de monos. Los disfraces fueron hechos con Goma EVA (*Ethylene Vinyl Acetate*, también conocido como *Foam* o *Foamy*).

Al tener al mono tití como personaje principal y usar las herramientas para que los niños aprendan a programar robots, se decidió poner como nombre a las herramientas TITIBOTS: TITI por el mono tití y BOTS por robots.

Con la herramienta TITIBOTS los niños pueden programar un robot a través de un software altamente intuitivo y didáctico, usando un estilo de programación gráfica en el cual van arrastrando comandos que se juntan de forma secuencial para darle instrucciones al robot. El programa creado por los niños se envía al robot, y este ejecuta las órdenes generadas por ellos.

La metodología de enseñanza-aprendizaje se basa en la experimentación. El niño aprende jugando, y para este propósito se les plantean desafíos que deben resolver a través de la programación del robot.

En la siguiente sección se muestra la guía de lineamientos desarrollada como parte del segundo objetivo de esta investigación.

6.2 Guía de Lineamientos

Al diseñar una aplicación orientada a niños, se enfrenta un importante reto, ya que los niños son una población con habilidades, preferencias y necesidades diferentes al “usuario medio”, y estas van cambiando conforme el niño crece. A este hecho hay que sumarle que, en esta etapa de la vida, las diferencias de género en la relación, uso y consumo de las nuevas tecnologías son mayores que nunca (Hassan Montero, 2004).

Por tanto, debemos tener en cuenta que los niños representan un tipo de audiencia heterogénea, para los que no existe una receta única de diseño. De este modo, el diseño de aplicaciones a este tipo de audiencia requiere de la adopción y la adaptación metodológica necesaria para asegurar su usabilidad (Hassan Montero, 2004).

En uno de esos estudios se muestran algunas diferencias entre niños y adultos al usar sitios web, las cuales dan pistas para saber cómo diseñar juegos para niños (Jakob Nielsen, 2002):

- **Los sonidos y las animaciones son apreciados de forma positiva por los niños.** Factores emocionales como la diversión juegan un papel crucial en el diseño de sitios web usables para niños, motivándolos en la interacción y en la consecución de objetivos.
- **Los niños suelen recorrer la pantalla con el ratón.** Lo hacen para ver qué zonas están disponibles para hacer clic o simplemente para disfrutar de los efectos de sonido que reproducen los diferentes elementos de la interfaz al pasar el ratón por ellos.
- **Las metáforas funcionan.** El objetivo del uso de metáforas en el diseño es hacer familiar y comprensible lo desconocido. Además, el uso de metáforas visuales (ya sean geográficas o de otro tipo) es una acertada decisión de diseño para niños, ya que estos presentan menos conocimiento, habilidad y capacidad para la lectura.
- **Los niños no suelen usar la barra de desplazamiento.** Este hecho, que en un principio también se daba en usuarios adultos, es posible que cambie con el tiempo (por los mouses con rueda de *scroll*), aunque por el momento sugiere que en un diseño orientado a niños funcionarán mejor las páginas cortas.
- **Los niños tienden a leer las instrucciones de uso.** Aun así, cualquier sección de ayuda debería ser redactada de la forma más sintética posible, sin palabras complicadas ni tecnicismos.

Nielsen concluye en su estudio que los niños quieren buenas gráficas, facilidad de navegación, animación, interacción, narrativa amigable y entretenimiento. Además, si se quiere diseñar aplicaciones para aprendizaje, estas deben ser educativas y divertidas. Por otro

lado, la documentación o la ayuda debe ser ilustrada (fotos o dibujos) y escrita especialmente para la audiencia meta.

Con base en las conclusiones del estudio de Nielsen, los principios y heurísticas generales de HCI (Jakob Nielsen, 1994a), los estudios de varios autores (Chau, 2014; Chaudron, 2015; Díaz, Harari, & Amadeo, 2013; Leavitt & Shneiderman, 2013; Pane, 2002; Mitchel Resnick, 1998; Sesame Workshop, 2012) y los resultados que se obtuvieron en las diferentes evaluaciones llevadas a cabo en esta investigación, se creó una guía de los lineamientos pertinentes a la investigación obtenidos de este proceso.

Esta guía de lineamientos se muestra de forma clasificada (las categorías de la clasificación fueron tomadas de Sesame Workshop). También, de la Figura 26 a la Figura 33 se muestra un diagrama por cada una de las categorías de lineamientos generados. En cada diagrama se establece la relación entre la literatura consultada y las evaluaciones de la investigación de donde se obtuvieron los lineamientos de cada categoría.

Diseño de Interacción

1. **Personaje.** La elección del personaje es importante, ya que el personaje debe crear en los niños un sentimiento de amistad y debe ser divertido. Esto es fundamental para construir una buena relación entre los niños y la aplicación, creando experiencias amenas en medios digitales. Además, se crea un ambiente de compromiso prolongado y de diálogo entre los niños y el personaje digital.
2. **Saludos.** La mayoría del contenido comienza con un personaje amigable saludando al niño.
3. **Instrucciones.** Se debe indicar el objetivo y la forma de lograrlo.
4. **Tiempos de espera.** Usar tiempos de espera para realizar sugerencias concisas de qué hacer a continuación. Deben producirse después de 6-8 segundos de inactividad.
5. **Respuestas incorrectas (buenos mensajes de error).** La aplicación debe reportar los errores al niño en un lenguaje claro, usando instrucciones precisas y brindar ayuda constructiva para resolver el problema. Una respuesta incorrecta es una oportunidad

para el aprendizaje. El uso de retroalimentación visual y de audio debe ser alentadora e incremental. La retroalimentación ante una respuesta incorrecta se produce principalmente en tres niveles de andamiaje:

- a. **Primera respuesta incorrecta.** Identificar la elección incorrecta y ofrecer aliento, por ejemplo: “Eso no es correcto. ¡Inténtalo de nuevo!”.
 - b. **Segunda respuesta incorrecta.** Identificar la elección incorrecta, reformular el objetivo, ofrecer una sugerencia y proporcionar estímulo. Por ejemplo: “Eso no es correcto. Es necesario encontrar los comandos incorrectos. Los comandos incorrectos se resaltan con rojo. ¡Inténtalo de nuevo!”.
 - c. **Tercera respuesta incorrecta.** Identificar la elección incorrecta, reformular el objetivo, ofrecer una sugerencia, y resaltar la respuesta correcta. Por ejemplo: “Eso no es correcto. Es necesario encontrar los comandos incorrectos. Los comandos incorrectos se resaltan con rojo. <Resaltar comandos incorrectos>. ¡Elimina los comandos incorrectos!”. Esto se repite hasta que se seleccione la respuesta correcta. En algunos casos, se puede sugerir que el niño avance, si lleva un período de tiempo determinado sin lograrlo.
6. **Respuestas correctas.** Las recompensas son importantes para los niños, ya que los motiva. Si la recompensa es de audio, se debe incluir efectos de sonido (por ejemplo: ruidos de celebración, ding-ding-ding, entre otros). Cuando sea posible, se debe incluir también una recompensa visual a través de una animación o un movimiento de luces.
 7. **Retroalimentación.** Algunas actividades por lo general no tienen respuestas incorrectas o correctas. Sin embargo, la retroalimentación es necesaria en las formas de estímulo y reacción a la entrada del niño. La aplicación debe informar continuamente al niño acerca de lo que está haciendo y cómo se está interpretando su entrada.
 8. **Ayuda.** La ayuda y la documentación deben proporcionar una forma rápida a los niños de encontrar información específica de la tarea cuando están teniendo algún problema. Por lo que debe tener ilustraciones (fotos o dibujos) y debe estar escrita

especialmente para la audiencia meta (objetivo). Puesto que los niños en edad preescolar normalmente no saben leer, no es útil tener secciones de texto de ayuda. En su lugar, se recomienda la ayuda en forma de diálogos específicos del contexto (audio) y de refuerzo visual. Se sugiere, además, incluir una sección de ayuda dirigida a los padres.

9. **Gestos de interacción.** La navegación y la interacción en la aplicación deben ser fáciles de usar. Los gestos de interacción más intuitivos para los niños son: *tap* (dar clic), dibujar/mover dedo, *swipe* (deslizar), *drag and drop* (arrastrar y soltar) y *slide* (deslizar); gestos de interacción menos intuitivos son: *pinch* (apretar), *tilt/shake* (inclinarse/sacudir), *multi-touch* (múlti-táctil), *flick/flip* (voltear/lanzar), *double tap* (dar doble clic).
10. **Atajos.** La aplicación debe posibilitar que los niños con experiencia lleven a cabo tareas de uso frecuente con rapidez.
11. **Salidas claramente marcadas.** La aplicación debe ofrecer al niño una forma fácil de salir de diferentes situaciones como sea posible, incluyendo modos de deshacer.
12. **Prevenir errores.** Cuando sea posible, la interfaz de usuario debe ser estructurada para evitar situaciones de error.
13. **Fin del juego / Volver a jugar.** El diseño interactivo debe considerar cómo terminar la experiencia y cómo volver a reiniciarla.

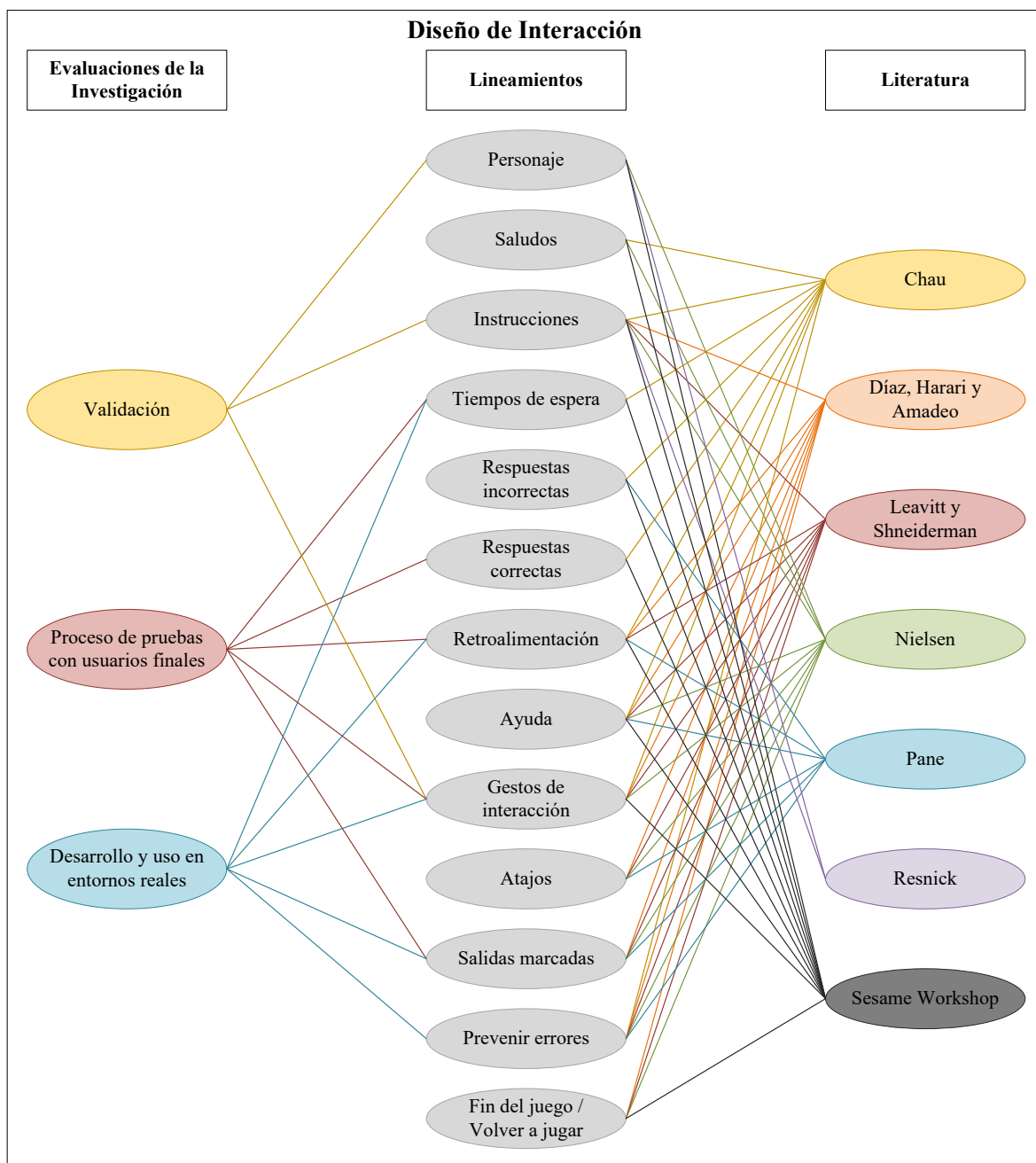


Figura 26. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño de Interacción”.

Diseño de Pantalla

1. **Diseño gráfico.** El diseño gráfico de las interfaces debe ser agradable y entretenido para los niños, pero se debe minimizar su carga gráfica (por ejemplo: disminución de

los colores y figuras del fondo) para eliminar distracciones visuales que obstaculicen la atención de los niños en los componentes relevantes de la aplicación.

2. **Cierres visuales.** Los cierres visuales ayudan a centrar la atención de los niños en los elementos relevantes de la aplicación. La habilidad de cierre visual tiene un papel importante en el procesamiento de la información visual, y participa en un gran número de actividades académicas y de la vida cotidiana de las personas.
3. **Uso de colores.** Los colores primarios son los tres colores naturales (básicos) a partir de los cuales se pueden conseguir otros colores (son la pieza clave a partir de la cual se construye el círculo cromático) y, los colores secundarios son los colores que se obtienen de la mezcla de dos colores primarios y son un color complementario de un tercer color primario (aquel que no interviene en su elaboración). Se deben usar en aplicaciones para niños de preescolar colores primarios y secundarios. Debido a que los niños de preescolar deben conocer los colores en su alrededor, comenzando con los colores primarios, luego con los secundarios y poco a poco se irá trabajando con los demás colores.
4. **Objetivo del juego visualmente explícito.** Los pasos necesarios para lograr el objetivo del juego deben ser inmediatos e intuitivamente obvios en cada pantalla. El paso fundamental de la acción debe ser el elemento más llamativo e interactivo.
5. **Distinguir los elementos interactivos.** Los elementos interactivos (por ejemplo: botones, objetos del juego) deben ser visualmente distintos (por ejemplo: color, grosor de línea, estilo de arte) del resto de la pantalla. Se debe considerar el uso sutil de animación, así como la intensidad del grosor de línea y color diferente.
6. **Consistencia.** El mismo comando o acción debe tener siempre el mismo efecto.
7. **Organización y continuidad.** Los conceptos y funciones similares deben ser lucir similares y pueden agruparse.
8. **Indicador “más”.** Si hay elementos ocultos de una lista de objetos, incluir un indicador visual (como flechas, una vista parcial del próximo ítem, o una animación) para indicar que hay más elementos para explorar.

9. **Desplazamiento.** El desplazamiento vertical por debajo de la página es conceptualmente difícil para los niños. El desplazamiento horizontal es más intuitivo. Se sugiere que todos los elementos interactivos importantes deben estar en la pantalla en la carga inicial para apoyar mejor al niño.
10. **Color resaltado.** El realce del color debe ser considerado al principio del proceso de diseño. Debe ser distinto de otros colores en la actividad. Sin embargo, es bueno usar más de un color en la misma aplicación sobre la base de su paleta de colores.
11. **Puntos “calientes”.** La motricidad fina de los niños pequeños está en desarrollo. Los puntos “calientes” (puntos importantes) deben ser grandes y adecuadamente aislados unos de otros. Se aconseja ayudar al niño a identificar los puntos calientes mediante la utilización de un “resplandor” o “chispa”, como un tiempo de espera después de las instrucciones.

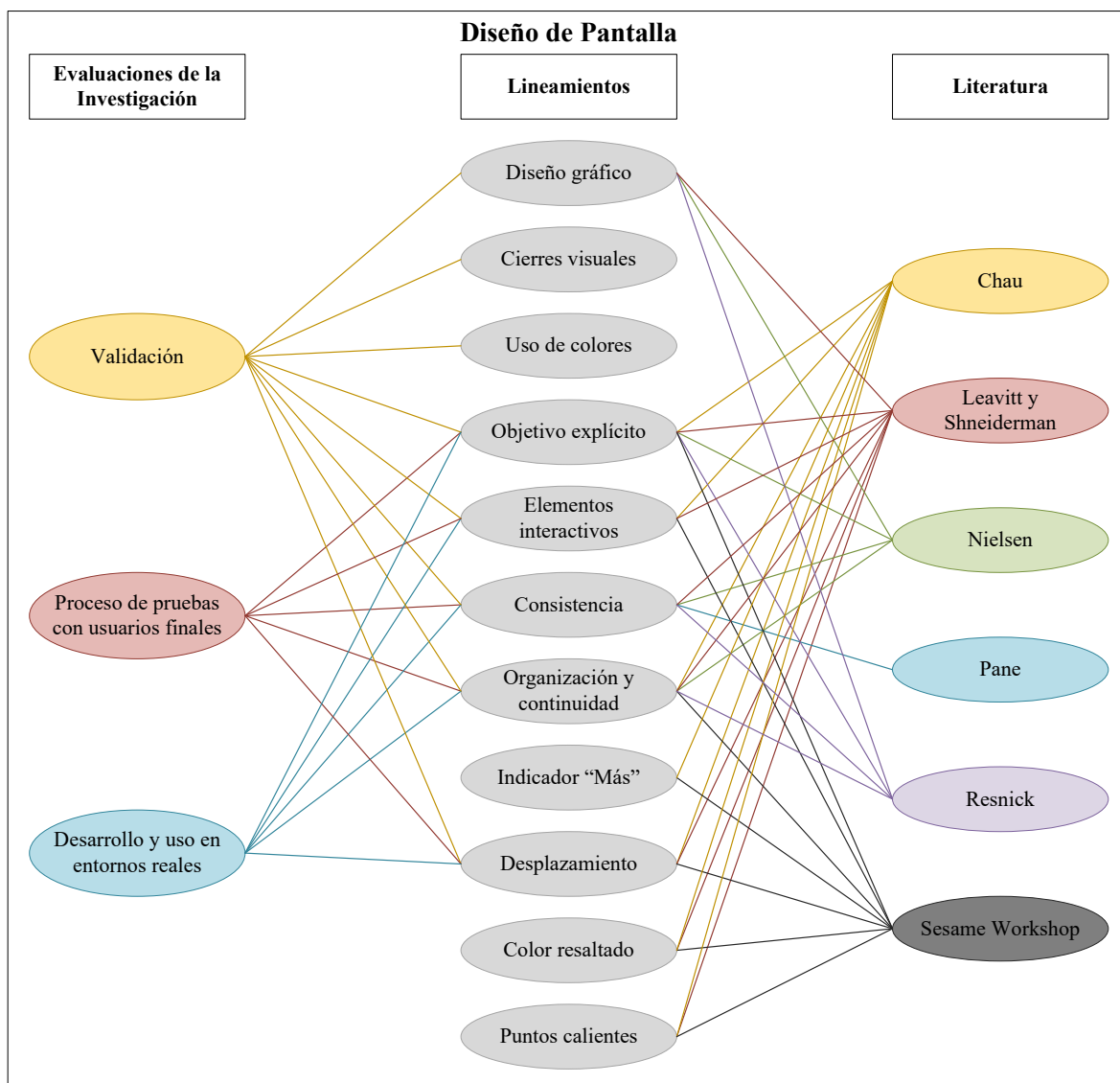


Figura 27. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño de Pantalla”.

Texto en la Pantalla

1. **Instrucciones textuales.** La mayoría de los niños de preescolar no pueden leer, por lo que se debe diseñar experiencias que pueden usar sin ayuda de un adulto. Dado esto, las instrucciones se deben comunicar visual y verbalmente con una narrativa amigable y simple.
2. **Diálogo sencillo y natural.** Las interfaces de usuario para niños de preescolar deben simplificarse, y deben coincidir con la tarea del niño de una forma tan natural como

sea posible, de manera que el mapeo entre los conceptos de la aplicación y los conceptos del niño se vuelva sencillo.

3. **Hablar el idioma del usuario.** La terminología de las interfaces de usuario debe basarse en el lenguaje del niño, ser una narrativa amigable, y se debe usar palabras familiares para los niños, en lugar de usar términos orientados al sistema o colocar significados que no son estándar.
4. **Tipos de letra orientadas a niños.** Las fuentes del texto orientadas a niños deben ser lo más cerca posible del tipo de letra *Zaner-Bloser* (excepto cuando el texto es una parte integrada de los gráficos en general). Se recomienda usar aproximaciones de *Zaner-Bloser*, como: *Sophie Joy* y *Furball*. Las fuentes no deben incluir *serifs*¹⁴.
5. **Contadores en pantalla.** Cuando hay elementos para contar, se recomienda que los números aparezcan en pantalla para reforzar el reconocimiento numeral.

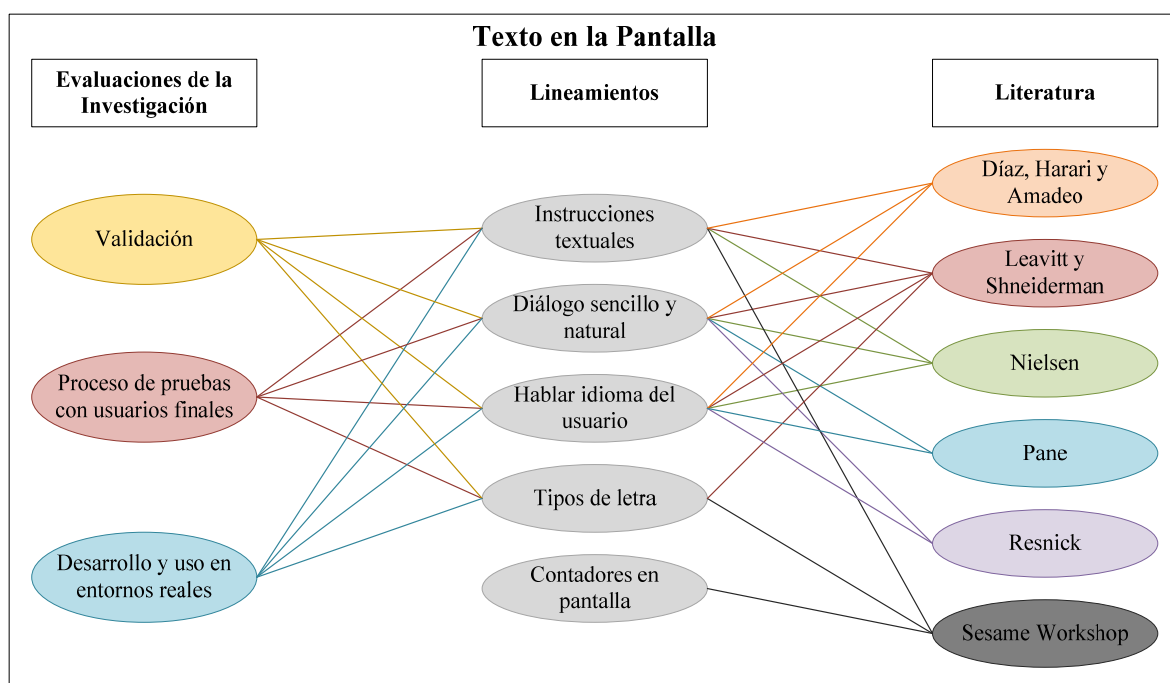


Figura 28. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Texto en la Pantalla”.

¹⁴ Las gracias, serifs (del inglés *serif*), remates o terminales son pequeños adornos ubicados generalmente en los extremos de las líneas de los caracteres tipográficos.

Diseño Visual

1. **Aplicaciones de aprendizaje.** Al diseñar aplicaciones de aprendizaje, las interfaces de usuario deben ser educativas y divertidas.
2. **Visualmente agradable.** El diseño de las interfaces de usuario dirigidas a niños en edad preescolar debe tener buenas gráficas y debe ser entretenido, ya que los factores emocionales, como el gusto visual y la diversión, juegan un papel crucial en el diseño de aplicaciones usables para niños, motivándolos a la interacción y en la consecución del objetivo, sin sobrecargar el diseño.
3. **Instrucciones visuales.** Por lo general los niños no prestan atención a instrucciones exclusivamente de audio. Las instrucciones son más beneficiosas si el audio se acompaña de componentes visuales, por ejemplo: si el personaje indica a un niño mover algo de un lugar a otro, este camino debe ponerse momentáneamente en relieve o indicarlo visualmente de alguna otra manera.
4. **Indicación de interactividad.** Se debe emplear señales explícitas en la pantalla para indicar cuándo el niño puede interactuar. Cuando se da la transición de una experiencia lineal a una experiencia interactiva, es importante reforzar las transiciones a través de los sonidos y efectos visuales.
5. **Llamatividad.** Los ítems que son “calientes” necesitan verse activos y tocables, por ejemplo: tener un fuerte resalte visual (normalmente amarillo) detrás de un ícono activo. Esto ayuda a los niños a saber que es interactivo. Los objetos deberían fijarse solo cuando se deben tocar. También se recomienda animar objetos interactivos.
6. **Iconografía y botones.** Se sugiere el uso de íconos consistentes y representativos que sigan la convención estándar.

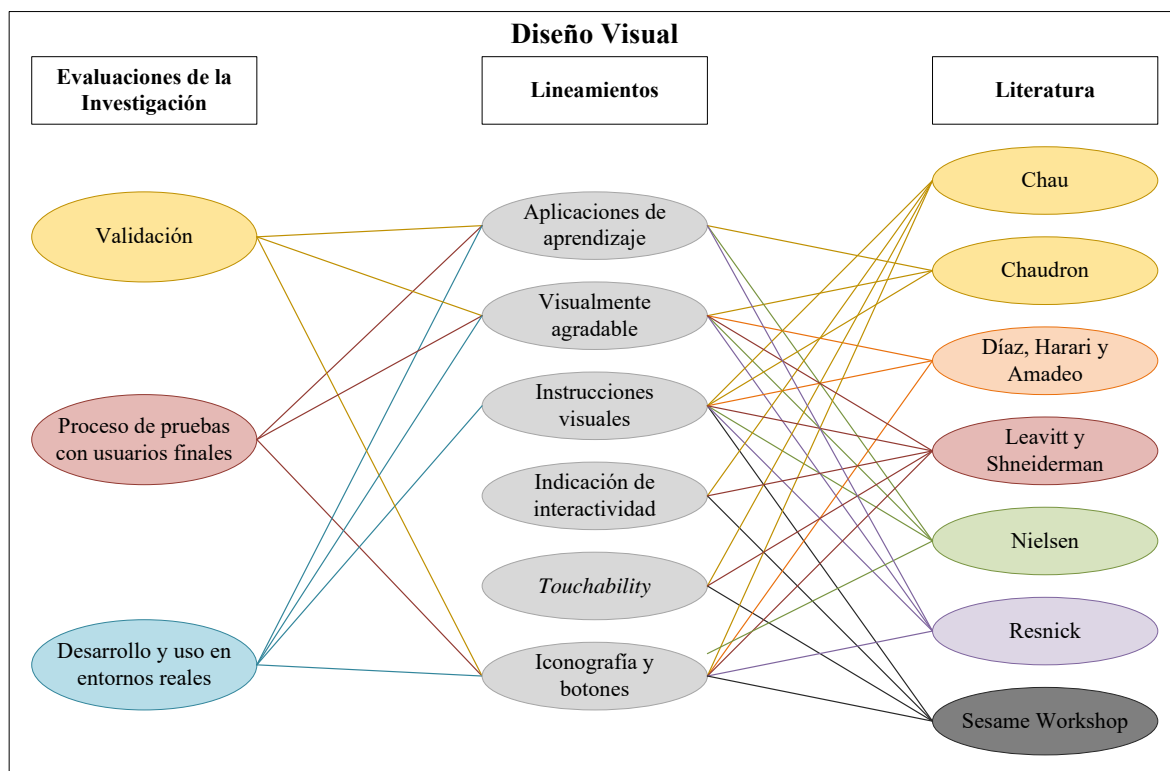


Figura 29. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño Visual”.

Composición Visual

1. **Menú.** En la mayoría de los casos, el diseño debe tener un menú siempre accesible.
2. **Orientación.** Los niños en edad preescolar tienden a mantener la tableta en posición horizontal, por lo que se recomienda crear aplicaciones con este tipo de orientación.
3. **Escaneo.** Los niños tienden a escanear las pantallas de la forma en la que leen. Para el público meta, esto generalmente significa de izquierda a derecha, y de arriba a abajo.
4. **Bordes de la pantalla.** Debido al peso y tamaño de las tabletas, los niños tienden a descansar sus muñecas a lo largo del borde final de la pantalla. Si los íconos activos se colocan allí, los niños son propensos a tocarlos accidentalmente, con lo que podrían sacar la aplicación de la actividad. Debido a esto, se debe realizar un posicionamiento estratégico de los íconos en la pantalla con el fin de minimizar la frustración y la fatiga.

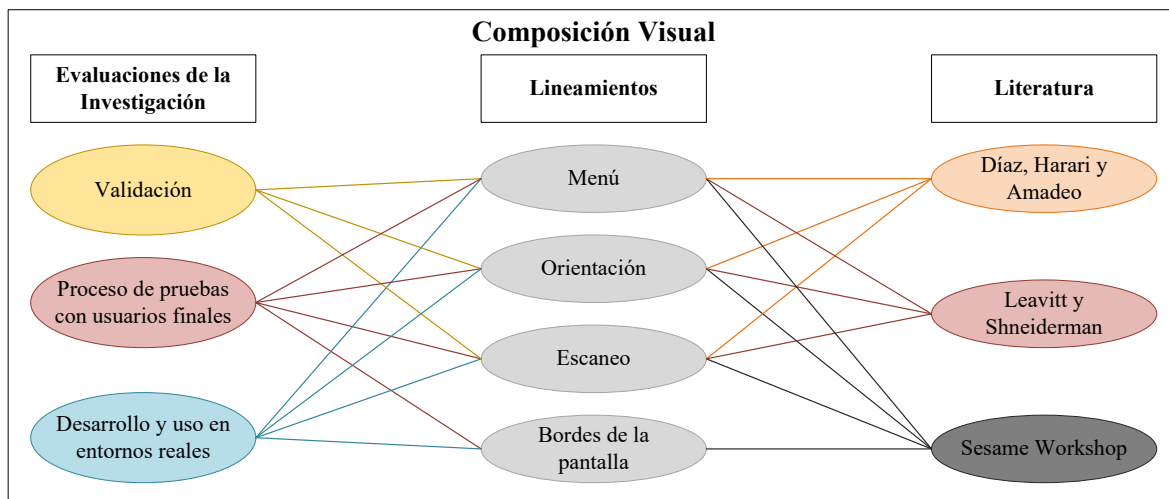


Figura 30. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Composición Visual”.

Diseño de Audio

1. **Conciso.** Los niños por lo general no prestan atención a las instrucciones de audio exclusivamente, por lo que se debe tener el audio esencial (por ejemplo: tareas o instrucciones específicas), mínimo y preciso. Combinar el sonido principal con cualquier soporte visual posible (por ejemplo: resaltar íconos y botones).
2. **Terminología de interacción.** La terminología de la pantalla táctil (tocar, deslizar, apretar) debe ser explícita para los niños. Por ejemplo: para instruir a los niños sobre la manera arrastrar y soltar, es útil dar instrucciones específicas, tales como: “Toque el comando con el dedo y arrástrelo a la pantalla”.
3. **Interrupciones.** Los niños a menudo se vuelven impacientes con los mensajes de audio continuos. Se sugiere hacer interrupciones en las indicaciones no esenciales, especialmente en la repetición.
4. **Efectos de sonido.** Los niños esperan una respuesta inmediata a su toque. Los efectos de sonido son una forma eficaz de comunicar el registro de entrada. Cuando sea posible, las acciones del niño deben ser reconocidas con un sonido.
5. **Indicación auditiva de interactividad.** Se sugiere el uso de un efecto de sonido compatible o un cambio en la música de fondo para indicar claramente la transición de una experiencia lineal a una experiencia interactiva.

6. **Música de fondo.** La música de fondo puede mejorar el compromiso de un niño. Sin embargo, se debe tener cuidado de no perjudicar el modo actual de la aplicación. Hay que elegir el estilo con cuidado y vigilar el volumen de la música de fondo.

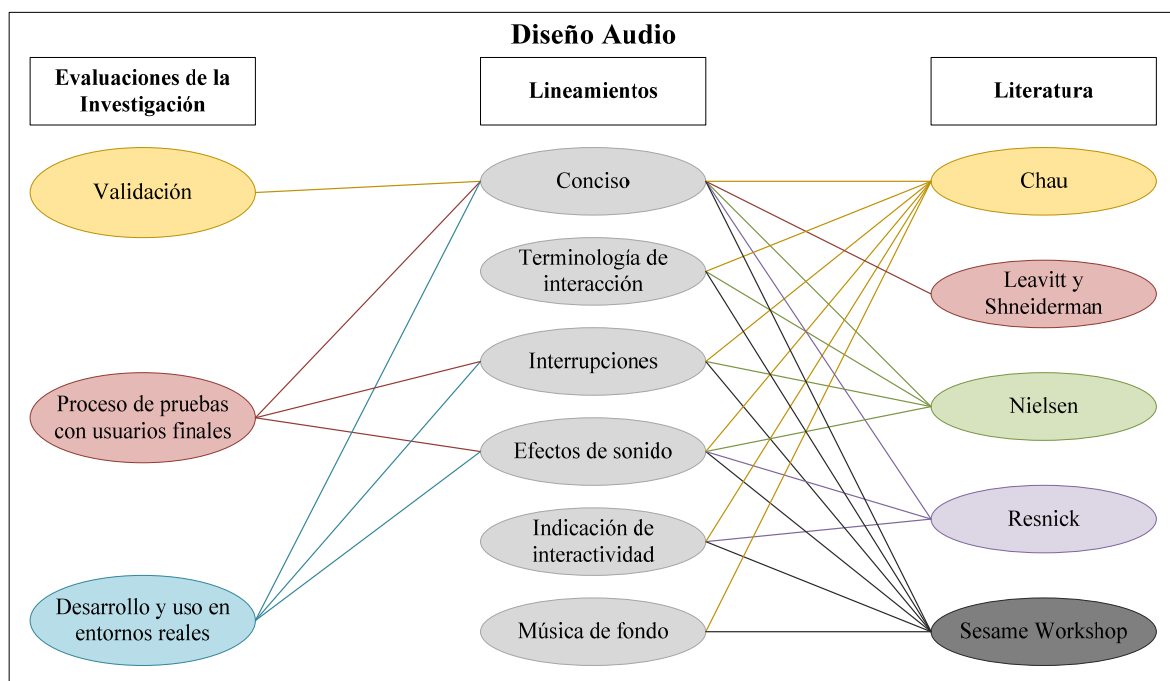


Figura 31. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Diseño de Audio”.

Intencionalidad

1. **Registro de entrada táctil.** Se recomienda el registro de entrada de programación táctil porque los niños tienden a tocar la pantalla demasiado duro, por mucho tiempo, o varias veces hasta que vean evidencia de que su interacción se ha registrado.
2. **Involuntario multitáctil.** Al desarrollar para dispositivos táctiles, se debe estar preparado para enfrentar problemas con el modo multitáctil, ya que los niños están desarrollando sus habilidades motoras finas. Se sabe por otras investigaciones que existen grandes diferencias de la destreza entre niños de 4, 5 y 6 años.

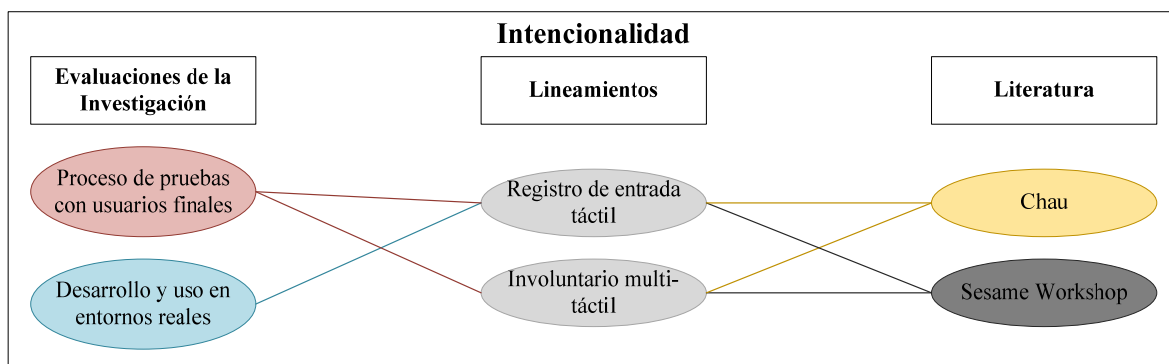


Figura 32. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Intencionalidad”.

Otras Notas de Diseño

1. **Tiempo de carga.** Con los niños solo se tiene corto tiempo para captar y mantener su atención. Por lo que, las diferentes componentes de la aplicación deben cargar rápidamente.
2. **Reducción al mínimo de la carga de memoria del usuario.** La aplicación debe asumir el peso de la memoria del niño, y ser simple.
3. **Registro e inicio de sesión.** Se supone que los niños más pequeños serán asistidos por un adulto durante cualquier proceso de registro e inicio de sesión. Sin embargo, se recomienda que los inicios de sesión tengan un diseño simple e intuitivo para que los niños puedan reconocer su propio perfil (tales como su nombre y un ícono exclusivo). En todos los casos, se sugiere mantener registro de la cuenta y los procesos de inicio de sesión lo más simples posible.
4. **Consejos para los padres.** Se debe incluir una sección de consejos para padres cuando sea apropiado. Estos consejos deben reforzar el plan de estudios, de ser posible, a través de sugerencias sobre la manera de fomentar la interacción entre padres e hijos durante la experiencia digital, así como ideas para la ampliación de los objetivos generales de aprendizaje.

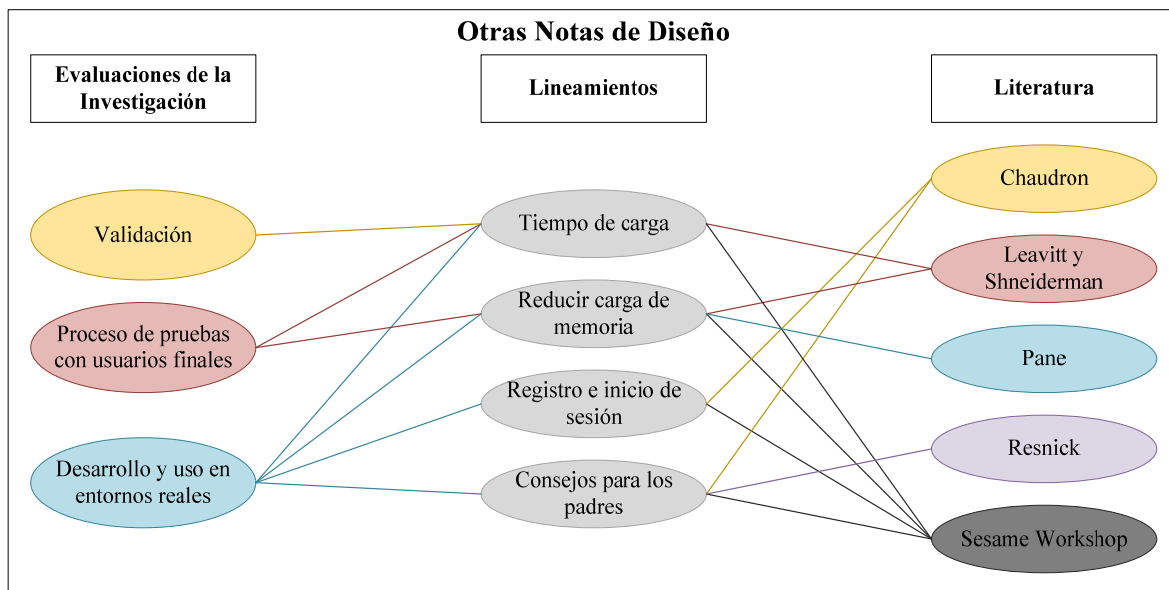


Figura 33. Diagrama de la relación entre la literatura consultada y las evaluaciones de la investigación donde se obtuvieron los lineamientos de la categoría “Otras Notas de Diseño”.

Por otra parte, en el diseño de sistemas de programación, varios autores recomiendan tener en cuenta los criterios de evaluación más específicos en el marco de trabajo *Cognitive Dimensions of Notations* (*Cognitive Dimensions*, para abreviar) (Alan F. Blackwell & Green, 2000; Green & Petre, 1996):

- **Viscosidad.** El sistema no debe resistirse a los cambios, no debe requerir muchas acciones del usuario para lograr una meta simple.
- **Visibilidad.** La información que necesita el programador en un momento determinado debe ser visible o de fácil acceso.
- **Compromiso prematuro.** El sistema no debe obligar al usuario a trabajar en un orden particular, o tomar una decisión antes de que la información necesaria está disponible.
- **Dependencias ocultas.** Los enlaces entre las entidades deben ser visibles.
- **Expresividad de roles.** El propósito de la entidad debe ser evidente.
- **Predisposición al error.** La aplicación debe protegerse contra descuidos y errores.
- **Cercanía de mapeo.** Las operaciones del sistema deben ser iguales a la forma en que los usuarios piensan acerca de las soluciones de problemas.

- **Notación secundaria.** El sistema debe permitir al programador comunicar información adicional mediante comentarios, tipografía y diseño, entre otros.
- **Evaluación progresiva.** El sistema debe permitir a los usuarios probar programas parciales.
- **Difuso.** Metas simples no deben requerir soluciones extraordinariamente largas o de grandes cantidades de espacio en la pantalla.
- **Provisionalidad.** El sistema debe permitir al usuario esbozar partes inciertas de su solución.
- **Operaciones mentales difíciles.** Ninguna de las operaciones del sistema debe requerir un gran esfuerzo mental para usarla.
- **Consistencia.** Notaciones similares deben significar cosas similares y viceversa.
- **Administración de la abstracción.** El sistema debe proporcionar una manera de definir nuevas facilidades o términos que permiten al usuario expresar las ideas con mayor claridad o de manera concisa, pero no debe obligar a los usuarios a usar esta capacidad desde el principio.

En la creación de TITIBOTS (ambas herramientas) se siguió la guía de lineamientos presentada. La mayoría de los lineamientos se implementaron en las herramientas. Los lineamientos no implementados, pero que serán implementados en un trabajo futuro son: consejos para los padres, la ayuda y la documentación dirigida a los niños, registro e inicio de sesión, incorporar más ayuda y retroalimentación visual que acompañe a la de audio. Asimismo, se cumplieron los criterios del marco de trabajo *Cognitive Dimensions of Notations*.

Por otra parte, los gestos utilizados en TITIBOTS para la interacción forman parte de los recomendados en los lineamientos (los más intuitivos) (Sesame Workshop, 2012):

- **Dar clic (*tap*).** Este gesto es la interacción táctil más intuitiva y fundamental para los niños. Se utilizó para los botones e íconos que representan alguna acción de la aplicación.

- **Arrastrar y soltar (*drag and drop*)**. Los niños pueden tocar un objeto y arrastrarlo a una nueva ubicación, pero a veces tienen dificultad con la continuidad con los dedos en la pantalla. Por lo que, se apoya la terminación parcial. Se utilizó para los comandos de programación.

En relación a la automatización de las actividades del aprendizaje colaborativo, se recomienda como lineamiento automatizar aquellas actividades que ayuden a la docente en el desarrollo de la teoría educativa, mediando la interacción y la comunicación entre los participantes sin obstaculizar. En la Sección 6.4 se detalla más al respecto.

En las secciones siguientes se describen las herramientas de programación desarrolladas en la investigación.

6.3 TITIBOTS – Herramienta Individual de Programación

TITIBOTS es una herramienta de programación desarrollada para los niños de la primera infancia que les permite crear un programa y enviarlo a un robot con el fin de ejecutarlo. Los niños de entre 4 y 6 años son los usuarios meta. La herramienta funciona en dispositivos móviles Android y ha sido utilizada con el kit robótico LEGO Mindstorms NXT y robots que usan tecnología Arduino.

El sistema creado tiene una interfaz de usuario simple constituida por símbolos iconográficos y sonidos, que toma en cuenta el desarrollo cognitivo, personal, social y emocional de los niños, permitiendo ser usada por aquellos que aún no han aprendido a leer y a escribir. El diseño está pensado para ser intuitivo y usable por ellos. El uso de la herramienta es individual, cada niño tiene una tableta y un robot.

Dado esto, se plantea que los niños puedan resolver problemas previamente definidos, y que, usando la herramienta con mediación docente, programen las soluciones, donde cada problema tiene asociado uno o más robots previamente construidos. Por tanto, se quiere

lograr que los niños creen un conjunto de instrucciones ordenadas (programa) para resolver un problema específico, y puedan ver al robot ejecutando estas instrucciones.

Cada sistema operativo móvil (SOM) ofrece una versión diferente en la experiencia de usuario, y ningún SOM cumple con los requisitos de cada usuario potencial; por lo que elegir el SOM correcto depende de las necesidades del usuario o proyecto en desarrollo. Para esta investigación, el SOM Android es la mejor opción, ya que es una plataforma versátil que satisface las necesidades de prácticamente cualquier usuario. Además, Android cuenta con una gran variedad de dispositivos móviles de diferentes empresas y a diferentes precios, con una variedad mucho más amplia de opciones de personalización, y una mayor libertad de ajustar el dispositivo si se decide hacerlo. Específicamente, se utilizó el SO Android Jelly Bean (v4.1.2), es uno de los SOM más recientes con las características necesarias para el buen funcionamiento del sistema y cumple con las especificaciones de los expertos. Además, esta versión cuenta con la funcionalidad de texto a voz (TextToSpeech, TTS).

Con respecto al tipo de dispositivo móvil, se usaron tabletas de 7 u 8 pulgadas, siguiendo las recomendaciones de los expertos de la FOD y los docentes de preescolar, ya que ese tamaño es apropiado para las manos de los niños y hay espacio suficiente para mostrar diferentes elementos de la interfaz con un tamaño adecuado.

En la Figura 34 se muestra la línea de tiempo relacionada a la evolución de la herramienta de programación TITIBOTS, según los cambios sufridos por las evaluaciones de usabilidad realizados. En la Figura 35 se muestra el diseño final de la interfaz de TITIBOTS.

En la Figura 34 se aprecia que el primer prototipo de la herramienta se desarrolló en marzo del 2013. El objetivo de este primer prototipo fue comprobar la factibilidad técnica (comunicación entre la tableta y el robot NXT mediante Bluetooth). Una vez que se comprobó la factibilidad técnica, se procedió a trabajar en su diseño gráfico. Durante el 2014, se realizaron varias evaluaciones para validar el diseño gráfico y la usabilidad de la herramienta. Finalmente, en enero del 2015, se tuvo el primer prototipo con un diseño gráfico validado por expertos y niños de edad preescolar, intuitivo y fácil de usar.



Figura 34. Evolución de TITIBOTS.

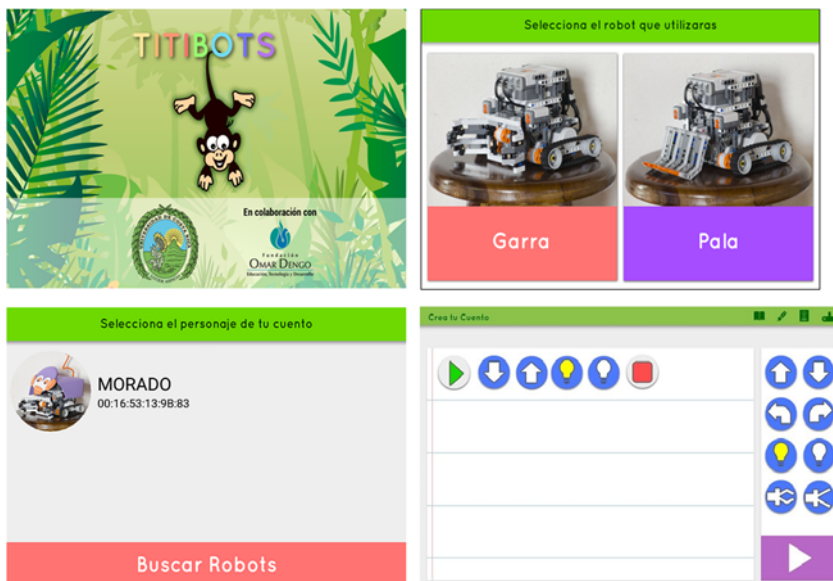


Figura 35. Herramienta de programación TITIBOTS.

En las secciones siguientes se describe la arquitectura y los diferentes componentes que posee TITIBOTS.

6.3.1 Arquitectura

La herramienta de programación TITIBOTS incluye comandos para que el robot pueda moverse de un lugar a otro y manipular objetos en el medio ambiente y comandos de control para señalar el comienzo y el final de la secuencia de instrucciones que se programe. La herramienta usa los robots para apoyar el proceso de aprendizaje de la programación, ya que la robótica es una buena herramienta de aprendizaje para niños pequeños, porque los robots les permiten a los niños tener un aprendizaje partiendo desde lo concreto hacia lo abstracto, de forma lúdica.

En cuanto a los conceptos básicos de programación que se abordan con el uso de la herramienta están:

- **Algoritmo.** Un algoritmo es una serie de instrucciones (pasos) para que se lleve a cabo una tarea. Es importante enfatizar que los algoritmos son el núcleo de la

programación. Así, cada programa es, simplemente, una lista de instrucciones que la computadora debe seguir en un orden determinado.

- **Secuencialización.** Se quiere lograr que los niños creen un conjunto de pasos (instrucciones) que se deben seguir, en un orden determinado, para resolver un problema específico.

Entre los conceptos de robótica que se abordan están (Ruíz del Solar & Salazar, 2000):

- **Efectores y actuadores.** Los efectores y actuadores son componentes de un robot. Un efector corresponde a cualquier dispositivo que afecte o modifique el ambiente. Un actuador es cualquier mecanismo que permita al efector ejecutar una acción, por ejemplo: los servos y las luces.
- **Locomoción y manipulación.** Los robots pueden tener un sistema de locomoción o de manipulación. El sistema de locomoción permite que el robot se mueva por el ambiente, mientras que el sistema de manipulación permite que el robot pueda mover o alcanzar objetos del ambiente que estén cerca de él.

Con el fin de evaluar la herramienta, se diseñó y construyó un robot con LEGO Mindstorm NXT. El robot es capaz de moverse hacia adelante y atrás, girar a la izquierda y derecha, encender y apagar luces, y tiene una garra que le permite agarrar y soltar objetos.

Los expertos de la FOD proporcionaron un conjunto de requisitos funcionales y no funcionales en un proceso de diseño participativo. La base de los requisitos de la herramienta es la siguiente:

- Usar una metáfora que permita una comprensión clara para los niños.
- Proporcionar un conjunto 8 a 10 de comandos que el robot pueda ejecutar.
- Implementar estructuras de control (Louden & Lambert, 2011, pp. 422-423).
- Tener actuadores: servos y luces, por lo menos.
- Guardar el programa creado.
- Conectar a través de Bluetooth con el robot.
- Permitir la capacidad de texto a voz (TTS).

Además, los docentes de preescolar definieron varias tareas a ser programadas por los niños. La mayor parte de las tareas son secuencias de movimientos (es decir, mover el robot de un lugar a otro o mover objetos).

La arquitectura general de TITIBOTS se muestra en la Figura 36. Hay dos componentes principales:

1. Una aplicación Android se ejecuta en la tableta y muestra una interfaz simple con los comandos y el espacio de trabajo de la herramienta de programación. Los comandos (protocolo de comunicación) creados son:
 - **Control:** inicio y fin.
 - **Movimiento (locomoción):** adelante, atrás, izquierda y derecha.
 - **Acción (manipulación):** encender luces, apagar luces, agarrar objeto y soltar objeto.
2. Un programa en leJOS NXJ (*LEGO Java System Operative NXJ*) creado para el ladrillo inteligente de LEGO Mindstorms NXT (v2.0). Este es el programa responsable de interpretar los comandos enviados a través de la conexión Bluetooth desde el dispositivo móvil al ladrillo NXT.



Figura 36. Arquitectura de TITIBOTS.

Se implementó, además, una herramienta asistente, fácil de usar, que permite la configuración de los tipos de robot a usar, la conexión entre el robot y la aplicación móvil. El docente puede configurar el sistema y añadir o eliminar robots. Una vez que la tableta está conectada con el robot, el niño puede crear un programa arrastrando y soltando los comandos disponibles, y, por último, él lo puede enviar al robot para que se ejecute.

El docente es el encargado de especificar los comandos disponibles para cada sesión de trabajo, ya que la herramienta cuenta con un enfoque por nivel, lo que permite que el docente utilice un gesto para desbloquear nuevos comandos (subir un nivel). Una vez que el docente piensa que el niño alcanza un cierto nivel de aprendizaje con los comandos disponibles, incluye un par de comandos más.

A continuación, se describe la interfaz móvil de TITIBOTS.

6.3.2 Interfaz Móvil de Usuario

La aplicación móvil de TITIBOTS se encarga de obtener las acciones y la información de los niños. El diagrama correspondiente a la interfaz de usuario móvil se muestra en la Figura 37.

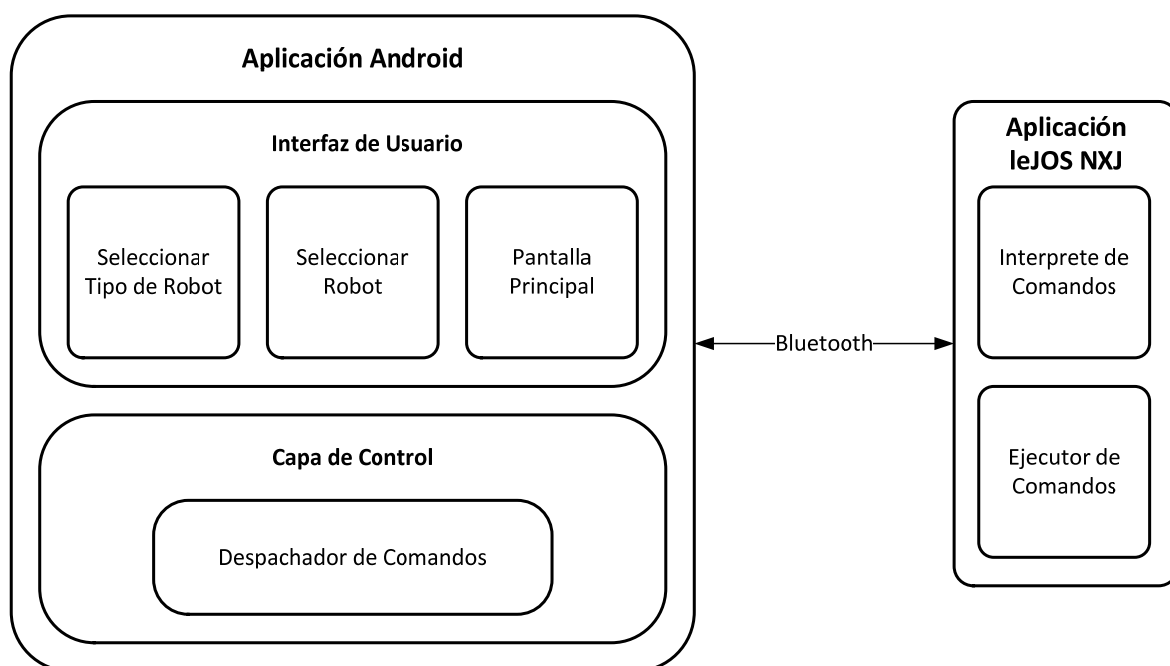


Figura 37. Diagrama con la arquitectura de TITIBOTS.

Cada niño con su dispositivo móvil debe pasar por una serie de pantallas. En primer lugar, en “Seleccionar Tipo de Robot” deben seleccionar el tipo de robot que van a usar. Las opciones son: robot móvil, robot móvil con garra o robot móvil con pala. Luego, en

“Seleccionar Robot” deben buscar los robots activos y elegir el que le fue asignado por la docente, mediante el descubrimiento de dispositivos bluetooth cercanos, y una vez que el o los robots aparecen en la lista, debe seleccionar el que le corresponda.

Ya con el robot seleccionado, la pantalla que aparece es la “Pantalla Principal” donde cada niño puede usar de la aplicación. La pantalla se compone principalmente de una metáfora de hoja de cuaderno con un tablero de comandos a la derecha. Así, el niño puede arrastrar los comandos desde el tablero y soltarlos en la hoja de cuaderno. La interfaz de usuario permite que los comandos sean agregados en cualquier orden, entre los comandos de inicio y fin (que se encuentran en la hoja de cuaderno previamente).

Con los comandos agregados a la hoja del cuaderno, el usuario puede enviar los comandos al robot previamente conectado. La capa de control (mediante el “Despachador de Comandos”) se encarga de validar los comandos (para evitar dañar el robot). Si encuentra algún problema en relación al orden o la cantidad de comandos que pueden dañar al robot (por ejemplo: enviar dos o más veces a cerrar la garra) lo soluciona, y en caso de no haber más problemas, envía los comandos al robot.

Por otra parte, en la aplicación del robot, el “Interprete de Comandos” se encarga de interpretar cada uno de los mensajes tipo entero¹⁵ que componen el programa enviado por el niño a la acción respectiva en el robot. Por último, el “Ejecutor de Comandos” ejecuta el programa conformado por el conjunto de acciones interpretadas.

A continuación, se describen los robots utilizados con la interfaz móvil de TITIBOTS.

¹⁵ Un tipo de dato entero en computación es un tipo de dato que puede representar un subconjunto finito de los números enteros.

6.3.3 Robots

Los robots utilizados fueron:

- Robots contruidos con el Kit Robótico LEGO Mindstorms NXT. Se crearon tres tipos (ver Figura 38a). Se utilizó el lenguaje de programación leJOS NXJ.
- Robot construido con el Robotics Shield Kit Parallax Arduino Uno (ver Figura 38b). Se utilizó el lenguaje de programación Arduino.

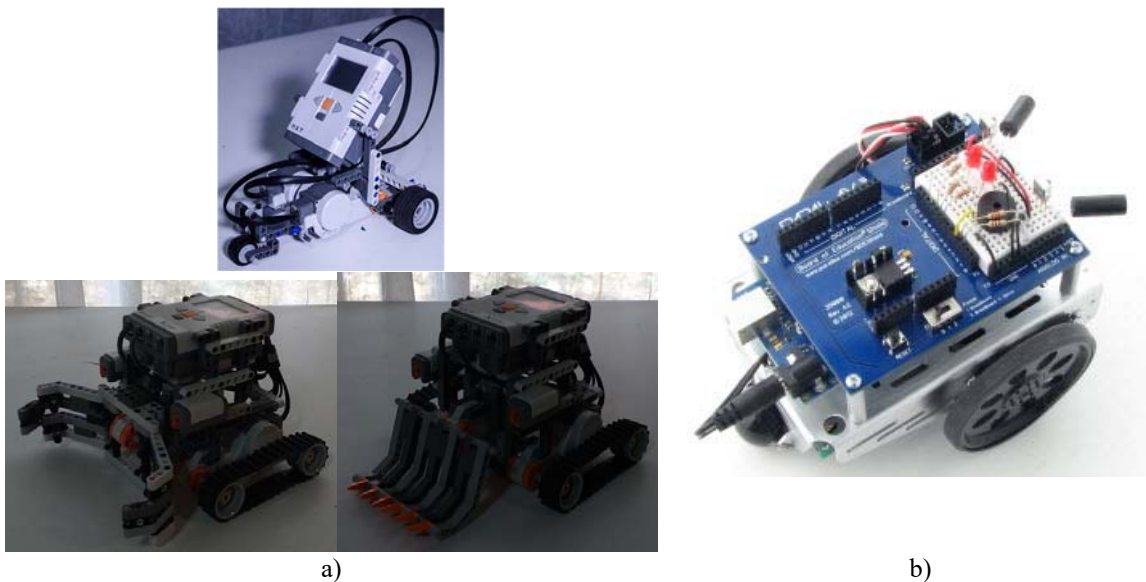












Figura 38. Robots utilizados en TITIBOTS.

A continuación, se describe el protocolo de comunicación de TITIBOTS.

6.3.4 Protocolo de Comunicación

El protocolo de comunicación es simplemente un conjunto abierto de comandos que permiten al robot realizar un conjunto de acciones. La comunicación entre la herramienta de programación y los robots es de forma inalámbrica usando Bluetooth (comunicación uno a uno). Por el canal de comunicación que se establece entre la tableta y el robot se envían mensajes tipo *int* (números enteros), donde cada uno corresponde a un comando de la herramienta. En la Tabla 11 se muestra el conjunto abierto de comandos de TITIBOTS.

Tabla 11. Conjunto abierto de comandos de TTIBOTS.

Tipo de Comando	Ícono	Código	Nombre	Descripción
Control		COMMAND_START = 1	Inicio	Comienzo de la secuencia de instrucciones
		COMMAND_END = 2	Fin	Final de la secuencia de instrucciones
Movimiento		COMMAND_FORWARD = 3	Adelante	Mueve hacia adelante el robot
		COMMAND_BACKWARD = 4	Atrás	Mueve hacia atrás el robot
		COMMAND_LEFT = 5	Izquierda	Gira a la izquierda el robot
		COMMAND_RIGHT = 6	Derecha	Gira a la derecha el robot
Acción		COMMAND_OPENED = 7	Soltar	Abre la garra, suelta un objeto
		COMMAND_CLOSED = 8	Agarrar	Cierra la garra, agarra un objeto
		COMMAND_ON = 9	Encender	Enciende las luces del robot
		COMMAND_OFF = 10	Apagar	Apaga las luces del robot

En la sección siguiente se describe la herramienta colaborativa de programación TITIBOTS Colab, sucesor de TITIBOTS.

6.4 TITIBOTS Colab – Herramienta Colaborativa de Programación

TITIBOTS Colab es una herramienta colaborativa de programación desarrollada para los niños de la primera infancia. Permite crear un programa de forma colaborativa y enviarlo a un robot para su ejecución. Los usuarios meta son los niños de entre 4 y 6 años. El proceso

de colaboración se propiciará mediante la distribución automática y equitativa de los recursos entre los miembros de cada grupo; o sea, la herramienta colaborativa distribuirá los comandos de programación disponibles de forma equitativa entre los estudiantes que conforman cada grupo. La herramienta es para dispositivos móviles Android y, ha sido utilizada con el kit robótico LEGO Mindstorms EV3 y robots que usan tecnología Arduino y Raspberry Pi.

El sistema propuesto tiene una interfaz de usuario simple constituida por símbolos iconográficos y sonidos, que toma en cuenta el desarrollo cognitivo, personal, social y emocional de los pequeños, permitiendo ser usada por niños que aún no han aprendido a leer y a escribir. El diseño está pensado para ser intuitivo y usable por los niños en un entorno mediado por docentes en el salón de clases.

Dado esto, se plantea que los niños puedan resolver problemas previamente definidos, y que, usando la herramienta con mediación docente, programen las soluciones en grupos colaborativos, que poseen un robot previamente construido. Por tanto, se quiere lograr que los niños creen un conjunto de instrucciones ordenadas (un programa) colaborando para resolver un problema específico, y puedan ver al robot ejecutando las instrucciones.

Al igual que en su antecesor TITIBOTS, el personaje es el mono tití (personaje virtual y físico: robot disfrazado) y el nombre lo hereda, pero con la incorporación de Colab por ser una herramienta colaborativa. En la Figura 39 se muestra la herramienta TITIBOTS Colab, con las diferentes pantallas del diseño gráfico y los robots que se usaron.

Dado que es una herramienta colaborativa, se automatizaron varias actividades del aprendizaje colaborativo. Cuando se automatiza una teoría de educación (como lo es el aprendizaje colaborativo), se debe determinar las actividades que se llevan a cabo en la teoría e identificar aquellas actividades que pueden ser automatizadas (tal como se discutió en la Sección 3.3.2 para el aprendizaje colaborativo). Se recomienda automatizar aquellas actividades que ayudan con el desarrollo del proceso a la docente, permitiendo la mediación de la interacción y la comunicación entre los participantes sin obstaculizarlas.

Posteriormente, del conjunto de actividades identificadas (ver Tabla 7), se seleccionaron aquellas que tenían relevancia en el contexto de uso, permitiendo que la tecnología apoye el proceso del aprendizaje colaborativo. Estas actividades se presentan en la Tabla 12 y se explican más adelante en las secciones 6.4.1 y 6.4.5.

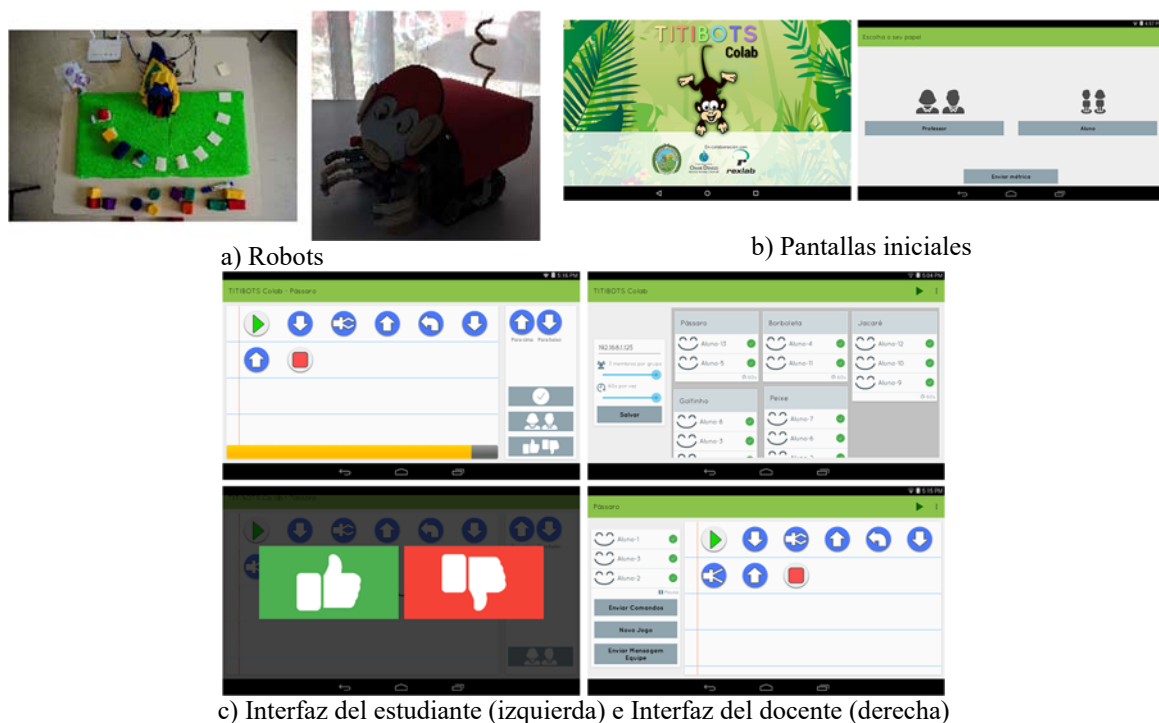


Figura 39. Herramienta colaborativa de programación TITIBOTS Colab.

Tabla 12. Actividades del proceso del AC que se automatizaron en TITIBOTS Colab.

Pre-Proceso	En-Proceso	Pos-Proceso
<ul style="list-style-type: none"> • Especificar el tamaño de los grupos • Organizar los grupos • Especificar los roles 	<ul style="list-style-type: none"> • Aplicar estrategias (interdependencia positiva de la meta, la motivación entre pares, la ayuda para aprender) • Organizar la cooperación intra-grupo • Probar los criterios de éxito • Monitorear a los grupos • Proporcionar ayuda (mediador y pares) • Verificar el reporte de los grupos • Proporcionar y fomentar la retroalimentación 	<ul style="list-style-type: none"> • Inspeccionar los criterios de éxito • Presentar el cierre de la actividad (en una pequeña parte)

Debido a la relevancia del contexto de uso, del conjunto de actividades automatizables del aprendizaje colaborativo (ver Tabla 7), la única que no se implementó fue “Definir los criterios de éxito”, ya que se consideró que los docentes debían tener la libertad de establecer los desafíos de programación que consideren pertinentes a contenidos relacionados al programa de preescolar según el nivel y el progreso de los niños.

En las secciones siguientes se describe la arquitectura y los diferentes componentes de TITIBOTS Colab.

6.4.1 Arquitectura

TITIBOTS Colab fue creado usando un diseño centrado en el usuario. Está constituido por dos módulos: Módulo del Estudiante y Módulo del Docente. Además, al ser una herramienta colaborativa, debe haber comunicación entre los dispositivos que se conecten. Para esto se implementó una arquitectura centralizada. La comunicación entre las tabletas y el robot se da a través de Wi-Fi Peer-to-Peer.

Al implementar aplicaciones en las que se deben conectar y comunicar varios dispositivos, es importante decidir cómo se van a conectar. La arquitectura distribuida de conexión entre los dispositivos es una buena opción, pero trae complicaciones en términos de cómo propagar la información:

- El mayor problema encontrado es cuando un dispositivo se desconecta de improvisto, ya que se requiere establecer todas las conexiones, no solo para el dispositivo desconectado, sino con todos los otros dispositivos conectados.
- Se debe decidir cuál dispositivo va a enviar los comandos al robot. Si el robot no tiene una función de detección, será poco práctico para la docente establecer la conectividad del robot con todos los dispositivos conectados.
- La cola de los conjuntos de comandos: cómo debe administrarse para decidir cuál conjunto de comandos debe ser ejecutado primero.

Estos tres problemas preocupaban a la hora de implementar la arquitectura distribuida, por lo que se decidió implementar la arquitectura centralizada. Aunque esto crea riesgo al tener un solo dispositivo que controla todos los otros, resuelve los problemas mencionados, por lo que la interacción de los niños es fácil de controlar por el mediador. Es posible reducir el riesgo de la implementación centralizada asegurándose de que otros clientes pueden manejar la desconexión y sean capaces de volver a conectarse, mientras que la aplicación docente debe ser capaz de restaurar el estado y volver a conectar todos los demás dispositivos.

La arquitectura centralizada desarrollada en TITIBOTS Colab (ver Figura 40) permite que la comunicación entre los dispositivos móviles inicie cuando la aplicación de la docente abre la conexión, permitiendo que los otros dispositivos (aplicaciones de los estudiantes) establezcan la comunicación con él. Además, la docente debe especificar en la herramienta la dirección IP del robot. Cuando se tienen todos los dispositivos conectados, la docente procede a crear los grupos según el tamaño seleccionado por ella. A partir de este momento, la herramienta crea de forma aleatoria los grupos y especifica los roles de cada miembro (y divide los comandos de la aplicación).

En el módulo del estudiante, dado que los niños de preescolar no saben leer, las instrucciones se comunican verbalmente con una narrativa amigable y simple, y visualmente mediante íconos representativos.

Con los grupos creados, en el módulo del estudiante, la herramienta notifica a los niños que la conexión se ha establecido (verbal y visualmente), y les solicita su sexo y edad. Luego de ingresados estos datos por los niños, les dice el grupo al cual forman parte.

Cuando los niños están listos en sus respectivos grupos, la docente, mediante su módulo, inicia el trabajo explicando el desafío a solucionar y cada grupo colaborativo empieza a trabajar para solucionarlo. En el proceso, los niños interactúan para solucionar el desafío y cada miembro del grupo puede colocar los comandos que la herramienta les asignó según la solución planificada por el grupo. Cada niño tiene un tiempo para trabajar asignado por la docente (por defecto está en 60 segundos). Al terminar este tiempo (turno), el niño envía el

trabajo realizado para que los otros miembros del grupo puedan verlo, y cede el turno a otro miembro. Si después de que pasa el tiempo asignado, algún niño no envía el trabajo realizado (o no está trabajando), la herramienta automáticamente lo envía y cede el turno a otro miembro del grupo. Durante este proceso, los niños se comunican para expresar sus ideas, coordinar sus acciones con los otros miembros y ofrecer ayuda a sus compañeros.

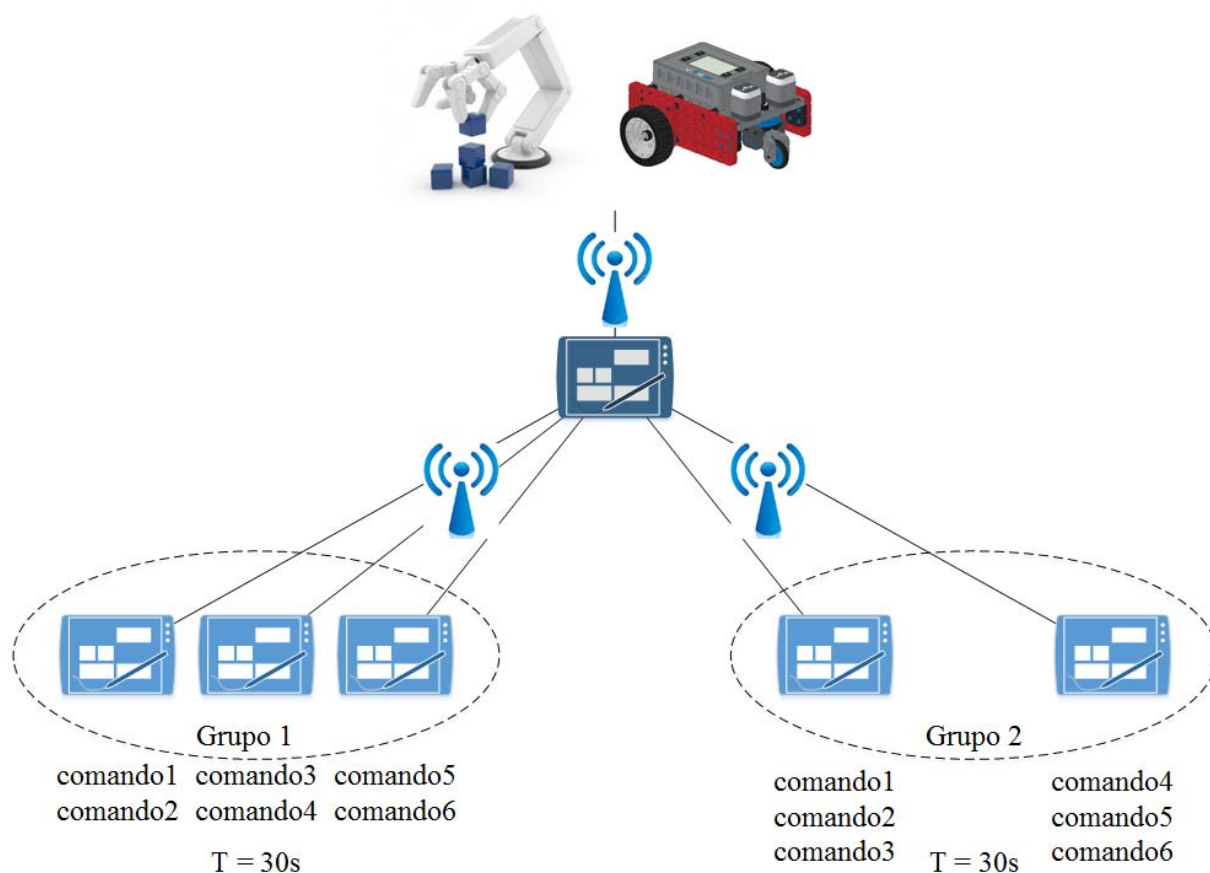


Figura 40. Arquitectura de TITIBOTS Colab.

Si en algún momento un miembro de algún grupo necesitara ayuda de la docente, puede solicitarla a través de la herramienta. En el módulo de la docente ella recibirá la solicitud y podrá atenderlo.

Mientras que los grupos aplican estrategias para crear un programa que solucione el desafío, arrastrando y soltando los comandos disponibles en el área de programación, la docente

puede en todo momento monitorear cada grupo y evaluar el trabajo realizado. De esta manera, la herramienta le permite darse cuenta del trabajo de cada grupo, si los niños trabajan o no, las solicitudes de ayuda y retroalimentar los grupos, entre otros.

Cuando algún niño piensa que la solución que realizó su grupo está terminada, crea una votación. Si los otros miembros están de acuerdo, el programa se envía al módulo de la docente y la herramienta se encarga de notificar que el grupo ha terminado. La docente es la responsable de enviar el programa creado por los grupos de colaboración para que el robot lo ejecute. La docente revisa el trabajo realizado, y si detecta un error, puede decidir si envía de nuevo el programa al grupo con un mensaje diciendo que verifiquen el programa, o si lo envía a ejecutar al robot para que el grupo observe y aprenda del error.

Finalmente, para cerrar la actividad y pasar a otro desafío, la docente puede enviar un mensaje de felicitación a cada grupo por su logro.

A continuación, se describe la interfaz móvil de la docente de TITIBOTS Colab.

6.4.2 Interfaz Móvil de Usuario de la Docente

La aplicación de la docente (mediador) permite a la docente organizar y monitorear los grupos de estudiantes que usan la interfaz de usuario “estudiante”. En la Figura 41 se muestra el diagrama de la interfaz móvil de usuario de la docente.

En la fase de preparación, la docente debe iniciar el servicio de emisión (*broadcast*) que envía la señalización correspondiente en la red, para que los dispositivos de los estudiantes lo puedan descubrir. Además, en esta pantalla puede ingresar la dirección IP, puerto del robot e indicar el tamaño de los grupos (2 o 3 integrantes) y la duración del turno (30, 45 o 60 segundos). Cuando todos los estudiantes se han conectado, la docente puede iniciar la organización de los grupos. La aplicación acomoda a los estudiantes en grupos, evitando que queden estudiantes sin grupo. Los grupos reciben además un nombre y una figura de un animal que los identifica.

Módulo Docente (Mediador)

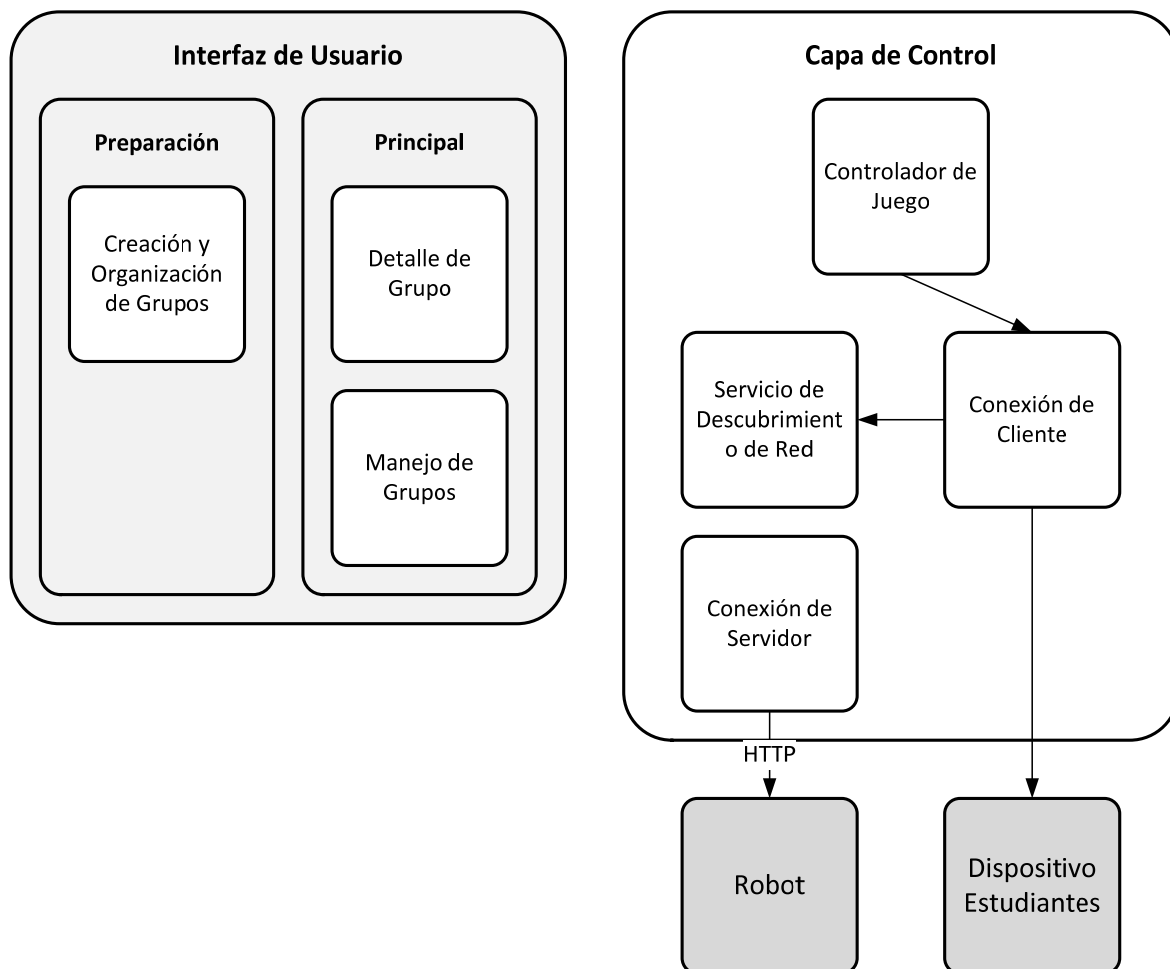


Figura 41. Diagrama de TITIBOTS Colab – Interfaz de Usuario de la Docente.

Una vez que los grupos han sido definidos, la docente da inicio a la sesión y permite que los estudiantes inicien el ejercicio. La pantalla cambia y puede ver un bloque por grupo (“Manejo de Grupos”). Este bloque contiene información básica sobre el estudiante que tiene el turno y el tiempo que le queda, si a alguno le ha expirado el turno o, inclusive, si el juego se ha pausado debido a una solicitud de votación o porque algún dispositivo se ha desconectado. También, cambia el color de aquel dispositivo cuyo estudiante ha solicitado asistencia a la docente. Este indicador puede ser desactivado por la docente una vez que considere que la asistencia ha sido brindada.

Si la docente selecciona un grupo en particular, la pantalla cambia a “Detalle de Grupo” donde puede monitorear qué comandos han sido agregados al área de programación, así como la información que se desplegaba anteriormente.

En caso de que los estudiantes hayan solicitado una votación y haya sido aprobada por todos los miembros, la docente puede revisar la solución propuesta e indicarles si es correcta o no. En caso contrario, la docente puede reanudar el juego para que los estudiantes revisen y corrijan, o envía al robot el programa (para que el grupo se dé cuenta del error al observar al robot ejecutando las instrucciones).

Cuando un grupo finaliza con éxito un desafío, la docente envía a limpiar el área de programación del grupo para que empiecen un nuevo desafío.

Al finalizar la sesión, la docente puede darla por terminada desconectando a todos los dispositivos y volviendo a la pantalla de preparación.

A continuación, se describe la interfaz móvil del estudiante de TITIBOTS Colab.

6.4.3 Interfaz Móvil de Usuario del Estudiante

La aplicación del estudiante permite a los estudiantes crear programas para el robot. En la Figura 42 se muestra el diagrama de la interfaz de usuario móvil del estudiante.

En primer lugar, solicita al usuario su sexo y edad, con el fin de mantener información estadística sobre el uso de la aplicación (ver Figura 43). El segundo paso es buscar el dispositivo de la docente. La aplicación busca una red disponible y al encontrarla se conecta automáticamente. Al hacerlo queda en modo de espera hasta que todos los demás usuarios estén conectados y listos para empezar.

Módulo Estudiante

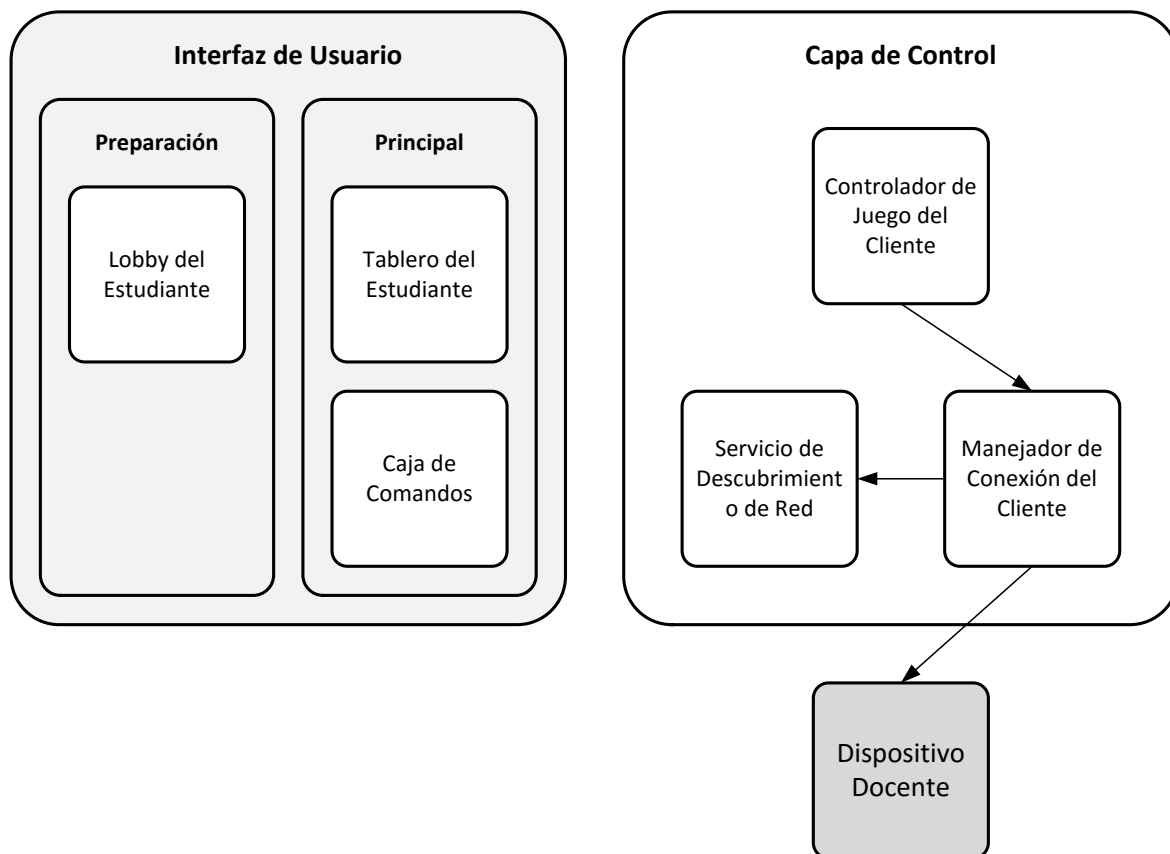
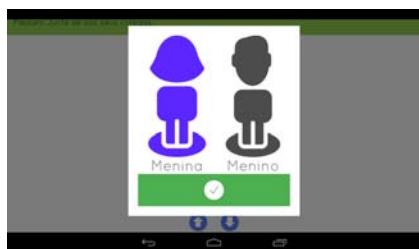
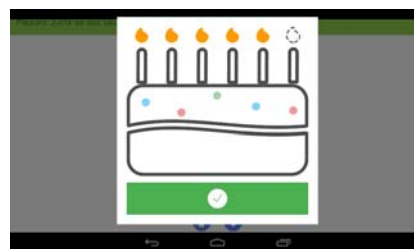


Figura 42. Diagrama de TITIBOTS Colab – Interfaz de Usuario del Estudiante.



a) Solicita el sexo del estudiante

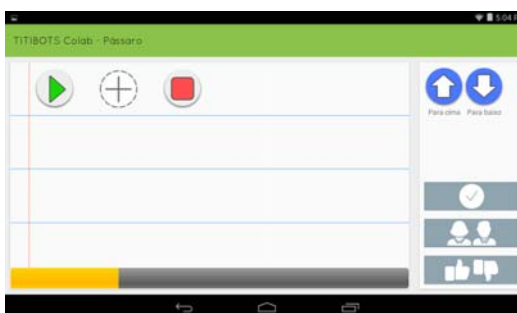


b) Solicita la edad del estudiante

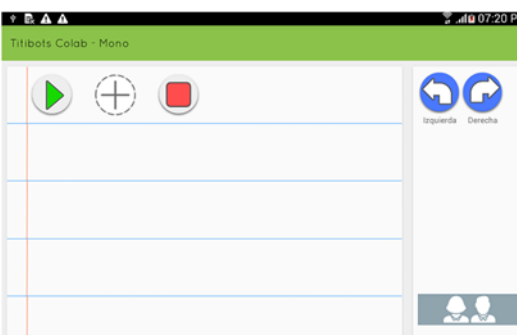
Figura 43. Interfaces de TITIBOTS Colab de solicitud del sexo y la edad.

Cuando la docente ha verificado que todos los estudiantes están listos, envía la notificación de empezar, agrupando los estudiantes por grupo. Cada estudiante recibe la notificación con el nombre y la figura del grupo al que pertenecen.

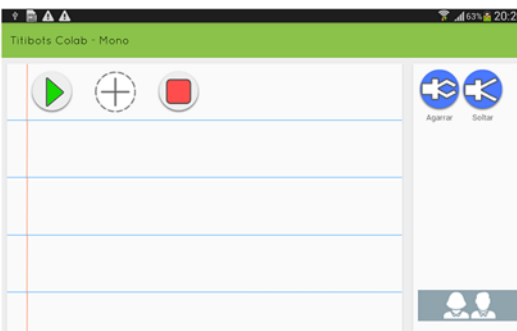
Una vez iniciada la sesión, los estudiantes pueden usar el “Tablero del Estudiante” que contiene la “Caja de Comandos” (ver Figura 44). Solo un estudiante puede manipular el programa a la vez (por turnos), y tendrá asignados una parte de la cantidad total de comandos (división de recursos). En cada desafío, los niños interactúan con su grupo para solucionarlo, dándose el proceso de comunicación para expresar sus ideas, coordinar sus acciones con los otros miembros y ofrecer ayuda a sus compañeros. La aplicación de la docente se encarga de mantener el control sobre cuál estudiante tiene el turno y qué rotación se realiza.



a) Estudiante con los comandos: Adelante y Atrás



b) Estudiante con los comandos: Izquierda y Derecha



c) Estudiante con los comandos: Agarrar y Soltar

Figura 44. Interdependencia positiva de distribución de recursos.

El “Controlador de Juego del Cliente” es el encargado de notificar a la interfaz y mantener el estado dependiendo de quien tenga al turno. Si el turno es del usuario actual, se activa el indicador del tiempo y se puede usar de la caja de comandos. En caso contrario, la aplicación del estudiante queda en pausa hasta que llegue su turno. El estudiante puede pasar el turno al terminar su trabajo (forma activa) o esperar que expire el tiempo del turno automáticamente (forma pasiva).

En caso que el estudiante necesite asistencia de la docente, puede solicitarla mediante la aplicación, y la docente será notificada. Este indicador puede ser desactivado por la docente una vez que considere que la asistencia ha sido brindada.

Cuando alguno de los integrantes de un grupo cree que la solución del desafío está terminada, puede solicitar la votación. En este proceso, los miembros del grupo se comunican, quien solicitó la votación les dice a sus compañeros el por qué lo hizo y los demás miembros dicen si están de acuerdo o no con él. Si la solicitud ha sido aprobada por todos los miembros, el juego se pausa y se envía la notificación de la solicitud a la aplicación de la docente, para que ella decida cuando se envía el programa al robot. En caso contrario, el juego se reanuda y sigue todo como estaba antes de pausarse el juego.

El “Manejador de la Conexión del Cliente” mantiene la conexión con la aplicación de la docente y usa el “Servicio de Descubrimiento de Red” al inicio para conectarse.

A continuación, se describen los robots utilizados en conjunto con la herramienta colaborativa de programación TITIBOTS Colab.

6.4.4 Robots

Los robots utilizados fueron:

- Robots contruidos con el Kit Robótico LEGO Mindstorms EV3. Se crearon tres tipos de robots (ver Figura 45a). Se utilizó el lenguaje de programación leJOS EV3. La

herramienta colaborativa de programación y los robots se comunican de forma inalámbrica por Wi-Fi.

- Robot construido con el Robot Lynx 5 Arm / RIOS / SSC-32 (ver Figura 45b) con Raspberry Pi 1 Modelo B. Se usaron los lenguajes de programación C y C++. La herramienta colaborativa de programación y el robot se comunican de forma inalámbrica por Wi-Fi.

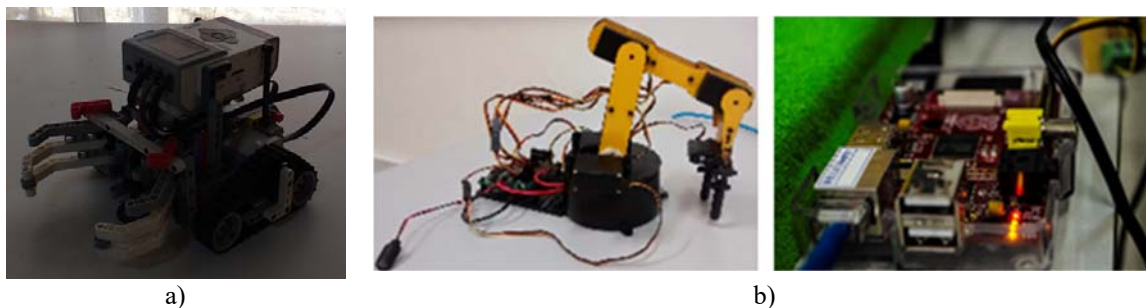


Figura 45. Robots utilizados en TITIBOTS Colab.

A continuación, se describe el protocolo de comunicación desarrollado para TITIBOTS Colab.

6.4.5 Protocolo de Comunicación

El diseño del protocolo tomó en cuenta algunas de las actividades de aprendizaje colaborativo, específicamente las relacionadas con la interacción docente-estudiantes y estudiantes-estudiantes.

Dado que las aplicaciones se construyeron para Android, se decidió usar JSON¹⁶ como formato de los mensajes del protocolo. La decisión fue tomada teniendo en cuenta las bibliotecas disponibles que ofrecen mensajes de serialización y deserialización rápida en objetos en Java.

¹⁶ URL: <http://www.json.org/>.

El protocolo tiene que proporcionar las herramientas que permiten una buena comunicación entre los dispositivos durante cada fase. Las siguientes secciones explican con detalle cuáles actividades del aprendizaje colaborativo se establecieron como parte del protocolo, desde la fase de pre-proceso, para configurar el tamaño de los grupos y cómo los dispositivos se conectan entre sí. Continuando con la fase en-proceso donde ocurre la interacción entre los estudiantes para encontrar la solución del problema planteado por la docente y por último la fase de pos-proceso en la que los estudiantes y la docente evalúan la solución antes de enviarla al robot.

Primera Fase: Pre-Proceso

Para la fase de pre-proceso (ver Tabla 7), se requiere el diseño del protocolo para el descubrimiento y organización del grupo. En las siguientes secciones se explica la configuración de algunas variables requeridas por la herramienta para habilitar la detección, y cómo se organizan los grupos.

Configuración

La docente debe seleccionar el tamaño del grupo (2 o 3 miembros) y el tiempo por turno que tendrá cada niño para trabajar (30, 45 o 60 segundos por defecto, aunque se puede configurar). La aplicación se encarga de los casos cuando el número de alumnos disponibles puede dejar un estudiante solo. Por ejemplo, si el tamaño del grupo es 2, esto significa que con 11 estudiantes se formarán cinco grupos de 2 y quedará un estudiante solo. La aplicación debe manejar estos escenarios y, hacer cuatro grupos de 2 y un grupo de 3. Esto es importante, con el fin de garantizar que todos los participantes estén en un grupo y los comandos se distribuyan adecuadamente (asignación de roles). En este punto el módulo de la docente (dispositivo de control) está listo para recibir conexiones de los estudiantes.

Descubrimiento

Cada interfaz de estudiante comienza a buscar el servidor maestro. Existen diferentes enfoques con el fin de asegurar una mejor conexión de aplicaciones del estudiante a la docente:

- **Bluetooth.** Este tipo de conexión se limita a 7 dispositivos esclavos por maestro. A pesar de que un esclavo puede ser maestro de otros 7 dispositivos, esta tecnología no es adecuada para una aplicación centralizada cuando el total de estudiantes supera este límite (Mander & Picopoulos, 2005).
- **Wi-Fi.** Este tipo de conexión ofrece una mayor cantidad de dispositivos interconectados, con las implementaciones de conexiones Peer-to-Peer (P2P): *Wi-Fi Direct* y *Network Service Discovery* (DNS). Para usar *Wi-Fi Direct*, es estrictamente necesario que los dispositivos sean compatibles con esta función (Android Developers, 2014a). Por el tipo de dispositivos móviles disponibles para la investigación, el *Wi-Fi Direct* no fue una buena opción, ya que eran incompatibles, por lo que se puso en práctica la solución universal DNS (la mayoría de dispositivos lo tienen) (Android Developers, 2014b). DNS requiere que la red actual (donde se conectan los dispositivos) apoye *Multicast* DNS.

La idea es que el dispositivo de la docente transmita el servicio de red, y los dispositivos de los estudiantes deben buscarla y conectarse. Esto proporciona una mejor experiencia del usuario que la conexión de forma manual de los dispositivos usando una dirección IP y el puerto.

Cuando un estudiante descubre la aplicación de la docente y se conecta correctamente, la aplicación de la docente asigna un identificador al estudiante que lo identifica dentro del grupo de estudiantes participantes. Esta identificación es importante cuando la aplicación de algún estudiante se desconecta, ya que será utilizada por la aplicación de la docente para saber quién está desconectado y conectarlo de nuevo, y si la sesión ya estaba en curso, a qué grupo pertenece para incorporarlo.

Organización de los Grupos

Una vez que las aplicaciones de todos los estudiantes están conectadas, la docente puede proceder a organizar los grupos. Esto se hace en función del número de estudiantes por grupo definido en la configuración. La aplicación creará cada grupo con un tamaño no inferior a 2 ni superior a 3, enviará un mensaje a todas las aplicaciones de los estudiantes informando sobre el grupo al que pertenece el estudiante y también la lista de comandos asignados a él.

Antes que la docente pueda iniciar la sesión con los estudiantes, debe asegurarse de que todos los estudiantes están listos. Una vez creado un grupo, un nuevo mensaje se envía, solicitando la preparación del estudiante. Este paso puede ser opcional dependiendo de cómo funciona normalmente el grupo, o de si hay más de una docente durante la sesión, las cuales puedan comprobar si los estudiantes están listos o no.

En la siguiente sección se explica el diseño de protocolo en la fase en-proceso del AC.

Segunda Fase: En-Proceso

Cuando se ha iniciado la sesión, la aplicación de la docente enviará un mensaje que contiene información sobre todos los grupos (pantalla de estado) y los estudiantes que pertenecen a cada grupo. La información que se envía a un grupo de estudiantes es un identificador del grupo, la fecha y hora en que se envió el mensaje (basado en el tiempo de la aplicación del docente), y los datos que indican cuál estudiante tiene el turno, el tiempo que lleva el turno, y la lista actual de los comandos en el área de trabajo de cada grupo (la cual está vacía la primera vez).

La marca de tiempo en el mensaje de estado de cada grupo es significativamente importante, ya que ayuda a la aplicación de los estudiantes a sincronizar su reloj. La importancia radica sobre todo para el estudiante, ya que su aplicación expirará en el momento justo, dando a todos los estudiantes un tiempo similar para trabajar.

En este punto el estudiante en turno puede realizar acciones de la lista de comandos asignados, como agregar, mover o eliminar comandos. Cuando el estudiante piensa que su trabajo está listo, debe informar a la aplicación de la docente. Este mensaje consiste en una lista de acciones o comandos, con un registro que contiene el tipo y la posición específica de cada comando, por ejemplo: el comando que se añadió con la posición específica, la posición original y la posición de destino cuando se mueve un comando. Esta lista de registro ayudará a la aplicación del docente para recrear los pasos realizados por el estudiante y reflejar los cambios correctamente a todos los miembros del grupo.

Mientras que los estudiantes están interactuando con la aplicación para resolver un problema, la docente puede controlar la actividad de todos los grupos, y observar el estado actual de cada grupo, cuál estudiante tiene el turno, los comandos que están en el área de trabajo, y otra información importante como cuál estudiante no ha estado trabajando. Esta información es obtenida de la aplicación del estudiante, puesto que puede decir si la acción se llevó a cabo de forma activa por parte del estudiante (estudiante añadió o no comandos) o pasiva (el estudiante deja que expire el tiempo de su turno enviando automáticamente un registro vacío con un indicador de inactividad).

Cuando la aplicación de la docente recibe el mensaje de las acciones realizadas por el estudiante, se actualiza el área de trabajo del grupo y se envía un nuevo estado a los estudiantes del grupo, asignando el turno al siguiente estudiante. Este proceso se realiza hasta que todos los estudiantes piensen la solución al problema planteado en el desafío está listo para voto. La votación corresponde a la última fase del proceso de AC, y se detalla en la sección siguiente.

Tercera Fase: Pos-Proceso

A pesar de que la docente puede monitorear el progreso de cada grupo, es difícil para ella saber cuándo un grupo está listo. Esa es la razón por la cual la votación se vuelve importante en el proceso y también un aspecto a incluir en el protocolo. Esta acción ayuda a la docente

a saber qué grupo está listo, por lo que puede inspeccionar la solución y ver si cumple o no con los criterios de éxito.

Si un estudiante piensa que la solución satisface el problema planteado en el desafío, puede presentar una solicitud de voto. Este es un mensaje de control que la aplicación del docente recibe y desencadena un voto, haciendo una pausa de la sesión y el envío de una solicitud de votación para los otros estudiantes del grupo. A continuación, la aplicación de la docente esperará a que el resto de los estudiantes voten. Se supone que el estudiante que envía la solicitud de voto ha dado un voto positivo a la solución, por lo que se detuvo su aplicación. Los otros estudiantes verán dos opciones, un voto positivo o uno negativo. Si todos los votos son positivos, la sesión permanece en pausa, y la docente puede llegar al grupo para una revisión final. Si la docente considera pertinente, le pide al grupo ir al robot y ver su solución en acción. Si el voto es negativo o la revisión con la docente aparece un problema en la solución, la docente vuelve a enviar la sesión para que los estudiantes pueden hacer cambios y luego solicitar una nueva votación.

Hay otros escenarios en los que la sesión puede ser interrumpida, como cuando algún estudiante se desconecta o cuando la docente hace una pausa en la sesión activa. Estos mensajes de interrupción son enviados por la aplicación de la docente. Esta interacción permite, también, decidir qué grupo va primero al robot, pero al final es la docente quien decide el orden de los grupos para enviar los comandos al robot.

Todos estos mensajes diseñados corresponden con el primer prototipo de diseño del protocolo destinado a facilitar la comunicación entre las aplicaciones relacionadas con la arquitectura centralizada. En la implementación del protocolo se definió la secuencia básica de los mensajes que son enviados entre las aplicaciones de la docente y los estudiantes. Esto sirve para determinar el tipo de mensaje y el contenido que debe ser incluido.

El primer paso de la implementación del protocolo fue definir el tipo de mensajes que se usaría. Un mensaje capaz de encapsular diferentes datos es particularmente útil cuando se

envía información como el estado de cada grupo. Cuando se definieron los tipos de mensajes del protocolo definidos, se estableció la estructura del mensaje, la cual es:

- **Message ID.** Identifica el mensaje, útil para evitar posibles mensajes duplicados.
- **Message Type.** Identifica el propósito del mensaje, por lo que el receptor sabe qué esperar de los datos.
- **Timestamp.** Fecha y hora en que se envió el mensaje. Esto es importante para los mensajes enviados desde la aplicación de la docente (que funciona como servidor), lo que hace posible que las aplicaciones de los estudiantes (que funcionan como clientes) ajusten su tiempo haciendo el trabajo correctamente con los mensajes de tiempo de espera.
- **Data.** Los datos del mensaje, esto dependerá del tipo de mensaje, pero los datos puede ser un valor booleano simple o una estructura compleja de objetos encapsulados en una lista.

En la Tabla 13 se muestran los tipos de mensajes creados para el protocolo de comunicación de TITIBOTS Colab, los datos que tendrá cada tipo de mensaje y su descripción.

Aunque `READY_CHECK` y `VOTE_REQUEST` son similares en estructura, se decidió mantener los dos tipos de mensajes, con el fin de mantener la semántica como acciones separadas, pero compartiendo los mismos datos. También, si se desea, la comprobación rápida puede ser una característica opcional, y es la docente quien decide si la aplica o no.

Tabla 13. Mensajes del protocolo de comunicación de TITIBOTS Colab.

Tipo de Mensaje	Datos	Descripción
ASSIGN_ID	Una cadena que contiene el identificador creado y asignado para el estudiante por la aplicación maestra.	Después del descubrimiento y de establecer conexión, la aplicación de la docente debe enviar un mensaje que indica el éxito de la conexión, indicando qué ID se ha asignado.
GROUP_ASSIGN	Una cadena que contiene información sobre el grupo como: identificación (id), nombre, imagen, tiempo de espera y una lista de comandos asignados al estudiante.	Cuando todos los estudiantes tienen su ID asignado, y la docente ha organizado los grupos, un nuevo mensaje se envía a todos los estudiantes desde la aplicación de la docente, informando acerca del grupo al que pertenecen.
READY_CHECK	Un tiempo de espera se establece acerca de cuánto tiempo la aplicación de la docente esperará a los estudiantes para una respuesta.	Opcionalmente, un mensaje puede ser enviado también por la solicitud de la docente, se pide a todos los estudiantes informar si están listos.
READY_RESPONSE	Contiene el ID del estudiante que responde, valor booleano que confirma si el estudiante está listo o no, y otro valor booleano establecido por la aplicación que indica si el usuario respondió activamente a la solicitud o había expirado.	Listas de mensajes de verificación de los estudiantes respondiendo.
BOARD_STATUS	Contiene el ID del grupo que pertenece el estado del área de trabajo, ID del estudiante en turno, tiempo que falta para que expire el turno, la lista de comandos del área de trabajo (debería estar vacía la primera vez).	Cuando todos los estudiantes están listos, la sesión se puede iniciar. La aplicación de la docente enviará a todos los estudiantes un primer estado del área de trabajo, que va a disparar en la aplicación del estudiante un cambio en la interfaz presentando el trabajo realizado por el grupo, los comandos asignados, y permitiendo solo un estudiante por turno.
ACTION	Contiene el ID del estudiante que presenta cambios en los comandos del área de trabajo, el ID del grupo al que pertenece (para comprobación de coherencia), un valor booleano que indica si lo hizo activa o pasivamente el estudiante, la lista de acciones realizadas en el área de trabajo (si las hay). Para agregar, se requiere la posición en la que se colocó el comando (agregado), la identificación de la orden, y el código de acción “ <i>add</i> ”. Para mover, se requiere la posición de inicio (donde el comando estaba originalmente), la posición de destino (donde el usuario trasladó el comando), y el código de acción “ <i>move</i> ”. Para eliminar, se requiere la posición que quiere quitar y el código de acción “ <i>remove</i> ”.	El estudiante a su vez puede agregar, mover y eliminar comandos a conveniencia, sobre la base de los comandos disponibles y lo que se adapte mejor para solucionar el problema dado por la docente. Una vez que el estudiante está listo se envía el mensaje para actualizar la aplicación de la docente, y esta a su vez las aplicaciones de los estudiantes del grupo.

Tabla 13. Mensajes del protocolo de comunicación de TITIBOTS Colab. (cont.)

Tipo de Mensaje	Datos	Descripción
VOTE_REQUEST	Contiene el ID del estudiante que solicita el voto, el ID del grupo al que pertenece (para comprobación de coherencia).	Los mensajes BOARD_STATUS y ACTION son enviados entre la docente y los estudiantes, hasta que un estudiante decida que la solución al problema planteado en el desafío está lista y debe ser enviado para revisión a la docente. Por lo cual, solicita la votación, esta solicitud es enviada a la aplicación de la docente. En la aplicación de la docente se crea un mensaje para hacer una pausa en las aplicaciones del grupo al que pertenece el estudiante que solicitó el voto, y se procede a enviar el mensaje a todos los estudiantes de la solicitud, menos el que solicitó el voto (ya que se supone es positiva). Este mensaje es similar a READY_CHECK, pero es tipo VOTE.
VOTE_RESPONSE	Contiene el ID del estudiante que responde, un valor booleano que confirma si el estudiante está listo o no, y otro valor booleano establecido por la aplicación que indica si el usuario respondió activamente a la solicitud o había expirado.	Listas de mensajes de verificación de los estudiantes respondiendo a la solicitud de voto. Cuando todos los votos se reciben un nuevo mensaje se envía desde la aplicación de la docente informando el resultado de la votación. Si el voto no se aprueba, un BOARD_STATUS se envía para reiniciar la sesión en el punto en quedó a la hora de ir a pausa. Todo sigue normalmente, el turno para cada estudiante, el uso de mensajes BOARD_STATUS y ACTION. Si el voto se aprueba, la sesión se mantiene en pausa y la docente puede enviar los comandos al robot. Si la docente quiere empezar una nueva sesión puede hacerlo, y la aplicación enviará un mensaje BOARD_STATUS sin comandos en los datos.
END_SESION	N/A	Termina la sesión, la aplicación de la docente es quien envía el mensaje, el cual hará que se cierre la conexión de todos los clientes y recursos.

En el siguiente capítulo se muestran los resultados y la discusión de la investigación con base en las diferentes pruebas realizadas.

CAPÍTULO 7. RESULTADOS Y DISCUSIÓN

En este capítulo se presentan y discuten los resultados obtenidos de las evaluaciones llevadas a cabo en esta investigación. Los resultados y la discusión fueron divididos en tres secciones que responden a las dos últimas preguntas de investigación, ya que las dos primeras fueron contestadas en el Capítulo 6.

7.1 Usabilidad de la Herramienta en la Primera Infancia

En la evaluación de usabilidad se realizaron varias pruebas pilotos que fueron mejorando la interfaz de la herramienta colaborativa de programación TITIBOTS Colab, en relación con su usabilidad y su funcionalidad. En la Sección 7.1.1 se muestran los resultados de la evaluación de TITIBOTS, y en la Sección 7.1.2 se muestran los resultados de la evaluación de TITIBOTS Colab.

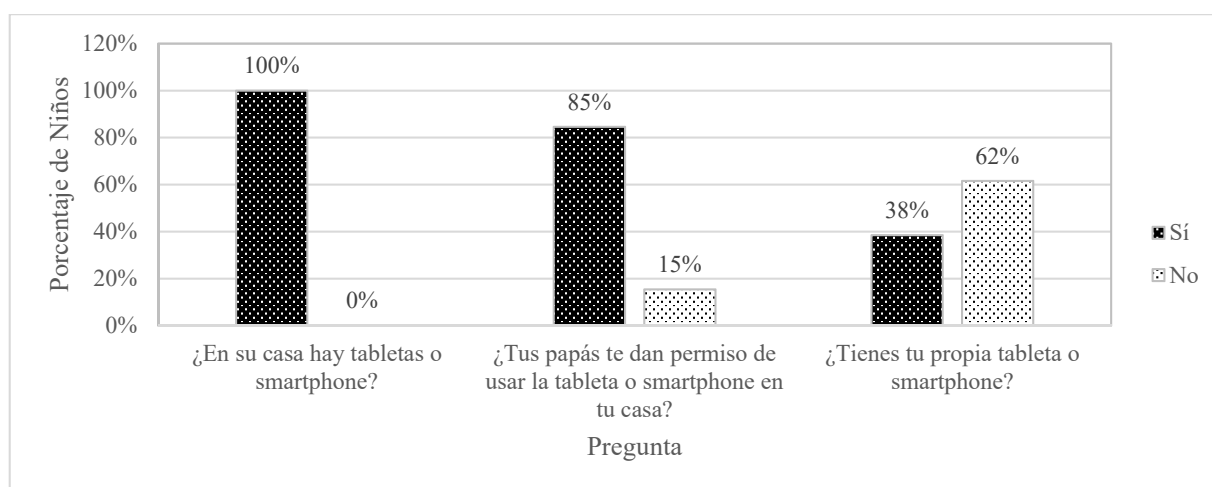
7.1.1 Usabilidad de TITIBOTS

En la primera fase de esta evaluación, se realizaron varias validaciones y dos pilotos con la herramienta de programación TITIBOTS (trabajo individual) (ver Sección 5.2.1). Las estadísticas de los niños participantes en esta primera fase se muestran en la Tabla 14, distribuidos por etapa, sexo y edad. Con respecto a la primera etapa, participaron 6 expertos y 2 docentes de preescolar en las validaciones de las interfaces (6 mujeres y 2 hombres). En la Figura 46 se muestra el gráfico con la información de los perfiles de usuarios obtenidos mediante entrevistas semiestructuradas, las cuales se llevaron a cabo en las pruebas de usabilidad con 13 niños (tercera y cuarta etapa).

La validación de la iconografía e interacción se centró en la capacidad de los niños para entender la interfaz e interactuar con la herramienta. Los resultados fueron divididos por edad, donde el 30% de los niños se consideraron como grupo de control. Por tanto, se tuvo un 25% de niños de 4 años, un 25% de 5 años, un 20% de 6 años y un 30% de control.

Tabla 14. Participantes de la evaluación de TITIBOTS.

Etapa	Número de Participantes	Sexo		Edad		
		Femenino	Masculino	4	5	6
Segunda etapa: Validación de iconografía e interacción con niños	40 (76%)	20 (50%)	20 (50%)	11 (27.5%)	19 (47.5%)	10 (25%)
Tercera etapa: Proceso de pruebas con usuarios finales	7 (13%)	4 (57%)	3 (43%)	7 (100%)	0 (0%)	0 (0%)
Cuarta etapa: Desarrollo y uso en un entorno real	6 (11%)	4 (67%)	2 (33%)	4 (67%)	2 (33%)	0 (0%)
Total (Porcentaje)	53 (100%)	28 (53%)	25 (47%)	22 (42%)	21 (40%)	10 (18%)

**Figura 46.** Información del perfil del usuario de TITIBOTS.

Los resultados fueron obtenidos de la observación de los investigadores según lo que los niños contestaron y realizaron en cada prueba, centrándose en su capacidad para entender la interfaz e interacción propuesta. La Tabla 15 muestra los resultados de esta validación con los niños. Los resultados fueron los siguientes:

- En general el concepto y el diseño fueron agradables para los niños, el 100% manifestó que les gustaba el mono (personaje).
- Los íconos de inicio y fin no fueron comprendidos por los niños. Agarrar y soltar tuvieron solo un 23% de exactitud, mientras que los otros comandos tenían más de 60% de reconocimiento (ver Tabla 15).
- En cuanto al diseño de la interfaz, se encontró que los resultados fueron similares a la hora de distinguir la mejor de ellas (ver Tabla 15).

- En cuanto a la interacción, aunque los resultados fueron similares para distinguir la mejor forma, se notó un mejor control de la interacción agarrar y soltar. Por tanto, y basados en la literatura (ver Sección 6.2), se optó por escoger este tipo de interacción (ver Tabla 15).

Tabla 15. Resultados de la validación de íconos, interfaz e interacción.

Ícono	Edad				% Respuestas Correctas
	4	5	6	Control	
Inicio	0%	0%	13%	0%	3%
Fin	0%	0%	13%	0%	3%
Adelante	40%	90%	100%	42%	65%
Atrás	40%	90%	100%	42%	65%
Izquierda	40%	100%	100%	50%	70%
Derecha	40%	100%	100%	50%	70%
Apagar	50%	80%	88%	50%	65%
Encender	50%	80%	88%	42%	63%
Agarrar	0%	20%	63%	17%	23%
Soltar	0%	20%	63%	17%	23%

Interfaz	Edad				% Respuestas Correctas
	4	5	6	Control	
Huellas	30%	30%	63%	17%	33%
Líneas	30%	20%	63%	8%	28%
Sin guía	30%	40%	50%	17%	33%

Interacción	Edad				% Respuestas Correctas
	4	5	6	Control	
Inserción	100%	67%	100%	50%	77%
Agarrar y soltar	67%	100%	67%	100%	85%
Modo guiado	100%	100%	67%	50%	86%

Posteriormente, los resultados fueron presentados y discutidos con tres docentes de preescolar (reunión organizada en conjunto con PROTEA). Las docentes brindaron sus comentarios y recomendaciones con base en los resultados obtenidos.

Con los resultados obtenidos de la validación y las recomendaciones de las docentes, se realizaron varios cambios en el diseño de la herramienta. Los más significativos fueron:

- Se minimizó la carga gráfica de la interfaz: disminución de los colores y figuras del fondo, eliminando distracciones visuales para permitir que los niños se centren en los elementos relevantes).
- Se realizaron cierres visuales (atención) y, el uso de colores primarios y secundarios.
- Se valoró el tamaño de los íconos y su consistencia.
- Se utilizaron instrucciones verbales concisas y sencillas, con palabras familiares para los niños. Usar narrativa amigable.
- Se utilizó un tipo de letra sin *serifs*.

Con respecto a la interfaz, se decidió realizar un diseño minimalista, lo más simple posible y sin sobrecarga gráfica. Por lograrlo, la pantalla donde los niños programan (la parte más importante del diseño de la interfaz) se rediseñó como una hoja de cuaderno, donde los niños pueden escribir sus historias para que el robot títi las interprete. Así, se logró un diseño simple, fácil de usar e intuitivo, para que los niños entiendan el concepto de secuencia y el orden que siguen las instrucciones que colocan en la interfaz de programación: arriba-abajo e izquierda-derecha.

Con los cambios realizados al diseño gráfico de la interfaz, se organizó una reunión con las tres docentes de preescolar (en PROTEA). Las docentes avalaron el nuevo diseño y destacaron lo siguiente:

- La interfaz sigue el proceso de lectoescritura: arriba-abajo e izquierda-derecha, lo cual es valioso y debe conservarse.
- El diseño de la herramienta asocia cada ícono a una palabra escrita. La palabra escrita es un reflejo del pensamiento, y el pensamiento se convierte en acciones. Esto es valioso y útil para el desarrollo del lenguaje de los niños.

En esta misma reunión, se les solicitó a las docentes brindar recomendaciones en la parte didáctica, con el objetivo de diseñar estrategias didácticas que se usarán en los futuros pilotos que se realizarían con niños. Entre las recomendaciones dadas están:

- Cualquier actividad donde se utilice tecnología con niños de preescolar debe ser precedida de actividades con los niños a nivel concreto.
- Los niños deben haber trabajado previamente con lateralidad, ubicación espacial, secuencias. Este tipo de experiencias son importantes y necesarias para la comprensión de la herramienta.
- La herramienta debe formar parte de los contenidos de preescolar, como complemento para estudiarlos en clase.

El diseño también fue avalado por los expertos de la FOD. Con el nuevo diseño avalado y las recomendaciones de las docentes en la parte didáctica, se procedió a implementar la herramienta y realizar el primer piloto con niños de preescolar. Este piloto duró solo un día y se realizaron tres desafíos.

Las observaciones de este piloto se realizaron con el fin de evaluar la usabilidad de la herramienta (Mathilde Bekker, Barendregt, Crombeen, & Biesheuvel, 2005), determinar la necesidad de la intervención del docente al usar TITIBOTS, y si la mediación juega un papel importante en la utilización de la herramienta y la interacción entre los niños. Las métricas de usabilidad se muestran en la Tabla 16.

Tabla 16. Resultados de las métricas de usabilidad en TITIBOTS.

Atributo de Usabilidad	Métricas	Meta de Usabilidad
Facilidad de aprendizaje	Tiempo promedio empleado para completar un desafío por primera vez = 11,7 minutos (entre 10 y 30 minutos)	Lograda
Eficiencia	Tiempo promedio de entrenamiento = 30 minutos (entre 30 y 60 minutos)	Lograda
	Porcentaje de desafíos exitosos = 33% (> 70%)	No lograda
Cualidad de ser recordado	Tiempo promedio para completar un desafío = 11,6 minutos (entre 10 y 20 minutos)	Lograda
	Porcentaje de respuestas correctas sobre la aplicación = 93% (> 80%)	Lograda
Efectividad	Promedio de errores = 4,3 (entre 5 y 10 errores, o menos)	Lograda
	Promedio de errores recuperados = 0,7, 16% del total de errores (> 60%)	No lograda
Satisfacción	Gusta = 100% (> 65%)	Lograda
	Fácil = 64% (> 65%)	No lograda

En este piloto, se realizaron dos sesiones, una sesión con mediación técnica (la docente explicaba el uso de la herramienta) y otra con mediación lúdica (la docente realizaba juegos

para explicar el uso de la herramienta). Además, se diseñaron tres desafíos para que los niños resolvieran. Sin embargo, debido a que no se aprende a programar en un lapso corto, se esperaba que no se lograran todos los desafíos en el plazo previsto. La evaluación mostró que la herramienta es fácil de usar para los niños de edad preescolar. La interfaz mostró ser simple e intuitiva. Por otra parte, todos los participantes mostraron interés en la herramienta y mostraron interés de seguir usándola después de la actividad.

Se observó que la mediación lúdica tiene una fuerte influencia en el uso de la herramienta y en el nivel de logro de los desafíos. En la sesión de mediación lúdica, el uso de la herramienta era más simple. Por el contrario, en la sesión de mediación técnica, los niños usaron la aplicación como un control remoto (es decir, colocaban un comando en la interfaz y lo enviaban varias veces al robot). Lo sucedido en esta última sesión es, precisamente, lo que no se quiere que los niños hagan con la herramienta, ya que se quiere que solucionen cada desafío creando una secuencia de pasos y luego la introduzcan completa en la herramienta antes de enviarla al robot. El uso de la herramienta como un control remoto no logra los resultados que se buscan en esta investigación.

Dos de los participantes en la sesión de mediación técnica terminaron frustrados y no terminaron los desafíos. Por otra parte, todos los niños de la sesión de mediación lúdica continuaron tratando hasta que el tiempo terminó. Como se esperaba, los niños solo fueron capaces de terminar el primer desafío: la mitad de los niños en la sesión no mediada y todos ellos en la sesión mediada. La diferencia principal fue que los que recibieron la mediación lúdica tuvieron éxito con una menor cantidad de intentos (ver Tabla 17) y en menor tiempo (ver Tabla 18).

Tabla 17. Cantidad de niños que realizaron con éxito los desafíos.

Tipo de Mediación	Desafío 1	Desafío 2	Desafío 3
Técnica	2 (50%)	0 (0%)	0 (0%)
Lúdica	3 (100%)	0 (0%)	0 (0%)

Tabla 18. Tiempo promedio y cantidad de intentos promedio de los desafíos exitosos.

Tipo de Mediación	Tiempo Promedio	Promedio de Intentos
Técnica	8,5min	5,5
Lúdica	4,1min	4,0

Se obtuvieron otros resultados del pilotaje a partir de las observaciones de los evaluadores y las grabaciones. Los evaluadores registraron en las notas de los folletos utilizados las siguientes observaciones:

- Los niños se mostraron interesados, felices y atentos con la herramienta de programación, les fue fácil usarla y se divertieron.
- La interfaz fue fácil de usar para los niños, comprendieron su funcionalidad, la interacción, el significado y el uso de los comandos, así como otros botones de la aplicación: cargar programa, desconectar, entre otros.
- Algunos niños usaron la herramienta de programación como control remoto. No había en los niños noción de lo que pasaba o lo que habían logrado (cuando lograban realizar con éxito el desafío).
- Algunos niños crearon machotes mentales y los usaron indiscriminadamente, como bloques de código funcionales. No había noción del estado anterior.
- La ubicación espacial en niños de estas edades es importante, ya que deben estar detrás o al lado del robot (mirar hacia el mismo punto que mira el robot) para decirle correctamente al robot hacia donde girar, ya que si están frente a él le dirán de forma incorrecta el giro a realizar (pues lo ven como espejo).
- La mediación es importante. En relación con la comprensión de los comandos, mantener el interés de los niños, y el control de la frustración (en la mediación técnica los niños abandonaron al frustrarse, mientras que en la mediación lúdica los niños esperaron y volvieron a intentarlo).
- Los niños no colaboraron (como se esperaba, ya que la literatura dice que en esta etapa son egocéntricos). Hubo solo curiosidad de ver qué hacían los otros niños.

Entre las observaciones dadas por los evaluadores se tienen las siguientes:

- La usabilidad y la funcionalidad de la aplicación fueron exitosas.

- El uso de juego con los niños es importante en este proceso.
- Los distintos comandos tienen distintas funciones y algunas son más complejas.
- Los errores generaron frustración.
- La lateralidad, la direccionalidad y la ubicación espacial (orientación espacial) de los niños entre 4 y 6 años se confirmó con lo que se reporta en la literatura. Tienen la noción de izquierda y derecha, pero con relación a su propio cuerpo, por lo que los niños deben estar al lado del robot, y no al frente, para indicarle con éxito el lado a girar. No obstante, los ejercicios que involucran los comandos de giro fueron buenas prácticas para que el niño vaya afianzando su conciencia a su posición con relación con los objetos que lo rodean, la cual queda afianzada totalmente hacia los 8 años.

Por otra parte, se encontraron varios problemas en la herramienta durante esta evaluación:

- La garra del robot se rompió fácilmente. Esto obligó a un rediseño tanto en el software como en el hardware (robot). En el software se agregaron reglas para evitar una orden de abrir si la garra ya estaba abierta, y de cerrar si estaba cerrada. En cuanto al hardware, se rediseñó la garra del robot.
- La conexión Bluetooth fue inestable. El software se rediseñó para que si se desconectaba se volviera a conectar de forma automática.
- Los comandos colocados en pares (por ejemplo: encender y apagar, agarrar y soltar) eran más fáciles de entender por los niños que los que estaban colocados en forma de teclado (adelante y atrás, izquierda y derecha). La interfaz del software fue rediseñada para colocar todos los comandos en pares.
- La retroalimentación en tiempo real por voz era requerida, para que el usuario supiera lo que está sucediendo. El software se rediseñó para que se estuviera retroalimentando por voz al usuario. Por ejemplo: si no hay instrucciones y el niño da clic en el botón Jugar (que envía los comandos al robot), hay que decirle que el robot no tiene instrucciones que ejecutar, y que debe dárselas.
- Las tabletas se apagaban si estaban unos minutos sin usar. El software se rediseñó para que las tabletas se mantuvieran encendidas mientras la herramienta estaba activa.

Además, por recomendación de los expertos y los evaluadores se agregaron ciertas funcionalidades:

- Se agregaron niveles a la herramienta. En cada nivel se muestra un nuevo par de comandos: Nivel 1 aparecen solo los comandos adelante/atrás, Nivel 2 aparecen los comandos del nivel anterior y los comandos izquierda/derecha, Nivel 3 aparecen los comandos del nivel anterior y los comandos encender/apagar, Nivel 4 aparecen los comandos del nivel anterior y los comandos agarrar/soltar.
- Se agregó un botón de limpiar para borrar los comandos que haya en la pantalla de programación. Además, se ejecuta un *script* que reinicia al robot en el estado inicial (garra abierta) cuando se quiere crear un nuevo programa.
- Se agregó un botón de cargar para abrir un programa anteriormente creado (la herramienta guarda el último programa creado).

Con los cambios realizados, se llevó a cabo el segundo piloto, el cual consistió en un taller de cuatro días en la FOD (prueba en un entorno real).

El resultado más importante del taller (obtenido a partir de los reportes de los evaluadores, las grabaciones y las métricas de usabilidad) fue que los niños estaban siempre felices y atentos con la herramienta. Los niños expresaron que era fácil de usar y divertida, de acuerdo con el cuestionario de satisfacción. Por otra parte, no tuvieron problemas para entender los comandos u otros botones, tales como limpiar, cargar programa y desconectar (de acuerdo con la prueba de memoria realizada). Las métricas de usabilidad obtenidas en el taller se muestran en la Tabla 19. En el Apéndice 18 se muestran los datos utilizados en el cálculo de las métricas de usabilidad de acuerdo con los desafíos realizados por los niños durante el taller. Las métricas de usabilidad se lograron, los resultados se encuentran dentro de las metas establecidas por Granic, Cukusic y Nielsen (Granic & Cukusic, 2011; Jakob Nielsen, 1993).

Los expertos de la FOD consideraron que la aplicación de este taller fue exitosa, porque observaron que todos los participantes lograron el conocimiento base propuesto con la

actividad. Además, la docente y los observadores consideraron que los eventos en que los niños realizaban los comandos fueron cruciales para el proceso de aprendizaje.

Tabla 19. Resultados de las métricas de usabilidad en TITIBOTS.

Atributo de Usabilidad	Métricas	Meta de Usabilidad
Facilidad de aprendizaje	Tiempo promedio empleado para completar un desafío por primera vez = 10,9 minutos (entre 10 y 30 minutos)	Lograda
Eficiencia	Tiempo promedio de entrenamiento = 43,7 minutos (entre 30 y 60 minutos)	Lograda
	Porcentaje de desafíos exitosos = 88% (> 70%)	Lograda
	Tiempo promedio para completar un desafío = 18 minutos (entre 10 y 20 minutos)	Lograda
Cualidad de ser recordado	Porcentaje de respuestas correctas sobre la aplicación = 100% (> 80%)	Lograda
Efectividad	Promedio de errores = 3,4 (entre 5 y 10 errores, o menos)	Lograda
	Promedio de errores recuperados = 2,3, 69% del total de errores (> 60%)	Lograda
Satisfacción	Gusta = 100% (> 65%)	Lograda
	Fácil = 83% (> 65%)	Lograda

Se encontró que los participantes mayores (6 años) alcanzaron una tasa de éxito excepcional, los evaluadores concluyen que los desafíos propuestos eran fáciles para ellos. Los participantes más jóvenes (4 años) tenían ciertas dificultades. Sin embargo, la interacción con los mayores les ayudó. Esto condujo a pensar que un taller donde se integran los niños de diferentes edades puede ser beneficioso.

Aunque el tiempo de cada sesión fue programado para dos horas, se trabajó hasta una hora y treinta minutos con los niños, ya que la mediadora aconsejó terminar en ese tiempo porque los niños estaban cansados. Debido a esto, se recomienda que el límite de tiempo para trabajar con la herramienta en niños de preescolar sea de una hora y treinta minutos.

Al discutir los resultados con los expertos de la FOD y las docentes de preescolar, se recomendó lo siguiente:

- El tiempo de cada sesión debe ser de hora y media, ya que este se considera como el límite superior posible para tener la atención de los niños. Se debe considerar reducir la duración o plantear recesos.
- El taller debe contar con la presencia de una asistente que apoye a la mediadora en las actividades, pues la tarea resulta sumamente absorbente para una sola persona.

Las métricas de satisfacción muestran que al 100% de los niños les gustó la herramienta de programación y el robot (los 13 niños de la tercera y la cuarta etapas). Además, 31% encontró la herramienta difícil de usar (aunque en observaciones y grabaciones esto no se reflejó). En cambio, el 77% de los niños afirman que el robot era fácil de usar. Como parte del cuestionario de satisfacción, se pidió a los niños que dibujaran lo que más les gustó de la actividad. El 62% de los niños dibujó un robot o un mono, un 8% dibujó una tableta, y el 30% dibujó una tableta y un robot. Cuando se les pidió describir sus dibujos, dijeron que amaban a los monos robots que usaron, y les gustaba decirles qué hacer.

Algunas de las características que se encuentran en la literatura para niños de este rango de edad evidencian el egocentrismo y la noción de direccionalidad referente al cuerpo (conciencia de la existencia de un lado derecho y un lado izquierdo de su cuerpo, y la capacidad para proyectarlos a el mundo que le rodea, es decir, la conciencia de tener su posición relativa según los objetos que lo rodean).

En relación con el egocentrismo, como era de esperar, al inicio cada niño se centró en su tableta y su robot, no hubo interacción entre ellos. A veces era solo curiosidad, pero a medida que avanzaba el taller, los niños comenzaban a ayudar a los demás, explicando qué hacer e indicaban lo que estaba mal.

En cuanto a la direccionalidad (lateralidad y la orientación espacial) de los niños entre 4 y 6 años, se ha confirmado que la noción de izquierda y derecha en relación con el cuerpo es más fácil. De esta manera, los niños deben estar de pie al lado del robot, y no al frente a él. Esta posición permitió que indicaran al robot con éxito los movimientos de izquierda y derecha. Por otro lado, los desafíos que implicaban los comandos de rotación son buenas prácticas para que el niño fortalezca la conciencia de su posición en relación con los objetos que lo rodean.

Dado los resultados de las evaluaciones, se puede concluir que TITIBOTS es fácil de usar para niños de 4 a 6 años. Los niños siempre estuvieron interesados, felices y atentos durante el uso de la herramienta, pero lo más importante es que se divirtieron. Fue evidente que esta

herramienta permitió a los niños programar robots con el fin de resolver tareas específicas de forma lúdica. Además, al final de todas las pruebas la herramienta quedó funcional, ya que en estas pruebas se ejecutaron las tareas especificadas en los requisitos de la herramienta (casos de uso), garantizando que hace lo que debe (casos positivos), que no falla (casos negativos) y que cumple con lo que se ha solicitado (satisface los requisitos).

Por último, debido a la característica del egocentrismo, se pensó en la modificación de la herramienta, para automatizar actividades del aprendizaje colaborativo y crear estrategias didácticas para que los niños colaboren en la solución de un problema dado. Esto es parte de TITIBOTS Colab.

7.1.2 Usabilidad de TITIBOTS Colab

En la segunda fase de la evaluación de usabilidad, se realizaron dos pilotos con la herramienta de programación TITIBOTS Colab (herramienta colaborativa) (ver Sección 5.2.2). Los niños participantes de esta fase se muestran en la Tabla 20 por etapa, distribuidos por sexo y edad. En la Figura 47 se muestran los gráficos de las preguntas 1-4 y 6, y las Figuras 47 y 48 los gráficos de las preguntas 5 y 7 (respectivamente) con la información del perfil de los usuarios obtenidos mediante entrevistas semiestructuradas realizadas a los encargados de los niños sobre el perfil tecnológico de los niños, las cuales se llevaron a cabo en las pruebas de usabilidad con los 30 niños participantes de esta fase.

Tabla 20. Participantes de la evaluación de TITIBOTS Colab.

Etapa	Número de Participantes	Sexo		Edad		
		Femenino	Masculino	4	5	6
Primera etapa: Proceso de pruebas con usuarios finales (Brasil)	16 (53%)	7 (44%)	9 (56%)	0 (0%)	14 (87%)	2 (13%)
Segunda etapa: Desarrollo y uso en un entorno real (Costa Rica)	14 (47%)	7 (50%)	7 (50%)	2 (14%)	12 (86%)	0 (0%)
Total (Porcentaje)	30 (100%)	14 (47%)	16 (53%)	2 (7%)	26 (86%)	2 (7%)

En la Figura 47 se observa que en el 90% de los hogares de los niños cuentan con dispositivos móviles (en Costa Rica se dio el 100% y en Brasil un 81%). El 83% de los encargados de los

niños les permiten usar algún dispositivo móvil (100% en Costa Rica y 69% en Brasil). En su mayoría, los niños usan los dispositivos móviles para juegos, el 48% para juegos educativos (ver Figura 48). Por su parte, el 33% de los niños tienen su propio dispositivo (43% de Costa Rica y 25% de Brasil), siendo la tableta el tipo de dispositivo escogido para los niños (ver Figura 49).

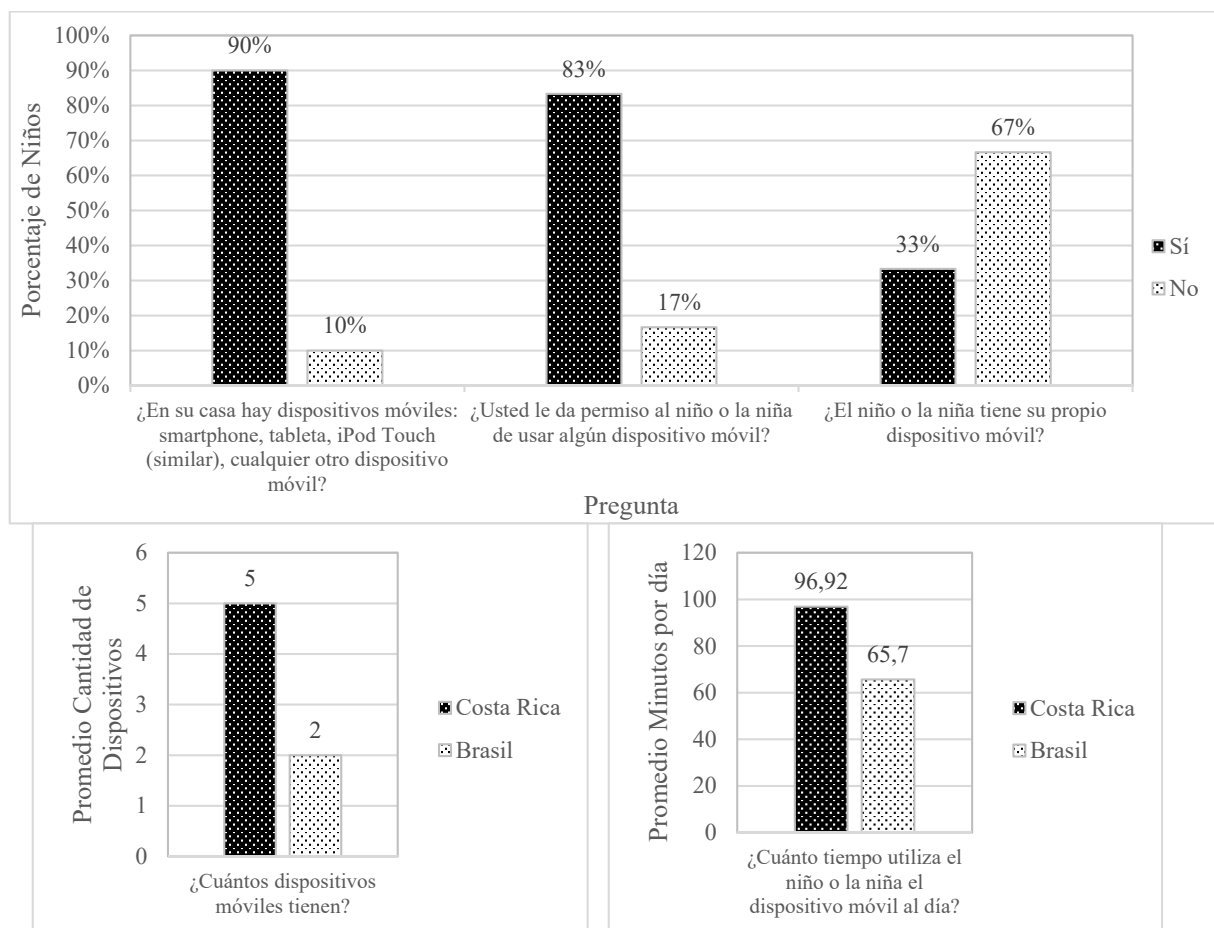


Figura 47. Información del perfil del usuario de TITIBOTS Colab.

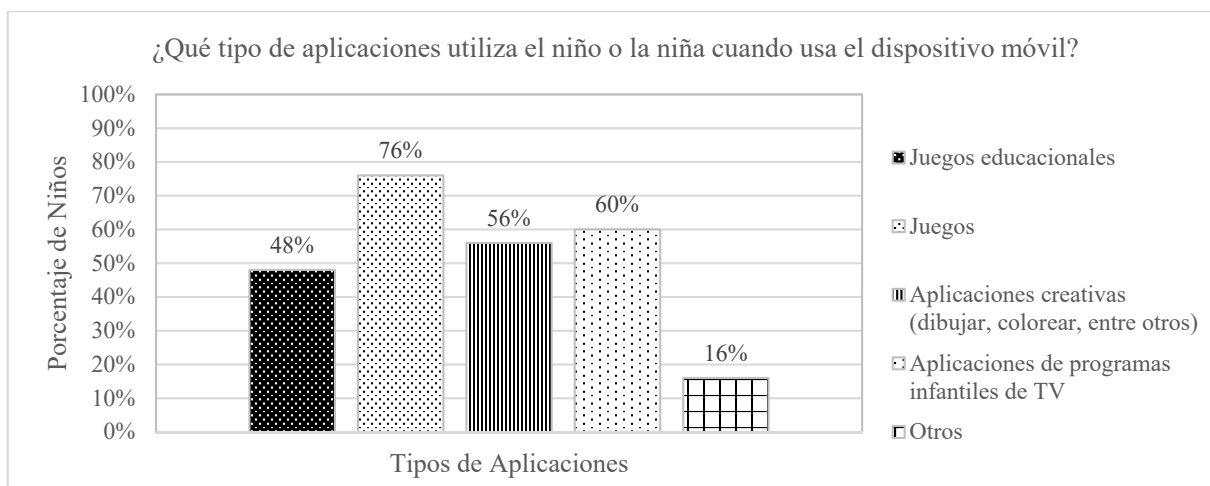


Figura 48. Información del perfil del usuario de TITIBOTS Colab – Pregunta 5.

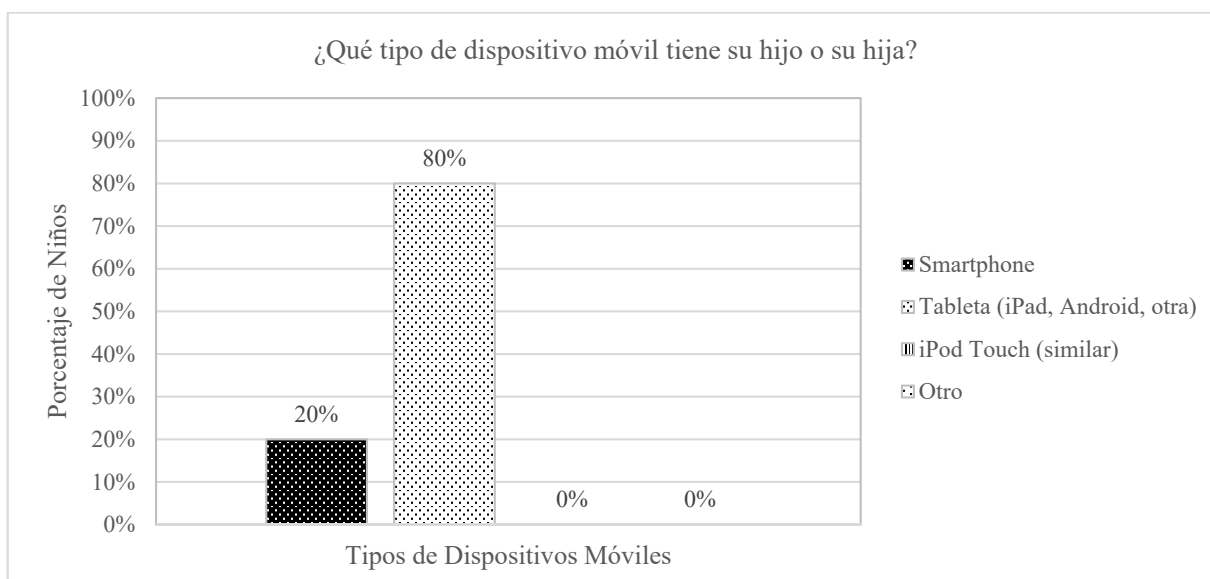


Figura 49. Información del perfil del usuario de TITIBOTS Colab – Pregunta 7.

El resultado más importante de los talleres de Brasil y Costa Rica (obtenido a partir de los reportes de los evaluadores, las grabaciones y las métricas de usabilidad) fue que los niños estaban siempre felices y atentos con la herramienta. Los niños expresaron que era fácil de usar y divertido, de acuerdo con el cuestionario de satisfacción. Por otra parte, no tuvieron problemas para entender los comandos u otros botones, tales como votar, llamar al docente y pasar el turno (de acuerdo con la prueba de memoria realizada).

Al inicio, el trabajo colaborativo de los grupos no fue el deseado, pero sí el esperado: en varios grupos se mostró la característica de egocentrismo en niños que eran dominantes, donde cada quién quería hacer solo lo que deseaba, y no escuchaba o atendía a sus compañeros de grupo. Conforme pasaban los días del taller y las actividades de colaboración (usando la herramienta colaborativa), los grupos comenzaron a trabajar juntos para conseguir un objetivo común. Se observó la cohesión de los grupos y la comunicación entre los miembros de cada grupo se hizo evidente. Además, se observó el análisis y la solución en grupo de los desafíos propuestos, mostrando el desarrollo del pensamiento computacional en los niños de preescolar.

Las métricas de usabilidad obtenidas en el taller de Brasil y Costa Rica se muestran en la Tabla 21 y la Tabla 22, respectivamente. En el Apéndice 19 y el Apéndice 20 se muestran los datos utilizados para calcular las métricas de usabilidad según los desafíos realizados por los niños de Brasil y Costa Rica, respectivamente, durante el taller. Las métricas de usabilidad se lograron en ambos talleres, cuyos resultados se encuentran dentro de las metas establecidas por Granic, Cukusic y Nielsen (Granic & Cukusic, 2011; Jakob Nielsen, 1993).

Tabla 21. Resultados de las métricas de usabilidad en Brasil.

Atributo de Usabilidad	Métricas	Meta de Usabilidad
Facilidad de aprendizaje	Tiempo promedio empleado para completar un desafío por primera vez = 19 minutos (entre 10 y 30 minutos)	Lograda
Eficiencia	Tiempo promedio de entrenamiento = 38,8 minutos (entre 30 y 60 minutos)	Lograda
	Porcentaje de desafíos exitosos = 100% (> 70%)	Lograda
Cualidad de ser recordado	Tiempo promedio para completar un desafío = 16,1 minutos (entre 10 y 20 minutos)	Lograda
	Porcentaje de respuestas correctas sobre la aplicación = 100% (> 80%)	Lograda
Efectividad	Promedio de errores = 2,5 (entre 5 y 10 errores, o menos)	Lograda
	Promedio de errores recuperados = 2,2, 89% del total de errores (> 60%)	Lograda
Satisfacción	Gusta = 100% (> 65%)	Lograda
	Fácil = 100% (> 65%)	Lograda

Al trabajar de forma colaborativa con niños de preescolar, se observó la necesidad que la docente (mediadora) cuente con la ayuda de al menos dos asistentes, otra docente (que ayude a repartir material, realizar las actividades y controlar los grupos) y alguien a cargo de la

parte técnica (que ayude al inicio en alistar las tabletas con la aplicación, conectar el robot y todas las tabletas a la red local asignada y solucionar cualquier problema técnico que surja).

Tabla 22. Resultados de las métricas de usabilidad en Costa Rica.

Atributo de Usabilidad	Métricas	Meta de Usabilidad
Facilidad de aprendizaje	Tiempo promedio empleado para completar un desafío por primera vez = 17,5 minutos (entre 10 y 30 minutos)	Lograda
Eficiencia	Tiempo promedio de entrenamiento = 34,8 minutos (entre 30 y 60 minutos)	Lograda
	Porcentaje de desafíos exitosos = 100% (> 70%)	Lograda
	Tiempo promedio para completar un desafío = 12,5 minutos (entre 10 y 20 minutos)	Lograda
Cualidad de ser recordado	Porcentaje de respuestas correctas sobre la aplicación = 100% (> 80%)	Lograda
Efectividad	Promedio de errores = 2 (entre 5 y 10 errores, o menos)	Lograda
	Promedio de errores recuperados = 1,7, 88% del total de errores (> 60%)	Lograda
Satisfacción	Gusta = 100% (> 65%)	Lograda
	Fácil = 100% (> 65%)	Lograda

Además, se recomienda trabajar con grupos de 15 estudiantes (máximo), por el trabajo que conlleva abordar con los niños habilidades relacionadas al aprendizaje de la programación y al trabajo en equipo, dado por su nivel de desarrollo y las características propias del rango de edad.

Por otra parte, las métricas de satisfacción muestran que al 100% de los niños (en ambos talleres: Brasil y Costa Rica) les gustó la herramienta colaborativa de programación, y la encontró fácil de usar. Además, al 97% de los niños les gustó trabajar en grupo y el 100% consideran que su participación fue importante en el grupo. Como parte del cuestionario de satisfacción, se pidió a los niños que dibujaran lo que más les gustó de la actividad. En el Apéndice 22 se presenta el dibujo más representativo del taller realizado por una niña de la Escuela Santa Mónica. El 40% de los niños dibujó a su equipo trabajando, un 57% dibujó un robot (Mono Tití o Serpiente Cora), un 47% dibujó una tableta, y el 30% dibujó una tableta y un robot. Cuando se les pidió describir sus dibujos, dijeron que amaban a los monos robots que usaron, y les gustaba decirles qué hacer.

En el gráfico de la Figura 50 se muestran los resultados de las primeras cuatro preguntas del cuestionario de satisfacción realizadas a los niños de ambos talleres (Brasil y Costa Rica), y

en el gráfico de la Figura 51 se muestra las categorías de los dibujos realizados por los niños que forma parte de la pregunta 5 de dicho cuestionario. En la pregunta 5, las categorías se realizaron según los contenidos abordados en los talleres y las descripciones dadas por los niños de los dibujos que realizaron, hay dibujos que se encuentran en una o más categorías. En la Figura 51 se puede observar como el robot y la tablet con la aplicación son de las cosas que más gustaron a los niños de ambos talleres, y se aprecia que trabajar de forma colaborativa les agradó.

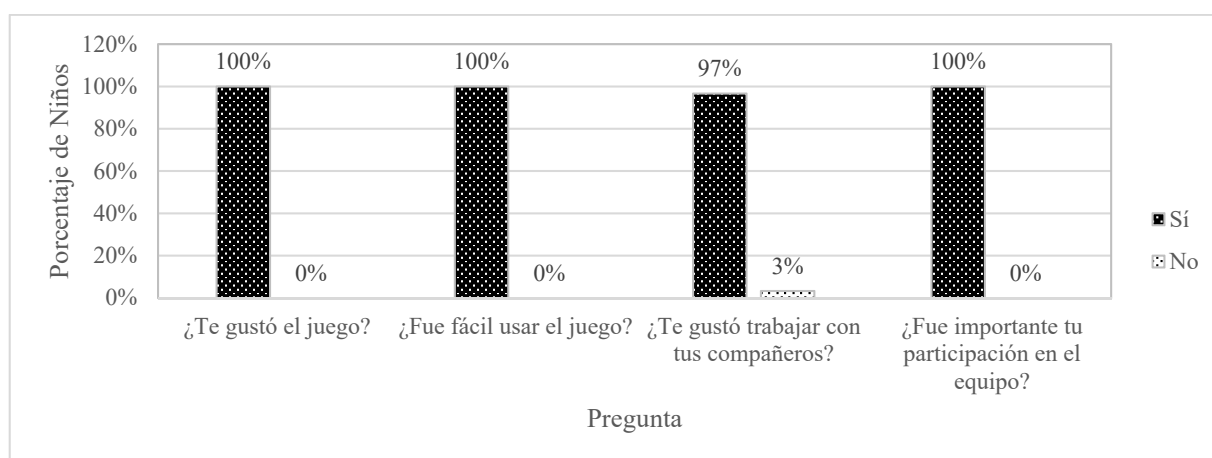


Figura 50. Resultados del cuestionario de satisfacción de TITIBOTS Colab.

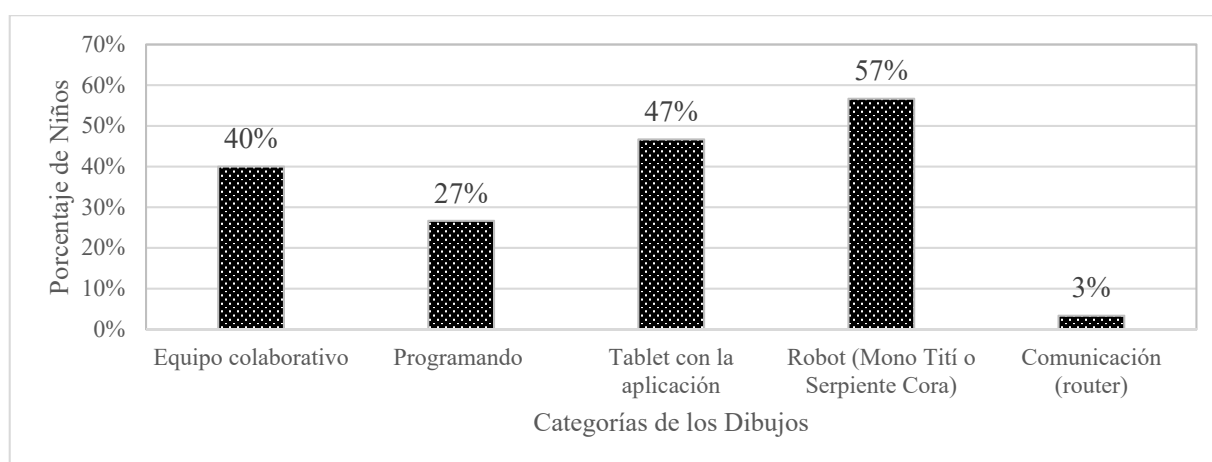


Figura 51. Categorías de los dibujos realizados por los niños.

Por otra parte, la interfaz de TITIBOTS Colab, al igual que su antecesor TITIBOTS, puede apoyar el proceso de lectura y escritura, por las características que posee en relación con este

aspecto, las cuales son valiosas y útiles para desarrollar habilidades del lenguaje y para mejorar la alfabetización en niños.

Dado los resultados de las evaluaciones realizadas en esta fase, se puede concluir que TITIBOTS Colab es fácil de usar para niños de 4 a 6 años. Los niños siempre estuvieron interesados, felices y atentos durante el uso de la herramienta, pero lo más importante: se divirtieron. Fue evidente que esta herramienta permite que los niños programen robots con el fin de resolver tareas específicas jugando de forma colaborativa. Además, al final de todas las pruebas, la herramienta quedó totalmente funcional; ya que en estas pruebas se ejecutaron las tareas especificadas en los requisitos de la herramienta (casos de uso), garantizando que hace lo que debe (casos positivos), que no falla (casos negativos) y que cumple con lo que se ha solicitado (satisface los requisitos).

En la siguiente sección, se muestra los resultados de la evaluación del aporte de la herramienta colaborativa en el proceso de aprendizaje de la programación en niños de preescolar.

7.2 Aporte en el Proceso de Aprendizaje de la Programación de la Primera Infancia

En las pruebas pilotos que se realizaron para evaluar las herramientas de programación, se realizaron varios desafíos de evaluación para comprobar que las herramientas apoyaran el aprendizaje de la programación en niños entre 4 y 6 años, donde los niños participantes aprendieron conceptos básicos de programación y robótica.

En cada uno de los desafíos de evaluación, tanto en el taller con TITIBOTS como los talleres con TITIBOTS Colab, se brindó tres intentos y un máximo de 30 minutos para lograr cada desafío. Los valores de la cantidad de intentos y el tiempo máximo por desafío fueron definidos por recomendación de los expertos de la FOD y de las docentes de preescolar que colaboraron en esta investigación.

En la prueba de usabilidad de desarrollo y uso en entornos reales de TITIBOTS (cuarta etapa), el taller fue de 4 días, el 67% de los desafíos de evaluación realizados por los niños fueron exitosos (intentos promedio de 2,44 y tiempo promedio de 19,9 minutos). Por otra parte, el promedio de errores fue de 3,38 y el de errores recuperados fue de 2,33 (69% del total de errores). En la Tabla 23 se muestra la cantidad de intentos requeridos por niño y el tiempo utilizado en cada desafío. Además, en la prueba de memoria, el 100% de los niños contestaron correctamente las preguntas relacionadas con conceptos de programación y robótica que fueron vistos en el taller (ver Apéndice 18).

En los pilotos de TITIBOTS Colab, el 100% de los grupos colaborativos, tanto de Brasil (el taller fue de 10 días) como Costa Rica (el taller fue de 7 días), realizaron con éxito los desafíos propuestos en un tiempo promedio menor a 15 minutos. Asimismo, en la prueba de memoria el 100% de los niños de Brasil y Costa Rica contestaron correctamente las preguntas relacionadas con conceptos de programación y robótica que fueron vistos en los talleres (ver Apéndice 19 y Apéndice 20).

Tabla 23. Cantidad de intentos y tiempo de los desafíos por niño en el taller con TITIBOTS.

Niños	Desafío 1	Desafío 2	Desafío 3	Promedio por Niño
Niña 1	(3+;30min)	(3+;30min)	(3+;30min)	(3+;30,00min)
Niña 2	(1;5min)	(3;10min)	(3;14min)	(2,33;9,67min)
Niño 1	(3;27min)	(3+;30min)	(3+;30min)	(3;29,00min)
Niño 2	(1;7min)	(3;18min)	(1;18min)	(1,67;14,33min)
Niña 3	(3+;30min)	(2;17min)	(2;23min)	(2,33;23,33min)
Niña 4	(3;11min)	(3;17min)	(1;12min)	(2,33;13,33min)
Promedio por Desafío	(2,33;18,33min)	(2,83;20,33min)	(2,17;21,17min)	(2,44;19,94min)

En la Tabla 24 y la Tabla 25 se muestran, por grupo colaborativo (GC), la cantidad de intentos requeridos y el tiempo utilizado, respectivamente, en cada desafío en el taller de Brasil. En las talas donde está “N/A” es porque los integrantes del grupo no asistieron ese día, por lo que no realizaron los desafíos. Se observa que en Brasil los grupos colaborativos realizaron una cantidad de intentos promedios de 1,17 y un tiempo promedio de 13,87. Por otra parte,

el promedio de errores fue de 0,80; de los cuales el 100% fueron recuperados. Ver Apéndice 19 para más detalle.

Tabla 24. Cantidad de intentos de los desafíos por grupo colaborativo en el taller de Brasil con TITIBOTS Colab.

Grupo Colaborativo	Desafío 1	Desafío 2	Desafío 3	Desafío 4	Desafío 5	Desafío 6	Promedio por GC
<i>Pássaro</i>	1	1	2	1	1	2	1,33
<i>Borboleta</i>	1	1	1	1	1	2	1,17
<i>Jacaré</i>	1	1	1	1	1	1	1,00
<i>Golfinho</i>	1	1	1	1	1	1	1,00
<i>Peixe</i>	N/A	N/A	N/A	1	1	2	1,33
Promedio por Desafío	1,00	1,00	1,25	1,00	1,00	1,60	1,17

Tabla 25. Tiempo de los desafíos por grupo colaborativo en el taller de Brasil con TITIBOTS Colab.

Grupo Colaborativo	Desafío 1	Desafío 2	Desafío 3	Desafío 4	Desafío 5	Desafío 6	Promedio por GC
<i>Pássaro</i>	5,00min	11,00min	23,00min	8,00min	16,00min	20,00min	13,83min
<i>Borboleta</i>	8,00min	12,00min	18,00min	7,00min	16,00min	25,00min	14,33min
<i>Jacaré</i>	6,00min	10,00min	18,00min	7,00min	15,00min	18,00min	12,33min
<i>Golfinho</i>	8,00min	13,00min	20,00min	6,00min	14,00min	18,00min	13,17min
<i>Peixe</i>	N/A	N/A	N/A	9,00min	17,00min	21,00min	15,67min
Promedio por Desafío	6,75min	11,50min	19,75min	7,40min	15,60min	20,40min	13,87min

En la Tabla 26 y la Tabla 27 se muestran, por grupo colaborativo (GC), la cantidad de intentos requeridos y el tiempo utilizado, respectivamente, en cada desafío en el taller de Costa Rica. Se observa que en Costa Rica los grupos colaborativos realizan una cantidad de intentos promedios de 1,47 y un tiempo promedio de 8,33. Por otra parte, el promedio de errores fue de 2,80; de los cuales el 100% fueron recuperados. Ver Apéndice 20 para más detalle.

Como se puede observar, los niños de los talleres donde se utilizó la herramienta colaborativa lograron todos los desafíos de evaluación realizando una cantidad de intentos promedio y tiempo promedio menor en relación con los talleres donde se utilizó la herramienta individual. Asimismo, se disminuyó el promedio de errores y todos los errores fueron

recuperados por los niños. En síntesis, los resultados obtenidos con TITIBOTS, en relación con las variables, son inferiores a los resultados de TITIBOTS Colab.

Tabla 26. Cantidad de intentos de los desafíos por grupo colaborativo en el taller de Costa Rica con TITIBOTS Colab.

Grupo Colaborativo	Desafío 1	Desafío 2	Desafío 3	Desafío 4	Desafío 5	Desafío 6	Promedio por GC
Mono	1	1	2	1	2	2	1,50
Tortuga	2	1	1	1	1	1	1,17
Delfín	1	2	3	1	1	3	1,83
Pájaro	1	1	1	1	1	1	1,00
Cocodrilo	2	3	2	1	1	2	1,83
Promedio por Desafío	1,40	1,60	1,80	1,00	1,20	1,80	1,47

Tabla 27. Tiempo de los desafíos por grupo colaborativo en el taller de Costa Rica con TITIBOTS Colab.

Grupo Colaborativo	Desafío 1	Desafío 2	Desafío 3	Desafío 4	Desafío 5	Desafío 6	Promedio por GC
Mono	5,00min	6,00min	10,00min	3,00min	7,00min	12,00min	7,17min
Tortuga	7,00min	7,00min	8,00min	2,00min	4,00min	11,00min	6,50min
Delfín	4,00min	11,00min	20,00min	3,00min	5,00min	22,00min	10,83min
Pájaro	5,00min	7,00min	7,00min	2,00min	4,00min	10,00min	5,83min
Cocodrilo	10,00min	23,00min	12,00min	4,00min	6,00min	13,00min	11,33min
Promedio por Desafío	6,20min	10,80min	11,40min	2,80min	5,20min	13,60min	8,33min

Con base en estos resultados, puede ser que el elemento colaboración influyó para lograr los buenos resultados de los desafíos de los dos últimos pilotos, dado los principios y los beneficios de trabajar de forma colaborativa que se encuentran en la literatura. Pero, no se puede dejar de lado otro elemento que pudo influir en los resultados: la duración del taller. Por estas razones, se podría plantear un estudio para comprobar si el aprendizaje colaborativo influye en un mejor aprendizaje de la programación.

Además, los expertos consideraron la aplicación de estos talleres fue exitosa, porque observaron que todos los participantes lograron el conocimiento base propuesto. Además, la docente y los observadores consideraron que los eventos en los que cada niño imitaba al robot y actuaba los comandos, fueron cruciales para el proceso de aprendizaje.

Los resultados obtenidos de las observaciones de los expertos y las pruebas de memoria, muestran que las dos herramientas de programación creadas apoyan el proceso de aprendizaje de la programación en niños de la primera infancia. Los niños entre 4 y 6 años pueden aprender a programar robots, obteniendo conocimientos básicos en las áreas de programación y robótica.

En la siguiente sección, se muestra los resultados de la evaluación del aporte de la herramienta colaborativa en el proceso de colaboración en niños de preescolar.

7.3 Aporte en el Proceso de Colaboración de la Primera Infancia

El aprendizaje colaborativo no se puede producir de forma aislada, sino con una actuación dinámica e interactiva de los individuos (Williams & Burden, 1999). En esta perspectiva el concepto de aprendizaje colaborativo toma importancia, ya que se trata de un grupo de personas que trabajan juntos para lograr los objetivos de aprendizaje en común (Nunan, 1993).

En esta investigación el objetivo principal fue evaluar el impacto de una herramienta colaborativa de programación en el proceso de colaboración en niños de la primera infancia. Para esto, se diseñó un cuasi-experimento en dos escuelas de Costa Rica para poder valorar el aporte de la herramienta colaborativa de programación TITIBOTS Colab en el proceso de colaboración en niños con edades entre 4 y 6 años.

Dado esto, en el cuasi-experimento se realizó una evaluación mediante la medición del desempeño del grupo, en términos de las acciones esperadas para alcanzar cada componente esencial del aprendizaje colaborativo. En la Sección 5.3 se describieron los elementos del diseño del cuasi-experimento (ver Figura 18). Además, la evaluación se complementó con las medidas de desempeño de la ejecución de las actividades colaborativas y las observaciones finales solicitadas a los evaluadores.

Con este cuasi-experimento se desea contestar la cuarta pregunta de investigación:

¿Es posible que una herramienta colaborativa de programación pueda incentivar la colaboración en niños con edades entre 4 y 6 años?

Al establecer la pregunta de investigación y el diseño del cuasi-experimento, se desea comprobar la hipótesis de la investigación:

Hay diferencia significativa en el porcentaje de acciones logradas del componente esencial X, al realizar un pre-test y un pos-test, en los dos grupos naturales

En la Sección 7.3.1 se describe el perfil de los niños, las docentes y las instituciones participantes en el cuasi-experimento. En la Sección 7.3.2 se muestra el proceso llevado a cabo en relación con la medición del desempeño de los grupos colaborativos y en la Sección 7.3.3 en relación con la medición del desempeño de las actividades colaborativas.

7.3.1 Perfil de los Participantes

El cuasi-experimento se realizó con dos grupos naturales de escuelas de Costa Rica, el grupo de control fue la Escuela El Castillo (EC) y el grupo experimental la Escuela Santa Mónica (SM). La información de los niños participantes del cuasi-experimento se muestra en la Tabla 28 y la información de las docentes participantes se muestra en la Tabla 29. Además, participaron 8 evaluadores (observadores): 4 mujeres y 4 hombres. En la Figura 52 se muestra los gráficos de las preguntas 1-4 y 6, y las Figuras 52 y 53 los gráficos de las preguntas 5 y 7 (respectivamente) con la información del perfil de los usuarios obtenidos mediante entrevistas semiestructuradas.

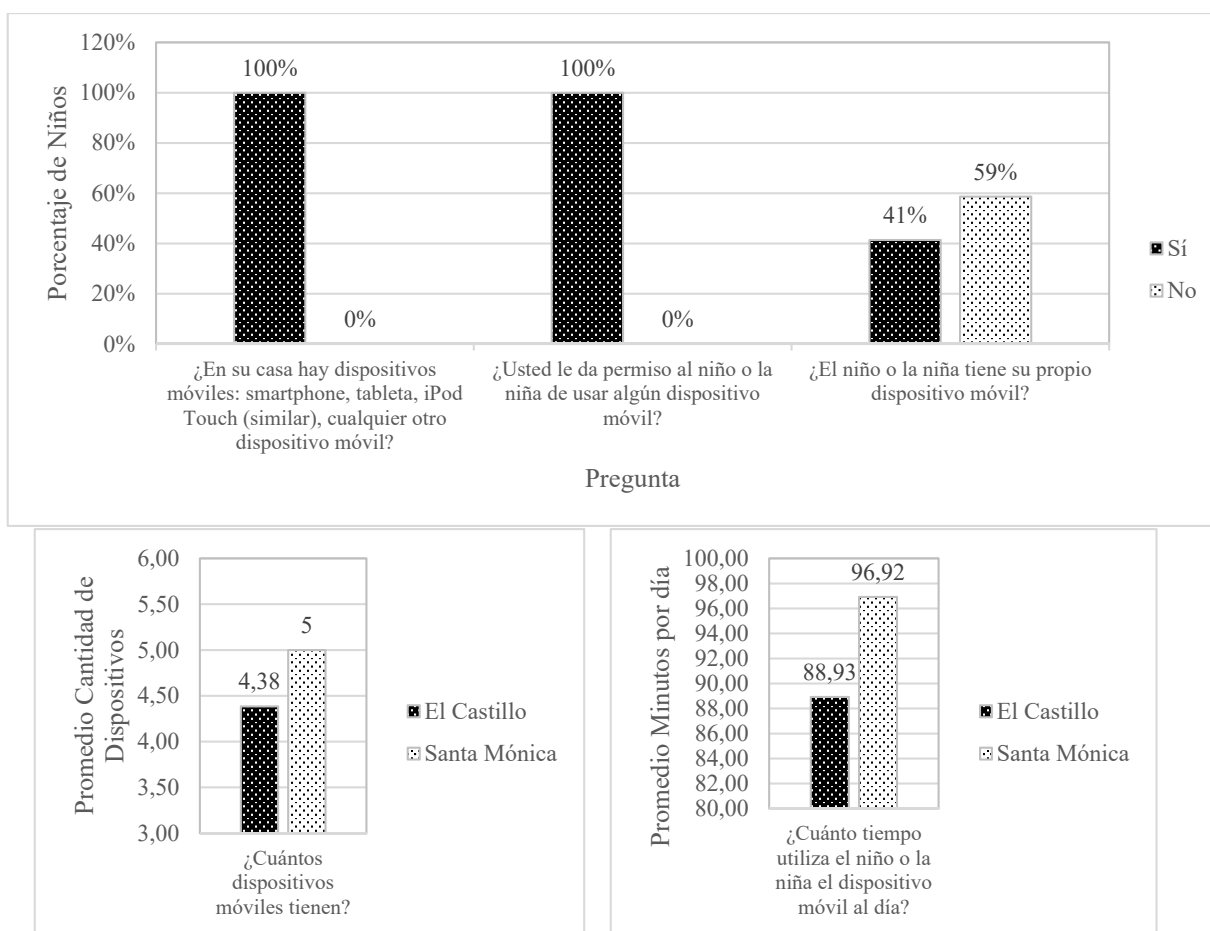
Tabla 28. Niños participantes del cuasi-experimento.

Escuela	Número de Participantes	Sexo		Edad	
		Femenino	Masculino	4	5
Grupo de control	15 (52%)	9 (60%)	6 (40%)	6 (40%)	9 (60%)
Grupo experimental	14 (48%)	7 (50%)	7 (50%)	2 (14%)	12 (86%)
Total (Porcentaje)	29 (100%)	16 (55%)	13 (45%)	8 (28%)	21 (72%)

Tabla 29. Docentes participantes del cuasi-experimento.

Escuela	Nombre	Sexo	Edad	Años de Servicio	Nivel
Grupo de control	Melissa Ávila Morera	F	25	5	Kínder
	Melissa Monge Quesada	F	28	5	Pre-Kínder
Grupo experimental	Gabriela Cortés Ramos	F	51	32	Kínder
	Vanessa Rojas Vásquez	F	33	1	Pre-Kínder

En la Figura 52 se observa que en el 100% de los hogares cuentan con dispositivos móviles. El 100% de los encargados de los niños les permiten usar algún dispositivo móvil. Los niños usan los dispositivos móviles para juegos educativos y aplicaciones creativas en su mayoría (ver Figura 53). Por su parte, el 41% de los niños tienen su propio dispositivo, siendo la tableta el tipo de dispositivo escogido para los niños (ver Figura 54).

**Figura 52.** Información del perfil del usuario del cuasi-experimento.

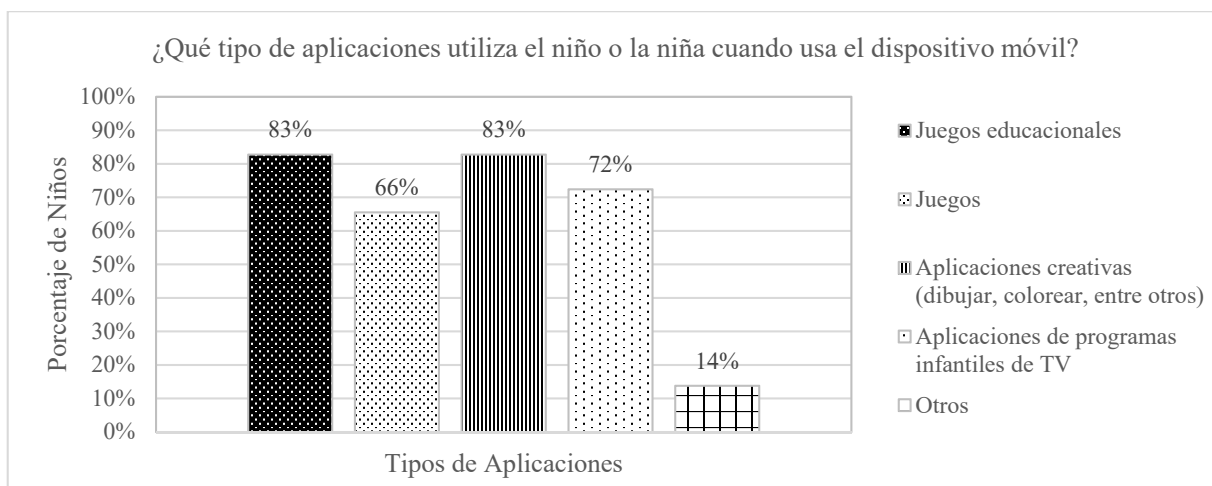


Figura 53. Información del perfil del usuario del cuasi-experimento – Pregunta 5.

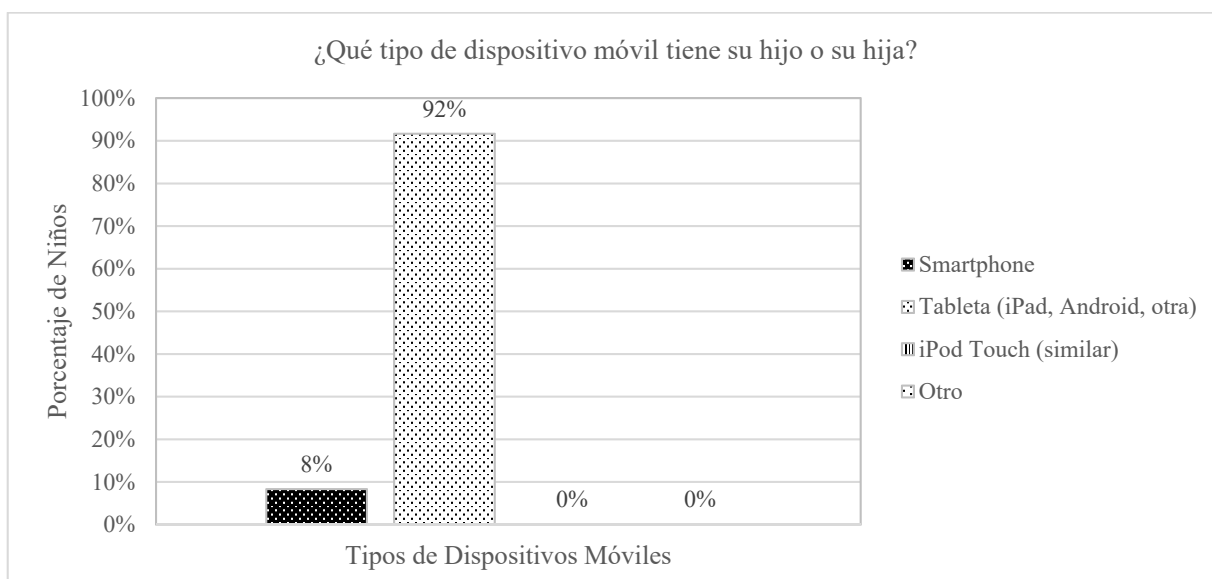


Figura 54. Información del perfil del usuario del cuasi-experimento – Pregunta 7.

En la Tabla 30 se muestra la información del perfil académico y laboral de las docentes participantes del cuasi-experimento.

En la siguiente sección se muestran el proceso llevado a cabo en la medición del desempeño de los grupos colaborativos.

Tabla 30. Perfil de las docentes participantes del cuasi-experimento.

Nº	Pregunta Descripción	Escuela			
		Grupo de Control (El Castillo)		Grupo Experimental (Santa Mónica)	
		Docente 1	Docente 2	Docente 1	Docente 2
1	¿Cuál es la misión de la Institución?	Brindar un servicio integral y multifacético para la estimulación del niño y la niña en sus primeros pasos de aprendizaje		Brindar una educación integral a nuestros alumnos, ofreciendo igualdad de oportunidades	
2	¿Cuál es la visión de la Institución?	Estar anuentes y abiertos a los cambios tecnológicos, de salud, pedagógicos y sociales que se vayan presentando para adaptarlas e implementarlas en la Institución; sin dejar de lado la filosofía, los valores morales y sus principios y valores		Proporcionar tranquilidad y seguridad a los padres de familia, sabiendo que sus hijos están siendo atendidos por personal altamente calificado	
3	¿Cuál fue la Universidad en la que se graduó?	Universidad de las Ciencias y el Arte	Universidad de las Ciencias y el Arte	Universidad Hispanoamericana	Universidad Hispanoamericana
4	¿Cuál fue el último título universitario obtenido?	Bachillerato (PT5)	Licenciatura (PT6)	Maestría en Administración Educativa (PT6)	Bachiller (PT5)
5	¿Cuántos años de experiencia tiene como docente?	18	5	32	1
6	¿Cuántos años imparte clases en los niveles de preescolar?	9	5	32	1
7	¿Cuántos años tiene de laborar en la Institución como docente?	4	4	15	1
8	¿Cuál es la metodología utilizada en preescolar de la Institución?	Aprender jugando, inteligencias múltiples		Jugando, trabajo favoreciendo las inteligencias múltiples	
9	¿Cuántos años lleva usando la metodología de trabajo?	9	5	15	1
10	¿Cómo es el planeamiento didáctico de la Institución?	Por medio de una minuta, tomando en cuenta las necesidades del niño y los lineamientos del MEP		Es un planeamiento integral tomando en cuenta las necesidades del niño y retomando los lineamientos del MEP y los propios de nuestra institución	
11	¿Cómo es la evaluación que se realiza en preescolar de la Institución?	Lista de cotejo, observación, evaluación cualitativa		Lista de cotejo, observación, evaluación cualitativa	
12	¿Realiza en el aula actividades colaborativas? Si la respuesta es afirmativa: ¿Qué actividades colaborativas realiza en el aula?	Sí. Las actividades que realizamos son: realizar carteles en grupo, juegos, manualidades, etc.		Sí. Las actividades que realizamos son: realizar carteles en grupo, juegos, decoraciones, etc.	

7.3.2 Medición del Desempeño de los Grupos Colaborativos

En relación con el desempeño de los grupos colaborativos, se calculó el porcentaje de acciones logradas de las muestras: individuales y grupos colaborativos de los dos grupos

naturales. En primer lugar, a cada estudiante se le calculó el porcentaje de acciones logradas por componente esencial (las cinco variables dependientes) en los dos tests. Luego, si todos los miembros de cada grupo lograban la acción, al grupo se le ponía lograda la acción. Seguidamente, a cada grupo se le calculó el porcentaje de acciones logradas en cada componente esencial (las cinco variables dependientes) en los dos test. En la Tabla 31 y Tabla 32 se muestran los porcentajes de acciones logradas de la muestra de los grupos colaborativos, y en la Tabla 33 y Tabla 34 se muestran los datos individuales (cada uno de los niños).

Dado que se tienen cinco componentes esenciales del aprendizaje colaborativo, se tienen cinco variables dependientes (VD). Por lo que se tendrá un porcentaje de acciones logradas para cada componente esencial, representados de la siguiente manera:

1. AC1: Independencia positiva.
2. AC2: Interacción estimuladora (comunicación cara a cara).
3. AC3: Responsabilidad individual y grupal.
4. AC4: Habilidades interpersonales y grupales (trabajo en equipo).
5. AC5: Evaluación grupal (proceso de grupo).

Tabla 31. Porcentaje de acciones logradas de los grupos colaborativos en el pre-test.

Grupo	Grupo de Control					Grupo Experimental				
	AC1	AC2	AC3	AC4	AC5	AC1	AC2	AC3	AC4	AC5
Mono	27%	33%	33%	33%	27%	33%	25%	25%	14%	13%
Tortuga	42%	29%	42%	43%	13%	27%	13%	33%	33%	20%
Delfín	52%	38%	25%	33%	7%	46%	25%	42%	29%	20%
Pájaro	39%	25%	33%	29%	27%	61%	33%	58%	57%	27%
Media μ	40%	31%	33%	35%	18%	42%	24%	40%	33%	20%

Tabla 32. Porcentaje de acciones logradas de los grupos colaborativos en el pos-test.

Grupo	Grupo de Control					Grupo Experimental				
	AC1	AC2	AC3	AC4	AC5	AC1	AC2	AC3	AC4	AC5
Mono	42%	33%	39%	48%	27%	100%	100%	100%	95%	100%
Tortuga	49%	33%	42%	52%	33%	94%	88%	83%	95%	100%
Delfín	33%	46%	13%	57%	7%	94%	83%	83%	86%	100%
Pájaro	39%	29%	33%	29%	40%	100%	100%	100%	100%	100%
Media μ	41%	35%	32%	46%	27%	97%	93%	92%	94%	100%

Tabla 33. Porcentaje de acciones logradas de los niños de preescolar en el pre-test.

Estudiante	Grupo de Control					Grupo Experimental				
	AC1	AC2	AC3	AC4	AC5	AC1	AC2	AC3	AC4	AC5
1	30%	33%	33%	33%	27%	52%	33%	33%	43%	27%
2	27%	38%	50%	48%	33%	39%	29%	25%	14%	13%
3	36%	33%	33%	38%	33%	33%	25%	33%	24%	13%
4	45%	38%	58%	52%	27%	27%	29%	50%	33%	47%
5	45%	29%	50%	52%	13%	39%	33%	42%	43%	47%
6	42%	38%	42%	43%	27%	42%	13%	33%	48%	20%
7	58%	54%	33%	33%	33%	61%	46%	58%	52%	67%
8	58%	38%	25%	52%	17%	55%	54%	58%	52%	27%
9	52%	42%	42%	43%	7%	46%	25%	42%	29%	20%
10	39%	29%	33%	33%	33%	64%	67%	58%	57%	40%
11	39%	29%	42%	29%	27%	61%	33%	58%	57%	27%
12	39%	25%	42%	33%	27%	82%	83%	67%	67%	53%
Media μ	43%	35%	40%	41%	25%	50%	39%	46%	43%	33%

Tabla 34. Porcentaje de acciones logradas de los niños de preescolar en el pos-test.

Estudiante	Grupo de Control					Grupo Experimental				
	AC1	AC2	AC3	AC4	AC5	AC1	AC2	AC3	AC4	AC5
1	42%	46%	58%	48%	27%	100%	100%	100%	100%	100%
2	45%	38%	50%	52%	27%	100%	100%	100%	100%	100%
3	45%	33%	39%	53%	33%	100%	100%	100%	95%	100%
4	49%	42%	42%	52%	40%	100%	100%	100%	100%	100%
5	49%	38%	50%	57%	47%	94%	96%	92%	95%	100%
6	58%	33%	50%	52%	33%	97%	88%	83%	100%	100%
7	33%	50%	46%	57%	33%	100%	100%	100%	100%	100%
8	38%	46%	48%	67%	17%	94%	83%	83%	86%	100%
9	42%	46%	50%	57%	7%	100%	94%	92%	100%	100%
10	42%	29%	42%	29%	53%	100%	100%	100%	100%	100%
11	39%	33%	33%	29%	47%	100%	100%	100%	100%	100%
12	42%	29%	50%	29%	40%	100%	100%	100%	100%	100%
Media μ	44%	39%	47%	48%	34%	99%	97%	96%	98%	100%

A partir de los datos de las tablas, se graficaron los promedios de los porcentajes de acciones logradas de cada componente esencial del aprendizaje colaborativo para la muestra de los grupos colaborativos y la muestra de estudiantes, con el objetivo de visualizar los datos. Como se puede observar en la Figura 55 y la Figura 56, en el pre-test las muestras de los dos grupos naturales tienen promedios similares, y en el pos-test los promedios de las muestras del grupo experimental difieren de las muestras del grupo de control, donde los promedios de los porcentajes de acciones logradas de cada componente esencial de las muestras del grupo experimental aumentaron.

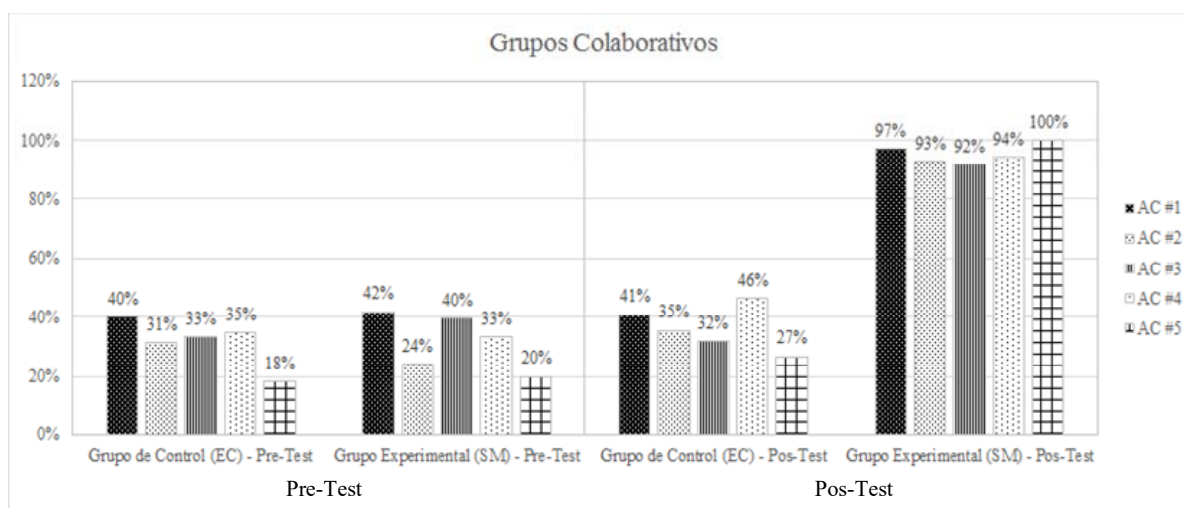


Figura 55. Promedios de los porcentajes de acciones logradas de cada componente esencial del aprendizaje colaborativo de los dos test para la muestra de grupos colaborativos.

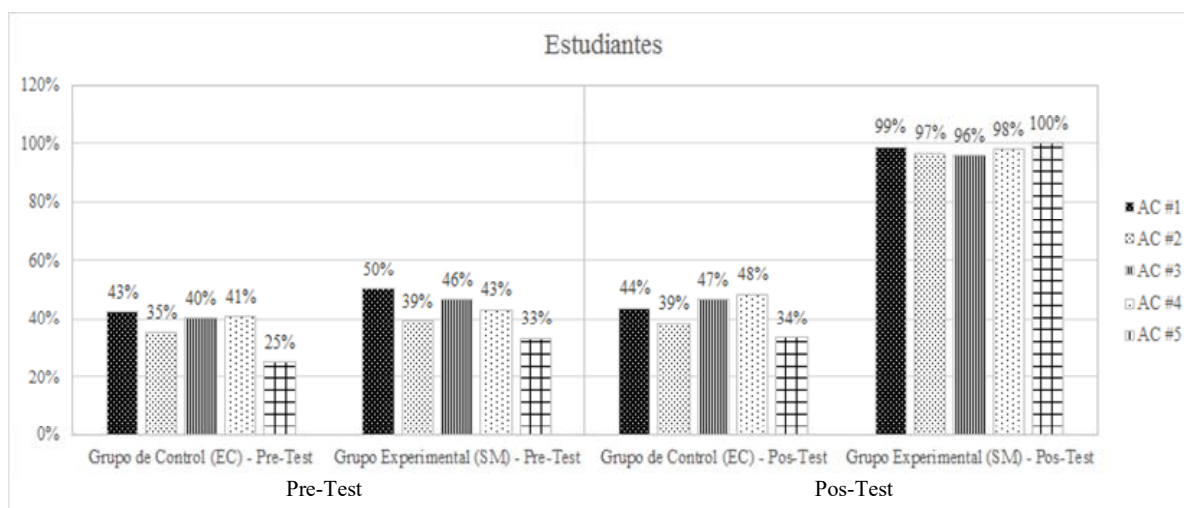


Figura 56. Promedios de los porcentajes de acciones logradas de cada componente esencial del aprendizaje colaborativo de los dos test para la muestra de estudiantes.

Seguidamente, se comienzan a realizar las pruebas estadísticas correspondientes para ver si la diferencia del grupo experimental en el pos-test que se muestra en los gráficos anteriores es significativa. En primer lugar, se realizaron diez pruebas-*t* con muestras independientes de las puntuaciones diferenciales para comprobar si existe una diferencia estadísticamente significativa entre ambos grupos naturales: En el Apéndice 21 se muestran con detalle los

resultados de las pruebas. Las primeras cinco pruebas-*t* se realizaron con los valores de las variables dependientes de los grupos colaborativos y las otras cinco con los valores de las variables dependientes de los estudiantes. En estas pruebas se esperaba que las medias de los valores de las VD fueran diferentes (diferencia estadísticamente significativa) como resultado de la aplicación del tratamiento en el grupo experimental. Las pruebas-*t* se pudieron realizar dado que los requisitos y las suposiciones de la prueba se cumplieron (ver Apéndice 21 para más detalle).

Luego, por contar con muestras pequeñas (tanto grupos colaborativos como estudiantes), se realizaron diez pruebas *U* de Mann-Whitney con muestras independientes de las puntuaciones diferenciales, para comprobar si existe una diferencia estadísticamente significativa entre las distribuciones de ambos grupos naturales. En el Apéndice 21 se muestra con detalle los resultados de las pruebas. Las primeras cinco pruebas se realizaron con los valores de las variables dependientes de los grupos colaborativos y las otras cinco con los valores de las variables dependientes de los estudiantes. En estas pruebas se esperaba que existiera una diferencia en la distribución de los valores de las VD (diferencia estadísticamente significativa) como resultado de la aplicación del tratamiento en el grupo experimental. Las pruebas *U* de Mann-Whitney se pudieron realizar dado que los requisitos y las suposiciones de la prueba se cumplieron (ver Apéndice 21 para más detalle).

En la Tabla 35 y la Tabla 36, se presenta un resumen de las pruebas de hipótesis realizadas para el contraste de medias de los grupos naturales. En estas tablas se presentan las medias de las puntuaciones diferenciales (PD) de cada componente esencial del aprendizaje colaborativo para los dos grupos naturales. Si el valor de la media es positivo significa que ocurrió un aumento en el porcentaje de acciones logradas en el pos-test; en caso contrario, que ocurrió un decremento en el porcentaje de acciones logradas en el pos-test.

Como se puede observar en la Tabla 35 y la Tabla 36, hay una diferencia estadísticamente significativa entre las medias y las distribuciones de ambos grupos naturales al realizar el análisis de contraste de medias en las puntuaciones diferenciales en ambas muestras: por grupo y por individuo, ya que las probabilidades $p < 0,05$. Por esto, controlando las variables,

se puede mostrar que la diferencia que hubo en ambos grupos naturales fue la aplicación del tratamiento en el grupo experimental, dado que se comprobó la comparabilidad inicial de ambos grupos. Esto comprueba que la utilización de TITIBOTS Colab incentiva la colaboración en niños con edades entre 4 y 6 años (ver Apéndice 21 para más detalles).

Finalmente, se calculó el tamaño del efecto usando la fórmula recomendada por Glass, McGaw y Smith para realizar el análisis de la diferencia tipificada (diferencia de medias estandarizada) en pruebas de contrastes de medias de dos muestras independientes (experimental y de control) con pre-test y pos-test (Glass et al., 1981). Se usa la categorización del valor d de Cohen (Cohen, 1988), que es la utilizada en el área de educación (Pérez Juste et al., 2012). Al mismo tiempo, se transformó el tamaño del efecto en un coeficiente de correlación, útil por el tipo de estudio estadístico que se realiza en esta investigación, el cual se interpreta como el porcentaje de niños que mejoraron el porcentaje de acciones logradas de cada componente esencial del aprendizaje colaborativo por la aplicación del tratamiento (Cohen, 1988, p. 533).

Tabla 35. Resumen de pruebas de hipótesis para el contraste de medias de la muestra de los grupos colaborativos.

Tipo de Prueba	Prueba	Variable Dependiente	Hipótesis Nula (H_0)	μ_1 : Media de las PD del grupo de control (EC)	μ_2 : Media de las PD del grupo experimental (SM)	Valor p (Sig.)	Decisión
Paramétrica	Prueba- t : Muestras Independientes	AC1: Independencia positiva	Las medias de las puntuaciones diferenciales del componente esencial X de los dos grupos naturales son iguales	0,50	55,25	0,002	Rechazar H_0
		AC2: Interacción estimuladora		4,00	68,75	0,000	Rechazar H_0
		AC3: Responsabilidad individual y grupal		-1,50	52,25	0,001	Rechazar H_0
		AC4: Habilidades interpersonales y grupales		12,00	60,75	0,002	Rechazar H_0
		AC5: Evaluación grupal		8,25	80,00	0,000	Rechazar H_0
No paramétrica	Prueba U de Mann-Whitney	AC1: Independencia positiva	Las distribuciones de las puntuaciones diferenciales del componente esencial X de los dos grupos naturales son iguales	0,50	55,25	0,029	Rechazar H_0
		AC2: Interacción estimuladora		4,00	68,75	0,029	Rechazar H_0
		AC3: Responsabilidad individual y grupal		-1,50	52,25	0,029	Rechazar H_0
		AC4: Habilidades interpersonales y grupales		12,00	60,75	0,029	Rechazar H_0
		AC5: Evaluación grupal		8,25	80,00	0,029	Rechazar H_0

Tabla 36. Resumen de pruebas de hipótesis para el contraste de medias de la muestra de los estudiantes.

Tipo de Prueba	Prueba	Variable Dependiente	Hipótesis Nula (H_0)	μ_1 : Media de las PD del grupo de control (EC)	μ_2 : Media de las PD del grupo experimental (SM)	Valor p (Sig.)	Decisión
Paramétrica	Prueba- t : Muestras Independientes	AC1: Independencia positiva	Las medias de las puntuaciones diferenciales del componente esencial X de los dos grupos naturales son iguales	1,17	48,67	0,000	Rechazar H_0
		AC2: Interacción estimuladora		3,08	57,58	0,000	Rechazar H_0
		AC3: Responsabilidad individual y grupal		6,25	49,42	0,000	Rechazar H_0
		AC4: Habilidades interpersonales y grupales		7,58	54,75	0,000	Rechazar H_0
		AC5: Evaluación grupal		8,25	66,58	0,000	Rechazar H_0
No paramétrica	Prueba U de Mann-Whitney	AC1: Independencia positiva	Las distribuciones de las puntuaciones diferenciales del componente esencial X de los dos grupos naturales son iguales	1,17	48,67	0,000	Rechazar H_0
		AC2: Interacción estimuladora		3,08	57,58	0,000	Rechazar H_0
		AC3: Responsabilidad individual y grupal		6,25	49,42	0,000	Rechazar H_0
		AC4: Habilidades interpersonales y grupales		7,58	54,75	0,000	Rechazar H_0
		AC5: Evaluación grupal		8,25	66,58	0,000	Rechazar H_0

En la Tabla 37 y la Tabla 38 se muestra el tamaño del efecto y el coeficiente de correlación para los grupos colaborativos y los estudiantes, respectivamente. En la Figura 57 y Figura 58 se muestran los gráficos de los tamaños del efecto de las tablas anteriores. El tamaño del efecto se calculó de las diferencias del grupo experimental respecto al grupo de control. El grupo experimental obtuvo mejores puntuaciones medias finales que el grupo de control en todos los componentes esenciales, además, son significativas y relevantes ($d > 0,8$, la VD posee una fuerza grande). Esto indica que el grupo experimental, al final, logró más acciones del proceso de colaboración que el grupo de control en todos los componentes esenciales del aprendizaje colaborativo. Se pudo observar en los coeficientes de correlación, que más del 89% de estudiantes del grupo experimental mejoraron en las acciones esperadas para alcanzar cada componente esencial del aprendizaje colaborativo por la utilización de la herramienta colaborativa de programación TITIBOTS Colab.

Tabla 37. Tamaño del efecto y coeficiente de correlación para el contraste de medias de la muestra de los grupos colaborativos.

Variable Dependiente	Tamaño del Efecto d	Fuerza	Coefficiente de Correlación r
AC1: Independencia positiva	8,46	Grande	0,97
AC2: Interacción estimuladora	7,76	Grande	0,97
AC3: Responsabilidad individual y grupal	4,58	Grande	0,92
AC4: Habilidades interpersonales y grupales	3,88	Grande	0,89
AC5: Evaluación grupal	5,23	Grande	0,93

Tabla 38. Tamaño del efecto y coeficiente de correlación para el contraste de medias de la muestra de los estudiantes.

Variable Dependiente	Tamaño del Efecto d	Fuerza	Coefficiente de Correlación r
AC1: Independencia positiva	8,68	Grande	0,97
AC2: Interacción estimuladora	8,09	Grande	0,97
AC3: Responsabilidad individual y grupal	7,46	Grande	0,97
AC4: Habilidades interpersonales y grupales	3,87	Grande	0,89
AC5: Evaluación grupal	5,06	Grande	0,93

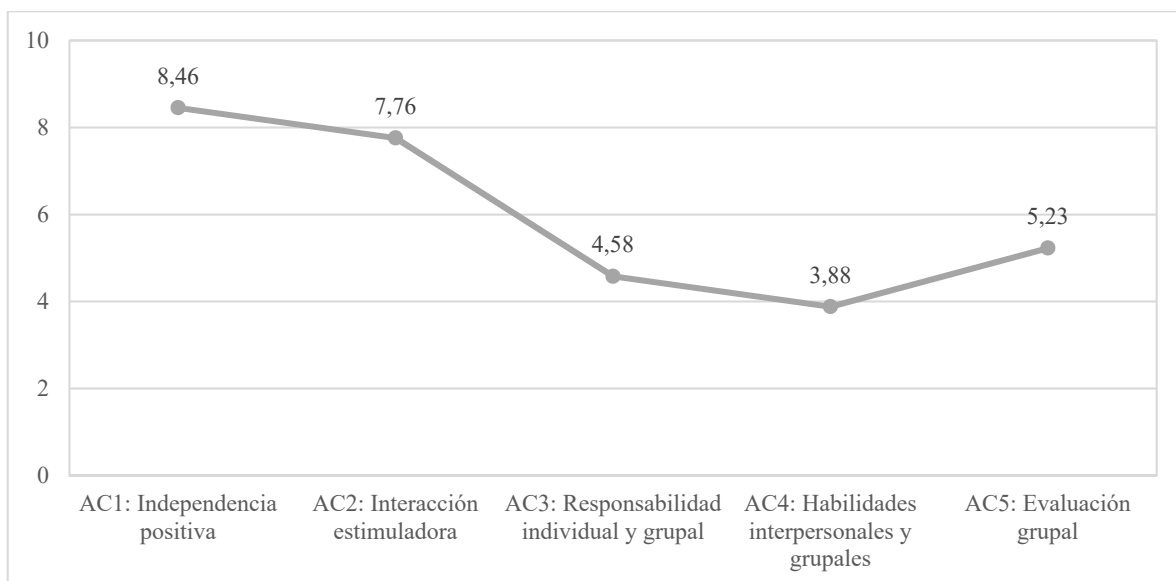


Figura 57. Tamaño del efecto para el contraste de medias de la muestra de los grupos colaborativos.

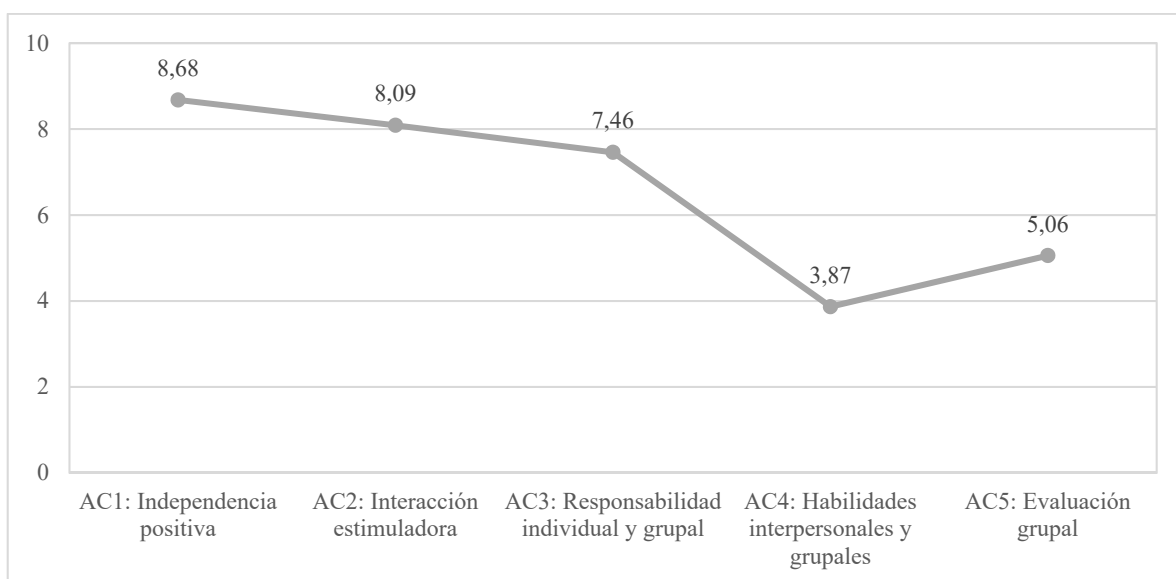


Figura 58. Tamaño del efecto para el contraste de medias de la muestra de los estudiantes.

En la Figura 59 y la Figura 60 se sintetiza los resultados, mostrando la medida del tamaño del efecto de las diferencias de las medias de los porcentajes de acciones logradas para cada componente esencial del AC (variables dependientes) entre el pre-test y el pos-test de ambos grupos naturales, comparando en cada componente esencial la magnitud del cambio de los dos grupos. Si el valor es positivo significa que ocurrió un aumento en el porcentaje de

acciones logradas en el pos-test; en caso contrario, ocurrió un decremento. Como se observa, la mejora lograda por el grupo experimental es superior a la del grupo control en todas las medias de los porcentajes de acciones logradas de cada componente esencial.

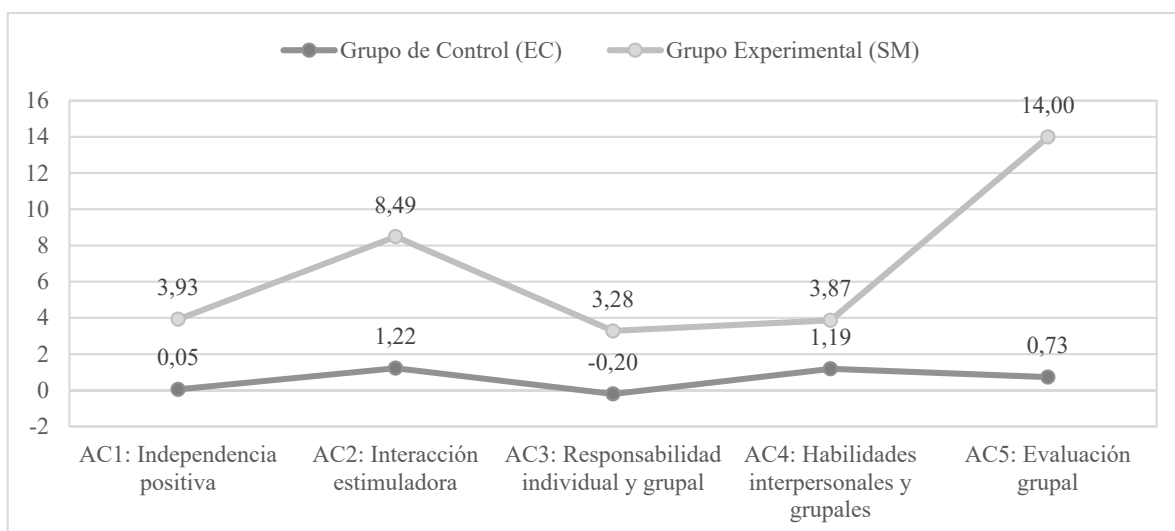


Figura 59. Tamaño del efecto de las diferencias para los porcentajes de acciones logradas entre el pre-test y el pos-test de la muestra de los grupos colaborativos.

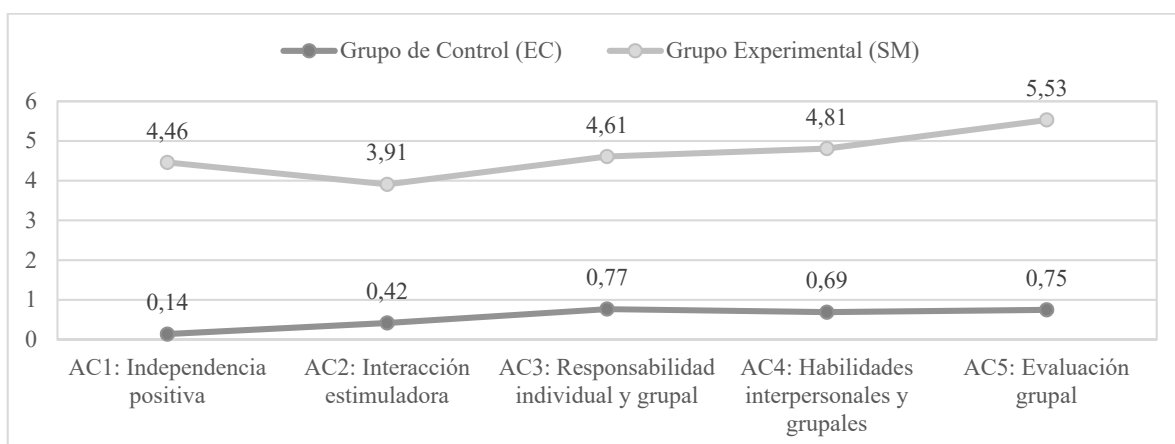


Figura 60. Tamaño del efecto de las diferencias para los porcentajes de acciones logradas entre el pre-test y el pos-test de la muestra de los estudiantes.

Con todas las pruebas realizadas, se comprobó que en las muestras de los grupos colaborativos y de los estudiantes de la Escuela Santa Mónica (grupo experimental) hubo una diferencia estadísticamente significativa en el pos-test en relación con las muestras de los

grupos colaborativos y de los estudiantes del grupo de control (Escuela El Castillo), donde ocurrió un aumento estadísticamente significativo en el grupo experimental (Escuela Santa Mónica) en el punto de tiempo pos-test. Por esto se concluye, con los resultados obtenidos en las poblaciones del cuasi-experimento, que la utilización de TITIBOTS Colab incentiva la colaboración en niños con edades entre 4 y 6 años.

En la siguiente sección se muestra la medición del desempeño de las actividades colaborativas.

7.3.3 Medición del Desempeño de las Actividades Colaborativas

La evaluación se complementó con las medidas de desempeño de la ejecución de las actividades colaborativas y las observaciones finales solicitadas a los evaluadores. En la Tabla 39 se muestra los resultados obtenidos en el pre-test en relación con estas medidas de desempeño. En la Tabla 40 los resultados del pos-test. Las observaciones finales de los evaluadores se muestran en el Anexo 23.

Estos resultados comprueban que, en el pos-test, el grupo experimental (Escuela Santa Mónica) logró con éxito, casi el 100%, la ejecución de las actividades colaborativas, mostrando cambios significativos con respecto al pre-test. Lo contrario sucede con el grupo de control (Escuela El Castillo), donde los resultados del pre-test y pos-test son similares.

Tabla 39. Medidas de desempeño de las actividades colaborativas en el pre-test.

Grupo	Grupo de Control (EC)			Grupo Experimental (SM)		
	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa
	1	2	3	1	2	3
Mono	No (5+, 20min)	Sí (3, 3min)	No (5+, 20min)	No (3, 20min)	Sí (3, 2min)	No (5+, 20min)
Tortuga	No (5+, 20min)	Sí (3, 7min)	No (5+, 20min)	Sí (3, 18min)	Sí (5, 4min)	No (5+, 20min)
Delfín	No (1, 20min)	Sí (4, 7min)	No (3, 20min)	No (3, 20min)	Sí (5+, 10min)	No (5+, 20min)
Pájaro	No (5+, 20min)	Sí (3, 20min)	No (5+, 20min)	Sí (1, 2min)	Sí (3, 2min)	No (5+, 20min)

Nota: Logro de la actividad colaborativa. En el paréntesis está la cantidad de intentos realizados y el tiempo.

Tabla 40. Medidas de desempeño de las actividades colaborativas en el pos-test.

Grupo	Grupo de Control (EC)			Grupo Experimental (SM)		
	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa	Actividad Colaborativa
	1	2	3	1	2	3
Mono	No (5, 10min)	Sí (1, 10s)	No (3, 10min)	Sí (2, 9min)	Sí (1, 1s)	Sí (2, 5min)
Tortuga	No (4, 10min)	Sí (1, 1s)	No (5+, 20min)	No (5+, 20min)	Sí (2, 1s)	Sí (1, 1min)
Delfín	No (5, 8min)	Sí (3, 1min)	No (5+, 20min)	Sí (2, 17min)	Sí (1, 1s)	Sí (2, 15min)
Pájaro	No (5+, 20min)	Sí (1, 4s)	No (5+, 20min)	Sí (2, 15min)	Sí (1, 1s)	Sí (1, 6min)

Nota: Logro de la actividad colaborativa. En el paréntesis está la cantidad de intentos realizados y el tiempo.

La actividad colaborativa 1 (realización de un patrón con bloques de construcción) fue lograda durante el pre-test por dos grupos colaborativos del grupo experimental (Escuela Santa Mónica). En ambos grupos, fue solo un niño quien resolvió el desafío del patrón en el pre-test. En cambio, en el pos-test, la actividad la lograron tres grupos de esa institución. El grupo que logró dicha actividad en ambos test, en el pos-test dura más tiempo, debido a que en el grupo había cohesión, se comunicaban y entre todos resolvieron el desafío (basado en las observaciones del evaluador).

La actividad colaborativa 2 (balance de un objeto) fue una actividad fácil de realizar por todos los grupos desde el pre-test. En el pre-test se duró más en realizarla, dado que los grupos tuvieron que organizarse para lograr el desafío y eso tomó tiempo.

Por otra parte, en la actividad colaborativa 3 (realización de un desafío programado con la herramienta) no fue lograda en el pre-test, y solo los grupos colaborativos del grupo experimental (Escuela Santa Mónica) la lograron en el pos-test.

En el capítulo siguiente se muestran las conclusiones finales de la investigación, así como los aportes adicionales y el trabajo futuro que se puede desprender de este trabajo.

CAPÍTULO 8. CONCLUSIONES

En este trabajo de investigación se propuso la evaluación del impacto de la herramienta colaborativa de programación TITIBOTS Colab en el proceso de colaboración en niños con edades entre 4 y 6 años.

La herramienta colaborativa de programación permitió a los niños de la primera infancia crear programas usando las tabletas y ejecutarlos con los robots (permitiendo usar cualquier robot donde se implemente un interpretador de los comandos). Esta herramienta fue implementada para dispositivos móviles para programar robots, con el fin de que en la niñez se promueva el desarrollo del pensamiento computacional desde edades tempranas a través de la programación de robots de forma colaborativa.

Uno de los mayores retos que se presentó en la investigación fue diseñar aplicaciones para niños de preescolar, ya que el diseño de las herramientas debía captar la atención visual de los niños y motivarlos y, al mismo tiempo, fáciles de usar por ellos. Esto se logró usando el proceso de diseño centrado en el usuario, el cual fue útil para alcanzar la usabilidad de las herramientas. Dado que se debió identificar y especificar el contexto de uso, y caracterizar el público meta. Luego, se realizó un proceso de refinamiento en relación con su diseño y funcionalidad, basado en los diferentes pilotos realizados. Finalmente fueron evaluadas en entornos reales. En este proceso de refinamiento se evaluó la usabilidad mediante prototipos y pruebas de usabilidad.

En este proceso se generó la guía de lineamientos para construir herramientas de programación orientadas a niños de preescolar. Esta guía se presentó en la Sección 6.2, a partir de diferentes estudios realizados por diversos autores y las evaluaciones realizadas en la investigación, cumpliendo los criterios del marco de trabajo *Cognitive Dimensions of Notations* (Alan F. Blackwell & Green, 2000; Green & Petre, 1996). Es importante destacar los siguientes lineamientos generados: elegir y diseñar un personaje, elaborar buenas gráficas y cierres visuales, tener un diseño gráfico simple, agradable y entretenido, facilidad en la

navegación e interacción, narrativa agradable y simple, comunicar instrucciones verbalmente y visualmente.

Las dos herramientas de programación creadas, TITIBOTS y TITIBOTS Colab, poseen un conjunto único de características:

- Poseen una interfaz de usuario simple, intuitiva y fácil de usar para los niños de la primera infancia.
- Están constituidas por símbolos iconográficos y sonidos tomando en cuenta el desarrollo cognitivo, personal, social y emocional de los pequeños, permitiendo ser usadas por niños que aún no han aprendido a leer y a escribir.
- Ofrecen un conjunto de comandos abiertos, permitiendo que se puedan usar con cualquier robot donde se implemente un interpretador de los comandos.
- Están disponibles en tres idiomas: español, portugués e inglés.

La interfaz de las herramientas fue diseñada para colocar los comandos de arriba a abajo y de izquierda a derecha. En la creación de un programa (un conjunto ordenado de acciones) los niños pueden asociar cada ícono con una palabra escrita y el programa con una oración. Cada palabra, o comando, es un reflejo del pensamiento, y los pensamientos se convierten en acciones. Las docentes y los expertos hicieron hincapié en que estas características son valiosas y útiles para el desarrollo de habilidades del lenguaje y para la mejora de la alfabetización de los niños.

Por otro lado, la evaluación de usabilidad de TITIBOTS fue exitosa, ya que las herramientas han sido intuitivas y fáciles de usar por el público meta. Los niños infirieron el significado de los comandos y usaron la aplicación sin mayor problema. Además, los niños siempre se mostraron motivados, interesados, felices y atentos durante la utilización de las herramientas en los diferentes pilotos desarrollados.

Asimismo, se comprueba que las herramientas de programación apoyan el proceso de aprendizaje de la programación en niños de la primera infancia, ya que es posible que los

niños aprendan los conceptos básicos de programación, tales como la resolución de problemas de forma secuencial, ya que los niños fueron capaces de verbalizar su pensamiento cuando se les pidió. Se piensa que el uso de ejercicios concretos y físicos con los niños facilita el uso de la herramienta desarrollada. Los niños que usaron la herramienta adquirieron una primera aproximación al proceso básico para resolver un problema, siguiendo los pasos de programación comunes: planificación, ejecución del programa (escritura) y prueba (enviar los comandos al robot). Las actividades de enseñanza-aprendizaje diseñadas fueron exitosas, llamaron la atención de los niños, y la consecución de los objetivos previstos para cada una. Con ello, los niños entre 4 y 6 años aprendieron fundamentos de programación y robótica, favoreciendo el desarrollo del pensamiento lógico, la abstracción, la solución de problemas y la colaboración.

Por su parte, en TITIBOTS Colab se automatizaron la mayoría de las actividades del proceso del aprendizaje colaborativo, brindando el proceso de colaboración mediante la distribución automática y equitativa de los recursos. En relación con la automatización de actividades de una teoría de educación (como lo es el aprendizaje colaborativo), es importante destacar que se deben automatizar las actividades que ayudan con el desarrollo del proceso a la docente y que tengan relevancia en el contexto de uso. Por otra parte, se debe mediar sin obstaculizar las actividades que involucran interacción y comunicación por parte de los participantes. En el protocolo de comunicación implementado en esta investigación se tomó en cuenta estos aspectos para lograr la interacción docente-estudiantes y estudiantes-estudiantes en la herramienta colaborativa de programación.

Con los resultados obtenidos en el cuasi-experimento, se comprobó que en el grupo experimental hubo una diferencia estadísticamente significativa en los componentes del aprendizaje colaborativo en el pos-test en relación con los grupos colaborativos del grupo de control. La diferencia que hubo entre los grupos naturales fue la aplicación del tratamiento en el grupo experimental, dado que se comprobó la comparabilidad inicial de ambos grupos. Por lo que, se concluye que la utilización de TITIBOTS Colab incentiva la colaboración en niños con edades entre 4 y 6 años.

Por otra parte, como se apunta en las conclusiones del reporte generado por las evaluadoras de PROINNOVA, todos los niños participantes del taller lograron trabajar de forma colaborativa y resaltaron dos situaciones (ver Apéndice 23). La primera situación fue la comunicación que se daba entre miembros de un equipo para dialogar sobre la solución de un desafío. Resaltan que este diálogo resulta sorprendente para niños de esta edad. La segunda situación fue el trabajo intergrupar que se suscitó, donde la situación competitiva se dejaba atrás y ciertos niños, cuando terminaban su aporte en el equipo, se iban a ayudarle a otros equipos. Las evaluadoras resaltan las fases observadas por las que los niños atraviesa durante su interacción con TITIBOTS Colab:

- **Fase 1.** El primer contacto con la herramienta colaborativa ocasiona una etapa de descubrimiento y experimentación.
- **Fase 2.** Cuando se dominaba un poco más la herramienta colaborativa, el sentimiento de colaboración dentro del equipo y competitivo entre equipos aparece.
- **Fase 3.** Al tener dominada la herramienta colaborativa se da el trabajo intergrupar, donde la situación competitiva desaparece.

La herramienta colaborativa de programación TITIBOTS Colab logró que los niños realizaran la mayoría de las acciones definidas en cada uno de los componentes esenciales del aprendizaje colaborativo; incentivando el proceso de colaboración intragrupal e intergrupar en los niños de preescolar, haciendo que dejen de lado su egocentrismo inicial, en un periodo corto de tiempo (una semana).

Por lo que se concluye, que a través de TITIBOTS Colab, se incentiva en los niños el desarrollo de habilidades blandas y técnicas que son necesarias para asegurar un futuro lleno de oportunidades y próspero para la niñez costarricense. Con ello, se pretende impactar de la mejor manera en la sociedad costarricense para despertar el interés de los niños en áreas STEM desde temprana edad, dado que la programación es considerada la futura alfabetización.

Finalmente, las contribuciones de esta tesis, en relación con las herramientas de programación TITIBOTS y TITIBOTS Colab, el proceso de diseño descrito, la guía de lineamientos generada y el protocolo de comunicación que automatizó la mayoría de las actividades del proceso del aprendizaje colaborativo, prometen tener un gran impacto en la mejora de la usabilidad en herramientas para el aprendizaje de la programación de niños de preescolar. Esperando que esta investigación inspire a los futuros desarrolladores a crear más aplicaciones móviles colaborativas y educativas para el beneficio de los niños de la primera infancia.

En las secciones siguientes se muestran otros aportes de la investigación y el trabajo futuro.

8.1 Aportes Adicionales

Con esta investigación se obtuvieron varios aportes importantes en las disciplinas de Ciencias de la Computación y Educación que convergen en el área de informática educativa y del aprendizaje de la programación.

Los tres aportes más importantes son:

- Diseño, implementación y evaluación de dos herramientas de programación en dispositivos móviles orientado a niños entre 4 y 6 años que permite programar aplicaciones para robots. Ambas ofrecen un protocolo de comunicación abierto para poder ser utilizadas con cualquier tipo de tecnología robótica, y una de ellas que automatiza diferentes actividades del aprendizaje colaborativo. La interfaz de usuario es simple, intuitiva y usable por los niños de la primera infancia, constituida por símbolos iconográficos y sonidos, tomando en cuenta el desarrollo cognitivo, personal, social y emocional de los pequeños, permitiendo ser usada por niños que aún no han aprendido a leer y a escribir.
- Propuesta de una estrategia didáctica para el aprendizaje de la programación y el desarrollo de habilidades de colaboración en niños entre 4 y 6 años usando las herramientas de programación creadas. La metodología de enseñanza-aprendizaje se

basa en la experimentación y el aprendizaje lúdico. El niño aprende jugando, planteando desafíos que los niños debe resolver a través de la programación del robot (ver Figura 61). Con el fin de desarrollar el pensamiento computacional, incentivando en los niños el desarrollo de habilidades blandas y técnicas desde temprana edad.

- Propuesta del proceso básico para la solución de un problema dado, siguiendo los pasos de programación comunes: planificación, ejecución del programa (escritura), y prueba (enviar los comandos al robot) (ver Figura 62).



Figura 61. Metodología de enseñanza-aprendizaje de los talleres.

Por otra parte, se realizaron varias alianzas estratégicas:

- Fundación Omar Dengo (FOD), mediante la Unidad de Aprendizaje lógico, científico y robótica, y el Programa Nacional de Informática Educativa MEP – FOD (PRONIE MEP – FOD). Con el fin de contribuir con la mejora de la calidad de la educación pública, a través de propuestas pedagógicas innovadoras apoyadas en las tecnologías digitales, concebidas como herramientas de aprendizaje. Los investigadores con los que se trabaja son: Ana Lourdes Acuña Zúñiga (Coordinadora de la Unidad), Hugo Esteban Arroyo Fallas, Jorge López Gamboa, Diana Quesada Hidalgo y Suhany Chavarría Artavia (PRONIE-MEP-FOD).
- Programa de Tecnologías Educativas Avanzadas (PROTEA), donde se trabajó con: MTE. Mónica Villalobos López (Coordinadora) y MTE. Karol Ríos Cortés (Coordinadora).

- RExLab-UFSC, donde se trabajó con los investigadores Dr. João Bosco da Mota Alves (Profesor – Investigador) y Dr. Juarez Bento da Silva (Profesor – Investigador).

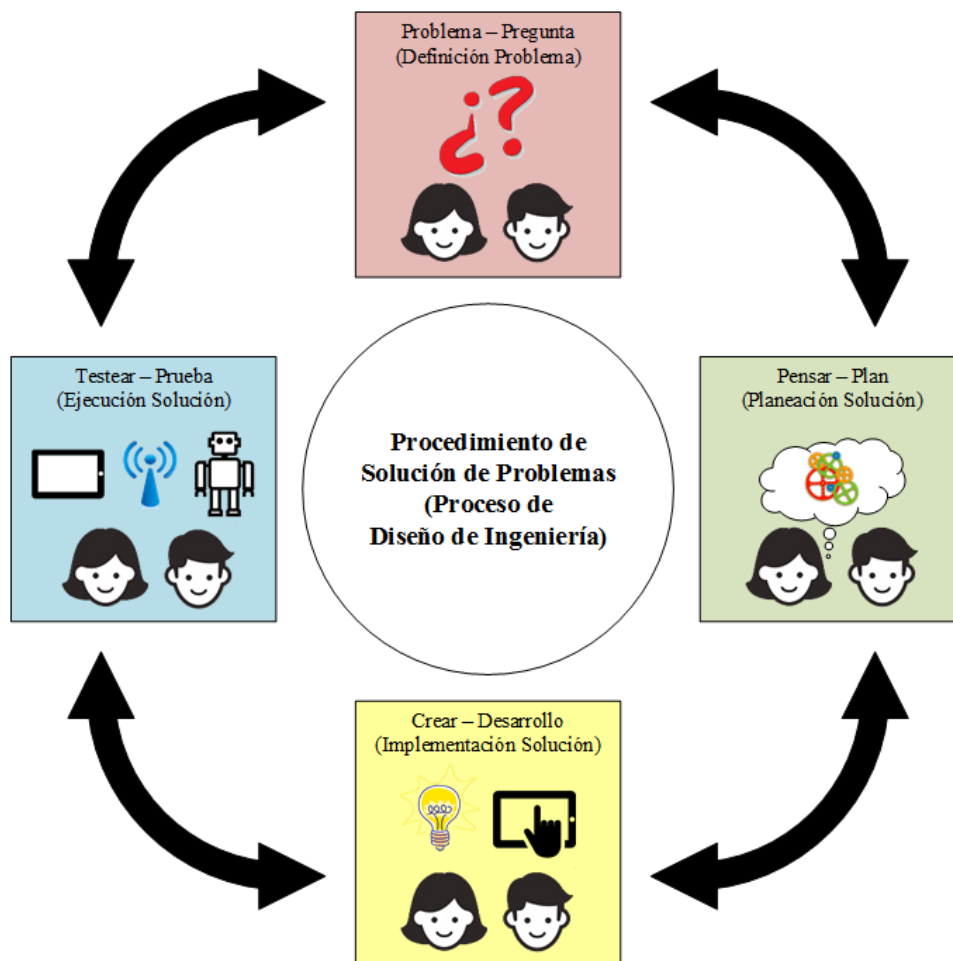


Figura 62. Procedimiento de Solución de Problemas.

Asimismo, con la investigación se ha generado divulgación a nivel nacional e internacional en el desarrollo de tecnología para la educación costarricense (ver Apéndice 1). Dicha divulgación repercute en la buena imagen y proyección de la Escuela de Ciencias de la Computación e Informática (ECCI) y el Centro de Investigaciones de Tecnologías de Información y Comunicación (CITIC) a nivel nacional e internacional. Al mismo tiempo, la Universidad de Costa Rica y el país obtuvieron presencia y reconocimiento internacional.

Finalmente, se está trabajando junto con el personal de PROINNOVA-UCR para realizar el proceso de gestión de transferencia tecnológica de TITIBOTS (individual y colaborativo) a la Sociedad Costarricense, ya que la investigación pasó los estudios de Inteligencia Competitiva y de Mercados preliminares realizados por esta Unidad. PROINNOVA consideró la investigación una Oportunidad para Innovar (OPI) (expediente N° ING-031, notificado a la investigadora el 27 de febrero del 2015), porque determinaron que posee novedad y potencial valor comercial. Además, consideran que tiene una capacidad de poder extenderse a niños de otras edades.

Como parte de la estrategia para realizar la transferencia tecnológica, se solicitará la protección de la propiedad intelectual ante el Registro Nacional y eventualmente en cada país donde se difundan las herramientas. Dado es el caso de Brasil, cuyas autoridades de Educación están interesadas en la adquisición de la herramienta colaborativa.

Aparte de los aportes presentados, a continuación, se muestra el trabajo futuro de esta investigación, donde se ofrecen diferentes ideas surgidas de este proyecto.

8.2 Trabajo Futuro

En la creación de TITIBOTS (ambas herramientas) se siguió la guía de lineamientos presentada en la Sección 6.2, como se mencionó en dicha sección, se implementó la mayoría de los lineamientos en las herramientas. Los lineamientos que no aún no se han implementado son: consejos para los padres, la ayuda y la documentación dirigida a los niños, registro e inicio de sesión, incorporar más ayuda y retroalimentación visual que acompañe a la de audio. En estos lineamientos se está trabajando actualmente para obtener una nueva versión mejorada de las herramientas (individual y colaborativa).

Entre los comandos que fueron implementados no se encuentran las estructuras de control: la condición y el ciclo, ya que se consideró que pueden ser conceptos demasiado difíciles de comprender por parte de los niños de preescolar. Lo mismo ocurre con los sensores (otros componentes de los robots). En relación con estos temas, se consultó a expertos en el área de

la educación preescolar, pero nadie pudo indicar si estos conceptos podrán ser demasiado complicados o no. Debido a su complejidad se han dejado para implementarse posteriormente.

En relación al protocolo de comunicación de TITIBOTS Colab, se desea encontrar un formato más compacto para los mensajes o definir un formato personalizado como trabajo futuro. Además, se ampliará el protocolo para permitir que las asistentes de las docentes puedan tener una tableta con la aplicación para supervisar el trabajo de los estudiantes, de manera similar a la docente principal, pero sin interacción directa.

Asimismo, se está trabajando con Tomás De Camino Beck y el equipo de la Inventoría¹⁷, en la creación de un robot de bajo costo y propio para las herramientas de programación. A su vez se trabajará con diferentes docentes de preescolar para mejorar la estrategia didáctica para el aprendizaje de la programación y el desarrollo de habilidades de colaboración en niños entre 4 y 6 años usando las herramientas de programación creadas; con el fin de ofrecer una guía didáctica a las docentes en la utilización de las herramientas de programación.

Lo anterior forma parte del trabajo que se está realizando junto con PROINNOVA-UCR¹⁸ en el proceso de gestión de transferencia tecnológica de TITIBOTS (individual y colaborativo) a la Sociedad Costarricense. Con el objetivo que la investigación trascienda las barreras de la academia e impacten de la mejor manera en beneficio de nuestra sociedad.

Además, se seguirá trabajando en el Proyecto 834-B5-A04: Creación de un entorno de programación colaborativo orientado a niños entre 4 y 6 años que fomente el pensamiento lógico-matemático y la solución de problemas; el cual está activo y tiene vigencia entre Julio 2015 – Junio 2017.

¹⁷ Inventoría: Laboratorio de innovación ciudadana, científica y tecnológica. URL: <https://sites.google.com/site/inventoriafuncostarica/>.

¹⁸ Unidad que forma parte de la Vicerrectoría de Investigación de la UCR. Organización responsable de la gestión y transferencia de los conocimientos generados por la Universidad de Costa Rica.

Entre los meses de febrero y junio se estarán realizando varios talleres con la herramienta TITIBOTS Colab, uno de estos talleres será en la Escuela El Castillo, por motivo que fue la Institución que participó en el cuasi-experimento como grupo de control. Otra Institución que solicitó un taller fue la Centeno Güell, para trabajar el pensamiento lógico y abstracto con estudiantes sordos de la Institución. La investigadora fue contactada por M. Psi. Juan José Solano Barboza (psicólogo educativo) que labora en la Institución, y se dio cuenta por el artículo de la Vicerrectoría de Investigación de la UCR que realizó sobre la investigación. Representantes del CITIC y PROINNOVA se reunieron con autoridades de la Centeno Güell para hablar sobre la investigación y su aporte en esa institución.

Por otro lado, esta investigación ha motivado la realización de dos proyectos de investigación de Licenciatura:

- Título: “Creación de una Herramienta Informática para Diseño de Entornos Virtuales Colaborativos”, de los estudiantes: David Arias Retana y Eduardo Vitoria Barboza. El objetivo principal de este trabajo es la creación de un mundo virtual para la aplicación TITIBOTS Colab, que ya cuenta con un entorno físico (robot físico), en el cual se desea aumentar con un mundo virtual (robots virtuales) en caso de que no existan los medios físicos para poder usar el programa.
- Título: “Inclusión de nuevas características en la herramienta de programación TITIBOTS, para que niños con edades entre los 6 y 12 años puedan trabajar en ambientes de aprendizaje basados en problemas”, de los estudiantes: Javier Andrés Muñoz Barquero, Gregory José Obando Araya y David Eduardo Villegas Vargas. El objetivo principal de este trabajo es adaptar la herramienta de programación TITIBOTS para dirigirla a una audiencia con mayor edad, específicamente niños entre 6 y 12 años, incluyendo conceptos básicos de programación, como: condicionales, ciclos, variables, entre otros.

En la Sección 7.2 se mostraron los resultados obtenidos sobre las variables relacionadas a los desafíos de evaluación que realizaron los niños en los talleres. En los resultados obtenidos con TITIBOTS se observa que, en relación con estas variables, son inferiores a los resultados de TITIBOTS Colab, tanto en Brasil como Costa Rica (ver Tabla 41). Los niños de los

talleres donde se utilizó la herramienta colaborativa lograron todos los desafíos de evaluación realizando una cantidad de intentos promedio y tiempo promedio menor. Además, se disminuyó el promedio de errores y todos los errores fueron recuperados por los niños.

Con base a estos resultados, puede ser que el elemento colaboración influyó para lograr los buenos resultados de los desafíos de los dos últimos pilotos, dado los principios y los beneficios de trabajar de forma colaborativa que se encuentran en la literatura. Otro elemento que pudo influir en los resultados es la duración del taller. Por estas razones, se formuló la hipótesis que el aprendizaje de la programación en niños de preescolar es mejor cuando se trabaja con una herramienta colaborativa que incentiva el proceso de colaboración, como TITIBOTS Colab, y se desea realizar un estudio formal para llegar a comprobar dicha hipótesis. En dicho estudio se trabajaría con dos grupos naturales, uno usando la herramienta individual y el otro la colaborativa en un mismo período de tiempo (misma duración).

Tabla 41. Resumen de las variables de los desafíos de evaluación de los talleres.

Variable	TITIBOTS (CR)	TITIBOTS Colab (Brasil)	TITIBOTS Colab (CR)
Duración del taller	4 sesiones (1 hora y media cada sesión)	10 sesiones (1 hora y media cada sesión)	7 sesiones (1 hora y media cada sesión)
Porcentaje de desafíos exitosos (Eficiencia)	67%	100%	100%
Cantidad de intentos promedio para completar un desafío (Eficiencia)	2,44	1,17	1,47
Tiempo promedio para completar un desafío (Eficiencia)	19,94min	13,87min	8,33min
Promedio de errores (Efectividad)	3,38	0,80	2,80
Promedio de errores recuperados (Efectividad)	2,33	0,80	2,80
Porcentaje de errores recuperados (Efectividad)	69%	100%	100%

Finalmente, se seguirá trabajando con los investigadores del RExLab en el entorno remoto de TITIBOTS Colab que se creó durante el periodo de la pasantía de la investigadora: Cobra Cora¹⁹, y se quiere llegar a ensamblar un entorno remoto en Costa Rica.

¹⁹ URL: <http://relle.ufsc.br/>.

REFERENCIAS

- Acuña-Zúñiga, A. L. (2006a). Proyectos de robótica educativa: motores para la innovación. En *Current Development in Technology- Assisted Education* (pp. 951-956). Recuperado a partir de http://www.fod.ac.cr/pdf/publicaciones/articulos/2006/proyectos_de_robotica_educativa_motores_para_la_innovacion.pdf
- Acuña-Zúñiga, A. L. (2006b). Robótica: para el desarrollo de habilidades en diseño con niños, niñas y jóvenes en América Latina. En *II Jornada de Informática Educativa y Tecnología Educativa* (p. 5). Santa Ana de Coro. Recuperado a partir de http://www.fod.ac.cr/pdf/publicaciones/articulos/2004/robotica_para_el_desarrollo_de_habilidades.pdf
- Adams, D. M., & Hamm, M. (1996). *Cooperative Learning: Critical Thinking and Collaboration Across the Curriculum*. Springfield, IL: Charles C Thomas Pub Ltd.
- Aedo, C. (2005). *Evaluación del impacto* (1.^a ed.). Santiago, Chile: Naciones Unidas.
- Aho, A. V. (2012). Computation and Computational Thinking. *The Computer Journal*, 55(7), 832-835. <http://doi.org/10.1093/comjnl/bxs074>
- Alvarez, C., Alarcon, R., & Nussbaum, M. (2011). Implementing collaborative learning activities in the classroom supported by one-to-one mobile computing: A design-based process. *Journal of Systems and Software*, 84(11), 1961-1976. <http://doi.org/10.1016/j.jss.2011.07.011>
- Amigues, R., & Caillot, M. (1990). Les représentations graphiques dans l'enseignement et l'apprentissage de l'électricité. *European Journal of Psychology of Education*, 5(4), 477-488. <http://doi.org/10.1007/BF03173133>
- Android Developers. (2014a). Creating P2P Connections with Wi-Fi. Recuperado a partir de <http://developer.android.com/training/connect-devices-wirelessly/wifi-direct.html>
- Android Developers. (2014b). Using Wi-Fi P2P for Service Discovery. Recuperado a partir de <http://developer.android.com/training/connect-devices-wirelessly/nsd-wifi-direct.html>
- Arce-Ulloa, A. I., Blanco-García, S., Cerdas-González, A. I., & Zúñiga-León, I. (1995). *Programa de Estudio Ciclo de Transición - Educación Preescolar / Ministerio de Educación Pública* (1.^a ed.). San José, Costa Rica: Ministerios de Educación Pública CR.
- Barnum, C. M. /Dragga, S. (2001). Usability Testing and Research. Recuperado a partir de <http://dl.acm.org/citation.cfm?id=559526>
- Barros, B., & Verdejo, M. F. (1999). An approach to analyse collaboration when shared structured workspaces are used for carrying out group learning processes. En *International Conference on Artificial Intelligence in Education*. Le Mans, Francia.
- Begel, A., & Klopfer, E. (2007). Starlogo TNG: An introduction to game development. *Journal of E-Learning*.

- Bers, M., Rogers, C., Beals, L., Portsmore, M., Staszowski, K., Cejka, E., ... Barnett, M. (2006). Innovative Session: Early Childhood Robotics for Learning. En *ICLS '06 Proceedings of the 7th international conference on Learning sciences* (pp. 1036-1042). Bloomington, Indiana: International Society of the Learning Sciences. Recuperado a partir de <http://dl.acm.org/citation.cfm?id=1150034.1150229>
- Bers, M. U. (2008). *Blocks to Robots: Learning with Technology in the Early Childhood Classroom*. (Teachers College Press, Ed.). Teachers College Press.
- Bers, M. U. (2010). The TangibleK Robotics Program: Applied Computational Thinking for Young Children. *Early Childhood and Parenting Collaborative*, 12(2).
- Bers, M. U. (2010). The TangibleK Robotics Program: Applied Computational Thinking for Young Children. *Early Childhood Research & Practice (ECRP)*, 12(2). Recuperado a partir de <http://ecrp.uiuc.edu/v12n2/bers.html>
- Bers, M. U. (2012a). *Designing Digital Experiences for Positive Youth Development*. (Oxford University Press, Ed.) (1.^a ed.). Oxford University Press. Recuperado a partir de <http://global.oup.com/academic/product/designing-digital-experiences-for-positive-youth-development-9780199757022;jsessionid=618059CD74508459C4442D7BFF0EACBC?cc=cr&lang=en&>
- Bers, M. U. (2012b). Programming in Kindergarten: A Playground Experience. *Computer Science K-8: Building a Strong Foundation*, (Special), 7-8.
- Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72, 145-157. <http://doi.org/10.1016/j.compedu.2013.10.020>
- Bers, M. U., Seddighin, S., & Sullivan, A. (2013). Ready for Robotics: Bringing Together the Y and E of STEM in Early Childhood Teacher Education. *Journal of Technology and Teacher Education*, 21(3), 355-377. Recuperado a partir de <http://ase.tufts.edu/DevTech/publications/BringingTogetherT.pdf>
- Binkley, M., Erstad, O., Herman, J., Raizen, S., Ripley, M., Miller-Ricci, M., & Rumble, M. (2012). Defining twenty-first century skills. En *Assessment and teaching of 21st century skills* (pp. 17-66). Springer. Recuperado a partir de <http://atc21s.org/wp-content/uploads/2011/11/1-Defining-21st-Century-Skills.pdf>
- Blackwell, A. F., & Green, T. R. G. (2000). A Cognitive Dimensions Questionnaire Optimised for Users. En A. F. Blackwell & E. Bilotta (Eds.), *12th Workshop of the Psychology of Programming Interest Group* (pp. 137-154). Cozenza, Italy.
- Brna, P., & Burton, M. (1997). Roles, Goals and Effective Collaboration. En *Proceedings of the IV Collaborative Learning Workshop in the 8th World Conference on Artificial Intelligence in Education* (pp. 3-10). Kobe, Japón.
- Cambell, D. T., & Stanley, J. C. (1995). *Diseños experimentales y cuasiexperimentales en la investigación social* (7.^a ed.). Buenos Aires, Argentina: Rand McNally & Company. Recuperado a partir de <https://sociologiaycultura.files.wordpress.com/2014/02/campbell-stanley-disec3b1os-experimentales-y-cuasiexperimentales-en-la-investigacic3b3n-social.pdf>

- Campe, S., Werner, L., & Denner, J. (2012). Game Programming with Alice: A Series of Graduated Challenges. *Computer Science K-8: Building a Strong Foundation*, (Special), 13-14.
- Carmioli, A. M., & Pérez, R. (2015a). *El andamiaje parental durante el visionado televisivo conjunto entre madres y sus hijos/as en edad preescolar: Caracterización y efecto en la comprensión del contenido televisivo de los niños y las niñas*. San José, Costa Rica.
- Carmioli, A. M., & Pérez, R. (2015b). *Reporte Preliminar: Uso de TICs en niñas y niños preescolares*. San José, Costa Rica.
- Caron, D. (2010). Competitive Robotics Brings Out the Best in Students. *Tech Directions*, 69(6), 21-23.
- Castro-Rojas, M. D., & Acuña-Zúñiga, A. L. (2012). Propuesta comunitaria con robótica educativa: valoración y resultados de aprendizaje. *Teoría de la Educación: Educación y Cultura en la Sociedad de la Información*, 13(2), 91-119. Recuperado a partir de http://campus.usal.es/~revistas_trabajo/index.php/revistatesi/article/view/9001/9246
- Cerdas-González, A. I., & Mata-Solano, A. L. (2001). *Programa de Estudio Ciclo Materno Infantil - Educación Preescolar / Ministerio de Educación Pública* (2.ª ed.). San José, Costa Rica: Ministerios de Educación Pública CR.
- Chau, C. L. (2014). *Positive Technological Development for Young Children in the Context of Children's Mobile Apps*. Tufts University. Recuperado a partir de http://ase.tufts.edu/DevTech/resources/Theses/CChau_2014.pdf
- Chaudron, S. (2015). *Young Children (0-8) and digital technology: A qualitative exploratory study across seven countries*. Publications Office of the European Union. <http://doi.org/10.2788/00749>
- Cheng, S. (2011). The research framework of eye-tracking based mobile device usability evaluation. En *Proceedings of the 1st international workshop on Pervasive eye tracking & mobile eye-based interaction - PETMEI '11* (p. 21). New York, New York, USA: ACM Press. <http://doi.org/10.1145/2029956.2029964>
- codeSpark. (2014). The Foos Are Here! The Foos Are Here! Recuperado 10 de agosto de 2015, a partir de <http://codespark.org/the-foos-are-here-the-foos-are-here/>
- Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2.ª ed.). Routledge.
- Colelli, R. (2009). Model Program: Southern Lehigh High School, Center Valley, PA. *Technology Teacher*, 68(4), 27-32.
- Collazos, C. (2003). *Una metodología para el apoyo computacional de la evaluación y monitoreo en ambientes de aprendizaje colaborativo*. Universidad de Chile.
- Collazos, C., Guerrero, L. A., Llaña, M., & Oetzel, J. (2001). El rol del género dentro del proceso de aprendizaje colaborativo. En *Memorias del Taller Internacional de Software Educativo (TISE2001)*. Santiago, Chile.
- Collazos, C., Guerrero, L. A., & Vergara, A. (2001). Aprendizaje Colaborativo: un cambio en el rol del profesor. En *Memorias del III Congreso de Educación Superior en Computación*. Punta Arenas, Chile: Jornadas Chilenas de la Computación.

- Collazos, C., Muñoz, J., & Hernández, Y. (2014). *Aprendizaje Colaborativo Apoyado por Computador. Journal of Chemical Information and Modeling* (1.^a ed.). Iniciativa Latinoamericana de Libros de Texto Abiertos (LATIn).
<http://doi.org/10.1017/CBO9781107415324.004>
- Common Sense Media Research Study. (2013). *Zero to Eight: Children's Media Use in America 2013*. USA. Recuperado a partir de
<https://www.commonsensemedia.org/research/zero-to-eight-childrens-media-use-in-america-2013>
- Conway, M., Pausch, R., Gossweiler, R., & Burnette, T. (1994). Alice: a rapid prototyping system for building virtual environments. En *Conference Companion on Human Factors in Computing Systems* (pp. 295-296).
- Cooper, S., Dann, W., & Pausch, R. (2000). Alice: a 3D Tool For Introductory Programming Concepts. En *Journal of Computing Sciences in Colleges* (Vol. 15, pp. 107-116). Consortium for Computing Sciences in Colleges.
<http://doi.org/10.1145/259963.260503>
- Coursaris, C. K., & Kim, D. (2007). A research agenda for mobile usability. *CHI '07 extended abstracts on Human factors in computing systems - CHI '07*, 2345-2350.
<http://doi.org/10.1145/1240866.1241005>
- Coxon, S. V. (2010). *Design to succeed in LEGO WeDo challenges: An enrichment unit for ages 7-10*.
- Coxon, S., & Chandler, K. (2010). LEGO WeDo.
- de Leeuw, E. D. (2011). *Improving Data Quality when Surveying Children and Adolescents: Cognitive and Social Developmente and its Role in Questionnaire Construction and Pretesting*. Naantali Finland.
- Denning, P. J. (2009). The profession of IT: Beyond computational thinking. *Communications of the ACM*, 52(6), 28-30. <http://doi.org/10.1145/1516046.1516054>
- Díaz Barriga, F., & Hernández Rojas, G. (2002). *Estrategias docentes para un aprendizaje significativo: Una interpretación constructivista*. México: McGraw Hill.
- Díaz, J., Harari, I., & Amadeo, A. P. (2013). *Guía de Recomendaciones para Diseño de Software Centrado en el Usuario* (1.^a ed.). La Plata: Universidad Nacional de La Plata.
- Dillenbourg, P. (1999). What do you mean by «collaborative learning»? *Collaborative-learning: Cognitive and Computational Approaches*, 1, 1-19.
- Dillenbourg, P., Baker, M. J., Blaye, A., O'Malley, C., Blake, A., & O'Malley, C. (1995). The evolution of research on collaborative learning. En H. In Spada & P. Reimann (Eds.), *Learning in Humans and Machine: Towards an interdisciplinary learning science* (pp. 189-211). Oxford: Elsevier.
- Downie, N. M., & Heath, R. W. (1973). *Métodos estadísticos aplicados*. (Harla, Ed.). Harper & Row Publisher Inc.
- Druin, A. (1999). Cooperative Inquiry: Developing New Technologies for Children with Children. En *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99* (pp. 592-599). New York, New York, USA:

- ACM Press. <http://doi.org/10.1145/302979.303166>
- Duh, H. B.-L., Tan, G. C. B., & Chen, V. H. (2006). Usability evaluation for mobile device: a comparison of laboratory and field tests. En *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services - MobileHCI '06* (pp. 181-186). New York, New York, USA: ACM Press. <http://doi.org/10.1145/1152215.1152254>
- Dutson, P. (2014). *Responsive Mobile Design: Designing for Every Device (Usability)* (1.^a ed.). Addison-Wesley Professional.
- Eberts, R. E. (1994). *User Interface Design*. (Prentice Hall, Ed.). Englewood Cliffs, NJ: Prentice Hall College Div.
- Echeverría, A., García-Campo, C., Nussbaum, M., Gil, F., Villalta, M., Améstica, M., & Echeverría, S. (2011). A framework for the design and integration of collaborative classroom games. *Computers & Education*, 57(1), 1127-1136. <http://doi.org/10.1016/j.compedu.2010.12.010>
- Echeverría, H. D. (2005). *Los diseños de investigación y su implementación en Educación* (1.^a ed.). Rosario, Santa Fe, Argentina: Homo Sapiens Ediciones.
- EGM Robotics, & Clear Minds. (2015). ¿Qué es Teebot? Recuperado 25 de abril de 2015, a partir de http://teebot.com.ec/que_es_teebot/
- Einhorn, S. (2011). MicroWorlds, Computational Thinking, and 21st Century Learning. *LCSI White Paper*, 1-10.
- endshelf. (2013). Kodable. Recuperado a partir de <https://edshelf.com/tool/kodable>
- Fails, J. A. (2007). Mobile collaboration for young children. En *Proceedings of the 6th international conference on Interaction design and children - IDC '07* (pp. 181-184). New York, New York, USA: ACM Press. <http://doi.org/10.1145/1297277.1297324>
- Fetaji, B., & Fetaji, M. (2011). Analyses and Review of M-learning Feasibility , Trends , Advantages and Drawbacks in the past Decade (2000 -2010). En *Proceedings of the 5th European conference on European computing conference* (pp. 474-479). World Scientific and Engineering Academy and Society (WSEAS).
- Flannery, L. P., Kazakoff, E. R., Bontá, P., Silverman, B., Bers, M. U., Resnick, M., ... Resnick, M. (2013). Designing ScratchJr: Support for Early Childhood Learning Through Computer Programming. En *Proceedings of the 12th International Conference on Interaction Design and Children (IDC '13)* (pp. 1-10). New York, NY, USA: ACM. <http://doi.org/10.1145/2485760.2485785>
- Frohberg, D., Göth, C., & Schwabe, G. (2009). Mobile Learning projects - a critical analysis of the state of the art. *Journal of Computer Assisted Learning*, 25(4), 307-331. <http://doi.org/10.1111/j.1365-2729.2009.00315.x>
- Gifford, B. R., & Enyedy, N. D. (1999). Activity centered Design: CSCL Towards a Theoretical Framework for CSCL. En *Proceedings of Computer Supported for Collaborative Learning* (pp. 189-196). Stanford, USA.
- Ginsburg, H., & Opper, S. (1988). *Piaget y la teoría del desarrollo intelectual*. Distrito Federal, México: Prentice-Hall Hispanoamericana, S.A.

- Glass, G. V., McGaw, B., & Smith, M. L. (1981). *Meta-analysis in social research*. California, USA: SAGE Publications.
- Gordon, M., Rivera, E., Ackermann, E., & Breazeal, C. (2015). Designing a Relational Social Robot Toolkit for Preschool Children to Explore Computational Concepts. En *Proceedings of the 14th International Conference on Interaction Design and Children - IDC '15* (pp. 355-358). New York, New York, USA: ACM Press. <http://doi.org/10.1145/2771839.2771915>
- Granic, A., & Cukusic, M. (2011). Usability Testing and Expert Inspections Complemented by Educational Evaluation: A Case Study of an e-Learning Platform. *Educational Technology & Society*, 14(2), 107-123.
- Green, T. R. G., & Petre, M. (1996). Usability Analysis of Visual Programming Environments: A 'Cognitive Dimensions' Framework. *Journal of Visual Languages & Computing*, 7(2), 131-174. <http://doi.org/10.1006/jvlc.1996.0009>
- Greenberg, S., & Bohnet, R. (1991). GroupSketch: A multi-user sketchpad for geographically-distributed small groups. En *Proceedings of Graphics Interface '91* (pp. 207-215). Toronto, Ontario, Canada: Canadian Man-Computer Communications Society. Recuperado a partir de <http://graphicsinterface.org/wp-content/uploads/gi1991-27.pdf>
- Guerrero, L. A., Alarcón, R., & Collazos, C. (2000). Indicadores de Cooperación en el Trabajo Grupal. En *Conferencia Latinoamericana de Informática, CLEI'2000* (p. 10). DF, México: CLEI. Recuperado a partir de <http://www.guerrero.cr/papers/CLEI-00.pdf>
- Guha, M. L., Druin, A., Chipman, G., Fails, J. A., Simms, S., & Farber, A. (2004). Mixing Ideas : A New Technique for Working with Young Children as Design Partners. En *Proceedings of the 2004 conference on Interaction design and children: building a community* (pp. 35-42). New York, NY, USA: ACM.
- Gupta, V., Gupta, S., & Greaves, M. (2013). Play-i. Recuperado 5 de mayo de 2014, a partir de <https://www.play-i.com/>
- Hamada, M., & Sato, S. (2010). Lego NXT as a learning tool. *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education - ITiCSE '10*, 321. <http://doi.org/10.1145/1822090.1822198>
- Hambrusch, S., Hoffmann, C., Korb, J. T., Haugan, M., & Hosking, A. L. (2009). A multidisciplinary approach towards computational thinking for science majors. En *Proceedings of the 40th ACM technical symposium on Computer science education - SIGCSE '09* (Vol. 41, pp. 183-187). New York, New York, USA: ACM Press. <http://doi.org/10.1145/1508865.1508931>
- Hamdan, N. A., & Schaper, H. (2012). Collaboration in Mobile Learning. En *Computer-Supported Learning Research Group* (pp. 1-11).
- Hassan Montero, Y. (2004, septiembre). Diseño web orientado a niños. *No Solo Usabilidad*. Recuperado a partir de nosolousabilidad.com
- Hassan-Montero, Y., & Ortega-Santamaría, S. (2009). *Informe APEI sobre usabilidad*.

- (APEI Asociación Profesional de Especialistas en Información, Ed.) (Informes A). APEI Asociación Profesional de Especialistas en Información. Recuperado a partir de <http://hdl.handle.net/10760/13253>
- Henderson, P. B., Cortina, T. J., Hazzan, O., & Wing, J. M. (2007). Computational thinking. En *SIGCSE'07* (Vol. 49, p. 33). Covington, Kentucky, USA: ACM. <http://doi.org/10.1145/1118178.1118215>
- Horn, M. S., Solovey, E. T., & Jacob, R. J. K. (2008). Tangible Programming and Informal Science Learning: Making TUIs Work for Museums. En *Proceedings of the 7th international conference on Interaction design and children - IDC '08* (pp. 194-201). Chicago, IL, USA: ACM Press. <http://doi.org/10.1145/1463689.1463756>
- Huebner, G., & Mattingly, J. (2012). Kodable. Recuperado 4 de mayo de 2014, a partir de <http://www.kodable.com/>
- Hwang, W., & Salvendy, G. (2010). Number of people required for usability evaluation. *Communications of the ACM*, 53(5), 130. <http://doi.org/10.1145/1735223.1735255>
- International Organization for Standardization. (1999). *ISO 13407:1999 Human-centred design processes for interactive systems*. Recuperado a partir de http://www.iso.org/iso/catalogue_detail.htm?csnumber=21197
- International Organization for Standardization. (2000). *ISO TR-18529:2000 Human-centred lifecycle process descriptions*. Recuperado a partir de http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=33499
- International Organization for Standardization. (2001). *ISO 9126-1:2001 Software engineering – Product quality, Part 1: Quality model*. Recuperado a partir de http://www.iso.org/iso/catalogue_detail.htm?csnumber=22749
- International Organization for Standardization. (2010). *ISO 9241-210:2010 Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems*. Recuperado a partir de http://www.iso.org/iso/catalogue_detail.htm?csnumber=52075
- International Society for Technology in Education, & Computer Science Teachers Association. (2011). *Operational Definition of Computational Thinking for K–12 Education*. Recuperado a partir de <http://csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf>
- Ivey, D., & Quam, G. (2009). 4-H and Tech Ed Partnership Gets Students Geeked about STEM. *Tech Directions*, 69(3), 19-21. Recuperado a partir de <http://www.uwex.edu/ces/4h/set/documents/StudentsGeekedaboutSTEM10-09.pdf>
- James, C. (2012). An 8-Day Plan with Etoys. En P. Phillips (Ed.), *Computer Science K–8: Building a Strong Foundation* (pp. 23-24). New York, New York, USA: ACM.
- Johnson, D. W., & Johnson, F. P. (1997). *Joining Together: Group Theory and Group Skills*. (M. Needham Heights, Ed.). Allyn & Bacon.
- Johnson, D. W., & Johnson, R. T. (1999). *Aprender Juntos y Solos: Aprendizaje Cooperativo, Competitivo e Individualista* (1.ª ed.). Buenos Aires, Argentina: Grupo Editorial Aique S.A.

- Johnson, D. W., & Johnson, R. T. (2000). *Cooperative Learning and Conflict Resolution*. Minneapolis, Minnesota. Recuperado a partir de <http://www.tablelearning.com/uploads/File/EXHIBIT-D.pdf>
- Johnson, D. W., Johnson, R. T., & Holubec, E. J. (1998). *Cooperation in the classroom*. (Edina, Ed.) (1.^a ed.). Interaction Book Company.
- Johnson, D. W., Johnson, R. T., & Stanne, M. B. (2000). *Cooperative Learning Methods: A Meta-Analysis*. Minneapolis, Minnesota. Recuperado a partir de <http://www.ccsstl.com/sites/default/files/Cooperative Learning Research .pdf>
- Johnson, J. (2003). Children, robotics, and education. *Artificial Life and Robotics*, 7(1-2), 16-21. <http://doi.org/10.1007/BF02480880>
- Kallio, T., & Kaikkonen, A. (2005). Usability testing of mobile applications: A comparison between laboratory and field testing. *Journal of Usability studies*, 1(4), 23-28.
- Kaye, A. R. (Ed.). (1992). *Collaborative Learning Through Computer Conferencing*. Berlin, Heidelberg: Springer Berlin Heidelberg. <http://doi.org/10.1007/978-3-642-77684-7>
- Kazakoff, E. R., Sullivan, A., & Bers, M. U. (2012). The Effect of a Classroom-Based Intensive Robotics and Programming Workshop on Sequencing Ability in Early Childhood. *Early Childhood Education Journal*, 41(4), 245-255. <http://doi.org/10.1007/s10643-012-0554-5>
- Kim, S. H., & Jeon, J. W. (2007). Programming LEGO mindstorms NXT with visual programming. En *International Conference on Control, Automation and Systems, 2007. ICCAS '07*. (pp. 2468-2472). Seoul. Recuperado a partir de <https://sibdi.ucr.ac.cr/http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=4406778&queryText=Programming+LEGO+mindstorms+NXT+with+visual+programming>
- Kitchenham, B. (2004). *Procedures for Performing Systematic Reviews*. Keele, UK.
- Kjeldskov, J., & Graham, C. (2003). A Review of Mobile HCI Research Methods. En L. Chittaro (Ed.), *Proceedingd of 5th International Mobile HCI 2003 conference* (pp. 317-335). Udine, Italy: Springer-Verlag.
- Kjeldskov, J., Graham, C., Pedell, S., Vetere, F., Howard, S., Balbo, S., & Davies, J. (2005). Evaluating the usability of a mobile guide: The influence of location, participants and resources. *Behaviour & Information Technology*, 24(1), 51-65. <http://doi.org/10.1080/01449290512331319030>
- Kolodner, J., & Nagel, K. (1999). The Design Discussion Area: A Collaborative Learning Tool in Support of Learning from Problem-Solving and Design Activities. En *Proceedings of Computer Supported for Collaborative Learning* (pp. 300-307). Stanford, USA.
- Kuniavsky, M. (2003). *Observing The User Experience: A Practitioner's Guide to User Research* (1.^a ed.). San Francisco, CA, USA: Elsevier Science.
- Labinowicz, E. (1982). *Introducción a Piaget: pensamiento, aprendizaje, enseñanza* (1.^a ed.). Fondo Educativo Interamericano.

- Laboratorio de Innovación Educativa. (2012). *Qué - Por qué - Para qué - Cómo: Aprendizaje Cooperativo*. Madrid, España. Recuperado a partir de http://www.madrid.org/dat_capital/upe/impresos_pdf/AprendizajeCooperativo2012.pdf
- Laerd Statistics. (2015a). Independent-samples t-test using SPSS Statistics. Recuperado 15 de diciembre de 2015, a partir de <https://statistics.laerd.com/>
- Laerd Statistics. (2015b). Mann-Whitney U test using SPSS Statistics. Recuperado 25 de febrero de 2016, a partir de <https://statistics.laerd.com/>
- Landau, P. R., Mulder, G., Holmes, R., Borinskaya, S., Kang, N., & Bordeianu, C. (2013). INSTANCES: incorporating computational «scientific» thinking advances into education & science courses. En *Proceedings of the Conference on Extreme Science and Engineering Discovery Environment Gateway to Discovery - XSEDE '13* (pp. 1-4). New York, New York, USA: ACM Press. <http://doi.org/10.1145/2484762.2484769>
- Leavitt, M. O., & Shneiderman, B. (2013). *Research-Based Web Design & Usability Guidelines* (1.^a ed.). Wa: U.S. Government Printing Office. Recuperado a partir de http://www.usability.gov/sites/default/files/documents/guidelines_book.pdf?post=yes
- Lee, Y.-J. (2011). Empowering teachers to create educational software: A constructivist approach utilizing Etoys, pair programming and cognitive apprenticeship. *Computers & Education*, 56(2), 527-538. <http://doi.org/10.1016/j.compedu.2010.09.018>
- LEGO. (2013). LEGO Mindstorms EV3. Recuperado a partir de <http://www.lego.com/en-us/mindstorms/?domainredir=mindstorms.lego.com>
- Lentz, T. A. (2014). *Kids, Robotics, and Gender: a pilot study*. Tufts University. Recuperado a partir de http://ase.tufts.edu/DevTech/resources/Theses/TLentz_2014.pdf
- León Sáenz, A. T. (2010). *Desarrollo y atención del niño de 0 a 6 años (Primera Parte)* (2.^a ed.). San José, Costa Rica: EUNED.
- Liukas, L. (2014). Hello Ruby. Recuperado 5 de mayo de 2014, a partir de <https://www.kickstarter.com/projects/lindaliukas/hello-ruby>
- Louden, K. C., & Lambert, K. A. (2011). *Programming Languages: Principles and Practices* (3.^a ed.). Cengage Learning.
- Maloney, J. H., Peppler, K., Kafai, Y., Resnick, M., & Rusk, N. (2008). Programming by Choice Urban Youth Learning Programming with Scratch. *ACM SIGCSE Bulletin*, 40(1), 367. <http://doi.org/10.1145/1352322.1352260>
- Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The Scratch Programming Language and Environment. *ACM Transactions on Computing Education*, 10(4), 1-15. Recuperado a partir de <http://web.media.mit.edu/~jmaloney/papers/ScratchLangAndEnvironment.pdf>
- Mander, J., & Picopoulos, D. (2005). *Bluetooth Piconet Applications*. London, England. Recuperado a partir de <http://www.ee.ucl.ac.uk/~afernand/Example1.pdf>
- Mathilde Bekker, Barendregt, W., Crombeen, S., & Biesheuvel, M. (2005). Evaluating Usability and Challenge during Initial and Extended Use of Children's Computer

- Games. En Springer-Verlag London Limited (Ed.), *People and Computers XVIII — Design for Life* (pp. 331-345). London: Springer London. http://doi.org/10.1007/1-84628-062-1_21
- Mayerová, K. (2012). Pilot Activities: LEGO WeDo at Primary School. En *Proceedings of 3rd International Workshop Teaching Robotics, Teaching with Robotics: Integrating Robotics in School Curriculum* (pp. 32-39). Riva del Garda, Italy.
- McConnell, D. (1994). *Implementing computer supported cooperative learning*. London: Kogan Page Limited.
- McConnell, D. (1999). Examining a collaborative assessment process in networked lifelong learning. *Journal of Computer Assisted Learning*, 15(3), 232-243. <http://doi.org/10.1046/j.1365-2729.1999.153097.x>
- McNerney, T. S. (2004). From turtles to Tangible Programming Bricks: explorations in physical language design. *Personal and Ubiquitous Computing*, 8(5), 326-337. <http://doi.org/10.1007/s00779-004-0295-6>
- Morales Vallejo, P. (2012). *El tamaño del efecto (effect size): análisis complementarios al contraste de medias. Estadística aplicada a Ciencias Sociales*. Madrid, España. Recuperado a partir de <http://web.upcomillas.es/personal/peter/investigacion/Tama%fl0DelEfecto.pdf>
- Morales Vallejo, P. (2013). *Investigación experimental, diseños y contraste de medias. Estadística aplicada a Ciencias Sociales*. Madrid, España. Recuperado a partir de <http://web.upcomillas.es/personal/peter/investigacion/Dise?osMedias.pdf>
- Mullins, P., Whitfield, D., & Conlon, M. (2009). Using Alice 2.0 as a first language. *Journal of Computing Sciences in Colleges*, 24(3), 136-143. Recuperado a partir de <http://dl.acm.org/citation.cfm?id=1409873.1409900>
- Muukkonen, H., Hakkarainen, K., & Lakkala, M. (1999). Collaborative Technology for Facilitating Progressive Inquiry: The Future Learning Environments Tools. En *Proceedings of Computer Supported for Collaborative Learning* (pp. 406-415). Stanford, USA.
- Nam, D., & Lee, T. (2011). The Effect of Robot Programming Education by Pico Cricket on Creative Problem-Solving Skills. En *Proceedings of the 19th International Conference on Computers in Education* (pp. 1-6). Chiang Mai, Thailand: Asia-Pacific Society for Computers in Education: Computers in Education.
- Next is Great. (2012). Move The Turtle: Programming for Kids. Recuperado a partir de <http://movetheturtle.com/>
- Nielsen, C. M., Overgaard, M., Pedersen, M. B., Stage, J., & Stenild, S. (2006). It's worth the hassle!: the added value of evaluating the usability of mobile systems in the field. En *Proceedings of the 4th Nordic conference on Human-computer interaction changing roles - NordiCHI '06* (pp. 272-280). New York, New York, USA: ACM Press. <http://doi.org/10.1145/1182475.1182504>
- Nielsen, J. (1993). *Usability Engineering* (1.^a ed.). Cambridge: Elsevier.
- Nielsen, J. (1994a). Heuristic evaluation: How to Conduct a Heuristic Evaluation. En J.

- Nielsen & R. L. Mack (Eds.), *Usability Inspection Methods*. New York, NY, USA: John Wiley & Sons. Recuperado a partir de <http://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/>
- Nielsen, J. (1994b). Usability inspection methods. En *Conference companion on Human factors in computing systems - CHI '94* (pp. 413-414). New York, New York, USA: ACM Press. <http://doi.org/10.1145/259963.260531>
- Nielsen, J. (2002). Kids' Corner: Website Usability for Children. Recuperado 26 de enero de 2015, a partir de <https://www.nngroup.com/articles/childrens-websites-usability-issues/>
- Nielsen, J. (2003). Usability 101: Introduction to Usability. Recuperado 25 de abril de 2013, a partir de <http://www.useit.com/alertbox/20030825.html>
- Nielsen, J. (2005). 10 Usability Heuristics for User Interface Design. Recuperado 22 de noviembre de 2014, a partir de <https://www.nngroup.com/articles/ten-usability-heuristics/>
- Nielsen, J., & Budiu, R. (2012). *Mobile Usability* (1.^a ed.). New Riders.
- Nielsen, J., & Landauer, T. K. (1993). A mathematical model of the finding of usability problems. En *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '93* (pp. 206-213). New York, New York, USA: ACM Press. <http://doi.org/10.1145/169059.169166>
- Norman, D. A. (1983). Design principles for human-computer interfaces. En *Proceedings of the SIGCHI conference on Human Factors in Computing Systems - CHI '83* (pp. 1-10). New York, New York, USA: ACM Press. <http://doi.org/10.1145/800045.801571>
- Nunan, D. (1993). *Collaborative Language Learning and Teaching*. New York, NY, USA: Cambridge University Press.
- Orsini, L. (2013). Kodable Teaches Kids To Code Before They Learn To Read. Recuperado a partir de <http://readwrite.com/2013/04/23/kodable-teaches-kids-to-code-before-they-learn-to-read#awesm=~oDmcKub1ryv8TN>
- Pane, J. F. (2002). *A programming system for children that is designed for usability*. Carnegie Mellon University. Recuperado a partir de <http://www.cs.cmu.edu/~pane/ftp/PaneThesis.pdf>
- Panitz, T. (1998). *Sí, hay una gran diferencia entre el Paradigma del aprendizaje Cooperativo y el Aprendizaje Colaborativo*. Recuperado a partir de www.lag.itesm.mx/profesores/servicio/congreso/documentos/
- Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. (B. Books, Ed.) (2.^a ed.). Basic Books, Inc. Recuperado a partir de <http://dl.acm.org/citation.cfm?id=1095592>
- Pausch, R., Burnette, T., Capeheart, A. C., Conway, M., Cosgrove, D., DeLine, R., ... White, J. (1995). A Brief Architectural Overview of Alice, a Rapid Prototyping System for Virtual Reality. *IEEE Computer Graphics and Applications*, 15(3), 8-11. Recuperado a partir de <http://www.cs.cmu.edu/~stage3/publications/95/journals/IEEEcomputer/CGandA/pap>

er.html

- Pérez Juste, R., Galán González, A., & Quintanal Díaz, J. (2012). *Métodos y diseños de investigación en educación*. Madrid, España: Editorial UNED.
- Piaget, J. (1985). *Seis estudios de Psicología*. (Editorial Labor S.A., Ed.). México: Origen/Planeta.
- Piaget, J., & Cook, M. (Trans). (1954). *The Construction of Reality in the Child* (1.^a ed.). New York, NY, USA: Basic Books.
- Po, S., Howard, S., Vetere, F., & Skov, M. B. (2004). Heuristic evaluation and mobile usability: bridging the realism gap. En S. Brewster & M. Dunlop (Eds.), *Proceedings of 6th International Symposium, MobileHCI* (pp. 49–60). Glasgow, UK: Springer Berlin Heidelberg.
- Prensky, M. (2001). Digital Natives, Digital Immigrants Part 1. *On the Horizon*, 9(5), 1-6. Recuperado a partir de <http://onlinelibrary.wiley.com/doi/10.1002/cbdv.200490137/abstract>
- Prensky, M. (2003). Digital Game-Based Learning. *Computers in Entertainment*, 1(1), 21. <http://doi.org/10.1145/950566.950596>
- Resnick, M. (1996). StarLogo: an environment for decentralized modeling and decentralized thinking. En *Conference companion on Human factors in computing systems common ground - CHI '96* (pp. 11-12). New York, New York, USA: ACM Press. <http://doi.org/10.1145/257089.257095>
- Resnick, M. (1998). Technologies for Lifelong Kindergarten. *Educational Technology Research and Development*, 46(4), 1-18. Recuperado a partir de <http://web.media.mit.edu/~mres/papers/lifelongk/lifelongk.pdf>
- Resnick, M., Martin, F., Sargent, R., & Silverman, B. (1996). Programmable Bricks: Toys to think with. *IBM Systems Journal*, 35(3.4), 443-452. <http://doi.org/10.1147/sj.353.0443>
- Robo Wunderkind. (2015). Robo Wunderkind: A robot anyone can build. Recuperado 23 de septiembre de 2015, a partir de <https://www.kickstarter.com/projects/startrobo/robo-wunderkind-a-programmable-robot-for-kids-of-a>
- Rollins, M. (2014). *Beginning LEGO MINDSTORMS EV3*. Apress. <http://doi.org/10.1007/978-1-4302-6437-8>
- Romero, S., Angulo, I., Ruíz, I., & Angulo, J. M. (2008). Competencias y Habilidades con el Robot «MOWAY». En *TAAE: Tecnologías Aplicadas a la Enseñanza de la Electrónica* (p. 8).
- Rubin, J., & Chisnell, D. (2008). *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests* (2.^a ed.). Wiley.
- Ruíz del Solar, J., & Salazar, R. (2000). *Introducción a la Robótica*. Santiago, Chile.
- Santoro, F. M., Borges, M. R. S., & Santos, N. (1999). Towards a model for developing a cooperative learning environment. *Proceedings of TISE*, 99.

- Sesame Workshop. (2012). Best Practices: Designing Touch Tablet Experiences for Preschoolers. Recuperado a partir de http://www.sesameworkshop.org/wp_install/wp-content/uploads/2013/04/Best-Practices-Document-11-26-12.pdf
- Setapen, A. (2012). *Creating Robotic Characters for Long-term Interaction*. Massachusetts Institute of Technology. Recuperado a partir de <http://robotic.media.mit.edu/wp-content/uploads/sites/14/2015/01/SetapenSMThesis12.pdf>
- Shackel, B. (1991). Usability: Context, Framework, Definition, Design and Evaluation. En B. Shackel & S. Richardson (Eds.), *Human Factors for Informatics Usability* (pp. 21-38). Cambridge: Cambridge University Press.
- Shapiro, D. (2013). Robot Turtles: The Board Game for Little Programmers. Recuperado a partir de <http://www.robotturtles.com/>
- Slavin, R. (1991). Synthesis of Research on Cooperative Learning. *Educational Leadership*, 48(5), 71-82.
- Stahl, G. (1999). Reflections on WebGuide: Seven Issues for the Next Generation of Collaborative Knowledge Building Environments. En *Proceedings of Computer Supported for Collaborative Learning* (pp. 600-610). Stanford, USA.
- Stahl, G., Koschmann, T., & Suthers, D. (2006). Computer-supported collaborative learning: An historical perspective. En R. K. Sawyer (Ed.), *Cambridge handbook of the learning sciences* (pp. 409-426). Cambridge, UK: Cambridge University Press. Recuperado a partir de http://gerrystahl.net/cscl/CSCL_English.htm
- Tempel, M. (2012). Logo: A Language for All Ages. En P. Phillips (Ed.), *Computer Science K-8: Building a Strong Foundation* (pp. 16-17). New York, New York, USA: ACM. Recuperado a partir de http://www.csta.acm.org/Curriculum/sub/CurrFiles/CS_K-8_Building_a_Foundation.pdf
- Thomas, D. (2013). KineScript: Visual Programming. Recuperado 5 de junio de 2014, a partir de <http://www.graphite.org/app/kinescript-visual-programming>
- Tomoyose, G. (2014, mayo 9). miniBloq, el lenguaje de programación argentino para robots que llega a todo el mundo. *LA NACION*, p. 1. Argentina. Recuperado a partir de <http://www.lanacion.com.ar/1688842-minibloq-el-lenguaje-de-programacion-para-robots-con-sello-argentino>
- Umaschi Bers, M., & Horn, M. S. (2010). Tangible Programming in Early Childhood: Revisiting Developmental Assumptions through New Technologies. *High-tech tots: Childhood in a digital world*.
- Underwood, G., Mc.Caffrey, M., & Underwood, J. (1990). Gender differences in a cooperative computer-based language task. *Educational Research*, 32(1), 44-49.
- Verdugo, R., Barros, L., Albornoz, D., Nussbaum, M., & McFarlane, A. (2014). Scripting For Collaborative Search Computer-Supported Classroom Activities. *Educational Technology & Society*, 17(3), 308-319. Recuperado a partir de http://www.jstor.org/stable/jeductechsoci.17.3.308#pdf_only_tab_contents
- Verner, I. M., & Ahlgren, D. J. (2004). Conceptualizing Educational Approaches in

- Introductory Robotics. *International Journal of Electrical Engineering Education*, 41(3), 183-201.
- Vidanes-Díez, J. (2007). La educación para la paz y la no violencia. *Revista Iberoamericana de Educación*, 42(2). Recuperado a partir de <http://www.rieoei.org/experiencias146.htm>
- Vygotsky, L. S. (1978). *El desarrollo de los procesos psicológicos superiores*. Madrid, España: Editorial Grijalbo.
- Vygotsky, L. S. (2010). *Pensamiento y Lenguaje* (2.ª ed.). Paidós Iberica.
- Vygotsky, L. S., Leontiev, A., & Luria, A. R. (2011). *Psicología y Pedagogía*. (AKAL, Ed.) (4.ª ed.).
- Wang, D., Qi, Y., & Zhang, L. (2015). A Tangible Programming System Conveying Event Handling Concept. En *Proceedings of the 14th International Conference on Interaction Design and Children - IDC '15* (pp. 319-322). Medford, MA, USA: ACM Press. <http://doi.org/10.1145/2771839.2771906>
- Williams, M., & Burden, R. L. (1999). *Psychology for Language Teachers: A Social Constructivist Approach*. New York, NY, USA: Cambridge University Press. Recuperado a partir de http://www.langtoninfo.com/web_content/9780521498807_frontmatter.pdf
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35. <http://doi.org/10.1145/1227504.1227378>
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 366(1881), 3717-25. <http://doi.org/10.1098/rsta.2008.0118>
- Wonder Workshop. (2014a). Play-i is now Wonder Workshop. Recuperado 26 de septiembre de 2014, a partir de <http://blog.makewonder.com/play-now-wonder-workshop/>
- Wonder Workshop. (2014b). Wonder Workshop. Recuperado 30 de octubre de 2014, a partir de <https://www.makewonder.com/robots/dashanddot>
- Wyeth, P., & Purchase, H. C. (2002). Tangible Programming Elements for Young Children. En *CHI '02 extended abstracts on Human factors in computing systems - CHI '02* (pp. 774-775). Minneapolis, Minnesota, USA: ACM Press. <http://doi.org/10.1145/506443.506591>
- Yacob, F., Loglio, M., Di Cuia, D., Leonardi, V., Rabago, L., & Valman, J. (2013). PRIMO. Recuperado 5 de mayo de 2014, a partir de <http://primo.io/>
- Yera-Gil, A. (2010). *Iniciación a la programación visual e interactiva y desarrollo de robótica educativa con Scratch y Enchanting*. Universidad Pública de Navarra. Recuperado a partir de <http://academica-e.unavarra.es/handle/2454/10051>
- Zúñiga, M. (2003). Aprendizaje mediado por tecnologías digitales: la experiencia de Costa Rica. En *Educación y Nuevas Tecnologías* (pp. 99-114). Buenos Aires, Argentina: International Institute for Educational Planning. UNESCO. Recuperado a partir de http://www.fod.ac.cr/pdf/publicaciones/articulos/2003/mz_2003_aprendizaje_mediado

.pdf

Zurbano Díaz de Cerio, J. L. (1998). *Bases de una Educación para la Paz y la Convivencia*. (Gobierno de Navarra - Departamento de Educación y Cultura, Ed.). Fondo de Publicaciones del Gobierno de Navarra - Departamento de Presidencia. Recuperado a partir de http://www.comisionunesco.mec.gub.uy/innovaportal/file/29930/1/bases_de_una_ed_para_la_paz_y_la_convivencia.pdf

APÉNDICES

APÉNDICE 1. DIVULGACIÓN DE LA INVESTIGACIÓN

Conferencias Científicas

(Ponencia y artículo publicado en las memorias de REV2014, indexado por IEEE Xplore)

Ramírez-Benavides, Kryscia y Guerrero, Luis A. Programación de robots para niños de la primera infancia con edades comprendidas entre 4-6 años. REV2014: 11th International Conference on Remote Engineering and Virtual Instrumentation. Special Track on Experimental Teaching based on Remote and Virtual Labs. Porto, Portugal, Febrero 26-28, 2014.

(Ponencia y artículo publicado en las memorias de TEEM'15, indexado por ACM)

Ramírez-Benavides, Kryscia; García, Franklin y Guerrero, Luis A. Creating a protocol for collaborative mobile applications for kids between 4 and 6 years old. TEEM'15: Proceedings of the 3rd International Conference on Technological Ecosystems for Enhancing Multiculturality. Páginas: 317-324. ACM, New York, NY, USA, 2015. ISBN: 978-1-4503-3442-6. URL: <http://doi.acm.org/10.1145/2808580.2808628>.

Revistas Científicas

(Artículo publicado en VAEP-RITA)

Ramírez-Benavides, Kryscia y Guerrero, Luis A. MODEBOTS: Entorno de programación de robots para niños de la primera infancia con edades comprendidas entre 4-6 años. VAEP-RITA: Versión Abierta Español-Portugués – Revista Iberoamericana de Tecnologías del Aprendizaje. Vol. 2, Num. 3. Páginas: 139-146. Septiembre, 2014. URL: http://rita.det.uvigo.es/VAEPRITA/index.php?content=Num_Pub&idiom=Es&visualiza=1&volumen=2&numero=3&orden=desc.

(Artículo publicado en IEEE-RITA, indexado por IEEE Xplore, DBLP, Ulrichs y SCOPUS, SJR = 0.1)

Ramírez-Benavides, Kryscia y Guerrero, Luis A. MODEBOTS: Environment for Programming Robots for Children between the ages of 4 and 6. IEEE-RITA: Revista Iberoamericana de Tecnologías del Aprendizaje. Vol. 10, Num. 3. Páginas: 152-159. Octubre, 2015. ISSN: 192-8540. DOI: 10.1109/RITA.2015.2452692. URL: <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7155494>.

Proyectos de Investigación

Ramírez-Benavides, Kryscia y Guerrero, Luis A. Creación de un entorno de programación en dispositivos móviles orientado a niños entre 4 y 6 años que permita programar aplicaciones para robots. Proyecto de Investigación No.: 834-B3-260. Inscrito en el Centro de Investigaciones en Tecnologías de la Información y Comunicación (CITIC) y en la Escuela de Ciencias de la Computación e Informática (ECCI), Universidad de Costa Rica (UCR). Estado del Proyecto: Finalizado. Vigencia: Julio 1, 2013 – Junio 30, 2015.

Ramírez-Benavides, Kryscia y Guerrero, Luis A. Creación de un entorno de programación colaborativo orientado a niños entre 4 y 6 años que fomente el pensamiento lógico-matemático y la solución de problemas. Proyecto de Investigación No.: 834-B5-A04. Inscrito en el Centro de Investigaciones en Tecnologías de la Información y Comunicación (CITIC) y en la Escuela de Ciencias de la Computación e Informática (ECCI), Universidad de Costa Rica (UCR). Estado del Proyecto: Activo. Vigencia: Julio 1, 2015 – Junio 30, 2017.

Ramírez-Benavides, Kryscia; López, Gustavo y Guerrero, Luis A. TITIBOTS: Entorno que apoya el proceso de aprendizaje colaborativo de la programación en niños con edades comprendidas entre 4 y 6 años. Proyecto ganador del 2^{do} lugar de la Categoría General. VIII Edición del Premio Ideas Innovadoras 2014. Cátedra Innovación y Desarrollo Empresarial UCR-CICR. San José, Costa Rica. Noviembre 20, 2014.

Otras Publicaciones y Conferencias

(Ponencia en Consejo Doctoral UCAmI 2013)

Ramírez-Benavides, Kryscia. Programación de robots para niños de la primera infancia con edades comprendidas entre 4-6 años. UCAmI 2013: 7th International Conference on Ubiquitous Computing & Ambient Intelligence. Doctoral Consortium. Guanacaste, Costa Rica, Diciembre 2-6, 2013

(Ponencia y artículo publicado en las memorias de JoCICI 2015)

Jadán-Guerrero, Janio; Ramírez-Benavides, Kryscia y Romero-Bonilla, Shirley. Reconocimiento facial a través de la edad: aspectos de percepción y cognición social. II Jornadas Costarricenses de Investigación en Computación e Informática (JoCICI 2015). Universidad de Costa Rica, San José, Costa Rica, Marzo 4-5, 2015.

(Ponencia en IX Conferencia Internacional GUIDE)

Nicolete, Priscila Cadorin; da Silva Cristiano, Marta Adriana; da Silva, Juarez Bento; Sommer Bilessimo, Simone Meister y Ramírez-Benavides, Kryscia. Integración de Tecnologías en la Educación Básica de Brasil: Un Estudio de Caso en la Enseñanza de la Matemática. IX Conferencia Internacional GUIDE: Educación y sociedad en red – Los desafíos de la era digital. Universidad del Salvador (USAL), Ciudad de Buenos Aire, Argentina, Mayo 6-8, 2015.

(Ponencia y artículo publicado en las memorias de TEEM'15, indexado por ACM)

Nicolete, Priscila Cadorin; da Silva Cristiano, Marta Adriana; da Silva, Juarez Bento; da Luz Filho, Silvio Serafim; Sommer Bilessimo, Simone Meister; Schardosim Simão, José Pedro y Ramírez-Benavides, Kryscia. Mathematics and technological integration in the Brazilian basic education as motivation to STEM. TEEM'15: Proceedings of the 3rd International Conference on Technological Ecosystems for Enhancing Multiculturality. Páginas: 591-598. ACM, New York, NY, USA, 2015. ISBN: 978-1-4503-3442-6. URL: <http://dl.acm.org/citation.cfm?doid=2808580.2808671>.

Participación en Eventos

SUPER Women in Technology – SUPER Women Costa Rica. IBM. Heredia, Costa Rica. Marzo 26, 2014.

Semana TEC. Sede Regional del ITCR. San Carlos, Costa Rica. Abril 7-8, 2014.

RExLab – UFSC. Presentación de la investigación a los estudiantes del Programa de PósGraduação em Tecnologias da Informação e Comunicação da UFSC (PPGTIC) y a los investigadores del RExLab. Universidade Federal de Santa Catarina, Araranguá-SC, Brasil. Marzo 10, 2015. RExLab UFSC – Blog: “Doutoranda da Costa Rica em estágio no RExLab apresenta projeto à turma do PPGTIC”. URL.: <http://rexlab.ufsc.br/blog/>.

Reunión entre el equipo del RExLab, las autoridades de la Prefeitura de Araranguá y la investigadora Kryscia Ramírez, en la cual se discutió asuntos relacionados a la labor del RExLab y el desarrollo de la investigación en las escuelas públicas de la ciudad de Araranguá. RExLab UFSC – Blog: “Prefeitura de Araranguá realiza reunião com o RExLab”. Abril 22, 2015. URL.: <http://rexlab.ufsc.br/blog/>.

Genuino DAY 2016. Universidad de Costa Rica. San Pedro de Montes de Oca, San José, Costa Rica. Abril 2, 2016.

Medios de Comunicación Nacionales e Internacionales

Reportaje en el Journal Enfoque Popular: “Doutoranda da Costa Rica desenvolve projeto de robótica com alunos do pré-escolar”. Mayo 5, 2015. Brasil.

Entrevista en Rádio Araranguá, programa Estúdio 1290, sobre la investigación realizada en Araranguá junto al RExLab-UFSC. Mayo 5, 2015. Brasil. URL: <http://www.radioararangua.com.br/index.php?action=internaVariedades&codigo=6132>.

Publicación en el Blog del RExLab sobre el desarrollo de la investigación con alumnos de preescolar de las escuelas públicas de la ciudad de Araranguá. RExLab UFSC – Blog: “Doutoranda da Costa Rica em estágio no RExLab desenvolve projeto de robótica com alunos do pré-escolar”. Mayo 6, 2015. Brasil. URL.: <http://rexlab.ufsc.br/blog/>.

Reportaje en el Journal O Tempo: “Doutoranda da Costa Rica desenvolve projeto com alunos de Araranguá”. Mayo 7, 2015. Brasil.

Reportaje en el Journal Correio do Sul: “Robótica na pré-escola”. Mayo 9, 2015. Brasil.

Entrevista en ARTV, programa Revista no Ar, sobre la investigación realizada en Araranguá junto al RExLab-UFSC. Mayo 15, 2015. Brasil. URL: <http://www.artv.com.br/revista-no-ar/2453/ufsc-alia-tecnologia-com-educacao.html>.

Publicaciones en el Fabebook del RExLab UFSC sobre los diferentes eventos de divulgación de mi investigación y su desarrollo en las escuelas públicas de Araranguá. Marzo-Junio, 2015. Brasil. URL: <https://www.facebook.com/rexlabARA?fref=ts>.

Reportaje en el Portal de la Investigación (Vicerrectoría de Investigación – UCR): “Preescolares programan robots antes de aprender a leer”. Noviembre 5, 2015. Costa Rica. URL: http://www.vinv.ucr.ac.cr/index.php?option=com_content&view=article&id=2580:preescolares-programan-robots-antes-de-aprender-a-leer&catid=1&Itemid=68.

Reportaje en el Sitio Web de Noticias Amelia Rueda: “Jóvenes de la UCR crean novedoso software para que preescolares puedan programar robots en grupos”. Noviembre 10, 2015. Costa Rica. URL: <http://www.ameliarueda.com/nota/joven-ucr-crea-novedoso-software-preescolares-puedan-programar-robots>.

Reportaje en el Periódico La Segunda: “Preescolares programan robots antes de aprender a leer”. Noviembre 12, 2015. Costa Rica.

Reportaje en el Periódico La Nación: “Niños aprenden a programar en grupo antes que a escribir”. Noviembre 23, 2015. Costa Rica. URL: http://www.nacion.com/beta/tecnologia/informatica/Ninos-aprenden-programar-grupo-escribir_0_1526047399.html.

Reportaje en el Periódico Semanario Universidad, Suplemento C+T Ciencia Más Tecnología: “Aprender a programar en equipo”. Mayo 11-17, 2016. Costa Rica. URL: https://issuu.com/ct.ucr/docs/c_t_6_mayo_2016.

APÉNDICE 2. REFERENCIAS DE LA REVISIÓN SISTEMÁTICA DE LITERATURA

- [1] D.-Y. Kwon, H.-S. Kim, J.-K. Shim, y W.-G. Lee, “Algorithmic Bricks: A Tangible Robot Programming Tool for Elementary School Students”, *IEEE Trans. Educ.*, vol. 55, n.o 4, pp. 474-479, nov. 2012.
- [2] C. Jost, B. Le Pevedic, y D. Duhaut, “ArCo: An architecture for children to control a set of robots”, en *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, 2012, pp. 158-164.
- [3] H. S. Raffle, A. J. Parkes, y H. Ishii, “Topobo: a constructive assembly system with kinetic memory”, en *Proceedings of the 2004 conference on Human factors in computing systems - CHI’04*, 2004, pp. 647-654.
- [4] H. Raffle, A. Parkes, H. Ishii, y J. Lifton, “Beyond Record and Play: Backpacks: Tangible Modulators for Kinetic Behavior”, en *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI’06*, 2006, pp. 681-690.
- [5] M. Nishiguchi y K. Sakamoto, “Easy Robot Programming for Beginners and Kids Using Command and Instruction Marker Card”, en *Entertainment Computing - ICEC 2010*, vol. 6243, H. S. Yang, R. Malaka, J. Hoshino, y J. H. Han, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 460-463.
- [6] A. Strawhacker y M. U. Bers, “I want my robot to look for food”: Comparing Kindergartner’s programming comprehension using tangible, graphic, and hybrid user interfaces”, *Int. J. Technol. Des. Educ.*, vol. 25, n.o 3, pp. 293-319, sep. 2014.
- [7] P. Wyeth y H. C. Purchase, “Programming without a computer: a new interface for children under eight”, en *Proceedings First Australasian User Interface Conference. AUIC 2000 (Cat. No.PR00515)*, 2000, pp. 141-148.
- [8] P. Wyeth y G. Wyeth, “Electronic Blocks: Tangible Programming Elements for Preschoolers”, en *Proceedings of IFIP (International Federation for Information Processing) TC13 Interact’01 - INTERACT: IFIP International Conference on Human-Computer Interaction*, 2001, pp. 496-503.
- [9] P. Wyeth y H. C. Purchase, “Tangible Programming Elements for Young Children”, en *CHI’02 extended abstracts on Human factors in computing systems – CHI’02*, 2002, pp. 774-775.
- [10] A. C. Smith, “Symbols for children’s tangible programming cubes”, en *Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers’ Association on – SACLA’09*, 2009, pp. 105-109.
- [11] A. C. Smith, “Simple tangible language elements for young children”, en *Proceedings of the 8th International Conference on Interaction Design and Children - IDC’09*, 2009, p. 288.

- [12] S. Ludi, "A Tablet-Based Lego Mindstorms Programming Environment for Children", en *Learning and Collaboration Technologies*, vol. 9192, P. Zaphiris y A. Ioannou, Eds. Cham: Springer International Publishing, 2015, pp. 106-114.
- [13] A. Sullivan, M. Elkin, y M. U. Bers, "KIBO Robot Demo: Engaging Young Children in Programming and Engineering", en *Proceedings of the 14th International Conference on Interaction Design and Children - IDC'15*, 2015, pp. 418-421.
- [14] R. Goldman, M. Q. Azhar, y E. Sklar, "From RoboLab to Aibo: A Behavior-Based Interface for Educational Robotics", en *RoboCup 2006: Robot Soccer World Cup X*, vol. 4434, G. Lakemeyer, E. Sklar, D. G. Sorrenti, y T. Takahashi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 122-133.
- [15] J. Gindling, A. Ioannidou, J. Loh, O. Lokkebo, y A. Repenning, "LEGOsheets: a rule-based programming, simulation and manipulation environment for the LEGO Programmable Brick", en *Proceedings of Symposium on Visual Languages*, 1995, pp. 172-179.
- [16] A. Karatrantou y C. Panagiotakopoulos, "Educational Robotics and Teaching Introductory Programming Within an Interdisciplinary Framework", en *Research on e-Learning and ICT in Education*, A. Jimoyiannis, Ed. New York, NY: Springer New York, 2012, pp. 197-210.
- [17] K. Ramirez-Benavides y L. A. Guerrero, "MODEBOTS: Environment for Programming Robots for Children Between the Ages of 4 and 6", *IEEE Rev. Iberoam. Tecnol. del Aprendiz.*, vol. 10, n.o 3, pp. 152-159, ago. 2015.
- [18] G. B. Demo, G. Marcianò, y S. Siega, "Concrete Programming: Using Small Robots in Primary Schools", en *2008 Eighth IEEE International Conference on Advanced Learning Technologies*, 2008, pp. 301-302.
- [19] M. D. Lytras, J. M. Carroll, E. Damiani, y R. D. Tennyson, Eds., "Programming Robots in Primary Schools Deserves a Renewed Attention", en *Emerging Technologies and Information Systems for the Knowledge Society*, vol. 5288, Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 322-331.
- [20] G. B. Demo, S. Siega, y M. S. De Michele, "University and Primary Schools Cooperation for Small Robots Programming", en *Visioning and Engineering the Knowledge Society. A Web Science Perspective*, vol. 5736, M. D. Lytras, E. Damiani, J. M. Carroll, R. D. Tennyson, D. Avison, A. Naeve, A. Dale, P. Lefrere, F. Tan, J. Sipior, y G. Vossen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 238-247.
- [21] S. Kakehashi, T. Motoyoshi, K. Koyanagi, T. Ohshima, y H. Kawakami, "P-CUBE: Block Type Programming Tool for Visual Impairments", en *2013 Conference on Technologies and Applications of Artificial Intelligence*, 2013, pp. 294-299.
- [22] D. J. Christensen, R. Fogh, y H. H. Lund, "Playte, a tangible interface for engaging human-robot interaction", en *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, 2014, pp. 56-62.
- [23] T. Sapounidis y S. N. Demetriadis, "Exploring Children Preferences regarding Tangible and Graphical Tools for Introductory Programming: Evaluating the

- PROTEAS Kit”, en 2012 IEEE 12th International Conference on Advanced Learning Technologies, 2012, pp. 316-320.
- [24] T. Sapounidis y S. Demetriadis, “Tangible versus graphical user interfaces for robot programming: exploring cross-age children’s preferences”, *Pers. Ubiquitous Comput.*, vol. 17, n.o 8, pp. 1775-1786, mar. 2013.
- [25] T. Sapounidis, S. Demetriadis, y I. Stamelos, “Evaluating children performance with graphical and tangible robot programming tools”, *Pers. Ubiquitous Comput.*, vol. 19, n.o 1, pp. 225-237, ene. 2015.
- [26] A. F. Silva, R. P. Barros, S. O. Azevedo, A. Silva, y L. M. G. Gonçalves, “Diagnostic Robotic Agent in the RoboEduc Environment for Educational Robotics”, en 2008 IEEE Latin American Robotic Symposium, 2008, pp. 131-136.
- [27] S. T. Levy y D. Mioduser, “Does it “want” or “was it programmed to...”? Kindergarten children’s explanations of an autonomous robot’s adaptive functioning”, *Int. J. Technol. Des. Educ.*, vol. 18, n.o 4, pp. 337-359, abr. 2007.
- [28] D. Mioduser y S. T. Levy, “Making Sense by Building Sense: Kindergarten Children’s Construction and Understanding of Adaptive Robot Behaviors”, *Int. J. Comput. Math. Learn.*, vol. 15, n.o 2, pp. 99-127, jun. 2010.
- [29] S. T. Levy y D. Mioduser, “Approaching Complexity Through Planful Play: Kindergarten Children’s Strategies in Constructing an Autonomous Robot’s Behavior”, *Int. J. Comput. Math. Learn.*, vol. 15, n.o 1, pp. 21-43, abr. 2010.
- [30] H. H. Lund y L. Pagliarini, “RoboCup Jr. with LEGO MINDSTORMS”, en Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065), 2000, vol. 1, pp. 813-819.
- [31] E. Wang y R. Wang, “Using Legos and RoboLab (LabVIEW) with elementary school children”, en 31st Annual Frontiers in Education Conference. Impact on Engineering and Science Education. Conference Proceedings (Cat. No.01CH37193), 2001, vol. 1, p. T2E-T11.
- [32] M. Bers, C. Rogers, L. Beals, M. Portsmore, K. Staszowski, E. Cejka, A. Carberry, B. Gravel, J. Anderson, y M. Barnett, “Innovative Session: Early Childhood Robotics for Learning”, en ICLS’06 Proceedings of the 7th international conference on Learning sciences, 2006, pp. 1036-1042.
- [33] A. Karoulis, “Evaluating the LEGO--RoboLab Interface with Experts”, *ACM Comput. Entertain.*, vol. 4, n.o 2, pp. 1-13, abr. 2006.
- [34] L. Uribe y A. Eguchi, “4th Graders creating robots with sensors”, en 2015 IEEE Integrated STEM Education Conference, 2015, pp. 126-128.
- [35] K. A. Mills, V. Chandra, y J. Y. Park, “The architecture of children’s use of language and tools when problem solving collaboratively with robotics”, *Aust. Educ. Res.*, vol. 40, n.o 3, pp. 315-337, abr. 2013.
- [36] M. Ucgul y K. Cagiltay, “Design and development issues for educational robotics training camps”, *Int. J. Technol. Des. Educ.*, vol. 24, n.o 2, pp. 203-222, nov. 2013.

- [37] M. Gordon, E. Rivera, E. Ackermann, y C. Breazeal, “Designing a Relational Social Robot Toolkit for Preschool Children to Explore Computational Concepts”, en Proceedings of the 14th International Conference on Interaction Design and Children - IDC’15, 2015, pp. 355-358.
- [38] A. Smith, “Tangible Cubes as Programming Objects”, en 16th International Conference on Artificial Reality and Telexistence--Workshops (ICAT’06), 2006, pp. 157-161.
- [39] D. Wang, C. Zhang, y H. Wang, “T-Maze: A Tangible Programming Tool for Children”, en Proceedings of the 10th International Conference on Interaction Design and Children - IDC’11, 2011, pp. 127-135.
- [40] D. Wang, Y. Qi, y L. Zhang, “A Tangible Programming System Conveying Event Handling Concept”, en Proceedings of the 14th International Conference on Interaction Design and Children - IDC’15, 2015, pp. 319-322.
- [41] M. S. Horn, E. T. Solovey, y R. J. K. Jacob, “Tangible Programming and Informal Science Learning: Making TUIs Work for Museums”, en Proceedings of the 7th international conference on Interaction design and children - IDC’08, 2008, pp. 194-201.
- [42] M. S. Horn, “Tangible Computer Programming for Informal Science Learning”, en Proceedings of the 7th international conference on Interaction design and children - IDC’08, 2008, pp. 21-24.
- [43] M. S. Horn, M. U. Bers, y R. J. K. Jacob, “Tangible Programming in Education: A Research Approach”, en Conference on Human Factors in Computing Systems (CHI’2009), 2009, pp. 1-4.
- [44] M. S. Horn, R. J. Crouser, y M. U. Bers, “Tangible interaction and learning: the case for a hybrid approach”, Pers. Ubiquitous Comput., vol. 16, n.o 4, pp. 379-389, jun. 2011.
- [45] K. Ramírez-Benavides, F. García, y L. A. Guerrero, “Creating a protocol for collaborative mobile applications for kids between 4 and 6 years old”, en Proceedings of the 3rd International Conference on Technological Ecosystems for Enhancing Multiculturality - TEEM’15, 2015, pp. 317-324.
- [46] N. Rudolph, “TOTA: A Construction Set for the Impending Apocalypse”, en Proceedings of the 12th International Conference on Interaction Design and Children - IDC’13, 2013, pp. 554-556.

APÉNDICE 3. GUÍA DE VALIDACIÓN DE TITIBOTS

Primera Prueba

Los diez íconos (comandos) se les dan a todos los niños.

¿Qué entiende el niño o la niña por cada ícono (comando)?

Ícono	Entendió (Sí / No)
Inicio	
Fin	
Adelante	
Atrás	
Izquierda	
Derecha	
Encender	
Apagar	
Agarrar	
Soltar	

Segunda Prueba

Los tres tipos de interfaz se les dan a todos los niños.

¿Entiende el niño o la niña la secuencia de comandos en la interfaz dada?

Interfaz	Entendió (Sí / No)	Comandos Correctos	Comandos Incorrectos
Huellas			
Líneas			
Sin camino			

Tercera Prueba

El conjunto de pasos de la aplicación se le da a un conjunto de 13 niños, se tendrá tres grupos de 13 niños (aproximadamente), para que cada grupo tenga un tipo de interacción diferente: Inserción, Arrastrar y soltar, y Guiado.

¿Lograron completar el conjunto de pasos de la aplicación? ¿Fue fácil realizar el tipo de interacción?

Interacción	Completó (Sí / No)	Fue fácil (Sí / No)
Inserción		
Arrastrar y soltar		
Guiado		

Comentarios:

APÉNDICE 4. GUÍA DE OBSERVACIÓN DE TITIBOTS

Fecha: _____ Hora: _____ Lugar: _____

Nombre del observador: _____

Nombre del niño: _____

Inicio de la sesión	
Interés del niño o la niña por la actividad	
Desarrollo de la sesión	
Respuesta del niño o la niña ante la presentación de los contenidos (explicación)	
Respuesta del niño o la niña ante los ejercicios dados por el(la) mediador(a) u otros niños durante la explicación del entorno.	
El niño o la niña realiza los ejercicios, los realiza correctamente	
Los retos quedaron claros para el niño o la niña	

<p>El niño o la niña interactúa con otros niños Describa la interacción, si la hay</p>	
<p>Describa los comportamientos, actitudes, emociones, gestos, lo que dice, etc. del niño o niña a la hora de usar el entorno de programación</p>	
<p>Indique: ¿Cuántos retos logra realizar el niño o la niña? ¿Cuánto tiempo (en segundos) le tomó cada reto? ¿Cuántos intentos realiza por reto?</p>	
<p>Indique (si hubo errores): ¿El niño o la niña cometieron errores? ¿Cuántos errores cometieron por reto? ¿Los errores cometidos fueron por un asunto cognitivo, por falta de entendimiento del entorno, o por el entorno en sí?</p>	
Final de la sesión	
<p>Describa los comportamientos, actitudes, emociones, gestos, lo que dice, etc. del niño o niña a la hora de realizar la encuesta con el(la) mediador(a)</p>	
<p>¿Cuál es su opinión sobre la percepción del niño o niña del desarrollo de la actividad?</p>	

APÉNDICE 5. ENTREVISTA SEMIESTRUCTURADA DE TITIBOTS – PERFIL DE USUARIO

Sesión: _____

Fecha: _____

Pregunta	Niña 1	Niña 2	Niño 1	Niño 2
¿En tu casa hay tabletas o <i>smartphones</i> ?				
¿Tus papás te dan permiso de usar la tableta o <i>smartphone</i> en tu casa?				
¿Tienes tu propia tableta o <i>smartphone</i> ?				

Sesión: _____

Fecha: _____

Pregunta	Niña 1	Niña 2	Niño 1	Niño 2
¿En tu casa hay tabletas o <i>smartphones</i> ?				
¿Tus papás te dan permiso de usar la tableta o <i>smartphone</i> en tu casa?				
¿Tienes tu propia tableta o <i>smartphone</i> ?				

APÉNDICE 6. CUESTIONARIO DE SATISFACCIÓN DE TITIBOTS

Nombre: _____

Edad: _____

Sexo: _____

1. Señala la carita que más representa lo que piensas en relación si te gustó el juego:



No me gustó



Me gustó

2. Señala la carita que más representa lo que piensas en relación si fue fácil usar el juego:



No me gustó



Me gustó

3. Señala la carita que más representa lo que piensas en relación si fue fácil usar el robot:



No me gustó



Me gustó

4. Dibuja con lápiz y papel lo que más te gustó del taller.

APÉNDICE 7. ENTREVISTA SEMIESTRUCTURADA DE TITIBOTS COLAB – PERFIL DE USUARIO

Nombre del encargado: _____
 Fecha: _____
 Nombre completo del niño: _____
 Edad: _____ Sexo: _____

1. ¿En su casa hay dispositivos móviles: *smartphone*, tableta (iPad, Android, otra), iPod Touch (similar), cualquier otro dispositivo móvil?
 Sí _____ No _____

2. Si la respuesta a la pregunta anterior es afirmativa: ¿Cuántos dispositivos móviles tienen?

3. ¿Usted le da permiso al niño o la niña de usar algún dispositivo móvil?
 Sí _____ No _____

4. Si la respuesta a la pregunta anterior es afirmativa: ¿Cuánto tiempo usa el niño o la niña el dispositivo móvil?

5. ¿Qué tipo de aplicaciones usa el niño o la niña cuando usa el dispositivo móvil?
 _____ Juegos educativos
 _____ Juegos
 _____ Aplicaciones creativas (dibujar, colorear, entre otros)
 _____ Aplicaciones de programas infantiles de TV
 _____ Otros:

6. ¿El niño o la niña tiene su propio dispositivo móvil?
 Sí _____ No _____

7. Si la respuesta a la pregunta anterior es afirmativa: ¿Qué tipo de dispositivo móvil tiene su hijo o su hija?
 _____ Smartphone
 _____ Tableta (iPad, Android, otra)
 _____ iPod Touch (similar)
 _____ Otro:

APÉNDICE 8. PRUEBA DE MEMORIA DE TITIBOTS Y TITIBOTS COLAB

La docente realiza una serie de preguntas en plenaria y los niños contestan. Las preguntas están relacionadas a aspectos del diseño del entorno, por ejemplo: el personaje del entorno, el robot utilizado, los colores de los comandos, lo que significan los comandos, etc.

1. ¿Cuál fue la manera en que se trabajó?
2. ¿Qué es un robot?
3. ¿Cuáles son los pasos para la solución de un problema?
4. ¿En dónde se planifica?
5. ¿En dónde se implementa?
6. ¿En dónde se prueba?
7. ¿Quién le dice al robot Tití qué hacer? ¿Por qué medio se le dice qué hacer?
8. ¿Cuáles son las acciones que puede hacer el robot Tití?
9. Enseñar los comandos y preguntar para cada uno: ¿Qué le permite hacer este comando al robot Tití?
10. Enseñar diferentes elementos de la aplicación y preguntar para cada uno: ¿Para qué era este botón? o ¿Qué hacía este ícono?

APÉNDICE 9. CUESTIONARIO DE SATISFACCIÓN DE TITIBOTS COLAB

Nombre: _____

Edad: _____ Sexo: _____

Pinta la carita que más representa lo que piensas en relación:

1. ¿Te gustó el juego?



No



Sí

2. ¿Fue fácil usar el juego?



No



Sí

3. ¿Te gustó trabajar con tus compañeros de equipo?



No



Sí

4. ¿Fue importante tu participación en el equipo?



No



Sí

5. Dibuja lo que más te gustó del taller.

APÉNDICE 10. ENTREVISTA SEMIESTRUCTRADA DE LAS DOCENTES – PERFIL ACADÉMICO Y LABORAL

Nombre: _____

Edad: _____

Responda las siguientes preguntas, de acuerdo al perfil académico y laboral, tanto de la institución como el suyo.

1. ¿Cuál es la misión de la Institución?
2. ¿Cuál es la visión de la Institución?
3. ¿Cuál fue la Universidad en la que se graduó?
4. ¿Cuál fue el último título universitario obtenido?
5. ¿Cuántos años de experiencia tiene como docente?
6. ¿Cuántos años imparte clases en los niveles de preescolar?
7. ¿Cuántos años tiene de laborar en la Institución como docente?
8. ¿Cuál es la metodología utilizada en preescolar de la Institución?
9. ¿Cuántos años lleva usando la metodología de trabajo?
10. ¿Cómo es el planeamiento didáctico de la Institución?
11. ¿Cómo es la evaluación que se realiza en preescolar de la Institución?
12. ¿Realiza en el aula actividades colaborativas? Si la respuesta es afirmativa: ¿Qué actividades colaborativas realiza en el aula?

APÉNDICE 11. LISTA DE COTEJO PARA LA EVALUACIÓN DEL APRENDIZAJE COLABORATIVO

Institución Educativa: _____

Nombre de la Docente: _____

Nombre de la Actividad Colaborativa: _____

Cantidad de Intentos: _____ Tiempo Total: _____

¿Actividad Lograda? _____ Sí _____ No

Nombre del Equipo Colaborativo: _____

Integrantes del Equipo Colaborativo: _____

N°	Nombre Completo	Edad	Género	Rol Asignado
1				
2				
3				

Nombre del Evaluador: _____

Fecha de la Evaluación: _____

Interdependencia Positiva		Integrante #1		Integrante #2		Integrante #3	
N°	Acción a evaluar	¿Lo logra?		¿Lo logra?		¿Lo logra?	
		Sí	No	Sí	No	Sí	No
1	Acepta la división de tareas establecida (roles)						
2	Identifica las tareas del rol que desempeña						
3	Identifica las tareas de los roles que desempeñan sus compañeros						
4	Reconoce la importancia de su rol para cumplir el objetivo						
5	Reconoce la importancia de los roles del equipo para cumplir el objetivo						
6	Facilita los recursos asignados para cumplir el objetivo						
7	Apoya las decisiones del equipo enfocadas a lograr el objetivo						
8	Logra junto a sus compañeros el objetivo						
9	Ofrece apoyo a sus compañeros						
10	Celebra junto con sus compañeros el éxito						
11	Sobrelleva el fracaso con sus compañeros de forma positiva						
Observaciones							

Interacción estimuladora (comunicación cara a cara)		Integrante #1		Integrante #2		Integrante #3	
N°	Acción a evaluar	¿Lo logra?		¿Lo logra?		¿Lo logra?	
		Sí	No	Sí	No	Sí	No
1	Se comunica e interactúa con sus compañeros						
2	Expresa sus ideas con claridad						
3	Presta atención a las ideas y las aportaciones de sus compañeros						
4	Acepta las ideas y las aportaciones de sus compañeros						
5	Trabaja junto a sus compañeros en las actividades para cumplir el objetivo						
6	Ayuda y asiste a sus compañeros						
7	Anima a la participación de las actividades propuestas						
8	Hace preguntas para obtener aclaraciones						
Observaciones							

Responsabilidad individual y grupal		Integrante #1		Integrante #2		Integrante #3	
N°	Acción a evaluar	¿Lo logra?		¿Lo logra?		¿Lo logra?	
		Sí	No	Sí	No	Sí	No
1	Cumple con las tareas asignadas (de acuerdo a su rol)						
2	Comprende las tareas asignadas a los otros compañeros (roles del equipo)						
3	Cuida los recursos que utiliza						
4	Proporciona retroalimentación a sus compañeros para mejorar las actividades						
Observaciones							

Habilidades interpersonales y grupales (trabajo en equipo)		Integrante #1		Integrante #2		Integrante #3	
N°	Acción a evaluar	¿Lo logra?		¿Lo logra?		¿Lo logra?	
		Sí	No	Sí	No	Sí	No
1	Acepta y apoya a su equipo						
2	Se adapta al equipo de trabajo						
3	Trabaja junto con sus compañeros						
4	Se comunica con sus compañeros claramente						
5	Resuelve conflictos constructivamente (diálogo)						
6	Sigue las reglas establecidas						
7	Pide ayuda cuando lo necesita						
Observaciones							

Evaluación grupal (proceso de grupo)		Integrante #1		Integrante #2		Integrante #3	
N°	Acción a evaluar	¿Lo logra?		¿Lo logra?		¿Lo logra?	
		Sí	No	Sí	No	Sí	No
1	Indica las actividades realizadas por él(ella)						
2	Indica las actividades realizadas por sus compañeros						
3	Indica el trabajo del equipo						
4	Identifica cuáles acciones son útiles para cumplir con el objetivo						
5	Toma decisiones para lograr el objetivo						
Observaciones							

APÉNDICE 12. EVALUACIÓN DE LOS DESAFÍOS

Esta hoja de evaluación es para los desafíos de cada día de evaluación. Son tres desafíos, solo se permitirá que los grupos colaborativos realicen tres intentos por desafío. La docente debe poner un *check* en la columna respectiva a la cantidad de intentos realizados por desafío de cada grupo. Además, se evaluará la colaboración que se da en cada grupo, anote sus observaciones en relación con este aspecto para cada grupo e indique si cree que se logró trabajar colaborativamente.

Nombre del Equipo	Reto 1			Reto 2			Reto 3		
	1	2	3	1	2	3	1	2	3
Equipo Mono									
Equipo Tortuga									
Equipo Delfin									
Equipo Pájaro									
Equipo Cocodrilo									

Nombre del Equipo	¿Se logró el trabajo colaborativo?	Observaciones
Equipo Mono		
Equipo Tortuga		
Equipo Delfin		
Equipo Pájaro		
Equipo Cocodrilo		

APÉNDICE 13. VALIDACIÓN DE LA ICONOGRAFÍA Y LA INTERACCIÓN CON NIÑOS DE PREESCOLAR

El objetivo de esta etapa fue validar el diseño de la interfaz, la iconografía y la interacción de la herramienta de programación TITIBOTS. Por lo cual, se creó un conjunto de instrumentos y se validó la forma, el color, y la posible interacción con una población de 40 niños. Para la validación se llevó a cabo una prueba de laboratorio (Eberts, 1994), donde se realizaron observaciones y se obtuvieron datos cualitativos. El fin de esta validación fue generar una guía de lineamientos de diseño y construcción de herramientas de aprendizaje de la programación orientadas a niños entre 4 y 6 años; y así, obtener una propuesta final de diseño y desarrollar el primer prototipo funcional.

Participantes

Esta validación se llevó a cabo en la Escuela San Bosco de Mora con dos grupos de niños entre 4 y 6 años de edad: Ciclo Materno Infantil y Ciclo de Transición. Se contó con dos mediadoras (docentes de preescolar), tres evaluadores y una persona responsable de aspectos logísticos (ver Figura 63).

Espacio Físico y Materiales

La validación se llevó a cabo en el pasillo de las aulas de preescolar de la Escuela, donde se colocaron tres mesas (una por evaluador), con 1 metro de distancia entre ellas, para trabajar con los niños (ver Figura 63). Los niños estaban en el aula de clases respectiva con la docente, y eran llamados por el responsable de logística, quien los llevaba con el evaluador respectivo.

Los materiales utilizados en la validación fueron: un prototipo físico de la interfaz gráfica (ver Figura 63) y una guía de validación (ver Apéndice 3). El evaluador usa la guía para tomar notas y seguir el proceso, allí se registra las respuestas y las reacciones de los niños. La interfaz física consistió en:

- Una hoja con los íconos diseñados para programar (comandos).

- Tres hojas con las versiones de las opciones para realizar la secuencia de comandos.
- Tres hojas con los patrones de interacción.

La Figura 64 muestra los patrones de interacción y opciones de secuencia evaluados. La Figura 64a muestra la interacción de inserción, donde el usuario presiona el lugar en el que quiere insertar un comando y aparecen las opciones. Figura 64b muestra la interacción de arrastrar y soltar, donde los comandos se seleccionan y se arrastran al lugar en el que el usuario desea colocarlo. Finalmente, la Figura 64c muestra las opciones de secuencia: impresiones de huellas de mono, línea de puntos, y sin guía.

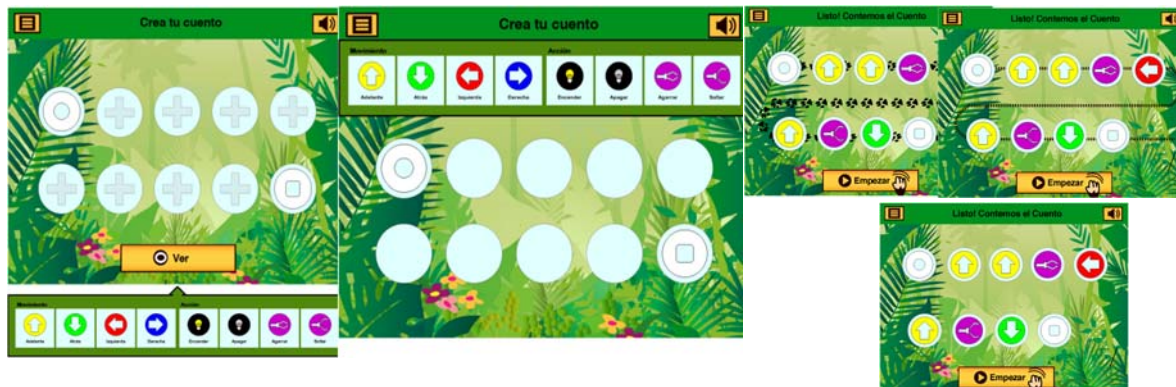


Figura 63. Espacio físico, materiales y proceso de la validación.

Procedimiento

La actividad duró cuatro horas, durando en promedio 24 minutos por niño, tres niños al mismo tiempo (uno por evaluador). El evaluador primero se presentó y le preguntó al niño su nombre y edad. Luego, los instrumentos se presentaron y se explicó el proceso. Se

realizaron tres validaciones diferentes: la iconografía (10 íconos), los diseños de interfaz del área de programación (opciones de secuencias), y los modos de interacción.



a) Patrón de interacción:
Inserción

b) Patrón de interacción:
Arrastrar y soltar

c) Opciones de secuencia:
huellas, líneas y sin guía

Figura 64. Patrones de interacción y opciones de secuencia.

Con la ayuda de un diseñador gráfico y los expertos de la FOD se crearon 10 íconos que permiten ejecutar las acciones descritas en los requisitos. El investigador le preguntó a cada niño el significado de cada ícono e hizo anotaciones si el niño lo hizo bien o mal, y cualquier otra interpretación dicha por el niño.

La segunda validación fue sobre los diseños de interfaz del área de programación (opciones de secuencias). Tres enfoques diferentes se diseñaron con el fin de determinar la forma de guiar a los niños en la interfaz: huellas de mono, línea de puntos y una interfaz sin directrices (véase la Figura 64c). Las tres opciones fueron presentadas a cada niño y se les pidió que explicaran el orden de la secuencia de un conjunto de íconos. Cada evaluador presentó las tres posibilidades en orden aleatorio.

La tercera validación se centró en la interacción. Una vez más, hay tres opciones disponibles: inserción (Figura 64a), arrastrar y soltar (Figura 64b), y un modo guiado. La interacción de inserción necesitaba que el niño señalara el lugar seleccionado, entonces aparecía el conjunto de comandos, y el niño debía seleccionar el comando a insertar. La opción de arrastrar y soltar permitió que los niños señalaran un comando, lo arrastraran hasta el lugar deseado, y

lo soltaran. Las interacciones guiadas permiten dos opciones cada vez. Esta actividad de validación se realiza solo una vez por cada niño (es decir, se realizaron 13 validaciones por modo de interacción).

El procedimiento llevado a cabo, de forma general, fue el siguiente:

1. Presentación del investigador y preguntas al niño de su edad.
2. Presentación y explicación de los instrumentos que se usaron en el proceso.
3. Realización de la primera validación.
4. Realización de la segunda validación.
5. Realización de la tercera validación.
6. Despedida y agradecimiento.

Análisis de Datos

El análisis de los datos se realizó de forma cualitativa, mediante observaciones, comentarios y valoraciones de evaluadores. Además, se realizaron gráficos y tablas de frecuencias (estadística descriptiva).

APÉNDICE 14. PROCESO DE PRUEBAS CON USUARIOS FINALES – TITIBOTS

El objetivo de esta etapa fue evaluar la usabilidad y la funcionalidad de la herramienta de programación TITIBOTS con niños de preescolar. Para ello, se crearon una serie de desafíos para los niños y se realizaron observaciones para evaluar su comportamiento. El objetivo principal de esta evaluación fue para observar la reacción de los niños con la herramienta y, encontrar problemas de usabilidad y funcionalidad.

Participantes

Esta evaluación se realizó en las instalaciones de la FOD con dos grupos, cuatro niños en cada uno, durante 1 día, se realizaron dos sesiones de una hora y veinte minutos cada una. En la primera sesión se llevó a cabo una mediación técnica y en la segunda se llevó a cabo una mediación lúdica. Se siguió la recomendación de Nielsen de realizar el test de usabilidad con 3 a 5 participantes (Jakob Nielsen, 1993).

Por un problema de último minuto, un niño no participó en la evaluación y se trabajó con 7 niños entre 4 y 5 años. La selección de los participantes se realizó mediante la solicitud de participación de niños de preescolar en el taller (la FOD se encargó de este proceso).

La evaluación fue realizada con un total de 12 personas, entre ellas: ocho observadores, una mediadora, tres expertos de soporte técnico, y un camarógrafo.

Espacio Físico y Materiales

La evaluación se llevó a cabo en un laboratorio de robótica de la FOD. Cuatro espacios de trabajo fueron marcados con cinta adhesiva en el piso, cada uno separado por 2 metros (ver Figura 65). El espacio central se utilizó para que los observadores se movieran sin interferir con el trabajo de los niños y para colocar las cámaras. Cada niño tenía una tableta con la

aplicación y un robot. Se definieron guías didácticas de trabajo, guías de observación y guías de evaluación para cada actividad.



Figura 65. Niños trabajando en el respectivo espacio de trabajo.

Procedimiento

El proceso de evaluación se realizó en dos sesiones de una hora y veinte minutos cada una. En la primera sesión se llevó a cabo una mediación técnica, donde la mediadora explicó las funciones generales de TITIBOTS, la interacción y el significado de cada comando. En la segunda sesión se llevó a cabo una mediación lúdica, donde la mediadora jugó con los niños, y en cada juego introdujo los comandos de TITIBOTS.

Antes de cada sesión la mediadora hizo una actividad de bienvenida, mientras que otra persona realizaba una entrevista semiestructurada a los padres de familia sobre el perfil tecnológico de los niños (ver Apéndice 5).

Luego, la mediadora les explicó a los niños tres desafíos que debían resolver usando la herramienta de programación. Cada niño tenía una cámara que grababa lo que hacía en su espacio de trabajo durante toda la sesión, y un observador que miró cómo resolvía los

desafíos y tomó notas en la guía dada (ver Apéndice 4). Cada observador elaboró un reporte y con los videos se extrajeron más datos. Los expertos de soporte técnico estuvieron presentes para arreglar los robots, si era necesario.

La evaluación fue diseñada para evaluar si los niños eran capaces de lograr cada desafío en un periodo de 30 minutos. Estos desafíos fueron:

1. Mover el robot de un punto de inicio a un punto final, en línea recta y pasando por un túnel (el niño debe encender la luz en el interior del túnel).
2. Mover el robot de un punto de inicio a un punto final, en línea recta, agarrar una bola en el punto final y volver al punto de inicio.
3. Mover el robot de un punto de inicio a un punto final, donde debe ir hasta la bola en línea recta, agarrar la bola, girar a la izquierda, mover en línea recta hasta el punto final y soltar la bola.

Al final de la sesión la mediadora realizó la prueba de memoria (ver Apéndice 8) y el cuestionario de satisfacción (ver Apéndice 6) sobre la herramienta de programación y el robot.

Análisis de Datos

El análisis de los datos se realizó de forma cualitativa y cuantitativa. Cualitativamente mediante observaciones, comentarios, gráficos y tablas de frecuencias (estadística descriptiva). Cuantitativamente mediante métricas de usabilidad.

APÉNDICE 15. DESARROLLO Y USO EN UN ENTORNO REAL – TITIBOTS

Una vez que se probó el funcionamiento correcto y el diseño de TITIBOTS, se organizó con la FOD un taller de robótica para niños entre 4 y 5 años de edad, con el objetivo de usar la herramienta de programación TITIBOTS para introducir conceptos básicos de programación y robótica, y observar si los niños pueden aprender a programar robots mediante la utilización de dicha herramienta.

Participantes

Esta evaluación se realizó en las instalaciones de la FOD con un grupo de seis niños durante 4 días, con un total de 8 horas (2 horas por día). Se siguió la recomendación de Nielsen de realizar el test de usabilidad con 3 a 5 participantes (Jakob Nielsen, 1993).

Originalmente, eran tres niñas y tres niños, pero se presentó un problema de último minuto y un niño fue sustituido por una niña. El rango de edad de las niñas fue de 4 años y 3 meses a 5 años y 6 meses. El rango de edad de los niños fue de 4 años y 4 meses a 4 años y 10 meses. La selección de los participantes se realizó mediante la solicitud de participación de niños de preescolar en el taller (la FOD se encargó de este proceso).

La evaluación fue realizada por un total de 7 personas, entre ellas: tres observadores, una mediadora, dos expertos de soporte técnico, y un camarógrafo.

Espacio Físico y Materiales

La evaluación se llevó a cabo en un laboratorio de robótica de la FOD durante 4 días. La Figura 66 muestra algunas fotos del taller en la realización de diferentes actividades y usando los materiales creados.

Entre los materiales que se usaron en el taller están: juegos, túneles, focos, bolas, láminas de secuencias, pizarras magnéticas, comandos impresos y cartones de las rutas con los desafíos señalados. Además, cada niño tiene una tableta con la aplicación y un robot. Los observadores se movían sin interferir con el trabajo de los niños y las cámaras se colocan en posiciones estratégicas. Se definieron guías didácticas de trabajo, guías de observación y guías de evaluación para cada actividad.



a) Actividad lúdica para comandos de movimiento



b) Actividad lúdica para comandos de las luces



c) Actividad lúdica para comandos de la garra



d) Mediadora explicando secuencias a los niños



e) Mediadora explica el proceso de solución de problemas



f) Área de trabajo durante la solución de los desafíos



g) Los niños planean la solución del problema en una pizarra



h) Los niños implementan la solución usando TITIBOTS



i) Los niños observan al robot ejecutando el programa creado

Figura 66. Actividades realizadas en el taller de TITIBOTS.

Procedimiento

En el taller se utilizó una mediación lúdica, donde la mediadora jugó con los niños, y en cada juego se describió los comandos de TITIBOTS, los cuales se introdujeron en parejas y se

crearon niveles: una pareja por nivel (1^{er} nivel: adelante y atrás, 2^{do} nivel: izquierda y derecha, 3^{er} nivel: encender y apagar las luces, 4^{to} nivel: agarrar y soltar).

Se contó con dos cámaras para grabar las acciones y las expresiones de los niños durante los cuatro días del taller, y tres observadores que miraron cómo resolvían los desafíos los niños. Estos observadores tomaron notas en la guía dada (ver Apéndice 4). Cada observador elaboró un reporte y con los videos se extrajeron más datos. Los expertos de soporte técnico estuvieron presentes para arreglar los robots, en caso necesario.

El primer día, la mediadora hizo una actividad de bienvenida, mientras que otra persona realizaba una entrevista semiestructurada a los padres de familia sobre el perfil tecnológico de los niños (ver Apéndice 5). En ese día se realizaron actividades lúdicas y de programación con la herramienta sobre los niveles 1 y 2.

En el segundo día, se realizaron actividades lúdicas y de programación con la herramienta sobre los niveles 3 y 4. En el tercer día, se introdujeron conceptos básicos de programación y robótica, donde se dieron tres de los pasos básicos del proceso de solución de problemas: planeación, implementación y ejecución (prueba).

El cuarto y último día del taller, la mediadora les explicó a los niños tres desafíos que debían resolver usando la herramienta de programación. La evaluación fue diseñada para evaluar si los niños fueron capaces de lograr cada desafío en un periodo de 30 minutos. Los desafíos fueron los siguientes:

1. Mover el robot de un punto de inicio a un punto final, en línea recta y pasando por un túnel (el niño debe encender la luz en el interior del túnel).
2. Mover el robot de un punto de inicio a un punto final, en línea recta, agarrar una bola en el punto final y volver al punto de inicio.
3. Mover el robot de un punto de inicio a un punto final, donde debe ir hasta la bola en línea recta, agarrar la bola, girar a la izquierda, mover en línea recta hasta el punto final y soltar la bola.

Al final del cuarto día, la mediadora realizó la prueba de memoria (ver Apéndice 8) y el cuestionario de satisfacción (ver Apéndice 6) sobre la herramienta de programación y el robot.

Análisis de Datos

El análisis de los datos se realizó de forma cualitativa y cuantitativa. Cualitativamente mediante observaciones, comentarios, gráficos y tablas de frecuencias (estadística descriptiva). Cuantitativamente mediante métricas de usabilidad.

APÉNDICE 16. PROCESO DE PRUEBAS CON USUARIOS FINALES – TITIBOTS COLAB

En esta etapa se evaluó la usabilidad y la funcionalidad de la herramienta colaborativa de programación TITIBOTS Colab. Esta evaluación se llevó a cabo en un taller con 16 niños entre 5 y 6 años. Para ello, se crearon una serie de actividades lúdicas y desafíos para los niños, y se realizaron observaciones para evaluar su comportamiento. El objetivo principal de esta evaluación fue encontrar problemas de usabilidad y funcionalidad. Además, se quería observar el comportamiento de los niños al trabajar de forma colaborativa en las actividades y los desafíos, para ver si con el uso de la herramienta incentiva la colaboración.

Participantes

Esta evaluación se realizó en la Escuela de Educación Básica Municipal Otávio Manoel Anastácio, en Araranguá – Santa Catarina – Brasil, con 16 niños entre 5 y 6 años del grupo de preescolar de la Institución, durante 10 días (dos días por semana, una hora y treinta minutos cada día). La evaluación fue realizada por un total de 10 personas, entre ellas: cuatro observadores, tres mediadoras, dos expertos de soporte técnico, y un camarógrafo.

Espacio Físico y Materiales

La evaluación se llevó a cabo en el salón de clases de preescolar de la Escuela de Educación Básica Municipal Otávio Manoel Anastácio. El salón se acomodó con cinco mesas (una por grupo colaborativo, donde se sentaban tres niños) en forma de media luna y una mesa en el centro donde se colocaba el robot. Los observadores se movían sin interferir con el trabajo de los niños.

Entre los materiales utilizados están: cartel con las reglas del trabajo colaborativo, juegos, bolas, láminas de secuencias, fieltros, comandos impresos y cartel con tablas de éxitos para ver el progreso de los grupos en los desafíos (ver Figura 67). Además, cada niño tiene una tableta con la aplicación, y un robot compartido por todos los grupos colaborativos. El robot

era un brazo robótico disfrazado de serpiente. Se definieron guías didácticas de trabajo, guías de observación y guías de evaluación para cada actividad.



a) La docente muestra las reglas del trabajo colaborativo



b) Los grupos juegan de forma colaborativa



c) Un grupo ve su progreso en los desafíos en las tablas de éxito



d) Los grupos planean la solución del problema en una pizarra



e) Los grupos implementan la solución usando TITIBOTS Colab



f) Los grupos observan al robot ejecutando el programa creado

Figura 67. Actividades realizadas en el taller de TITIBOTS Colab en Brasil.

Procedimiento

En el proceso de evaluación se realizó un taller conformado por diez sesiones con 16 niños de preescolar de la Institución. Como parte del proceso fue necesario realizar antes del taller, una capacitación dirigida a las docentes de la Escuela y los investigadores del RExLab sobre los contenidos a desarrollar durante el taller, así como la utilización de la herramienta colaborativa de programación. En todo el proceso los investigadores tomaron notas, fotos y videos.

En relación con la capacitación, esta se realizó en las instalaciones de la Universidade Federal de Santa Catarina (UFSC) con un tiempo presencial de 20 horas. Para la capacitación se diseñó un plan de trabajo y la elaboración de un curso con contenidos sobre conceptos lógicos-matemáticos, colaboración, programación y robótica. Además, se diseñó estrategias didácticas, material didáctico orientado a docentes y estudiantes, y actividades lúdicas. Todos los recursos diseñados fueron revisados y avalados por las docentes. Las docentes tenían a su cargo los niveles de preescolar y primer grado; una docente de preescolar, con un grupo

de 21 niños con edades entre 5 y 6 años, y dos docentes de primer grado, con grupos de 20 alumnos (aproximadamente) con edades entre 6 y 7 años.

Los grupos colaborativos se formaron a criterio de la docente de preescolar, un total de 5 grupos. Los grupos no siempre estuvieron conformados por los mismos niños durante todo el taller, fueron grupos informales (D. W. Johnson & Johnson, 1997). En las primeras 5 sesiones los grupos variaron en la conformación de los miembros, pero en las últimas 5 la investigadora le solicitó a la docente no variar los miembros de los grupos y dejarlos fijos, por motivo de los desafíos que se realizarían en las dos últimas sesiones.

El procedimiento llevado a cabo, de forma general, fue el siguiente:

- Diseño del taller con su respectivo plan de trabajo y material didáctico.
- Implementación de la herramienta colaborativa de programación TITIBOTS Colab (aplicación móvil) y el robot Cora (brazo robótico).
- Realización de la capacitación para las docentes de la Escuela y los investigadores del RExLab.
- Realización de una reunión con los padres de familia para explicarles sobre el taller y solicitarles la autorización correspondiente para la asistencia de los niños (poner tomar fotos y videos).
- Realización del taller dirigido a niños de preescolar de la Escuela.
- Evaluación de las actividades realizadas en el taller.

En la capacitación, se mostró el plan de trabajo y el taller diseñados por los investigadores. Ambos fueron comentados con las docentes y se realizaron los cambios que fueron sugeridos por ellas. Además, se les enseñó a usar la herramienta colaborativa de programación, tanto el módulo de la docente como el módulo de los estudiantes. En este aspecto, se les enseñó conceptos básicos de programación y robótica a las docentes.

En relación al taller desarrollado con los niños, se realizó durante dos mañanas por semana (lunes y viernes) como una actividad extra del tiempo lectivo. En el taller se trabajó mediante

la metodología del aprendizaje colaborativo, donde las docentes fueron mediadoras del proceso de aprendizaje: guiaban el proceso de aprendizaje de diferentes contenidos mediante las actividades (juegos, ejercicios y desafíos), y realizaban preguntas que orientaran a los niños a la solución o a aprender de los errores cometidos. En cada uno de los contenidos desarrollados en el taller se realizó el siguiente procedimiento:

- Bienvenida y presentación.
- Introducción al contenido.
- Motivación, mediante imágenes representativas, para que los niños muestren sus conocimientos previos acerca del contenido.
- Explicación breve del contenido.
- Realización de juegos con los niños para concretar y practicar conocimientos del contenido.
- Realización de diferentes actividades (ejercicios y desafíos) para fomentar el aprendizaje de los conocimientos del contenido (ver Figura 67).

En las últimas dos sesiones se realizaron desafíos para realizar la evaluación de los conocimientos adquiridos, así como la usabilidad y funcionalidad de la herramienta colaborativa de programación. En esta parte, se contabilizó la cantidad de desafíos realizados y el tiempo promedio de solución por cada grupo colaborativo. En total se realizaron 6 desafíos, divididos en tres desafíos por día. En los primeros tres desafíos fueron realizados por cuatro equipos (faltaron tres estudiantes ese día) y los tres últimos desafíos fueron realizados por cinco equipos.

La evaluación fue diseñada para evaluar si los niños eran capaces de lograr cada desafío en un periodo de 30 minutos y con un máximo de tres intentos (ver Apéndice 12). Los desafíos fueron los siguientes:

1. Agarrar la pieza anaranjada y luego soltarla.
2. Agarrar la pieza triangular y luego soltarla seis espacios a la derecha con respecto a la posición de la pieza.

3. Colocar la cantidad de piezas en el recipiente que tiene la representación del número de la cantidad (hay tres piezas y el recipiente tiene el número 1).
4. Agarrar la pieza circular y luego soltarla.
5. Agarrar la pieza amarilla y luego soltarla tres espacios a la izquierda con respecto a la posición de la pieza.
6. Colocar la cantidad de piezas en el recipiente que tiene la representación del número de la cantidad (hay tres piezas y el recipiente tiene el número 2).

En relación con la usabilidad se usaron las métricas definidas por Nielsen (Jakob Nielsen, 1993). La evaluación de funcionalidad está contemplada en la evaluación de usabilidad, ya que se ejecutaron las tareas especificadas en los requisitos de la aplicación móvil (casos de uso), garantizando que hace lo que debe (casos positivos), que no falla (casos negativos) y que cumple con lo que se ha solicitado (satisface los requisitos). Si el resultado era distinto al esperado, el defecto fue reportado con los detalles: descripción, datos utilizados, capturas de pantalla, entre otros, para facilitar la solución del defecto.

En la reunión de padres de familia, se pasó una entrevista semiestructurada sobre el perfil tecnológico de los niños (ver Apéndice 7). Una semana después de la última sesión del taller, se llevó a cabo la prueba de memoria (ver Apéndice 8) y el cuestionario de satisfacción (ver Apéndice 9). Las docentes se hicieron cargo de pasar este cuestionario a los niños.

Análisis de Datos

El análisis de los datos se realizó de forma cualitativa y cuantitativa. Cualitativamente mediante observaciones, comentarios, gráficos y tablas de frecuencias (estadística descriptiva). Cuantitativamente mediante métricas de usabilidad.

APÉNDICE 17. DESARROLLO Y USO EN UN ENTORNO REAL – TITIBOTS COLAB

En esta etapa se evaluó la herramienta colaborativa de programación TITIBOTS Colab en un entorno real. Esta evaluación se llevó a cabo en un taller con 14 niños entre 4 y 5 años. Para ello, se crearon una serie de actividades lúdicas y desafíos para los niños, y se realizaron observaciones para evaluar su comportamiento. El objetivo principal de esta evaluación fue valorar el aporte de la herramienta en el proceso de colaboración en niños de preescolar, se observó el comportamiento de los niños al trabajar de forma colaborativa en las actividades y los desafíos, para ver si con el uso de la herramienta incentiva la colaboración.

Participantes

Esta evaluación se realizó en la Escuela Santa Mónica, en Montelimar, Guadalupe, Costa Rica, con 14 niños entre 4 y 5 años del grupo de kínder de la Institución, durante 7 días (una hora y treinta minutos cada día). La evaluación fue realizada por un total de 12 personas, entre ellas: ocho observadores, dos mediadoras, un experto de soporte técnico, y un camarógrafo.

Espacio Físico y Materiales

La evaluación se llevó a cabo en el salón de clases de preescolar de la Escuela Santa Mónica. El salón se acomodó con cinco mesas (una por grupo colaborativo, donde se sentaban tres niños) en forma de media luna y un espacio en el centro donde se colocaba el área de trabajo del robot. Los observadores se movían sin interferir con el trabajo de los niños.

Entre los materiales utilizados están: cartel con las reglas del trabajo colaborativo, juegos, bolas, láminas de secuencias, pizarras magnéticas, comandos impresos, cartones de las rutas con los desafíos señalados y cartel con tablas de éxitos para ver el progreso de los grupos en los desafíos (ver Figura 68). Además, cada niño tiene una tableta con la aplicación, y un robot

compartido por todos los grupos colaborativos. Se definieron guías didácticas de trabajo, guías de observación y guías de evaluación para cada actividad.



a) Los grupos juegan de forma colaborativa



b) Actividades lúdicas para concretizar los comandos



c) Mediadora explicando secuencias a los niños



d) Los grupos planean la solución del problema en una pizarra



e) Los grupos implementan la solución usando TITIBOTS Colab



f) Los grupos observan al robot ejecutando el programa creado

Figura 68. Actividades realizadas en el taller de TITIBOTS Colab en Costa Rica.

Procedimiento

En el proceso de evaluación se realizó un taller conformado por siete sesiones de una hora y treinta minutos cada una, una sesión por día, con 14 niños de kínder de la Institución. Como parte del proceso fue necesario realizar, antes del taller, una capacitación dirigida a las docentes de la Escuela sobre los contenidos a desarrollar durante el taller, así como la utilización de la herramienta colaborativa de programación. En todo el proceso los investigadores tomaron notas, fotos y videos.

La capacitación, se realizó en las instalaciones de la Escuela Santa Mónica con un tiempo presencial de 4 horas. Para la capacitación se diseñó un plan de trabajo y la elaboración de un curso con contenidos sobre conceptos de colaboración, programación y robótica. Además, se diseñó estrategias didácticas, material didáctico orientado a docentes y estudiantes, y actividades lúdicas. Todos los recursos diseñados fueron revisados y avalados por las docentes. Las docentes tenían a su cargo los niveles de preescolar; una docente de

pre-kínder, con un grupo de 10 niños con edades entre 3 y 4 años, y la otra docente de kínder con un grupo de 14 alumnos con edades entre 4 y 5 años.

Los grupos colaborativos se formaron aleatoriamente, un total de 5 grupos. Los grupos siempre estuvieron conformados por los mismos niños durante todo el taller, es decir, eran grupos formales (D. W. Johnson & Johnson, 1997).

El procedimiento llevado a cabo, de forma general, fue el siguiente:

- Diseño del taller con su respectivo plan de trabajo y material didáctico.
- Realización de la capacitación para las docentes de la Escuela.
- Comunicado a los padres de familia por escrito para explicarles sobre el taller y solicitarles la autorización correspondiente para la asistencia de los niños al taller (poner tomar fotos y videos).
- Realización del taller dirigido a niños de kínder de la Escuela.
- Evaluación de las actividades realizadas en el taller.

En la capacitación, se mostró el plan de trabajo y el taller diseñados por la investigadora, ambos fueron comentados con las docentes y se realizaron los cambios que fueron sugeridos por ellas. Además, se les enseñó a usar la herramienta colaborativa de programación, tanto el módulo de la docente como el módulo de los estudiantes. En este aspecto, se les enseñó conceptos básicos de programación y robótica a las docentes.

En relación al taller desarrollado con los niños, se realizó durante siete mañanas durante el tiempo lectivo. En el taller se trabajó mediante la metodología del aprendizaje colaborativo; donde las docentes fueron mediadoras del proceso de aprendizaje: guiaban el proceso de aprendizaje de diferentes contenidos mediante las actividades (juegos, ejercicios y desafíos) y, realizaban preguntas que orientaran a los niños a la solución o a aprender de los errores cometidos. En cada uno de los contenidos desarrollados en el taller se realizó el siguiente procedimiento:

- Bienvenida y presentación.

- Introducción al contenido.
- Motivación, mediante imágenes representativas, para que los niños muestren sus conocimientos previos acerca del contenido.
- Explicación breve del contenido.
- Realización de juegos con los niños para concretar y practicar conocimientos del contenido.
- Realización de diferentes actividades (ejercicios y desafíos) para fomentar el aprendizaje de los conocimientos del contenido (ver Figura 68).

En las últimas dos sesiones se realizaron desafíos para realizar la evaluación de los conocimientos adquiridos, así como la usabilidad y funcionalidad de la herramienta colaborativa de programación. En esta parte, se contabiliza la cantidad de desafíos realizados y el tiempo promedio de solución por cada grupo colaborativo. En total se realizaron 6 desafíos, divididos en tres desafíos por día.

La evaluación fue diseñada para evaluar si los niños eran capaces de lograr cada desafío en un periodo de 30 minutos y con un máximo de tres intentos (ver Apéndice 12). Los desafíos fueron los siguientes:

1. Mover el robot hasta donde esté la bola amarilla y llevarla al espacio correspondiente según el color.
2. Mover el robot hasta donde esté la bola roja y llevarla al espacio correspondiente según el color.
3. Mover el robot hasta donde esté la bola azul y llevarla al espacio correspondiente según el color.
4. Mover el robot de un punto de inicio a un punto final, en línea recta.
5. Mover el robot de un punto de inicio a un punto final, en línea recta, agarrar una bola en el punto final y volver al punto de inicio.
6. Mover el robot de un punto de inicio a un punto final, donde debe ir hasta la bola en línea recta, agarrar la bola, girar a la izquierda, mover en línea recta hasta el punto final y soltar la bola.

En relación con la usabilidad se usaron las métricas de usabilidad definidas por Nielsen (Jakob Nielsen, 1993). La evaluación de funcionalidad está contemplada en la evaluación de usabilidad, ya que se ejecutaron las tareas especificadas en los requisitos de la aplicación móvil (casos de uso), garantizando que hace lo que debe (casos positivos), que no falla (casos negativos) y que cumple con lo que se ha solicitado (satisface los requisitos). Si el resultado era distinto al esperado, el defecto fue reportado con los detalles: descripción, datos utilizados, capturas de pantalla, entre otros, para facilitar la solución del defecto.

A los padres de familia se les envió por medio de una circular la información sobre el taller, la autorización y la entrevista semiestructurada sobre el perfil tecnológico de los niños (ver Apéndice 7). Una semana después de la última sesión del taller, se llevó a cabo la prueba de memoria (ver Apéndice 8) y el cuestionario de satisfacción (ver Apéndice 9), las docentes se hicieron cargo de realizarlos con los niños.

Análisis de Datos

El análisis de los datos se realizó de forma cualitativa y cuantitativa. Cualitativamente mediante observaciones, comentarios, gráficos y tablas de frecuencias (estadística descriptiva). Cuantitativamente mediante métricas de usabilidad.

APÉNDICE 18. DATOS DE LOS DESAFÍOS DEL TALLER CON TITIBOTS

En este apartado se muestran las tablas con los datos que fueron analizados para cada una de las métricas de usabilidad definidas en la evaluación de usabilidad de la investigación. Estos datos fueron obtenidos en el taller realizado en la FOD: “Jugando con TITIBOTS”. En el taller, el cuarto día fue el día donde se realizaron los desafíos de evaluación.

En la Tabla 42 se muestran los datos de la métrica de usabilidad: “Facilidad de aprendizaje”, donde se observa el tiempo promedio para completar el primer desafío y de entrenamiento.

Tabla 42. Métrica de usabilidad “Facilidad de aprendizaje” con TITIBOTS.

Estudiante	Tiempo del 1 ^{er} Desafío	Tiempo de Entrenamiento
Niña 1	15,00min	60,00min
Niña 2	8,75min	35,00min
Niño 1	10,50min	42,00min
Niño 2	10,00min	40,00min
Niña 3	12,00min	48,00min
Niña 4	9,25min	37,00min
Promedio	10,92min	43,67min

En la Tabla 43 se muestra la cantidad de desafíos realizados con éxito por cada niño en cada día del taller, indicando en cada día la cantidad de desafíos propuestos, y la Tabla 44 se muestra el tiempo total que duró cada niño para realizar los desafíos que completó por día, indicando el tiempo total dado por día para completar los desafíos propuestos.

Tabla 43. Métrica de usabilidad “Eficiencia”, porcentaje de desafíos exitosos, con TITIBOTS.

Estudiante	Nº Día del Taller				Total	% Desafíos Exitosos
	1	2	3	4		
Niña 1	4	2	1	0	7	50%
Niña 2	4	4	3	3	14	100%
Niño 1	4	4	3	1	12	86%
Niño 2	4	4	3	3	14	100%
Niña 3	4	4	3	2	13	93%
Niña 4	4	4	3	3	14	100%
Cantidad por día	4	4	3	3	14	88%

Tabla 44. Métrica de usabilidad “Eficiencia”, tiempo promedio para completar un desafío, con TITIBOTS.

Estudiante	N° Día del Taller				Total	Tiempo Promedio
	1	2	3	4		
Niña 1	60,00	60,00	90,00	90,00	300,00	42,86min
Niña 2	35,00	25,00	27,00	29,00	116,00	8,29min
Niño 1	42,00	30,00	60,00	87,00	219,00	18,25min
Niño 2	40,00	27,00	45,00	43,00	155,00	11,07min
Niña 3	48,00	30,00	85,00	70,00	233,00	17,92min
Niña 4	37,00	25,00	33,00	40,00	135,00	9,64min
Tiempo por día	60,00min	60,00min	90,00min	90,00min	300,00min	18,01min

En la Tabla 45 se muestran los datos de la métrica de usabilidad: “Cualidad de ser recordado”, según una serie de preguntas sobre los elementos de la herramienta, las docentes fueron las encargadas de realizar las preguntas a los niños (ver Apéndice 8). Las preguntas realizadas sobre la herramienta de programación TITIBOTS a los niños fueron las siguientes:

1. ¿Cuáles son las acciones que puede hacer el robot Tití? Acá se deben indicar 8 acciones.
2. Enseñar los comandos y preguntar para cada uno: ¿Qué le permite hacer este comando al robot Tití? Acá se muestran 10 comandos.
3. Enseñar diferentes elementos de la herramienta y preguntar para cada uno: ¿Para qué era este botón? o ¿Qué hacía este ícono? Acá se muestran 6 elementos.

Tabla 45. Métrica de usabilidad “Cualidad de ser recordado” con TITIBOTS.

Estudiante	N° Pregunta			% Respuestas Correctas
	1	2	3	
Niña 1	8	10	6	100%
Niña 2	8	10	6	100%
Niño 1	8	10	6	100%
Niño 2	8	10	6	100%
Niña 3	8	10	6	100%
Niña 4	8	10	6	100%
	% Total			100%

En la Tabla 46 y la Tabla 47 se muestran los datos de la métrica de usabilidad: “Efectividad”, de estos datos al dividir el promedio de errores recuperados entre el promedio de errores se obtiene 0,71 ($2,33/3,38 = 0,71$), lo cual da un 69% de errores recuperados del total de errores cometidos por los niños.

Tabla 46. Métrica de usabilidad “Efectividad”, promedio de errores, con TITIBOTS.

Estudiante	N° Día del Taller				Total
	1	2	3	4	
Niña 1	7	10	7	6	30,00
Niña 2	2	0	2	4	8,00
Niño 1	3	1	3	6	13,00
Niño 2	3	0	1	2	6,00
Niña 3	6	2	4	4	16,00
Niña 4	1	0	3	4	8,00
Promedio de errores	3,67	2,17	3,33	4,33	3,38

Tabla 47. Métrica de usabilidad “Efectividad”, promedio de errores recuperados, con TITIBOTS.

Estudiante	N° Día del Taller				Total
	1	2	3	4	
Niña 1	3	2	2	0	7,00
Niña 2	2	0	2	4	8,00
Niño 1	3	1	3	4	11,00
Niño 2	3	0	1	2	6,00
Niña 3	6	2	4	4	16,00
Niña 4	1	0	3	4	8,00
Promedio de errores recuperados	3,00	0,83	2,50	3,00	2,33

En la Tabla 48 se muestran los datos de la métrica de usabilidad: “Satisfacción”, donde se observa las respuestas de los niños a las preguntas del cuestionario (ver Apéndice 6).

Tabla 48. Métrica de usabilidad “Satisfacción” con TITIBOTS.

Estudiante	Pregunta	
	¿Te gustó el juego?	¿Fue fácil usar el juego?
Niña 1	Sí	No
Niña 2	Sí	Sí
Niño 1	Sí	Sí
Niño 2	Sí	Sí
Niña 3	Sí	Sí
Niña 4	Sí	Sí
Porcentaje	100%	83%

Además, se realizaron siete preguntas sobre conceptos básicos de programación y robótica a los niños participantes, las docentes fueron las encargadas de realizar las preguntas a los niños (ver Apéndice 8). Los datos obtenidos se muestran en la Tabla 49. Las preguntas realizadas a los niños fueron las siguientes:

1. ¿Qué es un robot?
2. ¿Cuáles son los pasos para la solución de un problema?
3. ¿En dónde se planifica?
4. ¿En dónde se implementa?
5. ¿En dónde se prueba?
6. ¿Quién le dice al robot Tití qué hacer? ¿Por qué medio se le dice qué hacer?
7. ¿Cuáles son las acciones que puede hacer el robot Tití?

Tabla 49. Preguntas sobre conceptos básicos de programación y robótica con TITIBOTS.

Estudiante	Pregunta							% Respuestas Correctas
	1	2	3	4	5	6	7	
Niña 1	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Niña 2	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Niño 1	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Niño 2	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Niña 3	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Niña 4	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
							% Total	100%

APÉNDICE 19. DATOS DE LOS DESAFÍOS DEL TALLER CON TITIBOTS COLAB EN BRASIL

En este apartado se muestran las tablas con los datos que fueron analizados para cada una de las métricas de usabilidad definidas en la evaluación de usabilidad de la investigación. Estos datos fueron obtenidos en el taller realizado en Brasil: “*Jogando com TITIBOTS Colab*”. En los dos últimos días del taller se realizaron los desafíos de evaluación.

En la Tabla 50 se muestran los datos de la métrica de usabilidad: “Facilidad de aprendizaje”, donde se observa el tiempo promedio para completar el primer desafío y de entrenamiento.

Tabla 50. Métrica de usabilidad “Facilidad de aprendizaje” con TITIBOTS Colab en Brasil.

Grupo	Tiempo del 1^{er} Desafío	Tiempo de Entrenamiento
<i>Pássaro</i>	20,00min	40,00min
<i>Borboleta</i>	17,50min	35,00min
<i>Jacaré</i>	21,00min	42,00min
<i>Golfinho</i>	20,00min	40,00min
<i>Peixe</i>	18,50min	37,00min
Promedio	19,00min	38,80min

En la Tabla 51 se muestra la cantidad de desafíos realizados con éxito por cada grupo en cada día del taller, indicando en cada día la cantidad de desafíos propuestos, y la Tabla 52 se muestra el tiempo total que duró cada grupo para realizar los desafíos que completó por día, indicando el tiempo total dado por día para completar los desafíos propuestos.

Tabla 51. Métrica de usabilidad “Eficiencia”, porcentaje de desafíos exitosos, con TITIBOTS Colab en Brasil.

Grupo	Nº Día del Taller										Total	% Desafíos Exitosos
	1	2	3	4	5	6	7	8	9	10		
<i>Pássaro</i>	2	2	3	3	3	3	3	3	3	3	28	100%
<i>Borboleta</i>	2	2	3	3	3	3	3	3	3	3	28	100%
<i>Jacaré</i>	2	2	3	3	3	3	3	3	3	3	28	100%
<i>Golfinho</i>	2	2	3	3	3	3	3	3	3	3	28	100%
<i>Peixe</i>	2	2	3	3	3	3	3	3	3	3	28	100%
Cantidad por día	2	2	3	3	3	3	3	3	3	3	28	100%

Tabla 52. Métrica de usabilidad “Eficiencia”, tiempo promedio para completar un desafío, con TITIBOTS Colab en Brasil.

Grupo	N° Día del Taller										Total	Tiempo Promedio
	1	2	3	4	5	6	7	8	9	10		
<i>Pássaro</i>	40,00	36,00	77,00	75,00	70,00	71,00	63,00	54,00	39,00	44,00	569,0	20,32min
<i>Borboleta</i>	35,00	25,00	33,00	35,00	30,00	32,00	37,00	35,00	38,00	48,00	348,0	12,43min
<i>Jacaré</i>	38,00	40,00	62,00	66,00	61,00	45,00	54,00	52,00	34,00	40,00	492,0	17,57min
<i>Golfinho</i>	40,00	30,00	45,00	40,00	40,00	41,00	39,00	40,00	41,00	38,00	394,0	14,07min
<i>Peixe</i>	37,00	35,00	74,00	58,00	55,00	50,00	45,00	48,00	0,00	47,00	449,0	16,04min
Tiempo por día	40,00	40,00	90,00	90,00	90,00	90,00	90,00	90,00	90,00	90,00	800,0	16,09min

En la Tabla 53 se muestran los datos de la métrica de usabilidad: “Cualidad de ser recordado”, según una serie de preguntas sobre los elementos de la herramienta, las docentes fueron las encargadas de realizar las preguntas a los niños (ver Apéndice 8). Las preguntas realizadas sobre la herramienta de programación TITIBOTS a los niños fueron las siguientes:

1. ¿Cuáles son las acciones que puede hacer el robot Cora? Acá se deben indicar 6 acciones.
2. Enseñar los comandos y preguntar para cada uno: ¿Qué le permite hacer este comando al robot Cora? Acá se muestran 8 comandos.
3. Enseñar diferentes elementos de la herramienta y preguntar para cada uno: ¿Para qué era este botón? o ¿Qué hacía este ícono? Acá se muestran 8 elementos.

Tabla 53. Métrica de usabilidad “Cualidad de ser recordado” con TITIBOTS Colab en Brasil.

Grupo	N° Pregunta			% Respuestas Correctas
	1	2	3	
<i>Pássaro</i>	6	8	8	100%
<i>Borboleta</i>	6	8	8	100%
<i>Jacaré</i>	6	8	8	100%
<i>Golfinho</i>	6	8	8	100%
<i>Peixe</i>	6	8	8	100%
			% Total	100%

En la Tabla 54 y la Tabla 55 se muestran los datos de la métrica de usabilidad: “Efectividad”, de estos datos al dividir el promedio de errores recuperados entre el promedio de errores se obtiene 0,89 ($2,20/2,46 = 0,89$), lo cual da un 89% de errores recuperados del total de errores cometidos por los niños.

Tabla 54. Métrica de usabilidad “Efectividad”, promedio de errores, con TITIBOTS Colab en Brasil.

Grupo	N° Día del Taller										Total
	1	2	3	4	5	6	7	8	9	10	
<i>Pássaro</i>	5	4	7	6	5	3	3	3	1	1	38,00
<i>Borboleta</i>	2	3	2	0	3	4	3	3	0	1	21,00
<i>Jacaré</i>	3	1	3	2	4	5	2	1	0	0	21,00
<i>Golfinho</i>	3	0	1	1	2	3	2	1	0	0	13,00
<i>Peixe</i>	6	2	6	4	3	3	4	1	0	1	30,00
Promedio de errores	3,80	2,00	3,80	2,60	3,40	3,60	2,80	1,80	0,20	0,60	2,46

Tabla 55. Métrica de usabilidad “Efectividad”, promedio de errores recuperados, con TITIBOTS Colab en Brasil.

Grupo	N° Día del Taller										Total
	1	2	3	4	5	6	7	8	9	10	
<i>Pássaro</i>	5	4	3	3	4	3	3	3	1	1	7,00
<i>Borboleta</i>	2	3	2	0	3	4	3	3	0	1	8,00
<i>Jacaré</i>	3	1	3	2	4	5	2	1	0	0	11,00
<i>Golfinho</i>	3	0	1	1	2	3	2	1	0	0	6,00
<i>Peixe</i>	3	2	4	4	3	3	4	1	0	1	16,00
Promedio de error. recup.	3,20	2,00	2,60	2,00	3,20	3,60	2,80	1,80	0,20	0,60	2,20

En la Tabla 56 se muestran los datos de la métrica de usabilidad: “Satisfacción”, donde se observa las respuestas de los niños a las preguntas del cuestionario (ver Apéndice 6).

Tabla 56. Métrica de usabilidad “Satisfacción” con TITIBOTS Colab en Brasil.

Grupo	Pregunta	
	¿Te gustó el juego?	¿Fue fácil usar el juego?
<i>Pássaro</i>	Sí	Sí
<i>Borboleta</i>	Sí	Sí
<i>Jacaré</i>	Sí	Sí
<i>Golfinho</i>	Sí	Sí
<i>Peixe</i>	Sí	Sí
Promedio	100%	100%

Las preguntas realizadas sobre la herramienta colaborativa de programación y su satisfacción, para las métricas “Calidad de ser recordado” y “Satisfacción” respectivamente, fueron contestadas por todos los niños participantes, pero los datos fueron agrupados por los grupos colaborativos.

Además, se realizaron siete preguntas sobre conceptos básicos de programación y robótica a los niños participantes, las docentes fueron las encargadas de realizar las preguntas a los niños (ver Apéndice 8). Los datos obtenidos se muestran en la Tabla 57. Las preguntas realizadas a los niños fueron las siguientes:

1. ¿Cuál fue la forma en que se trabajó?
2. ¿Qué es un robot?
3. ¿Cuáles son los pasos para la solución de un problema?
4. ¿En dónde se planifica?
5. ¿En dónde se implementa?
6. ¿En dónde se prueba?
7. ¿Quién le dice al robot Tití qué hacer? ¿Por qué medio se le dice qué hacer?
8. ¿Cuáles son las acciones que puede hacer el robot Tití?

Tabla 57. Preguntas sobre conceptos básicos de programación y robótica con TITIBOTS Colab en Brasil.

Grupo	Pregunta								% Resp. Correctas
	1	2	3	4	5	6	7	8	
<i>Pássaro</i>	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
<i>Borboleta</i>	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
<i>Jacaré</i>	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
<i>Golfinho</i>	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
<i>Peixe</i>	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
								% Total	100%

APÉNDICE 20. DATOS DE LOS DESAFÍOS DEL TALLER CON TITIBOTS COLAB EN COSTA RICA

En este apartado se muestran las tablas con los datos que fueron analizados para cada una de las métricas de usabilidad definidas en la evaluación de usabilidad de la investigación. Estos datos fueron obtenidos en el taller realizado en Costa Rica: “Jugando con TITIBOTS Colab”. En los dos últimos días del taller se realizaron los desafíos de evaluación.

En la Tabla 58 se muestran los datos de la métrica de usabilidad: “Facilidad de aprendizaje”, donde se observa el tiempo promedio para completar el primer desafío y de entrenamiento.

Tabla 58. Métrica de usabilidad “Facilidad de aprendizaje” con TITIBOTS Colab en Costa Rica.

Grupo	Tiempo del 1^{er} Desafío	Tiempo de Entrenamiento
Mono	18,50min	37,00min
Tortuga	15,50min	31,00min
Delfin	20,00min	40,00min
Pájaro	15,00min	30,00min
Cocodrilo	18,00min	36,00min
Promedio	17,40min	34,80min

En la Tabla 59 se muestra la cantidad de desafíos realizados con éxito por cada grupo en cada día del taller, indicando en cada día la cantidad de desafíos propuestos, y la Tabla 60 se muestra el tiempo total que duró cada grupo para realizar los desafíos que completó por día, indicando el tiempo total dado por día para completar los desafíos propuestos.

Tabla 59. Métrica de usabilidad “Eficiencia”, porcentaje de desafíos exitosos, con TITIBOTS Colab en Costa Rica.

Grupo	Nº Día del Taller							Total	% Desafíos Exitosos
	1	2	3	4	5	6	7		
Mono	2	3	3	3	3	3	3	20	100%
Tortuga	2	3	3	3	3	3	3	20	100%
Delfin	2	3	3	3	3	3	3	20	100%
Pájaro	2	3	3	3	3	3	3	20	100%
Cocodrilo	2	3	3	3	3	3	3	20	100%
Cantidad por día	2	3	3	3	3	3	3	20	100%

Tabla 60. Métrica de usabilidad “Eficiencia”, tiempo promedio para completar un desafío, con TITIBOTS Colab en Costa Rica.

Grupo	N° Día del Taller							Total	Tiempo Promedio
	1	2	3	4	5	6	7		
Mono	37,00	38,00	41,00	36,00	37,00	21,00	22,00	232,00min	11,60min
Tortuga	31,00	27,00	34,00	35,00	30,00	22,00	17,00	196,00min	9,80min
Delfín	40,00	42,00	57,00	55,00	39,00	35,00	30,00	298,00min	14,90min
Pájaro	30,00	25,00	33,00	32,00	35,00	19,00	16,00	190,00min	9,50min
Cocodrilo	36,00	35,00	56,00	53,00	41,00	45,00	23,00	289,00min	14,45min
Tiempo por día	40,00	60,00	90,00	90,00	90,00	90,00	90,00	550,00min	12,05min

En la Tabla 61 se muestran los datos de la métrica de usabilidad: “Cualidad de ser recordado”, según una serie de preguntas sobre los elementos de la herramienta, las docentes fueron las encargadas de realizar las preguntas a los niños (ver Apéndice 8). Las preguntas realizadas sobre la herramienta de programación TITIBOTS a los niños fueron las siguientes:

1. ¿Cuáles son las acciones que puede hacer el robot Tití? Acá se deben indicar 6 acciones.
2. Enseñar los comandos y preguntar para cada uno: ¿Qué le permite hacer este comando al robot Tití? Acá se muestran 8 comandos.
3. Enseñar diferentes elementos de la herramienta y preguntar para cada uno: ¿Para qué era este botón? o ¿Qué hacía este ícono? Acá se muestran 8 elementos.

Tabla 61. Métrica de usabilidad “Cualidad de ser recordado” con TITIBOTS Colab en Costa Rica.

Grupo	N° Pregunta			% Respuestas Correctas
	1	2	3	
Mono	6	8	8	100%
Tortuga	6	8	8	100%
Delfín	6	8	8	100%
Pájaro	6	8	8	100%
Cocodrilo	6	8	8	100%
			% Total	100%

En la Tabla 62 y la Tabla 63 se muestran los datos de la métrica de usabilidad: “Efectividad”, de estos datos al dividir el promedio de errores recuperados entre el promedio de errores se obtiene 0,88 ($1,74/1,87 = 0,88$), lo cual da un 88% de errores recuperados del total de errores cometidos por los niños.

Tabla 62. Métrica de usabilidad “Efectividad”, promedio de errores, con TITIBOTS Colab en Costa Rica.

Grupo	N° Día del Taller							Total
	1	2	3	4	5	6	7	
Mono	5	3	3	2	1	1	2	17,00
Tortuga	2	2	1	0	0	1	0	6,00
Delfín	7	4	3	2	2	3	2	23,00
Pájaro	3	1	0	0	0	0	0	4,00
Cocodrilo	4	2	4	2	2	4	1	19,00
Promedio de errores	4,20	2,40	2,20	1,20	1,00	1,80	1,00	1,97

Tabla 63. Métrica de usabilidad “Efectividad”, promedio de errores recuperados, con TITIBOTS Colab en Costa Rica.

Grupo	N° Día del Taller							Total
	1	2	3	4	5	6	7	
Mono	3	3	3	2	1	1	2	7,00
Tortuga	2	2	1	0	0	1	0	8,00
Delfín	3	3	3	2	2	3	2	11,00
Pájaro	3	1	0	0	0	0	0	6,00
Cocodrilo	3	2	4	2	2	4	1	16,00
Promedio de errores recuperados	2,80	2,20	2,20	1,20	1,00	1,80	1,00	1,74

En la Tabla 64 se muestran los datos de la métrica de usabilidad: “Satisfacción”, donde se observa las respuestas de los niños a las preguntas del cuestionario (ver Apéndice 6).

Tabla 64. Métrica de usabilidad “Satisfacción” con TITIBOTS Colab en Costa Rica.

Grupo	Pregunta	
	¿Te gustó el juego?	¿Fue fácil usar el juego?
Mono	Sí	Sí
Tortuga	Sí	Sí
Delfín	Sí	Sí
Pájaro	Sí	Sí
Cocodrilo	Sí	Sí
Promedio	100%	100%

Las preguntas realizadas sobre la herramienta colaborativa de programación y su satisfacción, para las métricas “Cualidad de ser recordado” y “Satisfacción” respectivamente, fueron contestadas por todos los niños participantes, pero los datos fueron agrupados por los grupos colaborativos.

Además, se realizaron siete preguntas sobre conceptos básicos de programación y robótica a los niños participantes, las docentes fueron las encargadas de realizar las preguntas a los niños (ver Apéndice 8). Los datos obtenidos se muestran en la Tabla 65. Las preguntas realizadas a los niños fueron las siguientes:

1. ¿Cuál fue la forma en que se trabajó?
2. ¿Qué es un robot?
3. ¿Cuáles son los pasos para la solución de un problema?
4. ¿En dónde se planifica?
5. ¿En dónde se implementa?
6. ¿En dónde se prueba?
7. ¿Quién le dice al robot Tití qué hacer? ¿Por qué medio se le dice qué hacer?
8. ¿Cuáles son las acciones que puede hacer el robot Tití?

Tabla 65. Preguntas sobre conceptos básicos de programación y robótica con TITIBOTS Colab en Costa Rica.

Grupo	Pregunta								% Resp. Correctas
	1	2	3	4	5	6	7	8	
Mono	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Tortuga	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Delfín	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Pájaro	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
Cocodrilo	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	Correcta	100%
% Total									100%

APÉNDICE 21. DATOS DEL CUASI-EXPERIMENTO

Es este apartado se muestran los datos de los componentes esenciales del aprendizaje colaborativo, para la evaluación del proceso de colaboración de la investigación, que fueron obtenidos en el desarrollo del cuasi-experimento realizado en Costa Rica en la Escuela El Castillo (grupo de control) y la Escuela Santa Mónica (grupo experimental). Así mismo, se realizaron diferentes pruebas estadísticas para analizar los datos, y los resultados obtenidos se presentan también en este apartado.

En las diferentes tablas y figuras se usará la siguiente notación para los componentes esenciales del aprendizaje colaborativo:

1. AC1: Independencia positiva.
2. AC2: Interacción estimuladora (comunicación cara a cara).
3. AC3: Responsabilidad individual y grupal.
4. AC4: Habilidades interpersonales y grupales (trabajo en equipo).
5. AC5: Evaluación grupal (proceso de grupo).

En el análisis estadístico realizado en esta investigación se realizaron pruebas-*t* con muestras independientes (pruebas paramétricas) y pruebas *U* de Mann-Whitney con muestras independientes (pruebas no paramétricas). En ellas se realizó un contraste de medias en las puntuaciones diferenciales. Además, se realizó el análisis del tamaño del efecto.

En la Tabla 66 se muestran los datos de las puntuaciones diferenciales de la muestra de los grupos colaborativos de cada grupo natural, y en la Tabla 67 se muestran los datos las puntuaciones diferenciales de la muestra estudiantes de cada grupo natural.

Tabla 66. Puntuaciones diferenciales de la muestra de los grupos colaborativos.

Grupo	Grupo de Control (EC)					Grupo Experimental (SM)				
	AC1	AC2	AC3	AC4	AC5	AC1	AC2	AC3	AC4	AC5
Mono	15%	0%	6%	14%	0%	67%	75%	75%	81%	87%
Tortuga	6%	4%	0%	10%	20%	67%	75%	50%	62%	80%
Delfin	-19%	8%	-12%	24%	0%	48%	58%	42%	57%	80%
Pájaro	0%	4%	0%	0%	13%	39%	67%	42%	43%	73%
Media μ	1%	4%	-2%	12%	8%	55%	69%	52%	61%	80%

Tabla 67. Puntuaciones diferenciales de la muestra de los estudiantes.

Estudiante	Grupo de Control (EC)					Grupo Experimental (SM)				
	AC1	AC2	AC3	AC4	AC5	AC1	AC2	AC3	AC4	AC5
1	12%	13%	25%	15%	0%	48%	67%	67%	57%	73%
2	18%	0%	0%	5%	-7%	61%	71%	75%	86%	87%
3	9%	0%	6%	15%	0%	67%	75%	67%	71%	87%
4	4%	4%	-17%	0%	13%	73%	71%	50%	67%	53%
5	4%	9%	0%	5%	34%	55%	63%	50%	52%	53%
6	16%	-5%	8%	9%	6%	55%	75%	50%	52%	80%
7	-25%	-4%	13%	24%	0%	39%	54%	42%	48%	33%
8	-20%	8%	23%	14%	0%	39%	29%	25%	34%	73%
9	-10%	4%	8%	14%	0%	54%	69%	50%	71%	80%
10	3%	0%	9%	-5%	20%	36%	33%	42%	43%	60%
11	0%	4%	-8%	0%	20%	39%	67%	42%	43%	73%
12	3%	4%	8%	-5%	13%	18%	17%	33%	33%	47%
Media μ	1%	3%	6%	8%	8%	49%	57%	49%	55%	67%

En la Figura 22 (ver Sección 5.3.3) se mostró el proceso que se llevó a cabo para realizar cada prueba-*t* con muestras independientes. Este tipo de prueba posee una serie de requisitos y suposiciones que se mencionan a continuación.

Los requisitos de este tipo de prueba son:

- Dos muestras diferentes.
- Misma medida de la variable dependiente (VD).
- Un punto en el tiempo.

Las suposiciones de este tipo de prueba son:

1. Debe haber una variable dependiente (VD) que sea continua.
2. Debe haber una variable independiente (VI) que consta de dos categorías, grupos independientes.

3. Debe haber observaciones independientes, lo que significa que no existe una relación entre las observaciones en cada grupo de la variable independiente o entre los propios grupos, individuos diferentes en cada muestra.
4. No debe haber valores atípicos significativos en los dos grupos de la variable independiente en términos de la variable dependiente.
5. La distribución de la variable dependiente debe ser una distribución aproximadamente normal para cada grupo de la variable independiente.
6. Debe haber homogeneidad de varianzas (es decir, la varianza es igual en cada grupo de la variable independiente).

En la Figura 23 (ver Sección 5.3.3) se mostró el proceso que se llevó a cabo para realizar cada prueba U de Mann-Whitney con muestras independientes. Este tipo de prueba posee una serie de requisitos y suposiciones que se mencionan a continuación.

Los requisitos de este tipo de prueba son:

- Dos muestras diferentes.
- Misma medida de la variable dependiente (VD).
- Un punto en el tiempo.

Las suposiciones de este tipo de prueba son:

1. Debe haber una variable dependiente (VD) que sea continua u ordinal.
2. Debe haber una variable independiente (VI) que consta de dos categorías, grupos independientes.
3. Debe haber observaciones independientes, lo que significa que no existe una relación entre las observaciones en cada grupo de la variable independiente o entre los propios grupos, individuos diferentes en cada muestra.
4. La distribución de los valores de la variable dependiente para cada grupo de la variable independiente puede tener una forma similar o diferente, lo cual se debe determinar.

Los datos de las muestras de ambos grupos naturales, grupo de control (GC) y grupo experimental (GE), cumplen con los requisitos que se establecen en la pruebas-*t* y la prueba *U* de Mann-Whitney. En la Tabla 68, la Tabla 69, la Tabla 70 y la Tabla 71 se muestra la estadística descriptiva de los datos de las muestras de ambos grupos naturales, y en la Tabla 72 y la Tabla 73 la estadística descriptiva de las puntuaciones diferenciales. Además, de la Tabla 74 a la Tabla 77 se comprueba que los grupos naturales tienen una comparabilidad inicial, ya que las pruebas-*t* y pruebas *U* de Mann-Whitney realizadas a los datos del pre-test de ambos grupos muestran que son iguales (no hay una diferencia estadísticamente significativa, valor $p > 0,05$).

Tabla 68. Estadística descriptiva de los datos de los grupos colaborativos del GC.

Variable Dependiente	Pre-test			Pos-test			Tamaño del Efecto de la Diferencia	Fuerza de la Diferencia
	<i>N</i>	Media	Desviación Estándar	<i>N</i>	Media	Desviación Estándar		
AC1	4	40,00	10,2956	4	40,75	6,6521	0,0516	Pequeño
AC2	4	31,25	5,5603	4	35,25	7,4106	1,2247	Grande
AC3	4	33,25	6,9462	4	31,75	13,0480	-0,1987	Pequeño
AC4	4	34,50	5,9722	4	46,50	12,2338	1,1882	Grande
AC5	4	18,50	10,1160	4	25,75	14,1980	0,7290	Moderado

Tabla 69. Estadística descriptiva de los datos de los grupos colaborativos del GE.

Variable Dependiente	Pre-test			Pos-test			Tamaño del Efecto de la Diferencia	Fuerza de la Diferencia
	<i>N</i>	Media	Desviación Estándar	<i>N</i>	Media	Desviación Estándar		
AC1	4	41,75	15,0859	4	97,00	3,4641	3,9306	Grande
AC2	4	24,00	8,2462	4	92,75	8,6168	8,4894	Grande
AC3	4	39,50	14,1539	4	91,50	9,8150	3,2800	Grande
AC4	4	33,25	17,8209	4	94,00	5,8310	3,8661	Grande
AC5	4	20,00	5,7155	4	100,00	0,0000	13,9971	Grande

Tabla 70. Estadística descriptiva de los datos de los estudiantes del GC.

Variable Dependiente	Pre-test			Pos-test			Tamaño del Efecto de la Diferencia	Fuerza de la Diferencia
	<i>N</i>	Media	Desviación Estándar	<i>N</i>	Media	Desviación Estándar		
AC1	12	42,58	9,5647	12	43,74	6,3399	0,1430	Pequeño
AC2	12	35,44	7,6683	12	38,54	7,1883	0,4171	Pequeño
AC3	12	40,25	9,3125	12	46,50	6,6065	0,7741	Pequeño
AC4	12	40,85	8,7231	12	48,45	12,8132	0,6934	Pequeño
AC5	12	25,36	8,6579	12	33,64	13,1168	0,7451	Pequeño

Tabla 71. Estadística descriptiva de los datos de los estudiantes del GE.

Variable Dependiente	Pre-test			Pos-test			Tamaño del Efecto de la Diferencia	Fuerza de la Diferencia
	N	Media	Desviación Estándar	N	Media	Desviación Estándar		
AC1	12	50,03	15,2603	12	98,75	2,3809	4,4611	Grande
AC2	12	39,22	19,9436	12	96,69	5,7800	3,9142	Grande
AC3	12	46,44	13,5448	12	95,81	6,7442	4,6144	Grande
AC4	12	43,19	15,5813	12	98,04	4,2129	4,8058	Grande
AC5	12	33,36	17,0445	12	100,00	0,0000	5,5292	Grande

Tabla 72. Estadística descriptiva de las puntuaciones diferenciales de la muestra de los grupos colaborativos.

Variable Dependiente	Escuela	N	Media	Mediana	Desviación Estándar	Error Estándar de la Media
AC1	El Castillo	4	0,5000	3,0000	14,38749	7,19375
	Santa Mónica	4	55,2500	57,5000	14,05643	7,02822
AC2	El Castillo	4	4,0000	4,0000	3,26599	1,63299
	Santa Mónica	4	68,7500	68,7500	8,09835	4,04918
AC3	El Castillo	4	-1,5000	0,0000	7,54983	3,77492
	Santa Mónica	4	52,2500	46,0000	15,6285	7,81425
AC4	El Castillo	4	12,0000	12,0000	9,93311	4,96655
	Santa Mónica	4	60,7500	59,5000	15,71358	7,85679
AC5	El Castillo	4	8,2500	6,5000	9,94569	4,97284
	Santa Mónica	4	80,0000	80,0000	5,71548	2,85774

Tabla 73. Estadística descriptiva de las puntuaciones diferenciales de la muestra de los estudiantes.

Variable Dependiente	Escuela	N	Media	Mediana	Desviación Estándar	Error Estándar de la Media
AC1	El Castillo	12	1,1667	3,5000	13,3609562	3,8569758
	Santa Mónica	12	48,6667	51,0000	15,2871505	4,4130202
AC2	El Castillo	12	3,0833	4,0000	5,2649499	1,5198601
	Santa Mónica	12	57,5833	67,0000	19,9656903	5,7635983
AC3	El Castillo	12	6,2500	8,0000	11,7405668	3,3892097
	Santa Mónica	12	49,4167	50,0000	14,4754233	4,1786948
AC4	El Castillo	12	7,5833	7,0000	9,0900378	2,6240679
	Santa Mónica	12	54,7500	52,0000	16,2375827	4,6873864
AC5	El Castillo	12	8,2500	3,0000	11,9173289	3,4402365
	Santa Mónica	12	66,5833	73,0000	17,1647542	4,9550377

Tabla 74. Pruebas-*t* para las VD en el pre-test de la muestra de grupos colaborativos.

Variable Dependiente	Prueba- <i>t</i> para igualdad de medias						
	<i>t</i>	<i>gl</i>	<i>p</i>	Media de la Diferencia	Error Estándar de la Diferencia	Intervalo de Confianza de la Diferencia 95%	
						Inferior	Superior
AC1	-0,192	6	0,854	-1,75000	9,13213	-24,09552	20,59552
AC2	1,458	6	0,195	7,25000	4,97284	-4,91811	19,41811
AC3	-0,793	6	0,458	-6,25000	7,88326	-25,53965	13,03965
AC4	0,133	6	0,899	1,25000	9,39747	-21,74479	24,24479
AC5	-0,258	6	0,805	-1,50000	5,80948	-15,71527	12,71527

Tabla 75. Pruebas *U* de Mann-Whitney para las VD en el pre-test de la muestra de grupos colaborativos.

Variable Dependiente	<i>N</i>	<i>U</i> de Mann-Whitney	<i>W</i> de Wilcoxon	<i>Z</i> (Estadística de prueba estandarizada)	Sig. asintótica (bilateral)	Sig. exacta [2*(Sig. unilateral)]
AC1	8	8,500	18,500	0,145	0,885	1,000
AC2	8	3,500	13,500	-1,340	0,180	0,200
AC3	8	10,000	20,000	0,599	0,549	0,686
AC4	8	6,500	16,500	-0,447	0,655	0,686
AC5	8	8,500	18,500	0,150	0,881	1,000

Tabla 76. Pruebas-*t* para las VD en el pre-test de la muestra de estudiantes.

Variable Dependiente	Prueba- <i>t</i> para igualdad de medias						
	<i>t</i>	<i>gl</i>	<i>p</i>	Media de la Diferencia	Error Estándar de la Diferencia	Intervalo de Confianza de la Diferencia 95%	
						Inferior	Superior
AC1	-1,523	22	0,142	-7,25000	4,75956	-17,12072	2,62072
AC2	-1,207	22	0,240	-6,91667	5,73175	-18,80358	4,97025
AC3	-2,039	22	0,054	-7,58333	3,71889	-15,29585	0,12918
AC4	-0,901	22	0,377	-3,75000	4,16113	-12,37966	4,87966
AC5	0,385	22	0,704	2,16667	5,62810	-9,50530	13,83863

Tabla 77. Pruebas *U* de Mann-Whitney para las VD en el pre-test de la muestra de estudiantes.

Variable Dependiente	<i>N</i>	<i>U</i> de Mann-Whitney	<i>W</i> de Wilcoxon	<i>Z</i> (Estadística de prueba estandarizada)	Sig. asintótica (bilateral)	Sig. exacta [2*(Sig. unilateral)]
AC1	24	95,500	173,500	1,370	0,171	0,178
AC2	24	87,000	165,000	0,874	0,382	0,410
AC3	24	102,500	180,500	1,832	0,067	0,078
AC4	24	90,500	168,500	1,082	0,279	0,291
AC5	24	63,500	141,500	-0,498	0,619	0,630

Con la descripción de las pruebas que se llevaron a cabo y la estadística descriptiva de los datos de las muestras de ambos grupos naturales, se comenzó a comprobar las suposiciones de las pruebas. Las tres primeras suposiciones de ambas pruebas estadísticas se cumplieron, se puede observar en los datos. No se encontró valores atípicos en las puntuaciones diferenciales, lo cual se evaluó mediante la inspección de diagramas de caja (Suposición 4, prueba-*t*). Las puntuaciones diferenciales de las variables dependientes se distribuyen normalmente, según la evaluación de la prueba de Shapiro-Wilks ($p > 0,05$) (Suposición 5, prueba-*t*), y hay homogeneidad de varianzas, tal como se evaluó por la prueba de igualdad de varianzas de Levene ($p > 0,05$) (Suposición 6, prueba-*t*). Las distribuciones de las puntuaciones diferenciales de las variables dependientes para cada grupo natural tienen una forma diferente, lo cual se evaluó mediante histogramas (Suposición 4, prueba-*U* de Mann-Whitney) (ver la Tabla 78 y la Tabla 79).

Tabla 78. Comprobación de suposiciones para la muestra de grupos colaborativos.

Variable Dependiente	Escuela	Prueba de Shapiro-Wilk			Prueba de Leven	
		Estadístico	<i>gl</i>	<i>p</i>	<i>F</i>	<i>p</i>
AC1	El Castillo	0,956	4	0,754	0,141	0,721
	Santa Mónica	0,847	4	0,216		
AC2	El Castillo	0,945	4	0,683	3,836	0,098
	Santa Mónica	0,861	4	0,263		
AC3	El Castillo	0,895	4	0,406	1,631	0,249
	Santa Mónica	0,787	4	0,081		
AC4	El Castillo	0,995	4	0,980	0,446	0,529
	Santa Mónica	0,980	4	0,904		
AC5	El Castillo	0,853	4	0,235	3,684	0,103
	Santa Mónica	0,945	4	0,683		

Tabla 79. Comprobación de suposiciones para la muestra de estudiantes.

Variable Dependiente	Escuela	Prueba de Shapiro-Wilk			Prueba de Leven	
		Estadístico	<i>gl</i>	<i>p</i>	<i>F</i>	<i>p</i>
AC1	El Castillo	0,908	12	0,201	0,415	0,526
	Santa Mónica	0,964	12	0,839		
AC2	El Castillo	0,954	12	0,701	3,334	0,081
	Santa Mónica	0,943	12	0,538		
AC3	El Castillo	0,951	12	0,646	0,365	0,552
	Santa Mónica	0,940	12	0,501		
AC4	El Castillo	0,940	12	0,492	3,648	0,069
	Santa Mónica	0,951	12	0,659		
AC5	El Castillo	0,888	12	0,109	2,564	0,124
	Santa Mónica	0,923	12	0,683		

Finalmente, se muestra en la Tabla 80 y la Tabla 81 las pruebas-*t* y, en la Tabla 82 y la Tabla 83 las pruebas *U* de Mann-Whitney para el contraste de medias en las puntuaciones diferenciales de las observaciones de las muestras de ambos grupos naturales.

Tabla 80. Pruebas-*t* para las puntuaciones diferenciales de las VD de la muestra de grupos colaborativos.

Variable Dependiente	Prueba- <i>t</i> para igualdad de medias						
	<i>t</i>	<i>gl</i>	<i>p</i>	Media de la Diferencia	Error Estándar de la Diferencia	Intervalo de Confianza de la Diferencia 95%	
						Inferior	Superior
AC1	5,444	6	0,002	54,75000	10,05713	30,14109	79,35891
AC2	14,830	6	0,000	64,75000	4,36606	54,06663	75,43337
AC3	6,194	6	0,001	53,75000	8,67828	32,51502	74,98498
AC4	5,245	6	0,002	48,75000	9,29494	26,00611	71,49389
AC5	12,510	6	0,000	71,75000	5,73549	57,71576	85,78424

Tabla 81. Pruebas *U* de Mann-Whitney para las puntuaciones diferenciales de las VD de la muestra de grupos colaborativos.

Variable Dependiente	<i>N</i>	<i>U</i> de Mann-Whitney	<i>W</i> de Wilcoxon	<i>Z</i> (Estadística de prueba estandarizada)	Sig. asintótica (bilateral)	Sig. exacta [2*(Sig. unilateral)]
AC1	8	16,000	26,000	2,323	0,020	0,029
AC2	8	16,000	26,000	2,337	0,019	0,029
AC3	8	16,000	26,000	2,337	0,019	0,029
AC4	8	16,000	26,000	2,309	0,021	0,029
AC5	8	16,000	26,000	2,337	0,019	0,029

Tabla 82. Pruebas-*t* para las puntuaciones diferenciales de las VD de la muestra de estudiantes.

Variable Dependiente	Prueba- <i>t</i> para igualdad de medias						
	<i>t</i>	<i>gl</i>	<i>p</i>	Media de la Diferencia	Error Estándar de la Diferencia	Intervalo de Confianza de la Diferencia 95%	
						Inferior	Superior
AC1	8,104	22	0,000	47,50000	5,86097	35,34509	59,65492
AC2	9,143	22	0,000	54,50000	5,96062	42,13842	66,86158
AC3	8,023	22	0,000	43,16667	5,38036	32,00849	54,32484
AC4	8,780	22	0,000	47,16667	5,37190	36,02603	58,30731
AC5	9,670	22	0,000	58,33333	6,03222	45,82328	70,84338

Tabla 83. Pruebas *U* de Mann-Whitney para las puntuaciones diferenciales de las VD de la muestra de estudiantes.

Variable Dependiente	<i>N</i>	<i>U</i> de Mann-Whitney	<i>W</i> de Wilcoxon	<i>Z</i> (Estadística de prueba estandarizada)	Sig. asintótica (bilateral)	Sig. exacta [2*(Sig. unilateral)]
AC1	24	143,500	221,500	4,135	0,000	0,000
AC2	24	144,000	222,000	4,172	0,000	0,000
AC3	24	143,500	221,500	4,147	0,000	0,000
AC4	24	144,000	222,000	4,164	0,000	0,000
AC5	24	143,000	221,000	4,125	0,000	0,000

En la Tabla 34 y la Tabla 35 de la Sección 7.3 se mostró un resumen de las pruebas de hipótesis llevadas a cabo y sus detalles se pueden observar en las tablas anteriores. Como se observa hay una diferencia estadísticamente significativa entre las medias y las distribuciones de ambos grupos naturales (Instituciones) al realizar el análisis de contraste de medias en las puntuaciones diferenciales (muestra de grupos colaborativos y muestra de estudiantes), ya que las probabilidades $p < 0,05$.

APÉNDICE 22. DIBUJO DE UNA NIÑA SOBRE EL TALLER



APÉNDICE 23. OBSERVACIONES FINALES DE LOS EVALUADORES CON RESPECTO AL PROCESO DE COLABORACIÓN

A los evaluadores se les pidió realizar observaciones finales en relación con aspectos importantes del proceso de colaboración e información que pudieran brindar con respecto a la evaluación realizada en el pre-test y el pos-test.

Algunas de las observaciones finales de los evaluadores con respecto a la Escuela Santa Mónica fueron las siguientes:

- “En el pos-test cada niño asumió el rol correspondiente, siempre existe el impulso por querer hacer el trabajo de otros, pero en el pos-test se observó mejor manejo de rol que lo sucedido en el pre-test, se mantuvieron con mayor firmeza en su rol.”
- “Se observa mejor coordinación en el pos-test a la hora de realizar las actividades, ambas chicas comprenden que su compañero tiene más dificultad de seguir sus opiniones, pero lo manejaron mejor en el pos-test.”
- “Dialogan entre ellos; aspecto que se les dificultó más en el pre-test, pero lograron mejor comunicación en el post test.”
- “Observé un mejor manejo del tema de colaboración en el pos-test que no se observó con claridad en el pre-test; los chicos comprendían que cada uno tenía un rol determinado y que debían esperar su tiempo para realizarlo.”
- “Todos dijeron que para realizar las actividades era necesario el trabajo en equipo, y que si alguno faltaba otro podría asumir el rol. Siempre para que el equipo terminara el reto. Me pareció que se logra la intencionalidad del trabajo en equipo.”
- “Mencionaron que el trabajo con TITIBOTS Colab fue divertido, porque hacían mover el robot y que debían tener mucho cuidado para cumplir con las órdenes.”
- “En esta escuela, la principal diferencia fue en el uso de la tableta, se notó claramente que el taller apoyó para que lograran aprender a usar la herramienta, y apoya el proceso colaborativo ya que lo fuerza.”

Algunas de las observaciones finales de los evaluadores con respecto a la Escuela El Castillo fueron las siguientes:

- “Una de las niñas tomó el rol de líder y llevo al equipo, tanto en el pre-test como el pos-test. Sin embargo, tanto en legos como en el balance en el pos-test fue mejor, ya que habían hecho la actividad anteriormente, con la tableta no se vio buenos resultados. De hecho, fueron peores en el pos-test, porque descubrieron como enviar al robot instrucciones y, solo enviaban instrucciones sin sentido.”
- “No trabajaron juntos ni en el pre-test ni en el pos-test. Solo en la actividad de balancear el vaso trabajaron juntos.”
- “En este equipo la dinámica de comunicación era mejor, sobre todo porque eran dos niñas chispas y un niño (el más problemático).”
- “Todos estaban ahí, pero trabajaban como por turnos, sin colaborar sincrónicamente.”
- “No se notó cohesión en el grupo, ni en el pre-test ni el pos-test, cada quien por su lado haciendo cosas diferentes a lo solicitado en las actividades.”
- “No observe mejora entre el pre-test y el post-test para ninguna de las actividades, de hecho, algunas fueron peor.”
- “A diferencia del grupo experimental, este grupo de control no se vio el avance en el uso de la herramienta colaborativa, lo que indica que la mediación o el taller es necesario para agilizar el proceso.”

Además, dos investigadoras de PROINNOVA fueron a observar la realización del taller con el fin de realizar una validación la experiencia de usuario de los niños con la herramienta colaborativa de programación; así como la experiencia general relacionada a los principios básicos de programación y robótica. Posteriormente, enviaron un informe a la investigadora con los resultados de la validación realizada por ellas, entre las conclusiones relacionadas al uso de la herramienta, al aprendizaje de la programación y al aporte en el proceso de colaboración dadas en el informe se encuentran:

- El 100% de los niños se encontraron familiarizados con la tecnología. Recuérdese que la validación se realizó en una escuela privada bilingüe, donde generalmente el

nivel socioeconómico de los padres es medio-alto a alto, y donde la penetración tecnológica es bastante alta.

- En las sesiones había dos docentes: una principal y otra colaboradora. La docente guía principal presentó en los dos primeros días confusión en el rol docente de la aplicación (por ejemplo: en el uso de los distintos botones, ya que en ciertas ocasiones les reiniciaba el juego a algunos equipos). El quinto día de la validación, la docente logró dominar la aplicación rol docente.
- Cuando el equipo logra el objetivo, la docente les enviaba mediante audio un mensaje personalizado, lo cual logró despertar en el 100% de niños y niñas una gran motivación en cumplir el desafío para escuchar lo que “Tití” les decía. Por ende, dicho mensaje personalizado como recompensa funciona bien.
- En cuanto al procedimiento de solución de problemas (proceso de programación): planear, implementar y probar, el 100% de los niños lo lograron a partir del cuarto día del taller.
- El 100% de los niños lograron trabajar de forma colaborativa. Cabe destacar dos situaciones particulares sucedidas en el cuarto y quinto día del taller. En primer lugar, en el equipo pájaro un niño dialogaba con su compañero para convencerlo de porqué él pensaba que tenía razón en el comando seleccionado, y su compañero escuchaba y también daba su opinión acerca de cuál comando era el correcto para él. Este diálogo resulta sorprendente para niños de esta edad. La segunda situación fue el trabajo intergrupar que se suscitó, donde la situación competitiva se dejaba atrás y ciertos niños cuando terminaban su aporte en el equipo, se iban a ayudarle a otros equipos. Lo que se debe resaltar son las fases que el niño atraviesa durante su interacción con TITIBOTS Colab. Las fases observadas fueron: primer contacto con la aplicación y el robot donde se presentaba una etapa de descubrimiento y experimentación; luego cuando dominaban un poco más la aplicación, el sentimiento de colaboración dentro del equipo y competitivo entre equipos empezó a aparecer, y posteriormente, se da el trabajo intergrupar donde se va más allá de una situación competitiva.