



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**DESARROLLO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES INTELIGENTES
ENFOCADA A LA MEJORA DE LA SALUD DE LOS USUARIOS: HEALTHYFOOD
IMPLEMENTANDO EL MODELO DE ARQUITECTURA DE 4+1 VISTAS**

Angel Estuardo Cajas Ochoa

Asesorado por el Ing. Edgar Estuardo Santos Sutuj
Coasesorado por el Ing. Eddy Orlando Méndez Paz

Guatemala, mayo de 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DESARROLLO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES INTELIGENTES
ENFOCADA A LA MEJORA DE LA SALUD DE LOS USUARIOS: HEALTHYFOOD
IMPLEMENTANDO EL MODELO DE ARQUITECTURA DE 4+1 VISTAS**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

ANGEL ESTUARDO CAJAS OCHOA

ASESORADO POR EL ING. EDGAR ESTUARDO SANTOS SUTUJ
COASESORADO POR EL ING. EDDY ORLANDO MÉNDEZ PAZ

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, MAYO DE 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Pedro Antonio Aguilar Polanco
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pablo Christian de León Rodríguez
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Raúl Eduardo Ticún Córdova
VOCAL V	Br. Henry Fernando Duarte García
SECRETARIA	Inga. Lesbia Magalí Herrera López

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Pedro Antonio Aguilar Polanco
EXAMINADOR	Ing. César Augusto Fernández Cáceres
EXAMINADOR	Ing. José Ricardo Morales Prado
EXAMINADOR	Ing. Marlon Francisco Orellana López
SECRETARIA	Inga. Lesbia Magalí Herrera López

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**DESARROLLO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES INTELIGENTES
ENFOCADA A LA MEJORA DE LA SALUD DE LOS USUARIOS: HEALTHYFOOD
IMPLEMENTANDO EL MODELO DE ARQUITECTURA DE 4+1 VISTAS**

Tema que me fuera asignado por la dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha de octubre de 2014.



Angel Estuardo Cajas Ochoa

Guatemala, 6 de Mayo de 2015

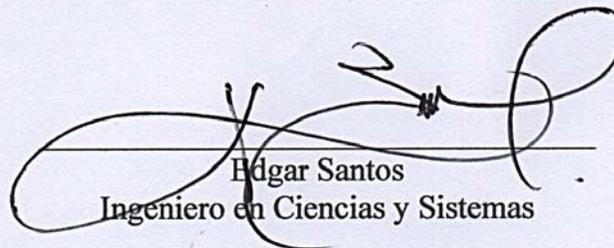
Ingeniero
Carlos Azurdia
Revisor de Trabajo de Graduación
Escuela de Ciencias y Sistemas
Facultad de Ingeniería

Respetable Ingeniero Azurdia

Por este medio hago de su conocimiento que he revisado el trabajo de graduación de los estudiantes **Angel Estuardo Cajas Ochoa**, identificado con el número de carné **201020757**, titulado: **“DESARROLLO DE UNA APLICACIÓN PARA TELÉFONOS MÓVILES INTELIGENTES ENFOCADA A LA MEJORA DE LA SALUD DE LOS USUARIOS_ HEALTHYFOOD. IMPLEMENTANDO EL MODELO DE ARQUITECTURA 4+1 VISTAS”**, y a mi criterio el mismo cumple con los objetivos propuestos para su elaboración de acuerdo al protocolo presentado.

Sin otro particular, me suscribo de usted,

Atentamente,



Edgar Santos
Ingeniero en Ciencias y Sistemas

Edgar Santos
INGENIERO EN CIENCIAS Y SISTEMAS
Colegiado 5266



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 27 de Mayo de 2015

Ingeniero
Marlon Antonio Pérez Türk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **ÁNGEL ESTUARDO CAJAS OCHOA** con carné **201020757**, titulado: **"DESARROLLO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES INTELIGENTES ENFOCADA A LA MEJORA DE LA SALUD DE LOS USUARIOS: HEALTHYFOOD. IMPLEMENTANDO EL MODELO DE ARQUITECTURA DE 4+1 VISTAS"**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados

y Revisión de Trabajos de Graduación



UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN
CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“DESARROLLO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES INTELIGENTES ENFOCADA A LA MEJORA DE LA SALUD DE LOS USUARIOS: HEALTHYFOOD IMPLEMENTANDO EL MODELO DE ARQUITECTURA DE 4+1 VISTAS”**, realizado por el estudiante ANGEL ESTUARDO CAJAS OCHOA, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

Ing. Marlon Antonio Pérez Türk

Director

Escuela de Ingeniería en Ciencias y Sistemas

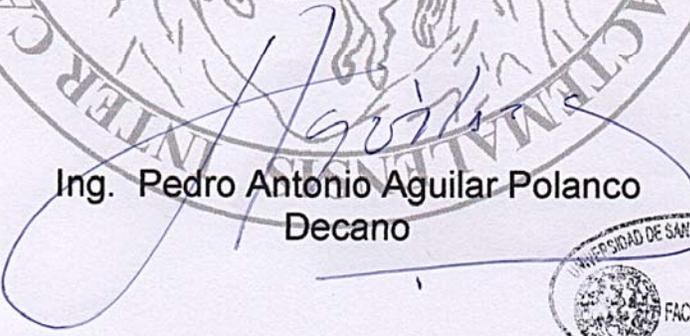


Guatemala, 13 de mayo de 2016



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **DESARROLLO DE UNA APLICACIÓN PARA DISPOSITIVOS MÓVILES INTELIGENTES ENFOCADA A LA MEJORA DE LA SALUD DE LOS USUARIOS: HEALTHYFOOD IMPLEMENTANDO EL MODELO DE ARQUITECTURA DE 4+1 VISTAS**, presentado por el estudiante universitario: **Angel Estuardo Cajas Ochoa**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.


Ing. Pedro Antonio Aguilar Polanco
Decano



Guatemala, mayo de 2016

ACTO QUE DEDICO A:

- Dios** Por acompañarme en todo momento de mi vida.
- Mis padres** Por el apoyo incondicional, amor e innumerables oportunidades que me han dado durante toda mi vida.
- Mis hermanos** Alejandro y Mildred Cajas, por ser personas con las que he compartido toda mi vida y con las que siempre puedo contar sin importar cuál sea la situación.
- Mis amigos** Por la amistad y apoyo que me han brindado.

AGRADECIMIENTOS A:

**Universidad de San
Carlos de Guatemala**

Por ser la institución educadora que me ha proporcionado la oportunidad de mejorar mis conocimientos y habilidades en mi formación como profesional.

Mi compañero

Luis Fernando Morales Mejicanos, por la ayuda y el apoyo proporcionado en la elaboración de este trabajo de graduación.

**Mis amigos y
compañeros de la
Facultad de Ingeniería**

Por todas las experiencias compartidas durante el transcurso de la carrera y por los momentos en los que me han apoyado para mejorar y seguir adelante.

**Mis asesores de
trabajo de graduación**

Edgar Santos y Eddy Méndez, por habernos ayudado a sacar adelante este proyecto a través de su orientación.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	VII
LISTA DE SÍMBOLOS	XI
GLOSARIO	XIII
RESUMEN.....	XXIII
OBJETIVOS.....	XXV
INTRODUCCIÓN	XXVII
1. TECNOLOGÍA MÓVIL.....	1
1.1. Impacto de la tecnología móvil en Guatemala.....	2
1.2. Impacto de la tecnología móvil a nivel mundial	3
1.3. Plataformas móviles inteligentes	3
1.3.1. Hardware relacionado a dispositivos móviles	4
1.3.1.1. Teléfonos inteligentes (<i>smartphones</i>)....	4
1.3.1.2. Tabletas.....	4
1.3.2. Sistemas operativos	5
1.3.2.1. Android	6
1.3.2.2. iOS.....	6
1.3.2.3. Windows Phone.....	7
1.3.2.4. BlackBerry	7
2. IDENTIFICACIÓN DEL PROBLEMA.....	9
2.1. Análisis e interpretación de los datos de la encuesta	9
2.2. Metodología de investigación a utilizar	24
2.2.1. Technology Acceptance Model (TAM).....	25

2.2.1.1.	Utilidad percibida (<i>perceived usefulness</i>).....	26
2.2.1.2.	Facilidad de uso percibida (<i>perceived ease-of-use</i>).....	26
2.2.1.3.	Actitud ante el uso (<i>attitude toward using</i>).....	26
2.2.1.4.	Intención de conducta ante el uso (<i>behavioral intention to use</i>)	27
2.3.	<i>Benchmarking</i> de la aplicación.....	27
2.3.1.	MyFitnessPal.....	28
2.3.1.1.	Funciones principales.....	28
2.3.1.2.	Ventajas	29
2.3.1.3.	Desventajas.....	29
2.3.2.	FatSecret.....	29
2.3.2.1.	Funciones principales.....	30
2.3.2.2.	Ventajas	30
2.3.2.3.	Desventajas.....	31
2.3.3.	Wii Fit	31
2.3.3.1.	Funciones principales.....	31
2.3.3.2.	Ventajas	33
2.3.3.3.	Desventajas.....	33
2.4.	Solución tecnológica	34
2.4.1.	Cálculo del índice de masa corporal	35
2.4.2.	Ecuación de Harris-Benedict (ingesta diaria recomendada)	36
3.	ANÁLISIS Y DISEÑO DE LA APLICACIÓN.....	39
3.1.	Lenguaje Unificado de Modelado (UML)	40
3.1.1.	Diagramas UML	41

	3.1.1.1.	Diagramas estructurales.....	41		
	3.1.1.2.	Diagramas de comportamiento.....	41		
3.2.		Modelo de 4+1 vistas.....	41		
	3.2.1.	Vista de casos de uso.....	43		
		3.2.1.1.	Diagrama de casos de uso extendido del sistema.....	44	
		3.2.1.2.	Definición de casos de uso	45	
	3.2.2.	Vista lógica	56		
		3.2.2.1.	Diagramas de clases de la aplicación..	57	
		3.2.2.2.	Diagrama de paquetes de la aplicación.....	77	
	3.2.3.	Vista de procesos	78		
		3.2.3.1.	Diagramas de secuencias	79	
	3.2.4.	Vista de despliegue	83		
	3.2.5.	Vista de implementación.....	84		
		3.2.5.1.	Diagrama de componentes.....	84	
		3.2.5.2.	Diagrama de despliegue.....	86	
3.3.		Modelo relacional de base de datos	86		
4.		ASPECTOS TÉCNICOS DE LA APLICACIÓN	89		
	4.1.	Servicios de datos	89		
		4.1.1.	SQLite.....	89	
		4.1.2.	MySQL.....	90	
	4.2.	Servicios RESTful.....	91		
		4.2.1.	Aplicación de REST en Web Services.....	92	
	4.3.	Plataforma Android.....	92		
		4.3.1.	Arquitectura del sistema operativo.....	93	
			4.3.1.1.	Núcleo Linux.....	94
			4.3.1.2.	Bibliotecas	95

4.3.1.3.	Runtime de Android.....	95
4.3.1.4.	Marco de trabajo de aplicaciones.....	96
4.3.1.5.	Aplicaciones	96
4.3.2.	Android APIs	97
4.3.3.	Android SDK.....	97
4.4.	Entorno de desarrollo Eclipse	98
4.4.1.	Android Developer Tools (ADT)	99
4.5.	Entorno de desarrollo Android Studio.....	100
4.6.	Sistema de versiones GIT	101
4.7.	Repositorio centralizado en el sitio GitHub.....	102
4.8.	Bibliotecas de terceros utilizadas	103
4.8.1.	MPAndroidChart.....	104
4.8.1.1.	Gráficos utilizados	104
4.8.1.1.1.	Gráfico de línea.....	104
4.8.1.1.2.	Gráfico de pie.....	105
4.8.2.	YouTube Android Player API.....	106
4.8.3.	Facebook SDK	107
4.9.	Requisitos de uso de la aplicación	108
5.	APLICACIÓN DESARROLLADA	109
5.1.	Utilización de la aplicación	109
5.1.1.	Registro de usuarios	109
5.1.2.	Autenticación de usuarios	110
5.1.3.	Pantalla principal	113
5.1.3.1.	Medición nueva	115
5.1.3.2.	Progreso.....	116
5.1.3.3.	Medidor de calorías.....	119
5.1.3.4.	Ver mis recetas	120
5.1.3.5.	Catálogo de ingredientes.....	120

5.1.4.	Menú lateral	122
5.1.5.	Recetas	123
5.1.5.1.	Publicación de recetas.....	124
5.1.5.2.	Buscar recetas.....	127
5.1.5.3.	Mis recetas	128
5.1.5.4.	Recetas favoritas	129
5.1.5.5.	Visualización de recetas	130
5.1.6.	Ejercicios	132
5.2.	Archivo ejecutable para la instalación de la aplicación	134
5.3.	Licenciamiento de la aplicación	134
CONCLUSIONES		135
RECOMENDACIONES.....		137
BIBLIOGRAFÍA.....		139
APÉNDICES		141

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Gráfico de encuestados por carrera	10
2.	Gráfico de edades de los encuestados	11
3.	Gráfico del sexo de los encuestados.....	12
4.	Pertenencia de dispositivos móviles.....	13
5.	Sistemas operativos por persona	14
6.	Versión de Android.....	15
7.	Gráfico de utilización de plan de datos.....	16
8.	Gráfico de preferencia de aplicaciones	17
9.	Gráfico de preferencias sobre aspectos importantes en una aplicación	18
10.	Gráfico sobre calorías	19
11.	Gráfico sobre conocimiento de calorías en alimentos	20
12.	Gráfico sobre conocimiento del peso actual.....	21
13.	Gráfico sobre conocimiento de la ingesta diaria de calorías	22
14.	Gráfica sobre la cantidad de calorías consumidas por tiempo de comida.....	23
15.	Gráfico sobre actividad física	24
16.	Modelo de aceptación de la tecnología	25
17.	Clasificación de la OMS del estado nutricional de acuerdo al IMC	36
18.	Modelo de arquitectura de 4+1 vistas	43
19.	Diagrama de casos de uso extendido	45
20.	Diagrama de caso de uso de autenticación	46
21.	Diagrama de caso de uso de publicación de recetas	48

22.	Diagrama de caso de uso de búsqueda de recetas.....	50
23.	Diagrama de caso de uso de búsqueda de ejercicios.....	52
24.	Caso de uso de creación de Perfil	53
25.	Diagrama de caso de uso de ver Perfil	55
26.	Diagrama de clases del paquete “healthyfood.sqlite”	57
27.	Diagrama de clases del paquete “healthyfood.sobj”	62
28.	Diagrama de clases del paquete “healthyfood”	65
29.	Diagrama de paquetes de la aplicación	78
30.	Diagrama de secuencia de autenticación	80
31.	Diagrama de secuencia de publicación de recetas.....	81
32.	Diagrama de secuencia de medición	82
33.	Diagrama de componentes de la aplicación	85
34.	Diagrama de despliegue para el sistema completo	86
35.	Modelo relacional de la base de datos remota	87
36.	Arquitectura del sistema operativo Android	94
37.	Entorno de desarrollo Eclipse	99
38.	Editor de interfaz gráfica del ADT	100
39.	Entorno de desarrollo Android Studio	101
40.	Repositorio público en GitHub desde un navegador web	103
41.	Gráfico de línea.....	105
42.	Gráfico de pie	106
43.	Reproductor embebido en una aplicación de Android	107
44.	Registro de usuarios	110
45.	Autenticación	111
46.	Inicio de sesión por redes sociales	112
47.	Pantalla de identificación de aplicación	113
48.	Pantalla principal.....	114
49.	Medición nueva.....	115
50.	Gráfico del progreso del usuario	116

51.	Transición entre pantallas de progreso	117
52.	Matriz del historial de mediciones del usuario	118
53.	Medidor de calorías.....	119
54.	Catálogo de ingredientes	121
55.	Menú lateral	122
56.	Pantalla de recetas.....	123
57.	Pantalla de publicación de recetas.....	125
58.	Pantalla para agregar ingredientes	126
59.	Pantalla de búsqueda de recetas	127
60.	Pantalla de recetas propias.....	128
61.	Listado de recetas favoritas	129
62.	Visualización de una receta	131
63.	Listado de ejercicios recomendados	132
64.	Visualización de ejercicios	133

TABLAS

I.	Características deseadas en la solución tecnológica	34
II.	Ecuaciones para el cálculo de la tasa metabólica basal	37
III.	Definición de caso de uso de autenticación	46
IV.	Definición de caso de uso de publicación de recetas.....	48
V.	Definición de caso de uso de búsqueda de recetas	50
VI.	Definición de caso de uso de búsqueda de ejercicios.....	52
VII.	Definición de caso de uso de creación de perfil	54
VIII.	Definición de caso de uso de ver perfil.....	55
IX.	Clase Medición.....	58
X.	Clase Sesión	59

XI.	Clase SQLiteHelper	60
XII.	Clase SQLite.....	61
XIII.	Clase Ingrediente_Receta	63
XIV.	Clase ContenedorIngredientes	64
XV.	Clase MainActivity.....	66
XVI.	Clase Login.....	66
XVII.	Clase Registro	68
XVIII.	Clase DrawerMenuActivity	69
XIX.	Clase Medidor.....	70
XX.	Clase Perfil	71
XXI.	Clase Recetas.....	72
XXII.	Clase Ejercicios	73
XXIII.	Clase PublicaReceta.....	74
XXIV.	Clase BuscaReceta	76
XXV.	Clase VisualizaReceta	77

LISTA DE SÍMBOLOS

Símbolo	Significado
cm	Centímetro
kg	Kilogramo

GLOSARIO

Android	Sistema operativo móvil desarrollado por Google. Basa su interfaz de interacción con el usuario en actividades, utilizando, como núcleo, el kernel Linux
AndroidStudio	Entorno integrado de desarrollo basado en IntelliJ de JetBrains. Diseñado específicamente para desarrollar aplicaciones para el sistema operativo móvil Android.
API	<i>Application Programming Interface</i> . Conjunto de subrutinas, funciones y procedimientos ofrecidos por una biblioteca de software para ser utilizado por otro programa.
Aplicación	En informática es una solución de software o programa informático diseñado como herramienta con un propósito específico para el usuario final.
Apple	Empresa multinacional estadounidense que diseña y produce equipos electrónicos, con sedes en California y Dublín. Entre sus productos más vendidos se encuentran los iPhone, iPads y las computadoras Macintosh.

Archivo Binario	Archivo que contiene información de cualquier tipo, la cual se encuentra codificada en binario para su almacenamiento y procesamiento en computadores.
Base de datos	En informática es un conjunto de datos organizados con el objetivo de obtener, con rapidez, la información almacenada.
<i>Benchmarking</i>	Conjunto de procedimientos utilizados para la evaluación y funcionamiento de sistemas, comparándolos con otros índices conocidos.
Biblioteca	Conjunto de implementaciones funcionales, codificadas en cualquier lenguaje de programación, las cuales ofrecen una interfaz definida para su utilización e invocación. No se espera que se utilicen como un programa, el cual se ejecuta de forma autónoma.
Caloría	Unidad de energía del Sistema Técnico de Unidades la cual se basa en el calor específico del agua.
Clase	En programación orientada a objetos, es un tipo de dato definido por el usuario que especifica un conjunto de objetos con las mismas características.

Código	Se refiere a las instrucciones contenidas en un programa, y entendibles por la computadora.
Commit	Operación de confirmación de cambios realizados en un repositorio almacenado en un sistema de control de versiones.
Computadora	También denominada ordenador o computador. Es una máquina procesadora de datos cuyo fin es convertirlos en información conveniente y útil.
Eclipse	Entorno de desarrollo integrado compuesto por múltiples herramientas de código abierto, desarrollado por IBM y mantenido por Eclipse Foundation en la actualidad.
Facebook	Sitio web de redes sociales creada por Mark Zuckerberg, inicialmente dirigida para los estudiantes de Harvard. Está expandida para todo aquel con una cuenta de correo electrónico. Cuenta con más de 1 350 millones de usuarios.
Fork	Traducida literalmente del inglés como tenedor. Dentro de los términos utilizados en la gestión del cambio es la creación de un proyecto nuevo derivado del previo desarrollo de otro a partir de una versión específica.

Google	Empresa multinacional estadounidense especializada en productos informáticos y servicios relacionados a internet. La empresa posee los derechos de propiedad del sistema operativo móvil Android. Es conocida por su buscador en internet.
Google Forms	Aplicación en línea de Google cuya finalidad reside en la creación de formularios en línea.
Hardware	Conjunto de componentes físicos o materiales de un sistema tecnológico o informático.
Herencia	En programación orientada a objetos, la herencia implica la creación de clases e interfaces a partir de otras, adhiriendo las características y comportamiento de quien hereda.
HTTP	<i>Hypertext Transfer Protocol</i> es un protocolo altamente utilizado en redes, utilizado en cada transacción dentro de internet.
IDR	Ingesta diaria recomendada. En nutrición es la dosis mínima a consumir de un nutriente, o conjunto de nutrientes, para mantener una salud alimenticia óptima.
IMC	Índice de Masa Corporal, también conocido como índice de Quetelet. Es una medida de asociación entre la masa y la talla de un individuo.

Internet	Red de computadoras de área global, creada a partir de redes de menor tamaño, cuyo origen reside en la cooperación de dos universidades estadounidenses. Actualmente es la red más grande del mundo.
JSON	<i>JavaScript Object Notation</i> . Es un formato ligero para el intercambio de datos, su notación está basada en las colecciones de objetos de JavaScript.
Microsoft	Compañía de software, la cual es considerada la más grande del mundo. También se conoce por sus lenguajes de programación y aplicaciones para computadoras personales, así como por sus sistemas operativos.
Multimedia	Cualquier elemento o sistema de medios que presentan o combinan sonido, gráficos, animación y vídeo para presentar o comunicar información
Nintendo	Compañía japonesa fabricante de videojuegos y videoconsolas.
Nokia	Empresa finlandesa orientada a las telecomunicaciones y a la tecnología. Es conocida principalmente por la fabricación de teléfonos móviles.

Objeto	Es una ejemplificación general de datos de una clase, que posee características propias y métodos para manipular su comportamiento.
Ofimática	Concepto aplicable a todas aquellas técnicas, procedimientos y servicios soportados por tecnologías de la información y cuya implementación se realiza en el ámbito del trabajo de oficina y entornos similares.
OMS	Organización Mundial de la Salud. Se especializa en gestionar políticas de prevención, promoción e intervención en salud a nivel mundial.
<i>Open Source</i>	Software desarrollado bajo la línea del código abierto, casi siempre distribuido de manera gratuita y permitiendo su modificación.
Paquete	Es un conjunto de clases e interfaces relacionadas entre sí, se encuentran organizadas e identificadas por medio de un nombre.
Paradigma	Técnica, modelo o conjunto de herramientas para representar la solución de problemas específicos.
Plataforma	Término usado normalmente para referirse a una arquitectura de hardware específica. El término también es utilizado para sistemas operativos o para el conjunto de ambos.

<i>Plugin</i>	Es una aplicación informática que interactúa con otra aplicación para aportarle una función o utilidad específica.
Programa	Conjunto de instrucciones ordenadas lógicamente que permiten realizar una tarea o trabajo específico.
<i>Release</i>	Es la distribución de la versión final en una etapa del ciclo de vida de un software.
Repositorio	Sitio centralizado donde se mantiene almacenada, de forma ordenada, información digital, tales como bases de datos o archivos.
REST	<i>Representational State Transfer</i> o Transferencia de estado representacional. Arquitectura de software basada en la transferencia y comunicación por medio de hypermedia.
Sitio Web	Hace referencia a una página o grupo de páginas web alojadas en un servidor web, las cuales están relacionadas entre sí en un mismo dominio de internet.
SDK	<i>Software Development Kit</i> o Kit de desarrollo de software. Provee herramientas para los programadores con el objetivo de facilitar el desarrollo de software para una plataforma

SOAP	<i>Simple Object Access Protocol</i> . Provee los mecanismos para la ejecución de llamadas a procedimientos remotos entre programas, de tal manera que se establezcan eficientemente las comunicaciones de tipo "programa a programa".
Software	Término genérico que designa al conjunto de programas de distinto tipo, sean sistemas operativos o aplicaciones diversas, que hacen posible operar un equipo de cómputo.
SQL	<i>Structured Query Language</i> . Es un lenguaje estándar para el acceso, manipulación y definición de bases de datos.
Stakeholder	Persona o grupo de personas que poseen un interés o se ven afectadas por la toma de decisiones dentro de una organización.
Symbian	Sistema operativo para dispositivos móviles, encontrado especialmente en teléfonos móviles Nokia.
Tasa metabólica basal	Cantidad de energía mínima que necesita un individuo para mantener sus constantes vitales en funcionamiento

Twitter	Red social cuyo principal objetivo es compartir mensajes no mayores a 140 caracteres entre usuarios.
Videoconsola	Dispositivo electrónico que transmite señales de video o imagen visual que permiten visualizar e interactuar con un videojuego.
Videojuego	Juego electrónico en el cual interactúan una o varias personas a través de una videoconsola, computadora o dispositivo móvil.
Web Service	Servicio Web en español. Es un sistema de software diseñado para soportar comunicación interoperable entre dispositivos a través de una red.
Wii	Videoconsola fabricada por Nintendo.
Wii Balance Board	Accesorio de la videoconsola Wii de Nintendo que calcula el peso y posición de una persona
World Wide Web	Es un sistema de distribución de hipertexto o hipermedios accesibles e interconectados a través de internet.
Xbox	Videoconsola fabricada por Microsoft.

XML

Extensible Markup Language. Es un metalenguaje de etiquetado, simple y estricto, adecuado a usos determinados. En la práctica corresponde a un estándar que permite a diferentes aplicaciones interactuar con facilidad a través de la red.

YouTube

Es un servicio para compartir videos que permite a un usuario ver videos publicados por otros usuarios y subir sus propios videos.

RESUMEN

La tecnología móvil inteligente ha cambiado la forma de ver las soluciones de software. Hoy en día es más común observar que las personas utilizan sus dispositivos móviles para realizar tareas, que estaban anteriormente destinadas a realizarse desde una terminal, computadora u ordenador personal. Este constante cambio ha permitido a los desarrolladores de software crear nuevas herramientas para la realización de tareas cotidianas, así como enfocarse en otros campos aún no explorados por completo, en cuanto a desarrollo de aplicaciones se refiere.

Uno de los campos que ha crecido es el campo de las aplicaciones enfocadas a la salud, tales como aplicaciones que registran actividad física, que miden el consumo de calorías, entre otras. De estas han surgido todo tipo de aplicaciones web y móviles, incluso videojuegos. Además, han conseguido que los usuarios tengan un interés mayor en el cuidado de su salud alimenticia.

La ingeniería de software es la ciencia del desarrollo de software, la cual fomenta buenas prácticas y patrones para el mismo, por lo que el desarrollo de aplicaciones móviles no queda exento de su implementación. Utilizando el concepto de ciclo de vida del desarrollo de software es posible organizar las diversas fases del mismo, permitiendo utilizar diversos modelos para la creación de las mismas. Al implementar el modelo de 4+1 vistas es posible documentar, de una manera gráfica y entendible, la arquitectura que compone una aplicación móvil, así como comprender la interacción que se espera tener con los usuarios finales.

OBJETIVOS

General

Desarrollar una aplicación útil y funcional para dispositivos móviles enfocada a la salud de los usuarios.

Específicos

1. Desarrollar una aplicación móvil enfocada en el apoyo a la salud de los usuarios a través de sugerencias para la mejora de sus hábitos alimenticios, por medio del cálculo de calorías consumidas, ingreso de medidas de masa corporal y sugerencia de ejercicios.
2. Disponer a la población de habla hispana en general, una aplicación enfocada a la salud con conectividad a redes sociales que permitirá la publicación de contenido referente a los hábitos alimenticios y calorías consumidas por los usuarios.
3. Implementar el Modelo de Aceptación de la Tecnología y el Modelo de Arquitectura de 4+1 vistas para definir el alcance de una aplicación móvil y el análisis, diseño y desarrollo de la misma, respectivamente.

INTRODUCCIÓN

Los dispositivos móviles inteligentes son un ejemplo de tecnologías adaptadas a nivel mundial que mayor éxito han tenido. Por consiguiente, existe una mayor cantidad de usuarios que adoptan esta tecnología día con día, permitiendo que simplifiquen tareas y actividades de la vida cotidiana. Estas son desde tareas relacionadas con ofimática, hasta planificaciones de eventos y manejo de agendas.

Las aplicaciones móviles se enfocan en distintos propósitos. Dentro de ellas han surgido algunas cuya temática está relacionada con la salud de los usuarios. Estas aplicaciones buscan mejorar la calidad de vida del usuario a través de asistencia con respecto a la actividad física y el plan alimenticio diario que les de una mejor calidad de vida.

Utilizando como caso de estudio la población estudiantil de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala se logra observar el interés en aplicaciones móviles enfocadas a la salud. Esto debido a que algunos de los miembros de dicha población presentan desconocimiento sobre la cantidad de calorías que debe consumir para mantenerse saludables. Por ello se ha determinado el desarrollo de una aplicación móvil con enfoque en la salud alimenticia y actividad física.

1. TECNOLOGÍA MÓVIL

La tecnología, como conjunto de conocimientos técnicos y científicos ordenados, permite el diseño, creación de bienes y servicios que facilitan la adaptación al medio ambiente y satisfacen, tanto deseos como necesidades de la humanidad.

Cada avance tecnológico le ha permitido al ser humano comprobar teorías descubiertas o propuestas con anterioridad, facilitando actividades de la vida cotidiana y permitiendo una reducción de esfuerzo para la realización de las mismas.

Existen diversos campos tecnológicos desde que el hombre tiene uso de razón. Dentro de ellos se encuentra el campo de la tecnología móvil, la cual pertenece, a su vez, al campo de las tecnologías de la información y comunicación.

Los elementos propios de la tecnología móvil son llamados dispositivos móviles, los cuales están altamente relacionados al concepto de computadora de bolsillo. En la actualidad, los dispositivos móviles permiten la realización y simplificación de diversas tareas en múltiples campos, como en la industria, educación, finanzas, ciencia, inteligencia de negocios, entretenimiento, entre otros.

En ocasiones se minimiza el concepto de tecnología móvil únicamente a dispositivos de la telefonía móvil. Sin embargo, se le denomina dispositivo móvil a todo aquel artefacto de funcionamiento electrónico que cumpla con la

característica de la movilidad, facilitando su transporte y su uso durante el mismo. Además debe contar con capacidad de procesamiento propia y almacenamiento de datos.

Los dispositivos móviles más utilizados en la actualidad son aquellos que se encuentran dentro del campo de la tecnología móvil inteligente. Entre ellos, se pueden mencionar desde teléfonos y tabletas, hasta relojes, gafas, reproductores y lectores de libros.

1.1. Impacto de la tecnología móvil en Guatemala

La tecnología móvil en Guatemala se ha introducido de manera acelerada durante los últimos años. De acuerdo al informe del segundo semestre del 2013 sobre crecimiento de la telefonía móvil de la Superintendencia de Telecomunicaciones (SIT), en Guatemala se encuentran activos 21 716 357 teléfonos móviles, mostrando un crecimiento del 4,47 % de activaciones más con respecto al año anterior.

Los dispositivos móviles más utilizados en Guatemala son los teléfonos móviles o teléfonos celulares, cuya cantidad de activaciones supera incluso a la población guatemalteca. Sin embargo, debido a que el crecimiento en este mercado ha ido disminuyendo en los últimos años, los oferentes de servicios relacionados a esta tecnología han ideado nuevas formas para atraer clientes.

Gracias a ello se han incrementado las ventas de dispositivos móviles inteligentes (tales como *smartphones* y tabletas). Los distribuidores de servicios de telefonía han optado por la distribución de planes de datos para el acceso a internet desde diferentes dispositivos, disminuyendo la utilización de la red celular.

1.2. Impacto de la tecnología móvil a nivel mundial

El crecimiento en el mercado de dispositivos electrónicos ha sido significativo durante los últimos años. Para el 2014 se alcanzarán aproximadamente ventas por 750 000 miles de millones de dólares estadounidenses en concepto de soluciones tecnológicas. Implicando *smartphones*, tabletas, computadoras portátiles, videoconsolas, televisores, entre otros.

De acuerdo a un estudio realizado por la empresa Deloitte, en la actualidad se está dando una estratificación en la oferta de tabletas. Por ello, la distribución de la oferta y la demanda de dispositivos móviles inteligentes se diversifica. Sin embargo, el crecimiento de ventas se está estancando, esto comparando estudios de años anteriores al 2014.

1.3. Plataformas móviles inteligentes

Los dispositivos móviles, en la actualidad y en su mayoría, operan bajo una plataforma en común. La administración de la capacidad de procesamiento, de manejo de recursos de entrada y salida, y de almacenamiento de los dispositivos móviles está regida bajo una arquitectura de hardware y un sistema operativo.

Los fabricantes de dispositivos móviles inteligentes, a manera de estandarizar, utilizan elementos de hardware similares. Por ello, es posible para ellos utilizar un sistema operativo en común, en algunos casos.

1.3.1. Hardware relacionado a dispositivos móviles

Dentro de los dispositivos móviles se encuentran diferentes tipos de hardware, cuyo objetivo varía de acuerdo al propósito por el cual son fabricados. A continuación se describen algunos usos de dispositivos móviles.

1.3.1.1. Teléfonos inteligentes (*smartphones*)

Los teléfonos inteligentes son dispositivos pertenecientes a la tecnología de la telefonía móvil, construidos sobre una plataforma informática. Su característica más notable es la forma revolucionaria en la que, incluyendo las características de un teléfono celular, incrementa las capacidades de procesamiento y almacenamiento para dar paso a nuevas características que realizan tareas más complejas. Esto, desde conectividad a internet, geolocalización, hasta la ejecución de aplicaciones de ofimática, aplicaciones de control de ritmo cardíaco, entre otras.

1.3.1.2. Tablet

Son computadoras portátiles operables por medio de un panel de pantalla táctil. Anteriormente la mayoría de estos dispositivos requerían exclusivamente de un lápiz electrónico para su uso, pero en la actualidad son operadas por medio de los dedos y el uso de los lápices electrónicos es opcional. Poseen solamente unos cuantos botones, para su escritura cuentan con un teclado virtual y en algunos casos es posible conectar un teclado físico. Se diferencian de los teléfonos inteligentes gracias a su tamaño, memoria, capacidad de almacenamiento y procesamiento, y a su finalidad.

A pesar que ya existían varios dispositivos con características de una tableta a finales del siglo XX, no fue sino hasta a finales de la década de los 2000 en que su popularidad incrementó con la llegada del iPad de Apple.

1.3.2. Sistemas operativos

Un sistema operativo es un software que administra todos los recursos de hardware y software de una computadora. Existen sistemas operativos con distinto enfoque, uno de ellos es el enfoque hacia dispositivos móviles, los cuales difieren de las computadoras en cuanto al hardware que poseen y a las tareas que deben realizar.

Inicialmente los sistemas operativos utilizados por teléfonos móviles eran sistemas embebidos, cuya tarea era manejar operaciones básicas requeridas por el dispositivo, tales como realizar llamadas, activar la radio, enviar mensajes, entre otros. Estos sistemas dependían directamente del hardware del dispositivo y de su fabricante.

Con la llegada de los teléfonos inteligentes, los sistemas operativos para dispositivos móviles se fortalecieron y estandarizaron el comportamiento de los dispositivos. Esto, hasta tomar una estructura similar a los sistemas tradicionales funcionales en ordenadores, adquiriendo características de administración de almacenamiento y memoria de gran tamaño, y de procesamiento para ejecutar aplicaciones más complejas, ya que serían necesarias para controlar todas las nuevas capacidades de hardware que estos ofrecen.

Durante el 2013 los sistemas operativos presentes en teléfonos móviles fueron Android, iOS, Windows Phone y Blackberry. Estando Android presente

en un 81 % de los teléfonos móviles vendidos durante ese año, seguido por iOS con un 12,9 %, Windows Phone con un 3,6 % y Blackberry con un 1,7 %, el 0,6 % restante corresponde a otros sistemas operativos.

1.3.2.1. Android

Es un sistema operativo móvil desarrollado por Google que utiliza el kernel Linux. Se considera un software gratuito y de código abierto (*open source*), política que permite a los desarrolladores la modificación y adecuación del sistema operativo a las necesidades del dispositivo en el que se implementará. Sin embargo, varios dispositivos incluyen software propietario en la compilación del sistema.

Está diseñado para su utilización en diversos dispositivos inteligentes, tales como teléfonos, tabletas, televisores, entre otros. Para los desarrolladores de aplicaciones para este sistema, la diversidad de dispositivos que utilizan Android, provee un gran mercado.

Según las ventas de 2013 se considera a Android como el sistema operativo móvil de mayor uso con un 81 % de presencia en los teléfonos inteligentes.

1.3.2.2. iOS

Es un sistema operativo móvil desarrollado por Apple y se encuentra exclusivamente en dispositivos fabricados por esta misma empresa. Fue desarrollado para uso en los iPhone y en las tabletas iPad. Es de código cerrado, por lo que no permite modificar el sistema operativo e inicialmente no soportaba aplicaciones de terceros. Es el segundo sistema operativo móvil de

mayor uso con un 12,9 % de presencia en teléfonos inteligentes de acuerdo a las ventas del 2013.

1.3.2.3. Windows Phone

Es un sistema operativo móvil desarrollado para teléfonos inteligentes por Microsoft de código cerrado y propietario. El sistema operativo posee primordialmente opciones de sincronización a varios de los servicios de Microsoft como lo son OneDrive, Outlook, servicios de Xbox, entre otros.

Luego de la adquisición de Nokia por parte de Microsoft, la compañía decidió que todos sus dispositivos tuviesen el sistema operativo Windows Phone en reemplazo de Symbian, por lo que generalmente se encuentra en dispositivos de esa marca. Durante el 2013 fue el tercer sistema operativo móvil de mayor uso con un 3 % de presencia en teléfonos inteligentes en las ventas de dicho año.

1.3.2.4. BlackBerry

Es el sistema operativo móvil de código cerrado y propietario. Fue desarrollado exclusivamente para dispositivos BlackBerry y se considera el cuarto sistema operativo móvil de mayor uso.

2. IDENTIFICACIÓN DEL PROBLEMA

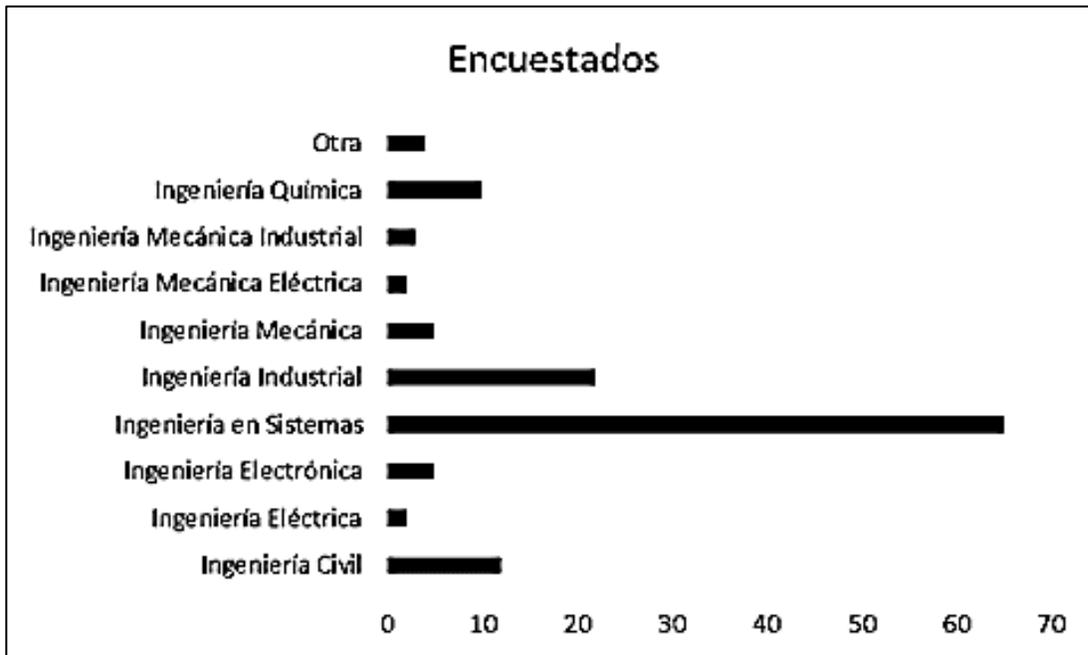
Se realizó una encuesta para identificar hábitos de alimentación y ejercicio que tienen los estudiantes de la Facultad de Ingeniería. Esta fue partiendo de la hipótesis que su alimentación no es la adecuada y cuál es la cantidad de ejercicio que realizan, para reconocer a qué aspectos se le debe dar un mayor enfoque durante el desarrollo de la aplicación.

2.1. Análisis e interpretación de los datos de la encuesta

La encuesta fue realizada por medio de Google Forms y contó con la participación de 130 personas. Se dividieron las preguntas de la encuesta basados en 3 diferentes tipos de preguntas: preferencia de aplicaciones móviles, conocimiento sobre calorías y deportes que practican.

El público objetivo de la encuesta se centraba en:

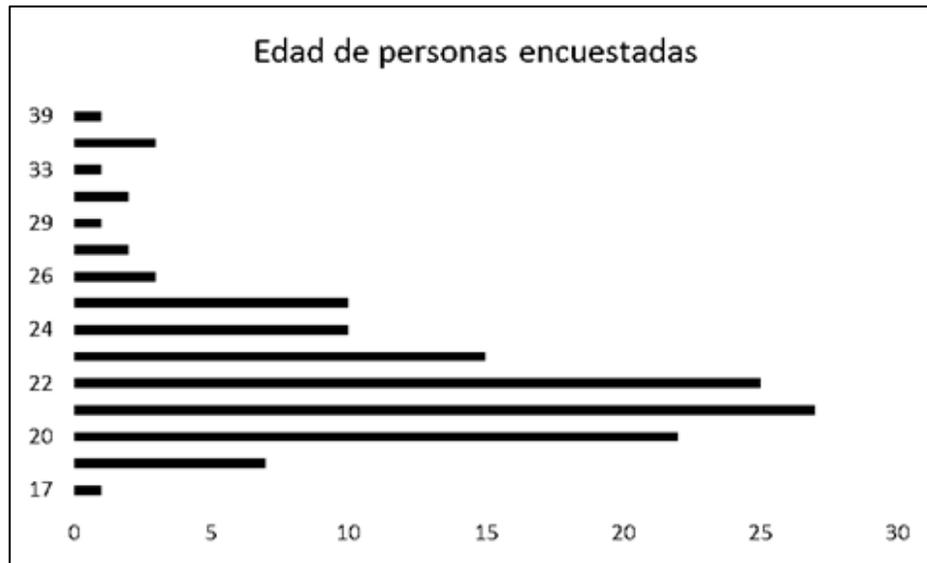
Figura 1. **Gráfico de encuestados por carrera**



Fuente: elaboración propia.

Del total de los usuarios encuestados, la mayoría se encuentra conformada por estudiantes de la carrera de Ingeniería en Sistemas con un total de 65 estudiantes. Seguidos por Ingeniería Industrial con un total de 22 estudiantes, y finalmente por estudiantes de Ingeniería Civil con 12.

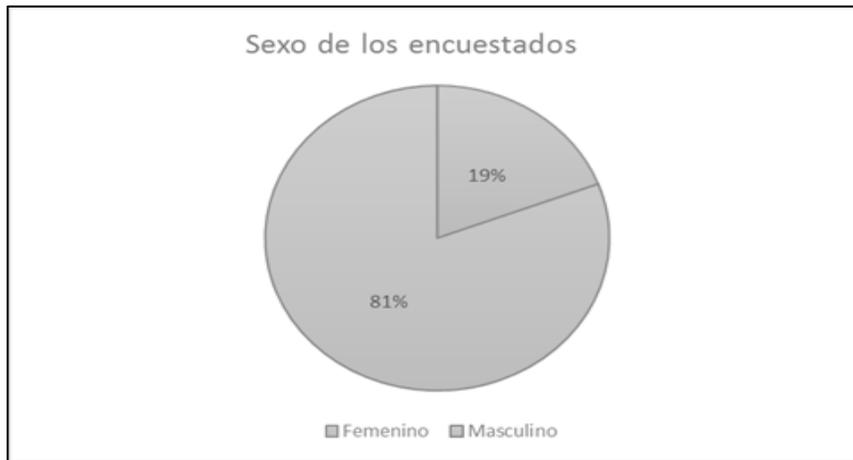
Figura 2. **Gráfico de edades de los encuestados**



Fuente: elaboración propia.

Entre los usuarios encuestados, la mayoría se encuentra en el rango de edad de 19 a 25 años.

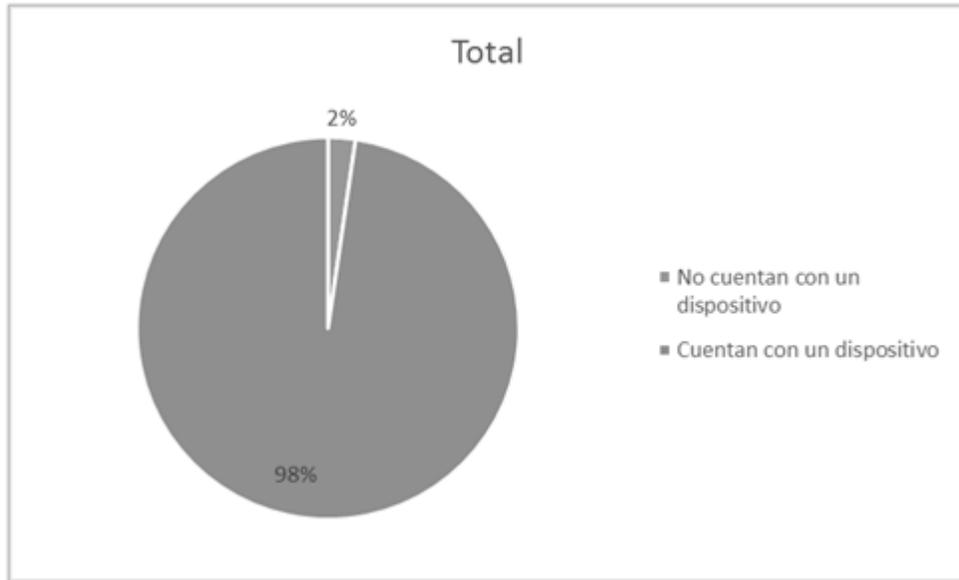
Figura 3. **Gráfico del sexo de los encuestados**



Fuente: elaboración propia.

El 81 % de los encuestados resultaron pertenecer al género masculino, 105 hombres participaron en la encuesta. Mientras que la cantidad de mujeres que participaron fue de 25.

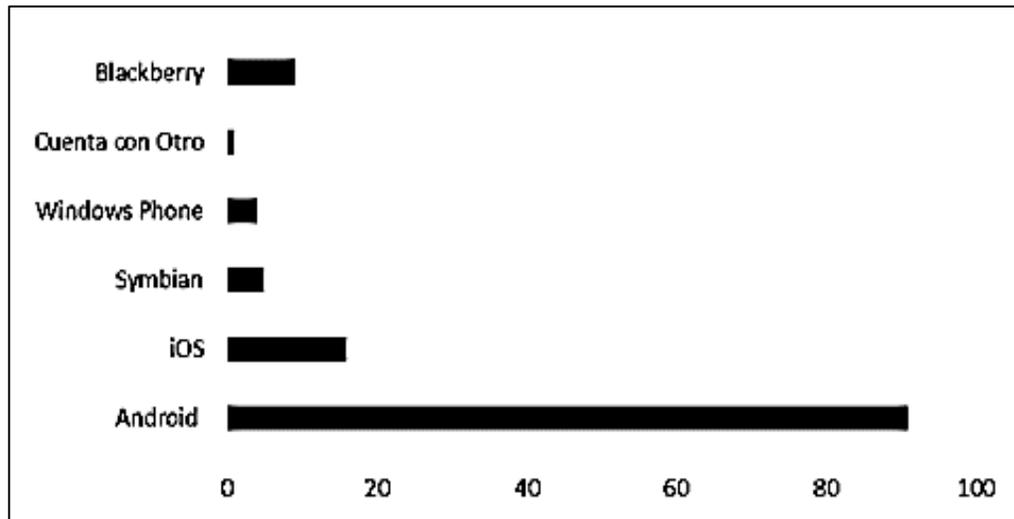
Figura 4. **Pertenencia de dispositivos móviles**



Fuente: elaboración propia.

Del total de encuestados se muestra que el 98 % tienen un dispositivo móvil que cuenta con la capacidad de poder instalar y usar aplicaciones. Esto indica que de llegar a publicar la aplicación que se desarrollará y podría ser accesible a la mayoría de los estudiantes.

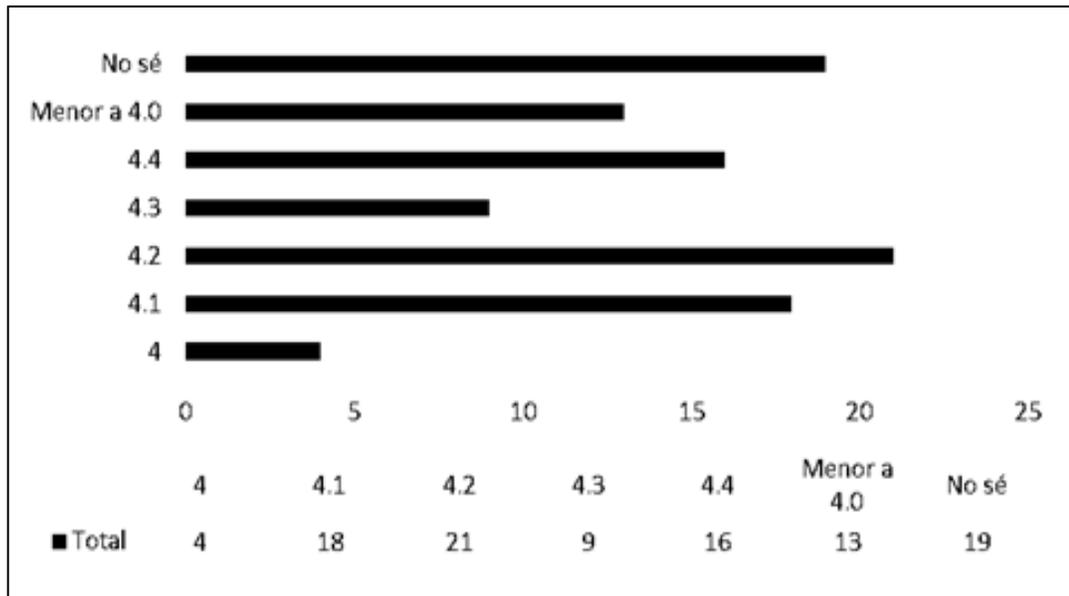
Figura 5. **Sistemas operativos por persona**



Fuente: elaboración propia.

Se observa que el sistema operativo de mayor uso dentro de estos dispositivos es el sistema operativo Android. Este dato es muy importante, ya que cada uno de los sistemas operativos llega a funcionar de manera distinta. Para desarrollar una aplicación es importante enfocarse a un sistema operativo en específico, así como la versión que se usa para evitar problemas de compatibilidad.

Figura 6. Versión de Android

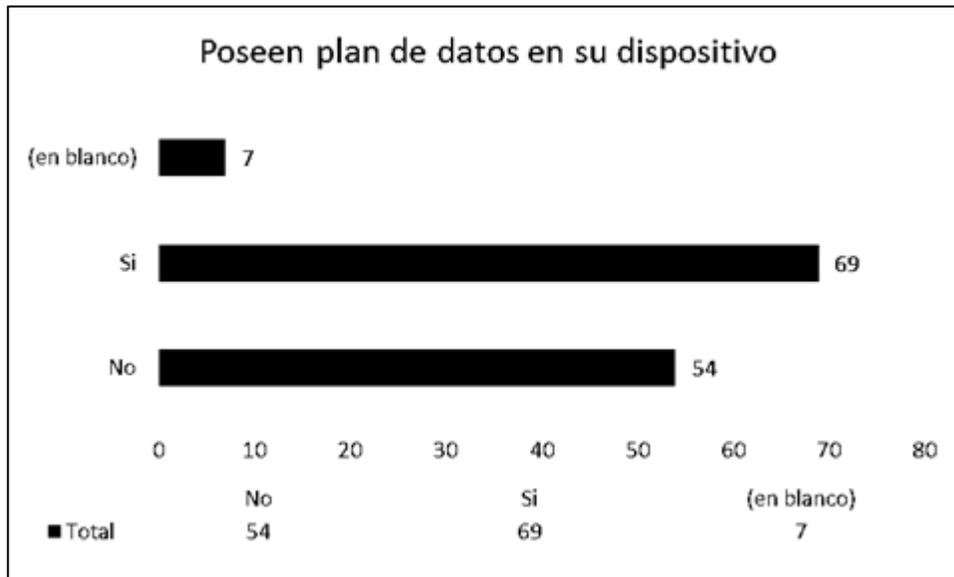


Fuente: elaboración propia.

Del total de las personas encuestadas que dijeron tenían un dispositivo con sistema operativo Android, 19 afirman no saber qué versión del sistema operativo tiene el dispositivo que usan. Mientras 13 indican poseer una versión menor a 4,0 y 68 utilizan la versión 4,0 o superior.

Debido a que un porcentaje mayor del 66 % posee la versión 4 o superior se puede inferir que se debe centrar el desarrollo de la aplicación a que funcione primordialmente en la versión 4,2, que es la que tiene mayor número de usuarios posee y se debe probar la compatibilidad con las demás.

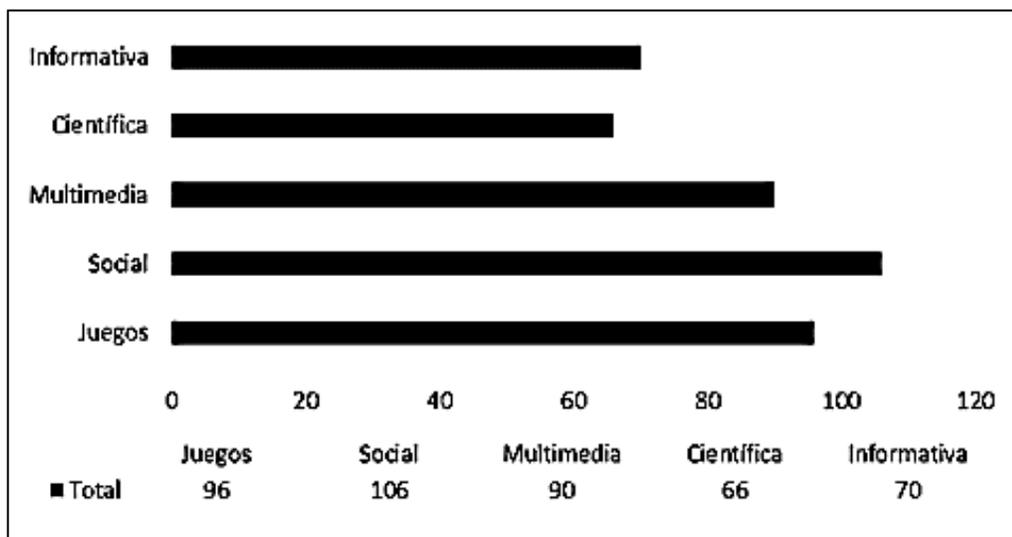
Figura 7. **Gráfico de utilización de plan de datos**



Fuente: elaboración propia.

Del total de encuestados 69 de ellos tienen acceso a internet por medio de un plan de datos. Mientras que 54 de ellos no poseen acceso a internet y 7 no dieron respuesta. Esto es un indicador para saber cuántas personas tendrían acceso a la aplicación inmediatamente, sin tener que necesitar de puntos de acceso inalámbricos.

Figura 8. **Gráfico de preferencia de aplicaciones**



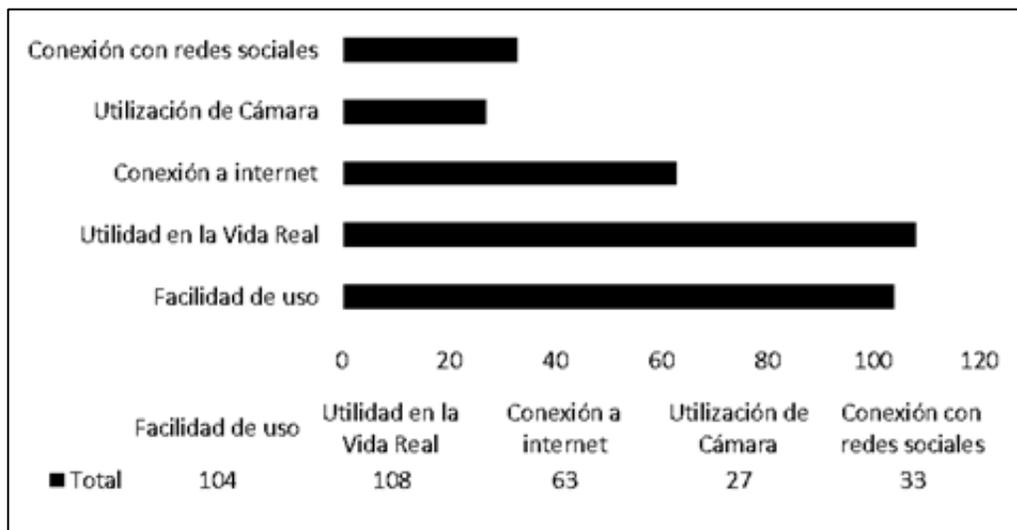
Fuente: elaboración propia.

Según los resultados de la encuesta, la mayoría de las personas prefiere el uso de aplicaciones del tipo social. Entre las cuales se incluyen las aplicaciones de las redes sociales y de llamadas y mensajería instantánea como lo son Twitter, Facebook y WhatsApp con 106 personas de las encuestadas usando este tipo de aplicaciones.

El segundo tipo de aplicaciones que tiene mayor uso es la de los juegos con 96 personas de las encuestadas afirmando que usan aplicaciones de juegos. El tercer tipo de aplicaciones de mayor uso son las aplicaciones de multimedia, que se refiere a los reproductores de música, de video, cámara y grabadoras, 90 personas afirman que hacen uso de este tipo de aplicaciones. Las menos populares son las de tipo científico e informativa con 66 y 70 personas respectivamente.

Con los datos anteriores se infieren los gustos de la gente al momento de usar una aplicación. Debido a que la aplicación a desarrollar combina la publicación de recetas con planes de ejercicio y control de calorías consumidas, esta entraría en los tipos informativa y científica que resultan al mismo tiempo ser las menos populares. Para aumentar su popularidad, entre los usuarios, se podría implementar una forma de compartir la información hacia las redes sociales o que en la misma aplicación haya interacción entre los usuarios.

Figura 9. **Gráfico de preferencias sobre aspectos importantes en una aplicación**



Fuente: elaboración propia.

Entre los aspectos más importantes que las personas buscan en una aplicación que ofrece algún tipo de servicio, las personas afirman que lo que más es que esta sea útil en la vida real. El segundo aspecto más importante es la facilidad de uso de la aplicación, por lo que el diseño no debe ser demasiado complicado para que los usuarios intuitivamente entiendan su funcionamiento.

Los siguientes aspectos son la conexión a internet y la conexión a redes sociales, por lo que se podría tomar en cuenta que las recetas o actividad física sean publicadas o vinculadas una cuenta de Facebook o Twitter. Esto para utilizar la aplicación. El último aspecto es el uso de la cámara del teléfono, pues 27 personas afirmaron es un aspecto importante.

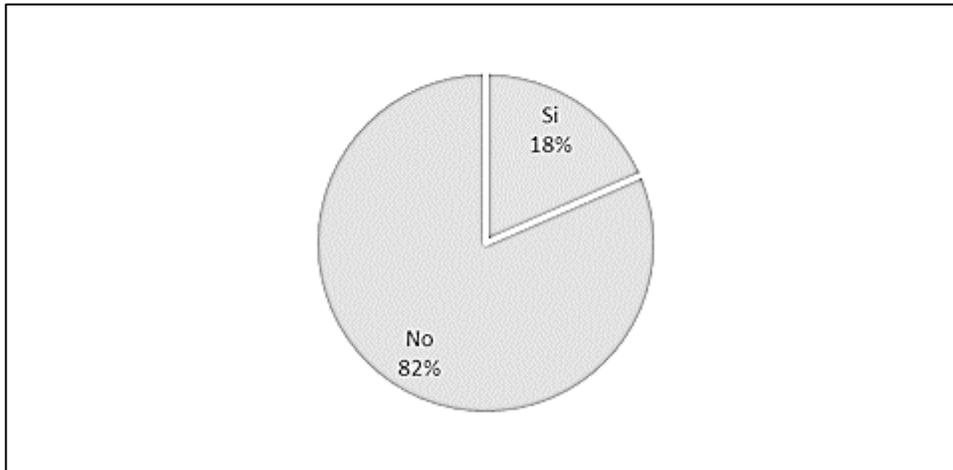
Figura 10. **Gráfico sobre calorías**



Fuente: elaboración propia.

De la cantidad total de personas encuestadas el 81 % afirma saber qué son las calorías. Es un aspecto importante a tomar en cuenta, ya que uno de los objetivos de la aplicación es mantener el control de la cantidad de calorías consumidas en el día y sugerir ejercicios basados en esa cantidad. Este dato es un buen indicio, ya que sabrán el valor que ofrece la aplicación sin necesitar de muchas explicaciones.

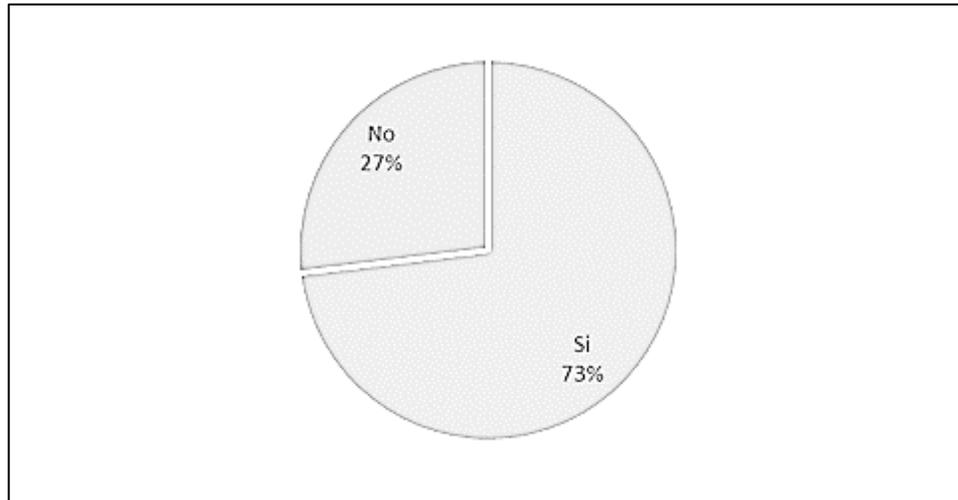
Figura 11. **Gráfico sobre conocimiento de calorías en alimentos**



Fuente: elaboración propia.

El 82 % de los encuestados afirma no saber la cantidad de grasa y calorías dentro de los alimentos consumidos. La aplicación deberá ofrecer esta información por lo que el hecho de que las personas no sepan dicho dato y quieran llegar a saberlo, podría llevarlos a usar la misma.

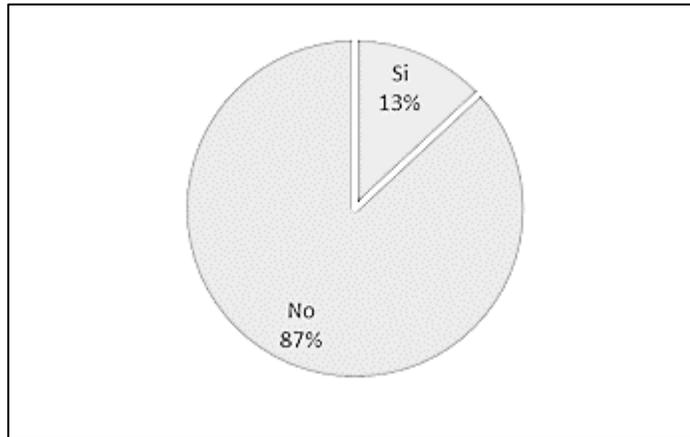
Figura 12. **Gráfico sobre conocimiento del peso actual**



Fuente: elaboración propia.

El peso es una de las variables que se toman en cuenta para calcular los indicadores dentro de la aplicación. El hecho que las personas sepan su peso actual indica que tienen un buen hábito. El 73 % de los encuestados dice que saben cuál es su peso actual.

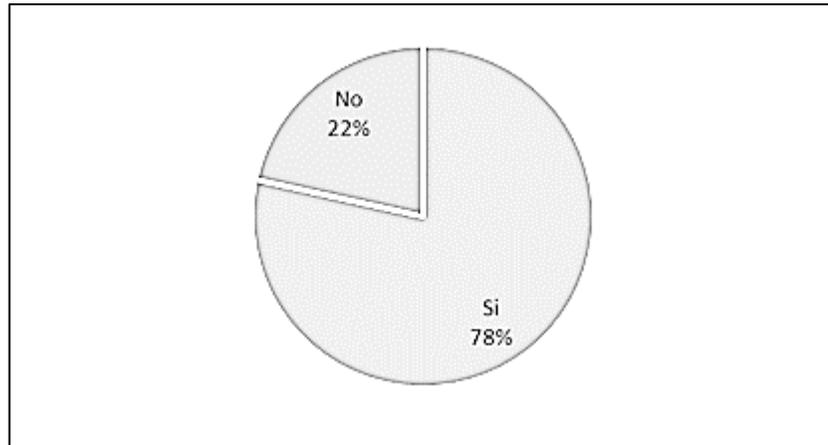
Figura 13. **Gráfico sobre conocimiento de la ingesta diaria de calorías**



Fuente: elaboración propia.

El 87 % de los encuestados no sabe la cantidad exacta de calorías que deben consumir. Dato que permite incluir el cálculo de calorías a través de la aplicación a desarrollar.

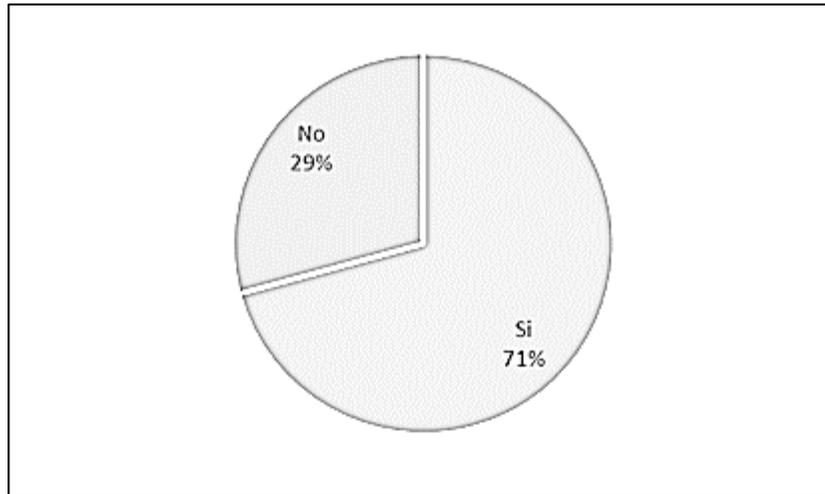
Figura 14. **Gráfica sobre la cantidad de calorías consumidas por tiempo de comida**



Fuente: elaboración propia.

Un dato interesante dentro de los resultados de la encuesta fue que el 78 % de los encuestados, sí conoce la cantidad de calorías que consume dentro de un tiempo de comida. Mientras que anteriormente afirmaron que no sabían la cantidad de calorías que deberían de consumir. Lo que indica que las personas saben aproximadamente cuantas calorías consumen, pero no saben si la cantidad consumida es la adecuada.

Figura 15. **Gráfico sobre actividad física**



Fuente: elaboración propia.

La mayoría de los encuestados dice practicar un deporte o actividad física. Un 71 % de los encuestados dice que realiza algún tipo de actividad física. Ya que la aplicación debería sugerir ejercicios que se deben realizar, las personas que lo hacen pueden llegar a encontrar útil esta función, para llevar el control del tiempo total de actividad física realizada.

2.2. Metodología de investigación a utilizar

Con base en los resultados tabulados de la encuesta mostrados con anterioridad, se presenta la siguiente metodología de investigación, la cual permitirá definir los parámetros necesarios para la aceptación de la aplicación a desarrollar entre los usuarios finales

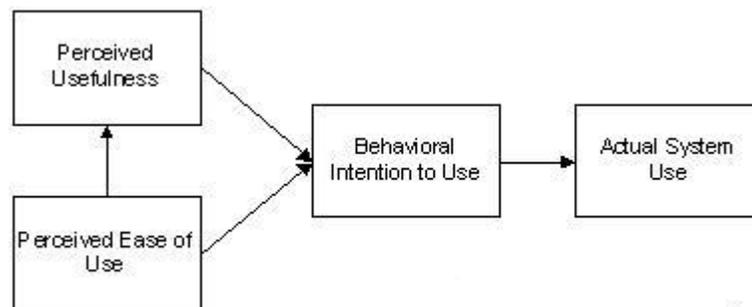
2.2.1. Technology Acceptance Model (TAM)

El modelo de aceptación de tecnología, por sus siglas en inglés, es una teoría sobre sistemas de la información que modela cómo los usuarios aceptan y utilizan un nuevo elemento tecnológico. Sugiere que cuando se presenta una nueva tecnología a los usuarios, una cierta cantidad de factores influyen la decisión de los mismos sobre cómo y cuándo utilizarán la misma.

El modelo de investigación a utilizar propone varios factores que se enfocan en el comportamiento al utilizar un elemento tecnológico. Entre estas variables o factores se encuentran:

- Utilidad percibida (*perceived usefulness*)
- Facilidad de uso percibida (*perceived ease-of-use*)
- Actitud ante el uso (*attitude toward using*)
- Intención de conducta ante el uso (*behavioral intention to use*)

Figura 16. **Modelo de aceptación de la tecnología**



Fuente: VENKATESH, V. *Theoretical IT Models*.
http://www.vvenkatesh.com/it/organizations/theoretical_models.asp.
Consulta: 15 de octubre de 2014.

2.2.1.1. Utilidad percibida (*perceived usefulness*)

Consiste en el grado en el que una persona cree, que utilizar un determinado sistema, puede mejorar el desempeño en su trabajo o en el alcance de un objetivo.

Las constantes mediciones que la aplicación realizará al usuario a través de su índice de masa corporal, le dará resultados que le permitan observar los cambios en su salud alimenticia y cómo mejora respecto al tiempo. En caso de empeorar, le alertará sobre ello.

2.2.1.2. Facilidad de uso percibida (*perceived ease-of-use*)

Consiste en el grado en el que las personas creen que la utilización de un determinado sistema estará libre de esfuerzos innecesarios y que tan fácil podrá utilizarse.

El enfoque de las aplicaciones móviles, en general, es facilitar la realización de tareas cotidianas a través de dispositivos móviles. La aplicación se ejecutará en dichos dispositivos, por lo que deberá contener fácil acceso a la mayoría de sus características, tales como las recetas y los ejercicios diarios. Además de ello, se pretende tomar en cuenta el uso de notificaciones que le permitan al usuario recordar su objetivo y su plan alimenticio.

2.2.1.3. Actitud ante el uso (*attitude toward using*)

Es el sentimiento positivo o negativo de los individuos ante el uso de un determinado sistema.

Se pretende realizar un sondeo, a partir de los datos recopilados, sobre el uso constante que le dan los usuarios a la aplicación. De esa forma, es posible determinar qué tan aceptada es dentro de la comunidad de usuarios.

2.2.1.4. Intención de conducta ante el uso (*behavioral intention to use*)

Consiste en el grado en el que una persona ha formulado, de manera consciente, planes para tener o no un comportamiento futuro específico.

Se desea que los consumidores de la aplicación la utilicen de manera constante. Además, los mismos basarán su plan alimenticio para mejorar su salud con las recomendaciones de la aplicación. También se espera que los usuarios compartan, al menos una receta, a la comunidad.

2.3. *Benchmarking* de la aplicación

En el mercado del desarrollo de soluciones tecnológicas se encuentra un sector especializado en brindar servicios referentes a la salud. Tomando en cuenta los problemas planteados anteriormente y los resultados obtenidos de la encuesta, se ha decidido participar en este sector de mercado, desarrollando una aplicación enfocada en la salud alimenticia de los usuarios.

Antes de comenzar el desarrollo de la aplicación, se estudiaron algunos sistemas de software y hardware que se encuentran ya consolidados dentro del mercado de soluciones tecnológicas para la salud. A continuación se describen algunos de estos sistemas.

2.3.1. MyFitnessPal

Es una aplicación para dispositivos móviles enfocada a la salud de los usuarios que, utilizando su contador de calorías, le indica al usuario la cantidad que debe consumir de forma diaria. Además, permite llevar un diario de consumo de calorías, dando la facilidad de observar el progreso a través del tiempo.

2.3.1.1. Funciones principales

- Registro de ingesta diaria: esta función permite a los usuarios registrar los alimentos que ha ingerido en los diferentes tiempos de comida durante el día. Muestra la cantidad de calorías consumidas en cada tiempo.
- Registro de ejercicio realizado: permite a los usuarios registrar, de forma diaria, la cantidad de actividad física llevada a cabo por los mismos. Aunado al registro de ingesta diaria, muestra la cantidad de calorías posiblemente quemadas.
- Visualización de recetas propias: permite obtener recetas planteadas por los usuarios, para una gestión más rápida de registro de alimentos.

- Recordatorio: mediante notificaciones, la aplicación le avisa al usuario que debe registrar lo ingerido o el ejercicio realizado.

2.3.1.2. Ventajas

- Permite a los usuarios conocer la cantidad de calorías consumidas por tiempo de comida al día.
- Muestra información verídica a los usuarios.
- Permite el ingreso de alimentos empacados por medio del código de barras.

2.3.1.3. Desventajas

- La aplicación posee demasiadas opciones, que no permiten su fácil utilización.
- No recomienda los ejercicios a realizar con base en el estado actual y al objetivo del usuario.
- No es compatible con versiones antiguas de Android, aún utilizadas.

2.3.2. FatSecret

Es una aplicación para dispositivos móviles encargada de llevar el control de las calorías consumidas por el usuario durante el día. Basa su control en un objetivo puesto por el mismo usuario, permitiéndole modificar su peso (sea

ganar o perder) dependiendo del caso. Lleva el control de un diario de comidas y un diario de ejercicios.

2.3.2.1. Funciones principales

- Registro de alimentos consumidos: permite al usuario indicar los alimentos que ha consumido durante el día, permitiendo realizar el cálculo de calorías provenientes de los mismos. Además de ello le indica la cantidad de calorías que necesita para cumplir el objetivo propuesto.
- Búsqueda de información sobre restaurantes: realiza una estimación acerca de los alimentos que proveen algunos restaurantes. Sin embargo, también incluye tiendas y cadenas de supermercado.
- Estimaciones gráficas de la ingesta diaria recomendada (IDR): muestra a los usuarios en una gráfica el porcentaje del IDR que han completado durante el día, basándose en los alimentos ingeridos.

2.3.2.2. Ventajas

- Posee un gran manejo de información sobre alimentos para el cálculo de calorías recomendadas.
- Presenta una interfaz gráfica sencilla de utilizar.
- Presenta abstracción de sus elementos para que el usuario no se confunda al utilizarla.

2.3.2.3. Desventajas

- No incita a los usuarios a alimentarse con comida casera.
- No sincroniza los datos del usuario con los servidores de la empresa que provee el servicio de forma automática.

2.3.3. Wii Fit

Es un videojuego desarrollado por Nintendo para la consola Wii. El juego consiste en varios ejercicios de aproximadamente 2 a 3 minutos de duración divididos en distintas categorías como lo serían ejercicios de Yoga, de equilibrio, aeróbicos, y otros. Utiliza principalmente una tabla de medición de peso llamada Wii Balance Board. El juego cuenta con tres versiones hasta la fecha Wii Fit, Wii Fit Plus y Wii Fit U.

2.3.3.1. Funciones principales

- Medición de peso, centro de gravedad e índice de masa corporal: el videojuego permite calcular el peso de una persona por medio de la Wii Balance Board y al mismo tiempo mostrar su centro de gravedad según la postura de la persona al hacer esta medición. Luego de esta medición calcula el índice de masa corporal, según estos resultados se le indica si el peso de la persona es el adecuado o si se encuentra con sobrepeso.
- El videojuego registra dentro del sistema estas medidas, permitiendo guardar estos datos por día y mostrando una gráfica de las mediciones para ver el avance de las personas. Wii Fit también permite fijarse un objetivo de peso con tiempo definido para alcanzarlos.

- Ejercicios: tiene una gran cantidad de ejercicios entre los cuales se encuentran ejercicios de yoga, aeróbicos y de equilibrio. Por medio de la Wii Balance Board el videojuego detecta si los ejercicios se están realizando de la manera correcta y les asigna una puntuación luego de haberlos realizado. Al finalizar un ejercicio indica aproximadamente las calorías quemadas y lleva la cuenta de cuántos minutos de ejercicio tiene en total.
- Ingreso de ejercicios externos y calorías: a partir de la versión Wii Fit Plus el videojuego permite ingresar alimentos consumidos diariamente para llevar el total de calorías. Facilita el ingreso y verificación otras actividades físicas como practicar un deporte y la duración que tuvo dicha actividad. Esto lo añade al diario de actividad para sumarlo a la cantidad de calorías quemadas. Se puede plantear un objetivo de consumo de calorías diario para que las personas no traten de sobrepasarlo.
- Rutinas y ejercicios recomendados: desde la versión Wii Fit Plus se crean rutinas de ejercicio. El usuario puede elegir los ejercicios que desee realizar y el orden en los que los realizará, sin necesidad de escogerlos uno por uno. También recomienda ejercicios según lo que se desea lograr, como lo serían dormir bien en la noche, mantenerse joven, para ayudar a la espalda, y otros.

- Fit meter: la versión Wii Fit U posee un podómetro que calcula la cantidad de pasos diarios. El aparato también incluye: un acelerómetro para medir la intensidad con la que se están realizando, identificando si el usuario está corriendo o caminando, y un barómetro para calcular la altitud de la persona.

2.3.3.2. Ventajas

- Permite realizar la medición del peso y centro de gravedad inmediatamente, sin necesidad de que el usuario lo ingrese.
- Ofrece ejercicios que pueden ser realizados para personas de cualquier edad.
- Es amigable y fácil de usar.
- Provee retroalimentación al usuario durante la realización de los ejercicios para evaluar su desempeño.

2.3.3.3. Desventajas

- Para consultar el avance de la persona y rutinas se debe usar la consola.
- No es portátil.
- Los ejercicios ofrecidos dentro del videojuego son cortos y mínimos por lo que no ofrecen una mejor solución a realizarlos en un gimnasio o hacer deporte.

- Se debe comprar una nueva versión para tener acceso a los nuevos ejercicios y actualizaciones del software.
- El medidor Fit Meter se limita solamente a llevar el registro de actividad de la persona. Aunque, este es portátil no se puede consultar la actividad diaria dentro del mismo.

2.4. Solución tecnológica

Debido a la interpretación de los resultados obtenidos de la encuesta y de lo observado en soluciones tecnológicas referentes a la salud alimenticia, se ha decidido desarrollar una aplicación que cuente con las siguientes características básicas:

Tabla I. **Características deseadas en la solución tecnológica**

Registro y autenticación de usuarios	Para utilizar la aplicación, los usuarios deberán registrarse en ella.
Recetas	Se incluirá la opción compartir recetas entre usuarios. Se podrán consultar las instrucciones para realizar la receta, ingredientes que contiene y la cantidad de calorías que contiene cada uno de ellos.
Objetivos	Se podrán plantear objetivos que el usuario desee para su peso y consumo de calorías, indicando a los usuarios cuál es el límite que desean consumir.

Continuación de la tabla I.

Ejercicios sugeridos	Según datos ingresados por el usuario, su progreso y objetivo, se le sugerirá que ejercicios son los que deben de realizar.
Registro de calorías consumidas	Se podrá registrar qué alimentos han sido ingeridos y la cantidad de calorías que han ganado al consumirlos.

Fuente: elaboración propia.

2.4.1. Cálculo del índice de masa corporal

La aplicación deberá llevar a cabo el cálculo del índice de masa corporal de los usuarios basándose en el peso y la altura de los mismos. A partir de dicho índice, la aplicación deberá indicarle al usuario en qué estado nutricional se encuentra, basándose en la siguiente tabla.

Figura 17. **Clasificación de la OMS del estado nutricional de acuerdo al IMC**

Classification	BMI (kg/m ²)	
	Principal cut-off points	Additional cut-off points
Underweight	<18.50	<18.50
Severe thinness	<16.00	<16.00
Moderate thinness	16.00 - 16.99	16.00 - 16.99
Mild thinness	17.00 - 18.49	17.00 - 18.49
Normal range	18.50 - 24.99	18.50 - 22.99
		23.00 - 24.99
Overweight	≥25.00	≥25.00
Pre-obese	25.00 - 29.99	25.00 - 27.49
		27.50 - 29.99
Obese	≥30.00	≥30.00
		30.00 - 32.49
Obese class I	30.00 - 34.99	32.50 - 34.99
		35.00 - 37.49
Obese class II	35.00 - 39.99	37.50 - 39.99
		≥40.00
Obese class III	≥40.00	≥40.00

Fuente: Organización Mundial de la Salud.

http://apps.who.int/bmi/index.jsp?introPage=intro_3.html. Consulta: 20 de octubre de 2014.

2.4.2. Ecuación de Harris-Benedict (ingesta diaria recomendada)

Para el cálculo de la ingesta diaria recomendada de calorías, la aplicación utilizará la ecuación de Harris-Benedict. Esta permite conocer la cantidad de calorías recomendadas a partir de una tasa metabólica basal, la cual es calculada a partir del peso, altura, edad de la persona y sexo.

Tabla II. **Ecuaciones para el cálculo de la tasa metabólica basal**

Hombres	$TMB = [(10 \times \text{masa en kilogramos}) + (6,25 \times \text{altura en centímetros}) - (5 \times \text{edad en años}) + 5]$
Mujeres	$TMB = [(10 \times \text{masa en kilogramos}) + (6,25 \times \text{altura en centímetros}) - (5 \times \text{edad en años}) - 161]$

Fuente: elaboración propia.

3. ANÁLISIS Y DISEÑO DE LA APLICACIÓN

Para el análisis y diseño de la aplicación se ha seleccionado un modelo de arquitectura de software. Este facilitará la comprensión de la estructura y comportamiento del sistema.

Un modelo de arquitectura de software consiste en una serie de vistas, las cuales permitirán al arquitecto de software organizar, de mejor manera, el diseño de la arquitectura de un producto de desarrollo de software. A partir de ello se podrá obtener, como una representación de alto nivel, el funcionamiento interno y las interacciones que tendrá el software construido. Se podría decir que un modelo de arquitectura de software es la organización de los planos para la construcción de un programa.

Además de contar con un modelo de arquitectura, los diseñadores y arquitectos de software deben establecer el tipo de planos que utilizarán para presentar la estructura de la arquitectura de software. Las diversas formas de representar cada vista de un modelo de arquitectura están regidas por estándares que determinan la forma en la que se presentarán los componentes de la estructura de un sistema.

Generalmente a estos se les conoce como lenguajes de modelado, dentro de los cuales cabe mencionar: *Unified Modeling Language* (UML), utilizado para modelar diversos componentes de arquitectura de software; *Business Process Model and Notation* (BPMN), cuya utilización se enfoca en modelar el manejo de procesos de negocio; entre otros.

Para la representación de alto nivel de la estructura de la aplicación *HealthyFood* se ha seleccionado el modelo de arquitectura de 4+1 vistas. Para la notación utilizada dentro del modelo de arquitectura se seleccionó UML. Utilizando como paradigma de programación, tanto para el diseño, como para la codificación de la aplicación, el paradigma orientado a objetos.

3.1. Lenguaje Unificado de Modelado (UML)

Propuesto en 1997 UML es el lenguaje de modelado de sistemas de software más conocido y utilizado actualmente. Es un lenguaje gráfico cuyo objetivo es visualizar, especificar, construir y documentar un sistema, estableciendo un estándar para describir el modelo del mismo. Así como también elementos conceptuales, como procesos de negocio, funciones del sistema, y aspectos específicos.

Generalmente es aplicado en el desarrollo de software para describir la arquitectura de los componentes de un sistema. Además que es utilizado para dar soporte a una metodología de desarrollo (como el Proceso Unificado Racional o RUP). Sin embargo, debido a su definición como lenguaje, no se debe confundir UML con una metodología por sí solo.

A partir de su creación, UML ha sufrido cambios a través del tiempo. Actualmente se encuentra en la versión 2,0 que contiene 13 tipos de diagramas que dan soporte a modelos como el Modelo Dirigido por la Arquitectura (MDA) y el Modelo Dirigido por el Desarrollo.

3.1.1. Diagramas UML

UML está comprendido por 13 diagramas, los cuales están divididos en dos categorías principales.

3.1.1.1. Diagramas estructurales

Dentro de esta categoría se encuentran los diagramas de UML que describen y definen una arquitectura estática. Son utilizados, principalmente, para definir construcciones estáticas como clases, objetos y componentes, además de representar las diversas relaciones entre ellos. Dentro de los diagramas estructurales se encuentran 6 tipos de diagramas: paquetes, clases, objetos, estructura compuesta, componentes y despliegue.

3.1.1.2. Diagramas de comportamiento

Son diagramas cuyo objetivo está encaminado a representar arquitecturas dinámicas. Están comprendidos por estructuras dinámicas como actividades, estados, líneas de tiempo y mensajes entre objetos. UML contiene 7 diagramas de comportamiento, los cuales son: casos de uso, actividad, estados, secuencia y el global de interacciones.

3.2. Modelo de 4+1 vistas

La organización fundamental de un sistema de software es representada de la siguiente manera:

- Elementos estructurales e interfaces que componen el sistema.

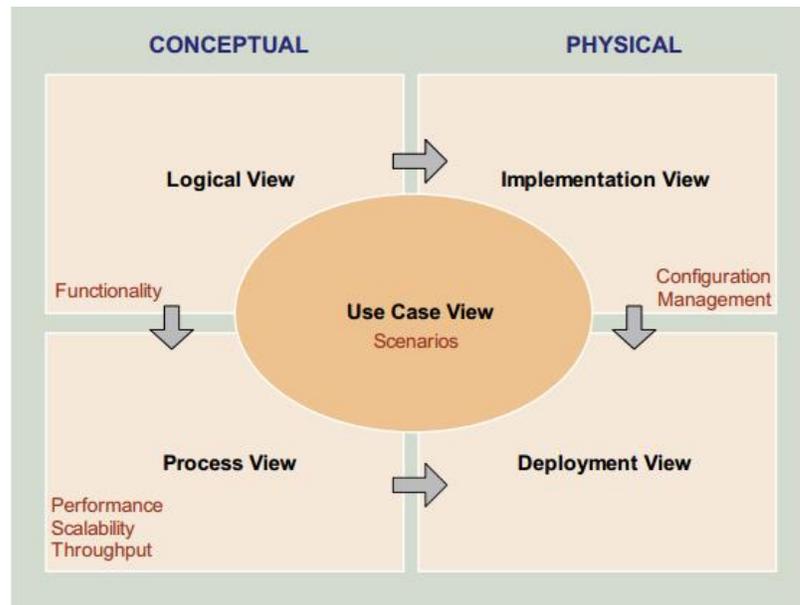
- Comportamiento, representado por las interacciones entre los elementos del sistema.
- Composición entre elementos estructurales y de comportamiento dentro de sistemas mayores.

Los elementos estructurales y de comportamiento están dados por los requerimientos funcionales del sistema, mientras que las composiciones están guiadas por los atributos de calidad o características deseadas de un sistema (representadas por los requerimientos no funcionales). Para ello se requiere el diseño y construcción de una arquitectura que enlace de manera completa dichos elementos.

El modelo de 4+1 vistas tiene como objetivo crear una arquitectura de vistas para organizar la arquitectura de un sistema. Posee cuatro vistas principales divididas en dos grupos, estas son las vistas con enfoque conceptual y las vistas con enfoque físico, y una vista para modelar escenarios.

Dentro de las vistas con enfoque conceptual se encuentran: la vista lógica y la vista de procesos. Mientras que dentro de las vistas con enfoque físico se tienen: la vista de implementación y la de despliegue. La vista de escenarios (también llamada de casos de uso) se encuentra en ambos grupos y se relaciona con las otras cuatro vistas. En la siguiente imagen se muestra la relación entre ellas.

Figura 18. **Modelo de arquitectura de 4+1 vistas**



Fuente: SPYRAX Systems. *Applying 4+1 View Architecture with UML 2*. p. 3.

Para la aplicación *HealthyFood* se propuso la utilización del modelo de 4+1 vistas que provee una manera compleja, pero ordenada de documentar el desarrollo de cualquier sistema de software.

A continuación se describe cada una de las vistas del modelo. Además, la implementación dentro del desarrollo de *HealthyLife*, incluyendo los diagramas UML que implica ello.

3.2.1. Vista de casos de uso

Es la integración de las cuatro vistas del modelo 4+1 vistas. Es la vista central que captura los escenarios en que se identifican las necesidades y requerimientos de la aplicación, describiendo el comportamiento del sistema y

la interacción que este tendrá con los usuarios finales o con otros *stakeholders*. Es la primera vista del sistema que se deberá crear.

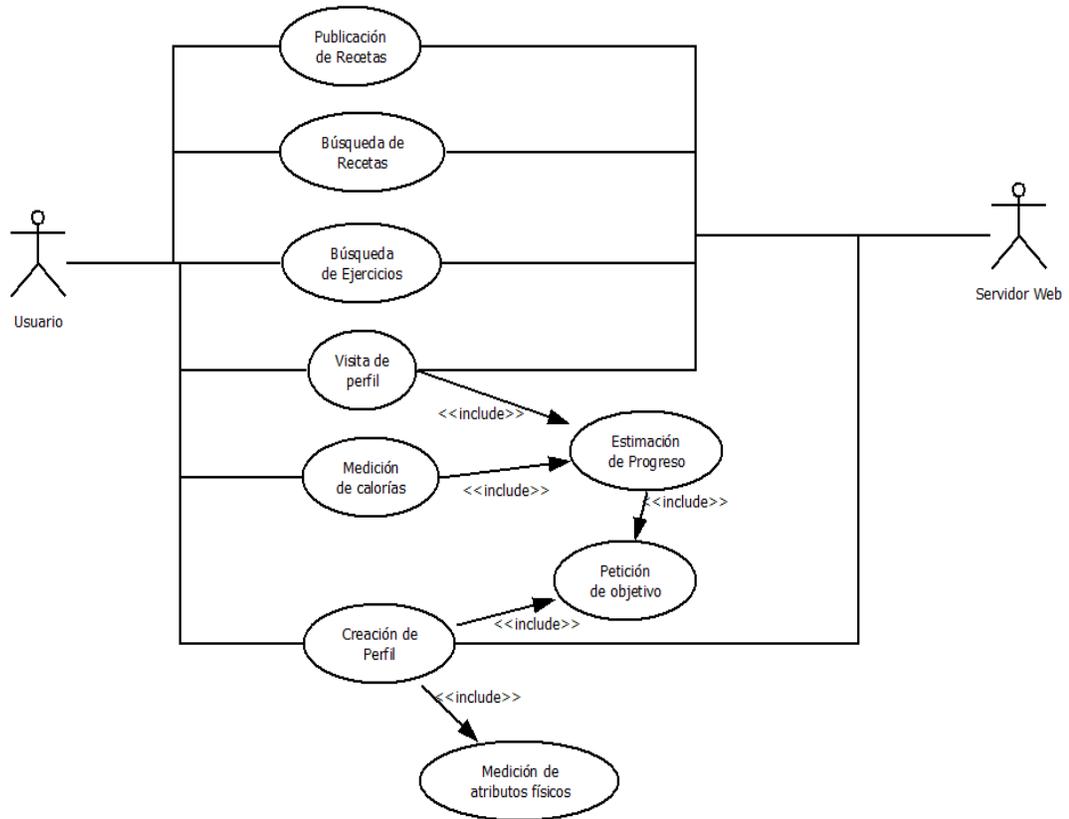
En el caso de *HealthyFood*, la vista de casos de uso muestra la relación que tendrá el usuario final con la aplicación. Además, muestra cómo es proveída de servicios por un agente o actor externo.

Para esta vista, dentro de UML se ha definido un diagrama específico dentro de los diagramas de comportamiento, el cual es el diagrama de casos de uso.

3.2.1.1. Diagrama de casos de uso extendido del sistema

A continuación se presenta el diagrama de casos de uso extendido del sistema. Este diagrama representa las interacciones del usuario de la aplicación con la misma, mostrando así los módulos de software a implementar.

Figura 19. Diagrama de casos de uso extendido



Fuente: elaboración propia, con programa DIA.

3.2.1.2. Definición de casos de uso

Cada caso de uso se representa por un óvalo, para cada uno se detallan los aspectos necesarios para llevar a cabo el desarrollo del mismo.

- Autenticación

Figura 20. **Diagrama de caso de uso de autenticación**



Fuente: elaboración propia, con programa DIA.

Tabla III. **Definición de caso de uso de autenticación**

Caso de uso	Autenticación
Actores	Usuario de la aplicación, servidor web
Propósito	Identificar de manera única e inequívoca a cada usuario que ingrese a la aplicación.
Resumen	Cada usuario deberá ingresar su nombre de usuario y su contraseña. A partir de ello, se autorizará el ingreso a los demás módulos de la aplicación representados por posteriores casos de uso.

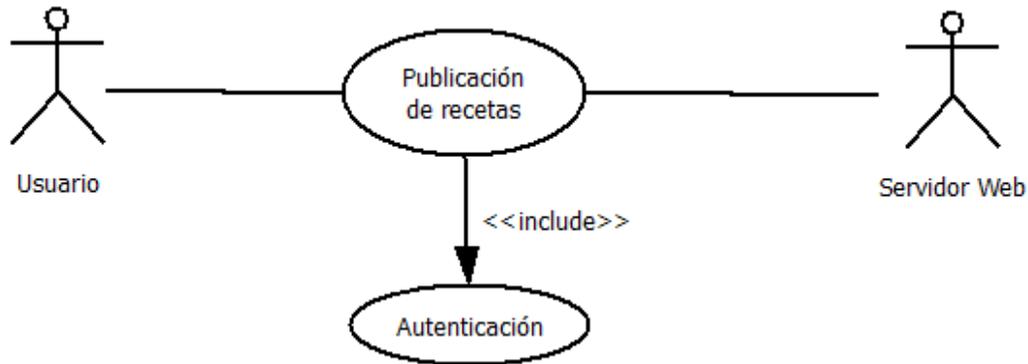
Continuación de la tabla III.

Flujo principal	<p>El usuario ingresa su nombre de usuario y su contraseña. Posterior a ello podrá autenticarse con el botón de la acción correspondiente a la autenticación. La aplicación enviará la información al servidor web, el cual, mediante su propia lógica de manejo de datos determinará la autenticidad del usuario, devolviendo la información del mismo. Una vez autenticado, la aplicación permitirá su paso a cualquiera de los otros casos de uso. Si la autenticación la realiza con redes sociales, el usuario no deberá ingresar ningún dato. Sin embargo el procedimiento con el servidor web será el mismo.</p>
Excepciones	<p>E1 - Nombre de usuario o contraseña inválida.</p> <p>E2 - No hay conexión a internet.</p>

Fuente: elaboración propia.

- Publicación de recetas

Figura 21. **Diagrama de caso de uso de publicación de recetas**



Fuente: elaboración propia, con programa DIA.

Tabla IV. **Definición de caso de uso de publicación de recetas**

Caso de uso	Publicación de recetas
Actores	Usuario de la aplicación, servidor web
Casos de uso relacionados	Autenticación
Propósito	Permitir a los usuarios de la aplicación crear recetas que serán enviadas al sistema servidor para su almacenamiento.

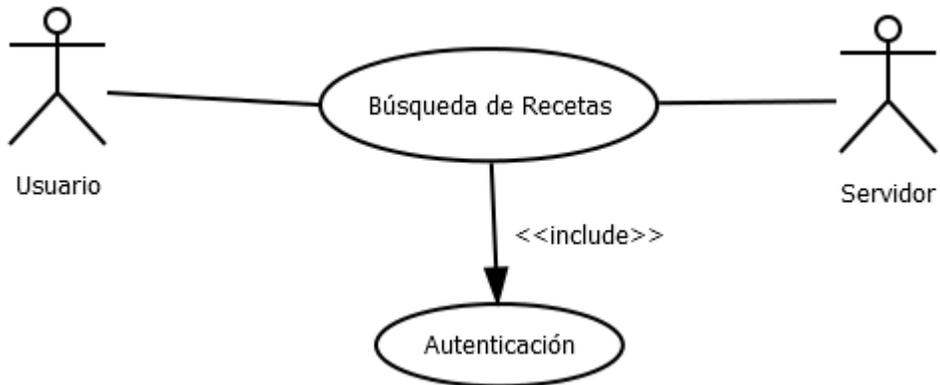
Continuación de la tabla IV.

Resumen	El caso de uso de publicación de recetas permitirá a los usuarios crear una nueva receta que se almacenará en el repositorio de datos utilizado por la aplicación. Los usuarios podrán seleccionar los ingredientes y la cantidad de los mismos, además de incluir una descripción de la preparación de la receta y una foto del resultado de la misma.
Precondiciones del flujo principal	Antes de acceder al caso de uso de publicación de recetas, los usuarios deben haberse autenticado a partir del caso de uso de Autenticación.
Flujo principal	<p>El usuario coloca el nombre de la receta.</p> <p>Desplegar la lista de ingredientes existentes en el sistema.</p> <p>Seleccionar los ingredientes para la receta.</p> <p>Seleccionar la categoría de la receta.</p> <p>Seleccionar o tomar una fotografía de la receta.</p> <p>Colocar instrucciones de la receta.</p> <p>Enviar la receta para su registro.</p>
Excepciones	<p>E3 - No se llenaron los campos necesarios.</p> <p>E4 - La receta que se está registrando ya existe</p> <p>E5 - No hay conexión a internet.</p>

Fuente: elaboración propia.

- Búsqueda de recetas

Figura 22. Diagrama de caso de uso de búsqueda de recetas



Fuente: elaboración propia, con programa DIA.

Tabla V. Definición de caso de uso de búsqueda de recetas

Caso de uso	Búsqueda de recetas
Actores	Usuario de la aplicación, servidor web
Casos de uso relacionados	Autenticación
Propósito	Mostrar resultados deseados a un usuario con base en las palabras clave proporcionadas por el mismo.

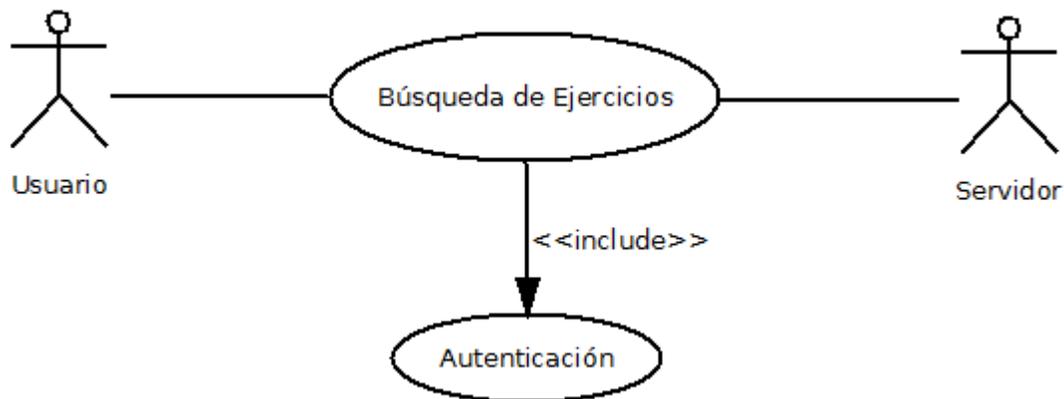
Continuación de la tabla V.

Resumen	El caso de uso de búsqueda de recetas permitirá a los usuarios buscar entre las recetas existentes alguna que se adecue a los datos ingresados por el usuario para dicha búsqueda.
Precondiciones del flujo principal	Antes de acceder al caso de uso de búsqueda de recetas, los usuarios deben haberse autenticado a partir del caso de uso de Autenticación.
Flujo principal	<p>El usuario ingresa las palabras clave para la búsqueda.</p> <p>Se ingresa datos como el rango de calorías que quiere.</p> <p>Se selecciona la categoría en la cual se hará la búsqueda.</p> <p>Se iniciará la búsqueda.</p> <p>Se desplegarán los resultados en la pantalla.</p>
Excepciones	<p>E6 - No se llenaron los campos necesarios.</p> <p>E7 - No se encontraron resultados disponibles</p> <p>E8 - No hay conexión a Internet.</p>

Fuente: elaboración propia.

- Búsqueda de ejercicios

Figura 23. Diagrama de caso de uso de búsqueda de ejercicios



Fuente: elaboración propia, con el programa DIA.

Tabla VI. Definición de caso de uso de búsqueda de ejercicios

Caso de uso	Búsqueda de ejercicios
Actores	Usuario de la aplicación, servidor web
Casos de uso relacionados	Autenticación
Propósito	Muestra un ejercicio con base en los datos proporcionados por el usuario
Resumen	El caso de uso de búsqueda de ejercicios permitirá a los usuarios seleccionar uno de los ejercicios desplegados por la aplicación.

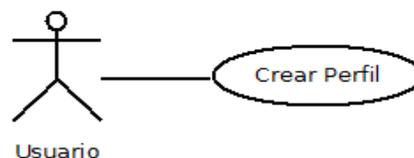
Continuación de la tabla VI.

Precondiciones del flujo principal	El usuario debe haberse autenticado con anterioridad para usar esta opción.
Flujo principal	El usuario ingresa las palabras clave para la búsqueda. Se selecciona la categoría en la cual se hará la búsqueda. Se iniciará la búsqueda. Se desplegarán los resultados en la pantalla.
Excepciones	E9 - No se llenaron los campos necesarios. E10 - No se encontraron resultados disponibles. E11 - No hay conexión a internet.

Fuente: elaboración propia.

- Creación de Perfil

Figura 24. **Caso de uso de creación de Perfil**



Fuente: elaboración propia, con programa DIA.

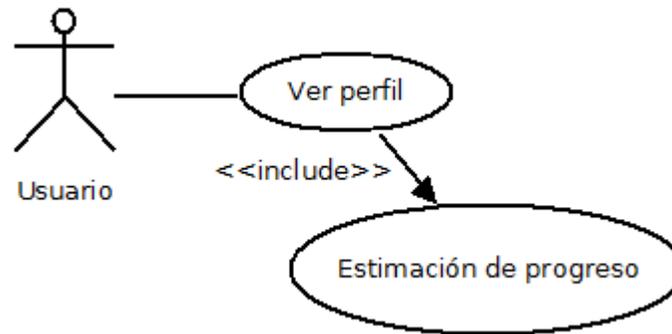
Tabla VII. **Definición de caso de uso de creación de perfil**

Caso de uso	Creación de Perfil
Actores	Usuario de la aplicación, servidor web
Casos de uso relacionados	Autenticación
Propósito	Crear un perfil de usuario con toda la información necesaria para hacer uso de las demás funciones de la aplicación
Resumen	El caso de uso de creación de Perfil.
Precondiciones del flujo principal	Ninguna.
Flujo principal	El usuario selecciona la opción de Crear un Perfil. Ingresar sus datos. Obtiene una confirmación de usuario creado
Excepciones	E12 - No se llenaron los campos necesarios. E13 - Hubo un error en algunos de los campos escritos. E14 - No hay conexión a internet.

Fuente: elaboración propia.

- Ver perfil

Figura 25. **Diagrama de caso de uso de ver Perfil**



Fuente: elaboración propia, con programa DIA.

Tabla VIII. **Definición de caso de uso de ver Perfil**

Caso de uso	Ver Perfil
Actores	Usuario de la aplicación, servidor web
Casos de uso relacionados	Autenticación
Propósito	Mostrar la información de Perfil del usuario.
Resumen	El caso de uso ver Perfil se encarga de mostrar al usuario información sobre su peso, altura y la estimación de su progreso reciente.

Continuación de la tabla VIII.

Precondiciones del flujo principal	Antes de acceder al caso de uso de vista de Perfil, los usuarios deben haberse autenticado a partir del caso de uso de Autenticación.
Flujo principal	El usuario selecciona la opción de ver su Perfil. Se despliega toda la información del usuario en la pantalla.
Excepciones	E15 - Se perdió la conexión a internet. E16 - No se encuentran los datos del usuario dentro de la base de datos

Fuente: elaboración propia.

3.2.2. Vista lógica

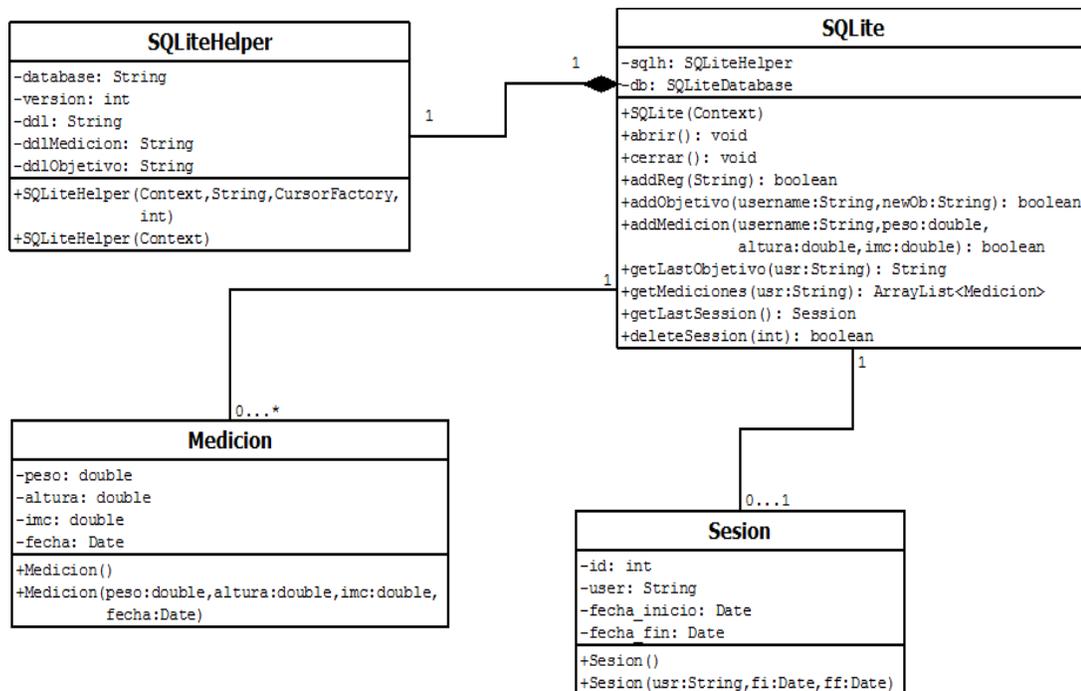
Dentro del modelo, esta vista se enfoca en la realización de la funcionalidad de la aplicación en términos de sus elementos estructurales a nivel lógico. Se centra principalmente en el análisis y diseño de los elementos funcionales de la aplicación.

Para modelar la vista lógica de la aplicación se utilizarán algunos diagramas UML pertenecientes a la categoría de diagramas estructurales. Entre ellos están: diagrama de clases y diagrama de paquetes.

3.2.2.1. Diagramas de clases de la aplicación

A continuación se presentan los diagramas de clases pertenecientes a la aplicación *HealthyFood* dados por los distintos paquetes en los que se divide. Además, se presenta la definición de cada clase.

Figura 26. Diagrama de clases del paquete “healthyfood.sqlite”



Fuente: elaboración propia, con programa DIA.

A continuación se muestra la definición de cada una de las clases.

Tabla IX. **Clase Medición**

Descripción	Clase encargada de crear objetos que contengan la información de cada medición realizada por el usuario.
Atributos	<ul style="list-style-type: none">• peso: almacena el peso del usuario en la medición.• altura: almacena la altura del usuario en la medición.• imc: número de coma flotante asociado al índice de masa corporal del usuario.• fecha: atributo encargado de almacenar la fecha en la que se realizó la medición.
Acciones	<ul style="list-style-type: none">• Medicion: método constructor encargado de inicializar los objetos de la clase. Posee dos sobrecargas.
Relaciones	<ul style="list-style-type: none">• La clase se encuentra asociada con la clase SQLite.

Fuente: elaboración propia.

Tabla X. **Clase Sesión**

Descripción	Clase encargada de la creación de sesiones internas para evitar el inicio de sesión cada vez que el usuario entre en la aplicación.
Atributos	<ul style="list-style-type: none"> • id: es el identificador único de la sesión que será almacenada en la base de datos interna. • user: atributo que almacena el nombre del usuario que inicia la sesión. • fecha_inicio: fecha en la que se inicia la sesión. • fecha_fin: fecha en la que se finaliza la sesión.
Acciones	<ul style="list-style-type: none"> • <i>Sesion</i>: es el constructor de la clase, tiene dos sobrecargas.
Relaciones	<ul style="list-style-type: none"> • Está asociada con la clase SQLite.

Fuente: elaboración propia.

Tabla XI. **Clase SQLiteHelper**

<p>Descripción</p>	<p>Clase encargada de crear y actualizar la base de datos interna de la aplicación.</p>
<p>Atributos</p>	<ul style="list-style-type: none"> • database: representa el nombre de la base de datos interna de la aplicación. • versión: es la versión de la base de datos. • ddl: cadena de lenguaje de definición de datos para crear la tabla de sesiones en la base de datos. • ddlMedicion: cadena encargada de crear la tabla de mediciones en la base de datos. • ddlObjetivo: cadena encargada de crear la tabla de objetivos de los usuarios en la base de datos.
<p>Acciones</p>	<ul style="list-style-type: none"> • SQLiteHelper: constructor de la clase, tiene 2 sobrecargas.
<p>Relaciones</p>	<ul style="list-style-type: none"> • La clase SQLite está compuesta por un objeto de tipo SQLiteHelper, el cual no puede existir sin SQLite. • SQLiteHelper hereda de la clase SQLiteOpenHelper cuya implementación está fuera del proyecto de software.

Fuente: elaboración propia.

Tabla XII. **Clase SQLite**

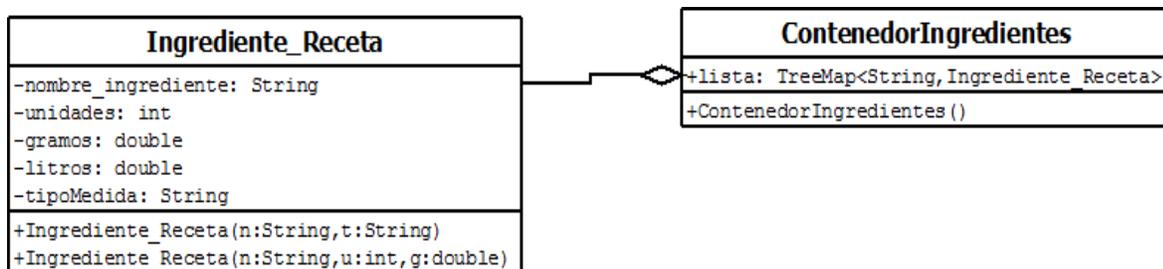
<p>Descripción</p>	<p>Clase encargada de realizar las operaciones necesarias sobre la base de datos interna de la aplicación.</p>
<p>Atributos</p>	<ul style="list-style-type: none"> • sqlh: objeto de la clase SQLiteHelper utilizado para las conexiones a la base de datos. • db: objeto de la clase SQLiteDatabase, crea una instancia del sistema de gestión de base de datos para la aplicación.
<p>Acciones</p>	<ul style="list-style-type: none"> • SQLite: constructor de la clase, tiene solo una sobrecarga. • abrir: método encargado de realizar la apertura de la base de datos interna. • cerrar: método encargado de cerrar la conexión con la base de datos. • addReg: se encarga de agregar un registro en la tabla de sesiones. • addObjetivo: se encarga de agregar un nuevo objetivo para un usuario en específico. • addMedición: se encarga de agregar una nueva medición física del usuario. • getLastObjetivo: obtiene el último objetivo de un usuario. • getMediciones: obtiene las mediciones de un usuario. • getLastSesion: obtiene la última sesión abierta.

Continuación de la tabla XII.

	<ul style="list-style-type: none"> • deleteSesion: destruye la sesión indicada por su id.
Relaciones	<ul style="list-style-type: none"> • Se encuentra asociada con la clase Sesión. • Se encuentra asociada con la clase Medición. • Está compuesta por un objeto de la clase SQLiteHelper. • Posee una agregación de un objeto de tipo SQLiteDatabase. Sin embargo, dicha clase está fuera de la implementación del proyecto.

Fuente: elaboración propia.

Figura 27. Diagrama de clases del paquete “healthyfood.sobj”



Fuente: elaboración propia, con programa DIA.

A continuación se describen las clases involucradas en este paquete.

Tabla XIII. **Clase Ingrediente_Receta**

Descripción	Clase encargada de almacenar en memoria la relación existente entre los ingredientes existentes en el sistema y las recetas.
Atributos	<ul style="list-style-type: none"> • nombre_ingrediente: representa el nombre del ingrediente relacionado con la receta. • unidades: contiene la cantidad de unidades del ingrediente. • gramos: contiene la cantidad de gramos del ingrediente. • litros: contiene la cantidad de litros del ingrediente. • tipoMedida: indica el tipo de medida en la que debe estar el ingrediente dentro de la relación, puede ser gramos, litros o unidades.
Acciones	<ul style="list-style-type: none"> • Ingrediente_Receta: constructor de la clase, tiene 2 sobrecargas.
Relaciones	<ul style="list-style-type: none"> • Está agregada dentro de la clase ContenedorIngredientes.

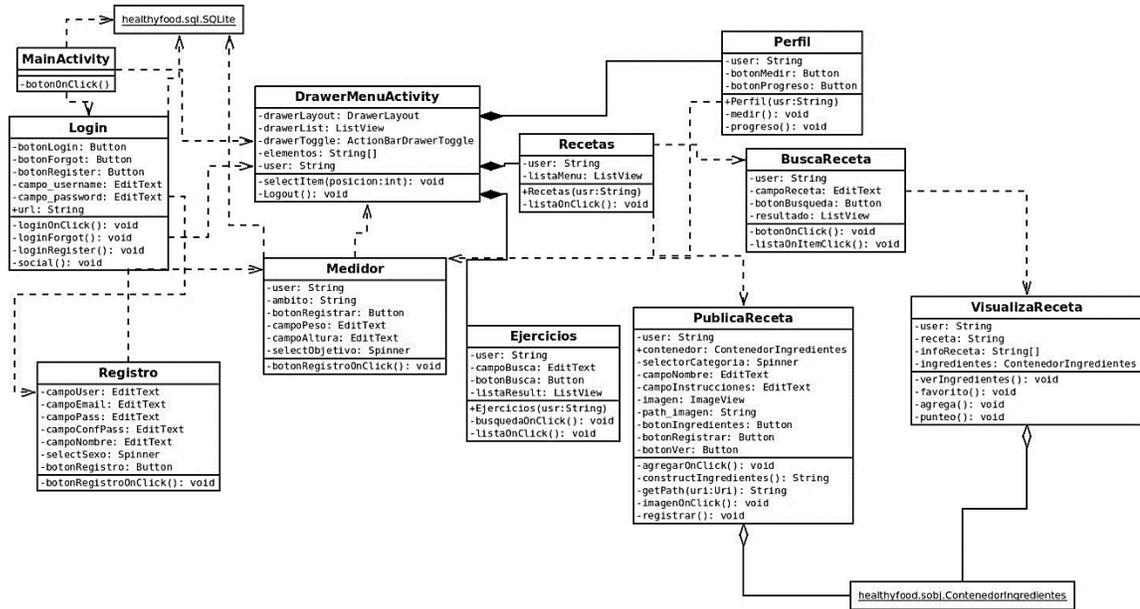
Fuente: elaboración propia.

Tabla XIV. **Clase ContenedorIngredientes**

Descripción	Clase encargada de almacenar de manera ordenada en memoria los ingredientes seleccionados para formar parte de los de una receta.
Atributos	<ul style="list-style-type: none"> • lista: contenedor de tipo TreeMap encargado de almacenar, en memoria, en forma de clave-valor los ingredientes que se relacionan a una receta.
Acciones	<ul style="list-style-type: none"> • ContenedorIngredientes: constructor de la clase.
Relaciones	<ul style="list-style-type: none"> • Posee una relación de agregación con la clase Ingrediente_Receta, de manera que tiene agregado un objeto de dicha clase. • Implementa la interfaz Serializable, cuya definición está fuera de este proyecto.

Fuente: elaboración propia.

Figura 28. Diagrama de clases del paquete “healthyfood”



Fuente: elaboración propia, con programa DIA.

Este paquete es el encargado de la interfaz con el usuario. Cada una de las clases del paquete hereda elementos visuales propios del sistema operativo Android. A continuación se muestra la descripción de cada una de ellas.

Tabla XV. **Clase MainActivity**

Descripción	Clase que representa la actividad principal por la que iniciará la aplicación.
Atributos	No posee atributos.
Acciones	<ul style="list-style-type: none"> • botonOnClick: es la acción encargada de iniciar el contenido de la aplicación.
Relaciones	<ul style="list-style-type: none"> • Dependencia de las clases Login, DrawerMenuActivity y SQLite. • Herencia de Activity, cuya implementación no está en este proyecto.

Fuente: elaboración propia.

Tabla XVI. **Clase Login**

Descripción	Clase encargada de la autenticación del usuario.
Atributos	<ul style="list-style-type: none"> • botonLogin: botón que ejecuta la acción de autenticación. • botonForgot: botón que envía al usuario al apartado para recuperar credenciales. • botonRegister: botón que envía al usuario al apartado de registro. • campoUsername: campo donde se deberá colocar el nombre de usuario único.

Continuación de la tabla XVI.

	<ul style="list-style-type: none">• campoPassword: campo para colocar la contraseña.
Acciones	<ul style="list-style-type: none">• login: efectúa la autenticación.• forgot: ejecuta la acción de envío al formulario de recuperación.• register: ejecuta la acción de envío al formulario de registro.• Social: permite ingresar al sistema a través de una autenticación de Facebook.
Relaciones	<ul style="list-style-type: none">• Dependencia de las clases Registro, SQLite, y DrawerMenuActivity.• Herencia de Activity, cuya implementación no está en este proyecto.

Fuente: elaboración propia.

Tabla XVII. **Clase Registro**

Descripción	Clase encargada del formulario de registro de nuevos usuarios.
Atributos	<ul style="list-style-type: none"> • campoUser: campo para colocar el nombre de usuario único. • campoEmail: campo para colocar el correo electrónico. • campoPass: campo para colocar la contraseña. • campoConfPass: campo para colocar la confirmación de la contraseña. • campoNombre: campo para el nombre real del usuario. • selectSexo: seleccionador de opciones para definir el sexo del usuario. • botonRegistro: botón encargado de ejecutar la acción de registro.
Acciones	<ul style="list-style-type: none"> • botonRegistroOnClick: acción que envía los datos del usuario para su registro.
Relaciones	<ul style="list-style-type: none"> • Dependencia de la clase Medidor. • Herencia de Activity, cuya implementación no está en este proyecto.

Fuente: elaboración propia.

Tabla XVIII. **Clase DrawerMenuActivity**

<p>Descripción</p>	<p>Clase encargada de manejar la interfaz gráfica para el menú de la aplicación. Coloca una barra lateral en la pantalla en la que despliega las opciones del menú de la aplicación.</p>
<p>Atributos</p>	<ul style="list-style-type: none"> • user: nombre de usuario único que ha iniciado la sesión. <p>Posee atributos propios para la creación de la interfaz gráfica en los que no se necesita profundizar.</p> <p>A pesar de no estar dentro de los atributos visibles, posee un contenedor de fragmentos que se ejecutarán como vistas a partir del menú.</p>
<p>Acciones</p>	<ul style="list-style-type: none"> • selectItem: acción ejecutada cuando se selecciona un ítem del menú. • Logout: cierre de sesión por parte del usuario.
<p>Relaciones</p>	<ul style="list-style-type: none"> • Contiene las clases Perfil, Recetas y Ejercicios. • Hereda de la clase ActionBarActivity, cuya implementación está fuera de este proyecto.

Fuente: elaboración propia.

Tabla XIX. **Clase Medidor**

<p>Descripción</p>	<p>Clase encargada de tomar las medidas del usuario, tales como el peso y la altura para calcular su índice de masa corporal.</p>
<p>Atributos</p>	<ul style="list-style-type: none"> • user: usuario que ha iniciado la sesión. • ámbito: actividad anterior que la invocó, puede ser la clase Registro o la clase Perfil. • botonRegistrar: botón que ejecuta la acción de registrar las medidas del usuario. • campoPeso: campo en el que se ingresa el peso del usuario. • campoAltura: campo en el que se ingresa la altura del usuario. • selectObjetivo: le permite al usuario seleccionar el objetivo por el cual utiliza la aplicación.
<p>Acciones</p>	<ul style="list-style-type: none"> • botonRegistroOnClick: lleva a cabo el registro de la medición del usuario.

Continuación de la tabla XIX.

<p>Relaciones</p>	<ul style="list-style-type: none"> • Depende de las clases SQLite y DrawerMenuActivity. • Hereda de la clase Activity, la cual no está implementada en el proyecto.
--------------------------	---

Fuente: elaboración propia.

Tabla XX. **Clase Perfil**

<p>Descripción</p>	<p>Clase encargada de manejar los aspectos referentes al perfil del usuario, tales como la toma de sus medidas y su progreso.</p>
<p>Atributos</p>	<ul style="list-style-type: none"> • user: usuario que ha iniciado la sesión. • botonMedir: botón que permite el cambio a la actividad Medidor. • botonProgreso: botón que realiza la acción de mostrar el progreso.
<p>Acciones</p>	<ul style="list-style-type: none"> • Perfil: es el constructor de la clase, solamente tiene una implementación y ninguna sobrecarga. • medir: muestra una instancia de la clase Medidor. • progreso: muestra el progreso del usuario, utilizando su índice de masa corporal a través del tiempo.

Continuación de la tabla XX.

Relaciones	<ul style="list-style-type: none"> • Dependencia de la clase Medidor. • Conteneda dentro de la clase DrawerMenuActivity. • Hereda de la clase Fragment de Android, la cual no está implementada en el proyecto.
-------------------	--

Fuente: elaboración propia.

Tabla XXI. **Clase Recetas**

Descripción	Clase encargada del manejo de las diversas operaciones a realizar con las recetas.
Atributos	<ul style="list-style-type: none"> • user: usuario que ha iniciado la sesión. • listadoMenú: submenú correspondiente a las acciones a realizar sobre las recetas.
Acciones	<ul style="list-style-type: none"> • Recetas: constructor de la clase sin sobrecargas. • listaOnClick: inicializa la actividad seleccionada.
Relaciones	<ul style="list-style-type: none"> • Depende de las clases BuscaReceta y PublicaReceta. • Conteneda dentro de la clase DrawerMenuActivity. • Hereda de la clase Fragment, cuya implementación no está incluida dentro del proyecto.

Fuente: elaboración propia.

Tabla XXII. **Clase Ejercicios**

Descripción	Clase encargada de mostrar los ejercicios recomendados al usuario con base en el objetivo actual planteado.
Atributos	<ul style="list-style-type: none"> • user: nombre del usuario que inició la sesión. • campoBusca: campo en el que el usuario ingresa el ejercicio que desea encontrar. • botonBusca: botón que activa la acción para buscar un ejercicio. • listaResult: lista en la que se mostrarán los ejercicios encontrados.
Acciones	<ul style="list-style-type: none"> • Ejercicios: constructor de la clase. • busquedaOnClick: realiza la búsqueda de los ejercicios que cumplan con el patrón de búsqueda. • listaOnClick: muestra el ejercicio seleccionado.
Relaciones	<ul style="list-style-type: none"> • Contenida dentro de la clase DrawerMenuActivity. • Hereda de la clase Fragment, cuya implementación no se encuentra dentro del proyecto.

Fuente: elaboración propia.

Tabla XXIII. **Clase PublicaReceta**

<p>Descripción</p>	<p>Clase encargada de publicar recetas de los usuarios. Estas recetas podrán ser consultadas por cualquier usuario de la aplicación.</p>
<p>Atributos</p>	<ul style="list-style-type: none"> • user: usuario que registra la receta. • contenedor: contenedor de ingredientes asociados a la receta. • selectorCategoría: permite seleccionar la categoría a la que pertenece la receta. • campoNombre: nombre de la receta (es único). • campoInstrucciones: incluye las instrucciones para realizar la receta. • imagen: vista de imagen de la receta. • path_imagen: ruta de la imagen de la receta. • botonIngredientes: permite activar la acción de agregar o quitar ingredientes. • botonRegistrar: permite activar la acción de registrar o publicar la receta. • botonVer: permite activar la acción para encargada de verificar los ingredientes asociados a la receta.

Continuación de la tabla XXIII.

Acciones	<ul style="list-style-type: none">• agregarOnClick: agrega o quita ingredientes a la receta.• constructIngredientes: permite construir la visualización de los ingredientes de la receta.• getPath: obtiene la dirección de la imagen en el dispositivo móvil.• imagenOnClick: permite seleccionar una imagen desde el dispositivo para agregarla a la receta.• registrar: publica la receta en el servidor remoto para que otros usuarios puedan acceder a ella.
Relaciones	<ul style="list-style-type: none">• Tiene agregado un objeto de la clase ContenedorIngredientes.• Hereda de la clase Activity, la cual no está implementada en el proyecto.

Fuente: elaboración propia.

Tabla XXIV. **Clase BuscaReceta**

Descripción	Clase encargada de realizar la búsqueda de recetas existentes dentro del sistema remoto para cualquier usuario.
Atributos	<ul style="list-style-type: none"> • user: nombre del usuario que ha iniciado la sesión. • campoReceta: campo en el que se ingresa el patrón de búsqueda para las recetas. • botonBusqueda: botón que realiza la acción de búsqueda. • resultado: lista de recetas en la que se almacena el resultado de la búsqueda.
Acciones	<ul style="list-style-type: none"> • botonOnClick: realiza la petición de búsqueda al servidor externo. • listaOnClickItem: selecciona una receta y la muestra en una instancia de la clase VisualizaReceta.
Relaciones	<ul style="list-style-type: none"> • Depende de la clase VisualizaReceta. • Hereda de la clase Activity, la cual no está implementada dentro del proyecto.

Fuente: elaboración propia.

Tabla XXV. **Clase VisualizaReceta**

Descripción	Clase encargada de mostrar el contenido de una receta publicada en el servidor externo.
Atributos	<ul style="list-style-type: none"> • user: nombre del usuario que inició la sesión. • receta: nombre de la receta que se muestra. • infoReceta: contiene toda la información de la receta de forma lógica, la cual es mostrada en los campos de de interfaz gráfica de la actividad. • ingredientes: contenedor de ingredientes relacionados a la receta.
Acciones	<ul style="list-style-type: none"> • verIngredientes: muestra los ingredientes asociados a la receta. • favorito: permite agregar a la lista de recetas favoritas del usuario la receta mostrada. • agrega: permite agregar al consumo diario del usuario la receta. • punteo: permite realizar la acción de punteo de la receta.

Fuente: elaboración propia.

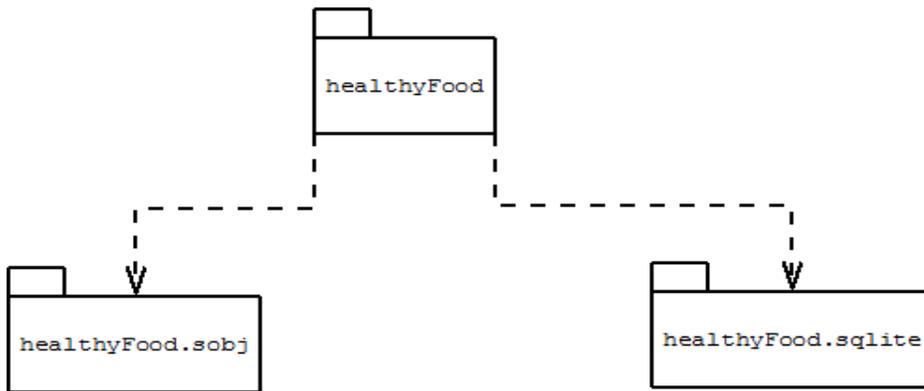
3.2.2.2. Diagrama de paquetes de la aplicación

La aplicación está comprendida por tres paquetes principales: healthyfood, que incluye la interfaz gráfica de toda la aplicación, subj, que contiene las clases referentes a los ingredientes; y sqlite, que contiene las clases

correspondientes al almacenamiento de información referente a las sesiones y mediciones del usuario en el dispositivo móvil.

Los paquetes están relacionados entre sí, siendo el paquete central el de la interfaz gráfica. A continuación se muestra el diagrama de paquetes de la aplicación.

Figura 29. **Diagrama de paquetes de la aplicación**



Fuente: elaboración propia, con programa DIA.

El paquete healthyFood tiene dependencia de los paquetes sobj y sqlite debido a que los elementos de la interfaz gráfica realizan acciones que utilizan instancias de clases incluidas en dichos paquetes, tal a como se muestra en el apartado anterior con los diagramas de clases.

3.2.3. Vista de procesos

La vista de procesos considera aspectos no funcionales o atributos de calidad tales como performance, escalabilidad y rendimiento. Muestra

abstracciones principales de la vista lógica ejecutándose sobre hilos a través del tiempo como una operación.

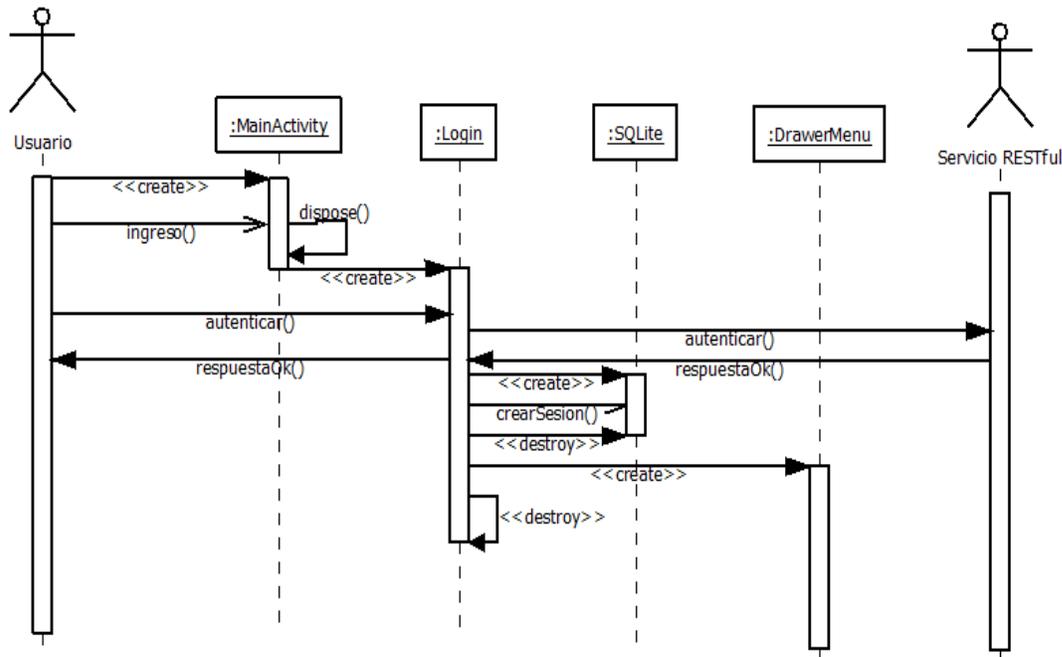
Cada proceso está formado como un grupo de tareas a realizar desde una unidad ejecutable. Un sistema de software está comprendido de múltiples tareas divididas en partes. Cada una se ejecuta en colaboración con diversos elementos estructurales del sistema. Por ello, dentro de la vista de procesos, se modela la forma en la que los elementos de la vista lógica interactúan a nivel de ejecución y no solamente a nivel de estructura.

UML 2 provee de diagramas de comportamiento para construir esta vista. Para la aplicación *HealthyFood* se han creado diagramas de secuencias de los procesos principales, mostrando la interacción entre los elementos estructurales definidos previamente.

3.2.3.1. Diagramas de secuencias

A continuación se muestran las secuencias principales de la aplicación.

Figura 30. Diagrama de secuencia de autenticación



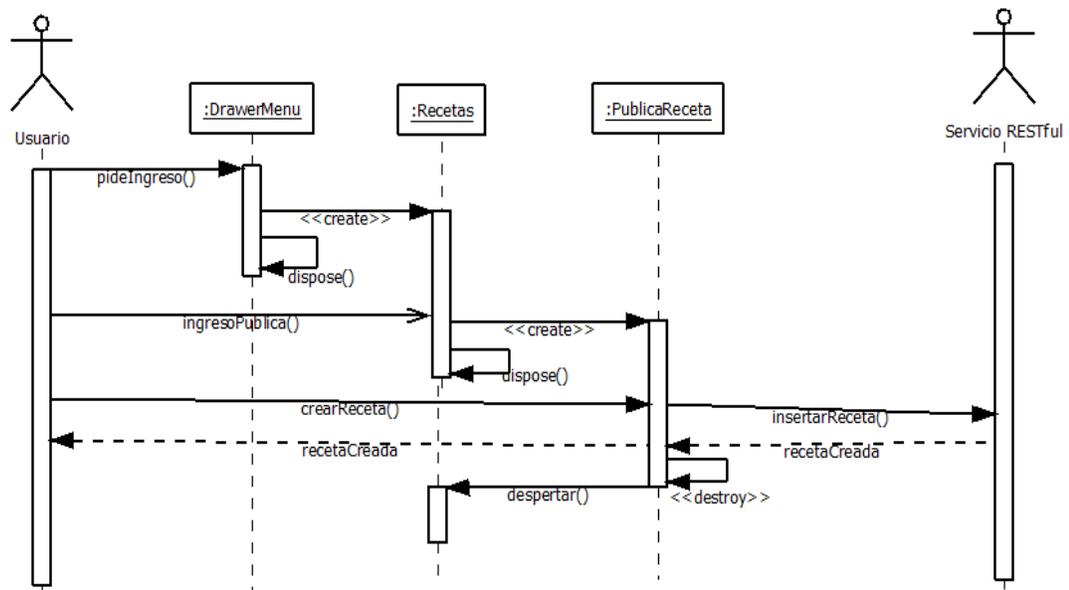
Fuente: elaboración propia, con programa DIA.

A continuación se describe la lógica del diagrama de secuencias de la autenticación:

- El usuario inicia la aplicación creando una instancia de MainActivity.
- El usuario solicita el ingreso a la aplicación, una vez realizada esta acción, el objeto MainActivity se esconde y crea una actividad de Login.
- El usuario solicita su autenticación a la aplicación. La aplicación, a través de la actividad de Login, solicita al servicio RESTful que identifique al usuario entrante.
- EL servicio RESTful envía un mensaje de respuesta para autenticar al usuario en la aplicación.

- La aplicación recibe el mensaje y, por medio de la actividad Login, informa al usuario que se ha autenticado. Al mismo tiempo crea un objeto de SQLite y le ordena que almacene la información de la sesión.
- Una vez almacenada la sesión, la actividad Login solicita la creación del menú principal de la aplicación y se destruye, completando así la secuencia.

Figura 31. **Diagrama de secuencia de publicación de recetas**



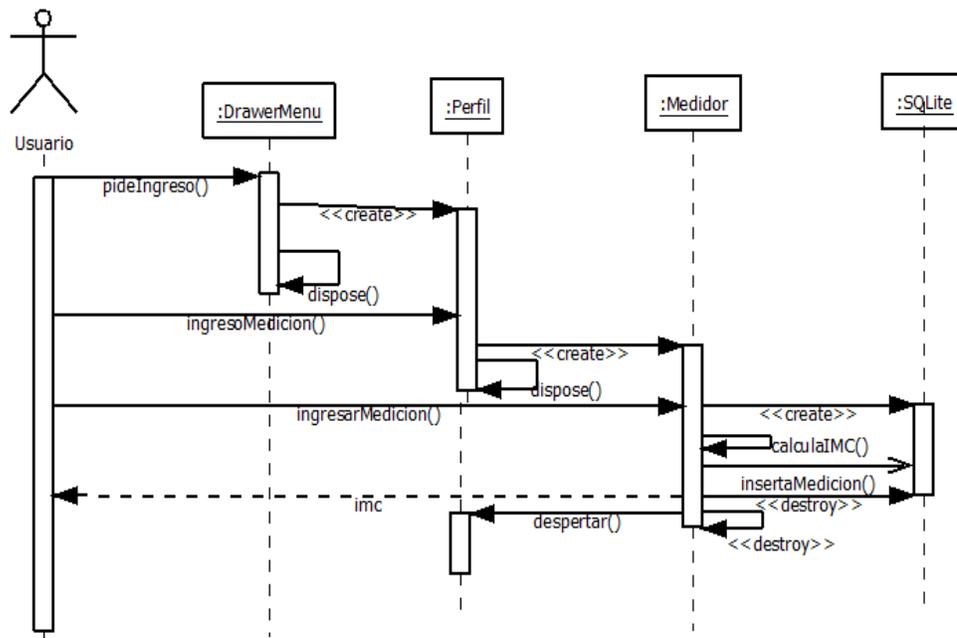
Fuente: elaboración propia, con programa DIA.

A continuación se describe la secuencia necesaria para la publicación de una receta por parte de un usuario:

- El usuario le solicita a la aplicación el ingreso al apartado de recetas desde el menú principal (DrawerMenu).

- El menú principal crea el objeto correspondiente al apartado de recetas y luego se esconde hasta nuevo aviso.
- El usuario solicita ingresar al apartado de publicación de recetas, por lo que el objeto Recetas envía un mensaje de creación al apartado de publicación de recetas. Posterior a ello se esconde hasta nuevo aviso.
- El usuario solicita a la aplicación la inserción de una nueva receta, por lo que, a través de un objeto de tipo PublicaReceta, solicita al servicio RESTful correspondiente la creación de dicha receta.
- El servidor RESTful responde con un mensaje de inserción correcta, por lo que la aplicación notifica al usuario que su inserción fue realizada, inmediatamente se destruye el objeto de tipo PublicaReceta, despertando el apartado de recetas.

Figura 32. Diagrama de secuencia de medición



Fuente: elaboración propia, con programa DIA.

A continuación se describe el comportamiento de la secuencia para realizar mediciones de los usuarios:

- El usuario solicita a la aplicación entrar en el módulo de perfil a través del menú principal.
- El menú principal crea el módulo de perfil y se esconde hasta nuevo aviso.
- El usuario solicita ingresar al módulo de medición. La aplicación, a través de un objeto Perfil crea un objeto Medidor, y coloca el objeto Perfil en suspensión hasta nuevo aviso.
- El usuario solicita el ingreso de una nueva medición, por lo que el Medidor instancia la clase SQLite para almacenar los datos.
- El objeto Medidor realiza el cálculo del IMC antes de almacenar los datos de la medición en la base de datos interna.
- El objeto Medidor envía una confirmación de inserción al mismo tiempo que inserta en la base de datos de SQLite.
- Una vez finalizada la inserción, se destruye el objeto Medidor y el módulo de Perfil despierta.

3.2.4. Vista de despliegue

Esta vista contiene los nodos que conforman la topología de hardware en la cual se ejecuta el sistema de software. Se enfoca principalmente en la distribución y la comunicación de los nodos.

En el caso de *HealthyFood*, la aplicación es ejecutada en dispositivos móviles inteligentes, los cuales representan nodos de la red de comunicación del sistema. Además de ello, la vista de despliegue incluye nodos que representan los servicios prestados a la aplicación.

La forma de representar la vista de despliegue con UML es a través de uno de los diagramas estructurales, el diagrama de despliegue.

3.2.5. Vista de implementación

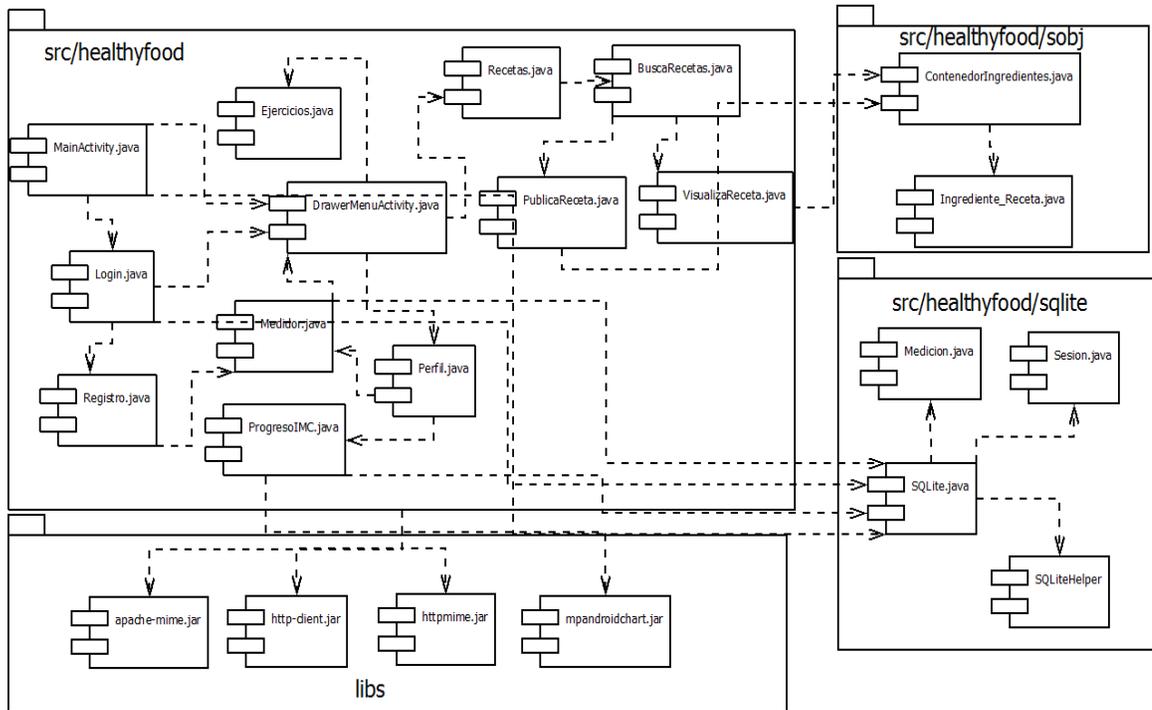
La vista de implementación de la arquitectura de un sistema está compuesta por los elementos utilizados para ensamblar la distribución del sistema físico. Está altamente ligada a la gestión de la configuración del sistema y se enfoca en modelar la organización actual de los componentes del software en su entorno de desarrollo. La vista de implementación muestra la organización física de los elementos estructurales lógicos descritos en la vista lógica.

Para modelar la vista de implementación se utiliza uno de los diagramas estructurales de UML. El diagrama de componentes muestra la configuración del entorno y las dependencias de cada componente para su correcta funcionalidad, en términos de interfaces requeridas y proveídas.

3.2.5.1. Diagrama de componentes

El diagrama de componentes contiene los ficheros de código fuente y las dependencias entre los mismos en el entorno de desarrollo de la aplicación. Se muestran también los directorios y las bibliotecas utilizadas en el proyecto.

Figura 33. Diagrama de componentes de la aplicación

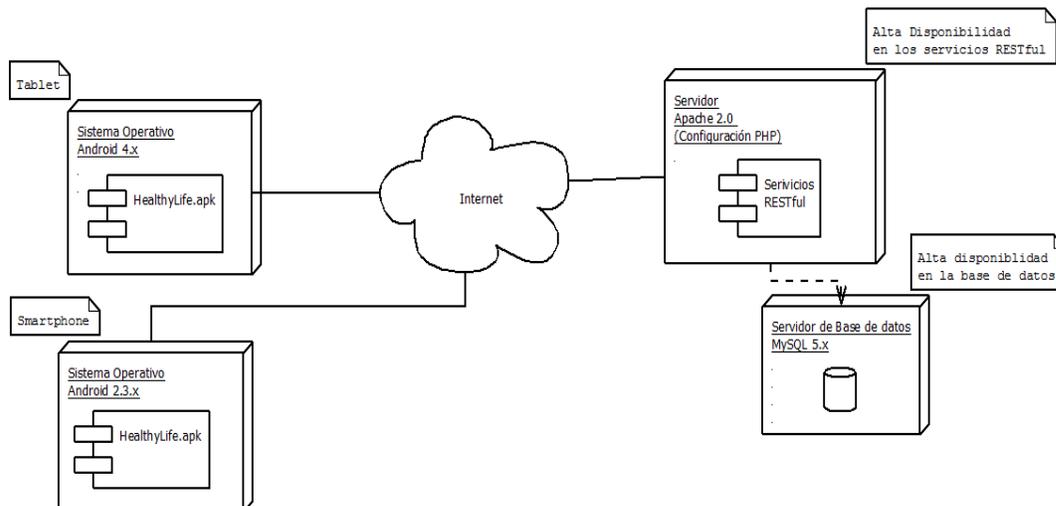


Fuente: elaboración propia, con programa DIA.

3.2.5.2. Diagrama de despliegue

La conexión con los servicios RESTful se realizará a través de internet. Por ello los dispositivos móviles deben tener una conexión de banda ancha móvil o estar conectados a una red inalámbrica que provea conexión a internet, para utilizar de manera correcta y satisfactoria la aplicación.

Figura 34. Diagrama de despliegue para el sistema completo



Fuente: elaboración propia, con programa DIA.

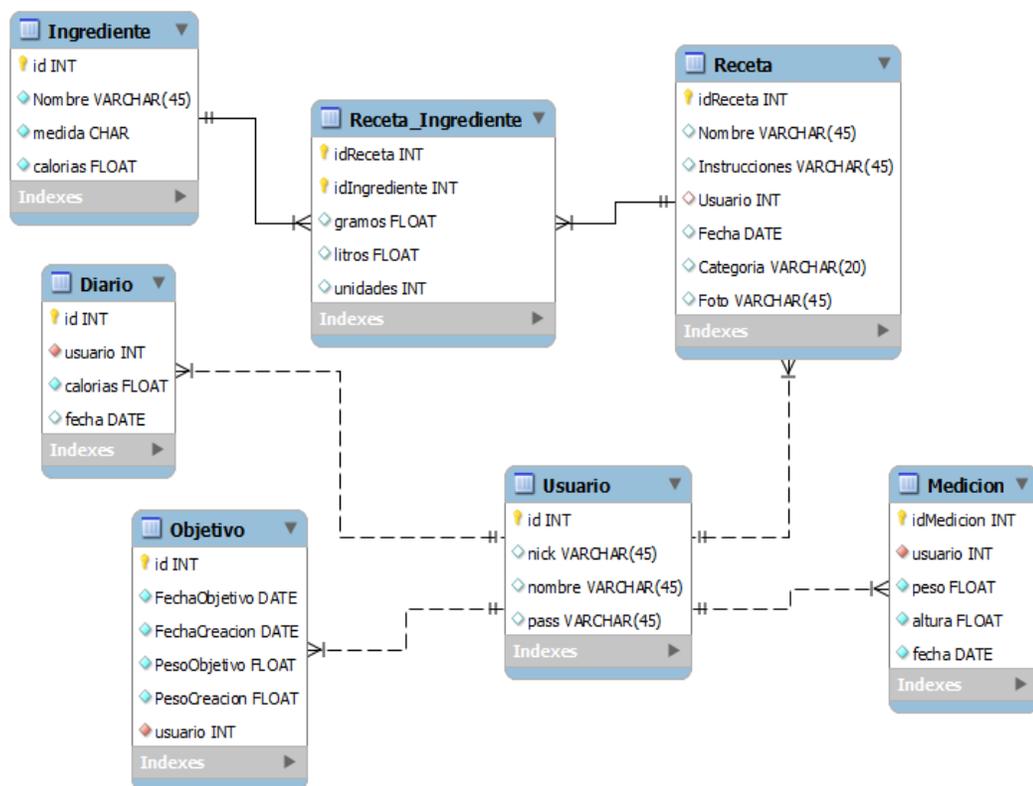
La aplicación es desarrollada para dispositivos móviles con sistema operativo Android, versión 4,2. Sin embargo, se plantea la utilización de bibliotecas de compatibilidad con versiones inferiores del sistema.

3.3. Modelo relacional de base de datos

HealthyFood está diseñado para almacenar toda la información de los usuarios dentro de una base de datos en la nube, accediendo a ella por medio

de internet. El diagrama relacional de la base de datos de la aplicación fue planteado de la siguiente manera:

Figura 35. **Modelo relacional de la base de datos remota**



Fuente: elaboración propia, con programa MySQL Workbench.

Las tablas dentro de la base de datos fueron creadas obedeciendo aspectos específicos de la aplicación.

La tabla usuario fue creada para llevar el control de los usuarios registrados y su información de inicio de sesión. Dentro de ella se almacena la información referente al *nickname* o apodo, nombre y contraseña. Cuando una

persona inicia sesión se accede a los datos de esta tabla para realizar la validación de usuario.

La tabla receta sirve para almacenar la información de todas las recetas creadas por los usuarios, su descripción, categoría y quienes fueron los creadores. Esta se relaciona con la tabla receta-ingrediente que a su vez está relacionada con la tabla ingrediente.

Las llaves primarias de receta-ingrediente son los identificadores de la tabla receta e ingrediente, esto se hizo ya que se quiere evitar que el mismo ingrediente fuera añadido varias veces a la misma receta. Dentro de los valores que se almacenan en la tabla ingrediente son la cantidad de calorías que contiene el ingrediente en específico y su unidad de medida según se indica en las recetas.

La tabla diario fue creada para ver el consumo de los usuarios de calorías según la fecha en que se ingresó el registro en particular. Esto se hace para ver cuántas calorías se han consumido durante un día o una fecha en específico aplicando filtros de fecha inicial y fecha final. Para ello también se creó la tabla Objetivo que indica cuando un objetivo comenzará y terminará, así como el progreso del usuario desde que se registró hasta el día actual.

4. ASPECTOS TÉCNICOS DE LA APLICACIÓN

4.1. Servicios de datos

Existen diversas formas para proveer y almacenar los datos requeridos por una aplicación. El objetivo principal de un servicio de datos es evitar que los clientes almacenen datos sensibles dentro de sus terminales, por lo que, siempre que se utilice una aplicación destinada a ser cliente, debe existir algún servidor que provea los datos necesarios. Generalmente, esto se puede llevar a cabo utilizando un Sistema Gestor de Base de Datos (SGBD).

La aplicación *HealthyFood* almacena datos de dos maneras, la primera es internamente, utilizando una base de datos SQLite para administrar las sesiones abiertas. La otra es utilizando una base de datos MySQL en la nube, en la que se almacenan los datos de todos los usuarios de la aplicación, utilizando el modelo relacional descrito anteriormente.

4.1.1. SQLite

Es un sistema de gestión de base de datos relacional (SGBDR) incluido como una biblioteca de programación del lenguaje C. La característica principal se basa en que, el sistema en sí, no está separado de la aplicación cliente que lo ejecuta, formando parte de ella al momento de la ejecución. Fue creada por Dwayne Richard Hipp en el 2000 y su código está disponible bajo una licencia de dominio público.

Se caracteriza por ser un sistema que proporciona transacciones ACID (*Atomicity, Consistency, Isolation, Durability*) e implementa la mayoría de las instrucciones estándar de SQL. Su uso se generaliza como una base de datos embebida dentro de programas como navegadores, sistemas operativos y sistemas embebidos, entre otros.

Dentro del sistema operativo Android, SQLite está incluida como una de las bibliotecas del sistema. Esto permite su utilización en el desarrollo de aplicaciones que necesiten almacenar información en dispositivo móvil. Sin embargo, Android no es el único sistema operativo móvil que la utiliza, ya que BlackBerry 10 OS, Windows Phone 8, iOS y Symbian también la incluyen dentro de su kit de desarrollo de software.

4.1.2. MySQL

Es un sistema de gestión de base de datos relacional (SGBDR) de código abierto, creado por una compañía sueca, MySQL AB, que actualmente es desarrollado por Oracle Corporation. Hasta marzo de 2014, MySQL ha sido el segundo sistema de gestión de base de datos relacional de código abierto más utilizado a nivel mundial.

Es uno de los sistemas de base de datos más populares para el desarrollo de aplicaciones web. Este es uno de los principales componentes del conjunto de aplicaciones de código abierto conocido como LAMP (Linux, Apache, MySQL, PHP/Perl/Python), además de todas las variantes de “AMP”.

Dentro de las aplicaciones que utilizan MySQL es posible mencionar varios gestores de contenido como Joomla, WordPress y Drupal. Además de

sitios cuya utilización es masiva y de gran escala como Google, Twitter, Facebook y YouTube.

MySQL permite la ejecución de transacciones ACID. Además de utilizar el estándar de SQL para sus diferentes operaciones. Posee una gran cantidad de interfaces gráficas que permiten la administración del sistema, incluida la línea de comandos por defecto. Dentro de los paneles de administración con entorno gráfico se encuentran: MySQL Workbench, Adminer, phpMyAdmin, entre otros.

4.2. Servicios RESTful

Las siglas REST responden al acrónimo de *Representational State Transfer*, el cual es un estilo de arquitectura de la *World Wide Web*. Este consiste en un conjunto coordinado de restricciones de arquitectura aplicadas a componentes, conectores y elementos de datos, distribuido a través de un sistema de hypermedia.

REST no necesita conocer el formato en el que se envían los mensajes, ya que se enfoca únicamente en los roles, las restricciones y la interacción entre componentes. También en la interpretación de elementos de datos significativos.

Como arquitectura, REST posee las siguientes propiedades:

- *Performance*, factor dominante en el que se puede percibir del lado del usuario la eficiencia de las transferencias en la red.
- Simplicidad de interfaces.

- Escalabilidad para soportar una gran cantidad de componentes e interacciones entre los mismos.
- Modificabilidad de los componentes.
- Visibilidad de la comunicación entre componentes.
- Portabilidad de los componentes.
- Confiabilidad.

4.2.1. Aplicación de REST en Web Services

Los servicios Web que implementan la arquitectura REST como medio de transmisión de información son llamados RESTful. Estos servicios utilizan HTTP como protocolo de transferencia, así como los métodos de dicho protocolo.

Los métodos del protocolo HTTP más utilizados por una API RESTful son el GET y el POST. Sin embargo, también utiliza los métodos PUT y DELETE.

A diferencia de los servicios web basados en SOAP, no existe un estándar oficial para desarrollar APIs RESTful. Esto debido a que REST es un estilo de arquitectura, mientras que SOAP es un protocolo, lo que le permite que dichas API's utilicen cualquier tipo de dato de internet para la transferencia de mensajes, tales como JSON, XML, micro formatos, imágenes, entre otros.

4.3. Plataforma Android

Android, más allá de ser un sistema operativo. Es toda una plataforma creada con el objetivo de brindar un sistema operativo en común para múltiples dispositivos móviles inteligentes, los cuales poseen una alimentación de energía de una batería y que están provistos de hardware complejo como GPS, cámara, acelerómetro, conexión a internet, pantallas táctiles, entre otros. Además,

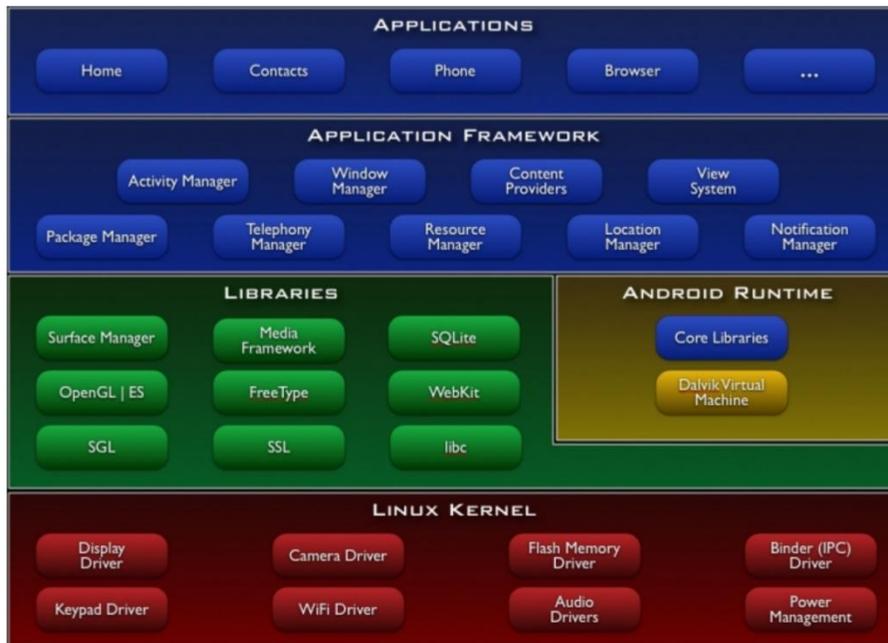
cuentan con procesadores con arquitecturas capaces de soportar el manejo de instrucciones complejas, tal es el caso de la arquitectura ARM. Gracias a su lanzamiento, un grupo de 78 empresas conformaron la *Open Handset Alliance* (OHA), con el objetivo de evolucionar y distribuir el sistema.

Android se diferencia técnicamente de otros sistemas operativos por diversos motivos, uno de los más destacados es el desarrollo de aplicaciones. Estas, por lo general, son escritas en el lenguaje de programación Java y son ejecutadas por el sistema dentro de una máquina virtual. Estas aplicaciones son distribuidas, en su mayoría, en la tienda multimedia de Google, Google Play Store.

4.3.1. Arquitectura del sistema operativo

El sistema operativo móvil Android está construido sobre una arquitectura de cinco capas. Estas permiten la inclusión de distintos niveles de abstracción al funcionamiento de los dispositivos que operan bajo él. Las capas incrementan su nivel de complejidad, mientras disminuyen su nivel de abstracción, siendo la primera capa la encargada de administrar el hardware y la última, la encargada de interactuar con los usuarios finales. En la siguiente figura se muestra la división por capas del sistema.

Figura 36. **Arquitectura del sistema operativo Android**



Fuente: BRÄHLER, Stefan. *Analysis of the Android Architecture*. p. 4.

A continuación, se describe cada una de las capas, desde el nivel inferior al superior, de la arquitectura de Android:

4.3.1.1. Núcleo Linux

Es un núcleo de sistema operativo, basado en el núcleo de UNIX, del tipo monolítico, de código libre y abierto, desarrollado inicialmente por Linus Torvalds. Es utilizado en una amplia variedad de dispositivos en conjunto con sistemas operativos con distintos propósitos, debido que, a pesar de ser un núcleo monolítico, permite la carga de módulos de forma dinámica sin necesidad de compilar por completo el código, administrando, de una forma rápida, múltiples dispositivos periféricos. Los sistemas operativos que más

implementan su uso son los de la familia GNU/Linux, aunque también se tienen sistemas operativos enfocados a dispositivos móviles, tal cual es el caso de Android.

Dentro de la arquitectura de Android, el núcleo Linux es la capa que posee interacción directa con el hardware. Esta se encarga de administrar la energía del dispositivo, la gestión de memoria, de procesamiento, de almacenamiento, y de los dispositivos de entrada y salida.

Android ha utilizado las versiones 2,6 y 3,x del núcleo para las diferentes versiones del sistema operativo.

4.3.1.2. Bibliotecas

Capa compuesta por bibliotecas escritas en C y C++, siendo compiladas de forma nativa para los diversos dispositivos que soportan el sistema operativo.

A diferencia de la capa descrita anteriormente, el nivel de abstracción de las bibliotecas de ella es más alto. Por ello se exponen funcionalidades específicas a los desarrolladores de aplicaciones, tales como bibliotecas de gráficos 2D, 3D y bases de datos SQLite.

4.3.1.3. Runtime de Android

Esta capa está compuesta por bibliotecas necesarias para establecer un entorno de ejecución necesario para las diversas aplicaciones que componen el sistema operativo. Estas bibliotecas permiten la comunicación entre bibliotecas de alto nivel, aplicaciones y la capa de núcleo.

La capa contiene las siguientes características:

- Bibliotecas de la base del lenguaje de programación Java, en el cual están escritas las aplicaciones, permitiendo el uso de objetos propios de dicho lenguaje.
- Máquina virtual Dalvik, la cual se instancia por cada proceso en el sistema. Permite que cada aplicación utilice una máquina virtual propia, siendo ejecutadas por medio de archivos *dalvik executable* (.dex)

4.3.1.4. Marco de trabajo de aplicaciones

Esta capa provee APIs para el manejo de multimedia, redes, telefonía, localización, interfaz gráfica, actividades, contenido, preferencias compartidas, paquetes, notificaciones y recursos compartidos. Provee administradores que evitan una interacción directa y poco eficiente a las bibliotecas internas, optimizando el uso de características de hardware y del sistema por parte de aplicaciones cliente.

Las APIs del marco de trabajo están organizadas por niveles, los cuales avanzan junto con las versiones del sistema operativo. De ello se hablará más adelante.

4.3.1.5. Aplicaciones

Está compuesta por todas las aplicaciones que interactúan con el usuario, incluidas entre ellas la aplicación de teléfono, calendarios, juegos, lanzadores, menús, entre otras.

4.3.2. Android APIs

Android provee una gran cantidad de APIs (Application Programming Interface), las cuales son accesibles desde la capa del marco de trabajo de aplicaciones. Cualquier aplicación dispone de las clases necesarias para su funcionamiento desde el marco de trabajo. Sin embargo, cambian con el pasar del tiempo, por lo tanto, para mantener la historia de las funciones y compatibilidad con diversos dispositivos, Android provee niveles de API, los cuales son lanzados con cada *release*, sin importar si son mayores o menores.

Con el pasar del tiempo, los niveles de API van incrementando, lo que implica que los antiguos niveles quedan obsoletos y eventualmente son borrados.

4.3.3. Android SDK

Google provee a los desarrolladores de aplicaciones Android un kit de desarrollo de software nativo para su sistema operativo. El Android SDK cuenta con las herramientas necesarias para la creación de aplicaciones, tales como manejadores de máquinas virtuales, depuradores de aplicaciones, creadores de archivos binarios (.dex), empaquetadores de aplicaciones (apk) y un manejador de APIs, permitiendo seleccionar las herramientas y bibliotecas a utilizar de las versiones de Android para las que se enfoque una aplicación.

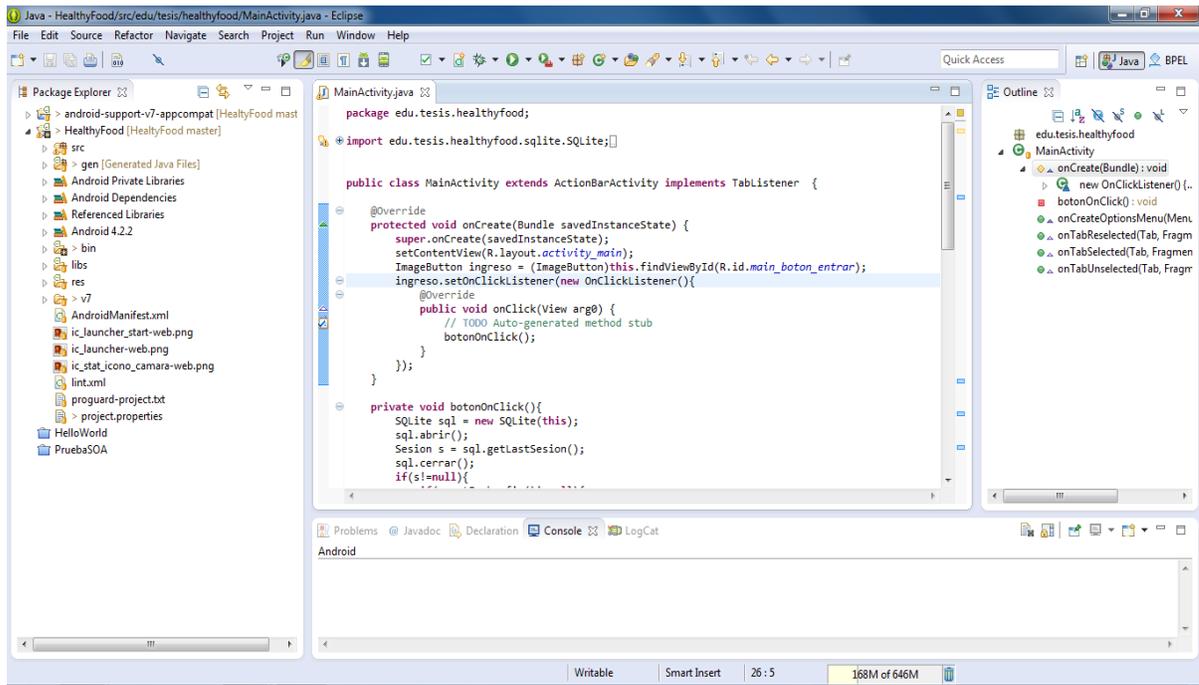
Android SDK es accesible de forma independiente de cualquier entorno de desarrollo integrado. Sin embargo, existen algunos que facilitan su uso, permitiendo simplificar los procesos complejos para compilación, depuración, ejecución y liberación de aplicaciones.

4.4. Entorno de desarrollo Eclipse

Para el desarrollo de la aplicación *HealthyFood* se ha utilizado el entorno de desarrollo integrado Eclipse, el cual posee la característica de ser un sistema extensible a través de la instalación de *plugins*. Además, todos los proyectos y trabajos de desarrollo incluidos están almacenados en un espacio de trabajo o *workspace*. Es altamente personalizable y adaptable a las necesidades del desarrollador. Su desarrollo se lleva a cabo bajo una licencia de uso público de Eclipse, mayoritariamente está constituido por componentes escritos en Java.

Debido a la diversidad de *plugins* que Eclipse maneja es posible desarrollar proyectos en diferentes lenguajes de programación. Algunos pueden ser Ada, ABAP, C/C++, COBOL, Fortran, Haskell, Perl, PHP, Prolog, Python, Javascript, R, Ruby, entre otros, siendo Java el lenguaje principal para el desarrollo de proyectos.

Figura 37. Entorno de desarrollo Eclipse



Fuente: elaboración propia, con programa Microsoft Paint.

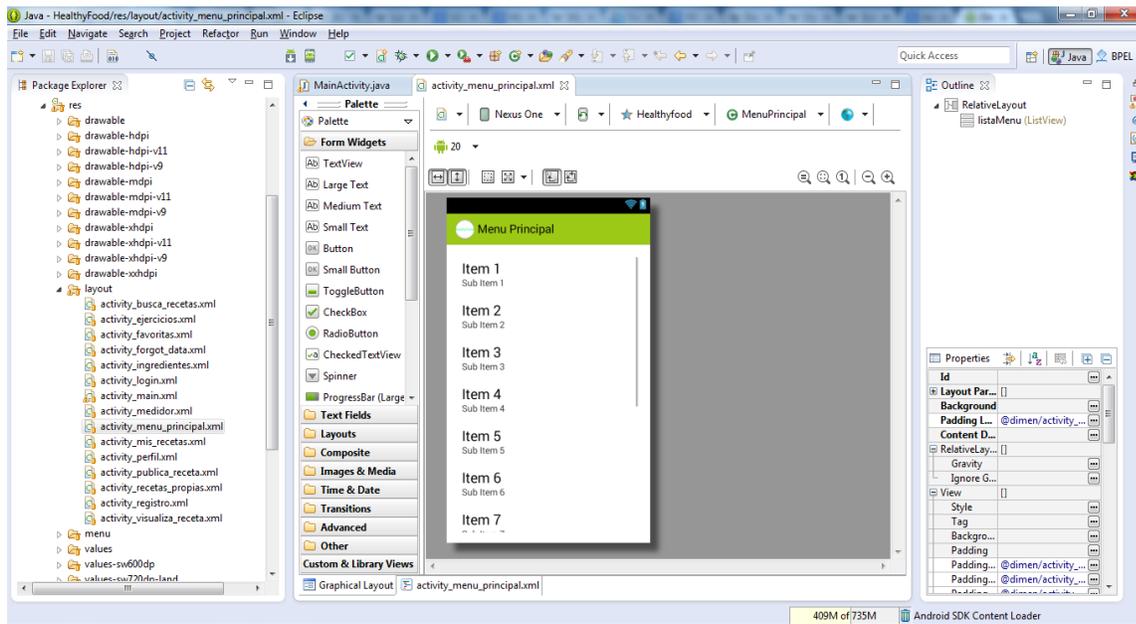
4.4.1. Android Developer Tools (ADT)

Es un *plugin* para el entorno de desarrollo Eclipse que provee un entorno de desarrollo profesional para construir aplicaciones orientadas para el sistema operativo Android. Al integrarlo con Eclipse, provee un entorno completamente para el lenguaje Java que permitirá la construcción, edición, depuración y realización de pruebas de aplicaciones móviles de Android.

Posee un entorno gráfico para la construcción de interfaces de usuario gráficas con la característica de *Drag and Drop*. Cuenta, además del editor de Java, con un editor de XML que permitirá editar los documentos pertenecientes

a un proyecto de Android. Permite la ejecución de las aplicaciones en diversos dispositivos, así como también en dispositivos virtuales, además de soportar compilaciones en C/C++.

Figura 38. Editor de interfaz gráfica del ADT



Fuente: elaboración propia, con programa Microsoft Paint.

4.5. Entorno de desarrollo Android Studio

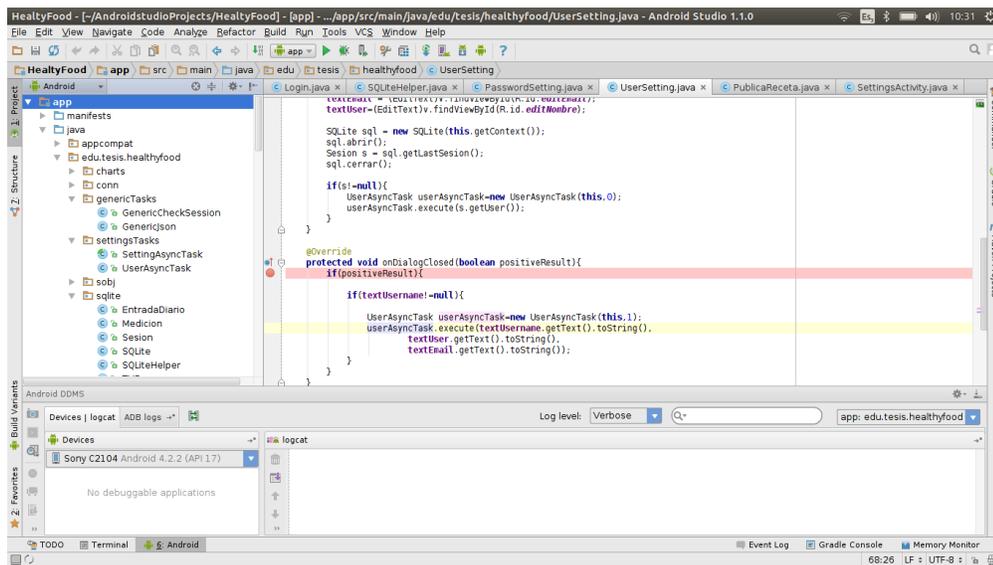
Debido a que Google decidió cambiar de entorno de desarrollo oficial para Android, se finalizó la etapa incremental de desarrollo de la aplicación en Android Studio, el cual está basado en el entorno empresarial de desarrollo IntelliJ IDEA de JetBrains.

Android Studio es un entorno de desarrollo especializado únicamente en la creación, depuración y construcción de aplicaciones para el sistema operativo

móvil de Google, Android. La construcción de las mismas son realizadas utilizando el constructor de binarios Gradle.

Cada proyecto dentro de Android Studio utiliza archivos con *build scripts* que permiten su construcción, resolviendo dependencias para la compilación, ya sea de bibliotecas cuyo fichero binario se incluye en el proyecto, módulos internos de proyectos externos de Android Studio o bibliotecas encontradas en repositorios externos (en este caso Gradle procede a descargarlas al momento de compilar).

Figura 39. Entorno de desarrollo Android Studio



Fuente: elaboración propia, con programa de captura de pantalla de Ubuntu.

4.6. Sistema de versiones GIT

Es un controlador de versiones que se usa para almacenar la historia del desarrollo de un proyecto a través del tiempo. Se utiliza a través de comandos

que sirven para actualizar el proyecto, copiarlo o crear una nueva rama de desarrollo independiente de la principal.

GIT también es utilizado para evitar que exista un conflicto entre las distintas versiones de trabajo entre varias personas. Este lleva el control de los cambios realizados desde el último *commit* o confirmación de guardado y puede unir a cualquier documento solo los cambios última actualización sin necesidad de reemplazarlo todo. El cambio se hace solamente a los documentos que hayan sido indicados.

GIT es gratuito y de código libre por lo que no hace falta instalar algo más o pagar por usar un servicio. Existen varios sitios en la nube donde se guardan los repositorios de GIT y que estos se encuentren disponibles en cualquier momento y lugar a través de internet.

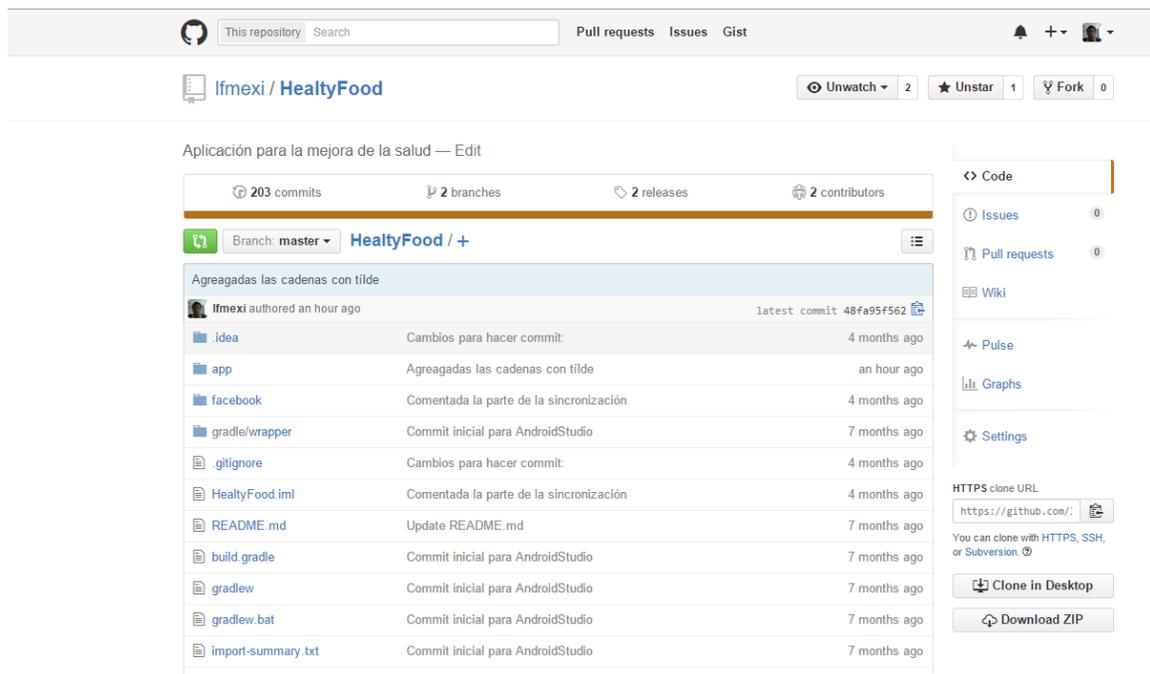
4.7. Repositorio centralizado en el sitio GitHub

Es un sitio web que ofrece almacenamiento de código fuente en la nube a través de repositorios que utilizan el sistema de control de versiones Git. Los servicios básicos de GitHub son gratuitos para cualquier usuario registrado en su sitio. Sin embargo, existen servicios y funcionalidades reservadas para usuarios de paga, como lo es la creación de repositorios privados.

GitHub posee conectividad con diversos orígenes de código fuente, permitiendo a los usuarios clonar repositorios, crear *forks*, realizar modificaciones y mantener los mismos actualizados con base en un gestor de versiones Git, sin importar el lenguaje de programación, entorno de desarrollo o sistema operativo.

El código fuente de la aplicación se encuentra en un repositorio público de GitHub, permitiendo la colaboración en el desarrollo de la misma.

Figura 40. Repositorio público en GitHub desde un navegador web



Fuente: elaboración propia, con programa de captura de pantalla de Windows 10.

4.8. Bibliotecas de terceros utilizadas

El código fuente de la aplicación depende de la utilización de bibliotecas externas que no están disponibles en kit de desarrollo de Android. A continuación se describen las bibliotecas utilizadas, tanto para mostrar datos de forma gráfica como para mostrar videos.

4.8.1. MPAndroidChart

Es una biblioteca para la utilización de gráficos para Android, brinda gráficos de barras, de línea, en forma de pie, histogramas, entre otras. Es soportada por el sistema operativo Android a partir de la versión 2,2. Es una biblioteca de código abierto, con una licencia Apache, Versión 2,0. El código fuente y la biblioteca en sí, junto con una aplicación de ejemplo y su respectivo código, se encuentran en un repositorio de GitHub (<https://github.com/PhilJay/MPAndroidChart>).

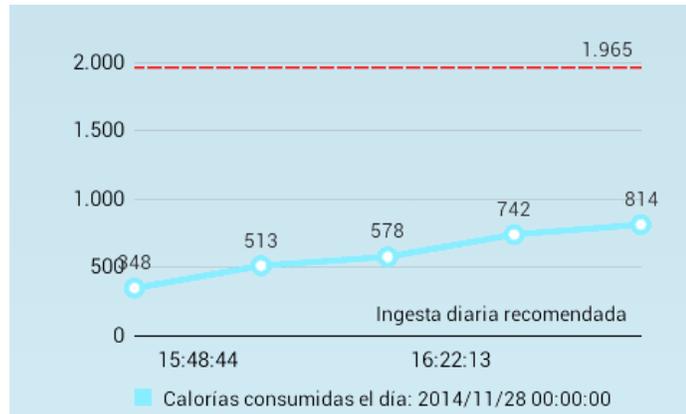
4.8.1.1. Gráficos utilizados

De los gráficos que provee MPAndroidChart se utilizaron 2 tipos en la aplicación. A continuación se muestran algunos ejemplos.

4.8.1.1.1. Gráfico de línea

Fue utilizado en la aplicación para mostrar el progreso del usuario en cuanto al índice de masa corporal del mismo. Además se utilizó para mostrar la cantidad de calorías consumidas en un día.

Figura 41. **Gráfico de línea**

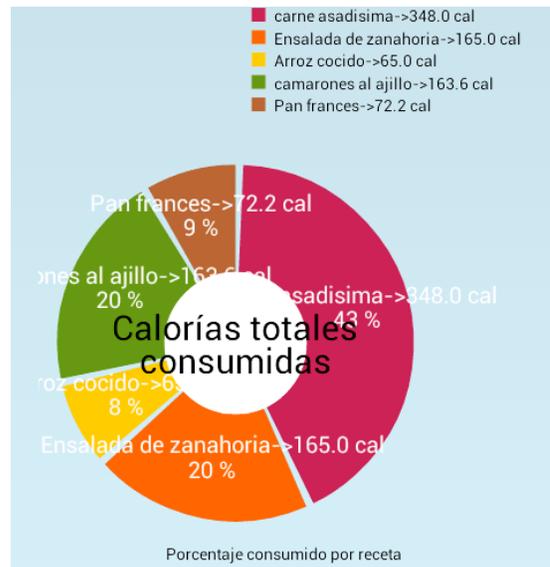


Fuente: elaboración propia, con programa de captura de pantalla de Android.

4.8.1.1.2. **Gráfico de pie**

Es utilizado en la aplicación para mostrar los porcentajes que ocupa cada receta agregada al diario por el usuario de su ingesta diaria de calorías.

Figura 42. Gráfico de pie



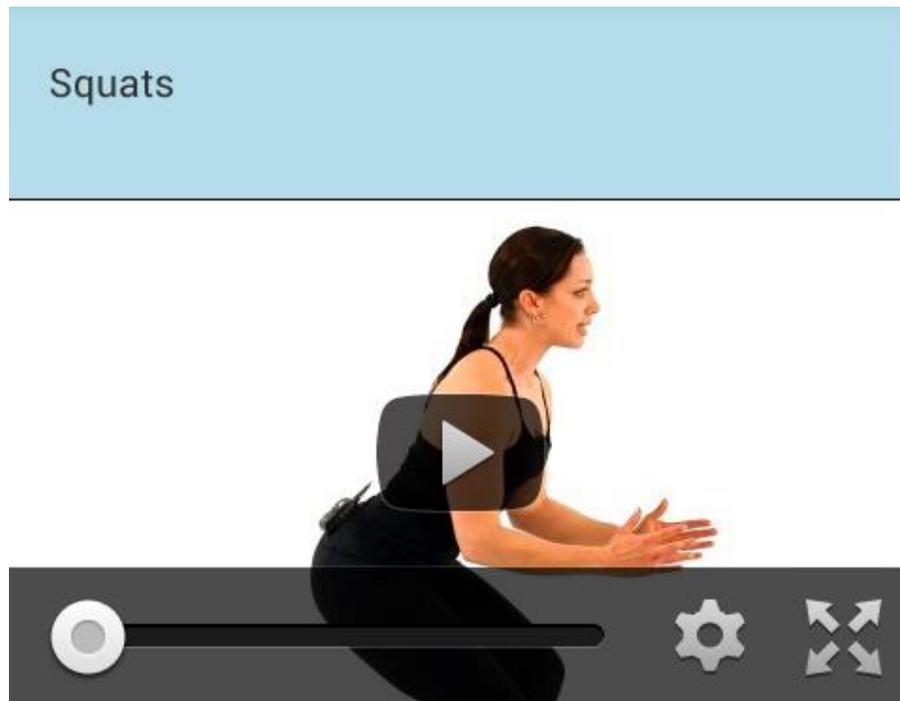
Fuente: elaboración propia, con programa de captura de pantalla de Android.

4.8.2. YouTube Android Player API

Google posee una biblioteca especial para que las aplicaciones Android puedan incluir un reproductor embebido de YouTube que reproduzca videos o listas de reproducción almacenados en dicho servicio. Para utilizar el API de YouTube es necesario registrar la aplicación que la utilizará en la consola de desarrollo de Google. Para ello se necesita contar con una cuenta de Google vinculada a Google Developers, posterior a ello se registra en el sitio <https://console.developers.google.com>.

La utilización del API de YouTube consiste en la importación de un archivo *.jar* al proyecto de Android, el cual contiene los archivos binarios necesarios para la utilización de la biblioteca.

Figura 43. **Reproductor embebido en una aplicación de Android**



Fuente: elaboración propia, con programa de captura de pantalla de Android.

4.8.3. Facebook SDK

Para agregar integración con redes sociales se incluyó la funcionalidad de iniciar sesión en la aplicación con Facebook. Para ello fue necesario incluir el kit de desarrollo de software de Facebook o Facebook SDK.

Facebook SDK permite el desarrollo de aplicaciones con conectividad a la red social Facebook. Este registra sesiones y permite a los desarrolladores crear aplicaciones que basen parte de su funcionamiento en la información obtenida de dicha red social, así como también las preferencias de los usuarios,

con el objetivo de brindar asistencia personalizada o interacciones más acertadas.

4.9. Requisitos de uso de la aplicación

Para el uso de la aplicación se debe contar con un dispositivo móvil con las siguientes indicaciones:

- Sistema operativo: Android.
- Versión de sistema operativo: Mayor a 2,3, preferiblemente 4,2 o superior.
- La aplicación pide permisos para hacer uso de la conexión de internet y el acceso al almacenamiento externo del dispositivo.
- El dispositivo debe contar con cámara.

5. APLICACIÓN DESARROLLADA

5.1. Utilización de la aplicación

La aplicación ha sido probada en un dispositivo móvil que cuenta con los requerimientos técnicos descritos en el capítulo anterior. A continuación se muestra cada uno de los módulos desarrollados.

5.1.1. Registro de usuarios

Dentro del sistema, los usuarios deberán registrarse para utilizar la aplicación. Para ello deberán llenar un formulario de registro, el cual pide datos como el nombre de usuario, correo electrónico, nombre real, sexo, fecha de nacimiento y contraseña para identificarse.

Figura 44. **Registro de usuarios**

The screenshot shows a registration form with the following fields and elements:

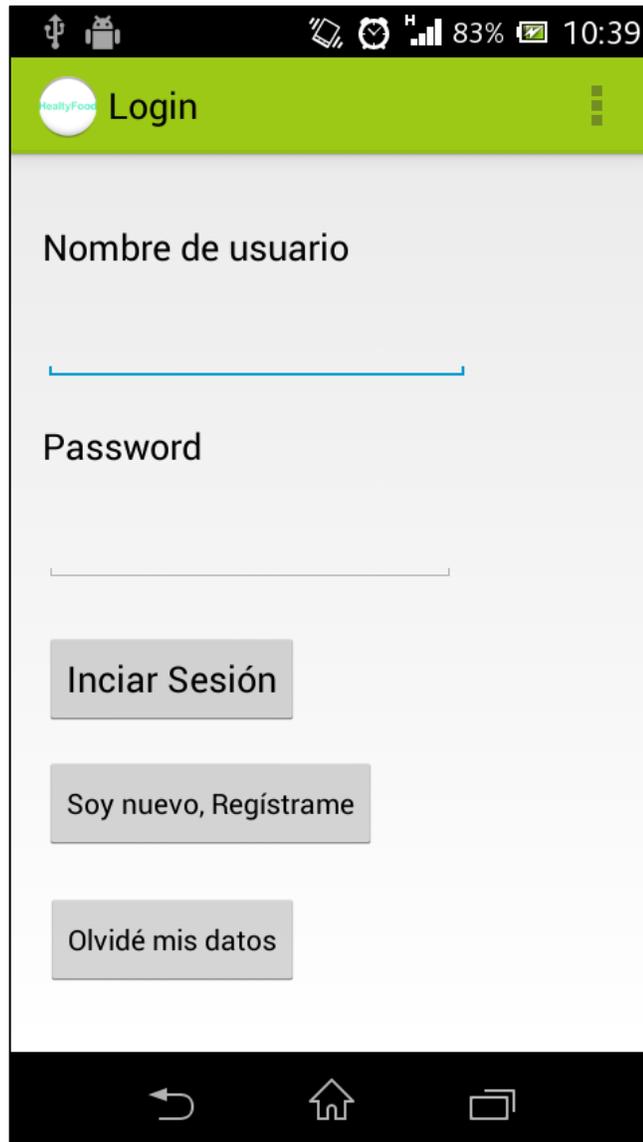
- Header: HealthyFood logo and the word "Registro" in a green bar.
- Username: A text input field.
- e-mail: A text input field.
- Nombre: A text input field.
- Sexo: A dropdown menu currently showing "Hombre".
- Fecha de nacimiento: A date picker showing "2014-11-30" with a "Seleccionar Fecha" button.
- Password: A text input field.
- Confirma Password: A text input field.

Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.2. Autenticación de usuarios

Para ingresar al sistema con un usuario ya existente, se deberá acceder a través de la autenticación de usuarios. El formulario solicita el nombre de usuario único y la contraseña.

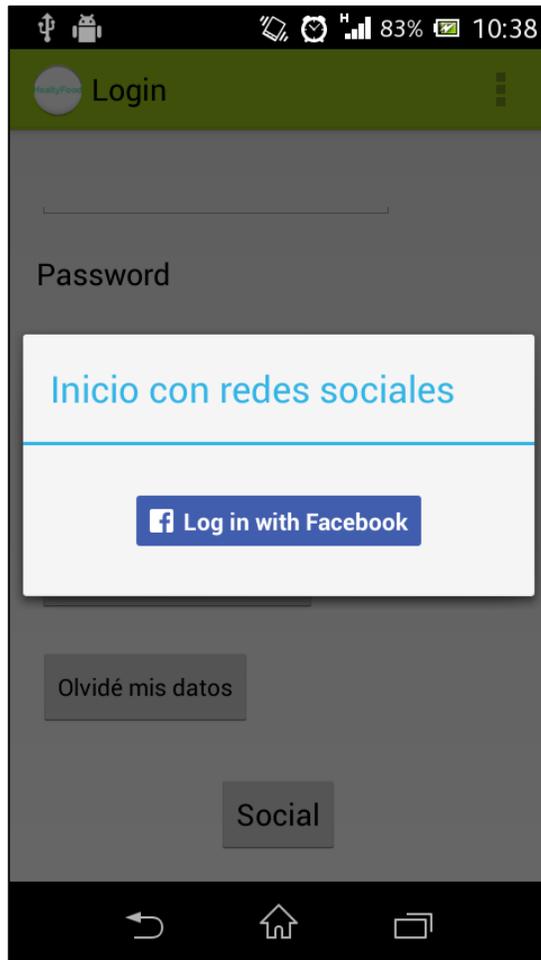
Figura 45. **Autenticación**



Fuente: elaboración propia, con programa de captura de pantalla de Android.

Además de la autenticación estándar, los usuarios pueden acceder por medio de su cuenta de Facebook. Tal a como se muestra en las imágenes a continuación.

Figura 46. Inicio de sesión por redes sociales



Fuente: elaboración propia, con programa de captura de pantalla de Android.

Figura 47. **Pantalla de identificación de aplicación**



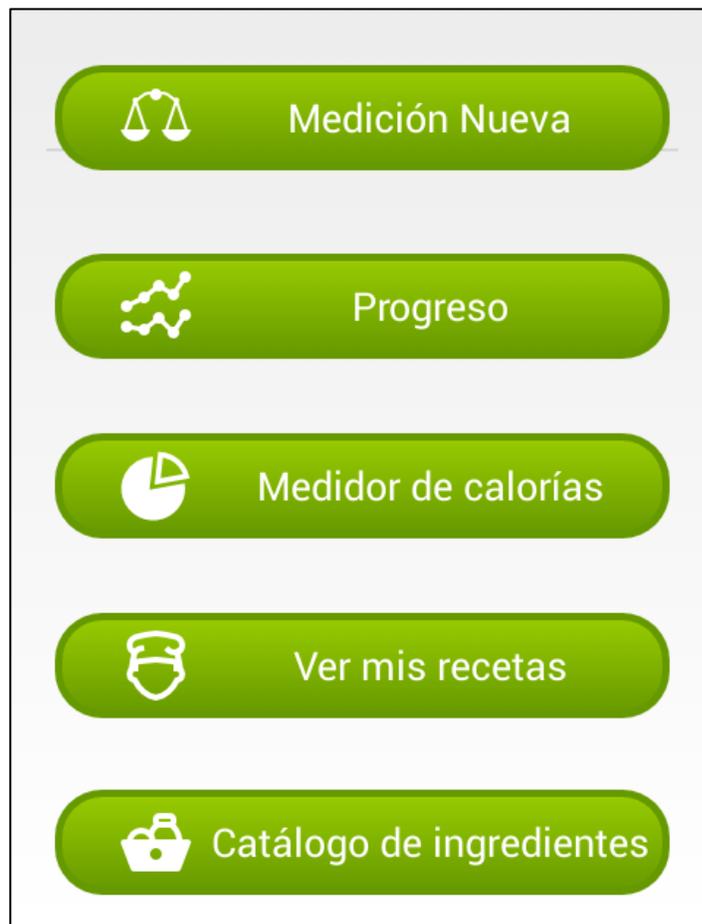
Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.3. Pantalla principal

La aplicación cuenta con una página principal, desde la cual los usuarios podrán realizar diferentes acciones. Por ejemplo: realizar mediciones nuevas, consultar su progreso, consultar su diario de consumo de calorías, ver las

recetas publicadas por ellos mismos y ver los ingredientes disponibles actualmente en el sistema.

Figura 48. **Pantalla principal**



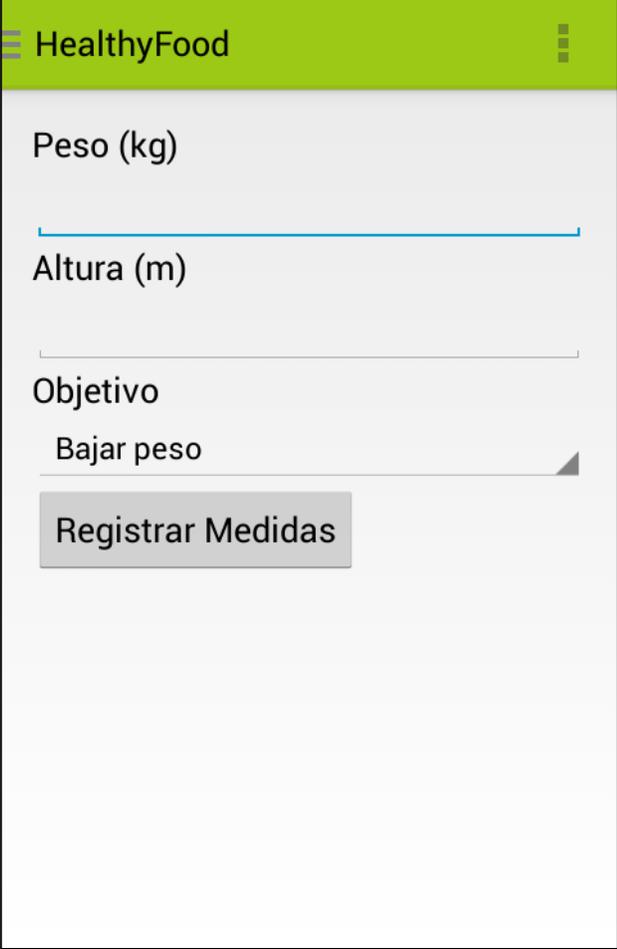
Fuente: elaboración propia, con programa de captura de pantalla de Android

A continuación se describe cada una de las acciones que los usuarios pueden realizar.

5.1.3.1. Medición nueva

A partir de esta opción, los usuarios podrán ingresar los datos de una nueva medición física de su estado. Pide medidas como el peso y la altura, además de registrar el objetivo que quieren lograr, el cual puede ser: bajar, mantener o ganar peso.

Figura 49. Medición nueva



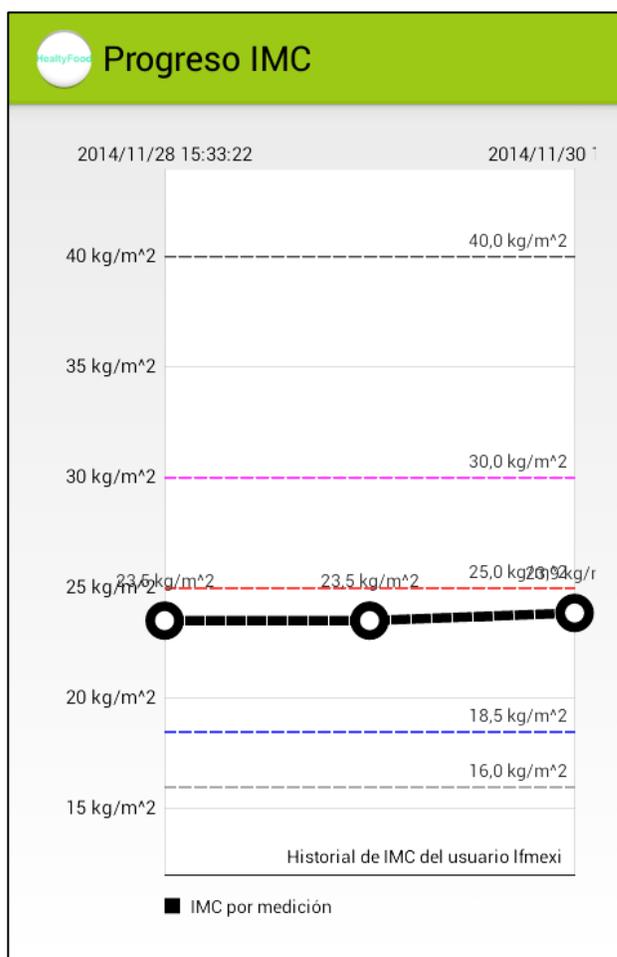
The screenshot shows a mobile application interface for recording a new measurement. At the top, there is a green header bar with the text 'HealthyFood' and a menu icon on the left, and a vertical ellipsis on the right. Below the header, the form consists of three input fields: 'Peso (kg)' with a blue underline, 'Altura (m)' with a grey underline, and 'Objetivo' with a dropdown menu currently showing 'Bajar peso'. At the bottom of the form is a grey button with the text 'Registrar Medidas'.

Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.3.2. Progreso

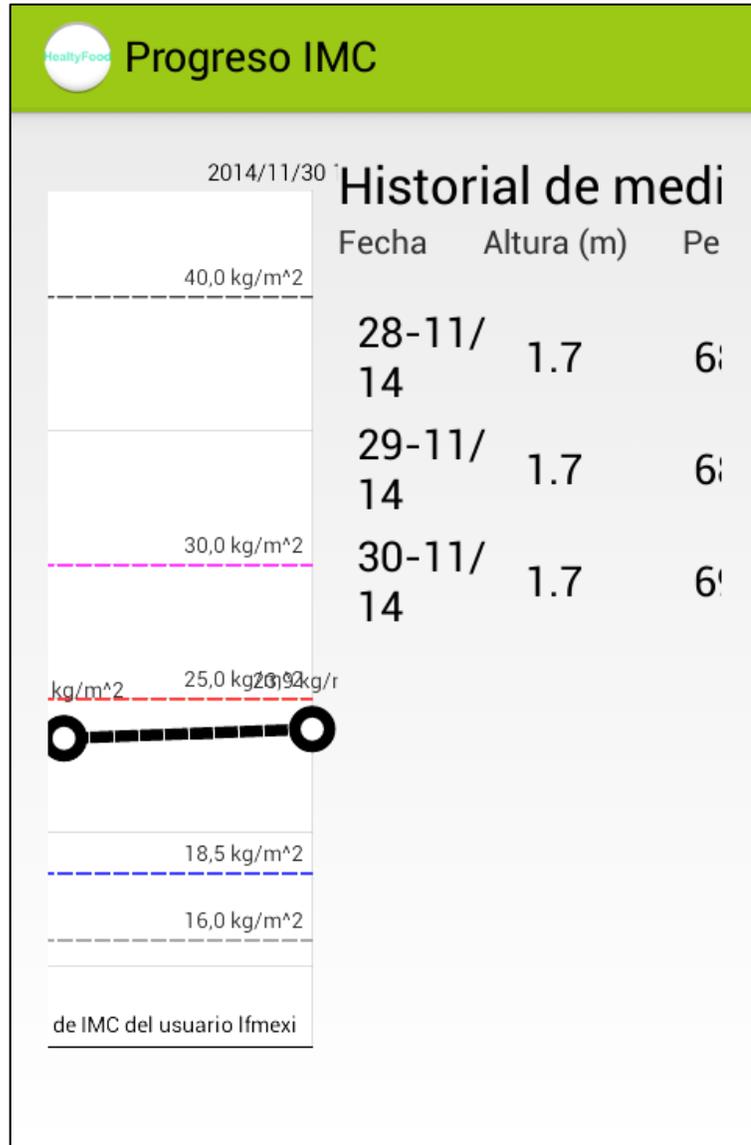
Permite a los usuarios verificar, de manera histórica, su progreso con respecto a las medidas ingresadas. Toma en cuenta la altura, peso e índice de masa corporal. Muestra dos pantallas, en una muestra un gráfico de línea con los distintos valores del índice de masa corporal del usuario y en otra muestra una tabla con las distintas mediciones realizadas.

Figura 50. Gráfico del progreso del usuario



Fuente: elaboración propia, con programa de captura de pantalla de Android.

Figura 51. Transición entre pantallas de progreso



Fuente: elaboración propia, con programa de captura de pantalla de Android.

Figura 52. **Matriz del historial de mediciones del usuario**



The image shows a screenshot of a mobile application interface. At the top, there is a green header bar with a white circular logo containing the text 'HealthyFood' and the title 'Progreso IMC'. Below the header, the main content area has a light gray background and is titled 'Historial de mediciones'. It contains a table with four columns: 'Fecha', 'Altura (m)', 'Peso (kg)', and 'IMC (kg/m^2)'. The table lists three rows of data representing measurements taken on different dates in November 2014.

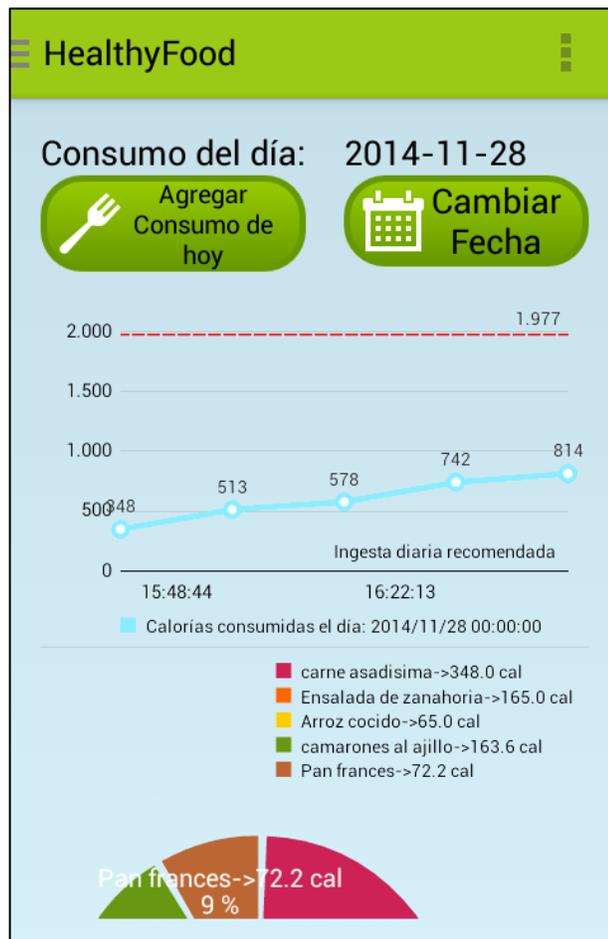
Fecha	Altura (m)	Peso (kg)	IMC (kg/m ²)
28-11/ 14	1.7	68.0	23.53
29-11/ 14	1.7	68.0	23.53
30-11/ 14	1.7	69.0	23.88

Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.3.3. Medidor de calorías

Al utilizar esta opción, los usuarios podrán ver de forma gráfica el consumo de calorías de un determinado día, por defecto muestra las calorías consumidas en el día actual. Sin embargo, permite la consulta de días anteriores. Además de ello, los usuarios podrán agregar recetas al consumo del día.

Figura 53. Medidor de calorías



Fuente: elaboración propia, con programa de captura de pantalla de Android.

Al presionar el botón de agregar consumo se mostrará el buscador de recetas, en el que los usuarios podrán seleccionar cualquier receta que deseen visualizar y consumir.

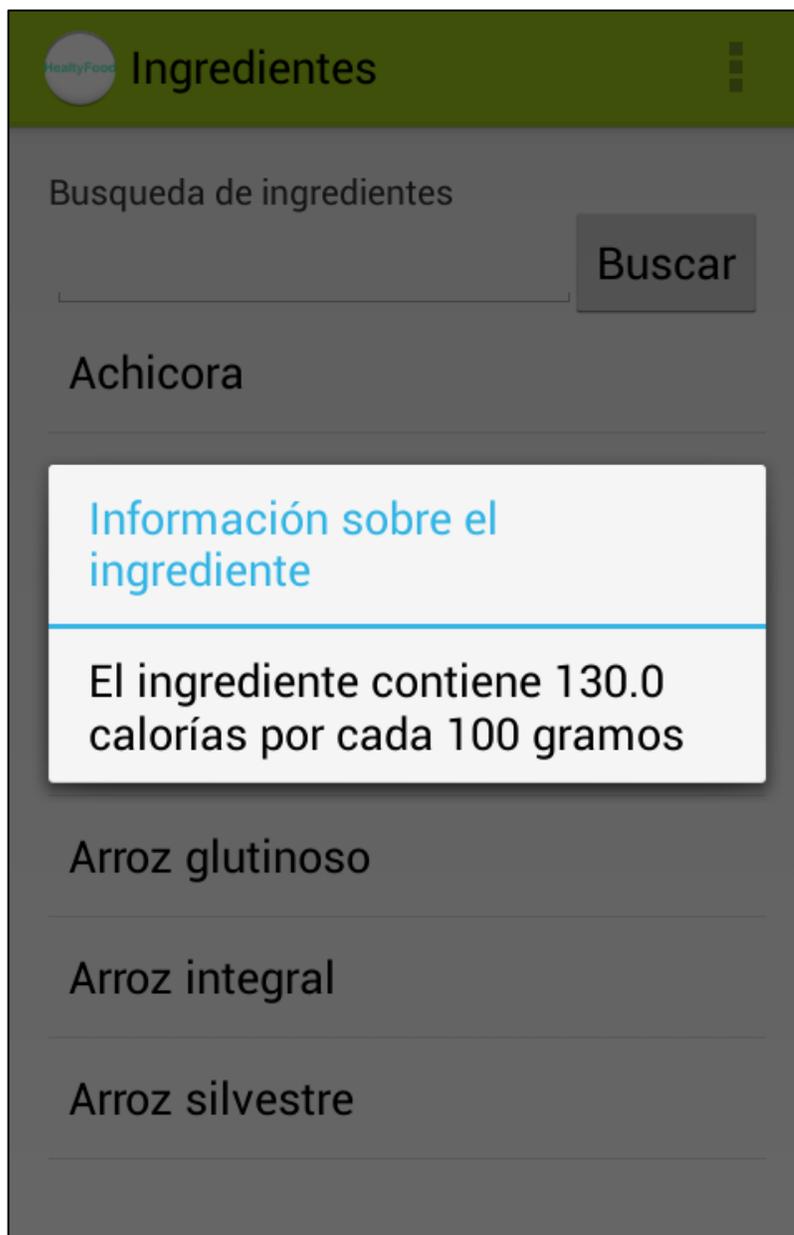
5.1.3.4. Ver mis recetas

Muestra un listado de recetas ingresadas por el usuario, desde ese listado es posible seleccionar una receta, visualizar su contenido y agregarla al consumo diario.

5.1.3.5. Catálogo de ingredientes

Muestra un buscador de ingredientes, en el cual los usuarios podrán consultar la cantidad de calorías que poseen. La búsqueda la realiza con todos los ingredientes almacenados en el sistema.

Figura 54. Catálogo de ingredientes

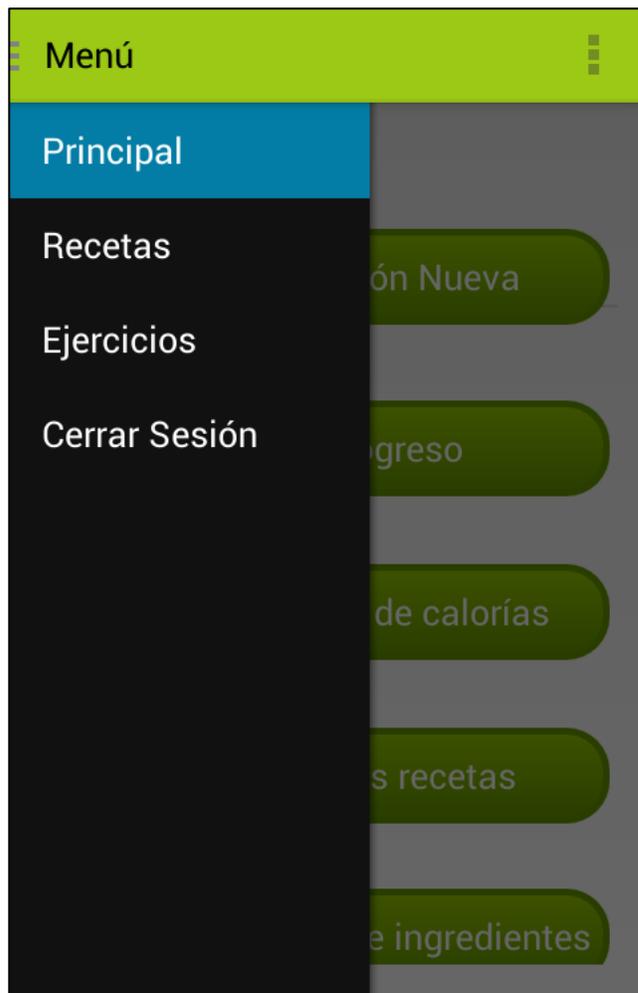


Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.4. Menú lateral

La aplicación cuenta con un menú lateral, el cual posee un listado de funciones disponibles dentro de la aplicación. Cada opción del listado lleva a una pantalla distinta. Dentro de ellas se encuentran: la pantalla principal, la de recetas, la de ejercicios y la opción de cerrar sesión.

Figura 55. Menú lateral



Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.5. Recetas

La aplicación cuenta con una pantalla especial para las recetas. En esta se muestra un menú en el cual los usuarios podrán seleccionar una acción relacionada con las recetas. Dentro de estas acciones se encuentran: publicar una nueva receta, buscar recetas dentro del sistema, visualizar las recetas propias y visualizar las recetas favoritas.

Figura 56. **Pantalla de recetas**



Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.5.1. Publicación de recetas

Los usuarios podrán publicar sus recetas en el sistema, lo que permitirá a otros usuarios utilizarlas y registrarlas en su diario de consumo. Desde la pantalla mostrada para la publicación, se agregan ingredientes a una nueva receta, visualizar los ingredientes ingresados, las calorías de la receta, se agregan instrucciones y una foto de la receta.

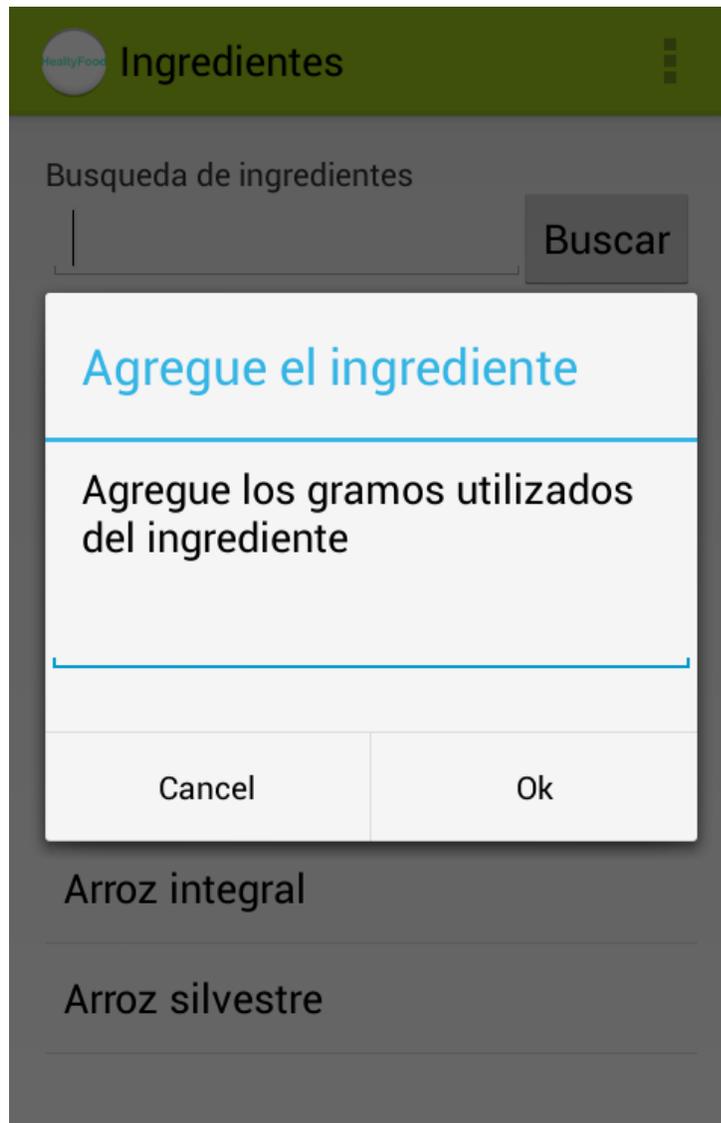
Además de ello, al presionar el botón para agregar ingredientes se mostrará una pantalla similar a la del catálogo de ingredientes, con la diferencia que al seleccionar un ingrediente, la aplicación pedirá que se ingrese la cantidad de gramos a utilizar del mismo. Si un ingrediente ya está ingresado en la receta, la aplicación permitirá al usuario eliminarlo.

Figura 57. Pantalla de publicación de recetas



Fuente: elaboración propia, con programa de captura de pantalla de Android.

Figura 58. **Pantalla para agregar ingredientes**

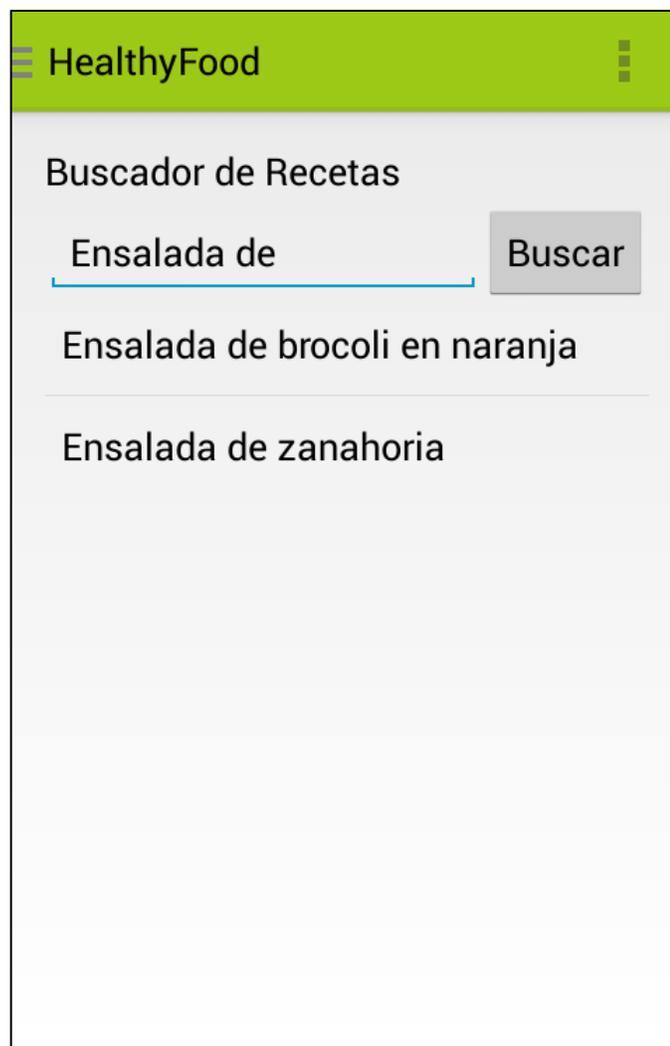


Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.5.2. Buscar recetas

Permite a los usuarios buscar recetas registradas dentro del sistema, filtrando únicamente por un patrón de búsqueda basado en el nombre de las recetas. Mostrará cualquier receta publicada por cualquier usuario.

Figura 59. Pantalla de búsqueda de recetas

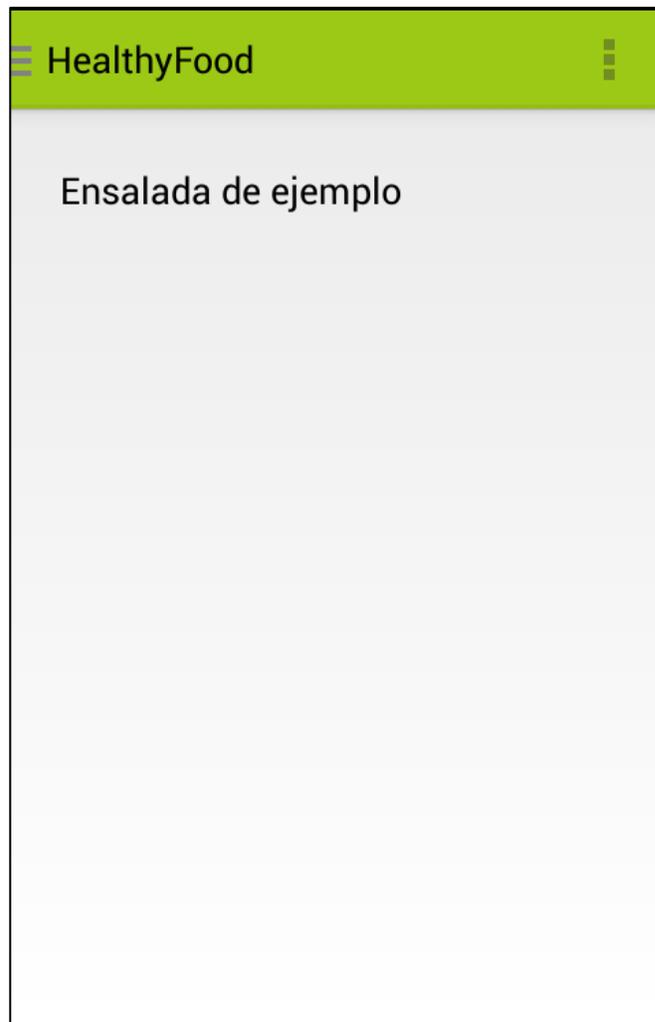


Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.5.3. Mis recetas

Al igual que en la opción de ver mis recetas en la pantalla principal, la opción de “mis recetas” muestra las recetas publicadas por el usuario únicamente.

Figura 60. **Pantalla de recetas propias**

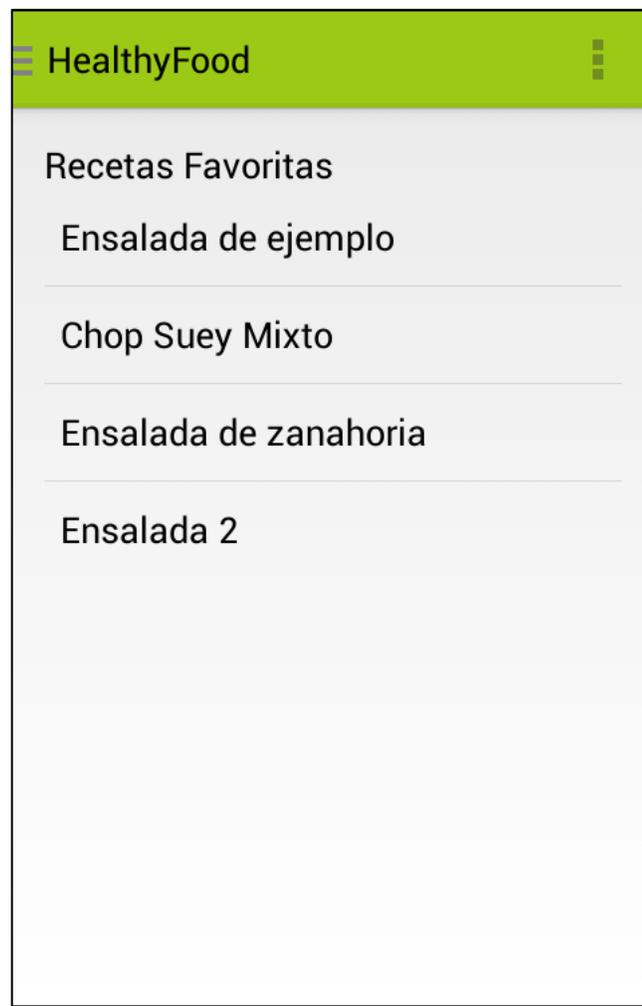


Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.5.4. Recetas favoritas

Muestra la lista de recetas que el usuario de la aplicación ha marcado como favoritas, sin importar qué usuario la haya publicado.

Figura 61. Listado de recetas favoritas



Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.5.5. Visualización de recetas

Los usuarios podrán visualizar las recetas que deseen, sin importar que hayan sido buscadas desde la opción de buscar recetas o seleccionadas desde las recetas propias del usuario y las recetas favoritas. La aplicación muestra una pantalla en la que los usuarios podrán ver la cantidad de calorías que proporciona la receta. Además, podrán ver el listado de ingredientes, las instrucciones de la receta, su tipo, la opción de marcarla como favorita, dar una puntuación a esta, y agregarla al diario de consumo.

Figura 62. Visualización de una receta

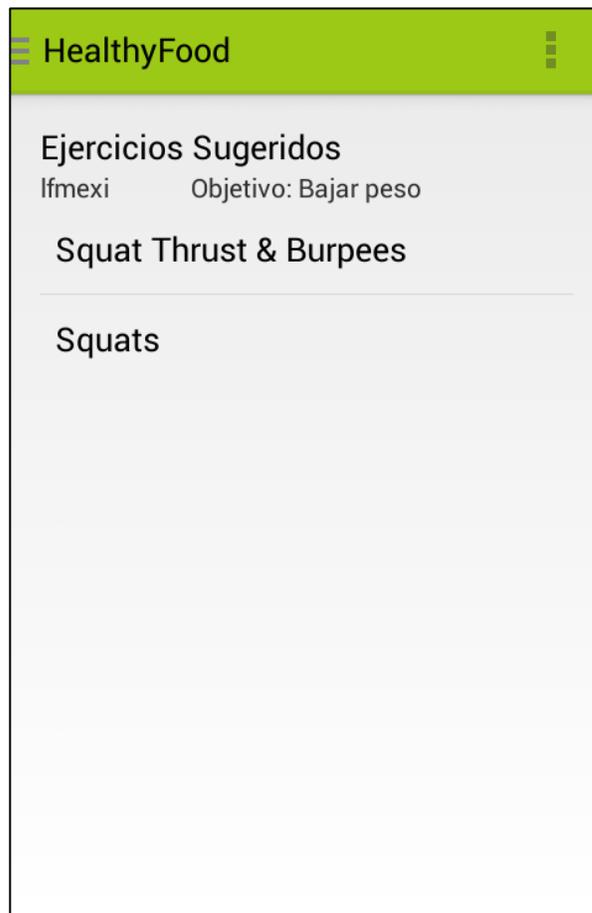


Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.1.6. Ejercicios

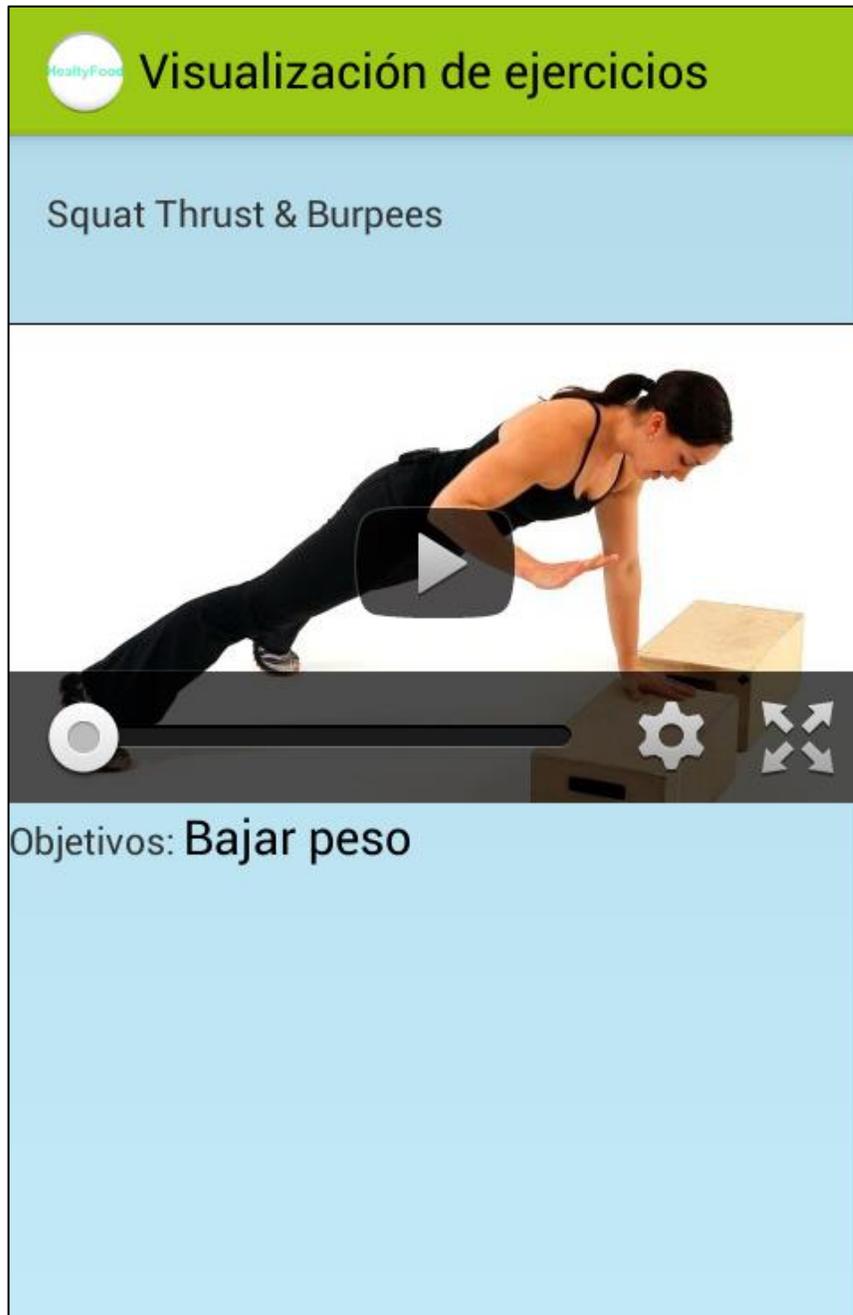
La aplicación es capaz de enlistar una serie de ejercicios recomendados para los usuarios de acuerdo al objetivo seleccionado en la última medición realizada. Cuando el usuario selecciona un ejercicio, podrá visualizar un video que muestra cómo se realiza el mismo con un reproductor de YouTube embebido.

Figura 63. Listado de ejercicios recomendados



Fuente: elaboración propia, con programa de captura de pantalla de Android.

Figura 64. Visualización de ejercicios



Fuente: elaboración propia, con programa de captura de pantalla de Android.

5.2. Archivo ejecutable para la instalación de la aplicación

Los *releases* de la aplicación podrán ser encontrados en el repositorio de GitHub, en la sección de *releases*, siguiendo el enlace: <https://github.com/lfmexi/HealtyFood/releases>.

5.3. Licenciamiento de la aplicación

La aplicación ha sido desarrollada como un proyecto de código abierto, por lo tanto contará con una licencia del mismo tipo.

Debido a que utiliza una biblioteca que ha sido liberada bajo una licencia Apache versión 2,0, la aplicación heredará dicha licencia de código abierto para su código fuente. El API de YouTube no está cubierto por la licencia, por lo que no se podrá utilizar el código fuente en sí de dicha biblioteca (para más información, seguir el enlace de condiciones: <https://developers.google.com/youtube/terms>).

Es posible encontrar el contenido de la licencia Apache versión 2,0 en el siguiente enlace: <http://www.apache.org/licenses/LICENSE-2.0>.

CONCLUSIONES

1. La aplicación desarrollada permite el ingreso de consumo de alimentos para calcular la ingesta de calorías y determinar la ingesta diaria recomendada de las mismas. Además, permite a los usuarios conocer su índice de masa corporal y su estado por medio del mismo.
2. La aplicación desarrollada incluye conectividad con redes sociales. Esto permitirá a los usuarios publicar las recetas que más les interese a sus contactos.
3. El incremento del sedentarismo y de la mala alimentación han generado un nuevo segmento de mercado para los desarrolladores de aplicaciones móviles. La orientación a la salud alimenticia por parte de algunas empresas de desarrollo de software se ha incrementado en los últimos años, permitiéndole al usuario seleccionar la aplicación que mejor se adecue a sus necesidades.
4. El sistema operativo Android es el más usado en los dispositivos móviles, lo cual provee una gran cantidad de usuarios potenciales a la aplicación.
5. La versión del sistema operativo Android es un factor muy importante a tomar en cuenta para el desarrollo del software debido a que no todas las personas tienen la última versión y siempre saldrán nuevas versiones en el futuro.

6. La utilización del modelo de arquitectura de 4+1 vistas durante el desarrollo de software provee un mejor entendimiento sobre la estructura y el comportamiento de los componentes de un sistema.

RECOMENDACIONES

1. Implementar la estructura de servicios y de base de datos externa de la aplicación en una plataforma en la nube con capacidad de escalabilidad, asumiendo costos necesarios para su mantenimiento y alta disponibilidad.
2. Colocar un módulo de publicidad en la aplicación para su auto sostenimiento, debido a que no se plantea cobrar por su uso.
3. Agregar conectividad con más redes sociales, tales como Twitter y Google+.
4. Publicar la aplicación en la tienda de aplicaciones Android de Google. Esto permitirá que sea conocida y le dará el ingreso al mercado de aplicaciones móviles.
5. Evaluar la aceptación de la aplicación en el mercado de aplicaciones móviles a largo plazo, con el objetivo de determinar la continuidad de su desarrollo incremental.
6. Para el desarrollo de aplicaciones se recomienda que los entornos de desarrollo y pruebas sean los mismos si es posible o similares, ya que las diferencias entre las versiones del software llegan a causar inconvenientes en cuanto el desempeño de la aplicación o retrasar el tiempo de desarrollo del proyecto.

7. Se debe analizar los gustos y preferencias del público objetivo al cual está enfocada la aplicación así como las tecnologías similares y ya existentes, para determinar cuáles son las tendencias en cuanto al funcionamiento que ofrecen para enriquecer y mejorar la experiencia de los usuarios al momento del uso de la aplicación.

BIBLIOGRAFÍA

1. BRÄHLER, Stefan. *Analysis of the Android Architecture*. Tesis de la Facultad de Informática. Instituto de Tecnología de Karlsruhe, Alemania 2010. 52 p.
2. CANAHUÍ, Enrique. *Telefonía: Guatemala es un mercado maduro* [en línea]. <<http://www.economista.net/2014/03/12/telefonia-guatemala-es-un-mercado-maduro>>. [Consulta: 4 de noviembre de 2014].
3. CNET, *Android dominates 81 percent of world smartphone market* [en línea]. <<http://www.cnet.com/news/android-dominates-81-percent-of-world-smartphone-market/>>. [Consulta: 4 de noviembre de 2014].
4. DELOITTE. *La tasa de crecimiento de dispositivos tecnológicos se ralentiza* [en línea]. <<http://www2.deloitte.com/es/es/pages/about-deloitte/articles/informe-predicciones-2014.html>>. [Consulta: 4 de noviembre 2014].
5. _____. *Tecnología, Medios y Telecomunicaciones* [en línea]: <http://www.deloitte.com/view/es_co/co/industrias/tmt/index.htm>. [Consulta: 4 de noviembre de 2014].
6. GOOGLE, inc. *Documentación con guías para la correcta implementación de APIs de Android 4.2* [en línea].

<<https://developer.android.com/guide/index.html>>. [Consulta: 10 de enero de 2015].

7. GUERRA, Daniel. *Estudio para la implementación de tecnologías en la nube desde dispositivos móviles para la formación combinada en la Escuela de Ciencias y Sistemas de FIUSAC*. Trabajo de graduación de Ingeniería en Ciencias y Sistemas. Universidad de San Carlos de Guatemala, Facultad de Ingeniería 2013. 96 p.
8. MUCHANDI, Veer. *Applying 4+1 View Architecture with UML 2 White paper*. Estados Unidos: Spyrax Systems, 2007. 11 p.
9. PCMAG. *Definition of Tablet computer* [en línea]. <<http://www.pcmag.com/encyclopedia/term/52520/tablet-computer>>. [Consulta: 5 de noviembre de 2014].
10. PÉREZ, Luis. *Análisis de plataformas populares de desarrollo de aplicaciones para dispositivos móviles*. Trabajo de graduación de Ingeniería en Ciencias y Sistemas. Universidad de San Carlos de Guatemala, Facultad de Ingeniería 2011. 100 p.
11. SIT. *Crecimiento de la telefonía fija y móvil en Guatemala* [en línea]: <http://www.sit.gob.gt/files/8913/9636/9391/crecimiento_de_la_telefonia_fija_y_movil_2do_2013.pdf>. [Consulta: 4 de noviembre de 2014].
12. WINWOOD, Anne. *Acelerar el metabolismo para quemar grasa: el programa revolucionario de calorías negativas*. 3a ed. Argentina: Lea Libros 2008. 80 p.

APÉNDICES

Encuesta

1. Indique la carrera que estudia
 - a. Ingeniería en Sistemas
 - b. Ingeniería Industrial
 - c. Ingeniería Civil
 - d. Ingeniería Mecánica
 - e. Ingeniería Química
 - f. Ingeniería Electrónica
 - g. Ingeniería Eléctrica
 - h. Ingeniería Mecánica Eléctrica
 - i. Ingeniería Mecánica Industrial
 - j. Otra
 - k. Indique su edad
2. Indique su edad
3. Indique su género
4. Posee un dispositivo móvil
 - a. Sí
 - b. No
5. Indique qué sistema operativo utilizan estos dispositivos
 - a. Android
 - b. iOS
 - c. Windows Phone

- d. Blackberry
 - e. Otro
6. En caso de tener Sistema operativo Android, indique la versión cuál es la versión del sistema operativo
- a. Menor a 4.0
 - b. 4.0
 - c. 4.1
 - d. 4.2
 - e. 4.3
 - f. 4.4
 - g. No sé
7. ¿Posee plan de datos en su dispositivo?
- a. Sí
 - b. No
8. Ha instalado alguna aplicación en su dispositivo
- a. Sí
 - b. No
9. Qué tipo de aplicaciones utiliza en su dispositivo
- a. Juegos
 - b. Social
 - c. Multimedia
 - d. Científica
 - e. Informativa
 - f. Otra
10. Liste algunas de sus aplicaciones favoritas
11. ¿Conoce usted la cantidad de grasas y calorías que contienen los alimentos que consume diariamente?
- a. Sí
 - b. No

- i. Gimnasia
- j. Natación
- k. Fútbol
- l. Otra

22. ¿Utiliza aplicaciones para la asistencia de actividades físicas?

- a. Sí
- b. No

23. ¿Utiliza aplicaciones para la asistencia de planes alimenticios?

- a. Sí
- b. No

24. ¿Le interesaría utilizar una aplicación que combine ambos aspectos en una sola?

- a. Sí
- b. No

25. ¿Es de su agrado compartir contenidos relacionados a la salud y la actividad física en redes sociales?

- a. Sí
- b. No

Fuente: elaboración propia