

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**  
**FACULTAD DE INGENIERÍA**  
**ESCUELA DE SISTEMAS**  
**INGENIERÍA EN SISTEMAS Y COMPUTACIÓN**

**DISERTACIÓN PREVIA A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO EN SISTEMAS Y COMPUTACIÓN**



**DESARROLLO DE UN APLICATIVO MÓVIL PARA LA  
GESTIÓN FINANCIERA PERSONAL**

**AUTOR: VÁSQUEZ VALLEJO ANDRÉS JOSÉ**

**DIRECTOR: MIGUEL D. ORTIZ NAVARRETE ING. MTR.**

**QUITO, JUNIO 2020**

## **Agradecimiento**

Vivo agradecido con mi familia por creer en mí y apoyarme en cada meta nueva. Ustedes son un soporte indispensable para mi vida.

Un agradecimiento a todos aquellos profesores que tienen la vocación de serlo, que les apasiona su trabajo y que se esfuerzan por sembrar verdadero conocimiento en sus estudiantes.

Agradezco también a la Pontificia Universidad Católica del Ecuador y al Instituto de Fomento al Talento Humano por sus programas de becas que vuelven realidad oportunidades de estudio como esta.

## **Dedicatoria**

A mi Madre, mi eterna Profesora favorita.

## Contenido:

Agradecimiento .....	ii
Dedicatoria .....	iii
Índice de figuras.....	v
Índice de tablas.....	vii
1. Introducción.....	1
1.1. Planteamiento del problema .....	1
1.2. Objetivos .....	1
1.2.1. Objetivo general .....	1
1.2.2. Objetivos específicos.....	2
1.3. Justificación .....	2
1.4. Alcance.....	2
2. Metodología.....	3
2.1. Metodología de investigación .....	3
2.2. Metodología informática .....	3
3. Marco de referencia .....	5
3.1. Marco teórico .....	5
3.1.1. Gestión financiera personal.....	5
3.1.2. Desarrollo de aplicativos móviles .....	6
3.2. Marco conceptual .....	8
3.3. Marco espacial .....	9
4. Desarrollo .....	10
4.1. Análisis exploratorio .....	10
4.2. Sistematización de la gestión financiera personal .....	10
4.3. Desarrollo del aplicativo .....	12
4.3.1. Planificación.....	12
4.3.2. Diseño .....	13
4.3.3. Codificación .....	26
4.3.4. Compilación.....	27
4.3.5. Pruebas.....	28
4.3.6. Postmortem.....	39
5. Conclusiones y recomendaciones.....	43
5.1. Conclusiones .....	43
5.2. Recomendaciones .....	44
Bibliografía .....	45

## Índice de figuras

Figura 1. Diagrama de casos de uso.....	13
Figura 2. Gestión de datos personales.....	14
Figura 3. Diagrama de secuencia F1.1.....	14
Figura 4. Diagrama de secuencia F1.2.....	15
Figura 5. Diagrama de secuencia F1.3.....	15
Figura 6. Gestión de registros.....	16
Figura 7. Diagrama de secuencia F2.1.....	16
Figura 8. Diagrama de secuencia F2.2.1.....	17
Figura 9. Diagrama de secuencia F2.2.2.....	18
Figura 10. Diagrama de secuencia F2.3.....	19
Figura 11. Diagrama de secuencia F2.4.....	20
Figura 12. Gestión de categorías. ....	20
Figura 13. Diagrama de secuencia F3.1.....	21
Figura 14. Diagrama de secuencia F3.2.....	21
Figura 15. Diagrama de secuencia F3.3.....	22
Figura 16. Diagrama de secuencia F3.4.....	23
Figura 17. Exportación de datos. ....	23
Figura 18. Diagrama de secuencia F4.....	24
Figura 19. Gráfico de gastos. ....	24
Figura 20. Diagrama de secuencia F5.....	24
Figura 21. Comandos de voz.....	25
Figura 22. Diagrama de secuencia F6.....	25
Figura 23. Modelo de la base de datos. ....	26
Figura 24. Pantalla principal.....	28
Figura 25. Menú lateral.....	28
Figura 26. Mensaje de aviso.....	29
Figura 27. Agregar datos personales ....	29
Figura 28. Ver datos personales. ....	30
Figura 29. Editar datos personales.....	30
Figura 30. Agregar registro.....	31
Figura 31. Selección de categoría. ....	31
Figura 32. Insertar fecha. ....	31
Figura 33. Consulta general.....	32
Figura 34. Detalles del registro.....	32
Figura 35. Consulta específica. ....	33
Figura 36. Editar registro.....	34
Figura 37. Registro a editar.. ....	34
Figura 38. Eliminar registro. ....	34
Figura 39. Agregar categoría. ....	35
Figura 40. Ver categorías.....	35
Figura 41. Editar categoría.....	36
Figura 42. Categoría a editar.. ....	36
Figura 43. Eliminar categoría.....	37
Figura 44. Exportación de datos. ....	37
Figura 45. Selección de tiempo.....	38
Figura 46. Gráfico de gastos. ....	38

Figura 47. Comandos de voz.....	39
Figura 48. Resumen PSP. ....	40
Figura 49. Tiempo en fase a la fecha.....	40
Figura 50. Defectos inyectados por fase a la fecha. ....	41
Figura 51. Defectos eliminados por fase a la fecha. ....	41

## Índice de tablas

Tabla 1. Agregar datos personales.....	29
Tabla 2. Ver datos personales. ....	29
Tabla 3. Editar datos personales.....	30
Tabla 4. Agregar registro. ....	32
Tabla 5. Consulta general. ....	32
Tabla 6. Consulta específica. ....	33
Tabla 7. Editar registro. ....	33
Tabla 8. Eliminar registro. ....	34
Tabla 9. Agregar categoría.....	35
Tabla 10. Ver categorías. ....	36
Tabla 11. Editar categoría.....	36
Tabla 12. Eliminar categoría. ....	36
Tabla 13. Exportación de datos. ....	37
Tabla 14. Gráfico de gastos. ....	38
Tabla 15. Comandos de voz. ....	39

# 1. Introducción

## 1.1. Planteamiento del problema

Según el Instituto Nacional de Estadísticas y Censos, se considera a una persona pobre si percibe un ingreso familiar per cápita menor a USD 84,82 mensuales y pobre extremo si percibe menos de USD 47,80. Para diciembre de 2019, la pobreza nacional estuvo en 25% y la pobreza extrema en 8,9%. La pobreza va en aumento, ya que al 2018 cerró con un porcentaje de pobreza del 23,2%. (INEC, 2019).

Con las cifras consultadas se puede deducir que uno de cada cuatro ecuatorianos es pobre. Una posible causa puede ser el tener una mala gestión económica. Los habitantes, en general, no poseen el capital de conocimiento suficiente como para poder trazar un adecuado plan de progreso financiero. El no registrar adecuadamente los gastos o ingresos es un requisito fundamental para poder hacer uso adecuado del dinero. El poseer un historial financiero otorga los datos necesarios para poder identificar las categorías que más recurso económico requieren.

En un aspecto diferente, la tecnología ha penetrado a nivel mundial. El uso de un teléfono celular es muy frecuente, más que un lujo, es una necesidad. Las personas son contactadas para trabajo a través de un teléfono. Además, se ha visto un mayor interés en el uso de aplicaciones móviles y no solo por personas dedicadas al mundo tecnológico, sino por el público en general, ya que se visualiza el potencial para poder lucrar con o a través del uso de las mismas.

El INEC (2018) nos indica que el 59% de la población nacional del Ecuador tiene un teléfono celular activado y el mercado actual cuenta con la supremacía para dos sistemas operativos móviles: Android e iOS, siendo el primero el que está presente en la mayor cantidad de dispositivos y con una notable diferencia. Google, en sus presentaciones desde el año 2013, publicó que había 900 millones de dispositivos Android activos. También hizo una proyección que, de continuar dicho crecimiento, para 2014 contaría con 1000 millones de dispositivos activos a finales de ese año. Actualmente, el 84,5% de los teléfonos en todo el mundo cuentan con un sistema operativo Android (Luján Castillo, 2015).

El presente trabajo se centrará en ofrecer una alternativa tecnológica de solución para la mejora de la gestión financiera de los ecuatorianos. Se propone la elaboración de un aplicativo móvil que le permita al usuario registrar sus gastos mediante el uso de su teléfono inteligente. A través de esta acción, se ambiciona poder generar un impacto en los usuarios que tienen Android como sistema operativo en sus dispositivos móviles y facilitarles la gestión de sus recursos económicos.

## 1.2. Objetivos

### 1.2.1. Objetivo general

Desarrollar un aplicativo móvil para la gestión financiera personal que permita al usuario registrar sus gastos personales, colocarles un límite y personalizar categorías, con la finalidad de ayudarlo a administrar su economía.

### 1.2.2. Objetivos específicos

- ❖ Consultar las características, ventajas y desventajas de los aplicativos más populares, mediante una investigación en la tienda oficial de Google, para obtener aspectos relevantes.
- ❖ Establecer las funcionalidades del aplicativo, mediante el análisis de los comentarios de las aplicaciones consultadas en el punto anterior, con el objetivo de realizar un producto útil para las personas.
- ❖ Desarrollar el aplicativo a través del uso del IDE oficial de Google: Android Studio, para que el mismo sea funcional en los dispositivos móviles que cuenten con versiones funcionales del sistema operativo Android.

### 1.3. Justificación

Hurtado (2018) refiere que, dentro de nuestra sociedad ecuatoriana en general, las personas que hacen fortuna depositan en bancos foráneos los recursos producidos por la economía nacional. Esta fuga de capitales provoca que no haya flujo económico suficiente como para que los ecuatorianos aprendan sobre la administración del dinero.

Se puede observar que el recurso económico generado en el Ecuador se va al exterior. Al no haber circulante en el país, hay menos oportunidades de empleo, al no haber empleo no se obtienen ingresos y sin ingresos no es fácil establecer una cultura financiera en nuestra sociedad. Así, cuando un individuo de clase media o baja obtiene empleo, no sabe cómo administrar de forma correcta su nueva fuente de ingreso.

En general, la idiosincrasia ecuatoriana no cuenta con la costumbre de preocuparse por la gestión financiera personal. Se pasa más tiempo buscando fuentes de dinero que aprendiendo sobre su administración. Sin embargo, aquellas personas con ingresos relativamente constantes se dan cuenta de la necesidad de la administración de sus recursos y registran sus gastos cada determinado periodo de tiempo. El problema con ello es que la mente humana es frágil y se pueden pasar por alto varios gastos realizados, y también, ingresos obtenidos.

La implicación práctica de este trabajo ayudará a resolver un problema real para una persona que desee administrar de mejor manera sus finanzas. En función de sus ingresos, el usuario podrá establecer un límite estimado de sus gastos con el objetivo de que los mismos no sobrepasen el ingreso económico del usuario.

### 1.4. Alcance

Primero se investigará sobre la temática del estudio relacionada a la gestión financiera personal y las aplicaciones que se han desarrollado hasta el momento con un objetivo similar. Se tomará nota de las ventajas que poseen y las críticas o comentarios que han dejado sus usuarios.

Seguidamente se establecerán las funcionalidades que deberá cumplir el aplicativo móvil y se elaborará el mismo utilizando Android Studio como herramienta principal. Para este proceso se seguirá la metodología propuesta por Watts Humphrey: *Personal Software Process* (PSP). Finalmente, el trabajo de titulación concluirá con un aplicativo totalmente funcional que permita la gestión financiera personal.

## 2. Metodología

### 2.1. Metodología de investigación

El presente trabajo de titulación utilizará el método deductivo para su desarrollo. Bernal et al. (2014) afirman que este método comienza analizando principios universales, de comprobada validez para darles una aplicación a hechos particulares.

En general, la ingeniería trata de buscar aplicaciones prácticas de la ciencia. Se parte de conceptos ya demostrados para otorgar soluciones enfocadas en satisfacer una determinada necesidad. Ñaupas, Mejía, Novoa y Villagómez (2013) aseguran que el método deductivo es el que se utiliza en la ciencia formal, ya que su criterio de verdad es la demostración.

Como principios universales, se tienen todos los adquiridos durante los estudios de Ingeniería en Sistemas y Computación. Sin embargo, para este trabajo en particular se ha especificado el nivel de pobreza en el país y el escenario actual de la tecnología a nivel mundial. La pobreza alcanza un nivel del 25% y casi el 60% de los ecuatorianos cuentan con un teléfono móvil. Por lo tanto, se puede buscar una solución tecnológica que involucre utilizar el dispositivo personal para ofrecer una herramienta que facilite la administración financiera de las personas.

La técnica de investigación que se utilizará es la observación participante. Esta técnica realiza una recopilación de datos e incluye la intervención directa del investigador en el objeto estudiado (Corbetta, 2007).

Esta técnica va acorde al primer objetivo específico expuesto en este trabajo, donde se menciona que se hará un análisis exhaustivo de la tienda oficial de aplicaciones de Google: "Play Store". Esta observación no se limitará a visualizar las aplicaciones en oferta, se podrá hacer comentarios, preguntas a los desarrolladores y se descargarán las mismas para obtener percepciones propias, de forma que se participe con el objeto estudiado.

Dicho análisis servirá para retroalimentar el presente trabajo, ya que producto de dicho análisis se obtendrán los requerimientos necesarios para poder realizar la correcta implementación de los mismos, con el propósito de agregar una característica que permita que el producto final destaque de entre los ya ofertados, además de enfocarlo hacia el mercado ecuatoriano.

### 2.2. Metodología informática

Ya que solo una persona estará a cargo del desarrollo de este trabajo de titulación, se utilizará como metodología informática el "*Personal Software Process*" (PSP) planteado por Watts Humphrey.

El proceso de software personal proporciona a los ingenieros un marco personal disciplinado para realizar el trabajo de software. El proceso de PSP consiste en un conjunto de métodos, formularios y scripts que muestran a los ingenieros de software cómo planificar, medir y administrar su trabajo. El PSP está diseñado para usarse con cualquier lenguaje de programación o metodología de diseño, y puede usarse para la mayoría de los aspectos del trabajo de software, incluidos los requisitos de escritura, la ejecución de pruebas, la definición de procesos y la reparación de defectos. Cuando los ingenieros usan un PSP, el

objetivo recomendado es producir productos sin defectos a tiempo y dentro de los costos planificados. El PSP ha sido eficaz para ayudar a los ingenieros a alcanzar estos objetivos (Humphrey, 1997).

El paradigma de programación a utilizar en este trabajo será el orientado a objetos, el cual es el paradigma dominante en la actualidad y ha reemplazado las técnicas de programación estructurada que se utilizaba en el pasado (Joyanes Aguilar, 2013). La programación orientada a objetos tiene como objetivo principal el organizar el código de un aplicativo y facilitar su entendimiento. Los objetos, dentro de este paradigma, son aquellos entes que hacen parte del problema (García Llinás L. , 2010).

El lenguaje a utilizar será Java ya que es el lenguaje principal del IDE Android Studio, cuya utilización se ha definido en el tercer objetivo específico como la herramienta elegida para el desarrollo del aplicativo.

## 3. Marco de referencia

### 3.1. Marco teórico

La realización del presente proyecto de disertación coloca el punto de partida en la aplicación del método deductivo, del cual se habló en la sección 2.1. Como se ha mencionado, el mismo comienza analizando principios universales y aplicándolos a hechos particulares. Los principios universales adquiridos durante los estudios de la Ingeniería en Sistemas y Computación serán la base para poder realizar este trabajo. Aunque, de forma más específica y haciendo uso de un nivel alto de abstracción, se puede deducir que los principios universales considerados más importantes para este trabajo son dos: la gestión financiera personal y el desarrollo de aplicativos móviles. El propósito es conjugar ambas áreas del conocimiento para obtener un producto útil y práctico.

#### 3.1.1. Gestión financiera personal

El dinero es un aspecto importante para todas las personas. La correcta administración del mismo puede garantizar un mejor futuro que cuando el dinero no es en absoluto administrado. En este trabajo se considerará que no solamente las empresas deben hacer uso de herramientas financieras. A nivel personal también se pueden adoptar y adaptar conceptos de nivel macro que permitan el control sobre el flujo de ingresos y egresos de una persona.

Sánchez (como se citó en Córdoba, 2012) piensa que la gestión financiera es un proceso que involucra los ingresos y egresos atribuibles a la realización del manejo racional del dinero en las organizaciones.

Esto permite definir el objetivo básico de la gestión financiera desde dos elementos: la generación de recursos o ingresos; y en segundo lugar, la eficiencia y eficacia en el control de recursos financieros, con el fin de obtener niveles aceptables y satisfactorios en su manejo.

La función primordial de las finanzas es decidir cómo asignar ciertos recursos, lo que incluye adquirirlos, invertirlos y administrarlos. Camacho y López (como se citó en Córdoba, 2012) afirman que la gestión financiera se interesa en la adquisición, financiamiento y administración de activos con alguna meta global en mente. La gestión financiera convierte una misión y visión en operaciones monetarias.

La meta global en mente que una empresa puede tener puede ser muy variada, pero generalmente giran en torno a ser la mejor en su ámbito, ganar clientela, reducir costes, incrementar utilidades, etc. La meta global en mente de una persona puede ser ahorrar, con el propósito de fortalecer su futuro financiero y volverlo más alentador en el caso de tener que afrontar imprevistos.

El ahorro es un tema importante para poder tener una correcta gestión financiera personal. No es suficiente no gastar todos los ingresos, hace falta plantear una estrategia económica que permita que los egresos sean siempre menores a los ingresos. Las ventajas del ahorro personal son variadas, permite afrontar calamidades con solvencia, atender emergencias o realizar inversiones que, a futuro, aumentarán el capital.

Además, es buena idea analizar la situación financiera personal actual. Es decir, cuantificar todos los ingresos y egresos que se realizan, clasificarlos y obtener decisiones basándose en dichas cifras.

Sucesivamente, se puede establecer un presupuesto tomando en cuenta la clasificación realizada, y aplicarlo. El punto final es llevar a cabo bajo una estrategia con un monitoreo constante, porque puede que las variables iniciales cambien. Por ejemplo, una persona puede obtener un mejor empleo y aumentar sus ingresos, por lo tanto, el análisis realizado debe modificarse, o si los ingresos disminuyen, tiene que reducirse el presupuesto asignado y recalcularse el monto dispuesto para gastos. Por otro lado, los egresos también pueden aumentar o disminuir y la estrategia debe adaptarse a ello. Una gestión financiera es un proceso constante y cambiante, dinámico en el tiempo.

### **3.1.2. Desarrollo de aplicativos móviles**

La Ingeniería en Computación, tiene diferentes facetas que se encargan de distintas problemáticas que se pueden presentar en el mundo de la tecnología. Como ejemplo, podemos hacer referencia al área del hardware, el área dedicada al estudio de redes de computadores, pero este trabajo va a versar sobre el área encargada del software y, más precisamente, sobre su desarrollo.

El software es la parte intangible del computador, es decir, aquello que hace que la parte tangible funcione (Rubio, 2011). Por lo tanto, se puede deducir que es imposible que el hardware funcione sin software y viceversa. Ambas partes se complementan entre sí.

Este trabajo utilizará un par de características del software que Pressman (2010) menciona y es necesario tener en mente para la elaboración del aplicativo propuesto. La primera es que el mismo no se elabora en el sentido clásico. Con ello se hace referencia a que no se utiliza materia prima para su realización. El software utiliza el intelecto y la imaginación para lograr obtener un producto útil. Se podría decir que el único requerimiento físico que se necesita para elaborar software es un computador, obviando aquellos insumos básicos que el mismo necesita para su funcionamiento: luz eléctrica, internet, etc.

La segunda característica propia del software considerada relevante es que el mismo no se desgasta. Esto tiene relación con la característica anterior, ya que un aplicativo no debe ser visto desde el punto de vista convencional de un producto común. Un programa computacional funciona mientras el hardware funcione y al usuario le parezca útil. Se podría decir que la vida del software termina cuando el hardware evoluciona y tiene un mejor acoplamiento con un software distinto, cuando deja de dársele mantenimiento o cuando hay mucha mayor oferta en el mercado que resulta ser mejor que la que se está utilizando.

Existen diferentes recursos tecnológicos que utilizan software. Los dispositivos más comunes son los computadores personales, pero ahora también los televisores inteligentes cuentan con sistema operativo propio, los autos, etcétera. La sociedad se encuentra en la era del internet de las cosas, pero el presente trabajo va a abarcar el tema del software utilizado en los dispositivos móviles, específicamente sobre el sistema operativo Android.

Es necesario tomar en cuenta que un sistema operativo tiene como función administrar los dispositivos de una computadora y permite que los programas se comuniquen de forma más fácil con el hardware (Anasagasti, Costoya, Pérez, & García, 2001). Cabe recalcar que no es un lenguaje de programación. Es decir, no es software que permita

generar más software. Sin embargo, un sistema operativo es una tecnología más compleja que un lenguaje de programación.

El sistema operativo Android fue desarrollado por la compañía del mismo nombre: Android Inc., fundada en el año 2003 y adquirida por Google en julio de 2005 (Luján Castillo, 2015). Ganó popularidad por ser una plataforma libre, en otras palabras, no es necesario pagar por una licencia para utilizarlo.

Android es utilizado como sistema operativo en los dispositivos móviles y cabe mencionar que esta es la industria que más ha avanzado en los últimos tiempos (Luján Castillo, 2015). La evolución de dicha industria ha dado como fruto los teléfonos inteligentes o *smartphones*, los cuales son aparatos de tamaño pequeño con varias capacidades de procesamiento (Baz Alonso, Ferreira Artime, Álvarez Rodríguez, & García Baniello, 2009).

La aparición de los teléfonos inteligentes trajo consigo un gran reto: lograr una comunicación y un funcionamiento veloz, flexible entre el hardware y software de dichos teléfonos y que cumplan las necesidades del usuario. Sin embargo, las empresas que lideran el mercado, no vieron este reto como un impedimento, sino como una oportunidad. Se habla de empresas como Apple, Google, Microsoft, entre otras.

En cuanto a la elaboración como tal, el desarrollo del aplicativo móvil utilizará la metodología propuesta por Watts Humphrey: *Personal Software Process*, conocida también por las siglas: PSP.

El proceso de software personal proporciona a los ingenieros un marco disciplinado para realizar el trabajo de software (Humphrey, 1997). El proceso de PSP consiste en un conjunto de métodos, formularios y scripts que muestran a los ingenieros de software cómo planificar, medir y administrar su trabajo. El PSP está diseñado para usarse con cualquier lenguaje de programación o metodología de diseño, y puede usarse para la mayoría de los aspectos del trabajo de software, incluidos los requisitos de escritura, la ejecución de pruebas, la definición de procesos y la reparación de defectos. Cuando los ingenieros usan un PSP, el objetivo recomendado es producir productos sin defectos a tiempo y dentro de los costos planificados. El PSP ha sido eficaz para ayudar a los ingenieros a alcanzar estos objetivos.

En cuanto al diseño se refiere, se hará un análisis exploratorio en la tienda oficial de aplicaciones de Google: Google Play. La misma es una plataforma que permite que los usuarios descarguen y usen una gran cantidad de aplicaciones elaboradas por terceras personas (Viennot, Garcia, & Nieh, 2014). Una ventaja de esta plataforma es que la misma ha implementado seguridades que impiden que su contenido sea rastreado o indexado por personas sin autorización. Cabe tomar en cuenta que Google Play no tiene el código fuente de los aplicativos ya que los mismos se envían como paquetes binarios comprimidos. Esta última característica garantiza que el código elaborado por el desarrollador no sea publicado o replicado sin autorización, suponiendo que se pueda dar el caso en el que alguna persona logre burlar las seguridades del sitio.

Lo ideal es utilizar un entorno de desarrollo integrado que concuerde con el sistema operativo objetivo. Entiéndase por entorno de desarrollo integrado a aquel programa que permite escribir código, compilarlo y ejecutarlo sin tener que cambiar de aplicación

(Phillips & Payne, 2012). También se conoce por sus siglas en inglés: “*Integrated Development Environment*” y sus siglas correspondientes: IDE.

### **3.2. Marco conceptual**

La información obtenida y mencionada hasta el momento marcará el rumbo en cuanto a la manera de cumplir con los objetivos marcados en este trabajo se refiere. Además, servirá para proponer requerimientos y, mediante una correcta aplicación de metodologías, se ambiciona obtener un aplicativo móvil que cumpla, no solo con los requerimientos adquiridos en la recolección de información, sino también con requerimientos implícitos referentes a calidad de software. De esto último, se encargará la metodología informática seleccionada.

En la sección anterior, específicamente en la 3.1.1. se hizo referencia a la importancia del control de recursos financieros, tanto en los ingresos como en los egresos. El objetivo de este trabajo de disertación no hará referencia sobre cómo aumentar los ingresos ya que eso es muy relativo y depende directamente de las actividades lucrativas del usuario final. Se pretende dar una herramienta que se enfoque en gestionar los egresos.

Como se menciona en el objetivo general, la meta es entregar un aplicativo que sirva para registrar los gastos realizados por el usuario, de forma tal que la administración de los mismos se pueda ver de forma cuantificada, se presten para su respectivo análisis y una posterior toma de decisiones en base a la información obtenida.

La situación actual del usuario permite establecer un punto de partida financiero, si no hay registros de los gastos no hay información suficiente sobre el estado actual de las finanzas personales y este desconocimiento impide estar preparado ante posibles calamidades futuras, sin mencionar el hecho de que la persona no estaría en capacidad de realizar inversiones puesto que el conocimiento que tiene no es suficiente para emprender en estas áreas.

Cabe recalcar la importancia de la organización de los egresos. No es útil contar con un solo repositorio que lleve el registro de todas las salidas de capital. Es aconsejable seccionarlo para una mayor comprensión de los gastos realizados. Con esta idea en mente, se pretenden establecer categorías y organizar las salidas de dinero con el fin de identificar aquellas que están consumiendo mayor cantidad de ingresos e idealizar una forma de minimizarlas.

Además, se hace mención a la importancia del ahorro. El objetivo del producto final debe estar acorde con ello. Por lo tanto, se establecerá un límite definido por el usuario y que le alerte en el momento que lo sobrepase, asumiendo que esto suceda. Entiéndase como un aviso que pretende poner en alerta a los usuarios para que los mismos limiten sus egresos. En cuanto a las variaciones en los ingresos y egresos económicos, el aplicativo presentará la facilidad de poder modificar o variar los mismos para que, dado el caso en el que alguno de ellos cambie, el aplicativo no se vuelva inutilizable.

En la sección 3.1.2. se especifica detalles propios de la informática y Pressman habla sobre visualizar el software de forma distinta a la que se utilizaría para analizar un producto común. El software no se deteriora de la manera en la que lo hace un producto típicamente perecible, depende mucho del hardware, pero éste es elaborado pensando en durar una buena cantidad de tiempo. La forma en la que el software se volvería obsoleto es cuando hay mejores ofertas en el mercado que el utilizado. Por ello, se piensa hacer uso de la observación participante mencionada en la sección 2.1. y realizar una pesquisa dentro de la tienda oficial de aplicaciones de Google para poder tomar ideas uno o varios requerimientos

que permitan obtener un aplicativo con alguna característica distintiva y que no se vuelva obsoleta demasiado rápido. Los resultados de este análisis exploratorio se muestran a profundidad en el apartado 4.1. del presente trabajo.

Pantaleo y Rinaudo (2015) aseguran que la necesidad de estos días es contar con herramientas robustas y procesos maduros. Por ello, como herramienta de programación se utilizará Android Studio que ha demostrado ser robusta en cuanto a elaboración de aplicativos móviles se refiere. Los aplicativos resultantes de trabajar con este programa son para ser utilizados exclusivamente dentro del sistema operativo definido en el presente trabajo de disertación. Otra ventaja de optar por esta herramienta es que es la herramienta oficial provista por Google, el cual es el entorno de desarrollo integrado oficial para el desarrollo de aplicaciones para Android. Para cumplir con el propósito de contar con procesos maduros, se hará uso del PSP. Ya que solo una persona estará a cargo del trabajo de programación, resulta ser una metodología útil, que cuenta con los procesos necesarios para un correcto desarrollo. Los pasos del mismo y su contenido se pueden encontrar en el apartado 4.3. del presente trabajo.

### **3.3. Marco espacial**

La información utilizada para el desarrollo del aplicativo será obtenida mediante la observación participante dentro de la tienda oficial de aplicaciones de Google: Google Play. Específicamente, se hará una recopilación de características que se consideren útiles, tomando como muestra solo los aplicativos cuyo propósito sea parecido al que se tiene pensado realizar, con la finalidad de establecer alguna o varias opciones que diferencien el presente producto de los ya ofertados.

En cuanto a la información que irá cargada por defecto en el aplicativo, será Ecuador el lugar elegido para recopilar información y establecer categorías que vengan inicialmente en la aplicación. De manera objetiva, las categorías por defecto serán los gastos deducibles oficiales publicados por el Servicio de Rentas Internas del Ecuador. Según esta institución (SRI), los gastos deducibles se listan a continuación:

- Vivienda.
- Alimentación.
- Vestimenta.
- Educación.
- Salud.

La unidad de análisis con la que se examinará el producto final será principalmente cualitativa. Se obtendrán opiniones de profesionales y de usuarios sobre el resultado final. Con ello, se podrá obtener retroalimentación para versiones posteriores e ideas de mejora o, si es necesario, una reingeniería completa.

## 4. Desarrollo

### 4.1. Análisis exploratorio

En la tienda oficial de Google se han desarrollado varias aplicaciones referentes a la administración de las finanzas personales. A continuación, se presentarán las observaciones vistas:

Una buena cantidad de aplicaciones dentro de la tienda son retos de tiempo, es decir retan al usuario a que por una cantidad de días puedan evitar gastos innecesarios. Por ejemplo “Reto 30 días” (30 Days Challenge Apps, 2019), cuyos usuarios deben evitar gastos superfluos por 30 días. La aplicación que se propone en el trabajo de disertación no tiene dicho tiempo límite porque lo más probable es que el usuario desinstale el aplicativo una vez transcurrida esa cantidad de tiempo, es decir, una vez finalizado el reto.

Otras aplicaciones son hechas pensando en ahorrar para lograr uno o varios objetivos. Un ejemplo es “SavePal” (Julio, 2020) que motiva al usuario a ahorrar específicamente para realizar un viaje, comprar un auto, comprar una casa, o algún objetivo puntual. El presente trabajo propone que el usuario ahorre de forma constante, que no solamente lo haga hasta poder cumplir una meta, sino que adopte la costumbre de ahorrar.

El aplicativo “Administrador de dinero” (Money Manager, Expense Tracker, Currency Exchange, 2018) es el que más se parece al objetivo de la disertación. En ella se pueden registrar gastos, establecer categorías y visualizar en cuál de ellas ha habido un gasto más fuerte. Sin embargo, el aplicativo propuesto se diferenciará de este porque le permitirá al usuario establecer un límite en sus gastos y avisarle cuando el mismo ha sobrepasado dicho límite.

Finalmente, se encontraron aplicativos que solamente son consejos financieros. Por ejemplo, el aplicativo “Plan de Ahorro e Invertir Dinero” (Negocios, salud y Felicidad, 2018) no contiene información que no se podría encontrar realizando una búsqueda en internet. Es más, se perdería menos tiempo consultando consejos financieros en algún motor de búsqueda que buscando un aplicativo, seleccionando el adecuado, instalándolo y ejecutándolo.

La aplicación que se plantea para el trabajo de disertación permitirá seleccionar una categoría, editar alguna existente, eliminar una que no se utilice o crear una categoría personalizada. Las categorías por defecto serán los gastos deducibles oficiales publicadas por el Servicio de Rentas Internas del Ecuador. También, en el aplicativo propuesto se podrán ver los gastos realizados por el usuario durante el día, la semana y el mes. Además, otra funcionalidad que incluiría el aplicativo, es el poder visualizar en una gráfica cuál es la categoría en la que más dinero se gasta.

En conclusión, el aplicativo móvil propuesto se distinguirá de las aplicaciones consultadas en otorgarle al usuario la facilidad de la completa administración de las categorías en las cuales clasificar sus gastos realizados. Además, dichas categorías, por defecto, serán los gastos deducibles establecidos por el SRI, es decir, será una aplicación enfocada para el uso de los ecuatorianos. Y finalmente, el aplicativo permitirá entregar una gráfica comparativa de las categorías con mayor cantidad de gasto.

### 4.2. Sistematización de la gestión financiera personal

Figueroa (2009) habla sobre cinco etapas para lograr ejercer una correcta planeación de las finanzas personales:

### Etapa 1: Revisar la situación financiera actual

En esta etapa se aconseja revisar los ingresos, egresos y patrimonio que se posee, con el fin de obtener un panorama claro sobre nuestras finanzas actuales.

### Etapa 2: Determinar metas financieras

Se debe establecer metas que estén acordes con el nivel de vida actual.

Puede haber tres tipos de metas, en función del tiempo que tome conseguirlas:

- Metas a corto plazo: Son aquellas que pueden ser alcanzadas en menos de un año.
- Metas a plazo intermedio: Pueden demorar entre 1 y 5 años en alcanzarse.
- Metas a largo plazo: Todas las que tomen más de 5 años en lograrse.

### Etapa 3: Desarrollar un plan de acción

El plan que se elabore no debe ser rígido, debe permitir cambios en cualquier momento debido a imprevistos que se puedan generar.

El resultado de esta etapa puede ser la realización de un presupuesto y estrategias que se adapten a nuestras necesidades, siempre enfocadas al objetivo principal del plan de acción.

### Etapa 4: Poner en marcha el plan de acción

Se aplica el resultado del paso anterior, utilizando el sentido común y cautela necesarios para aumentar la riqueza y mejorar la calidad de vida.

El plan debe tomarse como una hoja de ruta que recuerde los objetivos establecidos, ya que el esfuerzo realizado entregará una recompensa al final.

### Etapa 5: Controlar el progreso, reevaluar y reconsiderar el plan

Periódicamente se deben hacer evaluaciones para examinar el progreso realizado hasta el momento, con el propósito de ajustar el plan o las metas.

Es necesario estar dispuesto a recomenzar todo si el caso lo amerita.

### **Contrastación del proceso con el aplicativo:**

La propuesta de este trabajo de titulación estaría dispuesta a ser una herramienta para el usuario durante las etapas descritas por Figueroa, excepto la 3 y 4.

Durante la etapa 1 es una herramienta útil para registrar gastos e ingresos y poder elaborar el mencionado panorama financiero, necesario para tener un punto de partida con el cual guiarse.

Para la etapa 2, el aplicativo tiene como funcionalidad establecer un límite de gastos establecido por el usuario y se le avisará si los gastos han sobrepasado dicho límite. Por lo tanto, puede ayudar a establecer metas financieras.

Las etapas 3 y 4 están fuertemente relacionadas con la situación financiera, laboral y personal del usuario. El elaborar un plan depende mucho de su ambiente actual y qué tan fluctuante sea. El aplicar un determinado plan es un aspecto relativo que está ligado con la fuerza de voluntad del usuario, está fuera del alcance de cualquier aplicativo móvil.

La etapa final hace una revisión sobre el plan. Aquí se puede retomar la herramienta para revisar el historial de gastos realizados. Se podrá ver si el límite establecido fue rebasado o no. Se verán resultados con los que el usuario podrá formarse una conclusión con respecto a su esfuerzo realizado.

## **4.3. Desarrollo del aplicativo**

### **4.3.1. Planificación**

Se planifica que el aplicativo cuente con las siguientes funcionalidades:

#### **F1. Gestión de datos personales:**

Los datos personales son el conjunto de datos indispensables para el funcionamiento del aplicativo: nombre del usuario, y gasto estimado mensual.

El nombre del usuario es útil para poder personalizar el aplicativo. El gasto estimado mensual es el límite de gastos que el usuario se proponga realizar en el mes.

Los datos mencionados son indispensables para el funcionamiento de la aplicación y deberán ser ingresados desde el primer momento que el usuario utilice el software.

#### **F2. Gestión de registros:**

El usuario podrá ingresar, modificar, eliminar ingresos o egresos y filtrar los realizados durante el día, mes o año en curso.

Para el ingreso de un registro, se brindará la opción de agregar detalles del mismo con la finalidad que el usuario recuerde exactamente a qué se refiere la transacción realizada.

Luego de ingresado un registro, el usuario podrá ver en la pantalla principal si se ha sobrepasado o no el gasto estimado mensual.

#### **F3. Gestión de categorías:**

Como se ha definido, las categorías que irán inicialmente en la aplicación serán los gastos deducibles definidos por el SRI.

Sin embargo, a veces se necesita una mayor clasificación de los gastos. Para cubrir esta necesidad, el usuario podrá agregar, editar o eliminar categorías dependiendo de lo que necesite.

El propósito es entregar un producto que será lo más personalizable posible y que se adapte a lo que el usuario requiera.

#### F4. Exportación de datos:

Se tomará en cuenta la opción de poder exportar los datos como un archivo CSV para que el usuario pueda ver su información personal en una hoja de cálculo. Con esto, podrá llevar un registro de todos sus gastos en un ordenador o en cualquier otro dispositivo tecnológico que pueda manejar este tipo de archivos.

#### F5. Gráfico de gastos:

El aplicativo será capaz de mostrar visualmente las categorías proyectando la cantidad de dinero que se ha invertido en cada una de ellas. El usuario tendrá cuatro opciones para filtrar los datos: gastos del día, del mes, del año o desde siempre.

#### F6. Comandos de voz:

Se podrá hacer el registro de ingresos y egresos mediante comandos de voz. Para esta última parte, se debe utilizar obligatoriamente conexión a internet. Se aprovechará la ventaja de que Android Studio es un IDE provisto por Google y, aprovechando la característica de que todas las herramientas de esta empresa están relacionadas, se utilizará el reconocimiento de voz que tiene disponible para la implementación en aplicativos.

### 4.3.2. Diseño

#### Diagrama de casos de uso:



Figura 1. Diagrama de casos de uso.

## F1. Gestión de datos personales:

En la presente funcionalidad no se considera la opción de eliminación porque son datos que resultan indispensables para el funcionamiento del aplicativo.

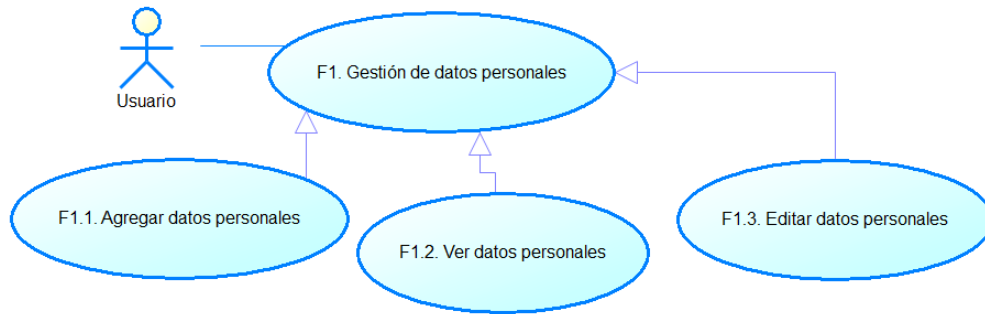


Figura 2. Gestión de datos personales.

### F1.1. Agregar datos personales:

#### Descripción:

En esta opción el usuario podrá ingresar su información personal.

#### Flujo:

1. El usuario abre la aplicación por primera vez.
2. El sistema despliega ventana para agregar/modificar datos del usuario.
3. El usuario ingresa nombre y gasto estimado mensual y presiona "Guardar".
4. El aplicativo verifica que el límite de gastos sea un dato numérico y guarda la información. (E1).

#### Excepciones:

Código	Causa	Acción
E1	Valor no numérico	Mostrar alerta y evitar guardado.

#### Diagrama de secuencia:

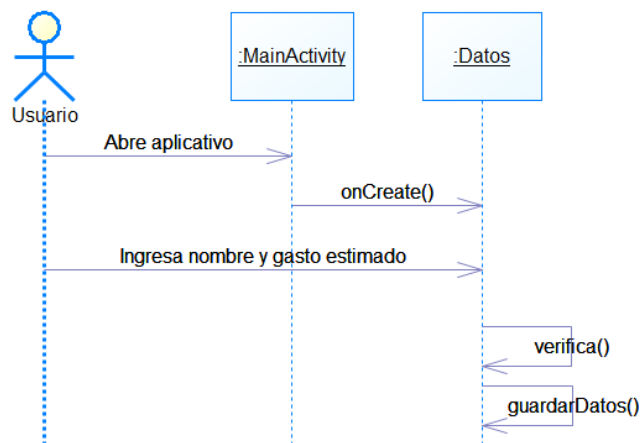


Figura 3. Diagrama de secuencia F1.1.

### F1.2. Ver datos personales

Descripción: En esta opción el usuario podrá visualizar su información personal.

Flujo:

1. El usuario selecciona “Mis datos” en un menú lateral.
2. El sistema despliega ventana con nombre y gasto estimado del usuario.

Diagrama de secuencia:

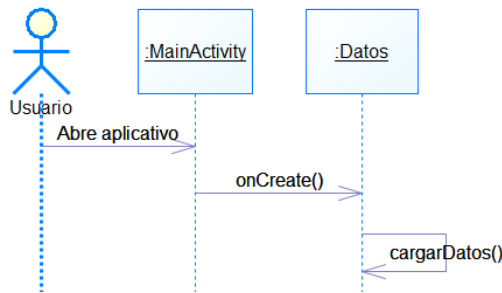


Figura 4. Diagrama de secuencia F1.2.

### F1.3. Editar datos personales:

Descripción: En esta opción el usuario podrá modificar su información personal.

Flujo:

1. El usuario selecciona “Mis datos” en un menú lateral.
2. El sistema despliega ventana para agregar/modificar datos del usuario con su nombre, ingreso y gasto estimado cargados.
3. El usuario modifica los datos necesarios y presiona “Guardar”.
4. El sistema revisa el gasto estimado y modifica los datos. (E1)

Excepciones:

Código	Causa	Acción
E1	Valor no numérico	Mostrar alerta y evitar guardado.

Diagrama de secuencia:

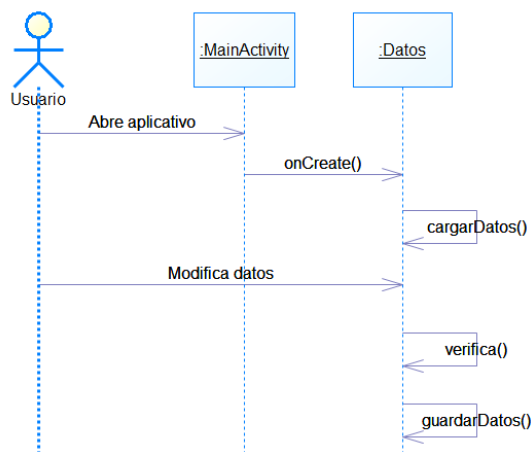


Figura 5. Diagrama de secuencia F1.3.

## F2. Gestión de registros:

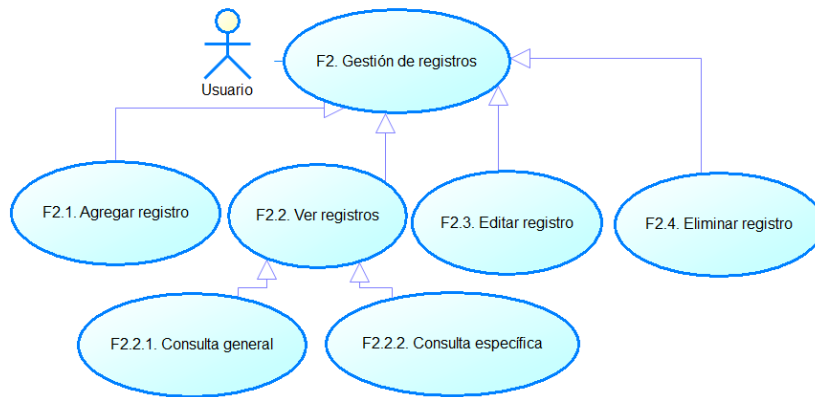


Figura 6. Gestión de registros.

### F2.1 Agregar registro:

Descripción: En esta opción el usuario podrá agregar una transacción realizada.

Flujo:

1. El usuario selecciona “Nuevo registro” en un menú lateral.
2. El sistema despliega ventana para agregar/modificar un registro con las categorías disponibles en un menú desplegable y la fecha actual preseleccionada.
3. El usuario ingresa cantidad, selecciona una categoría, selecciona una fecha e ingresa detalles (de forma opcional) y presiona “Registrar”.
4. El aplicativo verifica que la cantidad sea un dato numérico y guarda el registro. (E1).

Excepciones:

Código	Causa	Acción
E1	Error en la cantidad	Mostrar alerta y evitar registro.

Diagrama de secuencia:

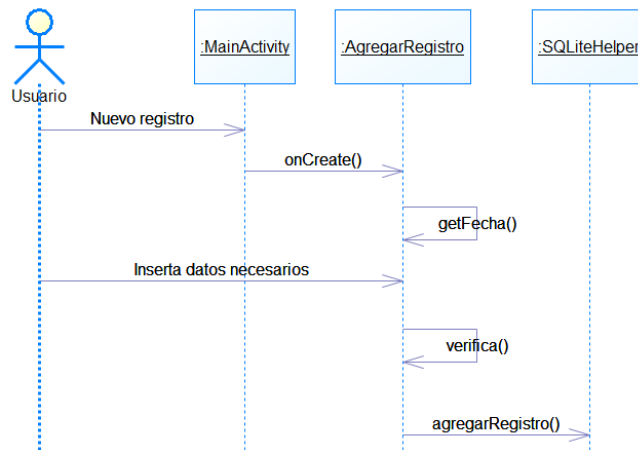


Figura 7. Diagrama de secuencia F2.1.

## F2.2. Ver registros

### F2.2.1. Consulta general

#### Descripción:

En esta opción el usuario podrá visualizar todos los registros realizados.

#### Flujo:

1. El usuario selecciona “Ver registros” en un menú lateral.
2. El sistema despliega ventana con la siguiente información de todos los registros realizados hasta el momento: cantidad, categoría, fecha.
3. El usuario presiona sobre un registro.
4. El sistema despliega los detalles de ese registro. (E1)

#### Excepciones:

Código	Causa	Acción
E1	Registro sin detalles	Informar que el registro no tiene detalles.

#### Diagrama de secuencia:

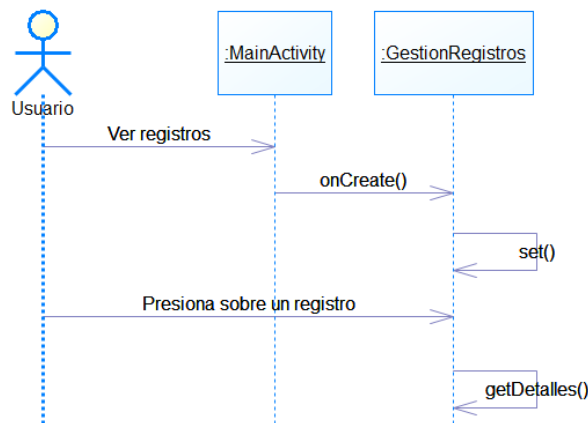


Figura 8. Diagrama de secuencia F2.2.1.

### F2.2.2. Consulta específica

#### Descripción:

En esta opción el usuario podrá filtrar los registros realizados por categoría y/o tiempo.

#### Flujo:

1. El usuario selecciona “Ver registros” en un menú lateral.
2. El sistema despliega ventana con la siguiente información de todos los registros realizados hasta el momento: cantidad, categoría, fecha.
3. El usuario selecciona las opciones de un menú superior.
4. El sistema despliega la opción de filtrar.
5. El usuario presiona sobre “Filtrar registros”.
6. El sistema despliega ventana con menús desplegables de tiempo (diario, mensual, anual, todos) y categorías.

7. El usuario selecciona un tiempo y/o una categoría y presiona “Filtrar”.
8. El sistema muestra los registros que cumplan con las características que el usuario seleccionó.

Diagrama de secuencia:

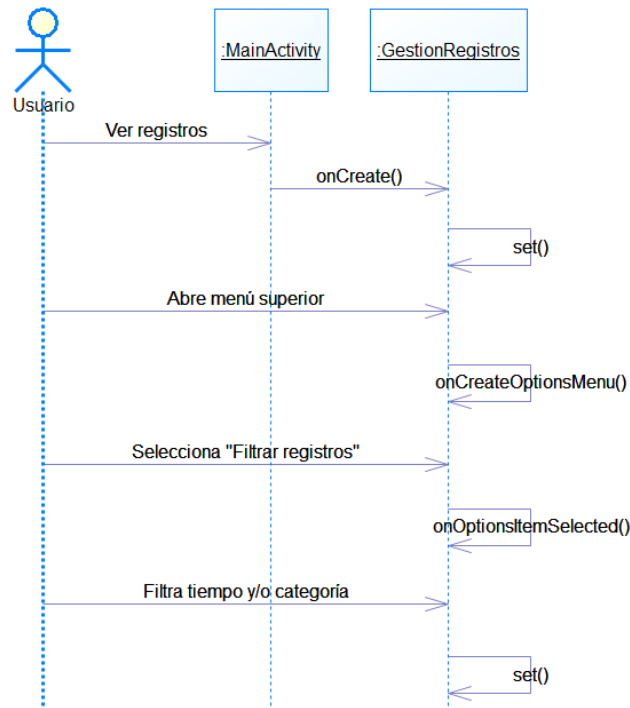


Figura 9. Diagrama de secuencia F2.2.2.

### F2.3. Editar registro

Descripción:

En esta opción el usuario podrá modificar un gasto registrado.

Flujo principal:

1. El usuario selecciona “Ver registros” en un menú lateral.
2. El sistema despliega ventana con la siguiente información de todos los registros realizados hasta el momento: cantidad, categoría, fecha.
3. El usuario presiona prolongadamente sobre un registro.
4. El sistema despliega las opciones de editar o eliminar.
5. El usuario presiona “Editar”.
6. El sistema carga la ventana para agregar/modificar un registro con los datos del registro en ella.
7. El usuario modifica los datos necesarios y presiona “Registrar”.
8. El sistema revisa la cantidad ingresada y modifica el registro. (E1)

Excepciones:

Código	Causa	Acción
E1	Error en la cantidad	Mostrar alerta y evitar registro.

### Diagrama de secuencia:

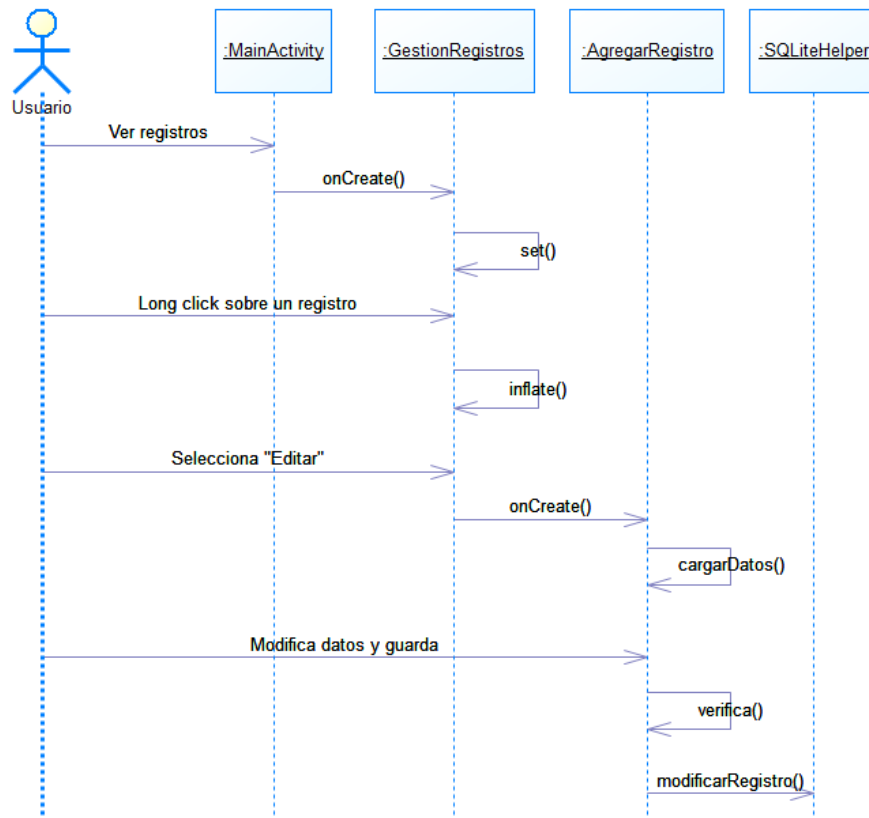


Figura 10. Diagrama de secuencia F2.3.

### F2.4. Eliminar registro:

#### Descripción:

En esta opción el usuario podrá eliminar un gasto registrado.

#### Flujo principal:

1. El usuario selecciona "Ver registros" en un menú lateral.
2. El sistema despliega ventana con la siguiente información de todos los registros realizados hasta el momento: cantidad, categoría, fecha.
3. El usuario presiona prolongadamente sobre un registro.
4. El sistema despliega las opciones de editar o eliminar.
5. El usuario presiona "Eliminar".
6. El sistema pide confirmación sobre la eliminación.
7. El usuario confirma eliminación.
8. El sistema elimina el registro.

#### Flujo alternativo:

7. Si el usuario no confirma eliminación, se regresa a la ventana para visualizar registros.

Diagrama de secuencia:

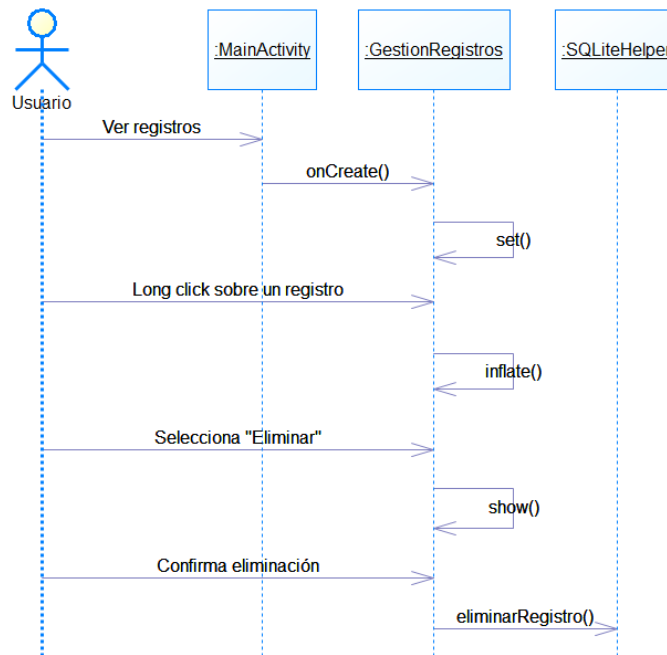


Figura 11. Diagrama de secuencia F2.4.

**F3. Gestión de categorías:**

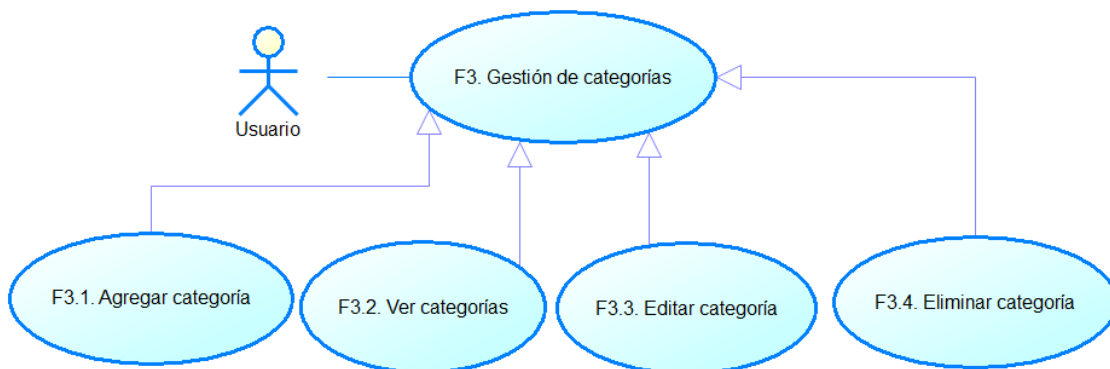


Figura 12. Gestión de categorías.

**F3.1. Agregar categoría:**

Descripción:

En esta opción el usuario podrá ingresar una nueva categoría aparte de las precargadas.

Flujo:

1. El usuario selecciona "Categorías" en un menú lateral.
2. El sistema despliega ventana con las categorías actuales.
3. El usuario selecciona un menú superior.
4. El sistema despliega la opción de agregar categoría.

5. El usuario elige agregar una nueva categoría.
6. El sistema despliega un recuadro para ingresar el nombre de la categoría.
7. El usuario ingresa el nombre y pulsa "Aceptar".
8. El sistema almacena la categoría.

Diagrama de secuencia:

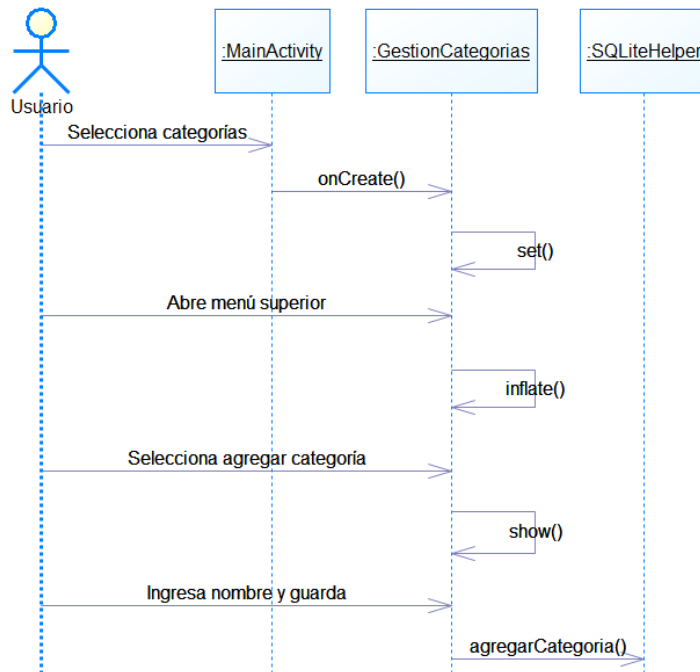


Figura 13. Diagrama de secuencia F3.1.

### F3.2. Ver categorías:

Descripción:

En esta opción el usuario podrá visualizar todas las categorías almacenadas.

Flujo:

1. El usuario selecciona "Categorías" en un menú lateral.
2. El sistema despliega ventana con las categorías actuales.

Diagrama de secuencia:

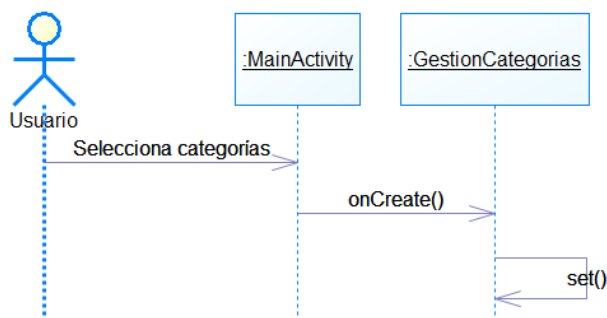


Figura 14. Diagrama de secuencia F3.2.

### F3.3. Editar categoría:

#### Descripción:

En esta opción el usuario modifica una categoría ingresada.

#### Flujo:

1. El usuario selecciona "Categorías" en un menú lateral.
2. El sistema despliega ventana con las categorías actuales.
3. El usuario presiona sobre una de las categorías.
4. El sistema muestra un menú con la opción de Editar/Eliminar.
5. El usuario elige la opción de "Editar".
6. El sistema despliega un recuadro para modificar el nombre de la categoría con el valor actual precargado.
7. El usuario modifica el nombre y pulsa "Aceptar".
8. El sistema modifica la categoría.

#### Diagrama de secuencia:

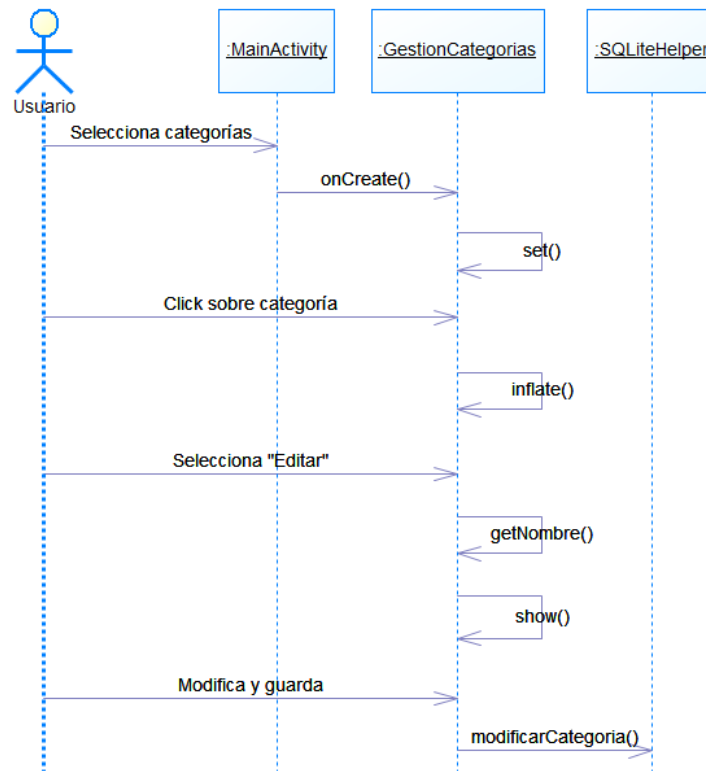


Figura 15. Diagrama de secuencia F3.3.

### F3.4. Eliminar categoría:

Descripción: En esta opción el usuario elimina una categoría ingresada.

#### Flujo principal:

1. El usuario selecciona "Categorías" en un menú lateral.
2. El sistema despliega ventana con las categorías actuales.
3. El usuario presiona sobre una de las categorías.
4. El sistema muestra un menú con la opción de Editar/Eliminar.

5. El usuario elige la opción de “Eliminar”.
6. El sistema pide confirmación sobre la eliminación.
7. El usuario confirma eliminación.
8. El sistema elimina el registro.

Flujo alternativo:

7. Si el usuario no confirma eliminación, se regresa a la ventana para visualizar categorías.

Diagrama de secuencia:

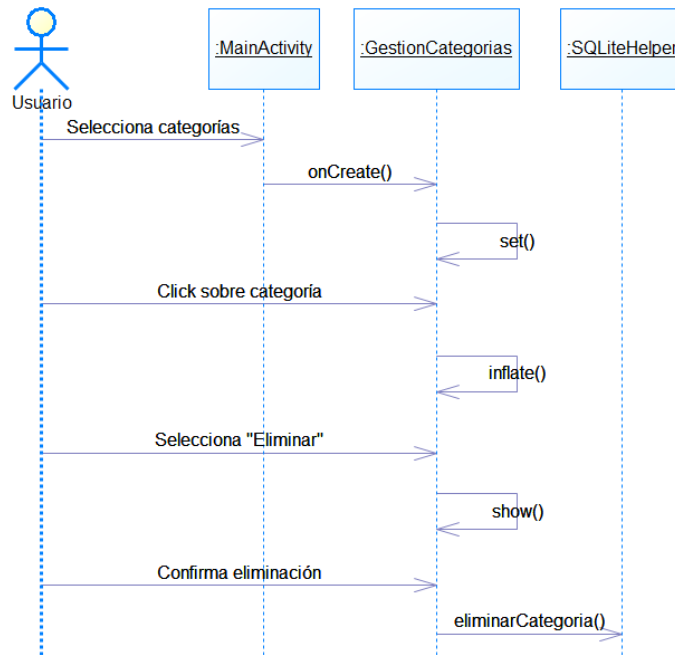


Figura 16. Diagrama de secuencia F3.4.

**F4. Exportación de datos:**

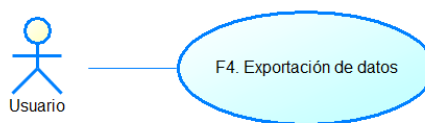


Figura 17. Exportación de datos.

Descripción:

En esta opción el usuario podrá exportar sus datos registrados en un archivo CSV.

Flujo:

1. El usuario selecciona “Descargar CSV” en un menú lateral.
2. El sistema genera archivo CSV y despliega opciones de compartición.
3. El usuario selecciona opción para compartir.
4. El sistema abre la aplicación seleccionada en el punto 2 con el archivo adjunto.

Diagrama de secuencia:

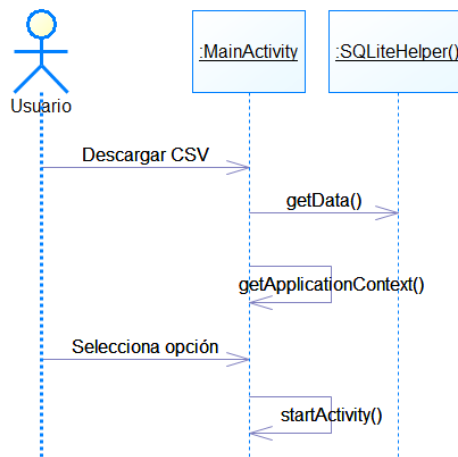


Figura 18. Diagrama de secuencia F4.

**F5. Gráfico de gastos:**

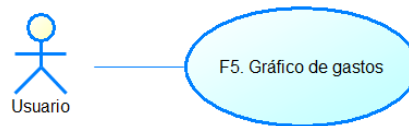


Figura 19. Gráfico de gastos.

Descripción: En esta opción el usuario podrá graficar los gastos de un determinado tiempo, agrupados por categorías.

Flujo:

1. El usuario selecciona “Gráfico” en un menú lateral.
2. El sistema solicita información para mostrar los registros: diarios, mensuales, anuales o todos los existentes.
3. El usuario selecciona una opción de filtrado.
4. El sistema realiza un gráfico de barras con las categorías en el eje X y la cantidad en el eje Y.

Diagrama de secuencia:

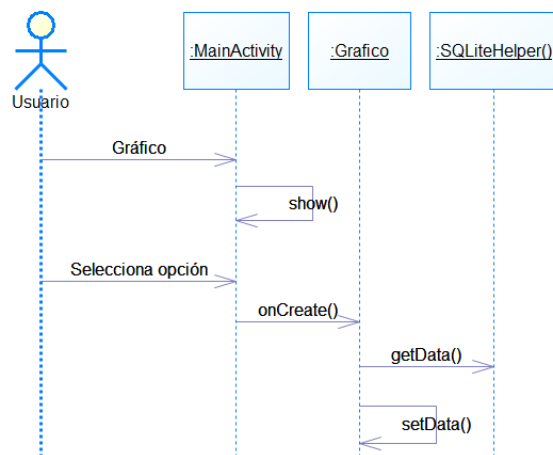


Figura 20. Diagrama de secuencia F5.

## F6. Comandos de voz:



Figura 21. Comandos de voz.

**Descripción:** En esta opción el usuario podrá registrar un ingreso o egreso utilizando comandos de voz y recibiendo confirmación audible por parte del sistema.

### Flujo:

1. El sistema despliega pantalla principal.
2. El usuario hace clic sobre el botón del micrófono.
3. El sistema comienza el reconocimiento de voz.
4. El usuario menciona la categoría y la cantidad, en ese orden.
5. El sistema valida la existencia de la categoría y la validez de la cantidad y almacena el registro. (E1, E2)
6. El usuario recibe confirmación audible del proceso.

### Excepciones:

Código	Causa	Acción
E1	Categoría no encontrada	Mostrar alertas y evitar guardado.
E2	Cantidad no numérica	Mostrar alertas y evitar guardado.

En esta funcionalidad en particular, las alertas deben ser visuales y audibles.

### Diagrama de secuencia:

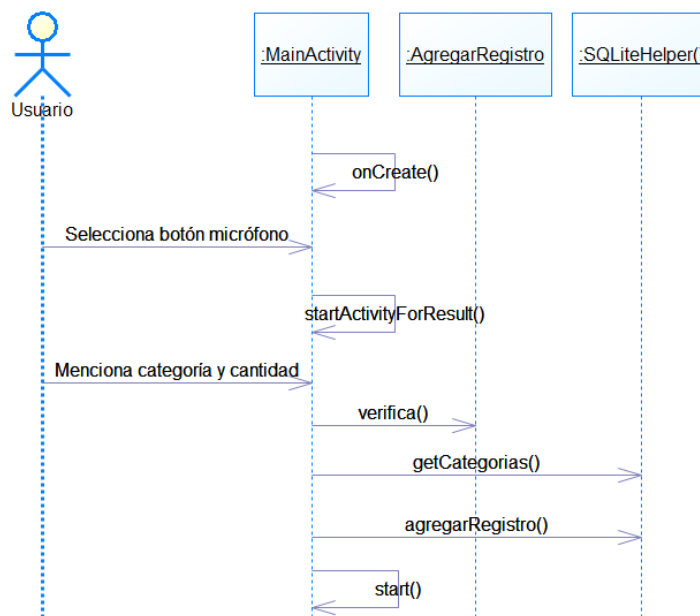


Figura 22. Diagrama de secuencia F6.

### 4.3.3. Codificación

El presente trabajo de titulación utilizó “DB Browser for SQLite” para la gestión de la base de datos ya que es una herramienta idónea para el tratamiento de bases para aplicativos móviles.

Se implementó una base de datos relacional son dos tablas: categorías y registros. El modelo conceptual de la base de datos está estructurado de la siguiente manera:

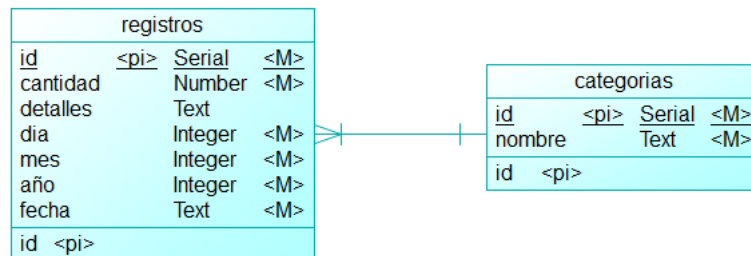


Figura 23. Modelo de la base de datos.

Para la codificación como tal, se aplicó el paradigma de la programación orientada a objetos. Al trabajar con este paradigma, lo principal es entender y modelar las entidades que forman parte del problema para posteriormente brindar una solución utilizando el modelo empleado (García Llinás L. F., 2010).

Se utilizó el estándar camel case para el nombrado de las variables. Este es un tipo de escritura que utiliza una letra mayúscula para identificar dos o más palabras distintas en una misma variable ya que, en programación, no se puede utilizar el espacio para separarlas.

Android Studio fue la herramienta principal de programación. Se utilizó la última versión estable a la fecha: 3.6.3. Este instrumento funciona mediante la implementación de Actividades. Estas Actividades, a su vez, constan de dos tipos de archivos: XML y Java. Los primeros contienen el código correspondiente a la interfaz gráfica visible para el usuario, pero son los segundos los que contienen el funcionamiento de dichos elementos visuales y la lógica de la programación.

A continuación, se listan los archivos Java utilizados para desarrollar el aplicativo, junto con una breve descripción. Cabe recalcar que no se utilizaron tildes en el nombrado de los mismos:

- ❖ AdaptadorRegistro: Sirve para presentar los registros del usuario de una manera personalizada. Con ayuda de esta clase, se puede definir los atributos del registro que son visibles para el usuario. En el diseño se definió que no se muestren los detalles del registro desde un comienzo, sino solamente cuando se haga clic sobre uno de ellos y, con ayuda de esta clase, se puede lograr dicho planteamiento.
- ❖ AdaptadorCategoria: Similar al anterior, pero se encarga de presentar, de manera ordenada, la lista de categorías contenidas en la base de datos.
- ❖ AgregarRegistro: Se encarga de todo lo relacionado con agregar y editar registros. Se aprovechó una misma clase porque la información necesaria para agregar o modificar un registro es la misma. Aquí se verifica que una cantidad monetaria ingresada tenga el formato correcto.

- ❖ Categoría: Contiene los atributos y métodos necesarios para instanciar objetos representativos de las categorías.
- ❖ DatabaserHelper: Ayuda a cargar la base de datos elaborada con el software “DB Browser for SQLite”. Cabe recalcar que esta base tiene las siguientes categorías por defecto: alimentación, educación, ingresos, salud, vivienda, vestimenta, otros.
- ❖ DatePickerFragment: Funciona como auxiliar al momento de seleccionar una fecha para el registro del gasto. Permite que el usuario cuente con un calendario visual del cual seleccionar una fecha, lo cual vuelve el aplicativo más amigable y fácil de utilizar.
- ❖ Datos: Responsable de la gestión de datos personales del usuario, es decir, de la funcionalidad número uno del presente documento. Administra los siguientes datos del usuario: nombre y gasto mensual estimado.
- ❖ GestionCategorias: Responsable de la gestión de las categorías. Es decir, todo lo relacionado con la funcionalidad 3 mencionada en el presente trabajo de titulación.
- ❖ GestionRegistros: Es responsable de mostrar los registros de los gastos realizados por el usuario. Contiene la opción de filtrado de los registros por tiempo y/o categoría. Es decir, el usuario puede filtrarlos según la fecha en que realizó el gasto, por la categoría del mismo o por ambos conceptos a la vez.
- ❖ Registro: Contiene los atributos y métodos necesarios para instanciar objetos representativos de los registros personales del usuario. Esta clase se instancia en muchos de los otros archivos del aplicativo ya que forma parte del objetivo principal del aplicativo.
- ❖ Grafico: En esta sección se cargan los datos necesarios para realizar un gráfico de las categorías con el gasto realizado en cada una de ellas. En otras palabras, se ocupa de la funcionalidad cinco.
- ❖ MainActivity: Es el archivo que se ejecuta al momento de iniciar el aplicativo. Es aquí donde se encuentra el menú lateral y el funcionamiento a realizar cuando se hace clic en alguna de sus opciones. Además, es el lugar donde se verifica si el usuario tiene o no agregados sus datos personales.
- ❖ SQLiteHelper: Se encarga del trabajo directo con la base de datos. Aquí es donde se encuentran los métodos de inserción, modificación, eliminación y lectura de información.

#### **4.3.4. Compilación**

Según la metodología de programación utilizada: PSP, la etapa de compilación se refiere a compilar el programa y corregir los defectos encontrados (Humphrey, 1997). La metodología de PSP menciona el registro de defectos para no volver a cometerlos en un futuro y que sirva como retroalimentación para el programador.

Además, gracias a la instrucción recibida durante los estudios de la Ingeniería en Sistemas y Computación, se conoce que es una buena práctica revisar el código visualmente antes de compilar y ejecutar el programa.

Previa a la compilación del aplicativo, se visualizó que el código tenía librerías instanciadas pero que no se las utilizaba en absoluto; es decir, estaban declaradas, pero no eran utilizadas nunca. Además, había código comentado que a la final se descartó,

pero seguía en los archivos consumiendo recursos de memoria y visualmente no aporta en la organización del código fuente de la aplicación.

Como se ha mencionado en repetidas ocasiones dentro de este documento, el IDE (Entorno de Desarrollo Integrado) utilizado es Android Studio y con este software se realizó la compilación del aplicativo.

Otra herramienta adicional que se utilizó mientras se desarrollaba el aplicativo fue Process Dashboard. Este software ha sido elaborado con el propósito de automatizar la metodología del Personal Software Process. En otras palabras, Process Dashboard permite registrar los proyectos que tiene un desarrollador y asignarles la plantilla del PSP. Con esta herramienta, se puede ir registrando los tiempos que al desarrollador le toma cada fase del proyecto ya que el mismo tiene un diseño de interfaz que tiene como propósito no molestar al programador mientras este se encuentra trabajando. También, permite ir registrando los errores que se encuentren durante el proyecto con el fin de cumplir el concepto de irlos reduciendo progresivamente, ya sea dentro del mismo proyecto o un proyecto consecuente.

#### 4.3.5. Pruebas

Android Studio tiene un componente muy útil al momento de probar los aplicativos: Android Virtual Device Manager, en castellano su nombre sería: Administrador de dispositivos virtuales Android. Este componente permite seleccionar el hardware donde se va a ejecutar el aplicativo desarrollado.

Las categorías de hardware virtual que se pueden elegir son: TV, teléfono, Wear OS, Tablet y Automóviles. Así recordamos que Android no solamente funciona en teléfonos inteligentes, sino en una variedad de dispositivos. Luego de seleccionado el hardware, se elige la imagen del sistema que, para este caso, es la variante del sistema operativo Android que se va a colocar dentro del dispositivo virtual.

Para probar el aplicativo, se utilizó un emulador de teléfono con Android 10 que es, a la fecha, el último lanzamiento del mencionado sistema operativo.

En la figura 24 se aprecia la pantalla principal del aplicativo, mientras que en la figura 25 es una imagen del menú lateral que se utiliza en varias funcionalidades.



Figura 24. Pantalla principal.

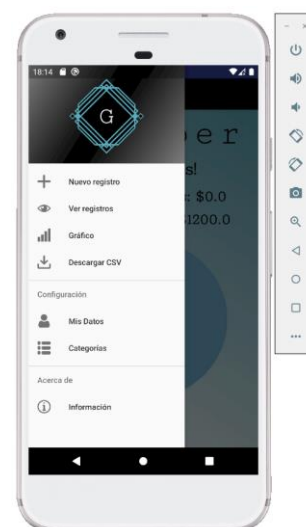


Figura 25. Menú lateral.

A continuación, se muestran tablas correspondientes al proceso de prueba junto con imágenes del proceso de prueba del aplicativo por cada funcionalidad planificada.

## F1. Gestión de datos personales:

### F1.1. Agregar datos personales:

Cuando el usuario inicia la aplicación por primera vez, aparece un recuadro informando acerca de la necesidad de trabajar con datos iniciales.

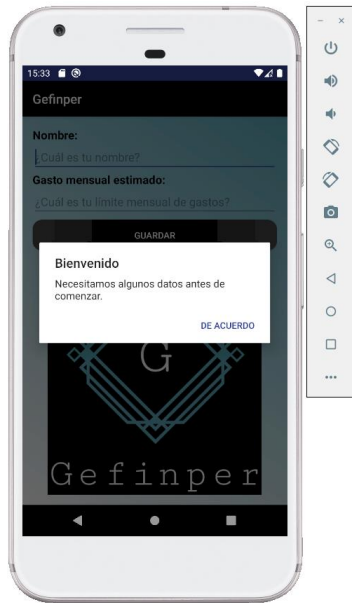


Figura 26. Mensaje de aviso.



Figura 27. Agregar datos personales

Descripción:	Es la primera actividad que realiza el usuario cuando abre la aplicación.
Caso de uso relacionado:	F1.1. Agregar datos personales
Condiciones/requerimientos iniciales:	Abrir el aplicativo por primera vez.
Resultado obtenido vs esperado:	El mensaje de bienvenida se deja de presentar cuando el usuario ingresa sus datos.

Tabla 1. Agregar datos personales.

### F1.2. Ver datos personales:

Se muestra la misma ventana que se le mostró para ingresar sus datos por primera vez, pero con sus datos ya cargados en ella. Para acceder a dicha opción, el usuario debe dar clic en la opción "Mis Datos" dentro del menú lateral del aplicativo.

Descripción:	Ventana para visualizar datos personales ingresados por el usuario.
Caso de uso relacionado:	F1.2. Ver datos personales.
Condiciones/requerimientos iniciales:	Tener datos personales guardados.
Resultado obtenido vs esperado:	Prueba funcional y alineada al diseño. Nombre y gasto se presentan.

Tabla 2. Ver datos personales.



Figura 28. Ver datos personales.

### F1.3. Editar datos personales:

El usuario sigue el mismo procedimiento a realizar para visualizar sus datos, pero realiza un cambio en los mismos y selecciona guardar. Cabe mencionar que, para valores numéricos, se le presenta al usuario un teclado que solo contenga los caracteres necesarios para ingresar cantidades monetarias.

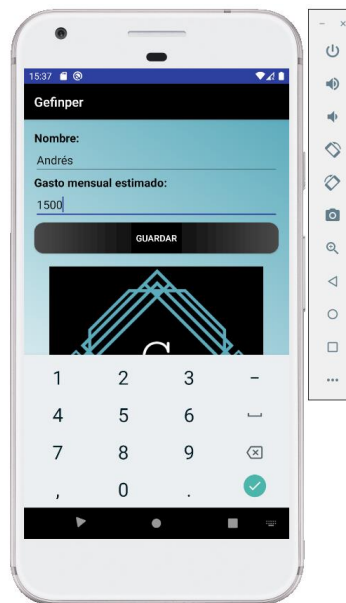


Figura 29. Editar datos personales.

Descripción:	Ventana para editar datos personales.
Caso de uso relacionado:	F1.3. Editar datos personales.
Condiciones/requerimientos iniciales:	Tener datos personales guardados.
Resultado obtenido vs esperado:	Prueba funcional y alineada al diseño. El gasto estimado se verifica antes de guardar.

Tabla 3. Editar datos personales.

## F2. Gestión de registros:

### F2.1. Agregar registro:

La figura 30 muestra la pantalla donde se puede agregar el registro de un ingreso o egreso realizado.

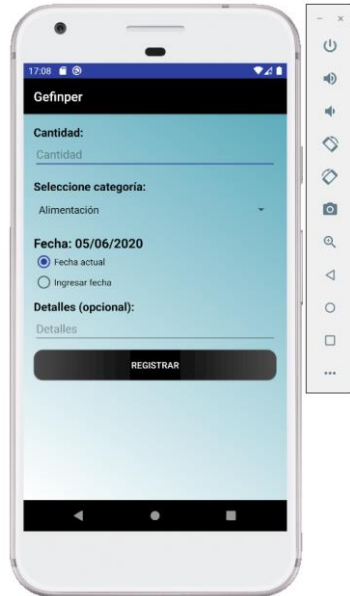


Figura 30. Agregar registro.

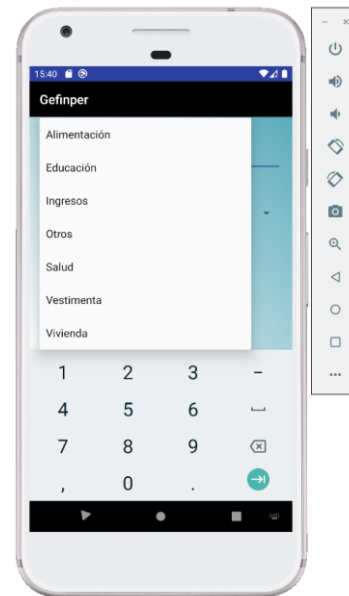


Figura 31. Selección de categoría.

La cantidad se ingresa solamente con el teclado numérico para minimizar errores por parte del usuario.

En la figura 31, se aprecia que la categoría se selecciona de un menú desplegable.

La fecha se puede seleccionar de un calendario gráfico, como en la figura 32.

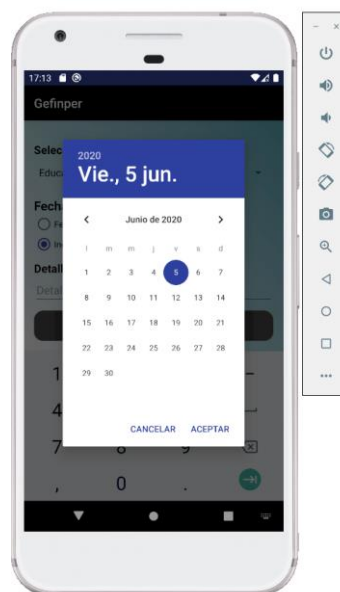


Figura 32. Insertar fecha.

Descripción:	Ventana para ingresar un registro.
Caso de uso relacionado:	F2.1. Agregar registro.
Condiciones/requerimientos iniciales:	Tener gasto estimado mensual guardado.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Si el usuario ingresa más de 2 decimales, la cifra monetaria se redondea automáticamente.

Tabla 4. Agregar registro.

### F2.2.1. Consulta general:

En la figura 33, se puede ver que dentro de una lista se visualiza la cantidad, categoría y fecha de los registros realizados.

Los detalles se visualizan haciendo clic sobre un elemento, si no hay detalles se informa al usuario de ello, como se visualiza en la figura 34.



Figura 33. Consulta general

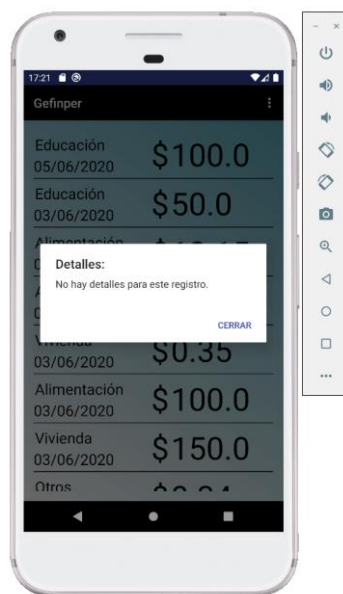


Figura 34. Detalles del registro.

Descripción:	Ventana para consultar registros.
Caso de uso relacionado:	F2.2.1. Consulta general
Condiciones/requerimientos iniciales:	Haber ingresado registros previamente.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Los registros se ordenan del más nuevo al más antiguo.

Tabla 5. Consulta general.

### F2.2.2. Consulta específica:

Haciendo clic en el menú superior derecho se accede a la opción de filtrado de registros.

El usuario puede elegir de un menú desplegable el tiempo y la categoría.

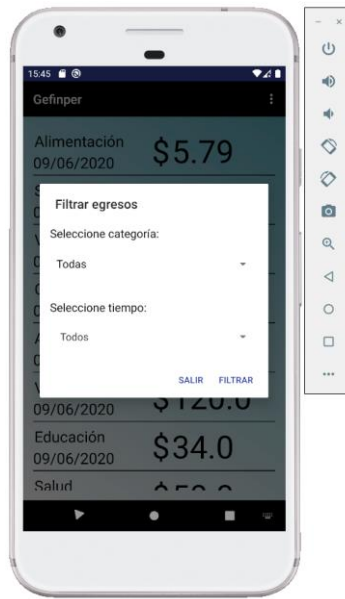


Figura 35. Consulta específica.

Descripción:	Ventana para filtrar registros por tiempo y/o categoría.
Caso de uso relacionado:	F2.2.2. Consulta específica.
Condiciones/requerimientos iniciales:	Haber ingresado registros previamente.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Después de filtrar, se presenta la suma de los registros filtrados.

Tabla 6. Consulta específica.

### F2.3. Editar registro:

La opción de edición se obtiene manteniendo presionado un registro.

La figura 36 indica un menú contextual con las opciones de edición y eliminación.

Se abre la ventana para agregar/modificar un registro con los datos cargados en ella, listos para ser modificados; como se muestra en la figura 37.

Descripción:	Ventana para editar un registro ingresado.
Caso de uso relacionado:	F2.3. Editar registro.
Condiciones/requerimientos iniciales:	Haber ingresado al menos un registro previamente.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Todos los parámetros son editables.

Tabla 7. Editar registro.



Figura 36. Editar registro.



Figura 37. Registro a editar.

#### F2.4. Eliminar registro:

Para eliminar, se sigue el mismo procedimiento de la funcionalidad F2.3. pero en lugar de seleccionar Editar, se selecciona Eliminar y se le pedirá confirmación al usuario de dicha acción.

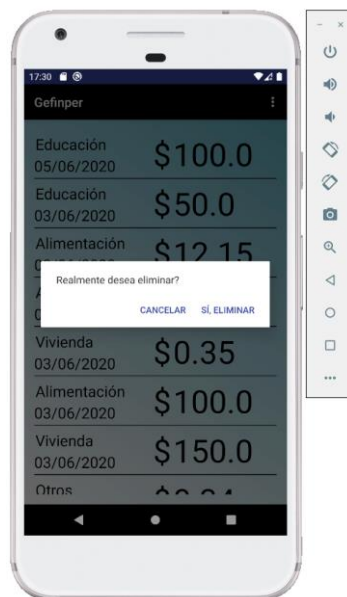


Figura 38. Eliminar registro.

Descripción:	Ventana para eliminar registros.
Caso de uso relacionado:	F2.4. Eliminar registro.
Condiciones/requerimientos iniciales:	Haber ingresado registros previamente.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Una vez confirmado, se elimina el registro de la lista.

Tabla 8. Eliminar registro.

### F3. Gestión de categorías

#### F3.1. Agregar categoría:

Para agregar categoría al usuario se le muestra un campo vacío para ingresar el nombre de la categoría a guardar.

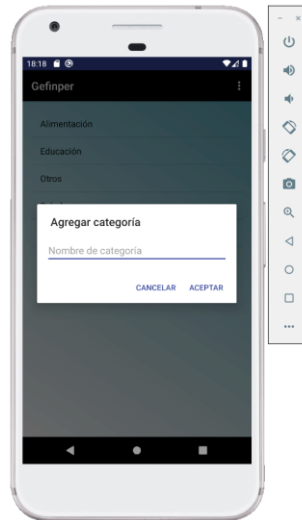


Figura 39. Agregar categoría.

Descripción:	Ventana para agregar categorías.
Caso de uso relacionado:	F3.1. Agregar categoría.
Condiciones/requerimientos iniciales:	Haber ingresado datos personales.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Una vez ingresada la categoría, se muestra en la lista.

Tabla 9. Agregar categoría.

#### F3.2. Ver categorías:

El usuario ve en forma de listado, las categorías disponibles.



Figura 40. Ver categorías.

Descripción:	Ventana para visualizar categorías.
Caso de uso relacionado:	F3.2. Ver categorías.
Condiciones/requerimientos iniciales:	Tener categorías ingresadas.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Una vez ingresada la categoría, se muestra en la lista.

Tabla 10. Ver categorías.

### F3.3. Editar categoría:

Similar que con los registros, para editar, el usuario debe dar un clic en la categoría a editar, saldrá un menú con la opciones Editar o Eliminar, como en la figura 41.

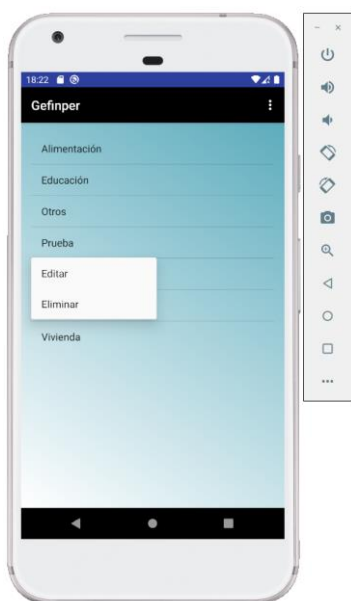


Figura 41. Editar categoría.

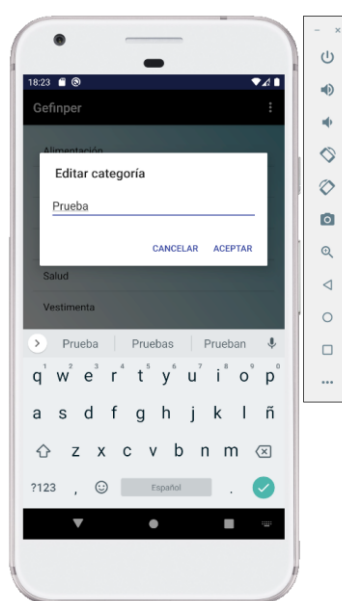


Figura 42. Categoría a editar.

Posteriormente, el sistema efectivamente muestra un mensaje de diálogo con el nombre de la categoría a editar en él, como en la figura 42.

Descripción:	Ventana para editar categorías.
Caso de uso relacionado:	F3.3. Editar categoría.
Condiciones/requerimientos iniciales:	Tener al menos una categoría ingresada.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Una vez editada la categoría, se actualiza la lista.

Tabla 11. Editar categoría.

### F3.4. Eliminar categoría:

El usuario hace clic sobre el registro, selecciona eliminar y el sistema pide confirmación.

Descripción:	Ventana para eliminar categorías.
Caso de uso relacionado:	F3.4. Eliminar categoría.
Condiciones/requerimientos iniciales:	Tener al menos una categoría ingresada.
Resultado obtenido vs esperado:	Prueba alineada al diseño. El nombre de la categoría se muestra en la confirmación de la eliminación.

Tabla 12. Eliminar categoría.

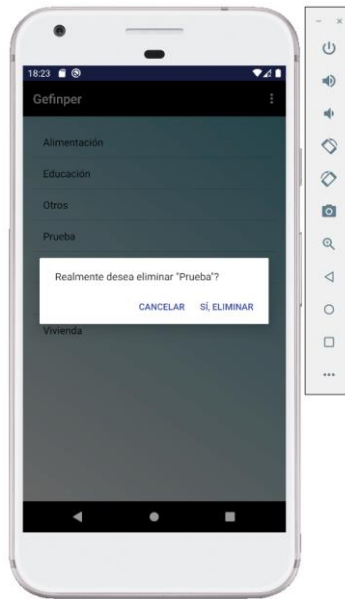


Figura 43. Eliminar categoría.

#### F4. Exportación de datos:

El usuario selecciona “Descargar CSV” del menú lateral, selecciona una forma de compartición y se procede al envío del archivo. En este caso, se probó la exportación a Google Drive.

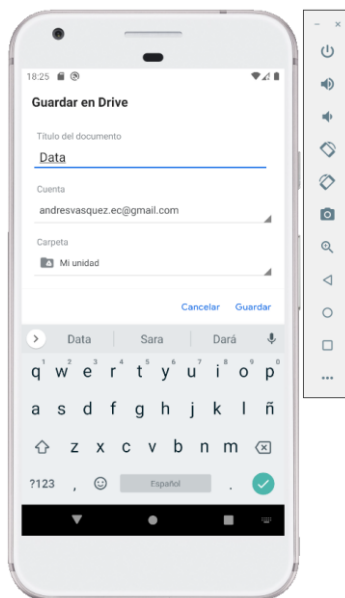


Figura 44. Exportación de datos.

Descripción:	Ventana para exportar registros.
Caso de uso relacionado:	F4. Exportación de datos
Condiciones/requerimientos iniciales:	Tener registros ingresados.
Resultado obtenido vs esperado:	Prueba alineada al diseño. El archivo es editable en una hoja de cálculo.

Tabla 13. Exportación de datos.

### F5. Gráfico de gastos:

El usuario selecciona la opción “Gráfico” del menú lateral y selecciona el tiempo de los registros que desea mostrar: diarios, mensuales, anuales, todos.

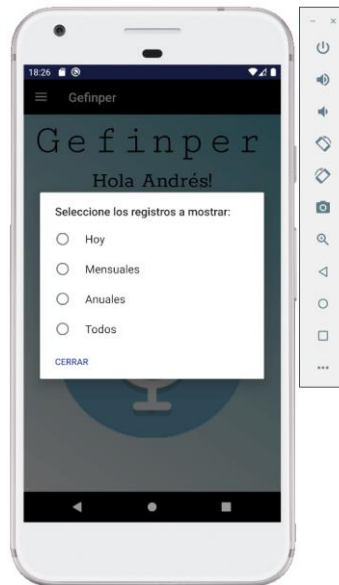


Figura 45. Selección de tiempo.

El sistema grafica los gastos con las categorías en el eje X y la cantidad en el eje Y. Cabe mencionar que el usuario puede hacer zoom en el gráfico para poder ampliar las cantidades o nombres de categorías que no se aprecien a simple vista.



Figura 46. Gráfico de gastos.

Descripción:	Ventana para graficar registros.
Caso de uso relacionado:	F5. Gráfico de gastos.
Condiciones/requerimientos iniciales:	Tener registros ingresados.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Los valores del eje Y se muestran en ambos lados.

Tabla 14. Gráfico de gastos.

## F6. Comandos de voz:

El usuario hace clic en el micrófono de la pantalla principal y se abre el asistente de Google que solicita la categoría y cantidad del registro a ingresar. El ingreso o egreso se agrega con la fecha actual. Si se desea modificarlo para adicionar detalles, lo puede hacer a través de la funcionalidad F2.3. Editar registro.

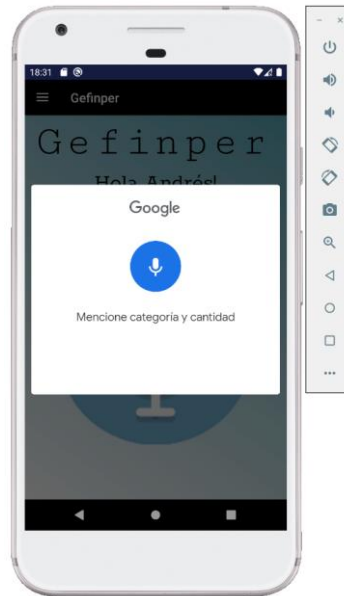


Figura 47. Comandos de voz.

Es importante mencionar que, si hubo un error en el ingreso por comandos de voz o si el registro se realiza de forma exitosa, el aplicativo reproduce un mensaje audible para que el usuario esté informado de la situación en ambas posibilidades.

Descripción:	Ventana para ingresar registros con comandos de voz.
Caso de uso relacionado:	F6. Comandos de voz.
Condiciones/requerimientos iniciales:	Tener conexión a internet.
Resultado obtenido vs esperado:	Prueba alineada al diseño. Los registros se agregan y se confirma audiblemente.

Tabla 15. Comandos de voz.

### 4.3.6. Postmortem

El Postmortem consiste en echar un vistazo atrás, examinar cómo se realizó el trabajo y utilizar este análisis como información que contribuya a la mejora de futuros trabajos.

Para esta etapa se analizaron los datos recolectados con ayuda de la herramienta Process Dashboard mencionada en la sección 4.3.4. Compilación. La misma ayuda a obtener los datos relacionados al tiempo que le toma al desarrollador llevar a cabo su tarea, registrar los defectos encontrados y seguir paso a paso la metodología del Personal Software Process.

Para el desarrollo del proyecto como tal, se estimó un total de 14 días trabajando 8 horas diarias. Es decir, se estimaron 112 horas en total, pero en realidad tomó 87 horas con 31 minutos.

Asumiendo que el presente trabajo se fuera a entregar a un determinado cliente, es mejor que el mismo espere que su producto va a llegar en dos meses y que llegue al mes y medio que ofrecer un producto al mes y medio y entregarlo a los dos meses. Aunque, por otra parte, el tiempo restante antes de la entrega puede ser útil para seguirle haciendo pruebas al producto y mejorar su calidad o agregar alguna funcionalidad extra que haga que el software tenga aún mayor valor adicional.

Process Dashboard entrega un resumen de la información del proyecto en forma de tabla, de la siguiente manera:

## /Gefinper/PSP0

### PSP0 Project Plan Summary

Project Owner	Andrés Vásquez		
Start Date	may 06, 2020 10:42:09 AM		
Completion Date	jun 04, 2020 10:33:44 PM	Completed:	<input checked="" type="checkbox"/>
Keywords	Gestión Financiera Personal		
Language	Java		

Time in Phase (min.)	Plan	Actual	To Date	To Date %
Planning		3:09	3:09	3,6%
Design		5:43	5:43	6,53%
Code		64:42	64:42	73,9%
Compile		5:47	5:47	6,61%
Test		6:59	6:59	7,98%
Postmortem		1:11	1:11	1,35%
Total	112:00	87:31	87:31	

Figura 48. Resumen PSP.

Otra ventaja de la herramienta mencionada, es que se puede ver el tiempo de cada fase en forma de gráfico de pastel:

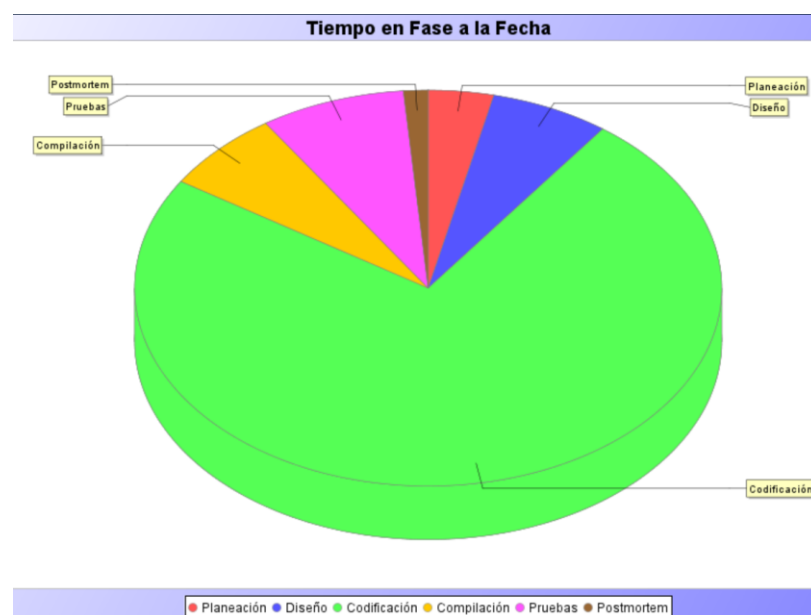


Figura 49. Tiempo en fase a la fecha.

El gráfico ilustra de manera visual los valores porcentuales que se aprecian en el resumen inicial. Se puede ver a simple vista que la codificación es la etapa del proceso que más tiempo toma con casi el 74% del total porque es la parte que se encarga de la construcción del aplicativo como tal. Sin embargo, sin un análisis previo, una planificación, un diseño, etc. No se tendría un punto de partida bien establecido como para empezar a codificar el aplicativo de manera correcta.

En cuanto a los defectos, se tiene las siguientes ilustraciones:

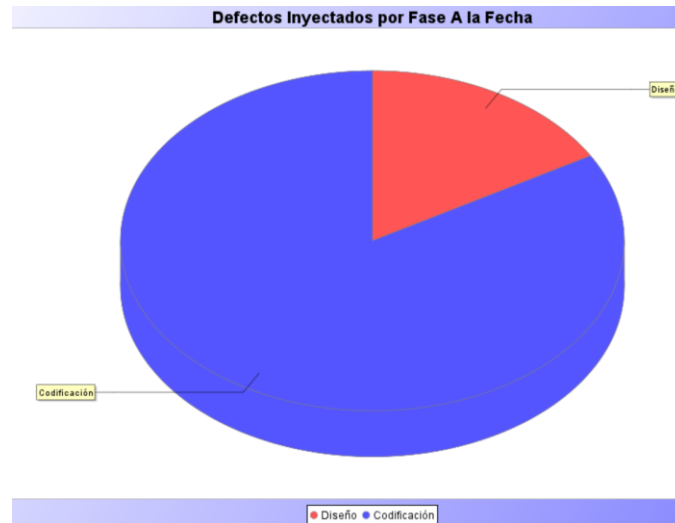


Figura 50. Defectos inyectados por fase a la fecha.

Se puede observar que el 17% de los defectos fueron introducidos en la etapa de diseño, mientras que el 83% restante se introdujeron en la etapa de codificación. Esto tiene sentido porque es también la fase que más tiempo consumió.

Por último, se tiene las etapas en las que los defectos fueron eliminados, con su respectivo valor porcentual:

- Codificación: 16,5%.
- Compilación: 16,5%.
- Pruebas: 67%.



Figura 51. Defectos eliminados por fase a la fecha.

En general, se puede deducir que el haberle dedicado tiempo a la planificación y al diseño hace que se reduzcan errores en la codificación, compilación y pruebas. Se sabe que corregir errores al comienzo de un proyecto es más económico que corregir errores del final ya que se encuentran dentro del funcionamiento de un software ya consolidado. No hace falta mencionar que, en trabajo en equipo, el costo de corregir errores es más costoso porque no solo toma más tiempo, consume también más recursos económicos.

## 5. Conclusiones y recomendaciones

### 5.1. Conclusiones

- ❖ Se concluye que las características recopiladas en el análisis exploratorio permitieron establecer puntos de partida para el diseño del aplicativo. Pero, también se tomaron en cuenta recomendaciones y sugerencias de profesionales que visualizaron el proyecto con las características que, como usuario, les resultaría útiles tener.
- ❖ El proceso de software personal o PSP ha resultado ser una metodología que sirve para el trabajo de una sola persona porque se enfoca en aportar datos que sirvan como retroalimentación para el desarrollador y permitan que sus capacidades puedan mejorar progresivamente con el transcurso del tiempo y la elaboración de más proyectos.
- ❖ El entorno de desarrollo integrado Android Studio fue una herramienta muy útil para la elaboración del aplicativo. Al momento de la creación del proyecto, el programa informa que la aplicación será capaz de ejecutarse en aproximadamente 99,8% de los dispositivos Android. No llega a la totalidad porque hay dispositivos que tienen versiones muy antiguas y Google ya no les proporciona mantenimiento. Sin embargo, esto sigue siendo un valor muy representativo tomando en cuenta que, según García (2019), es casi el 99% de 2.500 millones de dispositivos activos en todo el mundo.
- ❖ Las funcionalidades establecidas para el presente trabajo de titulación, no se encontraron dentro de un solo aplicativo al momento de llevar a cabo el análisis exploratorio. El registro de gastos es algo que se pudo hallar, pero el aplicativo elaborado en este proyecto ofrece funcionalidades útiles para no solamente tener un repositorio de gastos realizados, sino poder escrutarlos y que los datos se conviertan en información.
- ❖ Se puede decir que las tres primeras funcionalidades: gestión de datos personales, gestión de registros y gestión de categorías son indispensables para realizar un aplicativo como el desarrollado; es decir, son necesarias para la obtención y manipulación de datos. Por otra parte, las dos subsiguientes funcionalidades: exportación de datos y gráfico de gastos, son útiles para informar al usuario de su actividad financiera.
- ❖ Con la exportación de datos, el usuario puede utilizar el archivo CSV dentro de una hoja de cálculo y manipular sus registros de alguna forma no contemplada dentro del aplicativo. También, se puede utilizar la exportación de dicho archivo como un respaldo.
- ❖ El gráfico de gastos ayuda a que el usuario pueda analizar de forma visual el aspecto en el cual se está registrando mayor cantidad de gastos y podría ayudarle a tomar decisiones para mejorar su situación económica. Tomando en cuenta las etapas de planeación financiera de Figueroa (vistas en la sección 4.2. Sistematización de la gestión financiera personal), esta funcionalidad resulta muy útil para la primera etapa: revisar la situación financiera actual, porque proporciona una ayuda visual para que el usuario establezca un punto de partida y examine su progreso.
- ❖ Con respecto a la última funcionalidad: comandos de voz, resulta ser una característica llamativa para los usuarios porque ahorran tiempo y también permiten el registro de ingresos o egresos de forma simple y rápida. Para el

planeamiento de esta funcionalidad, se pensó que el usuario tenía su vista ocupada, por esta razón se encuentra en la pantalla principal, que es la primera a mostrarse cuando el usuario abre el aplicativo y, además, se reproduce un mensaje sonoro cuando el registro se realiza correctamente o cuando hay algún error en los datos de ingreso que impiden el almacenamiento del mismo. Los mensajes sonoros utilizados son originales, elaborados por el autor.

## 5.2. Recomendaciones

- ❖ Hacer costumbre del uso de PSP en la vida profesional porque esta metodología funciona mejor mientras más datos tenga con los cuales trabajar. Ayuda a que el desarrollador pueda estimar, de mejor manera, el tiempo que le tomaría implementar un determinado software y aprender de los errores introducidos en el pasado.
- ❖ Se recomienda tener el entorno de desarrollo siempre actualizado a la última versión. Android Studio es una herramienta que está en constante mantenimiento y Google siempre trata de implementar funcionalidades que sean útiles y atractivas para los programadores. Por lo tanto, el entender y manejar la versión más actual de este IDE puede significar ventaja competitiva dentro del desarrollo de aplicativos Android.
- ❖ Realizar más pruebas con mayor cantidad y variedad de usuarios para recolectar comentarios, opiniones y sugerencias de los mismos con la intención de hacer que el aplicativo crezca, ofrezca más funcionalidades y mejore las actuales. Todo esto con la finalidad de realmente poder competir contra aplicativos que actualmente puedan realizar propósitos semejantes o parecidos.
- ❖ Se aconseja seguir en constante análisis de la oferta de aplicativos en la tienda de Google Play para poder tener información suficiente con la cual darle un respectivo mantenimiento y actualización a la aplicación con el propósito que no se convierta en software obsoleto rápidamente.
- ❖ También, es recomendable analizar la posibilidad de realizar una aplicación híbrida. Es decir, un aplicativo que funcione tanto en dispositivos Android, como en dispositivos iOS. Para esto, se debería realizar un análisis exploratorio más extenso no solamente sobre las aplicaciones disponibles en la App Store, que es la tienda oficial de aplicaciones de usuarios iOS; sino también sobre las herramientas que permiten hacer un desarrollo de este tipo. Actualmente hay varias opciones para implementar un desarrollo híbrido, pero se debería hacer una comparativa sobre las ventajas y desventajas de cada una para optar por la que mejor se adapte a un proyecto como el desarrollado en el presente trabajo de titulación.

## Bibliografía

- 30 Days Challenge Apps. (8 de junio de 2019). *Reto 30 días Ahorro de dinero. Ahorra jugando (1.0) [Aplicación Móvil]*. Obtenido de <https://play.google.com/store/apps/details?id=com.challenge30days.savingmoney>
- Anasagasti, P., Costoya, F., Pérez, J., & García, F. (2001). *Sistemas Operativos*. Madrid: McGraw-Hill.
- Android Studio. (2017). *The Official IDE for Android*. Obtenido de Android Studio: <http://teamquadx.org/img/resources/pdf/ControlCheatSheet.pdf>
- Baz Alonso, A., Ferreira Artime, I., Álvarez Rodríguez, M., & García Baniello, R. (2009). *Dispositivos móviles*. EPSIG.
- Bernal Torres, C., Correa Pérez, A., Pineda Ramírez, M., Lemus Hernández, F., Fonseca Yerena, M., & Muñoz Razo, C. (2014). *Fundamentos de investigación*. México: Pearson Educación.
- Corbetta, P. (2007). *Metodología y técnicas de investigación social*. Madrid: McGraw-Hill.
- Córdoba Padilla, M. (2012). *Gestión financiera*. Bogotá: Ecoe.
- Cuello, J., & Vittone, J. (2014). *Diseñando apps para móviles*. Buenos Aires: Duque Giraldo.
- Figuerola Delgado, L. O. (2009). Las finanzas personales. *Escuela de Administración de Negocios*, 123-144. Obtenido de Escuela de Administración de Negocios.
- García Llinás, L. (2010). *Programación orientada a objetos en Java*. Bogotá: Ediciones de la U.
- García Llinás, L. F. (2010). *Programación orientada a objetos en Java*. Barranquilla: Universidad del Norte.
- García Nieto, J. (7 de mayo de 2019). *Xataka Android*. Obtenido de Android suma y sigue: 2.500 millones de dispositivos activos ya lo usan como sistema operativo: <https://www.xatakandroid.com/mercado/android-suma-sigue-2-500-millones-dispositivos-activo-usan-como-sistema-operativo>
- Humphrey, W. (1997). *Introduction to the Personal Software Process*. New York: Pearson Education S. A.
- Hurtado, O. (2018). *Las costumbres de los ecuatorianos*. Quito: Penguin Random House Grupo Editorial.
- INEC. (2018). *Tecnologías de la Información y Comunicación*. Obtenido de [https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas\\_Sociales/TIC/2018/201812\\_Principales\\_resultados\\_TIC\\_Multiproposito.pdf](https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_Sociales/TIC/2018/201812_Principales_resultados_TIC_Multiproposito.pdf)
- INEC. (2019). *Encuesta Nacional de Empleo, Desempleo Y Subempleo*. Obtenido de [https://www.ecuadorencifras.gob.ec/documentos/web-inec/POBREZA/2019/Diciembre-2019/201912\\_PobrezayDesigualdad.pdf](https://www.ecuadorencifras.gob.ec/documentos/web-inec/POBREZA/2019/Diciembre-2019/201912_PobrezayDesigualdad.pdf)
- Joyanes Aguilar, L. (2013). *Fundamentos generales de programación*. México: McGraw-Hill.

- Julio. (9 de febrero de 2020). *SavePal: Ahorra dinero y consigue tus objetivos (1.19.0) [Aplicación Móvil]*. Obtenido de <https://play.google.com/store/apps/details?id=com.saving>
- Luján Castillo, J. D. (2015). *Android. Aprende desde cero a crear aplicaciones*. Madrid: RC Libros.
- Money Manager, Expense Tracker, Currency Exchange. (1 de octubre de 2018). *Administrador de dinero, Rastreador de gastos (1.2.6) [Aplicación Móvil]*. Obtenido de <https://play.google.com/store/apps/details?id=money.expense.budget.wallet.manager.track.finance.tracker>
- Negocios, salud y Felicidad. (14 de agosto de 2018). *Plan de Ahorro e Invertir Dinero (1.0) [Aplicación Móvil]*. Obtenido de <https://play.google.com/store/apps/details?id=com.henry.ahorros>
- Ñaupas, H., Mejía, E., Novoa, E., & Villagómez, A. (2013). *Metodología de la investigación* (Tercera ed.). Bogotá: Ediciones de la U.
- Pantaleo, G., & Rinaudo, L. (2015). *Ingeniería de Software*. Buenos Aires: Alfaomega.
- Phillips, N., & Payne, A. (2012). *Desarrollo*. Madrid: Alianza.
- Pressman, R. (2010). *Ingeniería del Software. Un enfoque práctico*. New York: McGraw-Hill.
- Rubio, A. M. (2011). *Introducción a la informática*. Madrid: Anaya Multimedia.
- SRI. (s.f.). *Gastos deducibles personales*. Obtenido de <https://www.sri.gob.ec/DocumentosAlfrescoPortlet/download/64f73c61-7a22-4f57-96ef-8d07305301cf/Gastos+deducibles+personales.pdf>
- Viennot, N., Garcia, E., & Nieh, J. (2014). *A Measurement Study of Google Play*. doi: 10.1145/2591971.2592003.