



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

**CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES E
INFORMÁTICOS**

Tema:

**APLICACIÓN MÓVIL MULTIPLATAFORMA PARA LA GESTIÓN Y
CONTROL DE ENTREGAS DE LOS PORTEADORES DE LA EMPRESA
ESFLOSERV CIA. LTDA**

Trabajo de Titulación Modalidad: Proyecto de investigación, presentado previo a la obtención del título de Ingeniero en Sistemas Computacionales e Informáticos

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de Software

AUTOR: Anderson Javier Naranjo Guachambala

TUTOR: Ing. Dennis Vinicio Chicaiza Castillo, Mg

Ambato - Ecuador

septiembre - 2021

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el tema: APLICACIÓN MÓVIL MULTIPLATAFORMA PARA LA GESTIÓN Y CONTROL DE ENTREGAS DE LOS PORTEADORES DE LA EMPRESA ESFLOSERV CIA. LTDA, desarrollado bajo la modalidad Proyecto de Investigación por el señor Anderson Javier Naranjo Guachambala, 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, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, septiembre 2021

Ing. Dennis Vinicio Chicaiza Castillo, Mg.

TUTOR

AUTORÍA

El presente Proyecto de Investigación titulado: APLICACIÓN MÓVIL MULTIPLATAFORMA PARA LA GESTIÓN Y CONTROL DE ENTREGAS DE LOS PORTEADORES DE LA EMPRESA ESFLOSERV CIA. LTDA. 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, septiembre 2021



Anderson Javier Naranjo Guachambala

CC: 1805037619

AUTOR

APROBACIÓN TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final del Trabajo de Titulación presentado por el señor Anderson Javier Naranjo Guachambala, estudiante de la Carrera de de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado **APLICACIÓN MÓVIL MULTIPLATAFORMA PARA LA GESTIÓN Y CONTROL DE ENTREGAS DE LOS PORTEADORES DE LA EMPRESA ESFLOSERV CIA. LTDA**, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo con el Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora Presidenta del Tribunal.

Ambato, septiembre 2021.

Ing. Pilar Urrutia, Mg.
PRESIDENTA DEL TRIBUNAL

Ing. Leonardo Torres
PROFESOR CALIFICADOR

Ing. Fernando Ibarra
PROFESOR CALIFICADOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, septiembre 2021



Anderson Javier Naranjo Guachambala

CC: 1805037619

AUTOR

ÍNDICE GENERAL DE CONTENIDOS

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
APROBACIÓN TRIBUNAL DE GRADO	iv
DERECHOS DE AUTOR	v
ÍNDICE GENERAL DE CONTENIDOS	vi
ÍNDICE DE TABLAS	viii
ÍNDICE DE FIGURAS	x
CAPITULO I.- MARCO TEORICO	1
1.1 Tema de la Investigación.....	1
1.2 Antecedentes Investigativos.....	1
1.2.1 Contextualización del Problema	1
1.2.2 Fundamentación Teórica.....	3
1.2.2.1 Aplicaciones Móviles	3
1.2.2.2 Aplicaciones Nativas	3
1.2.2.3 Aplicaciones Multiplataforma	3
1.2.2.3.1 Aplicaciones Multiplataforma Híbridas.....	3
1.2.2.3.2 Aplicaciones Multiplataforma Interpretadas.....	4
1.2.2.4 Herramientas para el Desarrollo Multiplataforma.....	4
1.2.2.4.1 Ionic	4
1.2.2.4.1.1 Cordova.....	5
1.2.2.4.1.2 Capacitor	5
1.2.2.4.2 React Native	6
1.2.2.4.3 Flutter	6
1.2.2.5 API.....	6
1.2.2.5.1 SOAP(Protocolo de Acceso a Objetos simples)	7
1.2.2.5.2 REST.....	7
1.2.2.6 Seguridad en las API	8
1.2.2.6.1 Autorización.....	8
1.2.2.6.2 Autenticación	9
1.2.2.6.2.1 Autenticación básica.....	9
1.2.2.6.2.2 Autenticación basada en token	9
1.2.2.6.2.3 Autenticación basada en clave API.....	10
1.2.2.6.2.4 OAuth 2.0.....	10
1.2.2.7 Google Maps API.....	11
1.2.2.7.1 Directions API	11
1.2.2.7.2 Roads API	11

1.2.2.7.3 Maps JavaScript API.....	12
1.3 <i>Objetivos</i>	12
Objetivo General:.....	12
Objetivos Específicos:	12
CAPITULO II. METODOLOGÍA	13
2.1 <i>Materiales</i>	13
2.1.1 Humanos	13
2.1.2 Institucionales	13
2.1.3 Otros.....	13
2.1.4 Económicos (Presupuesto).....	13
2.2 <i>Métodos</i>	14
2.2.1 Modalidad de la investigación	14
2.2.2 Población y Muestra	15
2.2.3 Recolección de Información	15
2.2.4 Procesamiento y Análisis de Datos.....	16
2.2.4.1 Resultados de la Entrevista	16
CAPITULO III.- RESULTADOS Y DISCUSIÓN.	17
3.1 <i>Análisis y Discusión de resultados</i>	17
3.1.1 Metodologías de desarrollo ágil.....	17
3.1.2 Metodología Xtream Programin(XP).....	17
3.1.2.1 Roles de la metodología Xtream Programming	17
3.1.2.2 Etapas de la metodología Xtream Programmig.....	18
3.1.2.3 Ventajas y desventajas de la metodología XP	19
3.1.3 Metodología Scrum.....	19
3.1.3.1 Roles de la metodología Scrum	20
3.1.3.2 Características de la metodología Scrum	20
3.1.3.3 Ventajas y Desventajas de la metodología Scrum.....	21
3.1.4 Metodología Seleccionada.....	21
3.1.5 Tecnología de desarrollo multiplataforma elegida	22
3.1.6 Tecnología para el desarrollo de la API elegida	22
3.1.7 Uso de Google Maps API	23
3.1.8 Arquitectura de la Aplicación	24
3.2 <i>Desarrollo de la Propuesta</i>	24
3.2.1 Fase 1: Exploración	24
3.2.1.1 Levantamiento de la información.....	25
3.2.1.2 Análisis de Resultados	25
3.2.2 Fase 2: Planificación del Proyecto.....	27
3.2.2.1 Historias de usuario	27
3.2.2.2 Estimación de Historias de Usuario	32
3.2.2.3 Tareas	33
3.2.3 Fase 3: Iteraciones	38
3.2.3.1 Plan de Entregas	38

3.2.3.2 Iteraciones.....	39
3.2.4 Fase 4: Producción.....	40
3.2.4.1 Iteración 1.....	40
3.2.4.2 Iteración 2.....	44
3.2.4.3 Iteración 3.....	47
3.2.4.4 Iteración 4.....	50
3.2.5 Codificación.....	52
3.2.5.1 API.....	52
3.2.5.2 Aplicación IONIC	59
3.2.6 Fase 5: Pruebas	72
CAPITULO IV.-CONCLUSIONES Y RECOMENDACIONES	76
4.1 Conclusiones.....	76
4.2 Recomendaciones	77

ÍNDICE DE TABLAS

Tabla 1. Materiales Economicos (Presupuesto).....	13
Tabla 2 Población	15
Tabla 3 Plan de recolección de información.....	15
Tabla 4 Estructura Historia de Usuario.....	27
Tabla 5 Historia de Usuario 1	28
Tabla 6 Historia de Usuario 2	28
Tabla 7 Historia de Usuario 3	29
Tabla 8 Historia de Usuario 4	29
Tabla 9 Historia de Usuario 5	30
Tabla 10 Historia de Usuario 6	30
Tabla 11 Historia de Usuario 7	30
Tabla 12 Historia de Usuario 8	31
Tabla 13 Historia de Usuario 9	31
Tabla 14 Historia de Usuario 10	31
Tabla 15 Historia de Usuario 11	32
Tabla 16 Estimación de Historias de Usuario	32
Tabla 17 Tarea-Determinar la estructura del proyecto	33
Tabla 18 Tarea-Análisis de la base de datos	33
Tabla 19 Tarea-Análisis de la base de datos.....	33
Tabla 20 Tarea-Diseño y codificación de la pantalla de ingreso a la aplicación.....	34
Tabla 21 Tarea-Diseño de la pantalla de inicio de la aplicación.	34
Tabla 22 Tarea-Desarrollo módulo de paquetes de la API	34
Tabla 23 Tarea-Mostrado lista de paquetes.....	34
Tabla 24 Tarea-Buscador por información del paquete o cliente.	35
Tabla 25 Tarea-Buscador por código de barras.....	35
Tabla 26 Tarea-Diseño de pantalla para detalle de paquetes	35

Tabla 27 Tarea-Implementación de pantalla para detalle de paquetes	35
Tabla 28 Tarea-Diseño de pantalla para cambio de estado de los paquetes	36
Tabla 29 Tarea-Implementación de pantalla para cambio de estado de los paquetes.	36
Tabla 30 Tarea-Diseño de estructura de base de datos local.	36
Tabla 31 Tarea-Desarrollo de un servicio para el manejo de la base de datos local	36
Tabla 32 Tarea-Desarrollo del módulo para el envío de información almacenada en el dispositivo.	37
Tabla 33 Tarea-Desarrollo del módulo para recepción de información en la API	37
Tabla 34 Tarea-Integración con el API de Google Maps	37
Tabla 35 Tarea-Mostrar Marcadores de paquetes	37
Tabla 36 Tarea-Trasado de ruta sobre el mapa	38
Tabla 37 Tarea-Pruebas de aceptación	38
Tabla 38 Tarea-Carga de archivos al servidor principal de la empresa	38
Tabla 39 Plan de Entregas	38
Tabla 40 Planificación de Iteraciones	39
Tabla 41 Resumen Iteración 1	40
Tabla 42 Resumen Iteración 2	44
Tabla 43 Resumen Iteración 3	47
Tabla 44 Resumen Iteración 4	50
Tabla 45 Prueba de aceptación 1	72
Tabla 46 Prueba de aceptación 2	72
Tabla 47 Prueba de aceptación 3	73
Tabla 48 Prueba de aceptación 4	73
Tabla 49 Prueba de aceptación 5	73
Tabla 50 Prueba de aceptación 6	74
Tabla 51 Prueba de aceptación 7	74
Tabla 52 Prueba de aceptación 8	75

ÍNDICE DE FIGURAS

Figura 1 Arquitectura Ionic	5
Figura 2 Estructura del mensaje SOAP	7
Figura 3 Estructura JWT	10
Figura 4 Precio Ditections API	11
Figura 5 Precio Roads API.....	12
Figura 6 Estructura respuesta API	23
Figura 7 Arquitectura de la Aplicación	24
Figura 8 Instalación de IONIC	41
Figura 9 Generación del proyecto	41
Figura 10 Añadido de plataformas	42
Figura 11 Estructura del proyecto	42
Figura 12 Pantalla de Ingreso a la aplicación.....	44
Figura 13 Pantalla de inicio	45
Figura 14 Buscador de paquetes	45
Figura 15 Escaneo código de barras	46
Figura 16 Pantalla de detalle del paquete	47
Figura 17 Cambio de estado de paquete.....	48
Figura 18 Funcionamiento Offline.....	49
Figura 19 Menú Sincronización de Entregas	50
Figura 20 Marcadores de paquetes en Mapa.....	51
Figura 21 Marcaje de Ruta	52
Figura 22 Autenticación API	53
Figura 23 Creación del token.....	54
Figura 24 Validación del token	55
Figura 25 Encriptado de la información	56
Figura 26 Desencriptado de la información	56
Figura 27 Listado de paquetes	57
Figura 28 Listado de paquetes	58
Figura 29 Guardado de imágenes	59
Figura 30 Login.....	59
Figura 31 Verificar si existe usuario logeado	60
Figura 32 Encriptado IONIC.....	61
Figura 33 Desencriptado IONIC.....	61
Figura 34 Carga de paquetes en la base de datos local.....	62
Figura 35 Petición de paquetes a la API	62
Figura 36 Mostrado de información desde la base local.....	63
Figura 37 Sincronización de información	64
Figura 38 Buscador de Paquetes	65
Figura 39 Lector de código de barras.....	66
Figura 40 Obtener ubicación	66
Figura 41 Realizar llamada telefónica	67
Figura 42 Realizar fotografía	67
Figura 43 Seleccionar imagen de galería	68
Figura 44 Guardar nueva actualización	69
Figura 45 Cargar mapa y añadir marcadores.....	70
Figura 46 Marcar ubicación actual	70

Figura 47 Marcaje de ruta	71
Figura 48 Cerrar Sesión	71
Figura 49 Estado de la conexión.....	72

RESUMEN EJECUTIVO

El uso de aplicaciones móviles ha producido que estas se conviertan en herramientas de uso cotidiano cubriendo un amplio campo ya sea educación, salud, economía, etc. Pero el avance tecnológico ha obligado que diferentes empresas innoven su forma de trabajo además de ello podemos incluir la situación actual debido a la aparición del virus conocido como COVID-19, es por ello que las herramientas móviles se han tomado como fuentes para cumplir con las necesidades de los usuarios.

El presente proyecto de investigación tiene como perspectiva el desarrollo de una aplicación móvil multiplataforma centrada en la optimización del servicio de envío y entrega de encomiendas por parte de la empresa ESFLOSERV CIA. LTDA en vista del deseo de actualizar o mejor este servicio convirtiendo su metodología tradicional en moderna y sobresaliendo de otras empresas dedicadas a servicios similares.

El desarrollo de la aplicación móvil utilizando IONIC permite el ahorro de recursos y la implementación de lenguajes de programación como TypeScript con ergonomía aprovechando de manera segura y eficiente recursos con el fin de brindar una Aplicación móvil multiplataforma para la gestión y control de entregas de los portadores que cumplan a cabalidad los requerimientos de la empresa

La metodología seleccionada para la realización de este trabajo será Xtreme Programming o también conocida como XP debido a su alta aceptación y desenvolviéndose de forma excelente en el trabajo de desarrollo, además de ser una metodología que posee una comunicación entre usuarios y el grupo de desarrollo de la aplicación dando el apoyo necesario.

Palabras claves: IONIC, aplicación, optimización, servicio.

ABSTRACT

The use of mobile applications has caused them to become tools for everyday use covering a wide field of education, health, economics, etc. But technological progress has forced different companies to announce their way of working in addition to this we can include the current situation due to the appearance of the virus known as COVID-19, which is why mobile tools have been taken as sources to meet the needs of users.

In this research project, the perspective is the development of a mobile application focused on optimizing the service of sending and delivery of parcels by the company ESFLOSERV CIA. LTDA in view of the desire to update or improve this service by converting its traditional methodology into modern and exceling from other companies dedicated to similar services.

The development of the mobile application using IONIC allows the saving of resources and the implementation of programming languages such as TypeScript with ergonomics taking advantage of safely and efficiently resources in order to provide a multiplatform mobile application for the management and control of deliveries of the porters that fully meet the requirements of the company

The methodology selected for the realization of this work will be Xtreme Programming or also known as XP due to its high acceptance and performing excellently in the development work, in addition to being a methodology that has a communication between users and the development group of the application giving the necessary support.

Keywords: IONIC, application, optimization, service.

INTRODUCCIÓN

Como estructura para el desarrollo de nuestro proyecto de investigación con el tema “APLICACIÓN MÓVIL MULTIPLATAFORMA PARA LA GESTIÓN Y CONTROL DE ENTREGAS DE LOS PORTEADORES DE LA EMPRESA ESFLOSERV CIA. LTDA”, esta conformada por cinco temas puntuales los cuales describen de manera detallada el desarrollo de la aplicación móvil:

Capítulo I: “MARCO TEÓRICO”, de manera detallada se describe la situación problemática que esta cursando la empresa ESFLOSERV CIA. LTDA, tomando en cuenta la justificación de las diferentes razones por las cuales se está realizando la investigación, con ello también se establecerán objetivos generales y especificos que serán centro de nuestro desarrollo investigativo.

Capítulo II: “METODOLOGÍA”, son todas las etapas a ser tomadas en cuenta para el desarrollo de nuestro proyecto de investigación con esto también describiremos la modalidad de investigación y método de toma de información su procesamiento y análisis.

Capítulo III: “RESULTADOS Y DISCUSIÓN”, es la explicación detallada del desarrollo del proyecto de investigación, es decir las diferencias que existen en las metodologías que pueden ser utilizadas y con ello las herramientas que sean necesarias para la implementación y funcionamiento.

Capítulo IV: “CONCLUSIONES Y RECOMENDACIONES”, como su nombre lo indica en esta sección se nombran o describen ciertas conclusiones a las que llego el desarrollador además se realizan recomendaciones para trabajos posteriores las cuales deberán ser tomadas en cuenta.

CAPITULO I.- MARCO TEORICO

1.1 Tema de la Investigación

“Aplicación móvil multiplataforma para la gestión y control de entregas de los portadores de la empresa ESFLOSERV CIA. LTDA”

1.2 Antecedentes Investigativos

1.2.1 Contextualización del Problema

El crecimiento mundial en la adquisición de productos ha provocado que ciertas empresas evolucionen y lleguen a dedicarse exclusivamente a la entrega de productos en las viviendas de los clientes, a medida que los usuarios se hacen más exigentes; gigantes empresas como AMAZON se apegan a la filosofía que la logística es una herramienta estratégica que sirve para mejorar la competitividad de cualquier empresa, lo cual quiere decir que para que una empresa sea competitiva dentro de su mercado debe implementar gestiones que le permitan una buena administración de su almacén. Por esta razón Amazon trabaja constantemente en la conservación y gestión sistemática de mercancías en los almacenes, utilizando software que les permiten optimizar procesos de aprovisionamiento, estando a la vanguardia de sus competidores en las gestiones y operaciones de recepción, almacenamiento y movimiento de los productos.[1]

Otra de las empresas que día a día batalla con los problemas logísticos es SERVIENTREGA, empresa colombiana caracterizada por su eslogan “Servientrega es.... Entrega segura”, Servientrega apalanca la cadena de suministro del comercio electrónico, iniciando desde la logística internacional de aprovisionamiento de grandes volúmenes o productos unitarios, servicios de almacenamiento, que incluyen alistamiento, picking y packing, logística especializada e-commerce de recolección, distribución, cobro contra entrega y logística inversa, brindando acompañamiento e integración tecnológica que le permite conectarse con los servicios de rastreo y generación automática de guías.[2] Esta integración tecnológica con el rastreo y guías

de los diferentes paquetes ha logrado que sea la empresa en atender la mayor parte del mercado latinoamericano, ya que como se mencionó anteriormente es una de las empresas que más se fortalecen con tecnología de punta. Extendiendo así sus alcances de operaciones en logística y carga internacional brindando soluciones logísticas, ratificando así su eslogan “Servientrega Entrega Segura”. [3]

Es muy frecuente encontrar pequeñas y medianas empresas que enfrentan hábitos tradicionales que no les permiten avanzar y posicionarse en el mercado, por la carencia de una visión global, especialmente en los procesos logísticos, el mantener hábitos tradicionales es una de las mayores desventajas de este tipo de empresas, como es el caso de ESFLOSERV CIA. LTDA, a medida que gana mercado en las provincias de Tungurahua, Cotopaxi y Chimborazo aun no da un salto tecnológico importante para brindar un servicio de mayor calidad a sus usuarios.

En su mayoría la gestión logística en las entregas de los distintos paquetes se basa en documentación física, esto a medida que la cantidad de paquetes y repartidores aumenta en la empresa, genera una sobrecarga de trabajo en el personal administrativo encargado de gestionar y dar a conocer los reportes de entregas.

Al no contar rápidamente con reportes que detalles el estado actual de las entregas dificulta que se pueda tomar decisiones rápidas y acertadas sobre los problemas que puedan ocurrir durante el proceso de entrega de los distintos paquetes, problemas como el daño físico de los paquetes o direcciones incorrectas son comunes a la hora de realizar las entregas.

Desarrollar e implementar una aplicación destinada a gestionar las entregas de los porteadores, influirá significativamente en la forma en que la empresa adquiere los datos de las entregas, mismos datos que posteriormente podrán ser utilizados para tomar decisiones más rápidas y acertadas, permitiéndole a la empresa brindar un óptimo servicio a todos sus clientes.

1.2.2 Fundamentación Teórica

1.2.2.1 Aplicaciones Móviles

Las aplicaciones móviles son software desarrollado para ejecutarse en dispositivos como tabletas, teléfonos, relojes, televisores, radios, etc. Las aplicaciones móviles están diseñadas de acuerdo a las necesidades del usuario final, con el objetivo de facilitar varias de las tareas diarias.[4]

1.2.2.2 Aplicaciones Nativas

El desarrollo de aplicaciones nativas es una forma natural de implementar aplicaciones móviles. Las aplicaciones nativas están concebidas para ejecutarse en una plataforma específica, es decir, se debe considerar el tipo de dispositivo, el sistema operativo a utilizar y su versión.

Las aplicaciones nativas se desarrollan utilizando un entorno de desarrollo integrado (IDE), que proporciona las herramientas de desarrollo necesarias para crear y depurar aplicaciones. El código fuente se compila en código ejecutable y este proceso es similar al de una aplicación de escritorio tradicional.[5]

1.2.2.3 Aplicaciones Multiplataforma

El desarrollo multiplataforma se contrapone al nativo y se centra en la reutilización de código. Se procura optimizar la relación costo/beneficio compartiendo la misma codificación entre las versiones para las distintas plataformas. A continuación se muestra algunos tipos de desarrollo multiplataforma: [6]

1.2.2.3.1 Aplicaciones Multiplataforma Híbridas

Las aplicaciones híbridas son otro desarrollo multiplataforma basado en tecnologías Web (HTML, Javascript y CSS), no son ejecutadas por el navegador, por el contrario,

se ejecutan en un contenedor web especial y pueden acceder mejor al dispositivo a través de API específicas. Las aplicaciones híbridas permiten la reutilización del código en todas las plataformas, el acceso al hardware del dispositivo y la distribución a través de las tiendas de aplicaciones.[6]

1.2.2.3.2 Aplicaciones Multiplataforma Interpretadas

Se implementan en un lenguaje básico, la mayor parte del cual se traduce al código nativo, mientras que el resto se interpreta en tiempo de ejecución. Estas aplicaciones se implementan independientemente de las plataformas que utilizan diversas tecnologías y lenguajes como Javascript, Java, Ruby, XML, etc. Una de las principales ventajas de este tipo de aplicaciones es que puede obtener una interfaz de usuario completamente nativa. Sin embargo, los desarrolladores dependen completamente del entorno de desarrollo seleccionado. Los frameworks más famosos para generar estas aplicaciones son React Native y Flutter.[7]

1.2.2.4 Herramientas para el Desarrollo Multiplataforma

1.2.2.4.1 Ionic

Es una herramienta, gratuita y open source, para el desarrollo de aplicaciones híbridas basadas en HTML5, CSS y JavaScript. Está construido con Sass y optimizado con Angular. Está orientado al desarrollo de aplicaciones móviles híbridas. Debido a que Ionic es un marco de trabajo HTML5 necesita de un envoltorio nativo como Cordova o Capacitor para ejecutarse como una aplicación nativa.[8]

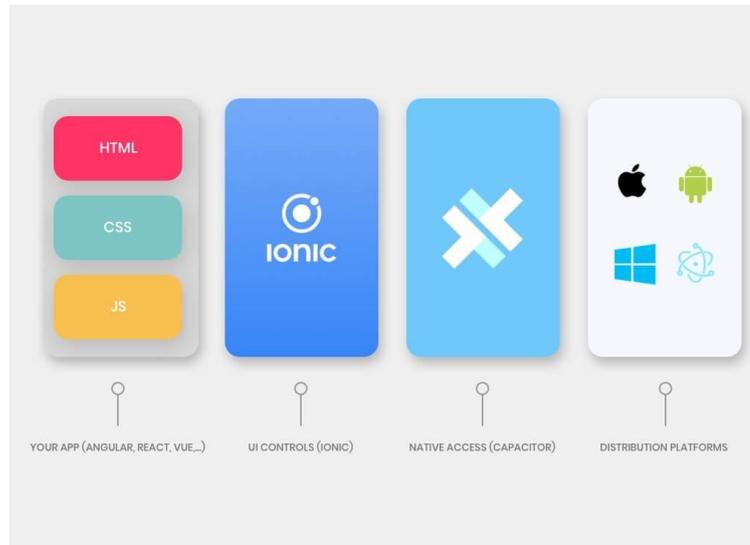


Figura 1 Arquitectura Ionic

Fuente:[8]

1.2.2.4.1.1 Cordova

Cordova es una herramienta para el desarrollo de aplicaciones multiplataforma a partir de tecnologías web estándar (HTML, CSS, JavaScript). Su principal objetivo es el de proporcionar un puente de acceso a la API de dispositivos nativos y agruparlos para su correcta distribución. Cordova da como resultado aplicaciones híbridas, lo que quiere decir que no son ni nativas y ni puramente basadas en web. [9]

1.2.2.4.1.2 Capacitor

Herramienta desarrollada por el equipo de Ionic como una alternativa a Cordova, Capacitor ofrece los mismos beneficios multiplataforma pero centrados en abordar las necesidades de desempeño y seguridad. Puede ejecutar aplicaciones web de forma nativa en iOS, Android, Electron y web, además de proporcionar una interfaz potente para el acceso a los SDK y API nativas de cada plataforma mientras mantiene la mayor parte de la aplicación ejecutándose en un control WebView que desbloquea los beneficios multiplataforma.[10]

1.2.2.4.2 React Native

React Native es un framework de JavaScript utilizado para crear aplicaciones nativas para iOS y Android. Basado en la biblioteca React de JavaScript para crear componentes visuales, cambiando su uso para que no se ejecuten en el navegador, sino directamente en plataformas móviles, en este caso iOS y Android. En otras palabras, lo que se obtiene es una verdadera aplicación nativa, no diferente de una aplicación que se desarrolla usando Objective-C o código Java.[11]

1.2.2.4.3 Flutter

Flutter es un framework desarrollado por Google y lanzado por primera vez como un proyecto de código abierto a fines de 2018 con el objetivo desarrollar aplicaciones para diferentes plataformas. El kit de desarrollo proporciona una gran biblioteca de elementos estándar de las interfaces de usuario de Android e iOS, también se puede utilizar para desarrollar aplicaciones web de escritorio.

La aplicación desarrollada con Flutter tiene la apariencia normal de la aplicación en cada sistema, y se ejecuta como se espera en todos los sistemas, sin requerir que los programadores presten atención a las peculiaridades de cada sistema.[12]

1.2.2.5 API

API es un conjunto de definiciones y protocolos que se utilizan para desarrollar e integrar software de aplicación. API significa Interfaz de programación de aplicaciones. Las API permiten que sus productos y servicios se comuniquen con otros sin saber cómo se implementan. Esto simplifica que el desarrollo de aplicaciones ahorra tiempo y dinero. Las API le brindan flexibilidad; simplifican el diseño, la administración y el uso de aplicaciones, brindan oportunidades de innovación, lo que las hace ideales para diseñar nuevos productos y herramientas.[13]

1.2.2.5.1 SOAP(Protocolo de Acceso a Objetos simples)

SOAP o también conocido como protocolo de acceso a objetos simples es un formato a través del cual se define el intercambio de datos basado en HTTP para la transmisión y XML para la codificación de datos entre dos o mas usuarios, sin tener en consideración el lenguaje o plataforma que se empleen. La usabilidad de SOAP está basada en la utilización de protocolos ligeros y estándares que permitan una óptima conexión.[14]

SOAP define un formato de mensajes sobre una envoltura que se compone de encabezado el cual es opcional y cuerpo en donde se transfiere los datos XML

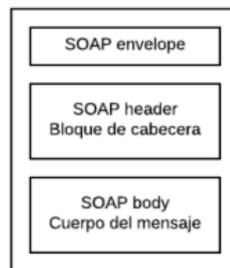


Figura 2 Estructura del mensaje SOAP

Fuente: [14]

1.2.2.5.2 REST

REST es un estilo de arquitectura utilizado para el desarrollo de API web, el estilo arquitectónico consiste en clientes y servidores, los cuales exponen funcionalidades mediante recursos identificados por URI, los clientes generan solicitudes mediante métodos que interactúan con el servidor el cual los procesa y genera una respuesta conveniente, la construcción de solicitudes y respuestas se realiza en torno a la representación de los recursos, siendo uno de los más importantes la representación del recurso actual. A diferencia de SOAP, mantiene un consume de datos menor ya que no utiliza el formato XML para la transmisión de datos, en su lugar utiliza JSON el cual no requiere de cabecera.[14]

REST sigue cuatro principios fundamentales:

- 1) Interfaz uniforme para la identificación de recursos

- 2) Utilización de métodos estandarizados HTTP
- 3) Comunicación sin mantener estados
- 4) Recursos con múltiples representaciones

HTTP expone los métodos GET, PUT, POST y DELETE los cuales se relacionan con las operaciones CRUD (Leer, Actualizar, Crear y Borrar):

GET. - Obtiene información del servidor.

PUT. - Actualiza o cambia un recurso.

POST. - Agrega información.

DELETE. -Elimina información del servidor.

1.2.2.6 Seguridad en las API

Una API está expuesta a vulnerabilidades similares a las de una aplicación web, datos personales como registros de salud, financieros y comerciales requieren de protección, para lo cual establecer las identidades de los usuarios antes que acceda a los recursos se considera una característica clave en el desarrollo de las API. [15]

Para lo cual se hará alusión a dos conceptos importantes en la seguridad de las API:

1.2.2.6.1 Autorización

La autorización es una función que especifica si algún usuario tiene el permiso o los privilegios para realizar cualquier acción sobre algún recurso específico, este garantiza que después de determinar que las credenciales de algún usuario son válidas y aplicado un control de acceso, limite las acciones del usuario dentro de determinado sistema.[16]

1.2.2.6.2 Autenticación

La autenticación es el proceso el cual verifica la identidad de un usuario, la cual no denota que esta persona pueda acceder o no a un recurso en particular.[17]

1.2.2.6.2.1 Autenticación básica

El proceso de autenticación básica puede ser considerando métodos tales como password o passphrase, tarjetas de identidad, dispositivos USB, pero en los últimos años el método de autenticación más utilizado son los relacionados con biometrías que caracterizan a cada uno de los usuarios. Cada uno de los métodos mencionados poseen sus niveles de seguridad protegiendo los datos o información que se dispone.

1.2.2.6.2.2 Autenticación basada en token

La autenticación basada en token permite que los clientes ingresen sus usuarios y contraseñas asignadas previamente las cuales no serán compartidas tanto por los administradores como por los clientes recibiendo una cadena encriptada la cual les permite el ingreso a dicha información esto puede realizarlo por un periodo de tiempo designado con el fin de precautelar la información.[17]

Uno de los métodos más conocidos de autenticación por token en denominado JWT (JSON Web Tokens) el cual es un estándar basado en JSON para la creación de tokens.

JWT generalmente está formado por tres partes principales.

Header o cabecera: contiene información del token, tales como el tipo de token y el algoritmo usado para generación del token.

Payload: Generalmente contiene información del usuario al que pertenece el token.

Firma: Se usa para verificar que el token no haya sido alterado o generado por otro remitente. [18]

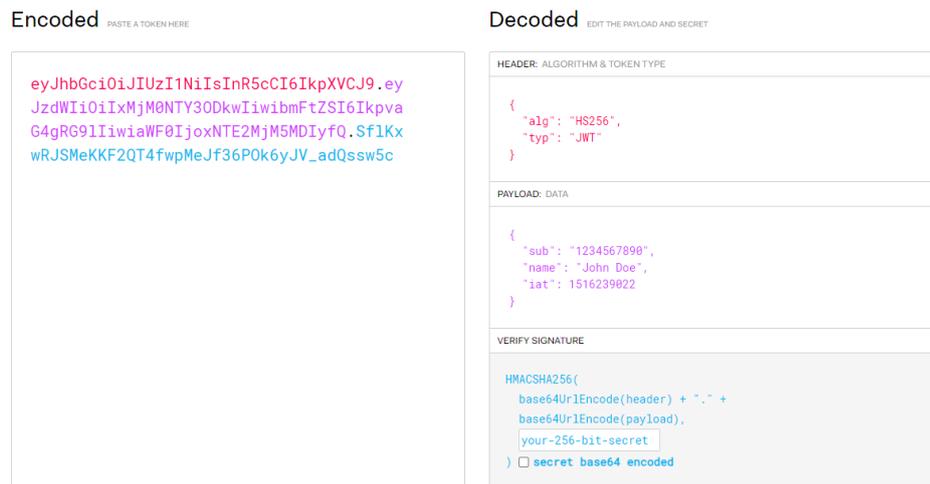


Figura 3 Estructura JWT

Fuente: [18]

1.2.2.6.2.3 Autenticación basada en clave API

Es un método más seguro que los mencionados anteriormente debido a que este tipo de autenticación para un correcto funcionamiento se debe configurar el acceso a los recursos de la API, esto se lo realiza con la generación de una clave o secret key para los clientes que deseen acceso.[19]

1.2.2.6.2.4 OAuth 2.0

Con el avance de la tecnología y los diferentes tipos de autenticaciones existentes la seguridad debe ser más eficiente y veraz al momento de interactuar con la aplicación es por ello que OAuth es un framework utilizado para autenticaciones permitiendo que las aplicaciones obtengan un acceso limitado a las diversas cuentas de los usuarios, el funcionamiento consiste en la delegación del permiso al servicio que controla las cuentas dando las autorizaciones necesarias.[19]

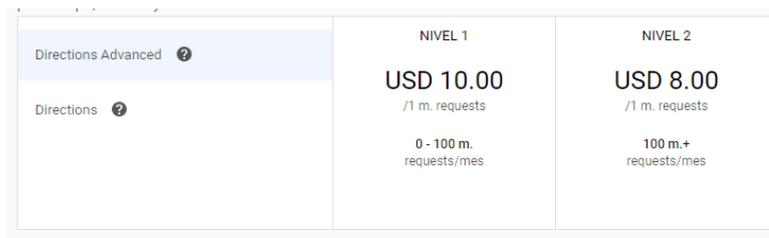
1.2.2.7 Google Maps API

Los conjuntos de interfaces de programación de aplicaciones o API desarrolladas por Google permiten la comunicación con Google Maps con el objetivo de implementar sus servicios de mapas como guías para el correcto funcionamiento de diversas aplicaciones que lo usen.

Dentro del API de Google Maps podemos encontrar diversos sub-servicios, siendo los principales:

1.2.2.7.1 Directions API

Este servicio utiliza una solicitud HTTP para devolver direcciones en formato JSON o XML entre ubicaciones. Puede recibir instrucciones para varios modos de transporte, como tránsito, conducción, caminar o andar en bicicleta.[20]



The image shows a pricing table for the Google Maps API. It has two columns for 'NIVEL 1' and 'NIVEL 2'. The 'Directions' service is selected. NIVEL 1 costs USD 10.00 per 1 million requests, with a limit of 0-100 million requests per month. NIVEL 2 costs USD 8.00 per 1 million requests, with a limit of 100 million+ requests per month.

	NIVEL 1	NIVEL 2
Directions Advanced		
Directions	USD 10.00 /1 m. requests 0 - 100 m. requests/mes	USD 8.00 /1 m. requests 100 m.+ requests/mes

Figura 4 Precio Ditections API

Fuente: [20]

1.2.2.7.2 Roads API

Permite asignar coordenadas GPS a la geometría de las carreteras y determinar el límite de velocidad a lo largo de esos tramos de carretera. La API está disponible a través de una interfaz HTTPS simple y expone diversos servicios tales como Snap to roads, Nearest roads, Speed limits.[20]

Roads - Nearest Road ?	NIVEL 1	NIVEL 2
Roads - Route Traveled ?	USD 10.00 /1 m. requests	USD 8.00 /1 m. requests
	0 - 100 m. requests/mes	100 m.+ requests/mes

Figura 5 Precio Roads API

Fuente:[20]

1.2.2.7.3 Maps JavaScript API

Permite personalizar mapas con su propio contenido e imágenes para mostrarlos en páginas web y dispositivos móviles. La API de JavaScript de Maps presenta cuatro tipos de mapas básicos (mapa de ruta, satélite, híbrido y terreno) que puede modificar utilizando capas, estilos, controles, eventos, varios servicios y bibliotecas.[20]

1.3 Objetivos

Objetivo General:

Desarrollar una aplicación móvil multiplataforma para la gestión y control de entregas de los portadores de la empresa ESFLOSERV CIA. LTDA.

Objetivos Específicos:

- 1) Identificar los procesos y requerimientos necesarios para el desarrollo de la aplicación.
- 2) Determinar la tecnología que mejor se adapte al desarrollo de la aplicación multiplataforma.
- 3) Implantar la aplicación multiplataforma para la gestión y control de entregas.

CAPITULO II. METODOLOGÍA

2.1 Materiales

2.1.1 Humanos

- Investigador
- Docente Tutor

2.1.2 Institucionales

- ESFLOSERV CIA. LTDA
- Biblioteca y repositorios virtuales de la Universidad Técnica de Ambato

2.1.3 Otros

- Computador portátil ASUS
- Materiales de Oficina (Esferos, lápices, hojas de papel)
- Libros, Tesis, Artículos Científicos
- Internet

2.1.4 Económicos (Presupuesto)

Tabla 1. Materiales Economicos (Presupuesto)

NO.	DETALLE	UNIDAD	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
1	Internet	horas	250	\$0.85	\$212,5
2	Resma de papel	c/u	3	\$5.00	\$15.00
3	Impresiones	c/u	600	\$0.20	\$120
4	Copias	c/u	150	\$0.02	\$3.00
5	Laptop	c/u	1	\$800	\$800
				Sub.Total	\$1150,5

				Imprevisto(10%)	\$115,05
				Total	\$1265,55

Elaborada por: El Investigador

La tabla numero 1 hace referencia a los gastos realizados durante el desarrollo de la investigación.

2.2 Métodos

2.2.1 Modalidad de la investigación

Investigación Bibliográfica: Se aplicará una investigación de carácter Bibliográfico ya que la investigación se apoyará en el uso de libros, documentos técnicos, tesis, revistas y artículos existentes sobre temas relacionados con el desarrollo de aplicaciones multiplataforma, geolocalización, servicios de TI (tecnología de la información) y gestión logística.

Investigación Aplicada: La investigación es de carácter aplicada ya que se empleará todo lo estudiado durante la carrera para satisfacer las necesidades de desarrollo de la presente aplicación.

Investigación de Campo: La investigación de campo tendrá como objetivo que el investigador realice levantamiento de información en el lugar de los hechos para obtener información relacionada con los objetivos del trabajo de grado.

2.2.2 Población y Muestra

Tabla 2 Población

Numero	Tipo de población	Cantidad
1	Encargado de Sistemas	1
2	Personal Administrativo	3
Total		4

Elaborada por: El Investigador

Tabla 2. Establece la población en un total de 4 personas, conformadas por el encargado de sistemas y el personal administrativo de la empresa interesada en el desarrollo del proyecto.

Debido a que el tamaño total de la población no excede las 100 personas no es necesario realizar un muestreo y se trabaja con la totalidad de la población

2.2.3 Recolección de Información

Tabla 3 Plan de recolección de información

PREGUNTAS BASICAS	EXPLICACION
¿Para qué?	Para alcanzar los objetivos de la investigación
¿De qué personas u objetos?	De la empresa ESFLOSERV CIA. LTDA
¿Sobre qué aspectos?	Gestión y entrega de paquetes
¿Quién, Quienes?	Investigador: Anderson Naranjo
¿Cuándo?	Octubre 2020 - Febrero 2021
¿Dónde?	Ciudad de Ambato
¿Cuántas veces?	Una

¿Qué técnicas de recolección?	Entrevista
¿Con que?	Cuestionario
¿En qué situación?	En el proceso de entrega de los diferentes paquetes de la empresa

Elaborada por: El Investigador

2.2.4 Procesamiento y Análisis de Datos

Para la recolección y análisis de los datos se realizará una entrevista con el encargado de sistemas de la empresa, ya que es necesario conocer el funcionamiento del actual sistema que administra los paquetes que serán entregados por la empresa y que requerimientos tiene para el desarrollo de la aplicación.

Posteriormente con la ayuda de la metodología seleccionada se realizará la interpretación de la información obtenida con el objetivo de definir los requerimientos para el desarrollo de la aplicación.

2.2.4.1 Resultados de la Entrevista

Los resultados obtenidos en base a la entrevista aplicada al encargado de sistemas de la empresa serán analizados basándose en la metodología designada, al utilizar XP, se realizará una interpretación en la fase I, debido a que en este apartado se realiza la interpretación de todos los datos obtenidos para el desarrollo del software, ya que serán necesarios para establecer las historias de usuario.

CAPITULO III.- RESULTADOS Y DISCUSIÓN.

3.1 Análisis y Discusión de resultados

3.1.1 Metodologías de desarrollo ágil

Las metodologías de desarrollo ágil suelen ser un proceso incremental esto quiere decir una entrega frecuente en ciclos rápidos, cooperación entre los clientes y desarrolladores continúan trabajando a través de una comunicación muy fina y continua, simple ya que el método es fácil de aprender y modificar para el equipo y finalmente adaptativo permitiendo los cambios de última hora. La metodología ágil proporciona una serie de pautas y principios, así como técnicas pragmáticas para hacer menos complicada la entrega de proyectos, satisfacer a los clientes y equipos de trabajo y evitar el camino burocrático de la metodología. Tradicionalmente, se generan pocos documentos y no se utilizan métodos formales. Estos métodos enfatizan la capacidad de afrontar el cambio. Lo importante es un seguimiento planificado y estricto.[21]

3.1.2 Metodología Xtream Programin(XP)

Es un método ágil que se enfoca en mejorar la relación entre las personas, que es la clave para un desarrollo de software exitoso. En este proceso, promueve el trabajo en equipo, presta atención al aprendizaje de los desarrolladores y proporciona un buen ambiente de trabajo. XP pone énfasis en la colaboración cliente-desarrolladores, XP es especialmente adecuado para proyectos que son inexactos, volátiles y requieren altos riesgos técnicos.[22]

3.1.2.1 Roles de la metodología Xtream Programming

Programador: Es el encargado de la creación del código fuente para el desarrollo de la aplicación o sistema

Cliente o usuario: Encargado de brindar la información necesaria para las historias de usuarios con el fin de establecer los recursos y prioridades en el desarrollo del sistema o aplicación

Tester o encarga de pruebas: Como su nombre lo indica es la persona asignada para realizar las evaluaciones al sistema o aplicación

Tracker o encargado del seguimiento: Persona la cual está en relaciona al proyecto dando el seguimiento necesario y verificando los resultados en los tiempos asignados.

Entrenador: Son personas que están en constate relación con la metodología XP con el fin de dar guía a los distintos usuarios y encaminarlos hacia las buenas prácticas de desarrollo de software.

Consultor: Son personas que no están involucradas en el proyecto, pero poseen cierta información de relevancia con respecto a temas los cuales brindan su asesoría.

Big boss o gestor: El trabajo asignado al gestor es de suma importancia ya que es el medio de comunicación entre los usuarios y programadores.

3.1.2.2 Etapas de la metodología Xtreme Programming

Exploración: El cliente describe brevemente los requisitos necesarios para desarrollar una aplicación o un sistema, y estos requisitos deben entenderse y explicarse claramente.

Planeación: En esta etapa intervienen las historias de usuarios las cuales deben ser tomadas en el mismo grado de importancia con el fin de cumplirlas.

Planificación de iteraciones: Es la etapa donde se realizan o se brindan los entregables.

Codificado: Etapa netamente asignada a los programadores encargados de la funcionabilidad del sistema o aplicación, en esta etapa se la puede realizar en parejas o puede asignarse más personal.

Pruebas de aceptación: La etapa final se encarga de analizar la aceptación de la solicitud.

3.1.2.3 Ventajas y desventajas de la metodología XP

La metodología XP es considerada una metodología ágil y posee varias ventajas, pero de igual forma esta presenta desventajas y son las siguientes:

Ventajas:

- Las correcciones de errores se las puede realizar tempranamente evitando inconvenientes en los siguientes pasos.
- Gracias al manejo de las historias de usuarios se cumple con satisfacción los requisitos establecidos por los clientes.
- La eficiencia del manejo de procesos en el desarrollo del proyecto con la colaboración de agentes externos permite una mejor comprensión de temas no conocidos

Desventajas:

- El uso de esta metodología es de un círculo pequeño ya que su utilización emerge en grandes empresas provocando fallos.
- Una de las fallas característica de esta metodología conocida por muchos expertos en el tema es que esta metodología no tiene una previsualización de los futuros errores que se puedan presentar.
- En el caso de implementar esta metodología en proyectos de gran escala produce errores por tal razón es más factible utilizarla en proyectos de corto plazo.

3.1.3 Metodología Scrum

La metodología de trabajo Scrum tuvo sus principios básicos en la década de 1980. Fue desarrollada por Goldratt, Takeuchi y Nonaka para satisfacer las necesidades de la reingeniería de procesos.

El concepto de Scrum se originó a partir del nuevo proceso de desarrollo. Productos de éxito en Japón y Estados Unidos. Los equipos que desarrollan estos productos tienen requisitos muy comunes, así como requisitos novedosos y deben ser puestos en el mercado en mucho menos tiempo que el que se tardó en lanzar el producto anterior. Estos equipos siguen un modelo de ejecución de proyectos muy similar. [23]

3.1.3.1 Roles de la metodología Scrum

Product Owner: Este rol es asignado a la persona para la toma de decisiones, es el encargado de conocer a profundidad el negocio además de tomar las ideas de los usuarios y ordenarlas de tal forma sean concebidas en prioridades.

Scrum Master: Es la persona encargada de llevar el control del cumplimiento de la metodología y la corrección de errores.

Equipo de desarrollo: Son los encargados del desarrollo del proyecto y pueden estar conformadas de 5 a 10 personas dependiendo la complejidad de las tareas.

Usuarios: Son los clientes que harán uso de la aplicación.

Stakeholders: Beneficiarios del funcionamiento del proyecto.

3.1.3.2 Características de la metodología Scrum

Product Backlog: Es el listado de las necesidades expresadas o manifestadas por los clientes o las empresas.

Sprint Backlog: Son los listados que conforman los sprint que están dentro del desarrollo del proyecto.

Incremento: Son cada una de las partes que poseen funcionalidades que serán adjuntas a cada uno de los sprint

3.1.3.3 Ventajas y Desventajas de la metodología Scrum

La utilización de la metodología Scrum posee ciertas ventajas y desventajas descritas a continuación:

Ventajas:

- Al desarrollar el proyecto se puede tener un seguimiento de fácil comprensión debido a las herramientas utilizadas para mejor entendimiento.
- De ser el caso de solicitar cambios estos se los puede realizar de forma sencillas vinculados de manera inmediata al proyecto
- Las entregas de resultados se las realizara en el tiempo estimado sin demoras y con funcionalidades.

Desventajas:

- La organización para el desarrollo del proyecto dependerá de cada uno de los interventores.
- La formación académica será esencial para la implementación de esta metodología
- Genera problemas en su desempeño al ser utilizada en proyectos complejos.

3.1.4 Metodología Seleccionada

En el mundo del desarrollo de software existen un sin número de metodologías ya sean tradicionales o ágiles, pero para el avance tecnológico lo mas adecuado es la utilización de metodologías que permitan un desenvolvimiento acorde siguiendo planes o plazos de entrega por la cual se tomó en consideración la metodología XP o Programación Extrema debido a:

- Es un método de ingeniería de software que se considera el proceso de desarrollo de software ágil más destacado.

- La principal diferencia entre la programación extrema y los métodos tradicionales es que enfatiza la adaptabilidad en lugar de la previsibilidad.
- La forma de trabajo de la metodología XP se centra en grupos pequeños brindando un mejor manejo y de fácil comprensión.
- La obtención de buenos resultados con la implementación de la metodología XP posee una gran aceptación debido a su gran adaptabilidad, eficiencia y factibilidad de uso.

3.1.5 Tecnología de desarrollo multiplataforma elegida

Se decide utilizar Ionic, debido a que permite desarrollar aplicaciones multiplataforma híbridas basadas en tecnologías web como angular, react y vuejs. Las cuales facilitan el desarrollo de las aplicaciones, además de tener una documentación muy amplia la cual se mantiene actualizada día tras día.

Ionic al poder utilizar 3 de tecnologías web diferentes, se opta por Angular ya que mantiene una gran comunidad de desarrolladores aportando para la resolución de dudas y errores, además de mantener una documentación muy amplia.

Para poder ejecutar la aplicación de forma nativa Ionic cuenta con dos contenedores de acceso nativo, siendo estos Capacitor y Cordova, para lo cual se optó por utilizar capacitor ya que esta desarrollado por el mismo equipo de Ionic con el objetivo de mejorar el rendimiento y seguridad de las aplicaciones en comparación con cordova el cual es su competidor.

3.1.6 Tecnología para el desarrollo de la API elegida

La empresa ESFLOSERV CIA. LTDA cuenta con un servidor el cual se encuentra configurado con Apache como servidor web y PHP como su lenguaje de programación, por lo cual se decidió por utilizar PHP para el desarrollo de la API lo cual facilitara su despliegue.

Para mantener segura la API se usará una autenticación por token, más específicamente JWT (JSON Web Token) lo cual permitirá mantener seguro el acceso a la información con tokens generados para cada usuario.

La estructura de la respuesta de la API comprende 3 partes principales:

Estado: El cual identifica si la petición fue exitosa o no.

Mensaje: Da a conocer más detalles del estado de la petición.

Respuesta: Datos solicitados por el cliente, los cuales se enviarán encriptados.

```
▼ {  
  "estado": true,  
  "mensaje": "Proceso exitoso",  
  "res":  
    "eyJjaXB0ZXJ0ZXh0IjoiaUZ4U2RxVHhyNEFW"  
}
```

Figura 6 Estructura respuesta API

Elaborado por: Anderson Naranjo

3.1.7 Uso de Google Maps API

Dentro de los varios servicios que brinda Google se encuentra la API de Google Maps la cual nos permite integrar y manipular mapas dentro de aplicaciones web y móviles, este tipo de mapas ayuda mostrando información geo-referencial de una forma muy entendible e interactiva con los usuarios de las aplicaciones. Lo cual facilitara al usuario encontrar los puntos de entrega de los paquetes.

Uno de los sub servicios de la API de Google Maps es el Directions el cual permite marcar sobre el mapa una sugerencia de ruta la cual facilitara al repartidor trasladarse al lugar asignado para la respectiva entrega. Google utiliza información como el tráfico y la distancia para sugerir la ruta más optima la cual puede o no ser considerada por el repartidor.

3.1.8 Arquitectura de la Aplicación

La arquitectura definida será una cliente servidor ya que se tiene como clientes varios dispositivos móviles que se comunicaran con una API para enviar y recibir información. Por su parte para el desarrollo de la aplicación, debido a que se utilizara IONIC con Angular los cuales mantienen como arquitectura predeterminada el modelo MVC(Modelo, Vista, Controlador), el modelo consta de la definición de la estructura de los datos a utilizarse en la aplicación y que se usaran para la comunicación con la API, la vista viene representada por todos los componentes visuales que dispone Angular junto con IONIC, el controlador que es el encargado de comunicar el modelo y la vista utilizando typescript que es el lenguaje por defecto utilizado por Angular.



Figura 7 Arquitectura de la Aplicación

Elaborado por: Anderson Naranjo

3.2 Desarrollo de la Propuesta

3.2.1 Fase 1: Exploración

XP (Xtreme Programming) define como primera fase la exploración en la cual los clientes definen los requerimientos los cuales serán transformados en historias de usuario, el estado inicial del proyecto.

3.2.1.1 Levantamiento de la información

Una fuente de información que permita conocer las necesidades de los usuarios son entrevistas directas realizando preguntas que buscan respuestas claras y a la vez llenas de información de importancia para el desarrollo de una herramienta acorde a dichas necesidades implementándolas en la aplicación móvil multiplataforma para la gestión y control de entregas de los portadores de la empresa ESFLOSERV CIA. LTDA, la cual fue realizada al Ing. Miguel Miniguano jefe del departamento de sistemas, dicha entrevista se le la cumplió en dos fases debido a la disponibilidad de tiempo la primera fase se utilizó la herramienta zoom, posteriormente en la segunda fase se pudo establecer de manera directa en la empresa cumpliendo con las normas de bioseguridad.

Se considero un tipo de entrevista semi dirigida la cual permite que el entrevistado pueda brindar respuestas concretas, pero dejando espacios los cuales pueden profundizarlos dando información de importancia, además que este tipo de entrevistas permite al entrevistador llevar guías dando la factibilidad de tocar temas principales utilizando terminología sencilla y fácil de comprender dando seguridad que el entrevistado pueda conocer de mejor forma los temas a tratar esto. La entrevista se la realizo en base a un cuestionario que se lo adjuntara en el ANEXO A

3.2.1.2 Análisis de Resultados

La información adquirida mediante la entrevista realizada al Ing. Miguel Miniguano jefe del área de sistemas de la empresa ESFLOSERV CIA. LTDA. En donde se da a conocer la siguiente información.

¿Se hace uso de aplicaciones de reparto regularmente?

Se manifestó que en muchas de las ocasiones los repartidores deben hacer uso de aplicaciones externas para llegar a los destinos como por ejemplo WhatsApp y Waze dando como resultados inconvenientes en el manejo de tiempos y provocando disgustos a los usuarios.

¿Su empresa posee las herramientas necesarias para implementar una aplicación móvil multiplataforma para la gestión y control de entregas de los porteadores?

Se menciona que la empresa posee los recursos necesarios para la implementación de aplicaciones o sistemas que permitan servir de guías para las entregas ya que se encuentran en constantes actualizaciones tecnológicas con el fin de brindar el mejor servicio a sus clientes.

¿Desearía que se destine un financiamiento para la implementación aplicación móvil multiplataforma para la gestión y control de entregas de los porteadores?

Consecuente a eso menciono que las autoridades superiores de la empresa están interesadas en destinar financiamiento para el desarrollo de software en diversas áreas.

¿A su criterio el uso de una aplicación móvil multiplataforma para la gestión y control de entregas de los porteadores podría desarrollar nuevos recursos y convertir a la empresa ESFLOSERV CIA LTDA en pionera en el servicio de entregas?

Manifestó el Ing. Miguel Miniguano que debido a la situación actual que vive el país el servicio de entregas se está catapultando ya que muchos de los usuarios desean adquirir diversos artículos sin correr riesgos con su salud es por ello que el uso constante de diversas aplicaciones que permiten acceder a estos beneficios es de gran auge y la empresa al incorporar una aplicación oficial estarían dando un paso gigantesco para evolucionar, además estarían brindando el apoyo y oportunidades para desarrolladores de software.

Concluyendo con la entrevista Ing. Miguel Miniguano profundizo en el tema de la tecnología afirmando que las personas dedican mayor tiempo al uso de teléfonos celulares ya sea que posean una conexión mediante wifi o datos móviles los usuarios convirtieron el teléfono celular en un accesorio de la vida cotidiana por estas razones y con el fin de brindar un apoyo la empresa propuso la creación de una aplicación

móvil multiplataforma para la gestión y control de entregas de los portadores que les permitirá mejorar los procesos y ahorrar recursos.

3.2.2 Fase 2: Planificación del Proyecto

XP define como principal actividad de desarrollo la interacción con los clientes o usuarios finales. Por lo tanto, se genera varias historias de usuario que tienen el objetivo de cumplir las tareas en los periodos de tiempo establecidos.

3.2.2.1 Historias de usuario

En una metodología de desarrollo ágil, las historias de usuario comprenden la unidad de trabajo más pequeña, se las considera como un objetivo y no una función, la cual es expresada desde la perspectiva del usuario del software.

Tabla 4 Estructura Historia de Usuario

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:		Usuario:	
Nombre de la Historia:			
Prioridad en el negocio:		Riesgo en el desarrollo:	
Puntos estimados:		Iteración asignada:	
Programador responsable:			
Descripción:			
Observación:			

Elementos de la plantilla de historia de usuario:

- **Numero:** Identificador de la historia de usuario.
- **Usuario:** Persona o grupo de personal a las que se asigna la historia de usuario.
- **Nombre de la historia:** Título o nombre asignado a la historia de usuario.
- **Prioridad en el negocio:** Asignación de valor (Alto, Medio y Bajo) dependiendo de la necesidad del usuario.

- **Riesgo en el desarrollo:** Asignación de valor (Alto, Medio y Bajo) dependiendo
- **Programador responsable:** Persona a la cual se le asigna la responsabilidad de llevar a cabo dicha historia de usuario.
- **Puntos estimados:** Cantidad de días estimados para el desarrollo de la historia de usuario.
- **Iteración asignada:** Identificación de la iteración a la cual se le asigna la historia de usuario.
- **Descripción:** Información adicional añadida por el usuario para mejorar en entendimiento de la historia de usuario.
- **Observación:** Campo opcional en el cual se detalla los procesos que tengan relación con la historia de usuario.

Tabla 5 Historia de Usuario 1

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	001	Usuario:	Desarrollador
Nombre de la Historia:	Determinar estructura del proyecto		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	4	Iteración asignada:	1
Programador responsable:	Anderson Naranjo		
Descripción: Estructurar el proyecto tanto en la parte de desarrollo multiplataforma y el desarrollo de la API			
Observación: Determinar las tecnologías necesarias para el desarrollo.			

Tabla 6 Historia de Usuario 2

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	002	Usuario:	Desarrollador
Nombre de la Historia:	Análisis de la base de datos		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto

Puntos estimados:	3	Iteración asignada:	1
Programador responsable:	Anderson Naranjo		
Descripción: Un claro entendimiento de la base de datos es determinante para un correcto manejo de la información			
Observación: La base de datos a utilizar ya se encuentra creada y funciona con el sistema principal de la empresa.			

Tabla 7 Historia de Usuario 3

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	003	Usuario:	Porteador
Nombre de la Historia:	Ingreso a la aplicación		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	4	Iteración asignada:	1
Programador responsable:	Anderson Naranjo		
Descripción: Un módulo de ingreso mediante un usuario y contraseña es fundamental para la protección de la información de los usuarios			
Observación: Para el ingreso a la aplicación se requiere de un usuario y contraseña, los cuales ya se encuentran establecidos en la base de datos de la empresa.			

Tabla 8 Historia de Usuario 4

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	004	Usuario:	Porteador
Nombre de la Historia:	Pantalla de inicio		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	5	Iteración asignada:	2
Programador responsable:	Anderson Naranjo		
Descripción: Pantalla principal de la aplicación en la cual se mostrará el listado de los paquetes a ser entregados por el usuario.			
Observación: La pantalla principal permitida cerrar la sesión además de indicadores para ayudar a la sincronización de información pendiente.			

Tabla 9 Historia de Usuario 5

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	005	Usuario:	Porteador
Nombre de la Historia:	Buscador de paquetes		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	4	Iteración asignada:	2
Programador responsable:	Anderson Naranjo		
<p>Descripción: Modulo el cual permita filtrar los paquetes mediante código, cliente, zona, sección, etc. Además de permitir escanear el código de barras de los paquetes para una búsqueda más rápida.</p> <p>Observación: Utilizar la cámara del dispositivo para escanear el código de barras del paquete y cotejarlo con los códigos de los paquetes del usuario para mostrar la información más detallada.</p>			

Tabla 10 Historia de Usuario 6

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	006	Usuario:	Porteador
Nombre de la Historia:	Pantalla de detalle del paquete		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	3	Iteración asignada:	2
Programador responsable:	Anderson Naranjo		
<p>Descripción: Pantalla en la cual se mostrará la información completa del paquete la cual incluye información del cliente, cantidad y detalle de objetos fuera de caja e información de la ubicación</p> <p>Observación: Los campos se mostrarán agrupados según la información del cliente, objetos fuera de caja y ubicación para permitir al usuario cotejar con la información física.</p>			

Tabla 11 Historia de Usuario 7

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	007	Usuario:	Porteador
Nombre de la Historia:	Cambio de estado de paquete		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	3	Iteración asignada:	3
Programador responsable:	Anderson Naranjo		

<p>Descripción: Modulo el cual permita seleccionar cualquiera de los estados posibles de los paquetes, los estados incluyen entregado, nadie en casa, siniestrado, etc. Además de añadir una observación y mínimo una o máximo 3 fotografías.</p>
<p>Observación: Guardar las fotografías en el dispositivo para usos futuros.</p>

Tabla 12 Historia de Usuario 8

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	008	Usuario:	Porteador
Nombre de la Historia:	Funcionamiento Offline		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	6	Iteración asignada:	3
Programador responsable:	Anderson Naranjo		
<p>Descripción: Permitir que la aplicación almacene en el dispositivo la información de los paquetes y demás información necesaria la cual facilite trabajar a la aplicación si necesidad de conexión a internet.</p>			
<p>Observación: Estructurar una base de datos local en los dispositivos para almacenar la información necesaria de forma ordenada y coherente.</p>			

Tabla 13 Historia de Usuario 9

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	009	Usuario:	Porteador
Nombre de la Historia:	Sincronización de entregas		
Prioridad en el negocio:	Alta	Riesgo en el desarrollo:	Alto
Puntos estimados:	8	Iteración asignada:	3
Programador responsable:	Anderson Naranjo		
<p>Descripción: Modulo que permita sincronizar la información almacenada localmente, mostrando la cantidad de registros a sincronizar y mostrar el progreso de la sincronización.</p>			
<p>Observación:</p>			

Tabla 14 Historia de Usuario 10

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	010	Usuario:	Porteador
Nombre de la Historia:	Marcadores de paquetes en Mapa		
Prioridad en el negocio:	Medio	Riesgo en el desarrollo:	Alto
Puntos estimados:	4	Iteración asignada:	4
Programador responsable:	Anderson Naranjo		
Descripción:	<p>Mostrar un mapa donde se muestren los marcadores de la ubicación de cada paquete. Los marcadores deberán mostrar información básica de los clientes.</p>		
Observación:	<p>Usar la API de Google Maps mapa mostrar el mapa y añadir los distintos marcadores</p>		

Tabla 15 Historia de Usuario 11

Elaborado por: Anderson Naranjo

Historia de Usuario			
Número:	011	Usuario:	Porteador
Nombre de la Historia:	Marcaje de ruta		
Prioridad en el negocio:	Medio	Riesgo en el desarrollo:	Alto
Puntos estimados:	4	Iteración asignada:	4
Programador responsable:	Anderson Naranjo		
Descripción:	<p>Mostrar una ruta desde la ubicación actual del usuario hacia la ubicación de entrega del paquete designado.</p>		
Observación:	<p>Usar la API de Google Maps para marcar la ruta sobre el mapa.</p>		

3.2.2.2 Estimación de Historias de Usuario

La metodología XP propone “Planning Poker” como técnica de estimación. Para los cual se consideró trabajar 5 horas por cada día laborable, para los cual se presenta el plan de entregas correspondiente a la aplicación.

Tabla 16 Estimación de Historias de Usuario

Elaborado por: Anderson Naranjo

Iteración	Número	Historia de Usuario	Tiempo Estimado
-----------	--------	---------------------	-----------------

			Días	Horas
1	001	Determinar estructura del proyecto	4	20
1	002	Análisis de la base de datos	3	15
1	003	Ingreso a la aplicación	4	20
2	004	Pantalla de inicio	5	25
2	005	Buscador de paquetes	4	20
2	006	Pantalla de detalle del paquete	3	15
3	007	Cambio de estado de paquete	3	15
3	008	Funcionamiento Offline	6	30
3	009	Sincronización de entregas	8	40
4	010	Marcadores de paquetes en Mapa	4	20
4	011	Marcaje de ruta	4	20
Tiempo Total de estimación (Días/Horas)			45	225

3.2.2.3 Tareas

Las tareas en la metodología XP representan un conjunto de actividades, las mismas que tienen un inicio y un fin, las tareas se asignan a un responsable que es el encargado del cumplimiento de la tarea en el tiempo establecido.

Tabla 17 Tarea-Determinar la estructura del proyecto

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T1	Código Historia: 001
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de la Tarea: Determinar la estructura del proyecto	
Fecha de Inicio: 15/03/2021	Fecha fin: 18/03/2021
Descripción: Se analizará la estructura del proyecto tanto para el desarrollo móvil multiplataforma y la API.	

Tabla 18 Tarea-Análisis de la base de datos

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T2	Código Historia: 002
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de la Tarea: Análisis de la base de datos	
Fecha de Inicio: 19/03/2021	Fecha fin: 23/03/2021
Descripción: Analizar y comprender las diferentes tablas de la base de datos de la empresa, además de comprender la información que maneja cada una de ellas.	

Tabla 19 Tarea-Análisis de la base de datos

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T3	Código Historia: 003

Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Implementación del módulo de autenticación de la API	
Fecha de Inicio: 24/03/2021	Fecha fin: 25/03/2021
Descripción: Desarrollar el módulo el cual permita autenticar al usuario en la API mediante JWT	

Tabla 20 Tarea-Diseño y codificación de la pantalla de ingreso a la aplicación

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T4	Código Historia: 003
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Diseño y codificación de la pantalla de ingreso a la aplicación	
Fecha de Inicio: 26/03/2021	Fecha fin: 29/03/2021
Descripción: Desarrollar el módulo que le permita ingresar su usuario y contraseña para poder ingresar a la aplicación	

Tabla 21 Tarea-Diseño de la pantalla de inicio de la aplicación.

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T5	Código Historia: 004
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Nombre de la Tarea: Diseño de la pantalla de inicio de la aplicación.	
Fecha de Inicio: 30/03/2021	Fecha fin: 30/03/2021
Descripción: Diseñar la distribución de los componentes que comprenderán la pantalla de inicio de la aplicación.	

Tabla 22 Tarea-Desarrollo módulo de paquetes de la API

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T6	Código Historia: 004
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Desarrollo módulo de paquetes de la API	
Fecha de Inicio: 31/03/2021	Fecha fin: 01/04/2021
Descripción: Desarrollar un módulo que exponga la información de los paquetes el cual será consumido por la aplicación móvil.	

Tabla 23 Tarea-Mostrado lista de paquetes.

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T7	Código Historia: 004
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Mostrado lista de paquetes.	
Fecha de Inicio: 02/04/2021	Fecha fin: 05/04/2021

Descripción: Consumir el módulo de paquetes de la API para poder mostrar la información en la pantalla de inicio.

Tabla 24 Tarea-Buscador por información del paquete o cliente.

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T8	Código Historia: 005
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Nombre de la Tarea: Buscador por información del paquete o cliente.	
Fecha de Inicio: 06/04/2021	Fecha fin: 06/04/2021
Descripción: Desarrollar un buscador el cual permita filtrar los paquetes mediante la información del cliente o del paquete.	

Tabla 25 Tarea-Buscador por código de barras.

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T9	Código Historia: 005
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de la Tarea: Buscador por código de barras.	
Fecha de Inicio: 07/04/2021	Fecha fin: 09/04/2021
Descripción: Desarrollar un buscador el cual permita escanear un código de barras mediante la cámara del dispositivo para realizar una búsqueda más directa.	

Tabla 26 Tarea-Diseño de pantalla para detalle de paquetes

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T10	Código Historia: 006
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Nombre de la Tarea: Diseño de pantalla para detalle de paquetes	
Fecha de Inicio: 12/04/2021	Fecha fin: 12/04/2021
Descripción: Diseñar una pantalla la cual permita mostrar de forma detallada un paquete determinado.	

Tabla 27 Tarea-Implementación de pantalla para detalle de paquetes

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T11	Código Historia: 006
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Implementación de pantalla para detalle de paquetes	
Fecha de Inicio: 13/04/2021	Fecha fin: 14/04/2021
Descripción: Implementar la funcionalidad que permita mostrar de forma detallada un paquete determinado.	

Tabla 28 Tarea-Diseño de pantalla para cambio de estado de los paquetes

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T12	Código Historia: 007
Tipo de Tarea: Desarrollo	Puntos Estimados: 1
Nombre de la Tarea: Diseño de pantalla para cambio de estado de los paquetes.	
Fecha de Inicio: 15/04/2021	Fecha fin: 15/04/2021
Descripción: Diseñar una pantalla la cual permita seleccionar uno de los diversos estados disponibles para un paquete, además de añadir una observación, fotografías y la ubicación.	

Tabla 29 Tarea-Implementación de pantalla para cambio de estado de los paquetes.

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T13	Código Historia: 007
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Implementación de pantalla para cambio de estado de los paquetes.	
Fecha de Inicio: 16/04/2021	Fecha fin: 19/04/2021
Descripción: Implementar una pantalla la cual permita seleccionar uno de los diversos estados disponibles para un paquete, además de añadir una observación, fotografías y la ubicación.	

Tabla 30 Tarea-Diseño de estructura de base de datos local.

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T14	Código Historia: 008
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Diseño de estructura de base de datos local.	
Fecha de Inicio: 20/04/2021	Fecha fin: 21/04/2021
Descripción: Diseñar la estructura de tablas para almacenar la información de forma local de una manera ordenada y entendible.	

Tabla 31 Tarea-Desarrollo de un servicio para el manejo de la base de datos local

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T15	Código Historia: 008
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de la Tarea: Desarrollo de un servicio para el manejo de la base de datos local	
Fecha de Inicio: 22/04/2021	Fecha fin: 27/04/2021
Descripción: Desarrollar un servicio con métodos el cual permita extraer e insertar información de forma fácil y rápida en la base de datos local.	

Tabla 32 Tarea-Desarrollo del módulo para el envío de información almacenada en el dispositivo.

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T16	Código Historia: 009
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de la Tarea: Desarrollo del módulo para el envío de información almacenada en el dispositivo.	
Fecha de Inicio: 28/04/2021	Fecha fin: 03/05/2021
Descripción: Desarrollar un módulo que permita enviar los registros que requieren ser actualizados en el servidor.	

Tabla 33 Tarea-Desarrollo del módulo para recepción de información en la API

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T17	Código Historia: 009
Tipo de Tarea: Desarrollo	Puntos Estimados: 4
Nombre de la Tarea: Desarrollo del módulo para recepción de información en la API	
Fecha de Inicio: 04/05/2021	Fecha fin: 07/05/2021
Descripción: Desarrollar un módulo en la API en el cual permita la recepción de la información a sincronizar, la cual será actualizada en la base de datos y las imágenes serán almacenadas en determinada ubicación del servidor.	

Tabla 34 Tarea-Integración con el API de Google Maps

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T18	Código Historia: 010
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Integración con el API de Google Maps	
Fecha de Inicio: 10/05/2021	Fecha fin: 11/05/2021
Descripción: Añadir a la aplicación los plugins necesarios para trabajar con el API de Google Maps.	

Tabla 35 Tarea-Mostrar Marcadores de paquetes

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T19	Código Historia: 010
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Mostrar Marcadores de paquetes	
Fecha de Inicio: 12/05/2021	Fecha fin: 13/05/2021
Descripción: Dibujar sobre el mapa los marcadores de los paquetes, los cuales señalan la ubicación de la entrega. Dichos marcadores presentan información básica del cliente.	

Tabla 36 Tarea-Trasado de ruta sobre el mapa

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T20	Código Historia: 011
Tipo de Tarea: Desarrollo	Puntos Estimados: 2
Nombre de la Tarea: Trasado de ruta sobre el mapa	
Fecha de Inicio: 14/05/2021	Fecha fin: 17/05/2021
Descripción: Trazar una ruta sobre el mapa desde la ubicación actual del porteador hacia el paquete seleccionado.	

Tabla 37 Tarea-Pruebas de aceptación

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T21	Código Historia: 011
Tipo de Tarea: Desarrollo	Puntos Estimados: 8
Nombre de la Tarea: Pruebas de aceptación	
Fecha de Inicio: 18/05/2021	Fecha fin: 27/05/2021
Descripción: El cliente realiza pruebas de la aplicación con el objetivo de exponer dudas e identificar errores.	

Tabla 38 Tarea-Carga de archivos al servidor principal de la empresa

Elaborado por: Anderson Naranjo

Tarea	Responsable: Anderson Naranjo
N° Tarea: T22	Código Historia: 011
Tipo de Tarea: Desarrollo	Puntos Estimados: 3
Nombre de la Tarea: Carga de archivos al servidor principal de la empresa	
Fecha de Inicio: 28/05/2021	Fecha fin: 01/06/2021
Descripción: Se despliega la API en el servidor principal de la empresa.	

3.2.3 Fase 3: Iteraciones

3.2.3.1 Plan de Entregas

Según los lineamientos establecidos por XP, se elabora un plan de entregas tomando en cuenta un mes de 4 semanas (5 días hábiles) y 5 horas por día.

Tabla 39 Plan de Entregas

Elaborado por: Anderson Naranjo

N°	Historia de Usuario	Tiempo Estimado		Iteración Asignada					Entrega Asignada				
		Días	Horas	1	2	3	4	5	1	2	3	4	5
001	Determinar estructura del proyecto	4	20	X					X				
002	Análisis de la base de datos	3	15	X					X				
003	Ingreso a la aplicación	4	20	X					X				
004	Pantalla de inicio	5	25		X					X			
005	Buscador de paquetes	4	20		X					X			
006	Pantalla de detalle del paquete	3	15		X					X			
007	Cambio de estado de paquete	3	15			X					X		
008	Funcionamiento Offline	6	30			X					X		
009	Sincronización de entregas	8	40			X					X		
010	Marcadores de paquetes en Mapa	4	20				X					X	
011	Marcaje de ruta	4	20				X					X	

3.2.3.2 Iteraciones

Tabla 40 Planificación de Iteraciones

Elaborado por: Anderson Naranjo

	N°	Historia de Usuario	Prioridad	Riesgo	Estado de Desarrollo	Prueba
Primera Iteración	1	Determinar estructura del proyecto	Alta	Alto	Completo	Aceptada
	2	Análisis de la base de datos	Alta	Alto	Completo	Aceptada
	3	Ingreso a la aplicación	Alta	Alto	Completo	Aceptada
Segunda Iteración	4	Pantalla de inicio	Alta	Alto	Completo	Aceptada
	5	Buscador de paquetes	Alta	Alto	Completo	Aceptada
	6	Pantalla de detalle del paquete	Alta	Alto	Completo	Aceptada
Tercera Iteración	7	Cambio de estado de paquete	Alta	Alto	Completo	Aceptada
	8	Funcionamiento Offline	Alta	Alto	Completo	Aceptada

	9	Sincronización de entregas	Alta	Alto	Completo	Aceptada
Cuarta Iteración	10	Marcadores de paquetes en Mapa	Media	Alto	Completo	Aceptada
	11	Marcaje de ruta	Media	Alto	Completo	Aceptada

3.2.4 Fase 4: Producción

3.2.4.1 Iteración 1

Tabla 41 Resumen Iteración 1

Elaborado por: Anderson Naranjo

Iteración	Número	Historia de Usuario	Tiempo Estimado	
			Días	Horas
1	1	Determinar estructura del proyecto	4	20
1	2	Análisis de la base de datos	3	15
1	3	Ingreso a la aplicación	4	20
Total			11	55

Determinar estructura del proyecto

Como primer paso se definen las herramientas a utilizar durante el desarrollo del proyecto, además de realizar las configuraciones necesarias para el ambiente de desarrollo.

Hardware

1 laptop Asus Tuf A15, 16GB de Ram, AMD Ryzen 5 4600H, Windows 10 Pro

1 Smartphone Samsung A20s, 5BG de Ram, Android 11

Software

Para el desarrollo de la API se utilizó el paquete XAMPP ya que dispone de las herramientas necesarias para trabajar con PHP, se lo puede descargar directamente desde su página oficial, para el presente proyecto se utilizó la versión 7.4.11.

Al utilizar IONIC para el desarrollo de la aplicación móvil se requiere del sistema de gestión de paquetes de Nodejs denominado NPM (Node Package Manager) el cual se instala en conjunto con nodejs que puede ser descargado gratuitamente desde su página oficial como un archivo de instalación, en el presente proyecto se utilizó la versión 14.15.0.

Para instalar IONIC utilizaremos NPM en conjunto con el comando de instalación que lo podemos encontrar en la página oficial de ionic.

```
C:\Users\ander>npm install -g @ionic/cli|
```

Figura 8 Instalación de IONIC

Elaborado por: Anderson Naranjo

Generar el proyecto

```
C:\Users\ander>ionic start AppEntregas
```

Figura 9 Generación del proyecto

Elaborado por: Anderson Naranjo

En el transcurso de la creación del proyecto el asistente nos realizara una serie de preguntas para determinar el framework a utilizar, en este caso seleccionaremos Angular, seguidamente nos preguntara la plantilla a utilizar en la aplicación la cual será blank que nos creara una aplicación en blanco y finalmente nos preguntara si deseamos integrar Capacitor a nuestro proyecto al cual le diremos que sí.

Capacitor requieren detallar las plataformas para las cuales se ejecutará el proyecto, esto con el objetivo de crear las carpetas que contendrán el código nativo de cada plataforma.

```
.\AppEntregasV02>ionic capacitor add android|
```

```
\AppEntregasV02>ionic capacitor add ios
```

Figura 10 Añadido de plataformas

Elaborado por: Anderson Naranjo

Estructura del proyecto

La estructura del proyecto está definida por ionic, la cual separa los archivos en diferentes carpetas dependiendo de su funcionalidad, el objetivo principal de esta estructura es facilitarle al desarrollador encontrar los archivos dependiendo de su funcionalidad, además de facilitar futuras modificaciones.

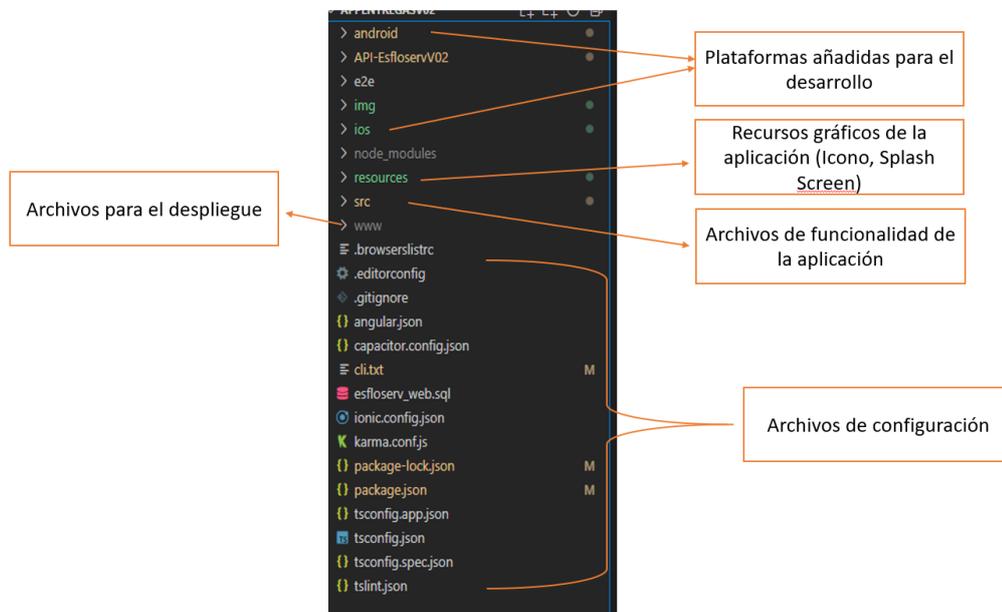


Figura 11 Estructura del proyecto

Elaborado por: Anderson Naranjo

Análisis de la base de datos

Actualmente la empresa trabaja con un sistema web el cual les ayuda a gestionar varios de los procesos de la empresa, entre ellos la administración de las entregas, la aplicación móvil trabajara compartiendo la misma base de datos, pero únicamente las tablas necesarias para su correcto funcionamiento.

Comprender cuales son las tablas relacionadas con las entregas y que datos almacena cada una de ellas será de vital importancia para evitar inconsistencias en los datos y problemas futuros con el sistema principal.

Las principales tablas por usarse son:

Usuarios. - Para los usuarios se maneja dos tablas, una tabla almacena información personal de los usuarios, mientras que la otra almacena las credenciales para la autenticación.

Paquetes. - Esta tabla contiene la información detallada de todos los paquetes a entregarse o ya entregados, de la misma se deberá filtrar los paquetes dependiendo del porteador y de si están listos para entregarse.

Entregas. – Almacena la información de las entregas, la observación, ubicación (Latitud, Longitud) y la referencia a las fotografías.

Configuración. - En la tabla de configuración se encontrarán los registros de los diferentes estados de los paquetes y demás información para usarse en la aplicación.

Por cuestiones de seguridad se evitó los nombres reales de las tablas y de los diferentes campos que las componen.

Ingreso a la aplicación

Para un correcto ingreso a la aplicación el usuario deberá ingresar sus credenciales (Usuario, Contraseña) las cuales serán validadas en el API que en caso de ser correctas devolverá un token que será almacenado en la aplicación para futuras peticiones, como medidas de seguridad después de 5 intentos fallidos se bloquea la interfaz por un lapso de 20 segundos los cuales serán mostrados en la interfaz.

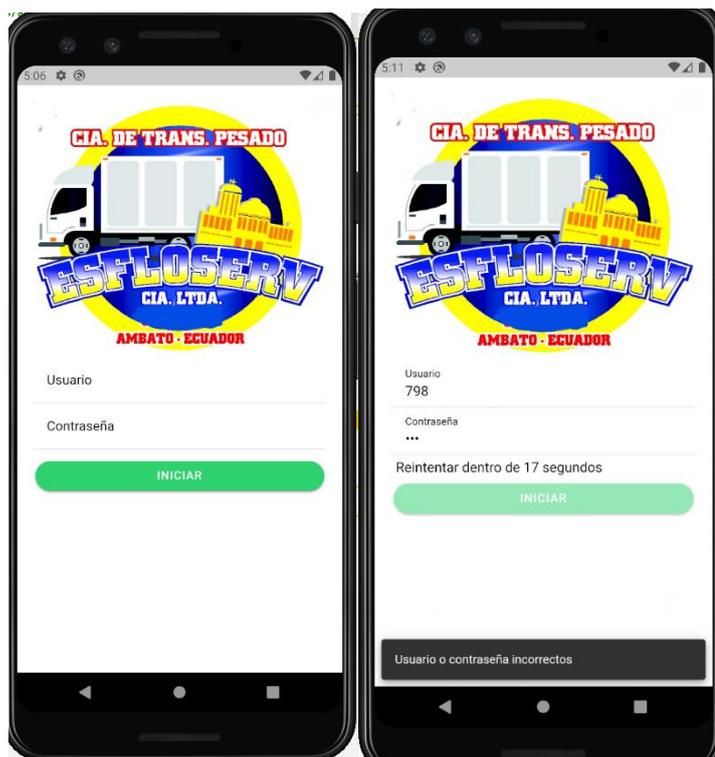


Figura 12 Pantalla de Ingreso a la aplicación

Elaborado por: Anderson Naranjo

3.2.4.2 Iteración 2

Tabla 42 Resumen Iteración 2

Elaborado por: Anderson Naranjo

Iteración	Número	Historia de Usuario	Tiempo Estimado	
			Días	Horas
2	4	Pantalla de inicio	5	25
2	5	Buscador de paquetes	4	20
2	6	Pantalla de detalle del paquete	3	15
Total			12	60

Pantalla de inicio

La pantalla de inicio muestra el listado de los paquetes asignados al porteador para su entrega, además contiene los controles de búsqueda y en la parte superior derecha el acceso al menú de sincronización en el cual se puede observar de coloración roja la cantidad de registros que requieren ser sincronizados.



Figura 13 Pantalla de inicio

Elaborado por: Anderson Naranjo

Buscador de paquetes

El buscador de paquetes comprende dos partes, la primera permite el ingreso de texto por el cual será filtrado el listado de paquetes, esta búsqueda se puede realizar por el nombre de la persona perteneciente el paquete, su número de cedula o la dirección a entregar el paquete.

La segunda parte consta de un acceso a la cámara que permite escanear el código de barra lo cual facilitara la búsqueda de un determinado paquete.



Figura 14 Buscador de paquetes

Elaborado por: Anderson Naranjo



Figura 15 Escaneo código de barras

Elaborado por: Anderson Naranjo

Pantalla de detalle del paquete

Una vez seleccionado el paquete a ser entregado o al cual se le realizara un cambio de estado, se muestra la información de este paquete de una forma más detallada, en esta pantalla se detalla la información del cliente, del paquete y de los productos fuera de caja con el objetivo que le porteador disponga de información más detallada a la hora de realizar alguna entrega.

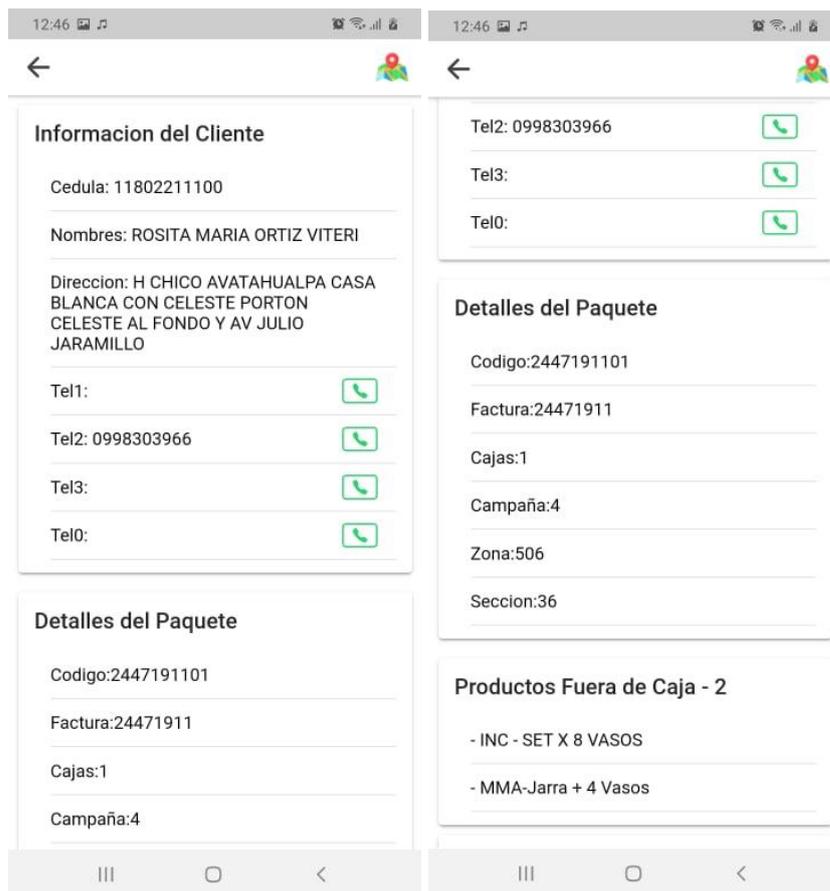


Figura 16 Pantalla de detalle del paquete

Elaborado por: Anderson Naranjo

3.2.4.3 Iteración 3

Tabla 43 Resumen Iteración 3

Elaborado por: Anderson Naranjo

Iteración	Número	Historia de Usuario	Tiempo Estimado	
			Días	Horas
3	7	Cambio de estado de paquete	3	15
3	8	Funcionamiento Offline	6	30
3	9	Sincronización de entregas	8	40
Total			17	85

Cambio de estado de paquete

Para realizar el cambio de estado de algún paquete se requiere 4 ítems de información:

Nuevo estado: El cual puede ser entregado, siniestrado, zona equivocada, etc.

Fotografía: La fotografía se puede obtener directamente desde la cámara o seleccionando de la galería del dispositivo, se puede seleccionar mínimo 1 y máximo 3.

Observación: Campo opcional en el cual se puede detallar cualquier tipo de inconveniente ocurrido durante el transcurso del cambio de estado.

Ubicación: Se obtiene directamente del dispositivo, la cual solo se muestra en la parte inferior.

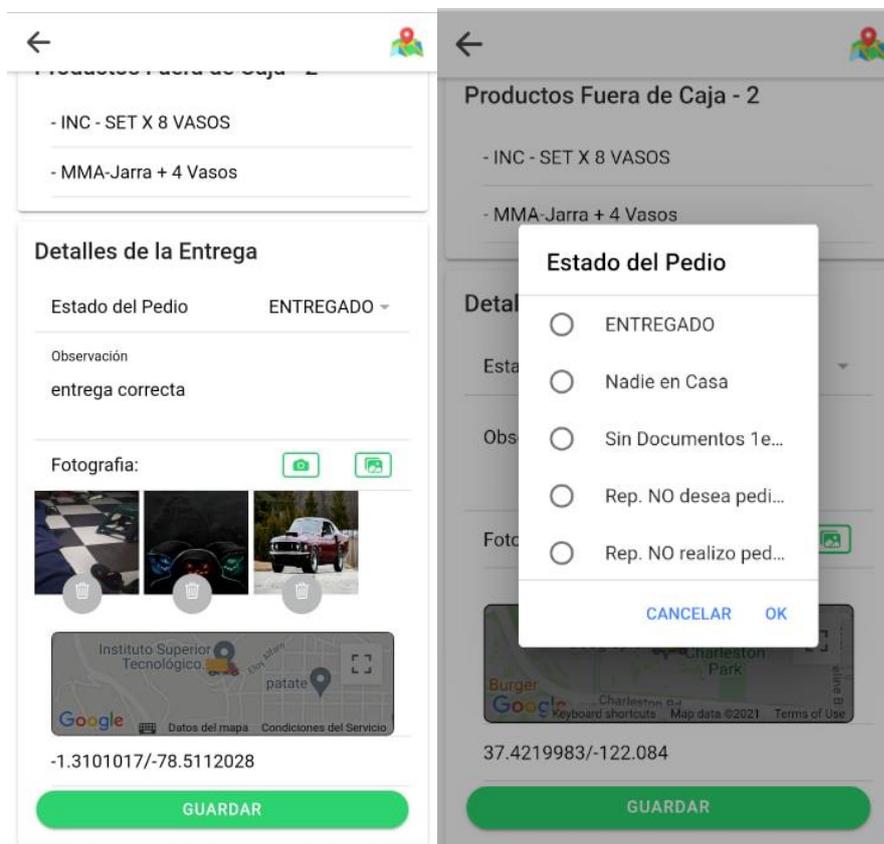


Figura 17 Cambio de estado de paquete

Elaborado por: Anderson Naranjo

Funcionamiento Offline

El funcionamiento offline es de vital importancia para el correcto desempeño de la aplicación ya que esto permitirá que los portadores mantengan un acceso a la información de los paquetes sin la necesidad de una constante conexión a internet.

Para poder almacenar la información en el dispositivo se utiliza SQL Lite, al cual se podrá acceder mediante un servicio que dispone de los métodos necesarios para manejar toda la información almacenada en el dispositivo, de esta manera se mantiene toda la funcionalidad centralizada en un solo lugar.

```
20  })
21  export class BaseLocalService {
22  private database: SQLiteObject;
23  private dbReady = new BehaviorSubject<boolean>(false);
24
25  constructor(public toastController: ToastController, private plt: Platform, private sqlitePorter: SQLi
26
27  this.plt.ready().then(() => {
28
29  this.sqlite.create({ name: "LocalBF407.db", location: "default" }).then((db: SQLiteObject) => {
30  this.database = db;
31
32  this.crearBaseDeDatos();
33  }).catch((error) => {
34  console.log(error);
35  })
36  });
37  }
38  > getDatabaseState() { ...
40  }
41  > crearBaseDeDatos() { ...
51  }
52  > añadirUsuarioLog(usr: Usuario) { ...
68  }
69  > actualizarUSR(id: string) { ...
73  }
74  > actualizarUSRToken(id: string, token: string) { ...
78  }
79  > actualizarUSRDIS(id: string) { ...
83  }
84  > eliminarUSR(id: string) { ...
88  }
89  > getUsuarioLog(): Promise<Usuario> { ...
132  }
133  > getUsuario(nom): Promise<Usuario> { ...
175  }
176  > getUsuarioLog(): Promise<number> { ...
184  }
185  > getURL(): Promise<string> { ...
197  }
198  > getTiempo(): Promise<string> { ...
209  }
210  > actualizarURL(url: string) { ...
214  }
215  > actualizarTiempo(url: string) { ...
219  }
220  > añadirEstado(est: Estado) { ...
227  }
```

Figura 18 Funcionamiento Offline

Elaborado por: Anderson Naranjo

Sincronización de entregas

Los paquetes que hayan sufrido algún cambio de estado requieren ser enviados al servidor para ser almacenado en la base de datos de la empresa, en la parte superior derecha de la aplicación se muestra resaltado de rojo la cantidad de paquetes que requieren ser sincronizados, de igual manera se podrá acceder a un menú mas detallado en el cual no solo se encuentra la opción de sincronizar, sino que también se muestra la información del usuario que a iniciado sesión en la aplicación y por último la opción para el cierre de sesión.

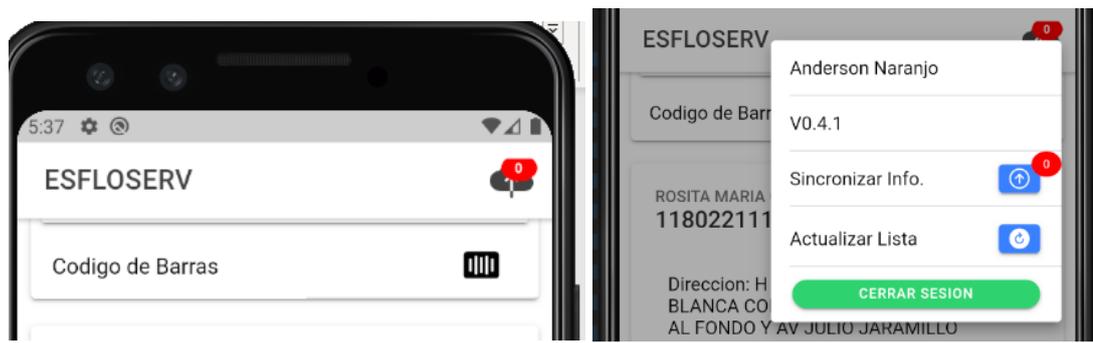


Figura 19 Menú Sincronización de Entregas

Elaborado por: Anderson Naranjo

3.2.4.4 Iteración 4

Tabla 44 Resumen Iteración 4

Elaborado por: Anderson Naranjo

Iteración	Número	Historia de Usuario	Tiempo Estimado	
			Días	Horas
4	10	Marcadores de paquetes en Mapa	4	20
4	11	Marcaje de ruta	4	20
Total			8	40

Marcadores de paquetes en Mapa

Gracias a la API de Google Maps se pudo integrar un mapa el cual contiene marcadores los cuales señalan los lugares a entregar los paquetes, dichos marcadores pueden desplegar información básica del lugar de entrega.

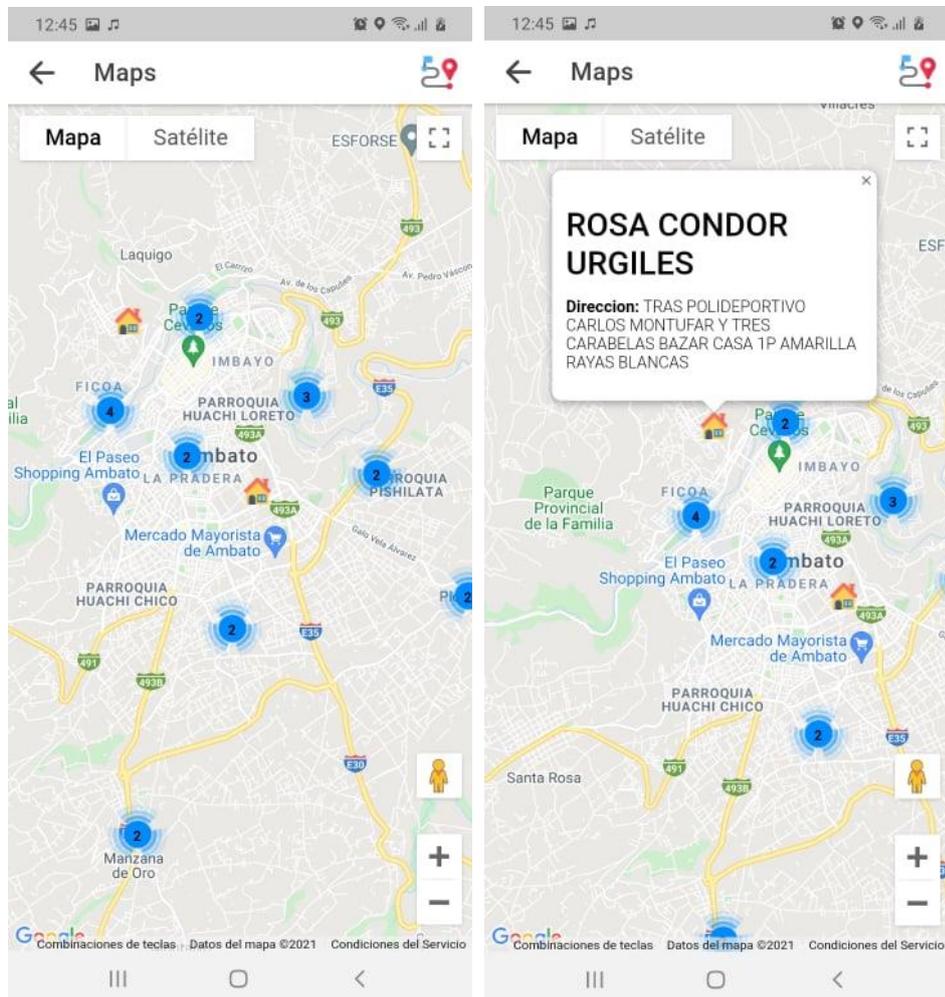


Figura 20 Marcadores de paquetes en Mapa

Elaborado por: Anderson Naranjo

Marcaje de Ruta

Con la finalidad de ayudar a los porteadores a trasladarse a los lugares de entregas se puede marcar sobre el mapa una ruta sugerida, para esto se utiliza el sub servicio Directions API, el cual traza una sugerencia de ruta entre un punto A y un punto B.

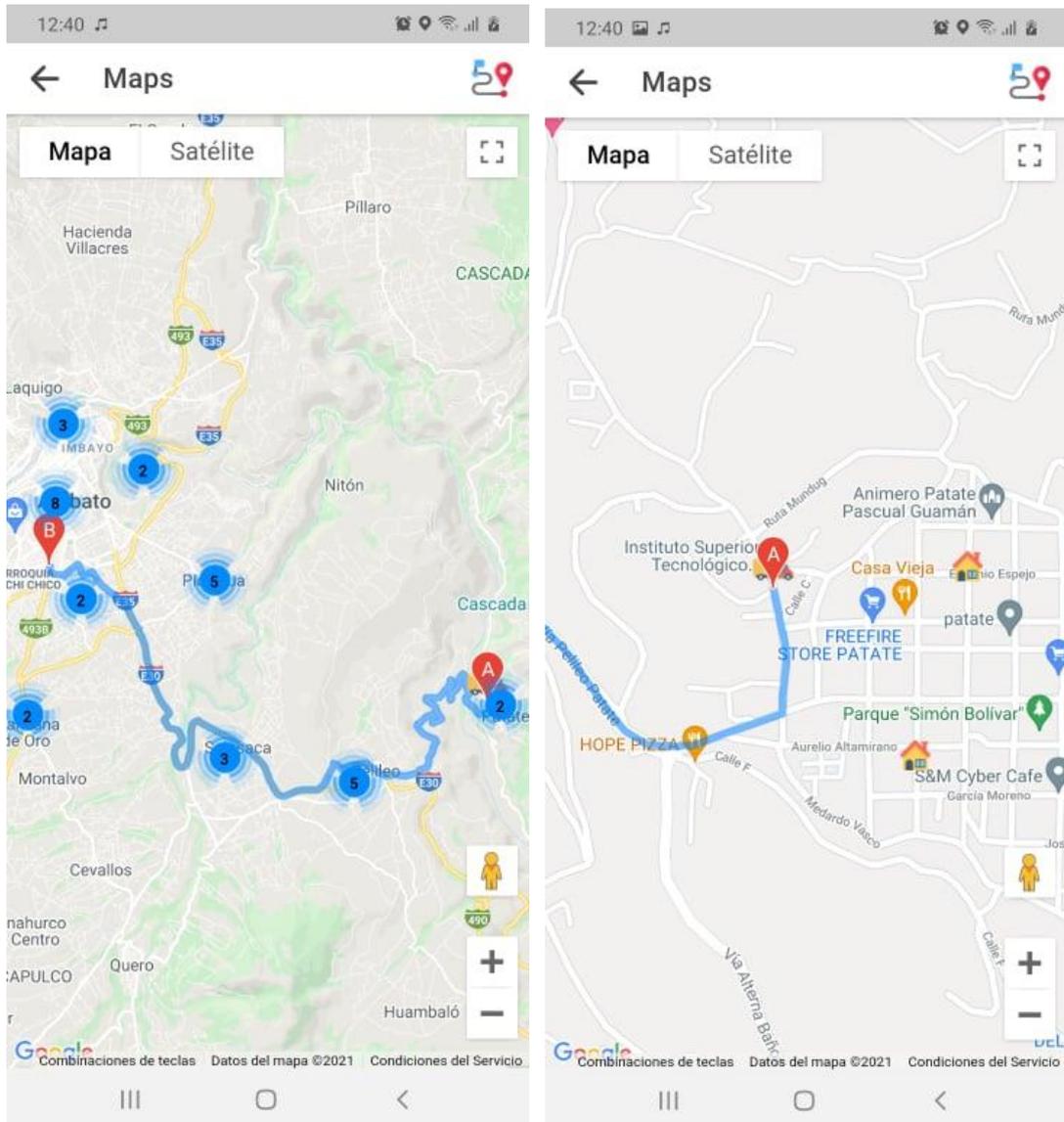


Figura 21 Marcaje de Ruta

Elaborado por: Anderson Naranjo

3.2.5 Codificación

3.2.5.1 API

Autenticación

El método de autenticación de la API requiere de dos parámetros (Usuario, Contraseña) los cuales llegarán encriptados, una vez desencriptados, la contraseña será concatenada con una frase predeterminada para posteriormente obtener un hash el cual

será cotejado con la base de datos, ya validada la información del usuario se procede a la generación del token el cual será utilizado en futuras peticiones.

```
if ($_SERVER['REQUEST_METHOD'] == 'POST') {  
  
    try {  
  
        $input = (array) json_decode(file_get_contents('php://input'), TRUE);  
        $decri= decrypt($input['value']);  
        $obj= json_decode($decri, false);  
        $archivo = __DIR__ . "/sgp.ini";  
        $contenido = parse_ini_file($archivo, false);  
        $hash= $contenido['HASH_SGP'];  
        $clave = md5( $hash. $obj->password );  
        $sql = $dbConn->prepare(  
        "SELECT du.usuario_id, ...  
        $sql->bindValue(':nombre', $obj->nombre);  
        $sql->bindValue(':passwor',$clave );  
        $sql->execute();  
        $sql->setFetchMode(PDO::FETCH_ASSOC);  
        $arr = $sql->fetchAll(PDO::FETCH_ASSOC);  
        $user='';  
        $jwt='';  
        $mensaje='OK';  
        $estado=false;  
        if(count($arr)>0){  
            $user=$arr[0];  
            if($user['usuario_bloqueado']!='1'){  
                $mensaje='Usuario bloqueado';  
            }else if($user['usuario_estado']!='desconectado'){  
                $mensaje='Usuario ya a iniciado una sesion';  
            }else{  
                $estado=true;  
                $time = time();  
                $dat= date("Y-m-d H:i:s", $time);  
                $jwt=jwtCreate($user['usuario_identificacion'],$dat);  
                $arr[0]['token']=$jwt;  
                actualizarDB($dbConn,$obj->intentos,$user['usuario_identificacion'],$dat);  
            }  
        }else{  
            $mensaje='Usuario o contraseña incorrectos';  
        }  
        $res= array( 'estado' => $estado, 'mensaje'=>$mensaje, 'user'=>encrypt($arr,false));  
        header("HTTP/1.1 200 OK");  
        echo json_encode($res);  
    } catch (Exception $e) {  
        echo 'Excepción capturada: ', $e->getMessage(), "\n";  
    }  
}
```

Figura 22 Autenticación API

Elaborado por: Anderson Naranjo

Creación del Token

La función para la creación del token requiere de dos parámetros los cuales serán incluidos en el cuerpo del token, la primera es la información del usuario al que

pertenece el token y la segunda será la fecha de creación del token, con esta información se genera el token y se regresa el mismo codificado en base64.

```
function jwtCreate($user,$dat){
    $arr = array('alg' => 'HS256', 'typ' => 'JWT');
    $arr2 = json_encode($arr);
    $encoded_header = base64_encode($arr2);
    $secret_key = getClaveSecreeta();

    $arr3 = array( 'usuario' => $user,'date'=>$dat,'timeZone'=>date_default_timezone_get());
    $arr33 = json_encode($arr3);

    $header_payload = $encoded_header . '.' . $encoded_payload;

    $signature = base64_encode(hash_hmac('sha256', $header_payload, $secret_key, true));

    $jwt_token = $header_payload . '.' . $signature;

    return $jwt_token;
}
```

Figura 23 Creación del token

Elaborado por: Anderson Naranjo

Validación de token

Para dar un token como valido tendrá que atravesar 3 filtros, el primero verifica la firma del token, la segunda compara la fecha de creación del token con la base de datos y la tercera verifica el estado del usuario (activo, bloqueado).

```

include("../conexion.php");
include("../jwt.php");
$dbConn = connect($db);
$hed=apache_request_headers();
$auth=false;
$token='';
$res= array( 'estado' => true, 'mensaje' => '', 'res'=>'');
$usuario_identificacion='';
try {
    if(array_key_exists('token', $hed)){
        $token=$hed['token'];
        $auth=jwtVali($token);
    }
} catch (Exception $e) {
    $auth=false;
}
if(!$auth){
    $res['estado']=false;
    $res['mensaje']='No autorizado';
    echo json_encode($res);
}
else{
    $decodeInfo=json_decode(getInfo($token), true);
    $usuario_identificacion=$decodeInfo['usuario'];
    $ingreso=$decodeInfo['date'];
    if(verificarUsuario($dbConn,$usuario_identificacion,$ingreso)){
        $res['mensaje']=$decodeInfo;
        $res['estado']=$auth;
    }
    else{
        $res['mensaje']='Bloqueado';
        $res['estado']=false;
        $auth=false;
        echo json_encode($res);
    }
}
}
function verificarUsuario($Conn,$idUsr,$ingreso){
    $sql = $Conn->prepare("SELECT ...");
    $sql->bindValue(':nombre', $idUsr);
    $sql->bindValue(':ingreso', $ingreso);
    $sql->execute();
    $sql->setFetchMode(PDO::FETCH_ASSOC);
    $arr = $sql->fetchAll(PDO::FETCH_ASSOC);
    if(count($arr)>0){
        return true;
    }
    else{
        return false;
    }
}
}

```

Figura 24 Validación del token

Elaborado por: Anderson Naranjo

Encriptado de la Información

Para el encriptado de la información se utilizó la librería OpenSSL de PHP la cual permite utilizar varios algoritmos de encriptado, en este caso se utilizó el algoritmo AES-256.

```
function encrypt($string,$isString)
{
    $key=getkey();
    $ivLength = openssl_cipher_iv_length('AES-256-CBC');
    $iv = openssl_random_pseudo_bytes($ivLength);

    $salt = openssl_random_pseudo_bytes(256);
    $iterations = 999;
    $hashKey = hash_pbkdf2('sha512', $key, $salt, $iterations, 64);

    if(!$isString){
        $string= json_encode($string);
    }
    $encryptedString = openssl_encrypt($string, 'AES-256-CBC', hex2bin($hashKey), OPENSAL_RAW_DATA, $iv);

    $encryptedString = base64_encode($encryptedString);
    unset($hashKey);

    $output = ['ciphertext' => $encryptedString, 'iv' => bin2hex($iv), 'salt' => bin2hex($salt), 'iterations' => $iterations];
    unset($encryptedString, $iterations, $iv, $ivLength, $salt);

    return base64_encode(json_encode($output));
}
```

Figura 25 Encriptado de la información

Elaborado por: Anderson Naranjo

Desencriptado de la información

```
function decrypt($encryptedString)
{
    $key=getkey();
    $json = json_decode(base64_decode($encryptedString), true);

    try {
        $salt = hex2bin($json["salt"]);
        $iv = hex2bin($json["iv"]);
    } catch (Exception $e) {
        return null;
    }

    $cipherText = base64_decode($json['ciphertext']);

    $iterations = intval(abs($json['iterations']));
    if ($iterations <= 0) {
        $iterations = 999;
    }
    $hashKey = hash_pbkdf2('sha512', $key, $salt, $iterations, 64);
    unset($iterations, $json, $salt);

    $decrypted= openssl_decrypt($cipherText , 'AES-256-CBC', hex2bin($hashKey), OPENSAL_RAW_DATA, $iv);
    unset($cipherText, $hashKey, $iv);

    return $decrypted;
}
```

Figura 26 Desencriptado de la información

Elaborado por: Anderson Naranjo

Listado de paquetes

El método permite exponer el listado de los paquetes a ser entregados por un determinado usuario.

```
function formatearCadena($cadena){
    $cadena=trim($cadena);
    $formato=substr($cadena,1,strlen($cadena)-2);
    return $formato;
}
function listaPaquetes($idPorteador,$db){...
}
function getFueraCaja($idPorteador,$db){...
}
//
if ($_SERVER['REQUEST_METHOD'] == 'GET' && $auth ) {
    try {
        $paquetes=listaPaquetes($usuario_identificacion,$dbConn);
        $FC=getFueraCaja($usuario_identificacion,$dbConn);
        header("HTTP/1.1 200 OK");
        for($i=0;$i<count($paquetes);$i++){
            for($j=0;$j<count($FC);$j++){
                if ($paquetes[$i]['codpaquete'] == $FC[$j]['codpaquete']) {
                    $paquetes[$i]['FueraCaja'] = strval($FC[$j]['FueraCaja']);
                    break;
                }
                $paquetes[$i]['FueraCaja'] = '0';
            }
        }
        $res['res']=encrypt($paquetes,false) ;
        echo json_encode($res);
    } catch (Exception $e) {
        $res['estado']=false;
        $res['mensaje']='Excepción capturada: '. $e->getMessage(). "\n";
        echo json_encode($res);
    }
}
```

Figura 27 Listado de paquetes

Elaborado por: Anderson Naranjo

Cambio de estado de paquetes

El método permite almacenar la información del cambio el estado de un determinado paquete.

```
if ($_SERVER['REQUEST_METHOD'] == 'POST' && $auth) {
    try {
        $input = (array) json_decode(file_get_contents('php://input'), TRUE);
        $decri= decrypt($input['value']);
        $obj= json_decode($decri, false);
        $sql = "INSERT INTO `paquete_entrega` (...
    )
    VALUES (...
    )";
        $statement = $dbConn->prepare($sql);
        $nomI='';
        if(strlen($obj->imagencte01)>0){
            $nomI=GuardarImg($obj->imagencte01,$obj->codigo,$obj->codigoPaquete,$obj->campania,'1');
        }
        $nomI2='';
        if(strlen($obj->imagencte02)>0){
            $nomI2=GuardarImg($obj->imagencte02,$obj->codigo,$obj->codigoPaquete,$obj->campania,'2');
        }
        $nomI3='';
        if(strlen($obj->imagencte03)>0){
            $nomI3=GuardarImg($obj->imagencte03,$obj->codigo,$obj->codigoPaquete,$obj->campania,'3');
        }
        $statement->bindValue(':paqentrega_usuarioid', $obj->idUsr);...
        if(strlen($obj->codigoPaquete)>0){
            $statement->execute();
            guardarMovPaquetes($obj->codigoPaquete,$obj->status10,$dbConn);
        }
        header("HTTP/1.1 200 OK");
        $res['res']=$input;
        echo json_encode($res);
        //echo json_encode($input);
    } catch (Exception $e) {

        $res['estado']=false;
        $res['mensaje']='Excepción capturada: '. $e->getMessage(). "\n";
        echo json_encode($res);
    }
}
```

Figura 28 Listado de paquetes

Elaborado por: Anderson Naranjo

Guardado de imágenes

Las imágenes viajan codificadas en base64, una vez decodificadas son almacenadas en un directorio específico, cada imagen es nombrada utilizando una concatenación del código del cliente, código del paquete, campaña, año, mes, día, hora, minutos y número de la fotografía correspondiente.

```

function GuardarImg($img,$codClien,$codPaq,$campania,$num)
{
    $base_to_php = explode(',', $img);
    $data = base64_decode($base_to_php[1]);
    //codigoCliente_codPaquete_campania-anio -mes-dia-hora-minuto.tipo date("d") . " del " . date("m")
    $nomImg=$codClien."_".$codPaq."_".$campania."_".date("Y")."-".date("m")."-".date("d")."-".date("G")."-".date("i")."-".date("s").".png";
    $filepath = "..\..\..\imagenes/".$nomImg;
    file_put_contents($filepath, $data);
    return $nomImg;
}
> function guardarMovPaquetes($cod,$status,$dbConn){ ...
}

```

Figura 29 Guardado de imágenes

Elaborado por: Anderson Naranjo

3.2.5.2 Aplicación IONIC

Inicio de sesión

```

async loginConectado() {
    var Usuario = (<HTMLInputElement>document.getElementById("txtUsuario2")).value;
    var ContraseñaMd5 = Md5.hashStr((<HTMLInputElement>document.getElementById("txtContraseña2")).value);
    var Contraseña = (<HTMLInputElement>document.getElementById("txtContraseña2")).value;
    console.log(Usuario, Contraseña, 'usrcon', (<HTMLInputElement>document.getElementById("txtUsuario2")));
    let usr = { nombre: Usuario, password: Contraseña, intentos: this.intentos };
    var valEncrip = await this.criptoSer.encriptar(JSON.stringify(usr));

    console.log(usr, valEncrip, this.criptoSer.key);

    this.serUsuario.getUsuario(valEncrip).subscribe(
        async res => {
            try {
                if (res.estado) {
                    var usuarios = await this.criptoSer.desencriptar(res.user, false);
                    usuarios[0].usuario_intentos = this.intentos;
                    this.GuardarTiempo();
                    this.GuardarUsuario(usuarios[0]);
                } else {
                    this.presentToast(res.mensaje);
                }
            } catch (error) {
                this.presentToast('Error al ingresar con este usuario...!!!');
                console.log(error);
            }
        },
        err => {
            console.log(err);
            this.presentToast('Error en la coneccion');
        }
    );
};

```

Figura 30 Login

Elaborado por: Anderson Naranjo

Verificar si un usuario a iniciado sesión

Uno de los primeros métodos a ejecutarse cuando la aplicación se inicia es la comprobación de si un usuario a iniciado sesión, de ser así se procede a cargar la información de dicho usuario y caso contrario se muestra la pantalla de inicio de sesión.

```
    this.database.getDatabaseState().subscribe(rdy => {
      if (rdy) {
        (property) HomePage.database: BaseLocalService
        this.database.getUsuarioLog().then(async (data) => {
          if (data == 0) {
            this.MostrarLogin();
          } else {
            await this.cargarUsuarioBD();
            await this.cargarEstadosDB();
            this.ContadorNoSincronizado();
          }
        }, (error) => {
          this.msjConsola(error);
          this.database.GuardarError('c007::' + error);
          this.MostrarLogin();
        });
      }
    });
```

Figura 31 Verificar si existe usuario logeado

Elaborado por: Anderson Naranjo

Encriptado

Para realizar el encriptado de la información en la aplicación se utiliza la librería crypto-js la cual se puede instalar mediante npm con **npm install crypto-js**, la librería permite utilizar varios algoritmos de encriptado entre ellos AES-256.

```

encrypt(text: string, key: string) {
    var iv = CryptoJS.lib.WordArray.random(16); // the reason to be 16, please read on `encryptMethod` property.
    var salt = CryptoJS.lib.WordArray.random(256);
    var iterations = 999;
    var encryptMethodLength = 64; // example: AES number is 256 / 4 = 64
    var hashKey = CryptoJS.PBKDF2(key, salt, { 'hasher': CryptoJS.algo.SHA512, 'keySize': (encryptMethodLength / 8), 'iterations': iterations });

    var encrypted = CryptoJS.AES.encrypt(text, hashKey, { 'mode': CryptoJS.mode.CBC, 'iv': iv });
    var encryptedString = CryptoJS.enc.Base64.stringify(encrypted.ciphertext);

    var output = {
        'ciphertext': encryptedString,
        'iv': CryptoJS.enc.Hex.stringify(iv),
        'salt': CryptoJS.enc.Hex.stringify(salt),
        'iterations': iterations
    };

    return CryptoJS.enc.Base64.stringify(CryptoJS.enc.Utf8.parse(JSON.stringify(output)));
}

```

Figura 32 Encriptado IONIC

Elaborado por: Anderson Naranjo

Desencriptado

```

decrypt(encryptedString, key: string) {
    var json = JSON.parse(CryptoJS.enc.Utf8.stringify(CryptoJS.enc.Base64.parse(encryptedString)));

    var salt = CryptoJS.enc.Hex.parse(json.salt);
    var iv = CryptoJS.enc.Hex.parse(json.iv);

    var encrypted = json.ciphertext; // no need to base64 decode.

    var iterations = parseInt(json.iterations);
    if (iterations <= 0) {
        iterations = 999;
    }
    var encryptMethodLength = (64); // example: AES number is 256 / 4 = 64
    var hashKey = CryptoJS.PBKDF2(key, salt, { 'hasher': CryptoJS.algo.SHA512, 'keySize': (encryptMethodLength / 8), 'iterations': iterations });

    var decrypted = CryptoJS.AES.decrypt(encrypted, hashKey, { 'mode': CryptoJS.mode.CBC, 'iv': iv });

    return decrypted.toString(CryptoJS.enc.Utf8);
} // decrypt

```

Figura 33 Desencriptado IONIC

Elaborado por: Anderson Naranjo

Carga de paquetes en la base de datos local

Para realizar la carga de los paquetes en la base local, primero obtendremos el listado de paquetes desde la API, para poder realizar la petición a la API se enviará el token en la cabecera, una vez obtenido los paquetes cada uno de los registros serán almacenados en la base de datos local de la aplicación.

```

traerInformacionGuardarDB() {
  this.PantallaSincronizado(true, 'Obteniendo Información', 100, 5);

  this.paquetesService.getListPaquetes(this.UsuarioLog.token).subscribe(res => {
    console.log(res);
    if (res.estado) {
      let ListaPaquetes: any = this.criptoSer.desencriptar(res.res, false);
      this.PantallaSincronizado(true, 'Obteniendo Información', ListaPaquetes.length, 0);
      this.database.eliminarPaquetes();

      for (let index = 0; index < ListaPaquetes.length; index++) {
        this.database.añaddirePaquete(ListaPaquetes[index]);
        this.PantallaSincronizado(true, 'Obteniendo Información', ListaPaquetes.length, index + 1);
      }
      this.PantallaSincronizado(false, 'Obteniendo Información', 100, 0);
      this.cargarInformacionDB();
    } else {
      this.procesarError(res);
      this.PantallaSincronizado(false, 'Obteniendo Información', 100, 0);
    }
  });
}, err => {
  this.PantallaSincronizado(false, 'Obteniendo Información', 100, 0);
  console.log(err);
});
}

```

Figura 34 Carga de paquetes en la base de datos local

Elaborado por: Anderson Naranjo

```

getListPaquetes(tok:string) {
  return this.http.get<any>(this.url + 'Paquetes/Paquetes.php', { headers: { 'token': tok } })
}

```

Figura 35 Petición de paquetes a la API

Elaborado por: Anderson Naranjo

Carga de información desde la base local

Para realizar la carga de información desde la base de datos local, al listado de paquetes se le añade información adicional como el nombre del estado y son filtrados para mostrar solo los paquetes listos para entregar o con estados 02 o 05.

```

cargarInformacionDB() {
  this.PantallaSincronizado(true, 'Mostrando Información', 100, 50);
  this.database.getPaquetesPQ().then((data) => {

    this.listaPaquetes = data;
    this.listaPaquetesAux = data;
    for (var i = 0; i < this.listaPaquetes.length; i++) {
      for (var j = 0; j < this.listaEstados.length; j++) {
        if (this.listaPaquetes[i].status10 == this.listaEstados[j].codigo) {
          this.listaPaquetes[i].estado = this.listaEstados[j].NombreEstado;
        }
      }
    }
    //this.listaPaquetes = this.listaPaquetesAux;
    const result = this.listaPaquetes.filter(paquete => paquete.status10.search('02') == 0
    || paquete.status10.search('05') == 0);
    this.listaPaquetes = result;
    this.listaPaquetesAux = result;
    this.PantallaSincronizado(false, 'Mostrando Información', 100, 50);
    //this.ContadorNoSincronizado();

  }), (error) => {
    console.log(error);
    this.database.GuardarError('md004:' + error);
    this.PantallaSincronizado(false, 'Mostrando Información', 100, 50);
  });
}

```

Figura 36 Mostrado de información desde la base local

Elaborado por: Anderson Naranjo

Sincronización de información

Este método permite enviar todos los registros modificados a la API para que sean almacenados en la base de datos de la empresa.

```

try {
  let enviados = 0;
  let errores = 0;
  let total = 0;
  this.database.getActualisaciones2().then(async (data) => {
    this.msjConsole("Cargado Base Actualizaciones");
    this.PantallaSincronizado(true, 'Sincronizando...', data.length, 0);
    for (var i = 0; i < data.length; i++) {
      let act: Actualizacion = { ...
    }
    let img = true;
    let cImage = true;
    await this.database.getPagenes().then((data) => {
      cImage = this.transBool(data);
    }, (errordb) => {

    });
    if (cImage) {
      await this.database.getActualisacionesImg(data[i].id).then((dataimg) => {

        try {
          if (dataimg.length > 0) {
            act.imagencte01 = dataimg[0].imagencte01;
            act.imagencte02 = dataimg[0].imagencte02;
            act.imagencte03 = dataimg[0].imagencte03;
          } else {
            img = false;
          }

        } catch (errorimg) {
          this.database.GuardarError('c1100:' + errorimg);
        }

      }, (error1) => {
        console.log(error1);
      });
    }

    if (img) {
      var actEncri = this.criptoSer.encrypted(JSON.stringify(act));
      this.paquetesService.PostPaquetesAct(actEncri, this.UsuarioLog.token).subscribe(
        res => {
          if (res.estado) {
            this.database.delActualizacionesById(act.id);
            enviados += 1;
            this.PantallaSincronizado(true, 'Sincronizando...', data.length, enviados);
            total = enviados + errores;
            if (total == data.length) {
              //this.presentToast('Se han sincronizado ' + enviados + ' de ' + data.length);
              Swal2("Listo!!", 'Se han sincronizado ' + enviados + ' de ' + data.length + ' paquetes', "success");
              //this.cargarInformacion();
              this.ContadorNoSincronizado();
              this.PantallaSincronizado(false, 'Sincronizando...', data.length, 0);
            }
          } else {
            i = data.length;
            this.PantallaSincronizado(false, 'Sincronizando...', data.length, 0);
            this.procesarError(res);
          }
        },
        err => { ...
      }
    });
  } else {
    this.presentToast("Error,un paquete no se a podido sincronizar,revise su conexión a internet");

    errores += 1;
    total = enviados + errores;
    if (total == data.length) {
      //this.presentToast('Se han sincronizado ' + enviados + ' de ' + data.length);
      Swal2("Listo!!", 'Se han sincronizado ' + enviados + ' de ' + data.length + ' paquetes', "success");
      //this.cargarInformacion();
      this.cargarInformacionDB();
      this.PantallaSincronizado(false, 'Sincronizando...', data.length, 0);
    }
  }
}

if (data.length == 0) {
  this.PantallaSincronizado(false, 'Sincronizando...', data.length, 0);
  // this.cargarInformacion();
  this.ContadorNoSincronizado();
}

```

Figura 37 Sincronización de información

Elaborado por: Anderson Naranjo

Buscador de Paquetes

```
busquedaPaquete(ev: any) {
  console.log(ev);
  this.listaPaquetes = this.listaPaquetesAux;

  let value = (<HTMLInputElement>event.target).value;
  console.log('value', value);
  if (value != "") {
    const result = this.listaPaquetes.filter((paquete: Paquete) => paquete.codigo.search(value) >= 0
      || paquete.nomcte01.toUpperCase().search(value.toUpperCase()) >= 0
      || paquete.direccion.toUpperCase().search(value.toUpperCase()) >= 0
      || paquete.codpaquete.toUpperCase().search(value.toUpperCase()) >= 0);
    this.listaPaquetes = result;
  } else {
    this.listaPaquetes = this.listaPaquetesAux;
  }
}
```

Figura 38 Buscador de Paquetes

Elaborado por: Anderson Naranjo

Lector de código de barras

El método permite abrir la cámara y escáner un código de barras para realizar la búsqueda de algún paquete, el plugin utilizado para este proceso es Barcode Scanner el cual puede ser instalado por npm con los siguientes comandos “**npm install phonegap-plugin-barcode-scanner**” y “**npm install @ionic-native/barcode-scanner**”

```
leerCodigoBarras() {
  this.barcodeScanner.scan().then(barcodeData => {
    this.msjConsola2('Barcode data', barcodeData);
    this.database.getCodigo(barcodeData.text).then((data) => {
      if (data != 0) {
        this.presentModal(data, barcodeData.text);
      } else {
        this.presentToast('No se encontro ningun Paquete');
      }
    });
  }, (error) => {
    this.msjConsola(error);
    this.database.GuardarError('c023::' + error);
  });
}).catch(err => {
  this.msjConsola2('Error', err);
  this.database.GuardarError('c023::' + err);
});
});
```

Figura 39 Lector de código de barras

Elaborado por: Anderson Naranjo

Obtener ubicación

El método permite obtener la ubicación actual del dispositivo en caso de no ser posible obtener ninguna ubicación, en método tomara al ultima ubicación que haya sido registrada en la base de datos del dispositivo, esto se realiza gracias al plugin geolocation el cual ya viene integrado con capacitor y no requiere de ninguna instalación adicional.

```
await Geolocation.getCurrentPosition({ enableHighAccuracy: true, maximumAge: 86400000, timeout: 60000 }).then((resp) => {
  this.lat = resp.coords.latitude;
  this.lon = resp.coords.longitude;
})
if (this.lat == null) {
  this.database.getLat().then((data) => {
    this.lat = data;
  }, (error) => {
    console.log(error);
  });
  this.database.getLon().then((data) => {
    this.lon = data;
  }, (error) => {
    console.log(error);
  });
} else {
  this.database.actualizarPos(this.lat, this.lon);
}
}).catch((error) => {
  console.log('error locatio', error);
  this.database.getLat().then((data) => {
    this.lat = data;
  }, (error) => {
    console.log(error);
  });
  this.database.getLon().then((data) => {
    this.lon = data;
  }, (error) => {
    console.log(error);
  });
});
```

Figura 40 Obtener ubicación

Elaborado por: Anderson Naranjo

Realizar llamada telefónica

El método permite realizar llamadas telefónicas directamente desde la aplicación gracias al plugin Call Number el cual se instala por npm con los comandos “**npm install call-number**” y “**npm install @ionic-native/call-number**”

```
callNow(number) {  
  this.callNumber.callNumber(number, true)  
    .then(res => console.log('Launched dialer!', res))  
    .catch(err => console.log('Error launching dialer', err));  
}
```

Figura 41 Realizar llamada telefónica

Elaborado por: Anderson Naranjo

Realizar fotografía

El método permite realizar una fotografía desde la aplicación gracias a la utilización del plugin camera el cual ya viene incluido con capacitor y no requiere de ninguna instalación.

```
const options: CameraOptions = {  
  quality: 30,  
  responseType: CameraResultType.Base64,  
  saveToGallery: true,  
  source: CameraSource.Camera  
}  
Camera.getPhoto(options).then((imageData) => {  
  console.log('img hacer', imageData);  
  let fotoAux = 'data:image/jpeg;base64,' + imageData.base64String;  
  if (this.foto.length == 0) {  
    this.foto = fotoAux;  
    //this.guardarGaleria(imageData, imageData);  
  } else if (this.foto2.length == 0) {  
    this.foto2 = fotoAux;  
    //this.guardarGaleria(imageData, imageData);  
  } else if (this.foto3.length == 0) {  
    this.foto3 = fotoAux;  
    // this.guardarGaleria(imageData, imageData);  
  }  
  
  console.log('Tamaño Foto: ' + this.foto.length);  
}, (err) => {  
  console.log(err);  
  this.database.GuardarError('md001:' + err);  
});
```

Figura 42 Realizar fotografía

Elaborado por: Anderson Naranjo

Seleccionar imagen de galería

El método permite seleccionar una imagen directamente desde la galería del dispositivo, se utiliza el plugin camera al cual se le configura la opción “source” para que pueda extraer las imágenes de la galería del dispositivo.

```
SelecFoto() {  
  const options: CameraOptions = {  
    quality: 30,  
    responseType: CameraResultType.Base64,  
    source: CameraSource.Photos  
  }  
  Camera.getPhoto(options).then((imageData) => {  
    let fotoAux = 'data:image/jpeg;base64,' + imageData.base64String;  
    if (this.foto.length == 0) {  
      this.foto = fotoAux;  
    } else if (this.foto2.length == 0) {  
      this.foto2 = fotoAux;  
    } else if (this.foto3.length == 0) {  
      this.foto3 = fotoAux;  
    }  
  })  
  
  console.log('Tamaño Foto: ' + this.foto.length);  
}, (err) => {  
  console.log(err);  
  this.database.GuardarError('md002:' + err);  
});  
}
```

Figura 43 Seleccionar imagen de galería

Elaborado por: Anderson Naranjo

Guardar nueva actualización

El método permite guardar un registro de una actualización de un paquete determinado, esto después de realizar las validaciones correspondientes.

```

GuardarACT() {
  (<HTMLElement>(document.getElementById('btnGuardarACT'))).disabled = true;
  if (this.est == '00' || this.est == '02') {
    this.presentToast('Seleccione un nuevo estado del paquete');
    console.log(this.est);
    (<HTMLElement>(document.getElementById('btnGuardarACT'))).disabled = false;
  } else {
    //getCodigosPaquetes
    if (this.foto.length > 0) {
      this.database.getCodigosPaquetes(this.idP, this.codpaq).then((data) => {
        console.log('Codigos', data);
        for (var i = 0; i < data.length; i++) {
          this.GuardadoCod(data[i].toString().split(',')[0], data[i].toString().split(',')[1]);
        }
        (<HTMLElement>(document.getElementById('btnGuardarACT'))).disabled = false;
        this.regresarEstado();
      }, (error) => {
        console.log(error);
        (<HTMLElement>(document.getElementById('btnGuardarACT'))).disabled = false;
        this.database.GuardarError('md005:' + error);
      });
    } else {
      this.presentToast('Añada minimo 1 fotografia');
      (<HTMLElement>(document.getElementById('btnGuardarACT'))).disabled = false;
    }
  }
}
}

```

Figura 44 Guardar nueva actualización

Elaborado por: Anderson Naranjo

Cargar mapa y añadir marcadores

El método carga la vista del mapa y añade los marcadores con su respectiva información.

Marcaje de ruta

El método marca sobre el mapa una rota sugerida desde la ubicación actual del dispositivo hacia un paquete seleccionado.

```
calculateAndDisplayRoute() {
  var paqueteUsar= this.listaPuntos.find(element => element.codigo == this.codigoClk);

  this.directionsService.route(
    {
      origin: { lat: this.lat, lng: this.lon },
      destination: { lat: Number.parseFloat( paqueteUsar.paqentrega_coordx), lng: Number.parseFloat(paqueteUsar.paqentrega_coordy) },
      travelMode: google.maps.TravelMode.DRIVING,
    },
    (response, status) => {
      if (status == "OK") {
        this.directionsRenderer.setDirections(response);
      } else {
        window.alert("Error al trazar la ruta");
      }
    }
  );
}
```

Figura 47 Marcaje de ruta

Elaborado por: Anderson Naranjo

Cerrar Sesión

El método elimina el usuario de la base de datos local y comunica a la API para que se realice la eliminación del token.

```
this.database.actualizarUSRDIS(this.UsuarioLog.usuario_id)
  .then(_ => {

    this.serUsuario.UsuarioDelSession(this.UsuarioLog.token).subscribe(
      res => {

        this.msjConsola('usr actualizado dis');
        this.MostrarLogin();
      }
    );
  });
```

Figura 48 Cerrar Sesión

Elaborado por: Anderson Naranjo

Estado de la conexión

El método se mantiene escuchando el estado actual de la conexión del dispositivo, se utiliza el plugin network el cual viene integrado con capacitor y no requiere de ninguna instalación.

```

this.networkListener = Network.addListener('networkStatusChange', (status) => {
  console.log("Network status changed", status);
  this.networkStatus = status;
  this.isConnected = status.connected;
});

```

Figura 49 Estado de la conexión

Elaborado por: Anderson Naranjo

3.2.6 Fase 5: Pruebas

Tabla 45 Prueba de aceptación 1

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 1
N° de Historia de usuario: 003	
Nombre: Ingreso a la aplicación.	
Descripción: Un módulo de ingreso mediante un usuario y contraseña es fundamental para la protección de la información de los usuarios.	
Condiciones de ejecución: La aplicación muestra una pantalla de inicio de sesión en la cual se muestran los campos de usuario y contraseña.	
Interfaz: La pantalla muestra el logo de la empresa y los campos para el ingreso de usuario y contraseña, además de un botón el cual desencadena la acción.	
Resultado esperado: Al ingresar las credenciales correctas el usuario podrá visualizar la pantalla principal, caso contrario el usuario dispondrá de 5 intentos después de los cuales se bloqueará el acceso por 20 segundos.	
Resultado de la prueba: Prueba satisfactoria.	

Tabla 46 Prueba de aceptación 2

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 2
N° de Historia de usuario: 004	
Nombre: Pantalla de inicio.	
Descripción: Pantalla principal de la aplicación en la cual se mostrará el listado de los paquetes a ser entregados por el usuario.	
Condiciones de ejecución: La aplicación muestra una pantalla principal donde muestra el listado de los paquetes a ser entregados por el usuario.	
Interfaz: La pantalla muestra información básica de los paquetes pertenecientes al usuario.	
Resultado esperado: Al ingresar a la aplicación se deberán cargar automáticamente un listado de los paquetes que correspondan a ese usuario, el listado deberá mostrar información básica de los paquetes.	
Resultado de la prueba: Prueba satisfactoria.	

Tabla 47 Prueba de aceptación 3

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 3
N° de Historia de usuario: 005	
Nombre: Buscador de paquetes	
Descripción: Modulo el cual permita filtrar los paquetes mediante código, cliente, zona, sección, etc. Además de permitir escanear el código de barras de los paquetes para una búsqueda más rápida.	
Condiciones de ejecución: La aplicación muestra un campo para ingresar información por la cual deseo que se filtre la información de los paquetes, además de un botón para abrir la cámara y escanear el código de barras.	
Interfaz: En la parte superior de la pantalla se muestra el respectivo campo para ingresar la información para realizar el filtrado, el filtrado se realiza según el usuario ingrese la información. En la parte inferior se muestra un botón que permite ingresar a cámara del dispositivo, en la cámara se muestra un recuadro el cual deberá ser alineado con el código de barras.	
Resultado esperado: Al ingresar información para realizar el filtrado el listado de paquetes se deberá actualizar automáticamente. Al escáner el código de barras se realizará la búsqueda automáticamente y de ser encontrado el paquete se mostrará toda la información automáticamente.	
Resultado de la prueba: Prueba satisfactoria.	

Tabla 48 Prueba de aceptación 4

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 4
N° de Historia de usuario: 006	
Nombre: Pantalla de detalle del paquete	
Descripción: Pantalla en la cual se mostrará la información completa del paquete la cual incluye información del cliente, cantidad y detalle de objetos fuera de caja e información de la ubicación	
Condiciones de ejecución: Al seleccionar uno de los paquetes a ser entregados se mostrar una nueva pantalla en la cual se mostrará toda la información del paquete a entregarse.	
Interfaz: La pantalla se dividirá en tres secciones, la primera sección contiene la información del cliente además de sus números telefónicos. La segunda sección mostrara información mas detallada del paquete a ser entregado. La tercera sección muestra detalles de los productos fuera de caja que correspondan a la entrega.	
Resultado esperado: Al seleccionar uno de los paquetes se mostrará la información del cliente, paquetes y productos fuera de caja del paquete seleccionado.	
Resultado de la prueba: Prueba satisfactoria.	

Tabla 49 Prueba de aceptación 5

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 5
N° de Historia de usuario: 007	
Nombre: Cambio de estado de paquete	
Descripción: Modulo el cual permita seleccionar cualquiera de los estados posibles de los paquetes, los estados incluyen entregado, nadie en casa, siniestrado, etc. Además de añadir una observación conjuntamente con mínimo una fotografía y máximo 3.	
Condiciones de ejecución: Al seleccionar uno de los paquetes a ser entregados se le podrá realizar un cambio de estado para lo cual se requiere de una observación, de mínimo una fotografía y de la ubicación.	
Interfaz: La pantalla muestra un ion-select en donde el usuario podrá seleccionar el nuevo estado del producto, también se cuenta con un campo de texto para el ingreso de la observación, dos botones que permiten tomar una fotografía o seleccionarla de la galería del dispositivo y por último se muestra información de la ubicación del dispositivo.	
Resultado esperado: Al intentar cambiar el estado de un paquete la aplicación deberá verificar que se haya añadido como mínimo una fotografía y máximo 3, la observación será de carácter opcional y por último se deberá verificar que se obtuvo la ubicación del dispositivo.	
Resultado de la prueba: Prueba satisfactoria.	

Tabla 50 Prueba de aceptación 6

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 6
N° de Historia de usuario: 009	
Nombre: Sincronización de entregas	
Descripción: Modulo que permita sincronizar la información almacenada localmente, mostrando la cantidad de registros a sincronizar y mostrar el progreso de la sincronización.	
Condiciones de ejecución: Una vez realizado el cambio de estado de uno o más paquetes se podrá enviar la información almacenada en el dispositivo a la API para ser almacenada en la base de datos de la empresa.	
Interfaz: En la parte superior derecha de la aplicación se mostrará un indicador de color rojo con la cantidad de registros a ser sincronizados además de permitir el ingreso al menú de sincronización, una vez decidida realizar la sincronización en la pantalla se mostrará una barra de carga con el porcentaje de sincronización.	
Resultado esperado: La aplicación deberá enviar toda la información a la API la cual la validara y la ingresara en la base de datos de la aplicación, las imágenes serán almacenadas en un directorio específico y en la aplicación mostrara que no existen registros a sincronizar.	
Resultado de la prueba: Prueba satisfactoria.	

Tabla 51 Prueba de aceptación 7

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 7
N° de Historia de usuario: 010	
Nombre: Marcadores de paquetes en Mapa	
Descripción: Mostrar un mapa donde se muestren los marcadores de la ubicación de cada paquete. Los marcadores deberán mostrar información básica de los clientes.	
Condiciones de ejecución: La aplicación mostrará una pantalla en la cual se encontrará un mapa y los distintos marcadores de las ubicaciones de los paquetes.	
Interfaz: La interfaz presenta un mapa en el cual se encuentran marcadores que representan la ubicación a entregarse los distintos paquetes, cada uno de los marcadores desplegara información básica del paquete al que representan.	
Resultado esperado: La interfaz deberá cargar exitosamente el mapa en la pantalla y colocar los distintos marcadores en los lugares correspondientes.	
Resultado de la prueba: Prueba satisfactoria.	

Tabla 52 Prueba de aceptación 8

Elaborado por: Anderson Naranjo

Prueba de aceptación	Número: 8
N° de Historia de usuario: 011	
Nombre: Marcaje de ruta	
Descripción: Mostrar una ruta desde la ubicación actual del usuario hacia la ubicación de entrega del paquete designado.	
Condiciones de ejecución: La aplicación presenta un botón en el cual calcula y muestra una ruta desde el lugar actual del dispositivo hacia uno de los lugares de entrega seleccionados.	
Interfaz: En la parte superior izquierda del mapa se muestra un botón el cual realiza el traseo de la ruta sobre el mapa.	
Resultado esperado: La interfaz deberá mostrar una ruta desde el lugar actual del dispositivo hacia uno de los lugares de entrega seleccionados.	
Resultado de la prueba: Prueba satisfactoria.	

CAPITULO IV.-CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

Actualmente estamos sufriendo una transformación digital que se ve acelerada gracias a la incidencia del COVID-19, por lo que muchas empresas han tomado la decisión de implementar tecnología para la administración de sus procesos y aumentar la eficiencia de los servicios que brindan a sus clientes.

Al utilizar IONIC como tecnología de desarrollo multiplataforma reducimos los costes y aumentamos la productividad. El desarrollo de una aplicación para distintas plataformas como son Android e iOS basándose en un único código genera menor tiempo de desarrollo, hace que su mantenimiento y escalabilidad sean mucho más sencillos y de menor costo que de las aplicaciones nativas.

IONIC al ser un proyecto de código abierto mantiene una comunidad de desarrolladores muy amplia, por lo cual resulta muy fácil encontrar ayuda acerca de determinados errores y problemas que puedan ocurrir en el proceso de desarrollo.

Al realizar la implementación de la aplicación y realizar diversas pruebas se pudo identificar errores concretos que gracias a la utilización de la metodología ágil XP se pudieron solucionar de manera rápida y satisfactoria. El almacenamiento de las imágenes directamente en la base de datos provocó un incremento muy rápido del tamaño de la tabla de entregas generando que a futuro pueda reducir el rendimiento de las consultas por lo que se tomó la decisión de almacenar las imágenes en un directorio específico.

4.2 Recomendaciones

Actualmente la evolución de la tecnología provoca en los usuarios el deseo y la necesidad de incorporar funcionalidades que faciliten sus labores cotidianas, todo esto implica que los desarrolladores se mantengan actualizados en la implementación de nuevas tecnologías como es el caso de Ionic, la cual facilita el desarrollo de nuevas funcionalidades acorde con las necesidades de los usuarios actuales.

Se recomienda a la empresa el cambio de equipos telefónicos o adquirir equipos mas modernos con un nivel de API 26 (Android 8) o posteriores ya que muchos de los empleados poseen equipos en los cuales se presentaron problemas con respecto a la utilización de la aplicación.

Implementar un certificado SSL para una comunicación segura entre los clientes y el servidor, debido a que se utiliza JWT la transmisión debe realizarse a través de HTTPS para que los datos estén protegidos.

El uso constante de la aplicación genera nueva información o datos de los usuarios y estos deben ser actualizados en la base de datos, además de ello se recomiendo dar capacitaciones a los empleados sobre nuevas tecnologías con el fin de dar un apoyo básico en la enseñanza de manejo de nuevos dispositivos tecnológicos.

Para finalizar a la empresa se le recomiendo la implementación de controladores de versiones con la finalidad de que la aplicación siga mejorando y actualizando sus componentes previniendo futuros problemas o perdida de información esto permitirá a la empresa seguir evolucionando y con ello convertirse en una empresa amigable con la tecnología.

BIBLIOGRAFÍA

- [1] C. R. G. V. P. Yepes, “Logística de Almacenamiento Como Factor de Desarrollo Competitivo de la Empresa Amazon,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2019, doi: 10.1017/CBO9781107415324.004.
- [2] Y. G. Garzón, R. D. E. Jesus, and A. Robles, “Salida intercultural Estados Unidos logística empresarial Servientrega,” <https://repository.ucatolica.edu.co/handle/10983/24194>, 2019.
- [3] N. Velasquez Abril and N. I. Mora Arciniegas, “Internacionalización de operadores logísticos casos exitosos: SERVIENTREGA FEDEX DEUTSCHE POST - DHL,” 2013, [Online]. Available: <http://repository.urosario.edu.co/handle/10336/4551>.
- [4] V. Z. Á. Humberto, “La TECNOLOGÍA ANDROID y su incidencia en el desarrollo de una aplicación móvil para la geo-localización de los centros asistenciales y farmacias de turnos para la DIRECCIÓN PROVINCIAL DE SALUD LOS RÍOS ubicada en la ciudad de Babahoyo,” 2016, [Online]. Available: <http://dspace.utb.edu.ec/bitstream/handle/49000/1206/T-UTB-FAFI-SIST-000113.pdf;jsessionid=ABD0F742478F4FBA916CB6EED63FDF63?sequence=1>.
- [5] N. D. Lisandro and P. J. Thomas, “Desarrollo de Aplicaciones Móviles Multiplataforma,” *Univ. Nac. La Plata - Fac. Informática*, p. 84, 2017, [Online]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/60497/Documento_completo___.pdf-PDFA.pdf?sequence=3.
- [6] P. Thomas, N. Galdamez, L. Delia, L. Corbalán, and P. Pesado, “Dispositivos Móviles: Desarrollo de Aplicaciones Multiplataforma,” *XVII Work. Investig. en Ciencias la Comput.*, no. 1, p. 4, 2015, [Online]. Available: <http://digital.cic.gba.gob.ar/handle/11746/2187>.
- [7] M. C. Fennema *et al.*, “Aproximaciones para el desarrollo multiplataforma y

- mantenimiento de aplicaciones móviles,” pp. 446–450, 2017, [Online]. Available: <http://sedici.unlp.edu.ar/handle/10915/61922>.
- [8] “Ionic,” 2020. <http://ionicframework.com/docs/guide/pre%0Aface.html>.
- [9] Apache, “Cordova,” 2021. <https://cordova.apache.org/docs/en/latest/>.
- [10] ionic O. Source, “Capacitor,” 2021. <https://capacitorjs.com/>.
- [11] Facebook, “React Native,” 2019. .
- [12] Google, “Flutter,” 2021. <https://flutter.dev/docs>.
- [13] R. Hat, “What are application programming interfaces,” 2021. <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>.
- [14] J. Sayago H., E. Flores C., and A. Recalde, “Análisis Comparativo entre los Estándares Orientados a Servicios Web SOAP, REST y GRAPHQL (Comparative Analysis between Standards Oriented to Web Services SOAP, REST and GRAPHQL),” *Rev. Antioqueña las Ciencias Comput. y la Ing. Softw.*, vol. 9, no. March 2020, pp. 10–22, 2019, doi: 10.5281/zenodo.3592004.
- [15] C. Muyón and F. Montaluisa, “Information security methods to protect rest web services communication and data in http requests using json web token and keycloak red hat single sign on | Métodos de seguridad de la información para proteger la comunicación y los datos de servicios web,” *Risti*, vol. 2020, no. E29, pp. 198–213, 2020.
- [16] R. Hat, “API security,” 2021. <https://www.redhat.com/es/topics/security/api-security>.
- [17] E. Salas, “Aplicando seguridad a una API REST con JSON Web Tokens,” 2020, [Online]. Available: <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/126696/1/emsalasGTFM2020.pdf>.
- [18] Auth0, “JWT.io,” 2021. <https://jwt.io/>.

- [19] R. Perales, “Sistemas de Autenticación,” 2017.
- [20] Google, “Google Maps Platform,” 2021.
<https://developers.google.com/maps/documentation/>.
- [21] E. G. Maida and J. Pacienza, “METODOLOGIAS DE DESARROLLO DE SOFTWARE,” 2015, [Online]. Available:
<http://bibliotecadigital.uca.edu.ar/repositorio/tesis/metodologias-desarrollo-software.pdf>.
- [22] P. D. G. Gudiño, “ANÁLISIS DE LAS METODOLOGÍAS ÁGILES Y SU INCIDENCIA EN LA CREACIÓN DEL PORTAFOLIO DE SERVICIO PARA LA UNIDAD DE EXTENSIÓN UNIVERSITARIA DE LA UNIVERSIDAD TÉCNICA DEL NORTE DE LA CIUDAD DE IBARRA,” p. 6, 2015, [Online]. Available:
<https://repositorio.uta.edu.ec/jspui/handle/123456789/23055>.
- [23] Ing. Jorge Eduardo Chapaca Garzón, “Gerencia de TI orientada a la aplicación de metodologías ágiles en la gestión de proyectos de tecnologías de información (TI),” 2011, [Online]. Available:
<https://repositorio.uta.edu.ec/jspui/handle/123456789/30568>.

ANEXOS Y APÉNDICES

ANEXO A

Cuestionario implementado para la obtención de información dirigido hacia el Ing. Miguel Miniguano jefe del departamento de sistemas de la empresa ESFLOSERV CIA. LTDA

N	Pregunta
1.	¿Hace uso de aplicaciones de reparto regularmente?
2.	¿La empresa incentiva el uso de las TIC's?
3.	¿Conoce de que se trata la tecnología de aplicación móvil multiplataforma?
4.	¿Usaría la aplicación móvil multiplataforma para gestión y control de entregas?
5.	¿Su empresa posee las herramientas necesarias para implementar una aplicación móvil multiplataforma para la gestión y control de entregas de los portadores?
6.	¿Desearía que se destine un financiamiento para la implementación aplicación móvil multiplataforma para la gestión y control de entregas de los portadores?
7.	¿Cree que la metodología tradicional de entregas pueda mejorar con la ayuda de una aplicación móvil multiplataforma para la gestión y control de entregas de los portadores?
8.	¿A su criterio el uso de una aplicación móvil multiplataforma para la gestión y control de entregas de los portadores podría desarrollar nuevos recursos y convertir a la empresa ESFLOSERV CIA LTDA en pionera en el servicio de entregas?
9.	¿Apoyaría la implantación de la aplicación móvil multiplataforma para la gestión y control de entregas de los portadores para la empresa ESFLOSERV CIA LTDA?
10.	¿Cuál es el proceso que llevan a cabo los portadores a la hora de entregar los distintos paquetes?

ANEXO B

Manual de usuario

Requisitos

- Versión de Android mínima 7
- Almacenamiento mínimo 60MB

Acceso a la aplicación

Una vez instalada la aplicación se podrá acceder directamente desde su icono.



Ingreso a la aplicación



La aplicación presenta una pantalla de inicio de sesión con el siguiente formulario de datos:

- 1) Nombre de usuario a ingresar en la aplicación.
- 2) Contraseña del usuario
- 3) Botón de ingreso

Pantalla principal



La pantalla principal consta de los siguientes elementos fundamentales:

- 1) Indicador de registros a sincronizar y menú de sincronización.
- 2) Barra de búsqueda de paquetes
- 3) Acceso al escáner de código de barras
- 4) Listado de paquetes a entregarse.
- 5) Botón de acceso al Mapa

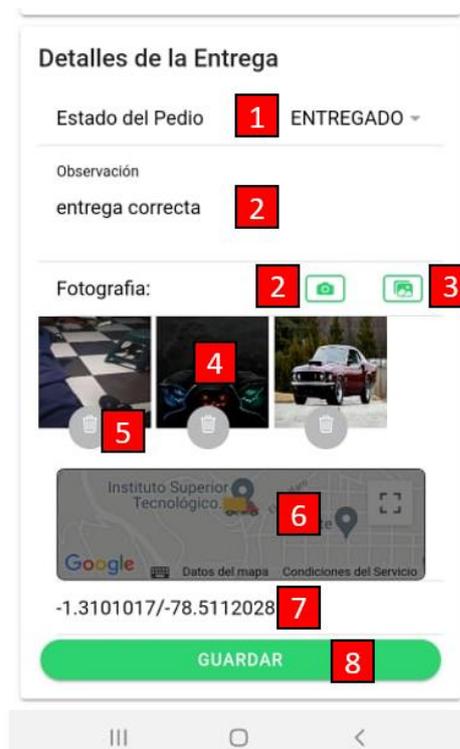
Menú se sincronizado



El menú de sincronizado dispone de varias opciones entre las cuales se encuentra:

- 1) Nombre del usuario.
- 2) Versión de la aplicación.
- 3) Opción de sincronización de información.
- 4) Opción de actualización del listado de paquetes.
- 5) Botón de cierre de sesión.

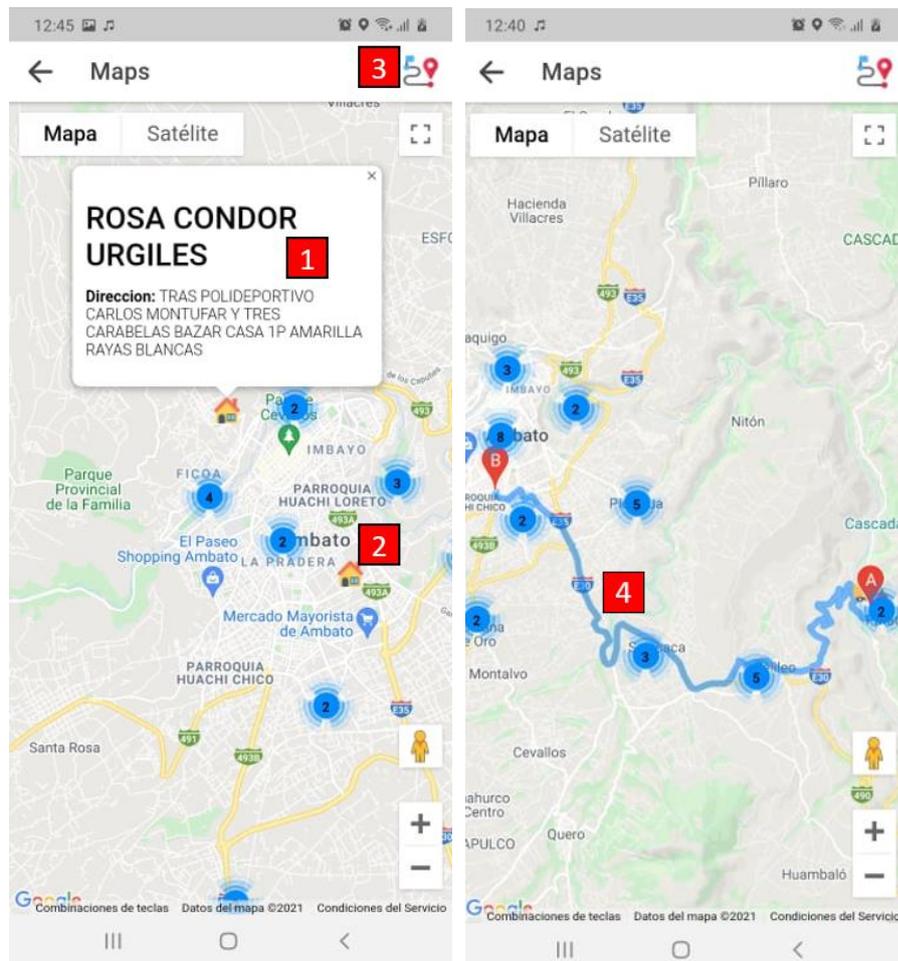
Cambio de estado de paquete



La pantalla de cambio de estado de paquetes dispone de los campos necesarios para el ingreso de la información.

- 1) Selector del nuevo estado.
- 2) Campo de ingreso de observación (Opcional).
- 3) Botón de acceso a cámara.
- 4) Botón de selección de imágenes de galería.
- 5) Botón para eliminar fotografías.
- 6) Mapa de la ubicación actual del dispositivo.
- 7) Coordenadas Latitud y longitud.
- 8) Botón de Guardado.

Mapa



La aplicación muestra un mapa con diversos marcadores ubicados en los lugares de entrega, además de graficar una ruta sugerida sobre el mapa.

- 1) Información básica de la entrega.
- 2) Marcador de ubicación.
- 3) Botón de tráfico de la ruta.
- 4) Ruta.