

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA EN SISTEMAS

DESARROLLO DE UNA PWA PARA LA GESTIÓN DE COURIERS MOTORIZADOS PARA MICROEMPRESAS DE QUITO

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO DE SISTEMAS
INFORMÁTICOS Y DE COMPUTACIÓN**

CHRISTIAN VINICIO BENÍTEZ CHÁVEZ

christian.benitez@epn.edu.ec

RICARDO FIDEL BETANCOURT HIDALGO

ricardo.betancourt@epn.edu.ec

DIRECTOR: HERNÁN DAVID ORDÓÑEZ CALERO

hernan.ordonez@epn.edu.ec

CODIRECTOR: JULIÁN ANDRÉS GALINDO LOSADA

julian.galindo@epn.edu.ec

Quito, septiembre 2021

DECLARACIÓN

Nosotros, CHRISTIAN VINICIO BENÍTEZ CHÁVEZ y RICARDO FIDEL BETANCOURT HIDALGO, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría, que no ha sido previamente presentada para ningún grado o calificación profesional, y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.



Christian Vinicio Benítez Chávez



Ricardo Fidel Betancourt Hidalgo

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Christian Vinicio Benítez Chávez y Ricardo Fidel Betancourt Hidalgo, bajo mi supervisión.



MSc. Hernán David Ordoñez Calero

DIRECTOR



PhD. Julián Andrés Galindo Losada

CODIRECTOR

AGRADECIMIENTO

Agradezco infinitamente a Dios, por darme la salud, fuerza y sabiduría durante todo este proceso y permitirme obtener un logro más en mi vida.

Agradezco a mis padres que me han acompañado durante todo este camino, siendo un apoyo incondicional y absoluto durante mis estudios y mi vida, que con sus palabras de aliento, cariño y amor me han ayudado a salir adelante.

A mis abuelitos, tíos y primos, que me han apoyado de distintas maneras, permitiéndome culminar exitosamente con una nueva etapa de mi vida.

Agradezco a mis profesores, que, con su esfuerzo y dedicación, me ayudaron a crecer profesionalmente, brindándome nuevas enseñanzas y valores durante toda la carrera.

A mis tutores, Msc Hernán Ordóñez y Phd. Julian Galindo, por todo su apoyo y esfuerzo dedicado, tanto en el aula de clases como en el desarrollo del proyecto.

Muchas gracias.

Christian Benítez Chávez

AGRADECIMIENTO

Agradezco a Dios por darme la oportunidad de cumplir un logro más en mi vida profesional. Agradezco a mi Esposa Elena Mayte Carrera por ser la motivación de cada día para salir a delante y por tu apoyo incondicional en los momentos más difíciles para que nunca me rinda. A mis padres Diego Fernando Betancourt y Verónica Hidalgo por ser el pilar de mi formación personal por enseñarme que detrás de ser un profesional primero debo ser un gran ser humano, gracias por su amor y su apoyo en cada uno de mis sueños y metas.

Agradezco de todo corazón a mi abuelita Martha Nieto y mi abuelito Juan Hidalgo por su amor sin condición, por acompañarme cada día de mi vida enseñándome e inculcando valores para convertirme en la persona que soy hoy. Agradezco a todos mis hermanos y familiares por su ayuda por su apoyo en las diferentes circunstancias de este proceso.

De igual forma agradezco con gratitud a cada uno de mis maestros que con su esfuerzo impartieron sus conocimientos para ser un buen profesional. Y a mis tutores Msc Hernán Ordóñez y Phd. Julian Galindo por su apoyo y acompañamiento en la realización de este proyecto.

Ricardo Betancourt Hidalgo

DEDICATORIA

Dedico este proyecto a mis padres, Mónica Chávez y Gonzalo Benítez, que, gracias a sus esfuerzos, palabras, consejos y amor, me han guiado y acompañado durante todo el transcurso de mi carrera estudiantil hasta la culminación de esta importante etapa de mi vida. También a toda mi familia, por todo el apoyo incondicional que me han brindado en cada momento.

Christian Benítez Chávez

DEDICATORIA

Por mostrarme que en la vida lo más importante es el calor de la familia, que no se necesita tener riquezas para poder ser feliz, por enseñarme a dar el mayor de los esfuerzos cada día y en cada actividad que me acerque a cumplir mis sueños, quiero dedicarte este proyecto y logro abuelito Juan Hidalgo, tu luz y tus enseñanzas las hare resaltar siempre.

Ricardo Betancourt Hidalgo

Tabla de contenido

RESUMEN	1
ABSTRACT	2
1. CAPÍTULO 1: INTRODUCCIÓN	3
1.1. Planteamiento	3
1.2. Problemática	6
1.2.1. Encuestas	6
1.3. Justificación.....	7
1.3.1. Justificación Teórica	7
1.3.2. Justificación Metodológica	9
1.3.3. Justificación Práctica	11
1.4. Objetivos.....	12
1.4.1. Objetivo General	12
1.4.2. Objetivo Específico	12
1.5. Herramientas.....	12
2. CAPÍTULO 2: DESARROLLO DE LA APLICACIÓN	15
2.1. Selección de la muestra	15
2.2. Especificación de requerimientos	16
2.2.1. Historias de usuario épicas.....	16
2.2.2. Historias de Usuario	17
2.2.3. Módulos.....	18
2.2.4. Seguridad.....	19
2.2.5. Roles	20
2.3. Metodología.....	20
2.3.1. SCRUM.....	20
2.3.2. Definición de roles SCRUM	20
2.3.3. Implementación SCRUM	21
2.4. Diseño del sistema	39
2.4.1. Mockups.....	39
2.4.1.4. Módulo Conductor: Página inicio	41
2.4.1.5. Módulo Conductor: Envíos	42
2.4.2. Repositorio Github.....	43
2.4.3. Diseño de la base de datos	43
2.4.4. Arquitectura de la aplicación	47

3. CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS	48
3.1. Implementación de la aplicación	48
3.2. Pruebas	54
3.2.1. <i>Definición de pruebas</i>	54
3.2.2. <i>Pruebas unitarias</i>	54
3.2.3. <i>Pruebas de usabilidad</i>	58
3.2.4. <i>Pruebas de cumplimiento PWA</i>	60
4. CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES	63
4.1. Conclusiones	63
4.2. Recomendaciones	64
5. REFERENCIAS BIBLIOGRÁFICAS	65
6. ANEXOS	69
6.1. ANEXO 1: Encuesta sobre la situación actual empresas	69
6.2. ANEXO 2: Proceso de configuración inicial del proyecto	69
6.3. ANEXO 3: Mockups	69
6.4. ANEXO 4: Pantallas del sistema	69
6.5. ANEXO5: Pruebas unitarias.....	69
6.6. ANEXO 6: Encuesta de usabilidad	70
6.7. ANEXO 7: Historias de Usuario.....	70
6.8. ANEXO 8: Manual de Usuario.....	70
6.9. ANEXO 9: Manual de Instalación	70
6.10. ANEXO 10: Resultados Encuesta de Usabilidad	70

FIGURAS

FIGURA 1. PREGUNTA DE USO DE TECNOLOGÍAS EN LAS MICROEMPRESAS.....	7
FIGURA 2. PREGUNTA SOBRE REALIZAR ENTREGAS A DOMICILIO DURANTE LA PANDEMIA COVID-19.	7
FIGURA 3. PROCESO METODOLOGÍA SCRUM.	10
FIGURA 4. HISTORIAS ÉPICAS.....	17
FIGURA 5. HISTORIA DE USUARIO: REGISTRO PARA EMPRESAS.	18
FIGURA 6. SPRINT 1: BURNDOWN CHART.	23
FIGURA 7. SPRINT 2: BURNDOWN CHART.	27
FIGURA 8. SPRINT 3: BURNDOWN CHART	32
FIGURA 9. SPRINT 4: BURNDOWN CHART.	36
FIGURA 10. SPRINT 5: BURNDOWN CHART.....	39
FIGURA 11. MOCKUP PANTALLA DE INICIO DE SESIÓN.	40
FIGURA 12. MOCKUP PANTALLA DE INICIO ADMINISTRADOR.....	40
FIGURA 13. MOCKUP PANTALLA DE LISTADO DE ENVÍOS.....	41
FIGURA 14. MOCKUP PANTALLA DE INICIO CONDUCTOR.	41
FIGURA 15. MOCKUP PANTALLA DE ENVÍOS CONDUCTOR.....	42
FIGURA 16. MOCKUP PANTALLA DE CLIENTE.	42
FIGURA 17. MODELO DE DATOS CENTRAL.	43
FIGURA 18. MODELO DE DATOS DE CLIENTE.....	43
FIGURA 19. MODELO DE DATOS DE CONDUCTOR.	44
FIGURA 20. MODELO DE DATOS DE DIRECCIÓN.....	44
FIGURA 21. MODELO DE DATOS DE ENVÍO.	45
FIGURA 22. MODELO DE DATOS DE INFORMACIÓN DEL ENVÍO.	45
FIGURA 23. MODELO DE DATOS DE NEGOCIO.	45
FIGURA 24. MODELO DE DATOS DE REGISTRO.....	46
FIGURA 25. MODELO DE DATOS DE VEHÍCULO.	46
FIGURA 26. MODELO DE DATOS DE ZONAS DE COBERTURA.....	46
FIGURA 27. ARQUITECTURA DEL PROYECTO.	47
FIGURA 28. PANTALLA DE LOGIN.	48
FIGURA 29. PANTALLA DE LOGIN (VERSIÓN MÓVIL).....	48
FIGURA 30. PANTALLA DE INICIO ADMINISTRADOR.	49
FIGURA 31. PANTALLA DE INICIO ADMINISTRADOR (VERSIÓN MÓVIL).	49
FIGURA 32. PANTALLA DE LISTADO DE ENVÍOS.	50
FIGURA 33. PANTALLA DE LISTADO DE ENVÍOS (VERSIÓN MÓVIL).....	50
FIGURA 34. PANTALLA DE INICIO CONDUCTOR.....	51
FIGURA 35. PANTALLA DE INICIO CONDUCTOR (VERSIÓN MÓVIL).	51
FIGURA 36. PANTALLA DE ENVÍOS DEL CONDUCTOR.	52
FIGURA 37. PANTALLA DE ENVÍOS DEL CONDUCTOR (VERSIÓN MÓVIL).....	52
FIGURA 38. PANTALLA CLIENTE.	53
FIGURA 39. PANTALLA CLIENTE (VERSIÓN MÓVIL).....	53
FIGURA 40. REPORTE DE PRUEBAS UNITARIAS KARMA.	57
FIGURA 41. RESULTADOS ENCUESTA DE USABILIDAD.....	59
FIGURA 42. RESULTADOS DE LIGHTHOUSE.....	61
FIGURA 43. INSTALADOR APLICACIÓN OWNCOURIER.....	62

TABLAS

TABLA 1. DIFERENCIAS ENTRE APLICACIÓN NATIVA, APLICACIÓN WEB Y PWA.	8
TABLA 2. HERRAMIENTAS Y PROGRAMAS UTILIZADOS.	12
TABLA 3. ROLES SCRUM.....	20
TABLA 4. SPRINT 1: SPRINT PLANNING.	21
TABLA 5. SPRINT 1: PUNTOS DE ESTIMACIÓN.....	21
TABLA 6. SPRINT 1: SPRINT REVIEW.....	22
TABLA 7. SPRINT 2: SPRINT PLANNING.	24
TABLA 8. SPRINT 2: PUNTOS DE ESTIMACIÓN.....	26
TABLA 9. SPRINT 2: SPRINT REVIEW.....	26
TABLA 10. SPRINT 3: SPRINT BACKLOG.	28
TABLA 11. SPRINT 4: PUNTOS DE ESTIMACIÓN.....	30
TABLA 12. SPRINT 3: SPRINT REVIEW.....	30
TABLA 13. SPRINT 4: SPRINT PLANNING.	33
TABLA 14. SPRINT 4: PUNTOS DE ESTIMACIÓN.....	34
TABLA 15. SPRINT 4: SPRINT REVIEW.....	35
TABLA 16. SPRINT 5: SPRINT PLANNING.	37
TABLA 17. SPRINT 5: PUNTOS DE ESTIMACIÓN.....	38
TABLA 18. SPRINT 5: SPRINT REVIEW.....	38
TABLA 19. PRUEBA UNITARIA PARA COMPONENTE EXISTENTE.	54
TABLA 20. PRUEBA UNITARIA PARA FORMULARIO CONDUCTOR.	55
TABLA 21. PRUEBA UNITARIA PARA FORMATO CORRECTO DE CÉDULA.	56
TABLA 22. PRUEBA UNITARIA PARA FORMATO INCORRECTO DE CÉDULA.....	56
TABLA 23. PRUEBA UNITARIA PARA ENVÍO EN ZONA DE COBERTURA.	56
TABLA 24. ESCALA DE RESPUESTA EN ENCUESTA DE USABILIDAD.....	59

RESUMEN

La pandemia del COVID-19 tuvo un efecto negativo sobre varias de las microempresas en algunos lugares de Quito, las cuales tuvieron que optar por otros mecanismos para no perder sus ventas y clientes, la mayoría de estas microempresas tuvieron la necesidad por realizar entregas a domicilio como una solución viable ante la crisis de la pandemia.

El presente proyecto, mediante una encuesta y análisis inicial de la situación actual de las microempresas en la ciudad de Quito se determina los requerimientos necesarios para la implementación de un sistema que permita la agilización de la logística interna de las microempresas. A continuación, se desarrolla una PWA para la implementación de un sistema de gestión de couriers para las microempresas, que por medio de la metodología ágil SCRUM, la cual brinda una mejor organización en el proyecto y permite realizar entregas de valor por cada sprint. De igual manera, mediante el uso de herramientas y librerías externas como Angular, Git, Jira, Lighthouse, Karma, Jasmine, etc., se facilitó el desarrollo de la aplicación, logrando cumplir cada uno de los objetivos.

Finalmente se realizan pruebas unitarias y de usabilidad, donde cada una de estas pruebas cumplieron con su objetivo esperado y de igual manera, se realizó una encuesta de usabilidad hacia varios usuarios finales donde se obtuvo un 91.17% de usabilidad del proyecto, llegando a la conclusión de que el sistema puede agilizar y facilitar los procesos internos de distribución de envíos y entregas a domicilio.

Palabras clave: PWA, distribución de envíos, mensajero motorizado, logística, SCRUM

ABSTRACT

The COVID-19 pandemic had a negative effect on several of the micro-businesses in some parts of Quito, which had to opt for other mechanisms in order not to lose their sales and customers, most of these micro-businesses had the need to make home deliveries as a viable solution to the crisis of the pandemic.

The present project, through a survey and initial analysis of the current situation of microenterprises in the city of Quito, determines the necessary requirements for the implementation of a system that allows the streamlining of the internal logistics of microenterprises. Next, a PWA is developed for the implementation of a courier management system for micro-companies, which through the agile SCRUM methodology, which provides a better organization in the project and allows value deliveries for each sprint. Similarly, through the use of external tools and libraries such as Angular, Git, Jira, Lighthouse, Karma, Jasmine, etc., the development of the application was facilitated, achieving each of the objectives.

Finally, unit and usability tests are carried out, where each of these tests fulfilled its expected objective and in the same way, a usability survey was carried out towards several end users where a 91.17% usability of the project was obtained, reaching the conclusion that the system can streamline and facilitate internal processes for the distribution of shipments and home deliveries.

Keywords: PWA, shipping distribution, motorized courier, logistics, SCRUM

1. CAPÍTULO 1: INTRODUCCIÓN

1.1. Planteamiento

Las tecnologías de información y comunicación (TICS) en la actualidad han ido evolucionando de una manera exponencial, a tal punto que, aproximadamente más de dos tercios de la población mundial cuentan con un teléfono móvil [1], de igual manera acceso a estas tecnologías y a internet.

Varias empresas y microempresas, al notar este crecimiento que ha surgido en las TICs en el mundo actual, optan por trasladar su negocio a una plataforma en línea, donde se tenga acceso a la información respectiva de cada empresa y así poder obtener más clientes. La compra en línea se ha vuelto una tendencia y moda en los últimos años, esto ha ido surgiendo ya que evita que las personas se trasladen físicamente al sitio de la empresa. La tendencia indica que las personas prefieren realizar compras en línea, ya que ayuda con los siguientes factores [2]:

- Ahorro de tiempo
- Visualizar y elegir cualquier producto que desee
- Ver opiniones de otras personas sobre el producto o la empresa
- Posibilidad de tener una variedad de precios.

Debido a la pandemia ocurrida en el año 2020 a causa del virus COVID-19, muchas empresas tuvieron que optar por transformar su sistema de negocios. El cambio fue tan radical que las ventas en locales físicos disminuyeron drásticamente, por lo cual para poder sobrevivir y mantener sus operaciones en marcha, muchas empresas y negocios optaron por servicios de entrega a domicilio.

Muchas microempresas por la falta de presupuesto no pueden optar por una plataforma propia para el manejo de su logística o su courier; sin ello, y en las circunstancias actuales, potenciar una empresa o lograr mantenerlo se ha vuelto más complicado.

Los servicios que son dados por medios de couriers, permite que personas, microempresas y empresas puedan dar a conocer sus productos y además de ofrecer entregas en puntos específicos, donde cada cliente pueda escoger su respectivo lugar de entrega. Este tipo de servicio ha ido incrementando de mismo modo que el comercio electrónico (e-commerce).

Por lo general, un sistema de gestión de courier está compuesto por dos partes, La parte del cliente o dispositivo del cliente, donde su principal objetivo es recibir datos de

productos y la parte del servidor, que, en la mayoría de las situaciones, se tiene sobre un proveedor de servicios, el mismo que proporciona mecanismos y herramientas para mantener una relación directa con el cliente [3]. Esto causaría una relación solo con el cliente lo cual hace que la gestión de courier sea mucho más rápido y eficaz. Por lo general el procedimiento que realiza un cliente para realizar un pedido a un sistema es, teniendo en cuenta que es un procedimiento que se repite [3]:

- El cliente tiene algún tipo de gusto por un producto.
- El cliente da la orden para obtener el producto.
- La orden del cliente llega a su destino.
- El cliente recibe su pedido.

Se han obtenido estudios donde se da a conocer que, en varios países en Latinoamérica, dan mucha importancia a este tipo de servicios de couriers [4], ya que permite que los negocios tengan una forma más rápida y sencilla de llegar a más clientes.

Los envíos por medio de couriers hacen uso de mejores tecnologías, ya que ayuda al rastreo de los productos y de igual manera a saber la ubicación del encargado de llevar el producto [4]. Cada microempresa que aplique este servicio de Courier debe contar con un monitoreo y control total de su producto y obtener la mayor cantidad de información del mismo.

Una plataforma que administre el área de Courier de una empresa puede resultar una ayuda para los procesos organizativos al interior del negocio, pero otro factor beneficioso es que puede dar un mayor grado de confianza al cliente al momento de realizar sus compras ya que estos proveen rastreo de su compra. De igual manera los dueños de los negocios o empresas podrán tener un control total de la entrega de sus productos, obteniendo mayor confianza de sus clientes y garantizando fidelidad de los mismos. Por lo tanto, al optar por este tipo de plataformas, varias empresas irían actualizándose tecnológicamente y en un futuro poder tener ya estable su negocio tanto físicamente como virtualmente.

Normalmente existen dos formas de acceder a una aplicación, ya sea web o móvil, esto puede ser por medio de un navegador web o por medio de la descarga de aplicaciones nativas, pero hay que tener en cuenta que estas formas tienen sus limitaciones y desventajas, como es el caso de ocupar almacenamiento interno del dispositivo móvil o de tener acceso a internet para poder ingresar a un sitio web [5].

Esta es la razón por la cual Google en 2015 [6], crea una solución a las desventajas dichas anteriormente, llamadas aplicaciones web progresivas, donde hace una

combinación de aplicaciones web y móviles, dando una mejor experiencia de usuario [5], consiguiendo un funcionamiento bastante similar a una aplicación móvil nativa.

En sí, es una aplicación web que es creada con tecnologías web la cual permite instalarla o no en un dispositivo móvil, dependiendo del gusto del usuario y de igual manera, no necesita de internet para su uso, ya que esta puede ser usada sin necesidad de tener una conexión a internet, ya que cuando el usuario tenga internet está generará actualizaciones hacia la misma.

Las PWA tienen varias ventajas y características que beneficia al momento de crear aplicaciones web y aplicaciones móviles nativas [6], entre estas se tiene:

- Tiene un costo mejor en base a mantenimientos y actualizaciones.
- Requiere de menos de recursos de almacenamiento en dispositivos móviles.
- Tiene protocolos seguros para su navegación.
- Permite tener un mejor posicionamiento SEO.
- Fáciles de compartir.
- Se puede configurar para que estén disponibles en acceso rápido o pantalla de inicio.
- Permite tener un acceso sin conexión a internet.
- Se puede configurar para que reciba notificaciones sobre alguna actividad realizada en la aplicación (push notifications).
- Acceso simple, facilidad de uso para usuarios o clientes. Genera ventaja competitiva.
- No es necesario publicarlas en tiendas de aplicaciones para su descarga.
- Tienen un rápido tiempo de carga y navegación.
- Diseño responsivo. Se adapta a cualquier tamaño de dispositivo.

Una PWA, propuesta de desarrollo de este proyecto, permitiría crear y desarrollar una aplicación web y móvil para la gestión de couriers que puede tener una microempresa con el fin de facilitar la logística de entregas a domicilio que tienen las microempresas con sus clientes, permitiendo agilizar este tipo de procesos y manteniendo una mejor gestión.

Esta aplicación, mediante sus módulos, tendrá un enfoque total al cliente y al administrador o dueño de la microempresa, permitiendo que el administrador aplique una logística completa a la entrega de sus productos mediante los couriers y de igual manera que tenga una visión total de como el producto es entregado y trasladado.

Por otra parte, el cliente podrá ver el recorrido de su pedido y confirmar el estado de su entrega, donde esto, de igual manera, es una ayuda para el administrador, donde podrá

verificar que toda la logística de couriers está bien implementada en su negocio y de igual manera no exista problemas en la entrega de productos.

1.2. Problemática

En base a la encuesta lanzada por ITC (Centro de Comercio Internacional) [41] sobre el impacto que causó la pandemia a las empresas del mundo, se realizó una encuesta hacia tres microempresas de la ciudad de Quito, donde se pudo saber la situación actual y previa a la pandemia de cada una de ellas, conociendo cuales fueron los cambios internos de logística necesarios que tuvieron que implementar para ganar y retener clientes. De igual manera se pudo conocer ventajas que se tendría al implementar tecnología y automatizar procesos que cada empresa tiene bajo su administración.

1.2.1. Encuestas

Las preguntas y sus respectivos resultados de las encuestas se pueden ver en el ANEXO 1. Estas fueron realizadas hacia las distintas empresas. La encuesta fue realizada en base a ITC, estas fueron las siguientes:

- Tipo de empresa
- ¿Su negocio realizaba o realiza entregas a domicilio?
- ¿Cómo evolucionó su negocio durante la crisis de la pandemia del COVID-19?
- ¿Antes de la pandemia utilizaba una plataforma para ventas a Domicilio?
- ¿Cómo sus clientes adquirían sus productos?
- ¿Cuántas ventas a Domicilio realizaban?
- ¿Durante la pandemia del COVID-19 usted opto en realizar entregas a domicilio?
- ¿Su negocio puede adaptarse a entregas a domicilio?
- ¿Optaría por tener una plataforma que administre sus envíos a Domicilio?
- ¿Actualmente hace uso de tecnologías para operar en su negocio?
- ¿Qué beneficios o ventajas esperaría con un sistema de gestión de courier?
- ¿Cree que un sistema de gestión de couriers podría agilizar y aumentar las ventas? ¿si, no y por qué?

Con los resultados obtenidos, se puede evidenciar que las microempresas en la ciudad de Quito tuvieron un cambio total con la llegada de la pandemia sucedida desde el año 2019 hasta el momento. Ya que sus ventas disminuyeron y como se muestra

en la Figura 1, tuvieron que optar por el uso de herramientas tecnológicas para continuar con su negocio y no perder clientes.

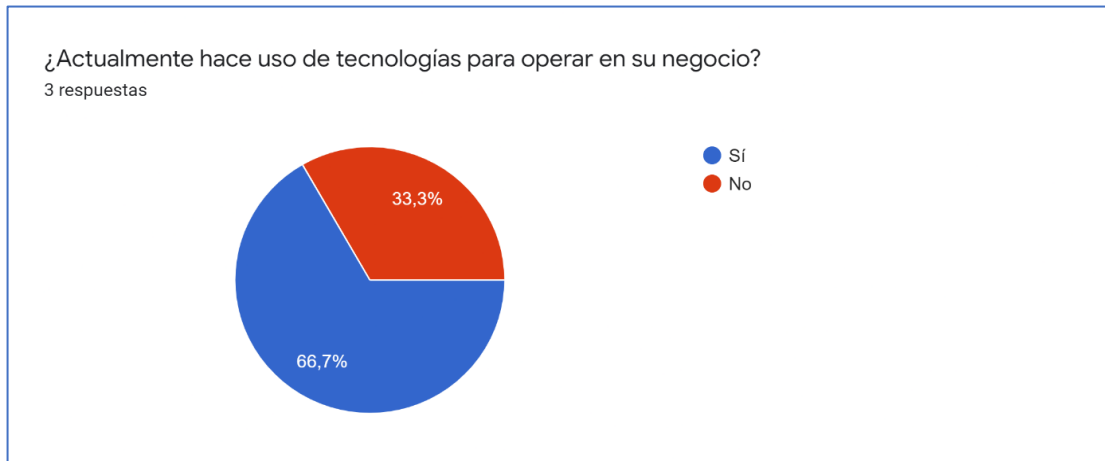


Figura 1. Pregunta de uso de tecnologías en las microempresas.

Además, como se puede ver en la Figura 2, estas empresas han optado por realizar entregas a domicilio de sus productos, dando a conocer que un sistema de gestión de couriers podría ser una ayuda total en el incremento del negocio y poder llegar a más clientes tanto dentro como fuera de la ciudad de Quito.



Figura 2. Pregunta sobre realizar entregas a domicilio durante la pandemia COVID-19.

1.3. Justificación

1.3.1. Justificación Teórica

Las microempresas han optado en cambiar su modelo de negocio ante la pandemia ocurrida en el año 2020 por el COVID-19, muchas de estas han incrementado varios procesos entre ellos de logística y courier para poder ser competitivos dentro del mercado y llegar a ofrecer sus productos a más clientes, esta es una de las razones

por la que se recomienda ver a la entrega a domicilio como un negocio a futuro y no solo para situaciones de emergencia [7].

El manejo de courier es un nuevo proceso que se está implementando en las microempresas para poder enviar sus productos a domicilio de una forma rápida y segura [8]. Esta es una manera de evitar que los clientes tengan que trasladarse a los locales y realizar sus compras. Por esta razón una aplicación de gestión de courier automatizara estos procesos y con una PWA las microempresas podrán reducir el costo de desarrollo e implementación de esta aplicación.

Las aplicaciones web progresivas (PWA) es una tecnología para el desarrollo de aplicaciones web con ciertas características que le hacen ver como una aplicación nativa tanto para teléfonos, tabletas, o computadores. Entre algunas características que se puede implementar dentro de una PWA son notificaciones, funcionamiento sin conexión a Internet [9]. Generalmente el desarrollo de una PWA tiene un costo menor que una aplicación nativa por la cual es una buena opción para microempresas o emprendimientos.

A continuación, se mostrará qué diferencias existe entre una aplicación nativa, una aplicación web y una PWA [5]:

Tabla 1. Diferencias entre aplicación nativa, aplicación web y PWA.

	Aplicación nativa	Aplicación web	PWA
Instalación	Descargar desde App Store o Play Store	No necesita instalación	Un click para añadir a la página de inicio del móvil
Actualizaciones	Poner en la tienda para que pueda ser descargada por el usuario	Actualizaciones inmediatas	Actualizaciones instantáneas
Tamaño	Ocupa más espacio de almacenamiento. Tarda en descargar	Pequeño y rápido	Pequeño y rápido
Acceso fuera de línea	Disponible	No disponible	Se necesita un uso en línea y después ya puede usarse fuera de línea

	Aplicación nativa	Aplicación web	PWA
Experiencia de usuario	Es bueno cuando tiene buen diseño	Igual que una PWA	Confuso por la disponibilidad de doble menú
Notificaciones push	Si	Si (tercerizado)	Si (Solo Android)
Descubrimiento	Mejorar optimización	No requiere	Buena. Optimización SEO para aparecer en resultados de búsqueda

1.3.2. Justificación Metodológica

Para el desarrollo del sistema de gestión planteado, se ha optado por usar las tecnologías dadas por las PWA, debido a que es una tecnología que permite el desarrollo de aplicaciones web con características adicionales que le permiten verse como si fuera una aplicación nativa.

En un ciclo de vida Ágil se va a utilizar iteraciones, repeticiones de un proceso con el objetivo de cumplir el resultado deseado. Donde cada resultado de una iteración sirve como punto de partida para la siguiente iteración [10]. También da un conjunto de buenas prácticas para el trabajo colaborativo y en equipo con la finalidad de cumplir el objetivo deseado.

De igual manera se optó por usar la metodología Ágil Scrum, ya que es una metodología que permite trabajar por medio de iteraciones (Sprints) y así tener una mejora continua de la aplicación [11]. Esta metodología será de mucha ayuda ya que se tendrá bien claro los requisitos para el desarrollo de la planificación y de igual modo los tiempos de desarrollo que se deberán de considerar para el mismo.

1.3.2.1. Scrum

Scrum es un marco ágil y ligero, este proporciona procesos para la gestión y control de desarrollo de software y productos. Scrum fue diseñado para [33]:

- Aumentar la velocidad del desarrollo.
- Alinear los lemas individuales y de las organizaciones.
- Definir una cultura centrada en el desempeño.

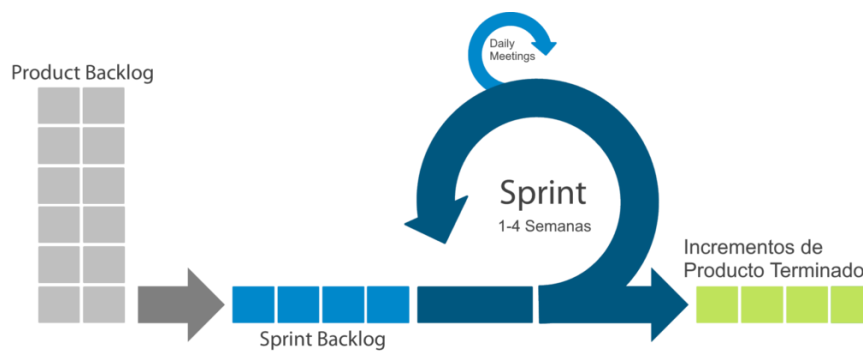


Figura 3. Proceso metodología SCRUM.

1.3.2.1.1. Artefactos Scrum

- **Product Backlog**

Es una lista de requisitos que determina el product owner, estos requisitos se denominan historias de usuario.

- **Sprint Backlog**

Es la documentación de todos los requisitos que se van a trabajar en el *sprint actual*.

- **Sprints**

Los Sprints son los artefactos más pequeños de scrum que trata de un pequeño equipo con tareas asignadas con una duración de 1 a 3 semanas. El objetivo de cada sprint es entregar un producto potencial viable.

- **Historias de Usuario Épicas**

Las historias de usuario épicas es un conjunto de trabajo grande que puede dividirse en tareas específicas denominadas “historias de usuario” [34].

- **Historias de Usuario**

Una historia de usuario es una explicación escrita de una función de software escrita desde la perspectiva del usuario final. Su propósito es articular cómo proporcionará una función de software valor al cliente [35].

- **Meetings**

En scrum se tiene varias reuniones que se realizan durante la duración de todo un sprint. Estas reuniones ayudan a los desarrolladores a entender el

progreso de la evolución del proyecto, facilitando dudas, inconvenientes o problemas de este [36]. Además de una planificación inicial como una conclusión de todo lo realizado durante el sprint.

Las reuniones que se tiene en scrum son [36]:

- ***Sprint planning***: Es una reunión que se realiza al inicio de cada sprint, donde todos los miembros del equipo deben estar presentes. En esta reunión se aclaran todos los temas a realizarse durante el sprint y la respectiva duración que tendrá cada desarrollador en terminar lo asignado.
- ***Daily***: Estas reuniones diarias, por lo general tiene una duración aproximadamente de 15 min dependiendo del tamaño del equipo. Como su nombre lo dice, son reuniones diarias que se realiza con todo el equipo, por lo general cada integrante debe responder a las siguientes preguntas:
 - ¿Qué se hizo ayer?*
 - ¿Qué se hará hoy?*
 - ¿Existe algún impedimento o bloqueante?*
- ***Sprint review***: Esta reunión por lo general se la realiza al final de cada sprint con el objetivo de ver el resultado final y funcionalidad lograda durante la ejecución del sprint.
- ***Sprint retrospective***: Este tipo de reuniones por lo general son planificadas ya que el único propósito de este tipo de reuniones es para revisar y aclarar lo que salió bien y mal durante el sprint. También brinda una ayuda para realizar cualquier mejora necesaria para el desarrollador o el equipo en general.

1.3.3. Justificación Práctica

El presente trabajo se realizará para facilitar a las microempresas a tener un mejor control sobre la gestión de couriers y los procesos de entrega a domicilio que tienen cada una. De esta manera cada empresa podrá monitorear y administrar el control y seguimiento de las entregas de sus productos.

La aplicación se la desarrollará empleando las tecnologías que brinda las PWA, para así poder generar aplicaciones web y móviles de forma nativa. Tener una aplicación

web y móvil que pueda gestionar los couriers y procesos de entrega a domicilio ayudará a tener una ventaja competitiva sobre las demás empresas.

1.4. Objetivos

1.4.1. Objetivo General



Desarrollar una aplicación de gestión de couriers motorizados mediante el uso de Progressive Web Application para microempresas en la ciudad de Quito.

1.4.2. Objetivo Específico

- Analizar la arquitectura, herramientas y requisitos necesarios para el desarrollo de cada módulo de la PWA.
- Diseñar e implementar pruebas de usabilidad mediante el uso de mockups para determinar la comodidad, facilidad o complejidad del producto en base a la experiencia de usuario.
- Aplicar la metodología ágil SCRUM para la planificación y mejor organización en el desarrollo del proyecto.
- Implementar características necesarias dentro de la aplicación para que cumpla con los objetivos del desarrollo de una PWA.
- Probar el funcionamiento del software por medio de pruebas del sistema, garantizando integración y cumplimiento de todas las especificaciones funcionales de cada uno de los componentes y módulos del sistema.

1.5. Herramientas

Tabla 2. Herramientas y programas utilizados.

Herramienta	Descripción	Se utilizo:
 Visual Studio Code Versión: 1.58	<p>Visual Studio Code es un editor de código creado por Microsoft [12]. Es uno de los editores más utilizados por sus pluggins y herramientas disponibles para los usuarios y desarrolladores de software. Brindando depuración de código, versionamiento de código por medio de la herramienta de Git, refactorización de código, entre otras características.</p>	<p>Como IDE de desarrollo del proyecto.</p>
 Angular Versión: 11	<p>Angular es un framework que actualmente es muy demandado debido a que se enfoca en el patrón de diseño MVC (Modelo-Vista-Controlador), logrando facilitar al momento de desarrollo y pruebas. Además, este framework es uno de los mejores para la creación de aplicaciones de una sola página (SPAs) [13].</p>	<p>Como Framework de desarrollo del proyecto</p>

Herramienta	Descripción	Se utiliza:
<p>Angular PWA</p>  <p>Versión: 11.0.1</p>	<p>Las PWA (Progressive Web App) permite crear aplicaciones mediante programación dirigida a la web y de igual manera poderla usar como aplicación nativa.</p> <p>Una de las características que ha sorprendido de una PWA es que también se puede configurar y desarrollar para un funcionamiento dentro de un dispositivo móvil mediante su respectiva instalación [17].</p>	<p>Como una extensión para la configuración de una PWA con Angular.</p>
<p>Angular Material</p>  <p>Versión 11.0.3</p>	<p>Angular Material es una librería de estilos creada por Angular, misma que tiene como base Material Design y se adapta perfectamente al framework de Angular. Esta librería posee características que le hace una de las mejores librerías de diseño para el desarrollo web [18].</p>	<p>Como librería de estilos de componentes del Frontend</p>
<p>Flexbox Grid</p>  <p>Versión: 6.3.1</p>	<p>Flexbox Grid es una librería de CSS que utiliza como base Flexbox. Este es un método de diseño que permite mezclar elementos de una página web y ordenarlos por medio de filas y columnas [21].</p>	<p>Como librería de CSS para el grid del Frontend</p>
<p>Service workers</p>  <p>Versión: 11.0.1</p>	<p>Un service worker es una secuencia de comandos ejecutados por el navegador en segundo plano. Se trata de un fichero JavaScript que continúa ejecutándose, aunque el sitio web esté cerrado [15]. Las principales funciones que tiene un service worker son: sincronización de tareas, mecanismos de almacenamiento, interpretación de solicitudes, etc</p>	<p>Como configuración para la implementación de una PWA</p>
<p>Leaflet maps</p>  <p>Versión: 1.7.1</p>	<p>Es una biblioteca que se ha ido utilizando cada día más por la variedad de funciones de mapeo que tiene a su disposición. Leaflet se destaca por poseer simplicidad, rendimiento y una facilidad de uso, tanto para los desarrolladores como para los usuarios [23]. Su API brinda una simplicidad en el manejo de sus componentes al igual que sus paquetes de mapview [22].</p>	<p>Como librería de mapas.</p>
<p>Firebase SDK</p>  <p>Versión: 8.0</p>	<p>Firebase, creado por Google, es una plataforma digital que facilita un mejor y rápido desarrollo de aplicaciones web y móviles [26]. Tiene como principal objetivo la mejora de rendimiento de apps donde se puede implementar diferentes funcionalidades, garantizando una mejor seguridad, más manejable, fácil uso para usuarios, etc. [24].</p>	<p>Como plataforma de servicios backend como base de datos, autenticación, hosting y cloud functions.</p>

Herramienta	Descripción	Se utiliza:
Invision Studio  Versión: 2.0	Una aplicación de escritorio para crear y editar trabajos de diseño, para diseñar aplicaciones y sitios receptivos de principio a fin [27]. Posee varias herramientas para el diseño de pantallas tanto para dispositivos móviles como de escritorio, además, tiene la capacidad de insertar animaciones y transiciones.	Como herramienta de diseño de mockups
Jasmine  Versión: 3.6.0	Jasmine es un framework para la implementación de pruebas unitarias en proyectos con JavaScript. Las características principales de este framework es su sintaxis clara, funciones que facilitan al desarrollador y cada prueba de usuario es una función.	Como framework de pruebas unitarias
Karma  Versión: 5.1.0	Karma js es un test runner, un lanzador de test, desarrollado por el equipo de Angular, que nos permite automatizar algunas tareas de los frames de tests, como jasmine [28].	Como test runner de las pruebas unitarias
Git  Versión: 2.20.1	Git es el sistema de control de versiones mediante una arquitectura distribuida, es un ejemplo de DVCS (sistema de control de versiones distribuido, por sus siglas en inglés) [29]. También tiene en cuenta el rendimiento, la seguridad y la flexibilidad.	Como sistema de versionamiento
Github  Versión: 2.9.0	Es un Servicio en la nube que aloja a GIT como controlador de versiones. Permite a los desarrolladores realizar cambios en proyectos compartidos y mantiene un seguimiento de los detalles de su progreso [30].	Como proveedor del sistema de versionamiento
Lighthouse  Versión: 7.2.0	Lighthouse es una herramienta automatizada open-source, que busca mejorar la calidad de las páginas web [31]. Con lighthouse podemos hacer pruebas de performance, accesibilidad, Progressive web apps PWA, SEO.	Como tester de funcionalidades enfocadas a PWA
Jira Software 	Es una gama de productos diseñada para ayudar a equipos en la gestión del trabajo, uno de los principales usos es para el desarrollo de software ágil [32].	Como software de gestión de la metodología ágil SCRUM

2. CAPÍTULO 2: DESARROLLO DE LA APLICACIÓN

2.1. Selección de la muestra

El presente proyecto tiene como finalidad automatizar los procesos de gestión de couriers en las microempresas de Quito. El sistema permitiría al administrador o dueño de la empresa poder administrar la información de esta, lo que incluye clientes, envíos, conductores, vehículos y su respectiva lógica para la asignación y ruta de cada uno de los pedidos.

El uso del sistema se dará mediante vía web, pero se debe de considerar que al hacer uso de services workers, este podrá ser usado con o sin internet, ya que toda su información será almacenada en caché y al momento de que el usuario se conecte nuevamente a la red, la información del sistema se guardará automáticamente. El sistema presentará hacia el usuario una interfaz amigable, entendible y adaptable, facilitando su uso y comprensión de este.

Una de las maneras de obtener los requerimientos necesarios para el sistema se hizo el análisis y preguntas a diferentes empresas dentro de la ciudad de Quito, verificando su estado actual y la logística que cada una de estas empresas aplica para realizar sus entregas o pedidos. Entre las empresas seleccionadas se tiene:

- **Club Catálogos**

Microempresa que se dedica a la distribución de ropa, calzado, productos naturales, productos del hogar y tecnológicos en la ciudad de Quito. Con 11 años en el mercado donde los tres últimos modernizó su sistema de ventas para poder adaptarse a los nuevos cambios tecnológicos, una de sus fortalezas es el manejo de las redes sociales donde por medio de estas realiza hoy en día el 100% de sus ventas. Club catálogos además tiene un sistema de socios vendedores que se los inscribe de forma online y por la cual realiza envíos de material de trabajo periódicamente para que estos puedan realizar sus ventas al cliente final.

- **Coati's Coffee Shop**

Empresa dedicada a la producción de café de especialidad con orígenes en Vilcabamba, su producción de café posee un lineamiento orgánico sin la utilización de pesticidas en sus cosechas. Poseen una plaza física en el centro norte de la ciudad de Quito, en la plaza Artigas, donde distribuye sus productos, servicios y especialmente una experiencia cafetera.

Esta empresa distribuye sus productos en su local físico y bajo pedido en forma a domicilio, por lo cual en los últimos meses está modernizando sus herramientas para ofrecer a sus clientes la experiencia cafetera sin que ellos tengan que salir de sus hogares.

- **Empresa privada – Médico Veterinario Zootecnista**

Microempresa dedicada a la venta y distribución de productos ganaderos dentro y fuera de la ciudad de Quito, además de brindar servicios médicos veterinarios zootecnistas. Es una microempresa que lleva alrededor de 5 años de funcionamiento con el mismo esquema de trabajo. La mayor parte de sus productos son entregados a domicilio del cliente o el punto de encuentro que el cliente mencione, de igual manera, al realizar servicios médicos, se debe de acercar al lugar de domicilio o destino del cliente.

Actualmente esta microempresa hace la gestión de distribución de productos por medio de la aplicación de Whatsapp ya que permite una comunicación directa con el cliente, lo cual garantiza un buen servicio y una entrega de los productos segura.

2.2. Especificación de requerimientos

2.2.1. Historias de usuario épicas

Se ha organizado la distribución del desarrollo en 5 historias épicas que concuerdan con los módulos de la aplicación. Cada Épica tiene un grupo de tareas en común que se fue describiendo en historia de usuario para cumplir el desarrollo de cada módulo. Las demás historias épicas se podrán observar en el ANEXO 7.

Epic		
+	PPLGDCMMQ-15 Módulo de Empresa	
■	PPLGDCMMQ-16 Registro para empresas	TAREAS POR... 88
■	PPLGDCMMQ-17 Login para empresas	TAREAS POR...
■	PPLGDCMMQ-18 Perfil Empresas	TAREAS POR...
■	PPLGDCMMQ-36 Zonas de Cobertura	TAREAS POR...
+	PPLGDCMMQ-21 Módulo de Envío de pedidos	
■	PPLGDCMMQ-19 Registro de Envío de pedidos	TAREAS POR...
■	PPLGDCMMQ-22 Listado de Envíos	TAREAS POR...
+	PPLGDCMMQ-23 Módulo de Motorizados	
■	PPLGDCMMQ-25 Registro de Conductores	TAREAS POR...
■	PPLGDCMMQ-28 Login Conductores	TAREAS POR...
■	PPLGDCMMQ-26 Listado de Conductores	TAREAS POR...
■	PPLGDCMMQ-33 Registro de Motos	TAREAS POR...
■	PPLGDCMMQ-29 Asignación de Envíos	TAREAS POR...
+	PPLGDCMMQ-24 Módulo Clientes	
■	PPLGDCMMQ-27 Rastreo de Envíos	TAREAS POR...
+	PPLGDCMMQ-30 Configuración de PWA	
■	PPLGDCMMQ-37 Configuración Service Workers	TAREAS POR...
■	PPLGDCMMQ-38 Configuración Manifest	TAREAS POR...
■	PPLGDCMMQ-31 Notificaciones a Conductores	TAREAS POR...

Figura 4. Historias épicas.

2.2.2. Historias de Usuario

Las historias de usuario realizadas en el presente proyecto se encuentran en el ANEXO 7. Dónde cada historia se ha detallado la descripción de la historia, sus respectivos criterios de aceptación y en caso de que sea necesario se le ha añadido el modelo de datos respectivo o necesario para el cumplimiento de la historia.

Registro para empresas

Adjuntar Añadir una incidencia secundaria Vincular incidencia

Descripción

Como representante de la Empresa X **Deseo** registrar mi empresa **Para** poder utilizar la aplicación de gestión de Courier.

Criterios de Aceptación;

- Validar que todos los datos requeridos estén llenos.
- Validar que el correo ingresado este en un formato de correo válido.
- Verificar que los campos de contraseña y verificar contraseña tengan el valor idéntico.
- Generar mensajes de éxito y fallo dependiendo de la acción realizada una vez que el proceso de registro haya finalizado.

Modelo

```
1 export interface RegistroNegocio {
2     nombreEmpresa: string;
3     correoAdmin: string;
4     celular: string;
5     password?: string;
6     verificarPass?: string;
7     uidPhone?: string;
8     uidAdmin?: string;
9     $key?: string;
10 }
```

Figura 5. Historia de Usuario: Registro para empresas.

2.2.3. Módulos

2.2.3.1. Sección registro / login

Esta sección del sistema hace referencia al ingreso hacia el mismo. Por lo tanto, aquí se tendrá la forma de ingreso y login al sistema, con su respectiva seguridad para la autenticación del Usuario. Se debe tener en cuenta que el login para ingresar al sistema será el mismo para el Administrador y Conductor, ya que se diferenciará por medio de roles.

2.2.3.2. Empresa

Este módulo tendrá toda la información representativa para el administrador del sistema, como son los envíos, motorizados, envíos y ajustes generales. Este módulo será el encargado de administrar toda la lógica del sistema.

2.2.3.3. Motorizados

En este módulo el conductor podrá observar toda la información sobre sus pedidos asignados y de igual manera, podrá realizar diferentes acciones sobre sus pedidos para así poder saber el estado en el que se encuentra el envío.

2.2.3.4. Envío de pedidos

Este módulo tendrá toda la lógica para la asignación de envíos a los diferentes conductores, dependiendo de su zona de cobertura, estado, peso máximo del envío y cantidad de envíos por conductor.

2.2.3.5. Configuración PWA

En esta sección se tendrá toda la configuración y ajustes necesarios para poder convertir a la aplicación en una PWA y que cumpla con todo lo necesario.

2.2.3.6. Clientes

Este módulo se encargará de mostrar al cliente el estado de su envío. Además de poder ver su propia información, datos del conductor y del envío a entregarse.

2.2.4. Seguridad

- Para ingresar al sistema se tendrá que ingresar correo y contraseña además de seleccionar el rol con el cual se quiere acceder.
- Se podrá hacer registro de varias empresas, que mediante la autenticación de Google estas serán validadas y autorizadas para su registro.
- Solo el administrador podrá realizar acciones como cambio de contraseña y cambio de correo.
- El correo y la cédula del conductor constarán como datos de ingreso para la plataforma en el rol de Conductor.

2.2.5. Roles

El sistema consta de dos principales roles, donde cada uno tiene su correo y contraseña para su respectivo ingreso:

- Administrador
- Conductor

2.3. Metodología

2.3.1. SCRUM

Para el desarrollo de la aplicación se aplicó la metodología ágil SCRUM, donde la estimación de cada historia se la realizó por medio histories points, estos representan a un número de la serie de Fibonacci. [42]

Se consideró que la duración de cada sprint sea de 3 semanas con una duración máxima de 120 horas de trabajo, teniendo en cuenta que no se laboró en fines de semanas y días feriados. Para el desarrollo de cada historia de usuario se realizó una equivalencia donde 1 history point representa 4 horas de trabajo de un desarrollador.

2.3.2. Definición de roles SCRUM

Para el proyecto se tiene los siguientes roles:

Tabla 3. Roles SCRUM.

Scrum rol	Responsable
Desarrolladores	Christian Benítez Ricardo Betancourt
Product Owner	Dueños o administradores de microempresas
Scrum Master	Msc. Hernán Ordoñez PhD. Julián Galindo

2.3.3. Implementación SCRUM

2.3.3.1. Sprint 1

- **Objetivos Sprint**

- Diseñar pruebas para comprobar el correcto funcionamiento del sistema.
- Diseñar mockups para poder tener una base de diseño para la implementación del proyecto.
- Instalar las herramientas y librerías necesarias para el proyecto.

- **Sprint Planning**

Tareas a realizarse durante el sprint:

Tabla 4. Sprint 1: Sprint Planning.

Historia de Usuario	Estimación
Instalación herramientas y librerías	3
Creación y preparación proyecto	3
Diseño de mockups del aplicativo	8
Especificación de pruebas unitarias	8
Especificación de pruebas de usabilidad	8

- **Sprint Review**

Se muestra los puntos estimados que se cumplieron durante el sprint.

Tabla 5. Sprint 1: Puntos de estimación.

Historia de Usuario	Puntos estimados	Puntos completados	Puntos pendientes
Instalación herramientas y librerías	3	3	0
Creación y preparación proyecto	3	3	0
Diseño de mockups del aplicativo	8	8	0
Especificación de pruebas unitarias	8	8	0
Especificación de pruebas de usabilidad	8	8	0

En la Tabla 6 se muestra las tareas cumplidas durante el sprint:

Tabla 6. Sprint 1: Sprint Review.

Nº	Historia de Usuario	Cumplida
1	Instalación herramientas y librerías	SI
2	Creación y preparación proyecto	SI
3	Diseño de mockups del aplicativo	SI
4	Especificación de pruebas unitarias	SI
5	Especificación de pruebas de usabilidad	SI

- **Sprint Retrospective**

En este sprint se hizo la instalación de las herramientas y librerías necesarias para realizar el proyecto. Esa instalación fue realizada sobre cada uno de los computadores de los desarrolladores del proyecto. Se especificó que la forma de trabajo será por medio de la herramienta de Github.

Además, se realizó y diseño los mockups del sistema, teniendo una base del FrontEnd del sistema. Finalmente concluyendo el sprint, se definió las pruebas unitarias y de usabilidad que tendrá que ser validadas por el sistema para completar con su desarrollo.

- **Sprint BurnDown Chart**

A continuación, como se observa en la Figura 6, se puede ver el proceso de desarrollo del Sprint 1.



Figura 6. Sprint 1: Burndown chart.

2.3.3.2. Sprint 2

- **Objetivos Sprint**

- Diseñar y desarrollar los módulos necesarios para que la empresa pueda acceder al inicio de la plataforma.
- Definir los tipos de autenticación que se tendrá en el login de la plataforma para la empresa
- Establecer que campos serán necesarios para poder realizar el registro y el ingreso a la plataforma, tanto de la empresa como del conductor.
- Desarrollar la sección de “Configuración”, para que el administrador de la empresa pueda realizar modificaciones de la información de la misma.
- Permitir al administrador de la empresa poder registrar la información personal de sus conductores mediante un formulario.

- **Sprint Planning**

En la Tabla 7 se detalla las historias de usuario a realizarse:

Tabla 7. Sprint 2: Sprint Planning.

Módulo	Historia de usuario	Estimación	Criterio de aceptación	Tareas
Módulo empresa	Registro para empresas	8	Validar campos requeridos llenos	Crear campos para formulario
				Programar vista de formulario
				Programar validaciones para campos
			Validar formato de correo válido	Programar validación personalizada para correo electrónico
				Aplicar validación de correo al campo respectivo en el formulario
			Validar campo "contraseña" y "verificar contraseña" sean iguales	Programar validación de campos
	Generar mensajes de éxito y fallo	Programar mensajes de error		
		Programar mensajes de éxito		
		Definir momentos para mostrar los respectivos mensajes		
	Login para empresas	5	Verificar credenciales sean de un usuario autorizado	Desarrollar e implementar validaciones de autorización
Validar que sea usuario registrado				
Verificar que campos estén llenos			Programar validaciones para campos obligatorios	
			Programar campos para el respectivo login	
Generar mensajes de éxito y fallo			Programar mensajes de error	
			Programar mensajes de éxito	
	Definir momentos para mostrar los respectivos mensajes			

Módulo	Historia de usuario	Estimación	Criterio de aceptación	Tareas
	Perfil empresas	3	Actualizar cada campo sobre la información de la empresa	Desarrollar formulario con sus respectivos campos Desarrollar validaciones para cada campo Programar botón y su lógica respectiva para la información de la empresa Guardar o actualizar información en la base de datos
Módulo motorizados	Registro de conductores	8	Validar campos llenos	Desarrollar formulario Desarrollar validaciones de cada campo Aplicar validaciones a los campos
			Validar campo cédula	Desarrollar validación personalizada para cédula Aplicar validación
			Validar peso máximo de imágenes subidas	Desarrollar validación para que la imagen no sobrepase 2MB Aplicar validación al momento de guardar Aplicar validación en cada campo de imagen
			Validar formato de correo válido	Programar validación personalizada para correo electrónico Aplicar validación de correo al campo respectivo en el formulario
			Generar mensajes de éxito y fallo	Programar mensajes de error Programar mensajes de éxito Definir momentos para mostrar los respectivos mensajes

- **Sprint Review**

En la Tabla 8 se muestra los puntos estimados que se cumplieron durante el sprint.

Tabla 8. Sprint 2: Puntos de estimación.

Historia de Usuario	Puntos estimados	Puntos completados	Puntos pendientes
Registro para empresas	8	8	0
Login para empresas	5	5	0
Perfil empresas	3	3	0
Registro de conductores	8	8	0

En la Tabla 9 se detalla las tareas completadas durante el sprint:

Tabla 9. Sprint 2: Sprint Review.

N°	Tarea	Cumplida
1	Crear campos para formulario	SI
2	Programar vista de formulario	SI
3	Programar validaciones para campos	SI
4	Programar validación personalizada para correo electrónico	SI
5	Aplicar validación de correo al campo respectivo en el formulario	SI
6	Programar validación de campos	SI
7	Programar mensajes de error	SI
8	Programar mensajes de éxito	SI
9	Definir momentos para mostrar los respectivos mensajes	SI
10	Desarrollar e implementar validaciones de autorización	SI
11	Validar que sea usuario registrado	SI
12	Programar validaciones para campos obligatorios	SI
13	Programar campos para el respectivo login	SI
14	Programar mensajes de error	SI
15	Programar mensajes de éxito	SI
16	Definir momentos para mostrar los respectivos mensajes	SI
17	Desarrollar formulario con sus respectivos campos	SI
18	Desarrollar validaciones para cada campo	SI
19	Programar botón y su lógica respectiva para la información de la empresa	SI
20	Guardar o actualizar información en la base de datos	SI
21	Desarrollar formulario	SI
22	Desarrollar validaciones de cada campo	SI
23	Aplicar validaciones a los campos	SI
24	Desarrollar validación personalizada para cédula	SI
25	Aplicar validación	SI
26	Desarrollar validación para que la imagen no sobrepase 2MB	SI
27	Aplicar validación al momento de guardar	SI
28	Aplicar validación en cada campo de imagen	SI
29	Programar validación personalizada para correo electrónico	SI
30	Aplicar validación de correo al campo respectivo en el formulario	SI
31	Programar mensajes de error	SI
32	Programar mensajes de éxito	SI
33	Definir momentos para mostrar los respectivos mensajes	SI

- **Sprint Retrospective**

El resultado final del segundo sprint se puede observar el registro de la empresa ya finalizado, permitiendo ingresar el nombre de la empresa y el número de celular, donde finalmente tendrá que validar el captcha e iniciar sesión.

Esto hará que le llegue un mensaje de texto con un código al celular y tendrá que ingresar el código correctamente para finalizar para pasar al ingreso del correo y la contraseña, los cuales servirán para su inicio de sesión.

- **Sprint BurnDown Chart**

A continuación, como se observa en la Figura 7, se puede ver el proceso de desarrollo del Sprint 2.

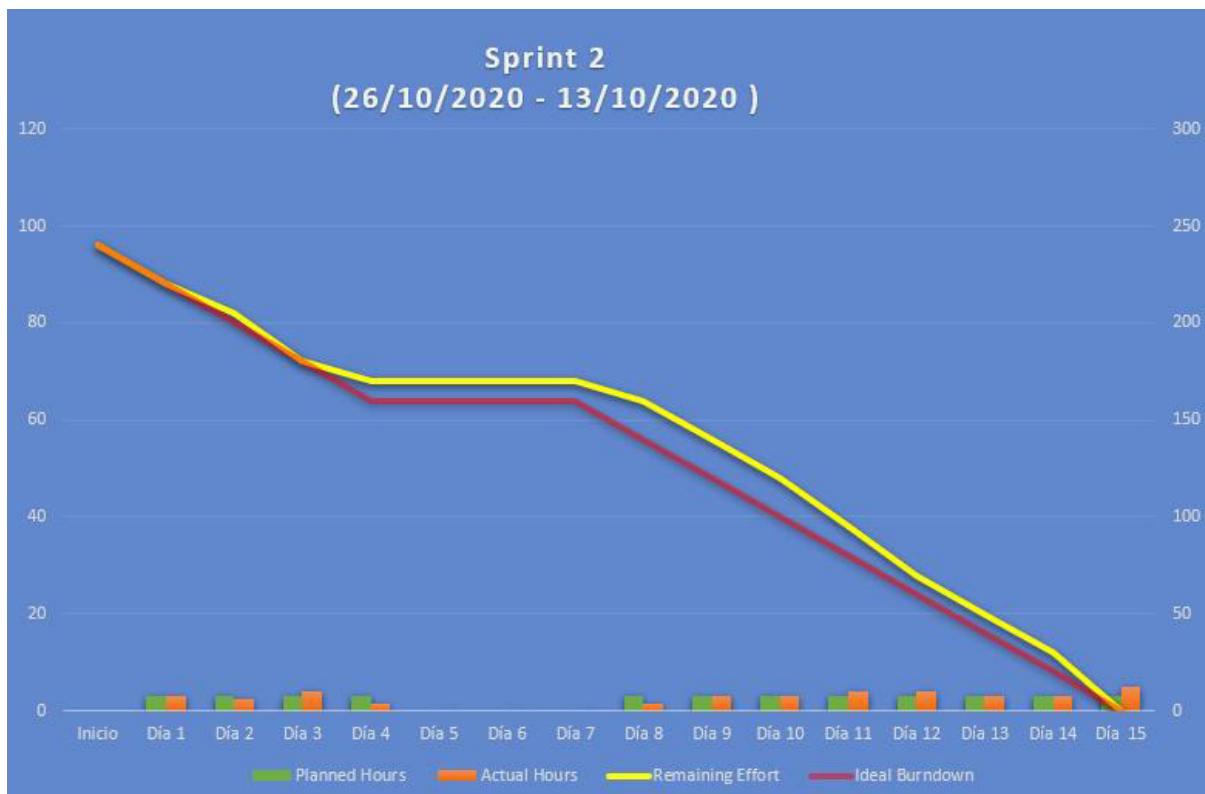


Figura 7. Sprint 2: Burndown chart.

2.3.3.3. Sprint 3

- **Objetivos Sprint**

- Diseñar y desarrollar los módulos necesarios para que el conductor pueda acceder al inicio de la plataforma.
- Desarrollar una sección en el administrador para que se pueda ver el listado de todos los conductores que pertenecen a la empresa.
- Diseñar y desarrollar el módulo para realizar el registro de todas las motos pertenecientes a la empresa.
- Desarrollar y plantear toda la lógica necesaria para la asignación de envíos a cada conductor.

- **Sprint Planning**

En la Tabla 10 se detallan las historias y tareas a realizarse durante el sprint.

Tabla 10. Sprint 3: Sprint Backlog.

Módulo	Historia de Usuario	Estimación	Criterio de aceptación	Tareas
Módulo de motorizados	Login Conductores	5	Validar campos llenos	Desarrollar formulario
				Desarrollar validaciones de cada campo
				Aplicar validaciones a los campos
			Verificar credenciales sean de un usuario autorizado	Desarrollar e implementar validaciones de autorización
				Validar que sea usuario registrado
				Generar mensajes de éxito y fallo
	Programar mensajes de éxito			
	Definir momentos para mostrar los respectivos mensajes			
	Listado de Conductores	2	Listar conductores que pertenecen a la empresa	Obtener información de la base de datos
				Desarrollar tabla e interfaz para mostrar datos del conductor
Asignar datos obtenidos a cada columna de la tabla del conductor				

Módulo	Historia de Usuario	Estimación	Criterio de aceptación	Tareas
	Registro de motos	8	Tener opciones para editar, eliminar y visualizar información del conductor	Definir botones para las diferentes acciones
				Programar la lógica respectiva para cada botón
			Validar campos llenos	Desarrollar formulario
				Desarrollar validaciones de cada campo
				Aplicar validaciones a los campos
			Validar peso máximo de imágenes subidas	Desarrollar validación para que la imagen no sobrepase 2MB
				Aplicar validación al momento de guardar
				Aplicar validación en cada campo de imagen
			Generar mensajes de éxito y fallo	Programar mensajes de error
				Programar mensajes de éxito
				Definir momentos para mostrar los respectivos mensajes
			Guardar información	Agregar botón para guardar datos
				Programas lógica respectiva para el guardado de información
				Programar validaciones dentro de la lógica del botón
			Asignación de envíos	13
Programar la lógica para la carga máxima de cada motorizado				
Validar cantidad máxima	Agregar campo para la cantidad máxima			
	Programar la lógica para la cantidad máxima de envíos por motorizado			
Asignar fecha de envío	Agregar campo para selecciona fecha			

Módulo	Historia de Usuario	Estimación	Criterio de aceptación	Tareas
			Asignar envíos	Obtener información de la base de datos
				Programar lógica para asignar los envíos
				Guardar información en la base de datos

- **Sprint Review**

En la Tabla 11 se muestra los puntos estimados que se cumplieron durante el sprint.

Tabla 11. Sprint 4: Puntos de estimación.

Historia de Usuario	Puntos estimados	Puntos completados	Puntos pendientes
Registro para empresas	5	5	0
Login para empresas	2	2	0
Perfil empresas	8	8	0
Registro de conductores	13	13	0

En la Tabla 12 se detalla el cumplimiento de las diferentes actividades realizadas durante el sprint.

Tabla 12. Sprint 3: Sprint Review.

N°	Tarea	Cumplida
1	Desarrollar formulario	SI
2	Desarrollar validaciones de cada campo	SI
3	Aplicar validaciones a los campos	SI
4	Desarrollar e implementar validaciones de autorización	SI
5	Validar que sea usuario registrado	SI
6	Programar mensajes de error	SI
7	Programar mensajes de éxito	SI
8	Definir momentos para mostrar los respectivos mensajes	SI
9	Obtener información de la base de datos	SI
10	Desarrollar tabla e interfaz para mostrar datos del conductor	SI
11	Asignar datos obtenidos a cada columna de la tabla del conductor	SI
12	Definir botones para las diferentes acciones	SI
13	Programar la lógica respectiva para cada botón	SI
14	Desarrollar formulario	SI
15	Desarrollar validaciones de cada campo	SI
16	Aplicar validaciones a los campos	SI
17	Desarrollar validación para que la imagen no sobrepase 2MB	SI
18	Aplicar validación al momento de guardar	SI
19	Aplicar validación en cada campo de imagen	SI
20	Programar mensajes de error	SI
21	Programar mensajes de éxito	SI
22	Definir momentos para mostrar los respectivos mensajes	SI

N°	Tarea	Cumplida
23	Agregar botón para guardar datos	SI
24	Programas lógica respectiva para el guardado de información	SI
25	Programar validaciones dentro de la lógica del botón	SI
26	Agregar campo para la carga máxima	SI
27	Programar la lógica para la carga máxima de cada motorizado	SI
28	Agregar campo para la cantidad máxima	SI
29	Programar la lógica para la cantidad máxima de envíos por motorizado	SI
30	Agregar campo para selecciona fecha	SI
31	Obtener información de la base de datos	SI
32	Programar lógica para asignar los envíos	SI
33	Guardar información en la base de datos	SI

- **Sprint Retrospective**

Al igual que la empresa, el conductor deberá seleccionar el rol “Conductor” e ingresar su correo y contraseña para completar con su inicio de sesión. Además de dar una funcionalidad a la empresa permitiendo poder observar en un listado todos los envíos registrados.

Al finalizar el sprint, el administrador de la empresa podrá registrar los vehículos que se tendrá disponibles dentro de la misma. Finalmente se realizó toda la logística necesaria para poder asignar los diferentes pedidos a cada conductor.

Nota: el correo y contraseña del conductor serán asignados y registrados al momento que el administrador de la empresa registre al conductor. Por el momento la contraseña del conductor será su cédula con la cual fue registrado.

- **Sprint BurnDown Chart**

A continuación, como se observa en la Figura 8, se puede ver el proceso de desarrollo del Sprint 3.

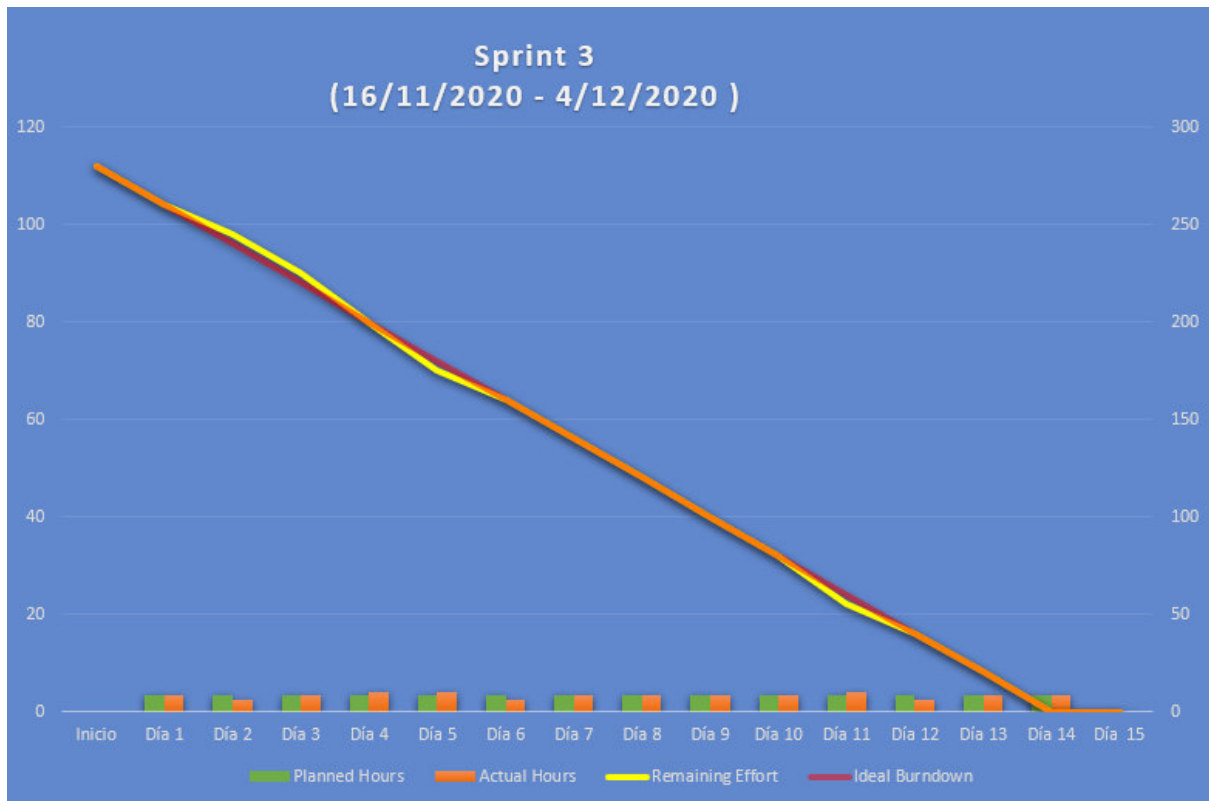


Figura 8. Sprint 3: Burndown chart

2.3.3.4. Sprint 4

- **Objetivos Sprint**

- Permitir al administrador poder registrar nuevos envíos, para así posteriormente asignar a su respectivo conductor.
- Poder observar todos los envíos registrados.
- Brindar al cliente la funcionalidad de poder observar el estado en el que se encuentra el envío.
- Mediante notificaciones, mantener informado al conductor cuando haya sido asignado a un envío y de igual manera dar información al cliente sobre el pedido.

- **Sprint Planning**

En la Tabla 13 se detalla las historias y tareas a realizarse durante el sprint.

Tabla 13. Sprint 4: Sprint Planning.

Módulo	Historia de usuario	Estimación	Criterio de aceptación	Tareas
Módulo de envíos	Registro de envío de pedidos	8	Actualizar cada campo sobre la información del envío	Desarrollar formulario con sus respectivos campos
				Desarrollar validaciones para cada campo
				Programar botón y su lógica respectiva para guardar la información del envío
				Guardar o actualizar información en la base de datos
			Validar que los campos obligatorios estén llenos	Desarrollar formulario
				Desarrollar validaciones de cada campo
	Aplicar validaciones a los campos			
	Generar mensajes de éxito y fallo	Programar mensajes de error		
		Programar mensajes de éxito		
		Definir momentos para mostrar los respectivos mensajes		
Listado de envíos	3	Listar envíos registrados por la empresa	Obtener información de la base de datos	
			Desarrollar tabla e interfaz para mostrar los datos de los envíos registrados	
		Tener opciones para editar, eliminar y visualizar información del conductor	Asignar datos obtenidos a cada columna de la tabla del conductor	
			Definir botones para las diferentes acciones	
Módulo de clientes	Rastreo de envíos	13	Verificar posición del envío	Programar la lógica respectiva para cada botón
				Desarrollar e implementar mapas para el rastreo
				Configurar información de mapa para obtener los datos de las ubicaciones
				Guardar información en la base de datos
				Desarrollar algoritmo para calcular las distancias del envío

Módulo	Historia de usuario	Estimación	Criterio de aceptación	Tareas
Configuración PWA	Notificaciones a conductores	3	Enviar notificaciones push al conductor con información del envío	Añadir la configuración de las notificaciones push al momento de asignar un envío
				Implementar botón para que el conductor pueda suscribirse
				Configurar notificaciones por parte del conductor para recibir las notificaciones
				Desarrollar la lógica para la suscripción de las notificaciones push
	Notificaciones a clientes	3	Permitir al cliente recibir notificaciones	Implementar cambio de estado cuando el conductor empiece el envío
				Desarrollar pantalla para visualizar información del envío
Configurar notificaciones en la sección del cliente para ver el estado del envío				

- **Sprint Review**

En la Tabla 14 se muestra los puntos estimados que se cumplieron durante el sprint.

Tabla 14. Sprint 4: Puntos de estimación.

Historia de Usuario	Puntos estimados	Puntos completados	Puntos pendientes
Registro de envío de pedidos	8	8	0
Listado de envíos	3	3	0
Rastreo de envíos	13	13	0
Notificaciones a conductores	3	3	0
Notificaciones a clientes	3	1	2

En la Tabla 15 se detalla el cumplimiento de las diferentes actividades realizadas durante el sprint.

Tabla 15. Sprint 4: Sprint Review.

N°	Tarea	Cumplida
1	Desarrollar formulario con sus respectivos campos	SI
2	Desarrollar validaciones para cada campo	SI
3	Programar botón y su lógica respectiva para guardar la información del envío	SI
4	Guardar o actualizar información en la base de datos	SI
5	Desarrollar formulario	SI
6	Desarrollar validaciones de cada campo	SI
7	Aplicar validaciones a los campos	SI
8	Programar mensajes de error	SI
9	Programar mensajes de éxito	SI
10	Definir momentos para mostrar los respectivos mensajes	SI
11	Obtener información de la base de datos	SI
12	Desarrollar tabla e interfaz para mostrar los datos de los envíos registrados	SI
13	Asignar datos obtenidos a cada columna de la tabla del conductor	SI
14	Definir botones para las diferentes acciones	SI
15	Programar la lógica respectiva para cada botón	SI
16	Desarrollar e implementar mapas para el rastreo	SI
17	Configurar información de mapa para obtener los datos de las ubicaciones	SI
18	Guardar información en la base de datos	SI
19	Desarrollar algoritmo para calcular las distancias del envío	SI
20	Añadir la configuración de las notificaciones push al momento de asignar un envío	SI
21	Implementar botón para que el conductor pueda suscribirse	SI
22	Configurar notificaciones por parte del conductor para recibir las notificaciones	SI
23	Desarrollar la lógica para la suscripción de las notificaciones push	SI
24	Implementar cambio de estado cuando el conductor empiece el envío	SI
25	Desarrollar pantalla para visualizar información del envío	SI
26	Configurar notificaciones en la sección del cliente para ver el estado del envío	SI

- **Sprint Retrospective**

En este sprint se tendrá la funcionalidad para que el administrador de la empresa pueda registrar nuevos envíos al igual que poder visualizar los mismos. De igual manera se brinda una funcionalidad al cliente de poder ver el estado de su envío y toda la información relacionada al mismo. Finalmente se implementará notificaciones push, tanto para el cliente y el conductor, permitiendo una suscripción a los envíos y poder estar al tanto cuando un envío haya sido asignado al conductor y enviado al cliente respectivamente.

Para este sprint quedó pendiente por desarrollar 2 histories points (8 horas) que hace referencia a la historia de usuario de “Notificaciones a clientes”, donde se tendrá una estimación para esta historia en el siguiente sprint. El motivo por el cual no se pudo cumplir con el objetivo del sprint fue porque tomó más tiempo de lo esperado en la implementación del servicio de notificaciones, este desfase se lo puede ver en la Figura 9.

- **Sprint BurnDown Chart**

A continuación, como se observa en la Figura 9, se puede ver el proceso de desarrollo del Sprint 4.

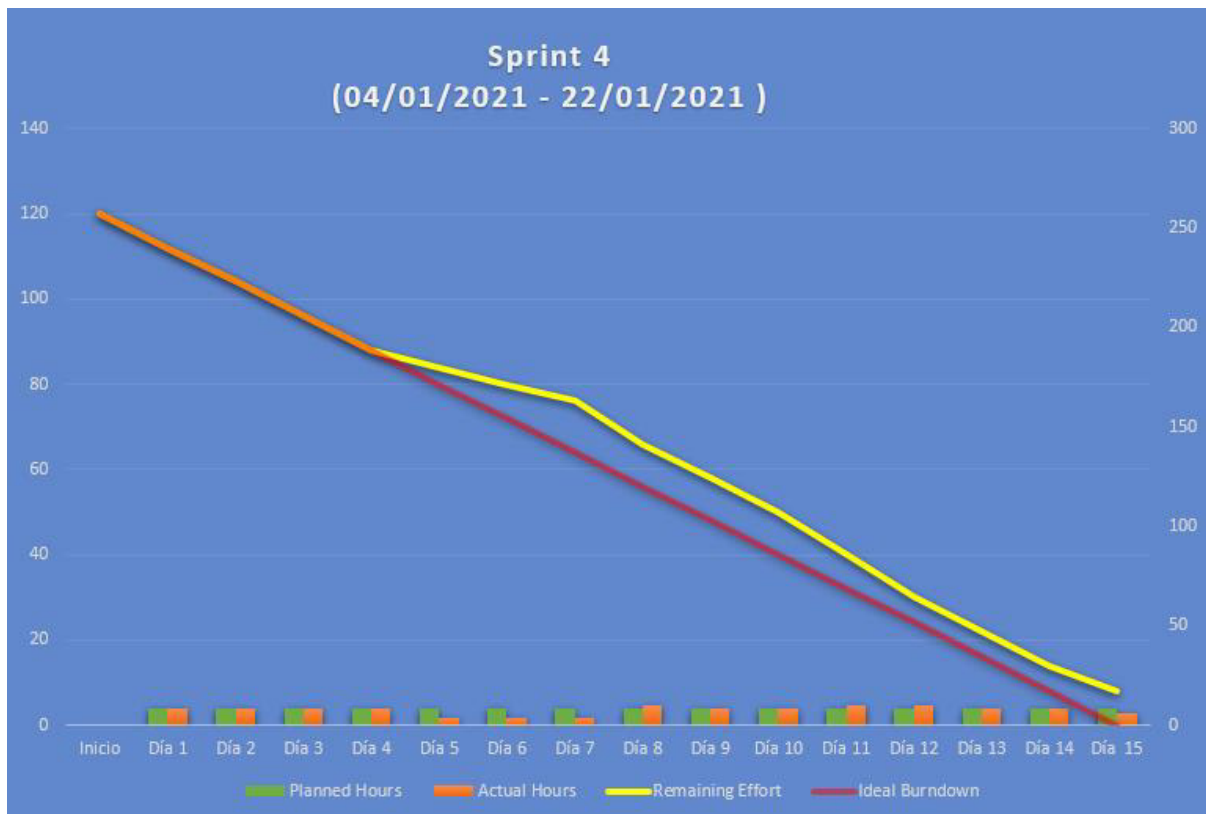


Figura 9. Sprint 4: Burndown chart.

2.3.3.5. Sprint 5

- **Objetivos Sprint**

- Configurar los services workers para tener una mejor funcionalidad de la aplicación y permitir hacer uso de la misma sin necesidad de tener conexión a internet.
- Crear una sección para estableces y crear las zonas de cobertura que tendrá la empresa para poder realizar sus envíos.

- **Sprint Planning**

En la Tabla 16 se detalla las historias y tareas a realizarse durante el sprint.

Tabla 16. Sprint 5: Sprint Planning.

Módulo	Historia de Usuario	Estimación	Criterio de aceptación	Tarea
Configuración PWA	Configuración de service workers	13	Configurar service workers	Programar y configurar service workers para módulo Administrador
				Programar y configurar service workers para módulo Motorizados
				Programar y configurar service workers para módulo Clientes
	Configuración Manifest	2	Configurar archivo manifest	Programar y configurar archivo manifest
Módulo empresa	Zonas de cobertura	13	Configurar mapas leaflet	Instalar librería de leaflet
				Configurar componente para llamar mapas
				Programar pantalla para seleccionar y configurar zonas de cobertura
				Probar coordenadas seleccionadas
Configuración PWA	Notificaciones a clientes	2	Permitir al cliente recibir notificaciones	Implementar cambio de estado cuando el conductor empiece el envío
				Desarrollar pantalla para visualizar información del envío
				Configurar notificaciones en la sección del cliente para ver el estado del envío

- **Sprint Review**

En la Tabla 17 se muestra los puntos estimados que se cumplieron durante el sprint.

Tabla 17. Sprint 5: Puntos de estimación.

Historia de Usuario	Puntos estimados	Puntos completados	Puntos pendientes
Configuración de service workers	13	13	0
Configuración Manifest	2	2	0
Zonas de cobertura	13	13	0
Notificaciones a clientes	2	2	0

En la Tabla 18 se detalla el cumplimiento de las diferentes actividades realizadas durante el sprint.

Tabla 18. Sprint 5: Sprint Review.

Nº	Tarea	Cumplida
1	Programar y configurar service workers para módulo Administrador	SI
2	Programar y configurar service workers para módulo Motorizados	SI
3	Programar y configurar service workers para módulo Clientes	SI
4	Programar y configurar archivo manifest	SI
5	Instalar librería de leaflet	SI
6	Configurar componente para llamar mapas	SI
7	Programar pantalla para seleccionar y configurar zonas de cobertura	SI
8	Probar coordenadas seleccionadas	SI

- **Sprint Retrospective**

Al final del sprint ya se contará con la funcionalidad del proyecto por medio de service workers garantizando una funcionalidad completa con y sin conexión a internet. También se implementó una característica para que el administrador del sistema pueda registrar todas las zonas en donde realizará sus respectivos envíos y a su vez estas zonas poder asignar a los distintos conductores distribuyendo de mejor manera la entrega de productos y envíos.

- **Sprint Burndown chart**

A continuación, como se observa en la Figura 10, se puede ver el proceso de desarrollo del Sprint 5.



Figura 10. Sprint 5: Burndown Chart.

2.4. Diseño del sistema

2.4.1. Mockups

El presente proyecto consta de dos principales módulos (Administrador, Conductor). Dentro de cada módulo existen submódulos donde se especifica la lógica necesaria para cumplir con el objetivo del proyecto.

A continuación, se detallará una perspectiva de las interfaces principales que se tendrá en el proyecto. La muestra completa de todas las interfaces, se encuentran en el Anexo 3.

2.4.1.1. Inicio de Sesión

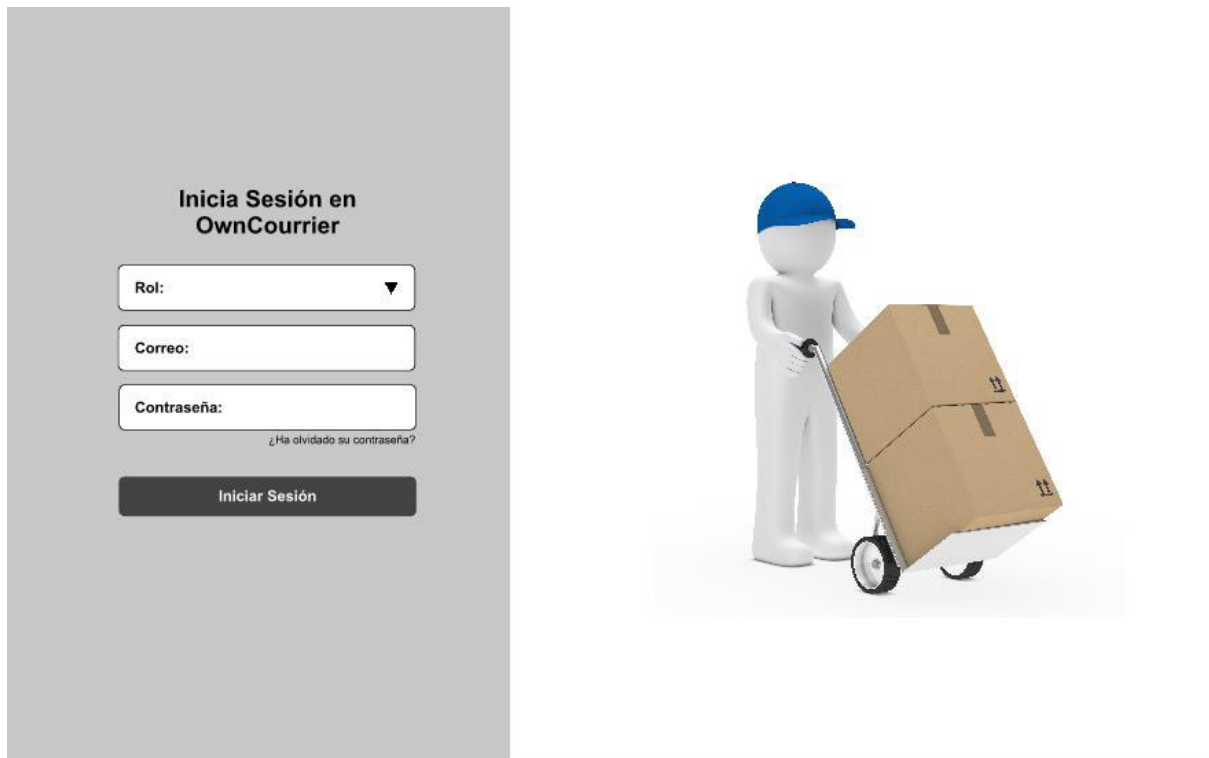


Figura 11. Mockup pantalla de Inicio de Sesión.

2.4.1.2. Módulo Empresa: Página de inicio

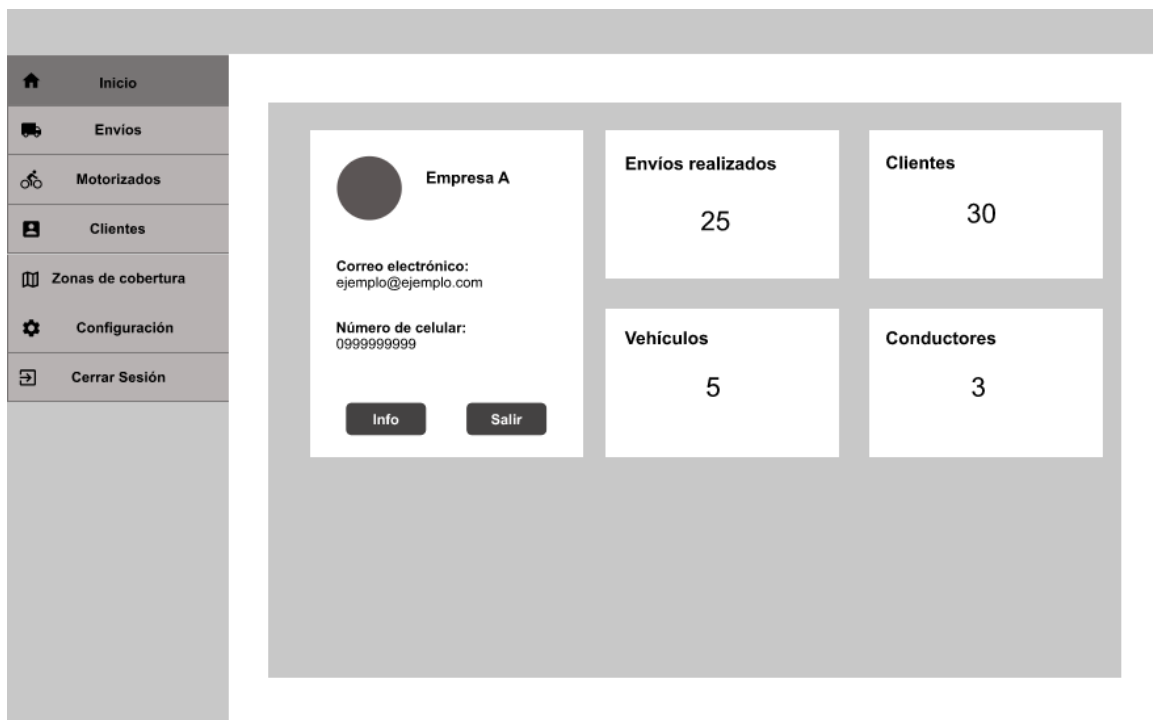


Figura 12. Mockup pantalla de Inicio Administrador.

2.4.1.3. Módulo Empresa: Listado de envíos



Figura 13. Mockup pantalla de Listado de Envíos.

2.4.1.4. Módulo Conductor: Página inicio

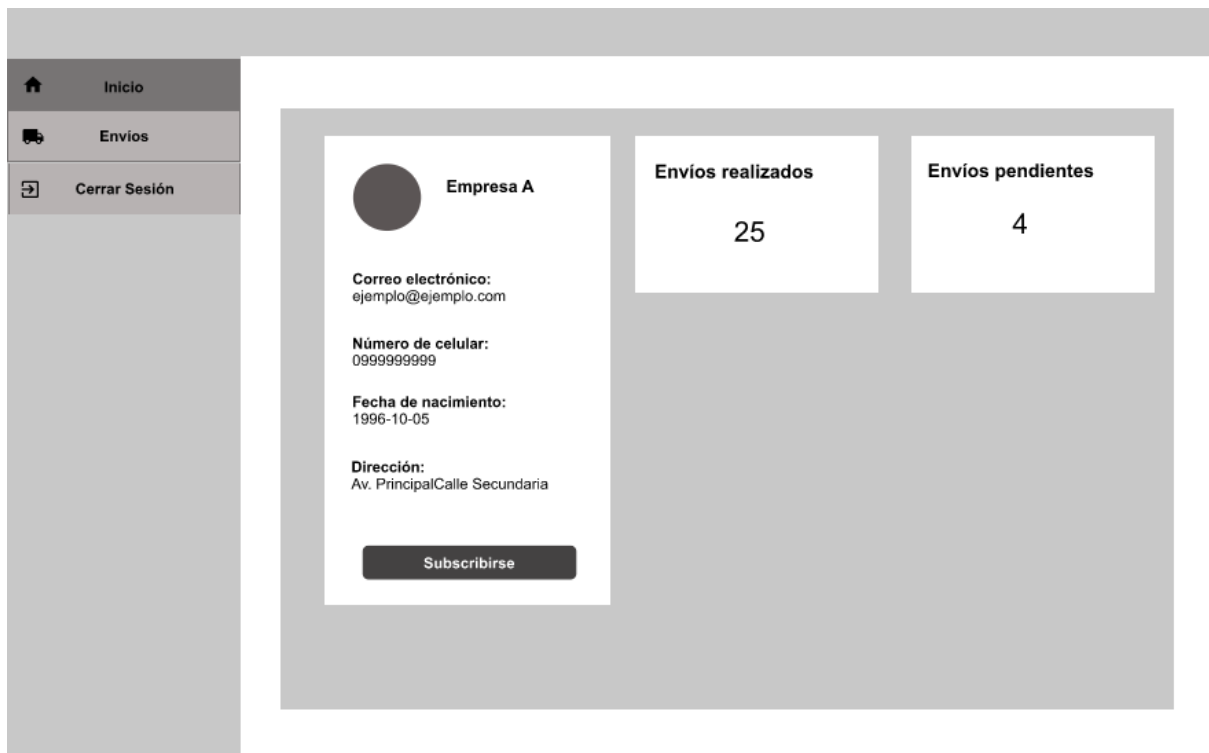


Figura 14. Mockup pantalla de Inicio Conductor.

2.4.1.5. Módulo Conductor: Envíos

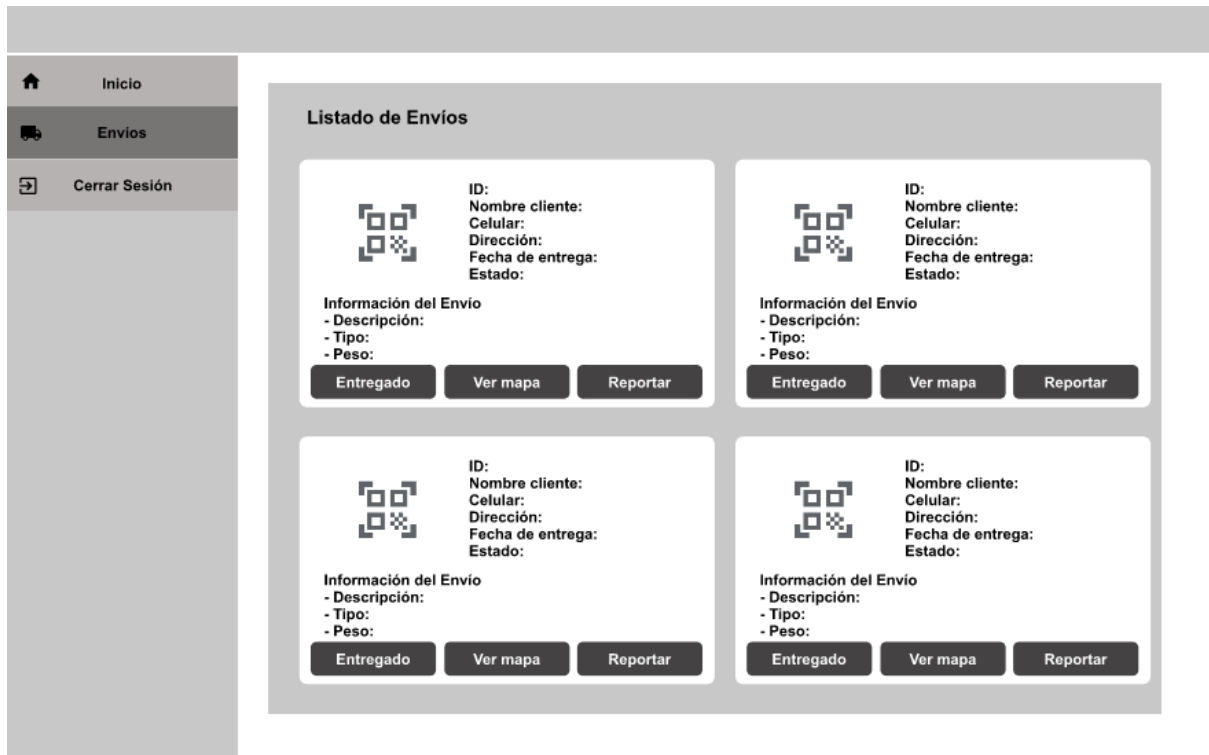


Figura 15. Mockup pantalla de Envíos Conductor.

2.4.1.6. Módulo Clientes



Figura 16. Mockup pantalla de Cliente.

2.4.2. Repositorio Github

El repositorio del proyecto se encuentra en el siguiente link:

<https://github.com/ricardobeta/Tesis-Benitez-Betancourt.git>

Este repositorio consta de las siguientes ramas:

- main
- Desarrollo
- Test

2.4.3. Diseño de la base de datos

En el desarrollo del proyecto se utilizará una base de datos no relacional, para el modelamiento de los datos se realizó en interfaces de Typscript. Cada interfaz está compuesta por las propiedades que serán parte de cada una de las entidades. Junto a Realtime Database estas interfaces se almacenan en formato JSON con las propiedades obligatorias y opcionales que estas tengan, dando el control en el desarrollo en el manejo de datos de cada Entidad.

2.4.3.1. Central

```
export interface Central {  
  latitud: number;  
  longitud: number;  
}
```

Figura 17. Modelo de datos Central.

2.4.3.2. Cliente

```
export interface Cliente {  
  cedula:string;  
  celular:string;  
  correo:string;  
  nombreCompleto:string;  
}
```

Figura 18. Modelo de datos de Cliente.

2.4.3.3. Conductor

```
import { Envio } from "../envio.model";

export interface Conductor {
  cedula: string;
  celular: string;
  correo: string;
  direccion: string;
  fechaNacimiento: string;
  nombreCompleto: string;
  urlFoto?: string;
  urlLicencia?: string;
  pathFoto?: string;
  pathLicencia?: string;
  fileF?: any;
  fileL?: any;
  //user information
  uidConductor: string;
  keyNegocio: string;
  $key?: string;
  estado?: string;
  keyVehiculo?: string;
  keyZona?: string
  //envios Asignado
  envios: Envio[]
}
```

Figura 19. Modelo de datos de Conductor.

2.4.3.4. Dirección

```
export interface Direccion {
  descripcion: string;
  longitud: number;
  latitud: number;
  urlMapa: string;
  zoom: number;
}
```

Figura 20. Modelo de datos de Dirección.

2.4.3.5. Envío

```
import { Cliente } from "../cliente.model";
import { Direccion } from "../direccion.model";
import { InfoEnvio } from "../info-envio.model";

export interface Envio {
  cliente: Cliente;
  direccion: Direccion;
  infoEnvio: InfoEnvio;
  fecha: string;
  estado: string;
  // valores opcionales
  $key?: string;
  keyConductor?: string;
}
```

Figura 21. Modelo de datos de Envío.

2.4.3.6. Información Envío

```
export interface InfoEnvio {
  descripcion: string;
  tipo: string;
  prioridad: string;
  peso: number;
}
```

Figura 22. Modelo de datos de Información del Envío.

2.4.3.7. Negocio

```
export interface Negocio {
  nombreEmpresa: string;
  celular: string;
  correoAdmin: string;
  descripcion: string;
  fileLogo?: any;
  pathLogo?: string;
  urlLogo?: string;
}
```

Figura 23. Modelo de datos de Negocio.

2.4.3.8. Registro

```
export interface RegistroNegocio {
  nombreEmpresa: string;
  correoAdmin: string;
  celular: string;
  password?: string;
  verificarPass?: string;
  uidPhone?: string;
  uidAdmin?: string;
  $key?: string;
}
```

Figura 24. Modelo de datos de Registro.

2.4.3.9. Vehículo

```
export interface Vehiculo {
  marca: string;
  placa: string;
  color: string;
  fileM?: string;
  pathMatricula?: string;
  urlMatricula?: string;
  estado?: string;
  asignado?: boolean;
  $key?: string;
}
```

Figura 25. Modelo de datos de Vehículo.

2.4.3.10. Zona de Cobertura

```
import { Conductor } from "../conductor.model";
import { Envio } from "../envio.model";

export interface ZonaCobertura {
  nombre: string;
  color: string;
  vertices: string;
  $key?: string;
  envios?: Envio[];
  conductores?: Conductor[];
}
```

Figura 26. Modelo de datos de Zonas de Cobertura.

2.4.4. Arquitectura de la aplicación

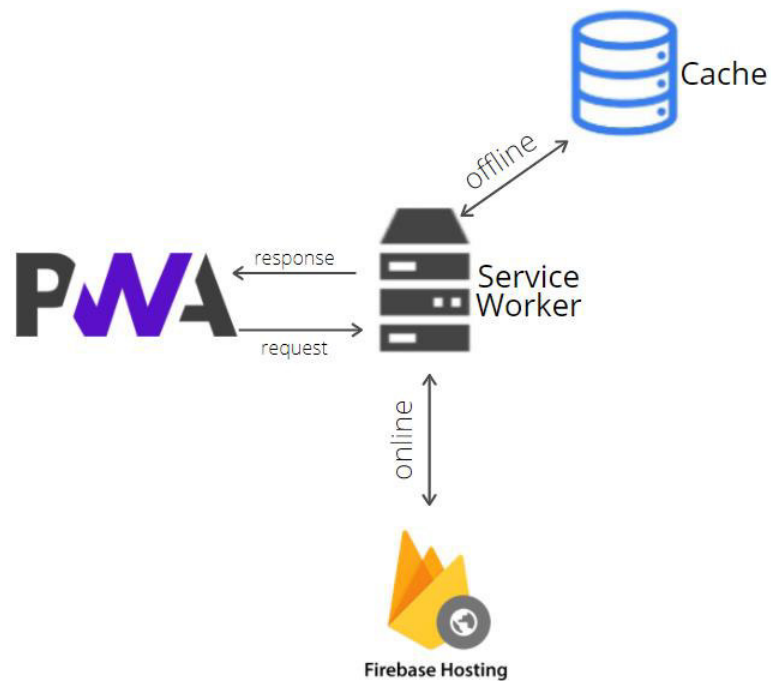


Figura 27. Arquitectura del proyecto.

Como se puede observar la Figura 27 en la arquitectura del proyecto se tiene a la PWA alojada en el servidor de firebase hosting que nos provee una conexión https con un dominio para poder acceder. La aplicación tiene un Service worker configurado con el patrón “primero cache” esta estrategia trata si un recurso está almacenado en caché y disponible sin conexión, devuélvelo primero antes de intentar descargarlo del servidor. Si aún no está en la caché, descárgalo y almacénalo para uso futuro. [38]

3. CAPÍTULO 3: IMPLEMENTACIÓN Y PRUEBAS

3.1. Implementación de la aplicación

A continuación, se mostrará una parte de las interfaces del sistema tanto del Administrador como del Conductor, todas las interfaces se encuentran en el Anexo 4.

- **Login**



Figura 28. Pantalla de Login.

- **Login (Versión Móvil)**



Figura 29. Pantalla de Login (Versión móvil).

- **Módulo Empresa: Página inicio**

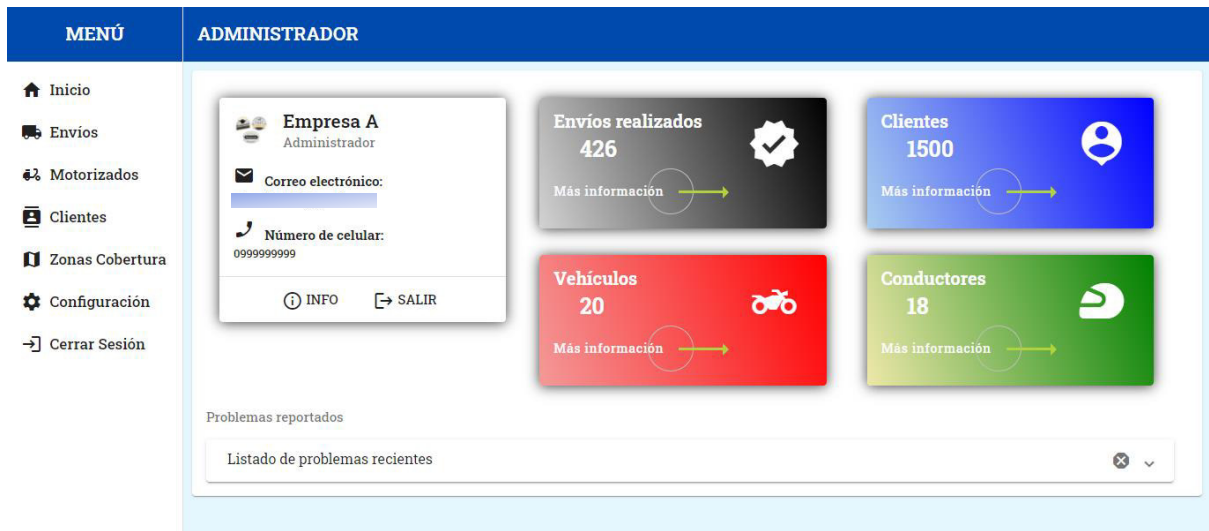


Figura 30. Pantalla de Inicio Administrador.

- **Módulo Empresa: Página inicio (Versión móvil)**



Figura 31. Pantalla de Inicio Administrador (Versión móvil).

- **Módulo Empresa: Listado de envíos**

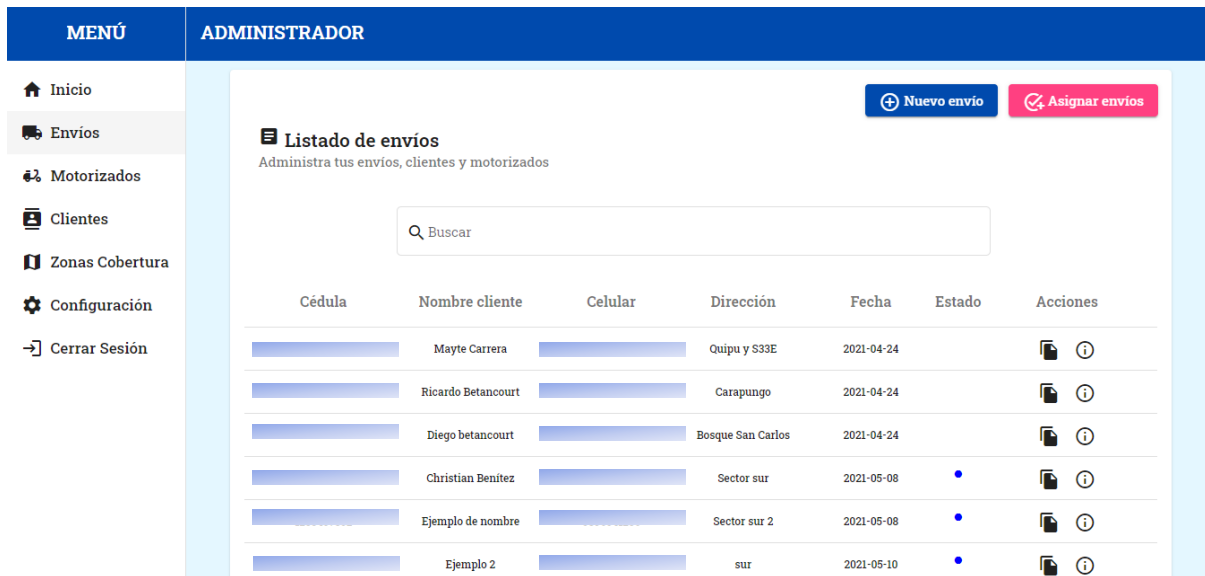


Figura 32. Pantalla de Listado de Envíos.

- **Módulo Empresa: Listado de envíos (Versión móvil)**



Figura 33. Pantalla de Listado de Envíos (Versión móvil).

- **Módulo Conductor: Página inicio**

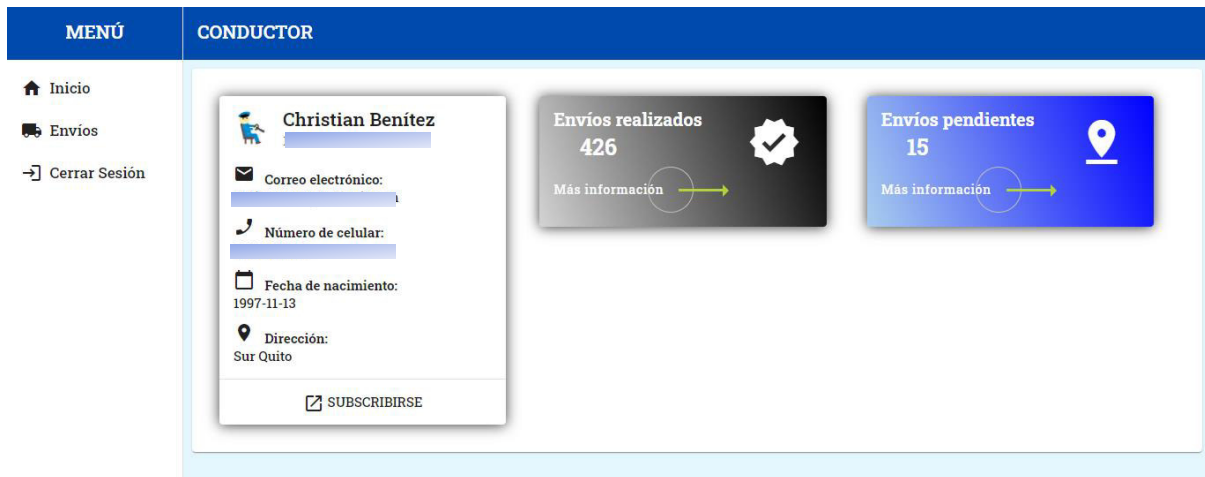


Figura 34. Pantalla de Inicio Conductor.

- **Módulo Conductor: Página inicio (Versión móvil)**



Figura 35. Pantalla de Inicio Conductor (Versión móvil).

- **Módulo Conductor: Envíos**

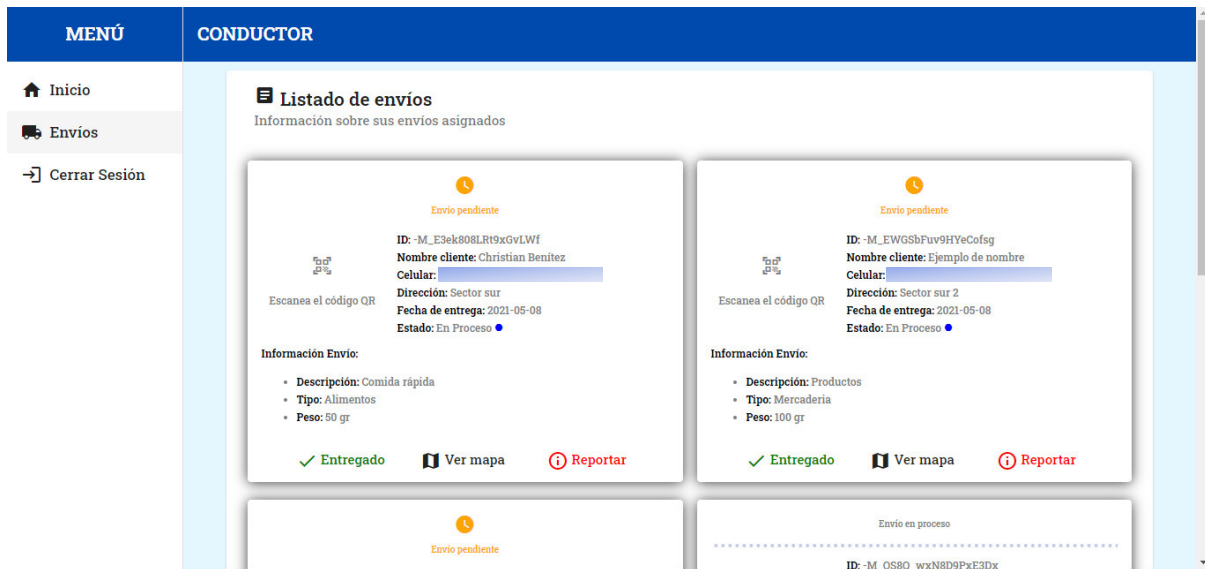


Figura 36. Pantalla de Envíos del Conductor.

- **Módulo Conductor: Envíos (Versión móvil)**

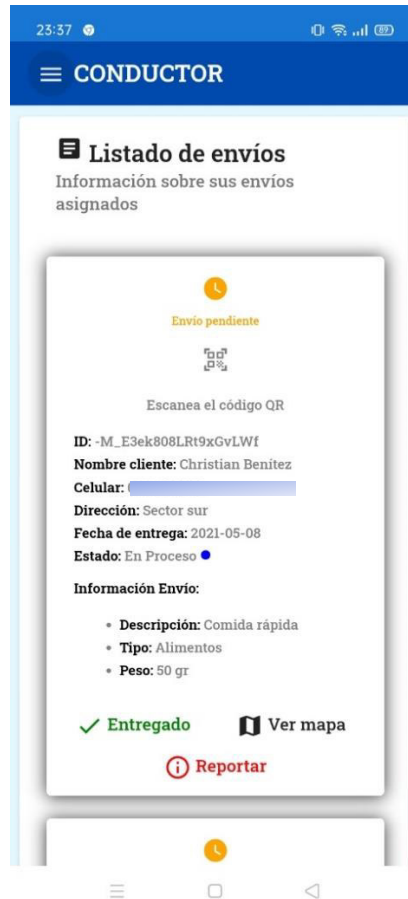


Figura 37. Pantalla de Envíos del Conductor (Versión móvil).

- **Cliente**



Figura 38. Pantalla cliente.

- **Cliente (Versión móvil)**

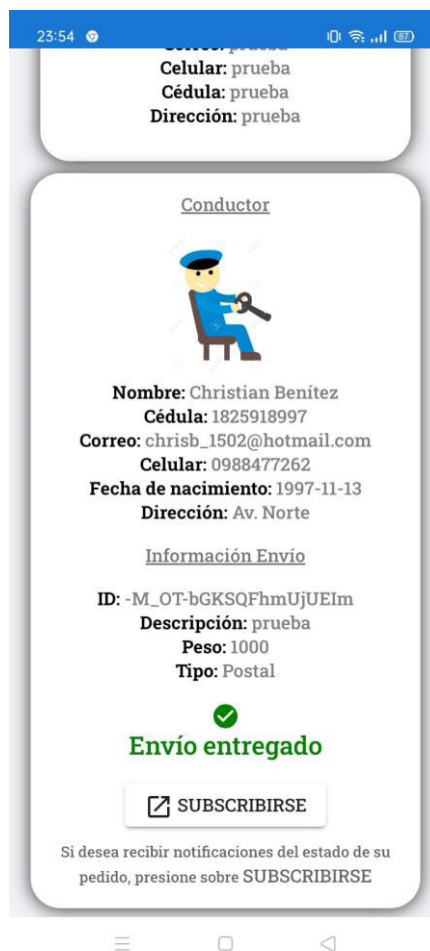


Figura 39. Pantalla cliente (Versión móvil).

3.2. Pruebas

3.2.1. Definición de pruebas

En esta sección se expondrá las pruebas unitarias realizadas hacia el sistema para comprobar que cumpla con su correcto funcionamiento y garantizar la calidad en el producto software que se va a entregar. En el proceso de planificación y desarrollo se ejecutó pruebas unitarias, pruebas de usabilidad y la prueba de cumplimiento de que la aplicación cumpla con todos los parámetros de una PWA.

3.2.2. Pruebas unitarias

Las pruebas unitarias es el proceso de aislar partes de código para el funcionamiento correcto del mismo, garantizando la calidad en la codificación. Al realizar estas pruebas se puede tener los siguientes beneficios [37]:

- Reducir tiempos en solucionar problemas en el proceso de Integración.
- Depurar un módulo independiente sin tener el sistema completo.
- Fomentar la refactorización.
- Ayuda a entender el código.

Entre las pruebas unitarias realizadas tenemos las siguientes.

- La Prueba de creación de componentes se realiza en todos los componentes de negocio que dan valor al producto con el objetivo de comprobar que este se construya correctamente y así evitar un valor indefinido.

Tabla 19. Prueba unitaria para componente existente.

Pruebas Unitarias	OwnCourier
N°	1
Descripción	Existe Componente
Objetivos	<ul style="list-style-type: none">• Probar que el componente se construya correctamente.• Componente no debe ser undefined.
Condiciones	<ul style="list-style-type: none">• Armar un testbed para el renderizado del componente.• Importar los componentes y librerías que no sean de integración a APIs o librerías externas que no se tienen control.
Componentes donde se aplicó la prueba.	<ul style="list-style-type: none">• RegistroComponent• AsignarEnviosComponent• RegistroAdministradorComponent• LoginComponent• VehiculosComponent

Pruebas Unitarias	OwnCourier
	<ul style="list-style-type: none"> • CrearVehiculoComponent • RegistroEnvioComponent • InfoEmpresaComponent • CrearConductorComponent
Resultado esperado	Asegurarse de que valor esperado sea verdadero en un contexto booleano. toBeTruthy = true.
Resultado obtenido	True.

- Pruebas de creación de formularios, cada formulario dentro de la aplicación es importante que se construya con sus campos y sus validadoras iniciales necesarios. El manejo correcto de los datos es lo que se espera que este tipo de prueba controle en cada formulario de negocio que se posee. Se presenta el ejemplo del formulario de creación de conductor.

Tabla 20. Prueba unitaria para formulario Conductor.

Pruebas Unitarias	OwnCourier
N°	12
Descripción	Formulario Crear Conductor Campos Iniciales
Objetivos	<ul style="list-style-type: none"> • Comprobar que el formulario se esté inicializando con los campos que se requieren.
Condiciones	<ul style="list-style-type: none"> • Importar la librería de formularios reactivos al testbed
Componentes donde se aplicó la prueba.	<ul style="list-style-type: none"> • CrearConductorComponent
Resultado esperado	{cedula: "", celular: "", correo: "", direccion: "", fechaNacimiento: "", estado: 'disponible', nombreCompleto: "", fileF: null, fileL: null}
Resultado obtenido	{cedula: "", celular: "", correo: "", direccion: "", fechaNacimiento: "", estado: 'disponible', nombreCompleto: "", fileF: null, fileL: null}

- Pruebas de validaciones de datos, esta prueba da valor al negocio porque valida que los datos que se ingresan a los formularios se encuentren correctamente validados. Entre estas tenemos la validación de una cédula ecuatoriana, validación del formato del correo electrónico. Para las validaciones se realizan dos pruebas, la primera valida una entrada valida que sea exitosa y la segunda una entrada errónea que reconozca que es incorrecta.

Tabla 21. Prueba unitaria para formato correcto de cédula.

Pruebas Unitarias	OwnCourier
N°	20
Descripción	Cédula Formato Correcto
Objetivos	<ul style="list-style-type: none"> Validar que un número de cédula escrito correctamente el input lo acepte como valido.
Condiciones	<ul style="list-style-type: none"> El campo de cédula debe tener el validador de cedulas para el correcto funcionamiento. Se enviará un número de cédula valido (1726079625) y se espera que la prueba devuelva un true.
Componentes donde se aplicó la prueba.	<ul style="list-style-type: none"> RegistroEnvioComponent CrearConductorComponent
Resultado esperado	true
Resultado obtenido	true

Tabla 22. Prueba unitaria para formato incorrecto de cédula.

Pruebas Unitarias	OwnCourier
N°	21
Descripción	Cédula Formato Incorrecto
Objetivos	<ul style="list-style-type: none"> Validar que un número de cédula escrito incorrectamente el input lo acepte como invalido.
Condiciones	<ul style="list-style-type: none"> El campo de cédula debe tener el validador de cedulas para el correcto funcionamiento. Se enviará un número de cédula incorrecta (1726079626) y se espera que la prueba devuelva un false.
Componentes donde se aplicó la prueba.	<ul style="list-style-type: none"> RegistroEnvioComponent CrearConductorComponent
Resultado esperado	false
Resultado obtenido	false

- Pruebas de funciones de negocio, se realizó pruebas unitarias a funciones que son necesarias para la lógica del negocio en este ejemplo se prueba la función que analiza si un envío se encuentra dentro de una zona de cobertura del negocio.

Tabla 23. Prueba unitaria para envío en zona de cobertura.

Pruebas Unitarias	OwnCourier
N°	19
Descripción	Envío dentro de la zona de cobertura
Objetivos	<ul style="list-style-type: none"> Comprobar que un envío se encuentra dentro de una zona de cobertura del negocio.
Condiciones	<ul style="list-style-type: none"> Se genera una zona de cobertura y un envío La función debe evaluar que las coordenadas del envío se encuentren dentro de un punto de la zona de cobertura.

Pruebas Unitarias		OwnCourier
Componentes donde se aplicó la prueba.	<ul style="list-style-type: none"> AsignarEnviosComponent 	
Resultado esperado	true	
Resultado obtenido	true	

3.2.2.1. Resultado de las Pruebas Unitarias

Para la ejecución de todas las pruebas unitarias se realizó mediante el uso del framework de pruebas karma y Jasmine, al ejecutar todas estas pruebas se nos genera el siguiente reporte.



Figura 40. Reporte de pruebas unitarias Karma.

Como se observa en la Figura 40 en la ejecución de las pruebas unitarias con Karma y Jasmine, las 40 pruebas realizadas en cada uno de los componentes tienen un resultado exitoso. Con el cumplimiento de las pruebas unitarias se puede asegurar el correcto funcionamiento de las funciones de negocio, de la construcción de los formularios para el uso correcto del usuario. También se asegura las validaciones de campos sean óptimas para tener datos correctos para poder almacenarlos.

Con la implementación de las pruebas unitarias estas también sirven de documentación del funcionamiento del código ya que cada prueba explica el funcionamiento de cada una de las funciones [37].

Todos los casos de prueba y el reporte de Karma y Jasmine se lo encuentran en el ANEXO 5.

3.2.3. Pruebas de usabilidad

La forma de evaluación del proyecto en base a la usabilidad se ha hecho uso de la Escala de Usabilidad del Sistema (SUS). Un método que permite la evaluación de sistemas web, software, etc. [39], este método de evaluación fue utilizado por su facilidad de puntuación, simplicidad y fiabilidad.

En base al método de evaluación SUS se ha establecido las diferentes escalas de evaluación, donde se le dará a conocer al usuario el valor correspondiente de cada uno de los niveles de evaluación.

La forma que SUS obtiene la usabilidad de un proyecto, aplicación o software es mediante un cálculo que realiza dependiendo de las preguntas realizadas. En el ANEXO 6 se detallan las preguntas realizadas para la encuesta. El cálculo que SUS realiza es el siguiente:

“La contribución de cada punto valdrá entre 0 y 4. Para los puntos 1, 3, 5, 7 y 9, la contribución será la posición de la escala menos 1. Para los puntos 2, 4, 6, 8 y 10, la contribución será 5 menos la posición en la escala. Se multiplica la suma de los resultados por 2.5 para obtener el valor global del SUS. El resultado estará entre 0 y 100”. [39]

En esta ocasión el protocolo que se tomó para realizar las pruebas de usabilidad fue mediante una encuesta en línea, dando la facilidad de poder realizar la prueba y su respectiva encuesta en cualquier momento que el usuario final desee. Esta encuesta fue realizada para un total de 15 usuarios finales.

3.2.3.1. Encuestas y Resultados

En esta sección se muestra los resultados en base a la encuesta de usabilidad realizada hacia las distintas empresas, la cual se encuentra en el ANEXO 6. En esta encuesta se ha utilizado una escala de respuestas dadas en la Tabla 24, esto permitirá identificar la usabilidad completa que tiene el proyecto para los distintos usuarios.

Tabla 24. Escala de respuesta en encuesta de usabilidad.

Respuesta	Valor
Completamente desacuerdo	1
Poco desacuerdo	2
Neutro	3
Parcialmente de acuerdo	4
Completamente de acuerdo	5

En la Figura 41 se puede observar el puntaje obtenido por cada pregunta en base a la escala de respuesta planteada previamente. De igual manera el cálculo respectivo para determinar la usabilidad de la aplicación se lo puede encontrar en el ANEXO 10. Se debe tener en cuenta que para la encuesta se optó que cada usuario hizo las diferentes pruebas tanto con el rol de Administrador como el de Conductor. Además, se recalca que 8 de los usuarios realizaron la respectiva prueba en la web, mientras que los 7 restantes hicieron uso de la aplicación mediante un teléfono móvil.

Encuesta usabilidad de aplicación OwnCourier

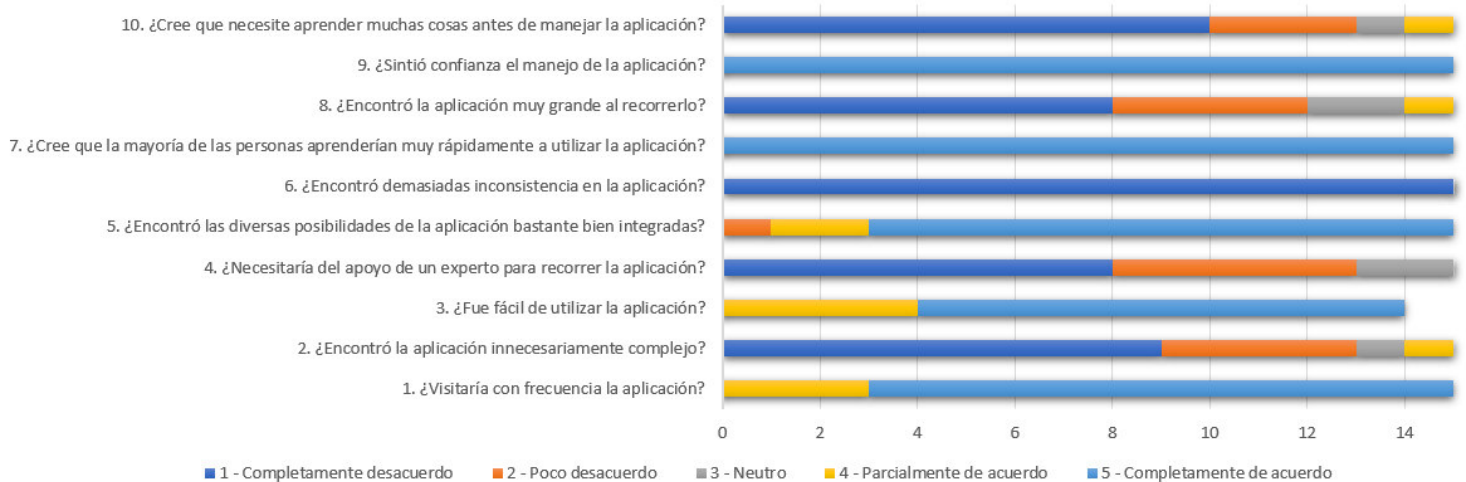


Figura 41. Resultados Encuesta de usabilidad.

En base a los resultados obtenidos de la encuesta, se puede llegar a la conclusión de que el sistema:

- Permite un entendimiento del mismo, llegando a ser intuitivo y fácil de usar.
- Brinda confiabilidad y seguridad sobre toda la información guardada.
- Es fácil de aprender y manejar sin la necesidad de un experto, pero también se puede considerar que un usuario con bajos conocimientos tecnológicos se podría optar por brindar un soporte para el uso de la aplicación.
- Cumple con la experiencia de usuario requerida.
- No muestra errores ni fallos en sus interfaces y lógica.
- En base a los cálculos realizados mediante el método de evaluación SUS se puede decir que cumple con una usabilidad de 91.17%.

Se puede observar que la usabilidad tiene un resultado positivo en los diferentes roles que se tiene en el sistema. Se tiene un 8.73% faltante que da a entender un margen de mejora que necesita la aplicación para una usabilidad completa. Entre las limitaciones encontradas se puede observar que:

- Al tener varias funcionalidades la complejidad aumenta, por lo tanto, los usuarios necesitarán tener conocimientos previos sobre los procesos de courier que se encuentran en la aplicación.
- La aplicación cuenta con varios procesos lo cual hace que tenga una navegación extensa.

3.2.4. Pruebas de cumplimiento PWA

Para que una aplicación sea considerada como una aplicación web progresiva esta debe cumplir con un grupo de funciones integradas en su desarrollo. Para poder hacer una auditoria a la aplicación desarrollada utilizamos la herramienta de Google Lighthouse, esta proporciona un grupo de métricas que se deben cumplir para la creación de una PWA [40].

3.2.4.1. Resultados Lighthouse

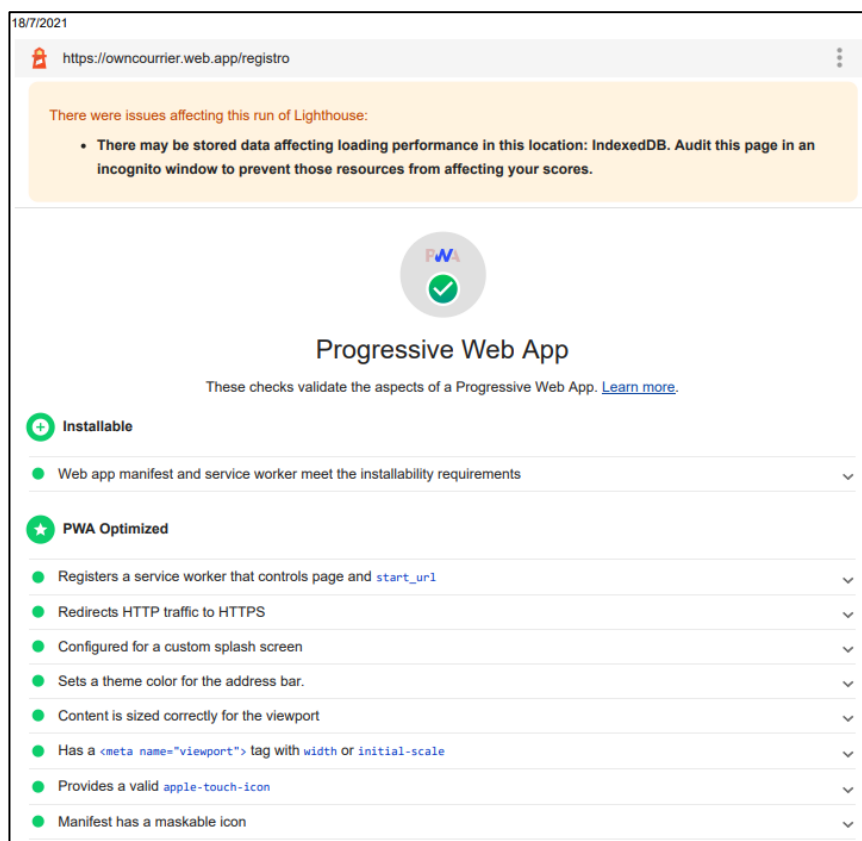


Figura 42. Resultados de Lighthouse.

En la Figura 42 podemos observar los resultados obtenidos en la ejecución de la herramienta en la pantalla principal, donde se analizó que la aplicación Owncourrier cumple con las siguientes funciones y configuraciones:

- Que la aplicación sea instalable.
- Tenga una conexión segura https.
- Que sea rápido y fluida
- Que funcione en cualquier navegador
- Se adapte a cualquier tamaño de pantalla
- Que proporcione una experiencia sin conexión

Mediante el resultado obtenido de lighthouse podemos asegurar al usuario que la aplicación es considerada una PWA y podrá utilizar cada una de las características que se ha mencionado. En la Figura 43, se muestra un ejemplo que la App OwnCourier cumple exitosamente con la funcionalidad de ser instalable. Al momento de ingresar al sitio web, este puede ser descargado en el dispositivo móvil como una aplicación nativa. El manual de instalación se podrá ver en el ANEXO 9.

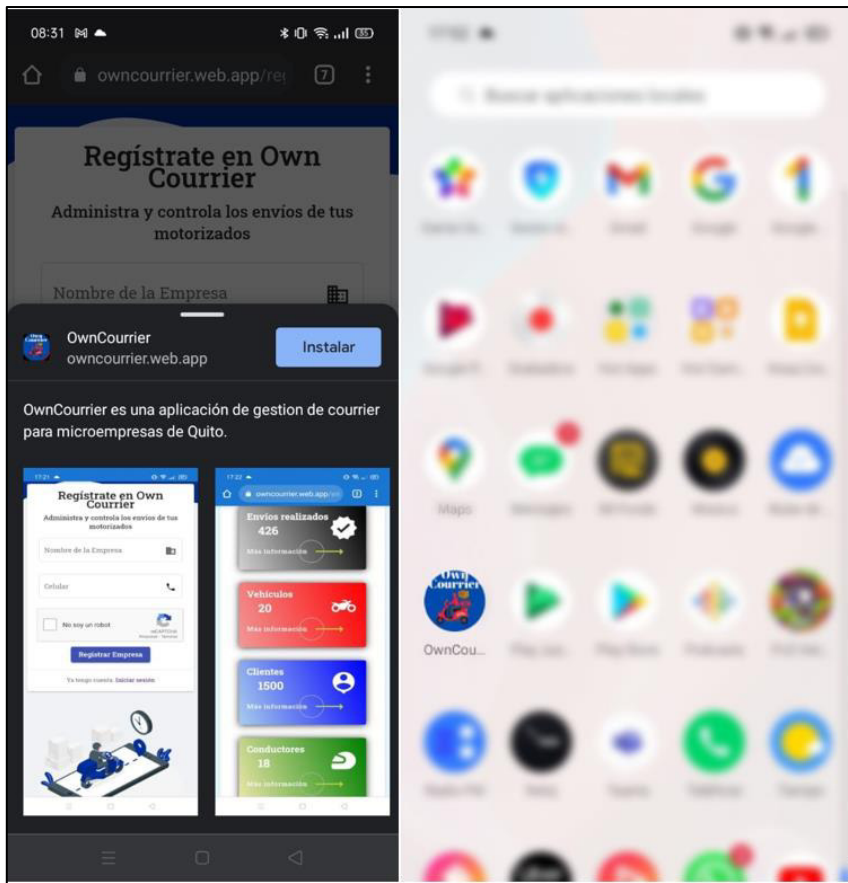


Figura 43. Instalador aplicación OwnCourier.

4. CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- Se desarrolló una PWA de gestión de courier para las microempresas de la ciudad de Quito brindando una mayor facilidad en la administración de logística interna.
- Al analizar las diferentes herramientas se seleccionaron a las más apropiadas para el desarrollo de una PWA y cada uno de sus módulos en base a la arquitectura planteada en el proyecto.
- Con la implementación de la encuesta de usabilidad de la aplicación se puede observar que la interfaz gráfica es fácil de utilizar, además de brindar una intuitiva experiencia de usuario en cada uno de sus módulos.
- Se aplicó la metodología ágil SCRUM, esta ayudó a disminuir el tiempo de desarrollo y a organizar las tareas del proyecto, donde se pudo entregar de forma progresiva componentes que dan valor al producto final.
- Se realizó la integración de características de una PWA como tener un service worker integrado, que posea una conexión segura con el protocolo https, funcione sin conexión a internet mediante el uso de cache y la creación de un manifiesto para que sea instalable en cualquier dispositivo.
- Se implementó pruebas unitarias en el proyecto que permitieron validar que cada uno de los componentes y funciones que dan valor al negocio cumplan con los objetivos establecidos.
- Con el análisis de herramientas de calidad como lighthouse se audito a la aplicación para comprobar que la misma posea las funcionalidades y características necesarias para ser considerada una PWA.
- Mediante una encuesta se consideró la situación actual de las microempresas en la ciudad de Quito previo y durante la pandemia de COVID-19, adquiriendo datos y características similares entre las distintas empresas. Con esto se obtuvo los requerimientos para el desarrollo del proyecto.

4.2. Recomendaciones

- Se recomienda realizar una integración de un módulo de ventas como una tienda online para tener el control directo de los pedidos de cada uno de los clientes.
- Se recomienda mantener el plan Blaze de firebase para que cada una de las funcionalidades de la nube (cloud functions) se puedan ejecutar correctamente ya que necesitan de un plan de pago.
- Se recomienda crear una función de la nube (cloud function) para optimizar las imágenes que se almacenan en la aplicación, para que la facturación de almacenamiento sea mínima.
- Se recomienda agregar un módulo de seguimiento en tiempo real, para brindar más información al cliente sobre los envíos en curso.
- Se recomienda implementar un servicio de análisis de rutas para disminuir costos y dar mayor agilidad en los tiempos de entrega de pedidos.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] C. Salas, "El 67% de la población cuenta con un dispositivo móvil", KienyKe, 2020. [Online]. Disponible en: <https://www.kienyke.com/tecnologia/el-67-de-la-poblacion-cuenta-con-un-dispositivo-movil>. [Accedido: 06- Jul- 2020].
- [2] G. Bit, "El 2019 es el año de las compras online", Business-intelligence.grupobit.net, 2019. [Online]. Disponible en: <https://business-intelligence.grupobit.net/blog/el-2019-es-el-a%C3%B1o-de-las-compras-online>. [Accedido: 06- Jul- 2020].
- [3] Brian G. McHale, Bruce W. Wilkinson, Todd D, Mattingly, Anton Valkov, Donald R. High, John J. O'Brien, Sai Phaneendra Sri Harsha Viswanath Putcha, Balaraman Kirthihgaivasan (2018). "Courier Managment System". Arkansas, United States.
- [4] G. Landa Gálvez, D. Núñez Pauca and M. Valdivia Navarro (2017). IMPACTO DE LA CADENA DE SUMINISTROS COURIER EN EL COMERCIO INTERNACIONAL DEL PAÍS. Universidad del Pacífico Escuela de Postgrado. Lima - Perú.
- [5] Sunil Tandel S., Abhishek Jamadar (2018). "Impact of Progressive Web Apps on Web App Development" Department of Computer Engineering, Smt.Indira Gandhi College, Navi Mumbai. Maharashtra, India.
- [6] S. López, "Qué es PWA: características, ventajas y desventajas", DIGITAL55, 2020. [Online]. Disponible en: <https://www.digital55.com/desarrollo-tecnologia/que-es-pwa-ventajas-desventajas/>. [Accedido: 24- Sep- 2020].
- [7] "Emprendedores se mantienen a través de entrega a domicilio en Ecuador", El Universo, 2020. [Online]. Disponible en: <https://www.eluniverso.com/noticias/2020/03/19/nota/7787379/emprendedores-se-mantienen-traves-entrega-domicilio>. [Accedido: 20- Jul- 2020].
- [8] "CÓMO FUNCIONA EL SERVICIO DE COURIER", Courier-pits.blogspot.com, 2011. [Online]. Disponible en: <http://courier-pits.blogspot.com/2011/09/como-funciona-el-servicio-de-courier.html#:~:text=El%20servicio%20de%20Courier%20se,limitado%20a%20un%20determinado%20costo.&text=Est%C3%A1s%20empresas%20que%20cuentan%20con,a%20pagar%20por%20dicho%20servicio>. [Accedido: 20- Jul- 2020].
- [9] I. Ramírez, "¿Qué es una Aplicación Web Progresiva o PWA?", Xataka.com, 2018. [Online]. Disponible en: <https://www.xataka.com/basics/que-es-una-aplicacion-web-progresiva-o-pwa>. [Accedido: 20- Jul- 2020].

- [10] "Ciclo de vida Ágil - Proagilist", Proagilist, 2016. [Online]. Disponible en: <https://proagilist.es/blog/agilidad-y-gestion-agil/agile-scrum/ciclo-de-vida-agil/>. [Accedido: 20-Jul- 2020].
- [11] E. Ledesma, "Por qué Scrum?", Capacitación, Consultoría Dirección Proyectos TenStep, 2016. [Online]. Disponible en: <https://www.tenstep.ec/portal/servicios-de-formacion/por-que-scrum>. [Accedido: 20- Jul- 2021].
- [12] Microsoft (2021). Documentation for Visual Studio Code. Retrieved 3 January 2021, from <https://code.visualstudio.com/docs>
- [13] Y. Fain and A. Moiseev, Angular 2 development with TypeScript. Shelter Island, NY: Manning Publications Co., 2017.
- [14] "Angular", Angular.io, 2020. [Online]. Disponible en: <https://angular.io/>. [Accedido: 12-Dec- 2020].
- [15] Biørn-Hansen, A., A. Majchrzak, T., & Grønli, T. (2017). Progressive Web Apps: The Possible Web-native Unifier for Mobile Development [Ebook] (1st ed., p. 8). Oslo, Norway: Faculty of Technology, Westerdals Oslo ACT. Retrieved from <https://www.scitepress.org/papers/2017/63537/63537.pdf>
- [16] Hajian M. (2019) PWA with Angular and Workbox. In: Progressive Web Apps with Angular. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-4448-7_13
- [17] F. Sergio, COMO CONVERTIR UNA APLICACIÓN ANGULAR A UNA APLICACIÓN WEB PROGRESIVA, 1st ed. Cochabamba, Bolivia, 2020.
- [18] A. Team, "Angular Material", Angular Material, 2020. [Online]. Disponible en: <https://material.angular.io/>. [Accedido: 16- Dec- 2020].
- [19] Attardi J. (2020) Flexbox. In: Modern CSS. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-6294-8_10
- [20] G. Niess, A. Roubal, S. Thurner and E. Barba Roque, CSS Grid Layouts, 1st ed. A-8010 Graz, Austria: 1, 2019, p. 21.
- [21] "Flexbox - Aprende sobre desarrollo web | MDN", Developer mozilla, 2020. [Online]. Disponible en: https://developer.mozilla.org/es/docs/Learn/CSS/CSS_layout/Flexbox. [Accedido: 16- Dec- 2020].
- [22] R. E. Plant, Interactive maps with the leafletand mapviewpackages, 2nd ed. 2018, p. 10.

- [23] "Leaflet — an open-source JavaScript library for interactive maps", Leafletjs.com, 2020. [Online]. Disponible en: <https://leafletjs.com/>. [Accedido: 16- Dec- 2020].
- [24] L. Moroney, *The Definitive Guide to Firebase*. Berkeley, CA: Apress, 2017.
- [25] N. Chatterjee, S. Chakraborty, A. Decosta and A. Nath, *Real-time Communication Application Based on Android Using Google Firebase*, 6th ed. Kolkata, India: IJARCSMS, 2018, p. 6.
- [26] S. Khedkar and S. Thube, *Real Time Databases for Applications*, 4th ed. Wagle Industrial Estate, Near Mulund: International Research Journal of Engineering and Technology (IRJET), 2017, p. 5.
- [27] P. Moreno, "Invision Studio, una herramienta de diseño y prototipado UI/UX que promete mucho.", fcomoreno.net, 2017. [Online]. Disponible en: <https://www.fcomoreno.net/invision-studio-una-herramienta-de-diseno-y-prototipado-uiux-que-promete-mucho/>. [Accedido: 20- Jul- 2021].
- [28] "Introducción a Karma Js", 2021. [Online]. Disponible en: <https://www.mmfilesi.com/blog/introduccion-a-karma-js/>. [Accedido: 20- Jul- 2021].
- [29] "Qué es Git: conviértete en todo un experto en Git con esta guía", Atlassian, 2021. [Online]. Disponible en: <https://www.atlassian.com/es/git/tutorials/what-is-git>. [Accedido: 20- Jul- 2021].
- [30] "GitHub: ¿Qué Es GitHub Y Cómo Utilizarlo?", Tutoriales Hostinger, 2021. [Online]. Disponible en: <https://www.hostinger.es/tutoriales/que-es-github#Que-es-GitHub-y-por-que-es-tan-popular>. [Accedido: 20- Jul- 2021].
- [31] "Lighthouse | Tools for Web Developers | Google Developers", Google Developers, 2021. [Online]. Disponible en: <https://developers.google.com/web/tools/lighthouse?hl=es#:~:text=Lighthouse%20es%20un%20herramienta%20automatizada,web%20apps%2C%20SEO%20y%20m%C3%A1s>. [Accedido: 20- Jul- 2021].
- [32] "¿Para qué se utiliza Jira Software? | Atlassian", *Atlassian*, 2021. [Online]. Disponible en: <https://www.atlassian.com/es/software/jira/guides/use-cases/what-is-jira-used-for>. [Accedido: 20- Jul- 2021].
- [33] Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017). SCRUM model for agile methodology. 2017 International Conference on Computing, Communication and Automation (ICCCA). doi:10.1109/cca.2017.8229928

- [34] M. REHKOPF, "Epics | Atlassian", Atlassian. [Online]. Disponible en: <https://www.atlassian.com/es/agile/project-management/epics>. [Accedido: 20- Jul- 2021].
- [35] M. REHKOPF, "Historias de usuario | Ejemplos y plantilla | Atlassian", Atlassian. [Online]. Disponible en: <https://www.atlassian.com/es/agile/project-management/user-stories>. [Accedido: 20- Jul- 2021].
- [36] Cho, J. (2008). ISSUES AND CHALLENGES OF AGILE SOFTWARE DEVELOPMENT WITH SCRUM. *Issues In Information Systems*, 1(1), 8. doi: 10.48009/2_iis_2008_188-195
- [37] Pruebas unitarias, 1st ed. Universidad de Alicante, 2021, pp. 1-12.
- [38] "Hacer que las PWAs trabajen desconectadas con servicio workers - Aplicaciones Web Progresivas | MDN", Developer.mozilla.org, 2021. [Online]. Disponible en: https://developer.mozilla.org/es/docs/Web/Progressive_web_apps/Offline_Service_workers. [Accedido: 20- Jul- 2021].
- [39] Peres, S. C., Pham, T., & Phillips, R. (2013). Validation of the System Usability Scale (SUS): SUS in the Wild. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 57(1), 192–196. <https://doi.org/10.1177/1541931213571043>
- [40] "Lighthouse PWA Analysis Tool | Web | Google Developers", *Google Developers*. [Online]. Disponible en: <https://developers.google.com/web/ilt/pwa/lighthouse-pwa-analysis-tool>. [Accedido: 20- Jul- 2021].
- [41] Encuesta: Cuéntenos cómo afecta el COVID-19 a su empresa. (n.d.). Retrieved August 4, 2021, from Intracen.org website: <https://www.intracen.org/noticias/Encuesta-Cuentenos-como-afecta-el-COVID-19-a-su-empresa/>
- [42] "Fibonacci Agile Estimation", *Productplan.com*, 2021. [Online]. Disponible en: <https://www.productplan.com/glossary/fibonacci-agile-estimation/>. [Accedido: 20- Jul- 2021].

6. ANEXOS

6.1. ANEXO 1: Encuesta sobre la situación actual empresas

Toda la información respecto al anexo 1 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/EfruNm2VECdNjJ_eATAvz5MBirp6ULh9D-Q6fG51yodfCQ?e=VBUdEh

6.2. ANEXO 2: Proceso de configuración inicial del proyecto

Toda la información respecto al anexo 2 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/EQ6Kdaol5QdEhZT4v_W6nbIBzOz-ym9QqXX78JhhPRaT0Q?e=otiSZM

6.3. ANEXO 3: Mockups

Toda la información respecto al anexo 3 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/ETL1ayhUFQREvEUE-eKF5ToBxmimUPt2Tv5Xon28Du5LA?e=y0nriN

6.4. ANEXO 4: Pantallas del sistema

Toda la información respecto al anexo 4 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/ESUtSUUrz2RHtS2BuFdFa0gBoLQ8tTX9KEhtKFlx-aqNyA?e=s0Agaq

6.5. ANEXO5: Pruebas unitarias

Toda la información respecto al anexo 5 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/Ea_SxdNQriVNjvDjSU Y1JXEBEwUyWMPiCTVMZsMrDce6kA?e=sdnVb1

6.6. ANEXO 6: Encuesta de usabilidad

Toda la información respecto al anexo 6 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/EabBIM7kxXRNhuQZgwJNm6UBhOIRdXZ1mAn2tf3DJHM0TA?e=AwyOE

6.7. ANEXO 7: Historias de Usuario

Toda la información respecto al anexo 7 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/EUd4-Uhw5ppKoVQNjkO01XlBgenqLS24UzF2lYOe1PuWPQ?e=lyahew

6.8. ANEXO 8: Manual de Usuario

Toda la información respecto al anexo 8 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/ES1coZLBbG5JifeQBZhVDIIBmCZLftXu6YdgmtDGxMVtA?e=y81J5m

6.9. ANEXO 9: Manual de Instalación

Toda la información respecto al anexo 9 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:w:/g/personal/christian_benitez_epn_edu_ec/EV2weBLpKltGImDfjFt5fncBgGr6Va0BFDEb_CCV9NAhfA?e=a5MLBx

6.10. ANEXO 10: Resultados Encuesta de Usabilidad

Toda la información respecto al anexo 10 se encuentra en el siguiente link:

https://epnecuador-my.sharepoint.com/:x:/g/personal/christian_benitez_epn_edu_ec/EanTQuspNAhJuRzB4VrQxOsBxkopU2u-hDGFmki7C_JLHw?e=i471bM