

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA



TESIS DE GRADO

**“APLICACIÓN ANDROID DE SIMULACIÓN DE PRUEBAS PISA
PARA ESTUDIANTES DE NIVEL SECUNDARIO“**

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS

POSTULANTE: JOEL LUIS CHAMBILLA RENJEL

TUTOR METODOLÓGICO: LIC. FREDDY MIGUEL TOLEDO PAZ

ASESOR: M.SC. CARLOS MULLISACA CHOQUE

LA PAZ - BOLIVIA

2018



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

DEDICATORIA

A mis padres gracias a sus consejos y apoyo he podido alcanzar esta meta y mejorar muchos aspectos de mi vida. A mis hermanos por el constante apoyo y animo en momentos de dificultad, sin el cual no hubiera sido posible llegar hasta este momento de mi vida.

AGRADECIMIENTOS

Agradecer a Dios por darme la oportunidad de lograr mis metas y acompañarme siempre en mi camino de vida.

Agradecer a la Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales, Carrera de Informática, la casa de estudios que me acogió durante mi formación para llegar a esta etapa profesional y en ella a los distinguidos docentes por su profesionalismo y ética.

Agradezco a mi tutor metodológico Lic. Freddy Miguel Toledo Paz, que me ha guiado en mi investigación durante cada etapa de este trabajo, sus observaciones y correcciones fueron de gran ayuda para mí.

De igual manera un agradecimiento especial a mi asesor M.Sc. Carlos Mullisaca Choque, por brindarme su tiempo dándome sus correcciones siempre con la verdad y con mucha paciencia.

También agradezco a todo el plantel Docente que durante todo este tiempo instruyó y capacitó, transmitiéndome sus conocimientos y experiencias.

INDICE

CAPÍTULO I	Pág.
INTRODUCCIÓN.....	1
1.1 Antecedentes.....	1
1.2 Planteamiento del problema.....	4
1.2.1 Problemas específicos.....	5
1.3 Justificación.....	6
1.4 Hipótesis.....	7
1.5 Objetivos.....	7
1.5.1 Objetivos general.....	7
1.5.2 Objetivos específicos.....	7
1.6 Alcances y limites.....	7
1.6.1 Delimitación temática.....	7
1.6.2 Delimitación temporal.....	8
1.6.2 Delimitación espacial.....	8
1.7 Aportes.....	8
 CAPITULO II	
 MARCO TEORICO	
2.1 Educación.....	9
2.2 Educación formal.....	9
2.3 Pisa.....	11
2.4 Aplicaciones académicas.....	15

2.4.1 Beneficios de la aplicaciones académicas.....	16
2.5 Simulación.....	16
2.6 Dispositivos móviles.....	18
2.7 Sistemas operativos móviles.....	19
2.7.1 Características básicas.....	20
2.8 Sistemas operativos mas utilizados.....	21
2.9 Sistemas operativos más utilizados.....	21
2.9 Sistema operativo Android.....	22
2.9.1 Características.....	22
2.9.2 Almacenamiento y recuperación de datos.....	24
2.9.3 SQLite.....	25
2.9.4 Características de SQLite.....	25
2.9.5 Clases java de SQLite.....	26
2.10 Android Studio.....	26
2.10.1 Principales características.....	28
2.10.2 Requerimientos del sistema.....	30
2.10.3 Comparativa entre Android studio y Eclipse.....	30
2.10.4 Ventajas de uso de Android studio.....	32
2.10.5 Desventajas de uso de Android studio.....	33
2.11 Servicios Web.....	33
2.11.1 Características.....	34
2.11.2 Tipos de servicios Web.....	35
2.11.3 Servicios web SOAP.....	35
2.11.4 Servicios web RESTful.....	36

2.11.5 XML vs JSON.....	36
2.12 Firebase.....	37
2.11.1 Autenticación de usuarios con Firebase.....	40
2.11.2 Almacenamiento y analítica con Firebase.....	40
2.13 Metodología de desarrollo móvil.....	42
2.14 Mobile-D.....	42
2.14.1 Fase de exploración.....	43
2.14.1.1 Definición de partes interesadas.....	44
2.14.1.2 Definición de alcance.....	44
2.14.1.3 Definición del proyecto.....	45
2.14.2 Fase de inicialización.....	45
2.14.2.1 Configuración del proyecto.....	45
2.14.2.2 Planificación inicial.....	45
2.14.3 Fase de producto.....	46
2.14.4 Fase de estabilización.....	48
2.14.4.1 Etapas.....	48
2.14.5 Fase de pruebas y reparación del sistema.....	49

CAPITULO III

MARCO APLICATIVO

3.1 Exploración.....	51
3.1.1 Establecimiento de los usuarios.....	51
3.1.1.1 Estudiante de secundaria.....	51
3.1.1.2 Administrador.....	52

3.1.2 Establecimiento del proyecto.....	52
3.1.3 Requisitos de sistema.....	52
3.2 Inicialización.....	53
3.2.1 Configuración del proyecto.....	53
3.2.1.1 Android.....	53
3.2.1.2 Firebase.....	53
3.2.2 Requerimientos iniciales.....	53
3.2.3 Historias de usuario.....	54
3.2.4 Análisis de requerimientos.....	58
3.2.5 Arquitectura de la aplicación.....	59
3.2.6 Planificación de fases.....	60
3.2.7 Diagramas de casos de uso.....	61
3.3 Fase de producción.....	61
3.3.1 Implementación de la aplicación.....	62
3.3.1.2 Pantalla de la aplicación.....	62
3.3.2 Modelo de base de datos.....	70
3.4 Fase de estabilización.....	71
3.5 Pruebas del sistema.....	72

CAPÍTULO IV

ANÁLISIS Y RESULTADO

4.1. Experimentación.....	73
4.2 Evaluación De La Variable Dependiente.....	73
4.2.1 Determinación del tamaño de muestra.....	74

4.2.2 Proceso.....	74
--------------------	----

4.3 Resultados de la encuesta.....	75
------------------------------------	----

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones.....	81
------------------------	----

5.2. Recomendaciones.....	82
---------------------------	----

BIBLIOGRAFÍA.....	83
-------------------	----

INDICE DE TABLAS

Tabla 2.1 Requerimientos de Android Studio.....	30
Tabla 2.2 Comparativa Android Studio y Eclipse.....	31
Tabla 3.1 Historia de Usuario 1.....	54
Tabla 3.2 Tarea 1.1: Generación de la base de datos.....	55
Tabla 3.3 Historia de Usuario 2.....	55
Tabla 3.4 Tarea 2.1: consulta a la base de datos.....	56
Tabla 3.5 Historia de Usuario 3.....	56
Tabla 3.6 Tarea 3.1 Registro de respuestas del estudiante.....	57
Tabla 3.7 Historia de Usuario 4.....	57
Tabla 3.8 Tarea 3.1 Registro de respuestas del estudiante.....	58
Tabla 3.9 Planificación de fases.....	60
Tabla 4.1 Distribucion de intervalos de puntuacion de estudiantes.....	74
Tabla 4.2 Resultados pregunta 1.....	75
Tabla 4.3 Resultados pregunta 2.....	76
Tabla 4.4 Resultados pregunta 3.....	77
Tabla 4.5 Resultados pregunta 4.....	78
Tabla 4.6 Resultados pregunta 5.....	79
Tabla 4.7 Resultados pregunta 6.....	80

INDICE DE FIGURAS

Figura 2.1 Países participantes de la prueba PISA.....	14
Figura 2.2 Arquitectura sistemas operativos móviles.....	20
Figura 2.3 Cuota de mercado de sistemas operativos.....	22
Figura 2.4 Arquitectura Android.....	24
Figura 2.5 Pantalla principal de Android Studio.....	27
Figura 2.6 Diferencias SOAP y REST.....	37
Figura 2.7 Firebase multiplataforma	39
Figura 2.8 Entorno del motor de aplicaciones firebase.....	41
Figura 2.9 Fases de la metodología Mobile-D.....	43
Figura 2.10 Proceso Fase de exploración.....	44
Figura 2.11 Fase de inicialización.....	46
Figura 2.12 Proceso de la fase de producto.....	47
Figura 2.13 Fase de estabilización.....	48
Figura 2.14 Fase de pruebas.....	49
Figura 3.1: Diagrama de contexto.....	59
Figura 3.2 Diagrama de casos de uso	61
Figura 3.3: Pantalla inicial.....	62
Figura 3.4 Pantalla de registro.....	63
Figura 3.5 Menu principal.....	64
Figura 3.7 Menu de la prueba.....	65
Figura 3.8 Realizacion de la prueba.....	66
Figura 3.9 Pantalla de fallo de respuesta.....	67

Figura 3.10 Pantalla de acierto de respuesta.....	68
Figura 3.11 Despliegue de resultados.....	69
Figura 3.12: Diagrama relacional.....	70
Figura 4.1 Resultados pregunta 1.....	70

RESUMEN

La educación es fundamental para alcanzar condiciones de vida satisfactorias según las distintas necesidades y expectativas de una sociedad tan heterogénea y compleja como la boliviana. La problemática de la educación es uno de los temas de mayor preocupación para la sociedad civil y tiene una importancia gravitacional sobre todos los aspectos de la vida política, económica y social. Por tanto es de vital importancia realizar mediciones y evaluaciones para reconocer e identificar el grado de conocimiento que poseen los estudiantes del sistema nacional de educación. Y de esta forma obtener resultados para su posterior análisis que permitan el encaminamiento y posterior toma de decisiones que permitan encaminar acciones y políticas que permitan mejorar la educación impartida a los estudiantes, innovar en metodologías y enfoques de educación.

En este trabajo se ha desarrollado una aplicación android, que permite realizar las preguntas del Programa Internacional para la Evaluación de Estudiantes o Informe **PISA**, obtener informes, resultados sobre el nivel académico que poseen los estudiantes del ciclo secundario de educación, y a través de los resultados obtenidos ver sus fortalezas y debilidades y de esta manera poder coadyuvarlos a mejorar y ampliar sus conocimientos.

La metodología empleada es Mobile-D, y se centra en cinco fases: exploración, donde se definen los grupos de interés y otros detalles, inicialización, se realiza la planificación de las iteraciones, producción, se planifica y realiza la implementación de las iteraciones, estabilización, realizan pruebas de aceptación, además de la documentación, prueba de sistema, la última etapa consta de probar por última vez la aplicación. Para finalizar se hizo pruebas con estudiantes de unidades educativas y se tomaron los resultados obtenidos por los mismos para su posterior análisis.

ABSTRACT

Education is fundamental to achieve satisfactory living conditions according to the different needs and expectations of a society as heterogeneous and complex as Bolivian society. The problem of education is one of the issues of greatest concern for civil society and has a gravitational importance on all aspects of political, economic and social life.

Therefore, it is of vital importance to make measurements and evaluations to recognize and identify the degree of knowledge that the students of the national education system possess. And in this way obtain results for further analysis that allow the routing and subsequent decision making to direct actions and policies to improve the education provided to students, innovate in methodologies and approaches to education.

In this work an android application has been developed, which allows to ask the questions of the International Program for Student Evaluation or PISA Report, obtain reports, results on the academic level possessed by students of the secondary education cycle, and through the results obtained to see their strengths and weaknesses and in this way be able to help them improve and expand their knowledge.

The methodology used is Mobile-D, and it focuses on five phases: exploration, where stakeholders are defined and other details, initialization, planning of iterations, production, planning and implementation of the iterations, stabilization, perform acceptance tests, in addition to the documentation, system test, the last stage consists of testing the application for the last time.

Finally, tests were done with students from educational units and the results obtained by them were taken for later analysis.

CAPITULO I

1. Introducción

El avance exponencial de la tecnología, el descubrimiento de nuevos conocimientos y el desarrollo económico y social, deben ir de la mano del desarrollo de la educación.

Una sociedad es una comunidad donde sus miembros han llegado a ser socialmente conscientes de su modo de vivir y están unidos por una escala común de objetivos y valores.

La educación es fundamental para alcanzar condiciones de vida satisfactorias según las distintas necesidades y expectativas de una sociedad tan heterogénea y compleja como la boliviana.

La problemática de la educación es uno de los temas de mayor preocupación para la sociedad civil y tiene una importancia gravitacional sobre todos los aspectos de la vida política, económica y social.

El entramado social, político, jurídico e institucional es muy importante para conocer y entender cómo se presenta la configuración de políticas educativas y cuáles son sus propuestas, fundamentos, innovaciones y de qué manera se traza el camino de su implementación, consolidando a futuro la posición y definición que adoptará Bolivia en el contexto regional y mundial desde la educación.

Durante las últimas décadas, la evaluación de los sistemas educativos se ha consolidado como una prioridad para las autoridades educativas de todo el mundo. Hoy son prácticas imprescindibles de la comunidad educativa la evaluación, como medida u observación del grado de consecución de un logro, y también como los juicios de valor que se

realizan sobre los objetivos alcanzados por el sistema educativo. Se han convertido en una herramienta útil para orientar las políticas educativas, un mecanismo de rendición de cuentas y transparencia, un medio que inspira soluciones para fomentar la excelencia y la equidad asentadas sobre unos fundamentos fiables y válidos.

Uno de los métodos mundialmente reconocidos para medir cualitativamente el nivel de educación de los estudiantes es el denominado Informe del Programa Internacional para la evaluación de estudiantes o Informe PISA (Programme for International Student Assessment), el cual es un estudio llevado a cabo por la Organización para la Cooperación y el Desarrollo Económicos (OCDE) a nivel mundial que mide el rendimiento académico de los alumnos en matemáticas, ciencia y lectura. Su objetivo es proporcionar datos comparables que posibiliten a los países mejorar sus políticas de educación y sus resultados.

PISA es una de las bases más sólidas sobre las que se pueden hacer juicios de valor respecto a los sistemas educativos. Se basa en un modelo dinámico de aprendizaje a lo largo de la vida centrado en lo que los jóvenes son capaces de hacer con lo que han aprendido tanto dentro como fuera del centro escolar, sopesando sus elecciones y tomando decisiones. Aparte de evaluar el conocimiento de los alumnos, PISA también examina su capacidad para aplicar ese conocimiento y experiencia a la vida real. Este estudio ha demostrado hasta qué punto las comparaciones son siempre estimulantes, las perspectivas múltiples son necesarias para valorar nuestra propia situación, más allá de ser una simple clasificación numérica, se ha reconocido mundialmente.

Bolivia no participa en mediciones internacionales de calidad educativa, de ningún tipo ya que se consideran de carácter “imperialista”, pero es necesario obtener algún parámetro,

para diagnosticar el estado actual de educación y a futuro poder mejorar las áreas en las que se tenga deficiencia.

1.1 Antecedentes

Bajo el argumento de construir propios sistemas y modelos de evaluación y seguimiento a la calidad educativa, el gobierno boliviano, junto a otros países de la región, el 2012 tomó la decisión de abandonar el Programa para la Evaluación Internacional de los Estudiantes PISA y no se lleva adelante un estudio de medición de la calidad educativa en el país desde el año 1997.

El 28 de junio de 2016, Bolivia se incorporó al Laboratorio Latinoamericano de Evaluación de la Calidad de la Educación (LLECE), instancia de la Unesco que emprenderá procesos de evaluación de la calidad educativa en el país, lo que permitirá visibilizar el estado de situación de la educación en Bolivia. El diagnóstico sobre la calidad de la educación se podrá conocer recién en la siguiente gestión.

El libro de la investigación “Medición de la calidad educativa en el Municipio de La Paz”, realizado por la Secretaría Municipal de Planificación para el Desarrollo, fue presentado este el 5 de julio de 2016. El trabajo, que midió la calidad educativa, otorga insumos para las propuestas de políticas públicas que permitan que las niñas, niños y jóvenes sean educados en escuelas dignas con mejor calidad educativa.

"El año 1997 la Unesco realizó el primer estudio internacional comparativo de lenguaje, matemáticas y factores asociados donde se aplicaron pruebas a los alumnos de tercero y cuarto grado de enseñanza básica, en trece países de América Latina a través de la prueba internacional Llece. De este modo mediante el Sistema de Medición y evaluación de la

Calidad de la educación y la UNESCO se realizó una primera aproximación para la medición de la calidad de la educación en Bolivia donde se paliaron exámenes de lenguaje y matemáticas a los alumnos de tercero y sexto de primaria" (Marcelo Arroyo, responsable GAMLP).

“El trabajo de campo de esta investigación se le realizó el 2014 con el objetivo de tener una comparabilidad internacional para medir la educación en el municipio de La Paz. Se tomó pruebas a estudiantes de 89 establecimientos educativos del municipio de La Paz, de los cuales son 65 públicos y 24 privados”, informó la gerente del Programa de Análisis e Investigación Municipal, Carla Cordero.

En este proyecto se evaluó a estudiantes de tercero y sexto de primaria y primero, tercero y sexto de secundaria en las materias de lenguaje y matemáticas. Entre los principales resultados alcanzados en la investigación figura que el promedio global de respuestas correctas a nivel general fue de 35% (estudiantes de primaria obtuvieron el 40,7% y los de secundaria, el 31,5%).

Se observó que a medida que sube la exigencia en cuanto a rendimiento aumentan las brechas entre centros educativos públicos y privados. Se identificó también que los estudiantes de instancias privadas y de nivel socioeconómico alto tienen un mejor desempeño.

En el portal web del informe PISA se ofrece un sistema de simulación de exámenes.

1.2 Planteamiento del problema

Uno de los principales problemas de la educación boliviana es el de la calidad. La formación que reciben los niños y jóvenes es deficiente, y presenta todavía grandes brechas entre lo urbano y rural, y de equidad de género.

El debate sobre educación, que debiera estar abierto a las voces y experiencias de quienes conocen la materia, es pobre y ha quedado reducido a comentarios que critican más periodos de gobierno que políticas educativas.

Si a este panorama agregamos que Bolivia no participa en mediciones internacionales de calidad educativa, la situación parece empeorar. El gobierno denuncia que PISA es una medición "imperialista" y se niega a asumir el desafío de instaurarla a nivel nacional.

Por tanto, no se cuenta con un parámetro para medir la calidad de educación que se da en nuestro país, que se refleja en bajos índices de rendimiento académico posterior a la educación secundaria.

Se plantea el siguiente problema:

¿Cómo coadyuvar a mejorar el nivel académico de los estudiantes de secundaria?

1.2.1 Problemas específicos

- Falta de voluntad del gobierno para participar de alguna prueba de medición de la calidad educativa a nivel internacional.
- Planes curriculares obsoletos en relación a los requerimientos académicos de la sociedad actual.
- Falta de motivación e incentivo a los estudiantes de secundaria.
- Falta de aplicación de metodologías de enseñanza efectivas y basadas en resultados.
- No se cuentan con indicadores y estándares de calidad, debidamente validados, no se puede evaluar la calidad de los programas de enseñanza.
- Falta de implementación de herramientas tecnológicas aplicadas en el sistema nacional de educación.

- Falta de políticas de promoción de programas educativos innovadores acordes al contexto actual en el cual se desarrolla la educación.

- No se tiene acceso al material de evaluación propuesto por PISA.

1.3 Justificación

1.3.1 Justificación social

La presente tesis pretende implementar una herramienta Android que principalmente sea de uso de los estudiantes de secundaria del sistema nacional de educación, aplicable al sistema fiscal, privado y de convenio, independientemente al nivel económico y social de la familia de estudiante, ya que las características que se pretenden alcanzar con los objetivos, son consideradas de utilidad para diagnosticar el nivel educativo que poseen los estudiantes, en un momento dado.

1.3.2 Justificación técnica

Para el desarrollo, pruebas e implementación del prototipo de la aplicación Android de simulación de pruebas PISA, se hará uso con dispositivos móviles de tipo Smartphone, móviles y tablets con sistema operativo android. El cual permite implementar de forma eficaz y eficiente la herramienta de simulación de exámenes.

1.3.3 Justificación económica

En el presente trabajo la justificación económica es parte fundamental del desarrollo de la misma, debido que al elegir la plataforma de desarrollo Android, el costo de implementar aplicaciones es prácticamente cero, ya que no es necesario pagar licencias de uso, y además de poseer herramientas de tipo Código abierto, la aplicación implementada puede ser distribuida libremente por uno de los canales oficiales de distribución.

1.4 Hipótesis

La aplicación android de simulación de pruebas PISA para estudiantes de nivel secundario incentiva a los estudiantes a mejorar su nivel académico.

1.5 Objetivos

1.5.1. Objetivo general

Desarrollar una aplicación android que permita a los estudiantes de nivel secundario acceder a pruebas PISA para evaluar su nivel académico.

1.5.2. Objetivos específicos

- Brindar un panorama de la situación educativa actual con referentes internacionales.
- Desarrollar una interfaz interactiva de fácil manejo para el estudiante de secundaria.
- Desarrollar exámenes para que sean resueltos y posteriormente calificados en tiempo real.
- Mostrar que los exámenes desarrollados para un dispositivo móvil son un método didáctico para diagnosticar el nivel académico.
- Desplegar información clara y concisa sobre los resultados obtenidos de las pruebas realizadas.
- Centralizar la información en una base de datos accesible desde cualquier plataforma.

1.6 Límites y alcances

1.6.1 Delimitación temática

La presente tesis de grado se desarrollará a través de una plataforma móvil, analizando y desarrollando las áreas del conocimiento a las cuales se enfoca el informe PISA que son: lectura, matemática y ciencias.

1.6.2. Delimitación temporal

El periodo de tiempo a ser tomado en cuenta será a partir de la fecha de planificación hacia adelante, debido a que no existen antecedentes oficiales sobre el presente trabajo de investigación.

1.6.3. Delimitación espacial

La presente tesis es aplicable a estudiantes del nivel secundario del sistema nacional de educación de carácter fiscal, privado y de convenio.

1.7. Aporte

Se incluirá una orientación individualizada donde los estudiantes podrán ver el grado de conocimiento que poseen.

La informática aplicada a la educación permitirá obtener un panorama de la calidad de educación que se da en nuestro país, específicamente en la ciudad de La Paz.

CAPITULO II

2. Marco teórico

2.1. Educación

La Educación es la formación práctica y metodológica que se le da a una persona en vías de desarrollo y crecimiento. La Educación es un proceso mediante el cual al individuo se le suministran herramientas y conocimientos esenciales para ponerlos en práctica en la vida cotidiana. En su sentido más amplio, la educación se entiende como el medio en el que los hábitos, costumbres y valores de una comunidad son transferidos de una generación a la siguiente generación. La educación se va desarrollando a través de situaciones y experiencias vividas por cada individuo durante toda su vida.

El concepto de educación comprende también el nivel de cortesía, delicadeza y civismo demostrado por un individuo y su capacidad de socialización. La educación es lo que transmite la cultura, permitiendo su evolución.

En el sentido técnico, la educación es el proceso continuo de desarrollo de las facultades físicas, intelectuales y morales del ser humano, con el fin de integrarse mejor en la sociedad o en su propio grupo, es un aprendizaje para vivir.

2.2 Educación formal

La educación denominada “formal” es aquella efectuada por maestros profesionales. Esta se vale de las herramientas que postula la pedagogía para alcanzar sus objetivos. En general, esta educación suele estar dividida según las áreas del saber humano para facilitar la asimilación por parte del educando. La educación formal se ha sistematizado a lo largo de los últimos 2 siglos en las instituciones escolares y en las universidades, en la actualidad el modelo de educación a distancia o semi presencial ha comenzado a abrirse camino como un nuevo paradigma.

En las sociedades modernas, la educación es considerada un derecho humano elemental; es por ello que suele ser ofrecida gratuitamente a los estudiantes por parte del estado. No obstante esta circunstancia, existen escuelas privadas que llenan las carencias que suelen tener las escuelas públicas.

La educación formal tiene distintos niveles que abarcan la niñez, adolescencia y vida adulta de una persona. Así, los primeros años de aprendizaje corresponden a la llamada educación primaria y tiene lugar durante la infancia.

A continuación, le siguen los años de educación secundaria, que corresponden a la adolescencia. Finalmente, en la edad adulta de una persona, la educación está reglada por el terciario o las carreras universitarias. Se advierte que, mientras que tanto la enseñanza primaria como la secundaria presentan carácter obligatorio en numerosas naciones, la

proporción de sujetos que completa estas etapas resulta en realidad reducida, en especial en los países no industrializados.

Este fenómeno da lugar a una reducción de las oportunidades futuras de trabajo y a un mayor riesgo de precarización laboral.

En algunas regiones del globo la educación se ve severamente afectada por las dificultades económicas. Así, la enseñanza que brinda el Estado puede considerarse de baja calidad con respecto a las posibilidades que ofrece una institución privada. Esta situación hace que aquellas personas con un entorno socioeconómico comprometido se vean desfavorecidas, situación que redonda en una desigualdad de oportunidades.

2.3. PISA

PISA (Programme for International Student Assessment) es un estudio internacional de evaluación educativa de las competencias alcanzadas por los alumnos a la edad de 15 años. Es un estudio cíclico que se repite cada tres años con el fin de apreciar la evolución en el tiempo del rendimiento escolar individual en las áreas evaluadas.

El estudio es impulsado por la Organización para la Cooperación y el Desarrollo Económico (OCDE), organización internacional de ayuda a los gobiernos en la formulación de políticas eficaces. Se orienta a la toma de decisiones para la conducción del sistema educativo y no a medir el rendimiento de los procesos educativos en cada aula o centro.

La primera aplicación del estudio PISA se realizó en el año 2000; las siguientes ediciones han sido en 2003, 2006, 2009, 2012 y 2015, proporcionando todas ellas una rica cantidad de información que se recoge en el informe de cada una de las ediciones que se publica posteriormente. El estudio PISA trata de contribuir a la evaluación de lo que los

jóvenes de 72 países saben y son capaces de hacer, este programa se centra en tres competencias consideradas troncales: ciencias, lectura y matemáticas. Dentro de la competencia de ciencias, PISA considera las siguientes formas de conocimiento: biología, geología, física, química y tecnología. Evalúa no sólo lo que el alumno ha aprendido en el ámbito escolar, sino también lo adquirido por otras vertientes no formales e informales de aprendizaje, fuera del centro escolar. Se valora cómo pueden extrapolar su conocimiento, sus destrezas cognitivas y sus actitudes a contextos en principio extraños al propio alumno, pero con los que se tendrá que enfrentar a diario en su propia vida. En cada ciclo, a los tres dominios troncales se añade un cuarto innovador, como fue la resolución creativa de problemas en 2012, la resolución colaborativa de problemas en 2015 o la competencia global, para el ciclo de PISA 2018.

Una característica destacada de este estudio es la muestra elegida. PISA no trata de evaluar a los alumnos escolarizados en un determinado curso académico, sino a aquéllos que hayan alcanzado la edad de 15 años. El muestreo se lo realiza en dos etapas: primero se elige un determinado número de centros educativos y luego se eligen unos 35 alumnos de 15 años en cada centro. Una vez descontados los alumnos excluidos y los que no participan por distintas causas, el número de alumnos evaluados por país debe ser superior a 4.500.

PISA evalúa tres competencias: lectura, matemáticas y ciencias. Todas las competencias se valoran en cada ciclo de PISA, pero una de ellas ocupa la mayor parte del tiempo de evaluación mientras que de las otras dos solo se hace un breve sondeo.

El estudio también recoge importante información de contexto de los propios alumnos y de los directores de los centros educativos. Esta información contextual, puesta en relación

con los resultados de las pruebas de rendimiento, ofrece una rica comparación de los rasgos que caracterizan a los sistemas educativos de los países participantes.

Los objetivos específicos de PISA son:

- Orientar las políticas educativas, al enlazar los resultados de los alumnos en las pruebas cognitivas con su contexto socio-económico y cultural, además de considerar sus actitudes y disposiciones, y al establecer rasgos comunes y diferentes en los sistemas educativos, los centros escolares y los alumnos.
- Profundizar en el concepto de ‘competencia’, referida a la capacidad del alumno para aplicar el conocimiento adquirido dentro y fuera de su entorno escolar, en las tres áreas clave objeto de evaluación del estudio.
- Relacionar los resultados de los alumnos con sus capacidades para el auto-aprendizaje y el aprendizaje a lo largo de la vida, incluyendo su motivación e interés, su autopercepción y sus estrategias de aprendizaje.
- Elaborar tendencias longitudinales para mostrar la evolución de los sistemas educativos en un plano comparativo internacional.

Todos los países participantes, tanto los pertenecientes a la OCDE como los asociados, han realizado un gran esfuerzo de coordinación para establecer unos estándares técnicos precisos y consensuados y así lograr una evaluación educativa fiable y válida, que sirva para la comparación internacional. De esta manera, se intenta orientar las políticas educativas hacia la mejora de los puntos débiles observados y la difusión de las buenas estrategias y prácticas.

PISA es una prueba que se aplica en gran parte del mundo. En su primera edición participaron 43 países y economías (2000); 45 países, en la segunda edición (2003); 57, en la

tercera (2006); 75, en la cuarta (2009) y 65, en la quinta (2012). En la edición de 2015 han participado 72 países y economías.

PISA es un esfuerzo cooperativo y colectivo. Los países participantes actúan por medio de sus representantes y expertos en los diversos grupos de trabajo e instituciones del estudio. En un programa de estas características, internacional y comparativo, se intenta siempre reducir los posibles sesgos culturales y lingüísticos, además de garantizar con múltiples verificaciones y controles internacionales su validez y fiabilidad, desde el diseño de las pruebas y su traducción hasta el muestreo y la recogida de datos.

En cada edición PISA, hay una competencia que constituye el dominio principal. En esta edición es ciencias, como ya lo fue en 2006, por lo que ahora se cierra el ciclo longitudinal (de nueve años) en esta competencia y se pueden comparar los resultados a lo largo del mismo. Este tipo de evaluación hace hincapié en la comprensión de los conceptos y en la capacidad para aplicarlos. La OCDE define el conocimiento en ciencias como la capacidad de involucrarse con cuestiones, así como las ideas, relacionadas con la ciencia, como un ciudadano reflexivo. Una persona competente en ciencias es capaz de explicar fenómenos científicos, de evaluar y diseñar investigaciones científicas y de interpretar información y evidencias científicas.

Participación en PISA

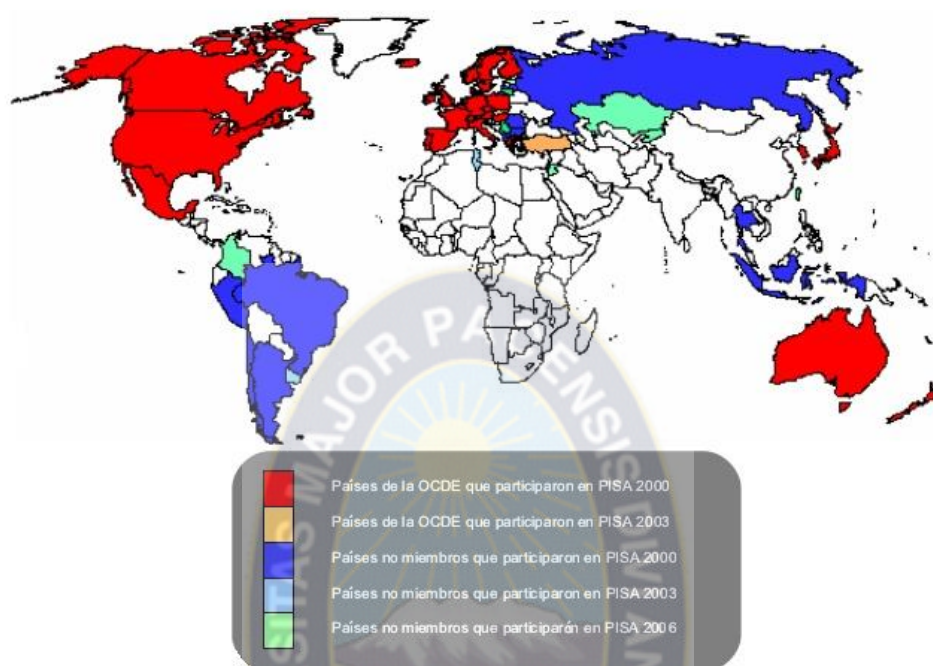


Figura 2.1 Países participantes de la prueba PISA

Además de las ciencias, en PISA también se evalúan la lectura, las matemáticas y la resolución colaborativa de problemas.

PISA para Centros Educativos es una prueba de lápiz y papel de aproximadamente dos horas de duración durante la cual los alumnos tendrán que responder a las preguntas de lectura, matemáticas y ciencias.

Los alumnos también deben cumplimentar un cuestionario de contexto de aproximadamente 35 a 40 minutos de duración. El cuestionario trata de recopilar información sobre los propios alumnos, sus hogares y sus centros escolares y experiencias de aprendizaje.

Los directores de centro deben responder a un cuestionario sobre el sistema de organización del centro escolar y el entorno de aprendizaje- Para recopilar más información,

algunos países como España, también realizan el cuestionario de profesor, así como los cuestionarios de padres, en los que se les pregunta a estos.

2.4 Aplicaciones académicas

Uno de los principales beneficios de las aplicaciones académicas es la opción de versatilidad, debido a que los usuarios pueden llevarla y usarla en cualquier momento, junto con el rápido crecimiento del uso de Smartphone por parte de la población, muestran que los teléfonos inteligentes son más que solo dispositivos de comunicación.

Las aplicaciones académicas son software con características específicas que tienen como fin estimular el aprendizaje de quienes hagan uso de él. Los objetivos con permitir a los estudiantes:

- Identificar los beneficios del uso de las aplicaciones académicas.
- Utilizar aplicaciones académicas para lograr alcanzar el máximo aprendizaje posible.
- Alcanzar las metas que se proponen apoyándose en estas aplicaciones de manera entretenida y didáctica.

2.4.1 Beneficios de las aplicaciones académicas.

Entre los principales beneficios que traen las aplicaciones académicas están la integración de las áreas de informática, ciencias sociales y recursos pedagógicos.

Las aplicaciones académicas además que brindan el apoyo sustentable para que el estudiante pueda enriquecer su conocimiento, por consiguiente, tener mejor preparación académica y de esta manera crear profesionales de alta gama.

El objetivo de una aplicación académica es darle la capacidad a un dispositivo electrónico de involucrarse en el proceso de enseñanza, convirtiéndola en un vehículo de información.

2.5 Simulación

La simulación es una herramienta de análisis que nos permite sacar resultados y conclusiones sin necesidad de trabajar directamente con el sistema real que se está simulando. La simulación es especialmente útil cuando no disponemos de dicho sistema real o nos resulta demasiado arriesgado realizar experimentos sobre él.

Es un intento de modelar situaciones de la vida real por medio de un programa informático, lo que requiere ser estudiado para ver cómo es que trabaja el sistema. La simulación informática se ha convertido en una parte útil del modelado de muchos sistemas naturales en física, química y biología y sistemas humanos como la economía y las ciencias sociales.

En muchas áreas de la ingeniería se utilizan los simuladores como una herramienta de trabajo más. Por ejemplo, en el diseño de nuevos fármacos se suelen utilizar modelos moleculares que sirven para simular mediante un ordenador la interacción entre compuestos químicos. Los ingenieros de automóviles también utilizan modelos computarizados para analizar el impacto de los choques en la seguridad de los viajeros.

En informática la simulación tiene todavía mayor significado especializado: Alan Turing usó el término “simulación” para referirse a lo que pasa cuando una computadora digital corre una tabla de estado (corre un programa) que describe las transiciones de estado, las entradas y salidas de una máquina sujeta a discreto-estado. La simulación computarizada de una máquina sujeta.

En programación, un simulador es a menudo usado para ejecutar un programa que tiene que correr en ciertos tipos de inconvenientes de computadora o en un riguroso controlador de prueba de ambiente. Por ejemplo, los simuladores son frecuentemente usados

para depurar un microprograma o algunas veces programas de aplicación comercial. Dado que, la operación de computadoras es simulada, toda la información acerca de la operación de computadoras es directamente disponible al programador, y la velocidad y ejecución pueda variar a voluntad.

Los simuladores pueden ser usados para interpretar la ingeniería de seguridad o la prueba de diseño de lógica VLSI, antes de que sean construidos. En informática teórica el término “simulación” representa una relación entre los sistemas de transición de estado. Esto es usado en el estudio de la semántica operacional.

En el área de las ciencias son de gran ayuda ya que los estudiantes relacionan conceptos abstractos con reales y también ayuda en el sentido de los recursos ya que solo se tiene que disponer con un par de computadores y no con todo el aparataje de un laboratorio entero.

Ventajas

- Nuevas políticas, procedimientos, reglas, flujos de información, entre otros pueden ser probados sin interrumpir el funcionamiento del sistema real.
- Nuevos diseños de hardware, emplazamientos físicos, sistemas de transporte, entre otros pueden ser testeados sin comprometer los recursos.
- Probar hipótesis acerca del ¿Qué? Y el ¿Cómo? de algún fenómeno.
- El tiempo puede ser comprimido y expandido.

Desventajas

- La construcción de un modelo no es fácil ni cualquiera puede llegar y hacerlo, requiere de una preparación especial.
- Los resultados de la simulación pueden ser difíciles de interpretar.
- Hacer el análisis y el modelo de simulación puede ser muy caro tanto en tiempo como en Dinero.

2.6 Dispositivos móviles

Un dispositivo móvil es un equipo electrónico que cuenta con características de procesamiento, capacidades de memoria y conexión permanente diseñada para una función específica pero que puede realizar otros trabajos generales. Este tipo de terminal no solo es un teléfono inteligente, sino que se convierte en un computador que podría sin problemas estar en las mismas categorías que los computadores de alto rendimiento como, por ejemplo: pc portátiles, tabletas, entre otros. Con todas estas posibilidades hace muy difícil estandarizar las aplicaciones para que funcione en todos los dispositivos y en todas las plataformas. La principal dificultad es que los dispositivos poseen arquitecturas de hardware y software, lo que resulta un reto para cualquier desarrollador realizar una aplicación multiplataforma.

Entre las características más sobresalientes de los dispositivos móviles están:

- Son pequeños.
- Capacidad de procesar información.
- Memoria RAM.
- Conexión permanente.

Existen varios tipos de dispositivos móviles que dependiendo de las necesidades prestan funciones diferentes al usuario:

- Teléfonos móviles.

- Web-enabled phones.
- Two-way pagers.
- Smartphones.
- Handheld pc.
- Tablet pc.

2.7 Sistemas operativos móviles

El sistema operativo es un conjunto de programas informáticos enfocados a la administración de los recursos que ofrece un computador, es el encargado de realizar un vínculo entre los recursos hardware, el usuario, el usuario y las aplicaciones finales. Existen varios tipos de sistemas operativos, las clasificaciones son extensas dependiendo de la arquitectura del procesador.

Un sistema operativo móvil es un sistema operativo que controla un dispositivo móvil al igual que las computadoras, sin embargo, los sistemas operativos móviles son mucho mas simples y están orientados a la conectividad inalámbrica.



Figura 2.2 Arquitectura sistemas operativos móviles

2.7.1 Características básicas

- **Kernel:** El núcleo o kernel, proporciona el acceso a los distintos elementos del hardware del dispositivo. Ofrece distintos servicios a los demás elementos del dispositivo como son los controladores o drivers para el hardware, la gestión de procesos, el sistema de archivos y el acceso y gestión de memoria.
- **Middleware:** El middleware es el conjunto de módulos que hacen posible la propia existencia de aplicaciones para móviles. Es totalmente transparente para el usuario y ofrece servicios claves como el motor de mensajería y comunicaciones.
- **Entorno de ejecución de aplicaciones:** El entorno de ejecución de aplicaciones consiste en un gestor de aplicaciones y un conjunto de interfaces programables abiertas y programables por parte de los desarrolladores para facilitar la creación de software.

- **Interfaz de usuario:** Las interfaces de usuario facilitan la interacción con el usuario y el diseño de la presentación visual de la aplicación. Los servicios que incluye son el de componentes gráficos y el del marco de interacción.

2.8 Sistemas operativos más utilizados

- **Android:** Pertenece a google, es un sistema abierto, cualquier fabricante puede desarrollar en él sus productos. Las actualizaciones de Android se lanzan cada 6 o 9 meses, cada una con nombre distinto y mejoras de rendimiento. No obstante, su llegada al usuario final, tarda más, ya que cada fabricante trabaja para crear su propia versión respecto al hardware de sus teléfonos.

La principal ventaja de Android es su flexibilidad sobre muchos dispositivos, incluyendo tablets, ya que es un sistema operativo de código abierto. También permite la personalización gráfica y de animaciones de elementos del sistema.

- **IOS:** Es un sistema operativo móvil de Apple Inc. desarrollado originalmente para el iPhone siendo después usado en el iPod Touch y en el iPad, productos en los que únicamente se encuentra presenta. Permite una personalización limitada el sistema.

- **Windows Phone:** Desarrollado por Microsoft en el año 2000, su principal ventaja de este sistema es la compatibilidad de cualquier producto de Microsoft y su integración con sistemas Windows. Actualmente se ha descontinuado la producción de dispositivos móviles con este sistema operativo debido a la baja cuota de mercado que posee.



Figura 2.3 Cuota de mercado de sistemas operativos

2.9 Sistema operativo Android

Android es un sistema operativo basado en el núcleo de Linux creado inicialmente por Android Inc. que fue absorbido más tarde por Google Inc., en unión con la empresa de software Open Handset Alliance, un consocio experto en hardware, software y telecomunicaciones.

Gracias a que su plataforma es completamente abierta, su capacidad de adaptación a cualquier dispositivo, aplicaciones desarrolladas en java, pantalla adaptable, alta calidad de gráficos y procesamiento, la filosofía de siempre conectado a internet, entre otros hace que Android lidere el mercado estando presente en la mayoría de los dispositivos móviles.

El sistema permite programar aplicaciones en una variación de java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono de una forma muy sencilla en un lenguaje de programación muy conocido como es Java.

2.9.1 Características

Los componentes principales del sistema operativo Android son:

- Aplicaciones: Las aplicaciones base, incluyen un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y otros. Todas las aplicaciones están escritas en lenguaje de programación Java.

- Marco de trabajo de aplicaciones: Los desarrolladores tienen acceso completo a las mismas APIs del framework usadas por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes. Cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades. Este mismo mecanismo permite que los componentes sean reemplazados por el usuario.

- Bibliotecas: Android incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android. Algunas son: System C library, bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite entre otras.

- Runtime de Android: Android incluye un conjunto de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrito de forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente. Dalvik ejecuta archivos en el formato Dalvik Executable “.dex”, el cual está optimizado para memoria mínima. La máquina virtual está basada en registros y corre clases compiladas por el compilador de Java que se transformaron al formato .dex por la herramienta incluida “dx”.

- Núcleo Linux: Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El

núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software.

La estructura de Android está formada por varias capas: Kernel de Linux, librerías, framework, y aplicaciones.

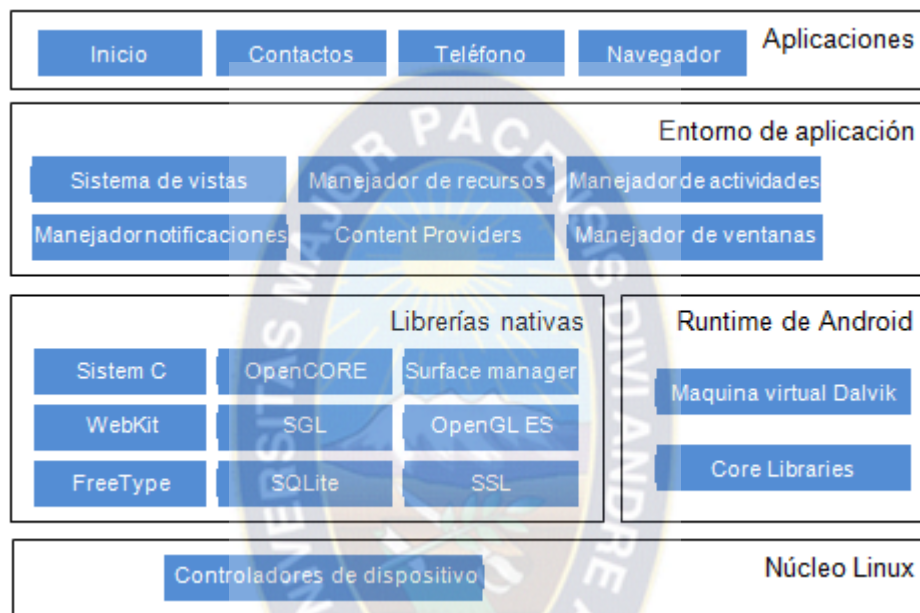


Figura 2.4 Arquitectura Android

2.9.2 Almacenamiento y recuperación de datos

Android ofrece diferentes opciones para almacenar datos persistentes de una aplicación. La solución que se elija depende de las necesidades dependiendo si los datos deben ser privados para tu aplicación o estar disponibles para otras aplicaciones, y la cantidad de espacio que requieren tus datos.

El almacenamiento y recuperación de datos de manera rápida y eficiente es esencial para los dispositivos cuya capacidad de almacenamiento se encuentra limitada.

El sistema operativo Android provee una base de datos relacional, ágil y liviana para aquellas aplicaciones que utilizan SQLite. Las aplicaciones pueden tomar ventaja del motor de base de datos relacional para almacenar datos de manera segura y eficiente.

2.9.3 SQLite

SQLite de datos Open Source, es muy popular en muchos dispositivos móviles pequeños, como android. El gestor de base de datos por defecto de Android es Lite. SQLite es una base de datos transaccional ligera que ocupa una cantidad muy pequeña de espacio en disco y memoria, de manera que es la elección perfecta para crear bases de datos en sistemas operativos para móviles como Android o IOS.

Aspectos a tener en cuenta cuando se maneja SQLite:

- SQLite no dispone de control de integridad por lo que puede almacenar un valor de un cierto en un campo de otro tipo distinto, como poner un string en un integer o viceversa.
- SQLite no gestiona directamente la integridad referencial, no soporta restricciones de clave foránea ni sentencias join. No obstante, se puede controlar mediante triggers.
- El soporte completo de Unicode es opcional y no está instalado por defecto.

2.9.4 Características de SQLite

SQLite es un sistema completo de bases de datos que soporta múltiples tablas, índices, triggers y vistas. No necesita un proceso separado funcionando como un servidor, ya que lee y escribe directamente sobre archivos que se encuentran en el disco duro.

El formato de la base de datos es multiplataforma e indistintamente se puede utilizar el mismo archivo en sistemas de 32 y 64 bits. La base de datos se almacena en un único fichero a diferencia de otros sistemas de gestión de base de datos que hacen uso de varios archivos.

SQLite emplea registros de tamaño variable de forma tal que se utiliza el espacio en disco que es realmente necesario en cada momento.

Existe un programa independiente de nombre SQLite que puede ser utilizado para consultar y gestionar los ficheros de base de datos SQLite.

2.9.5 Clases java de SQLite

SQLite esta escrito en C y envuelto en java por el SDK de Android. Este envoltorio esta formado por un conjunto de clases java que interactúan directamente con el sistema gestor de base de datos de SQLite.

Para crear y gestionar SQLite se usa `android.database.sqlite` que se trata de un paquete genérico, no es específico de ninguna base de datos. Aquí se encuentra las clases necesarias para crear, actualizar la base de datos, para realizar query, sentencias precompiladas, inserts, updates y deletes, la implementación de cursores, etc.

2.10 Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android. Posee un potente editor de códigos y herramientas para desarrolladores, Android Studio ofrece aún más funciones que aumentan tu productividad durante la compilación de apps para Android

- Un sistema de compilación basado en Gradle flexible
- Un emulador rápido con varias funciones
- Un entorno unificado en el que puedes realizar desarrollos para todos los dispositivos Android
- Instant Run para aplicar cambios mientras tu app se ejecuta sin la necesidad de compilar un nuevo APK

- Integración de plantillas de código y GitHub para ayudarte a compilar funciones comunes de las apps e importar ejemplos de código
- Gran cantidad de herramientas y frameworks de prueba
- Herramientas Lint para detectar problemas de rendimiento, usabilidad, compatibilidad de versión, etc.
- Compatibilidad con C++ y NDK
- Soporte incorporado para Google Cloud Platform, lo que facilita la integración de Google Cloud Messaging y App Engine.



Figura 2.5 Pantalla principal de Android Studio

Cada proyecto en Android Studio contiene uno o más módulos con archivos de código fuente y archivos de recursos. Entre los tipos de módulos se incluyen los siguientes:

- módulos de apps para Android
- módulos de bibliotecas

- módulos de Google App Engine

Android Studio consta de varias áreas lógicas, donde se puede realizar búsquedas en el código fuente, bases de datos, acciones, elementos de la interfaz de usuario, etc.

Además, Android Studio aplica automáticamente formatos y estilos según lo especificado en la configuración de estilo de código, brindando mayor legibilidad del código además de una combinación agradable de colores.

Android Studio usa Gradle como la base del sistema de compilación, Este sistema de compilación se ejecuta en una herramienta integrada desde el menú de Android Studio, y lo hace independientemente de la línea de comandos. Se puede usar las funciones del sistema de compilación para diferentes tareas.

Android Studio se ha mantenido durante todo este tiempo en versión beta, pero desde el 8 de diciembre de 2014, en que se liberó la versión estable de Android Studio 1.0, Google ha pasado a recomendarlo como el IDE para desarrollar aplicaciones para su sistema operativo, dejando el plugin ADT para Eclipse de estar en desarrollo activo. Esta versión la puedes descargar desde la web de Android Developer.

2.10.1 Principales características

Soporte para programar aplicaciones para Android Wear (sistema operativo para dispositivos corporales como por ejemplo un reloj).

- Herramientas Lint (detecta código no compatible entre arquitecturas diferentes o código confuso que no es capaz de controlar el compilador) para detectar problemas de rendimiento, usabilidad y compatibilidad de versiones.

- Utiliza ProGuard para optimizar y reducir el código del proyecto al exportar a APK (muy útil para dispositivos de gama baja con limitaciones de memoria interna).
- Integración de la herramienta Gradle encargada de gestionar y automatizar la construcción de proyectos, como pueden ser las tareas de testing, compilación o empaquetado.
- Nuevo diseño del editor con soporte para la edición de temas.
- Nueva interfaz específica para el desarrollo en Android.
- Permite la importación de proyectos realizados en el entorno Eclipse, que a diferencia de Android Studio (Gradle) utiliza ANT.
- Posibilita el control de versiones accediendo a un repositorio desde el que poder descargar Mercurial, Git, Github o Subversion.
- Alertas en tiempo real de errores sintácticos, compatibilidad o rendimiento antes de compilar la aplicación.
- Vista previa en diferentes dispositivos y resoluciones.
- Integración con Google Cloud Platform, para el acceso a los diferentes servicios que proporciona Google en la nube.
- Editor de diseño que muestra una vista previa de los cambios realizados directamente en el archivo xml.

2.10.2 Requerimientos del sistema

Windows	Mac OS	Linux
----------------	---------------	--------------

Microsoft Windows 8/7/Vista/2003 (32 o 64 bit)	Mac OS X 10.8.5 o superior, hasta la 10.9	GNOME o entorno de escritorio KDE
Mínimo de 2 GB de RAM, recomendado 4 GB de RAM		
400 MB de espacio en disco		
Necesita de al menos 1 GB para Android SDK, emulador de imágenes del sistema, y cachés		
Resolución mínima de pantalla de 1280 x 800		
Java Development Kit (JDK) 7 o superior		

Tabla 2.1 Requerimientos de Android Studio

2.10.3 Comparativa entre Android Studio y ADT Eclipse

El IDE Eclipse con el plugin ADT instalado permitía el desarrollo de aplicaciones android. Después del lanzamiento de Android Studio, los desarrolladores comenzaron a migrar de plataforma, debido a todas las ventajas que ofrecía el nuevo IDE.

Para especificar las diferencias y novedades que presenta Android Studio con respecto al IDE Eclipse, y más concretamente con el ADT para Android, se propone la siguiente tabla comparativa entre ambas opciones:

Características	Android Studio	ADT
Sistema de construcción	Gradle	ANT

Construcción y gestión de proyectos basado en Maven (herramienta de software para la gestión y construcción de proyectos Java, similar a Apache ANT, pero su modelo es más simple ya que está basado en XML)	Si	No (es necesario instalar un plugin auxiliar)
Construir variantes y generación de múltiples APK (muy útil para Android Wear)	Si	No
Refactorización y completado avanzado de código Android	Si	No
Diseño del editor gráfico	Si	Si
Firma APK y gestión de almacén de claves	Si	Si
Soporte para NDK (Native Development Kit: herramientas para implementar código nativo escrito en C y C++)	Próximas versiones	Si
Soporte para Google Cloud Platform	Si	No
Vista en tiempo real de renderizado de layouts	Si	No
Nuevos módulos en proyecto	Si	No
Editor de navegación	Si	No
Generador de assets	Si	No
Datos de ejemplo en diseño de layout (sin renderizar en tiempo de ejecución)	Si	No
Visualización de recursos desde editor de código	Si	No

Tabla 2.2 Comparativa Android Studio y Eclipse

2.10.4 Ventajas uso Android Studio

- Android Studio ha pasado a ser el entorno recomendado para el desarrollo de aplicaciones en Android, al tratarse de un IDE oficial de Google en colaboración con JetBrains (compañía de desarrollo software especializada en diseño de IDEs).
- Android Studio permite la creación de nuevos módulos dentro de un mismo proyecto, sin necesidad de estar cambiando de espacio de trabajo para el manejo de proyectos, algo habitual en Eclipse.
- Con la simple descarga de Android Studio se disponen de todas las herramientas necesarias para el desarrollo de aplicaciones para la plataforma Android.
- Su nueva forma de construir los paquetes .apk, mediante el uso de Gradle, proporciona una serie de ventajas más acorde a un proyecto Java:
 - Facilita la distribución de código, y por lo tanto el trabajo en equipo.
 - Reutilización de código y recursos.
 - Permite compilar desde línea de comandos, para aquellas situaciones en las que no esté disponible un entorno de desarrollo.
 - Mayor facilidad para la creación de diferentes versiones de la misma aplicación, que proporciona numerosas ventajas como puede ser la creación de una versión de pago y otra gratuita, o por ejemplo diferentes dispositivos o almacén de datos.
 - Integración con herramientas de versionamiento como Git

2.10.5 Desventajas uso Android Studio

- Aunque ya se ha lanzado la primera versión estable, la v1.0, al estar en una fase inicial, siempre es susceptible de introducirse más cambios que puedan provocar inestabilidad entre proyectos de diferentes versiones.
- Curva de aprendizaje más lenta para nuevos desarrolladores de Android.
- El sistema de construcción de proyectos Gradle puede resultar complicado inicialmente.
- En comparativa con Eclipse, menor número de plugins.

2.11 Servicios Web

Un Web Service, o Servicio Web, es un método de comunicación entre dos aparatos electrónicos en una red. Es una colección de protocolos abiertos y estándares usados para intercambiar datos entre aplicaciones o sistemas. Las aplicaciones escritas en varios lenguajes de programación que funcionan en plataformas diferentes pueden utilizar web services para intercambiar información a través de una red. La interoperabilidad, por ejemplo entre Java y Python o Windows y Linux se debe al uso de estándares abiertos.

Un servicio es una aplicación que desempeña una actividad de negocio, la cual proporciona una interfaz que puede llamarse desde otro programa, se registra y se localiza por medio de un servicio de registro. Los sistemas orientados a servicios constituyen la infraestructura de éstos, los Servicios Web pueden definirse como entidades programables que actúan como cajas negras para proporcionar una determinada funcionalidad, tal y como la de una aplicación o una funcionalidad de sistema. Emplean estándares de internet como HTTP o XML.

2.11.1 Características

Las características principales de los servicios Web, son las siguientes:

- Utilización de estándares de internet. La única forma para que los servicios Web sean utilizados por la cantidad de sistemas heterogéneos existentes en Internet es el empleo del protocolo de transferencia de datos HTTP utilizado por todos los navegadores Web y XML.
- Basados en tecnologías de paso de mensajes. La interacción entre el cliente y el proveedor del servicio es empaquetada en unidades autodescriptivas denominadas mensajes. Dicha interacción se describe en función de los mensajes intercambiados.
- Combinan lo mejor de la tecnología de componentes y de la tecnología Web. Los servicios Web presentan una funcionalidad de caja negra que puede ser reutilizada sin preocuparse de cómo es implementada y ello proporciona interfaces bien definidas.

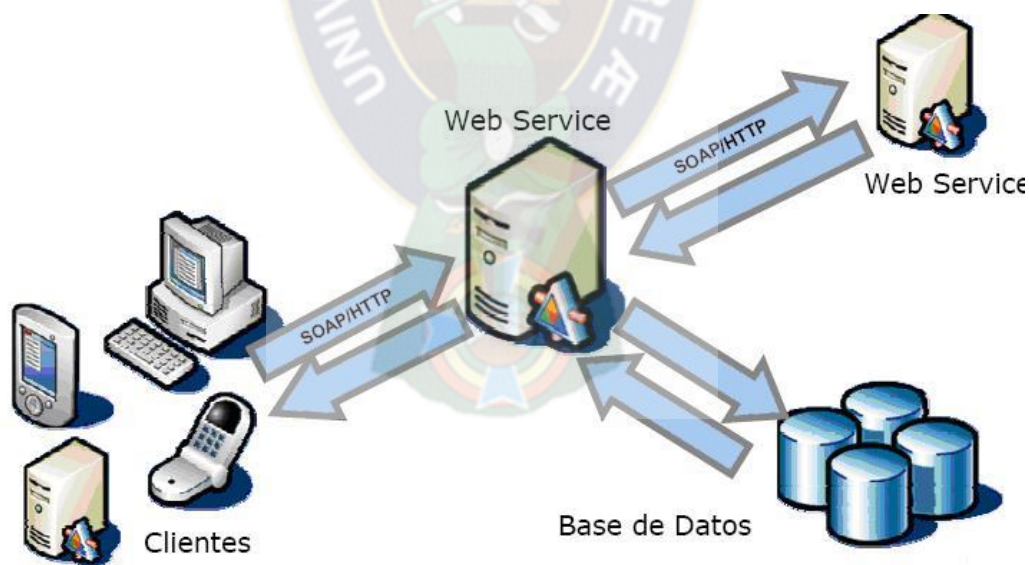


Figura 2.4 Diagrama Web Services

2.11.2 Tipos de servicios web

Un servicio es un componente software proporcionado a través de un endpoint accesible a través de la red. Los servicios productores y consumidores utilizan mensajes para intercambiar información de invocaciones de petición y respuesta en forma de documentos auto-contenidos que hacen muy pocas asunciones sobre las capacidades tecnológicas de cada uno de los receptores.

A nivel técnico, los servicios pueden implementarse de varias formas entre las cuales están los servicios Web SOAP, y servicios Web RESTful.

2.11.3 Servicios Web SOAP

Los servicios Web SOAP, utilizan mensajes XML para comunicarse que siguen el estándar SOAP (*Simple Object Access Protocol*), un lenguaje XML que define la arquitectura y formato de los mensajes. Dichos sistemas normalmente contienen una descripción legible por la máquina de la descripción de las operaciones ofrecidas por el servicio, escrita en WSDL (*Web Services Description Language*), que es un lenguaje basado en XML para definir las interfaces sintácticamente.

El formato de mensaje SOAP y el lenguaje de definición de interfaces WSDL se ha extendido bastante, y muchas herramientas de desarrollo, pueden reducir la complejidad de desarrollar aplicaciones de servicios Web.

El diseño de un servicio basado en SOAP debe establecer un contrato formal para describir la interfaz que ofrece el servicio Web. WSDL puede utilizarse para describir los detalles del contrato, que pueden incluir mensajes, operaciones, bindings, y la localización del servicio Web. También deben tenerse en cuenta los requerimientos no funcionales, como por ejemplo las transacciones, necesidad de mantener el estado, seguridad y coordinación.

2.11.4 Servicios Web RESTful

Los servicios Web RESTful (*Representational State Transfer Web Services*) son adecuados para escenarios de integración básicos *ad-hoc*. Dichos servicios Web se suelen integrar mejor con HTTP que los servicios basados en SOAP, ya que no requieren mensajes XML o definiciones del servicio en forma de fichero WSDL.

Los servicios Web REST utilizan estándares muy conocidos como HTTP, SML, URI, MIME, y tienen una infraestructura "ligera" que permite que los servicios se construyan utilizando herramientas de forma mínima. Gracias a ello, el desarrollo de servicios RESTful es barato y tiene muy pocas "barreras" para su adopción.

REST tiende a simplificar las cosas y usa el lenguaje JSON. En la actualidad REST y JSON se han convertido en la opción más sencilla y por tanto más recomendable para implementar un servicio web.

2.11.5 XML vs JSON

Para los sistemas de comunicación entre máquinas se requiere una serie de características, básicamente marcadas porque las máquinas implicadas en la comunicación pueden tener sistemas muy diferentes. Pueden usar lenguajes de programación o bases de datos diferentes y hasta los sistemas operativos suelen ser distintos entre clientes y servidores. Por ello, para la comunicación de los datos se usa básicamente lenguajes escritos en archivos de texto plano.

Entre los lenguajes más usados para la comunicación de los datos en los servicios web, encontramos XML y JSON.

- XML está basado en etiquetas, como HTML. Es más tradicional pero también es un lenguaje más avanzado, que presenta diversas utilidades para su extensión, validación de la información y sintaxis de los datos, etc.

- JSON es un lenguaje reciente, basado en sintaxis Javascript. Generalmente, es más ligero y requiere mucha menor carga del servidor para su procesamiento.



Figura 2.6 Diferencias SOAP y REST

2.12 Firebase

Firebase es una plataforma para el desarrollo de aplicaciones web y aplicaciones móviles desarrollada por James Tamplin y Andrew Lee en 2011 y adquirida por Google en 2014.

Proporciona funciones como estadísticas, bases de datos, informes de fallos y mensajería, de manera rápida y flexible.

Firebase utiliza la infraestructura de Google y se escala automáticamente, incluso para las apps más grandes. Los productos de Firebase funcionan a la perfección por sí solos, pero comparten datos y estadísticas, por lo que funcionan aún mejor juntos.

Google basa el posicionamiento de producto de Firebase en cuatro grandes bases de negocio, que pretende que sean un catalizador del alojamiento de apps de terceros:

- Una serie de características para el desarrollo de aplicaciones móviles: Google ofrece almacenamiento de datos en la nube, procesos de autenticación de usuarios, servicio de base de datos, reporte de errores.
- Servicio de analítica: Google dispone de Firebase Analytics, una solución de analítica incluida dentro del servicio gratuito para medir el comportamiento del usuario dentro de los productos desde un mismo panel de métricas.
- Recursos para el crecimiento: el buscador presta atención a la distribución para el crecimiento de la audiencia. Por ejemplo, a través de notificaciones. Aquí también entraría todo lo relacionado con publicidad dentro de Adwords.
- Monetización: Firebase está integrada con una herramienta de monetización a través de la inclusión de publicidad en las aplicaciones llamada AdMob.

Para desarrollar un proyecto con Google Firebase lo único que es necesario es disponer de una cuenta de correo electrónico Gmail, como casi todos los productos del buscador. Firebase da a los desarrolladores de aplicaciones una plataforma que almacena y sincroniza con cada aplicación la información alojada en una base de datos NoSQL en la nube. Esos datos siempre se sincronizan en tiempo real y siguen estando disponibles cuando la aplicación pierde la conexión a internet.

Esa base de datos NoSQL recibe el nombre de Firebase Realtime Database. En ella se almacenan los datos en formato JSON y se sincronizan con cada uno de los clientes que se

encuentran conectados en ese momento. La idea es que, cuando un equipo de desarrollo crea una aplicación con los SDKs de Android, iOS o JavaScript de Firebase (Figura 2.7), todos esos clientes comparten una misma instancia dentro de esa base de datos NoSQL y las peticiones que hacen los usuarios reciben siempre como respuesta los datos actualizados para su petición en tiempo real.

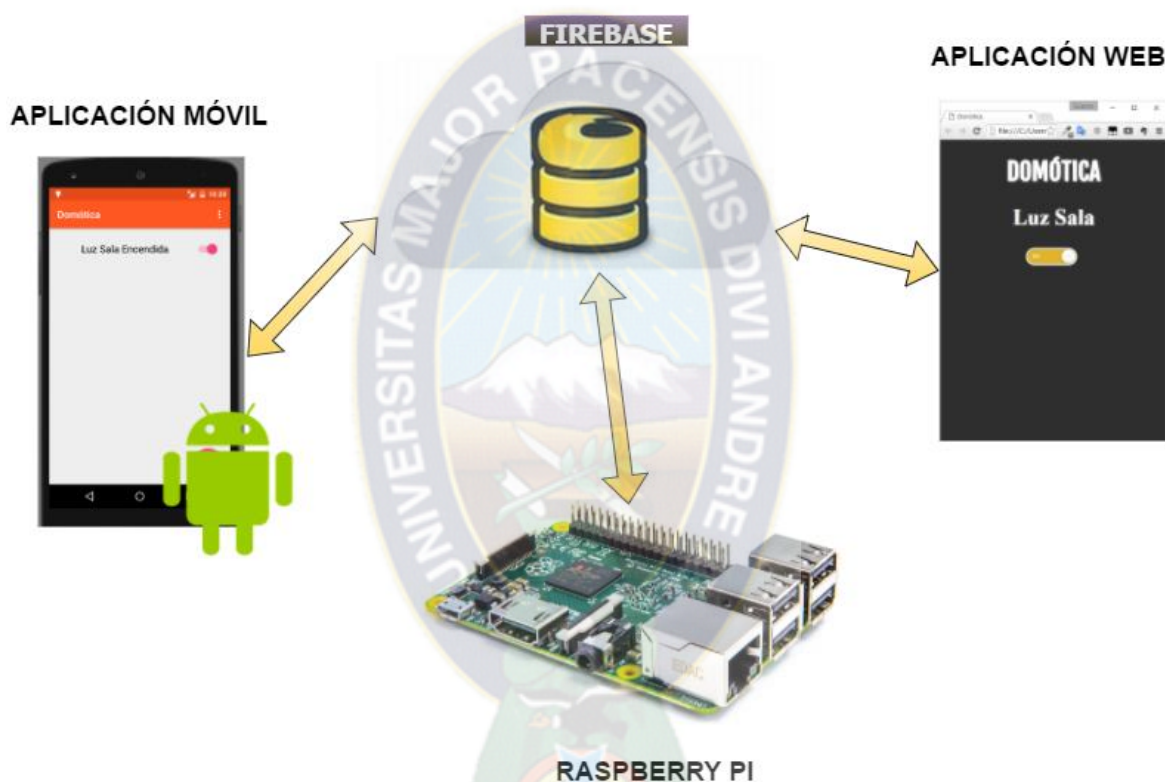


Figura 2.7 Firebase multiplataforma

Las principales ventajas que proporciona firebase son:

2.12.1 Autenticación de usuarios con Firebase

Todo el proceso de autenticación de la identidad de los usuarios en Firebase, bajo el servicio Firebase Authentication, se hace bajo estándares del mercado como OAuth 2.0 y OpenID Connect. La idea es que las aplicaciones sean capaces de reconocer la identidad del

usuario, guardar sus datos personales y ofrecer la misma experiencia personalizada desde cualquier dispositivo y desde cualquier sistema operativo.

Para iniciar sesión de usuario en una aplicación alojada en la nube como Firebase es necesario facilitar unas credenciales que te identifiquen como ese usuario concreto, a través de un correo electrónico y una contraseña o mediante el uso de un token OAuth de un proveedor de identidades federadas si el *login* se produce a través de servicios externos populares como Facebook y Twitter o Google Plus.

2.12.2 Almacenamiento y analítica con Firebase

Los otros dos grandes pilares de Firebase son el almacenamiento y el servicio de analítica pensado para las métricas relacionadas con el comportamiento del usuario. El primero de los productos, Firebase Storage, ofrece la posibilidad de hospedar imágenes, audio, vídeo y cualquier otro contenido generado por el propio usuario. Detrás de Firebase Storage está Google Cloud Storage, un servicio mayor que le sirve de respaldo. La idea de este almacenamiento es dar servicio a aquellas aplicaciones que almacenan y proporcionan contenido creado por sus usuarios.

Aunque Firebase Storage está respaldado por detrás por Google Cloud Storage, eso no significa que sea un servicio no escalable, es todo lo contrario. Esa escalabilidad, siempre y cuando sea requerida por el cliente, se hace de forma automática. El acceso y la administración de los datos acumulados en Firebase Storage se hace a través de sus SDKs o a través de las APIs de Google Cloud, que gestionan toda esa información en dos formatos habituales: XML y JSON.

Firestore es una solución gratuita y casi ilimitada, dispone de la capacidad de generar informes de métricas a partir de un total de 500 eventos, un número bastante elevado para un servicio totalmente gratuito. Esos eventos se pueden implementar en cualquier aplicación nativa, sea para el entorno iOS o Android, a través del SDK del servicio de analítica.

El entorno estándar de App Engine es una plataforma de aplicaciones que monitorea, actualiza y escala el entorno de alojamiento (Figura 2.8); todo lo que necesita hacer es escribir su código de servicio de back-end móvil.

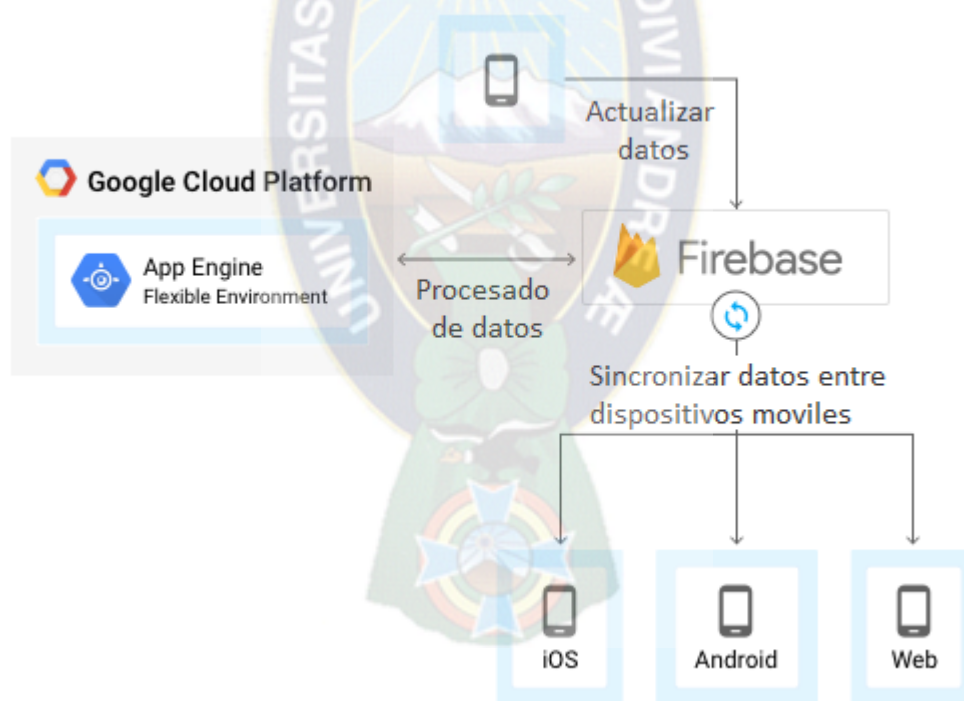


Figura 2.8 Entorno del motor de aplicaciones firebase

2.13 Metodología de desarrollo móvil

“Una metodología es una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por

implementar nuevos sistemas de información. Una metodología está formada por fases, cada una de las cuales se puede dividir en sub-fases, que guiarán a los desarrolladores de sistemas a elegir las técnicas más apropiadas en cada momento del proyecto y también a planificarlo, gestionarlo, controlarlo y evaluarlo” (Avison y Fitzgerald, 2006).

El desarrollo de aplicaciones para proveer servicios móviles, difiere del desarrollo de software tradicional en muchos aspectos, lo que provoca que las metodologías usadas para estos entornos móviles, también difieran de las del software clásico (Rahimian y Ramsin, 2008).

Las metodologías difieren ya sea por la cantidad de fases, las técnicas de cada fase, el contenido de la fase o en su base filosófica, todo esto se aplica, dependiendo del contexto de desarrollo, tamaño del proyecto o del equipo de trabajo, cultura organizacional, entre otros aspectos, por lo que, en el caso de los desarrollos móviles, es de vital importancia su selección, para garantizar un producto de calidad.

2.14 Mobile-D

Esta metodología se creó en el año 2004, por un grupo de investigadores finlandeses, pertenecientes al Instituto de Investigación Finlandés. Aunque ya tiene muchos años de vigencia, ésta es muy útil al crear proyectos con éxito, gracias a sus técnicas muy funcionales. Su objetivo principal es alcanzar ciclos de desarrollo rápidos en equipos muy pequeños, trabajando en un mismo espacio físico. Mobile-D es una metodología ágil basado en una mezcla de muchas técnicas conocidas como: extreme programming (XP), Crystal Methodologies y Rational Unified Process (RUP) (Blanco, Camarero, Fumero, Warterski, & Rodríguez, 2009).

El ciclo del proyecto consta de cinco fases: Exploración, Inicialización, Producción, Estabilización y Pruebas del Sistema (Figura 2.9). En general todas las fases (con la excepción de la primera fase de exploración) consta de tres días de desarrollo distintos: planificación, trabajo y liberación. Se le añadirán días para acciones adicionales en casos particulares (Blanco, Camarero, Fumero, Werterski, & Rodríguez, 2009).



Figura 2.9 Fases de la metodología Mobile-D

Fuente: (Escobar y Campaña, 2014).

2.14.1 Fase de exploración

En la fase de exploración se centra la atención en la planificación y a los conceptos básicos del proyecto. Aquí es donde hacemos una definición del alcance del proyecto y su establecimiento con las funcionalidades donde queremos llegar. Una buena planificación es completar al menos la mitad del proyecto. Esta fase también es muy importante porque se puede definir una implementación inicial de la arquitectura en el proceso de desarrollo.

Los objetivos de Exploración son:

- Definición de grupo de interesados necesarios para la planificación y monitoreo del proceso de desarrollo de software del proyecto.
- Definir y acordar objetivos y alcances por el software a desarrollarse.
- Planificación del proyecto relacionado al ambiente y personal

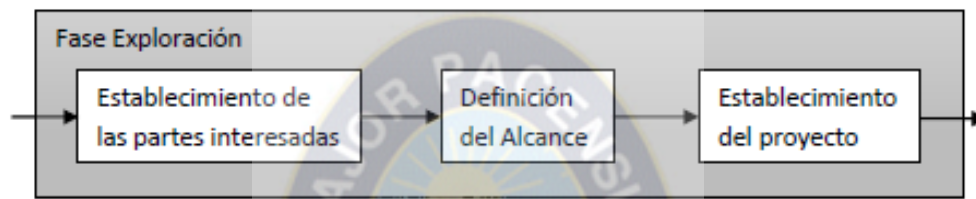


Figura 2.10 Proceso Fase de exploración

2.14.1.1 Definición de partes interesadas

Es una etapa donde todos los relevantes grupos de interés necesitan establecer diferentes tareas para la parte inicial del proyecto donde son definidos por roles y recursos. Por ejemplo, se debe definir diferentes grupos de especialistas como el grupo de arquitectura, desarrolladores de software.

2.14.1.2 Definición de alcance

Es una etapa donde los objetivos y alcances de la parte inicial del proyecto de desarrollo son definidos y en acuerdo con los dueños del producto. Esto incluye requerimientos iniciales para el producto en un tiempo límite del proyecto.

2.14.1.3 Definición del proyecto

Es la etapa para definir el ambiente del proyecto, tanto físico como técnico para la inicialización del proceso de desarrollo, todas las necesidades y requerimientos en este ámbito son definidos en esta etapa.

2.14.2. Fase de inicialización

En la iniciación configuramos el proyecto identificando y preparando todos los recursos necesarios como hemos comentado anteriormente en esta fase la dedicaremos un día a la planificación y el resto al trabajo y publicación. También en esta fase se identifica todas las tareas críticas que se tienen que desarrollar en el proyecto.

2.14.2.1 Configuración del proyecto

- Configuración física y técnica de recursos para el ambiente del proyecto.
- Establecer maneras específicas de comunicación con el cliente.

2.14.2.2 Planificación inicial

- Principalmente entender bien los objetivos del producto para que el equipo pueda tener una base inicial de acuerdo a los requerimientos de arquitectura.
- Refinar la arquitectura y el plan del proyecto.
- Crear un plan para la definición de la arquitectura.
- Preparar un plan para revisar el estado del proyecto, tareas críticas etc.

Las etapas que comprenden esta fase son:

Proyecto Set-Up. El objetivo de la etapa de proyecto Set-Up es:

- Establecimiento de los recursos físicos y técnicos para el proyecto, así como el medio ambiente para el seguimiento del proyecto,
- Entrenar al equipo del proyecto, según sea necesario, y
- Establecer las formas específicas de proyectos de comunicación con el grupo de clientes.

Planificación inicial. El objetivo de la fase de planificación inicial es:

- Obtener una buena comprensión global del producto para el equipo del proyecto en base a los requisitos iniciales y descripciones de línea arquitectura,
- Perfeccionar y descripciones de línea de detalle de la configuración y el plan del proyecto,
- Crear un plan de línea de arquitectura, y
- Preparar planes para comprobar el estado de preparación de las cuestiones fundamentales del desarrollo, tales como el entorno de desarrollo, elementos arquitectónicos importantes y otras entidades externas del software a desarrollar, así como la comunicación entre los elementos y entidades.

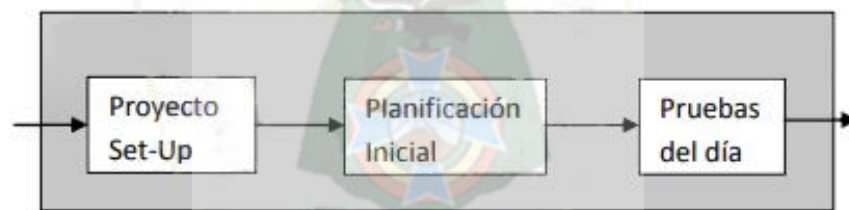


Figura 2.11 Fase de inicialización

2.14.3 Fase de producto

El propósito de esta fase es implementar toda la funcionalidad requerida del producto aplicando ciclos de desarrollos iterativos e incrementales. Los objetivos de esta fase son:

- Implementar la funcionalidad prioritaria requerida por el usuario del producto.
- Enfocarse en la funcionalidad principal permitiéndose múltiples ciclos de desarrollo

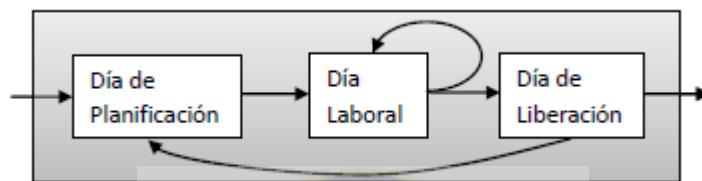


Figura 2.12 Proceso de la fase de producto

Los objetivos de la fase Productización son:

- Implementar la funcionalidad priorizado cliente para el producto.
- Enfoque en la funcionalidad del núcleo fundamental de su ejecución a principios del incremento para permitir que múltiples ciclos de mejora.

Las etapas que comprenden esta fase son:

- Día Planificación. El objetivo de la etapa del día de planificación es para definir el contenido (es decir, historias y tareas) para la iteración.
- Día Laboral. El objetivo del trabajo es la etapa para implementar la funcionalidad decidida de una manera controlada y gestionada.
- Día de liberación. El objetivo de la fase de lanzamiento es el día para verificar y validar la funcionalidad implementada. Normalmente, el día de lanzamiento culmina en un lanzamiento real, pero el lanzamiento no oficial puede ser de manera que se evalúa el producto dentro de los interesados en el proyecto de núcleo.

2.14.4 Fase de estabilización

Después de la fase de producto llega la fase de estabilización en la que se realizan las acciones de integración para enganchar los posibles módulos separados en una única aplicación y a la vez asegurar la calidad de la implementación del proyecto.

Entre los objetivos de estabilización del proyecto tenemos:

- Finalizar la implementación del producto
- Mejorar y asegurar la calidad del producto
- Finalizar la documentación del producto

Proceso El modelo de la fase de Estabilización (figura 2.13) puede llevarse a cabo a través de las siguientes etapas:

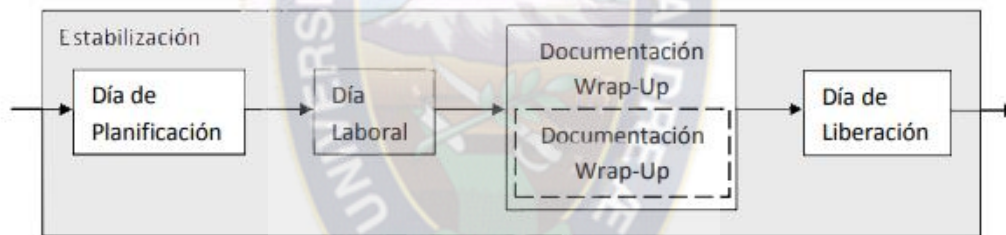


Figura 2.13 Fase de estabilización

2.14.4.1 Etapas

Etapas Día de Planificación. El objetivo de la etapa de día de planificación de la fase Estabilización es definir el contenido (es decir, historias y tareas) para la ejecución de las funciones del producto que quedan y para la mejora de la calidad externa e interna del producto (Refactoring). **Día Laboral.** El objetivo de la etapa de días de trabajo de la fase Estabilización es finalizar la implementación del producto, así como mejorar y asegurar la calidad del producto.

2.14.5 Fase de pruebas y reparación del sistema

El propósito de la fase de pruebas y solución de errores es para verificar si el producto desarrollado implementa y tiene la funcionalidad requerida y definida por el usuario correctamente, también se realiza la detección de errores de funcionalidad los cuales deben ser solucionados. En esta fase no se considera ninguna nueva implementación.

La fase de pruebas y solución de errores tiene por objetivos:

- Probar el sistema basada en la documentación producida para el proyecto.
- Proveer información sobre defectos encontrados
- Diseñar un plan de solución de errores para los defectos encontrados
- Solucionar defectos del sistema

La siguiente figura ilustra las etapas de las cuales la prueba del sistema y la fase de reparación es compuesta por:

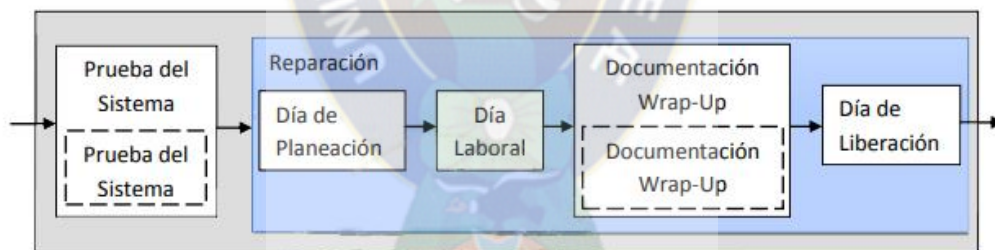


Figura 2.14 Fase de pruebas

Las etapas individuales de la prueba del sistema y la fase del arreglo son:

- La prueba del sistema es una etapa en la que el sistema se prueba como se describe en el modelo de tarea de prueba del sistema. Los defectos encontrados se documentan con el fin de la iteración Reparación.

- Reparación es una variación de la iteración normal; sin embargo, ninguna nueva funcionalidad se implementa y la escala de tiempo puede ser notablemente más corto. La entrada para esta iteración son los defectos encontrados en la fase de prueba del sistema.

- o Día de Planificación. El objetivo de la etapa de día de planificación de la prueba del sistema y la fase de reparación es definir el contenido (es decir, historias y tareas) para la iteración reparación. Los defectos encontrados en la fase de prueba del sistema son insumos para las descripciones de tareas.

- o Día de Trabajo. El objetivo de la etapa de días de trabajo de prueba del Sistema y la fase de reparación es arreglar los defectos encontrados en la fase de prueba del sistema y para finalizar la implementación del producto.

- o Documentación Wrap-Up. El objetivo de la conclusión de la fase de documentación es finalizar los documentos de arquitectura de software, el diseño y la interfaz de usuario. La documentación se ha actualizado para corresponder con los cambios realizados durante la iteración Reparación. La conclusión de la fase de documentación incluye sólo una tarea, la tarea Wrap-Up de documentación, que se puede llevar a cabo utilizando el patrón de tareas, resumen de la documentación.

- o Día de Liberación. El propósito de la etapa del día de lanzamiento es para verificar y validar la funcionalidad y la calidad del conjunto implementado del software y su documentación. El día del lanzamiento culmina en la liberación final de todo el software.

CAPITULO III

MARCO APLICATIVO

La aplicación y todo su desarrollo siguen los lineamientos de la metodología Mobile-D, esta metodología está orientada al desarrollo de aplicaciones móviles, proyectos que cuentan con pocos recursos en su equipo de desarrollo, además de dar más énfasis a los entregables o releases, en este capítulo se describe el desarrollo del prototipo en cada una de sus fases.

3.1 Exploración

Esta fase trata principalmente sobre un análisis previo de la situación, identifica los usuarios involucrados en este proyecto, como también los requerimientos, los stakeholders, las tareas que se realizarán y la planeación del desarrollo del proyecto.

3.1.1 Establecimiento de los usuarios

Según la investigación los usuarios que se encuentran involucrados para el desarrollo de este proyecto son:

3.1.1.1 Estudiante de secundaria

Es la persona que se registra e inicia la aplicación de simulación de pruebas PISA. El usuario debe en principio descargar la aplicación móvil desde la tienda de Google Play, donde estará disponible la app para que pueda ser instalado en los dispositivos móviles.

Luego de haber instalado la aplicación móvil, debe registrarse introduciendo los datos personales del usuario además de introducir sus datos personales, deben introducir información relacionada con su entorno educacional.

3.1.1.2 Administrador

Encargado de monitorear el sistema, es la persona encargada de administrar las calificaciones de las pruebas realizadas por los estudiantes, además de analizar y obtener estadísticas de los datos obtenidos, puede ser un docente, el director de una unidad educativa o personal de la entidad encargada de la organización educativa de un municipio o población.

3.1.2 Establecimiento del proyecto

El proyecto consiste en el desarrollo de una aplicación que ofrezca a los estudiantes de nivel secundario la posibilidad de realizar un simulacro de las pruebas PISA a fin de poner en práctica los conocimientos con los que cuenta el estudiante, brindando a estos, información relacionada con el nivel académico que poseen, y a los docentes o responsables de educación, reportes de estadísticas de información relevante del área social, económica y académica estudiantil, para su respectivo análisis.

3.1.3 Requisitos del sistema

Se define el análisis de los requisitos mínimos que debe cumplir la aplicación móvil para una buena funcionalidad:

- La aplicación móvil se ejecuta en dispositivos móviles con plataforma Android versión 4.4 o superior.
- Es necesaria una conexión a internet.
- Incluir preguntas liberadas de las pruebas PISA pasadas.
- Interacción con un servidor remoto para la centralización de los datos.
- La aplicación desarrollada debe ajustarse de manera automática a diferentes dispositivos móviles inteligentes.

3.2 Inicialización

En esta fase se realizó la preparación y definición de los recursos y herramientas utilizadas en la construcción del producto, nos encargaremos de preparar y verificar todos los asuntos críticos para el desarrollo del proyecto.

3.2.1 Configuración del proyecto

A continuación, se describirán las herramientas utilizadas para la implementación de la aplicación.

3.2.1.1 Android

Plataforma libre y de uso masivo en el mercado de telefonía móvil actual. Las herramientas que se utilizaran para el desarrollo son las siguientes:

- Android Studio versión 2.2.
- SDK de Android API 19 para la integración con las herramientas adecuadas.

3.2.1.2 Firebase

Para hospedar el servicio REST y la base de datos, se hace uso de Google Firebase, que es la plataforma en la nube de Google, esta plataforma permite alojar una gran variedad de sistemas y servicios, garantiza una alta disponibilidad.

- La base de datos está alojada en el servicio de storage en Firebase.

3.2.2 Requerimientos iniciales

De acuerdo a la definición de los grupos de interés y el establecimiento del proyecto, se identifican los requerimientos iniciales.

- Guardar datos del usuario.
- Guardar datos del administrador.
- Desplegar preguntas por pantalla incluyendo opciones de respuesta,

- Guardar base de datos de preguntas y respuestas de las pruebas PISA.
- Guardar las notas de los resultados obtenidos.
- Registrar los resultados obtenidos, para estadísticas y posterior análisis.
- Mostrar graficas de comportamiento de los resultados.
- Generar reportes e informes de los resultados obtenidos.

3.2.3 Historias de usuario

Las historias de usuario nos permiten entender la especificación de requisitos del sistema. Se especifica la Historia de Usuario, la cual brinda información oportuna y necesaria.

Historia de usuario		Registrar datos del usuario en el sistema
Numero	Tipo	
1	Desarrollo(BackEnd)	
Descripción	Generar los servicios para poder registrar a los usuarios(nombre, apellidos, dirección, sexo, tipo de institución educativa),	
Prioridad	Alta	
Esfuerzo	Estimado	20
	Real	22
Dificultad	Moderada	

Tabla 3.1 Historia de Usuario 1

Tarea		Generar la base de datos para almacenar los datos solicitados
Numero	Tipo	Notas
1.1	Desarrollo	
Dificultad	Antes	3
	Después	4
Descripción		
Se debe diseñar una base de datos que permita almacenar los datos básicos del usuario de la aplicación (nombre, fecha de nacimiento, edad, domicilio, sexo), ya sea este estudiante o administrador de establecimiento.		

Tabla 3.2 Tarea 1.1: Generación de la base de datos

Historia de usuario		Validar los datos de los usuarios al ingresar a la aplicación
Numero	Tipo	
2	Desarrollo(BackEnd, frontEnd)	
Descripción	Generar los servicios para poder validar a los usuario tanto estudiantes como administradores.	
Prioridad	Alta	
Esfuerzo	Estimado	25
	Real	22
Dificultad	Moderada	

Tabla 3.3 Historia de Usuario 2

Tarea		Consulta a la base de datos para validar los datos del usuario
Numero	Tipo	Notas
2.1	Desarrollo	
Dificultad	Antes	3
	Después	4
Descripción		
Se debe diseñar una base de datos que permita almacenar los datos básicos del usuario de la aplicación (nombre, fecha de nacimiento, edad, domicilio, sexo), ya sea este estudiante o administrador de establecimiento.		

Tabla 3.4 Tarea 2.1: consulta a la base de datos

Historia de usuario		Realizar la prueba PISA
Numero	Tipo	
3	Desarrollo(BackEnd, frontEnd)	
Descripción	Realizar la prueba PISA, en el dispositivo en el tiempo determinado por la aplicación	
Prioridad	Alta	
Esfuerzo	Estimado	40
	Real	40
Dificultad	Moderada	

Tabla 3.5 Historia de Usuario 3

Tarea	Responder las preguntas realizadas por la aplicación	
Numero	Tipo	Notas
3.1	Desarrollo	
Dificultad	Antes	3
	Después	4
Descripción		
Se registra las respuestas dadas por el estudiante para su posterior ponderación y despliegue al usuario.		

Tabla 3.6 Tarea 3.1 Registro de respuestas del estudiante

Historia de usuario	Visualizar resultados y estadísticas	
Numero	Tipo	
4	Desarrollo(frontEnd)	
Descripción	El administrador o responsable en educacion, podrá visualizar el resultados obtenido por los estudiantes.	
Prioridad	Alta	
Esfuerzo	Estimado	40
	Real	40
Dificultad	Moderada	

Tabla 3.7 Historia de Usuario4

Tarea	Desplegar estadísticas de los resultados obtenidos	
Numero	Tipo	Notas
3.1	Desarrollo	
Dificultad	Antes	3
	Después	4
Descripción		
Se despliega de forma amigable y agradable al usuario, la información resultante obtenida después del proceso de evaluación.		

Tabla 3.8 Tarea 3.1 Registro de respuestas del estudiante

3.2.4 Análisis de requerimientos

En base a los requerimientos iniciales se han identificado los diferentes módulos con los que cuenta la aplicación, dividiéndose en los siguientes módulos:

- Modulo Estudiante: Estudiante de secundaria del sistema de educación ya sea privado, público o de convenio.
- Modulo Administrador: Persona en cargo jerárquico ya sea en un colegio o en un ente que norme el servicio educativo.
- Módulo de exámenes, que es el más importante de todos los módulos, ya que permite recoger las respuestas del estudiante a ser evaluado.
- El módulo de resultados, donde se despliega las estadísticas de los resultados obtenidos.

3.2.5 Arquitectura de la aplicación

El esquema principal posee como participantes tanto a estudiantes como docentes quienes interactuarán directamente con la aplicación ya sea en la resolución de la prueba y posterior toma de datos, como en la visualización de los resultados de la misma, además de incorporar un servidor donde se almacenarán los datos de los participantes en la nube a través del servicio de almacenamiento de datos de firebase, permite la posibilidad de visualización de la información almacenada, en diferentes tipos de dispositivos independientemente de la arquitectura.

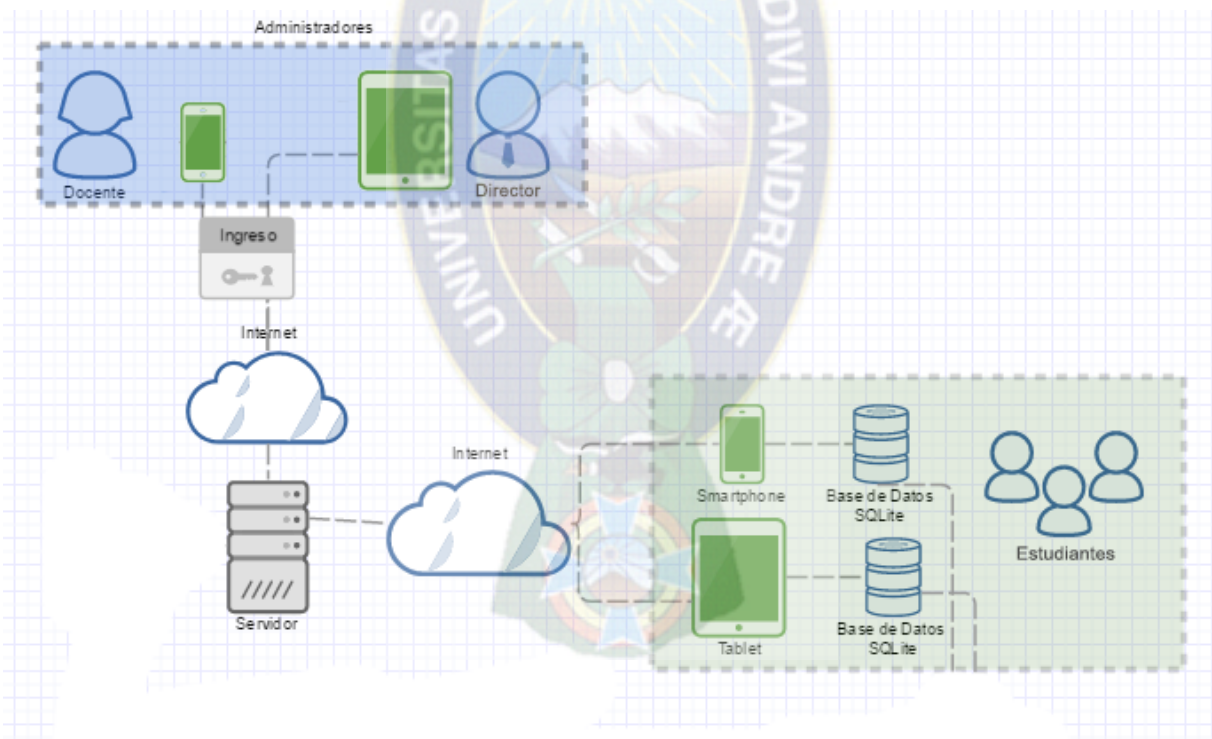


Figura 3.1: Diagrama de contexto

3.2.6 Planificación de fases

FASE	ITERACIÓN	DESCRIPCIÓN
Exploración	Iteración 0	Establecimiento de interesados, definición del alcance, establecimiento del proyecto.
Inicialización	Iteración 0	Configuración del entorno, formación, establecimiento del medio de comunicación con el cliente, planeamientos iniciales y definición de los requerimientos
Producción	Iteración de módulo de base de datos	Diseño y desarrollo de las bases de datos en SQLite
	Iteración de módulo de simulador de pruebas	Carga las pruebas habilitadas para el uso de los usuarios y permite que puedan desplazarse por cada pregunta seleccionando una respuesta correcta
	Iteración de módulo de reportes de resultados.	Implementar el módulo de reporte de resultados, refinamiento de interface, generación de ejecución de pruebas de aceptación.
Estabilización	Iteración de módulo de estabilización	Se cumplirán las pruebas determinadas para la aplicación.
	Iteración de módulo de reporte de resultados.	Refactorización del módulo de reporte de resultados, refinamiento de interface, generación de ejecución de pruebas de aceptación.
Pruebas	Iteración pruebas del sistema	Se cumplirán las pruebas determinadas para la aplicación.

Tabla 3.7 Planificación de fases

3.2.7 Diagrama de casos de uso

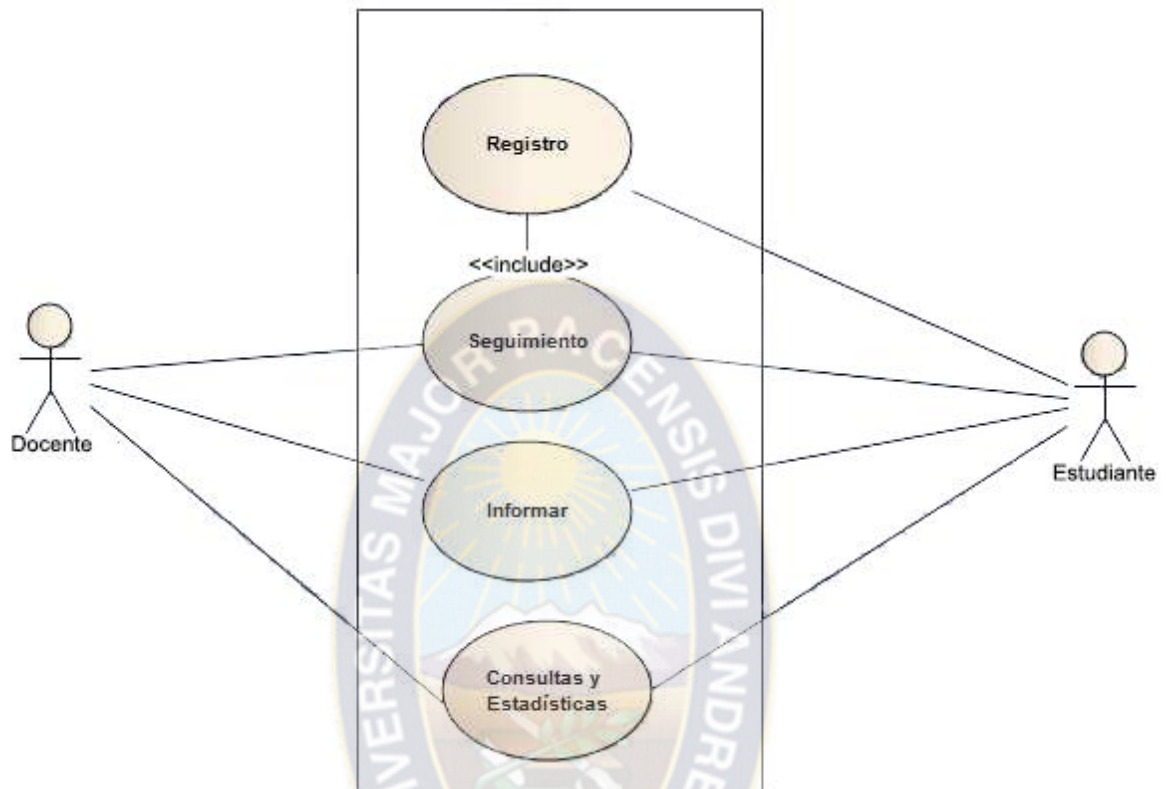


Figura 3.2 Diagrama de casos de uso

3.3 Fase de producción

La fase de producción trata del desarrollo de la aplicación en su totalidad. A partir de las definiciones realizadas en la fase de inicialización es que procede con la implementación de las interfaces de usuario definidas durante la fase de inicialización, la creación de las tablas de la base de datos local, la implementación de la lógica encargada de la calificación de las pruebas y la presentación de los resultados y su posterior almacenamiento tanto de forma local como en el servidor de firebase. La metodología recomienda seguir los pasos de planificación, trabajo y liberación.

3.3.1 Implementación de la aplicación

3.3.1.1 Implementación de la interfaz de usuario

La primera tarea asignada para la fase de producción es el diseño de la interfaz de usuario. Inicialmente se debe crear un nuevo proyecto en Android Studio, que provee el espacio que se precisa para proceder con interfaz (Fig. 3.3.). Donde, empleando campos de texto, *labels* y botones es que se diseñan las interfaces de usuario para la aplicación.

3.3.1.2 Pantallas de la aplicación



Figura 3.3: Pantalla inicial

Es la primera pantalla de la aplicación (Fig. 3.3.), se presenta la bienvenida a la aplicación, da acceso a la aplicación, como estudiante que va a realizar las pruebas y posteriormente para visualizar estadísticas y resultados, además de un botón de salida.

Si el usuario ingresa por primera vez a la aplicación debe registrarse, haciendo clic en el boton registrar y proceder al llenado de sus datos personales (Figura 3.4).

The image shows a mobile application registration screen titled "REGISTRO". At the top, there is a status bar with the time 6:00 and icons for Wi-Fi and battery. Below the title, there is a cursor icon and a "Ci" label. The main form area contains several input fields: "NOMBRES", "APELLIDOS", "EDAD", and "Ingreso mensual aprox.". Below these is a section for "TIPO DE U.E.:" with three radio button options: "Privado", "Fiscal" (which is selected), and "Convenio". Underneath is a "SEXO:" section with two radio button options: "Masculino" (selected) and "Femenino". At the bottom of the form, there are two buttons: "GUARDAR" and "CANCELAR". A large, semi-transparent watermark of the University of the Pacific (Universidad Mayor Pacensis Divi Andre) is visible in the background of the form area.

Figura 3.4 Pantalla de registro

Una vez ingresando a la aplicación podemos escoger entre una de varias opciones entre aprender sobre PISA y conceptos generales sobre la aplicación, realizar la prueba y ver videos interactivos sobre la realizacion de la prueba. **Figura 3.5** Menu principal

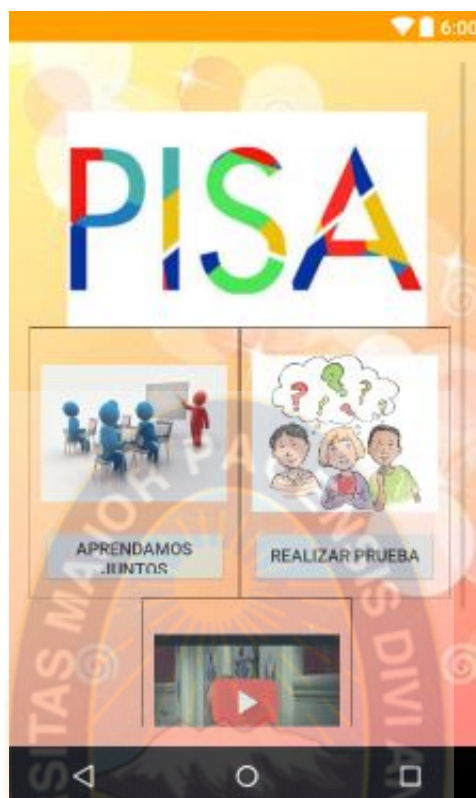


Figura 3.5 Menu principal

Dentro del menu tenemos diferentes opciones entre las cuales podemos acceder al area de informacion sobre las pruebas pisa, conocer su historia, la forma de evaluacion, la presencia a nivel internacional y todo lo referido con las pruebas academicas, como se muestra en la figura 3.6

b

Figura 3.6 Pantalla de informacion sobre PISA

Si escogemos la opcion “realizar la prueba”, accedemos a al menu de opciones de prueba, donde podemos seleccionar el modo de realizar la prueba, en modo normal o en modo por tiempo cronometrado, cada uno con características diferenciadas dándole mayor interactividad al aplicativo, para el desarrollo de la prueba, una vez de terminada la misma se

accede a la pantalla donde se puede visualizar los resultados, sobre una base de 100 puntos con la información de fecha y hora.



Figura 3.7 Menu de la prueba

Después que la organización de PISA realiza la prueba en los diferentes países asociados y publica los resultados de los mismos, se procede a la liberación de las preguntas de la prueba, siendo de dominio público. Las preguntas empleadas para el desarrollo de la aplicación android fueron obtenidas de la página web de PISA.

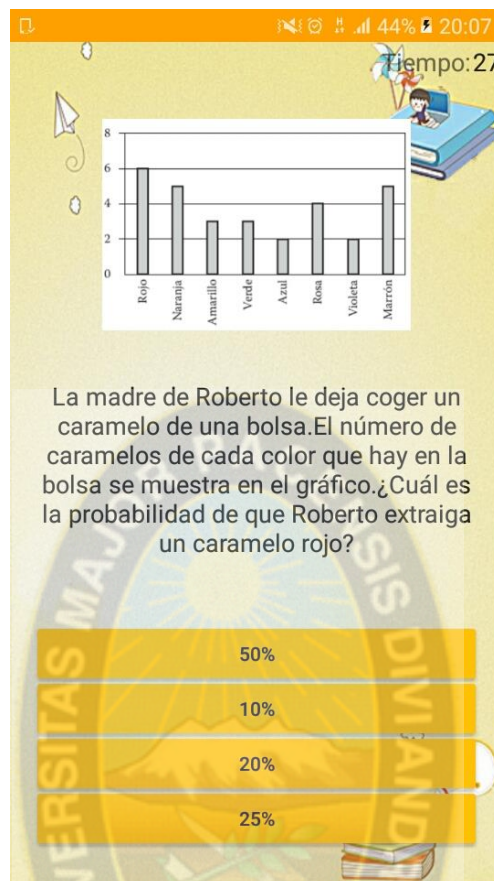


Figura 3.8 Realización de la prueba

PISA se enfoca en 3 áreas principales del saber: matemática, lectura y ciencias. En la aplicación se presenta una miscelánea de preguntas de las 3 áreas, las cuales son desplegadas de forma aleatoria, evitando que se repitan las mismas o se presenten en orden similar.

Una vez escogido el tipo de modalidad procedemos a desarrollar la prueba, la cual consta de un número determinado de preguntas a responder, en forma de selección de una respuesta de entre 4 posibilidades (figura 3.8), después de seleccionar la misma, se alerta al usuario si la respuesta estuvo correcta o incorrecta (Figura 3.9), si el usuario desea salir antes de finalizar la prueba se le alerta de la finalización de la misma, registrándose el resultado obtenido hasta el momento.

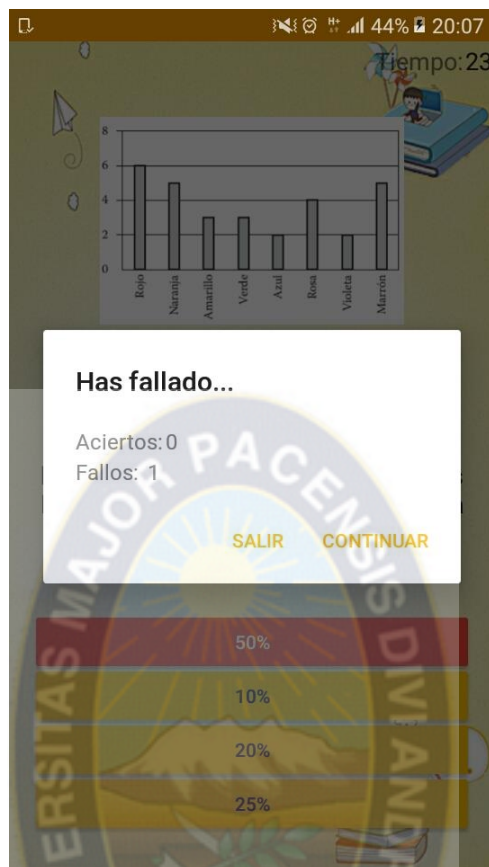


Figura 3.9 Pantalla de fallo de respuesta

Las modalidades de realizar la prueba ya sea modo normal o modo por puntos, brindan una experiencia divertida y en forma de reto de realización de la misma, de esta forma se intenta evitar el estrés o la monotonía que representa el desarrollo de un examen o prueba.

Cuando el estudiante logra acertar una respuesta se le presenta una pequeña ventana donde puede visualizar un mensaje confirmando su resultado (Figura 3.10)

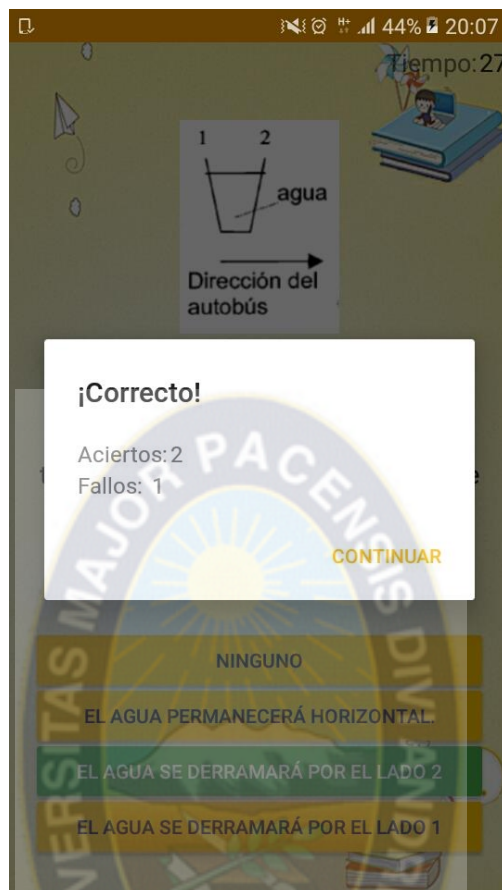


Figura 3.10 Pantalla de acierto de respuesta

Una vez finalizada la respuesta de la totalidad de preguntas o en otro caso de la cancelación de la prueba, la aplicación realiza el cálculo de la nota obtenida, guardando en la base de datos local la información personal del estudiante que realizó la prueba, la cantidad de respuestas correctas y erradas, la hora y fecha de realización.

Se presentan los resultados en una pantalla de la aplicación (figura 3.11) y de forma complementaria, envía los datos al servidor de almacenamiento de firebase, sincronizando los resultados en el momento que el dispositivo cuente con acceso a internet, permitiendo de esta

forma el posterior acceso a los resultados guardados en diferentes tipos de dispositivos o programas.

Los resultados pueden ser borrados de la base de datos local, pero se encuentran respaldados en el servicio de almacenamiento de firebase.



LIMPIAR RESULTADOS	
Aciertos: 4	05-06-2018 03:37
Fallos: 6	
Aciertos: 2	05-06-2018 03:37
Fallos: 8	
Aciertos: 3	05-06-2018 20:08
Fallos: 7	

Figura 3.11 Despliegue de resultados

3.3.2 Modelo de base de datos

El modelo de base de datos desarrollado para su posterior implementacion consta del siguiente diagrama:

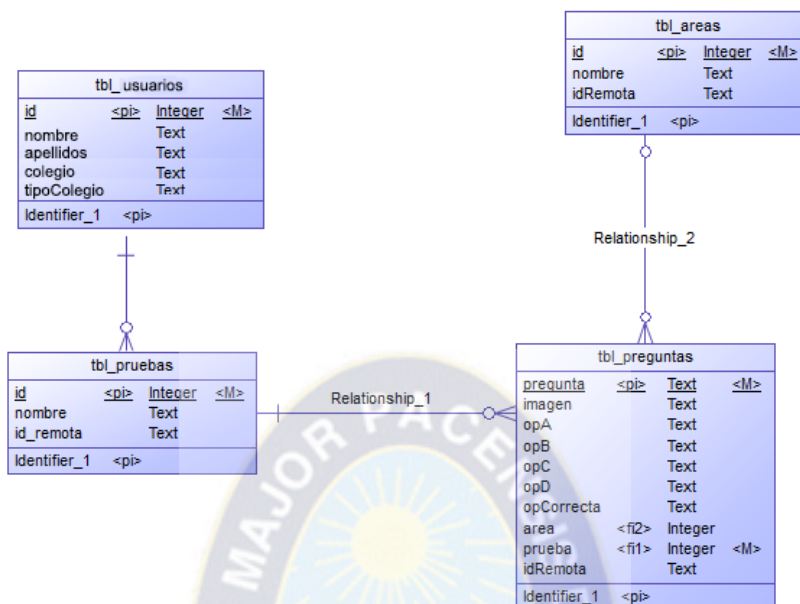


Figura 3.12: Diagrama relacional

3.4 Fase de estabilización

En esta fase se finaliza el desarrollo de la plataforma y se asegura la calidad del producto desarrollado. Se procede a estabilizar la correcta funcionalidad del prototipo de la aplicación de simulación de exámenes PISA, se continúa aplicando el desarrollo del ciclo iterativo.

Los errores que se encuentran durante la fase de producción, así como los que surgen durante las primeras ejecuciones van siendo corregidos poco a poco.

En esta fase se llegó a establecer la integración de los módulos desarrollados de forma separada en una única aplicación, realizando las siguientes tareas:

- Se mejoró los estilos de interfaz gráfica de la aplicación Android de simulación de pruebas PISA, dándole una mejor apariencia y presentación al usuario final y al administrador respectivamente.
- La aplicación Android, funciona de manera offline parcialmente, para la funcionalidad total y adecuada es necesario contar con conexión a internet.
- Para la integración de la aplicación móvil, se tuvo que conectar el menú con los distintos actividades creados, como también la interacción entre escenas. También se hizo una verificación de cada módulo de la aplicación haciendo énfasis en algunos cambios necesarios para que la aplicación móvil funcione correctamente.

3.5 Pruebas del sistema

Para el testeo de la aplicación Android, se determina la funcionalidad de cada módulo construido, realizando las pruebas con datos reales. Siendo el criterio más importante a tomarse en cuenta es la obtención y análisis del comportamiento de las respuestas según el nivel socioeconómico y educativo de los estudiantes, ya que este determina la funcionalidad total de la aplicación Android desarrollada, se realizan las pruebas comparando con datos teóricos.

Para cada pantalla se prueba lo siguiente:

- Datos válidos.
- Datos inválidos.
- Valores límite.

Para las pruebas de tiempo de carga se tomó en cuenta los siguientes criterios:

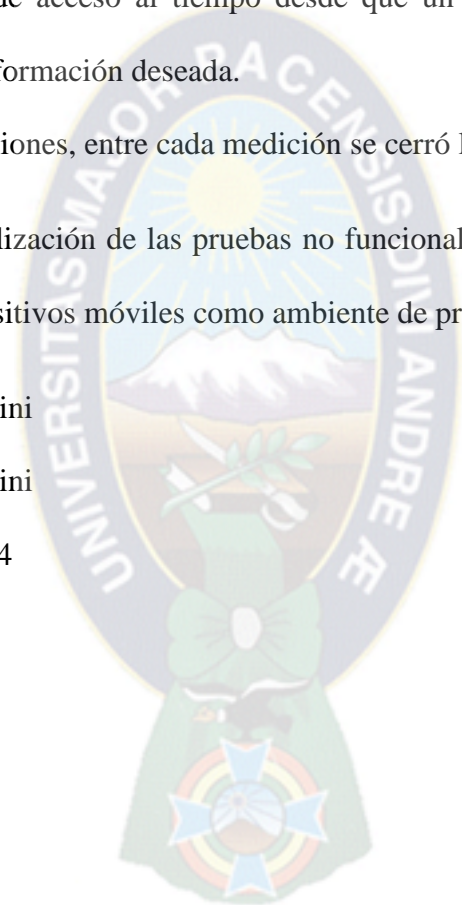
- Tomamos los requerimientos más importantes.
- Se realizaron las pruebas a 10 usuarios, repitiendo varias veces el proceso. Se espera que cada usuario simulado termine de realizar la petición.

Para las pruebas de tiempo de acceso se tomó en cuenta los siguientes criterios:

- Se considera tiempo de acceso al tiempo desde que un usuario abre una aplicación y recibe totalmente la información deseada.
- Se realizaron 10 mediciones, entre cada medición se cerró la respectiva aplicación.

Para el momento de la realización de las pruebas no funcionales y funcionales se optó por el uso de los siguientes dispositivos móviles como ambiente de pruebas:

- Samsung Galaxy S3 mini
- Samsung Galaxy S4 mini
- Samsung Galaxy ACE4



CAPÍTULO IV

Análisis y resultado

4.1. Experimentación

Se considera la hipótesis planteada en el primer capítulo, como recurso de análisis para la prueba del prototipo de la aplicación Android de simulación de pruebas PISA para estudiantes de nivel secundario.

4.2 Evaluación De La Variable Dependiente

Variable dependiente: Estudiantes motivados después de usar la aplicación.

Indicador: % Porcentaje de aprobación

Instrumento: Se utiliza la escala likert, verificando que tipo de relación tiene una muestra aleatoria de tamaño n y su opinión acerca de la utilidad de la herramienta, si los resultados obtenidos son dependientes o independientes, es decir, que si estas fueran dependientes, las variables están relacionadas, por otra parte si los resultados muestran que son independientes es posible afirmar que no existe relación entre ellas.

A continuación se formulan las siguientes hipótesis H_0 como hipótesis nula y H_1 como hipótesis alternativa.

H_A = “La aplicación Android de simulación de pruebas PISA para estudiantes de nivel secundario incentiva a los estudiantes a mejorar su nivel académico”

Es también necesario contemplar una hipótesis nula como parte del análisis:

H_0 : “La aplicación Android de simulación de pruebas PISA para estudiantes de nivel secundario, no incentiva a los estudiantes a mejorar su nivel académico”.

4.2.1. Determinación de la población

La población que se tomó en cuenta para esta investigación son estudiantes de secundaria. Donde la muestra realizada fue de 50 estudiantes de los últimos cursos de secundaria entre los 16 y 17 años de edad.

4.2.2. Proceso

Tomando como población de estudio, los usuarios de la muestra, se verifica si existe correlación entre el grado o porcentaje de estudiantes motivados o incentivados después del uso de la herramienta de simulación. Se procedió a que cada estudiante se autoevalúe y ponga una puntuación al nivel de conocimientos que crea tener en ese momento, posteriormente se les hizo realizar la prueba en el prototipo de la aplicación. Los resultados obtenidos presentan la siguiente distribución de intervalos de puntuación.

Intervalo de puntuación	Cantidad de estudiantes
0% - 25%	7
26% - 50%	24
51% - 75%	13
76% - 100%	6

Tabla 4.1 Distribución de intervalos de puntuación de estudiantes

La puntuación obtenida de la mayoría de los estudiantes aproximadamente el 61% no llega a obtener ni la mitad de puntuación global de la prueba. Posteriormente se procedió al llenado de la encuesta. La evaluación fue realizada con preguntas cerradas, midiéndose con una escala del 1 al 5, siendo '1' muy mala y '5' muy buena.

4.3 Resultados de la encuesta

Pregunta 1: ¿La aplicación cuenta con una interfaz dinámica agradable a la vista?

Opcion	Frecuencia
1	0
2	0
3	25
4	10
5	15

Tabla 4.2 Resultados pregunta 1

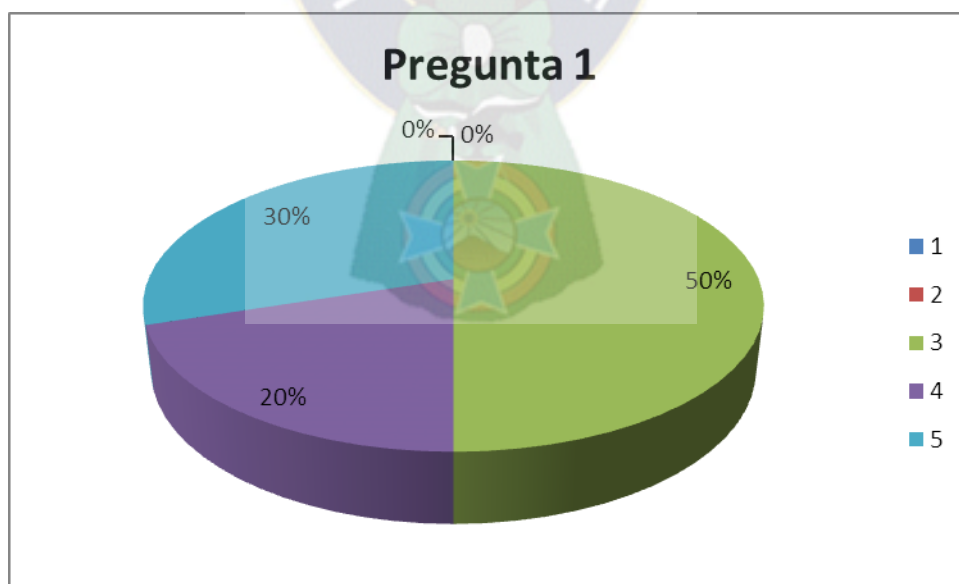


Figura 4.1 Resultados pregunta 1

Pregunta 2: ¿Crees que las preguntas de la aplicación fueron adecuadas?

Opcion	Frecuencia
1	0
2	0
3	25
4	10
5	15

Tabla 4.3 Resultados pregunta 2

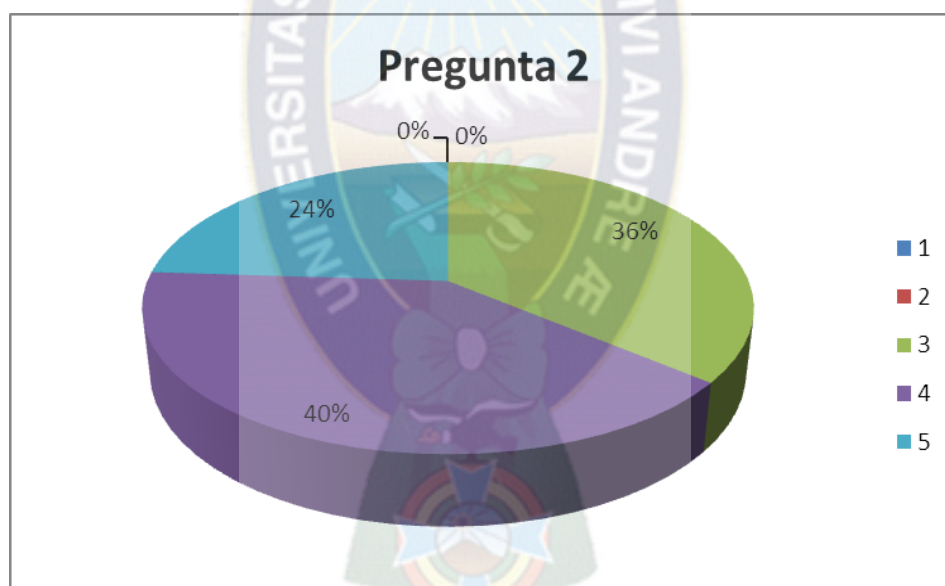


Figura 4.1 Resultados pregunta 2

La mayoría de los estudiantes cree que las preguntas fueron adecuadas a su nivel de conocimiento actual, en el grado en el que se encuentran y a la edad que tienen.

Pregunta 3: ¿Estarías de acuerdo en incorporar este tipo de pruebas en nuestro país?

Opcion	Frecuencia
1	0
2	0
3	20
4	15
5	15

Tabla 4.4 Resultados pregunta 3

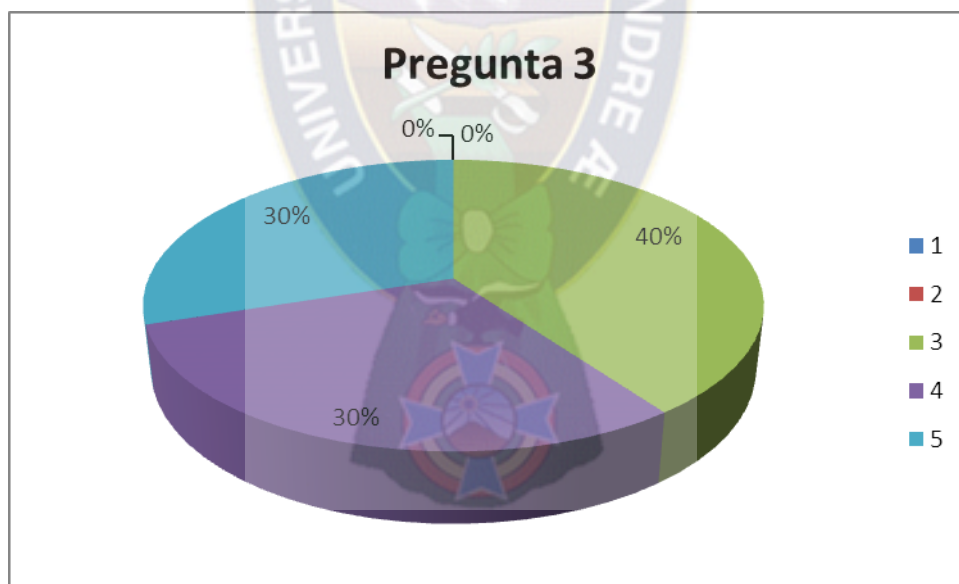


Figura 4.3 Resultados pregunta 3

La mayoría de los estudiantes aprueba este tipo de pruebas para poder evaluar el nivel de conocimiento que tiene.

Pregunta 4: ¿Estas de acuerdo que la aplicación permite establecer el grado de conocimiento que poseen los estudiantes de secundaria?

Opcion	Frecuencia
1	0
2	0
3	25
4	10
5	15

Tabla 4.5 Resultados pregunta 4

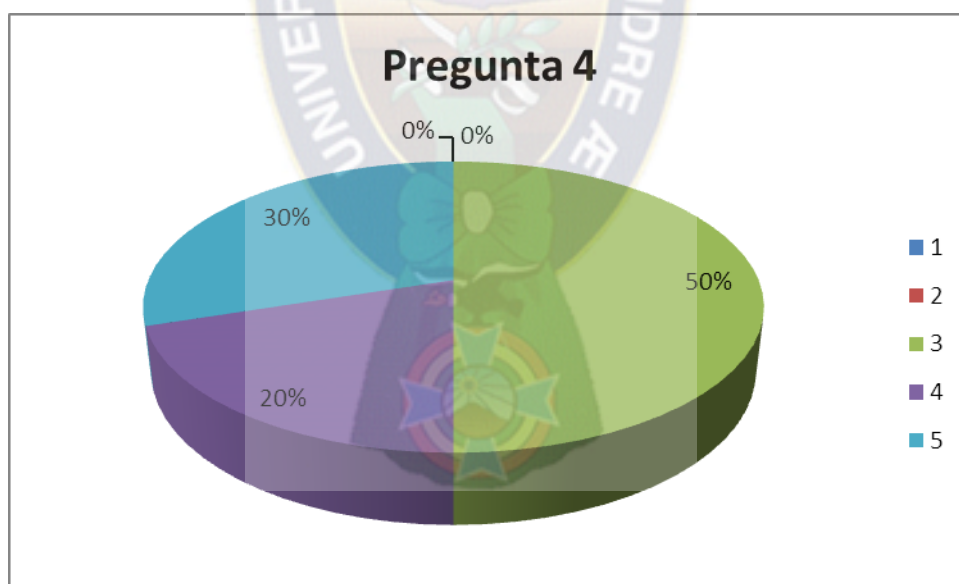


Figura 4.4 Resultados pregunta 4

La mayoría de los estudiantes esta de acuerdo que este tipo de prueba, permite establecer el grado de conocimiento academico que posee un estudiante.

Pregunta 5: ¿Cree usted que la prueba realizada a través de la aplicación android, le incentivó a mejorar su nivel academico y de conocimientos?

Opcion	Frecuencia
1	0
2	0
3	20
4	12
5	18

Tabla 4.6 Resultados pregunta 5

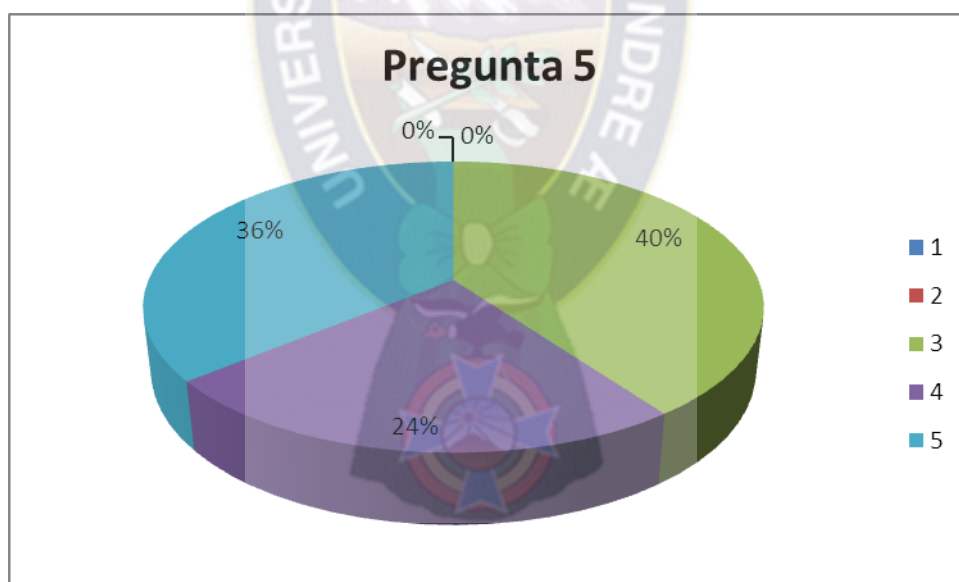


Figura 4.5 Resultados pregunta 5

Los estudiantes en su mayoría afirman que el metodo de realizar la prueba a traves de una aplicaci3n android, hace mas accesible y menos tediosa el desarrollo de un examen.

Pregunta 6: ¿Volvería a usar la aplicación?

Opcion	Frecuencia
SI	0
NO	50

Tabla 4.7 Resultados pregunta 6

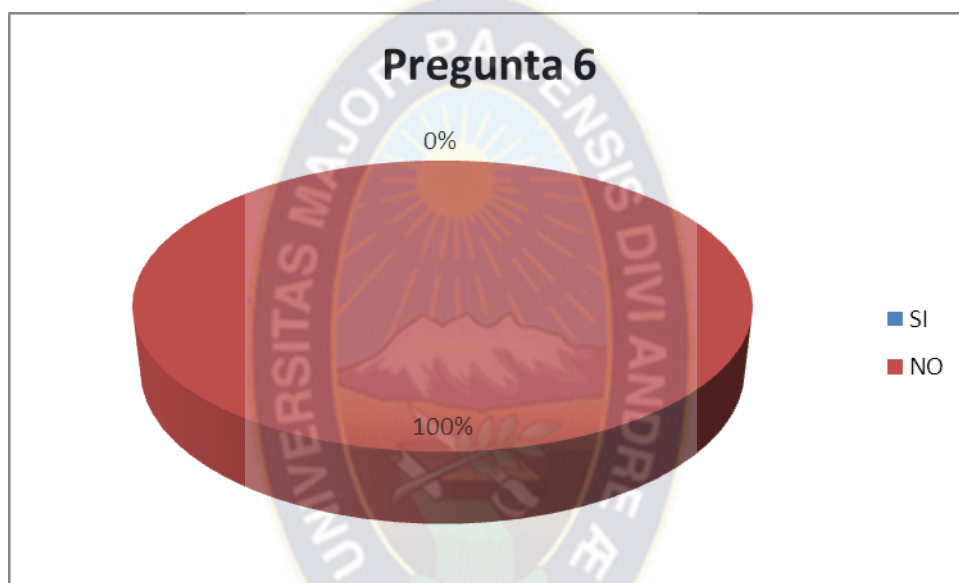


Figura 4.6 Resultados pregunta 6

4.4 Análisis de resultados de la evaluación

Se tiene un rango de aceptación muy elevado:

- 40% de estudiantes aceptan que la aplicación les motivo
- 24% de los estudiantes esta de acuerdo que la aplicación incentiva a mejorar el grado de conocimiento que poseen
- Y un 36% esta muy de acuerdo que la aplicación les motiva a adquirir mas conocimiento y mejorar el nivel academico que poseen actualmente

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

El presente trabajo de tesis demuestra que es posible el desarrollo de una aplicación Android de simulación de pruebas PISA para el nivel secundario que sigue la metodología Mobile-D, incentivando a los estudiantes participantes a mejorar su nivel académico adquiriendo más conocimientos y motivándolos a ser más competitivos.

De la investigación se resaltan los siguientes puntos:

Al concluir la presente investigación, se establecen las siguientes conclusiones:

- Se destaca los resultados obtenidos en la prueba de aceptación, que con un 60% de estudiantes muy motivados no solo comprobaron el funcionamiento de la aplicación, sino que también demostraron que la aplicación android contribuye en un grado significativo al conocimiento de las disciplinas más importantes de la educación secundaria.
- Haciendo uso de la Metodología Mobile-D, se permitió al presente trabajo organizar de manera eficiente en la parte de la fase de Análisis y Diseño Inicial del prototipo de la aplicación.
- El banco de preguntas y respuestas fue obtenido de las preguntas liberadas anualmente por PISA, contando de tal manera con un alto contenido pedagógico, investigativo en el área sociológica y psicológica.
- Los resultados obtenidos por los estudiantes posteriormente finalizada la prueba, permitieron establecer la situación educativa en la que se encuentran tanto ellos como

sus centros educativos, con algunas deficiencias en unas áreas como también fortalezas en otras.

- La implementación y aplicación de plataformas tecnológicas alternativas a la educación tradicional, permite incentivar y mejorar la manera de impartir conocimiento a los estudiantes, especialmente a los de unidades educativas del sistema nacional.
- Incentivar y promover la adopción de algún sistema de medición y parametrización del nivel educativo del sistema nacional de educación, ya que de esta forma podremos estar conscientes de nuestra realidad y proponer políticas educativas que se ajusten a las necesidades del mercado laboral actual, gran parte de los jóvenes bachilleres no están preparados para la educación universitaria, por tanto debería poner más énfasis en esta etapa del desarrollo de los estudiantes.

5.2 Recomendaciones

Son numerosas las nuevas líneas de investigación que se van perfilando conforme se lleva a cabo la tesis, las que se mencionan a continuación son de cierta manera recomendaciones a tomar en cuenta si se desea dar continuidad al proyecto.

- Integrar un módulo de evaluación para aquellos usuarios que tienen algún conocimiento previo en el tema, para recomendar el nivel con el cual deberían comenzar a realizar las pruebas.

- Incluir más temas lecciones a fin de dar una cobertura más amplia a las disciplinas del conocimiento en la etapa secundaria.

- Incorporar medios adicionales para dar seguimiento al progreso del usuario.

- Emplear el uso de contenido multimedia para cada lección con el propósito de llamar la atención del estudiante y hacer la aplicación más interactiva.

- Se alienta a emplear a aplicaciones android como herramienta de capacitación en diversas áreas, como herramienta auxiliar de materia, y otros ámbitos en los cuales pueda ser de utilidad.

- Una necesidad específica es subir el juego a Google Play para que esté disponible en toda la web, sin embargo la aplicación está en etapa de prototipo una vez pasada esta etapa será publicada.



Bibliografía

OCDE (2015). *PISA 2015. Explotación de los ficheros de microdatos anonimizados*. París: OECD Publishing.

Informe Pisa *Pisa en español* <http://www.oecd.org/pisa/pisaenespaol.htm>

Diario El Pais Resultados del Informe PISA

https://elpais.com/elpais/2016/12/05/media/1480958752_164797.html

BBVA ¿Qué es el informe PISA? Cinco claves sobre el examen más influyente del mundo

<https://www.bbva.com/es/informe-pisa-cinco-claves-examen-mas-influyente-del-mundo/>

INEE (2016). *Marcos teóricos PISA 2015*. Madrid: Ministerio de Educación, Cultura y Deporte. <http://www.mecd.gob.es/inee/publicaciones/estudios-internacionales.html>

INEE (2016). *Marcos teóricos PISA 2015*. Madrid: Ministerio de Educación, Cultura y Deporte. <http://www.mecd.gob.es/inee/publicaciones/estudios-internacionales.html>

Munoz C. (1998): Como elaborar y asesorar una investigación de tesis. Mexico: 1ra Ed. Prentice Hall Hispanoamericano S.A.

Centuria (2010). Grupo centuria:“Seminario de desarrollo de proyectos de investigación”. Mexico

Larman, C. (1999). *UML y Patrones, Introducción al Análisis y Diseño Orientado a Objetos* (Primera ed.). México: Prentice Hall.

Txema Rodríguez, 2011, Métodos Aplicables para el desarrollo de aplicaciones móviles,
<http://www.genbetadev.com/desarrolloaplicacionesmoviles/>

VTT, 2008 Agile (Mobile-D), 2018, <http://agile.vtt.fi/index.html>

Kelly Waters, 2007, 10 Principios de desarrollo Ágil, <http://www.allaboutagile.com/what-is-agile-10-key-principles/>

Miguel Zabalza, 2001, Didáctica Estudiantil Infantil

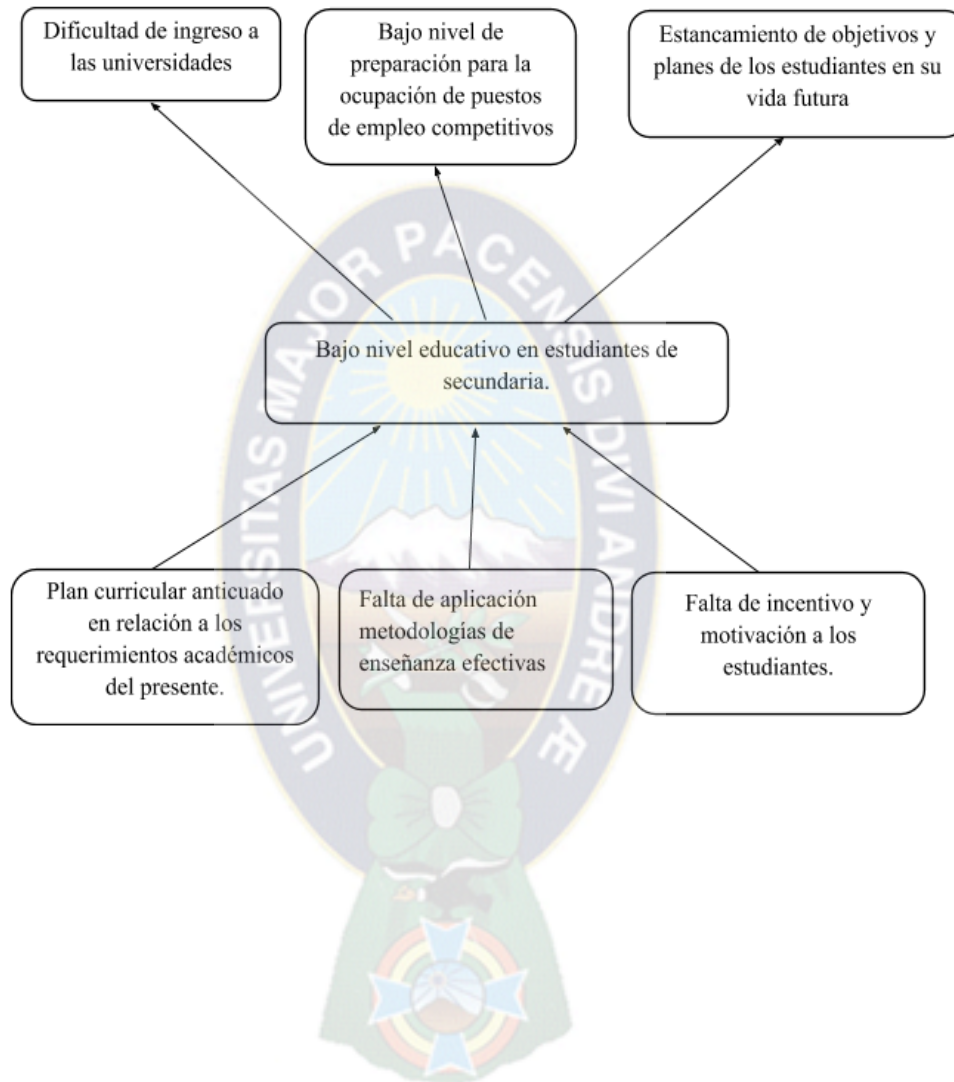
<https://oposicionesprofesor.wordpress.com/2008/02/13/la-programacion-didactica-en-educacion-primaria>

Gomez Oliver, Salvador. (2011). Manual Programación Android V2, Version 2.0, Noviembre 2011, Autor.



ANEXOS



ANEXO 1**Árbol de problemas**

ANEXO 2

Elementos	Muy Aceptable	Aceptable	Poco Aceptable	Nada Aceptable
Facilidad de uso				
Contenidos				
Accesibilidad de los datos				
Atractivo				
Sencillo				
Manejo fácil				
Guía de aprendizaje				
Motiva el aprendizaje				
Comprensible				
Manejable				
Adecuación al currículo				
Apto para adolescentes				
Apoyo visual				
Interactividad				
Facilidad de inicio e salida				
Mejor comprensión con videos				