**Análisis:**

Historias de usuarios

**Conductor:**

|  |
| --- |
| HUC1 |
| **Yo como** conductor…  **Quiero** poder ingresar a la aplicación y modificar mi estado a conductor activo…  **Para que** el sistema registre mi ubicación y me envié solicitudes de viaje. |
| **Detalles adicionales:**   * Tiene la opción de aceptar alguna solicitud de viaje que se le presente. (**pos-re**) * Al final de cada viaje, el conductor puede calificar al o los pasajeros. * Tiene una función que le permite modificar su estado a inactivo. * Puede llevar a varios pasajeros en el mismo viaje, que se pueden bajar en distintas paradas. |
| **Criterios de aceptación:**   * El conductor posee por lo menos un carro y tiene todo el papeleo con respecto a este y su uso en orden. (**pre-re**). Si el algún tipo de papeleo no está en orden, deberá corregirlo y comunicarse con el administrador. (**flujo alternativo 🡪 HUA1**) * El conductor ya está registrado en la plataforma con sus datos y los de sus vehículos. (**pre-re**). Si el algún tipo de papeleo no está en orden, deberá corregirlo y comunicarse con el administrador. (**flujo alternativo 🡪 HUA1**) * Tiene la aplicación y está logueado en esta. (**pre-re**) * Tener credenciales bancarias para que la empresa pueda pagarle. (**pre-re**) Si no se puede acceder a la cuenta bamcaria del conductor, se genera una solicitud por parte del administrador para solucionar el problema. (**flujo alternativo 🡪 HUA1**) * Permite que la aplicación acceda al GPS de su celular. (**pre-re**) La aplicación genera una advertencia para que se active el GPS o no se podrá segur con la actividad. (**flujo alternativo 🡪 HUA1**) |

**Pasajero:**

|  |
| --- |
| HUP1 |
| **Yo como** pasajero…  **Quiero** poder ingresar a la aplicación y que esta me reconozca…  **Para que** el sistema registre mi ubicación y pueda publicar una solicitud de viaje a un lugar de mi preferencia. |
| **Detalles adicionales:**   * Puede ver la ubicación en tiempo real del conductor que ha aceptado su solicitud. * En todo momento, el pasajero puede tener la opción de registrar una tarjeta de crédito en la aplicación en caso de que aún no lo esté, por si quiere pagar por este medio. * Al final de cada viaje, el pasajero puede calificar a su conductor. * Puede redactar una queja sobre un viaje. * Puede escoger las características del vehículo en el que quiere abordar. (las clases A, B y C) * Tiene la capacidad de crear un viaje que va desde su ubicación actual hasta el destino que escoja (con una o más paradas). * El pasajero puede ver características de su conductor asignado, como la placa del vehículo y su modelo, al igual que los datos de conductor (nombre y calificación promedio.) |
| **Criterios de aceptación:**   * El pasajero ya está registrado en la plataforma. (**pre-re**) * Tiene la aplicación y está logueado en esta. (**pre-re**) * Permite que la aplicación acceda al GPS de su celular. (**pre-re**) La aplicación genera una advertencia para que se active el GPS o no se podrá segur con la actividad. * Tener tarjeta de crédito registrada en la aplicación. (opcional) |

**Administrador:**

|  |
| --- |
| HUA1 |
| **Yo como** administrador…  **Quiero** poder ingresar a la plataforma y registrar a los conductores…  **Para que** la aplicación pueda funcionar correctamente prestando el servicio. |
| **Detalles adicionales:**   * Puede registrar a los conductores. (**pos-re**) * Poder ver el historial de viajes de cada conductor (método de pago de los viajes, calificación y quejas). * Puede verificar si un conductor estuvo activo por una zona concurrida de modo que pueda asignarle bonificaciones económicas. |
| **Criterios de aceptación:**   * Contar con acceso a la base de datos de la empresa para poder verla y modificarla. (**pre-re**) * Contar con la información personal y vehicular de un posible conductor para poderla validar y registrarlo tanto en la base de datos como en la aplicación. (**pre-re**) * Contar con la información bancaria de los conductores formales para poder realizar los pagos a estos. (**pre-re**) |
|  |

**Diagram

Description automatically generatedMODELO**

**Clases**

**Conductor**  
Requerimientos funcionales:

-Aceptar o rechazar viaje: Al conductor le deben aparecer en pantalla los pedidos de viaje en tiempo real con su tarifa, y poder escoger sí aceptarlo o rechazarlo. En caso de aceptarlo, se inicia un nuevo viaje.

-Recibir dinero: Al finalizar el ¿mes? ¿O el viaje? El conductor recibe el dinero correspondiente a su trabajo

- Calificar usuario: Al finalizar un viaje, le aparece al conductor la opción para calificar a su pasajero de 1 a 5.

Atributos:

-nombre (String)

-numeroCelular (int)  
-tarjeta (CuentaBancaria)  
-calificacionPromedio (int)  
-vehículo(s) (Vehículo)  
-ubicacionConductor (string)

-registrado (bool)

-historial (Hashmap<Date, Viaje>)

Métodos:

actualizarDatosBancarios()

añadirVehiculo()

calificarUsuario()

**Pasajero:**

Requerimientos funcionales:

- Pedir/cancelar un viaje. El pasajero debe poder hacer un pedido de un viaje, ingresando su ubicación de partida y los destinos que quiera. Una vez ingresada esta información, se calcula la ruta y se hace la solicitud a los conductores disponibles. En cuando un conductor acepte, se muestra la información del conductor y se inicia un nuevo viaje.

- Ingresar ubicación de salida y llegada: El usuario ingresa la información sobre el punto de salida y el destino.

-Ingresar múltiples paradas: Si el usurario lo desea puede añadir múltiples paradas al viaje.

-Calificar al conductor: Al finalizar un viaje, el pasajero debe poder calificar al conductor (1 a 5 estrellas)

-Registrarse en la app: Un nuevo usuario debe poder crear un nuevo perfil en la aplicación.

-Elegir método pago: Al solicitar un viaje, el usuario debe poder escoger su método de pago (efectivo o tarjeta).

-Escribir quejas: Al finalizar el viaje, el usuario puede escribir alguna queja o comentario que tenga al respecto del servicio o la app

Atributos:

- nombre(String)

- correoElectronico (String)

- numeroCelular(String)

- tarjeta (CuentaBancaria)

-ubicacionPasajeros(String)

-registrado(bool)

Métodos:

calificarConductor()

pedirViaje()

cancelarViaje()

escribirQuejas()

añadirParadas()

registrarseApp()

elegirPago()

**Viaje**

Requerimientos funcionales:

Ruta: Debe calcular la ruta óptima según el tráfico actual, irregularidades en las rutas, etc.

Atributos:  
-fecha (Date)

-costo (double)

-distanciaTotal(double)

-ciudad (String)

-condicionTrafico (String)

-demanda(String)

-calificacionPasajeros (int)

-metodoPago (String)

**Vehículo**

Requerimientos funcionales:

Atributos:

-placa (String)

-color (String)

-modelo (String)

-tipo (String)

**Administrador**

Requerimientos funcionales:

-Poder registrar conductores: El administrador debe tener la capacidad de poder registrar a un nuevo conductor en la aplicación. Esto agregara un usuario a la base de datos de la empresa.

-Hacer el pago a conductores: Al finalizar el mes basados en los viajes, los pagos y los descuentos. Se debe realizar el pago a los conductores según corresponde

-Descontar dinero a conductores: A final de mes se revisa el balance entre el pago total y pago en efectivo, luego se acomoda la cuenta del conductor quitándole el efectivo para que todo quede en orden

Atributos:

-puedeAdministrar (bool)

-tarjeta (CuentaBancaria)

Métodos:

RegistrarConductor ()

HacerPagoConductor()

DescontarDinero()

**CuentaBancaria**

Atributos:

- nombrePropietario(String)

-númeroTarjeta(int)

-saldo (int)