**UNIVERSIDAD TECNOLÓGICA DE TLAXCALA**

**Organismo Público Descentralizado del Gobierno del Estado**

**MEMORIA DE ESTADÍA PROFESIONAL**

APP MÓVIL DE CONTROL DE SERVICIO Y PERSONAL DE TRANSPORTE, URBANOS Y SUB URBANOS DE TLAXCALA S.A de C.V.

**QUE PARA OBTENER EL TÍTULO DE**

**TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA INFORMACIÓN ÁREA DESARROLLO DE SOFTWARE MULTIPLATAFORMA**

**PRESENTA**

Santiago Alameda Sánchez

**ASESORES (AS)**

M.S.C Francisco López Briones

Ing. José Antonio Martínez Lozano

EL CARMEN XALPATLAHUAYA, HUAMANTLA, TLAXCALA,

**UNIVERSIDAD TECNOLÓGICA DE TLAXCALA**

**Organismo Público Descentralizado del Gobierno del Estado**

**MEMORIA DE ESTADÍA PROFESIONAL**

APP MÓVIL DE CONTROL DE SERVICIO Y PERSONAL DE TRANSPORTE, URBANOS Y SUB URBANOS DE TLAXCALA S.A de C.V.

**QUE PARA OBTENER EL TÍTULO DE**

**TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS DE LA INFORMACIÓN ÁREA DESARROLLO DE SOFTWARE MULTIPLATAFORMA**

**PRESENTA**

Santiago Alameda Sánchez

**ASESORES (AS)**

M.S.C Francisco López Briones

Ing. José Antonio Martínez Lozano

EL CARMEN XALPATLAHUAYA, HUAMANTLA, TLAXCALA,

AGOSTO DE 2025

**DICTAMEN DE APROBACIÓN**

**AGRADECIMIENTOS**

**ÍNDICE**

**RESUMEN**

Este es un resumen de mi tesina

**INTRODUCCIÓN**

El presente proyecto consistirá en realizar una aplicación móvil de control de servicio y personal de transporte en la empresa **Urbanos y Sub Urbanos de Tlaxcala S.A de C.V.** con el objetivo de solucionar problemáticas, que presenta la empresa principalmente con sus operadores es de suma importancia resolver debido al incumplimiento de los choferes con sus tiempos establecido o el no respetar las rutas que se les asigna, con esta app móvil se busca agilizar los tiempos, y el constante monitoreo y control interno de la empresa.

Para esta tesina se ocuparon varias tecnologías que se usaron para la elaboración de este proyecto **Flutter** como nuestro principal framework de desarrollo móvil, **Git** para nuestro control de versiones del proyecto a elaborar, **Git Hub** como nuestro repositorio de proyecto en la nube, **Firebase** como herramienta y plataforma de desarrollo en la nube que permitió el uso de (“**Firebase Firestore**”, “**Firebase Realtime Data Base**” y “**Fire Storage”**) estos sin usarse por las limitaciones que veremos en el punto “**1.6. limitaciones**”, con este proyecto lo que se busca aportar es tener un mayor control de los trabajadores de “**USU**”, para poder sancionar a los trabajadores irresponsables, teniendo una manera más práctica de manejar los registros de sus nuevos trabajadores y los nuevos vehículos, permitiendo a “**futuro**” integrarlo con una aplicación web para un mayor manejo y facilidad en caso de las gestiones de operadores y de unidades, así mismo teniendo la facilidad de si por el movimiento o las prisas necesitan hacer un cambio lo puedan hacer desde la comodidad de su dispositivo móvil.

Durante la tesina hay varios capítulos que tienen distintos enfoques en el “**Capítulo I Descripción del Proyecto**” veremos cosas como son: la 1.1 caracterización del proyecto, que es el cómo se va desarrollar el proyecto, 1.2 Planteamiento de problema cual es el problema y que se problemática se busca resolver, Los Objetivos del proyecto, generales y específicos, los alcances que se llegara con el proyecto y las limitaciones, y el cronograma de actividades, en el “**Capítulo II** Marco de Referencia” sin embargo no tiene este nombre sino el nombre del proyecto el cual es app móvil de control de servicio y personal de transporte urbanos y sub urbanos de tlaxcala s.a de c.v. en este capítulo se habla sobre la historia de de la empresa, su misión como empresa, su visión de empresa, su ubicación que su tipo de actividad empresarial en el “**Capitulo lll Marco Teórico**” se hacen citas de temas relacionados el proyecto y con lo aprendido en el “Capitulo IV” nos narra de manera más detallada el desarrollo del proyecto de estadías y los resultados obtenidos

CAPÍTULO I

APP MÓVIL DE CONTROL DE SERVICIO Y PERSONAL DE TRANSPORTE, URBANOS Y SUB URBANOS DE TLAXCALA S.A de C.V.

1. APP MÓVIL DE CONTROL DE SERVICIO Y PERSONAL DE TRANSPORTE, URBANOS Y SUB URBANOS DE TLAXCALA S.A de C.V.
   1. Caracterización del área del desarrollo del proyecto

El proyecto se llevará a cabo bajo la modalidad de trabajo remoto, ya que esta opción se ajusta de manera más adecuada a los tiempos y espacios disponibles en la empresa. El desarrollo se realizará desde casa utilizando equipo de cómputo propio, lo que permitirá una mejor organización y flexibilidad en las actividades. Para garantizar una comunicación constante y efectiva, se establecerán reuniones periódicas con la empresa, complementadas con contacto mediante llamadas telefónicas, mensajes y correo electrónico, a fin de dar seguimiento oportuno al proyecto.

Este proceso de estadías se llevará a cabo en el periodo mayo – agosto, en el cual se pondrán en práctica los conocimientos adquiridos dentro de esta institución académica.

* 1. Planteamiento de problema

La empresa denominada Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V. (USU) dedicada al servicio de transporte público mediante unidades tipo combi con aproximadamente 363 vehículos de diferentes marcas, en su mayoría Nissan y Toyota, enfrentan una problemática relacionada con el control de sus choferes/operadores. Actualmente no existe un mecanismo eficaz que garantice el total cumplimiento de las rutas y horarios establecidos, lo que genera conflictos en la prestación del servicio y dificulta la supervisión del personal y control del mismo.

Aun cuando la empresa ha implementado a lo largo de los años medidas de control de tiempos por ejemplo con los relojes checadores, estos no son del todo eficaces debido a que los operadores de las unidades mediante astucias, engaños y manipulación del reloj checador han elaborado estrategias que alteran y burlan el tiempo de arribo de llegada a los puntos de control, por lo cual el sistema de chequeo de las unidades de la empresa no es del todo fiable. Esta situación deriva en una falta de precisión en el registro de tiempos establecidos a los operadores para terminar su ruta establecida y las ubicaciones de estos mismo. Debido a este punto de la ubicación de las unidades en tiempo real es necesario implementar también mecanismos para el rastreo GPS y así tener una ubicación precisa de cada operador y así para poder sancionar a los que incumplan la hora y rutas asignadas.

Para abordar esta problemática se plantea el desarrollo de una app móvil para dispositivos con Sistema Operativo Android, que incluya la gestión de unidades, choferes/operadores y tiempos para operadores de planta.

* 1. Justificación

La elaboración de este proyecto es suma importancia debido a que los socios, administradores y las personas al mando de la empresa sufren la problemática de no tener un control total de sus choferes/operadores que incumplen con las horas establecidas por parte de la empresa violando las reglas siguiendo otras rutas, llegando tardíamente sin recibir alguna sanción.

Con la aplicación se dará solución a estos inconvenientes para que los operadores que manejan las unidades sean puntuales con los tiempos establecidos por la empresa, así mismo se busca que sigan las rutas establecidas y no tomen otros caminos cuando no sea necesario o no haya algún problema con estas mismas rutas.

Agregar el porqué de una app móvil.

* 1. Objetivos
     1. Objetivo general

Desarrollar una aplicación móvil que permita la gestión de información sobre las unidades, rutas y choferes para la empresa **Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V.** (**USU**). Mediante el framework Flutter, apoyándose de metodologías ágiles como scrum.

* + 1. Objetivos específicos
* Realizar un cronograma de actividades para el desarrollo de la aplicación móvil.
* Elegir una metodología de desarrollo de software.
* Establecer requerimientos funcionales y no funcionales.
* Investigar y establecer las tecnologías para el desarrollo.
  1. Alcances

El proyecto contempla el desarrollo de múltiples funciones con las que contará esta aplicación móvil diseñadas de manera precisa, así mismo, la aplicación integrará el logotipo de la empresa, incluyendo un inicio de sesión donde los usuarios partícipes puedan iniciar sesión en sus respectivas cuentas con los roles asignados basándose en su área de trabajo en la empresa (administradores, despachadores o choferes). , Este proyecto será de gran ayuda para la gestión dentro de la empresa de “**Urbanos y Sub Urbanos de Tlaxcala S.A de C.V**”

A través de esta aplicación se busca y se pretende agrupar la información y llevar un historial de los choferes como de las unidades, logrando tener un mayor control del personal de la empresa y de las operaciones en general, mejorando la comunicación entre los roles participes en el uso de la aplicación, los cuales son:

* **Choferes:** Los cuales solo podrán enviar mensajes y cambiar ciertos aspectos estéticos de la aplicación**.**
* **Administradores:** Teniendo un control absoluto de los choferes pudiendo dar de alta cuentas de la aplicación como editando aspectos como lo son nombres, contraseñas o eliminando cuentas.
* **Despachadores** Ellos juegan un rol bastante sencillo el cual es dar de alta a un chofer, pero sin poder eliminarlo, solamente pudiendo añadirlo u editarlo con limitantes**.**

Estos roles son los que tendrán acceso a la aplicación móvil, cada uno desempeñándose en diferentes aspectos según el mismo, los despachadores al hacer un cambio la aplicación mandaría una notificación a los administradores si el despachador registrara un chofer/operador o lo modificara, así los administradores tendrán un control total del personal de la empresa.

Se propone que los administradores y los despachadores tengan la capacidad de dar de alta las cuentas de los choferes. El sistema de inicio de sesión estará principalmente contemplado que en el uso del “**nombre de usuario**” y una “**contraseña**”como principales datos para el inicio de sesión, sin embargo, de manera opcional se podrá iniciar de igual manera con el correo electrónico si es que algún operador cuenta con uno, debido que los operadores pertenecen a un rango de edad avanzada y presentan dificultades con el correo electrónico.

Por esta misma razón se opta que el “**correo electrónico**” sea un mecanismo de autentificación opcional, mientras el “**nombre de usuario**” y la “**contraseña**” siendo los medios de autentificación principales.

La aplicación contará con un módulo de gestión de operadores (CRUD), como otro módulo de gestión (CRUD) de las unidades lo cual le permitirá consultar y administrar el historial de cada uno respectivamente. Se integrará un apartado de configuración el cual implementará un modo oscuro y un apartado para ajustar el tamaño de la fuente, así como un menú lateral (sidebar) o un navegador inferior (bottom navigator) con el objetivo de garantizar una experiencia más cómoda e intuitiva.

De igual manera, se proyecta llegar a un sistema de mensajería interna (chat de texto) donde facilite la comunicación en la empresa, igual se tiene las expectativas de añadir un módulo de mi perfil donde los usuarios con roles despachador y administrador podrán editar su información mientras los usuarios que tengan el rol chofer únicamente puedan visualizar su información.

El desarrollo de esta aplicación móvil se llevará a cabo mediante el framework **Flutter** el cual utiliza el lenguaje de programación **Dart**, mientras que como gestor de bases de datos en la nube una base de no SQL (no relacional) **Firebase**.

* 1. Limitaciones

Por razones monetarias y un periodo corto destinado al desarrollo, es necesario delimitar el alcance del proyecto, La función principal la cual es el Geolocalización por medio de GPS más aparte otras como la carga de archivos multimedia como (**pdf, docx, xls**) el cual genera un costo de almacenamiento adicional debido a **Firebase** **Storage** el cual el modelo se basa en el consumo de almacenamiento y transferencias.

Asimismo para la visualización de mapas es necesario activar la facturación automática en **Google Cloud** aun cuando el servicio otorga $200 dólares gratuitos pero el cual resulta suficiente siempre que no se superen los límites de peticiones (**escrituras y lecturas 50,000 al mes**) sin embargo al no contar con alguna tarjeta de crédito o débito es necesario para poder activar este servicio antes mencionado y en caso de exceder los recursos gratuitos $200 dólares gratis al mes, se generarían cargos que yo no cuento ni con el recurso, ni me corresponden, debido a ello se decidió delimitar ambas función (GPS y subir archivos multimedia).

Adicionalmente, en caso de implementar dichas funciones ya mencionadas, como lo son subir archivos multimedia o la geolocalización por GPS en tiempo real como son 360 operadores aproximadamente, ejemplo si un conductor mandara su ubicación cada 5 segundos, esto implicaría 259,000 escrituras por hora esto superaría el limite gratuito que son 50,000 escrituras gratuitas y contando que el servicio te cobra solo si te excedes del límite y debido a mi situación económica, sin contar que esos gastos no me corresponden a mí, sino a la empresa, esta función y siendo la función principal del sistema estará delimitada.

* 1. Cronograma

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No.** | **Actividades** | **Meses.** | | | | | | | | | | | | | | | |
| **Mayo** | | | | **Junio** | | | | **Julio** | | | | **Agosto** | | | |
|  |  | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** | **1** | **2** | **3** | **4** |
| **1** | Estudio del Framework Flutter |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **2** | Estudio de Firebase y MongoDB |  | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **3** | Estudio del Lenguaje Dart |  | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **4** | Analizar posibles problemáticas |  |  | **x** |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **5** | Reuniones semanales |  |  |  | **x** | **x** |  |  |  |  |  |  |  |  |  |  |  |
| **6** | Definición tipo de aplicación (Web, móvil o escritorio) |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |  |  |
| **7** | Levantamiento de requerimientos |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |  |
| **8** | Definición de roles |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |  |
| **9** | Maquetación (Wireframes) |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |  |
| **10** | Aprobación del diseño |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |
| **11** | Análisis de sistema de verificación por lector óptico IOT |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |
| **12** | Riesgos del proyecto |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |
| **13** | Límites del proyecto |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |  |
| **14** | Metodología de desarrollo |  |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |
| **15** | Desarrollo del Sistema |  |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |  |
| **16** | Crud de choferes |  |  |  |  |  |  |  |  | **x** |  |  |  |  |  |  |  |
| **17** | Crud de las unidades |  |  |  |  |  |  |  |  |  | **x** |  |  |  |  |  |  |
| **18** | Historial de choferes |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |  |  |
| **19** | Historial de Unidades |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |  |  |
| **20** | Menú lateral (Dashboard) |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |  |  |
| **21** | Inicio de sesión |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |  |
| **22** | Configuración |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |
| **23** | Modo oscuro |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |  |
| **24** | Tipo de letra |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |
| **25** | Ordenamiento alfabéticamente en choferes |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |
| **26** | Log Out (Cerrar Sesión) |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |
| **27** | Vigencia de permiso y vigencia licencia en la crud de choferes |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |  |
| **28** | Pantalla de Mensajería |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |
| **29** | Pantalla de Mi perfil |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |  |
| **30** | Testeo |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** |

CAPÍTULO II

URBANOS Y SUB URBANOS DE TLAXCALA S.A. DE C.V

1. ****Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V****
   1. Antecedentes de la empresa

La empresa **“Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V.”** se fundó en el año de 1963 y se consolidó a partir de la unión de múltiples rutas, iniciando por caminos de terracería y camiones relativamente sencillos. Antes de llamarse “Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V.” se solía llamar **“Circuitos Urbanos y Suburbanos de Tlaxcala”**. Fueron varias familias las que fundaron esta empresa, como lo son “Los Morales” de San Francisco Tetlanohcan y “Los Caporal”. Todos ellos tienen algo en común: ser personas que dedicaron la mayor parte de su vida al transporte.

“Su base estaba en el lado este del entonces Cine Matamoros, en el lugar donde actualmente está la Plaza Xicohténcatl; eso fue por los años de 1968, 1970.” (Reyes, Historia de la empresa Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V. [Manuscrito no publicado], 2025).

Cabe mencionar que poco a poco la empresa fue creciendo en un 70%. Según las palabras de José Antonio Martínez Lozano, recalca que durante su gestión como representante legal de la empresa “USU” modernizaron sus unidades, pasando de usar camiones viejos a combis. También narra que durante la época en que el Lic. Tulio Hernández Gómez fue gobernador del estado, se les obligó a cambiar los camiones por combis de la marca Volkswagen y por microbuses.

Asimismo, el Ing. José Antonio Martínez Lozano, representante legal de “Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V.”, quien llevó 11 años como socio de la empresa, en 2003 llegó a ser presidente de la compañía durante un lapso de 12 años, lo que corresponde a 4 periodos. Posteriormente fue sustituido de su cargo, pero regresó como presidente a partir del año 2017 hasta la actualidad, coincidiendo con los periodos de gobierno del Lic. Marco Antonio Mena Rodríguez y de la Lic. Lorena Cuéllar Cisneros.

El Ing. José Antonio Martínez Lozano narra con enorme tristeza que la mayoría de los representantes originales de Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V. ya han fallecido, tales como Ignacio Pérez Lima, quien fue un hombre muy importante para la USU, siendo presidente de esta misma empresa durante un periodo de 20 años, logrando hacerla crecer y siendo un representante realmente apreciado. Otros fundadores como Salvador Mata Primo también dejaron huella. Su papá ingresó en el año de 1976, comenta que es de las pocas personas fundadoras que aún sigue con vida. Él llegó unos treinta años después de la fundación y reconoce que, gracias al trabajo de los fundadores, la empresa aún ofrece el servicio de transporte. Es gracias a ellos que el transporte público sigue vigente hasta el día de hoy. (Reyes., 2025)

* 1. Misión

Prestar y mejorar en todo momento el servicio de transporte público en el estado en zonas urbanas y suburbanas del estado de Tlaxcala, con visión a ser mejor empresa.

* 1. Visión

Ser la mejor empresa de transporte público en el estado de Tlaxcala, con la aspiración a mejorar en todo momento

* 1. Ubicación



Figura 2.1 Ubicación de Urbanos y Sub Urbanos de Tlaxcala, Fuente: google.com/maps

**Calle Nayarit esquina con Michoacán, Col. El Alto, Chiautempan.**

* 1. Giro

Empresa dedicada al transporte público de pasajeros en zonas urbanas y suburbanas.

CAPÍTULO lll.

MARCO TEÓRICO

1. MARCO TEÓRICO
   1. Metodologías de desarrollo de Software

Las metodologías de desarrollo de software se definen como un proceso mediante el cual los desarrolladores diseñan, desarrollan y prueban nuevos programas informáticos, y se utilizan para optimizar los proyectos de software. Estas metodologías siguen una filosofía de diseño específica que ayuda a los desarrolladores a alinear estos procesos con las características del producto. Con la ayuda de estas metodologías ágiles, también se simplifican las tareas y se mejora la colaboración en las empresas (GeeksforGeeks, 2024) [traducción propia].

* + 1. Metodologías clásicas

Las metodologías clásicas se caracterizan y distinguen de las ágiles por su carácter estricto en cada etapa del desarrollo con el objetivo de ser más meticulosas en la elaboración del software. Se caracterizan por detallar de una manera más precisa todos los requisitos del proyecto desde el inicio de este mismo; sin embargo, presentan una baja flexibilidad a los errores, limitando las posibilidades de modificaciones durante su ejecución, además siendo muy lentas para proyectos con poco tiempo de desarrollo y siendo tardías, son más rigurosas al llevar un control del proceso desde el inicio, pero a la vez más lentas en el proceso. (ginzo.tech, 2021)

* + - 1. Cascada.

Cascada es una metodología clásica conocida por su visión de gestión de proyectos siguiendo una secuencia con forma de cascada por eso mismo de su nombre “**Waterfall model”** su nombre es por sigue una estructura en forma de cascada que sigue la corriente.

**Los 6 pasos del modelo de cascada:**

1. **Fase de requerimientos**: Es la fase donde los miembros del equipo reúnen la información del cliente para lograr conseguir el éxito en el proyecto, este se podría decir que es la fase más importante durante el desarrollo debido a que las siguientes fases dependen de las fases anteriores es necesario prever todo en detalle antes de empezar el proyecto, debido a esto la mayor parte está, dedicada a la planificación. Al finalizar la fase de requerimientos deberás tener un esquema muy claro del proyecto de principio a fin que incluya lo siguiente:

* Cada etapa del proceso.
* Quien trabaja en cada etapa.
* Las dependencias clave.
* Los recursos necesarios.
* Un cronograma en el que se detalla cuánto dura cada etapa.

1. **Diseño del Sistema**: Es el proceso de desarrollo de software, donde se especifica que hardware se va utilizar, que lenguaje de programación.   
   Existen 2 pasos fundamentales durante la fase de desarrollo que son: fase de diseño de **alto nivel** y fase de diseño **detallado**, la de alto nivel se podría decir que el equipo elabora un esqueleto de como funcionara el software como se accederá a la información, mientras que, durante la fase de diseño detallado, se definen los detalles particulares o mínimos, se podría decir que el de alto nivel es el esqueleto de la fase de diseño y el detallado son los órganos que lo componen.   
   Los miembros que hacen su proyecto con la metodología cascada, deben tener todo bien documentando para que el resto del equipo puedan consultar que se ha hecho a medida que el proyecto avanza.
2. **Etapa de implementación**: Está fase del modelo de cascada va de la mano de las fases “1” y “2”, ya aquí se define que lenguaje de programación se va utilizar, que tecnologías, se comienzan a desarrollar los módulos antes diseñados en la fase “2” junto a los requerimientos redactados en la fase “1” es por eso mismo que ya debe estar establecido el esquema, debido una fase depende de otra.
3. **Etapa de Pruebas**: En etapa de cascada se realizan pruebas donde los “QA testers” verifican cualquier error que deba corregirse. Debido a esto los encargados de hacer las pruebas deben documentar los errores encontrados para así tener un control de calidad, en caso de que desarrollador encuentra un error similar se podrá consultar en la documentación anterior para poder corregir el error.
4. **Fase de desarrollo**: En la fase de desarrollo es donde se implementa la versión final del software, es la que se implementa para los usuarios finales, en este caso se lanza el entregable final para el cliente.
5. **Fase de mantenimiento**: Una vez el proyecto ya se ha implementado puede haber situaciones donde se descubra algún nuevo error y sea necesario implementar alguna nueva actualización del software, a esto se le conoce como fase de mantenimiento y es demasiado común en el desarrollo de software y está etapa es continua(asana.com, 2025)**.**

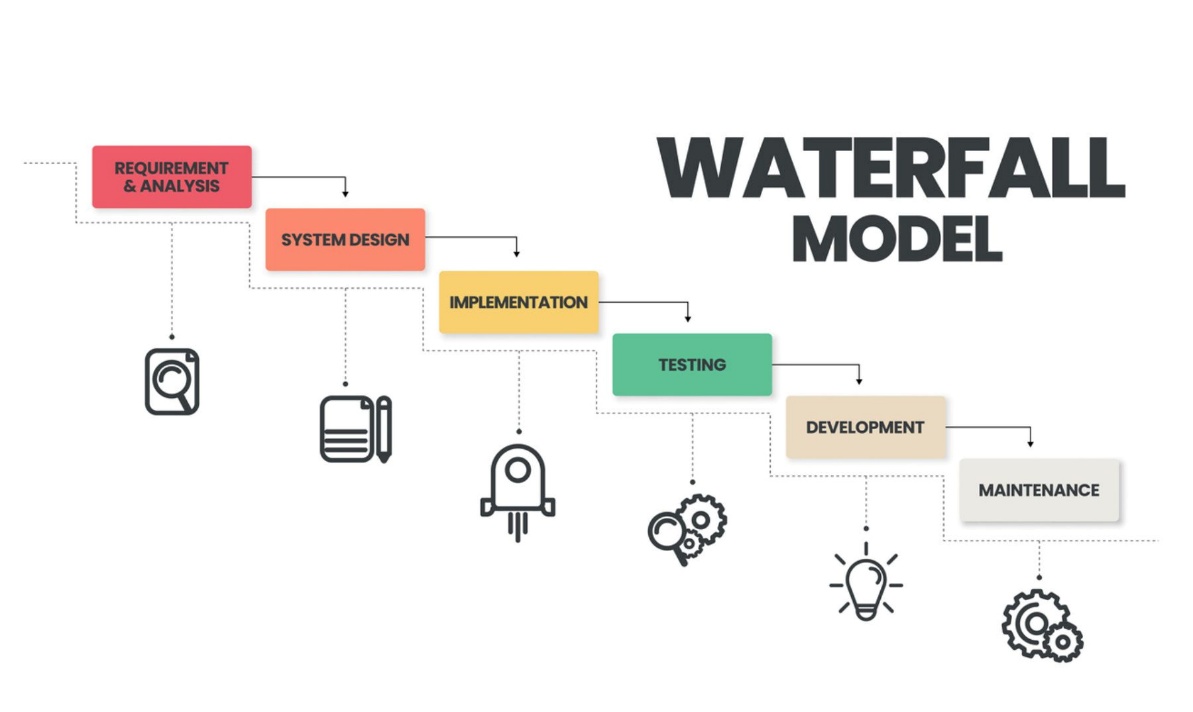


Figura 3.1 [Ciclo de Cascada] Fuente: coworkingfy.com (2020)

* + 1. Metodologías ágiles.

Las metodologías agiles son aquellas metodologías de desarrollo de software enfocados a entornos rápidos donde el tiempo de desarrollo es corto y se caracterizan por ser muy flexibles y se adaptan a los cambios constantes a diferencia de las metodologías clásicas que son muy poco flexibles a los cambios y donde ponen un enfoque claro desde el inicio además de contar con interacciones rápidas y trabajo más colaborativo a diferencia de las clásicas las metodologías clásicas no siguen un linaje y se pueden hacer varias fases al mismo tiempo para garantizar la optimización de tiempo se busca evitar perder tiempo y recursos. (udemy.com, 2020)

* + - 1. Scrum

Scrum es una metodología ágil basada en un ciclo, donde hay varios roles de desarrollo donde cada uno juega un papel muy importante como:

**Product Owner:** el se encarga de gestionar el **Product Backlog** y aportar la detallar y priorizar las funciones y requisitos más esenciales para el cliente.

**Scrum Master:** Es el encargado de resolver los obstáculos que el equipo pudiera tener fomenta la mejora entre miembros del equipo asegurándose que el equipo de desarrollo entienda todo con lo referente a scrum.

**Development Team** (Equipo de desarrollo): Es un equipo de profesionales que se compone de las personas responsables para dar solución a la necesidad del cliente. (Henao, 2018)

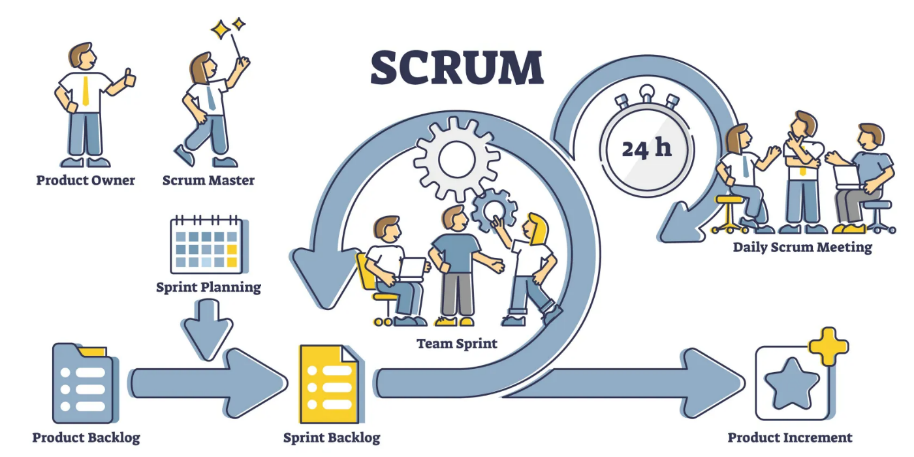


Figura 3.2 Ciclo de vida de scrum Fuente: ilunion.com (24 de enero de 2025)

* + - 1. Kanban

Kanban es una metodología ágil creada por Taiichi Ohno en 1940 para la empresa Toyota se apoya de un sistema visual para gestionar un trabajo a medida que avanza en un proceso, el objetivo es identificar posibles cuellos de botella y solucionarlos para que el trabajo fluya a una velocidad óptima.

Fases de Kanban

1. Visualizar el flujo del trabajo en un modelo clásico de Kanban hay 3 columnas:



Figura 3.3 [Funcionamiento de Kanban] Fuente: valtx.pe (2025)

1. Limitar los trabajos en proceso también conocidos por si abreviatura en inglés “WIP” son todas las tareas que un equipo está trabajando actualmente.
2. Administrar el flujo de trabajo.
3. Definir las reglas del proceso.
4. Bucles de retroalimentación.
   1. Aplicaciones Móviles.

Cifras de junio de 2024, Android es el sistema operativo móvil a nivel mundial más usado con un 72,15% de la población, mientras que iOS representa el 27,19%. (statista, 2024).

**Tipos de aplicaciones móviles:**

**Aplicaciones nativas:** Diseñadas para funcionar en un sistema operativo específico (Android o iOS). Son rápidas, estables y ofrecen una mejor experiencia de usuario.

**Aplicaciones híbridas:** Son multiplataforma diseñadas para funcionar en más de un sistema operático, con el fin de llegar varios dispositivos y ganar más publico u ser accesible para todos ellos.

1. Redes sociales (como Instagram, X, TikTok)
2. Juegos (Candy Crush, Among Us, etc.)
3. Comercio electrónico (Amazon, Mercado Libre)
4. Productividad (Trello, Google Drive)
5. Comunicación (WhatsApp, Telegram)
6. Educación (Duolingo, Coursera)
7. Salud y bienestar (MyFitnessPal, Calm)
8. Atención al cliente (como la app de **B2Chat**)
   * 1. Aplicaciones Nativas

Las aplicaciones nativas son software desarrollado para un sistema operativo en específico, esto permite aprovechar de manera más eficiente los recursos y capacidades del dispositivo accediendo directamente a funciones de hardware, como lo son: cámara, procesador, memoria, etc.

La desventaja es que, si se desea que la misma aplicación esté disponible en diferentes sistemas operativos, es necesario crear versiones separadas para cada uno, esto hace que el tiempo y el costo de desarrollo sea más elevado y más cansado, pero con mejores resultados en cuanto a compatibilidad se refiere, aunque también asegura un mejor rendimiento y mejor manejo de recursos en cada plataforma.

Ejemplos:

* **Android**: Google Maps se integra de mejor forma y más eficiente en dispositivos Android, ya que es desarrollado directamente por Google se facilita la integración con su propio sistema operativo.
* **iOS**: Safari es el navegador web desarrollado de manera nativa y exclusivamente para iOS, este software está optimizado para funcionar con su hardware y servicios como son: iCloud o Apple Pay por ejemplo.
* **Windows** Phone: Microsoft lanzó una versión de Office desarrollada de manera nativa para este sistema, ofreciendo mejor rendimiento al ser desarrollados por la misma empresa y permite una mejor integración que en otras plataformas.

Importante no confundir una aplicación nativa con una multiplataforma, ejemplo, Word es una app multiplataforma debido a que está disponible en Android, iOS, Windows y macOS. **Cada sistema puede contar con una versión nativa** **adaptada específicamente para este mismo**. En el caso de Windows Phone, Word es desarrollado de cero para integrarse con el sistema operativo, lo que lo convierte en una versión nativa, aunque siga siendo la misma aplicación disponible en otras plataformas. (FutureSpace, s.f.)

* + 1. Aplicaciones Multiplataforma

Estas aplicaciones son desarrolladas bajo un único lenguaje o framework y luego son modificadas para ejecutarse en distintos sistemas operativos sin necesidad de rehacer la app desde cero, a diferencia de las apps nativas donde, si quieres hacer una app nativa para cada sistema, sí debes rehacerla desde cero.

Facilita su exportación a otros sistemas; a diferencia de las aplicaciones nativas, estas se caracterizan por no estar pensadas para un único sistema, sino para llegar a bastantes sistemas, lo que permite que este tipo de aplicaciones lleguen a más personas.

Sin embargo, su rendimiento es peor frente a las apps nativas, debido a que no están pensadas para un único sistema, conteniendo en algunos casos ciertas limitaciones.

**Ventajas**:

* Ahorra tiempo, gastos y recursos, ya que con un único código se puede exportar a varias plataformas.
* Compatibilidad con varios dispositivos, lo cual permite su fácil acceso.
* Fácil desarrollo para lanzarse en múltiples plataformas.

**Desventajas**:

* Aunque funcionan en varios sistemas, pueden presentar problemas de rendimiento frente a las apps nativas; por ejemplo: mayor consumo de batería y mayor gasto de recursos del hardware.
* Se requiere testear en todos los dispositivos para el correcto funcionamiento en las plataformas donde será lanzado.
* Se deben generar paquetes específicos (APK e IPA) para lanzarse en tiendas como Play Store y App Store. (abamobile, 2024)
  + - 1. React native

React Native es una tecnología de código abierto desarrollada por META en su tiempo Facebook para crear aplicaciones móviles nativas y multiplataforma a partir de un mismo código, escrita en JavaScript y React, algo que se ha hecho típico en el desarrollo de apps. React Native te permite crear aplicaciones móviles usando React, pero no es el mismo React para la web si no un React totalmente diferente enfocado en el desarrollo de aplicaciones móviles, esto quiere decir que cuando desarrolles, si bien puedes usar muchos conceptos de React como los componentes, ya no estarás utilizando elementos WEB, si no ahora los utilizarás con elementos de interfaces móviles nativas. (Microsoft, 2024)

* + - 1. Ionic

Ionic es un framework frontend de código abierto para el desarrollo de aplicaciones híbridas basado en tecnologías web como l “**HTML, CSS y JS**”. Es un framework que permite el desarrollo de aplicaciones para iOS, Android y web, desde un único código como al igual que lo hace flutter. Se integra con frameworks frontend, como Angular, React y Vue, aunque también se puede usar JavaScript.

* Permite desplegar aplicaciones híbridas, en múltiples plataformas de manera nativa como Android, iOS y web, web progresivas
* Está basado en tecnologías web: **HTML, CSS y JavaScript**.
* Compatible con frameworks frontend, como Angular, React y Vue.
* **Interfaz de usuario UI, tarjetas ion-cards:** se usan mostrar la información de manera resumida, título, subtítulo etc.
* **Listas ion-list:** permite organizar los elementos en filas que incluyen imágenes pequeñas, botones o textos.
* **Pestañas ion-tabs:** Navegación por secciones tipo appbar, mediante una barra de pestañas ion-tab-bar.

(profile.es, 2025)

* 1. Geolocalización en aplicaciones móviles

**Cómo funciona la geolocalización en aplicaciones móviles**: La geolocalización utiliza el GPS integrado en los dispositivos móviles para obtener coordenadas de latitud y longitud precisas Así es como las aplicaciones móviles son localizadas en siendo ubicación precisa o un aproximado, (reanimasoluciones, s.f).

* + 1. GPS En dispositivos móviles

Sistema de posicionamiento Global o mejor conocido por su abreviatura GPS funciona mediante la recepción de señales satelitales se originó en 1970 por el departamento de seguridad de los Estados Unidos de América, el GPS funciona mediante un proceso llamado trilateración (Método para determinar la posición de un punto usando únicamente las distancias a otros tres puntos de referencia conocidos) los satélites mandan señales como datos, tiempo y posición donde mide el tiempo la señal GPS que tarda en llegar y calcula la distancia entre cada uno de los satélites. Así con al menos 4 satélites mide: latitud, longitud y altitud dando con tu posición exacta. (tsomobile, 2025) (equiposalfa, 2025)

* + 1. Google Maps SDK

Google Maps SDK son un conjunto de herramientas de desarrollo que otorga a los desarrolladores integrar los mapas de Google maps en sus dispositivos móviles o web, y funcionalidades de Google Maps (developers.google.com, 2024)

* 1. Seguridad en aplicaciones móviles

La seguridad en los dispositivos móviles ha aumentado constantemente en los últimos años. Las estadísticas muestran alrededor de un 90 % de la población global de Internet utiliza un dispositivo móvil para conectarse. Esto para los piratas informáticos significa que más personas deben victimizar, lo que hace que la seguridad de endpoint para dispositivos móviles sea cada vez más vital y métodos esenciales como la encriptación de contraseñas o tokens que se destruyan después de un tiempo. (fortinet, 2025)

* 1. Flutter.

Flutter es un framework creado por la empresa Google para el desarrollo de aplicaciones. Es un (SDK) Software Development Kit, que es un kit de desarrollo lanzado en mayo de 2017 y cuenta con soporte tanto para IOS como para Android, a su vez trabaja con el lenguaje orientado a objetos Dart.

Permite a los desarrolladores escribir un solo código y implementarlo en diferentes medios como: IOS, Android, Windows, Linux, etc.

**Funcionamiento de Flutter**

Flutter utiliza un único código base que se compila en código máquina nativo para diferentes plataformas, como Android, iOS, web y escritorio.

Al crear una aplicación Flutter, por default se crea una (UI) mediante widgets. Los widgets son los componentes Flutter. Todo Flutter es un widget, todos sus componentes como son los botones o el texto. Estos widgets se combinan así creando una “**UI**”, y el código se compila posteriormente en código específico mediante el motor de Flutter



Figura 3.4 [Estructura básica de un App en Flutter] Fuente:www.browserstack.com (2025)

Este código define la estructura de una aplicación básica de Flutter, donde MaterialApp y Scaffold arman el diseño en general, el AppBar se encarga de crear un encabezado y ElevatedButton añade un botón que permite hacer taps en la pantalla.

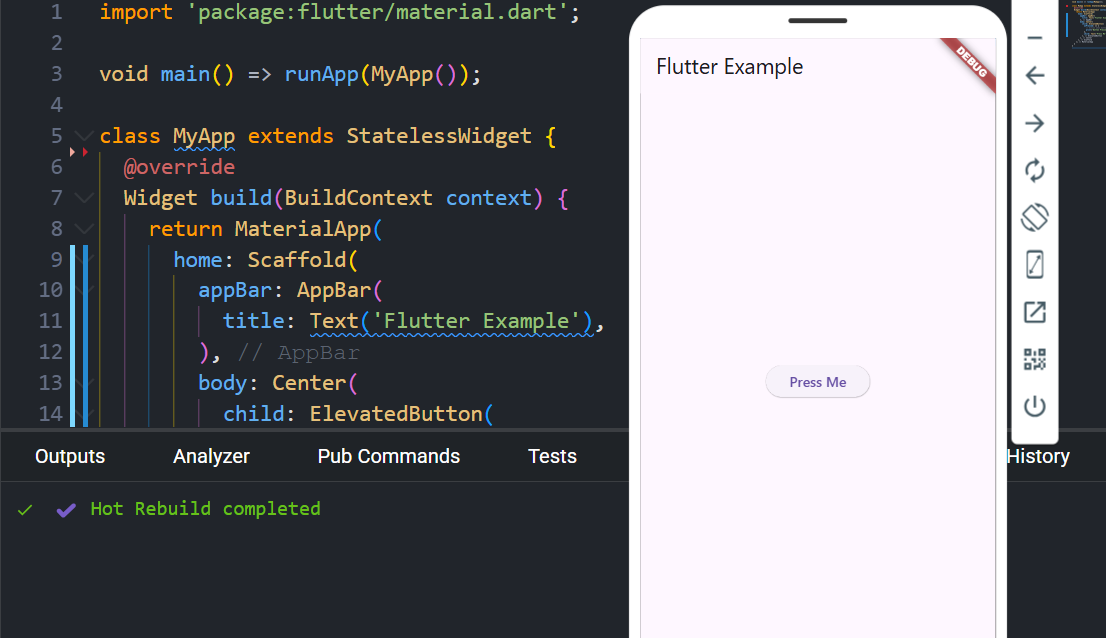


Figura 3.5 [Estructura básica de un botón que cuenta los taps] Fuente:www.browserstack.com (2025)

**Pasos para iniciar un proyecto en Flutter**

**Paso 1: Configurar el SDK de Flutter**

Descarga la última versión del SDK de Flutter desde el sitio web oficial de Flutter. Una vez descargado el archivo .zip, extráelo y mueve la carpeta Flutter a un directorio de tu elección.

. **Paso 2: Agregar Flutter a la RUTA del sistema**

Para que Flutter sea accesible desde cualquier lugar de su sistema, es una buena idea agregarlo a su variable **PATH**:

* Abra el **Panel de control** y busque Editar variables de entorno para su cuenta.
* Si existe, simplemente agregue una nueva entrada que apunte a **flutter\bin** .
* De lo contrario, cree una nueva variable llamada **Path** e ingrese la ubicación completa del directorio **flutter\bin** .
* Después reinicia el PC.

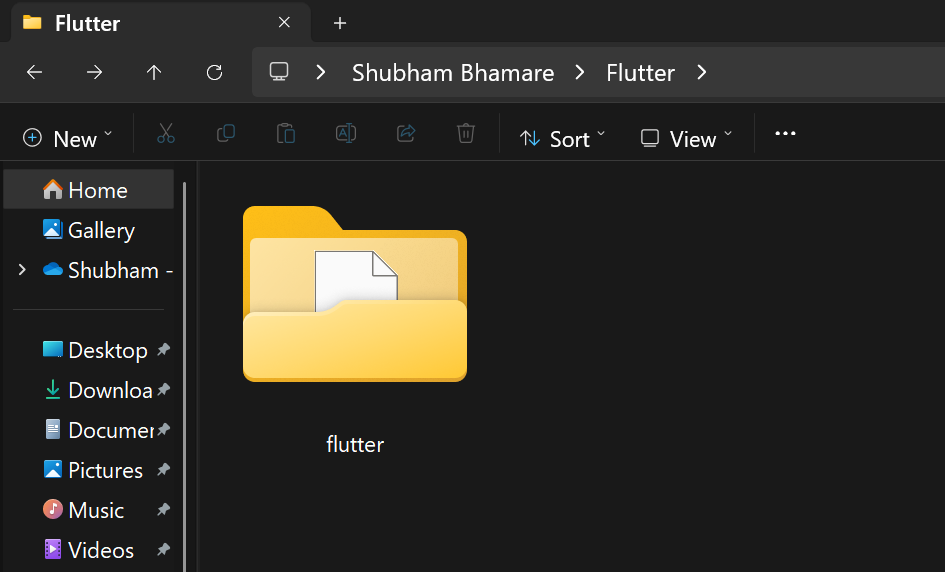


Figura 3.6 [Ruta de la carpeta Flutter] Fuente:www.browserstack.com (2025)

**Paso 3: Instalar Android Studio**

Flutter requiere herramientas de Android, que vienen incluidas con **Android Studio**:

* Descargue e instale **Android Studio** desde su sitio web oficial.
* Abra Android Studio e instale los **SDK y las herramientas de compilación**.

**Paso 4: Instalar o tener ya instlado VS Code**

Aunque Android Studio es un entorno de desarrollo completo, muchos desarrolladores prefieren **Visual Studio Code (VS Code)** por su rendimiento ligero y rápido.

En esta guía, usaremos VS Code para la demostración. Sin embargo, puedes usar cualquier editor de código que prefieras.

* Descargue VS CODE del sitio oficial
* Se recomienda instalar estas dos extensiones experiencia son las extensiones **de Flutter** y **Dart**.

**Paso 5: Verificar la instalación con Flutter Doctor**

Flutter proporciona una herramienta de línea de comandos llamada flutter doctor, que verifica si todo está configurado correctamente:

Abra **CMD** y ejecute: **flutter doctor**

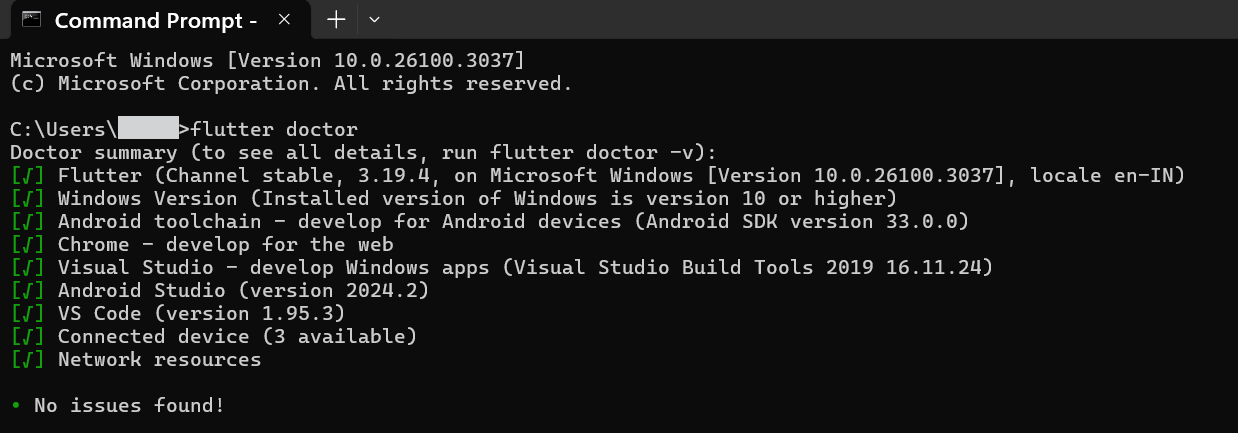
Este comando analizara si llegara a hacer falta algunas de las dependencias:

Figura 3.7 [Comando de flutter doctor] Fuente:www.browserstack.com (2025)

**Paso 6: Crea un proyecto Flutter**

Ve a la carpeta donde quieres crear tu proyecto de Flutter. Haz clic derecho y selecciona Abrir **en la terminal**. Luego, escribe el siguiente comando:

***flutter create*** *nombre\_de\_tu\_proyecto*

Si su proyecto se llama **flutterbrowserstack** , ejecuta:

***Flutter create Flutterbrowserstack***

Esto creará un nuevo proyecto Flutter con los archivos y la estructura de carpetas necesarios.

**Otro método**: Ve a la carpeta de tus proyectos abre VS Code en esa ruta, seguido haz la siguiente combinación de teclas Cntrl + Shift + P te saldrá un menú donde le podrás dar a Flutter: New Project

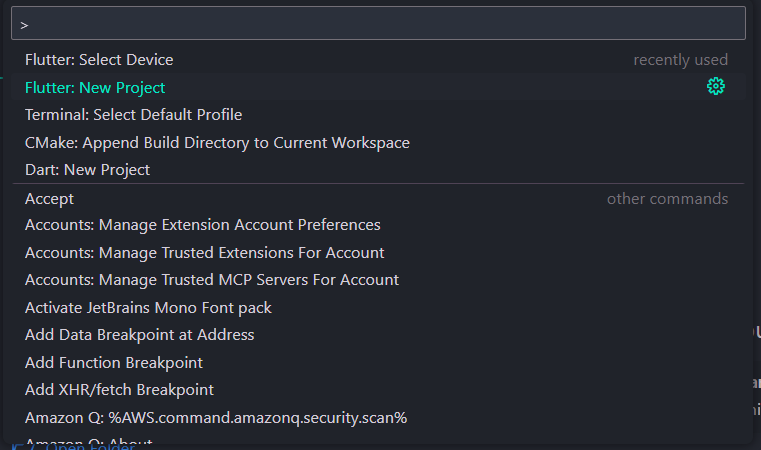


Figura 3.8 [Crear proyecto desde VS Code] Fuente: Propia

Después de eso te saldrá esta parte donde puedes elegir si crear una aplicación totalmente vacía “Empty Aplication” o con un breve comentario “Aplication”

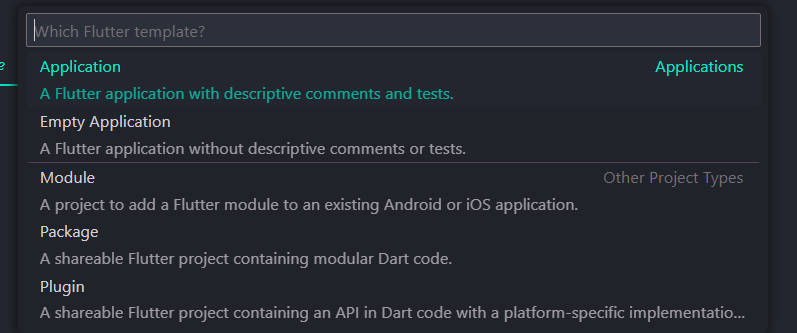


Figura 3.9 [Crear proyecto de Flutter desde VS Code tipo de aplicación] Fuente: Propia

**Paso 7: Abre tu proyecto Flutter en VS Code**

Para abrir su proyecto Flutter recién creado en **VS Code**, siga cualquiera de estos métodos:

* **Método 1:** En VSCode, haga Ctrl + Shift + E y le das a Open folder en la carpeta **flutterbrowserstack** y seleccione " **Abrir carpeta**".
* **Método 2:** Abra una terminal, navegue a la carpeta del proyecto y ejecute:
* **“code .”** dentro del directorio del proyecto.

**Paso 8:** Correr tu aplicación en VS Code

* **Método 1:** Dentro de VS Code dentro de la clase principal de Dart presiona F5.
* **Método 2:** Dentro de una terminal dentro de la ruta ejecuta flutter run.

**2. Motor Flutter Engine**

En Flutter se encuentra Flutter Engine, escrito en C++. Este motor se encarga de renderizar la interfaz el (IU) de la aplicación,que a su vez utiliza **Skia,** una biblioteca gráfica, para dibujar los componentes en pantalla.

**3. Integrador (Embedder)**

La capa Embedder actúa como puente entre la aplicación Flutter y el sistema operativo en pocas palabras es como un puente entre la aplicación de flutter y el sistema operativo. Permite que Flutter interactúe con el código nativo y se integre con las API de la plataforma, el Embedder es específico dependiendo cada plataforma (Android, iOS, web, escritorio).

Gestiona funciones específicas como la integración con servicios a nivel de sistema y el manejo de dispositivos de entrada (táctil, teclado, etc.).

**Ventajas de Flutter**

Flutter hace que el desarrollo de aplicaciones sea veloz, eficiente y económico. Grandes empresas y desarrolladores FreeLancer.

* **Único código: para múltiples plataformas:** Con un único código se puede adaptar a múltiples dispositivos
* **UI atractiva con widgets personalizables**: Con muchas librerías enfocadas a los widgets haciendo una mejor implementación con flutter.
* **Rendimiento alto**: “Las aplicaciones Flutter compilan código de máquina nativo, lo que garantiza animaciones fluidas y tiempos de carga más rápidos, haciéndolas sentir como aplicaciones nativas.”
* **Código abierto**: Flutter al ser de código abierto la comunidad puede hacer modificaciones y contribuir haciendo mejoras.
* **Integración flexible**: funciona con tecnologías y servicios existentes.
* **Recarga activa para un desarrollo más rápido**: una de las funciones más populares de Flutter, Recarga activa, permite a los desarrolladores ver los cambios de código instantáneamente sin reiniciar toda la aplicación, lo que facilita la depuración y los ajustes de la interfaz de usuario.
* **Soporte de Google**: Flutter es creado por Google lo que permite que tenga un gran soporte por parte de este, siendo actualizando frecuentemente.

**Limitaciones de Flutter**

Flutter da muchas ventajas, sin embargo, es importante saber sus limitaciones.

* **Compatibilidad nativa limitada**: Flutter podría no tener compatibilidad integrada con todas las funciones o características nativas del dispositivo, lo que significa que podrías tener que recurrir a otros frameworks ya que no tendría control total de las funciones nativas del teléfono como: Huella Dactilar, Bluetooth, NFC, entre otros.
* **Bibliotecas y herramientas limitadas**: El ecosistema de Flutter ha estado creciendo, sin embargo. tiene menos librerías a diferencia de otros frameworks como React Native.
* **Mayor tamaño de la aplicación**: Las aplicaciones desarrolladas en Flutter pueden llegar a ser más pesadas que las aplicaciones nativas. Esto se debe a que Flutter incluye su propio motor de renderizado.
* **Apariencia**: Flutter utiliza su propio motor de renderizado para crear una interfaz de usuario en vez de usar los componentes de Android y de iOS.

**Casos de uso de Flutter:**

* **Aplicaciones móviles**: desarrollo de aplicaciones con múltiples funciones en Android e iOS.
* **Plataformas web**: Plataformas web que se integran en una móvil.
* **Creación de prototipos**: Desarrollo rápidamente los conceptos de UI/UX
* **Sistemas integrados**: Creación de aplicaciones con integración de dispositivos iot (browserstack, 2025) [Traducción propia]
  + 1. Dart.

Es un lenguaje de programación creado por la empresa Google en el año 2011 con el objetivo de ser una alternativa a JavaScript, fue desarrollado por Lars Bak y Kasper Lund. Su popularidad creció gracias justamente al framework de Google, Flutter en año 2017 así ganado más popularidad entre la comunidad de programación. (manualweb, 2024)

* 1. Servicio en la Nube

Un servicio en la nube un proveedor atravé es una empresa externa que proporciona u ofrece los recursos de computación escalables a los que pueden acceder bajo demanda a través de una vía remota en la red, como computación, almacenamiento, plataforma y aplicaciones basados en la nube.

* + 1. Bases de datos en la nube.

Las bases de datos en la nube permiten tener una colección de los datos estructurados, no estructurados y semiestructurados, proporciona ventajas a diferencia de las bases de datos locales, al estar alojadas en un servidor con ingresar una cuenta vinculada tienes tus datos, incluye otros beneficios como la velocidad o escalabilidad y la facilidad y eficiencia de uso para el desarrollador. (Amazon Web Services [AWS], 2025) [traducción propia].

* + 1. Firebase

**Firebase** es una plataforma en la nube para el desarrollo de aplicaciones web y móviles, principalmente conocida por su servicio de base de datos no relacional. Cuenta con servicios como **Realtime Database** y **Cloud** **Firestore**, los cuales se encargan de sincronizar los datos en tiempo real; Firebase Auth para la autenticación de usuarios, **Firebase Analytics** para el análisis del uso y Firebase Storage, entre otros, que sirven para el desarrollo backend y que se enfoca al desarrollo móvil.  
Además, **Firebase** se integra bien con **Google Cloud** y **Flutter**, estos siendo de **Google**, lo que permite a los desarrolladores escalar sus aplicaciones conforme crecen las necesidades del proyecto, contando con la peculiaridad de soporte multiplataforma (**Android, iOS, Web y Flutter**).**Firebase Realtime Database**.

**Firebase Realtime Databas**e es una base de datos no relacional (noSQL) en tiempo real, se alojada en formato JSON en la nube se sincroniza en tiempo real con cada usuario una alternativa **a Realtime Database** es **Cloud Firestore** (Google, 2025)

* + 1. Firebase Firestore

Cloud firestore es una base de datos noSQL que se aloja en la nube de Firebase que permite las conexiones con tus apps Web o Móviles compatible con la mayoría de los sistemas operativos. (Google, 2025)

* + 1. Firebase Storage

Cloud Storage es un servicio de almacenamiento en la nube que forma parte de Firebase este nos permite almacenar contenido como imágenes videos y otros archivos como .pdf .docx o .xls multimedia de forma segura y escalable en la nube de Firebase es uno de los servicios que nos ofrece la plataforma de Firebase para el alojamiento de archivos en la nube.

* + 1. (Firebase Auth)

La mayoría de apps tienen la necesidad de identificar a sus usuarios; se necesita conocer la identidad de un usuario, lo que permite que una app guarde sus datos en la nube de forma segura y proporcione una experiencia similar para todos sus dispositivos, enfocado en él. **Firebase Authentication** proporciona servicios de backend y SDK sencillos, cuenta con librerías IU ya elaboradas para la autenticación de los usuarios en las apps de los desarrolladores. Es un sistema de autenticación que ya está elaborado por Google, únicamente listo para implementarlo en las aplicaciones. La autenticación es mediante contraseñas, números de teléfono y proveedores de identidad federada populares, como son las redes sociales Facebook o Twitter, etc.  
**Firebase Authenticatio**n se integra con otros servicios de Firebase y aprovecha estos mismos como OAuth 2.0 y OpenID Connect, por lo que se puede integrar con fácilmente en tu backend personalizado. (firebase, 2025).

* + 1. API

Significa, interfaz de programación de aplicaciones. En el contexto de las API, la palabra aplicación se refiere a cualquier software con una función distinta. La interfaz puede considerarse como un contrato de servicio entre dos aplicaciones. Se podría decir que funciona como un intermediario entre un servicio y tu ejemplo como un mesero y un chef tu ordenas y el mesero lo comunica con el servicio real el cheft. (Amazon Web Services [AWS], 2025)

* 1. Bases de datos

Una base de datos, es un medio electrónico se componen por permitir la facilidad de manipular la información pudiendo tener una gestión de los datos almacenados existen múltiples tipos de Base de datos, pero las más usadas por los desarrolladores son las bases de datos relacionales (SQL) que se manejan por uso de tablas también deben ir normalizadas, no repetir campos y organizarlos en una sola y las otras más usadas por los desarrolladores son las no relacionales (no SQL) que a diferencia de las relacionales, las no relacionales no se manejan por tablas sino por documentos y por colecciones permitiendo repetir campos y en vez de almacenar los datos en una tabla se almacenan por medio de las colecciones y los documentos.

Una base de datos es un repositorio digital donde se almacenan colecciones enormes de información. Se almacenan de distintas maneras; ejemplo: las bases de datos relacionales almacenan la información en tablas, mientras que las bases de datos no relacionales se almacenan en documentos y colecciones.

“Una base de datos es un sistema para almacenar y gestionar datos, que abarca desde el hardware físico en el que se almacenan los datos hasta el software que organiza y controla el acceso a los datos.”

También cabe señalar que Microsoft Excel no es una base de datos, debido a que mucha gente lo toma como una base de datos, cuando es una aplicación de hoja de cálculo. Una hoja de cálculo de Excel organiza los datos en filas y columnas de forma muy similar a como se realiza en una base de datos relacional, pero las hojas de cálculo son un único archivo.

En cambio, las bases de datos son sistemas sólidos de una forma más centralizada que pueden almacenar muchos tipos diferentes de datos, en muchos formatos diferentes. Esto permite realizar consultas avanzadas.

Por último, cabe señalar que hay distintos tipos de organización de base de datos, como lo son:

* Bases de datos relacionales (SQL)
* Bases de datos no relacionales (NoSQL)
* Bases de datos en la nube
* Bases de datos orientadas a objetos
* Bases de datos vectoriales
* Bases de datos de navegación

Siendo las más usadas las bases de datos relacionales y no relacionales. (ibm, 2024)

* + 1. Bases de datos relacionales

Las bases de datos relacionales están conformadas por filas y columnas que colectivamente forman tablas pueden estar conformadas por varias tablas que se relacionan por llave primaria y llave foránea se usan consultas SQL para la búsqueda de datos.

Normalmente este tipo de base de datos están normalizadas, es decir la normalización no solo evita campos repetidos, también busca **reducir la redundancia** y **mejorar la integridad de los datos**

Algunos ejemplos de base de datos relacionales son: MySQL, PostgreSQL Oracle Data base SQL Server etc. (ibm, 2021)

* + 1. Bases de datos no relacionales

Las bases de datos no relacionales o también conocidas como bases de datos (NoSQL) son aquellas bases de datos las cuales no utilizan el modelo tabular de filas y columnas que se encuentra en la mayoría de bases de datos tradicionales.

En su lugar las bases de datos no relacionales usan un modelo que esta optimizado para los requisitos específicos del tipo de datos,

Estos se pueden almacenar como: pares clave/valor simple, como documentos JSON o como un grafo que consta de bordes y vértices. Todos estos almacenes de datos tienen algo en común que es que no utilizan el modelo relacional.

La definición de NoSQL se refiere a que no utilizan el modelo SQL para las consultas, en su lugar usan otros lenguajes de programación para las consultas. El significado de NoSQL es “No relacional” normalmente hay distintos tipos de almacenes de base de datos, pero la forma más común de almacenar los datos es en documentos JSON, mediante colecciones y documentos. (Microsoft, 2025)

Algunas de las bases de datos no relacionales son:

* MongoDB
* Firebase
* Cassandra
* Redis
* Neo4j
* DynamoDB

(datacamp, 2024)

* + 1. Repositorio de Código

Un repositorio es un sistema donde se almacena y se organiza una colección de datos, recursos o archivos importantes durante la elaboración de un proyecto, es el lugar donde almacenas y guardas el código para tener un control de las versiones del proyecto, desde su fase inicial hasta su fase final, sirviendo como una línea del tiempo que rebobina los cambios hechos durante la elaboración para así poder tener un orden, gestionándolo mejor.

* + 1. (UI)

El “UI” o también conocido como (User Interface) es todo ese apartado visual con el que interactúa el usuario, incluyendo: botones, colores menús iconos tipografías fuente de letras, personalización. Se enfoca totalmente en la apariencia de la aplicación final teniendo un diseño llamativo con el fin satisfacer los colores y el diseño objetivo así haciendo que el usuario final sea el que se sienta satisfecho con el diseño que debe tener y permitiendo personalidad como más le guste.

* + 1. (UX)

El “UX” o también conocido como (User Experience) es aquel apartado donde el usuario siente una comodidad y satisfacción con el producto debido a la eficacia, ser intuitivita, o fácil manejo del producto final, como un fácil manejo de la aplicación o ser muy comprensible con el usuario final cubriendo todas las necedades que este mismo tiene siendo bastante cómodo de usar y amigable con el usuario general.

* + 1. Programación Orientada a objetos

La programación orientada a objetos

* 1. Herramientas de desarrollo.

Aquí mencionaremos las principales herramientas para el desarrollo de la App Móvil que nos sirvieron para la elaboración de esta misma, son aplicaciones que se usaron para el desarrollo del proyecto de “Urbanos y Sub Urbanos” con el fin de conocer cuáles son estas tecnologías empleadas durante la elaboración del proyecto.

* + 1. WhatsApp

WhatsApp es una aplicación de comunicación vía SMS fue lanzada al público en 2009 principalmente para dispositivos iOS, mientras que un año más tarde llegaría a Android revolucionando la mensajería tradicional como la conocemos hoy en día esta nos permite el envío de mensajes, audios, videos, fotos, documentos contactos, gifs, stickers y demás. (arimetrics, 2025)

Está aplicación fue de gran ayuda debido a que permite el contacto con la empresa siendo bastante importante para la comunicación.

* + 1. Chatgpt.

ChatGPT es un chatbot de IA generativa desarrollado por OpenAI y potenciado por su familia patentada GPT de modelos de inteligencia artificial generativa (gen AI) . Emplea procesamiento de lenguaje natural (PLN) para mantener conversaciones realistas con los usuarios y generar contenido que incluye artículos, resúmenes de textos, consejos y más.

**ChatGPT**, una de las implementaciones más conocidas de IA conversacional, se ejecuta en los modelos de transformadores generativos preentrenados (GPT) de OpenAI. En el momento de escribir este artículo, **ChatGPT** se basa en el modelo multimodal de IA **GPT-4o**, lo que le permite generar imágenes y mantener conversaciones de audio, además de generar texto.

**ChatGPT** ha atraído una gran atención de los medios desde su lanzamiento inicial en 2022. Su popularidad ha contribuido en gran medida al auge actual de la IA, incluido el desarrollo de productos de la competencia, como Google Gemini (anteriormente conocido como Bard) y Claude de Anthropic .

Sin embargo, ChatGPT también ha inspirado preocupaciones en torno a cuestiones relacionadas con la Tecnología de Modelos de lenguaje extensos/grandes, incluido el plagio y el uso de IA generativa para crear información errónea y reemplazar a los trabajadores humanos. (ibm, 2025).

* + 1. Google Drive

Google Drive es una plataforma gratuita que te permite almacenar archivos y acceder a ellos desde cualquier lugar utilizando la nube, allí también puedes crear documentos, hojas de cálculo, presentaciones y más.

Para acceder a esta plataforma, debes tener una cuenta de Google. En la página ¿Cómo iniciar sesión en Google Drive?, te explicaremos el proceso paso a paso para hacerlo. Todas las cuentas de Google Drive incluyen 15 Gigabytes de almacenamiento gratuito, si deseas obtener más espacio podrás pagar una suscripción mensual o anual para tener mayor almacenamiento en tu cuenta.

* Esta plataforma cuenta con varias aplicaciones para crear y compartir archivos, entre las más populares están:
* Documentos: puedes redactar cartas, volantes, ensayos y otros archivos de texto, similares a los documentos de Microsoft Word.
* Hojas de cálculo: te permiten almacenar y organizar información, similar a los libros de trabajo de Excel.
* Presentaciones: aquí puedes crear diapositivas.
* Formularios: recopilar y organizar datos.

(gcfglobal, s.f)

* + 1. Word

Microsoft Word es un procesador de textos que te permite crear documentos como lo son, cartas, informes, trabajos académicos y mucho más. Este programa forma parte de Microsoft Office, permitiéndote compartir tus documentos en las versiones más nuevas. Usar Microsoft Word es bastante sencillo, primero, compra la licencia de Office 365 en el cual ya viene incluido Word, Excel, PowerPoint entre otros, con esto ya tienes acceso a Microsoft Word.

Word tiene muchas ventajas para la creación de documentos profesionales o personales de forma sencilla, su entorno está enfocado al área laboral como lo podría ser un profesional o algún alumno de cualquier grado escolar el programa es muy fabuloso en funciones facilita el formateo del texto y la incorporación de imágenes corrector ortográfico, tablas, funciones de títulos, estilos, el cambio de color para las letras palabras, incluso múltiples fuentes de letra para la personalización según lo requiera el usuario etcétera, etcétera.

Este programa fue desarrollado por Charles Simonyi y fue lanzado por Microsoft en el año 1983, se diseñó para computadoras de IBM con el sistema operativo MS-DOS. (lenovo, 2025)

* + 1. Adobe Photoshop

Adobe Photoshop es una aplicación de edición y diseño de imágenes desarrollado por la compañía **Adobe Systems**, fue creado en el año 1986 por los hermanos Knoll.

Esta es bastante popular entre artistas, fotógrafos y diseñadores para múltiples tareas distintas como lo son: edición de imágenes, dibujo, creación de diseños, corrección de imágenes, entre otros.

**Ventajas**:

* Tiene muchas herramientas, con las que puedes editar tus fotos, vídeos e ilustraciones, tales como la luz, las sombras, el brillo, el contraste, entre otros.
* Te permite editar y exportar tu trabajo a diferentes formatos: PSD, EPS, DCS, BMP, GIF, JPEG, TIFF, PICT, PNG, PDF, ICO, RAW, TGA, JPEG2000, Webp, etc.
* Puedes trabajar en Photoshop desde cualquier dispositivo, ya sea una computadora, una tablet o un teléfono.
* Se actualiza constantemente.
* Es una herramienta que aumenta tus posibilidades de trabajo.

**Desventajas**:

* Su precio es muy elevado comparado con otras herramientas en el mercado.
* Necesitas una computadora bastante potente con un buen procesador y bastante ram para que el programa pueda trabajar correctamente.
* Al ser una suscripción es complicado su accesibilidad.

(ebac, 2023)

* + 1. Android Studio

Android Studio es un IDE de desarrollo para la elaboración de aplicaciones móviles en dispositivos Android, fue desarrollado por Google en diciembre del 2014, es un IDE que cuenta con múltiples herramientas funcionales para los desarrolladores una de las más usadas por los desarrolladores Android Emulador.

* + 1. Visual Studio Code

Visual Studio Code, abreviado como VS code, es un editor de código siendo desarrollado por Microsoft siendo lanzado al público en año 2016 sin embargo fue anunciado al publico 1 año antes en la conferencia Build de 2015 lanzando una versión preliminar, fue desarrollado para múltiples sistemas operativos, como Windows, MacOs y Linux.

* + 1. Git

Git es un sistema de control de versiones, para tener un control del proyecto sirve mucho cuando muchas personas trabajan en un mismo proyecto funcionando como una línea del tiempo para volver a cierto punto donde el código funcionaba bien o navegar entre versiones anteriores de un proyecto de manera local.

* + 1. Git Hub

Git Hub es una plataforma en la web que funciona como un alojamiento en la nube para un repositorio remoto de Git para guardar y gestionar tu proyecto de manera remota implementando nuevas funciones para el trabajo en equipo, añadiendo nuevas funcionalidades como las stars y los forks, los issues , repositorio remoto y un control de versiones a través de un repositorio remoto.

* 1. Transporte en Tlaxcala

El Origen del transporte público en Tlaxcala se remonta a 1924 gracias a el surgimiento de ATAH (Autotransportes Tlaxcala Apizaco y Huamantla esto inicio debido a la falta de transporte público en el estado, aunque en 1867 se inauguró el tren en Tlaxcala Mexico-Apizaco, le faltaban rutas a lugares específicos en el estado y no era muy accesible por esos años se podría decir que el transporte en el estado de Tlaxcala tiene origen desde 1867 pero el transporte para el sector publico tiene origen en 1924 gracias a ATAH posteriormente USU (**Urbanos y Sub Urbanos de Tlaxcala S.A de**) se daría origen a esta empresa transportista en 1968 (scielo.org, 2010) (Guzman, 2025)

* 1. Sistemas similares

Tras la búsqueda de sistemas similares, se encontró dos sistemas similares uno usando el mismo Framework (Flutter), y siendo bastante similar a este proyecto, debido que igual maneja el termino unidades de transporte, pero con un enfoque hacia el transporte de carga y otro sistema no tan parecido, pero igual está enfocado al transporte público, en este caso de Perú este siendo una aplicación para evaluar el servicio del transporte público pero la primera 3.10.1 siendo más parecida a este proyecto.

* + 1. Aplicación móvil para gestión de línea de transporte y carga (FREIGHTEM)

La tesis elaborada por el Licenciado en Ciencias Computacionales: Miguel Ángel Pérez Munguía presenta el desarrollo del sistema móvil “**FREIGHTEM**”, cuyo objetivo es crear una aplicación móvil destinada a la gestión de unidades de transporte de carga. Esta aplicación permite realizar solicitudes de asistencia, como cambios de aceite o requerimiento de refacciones, y ofrece diversos apartados funcionales similares a los del sistema en desarrollo, tales como la gestión de operadores (CRUD), el registro de datos personales y la información técnica de las unidades.

Sin embargo, ambos sistemas difieren en su orientación: mientras que el sistema **FREIGHTEM** está enfocado en el sector privado del transporte de carga, el sistema **CDSYPDT** **(Control de Servicio y Personal de Transporte Urbanos y Suburbanos de Tlaxcala)** se dirige al sector de transporte público.

El sistema **FREIGHTEM** resuelve la problemática de la gestión del autotransporte de carga, al permitir un control integral de tracto camiones, remolques, operadores y pólizas desde una sola aplicación móvil. Su propósito la digitalización, organizar la información, reduciendo el uso de registros en papel y facilitando el monitoreo de operadores, además de brindar asistencia técnica mediante solicitudes de refacciones y servicios.

En conclusión, la finalidad del sistema **FREIGHTEM** es tener un control de papeleo y de gestión en la parte de la administración, y en la parte del personal pueden solicitar refacciones para garantizar su seguridad de los operadores y de las unidades, implementando historial de cambios de aceite, algo que cabe mencionar es que ambos sistemas usan el Framework (Flutter). (Munguía, 2025)

En la siguiente **figura 3.10** podemos ver como el autor: **Miguel Ángel Pérez Munguía** nos muestra un apartado donde, tenemos los datos del operador, pero este proyecto se diferencia, debido a que tiene campos diferentes con el proyecto **CDSYPDT** en cambio el sistema **FREIGHTEM** cuenta con otros campos diferentes, ejemplo esté sistema tiene parentesco, Número de **INE, RFC, NSS** a diferencia del de **USU** debido a que los sistemas a desarrollar no se verán necesario usar esos campos.

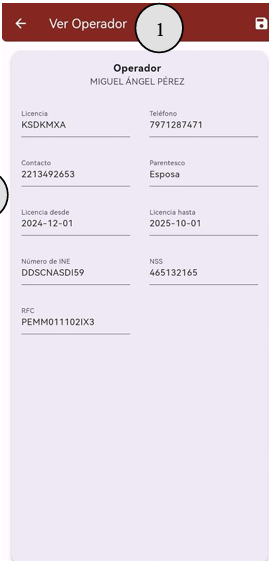


Figura 3.10 [Sistemas similares] Fuente: Pérez Munguía, (2025), Aplicación móvil para gestión de línea de transporte y carga (FREIGHTEM) [Tesis de sistema].

En la siguiente figura 3.11 se puede observar como el autor: **Miguel Ángel Pérez Munguía** nos muestra la screen del registro de Operadores, con datos como: Nombre, licencia, teléfono, contacto del operador, parentesco del contacto, vigencia de la licencia, numero de INE, número de seguro social, y su RFC.

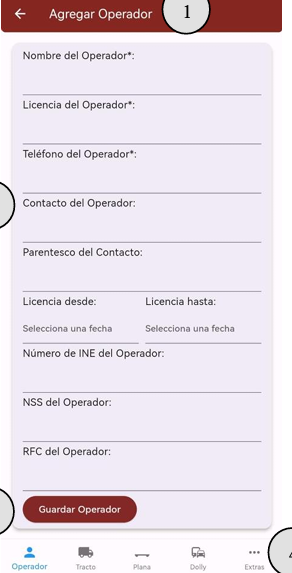


Figura 3.11 [Sistemas similares] Fuente: Pérez Munguía, M. Á. (2025), Aplicación móvil para gestión de línea de transporte y carga (FREIGHTEM) [Tesis de sistema].

En la siguiente figura 3.12 se muestra una screen parecida, con los wireframes propuestos sin embargo en el sistema CDSYPDT, implementa un atributo que no tienen otros operadores el cual es el operador posturero, capas de que una unidad tenga más de un solo operador y así mismo ver el historial de una unidad, debido que en el sistema CDSYPDT una unidad puede tener más de un operador fijo en cambio en el sistema **FREIGHTEM** los operadores son de camiones de carga, por lo tanto son fijos a diferencia de en USU.

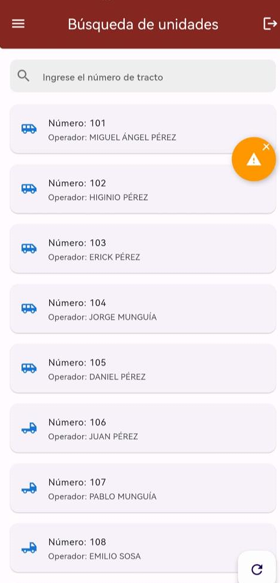


Figura 1Figura 3.12 [Sistemas similares] Fuente: Pérez Munguía, M. Á. (2025), Aplicación móvil para gestión de línea de transporte y carga (FREIGHTEM) [Tesis de sistema].

En la figura 3.13 se puede apreciar como asignar un operador es de manera manual, en cambio en el sistema de USU toma los datos automáticamente al registrar un operador se le asigna su unidad, y la app ya toma esos datos y ya sabe que al chofer Juan se le asigno la unidad #101 y en dado caso de ser posturero ellos al iniciar sesión les solicitara que unidad manejaran cada cierto tiempo ejemplo de 9:00 am a 11:00 am le va preguntar en su cuenta si cambio de unidad o sigue manejando la misma

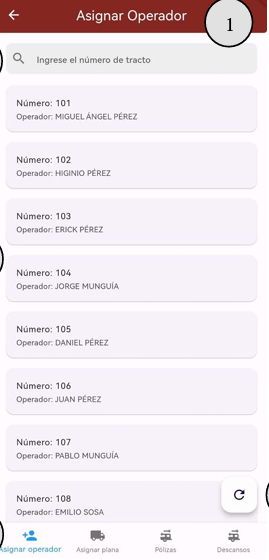


Figura 3.13 [Sistemas similares] Fuente: Pérez Munguía, M. Á. (2025), Aplicación móvil para gestión de línea de transporte y carga (FREIGHTEM) [Tesis de sistema].

* + 1. Desarrollo de un aplicativo móvil para el registro del cumplimiento y desempeño de líneas de transporte público en la ciudad de Lima

En este caso es un sistema que no se asemeja tanto, pero comparte similitudes ejemplo como el enfoque donde ambos sistemas van enfocados al sector público, en este caso el autor de esta tesis es del: Ing. de sistemas e Informática Eduardo Ydrogo Ramírez este autor nos narra en su tesina el desarrollo de su Sistema de **Desarrollo de un aplicativo móvil para el registro del cumplimiento y desempeño de líneas de transporte público en la ciudad de Lima**, como mencionaba anteriormente, en la única similitud es el sector público, pero a diferencia de CDSYPDT **(Control de Servicio y Personal de Transporte Urbanos y Suburbanos de Tlaxcala)** uno se va enfocado al control y gestión de los operadores y el cumplimiento de sus rutas asignadas, en cambio el sistema de Desarrollo de un aplicativo móvil para el registro del cumplimiento y desempeño de líneas de transporte público en la ciudad de Lima, se enfoca en la evaluación general del público , que tan buena calidad tiene el transporte de lima, siendo más este sistema como un cuestionario de los pasajeros para evaluar la calidad del transporte. (Ramírez, 2020)

CAPÍTULO IV

1. Metodología y desarrollo

El objetivo específico es establecer un sistema de monitoreo en tiempo real de la ubicación de los operadores por medio de sus dispositivos móviles. Al momento de escanear su credencial mediante un lector óptico inteligente, dicha credencial un código QR que idéntica el número de unidad vehicular la cual al llegar al punto de control escaneando su credencial mandara una notificación a la aplicación móvil que notificara a los administradores confirmando que una unidad determinada ha llegado a cierto punto de control así teniendo un control más preciso del movimiento de las unidades y sus operadores.

En el caso de los choferes de “planta”, cada uno mantiene asignada una unidad de manera fija por lo cual cuentan con un vehículo ya establecido, lo que facilita y hace más eficiente su control. Sin embargo, existen choferes denominados “postureros” quienes cambian de vehículo al llegar a determinados puntos de control. Para estos mismos la aplicación solicitara registrar que unidad operarán diariamente y, en intervalos deberán confirmar el cambio de unidad especificando la franja horaria correspondiente, ejemplo: de 10:00 a.m. a 1:00p.m. un operador maneja la unidad #23 pero a partir de la 1:00p.m. el chofer deberá cambiar de unidad a la #245.

De esta manera, el sistema genera un historial tanto de los choferes como de las unidades, permitiendo identificar que operador manejo determinada combi en un tiempo determinado y así garantizando un mayor control, flujo y eficacia de las operaciones y procesos realizados en la empresa de “USU” agilizando todo al máximo sacando un mejor provecho en todos los involucrados.

* 1. Análisis de requerimientos
     1. Historias de usuario.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Historia de Usuario** | **Criterios de**  **Aceptación** | **Prioridad** |
| **HU-01** | Como usuario del sistema, quiero iniciar sesión con mi nombre de usuario/correo electrónico en caso de no tener correo electrónico el método principal de inicio de sesión es con su nombre de usuario y su contraseña, pero si se tiene correo se puede iniciar sesión con ambos correo electrónico/email, o nombre de usuario/username. | El sistema debe validar las credenciales ingresadas y mostrar un mensaje de error si son incorrectas, redirigiendo al panel correspondiente según el rol del usuario | **Alta** |
| **HU-02** | Como usuario quiero tener un apartado de configuración para poner el sistema en modo oscuro o cambiar el tamaño de la fuente. | El sistema debe permitir activar o desactivar el modo oscuro y cambiar el tamaño de fuente, guardando las preferencias del usuario localmente | **Alta** |
| **HU-03** | Como usuario quiero cerrar sesión en mi cuenta | El sistema debe cerrar la sesión y regresar a la pantalla de inicio de sesión eliminando los datos temporales del usuario | **Alta** |
| **HU-04** | Como administrador quiero dar de alta la cuenta de un chofer para registrarlo en el sistema. | El formulario debe validar los campos obligatorios y crear el registro del chofer en Firebase mostrando un mensaje de confirmación | **Alta** |
| **HU-05** | Como administrador, quiero editar la información de un operador, para así mantenerla actualizada. | El sistema debe permitir modificar los datos existentes del operador y guardar los cambios correctamente en la base de datos | **Alta** |
| **HU-06** | Como administrador quiero eliminar choferes | El sistema debe solicitar confirmación antes de eliminar un chofer y actualizar la base de datos eliminando su registro | **Alta** |
| **HU-07** | Como administrador quiero registrar unidades | El formulario debe permitir registrar nuevas unidades verificando que los campos estén completos antes de guardar | **Alta** |
| **HU-08** | Como administrador quiero editar unidades | El sistema debe permitir modificar los datos de las unidades y guardar los cambios correctamente en Firebase | **Alta** |
| **HU-09** | Como administrador quiero eliminar unidades | El sistema debe solicitar confirmación antes de eliminar una unidad y actualizar la base de datos eliminando su registro | **Alta** |
| **HU-10** | Como administrador quiero modificar la información de cada cuenta. | El sistema debe permitir al administrador modificar la información general de cualquier cuenta registrada | **Alta** |
| **HU-11** | Como administrador quiero mandar mensajes a cualquier personal de la empresa, con mi rol en mi perfil | El sistema debe permitir el envío y recepción de mensajes en tiempo real y notificar a los usuarios cuando reciban un nuevo mensaje | **Media** |
| **HU-12** | Como administrador quiero ver mi perfil personal | El sistema debe mostrar los datos personales del administrador en una interfaz de perfil accesible desde el menú principal | **Media** |
| **HU-13** | Como administrador quiero editar mi información de mi perfil | El sistema debe permitir que el administrador edite su propia información personal y guarde los cambios correctamente | **Media** |
| **HU-14** | Como administrador solo yo puedo eliminar un perfil. | Solo los administradores podrán acceder a la opción de eliminar perfiles y el sistema debe mostrar un mensaje de confirmación antes de hacerlo | **Alta** |
| **HU-15** | Como despachador quiero dar de alta la cuenta de un chofer para registrarlo en el sistema. | El sistema debe permitir al despachador registrar nuevos choferes verificando la validez de los datos antes de guardarlos | **Alta** |
| **HU-16** | Como administrador solo yo puedo editar la información de los choferes y de los despachadores | Solo los administradores pueden editar la información de los choferes y de los despachadores asegurando el control de privilegios | **Alta** |
| **HU-17** | Como despachador, quiero editar la información de un operador, para así mantenerla actualizada. | El sistema debe permitir al despachador modificar ciertos campos limitados del operador sin acceso a funciones restringidas | **Media** |
| **HU-18** | Como despachador quiero mandar mensajes a cualquier personal de la empresa, con mi rol en mi perfil | El sistema debe permitir al despachador enviar y recibir mensajes dentro del chat interno en tiempo real | **Media** |
| **HU-19** | Como despachador quiero ver mi perfil personal | El sistema debe mostrar los datos personales del despachador en la sección de perfil de usuario | **Media** |
| **HU-20** | Como operador quiero ver mi información personal en mi perfil y si hay algún error contactar con los administradores y que la corrijan. | El sistema debe mostrar la información personal del operador en modo de solo lectura e incluir un botón para contactar al administrador en caso de errores | **Alta** |
| **HU-21** | Como operador quiero que en mi perfil diga si soy de planta o posturero | El sistema debe mostrar en el perfil del operador una etiqueta que indique si es de planta o posturero | **Media** |
| **HU-22** | Como operador posturero quiero que cada cierto tiempo me pregunte el sistema si quiero cambiar de unidad | El sistema debe solicitar al operador posturero confirmar si desea cambiar de unidad en intervalos definidos y registrar el cambio en el historial | **Alta** |
| **HU-23** | Como operador quiero mandar mensajes a cualquier personal de la empresa, con mi rol en mi perfil | El sistema debe permitir al operador enviar y recibir mensajes dentro del chat de la empresa interno en tiempo real | **Media** |

* + 1. Requerimientos funcionales

|  |  |  |
| --- | --- | --- |
| **ID** | **Requerimientos Funcionales** | **Prioridad** |
| **RF\_01** | El sistema debe permitir el inicio de sesión de los  usuarios mediante:   * nombre\_de\_usuario * correo\_electronico * password * rol | **Alta** |
| **RF\_02** | El sistema debe permitir el registro de nuevos choferes mediante:   * nombre * nombre\_de\_usuario * email * password * edad * direccion * telefono * unidad * vigencia\_licencia * vigencia\_permiso | **Alta** |
| **RF\_03** | El sistema debe permitir al administrador editar la información de choferes existentes mediante:   * nombre * nombre\_de\_usuario * email * password * edad * direccion * telefono * unidad * vigencia\_licencia * vigencia\_permiso | **Alta** |
| **RF\_04** | El sistema debe permitir eliminar choferes del sistema de manera controlada mediante:   * colección * documento | **Alta** |
| **RF\_05** | El sistema debe permitir registrar nuevas unidades mediante:   * numero\_unidad * color * placas | **Alta** |
| **RF\_06** | El sistema debe permitir editar la información de las unidades registradas mediante:   * numero\_unidad * color * placas | **Alta** |
| **RF\_07** | El sistema debe permitir al administrador eliminar las unidades mediante:   * colección * documento | **Alta** |
| **RF\_08** | El sistema debe mostrar la información personal de los usuarios mediante:   * nombre * telefono * direccion * edad * email | **Alta** |
| **RF\_09** | El sistema debe permitir al administrador modificar la información del perfil del usuario activo mediante:   * nombre * telefono * direccion * edad * email | **Media** |
| **RF\_11** | El sistema debe permitir activar el modo oscuro y modificar el tamaño de fuente en el apartado de configuración | **Media** |
| **RF\_12** | El sistema debe permitir el cierre de sesión, eliminando los datos temporales del usuario | **Alta** |
| **RF\_13** | El sistema debe registrar en la colección “historial” los accesos de los choferes mediante   * chofer * unidad * fecha\_inicio\_sesion | **Media** |
| **RF\_14** | El sistema debe permitir a los usuarios enviar y recibir mensajes dentro del chat interno en tiempo real | **Media** |
| **RF\_15** | El sistema debe mostrar el tipo de operador (Planta/Posturero) dentro del perfil del chofer | **Alta** |
| **RF\_16** | El sistema debe solicitar al operador posturero confirmar el cambio de unidad en intervalos definidos, registrando el cambio en el historial mediante:   * chofer * unidad | **Alta** |
| **RF\_17** | El sistema debe restringir la eliminación y edición de cuentas únicamente a usuarios con rol “admin” | **Media** |
| **RF\_18** | El sistema debe enviar notificaciones automáticas al administrador cuando un despachador registre o modifique un chofer | **Alta** |
| **RF\_19** | El sistema debe guardar y sincronizar en tiempo real los cambios realizados en las colecciones de Firebase | **Media** |
| **RF\_20** | El sistema debe permitir visualizar la lista general de unidades disponibles y su estado mediante:   * numero\_unidad * color * placas | **Alta** |
| **RF\_21** | * El sistema debe permitir al despachador registrar nuevos choferes mediante: nombre * nombre\_de\_usuario * email * password * edad * direccion * telefono * unidad * vigencia\_licencia * vigencia\_permiso |  |

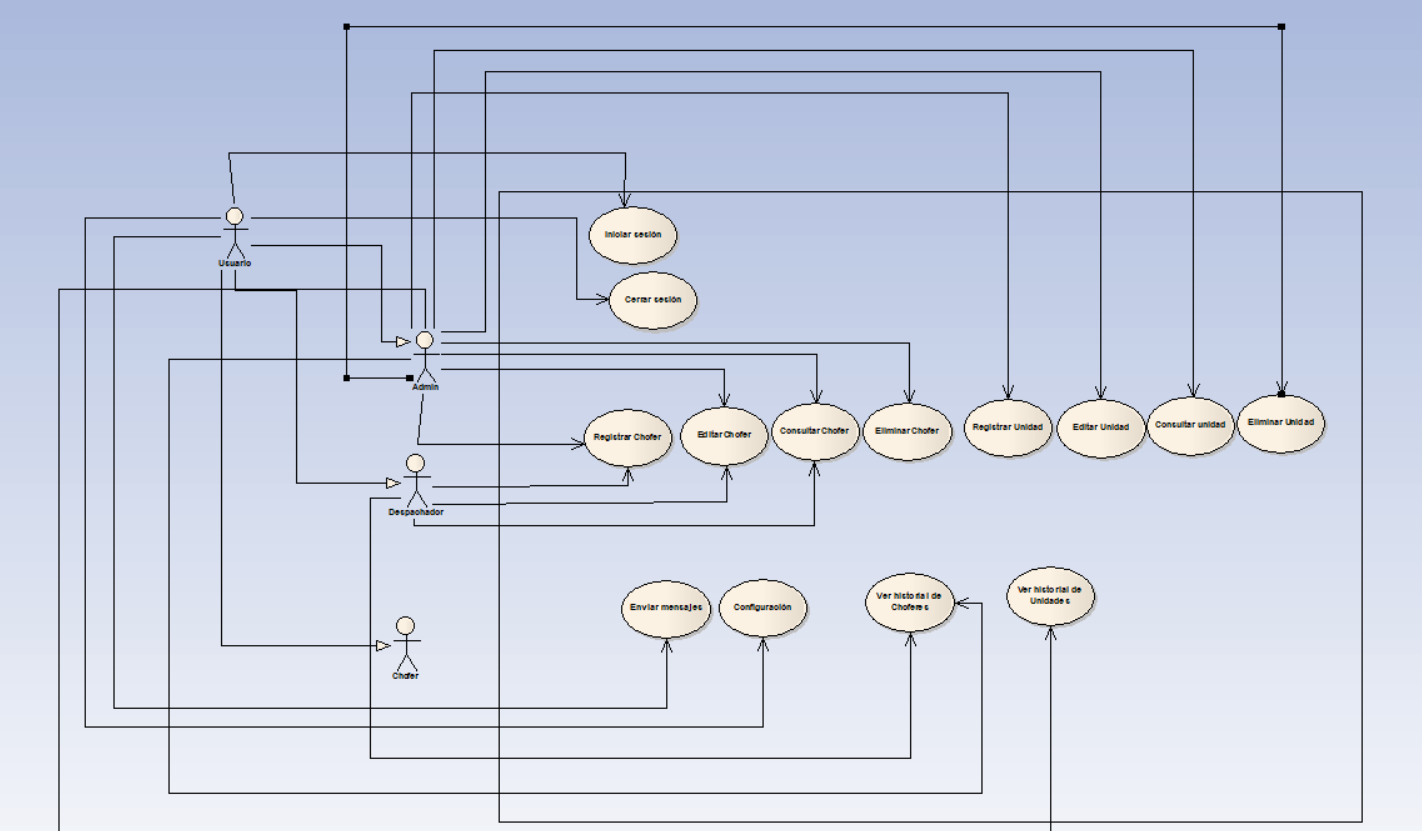
* + 1. Requerimientos no funcionales

|  |  |  |
| --- | --- | --- |
| ID | Requerimientos No Funcionales | Prioridad |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* 1. Diseño del sistema

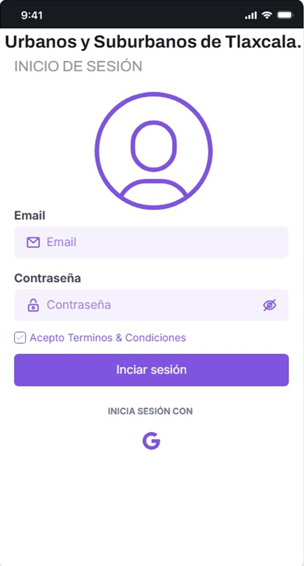
En este apartado veremos cómo se está diseñado este sistema y como está conformado

* + 1. Diagrama de casos de uso



*Figura 4 Fuente: propia*

* + 1. Diagrama colecciones y documentos
    2. Wireframes

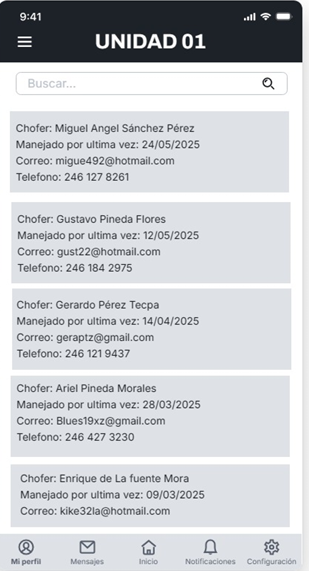


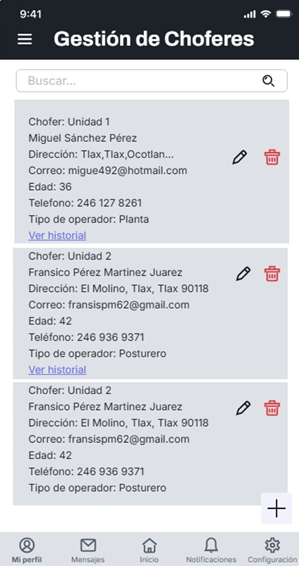




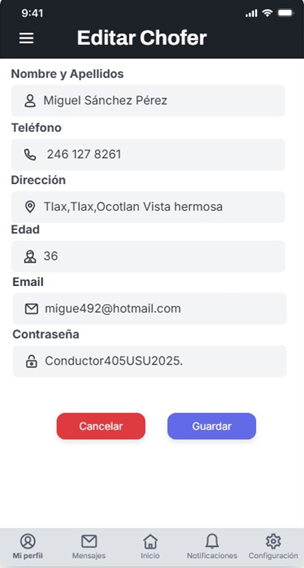




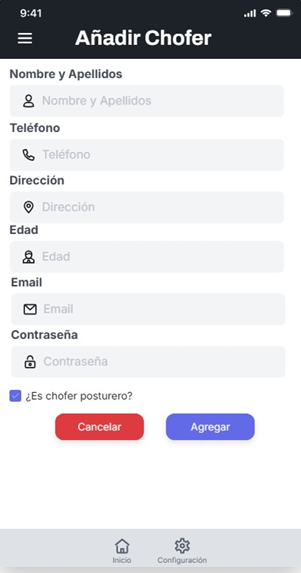


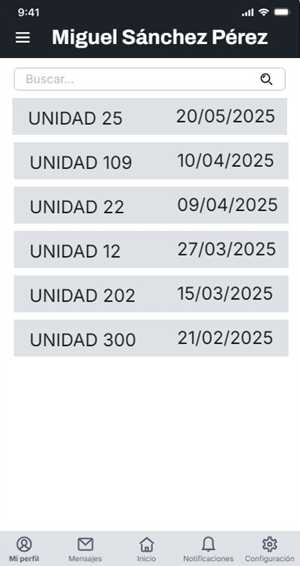






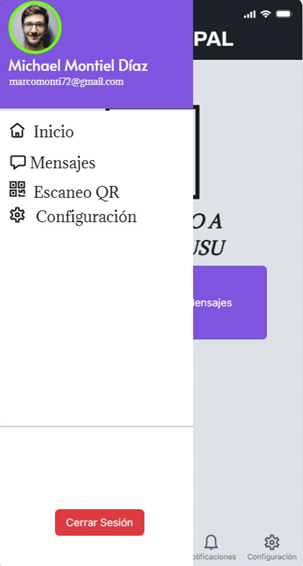


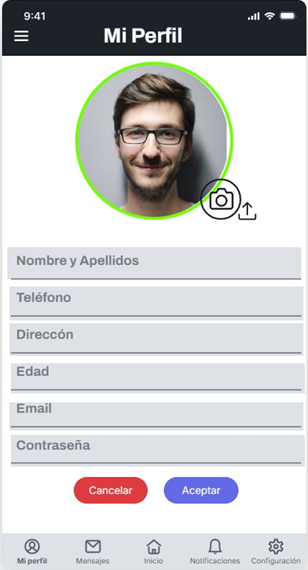




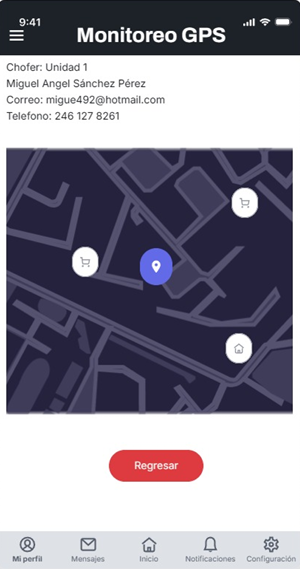






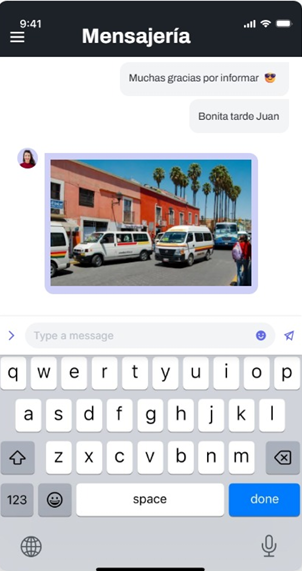


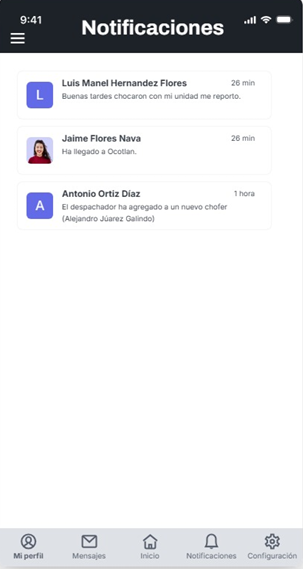


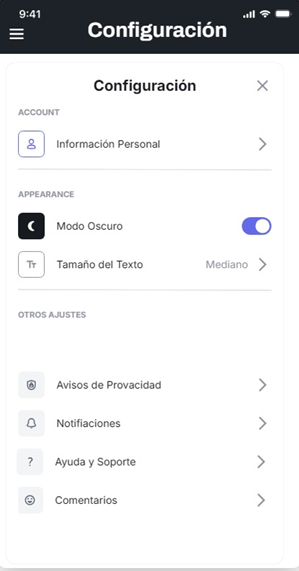


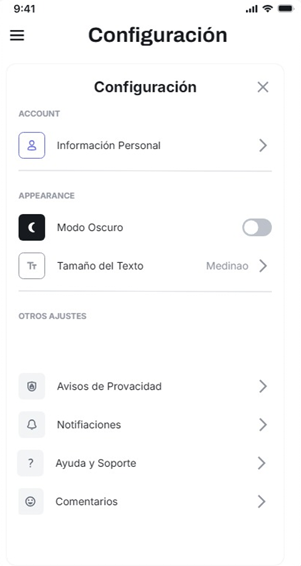












* 1. Desarrollo del sistema
     1. Análisis del sistema actual

La empresa USU a trabajado a lo largo de los años por medio

* + 1. Configuración del entorno de desarrollo
    2. Tabla de variables

|  |  |  |
| --- | --- | --- |
| Variable | Tipo de dato | Descripción |
| apiKey | String | Clave pública usada para conectar la app con Firebase. |
| appId | String | Identificador único de la app dentro del proyecto Firebase. |
| messagingSenderId | String | ID usado por el servicio de notificaciones (Firebase Cloud Messaging). |
| projectId | String | Identificador del proyecto en Firebase. |
| storageBucket | String | Ruta de almacenamiento de archivos en la nube (Firebase Storage). |
| rol | String | Rol actual del usuario (admin, chofer o despachador). |
| config | String | Configuración global de la aplicación obtenida del Provider. |
| \_usuarioCorreoController | TextEditingController | Controlador del campo de texto para el nombre de usuario o correo. |
| \_passwordController | TextEditingController | Controlador del campo de texto para la contraseña del usuario. |
| \_mostrarContrasena | bool | Indica si la contraseña debe mostrarse o mantenerse oculta. |
| \_cargando | bool | Indica si el proceso de inicio de sesión está en curso. |
| \_usuarioEncontrado | String | ID del documento de usuario encontrado en la colección gestion\_usuarios. |
| usuarioId | String | Identificador único del usuario cargado desde la base de datos. |
| fotoUrl | String | URL de la foto de perfil almacenada en Firebase Storage. |
| \_nombreController | TextEditingController | Controlador del campo de texto del nombre del usuario. |
| \_telefonoController | TextEditingController | Controlador del campo de texto del teléfono del usuario. |
| \_direccionController | TextEditingController | Controlador del campo de texto de la dirección del usuario. |
| \_edadController | TextEditingController | Controlador del campo de texto para la edad (convertida a int al guardar). |
| \_emailController | TextEditingController | Controlador del campo de texto del correo electrónico. |
| \_usuarioController | TextEditingController | Controlador del campo de texto del nombre de usuario. |
| \_unidadController | TextEditingController | Controlador del campo de texto de la unidad asignada. |
| \_searchController | TextEditingController | Controlador del campo de búsqueda. |
| \_vigenciaLicenciaController | TextEditingController | Controlador del campo de texto de la vigencia de la licencia. |
| \_vigenciaPermisoController | TextEditingController | Controlador del campo de texto de la vigencia del permiso. |
| \_esPosturero | bool | Indica si el operador es “Posturero”. |
| \_modoFormulario | bool | Indica si el formulario está activo o cerrado. |
| \_modoEdicion | bool | Indica si se está editando un chofer existente. |
| \_busqueda | String | Texto actual de búsqueda o filtro. |
| \_idEditar | String | ID del documento que se está editando en Firestore. |
| nombreChofer | String | Nombre del chofer mostrado en el historial. |
| choferId | String | Identificador del chofer en el mapa. |
| nombre | String | Nombre del chofer (se muestra en la UI). |
| correo | String | Correo electrónico del chofer (mostrado en pantalla). |
| telefono | String | Teléfono del chofer (mostrado en la interfaz). |
| lat | double | Latitud de la ubicación del chofer. |
| lng | double | Longitud de la ubicación del chofer. |
| esChofer | bool | Indica si el usuario pertenece al rol “chofer”. |
| esOscuro | bool | Indica si el tema actual de la app es oscuro. |
| fecha\_inicio\_sesion | Timestamp | Fecha y hora del inicio de sesión del chofer (Firestore). |
| color | String | Color asignado a la unidad (por ejemplo “verde”). |
| numero\_unidad | String | Número identificador de la unidad. |
| placas | String | Placas del vehículo registrado. |

* + 1. Integración de Firebase y APIs
    2. Pruebas y corrección de errores
    3. Documentación técnica
    4. Conclusión del desarrollo

CAPÍTULO V

RESULTADOS

Resultados

En la siguiente imagen se puede apreciar el Login con su respectivo inicio de sesión se puede ingresar con el correo o con el nombre de usuario

En la siguiente Figura 5.1 se aprecia los “**términos y condiciones**” para el uso de la aplicación si el usuario dado de alta no acepta los términos y condiciones no podrá usar la aplicación “Urbanos y Sub Urbanos de Tlaxcala”



Figura 5.1 [Login/Inicio de Sesión] Fuente: Propia

En la siguiente figura 5.2 se muestra el apartado de términos y condiciones donde el usuario debe aceptar los términos y condiciones, que necesita para poder a utilizar la aplicación y sus funcionalidades.

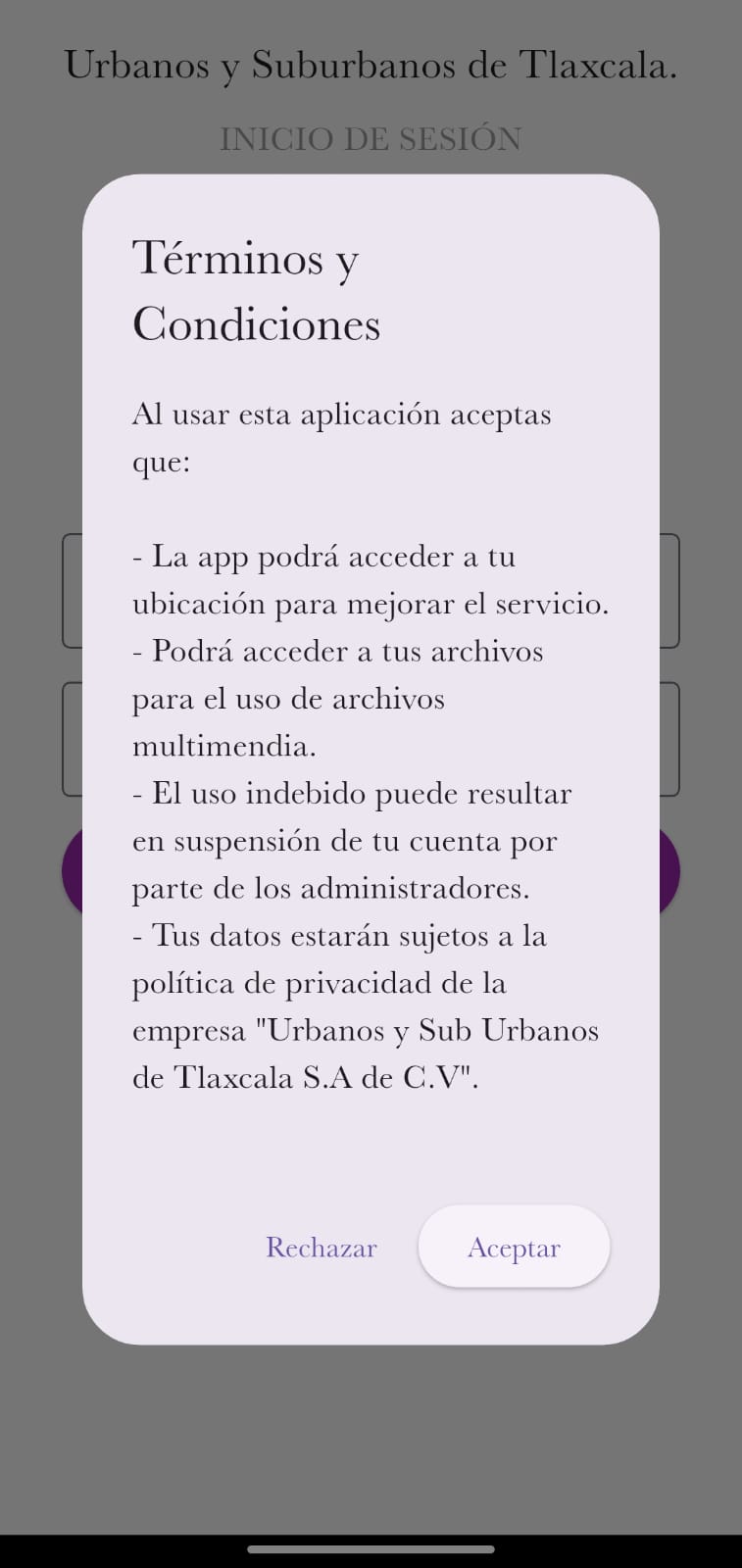


Figura 5.2 [Términos y Condiciones] Fuente: Propia

En la siguiente figura 5.3 se muestra el menú principal donde podemos ver las opciones principales donde los botones y módulos cambian según el rol.



Figura 5.3 [Menú Principal] Fuente: Propia

En la siguiente figura 5.3 se muestra la barra lateral donde podemos ver los módulos según el rol y más aparte el apartado para cerrar sesión más aparte muestra el rol, su nombre, su correo (si es que tiene correo electrónico) y sino muestra su nombre de usuario.

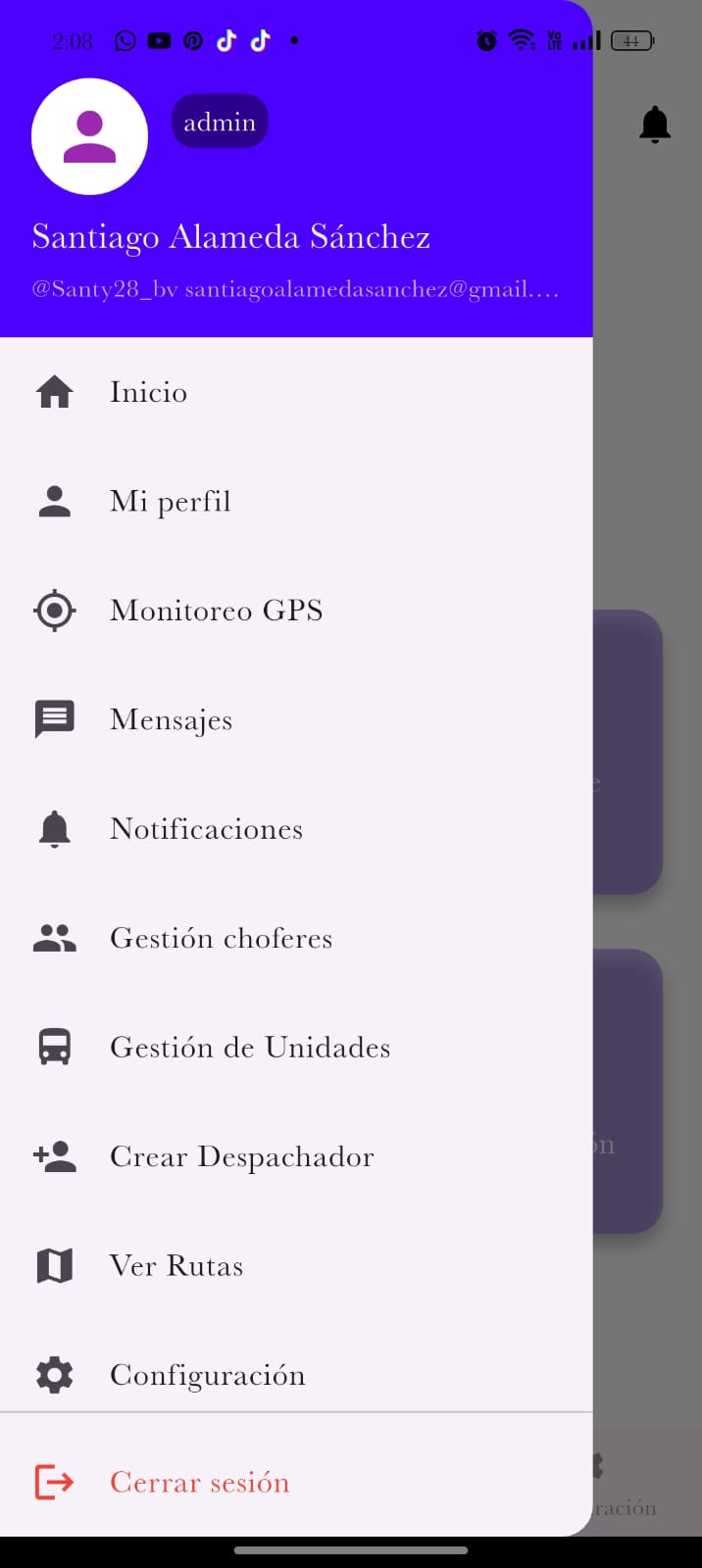


Figura 5.3 [Barra lateral de navegación] Fuente: Propia

En la siguiente figura 5.4 se muestra el modulo de gestión de choferes donde podemos ver apartados básicos como lo son registrar un chofer, editar, eliminar o verlos así mismo muestra su historial donde nos dice las unidades que a manejado algún chofer, ahora en datos del registro tenemos datos como: nombre, numero de unidad que manejan, nombre de usuario, su dirección, su correo electronico, su edad, su número de teléfono, el tipo de operador que son planta o posturero y su vigencia de permiso de conducir en la empresa y su vigencia de licencia de conducir.

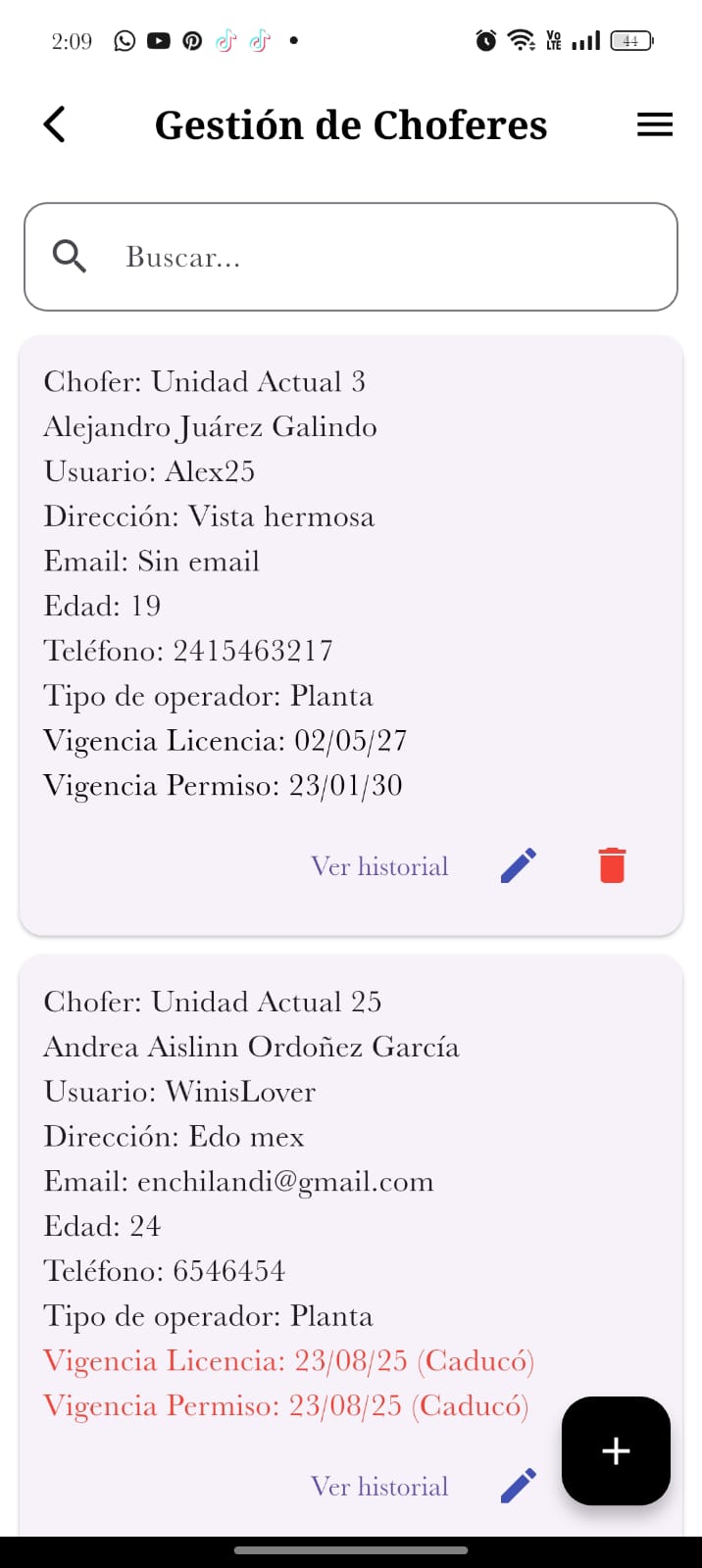


Figura 5.4 [Gestión de choferes] Fuente: Propia

La siguiente figura 5.5 muestra el módulo de registro de la cuenta de un chofer con sus respectivos datos, el email es opcional, y se puede elegir un chofer de planta o posturero.

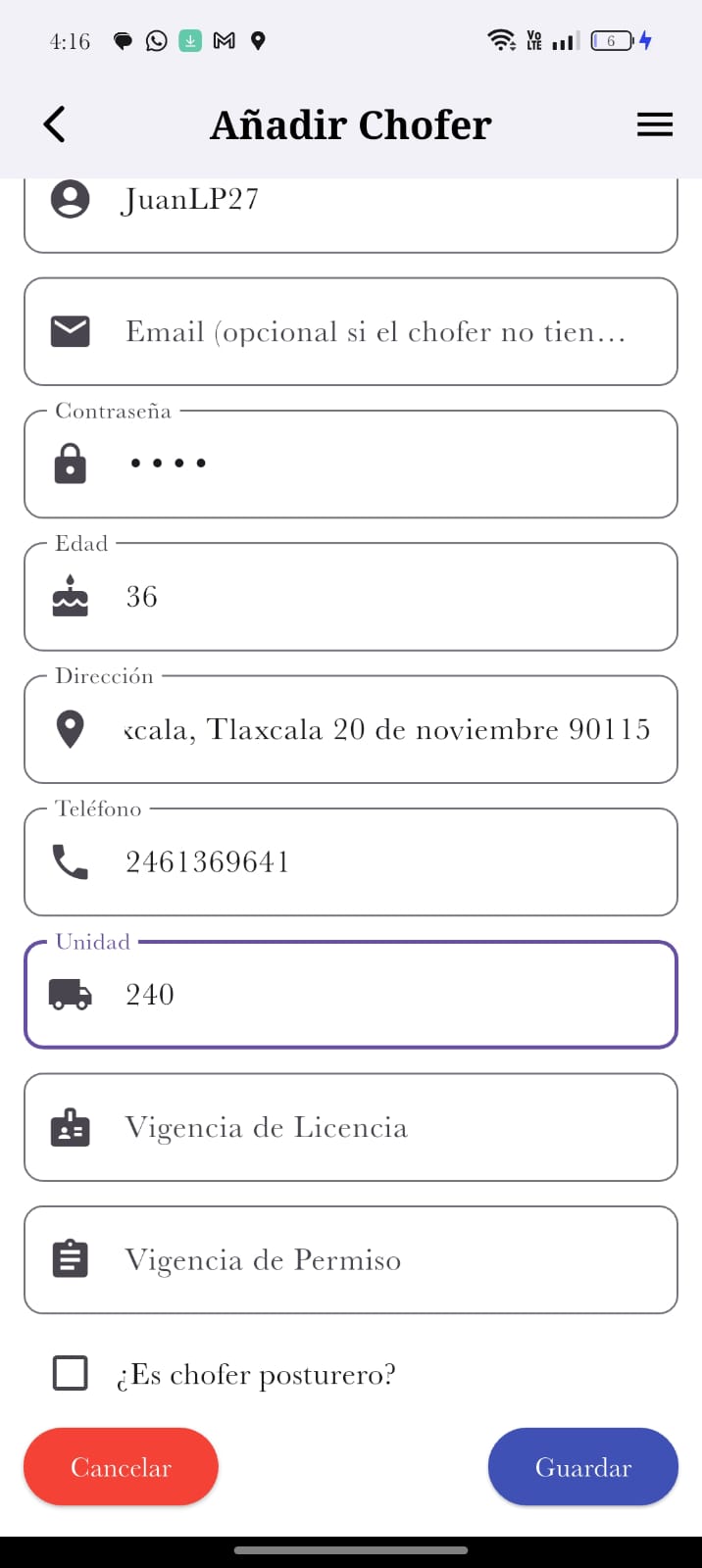


Figura 5.5 [Registro de Choferes] Fuente: Propia

En la figura 5.6 se muestra el calendario para las vigencias de permiso y licencia para establecer una fecha



Figura 5.6 [Fecha de vigencias] Fuente: Propia

En la figura 5.7 se logra apreciar el registro ya con los datos ingresados

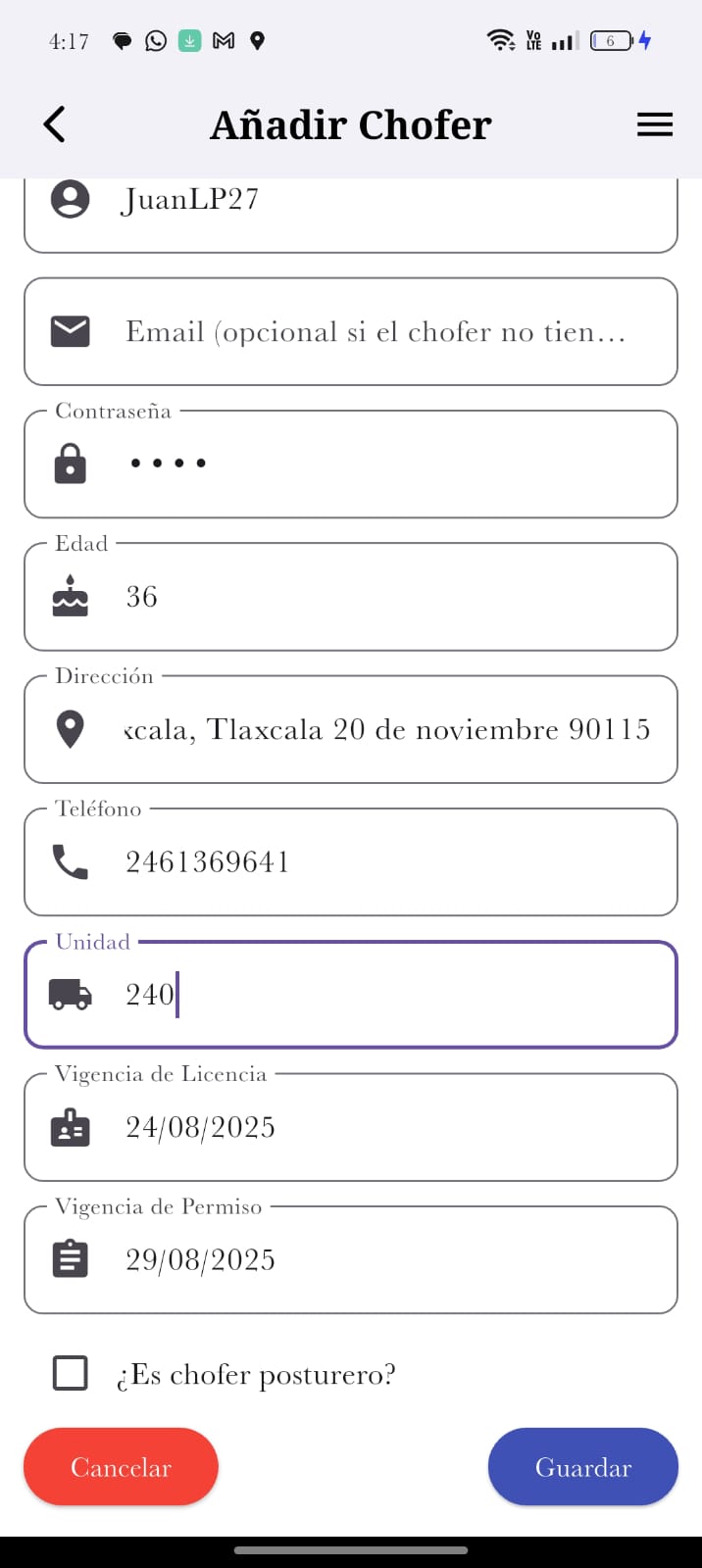


Figura 5.7 [Campos del registro] Fuente: Propia

En la siguiente imagen 5.8 se puede ver la función de búsqueda por filtrado en práctica dentro del apartado de gestión de choferes.

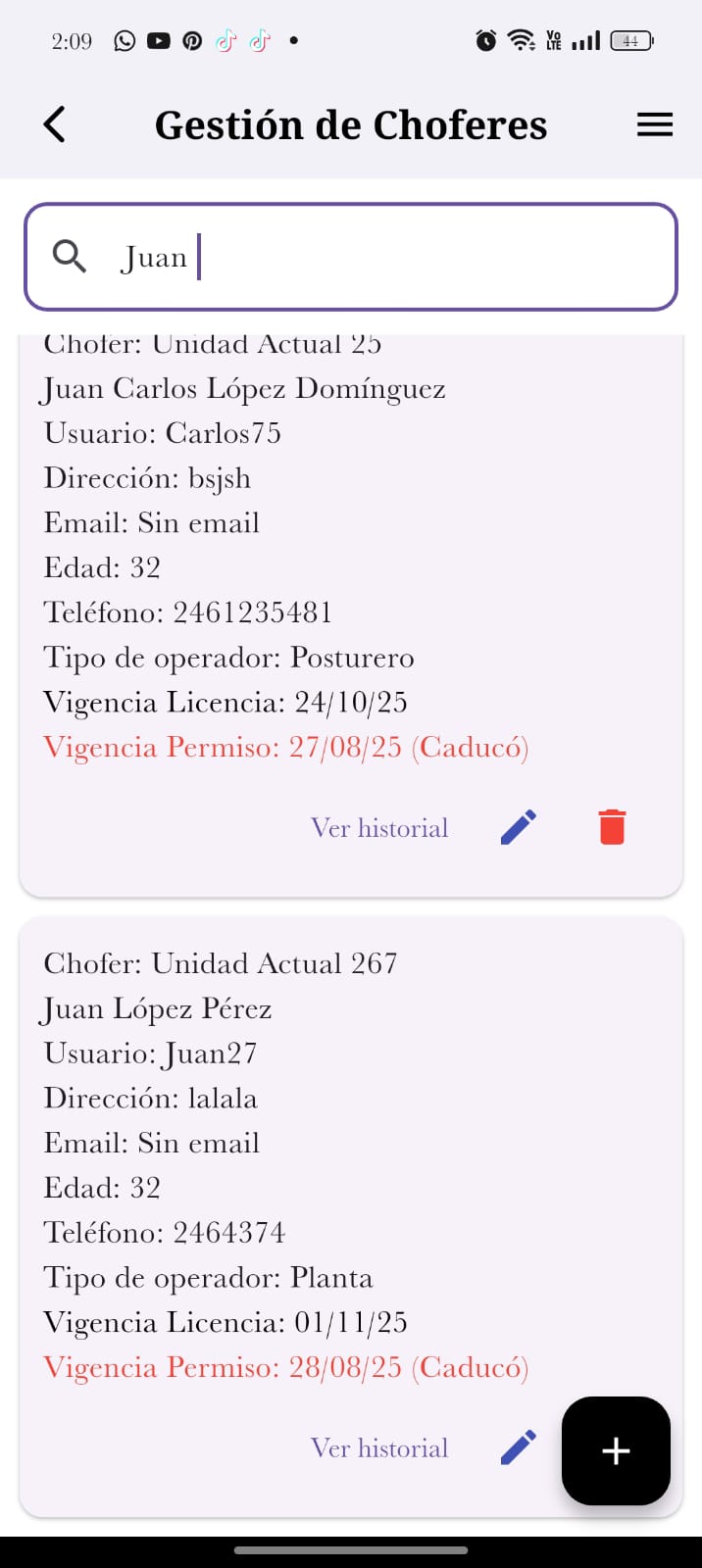


Figura 5.8 [Búsqueda de choferes por lupa] Fuente: Propia

En la siguiente figura 5.9 se puede visualizar la función de historial de choferes, donde al seleccionar un chofer podemos ver todas las unidades que a manejado un chofer y cuando fue la ultima vez que las manejo.

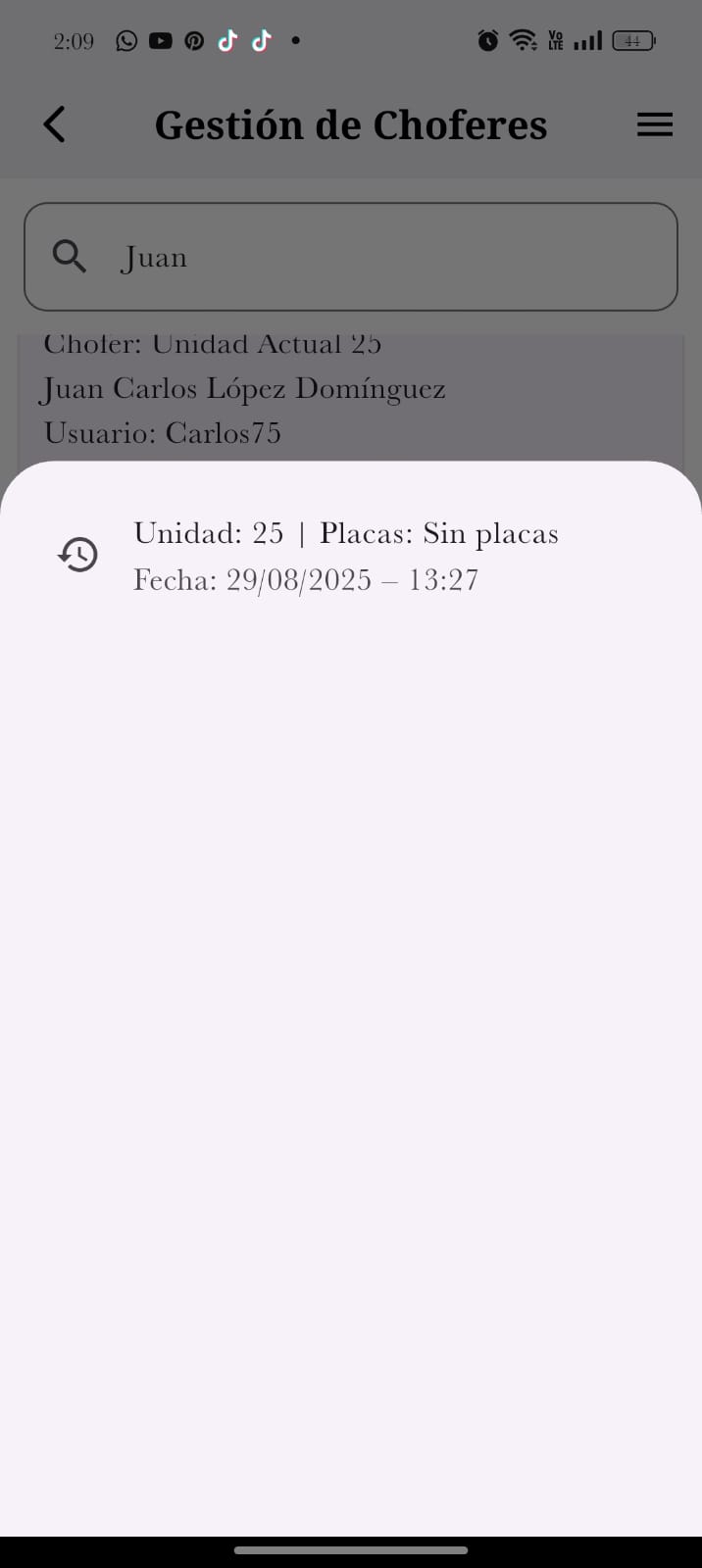


Figura 5.9 [Historial de Choferes] Fuente: Propia

En la siguiente figura 5.10 podemos ver el apartado del módulo de gestión de unidades, donde podemos visualizar las unidades, sus placas y su color.

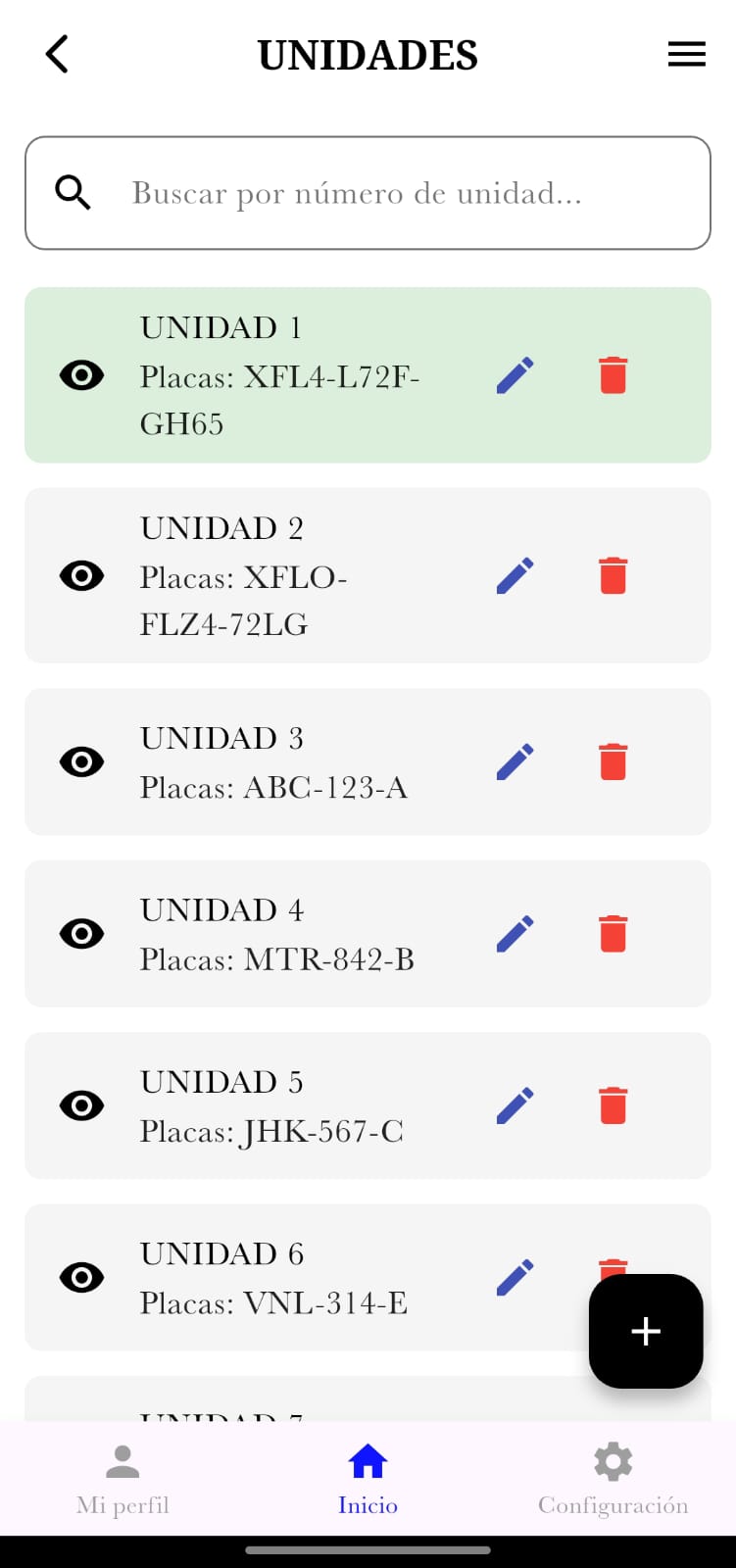


Figura 5.10 [Gestión de Unidades] Fuente: Propia

En la siguiente figura 5.11 podemos ver el módulo de registro de una unidad donde se puede poner el número de la unidad, sus placas y su color según el color de la urban.



Figura 5.11 [Registro de Unidades] Fuente: Propia

En la siguiente figura 5.12 se relaciona con la anterior figura 5.11 donde ya agregamos los datos que vamos a registrar.

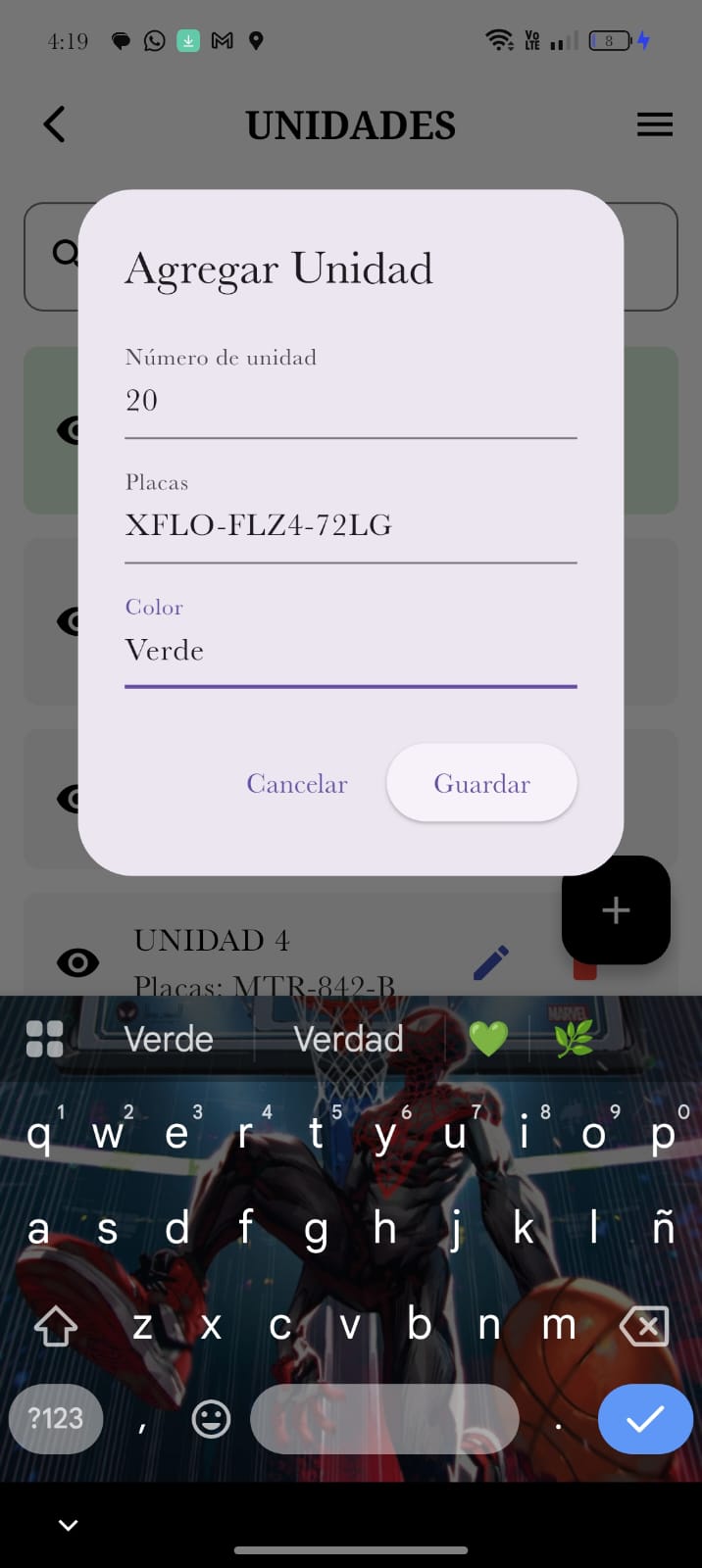
+

Figura 5.12 [Color de las combis] Fuente: Propia

En la siguiente figura 5.13 se relaciona con las figuras anteriores 5.11 y 5.12 donde ya podemos ver la unidad ya registrada.

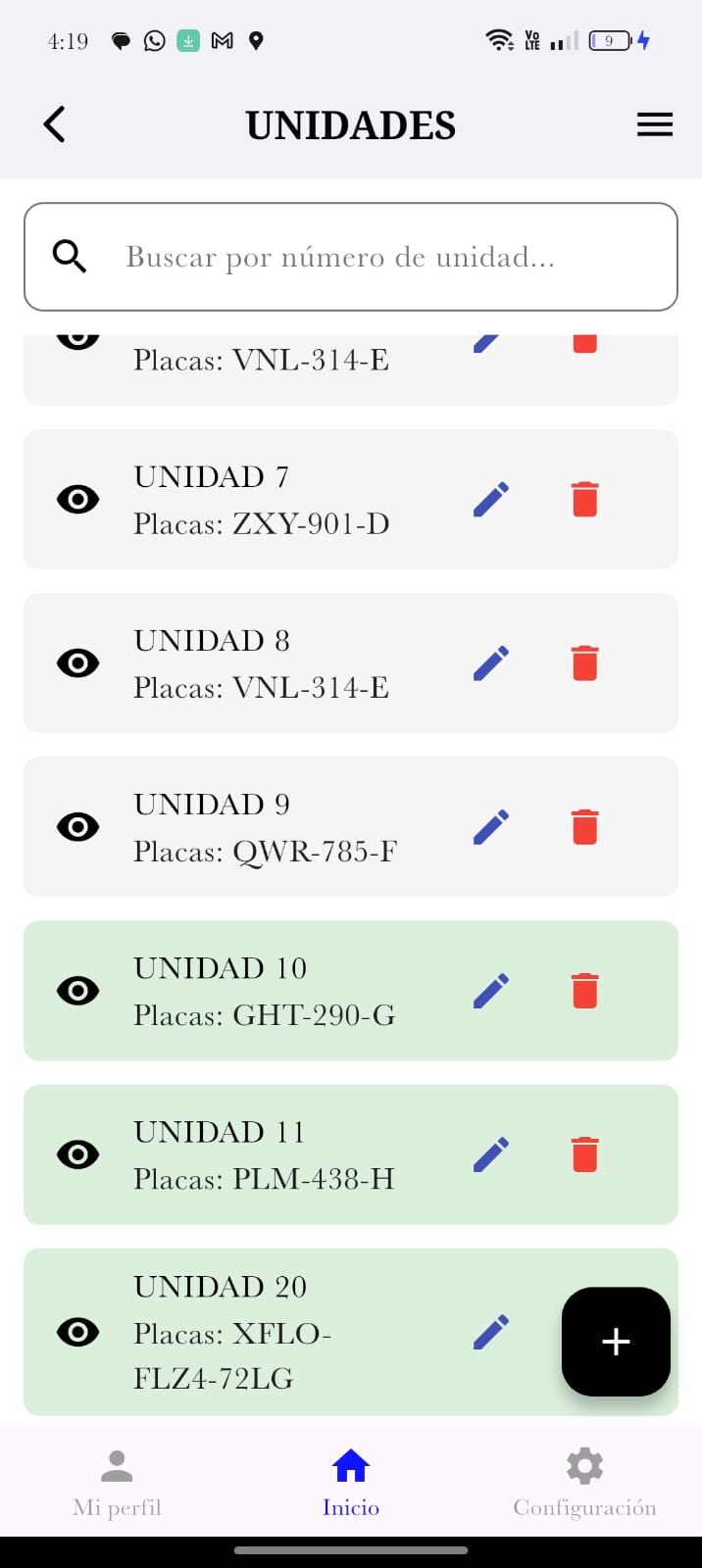


Figura 5.13 [Ejemplo con relación a la Figura 5.12] Fuente: Propia

En la siguiente figura 5.14 al darle tap en alguna unidad nos abre su historial y como se aprecia la unidad ha sido manejada por el chofer con el nombre de usuario Alex25 las placas y la fecha y hora.

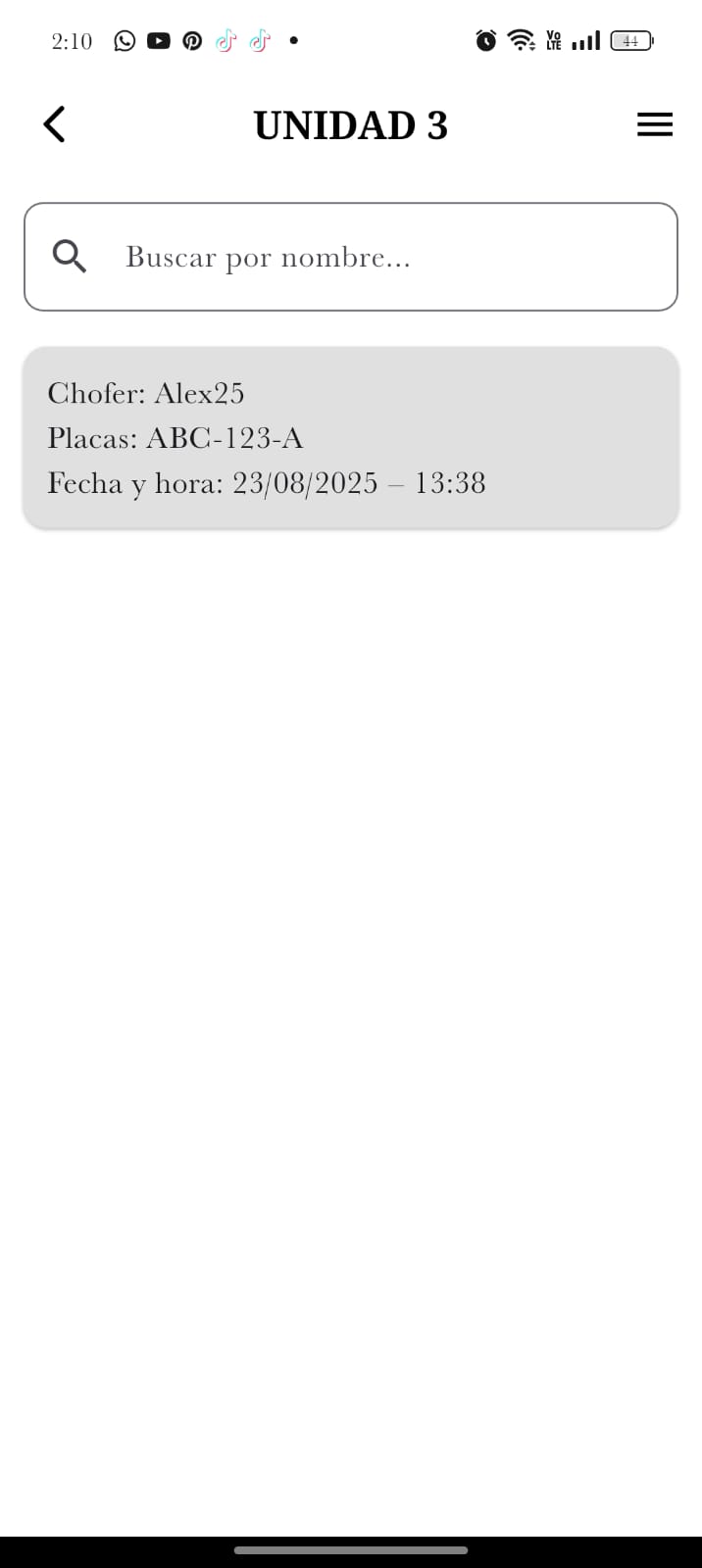


Figura 5.14 [Historial de Unidades] Fuente: Propia

En la siguiente figura 5.15 se muestra el menú principal para un chofer donde ellos no tienen las opciones de gestión de choferes ni de unidades.



Figura 5.15 [Menú principal para choferes] Fuente: Propia

En la siguiente figura 5.16 Se muestra la barra lateral de un chofer donde no hay ninguna opción de control o gestión y muestra opciones básicas para el chofer como lo son: “**Inicio**”, “**Mi perfil**”, “**Mensajes**”, “**Configuración**” y “**Cerrar sesión**”

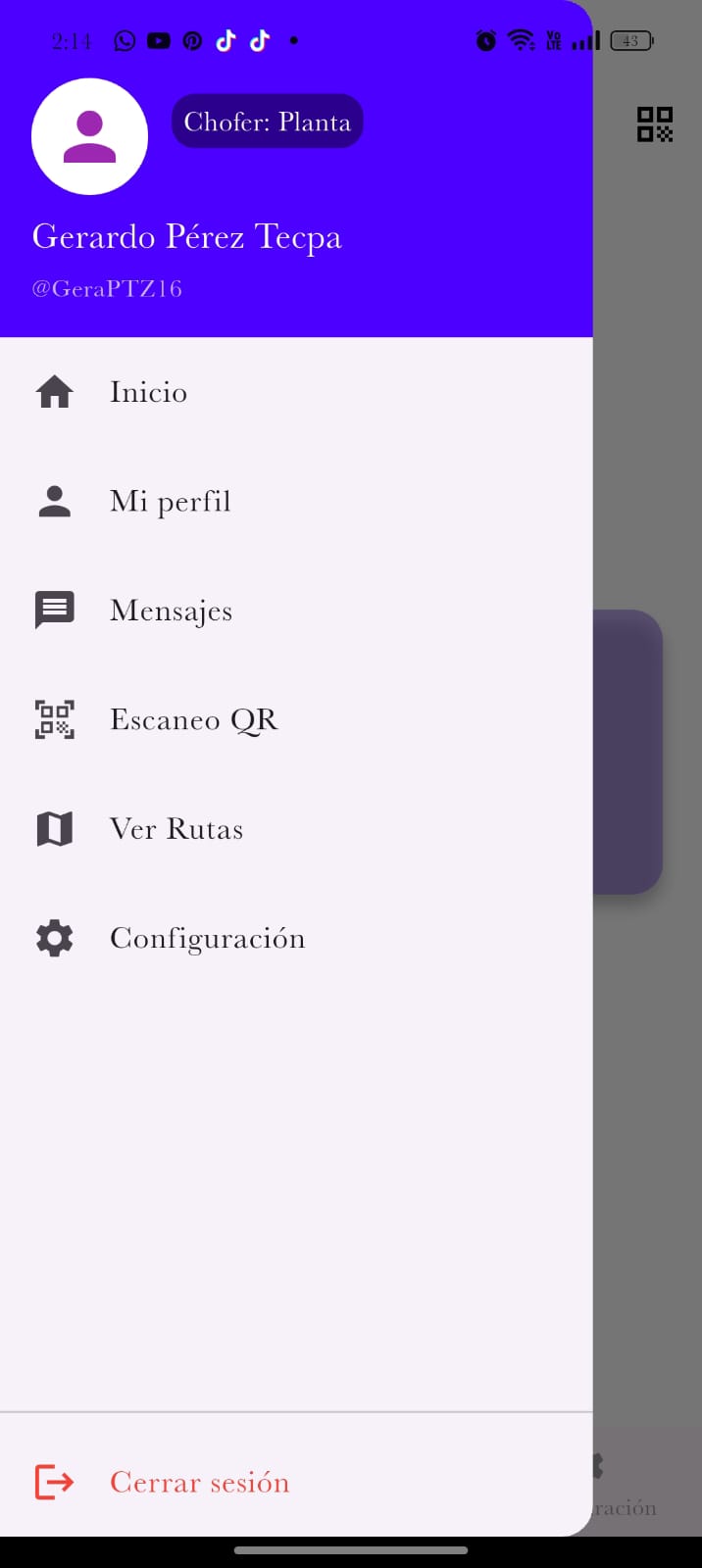


Figura 5.16 [Menú lateral de Choferes] Fuente: Propia

En la siguiente figura 5.17 se logra apreciar el chat que trae la aplicación para la comunicación entre socios de la empresa.

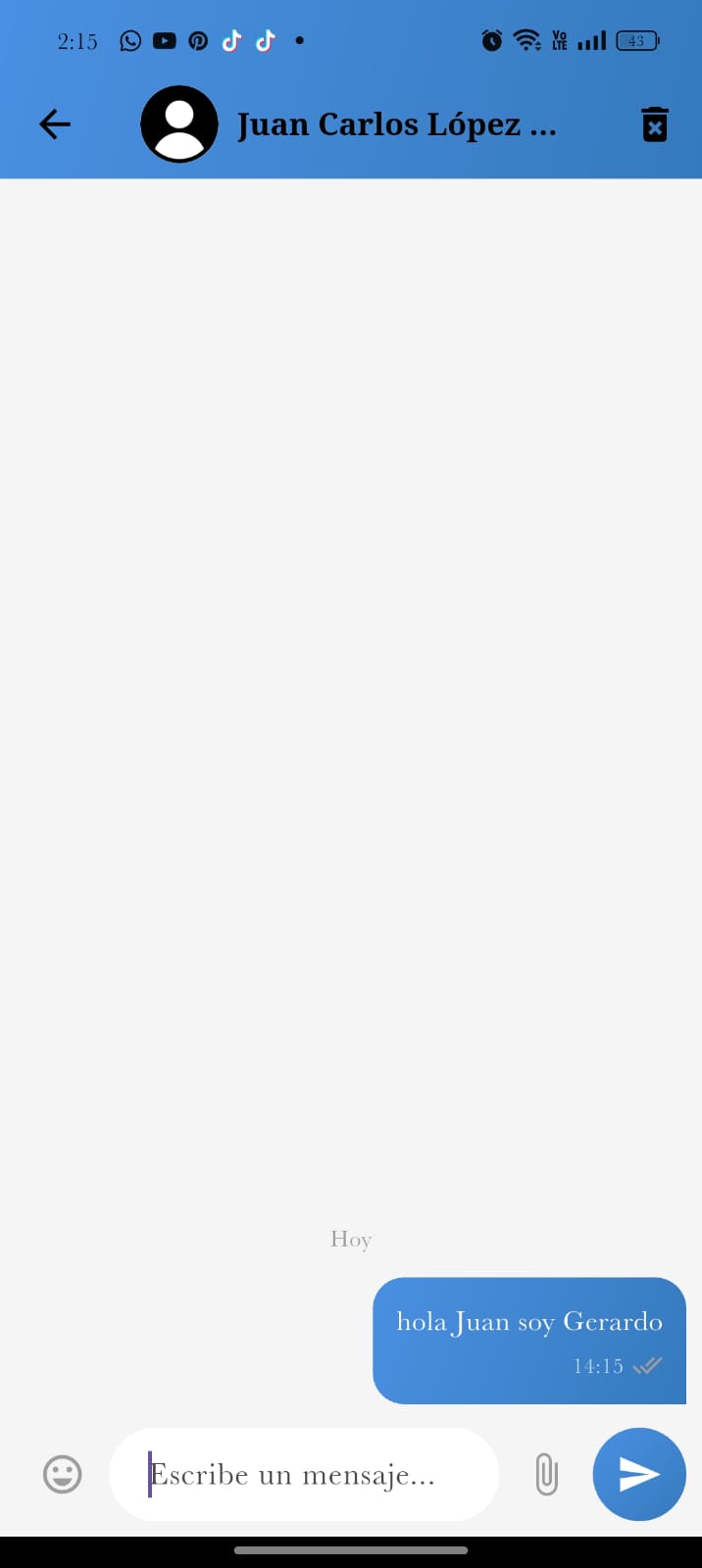


Figura 5.17 [Chat personal, para socios y trabajadores de la empresa] Fuente: Propia

En la siguiente figura 5.18 se muestra la función de entregado al enviar un mensaje para saber cuándo un mensaje ya llego.

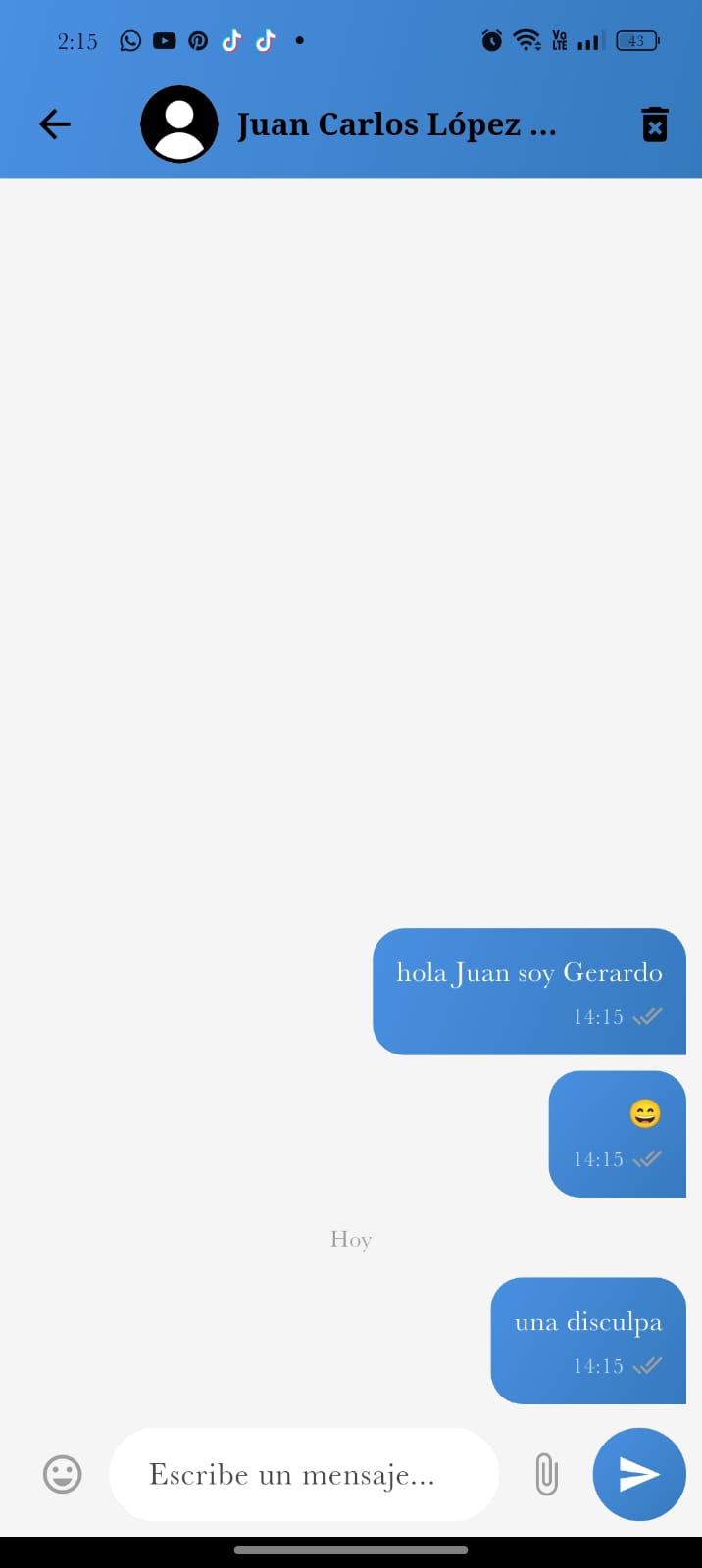


Figura 5.18 [Función de envíado en el chat] Fuente: Propia

En la siguiente figura 5.19 se logra apreciar la función de editar un mensaje en dado caso que se hayan equivocado en algo y se necesite corregir.

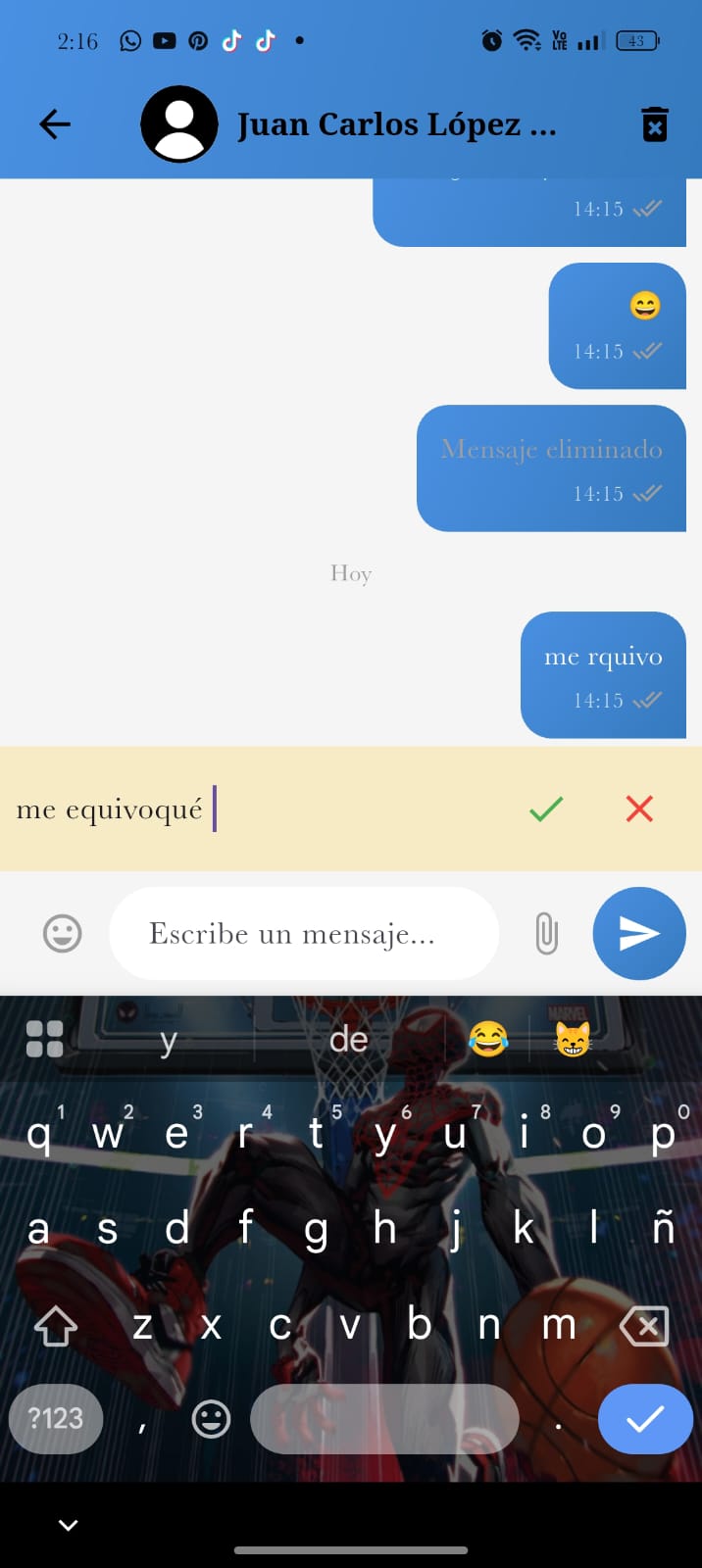


Figura 5.19 [Edición de mensajes] Fuente: Propia

En la siguiente figura 5.20 va relacionada con la figura 5.19 y muestra en texto editado cuando un mensaje fue editado.

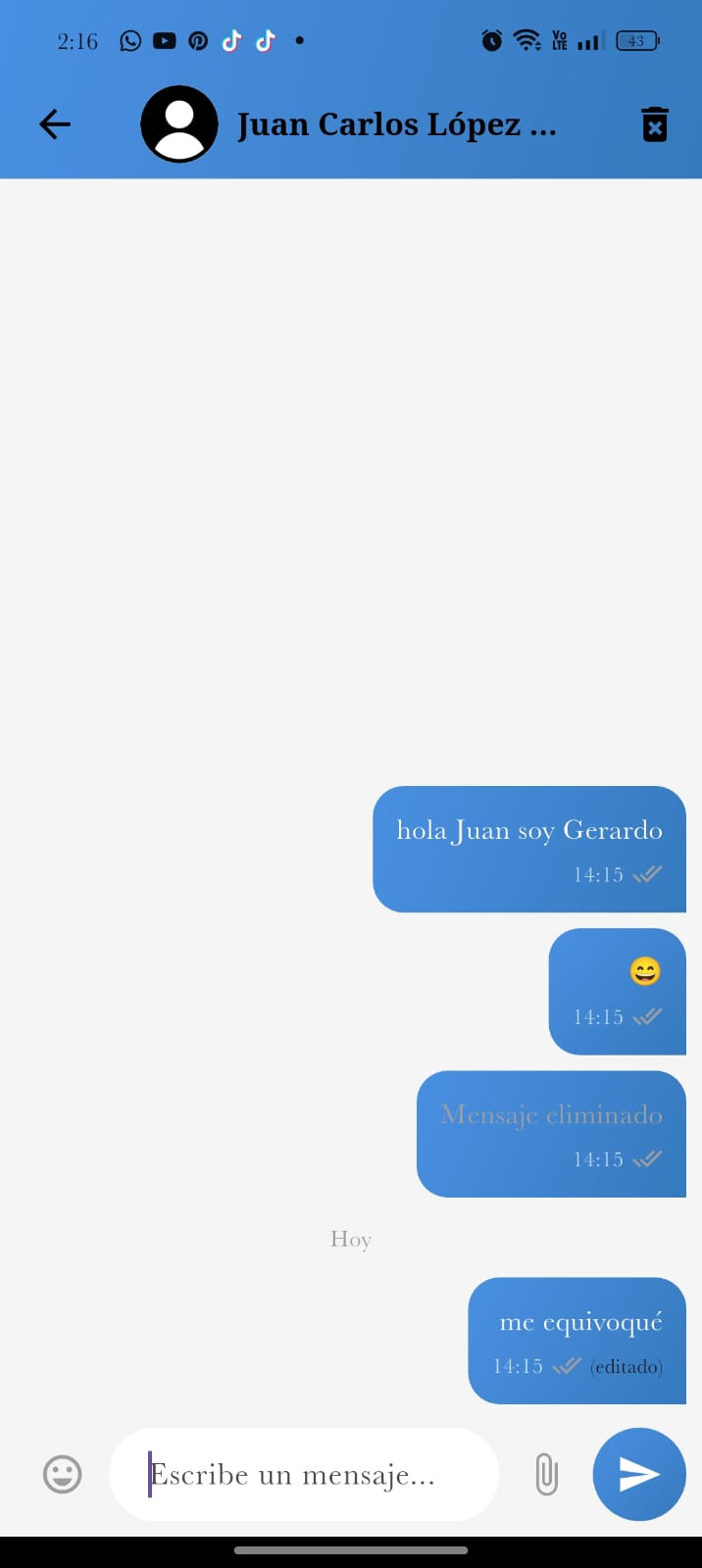


Figura 5.20 [Avisos de mensaje eliminado y mensaje editado] Fuente: Propia

En la siguiente figura 5.21 todos los socios de la empresa salen para que se comuniquen entre sí a través de la aplicación de la empresa incluyendo todos los usuarios registrados en la app, sin importar su rol: despachador, chofer o administrador esto para la comunicación entre ellos.



Figura 5.21 [Lista de todo el personal de la empresa para la comunicación entre trabajadores y administradores] Fuente: Propia

En la siguiente figura 5.22 Se muestra la barra desplegable de un chofer posturero que no tiene correo electrónico y en vez de decir su email dice su nombre de usuario y si tiene correo electrónico, muestra ambos por igual

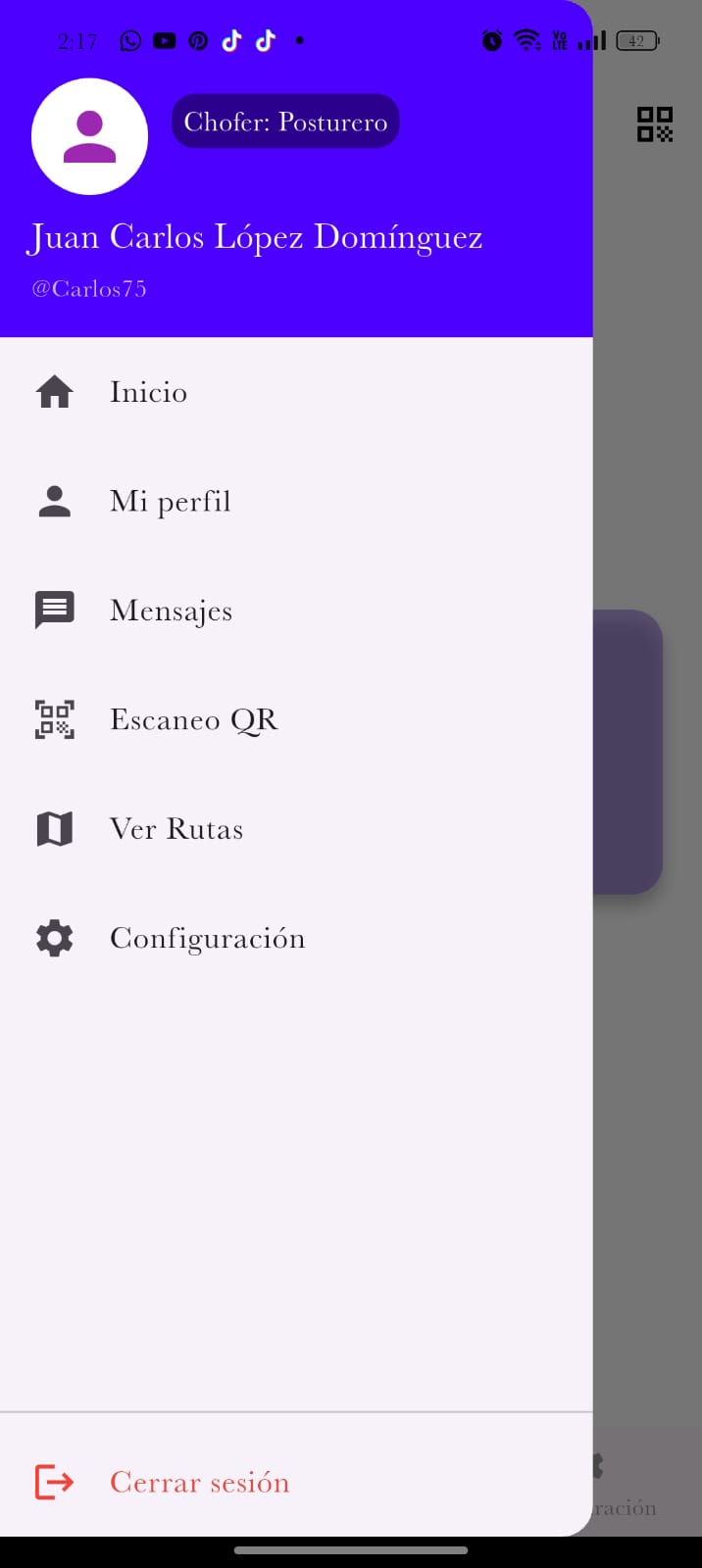


Figura 5.22 [Menú de chofer posturero] Fuente: Propia

En la siguiente figura 5.23 se muestra la función de visto de la empresa, y la comunicación personal entre 2 socios de la empresa.

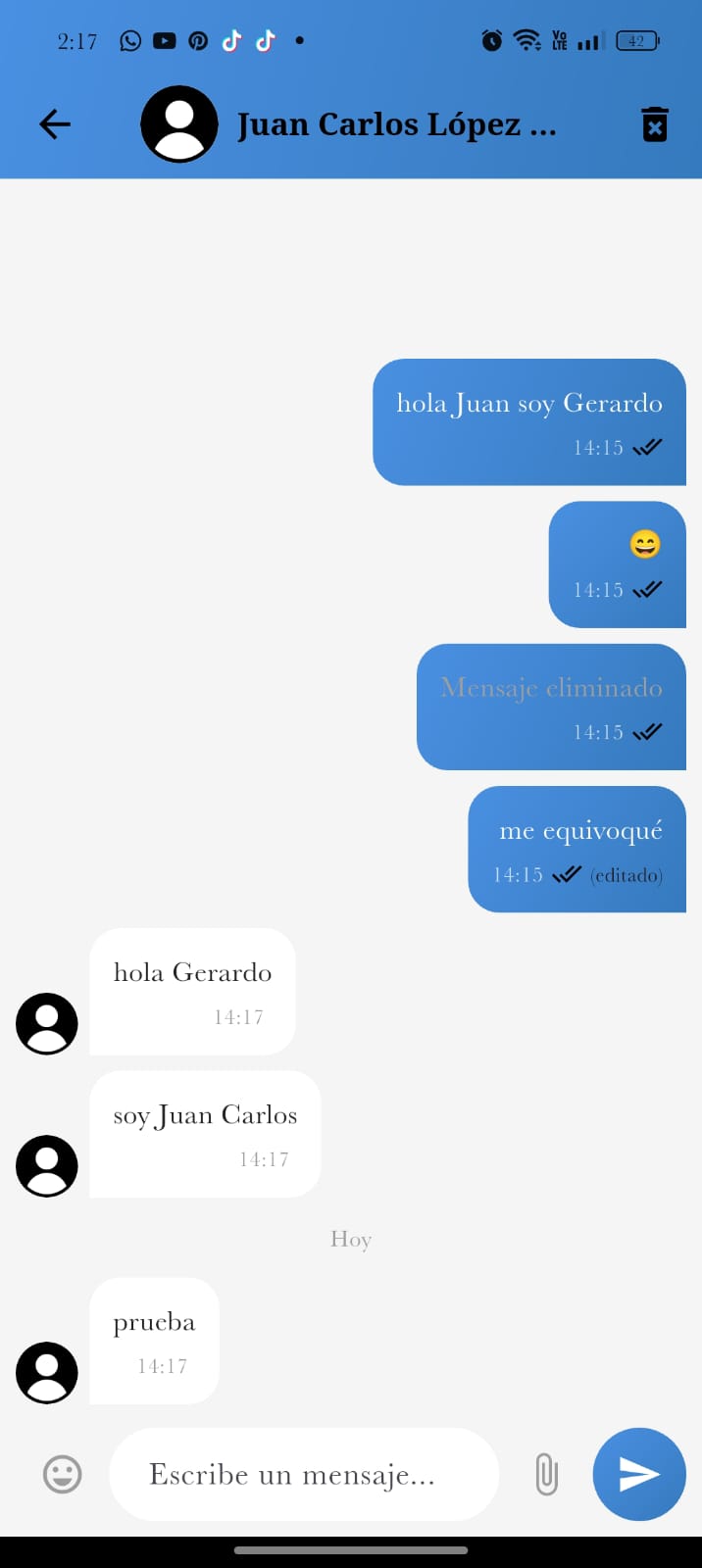


Figura 5.23 [Visto en mensajes] Fuente: Propia

En la siguiente figura 5.24 se muestra el menú en modo oscuro adaptando los colores para entornos de noche y adaptándose a cada usuario



Figura 5.24 [Modo Oscuro] Fuente: Propia

En la siguiente figura 5.25 se muestra el módulo de configuración donde se puede mostrar las opciones que tiene el apartado de configuración como lo son el modo oscuro, el tamaño de la letra y el formato de 12 horas para los mensajes 14:00 horas/ 2:00 p.m. avisos de privacidad o información personal



Figura 5.25 [Configuraciones] Fuente: Propia

En la siguiente figura 5.26 se ve la función del modo oscuro desactivada.



Figura 5.26 [Swicht de modo oscuro y modo claro] Fuente: Propia

En la siguiente figura 5.27 se visualiza el tamaño de la letra en mediano.



Figura 5.27 [Tamaño de letra mediano] Fuente: Propia

En la siguiente figura 5.28 se muestra el tamaño del texto en grande

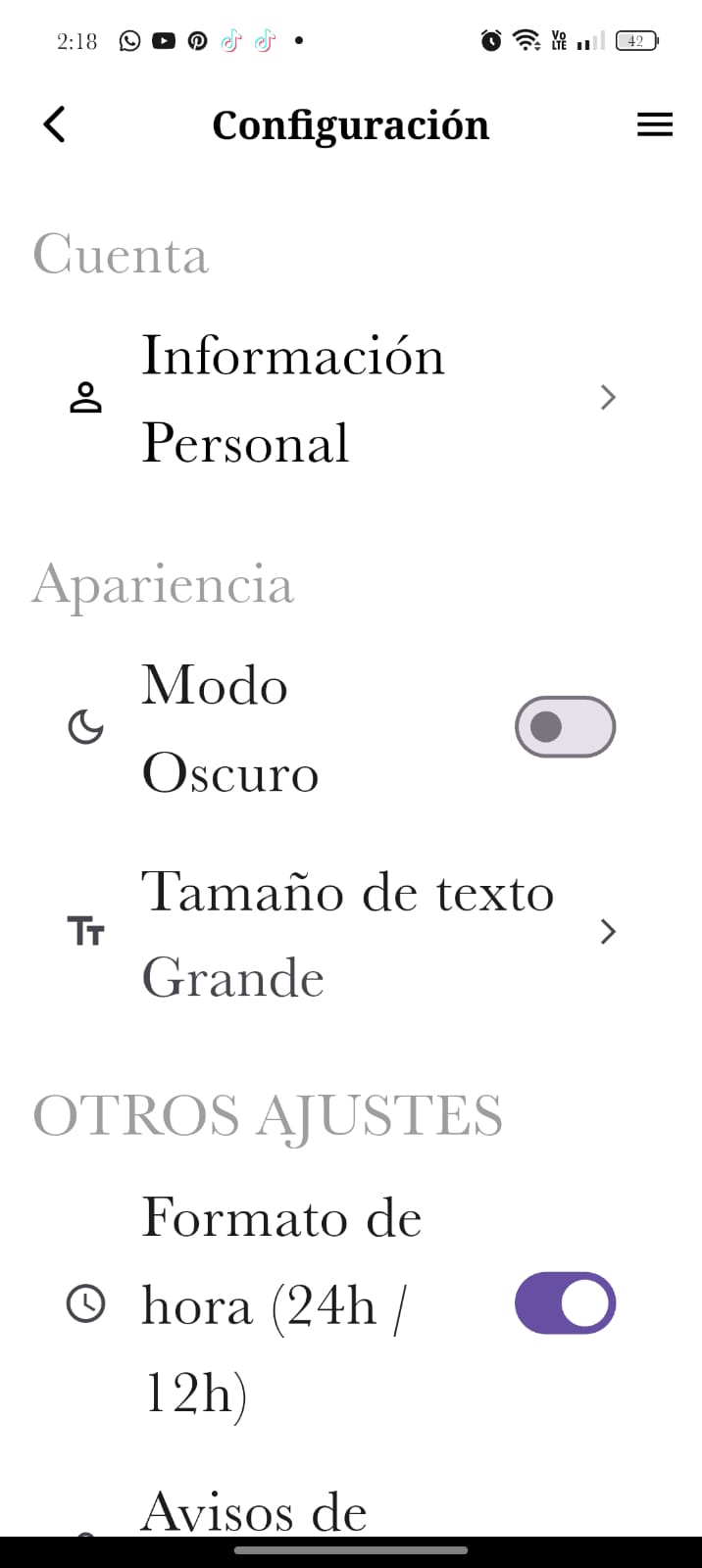


Figura 5.28 [Tamaño de letra Grande] Fuente: Propia

En la siguiente figura 5.29 se muestra el tamaño del texto cambiando entre valores del 1 al 30 donde 1 es muy pequeña y 30 muy grande



Figura 5.29 [Selección del tamaño de letra] Fuente: Propia

**Conclusiones y/o propuestas**

**BIBLOGRAFÍA**

# Referencias

# abamobile. (julio de 2024). *abamobile.com*. Obtenido de https://abamobile.com/web/apps-multiplataforma-que-son-y-caracteristicas/

Amazon. (2025). *aws.amazon.com*. Obtenido de https://aws.amazon.com/es/what-is/api/#:~:text=API%20significa%20%E2%80%9Cinterfaz%20de%20programaci%C3%B3n,de%20servicio%20entre%20dos%20aplicaciones.

Amazon Web Services [AWS]. (17 de Septiembre de 2025). *aws.amazon.com*. Obtenido de https://aws.amazon.com/es/free/database/?trk=57275721-a36d-44bc-9cf0-eb1fa3623abe&sc\_channel=ps&ef\_id=Cj0KCQjwzt\_FBhCEARIsAJGFWVmJpmWHdbs8d8KdGAiN9dkF3MZ526wDtSXCv12yoro35MVV-Pak-waAmPKEALw\_wcB:G:s&s\_kwcid=AL!4422!3!646924994827!e!!g!!bases%20de%20datos%

arimetrics. (2025). *arimetrics.com*. Obtenido de https://www.arimetrics.com/glosario-digital/whatsapp#:~:text=WhatsApp%20es%20una%20aplicaci%C3%B3n%20de,contrase%C3%B1a%20para%20acceder%20al%20servicio.&text=3%20%C2%BFWhatspApp%20es%20gratis?

*asana.com*. (06 de Febrero de 2025). Obtenido de https://asana.com/es/resources/waterfall-project-management-methodology

*atlassian.com*. (s.f.). Obtenido de https://www.atlassian.com/es/agile/scrum#:~:text=Scrum%20es%20un%20marco%20de,estructurar%20y%20gestionar%20su%20trabajo.

browserstack. (4 de 3 de 2025). *.browserstack.com*. Obtenido de https://www.browserstack.com/guide/what-is-flutter

datacamp. (29 de julio de 2024). *datacamp.com*. Obtenido de https://www.datacamp.com/es/blog/types-of-databases-overview

demandsage. (3 de julio de 2025). *demandsage.com*. Obtenido de https://www.demandsage.com/android-statistics/#:~:text=Conclusi%C3%B3n:%20Android%20posee%20el%2071,m%C3%A1s%20utilizado%20a%20nivel%20mundial.

developers.google.com. (09 de mayo de 2024). *developers.google.com*. Obtenido de https://developers.google.com/maps/documentation/android-sdk/overview?hl=es-419#:~:text=Con%20el%20SDK%20de%20Maps%20para%20Android%2C,del%20mapa%20y%20facilitar%20la%20interacci%C3%B3n%20con

ebac. (1 de junio de 2023). *ebac.mx*. Obtenido de https://ebac.mx/blog/que-es-adobe-photoshop

equiposalfa. (2025). *equiposalfa.mx*. Obtenido de https://equiposalfa.mx/historia-del-gps/#:~:text=El%20desarrollo%20del%20GPS%20comenz%C3%B3%20en%20la,conocido%20como%20Navstar%2C%20fue%20lanzado%20en%201978.

firbase. (09 de septiembre de 2025). *firebase.google.com/*. Obtenido de https://firebase.google.com/docs/auth?hl=es-419

fortinet. (2025). *fortinet.com*. Obtenido de https://www.fortinet.com/lat/resources/cyberglossary/mobile-app-security#:~:text=Significado%20de%20seguridad%20de%20aplicaciones,exploit%20afecte%20a%20su%20entorno.

FutureSpace. (s.f.). *futurespace.es*. Obtenido de https://www.futurespace.es/tipos-de-aplicaciones-moviles/#:~:text=Aplicaci%C3%B3n%20nativa,m%C3%A1ximo%20las%20caracter%C3%ADsticas%20del%20dispositivo.

gcfglobal. (f.f). *edu.gcfglobal.org*. Obtenido de https://edu.gcfglobal.org/es/google-drive/que-es-google-drive-/1/

GeeksforGeeks. (16 de Enero de 2024). *What are Software Development Methodologies | 15 Key Methodologies*. Obtenido de https://www.geeksforgeeks.org/: https://www.geeksforgeeks.org/software-engineering/what-is-software-development-methodology-15-key-methodologies/?utm\_source=chatgpt.com

*ginzo.tech*. (28 de diciembre de 2021). Obtenido de https://ginzo.tech/metodologia-clasica-desarrollo-software/

Google. (4 de Agosto de 2025). *firebase.google.com*. Obtenido de https://firebase.google.com/docs/database?hl=es-419

Guzman, R. (12https://www.facebook.com/watch/?v=1249703429846309 de abril de 2025). *facebook*.

Henao, C. (21 de Junio de 2018). *https://www.youtube.com/@CristianDavidHenao*. Obtenido de https://www.youtube.com/watch?v=HhC75IonpOU

ibm. (20 de octubre de 2021). *ibm.com*. Obtenido de https://www.ibm.com/mx-es/think/topics/relational-databases

ibm. (30 de septiembre de 2024). *www.ibm.com*. Obtenido de https://www.ibm.com/mx-es/think/topics/database

ibm. (25 de abril de 2025). Obtenido de https://www.ibm.com/mx-es/think/topics/chatgpt

ibm. (19 de abril de 2025). *ibm.com*. Obtenido de https://www.ibm.com/mx-es/think/topics/chatgpt

learn.microsoft.com. (2025). *learn.microsoft.com*. Obtenido de https://learn.microsoft.com/es-es/azure/architecture/data-guide/big-data/non-relational-data

lenovo. (2025). *lenovo.com*. Obtenido de https://www.lenovo.com/mx/es/glosario/microsoft-word/?orgRef=https%253A%252F%252Fwww.google.com%252F&srsltid=AfmBOorQxsrRU7R2tVglttQOLRSXDB4iCEAa4lBBdX159wpyvxMM2lDa

manualweb. (febrero de 2024). Obtenido de manualweb: https://www.manualweb.net/dart/historia-de-dart/#:~:text=Histor%C3%ADa%20de%20Dart,creaci%C3%B3n%20de%20aplicaciones%20m%C3%B3viles%20multiplataforma

Microsoft. (09 de septiembre de 2024). *learn.microsoft.com*. Obtenido de https://learn.microsoft.com/es-es/windows/dev-environment/javascript/react-native-for-windows

Microsoft. (2025). *learn.microsoft.com*. Obtenido de https://learn.microsoft.com/es-es/azure/architecture/data-guide/big-data/non-relational-data

Munguía, M. Á. (2025). *Aplicación móvil para gestión de linea de transporte y carga.* Hidalgo: Universidad Autónoma Del Estado De Hidalgo.

profile.es. (2025). *profile.es*. Obtenido de https://profile.es/blog/que-es-ionic/#:~:text=M%C3%A1s%20productividad%20y%20menos%20costes,desarrollo%20de%20una%20aplicaci%C3%B3n%20nativa.

Ramírez, E. Y. (2020). *Desarollo de un aplicativo móvil para el registro del cumplimiento y desempeño de líneas de transporte público en la ciudad de lima.* Lima: Escuela Académico Profesional de Ingeniería de Sistemas e Informática.

reanimasoluciones. (s.f). *reanimasoluciones.com*. Obtenido de https://www.reanimasoluciones.com/actualidad/329020-desarrollo-de-aplicaciones-moviles-con-geolocalizacion-para-negocios#:~:text=La%20geolocalizaci%C3%B3n%20utiliza%20el%20GPS,mapas%20interactivos%20y%20listas%20personalizadas.

Reyes., D. A. (2025). *Historia de la empresa Urbanos y Sub Urbanos de Tlaxcala S.A. de C.V. [Manuscrito no publicado].* Tlaxcala.

*santanderopenacademy.com*. (21 de diciembre de 2020). Obtenido de https://www.santanderopenacademy.com/es/blog/metodologias-desarrollo-software.html

scielo.org. (junio de 2010). *scielo.org.mx*. Obtenido de https://www.scielo.org.mx/scielo.php?script=sci\_arttext&pid=S1665-80272010000100009#:~:text=El%20tren%20recorri%C3%B3%20simplemente%20el,en%20medio%20de%20ciudades%20importantes%22.

statista. (3 de 6 de 2024). *es.statista.com*. Obtenido de https://es.statista.com/grafico/29620/sistema-operativo-movil-con-la-mayor-cuota-de-mercado-por-pais/

tsomobile. (2025). *tsomobile.com.pe*. Obtenido de https://tsomobile.com.pe/blog/que-es-un-gps-y-como-funciona/#:~:text=%C2%BFC%C3%B3mo%20funciona%20un%20GPS?,de%20latitud%2C%20longitud%20y%20altitud.

*udemy.com*. (mayo de 2020). Obtenido de udemy.com/course/metodologias-agiles-enfoque-pmi/?utm\_source=adwords&utm\_medium=udemyads&utm\_campaign=Search\_DSA\_Alpha\_Prof\_la.ES\_cc.MX&campaigntype=Search&portfolio=Mexico&language=ES&product=Course&test=&audience=DSA&topic=Project\_Management&priority=Al