



UNIVERSIDAD DE ESPECIALIDADES ESPÍRITU SANTO

Facultad de Ingeniería

Escuela de Ingeniería en Sistemas y Telecomunicaciones

**IMPLEMENTACIÓN DE UN APLICATIVO MÓVIL COMO ASISTENTE PARA
LA CONSTRUCCIÓN DE ROBOTS UTILIZANDO KITS LEGO MINDSTORMS
EV3 Y NXT DIRIGIDO A ESTUDIANTES DE SECUNDARIA**

Trabajo de Titulación que se presenta como requisito para el título de Ingeniero
en Sistemas de Información Gerencial

Autor: Daniel Alfredo Silva Moncayo

Tutor: Ing. Washington Antonio Caraguay Ambuludi, M.Sc.

Samborondón, agosto 2020



APROBACIÓN DEL TUTOR

En mi calidad de Tutor del estudiante Daniel Alfredo Silva Moncayo, que cursa estudios en el programa de TERCER nivel: Ingeniería en Sistemas de Información Gerencial, dictado en la Facultad de Sistemas, Telecomunicaciones y Electrónica de la UEES, en modalidad presencial.

CERTIFICO

Que he revisado el Trabajo de Titulación denominado: "Implementación de un aplicativo móvil como asistente para la construcción de robots utilizando kits LEGO Mindstorms EV3 y NXT dirigido a estudiantes de secundaria", presentado por el estudiante Daniel Alfredo Silva Moncayo, como requisito previo para optar por el Grado Académico de Ingeniero en Sistemas de Información Gerencial CERTIFICO que el Trabajo de Titulación ha sido analizado y reúne todos los requisitos para ser presentado y sometido a los procesos de revisión estipulados por la Facultad.

Atte.

Ing. Washington Antonio Caraguay Ambuludi, M. Sc
0703527465

AGRADECIMIENTOS

Agradezco primero que nada a Dios por permitirme cumplir esta maravillosa meta. A mi hija por motivarme con su amor y sonrisa. A mi esposa por estar siempre a mi lado guiándome. A mi hermana por enseñarme a ser paciente y humanista. A mi papá por estar siempre presente y poder contar en todo momento con su apoyo. Y un agradecimiento en especial para mamá. Mujer valiente, esforzada, trabajadora y sobre todo incondicional. Eres el mejor ejemplo de vida que puedo tener. Gracias a ti me he convertido en el hombre que soy.

Los amo.

ÍNDICE GENERAL

CAPÍTULO 1	INTRODUCCIÓN	2
1.1	Antecedentes	4
1.2	Descripción del Problema	6
1.3	Alcance	8
1.4	Objetivos	9
1.4.1	Objetivo General	9
1.4.2	Objetivos Específicos	9
1.5	Justificación	9
CAPÍTULO 2	MARCO TEÓRICO	12
2.1	Introducción	12
2.2	Robótica Educativa	12
2.3	La Robótica Educativa en el Ecuador	13
2.4	Torneos de Robótica y su impacto en la educación	16
2.5	Tecnología utilizada en el diseño de robots de competencias	19
2.6	Laboratorios y aplicaciones virtuales de robótica de competencia	22
2.6.1	Scrum	25
2.7	Tecnologías de la información	25
2.7.1	Android	25
2.7.1.1	Android Studio	26
2.7.2	MySQL	26
2.7.3	PHP	27
2.7.4	RESTful API	27
CAPÍTULO 3	METODOLOGÍA	29
3.1	Etapa 1: Revisión y selección de robots de competencias	29
3.2	Etapa 2: Desarrollo del asistente	30
3.3	Etapa 3: Ejecución de la experimentación	32
3.4	Etapa 4: Análisis de resultados	33
3.5	Resumen de la metodología	33
CAPÍTULO 4	DESARROLLO DE LA PROPUESTA	35
4.1	Etapa 1: Revisión y selección de robots de competencias	35

4.1.1	Recopilación de información básica	35
4.1.1.1	LEGO Mindstorms NXT	35
4.1.1.2	LEGO Mindstorms EV3	36
4.1.2	Selección de las categorías, registro de componentes y pasos 37	
4.1.2.1	Robot sumo	39
4.1.2.2	Seguidor de línea.....	42
4.1.2.3	Robot insecto.....	45
4.2	Etapa 2: Desarrollo del asistente	48
4.2.1	Planeación de sprints	49
4.2.2	Sprint 1	49
4.2.2.1	Planificación del Sprint 1	50
4.2.2.2	Desarrollo del Sprint 1	51
4.2.2.3	Revisión del Sprint 1	52
4.2.3	Sprint 2.....	53
4.2.3.1	Planificación del Sprint 2	53
4.2.3.2	Desarrollo del Sprint 2	54
4.2.3.3	Revisión del Sprint 2.....	55
4.2.4	Sprint 3.....	56
4.2.4.1	Planificación del Sprint 3	56
4.2.4.2	Desarrollo del Sprint 3	58
4.2.4.3	Revisión del Sprint 3.....	62
4.2.5	Sprint 4.....	64
4.2.5.1	Planificación del Sprint 4	64
4.2.5.2	Desarrollo del Sprint 4	66
4.2.5.3	Revisión del Sprint 4.....	73
4.2.6	Pruebas del desarrollo	75
4.2.6.1	Definición de pruebas	75
4.2.6.2	Resumen de pruebas	80
4.3	Etapa 3: Ejecución de la experimentación.....	81
4.3.1	Instrucciones de uso	83
4.3.2	Uso del asistente.....	84
4.3.3	Ejecución de la evaluación.....	87
4.4	Etapa 4: Análisis de resultados	90

CAPÍTULO 5 CONCLUSIONES Y RECOMENDACIONES.....	92
REFERENCIAS	94

ÍNDICE DE TABLAS

Tabla 2.1 Problemas fundamentales para crear ambientes de aprendizaje con la robótica.....	14
Tabla 2.2 Categoría de robots en torneos	18
Tabla 2.3 Comparación de características y contenidos de un set básico de plataforma LEGO	21
Tabla 3.1 Sprints y tareas	31
Tabla 3.2 Pasos para ejecutar la experimentación	33
Tabla 4.1 Comparación de características y contenidos de un set LEGO Mindstorms EV3 y NXT.....	36
Tabla 4.2 Características del robot sumo implementado con LEGO Mindstorms EV3.....	39
Tabla 4.3 Características del robot sumo implementado con LEGO Mindstorms NXT	41
Tabla 4.4 Características principales de los robots seguidores de línea armados.....	42
Tabla 4.5 Características principales de los robots seguidores de línea armados.....	44
Tabla 4.6 Características principales del robot insecto armado con LEGO Mindstorms EV3.....	45
Tabla 4.7 Características principales del robot insecto armado con LEGO Mindstorms EV3.....	47
Tabla 4.8 Tareas del Sprint 1 y su esfuerzo	50
Tabla 4.9 Planificación de tareas del Sprint 1	50
Tabla 4.10 Verificación de tareas completadas en el Sprint 1	52
Tabla 4.11 Lista de tareas completadas en el Sprint 1	52
Tabla 4.12 Tareas del Sprint 2 y su esfuerzo	53
Tabla 4.13 Planificación de tareas del Sprint 2.....	53
Tabla 4.14 Funciones de la RESTful API.....	54
Tabla 4.15 Verificación de tareas completadas en el Sprint 2	55
Tabla 4.16 Lista de tareas completadas en el Sprint 2	55
Tabla 4.17 Tareas del Sprint 3 y su esfuerzo	57

Tabla 4.18 Planificación de tareas del Sprint 3.....	57
Tabla 4.19 Verificación de tareas completadas en el Sprint 3	63
Tabla 4.20 Lista de tareas completadas en el Sprint 3	63
Tabla 4.21 Tareas del Sprint 4 y su esfuerzo	65
Tabla 4.22 Planificación de tareas del Sprint 4	65
Tabla 4.23 Verificación de tareas completadas en el Sprint 4	73
Tabla 4.24 Lista de tareas completadas en el Sprint 4	74
Tabla 4.25 Características del usuario.....	82
Tabla 4.26 Características del smartphone empleado en la experimentación.....	82
Tabla 4.27 Resultados obtenidos.....	91

ÍNDICE DE FIGURAS

Figura 2.1 Beneficios de la implementación robots en competencias.....	17
Figura 2.2 Plataforma LEGO WeDo.....	20
Figura 2.3 Plataforma LEGO Mindstorms	21
Figura 2.4 Aplicaciones orientadas al armado de robots publicadas en Google Play Store.....	23
Figura 3.1 Diagrama de funcionamiento del asistente	30
Figura 3.2 Resumen del diseño metodológico	34
Figura 4.1 Tareas realizadas para el armado de cada robot	37
Figura 4.2 Simulación de armado empleando LEGO Digital Designer	38
Figura 4.3 Armado de robots empleando los kits EV3 y NXT de LEGO Mindstorms	38
Figura 4.4 Programación de robots EV3 y NXT	39
Figura 4.5 Armado del robot sumo empleando el kit LEGO Mindstorms EV3	40
Figura 4.6 Robot sumo empleando el kit LEGO Mindstorms EV3	40
Figura 4.7 Armado del robot sumo empleando el kit LEGO Mindstorms NXT.....	41
Figura 4.8 Robot sumo empleando el kit LEGO Mindstorms NXT	42
Figura 4.9 Armado del robot seguidor de línea empleando el kit LEGO Mindstorms EV3.....	43
Figura 4.10 Robot seguidor de línea empleando el kit LEGO Mindstorms EV3	43
Figura 4.11 Armado del robot seguidor de línea empleando el kit LEGO Mindstorms NXT	44
Figura 4.12 Robot seguidor de línea empleando el kit LEGO Mindstorms NXT.....	45
Figura 4.13 Armado del robot insecto empleando el kit LEGO Mindstorms EV3	46
Figura 4.14 Robot insecto empleando el kit LEGO Mindstorms EV3.....	46
Figura 4.15 Armado del robot insecto empleando el kit LEGO Mindstorms XT	47

Figura 4.16 Robot insecto empleando el kit LEGO Mindstorms NXT	48
Figura 4.17 Arquitectura del asistente	48
Figura 4.18 Diagrama de secuencia del asistente	49
Figura 4.19 Arquitectura de base de datos del asistente	51
Figura 4.20 Creación de bocetos para el asistente.....	59
Figura 4.21 Flujo de comunicación de actividades del asistente a desarrollar	60
Figura 4.22 Diseño de la actividad splash screen en Android Studio	62
Figura 4.23 Diagrama de casos de uso para la actividad de ingreso.....	67
Figura 4.24 Diseño de la actividad splash screen y onboarding.....	67
Figura 4.25 Diagrama de casos de uso para el dashboard del asistente	68
Figura 4.26 Diseño de la actividad splash screen y onboarding.....	69
Figura 4.27 Opción de ayuda ofrecida por el asistente.....	69
Figura 4.28 Diagrama de casos de uso para la selección versión y tipo de paso	70
Figura 4.29 Diseño de la actividad de selección de versión de LEGO Mindstorms	71
Figura 4.30 Diseño de la actividad de selección de tipo de paso	71
Figura 4.31 Diagrama de casos de uso la actividad de visualización de pasos de armado y programación.....	72
Figura 4.32 Diseño de la actividad de visualización de pasos de armado y programación	73
Figura 4.33 Verificación de funcionamiento empleando Android Virtual Device y un smartphone	76
Figura 4.34 Escenario de prueba de ingreso al asistente	77
Figura 4.35 Escenario de prueba de selección de categoría de robot.....	78
Figura 4.36 Escenario de prueba de selección de versión y tipo de paso	79
Figura 4.37 Escenario de prueba de visualización de instrucciones de armado.....	80
Figura 4.38 Set LEGO Mindstorms Education EV3 empleado para la experimentación.....	83

Figura 4.39 Set LEGO Mindstorms NXT 2.0 empleado para la experimentación.....	83
Figura 4.40 Indicaciones para el uso del asistente y los kits LEGO Mindstorms	84
Figura 4.41 Armado de un robot de sumo empleando al asistente móvil desarrollado y el kit EV3	85
Figura 4.42 Asistencia para la búsqueda de piezas para el armado de un robot de sumo empleando el kit EV3	86
Figura 4.43 Asistencia para la identificación de piezas del kit	87
Figura 4.44 Ejecución de la evaluación empleando el asistente móvil desarrollado	88
Figura 4.45 Ejecución de la evaluación con un estudiante de secundaria armando de un seguidor de línea	89
Figura 4.46 Ejecución de la evaluación mediante el armado de un seguidor de línea	90

RESUMEN

La evolución de la tecnología hace necesario implementar cambios dentro de la estrategia educativa, de manera que las nuevas generaciones se sientan motivadas dentro de las actividades de enseñanza-aprendizaje, pasando de solo emplear materiales bibliográficos y explicaciones teóricas a la inclusión de nuevas tecnologías, como la robótica educativa, siendo un recurso que fomenta el desarrollo de competencias, incrementando el interés de estas nuevas generaciones por la tecnología mediante la resolución de problemas.

Por este motivo, el presente trabajo de titulación describe el desarrollo de un asistente móvil de un laboratorio digital de robótica de competencias estudiantiles que consiste en una aplicación móvil desarrollada en Android Studio para facilitar el armado y programación de robots empleando los kits *LEGO Mindstorms EV3* y *NXT*. Para conseguirlo, se empleó una metodología con enfoque cualitativo y alcance exploratorio. Para el desarrollo del aporte práctico se definieron cuatro etapas conformadas por la revisión y selección de robots de competencias, el desarrollo del asistente bajo la metodología de desarrollo *Scrum*, la ejecución de la experimentación empleando el asistente desarrollado con un estudiante de secundaria y el análisis de resultados obtenidos.

Los resultados en este trabajo de titulación evidencian, que con las apropiadas indicaciones expuestas en el asistente móvil un estudiante sin conocimientos previos de robótica puede armar un robot y generar una curiosidad de poder armar otros modelos distintos. Además, se comprobó que el uso del asistente móvil reduce el tiempo de armado y programación de robots de competencias. Como trabajo futuro, se puede dotar al asistente de otras categorías de robots que puedan armarse con otros kits tecnológicos de otras empresas para otorgar al estudiante de una mayor variedad de robots para explorar.

CAPÍTULO 1 INTRODUCCIÓN

En la actualidad la tecnología ha impulsado el desarrollo en varias áreas del saber humano, logrando inclusive contribuir con herramientas precisas que ayudan a la toma de decisiones cotidianas, tan relevantes como asistir a un médico para ayudarlo a detectar una enfermedad en su paciente, u ofrecerle a una persona recomendaciones de películas desde una aplicación en un dispositivo móvil inteligente.

El impacto de estas nuevas tecnologías ha desembocado en lo que se ha denominado desde algunos años la sociedad de la información y el conocimiento. Es en esta sociedad donde se requiere que sus ciudadanos estén preparados para enfrentarse a un entorno que cambia constantemente, para ello es necesario que cuenten con una apropiada comprensión y que posean competencias, las cuales pueden desarrollarse incorporando tecnologías en sus primeros años de estudio.

En otros países no es extraño el uso de tecnologías dentro del aula como, por ejemplo, García-Ajofrín (2017), describe que en Holanda desde el 2013 se implementó un modelo educativo creado por Steve Jobs en el que se propone que el estudiante establezca sus propias metas, guiado por el maestro y empleando la tecnología como elemento principal y un iPad como herramienta. De esta forma se busca fomentar la colaboración y la independencia del estudiante empleando la tecnología.

Una de estas tecnologías es la robótica enfocada a la educación, o también denominada robótica educativa, que de acuerdo con García & Caballero (2019) actualmente se incorpora como un recurso educativo altamente valioso en el desarrollo de competencias, mejorando la calidad del aprendizaje de una manera lúdica buscando incrementar el interés del estudiante por la tecnología mediante la resolución de problemas.

En el Ecuador la robótica educativa ha logrado un espacio en algunas instituciones educativas. Sin embargo, no todas poseen los recursos para integrarla dentro de sus planes de estudio fomentando una brecha digital tanto de profesores y alumnos, lo que se traduce

directamente, como lo indica López (2018), en una ausencia del desarrollo de competencias para el uso de las tecnologías por parte de los actores del proceso educativo. En este trabajo de titulación se propone el desarrollo de un asistente móvil de un laboratorio digital de robótica de competencias estudiantiles que consiste en la implementación de una aplicación móvil desarrollada en *Android Studio* mediante la cual se puedan consumir los recursos de un servidor web para desplegar los pasos de armado y programación de robots empleando los kits *LEGO Mindstorms*.

La importancia del trabajo consiste en aportar con el desarrollo de una herramienta digital partiendo de una revisión exhaustiva de aportes científicos de otros investigadores para su implementación, con la finalidad de contribuir al estado del arte de la robótica educativa.

Este trabajo de investigación está constituido por cinco capítulos en los que se describe contenido teórico y los procedimientos que fueron necesarios para el desarrollo y pruebas del asistente.

El capítulo 1 contiene los antecedentes, la descripción del problema, el alcance del trabajo, sus objetivos y la importancia por las cuales debe de ser considerado su estudio.

En el capítulo 2 se detalla el marco teórico mediante una revisión bibliográfica de los conceptos de robótica educativa, su aplicación en Ecuador, los torneos de robótica y su impacto de la educación, los laboratorios y aplicaciones virtuales de robótica de competencia, tecnologías utilizadas en el diseño de estos robots y los componentes a emplearse para la implementación del asistente propuesto.

En el capítulo 3 se revisa la metodología empleada. Se define el enfoque de la investigación. También se describe la arquitectura, etapas y actividades necesarias para desarrollar el asistente.

En el capítulo 4 se registran los procedimientos realizados para el desarrollo del asistente móvil, así como también las pruebas de funcionalidad del software, además de la ejecución de la experimentación propuesta y el análisis de los resultados obtenidos basados en el uso y

efectividad del asistente móvil desarrollado para la tarea de armado de robots.

Finalmente, en el capítulo 5 se presentan las conclusiones a partir de los resultados y el aporte del asistente para hacer frente al problema que se plantea abordar en esta investigación. También se exponen recomendaciones para trabajos futuros.

1.1 Antecedentes

El proceso formativo de los jóvenes necesita que se potencie y actualice mediante el uso de tecnologías, de tal manera que esté acorde a la sociedad actual, empleado nuevas herramientas destinadas a mejorar el aprendizaje y que estén al alcance de las instituciones, como teléfonos inteligentes, tablets, aplicaciones, robots entre otros (Luna, 2018).

Kägi, Herzog, & Stähli (2019) mencionan que de acuerdo con informe federal suizo sobre educación digital se establece, por ejemplo, que se debe reducir el analfabetismo digital en la sociedad. Adicionalmente, el Foro Económico Mundial enumera como competencias clave del siglo XXI las competencias de las tecnologías de la información y comunicación, de la misma manera que las competencias transversales tales como el pensamiento crítico, la creatividad, capacidad de resolución de problemas, la comunicación o la cooperación (World Economic Forum, 2017).

En este contexto, tiene cada vez más presencia una corriente que promueve trabajar en de la alfabetización tecnológica en niños y jóvenes, y hacer que el pensamiento computacional se convierta en una competencia relevante en los entornos escolares (Caballero & García-Valcárcel, 2017). La introducción de la robótica educativa en las escuelas es una estrategia que permite trabajar el pensamiento computacional, la creatividad, la curiosidad, el trabajo en equipo y el aprendizaje a partir del error. Estos elementos, según Isnaini & Wawan (2018), pueden incidir en una mejorar de los resultados académicos de los estudiantes y prepararlos para sus profesiones futuras de una mejor manera.

A lo largo del tiempo se han desarrollado varias soluciones tecnológicas con el objetivo de dar un mayor soporte al aprendizaje de la robótica en las escuelas. Dentro de estas soluciones se encuentra *LEGO Mindstorms*, que de acuerdo con Chevalier, Riedo, & Mondada (2016), es el sistema más utilizado y estudiado para aplicaciones de robótica. Consta de un controlador programable, motores, sensores y piezas de construcción diseñados para encajar y funcionar entre sí. Cuenta con versiones, la primera denominada *LEGO Education WeDo 2.0* destinada a niños a partir de los 7 años, y otra versión de nombre *LEGO MINDSTORMS Education EV3* orientada para jóvenes a nivel secundario y superior. Ambas versiones permiten diseñar, construir y programar robots, también poseen aplicaciones móviles que permite programar y explorar funcionalidades de los robots a construir (Fernández & Pastor-Mendoza, 2018).

Otro tipo de soluciones dentro de este campo son las plataformas electrónicas de arquitectura abierta, que hacen uso de un procesador que permite manejar varias instrucciones en conjunto con varios sensores y actuadores. Estas placas se conocen comercialmente como *Arduino* y *Raspberry Pi* (Plaza , Sancristobal, Carro, & Castro, 2016), y se caracterizan por emplear software libre y contar con distintos tipos de proyectos para armar, incluyendo robots. También se integran con aplicaciones móviles enfocadas a su uso como asistentes que permiten la programación y control del robot a distancia.

Estas aplicaciones que funcionan como asistentes, de acuerdo con Cheng, Sun, & Chen (2018) permiten al estudiante de primaria y secundaria ayudarlos en sus tareas para desarrollar el conocimiento básico para armar un robot, y en el caso de estudiantes universitarios complementar su desarrollo mediante guías y sugerencias para trabajos más avanzados con robots.

Un buen ejemplo de uso de esta plataforma en conjunto con este tipo de aplicaciones se encuentra en los robots de *mBlock* desarrollados por *Makeblock*, los cuales cuentan con una aplicación multiplataforma

desde la cual los niños pueden programar los robots de una manera sencilla (Voštinár & Klimová, 2019). *Bee-Bot* también otro ejemplo, que consiste en un robot educativo con forma de abeja diseñado para el desarrollo de las capacidades elementales de programación. Cuenta con una aplicación móvil que permite escribir algoritmos sencillos y enviar instrucciones a distancia al robot, también permite usar una funcionalidad de teclado con la finalidad de mejorar las habilidades de lenguaje direccional de los niños (Pérez Buj & Diago, 2018).

En el Ecuador también se emplean estas tecnologías para el desarrollo de robots. Es así como dentro de los proyectos más representativos se encuentra, por ejemplo, *Teebot* que es el primer robot educativo que ha sido diseñado y ensamblado en el país. Está conformado por un kit de robótica y permite a los niños aprender conceptos de programación, electrónica, matemáticas, tecnología y ciencia (Alarcón, 2015).

En el campo de las aplicaciones orientadas a la robótica educativa se encuentra la aplicación móvil *Yasashun*, que contiene minijuegos y trabaja en conjunto con un asistente robótico, este proyecto está destinado a niños y jóvenes de hasta 12 años y permite el desarrollo de motricidad fina, relación lógica matemática, comprensión y expresión del lenguaje, entre otros (Luna, 2018).

Por lo tanto, con lo anteriormente expuesto, se evidencia algunos aportes importantes que contribuyen al mejorar el estado del arte de la robótica educativa tanto a nivel local como internacional. Para la realización del aporte práctico del presente proyecto, se hará uso de una exhaustiva revisión bibliográfica de las tendencias tecnológicas apegadas al contexto con la finalidad de tomar como punto de partida para su desarrollo.

1.2 Descripción del Problema

Las prácticas educativas tradicionales, de acuerdo con Barrera (2015), se caracterizan por ser unidireccionales y centradas en el maestro,

pero esto ha ido mejorando con la introducción de la tecnología en el campo educativo, puesto que es una necesidad que los ciudadanos del futuro no solo cuenten con un dominio lecto-escritor, sino que también deben de poder interactuar y hacer uso de nuevas tecnologías para comunicarse, puesto que son las herramientas básicas para intercambiar información en el siglo XXI (Sara, 2011). Es precisamente por este advenimiento de nuevas tecnologías para la generación Z y T que se torna necesario un cambio estructural dentro de las actividades de enseñanza- aprendizaje puesto que para ciertos jóvenes pueden presentarse dificultades al abordar un tema empleando solo materiales bibliográficos y explicaciones teóricas (Soria & Rivero , 2019).

Como lo explica el filósofo y académico, Piscitelli (2018), la tecnología siempre ha sido un aliado para la educación, han coexistido y deben de mejorar su integración. Es por este motivo que desde inicios de la última década se ha considerado necesario involucrar a los estudiantes en actividades lúdicas que integren el uso de tecnologías de forma práctica sencilla y contextualizada (Gómez-Álvarez, Manrique-Losada, Arbeláez, Villada, & González, 2019) siendo una de estas la robótica educativa, que de acuerdo con Pittí, Curto, Moreno, & Rodríguez Conde (2014) la describen como el proceso sistemático y organizado, en el que intervienen elementos tecnológicos interrelacionados como herramienta cuyo fin es lograr aprendizaje.

En la ciudad de Guayaquil algunos colegios tanto particulares como fiscales están incluyendo a la robótica educativa dentro de sus actividades de enseñanza, organizan y convocan eventos de torneos y competencias en donde los estudiantes tienen un espacio para innovar y estimular su conocimiento por medio de retos (Sánchez C. , 2019). Sin embargo, no todas estas instituciones cuentan con los recursos o espacios necesarios para que sus estudiantes puedan participar en este tipo de actividades.

Este trabajo de investigación permitiría dotar a los jóvenes de un asistente móvil, programado en el entorno de desarrollo integrado Android

Studio, que se ejecutará en un smartphone por medio del cual permitirá visualizar las instrucciones necesarias para que puedan armar y programar robots de competencias basados en dos versiones de *LEGO Mindstorms: EV3* y *NXT*.

1.3 Alcance

El presente trabajo de titulación se limita al desarrollo de una herramienta tecnológica como lo es un asistente móvil que contribuye a que los estudiantes puedan aprender haciendo y también estimulen su curiosidad en temas relacionados con el uso de las tecnologías, buscando como meta final aportar con una preparación acorde a las necesidades profesionales y creativas que les esperan a futuro en la sociedad. Como entregable, se desarrollará bajo la metodología *Scrum* un asistente móvil, empleando como entorno de desarrollo *Android Studio*, una base de datos *MySQL* y una *RESTful API* alojados en un servidor web. Este asistente móvil se ejecutará en un smartphone con sistema operativo *Android*, el cual se comunicará con un servidor para desplegar información correspondiente al armado y programación de robots para competencias estudiantiles que se armarán empleando los kits *LEGO Mindstorms EV3* y *LEGO Mindstorms NXT*. El usuario que emplee el asistente móvil podrá seleccionar el tipo de robot que desea construir y se indicará por medio de instrucciones visuales los componentes y pasos necesarios para cumplir con este objetivo. Su funcionamiento será evaluado mediante pruebas con estudiantes.

1.4 Objetivos

1.4.1 Objetivo General

Desarrollar un asistente móvil de un laboratorio digital de robótica de competencias estudiantiles tomando como base una revisión exhaustiva de aportes científicos de otros investigadores para su implementación, con la finalidad de contribuir al estado del arte de la robótica educativa.

1.4.2 Objetivos Específicos

1. Realizar una revisión bibliográfica de las tendencias actuales en robótica educativa, estableciendo un punto de partida para el desarrollo del asistente móvil.
2. Implementar un asistente móvil haciendo uso de la metodología *Scrum* con la finalidad de guiar a los estudiantes a construir robots para competencias académicas.
3. Realizar pruebas de la funcionalidad de la aplicación con estudiantes para armar y programar robots.

1.5 Justificación

Los contenidos digitales y tecnológicos que apoyan al aprendizaje son cada vez más utilizados en escuelas y colegios. Esta importancia se genera por la realidad de una sociedad en la que la necesidad de manejar nuevas tecnologías está en constante crecimiento, y es en las aulas donde los futuros profesionales deben formarse para hacer frente a estas exigencias (Orso, Pluchino, Menegatti, & Agatolio, 2018).

De acuerdo con un análisis realizado por la Comisión Europea, se espera que en el 2020 existan aproximadamente 825.000 puestos de trabajo sin ser cubiertos por la falta de profesionales de las ciencias de la computación y tecnologías de la información y comunicación (Rivas, Gálvez, Gértudix Barrio, & Esteban, 2017). Por su parte, en Latinoamérica

de acuerdo con la consultora International Data Corporation (IDC) en el 2019 no se han cubierto 450.000 puestos de trabajo destinados a profesionales con destrezas en tecnologías emergentes (Pineda & González, 2016). Este déficit de profesionales en tecnología se transforma en una barrera no solo para las empresas sino también para los gobiernos.

Sánchez J. (2011) expresa que ante esta realidad existen múltiples programas educativos que proponen la enseñanza de competencias desde la niñez, siendo esta etapa donde se plantea la enseñanza de la robótica como una materia innovadora, que busque motivar a los jóvenes para que puedan aprender a generar proyectos e integrar el factor de colaboración en el grupo.

Denis Salazar, representante de la Sociedad Latinoamericana de Ciencia y Tecnología (SOLACYT) indica que los nuevos desafíos científicos y tecnológicos llevan a los colegios a desarrollar nuevas metodologías de enseñanza donde la robótica permite que los jóvenes también aprendan conceptos de programación, estructuren mejor sus ideas y desarrollen habilidades de pensamiento computacional (Sánchez C. , 2019).

Ante los motivos y datos anteriormente expuestos, la importancia de este trabajo radica en la posibilidad de brindar un asistente móvil que fomente el desarrollo estas habilidades permitiendo a los estudiantes de nivel secundario construir robots para que puedan desarrollar conceptualizaciones que les permitan enfrentarse a problemas cotidianos relacionados al uso apropiado de la tecnología, puesto que estos conocimientos se hacen imprescindibles en la interacción sociocultural y la interacción con el entorno natural de los ciudadanos digitales (Barrera Lombana, 2015).

Para lograr el desarrollo de este asistente se optó por programarlo en el entorno de desarrollo Android Studio por motivo de costos, conocimientos y acceso a equipos para desarrollar las pruebas de funcionalidad de software. En cuanto a los robots a emplearse como base para mostrar sus pasos de programación y armado se seleccionaron dos

versiones de kits *LEGO Mindstorms* como son *EV3* y *NXT* puesto que son apropiados para el tipo de usuarios que se espera empleen el asistente, así como también por su facilidad de acceso durante la cuarentena comunitaria obligatoria establecida en el Decreto Ejecutivo del 16 de marzo del 2020 en su artículo 4.

CAPÍTULO 2 MARCO TEÓRICO

2.1 Introducción

En este capítulo se desarrolla una revisión exhaustiva de temas y tecnologías empleadas para el desarrollo del asistente móvil de un laboratorio digital de robótica de competencias estudiantiles. Se parte desde el concepto de la robótica educativa, seguido de su aplicación en el Ecuador. Posteriormente, se revisan los torneos de robótica y su impacto en la educación. Una revisión de las tecnologías utilizadas en el diseño de robots de competencias que son empleadas en los laboratorios y aplicaciones virtuales utilizadas en el ámbito de la robótica de competencia. Posteriormente se describe brevemente *Scrum* como metodología de desarrollo ágil y las tecnologías necesarias para el desarrollo del asistente móvil como son *Android*, la plataforma en la que se ejecutará el asistente, *Android Studio* como entorno de programación, *MySQL* y *PHP*.

2.2 Robótica Educativa

De acuerdo con García & Caballero (2019) el actual escenario digital exige que se desarrollen estrategias que busquen modernizar los procesos de aprendizaje, incluyendo la necesidad de que los niños y jóvenes adquieran competencias digitales que les permitan desenvolverse apropiadamente en la sociedad altamente tecnificada, para lo que Vilhete & Orlando (2017) proponen como estrategia la incorporación de la robótica educativa como recurso educativo.

La incorporación de este recurso educativo, como lo expresa Barrera Lombana (2015), es coherente con la reconversión de la práctica pedagógica puesto que busca promover los actuales métodos de enseñanza cambiando los roles y funciones de los actores educativos. Bajo esta perspectiva, Oliveira, Bezerra, & Cunha (2017) afirman que se conceptualiza el uso de robots con fines educativos, como herramienta de apoyo para el proceso de enseñanza y aprendizaje.

Esta herramienta, añaden Segovia & Souza (2018), emplea dispositivos robóticos controlados por computadora o máquinas controladas por dispositivos digitales. Pittí, Curto, Moreno, & Rodríguez Conde (2014) formalizan este concepto indicando que la robótica educativa es un proceso sistemático y organizado, en el que existe el uso de elementos tecnológicos interrelacionados (plataforma robótica, dispositivos inteligentes y software de programación) empleados como herramientas mediadoras que buscan como meta lograr aprendizajes. Vilhete & Orlando (2017) también afirman que el interés por la robótica educativa ha incrementado a un ritmo sorprende en la comunidad educativa en los últimos años.

Aprender empleando robots, como lo explica Amy Eguchi (2017), brinda a los estudiantes de nuevas generaciones, como son la Z y T, que se caracterizan por ser generalmente consumidores de tecnología, oportunidades para detenerse, cuestionar y pensar profundamente sobre la tecnología. De esta manera, una metodología educativa que utiliza la robótica educativa puede traer beneficios pedagógicos y socioeconómicos.

2.3 La Robótica Educativa en el Ecuador

El Estado ecuatoriano busca la incorporación este tipo de tecnologías en la educación, tal como lo declara la Constitución en el Artículo 347 Numeral 8, donde se indica la necesidad de incorporar estas tecnologías en el proceso educativo y propiciar el enlace de la enseñanza con las actividades productivas o sociales (Chuquimarca Abad, 2016).

La robótica educativa, como parte de estas tecnologías, no es ajena a su incorporación como herramienta de enseñanza y aprendizaje. Según el doctor Danilo Chávez, jefe de del Laboratorio de Robótica y Sistemas Inteligentes de la Escuela Politécnica Nacional, en el país no ha empezado una exploración profunda de la robótica puesto que recién se está empezando a dar los primeros pasos en comparación con otros países desarrollados (ARCOTEL, 2018).

Adicionalmente se tiene que considerar que, para una implementación apropiada en contenidos de robótica educativa, que incluya actividades, prácticas, tareas y dispositivos, se requiere de un ambiente apropiado con personal capacitado, de manera que en conjunto logren captar la atención de los niños y jóvenes. Para ello, según Preciado Alvarado (2016) se deben de dar soluciones a problemas fundamentales, como se puede apreciar en la Tabla 2.1, para lo cual se requiere transformaciones en el sistema escolar.

*Tabla 2.1
Problemas fundamentales para crear ambientes de aprendizaje con la robótica*

Problemas	Detalle
Falta de lineamientos pedagógicos y metodológicos	El docente no tiene una orientación sobre el aprendizaje y aplicación de la robótica. Las instituciones gubernamentales no proveen de propuestas estructuradas para la implementar la robótica.
Necesidad de formación docente	Para poder emplear los recursos y aplicar una metodología para el aprendizaje de los conceptos y actividades de robótica.
Falta de infraestructura y recursos	Necesarios para dotar a las instituciones educativas para la implementación de plataformas de robótica.

Fuente: Modificación propia basado en (Preciado Alvarado, 2016)

A pesar de los problemas mencionados anteriormente para la creación de estos ambientes, en el país existen reglamentos, instituciones educativas y programas que incorporan la enseñanza de la robótica como parte de la formación académica de sus estudiantes mediante clubes, laboratorios y participación en torneos. Por ejemplo, gracias al Acuerdo Ministerial No. 041-14 se propone que en la malla curricular de Educación General Básica se dediquen tres horas de clases destinadas a clubes orientados al desarrollo de destrezas de los alumnos en su tiempo libre, lo

que ha fomentado en algunas instituciones la creación de clubes de robótica (Nevarez, Cedeño, Quiñonez, & Godoy, 2016).

También existen escuelas de carácter privado de robótica educativa como Robotic Minds, localizada en Quito, fundada en el 2015 y acreditada por la Sociedad Latinoamericana de Ciencia y Tecnología (SOLACYT, 2014). Ofrece un programa de desarrollo de habilidades en robótica diseñado para niños y jóvenes de entre los 6 a 16 años que cuenta con diez niveles de formación en los que se abordan aspectos relacionados a la construcción de robots como la mecánica, programación y electrónica orientadas a la construcción de robots. También cuentan con participación en torneos nacionales e internacionales (Robotic Minds, 2020). Su fundador, Diego Astudillo señala que no espera que sus alumnos sean solo usuarios de la tecnología, sino que espera que los jóvenes desarrollen habilidades para que estén preparados cambios de la sociedad a futuro (Rodríguez, 2019).

Otra escuela de robótica es Robotic Kids, ubicada en Guayaquil, que imparte cursos vacacionales y escuela permanente de robótica proponiendo programas curriculares que potencian las áreas de conocimientos de *STEAM* de manera lúdica mediante programación con Scratch y construcción de robots empleando *Arduino* y *mBlock*. También es una institución acreditada por la Sociedad Latinoamericana de Ciencia y Tecnología, y participa en torneos nacionales e internacionales (Robotic Kids, 2018).

En cuanto a instituciones educativas particulares se refiere, algunas han incorporado la robótica educativa en su malla curricular como, por ejemplo, el Nuevo Mundo, Liceo Panamericano, Delfos, Alemán Humboldt y la incluyen dentro del pènsu. Inclusive ciertas instituciones participan y organizan torneos de robótica, como lo mencionan Villacrés, Sampedro, & Andrade (2020), el Colegio Internacional SEK en el 2018 formó parte del certamen *First Lego League (FLL)* por sus siglas en inglés), este colegio incorpora la robótica como parte de la malla curricular y sus alumnos

dedican al menos una hora a la semana a la robótica aplicada y también la relacionan con otras materias.

También de parte de gobiernos autónomos descentralizados, como por ejemplo el Municipio de Guayaquil, inició en el 2018 un programa bajo el nombre “Guayaquil In”, el cual posee programas, de entre los cuales está “Escuelas In” en el que se busca contribuir al mejoramiento de la calidad en la educación fomentando capacidades académicas y digitales a través de la entrega de computadoras y equipos en los que se incluye a la robótica. Adicionalmente, en el Municipio se ofrecen cursos vacacionales de robótica en los que se enseña a los estudiantes a usar la plataforma *LEGO Mindstorms* para construir robots (Municipalidad de Guayaquil, 2018).

2.4 Torneos de Robótica y su impacto en la educación

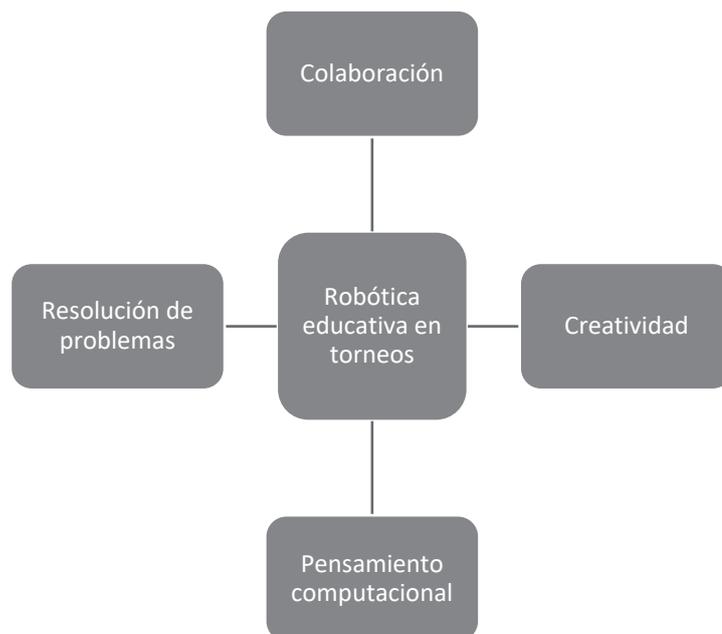
Como se pudo apreciar en la sección anterior, el Ecuador está incursionando en la implementación de la robótica dentro de las actividades de enseñanza. El Gobierno e instituciones educativas buscan los medios para poder dotar de recursos a los docentes y que estos puedan desarrollar actividades de robótica educativa mediante cursos independientes o como parte de las actividades extracurriculares.

Por su parte, los docentes buscan continuamente nuevas maneras de fomentar la curiosidad de sus estudiantes por medio de proyectos que puedan ser realizados en el entorno real, puesto que, como lo especifica Costa, Machado, & Cruz (2017), las nuevas generaciones como la Z suelen gozar de menos concentración, buscan una manera de adaptarse a la tecnología y se caracterizan especialmente por buscar resultados inmediatos.

En este contexto, las competencias y torneos logran captar la atención de los estudiantes y son muchos los que quieren involucrarse en estos. De la misma manera en que son comunes los torneos y competencias de ortografía, matemáticas o ferias de ciencia, la realización de torneos de robótica a nivel nacional ha incrementado en los últimos

años, captando la atención de los estudiantes, docentes y padres que anteriormente se encontraban casi exclusivamente en deportes, matemáticas u ortografía más tradicionales (Calderón & Flórez, 2018).

La construcción de un robot para torneos agudiza las habilidades de los estudiantes, los entusiasma con el uso de las tecnologías vinculadas, no solo aprenden cómo funciona la tecnología, sino que también aplican las habilidades y los conocimientos de los contenidos aprendidos en la escuela de una manera significativa y emocionante (Ruge, Jiménez, & Henández, 2017). También, como lo describen Oliveira de Azevedo, Etiene Bezerra, & Cunha de Miranda (2017), el estudiante desarrolla una mayor autoestima y aprende a trabajar en equipo. En la Figura 2.1 se pueden apreciar los beneficios de armar este tipo de robots para estas actividades.



*Figura 2.1 Beneficios de la implementación robots en competencias
Fuente: Modificación propia basado en (Romero & Dupont, 2106)*

Muchas universidades participan en torneos de robótica de manera extracurricular como parte de sus planes de estudio. Algunos crean torneos de clase o usan kits de robótica para ese propósito. También están los torneos masivos dirigidos a estudiantes de primaria, secundaria y bachillerato. En el Ecuador son varios los torneos e instituciones educativas

que participan y acogen estos. Por citar algunos ejemplos: el Ministerio de Educación menciona que los estudiantes del club de robótica *Robotronic*, obtuvo el primer lugar en la “II Liga Latinoamericana de Robótica Educativa”, realizada en la Universidad de las Américas de Quito (Ministerio de Educación, 2017). La Unidad Educativa COPOL que en el 2019 realizó la primera edición del *Copol Metal Challenge Robotics Competition*, una competencia de robótica para estudiantes de primaria y secundaria de colegios particulares y públicos donde se crearon y probaron robots en varias categorías (COPOL, 2019).

Estos torneos generalmente están conformados por categorías de acuerdo al tipo de robot a armar y también acorde a un reglamento con las especificaciones que deberá de cumplir el robot para que pueda participar. Estos lineamientos suelen ser definidos por el organizador. En la Tabla 2.3 se describen de manera breve ciertas categorías de robots que suelen elaborarse en estos torneos.

Tabla 2.2
Categoría de robots en torneos

Categoría	Detalle
Sumobot o robot sumo	Robots que se caracterizan por empujar a otro hasta sacarlo del área de combate, también conocida como <i>dohyo</i> . Pueden ser autónomos o radio controlados.
Seguidor de líneas	Son robots que como su nombre lo indica, siguen una línea marcada en el suelo que suele ser de color negro en un tablero pintado de blanco. Pueden tener variantes de acuerdo con la complejidad de la línea, circuito o componentes.
Robots de batalla	Destinados al enfrentamiento directo, cuentan con mecanismos para luchar contra otro robot. Pueden variar de acuerdo al peso, su control o autonomía.
Robots de laberinto	Tiene como meta lograr salir de un laberinto. Detecta obstáculos y paredes, trata de buscar la salida en el menor tiempo posible.

Categoría	Detalle
Robofut o Robots de fútbol	Como su nombre lo indica son robots que se disponen en una cancha y tienen como finalidad lograr introducir una pelota en el arco del rival.
Robot Insecto	Como su nombre lo indica son robots que cuentan con atributos similares a los insectos. Su construcción busca fomentar la creatividad.

Fuente: Elaboración propia basado en (Asociación de Educadores en Robótica del Ecuador, 2019)

2.5 Tecnología utilizada en el diseño de robots de competencias

De acuerdo con lo descrito en la Tabla 2.2 en los torneos de robótica suelen haber categorías de robots, los cuales compiten para lograr un reconocimiento en estos eventos. Para poder construir este tipo de robots existen plataformas tecnológicas que cuentan con los dispositivos necesarios para este fin (Romero & Dupont, 2106).

Estas plataformas tecnológicas, según Chevalier, Riedo, & Mondada (2016), se han desarrollado a lo largo del tiempo y brindan un soporte al aprendizaje de la robótica en las escuelas, colegios y universidades. Se pueden considerar de manera representativa la tecnología utilizada en el diseño de robots de competencias que es ofrecida por la empresa *LEGO*. Cuenta con los componentes de hardware y software necesarios para esta labor.

Fernández & Pastor-Mendoza (2018) mencionan que *LEGO*, ofrece plataformas que son muy utilizadas para aplicaciones de robótica. Una de estas es *LEGO WeDo*, destinada para niños; y para jóvenes de nivel de estudios secundario y superior se cuenta con *LEGO Mindstorms*.

Según Sović, Jagušć, & Seršić (2014) *LEGO WeDo* integra la construcción de robots con un software intuitivo. Los niños que emplean esta tecnología suelen trabajar en parejas y distribuyen su tiempo entre armar el robot y hacer el programa para que este funcione. En la Figura 2.2

se puede apreciar un kit básico y un ejemplo del tipo de robot construido con este.



Figura 2.2 Plataforma LEGO WeDo
Fuente: Tomado de (The LEGO Group, 2019)

Por su parte, *LEGO MINDSTORMS* es adecuado para usar en entornos educativos debido a: (1) su versatilidad: porque permite a los estudiantes diseñar, construir y programar una variedad de robots mediante el uso de diseños preparados o los diseños que comparten otros usuarios; (2) su atractivo para el estudiante: estos desean obtener experiencia práctica en el armado y sienten satisfacción inmediata una vez que pueden construir con éxito un robot; (3) sus características: puesto que incluye todos los componentes necesarios para construir una variedad de robots con el uso de sensores, servo-motores y dispositivos programables que controlan el movimiento y la reacción del robot (Afari & Khine, 2017). En la Figura 2.3 se puede apreciar un kit básico y un ejemplo del tipo de robot construido con este.



Figura 2.3 Plataforma LEGO Mindstorms
Fuente: Tomado de (The LEGO Group, 2019)

Ambas plataformas cuentan con un controlador programable, motores, sensores y piezas de construcción diseñados para encajar y funcionar entre sí. En la Tabla 2.3 se pueden apreciar las características y contenidos de un set básico de ambas plataformas.

Tabla 2.3
Comparación de características y contenidos de un set básico de plataforma LEGO

	<i>LEGO Education WeDo</i>	<i>LEGO MINDSTORMS</i>
Dirigido a	Niños a partir de los 7	Estudiantes de nivel de estudios secundario, bachillerato, universitario
Controlador	<i>Smarthub</i>	<i>EV3 brick</i>
Piezas	280 piezas <i>LEGO</i>	541 piezas <i>LEGO</i>
Sensores	Movimiento e inclinación	Giroscopio, ultrasónico de distancia precisa, colores y contacto
Motores	Motor mediano	Tres servo-motores (dos grandes y uno mediano)
Software	<i>LEGO Education WeDo 2.0</i>	<i>LEGO MINDSTORMS Education EV3</i>

Fuente: Elaboración propia basado en (The LEGO Group, 2019)

Cabe mencionar que, aunque la tecnología desarrollada por la empresa *LEGO* es la más empleada en los robots educativos de competencias, en el mercado también existen otras tecnologías como, por ejemplo, *Arduino*, *Tetrix robotics*, *Vex Robotic Competition*, *Makeblock* o *Luxe Tech Robotics* y muchas más las cuales también cuentan con sus propios kits para armado y programación de robots (Terranova, 2017).

2.6 Laboratorios y aplicaciones virtuales de robótica de competencia

Mediante las tecnologías descritas anteriormente es posible la construcción de robots para torneos y competencias, pero lograr este objetivo se debe de considerar algunas etapas o pasos dentro de los cuales los estudiantes requieren considerar los componentes necesarios para el armar un tipo de robot específico, definir los aspectos de diseño mecánico donde se evidencia la integridad estructural del robot para determinar si es apto para su categoría en la competencia. Posteriormente será necesario realizar la programación del robot, es en esta etapa donde existen complementos de software para el uso de las tecnologías revisadas en la sección 2.5, y mediante este complemento se podrá mejorar constantemente el comportamiento del robot y sus funcionalidades dentro de la competencia (Strnad, 2017).

En el caso del *LEGO WeDo*, como mencionan Sović, Jagušć, & Seršić (2014) integra la construcción del robot con un software de control intuitivo basado en arrastrar y soltar. Esta aplicación es multiplataforma y por medio de esta, el docente puede enseñarles a sus alumnos conocimientos básicos de programación.

LEGO Mindstorms, Zhan & Hsiao (2018) indican que el trabajo de armado suele ser en parejas, pero se requiere más tiempo para el diseño y la programación del robot. Esta tecnología también cuenta con su propia aplicación multiplataforma similar a la mencionada anteriormente, orientada a arrastrar y soltar bloques, sin embargo, cuenta con más opciones de

programación. Ofrece a las estudiantes prácticas que permiten desarrollar sus habilidades de resolución de problemas y creatividad.

En las aplicaciones multiplataforma de *LEGO WeDo* y *LEGO Mindstorms* anteriormente descritas, emplean un entorno de programación denominado *Scratch*, que es una de las herramientas más difundidas y utilizadas para introducir y explorar la programación. Fue diseñado para facilitar la participación de los niños y adolescentes en el pensamiento computacional, promoviendo no solo las habilidades informáticas, sino también la resolución de problemas, el pensamiento crítico, la creatividad y las habilidades de colaboración (Azevedo Gomes, y otros, 2019).

En lo referente a aplicaciones orientadas a guiar en el armado de robots, se puede encontrar, por ejemplo, en *Google Play Store* algunas aplicaciones que guiar al usuario en el armado de robots basados en *LEGO*, *Arduino* y otras tecnologías, como *Technic Guide for Nxt* que es una aplicación que contiene indicaciones de armado para robots basados en la plataforma *LEGO* empleando el kit *NXT*. Otro ejemplo es *100+ DIY Arduino Projects* que ofrece al usuario indicaciones para el armado de proyectos, incluidos robots empleando *Arduino* (Google, 2020). En la Figura 2.4 se muestra una captura de ambas aplicaciones.

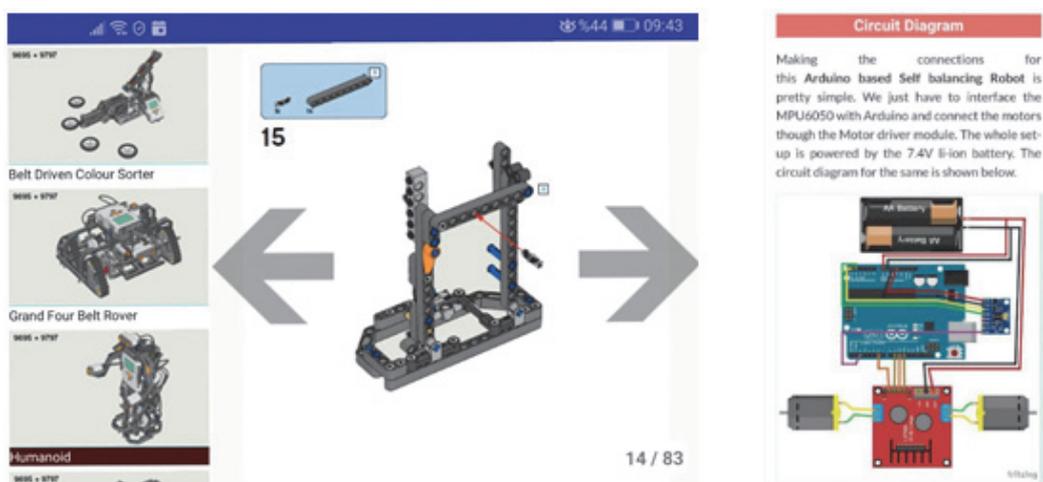


Figura 2.4 Aplicaciones orientadas al armado de robots publicadas en Google Play Store
Fuente: Tomado de (Google, 2020)

Mediante el software y los kits para cada plataforma revisados en el presente trabajo, varias instituciones han creado planes para incorporar la robótica educativa dedicando espacios para estas actividades en laboratorios, como, por ejemplo, en las escuelas primarias de la ciudad de Viseu ubicada en Portugal, se implementaron laboratorios en los cuales los docentes explican y exploran los entornos de programación mediante *Scratch* y los robots *Bee-Bot* (Torrejón Marín & Ventura-Campos, 2019).

Otro ejemplo es el ofrecido por la Universidad de Bolonia en el que se imparten laboratorios para estudiantes con la finalidad de impulsarlos a adquirir conocimientos prácticos sobre robótica y enseñarles cómo escribir código eficiente. Durante estos laboratorios se realizan tareas asignadas a los estudiantes que les permiten diseñar mecánicamente y programar robots para ser utilizados en una competencia entre equipos empleando kits *LEGO Mindstorms*.

En el Ecuador existen proyectos que buscan fomentar la creación de estos laboratorios de robótica, como es el que lleva a cabo la empresa *Bright World Education Enterprises* (BWEE , 2018) que en conjunto con instituciones educativas como, por ejemplo, la Universidad Metropolitana del Ecuador o el colegio John Harvard han establecido convenios para implementar estos laboratorios mediante kits, libros y capacitación para los docentes (Universidad Metropolitana del Ecuador, 2019).

Como se ha podido apreciar, los laboratorios y aplicaciones virtuales para robótica de competencia presentan sus propias variantes de acuerdo con la plataforma tecnológica que implemente la institución educativa que imparte estos contenidos. En el Ecuador, como se mencionó en apartados anteriores, existen instituciones educativas que incluyen estos espacios de competencia y de acuerdo a los proyectos y empresas descritos, se espera que en los próximos años se cuenten con 400 laboratorios de robótica para instituciones tanto públicas como privadas.

En las siguientes secciones se realiza una revisión de los componentes empleados para el desarrollo del asistente móvil propuesto en el presente trabajo.

2.6.1 Scrum

Scrum es una metodología de desarrollo ágil que se caracteriza por su sencillez de uso y facilidad de adaptación por parte de los equipos de trabajo en lo correspondiente al desarrollo de software (Lei, Ganjeizadeh, Jayachandran, & Ozcan, 2017).

El corazón de esta metodología es el sprint, su finalidad es crear un producto que pueda ser usado y potencialmente liberado, a partir de reuniones e inspección de tareas se van agregando funcionalidades al producto final (Shuber, 2013).

El uso de esta metodología, de acuerdo con Babativa, Briceno, Nieto, & Salazar (2016), en conjunto con otras herramientas que apoyan el desarrollo de software y les permitió lanzar una aplicación que integra el GPS del smartphone para realizar el cobro del servicio de taxis.

2.7 Tecnologías de la información

En las siguientes secciones se definen y describen brevemente las tecnologías empleadas para el desarrollo e implementación del asistente móvil.

2.7.1 Android

Android, de acuerdo con Almomani & Khayer (2019), es el sistema operativo para móviles que domina entre los vendedores de teléfonos inteligentes con aproximadamente el 84.8% de la cuota de mercado total en 2018. Además, su uso se extiende a múltiples dispositivos como son relojes, tablets, automóviles o electrodomésticos.

Las aplicaciones desarrolladas para las plataformas destinadas a la robótica educativa de la empresa *LEGO* pueden también ejecutarse en este sistema operativo, y permiten el uso de los sensores que incorpora el smartphone en el que se ejecuta, permitiendo, por ejemplo, el control a distancia mediante una conexión bluetooth de un robot armado (The LEGO Group, 2020).

2.7.1.1 Android Studio

Android Studio es un entorno de desarrollo integrado, que de acuerdo con Craig & Gerber (2015), surge de una colaboración entre *JetBrains* y Google. Es revolucionario porque simplifica el proceso de desarrollo de aplicaciones para *Android* y hace que sea mucho más accesible de lo que había sido anteriormente con otros entornos.

En el 2013 se lanzó la beta de Android Studio, convirtiéndose desde esa fecha en la *IDE* oficial para el desarrollo de aplicaciones para Android (Hagos, 2019). Está basado en *IntelliJ* de *JetBrains* el cual es un *IDE* comercial de *Java* que tiene también una versión para la comunidad gratuita. Esta versión sirve como base para *Android Studio*.

En la actualidad, Android Studio proporciona varias herramientas para crear aplicaciones en cada tipo de dispositivo Android, como por ejemplo un editor de diseño visual, virtualización rápida y ligera, editor de código inteligente, entre otras (Google Developers, 2020).

2.7.2 MySQL

MySQL es un servidor de base de datos relacionales lanzado al público en general a finales de 1996. En la actualidad, y según Dawodi, Hadi Hedayati, Ahmad Baktash, & Latif Erfan (2019), es uno los sistemas de gestión de bases de datos más populares para proveedores de servicios de alojamiento como *Rackspace*, *GoDaddy*, *WHM*, entre otros.

Ofrece las mismas características que se encuentran en los productos patentados de su competencia dentro de esta categoría, sin embargo, destacan sus características de código abierto, ser gratuito y relacional (Kromann, 2018). Adicionalmente, *MySQL* está se puede instalar y configurar en prácticamente todos los servidores de alojamiento y también proporcionan herramientas gratuitas para su configuración y uso.

2.7.3 PHP

PHP es un lenguaje de programación cliente-servidor ampliamente utilizado alrededor del mundo. Varias plataformas web como *Facebook*, *WordPress* o *Yahoo* emplean este lenguaje (Dimes, 2016).

Fue creado a mediados de la década de 1990 por un desarrollador de software canadiense/danés de nombre Rasmus Lerdorf que tenía como finalidad mejorar las capacidades de su sitio web personal (Kromann, 2018).

Actualmente está disponible la versión 7.48 y la versión alfa 8.0, lo que supone que desde su creación se han incluido un incremento del rendimiento, mejoras y correcciones continuas de seguridad (Powers, 2019). Este lenguaje permitirá el desarrollo de una *RESTful API* para atender las solicitudes del asistente.

2.7.4 RESTful API

REST (Representational State Transfer) fue presentado en primer lugar por Roy Fielding en 2000 y es el estándar de la arquitectura de comunicación basada en la web que se aplica principalmente en el desarrollo de servicios basados en este ambiente. Utiliza *HTTP* como protocolo para la comunicación de datos y posee métodos que permiten la lectura, actualización, creación y eliminación de un recurso (Nugroho, Azis, Mustika, & Selo, 2017).

Una *RESTful API* presenta una capa de abstracción que permite la utilización de estos métodos para que dos sistemas, independientemente de su plataforma de uso se comuniquen (Ribas, 2018).

Según lo revisado en este capítulo, la robótica educativa es una poderosa herramienta que brinda a los estudiantes de nuevas generaciones, como son la Z y T, la oportunidad de desarrollar competencias que les permitan desenvolverse apropiadamente en una sociedad altamente tecnificada. En el Ecuador existen instituciones educativas que brindan esta instrucción para niños desde temprana edad, hasta estudios especializados en universidades, sin embargo, se espera que mediante una colaboración entre el Gobierno Ecuatoriano y más instituciones educativas, tanto públicas como privadas, se pueda fomentar su inclusión de una manera más amplia, no solo con kits tecnológicos, sino también con la apropiada construcción de espacios y capacitación para que los docentes puedan adaptar e incorporar nuevas metodologías de enseñanza que involucren al estudiante más en los diferentes cursos que toma mediante la realización de estas actividades.

CAPÍTULO 3 METODOLOGÍA

En este capítulo se revisan los procedimientos necesarios para cumplir con los objetivos y el propósito de la investigación de acuerdo al alcance definido para este trabajo de titulación. En los párrafos siguientes se detallarán el enfoque, alcance y diseño de la investigación

El diseño de la investigación tiene un enfoque exploratorio, cualitativo y aplicado. Es exploratorio porque se realiza una investigación exhaustiva del estado del arte de la robótica educativa a nivel local y nacional, su presencia en las actividades académicas en las instituciones de educación media, el desarrollo de torneos y su impacto en la educación. Es cualitativo porque toma como base la revisión de las teorías y tecnologías empleadas, y ajustadas al contexto local para el desarrollo del asistente móvil. Es aplicada porque se desarrollan soluciones prácticas donde se aplican las teorías expuestas en el marco teórico para la elaboración del aporte práctico de la presente investigación.

Para el desarrollo del asistente móvil, se ha estructurado cuatro fases: revisión y armado de robots de competencias, desarrollo del asistente, ejecución de la experimentación, y análisis de resultados. Sin embargo, es de aclarar que, durante el desarrollo del presente trabajo de titulación, se realizaron ligeras modificaciones debido al marco de emergencia por pandemia sanitaria en el Ecuador en la que por Decreto Ejecutivo del 16 de marzo del 2020 en su artículo 4 se estableció la cuarentena comunitaria obligatoria.

3.1 Etapa 1: Revisión y selección de robots de competencias

En esta etapa se detallan los robots que se armarán y mostrarán en asistente a desarrollarse, para ello se ejecutarán las siguientes actividades:

1. Recopilación de información sobre las plataformas disponibles y a utilizarse para el armado de los robots.
2. Selección de las categorías de robots, registro de componentes y pasos tanto para el armado y programación de cada robot.

3.2 Etapa 2: Desarrollo del asistente

Para el asistente a desarrollar, se elaboró un esquema correspondiente a su funcionamiento general que se puede apreciar en la Figura 3.1. El asistente se ejecutará en un smartphone y se plantea su uso en un laboratorio digital de robótica de competencias.

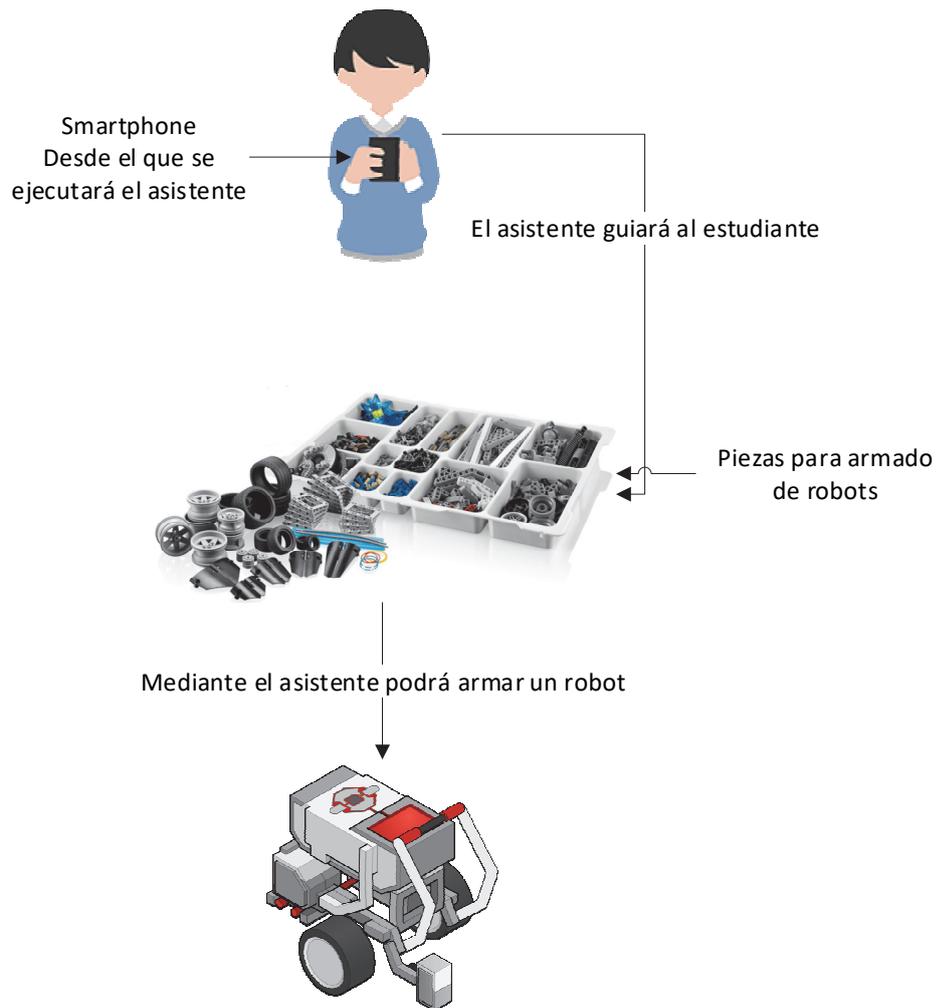


Figura 3.1 Diagrama de funcionamiento del asistente
Fuente: Elaboración propia

La interacción entre el asistente y el niño le permitirá cumplir con el objetivo de construir un robot de competencias de la siguiente manera:

1. El usuario inicia el asistente en el smartphone.

2. El asistente móvil muestra los robots por categorías que puede armar.
3. Seleccionado el tipo de robot, el asistente móvil le consulta al usuario por la tecnología en la que podrá armar el robot.
4. Seleccionada la tecnología, el asistente móvil le consulta al usuario si desea desplegar los pasos de armado del robot o su programación.
5. Indicado el tipo de pasos que desea visualizar, el asistente muestra mediante imágenes y una breve descripción los pasos a seguir para concluir su actividad de armado o programación.

Para el desarrollo del asistente propuesto en el presente trabajo se empleó la metodología ágil *Scrum*, que de acuerdo con Bermón Angarita & Guapachá Hernández (2019) es una de las que cuenta con mayor uso en equipos de desarrollo en los que los proyectos de software pueden presentar cambios en sus requerimientos (Temitayo Faniran, Badru, & Ajayi, 2017), y es aquí donde la gestión que realiza le otorga una gran ventaja como metodología.

Siguiendo los lineamientos de *Scrum*, se definen las historias de usuario para luego especificar las tareas que conformarán los sprints como se puede apreciar en la Tabla 3.1.

Tabla 3.1
Sprints y tareas

Sprint 1	Sprint 2	Sprint 3	Sprint 4
T1. Diseñar la arquitectura de base de datos	T1. Detallar funciones de <i>RESTful API</i>	T1. Diseñar bocetos	T1. Ingresar al asistente
T2. Crear estructuras	T2. Desarrollar <i>RESTful API</i>	T2. Establecer la comunicación de actividades	T2. Seleccionar robots
		T3. Diseñar las actividades en el	T3. Selección por versión y tipo de paso

Sprint 1	Sprint 2	Sprint 3	Sprint 4
		entorno de programación	
			T4. Visualizar instrucciones de armado y programación

Fuente: Elaboración propia

Como se puede observar se definieron cuatro sprints con sus respectivas tareas (T#). El Sprint 1 está enfocado en las tareas relacionadas con la arquitectura y creación de la base de datos. El Sprint 2 contiene tareas destinadas a la creación de la *RESTful API*. En el Sprint 3 se ejecutarán tareas encaminadas al diseñar las actividades que contendrán el asistente. En el Sprint 4 se realiza la codificación del asistente y pruebas para validar su funcionamiento. En las siguientes secciones se revisará el desarrollo de cada una de estos sprints.

3.3 Etapa 3: Ejecución de la experimentación

Debido a la suspensión de las clases en modalidad presencial en el territorio ecuatoriano por motivos de la pandemia COVID 19, se realizará la evaluación del asistente con un estudiante de secundaria de la ciudad de Guayaquil. Se explicará el funcionamiento y se facilitarán los recursos necesarios para que puedan efectuar el armado de un robot de competencias. Detallados los sujetos de experimentación, los pasos para la ejecución de la experimentación se detallan en la Tabla 3.2.

Tabla 3.2
Pasos para ejecutar la experimentación

Actividad	Descripción
Instrucciones de uso	El instructor ofrecerá las indicaciones necesarias para que el estudiante pueda emplear el asistente móvil y los kits <i>LEGO</i> .
Uso del asistente	Será un periodo en el que el estudiante podrá emplear el asistente móvil para armar los robots.
Ejecución de la evaluación	Se evaluará con un estudiante la funcionalidad del asistente móvil en la tarea de armado de un robot de competencias a su elección, con intervención mínima del instructor.

Fuente: *Elaboración propia*

3.4 Etapa 4: Análisis de resultados

Concluida la validación de la parte funcional realizada empleando el asistente móvil desarrollado con un estudiante de secundaria, se analizará la utilidad del asistente durante el proceso de armado de robots, si pudo armarlo sin mayor dificultad, si se pudo reducir el tiempo para estas tareas empleando el asistente considerando la ausencia de preparación o conocimientos previos en los temas relacionados a la robótica educativa.

3.5 Resumen de la metodología

El resumen del diseño metodológico propuesto para el desarrollo de un asistente móvil de un laboratorio digital de robótica de competencias estudiantiles se muestra en la Figura 3.2.

Etapa 1: Revisión y selección de robots de competencias

- Registro y selección de componentes basados en tecnologías

Etapa 2: Desarrollo del asistente

- Definir los sprints
- Codificación del asistente

Etapa 3: Ejecución de la experimentación

- Uso del asistente desarrollado
- Evaluación del asistente

Etapa 4: Análisis de resultados

- Revisar evaluaciones.

Figura 3.2 Resumen del diseño metodológico
Fuente: Elaboración propia

CAPÍTULO 4 DESARROLLO DE LA PROPUESTA

En este capítulo se desarrollan las etapas definidas en el diseño metodológico revisado anteriormente. Se muestra el desarrollo del asistente, sus pruebas y la experimentación con los estudiantes de secundaria.

4.1 Etapa 1: Revisión y selección de robots de competencias

4.1.1 Recopilación de información básica

Se emplea la tecnología de *LEGO Mindstorms* mediante el uso de kits en dos versiones: *EV3* y *NXT*. Estos recursos serán utilizados tanto como para el armado y programación de los robots. También sirvieron como fuente de información para elaborar el asistente móvil. En los siguientes apartados se describen brevemente sus principales características.

4.1.1.1 LEGO Mindstorms NXT

LEGO Mindstorms NXT es una de las plataformas desarrolladas por la compañía *LEGO* y lanzada en julio de 2006, dirigida a la educación tecnológica (Benedettelli, 2008).

Está compuesto por un procesador *ARM7*, software patentado, tres servomotores y sensores táctiles, de color y ultrasonidos, que permiten la programación y el ensamblaje de robots con nociones de distancia, capaces de reaccionar ante obstáculos y colores, y ejecutar movimientos con cierto grado de precisión (Benedettelli, 2008).

Adicionalmente, el kit permite la programación de comportamientos, lo que resulta en el desarrollo de modelos interactivos vinculados a los conceptos de ciencia e ingeniería (Banachi Barreto, L'Erario, & Augusto Fabri, 2015).

4.1.1.2 LEGO Mindstorms EV3

EV3 está disponible para su compra desde el 2013 y corresponde a la tercera generación de la plataforma *LEGO Mindstorms*. Su característica más importante es el nuevo procesador, un *ARM9*. El procesador ejecuta un sistema operativo basado en Linux, que ofrece muchas posibilidades nuevas a los desarrolladores (Rollin, 2014).

LEGO también permitió a la comunidad de desarrolladores que el código fuente del firmware y la documentación estén disponibles gratuitamente, lo que permite una fácil extensión de la funcionalidad del kit y los robots (Rodríguez Perez, Fatoum, & Abke, 2016).

Ambas plataformas cuentan con un controlador programable, motores, sensores y piezas de construcción diseñados para encajar y funcionar entre sí. En la Tabla 4.1 se pueden apreciar las características y contenidos de un set básico de ambas plataformas.

Tabla 4.1
Comparación de características y contenidos de un set *LEGO Mindstorms EV3* y *NXT*

	<i>NXT</i>	<i>EV3</i>
Año de lanzamiento	2006	2013
Procesador del bloque	ARM7	ARM9
Piezas	619	541
Sensores	Ultrasónico de distancia precisa, colores y contacto	Giroscopio, ultrasónico de distancia precisa, colores y contacto
Motores	Tres servomotores	Tres servo-motores

Fuente: Elaboración propia basado en (The LEGO Group, 2019)

Cabe mencionar que *EV3* es retro compatible y se pueden emplear los conectores, sensores, motores, alimentación de un set *NXT* para armar un robot con *EV3*, así como también, en el apartado de software si se

cuenta con los dos kits, el usuario puede optar por programar las instrucciones en el software del *EV3* para ambos kits (LEGO Group, 2018).

4.1.2 Selección de las categorías, registro de componentes y pasos

Revisadas las versiones de *LEGO Mindstorms* que se van a emplear en el presente proyecto, como categorías de robots de competencias se han seleccionado el:

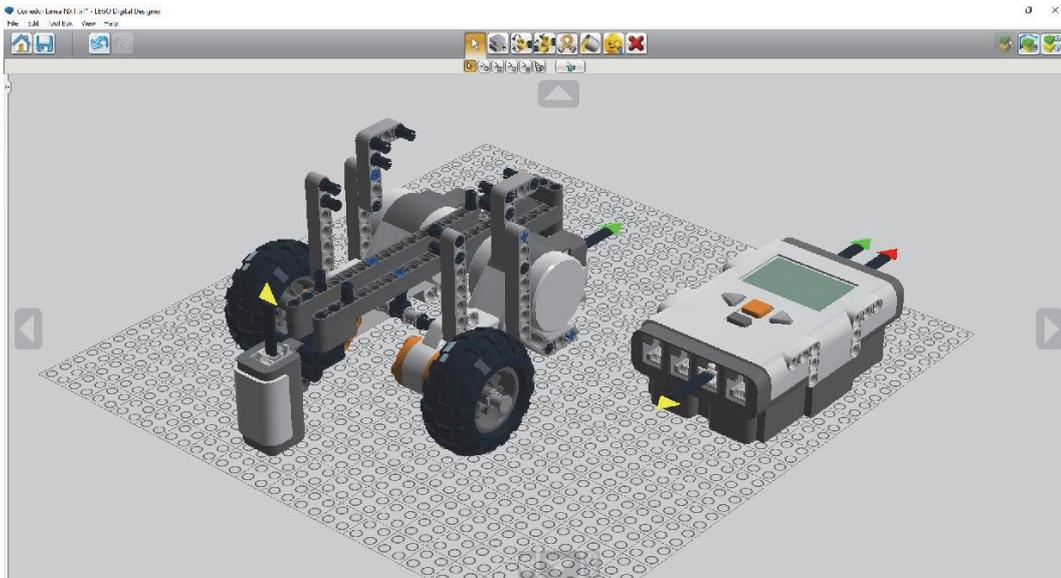
- Robot sumo
- Seguidor de línea
- Robot insecto.

Para cada una de estas categorías se armará su robot correspondiente indicando los recursos necesarios de acuerdo con las versiones *EV3* y *NXT* realizando la secuencia de tareas que se muestra en la Figura 4.1.



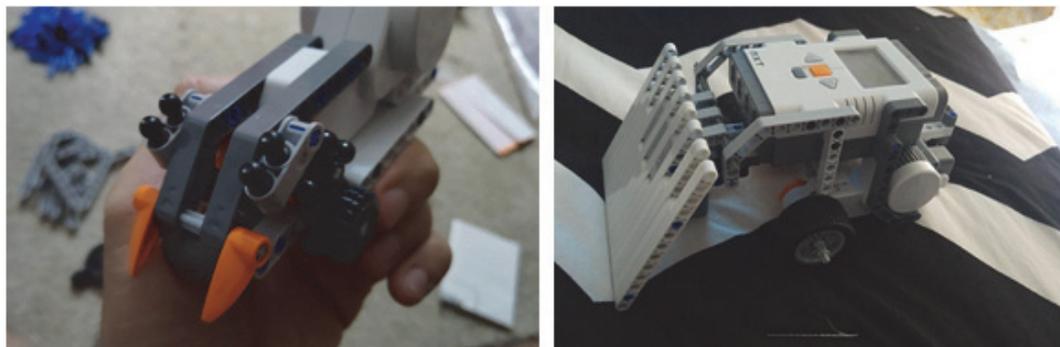
Figura 4.1 Tareas realizadas para el armado de cada robot
Fuente: Elaboración propia

Para cada robot, se empezó con una tarea de diseño empleando el software *LEGO Digital Designer*. De esta manera se pudo simular el armado por pasos de cada robot como se muestra en la Figura 4.2 y obtener imágenes que posteriormente se visualizarán en el asistente móvil.



*Figura 4.2 Simulación de armado empleando LEGO Digital Designer
Fuente: Elaboración propia*

Realizada esta simulación, empieza la tarea de armado del robot empleando los kits de las dos versiones de *LEGO Mindstorms* definidas anteriormente, como se puede apreciar en la Figura 4.3. Para cada tarea de armado se registraron los pasos e imágenes necesarias que serían incluidas para mostrarse en el asistente.



*Figura 4.3 Armado de robots empleando los kits EV3 y NXT de LEGO Mindstorms
Fuente: Elaboración propia*

Posteriormente, en la tarea de programación, se indican las instrucciones mediante las cuales se programará el bloque inteligente para darle el funcionamiento apropiado al robot, como se muestra en la Figura 4.4.

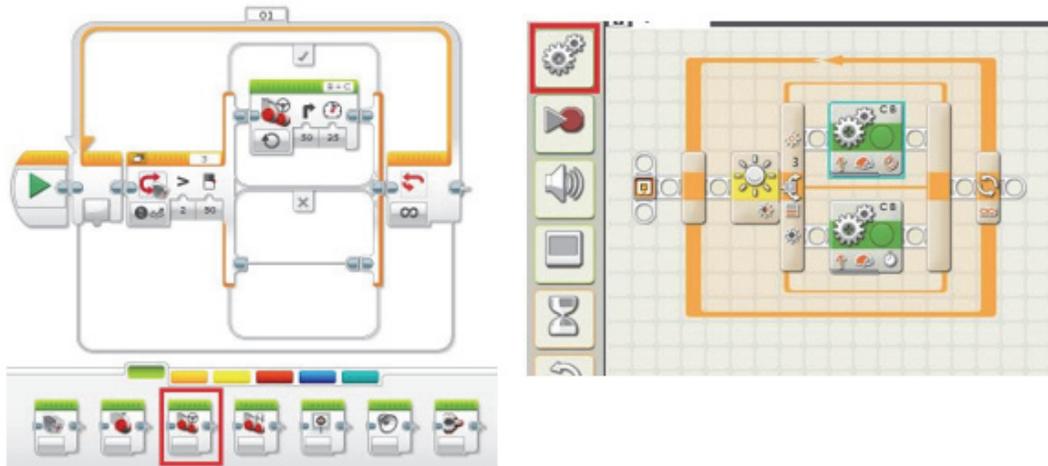


Figura 4.4 Programación de robots EV3 y NXT
Fuente: Elaboración propia

4.1.2.1 Robot sumo

Implementación con EV3

En Tabla 4.2 se detallan las principales características del robot sumo implementado con el kit *LEGO Mindstorms EV3*.

Tabla 4.2
Características del robot sumo implementado con *LEGO Mindstorms EV3*

Categoría		EV3
Robot sumo	# Piezas	183
	# Piezas principales	<ul style="list-style-type: none"> ▪ Brick EV3 x1 ▪ Servomotor x2 ▪ Sensor de color x1 ▪ Sensor ultrasónico x1
	# pasos de armado	103
	# de pasos de programación	18

Fuente: Elaboración propia

En la Figura 4.5 se muestra parte del proceso de armado del robot sumo detallado en la Tabla 4.2 empleando el *kit LEGO Mindstorms EV3*.

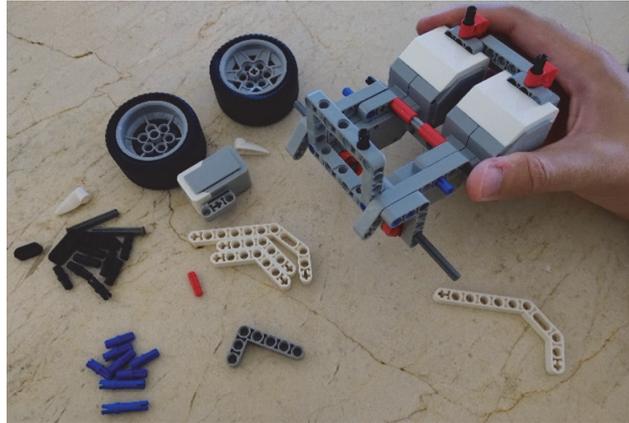


Figura 4.5 Armado del robot sumo empleando el kit LEGO Mindstorms EV3
Fuente: Elaboración propia

En la Figura 4.6 se muestra el robot sumo armado y programado empleando el kit *LEGO Mindstorms*. En esta competencia ambos robots competidores son colocados en el *dohyo* uno a espaldas del otro. Este robot hace un giro de 180° para colocarse frente al adversario y tratar de empujarlo fuera. En caso de que no tenga a su adversario de frente tiene un sensor de color que le permitirá detectar los límites del ring para no salirse del mismo y voltearse para tratar de encontrarlo. También cuenta con un sensor ultrasónico para detectar si tiene a su oponente en frente para activar la mayor velocidad de los servomotores.

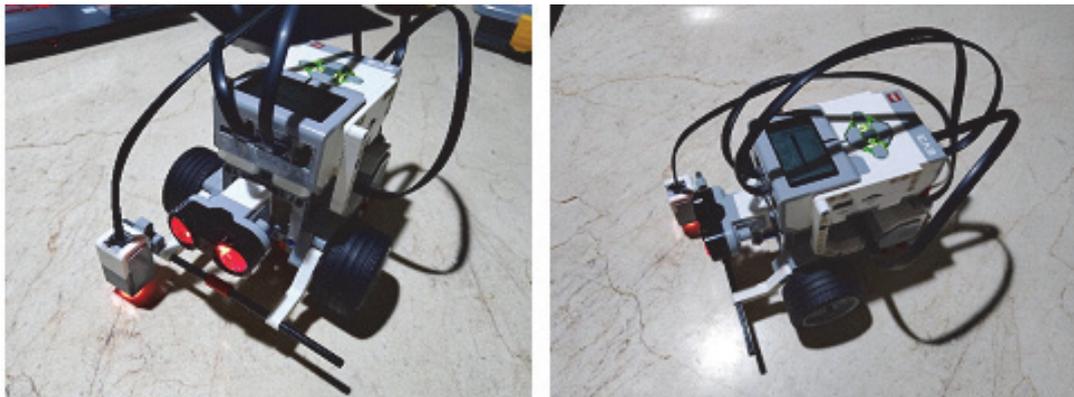


Figura 4.6 Robot sumo empleando el kit LEGO Mindstorms EV3
Fuente: Elaboración propia

Implementación con NXT

En Tabla 4.3 se detallan las principales características del robot sumo implementado con el kit *LEGO Mindstorms NXT*.

Tabla 4.3

Características del robot sumo implementado con *LEGO Mindstorms NXT*

Categoría		NXT
Robot sumo	# Piezas	135
	# Piezas principales	<ul style="list-style-type: none">▪ Brick NXT x1▪ Servomotor x2▪ Sensor de color x1
	# pasos de armado	45
	# de pasos de programación	14

Fuente: Elaboración propia

En la Figura 4.7 se muestra parte del proceso de armado del robot sumo detallado en la Tabla 4.3 empleando el kit *LEGO Mindstorms NXT*.



Figura 4.7 Armado del robot sumo empleando el kit *LEGO Mindstorms NXT*

Fuente: Elaboración propia

En la Figura 4.8 se muestra el robot sumo armado y programado empleando el kit *LEGO Mindstorms*. Al igual que su contraparte con EV3, al ser colocado en el *dohyo*, hace un giro de 180° para colocarse cara a cara con su adversario y tratar de empujarlo. También cuenta con un sensor de color para detectar los límites del ring.



Figura 4.8 Robot sumo empleando el kit LEGO Mindstorms NXT
Fuente: Elaboración propia

4.1.2.2 Seguidor de línea

Implementación con EV3

En Tabla 4.4 se detallan las principales características del robot seguidor de línea implementado con el kit *LEGO Mindstorms EV3*.

Tabla 4.4
Características principales de los robots seguidores de línea armados

Categoría		EV3
Seguidor de línea	# Piezas	88
	# Piezas principales	<ul style="list-style-type: none"> ▪ Brick EV3 x1 ▪ Servomotor x2 ▪ Sensor de color x1
	# pasos de armado	34
	# de pasos de programación	6

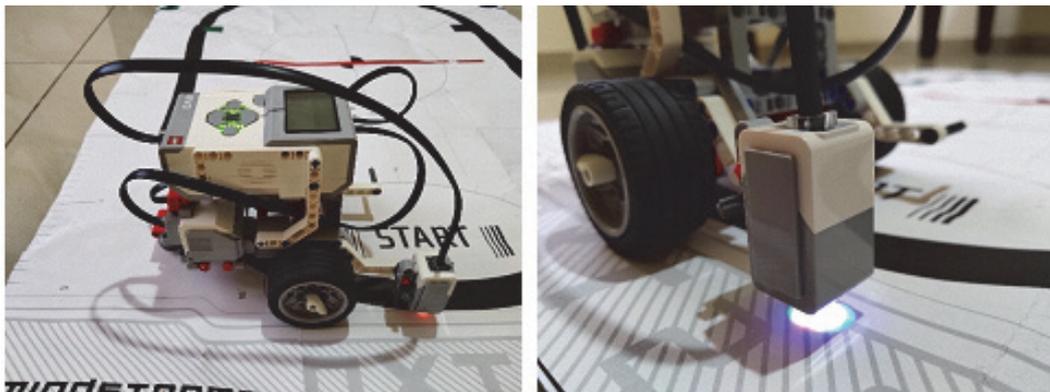
Fuente: Elaboración propia

En la Figura 4.9 se muestra parte del proceso de armado del robot seguidor de línea detallado en la Tabla 4.4 empleando el *kit LEGO Mindstorms EV3*.



*Figura 4.9 Armado del robot seguidor de línea empleando el kit LEGO Mindstorms EV3
Fuente: Elaboración propia*

En la Figura 4.10 se muestra el robot seguidor de línea armado y programado. Este robot tiene como objetivo seguir el circuito en el que ha sido colocado, en este caso una pista. Con la ayuda del sensor de color el robot distingue si está sobre la línea negra o blanca de la pista, de esta manera puede controlar la velocidad de los servomotores para que pueda alinear el recorrido y velocidad de cada rueda.



*Figura 4.10 Robot seguidor de línea empleando el kit LEGO Mindstorms EV3
Fuente: Elaboración propia*

Implementación con NXT

En Tabla 4.5 se detallan las principales características del robot seguidor de línea implementado con el kit *LEGO Mindstorms NXT*.

Tabla 4.5
Características principales de los robots seguidores de línea armados

Categoría		NXT
Seguidor de línea	# Piezas	92
	# Piezas principales	<ul style="list-style-type: none">▪ Brick NXT x1▪ Servomotor x2▪ Sensor de color x1
	# pasos de armado	91
	# de pasos de programación	10

Fuente: Elaboración propia

En la Figura 4.11 se muestra parte del proceso de armado del robot seguidor de línea detallado en la Tabla 4.5 empleando el *kit LEGO Mindstorms NXT*.

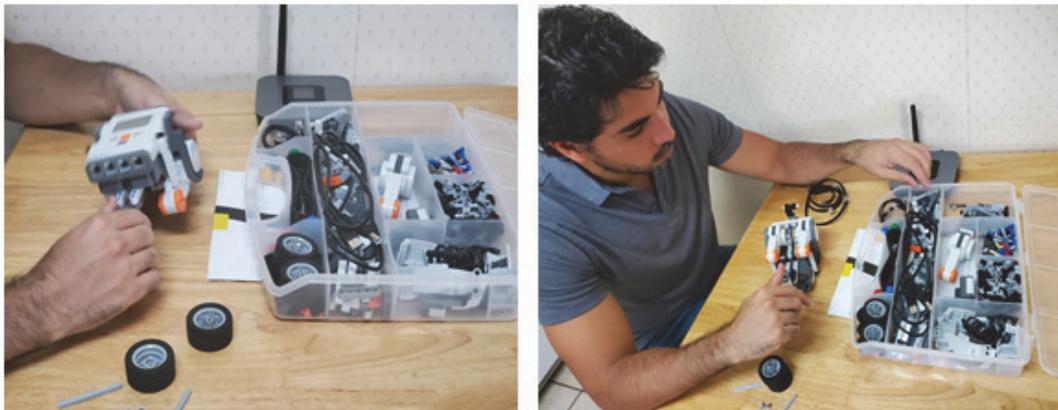


Figura 4.11 Armado del robot seguidor de línea empleando el kit LEGO Mindstorms NXT
Fuente: Elaboración propia

En la Figura 4.12 se muestra el seguidor de línea armado y programado empleando el *kit LEGO Mindstorms NXT* se usó una pista para

probar el funcionamiento del robot. Tiene la misma funcionalidad que su contraparte EV3.



Figura 4.12 Robot seguidor de línea empleando el kit LEGO Mindstorms NXT
Fuente: Elaboración propia

4.1.2.3 Robot insecto

Implementación con EV3

En Tabla 4.6 se detallan las principales características del robot insecto implementado con el kit *LEGO Mindstorms EV3*.

Tabla 4.6
Características principales del robot insecto armado con *LEGO Mindstorms EV3*

Categoría		EV3
Robot insecto	# Piezas	207
	# Piezas principales	<ul style="list-style-type: none"> ▪ Brick EV3 x1 ▪ Servomotor x2 ▪ Sensor Táctil x1 ▪ Sensor Ultrasónico x1
	# pasos de armado	73
	# de pasos de programación	21

Fuente: Elaboración propia

En la Figura 4.13 se muestra parte del proceso de armado del robot insecto detallado en la Tabla 4.6 empleando el *kit LEGO Mindstorms EV3*.



Figura 4.13 Armado del robot insecto empleando el kit LEGO Mindstorms EV3
Fuente: Elaboración propia

En la Figura 4.14 se muestra el robot insecto armado y programado. Este robot está programado para que realice un movimiento giratorio y avance hasta un obstáculo, que puede detectarlo con el sensor de táctil, para detenerse y activar una alerta sonora imitando a un insecto.

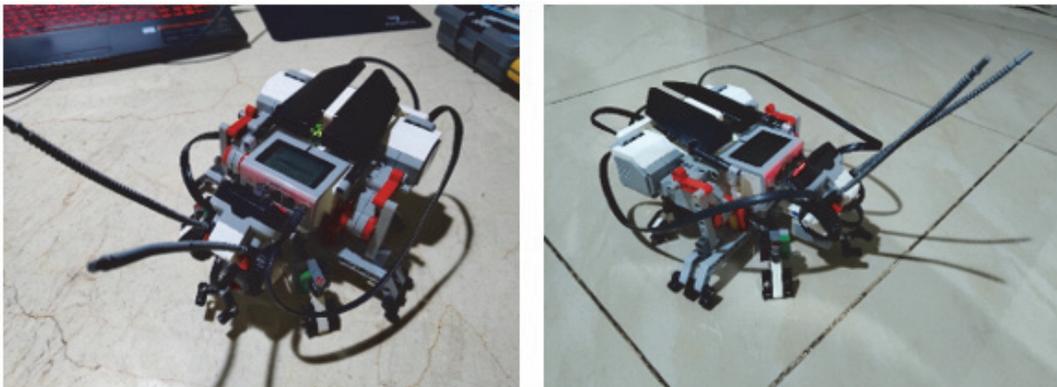


Figura 4.14 Robot insecto empleando el kit LEGO Mindstorms EV3
Fuente: Elaboración propia

Implementación con NXT

En Tabla 4.7 se detallan las principales características del robot insecto implementado con el *kit LEGO Mindstorms NXT*.

Tabla 4.7
Características principales del robot insecto armado con LEGO Mindstorms EV3

Categoría		NXT
Robot insecto	# Piezas	101
	# Piezas principales	<ul style="list-style-type: none"> ▪ Brick NXT x1 ▪ Servomotor x1
	# pasos de armado	53
	# de pasos de programación	15

Fuente: Elaboración propia

En la Figura 4.15 se muestra parte del proceso de armado del robot insecto detallado en la Tabla 4.7 empleando el *kit LEGO Mindstorms NXT*.



Figura 4.15 Armado del robot insecto empleando el kit LEGO Mindstorms XT
Fuente: Elaboración propia

En la Figura 4.16 se muestra el robot insecto armado y programado. Este robot es controlado por medio del *brick NXT* mediante el cual el usuario puede indicar comandos para que este se desplace hacia adelante, retroceda, o se detenga.

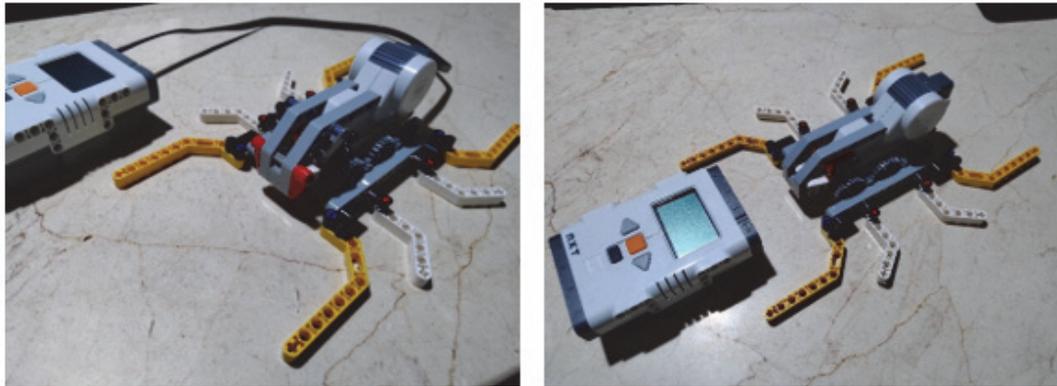


Figura 4.16 Robot insecto empleando el kit LEGO Mindstorms NXT
Fuente: Elaboración propia

4.2 Etapa 2: Desarrollo del asistente

En esta etapa se diseñó un diagrama de la arquitectura del asistente que se muestra en la Figura 4.17 en la que se puede apreciar como el usuario puede acceder desde un teléfono Android a los datos que se visualizarán en el asistente móvil.

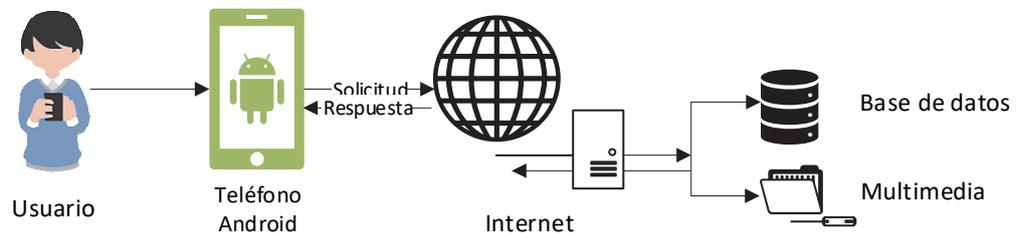


Figura 4.17 Arquitectura del asistente
Fuente: Elaboración propia

En la Figura 4.18 se puede observar un diagrama de secuencia donde se muestran módulos que conforman el asistente. Se puede apreciar que el usuario inicia el asistente móvil desde el smartphone. El asistente por su parte, cuenta con un módulo de acceso a la red en el que se verifica si el dispositivo cuenta con acceso a internet antes de realizar cualquier operación. Realizada esta operación se envía la petición que será atendida por la *RESTful API* en la que se verifican los parámetros para posteriormente acceder y realizar la consulta necesaria a la base de datos

y enviar la respuesta necesaria para que sea manejada apropiadamente por la capa de presentación del asistente móvil.

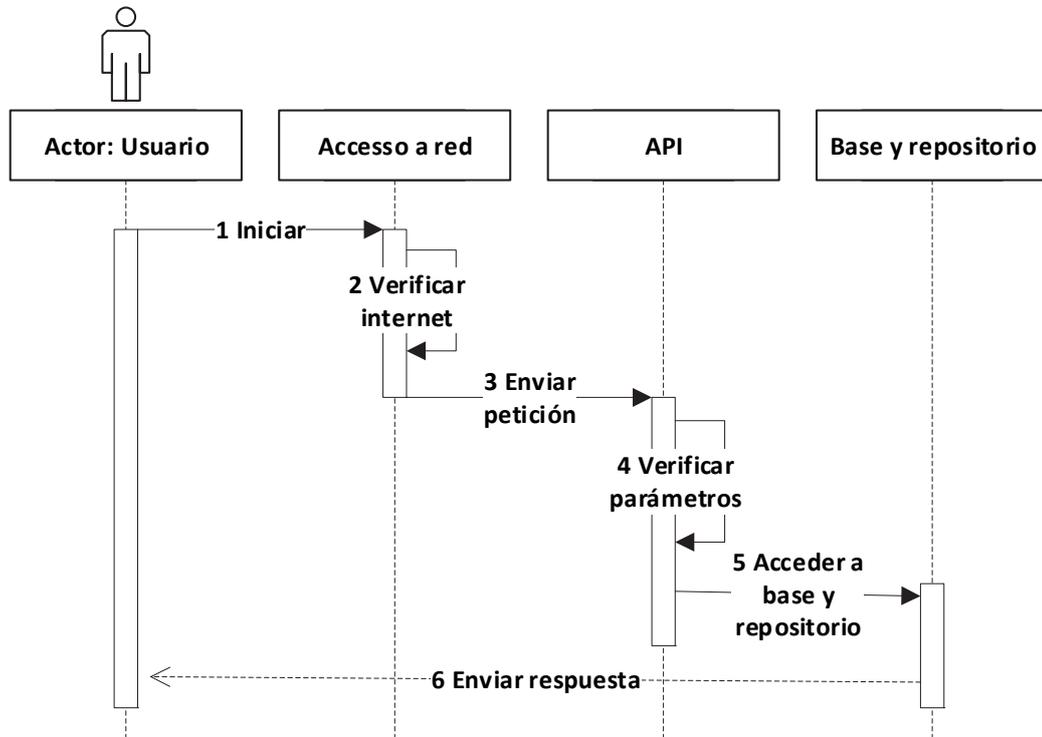


Figura 4.18 Diagrama de secuencia del asistente
Fuente: Elaboración propia

4.2.1 Planeación de sprints

Definida la arquitectura se establecieron las actividades relacionadas que permitan implementar funcionalidades dentro de la aplicación como se pudo observar en la Tabla 3.1 definida en el Capítulo 3 del presente trabajo.

En las siguientes secciones se desarrollarán cada una de las tareas definidas para cada sprint.

4.2.2 Sprint 1

Este sprint tiene con finalidad crear las estructuras necesarias en la base de datos para albergar la información referente a los robots y sus

pasos tanto de armado como de programación. Se trabajo con la base de datos *MySQL* y su software *MySQL Workbench*.

4.2.2.1 Planificación del Sprint 1

Para cada sprint, las tareas son evaluadas de acuerdo al esfuerzo en horas estimado a requerirse para completarlas. En la Tabla 4.8 se detallan las tareas y su esfuerzo.

Tabla 4.8
Tareas del Sprint 1 y su esfuerzo

T#	Tareas planificadas	Esfuerzo
T1	Diseñar la arquitectura de base de datos	5h
T2	Crear estructuras	3h
Total		8h

Fuente: Elaboración propia

Con base en las tareas definidas anteriormente para este sprint, se elaboró una planificación del trabajo en horas necesarias para completarlas, lo cual se puede observar en la Tabla 4.9. Mediante esta planificación se espera entregar la base de datos con sus estructuras creadas.

Tabla 4.9
Planificación de tareas del Sprint 1

	T1	T2	Total
Día 1	2h	-	2h
Día 2	3h	-	3h
Día 3	-	2h	2h
Día 4	-	1h	1h
Total	5h	3h	8h

Fuente: Elaboración propia

4.2.2.2 Desarrollo del Sprint 1

T1. Diseñar la arquitectura de base de datos

En esta tarea se diseñó la arquitectura de base de datos estableciendo las estructuras necesarias. Esta tarea se ejecutó empleando primero el software Microsoft Visio y posteriormente MySQL Workbench. El diseño se muestra en la Figura 4.19.

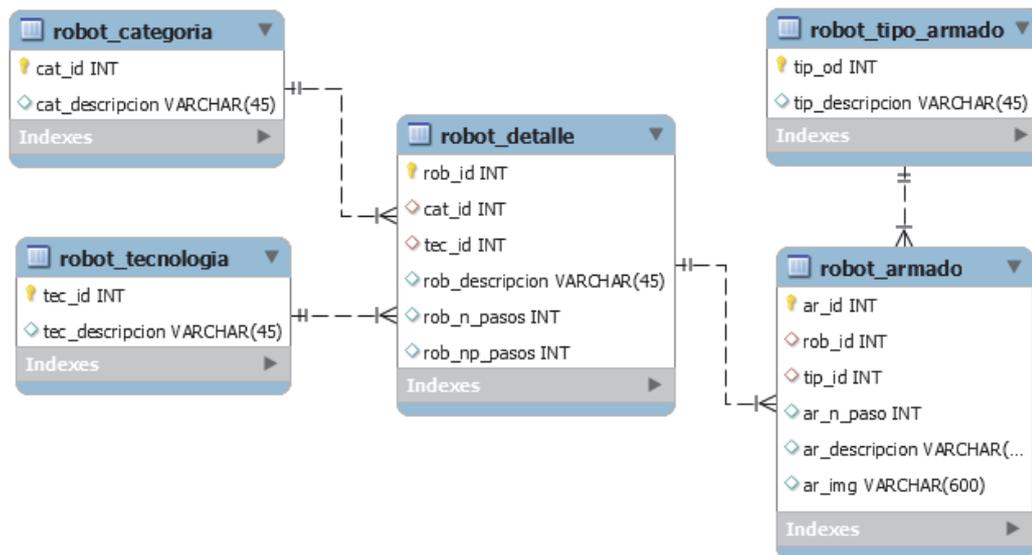


Figura 4.19 Arquitectura de base de datos del asistente
Fuente: Elaboración propia

T2. Crear estructuras

Diseñada la base de datos, se escribieron las sentencias para crear las tablas y sus relaciones en *MySQL Workbench*. Entre las tablas que conforman base tenemos:

- **robot_categoria:** Es una tabla destinada a registrar la descripción de las categorías de robots de competencias.
- **robot_tecnologia:** En esta tabla se almacenan las versiones de *LEGO Mindstorms* disponibles para armar y programar robots.
- **robot_detalle:** Tiene registros de las características de cada robot.
- **robot_tipo_armado:** Es una tabla que sirve para identificar si un paso corresponde a un armado físico o pasos para programar un robot.

- **robot_armado:** En esta tabla se registran los pasos con sus detalles tanto para el armado y programación del robot.

4.2.2.3 Revisión del Sprint 1

En la revisión del Sprint 1 se verificó el cumplimiento de las actividades planificadas en cuanto a la arquitectura de la base de datos que empleará el asistente, como se puede apreciar en la Tabla 4.10.

*Tabla 4.10
Verificación de tareas completadas en el Sprint 1*

T#	Tareas planificadas	Estado
T1	Diseñar la arquitectura de base de datos	Completado
T2	Crear estructuras	Completado

Fuente: Elaboración propia

En la Tabla 4.11 se muestra el resumen de las tareas completadas y su tiempo de desarrollo, donde se refleja que se tomó una hora adicional para completar las tareas planificadas en la Tabla 4.9.

*Tabla 4.11
Lista de tareas completadas en el Sprint 1*

Tareas completadas (T#)		Total
Día 1	T1. Diseñar la arquitectura de base de datos	2h
Día 2		3h
Día 3	T2. Crear estructuras	4h
Día 4		2h
Total		9h

Fuente: Elaboración propia

Después de haber completado todas estas actividades el asistente ya cuenta con una base de datos. En el siguiente sprint se enfocará en actividades de desarrollo de la RESTful API.

4.2.3 Sprint 2

En este sprint se ejecutan tareas encaminadas a la definición de funciones y codificación de la RESTful API las cuales permitirán manejar la información que se encuentra alojado en un servidor web.

4.2.3.1 Planificación del Sprint 2

En este sprint se ejecutarán tareas enfocadas para el desarrollo de la API que permitirá atender las peticiones de la aplicación y acceder a los recursos necesarios. En la Tabla 4.12 se detallan las tareas y su esfuerzo.

Tabla 4.12
Tareas del Sprint 2 y su esfuerzo

T#	Tareas planificadas	Esfuerzo
T1	Detallar funciones de RESTful API	8h
T2	Desarrollar RESTful API	12h
Total		20h

Fuente: Elaboración propia

En la Tabla 4.13 se muestra la planificación del trabajo en horas necesarias para completar las tareas definidas en este sprint.

Tabla 4.13
Planificación de tareas del Sprint 2

	T1	T2	Total
Día 1	3h	-	3h
Día 2	2h	-	2h
Día 3	3h	-	3h
Día 4	-	4h	4h
Día 5	-	4h	4h

	T1	T2	Total
Día 6	-	2h	2h
Día 7	-	2h	2h
Total	8h	12h	20h

Fuente: *Elaboración propia*

4.2.3.2 Desarrollo del Sprint 2

T1. Detallar funciones

En esta tarea se definieron las funciones que contará la *RESTful API* para atender los requerimientos de comunicación entre el asistente y el servidor. Estas funciones se pueden apreciar en la Tabla 4.14.

Tabla 4.14
Funciones de la *RESTful API*

Función	Descripción
Robots	Permite realizar acciones para crear, editar y borrar registros referentes a los robots registrados en la base datos.
Categorías	Permite realizar acciones para crear, editar y borrar registros referentes a las categorías de robots registrados en la base datos.
Tecnologías	Permite realizar acciones para crear, editar y borrar registros referentes a las tecnologías de robots registrados en la base datos.
Pasos de armado	Permite realizar acciones para crear, editar y borrar registros referentes a pasos de armado de un robot registrados en la base datos.
Pasos de programación	Permite realizar acciones para crear, editar y borrar registros referentes a los pasos de programación de un robot registrados en la base datos.
Consulta de pasos de armado	Permite realizar la consulta de un registro correspondiente a un paso de armado de un robot específico.
Consulta de pasos de programación	Permite realizar la consulta de un registro correspondiente a un paso de programación de un robot específico.

Fuente: *Elaboración propia*

T2. Desarrollar RESTful API

El API fue desarrollado usando el lenguaje de programación *PHP*, se realizaron configuraciones para la conexión con la base de datos y posteriormente la codificación de las funciones detalladas en la tarea anterior. Este desarrollo tuvo como objetivo principal codificar estas funciones para en sprints posteriores se empleadas durante el desarrollo del asistente.

4.2.3.3 Revisión del Sprint 2

Como se puede apreciar en la Tabla 4.15, se verifica el cumplimiento de las actividades planificadas para el Sprint 2.

Tabla 4.15
Verificación de tareas completadas en el Sprint 2

T#	Tareas planificadas	Estado
T1	Detallar funciones de RESTful API	Completado
T2	Desarrollar RESTful API de comunicación	Completado

Fuente: Elaboración propia

En la Tabla 4.16 se muestra el resumen de las tareas completadas y su tiempo de desarrollo, donde se refleja que se tomó cinco horas adicionales para completar las tareas planificadas en la Tabla 4.13, especialmente la tarea de desarrollo de las funciones.

Tabla 4.16
Lista de tareas completadas en el Sprint 2

	Tareas completadas (T#)	Total
Día 1	T1. Detallar funciones de RESTful API	2h
Día 2	T1. Detallar funciones de RESTful API	3h
Día 3	T1. Detallar funciones de RESTful API	2h

Tareas completadas (T#)		Total
Día 4	T1. Detallar funciones de RESTful API	1h
Día 5	T2. Desarrollar RESTful API	4h
Día 6	T2. Desarrollar RESTful API	2h
Día 7	T2. Desarrollar RESTful API	2h
Día 8	T2. Desarrollar RESTful API	3h
Día 9	T2. Desarrollar RESTful API	4h
Día 10	T2. Desarrollar RESTful API	2h
Total		25h

Fuente: Elaboración propia

Al completar las tareas en este sprint el asistente cuenta con una interfaz que le permitirá consumir los datos y recursos alojados en un servidor web y que se visualizarán cuando estén completadas las interfaces de usuario.

4.2.4 Sprint 3

Las tareas de este sprint tienen como objetivo liberar el producto con las interfaces de usuario diseñadas dentro del entorno de desarrollo. Se crean esquemas y bocetos para posteriormente definir los componentes que se emplearán para en el próximo sprint efectuar la programación de estos.

4.2.4.1 Planificación del Sprint 3

Este sprint se planificó considerando tareas enfocadas al diseño de las interfaces de usuario que contaría el asistente. En la Tabla 4.17 se detallan las tareas y su esfuerzo.

Tabla 4.17
Tareas del Sprint 3 y su esfuerzo

T#	Tareas planificadas	Esfuerzo
T1	Diseñar bocetos	8h
T2	Establecer la comunicación de actividades	4h
T3	Diseñar las actividades en el entorno de programación	30h
Total		42h

Fuente: Elaboración propia

Con base en las tareas definidas anteriormente para este sprint, se elaboró una planificación del trabajo en horas necesarias para completarlas, lo cual se puede observar en la Tabla 4.18.

Tabla 4.18
Planificación de tareas del Sprint 3

	T1	T2	T3	Total
Día 1	4h	-	-	4h
Día 2	3h	-	-	3h
Día 3	1h	-	-	1h
Día 4	-	2h	-	2h
Día 5	-	1h	-	1h
Día 6	-	1h	-	1h
Día 7	-	-	2h	2h
Día 8	-	-	2h	2h
Día 9	-	-	2h	2h
Día 10	-	-	2h	2h
Día 11	-	-	2h	2h
Día 12	-	-	2h	2h

	T1	T2	T3	Total
Día 13	-	-	2h	2h
Día 14	-	-	2h	2h
Día 15	-	-	2h	2h
Día 16	-	-	2h	2h
Día 17	-	-	2h	2h
Día 18	-	-	2h	2h
Día 19	-	-	2h	2h
Día 20	-	-	2h	2h
Día 21	-	-	2h	2h
Total				42h

Fuente: Elaboración propia

4.2.4.2 Desarrollo del Sprint 3

T1. Diseñar bocetos

Para empezar a desarrollar las interfaces mediante las cuales el usuario podrá acceder y visualizar la información que desea consultar se hicieron bocetos primero en papel y posteriormente se digitalizaron empleando *Adobe Photoshop*, como se puede apreciar en la Figura 4.20.

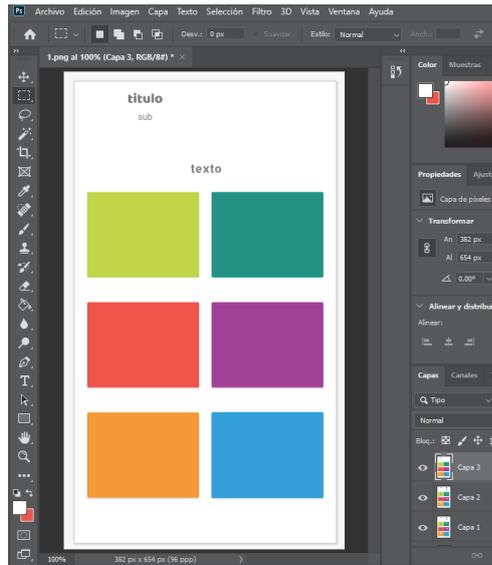


Figura 4.20 Creación de bocetos para el asistente
Fuente: Elaboración propia

De esta manera se va definiendo una idea de cómo va a quedar estructurada cada interfaz, los componentes necesarios para posteriormente desarrollarlas en el entorno de programación y las funcionalidades que incluiría.

A estas interfaces de usuario se las conoce como actividades dentro del entorno de desarrollo para Android y representan una unidad de interacción con el usuario.

T2. Establecer la comunicación de actividades

Concluido el diseño de las actividades que usará el asistente, se empleó una herramienta web para poder establecer el flujo de comunicación entre estas actividades. Un ejemplo del desarrollo de esta tarea se muestra en la Figura 4.21 en donde tenemos las actividades diseñadas y permiten establecer la manera en que se comunican estas y definir los eventos a tener en cuenta para iniciar o concluir una actividad.

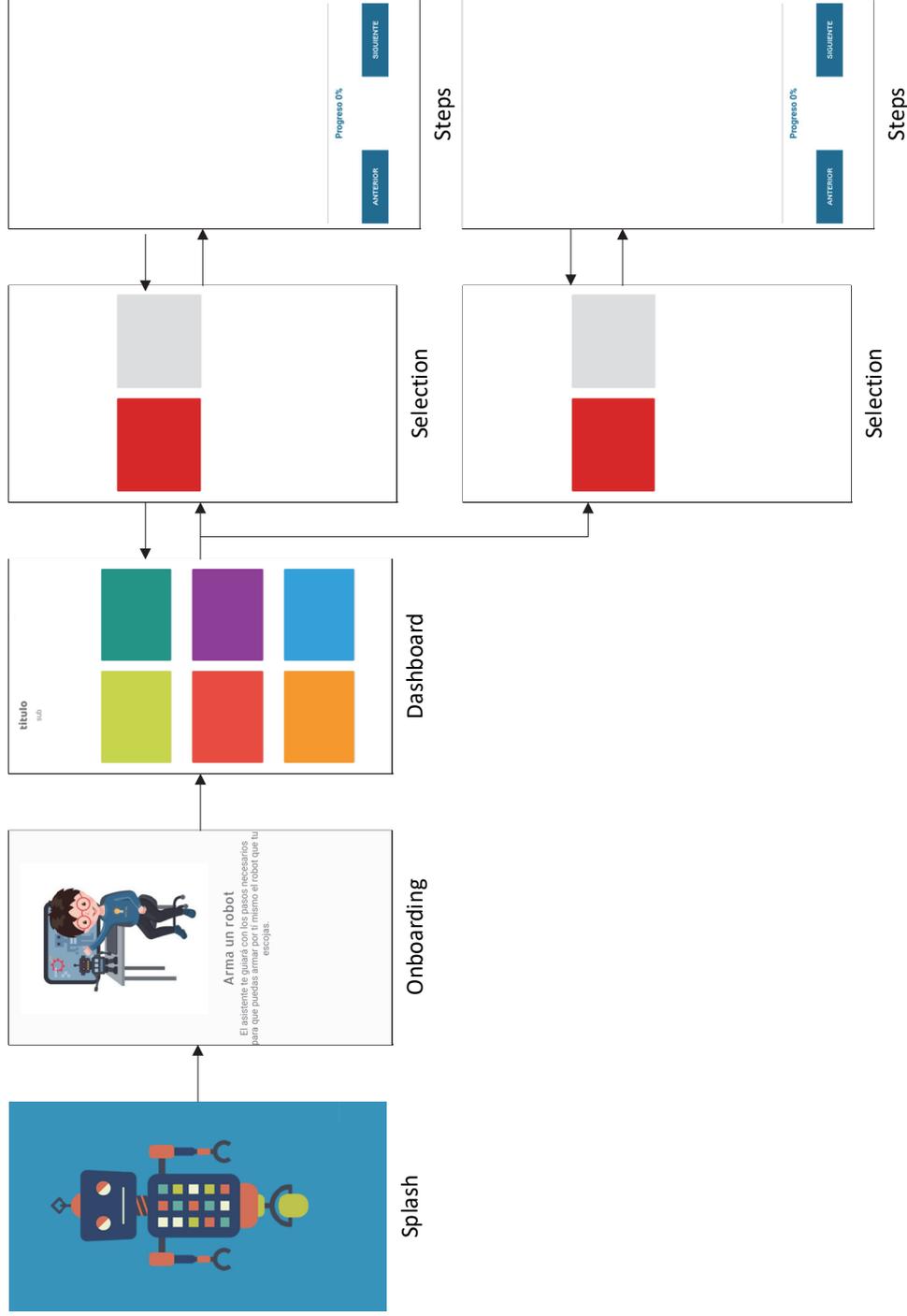


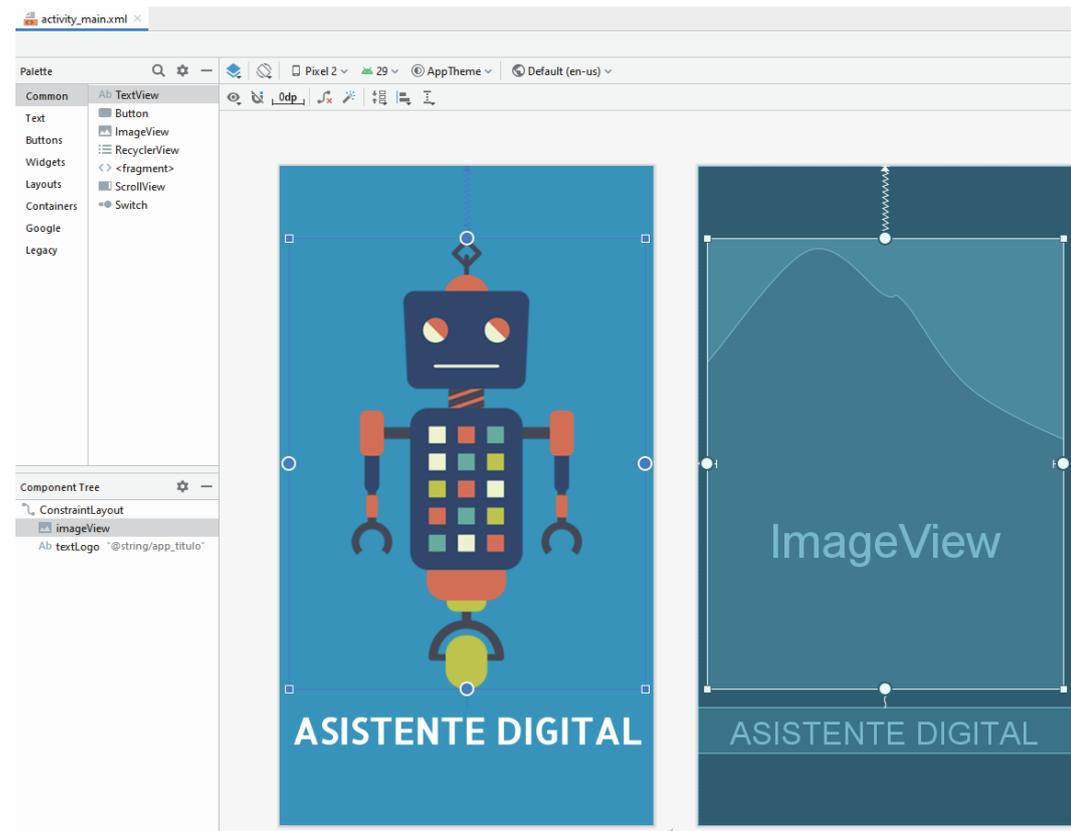
Figura 4.21 Flujo de comunicación de actividades del asistente a desarrollar
Fuente: Elaboración propia

Esta tarea es de importancia porque da pautas al momento de realizar la codificación de cada actividad en cuanto a datos necesarios para realizar consultas, definir métodos para escuchar y controlar eventos, inicio o finalización de otras actividades y brindarle al usuario una secuencia ordenada para que pueda llegar hasta la información que ofrece el asistente.

T3. Desarrollar las actividades

Definido el diseño de las actividades y su comunicación, se empleó Android Studio para empezar al desarrollar el asistente empezando por el desarrollo y diseño de las actividades.

Con base en las tareas a los diseños de seleccionó los objetos correspondientes dentro del entorno de desarrollo, se definieron sus identificadores, valores generales de texto, medidas y estilos, entre otras acciones. En la Figura 4.22 se puede apreciar el diseño de una de las actividades correspondiente a la actividad de inicio del asistente, que también se la puede denominar *splash screen*.



*Figura 4.22 Diseño de la actividad splash screen en Android Studio
Fuente: Elaboración propia*

Esta tarea tuvo una larga duración ya que se concluyó con la mayoría de actividades diseñadas ya en Android Studio, de esta forma en los siguientes sprints se enfocarían en tareas relacionadas más a la codificación del asistente.

4.2.4.3 Revisión del Sprint 3

Como se puede apreciar en la Tabla 4.19, se verifica el cumplimiento de las actividades planificadas para el Sprint 3.

Tabla 4.19
Verificación de tareas completadas en el Sprint 3

T#	Tareas planificadas	Estado
T1	Diseñar bocetos	Completado
T2	Establecer la comunicación de actividades	Completado
T3	Diseñar las actividades en el entorno de programación	Completado

Fuente: Elaboración propia

En la Tabla 4.20 se muestra el resumen de las tareas completadas y su tiempo de desarrollo, se puede apreciar para completar las tareas planificadas se requirió cuatro horas adicionales de acuerdo con lo planificado en la Tabla 4.18, especialmente la tarea de diseño de las actividades.

Tabla 4.20
Lista de tareas completadas en el Sprint 3

	Tareas completadas (T#)	Total
Día 1	T1. Diseñar bocetos	3h
Día 2	T1. Diseñar bocetos	3h
Día 3	T1. Diseñar bocetos	2h
Día 4	T2. Establecer la comunicación de actividades	2h
Día 5	T2. Establecer la comunicación de actividades	1h
Día 6	T2. Establecer la comunicación de actividades	1h
Día 7	T3. Diseñar las actividades en el entorno de programación	3h
Día 8	T3. Diseñar las actividades en el entorno de programación	2h
Día 9	T3. Diseñar las actividades en el entorno de programación	2h
Día 10	T3. Diseñar las actividades en el entorno de programación	2h
Día 11	T3. Diseñar las actividades en el entorno de programación	2h
Día 12	T3. Diseñar las actividades en el entorno de programación	2h

Tareas completadas (T#)		Total
Día 13	T3. Diseñar las actividades en el entorno de programación	2h
Día 14	T3. Diseñar las actividades en el entorno de programación	3h
Día 15	T3. Diseñar las actividades en el entorno de programación	2h
Día 16	T3. Diseñar las actividades en el entorno de programación	2h
Día 17	T3. Diseñar las actividades en el entorno de programación	2h
Día 18	T3. Diseñar las actividades en el entorno de programación	2h
Día 19	T3. Diseñar las actividades en el entorno de programación	2h
Día 20	T3. Diseñar las actividades en el entorno de programación	3h
Día 21	T3. Diseñar las actividades en el entorno de programación	3h
Total		46h

Fuente: Elaboración propia

Completadas las tareas planificadas en este sprint, el asistente ya cuenta con las actividades diseñadas en el entorno de programación Android Studio. En el siguiente sprint se procederá a codificar las funcionalidades que incluirá el asistente.

4.2.5 Sprint 4

Completadas las tareas de este sprint se espera entregar un producto que incluya las principales funcionalidades implementadas y probadas. Para estas tareas se empleará el entorno de desarrollo Android Studio.

4.2.5.1 Planificación del Sprint 4

Este sprint se planificó considerando tareas a la codificación de actividades más importantes que empleará el asistente. En la Tabla 4.21 se detallan estas tareas y su esfuerzo.

Tabla 4.21
Tareas del Sprint 4 y su esfuerzo

T#	Tareas planificadas	Esfuerzo
T1	Ingresar al asistente	8h
T2	Seleccionar robots	8h
T3	Seleccionar robot por versión y tipo de paso	4h
T4	Visualizar instrucciones de armado y programación	12h
Total		32h

Fuente: Elaboración propia

Con base en a las tareas definidas anteriormente para este sprint, se elaboró una planificación del trabajo en horas necesarias para completarlas, como se detalla en la Tabla 4.22.

Tabla 4.22
Planificación de tareas del Sprint 4

	T1	T2	T3	T4	Total
Día 1	2h	-	-	-	2h
Día 2	3h	-	-	-	3h
Día 3	2h	-	-	-	2h
Día 4	1h	-	-	-	1h
Día 5	-	2h	-	-	2h
Día 6	-	2h	-	-	2h
Día 7	-	2h	-	-	2h
Día 8	-	2h	-	-	2h
Día 9	-	-	2h	-	2h
Día 10	-	-	1h	-	1h

	T1	T2	T3	T4	Total
Día 11	-	-	1h	-	1h
Día 12	-	-	-	2h	2h
Día 13	-	-	-	1h	1h
Día 14	-	-	-	1h	1h
Día 15	-	-	-	1h	1h
Día 16	-	-	-	1h	1h
Día 17	-	-	-	1h	1h
Día 18	-	-	-	1h	1h
Día 19	-	-	-	1h	1h
Día 20	-	-	-	1h	1h
Día 21	-	-	-	1h	1h
Día 22	-	-	-	1h	1h
Total					32h

Fuente: Elaboración propia

4.2.5.2 Desarrollo del Sprint 4

La codificación del asistente se realizó en *Android Studio* empleando el lenguaje de programación Java y las clases que proporciona Android SDK.

T1. Ingresar al asistente

En esta tarea se programó una actividad con la que se inicia el asistente y despliega una *splash screen* para el usuario y posteriormente una pequeña guía mediante un *onboarding screen*. En su codificación contiene un módulo de verificación de acceso a internet. En la Figura 4.23 se puede observar el diagrama de casos de uso entre el usuario y el asistente.

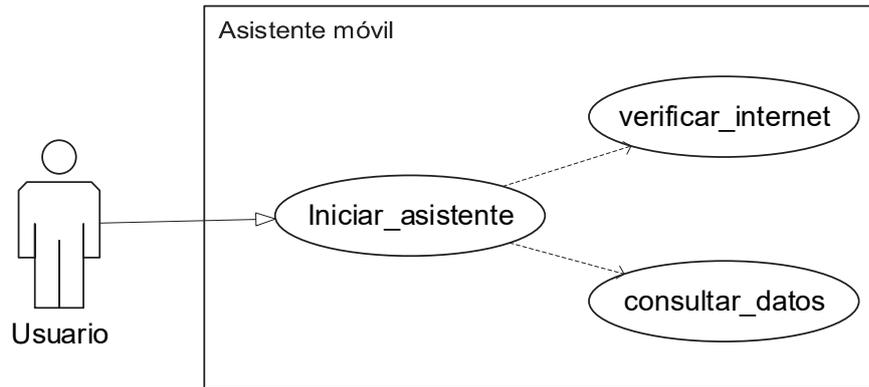


Figura 4.23 Diagrama de casos de uso para la actividad de ingreso
Fuente: Elaboración propia

En la Figura 4.24 se puede observar el diseño final de la actividad de *splash screen* y *onboarding*. En la actividad de onboarding se muestra información adicional al usuario en forma de pequeñas guías de cómo puede aprovechar el asistente, así como su finalidad.

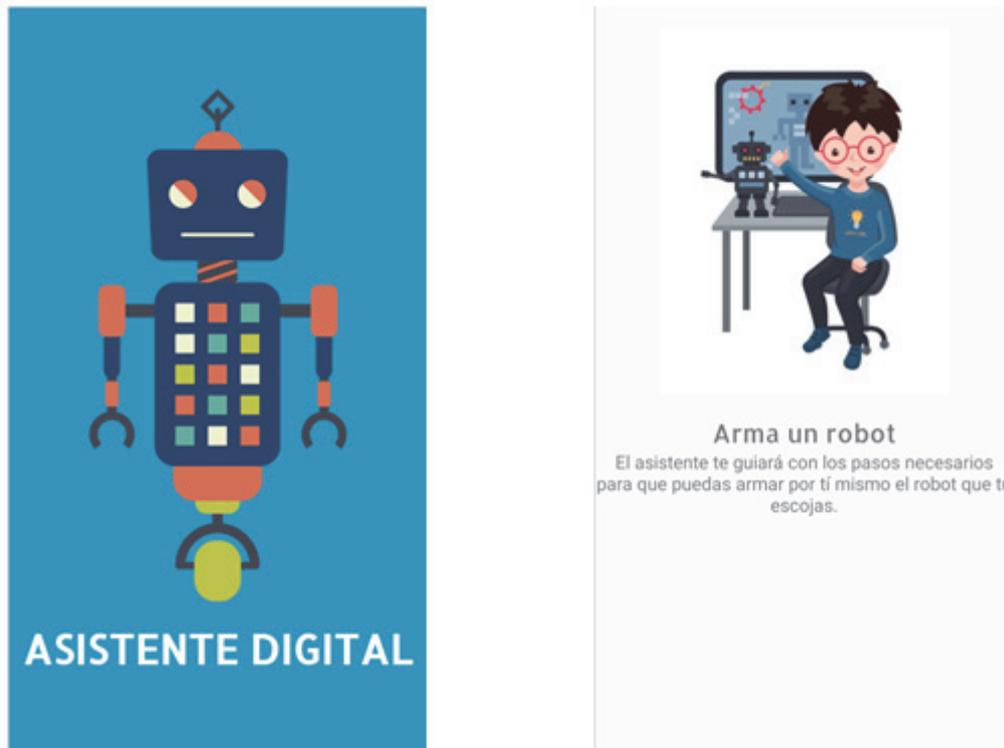


Figura 4.24 Diseño de la actividad splash screen y onboarding
Fuente: Elaboración propia

T2. Seleccionar robots

En esta tarea se codificó una actividad que servirá como *dashboard* para en la que se el usuario podrá escoger una de las opciones que se le mostrará en la pantalla del smartphone. Cuenta con un módulo de consulta de datos y verificación de internet. Se puede apreciar el diagrama de casos de uso en la Figura 4.25 entre el usuario y el asistente.

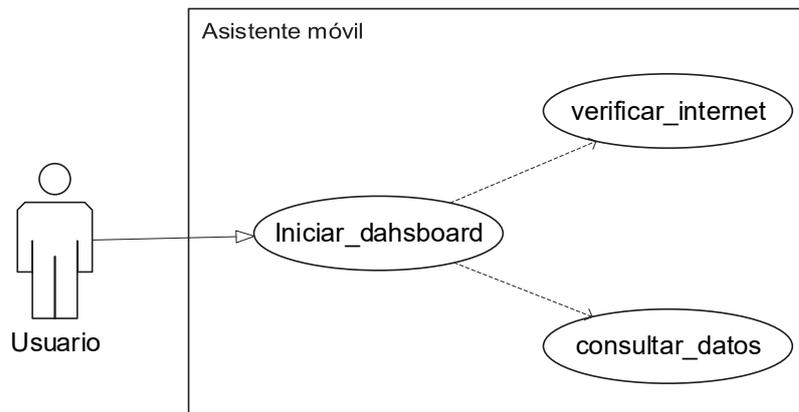


Figura 4.25 Diagrama de casos de uso para el dashboard del asistente
Fuente: Elaboración propia

En la Figura 4.26 se muestra el *dashboard* del asistente. El usuario puede observar las categorías de robots y seleccionar la que esté interesado en armar.



Figura 4.26 Diseño de la actividad splash screen y onboarding
Fuente: Elaboración propia

También se muestra una opción de ayuda en la que se puede acceder a una pequeña guía en la que se detalla el uso de la aplicación, como se puede apreciar en la Figura 4.27.



Figura 4.27 Opción de ayuda ofrecida por el asistente
Fuente: Elaboración propia

T3. Seleccionar robot por versión y tipo de paso

En esta tarea se codifica una actividad que permite la selección del robot a armar de acuerdo al tipo de versión de *LEGO Mindstorms* con la que se va a trabajar y también el tipo de paso que se desea visualizar, siendo estos, pasos de armado o pasos de programación. En la Figura 4.28 se puede apreciar el diagrama de casos de uso entre el usuario y el asistente.

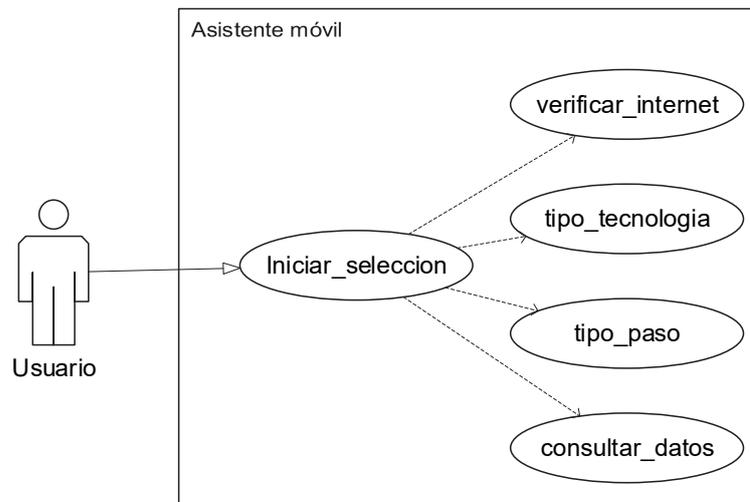


Figura 4.28 Diagrama de casos de uso para la selección versión y tipo de paso
Fuente: Elaboración propia

En esta actividad se verifica el acceso a internet para permitir al usuario que escoja, a partir de la categoría de robot seleccionado anteriormente, las opciones de versión orientadas a *EV3* o *NXT*, como se puede apreciar en la Figura 4.29.



Figura 4.29 Diseño de la actividad de selección de versión de LEGO Mindstorms
Fuente: Elaboración propia

También se emplea esta actividad, como se puede apreciar en la Figura 4.30, para que el usuario pueda optar por visualizar los pasos con los que va a guiarse para armar paso a paso el robot o visualizar los pasos con los que puede programarlo empleado el software de *LEGO*.



Figura 4.30 Diseño de la actividad de selección de tipo de paso
Fuente: Elaboración propia

T4. Visualizar instrucciones de armado y programación

En esta actividad se realiza una verificación del acceso a una red de datos, así como también, la codificación necesaria para cargar los datos

que proporciona la base de datos por medio de la RESTful API. En la Figura 4.31 se muestra el diagrama de casos de uso entre el usuario y el asistente

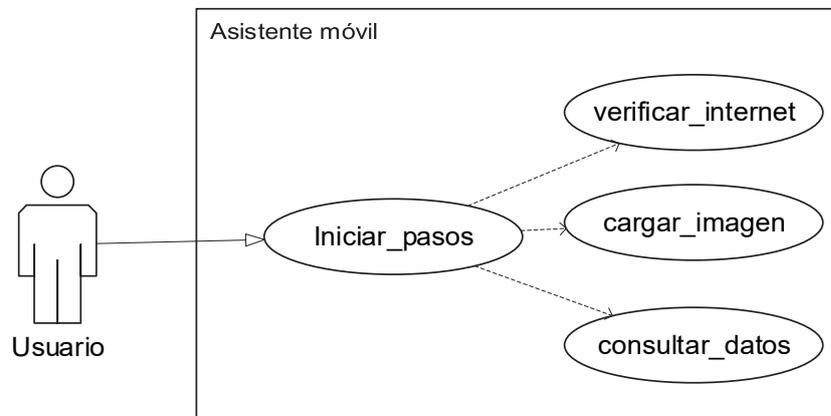


Figura 4.31 Diagrama de casos de uso la actividad de visualización de pasos de armado y programación

Fuente: Elaboración propia

En la Figura 4.32 se muestra el diseño de esta actividad. El usuario puede apreciar la imagen del paso que está realizando, una breve descripción de cómo debería de armar esa sección, las piezas que necesita y una pequeña barrar de progreso para que pueda conocer el avance en su tarea de armado o programación.

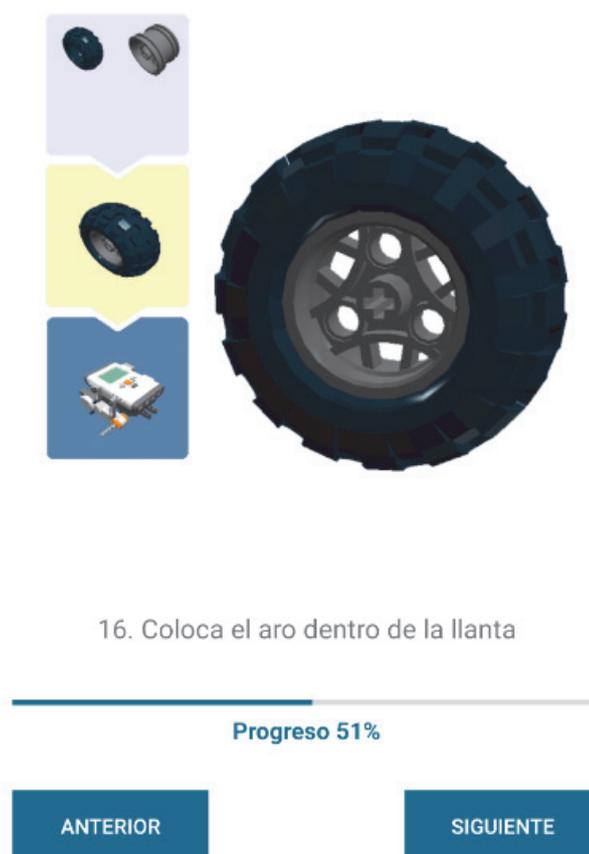


Figura 4.32 Diseño de la actividad de visualización de pasos de armado y programación
Fuente: Elaboración propia

4.2.5.3 Revisión del Sprint 4

Las tareas de este sprint estaban enfocadas a la codificación de funcionalidades del asistente. Como se puede apreciar en la Tabla 4.23, se verifica el cumplimiento de las actividades planificadas para el Sprint 4.

Tabla 4.23
Verificación de tareas completadas en el Sprint 4

T#	Tareas planificadas	Estado
T1	Ingresar al asistente	Completado
T2	Seleccionar robots	Completado
T3	Seleccionar robot por versión y tipo de paso	Completado
T4	Visualizar instrucciones de armado y programación	Completado

Fuente: Elaboración propia

En la Tabla 4.24 se muestra el resumen de las tareas completadas y su tiempo de desarrollo, se puede apreciar para completar las tareas planificadas se requirió nueve horas adicionales de acuerdo con lo planificado en la Tabla 4.22, especialmente la tarea de diseño de las actividades.

*Tabla 4.24
Lista de tareas completadas en el Sprint 4*

Tareas completadas (T#)		Total
Día 1	T1. Ingresar al asistente	2h
Día 2	T1. Ingresar al asistente	3h
Día 3	T1. Ingresar al asistente	2h
Día 4	T1. Ingresar al asistente	1h
Día 5	T2. Seleccionar robots	2h
Día 6	T2. Seleccionar robots	2h
Día 7	T2. Seleccionar robots	2h
Día 8	T2. Seleccionar robots	2h
Día 9	T3. Seleccionar robot por versión y tipo de paso	2h
Día 10	T3. Seleccionar robot por versión y tipo de paso	1h
Día 11	T3. Seleccionar robot por versión y tipo de paso	2h
Día 12	T3. Seleccionar robot por versión y tipo de paso	2h
Día 13	T4. Visualizar instrucciones de armado y programación	2h
Día 14	T4. Visualizar instrucciones de armado y programación	2h
Día 15	T4. Visualizar instrucciones de armado y programación	2h
Día 16	T4. Visualizar instrucciones de armado y programación	2h
Día 17	T4. Visualizar instrucciones de armado y programación	2h
Día 18	T4. Visualizar instrucciones de armado y programación	1h
Día 19	T4. Visualizar instrucciones de armado y programación	1h
Día 20	T4. Visualizar instrucciones de armado y programación	2h

Tareas completadas (T#)		Total
Día 21	T4. Visualizar instrucciones de armado y programación	1h
Día 22	T4. Visualizar instrucciones de armado y programación	1h
Día 23	T4. Visualizar instrucciones de armado y programación	2h
Total		41h

Fuente: *Elaboración propia*

Completadas las tareas planificadas en este sprint, el asistente ya cuenta con las principales actividades codificadas, así como demás funcionalidades, todo en el entorno de programación Android Studio. En la siguiente sección se revisarán las pruebas de desarrollo realizadas al asistente.

4.2.6 Pruebas del desarrollo

Durante el desarrollo del asistente, para garantizar la correcta ejecución de sus funcionalidades se realizaron pruebas para cada módulo y previo a la finalización de cada sprint. Se definieron pruebas dependiendo de los requerimientos a verificar y el tipo de tareas a desarrollar en el sprint.

4.2.6.1 Definición de pruebas

Para las pruebas funcionales se emplearon las herramientas que ofrece el propio entorno, como AVD (*Android Virtual Device*) y también un smartphone *Razer Phone 2* también, en conjunto con el Android Studio. Esto se puede apreciar en la Figura 4.33.

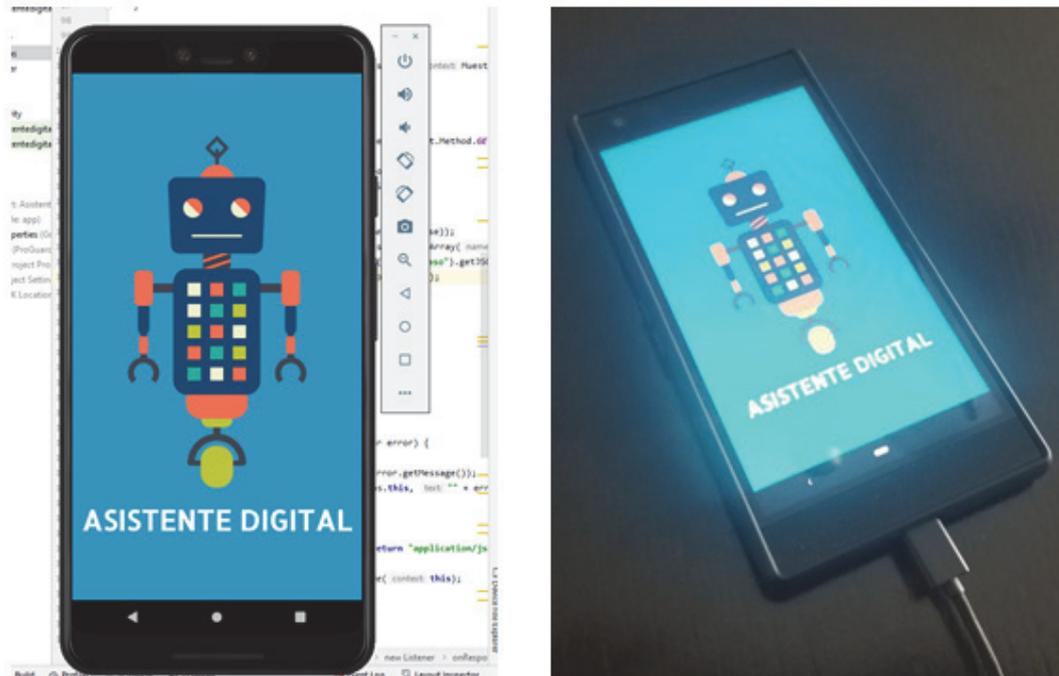


Figura 4.33 Verificación de funcionamiento empleando Android Virtual Device y un smartphone
Fuente: Elaboración propia

La mayor parte de las pruebas funcionales fueron realizadas en el cuarto sprint, puesto que, en las tareas de este sprint están enfocadas a la codificación. Estas pruebas fueron realizadas por el programador.

Sprint 4. Pruebas funcionales

En el primer escenario de prueba se evalúa el proceso de inicio del asistente instalado en el smartphone. Se verifica si existe acceso a internet. En caso de no contar con este acceso se mostrará un *toast* indicando la falta de este requisito y el asistente se cerrará.

En el caso de que esta verificación es exitosa se procede a cargar la información de la *onboarding screen*. En la Figura 4.34 se muestra su flujo.

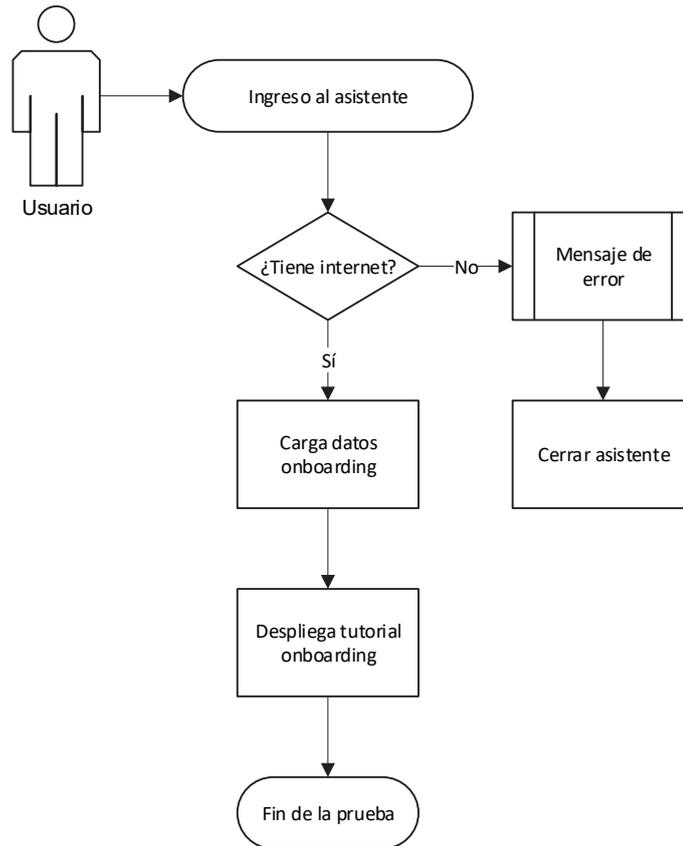


Figura 4.34 Escenario de prueba de ingreso al asistente
Fuente: Elaboración propia

En la Figura 4.35 se puede apreciar el flujo para el segundo escenario de prueba que permite verificar la correcta visualización del *dashboard* del asistente. Se realiza una verificación de la conexión a internet, y en caso de contar con este acceso, se cargan los datos mediante los cuales el usuario podrá seleccionar la categoría de robot que está buscando. Se controlan los eventos *onClick* en la actividad y en el caso de realizar una selección, esta se guarda para su posterior uso otra actividad.

En el caso de no contar con acceso a internet, se le desplegará un mensaje informando la ausencia de esta conectividad y se cerrará el asistente.

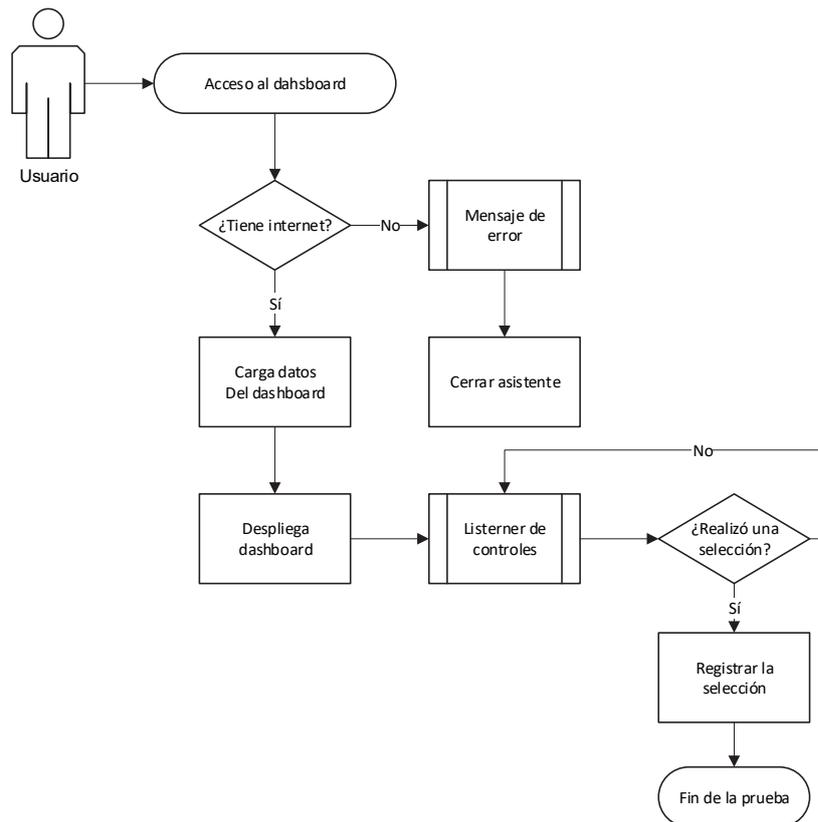


Figura 4.35 Escenario de prueba de selección de categoría de robot
Fuente: Elaboración propia

Otro escenario de pruebas se puede observar en la Figura 4.36 en el que se verifica el funcionamiento para que le usuario pueda indicar la versión y tipo de paso a visualizar. Como primer paso se verifica la conexión a internet, en el caso de ser afirmativa esta verificación, se procede verificar el registro de la categoría de robot seleccionada anteriormente. Se controlan los eventos *onClick* en la actividad para registrar si desea visualizar las versiones de *LEGO* o si va a visualizar pasos de armado. Realizada esta segunda verificación se cargan los datos para que el usuario pueda escoger una opción.

En el caso de no contar con una conexión a internet, se muestra un mensaje de la ausencia de esta conexión y se cierra el asistente.

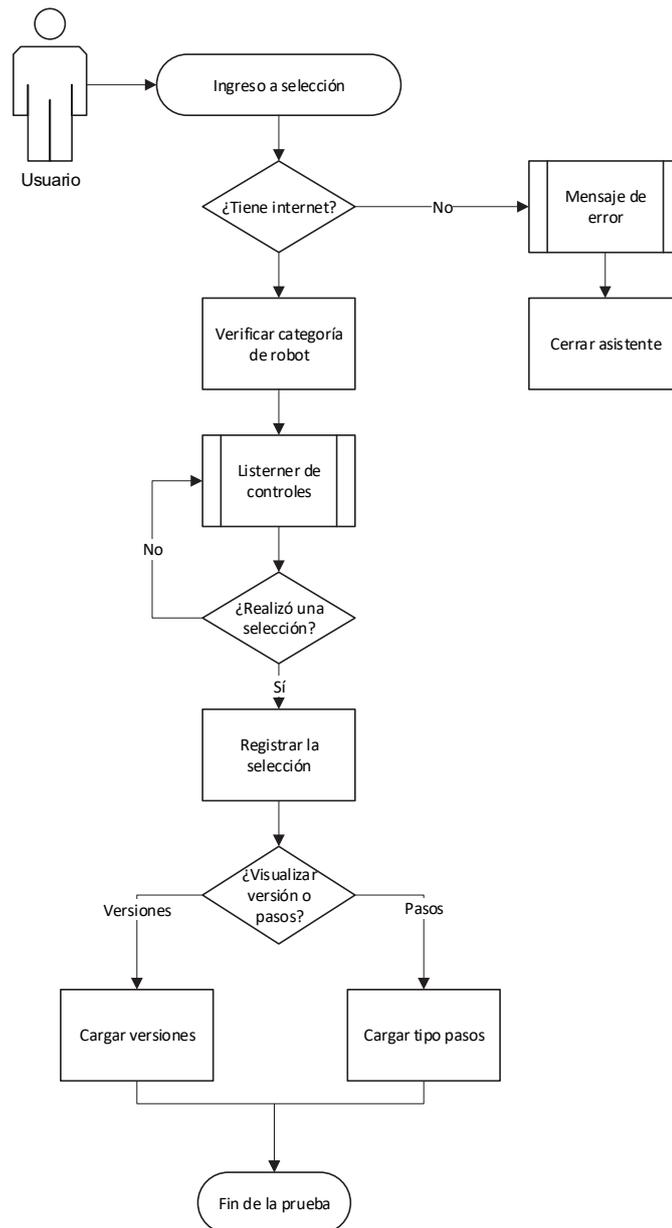


Figura 4.36 Escenario de prueba de selección de versión y tipo de paso
Fuente: Elaboración propia

En la Figura 4.37 se puede apreciar el flujo para el escenario de prueba en el que se busca comprobar el correcto funcionamiento de visualización de instrucciones de armado y programación. Es importante realizar una verificación de la existencia de conexión a internet, puesto que, en caso de contar con esta conectividad, se verificarán las selecciones realizadas en etapas anteriores para realizar una consulta que permitirá desplegar un paso de armado o programación en la actividad que se le

presenta al usuario. Para ambos casos se cargará la imagen correspondiente al número de paso solicitado y en caso no cargar la imagen tendrá una por *default*.

En el caso de no contar con conectividad a internet, se cerrará el asistente mostrando un mensaje mediante un *toast* para indicar la falta de este requerimiento.

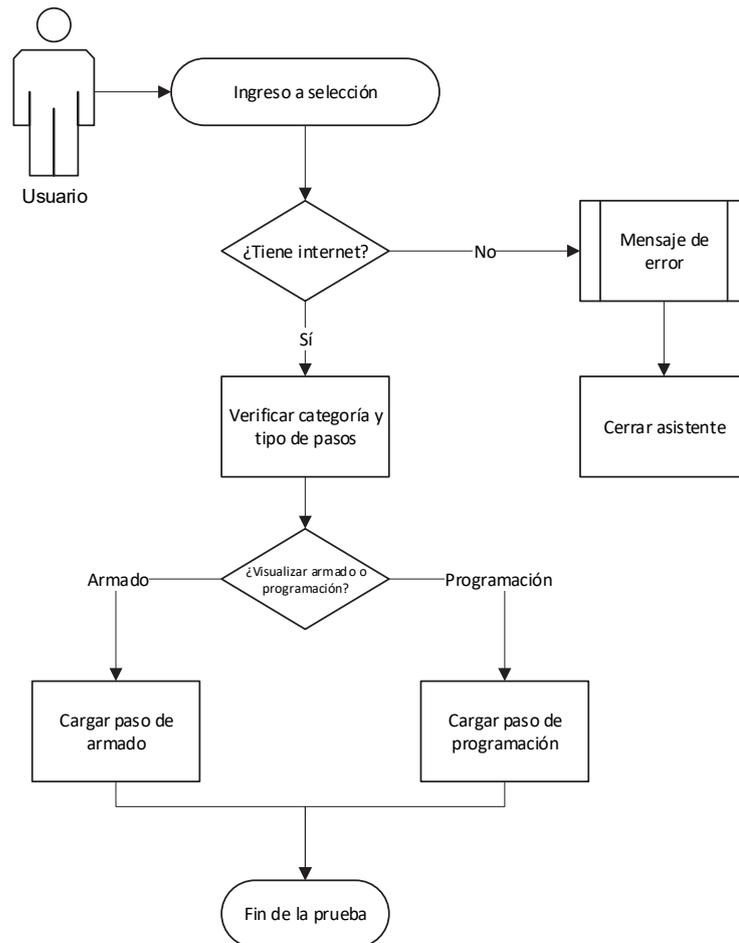


Figura 4.37 Escenario de prueba de visualización de instrucciones de armado
Fuente: Elaboración propia

4.2.6.2 Resumen de pruebas

Las pruebas realizadas al asistente mostraron que es necesario antes de realizar una consulta a la base de datos almacenada en el servidor verificar que el smartphone cuente con acceso a internet.

De esta forma se pudo verificar que se puede realizar un apropiado acceso al iniciar el asistente y cargar la *onboarding screen* apropiadamente. Se corrigieron unas pequeñas falencias en el diseño esta actividad para que sea más legible la información que se despliega en esta actividad.

También se pudo comprobar que se carguen apropiadamente las categorías de robots que se habían definido en apartados anteriores para que el usuario pudiese seleccionar una de estas para empezar su labor. Durante la ejecución de esta prueba se actualizó el diseño para que se muestre una pequeña imagen que haga referencia a la categoría de robot y mejorar la distribución de los botones.

Al realizar las pruebas de selección por versión de robot y tipo de paso también se decidió actualizar el diseño y como en la actividad anterior incluir una imagen que distinta cada opción.

En las pruebas realizadas para la visualización de instrucciones de armado y programación se actualizó el diseño de la barra de progreso de manera que sea más legible y pudiese estar en una ubicación más apropiada dentro de la actividad.

Adicionalmente, se ajustó la propiedad de orientación de la aplicación para que se ejecutara en modo *portrait*, se definió una imagen por default en caso de no cargar apropiadamente la imagen correspondiente a un paso y se verificó el estilo y colores de algunos componentes en las actividades.

4.3 Etapa 3: Ejecución de la experimentación

La ejecución de la experimentación se basó en la utilización del asistente móvil de un laboratorio digital de robótica de competencias estudiantiles como una herramienta de experimentación, en un estudiante de secundaria de la ciudad de Guayaquil. Las características de este estudiante se detallan en la Tabla 4.25.

Tabla 4.25
Características del usuario

Características	Datos
Edad	11 años
Institución en la que estudia	Unidad Educativa Lemas
Nivel de estudios	Cursando la secundaria
¿Cuenta con experiencia en robótica educativa?	No

Fuente: Elaboración propia

Se trató de conseguir que el estudiante pruebe las funcionalidades del asistente en un ambiente apropiado debido a la pandemia sanitaria mundial y considerando la suspensión de clases en modalidad presencial en todo el territorio nacional.

El asistente ya desarrollado se instaló en un smartphone *Razer Phone 2*, cuyas principales características se muestran en la Tabla 4.26.

Tabla 4.26
Características del smartphone empleado en la experimentación

Características	Datos
Modelo	Razer Phone 2
Dimensiones	158,5 x 79 x 8,5 milímetros
Resolución	2.560 x 1.440 píxeles
Procesador	Qualcomm Snapdragon 845
Conectividad	LTE, Wi-Fi 802.11ac con MIMO 2x2, Bluetooth 5.0, NFC

Fuente: Elaboración propia basado en (GSM Arena, 2020)

Para el armado de los robots se emplearon los sets *LEGO Mindstorms Education EV3* que se muestra en la Figura 4.38 y *LEGO Mindstorms NXT 2.0* mostrado en la Figura 4.39 que estuvieron disponibles para su uso.



Figura 4.38 Set LEGO Mindstorms Education EV3 empleado para la experimentación
Fuente: Elaboración propia



Figura 4.39 Set LEGO Mindstorms NXT 2.0 empleado para la experimentación
Fuente: Elaboración propia

4.3.1 Instrucciones de uso

En esta fase se explicó al estudiante sobre el uso y funcionalidades del asistente. Al no contar con conocimientos previos de robótica puesto que no forma parte de sus estudios a nivel secundario, también se dieron

indicaciones sobre los componentes y uso de los kits *LEGO Mindstorms* como se puede apreciar en la Figura 4.40.



Figura 4.40 Indicaciones para el uso del asistente y los kits LEGO Mindstorms
Fuente: Elaboración propia

4.3.2 Uso del asistente

Impartidas las instrucciones de uso, el estudiante con un poco de ayuda del administrador del asistente móvil procedió a armar un robot de acuerdo con su interés, seleccionado un robot de la categoría *sumobot* con el kit *LEGO Mindstorms Education EV3* como se puede apreciar en la Figura 4.41.



Figura 4.41 Armado de un robot de sumo empleando al asistente móvil desarrollado y el kit EV3

Fuente: Elaboración propia

El estudiante fue capaz de armar y programar el robot sumo siguiendo las indicaciones del asistente móvil en un tiempo de 2H40. La ayuda del administrador del asistente móvil se basó en orientar en la búsqueda de ciertas piezas como se puede observar en la Figura 4.42.



*Figura 4.42 Asistencia para la búsqueda de piezas para el armado de un robot de sumo empleando el kit EV3
Fuente: Elaboración propia*

También se dio asistencia para el armado de piezas con las que el estudiante no estaba familiarizado en este tipo de kits de robótica como se muestra en la Figura 4.43.

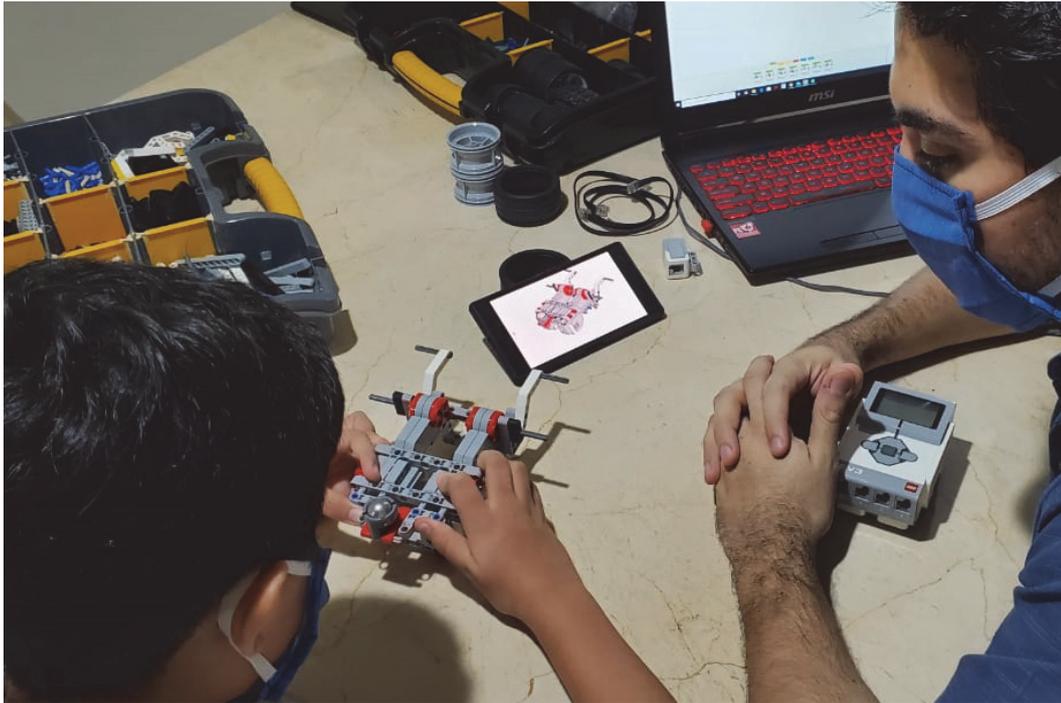


*Figura 4.43 Asistencia para la identificación de piezas del kit
Fuente: Elaboración propia*

Al culminar con este armado se dieron ciertas recomendaciones para que continúe con el armado de otros robots y continuase explorando los contenidos del asistente.

4.3.3 Ejecución de la evaluación

Para la evaluación, se ejecutó empleando el asistente móvil para que el estudiante de secundaria arme un robot de la categoría seguidor de línea empleando el kit *LEGO Mindstorms Education EV3* con muy poca intervención del administrador del asistente móvil, como se puede observar en la Figura 4.44.



*Figura 4.44 Ejecución de la evaluación empleando el asistente móvil desarrollado
Fuente: Elaboración propia*

El administrador del asistente móvil verificó el armado y programación del robot seguidor de línea cuando el estudiante indicó que había terminado estas tareas. También se dio asistencia en para reconocer ciertas piezas de armado como se puede observar en la Figura 4.45.



*Figura 4.45 Ejecución de la evaluación con un estudiante de secundaria armando de un seguidor de línea
Fuente: Elaboración propia*

Luego de 1H40 el estudiante terminó de armar y programar el robot seguidor de línea, como se muestra en la Figura 4.46.



*Figura 4.46 Ejecución de la evaluación mediante el armado de un seguidor de línea
Fuente: Elaboración propia*

4.4 Etapa 4: Análisis de resultados

El análisis de resultados reflejó que el asistente desarrollado sirve para guiar a un estudiante en la tarea de armado y programación de un robot de competencias.

El estudiante, a pesar de no contar con experiencia previa usando los kits *LEGO Mindstorms*, pero, al estar familiarizado con las tecnologías actuales, como un smartphone y una laptop, pudo armar un robot empleando el asistente desarrollado en un menor tiempo luego de la etapa de introducción a su uso.

Al principio el estudiante al no estar familiarizado con las piezas que confirman el kit ni con tareas de robótica educativa, solicitó ayuda al

programador durante el primer armado de un robot. Posteriormente y luego de la evaluación continuó armado otro robot y explorando por su cuenta los bloques de programación.

La utilización del asistente fue apropiada para que el estudiante se guíe sin necesidad de buscar información en otra fuente o documento. El estudiante indicó que las imágenes mostradas y la descripción de los pasos permitieron realizar el armado de una manera sencilla.

Con las instrucciones apropiadas para el uso del asistente y los kits de *LEGO Mindstorms* se logró reducir el tiempo de armado de un robot a 1H00, considerando que se dio asistencia para el reconocimiento de ciertas piezas que no eran reconocibles para el estudiante de secundaria. Estos resultados se pueden apreciar en la Tabla 4.27.

*Tabla 4.27
Resultados obtenidos*

Característica	Resultado
Tiempo de armado del robot	<ul style="list-style-type: none"> ▪ 2H00 sin experiencia previa en robótica con el asistente desarrollado. ▪ 1H20 empleando el asistente desarrollado.
Tiempo de programación del robot	<ul style="list-style-type: none"> ▪ 40 minutos sin experiencia previa en robótica con el asistente desarrollado. ▪ 20 minutos empleando el asistente desarrollado.

Fuente: Elaboración propia

También comentó dentro de los aspectos a mejorar en el asistente, la posibilidad de agregar más variedad de robots, incluyendo drones. Adicionalmente sugirió que la aplicación pueda usarse en tablets.

CAPÍTULO 5 CONCLUSIONES Y RECOMENDACIONES

En el presente trabajo de titulación se desarrolló un asistente móvil de un laboratorio digital de robótica de competencias estudiantiles, el mismo que fue evaluado por un estudiante comprobando que su uso contribuye favorablemente a reducir el tiempo durante la realización de estas actividades, de la misma manera que facilita una guía práctica y reconocible de los componentes y pasos para armar y programar el robot.

Para cumplir con el primer objetivo planteado se realizó una revisión exhaustiva de la robótica educativa en el Ecuador para determinar su vinculación dentro de las instituciones educativas particulares y públicas. Se verificó que son pocas las instituciones que implementan robótica educativa dentro del aula, participando y organizando torneos. De la misma manera también se evidenció que existen organizaciones interesadas en fomentar la robótica educativa en el país mediante proyectos e inversiones conjuntas con instituciones educativas.

Para lograr el segundo objetivo formulado en el presente trabajo, se empleó se empleó la metodología *Scrum* y sus artefactos para el desarrollo del asistente móvil que guíe a los estudiantes a construir robots para competencias académicas. Se diseñó la arquitectura de la base de datos y la creación de sus estructuras. Se ejecutaron tareas para la creación de una *RESTful API*. Se diseñaron y programaron las actividades que componen el asistente. Finalmente se realizaron pruebas de software para asegurar la correcta ejecución del asistente obteniendo como resultados que estas funcionalidades programadas se ejecutaban apropiadamente tanto en el simulador como el en smartphone.

El tercer objetivo se completó realizando la validación funcional se realizó con un estudiante de secundaria empleando el asistente móvil desarrollado para que pudiese armar y programar un robot de competencias. Luego de dar las indicaciones de uso de los kits *LEGO Mindstorms* y el asistente móvil, el estudiante armó un robot sumo y

posteriormente con una ligera guía del administrador del asistente móvil, logró armar un robot seguidor de línea. Los resultados indicaron que el uso del asistente facilita estas tareas de armado y configuración de robots, reduciendo el tiempo invertido en estas tareas en 1H00, así como también fomenta a que el estudiante desee construir robots de acuerdo con su gusto. El estudiante también indicó que espera en un futuro se agreguen más robots en el asistente, como, por ejemplo, los drones.

Como limitaciones, en la experimentación no se pudo realizar pruebas con varios estudiantes de una o varias instituciones por los motivos de seguridad ante el estado de pandemia mundial, lo que generó cierta resistencia en varios representantes de estudiantes para poder disponer de un tiempo para hacer las pruebas, así como tampoco contribuyó la suspensión de clases en modalidad presencial en las instituciones educativas.

Como recomendaciones para trabajos futuros para otros investigadores, se sugiere el desarrollo de nuevas categorías de robots para su armado y programación, así como también tener en consideración los kits que ofrecen otras empresas con sus tecnologías como *Arduino*, *Tetrix robotics*, *Vex Robotic Competition*, *Makeblock* o *Luxe Tech Robotics*. También añadir un desarrollo para *iOS* de manera que una institución que ya cuente con equipos con este sistema operativo pueda también emplear el asistente.

REFERENCIAS

- Afari, E., & Khine, M. (2017). Robotics as an Educational Tool: Impact of LEGO Mindstorms. *International Journal of Information and Education Technology*.
- Alarcón, I. (2015). *El Comercio*. Obtenido de <https://www.elcomercio.com/tendencias/teebot-robot-tecnologia-educacion-ecuador.html>
- Almomani, I., & Khayer, A. (2019). Android Applications Scanning: The Guide. *2019 International Conference on Computer and Information Sciences (ICCIS)*.
- ARCOTEL. (2018). Robótica en las universidades. *Revista institucional de ARCOTEL*, 19.
- Asociación de Educadores en Robótica del Ecuador. (2019). *Sensor - Asociación de Educadores en Robótica del Ecuador*. Obtenido de <http://sensor.260mb.net/xsensor/page2.html?i=2>
- Azevedo Gomes, C., Gomes, H., Rego, B., Sousa, B., Loureiro, M., & Rocha, P. (2019). Smart City Kids Lab: Creative Computing in Primary School. *2019 International Symposium on Computers in Education (SIIE)*.
- Babativa, A., Briceno, P., Nieto, C., & Salazar, O. (2016). Agile Application Development for Mobile Devices. Case Study: Mobile Taximeter. *Revista Ingeniería*.
- Banachi Barreto, V., L'Erario, A., & Augusto Fabri, J. (2015). Teaching programming for high school students using the Lego Mindstorms robot. *2015 10th Iberian Conference on Information Systems and Technologies (CISTI)*, (págs. 1-7). Aveiro.
- Barbosa, A., Oliveira, A., Silva, A., Vilas Boas, A., Vilas Boas, E., Almeida, F., . . . Vilas Boas, M. (2018). Design and Development of a Manipulator for Educational Robotics. *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*.
- Barrera Lombana, N. (2015). Use of educational robotics as a teaching strategy in the classroom. *Praxis & Saber*, 215-234.
- Benedettelli, D. (2008). *Creating Cool MINDSTORMS NXT Robots*. Apress.
- Bermón Angarita, L., & Guapachá Hernández, J. (2019). Gamified system for learning of Scrum development process. *2019 14th Iberian Conference on Information Systems and Technologies (CISTI)*.

- BWEE . (2018). *BWEE* . Obtenido de <http://ecuador.bweegroup.com/>
- Caballero, Y., & García-Valcárcel, A. (2017). Development of computational thinking skills and collaborative learning in initial education students through educational activities supported by ICT resources and programmable educational robots. *TEEM 2017: Proceedings of the 5th International Conference on Technological Ecosystems for Enhancing Multiculturality*, 1-6.
- Calderón, F., & Flórez, J. (2018). torneos de robótica: una herramienta de educación en ingeniería efectiva para la generación posmilenial. *Asociación Colombiana de Facultades de Ingeniería*.
- Cheng, Y.-W., Sun, P.-C., & Chen, N.-S. (2018). The essential applications of educational robot: Requirement analysis from the perspectives of experts, researchers and instructors. *Computers & Education*, 399-416.
- Chevalier, M., Riedo, F., & Mondada, F. (2016). How Do Teachers Perceive Educational Robots in Formal Education? *IEEE Robotics & Automation Magazine*, 16-23.
- Chuquimarca Abad, M. (2016). *El uso de lego mindstorms ev3 como recurso didáctico basado en la robótica educativa para desarrollar el pensamiento lógico en la asignatura de matemática de noveno año de educación general básica en la escuela Miguel Riofrío nº 2 de la ciudad de Loja*. Loja: Universidad Nacional de Loja .
- Clara Gómez-Álvarez, M., González Palacio, L., Manrique-Losada, B., Bryan, V., & Arbeláez, S. (2019). Experiencias exitosas de enseñanza de programación y robótica en educación básica y media. *Iberian Conference on Information Systems & Technologies*, 1-6.
- COPOL. (2019). *COPOL Metal Challenge Robotics Competition*. Obtenido de <https://www.copol.edu.ec/eventos/copolrobotics.html>
- Costa, S., Machado, J., & Cruz, M. (2017). The influence of WOM and Peer Interaction in the Decision-Making Process of Generation Z within the family. *International Journal of Marketing, Communication and New Media*.
- Craig, C., & Gerber, A. (2015). *Learn Android Studio*. Apress.
- Dawodi, M., Hadi Hedayati, M., Ahmad Baktash, J., & Latif Erfan, A. (2019). Facebook MySQL Performance vs MySQL Performance. (págs. 0103-0109). Vancouver: IEEE.
- Dimes, T. (2016). *PHP*.

- Fernández, E., & Pastor-Mendoza, J. (2018). Synergy LEGO Mindstorms – Arduino: Taking advantage of both platforms. *2018 XIII Technologies Applied to Electronics Teaching Conference (TAE)*.
- Francesca, A., Orso, V., Pluchino, P., & Menegatti, E. (2018). How Robots Impact Students' Beliefs about Their Learning Skills. *Companion of the 2018 ACM/IEEE International Conference*, 47-48.
- Fundación Telefónica. (19 de Enero de 2018). *Fundación Telefónica*. Obtenido de <https://fundaciontelefonica.com.ec/2018/01/19/la-tecnologia-o-la-autoridad-que-crea-la-brecha-entre-alumnos-y-docentes/>
- García, A., & Caballero, Y.-A. (2019). Robotics to develop computational thinking in early Childhood. *Revista Científica de Educomunicación*.
- García-Ajofrín, L. (2017). *Gigantes de la Educación*. Grupo Planeta.
- Google. (2020). *Google Play*. Obtenido de <https://play.google.com/store>
- Google Developers. (2020). *Developers*. Obtenido de <https://developer.android.com/studio>
- GSM Arena. (2020). *GSM Arena*. Obtenido de https://www.gsmarena.com/razer_phone_2-9363.php
- Hagos, T. (2019). *Android Studio IDE Quick Reference*. Apress.
- Isnaini, R., & Wawan, C. (2018). The Influence of Educational Robotics to Computational Thinking Skill in Early Childhood Education. *The 1st International Conference on Computer Science and Engineering Technology Universitas Muria Kudus*.
- Kägi, A., Herzog, J., & Stähli, D. (2019). *Mietrecht: Entwicklungen 2018*. Stämpfli Verlag.
- Kikitamara, S., & Ary Noviyanti, A. (2018). A Conceptual Model of User Experience in Scrum Practice. *2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE)*.
- Kromann, F. (2018). *Beginning PHP and MySQL*. Apress.
- LEGO Group. (2018). *Lego Education*. Obtenido de <https://education.lego.com/en-au/support/mindstorms-ev3/nxt-compatibility>
- Lei, H., Ganjeizadeh, F., Jayachandran, P., & Ozcan, P. (2017). A statistical analysis of the effects of Scrum and Kanban on software development projects. *Robotics and Computer-Integrated Manufacturing*, 59-67.

- López, R. (2018). Ecuador: La brecha digital entre profesores y alumnos. *Arcondecio*.
- Luna, T. (2018). *Diseño y construcción de un asistente robótico y una aplicación móvil lúdica para brindar soporte educativo a niños de 3 a 7 años*.
- Ministerio de Educación. (10 de Noviembre de 2017). *Ministerio de Educación*. Obtenido de <https://educacion.gob.ec/estudiantes-de-orellana-entre-los-ganadores-en-concurso-latinoamericano-de-robotica/>
- Municipalidad de Guayaquil. (2018). Obtenido de Guayaquil IN: <http://guayaquilin.org/#escuelas-in>
- Nevarez, M., Cedeño, G., Quiñonez, V., & Godoy, P. (2016). *Robótica educativa aplicada en estudiantes de educación general básica utilizando técnicas de aprendizaje colaborativo*.
- Nugroho, L., Azis, A., Mustika, I., & Selo. (2017). Development of RESTful API to support the oil palm plantation monitoring system. *2017 7th International Annual Engineering Seminar (InAES)*, (págs. 1-5). Yogyakarta.
- Oliveira de Azevedo, S., Etienne Bezerra, J., & Cunha de Miranda, L. (2017). A methodology of contextualized educational robotics. *2017 IEEE Frontiers in Education Conference (FIE)*.
- Pérez Buj, G., & Diago, P. (2018). Exploratory study on symbolic programming languages in problem-solving activities with Bee-bot.
- Pineda, E., & González, C. (2016). *Destrezas en materia de redes en América Latina*.
- Pittí, K., Curto, B., Moreno, V., & Rodríguez Conde, M. (2014). Uso de la Robótica como herramienta de aprendizaje en Iberoamérica y España. *VAEP-RITA*, 41-48.
- Plaza, P., Sancristobal, E., Carro, G., & Castro, M. (2016). Robotic Educational Tool to engage students on Engineering. *2016 IEEE Frontiers in Education Conference (FIE)*.
- Powers, D. (2019). *PHP 7 Solutions*. Apress.
- Preciado Alvarado, C. (2016). *Determinación e implementación de una metodología en robótica que potencialice los conocimientos de enseñanza aprendizaje en los estudiantes*. Machala: Universidad Técnica de Machala.

- Ribas, E. (29 de Mayo de 2018). *IEBS*. Obtenido de <https://www.iebschool.com/blog/que-es-api-rest-integrar-negocio-business-tech/>
- Rivas, B., Gálvez, M., Gétrudix Barrio, M., & Esteban, N. (2017). *La innovación educativa como agente de transformación digital en la Educación Superior. Acciones para el cambio*. Midac.
- Robomatrix. (2019). *Robomatrix*. Obtenido de <http://robomatrix.org/>
- Robotic Kids. (2018). *Robotic Kids*. Obtenido de <https://www.robotickids-ecuador.com/>
- Robotic Minds. (2020). *Robotic Minds*. Obtenido de <https://www.roboticminds.com.ec/>
- Rodriguez Perez, S., Fatoum, A., & Abke, J. (2016). Development of an Eclipse plug-in for using the LEGO Mindstorms EV3 in education. *2016 IEEE Global Engineering Education Conference (EDUCON)*, (págs. 631-636). Abu Dhabi.
- Rodríguez, V. (2019). Obtenido de Primicias: <https://www.primicias.ec/noticias/tecnologia/educacion-stem-impulsa-ninos-robotica/>
- Rollin, M. (2014). *Beginning LEGO MINDSTORMS EV3*. Apress.
- Romero, M., & Dupont, Y. (2106). Educational robotics: from procedural learning to co-creative project oriented challenges with Lego WEDO. 6159-6163.
- Ruge, I., Jiménez, F., & Henández, G. (2017). Daro Robot: Robotic Platform for Engineering Education. *Ingenium*, 58-74.
- Sánchez, C. (27 de Agosto de 2019). *El Universo*. Obtenido de <https://www.eluniverso.com/noticias/2019/08/27/nota/7489359/robotica-cada-vez-mas-espacios-sistema-educativo>
- Sánchez, J. (2011). *Diagnóstico y aplicación de los estilos de aprendizaje en los estudiantes del bachillerato internacional: Una propuesta pedagógica para la enseñanza eficaz de la robótica educativa*. Madrid.
- Sara, M. (2011). A study on the usefulness of educational robotics from educators' perspective. *Revista de Pedagogía*, 81-117.
- Shuber, K. (2013). *Scrum Guides*. Obtenido de <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf>
- SOLACYT. (2014). *SOLACYT*. Obtenido de <http://solacyt.org/>

- Soria , E., & Rivero , C. (2019). Pensamiento computacional: una nueva exigencia para la educación del siglo XXI. *Revista Espaço Pedagógico*, 323-337.
- Sović, A., Jagušć, T., & Seršić, D. (2014). How to teach basic university-level programming concepts to first graders? *2014 IEEE Integrated STEM Education Conference*.
- Strnad, B. (2017). Programming Lego Mindstorms for First Lego League Robot game and technical interview. *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*.
- Swe Khine, M. (2017). *Robotics in STEM Education*. Springer.
- Temitayo Faniran, V., Badru, A., & Ajayi, N. (2017). Adopting Scrum as an Agile approach in distributed software development: A review of literature. *2017 1st International Conference on Next Generation Computing Applications (NextComp)*.
- Terranova, A. (14 de Febrero de 2017). *Make: Community*. Obtenido de <https://makezine.com/2017/02/14/robotics-kit-right/>
- The LEGO Group. (2019). *LEGO Education*. Obtenido de <https://education.lego.com/en-us>
- The LEGO Group. (2020). *LEGO*. Obtenido de <https://www.lego.com/es-ar/themes/mindstorms/downloads>
- Torrejón Marín, M., & Ventura-Campos, N. (2019). Music-mathematical teaching-learning using educational robotics. *Cuadernos de desarrollo aplicados a las TIC*.
- Universidad Metropolitana del Ecuador. (30 de Julio de 2019). *Universidad Metropolitana del Ecuador*. Obtenido de <https://www.umet.edu.ec/reunion-entre-umet-y-bright-world-education-enterprises/>
- Verónica Segovia, M., & Abner de S. Souza, A. (2018). Educational Robotics as a Motivational Tool for the English Teaching-Learning Process for Children. *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*.
- Vijayarathy, L., & Butler, C. (2016). Choice of Software Development Methodologies: Do Organizational, Project, and Team Characteristics Matter? *IEEE Software*, 86-94.
- Vilhete, J., & Orlando, K. (2017). Education and Educative Robotics. *Revista de Educación a Distancia*.

- Villacrés, J., Sampedro, M., & Andrade, C. (2020). Educational robotics applied to the understanding of propositional logic. *Polo del Conocimiento*.
- Voštinár, P., & Klimová, N. (2019). Experience with Using Robots for Teaching Programming. *2019 International Conference on Information and Digital Technologies (IDT)*.
- World Economic Forum. (2017). *World Economic Forum: New Vision for Education: Fostering Social and Emotional*.
- Zhan, Y., & Hsiao, M. (2018). A Natural Language Programming Application for Lego Mindstorms EV3. *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*.