



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA SOFTWARE

DESARROLLO DE UNA APLICACIÓN MÓVIL Y BACKEND
PARA APOYAR A LOS CONDUCTORES EN LA GESTIÓN DE
REPARTO DE PAQUETERÍA EN LA EMPRESA "CORPORACIÓN
WOLF S.A."

Trabajo de Integración Curricular

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO DE SOFTWARE

AUTORES:

JOSÉ VICENTE OCHOA VALAREZO

EDWARD LEONARDO BONE CUSME

Riobamba – Ecuador

2022



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA SOFTWARE

DESARROLLO DE UNA APLICACIÓN MÓVIL Y BACKEND
PARA APOYAR A LOS CONDUCTORES EN LA GESTIÓN DE
REPARTO DE PAQUETERÍA EN LA EMPRESA "CORPORACIÓN
WOLF S.A."

Trabajo de Integración Curricular

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO DE SOFTWARE

AUTORES: JOSÉ VICENTE OCHOA VALAREZO

EDWARD LEONARDO BONE CUSME

DIRECTORA: Ing. GISEL KATERINE BASTIDAS GUACHO M.Sc

Riobamba – Ecuador

2022

© 2022, Edward Leonardo Bone Cusme, José Vicente Ochoa Valarezo

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo cita bibliográfica del documento, siempre y cuando se reconozca el Derecho del Autor.

Nosotros, EDWARD LEONARDO BONE CUSME Y JOSÉ VICENTE OCHOA VALAREZO, declaramos que el presente trabajo de titulación es de nuestra autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autores asumimos la responsabilidad legal y académica de los contenidos de este trabajo de titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 24 de junio de 2022



José Vicente Ochoa Valarezo
070644258-9



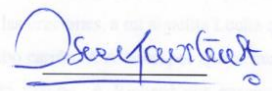


Edward Leonardo Bone Cusme
080415804-6

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

CARRERA SOFTWARE

El Tribunal del Trabajo de Integración Curricular certifica que: El Trabajo de Integración Curricular; tipo: Proyecto Técnico, **DESARROLLO DE UNA APLICACIÓN MÓVIL Y BACKEND PARA APOYAR A LOS CONDUCTORES EN LA GESTIÓN DE REPARTO DE PAQUETERÍA EN LA EMPRESA "CORPORACIÓN WOLF S.A."**, realizado por los señores: **JOSÉ VICENTE OCHOA VALAREZO** y **EDWARD LEONARDO BONE CUSME**, ha sido minuciosamente revisado por los Miembros del Trabajo de Integración Curricular, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	FIRMA	FECHA
Ing. Óscar Danilo Gavilánez Álvarez PRESIDENTE DEL TRIBUNAL		2022-06-24
Ing. Gisel Katerine Bastidas Guacho M.Sc DIRECTORA DEL TRABAJO DE INTEGRACIÓN CURRICULAR		2022-06-24
Ing. Ivonne Elizabeth Rodríguez Flores Ph.D MIEMBRO DEL TRIBUNAL		2022-06-24

DEDICATORIA

Este trabajo de integración curricular lo dedico a mis padres, Carmen Cusme y Victor Bone por su gran esfuerzo que han realizado durante toda mi vida universitaria anteponiendo mis necesidades antes que la suyas, en especial a mi madre que siempre confió en mí sin importar las circunstancias de la vida. A mis hermanas que siempre creyeron en mí y me apoyan a seguir adelante. A mis sobrinos que con sus sonrisas me motivaban a terminar mi carrera y me enseñaron a disfrutar de lo simple de la vida. A mis amigos José, Andrea, Ayza, Michelle, Brandon, Jhonnatan, Brayan que, entre juegos, desveladas y clases nos apoyábamos para superar cualquier reto para poder terminar la carrera.

Leonardo

El presente trabajo lo dedico a mis padres, Vicente Ochoa y Elida Valarezo por apoyarme y confiar en mí al momento de elegir esta carrera. A mi hermana por animarme ante las adversidades. A mi abuelita Rosa por todas las oraciones, a mi abuelita Lucha que, aunque no me pudo ver cumplir esta meta, me brindo mucho cariño. A Alisson por apoyarme en los momentos difíciles y tenerme paciencia durante tanto tiempo. A Richard por enseñarme a desarrollar software y ayudarme a crecer en el ámbito profesional. Y finalmente a mis amigos Leonardo, Andrea, Michelle, Brandon, Jhonnatan, Luis, Brayan que entre risas, peleas y enojos me apoyaron para poder llegar hasta aquí. Muchas gracias a todos. ¡Lo logre!

José

AGRADECIMIENTO

Nuestros más sinceros agradecimientos:

A la Escuela Superior Politécnica de Chimborazo y sus docentes quienes nos ayudaron a formarnos como personas y profesionales, compartiendo sus enseñanzas para mejorar cada día.

A la empresa “Corporación Wolf S.A.” por brindarnos la oportunidad de ser parte del desarrollo del sistema “Wolf Delivery” y afrontar un reto de la vida profesional.

A nuestra Directora de Tesis Ing. Gisel Bastidas y asesora Dra. Ivonne Rodríguez, quienes nos guiaron con sus conocimientos y experiencia durante todo el proceso de titulación para terminar satisfactoriamente el trabajo de integración curricular.

A nuestros familiares por apoyarnos incondicionalmente y siempre creer en nosotros.

Y finalmente al Ing. Omar Gómez, por dedicar un poco de su tiempo para atender nuestras consultas sobre la medición de métricas de calidad de software.

José, Leonardo

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	x
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE GRÁFICOS.....	xiii
ÍNDICE DE ANEXOS.....	xiv
RESUMEN	xv
SUMMARY	xvi
INTRODUCCIÓN	1

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA.....	2
1.1. Antecedentes	2
1.2. Formulación del problema.....	3
1.3. Sistematización del problema	3
1.4. Justificación del trabajo de titulación	3
1.4.1. <i>Justificación teórica</i>	4
1.4.2. <i>Justificación aplicada</i>	5
1.5. Objetivos	6
1.5.1. <i>Objetivo general</i>	6
1.5.2. <i>Objetivos específicos</i>	6

CAPÍTULO II

2. FUNDAMENTOS TEÓRICOS	7
2.1. Casilleros inteligentes.....	7
2.1.1. <i>Tipos de casilleros inteligentes</i>	7
2.1.2. <i>Paquete</i>	8
2.1.3. <i>Gestión de Paquetería</i>	8
2.2. Estándar BPMN	8
2.3. Aplicación móvil	9
2.4. Backend.....	10
2.4.1. <i>Servicios web</i>	10
2.4.1.1. <i>Servicio web REST</i>	11

2.5.	Requisitos de Software	11
2.5.1.	<i>Especificación de Requisitos Software</i>	<i>12</i>
2.6.	Metodología de desarrollo de software	12
2.6.1.	<i>Metodología SCRUM.....</i>	<i>13</i>
2.6.2.	<i>Proceso de aplicación de Scrum</i>	<i>13</i>
2.6.2.1.	<i>Sprint Planning.....</i>	<i>14</i>
2.6.2.2.	<i>Sprint</i>	<i>14</i>
2.6.2.3.	<i>Daily Scrum</i>	<i>14</i>
2.6.2.4.	<i>Sprint Review.....</i>	<i>14</i>
2.6.2.5.	<i>Sprint Retrospective.....</i>	<i>15</i>
2.6.3.	<i>Roles de Scrum.....</i>	<i>15</i>
2.6.3.1.	<i>Equipo de desarrollo.....</i>	<i>15</i>
2.6.3.2.	<i>Product Owner.....</i>	<i>15</i>
2.6.3.3.	<i>Scrum Master.....</i>	<i>16</i>
2.7.	Herramientas para desarrollo de software	16
2.7.1.	<i>Herramientas para el flujo de trabajo</i>	<i>16</i>
2.7.1.1.	<i>Slack</i>	<i>16</i>
2.7.1.2.	<i>GitHub</i>	<i>17</i>
2.7.2.	<i>Tecnología para el backend.....</i>	<i>17</i>
2.7.2.1.	<i>PostgreSQL.....</i>	<i>17</i>
2.7.2.2.	<i>TypeScript.....</i>	<i>17</i>
2.7.2.3.	<i>NestJS</i>	<i>18</i>
2.7.3.	<i>Tecnología para el frontend.....</i>	<i>18</i>
2.7.3.1.	<i>React Native.....</i>	<i>18</i>
2.7.3.2.	<i>Expo.....</i>	<i>19</i>
2.8.	Calidad del producto software	20
2.8.1.	<i>Mantenibilidad.....</i>	<i>20</i>
2.8.1.1.	<i>Capacidad para ser modificado.....</i>	<i>21</i>
2.8.2.	<i>Eficiencia de desempeño.....</i>	<i>23</i>
2.9.	Trabajos relacionados.....	24

CAPÍTULO III

3.	MARCO METODOLÓGICO.....	26
3.1.	Diseño de la investigación	26
3.1.1.	<i>Tipo de investigación</i>	<i>26</i>

3.1.2. Métodos de investigación	26
3.1.2.1. Método Analítico.....	27
3.1.2.2. Método Inductivo	27
3.1.2.3. Método Sintético	27
3.1.3. Técnicas de Investigación	27
3.1.3.1. Entrevista.....	27
3.1.3.2. Observación.....	28
3.1.3.3. Revisión de documentación.....	28
3.1.4. Variables e Indicadores	29
3.1.4.1. Operacionalización conceptual de variables	30
3.1.4.2. Operacionalización metodológica de variables	31
3.1.5. Población y muestra	32
3.1.5.1. Eficiencia de desempeño.....	32
3.1.5.2. Facilidad de modificación	33
3.2. Proceso de reparto de la empresa	33
3.2.1. Proceso actual de reparto de paquetería.....	33
3.2.2. Proceso de reparto de paquetería con el sistema implementado	34
3.3. Estudio de factibilidad	35
3.3.1. Facilidad técnica.....	36
3.3.2. Facilidad operativa	37
3.3.3. Factibilidad económica.....	38
3.3.3.1. Costos.....	38
3.3.3.2. Beneficios.....	38
3.4. Desarrollo de la aplicación mediante SCRUM	39
3.4.1. Fase de planificación	39
3.4.1.1. Miembros y Roles involucrado en el proyecto	39
3.4.1.2. Tipos de usuarios y Roles involucrados en el sistema.....	40
3.4.1.3. Especificación de requerimientos	40
3.4.1.4. Product Backlog	43
3.4.2. Fase de diseño.....	45
3.4.2.1. Diseño de la arquitectura	45
3.4.2.2. Organigrama de los módulos de del sistema	46
3.4.2.3. Diagrama de despliegue	47
3.4.2.4. Estándar de codificación	48
3.4.2.5. Diseño de base de datos.....	48
3.4.2.6. Diccionario de datos.....	49

3.4.2.7. <i>Diseño de interfaces de usuario</i>	50
3.4.3. Fase de desarrollo	51
3.4.3.1. <i>Sprint Backlog</i>	51
3.4.3.2. <i>Historia de Usuario</i>	53

CAPÍTULO IV

4. MARCO DE RESULTADOS	55
4.1. Criterios de evaluación	55
4.2. Facilidad de modificación	56
4.2.1. <i>Obtención de datos con la herramienta SonarQube</i>	56
4.2.2. <i>Datos generales del código</i>	58
4.2.3. <i>Datos de la facilidad de modificación en el backend</i>	59
4.2.4. <i>Datos de la facilidad de modificación en la aplicación móvil</i>	64
4.3. Eficiencia de desempeño	66
4.3.1. <i>Obtención de datos de utilización de recursos</i>	67
4.3.2. <i>Datos de la utilización de recursos en el backend</i>	69
4.3.3. <i>Datos de la utilización de recursos en la aplicación móvil</i>	73
4.4. Análisis de resultados de la evaluación	77
4.4.1. <i>Facilidad de modificación</i>	77
4.4.2. <i>Eficiencia de desempeño</i>	78
CONCLUSIONES	81
RECOMENDACIONES	83
GLOSARIO	
BIBLIOGRAFÍA	
ANEXOS	

ÍNDICE DE TABLAS

Tabla 1-2:	Métricas para evaluar la eficiencia de desempeño.	23
Tabla 1-3:	Métodos, Técnicas y Fuente.....	28
Tabla 2-3:	Operacionalización conceptual de variables.	30
Tabla 3-3:	Operacionalización metodológica de variables.	31
Tabla 4-3:	Procesos principales del sistema.	32
Tabla 5-3:	Recursos de hardware existente.	36
Tabla 6-3:	Recursos de hardware requerido.	36
Tabla 7-3:	Recursos de software existente.	37
Tabla 8-3:	Recursos de software requeridos.....	37
Tabla 9-3:	Recursos humanos requeridos.....	37
Tabla 10-3:	Costos del proyecto.....	38
Tabla 11-3:	Miembros y roles involucrados en el proyecto.....	39
Tabla 12-3:	Tipos de usuarios y roles involucrados en el sistema.	40
Tabla 13-3:	Matriz de trazabilidad de requisitos.	41
Tabla 14-3:	Product Backlog.....	43
Tabla 15-3:	Estándares de codificación.....	48
Tabla 16-3:	Diccionario de datos de la tabla Users.	49
Tabla 17-3:	Sprint Backlog.	52
Tabla 18-3:	Historia de usuario 1.	53
Tabla 1-4:	Criterios de evaluación.....	55
Tabla 2-4:	Datos generales del código del backend.....	58
Tabla 3-4:	Datos generales del código de la aplicación móvil.....	59
Tabla 4-4:	Resultados de la facilidad de modificación del módulo Drivers.	59
Tabla 5-4:	Resultados de la facilidad de modificación del módulo Auth.	60
Tabla 6-4:	Resultados de la facilidad de modificación del módulo Orders.....	60
Tabla 7-4:	Resultados de la facilidad de modificación del módulo Notifications.	61
Tabla 8-4:	Resultados de la facilidad de modificación del módulo Users.	61
Tabla 9-4:	Resultados de la facilidad de modificación del módulo Files.....	62
Tabla 10-4:	Resultados de la facilidad de modificación del módulo Email.	62
Tabla 11-4:	Resultados de la facilidad de modificación del módulo Database.	63
Tabla 12-4:	Resultados de la facilidad de modificación en la aplicación móvil.	64
Tabla 13-4:	Tiempos promedios de los procesos registrar conductor y repartir pedidos.	67
Tabla 14-4:	Pruebas de la utilización de CPU del proceso registrar conductor.	69

Tabla 15-4:	Pruebas de la utilización de Memoria del proceso registrar conductor.....	70
Tabla 16-4:	Pruebas de la utilización de CPU del proceso repartir pedidos.	70
Tabla 17-4:	Pruebas de la utilización de la Memoria del proceso repartir pedidos.	71
Tabla 18-4:	Ponderaciones de la utilización de CPU y Memoria del backend.	71
Tabla 19-4:	Pruebas de la utilización del CPU del proceso registrar conductor.	73
Tabla 20-4:	Pruebas de la utilización de la Memoria del proceso registrar conductor.	73
Tabla 21-4:	Pruebas de la utilización de la CPU del proceso repartir pedidos.....	74
Tabla 22-4:	Pruebas de la utilización de la Memoria del proceso repartir pedidos.	74
Tabla 23-4:	Ponderaciones de la utilización de CPU y Memoria de la aplicación móvil.	75
Tabla 24-4:	Valoración para la facilidad de modificación.	77
Tabla 25-4:	Valoración de la utilización de la memoria.	78
Tabla 26-4:	Valoración de la utilización del CPU.	78
Tabla 27-4:	Ponderación de la utilización de recursos en el backend.....	79
Tabla 28-4:	Ponderación de la utilización de recursos en la aplicación móvil.....	79
Tabla 29-4:	Valoración de la eficiencia de desempeño.	80

ÍNDICE DE FIGURAS

Figura 1-2:	Proceso de Scrum.....	13
Figura 1-3:	Matriz de trazabilidad de requisitos implementado en Power Designer.....	42
Figura 2-3:	Diseño de la base de datos módulo orders.....	49
Figura 3-3:	Pantalla de inicio de sesión.....	50
Figura 4-3:	Pantalla del perfil del conductor.....	51
Figura 1-4:	Configuraciones iniciales del proyecto.....	57
Figura 2-4:	Ejecución del análisis estático del código.....	57
Figura 3-4:	Resultados del análisis estático.....	58
Figura 4-4:	Android Profiler.....	68
Figura 5-4:	Selección del dispositivo y el proceso en Android Profiler.....	69

ÍNDICE DE GRÁFICOS

Gráfico 1-2:	Ejemplo del estándar BPMN (Proceso de envío de un minorista de hardware).	9
Gráfico 2-2:	Características de una buena ERS.	12
Gráfico 3-2:	Llamadas asincrónicas de React Native.	19
Gráfico 4-2:	Característica de la calidad del producto software.	20
Gráfico 5-2:	Subcaracterísticas de la mantenibilidad.	21
Gráfico 1-3:	Proceso actual de reparto de paquetería.	34
Gráfico 2-3:	Diagrama de procesos con el sistema implementado.	35
Gráfico 3-3:	Diagrama de proceso de la entrega de orden a domicilio.	35
Gráfico 4-3:	Diagrama la arquitectura del sistema empleando el modelado C4.	45
Gráfico 5-3:	Organigrama de los módulos del sistema.	46
Gráfico 6-3:	Diagrama de despliegue del sistema.	47
Gráfico 1-4:	Comparación de los indicadores de los módulos del backend.	63
Gráfico 2-4:	Comparación de la facilidad de modificación de los módulos del backend.	64
Gráfico 3-4:	Indicadores de la facilidad de modificación en la aplicación móvil.	65
Gráfico 4-4:	Comparación de la aplicación móvil vs backend.	66
Gráfico 5-4:	Estadísticas de la utilización del CPU.	67
Gráfico 6-4:	Estadísticas de la utilización de la memoria.	68
Gráfico 7-4:	Estadísticas de utilización de CPU y Memoria.	69
Gráfico 8-4:	Utilización de la CPU de los procesos registrar conductor y repartir pedidos.	72
Gráfico 9-4:	Utilización de la Memoria de los procesos en el backend.	72
Gráfico 10-4:	Utilización del CPU en la aplicación móvil.	75
Gráfico 11-4:	Utilización de la Memoria en la aplicación móvil.	76
Gráfico 12-4:	Comparación de la utilización del CPU del backend y la aplicación móvil.	76
Gráfico 13-4:	Comparación de la utilización de memoria del backend y la aplicación móvil.	77

ÍNDICE DE ANEXOS

ANEXO A: ENTREVISTA

ANEXO B: BPMN CON EL SISTEMA IMPLEMENTADO

ANEXO C: MATRIZ DE TRAZABILIDAD DE REQUISITOS

ANEXO D: DISEÑO Y DICCIONARIO DE LA BASE DE DATOS

ANEXO E: DISEÑO DE INTERFACES DE USUARIO

ANEXO F: HISTORIAS DE USUARIOS

RESUMEN

En el presente trabajo se desarrolló una aplicación móvil mediante el uso de React Native con Expo y backend con NestJS para apoyar la gestión de reparto de paquetería de los conductores en la empresa "Corporación Wolf S.A.". Para el desarrollo del producto software se realizó un análisis del proceso actual de gestión de paquetería liviana de la empresa, donde se identificaron diversos problemas que afectan a la calidad del servicio, por lo cual se buscó solucionar estos inconvenientes en un nuevo proceso que tiene como eje la aplicación móvil. Para el desarrollo de la aplicación móvil se empleó la metodología SCRUM. En la fase de planificación se definieron los roles y miembros del proyecto, también se realizó la especificación de requisitos con la ayuda de una matriz de trazabilidad y se creó el product backlog. En la fase de diseño se definieron los estándares de codificación, diseño de la base de datos, arquitectura del sistema, diagrama de despliegue y diseño de interfaces de usuario de la aplicación móvil. En la fase de desarrollo se realizaron 6 sprints con una duración de 240 horas en total. Por otro lado, se aplicó la ISO/IEC 25010 para evaluar la facilidad de modificación y eficiencia de desempeño de la aplicación móvil y backend. Usando la herramienta SonarQube se midió la facilidad de modificación, obteniendo un 99.61% en el backend y 99.21% en la aplicación móvil. Para la eficiencia de desempeño se utilizó la herramienta Android Profiler y las estadísticas del servidor, obteniendo un 75% en la aplicación móvil y 87.5% en el backend. Se concluye que el producto software es fácil de modificar y tiene una eficiencia de desempeño muy buena tanto en la aplicación móvil como en el backend. Se recomienda implementar los sistemas complementarios para un mejor uso de la aplicación.

Palabras clave: <METODOLOGÍA ÁGIL SCRUM> <APLICACIONES MULTIPLATAFORMA> <REACT NATIVE> <FRAMEWORK NESTJS> <FACILIDAD DE MODIFICACIÓN> <EFICIENCIA DE DESEMPEÑO>

1397-DBRA-UTP-2022



SUMMARY

We developed a mobile application in this work utilizing React Native with Expo and backend with NestJS to support the packaging delivery management of drivers in the company "Corporación Wolf SA." We carried out an analysis of the current process of light packaging management of the company for the development of the software product, where several problems that affect the quality of service were identified, so we tried to solve these problems in a new process that has as its axis the mobile application. The SCRUM methodology was used for the development of the mobile application. In the planning phase, we defined the roles and members of the project, the requirements specification was also made with the help of a traceability matrix, and we created the product backlog. In the design phase, we defined coding standards, database design, system architecture, deployment diagram and user interface design of the mobile application. In the development phase, 6 sprints were carried out with a total duration of 240 hours. On the other hand, we applied the ISO/IEC 25010 to evaluate the ease of modification and performance efficiency of the mobile and backend application. We measured the ease of modification utilizing the SonarQube tool, obtaining 99.61% in the backend and 99.21% in the mobile application. For performance efficiency, the Android Profiler tool and server statistics were utilized, obtaining 75% in the mobile application and 87.5% in the backend. We concluded that the software product is easy to modify and has a very good performance efficiency both in the mobile application and in the backend. It is recommended to implement the complementary systems for a better use of the application.

Key words: <MOBILE APPLICATION> <AGILE METHODOLOGY SCRUM>
<REACT NATIVE> <NESTJS FRAMEWORK> <EASE OF MODIFICATION>.

A handwritten signature in blue ink, appearing to be 'Antonio...', written over a light blue grid background.

INTRODUCCIÓN

Actualmente, las Tecnologías de la Información y Comunicación se encuentran presentes en la vida cotidiana de las personas, ayudando en la realización de sus tareas con herramientas o servicios básicos al alcance de sus manos por medio de las aplicaciones para dispositivos móviles. Las aplicaciones móviles son consideradas por las empresas como una gran oportunidad para crear nuevas fuentes de ingreso migrando muchos de los negocios o industrias tradicionales a servicios escalables y de fácil acceso para el público general.

La empresa “Corporación Wolf S.A.”, ubicada en la ciudad de Quito de la provincia de Pichincha, ha identificado ciertos inconvenientes en el proceso de transporte de paquetería liviana, en el cual se tiene un punto central de recepción de paquetes y a partir de ahí se coordina la logística para el transporte de estos, siendo lo común un horario fijo. Es por esta razón que se busca crear un servicio que abstraiga toda la logística del proceso de reparto de paquetería liviana en un sistema que permita tener tiempos de entrega más cortos con una calidad de servicio óptima. Este sistema contará con 3 módulos principales: módulo lockers, aplicación móvil del cliente y aplicación móvil del conductor. Sin embargo, el objetivo del presente trabajo de integración curricular es desarrollar la aplicación móvil del conductor y backend, de tal forma que permita tener a disposición la información necesaria para realizar el transporte de paquetería liviana. De esta manera, el sistema guiará a los conductores durante este mismo proceso.

Este documento está estructurado en cuatro capítulos, donde en el primer capítulo se define la problemática, en el segundo se realiza una documentación de las bases teóricas más relevantes, en el tercero se definen la metodología del trabajo y el desarrollo de la aplicación móvil y backend con la metodología Scrum, finalmente en el cuarto capítulo se dan los resultados del trabajo de integración curricular y se da respuesta a las preguntas de investigación.

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA

En la presente sección se analizan los antecedentes del trabajo de integración, la formulación del problema, justificación del trabajo y los objetivos planteados.

1.1. Antecedentes

Corporación Wolf S.A. es una empresa ecuatoriana de ingeniería electrónica y tecnologías de la información, que tiene 3 líneas de negocio: Robot Wolf que es un programa educativo para instituciones, Wolf Electronics el cual es un e-commerce de componentes electrónicos, sistemas de vigilancia, entre otros y su última línea llamada Corporación Wolf S.A. en la cual se imparten capacitaciones en ingeniería electrónica, mecánica y tics.

La Corporación Wolf está creando un nuevo servicio a través del cual busca mejorar el proceso de transporte de paquetería actual, para esto combinan la tecnología de casilleros inteligentes y un sistema de reparto (delivery) que estará compuesto de dos aplicaciones móviles, una para los clientes del sistema y otra para los conductores quienes realizarán el transporte de los paquetes, siendo esta última aplicación móvil el objeto de estudio que se desarrolla en este trabajo, también se incluye la construcción del backend para la aplicación móvil del conductor. Es importante definir los dos conceptos principales con los que está constituido el sistema: casilleros inteligentes y sistema de reparto. En (Gonzales, Urbiola y Vitale 2018, p. 20) definen los casilleros inteligentes como: "Un punto de entrega fijo, automatizado, de fácil acceso al usuario y con tecnología amigable". Por otro lado, la Real Académica Española (RAE) define un sistema de reparto (delivery en inglés) al servicio que ofrece un comercio para entregar sus productos en el domicilio del comprador.

Por otra parte, un desarrollo sin seguir convenciones de codificación deriva a tener código poco legible y en muchos casos con un nivel de complejidad alto y con bajo desempeño. Estas causas provocan que en el producto software final se tenga problemas al momento de dar mantenimiento por su elevada dificultad de modificación, así mismo aumenta el tiempo empleado para depurar errores y condiciona la capacidad de reparación.

Existen diversos trabajos relacionados con esta investigación como: (Eskola 2018) en su trabajo de "React Native Performance Evaluation" da luces sobre qué elementos de React son mejores para implementar en una aplicación móvil, otros autores como (Gonzales, Urbiola y Vitale 2018) cuyo trabajo de "Red de lockers para entrega y recolección de paquetes" integra a vendedores y compradores de e-commerce mediante plataforma en línea, que trata de gestión e implementación de casilleros inteligentes. Por último (Farfán y Vega 2020) en su artículo de "Software maintainability. Considerations for its specification and validation" dan a conocer las métricas que se emplean en la mantenibilidad del software, en contraste, el presente trabajo se enfoca en el desarrollo de una aplicación móvil usando tecnologías como React Native con Expo.

En base a lo mencionado es importante resaltar que la aplicación del conductor está dirigida a personas que cuenten con un automóvil y una licencia de conducir habilitada, que estén registrados y aprobados en la empresa. Esta aplicación está disponible para los dispositivos móviles con sistema operativo ANDROID e IOS.

1.2. Formulación del problema

¿Cómo se puede apoyar la gestión de reparto de paquetería de los conductores a través una aplicación móvil empleando React Native con Expo y backend con NestJS en la empresa “Corporación Wolf S.A.”?

1.3. Sistematización del problema

- ¿Cuál es el proceso actual para reparto de paquetería liviana de la empresa “Corporación Wolf S.A.”?
- ¿Cuáles son los requisitos necesarios para la aplicación móvil?
- ¿Cómo se van a desarrollar las funcionalidades de la aplicación móvil y backend empleando la metodología SCRUM?
- ¿Cuál es la eficiencia de desempeño y la facilidad de modificación de la aplicación móvil y backend?

1.4. Justificación del trabajo de titulación

1.4.1. Justificación teórica

React Native es un framework de aplicación móvil open source creado por Facebook, Inc. Se utiliza para desarrollar aplicaciones nativas híbridas para Android e IOS (Tom Occhino 2015, pArr. 21). Este framework se basa en los mismos conceptos que React, pero en lugar de renderizar elementos HTML, React Native utiliza los componentes fundamentales de la interfaz de usuario de la plataforma nativa (Eskola 2018, p. 17), es decir, que los desarrolladores pueden crear componentes nativos escribiendo en un lenguaje de nivel superior (javascript), mientras utiliza conceptos e ideas propias de React y se mantiene una sola base de código. Hay dos subprocesos que realizan el trabajo principal: el subproceso de los módulos nativos (principal) y el subproceso de JavaScript (Meta Platforms Inc 2021, pArr. 1).

Además, de utilizar React Native se optó por implementar Expo, el cual es un conjunto de herramientas y librerías que facilita la programación en React Native. Su finalidad es crear un puente entre el ordenador y el dispositivo móvil que se esté utilizando para ejecutar la aplicación. Expo proporciona un conjunto de ventajas respecto a la instalación manual del entorno de React Native, como, por ejemplo, no hace falta tener Android Studio y/o XCode, ya que se puede ejecutar el proyecto a partir de un enlace especial que proporciona el cliente Expo o a partir de un código QR. Otra ventaja es que Expo instala los componentes necesarios para realizar un proyecto React Native sin que el usuario se tenga que preocupar de instalar manualmente el conjunto de librerías necesarias para el correcto funcionamiento (como node, openjdk, etc) (Díaz 2021, p. 4). Como contrapartida, limita el Kit de Desarrollo de Software (SDK) haciendo que algunos módulos nativos o librerías no sean compatibles, no obstante, dado los requisitos de la aplicación móvil del conductor Expo es más que suficiente para el desarrollo.

Por otra parte (Farfán y Vega 2020, p. 663) mencionan las ventajas de un software mantenible:

- Aporta directamente a las medidas de analizabilidad, comprensibilidad del producto de software y la modificación del mismo.
- Permite determinar de acuerdo a la técnica de programación utilizada, por ejemplo, la programación orientada a objeto, qué tan mantenible es el sistema, a través del análisis de las estadísticas de acoplamiento entre objetos, herencia, profundidad en el árbol de herencia y métodos heredados por mencionar algunos.
- Permite denotar elementos de un software como el tiempo de vida de un sistema legado, cantidad de módulos (tamaño del sistema), estructura del código fuente que intervienen directamente con la mantenibilidad.

Es importante resaltar que un software mantenible influye positivamente en otros aspectos como la legibilidad y depuración del código, también ayuda durante la implementación de nuevas funcionalidades permitiendo reducir el tiempo y esfuerzo de un programador, además de que el software creado es mucho más escalable.

1.4.2. Justificación aplicativa

La aplicación móvil del conductor se centra en el reparto y gestión de paquetes, permitiendo al conductor tener mayor control y detalle de los paquetes a transportar, además se implementa el backend para que la aplicación móvil se pueda comunicar con los otros módulos del sistema del cual forma parte.

La integración de la aplicación móvil del conductor, aplicación móvil del cliente, backend y módulo de casilleros permite la creación del sistema "Wolf Delivery" que proporciona un seguimiento y control más detallado de los paquetes desde su entrega hasta la recepción del paquete al domicilio del cliente mediante el seguimiento en vivo del paquete, notificaciones de llegadas o alertas de cercanía, permitiendo reducir el tiempo entre el envío y la recepción debido a que los paquetes van a ser llevados por personas que tienen la misma ruta del paquete y por ende estarán con un mayor nivel de disponibilidad.

La aplicación del cliente permite a los usuarios registrar los paquetes que van a enviar, además proporciona ciertos beneficios o promociones por el uso de la misma, asimismo permite realizar el pago en efectivo o con tarjetas de crédito/débito. Por otro lado, la aplicación del conductor permite tener un mayor control sobre los paquetes que va a transportar, alineando los paquetes disponibles de acuerdo a su ruta y proporcionando información detallada sobre la paquetería que se encuentran en los casilleros en tiempo real, también se visualiza la ruta que debe seguir para el transporte de los paquetes a su destino o la ruta que debe seguir para llegar a un punto de recolección de paquetes. Otra funcionalidad con la que cuenta son las notificaciones por diferentes eventos, como por ejemplo para advertir al conductor que uno o más paquetes que se encuentran en un punto de recolección de paquetes cercano a su ubicación tienen el mismo destino de la ruta que se encuentra cursando, además, el conductor puede referir o recomendar a otro conductor para adquirir un bonus especial de acuerdo a las políticas de la empresa.

El desarrollo de software empleando estándares de mantenibilidad ayuda a disminuir considerablemente la complejidad al momento de introducir una nueva funcionalidad, reduciendo el tiempo de implementación, también mejora la capacidad de reparación de la aplicación móvil

y del backend ante posibles errores o cambios. Otro punto importante es que reduce la dificultad de depuración del código. Por otro lado, tener un software eficiente ayuda a reducir el consumo de recursos y permite un mejor rendimiento de la aplicación en el dispositivo. Es importante destacar que los estándares de mantenibilidad se aplican tanto en el desarrollo de la aplicación móvil del conductor como en su respectivo backend.

Finalmente, es importante mencionar que el presente trabajo está bajo la línea de investigación de Tecnología de la Información y Comunicación (Tic's) en el programa de ingeniería de software cuya línea de investigación es desarrollo de aplicaciones de software, hardware y telecomunicaciones. En el plan nacional de desarrollo “Creación de Oportunidades” se encuentra en el primer eje (económico), aplicando el objetivo 1 que trata de incrementar y fomentar, de manera inclusiva, las oportunidades de empleo y las condiciones laborales.

1.5. Objetivos

1.5.1. Objetivo general

- Desarrollar una aplicación móvil mediante el uso de React Native con Expo y backend con NestJS para apoyar la gestión de reparto de paquetería de los conductores en la empresa "Corporación Wolf S.A."

1.5.2. Objetivos específicos

- Analizar el proceso actual de reparto de paquetería liviana de la empresa “Corporación Wolf S.A.”
- Definir los requisitos necesarios de la aplicación móvil.
- Desarrollar las funcionalidades de la aplicación móvil y backend utilizando la metodología SCRUM.
- Evaluar la eficiencia de desempeño y facilidad de modificación de la aplicación móvil y backend.

CAPÍTULO II

2. FUNDAMENTOS TEÓRICOS

En el presente capítulo se indica las bases teóricas necesarias para el desarrollo de una aplicación móvil y backend para apoyar a los conductores en la gestión de reparto de paquetería en la empresa "Corporación Wolf S.A."

2.1. Casilleros inteligentes

Con base en (Ricoh Europe PLC 2019, p. 6) las unidades de los casilleros inteligentes son dispositivos ubicadas en lugares públicos como en zonas urbanas, que almacenan paquetes para todos los clientes en un área y tiempos específicos, además, estos dispositivos son puntos de entrega automatizados. Por otro lado, (Exela Technologies Inc. 2019, pArr. 1) define a los sistemas de casilleros inteligentes como “una solución de almacenamiento y entrega automatizada simple, segura, conveniente y fácil de usar que simplifica la administración de paquetes y almacenamiento”. Los sistemas de casilleros pueden funcionar como módulos independientes conectados en red y están disponibles en diferentes tamaños para configuraciones personalizables.

2.1.1. Tipos de casilleros inteligentes

Según (Ricoh Europe PLC 2019, p. 4) existen los siguientes tipos de casilleros inteligentes:

- **Casilleros para inventario:** son unidades o dispositivos de que pueden distribuir el inventario para las empresas donde se lo necesite, permitiendo reducir costos de almacenamientos de sus empleados.
- **Casilleros inteligentes para almacenamiento personal:** son casillero que ofrecen almacenamiento temporal o permanente, brindando un espacio seguro para sus pertenencias.
- **Casilleros inteligentes para gestión de paquetería:** son unidades que permite una mejor gestión de la oficina de correspondencia. Los casilleros seguros y monitoreados permiten el seguimiento del almacenamiento de correspondencia hasta que el destinatario esté disponible.

2.1.2. Paquete

Según la (Real Academia Española 2022, pArr. 1), define a un paquete como un “lío o envoltorio bien dispuesto y no muy abultado de cosas de una misma o distinta clase”, también se puede definir como un embalaje que tiene un nombre, ciudad y dirección con la finalidad de ser entregado a su destinatario, a través de los servicios de entrega exprés (transporte por carretera). Hoy en día los paquetes suelen ser más detallados teniendo un código de barra que permite seguir el recorrido del mismo.

2.1.3. Gestión de Paquetería

La gestión de paquetería es el proceso que empieza desde que el cliente lleva el paquete al casillero inteligente y llena los datos necesarios para subir al sistema hasta que el conductor obtiene los datos del paquete accediendo al sistema y acepta llevarlo hasta su destino final.

2.2. Estándar BPMN

La (Object Management Group 2021, pArr. 1) define al Modelo y Notación de Procesos de Negocio (BPMN por sus siglas en inglés) como un estándar que proporciona a las empresas la capacidad de comprender sus procedimientos comerciales internos en una notación gráfica, así mismo, permite la comunicación de estos procedimientos de manera estándar. Además, la notación gráfica facilita la comprensión de las colaboraciones de desempeño y las transacciones comerciales entre las organizaciones, lo cual garantiza que las empresas se comprendan a sí mismas y a todos los participantes de su negocio. Esto permite a las organizaciones adaptarse rápidamente a las nuevas circunstancias comerciales internas y B2B. En el Gráfico 1-2, se visualiza el proceso de envío de un minorista de hardware a través del estándar BPMN.

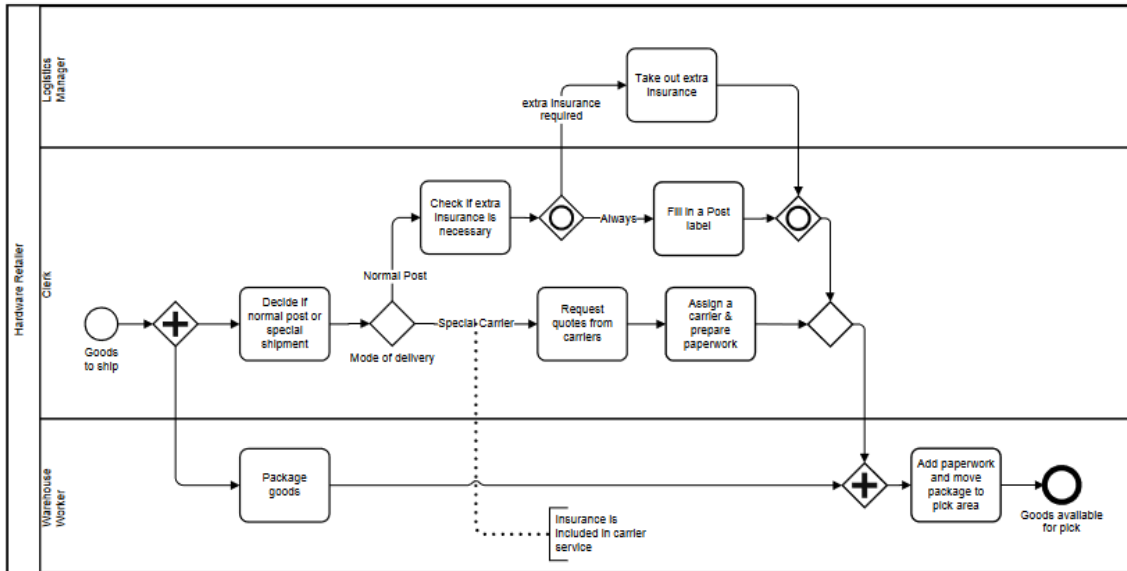


Gráfico 1-2. Ejemplo del estándar BPMN (Proceso de envío de un minorista de hardware).

Fuente: (Object Management Group 2014)

2.3. Aplicación móvil

Las aplicaciones móviles están teniendo un gran auge en los últimos años, propiciando una nueva generación de herramientas y servicios disponibles desde nuestros bolsillos. (Enriquez y Casas 2013, p. 26) definen aplicación móvil, como aquel software desarrollado para dispositivos móviles (refiriéndose a móvil a poder acceder desde cualquier lugar y momento a la información, las aplicaciones y los dispositivos). Este tipo de aplicaciones se desarrollan considerando las limitaciones de los propios dispositivos, como por ejemplo el ancho de banda limitado, bajo poder de cómputo, la poca capacidad de almacenamiento, etc. Los dispositivos móviles son suficientemente pequeños y livianos como para ser transportados con facilidad por personas, además disponen de la capacidad de batería adecuada para funcionar de forma autónoma. Cada uno de ellos tiene sus propias particularidades en lo que se refiere a la utilización por parte del usuario, así como también al momento de desarrollar una aplicación.

(Enriquez y Casas 2013, p. 35) en su trabajo definen dos tipos de aplicaciones móviles: Aplicación Nativa y Aplicación Web. Además, es importante resaltar que en la actualidad ha surgido otro tipo de aplicación como es la móvil híbrida. A continuación, se detallan estos tipos:

- **Aplicaciones nativas:** son aplicaciones que están desarrolladas específicamente para un sistema operativo móvil o para un tipo de dispositivo, tomando en cuenta las limitaciones de hardware que tiene. Este tipo de aplicaciones son rápidas y tienen la ventaja de acceder a las funciones del dispositivo de forma directa, como, por ejemplo: GPS (sistema de posicionamiento global), almacenamiento, cámaras, entre otros. El principal obstáculo de

estas aplicaciones es que se debe desarrollar una aplicación para cada plataforma incrementando el tiempo y costo de desarrollo.

- **Aplicaciones web:** son páginas web optimizadas para ser visualizadas en dispositivos móviles, estas se suelen desarrollar con HTML, JavaScript y CSS. Estas aplicaciones son accedidas a través de un navegador web, además son fáciles de desarrollar e implementar sin mencionar que necesitan menos requerimientos de hardware para funcionar. Sin embargo, tienen un gran problema, el cual es que no pueden acceder a funcionalidades propias del dispositivo como la cámara.
- **Aplicaciones móviles híbridas:** son aplicaciones capaces de ejecutarse en distintos sistemas operativos móviles teniendo una única base de código de fuente, además este tipo de aplicaciones permite acceder a los recursos de los dispositivos como cámara, GPS, almacenamiento, entre otros. Algo a tener en cuenta, es que, estas aplicaciones por lo general suelen tener un rendimiento un inferior a una aplicación nativa.

2.4. Backend

El término backend hace referencia a la sección de código fuente que no se puede visualizar desde el lado del cliente (aplicación) y se ejecuta en un servidor de aplicaciones. Este lado es el responsable de la lógica del negocio, la conexión a la base de datos, validación y autorización de los usuarios, creación de servicios web, seguridad y optimización entre otras más. Dicho de otra forma, la función principal del backend es la de acceder a la información que se solicita a través de la aplicación del cliente, para posteriormente combinarla y devolverla al cliente, este proceso se lo realiza por lo general haciendo uso de servicios web.

2.4.1. Servicios web

Según la (W3C 2021, pArr. 116) un servicio web es un sistema de software diseñado para admitir la interacción interoperable de máquina a máquina a través de una red. Básicamente, los servicios web son elementos que permiten a las aplicaciones comunicarse con otras aplicaciones, sin tener en cuenta el lenguaje o plataforma en que se desarrollen. Existen diferentes tipos de estándares para servicios web como SOAP, REST, GRAPHQL, entre otros, pero en este trabajo se utilizará los servicios REST.

2.4.1.1. Servicio web REST

De acuerdo con (Hidalgo y Jiménez 2016, p. 19) el modelo de Transferencia de Estado Representacional o simplemente llamado REST (servicios web rest) están teniendo un avance dentro del desarrollo de aplicaciones web o sistemas distribuidos de hipermedia, debido a que es un servicio orientado a recursos y ayuda a mejorar y facilitar el trabajo en la web. El estilo arquitectónico consiste en clientes y servidores, donde se exponen funcionalidades mediante recursos identificados por URI; los clientes interactúan con los recursos mediante métodos generando solicitudes a los servidores, que las procesan y generan una respuesta conveniente. De manera que la construcción de solicitudes y respuestas se realizan alrededor de la transferencia de representaciones de los recursos, siendo uno de los más importantes la representación del recurso capturado en estado actual.

Según (Sayago, Flores y Recalde 2019, p. 13) los servicios web REST tienen las siguientes características:

- Se orientan recursos descritos por un id único.
- Transfiere datos en formatos XML y JSON.
- Al no poseer estados optimiza el almacenamiento en memoria cache son asertivos en ambientes con un bajo ancho de banda.
- Para la identificación de cada recurso se utiliza un URL en donde los clientes lo invocan directamente.
- Permite reutilización de mecanismos de autorización, cifrado y autenticación por HTTP.

2.5. Requisitos de Software

Conforme a (Sommerville 2011, p. 83), un requisito “es solo una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste. En el otro extremo, es una definición específica y formal de una funcionalidad del sistema”. Los requisitos de software también se consideran como un documento estructurado que especifican las funcionalidades y limitaciones del sistema. Un claro ejemplo de este documento es la especificación de requisitos de software (ERS) que ayuda a registrar, gestionar los requisitos de software.

2.5.1. Especificación de Requisitos Software

Acorde con (Monferrer 2000, p. 1) el análisis de requisitos es el proceso de estudio de las necesidades de los usuarios para llegar a una definición de los requisitos del sistema. Una especificación de requisitos de software (ERS), es un documento escrito donde se define correctamente todos los requerimientos. Esta documentación no debe incluir ningún diseño detallado de la implementación o la gestión, ya que los requisitos se deben describir de tal manera que para el usuario sea fácil de comprenderlo. De igual manera, se da una mayor margen de flexibilidad a los desarrolladores para el desarrollo.

En el Gráfico 2-2, se muestran las características de una buena especificación de requisitos de software de acuerdo con (Monferrer 2000, p. 3).

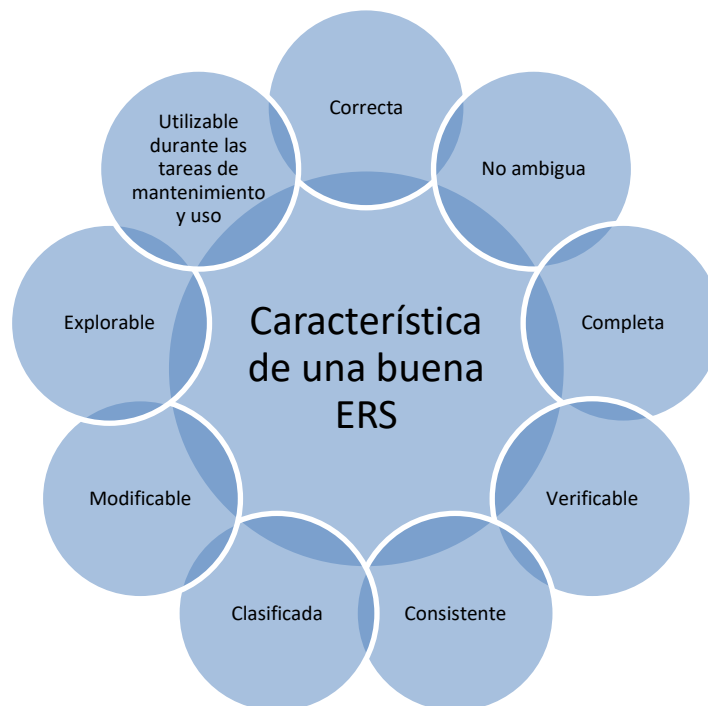


Gráfico 2-2. Características de una buena ERS.

Realizado por: Bone Edward y Ochoa José, 2022

2.6. Metodología de desarrollo de software

Para (Maida y Pacienza 2015, p. 13) una metodología de desarrollo de software o metodología de desarrollo de sistemas en ingeniería de software es un marco de trabajo que se usa para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información. Muchos de estos marcos de trabajo han evolucionado a lo largo del tiempo, cada uno con sus ventajas y

desventajas. No existe una metodología de desarrollo que se adecue para usarla en todos los proyectos existentes, en su lugar, cada una de las metodologías disponibles se puede adaptar de mejor o peor forma a los proyectos, basándose en consideraciones técnicas, organizacionales, de proyecto y de equipo.

2.6.1. Metodología SCRUM

De acuerdo a (Schwaber y Sutherland 2020, p. 3) Scrum es un marco de trabajo ligero que ayuda a las personas, los equipos y las organizaciones a generar valor a través de soluciones adaptables para problemas complejos al mismo tiempo que entregan productos de manera productiva y creativa. Una de las características principales de Scrum es que su aplicación se realiza mediante iteraciones (Sprints), cada uno de estas con una duración promedio de 30 días, otra característica importante son los diferentes tipos de reuniones que se tienen a lo largo del proyecto destacando la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración.

2.6.2. Proceso de aplicación de Scrum

Según (Maida y Pacienza 2015, p. 79) en Scrum un proyecto se ejecuta en bloques temporales cortos y fijos (iteraciones de 2 a 4 semanas como máximo). Cada una de las iteraciones tienen que generar un incremento del producto final que pueda ser entregado con el mínimo esfuerzo al cliente. En la Figura 1-2, se visualiza el proceso de Scrum.

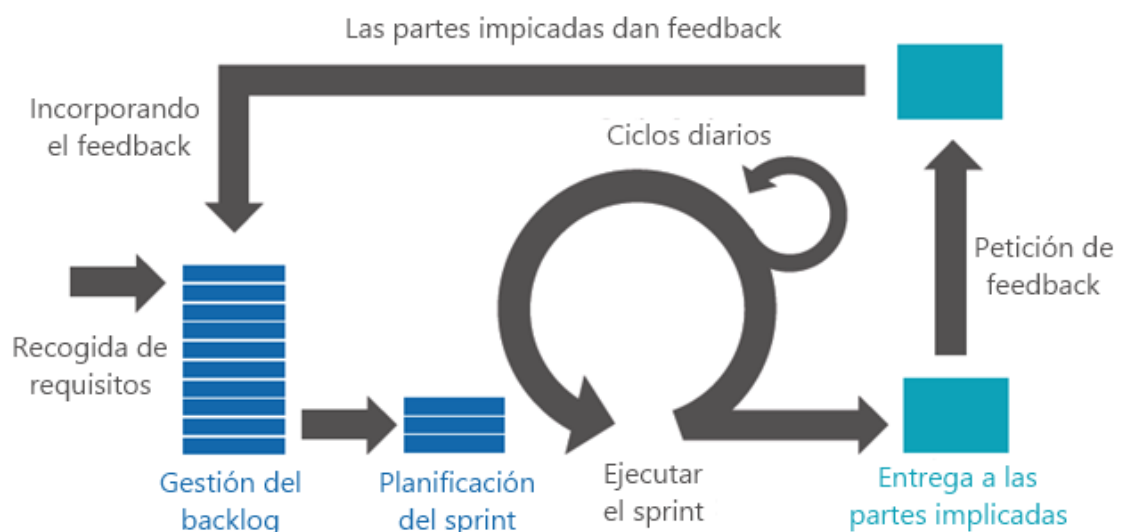


Figura 1-2. Proceso de Scrum.

Fuente: Maida y Pacienza, 2015

Las actividades que se realizan en Scrum son las siguientes:

2.6.2.1. *Sprint Planning*

En la fase de Sprint Planning se establece el trabajo que se realizará durante todo el Sprint. Este plan resultante es creado por el trabajo colaborativo de todo el Equipo Scrum. En el trabajo de (Maida y Pacienza 2015, p. 80), mencionan que el Sprint Planning se divide en dos partes, en la primera se presenta/escoge una lista de requisitos del product backlog del proyecto, donde se define una meta de la iteración y resuelven dudas específicas de estos requisitos; en la segunda parte se definen las tareas necesarias para poder completar los objetivos del sprint, se realiza una estimación del esfuerzo y se asignan las actividades que debe realizar cada miembro del equipo de desarrollo.

2.6.2.2. *Sprint*

Para (Schwaber y Sutherland 2020, p. 7) los Sprints son el latido del corazón de Scrum, donde las ideas se convierten en valor. Básicamente es la ejecución de las tareas que fueron seleccionadas en la fase de planificación (Sprint Planning), por lo general este Sprint tiene una duración de un mes o menos (2 a 4 semanas). Al final de cada iteración se tiene que proporcionar un resultado completo, un incremento del producto que sea potencialmente entregable, de tal forma que cuando el cliente (product owner) lo solicite sólo sea necesario un esfuerzo mínimo para que el producto pueda ser utilizado.

2.6.2.3. *Daily Scrum*

De acuerdo con (Schwaber y Sutherland 2020, p. 9), el Daily Scrum tiene como objetivo inspeccionar el progreso realizado hacia la meta del Sprint y adaptar el sprint backlog según sea necesario, ajustando el próximo trabajo planificado, además, aquí se facilita la transferencia de información y la colaboración entre los miembros del equipo para aumentar su productividad, al poder coordinar los puntos en los que se pueden ayudar unos a otros.

2.6.2.4. *Sprint Review*

Según (Maida y Pacienza 2015, p. 82), el Sprint Review tiene como objetivo inspeccionar el resultado del Sprint y determinar las futuras adaptaciones que se tendrán. Básicamente es una reunión donde el equipo de desarrollo presenta al cliente los requisitos completados en la iteración en forma de un incremento del producto, haciendo un recorrido por ellos lo más real y cercano posible al objetivo que se pretende cubrir.

2.6.2.5. *Sprint Retrospective*

Conforme a los trabajos de (Schwaber y Sutherland 2020, p. 10; Maida y Pacienza 2015, p. 83), en el **Sprint Retrospective** se buscan formas de aumentar la calidad y eficiencia del trabajo para los futuros Sprints a realizarse. El equipo analiza cómo ha sido su rendimiento durante la última iteración, por qué se están cumpliendo o no los objetivos establecidos al inicio de la iteración y por qué el incremento de producto que acaba de demostrar al cliente era lo que él esperaba o no.

2.6.3. *Roles de Scrum*

Los roles de Scrum funcionan como una unidad cohesionada de profesionales con una misma meta en común, el producto. El equipo de Scrum es responsable de todas las actividades relacionadas con el producto, desde la colaboración de las partes interesadas, operación, verificación, mantenimiento, experimentación y desarrollo; también tienen la responsabilidad de crear incrementos valiosos y útiles al final de cada Sprint. Para (Schwaber y Sutherland 2020, p. 5), Scrum define tres roles específicos:

2.6.3.1. *Equipo de desarrollo*

El equipo de desarrollo consiste en el grupo de personas encargadas de desarrollar el producto del proyecto, las responsabilidades que recaen sobre este grupo son las siguientes:

- Crear un plan para el Sprint, el Sprint Backlog.
- Incluir calidad en el producto dependiendo del estándar utilizado.
- Adaptar su plan cada día a la meta de Sprint.

2.6.3.2. *Product Owner*

El product owner, es la persona responsable de maximizar el valor del producto resultante del equipo de desarrollo, por lo cual es también responsable de la gestión eficaz del producto que incluye:

- Desarrollar y comunicar explícitamente el objetivo del producto.
- Crear y comunicar claramente los elementos del product backlog.
- Ordenar los de elementos del product backlog.
- Asegurarse que el product backlog sea transparente, visible y entendido.

El product owner es una persona, no un comité y puede representar las necesidades de muchas partes interesadas en el product backlog.

2.6.3.3. *Scrum Master*

El scrum master es el líder responsable de establecer la metodología Scrum, tanto dentro del equipo de desarrollo como de la organización. El scrum máster tiene muchas responsabilidades tanto para el product owner, el equipo de desarrollo y la organización, algunas de ellas son:

- Liderar, capacitar y asesorar a la organización en su adopción de Scrum.
- Facilitar la colaboración de las partes interesadas según se solicite o necesite.
- Eliminar las barreras entre las partes interesadas y los equipos Scrum.

2.7. Herramientas para desarrollo de software

Para el desarrollo de la aplicación se emplean diferentes herramientas que ayudan en todo el ciclo de construcción del software.

2.7.1. *Herramientas para el flujo de trabajo*

Son las herramientas que mejoran la eficiencia del flujo de trabajo durante todo el ciclo de desarrollo de software.

2.7.1.1. *Slack*

En la página oficial de (Slack 2021, pArr. 1), define a Slack como una herramienta de mensajería instantánea que tiene como fin comunicar a las personas de varios equipos de trabajo con la información. Slack innova la forma en que se comunican las empresas, reuniendo a las personas para que trabajen como un solo equipo. Esta herramienta busca que los equipos trabajen de manera más sincronizada, flexible e inclusiva para de esta forma conseguir los objetivos planteados y tener un ambiente de trabajo más amigable. Las principales características que tiene Slack son las siguientes:

- **Conectado:** Permite comunicar a cualquier persona dentro o fuera de la empresa, también tiene canales dedicados donde se reúnen a las personas e información adecuada.
- **Flexible:** Admite el trabajo asincrónico, debido a que toda la información está en un solo lugar y es accesible desde cualquier lugar y hora del mundo.

- **Inclusivo:** Todos los miembros de una empresa pueden acceder a la misma información compartida.

2.7.1.2. *GitHub*

De acuerdo con (Hernández 2018, p. 10), la plataforma del “octocat”, también conocido como Github, se trata de un sitio web con servicio en la nube que brinda a los desarrolladores un espacio donde alojar su código y tener un control sobre cualquier cambio sobre el mismo debido a su sistema de control de versiones de Git. Esta plataforma, además de ofrecer almacenamiento también incluye otras herramientas que permiten desarrollar un proyecto software con total integración, feedback, cooperación, organización, etc.

2.7.2. *Tecnología para el backend*

Son las tecnologías que se emplean del lado del servidor con el objetivo de interactuar con la base de datos, gestionar peticiones y de controlar la persistencia de los datos.

2.7.2.1. *PostgreSQL*

Según la página oficial (PostgreSQL 2021, pArr. 1), PostgreSQL es un gran sistema de base de datos relacional de objetos de código abierto que utiliza y amplía el lenguaje SQL combinado con muchas características que almacenan y escalan de forma segura las cargas de trabajo de datos más difíciles. Los inicios de PostgreSQL se remontan a 1986 como parte de un trabajo POSTGRES en la Universidad de California en Berkeley y tiene más de tres décadas de desarrollo funcional en la plataforma central. PostgreSQL ha obtenido una sólida reputación por su arquitectura probada, confiabilidad, integridad de datos, extensibilidad y el esmero de la comunidad de código abierto detrás del software para brindar soluciones revolucionarias y eficientes.

2.7.2.2. *TypeScript*

Según la página oficial de (TypeScript 2021, pArr. 1), TypeScript es un lenguaje de programación libre y de código abierto construido sobre JavaScript, en el cual se añaden tipos estáticos. Los tipos proveen una manera de describir la forma de un objeto, suministrando mejor documentación y permitiendo a TypeScript validar que el código funciona correctamente. Denotar tipos es opcional, porque la inferencia de tipos permite obtener mucha más potencia sin escribir código

adicional, algo importante es que todo código en JavaScript también es válido en TypeScript. Además, el código es transformado a JavaScript a través del compilador de TypeScript o Babel, siendo este código JavaScript un código limpio y simple que se puede ejecutar en cualquier lugar donde se ejecute JavaScript. Este lenguaje de programación es uno de los más utilizados a nivel mundial, sin mencionar que es mantenido por Microsoft.

2.7.2.3. *NestJS*

NestJS es un marco progresivo de Node.js enfocado para el lado del servidor (backend) desarrollado completamente en TypeScript. Este framework permite la creación de aplicaciones eficientes, confiables y escalables, proporcionando un flujo de trabajo predeterminado con buenas prácticas de desarrollo, además permite combinar elementos de la programación orientada a objetos, programación funcional y programación reactiva. Vale la pena mencionar que NestJS está inspirado en Angular, por lo que comparte muchos conceptos. Algo importante es que por debajo del framework se hace uso de librerías existentes y maduras que muchos desarrolladores de Node han utilizado por mucho tiempo como Express, TypeORM, entre otras; realizando una abstracción de estas librerías para un uso más práctico y robusto de inicio a fin.

2.7.3. *Tecnología para el frontend*

Son aquellas tecnologías que se usan del lado del cliente para poder crear la parte visible de la aplicación como las interfaces de usuario que conforman la aplicación.

2.7.3.1. *React Native*

React Native es un framework para la creación de aplicaciones híbridas nativas para Android e IOS el cual fue desarrollado por Facebook, Inc. como una forma de crear aplicaciones que usen los componentes nativos de los dispositivos móviles a partir de un solo repositorio de código fuente y empleando los principios de React para la creación de interfaces de usuario.

Según (Boduch 2017, p. 210), React Native usa una método que permite realizar llamadas asíncronas al sistema operativo móvil subyacente, que comunica a las API de widgets nativos. Hay un motor de JavaScript y la API de React que son muy similares al de React para la Web. La principal diferencia está en el objetivo; en lugar de un DOM, hay llamadas API asíncronas. En el Gráfico 3-2, se representa las llamadas asíncronas de React Native a través del engine al sistema operativo móvil que a su vez que llaman a las API de widgets nativos.

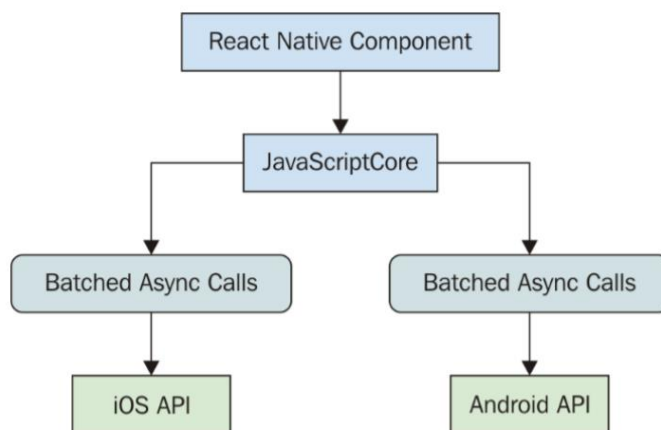


Gráfico 3-2. Llamadas asincrónicas de React Native.

Fuente: Adam Boduch, 2017

Es importante destacar de React Native es que si se ha trabajado con React es prácticamente lo mismo, ya que mantiene el JavaScript Extendend (JSX), para la creación de componentes, ciclo de vida de componentes, estilizado de componentes a través de hojas de estilo entre muchos otros aspectos, además permite tener un acceso nativo a las APIs de los recursos de los dispositivos móviles como cámara, micrófono, etc.

2.7.3.2. Expo

Expo en su página oficial se define como un framework y una plataforma para aplicaciones universales de React. Básicamente es un conjunto de herramientas y servicios construidos alrededor de React Native y plataformas nativas que ayudan a desarrollar, construir, implementar e iterar rápidamente en iOS, Android y aplicaciones web desde la misma base de código JavaScript/TypeScript. Además, facilita en gran medida en proceso de configuración inicial del proyecto, ya que al contrario del entorno vanilla de React Native, para la configuración inicial con Expo no hace falta tener Android Studio o Xcode para un primer compilado de la aplicación, sino que se puede ejecutar el proyecto de forma directa a partir de un enlace que genera el cliente de Expo. Otro punto importante es que Expo instala todas las librerías y componentes necesarios para que el proyecto de React Native funcione sin ningún problema, además, existen librerías optimizadas para expo que permiten acceder de forma sencilla a recursos de los dispositivos móviles y otras que están integradas con servicios de terceros como Stripe o Firebase.

2.8. Calidad del producto software

Conforme a la (ISO/IEC 2011, pArr. 2), la calidad del producto software se puede interpretar como “el grado en que dicho producto satisface los requisitos de sus usuarios aportando de esta manera un valor”. Son exactamente estos requisitos (funcionalidad, eficiencia, seguridad, mantenibilidad, etc.) los que se encuentran representados en el modelo de calidad, el cual secciona la calidad del producto en características y sub-características. En el Gráfico 4-2, se visualiza las 8 características de calidad que componen la ISO/IEC 25010:

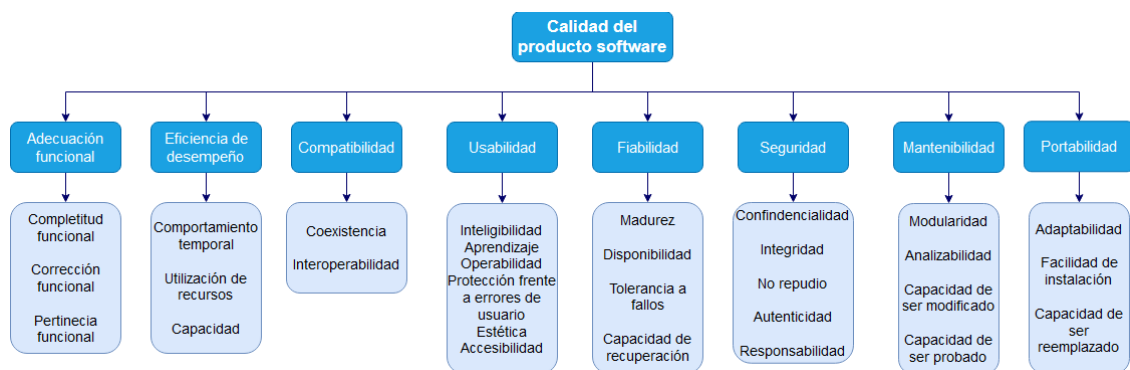


Gráfico 4-2. Característica de la calidad del producto software.

Fuente: ISO/IEC, 2011

Es importante resaltar que para este trabajo se evalúa la calidad del producto software basado en la eficiencia de desempeño y facilidad de modificación (también llamado capacidad para ser modificado), siendo esta última una subcaracterística de la mantenibilidad.

Se escogieron estas dos métricas para evaluar el producto software debido a que en la actualidad se requieren sistemas informáticos eficientes al momento de usar los recursos disponibles; y a su vez se necesita que estos sean fáciles de mantener, permitiendo implementar o corregir funcionales con un mínimo esfuerzo.

2.8.1. *Mantenibilidad*

De acuerdo a la (ISO/IEC 2011, pArr. 8), la mantenibilidad se define como “la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas o perfectivas”. En el Gráfico 5-2, se visualiza las subcaracterística de la mantenibilidad según la ISO/IEC 25010.

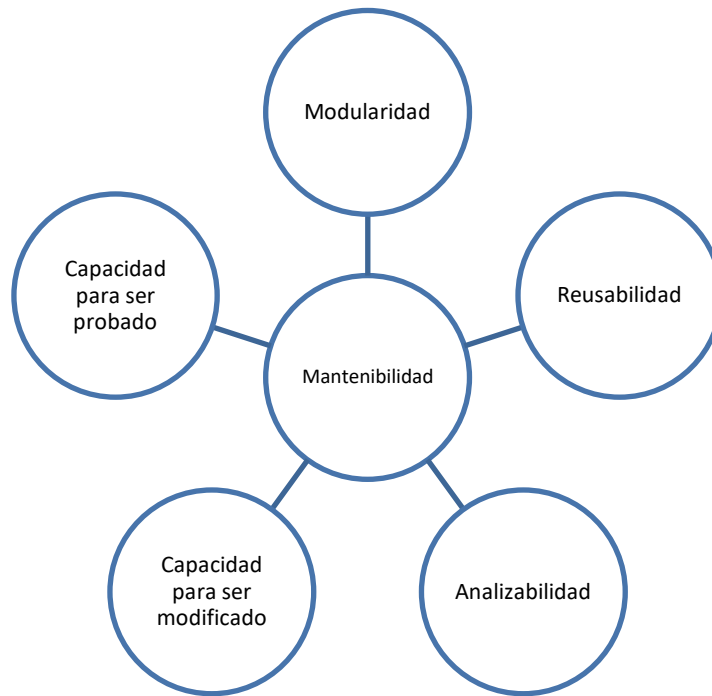


Gráfico 5-2. Subcaracterísticas de la mantenibilidad.

Realizado por: Bone Edward y Ochoa José, 2022

Conforme a la (ISO/IEC 2011, pArr. 8), las subcaracterísticas de la mantenibilidad son las siguientes:

- **Modularidad:** es la capacidad de un sistema o programa de ordenador que ayuda a que un cambio en un componente tenga un impacto mínimo en el resto del proyecto.
- **Reusabilidad:** se define como la capacidad de un activo que permite que sea utilizado en más de un sistema software o en la construcción de otros activos.
- **Analizabilidad:** es la facilidad con la que se puede evaluar el impacto de un cambio específico sobre el resto del software, diagnosticar las deficiencias o causas de fallos en el software, o identificar las partes a modificar.
- **Capacidad para ser modificado:** se define como la capacidad del producto que permite que sea modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño del sistema.
- **Capacidad para ser probado:** es la facilidad con la que se pueden establecer criterios de prueba para un sistema o componente y con la que se pueden llevar a cabo las pruebas para determinar si se cumplen dichos criterios.

2.8.1.1. *Capacidad para ser modificado*

En base a la (ISO/IEC 2011, pArr. 8) la capacidad de ser modificado es “la facultad del producto que permite que sea modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño”. Para poder medir la facilidad de modificación es necesario definir las métricas con

las cuales se evaluará el producto software, de acuerdo a (IEEE 1990, p. 47), define el termino métrica como “una medida cuantitativa del grado en el que un sistema, componente o proceso posee un atributo dado”.

Según la (ISO/IEC 2016, p. 41) las métricas que conforman la facilidad de modificación son las siguientes:

- **Complejidad ciclomática (métrica de calidad interna):** mide la complejidad estructural de un código fuente; para lo que cuenta las instrucciones condicionales, bucles, salidas de métodos y clausulas AND y OR dentro de los condicionales.
- **Profundidad de herencia (métrica de calidad interna):** mide qué tan profunda es la jerarquía de la herencia de las clases involucradas en una determinada función; para lo cual cuenta las jerarquías empleadas en una determinada función o método.
- **Grado de localización de corrección de impacto (métrica de calidad interna y externa):** mide hasta qué punto los problemas causados pueden tener como consecuencia un mantenimiento; para lo cual cuenta el número de fallas aparecidas después que se ha resuelto un fallo y cuenta el número de fallas resultas.
- **Complejidad de modificación (métrica de calidad externa):** mide con qué facilidad el desarrollador puede modificar el software para resolver problema; para lo cual toma el tiempo de trabajo que le toma al desarrollador modificar y contar el número de modificaciones.
- **Índice de éxito de modificación (métrica de calidad externa):** mide hasta qué punto puede el sistema ser operado sin fallas después del mantenimiento; para lo cual se cuenta el número de problemas dentro de un determinado período antes de mantenimiento y contar el número de problemas en el mismo período después del mantenimiento.

Por otra parte, (Mena 2020, p. 17) relaciona la facilidad de modificación con el índice de mantenibilidad que contiene las siguientes métricas:

- Volumen de halstead (HV).
- Complejidad ciclomática (CC).
- Número promedio de líneas de código (LOC).
- Porcentaje de comentarios a LOC en un módulo (COM).

Existe otro autor, (Pardo 2018, p. 43) quien utilizó un software de apoyo (SonarQube) para medir la mantenibilidad y las características que la conforman, basándose en el estándar ISO/IEC 25010. Los indicadores que conforman la capacidad de ser modificado son los siguientes:

- **Mala práctica:** se define como la codificación que complica la capacidad de lectura del código fuente.

- **Redundante:** es el código redundante en un punto del programa, donde, si se ha calculado previamente y se garantiza que su resultado está disponible en ese punto. Si tales cálculos pueden ser identificados, pueden obviamente ser eliminado sin afectar el comportamiento del programa.
- **Diseño:** se refiere al diseño del código fuente que hace que sea difícil de comprender.
- **Duplicaciones:** indica el porcentaje de código duplicado que hay en el software.

Es importante recalcar que para este trabajo de integración curricular se escoge el ultimo método para evaluar la capacidad de ser modificado, apoyándose en la herramienta SonarQube que permite obtener los indicadores del atributo de calidad ya mencionado.

2.8.2. Eficiencia de desempeño

De acuerdo con (ISO/IEC 2011, pArr. 4), la eficiencia de desempeño es “el rendimiento relativo a la cantidad de recursos utilizados bajo determinadas condiciones”. Esta se compone de las siguientes sub-características:

- **Comportamiento temporal:** Los tiempos de respuesta y procesamiento y las ratios de throughput del sistema cuando lleva a cabo sus funciones bajo ciertas condiciones en relación con un banco de pruebas (benchmark) establecido.
- **Utilización de recursos:** Las cantidades y tipos de recursos usados una vez que el sistema desarrolla su funcionalidad bajo ciertas condiciones.
- **Capacidad:** Nivel en que un parámetro de un producto o sistema software cumplen con los requisitos.

Conforme con (ISO/IEC 2016, p. 41), en la Tabla 1-2 se definen las métricas a usar para evaluar la eficiencia de desempeño:

Tabla 1-1: Métricas para evaluar la eficiencia de desempeño.

Sub-característica	Métrica
Comportamiento temporal	Tiempo de respuesta: Obtener el tiempo aproximado para finalizar una tarea.
	Tiempo de espera: Obtener el tiempo desde que se ejecuta una instrucción para que empiece un trabajo, hasta que lo finaliza.

Utilización de recursos	Utilización de la memoria: Medir la cantidad de memoria utilizada para terminar una tarea.
Capacidad	<p>Número de peticiones en línea: Obtener el número de peticiones online pueden ser atendidas en una unidad de tiempo.</p> <p>Número de accesos simultáneos: Obtener cuantos clientes pueden ingresar al sistema simultáneamente en un tiempo determinado.</p> <p>Ancho de banda: Obtener el valor máximo absoluto de transmisión requerida para cumplir con las funciones.</p>

Fuente: ISO/IEC, 2016

Realizado por: Bone Edward y Ochoa José, 2022

Para (Apache Software Foundation 2021, pArr. 1), otra forma de evaluar la eficiencia de desempeño es con la utilización de herramientas de software especializadas para ese propósito, una de estas es JMeter la cual es una herramienta de carga open source completa, implementada en Java que permite realizar test de comportamiento funcional y medir el rendimiento. También se puede utilizar para realizar pruebas de estrés, posee la capacidad de realizar desde una solicitud sencilla hasta secuencias de requisiciones que permiten diagnosticar el comportamiento de una aplicación en condiciones de producción

Cabe mencionar que en este trabajo de integración curricular para evaluar la eficiencia de desempeño se mide solo la subcaracterística de utilización de recursos, donde para la utilización de CPU se emplea los indicadores de la ISO/IEC 25023 y en la utilización de memoria se emplea un enfoque propio. Esto se debe a que para la utilización de memoria el estándar actual tiene un enfoque de utilizar la mayor cantidad de memoria posible, en cambio, el producto software desarrollado en este trabajo de integración curricular trata de consumir la menor cantidad de memoria posible (optimizar los recursos disponibles).

2.9. Trabajos relacionados

El trabajo de (Wehbe 2021) llamado “Análisis de Viabilidad para Contadores de Aforo” realiza el desarrollo de una aplicación usando un sistema de inteligencia artificial desarrollado en Python y OpenCV, además, la aplicación web desarrollada utiliza las tecnologías: React, NestJS, TypeScript y GraphQL. Al igual que (Wehbe 2021), en el presente trabajo de integración curricular se usa el framework de NestJS para el lado del servidor, el aporte que se realiza es la utilización

de servicios web REST en lugar de la tecnología GraphQL, para la comunicación de la aplicación móvil con el backend, también se implementan sockets para interactuar en tiempo real. El desarrollo se realiza en forma modular para obtener un producto de software más mantenible y escalable.

La investigación "Red de lockers para entrega y recolección de paquetes, integrando a vendedores y compradores de e-commerce mediante plataforma en línea" de (Gonzales Araki et al. 2018), propone el proceso para los casilleros inteligentes, mediante el cual se presenta el modelo de negocio de una red de puntos de entrega que utiliza casilleros inteligentes de apertura digital, a través del uso de una aplicación móvil. El aporte de esta investigación es que brinda un marco de trabajo de los casilleros inteligentes lo cual es uno de los objetos de estudios del presente trabajo de integración curricular. La diferencia radica que (Gonzales Araki et al. 2018) solo propone un marco el proceso de los casilleros inteligentes, mientras que el presente trabajo de integración curricular aplica el proceso en la empresa "Corporación Wolf S.A."

En el trabajo titulado "Software maintainability. Considerations for its specification and validation" de, (Farfán y Vega 2020), tratan sobre la criticidad de la mantenibilidad de los productos software. El objetivo de su trabajo es presentar algunas consideraciones importantes sobre el proceso de mantenimiento, de tal forma que, posteriormente se pueda validar que efectivamente el producto software es mantenible. Al igual que (Farfán y Vega 2020), el presente trabajo se enfoca en los aspectos de la mantenibilidad de una aplicación, sin embargo, en este trabajo solo se evalúa la capacidad para ser modificado, la cual es una subcaracterística de la mantenibilidad planteada dentro de la ISO 25010, esta puede ser medida durante el proceso de desarrollo.

CAPÍTULO III

3. MARCO METODOLÓGICO

En el presente capítulo correspondiente al marco metodológico se describen los mecanismos que se utiliza para la resolución de la problemática que tiene la empresa “Corporación Wolf S.A.”, tanto en técnicas como en herramientas utilizadas en el análisis y en el desarrollo de la aplicación móvil del conductor, además, para el proceso de desarrollo se utiliza la metodología SCRUM.

3.1. Diseño de la investigación

En la presente sección se detallan los métodos, técnicas y tipo de investigación que ayuda a resolver la problemática de la Empresa “Corporación Wolf S.A.”.

3.1.1. Tipo de investigación

Para el presente trabajo de integración curricular se emplean los tipos de investigación: aplicada, descriptiva y cualitativa, a continuación, se describe cada tipo de investigación:

- **Investigación aplicada:** Para el desarrollo de la aplicación móvil del conductor se emplea la investigación aplicada debido a que el enfoque es específico y delimitado, además, está dirigido al desarrollo tecnológico como mejorar o implementar sistemas.
- **Investigación descriptiva:** Esta investigación permite detallar las características y configuración básica que se necesitan para poder desarrollar los módulos de la aplicación móvil y backend, además de que se detalla todo el proceso desde la formulación del problema, implementación de los módulos y medición de los atributos de calidad seleccionados.
- **Investigación cuantitativa:** A través de esta investigación se analiza la eficiencia de desempeño y facilidad de modificación, para ello se realiza la medición de las métricas seleccionadas de cada atributo de calidad con el fin de obtener un valor numérico que los represente.

3.1.2. Métodos de investigación

A continuación, se presentan los métodos de investigación que se utilizan en el desarrollo del presente trabajo de investigación curricular:

3.1.2.1. Método Analítico

El método analítico sirve para dividir el desarrollo del trabajo de integración curricular, donde se fragmentan los objetivos en partes más pequeñas para facilitar la comprensión, elaboración y obtención de resultados. Así mismo el desarrollo del sistema está segmentado en diferentes submódulos, para tener un mejor control sobre el producto, de esta forma se garantiza calidad sobre el producto final. Además, se empleará la metodología SCRUM cuya filosofía de dividir el todo en partes se alinea con el método analítico.

3.1.2.2. Método Inductivo

El método inductivo es una técnica de razonamiento que se basa en la inducción para generar información para ello se basa en premisas particulares para generar conclusiones generales. Este método se usa para generalizar toda la información recolectada sobre el proceso actual que tiene la empresa para los envíos de paquetes.

3.1.2.3. Método Sintético

El método sintético es una forma de razonamiento que tiene como objetivo principal resumir los aspectos más relevantes de un proceso. Este método se usa en la evaluación de los atributos de calidad seleccionados de la aplicación móvil del conductor.

3.1.3. Técnicas de Investigación

Con el fin de obtener información relevante para desarrollar el presente trabajo de titulación se utilizaron las técnicas de investigación como: la entrevista para obtener los requisitos de la aplicación móvil del conductor, observación para analizar el proceso de la empresa y revisión de documentación para el desarrollo de las funcionalidades de la aplicación móvil del conductor. Estas técnicas se describen a continuación:

3.1.3.1. Entrevista

La entrevista es una técnica que consiste en establecer un diálogo directo entre el entrevistador y el entrevistado que tiene como fin obtener información relevante. Para el presente trabajo de integración curricular se realizó la entrevista al representante legal de la empresa Ing. José Villarruel y el director del departamento de sistemas Ing. Christian Guevara, con el fin de

recolectar los requisitos de la aplicación móvil del conductor y backend. Esta entrevista se la realiza mediante una serie de preguntas claras y directas. El documento completo de la entrevista se encuentra en el ANEXO A.

3.1.3.2. Observación

La observación es una técnica que permite visualizar el proceso actual de la empresa de paquetería liviana desde que accede al sistema hasta la entrega del paquete a una empresa de terceros.

3.1.3.3. Revisión de documentación

La revisión de documentación es una técnica que ayuda a tener una idea más clara y concisa sobre un tema en específico, en el presente trabajo se utiliza la documentación oficial de NestJS, PostgreSQL, Expo y React Native para el desarrollo de la aplicación móvil del conductor, además, se usa la documentación de la empresa para aclarar el proceso de reparto que tiene la empresa actualmente.

A continuación, en la Tabla 1-3, se visualizan cada uno de los objetivos específicos del presente trabajo de integración curricular asociados a su respectivo método, técnica y fuente.

Tabla 1-3: Métodos, Técnicas y Fuente.

Objetivos	Métodos	Técnicas	Fuentes
Analizar el proceso actual de reparto de paquetería liviana de la empresa “Corporación Wolf S.A.”	Analítico Inductivo	BPMN	Estándar BPMN Stakeholder
Definir los requisitos necesarios de la aplicación móvil.	Analítico	Entrevista	Stakeholder
Desarrollar las funcionalidades de la aplicación móvil y backend utilizando la metodología SCRUM.	Scrum	Revisión de documentación Sprint Planning Sprint Retrospective	Internet Documentación de la página oficial Stakeholder

Evaluar la eficiencia de desempeño y facilidad de modificación de la aplicación móvil y backend.	Analítico Sintético	Observación ISO/IEC 25010	Aplicación móvil Backend
--	------------------------	------------------------------	-----------------------------

Realizado por: Bone Edward y Ochoa José, 2022

3.1.4. Variables e Indicadores

En este trabajo de integración curricular se evalúa la facilidad de modificación la cual es una subcaracterística de la mantenibilidad. Además, se evalúa la eficiencia de desempeño mediante el estándar ISO/IEC 25010.

3.1.4.1. Operacionalización conceptual de variables

En la Tabla 2-3, se visualiza la operacionalización conceptual de las variables que se van a medir en este trabajo.

Tabla 2-3: Operacionalización conceptual de variables.

Formulación del problema	Variable	Tipo	Concepto
¿Cómo se va a evaluar la eficiencia de desempeño de la aplicación móvil y su backend?	Eficiencia de desempeño	Cuantitativo Compleja	Esta característica representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones.
¿Cómo se va a evaluar la facilidad de modificación de la aplicación móvil y su backend?	Facilidad de modificación	Cuantitativo Compleja	Esta característica representa la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas o perfectivas.

Realizado por: Bone Edward y Ochoa José, 2022

3.1.4.2. Operacionalización metodológica de variables

En la Tabla 3-3, se visualiza la operacionalización metodológica de las variables que se van a medir en este trabajo.

Tabla 3-3: Operacionalización metodológica de variables.

Formulación del problema	Variable	Categoría	Indicador	Técnica	Fuente
¿Cómo se va a evaluar la eficiencia de desempeño de la aplicación del móvil y su backend?	Eficiencia de desempeño	Utilización de recursos	Utilización del CPU Utilización de la memoria	Observación	Aplicación móvil Backend
¿Cómo se va a evaluar la facilidad de modificación de la aplicación móvil y su backend?	Facilidad de modificación	Mantenibilidad	Duplicaciones Mala Practica Redundancia Diseño	Observación	Aplicación móvil Backend

Realizado por: Bone Edward y Ochoa José, 2022

3.1.5. Población y muestra

La población y muestra que se estudia en el presente trabajo de integración curricular se divide en una muestra por cada atributo de calidad a medir, esto debido a que, cada uno de los atributos se enfocan en aspectos diferentes, por lo cual es no es factible compartir la misma muestra o escenario de evaluación.

3.1.5.1. Eficiencia de desempeño

Para medir la eficiencia de desempeño en la aplicación móvil y backend, se tomaron en cuenta todos los 19 requisitos del sistema, de los cuales se escogieron aquellos que pertenecen a los dos procesos principales: registrar conductor (3 requisitos) y repartir pedidos (4 requisitos). De los procesos escogidos se mide la utilización de recursos (utilización del CPU y utilización de la Memoria). En la Tabla 4-3, se visualizan los requisitos que conforman los procesos principales del sistema.

Tabla 4-3: Procesos principales del sistema.

Proceso	Requisito
Registrar conductor	El sistema permitirá crear la cuenta del conductor (correo, teléfono, ciudad, número de cedula, nombres completos).
	El sistema permitirá subir la documentación correspondiente al conductor (fotografía del conductor, fotografía de la cedula, fotografía de la licencia, fotografía de confirmación de identidad, fotografía de antecedentes no penales).
	El sistema permitirá subir la documentación del vehículo del conductor (fotografía de la matrícula, número de la placa, marca del vehículo, fotografía del vehículo mostrando la placa).
Repartir pedidos	El sistema permitirá al conductor establecer el destino al cual se dirige.
	El sistema mostrará la ruta que debe seguir el conductor para recoger los paquetes.
	El sistema mostrará la ruta que debe seguir el conductor para entregar los paquetes.
	El sistema determinará el valor de tarifa a cobrar por kilómetro recorrido de la ruta de transporte del pedido.

Realizado por: Bone Edward y Ochoa José, 2022

3.1.5.2. Facilidad de modificación

Para poder medir la facilidad de modificación se toma en cuenta todas las líneas de código del proyecto de la aplicación móvil, así mismo, para el backend se selecciona todas las líneas del código de los módulos que se implementaron durante el desarrollo, estos módulos son: Drivers, Auth, Orders, Notifications, Users, Files, Email y Database.

3.2. Proceso de reparto de la empresa

En la actualidad la empresa “Corporación Wolf S.A.” realiza el proceso de envío de paquetes de sus productos a los domicilios de sus clientes, para lo cual utiliza los servicios de una empresa externa de transporte de paquetería.

3.2.1. Proceso actual de reparto de paquetería

Mediante una reunión con el representante legal de la empresa “Corporación Wolf S.A.” se detalla el proceso actual de envío de paquetes, en este proceso intervienen tres actores principales el empleado de la empresa, la empresa de reparto (externa) y el vehículo de transporte.

A continuación, se presenta la secuencia de actividades del proceso para el envío de paquetes:

1. Acceder al sistema: el empleado ingresa al sistema con su usuario y contraseña.
2. Completar la guía de envío: el empleado genera la guía con los datos que se le solicitan (quien envía, que se envía, peso, datos del receptor).
3. Obtener el código de seguimiento: el sistema proporciona al empleado un código de seguimiento de la guía generada.
4. Imprimir la guía de envío: la empresa de reparto imprime la guía del paquete a enviar.
5. Solicitar un vehículo de transporte: el empleado coordina manualmente el envío con la empresa de reparto.
6. Enviar un vehículo: envían un camión con una persona a recoger los paquetes.
7. Cargar los paquetes a transportar: el conductor carga los paquetes al vehículo de transporte.

En el Gráfico 1-3, se visualiza el diagrama BPMN que contiene la secuencia de actividades del proceso actual de reparto de paquetería de la empresa.

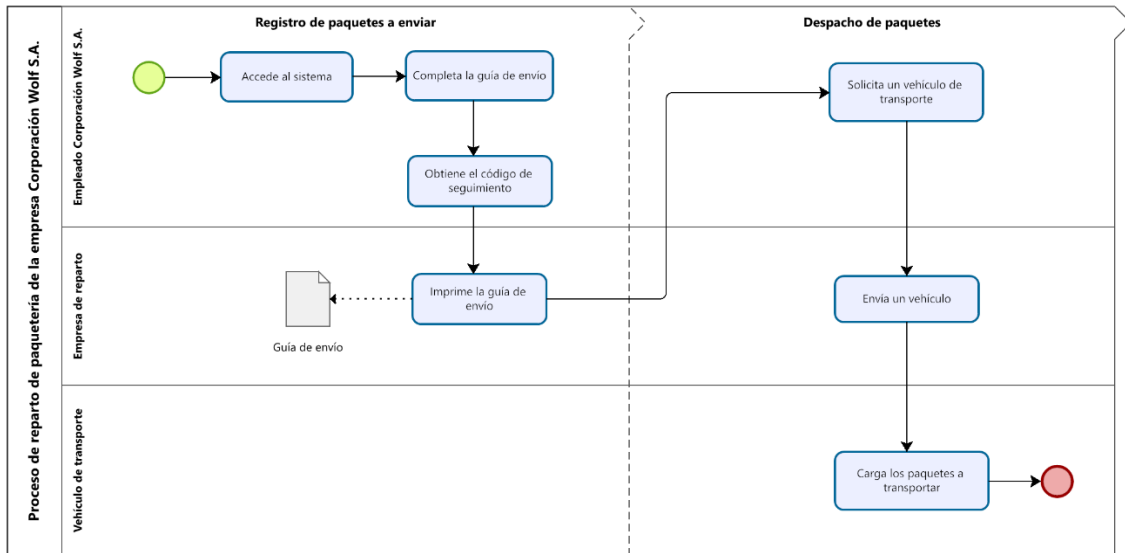


Gráfico 1-3. Proceso actual de reparto de paquetería.

Realizado por: Bone Edward y Ochoa José, 2022

3.2.2. *Proceso de reparto de paquetería con el sistema implementado*

En el proceso de reparto de paquetería con el sistema implementado intervienen cuatro actores principales: el cliente, sistema, conductor y destinatario; el sistema tiene un total de cuatro subprocesos: generar orden y medir paquetes a enviar, pago de la orden, recolección de los paquetes a transportar y entrega de orden a domicilio/casillero.

1. El cliente genera la orden, completa la información de envío y mide las dimensiones de los paquetes.
2. El cliente recibe una tarifa con el costo detallado del envío y realizará el pago del envío.
3. El conductor recibe las ordenes que coincidan con la ruta de destino y selecciona las ordenes que puede llevar.
4. Si la orden se entrega a domicilio, el conductor se dirige al domicilio del destinatario y realiza la entrega personalmente.
5. Si la orden se entrega a un punto de recolección (casillero), el conductor se dirige a la ubicación del casillero asignado y deposita el paquete.

En el Gráfico 2-3, se visualiza los subprocesos que contiene la secuencia de actividades que conforman el proceso de reparto de paquetería con el sistema implementado de la empresa.

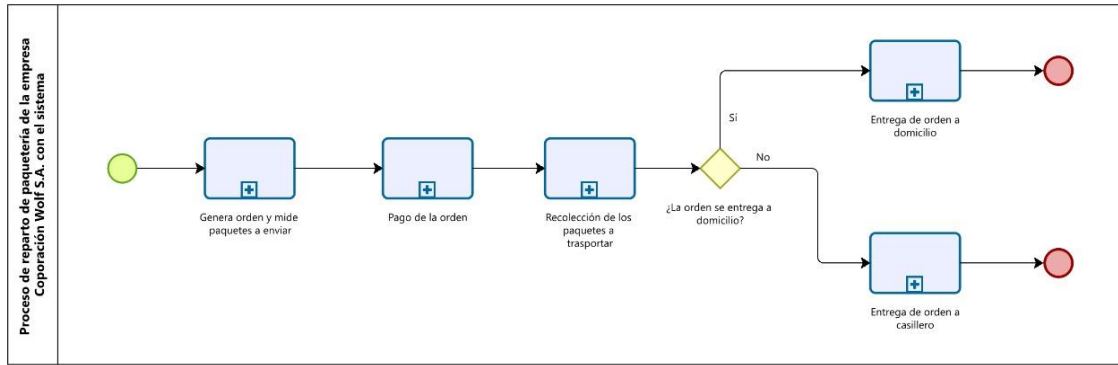


Gráfico 2-3. Diagrama de procesos con el sistema implementado.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 3-3, se visualiza el subproceso que contiene la secuencia de actividades que conforman el proceso entrega de la orden a domicilio.

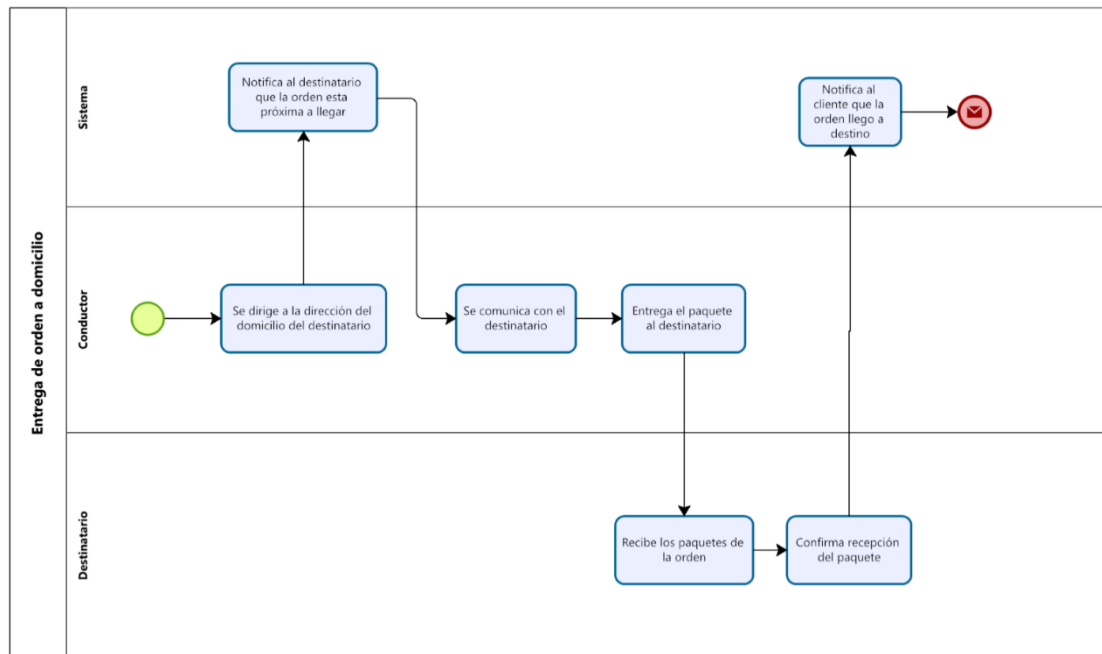


Gráfico 3-3. Diagrama de proceso de la entrega de orden a domicilio.

Realizado por: Bone Edward y Ochoa José, 2022

Para poder observar los subprocesos detallados del reparto de paquetería con el sistema implementado revisar el ANEXO B.

3.3. Estudio de factibilidad

El estudio de factibilidad determina los recursos económicos, técnicos y operativos para definir si el proyecto de integración curricular es viable, de esta manera se plantea al cliente un costo

aproximado de la aplicación del conductor y backend del sistema “Wolf Delivery”. Para lo cual se realiza el estudio de la factibilidad técnica, factibilidad operativa y la factibilidad económica. A continuación, se detallan de forma específica los diferentes tipos de factibilidad.

3.3.1. *Facilidad técnica*

Para realizar el estudio de factibilidad técnica en el trabajo de integración curricular “Desarrollo de una aplicación móvil y backend para apoyar a los conductores en la gestión de reparto de paquetería en la empresa "Corporación Wolf S.A.", se requiere conocer los recursos existentes y requeridos de hardware, software y recursos humanos.

En la Tabla 5-3 y Tabla 6-3, se puede visualizar el listado de los recursos de hardware existentes y requeridos correspondientemente.

Tabla 5-3: Recursos de hardware existente.

Cantidad	Descripción	Observación
1	Computadora portátil Dell Inspiron 5593	Uso para el desarrollo de la aplicación
1	Computadora portátil Toshiba Satellite S55-C	Uso para el desarrollo de la aplicación
1	Impresora EPSON	Impresión de documentos
1	Celular Android	Prueba de la aplicación
1	Celular Iphone	Prueba de la aplicación

Realizado por: Bone Edward y Ochoa José, 2022

Tabla 6-3: Recursos de hardware requerido.

Cantidad	Descripción	Observación
1	Droplet de DigitalOcean	Despliegue del servidor
1	Prototipo funcional de un casillero inteligente	Pruebas de la aplicación.

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 7-3 y Tabla 8-3, se puede visualizar el listado de los recursos software existente y requeridos correspondientemente.

Tabla 7-3: Recursos de software existente.

Cantidad	Descripción	Observación
2	Windows 10	Legal
1	DataGrip	Legal

Realizado por: Bone Edward y Ochoa José, 2022

Tabla 8-3: Recursos de software requeridos.

Cantidad	Descripción	Observación
2	Visual Studio Code	Libre
2	Insomnia	Libre
1	PgAdmin	Libre
1	NestJS (framework para el backend)	Libre
1	Expo (framework para la aplicación móvil)	Libre

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 9-3, se puede visualizar los recursos humanos requeridos para el desarrollo de la aplicación móvil y backend.

Tabla 9-3: Recursos humanos requeridos.

Cantidad	Función	Formación	Experiencia
2	Desarrollador	Estudios de tercer nivel	Desarrollador de aplicaciones con NodeJS

Realizado por: Bone Edward y Ochoa José, 2022

El desarrollo de la aplicación móvil del conductor y backend, es viable debido el hardware requerido es proporcionado por la empresa, además todo el software a empleado es software libre.

3.3.2. Facilidad operativa

La factibilidad operativa permite determinar la posibilidad que se utilice de forma correcta la aplicación móvil del conductor por parte de sus usuarios. Teniendo en cuenta este concepto, se determina que la aplicación móvil del conductor es viable ya que su objetivo es implementar el servicio de reparto de paquetería liviana, haciendo que este proceso sea eficiente y se realice sin mayor dificultad.

3.3.3. Factibilidad económica

El estudio de factibilidad económica permite determinar el costo-beneficio de la aplicación móvil del conductor y backend que se desarrolla, esto permite establecer si el proyecto es factible o no de realizar.

3.3.3.1. Costos

Los costos describen los gastos empleados en hardware, luz e internet durante el tiempo de desarrollo de la aplicación móvil y backend. En la Tabla 10-3, se realiza una descripción más detallada del valor de los gastos incurridos durante el desarrollo.

Tabla 10-3: Costos del proyecto.

Descripción	Costo/Mes (\$)	Mes	Total (\$)
Luz	10	4	40.00
Internet	20	4	80.00
Droplet de DigitalOcean (Basic Plan)	5	2	10.00
Computadora portátil Dell Inspiron 5593			1200.00
Computadora portátil Toshiba Satellite S55-C			984.00
Celular Android			200.00
Celular Iphone			448.00
Costo Total del proyecto			2962.00

Realizado por: Bone Edward y Ochoa José, 2022

El costo total del presente trabajo de integración curricular está financiado por la Corporación Wolf S.A.

3.3.3.2. Beneficios

Para comprender mejor los resultados, se analizan los beneficios tangibles e intangibles que surgirán después la implementación de esta aplicación móvil del conductor.

Los beneficios tangibles son:

- Menor tiempo empleado en el proceso de reparto de paquetería liviana.

- Mayor control sobre el proceso de reparto de paquetería.
- Reducción de gastos.

Los beneficios intangibles son:

- Mejor ambiente laboral.
- Ayuda en la toma de decisiones de los administradores al tener más información disponible del proceso.
- Información organizada del proceso de reparto de paquetería liviana.

3.4. Desarrollo de la aplicación mediante SCRUM

La metodología para el desarrollo de la aplicación es SCRUM una metodología ágil, la cual permite una comunicación directa con los stakeholders, que se centra en la entrega continua del producto final (Sprint) para poder verificar las funcionalidades del producto.

3.4.1. Fase de planificación

En esta fase se analiza el problema con el propósito de definir los tipos de usuarios en el sistemas y requerimientos que tiene el cliente para el sistema. Para recolectar esta información se realiza reuniones todos los involucrados en este proyecto.

3.4.1.1. Miembros y Roles involucrado en el proyecto

Las personas involucradas en el desarrollo del proyecto y sus respectivos roles se encuentran en la Tabla 11-3, que se puede visualizar a continuación:

Tabla 11-3: Miembros y roles involucrados en el proyecto.

Miembro	Rol	Contacto
Ing. José Villarruel	Product Owner	jose@roboticawolf.com
Ing. Christian Guevara	Scrum Máster	christian@roboticawolf.com
José Ochoa	Development Team	jose.ochoav@epoch.edu.ec
Edward Bone	Development Team	edward.bone@epoch.edu.ec

Realizado por: Bone Edward y Ochoa José, 2022

3.4.1.2. Tipos de usuarios y Roles involucrados en el sistema

En la Tabla 12-3, se detalla los tipos de usuarios con sus respectivos roles, cada uno asignado con diferentes funcionalidades dentro del sistema.

Tabla 12-3: Tipos de usuarios y roles involucrados en el sistema.

Tipo de usuario	Rol
Conductor	Se encarga de realizar el proceso de transporte de paquetes desde un punto a otro.
Administrador	Se encarga de gestionar todo el sistema a través del sistema web.
Cliente	Se encarga de generar las ordenes con los paquetes.

Realizado por: Bone Edward y Ochoa José, 2022

3.4.1.3. Especificación de requerimientos

Para la especificación de requerimientos se empleó una matriz de trazabilidad, la cual es una tabla donde se vinculan los requisitos del producto desde su origen hasta sus entregables, según la PMBOOK una matriz de trazabilidad de requerimientos (RTM por sus siglas en inglés) debe tener los siguientes ítems:

- Identificación
- Subidentificación
- Necesidades del negocio
- Oportunidades
- Metas objetivos
- Objetivo del proyecto
- Entregables
- Diseño del producto
- Desarrollo del producto
- Escenarios de prueba

En la Tabla 13-3, se muestra la matriz de trazabilidad de requerimientos basada la guía del PMBOOK 6ta edición

Tabla 13-3: Matriz de trazabilidad de requisitos.

Identificación	Sub identificación	Descripción del requisito	Versión	Estado actual	Última fecha estado registrado	Criterios de aceptación
001	1.1	El sistema permitirá crear la cuenta del conductor (Correo, Teléfono, Ciudad, Número de Cedula, Nombres Completos).	1.0.0	Completado	03/11/2021	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.
	1.2	El sistema permitirá subir la documentación correspondiente al conductor (Fotografía del Conductor, Fotografía de la Cedula, Fotografía de la Licencia, Fotografía de confirmación de Identidad, Fotografía de antecedentes no penales).	1.0.0	Completado	05/11/2021	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.
	1.3	El sistema permitirá subir la documentación del vehículo del conductor (Fotografía de la Matricula, Número de la placa, Marca del vehículo, Fotografía del vehículo mostrando la placa).	1.0.0	Completado	10/11/2021	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.

Realizado por: Bone Edward y Ochoa José, 2022

La matriz de la Tabla 13-3, se implementa para la gestión del proyecto en la herramienta Case “Power Designer” como se muestra en la Figura 1-3.

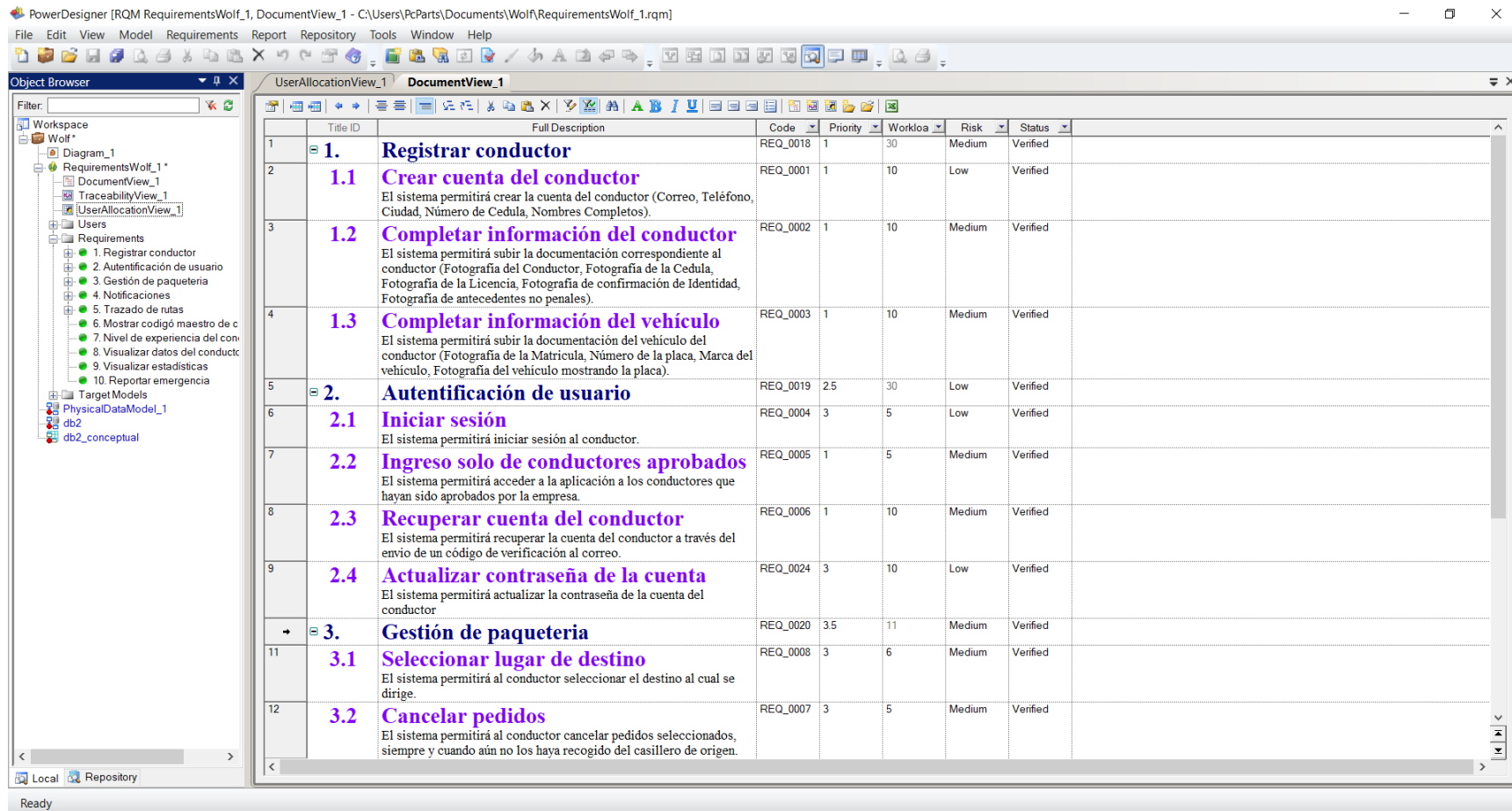


Figura 1-3. Matriz de trazabilidad de requisitos implementado en Power Designer.

Realizado por: Bone Edward y Ochoa José, 2022

Para poder visualizar la matriz de trazabilidad de requerimientos completa, esta adjunta en el ANEXO C.

3.4.1.4. Product Backlog

El product backlog se emplea para listar todas las funcionalidades de la aplicación que fueron determinadas por el Product Owner luego de haber mantenido diferentes reuniones. Con esta herramienta se puede llegar a dar una estimación del esfuerzo total de los requerimientos planteados de acuerdo con las necesidades del negocio, además se puede priorizar ciertos requerimientos que tengan mayor relevancia que otros. Una vez finalizado el análisis de requerimientos se obtuvieron 19 historias de usuario y 8 historias técnicas. En la Tabla 14-3, se listan las historias de usuario e historias técnicas utilizadas den el desarrollo del sistema.

Tabla 14-3: Product Backlog.

ID	Historias de usuario/técnicas	Puntos Estimados	Prioridad
HT-01	Entrevista con el cliente para la definición de requerimientos.	7	Alta
HT-02	Diseño de la arquitectura del sistema.	5	Alta
HT-03	Definición de estándar de codificación del sistema.	3	Media
HT-04	Diseño de las interfaces de usuario.	10	Media
HT-05	Diseño e implementación de la base de datos.	8	Alta
HT-06	Generar el diccionario datos (base de datos).	2	Baja
HT-07	Instalar los entornos de desarrollo (Visual Studio Code).	5	Media
HT-08	Despliegue del backend en el servidor web (Digital Ocean).	40	Media
HU-01	Crear cuenta del conductor.	10	Baja
HU-02	Completar información del conductor.	10	Baja

HU-03	Completar información del vehículo.	10	Baja
HU-04	Iniciar sesión.	5	Media
HU-05	Permitir ingreso solo a los conductores aprobados.	5	Baja
HU-06	Recuperar cuenta de conductor.	10	Media
HU-07	Actualizar contraseña de la cuenta.	10	Media
HU-08	Determinar tarifa a cobrar de la ruta.	10	Alta
HU-09	Seleccionar lugar de destino.	6	Media
HU-10	Generar notificaciones por eventos.	10	Baja
HU-11	Visualizar notificaciones.	5	Baja
HU-12	Mostrar ruta de recogida del pedido.	7	Alta
HU-13	Mostrar ruta de entrega del pedido.	7	Alta
HU-14	Mostrar código maestro de casillero.	5	Media
HU-15	Reportar emergencia.	10	Media
HU-16	Nivel de experiencia del conductor.	20	Media
HU-17	Visualizar datos del conductor y vehículo.	5	Baja
HU-18	Visualizar estadísticas.	10	Baja
HU-19	Cancelar pedidos.	5	Media

Realizado por: Bone Edward y Ochoa José, 2022

3.4.2. Fase de diseño

En esta fase se realizaron actividades de modelado, diagramas y definición de los estándares de codificación y de base de datos con el fin de tener una idea más específica de lo que se va a desarrollar.

3.4.2.1. Diseño de la arquitectura

El diseño de la arquitectura del sistema se realizó empleando el modelado C4, el cual consiste en cuatro diagramas (contexto, contenedores, componentes y código). Es importante aclarar que solo se utiliza el diagrama de contexto del modelado C4, debido a que especifica los módulos del sistema, además, se puede visualizar los sistemas externos que interactúan con el mismo.

En el Gráfico 4-3, se puede visualizar el diagrama de la arquitectura de todo el sistema empleando la técnica de modelado C4, es importante remarcar que la parte correspondiente a este trabajo de integración (aplicación móvil del conductor y backend) se ilustra con el color rojo dentro de toda la arquitectura.

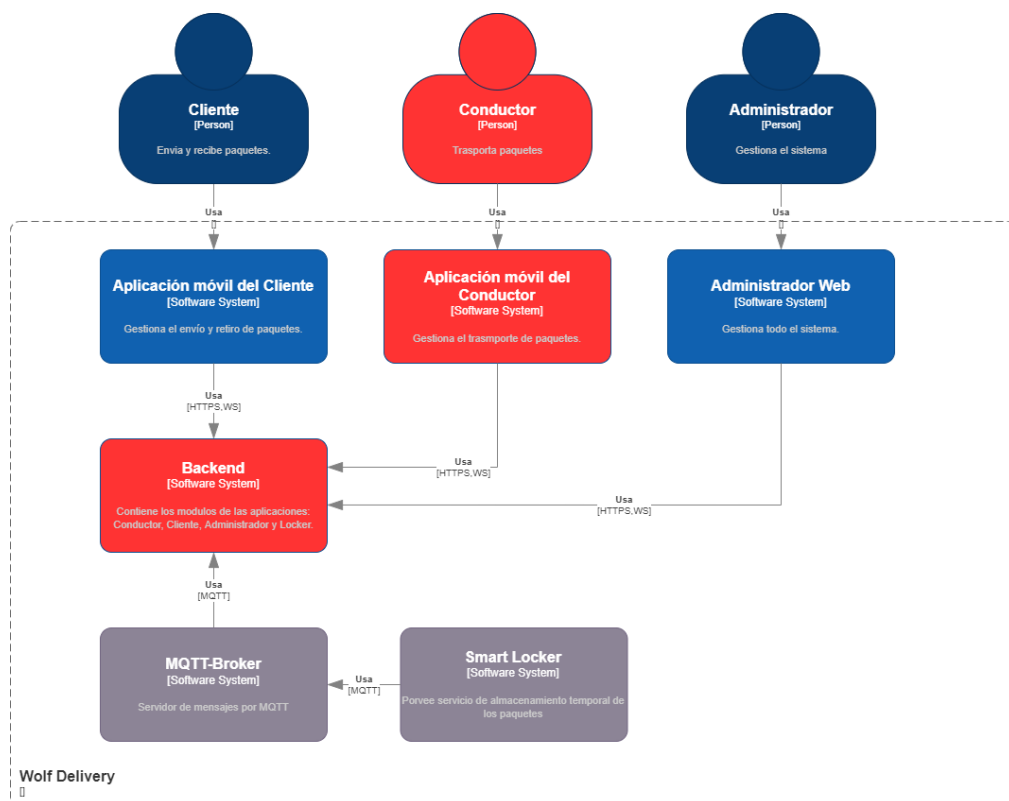


Gráfico 4-3. Diagrama la arquitectura del sistema empleando el modelado C4.

Realizado por: Bone Edward y Ochoa José, 2022

3.4.2.2. Organigrama de los módulos de del sistema

En el Gráfico 5-3, se presenta un organigrama de los módulos que tiene el sistema “Wolf Delivery” tanto para la aplicación móvil como el backend.

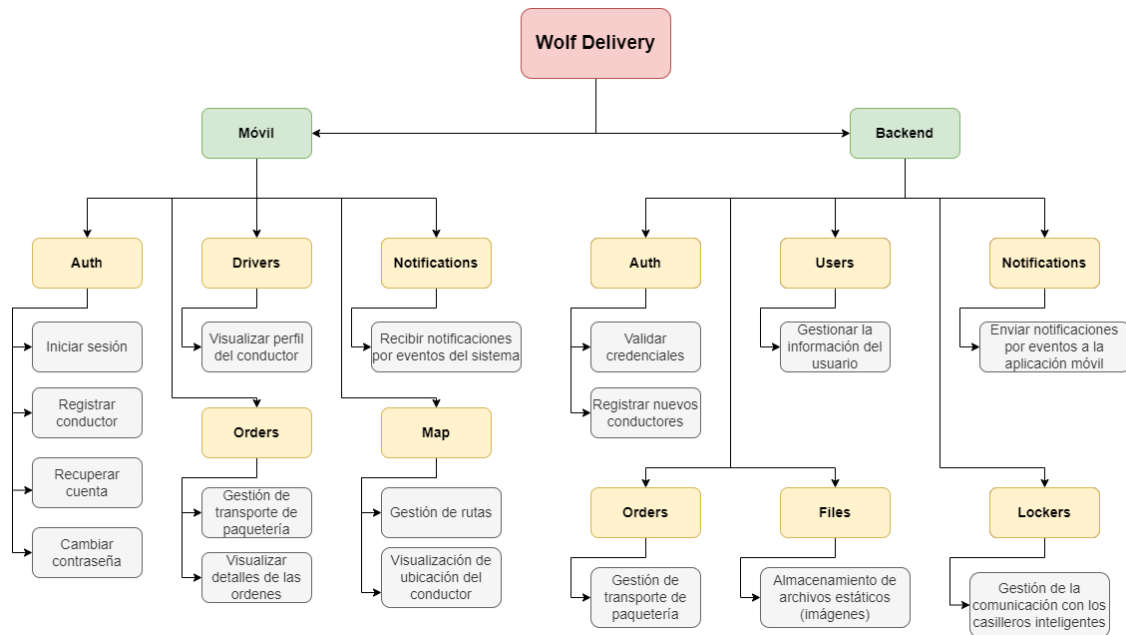


Gráfico 5-3. Organigrama de los módulos del sistema.

Realizado por: Bone Edward y Ochoa José, 2022

Los módulos correspondientes a la aplicación móvil del conductor se describen a continuación:

- **Módulo Auth:** este módulo es el responsable de realizar las acciones de inicio de sesión, registro de conductor, recuperación de la cuenta por parte del usuario y también permite cambiar la contraseña de la cuenta.
- **Módulo Drivers:** este módulo permite visualizar la información del conductor que ha iniciado sesión dentro de la aplicación.
- **Módulo Notifications:** este módulo es el responsable de recibir y visualizar las notificaciones recibidas en la aplicación desde el servidor por diferentes eventos.
- **Módulo Orders:** este módulo realiza la gestión de transporte de paquetería, visualizando los órdenes que se encuentren disponibles para ser recogidos por el conductor, además, se encarga de mostrar toda la información relacionada a las ordenes asignadas al conductor.
- **Módulo Map:** este módulo contiene toda la lógica necesaria para realizar el trazado de rutas, tanto para recoger las ordenes desde su origen, hasta la ruta a seguir para llevarlas a su destino. También es el responsable de visualizar la ubicación del conductor y realizar el seguimiento del mismo.

Los módulos correspondientes al backend de la aplicación móvil del conductor se describen a continuación:

- **Módulo Auth:** este módulo realiza la validación de las credenciales del usuario que intente iniciar sesión o realizar cualquier tipo de llamada a los diferentes servicios web. También, se encarga de realizar el registro de nuevos conductores.
- **Módulo Users:** este módulo es el responsable de gestionar toda la información de los usuarios y conductores.
- **Módulo Notifications:** este módulo es el encargado de enviar las notificaciones a la aplicación móvil del conductor por los diferentes eventos que se susciten durante el proceso de transporte de paquetería.
- **Módulo Orders:** este módulo realiza los procesos relacionados con la gestión de transporte de paquetería, además, asigna la orden al conductor que solicito transportarla.
- **Módulo Files:** este módulo es el responsable del almacenamiento de archivos estáticos (imágenes). También es el encargado de enviar los archivos estáticos que soliciten los clientes.
- **Módulo Lockers:** este módulo es el encargado de establecer la comunicación con los casilleros inteligentes para enviarles los diferentes comandos (abrir casillero, cerrar casilleros, entre otros).

3.4.2.3. Diagrama de despliegue

En el Gráfico 6-3, se presenta el diagrama de despliegue que tiene el sistema “Wolf Delivery”.

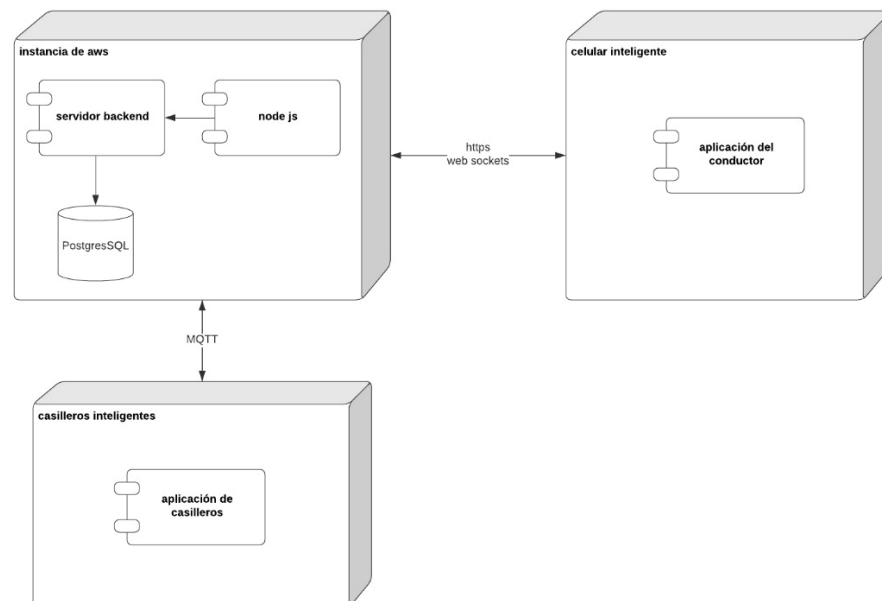


Gráfico 6-3. Diagrama de despliegue del sistema.

Realizado por: Bone Edward y Ochoa José, 2022

3.4.2.4. Estándar de codificación

Los estándares de codificación son un conjunto de buenas prácticas de desarrollo que ayudan a mantener un mismo estilo de código; permite la fácil lectura y comprensión del mismo, además, reduce el número de decisiones del desarrollador al momento de programar.

En la Tabla 15-3, se visualiza los estándares de codificación empleado en el desarrollo de la aplicación.

Tabla 15-3: Estándares de codificación.

	Tipo	Estándar	Ejemplo
Base de datos	Primary Key	snake_case	user_id
	Tablas	snake_case	user_role
	Atributos	snake_case	first_name
Código	Clase	PascalCase	User
	Método	camelCase	functionName
	Variable	camelCase	variableName

Realizado por: Bone Edward y Ochoa José, 2022

3.4.2.5. Diseño de base de datos

Con el propósito de mantener los datos organizados, almacenar y dar integridad a la información generada a través de la aplicación móvil del conductor del Sistema “Wolf Delivery”, se realiza el diseño de una base de datos. Esta base de datos se diseñó en base a los requerimientos del sistema, donde se puede identificar las entidades, atributos y sus respectivas relaciones.

En la Figura 2-3, se puede visualizar el modelo físico de la base de datos correspondiente al módulo orders.

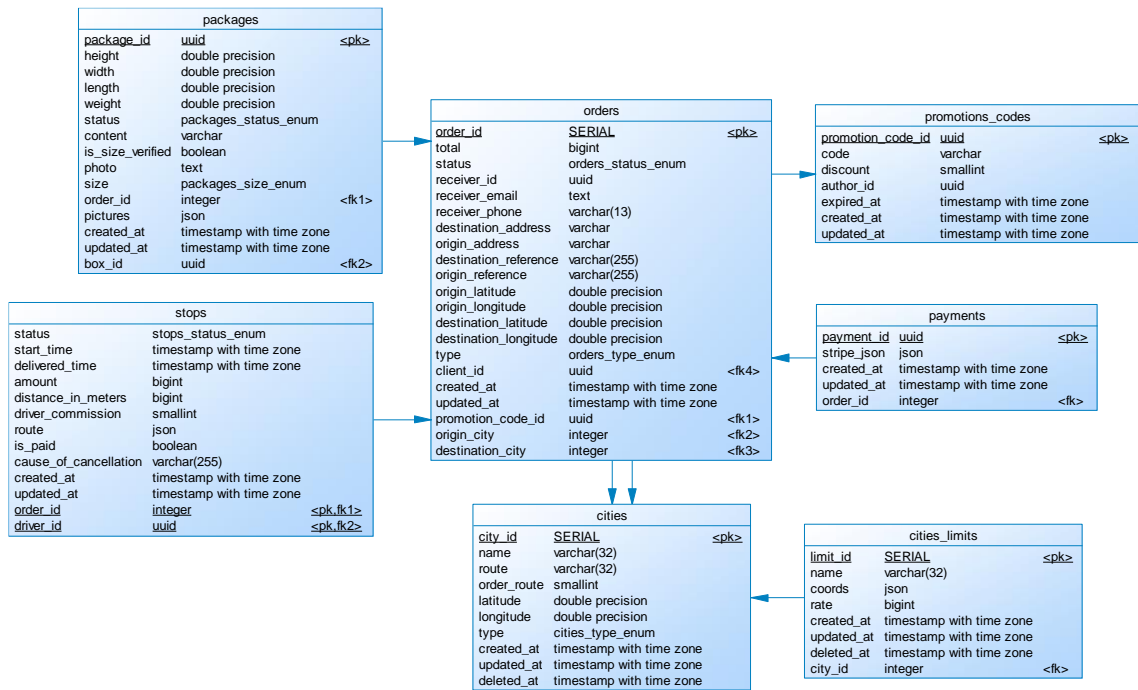


Figura 2-3. Diseño de la base de datos módulo orders.

Realizado por: Bone Edward y Ochoa José, 2022

Los modelos físicos restantes se encuentran adjuntos en el ANEXO D.

3.4.2.6. Diccionario de datos

El diccionario de datos de la base de datos permite visualizar la nomenclatura de los datos que se van a emplear durante la codificación del sistema, en este se describe el nombre, tipo de dato, valores aceptados, valores por defecto y claves primarias.

En la Tabla 16-3, se listan los tipos de datos pertenecientes a la entidad Users, creada para almacenar información del sistema.

Tabla 16-3: Diccionario de datos de la tabla Users.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
user_id	uuid	No	uuid_generate_v4()	Sí
dni	varchar(10)	No		No
first_name	varchar(20)	No		No
second_name	varchar(20)	No		No
first_last_name	varchar(20)	No		No
second_last_name	varchar(20)	No		No
email	varchar(150)	No		No
password	text	No		No

phone_number	varchar(13)	No		No
avatar	text	No	'https://...'	No
is_active	boolean	No	true	No
recovery_token	text	Sí		No
expo_push_token	text	Sí		No
created_at	timestamp with time zone default	No	now()	No
updated_at	timestamp with time zone default	No	now()	No

Realizado por: Bone Edward y Ochoa José, 2022

El diccionario de datos completo se encuentra detallado en el ANEXO D.

3.4.2.7. Diseño de interfaces de usuario

La interfaz de usuario es la responsable de establecer la comunicación entre el usuario y el sistema, por lo cual se deben realizar interfaces de usuario sencillas de usar. Los diseños realizados para la aplicación móvil del conductor fueron aprobados por el Scrum Master y el Product Owner.

En la Figura 3-3, se observa la pantalla de inicio de sesión de la aplicación, realizado en la herramienta de Figma.

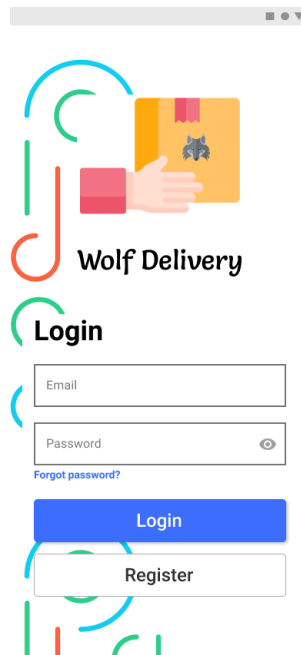


Figura 3-3. Pantalla de inicio de sesión.

Realizado por: Bone Edward y Ochoa José, 2022

En la Figura 4-3, se observa la pantalla de perfil del conductor, realizado en la herramienta de Figma.

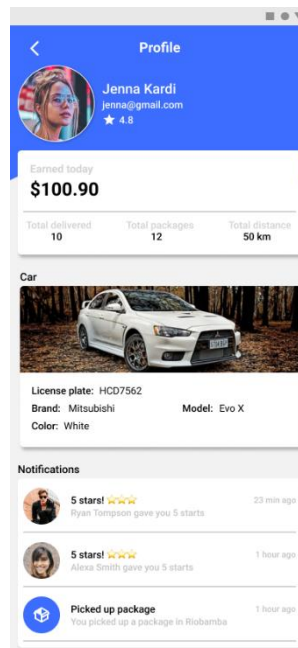


Figura 4-3. Pantalla del perfil del conductor.

Realizado por: Bone Edward y Ochoa José, 2022

El resto de diseños de las interfaces se encuentran en el ANEXO E.

3.4.3. Fase de desarrollo

En esta fase se presentan todas las actividades realizadas durante el desarrollo de la aplicación móvil del conductor como por ejemplo las historias de usuarios y el sprint backlog.

3.4.3.1. Sprint Backlog

Para desarrollar la aplicación móvil del conductor se obtuvieron 6 Sprints, los cuales tendrán máximo 40 puntos estimados, dado que un punto estimado equivale a una hora de trabajo, se establece que un Sprint se culminará en 2 semanas, donde cada día equivale a 4 horas de trabajo, una semana 20 horas, dando un total de 40 horas por sprint.

En la Tabla 17-3, se visualiza el sprint backlog donde están contenidas las historias usuarias/técnicas asignadas a sus correspondientes sprint.

Tabla 17-3: Sprint Backlog.

Sprint	ID	Historias de usuario/técnicas	Puntos Estimados	Fecha inicio	Fecha Fin	Puntos totales
1	HT-01	Entrevista con el cliente para la definición de requerimientos.	7	18/10/2021	19/10/2021	40
	HT-02	Diseño de la arquitectura del sistema.	5	19/10/2021	20/10/2021	
	HT-03	Definición de estándar de codificación del sistema.	3	21/10/2021	21/10/2021	
	HT-07	Instalar los entornos de desarrollo (Visual Studio Code).	5	21/10/2021	22/10/2021	
	HT-04	Diseño de las interfaces de usuario.	10	25/10/2021	27/10/2021	
	HT-05	Diseño e implementación de la base de datos.	8	27/10/2021	29/10/2021	
	HT-06	Generar el diccionario datos (base de datos).	2	29/10/2021	29/10/2021	
2	HU-01	Crear cuenta del conductor.	10	01/11/2021	03/11/2021	40
	HU-02	Completar información del conductor.	10	03/11/2021	05/11/2021	
	HU-03	Completar información del vehículo.	10	08/11/2021	10/11/2021	
	HU-04	Iniciar sesión.	5	10/11/2021	11/11/2021	
	HU-05	Permitir ingreso solo a los conductores aprobados.	5	11/11/2021	12/11/2021	
3	HU-06	Recuperar cuenta de conductor.	10	15/11/2021	17/11/2021	40
	HU-07	Actualizar contraseña de la cuenta.	10	17/11/2021	19/11/2021	
	HU-10	Generar notificaciones por eventos.	10	22/11/2021	24/11/2021	
	HU-17	Visualizar datos del conductor y vehículo.	5	24/11/2021	25/11/2021	
	HU-11	Visualizar notificaciones.	5	25/11/2021	26/11/2021	
4	HU-08	Determinar tarifa a cobrar de la ruta.	10	29/11/2021	01/12/2021	40
	HU-09	Seleccionar lugar de destino.	6	01/12/2021	02/12/2021	
	HU-12	Mostrar ruta de recogida del pedido.	7	03/12/2021	06/12/2021	
	HU-13	Mostrar ruta de entrega del pedido.	7	06/12/2021	08/12/2021	
	HU-15	Reportar emergencia.	10	08/12/2021	10/12/2021	

5	HU-16	Nivel de experiencia del conductor.	20	13/12/2021	17/12/2021	40
	HU-14	Mostrar código maestro del casillero.	5	20/12/2021	21/12/2021	
	HU-19	Cancelar pedidos.	5	21/12/2021	22/12/2021	
	HU-18	Visualizar estadísticas.	10	22/12/2021	24/12/2021	
6	HT-08	Despliegue del backend en el servidor web (Digital Ocean).	40	03/01/2022	14/01/2022	40

Realizado por: Bone Edward y Ochoa José, 2022

3.4.3.2. Historia de Usuario

Las historias de usuario se utilizan generalmente durante la metodología SCRUM, en estas se describen los requerimientos de la aplicación a desarrollar. Se recomienda escribirlas en un lenguaje no técnico, es decir, sin usar palabras técnicas; explicando el rol, la funcionalidad y el resultado esperado. En la Tabla 18-3, se muestra una historia de usuario con sus respectivos campos que componen la tabla perteneciente al Sprint 1.

Tabla 18-3: Historia de usuario 1.

Historia de Usuario 1			
ID: HU-01	Nombre de la historia: Crear una cuenta		
Usuario: Conductor	Sprint: 2		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 01/11/2021	Fecha Fin: 03/11/2021		
Descripción: Como conductor quiero crearme una cuenta para poder realizar el transporte de paquetería.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-01 PA-01	Ingresar datos incorrectos y mostrar mensaje de error (validar datos).	Exitoso	José Ochoa
HU-01 PA-02	Ingresar datos duplicados de otro usuario ya registrado y mostrar mensaje de que ya ha sido ingresado.	Exitoso	José Ochoa
HU-01 PA-03	No permitir el ingreso de campos vacíos.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-01 TI-01	Crear la interfaz de usuario para el registro de cuenta del conductor.	4	
HU-01 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2	

HU-01 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para la creación de cuenta del conductor.	3
HU-01 TI-04	Conectar la interfaz de usuario con el backend.	1

Realizado por: Bone Edward y Ochoa José, 2022

El resto de las historias de usuario se encuentran detalladas en el ANEXO F.

CAPÍTULO IV

4. MARCO DE RESULTADOS

En el presente capítulo se detallan los resultados obtenidos al haber finalizado el desarrollo de la aplicación móvil del conductor para la empresa “Corporación Wolf S.A.” ubicado en la ciudad de Quito de la provincia de Pichincha.

Para la evaluación de la facilidad de modificación y eficiencia de desempeño en la aplicación móvil del conductor y backend se usó el estándar ISO/IEC 25010, según este estándar la facilidad de modificación es una subcaracterística de la mantenibilidad. Por otro lado, basado en el mismo estándar la eficiencia de desempeño se evaluó con la subcaracterística de utilización de recursos (memoria RAM y CPU). Se evaluó estos atributos de calidad del software con el objetivo de determinar qué tan fácil es realizar modificaciones al código y que tan eficiente es la aplicación móvil del conductor y el backend.

4.1. Criterios de evaluación

Para poder evaluar la eficiencia de desempeño y la facilidad de modificación de la aplicación móvil y el backend, se establecieron las siguientes características e indicadores que se visualiza en la Tabla 1-4.

Tabla 1-4: Criterios de evaluación.

Variable	Indicador	Propósito	Tipo de Análisis	Estrategia
Facilidad de modificación	Duplicaciones Mala Practica Redundancia Diseño	Medir la facilidad de modificación	Descriptivo	Utilizar la herramienta SonarQube para obtener el valor de los indicadores y ponderar la facilidad de modificación del sistema.

Eficiencia de desempeño (Utilización de recursos)	Utilización de la Memoria	Medir el consumo de la memoria mientras se utiliza el sistema	Descriptivo	Estudiar cada proceso realizando al menos 10 medidas de cada uno, para poder ponderar el uso de recursos del Sistema.
	Utilización del CPU	Medir el consumo del CPU mientras se utiliza el sistema.		

Fuente: (Pardo 2018; ISO/IEC 2011).

Realizado por: Bone Edward y Ochoa José, 2022

4.2. Facilidad de modificación

Para obtener la facilidad de modificación se utilizó la herramienta SonarQube con el fin de obtener los valores de mala práctica, redundante, diseño y duplicaciones. Después de obtener los valores de los indicadores, se realizó un pequeño ajuste para su procesamiento: mala práctica, redundante y diseño se dividió para las líneas de código y para el indicador de duplicaciones se lo dividió para 100 con el objetivo de obtener un porcentaje del indicador.

Según (Pardo 2018, p. 95), para realizar la ponderación de la facilidad de modificación se debe aplicar la siguiente fórmula:

$$\text{Facilidad de modificación} = \frac{MP + R + D + DP}{4}$$

Donde:

MP = Mala practica

R = Redundante

D = Diseño

DP = Duplicaciones

4.2.1. Obtención de datos con la herramienta SonarQube

En la presente sección se detalla el procedimiento realizado para la obtención de los indicadores de la facilidad de modificación a través del análisis estático del código. A continuación, se detalla el proceso realizado:

- Creación del proyecto en SonarQube: se debe crear un proyecto en la herramienta, después se selecciona el lenguaje de programación del proyecto y el sistema operativo de la computadora donde se está ejecutando la herramienta. En la Figura 1-4, se visualiza las configuraciones iniciales que se deben realizar cuando se crea el proyecto en la herramienta.

2 Run analysis on your project

What option best describes your build?

Maven	Gradle	.NET	Other (for JS, TS, Go, Python, PHP, ...)
-------	--------	------	--

What is your OS?

Linux	Windows	macOS
-------	---------	-------

Figura 1-4. Configuraciones iniciales del proyecto.

Realizado por: Bone Edward y Ochoa José, 2022

- Análisis estático del código: para realizar el análisis estático del código se debe abrir una terminal en el directorio raíz del proyecto, después se ejecuta el comando para iniciar el análisis del código. En la Figura 2-4, se observa la ejecución del análisis del código del proyecto.

```
PS C:\Users\PcParts\Documents\src-modules-backend>
PS C:\Users\PcParts\Documents\src-modules-backend> sonar-scanner.bat -D"sonar.projectKey=wolf-server" -D"sonar.sources=src" -D"sonar.host.url=http://localhost:9080" -D"sonar.login=17d31229959c70f85f7028089e8d6c83a1f147f0"
INFO: Scanner configuration file: C:\Users\PcParts\Downloads\sonar-scanner-cli-4.6.2.2472-windows\sonar-scanner-4.6.2.2472-windows\bin\..conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: SonarScanner 4.6.2.2472
INFO: Java 11.0.11 AdoptOpenJDK (64-bit)
INFO: Windows 10 10.0 amd64
INFO: User cache: C:\Users\PcParts\.sonar\cache
INFO: Scanner configuration file: C:\Users\PcParts\Downloads\sonar-scanner-cli-4.6.2.2472-windows\sonar-scanner-4.6.2.2472-windows\bin\..conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: Analyzing on SonarQube server 9.2.4
INFO: Default locale: "es_EC", source code encoding: "windows-1252" (analysis is platform dependent)
INFO: Load global settings
INFO: Load global settings (done) | time=159ms
INFO: Server id: BF41A1F2-AX6NZRL3_MgJEH5A9Mxd
INFO: User cache: C:\Users\PcParts\.sonar\cache
INFO: Load/download plugins
INFO: Load plugins index
INFO: Load plugins index (done) | time=114ms
INFO: Load/download plugins (done) | time=292ms
INFO: Process project properties
INFO: Process project properties (done) | time=22ms
INFO: Execute project builders
INFO: Execute project builders (done) | time=2ms
INFO: Project key: wolf-server
INFO: Base dir: C:\Users\PcParts\Documents\src-modules-backend
INFO: Working dir: C:\Users\PcParts\Documents\src-modules-backend\scannerwork
INFO: Load project settings for component key: 'wolf-server'
INFO: Load project settings for component key: 'wolf-server' (done) | time=49ms
INFO: Load quality profiles
INFO: Load quality profiles (done) | time=222ms
INFO: Load active rules
INFO: Load active rules (done) | time=4449ms
WARN: SCM provider autodetection failed. Please use "sonar.scm.provider" to define SCM of your project, or disable the SCM Sensor in the project settings.
INFO: Indexing files...
```

Figura 2-4. Ejecución del análisis estático del código.

Realizado por: Bone Edward y Ochoa José, 2022

- Visualizar los resultados del análisis estático: una vez que la herramienta termina el análisis se puede observar los resultados obtenidos, en la pestaña de Issues se tiene una subdivisión llamada Tag donde se agrupan los valores de los indicadores. En la Figura 3-4, se observa los resultados obtenidos del análisis estático del código.

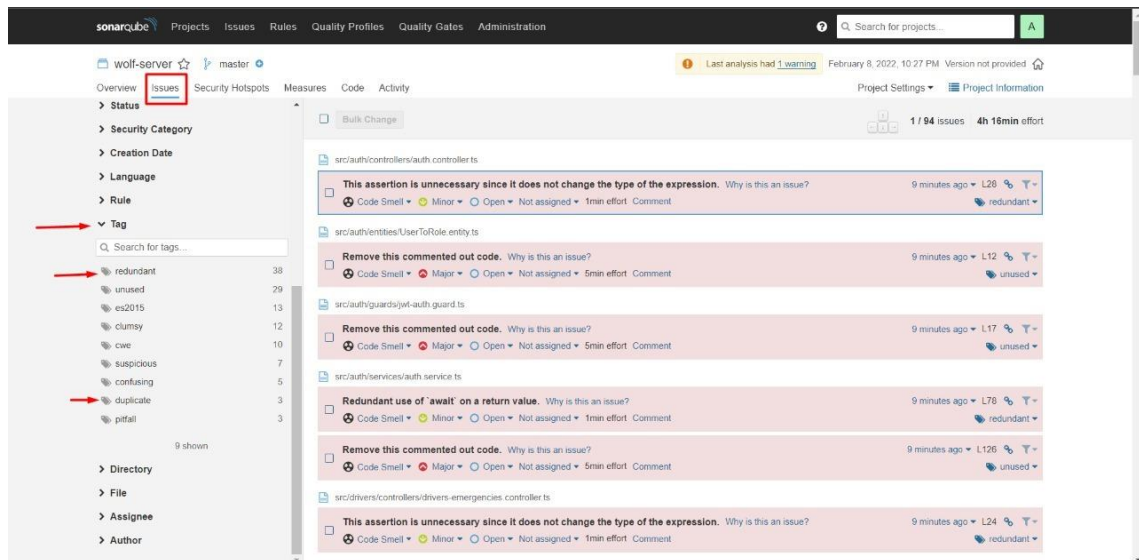


Figura 3-4. Resultados del análisis estático.

Realizado por: Bone Edward y Ochoa José, 2022

4.2.2. Datos generales del código

La herramienta SonarQube realiza un análisis estático del código a través del cual se obtienen diferentes medidas. En la Tabla 2-4 y en la Tabla 3-4, se presentan los datos generales del código del backend y aplicación móvil respectivamente, estos datos se obtuvieron después del análisis estático de la herramienta.

Tabla 2-4: Datos generales del código del backend.

Característica	Valor
Líneas de código	4807
Líneas	5967
Declaraciones	932
Funciones	330
Clases	93
Archivos	91
Líneas de comentarios	503
Comentarios (%)	9.5

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

Tabla 3-4: Datos generales del código de la aplicación móvil.

Característica	Valor
Líneas de código	9117
Líneas	10118
Declaraciones	1622
Funciones	425
Clases	23
Archivos	108
Líneas de comentarios	187
Comentarios (%)	2

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

4.2.3. Datos de la facilidad de modificación en el backend

Para obtener el valor de la facilidad de modificación del backend primero se calculó en cada módulo de forma individual, para después realizar una ponderación general de la facilidad de modificación del backend. A continuación, se detallan los datos de cada módulo:

- Módulo Drivers: este módulo tiene un total de 972 de líneas de código.

En la Tabla 4-4, se muestran los valores de los indicadores de calidad del módulo Drivers con respecto a la capacidad de ser modificado.

Tabla 4-4: Resultados de la facilidad de modificación del módulo Drivers.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	10	0.0103	1.03
Diseño	0	0	0
Duplicaciones	2	0.02	2
Cálculo de la variable en el módulo Drivers			
Suma total		0.0303	3.03
Aplicando la fórmula		0.0076	0.76
Valor de facilidad de modificación		0.9924	99.24

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Módulo Auth: este módulo tiene un total de 423 de líneas de código.

En la Tabla 5-4, se muestran los valores de los indicadores de calidad del módulo Auth con respecto a la capacidad de ser modificado.

Tabla 5-4: Resultados de la facilidad de modificación del módulo Auth.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	2	0.0047	0.47
Diseño	0	0	0
Duplicaciones	0	0	0
Cálculo de la variable en el módulo Auth			
Suma total		0.0047	0.47
Aplicando la fórmula		0.0012	0.12
Valor de facilidad de modificación		0.9988	99.88

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Módulo Orders: este módulo tiene un total de 1820 de líneas de código.

En la Tabla 6-4, se muestran los valores de los indicadores de calidad del módulo Orders con respecto a la capacidad de ser modificado.

Tabla 6-4: Resultados de la facilidad de modificación del módulo Orders.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	14	0.0077	0.77
Diseño	0	0	0
Duplicaciones	1	0.1000	1
Cálculo de la variable en el módulo Orders			
Suma total		0.0177	1.77
Aplicando la fórmula		0.0044	0.44
Valor de facilidad de modificación		0.9956	99.56

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Módulo Notifications: este módulo tiene un total de 134 de líneas de código.

En la Tabla 7-4, se muestran los valores de los indicadores de calidad del módulo Notifications con respecto a la capacidad de ser modificado.

Tabla 7-4: Resultados de la facilidad de modificación del módulo Notifications.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	9	0.0672	6.72
Diseño	0	0	0
Duplicaciones	0	0	0
Cálculo de la variable en el módulo Notifications			
Suma total		0.0672	6.72
Aplicando la fórmula		0.0168	1.68
Valor de facilidad de modificación		0.9832	98.32

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Módulo Users: este módulo tiene un total de 712 de líneas de código.

En la Tabla 8-4, se muestran los valores de los indicadores de calidad del módulo Users con respecto a la capacidad de ser modificado.

Tabla 8-4: Resultados de la facilidad de modificación del módulo Users.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	3	0.0042	0.42
Diseño	0	0	0
Duplicaciones	0	0	0
Cálculo de la variable en el módulo Users			
Suma total		0.0042	0.42
Aplicando la fórmula		0.0011	0.11
Valor de facilidad de modificación		0.9989	99.89

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Módulo Files: este módulo tiene un total de 144 de líneas de código.

En la Tabla 9-4, se muestran los valores de los indicadores de calidad del módulo Files con respecto a la capacidad de ser modificado.

Tabla 9-4: Resultados de la facilidad de modificación del módulo Files.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	0	0	0
Diseño	0	0	0
Duplicaciones	0	0	0
Cálculo de la variable en el módulo Files			
Suma total		0	0
Aplicando la fórmula		0	0
Valor de facilidad de modificación		1	100

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Módulo Email: este módulo tiene un total de 145 de líneas de código.

En la Tabla 10-4, se muestran los valores de los indicadores de calidad del módulo Email con respecto a la capacidad de ser modificado.

Tabla 10-4: Resultados de la facilidad de modificación del módulo Email.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	0	0	0
Diseño	0	0	0
Duplicaciones	0	0	0
Cálculo de la variable en el módulo Email			
Suma total		0	0
Aplicando la fórmula		0	0
Valor de facilidad de modificación		1	100

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Módulo Database: este módulo tiene un total de 196 de líneas de código.

En la Tabla 11-4, se muestran los valores de los indicadores de calidad del módulo Database con respecto a la capacidad de ser modificado.

Tabla 11-4: Resultados de la facilidad de modificación del módulo Database.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	0	0	0
Diseño	0	0	0
Duplicaciones	0	0	0
Cálculo de la variable en el módulo Database			
Suma total		0	0
Aplicando la fórmula		0	0
Valor de facilidad de modificación		1	100

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 1-4, se comparan el valor de los indicadores obtenidos de cada módulo del backend, obteniendo como valor máximo del indicador de redundante con 6.72% el cual pertenece al módulo Notifications, el valor máximo del indicador de duplicaciones con 2% el cual pertenece al módulo Drivers. Es importante mencionar que se omitieron los indicadores de mala práctica y diseño debido a que tenían un valor de cero en todos los módulos y no aportaban información relevante al gráfico.

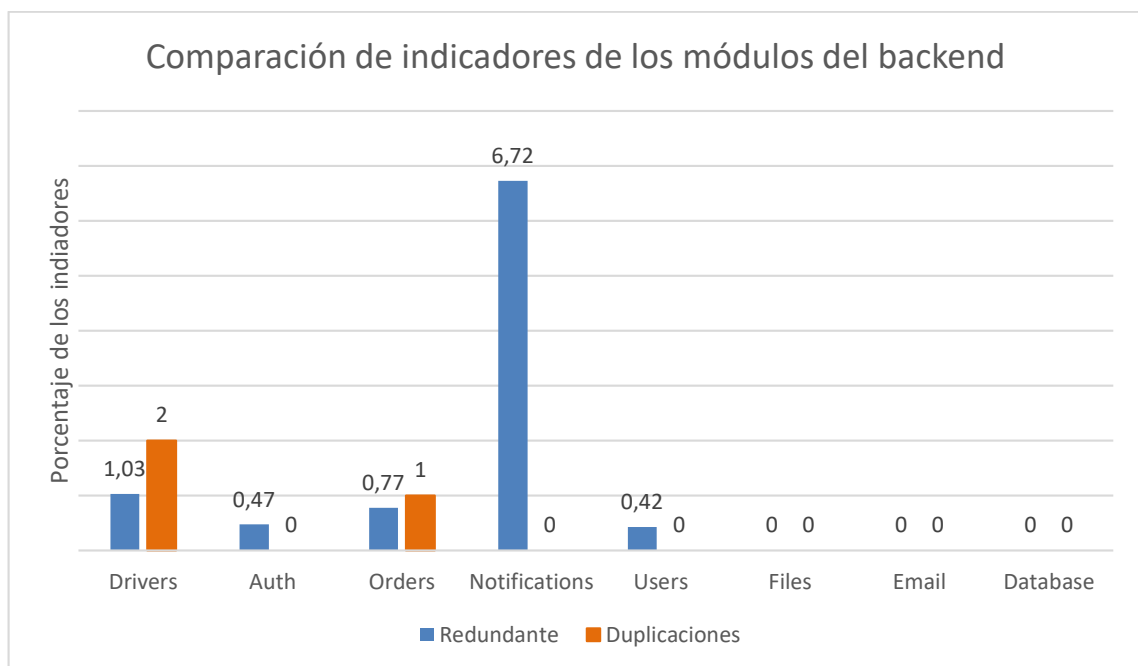


Gráfico 1-4. Comparación de los indicadores de los módulos del backend.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 2-4, se compara el porcentaje de la facilidad de modificación de cada módulo del backend, el porcentaje máximo de la facilidad de modificación fue obtenida por los módulos Users, Files, Emails y Database con un 100%, el valor mínimo obtenido fue de 98.32% perteneciente al módulo Notifications.

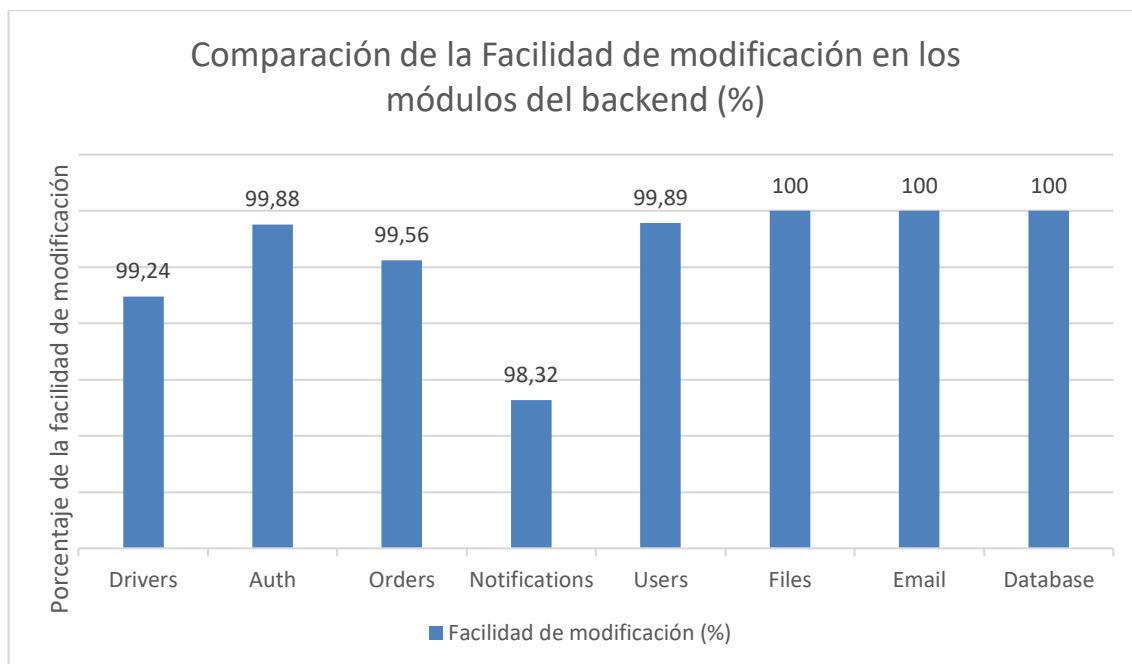


Gráfico 2-4. Comparación de la facilidad de modificación de los módulos del backend.

Realizado por: Bone Edward y Ochoa José, 2022

4.2.4. Datos de la facilidad de modificación en la aplicación móvil

Para obtener el valor de la facilidad de modificación de la aplicación móvil se calculó en base a todo el código. A continuación, se detallan los resultados:

- Aplicación móvil: la aplicación tiene un total de 9117 de líneas de código.

En la Tabla 12-4, se muestran los valores de los indicadores de calidad de la aplicación móvil con respecto a la capacidad de ser modificado.

Tabla 12-4: Resultados de la facilidad de modificación en la aplicación móvil.

Propiedad	Valor de la propiedad	Resultado	%
Mala practica	0	0	0
Redundante	13	0.0014	0.14
Diseño	0	0	0
Duplicaciones	3	0.0300	3

Cálculo de la variable en la aplicación móvil		
Suma total	0.0314	3.14
Aplicando la fórmula	0.0079	0.79
Valor de facilidad de modificación	0.9921	99.21

Fuente: SonarQube, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 3-4, se observa los indicadores obtenidos de la aplicación móvil, siendo el indicador duplicaciones el porcentaje más alto de 3% y el porcentaje más bajo con un 0.14% perteneciente al indicado redundante. Es importante mencionar que se omitieron los indicadores de mala práctica y diseño debido a que tenían un valor de cero en la aplicación móvil y no aportaban información relevante al gráfico.

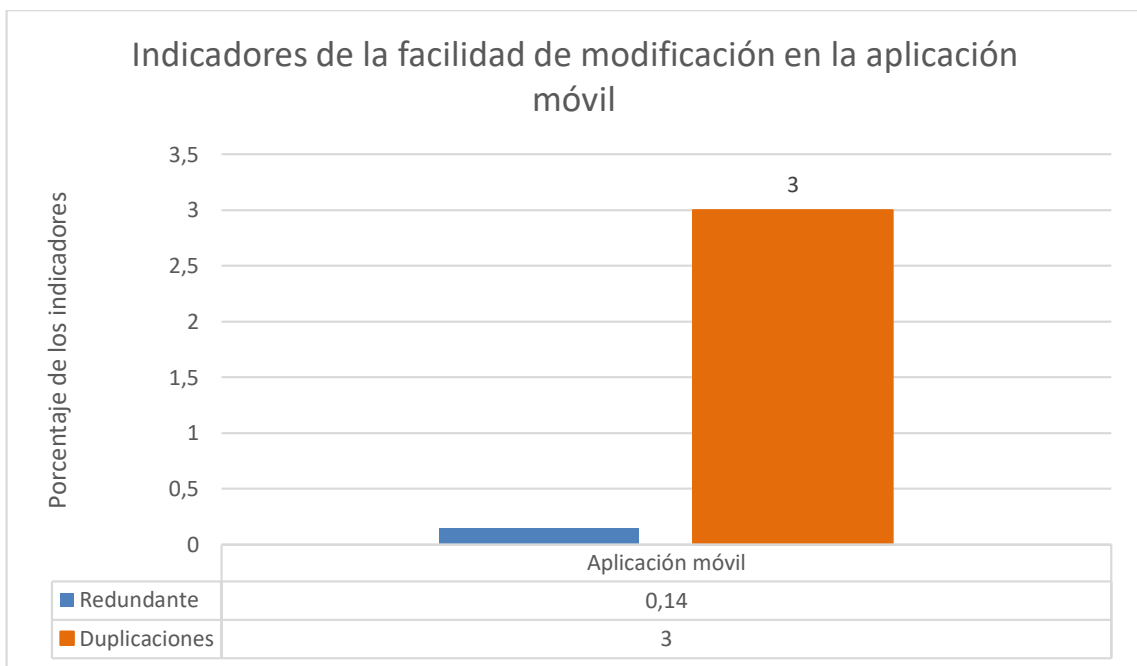


Gráfico 3-4. Indicadores de la facilidad de modificación en la aplicación móvil.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 4-4, se visualiza la comparación entre el backend y la aplicación móvil, donde el backend tiene un porcentaje de facilidad de modificación de 99.61 y la aplicación móvil tiene un porcentaje de facilidad de modificación de 99.21, siendo el backend más fácil de realizar cambios o implementar funciones.

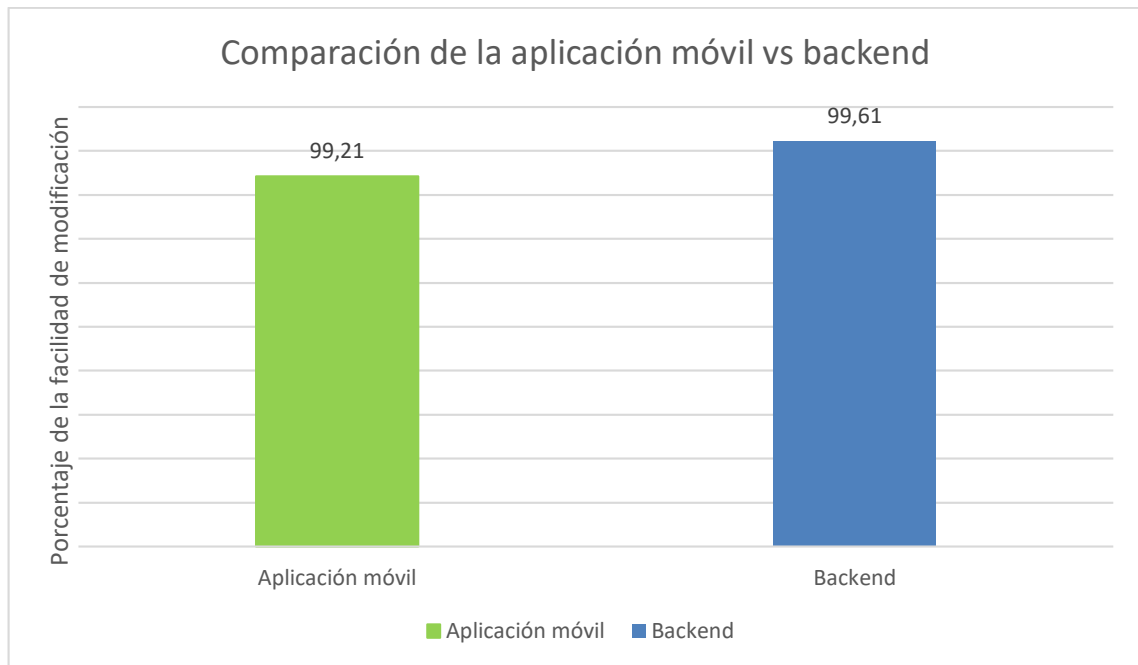


Gráfico 4-4. Comparación de la aplicación móvil vs backend.

Realizado por: Bone Edward y Ochoa José, 2022

4.3. Eficiencia de desempeño

Para obtener la eficiencia de desempeño únicamente se midió la subcaracterística utilización de recursos, el valor de los indicadores perteneciente al backend se determinó con el monitoreo de recursos que ofrece Digital Ocean y para determinar los indicadores de la aplicación móvil se empleó la herramienta Android Profiler.

Para realizar la ponderación de la utilización de recursos (utilización del cpu y memoria) se aplicaron las siguientes formulas:

- Utilización del CPU:

$$X = B - A$$

Donde:

A = La cantidad de tiempo de CPU que realmente es usado para realizar una tarea.

B = Tiempo de operación.

X = Utilización del CPU.

$0 \leq X \leq 1 \text{ min}$ Cuanto más se acerque a 0 es lo mejor. Donde el peor caso es $\geq 1 \text{ min}$.

- Utilización de la memoria:

$$X = \text{Utilización de la memoria.}$$

4.3.1. Obtención de datos de utilización de recursos

En la presente sección se detalla el procedimiento realizado para la obtención de los indicadores de la utilización de recursos, para el backend alojado en Digital Ocean se utilizaron las estadísticas que proporciona el propio administrador de este servicio. A continuación, se detallan el proceso realizado:

- Definir los tiempos de medición: para analizar la utilización de CPU y memoria se determinó el tiempo de inicio y final del proceso a medir. Esto con el fin de poder distinguir el consumo de recursos en las estadísticas proporcionados por Digital Ocean. En la Tabla 13-4, se visualiza el promedio de los tiempos de inicio y fin correspondiente a cada proceso.

Tabla 13-4: Tiempos promedios de los procesos registrar conductor y repartir pedidos.

Proceso	Tiempo promedio [s]
Registrar conductor	173.8
Repartir pedidos	188.8

Fuente: Digital Ocean, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Analizar el consumo de recurso: para medir la utilización de CPU y Memoria se tomó como guía las estadísticas proporcionadas por Digital Ocean durante el tiempo de ejecución de cada proceso. En el Gráfico 5-4 y en el Gráfico 6-4, se visualiza las estadísticas de la utilización de CPU y memoria de un proceso en un determinado tiempo respectivamente.

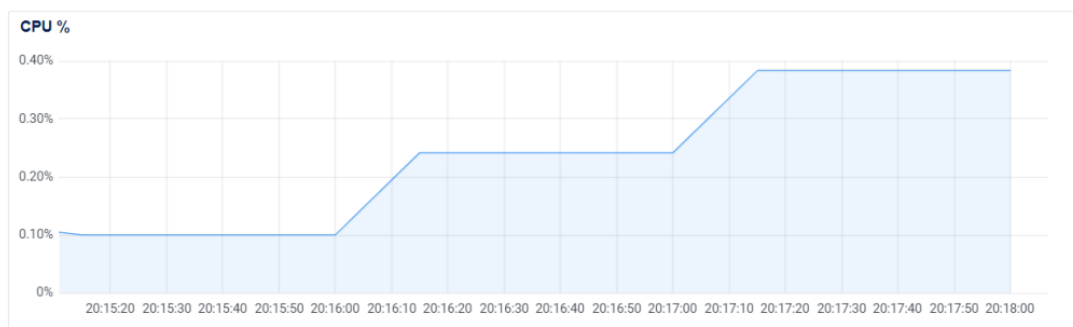


Gráfico 5-4. Estadísticas de la utilización del CPU.

Realizado por: Bone Edward y Ochoa José, 2022

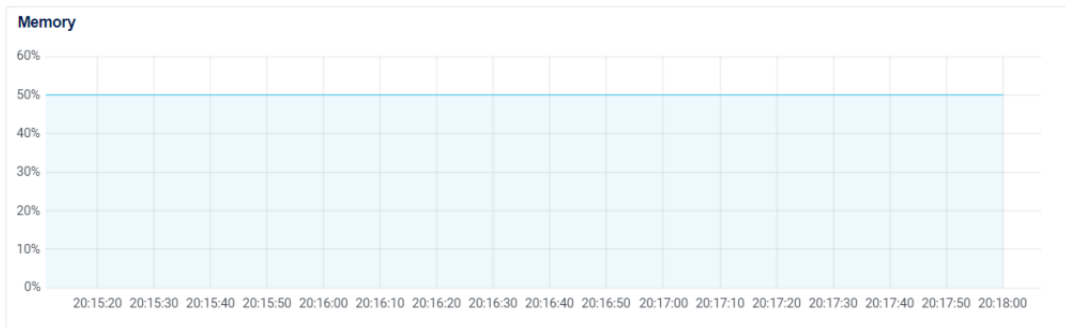


Gráfico 6-4. Estadísticas de la utilización de la memoria.

Realizado por: Bone Edward y Ochoa José, 2022

Para los indicadores de la utilización de recursos en la aplicación móvil se utilizó la herramienta Android Profiler. A continuación, se detallan el proceso realizado:

- **Abrir el Android Profiler:** se utilizó la herramienta Android Profiler de manera independiente, es decir sin ejecutar Android Studio. En la Figura 4-4, se visualiza la herramienta Android Profiler.

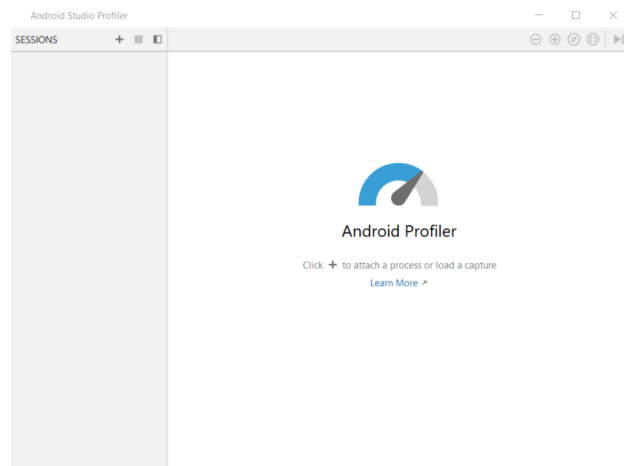


Figura 4-4. Android Profiler.

Realizado por: Bone Edward y Ochoa José, 2022

- **Empezar el registro del proceso:** se seleccionó el dispositivo donde se está ejecutando la aplicación para iniciar el registro de la utilización de recursos. En la Figura 5-4, se visualiza la selección del dispositivo y el nombre del proceso de la aplicación en el dispositivo móvil.

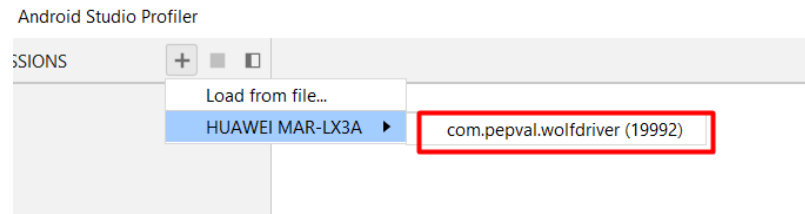


Figura 5-4. Selección del dispositivo y el proceso en Android Profiler.

Realizado por: Bone Edward y Ochoa José, 2022

- Analizar el consumo de recurso: para medir la utilización de CPU y Memoria se analizaron las estadísticas obtenidas del dispositivo durante la ejecución de cada proceso. En el Gráfico 7-4, se visualiza las estadísticas que proporciona la herramienta Android Profiler.

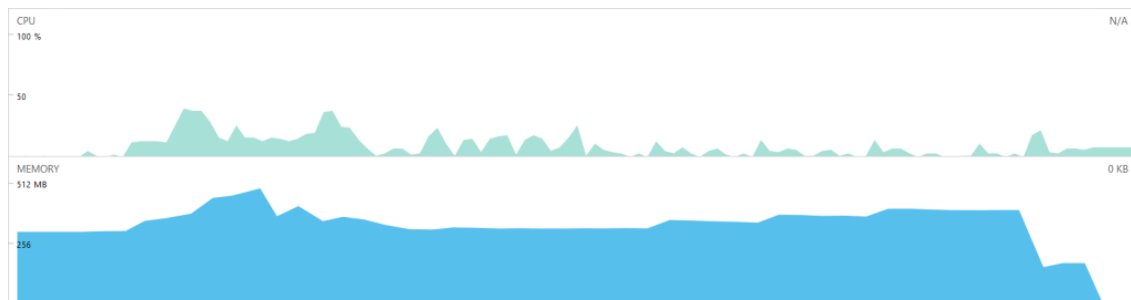


Gráfico 7-4. Estadísticas de utilización de CPU y Memoria.

Realizado por: Bone Edward y Ochoa José, 2022

4.3.2. Datos de la utilización de recursos en el backend

Para obtener el valor de la utilización de recursos del backend se realizó un total de 10 pruebas por cada proceso, con el fin de tener un valor más preciso de cada métrica. A continuación, se detallan los resultados de las pruebas realizadas:

- Pruebas del proceso registrar conductor: en la Tabla 14-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización del CPU.

Tabla 14-4: Pruebas de la utilización de CPU del proceso registrar conductor.

Num. Prueba	B [s]	A [s]	X [s] = B - A
1	180	120	60
2	178	133	45
3	175	175	0
4	180	135	45
5	170	170	0
6	170	170	0

7	180	135	45
8	165	120	45
9	170	170	0
10	170	125	45
Promedio			28.5

Fuente: Digital Ocean, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 15-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización de la memoria del proceso registrar conductor.

Tabla 15-4: Pruebas de la utilización de Memoria del proceso registrar conductor.

Num. Prueba	X [MB]
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
Promedio	0

Fuente: Digital Ocean, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Pruebas del proceso repartir pedidos: en la Tabla 16-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización del CPU.

Tabla 16-4: Pruebas de la utilización de CPU del proceso repartir pedidos.

Num. Prueba	B [s]	A [s]	X [s] = B - A
1	196	181	15
2	190	145	45
3	185	185	0
4	190	190	0
5	192	192	0
6	190	190	0
7	184	169	15
8	185	170	15
9	191	146	45
10	185	140	45
Promedio			18

Fuente: Digital Ocean, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 17-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización de la memoria del proceso repartir pedidos.

Tabla 17-4: Pruebas de la utilización de la Memoria del proceso repartir pedidos.

Num. Prueba	X [MB]
1	5.12
2	0.00
3	0.00
4	0.00
5	5.12
6	5.12
7	0.00
8	0.00
9	5.12
10	5.12
Promedio	2.56

Fuente: Digital Ocean, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 18-4, se visualiza las ponderaciones de la utilización de CPU y memoria de los procesos registrar conductor y repartir pedidos.

Tabla 18-4: Ponderaciones de la utilización de CPU y Memoria del backend.

Variable	Registrar conductor	Repartir pedidos	Promedio
Utilización del CPU [s]	28.5	18	23.25
Utilización de la memoria [MB]	0	2.56	1.28

Fuente: Digital Ocean, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 8-4, se visualiza la utilización de la CPU en el backend de los procesos registrar conductor y repartir pedidos.

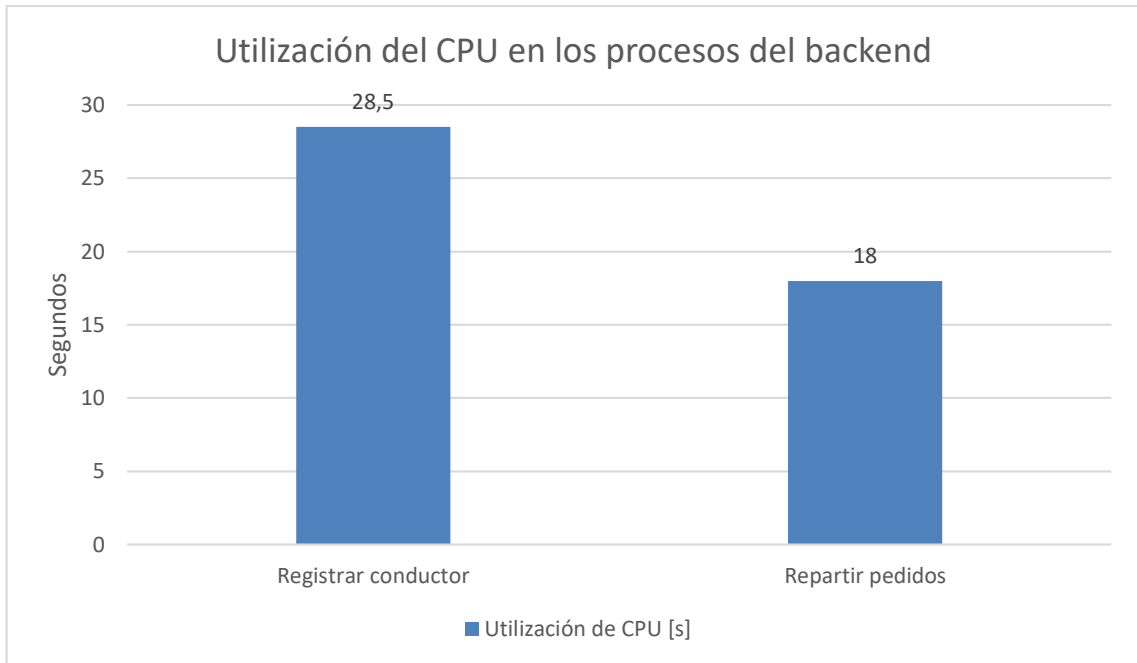


Gráfico 8-4. Utilización de la CPU de los procesos registrar conductor y repartir pedidos.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 9-4, se visualiza la utilización de la Memoria en el backend de los procesos registrar conductor y repartir pedidos.

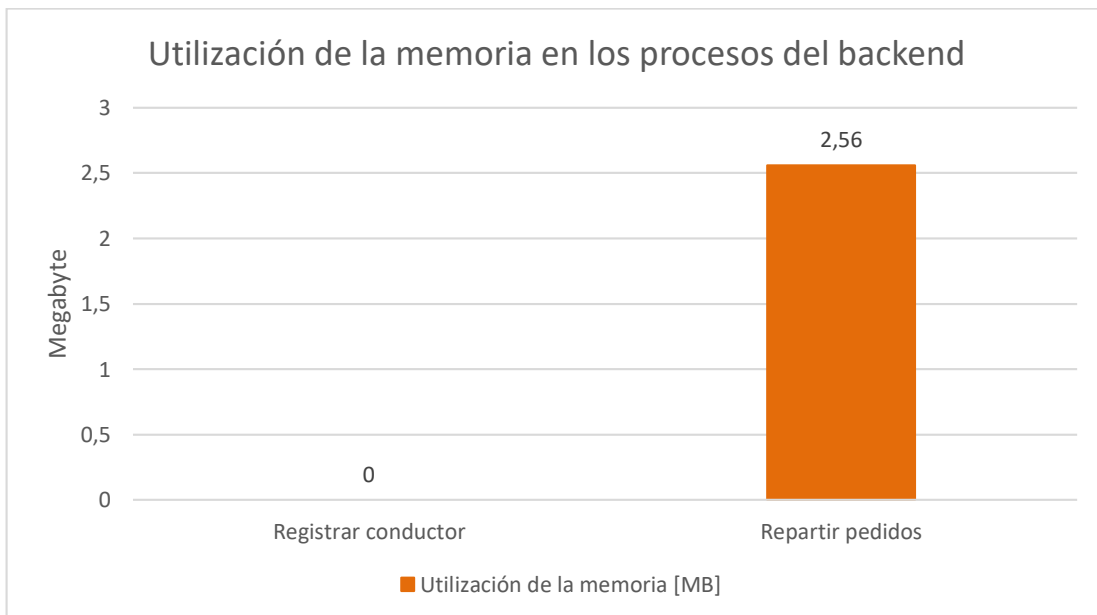


Gráfico 9-4. Utilización de la Memoria de los procesos en el backend.

Realizado por: Bone Edward y Ochoa José, 2022

4.3.3. Datos de la utilización de recursos en la aplicación móvil

Para obtener el valor de la utilización de recursos de la aplicación móvil se realizó un total de 10 pruebas por cada proceso, con el fin de tener un valor más preciso de cada métrica. A continuación, se detallan los resultados de las pruebas realizadas:

- Pruebas del proceso registrar conductor: en la Tabla 19-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización del CPU.

Tabla 19-4: Pruebas de la utilización del CPU del proceso registrar conductor.

Num. Prueba	B [s]	A [s]	X [s] = B - A
1	180	170	10
2	178	169	9
3	175	170	5
4	180	175	5
5	170	164	6
6	170	163	7
7	180	173	7
8	165	159	6
9	170	163	7
10	170	163	7
Promedio			6.9

Fuente: Android Profiler, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 20-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización de la memoria del proceso registrar conductor.

Tabla 20-4: Pruebas de la utilización de la Memoria del proceso registrar conductor.

Num. Prueba	X [MB]
1	362.45
2	270.98
3	279.71
4	364.41
5	358.53
6	329.41
7	337.64
8	353.92
9	349.70
10	321.46
Promedio	332.82

Fuente: Android Profiler, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

- Pruebas del proceso repartir pedidos: en la Tabla 21-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización del CPU.

Tabla 21-4: Pruebas de la utilización de la CPU del proceso repartir pedidos.

Num. Prueba	B [s]	A [s]	X [s] = B - A
1	196	192	4
2	190	185	5
3	185	181	4
4	190	188	2
5	192	191	1
6	190	187	3
7	184	183	1
8	185	184	1
9	191	190	1
10	185	183	2
Promedio			2.4

Fuente: Android Profiler, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 22-4, se visualizan los detalles de las 10 pruebas realizadas con respecto a la utilización de la memoria del proceso repartir pedidos.

Tabla 22-4: Pruebas de la utilización de la Memoria del proceso repartir pedidos.

Num. Prueba	X [MB]
1	350.65
2	323.61
3	325.45
4	373.61
5	331.18
6	329.41
7	363.31
8	318.01
9	380.38
10	339.11
Promedio	343.47

Fuente: Android Profiler, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 23-4, se visualiza las ponderaciones del uso de CPU y memoria de los procesos registrar conductor y repartir pedidos de la aplicación móvil.

Tabla 23-4: Ponderaciones de la utilización de CPU y Memoria de la aplicación móvil.

Variable	Registrar conductor	Repartir pedidos	Promedio
Utilización del CPU [s]	6.9	2.4	4.65
Utilización de la memoria [MB]	332.82	343.47	338.14

Fuente: Android Profiler, 2022.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 10-4, se visualiza la utilización de la CPU en la aplicación móvil de los procesos registrar conductor y repartir pedidos.

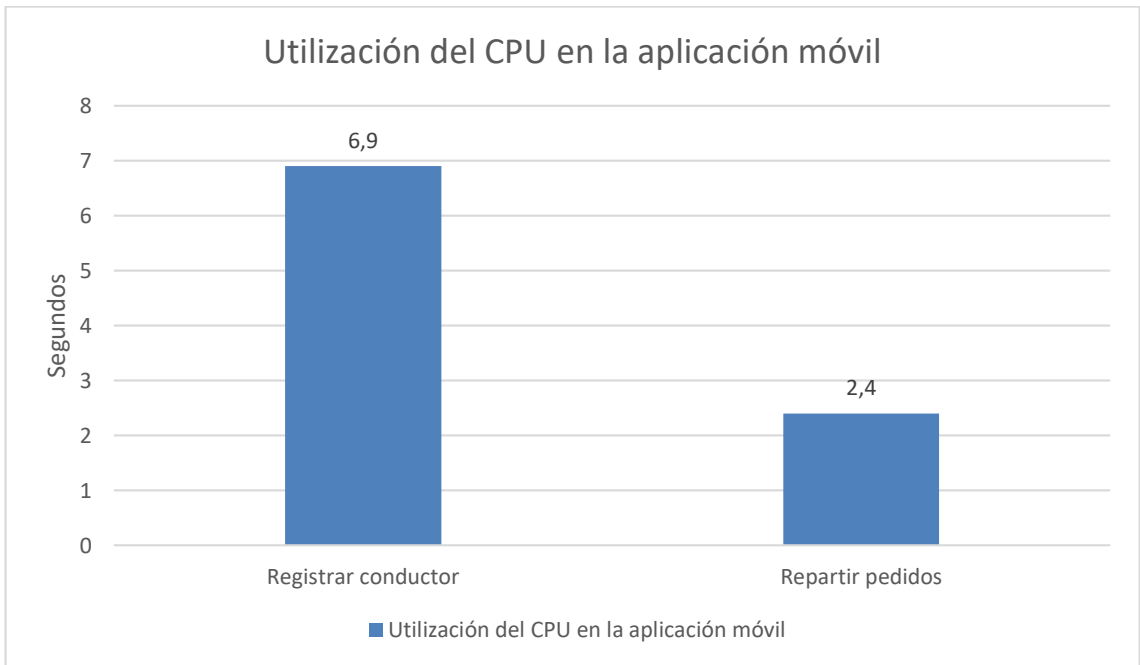


Gráfico 10-4. Utilización del CPU en la aplicación móvil.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 11-4, se visualiza la utilización de la Memoria en la aplicación móvil de los procesos registrar conductor y repartir pedidos.

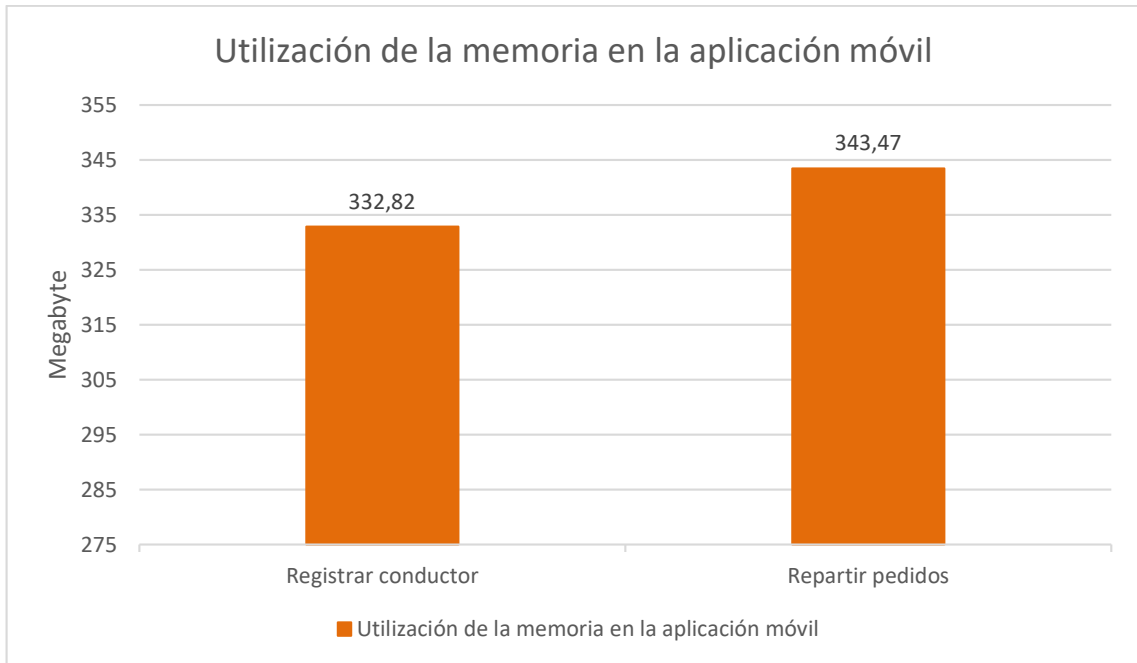


Gráfico 11-4. Utilización de la Memoria en la aplicación móvil.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 12-4, se visualiza la comparación de la CPU en el backend y la aplicación móvil.

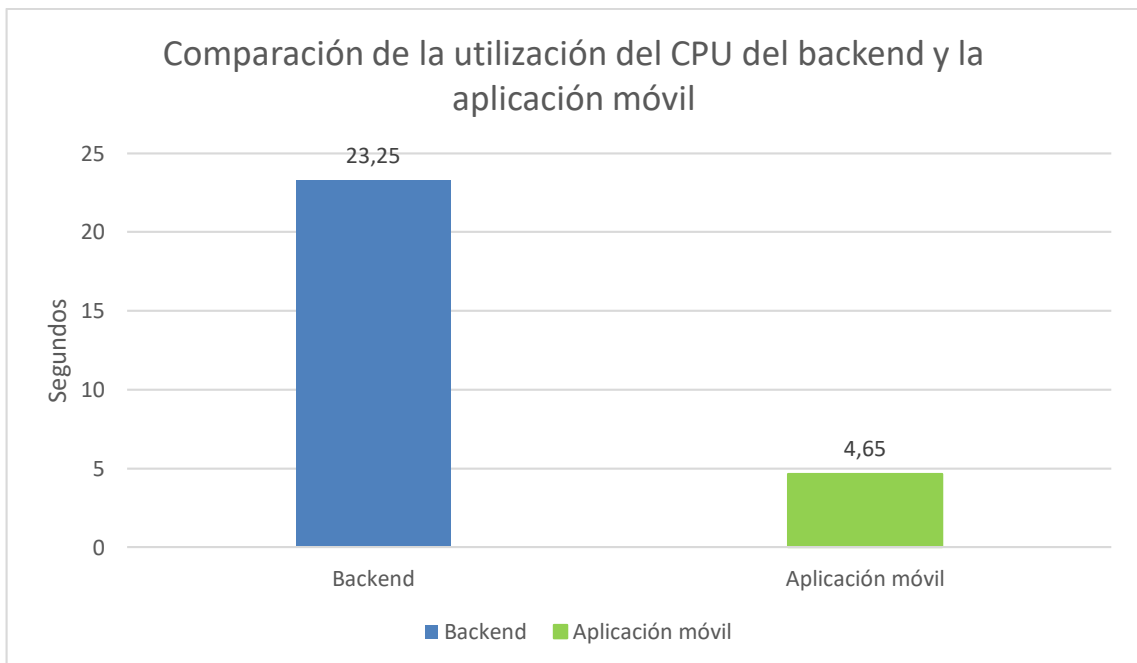


Gráfico 12-4. Comparación de la utilización del CPU del backend y la aplicación móvil.

Realizado por: Bone Edward y Ochoa José, 2022

En el Gráfico 13-4, se visualiza la comparación de la Memoria en el backend y la aplicación móvil donde se puede evidenciar la utilización de memoria en el backend es mínima con respecto a la aplicación móvil.

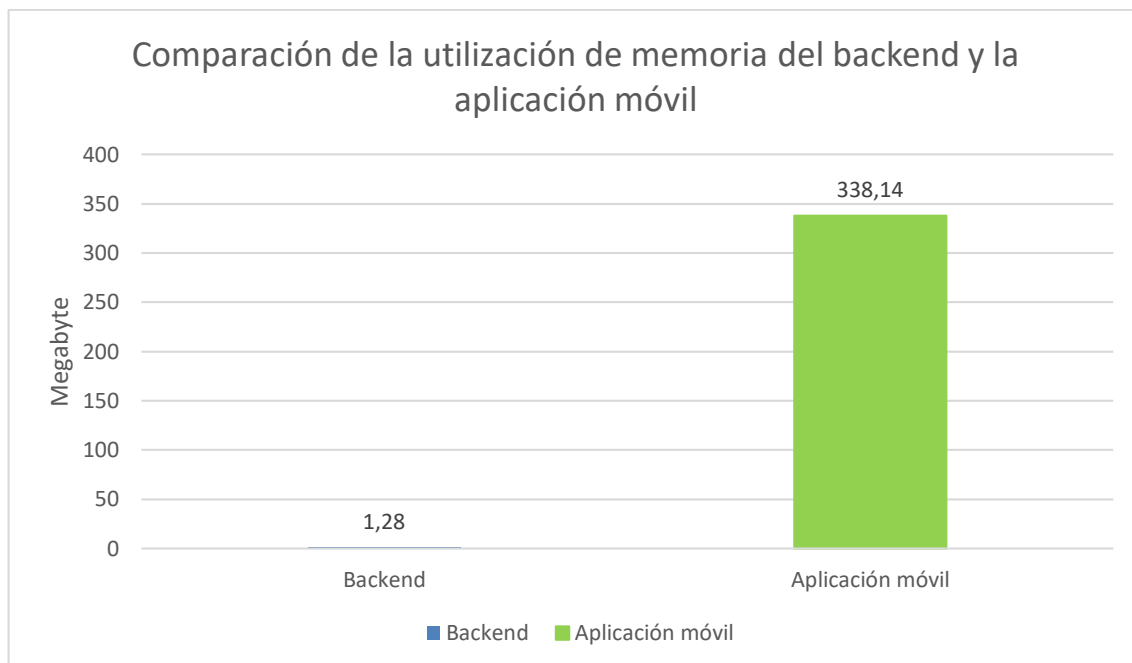


Gráfico 13-4. Comparación de la utilización de memoria del backend y la aplicación móvil.

Realizado por: Bone Edward y Ochoa José, 2022

4.4. Análisis de resultados de la evaluación

En la presente sección se analiza los resultados obtenidos de la evaluación de la facilidad de modificación y eficiencia de desempeño tanto para la aplicación móvil como para el backend.

4.4.1. *Facilidad de modificación*

Para evaluar la facilidad de modificación obtenida del backend y la aplicación móvil, se utilizó una valoración con la cual se determina el nivel de facilidad de modificación. En la Tabla 24-4, se visualiza la valoración para la facilidad de modificación.

Tabla 24-4: Valoración para la facilidad de modificación.

Valor de la facilidad de modificación	Interpretación
1 – 25%	Muy malo
26 – 50%	Malo
51 – 75%	Bueno
76 – 100%	Muy Bueno

Fuente: (Grefa Cerda 2019)

Realizado por: Bone Edward y Ochoa José, 2022

Como se puede observar en el Gráfico 4-4, el valor obtenido de la facilidad de modificación en el backend es de 99.61% ubicándose en el rango de 76 - 100% equivalente a Muy Bueno, en cambio el valor de facilidad de modificación obtenido de la aplicación móvil es de 99.21% ubicándose en el rango de 76 - 100% que corresponde a Muy Bueno.

4.4.2. Eficiencia de desempeño

Para determinar la eficiencia de desempeño de la aplicación móvil y backend, se optó por utilizar una valoración con la cual se puede obtener una evaluación del nivel de eficiencia de desempeño que se tiene en el sistema. En la Tabla 25-4, se visualiza la valoración para la utilización de la memoria.

Tabla 25-4: Valoración de la utilización de la memoria.

Memoria	Calificación	Interpretación
0 – 125 MB	100%	Excelente
126 – 300 MB	75%	Muy Bueno
301 – 425 MB	50%	Bueno
426 – 600 MB	25%	Regular
601 – ∞ MB	0%	Malo

Fuente: (Rea Gómez 2019)

Realizado por: Bone Edward y Ochoa José, 2022

Como se puede observar en el Gráfico 13-4, el valor obtenido de la utilización de la memoria en la aplicación móvil es de 338.14 MB ubicándose en el rango de 301 – 425 MB lo que equivale a Bueno, por otra parte, el resultado de la utilización de la memoria en el backend es de 1.28 MB, ubicándose en el rango de 0 – 125 MB, lo que equivale a Excelente. En la Tabla 26-4, se visualiza la valoración del uso de CPU.

Tabla 26-4: Valoración de la utilización del CPU.

CPU [s]	Calificación	Interpretación
0 – 15	100%	Excelente
16 – 30	75%	Muy Bueno
31 – 45	50%	Bueno
46 – 59	25%	Regular
60 – ∞	0%	Malo

Fuente: (Carrion Vaca 2018)

Realizado por: Bone Edward y Ochoa José, 2022

Como se puede observar en el Gráfico 12-4, el valor obtenido del uso de CPU para la aplicación móvil es de 4.65 segundos ubicándose en el rango de 0 – 15 segundos, lo que equivale a Excelente, en cambio, el valor obtenido para el uso de CPU del backend es de 23.25 segundos, ubicándose en el rango de 16 – 30 segundo, lo que equivale a Muy Bueno. En la Tabla 27-4, se visualiza la ponderación de la utilización de recursos en el backend.

Tabla 27-4: Ponderación de la utilización de recursos en el backend.

Variable	Métrica	Ponderación	Calificación obtenida	Calificación ponderada
Utilización de recursos	Utilización de la memoria [MB]	50%	100%	50%
	Utilización del CPU [s]	50%	75%	37.5%
Total		100%	-----	87.5%

Fuente: (Rea Gómez 2019)

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 28-4, se visualiza la ponderación de la utilización de recursos en la aplicación móvil.

Tabla 28-4: Ponderación de la utilización de recursos en la aplicación móvil.

Variable	Métrica	Ponderación	Calificación obtenida	Calificación ponderada
Utilización de recursos	Utilización de la memoria [MB]	50%	50%	25%
	Utilización del CPU [s]	50%	100%	50%
Total		100%	-----	75%

Fuente: (Rea Gómez 2019)

Realizado por: Bone Edward y Ochoa José, 2022

En la Tabla 29-4, se visualiza la valoración para la evaluación de la eficiencia de desempeño.

Tabla 29-4: Valoración de la eficiencia de desempeño.

Calificación	Interpretación
90 – 100%	Excelente
75 – 89%	Muy Bueno
50 – 74%	Bueno
25 – 49%	Regular
0 – 24%	Malo

Fuente: (Rea Gómez 2019)

Realizado por: Bone Edward y Ochoa José, 2022

Con el valor obtenido de 75% en la aplicación móvil, se puede interpretar que esta tiene un nivel de eficiencia de desempeño Muy Bueno, así mismo, con el valor de 87.5% obtenido del backend se interpreta que la eficiencia de desempeño de este es Muy Bueno.

CONCLUSIONES

Al finalizar el presente trabajo de integración curricular y después de haber obtenidos los resultados de la facilidad de modificación y eficiencia de desempeño de la aplicación móvil y backend para la gestión de reparto de paquetería liviana de la empresa “Corporación Wolf S.A.”, se llegaron a las siguientes conclusiones:

- Se analizó el proceso actual de reparto de paquetería liviana de la empresa “Corporación Wolf S.A.” empleando el estándar de BPMN, donde se pudo determinar que delegan la logística de reparto a una empresa de terceros, por lo cual se implementa la aplicación móvil en este trabajo de integración curricular de tal forma que la empresa realice su propia gestión de reparto de paquetería liviana.
- Se definieron los requisitos del sistema a través de una entrevista al representante legal de la empresa y al jefe de departamento de sistemas. Durante el desarrollo de la aplicación móvil y backend se utilizó una matriz de trazabilidad para realizar el seguimiento de los requisitos, es importante recalcar que se empleó una herramienta (Power Designer RQM) para facilitar el seguimiento de los requisitos.
- Para el desarrollo de la aplicación móvil se utilizó el framework de React Native con Expo debido a que facilita el desarrollo de la aplicación y se tiene una misma base de código para diferentes sistemas operativos. En el backend se utilizó el framework de NestJS ya que permite crear aplicaciones escalables y tiene un flujo de trabajo definido con buenas prácticas de desarrollo.
- El desarrollo de la aplicación móvil y backend se realizó utilizando la metodología Scrum. Se planificaron un total de 6 Sprints, cada uno con un total de 40 puntos de esfuerzo. Además, se obtuvieron 8 historias técnicas y 19 historias de usuarios.
- Para evaluar la facilidad de modificación y la eficiencia de desempeño se apoyó en la ISO/IEC 25010, donde en la métrica de facilidad de modificación se utilizó la herramienta SonarQube y para la eficiencia de desempeño se escogió la subcaracterística utilización de recursos con las variables utilización de CPU y para la variable utilización de memoria se utilizó un enfoque propio.

- Luego de haber realizado un análisis estático del código con la herramienta SonarQube y realizado los cálculos respectivos se obtuvo la facilidad de modificación, donde el backend y aplicación móvil consiguieron un valor equivalente a Muy Bueno, con 99.61% y 99.21% respectivamente.
- Después de haber realizado la evaluación de eficiencia de desempeño en la aplicación móvil se obtuvo el valor de 75% que se interpreta como Muy Bueno, con una utilización de Memoria del 338.14 MB (Bueno) y una utilización de CPU de 4.65 segundos (Excelente). Por otro lado, el backend obtuvo el valor de 87.5% que se interpreta como Muy Bueno, con una utilización de Memoria de 1.28 MB (Excelente) y una utilización de CPU de 23.25 segundos (Muy Bueno).

RECOMENDACIONES

- Se recomienda implementar los sistemas complementarios como el administrador web y la aplicación del cliente con el fin de mejorar la interacción de la aplicación móvil de conductor debido a que algunas funcionalidades están relacionadas de manera indirecta.
- Se sugiere utilizar el framework NestJS para futuros trabajos debido a que facilita crear aplicación con buenas prácticas y fáciles de mantener por medio de su flujo de trabajo ya predefinido.
- Se recomienda a la empresa “Corporación Wolf S.A.”, seguir mejorando la aplicación móvil del conductor con la retroalimentación de los usuarios finales para tener una mejor experiencia de usuario.
- Para la implementación de futuras funcionalidades se recomienda utilizar la metodología SCRUM debido a que agiliza la implementación de las nuevas funcionalidades con la ayuda de los stakeholders.

GLOSARIO

API: Una API o interfaz de programación de aplicaciones es un conjunto de reglas y protocolos que se utiliza para diseñar e integrar el software de las aplicaciones (RedHat 2017, pArr. 1).

BPMN: Son las siglas de Business Process Model and Notation (Modelo y Notación de Procesos de Negocio), el cual básicamente es un estándar que proporciona varios diagramas, que son diseñado para que lo usen las personas que diseñan y administran los Procesos de Negocios (Object Management Group 2014, p. 2).

CPU: Unidad de proceso central, se le conoce como procesador o CPU su función es controlar, coordinar y llevar a cabo todas las operaciones del sistema (Villazán 2009, p. 11).

GRAPHQL: Es un lenguaje de consultas y un tiempo de ejecución del servidor para interfaces de programación de aplicaciones; su principal función es permitir que los clientes obtengan los exactamente los datos que solicitan (RedHat 2019, pArr. 1).

ISO/IEC: ISO (Organización Internacional de Normalización) e IEC (Comisión Electrotécnica Internacional) forman el sistema especializado para la normalización mundial.

MB: Son las siglas de Megabyte. Básicamente es la medida de la capacidad de memoria de un ordenador que equivale a 1 millón de bytes (Lexico 2022, pArr. 1).

PMBOOK: Son las siglas de Project Management Body of Knowledge (cuerpo de conocimientos de la gestión de proyectos) y es la Guía de los Fundamentos para la Dirección de Proyectos que contiene una colección completa de procesos, prácticas recomendadas y directrices que se acepta como estándar dentro del sector de la gestión de proyectos (Project Managment Institute 2021, pArr. 1).

RQM: Son las siglas de Requirements Model (Modelo de Requisitos), el cual se usa para crear una jerarquía de requisitos del proyecto los cuales deben cumplirse durante el desarrollo del proyecto (Sybase 2007, pArr. 1).

RTM: Son las siglas de Matriz de Trazabilidad de Requisitos es una tabla que relaciona los requisitos con su origen y permite vincularlos para realizar un seguimiento desde su concepción hasta los entregables de proyecto (Marone 2000, pArr. 3).

SCV: Un Sistema de Control de Versiones, es un sistema que ayuda a gestionar código fuente, esto por lo general se refiere a que permite manejar las diferentes versiones de ficheros fuente. Algunos de los más populares son CVS, SVN y Git (Mozilla Foundation 2021, pArr. 1).

STAKEHOLDERS: Son individuos o empresas que intervienen activamente en el desarrollo de un proyecto, o cuyos intereses pueden verse afectados positiva o negativamente como resultado de la ejecución del proyecto o la finalización exitosa del proyecto (Smith 2000, pArr. 8).

FRAMEWORKS: En el desarrollo de software, un framework es una composición conceptual y tecnológica con un soporte bien definido, habitualmente con módulos de software concretos, en base a la cual otro proyecto de software puede ser fácilmente organizado y desarrollo (Cobo y Manzano 2014, p. 36).

BIBLIOGRAFÍA

APACHE SOFTWARE FOUNDATION, Apache JMeter. [en línea]. 2021. [Consulta: 12 diciembre 2021]. Disponible en: <https://jmeter.apache.org/>.

BODUCH, A., *React and React Native* [en línea]. S.l.: Packt Publishing Ltd. 2017. ISBN 978-1-78646-565-8. Disponible en: https://books.google.com.ec/books?id=jLkrDwAAQBAJ&lpg=PP3&ots=m2-rHtm2L_&dq=978-1-78646-565-8&hl=es&pg=PR4#v=onepage&q=978-1-78646-565-8&f=false.

CARRION VACA, G.E., *COMPARATIVA DE TRES HERRAMIENTAS DE REALIDAD AUMENTADA UTILIZANDO UNA METODOLOGÍA DE MEDICIÓN DE SOFTWARE ISO 25010*. S.l.: Universidad Tecnica del Norte. 2018.

COBO, A. y MANZANO, J., *Análisis comparativo de framework software libre para el desarrollo de aplicaciones de escritorio en Java* [en línea]. S.l.: Escuela Superior Politécnica de Chimborazo. 2014. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/3323>.

DÍAZ, S., Desarrollo de una aplicación móvil para el aprendizaje de una lengua extranjera. [en línea]. S.l.: 2021. [Consulta: 3 julio 2021]. Disponible en: <https://ddd.uab.cat/record/238426>.

ENRIQUEZ, J.G. y CASAS, S.I., USABILIDAD EN APLICACIONES MÓVILES, Vista de Usabilidad en aplicaciones móviles. *Itc* [en línea], pp. 23. 2013. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=5123524>.

ESKOLA, R., *React Native Performance Evaluation* [en línea]. S.l.: Aalto University. 2018. Disponible en: https://aaltodoc.aalto.fi/bitstream/handle/123456789/32475/master_Eskola_Rasmus_2018.pdf?sequence=1&isAllowed=y.

EXELA TECHNOLOGIES INC., Intelligent Lockers. *exelatech.com* [en línea]. 2019. [Consulta: 10 julio 2021]. Disponible en: https://emea.exelatech.com/sites/default/files/exela-solutions-sell/IntelligentLockers_0.pdf.

FARFÁN, J.A. y VEGA, V. zepeda, Software maintainability. Considerations for its specification and validation. *Ingeniare* [en línea], vol. 28, no. 4, pp. 654-667. 2020. [Consulta: 29 junio 2021]. ISSN 07183305. DOI 10.4067/S0718-33052020000400654. Disponible en: http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0718-33052020000400654&lng=es&nrm=iso&tlng=es.

GONZALES ARAKI, A., ALFREDO, R., SIERRA, U., MACHADO, V. y DANIEL CITATION GONZALES ARAKI, W., Red de lockers para entrega y recolección de paquetes, integrando a vendedores y compradores de e-commerce mediante plataforma en línea. *Universidad Peruana de Ciencias Aplicadas (UPC)*. S.l.: Universidad Peruana de Ciencias Aplicadas (UPC). 2018.

GONZALES, R., URBIOLA, W. y VITALE, W., Red de lockers para entrega y recolección de paquetes, integrando a vendedores y compradores de e-commerce mediante plataforma en línea. *Universidad Peruana de Ciencias Aplicadas (UPC)* [en línea]. S.l.: Universidad Peruana de Ciencias Aplicadas (UPC). 2018. [Consulta: 19 junio 2021]. Disponible en: <http://hdl.handle.net/10757/623238>.

GREFA CERDA, J.F., *DESARROLLO DE UN SISTEMA WEB PARA LA RECAUDACIÓN DE LA JUNTA ADMINISTRADORA DE AGUA POTABLE REGIONAL "KAWSAY YAKU" DE LA PARROQUIA*. S.l.: ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO. 2019.

HERNÁNDEZ, B., *Autocheck for Github Classroom* [en línea]. S.l.: Universidad de La Laguna. 2018. [Consulta: 22 mayo 2022]. Disponible en: [https://riull.ull.es/xmlui/bitstream/handle/915/10430/Autocheck for Github Classroom.pdf;jsessionid=5D167141D1D65F7C9E10B4431EA2124A?sequence=1](https://riull.ull.es/xmlui/bitstream/handle/915/10430/Autocheck%20for%20Github%20Classroom.pdf;jsessionid=5D167141D1D65F7C9E10B4431EA2124A?sequence=1).

HIDALGO, L.N. y JIMÉNEZ, M.E., *Estudio Comparativo de los Servicios Web Restfull Jersey y SOAP JAX-WS para el Desarrollo de una Aplicación Android con Wikitude Aplicada a la Gestión de Información Geolocalizada del Turismo de la Provincia de Chimborazo*. [en línea]. S.l.: Escuela Superior Politécnica de Chimborazo. 2016. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/4733>.

IEEE, *IEEE Standard Glossary of Software Engineering Terminology* [en línea]. 1990. S.l.: s.n. 1990. ISBN 1-55937-067-X. Disponible en:

http://www.mit.jyu.fi/ope/kurssit/TIES462/Materiaalit/IEEE_SoftwareEngGlossary.pdf.

ISO/IEC, ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models. [en línea]. 2011. [Consulta: 10 julio 2021]. Disponible en: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>.

ISO/IEC, ISO/IEC 25022:2016 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Measurement of quality in use. [en línea], pp. 41. 2016. Disponible en: <https://www.iso.org/standard/35746.html>.

LEXICO, Definición de Megabyte. [en línea]. 2022. [Consulta: 20 abril 2022]. Disponible en: <https://www.lexico.com/es/definicion/megabyte>.

MAIDA, E. y PACIENZA, J., *Metodologías de desarrollo de software* [en línea]. S.l.: Universidad Católica Argentina. 2015. Disponible en: <http://bibliotecadigital.uca.edu.ar/repositorio/tesis/metodologias-desarrollo-software.pdf>.

MARONE, O., Requirement traceability, a tool for quality results. *Requirement traceability, a tool for quality results* [en línea]. 2000. [Consulta: 20 abril 2022]. Disponible en: <https://www.pmi.org/learning/library/requirement-traceability-tool-quality-results-8873>.

MENA, D.A., *ANÁLISIS DE MANTENIBILIDAD Y PORTABILIDAD DEL FRAMEWORK REACT NATIVE APLICANDO LA NORMA ISO/IEC 25010 MEDIANTE UN CASO DE ESTUDIO EN LA APLICACIÓN DE GESTIÓN DE EVENTOS OAQ*. S.l.: ESCUELA POLITÉCNICA NACIONAL. 2020.

META PLATFORMS INC, Performance Overview. [en línea]. 2021. [Consulta: 11 julio 2021]. Disponible en: <https://reactnative.dev/docs/next/performance>.

MONFERRER, R., *Especificación de Requisitos de Software según el estándar de IEEE 830* [en línea]. Castelló: Universitat Jaume I. 2000. Disponible en: <http://textos.pucp.edu.pe/pdf/3134.pdf>.

MOZILLA FOUNDATION, SCV. [en línea]. 2021. [Consulta: 20 abril 2022]. Disponible en: <https://developer.mozilla.org/es/docs/Glossary/SCM>.

OBJECT MANAGEMENT GROUP, Business process model and notation (BPMN). *The Complete Business Process Handbook: Body of Knowledge from Process Modeling to BPM* [en línea], vol. 1, no. January, pp. 429-453. 2014. DOI 10.1016/B978-0-12-799959-3.00021-5. Disponible en: <https://www.omg.org/spec/BPMN/2.0/>.

OBJECT MANAGEMENT GROUP, BPMN Specification - Business Process Model and Notation. [en línea]. 2021. [Consulta: 1 noviembre 2021]. Disponible en: <https://www.bpmn.org/>.

PARDO, S., *Mantenibilidad De Productos De Software Según El Modelo Square ISO/IEC 25000* [en línea]. S.l.: Universidad Nacional Agraria de la Selva. 2018. Disponible en: <http://repositorio.unas.edu.pe/handle/UNAS/1360>.

POSTGRESQL, What is PostgreSQL? [en línea]. 2021. [Consulta: 11 julio 2021]. Disponible en: <https://www.postgresql.org/about/>.

PROJECT MANAGMENT INSTITUTE, PMBOK Guide | Project Management Institute. [en línea]. 2021. [Consulta: 20 abril 2022]. Disponible en: https://www.pmi.org/pmbok-guide-standards/foundational/pmbok?sc_camp=D750AAC10C2F4378CE6D51F8D987F49D.

REA GÓMEZ, J.I., *“DESARROLLO DE UN SISTEMA WEB PARA LA GENERACION DE HORARIOS ACADEMICOS A TRAVES DE TECNICAS METAHEURISTICAS PARA LA UNIDAD EDUCATIVA DEL MILENIO TARQUI*. S.l.: ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO. 2019.

REAL ACADEMIA ESPAÑOLA, Definición de paquete. [en línea]. 2022. [Consulta: 25 abril 2022]. Disponible en: <https://dle.rae.es/paquete?m=form>.

REDHAT, ¿Qué es una API? [en línea]. 2017. [Consulta: 20 abril 2022]. Disponible en: <https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces>.

REDHAT, ¿Qué es GraphQL? [en línea]. 2019. [Consulta: 20 abril 2022]. Disponible en: <https://www.redhat.com/es/topics/api/what-is-graphql>.

RICOH EUROPE PLC, Casilleros Inteligentes. [en línea], 2019. Disponible en: https://www.ricoh.es/media/Ricoh-Smart-Lockers-All-ES-online_tcm77-36004.pdf.

SAYAGO, J., FLORES, E. y RECALDE, A., Análisis Comparativo entre los Estándares Orientados a Servicios Web SOAP, REST y GraphQL. *Revista Antioqueña de las Ciencias Computacionales y la Ingeniería de Software (RACCIS)* [en línea], vol. 9, no. March 2020, pp. 10-22. 2019. DOI 10.5281/zenodo.3592004. Disponible en: https://www.researchgate.net/publication/339660087_Analisis_Comparativo_entre_los_Estandares_Orientados_a_Servicios_Web_SOAP_REST_y_GRAPHQL.

SCHWABER, K. y SUTHERLAND, J., The Scrum Guide. [en línea], no. November, pp. 133-152. 2020. Disponible en: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf#zoom=100>.

SLACK, ¿Qué es Slack? . [en línea]. 2021. [Consulta: 11 julio 2021]. Disponible en: <https://slack.com/intl/es-ec/help/articles/115004071768-¿Qué-es-Slack->.

SMITH, L.W., Stakeholder analysis. *Stakeholder analysis a pivotal practice of successful projects* [en línea]. 2000. [Consulta: 20 abril 2022]. Disponible en: <https://www.pmi.org/learning/library/stakeholder-analysis-pivotal-practice-projects-8905>.

SOMMERVILLE, I., *Software Engineering* [en línea]. 9 Edicion. Mexico: PEARSON EDUCACIÓN. 2011. ISBN 978-607-32-0603-7. Disponible en: https://sistemamid.com/panel/uploads/biblioteca/2018-06-11_03-37-12144643.pdf.

SYBASE, The PowerDesigner Requirements Model (RQM). [en línea]. 2007. [Consulta: 20 abril 2022]. Disponible en: https://infocenter-archive.sybase.com/help/index.jsp?topic=/com.sybase.stf.powerdesigner.docs_12.5.0/html/neug/neugp239.htm.

TOM OCCHINO, React Native: Bringing modern web techniques to mobile - Facebook Engineering. [en línea]. 2015. [Consulta: 11 julio 2021]. Disponible en: <https://engineering.fb.com/2015/03/26/android/react-native-bringing-modern-web-techniques-to-mobile/>.

TYPESCRIPT, TypeScript: Typed JavaScript at Any Scale. [en línea]. 2021. [Consulta: 11 julio 2021]. Disponible en: <https://www.typescriptlang.org/es/>.

VILLAZÁN, F.J., *Manual de informática I* [en línea]. IFCCA – UM. Mexico: Universidad de

Michoacana. 2009. [Consulta: 20 abril 2022]. Disponible en: <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwi1nPb91aP3AhWpRjABHcJbNAQFnoECAgQAQ&url=https%3A%2F%2Fwww.upg.mx%2Fwp-content%2Fuploads%2F2015%2F10%2FLIBRO-31-Manual-de-Informatica.pdf&usg=AOvVaw2hOAiWLSFXnlNAbcQ3>.

W3C, Web Services Glossary. [en línea]. 2021. [Consulta: 8 noviembre 2021]. Disponible en: <https://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/#webservice>.

WEHBE, D.N., *Análisis de Viabilidad para Contadores de Aforo* [en línea]. S.l.: Escuela Superior de Ingeniería y Tecnología Universidad de la Laguna. 2021. Disponible en: [https://riull.ull.es/xmlui/bitstream/handle/915/24212/Analisis de viabilidad para contadores de aforo..pdf?sequence=1](https://riull.ull.es/xmlui/bitstream/handle/915/24212/Analisis%20de%20viabilidad%20para%20contadores%20de%20aforo.pdf?sequence=1).

ANEXOS

ANEXO A: ENTREVISTA

Tema:

Requisitos de la aplicación móvil del conductor

Objetivo: Recopilar toda la información necesaria de la aplicación móvil del conductor para detallar los requisitos funcionales y no funcionales de la aplicación móvil del conductor.

Dirigida a:

Jefe de departamento de sistemas: Christian Guevara

Representante legal de la empresa: José Villarruel

Entrevista a la Empresa “Corporación Wolf S.A.”

1. ¿Cuántos repositorios en GitHub se necesita para la aplicación móvil del conductor?

Se ha definido 2 Repositorios 1 para la aplicación móvil del conductor y el otro donde va a ir el backend del sistema, tanto la aplicación móvil del conductor como la aplicación del cliente.

2. ¿Cuáles son los procesos que conforman o intervienen en la aplicación móvil del conductor?

En total son 5 procesos que contiene la aplicación móvil del conductor:

- Crear la cuenta de usuario.
- Subida de documentación para ser conductor.
- Recepción de notificaciones de paquetes a recoger.
- Entrega de paquetes
- Visualización del conductor

3. ¿Puede detallar cada proceso de la aplicación móvil del conductor?

Cada proceso se detalla en el presente documento que le voy a pasar, en ese documento se describe cada proceso como una secuencia de pasos:

Al final de la entrevista está el documento

4. ¿La base de datos va a ser relacional?

Si la base de datos será relacional y desarrollada en PostgreSQL

5 ¿Cuáles son las tecnologías que se van a usar en el desarrollo de la aplicación móvil del conductor?

En la parte del frontend: React Native y Expo.

En la parte del backend: PostgreSQL, TypeScript y NestJS.

En el flujo de trabajo: GitHub y Slack.

6. ¿Cada cuánto serán las reuniones para evaluar los avances de la aplicación móvil del conductor?

Las reuniones se darán cada semana los miércoles de 10:00 am a 11:00 am por medio de la herramienta Zoom, pero si necesitan saber detalles más específicos me pueden contactar por slack. Las reuniones se darán siempre y cuando no sean en días festivos.

7. ¿Es necesario implementar las funciones adicionales?

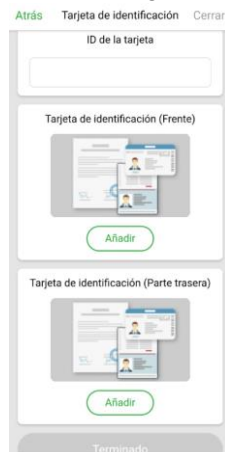
Por el momento solo están planteadas debido a que el producto que vamos a sacar solo es un MBP (Producto mínimo viable), lo que si es necesario es implementar los procesos descritos en la pregunta anterior.

PROCESO DEL CONDUCTOR

1. Crear cuenta de Usuario
 - a. Correo
 - b. Teléfono
 - c. Ciudad
 - d. Nombres completos
 - e. Cédula o DNI
2. Subida de documentación para ser conductor
 - a. Consentimiento legal para uso y revisión de datos proporcionados por el conductor.
 - b. Añadir fotografía



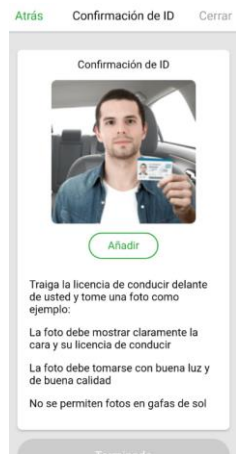
- c. Añadir fotografía de DNI o cédula de identidad



- d. Añadir fotografía de Licencia de conducir



e. Añadir fotografía de confirmación de identidad



f. Añadir fotografía de la matrícula del vehículo

Haz una foto de tu Matrícula del Vehículo

Asegúrate de que la foto contenga toda la información del documento de registro. Recuerda lo siguiente: - No se aceptan vehículos de 2 puertas ni coupés - No se aceptan taxis ni vehículos rentados - No se aceptan fotos de la matrícula



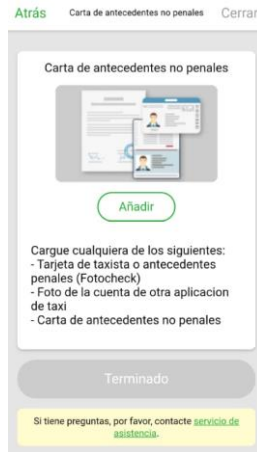
g. Añadir placa del vehículo

h. Añadir marca del vehículo

i. Añadir fotografía del vehículo mostrando la placa



j. Añadir fotografía de antecedentes no penales



3. Recepción de notificaciones de paquetes a recoger de acuerdo:
 - a. Ciudad donde se encuentra el conductor
 - b. Registrar Ciudad a la que se dirige:
 1. Hora de salida
 2. Ciudad a la que se dirige
 - c. Recibe notificaciones de recolección de paquete de un domicilio o lugar específico, de acuerdo con geolocalización cercana al conductor. Si el paquete se encuentra con ciudad de destino igual al conductor se lo lleva directamente y el valor del envío será fijo de acuerdo con la ciudad de destino que vaya. En caso de que el paquete no esté en dirección dejará en un centro de intercambio ubicado en gasolineras que pueden estar dentro de la ciudad, a las afueras de la ciudad o en gasolineras que lleven hacia otras zonas por ejemplo Aloag cuya via interconecta la sierra con la costa, ahí en ese caso podrá dejar en un locker para que otro conductor lo pueda pasar recogiendo y recibir una comisión (porcentaje del valor total de la entrega).
 1. Si los sectores son fuera del centro de la ciudad habrá un recargo por minuto de 10 ctvs (variable) o por km recorrido de 30 ctvs (variable), siempre el menor. Para esto se pondrá un circulo en el mapa y fuera de este se empezará a cobrar este valor.
 2. El conductor el momento de recoger el paquete en un domicilio deberá escanear el código que el sistema le generó al usuario que desea enviar el paquete, de esta manera se sabrá que ya fue recolectado, aquí las dimensiones y peso deberá ser puesto por el usuario (inicialmente no importará si coloca mal el tamaño y el peso el usuario) o arregla con el conductor una tarifa por ejemplo en la aplicación pone las dimensiones más pequeñas pero en realidad pone algo mucho más grande (viveza criolla de cualquier de las partes).
 3. El conductor pegará un código QR o código de barras que serán entregado a él previamente, y escanearla para validar que ese paquete esta con el código que a la persona se le generó.
 4. El usuario ya debería ingresar los datos de envío
 5. Ya le saldrá el valor a cancelar, el conductor podrá cobrar en efectivo o cargar a la tarjeta del cliente
 6. Se cobrará una comisión por la transacción
 - d. Recibe notificaciones de recolección de paquetes de Smart locker para ser llevados a centros de intercambio, de acuerdo con geolocalización cercana al conductor, aquí se le entregará un valor por la carrera.

- e. Recolección de paquetes de centro de intercambio (aparecerá al conductor dependiendo a la ciudad a donde se envié y se le abrirá solo los que van a la ciudad que él se dirige o las aledañas como por ejemplo va de Quito Ambato, pero se debe entregar en Latacunga). Inicialmente se manejará ciudades Quito, Latacunga, Salcedo, Ambato y Riobamba que se encuentran en la misma dirección que no se necesitaría inicialmente los centros de intercambio.
 - f. Le aparecerán alertas en ciudades que estén en el curso del automóvil, para que pueda recoger paquetes (por ejemplo, va de Quito Ambato, pero se debe entregar en Latacunga).
4. Entrega de paquetes
- a. El conductor le escaneará un código QR que se le generará al destinatario, para de esta manera saber que ya se entregó el paquete. El remitente podrá enviarle este código y ahí no ser necesario que se instale nuestra app.
 - b. Enviar un mensaje de texto al momento de leer el código QR, diciendo que el destinatario recibió el paquete.
 - c. Si son varios paquetes al mismo destinatario simplemente permitirá escanear varios en una sola sesión.
5. Visualización del conductor
- a. Guía para llegar a cualquiera de los Smart locker, domicilios a recolectar o centros de intercambio.
 - b. Aparecerá la ciudad a donde quiere enviar el usuario
 - c. Aparecerá un código maestro para retirar los paquetes de los Smart lockers
 - 1. Primero aparecerá la alerta de recolección del Smart locker, el podrá aceptarlo o no.
 - 2. Una vez aceptado tendrá un código de 6 dígitos o un QR para que pueda ser ingresado por teclado o mostrado al lector de códigos QR para activar la apertura de todos los lockers.
 - 3. Se le indicará que lockers estén con paquetes. Tomar en cuenta un volumen aproximado de la cajuela de un carro, en caso de ser mayor el volumen de los paquetes en los Smart lockers, partir en dos vehículos.
 - 4. Por la llevada de cada paquete de acuerdo a los 3 volúmenes de lockers ya se tendrá 3 valores definidos para llevarlos a un centro de intercambio que ya le aparecerá en la pantalla antes de ir a recoger los paquetes.
 - 5. Una vez recogidos los paquetes del Smart locker, el conductor debe cerrarlos todos, en caso de que no los haga tendrá una penalización, por cada locker sin cerrar.
 - 6. Al momento de recoger paquetes del centro de intercambio el conductor escaneará los paquetes que se lleva para que la aplicación le dé la mejor ruta de entrega.

Adicionales

1. Los conductores pueden referir otros conductores y ganar dinero o puntos
2. Por cada 10 envíos el usuario tendrá un envío gratis de hasta máximo el volumen medio de los casilleros
3. Los usuarios podrán poner códigos promocionales para rebajas en sus envíos
4. Los conductores mejores rangkiados serán los que puedan recoger los paquetes de los Smart lockers y llevarlos a centros de intercambios.
5. La plataforma puede permitir cobrar en efectivo, tarjeta de débito o crédito

ANEXO B: BPMN CON EL SISTEMA IMPLEMENTADO

- Proceso de reparto de paquetería de la empresa Corporación Wolf S.A. con el sistema.

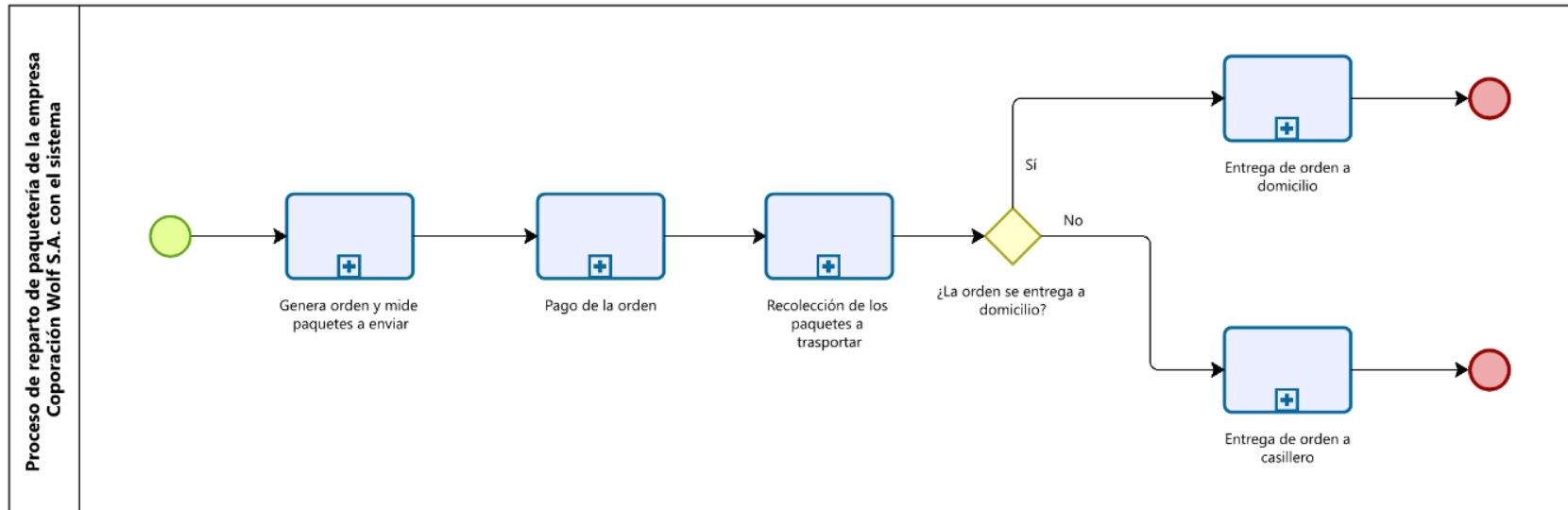


Figura 1-0. Diagrama BPMN con el sistema implementado

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Subproceso:** Generar orden y medir paquete

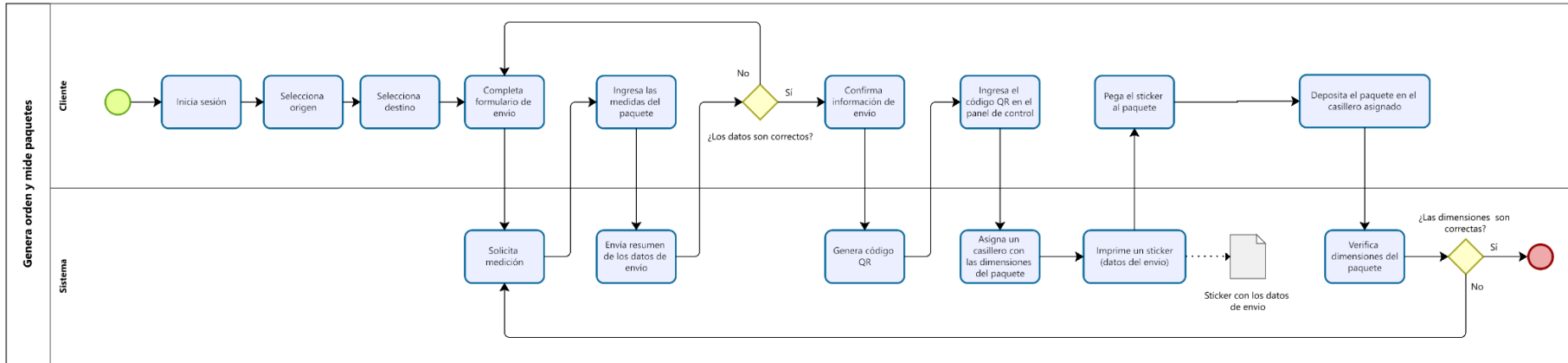


Figura 2-0. Diagrama BPMN de generar orden y medir paquete

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Subproceso:** Pago de la orden de envío.

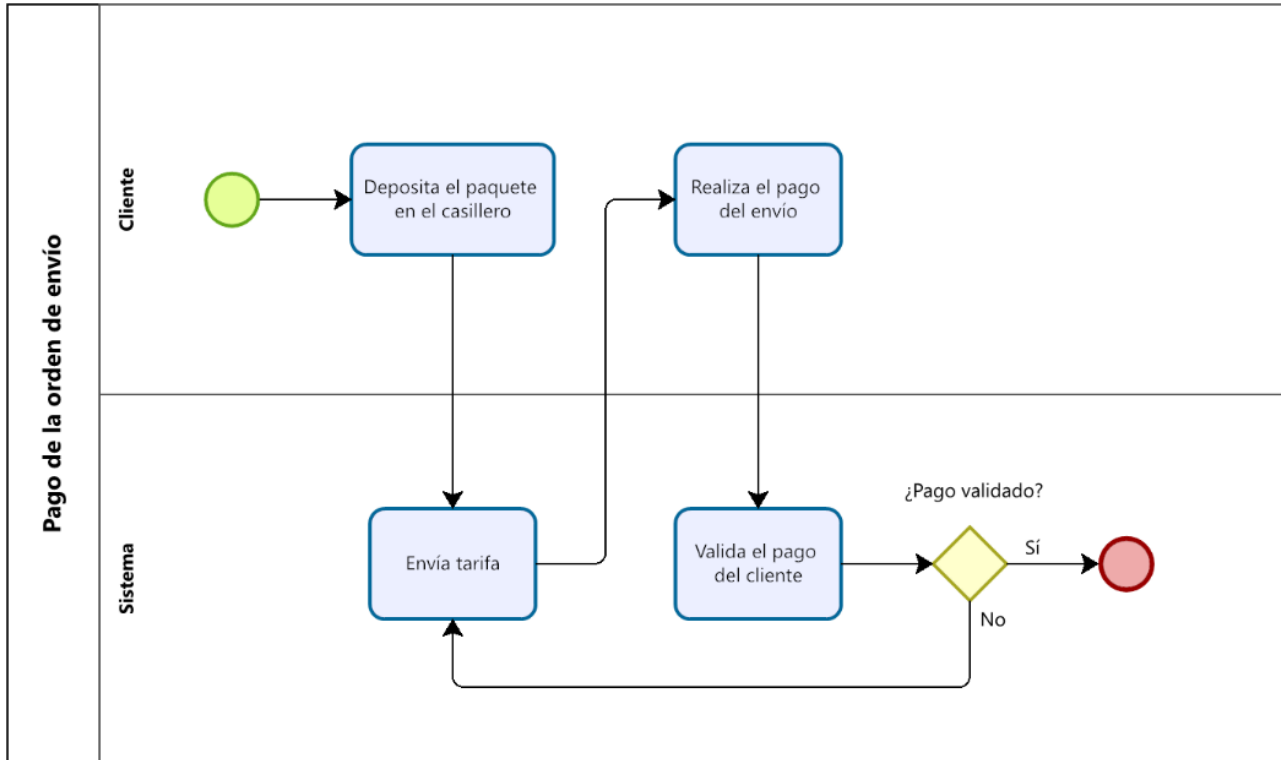


Figura 3-0. Diagrama BPMN de pago orden de envío.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Subproceso:** Recolección de los paquetes a transportar.

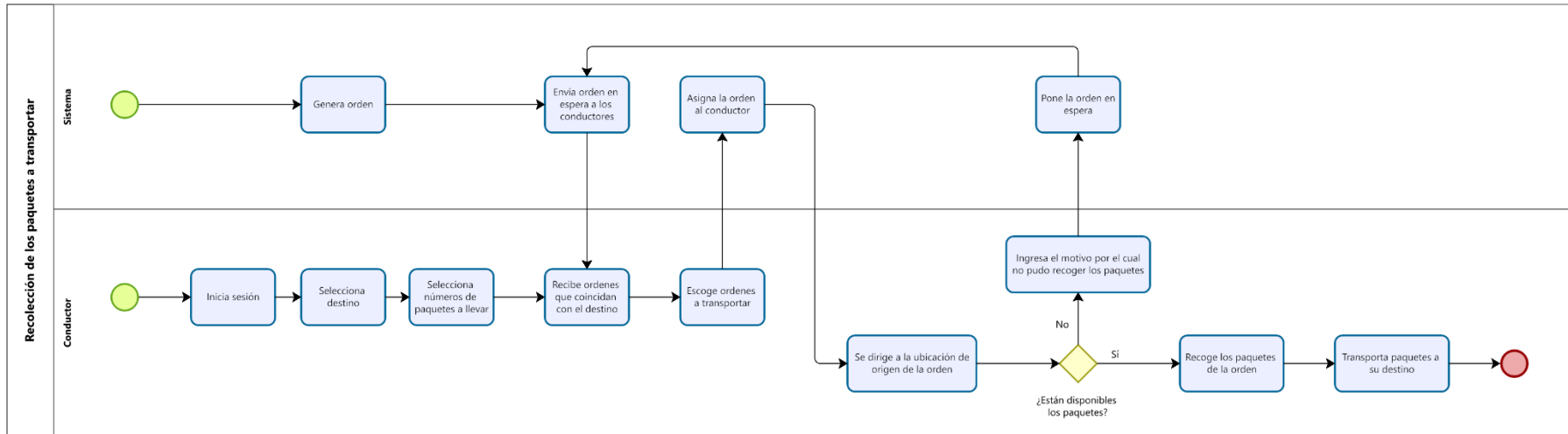


Figura 4-0. Diagrama BPMN de recolección de los paquetes a entregar

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Subproceso:** Entrega de orden a domicilio.

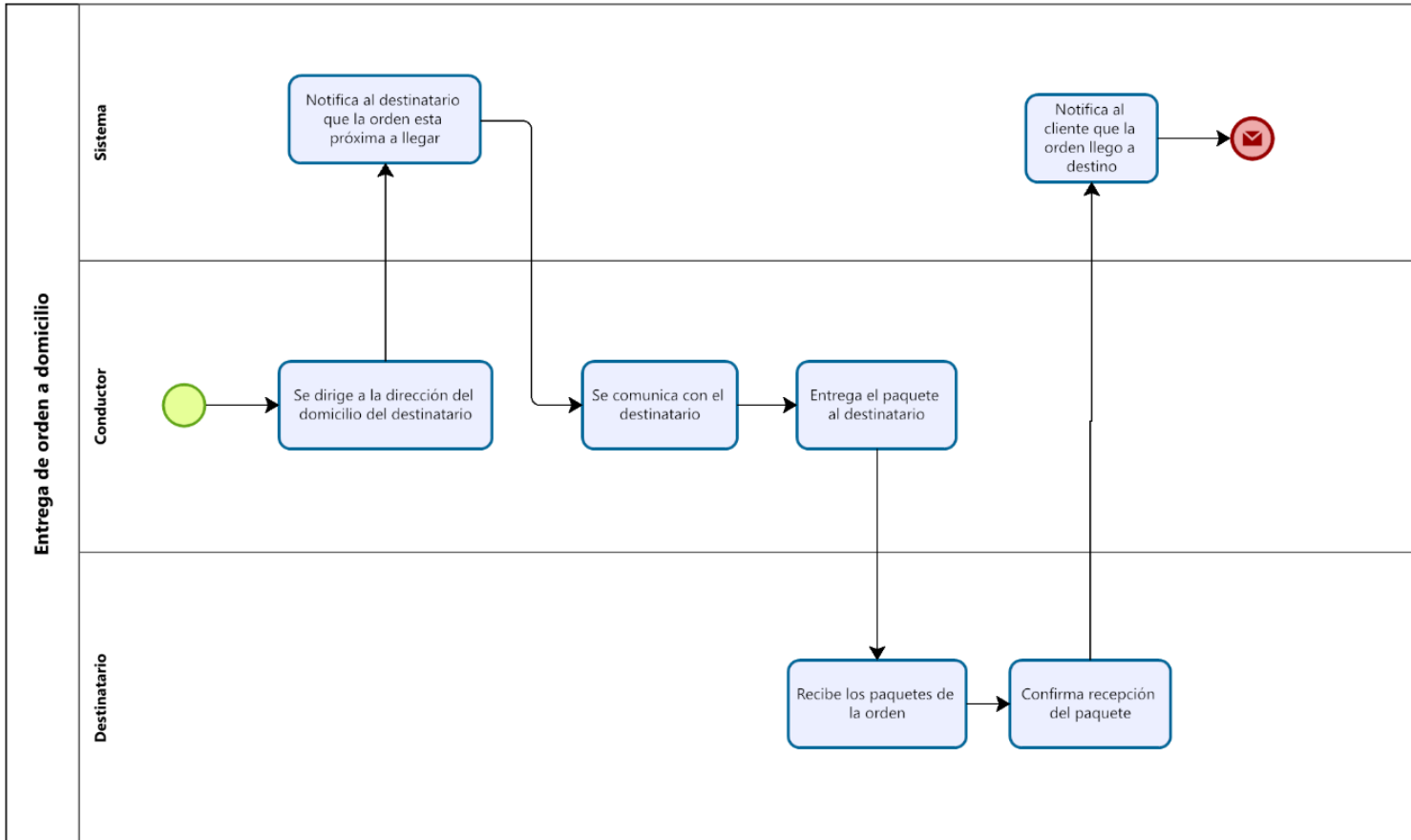


Figura 5-0. Diagrama BPMN de entrega orden a domicilio

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Subproceso:** Entrega de orden a casillero

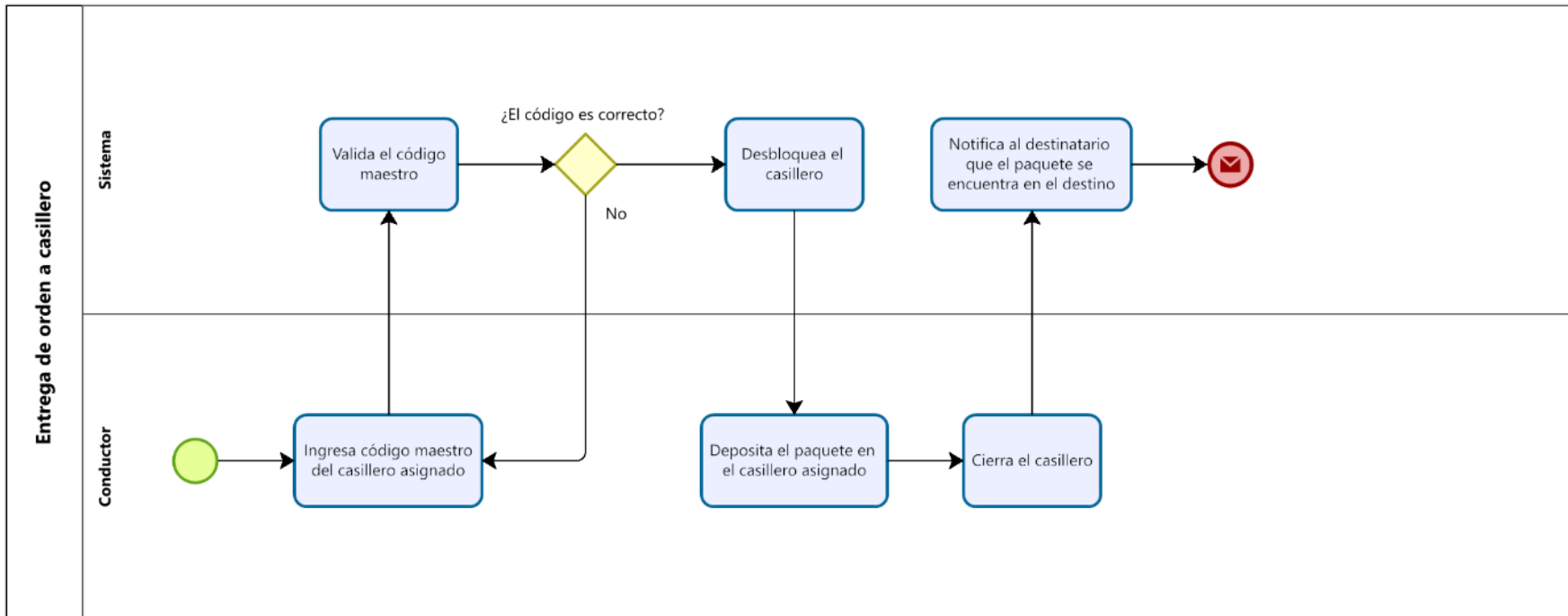


Figura 6-0. Diagrama BPMN de entrega de orden a casillero

Realizado por: Leonardo Bone, José Ochoa, 2021

ANEXO C: MATRIZ DE TRAZABILIDAD DE REQUISITOS

Por cuestiones de una mejor visualización de la matriz de trazabilidad de requerimientos basada en el PMBOK se optó por dividirla en diferentes tablas de requerimiento teniendo un total de diez.

Tabla 1-0: Matriz del requerimiento 001

Identificación	001		
Subidentificación	1.1	1.2	1.3
Descripción del requisito	El sistema permitirá crear la cuenta del conductor (Correo, Teléfono, Ciudad, Número de Cedula, Nombres Completos).	El sistema permitirá subir la documentación correspondiente al conductor (Fotografía del Conductor, Fotografía de la Cedula, Fotografía de la Licencia, Fotografía de confirmación de Identidad, Fotografía de antecedentes no penales).	El sistema permitirá subir la documentación del vehículo del conductor (Fotografía de la Matricula, Número de la placa, Marca del vehículo, Fotografía del vehículo mostrando la placa).
Versión	1.0.0	1.0.0	1.0.0
Estado actual	Completado	Completado	Completado
Última fecha estado registrado	03-11-2021	05-11-2021	10-11-2021

Criterios de aceptación	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio
Nivel de complejidad	Baja	Moderada	Moderada
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos	Agilizar procedimientos	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos	Mejorar la gestión de repartos	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil	Aplicación Móvil	Aplicación Móvil
Diseño del producto	Elaboración de casos de uso del sistema.	Elaboración de casos de uso del sistema.	Elaboración de casos de uso del sistema.
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	Se realizarán varios registros de prueba con diferentes parámetros	Se realizarán varios registros de prueba con diferentes parámetros	Se realizarán varios registros de prueba con diferentes parámetros

Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Baja	Baja	Baja

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 2-0: Matriz del requerimiento 002

Identificación	002			
Subidentificación	2.1	2.2	2.3	2.4
Descripción del requisito	El sistema permitirá iniciar sesión al conductor.	El sistema permitirá acceder a la aplicación a los conductores que hayan sido aprobados por la empresa.	El sistema permitirá recuperar la cuenta del conductor a través del envío de un código de verificación al correo electrónico vinculado a la cuenta.	El sistema permitirá cambiar actualizar la contraseña de la cuenta del conductor.
Versión	1.0.0	1.0.0	1.0.0	1.0.0
Estado actual	Completado	Completado	Completado	Completado
Última fecha estado registrado	11-11-2021	12-11-2021	17-11-2021	19-11-2021

Criterios de aceptación	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio
Nivel de complejidad	Baja	Moderada	Moderada	Baja
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos	Agilizar procedimientos	Agilizar procedimientos	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos	Mejorar la gestión de repartos	Mejorar la gestión de repartos	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil	Aplicación Móvil	Aplicación Móvil	Aplicación Móvil
Diseño del producto	Elaboración de casos de uso del sistema.	Elaboración de casos de uso del sistema.	Elaboración de casos de uso del sistema.	Diseño de la interfaz
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.

Estrategia y escenarios de pruebas	Se realizarán varios registros de prueba con diferentes parámetros.	Se realizarán varios inicios de sesión con usuarios de diferentes en diferentes estados.	Se realizarán varios inicios de sesión con usuarios de diferentes en diferentes estados.	Se realizarán varios registros de prueba con diferentes parámetros.
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Moderada	Baja	Baja	Moderada

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 3-0: Matriz del requerimiento 003

Identificación	003	
Subidentificación	3.1	3.2
Descripción del requisito	El sistema permitirá al conductor establecer el destino al cual se dirige.	El sistema permitirá al conductor cancelar pedidos seleccionados, siempre y cuando aún no los haya recogido del casillero de origen.
Versión	1.0.0	1.0.0
Estado actual	Completado	Completado

Última fecha estado registrado	02-12-2021	22-12-2021
Criterios de aceptación	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.
Nivel de complejidad	Moderada	Moderada
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil	Aplicación Móvil
Diseño del producto	Elaboración de casos de uso del sistema.	Elaboración de casos de uso del sistema.
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	Se realizarán varias solicitudes de apertura de casilleros.	Se aceptaron varios pedidos para cancelar con diferentes parámetros.

Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Moderada	Moderado

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 4-0: Matriz del requerimiento 004

Identificación	004	
Subidentificación	4.1	4.2
Descripción del requisito	El sistema enviará notificaciones al conductor por diferentes eventos (Evento Recoger Paquete, Evento Llevar Paquete, Evento Calificación-Conductor).	El sistema permitirá visualizar las diferentes notificaciones.
Versión	1.0.0	1.0.0
Estado actual	Completado	Completado
Última fecha estado registrado	24-11-2021	26-11-2021
Criterios de aceptación	Conexión con el servidor; Validación de datos; Creación del servicio	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del servicio

Nivel de complejidad	Moderada	Moderada
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil	Aplicación Móvil
Diseño del producto	Elaboración de casos de uso del sistema.	Diseño de la interfaz
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	Se enviarán varias notificaciones por cada evento.	Se enviarán varias notificaciones por cada evento.
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas

Nivel de prioridad	Baja	Baja
--------------------	------	------

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 5-0: Matriz del requerimiento 005

Identificación	005		
Subidentificación	5.1	5.2	5.3
Descripción del requisito	El sistema mostrará la ruta que debe seguir el conductor para recoger los pedidos.	El sistema mostrará la ruta que debe seguir el conductor para entregar los pedidos.	El sistema determinará el valor de tarifa a cobrar por kilómetro recorrido de la ruta de transporte del pedido
Versión	1.0.0	1.0.0	1.0.0
Estado actual	Completado	Completado	Completado
Última fecha estado registrado	06-12-2021	08-12-2021	01-12-2021
Criterios de aceptación	Maquetación terminada; Validación de datos; Trazo de rutas; Recalcular rutas.	Maquetación terminada; Validación de datos; Trazo de rutas; Recalcular rutas.	Conexion con el servidor; Validación de datos; Creación del servicio.
Nivel de complejidad	Baja	Baja	Alta

Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos	Agilizar procedimientos	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos	Mejorar la gestión de repartos	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil	Aplicación Móvil	Aplicación Móvil
Diseño del producto	Diseño de la interfaz	Diseño de la interfaz	Diseño de la interfaz
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	Se realizarán varias rutas con diferentes destinos.	Se realizarán varias rutas con diferentes destinos.	Se realizarán varias rutas con diferentes destinos.
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas	Gerente General y Director del departamento de sistemas

Nivel de prioridad	Alta	Alta	Alta
--------------------	------	------	------

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 6-0: Matriz del requerimiento 006

Identificación	006
Subidentificación	6.1
Descripción del requisito	El sistema visualizará el código maestro para abrir los casilleros de los paquetes asignados al conductor.
Versión	1.0.0
Estado actual	Completado
Última fecha estado registrado	21-12-2021
Criterios de aceptación	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.
Nivel de complejidad	Baja
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil
Diseño del producto	Diseño de la interfaz
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.

Estrategia y escenarios de pruebas	Se realizarán varias solicitudes de obtención del código maestro
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Moderada

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 7-0: Matriz del requerimiento 007

Identificación	007
Subidentificación	7.1
Descripción del requisito	El sistema calculará el nivel de experticia del conductor en base a la cantidad de pedidos repartidos satisfactoriamente.
Versión	1.0.0
Estado actual	Completado
Última fecha estado registrado	17-12-2021
Criterios de aceptación	Creación del servicio
Nivel de complejidad	Alta
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos

Entregables (EDT)	Aplicación Móvil
Diseño del producto	Elaboración de casos de uso del sistema.
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	Se validará el nivel de experiencia de diferentes tipos de conductores.
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Moderada

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 8-0: Matriz del requerimiento 008

Identificación	008
Subidentificación	8.1
Descripción del requisito	El sistema permitirá visualizar los datos del conductor y de su vehículo.
Versión	1.0.0
Estado actual	Completado
Última fecha estado registrado	25-11-2021
Criterios de aceptación	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.
Nivel de complejidad	Baja

Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil
Diseño del producto	Diseño de la interfaz
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	Se realizarán varios registros de prueba con diferentes parámetros.
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Baja

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 9-0: Matriz del requerimiento 009

Identificación	009
Subidentificación	9.1
Descripción del requisito	El sistema visualizara las estadísticas generales del conductor (Número de paquetes entregados, Número de ordenes entregadas, Total de Kilómetros recorridos).
Versión	1.0.0
Estado actual	Completado

Última fecha estado registrado	24-12-2021
Criterios de aceptación	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.
Nivel de complejidad	Moderado
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil
Diseño del producto	Diseño de la interfaz
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Baja

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

Tabla 10: Matriz del requerimiento 010

Identificación	010
Subidentificación	10.1

Descripción del requisito	El sistema permitirá al conductor reportar emergencias ocurridas durante el proceso transporte de pedidos.
Versión	1.0.0
Estado actual	Completado
Última fecha estado registrado	10-12-2021
Criterios de aceptación	Maquetación terminada; Conexión con el servidor; Validación de datos; Creación del endpoint; Creación del servicio.
Nivel de complejidad	Moderado
Necesidad, oportunidades u objetivos de negocio	Agilizar procedimientos
Objetivo del proyecto	Mejorar la gestión de repartos
Entregables (EDT)	Aplicación Móvil
Diseño del producto	Diseño de la interfaz
Desarrollo del producto	Codificación en base a las convenciones de programación definidas.
Estrategia y escenarios de pruebas	Se realizaron diferentes pruebas de la función de emergencia con diferentes parámetros.
Interesado (Stakeholder) dueño del requisito	Gerente General y Director del departamento de sistemas
Nivel de prioridad	Moderada

Fuente: PMBOK

Realizado por: Leonardo Bone, José Ochoa. 2021

ANEXO D: DISEÑO Y DICCIONARIO DE LA BASE DE DATOS

Diseño de la base de datos

- Diagrama físico completo de la base de datos.

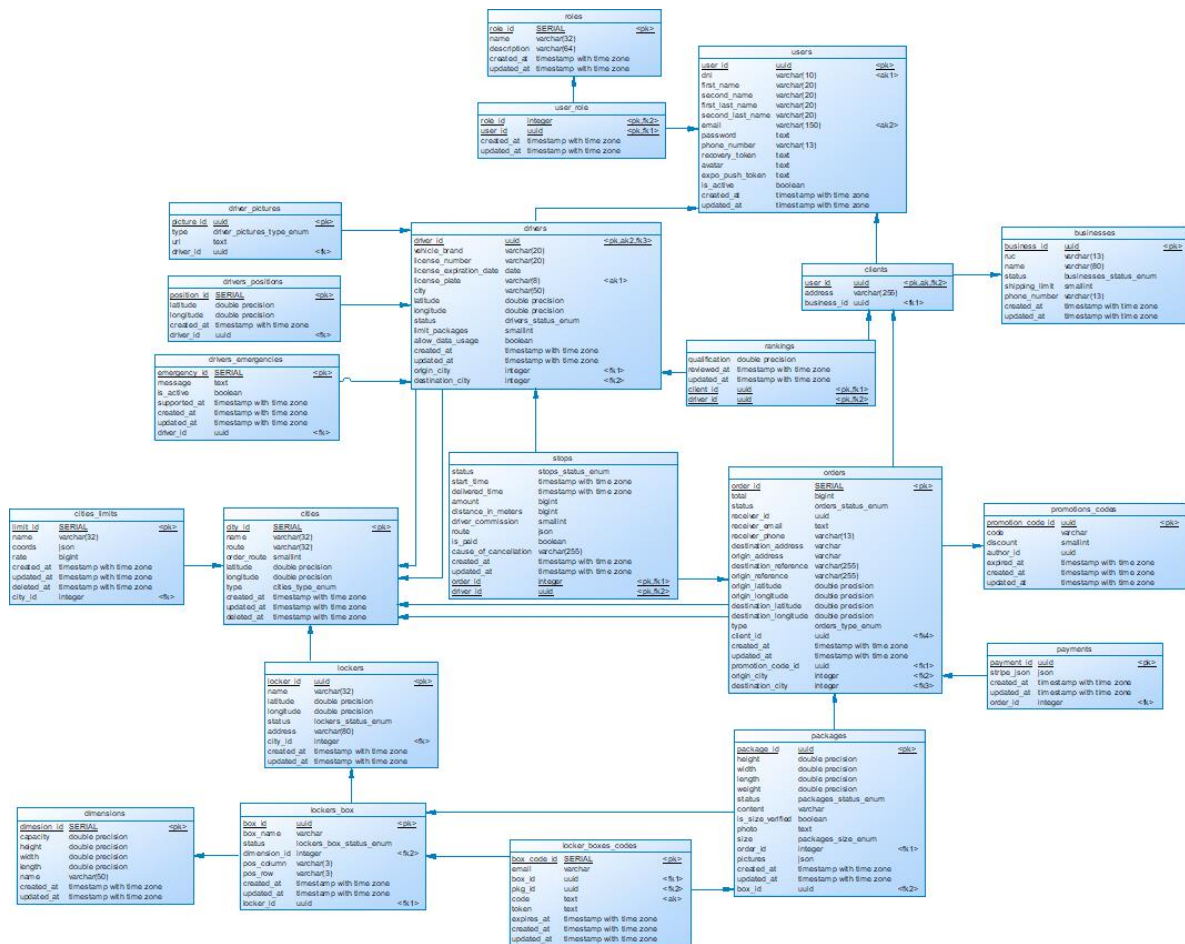


Figura 1-0. Diagrama físico de la base de datos.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Diagrama físico del módulo Users.**

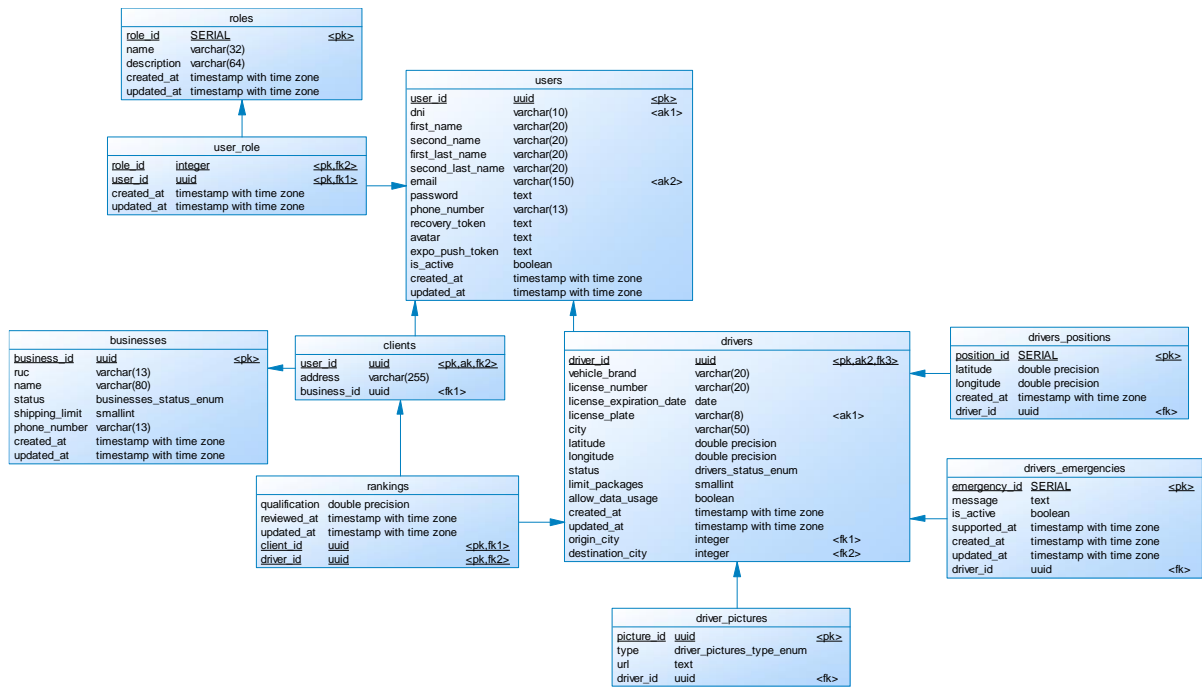


Figura 2-0. Diagrama físico del módulo Users.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Diagrama físico del módulo Orders.**

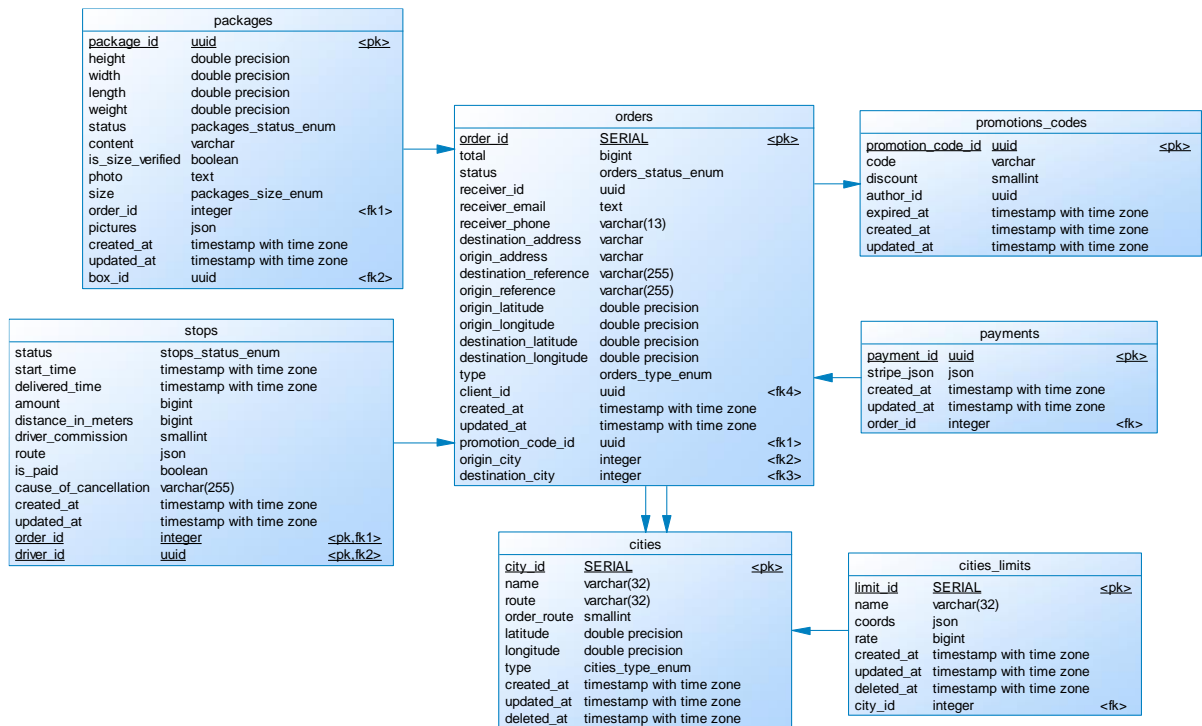


Figura 3-0. Diagrama físico del módulo Orders.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Diagrama físico del módulo Lockers.**

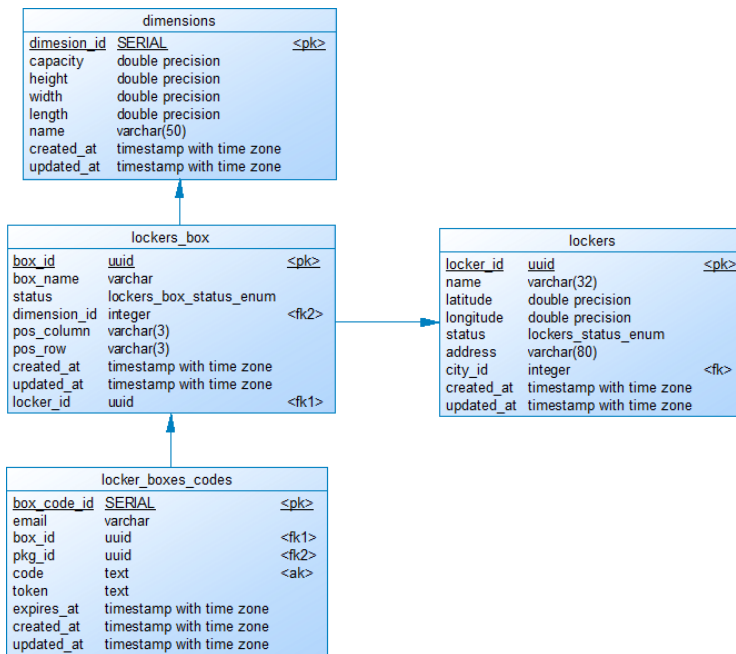


Figura 4-0. Diagrama físico del módulo Lockers.

Realizado por: Leonardo Bone, José Ochoa, 2021

Diccionario de datos

- **Tabla roles**

Tabla 1-0. Diccionario de datos de la tabla roles.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
role_id	serial	No	autoincrement()	Sí
name	varchar(32)	No		No
description	varchar(32)	No		No
created_at	timestamp with time zone default	No	now()	No
updated_at	timestamp with time zone default	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla user_role**

Tabla 2-0. Diccionario de datos de la tabla user_role.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
role_id	integer	No		Sí
user_id	uuid	No		Sí
created_at	timestamp with time zone default	No	now()	No
updated_at	timestamp with time zone default	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla users**

Tabla 3-0. Diccionario de datos de la tabla users.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
user_id	uuid	No	uuid_generate_v4()	Sí
dni	varchar(10)	No		No
first_name	varchar(20)	No		No
second_name	varchar(20)	No		No
first_last_name	varchar(20)	No		No
second_last_name	varchar(20)	No		No
email	varchar(150)	No		No
password	text	No		No
phone_number	varchar(13)	No		No
avatar	text	No	'https://...'	No
is_active	boolean	No	true	No
expo_push_token	text	Sí		No
recovery_token	text	Sí	null	No
created_at	timestamp with time zone default	No	now()	No
updated_at	timestamp with time zone default	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla drivers**

Tabla 4-0. Diccionario de datos de la tabla drivers.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
driver_id	uuid	No		Sí
vehicle_brand	varchar(20)	Sí		No
license_number	varchar(20)	No		No
license_expiration_date	date	No		No
license_plate	varchar(8)	Sí		No
city	varchar(50)	No		No
latitude	double precision	No	0	No
longitude	double precision	No	0	No
status	drivers_status_enum	No	'creating'	No
allow_data_usage	boolean	No	false	No
destination_city	integer	Sí		No
origin_city	integer	Sí		No
limit_packages	smallint	Sí		No
created_at	timestamp with time zone default	No	now()	No
updated_at	timestamp with time zone default	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla driver_pictures**

Tabla 5-0. Diccionario de datos de la tabla driver_pictures.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
picture_id	uuid	No	uuid_generate_v4()	Sí
type	driver_pictures_type_enum	No	'vehicle'	No
url	text	No		No
driver_id	uuid	No		No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla clients**

Tabla 6-0. Diccionario de datos de la tabla clients.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
user_id	uuid	No		Sí
address	varchar(255)	No		No
business_id	uuid	Sí		No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla businesses**

Tabla 7-0. Diccionario de datos de la tabla businesses.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
business_id	uuid	No	uuid_generate_v4()	Sí
ruc	varchar(13)	No		No
name	varchar(80)	No		No
status	businesses_status_emun	No	'waiting'	No
shipping_limit	smallint	No	3	No
phone_number	varchar(13)	No		No
created_at	timestamp with time zone default	No	now()	No
updated_at	timestamp with time zone default	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla orders**

Tabla 8-0. Diccionario de datos de la tabla orders.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
order_id	integer	No	autoincrement()	Sí
type	orders_type_enu m	No		No
total	bigint	No		No
status	orders_status_enu m	No	'waiting_package'	No
client_id	uuid	No		No

receiver_id	uuid	Sí		No
receiver_email	text	Sí		No
receiver_phone	varchar	Sí		No
promotion_code_id	uuid	Sí		No
origin_city	integer	No		No
destination_city	integer	No		No
origin_latitude	double precision	No	0	No
origin_longitude	double precision	No	0	No
origin_reference	varchar(255)	No		No
origin_address	varchar	No		No
destination_latitude	double precision	No	0	No
destination_longitude	double precision	No	0	No
destination_address	varchar	No		No
destination_reference	varchar(255)	No		No
created_at	timestamp with time zone default	No	now()	No
updated_at	timestamp with time zone default	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla promotions_codes**

Tabla 9-0. Diccionario de datos de la tabla promotions_code.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
promotion_code_id	uuid	No	uuid_generate_v4()	Si
autor_id	uuid	No		No
code	varchar	No		No
discount	smallint	No		No
expired_at	timestamp with time zone	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla payments**

Tabla 10-0. Diccionario de datos de la tabla payments.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
payment_id	uuid	No	uuid_generate_v4()	Si
stripe_json	json	No		No
order_id	uuid	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla packages**

Tabla 11-0. Diccionario de datos de la tabla packages.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
package_id	uuid	No	uuid_generate_v4()	Si
height	double precision	No		No
width	double precision	No		No
length	double precision	No		No
weight	double precision	No		No
status	packages_status_enum	No	'drop_off_on_box'	No
order_id	integer	No		No
content	varchar	No		No
size	packages_size_enum	No	'small'	No
is_size_verified	boolean	No	false	No
box_id	uuid	Sí		No
pictures	json	No		No
photo	text	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla stops**

Tabla 12-0. Diccionario de datos de la tabla stops.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
order_id	integer	No		Si
driver_id	uuid	No		Si
cause_of_cancellation	varchar	Sí		No
amount	bigint	Sí		No
distance_in_meters	bigint	Sí		No
driver_commission	smallint	Sí		No
route	json	Sí		No
is_paid	boolean	No	false	No
status	stops_status_enum	No	'waiting'	No
start_time	timestamp with time zone	Sí		No
delivered_time	timestamp with time zone	Sí		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla lockers**

Tabla 13-0. Diccionario de datos de la tabla lockers.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
locker_id	uuid	No	uuid_generate_v4()	Si
address	varchar	No		No
city_id	integer	No		No
name	varchar	No		No
latitude	double precision	No		No
longitude	double precision	No		No
status	lockers_status_enum	No	'closed'	No
created_at	timestamp with time zone	No	now()	No

updated_at	timestamp with time zone	No	now()	No
------------	--------------------------	----	-------	----

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla lockers_box**

Tabla 14-0. Diccionario de datos de la tabla lockers_box.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
box_id	uuid	No	uuid_generate_v4()	Si
status	lockers_box_status_enum	No	'closed'	No
dimension_id	integer	No		No
pos_column	varchar	No	'000'	No
pos_row	varchar	No	'000'	No
box_name	varchar	No		No
locker_id	uuid	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla rankings**

Tabla 15-0. Diccionario de datos de la tabla rankings.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
client_id	uuid	No		Si
driver_id	uuid	No		Si
qualification	double precision	No	5	No
reviewed_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla rates**

Tabla 16-0. Diccionario de datos de la tabla rates.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
rate_id	integer	No	autoincrement()	Sí
name	character varying	No		No
amount	bigint	No		No

type	rates_type_enum	No		No
deleted_at	timestamp with time zone	Sí		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla locker_boxes_codes**

Tabla 17-0. Diccionario de datos de la tabla locker_boxes_codes.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
box_code_id	integer	No	autoincrement()	Sí
pkg_id	uuid	No		No
box_id	uuid	YES	NULL	No
code	text	No		No
expires_at	timestamp with time zone	No		No
token	text	No		No
email	varchar	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla drivers_positions**

Tabla 18-0. Diccionario de datos de la tabla drivers_positions.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
position_id	integer	No	autoincrement()	Sí
latitude	double precision	No		No
longitude	double precision	No		No
driver_id	uuid	No		No
created_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla drivers_emergencies**

Tabla 19-0. Diccionario de datos de la tabla drivers_emergencies.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
emergency_id	integer	No	autoincrement()	Sí
supported_at	timestamp with time zone	Sí		No
is_active	boolean	No	true	No
message	text	No		No
driver_id	uuid	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla dimensions**

Tabla 20-0. Diccionario de datos de la tabla dimensions.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
dimension_id	integer	No	autoincrement()	Sí
capacity	double precisión	No		No
height	double precisión	No		No
width	double precisión	No		No
name	varchar	No		No
length	double precision	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla commissions**

Tabla 21-0. Diccionario de datos de la tabla commissions.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
commission_id	integer	No	autoincrement()	Sí
type_travel	commissions_type_travel_e num	No		No
locker_commission	smallint	No		No

own_commission	smallint	No		No
driver_commission	smallint	No		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla cities**

Tabla 22-0. Diccionario de datos de la tabla cities.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
city_id	integer	No	autoincrement()	Sí
longitude	double precision	No	0	No
latitude	double precision	No	0	No
type	cities_type_enum	No	'primary'	No
route	character varying	No		No
name	character varying	No		No
order_route	smallint	No		No
deleted_at	timestamp with time zone	Sí		No
created_at	timestamp with time zone	No	now()	No
updated_at	timestamp with time zone	No	now()	No

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Tabla cities_limits**

Tabla 23-0. Diccionario de datos de la tabla cities_limits.

Nombre	Tipo de dato	Permite nulos	Valor por defecto	Clave primaria
limit_id	integer	No	autoincrement()	Sí
rate	bigint	No		No
city_id	integer	Sí		No
name	character varying	No		No
coords	json	No		No
deleted_at	timestamp with time zone	Sí		No
created_at	timestamp with time zone	No	now()	No

updated_at	timestamp with time zone	No	now()	No
------------	--------------------------	----	-------	----

Realizado por: Leonardo Bone, José Ochoa, 2021

ANEXO E: DISEÑO DE INTERFACES DE USUARIO

- Inicio de sesión

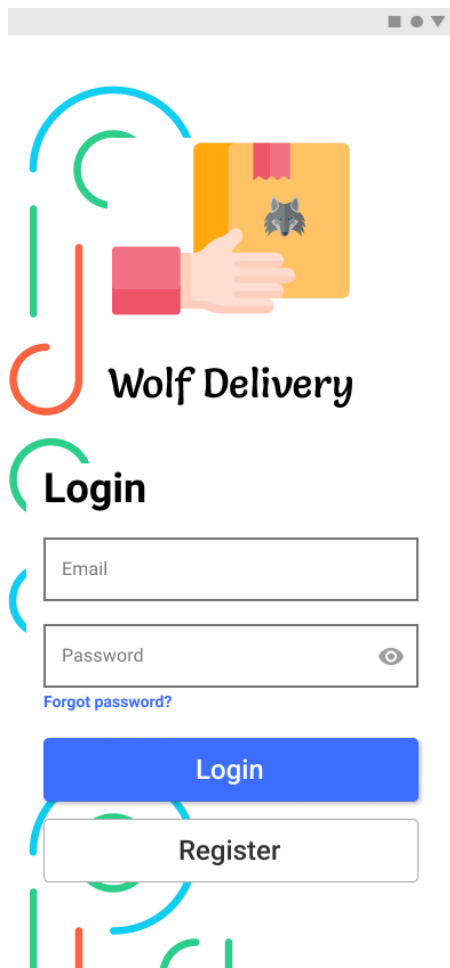


Figura 1-0. Pantalla de inicio de sesión.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Registrar conductor**

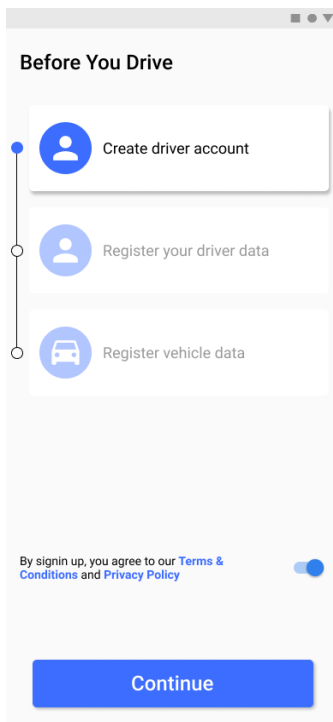


Figura 2-0. Pantalla de registrar conductor.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Lista de ordenes**

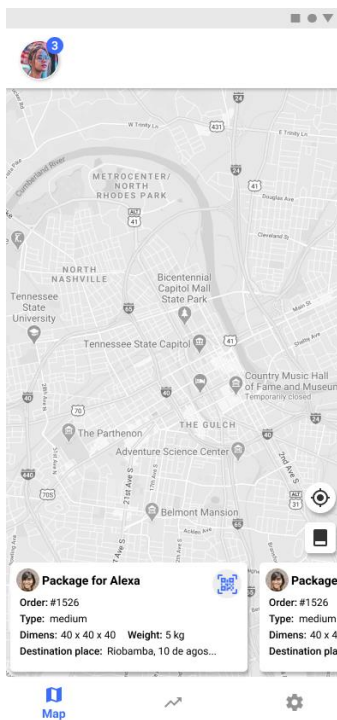


Figura 3-0. Pantalla de lista de ordenes en el mapa.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Detalles de la orden**

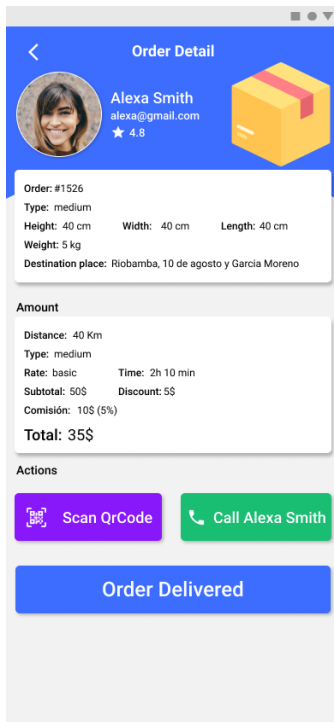


Figura 4-0. Pantalla de detalles de la orden.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Estadísticas del conductor.**

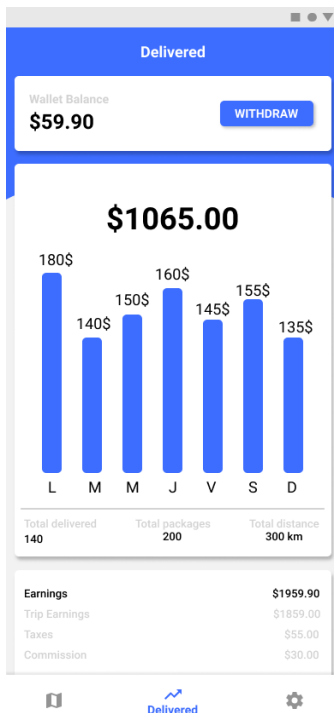


Figura 5-0. Pantalla de estadísticas del conductor.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Configuraciones de la aplicación**

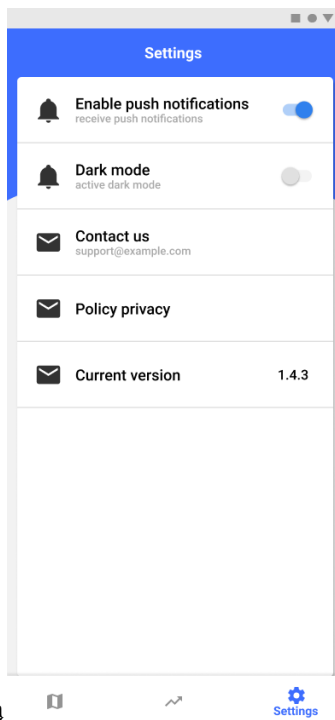


Figura 6-0. Pantalla de configuraciones de la aplicación.

Realizado por: Leonardo Bone, José Ochoa, 2021

- **Perfil del conductor**

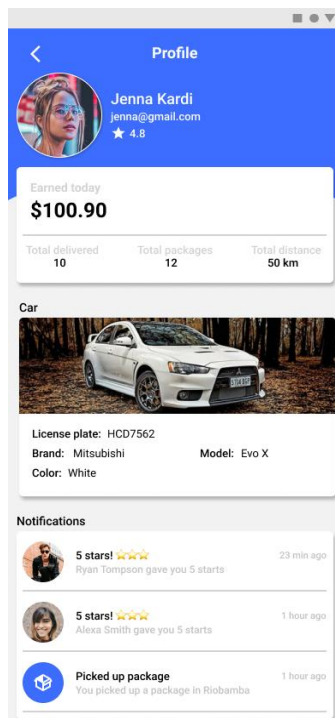


Figura 7-0. Pantalla de perfil del conductor.

Realizado por: Leonardo Bone, José Ochoa, 2021

ANEXO F: HISTORIAS DE USUARIOS

Tabla 1-0: Historia de usuario 1

Historia de Usuario 1			
ID: HU-01	Nombre de la historia: Crear una cuenta		
Usuario: Conductor	Sprint: 2		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 01/11/2021	Fecha Fin: 03/11/2021		
Descripción: Como conductor quiero crearme una cuenta para poder realizar el transporte de paquetería.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-01 PA-01	Ingresar datos incorrectos y mostrar mensaje de error (validar datos).	Exitoso	José Ochoa
HU-01 PA-02	Ingresar datos duplicados de otro usuario ya registrado y mostrar mensaje de que ya ha sido ingresado.	Exitoso	José Ochoa
HU-01 PA-03	No permitir el ingreso de campos vacíos.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-01 TI-01	Crear la interfaz de usuario para el registro de cuenta del conductor.	4	
HU-01 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2	
HU-01 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para la creación de cuenta del conductor.	3	
HU-01 TI-04	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 2-0: Historia de usuario 2

Historia de Usuario 2			
ID: HU-02	Nombre de la historia: Completar información del conductor		
Usuario: Conductor	Sprint: 2		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 03/11/2021	Fecha Fin: 05/11/2021		
Descripción: Como conductor quiero completar la información de registro de conductor para ser aprobado en el sistema.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-02 PA-01	Ingresar datos incorrectos y mostrar mensaje de error (validar datos).	Exitoso	Leonardo Bone
HU-02 PA-02	Ingresar datos duplicados de otro usuario ya registrado y mostrar mensaje de que ya ha sido ingresado.	Exitoso	Leonardo Bone

HU-02 PA-03	No permitir el ingreso de campos vacíos.	Exitoso	Leonardo Bone
HU-02 PA-04	Ingresar imágenes con formatos no permitidos y mostrar mensaje de error.	Exitoso	Leonardo Bone
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-02 TI-01	Crear la interfaz de usuario para completar la información del conductor.	3	
HU-02 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2	
HU-02 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para la completar la información del conductor.	2	
HU-02 TI-04	Guardar las imágenes subidas en el formulario.	2	
HU-02 TI-05	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 3-0: Historia de usuario 3

Historia de Usuario 3			
ID: HU-03	Nombre de la historia: Completar información del vehículo		
Usuario: Conductor	Sprint: 2		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 08/11/2021	Fecha Fin: 10/11/2021		
Descripción: Como conductor quiero completar la información del vehículo para ser aprobado en el sistema.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-03 PA-01	Ingresar datos incorrectos y mostrar mensaje de error (validar datos).	Exitoso	José Ochoa
HU-03 PA-02	Ingresar datos duplicados de otro vehículo ya registrado y mostrar mensaje de que ya ha sido ingresado.	Exitoso	José Ochoa
HU-03 PA-03	No permitir el ingreso de campos vacíos.	Exitoso	José Ochoa
HU-03 PA-04	Ingresar imágenes con formatos no permitidos y mostrar mensaje de error.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-03 TI-01	Crear la interfaz de usuario para completar la información del vehículo.	3	
HU-03 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2	
HU-03 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para la completar la información del vehículo.	2	
HU-03 TI-04	Guardar las imágenes subidas en el formulario.	2	

HU-03 TI-05	Conectar la interfaz de usuario con el backend.	1
----------------	---	---

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 4-0: Historia de usuario 4

Historia de Usuario 4			
ID: HU-04 Nombre de la historia: Iniciar sesión			
Usuario: Conductor		Sprint: 2	
Prioridad: Media (Alta/Media/Baja)		Puntos Estimados: 5	
		Puntos Reales: 8	
Fecha Inicio: 10/11/2021		Fecha Fin: 11/11/2021	
Descripción: Como conductor quiero iniciar sesión en la aplicación para poder recibir órdenes de transporte de paquetes.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-04 PA-01	Ingresar datos incorrectos y mostrar mensaje de error (validar datos).	Exitoso	Leonardo Bone
HU-04 PA-02	No permitir el ingreso de campos vacíos.	Exitoso	Leonardo Bone
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-04 TI-01	Crear la interfaz de usuario para el inicio de sesión.	2	
HU-04 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2	
HU-04 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para permitir el inicio de sesión del conductor.	3	
HU-04 TI-04	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 5-0: Historia de usuario 5

Historia de Usuario 5			
ID: HU-05 Nombre de la historia: Permitir ingreso solo a los conductores aprobados.			
Usuario: Administrador		Sprint: 2	
Prioridad: Baja (Alta/Media/Baja)		Puntos Estimados: 5	
		Puntos Reales: 5	
Fecha Inicio: 11/11/2021		Fecha Fin: 12/11/2021	
Descripción: Como administrador quiero que solo los conductores aprobados ingresen a la aplicación para tener conductores validados transportando paquetería.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-05 PA-01	Iniciar sesión con una cuenta sin aprobar y mostrar mensaje de que aún no ha sido aprobado.	Exitoso	José Ochoa
HU-05 PA-02	Iniciar sesión con una cuenta de otro rol diferente al conductor y mostrar mensaje de que la cuenta no tiene permisos suficientes.	Exitoso	José Ochoa

Tareas de Ingeniería		
ID TI	Descripción TI	Esfuerzo
HU-05 TI-01	Crear la interfaz de usuario de información del estado de aprobación del conductor.	1
HU-05 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	1
HU-05 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para validar el estado del conductor.	2
HU-05 TI-04	Conectar la interfaz de usuario con el backend.	1

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 6-0: Historia de usuario 6

Historia de Usuario 6			
ID: HU-06	Nombre de la historia: Recuperar cuenta de conductor.		
Usuario: Conductor	Sprint: 3		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 15/11/2021	Fecha Fin: 17/11/2021		
Descripción: Como conductor quiero recuperar mi cuenta para tener acceso a la aplicación.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-06 PA-01	Ingresar un email que no existe y mostrar mensaje de error (validar datos).	Exitoso	Leonardo Bone
HU-06 PA-02	Usar un enlace de recuperación de la cuenta caducado y mostrar mensaje de error	Exitoso	Leonardo Bone
HU-06 PA-03	No permitir el ingreso de campos vacíos.	Exitoso	Leonardo Bone
HU-06 PA-04	Ingresar contraseñas que no coincidan y mostrar mensaje de error.	Exitoso	Leonardo Bone
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-06 TI-01	Crear la interfaz de usuario para recuperar cuenta del conductor.	2	
HU-06 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	1	
HU-06 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para recuperar el acceso a la aplicación.	4	
HU-06 TI-04	Enviar email al correo del usuario con el enlace de recuperación de cuenta.	2	
HU-06 TI-05	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 7-0: Historia de usuario 7

Historia de Usuario 7			
ID: HU-07	Nombre de la historia: Actualizar contraseña de la cuenta.		
Usuario: Conductor	Sprint: 3		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 17/11/2021	Fecha Fin: 19/11/2021		
Descripción: Como conductor quiero actualizar mi contraseña de usuario para tener mayor seguridad.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-07 PA-01	Ingresar una contraseña actual incorrecta y mostrar mensaje de error (validar datos).	Exitoso	José Ochoa
HU-07 PA-02	Ingresar contraseñas que no coincidan y mostrar mensaje de error (validar datos).	Exitoso	José Ochoa
HU-07 PA-03	No permitir el ingreso de campos vacíos.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-07 TI-01	Crear la interfaz de usuario para la actualización de la contraseña de la cuenta.	2	
HU-07 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2	
HU-07 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para la actualizar la contraseña.	4	
HU-07 TI-04	Conectar la interfaz de usuario con el backend.	2	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 8-0: Historia de usuario 8

Historia de Usuario 8			
ID: HU-08	Nombre de la historia: Determinar tarifa a cobrar de la ruta.		
Usuario: Conductor	Sprint: 4		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 29/11/2021	Fecha Fin: 01/12/2021		
Descripción: Como conductor quiero conocer la tarifa asignada al pedido para saber cuándo voy a cobrar por su transporte.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-08 PA-01	Generar diferentes pedidos, con distintas rutas y distancias para obtener varias tarifas.	Exitoso	José Ochoa
HU-08 PA-02	Validar que el cálculo de costo total de la ruta sea correcto.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-08 TI-01	Crear la interfaz de usuario para mostrar la tarifa y el costo total de la ruta.	3	

HU-08 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2
HU-08 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para calcular la tarifa y costo total de la ruta del pedido.	3
HU-08 TI-04	Conectar la interfaz de usuario con el backend.	2

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 9-0: Historia de usuario 9

Historia de Usuario 9			
ID: HU-09	Nombre de la historia: Seleccionar lugar de destino.		
Usuario: Conductor			Sprint: 4
Prioridad: Media (Alta/Media/Baja)			Puntos Estimados: 6
			Puntos Reales: 6
Fecha Inicio: 01/12/2021	Fecha Fin: 02/12/2021		
Descripción: Como conductor quiero elegir el lugar de destino al cual me dirijo para que se me muestren las ordenes que coinciden con mi ruta.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-09 PA-01	No permitir el ingreso de campos vacíos.	Exitoso	Leonardo Bone
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-09 TI-01	Crear la interfaz de usuario para la selección del lugar de destino del conductor.	1	
HU-09 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	1	
HU-09 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para la selección del lugar de destino.	3	
HU-09 TI-04	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 10-0: Historia de usuario 10

Historia de Usuario 10	
ID: HU-10	Nombre de la historia: Generar notificaciones por eventos.
Usuario: Conductor	Sprint: 3
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 10
	Puntos Reales: 10
Fecha Inicio: 22/11/2021	Fecha Fin: 24/11/2021
Descripción: Como conductor quiero tener recibir diferentes notificaciones por los eventos ocurridos durante el proceso de reparto de paquetería para tener una noción de lo que ocurre en cada momento.	

Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-10 PA-01	Verificar que cada notificación corresponda a su propio evento.	Exitoso	José Ochoa
HU-10 PA-02	Verificar que el backend no emita notificaciones al azar.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-10 TI-01	Instalar la librería en el backend para el envío de notificaciones push.	1	
HU-10 TI-02	Crear el servicio encargado del envío de las notificaciones.	3	
HU-10 TI-03	Programar la lógica para enviar notificaciones segmentadas por eventos.	4	
HU-10 TI-04	Inyectar el servicio de notificaciones en los procesos correspondientes a cada evento del sistema.	2	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 11-0: Historia de usuario 11

Historia de Usuario 11			
ID: HU-11	Nombre de la historia: Visualizar notificaciones.		
Usuario: Conductor	Sprint: 3		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 5		
	Puntos Reales: 5		
Fecha Inicio: 25/11/2021	Fecha Fin: 26/11/2021		
Descripción: Como conductor quiero visualizar las diferentes notificaciones recibidas para tener un historial de las diferentes acciones realizadas.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-11 PA-01	Verificar que la aplicación despliegue las notificaciones.	Exitoso	Leonardo Bone
HU-11 PA-02	Verificar que cada notificación corresponda a su propio evento.	Exitoso	Leonardo Bone
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-11 TI-01	Instalar la librería en la aplicación móvil para poder recibir y visualizar las notificaciones.	1	
HU-11 TI-02	Crear el handler para recibir las notificaciones.	2	
HU-11 TI-03	Crear el servicio para enviar el token de notificaciones push del dispositivo.	2	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 12-0: Historia de usuario 12

Historia de Usuario 12			
ID: HU-12	Nombre de la historia: Mostrar ruta de recogida del pedido.		
Usuario: Conductor	Sprint: 4		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 7		
	Puntos Reales: 7		
Fecha Inicio: 03/12/2021	Fecha Fin: 06/12/2021		
Descripción: Como conductor quiero visualizar la ruta a seguir para poder recoger los pedidos.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-12 PA-01	Verificar que la ruta mostrada para recoger el pedido sea la correcta.	Exitoso	José Ochoa
HU-12 PA-02	Verificar que la ruta mostrada corresponda al pedido seleccionado.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-12 TI-01	Crear el componente que renderiza la ruta en el mapa.	3	
HU-12 TI-02	Programar la lógica para el recalcu de la ruta.	4	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 13-0: Historia de usuario 13

Historia de Usuario 13			
ID: HU-13	Nombre de la historia: Mostrar ruta de entrega del pedido.		
Usuario: Conductor	Sprint: 4		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 7		
	Puntos Reales: 7		
Fecha Inicio: 06/12/2021	Fecha Fin: 08/12/2021		
Descripción: Como conductor quiero visualizar la ruta a seguir para poder entregar el pedido.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-13 PA-01	Verificar que la ruta mostrada para entregar el pedido sea la correcta.	Exitoso	Leonardo Bone
HU-13 PA-02	Verificar que la ruta mostrada corresponda al pedido seleccionado.	Exitoso	Leonardo Bone
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-13 TI-01	Crear el componente que renderiza la ruta en el mapa.	3	
HU-13 TI-02	Programar la lógica para el recalcu de la ruta.	4	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 14-0: Historia de usuario 14

Historia de Usuario 14			
ID: HU-14	Nombre de la historia: Mostrar código maestro del casillero.		
Usuario: Conductor	Sprint: 5		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 5		
	Puntos Reales: 5		
Fecha Inicio: 20/12/2021	Fecha Fin: 21/12/2021		
Descripción: Como conductor quiero visualizar el código maestro del casillero para poder abrirlo.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-14 PA-01	Validar que cada paquete tenga su propio código maestro de casillero.	Exitoso	José Ochoa
HU-14 PA-02	Validar que se genere un nuevo código maestro.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-14 TI-01	Crear la interfaz de usuario para visualizar el código maestro del casillero asignado al paquete.	1	
HU-14 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	1	
HU-14 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para mostrar el código maestro del casillero.	2	
HU-14 TI-04	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 15-0: Historia de usuario 15

Historia de Usuario 15			
ID: HU-15	Nombre de la historia: Reportar emergencia.		
Usuario: Conductor	Sprint: 4		
Prioridad: Alta (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 08/12/2021	Fecha Fin: 10/12/2021		
Descripción: Como conductor quiero reportar una emergencia para poder informar a los administradores de cualquier situación imprevista durante el proceso de reparto de paquetería.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-15 PA-01	Validar que se ingrese una razón/causa al momento de reportar la emergencia.	Exitoso	José Ochoa
HU-15 PA-02	Validar que no se pueda crear otra emergencia mientras exista una emergencia activa.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-15 TI-01	Crear la interfaz de usuario para reportar emergencias.	3	

HU-15 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	2
HU-15 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para reportar la emergencia del conductor.	3
HU-15 TI-04	Conectar la interfaz de usuario con el backend.	2

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 16-0: Historia de usuario 16

Historia de Usuario 16			
ID: HU-16	Nombre de la historia: Nivel de experiencia del conductor.		
Usuario: Conductor	Sprint: 5		
Prioridad: Media (Alta/Media/Baja)	Puntos Estimados: 20		
	Puntos Reales: 20		
Fecha Inicio: 13/12/2021	Fecha Fin: 17/12/2021		
Descripción: Como conductor quiero visualizar mi nivel de experiencia para poder llevar más paquetes.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-16 PA-01	Verificar si el nivel de experiencia del conductor es correcto.	Exitoso	José Ochoa
HU-16 PA-02	Verificar que el nivel de experiencia corresponda al conductor.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-16 TI-01	Crear la interfaz de usuario para visualizar la experiencia del usuario.	5	
HU-16 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	5	
HU-16 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para calcular la experiencia del conductor.	5	
HU-16 TI-04	Conectar la interfaz de usuario con el backend.	5	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 17-0: Historia de usuario 17

Historia de Usuario 17	
ID: HU-17	Nombre de la historia: Visualizar datos del conductor y vehículo.
Usuario: Conductor	Sprint: 3
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 5
	Puntos Reales: 5
Fecha Inicio: 24/11/2021	Fecha Fin: 25/11/2021
Descripción: Como conductor quiero visualizar mis datos de conductor y del vehículo para tener un perfil en el sistema.	

Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-17 PA-01	Verificar si los datos corresponden al conductor.	Exitoso	José Ochoa
HU-17 PA-02	Verificar si los datos corresponden al vehículo del conductor.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-17 TI-01	Crear la interfaz de usuario para visualizar los datos del conductor y vehículo.	2	
HU-17 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	1	
HU-17 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para recuperar la información del conductor y del vehículo.	1	
HU-17 TI-04	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 18: Historia de usuario 18

Historia de Usuario 18			
ID: HU-18	Nombre de la historia: Visualizar estadísticas.		
Usuario: Conductor	Sprint: 5		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 10		
	Puntos Reales: 10		
Fecha Inicio: 22/12/2021	Fecha Fin: 24/12/2021		
Descripción: Como conductor quiero visualizar mis estadísticas de los viajes para tener un conocimiento de todo lo realizado hasta el momento.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-18 PA-01	Verificar si los datos corresponden al conductor.	Exitoso	José Ochoa
HU-18 PA-02	Verificar los datos se visualicen en un formato correcto.	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-18 TI-01	Crear la interfaz de usuario para visualizar las estadísticas de los viajes realizados por el conductor.	5	
HU-18 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	1	
HU-18 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para calcular las estadísticas generadas por el conductor.	3	
HU-18 TI-04	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.

Tabla 19-0: Historia de usuario 19

Historia de Usuario 19			
ID: HU-19	Nombre de la historia: Cancelar pedidos.		
Usuario: Conductor	Sprint: 5		
Prioridad: Baja (Alta/Media/Baja)	Puntos Estimados: 5		
	Puntos Reales: 5		
Fecha Inicio: 21/12/2021	Fecha Fin: 22/12/2021		
Descripción: Como conductor quiere tener la posibilidad de cancelar pedidos que no puedo llevar para permitir que otros conductores los transporten.			
Pruebas de Aceptación			
ID PA	Criterio	Estado	Responsable
HU-19 PA-01	Verificar que el pedido no haya sido recogido antes de cancelarlo.	Exitoso	José Ochoa
HU-19 PA-02	Verificar que el pedido vuelva a salir como disponible para el resto de conductores.	Exitoso	José Ochoa
HU-19 PA-03	Validar que siempre se ingrese una razón por la cual se cancela el pedido	Exitoso	José Ochoa
Tareas de Ingeniería			
ID TI	Descripción TI	Esfuerzo	
HU-19 TI-01	Crear la interfaz de usuario para visualizar el formulario de cancelación de pedido.	2	
HU-19 TI-02	Crear el controlador y endpoint para recibir la petición HTTP.	1	
HU-19 TI-03	Crear el acceso a datos (patrón repositorio), la lógica de negocio y servicios para cancelar el transporte del pedido.	1	
HU-19 TI-04	Conectar la interfaz de usuario con el backend.	1	

Fuente: Leonardo Bone y José Ochoa, 2021.

Realizado por: Leonardo Bone y José Ochoa, 2021.



epoch

**Dirección de Bibliotecas y
Recursos del Aprendizaje**

**UNIDAD DE PROCESOS TÉCNICOS Y ANÁLISIS BIBLIOGRÁFICO Y
DOCUMENTAL**

REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 22 / 07 / 2022

INFORMACIÓN DEL AUTOR/A (S)
Nombres – Apellidos: JOSÉ VICENTE OCHOA VALAREZO EDWARD LEONARDO BONE CUSME
INFORMACIÓN INSTITUCIONAL
Facultad: INFORMÁTICA Y ELECTRÓNICA
Carrera: SOFTWARE
Título a optar: INGENIERO DE SOFTWARE
f. Analista de Biblioteca responsable: Lcdo. Holger Ramos, MSc.

1397-DBRA-UPT-2022

