

ANEXO B. DESARROLLO DE UN PROTOTIPO DE UNA APLICACIÓN WEB DE CARPOOLING

En este anexo, se abordan los aspectos fundamentales relacionados con el desarrollo de la aplicación y la implementación del módulo de autenticación basado en tecnología *Blockchain*. Este anexo se compone de una serie de puntos clave como el estudio de requisitos, el diseño de vistas y diseño de la interfaz de usuario (UI), el diseño de la experiencia de usuario (UX), el desarrollo del prototipo de la aplicación de carpooling, la integración de módulo de autenticación y aplicación web y las pruebas de funcionamiento.

Contenido

1. Estudio de requisitos	2
2. Historias de usuario de la aplicación.	11
3. Diseño de vistas	29
5. Desarrollo de la aplicación	50
6. Pruebas de funcionamiento	72

1. Estudio de requisitos

El estudio de requisitos es una etapa fundamental para desarrollar una aplicación web, ya que permite identificar y documentar de manera precisa las necesidades y las expectativas de los usuarios al usar la aplicación. Esto es fundamental para que el producto final de la aplicación cumpla con las expectativas y necesidades de los usuarios finales [1].

La ingeniería de requisitos utiliza dos medios de comunicación para recoger los requisitos, los cuales son la comunicación verbal y la comunicación escrita, cada uno de ellos tiene diferentes ventajas. La comunicación verbal se adapta con facilidad a nuevos desarrollos y permite recibir un *feedback* de forma inmediata, mientras que, la comunicación escrita registra la información de forma permanente y se puede compartir de forma más sencilla con grupos, lo que permite su completa revisión [3].

Se pueden utilizar diferentes metodologías para realizar el estudio de requisitos, las más conocidas son:

- **Entrevistas con los usuarios.**

La entrevista es un proceso que consiste en un intento sistemático de recolectar información de otra persona a través de una comunicación interpersonal estructurada en forma de conversación.

En una entrevista, el entrevistador debe seguir una estrategia para obtener información relevante y útil. Para ello, puede hacer uso de preguntas abiertas y cerradas, así como de técnicas de escucha activa.

Es de gran utilidad para obtener información cualitativa de las actividades. Es bastante utilizada para recopilar información de primera mano sobre las necesidades y expectativas de los usuarios [2].

- **Desarrollo conjunto de aplicaciones.**

Más conocido como JAD por sus siglas en inglés, *Joint Application Design*. Es una metodología que se centra en la colaboración entre el equipo de desarrollo y los usuarios finales. En un proceso típico de *JAD* los desarrolladores y los usuarios se reúnen en una serie de sesiones de trabajo estructuradas. En estas sesiones, se discuten las necesidades y requisitos de la aplicación y se establecen los objetivos del proyecto y de acuerdo a ello, el equipo de desarrollo crea una especificación funcional detallada para la aplicación. Al involucrar a los usuarios desde el principio, se puede crear aplicaciones que satisfagan sus necesidades de forma efectiva y eficiente, acelerando el proceso de desarrollo, y reduciendo la necesidad de revisiones y pruebas adicionales [2].

- **Desarrollo de prototipos**

Consiste en versiones reducidas de la aplicación que no son necesariamente totalmente funcionales. Permiten al usuario experimentar con dicho prototipo para ver sus posibles funcionalidades. Generalmente, el uso de prototipos desemboca en nuevas ideas que se convierten en requerimientos de la aplicación. El uso de prototipos incrementa el costo de las etapas iniciales de un proyecto [1].

- **Introspección**

Consiste en que, el analista se pone en el lugar del cliente, intentando dar su punto de vista (como cliente) de como quisiera que fuera la aplicación. El problema radica en que un analista no es el usuario final, por lo que algunas funcionalidades deseadas por el usuario final no estarán incluidas [1].

- **Lluvia de ideas**

Se basa en reuniones con un grupo de personas, en donde proponen ideas sin juzgar su validez, y luego de compilar dichas ideas, se realiza un análisis detallado de cada una de ellas [2].

- **Historias de usuario.**

Son utilizadas en el desarrollo ágil de software para identificar y documentar los requisitos de una aplicación. Se enfoca en describir las necesidades y objetivos de

los usuarios de una forma clara y fácil de entender, además, combina las ventajas de los dos tipos de comunicaciones (verbal y escrita) mencionadas anteriormente [3].

La metodología en cascada fue la seleccionada para desarrollar el proyecto, desarrollando los paquetes de trabajo definidos, pero algunos paquetes de trabajo involucran el desarrollo de prototipos o aplicaciones en particular (por ejemplo, el paquete de trabajo 3, WP3) para los cuales se ha determinado utilizar el marco de trabajo (o *framework*) de *Scrum*.

Scrum es un marco de trabajo dentro del cual se pueden emplear varias técnicas o procesos para gestionar el desarrollo de productos complejos. *Scrum* se basa en la teoría de control de procesos empírica, que asegura que el conocimiento procede de la experiencia y de tomar decisiones basándose en lo que se conoce. Además, *Scrum* utiliza tres pilares fundamentales: transparencia, inspección y adaptación [4]. En cuanto a los procesos de la gestión de un proyecto, *Scrum* propone cinco fases: inicio, planeación y estimación, implementación, revisión y retrospectiva, y lanzamiento. En la fase de planeación y estimación, *Scrum* propone el uso de Historias de usuario (HU).

Se determinó conveniente seleccionar las HU para el estudio de requisitos de la aplicación web del proyecto, considerando que el *framework* *Scrum* las utiliza y adicionalmente, debido a las siguientes razones:

- Agiliza la administración de requisitos, reduciendo la cantidad de documentos formales necesarios, lo que se traduce en la disminución de tiempos.
- Están centradas en el usuario, se tiene mayor comprensión de lo que el usuario realmente necesita, en lugar de centrarse en una lista de requisitos técnicos abstractos.
- Son fáciles de comprender, ya que se presentan en un lenguaje natural, lo que facilita su comprensión por parte de todos los interesados en el proyecto.
- Son iterativas y evolutivas, ya que se pueden modificar y ajustar a medida que se avanza en el desarrollo del proyecto, lo que permite una mayor flexibilidad y adaptación a los cambios.

- Las historias de usuario se pueden priorizar en función de su importancia para el usuario y su valor para el negocio, lo que permite una mejor gestión de los recursos y una mayor eficiencia en el desarrollo del proyecto.

La aplicación a desarrollar fue denominada “*GoTogether*”. Es una aplicación web que presta un servicio de carpooling implementando un registro y una autenticación de usuario segura. Los usuarios conductores pueden compartir la ruta por la cual se movilizarán y los usuarios pasajeros pueden seleccionar la ruta que más les convenga.

1.1. Análisis de funcionalidades de una aplicación de carpooling

Se realizó una revisión de la literatura, encontrando artículos en donde se realizaron aplicaciones de *carpooling* para analizar las funcionalidades presentadas en ellos.

En el artículo *An Android based application: Cab pooling [5]*, se realiza una aplicación de *carpooling* que cuenta con un usuario y un administrador. Entre las funcionalidades del usuario está permitir gestionar el perfil, en donde el usuario puede modificar sus datos y recuperar su contraseña, y gestionar reservas en donde el usuario selecciona los taxis de acuerdo a sus requerimientos. Para el usuario administrador, su funcionalidad principal es gestionar taxis, en donde el administrador actualiza la información del vehículo.

Cuando el usuario encuentra una ruta conveniente, puede enviar un mensaje con la información correspondiente del viaje, el cual se enviará a la policía y a los contactos elegidos por el usuario si se detecta que este está en peligro [5].

El siguiente artículo encontrado fue *PoliUniPool: a carpooling system for universities [6]*, en el cual se propone un sistema de transporte compartido entre las universidades *Università Statale* y *Politecnico di Milano*, el cual es un sistema restringido al uso de estudiantes, administrativos y docentes dentro del campus universitario o a las estaciones de transporte como tren o metro. Permite a los usuarios pasajeros proporcionar el horario esperado de sus viajes y ver la información de los mismos, ya sean cambios o retrasos en el servicio. Dentro de sus funcionalidades también se encuentran estimar un costo por persona, comunicación entre conductores y pasajeros, y también incluir las personas con las que se desea

o no viajar. Para los conductores, permite postular su ruta, indicando origen, destino y días en los que se desea compartir el servicio, así como la disponibilidad y capacidad del automóvil [6].

El artículo *Car Pool'up – Real-time Carpooling using GPS* [7], los autores proponen un sistema *carpooling* en donde los usuarios pueden reservar su viaje con una persona que viaja en su misma ruta. Entre sus funcionalidades se encuentran las siguientes: los usuarios pueden registrarse por medio del sitio web de la aplicación, obteniendo información del automóvil más cercano a su ubicación; tiene servicio de *schedule drive* en donde el usuario puede reservar su viaje.

Luego de realizar la reserva, puede verificar el horario del automóvil reservado y rastrear su ubicación; el usuario puede publicar su oferta económica para que los usuarios interesados puedan verla; permite al usuario consultar las tarifas a un destino en particular; y es posible consultar los perfiles de las personas con las que comparte el vehículo por motivos de seguridad.

El artículo *A Permissioned Blockchain-Based Identity Management and User Authentication Scheme for E-Health Systems* [8], propone un sistema de autenticación y gestión de identidad para sistemas de salud electrónicos (*e-health*) basado en *Blockchain* que trata de solventar los problemas de seguridad y privacidad que introducen los sistemas de información en línea. El artículo propone un sistema de autenticación a través de datos biométricos, los cuales se almacenan en la *Blockchain*.

El artículo *FAME: Face Authentication for Mobile Encounter* [9], describe una aplicación desarrollada denominada FAME para dispositivos móviles, la cual proporciona identificación y verificación. La aplicación incluye la gestión de identidad para respaldar actividades sociales, por ejemplo, encontrar personas similares en una red social. Estas funcionalidades se implementan teniendo en cuenta la privacidad y la seguridad.

La implementación de autenticación biométrica ofrece una opción de seguridad necesaria para la aceptación y adopción de una aplicación por parte de los usuarios, tal como se presenta en el artículo *A Permissioned Blockchain-Based Identity*

Management and User Authentication Scheme for E-Health Systems. Por otra parte, la facilidad de uso y comodidad que brinda la autenticación facial, en comparación con otros métodos, la hace una opción interesante a considerar para el proyecto de investigación. En el artículo "*FAME: Face Authentication for Mobile Encounters*" se presentan varios aspectos interesantes que justifican el uso de una autenticación de tipo facial. Este tipo de autenticación, al aprovechar la tecnología biométrica, ofrece una alta precisión y robustez en la verificación de identidad. Debido a lo anterior, la implementación de la autenticación facial resulta ser la opción más adecuada, para garantizar la autenticación privada y segura de los usuarios.

Aunque los tres primeros artículos mencionados anteriormente ([5-7]) se enfocan de manera distinta en implementaciones relacionadas al transporte compartido, tienen algunas funcionalidades en común, tales como la programación de viajes que es indispensable en este tipo de servicios. El primer artículo ([5]) se centra en la seguridad de los usuarios pasajeros al momento de usar la aplicación, ya que su aporte principal es el envío de mensajes de alerta a contactos y policía en caso de que el usuario se encuentre en peligro. En este trabajo, los pasajeros son los que tienen el control de decidir sobre el vehículo y las características de este de acuerdo a sus requerimientos.

El segundo artículo ([6]) es más específico en cuanto a los usuarios a los que va dirigido, ya que se centra en dos universidades específicamente. Al igual que en el artículo anterior, este permite programar un conjunto de viajes a un usuario pasajero, pero dicho usuario solo envía su solicitud, es el conductor quien ingresa su ruta, su hora de salida y también puede decidir, al igual que el usuario pasajero respecto a las personas con las desea compartir su viaje.

Finalmente, del tercer artículo mencionado en esta subsección [7], cabe destacar el seguimiento que un pasajero puede realizar al vehículo antes del viaje. Por lo cual, la ubicación de los vehículos participantes en esta aplicación siempre está comprometida, sin brindar mucha seguridad al conductor a excepción de poder observar los perfiles de las personas con las que compartirá el vehículo.

Con respecto a los artículos mencionados, se seleccionó como funcionalidad principal para la aplicación de *carpooling* de esta investigación, la programación de viajes, debido a que, en una aplicación de este tipo, los usuarios deben poder organizar y planificar de manera eficiente sus viajes compartidos, decidiendo horarios, confirmación, recordatorios y otros ajustes. Además, se determinó conveniente darle más libertad a los conductores para que decidan el viaje que quieren realizar, las personas a las que desean transportar, y permitirle calificarlas al terminar el viaje.

Para el usuario pasajero, se decidió incluir la funcionalidad de postular la ruta que desea y calificar al conductor luego de un viaje, con el fin de que otros usuarios puedan observar las calificaciones entregadas y que dicha calificación funcione como un parámetro de selección.

En la siguiente subsección se explica en detalle las funcionalidades de la aplicación propuesta.

Cabe resaltar que la funcionalidad de la aplicación de *carpooling*, en este proyecto, es limitada, ya que el objetivo principal de la aplicación es **validar la seguridad de los usuarios haciendo uso de la aplicación**, más no prestar un servicio con una amplia gama de funcionalidades ya que esto incrementa el alcance del proyecto considerablemente.

1.2. Descripción de la funcionalidad seleccionada

La aplicación implementada en el presente documento es una aplicación de transporte compartido tipo *carpooling* que permite a las personas utilizar un medio de transporte compartido. Este medio de transporte busca minimizar problemas como congestión y contaminación ambiental, intentando reducir el número de vehículos que transitan las vías de una ciudad.

En primer lugar, la aplicación proporciona un sistema de registro y autenticación facial para los conductores y pasajeros.

Para un usuario pasajero, su interacción con la aplicación de *carpooling* inicia con el proceso de registro, para esto, el usuario pasajero debe completar un formulario con su información, además de proporcionar una fotografía frontal y un pin de seguridad. Si el sistema acepta la información brindada por el usuario, éste podrá iniciar sesión y hacer uso de la aplicación.

Para el inicio de sesión de un usuario pasajero, la aplicación pide al usuario la toma de una fotografía, con la cual el sistema verifica la identidad de la persona que intenta ingresar a la aplicación. En caso de que la información biométrica suministrada coincida con la almacenada, el sistema le pide al usuario el pin de seguridad y si este último es correcto, el usuario finaliza correctamente el proceso de inicio de sesión.

Una vez el pasajero ingresa al sistema, mediante la autenticación descrita, procede a agregar su requerimiento de viaje, proporcionando la ubicación del lugar inicial y final del viaje, así como la hora aproximada en la que desea realizarlo. Al ingresar dicha información, la aplicación realiza las consultas respectivas y luego procede a informar

los posibles conductores que pueden atender el servicio del pasajero, dependiendo tanto de la ubicación inicial como del destino.

Posteriormente, el pasajero selecciona el servicio del conductor que mejor se ajuste a sus necesidades (si existe alguno), con lo cual envía una solicitud al conductor elegido para realizar el viaje. Si la solicitud es aceptada por el conductor, el pasajero recibe a través de la aplicación la aceptación de su solicitud (indicando lugar de recogida, y lugar de destino), con lo cual quedaría programado el servicio. Está disponible para el usuario la opción de cancelar viaje, en caso de que ya no desee tomar el servicio.

Al terminar el viaje, el pasajero puede calificar al conductor según su experiencia en el trayecto. Adicionalmente, el usuario puede observar la información de su perfil y modificar algunos datos que no sean relevantes a la seguridad de la aplicación. Por último, el pasajero puede cerrar la sesión cuando termine de usar la aplicación.

Para un usuario conductor, luego de realizar el registro, proporcionando la fotografía solicitada, su nombre, la placa y modelo de su vehículo, puede iniciar sesión cuando quiera interactuar en la aplicación, realizando la autenticación facial respectiva. El conductor puede postular una ruta, ingresando un punto inicial, un punto final de su recorrido y puntos de referencia dentro de este. Dicha ruta es la que se muestra a los pasajeros para seleccionar un viaje.

El conductor es quien toma la decisión final de las personas a las que desea transportar, aceptando o denegando las solicitudes según sea el caso, en el momento que le aparezca cada una de las solicitudes provenientes de los usuarios. En cualquier momento, el conductor puede cancelar la ruta, es decir, su ruta anteriormente postulada no se mostrará más a ningún usuario de la aplicación. Si se cancela una ruta, y ya se han aceptado servicios para los pasajeros, estos son informados de dicho evento. En caso de que no se cancele la ruta y se realice el viaje, el conductor puede brindar una calificación a los pasajeros de acuerdo a su experiencia en el trayecto. Además, la aplicación brinda la opción de gestionar perfil, en donde el conductor puede editar información que no sea relevante para la seguridad de la aplicación. Por último, el conductor puede cerrar sesión cuando termine la interacción en la aplicación.

También se brinda acceso a un usuario administrador, el cual no va a poder registrarse ya que la información de ingreso es única. Para iniciar sesión como administrador, el usuario debe ingresar el correo y la contraseña. Al ingresar, este tiene dos opciones, visualización de los perfiles de los usuarios y cerrar sesión.

2. Historias de usuario de la aplicación.

De la funcionalidad descrita en el punto 1.2, se identificaron tres tipos de usuario que son: pasajero, conductor y administrador, para cada uno de ellos se identificaron las siguientes historias de usuario:

Historias de usuario pasajero.

- Registro del pasajero.
- Inicio de sesión del pasajero.
- Visualización de perfil del pasajero.
- Calificación al conductor.
- Cerrar sesión del pasajero.
- Agregar viaje como pasajero.
- Visualización de rutas de interés por parte del pasajero.
- Selección de ruta del pasajero.
- Cancelar viaje como pasajero.

Historias de usuario conductor

- Registro del conductor
- Inicio de sesión del conductor.
- Visualización de perfil del conductor.
- Calificación a pasajeros.
- Cerrar sesión del conductor.
- Postular ruta por parte del conductor.
- Selección de pasajeros.
- Cancelar ruta de usuario conductor.

Historias de usuario administrador

- Inicio de sesión del administrador.
- Gestión de usuarios

- Cerrar sesión del administrador.

Las historias de usuario mencionadas se detallan a continuación.

2.1. Historias de usuario del pasajero

2.1.1. Registro del pasajero

ID HISTORIA	HU1
NOMBRE	Registro del pasajero
DESCRIPCIÓN	<p>Como usuario pasajero</p> <p>Quiero registrar mis datos personales (nombre, cédula, correo) y biométricos (reconocimiento facial por medio de la cámara web) en la aplicación de carpooling</p> <p>Para guardar esta información y después utilizarla para iniciar sesión.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Registro exitoso</p> <p>Dado que los datos biométricos no se han registrado anteriormente</p> <p>Cuando se complete el formulario de registro</p> <p>Entonces se muestra el mensaje de registro finalizado.</p> <p>Escenario 2: Registro fallido</p> <p>Dado que los datos biométricos ya se encuentran registrados</p> <p>Cuando se complete el formulario de registro</p> <p>Entonces se muestra el mensaje el usuario</p>

	ya se encuentra registrado
PRIORIDAD	10
ESTIMACIÓN	3 días

2.1.2. Inicio de sesión del pasajero

ID HISTORIA	HU2
NOMBRE	Inicio de sesión del pasajero
DESCRIPCIÓN	<p>Como usuario pasajero</p> <p>Quiero iniciar sesión</p> <p>Para acceder a la aplicación web</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Inicio de sesión exitoso</p> <p>Dado que la aplicación detecta similitudes faciales</p> <p>Cuando se usa la autenticación facial</p> <p>Entonces se muestra un mensaje inicio de sesión exitoso</p> <p>Escenario 2: Inicio de sesión fallido</p> <p>Dado que la aplicación no detecta similitudes faciales</p> <p>Cuando se usa la autenticación facial</p> <p>Entonces se muestra un mensaje error al iniciar sesión, vuelva a intentarlo</p>
PRIORIDAD	11
ESTIMACIÓN	3 días

2.1.3. Visualización de perfil del pasajero.

ID HISTORIA	HU3
NOMBRE	Visualización de perfil del pasajero.
DESCRIPCIÓN	Como usuario pasajero Quiero revisar mi perfil Para ver mi información y editarla
CRITERIOS DE ACEPTACIÓN	Escenario 1: Ver información asociada a mi perfil. Dado que quiero observar mi información Cuando ingreso a “Perfil” Entonces puedo observar todos mis datos.
PRIORIDAD	14
ESTIMACIÓN	2 días

2.1.4. Cerrar sesión del pasajero

ID HISTORIA	HU4
NOMBRE	Cerrar sesión del pasajero.
DESCRIPCIÓN	Como usuario pasajero Quiero poder cerrar mi sesión en la aplicación. Para garantizar mi privacidad.
CRITERIOS DE ACEPTACIÓN	Escenario 1: proceso exitoso de cerrar sesión.

	<p>Dado que el usuario quiere cerrar sesión</p> <p>Cuando termina de usar temporalmente la aplicación</p> <p>Entonces la sesión es cerrada y para ingresar debe volver a realizar el proceso de autenticación.</p>
PRIORIDAD	16
ESTIMACIÓN	1 día

2.1.5. Agregar viaje como pasajero

ID HISTORIA	HU5
NOMBRE	Agregar viaje como pasajero
DESCRIPCIÓN	<p>Como usuario pasajero</p> <p>Quiero poder agregar mi lugar de inicio y destino</p> <p>Para poder solicitar el servicio de transporte según mis necesidades.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Ingresar punto inicial y final del recorrido</p> <p>Dado que un usuario agrega ubicación inicial y final</p> <p>Cuando desea realizar un viaje</p> <p>Entonces se postula el viaje para observar las opciones brindadas en la aplicación.</p>
PRIORIDAD	2

ESTIMACIÓN	5 días
-------------------	--------

2.1.6. Visualización de rutas de interés por parte del pasajero.

ID HISTORIA	HU6
NOMBRE	Visualización de rutas de interés por parte del pasajero
DESCRIPCIÓN	<p>Como usuario pasajero</p> <p>Quiero poder observar todas las rutas disponibles dependiendo de mi ubicación actual y mi destino</p> <p>Para poder escoger una ruta y el conductor que realice dicho viaje.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Existen rutas para las ubicaciones ingresadas.</p> <p>Dado que un usuario ingresa la información de ubicación inicial y de destino y se encuentran rutas que coinciden con dicha información</p> <p>Cuando desea realizar un viaje</p> <p>Entonces se observan todas las rutas de interés que están disponibles.</p> <p>Escenario 2: No existen rutas disponibles para las ubicaciones ingresadas.</p> <p>Dado que un usuario ingresa la información de ubicación inicial y de destino y no se encuentran rutas que</p>

	coinciden con dicha información Cuando desea realizar un viaje Entonces se muestra una notificación al usuario en donde se manifieste que no existen rutas disponibles para la ubicación ingresada.
PRIORIDAD	3
ESTIMACIÓN	5 días

2.1.7. Selección de ruta del pasajero

ID HISTORIA	HU7
NOMBRE	Selección de ruta del pasajero
DESCRIPCIÓN	Como usuario pasajero Quiero poder seleccionar una ruta que cubra algún conductor para solicitar cupo en el automóvil. Para poder enviar la solicitud al conductor relacionado.
CRITERIOS DE ACEPTACIÓN	Escenario 1: Mensaje de confirmación de ruta. Dado que se acepta una ruta seleccionada. Cuando se registra un punto de partida y un punto final Entonces aparece el recorrido del conductor con la información correspondiente al viaje e información del conductor que sea relevante.

	<p>Escenario 2: No se obtiene confirmación de ruta</p> <p>Dado no se acepta la solicitud del viaje</p> <p>Cuando se envía la solicitud al chofer</p> <p>Entonces aparece una notificación indicando que el conductor rechazó el viaje.</p>
PRIORIDAD	4
ESTIMACIÓN	5 días

2.1.8. Cancelar viaje como usuario pasajero

ID HISTORIA	HU8
NOMBRE	Cancelar viaje como usuario pasajero
DESCRIPCIÓN	<p>Como usuario pasajero</p> <p>Quiero poder cancelar una solicitud de viaje</p> <p>Para poder eliminar las ubicaciones brindadas cuando ya no deseo realizarlo.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Cancelar viaje.</p> <p>Dado que un usuario ha ingresado información suficiente para un viaje, entonces se elimina la información brindada</p> <p>Cuando ya no desea realizar el viaje</p>

	Entonces la solicitud se eliminará de la aplicación.
PRIORIDAD	6
ESTIMACIÓN	2 días

2.1.9. Calificación al conductor.

ID HISTORIA	HU9
NOMBRE	Calificación al conductor
DESCRIPCIÓN	<p>Como usuario pasajero</p> <p>Quiero poder calificar al conductor con quien realicé un viaje.</p> <p>Para recomendar y retroalimentar al conductor.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Guardar la información de calificación.</p> <p>Dado Que se realizó un viaje con un conductor determinado</p> <p>Cuando el viaje finaliza el usuario puede calificar al conductor</p> <p>Entonces la calificación es guardada y la pestaña de calificación al conductor se cierra.</p>
PRIORIDAD	8
ESTIMACIÓN	2 días

2.2. Historias del usuario conductor

2.2.1. Registro de usuario del conductor

ID HISTORIA	HU10
NOMBRE	Registro de usuario del conductor
DESCRIPCIÓN	<p>Como usuario conductor</p> <p>Quiero registrar mis datos personales (nombre, cédula, correo) y biométricos (reconocimiento facial por medio de la cámara web) en la aplicación de carpooling</p> <p>Para guardar esta información y después utilizarla para iniciar sesión.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Registro exitoso</p> <p>Dado que los datos biométricos no se han registrado anteriormente</p> <p>Cuando se complete el formulario de registro</p> <p>Entonces se muestra el mensaje de registro finalizado.</p> <p>Escenario 2: Registro fallido</p> <p>Dado que los datos biométricos ya se encuentran registrados</p> <p>Cuando se complete el formulario de registro</p> <p>Entonces se muestra el mensaje el usuario ya se encuentra registrado</p>
PRIORIDAD	12
ESTIMACIÓN	3 días

2.2.2. Inicio de sesión de conductor

ID HISTORIA	HU11
NOMBRE	Inicio de sesión de conductor
DESCRIPCIÓN	Como usuario conductor Quiero iniciar sesión Para acceder a la aplicación web
CRITERIOS DE ACEPTACIÓN	Escenario 1: Inicio de sesión exitoso Dado que la aplicación detecta similitudes faciales Cuando se usa la autenticación facial Entonces se muestra un mensaje inicio de sesión exitoso Escenario 2: Inicio de sesión fallido Dado que la aplicación no detecta similitudes faciales Cuando se usa la autenticación facial Entonces se muestra un mensaje error al iniciar sesión, vuelva a intentarlo
PRIORIDAD	13
ESTIMACIÓN	3 días

2.2.3. Visualización de perfil del conductor.

ID HISTORIA	HU12
NOMBRE	Visualización de perfil de usuario conductor

DESCRIPCIÓN	Como usuario conductor Quiero revisar mi perfil Para ver mi información y editarla
CRITERIOS DE ACEPTACIÓN	Escenario 1: Ver información asociada a mi perfil. Dado que quiero observar mi información Cuando ingreso a “Perfil” Entonces puedo observar todos mis datos.
PRIORIDAD	15
ESTIMACIÓN	2 días

2.2.4. Cerrar sesión del conductor

ID HISTORIA	HU13
NOMBRE	Cerrar sesión del conductor
DESCRIPCIÓN	Como usuario conductor Quiero poder cerrar mi sesión en la aplicación. Para garantizar mi privacidad.
CRITERIOS DE ACEPTACIÓN	Escenario 1: proceso exitoso de cerrar sesión. Dado que el usuario quiere cerrar sesión Cuando termina de usar temporalmente la aplicación Entonces la sesión es cerrada y para ingresar debe volver a realizar el

	proceso de autenticación.
PRIORIDAD	17
ESTIMACIÓN	1 día

2.2.5. Postular ruta por parte del conductor

ID HISTORIA	HU14
NOMBRE	Postular ruta por parte del conductor
DESCRIPCIÓN	<p>Como usuario conductor</p> <p>Quiero poder postular una ruta de mi interés.</p> <p>Para ofrecer cupos en mi automóvil a otras personas que compartan la misma ruta.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Ruta correctamente postulada</p> <p>Dado que el conductor ingresa información referente a su ubicación inicial, final y la hora en que realizará el viaje</p> <p>Cuando desea compartir un viaje</p> <p>Entonces la información es agregada a una nueva ruta.</p>
PRIORIDAD	1
ESTIMACIÓN	6 días

2.2.6. Selección de pasajeros

ID HISTORIA	HU15
NOMBRE	Selección de pasajeros
DESCRIPCIÓN	<p>Como usuario conductor Quiero poder aceptar o rechazar solicitudes de pasajeros Para tener control de las personas con las que compartiré mi ruta.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: solicitud de pasajero aceptada. Dado que el conductor acepta una solicitud de un pasajero Cuando postula un viaje Entonces aparece información relevante del pasajero junto con su lugar de recogida y llegada.</p> <p>Escenario 2: solicitud de pasajero rechazada. Dado que el conductor rechaza una solicitud de un pasajero Cuando postula un viaje Entonces la información de la solicitud del pasajero desaparece en la pantalla del dispositivo del conductor.</p>
PRIORIDAD	5
ESTIMACIÓN	8 días

2.2.7. Cancelar ruta como usuario conductor.

ID HISTORIA	HU16
NOMBRE	Cancelar ruta como usuario conductor
DESCRIPCIÓN	<p>Como usuario conductor</p> <p>Quiero poder cancelar una ruta brindada</p> <p>Para poder eliminar las ubicaciones dadas cuando ya no deseo realizarlo.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Cancelar viaje.</p> <p>Dado que un usuario ha ingresado información suficiente para postular una ruta, entonces se elimina la información brindada</p> <p>Cuando ya no desea realizar el viaje</p> <p>Entonces la solicitud se eliminará de la aplicación.</p>
PRIORIDAD	7
ESTIMACIÓN	2 días

2.2.8. Calificación a pasajeros.

ID HISTORIA	HU17
NOMBRE	Calificación a pasajeros.
DESCRIPCIÓN	<p>Como usuario conductor</p> <p>Quiero poder calificar a mis pasajeros</p> <p>Para recomendar y retroalimentar a cada uno de ellos</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Guardar la información de calificación.</p>

	<p>Dado Que se realizó un viaje con diferentes pasajeros</p> <p>Cuando el viaje finaliza el conductor puede calificar a cada pasajero que hizo parte del viaje</p> <p>Entonces la calificación es guardada y la pestaña de calificación es cerrada.</p>
PRIORIDAD	9
ESTIMACIÓN	1 día

2.3. Historias de usuario del administrador

2.3.1. Inicio de sesión del administrador

ID HISTORIA	HU18
NOMBRE	Inicio de sesión del administrador
DESCRIPCIÓN	<p>Como usuario administrador</p> <p>Quiero iniciar sesión</p> <p>Para acceder a la aplicación web</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: Inicio de sesión exitoso</p> <p>Dado que la información coincide con la almacenada</p> <p>Cuando se ingresa la contraseña</p> <p>Entonces se muestra un mensaje inicio de sesión exitoso</p> <p>Escenario 2: Inicio de sesión fallido</p> <p>Dado que la información no coincide</p>

	con la almacenada Cuando se ingresa la contraseña Entonces se muestra un mensaje error al iniciar sesión, vuelva a intentarlo
PRIORIDAD	1
ESTIMACIÓN	3 días

2.3.2. Gestión de usuarios

ID HISTORIA	HU19
NOMBRE	Gestión de usuarios
DESCRIPCIÓN	Como usuario administrador Quiero poder administrar información de usuarios en caso de alguna eventualidad. Para garantizar la seguridad y la eficacia de la aplicación.
CRITERIOS DE ACEPTACIÓN	Escenario 1: el usuario observa la información básica de los usuarios en caso de alguna eventualidad que así lo requiera. Como administrador, puede habilitar o deshabilitar un usuario si la situación lo requiere.
PRIORIDAD	2
ESTIMACIÓN	4 día

2.3.3. Cerrar sesión del administrador

ID HISTORIA	HU20
NOMBRE	Cerrar sesión del administrador
DESCRIPCIÓN	<p>Como usuario administrador</p> <p>Quiero poder cerrar mi sesión en la aplicación.</p> <p>Para garantizar mi privacidad.</p>
CRITERIOS DE ACEPTACIÓN	<p>Escenario 1: proceso exitoso de cerrar sesión.</p> <p>Dado que el usuario quiere cerrar sesión</p> <p>Cuando termina de usar temporalmente la aplicación</p> <p>Entonces la sesión es cerrada y para ingresar debe volver a realizar el proceso de autenticación.</p>
PRIORIDAD	3
ESTIMACIÓN	1 día

3. Diseño de vistas

El diseño de vistas es una etapa importante en el desarrollo de aplicaciones, las vistas son las interfaces gráficas o visuales a través de las cuales los usuarios interactúan con el sistema. Estas vistas pueden incluir pantallas, formularios, gráficos, tablas, etc. [10].

Las principales razones que se tuvieron en cuenta para incluir el diseño de vistas en el desarrollo de este proyecto son las siguientes:

- Cumplimiento de los requisitos del usuario. Al justificar el diseño de vistas, se asegura de que las necesidades del usuario se hayan tenido en cuenta. La identificación de los requisitos del usuario y el diseño de vistas para satisfacer estos requisitos asegura que el sistema se adapte a las necesidades y expectativas del usuario.
- Identificación de problemas. La justificación del diseño de vistas también ayuda a identificar posibles problemas o limitaciones en el diseño.

En la Figura 1 se muestra un diseño tentativo de la primera vista de la aplicación web, donde se encuentran los botones de inicio de sesión y de registro. Esta pantalla está disponible para todos los usuarios existentes.

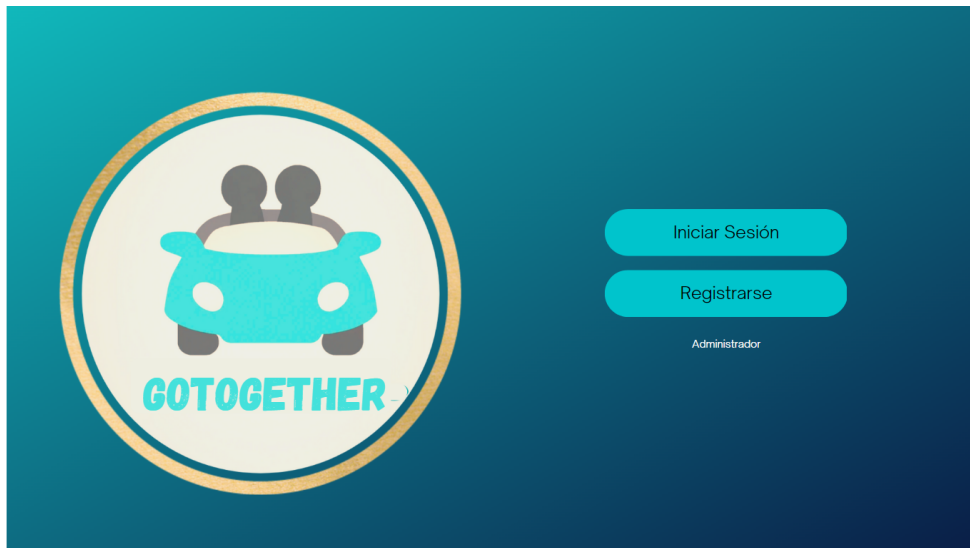


Figura 1. Inicio de aplicación.

Para los usuarios viajeros y conductores, la autenticación se realiza mediante reconocimiento facial, cuando se ingresa a la opción de iniciar sesión presentada en la Figura 1. En la parte inferior de los dos botones observados, aparece una opción para ingresar como usuario administrador, la cual no requiere autenticación biométrica. No hay posibilidad de registrarse como administrador, por restricciones del alcance de la aplicación.

3.1. Vistas de las historias de usuario de pasajero

3.1.1. Vista del registro de usuario pasajero

En la Figura 2 se observa el formulario necesario para poder registrarse en la aplicación



Figura 2. Registro de usuario pasajero.

En el formulario se piden datos básicos al usuario como el nombre y la cédula, además, se necesita que el usuario especifique que su registro es como pasajero. Adicional a dicha información, se requiere que el usuario se tome una foto que se utilizará para verificar su identidad al momento de registrarse.

En caso de que el registro sea erróneo, se muestra una notificación como en la Figura 3.

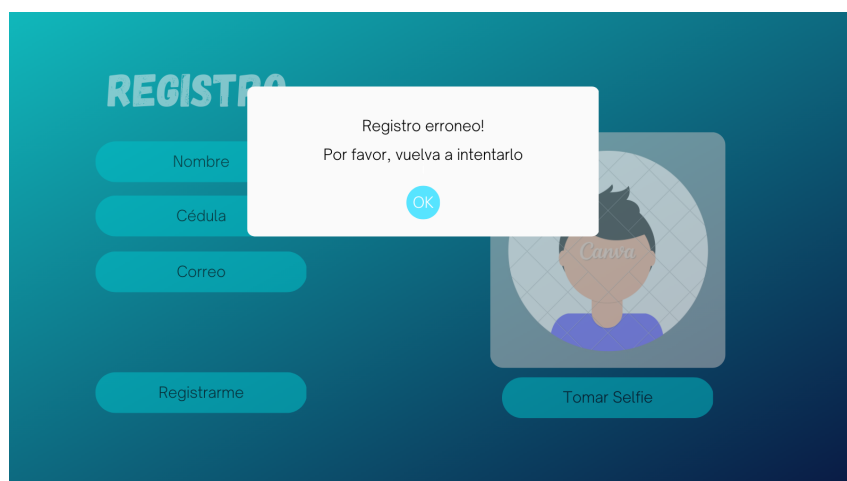


Figura 3. Registro erróneo.

En caso de que el registro sea correcto, se muestra un mensaje de éxito mostrado en la Figura 4.

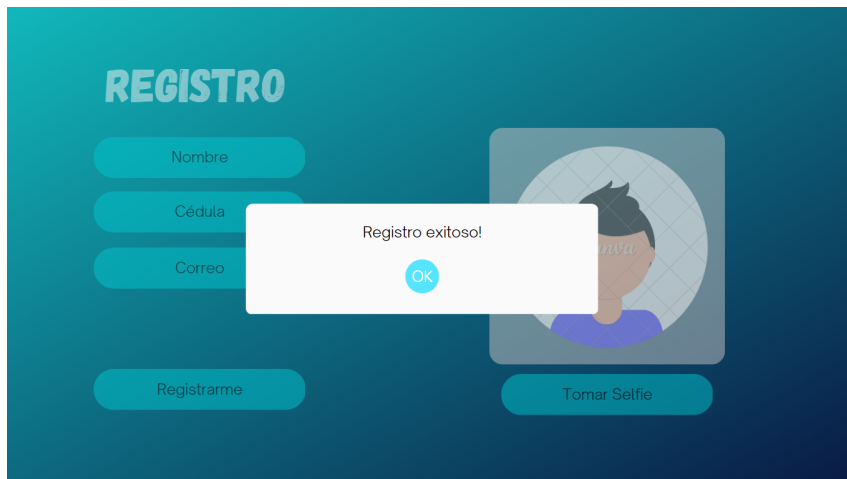


Figura 4. Registro exitoso.

3.1.2. Vista de inicio de sesión de pasajero.

El inicio de sesión se realiza cuando un usuario pasajero quiere ingresar a la aplicación, para lo cual, en la vista de la Figura 1 se ingresa a la opción "Iniciar sesión". Al abrir la vista de iniciar sesión, se despliega la cámara para que el usuario ingrese a la aplicación por medio de reconocimiento facial, como se muestra en la Figura 5.

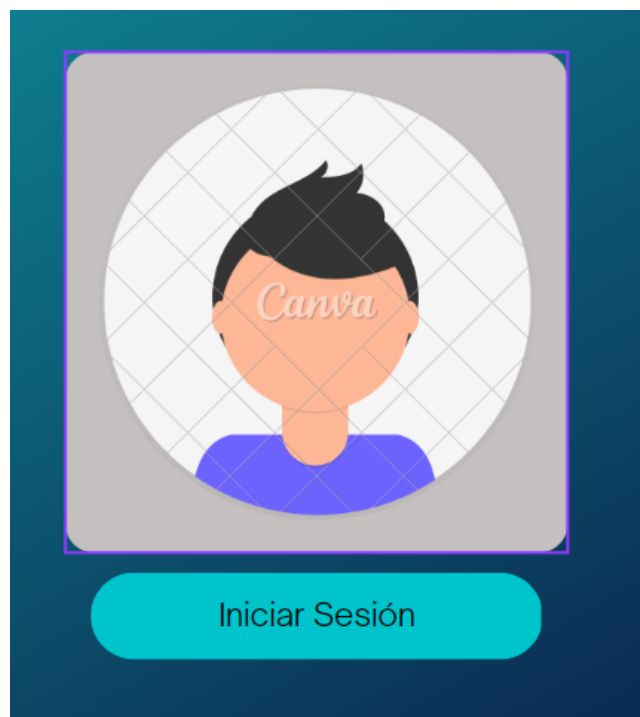


Figura 5. Inicio de sesión.

En caso de que no se pueda iniciar sesión, se muestra una notificación en donde indique que el proceso de iniciar sesión falló, tal como en la Figura 6.

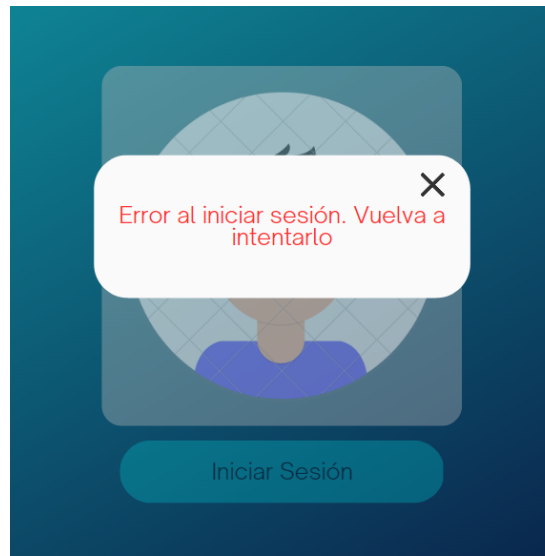


Figura 6. Inicio de sesión erróneo.

En caso contrario, el usuario pasajero puede ingresar a la aplicación y se dirige a la pantalla de inicio.

En la Figura 7, se observa la pantalla principal de un usuario tipo pasajero. Es la primera vista que tiene el usuario de la aplicación luego de iniciar sesión, aquí se encuentra un menú en la parte superior en donde el usuario puede ver su información de perfil (opción "Perfil"), ver la pantalla principal de la aplicación ("Inicio") y cerrar sesión. Además, se encuentra un mapa y en la parte inferior dos opciones, Agregar viaje y Cancelar Viaje. En un inicio, si el pasajero no tiene viajes programados, el botón de "Cancelar viaje" está deshabilitado.

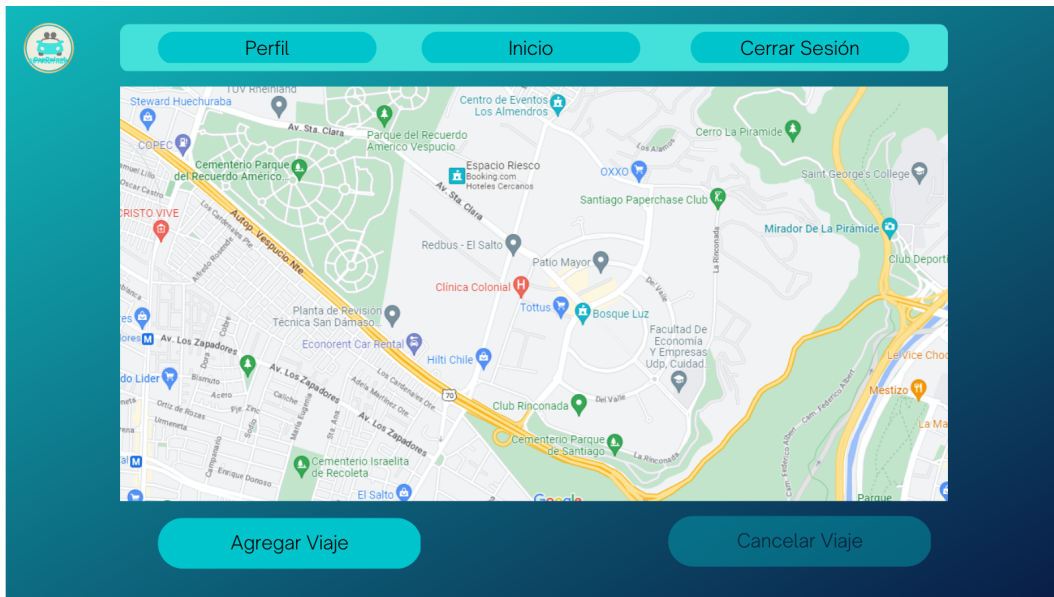


Figura 7. Pantalla de inicio usuario pasajero.

3.1.3. Vista de visualización del perfil de usuario pasajero.

Para que el pasajero pueda observar su información de perfil, en la vista presentada en la Figura 7 se dirige a la opción “Perfil”.

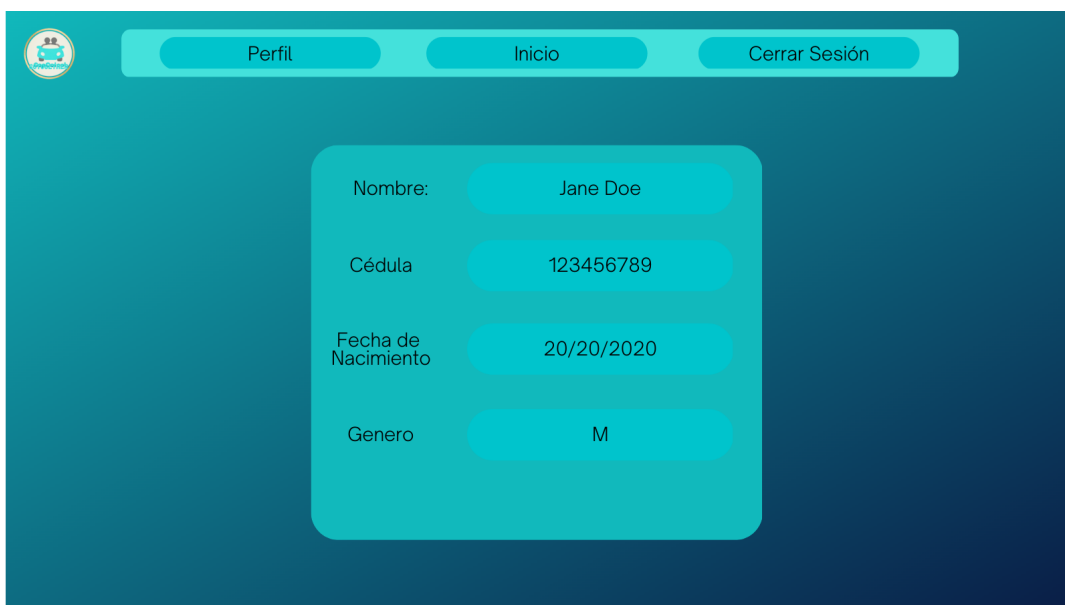


Figura 8. Perfil de usuario.

3.1.4. Vista de cerrar sesión del usuario pasajero

Para cerrar sesión, el usuario se dirige a la opción de “Cerrar sesión” que se encuentra en la parte superior del menú principal de la aplicación (vista de la Figura 7).

Al cerrar sesión, se redirige al usuario a la pantalla principal de la aplicación (vista de la Figura 1).

3.1.5. Vista de agregar viaje como pasajero.

Para agregar un viaje como usuario pasajero, se ingresa al botón presentado en la parte inferior izquierda de la Figura 7, llamado “Agregar viaje”, en donde se despliega la vista presentada en la Figura 9 que incluye un formulario.

El usuario pasajero ingresa su ubicación de inicio y su destino, así como la hora aproximada en la que desea el viaje, al aceptar los datos ingresados en el formulario (botón “OK”), se debe quedar en espera de resultados de rutas que coincidan.



Figura 9. Formulario para agregar viaje.

3.1.6. Vista de visualización de rutas de interés del usuario pasajero.

Para observar las rutas de interés, el pasajero debe ingresar primero la información correspondiente al viaje que desea realizar.

Al agregar un viaje, se observan las posibles rutas que se pueden tomar (Figura 10), el pasajero elige a qué conductor solicitarle un cupo en su auto.

Para observar una posible ruta, se selecciona al conductor deseado, en donde aparece en el mapa la ruta del conductor con la información de interés del viaje. Por ejemplo, en la Figura 10 se seleccionó al “Conductor 1” y en el mapa se muestra el lugar de recogida cercano a la ubicación inicial del pasajero, y la ruta hasta el lugar de destino, que debería estar cercano al punto solicitado por el pasajero como ubicación de llegada.

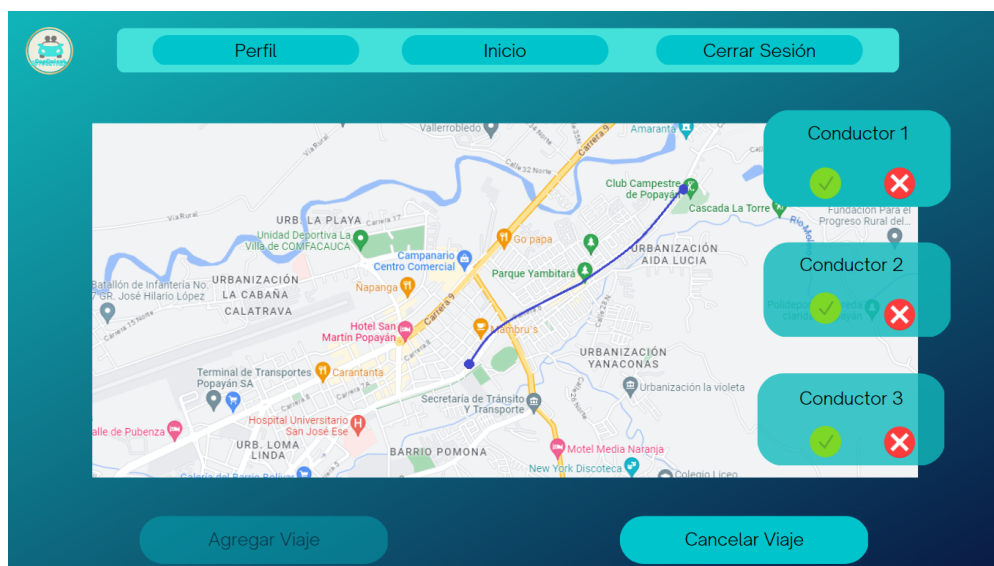


Figura 10. Rutas disponibles.

3.1.7. Vista de selección de ruta de usuario pasajero

Cuando un pasajero envía una solicitud a un conductor, el pasajero queda a la espera de la respuesta. En caso de que un conductor acepte, en el mapa se observa la ruta del conductor (color azul) y el punto de recogida (punto rojo) como en la Figura 11. Además, se observan algunos datos del conductor relevantes para brindar seguridad al pasajero. Es aquí, donde se habilita el botón “Cancelar viaje”, el cual le permite al pasajero eliminar un viaje que tenía programado.

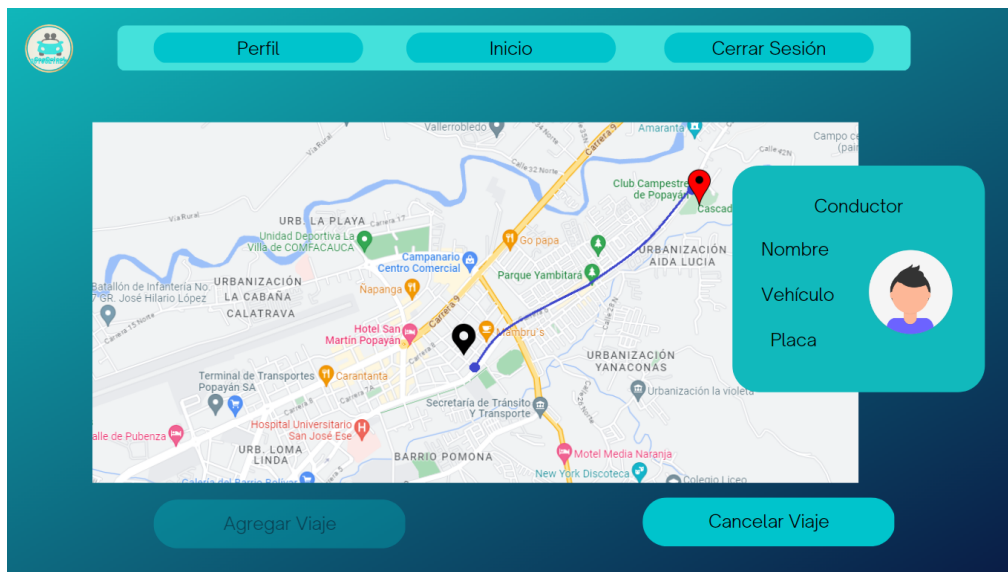


Figura 11. Vista del viaje.

3.1.8. Vista de cancelar viaje como usuario pasajero

Para cancelar un viaje, un pasajero puede seleccionar el botón de “Cancelar viaje”, ubicado en la parte inferior derecha de la Figura 7, al hacerlo, redirige al usuario a la pantalla principal de un usuario pasajero (Figura 7) y se elimina la información del viaje.

3.1.9. Vista de calificación al conductor

Al terminar un viaje, el pasajero tiene la posibilidad de calificar el servicio de un conductor, tal como se muestra en la Figura 12, al enviar la calificación, esta pestaña se cerrará automáticamente.

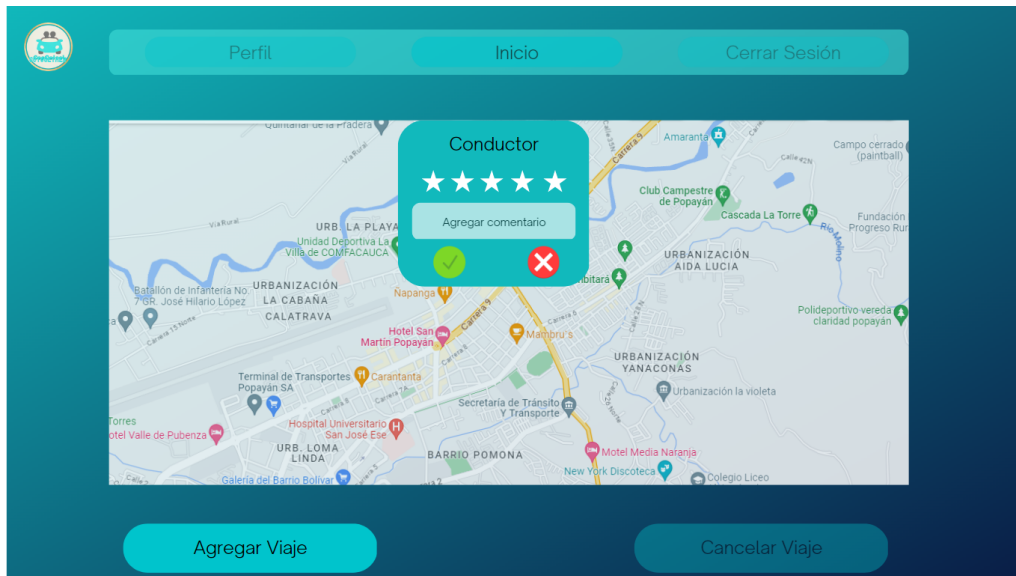


Figura 12. Calificación a conductor.

3.2. Vistas de las historias de usuario conductor

3.2.1. Vista de registro de usuario conductor

Para realizar un registro como usuario conductor, en la Figura 1 se ingresa a la opción de “Registrarse”, en donde al seleccionar un registro como conductor, se muestra el siguiente formulario (Figura 13).

En el formulario se piden datos básicos al usuario como el nombre, la cédula, la placa y la marca del vehículo. Adicional a dicha información, se requiere que el usuario se tome una foto que se utilizará para verificar su identidad al momento de registrarse.



Figura 13. Registro de usuario conductor.

En caso de que el registro sea erróneo, se muestra una notificación como en la Figura 14.

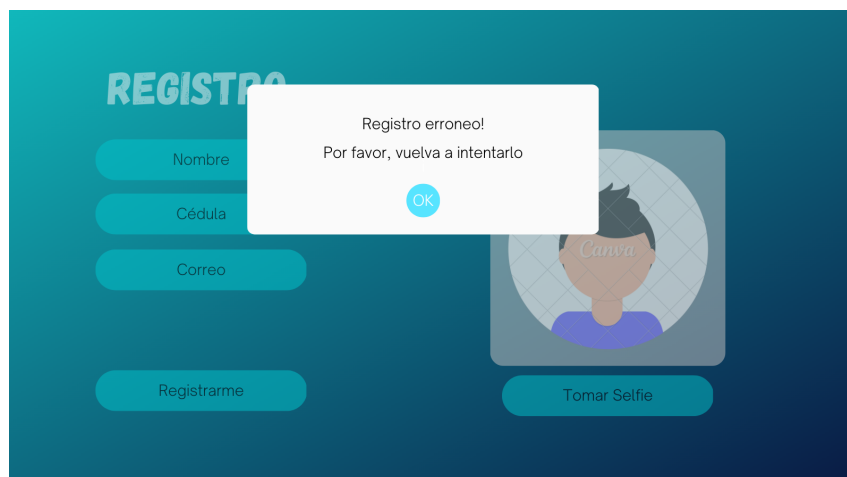


Figura 14. Registro erróneo.

En caso de que el registro sea correcto, se muestra un mensaje de éxito mostrado en la Figura 15.

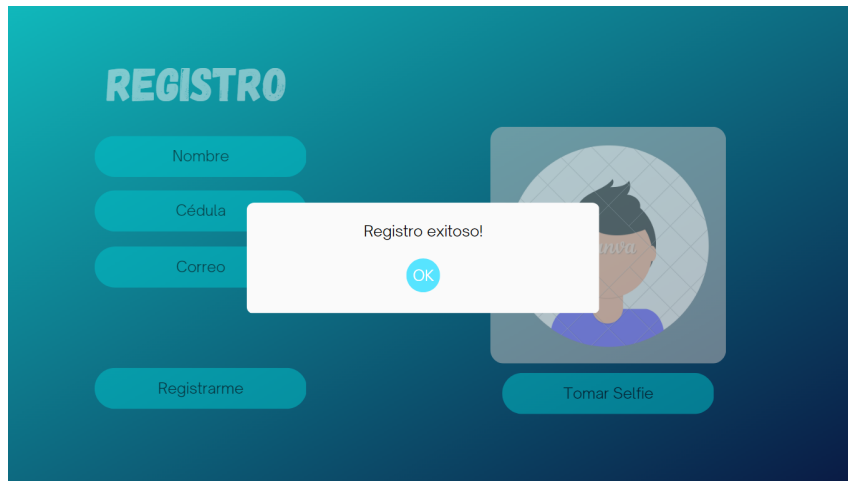


Figura 15. Registro exitoso.

3.2.2. Vista de inicio de sesión de usuario conductor

Para iniciar sesión como conductor, en la Figura 1 se ingresa a la opción “Iniciar sesión” en donde se despliega la cámara para que el usuario ingrese a la aplicación por medio de reconocimiento facial, como se muestra en la Figura 16.

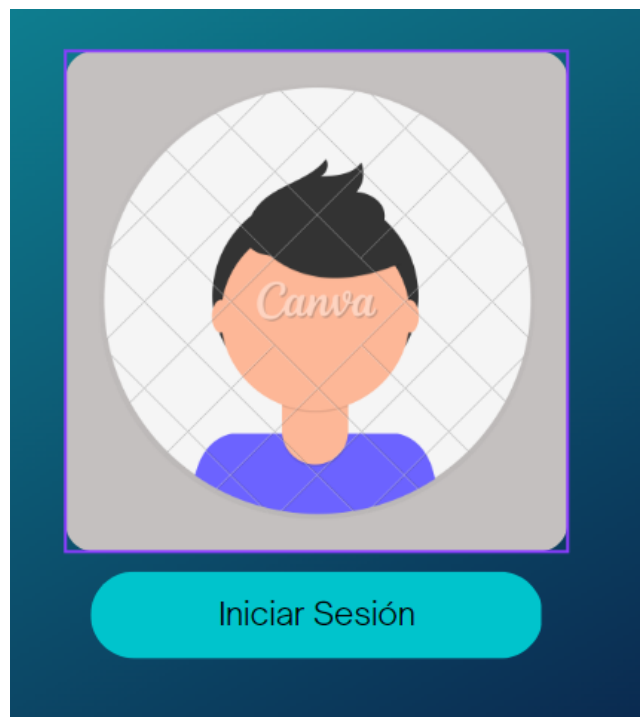


Figura 16. Inicio de sesión de conductor.

En caso de que no se pueda iniciar sesión, se muestra una notificación en donde indique que el proceso de iniciar sesión falló, tal como en la Figura 17.

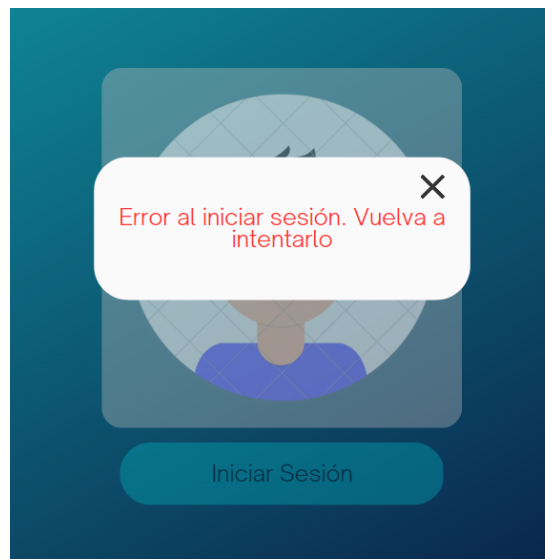


Figura 17. Inicio de sesión erróneo.

En caso contrario, el usuario conductor puede ingresar a la aplicación y se dirige a la pantalla de inicio.

Al ingresar sesión como usuario conductor, la primera vista será la presentada en la Figura 18, en donde en la parte superior se encuentra un menú en donde se muestran 3 opciones, perfil, inicio y cerrar sesión.

Bajo el menú mencionado, se encuentra un mapa general y bajo este se encuentran dos opciones: postular ruta y cancelar ruta. Al entrar a la aplicación sin una ruta postulada, la opción de "cancelar ruta" está deshabilitada ya que no hay rutas disponibles para eliminar.

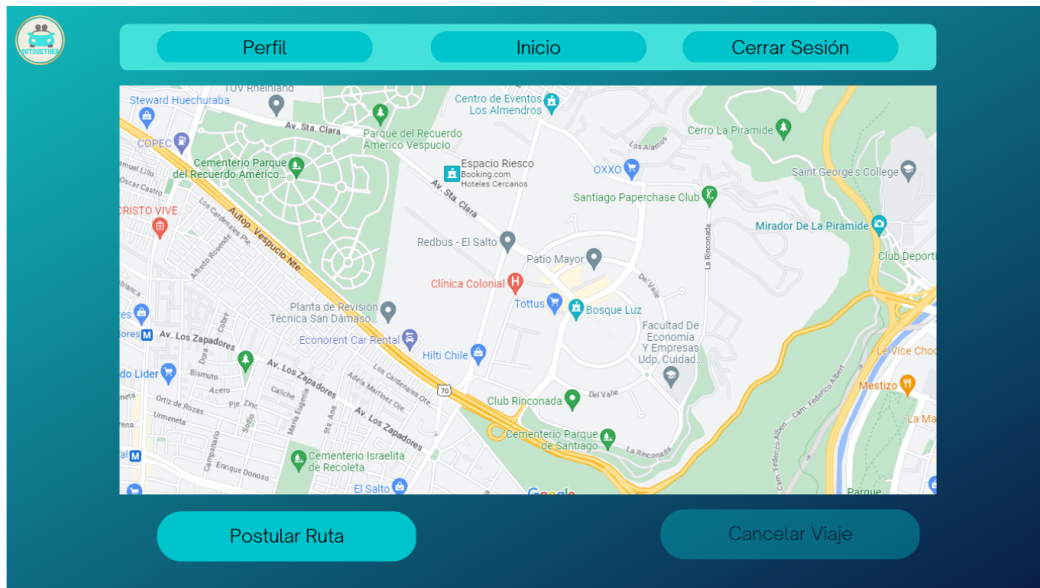


Figura 18. Vista principal de usuario conductor.

3.2.3. Vista de visualización de perfil de usuario conductor

Para observar la información del perfil de un conductor, en la vista presentada en Figura 18 se dirige a la opción “Perfil”. En dicha opción se tiene la vista presentada en la Figura 19, en donde puede observar la información del usuario.



Figura 19. Vista del perfil de usuario conductor.

3.2.4. Vista de cerrar sesión usuario conductor

Para cerrar sesión, al igual que el usuario pasajero, el conductor se dirige a la opción de “Cerrar sesión” que se encuentra en la parte superior del menú principal de la aplicación (Figura 18).

Posteriormente, se redirige al usuario conductor a la pantalla de inicio de la aplicación (vista de la Figura 1).

3.2.5. Vista de postular ruta usuario conductor

Para postular una ruta, en la vista de la Figura 18 el usuario ingresa a la opción de “Agregar Ruta”. En la vista de dicha opción aparece un formulario como el mostrado en la Figura 20, en donde se solicita información de inicio, fin y puntos de referencia de la ruta a crear, además de la hora en la que se inicia el recorrido.

Para ingresar puntos de referencia en la ruta a postular, se busca el punto de referencia ingresando el nombre en el campo del formulario de “Agregar punto de referencia”, si se desea agregar más puntos, se presiona el botón “más” de la Figura 20.



Figura 20. Formulario para agregar ruta.

3.2.6. Vista de selección de pasajeros

Luego de agregar una ruta, el usuario conductor puede consultar las solicitudes de los posibles pasajeros, junto con información básica que no comprometa la seguridad de estos. El conductor puede aceptar o denegar las solicitudes de los pasajeros, como se presenta en la vista de la Figura 21, en donde al seleccionar un pasajero, el conductor puede observar el lugar de recogida, la ubicación del pasajero y el punto de destino.

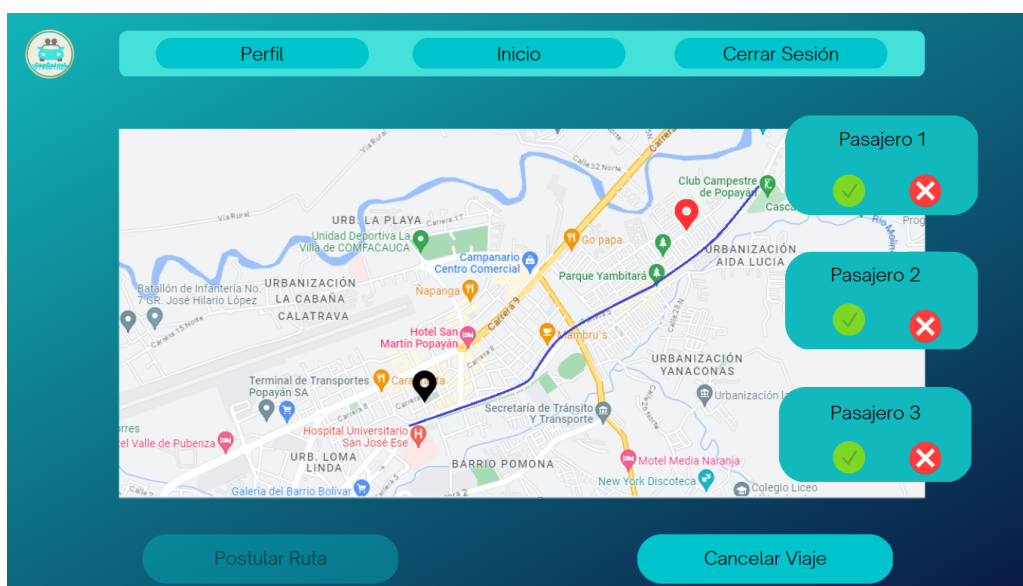


Figura 21. Solicitud de pasajeros.

3.2.7. Vista de calificación a pasajeros.

Al finalizar un viaje, el conductor puede agregar una calificación a los pasajeros, como se muestra en la Figura 22.

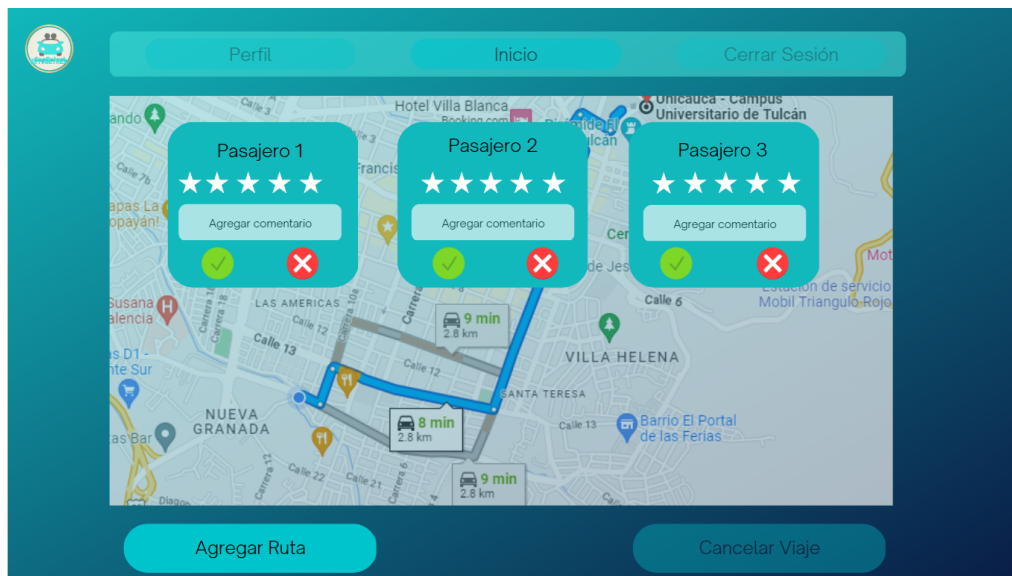


Figura 22. Calificación de pasajeros.

3.7.8. Vista de cancelar ruta.

Para cancelar un viaje, un conductor puede seleccionar el botón de “Cancelar ruta”, ubicado en la parte inferior derecha de la Figura 18, al hacerlo, elimina la ruta y no aparecerá a otros usuarios. Posteriormente, se redirige al usuario a la pantalla principal de un usuario conductor (Figura 18).

3.3. Vistas de las historias de usuario administrador

3.3.1. Vista de iniciar sesión de usuario administrador

Para iniciar sesión como usuario administrador, se necesita el usuario y contraseña de este. Cabe mencionar que un usuario no se registra en la aplicación como administrador, es decir, hay un único administrador con una contraseña predeterminada, debido a limitaciones de alcance de la aplicación.

En la vista de la Figura 23, se ingresa el usuario y la contraseña requeridos.

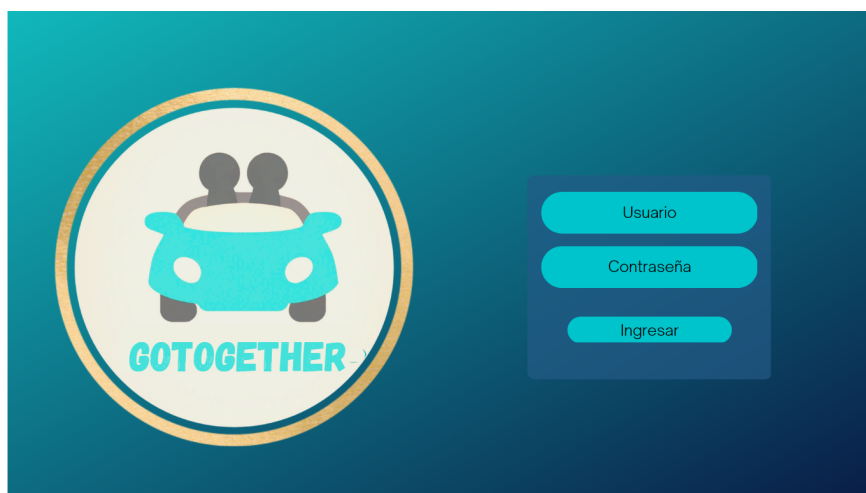


Figura 23. Inicio de sesión administrador.

En caso de que se ingresen credenciales incorrectas, aparece un mensaje de error en el proceso.

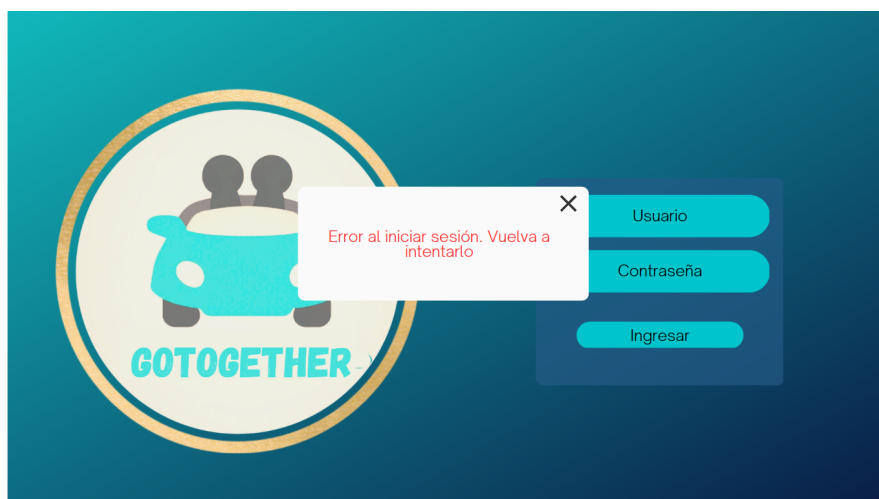


Figura 24. Inicio de sesión erróneo.

Si hubo un inicio de sesión correcto, aparece la pantalla principal del usuario administrador que corresponde a la mostrada en la Figura 25.

3.3.2. Vistas de gestión de usuarios



Figura 25. Pantalla de administración de usuarios.

En la Figura 25 se observan los diferentes usuarios con alguna información que se proporcionó al realizar la autenticación.

3.3.3. Vista de cerrar sesión usuario administrador

En la vista presentada en la Figura 25, en la parte superior derecha se muestra un botón que indica “Cerrar sesión”, el cual cierra la conexión del usuario administrador y redirige al usuario a la página principal de la aplicación (vista presentada en la Figura 1).

4. Diseño de la experiencia de usuario

El diseño de la experiencia de usuario (*UX design* por sus siglas en inglés), es el proceso de diseñar la interacción que los usuarios tienen con una aplicación, sitio web, producto o servicio, con el objetivo de crear una experiencia satisfactoria y efectiva para el usuario. El *UX design* se enfoca en la experiencia del usuario en todas las etapas del proceso de interacción, desde la primera impresión al descubrir la aplicación hasta la realización de las tareas que desea [11].

El *UX design* también se enfoca en factores como la arquitectura de la información, el diseño visual, la interacción y la usabilidad, todo ello con el objetivo de crear una experiencia coherente y atractiva para el usuario [11].

Para este proyecto, se establecieron los siguientes criterios de UX:

- Usabilidad. Es la facilidad con la que los usuarios pueden utilizar una aplicación para lograr sus objetivos. Los criterios de usabilidad incluyen la claridad del diseño, la facilidad de navegación y la eficiencia en la realización de tareas.
- Atractivo visual. Se refiere a la apariencia estética y la calidad del diseño de la aplicación. Los criterios de atractivo visual incluyen el diseño coherente, la consistencia en la marca, la calidad de las imágenes y la simplicidad visual.
- Funcionalidad. Es la capacidad de la aplicación para cumplir con las necesidades y objetivos de los usuarios. Los criterios de funcionalidad incluyen la capacidad para realizar tareas específicas, la rapidez y la capacidad de respuesta de la aplicación.

De acuerdo con los criterios mencionados, se hicieron algunos ajustes en algunas historias de usuario y vistas para mejorar la experiencia de usuario.

En primer lugar, en la historia de usuario HU5 (agregar viaje como pasajero), los puntos de inicio y final se había propuesto agregarlos escribiendo el lugar donde se encuentran. Sin embargo, teniendo en cuenta la usabilidad de la aplicación, sería más intuitivo para un usuario si este puede seleccionar estos puntos directamente en el mapa. Por otra parte, en la historia HU14 (postular ruta como conductor), un usuario conductor debería ser capaz de postular su ruta y los puntos intermedios, señalándolos directamente en el mapa y no sólo escribiendo una descripción de estos.

Por lo mencionado, las vistas correspondientes a las HU5 y HU14 fueron modificadas como se observa en las Figuras 26 y 27 respectivamente.

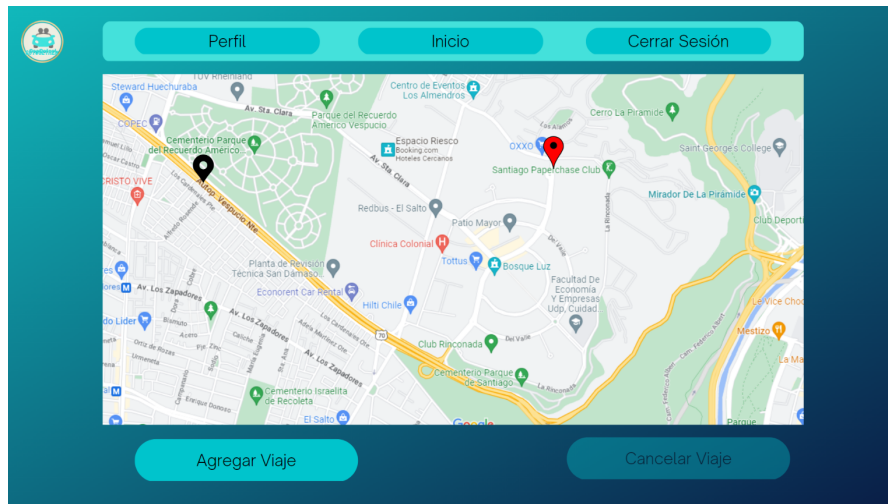


Figura 26. Agregar viaje como pasajero.

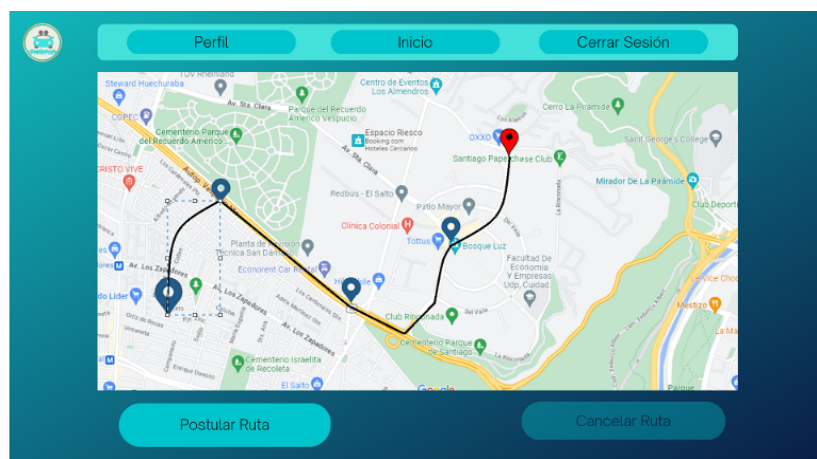


Figura 27. Postular ruta como conductor.

Finalmente, en las HU8 (cancelar viaje como usuario pasajero) e HU16 (Cancelar ruta como usuario conductor), cuando los usuarios ya sean pasajeros o conductores, deseen cancelar un viaje, se consideró pertinente que se notifique adecuadamente al usuario que le afecte directamente esta decisión.

Con lo mencionado anteriormente se pueden resaltar los siguientes casos.

Un usuario pasajero puede cancelar el viaje solicitado en cualquier momento, sin importar que el usuario conductor haya aceptado la solicitud de este viaje. Cuando un usuario conductor ha aceptado la solicitud y posteriormente el usuario pasajero realiza la cancelación del viaje, es necesario notificar al usuario conductor.

Si un conductor ha propuesto un viaje, puede cancelar en cualquier momento. En caso de que algún pasajero haya aceptado, éste último será notificado de la cancelación realizada por el conductor.

5. Desarrollo de la aplicación

A continuación, se presenta como fue el desarrollo de la aplicación, dividido por HU, mostrando específicamente el resultado de las diferentes vistas que proporcionan las funcionalidades descritas anteriormente.

En primer lugar, se realizó el desarrollo de la pantalla inicial de la aplicación, desde la cual los diferentes usuarios podrán acceder a las funcionalidades según corresponda.

En la Figura 28, se observa la pantalla de inicio de la aplicación, esta pantalla se muestra únicamente cuando el usuario ingresa a la aplicación. Se muestran diferentes opciones. El botón principal (“Iniciar sesión”) permite a los pasajeros y conductores iniciar sesión en la aplicación por medio de datos biométricos. Si aún los usuarios no tienen cuenta, aparecen dos opciones (“Registrarse como conductor” o “Registrarse como pasajero”) y, por último, para el administrador del sistema, aparece la opción de iniciar sesión como administrador.

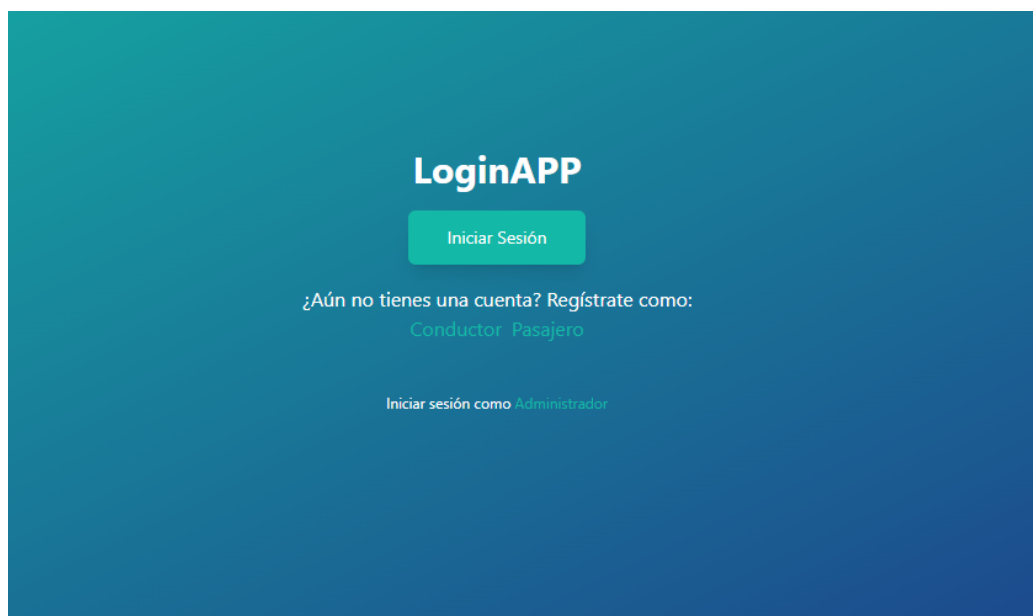


Figura 28. Inicio de la aplicación.

5.1. Desarrollo de Historias de Usuario pasajero

En primer lugar, se desarrollaron las HU del usuario pasajero.

Para un usuario pasajero que va a ingresar a la aplicación por primera vez, se requiere que se registre como usuario inicialmente, presionando en el botón de “pasajero” de la Figura 28. Posteriormente, se redirige al usuario a un formulario mostrado en la Figura 29, en donde el pasajero debe diligenciar cada uno de los campos que se muestran, que son: Nombre, Número de identificación y Correo electrónico.



LoginAPP Registro Pasajero

Nombre
Ingrese su nombre

Número de identificación
Ingrese su número de identificación

Correo electrónico
Ingrese su correo electrónico

Registrarse

Figura 29. Registro de usuario pasajero.

Luego de diligenciar el formulario y presionar en el botón de “Registrarse”, aparece la opción de tomar la foto con la que el usuario ingresaría a la aplicación cada vez que lo requiera, tal como se muestra en la Figura 30.

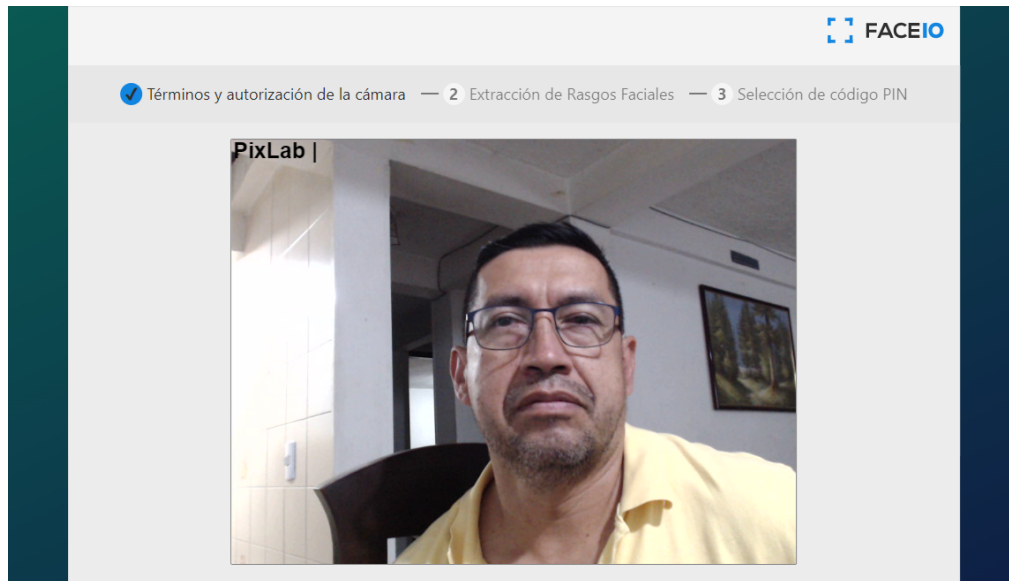


Figura 30. Toma de datos biométricos.

Así mismo, luego de proporcionar los datos biométricos, se solicita un pin, el cual puede ser de 4 a 16 dígitos, el cual también se solicitaría al momento de iniciar sesión. Con el pin proporcionado, finaliza el proceso de registro en la aplicación como usuario pasajero.



Figura 31. PIN solicitado.

Al ser un usuario registrado en la aplicación, ya es posible acceder a las diferentes funcionalidades.

Para iniciar sesión, se presiona el botón de “Iniciar sesión” de la Figura 28. Inmediatamente, la aplicación despliega la cámara, para tomar una captura a la cara de la persona y solicitar el acceso a la aplicación (Figura 32).

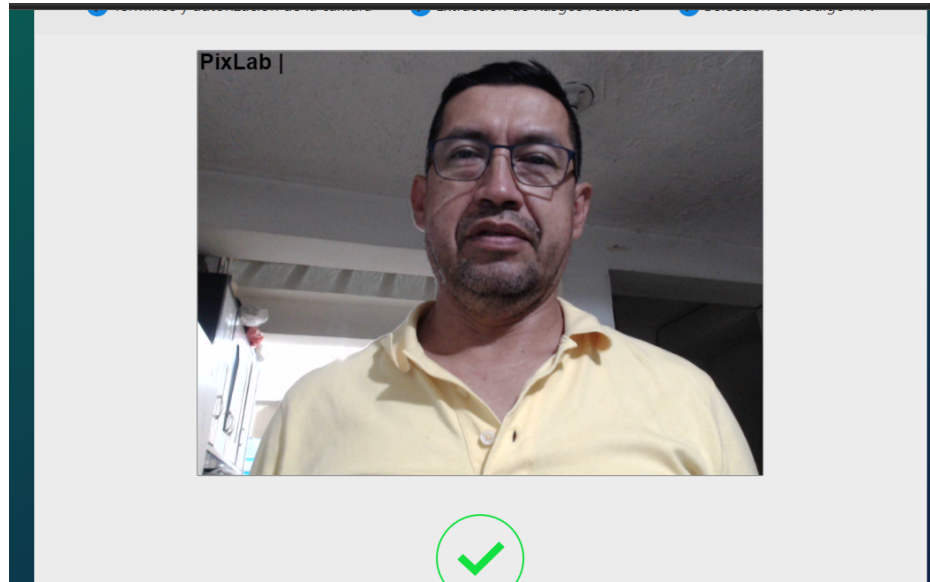


Figura 32. Toma de datos biométricos para iniciar sesión.

Posteriormente, se solicita digitar el pin proporcionado al momento de registrarse con el que finaliza el proceso de autenticación (Figura 33).



Figura 33. Ingreso de Pin.

Al iniciar sesión, se dirige al usuario pasajero a la vista inicial, presentada en la Figura 34, desde donde puede agendar un viaje.

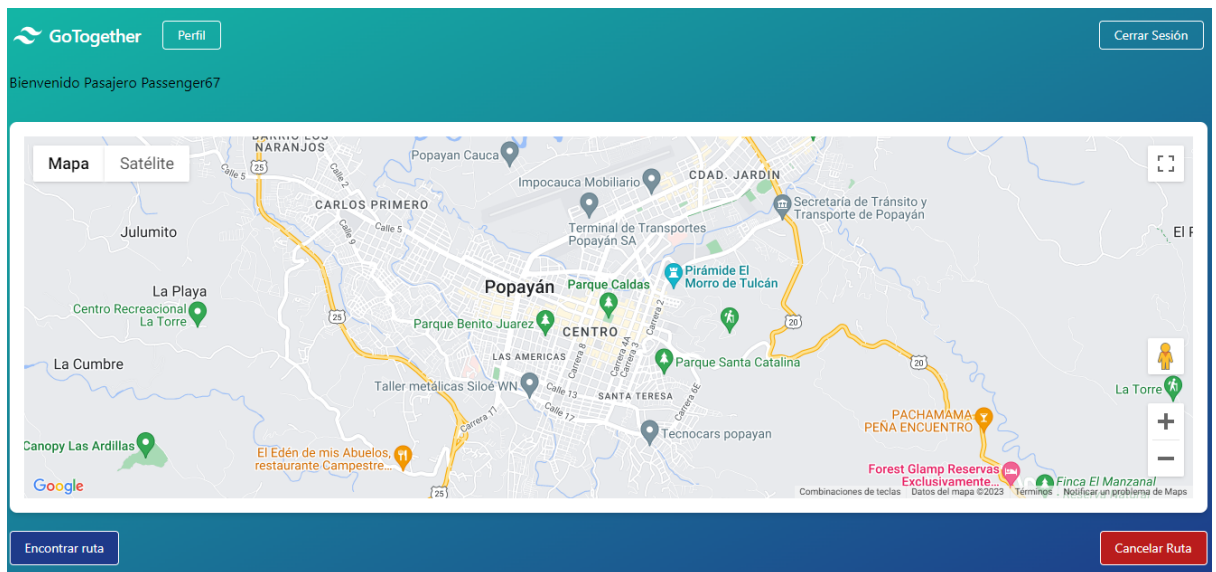


Figura 34. Vista principal de usuario pasajero.

En la parte superior de la Figura 34, se observa un botón denominado “Perfil”, en donde el pasajero puede ver la información que proporcionó al momento de registrarse en la aplicación, es importante aclarar que no puede modificar ninguno de los datos presentados (Figura 35).

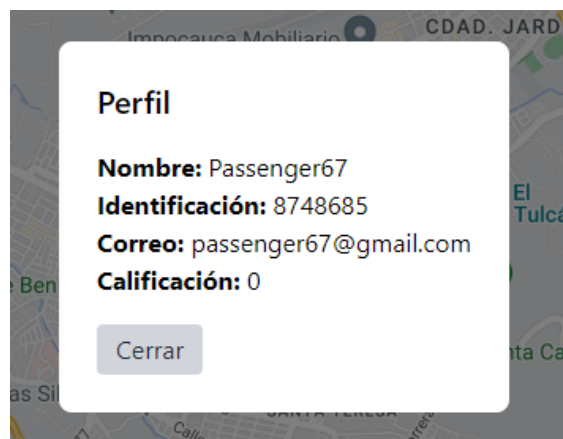


Figura 35. Visualización de perfil pasajero.

En la parte superior derecha se encuentra la opción de “Cerrar sesión”, en donde se cierra la sesión del usuario y se dirige a la vista mostrada en la Figura 28.

Para agendar un viaje, en primer lugar, el usuario pasajero debe seleccionar dos puntos en el mapa presentado en la Figura 28, si selecciona uno o más de dos, se presenta un mensaje indicando que los puntos proporcionados son inválidos ya que se requieren específicamente dos. Al seleccionar los puntos, se irán marcando en el mapa mencionado, tal como se muestra en la Figura 36.

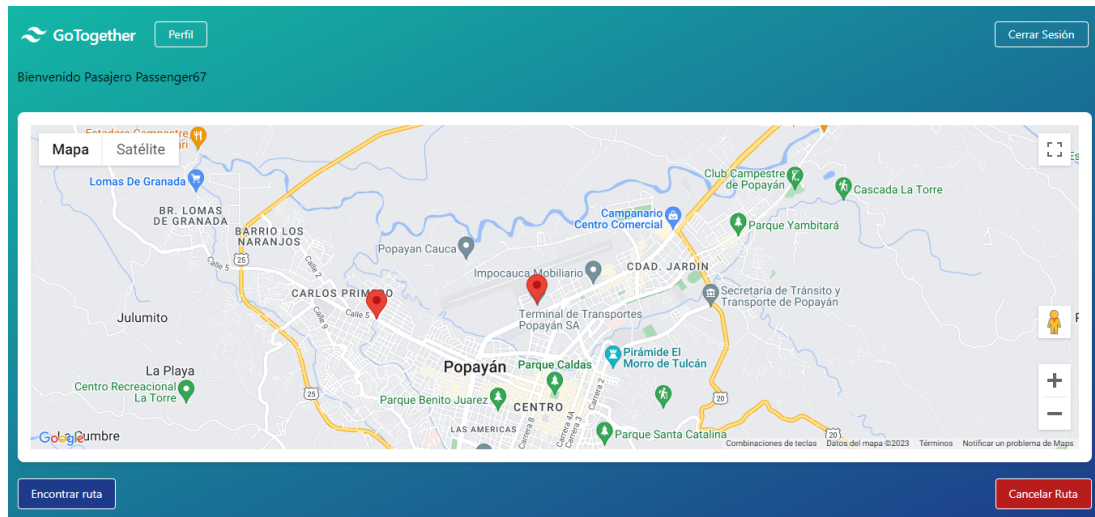


Figura 36. Selección de puntos de inicio y fin del viaje.

Luego de marcar los puntos, se selecciona la opción de “Encontrar ruta”, en donde aparece un formulario pidiendo al usuario indicar la fecha y la hora en la que desea hacer el viaje, por defecto, la fecha y la hora serán las actuales y no se permite seleccionar una fecha u hora anterior.

Indica la fecha y hora de tu ruta

Fecha

18/09/2023

Hora

19:34

Crear Ruta

Figura 37. Selección de fecha y hora del viaje.

Si existen rutas que coinciden con los puntos proporcionados en la Figura 36, aparecen al lado del mapa, tal y como se muestra en la Figura 38. Se considera que una ruta es cercana, si tanto el punto de inicio, como el punto de fin del viaje del conductor se encuentran a menos de 300 metros de la ruta de un conductor; y la hora de inicio del viaje está a menos de diez minutos del inicio de la ruta planteada por el conductor. Si es así, la ruta se presentaría, de lo contrario se descarta automáticamente por el sistema.

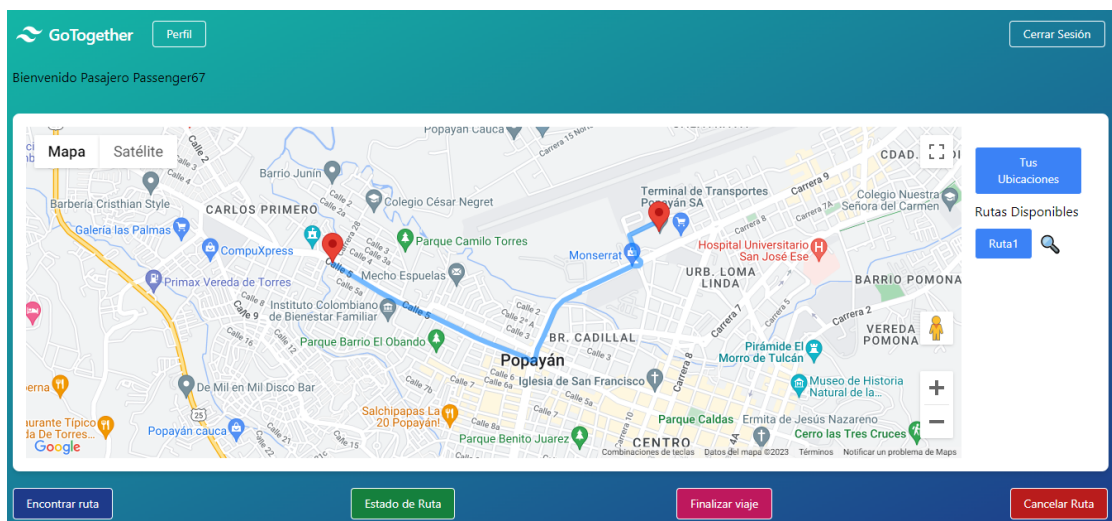


Figura 38. Rutas cercanas

Para poder ver las diferentes rutas, el pasajero puede presionar sobre cualquiera de los botones proporcionados de las posibles rutas. El usuario pasajero también puede observar la información relevante del conductor, como la calificación, el nombre y el modelo del vehículo (Figura 39).

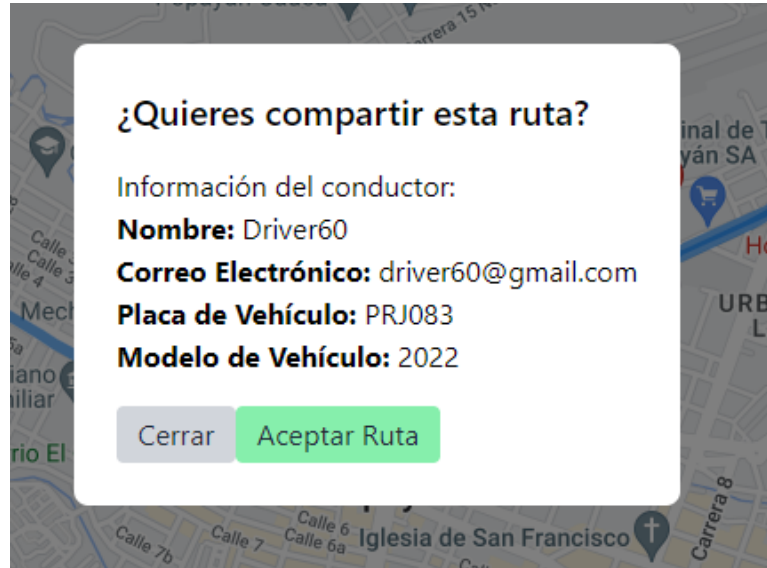


Figura 39. Información del conductor.

Si el usuario pasajero desea postularse a un viaje de un conductor, selecciona el botón “Aceptar Ruta”, el cual se presenta en la Figura 39.

Luego de seleccionar una ruta de un conductor, el pasajero puede ver el estado de la solicitud de viaje en el botón de “Estado de Ruta” de la Figura 38. En la ventana que se genera (Figura 40), se observa si el viaje ya fue aceptado por un conductor, si fue cancelado, rechazado o si aún no responde.

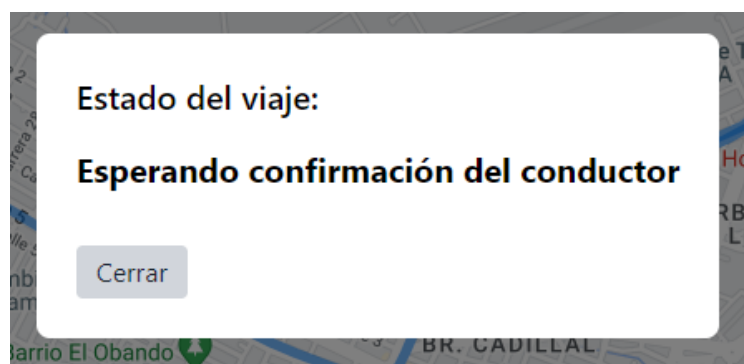


Figura 40. Estado de ruta.

Cuando el pasajero agenda un viaje, se habilitan otras opciones en su pantalla, como las presentadas en la parte inferior de la Figura 38.

Para cancelar un viaje, el pasajero debe presionar en la opción de “Cancelar Ruta” presentada en la Figura 38, en donde se borra los datos del viaje brindados por el usuario y se muestra de nuevo la vista de la Figura 36.

Cuando un usuario finaliza un viaje, puede presionar en la opción de “Finalizar viaje” presentada en la parte inferior de la Figura 38, en donde se puede calificar al conductor de 0 a 5, como se muestra en la Figura 41.

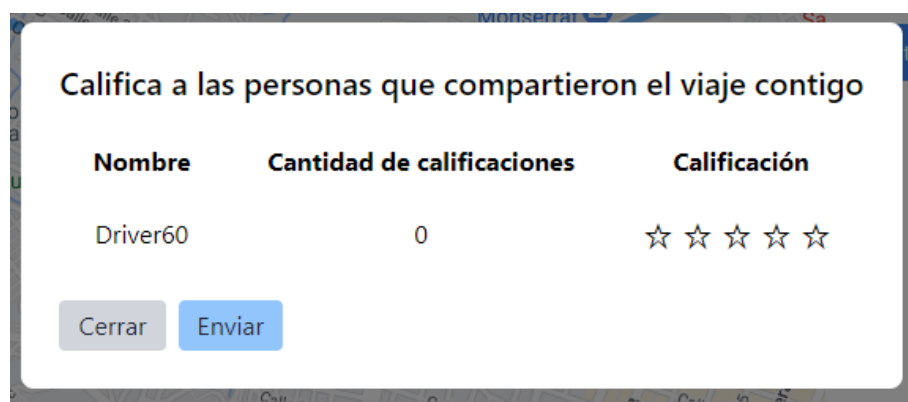


Figura 41. Calificación a conductor.

5.2. Desarrollo de Historias de Usuario conductor

De forma similar al usuario pasajero, a partir de la Figura 28, el usuario conductor puede elegir registrarse en la aplicación, presionando en el botón “Conductor”, en donde aparece un formulario que el usuario debe diligenciar la información que se presenta en la Figura 42.

LoginAPP Registro Conductor

Nombre
Ingrese su nombre

Número de identificación
Ingrese su número de indentificación

Correo electrónico
Ingrese su correo electrónico

Placa
Ingrese la placa de su vehículo

Modelo
Ingrese el modelo de su vehículo

Registrarse

Figura 42. Registro en la aplicación como conductor.

Cuando el usuario diligencia los campos solicitados, aparece en la pantalla la cámara abierta, para proceder con una captura de imagen del usuario y guardar los datos biométricos (Figura 43). Posteriormente, se solicita un pin de 4 a 16 dígitos con el que el usuario accedería a la aplicación (Figura 44).

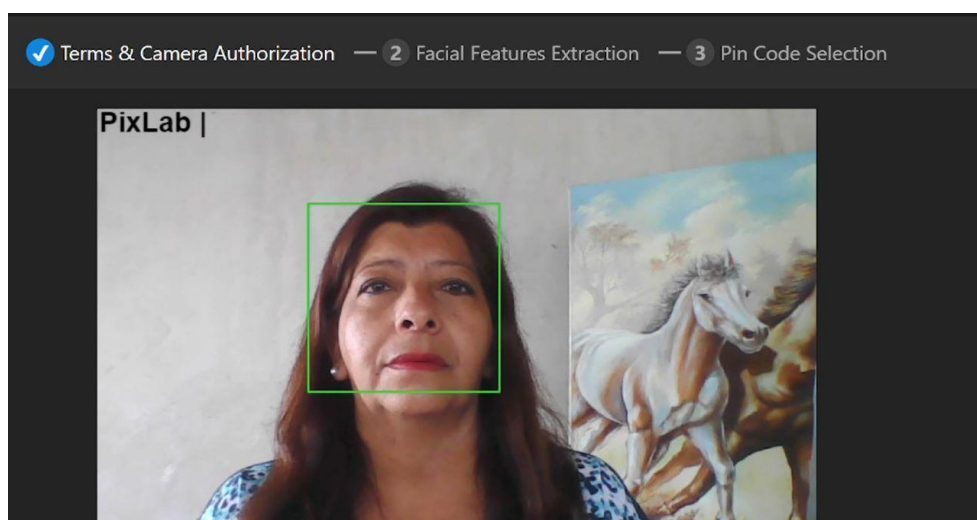


Figura 43. Registro de datos biométricos del usuario conductor.



Figura 44. Solicitud de Pin al conductor.

Una vez registrado el usuario conductor, se regresa a la vista de la Figura 28, al igual que el usuario pasajero, presionando en el botón “Iniciar sesión”, el usuario conductor puede acceder a la funcionalidad asignada. Al presionar dicho botón la aplicación solicita la toma de datos faciales (Figura 45) y el pin proporcionado anteriormente en el proceso de registro (Figura 46).

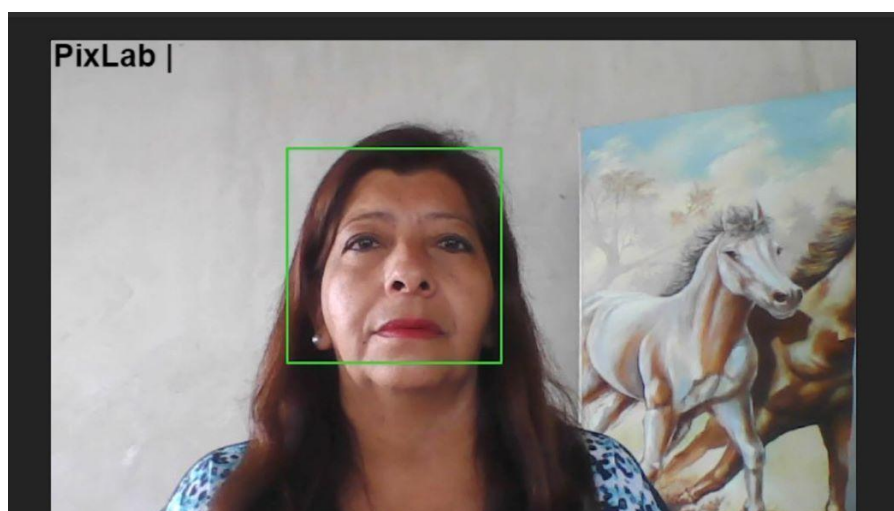


Figura 45. Autenticación biométrica usuario conductor.



Figura 46. Solicitud de Pin de usuario conductor.

Al iniciar sesión correctamente, se dirige al conductor a la pantalla principal, desde donde va a poder realizar diferentes funcionalidades (Figura 47).

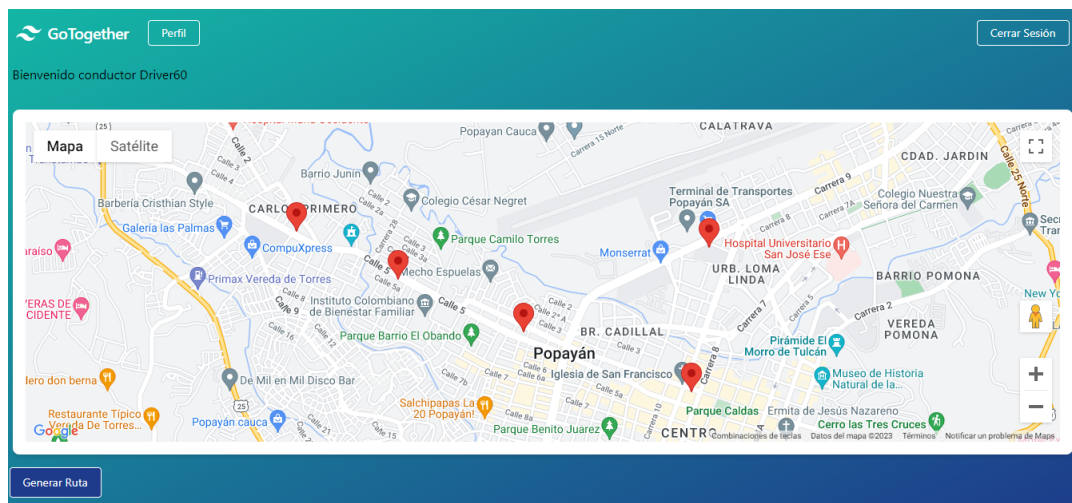


Figura 47. Vista principal usuario conductor.

El usuario conductor, también puede observar los datos proporcionados al momento del registro en la sección de “Perfil” que se encuentra en la parte superior derecha de la Figura 47.

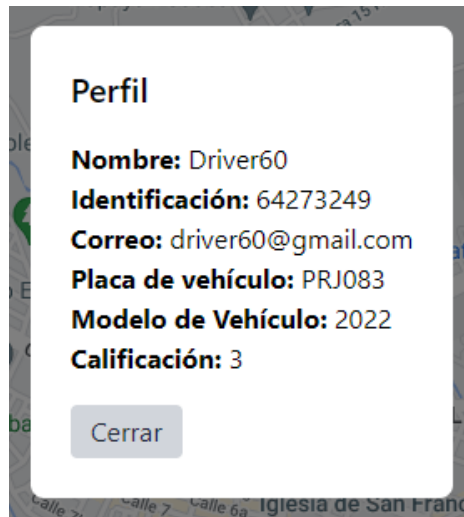


Figura 48. Visualización de perfil de conductor.

Como se puede observar en la Figura 47, el usuario conductor puede seleccionar varios puntos en el mapa, los cuales indican lugares específicos por donde pasa su ruta, tal y como se muestra en la Figura 49

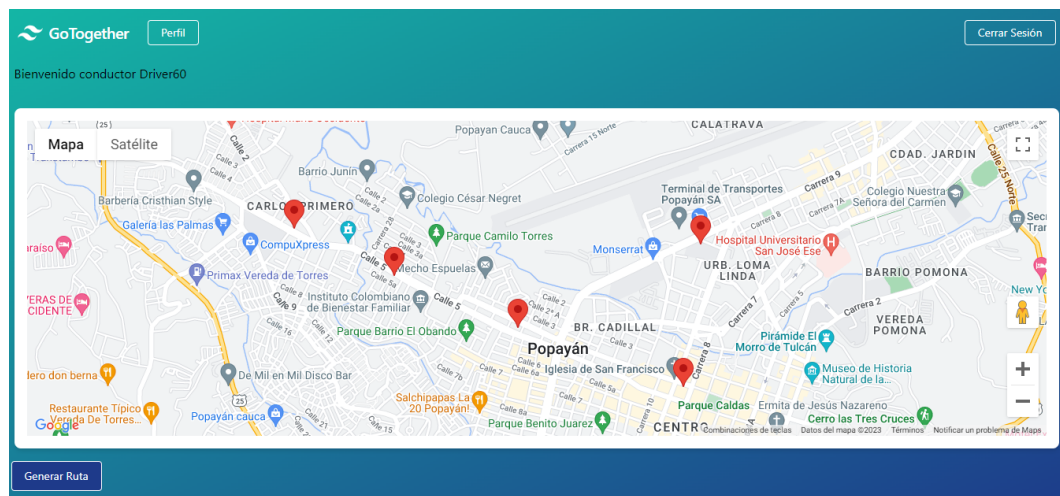


Figura 49. Selección de puntos de ruta.

Al terminar de seleccionar todos los puntos, el usuario puede presionar el botón de “Generar ruta”, en donde aparece un formulario con la hora y fecha en la que se quiere realizar la ruta, como se muestra en la Figura 50, por defecto, la fecha y la hora serán las actuales y no se permite seleccionar una fecha u hora anterior.

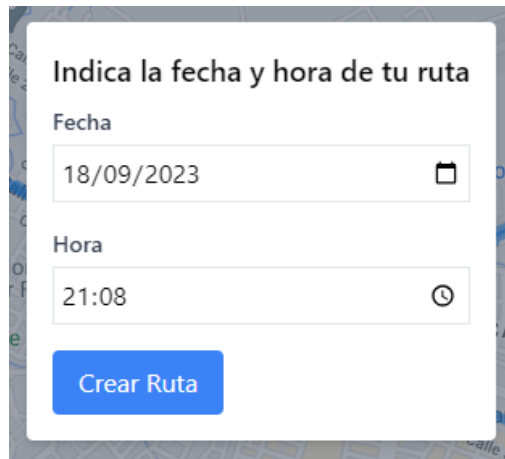


Figura 50. Selección de hora y fecha de ruta.

Cuando el usuario conductor crea una ruta, aparecen otras opciones de acuerdo a su viaje, las cuales se presentan en la Figura 51. Es importante anotar que, un usuario conductor sólo puede crear una ruta a la vez, si desea crear una ruta nueva, debe seleccionar la opción “Cancelar Ruta” e iniciar el proceso nuevamente.

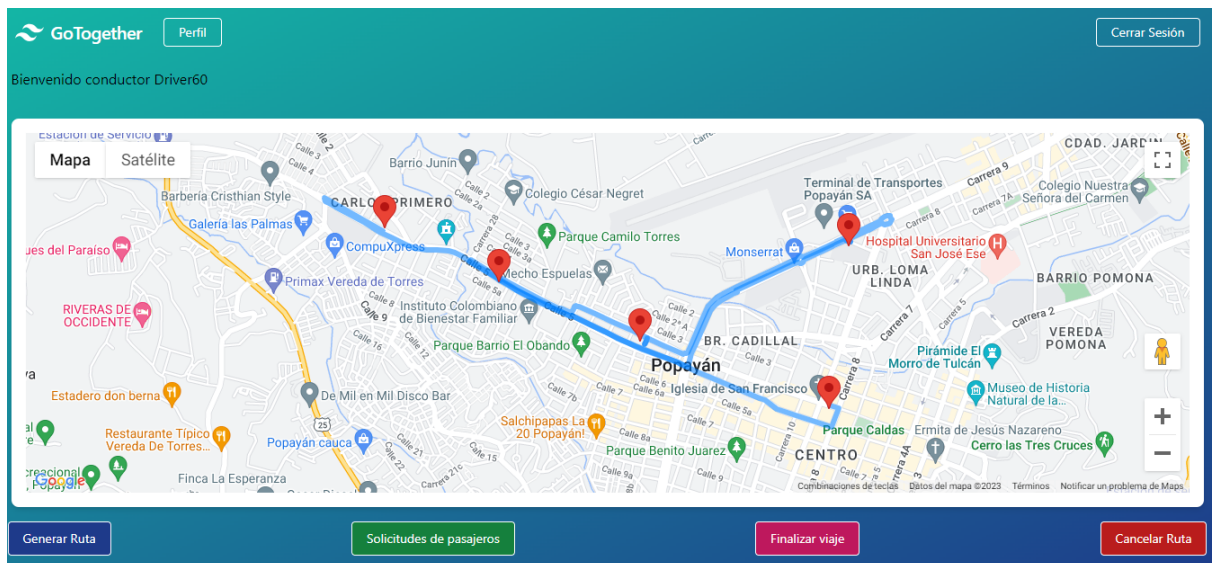


Figura 51. Vista de conductor después de agregar una ruta.

El usuario puede cancelar una ruta en el momento que desee, presionando el botón respectivo presentado en la Figura 51. Si se tienen pasajeros con los cuales va a compartir su viaje, se les notificará que el conductor ha cancelado la ruta. Esta opción redirige la vista del conductor a la mostrada en la Figura 47, eliminando la ruta propuesta.

En la opción de “Solicitudes de pasajeros”, el usuario puede ver los pasajeros que se han postulado a su ruta y aceptarlos o rechazarlos según decida, como se muestra en la Figura 52.

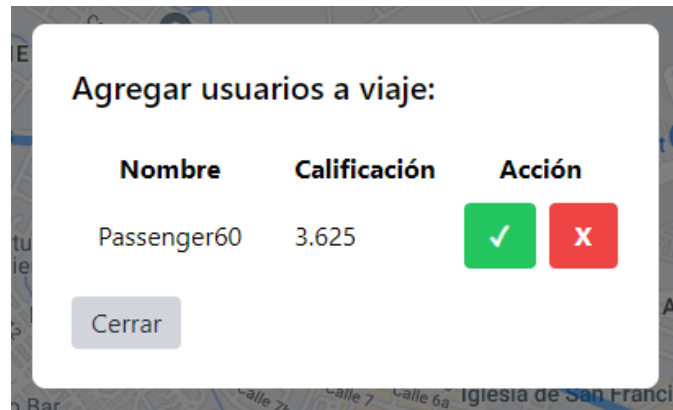


Figura 52. Gestión de solicitudes de pasajeros.

Para finalizar un viaje, el conductor puede presionar en el botón “Finalizar viaje” presentado en la Figura 51, en donde se mostrará los pasajeros con los que compartió viaje y le permite darles una calificación (Figura 53).

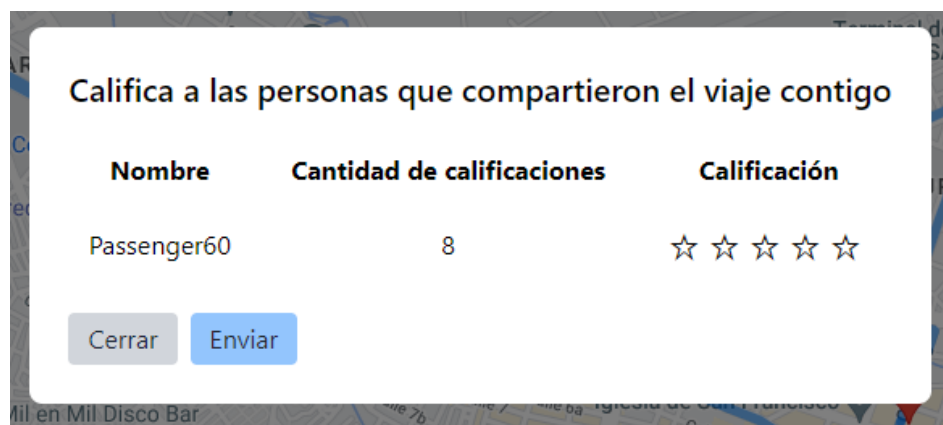


Figura 53. Calificación a pasajeros.

Finalmente, el conductor puede cerrar sesión, presionando el botón denominado “cerrar sesión”, ubicado en la parte superior derecha de la interfaz presentada en la Figura 47.

5.3. Desarrollo de Historias de Usuario Administrador

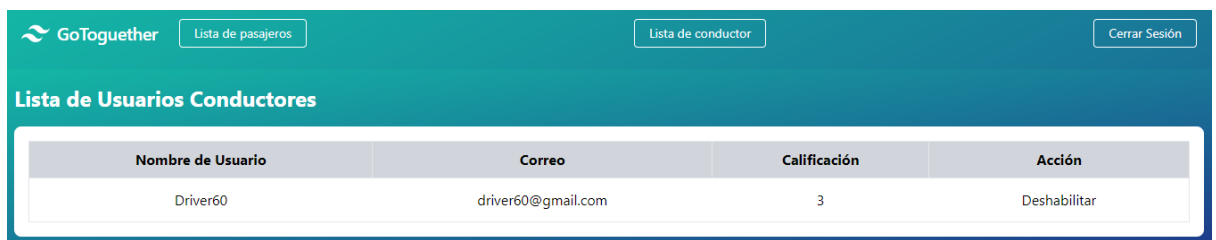
La aplicación cuenta con un usuario administrador, el cual no se registra en la aplicación, ya que este ya cuenta con un usuario y contraseña brindados por la aplicación. En primer lugar, para iniciar sesión como administrador, este debe proporcionar, en la interfaz de la Figura 54, tanto el usuario como la contraseña asignados.



El formulario de ingreso de administrador tiene un fondo azul oscuro con el título "Ingreso de administrador" en blanco. Incluye dos campos de entrada de texto blancos con bordes redondeados: el primero para el nombre con el placeholder "Ingrese su nombre" y el segundo para la contraseña con el placeholder "Ingrese la contraseña". Debajo de los campos hay un botón rectangular de color verde claro con el texto "Iniciar sesión" en blanco.

Figura 54. Ingreso usuario Administrador.

Al iniciar sesión correctamente como administrador, se muestra en primer lugar, la lista de usuarios conductores con su información correspondiente y la opción de habilitar o deshabilitar usuarios, según se requiera (Figura 55).



La interfaz muestra el logo "GoTogether" y tres botones de navegación: "Lista de pasajeros", "Lista de conductor" y "Cerrar Sesión". El título principal es "Lista de Usuarios Conductores".

Nombre de Usuario	Correo	Calificación	Acción
Driver60	driver60@gmail.com	3	Deshabilitar

Figura 55. Información de conductores.

En la parte superior de la Figura 56 se observa la opción de "Lista de pasajeros", en donde se observa la lista de pasajeros junto con la opción de habilitar o deshabilitar al usuario.

Nombre de Usuario	Correo	Calificación	Acción
Passenger60	passenger60@gmail.com	3.555556	Deshabilitar
Passenger67	passenger67@gmail.com	2	Habilitar

Figura 56. Información de pasajeros.

Por último, el usuario administrador puede cerrar sesión cuando lo requiera, presionando respectivo, el cual se encarga de cerrar la sesión actual del usuario y redirigirlo de nuevo a la Figura 28.

5.4 Integración con módulo de autenticación Blockchain.

Para integrar el módulo de autenticación *Blockchain* a la aplicación “*frontend*” implementada utilizando la librería *React* para desarrollar las HU presentadas para los 3 tipos de usuario, fue necesario en primera medida instalar la herramienta *Hardhat*, la cual permite instalar un nodo de la *Blockchain* de *Ethereum* de manera local para ello se utilizaron los siguientes comandos:

```
npx hardhat init
npm install --save-dev "hardhat@^2.18.0"
"@nomicfoundation/hardhat-toolbox@^3.0.0."
```

El primer comando inicia un proyecto *Hardhat*, se seleccionan los parámetros del proyecto con un menú de consola como se muestra en la Figura 57. Se nota también, como la aplicación de *Hardhat*, se ejecuta usando el puerto 8545, el cual sirve para interactuar con el proyecto desde la aplicación desarrollada con *React*.

```

PS C:\tesis\Eth\contracts> npx hardhat init
888 888 888 888 888
888 888 888 888 888
888 888 888 888 888
8888888888 8888b. 888d888 .d88888 88888b. 8888b. 888888
888 888 "88b 888P" d88" 888 888 "88b "88b 888
888 888 .d888888 888 888 888 888 .d888888 888
888 888 888 888 888 Y88b 888 888 888 888 Y88b.
888 888 "Y888888 888 "Y88888 888 888 "Y888888 "Y888

Welcome to Hardhat v2.18.0

✓ What do you want to do? · Create a TypeScript project
✓ Hardhat project root: · C:\tesis\Eth\contracts
✓ Do you want to add a .gitignore? (Y/n) · y

You need to install these dependencies to run the sample project:
  npm install --save-dev "hardhat@^2.18.0" "@nomicfoundation/hardhat-toolbox@^3.0.0"

Project created

See the README.md file for some example tasks you can run

Give Hardhat a star on Github if you're enjoying it!

https://github.com/NomicFoundation/hardhat
PS C:\tesis\Eth\contracts> npx hardhat node
Started HTTP and WebSocket JSON-RPC server at http://127.0.0.1:8545/

```

Figura 57. Creación de proyecto en *Hardhat*

El segundo comando agrega las herramientas necesarias para que *Hardhat* pueda compilar contratos inteligentes e instale las dependencias necesarias.

Una vez creado el entorno de desarrollo, se procede a agregar el contrato inteligente a la carpeta *Contracts*, la versión final del contrato inteligente se muestra en las Figuras 58, 59 y 60.

```

// SPDX-License-Identifier: UNLICENSED
pragma solidity ^0.8.9;

contract Lock {
    address public owner; // Dirección del propietario del contrato
    mapping(string => bool) public stringExists; // Mapeo para verificar si una cadena existe
    string[] public stringList; // Lista de cadenas almacenadas
    uint public unlockTime; // Momento en el que se puede acceder a las funciones

    event StringAdded(string value); // Evento emitido al agregar una cadena de Ids
    event StringRemoved(string value); // Evento emitido al eliminar una cadena de Ids

    // Constructor que recibe el tiempo de desbloqueo y el monto bloqueado
    constructor(uint _unlockTime) payable {
        // Verifica que el tiempo de desbloqueo esté en el futuro
        require(
            block.timestamp < _unlockTime,
            "El tiempo de desbloqueo debe estar en el futuro"
        );
        owner = msg.sender; // El creador del contrato se convierte en el propietario
        unlockTime = _unlockTime; // Configura el tiempo de desbloqueo
    }

    // Modificador para restringir el acceso solo al propietario
    modifier onlyOwner() {
        require(msg.sender == owner, "Solo el propietario puede acceder");
        _;
    }
}

```

Figura 58. Contrato Inteligente 1.

En la Figura 58 se especifica la licencia del contrato, la versión de *Solidity* y se define la dirección pública del contrato inteligente, que es la dirección donde es desplegado, en este caso de manera local. Además, se define el método constructor que configura una variable de tiempo *_unlockTime* que le indica al contrato inteligente en qué momento puede acceder a las funciones del mismo.

```

// Función para agregar un dato biométrico a la lista
function addString(string memory value) public onlyOwner {
    // Verifica que la cadena no exista en la colección
    require(!stringExists[value], "Ya ha sido agregado ese Id biométrico en la Blockchain");
    stringList.push(value); // Agrega la cadena a la lista
    stringExists[value] = true; // Marca la cadena como existente
    emit StringAdded(value); // Emite el evento de cadena agregada
}

```

Figura 59. Contrato Inteligente 2

En la Figura 59 se presenta la definición de la función más importante que es denominada *addString*, la cual recibe el Id biométrico en formato *String* y verifica si

este ya existe previamente, antes de ejecutar el proceso de almacenamiento en el nodo local de la *Blockchain* de *Ethereum*, este método es ejecutado cada vez que un usuario pasajero o conductor se registra en la aplicación.

```
// Función para verificar si una cadena existe en la colección
function checkString(string memory value) public view returns (bool) {
    return stringExists[value];
}

// Función para obtener la lista de cadenas almacenadas
function getStringList() public view returns (string[] memory) {
    return stringlist;
}
```

Figura 60. Contrato Inteligente 3

En la Figura 60 se presentan dos funciones, la función *checkString* es la más importante porque verifica la existencia de un dato biométrico en la *Blockchain* de *Ethereum*. Lo anterior posibilita que al momento en que un usuario intenta ingresar a la aplicación, se llame este método y se verifique que este usuario efectivamente se ha registrado con anterioridad y que sus datos biométricos están almacenados en la *Blockchain* de *Ethereum*.

Luego se procede a compilar el contrato inteligente con el siguiente comando:

```
npx hardhat compile
```

Si este proceso es satisfactorio, se puede proceder a levantar el nodo local con el comando *npx hardhat node*, lo cual arroja una lista de direcciones que pueden ser usadas para desplegar contratos inteligentes. Además, despliega el contrato inteligente en la dirección que fue configurado en las propiedades del código *deploy.ts*.

Es importante destacar que el archivo *deploy.ts* es un componente esencial en este proceso. Aunque es generado automáticamente al crear el proyecto de *Hardhat*, puede ser modificado para adaptarse a necesidades específicas. En este caso, se definió la dirección en la cual se desplegó el contrato inteligente.

Algunas de estas direcciones y el resultado del despliegue del contrato son presentados en la Figura 61.

```
Account #17: 0xbDA5747bFD65F08deb54cb465eB87D40e51B197E (10000 ETH)
Private Key: 0x689af8efa8c651a91ad287602527f3af2fe9f6501a7ac4b061667b5a93e037fd

Account #18: 0xdD2FD4581271e230360230F9337D5c0430Bf44C0 (10000 ETH)
Private Key: 0xde9be858da4a475276426320d5e9262ecfc3ba460bfac56360bfa6c4c28b4ee0

Account #19: 0x8626f6940E2eb28930eFb4CeF49B2d1F2C9C1199 (10000 ETH)
Private Key: 0xdf57089febbac7ba0bc227dafbffa9fc08a93fdc68e1e42411a14efcf23656e

WARNING: These accounts, and their private keys, are publicly known.
Any funds sent to them on Mainnet or any other live network WILL BE LOST.

eth_accounts
hardhat_metadata (20)
eth_blockNumber
eth_getBlockByNumber
eth_feeHistory
eth_sendTransaction
  Contract deployment: <UnrecognizedContract>
  Contract address: 0x5f8db2315678afecb367f032d93f642f64180aa3
  Transaction: 0xa66aa853e41a8c034ca4b4f02b04ec78fd58a168179a149168410ec96fb76dd9
  From: 0xf39fd6e51aad88f6f4ce6ab8827279cfff92266
  Value: 0.001 ETH
  Gas used: 326112 of 30000000
  Block #1: 0x113ef46561feba2744555512fb8234c0f94adb3e30f4423ff9fbd4cc9607feb5

eth_getTransactionByHash
eth_getTransactionReceipt
eth_blockNumber
```

Figura 61. Despliegue de contrato inteligente.

Una vez se realizan los pasos anteriormente mencionados, es necesario instalar la librería *ethers.js*, para poder interactuar con el contrato inteligente desde la aplicación elaborada con *React*. Para esto se ejecuta el siguiente comando:

```
npm install ethers
```

Luego de ello se importa la librería al código fuente con la siguiente línea:

```
import { ethers } from "ethers".
```

Finalmente se modifica el código de los métodos de registro y autenticación para la integración con *Blockchain*. En la Figura 62 se presenta el ejemplo para la HU de registro de un usuario, donde se observa cómo se crea una instancia de la librería

ether.js para el puerto 8545, que es el puerto en el cual se está ejecutando la aplicación de nodo *Blockchain*.

Luego se especifica la dirección del contrato inteligente y la variable en formato JSON ABI (*Application Binary Interface*, es decir, Interfaz de Aplicación Binaria) que resume en formato JSON cuáles son las funciones que contiene el contrato inteligente. Finalmente se llama de manera asíncrona el método *addString* presente en el contrato inteligente que almacena los datos biométricos codificados en formato String en el nodo *Blockchain* de *Ethereum*

```
// Crear una instancia del proveedor Ethereum y el signatario
const provider = new ethers.providers.JsonRpcProvider("http://127.0.0.1:8545");
const signer = provider.getSigner();

// Dirección y ABI de contrato inteligente Ethereum
const contractAddress = "0x5fbd2315678afecb367f032d93f642f64180aa3"; // dirección de contrato inteligente
const contractAbi = ABIjson; // Reemplaza con el ABI de tu contrato

// Crear una instancia de tu contrato inteligente
const contract = new ethers.Contract(contractAddress, contractAbi, signer);

// Llama al método addString del contrato para almacenar el dato en la cadena de bloques
const tx = await contract.addString(response.facialId);

// Espera a que se confirme la transacción
await tx.wait();
```

Figura 62. Llamado al contrato inteligente desde la aplicación utilizando *React*

En conjunto, se ejecutan tres aplicaciones de manera simultánea: la aplicación que simula un nodo *Blockchain* de *Ethereum* de *Hardhat*, que se encarga de desplegar el contrato inteligente y facilita la interacción con el mismo; la aplicación de *backend* de *Spring Boot*, que gestiona las operaciones en la base de datos relacional *SQL*; y la aplicación frontend de *React*. Este ecosistema de aplicaciones en paralelo conforma la solución completa de la aplicación web de *carpooling* con autenticación *Blockchain*.

6. Pruebas de funcionamiento

A continuación, se detallan las pruebas realizadas para comprobar el funcionamiento de las HU, de los usuarios conductor, pasajero y administrador.

6.1. Pruebas de HU de Pasajero

Nombre del tipo de prueba	Resultado esperado	Resultado obtenido
Registro exitoso	Mensaje de registro correcto. Redireccionamiento a la vista de inicio de sesión.	Se obtuvo el mensaje de registro correcto y posteriormente se hizo el redireccionamiento a la vista de inicio de sesión.
Registro no exitoso	Mensajes de alerta para completar campos de formulario de registro.	Se obtuvieron los mensajes de alerta bajo cada campo del formulario que era obligatorio diligenciar.
	Si ya se ha registrado un pasajero con los mismos datos, aparece una alerta y se redirecciona al formulario de registro de pasajero.	Fue obtenida una alerta en donde se indicaba que el usuario ya existía y posteriormente se redireccionó nuevamente al formulario de registro.
Inicio de sesión exitoso	Mensaje de alerta indicando la correcta identificación.	Se realizó un correcto inicio de sesión, apareció un mensaje indicando

	Redireccionamiento a la ventana de bienvenida.	que se realizó una identificación adecuada y apareció la ventana de bienvenida.
Inicio de sesión no exitoso	Mensaje de inicio de sesión fallido y redireccionamiento al inicio de sesión de nuevo.	Se recibió un mensaje de alerta indicando que no fue posible iniciar sesión y posteriormente la aplicación redireccionó al inicio de sesión, para realizar el ingreso nuevamente.
Visualización de perfil exitoso	Al presionar el botón "Perfil" se visualiza una ventana que muestra los datos del pasajero.	Se presionó el botón "Perfil" y se visualizó la ventana que mostraba correctamente los datos del pasajero.
Visualización de perfil no exitoso	Al presionar el botón "Perfil" y habiendo eliminado manualmente de la base de datos la sesión se espera que en consola aparezca un mensaje indicando que no se pudo obtener el usuario. Esto sólo se presenta si existe algún problema de comunicación entre la aplicación <i>frontend</i> y la	Se presionó el botón "Perfil" y se visualizó un mensaje de consola que dice "Error al obtener user"

	aplicación <i>backend</i>	
Cerrar sesión exitoso	Se cierra la sesión actual del conductor y se realiza el redireccionamiento a la vista principal de la aplicación.	Se cerró correctamente la sesión y se redirigió al usuario a la vista principal de la aplicación, fuera de las funcionalidades del conductor.
Cerrar sesión no exitoso	Al presionar el botón “Cerrar Sesión” habiendo eliminado manualmente de la base de datos la sesión se espera que la aplicación no muestre ningún mensaje y no se dirija a otra vista. Esto sólo se presenta si existe algún problema de comunicación entre la aplicación <i>frontend</i> y la aplicación <i>backend</i> .	No se observó ningún cambio en la pantalla, la vista se conservó en la especificada por el conductor.
Encontrar una ruta exitoso	Se seleccionan dos puntos para inicio y final de viaje, se especifica la fecha y la hora y se visualizan las rutas que puedan cumplir con las especificaciones ingresadas por el pasajero	Se seleccionaron dos puntos en el mapa, se especificó fecha y hora y se visualizaron los botones que muestran las rutas disponibles

Encontrar una ruta no exitoso	Se seleccionan más de dos puntos en el mapa, se espera que no permita la selección de fecha y hora y se borren los puntos marcados en el mapa para que puedan seleccionarse de nuevo.	Se seleccionaron tres puntos en el mapa y se presentó un mensaje que indica que se deben seleccionar exactamente dos puntos en el mapa, se borraron los puntos marcados y se mantuvo en la vista del pasajero.
	Se seleccionan dos puntos en el mapa, sin embargo, no hay rutas que cumplan con las especificaciones dadas por el pasajero. Se espera que se le informe al usuario que no hay rutas disponibles y que borre las marcas en el mapa, manteniéndose en la vista de pasajero	Se seleccionaron dos puntos en el mapa y se especificó la fecha y la hora, seguido se mostró un mensaje que indica que no había rutas disponibles, se borraron los puntos marcados y se mantuvo en la vista del pasajero.
Seleccionar una ruta exitoso	Dado que se visualizan las rutas disponibles, cuando el pasajero selecciona su ruta, puede visualizar los datos del conductor y presionar el botón aceptar, con lo cual, ya no es posible visualizar las rutas disponibles y es posible visualizar los botones Estado de ruta y	Se visualizaron las rutas disponibles, se presionó el ícono de la lupa y se visualizaron los datos del conductor que corresponde a la ruta seleccionada, se presionó “aceptar”, se dejaron de visualizar las rutas encontradas y se visualizaron los botones

	Finalizar viaje.	Estado de ruta y Finalizar viaje.
Seleccionar una ruta no exitoso	Se intenta seleccionar una ruta y se elimina la sesión de manera manual por base de datos, se espera que la consola muestre un mensaje indicando que no ha sido posible seleccionar una ruta. Esto sólo se presenta si existe algún problema de comunicación entre la aplicación <i>frontend</i> y la aplicación <i>backend</i> .	Se visualizaron las rutas disponibles, se presionó el ícono de la lupa y se visualizaron los datos del conductor que corresponde a la ruta seleccionada, se presionó en "aceptar" y no se cerró la ventana. Se visualiza un mensaje en consola que indica que no se pudo seleccionar la ruta.
	Dado que el pasajero visualiza los datos del conductor y decide no seleccionar dicha ruta, presiona en el botón cancelar y vuelve a visualizar las rutas disponibles	Se visualizaron los datos de un conductor, se presionó en cancelar y se presentó la vista de pasajero, en la cual persisten las rutas disponibles.
Revisar el estado de una ruta exitoso	Dado que el pasajero ha seleccionado un viaje de manera exitosa, puede presionar el botón "Estado de ruta", lo que muestra al usuario el estado de su	Se presionó el botón "Estado de ruta" y pudo visualizarse una ventana con el estado de la solicitud de viaje.

	solicitud al conductor.	
Revisar el estado de una ruta no exitoso	Dado que el pasajero ha seleccionado un viaje de manera exitosa, selecciona el botón “Estado de ruta”, lo que muestra al usuario el estado de su solicitud al conductor.	Se presionó el botón “Estado de ruta”, se visualizó una ventana que no tiene información. Se visualizó en consola un mensaje que indica que hay un error al obtener el estado de la ruta.
Cancelar ruta exitoso	Dado que el pasajero decide cancelar el viaje, presiona el botón “Cancelar ruta”, se dejan de observar las rutas disponibles y se ocultan los botones “Estado de ruta” y “Finalizar viaje”, además el mapa se limpia.	Se presionó el botón “Cancelar ruta”, con lo cual desaparecieron las marcas en el mapa y no se visualizaron los botones “Estado de ruta” y “Finalizar viaje”.
Cancelar ruta no exitoso	El pasajero decide cancelar el viaje, para ello selecciona el botón “Cancelar ruta” y se elimina la sesión de manera manual por base de datos, se espera que se siga observando la ruta en pantalla del conductor. Esto sólo se presenta si existe algún problema de	Se siguió observando la ruta en la pantalla del conductor.

	comunicación entre la aplicación <i>frontend</i> y la aplicación <i>backend</i> .	
Finalizar viaje exitoso	El pasajero puede presionar el botón “Finalizar viaje” cuando la ruta se ha completado, con lo cual aparece una ventana emergente donde es posible calificar al conductor seleccionando una nota de cero a cinco. Al presionar cerrar la vista de pasajero vuelve al estado de inicio de sesión como pasajero exitoso.	Se presionó el botón “Finalizar viaje”, se visualizó una ventana donde se seleccionó la calificación del conductor y se dio clic en Cerrar. Luego de ello se observó la vista del pasajero en su estado de inicio de sesión como pasajero exitoso.
Finalizar viaje no exitoso	El pasajero puede presionar el botón “Finalizar viaje” cuando la ruta se ha completado, con lo cual aparece una ventana emergente donde es posible calificar al conductor seleccionando una nota de cero a cinco. Al presionar cerrar la vista de pasajero vuelve al estado de inicio de sesión como pasajero exitoso.	No se visualizó correctamente la ventana emergente de calificación al conductor y al presionar cerrar se oculta la ventana sin haber cambios en la vista de pasajero. Se visualiza un mensaje en consola que indica que hubo un error al calificar al conductor.

Tabla 1. Pruebas a HU de pasajero

A continuación, se detallan los casos de prueba de la Tabla 1.

Registro como pasajero exitoso

Cuando un usuario pasajero logró registrarse correctamente en la aplicación, apareció un mensaje indicando que sus datos se almacenaron correctamente, acto seguido se dirigió el usuario a la pantalla principal de la aplicación, lo que se observa en la Figura 63.

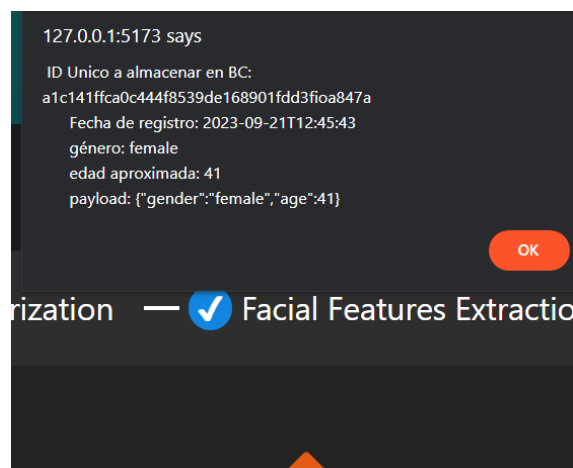


Figura 63. Registro de pasajero exitoso.

Registro como pasajero no exitoso

Para probar el caso en el que un pasajero no puede registrarse, se consideraron dos casos.

El primer caso, a la izquierda de la Figura 64, sucedió cuando el usuario no diligenció todos los campos del formulario, e intentó registrarse de esta manera. En esta situación, se confirmó que el resultado arrojado por la aplicación fue señalar los campos que no se ha diligenciado, marcándolos y agregado un mensaje bajo cada campo.

El segundo caso, a la derecha de la Figura 64, sucedió cuando un usuario ya está registrado en la plataforma. El resultado obtenido fue un mensaje indicando al usuario que ya se había registrado con el mismo email.



Figura 64. Registro de pasajero no exitoso

Inicio de sesión como pasajero exitoso

Cuando un usuario pasajero ingresó correctamente a la aplicación, es decir que su autenticación en el sistema fue correcta, antes de redirigir al usuario a la pantalla desde donde podrá agendar viajes, apareció un mensaje indicando que su autenticación fue exitosa. Inmediatamente después de presionar el botón OK se redirigió a la vista de pasajero, como se muestra en las Figuras 65 y 66.

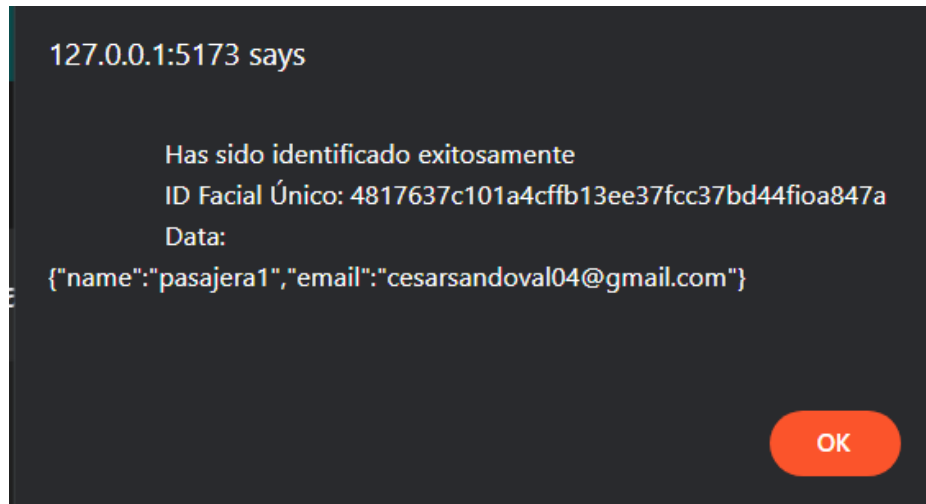


Figura 65. Identificación de pasajero exitosa

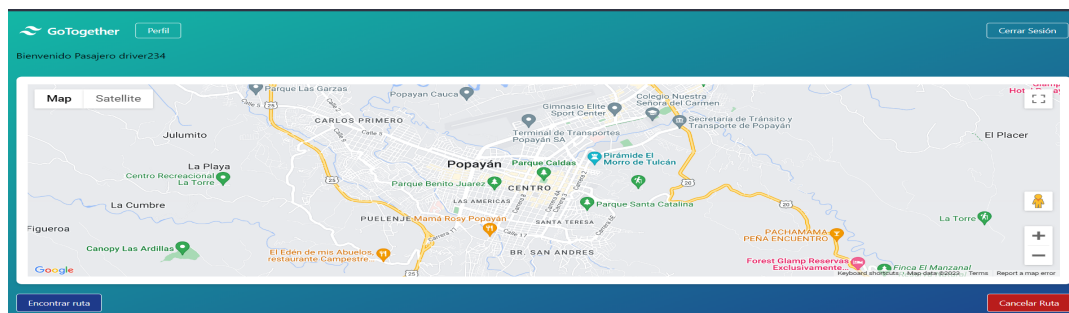


Figura 66. Identificación de pasajero exitosa, vista de pasajero

Inicio de sesión no exitoso

Cuando la aplicación no detectó un rostro registrado al momento de ingresar, o el pin del usuario fue erróneo, aparece el mensaje presentado en la Figura 67.

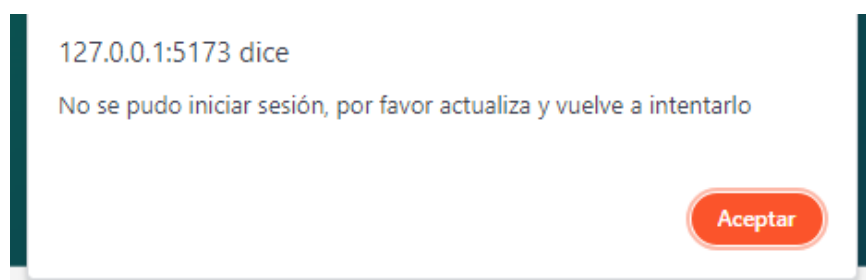


Figura 67. Inicio de sesión no exitoso.

Visualización de perfil exitoso.

Cuando el pasajero indicó que quería ver el perfil, presionando en el botón "Perfil", apareció la información brindada por el usuario al momento del registro (Figura 68).

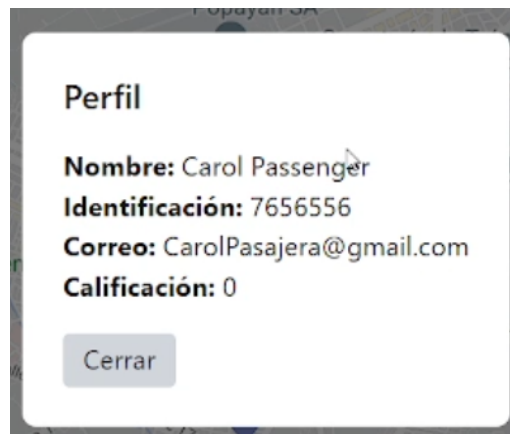


Figura 68. Visualización de perfil de pasajero exitoso.

Visualización de perfil no exitoso

Cuando se presentó algún problema en la aplicación, ya fuese en la comunicación con el "backend", o debido a que la sesión del usuario ya no exista, o a que el usuario haya sido deshabilitado, no aparece ninguna información brindada, los campos aparecen vacíos (tal como se presenta en la Figura 69).

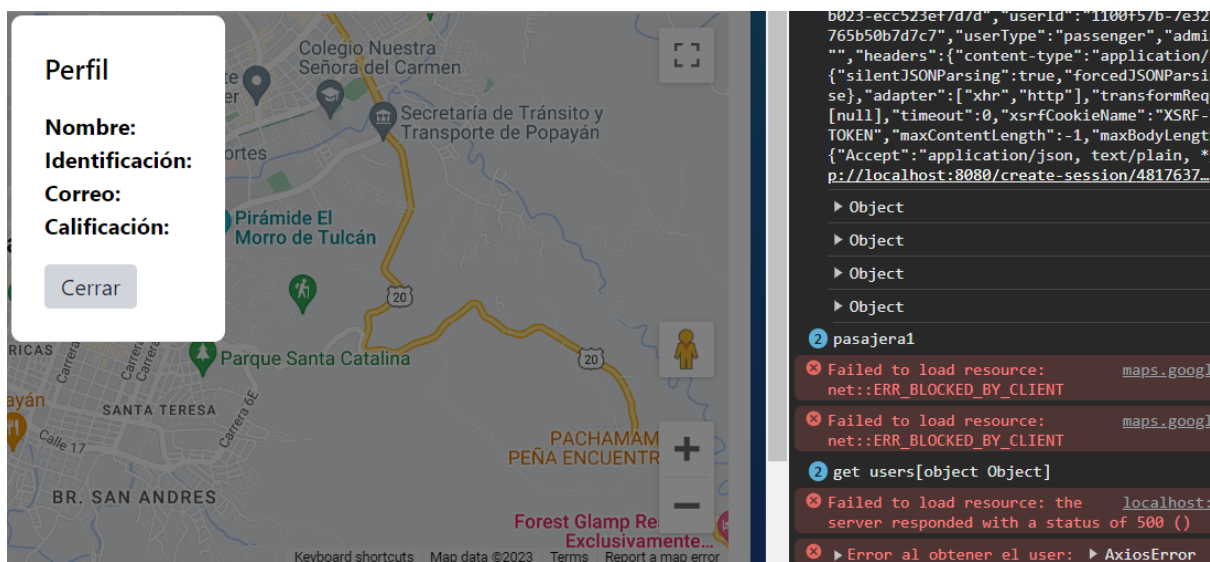


Figura 69. Visualización de perfil de pasajero no exitoso.

Cerrar sesión exitosa.

Al presionar el botón “cerrar sesión”, se redirigió el usuario a la vista principal de la aplicación, como se presenta en la Figura 70.

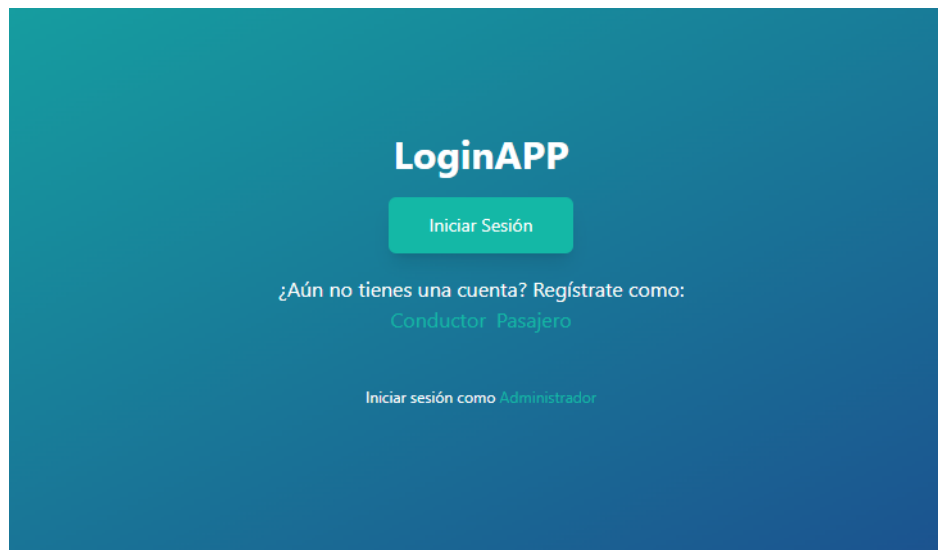


Figura 70. Cerrar sesión exitosa.

Cerrar sesión no exitosa.

En caso de algún error ocurrido en la aplicación cuando el usuario quería cerrar sesión, no se cierra la vista del usuario, lo que indica que el pasajero aún tiene la sesión activa, como se presenta en la Figura 71.

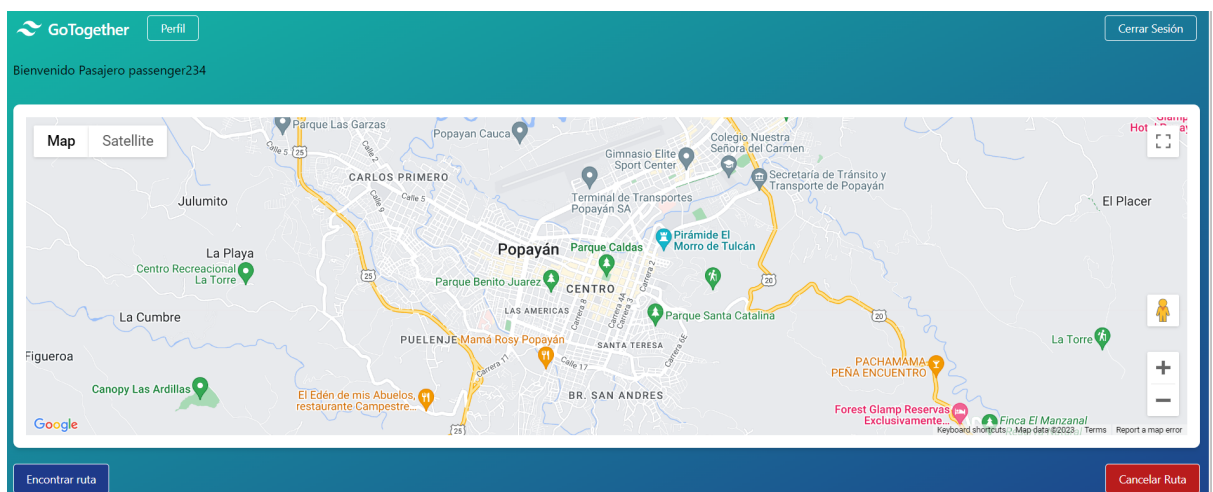


Figura 71. Cerrar sesión no exitoso.

Encontrar una ruta exitoso.

Como pasajero, se marcaron exactamente dos puntos en el mapa, se accedió a la opción “Encontrar ruta”, se visualizó una ventana emergente que indicó al usuario

que debía ingresar la fecha y hora de su viaje (Figura 72). Enseguida, el usuario presionó en el botón “Crear Ruta”, con lo cual se visualizaron las rutas disponibles para el pasajero, con diferentes opciones como visualizar sus puntos inicial y final en el botón “Tus ubicaciones”, visualizar las rutas en el mapa con los botones “Ruta1” y “Ruta2” y visualizar los datos de los conductores creadores de las respectivas rutas el botón que tiene un ícono lupa a la derecha de cada ruta. Además, se visualizaron los botones Estado de ruta y Finalizar viaje (Figura 73).



Indica la fecha y hora de tu ruta

Fecha

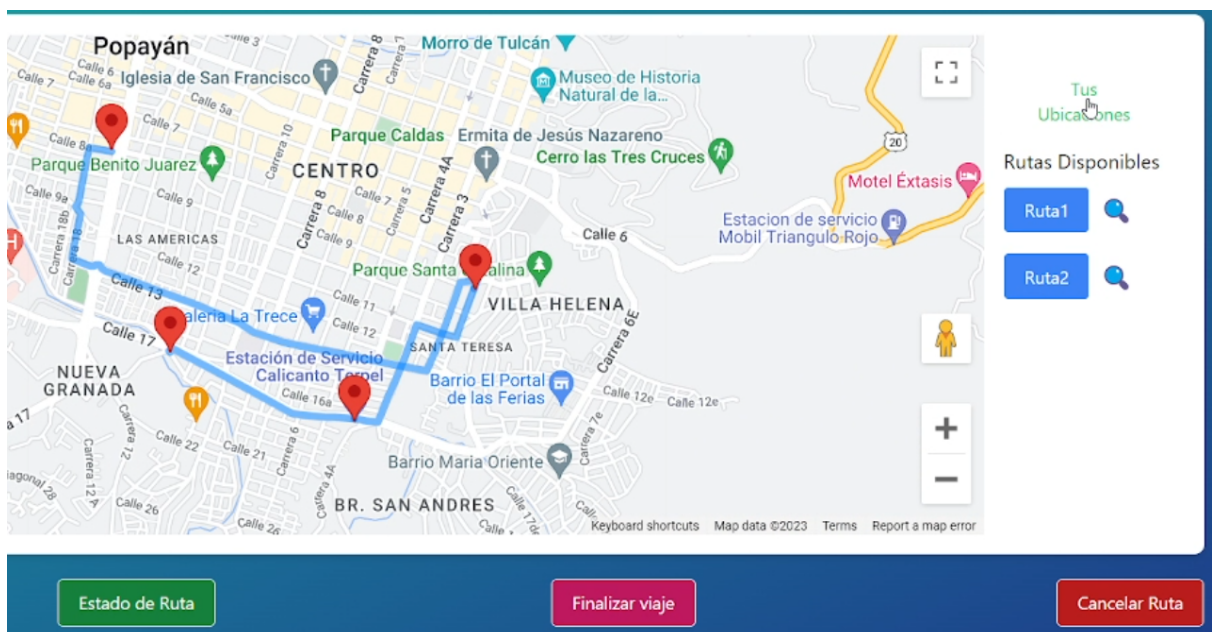
18/09/2023

Hora

19:34

Crear Ruta

Figura 72. Selección de fecha y hora.



Popayán

Morro de Tulcán

Iglesia de San Francisco

Parque Benito Juárez

CENTRO

Parque Caldas

Ermita de Jesús Nazareno

Cerro las Tres Cruces

Museo de Historia Natural de la...

Estación de servicio Mobil Triangulo Rojo

Motel Éxtasis

VILLA HELENA

Parque Santa Catalina

Estación de Servicio Calicanto Terminal

Barrio El Portal de las Ferias

SANTA TERESA

NUEVA GRANADA

Barrio Maria Oriente

BR. SAN ANDRES

Tus Ubicaciones

Rutas Disponibles

Ruta1

Ruta2

Estado de Ruta

Finalizar viaje

Cancelar Ruta

Figura 73. Encontrar rutas exitoso

Encontrar una ruta no exitoso.

Se probaron dos casos; primero se marcaron exactamente tres puntos en el mapa, la aplicación presentó una ventana emergente indicando que debían seleccionarse exactamente dos puntos, como se muestra en la Figura 74. Para el segundo caso se marcaron dos puntos en el mapa, pero sabiendo que no hay rutas disponibles que estén cerca de esos puntos, la aplicación mostró un mensaje indicando que no hay rutas disponibles, como se muestra en la Figura 75.

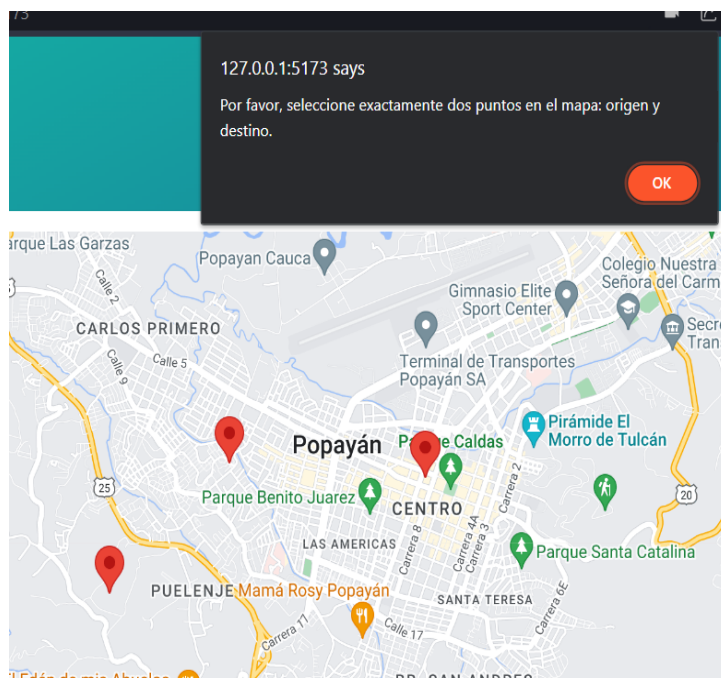


Figura 74. Encontrar rutas no exitoso 1.

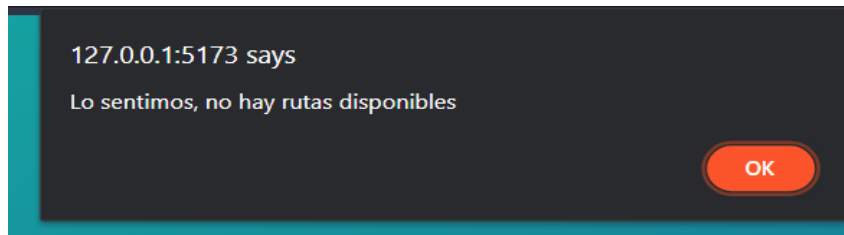


Figura 75. Encontrar rutas no exitoso 2.

Seleccionar una ruta exitoso

Se seleccionó el botón de la “lupa” y se presionó el botón “Aceptar ruta”, la aplicación mostró una ventana emergente indicando que la ruta ha sido seleccionada correctamente como se muestra en la Figura 76.

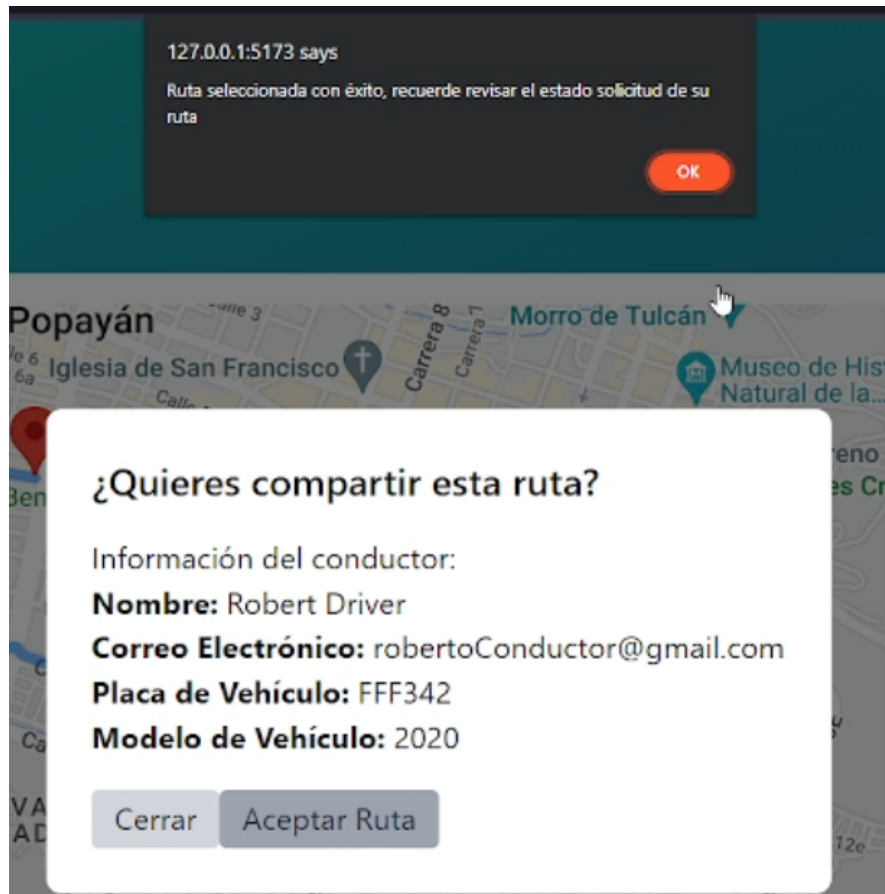


Figura 76. Seleccionar ruta exitoso.

Seleccionar una ruta no exitoso

Se selecciona el botón de la “lupa” y se presiona la opción “Aceptar ruta”, la aplicación presentó un mensaje indicando que hubo un error al seleccionar la ruta, como se muestra en la Figura 77.



Figura 77. Seleccionar ruta no exitoso.

Estado de ruta exitoso

Habiendo seleccionado una ruta exitosamente, se seleccionó la opción “Estado de ruta”, la aplicación presentó una ventana emergente indicando el estado de la solicitud del pasajero. En la Figura 78 (izquierda) se muestra el mensaje obtenido cuando el conductor aún no había aceptado la solicitud y en la Figura 78 (derecha) se muestra el mensaje obtenido cuando el viaje ya había sido aceptado por el conductor.

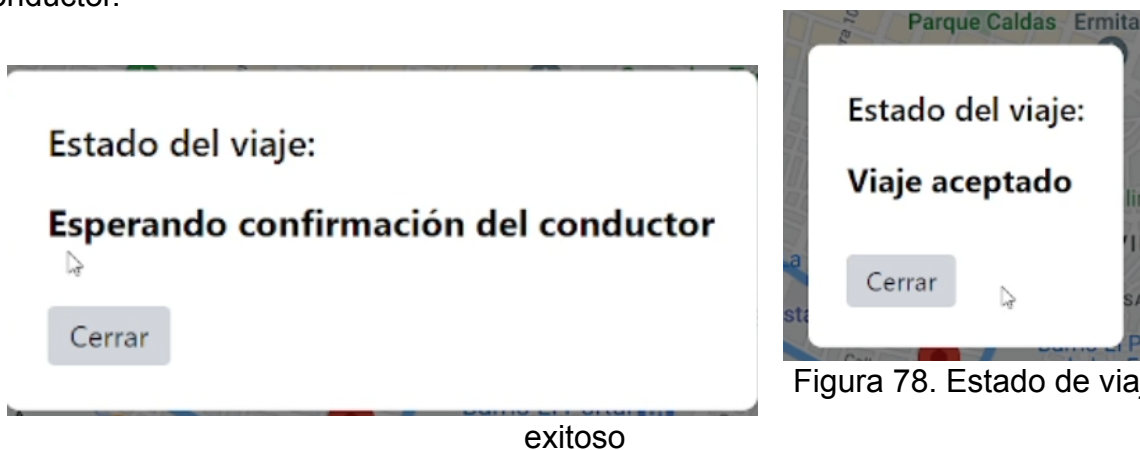


Figura 78. Estado de viaje

Estado de ruta no exitoso

Habiendo seleccionado una ruta exitosamente, pero eliminando en la base de datos dicha ruta para que el llamado al “*backend*” falle, se seleccionó la opción “Estado de ruta”, la aplicación muestra una ventana emergente indicando que hubo un error.

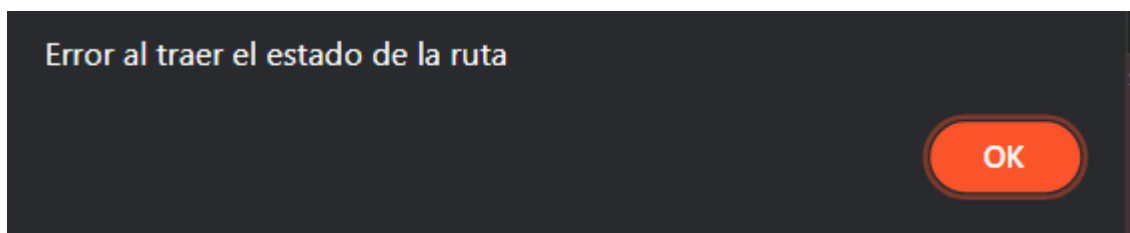


Figura 79. Estado de viaje no exitoso

Finalizar ruta exitoso

Se seleccionó la opción “Finalizar ruta”, la aplicación mostró una ventana emergente, que permite calificar al conductor, se seleccionaron la cantidad de estrellas. Se presionó luego el botón “Enviar” lo que llevó al usuario a la vista inicial de pasajero, es decir, ya no presentó las opciones de visualizar rutas, ni los botones “Estado de ruta” y “Finalizar ruta”.

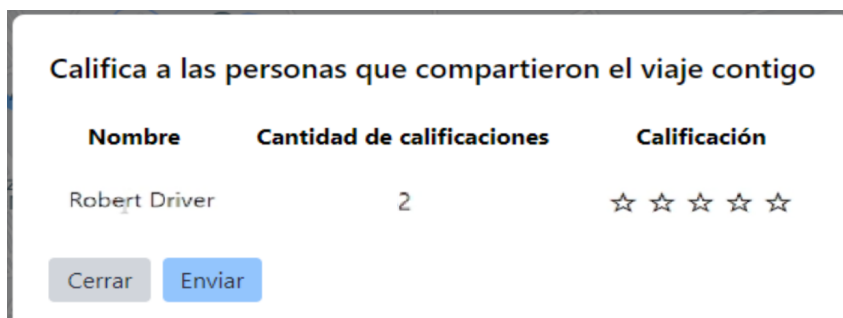


Figura 80. Finalizar viaje exitoso, calificar al conductor.

Finalizar ruta no exitoso

Se seleccionó la opción “Finalizar ruta”, la aplicación presentó la tabla de personas que compartieron el viaje vacía, cuando se cerró la ventana emergente se mantuvo en la vista de pasajero, sin que desaparecieran las opciones de visualizar rutas, ni los botones “Estado de ruta” y “Finalizar ruta”.

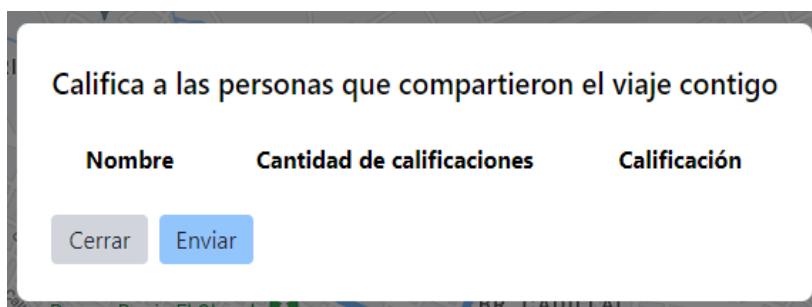


Figura 81. Finalizar un viaje no exitoso.

6.2. Pruebas de HU de Conductor

Nombre del tipo de prueba	Resultado esperado	Resultado obtenido
Registro exitoso como conductor	Mensaje de registro correcto. Redireccionamiento a la vista de inicio de sesión.	Se obtuvo el mensaje de registro correcto y posteriormente se redireccionó a la vista de inicio de sesión.

Registro no exitoso como conductor	Mensajes de alerta para completar campos de formulario de registro.	Se obtuvieron los mensajes de alerta bajo cada campo del formulario que era obligatorio diligenciar.
	Si ya se ha registrado una persona con los mismos datos, aparece una alerta y se redirecciona al formulario de registro.	Fue obtenida una alerta en donde se indicaba que el usuario ya existe y posteriormente se redireccionó nuevamente al formulario de registro.
Inicio de sesión exitoso	Mensaje de alerta indicando la correcta identificación. Redireccionamiento a la ventana de bienvenida.	Se realizó un correcto inicio de sesión, apareció un mensaje indicando que se efectuó una identificación adecuada y apareció la ventana de bienvenida.
Inicio de sesión no exitoso	Mensaje de inicio de sesión fallido y redireccionamiento al inicio de sesión de nuevo.	Se recibió un mensaje de alerta indicando que no se pudo iniciar sesión y posteriormente la aplicación redireccionó al inicio de sesión, para realizar el ingreso nuevamente.
Visualización de perfil exitoso	Se observa correctamente la información del usuario	La información del usuario se observó de manera satisfactoria.

<p>Visualización de perfil no exitoso</p>	<p>Se selecciona la opción “Perfil” y habiendo eliminado la sesión de manera manual por base de datos. Se observan los campos de información del usuario vacíos. Esto sólo se presenta si existe algún problema de comunicación entre la aplicación <i>frontend</i> y la aplicación <i>backend</i>.</p>	<p>No se observó ninguna información del usuario.</p>
<p>Cerrar sesión exitoso</p>	<p>Se cierra la sesión actual del conductor y se realiza el redireccionamiento a la vista principal de la aplicación.</p>	<p>Se cerró correctamente la sesión y se redirigió al usuario a la vista principal de la aplicación, fuera de las funcionalidades del conductor.</p>
<p>Cerrar sesión no exitoso</p>	<p>Se selecciona la opción “Cerrar Sesión” y habiendo eliminado la sesión de manera manual por base de datos. No se dirige al conductor a la pantalla principal de la aplicación. Esto sólo se presenta si existe algún problema de comunicación entre la aplicación <i>frontend</i> y la aplicación <i>backend</i>.</p>	<p>No se observó ningún cambio en la pantalla, la vista se conservó en la especificada por el conductor.</p>

Selección de puntos de ruta exitoso	Se dibuja la ruta que pasa por los puntos del mapa seleccionados del conductor e inmediatamente se abre un formulario.	Se dibujó en el mapa la ruta del conductor y posteriormente se observó en la pantalla un formulario para escoger fecha y hora de la ruta.
Selección de puntos de ruta no exitoso	Alerta indicando que los puntos proporcionados fueron insuficientes.	Se observó una alerta en la pantalla indicando que se debe seleccionar dos o más puntos.
Selección de fecha y hora del viaje exitoso.	Se cierra el formulario y la ruta del conductor se observa en el mapa	Se cerró el formulario y se observó la ruta del conductor en el mapa.
Selección de fecha y hora del viaje no exitoso.	Aparece una alerta en la pantalla del usuario, que indica que los datos ingresados fueron inválidos	Apareció un mensaje, indicando que se debe proporcionar una fecha y hora válidos, se mantuvo abierto el formulario.
Cancelar ruta exitoso	Se deja de observar la ruta brindada por el conductor en el mapa junto con los botones relacionados a acciones de rutas.	La ruta, dibujada en el mapa anteriormente, desapareció luego de presionar en el botón de Cancelar ruta y los botones de cancelar ruta, finalizar viaje y solicitudes de pasajeros no aparecieron.
Cancelar ruta no exitoso	Se selecciona la opción "Cancelar Ruta" y habiendo eliminado la	Se siguió observando la ruta en la pantalla del conductor.

	<p>sesión de manera manual por base de datos. La ruta se sigue observando en la pantalla del conductor. Esto sólo se presenta si existe algún problema de comunicación entre la aplicación <i>frontend</i> y la aplicación <i>backend</i>.</p>	
Calificación pasajeros exitoso	<p>Al presionar en el botón de finalizar ruta, se obtienen todos los pasajeros para poder calificar</p>	<p>Cuando se finalizó el viaje, apareció una lista con los pasajeros y permitió calificarlos. Al cerrar el formulario, la ruta realizada desapareció del mapa principal.</p>
Calificación pasajeros no exitoso	<p>La lista de pasajeros en la ventana emergente aparece Se selecciona la opción "Finalizar ruta" y habiendo eliminado el usuario a calificar de manera manual por base de datos. vacía y al cerrarla, se sigue observando la ruta en el mapa. Esto sólo se presenta si existe algún problema de comunicación entre la</p>	<p>La lista de pasajeros obtenida fue vacía, aun cuando había pasajeros asociados a la ruta, al cerrar la ventana, la ruta se siguió observando en el mapa.</p>

	aplicación <i>frontend</i> y la aplicación <i>backend</i> .	
--	---	--

Tabla 2. Pruebas HU de conductor

6.2.1. Resultados obtenidos

Registro como conductor exitoso

Cuando el usuario conductor logró registrarse correctamente en la aplicación, apareció un mensaje indicando que sus datos se almacenaron correctamente (Figura 82), acto seguido se dirigió la vista a la pantalla principal de la aplicación.

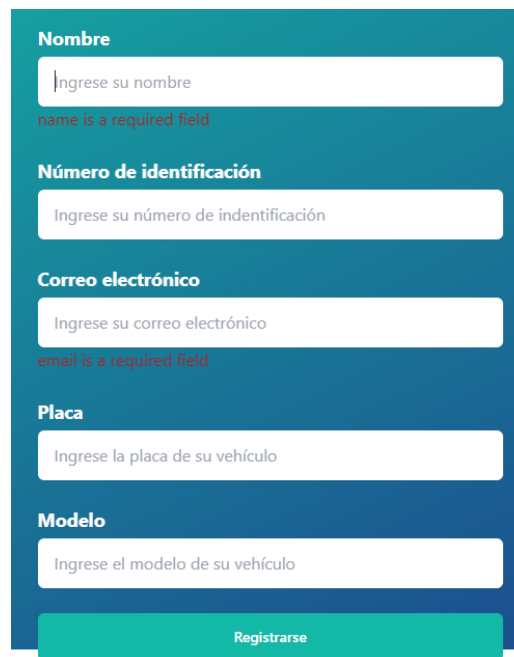


Figura 82. Registro de pasajero exitoso.

Registro de conductor no exitoso

Para probar la funcionalidad de un conductor que no puede registrarse, se consideraron dos casos.

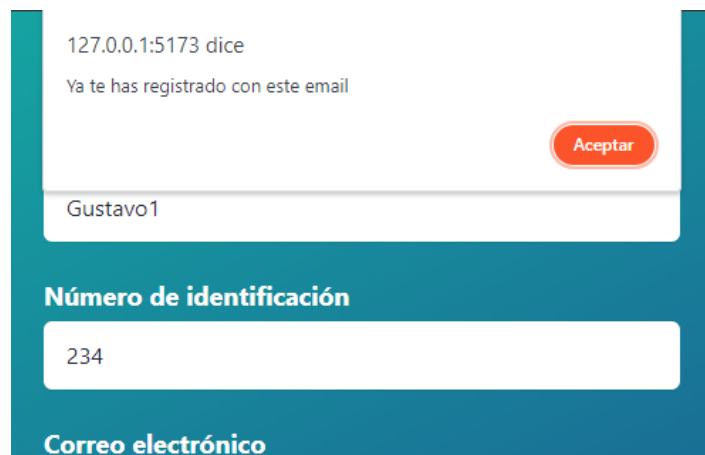
En el primer caso el usuario conductor no diligenció todos los campos del formulario, e intentó registrarse de esta manera. En esta situación, se confirmó que el resultado arrojado por la aplicación fue señalar los campos que no se han diligenciado, marcándolos y agregando un mensaje bajo cada campo, como se presenta en la Figura 83.



The image shows a registration form with a teal background. It contains five input fields, each with a label above it and a validation message below it. The labels are: 'Nombre', 'Número de identificación', 'Correo electrónico', 'Placa', and 'Modelo'. The validation messages are: 'name is a required field', 'email is a required field', and 'name is a required field' (repeated for the 'Placa' field). At the bottom of the form is a teal button labeled 'Registrarse'.

Figura 83. Campos requeridos.

En el segundo caso el usuario conductor ya estaba registrado en la plataforma. El resultado obtenido fue un mensaje indicando al usuario que ya se había registrado, como se presenta en la Figura 84.



The image shows a confirmation message on a teal background. At the top, it says '127.0.0.1:5173 dice' and 'Ya te has registrado con este email'. Below this is a white box containing the name 'Gustavo1'. To the right of this box is a red button labeled 'Aceptar'. Below the white box is another input field labeled 'Número de identificación' containing the number '234'. At the bottom, the label 'Correo electrónico' is visible.

Figura 84. Datos ya registrados.

Inicio de sesión exitoso

Cuando un usuario ingresó correctamente a la aplicación, es decir que su autenticación en el sistema fue correcta, antes de redirigir al usuario a la pantalla desde donde podrá agendar viajes, apareció un mensaje indicando que su autenticación fue exitosa, tal como se presenta en la Figura 85.

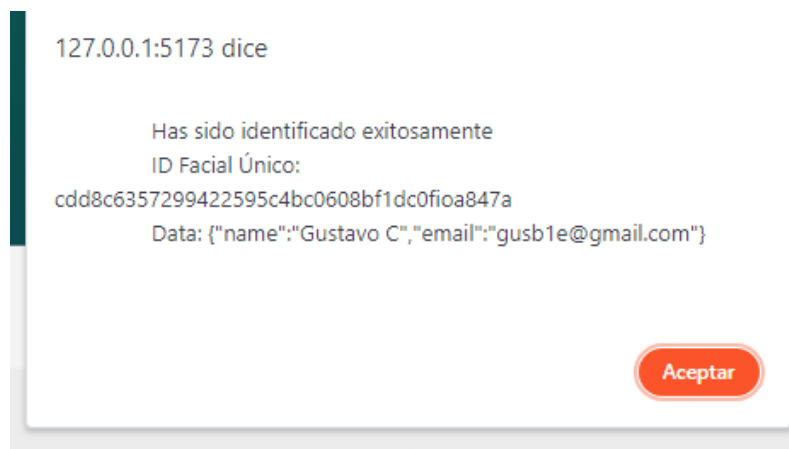


Figura 85. Inicio de sesión exitoso

Inicio de sesión no exitoso

Cuando la aplicación no detecta un rostro conocido al momento de ingresar, o el pin del usuario es erróneo, aparece el mensaje presentado en la Figura 86.

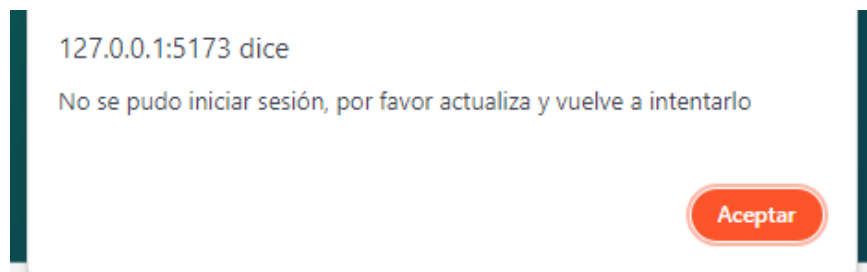


Figura 86. Inicio de sesión no exitoso.

Visualización de perfil exitoso.

Cuando el conductor indica que quiere ver el perfil, presionando en la opción "Perfil", apareció la información brindada por el usuario al momento del registro, lo cual se presenta en la Figura 87.

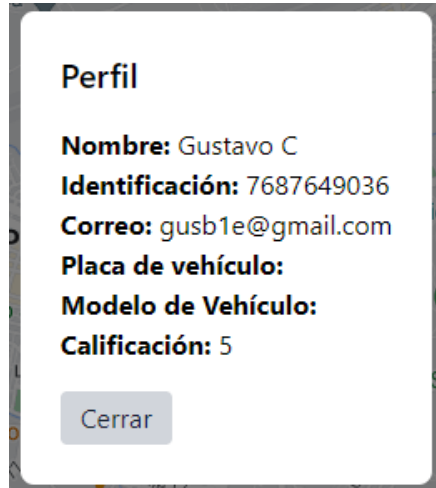


Figura 87. Visualización de perfil de conductor exitoso.

Visualización de perfil no exitoso

En caso de que haya algún problema en la aplicación, ya sea comunicación con el "backend", que la sesión del usuario ya no existe, o que el usuario fue deshabilitado, no aparece ninguna información brindada, los campos aparecen vacíos, tal como se presenta en la Figura 88.

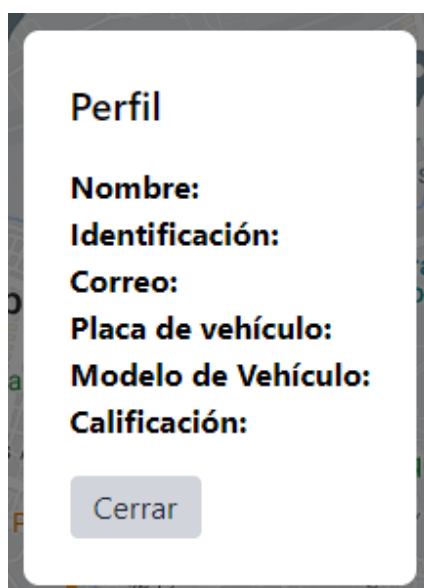


Figura 88. Visualización de perfil de conductor no exitoso.

Cerrar sesión exitosa.

Al presionar en el botón de cerrar sesión, se redirigió la pantalla a la vista principal de la aplicación (Figura 89).

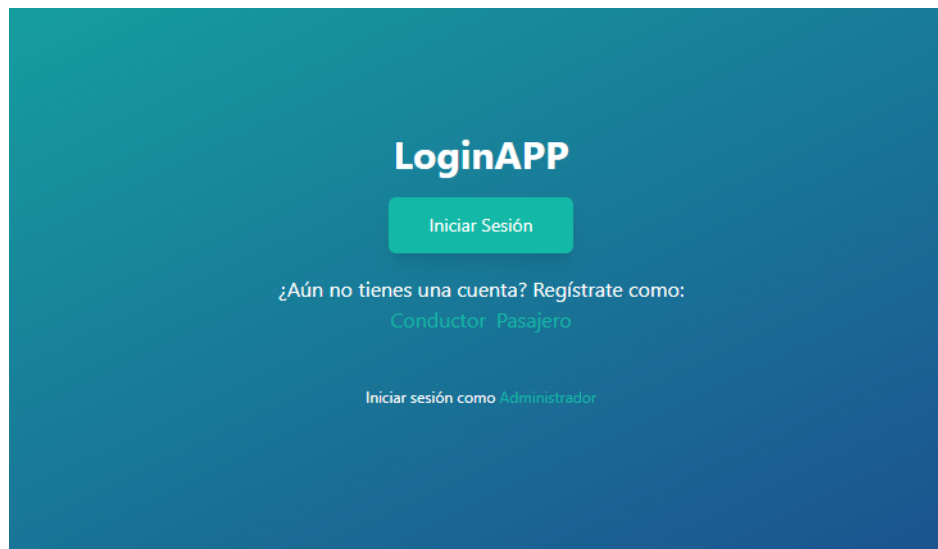


Figura 89. Cerrar sesión exitosa.

Cerrar sesión no exitosa.

En caso de algún error ocurrido en la aplicación cuando el usuario quiera cerrar sesión, no se cierra la vista del usuario. Lo anterior indica que el conductor no puede cerrar sesión, como se presenta en la Figura 90.

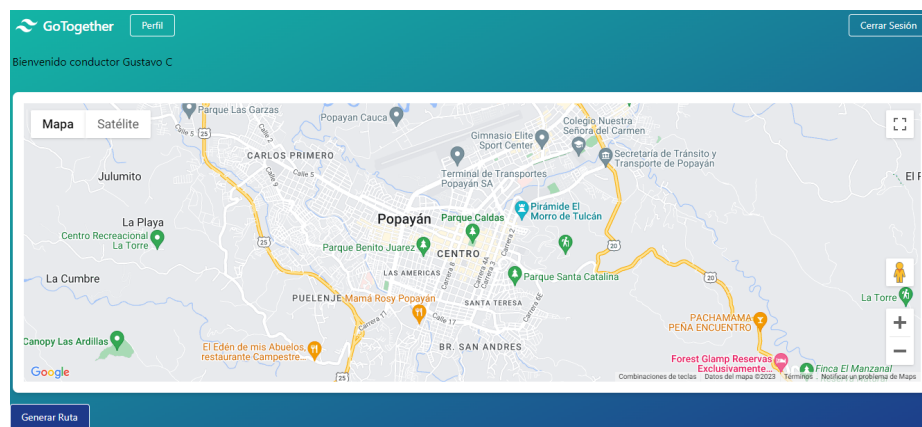
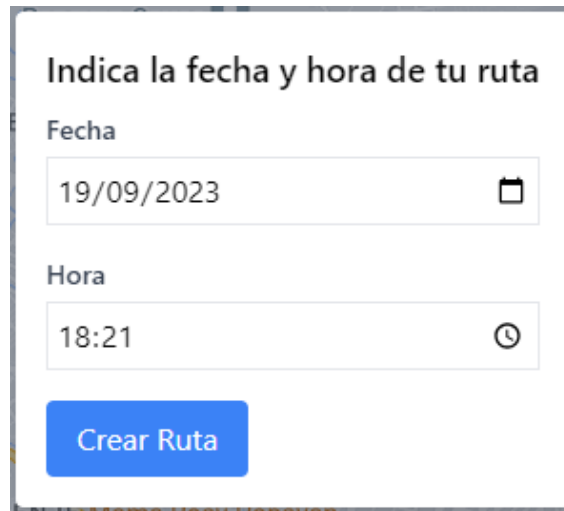


Figura 90. Cerrar sesión no exitoso.

Selección de puntos de ruta de conductor exitoso.

Para la prueba, se seleccionaron tres puntos en el mapa, al presionar en el botón de agregar ruta, apareció el formulario solicitando la hora y fecha de la ruta (Figura 91), lo que indica que los puntos seleccionados fueron válidos.



Indica la fecha y hora de tu ruta

Fecha

19/09/2023

Hora

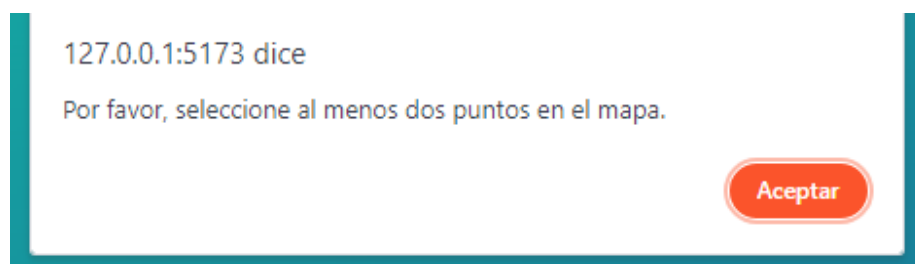
18:21

Crear Ruta

Figura 91. Selección de puntos de ruta de conductor exitoso.

Selección de puntos de ruta de conductor no exitoso.

Para este caso, se seleccionó un único punto en el mapa, al presionar en el botón de agregar ruta, apareció un mensaje, diciendo que se debe seleccionar dos o más puntos en el mapa (Figura 92).



127.0.0.1:5173 dice

Por favor, seleccione al menos dos puntos en el mapa.

Aceptar

Figura 92. Selección de puntos de ruta exitoso.

Selección de fecha y hora de ruta exitosos.

Si los datos suministrados en el formulario anterior son correctos, es decir se selecciona la fecha y la hora mayor o igual a la actual. Se cierra el formulario y se guarda la ruta postulada, compuesta por los puntos seleccionados en el mapa y el

horario seleccionado en el formulario, posteriormente, en el mapa ya se puede observar la ruta del conductor (Figura 93).

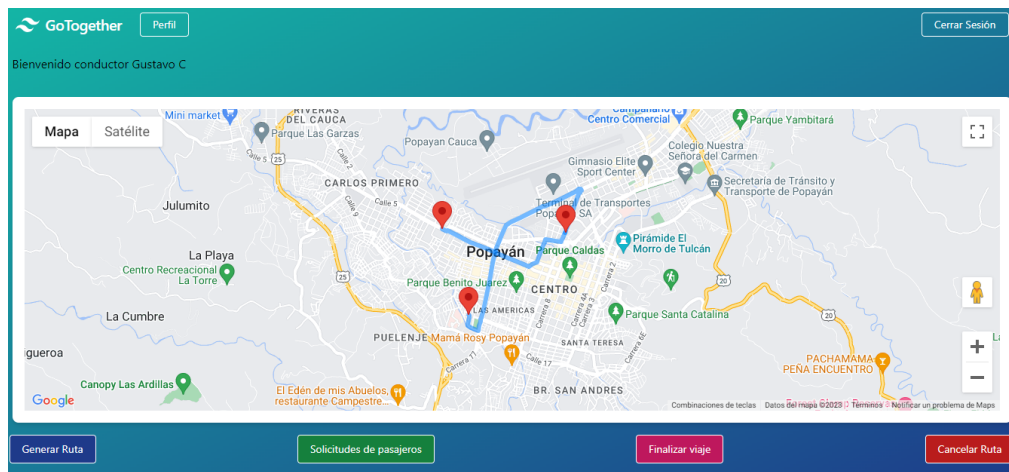


Figura 93. Selección de fecha y hora de viaje exitoso.

Selección de fecha y hora de ruta no exitosa.

Para esta prueba, se seleccionó en el formulario una fecha de un día anterior a la fecha actual, para el programa esta es una fecha inválida ya que no es posible agendar una ruta en una fecha previa. Por lo anterior, el resultado obtenido fue un mensaje, indicando que por favor se seleccione una fecha y hora válidos, tal como se presenta en la Figura 94.

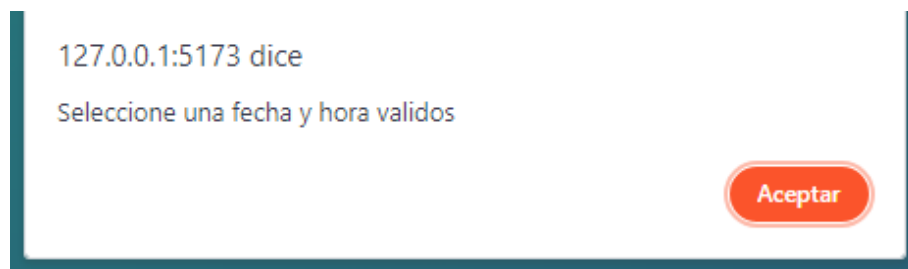


Figura 94. Selección de fecha y hora de viaje no exitoso

Cancelar ruta exitoso.

Se observó que cuando se genera una ruta, se habilitó el botón de “Cancelar Ruta”, al presionar dicho botón, desaparecieron los puntos seleccionados en el mapa, junto con las opciones de la parte inferior como solicitud de pasajeros, finalizar viaje y cancelar ruta, tal como se presenta en la Figura 95.

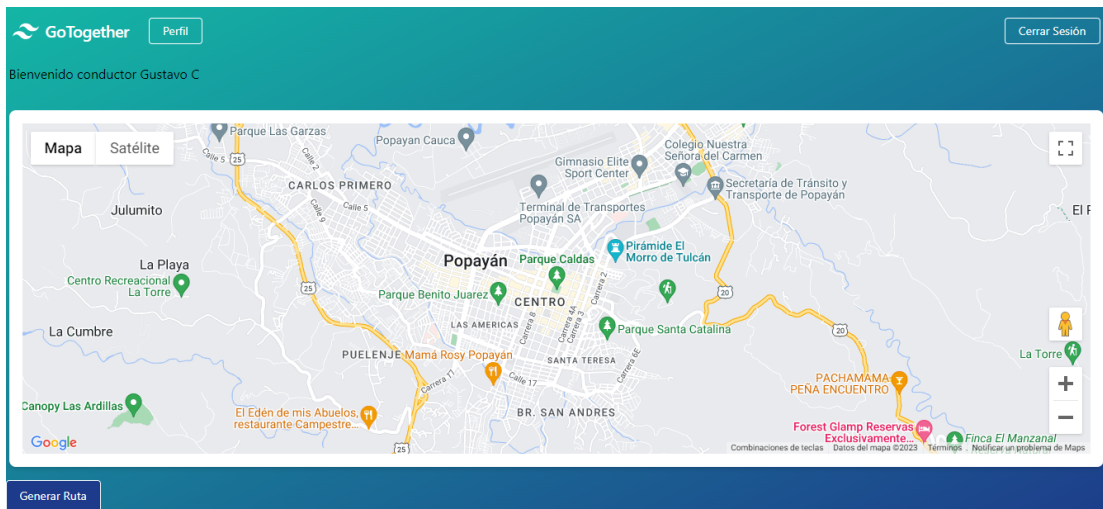


Figura 95. Cancelar ruta exitoso.

Cancelar ruta no exitoso.

Si al presionar en el botón de cancelar ruta, se presenta algún problema, no desaparecen los puntos seleccionados en el mapa, ya que la ruta no pudo eliminarse (sigue vigente), como se presenta en la Figura 96. Para la prueba, se realizó una desconexión de la aplicación con el “backend”, para realizar la simulación de un proceso de cancelación no exitoso.

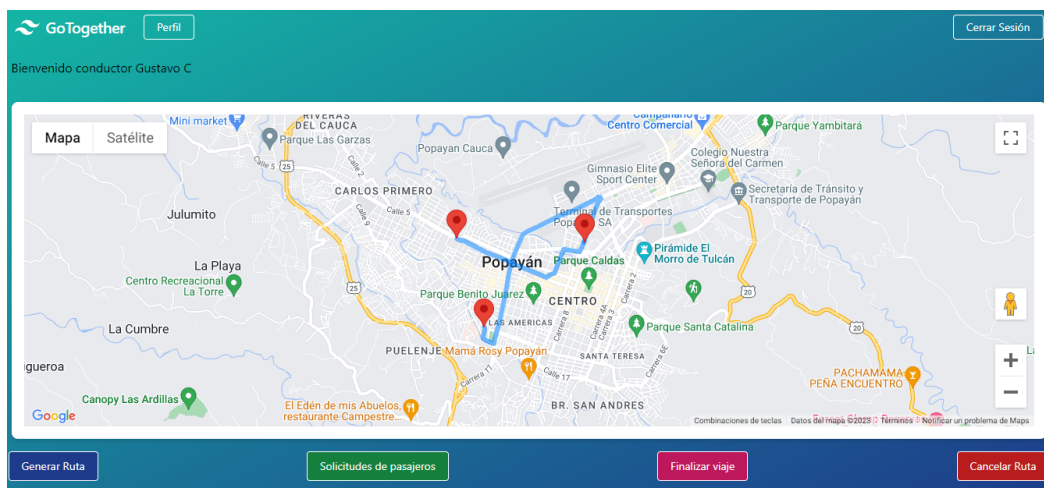


Figura 96. Cancelar ruta no exitoso.

Calificación a pasajeros exitosa.

Para la prueba, se asignó un pasajero a la ruta planteada por el conductor, al momento de finalizar la ruta (presionando el botón de finalizar ruta), se obtuvo una lista de usuarios pasajeros con los que se compartió el viaje, la cantidad de calificaciones por pasajero y 5 estrellas las cuales indican la calificación brindada (1 estrella = 1, 2 estrellas = 2, etc.), tal como se presenta en la Figura 97.

Luego de presionar en el botón de enviar para calificar a los usuarios, se cierra la lista mostrada y se retorna a la pantalla principal de la aplicación. En la vista del usuario conductor, sin ninguna ruta asignada ya que la ruta al finalizar se elimina.

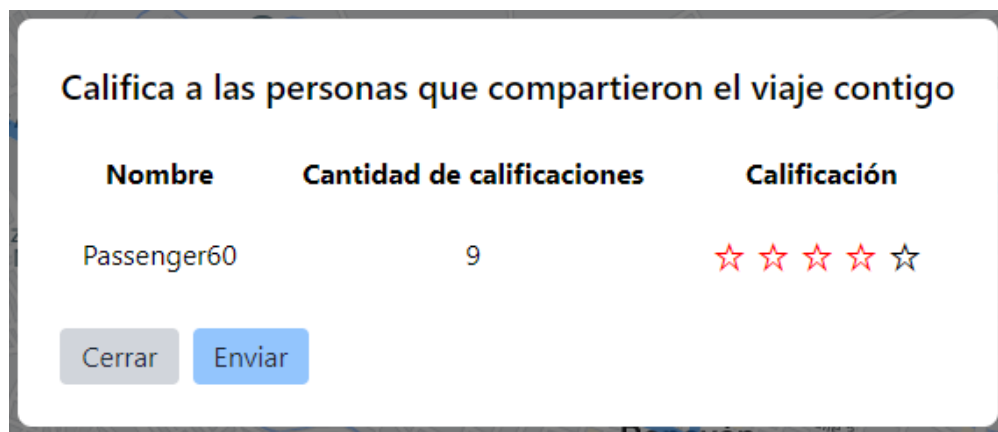


Figura 97. Calificación a pasajeros exitosa.

Calificación de pasajeros no exitosa

Al igual que en la prueba de cancelar ruta no exitosa, se realizó una desconexión de la aplicación con el backend. Cuando el conductor finalizó el viaje, no pudo ver los pasajeros asociados a la ruta, la lista apareció vacía (como se presenta en la Figura 98).

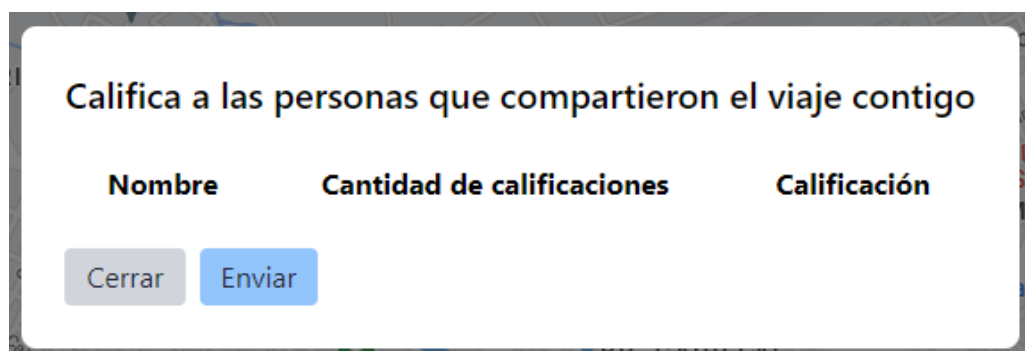


Figura 98. Calificación a pasajeros no exitosa.

6.3. Pruebas de HU administrador

Nombre del tipo de prueba	Resultado esperado	Resultado obtenido
Inicio de sesión exitoso	Redireccionamiento a la vista principal del usuario administrador	Se redireccionó correctamente a la ventana principal del usuario administrador
Inicio de sesión no exitoso	Mensaje de inicio de sesión erróneo	Mensaje indicando que no se pudo iniciar sesión.
Cierre de sesión exitoso	Redireccionamiento a vista inicial de la aplicación.	Se cerró la sesión del administrador y sus vistas asociadas y se redireccionó a la pantalla de inicio de la aplicación.
Cierre de sesión no exitoso	Permanece en la vista del administrador. Esto sólo se presenta si existe algún problema de comunicación entre la aplicación <i>frontend</i> y la aplicación <i>backend</i> .	No se cerró la sesión del administrador y la pantalla permaneció en las vistas del administrador.
Gestión de usuarios exitosa	Se observa los usuarios registrados en la aplicación y permite el cambio del estado (habilitado - deshabilitado)	Se observó a los usuarios de la aplicación tanto conductores como pasajeros, se realizó el cambio del estado de

		dos usuarios satisfactoriamente.
Gestión de usuarios no exitosa	No se observa ningún usuario en las tablas correspondientes	Las tablas en donde se encuentran los conductores y pasajeros estaban vacías.
Iniciar sesión con un usuario deshabilitado	Se deshabilita un usuario, se intenta iniciar sesión con este usuario y se espera que no sea posible.	Permitió ingresar el PIN, pero no fue posible iniciar sesión, se redirigió a la vista de inicio de la aplicación.
Mantener una sesión iniciada y deshabilitar el usuario de dicha sesión	Se deshabilita un usuario que tenga una sesión abierta, se espera que a partir de ese momento no pueda realizar ninguna acción.	El usuario no pudo realizar ninguna acción desde el momento en que se deshabilitó.

Tabla 3. Pruebas a HU de administrador

Inicio de sesión de administrador exitoso

Al proporcionar el usuario y la contraseña correctos, la aplicación inmediatamente permite el ingreso a la vista del usuario administrador tal y como se muestra en la Figura 99, en donde al ingresar se observa la lista de usuarios conductores.

Nombre de Usuario	Correo	Calificación	Acción
Driver60	driver60@gmail.com	3	Deshabilitar
Driver 56	driver4556@gmail.com	5	Deshabilitar
GUS	GUS@GMAIL.COMQ	5	Deshabilitar
Gustavo C	gusb1e@gmail.com	5	Deshabilitar

Figura 99. Inicio de sesión como administrador exitoso.

Inicio de sesión de administrador no exitoso

Si el usuario o la contraseña no son los correctos, aparece un mensaje indicando que no se pudo iniciar sesión y permanece la vista de autenticación del usuario administrador (como se muestra en la Figura 100).

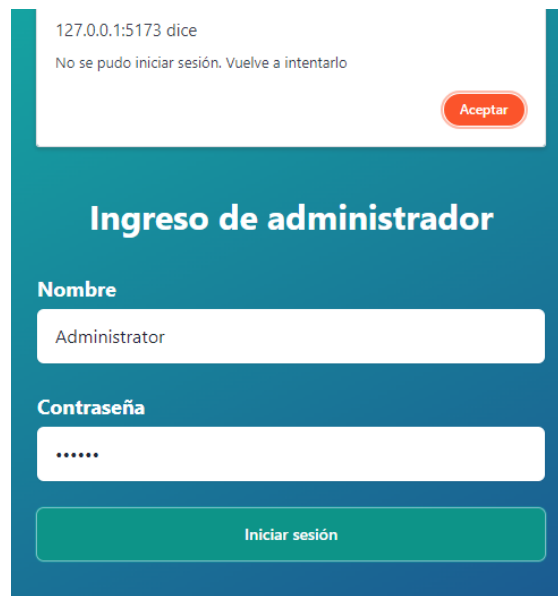


Figura 100. Inicio de sesión no exitoso.

Cerrar sesión como administrador exitoso

Al intentar cerrar sesión como usuario administrador, se presiona el botón de "Cerrar sesión", en donde la aplicación redirecciona al usuario a la vista inicial de toda la aplicación (Figura 101).



Figura 101. Cerrar sesión como administrador exitoso.

Cerrar sesión como administrador no exitoso

Si existe algún problema de conexión y el usuario administrador intenta cerrar sesión, la aplicación no puede hacerlo, por lo que se mantiene la vista del usuario administrador sin poder cerrar la sesión (como se presenta en la Figura 102).



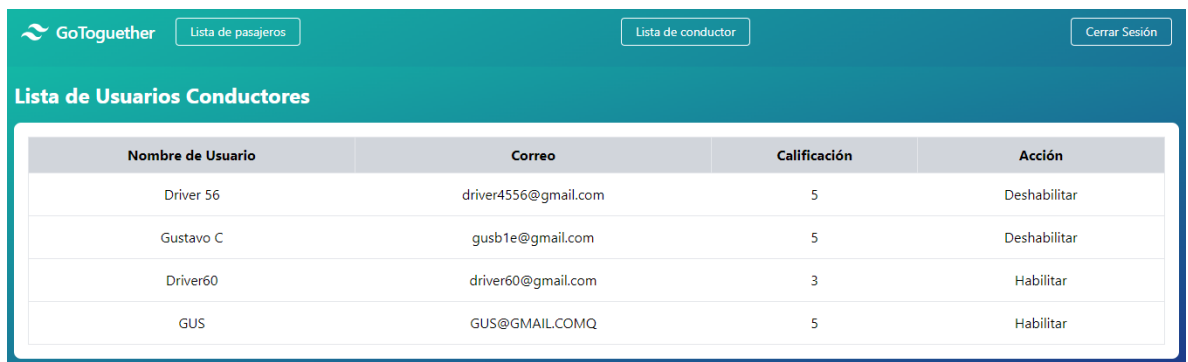
The screenshot shows the 'Lista de Usuarios Conductores' page in the GoTogether application. The page has a teal header with the GoTogether logo and three navigation buttons: 'Lista de pasajeros', 'Lista de conductor', and 'Cerrar Sesión'. Below the header is a table with the following data:

Nombre de Usuario	Correo	Calificación	Acción
Driver60	driver60@gmail.com	3	Deshabilitar
Driver 56	driver4556@gmail.com	5	Deshabilitar
GUS	GUS@GMAIL.COMQ	5	Deshabilitar
Gustavo C	gusb1e@gmail.com	5	Deshabilitar

Figura 102. Cerrar sesión como administrador no exitoso.

Gestión de usuarios exitoso

Cuando el usuario administrador inicia sesión en la aplicación, inmediatamente puede observar la información de los usuarios que se han registrado, y también tiene una opción de habilitar o deshabilitarlos, según sea el caso (Figura 103). Para la prueba, se deshabilitaron a los usuarios “Driver 56” y “Gustavo C”, lo cual se realizó de manera satisfactoria, cambiando el estado inmediatamente en la tabla mostrada.



The screenshot shows the 'Lista de Usuarios Conductores' page in the GoTogether application. The page has a teal header with the GoTogether logo and three navigation buttons: 'Lista de pasajeros', 'Lista de conductor', and 'Cerrar Sesión'. Below the header is a table with the following data:

Nombre de Usuario	Correo	Calificación	Acción
Driver 56	driver4556@gmail.com	5	Deshabilitar
Gustavo C	gusb1e@gmail.com	5	Deshabilitar
Driver60	driver60@gmail.com	3	Habilitar
GUS	GUS@GMAIL.COMQ	5	Habilitar

Figura 103. Gestión de usuarios exitosa.

Gestión de usuarios no exitoso

En caso de que haya algún problema con la sesión iniciada por el administrador o en la comunicación de la aplicación, el administrador no puede ver ninguno de los usuarios registrados, por lo cual tampoco puede habilitarlos o deshabilitarlos (Figura 104).

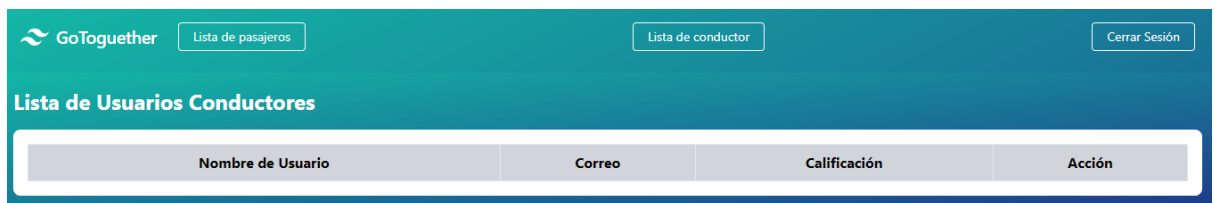


Figura 104. Gestión de usuarios no exitosa.

Iniciar sesión con un usuario deshabilitado

Se deshabilitó el usuario denominado “Ricardo” como se muestra en la Figura 105 y se procedió a intentar iniciar sesión.

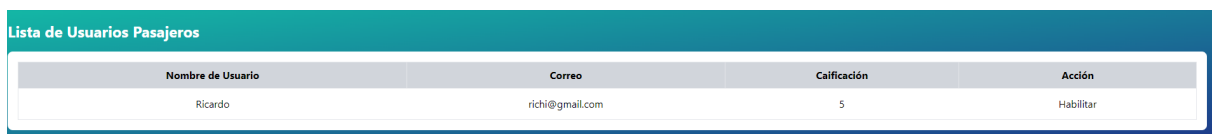


Figura 105. Deshabilitar un usuario.

La respuesta de la aplicación fue volver a la vista de inicio y mostrar un error en consola indicando que no ha sido posible conectarse con el *Backend*, como se presenta en la Figura 106.

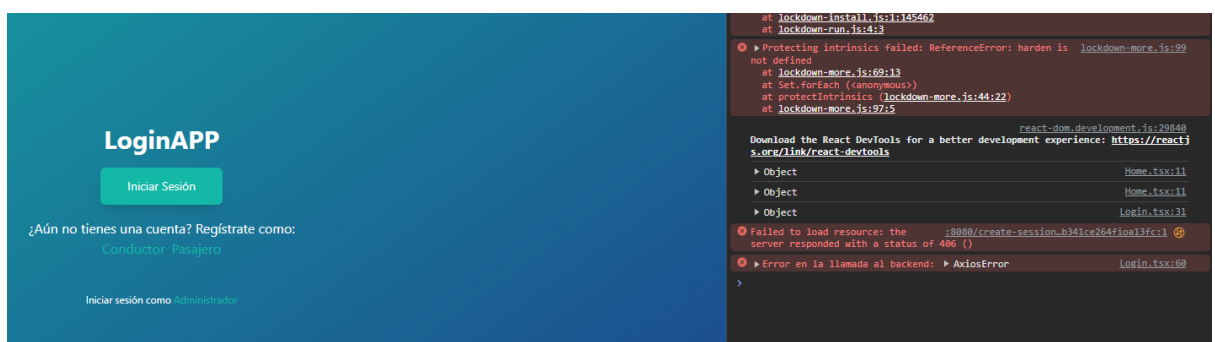


Figura 106. Inicio de sesión con usuario deshabilitado.

Mantener una sesión iniciada y deshabilitar el usuario de dicha sesión

Para este caso se inició sesión con un usuario pasajero como se muestra en la Figura 107 y desde otra instancia de la aplicación se deshabilitó este usuario como se muestra en la Figura 108.

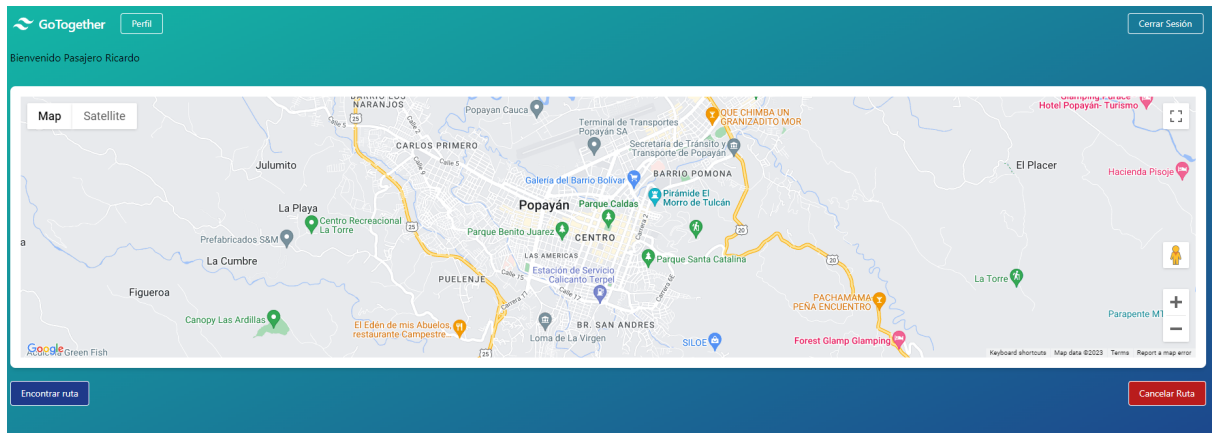


Figura 107. Inicio de sesión con un usuario pasajero.

Lista de Usuarios Pasajeros			
Nombre de Usuario	Correo	Calificación	Acción
Ricardo	richi@gmail.com	5	Habilitar

Figura 108. Deshabilitar usuario pasajero.

Posteriormente, se procedió a seleccionar las opciones "Perfil" y "Encontrar ruta". En el caso de la opción "Perfil", se observó que el cuadro emergente se generó sin datos y se presentó un error en consola indicando que no ha sido posible conectarse con el *backend* como se muestra en la Figura 109. En el caso de la opción "Encontrar ruta", permitió escoger la fecha y hora, pero no encontró rutas y se mostró un error en consola indicando que no ha sido posible conectarse con el *backend* como se muestra en la Figura 110.

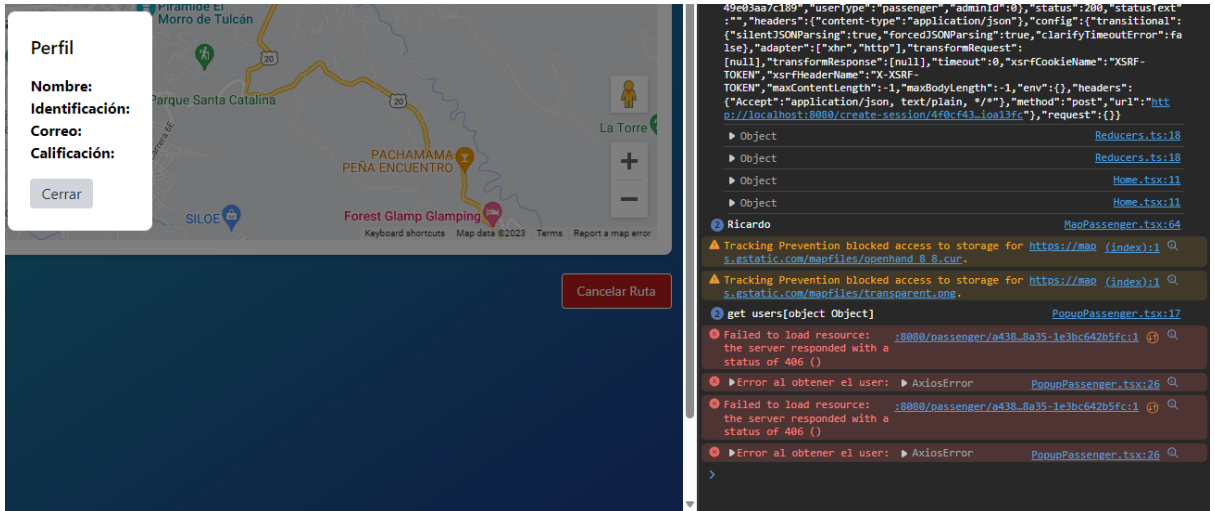


Figura 109. Visualización de Perfil con usuario deshabilitado

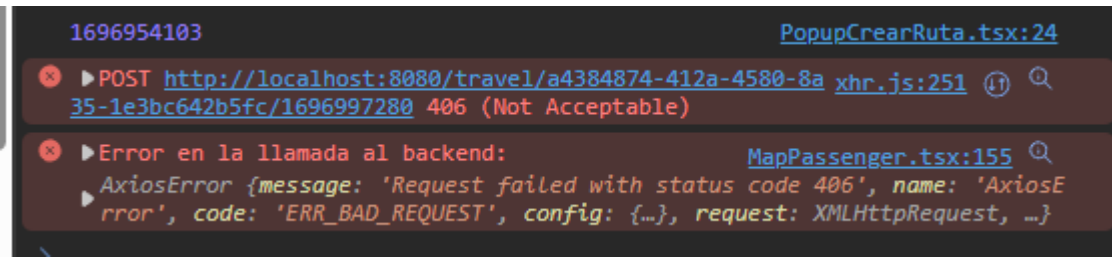


Figura 110. Mensaje de consola al intentar encontrar una ruta con usuario deshabilitado

Referencias

- [1] Zapata, Carlos Mario; Palacio, Carolina, Olaya, Natalí. Unc-Analista: Hacia La Captura De Un Corpus De Requisitos A Partir De La Aplicación Del Experimento Mago De Oz. Rev.EIA.Esc. Ing.Antioq [En línea]. 2007, n.7 [cited 2023-04-25], pp.25-40. Disponible: <http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1794-1237200700010003&lng=en&nrm=iso>. ISSN 1794-1237.
- [2] S. Raghavan, G. Zelesnik y G. Ford, "Lecture Notes on Requirements Elicitation", Defense Technical Information Center, Fort Belvoir, VA, marzo de 1994. [En línea]. Disponible: <https://doi.org/10.21236/ada278536>
- [3] A. Menzinsky, G. López, J. Palacio, M. Á. Sobrino, R. Álvarez y V. Rivas, Historias de usuario - Ingeniería de requisitos ági, 3a ed. Scrum Manager, 2022. [En línea]. Disponible: https://www.scrummanager.com/files/scrum_manager_historias_usuario.pdf
- [4] K. Schwaber and J. Sutherland, "The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game," Scrum.org, Nov. 2020. [En línea]. Disponible: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
- [5] Dangare, Chaitrali & Akila, Ms. (2016). IJARCCCE An Android based application: Cab pooling. International Journal of Advanced Research in Computer and Communication Engineering. 5. 10.17148/IJARCCCE.2016.53138.
- [6] M. Bruglieri, D. Ciccarelli, A. Colorni y A. Luè, "PoliUniPool: a carpooling system for universities", *Procedia - Social Behav. Sci.*, vol. 20, pp. 558–567, 2011. Accedido el 9 de mayo de 2023. [En línea]. Disponible: <https://doi.org/10.1016/j.sbspro.2011.08.062>
- [7] B. Akshay, G. Asmita, J. Kshetrapal and W. Archana, "Car Pool'up – Real-time Carpooling using GPS,". Proceedings of National Conference on New Horizons in IT - NCNHIT, pp. 126-129, 2013.

[8] X. Xiang, M. Wang and W. Fan, "A Permissioned Blockchain-Based Identity Management and User Authentication Scheme for E-Health Systems," in *IEEE Access*, vol. 8, pp. 171771-171783, 2020, doi: 10.1109/ACCESS.2020.3022429.

[9] S. Barra, M. De Marsico, C. Galdi, D. Riccio and H. Wechsler, "FAME: Face Authentication for Mobile Encounter," *2013 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications*, Napoli, Italy, 2013, pp. 1-7, doi: 10.1109/BIOMS.2013.6656140.

[10] N. A. Shamat, S. Sulaiman, and J. S. Sinpang, "A Systematic Literature Review on User Interface Design for Web Applications", *JTEC*, vol. 9, no. 3-4, pp. 57–61, Oct. 2017.

[11] C. Cordoba-Cely, "La experiencia de usuario: de la utilidad al afecto", *iconofacto*, vol. 9, n.º 12, p. 14, 2013.