



**UNIVERSIDAD TÉCNICA DE AMBATO**  
**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E**  
**INDUSTRIAL**  
**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**  
**E INFORMÁTICOS**

**Tema:**

---

**DESARROLLO DE UNA APLICACIÓN HÍBRIDA E-COMMERCE PARA LA**  
**GESTIÓN DE VENTAS DE LA EMPRESA “CALZADO ANABEL”.**

---

Proyecto de Trabajo de Graduación. Modalidad Proyecto de Investigación, presentado  
previo la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos.

**LÍNEA DE INVESTIGACIÓN:**

Desarrollo móvil

**AUTOR:** Chicaiza Palate Diego Baltazar.

**TUTOR:** Ing. Núñez Miranda Carlos Israel Mg.

Ambato - Ecuador  
Enero, 2020

## CERTIFICACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Investigación sobre el Tema:

**DESARROLLO DE UNA APLICACIÓN HÍBRIDA E-COMMERCE PARA LA GESTIÓN DE VENTAS DE LA EMPRESA “CALZADO ANABEL”**, del señor Chicaiza Palate Diego Baltazar, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el Art. 16 del Capítulo II, del Reglamento de Graduación para Obtener el Título Terminal de Tercer Nivel de la Universidad técnica de Ambato

Ambato, enero de 2020



---

Ing. Núñez Miranda Carlos Israel Mg.

TUTOR

## AUTORÍA DEL TRABAJO

El presente trabajo de investigación titulado: **DESARROLLO DE UNA APLICACIÓN HÍBRIDA E-COMMERCE PARA LA GESTIÓN DE VENTAS DE LA EMPRESA “CALZADO ANABEL”**. Es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, enero de 2020



Chicaiza Palate Diego Baltazar

CC: 1805220942

## DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este trabajo de titulación como un documento disponible para su lectura, consulta y procesos de investigación. Cedo los Derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, enero de 2020



---

Diego Baltazar Chicaiza Palate

CC: 18052209412

## APROBACIÓN DEL TRIBUNAL DE GRADO

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Dennis Chicaiza e Ing. Hernán Naranjo, revisó y aprobó el Informe Final del trabajo de graduación titulado “DESARROLLO DE UNA APLICACIÓN HÍBRIDA E-COMMERCE PARA LA GESTIÓN DE VENTAS DE LA EMPRESA CALZADO ANABEL”, presentado por el señor Chicaiza Palate Diego Baltazar de acuerdo al Art. 17 del Reglamento de Graduación para obtener el título Terminal de tercer nivel de la Universidad Técnica de Ambato.



---

Ing. Mg. Elsa Pilar Urrutia Urrutia  
PRESIDENTE DEL TRIBUNAL



---

Ing. Dennis Chicaiza  
DOCENTE CALIFICADOR



---

Ing. Hernán Naranjo  
DOCENTE CALIFICADOR

## **DEDICATORIA**

Primeramente, al forjador de mi camino, mi padre celestial, él que me acompaña a todo lugar y me levanta siempre del continuo tropiezo y me fortalece del cada golpe que da la vida.

A mi querida madre, quien fue y sigue siendo el pilar fundamental de mi existencia, ella que sin importar las consecuencias me apoyo en todo momento y en todo lugar, es por eso que esta meta de mi vida es fruto del amor y apoyo incondicional.

A mis hermanos y hermanas que me apoyaron en momentos difíciles, gracias a la persistencia y el apoyo que me brindaron.

**Diego Chicaiza**

## **AGRADECIMIENTO**

Agradezco a mi familia quienes me supieron guiar por el camino del bien sabiéndome brindar todo su apoyo, a mi querida madre por estar incondicionalmente en todo momento y siendo el pilar fundamental para estar donde estoy y cumplir un objetivo más en mi vida. A la empresa Creaciones Anabel por abrirme sus puertas y brindarme la información necesaria para la realización de este proyecto de investigación.

A la facultad de Ingeniería en Sistemas, Electrónica e Industrial por haberme brindado la oportunidad de adquirir conocimientos y formarme profesionalmente, así como también a todos los docentes que la conforman quienes supieron transmitir sus saberes a mi persona.

A mi tutor de tesis Ing. Carlos Israel Núñez Miranda Mg. por haberme guiado de manera profesional sin ningún interés.

## ÍNDICE GENERAL DE CONTENIDOS

<b>CERTIFICACIÓN DEL TUTOR .....</b>	<b>ii</b>
<b>AUTORÍA DEL TRABAJO .....</b>	<b>iii</b>
<b>APROBACIÓN DEL TRIBUNAL DE GRADO.....</b>	<b>iv</b>
<b>DERECHOS DE AUTOR .....</b>	<b>v</b>
<b>DEDICATORIA.....</b>	<b>vi</b>
<b>AGRADECIMIENTO .....</b>	<b>vii</b>
<b>RESUMEN.....</b>	<b>xvi</b>
<b>Abstract.....</b>	<b>xvii</b>
<b>Introducción .....</b>	<b>xviii</b>
<b>1. CAPÍTULO I: EL PROBLEMA .....</b>	<b>1</b>
1.1. Tema de investigación.....	1
1.2. Planteamiento del problema .....	1
1.3. Delimitación .....	2
1.4. Justificación.....	3
1.5. Objetivos .....	4
1.5.1. Objetivo General .....	4
1.5.2. Objetivos específicos .....	4
<b>2. CAPITULO II MARCO TEÓRICO .....</b>	<b>5</b>
2.1. Antecedentes Investigativos.....	5
2.2. Fundamentación teórica .....	6
2.2.1. Internet .....	6
2.2.2. Lenguaje de programación.....	7
2.2.3. Metodología de desarrollo.....	7
2.2.4. Aplicación móvil.....	7
2.2.4.1. Aplicación móvil híbrida.....	7
2.2.5. Framework de desarrollo .....	8
2.2.5.1. Análisis y comparación de aplicaciones móviles híbridas y nativas.....	8



2.2.6. Tecnologías web.....	14
2.2.7. Sistemas operativos para dispositivos móviles. ....	15
IOS .....	15
Android .....	15
2.2.8. Comercio Electrónico (E-Commerce).....	15
2.2.9. Pago por medio de Internet .....	16
2.2.10. Base de datos.....	16
2.2.10.1. Base de datos relacional .....	16
2.2.10.2. Bases de datos NoSQL.....	17
2.2.11. Visual Studio Code .....	19
2.2.12. Controlador de versiones.....	20
2.2.13. Node JS .....	23
2.2.14. Android SDK .....	23
2.2.15. GenyMotion .....	24
<b>3. CAPÍTULO III METODOLOGÍA .....</b>	<b>25</b>
3.1. Modalidad Básica de la investigación .....	25
3.2. Recolección de Información.....	26
3.3. Procesamiento y Análisis de Datos .....	26
3.4. Desarrollo del Proyecto .....	27
<b>4. CAPÍTULO IV DESARROLLO DE LA PROPUESTA.....</b>	<b>28</b>
4.1. Metodologías de desarrollo de la ingeniería de software. ....	28
4.1.1. Comparativa de metodologías de desarrollo de software .....	28
Metodologías de desarrollo tradicional .....	28
Metodologías de desarrollo ágil .....	29
Manifiesto por el Desarrollo Ágil de Software .....	29
4.1.2. Diferencias entre metodologías ágiles y no ágiles .....	30
Metodologías de desarrollo ágil .....	31

4.1.3. SCRUM.....	31
4.1.4. EXTREME PORGRAMING .....	35
4.1.5. Análisis comparativo de las metodologías.....	38
4.2. Fase I: Exploración.....	39
4.2.1. Levantamiento de información .....	39
4.2.2. Análisis de resultados.....	40
4.2.3. Estudio de la situación actual de la institución y sus políticas.....	43
4.2.4. Definición de roles .....	44
4.3. Fase II: Planificación del proyecto .....	45
4.3.1. Historias de usuario.....	45
4.3.2. Estimación Historias de Usuario.....	54
4.4. Fase III: Iteraciones .....	56
4.4.1. Plan de entrega .....	56
4.4.2. Iteraciones .....	57
4.5. Fase IV: Producción .....	58
4.5.1. Iteración 1.....	58
4.5.1.1. Definición de la estructura del proyecto .....	59
4.5.1.2. Modelado y estructura de la aplicación.....	59
4.5.1.3. Especificación de buenas prácticas de programación .....	61
4.5.1.4. Modelado de la base de datos.....	63
4.5.1.5. Ingreso al sistema por correo y contraseña .....	67
4.5.2. Iteración 2.....	70
4.5.2.1. Inicio de sesión y registro por redes sociales (Google).....	70
4.5.2.2. Crear una pantalla de inicio .....	71
4.5.2.3. Crear una pantalla para editar los datos del perfil de usuario .....	72
4.5.2.4. Crear una pantalla de productos favoritos.....	73

4.5.2.5. Definir guard para restringir o permitir el acceso a las distintas páginas .....	73
4.5.3. Iteración 3.....	75
4.5.3.1. Crear una pantalla para mostrar los detalles de un producto .....	76
4.5.3.2. Crear pantalla para agregar a favoritos .....	76
4.5.3.3. Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos.....	77
4.5.3.4. Crear una pantalla para enlistar los productos .....	78
4.5.4. Iteración 4.....	79
4.5.4.1. Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras.....	79
4.5.4.2. Crear la pantalla para la forma de pago con PayPal.....	80
4.5.4.3. Crear una pantalla para mostrar las compras realizadas .....	81
4.5.5. Iteración 5.....	82
4.5.5.1. Subir la aplicación al hosting de Firebase.....	82
4.5.5.2. Subir la aplicación a la Google Play .....	83
4.5.6. Codificación .....	84
4.6. Fase V: Pruebas .....	93
<b>5. CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>100</b>
5.1. Conclusiones .....	100
5.2. Recomendaciones .....	101
<b>6. Bibliografía .....</b>	<b>102</b>
<b>7. ANEXOS Y APÉNDICES .....</b>	<b>106</b>

## ÍNDICE DE TABLAS

Tabla 1. Comparativa entre aplicaciones móviles híbridas y nativas. ....	10
Tabla 2. Comparativa de Frameworks Ionic y React Native .....	12
Tabla 3. Funcionalidades principales que provee Apache Cordova .....	14
Tabla 4. Comparativa entre Realtime Database y Cloud Firestore.....	19
Tabla 5. Comparativa entre GitLab y GitHub.....	23
Tabla 6. Recolección de Información .....	26
Tabla 7. Diferencias entre metodologías ágiles y no ágiles .....	30
Tabla 8. Análisis comparativo de las metodologías.....	39
Tabla 9. Roles del proyecto.....	45
Tabla 10. Plantilla de Historia de Usuario. ....	46
Tabla 11. Historia de usuario - Definir la estructura del proyecto.....	47
Tabla 12. Historia de usuario - Especificación de buenas prácticas de programación .....	47
Tabla 13. Historia de usuario - Modelado de la base de datos.....	47
Tabla 14. Historia de usuario -Ingreso al sistema por correo y contraseña .....	48
Tabla 15. Historia de usuario - Registro de usuarios por medio de correo y contraseña .....	48
Tabla 16. Historia de usuario - Restablecer contraseña .....	49
Tabla 17 Historia de usuario - Inicio de sesión y registro de usuario por medio de red social (Google).....	49
Tabla 18. Historia de usuario - Crear una pantalla de inicio.....	49
Tabla 19. Historia de usuario - Crear una pantalla de productos favoritos.....	50
Tabla 20 Historia de usuario - Crear una pantalla para editar los datos del perfil de usuario.....	50
Tabla 21. Definir guard para restringir o permitir el acceso a las distintas páginas ..	51
Tabla 22. Historia de Usuario Crear una pantalla para mostrar los detalles de un producto.....	51
Tabla 23. Historia de Usuario - Agregar productos a favoritos .....	51
Tabla 24. Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos.....	52
Tabla 25. Historia de usuario - Crear una pantalla para enlistar los productos .....	52

Tabla 26. Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras .....	53
Tabla 27. Crear la pantalla para la forma de pago .....	53
Tabla 28. Estimación de historias de usuario.....	55
Tabla 29. Plan de entrega .....	57
Tabla 30. Resumen de iteraciones.....	58
Tabla 31. Colección – Categories .....	64
Tabla 32. Colección – Products .....	65
Tabla 33. Colección – Sales .....	66
Tabla 34. Colección – Countries.....	66
Tabla 35. Colección – States.....	66
Tabla 36. Colección Users .....	66
Tabla 37. Colección – Avatars .....	67
Tabla 38. Corrección – parameters .....	67
Tabla 41. Implementación de método de Pago PayPal.....	93
Tabla 42. Caso de prueba - Inicio de sesión con correo electrónico y contraseña.....	94
Tabla 43. Caso de prueba - Registrar usuario .....	94
Tabla 44. Caso de prueba - Recuperar cuenta.....	95
Tabla 45. Caso de prueba - Inicio de sesión con Google.....	95
Tabla 46. Caso de prueba - Pantalla de lista de productos favoritos.....	96
Tabla 47. Caso de prueba - Actualizar los detalles del usuario autenticado .....	96
Tabla 48. Caso de prueba - ver detalles de un producto seleccionado.....	97
Tabla 49. Caso de prueba - Listar productos.....	97
Tabla 50. Caso de prueba - Lista de productos en el carrito .....	98
Tabla 51. Caso de prueba - Lista de productos a comprar.....	98
Tabla 52. Caso de prueba - Mostrar las compras realizadas de un cliente.....	99

## INDICE DE FIGURAS

Figura 1. Comparativa básica de los planes que provee Firebase.....	19
Figura 2. Sistemas de control de versiones locales .....	21
Figura 3. Sistemas de control de versiones centralizados .....	21
Figura 4. Diagrama de control de versiones distribuido. ....	22
Figura 5. Proceso de la metodología ágil y metodología tradicional. ....	31
Figura 6. Ciclo de vida de Scrum.....	33
Figura 7. Proceso de la metodología ágil Fuente: [35] .....	35
Figura 8. Fase de la metodología de XP .....	37
Figura 9. Sitio web para Creaciones AnabelFuente: [37] .....	41
Figura 10. Ficha técnica Anabel 2019 Fuente: Creaciones Anabel .....	42
Figura 11. Crear una pantalla para mostrar las compras realizadas .....	53
Figura 12. Subir la aplicación en el Hosting que provee Firebase.....	54
Figura 13. Subir la aplicación para la tienda de Google Play .....	54
Figura 14. Definición de estructura del proyecto.....	61
Figura 15. Vista general de las colecciones de Firestore .....	64
Figura 16. Pantalla Inicio de sesión con correo electrónico y redes sociales .....	68
Figura 17. Pantalla Registro de usuario .....	68
Figura 18. Pantalla: Verificación de cuenta .....	69
Figura 19. Pantalla Recuperación de cuenta .....	70
Figura 20. Inicio de sesión y registro por redes sociales (Google) .....	71
Figura 21. Pantalla Principal .....	72
Figura 22. Pantalla - Perfil de Usuario.....	72
Figura 23. Pantalla - Lista de Favoritos .....	73
Figura 24. Guard AuthGuard .....	74
Figura 25. Guard NologinGuard .....	75
Figura 26. Definición de rutas en AppRoutingModule.....	75
Figura 27. Pantalla - Detalle del Producto .....	76
Figura 28. Agregar a la Lista de Favoritos.....	77
Figura 29. Control en la Pantalla de Lista de Favoritos.....	78
Figura 30. Pantalla - Lista de Productos .....	79
Figura 31. Pantalla - Carrito de compras .....	80

Figura 32. Proceso de pago .....	81
Figura 33. Pantalla – Ordenes .....	81
Figura 34. Administrador de sitios de Firebase .....	83
Figura 35. Aplicación desplegada en el Hosting de Google Play .....	83
Figura 36. Publicación en la tienda Google Play Store.....	84
Figura 37. Método de inicio de sesión con correo y contraseña .....	85
Figura 38. Inicio con red social con Google nativo y popup para navegadores .....	86
Figura 39. Inicio de sesión con Google en ambiente nativo. ....	86
Figura 40. Inicio de sesión con Facebook.....	87
Figura 41. Cerrar sesión .....	87
Figura 42. Método para listar productos .....	89
Figura 43. Añadir a la lista de Favoritos almacenados en LocalStorage .....	90
Figura 44. Mostrar productos en la vista .ts .....	90
Figura 45. Ejemplo para mostrar los productos en la vista .html.....	91

## **RESUMEN**

El presente trabajo de investigación está basado en solventar en parte la gestión de ventas, la administración de inventario, existencias y costos de los productos, debido a que la mayor parte de pequeñas y medianas empresas no tienen medios o no son eficientes para promocionar sus productos. El presente proyecto está enfocado al mercado del calzado.

Debido al auge que han tenido los teléfonos inteligentes, por tanto, muchas aplicaciones para diferente ámbito se han venido desarrollando. Además, la tecnología avanza en cada sector, es por eso por lo que el proyecto de investigación se enfoca en los dispositivos móviles. Al ser desarrollado con Ionic se tratará de un desarrollo multiplataforma, sin embargo, para el presente proyecto se trabajará con la plataforma de Android.

La aplicación se enfoca en promocionar los productos de la empresa para que los clientes mayoristas y los consumidores finales puedan adquirirlos a través de esta. Se añadió el método de pago PayPal con cuentas de desarrollador, de esta manera se pueden realizar simulaciones de pago.

El presente proyecto se realizó como el Framework de Ionic en su versión 4, mismo que utiliza Typescript como lenguaje de programación. A través de la metodología Extreme Programming ayudo a llevar tiempos cortos de entrega continua del producto y la satisfacción del cliente. Además, se utilizó un servicio como Backend como lo es Firebase, misma que ayuda la sincronización de los datos en tiempo real. Adicionalmente, la aplicación se encuentra publicada en la Play Store y en el Hosting Firebase.



## **ABSTRACT**

This research work is based on partly costing sales management, because most small and medium-sized enterprises (Pymes) do not have the means to promote their products and thus expand in the specific market of each.

Due to the boom that smartphones have had, therefore, many applications for different fields have been developing. In addition, technology advances in each sector, which is why the research project focuses on mobile devices. Being developed with Ionic will be a cross-platform development, however, for this project will work with the Android platform.

The app focuses on promoting the company's products so that wholesale customers and end consumers can purchase them through it. Added payment methods as is PayPal with developer accounts, this way you can perform payment simulations.

This project was realized as the Ionic Framework in version 4, which uses Angular 7 as the programming language. Through the Extreme Programming methodology to bring short times of continuous product delivery and customer satisfaction. In addition, a service such as Backend was used, such as Firebase, which helps synchronize data.

## INTRODUCCIÓN

El desarrollo del presente proyecto de investigación con el nombre de tema, "DESARROLLO DE UNA APLICACIÓN MÓVIL HÍBRIDA E-COMMERCE PARA LAS VENTAS DE LA EMPRESA CREACIONES ANABEL" consta de cinco capítulos que se muestran a continuación para la comprensión del contenido del proyecto.

**Capítulo I:** "EL PROBLEMA", trata sobre la necesidad de la empresa, identifica el problema expandir sus productos en el mercado, la justificación y cuáles son los objetivos a cumplir.

**Capítulo II:** "MARCO TEORICO", contiene la recopilación de información, en donde se presenta antecedentes investigativos, teórica que es necesaria para entender problemas similares al caso y de esa manera proponer solución a la problemática.

**Capítulo III:** "METODOLOGÍA", determina la modalidad de la investigación, así como la recopilación de información. el procesamiento, y las etapas del desarrollo del proyecto.

**Capítulo IV:** "DESARROLLO DE LA PROPUESTA", presenta el desarrollo de la aplicación bajo la metodología seleccionada.

**Capítulo V:** "CONCLUSIONES Y RECOMENDACIONES". presenta conclusiones adquiridas y recomendaciones a considerar.

Por último, se encuentran los anexos que corresponden al proyecto de investigación.

# **CAPÍTULO I:**

## **EL PROBLEMA**

### **1.1. Tema de investigación**

Desarrollo de una aplicación móvil híbrida e-commerce para las ventas de la empresa CREACIONES ANABEL.

### **1.2. Planteamiento del problema**

Con la aparición de las Tecnologías de la Información y Comunicación (TIC) entre los años 90, el mundo ha sufrido un cambio drástico debido a que las mismas se fueron creando para distintas disciplinas como, e-learning, e-government, e-business, e-commerce, entre otras tecnologías. E-Commerce ha revolucionado el mercado tradicional, tomando en cuenta que la mayoría de las empresas cuentan con acceso a Internet se han planteado a implementar servicios virtuales, flexibles conocido como las PyMEs en el comercio electrónico y en el mercado global.

El impacto que tienen las nuevas tecnológicas en la económica, se manifiesta en los cambios que suceden en las formas que adopta el comercio. En las nuevas estructuras organizativas y, en general, en los hábitos de productores y consumidores. Las nuevas tecnologías y su descomunal desarrollo han permitido el crecimiento de muchas empresas.

Internet ha llegado a ser algo indispensable en los hogares es así por medio de dispositivos electrónicos como tabletas, smartphones, computadoras, entre otros medios digitales pueden acceder a varios servicios como es redes, sociales, contenido multimedia, aplicaciones, entre otros. Según la revista M4rketing E-commerce para el 2018 el número de usuarios supera el 50% de la población, un estimado de 4000 millones de usuarios. [1]

E-Commerce o comercio electrónico es el medio por el cual un cliente o entidad puede realizar compraventa de productos o servicios en los distintos países por medio de un

dispositivo conectado a Internet. El crecimiento de comercio electrónico en los distintos países no solo depende de su infraestructura tecnológica sino también de la velocidad en que se eliminan las barreras en los consumidores. Genera cambios en su comportamiento, es diferente al que tiene cuando compra en una tienda física.[1]

La creación de la lealtad del cliente es importante por parte de los vendedores en línea para atraer y crear las tareas necesarios para mantener clientes es bastante alto, que es del 20% - 40% superior en comparación con los mercados tradicionales.[2]

En Ecuador las empresas familiares o pequeñas son conocidas como Pymes, por un lado, existen aquellos que tienen una estructura, organización y donde existe una gestión empresarial. Estas, en su mayoría, tienen un capital y se desarrollan dentro del sector formal de la economía. [3]. Por otro lado, se encuentran las que vienen de orígenes familiares cuya característica principal es la gestión guiada hacia la supervivencia sin darle importancia al costo, capital o inversión que permita el crecimiento de esta.[3]

Según un boletín estadístico publicado en junio del 2017 por la Agencia de Regulación y Control de Telecomunicaciones (ARCOTEL), la estadística del internet móvil ha tenido un crecimiento importante desde 2010, con un 46.4 % de usuarios del Servicio de Móvil Avanzado tienen un smartphone y 8.1 millones de personas cuentan con acceso a Internet.[4]

En la ciudad de Ambato, “Creaciones Anabel” empresa que da servicio de venta de calzado debido al interés de seguir expandiéndose y ganando mercado. Es por esta razón que ha visto necesario la implementación de un sistema para realizar ventas online de sus productos. A través de esta implementación, los clientes podrán ver su catálogo y si es el caso adquirirlos.

### **1.3. Delimitación**

**Área Académica:** Software

**Línea de Investigación:** Desarrollo de Software

**Sublíneas de Investigación:** Aplicaciones móviles.

**Delimitación Espacial:** Ciudad de Ambato. (La aplicación propuesta está delimitado a realizar compras online, el área geográfica se manejará a nivel de la provincia de Tungurahua, el objetivo es vender la aplicación a las empresas que ofrecen este servicio).

**Delimitación Temporal:** La presente investigación se desarrollará en el septiembre 2018- febrero 2020.

#### **1.4. Justificación**

El desarrollo del presente trabajo es de utilidad para las pequeñas y medianas empresas (Pymes), debido a que el avance tecnológico en dispositivos móviles es muy grande, y más aún cuando estas herramientas son de gran ayuda para realizar varias tareas que los usuarios requieren. Por medio de la implementación los usuarios tendrán acceso a los productos, ver precios, existencias y así mantener a los clientes satisfechos y buscar la lealtad hacia la empresa.

A través del E-COMMERCE o comercio electrónico, el tiempo y la distancia donde se encuentren los clientes ya no es un obstáculo, de esta manera un servicio se puede dirigir a un grupo de personas masivamente.

La presente investigación es de suma importancia ya que se desea demostrar que las Pymes de la provincia de Tungurahua pueden ingresar al mundo de los negocios por Internet, ya que el uso del comercio electrónico abre una serie de oportunidades y de retos primordiales para la sociedad, debido a las ventajas de la implementación.

Entre las principales ventajas se destacan el ahorro de tiempo en los desplazamientos, la desaparición de barreras entre comprador y vendedor, además de la reducción de gastos operativos de los productos y servicios; entregando de esta manera un matiz diferente y un valor agregado al consumidor.[5]

Tungurahua al ser una zona con amplios mercados como el calzado es viable crear una aplicación que mejore el posicionamiento comercial en el mercado, ya que impactará a sus clientes con su forma de dar a conocer sus productos.

El desarrollo del presente proyecto tendrá un gran impacto principalmente en las Pymes, debido a que muchas de las mismas no cuentan con un servicio de ventas ya sea en sitios web o en aplicaciones móviles, misma que podrá beneficiar a la empresa a ganar mercado por medios digitales.

La aplicación móvil híbrida se desarrollará para dispositivos móviles que cuenten con la plataforma Android, que hoy en día está a la vanguardia de las tecnologías móviles. La cual está creciendo a pasos agigantados en el mercado. Adicionalmente la aplicación estará disponible para acceder desde cualquier navegador web actualizado.

## **1.5. Objetivos**

### **1.5.1. Objetivo General**

Implementar una aplicación móvil E-Commerce para las ventas de la empresa Creaciones Anabel.

### **1.5.2. Objetivos específicos**

- Comprender el modelo de negocio de un sistema E-Commerce para la gestión de ventas de la empresa de calzado.
- Analizar los diferentes Frameworks y herramientas de desarrollo que serán utilizadas para el desarrollo del sistema.
- Desarrollar una aplicación móvil E-Commerce para la gestión de ventas.

## **CAPITULO II**

### **MARCO TEÓRICO**

#### **2.1. Antecedentes Investigativos**

El presente proyecto se basa en los siguientes antecedentes investigativos que fueron producto de una revisión tanto en los repositorios digitales como de la Universidad Técnica de Ambato y artículos, de las cuales se pude hacer referencia las siguientes citas:

Diego Mauricio Ramírez Parra, en su tesis “Social Media Marketing como Estrategia de Ventas para la Empresa My Shoes de la Ciudad de Ambato” realizado en la Universidad Técnica De Ambato afirma que: “... My Shoes mantiene informada a sus clientes, pero no de una manera adecuada ya que no utilizan correctamente lo que es páginas web, medios digitales. Cabe recalcar que la empresa si dispone de una página web la misma que no tiene un dominio propio ya que utilizan hostings gratuitas lo cual no ayuda a la imagen de la empresa”[6], de lo cual se concluye que es de suma importancia mantener informado al cliente de sus productos por medios seguros.

Además la empresa My Shoes afirma lo siguiente “... la empresa no difunde adecuadamente la información mediante los medios digitales y no se interesa en generar información atractiva y precisa, la empresa no utiliza una herramienta electrónica adecuada para difundir información de los productos que ofrece la empresa llegando a una calificación de información entre mala y buena indicando que la información electrónica no influye para atraer y retener a los clientes.”[7] de lo siguiente se puede analizar que el uso indebido de herramientas o medios para informar a los clientes puede ser trágico.

La tesis de grado “Los Canales de Distribución y su incidencia en las Ventas de la Empresa de calzado Bettini de la ciudad de Ambato” de Mónica Coque realizado en

la Universidad Técnica de Ambato concluye que: "... la gran mayoría de personas que adquiere este producto, no conocen de la existencia de nuestra marca en el mercado y forman parte de un grupo considerable, de clientes sólidos, con los que cuenta en los actuales momentos la empresa"[8], así dando a entender que es de suma importancia la manera de promocionar la marca a través de los medios digitales.

Luis Alfredo Muñoz en su tesis "Estrategias de Marketing y su incidencia en las ventas de la empresa de calzado Gabriel de la ciudad de Ambato" realizado en la Facultad de Ciencias Administrativas de la Universidad Técnica de Ambato recomienda:" ... Elaborar estrategias promocionales que permitan incrementar las ventas y la fidelización de los clientes, basando principalmente en descuentos.

Para de esta manera beneficiar a la empresa y a los clientes que cancelan sus obligaciones en tiempos no muy extensos "[9] por consiguiente da a entender que implementar nuevas estrategias ayudara a crecer como empresa y la fidelización de clientes.

Adicionalmente, menciona lo siguiente "...Elaborar un plan publicitario el cual debe resaltar las cualidades del producto, lo cual permitirá mejorar las ventas y a su vez lograr un mejor posicionamiento de la marca en el mercado"[9], en vista a ello una aplicación puede ser de gran ayuda para mostrar los productos, como bien dice resaltando sus cualidades y el posicionamiento de la marca.

## **2.2. Fundamentación teórica**

### **2.2.1. Internet**

Es un sistema de red conectado globalmente que utiliza TCP / IP para transmitir datos a través de varios tipos de medios. Internet es una red de intercambios globales, que incluye redes privadas, públicas, empresariales, académicas y gubernamentales, conectadas mediante tecnologías guiadas, inalámbricas y de fibra óptica[10].

Los términos internet y World Wide Web a menudo se usan indistintamente, pero no son exactamente lo mismo; Internet se refiere al sistema de comunicación global, incluido el hardware y la infraestructura, mientras que la web es uno de los servicios que se comunica a través de Internet[10].



### **2.2.2. Lenguaje de programación**

“Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente.”[11].

En la actualidad el mundo de desarrollo de software existe varios lenguajes de programación como: Java, C#, C++, Python, entre otros, los mismo que cumplen con el funcionamiento lógico, físico de un dispositivo electrónico.

### **2.2.3. Metodología de desarrollo**

las metodologías de desarrollo de software son un marco para el trabajo eficiente que se surgió por los años 70, con el objetivo de dar solución a los problemas que surgían en el desarrollo de los proyectos. En esos años, la creación de proyectos no tenía un control apropiado de las actividades del grupo de trabajo, lo que conlleva un producto con deficiencias y problemas o insatisfacción del cliente ya que en la mayoría de los casos recibían un software que no cumple sus necesidades[12].

“Actualmente existen muchas metodologías para el desarrollo de software que son utilizadas dependiendo del sistema a crear, los cuales pueden dividirse en grupos comunes como son: escritorio, móvil y web, siendo este último de los que más impulso ha venido acumulando, pues la necesidad del cliente de ser reconocido por medio del Internet se ha intensificado. ”[12]

### **2.2.4. Aplicación móvil**

Una aplicación móvil es una aplicación de software desarrollada específicamente para su uso en dispositivos informáticos inalámbricos pequeños, como teléfonos inteligentes y tabletas, en lugar de computadoras de escritorio o portátiles. Las aplicaciones móviles están diseñadas teniendo en cuenta las demandas y limitaciones de los dispositivos y también para aprovechar las capacidades especializadas que tienen.

#### **2.2.4.1. Aplicación móvil híbrida**

Una aplicación híbrida es aquella que combina elementos de aplicaciones nativas y web. Las aplicaciones nativas se desarrollan para una plataforma específica y se

instalan en un dispositivo informático. Las aplicaciones web se generalizan para múltiples plataformas y no se instalan localmente, sino que están disponibles en Internet por medio de un navegador. Las aplicaciones híbridas a menudo se mencionan en el contexto de la informática móvil[13].

El desarrollo de aplicaciones móviles híbridas necesita de un entorno de desarrollo o un framework, que básicamente es un marco de trabajo, que se ha desarrollado para solventar una problemática, es un tipo de abstracción en el que código común que provee una funcionalidad genérica puede ser sobre escrito o especializado por el usuario para implantar una funcionalidad específica[14].

Entre los ambientes de desarrollo para las aplicaciones híbridas se encuentra Ionic Framework, mismo que se utilizó para la realización del proyecto de titulación.

#### **2.2.5. Framework de desarrollo**

El término framework se emplea para varios ámbitos del desarrollo de software, en general, se refiere a una estructura compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación.

En otras palabras, un framework es un conjunto de bibliotecas, herramientas y normas que se debe seguir para que los desarrolladores vean una manera más amigable de desarrollar sus aplicaciones. Las aplicaciones pueden ser desarrolladas de manera más eficaz y eficiente si se utiliza un framework adecuado para un proyecto seleccionado, en lugar de tener que volver a inventar la rueda cada vez.[15]

Los objetivos principales que persigue un framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. En la actualidad existen Frameworks para lenguajes como html5, css y JavaScript. [16]

##### **2.2.5.1. Análisis y comparación de aplicaciones móviles híbridas y nativas.**

A la hora de desarrollar una aplicación móvil se analiza aspectos técnicos como si realizarlo con un lenguaje nativo de cada plataforma, o aplicaciones híbridas o aplicaciones generadas, o una aplicación web. Sin embargo, la elección depende de diversos factores, como si se va a reutilizar el código o el coste que se quiera destinar

al desarrollo, Actualmente, las aplicaciones nativas ofrecen la posibilidad de acceder a su máxima capacidad del dispositivo.

Las aplicaciones nativas tienen control total del software y hardware, por tanto, el rendimiento es óptimo, no precisamente estas aplicaciones necesitan hacer uso de Internet, se ejecutan en segundo plano dependiendo de la función que cumple dichas aplicaciones.

Adicionalmente, el esfuerzo que se consigue para desarrollar una aplicación se multiplica debido que para cada plataforma se requiere conocimiento del lenguaje de programación nativo para cada plataforma, por ende, los costos del desarrollo se elevan para llevar a cabo el mantenimiento, actualizaciones y distribuciones de nuevas versiones.

El desarrollo multiplataforma se contrapone al nativo y se centra en el reúso de código. La construcción de aplicaciones web móviles constituye un ejemplo representativo de este enfoque. Sin embargo, las limitaciones derivadas de su ejecución dentro de un navegador, ha motivado a los ingenieros de software a dirigir su atención hacia otro tipo de aplicaciones multiplataforma con el que se obtienen resultados más cercanos a las soluciones nativas. [26].

Estas aplicaciones, diseñadas para correr dentro de un navegador, se desarrollan con tecnología web (HTML, CSS y JavaScript) y cuentan con una serie de características favorables: no necesitan adecuarse a ningún entorno operativo, son independientes de la plataforma y su puesta en marcha es rápida y sencilla [25].

Por contrapartida, sus tiempos de respuesta decaen afectados por la interacción cliente-servidor, al mismo tiempo que resultan ser menos atractivas que las aplicaciones nativas ya que no se encuentran instaladas en el dispositivo. Además, las restricciones de seguridad impuestas a la ejecución de código por medio de un navegador limitan a estas aplicaciones que no pueden acceder a todas las capacidades del dispositivo [25].

## Comparativa entre aplicaciones móviles híbridas y nativas

	<b>APLICACIONES HÍBRIDAS</b>	<b>APLICACIONES NATIVAS</b>
<b>CURVA DE APRENDIZAJE</b>	El conocimiento se basa en tecnologías web como HTML, CSS, JavaScript. Curva de aprendizaje rápido.	Es necesario aprender a desarrollar en cada una de las plataformas en las que se va a trabajar, tanto en lo referente a su lenguaje de programación como a las API's existentes. Curva de aprendizaje lento. Requiere más esfuerzo.
<b>DISTRIBUCIÓN A LAS RESPECTIVAS PLATAFORMAS</b>	Facilidad y versatilidad de generar y exportar a las distintas plataformas.	Se requiere de programar para cada plataforma con el lenguaje de programación para cada una. Consumo de tiempo al realizar la misma lógica para otra plataforma.
<b>COSTE DEL DESARROLLO</b>	Coste menor, al requerir solo un desarrollo y ser este más simple	Coste más elevado, hay que desarrollar más veces y en lenguajes más complejos
<b>FACILIDAD DE ENCONTRAR DESARROLLADORES</b>	Muy sencillo, y una sola persona puede exportar a múltiples plataformas	Algo menos sencillo, requiere expertos para cada plataforma por lo general.
<b>RENDIMIENTO</b>	Muy bueno, excepto quizá para aplicaciones muy exigentes, juegos, 3D.	Óptimo
<b>ACCESO AL DISPOSITIVO</b>	Muy amplio, aunque no completo	Completo
<b>ASPECTO VISUAL</b>	Muy bueno, simulando comportamientos con HTML5 y CSS3, aunque puede no ser óptimo	Completo

Tabla 1. Comparativa entre aplicaciones móviles híbridas y nativas.

Elaborado por: Diego Chicaiza.

Las aplicaciones móviles híbridas son una composición de tecnologías web como HTML, CSS y JavaScript, que no son aplicaciones móviles nativas, ya que se ejecutan a través de un WebView. Por otra parte, tampoco son aplicaciones web ya que estas

aplicaciones son empaquetas para la distribución y subirlas a las respectivas tiendas como los Google Play para Android, AppStore para iOS. Cabe mencionar que dependiendo del objetivo que tiene una aplicación se decide que metodologías, herramientas, si es aplicación híbrida o nativa, entre otros aspectos.

En la

Tabla 1, la se puede analizar que las aplicaciones híbridas son una buena opción para el desarrollo del proyecto, ya que con respecto al coste que lleva implementar un sistema para distintas plataformas.

Para el desarrollo del proyecto de investigación se decidió realizar una aplicación móvil híbrida, pues permite centralizar toda la lógica de negocio y código bajo un mismo esquema general para luego generar la aplicación para la plataforma que lo requiera.

### **Ionic Framework**

Ionic es un framework de desarrollado por Drifty Co cuyo lanzamiento se dio en 2013 y bajo licencia MIT. Se diseñó sobre AngularJS y Apache Cordova que proporcionan un conjunto de herramientas y servicios que permite desarrollar aplicaciones móviles híbridas mediante tecnologías web. Estas aplicaciones son empaquetadas mediante apache Cordova para ser distribuido en las respectivas tiendas de cada plataforma.

Además, incorpora diversos mecanismos que simulan la interacción nativa con el sistema operativo. Es un Kit de Desarrollo de Software (SDK) de código abierto que proporciona herramientas y servicios para desarrollar aplicaciones móviles híbridas usando tecnología web como HTML5, CSS, JavaScript[17]. La filosofía “escribe una vez, y ejecuta en donde sea” significa que el mismo código se podrá ejecutar en las plataformas soportadas por Ionic[18].

Ionic adapta los elementos a cada una de las plataformas a la hora de ejecutarlas para dar la impresión de que fuese una aplicación nativa. Grandes marcas, como Baskin Robbins, Diesel, MacDonald’s han elegido a Ionic para el desarrollo de sus aplicaciones[18].

### **React Native**

React Native nació en 2015, creado por Facebook está basado en framework de React, pero presenta elementos de IU nativos en tiempo de ejecución, lo que le permite crear aplicaciones de iOS y Android. Utiliza JSX misma que es una extensión de JavaScript. React Native no permite utilizar el mismo código para todas las plataformas móviles que soporta la misma.

Toda la familia de aplicaciones de Facebook está desarrollada con React. La colección de aplicaciones representativas desarrollados con React Native son por ejemplo Instagram, Facebook Analytics, Bloomberg, Uber, Pinterest, etc.

	<b>React Native</b>	<b>Ionic Framework</b>
Desarrollo/ código	Javascript con Java, Swift, Objective C	Html, css, JavaScript con Typescript
Reutilización de código	Lógica de negocio compartida con diferentes códigos de UI	Una base de código, el código de interfaz de usuario permanece igual
Plataformas de destino	Aplicaciones móviles nativas para iOS y Android	Aplicaciones híbridas para iOS, Android, aplicaciones de escritorio, aplicaciones web progresivas, y en cualquier otro lugar donde se ejecute la web
Interfaz UI	Los elementos de la interfaz de usuario son específicos de la plataforma de destino y no se comparten.	Elementos de la interfaz de usuario web que se comparten en cualquier plataforma, de acuerdo con la apariencia y el estilo nativos de donde se implementan.
Filosofía	Aprende una vez, escribe donde sea.	Escribe una vez, ejecútalas en donde sea.
Rendimiento	Buen rendimiento	Mediano ya que se usa un WebView
Pruebas	Las pruebas se realizan en emuladores o en dispositivos reales	Las pruebas se pueden realizar en el navegador sin necesidad de un emulador o un dispositivo real

Tabla 2. Comparativa de Frameworks Ionic y React Native  
Elaborado por: Diego Chicaiza.

Por medio de la comparativa que se muestra en la Tabla 2 se decide por Ionic Framework ya que la función que va a cumplir la aplicación es de mostrar productos para poder adquirirlos. Como ventaja Ionic en cuestión de interfaz de usuario comparten el mismo código en todas las plataformas. Para acceder a la parte nativa de un dispositivo Ionic trabaja conjuntamente con Cordova que añade los plugins.

**Cordova:** En el año de 2009 una empresa canadiense llamada Nitobi creó un Framework llamado PhoneGap. La idea era crear aplicaciones móviles con tecnologías web como lo es HTML, CSS y dotándolas con una capa de JavaScript que permita acceder a las funciones nativas de los dispositivos móviles.

La idea cumple el objetivo propuesto, por lo cual la empresa ve que el proyecto funcionaría mejor con una comunidad grande que ayude a mejorar el proyecto PhoneGap. Así que por septiembre del 2011 deciden donar el código fuente del proyecto a la Fundación de Apache, PhoneGap se convierte en un proyecto Open Source con el soporte a la cual se donó.

Por otro lado, Adobe tras la caída de Flash, en octubre del 2011 decide comprar a la empresa Nitobi con todos sus empleados, productos y la marca PhoneGap. Llegando a un acuerdo de que el código se done a la Fundación de Apache.

El proyecto siguió en marcha, no obstante, PhoneGap al ser marca registrada por Adobe, para evitar problemas legales con respecto al crecimiento que puede tener, deciden cambiar el nombre del proyecto a Apache Cordova.

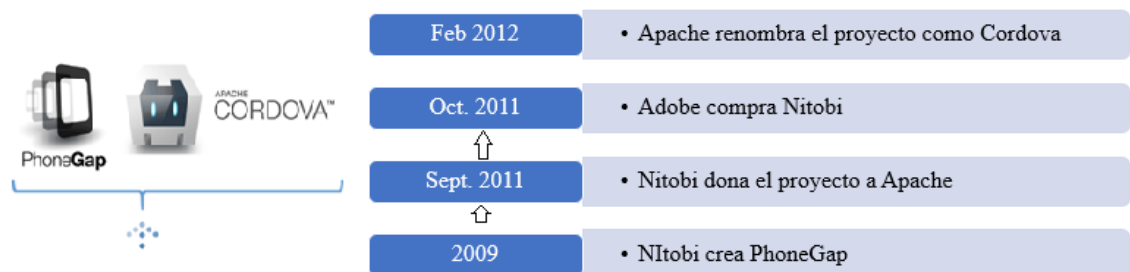


Ilustración 1. Historia de Apache Cordova  
Fuente: [19]

<b>Funcionalidad</b>	<b>Android</b>	<b>iPhone 3GS o más nuevo</b>	<b>Windows Phone</b>
Acelerómetro	Sí	Sí	Sí
Cámara	Sí	Sí	Sí
Brújula	Sí	Sí	Sí
Contactos	Sí	Sí	Sí
Archivos	Sí	Sí	Sí
Geolocalización	Sí	Sí	Sí
Media	Sí	Sí	Sí
Red	Sí	Sí	Sí
Notificaciones (alertas, sonido, vibración)	Sí	Sí	Sí

Almacenamiento	Sí	Sí	Sí
----------------	----	----	----

Tabla 3. Funcionalidades principales que provee Apache Cordova  
Elaborado por: Diego Chicaiza

## Angular

Angular fue lanzado en 2009 por Google y es un potente Framework que adicionalmente es open. La vinculación bidireccional también conocida como data binding es una característica innovadora intuitiva de Angular que permite la actualización en tiempo real de un sitio web desde diferentes dispositivos. Cualquier vista se cambia de forma simultánea en la aplicación móvil y el sitio web dependiendo de la implantación.

Actualmente, Angular se encuentra en la versión 7 estable, comúnmente se le conoce como Angular 2, este framework utiliza Typescript. Además, se basa en Componentes, mismas que pueden tener propiedades y métodos cuyas principales funciones son para hacer uso del binding de los datos o también conocido con enlace de datos.

### 2.2.6. Tecnologías web

**HTML:** Es el lenguaje de maquetado que se emplea para el desarrollo de páginas de internet. Está compuesto por una serie de etiquetas que el navegador interpreta y da forma en la pantalla. HTML (HyperText Markup Language) dispone de etiquetas para imágenes, hipervínculos que nos permiten dirigirnos a otras páginas, saltos de línea, listas, tablas, etc..[15]

**JavaScript:** JavaScript es un lenguaje de programación dinámico que en la actualidad se utiliza tanto en el cliente como en el servidor, además es ligero y el uso del mismo se ha ido popularizando para tener páginas web más amigables con los usuarios.

JavaScript fue conocido por primera vez como LiveScript, pero Netscape cambió su nombre a JavaScript, posiblemente debido a la emoción generada por Java. JavaScript hizo su primera aparición en Netscape 2.0 en 1995 con el nombre LiveScript. El propósito general del lenguaje se ha incorporado en Netscape, Internet Explorer, Google Chrome, Mozilla Firefox y otros navegadores web[20].



JavaScript fue declarado como estándar del European Computer Manufacturers Association (ECMA) en 1997, y poco después también fue estandarizado por ISO. Sin embargo, la estructura de objetos que implementaban los diferentes navegadores (Netscape y Explorer en aquellos momentos) no se ajustaba al estándar, lo que provocaba numerosos problemas de compatibilidad.

Para solventarlos el W3C publicó un nuevo modelo de objetos, DOM (Document Object Model), que incorporan la mayoría de los navegadores actuales[21].

**CSS:** Hojas de Estilo en Cascada, es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos.[14]

### **2.2.7. Sistemas operativos para dispositivos móviles.**

#### **IOS**

Es uno de los sistemas operativos móviles más populares desarrollado y creado por Apple Inc. Un dispositivo iOS es un dispositivo electrónico que se ejecuta en iOS. Los dispositivos Apple iOS incluyen: iPad, iPod Touch y iPhone. iOS es el segundo SO móvil más popular después de Android. Con los años, los dispositivos Android e iOS han estado compitiendo tanto por una mayor participación en el mercado[22].

#### **Android**

Es un sistema operativo móvil que se encuentra en una variedad de dispositivos modernos, siendo los más populares los teléfonos inteligentes. Además de eso, también encontrarás Android en tabletas, cajas de transmisión de TV y otros dispositivos portátiles. Android es básicamente una pieza de software que permite que tu hardware funcione[23].

El sistema operativo Android te da acceso a las aplicaciones, incluidas muchas de las creaciones propias de Google. Estos le permiten buscar información en la web, reproducir música y videos, verificar su ubicación en un mapa, tomar fotos con la cámara de su dispositivo y mucho más[23].

### **2.2.8. Comercio Electrónico (E-Commerce)**

El comercio electrónico (e-commerce) es la comercialización, compra y venta de mercancías o servicios a través de Internet. Abarca todo el alcance de las ventas de productos y servicios en línea de principio a fin. El concepto de comercio electrónico se relaciona con transacciones comerciales o financieras que facilitan los pagos electrónicos de artículos comprados en tiendas en línea y proveedores de servicios.

El comercio electrónico cubre una amplia gama de actividades comerciales, desde contenido digital utilizado para el consumo en línea hasta pedidos convencionales de productos en línea. La banca en línea es otra forma de e-commerce. Las transacciones de comercio electrónico se llevan a cabo entre empresas, consumidores, gobierno, empleados, consumidores y empresas[24].

### **2.2.9. Pago por medio de Internet**

“Un servicio de pago online es un servicio que ofrecen algunas empresas para gestionar el envío y recibo de pagos en Internet. Este tipo de servicios ofrecen una intermediación entre el cliente y el vendedor, de tal forma que el primero nunca tiene que compartir información privada (número de tarjeta, CVC, fecha de caducidad de la tarjeta, etc.) con el segundo.

Los servicios de pago online son una alternativa a los Terminales de Punto de Ventas virtuales y son mucho más inmediatos de conseguir puesto que no requieren de un proceso de obtención tan largo como las primeras, donde existe un proceso no muy corto (puede durar hasta 12 semanas) de gestión por parte del banco.”[25]

### **2.2.10. Base de datos**

#### **2.2.10.1. Base de datos relacional**

Una base de datos relacional es una base de datos en donde todos los datos visibles al usuario están organizados estrictamente como tablas de valores, y en donde todas las operaciones de la base de datos operan sobre estas tablas. Estas bases de datos son presentadas a los usuarios como una colección de relaciones normalizadas de diversos grados que varían con el tiempo.

El modelo relacional representa un sistema de bases de datos en un nivel de abstracción un tanto alejado de los detalles de la máquina subyacente. El modelo relacional puede considerarse como un lenguaje de programación más bien abstracto, orientado de

manera específica hacia las aplicaciones de bases de datos. En términos tradicionales una relación se asemeja a un archivo, una tupla a un registro, y un atributo a un campo[26].

#### **2.2.10.2. Bases de datos NoSQL**

La respuesta a la necesidad de gestionar volúmenes masivos de información surge de la base de datos NoSQL, término acuñado a finales de los 90 y que engloba todas las tecnologías de almacenamiento estructurado que no cumplen el esquema relacional. La cantidad de información manejada por comunidades, redes sociales, buscadores, y muchos otros proyectos en el ámbito de la Web 2.0 es abrumadora, lo que ha hecho que surjan nuevas arquitecturas de almacenamiento de información, que deben ser de alto rendimiento, escalables y distribuidas.[26]

Las bases de datos NoSQL surgen como una solución a los constantes requerimientos de procesamiento y análisis a gran escala de enormes cantidades de datos, y para los cuales los sistemas tradicionales de base de datos son insuficientes.[26]

#### **Firestore**

Firestore es una plataforma BaaS (Backend as a Service), desarrollada por Google que facilita el desarrollo de apps, proporcionando un servidor Backend para las aplicaciones. Además, el mismo Backend puede ser utilizado de forma común en diversas plataformas: Android, IOS y web[27]

Firestore proporciona una solución eficaz frente no solo a problemas de desarrollo, sino también de escalabilidad a medida que la base de usuarios de la aplicación crece, ya que los servidores son proporcionados por Google. Entre sus funcionalidades se encuentra un servicio de autenticación, base de datos en tiempo real, almacenamiento de archivos, solución de errores, funciones Backend, testeo, y medida de estadísticas recogidas de los usuarios[27]

#### **Funciones y servicios de Firestore**

**Base de datos en tiempo real:** Firestore provee una base de datos NoSQL que almacena datos y los sincroniza en tiempo real. Es una gran ventaja ya que para los datos que cambian en la base de datos de Firestore se actualizan en los clientes, es decir

que, si hay alguna novedad como, por ejemplo: descuentos por fechas, promociones, artículos nuevos, entre otros aspectos.

Firebase ofrece soluciones de base de datos en la nube para clientes que requieren de sincronización de datos en tiempo real.

**Realtime Database:** Es la base de datos original de Firebase. Se le considera eficiente y de baja latencia para aplicaciones móviles y sitios web que requieran sincronización entre los clientes en tiempo real

**Cloud Firestore:** Lanzado en el año de 2017 por Google, es la nueva base de datos de Firebase, aprovecha lo mejor de Realtime Database con un nuevo modelo de datos, también cuenta con consultas rápidas y la escala se ajusta mejor que Realtime Database.

**Autenticación:** Firebase provee un método de registro e inicio de sesión con varias opciones como lo es por medio de correo y contraseña, autenticación con redes sociales como Facebook, Google, GitHub, Twitter. La autenticación que provee Firebase es de gran ayuda para los desarrolladores ya que ahorra tiempo en desarrollar los distintos módulos que puede tener la autenticación. Para el proyecto se realiza el registro e inicio de sesión por medio de correo y contraseña, por medio del proveedor de Google.

**Almacenamiento de archivos:** Esta característica se ha utilizado para guardar imágenes que la aplicación utilizara, así como las fotos de perfil del usuario, fotos de los productos a ofrecer.

**Funciones backend:** Firebase permite ejecutar código JavaScript en el servidor. Ciertas funciones que no puede ejecutar en la parte del cliente se las puede ejecutar en el servidor, de tal manera que al producir un cambio los demás están pendientes, como por ejemplo las notificaciones push.

### **Comparativa entre Realtime Database y Cloud Firestore**

<b>Realtime Database</b>	<b>Cloud Firestore</b>
Almacena datos como un gran árbol JSON	Almacena datos en documentos organizados en colecciones
Los datos simples son muy fáciles de almacenar.	Los datos simples son fáciles de almacenar en documentos, que son muy similares a JSON.

Los datos complejos y jerárquicos son más difíciles de organizar a escala.	Los datos complejos y jerárquicos son más fáciles de organizar a escala, con su colección dentro de los documentos. Necesita menos des normalización y compactación de datos.
--	---

Tabla 4. Comparativa entre Realtime Database y Cloud Firestore.

Elaborador por: Diego Chicaiza

Cloud Firestore ofrece más servicios y beneficios que Realtime Database es considerable por tanto se utilizó Cloud Firestore. Sin embargo, Firebase ofrece varios planes, el que se seleccionó para el proyecto es Plan Spark, en la figura siguiente se hace una comparativa de los planes que provee Firebase.

Productos	Plan Spark Límites generosos para aficionados <b>Sin cargo</b>	Plan Flame Fixed pricing for growing apps <b>\$25/month</b>	Plan Blaze Calculate pricing for apps at scale <b>Pago por uso</b> <small>✓ Se incluye el uso gratuito del plan Spark*</small>
Pruebas A/B	Sin cargo		
Analytics	Sin cargo		
App Indexing	Sin cargo		
<b>Authentication</b>			
Phone Auth - US, Canada, and India	10,000 por mes	10,000 por mes	USD 0.01 por verificación
Phone Auth - All other countries	10,000 por mes	10,000 por mes	USD 0.06 por verificación
Other Authentication services	✓	✓	✓
<b>Cloud Firestore</b>			
Datos almacenados	1 GiB total	2.5 GiB total	\$0.18/GiB
Ancho de banda	10GiB/month	20GiB/month	<a href="#">Google Cloud Pricing</a>
Operaciones de escritura de documentos	20,000/día	100,000 por día	USD 0.18 por cada 100,000
Operaciones de lectura de documentos	50,000/día	250,000/día	USD 0.06 por cada 100,000
Operaciones de eliminación de documentos	20,000/día	100,000 por día	USD 0.02 por cada 100,000

Figura 1. Comparativa básica de los planes que provee Firebase

Fuente: [28]

Además, Firebase provee de una calculadora de plan Blaze, el cual hace referencia al pago por uso de los servicios que provee la plataforma de Firebase. Misma se puede observar en la siguiente referencia [28].

### 2.2.11. Visual Studio Code

Visual Studio Code (también conocido como VS Code) es "un editor de código fuente ligero pero potente que se ejecuta en su escritorio y está disponible para Windows, macOS y Linux"[29]. VS Code presenta varias ventajas:

Viene con un **soporte incorporado** para JavaScript, TypeScript, nodeJs (autocompletado, comprobación de sintaxis, depuración, ...).

Tiene un **gran ecosistema** de complementos para admitir otros lenguajes de programación (C, C ++, C #, Python, Java, etc.), incluso puede instalar mapas de teclado de editores de texto como texto sublime, átomo, vim, y otros.

Es **multiplataforma**: significa que siempre que esté en Windows, Mac o Linux, no es necesario aprender otra herramienta u otro mapa de teclas, solo descargar y configurar VSCode y comenzar a trabajar.

### **2.2.12. Controlador de versiones**

El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante. A pesar de que los ejemplos de este libro muestran código fuente como archivos bajo control de versiones, en realidad cualquier tipo de archivo que encuentres en un ordenador puede ponerse bajo control de versiones[30].

#### **Sistemas de control de versiones locales**

Un método de control de versiones usado por mucha gente es copiar los archivos a otro directorio. Este enfoque es muy común porque es muy simple, pero también tremendamente propenso a errores. Es fácil olvidar en qué directorio te encuentras, y guardar accidentalmente en el archivo equivocado o sobrescribir algún archivo no deseado[31].

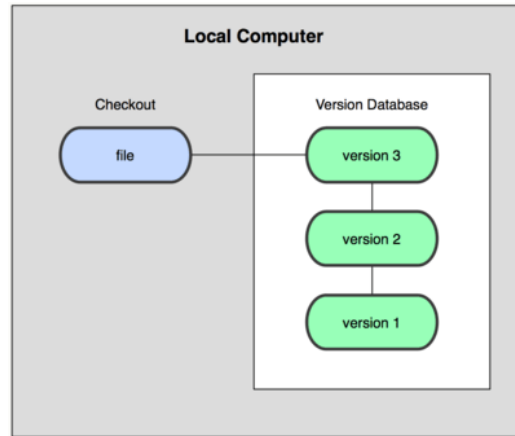


Figura 2. Sistemas de control de versiones locales

### Sistemas de control de versiones centralizados

El siguiente gran problema que se encuentra la gente es que necesitan colaborar con desarrolladores en otros sistemas. Para solventar este problema, se desarrollaron los sistemas de control de versiones centralizados (Centralized Version Control Systems o CVCS en inglés). Estos sistemas, como CVS, Subversion, y Perforce, tienen un único servidor que contiene todos los archivos versionados, y varios clientes que descargan los archivos desde ese lugar central.[30]

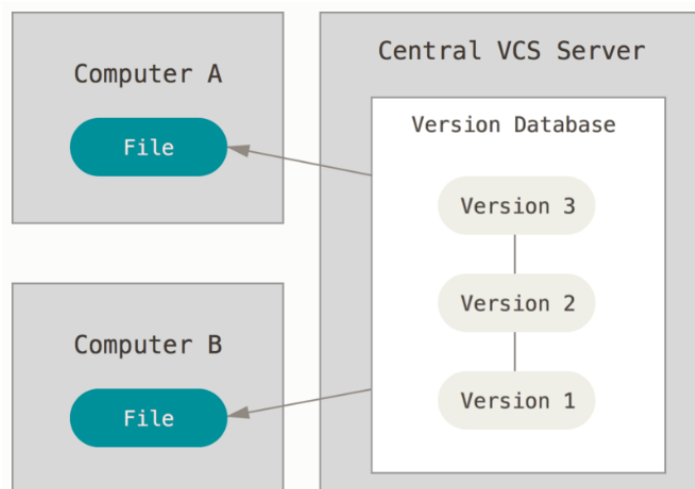


Figura 3. Sistemas de control de versiones centralizados

### Sistemas de control de versiones distribuidos

Es aquí donde entran los sistemas de control de versiones distribuidos (Distributed Version Control Systems o DVCS en inglés). En un DVCS, los

clientes no sólo descargan la última instantánea de los archivos: replican completamente el repositorio. Así, si un servidor muere, y estos sistemas estaban colaborando a través de él, cualquiera de los repositorios de los clientes puede copiarse en el servidor para restaurarlo.[30]

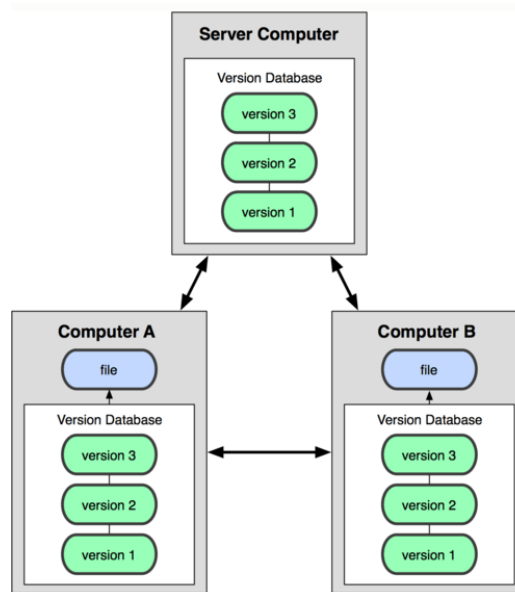


Figura 4. Diagrama de control de versiones distribuido.

## GitLab

Es un servicio web de control de versiones y desarrollo de software colaborativo basado en Git. Además de gestor de repositorios, ofrece alojamiento de wikis y un sistema de seguimiento de errores, todo ello publicado bajo una Licencia de código abierto.

Permite crear repositorios gratuitos de forma pública y privada, permite agregar colaboradores sin límite, por otra parte, GitHub es otra opción, pero a continuación se realiza una comparación entre ambos, cabe mencionar dicha comparativa es con el plan gratuito.

En la Tabla 5 se muestra ciertas características de GitLab y GitHub. GitLab ofrece más opciones en plan gratuito como la integración de colaboradores sin límites otra que es muy importante es la Integración y entrega continuas que se manejan en proyectos netamente de desarrollo con buenas prácticas de programación.



GitLab	GitHub
<b>Crear proyectos privados ilimitados.</b>	
<b>Crear colaboradores ilimitados.</b>	Repositorios públicos ilimitados.
<b>Soporte comunitario</b>	Repositorios privados ilimitados.
<b>CI/CD incorporado</b>	3 colaboradores para repositorios privados.
<b>Consejo de Emisión de Proyectos</b>	Problemas y seguimiento de errores.
<b>ChatOps</b>	Gestión de proyectos.

Tabla 5. Comparativa entre GitLab y GitHub

Elaborador por: Diego Chicaiza

### **GitKraken**

Es una herramienta potente y con una interfaz gráfica multiplataforma para control de versiones git, desarrollado por Electrón. De forma simple se puede realizar un seguimiento a los repositorios que se agreguen para observar, por ejemplo, las ramas, historial de trabajo, crear ramas, commits, push, pull. Esta herramienta se utilizó como alternativa de git bash en donde no se puede apreciar visualmente los cambios que se producen[30].

### **2.2.13. Node JS**

Es un entorno en tiempo de ejecución multiplataforma, de código abierto. que quiere decir que se puede programar en el lado del servidor con JavaScript. Sin embargo, para el proyecto nos interesa NPM (Node Package Manager) que es el gestor de paquetes de Node que será utilizado para instalar librerías[32].

### **2.2.14. Android SDK**

Es un conjunto de herramientas de desarrollo o Software Development Kit, por medio de uno de ellos se puede crear aplicaciones y ejecutar, realizar pruebas por medio de un emulador o un dispositivo Android.

Para el proyecto es necesario tener instalado el SDK de Android para poder realizar las debidas pruebas del proyecto de investigación.

### **2.2.15. GenyMotion**

Es un emulador de Android para ejecutar de forma fluida y rápida las aplicaciones desarrolladas, en comparación a los emuladores de Android Studio son significativamente rápidas. La instalación de un emulador específico es muy fácil de configurar, por ejemplo, seleccionar el nivel de API, versión del sistema operativo, densidad, etc., opciones para modificar o si no se descarga y luego se inicia.

## CAPÍTULO III

### METODOLOGÍA

#### 3.1. Modalidad Básica de la investigación

La investigación se centrará dentro de los siguientes ámbitos:

**Investigación bibliográfica o documentada,** La investigación bibliográfica tendrá como objetivo fortalecer la investigación porque se tomará información de Internet, libros virtuales, tesis de grados, videos, informes, proyectos, revistas, informes, para obtener cuestiones similares al problema a solucionar del presente proyecto de investigación.

**Investigación experimental** para realizar pruebas de comprobación el funcionamiento del sistema que se desarrollara.

**Investigación aplicada,** porque busca la aplicación o utilización de los conocimientos que se adquieren.

**Investigación de campo,** porque el investigador acudirá al lugar en donde se producen los hechos para la obtención de información relacionada con los objetivos del proyecto de investigación. La técnica para utilizar será la entrevista.

### 3.2. Recolección de Información

<b>PREGUNTAS BÁSICAS</b>	<b>EXPLICACIÓN</b>
<b>1. ¿Para qué?</b>	Para alcanzar los objetivos de la investigación
<b>2. ¿De qué personas u objetos?</b>	De la empresa de calzado “CREACIONES ANABEL”
<b>3. ¿Quién?</b>	Investigador: Chicaiza Palate Diego Baltazar
<b>4. ¿Sobre qué aspectos?</b>	Indicadores: Dispositivos móviles, Framework de desarrollo, Gestión de ventas
<b>5. ¿Cuándo?</b>	La presenta investigación se realizará en el periodo: octubre 2018 – febrero 2020.
<b>6. ¿Dónde?</b>	Ciudad de Ambato, provincia Tungurahua
<b>7. ¿Cuántas veces?</b>	Tres veces
<b>8. ¿Qué técnicas de recolección?</b>	Entrevista, cuestionario estructurado.
<b>9. ¿Con qué?</b>	Cuestionario estructurado
<b>10. ¿En qué situación?</b>	En condiciones normales

Tabla 6. Recolección de Información

### 3.3. Procesamiento y Análisis de Datos

La técnica que se necesitara para el desarrollo de la aplicación híbrida, en la toma de requisitos es la entrevista, ya que es necesario saber el funcionamiento de cada servicio que presta la empresa y que desean implementar en dicha aplicación. Se va a recolectar la información sobre como la empresa realiza su venta de los respectivos productos y como sigue ganando clientes en el mercado.

Todo de acuerdo, a la metodología seleccionada, para la toma de requisitos de la aplicación, realizada en dicha empresa.

Revisión crítica de la información recogida; es decir, limpieza de la información defectuosa: contradictoria, incompleta, no pertinente, etc.

Repetición de la recolección, en ciertos casos individuales, para corregir fallas de contestación.

### **3.4. Desarrollo del Proyecto**

Para el presente proyecto de investigación se seleccionó la metodología XP, además se seleccionará herramientas para el respectivo desarrollo de la aplicación, a continuación, los procesos y la metodología se cumplirá en el orden mencionado.

#### 1. Fase 1: Exploración

1.1. Levantamiento de información y análisis de resultados.

1.2. Comparativa de metodologías enfocado al desarrollo de app móviles.

1.3. Análisis sobre la situación del proceso de ventas de la empresa y sus políticas.

#### 2. Fase: Planificación del proyecto

2.1. Recolección de historias de usuario

2.2. Estimación requerida para el desarrollo de proyecto

#### 3. Fase 3: Iteraciones

3.1. Definir plan de entregas del desarrollo.

#### 4. Fase 4: Producción

4.1. Plan de iteraciones especificadas.

4.2. Codificación

#### 5. Fase 5: Pruebas

5.1. Realización de pruebas

5.2. Implementación del aplicativo móvil.

## **CAPÍTULO IV**

### **DESARROLLO DE LA PROPUESTA**

#### **4.1. Metodologías de desarrollo de la ingeniería de software.**

##### **4.1.1. Comparativa de metodologías de desarrollo de software**

En la década de los 50 el desarrollo de sistemas estaba a cargo de programadores más enfocados en la tarea de codificar, que en la de comprender y recoger las necesidades de los usuarios, que muy a menudo no quedaban satisfechos con el resultado final, es decir no era un software de calidad. La calidad no solo se refiere a la satisfacción del cliente, sino que también puede referirse a su velocidad, estabilidad, flexibilidad, seguridad, usabilidad, escalabilidad, entre muchos otros atributos[33].

##### **Metodologías de desarrollo tradicional**

Las metodologías de desarrollo tradicionales son también llamados modelos de proceso prescriptivo, y fueron planteadas para poner orden en el desorden del desarrollo de software que existía cuando se empezó a crear masivamente. Según la historia menciona que los modelos tradicionales fueron propuestas y dieron en cierta parte una estructura útil para el trabajo de la Ingeniería de software y hasta la actualidad constituyen un mapa razonable para los equipos de desarrollo[34].

Adicionalmente, las metodologías de desarrollo tradicional conciben al proyecto como algo grande con estructura definida, gran cantidad de especificaciones limitadas. El proceso es de manera secuencial, es decir en una dirección y sin marcha atrás, considerado como un proceso rígido que no se puede cambiar ya que los requerimientos de cliente o usuario final son acordados de una vez y para todo el proyecto, consumiendo enormes cantidades de tiempo para la planeación detallada previa y poca comunicación con el cliente en la etapa de desarrollo[35].

## **Metodologías de desarrollo ágil**

El mundo empresarial maneja en un entorno global que cambia apresuradamente; por ende, se debe responder a las nuevas necesidades y oportunidades del mercado, ya que el software es vital en los procesos empresariales, por consiguiente, se enfoca en desarrollar soluciones informáticas de manera oportuna y ágil para lograr la satisfacción del cliente, entregando productos según las necesidades del cliente.

Las metodologías de desarrollo ágil constituyen un conjunto de buenas prácticas de desarrollo, buscando la eficiencia y eficacia para la realización de un producto o servicio. Por lo general, estas metodologías son iterativas e incrementales, presentan alta comunicación con el cliente por lo que son más flexibles a cambios de último momento, lo cual hace que sean adaptativas[34].

Las metodologías ágiles exteriorizan como principal peculiaridad la flexibilidad, los proyectos en desarrollo se subdividen en proyectos más pequeños, incluye una comunicación constante con el usuario, son altamente colaborativos y es mucho más adaptable a los cambios. De hecho, el cambio de requerimientos por parte del cliente es una característica especial, así como también las entregas, revisión y retroalimentación constante[34].

## **Manifiesto por el Desarrollo Ágil de Software**

Las metodologías ágiles resuelven los problemas surgidos, posteriormente, a la masificación del uso del computador personal, dado que las expectativas y necesidades por parte de los usuarios se hicieron más urgentes y frecuentes. Fue así como a comienzo de los 90 surgieron propuestas metodológicas para lograr resultados más rápidos en el desarrollo de software sin disminuir su calidad.[36]

Después de casi una década de esfuerzos aislados, en febrero de 2001 en Utah-EEUU, se reunieron 17 empresarios de la industria del software y como resultado del debate respecto a las metodologías, principios y valores que deben regir el desarrollo de software de buena calidad, en tiempos cortos y flexible a los cambios, se aceptó el término ágil para hacer referencia a nuevos enfoques metodológicos en el desarrollo de software. [36]

El manifiesto ágil se respalda y hace énfasis en cuatro valores señalados que debe soportar el desarrollo de software [37].

- Individuos e interacciones sobre procesos y herramientas.
- El software que funciona, frente a la documentación exhaustiva.
- La colaboración con el cliente, por encima de la negociación contractual.
- La respuesta al cambio, por encima del seguimiento de un plan.

#### 4.1.2. Diferencias entre metodologías ágiles y no ágiles

En la siguiente tabla (Tabla 7) se puede observar ciertas diferencias entre las metodologías ágiles y las tradicionales, y de acuerdo el análisis se procederá a elegir la más idónea para realizar el proyecto.

<b>Metodologías ágiles</b>	<b>Metodologías tradicionales</b>
Basadas en heurísticas provenientes de prácticas de producción de código.	Basada en normas provenientes de estándares seguidos por el entorno de desarrollo.
El cliente interactúa con el equipo de desarrollo en todo el proyecto.	El cliente solo puede interactuar con el equipo de desarrollo con previa citación y si el equipo este de acuerdo.
Los equipos de trabajo son pequeños, regularmente son menos de 10 integrantes.	Los grupos siempre son grandes.
Hay más flexibilidad ante los cambios repentinos que surgen y se ejecutan con rapidez.	Los cambios o bien se realizan lentamente o según el contrato pueden restringir ciertos cambios.
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
Orientado a proyectos pequeños y medianos.	Proyecto de cualquier tamaño
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos.

Tabla 7. Diferencias entre metodologías ágiles y no ágiles

Elaborado por: Diego Chicaiza.



## Proceso de metodología ágil y metodología tradicional.



Figura 5. Proceso de la metodología ágil y metodología tradicional.

Fuente:[18]

Mediante la necesidad que tiene el proyecto se procedió a seleccionar la metodología ágil, debido que en el transcurso del proyecto pueden surgir cambios por parte del cliente, además el cliente estará informado de cómo va avanzando el proyecto y si cumple la necesidad a satisfacer.

A continuación, se realizó la comparación de distintas metodologías de desarrollo ágil mismas que son reconocidas en el mundo del desarrollo de software, como lo son SCRUM y Extremme Programming (XP).

### Metodologías de desarrollo ágil

#### 4.1.3. SCRUM

Esta metodología se estructuró para desarrollar productos en Japón, no se trata de un concepto nuevo, sino que ya en 1987 Ikujiro Nonaka y Hirotaka Takeuchi (1986) acuñaron este término, una estrategia utilizada en rugby en la que todos los integrantes del equipo actúan juntos para avanzar la pelota y ganar el partido, para denominar un nuevo tipo de proceso de desarrollo de productos[38].

Esta metodología es un proceso para la gestión y control del producto que trata de eliminar la complejidad en áreas específicas que atribuye a la construcción de software con el fin de satisfacer las necesidades de las organizaciones[38].

En Scrum se efectúan entregas parciales y regulares del producto final, prevalecidas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está

especialmente indicado para proyectos en entornos complejos, donde se requiere conseguir resultados rápidos, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

Esta metodología tiene una estrecha relación con el cliente durante todo el proyecto, dando como características.

- Predisposición y respuesta al cambio.
- Desarrollo iterativo e incremental con entregas frecuentes de funcionalidad del proyecto.
- Comunicación verbal directa.
- Simplicidad, sólo los artefactos necesarios.
- Motivación, compromiso y responsabilidad del equipo para la autogestión y auto organización.

### **Fases de la metodología SCRUM**

Esta reunión sucede una única vez al inicio del proyecto. Se reúne el grupo de trabajo con el representante del cliente. Se determinan los requisitos iniciales y la visión del producto desde el punto de vista del negocio. Se consolida esta información en el Product Backlog.[39]

### **Roles de SCRUM**

El equipo Scrum está formado por los siguientes roles:

**Scrum Master:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner para maximizar el ROI.

**Product Owner (PO):** Representante de los accionistas y clientes que usan el software. Se enfoca en la parte de negocio, es responsable del ROI del proyecto (entregar un valor superior al dinero invertido). Traspone la visión del proyecto al equipo, formaliza las prestaciones en historias a incorporar en el Product Backlog y las Re prioriza de forma regular.

**Development Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.

### El proceso SCRUM

**Product Backlog:** Su desarrollo parte de la visión del propietario del producto, la cual debe ser real, comprendida y compartida por parte de todo el equipo. Es un inventario de funcionalidades, mejoras, tecnología y corrección de errores que deben incorporarse al producto a través de las sucesivas iteraciones del desarrollo. En él se documenta todo lo que implique un trabajo. [39]

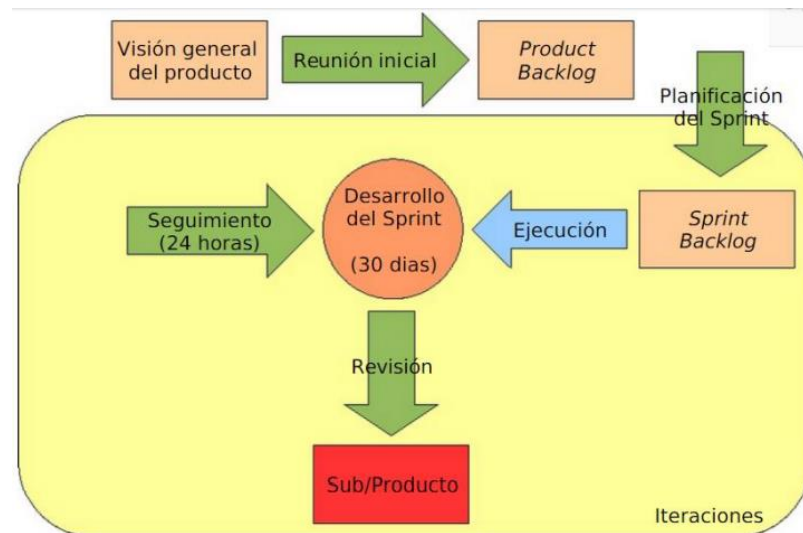


Figura 6. Ciclo de vida de Scrum

**Planificación del Sprint:** Es gestionada por el Scrum Manager. Se realiza en dos etapas. En total, su duración no deberá exceder un día de trabajo. La primera etapa dura máximo 4 horas y en ella se definen cuales funcionalidades del Product Backlog se implementarán en la siguiente iteración. La segunda se descompone las funcionalidades elegidas en las tareas necesarias para implementar el incremento del producto.

Cada sprint, a excepción del primero, deberá generar un producto probado y funcional de interés para el cliente. Se auto asignan las tareas a un integrante del equipo y se estiman los recursos necesarios para su desarrollo. El producto resultante de esta planificación es el Sprint Backlog.[39]

**Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo.

**Sprint Backlog:** Lista de las tareas necesarias para llevar a cabo las historias del sprint.

**Revisión del Sprint:** Al final de cada sprint se realiza una reunión en la que el equipo de trabajo presenta al propietario del producto y demás involucrados el incremento construido durante el sprint. Su duración es de máximo 4 horas. Su preparación es de máximo 1 hora. Le permite al propietario del producto conocer de primera mano el estado actual del desarrollo del proyecto.

El producto expuesto debe estar en su etapa final: terminado, probado y operando en el entorno del cliente (incremento). Puede incluir también documentación de usuario o técnica según se haya pactado.

Al final de la reunión se interroga individualmente a todos los asistentes para recabar impresiones, sugerencias de cambio y mejora, y su relevancia. El propietario del producto trata con los asistentes y con el equipo las posibles modificaciones en el Product Backlog. Se convoca para la reunión de planeación del próximo sprint. [39]

### **Retrospectiva del Sprint**

En Scrum, la Retrospectiva del Sprint es una reunión facilitada por el Scrum Master en la cual los Miembros del equipo de Scrum discuten el Sprint que acaba de finalizar, y determinan qué podría cambiarse en el próximo Sprint para que sea más productivo y mejor. La Revisión del Sprint se focaliza en "Qué" construye equipo, mientras que la Retrospectiva se centra en "Cómo" están construyendo el sistema. [39]

## Proceso de la metodología SCRUM

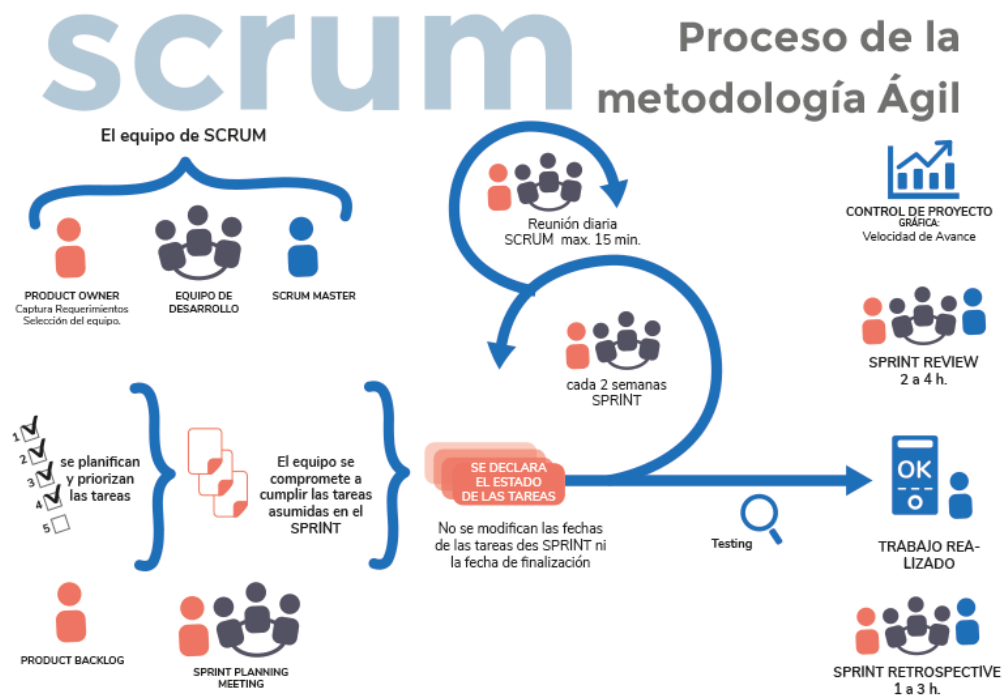


Figura 7. Proceso de la metodología ágil

Fuente: [40]

### 4.1.4. EXTREME PROGRAMING

XP es una metodología ágil enfocada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios.

XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico[39], se basa en lo siguiente:

XP es implementado adecuadamente para proyectos con requisitos imprecisos y requisitos muy cambiantes, y donde puede existir un alto riesgo en la parte técnica.

XP tiene como principio la realimentación continua entre el cliente y el equipo de desarrollo, partiendo de una comunicación fluida entre todos los participantes, manejo de simplicidad en las soluciones implementadas y coraje para enfrentar los cambios.

XP trabaja y estructura una serie de valores y principios que deben tenerse en cuenta y practicarlos durante el tiempo de desarrollo que dure el proyecto. XP se considera una disciplina, la cual está sostenida por valores y principios propios de las metodologías[38].

Los valores de XP, más que una metodología es una disciplina, la cual sostiene los principios y valores propios de las metodologías ágiles. se presentan los valores y el papel que lleva cada uno.

- La comunicación: Es importante que exista un ambiente de comunicación y colaboración en el equipo de desarrollo, así mismo la interacción con el cliente. La interacción con el cliente es estrecha y es considerado parte del equipo de desarrollo.
- La simplicidad: Se aplica desde los diseños sencillos donde lo primordial es la funcionalidad que requiere el cliente, así mismo la simplicidad del código mediante la refactorización y buenas prácticas de programación.
- La retroalimentación: se presenta al comienzo del proyecto, guía y da forma al proyecto. Desde dos puntos de vista, por parte del equipo de desarrollo con el objetivo de entregar información sobre la evolución del sistema, y del cliente al equipo con los aportes en el proyecto.
- El coraje: El equipo de desarrollo ante los continuos cambios que se presentan el transcurso del proyecto deben estar preparados para el cambio.

### **Roles XP**

En la metodología XP se definen los siguientes roles con su descripción respectiva.[41]

**Programador:** Es el encargado de codificar las pruebas unitarias y solución de las historias de usuarios.

**Cliente:** Se encarga de generar y priorizar las historias de usuarios, diseña las pruebas de aceptación y siempre debe estar disponible para cualquier interrogante.

**Encargado de pruebas (Tester):** Ayuda a diseñar las pruebas de aceptación, a ejecutar y automatizar las pruebas de aceptación e integración y difunde los resultados al equipo de desarrollo.

**Encargado de seguimiento (Tracker):** Se encarga de realizar el feedback de todo el proceso XP, evaluando las estimaciones y resultados generados en el proceso.

### **Fases de la metodología XP.**

#### **Exploración**

- El cliente define a grandes rasgos las historias de usuario primordiales

#### **Planificación de entrega**

- Los desarrolladores establecen prioridad y estimación de esfuerzo de cada historia de usuario

#### **Iteraciones**

- Incluye varias iteraciones sobre el sistema antes de ser entregado

#### **Producción**

- Requiere de pruebas adicionales y revisión de rendimiento antes de ser entregado al cliente

#### **Mantenimiento**

- El sistema debe seguir funcionando en producción mientras se desarrolla nuevas iteraciones

#### **Muerte del proyecto**

- Cuando el cliente no tiene más historias que añadir.

Figura 8. Fase de la metodología de XP

Desarrollado por: Diego Chicaiza.

### **Proceso**

XP inicia con una o varias reuniones con el cliente, en las cuales se da claridad a la necesidad puntual del mismo a través de las historias de usuario, donde se tiene presente[38].

- Reuniones continuas y comunicación efectiva.
- Los requisitos pueden (y van a) cambiar.
- Prioridades en los requisitos.
- Grupo pequeño y muy integrado (máximo 12 personas).
- Equipo con formación elevada y capacidad de aprender.
- Metodología basada en prueba y error.

- La fundamentación de los Valores y Prácticas.
- El equipo de trabajo tiene ideas generales de la aplicación a implementar.

#### 4.1.5. Análisis comparativo de las metodologías

El análisis a realizarse es una comparación de las principales características, adicionalmente de sus ventajas y desventajas de algunos puntos fundamentales tanto de la Metodología SCRUM como la de Extreme Programming y considerar cuando usarla. A través de esta comparación se podrá analizar cuál de ellas es la metodología que se adapta al proyecto de investigación.

	<b>SCRUM</b>	<b>EXTREMME PROGRAMMING</b>
<b>DIFERENCIAS</b>	Las iteraciones de entrega son de dos a cuatro semanas y se conocen como sprint.	Las iteraciones de entrega son de una a tres semanas.
	Al finalizar un sprint, las tareas que se han realizado del Sprint y en las que el dueño del producto ha mostrado su conformidad ya no se vuelven a tocar en ningún momento.	Las tareas que se van terminando en las diferentes entregas al cliente son susceptibles a modificaciones durante el transcurso de todo el proyecto.
	Cada miembro del equipo trabaja individualmente.	Los miembros programan en parejas.
	El equipo de trabajo puede modificar el orden del desarrollo de las tareas.	El equipo de trabajo sigue estrictamente el orden de prioridad de las tareas marcadas por el cliente.
	Está más enfocado en la administración del proyecto.	Se centra en la programación o creación del producto.
<b>CARACTERÍSTICAS</b>	<p>Enfatiza valores y prácticas de gestión, sin pronunciarse sobre requerimientos, prácticas de desarrollo, implementación y demás cuestiones técnicas.</p> <p>Equipos auto dirigidos y auto organizados.</p> <p>Desarrollo de software iterativos incrementales que se basan en prácticas ágiles.</p>	<p>Metodología basada en prueba y error para obtener un software que funcione correctamente.</p> <p>Fundamentada en buenas prácticas de programación.</p> <p>Expresada en forma de 12 practicas.</p> <p>Reduce el coste del cambio en todas las etapas del ciclo de vida del sistema.</p> <p>Combina las que han demostrado ser las mejores prácticas para desarrollar software.</p> <p>Los requisitos pueden cambiar.</p> <p>Fundamentalmente se trabaja en parejas.</p>



<b>VENTAJAS</b>	<p>Entrega de un producto funcional al finalizar cada sprint.</p> <p>Posibilidad de ajustar la funcionalidad con base a la necesidad de negocio al cliente.</p> <p>Visualización del proyecto día a día.</p> <p>Anticipa los cambios durante el desarrollo.</p>	<p>Da lugar a una programación organizada.</p> <p>Propicia la satisfacción del programador como la del cliente.</p> <p>Fomenta la comunicación entre el cliente y programador, de tal manera que los cambios son oportunos.</p>
-----------------	---	---

Tabla 8. Análisis comparativo de las metodologías

Elaborado por: Diego Chicaiza

### **Descripción de la metodología a aplicarse en el proyecto**

La metodología seleccionada es XP, ya que se implementa para una empresa y en este caso el cliente es el gerente, de tal manera que se puede obtener interacción directa con el cliente.

XP al ser una metodología enfocada a proyectos de mediano y corto plazo es una buena opción, pese a que las metodologías se enfocan a un grupo considerado para la realización de proyectos a producción, aunque lo esencial es que las metodologías ágiles están para ayudar en la administración de tiempos para la entrega oportuna mediante las iteraciones el cliente puede observar el avance del proyecto.

#### **4.2. Fase I: Exploración**

La primera fase de XP, en donde los clientes plantean sus requerimientos que posteriormente se transforman en historias de usuario, el estado con el que se da inicio el proyecto. Adicionalmente se definen Las herramientas que se utilizaran en el desarrollo del proyecto.

##### **4.2.1. Levantamiento de información**

El levantamiento de información sobre la necesidad de la empresa Creaciones Anabel, en el proceso de gestionar las ventas a través de las nuevas tecnologías fue realizado por medio de una entrevista personal, en donde se pudo obtener información de cómo lleva el modelo de negocio actual.

El tipo de entrevista utilizado fue semiestructurado, la cual permite que el entrevistador lleve una pauta o guía con los temas a cubrir, los términos a usar y con orden adecuado. Frecuentemente, los términos usados y el orden de los temas cambian en el curso de

la entrevista, pues tiende a abordar otro tema de interés para el entrevistador y de esta manera surgen preguntas para continuar la entrevista.

Con respecto a un cuestionario que son preguntas cerradas y no permiten captar el sentido de quien se necesita información, la entrevista permite realizar preguntas abiertas, aportan flexibilidad.

Este tipo de entrevistas permite ver con más profundidad y aclarar puntos que se desconoce, además permite ver hasta donde abarca el conocimiento del entrevistado. Para la realización de esta entrevista se utilizó un cuestionario que se muestra en la Tabla A1 de los Anexos.

#### **4.2.2. Análisis de resultados**

La entrevista realizada (ver Anexo A.1) al gerente propietario de Creaciones Anabel permitió determinar que por el momento cuentan con una página en donde pueden mostrar ciertos productos que fabrican, pero no la administran al gusto y conveniencia. Por tanto, Luis Pintado quien es gerente de Creaciones Anabel, expreso que es necesario que una empresa que fabrica pueda crecer en el mercado dando a conocer los productos elaborados a través de medios digitales.

En la Figura 9 se muestra el sitio web que de momento tienen para promocionar y los clientes conozcan los productos. La página de calzadoecuador es un sitio en donde se promocionan muchas empresas pequeñas o medianas para ofrecer y dar a conocer los productos de cada una de ellas. Y la manera de comprar es comunicándose con los dueños de las empresas respectivamente.

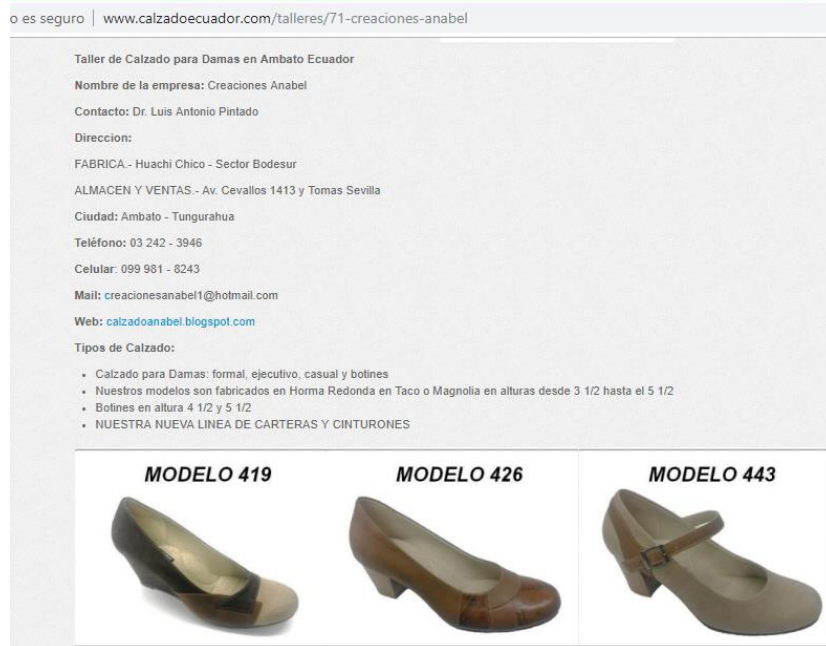


Figura 9. Sitio web para Creaciones Anabel Fuente: [42]

Elaborado por: Diego Chicaiza

Los productos fabricados por Creaciones Anabel están diseñados para calzado de dama, y ciertos accesorios como maletines, correas. Adicional a esto, el uso de la aplicación es para promocionar sus productos y que los clientes o clientes mayoristas las conozcan de tal forma que puedan realizar pedidos en línea.

También, explicó como da a conocer sus productos a ciertos clientes mayoristas para que los puedan adquirir, personalmente se dirige a los clientes y muestra su catálogo de productos con los respectivos modelos en un documento de texto, por tal motivo expreso que es necesario que se realice la aplicación ya que con la misma podrá ver producto por producto a más detalle.

**TACONES.-HORMA REDONDA** *Se pueden fabricar en taco a parte desde altura 3 ½ hasta 5 ½, los mismos modelos en Magnolia. Precio Final*  
**TACO**.....\$ 31,50  
**MAGNOLIA** .....\$ 34.50



Figura 10. Ficha técnica Anabel 2019  
 Fuente: Creaciones Anabel

Elaborado por: Diego Chicaiza

La existencia del inventario es esencial para que los usuarios estén al tanto de cuantos productos están disponibles para la compra respectiva, por ende, da a entender que los clientes pueden comprar o realizar un pedido.

Un punto importante con respecto al cazado es que el cliente final no se siente conforme con el producto hasta que lo siente en sus pies, manifestó el gerente de la empresa. Sin embargo, para los mayoristas es una buena opción poder ofrecer los productos mediante una aplicación móvil.

Por lo tanto, tiene gran expectativa de vender a través de Internet a través de una aplicación móvil o un sitio web, por otra parte, el desarrollo se enfoca para dispositivos con sistema operativo Android ya que es uno de los grandes en teléfonos inteligentes.

#### **4.2.3. Estudio de la situación actual de la institución y sus políticas**

La empresa de producción de calzado “Creaciones Anabel” es una empresa ecuatoriana que inicia sus actividades operativas el año 2000, en la ciudad de Ambato, la misma que está a cargo del Dr. Luis Antonio Pintado, quien ha venido cumpliendo las funciones de propietario y gerente general de la empresa[7].

Creaciones Anabel es una de las empresas de calzado más importantes de la zona, la fábrica se ubica en Huachi Chico en el sector Bodesur en el cantón Ambato y el almacén en la Av. Cevallos 1413 y Tomas Sevilla[7].

Su objeto social es dedicarse a la producción y comercialización de calzado casual, formal, ejecutivo y botines para damas, bajo el criterio de que el cliente tiene la razón, el Dr. Luis Pintado está convencido que la calidad y comodidad es lo que busca todo cliente y cuyo precio de venta se halle en el promedio del calzado femenino comercializado a nivel local, con el propósito de satisfacer adecuadamente las necesidades y expectativas de sus clientes intermedios y usuarias finales[7].

En la provincia de Tungurahua un gran porcentaje de pequeñas o medianas empresas de producción artesanal de calzado no tienen sistemas o aplicaciones que les permitan ganar más clientes en el mercado ya sea por anuncios o carritos de venta para así conocer los productos que ofrecen los mismos, de tal forma que esto podrá mejorar el posicionamiento ante la competencia en un mercado debidamente establecido. Por esta razón el propietario de Creaciones Anabel ha considerado necesario y oportuno desarrollar un plan de mejora y expandir en el mercado y dar a conocer sus productos a través de medios digitales[7].

La empresa Creaciones Anabel posee dentro de su línea de calzado, una amplia gama de modelos, éstos han presentado el mismo estilo desde el inicio de sus actividades comerciales a los clientes de la empresa dentro de los diferentes puntos de venta. Sin embargo, solo cuentan con una página web en donde la información es estática y presentan sus modelos y los clientes que visitan la misma no pueden conocer más acerca de un producto específico. Además cuenta con un blog la misma muestra cierta descripción y sus enlaces no son apropiados[7].

En el futuro Creaciones Anabel pretende aumentar su posición en el mercado local y nacional, en base a precios competitivos, distribución ágil y oportuna del producto y

la optimización de recursos, aspectos que permitirán consolidar el prestigio, confianza y credibilidad, en el ámbito comercial, asumiendo un compromiso integral de constituirse en líder en innovación, calidad y precio, satisfaciendo así las expectativas de sus clientes.

El gerente propietario tiene gran expectativa del desarrollo del proyecto, pues afirma que esto mejora sus ventas y captara clientes en el mercado del calzado para damas. Dar a conocer los productos que fabrica es su meta ya que los mismo son de gran calidad y comodidad.

Sus políticas de se basan en la calidad es por ello que Creaciones Anabel se centra en el objetivo principal y la base de su negocio, lograr la plena satisfacción de sus clientes, respetando el medio ambiente que nos rodea, para lograr sus objetivos, la empresa se compromete cada día a satisfacer las necesidades en especial de las damas ofreciendo un calzado de moda, diseño y confort que satisfaga las necesidades y expectativas de sus clientes, además de dotación que cumpla con las expectativas de comodidad, calidad, durabilidad y seguridad, esto contando con materiales de la alta calidad y fabricado por un personal altamente calificado[7].

El compromiso con el cliente, La empresa orientara sus acciones hacia el cliente, satisfaciendo su necesidades y expectativas, ofreciéndole un producto de excelente calidad. Todo esto para generar confianza en sus clientes y la empresa evitando de esta manera las quejas, reclamos y aumentando la satisfacción del cliente. Para la obtención de información se procedió a realizar una entrevista Anexo 1 al Gerente Dr. Luis Pintado.

#### **4.2.4. Definición de roles**

Los roles permiten organizar a todos los que se encuentran involucrados en el desarrollo del sistema, para evitar inconvenientes, cabe mencionar que, al ser un proyecto de investigación, no se contara con todos los roles de la metodología XP.

<b>Nombre</b>	<b>Nombre rol</b>	<b>Descripción</b>	<b>Rol xp</b>
Diego Chicaiza	Tesista	Persona encargada de la planificación. Diseño, codificación y pruebas para la entrega del aplicativo.	Programador

Ing. Carlos Núñez	Tutor del trabajo de investigación	Persona encargada de las revisiones mensuales para comprobar que se lleve a cabo la finalización del proyecto, según el cronograma de actividades.	Coach
Dr. Luis Pintado	Gerente propietario de la empresa Creaciones Anabel	Persona que necesita el aplicativo y conoce los procesos de venta.	Usuario

Tabla 9. Roles del proyecto

Elaborado por: Diego Chicaiza.

### 4.3. Fase II: Planificación del proyecto

Al emplear la metodología extreme Programming, la actividad principal del desarrollo de la aplicación móvil híbrida, es la interacción con el cliente o usuario final. En consecuencia, se definen varias historias de usuario con el objetivo de cumplir las actividades en periodos de tiempos cortos, en donde el usuario final es el que sugiere cual de todas ellas tiene un valor más significativo.

#### 4.3.1. Historias de usuario

Las historias de usuario son la base del éxito del producto, se trata de una lista de características que el cliente necesita que existan en su producto final, para ello el proyecto fue dividido en iteraciones y por cada iteración una entrega.

Una vez definidas las historias de usuario, se proceden a separarlas en actividades para estimar el tiempo que pueda cumplirse cada una de ellas.

Ya definidos los alcances del proyecto y la entrevista con el usuario y mantener reuniones para obtener los requisitos que la empresa necesita, se definen las siguientes historias de usuarios.

En la Tabla 10, se ilustra una plantilla con la que se detallara una historia de usuario.

<b>HISTORIA DE USUARIO</b>			
<b>Número:</b>		<b>Usuario:</b>	
<b>Nombre de historia:</b>			
<b>Prioridad en el negocio:</b>		<b>Riesgo en el desarrollo:</b>	
<b>Puntos estimados:</b>		<b>Iteración asignada:</b>	

<b>Programador responsable:</b>	
<b>Descripción:</b>	
<b>Observación:</b>	

Tabla 10. Plantilla de Historia de Usuario.

Elaborado por: Diego Chicaiza.

Detalle de cada elemento de la plantilla de historia de usuario.

- **Numero:** Es el identificador de la historia de usuario
- **Usuario:** Persona o grupo que asigna la historia de usuario.
- **Nombre de historia:** Titulo que se le asigna a la historia de usuario.
- **Prioridad en el negocio:** Se asigna los siguientes valores según la necesidad del cliente: Alta, Media y Baja.
- **Riesgo en el desarrollo:** Implica el riesgo que tendrá al desarrollar dicha historia de usuario, los valores son: Alta, Media y Baja.
- **Programador responsable:** Persona que está a cargo de llevar a cabo dicha historia de usuario.
- **Puntos estimados:** El o los días que se tardara en desarrollar la historia de usuario
- **Iteración asignada:** Iteración asignada a dicha historia de usuario.
- **Descripción:** El cliente puede escribir con sus propias palabras, si se desea validaciones o procesos que se deben llevar a cabo, etc.
- **Observación:** Se detallan los procesos que tienen relación con la historia de usuario.

**Prioridad de requerimientos:** Una vez recopilado los requerimientos se definen las prioridades de las historias de usuario.

Prioridad	Criterio
10	Indispensable
9	Esencial
8	Obligatorio
7	Necesario
6	Conveniente

A continuación, se detalla las historias de usuario que conforma los requerimientos del cliente con la respectiva prioridad.



HISTORIA DE USUARIO	
<b>Número:</b> 001	<b>Usuario:</b> Desarrollador
<b>Nombre de historia:</b> Definir la estructura del proyecto	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Una buena definición de la estructura es esencial para futuros mantenimiento y mejoras, por tanto, con la ayuda de Angular se especifica una estructura básica, a la que se debe añadir capa de objetos de negocio, servicios, pipes, componentes, paginas, módulos, recursos o assets.	
<b>Observación:</b>	

Tabla 11. Historia de usuario - Definir la estructura del proyecto

Elaborado por: Diego Chicaiza

HISTORIA DE USUARIO	
<b>Número:</b> 002	<b>Usuario:</b> Desarrollador
<b>Nombre de historia:</b> Especificación de buenas prácticas de programación	
<b>Prioridad en el negocio:</b> Baja	<b>Riesgo en el desarrollo:</b> Baja
<b>Puntos estimados:</b> 7	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Las buenas prácticas de programación ayudaran al mantenimiento futuro de la aplicación	
<b>Observación:</b>	

Tabla 12. Historia de usuario - Especificación de buenas prácticas de programación

Elaborado por: Diego Chicaiza

HISTORIA DE USUARIO	
<b>Número:</b> 003	<b>Usuario:</b> Desarrollador
<b>Nombre de historia:</b> Definición de la base de datos	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alto
<b>Puntos estimados:</b> 10	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Una buena estructura de base de datos será primordial ya que de ella depende el modelo de negocio para el funcionamiento de la aplicación	

Tabla 13. Historia de usuario - Modelado de la base de datos

Elaborado por: Diego Chicaiza

HISTORIA DE USUARIO	
<b>Número:</b> 004	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Ingreso al sistema por correo y contraseña	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> La aplicación al ser un sistema de venta de productos (calzado para dama), se requiere que exista un módulo de autenticación por medio de correo electrónico y contraseña. El usuario puede iniciar sesión para poder realizar todas las operaciones que la misma tenga.	
<b>Observación:</b> Para la autenticación de usuarios se requerirá también un módulo de registro de usuario por medio de correo electrónico y contraseña adicionalmente el nombre de usuario.	

Tabla 14. Historia de usuario -Ingreso al sistema por correo y contraseña

Elaborado por: Diego Chicaiza

HISTORIA DE USUARIO	
<b>Número:</b> 005	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Registrar usuario por correo y contraseña	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Los clientes podrán registrar sus datos para poder hacer uso de la aplicación sin restricción alguna.	
<b>Observación:</b> Los clientes en el caso de no iniciar sesión o no registrarse en la aplicación no podrán hacer uso de los módulos.	

Tabla 15. Historia de usuario - Registro de usuarios por medio de correo y contraseña

Elaborado por: Diego Chicaiza

HISTORIA DE USUARIO	
<b>Número:</b> 006	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Restablecer contraseña	

<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Baja
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 1
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Los clientes pueden restablecer la contraseña ingresando el correo electrónico con el que se registraron.	
<b>Observación:</b> Los clientes pueden restablecer la contraseña y esta se notificará el mismo correo con el que se registraron	

Tabla 16. Historia de usuario - Restablecer contraseña

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 007	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Inicio de sesión y registro por redes sociales (Google)	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 6	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Los clientes pueden ingresar a la aplicación por medio de una cuenta Google para hacer uso total de la aplicación	
<b>Observación:</b> Es necesario contar con una cuenta Google para poder realizar esta actividad.	

Tabla 17 Historia de usuario - Inicio de sesión y registro de usuario por medio de red social (Google)

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 008	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla de inicio	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> En esta se dará la bienvenida al sistema y se mostrará los productos	
<b>Observación:</b> La pantalla de inicio también permitirá agregar los productos a la cesta.	

Tabla 18. Historia de usuario - Crear una pantalla de inicio

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 009	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla de productos favoritos	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> cada producto agregado a favoritos solo se mostrará con la posibilidad de agregar al carrito o a la vez eliminar de favoritos.	
<b>Observación:</b> Cuando un producto es agregado a favoritos solo se agregará en localStorage, se guardará cuando se procese la compra.	

Tabla 19. Historia de usuario - Crear una pantalla de productos favoritos

Elaborado por: Diego Chicaiza.

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 010	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla para editar los datos del perfil de usuario.	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Una vez iniciado sesión por primera vez se contará con el nombre de usuario y correo electrónico, en esta pantalla se puede editar lo siguiente: <ul style="list-style-type: none"> <li>• la fecha de nacimiento.</li> <li>• Dirección / Domicilio.</li> <li>• Genero.</li> <li>• Foto de perfil.</li> <li>• Descripción breve de sí mismo.</li> </ul>	
<b>Observación:</b> Los campos que son requeridos se marcaran con un asterisco.	

Tabla 20 Historia de usuario - Crear una pantalla para editar los datos del perfil de usuario

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 011	<b>Usuario:</b> Desarrollador
<b>Nombre de historia:</b> Definir guard para restringir o permitir el acceso a las distintas páginas.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta

<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 2
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Cuando el usuario esta con estado de inicio de sesión correcto puede navegar por los distintos módulos, sino solo tiene acceso a la login y register.	
<b>Observación:</b>	

Tabla 21. Definir guard para restringir o permitir el acceso a las distintas páginas

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 012	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla para mostrar los detalles de un producto.	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Se debe crear una pantalla para que el usuario pueda ver los detalles de un producto seleccionado para posteriormente agregar a la lista de favoritos o al carrito.	
<b>Observación:</b> Para acceder a los distintos módulos, se verificará que este autenticado.	

Tabla 22. Historia de Usuario Crear una pantalla para mostrar los detalles de un producto

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 013	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear pantalla para agregar a favoritos	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> El usuario puede marcar un producto como favorito y se debe guardar para que en caso de gustar puede proceder agregar a la cesta de artículos.	
<b>Observación:</b> Estos artículos no se deben guardar para la orden hasta que se añada a la cesta de artículos. .	

Tabla 23. Historia de Usuario - Agregar productos a favoritos

Desarrollado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 014	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos	
<b>Prioridad en el negocio:</b> Media	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 6	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Una vez el cliente seleccione productos y los agregue a favoritos, al ir al menú favoritos debe aparecer dichos productos.	
<b>Observación:</b> Para acceder a los distintos módulos, se verificará que este autenticado.	

Tabla 24. Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 015	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla para enlistar los productos.	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 3
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> En esta pantalla se puede ver todos los productos, implementar un método de búsqueda para los artículos. Los mismos permitirán guardar a favoritos, ver el detalle del producto.	
<b>Observación:</b> Para acceder a los distintos módulos, se verificará que este autenticado. Se debe controlar que no se repitan al agregar a la cesta en tal caso debería seguir sumando el producto seleccionado.	

Tabla 25. Historia de usuario - Crear una pantalla para enlistar los productos

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 016	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras	

<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> En esta pantalla se debe listar los productos que fueron agregados al carrito, se debe verificar la existencia. Se debe realizar el cálculo correspondiente para observar el total de la compra.	
<b>Observación:</b> Se debe estar autenticado para acceder a este modulo	

Tabla 26. Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 017	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear la pantalla para la forma de pago	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> En esta pantalla se confirmará y seleccionara cual es la forma de pago que desea el cliente	
<b>Observación:</b> Se debe estar autenticado para acceder a este modulo	

Tabla 27. Crear la pantalla para la forma de pago

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 018	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Crear una pantalla para mostrar las compras realizadas	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 4
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> En esta pantalla se mostrará las compras que el cliente ha realizado.	
<b>Observación:</b> Se debe estar autenticado para acceder a este módulo	

Figura 11. Crear una pantalla para mostrar las compras realizadas

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 019	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Subir la aplicación en el Hosting que provee Firebase	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 5
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Una vez subida la aplicación al hosting de Firebase se puede acceder desde cualquier lugar para esto se debe estar conectado a Internet.	
<b>Observación:</b> Se debe estar autenticado para acceder a este modulo	

Figura 12. Subir la aplicación en el Hosting que provee Firebase

Elaborado por: Diego Chicaiza

<b>HISTORIA DE USUARIO</b>	
<b>Número:</b> 017	<b>Usuario:</b> Todos
<b>Nombre de historia:</b> Subir la aplicación para la tienda de Google Play	
<b>Prioridad en el negocio:</b> Alta	<b>Riesgo en el desarrollo:</b> Alta
<b>Puntos estimados:</b> 9	<b>Iteración asignada:</b> 5
<b>Programador responsable:</b> Diego Chicaiza	
<b>Descripción:</b> Se debe subir la aplicación a la tienda para poder instalar en los teléfonos Android.	
<b>Observación:</b> De acuerdo con las políticas de Google puede que este proceso tarde o de inconvenientes.	

Figura 13. Subir la aplicación para la tienda de Google Play

Elaborado por: Diego Chicaiza

#### 4.3.2. Estimación Historias de Usuario

Al ser un proyecto en el cual se llevó a cabo el desarrollo de dos aplicaciones, como lo es el aplicativo móvil híbrido que será utilizado por el o los clientes finales y la parte web que será la que administra los productos, clientes, parámetros, etc.

Para la estimación se consideró un trabajo de 2 horas por día laborable en las historias de usuario. Por lo tanto, se presenta el plan de entrega correspondiente a la aplicación móvil.



Iteración	Numero	Historia de usuario	Tiempo estimado	
			Días	Horas
1	001	Definir la estructura del proyecto	2	4
1	002	Especificación de buenas prácticas de programación	3	6
1	003	Definición de la base de datos	6	12
1	004	Ingreso al sistema por correo y contraseña	5	10
1	005	Registrar usuario por correo y contraseña	5	10
1	006	Restablecer contraseña	3	6
2	007	Inicio de sesión y registro por redes sociales (Google)	10	20
2	008	Crear una pantalla de inicio	5	10
2	009	Crear una pantalla de productos favoritos	2	4
2	010	Crear una pantalla para editar los datos del perfil de usuario.	4	8
2	011	Definir guard para restringir o permitir el acceso a las distintas páginas.	3	6
3	012	Crear una pantalla para mostrar los detalles de un producto.	3	6
3	013	Crear pantalla para agregar a favoritos	3	6
3	014	Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos	3	6
3	015	Crear una pantalla para enlistar los productos.	5	10
4	016	Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras	6	12
4	017	Crear la pantalla para la forma de pago con PayPal	6	12
4	018	Crear una pantalla para mostrar las compras realizadas	5	10
5	019	Subir la aplicación al hosting de Firebase	3	6
5	020	Subir la aplicación a la Google Play	12	24
<b>Total, de tiempo estimado (días, horas)</b>			<b>94</b>	<b>188</b>

Tabla 28. Estimación de historias de usuario

Elaborado por: Diego Chicaiza

#### 4.4. Fase III: Iteraciones

##### 4.4.1. Plan de entrega

Para la elaboración del Plan de Entrega del proyecto además con la aplicación de los parámetros que se implementa en la metodología XP, se procede a establecer el tiempo calendario de acuerdo con un mes de cuatro semanas (cinco días hábiles), en días de dos horas.

N°	Historia de usuario	Tiempo estimado		Iteración asignada					Entrega asignada				
		Días	Horas	1	2	3	4	5	1	2	3	4	5
001	Definir la estructura del proyecto.	2	4	x					x				
002	Especificación de buenas prácticas de programación.	3	6	x					x				
003	Definición de la base de datos	6	12	x					x				
004	Ingreso al sistema por correo y contraseña.	5	10	x					x				
005	Registrar usuario por correo y contraseña.	5	10		x					x			
006	Restablecer contraseña	3	6		x					x			
007	Inicio de sesión y registro por redes sociales (Google).	10	20		x					x			
008	Crear una pantalla de inicio.	5	10		x					x			
009	Crear una pantalla de productos favoritos.	2	4		x					x			
010	Crear una pantalla para editar los datos del perfil de usuario.	4	8		x					x			
011	Definir guard para restringir o permitir el acceso a las distintas páginas.	3	6		x					x			
012	Crear una pantalla para mostrar los detalles de un producto.	3	6			x					x		
013	Crear pantalla para agregar a favoritos	3	6			x					x		
014	Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos	3	6			x					x		
015	Crear una pantalla para enlistar los productos.	5	10			x					x		

016	Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras.	6	12				x					x	
017	Crear la pantalla para la forma de pago con PayPal.	6	12				x					x	
018	Crear una pantalla para mostrar las compras realizadas.	5	10				x					x	
019	Subir la aplicación al hosting de Firebase.	3	6					x					x
020	Subir la aplicación a la Google Play.	12	24					x					x

Tabla 29. Plan de entrega

Elaborado por: Diego Chicaiza.

#### 4.4.2. Iteraciones

Cada una de las historias de usuario en la Tabla 30 se representa con el valor estimado establecido entre los roles establecidos y el cliente.

	N°	Historia de usuario	Prioridad	Riesgo	Estado de desarrollo	Prueba
Primera iteración	001	Definir la estructura del proyecto	10	Alto	Completo	Aprobado
	002	Especificación de buenas prácticas de programación	7	Bajo	Completo	Aprobado
	003	Definición de la base de datos	10	Alto	Completo	Aprobado
	004	Ingreso al sistema por correo y contraseña	9	Alto	Completo	Aprobado
	005	Registrar usuario por correo y contraseña	9	Medio	Completo	Aprobado
	006	Restablecer contraseña	9	Medio	Completo	Aprobado
Segunda iteración	007	Inicio de sesión y registro por redes sociales (Google)	9	Alto	Completo	Aprobado
	008	Crear una pantalla de inicio	9	Medio	Completo	Aprobado
	009	Crear una pantalla de productos favoritos	9	Alto	Completo	Aprobado
	010	Crear una pantalla para editar los datos del perfil de usuario.	9	Alto	Completo	Aprobado

	011	Definir guard para restringir o permitir el acceso a las distintas páginas.	9	Alto	Completo	Aprobado
Tercera iteración	012	Crear una pantalla para mostrar los detalles de un producto.	9	Alto	Completo	Aprobado
	013	Crear pantalla para agregar a favoritos	9	Alto	Completo	Aprobado
	014	Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos	9	Alto	Completo	Aprobado
	015	Crear una pantalla para enlistar los productos.	9	Alto	Completo	Aprobado
Cuarta iteración	016	Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras	9	Alto	Completo	Aprobado
	017	Crear la pantalla para la forma de pago con PayPal	10	Alto	Completo	Aprobado
	018	Crear una pantalla para mostrar las compras realizadas	10	Alto	Completo	Aprobado
Quinta iteración	019	Subir la aplicación al hosting de Firebase	10	Alto	Completo	Aprobado
	020	Subir la aplicación a la Google Play	10	Alto	Completo	Aprobado

Tabla 30. Resumen de iteraciones

Elaborado por: Diego Chicaiza.

## 4.5. Fase IV: Producción

### 4.5.1. Iteración 1

En la planificación se definió las siguientes historias de usuario.

Iteración	Numero	Historia de usuario	Tiempo estimado	
			Días	Horas
1	001	Definir la estructura del proyecto	2	4
1	002	Especificación de buenas prácticas de programación	3	6
1	003	Definición de la base de datos	6	12
1	004	Ingreso al sistema por correo y contraseña	5	10
1	005	Registrar usuario por correo y contraseña	5	10
1	006	Restablecer contraseña	3	6

	Total	24	48
--	-------	----	----

#### 4.5.1.1. Definición de la estructura del proyecto

Para el desarrollo de proyecto se define las herramientas que fueron utilizadas para llevar a cabo, tanto como software como hardware.

##### Hardware

- 1 laptop Asus X542U 16 Gb de memoria RAM, core i7, 2 Ghz
- 1 Smartphone Samsung Galaxy J5 Prime 2 GB de Ram, Versión de Android 8.0.0

##### Software

Se trabajó con una conexión inalámbrica estable, el IDE para programar fue Visual Studio Code, para iniciar el proyecto lo primordial fue la instalación de Node Js que fue requerido para utilizar el gestor de paquetes NPM.

Posteriormente se instaló Cordova y el Framework de Ionic

```
npm install -g cordova ionic
```

A continuación, varios comandos que se utilizaron con su respectiva descripción:

##### Crear el proyecto

```
ionic start CreacionesAnabelApp tabs
```

Ejecutar el proyecto en el navegador

```
ionic serve --labs
```

Agregar plataforma Android

```
ionic cordova platform add android
```

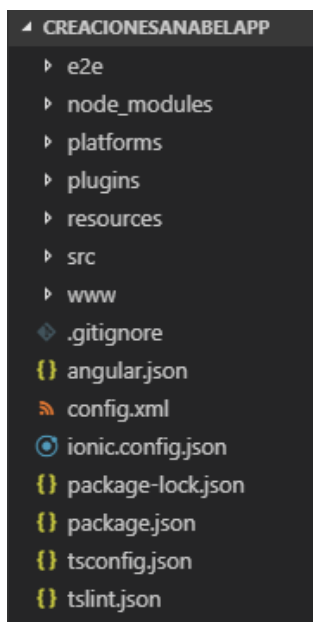
Construir y ejecutar en un dispositivo móvil Android.

```
ionic cordova run android
```

#### 4.5.1.2. Modelado y estructura de la aplicación

La estructura del proyecto ya es definida por el propio framework, en este caso Angular 7 en su nueva actualización a definido la siguiente manera de definir los directorio y archivos de configuración, de esta manera nos ahorra tiempo en vez de configurarlo parte por parte.

### Estructura del proyecto Ionic 4 con Angular 7



En la figura se muestra la estructura de proyecto que nos facilita angular cli.

La carpeta **e2e** es utilizada para la realización de pruebas de punto a punto o también conocidas como pruebas de integración, asegurando una aplicación de alto nivel con un funcionamiento correcto.

La carpeta **node\_modules**, es donde se almacenan los paquetes necesarios que son descargados en la instalación de Ionic, además los paquetes que se descarguen se realizaran a través del gestor de paquetes de node.js también conocido como npm.

La carpeta **platforms**, en esta residen las distintas plataformas añadidas por el usuario con el fin de convertirlas en app para instalarlas en los dispositivos móviles.

La carpeta **src** contiene básicamente toda la funcionalidad, ya que en la misma se encuentran servicio, paginas, pipes, componentes, módulos.

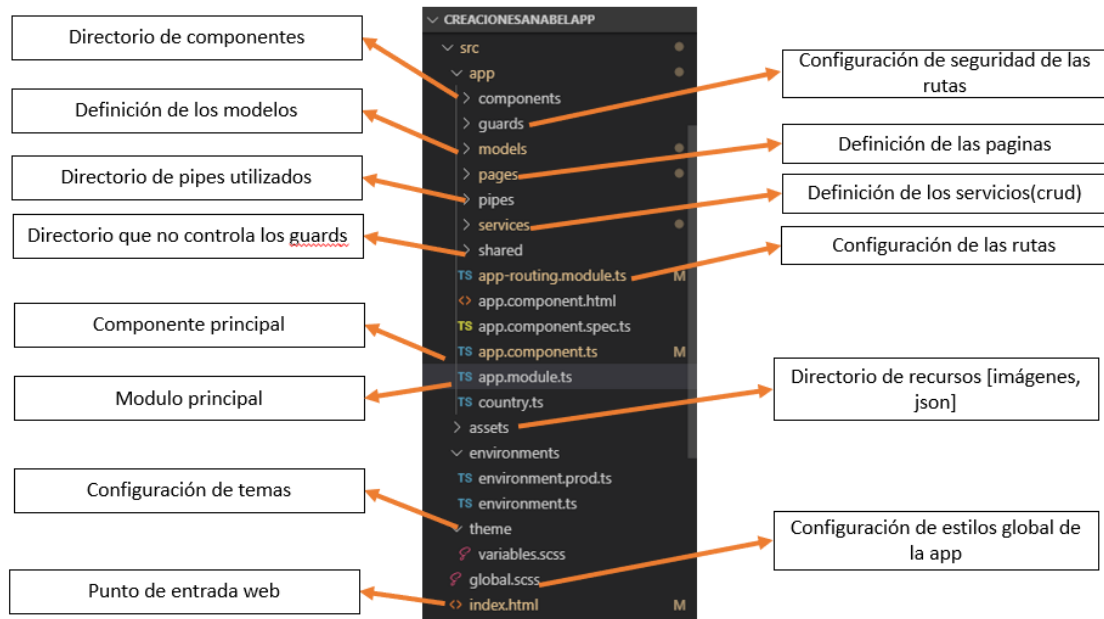


Figura 14. Definición de estructura del proyecto

Elaborado por: Diego Chicaiza

### 4.5.1.3. Especificación de buenas prácticas de programación

las buenas prácticas de programación envuelven un conjunto formal o informal de reglas, siendo estas opcionales u obligatorias. Se adoptan con el objetivo de aumentar la calidad del software obteniendo beneficios como los son[43]:

- Mejorar la legibilidad y mantenibilidad del código fuente, mismo que será importante para que otros desarrolladores entiendan la función de cada método, variable.[43]
- Facilita el proceso de desarrollo para el o los programadores.
- Evita la creación de clases o métodos innecesarios.
- Entre otras ventajas.

### Consejos para la mantenibilidad del código fuente

#### Utilizar nombres descriptivos

Las variables, métodos, clases, interfaces, etc. deben representar con un nombre descriptivo. En el caso de las clases debe representar una entidad como, por ejemplo: User, Product, Country, etc. Los métodos al representar acción deben contener alguna referencia con un verbo (getProducts, createProduct, deleteProduct, etc.). y las

variables deben describir el propósito a utilizar, se puede observar de la colección de Products que se describe en la Tabla 32 como se definieron los atributos[43].

```
createUser(user: User) {  
    return this._afs.collection("users").add(user);  
}
```

Al usar nombres descriptivos, comentar el código no será necesario ya que cualquier programador podrá leer la programación o tener alguna noción del funcionamiento.

### Evitar comentarios innecesarios

No es necesario explicar cosas obvias como por ejemplo una condicionar `x == 5` y sobre ella explicar con un comentario. En este caso el comentario es innecesario, se puede eliminar ya que no aporta.

```
// Comprueba si el valor es igual a todos  
if (valor == "todos") {  
    this.categorySearch = "";  
    return;  
}
```

### Evitar escribir funciones muy largas

Cuando una función o método es muy larga no se consigue una comprensión adecuada, dificultan su legibilidad y por tanto su mantenimiento, si por alguna manera falla es más engorroso buscar donde ocurrió el error.

### Indentación

Utilice indentación para las operaciones que se efectúan en una estructura de control, tal como un if, for, while o do-while, y para el contenido de funciones, registros y otros constructos del lenguaje. Esto resalta la estructura lógica del código y simplifica su lectura. Por ejemplo[43]:

Incorrecto	Correcto
<pre>doGoogleLogin() { if (this.platform.is("cordova")) { this.nativeGoogleLogin(); } else {</pre>	<pre>doGoogleLogin() { if (this.platform.is("cordova"))     this.nativeGoogleLogin(); else</pre>



<pre>this.AuthLogin(provider);} }</pre>	<pre>this.AuthLogin(provider); }</pre>
---	--

#### 4.5.1.4. Modelado de la base de datos

Como se menciona este proyecto utilizara el servicio BaaS (Backend as a Service) Firestore como almacén de datos y extracción de los mismos. Las operaciones son visibles para los usuarios que harán uso de la aplicación en tiempo real. En la

Tabla 1, se detalla porque utilizar Firestore y no Firebase Realtime Database. El modelo de la base de datos para la aplicación es la siguiente.

##### Colecciones

Al igual que los SGBDs, las tablas o mejor conocidas como colecciones donde se almacenan los registros que se conocen como documentos. Al ser NoSql, son libres de esquema como lo es la estrecha relación de entidad-relación, lo que quiere decir es que un documento no necesariamente debe tener el mismo número de atributos, ni el mismo tipo.

Sin embargo, deben contar con un atributo obligatorio y único que se tomara como clave primaria para hacer referencia desde todo el contexto de la base de datos.

Atributos: Son pares de clave – valor o asociaciones de clave – conjunto de valores.

En la contiene una vista general de las colecciones que utilizara la aplicación realizada en Ionic y así mismo para la aplicación web.

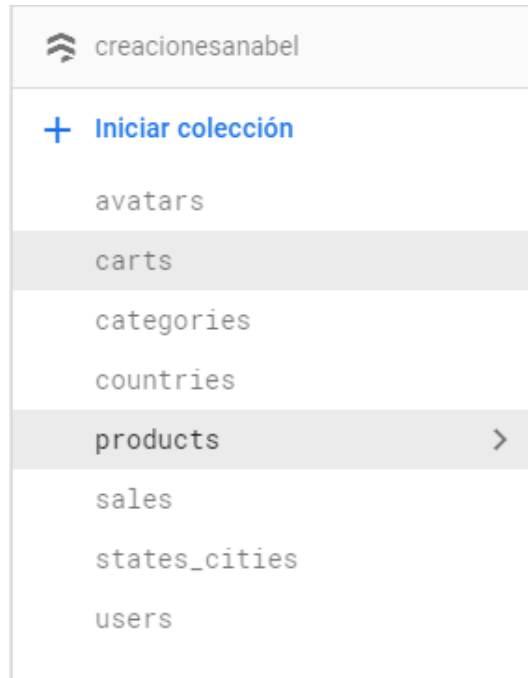


Figura 15. Vista general de las colecciones de Firestore

Elaborado por: Diego Chicaiza

A continuación, se presentarán el modelo de las colecciones con los atributos a utilizar.

Colección - Categories		
Campo	Tipo	Descripción
id	String	Almacena la identificación de la categoría
category_name	String	Almacena el nombre de la categoría
category_url	String	Almacena la ruta de la imagen que representa a la categoría.
Description	String	Una descripción de la categoría

Tabla 31. Colección – Categories

Elaborado por: Diego Chicaiza.

Colección – Products		
Campo	Tipo	Descripción
id	String	Almacena la identificación del producto
product_name	String	Almacena el nombre del producto
product_url	String	Almacena el enlace de la imagen del producto

available_price	String	Almacena un arreglo de los tipos de material de producto (taco, magnolia, casual)
available_sizes	String	Almacena un arreglo con las tallas disponibles por producto
category_name	String	Almacena el nombre de la categoría del producto
category_subname	String	Almacena el nombre de la subcategoría
colors	String	Almacena un arreglo con los colores de producto que dispone
insole	String	Almacena la especificación de la construcción de la plantilla
material	String	Almacena una descripción de la fabricación del producto
paking	String	Guarda la descripción del empaque del calzado
stock	Number	Almacena el stock de los productos que no son taco ni magnolia
stock_taco	Number	Almacena el stock del producto de material taco
stock_magnolia	Number	Almacena el stock del producto de material magnolia
unit_price	Number	Almacena el precio unitario del producto casual
unit_price_taco	Number	Almacena el precio unitario del producto de tipo taco
unit_price_magnolia	Number	Almacena el precio unitario del producto de tipo taco

Tabla 32. Colección – Products

Elaborado por: Diego Chicaiza

<b>Colección – Sales</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	String	Almacena la identificación de la compra realizada.
cartItem	Array	Almacena los productos que fueron agregados a esa compra.
create_time	String	Guarda la hora, fecha en la que se generó la compra
intent	String	Almacena el motivo de la compra
stated	String	Almacena el estado de la compra (Aprobada o Rechazada, etc.)

userId	String	Almacena el id del usuario que genero la compra
--------	--------	---

Tabla 33. Colección – Sales

Elaborado por: Diego Chicaiza

<b>Colección – Countries</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	String	Almacena la identificación del país.
name	String	Almacena el nombre del país
pone_code	String	Almacena el código de teléfono según el país
currency	String	Almacena el tipo de moneda dependiendo el país

Tabla 34. Colección – Countries

Elaborado por: Diego Chicaiza.

<b>Colección – States_cities</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	String	Almacena la identificación de la provincia.
name	String	Almacena el nombre de la provincia
id_contry	String	Almacena el código del país
Cities	Array	Almacena las ciudades que pertenecen a la provincia

Tabla 35. Colección – States

Elaborado por: Diego Chicaiza

<b>Colección – Users</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	String	Almacena la identificación del usuario.
userName	String	Almacena el nombre del usuario.
email	String	Almacena el correo electrónico.
photoURL	String	Almacena una url de la imagen de perfil.
phoneNumber	String	Almacena el teléfono del usuario.
emailVerified	String	Campo para verificar si comprobó el correo electrónico.
Address	String	Almacena la dirección del usuario.
Gender	String	Almacena el género del usuario.
Address	String	Almacena la dirección del usuario.

Tabla 36. Colección Users

Elaborado por: Diego Chicaiza

<b>Colección – Avatars</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	String	Almacena la identificación del avatar.

link	String	Almacena una url para cargar los avatars y cambiar en el perfil de usuario.
------	--------	---

Tabla 37. Colección – Avatars

Elaborado por: Diego Chicaiza

<b>Colección – Parameters</b>		
<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	String	Almacena la identificación del avatar.
name	String	Almacena una url para cargar los avatars y cambiar en el perfil de usuario.
Values	Array	Almacena varios tipos de valores de cadena para un propósito específico dentro de la configuración de la app, logo, nombre empresa, etc.

Tabla 38. Corrección – parameters

Elaborado por: Diego Chicaiza.

#### 4.5.1.5. Ingreso al sistema por correo y contraseña

A continuación, se definen las respectivas pantallas diseñadas en Ionic Framework en su versión 4.12.0. Ionic procura hacer uso de HTML5 y CSS3 conjunto con Typescript para ofrecer una experiencia agradable para el usuario consumidor. Además, aprovecha los principios de diseño web adaptativo o responsive con la finalidad de adaptar a la mayor parte de pantalla o densidad de pixeles, de esta manera hace que soporte a la mayor parte de dispositivos[44].

La aplicación debe ser observada desde el punto de vista del usuario, en otras palabras, pensar en la característica que hagan más flexibles el uso de la aplicación al cliente o usuario final.

#### **Pantalla: Inicio de sesión**

Para el inicio de sesión el usuario tiene tres formas de hacerlo:

- Con API de Facebook
- Con API de Google
- Con correo electrónico y contraseña

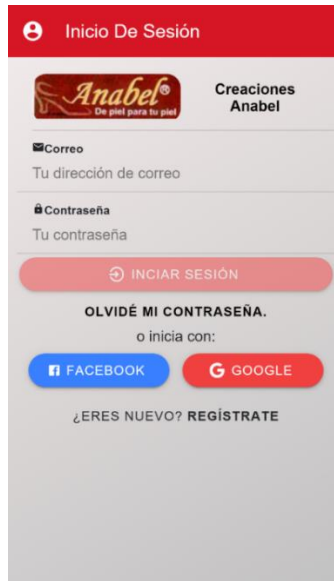


Figura 16. Pantalla Inicio de sesión con correo electrónico y redes sociales

Elaborado por: Diego Chicaiza

### **Pantalla: Registro de usuario**

La pantalla de registro de usuario permite crear un usuario ingresando el nombre de usuario, correo y contraseña, también tiene la posibilidad de ingresar con las redes sociales de Facebook y Google. Se podrá acceder a los módulos una vez se compruebe el correo electrónico.

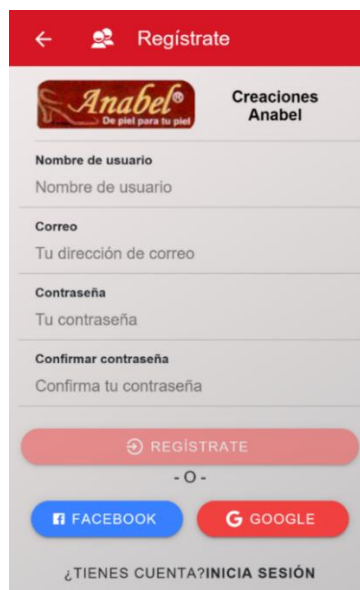


Figura 17. Pantalla Registro de usuario

Elaborado por: Diego Chicaiza

### **Pantalla: Verificación de cuenta**

Luego de realizar el proceso de registro como se muestra en la Figura 17, se enviará automáticamente un mensaje de confirmación de cuenta misma que provee Firebase, una vez aceptado y verificado la cuenta el usuario puede hacer uso de los demás módulos.

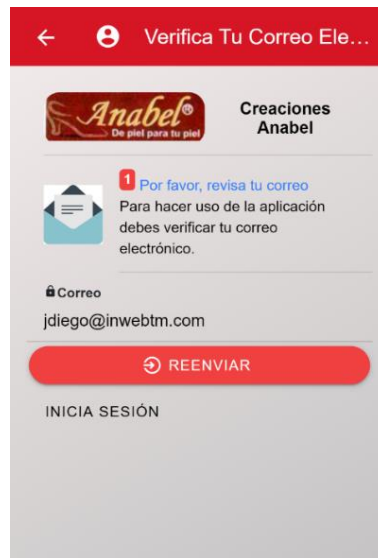


Figura 18. Pantalla: Verificación de cuenta

Elaborado por: Diego Chicaiza

### **Pantalla: Recuperación de contraseña**

Para la recuperación de contraseña Firebase provee de un servicio para recuperar la cuenta, para ellos se debe ingresar el correo electrónico con el que se registró en la aplicación, una vez ingresado el correo se debe enviar la solicitud pinchando o haciendo clic en el botón Restablecer la contraseña.

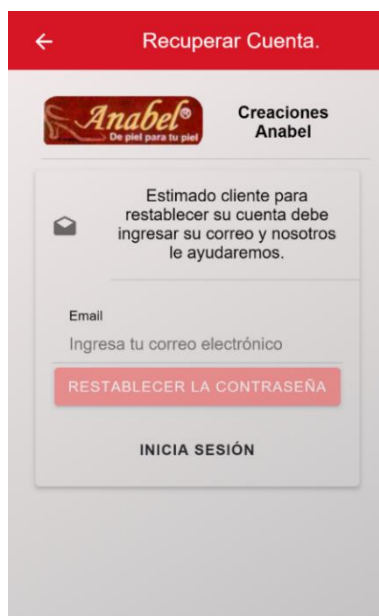


Figura 19. Pantalla Recuperación de cuenta

Elaborado por: Diego Chicaiza

#### 4.5.2. Iteración 2

Iteración	Numero	Historia de usuario	Tiempo estimado	
			Días	Horas
2	007	Inicio de sesión y registro por redes sociales (Google)	10	20
2	008	Crear una pantalla de inicio	5	10
2	009	Crear una pantalla de productos favoritos	2	4
2	010	Crear una pantalla para editar los datos del perfil de usuario.	4	8
2	011	Definir guard para restringir o permitir el acceso a las distintas páginas.	3	6
Total			24	48

##### 4.5.2.1. Inicio de sesión y registro por redes sociales (Google)

En la pantalla de inicio de sesión o registro se puede iniciar el proceso de autenticación con la red social de Google, funciona tanto como para la parte web y la parte nativa.



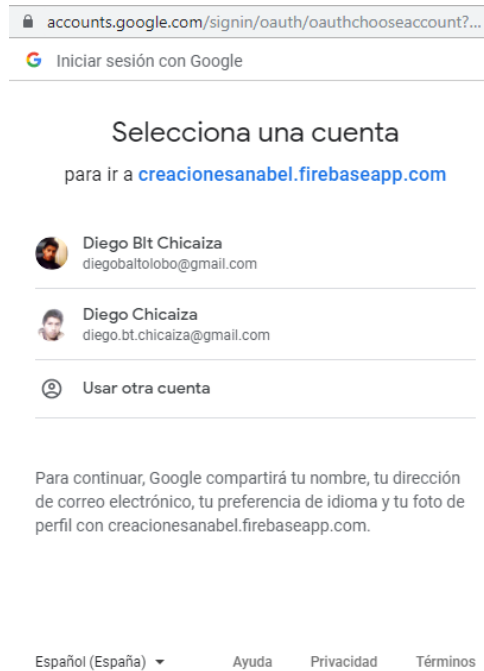


Figura 20. Inicio de sesión y registro por redes sociales (Google)

Elaborado por: Diego Chicaiza.

#### 4.5.2.2. Crear una pantalla de inicio

##### **Pantalla: Inicio o principal**

En la Figura 21, se muestra varias opciones, a continuación, se detalla las funcionalidades de la pantalla Inicio:

- **Menú de opciones:** Dispone de varios accesos a la aplicación, se puede visualizar desplazando con el dedo o cursor desde cualquier extremo izquierdo a la derecha. En este menú se puede cerrar sesión, ver el detalla mínimo del usuario y accesos como el carrito de compras, los productos favoritos, las ordenes, etc.
- **Slider de productos:** En esta sección se puede visualizar los productos de manera dinámica, así mismo permite ver los detalles del producto o añadir a la lista de favoritos.

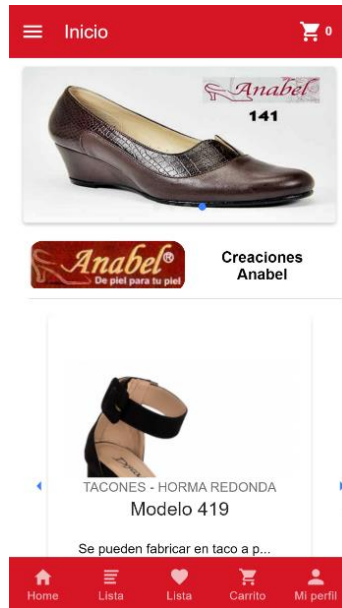


Figura 21. Pantalla Principal

Elaborado por: Diego Chicaiza

#### 4.5.2.3. Crear una pantalla para editar los datos del perfil de usuario

En la siguiente pantalla se puede ingresar datos adicionales del usuario que inicia sesión como se muestra en la Figura 22, los campos obligatorios son el nombre de usuario, correo, teléfono, genero, y una dirección.

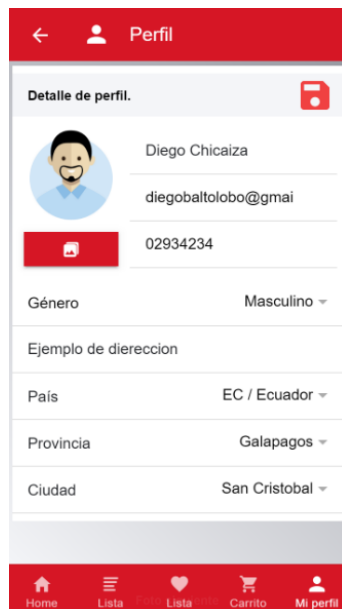


Figura 22. Pantalla - Perfil de Usuario

Elaborado por: Diego Chicaiza

#### 4.5.2.4. Crear una pantalla de productos favoritos

En la sección de Lista de Favoritos se puede agregar a la cesta de compra, se puede eliminar de la lista o a su vez se puede ver los detalles del producto. Solo se puede acceder a este módulo si se encuentra autenticado.



Figura 23. Pantalla - Lista de Favoritos

Elaborado por: Diego Chicaiza

#### 4.5.2.5. Definir guard para restringir o permitir el acceso a las distintas páginas

Guard: Estos permiten proteger los accesos a las rutas de la aplicación, dependiendo de la función el enrutador aceptara o no a la ruta solicitada. Hacen esta búsqueda mediante un valor de retorno booleano (true o false).

Existe varios guards y cada uno tiene su función particular. El comportamiento del enrutador se modifica de manera diferente con cada uno que se utilice. A continuación, se enlista varios guards.

CanActivate: Observa si el usuario puede acceder a una ruta determinada.

CanActivateChild: Observa si el usuario puede acceder a las rutas derivadas de una ruta superior.

CanDeactivate: Observa si el usuario puede salir de una ruta, por ejemplo, se puede lanzar un mensaje de confirmación si tiene cambios por guardar.

CanLoad: Para cargar la aplicación con Lazy Load.

A continuación, se detalla un guard CanActivate que permitirá o no acceder a la pantalla principal si inicia sesión y se ha comprobado su correo electrónico, caso contrario se mostrara un mensaje con información de que no puede o no tiene permisos para acceder a la pantalla principal. Figura 21 y otro guard que no permita regresar a las pantallas de autenticación si ya inicio sesión y su cuenta ha sido comprobada respectivamente se detalla en la Figura 24, Figura 25 respectivamente.

```
canActivate(  
  next: ActivatedRouteSnapshot,  
  state: RouterStateSnapshot  
) : Observable<boolean> | Promise<boolean> | boolean {  
  return this.afsAuth.authState.pipe(  
    map(auth => {  
      if (!isNullOrUndefined(auth)) {  
        if (!auth.emailVerified) {  
          this.toast.showToastWithCloseButton();  
        }  
        if (auth.emailVerified) return true;  
        else {  
          this.router.navigate(["/verify-email"]);  
          return false;  
        }  
      } else {  
        this.router.navigate(["/login"]);  
        this.toast.showToastWithCloseButton();  
        return false;  
      }  
    }));  
}
```

Figura 24. Guard AuthGuard

Elaborado por: Diego Chicaiza

```
canActivate(  
  next: ActivatedRouteSnapshot,  
  state: RouterStateSnapshot  
) : Observable<boolean> | Promise<boolean> | boolean {  
  return this.afsAuth.authState.pipe(  
    map(auth => {  
      if (isNullOrUndefined(auth)) {  
        return true;  
      } else {  
        this.router.navigate(["/home"]);  
        return false;  
      }  
    }));  
}
```

```

    }
  }));
}

```

Figura 25. Guard NologinGuard

Elaborado por: Diego Chicaiza.

Así mismo se definió las distintas rutas de la aplicación con los respectivos guards.

```

const routes: Routes = [
  {
    path: "login",
    loadChildren: "./pages/login/login.module#LoginPageModule",
    canActivate: [NologinGuard]
  },
  {
    path: "register",
    loadChildren: "./pages/register/register.module#RegisterPageModule",
    canActivate: [NologinGuard]
  },
  { . . . },
];
@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule {}

```

Figura 26. Definición de rutas en AppRoutingModuleModule

Elaborado por: Diego Chicaiza

### 4.5.3. Iteración 3

Iteración	Numero	Historia de usuario	Tiempo estimado	
			Días	Horas
3	012	Crear una pantalla para mostrar los detalles de un producto.	3	6
3	013	Crear pantalla para agregar a favoritos	3	6
3	014	Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos	3	6
3	015	Crear una pantalla para enlistar los productos.	5	10
Total			14	28

#### 4.5.3.1. Crear una pantalla para mostrar los detalles de un producto

En las siguientes pantallas se puede ir a ver el detalle del producto, Lista de Favoritos, Lista de Productos, Inicio. Por medio de un enlace con el icono de un ojo como se explica en la Figura 27.

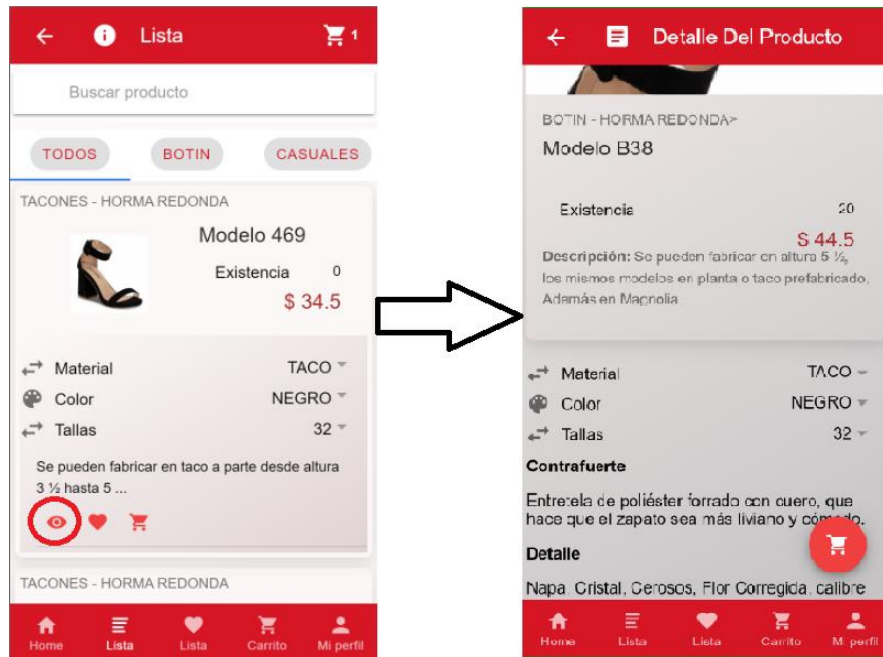


Figura 27. Pantalla - Detalle del Producto

Elaborado por: Diego Chicaiza

#### 4.5.3.2. Crear pantalla para agregar a favoritos

Para agregar productos a la lista de favoritos se debe presionar sobre el botón que tiene una imagen de corazón. Una vez agregado a la lista de favoritos verifica que no se duplique el producto y en el caso de que lo vuelva a seleccionar saldrá una advertencia para notificar que el producto ya ha sido agregado a Favoritos.

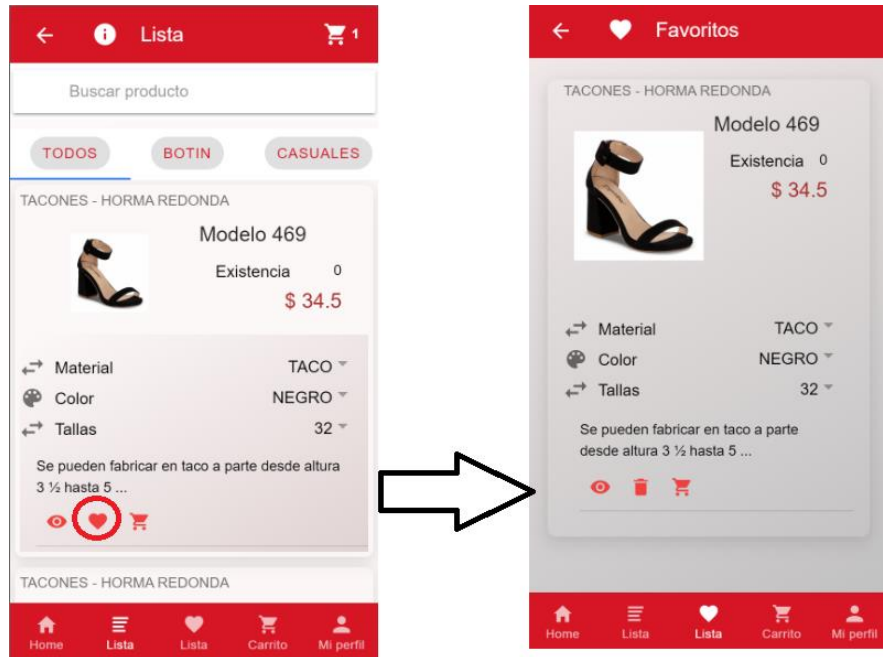


Figura 28. Agregar a la Lista de Favoritos

Elaborado por: Diego Chicaiza

#### 4.5.3.3. Crear una pantalla para mostrar los productos que fueron seleccionados como favoritos

Los productos que son añadidos a esta sección se guardaran localmente (localstorage), se realiza un proceso de control para no duplicar el producto como Favorito. Para acceder a este módulo se debe estar autenticado.

```

addProductToFavorite(product: Product)
  if (!existProductInListFav(product)) {
    localStorage.push(product);
    localStorage.setItem("avf_item", JSON.stringify(product));
    this.toast.showToast("Producto añadido a favoritos.");
  } else
    this.toast.showToast("El producto ya se encuentra añadido a favoritos.");
  });
}

```



Figura 29. Control en la Pantalla de Lista de Favoritos

Elaborado por: Diego Chicaiza

#### 4.5.3.4. Crear una pantalla para enlistar los productos

La pantalla de lista de productos con los siguientes objetivos:

- Se puede ver el detalle de un producto.
- se puede agregar a la lista de Favoritos.
- Se puede agregar a la cesta de compra.
- Se puede filtrar por categoría
- Se puede filtrar por nombre del producto en un campo de búsqueda.





Figura 30. Pantalla - Lista de Productos

Elaborado por: Diego Chicaiza

#### 4.5.4. Iteración 4

Iteración	Numero	Historia de usuario	Tiempo estimado	
			Días	Horas
4	016	Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras	6	12
4	017	Crear la pantalla para la forma de pago con PayPal	6	12
4	018	Crear una pantalla para mostrar las compras realizadas	5	10
		Total	17	34

##### 4.5.4.1. Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras

En la pantalla que se muestra en la Figura 31, se puede observar los productos que fueron añadidos al carrito de compra, en la misma se puede eliminar un ítem específico o también permite visualizar los detalles del producto.

Cuanta con opciones en la parte inferior para seguir agregando productos y para realizar el pago por medio de PayPal.

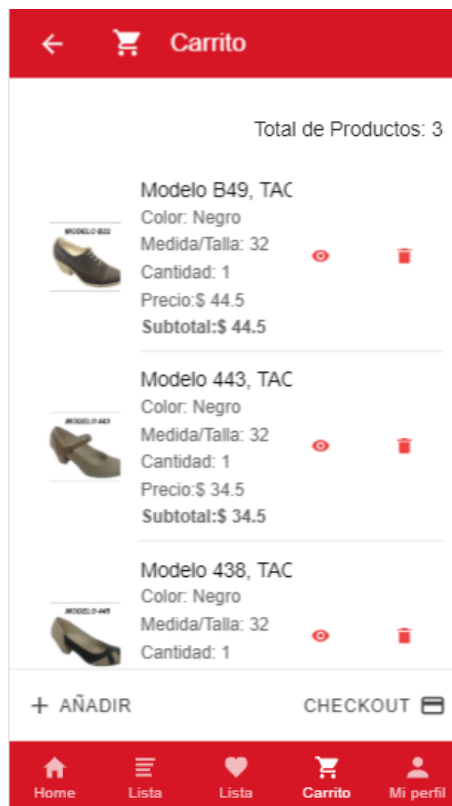


Figura 31. Pantalla - Carrito de compras

Elaborado por: Diego Chicaiza

#### 4.5.4.2. Crear la pantalla para la forma de pago con PayPal

Cuando se selecciona el botón CHECKOUT se dirige a la pantalla de Métodos de Pago que en el proyecto se especificó que se utilizara PayPal. Una vez optado la forma de pago se realiza el proceso de autenticación para posteriormente confirmar la compra de los productos.

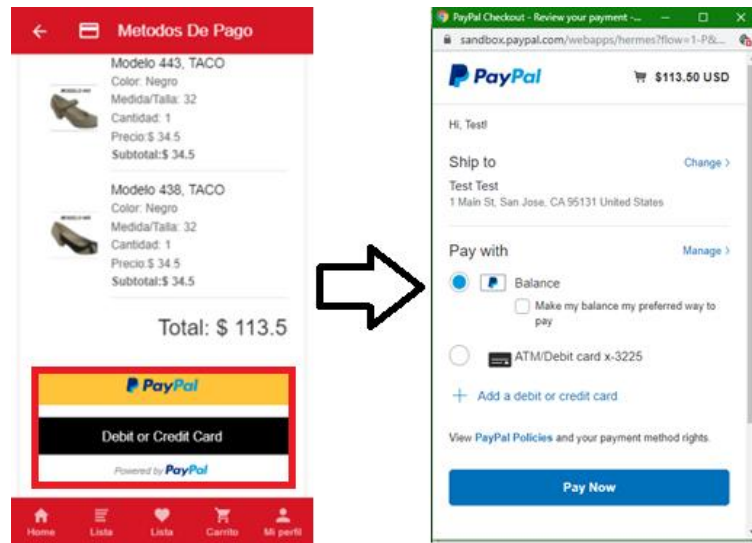


Figura 32. Proceso de pago

Elaborado por: Diego Chicaiza

#### 4.5.4.3. Crear una pantalla para mostrar las compras realizadas

La pantalla para mostrar las compras realizadas se las puede acceder desde el menú lateral izquierdo con el nombre de Ordenes. En esta pantalla se muestra clasificadas las ordenes que el usuario ha realizado.

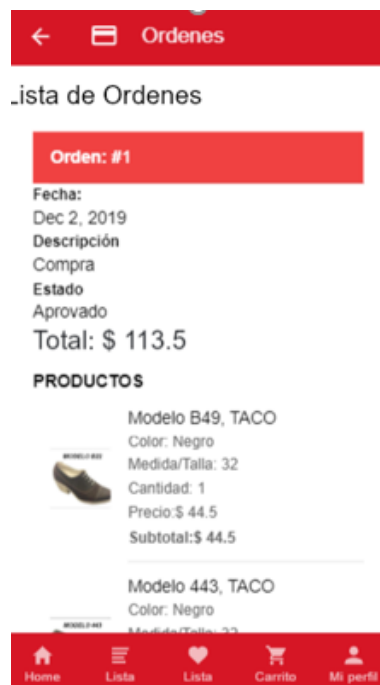


Figura 33. Pantalla – Ordenes

Elaborado por: Diego Chicaiza.

#### 4.5.5. Iteración 5

Iteración	Numero	Historia de usuario	Tiempo estimado	
			Días	Horas
5	019	Subir la aplicación al hosting de Firebase	3	6
5	020	Subir la aplicación a la Google Play	12	24
		<b>Total</b>	15	30

##### 4.5.5.1. Subir la aplicación al hosting de Firebase

Firebase Hosting proporciona hosting seguro y rápido para las aplicaciones web, el contenido dinámico y estático, y los micro servicios[45]. Adicionalmente en su Plan Spark (Sin cargo) ofrece para hospedar aplicaciones 1 GB de almacenamiento, hasta 10 GB de transferencia por cada mes, varios sitios por proyecto, SSL gratuito. Para conseguir el despliegue de la aplicación son necesarios los siguientes pasos:

- Crear una cuenta en la Consola de Firebase
- Instalar los paquetes de Firebase tools, el siguiente comando ejecutar en una terminal o un símbolo de sistema.

```
npm install -g firebase-tools
```

- Una vez terminado el proyecto Ionic, en una terminal ejecutar `firebase login` para iniciar sesión.
- Es necesario agregar los paquetes de `@angular/pwa` al proyecto.

```
ng add @angular/pwa
```

- Una vez agregado el paquete `@angular/pwa` ejecutar `ionic build -- prod`, el directorio `www` estará listo para publicarse.
- Para desplegar la aplicación se ejecuta el siguiente comando.

```
firebase deploy
```

En las siguientes figuras (Figura 34, Figura 35) se muestran el proyecto desplegado en el Hosting de Firebase, y como se ve en un navegador respectivamente.

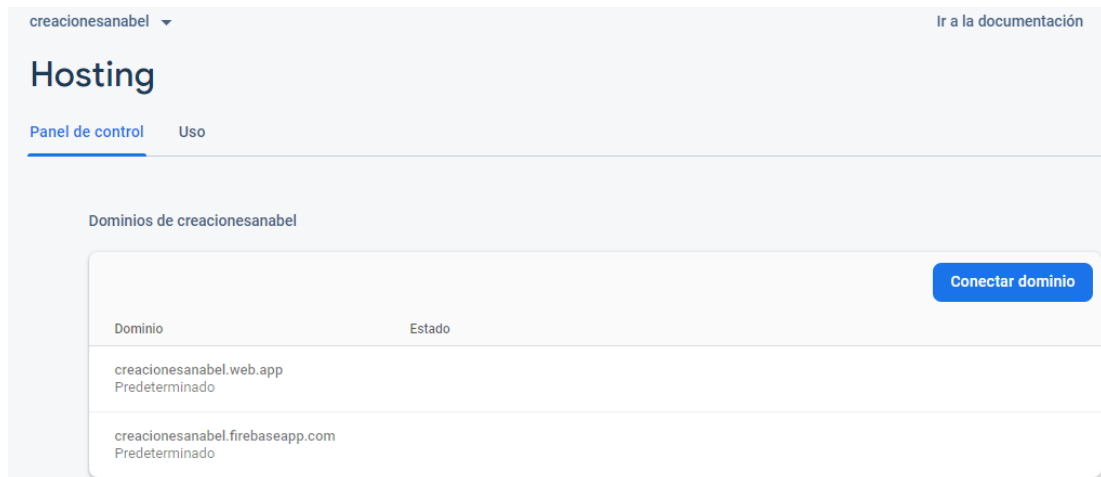


Figura 34. Administrador de sitios de Firebase

Elaborado por: Diego Chicaiza

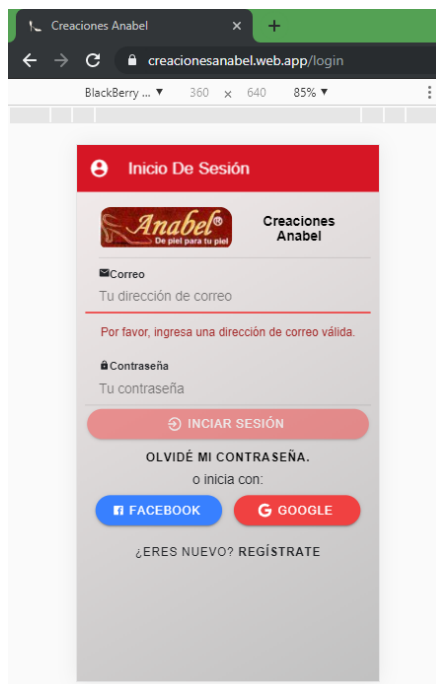


Figura 35. Aplicación desplegada en el Hosting de Google Play

Elaborado por: Diego Chicaiza.

#### 4.5.5.2. Subir la aplicación a la Google Play

Como requerimiento del cliente, la aplicación debe estar disponible para su uso en la tienda oficial para dispositivos Android como lo es Google Play.

En proceso de publicación de una aplicación en la Play Store se requiere una cuenta de desarrollador en Google Play Console, como recomendación de Google una app

debe subirse según Android App Bundle, que reemplaza al tradicional apk para optimizar los paquetes entre otras cosas.

Completar una ficha descriptiva de la aplicación, seleccionar idioma, agregar el nombre de la app, agregar la descripción de la aplicación, agregar imágenes tanto como promocionarles y las del icono si tiene videos que describan la aplicación, categorías entre otros. Una vez completado todos los datos descriptivos y de contacto se procede a subir un archivo ejecutable (apk o abb) con la respectiva firma del equipo de desarrollo, cabe mencionar que la firma es de suma importancia ya que con esta firma se realizara las futuras actualizaciones.

Una vez finalizada el proceso de publicación se debe esperar los días necesarios para que la comunidad de Google la apruebe, dando como resultado la siguiente como muestra de publicación.

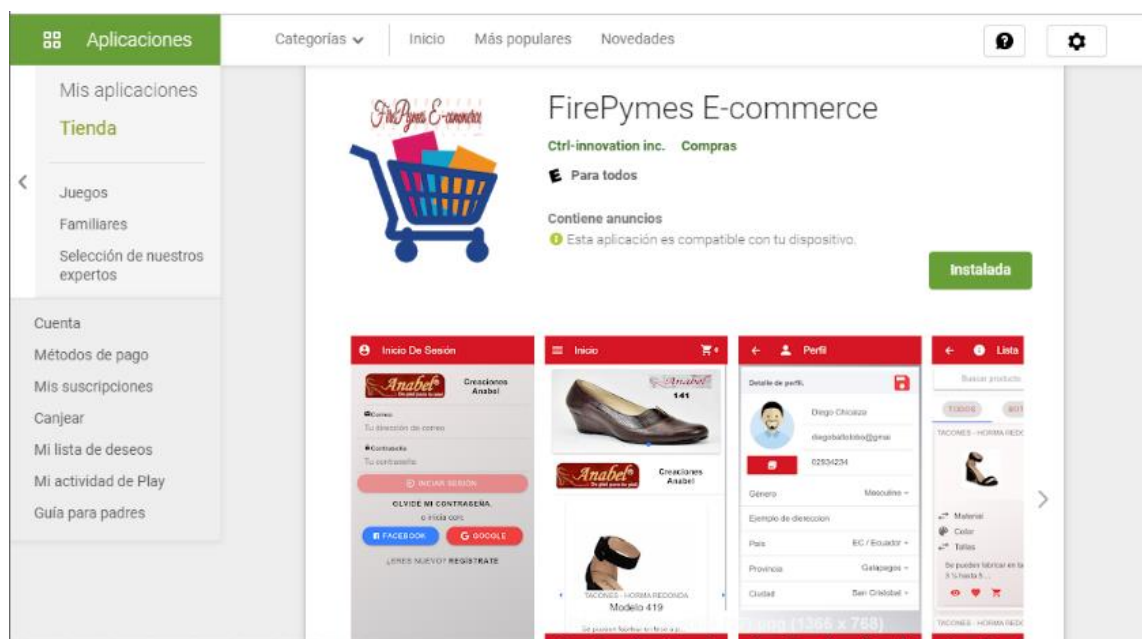


Figura 36. Publicación en la tienda Google Play Store

Elaborado por: Diego Chicaiza

#### 4.5.6. Codificación

- **Métodos de autenticación**

Para los métodos de autenticación, se realizó con correo electrónico y contraseña, inicio de sesión con Facebook o Google. Para las redes sociales se podrá iniciar tanto nativamente instalando la aplicación o por el ambiente web publicado en Firebase.

- Método de inicio de sesión con correo y contraseña.

```

SignIn(email, password) {
  return this.afAuth.auth
    .signInWithEmailAndPassword(email, password)
    .then(result => {
      this.ngZone.run(() => {
        if (result.user.emailVerified === false) {
          this.router.navigate(["/verify-email"]);
        } else {
          this.router.navigate(["/home"]);
          this.localStoreUser();
        }
      });
      this.SetUserData(result.user);
    })
    .catch(error => {
      this.SignOut();
      this.presentAlert(
        "Advertencia",
        this.errMessService.getErrorById("EC4").name,
        this.errMessService.getErrorById("EC4").solution
      );
    });
}

```

Figura 37. Método de inicio de sesión con correo y contraseña

Elaborado por: Diego Chicaiza.

- **Inicio de sesión con Google**

En el caso de los inicios de sesión con redes sociales como lo es Google se desarrolló tanto como para la parte nativa una vez agregada la plataforma y generada la aplicación y otra que puede ser utilizada en la parte web. Se detalla a continuación.

```

doGoogleLogin() {
  if (this.platform.is("cordova")) {
    this.nativeGoogleLogin();
  } else {
    let provider = new auth.GoogleAuthProvider();
    this.AuthLogin(provider);
  }
}

```

Figura 38. Inicio con red social con Google nativo y popup para navegadores

Elaborado por: Diego Chicaiza

```
async nativeGoogleLogin(): Promise<void> {
  try {
    await this.googlePlus
      .login({
        webClientId:
          "xxxxxx.apps.googleusercontent.com",
        offline: true,
        scopes: "profile email"})
      .then(response => {
        console.log(response);
        const googleCredential = auth.GoogleAuthProvider.credential(
          response.idToken);
        auth()
          .signInWithCredential(googleCredential)
          .then(user => {
            console.log(user);
            this.SetUserData(user);
            this.router.navigate(["/home"]);
          });
      });
  } catch (err) {
    console.log(err);
  }
}
```

Figura 39. Inicio de sesión con Google en ambiente nativo.

Elaborado por: Diego Chicaiza

Para el inicio de sesión con Facebook se debe crear y habilitar una aplicación como desarrollador (Facebook Developers), se detalla la implementación del método.

```
doFacebookLogin() {
  return new Promise<User>((resolve, reject) => {
    if (this.platform.is("cordova")) {
      this.fb.login(["public_profile"]).then( response => {
        const fbCredential = auth.FacebookAuthProvider.credential(
          response.authResponse.accessToken );
        auth().signInWithCredential(facebookCredential).then(user => {
          this.SetUserData(user);
          this.router.navigate(["/home"]);
        });
      });
    }, err => reject(err)
  );
}
```



```

    } else {
      this.afAuth.auth.signInWithPopup(new auth.FacebookAuthProvider()).
then( result => {
  console.log(result);
  this.router.navigate(["/home"]);
  var bigImgUrl = "https://graph.facebook.com/...."
  auth().currentUser.updateProfile({
    displayName: result.user.displayName,
    photoURL: bigImgUrl})
    .then( res => {
      this.SetUserData(result.user);},
      err => { reject(err); });
    }, err => {
  reject(err);
  });
  }
});
}

```

Figura 40. Inicio de sesión con Facebook

Elaborado por: Diego Chicaiza

- **Cerrar sesión**

El usuario al cerrar sesión puede ingresar con otra cuenta.

```

SignOut() {
  return this.afAuth.auth
    .signOut()
    .then(
      () => {
        localStorage.removeItem("user");
        this.router.navigate(["/login"]);
      },
      error => {}
    )
    .catch(err => {
      console.log(err);
      console.log("ocurrió un error");
      this.router.navigate(["/login"]);
    });
}

```

Figura 41. Cerrar sesión

Elaborado por: Diego Chicaiza

- **Método para recuperar la cuenta**

```
ForgotPassword(passwordResetEmail) {
  return this.afAuth.auth
    .sendPasswordResetEmail(passwordResetEmail)
    .then(() => {
      this.presentAlert(
        "",
        "",
        this.errMessService.getMessageById("MS1").name
      );
      this.router.navigateByURL("/login");
    })
    .catch(error => {
      this.presentAlert(
        "Error",
        "",
        this.errMessService.getErrorById("EC5").name
      );
    });
}
```

- **Método para verificación de la cuenta.**

```
// Send email verification when new user sign up
SendVerificationMail() {
  return this.afAuth.auth.currentUser
    .sendEmailVerification()
    .then(() => {
      this.toastCtrl.showToast(
        this.errMessService.getMessageById("MS2").name
      );
      this.router.navigate(["/verify-email"]);
    })
    .catch(error => {
      console.log(error.code);
      if (error.code == "auth/too-many-requests") {
        this.toastCtrl.showToast(
          this.errMessService.getErrorById("EC7").name +
            " " +
            this.errMessService.getErrorById("EC7").solution
        );
      }
    });
}
```

- **Servicio para operaciones de la colección products.**

Para todo proceso que requiere datos de Firestore se crearon servicios dependiendo de cada colección, en este caso para listar los productos el método es el siguiente.

### Obtener todos los productos

```
export class ProductService {
  private productsCollection: AngularFirestoreCollection<ProductId>;
  ...
  constructor(private _afs: AngularFirestore) {
    this.productsCollection = this._afs.collection<ProductId>('products')
  }
  getAllProducts(){
    return this.productsCollection.snapshotChanges().pipe(map(
      actions => actions.map(a => {
        const data = a.payload.doc.data() as ProductId;
        const id = a.payload.doc.id;
        return { id, ...data };
      })
    ));
  }
}
```

Figura 42. Método para listar productos

Elaborado por: Diego Chicaiza

- Añadir a la lista de favoritos.

```
addFavouriteProduct(data: ProductId): void {
  let a: ProductId[] = [];
  a = JSON.parse(localStorage.getItem("avf_item")) || [];
  if (a.length <= 0) {
    a.push(data);
    this.toast.showToast("Producto añadido a favoritos.");
    localStorage.setItem("avf_item", JSON.stringify(data));
  } else {
    let value = -1;
    for (let index = 0; index < a.length; index++) {
      const element = a[index];
      if (element.id == data.id) {
        value = index;
        break;
      }
    }
  }
}
```

```

if (value == -1) {
  a.push(data);
  localStorage.setItem("avf_item", JSON.stringify(data));
  this.toast.showToast("Producto añadido a favoritos.");
} else {
  this.toast.showToast(
    "El producto ya se encuentra añadido a favoritos."
  );
}
}
setTimeout(() => {
  localStorage.setItem("avf_item", JSON.stringify(a));
  this.calculateLocalFavProdCounts();
}, 1500);
}

```

Figura 43. Añadir a la lista de Favoritos almacenados en LocalStorage

Elaborado por: Diego Chicaiza.

En la vista de lista de productos se lo implementa de la siguiente manera, utilizando el método mencionado en la Figura 42. Tanto como la lógica(ts) y como la vista(html)

```

showProducts() {
  this.loadFirst = true;
  this.productService
    .getCategories()
    .subscribe(cat => (this.categories = cat));
  console.log(this.route.snapshot);
  if (this.route.snapshot.data["special"]) {
    const data = this.route.snapshot.data["special"];
    console.log(data);
    let filterCat: CategoryAuxiliar = data;
    this.productService.searchByCategory(filterCat).subscribe(result
=> {
      console.log(result);
    });
  } else
  this.productService.getAllProducts().pipe(
    finalize(async () => {
      // Hide the loading spinner on success or error
      await this.loading.dismiss();
    })
  ).subscribe(data => (this.products = data));
}

```

Figura 44. Mostrar productos en la vista .ts

Elaborado por: Diego Chicaiza

```
<div *ngFor="let product of products" class="card">

  <div class="card-body">
    <h3>{{ product.title }}</h3>
    <p> {{ product.description }}</p>
    <a href="{{article.url}}">Ver </a>
    <ion-button (click)="addCart"> Agregar al carrito</ion-button>
  </div>
</div>
<div *ngFor="let product of products" class="card">

  <div class="card-body">
    <h3>{{ product.title }}</h3>
    <p> {{ product.description }}</p>
    <a href="{{article.url}}">Ver </a>
    <ion-button (click)="addCart(product)"> Agregar</ion-button>
    <ion-button (click)="addFavotirite(product)"> Agregar</ion-button>
  </div>
</div>
```

Figura 45. Ejemplo para mostrar los productos en la vista .html

Elaborado por: Diego Chicaiza

El método de pago que se utilizó fue PayPal, se puede realizar la compra de los productos tanto en cualquier navegador accediendo desde la url, que se muestra en la Figura 35 o instalando la aplicación en un sistema operativo Android. Cabe mencionar que el proyecto aún está configurado con la cuenta de pruebas o Sandbox de PayPal.

```
payWithPaypal() {
  this.payPal
    .init({
      PayPalEnvironmentProduction: "",
      PayPalEnvironmentSandbox: config.paypal_client_id
    })
    .then(
      () => {
        this.payPal
          .prepareToRender(
            "PayPalEnvironmentSandbox",
            new PayPalConfiguration({
            })
          )
        .then(
```



```

        () => {
            // Error in initialization, maybe PayPal isn't supported or something else
            this.toastService.showToast(
                "Error en la inicialización, es posible que PayPal no es disponible o algo similar."
            );
        }
    );
}

```

Tabla 39. Implementación de método de Pago PayPal

Elaborado por: Diego Chicaiza

#### 4.6. Fase V: Pruebas

El objetivo es verificar el funcionamiento correcto, el rendimiento, aceptación y usabilidad. La fase de pruebas ayuda a ambas partes tanto como desarrollador como cliente o usuario final ante posibles fallos de funcionamiento y de esta manera saber si cumple y satisface las necesidades de cada historia de usuario.

Pruebas de aceptación: Su objetivo es validar el correcto funcionamiento de un sistema, esperando un resultado esperado para el usuario final para su aceptación.

Para llevar a cabo las mismas se utilizó las pruebas de caja negra, los resultados se muestran en las pruebas de aceptación para cada historia de usuario. De los mismo se obtiene resultados favorables.

<b>Prueba de aceptación</b>	
<b>Número:</b> 1	<b>N° Historia de usuario:</b> 004
<b>Nombre:</b> Ingreso al sistema por correo y contraseña	
<b>Descripción:</b> La aplicación al ser un sistema de venta de productos (calzado para dama), se requiere que exista un módulo de autenticación por medio de correo electrónico y contraseña. El usuario puede iniciar sesión para poder realizar todas las operaciones que la misma tenga.	
<b>Condiciones de ejecución:</b> Se mostrará una pantalla para ingresar sus credenciales correo electrónico y la contraseña respectivamente.	
<b>Entrega:</b> La pantalla de inicio de sesión tiene varios campos, los que debe utilizar para validar esta historia de usuario es el campo de correo, contraseña y el botón Iniciar Sesión	

<p><b>Resultado esperado:</b> Si el usuario ingresa correctamente sus credenciales se mostrará una pantalla de inicio con varios productos.</p>
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria.</p>

Tabla 40. Caso de prueba - Inicio de sesión con correo electrónico y contraseña

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 2	<b>N° Historia de usuario:</b> 005
<b>Nombre:</b> Registrar usuario por correo y contraseña	
<b>Descripción:</b> Los clientes podrán registrar sus datos para poder hacer uso de la aplicación sin restricción alguna.	
<b>Condiciones de ejecución:</b> Los clientes en el caso de no iniciar sesión o no registrarse en la aplicación no podrán hacer uso de los módulos. Una vez registrado se enviará un correo para que confirme la cuenta creada.	
<b>Entrega:</b> En la pantalla de registro cuenta con varios campos, los que se debe utilizar para validar esta historia de usuario es el campo nombre de usuario, correo, contraseña y el botón de regístrate.	
<b>Resultado esperado:</b> Si los datos son correctos se envía un correo de confirmación para verificar la cuenta creada, tras validar la cuenta se puede iniciar sesión.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Tabla 41. Caso de prueba - Registrar usuario

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 3	<b>N° Historia de usuario:</b> 006
<b>Nombre:</b> Restablecer contraseña	
<b>Descripción:</b> Los clientes pueden restablecer la contraseña ingresando el correo electrónico con el que se registraron.	
<b>Condiciones de ejecución:</b> El usuario podrá recuperar su contraseña ingresando el correo con el que creo la cuenta en la Pantalla Recuperar Cuenta.	
<b>Entrega:</b> En la Pantalla Recuperar Cuenta Figura 19, debe ingresar el correo electrónico y presionar sobre el botón Recuperar cuenta.	



<b>Resultado esperado:</b> Se enviará un correo a la bandeja de entrada del correo electrónico seleccionado, se debe acceder a la dirección y cambiar la contraseña.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 42. Caso de prueba - Recuperar cuenta.

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 4	<b>N° Historia de usuario:</b> 007
<b>Nombre:</b> Inicio de sesión y registro por redes sociales (Google)	
<b>Descripción:</b> Los clientes pueden ingresar a la aplicación por medio de una cuenta Google para hacer uso total de la aplicación.	
<b>Condiciones de ejecución:</b> En las pantallas de Inicio de sesión y registro se cuenta con botones adicionales para ingresar por las redes sociales de Google y Facebook.	
<b>Entrega:</b> Se debe hacer clic sobre el botón Google	
<b>Resultado esperado:</b> Se abrirá una nueva ventana para aceptar el inicio de sesión.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Tabla 43. Caso de prueba - Inicio de sesión con Google

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 6	<b>N° Historia de usuario:</b> 009
<b>Nombre:</b> Crear una pantalla de productos favoritos	
<b>Descripción:</b> Cada producto agregado a favoritos solo se mostrará con la posibilidad de agregar al carrito o a la vez eliminar de favoritos	
<b>Condiciones de ejecución:</b> Se debe acceder a la lista favoritos por el menú lateral o por los accesos directos de la parte inferior de cada pantalla.	
<b>Entrega:</b> Una vez seleccionado la lista de productos se mostrará cada producto añadido al mismo, dentro de las opciones se puede agregar al carrito de compras o se puede ver los detalles de un producto en particular.	
<b>Resultado esperado:</b> Al hacer clic sobre el botón del carrito se añade uno y se repite el proceso se sigue sumando uno en uno. Se controla la existencia del producto al agregar al carrito. Al hacer clic sobre el botón de detalle se enlaza a otra ventana donde muestra todos los detalles de un producto en particular.	

**Evaluación de la prueba:**

Prueba satisfactoria.

Tabla 44. Caso de prueba - Pantalla de lista de productos favoritos.

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 7	<b>N° Historia de usuario:</b> 010
<b>Nombre:</b> Crear una pantalla para editar los datos del perfil de usuario.	
<b>Descripción:</b> En esta pantalla el usuario autenticado podrá agregar más detalles de su información personal.	
<b>Condiciones de ejecución:</b> En la pantalla Perfil de usuario, se puede modificar el nombre de usuario, número telefónico o celular, la dirección o Domicilio, el género, país, provincia, cantón o ciudad.	
<b>Entrega:</b> Cada campo se puede modificar, tomar en consideración que el correo no se debe modificar el correo electrónico. Una vez ingresado los campos estén listos hacer clic en el botón Guardar.	
<b>Resultado esperado:</b> Al seleccionar los campos de país las provincias se demoran en cargar de 3 a 5 segundos.	
<b>Evaluación de la prueba:</b> Prueba aceptable, se requiere de algún campo de que está cargando las provincias.	

Tabla 45. Caso de prueba - Actualizar los detalles del usuario autenticado

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 8	<b>N° Historia de usuario:</b> 012
<b>Nombre:</b> Crear una pantalla para mostrar los detalles de un producto.	
<b>Descripción:</b> Se debe crear una pantalla para que el usuario pueda ver los detalles de un producto seleccionado para posteriormente agregar a favoritos o al carrito de compras.	
<b>Condiciones de ejecución:</b> En la pantalla principal o lista de productos se cuenta con un botón para ver todos los detalles de un producto en particular.	
<b>Entrega:</b> Al hacer clic sobre el botón ver detalle, se muestra toda la información de un producto en particular.	

<p><b>Resultado esperado:</b> Se muestra toda la información de un producto seleccionado, permite agregar al carrito y a la lista de favoritos.</p>
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria.</p>

Tabla 46. Caso de prueba - ver detalles de un producto seleccionado.

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 9	<b>N° Historia de usuario:</b> 015
<b>Nombre:</b> Crear una pantalla para enlistar los productos.	
<b>Descripción:</b> En esta pantalla se puede ver todos los productos, implementar un método de búsqueda para los artículos. Los mismos permitirán guardar a favoritos, ver el detalle del producto.	
<b>Condiciones de ejecución:</b> Todos los usuarios podrán ver los productos que dispone la empresa.	
<b>Entrega:</b> En la pantalla Lista de productos cuenta con varias opciones, se puede buscar por el nombre del producto en el cuadro de texto, se puede filtrar por categoría del producto. Dentro de cada producto cuenta con opciones como agregar al carrito de compras, agregar a favoritos o ver el detalle completo del producto.	
<b>Resultado esperado:</b> Se muestra todos los productos, permite filtrar por categoría y buscar un producto determinado por el nombre.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Tabla 47. Caso de prueba - Listar productos.

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 10	<b>N° Historia de usuario:</b> 016
<b>Nombre:</b> Crear una pantalla de enlistar los productos que fueron agregados a la canasta de compras o también conocido como carrito de compras.	
<b>Descripción:</b> En esta pantalla se debe listar los productos que fueron agregados al carrito, se debe verificar la existencia. Se debe realizar el cálculo correspondiente para observar el total de la compra.	

<b>Condiciones de ejecución:</b> Todos los usuarios pueden observar los productos que estarán listos para proceder con la compra.
<b>Entrega:</b> Cuando se selecciona un producto al carrito se verifica la existencia del mismo y de no tener más de lo ingresado informa al cliente que no puede añadir más de ese producto.
<b>Resultado esperado:</b> Se muestra los productos agregados al carrito con los siguientes detalles, nombre, cantidad, material, precio, color, tamaño.
<b>Evaluación de la prueba:</b> Prueba satisfactoria.

Tabla 48. Caso de prueba - Lista de productos en el carrito

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 2	<b>N° Historia de usuario:</b> 017
<b>Nombre:</b> Crear la pantalla para la forma de pago	
<b>Descripción:</b> En esta pantalla se confirmará y seleccionara cual es la forma de pago que desea el cliente	
<b>Condiciones de ejecución:</b> Todos los usuarios pueden acceder a esta pantalla previa autenticación.	
<b>Entrega:</b> Cuando se añade al menos un producto al permite continuar con el proceso.	
<b>Resultado esperado:</b> En la pantalla de Métodos de Pago al seleccionar el botón PayPal se abre una ventana para ingresar las credenciales de la cuenta PayPal, una vez autenticado se muestra el valor a pagar, para confirmar se debe dar clic en Pagar con PayPal.	
<b>Evaluación de la prueba:</b> Prueba satisfactoria.	

Tabla 49. Caso de prueba - Lista de productos a comprar.

Elaborado por: Diego Chicaiza.

<b>Prueba de aceptación</b>	
<b>Número:</b> 2	<b>N° Historia de usuario:</b> 018
<b>Nombre:</b> Crear una pantalla para mostrar las compras realizadas	
<b>Descripción:</b> En esta pantalla se mostrará las compras que el cliente ha realizado.	
<b>Condiciones de ejecución:</b> Todos los usuarios pueden ver esta información.	

<p><b>Entrega:</b> En el menú lateral izquierdo ir a la opción Ordenes.</p>
<p><b>Resultado esperado:</b> Se muestra todos los productos que se han comprado.</p>
<p><b>Evaluación de la prueba:</b> Prueba satisfactoria.</p>

Tabla 50. Caso de prueba - Mostrar las compras realizadas de un cliente.

Elaborado por: Diego Chicaiza.

La aplicación se realizó directamente con el gerente de la empresa Creaciones Anabel, dando como resultado una aplicación funcional con expectativas de seguir mejorando. Adicionalmente se encuentran publicadas en el hosting que provee Firebase y la tienda de Google Play(Ver Anexo B.2).

## CAPÍTULO V

### CONCLUSIONES Y RECOMENDACIONES

#### 5.1. Conclusiones

En la actualidad varias empresas sean estas pequeñas o medianas, se preocupan por la manera de como promocionar y expandirse en el mercado, de hacer conocer la marca, los productos, servicios entre otros. El proyecto FirePymes E-commerce nace en base a esa necesidad, enfocada específicamente al calzado, se pretende que varias empresas que están es el entorno del calzado se interesen.

La investigación constante de las herramientas utilizadas, la colaboración que tuvo el cliente, el enfoque de la metodología XP, permitieron cumplir los objetivos planteados y la implementación de la aplicación. La metodología XP, permitió que los cambios que se daban durante cada iteración no afectasen significativamente en el proyecto.

El Framework de Ionic en su versión 4 y plugins adicionales permitieron optimizar y reutilizar el código, así como La librería de Firebase permitió realizar las operaciones CRUD, como se especificó el proyecto la plataforma seleccionada fue Android. Adicionalmente la aplicación es una PWA (Aplicación Web Progresiva).

La empresa Creaciones Anabel a integrarse al entorno tecnológico, por medio de la aplicación FiresPymes E-commerce ha podido ver que se puede gestionar las ventas, ordenes, manejo de existencia y costo de los productos, para ello solo basta tener acceso a Internet.

En el momento de realizar las respectivas pruebas de validación con el usuario final se descubrieron debilidades de la aplicación, errores que no se percataron como desarrollador, como lo es entrada de datos, acciones que tardan. Adicionalmente, en la parte nativa (Android) en distintas versiones del sistema operativo de la aplicación no contralaban ciertos eventos como por ejemplo el back button o tecla atrás.

## **5.2. Recomendaciones**

Para la estructura de base de datos se recomienda tomar de los modelos ya existentes como woocommerce API Docs para modificar a la conveniencia propia o aumentar más campos de una colección. Adicionalmente se podría aumentar más módulos y que los mismos se adapten a la parte financiera de una empresa.

Se recomienda para los estudiantes que investiguen nuevas tecnologías, en este caso Frameworks de desarrollo como Ionic, Angular, entre otros, que ayudan a los desarrolladores con la estructura de proyecto que es de suma importancia.

Se recomienda promover y dictar clases a los estudiantes de la Carrea de Ingeniería en Sistemas Computacionales e Informáticos que se formen, ya que por lo general los contenidos de las mallas cambian con poca frecuencia.

Se recomienda utilizar un controlador de versiones, por ejemplo, GitLab, que permite respaldar el proyecto en la nube y evitar pérdida de información o cambios, o volver a restaurar el proyecto lo hace posible.

## Bibliografía

- [1] E. Andrade, “E-COMMERCE EN ECUADOR: Estado actual y sus perspectivas de crecimiento.”
- [2] N. Sohrabi Safa and M. Akmar Ismail, “A customer loyalty formation model in electronic commerce,” 2013.
- [3] D. Cabrera, “Análisis y aplicación de estrategias de E-Commerce en un portal web para PyMEs,” pp. 1–204, 2012.
- [4] CRDM, “Internet móvil 46,4%,” *ARCOTEL/INEC; Elabor. CRDM*, p. 2017, 2016.
- [5] M. G. V. Toapanta, “E-COMMERCE Y SU IMPACTO EN EL DESARROLLO ECONÓMICO DE LAS PEQUEÑAS Y MEDIANAS EMPRESAS DE LA PROVINCIA DE TUNGURAHUA,” 2016.
- [6] D. Ramírez, “Social Media Marketing como Estrategia de Ventas para la Empresa My Shoes de la Ciudad de Ambato,” 2015.
- [7] D. Ramírez, “TEMA :‘Social Media Marketing como Estrategia de Ventas para la Empresa My Shoes de la Ciudad de Ambato,’” p. 166, 2015.
- [8] M. Coque and H. Dalgo, “‘Los Canales de Distribución y su incidencia en las Ventas de la Empresa de calzado Bettini de la ciudad de Ambato,’” 2011.
- [9] L. A. Muñoz Mora and R. Villalba, “Estrategias de Marketing y su incidencia en las ventas de la empresa de calzado Gabriel de la ciudad de Ambato,” 2011.
- [10] A. M. Ruiz, “Diseño e Implementación de un aula,” 2016.
- [11] P. D. CUMBA ARMIJOS and B. A. BARRENO PILCO, “‘ANÁLISIS DE PYTHON CON DJANGO FRENTE A RUBY ON RAILS PARA DESARROLLO ÁGIL DE APLICACIONES WEB. CASO PRÁCTICO: DECH.’” 2012.
- [12] jiMMy Molina ríos and M. zea ordoñez, “METODOLOGÍAS DE DESARROLLO EN APLICACIONES WEB,” 2017.
- [13] CARLOS XAVIER RAMÍREZ MUÑOZ, *DESARROLLO DE UNA APLICACIÓN HÍBRIDA-MÓVIL PARA DETERMINAR LOS TIPOS DE ESPECIES DE VEGETACIÓN INDICADORAS SEGÚN LOS PISOS ZOOGEOGRÁFICOS DEL ECUADOR, MEDIANTE EL USO DE LAS HERRAMIENTAS PHONEGAP Y JQUERY MOBILE PARA LA PLATAFORMA ANDROID*, vol. 49, no. 23–6. 2015.
- [14] Á. M. VALBUENA APONTE and C. A. MENESES ESCOBAR, “GUÍA COMPARATIVA DE FRAMEWORKS PARA LOS LENGUAJES HTML 5, CSS Y JAVASCRIPT PARA EL DESARROLLO DE APLICACIONES WEB,” pp. 561–565, 2014.
- [15] L. Vivas, M. Cambarieri, N. G. Martínez, H. M. Abbate, and M. Petroff,



- “Desarrollo de Aplicaciones Multiplataforma con HTML5 para Teléfonos Inteligentes.”
- [16] D. Lisandro, N. Galdamez, P. Thomas, L. Corbalan, and P. Pesado, “Análisis Experimental de desarrollo de Aplicaciones Móviles Multiplataforma,” *XX Congr. Argentino Ciencias la Comput. (Buenos Aires, 2014)*, p. 10, 2014.
- [17] Á. A. Depablos Zanchetta, “ACTUALIZACIÓN TECNOLÓGICA Y DESARROLLO DE APLICACIÓN PARA DISPOSITIVOS MÓVILES BASADOS EN ANDROID DEL LABORATORIO REMOTO DE CONTROL,” 2015.
- [18] L. Yaguapaz, “UNIVERSIDAD TÉCNICA DEL NORTE Facultad de Ingeniería en Ciencias Aplicadas Carrera de Ingeniería en Sistemas Computacionales ESTUDIO DEL FRAMEWORK IONIC 2 PARA EL DESARROLLO DE APLICACIONES MÓVILES HIBRIDAS Trabajo de grado previo a la obtención del título,” 2018.
- [19] J. Alarcon, “PhoneGap o Apache Cordova ¿qué diferencia hay? | campusMVP.es.” [Online]. Available: <https://www.campusmvp.es/recursos/post/PhoneGap-o-Apache-Cordova-que-diferencia-hay.aspx>. [Accessed: 27-Apr-2019].
- [20] X. Yin, J. Zhang, and X. Wang, “Sequential injection analysis system for the determination of arsenic by hydride generation atomic absorption spectrometry,” *Fenxi Huaxue*, vol. 32, no. 10, pp. 1365–1367, 2004.
- [21] T. Navarrete, “El lenguaje JavaScript,” *Fonaments Cartogr. i SIG*, p. 11, 2007.
- [22] K. W. Tracy, “Mobile application development experiences on Apples iOS and Android OS,” *IEEE Potentials*, vol. 31, no. 4, pp. 30–34, 2012.
- [23] I. Corporation, “El Desarrollo De Aplicaciones Moviles Nativas,” p. 10, 2016.
- [24] A. L. O. Espinoza, Luis Enrique, Rosa de Guadalupe Cano, “Inteligencia de negocios + comercio electrónico = PYMES eficaces,” vol. 5, 2007.
- [25] C. J. Rodríguez Andrade and C. F. Aldas Flores, “Aplicación móvil para la reserva, pagos de servicio de transporte y envío de encomiendas de los clientes de la empresa Romero y Asociados Amaseur de la ciudad de Ambato,” 2017.
- [26] H. Gracia and O. Yanes, “Bases de datos NoSQL,” 2012.
- [27] M. Castello, A. González, and A. López, “Desarrollo de una aplicación Android de apuestas utilizando Firebase para la sincronización de datos.”
- [28] Firebase, “Firebase.” [Online]. Available: <https://firebase.google.com/pricing/?hl=es>. [Accessed: 03-May-2019].
- [29] “What is Visual Studio Code? Why is it used? - Quora.” [Online]. Available: <https://www.quora.com/What-is-Visual-Studio-Code-Why-is-it-used>. [Accessed: 26-Apr-2019].
- [30] “Git - Acerca del control de versiones.” [Online]. Available: <https://git-scm.com/book/es/v1/Empezando-Acerca-del-control-de-versiones>. [Accessed:

26-Apr-2019].

- [31] “Why Millions of Developers Choose GitKraken.” [Online]. Available: <https://www.gitkraken.com/customers>. [Accessed: 12-Dec-2019].
- [32] “Acerca | Node.js.” [Online]. Available: <https://nodejs.org/es/about/>. [Accessed: 12-Dec-2019].
- [33] Y. D. Amaya Balaguera, “Metodologías ágiles en el desarrollo de aplicaciones para dispositivos móviles,” *Rev. Tecnol. / J. Technol.*, vol. 12 número, pp. 111–124, 2013.
- [34] H. V. Cevallos, “Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software,” 2018.
- [35] L. E. D. CARMONA, LUIS MIGUEL ECHEVERRY. TOBÓN, “CASO PRÁCTICO DE LA METODOLOGÍA ÁGIL XP AL DESARROLLO DE SOFTWARE,” *Tesis*, vol. 67, no. 6, pp. 14–21, 2007.
- [36] E. H. Uribe and L. E. V. Ayala, “Del Manifiesto Ágil, sus valores y sus principios,” *Redalyc.Uaemex.Mx*, no. 34, pp. 381–385, 2007.
- [37] Kent Beck *et al.*, “Manifiesto por el Desarrollo Ágil de Software.” [Online]. Available: <https://agilemanifesto.org/iso/es/manifesto.html>. [Accessed: 19-Apr-2019].
- [38] L. Montoya Suarez, J. Sepulveda Castaño, and L. Maria Jiménez Ramos, “Análisis comparativo de las metodologías ágiles en el desarrollo de software aplicadas en Colombia,” 2016.
- [39] C. Cuenca and M. Juliana, ““ Desarrollo de una aplicación móvil Android para la búsqueda de plazas disponibles en un parqueadero ,”” p. 159, 2015.
- [40] GRUPO GARATU, “Que es la metodología Scrum y por qué la usamos en Garatu Development.” [Online]. Available: <https://development.grupogaratu.com/metodologia-scrum-desarrollo-software/>. [Accessed: 21-Apr-2019].
- [41] W. A. ZAMBRANO GARCÍA and L. F. BORJA BORJA, “ESTUDIO COMPARATIVO DE METODOLOGÍAS DE DESARROLLO ÁGIL EN BASE AL DESARROLLO DE UNA APLICACIÓN MÓVIL.”
- [42] calzadoecuador, “Creaciones Anabel.” [Online]. Available: <http://www.calzadoecuador.com/talleres/71-creaciones-anabel>. [Accessed: 25-Apr-2019].
- [43] I. Figueroa, “Breve Guía Buenas Prácticas de Programación en C.”
- [44] C. Fu, “Build Mobile Apps with Ionic 2 and Firebase: Hybrid Mobile App Development,” pp. 47–55, 2017.
- [45] “Firebase Hosting | Firebase.” [Online]. Available: <https://firebase.google.com/docs/hosting/?hl=es-419>. [Accessed: 29-Nov-2019].



## **ANEXOS Y APÉNDICES**

## Anexo A

### A.1. Cuestionario elaborado para la obtención de información

La presente encuesta se la realizo conjunto al Dr. Luis Pintado, gerente propietario de Creaciones Anabel. Al ser un cuestionario con

1	¿Su empresa cuenta con una página en la cual sus clientes ven los productos que ofrece?
2	¿Quiénes son los usuarios?
3	¿Tiene los usuarios experiencia con aplicación informática?
4	¿Cuáles son sus expectativas con respecto a la facilidad de uso de esta aplicación?
5	¿Cuáles son sus expectativas con respecto al tiempo de entrenamiento?
6	¿Ha utilizado medios para vender a través de internet?
7	¿Cuál es la estrategia que le funciona para promocionar sus productos a través de Internet?
8	¿Tiene interés de vender sus productos mediante el uso de internet?
9	¿Considera usted que el comercio electrónico (e-commerce) ayude a la empresa a crecer en el mercado?
10	El sistema operativo Android cubre con una gran cobertura de clientes, ¿le gustaría que la aplicación se pruebe con estos usuarios?
11	¿Con que medios prefiere que sus clientes adquieran sus productos? • PayPal • Tarjeta de crédito • Dinero en efectivo • Transferencia deposito en efectivo
12	¿Estaría dispuesto a invertir para llevar a cabo la aplicación a su correcto funcionamiento?
13	¿Tiene un diagrama de procesos de proceso de venta u organización? ¿Si la tiene detalle?
14	¿Qué entregables o productos produce?
15	¿Cómo se categorizan Sus productos?
16	¿Ofrece descuentos en sus productos a sus clientes para que los adquieran con facilidades?
17	¿Qué productos están en mejor posicionamiento en el mercado para sobresalir entre sus competidores?

Tabla A. 1. Cuestionario utilizado para la entrevista  
Elaborado por: Diego Chicaiza.

## Anexo B

### B.1. Manual de usuario

En este apartado se detalla las funciones de la aplicación en un manual de usuario. Las formas de acceder a la aplicación.

### B.2. Formas de acceder a la aplicación

#### B.2.1. Desde un navegador web.

Abrir un dispositivo con conexión a Internet abrir un navegador web, acceder al siguiente enlace: <https://creacionesanabel.firebaseio.com>

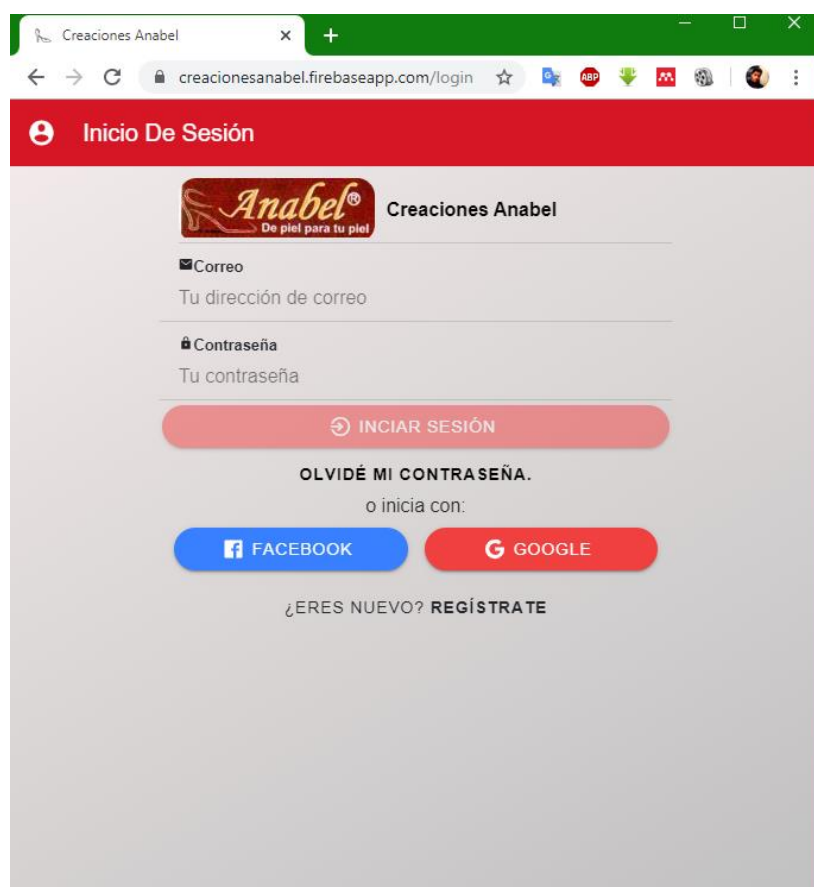


Figura A. 1. Acceso desde un navegador web  
Elaborado por: Diego Chicaiza

#### B.2.2. Descargando la aplicación de la Play Store.

En la tienda de Google Play Store buscar la aplicación FirePymes E-commerce descargar e instalarla. En el caso de no encontrar acceder mediante el siguiente enlace:

<https://play.google.com/store/apps/details?id=com.pymesolution.fisei>

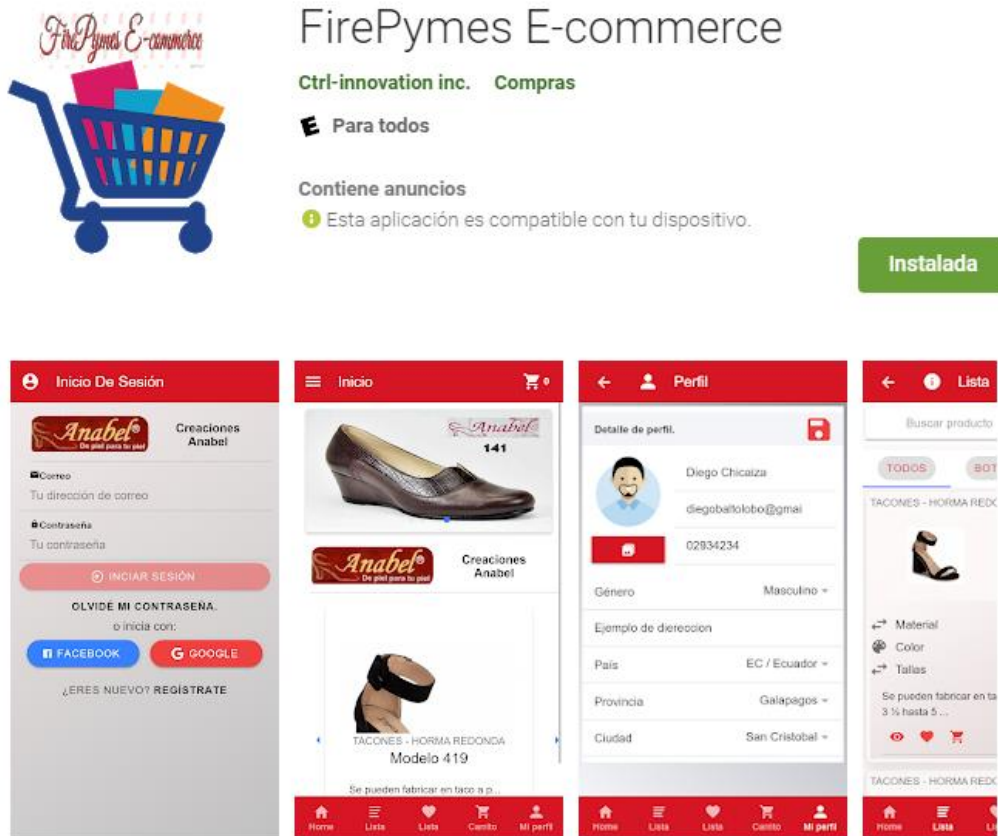


Figura A. 2. Aplicación subida a Play Store.  
Elaborado por: Diego Chicaiza.

### B.3. Requerimientos

Los requerimientos mínimos para que la aplicación nativa funcione sin inconvenientes son los siguientes:

- Sistema operativo Android mínimo versión 4.4
- Procesador de 400MHz
- Memoria mínima de 256 Mg
- Conectividad a Internet.
- Para el acceso Web solo se requiere conexión a Internet y navegadores actualizados.

### B.3.1. Ingreso a la aplicación



Figura A. 3. Ingreso a la aplicación por primera vez.  
Elaborado por: Diego Chicaiza.

Detalle del proceso:

Para ingresar a la aplicación se tiene varias formas según el ambiente donde se esté utilizando, si es nativo es decir en un Android nos permite ingresar por correo y contraseña y si ingresa por un navegador web permite ingresar por redes sociales como lo son Google y Facebook, adicional puede ingresar con el correo y contraseña.

1. Se debe ingresar el correo electrónico con el que creo la cuenta.
2. Se debe ingresar la contraseña.
3. Mediante el botón verifica si existe en la base de datos, verifica si la cuenta ha sido verificada, comprobado ello permite acceder a los demás módulos.
4. En el caso de que se olvide la contraseña se puede reenviar un correo para que notifique a la bandeja de entrada.
5. Si no se tiene una cuenta mediante este enlace se puede crear.
6. Se puede ingresar con la red social preferida, esto está disponible para el entorno web.



### B.3.2. Registro de usuario

The image shows a mobile application registration screen for 'Anabel'. The header is red with a back arrow, a user icon, and the text 'Regístrate'. Below the header is the 'Anabel' logo with the tagline 'De piel para tu piel' and the text 'Creaciones Anabel'. The registration form consists of several fields: 'Nombre de usuario' (step 1), 'Correo' (step 2), 'Contraseña' (step 3), and 'Confirmar contraseña' (step 4). Below these fields is a large red button labeled 'REGÍSTRATE' (step 5). Underneath the button is a separator '- O -'. Below the separator are two buttons for social media login: 'FACEBOOK' (step 6) and 'GOOGLE' (step 6). At the bottom, there is a link that says '¿TIENES CUENTA? INICIA SESIÓN' (step 7).

Figura A. 4. Registrar usuarios  
Elaborado por: Diego Chicaiza

Detalle del proceso:

Los usuarios que no tengan una cuenta la pueden crear desde la pantalla de inicio de sesión Figura A. 3 en el ítem 5.

1. Se debe ingresar el nombre de usuario.
2. Se debe ingresar el correo electrónico con la cual se desee registrar.
3. Se debe ingresar la contraseña.
4. Confirmar la contraseña.
5. Mediante el botón REGÍSTRATE verifica si existe en la base de datos, verifica si la cuenta ha sido verificada, comprobado ello permite acceder a los demás módulos.
6. Permite ingresar por redes sociales (desde un navegador web).
7. Permite regresar al inicio de sesión Figura A. 3.

### B.3.3. Verificación de la cuenta

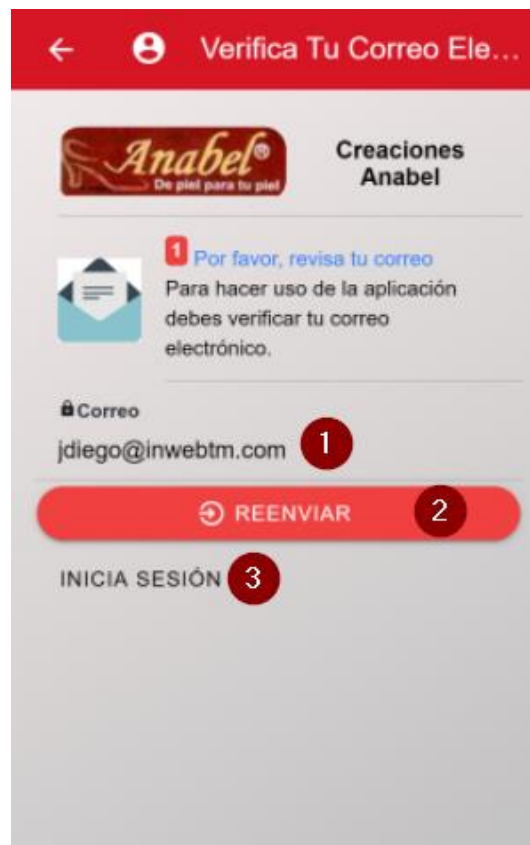


Figura A. 5. Verificar cuenta  
Elaborado por: Diego Chicaiza.

Detalle del proceso:

Una vez creada la cuenta, inmediatamente se envía un correo electrónico para que se verifique la cuenta, hasta no verificar la cuenta no podrá acceder a los productos, entre otras cosas.

1. Automáticamente se establece el correo que se registró.
2. Permite reenviar el mensaje en caso de que no llegue a la bandeja de entrada.
3. Permite regresar a inicio de sesión Figura A. 3.

Cabe mencionar que si ya se envió no sería necesario este paso.

### B.3.4. Recuperar cuenta



Detalle del proceso:

1. Ingresar el correo que se desee recuperar la cuenta.
2. Una vez comprobado que es correcto el correo se debe hacer clic sobre el botón RESTABLECER LA CONTRASEÑA.
3. Permite regresar a la pantalla de inicio de sesión.

### B.3.5. Pantalla principal



Figura A. 6. Pantalla principal  
Elaborado por: Diego Chicaiza.

Detalle del proceso:

La presente pantalla presenta accesos directos a las distintas rutas o pantallas.

1. Accede al menú lateral
2. Accede al carrito de compras.
3. Muestra los productos que se desplazan automáticamente cada cierto tiempo.
4. Ir o regresar a la pantalla principal
5. Ir a la pantalla de Lista de productos
6. Ir a la pantalla de Lista de favoritos
7. Ir a la pantalla de Carrito de compras.
8. Ir a la pantalla de edición de Mir perfil

### B.3.6. Menú lateral

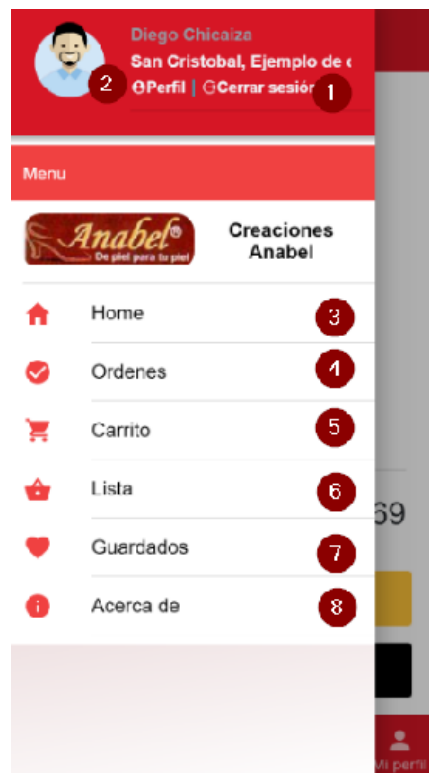


Figura A. 7. Menú lateral  
Elaborado por: Diego Chicaiza.

Detalle de proceso:

Una vez verificado la cuenta e iniciado sesión en la parte superior se puede acceder al menú lateral o deslizando de izquierda a derecha, se cuenta con las siguientes opciones:

1. Cerrar sesión, no permite navegar hasta que se vuelva a iniciar sesión
2. Muestra otra pantalla del perfil de usuario.
3. Regresa a la pantalla principal.
4. Ordenes, permite observar las compras efectuadas.
5. Carrito, permite observar cuantos productos han sido agregados para comprar.
6. Lista, permite ver cuantos productos dispone en la base de datos.
7. Guardados, o también conocidos como productos favoritos.
8. Acerca de, permite ver datos adicionales.

### B.3.7. Lista de productos

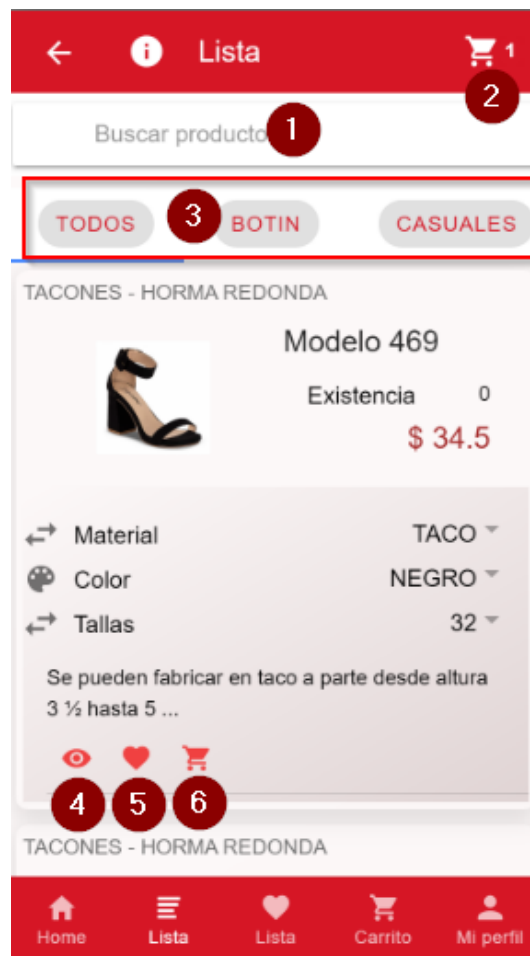


Figura A. 8. Lista de productos.  
Elaborado por: Diego Chicaiza.

Detalle del proceso:

1. Cuadro de texto para buscar el producto por nombre.
2. Ir a la pantalla de Carrito de compras.
3. Botones para filtrar los productos por categoría.
4. Ir a la pantalla de Detalle del producto.
5. Agrega a la Lista de favoritos controla que no se repita el producto.
6. Agrega al Carrito de compra, verifica la existencia del producto.

### B.3.8. Detalle del Producto

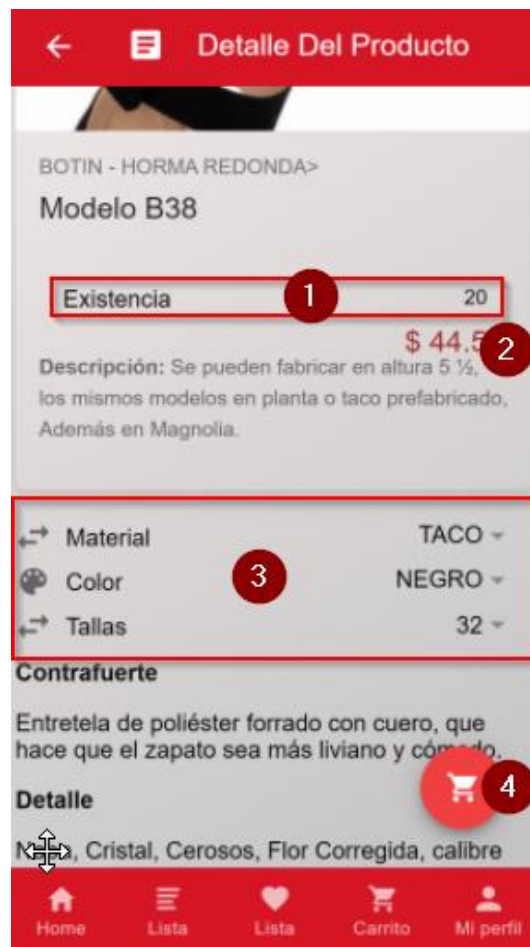


Figura A. 9. Detalle del Producto  
Elaborado por: Diego Chicaiza.

Detalle del proceso:

- 1- Muestra la existencia del producto.
- 2- Muestra el precio del producto según los campos del ítem 3.
- 3- Permite seleccionar respectivamente sus opciones como el material, el color y la talla.
- 4- Permite agregar el producto al carrito de compras si tiene existencia.

### B.3.9. Edición de Perfil de Usuario



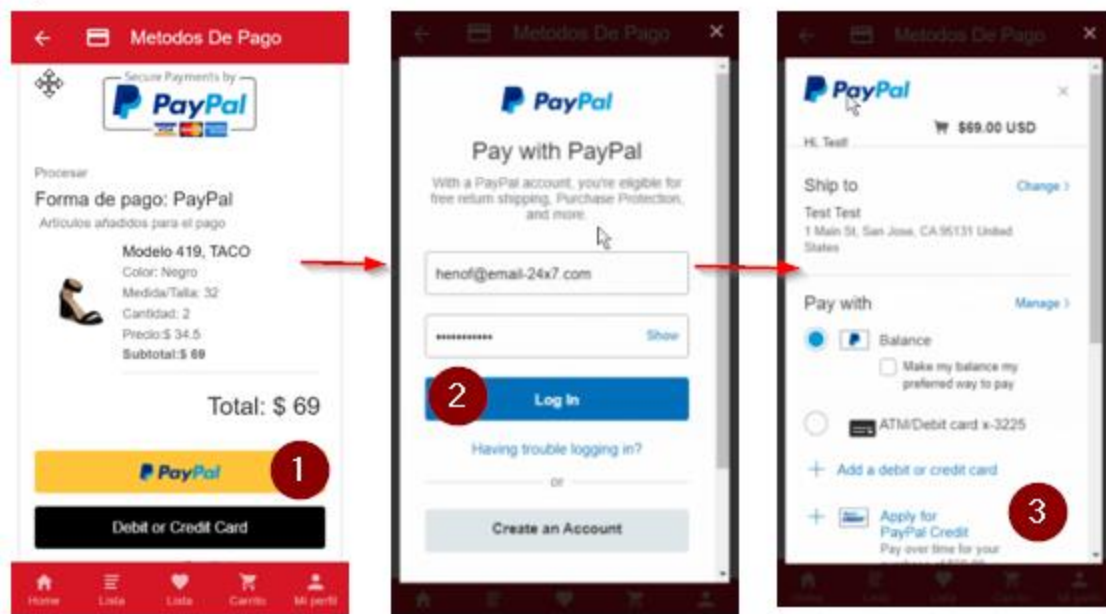
Figura A. 10. Edición de usuario.  
Elaborado por: Diego Chicaiza.

Detalle del proceso:

1. Actualiza los datos que se cambien en los campos enumerados.
2. Editar el nombre de usuario.
3. Correo electrónico, no se debe actualizar, aunque permite.
4. Editar el teléfono convencional o celular.
5. Seleccionar el género del usuario.
6. Seleccionar el país.
7. Seleccionar la provincia.
8. Seleccionar el cantón o ciudad.
9. Permite cambiar el avatar del usuario, se cargan los que estén configurados en la base de datos.



### B.3.10. Proceso de pago



Detalle del proceso:

- Una vez agregado los productos al carrito de compras, en la misma permite ver o eliminar el producto. Puede seguir agregando productos o pagar con PayPal
- 1. Seleccionar la opción de pago.
- 2.- Ingresar con la cuenta de PayPal.
- 3. Confirmar pago.

## Anexo C

### C.1. Servicios consumidos de Firebase

#### C.1.1. Almacenamiento de datos en Firestore

Firebase proporciona varios servicios, entre ellos tiene una base de datos, para el presente proyecto se utilizó Cloud Firestore.

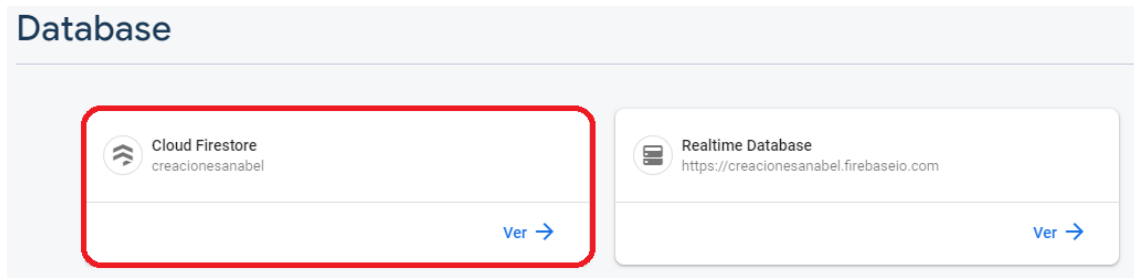


Figura A. 11. Base de datos seleccionada  
Elaborado por: Diego Chicaiza  
Fuente: [28]

Adicionalmente cuenta con en la pantalla principal del proyecto un resumen de todo lo que se va utilizando de cada servicio que provee Firebase, para el proyecto se utilizó lo siguiente:

- Hosting
- Firestore
- Authentication

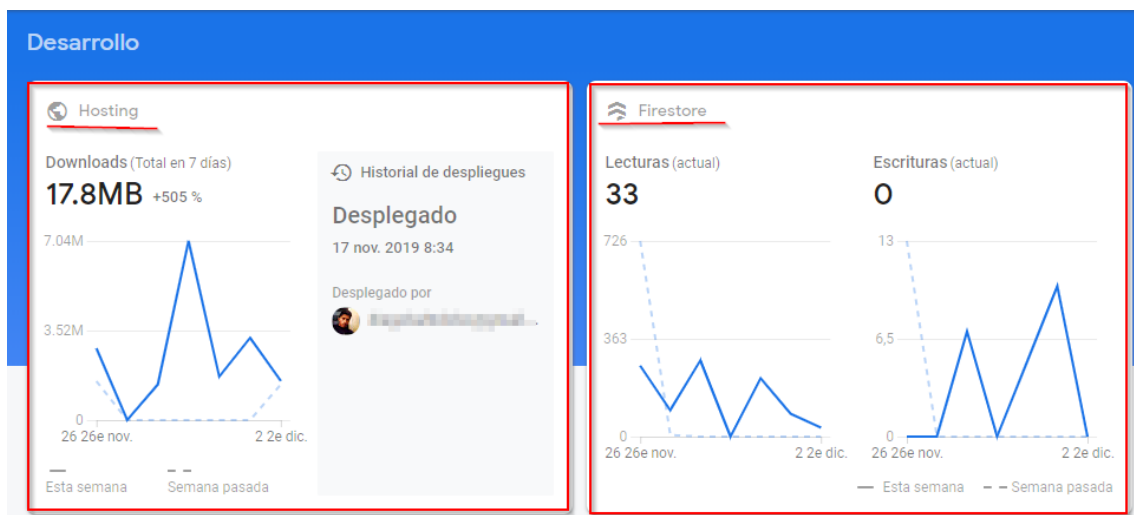
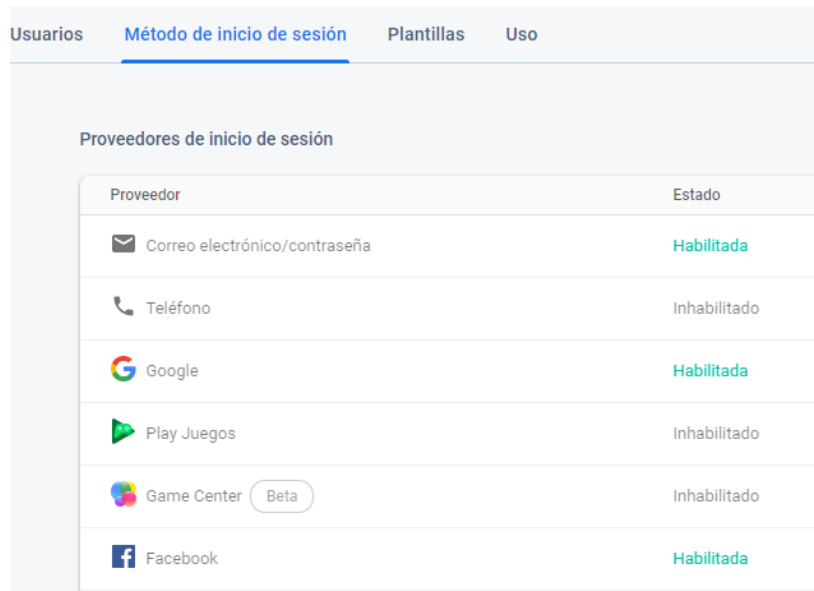


Figura A. 12. Resumen de consumo de servicios Firebase  
Elaborado por: Diego Chicaiza  
Fuente: [45]

### C.1.2. Autenticación

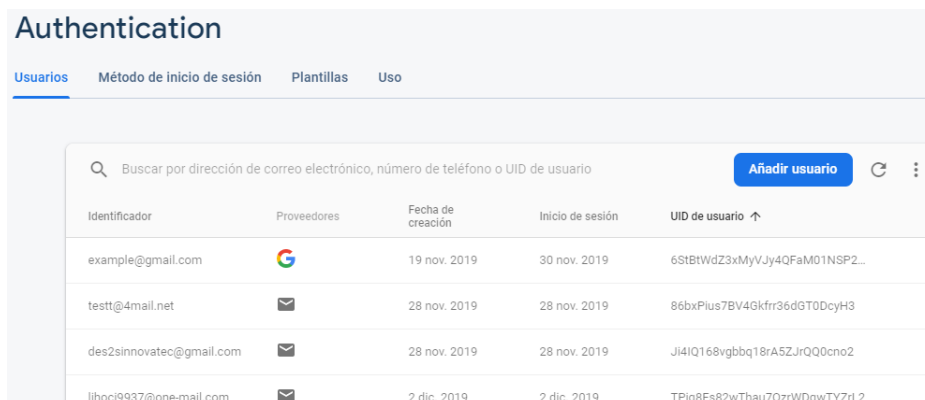
Firestore proporciona también la facilidad de autenticación, evitando así a los desarrolladores gastar tiempo en este proceso que es esencial para muchas aplicaciones. A continuación, se detalla los métodos de inicio de sesión que se implementó en la aplicación.



Proveedor	Estado
Correo electrónico/contraseña	Habilitada
Teléfono	Inhabilitado
Google	Habilitada
Play Juegos	Inhabilitado
Game Center (Beta)	Inhabilitado
Facebook	Habilitada

Figura A. 13. Métodos de autenticación Firebase  
Elaborado por: Diego Chicaiza  
Fuente: [45]

El resultado de iniciar sesión o registrarse se puede administrar dentro del módulo de Authentication de Firebase en el apartado Usuarios. Todos los usuarios mostrados son de prueba.



Identificador	Proveedores	Fecha de creación	Inicio de sesión	UID de usuario ↑
example@gmail.com	Google	19 nov. 2019	30 nov. 2019	6StBtWdZ3xMyVJy4QFaM01NSP2...
testt@4mail.net	Correo electrónico/contraseña	28 nov. 2019	28 nov. 2019	86bxPlus7BV4Gkfr36dGT0DcyH3
des2sinnovatec@gmail.com	Correo electrónico/contraseña	28 nov. 2019	28 nov. 2019	Ji4IQ168vgbbq18rA5ZJrQQ0cno2
lihoci9937@one-mail.com	Correo electrónico/contraseña	2 dic. 2019	2 dic. 2019	TPiq8Fs82wThau7OzrWDqwTYZrL2

Figura A. 14. Usuarios de ejemplo que han autenticado previamente.  
Elaborado por: Diego Chicaiza  
Fuente: [28]

## ANEXO D

### D.1. Versiones de herramientas utilizadas

#### D.1.1. Ionic

Para la realización del proyecto el Framework de Ionic utilizo lo siguiente, cada una se describe con las versiones. El comando para obtener el resultado es: ionic info

```
ionic (Ionic CLI) : 4.12.0 (C:\Users\DiegoDev\AppData\Roaming\npm\node_modules\ionic)
Ionic Framework : @ionic/angular 4.0.2
@angular-devkit/build-angular : 0.13.8
@angular-devkit/schematics : 7.2.4
@angular/cli : 7.2.4
@ionic/angular-toolkit : 1.4.0

Cordova:

cordova (Cordova CLI) : not installed
Cordova Platforms : not available
Cordova Plugins : cordova-plugin-ionic-keyboard 2.1.3, cordova-plugin-ionic-webview 3.1.2, (and 14 other plugins)

System:

Android SDK Tools : 26.1.1 (C:\Users\DiegoDev\AppData\Local\Android\Sdk)
NodeJS : v10.15.3 (C:\Program Files\nodejs\node.exe)
npm : 6.4.1
OS : Windows 10
```

Figura A. 15. Información Ionic  
Elaborado por: Chicaiza Diego.

#### D.1.2. Cordova

Para que la aplicación nativa funcione se utilizó varios plugins de Apache Cordova.

```
$ cordova info
cordova-lib@9.0.1 with:
  cordova-common@3.1.0
  cordova-create@2.0.0
  cordova-fetch@2.0.1
  cordova-serve@3.0.0

Environment:
  OS: win32
  Node: v10.15.3
  npm: 6.4.1

Plugins:
  card.io.cordova.mobilesdk
  com.paypal.cordova.mobilesdk
  cordova-plugin-android-permissions
  cordova-plugin-camera
  cordova-plugin-crop
  cordova-plugin-device
  cordova-plugin-facebook4
  cordova-plugin-file
  cordova-plugin-file-transfer
  cordova-plugin-firebase-authentication
  cordova-plugin-googleplus
  cordova-plugin-ionic-keyboard
  cordova-plugin-ionic-webview
  cordova-plugin-splashscreen
  cordova-plugin-statusbar
  cordova-plugin-telerik-imagepicker
  cordova-plugin-whitelist
  cordova-support-android-plugin
  cordova-support-google-services
```

Figura A. 16. Plugins Apache Cordova.  
Elaborado por: Diego Chicaiza.