

APLICACIÓN SHARE MY CAR MEMORIA FINAL

ESTUDIANTE: JOSE MARÍA MARÍN SÁNCHEZ

CONSULTOR: ROBERT RAMÍREZ VIQUE

ÍNDICE

1. DEFINICIÓN DEL PROYECTO	5
1.1. CONTEXTO DEL PROBLEMA	5
1.2. DESCRIPCIÓN BÁSICA DE LA SOLUCIÓN	5
1.3. DESCRIPCIÓN DETALLADA DE LA SOLUCIÓN	5
2. OBJETIVOS	7
3. ÁMBITO DEL PROYECTO	7
3.1. LIMITACIONES DEL SISTEMA	7
3.2. RECURSOS NECESARIOS	7
3.2.1. RECURSOS HARDWARE	7
3.2.2. RECURSOS SOFTWARE	8
4. CARACTERÍSTICAS DEL PROYECTO	8
4.1. PLANIFICACIÓN	8
4.1.1. CICLO DE VIDA	9
4.1.2. ACTIVIDADES A REALIZAR	9
5. ESPECIFICACIÓN DE REQUISITOS	10
5.1. INTRODUCCIÓN	10
5.2. DESCRIPCIÓN GENERAL	10
5.3. REQUISITOS FUNCIONALES DEL SISTEMA	11
5.3.1. FUNCIONALIDADES DEL ASISTENTE DE REGISTRO DE USUARIO	11
5.3.1. FUNCIONALIDADES DE LA APLICACIÓN	13
5.4. REQUISITOS NO FUNCIONALES DEL SISTEMA	16
5.4.1. Requisitos de Arquitectura	16
5.4.2. Requisitos de Seguridad	16
5.4.3. Requisitos de Disponibilidad	17
5.4.4. Requisitos de Información	17
5.4.4.1. Información del usuario	17
5.4.4.2. Información del vehículo	17
5.4.4.3. Información del mensaje	17

6. ANÁLISIS DEL SISTEMA	18
6.1. Actores	19
6.2. Inventario de casos de uso por sistema y actor	21
6.3. Especificación de casos de uso	23
6.3.1. Verificar dispositivo	23
6.3.2. Recibir confirmación de dispositivo	24
6.3.3. Registrar dispositivo usuario	25
6.3.4. Mostrar asistente	25
6.3.5. Establecer preferencias	26
6.3.6. Establecer vehículo	27
6.3.7. Gestionar rutas favoritas	28
6.3.8. Agregar ruta favorita	29
6.3.9. Eliminar ruta favorita	30
6.3.10. Recibir notificaciones	30
6.3.11. Visualizar viajes disponibles	32
6.3.12. Enviar Mensaje	33
6.3.13. Buscar viajes ofertados	33
6.3.14. Apuntarse a un viaje	34
6.3.15. Gestionar mensajes	34
6.3.16. Enviar notificación	36
6.3.17. Gestionar viajes	36
6.3.18. Valorar conductor	37
6.3.19. Desapuntar viaje	38
6.3.20. Organizar viajes	39
6.3.21. Publicar viaje	40
6.3.22. Gestionar seguimiento	41
6.3.23. Cancelar viaje	41
6.3.24. Consultar viajes realizados	42
6.4. Entidades del dominio	43
7. DISEÑO DEL SISTEMA	44
7.1. ARQUITECTURA DEL SISTEMA	44
7.2. DISEÑO ESTÁTICO	44
7.3. DISEÑO DE LA PERSISTENCIA	46
7.3.1. MODELO RELACIONAL	46
7.4. DISEÑO DE LAS INTERFACES DE USUARIO	47
7.4.1. Prototipos del asistente	47
7.5. DECISIONES TECNOLÓGICAS	50

7.5.1.	Lenguajes de Desarrollo	50
7.5.2.	Soporte de bases de datos local	50
7.5.3.	Soporte de bases de datos remota	50
7.5.4.	Acceso a datos	50
7.5.5.	Soporte de comunicaciones entre clientes y servidor	50

8. IMPLEMENTACIÓN 51

8.1.	HITOS CONSEGUIDOS	51
8.2.	HITOS A ACOMETER EN UNA SEGUNDA FASE	52
8.3.	FUTURAS MEJORAS A INCORPORAR	53
8.4.	INVENTARIO DE COMPONENTES SOFTWARE	53
8.4.1.	PAQUETES	53
8.4.2.	COMPONENTES	54
8.5.	CONCLUSIÓN	59

1. DEFINICIÓN DEL PROYECTO

1.1. CONTEXTO DEL PROBLEMA

Debido a la coyuntura económica que padecemos en la actualidad y el incesante incremento de los precios de los carburantes, cada vez más personas están buscando soluciones que permitan costearse sus gastos compartiendo sus propios recursos.

El caso que se trata de analizar en este documento es el caso de aquellas personas que ponen a disposición su vehículo para compartir con terceros a la hora de desplazarse a una ubicación determinada ya sea para ir a trabajar de forma rutinaria o bien para realizar un viaje puntual entre dos puntos en una fecha concreta. De esta forma, la persona que comparte su vehículo obtiene a cambio una compensación económica por parte de las personas que se desplazan con ésta con el objetivo de reducir sus gastos de desplazamiento. Por ejemplo, utilizando la herramienta carpooling, podemos observar que la oferta existente de conductores que se desplazan desde Madrid a Las Rozas es de 2€ por plaza libre. Teniendo en cuenta que en la mayoría de turismo, el número de plazas disponibles (sin tener en cuenta la plaza del conductor lógicamente) es de cuatro, el conductor se estaría embolsando un máximo de 8 € por desplazamiento cuando el coste que le supondría desplazarse él sólo es de 3,15 € (éste cálculo se ha realizado en base a un desplazamiento de 30 km, a un consumo medio de 7 litros cada 100 kilómetros y un coste de 1,50 € el litro de carburante y por tanto variará en función del vehículo que se emplee en el trayecto). Como se puede ver en este ejemplo, todas las partes ganan ya que sólo con cubrir una plaza, el conductor ya cubre más de la mitad de los gastos de dicha ruta.

1.2. DESCRIPCIÓN BÁSICA DE LA SOLUCIÓN

Se trata de desarrollar una herramienta interactiva que facilite el contacto entre personas dispuestas a compartir coche teniendo en cuenta su ubicación. A pesar de que actualmente existen algunas aplicaciones en el mercado tales como blablacar o carpooling, busvao, etc... estas herramientas carecen de la posibilidad de aprovechar las ventajas que ofrecen hoy en día las nuevas tecnologías en el campo de los dispositivos móviles –tales como la geolocalización, - y sin embargo se limitan a mostrar la información como una extensión de su página web sin tener en cuenta la ubicación del usuario que solicita la información. Esta forma de planteamiento de cara al usuario es poco eficaz e intuitiva y por tanto la solución se va a centrar en tener en cuenta la ubicación del usuario (ya sea facilitada por el propio usuario, la red o bien por el sensor GPS del dispositivo) y garantizar su usabilidad.

De esta forma se conseguirá localizar a aquellos conductores que ofrecen trayectos en un radio establecido por el usuario en función de su ubicación actual y destino; y por tanto, será mucho más cómodo para el usuario encontrar conductores próximos y también para los conductores; ya que se pueden reducir aún más los costes de desplazamiento ya que sus posibles clientes podrían encontrarse en un radio relativamente próximo.

1.3. DESCRIPCIÓN DETALLADA DE LA SOLUCIÓN

Cuando la aplicación se inicia, comprobará si existe información del usuario en el dispositivo. Si no existe, se iniciará un asistente que ayudará al usuario a registrarse en el sistema y a establecer unas preferencias básicas tales como la configuración de su ubicación (establecimiento de forma manual o automática), el radio de búsqueda y destinos favoritos. Si existe, comprobará que el identificador de dispositivo coincide con el que existe en su cuenta de usuario en el servidor. Si son iguales, arrancará la aplicación normalmente. Si no coinciden, se mostrará en pantalla un mensaje indicando al usuario que ha cambiado de dispositivo y deberá iniciarse el proceso de verificación de usuario. Con este procedimiento se evitará que un usuario pueda utilizar la aplicación en más de un dispositivo con una única cuenta de usuario.

El proceso de verificación de usuario se realizará mediante el envío de un SMS. En cuanto el terminal del usuario recibe el SMS en su dispositivo quedará registrado/validado en el sistema y almacenará en servidor el identificador del dispositivo, en la cuenta del usuario. Este sistema limita el uso exclusivo a un solo dispositivo móvil por usuario y en tablets no estará permitido; es decir, si

la aplicación detecta que el dispositivo no es móvil lo indicará en pantalla y en cuanto el usuario acepte, se cerrará la aplicación.

Para el establecimiento de la ubicación de forma manual se utilizarán los recursos de Google Maps que proporcionan sus servicios en formato JSON para que el usuario pueda seleccionarla de una lista. De ésta forma podremos obtener las coordenadas terrestres del usuario que proporciona Google Maps necesarias para cálculos posteriores (si éste opta por indicar su ubicación de forma manual). Para el establecimiento de la ubicación de forma automática existirían dos modalidades; o bien facilitadas por la red o bien por el sensor GPS del dispositivo.

La aplicación se diseñará para que soporte múltiples idiomas. En principio estarán soportados el español y el inglés. El idioma por defecto de la aplicación será obtenido del dispositivo móvil. Si el usuario opta por un idioma diferente soportado por la aplicación utilizará siempre éste.

Una vez iniciada la aplicación se mostrará un listado de todos los conductores disponibles para los trayectos que coinciden con su ubicación, sus destinos favoritos y el radio de acción tanto de origen como destino. Estos parámetros podrán ser modificados en cualquier momento.

La aplicación dispondrá de una sección de destinos favoritos. Ésto facilitará aún más las cosas a los usuarios puesto que la aplicación les notificará de aquellos nuevos trayectos que se vayan incorporando por nuevos conductores y que coincidan con los radios de búsqueda tanto de origen como destino.

La aplicación dispondrá de un buscador que permitirá informar al usuario de otras rutas ofertadas diferentes de sus favoritas y como punto de partida su ubicación y opcionalmente, otra ubicación alternativa. Este caso es útil cuando una persona tiene itinerancia y se desplaza a varios puntos con frecuencia y necesita saber si a donde se dirige hay rutas ofertadas.

Dispondrá de un sistema de notificaciones que será gestionado por un servicio de aplicación que realizará peticiones PULL cada cierto tiempo y comprobará si existen notificaciones pendientes de recibir por parte del usuario. Las notificaciones que pueden recibir los usuarios son o bien trayectos nuevos disponibles en base a su ubicación y sus destinos favoritos (teniendo en cuenta siempre su radio de acción); o bien mensajes de otros usuarios. Los trayectos disponibles serán persistentes en el servidor mientras que los mensajes que se reciban serán almacenados en un fichero de base de datos local y se borrarán del servidor (siempre y cuando el destinatario lea el mensaje enviado por su remitente). En función de la notificación recibida por parte del sistema operativo, se abrirá la aplicación en su sección correspondiente.

El usuario que establezca una ruta o trayecto por primera vez pasa a ser conductor y tendrá que introducir información adicional como por ejemplo, el modelo del coche, el tipo de carburante, etc... Para el trayecto deberá introducir el origen, destino, fecha (puntual o periódica). La fecha puntual servirá para indicar que se trata de un viaje esporádico y la periódica podrá establecer los días de la semana que realiza el viaje entre un intervalo de fechas y el horario. Por ejemplo, un conductor puede especificar que realizará el trayecto Madrid-Las Rozas durante el mes de Octubre de 2013, los lunes, miércoles y viernes a las 8:30.

El usuario que se apunta a un trayecto de un conductor se convierte en su cliente y por tanto, en ese momento deberá aceptar las condiciones del servicio. El cliente podrá desapuntarse de un trayecto a través de la aplicación indicando los motivos.

La aplicación también dispondrá de un sistema de mensajería que permita la comunicación entre un conductor y sus clientes. Se analizará si es conveniente que cuando un usuario envía un mensaje y se produce un error en la transmisión, se pueda almacenar temporalmente en el dispositivo hasta que lo pueda reenviar en un tiempo establecido.

Por último, la aplicación tendrá un sistema de valoración de los usuarios que servirá para conocer el nivel de confianza entre el conductor y sus clientes con objeto de medir el grado de satisfacción de cada una de las dos partes con objeto de garantizar a sus usuarios una buena calidad de servicio.

2. OBJETIVOS

Los objetivos que se pretenden conseguir con la realización de este trabajo son los siguientes:

- Poner en práctica los conocimientos adquiridos en Ingeniería de Software necesarios para elaborar la documentación requerida para el diseño de la solución.
- Adquirir los conocimientos necesarios para poder implementar la solución aprovechando el potencial que ofrece la plataforma de desarrollo android y sus herramientas.
- Emplear nuevos mecanismos de intercambio de datos tales como JSON.
- Utilizar algoritmos de encriptación en el intercambio de datos con el servidor para garantizar la seguridad de los datos transmitidos.
- Aprovechar al máximo las posibilidades que ofrecen los servicios web externos tales como Google Maps que permitan ahorrar tiempo y esfuerzo.
- Cumplir con los plazos estimados e intentar evitar las desviaciones debido a las posibles dificultades técnicas que podrían surgir durante la implementación.
- Garantizar la usabilidad de la aplicación, es decir, que sea intuitiva, eficaz, que disponga de una interfaz agradable y sea fácil de manejar; y sobre todo, que pueda aplicarse su diseño como patrón para desarrollar otras aplicaciones utilizando el mismo planteamiento.
- Demostrar que el aprovechamiento de las posibilidades de ubicuidad que ofrece la plataforma android garantiza la usabilidad de la aplicación.
- Demostrar la capacidad de la plataforma android y sus servicios integrados en los dispositivos móviles para la implementación de la solución propuesta.

3. ÁMBITO DEL PROYECTO

3.1. LIMITACIONES DEL SISTEMA

- La aplicación estará limitada a dispositivos móviles y no podrá ser utilizada en tablets. Se permitirá su uso en dispositivos con sistema operativo android a partir de la versión 2.3.3 Gingerbread (API 10).
- La aplicación sólo estará disponible con conexión a Internet. Si no existe conexión deberá indicarse en pantalla.
- Sería más completo si dispusiera de una interfaz web para que los usuarios pudieran gestionar sus trayectos desde un ordenador pero queda fuera del ámbito del proyecto.
- La obtención de datos se realizará a través de un servidor con soporte PHP 5 y tenga instalado el soporte de las librerías mcrypt, MySQL, mysqli y soporte formato de salida JSON-UTF8 vía HTTP.
- Por cuestiones de tiempo sólo se contemplará el envío de mensajes entre el ofertante (el conductor) y los interesados de forma individual. No se contempla crear grupos. Tampoco se podrá adjuntar ningún fichero de ningún tipo.
- Las posibles pantallas necesarias que contemplen los términos y condiciones del servicio quedarán fuera del ámbito del proyecto.

3.2. RECURSOS NECESARIOS

3.2.1. RECURSOS HARDWARE

- Equipo portátil con Windows 7 64 bits.
- Samsung Galaxy S con S.O. android Jelly Bean 4.2.2
- Tablet bq Kepler 2

3.2.2. RECURSOS SOFTWARE

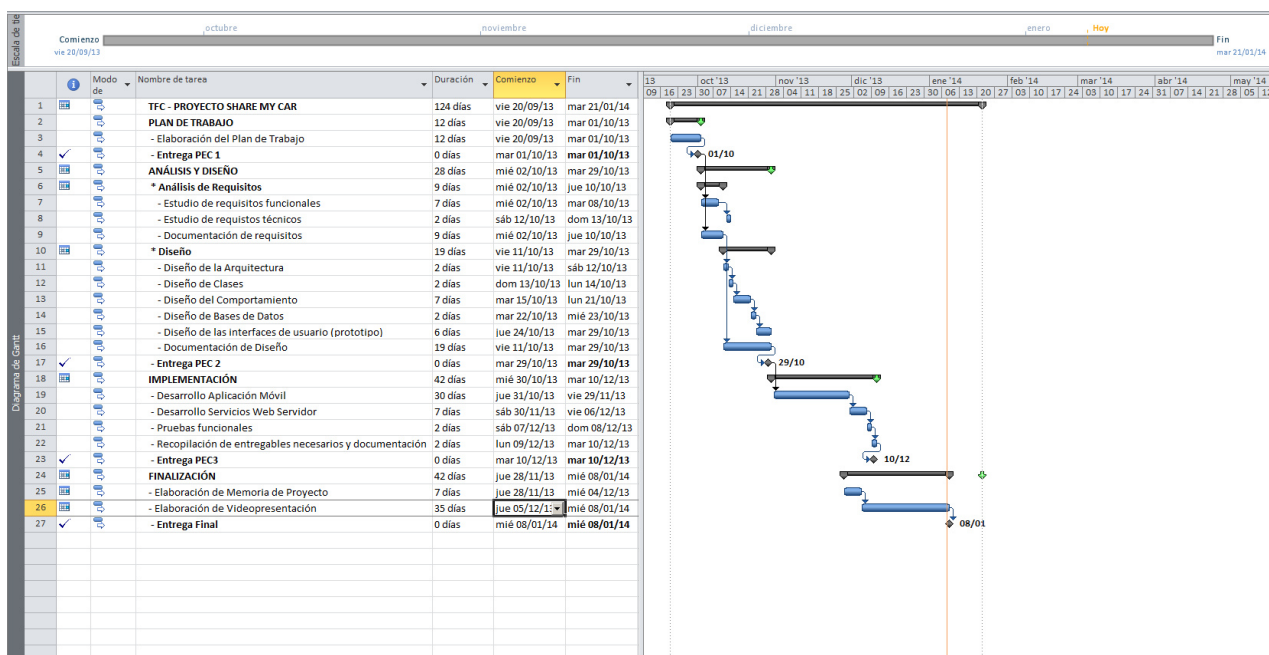
El software descrito a continuación se utilizará en plataforma Windows

- Para la documentación se utilizarán las herramientas Microsoft Word, Microsoft Visio y Adobe PhotoShop CS4 + Eclipse ADT (para el diseño de prototipos)
- Para el desarrollo se utilizarán las herramientas Eclipse ADT + Android SDK (API mínima soportada: 10); EMS MySQL Manager (IDE para MySQL); SQLite Expert (IDE para SQLite); Apache Server 2.2 + MySQL 5.6 + PHP 5.3; APIs de Google Maps y Adobe PhotoShop CS4 (para la creación de recursos gráficos)
- Para la videopresentación se utilizará Microsoft PowerPoint, Teamviewer Quick Support (en el dispositivo), Teamviewer 8 (para la grabación de la demo) y Sony Vegas Pro Video (para la videoedición).

4. CARACTERÍSTICAS DEL PROYECTO

4.1. PLANIFICACIÓN

A continuación se muestra el diagrama de Gantt de las actividades a realizar durante la duración del proyecto. Para la estimación de tiempos se ha tenido en cuenta los fines de semana como días laborables. El tiempo de duración total del proyecto se ha estimado en 124 días. Para la presentación de la PEC 2 se ha distribuido el tiempo en dos tandas; una para el análisis de requisitos estimado en 9 días y la segunda tanda correspondiente al diseño, de 19 días. La documentación se ha tenido en cuenta como una actividad paralela al estudio tanto en el análisis como en el diseño. En la implementación se destina más tiempo al desarrollo de la aplicación móvil, 30 días; respecto a los 7 días que se destinarán al desarrollo de los servicios en la capa de servidor necesarios para la aplicación. Para la entrega final se ha considerado emplear 7 días en la elaboración de la memoria y 35 días en la preparación de la presentación.



4.1.1. CICLO DE VIDA

Se ha optado por utilizar el ciclo de vida clásico para planificar las tareas adaptadas a las necesidades del Trabajo de Fin de Carrera.

4.1.2. ACTIVIDADES A REALIZAR

Actividad	Descripción
Elaboración del Plan de Trabajo (12 días)	Su objeto es el de tener una idea clara sobre el problema que se pretende abordar y documentarla; definir los objetivos, el ámbito y el alcance del proyecto de acuerdo con la metodología de desarrollo a utilizar y los hitos establecidos.
Estudio de los requisitos funcionales (7 días)	Se analizará en profundidad todos los casos de uso de la aplicación, identificando a los actores que interactuarán con el sistema.
Estudio de los requisitos técnicos (2 días)	Se identificarán que requisitos de interfaz, seguridad, operatividad, etc... serán necesarios para la aplicación.
Documentación de requisitos (9 días)	Mientras se realiza el estudio en profundidad de los casos de uso de la aplicación y sus requerimientos técnicos, se irán incorporando al documento final.
Diseño de la Arquitectura (2 días)	Se definirá la arquitectura global de la solución.
Diseño de Clases (2 días)	Se creará el modelo estático de clases
Diseño del Comportamiento (7 días)	Será muy útil para describir cómo la aplicación implementa su funcionalidad y su comportamiento en el caso de errores y otros escenarios. Permitirá conocer el flujo de ejecución de la aplicación.
Diseño de Bases de Datos (2 días)	Se creará el modelo relacional de las base de datos implicadas y se definirán sus entidades. Se crearán también los modelos de datos correspondientes.
Diseño de las interfaces de usuario (prototipo) (6 días)	Se definirán las diferentes actividades (interfaces de usuario del sistema android) que formarán parte de la aplicación.
Documentación de Diseño (19 días)	Conforme se vaya avanzando en el diseño, se irá incorporando a la información al documento de análisis y diseño para su entrega final.
Desarrollo de la Aplicación Móvil (30 días)	Se creará el software necesario siguiendo las pautas definidas en sus especificaciones.
Desarrollo de servicios Web Servidor (7 días)	Se creará el software externo necesario que de soporte a la aplicación siguiendo las especificaciones de diseño.
Pruebas funcionales (2 días)	Se realizarán las pruebas necesarias para detectar posibles errores en la implementación y corregirlos.
Recopilación de entregables necesarios y documentación (2 días)	Se recopilarán todos los elementos necesarios (código fuente, scripts de bases de datos, etc...) para poder distribuir la aplicación así como la documentación necesaria para su instalación.
Elaboración de Memoria de Proyecto (7 días)	Se elaborará la memoria del proyecto, donde se sintetizará el trabajo realizado y se mostrará que se han alcanzado los objetivos propuestos.

Elaboración de Videopresentación (35 días)	Resumirá de forma concisa el trabajo realizado y sus resultados.
Entrega Final	Se entregará la memoria del Proyecto, el video de presentación y el instalable de la aplicación.

5. ESPECIFICACIÓN DE REQUISITOS

A continuación se detallarán todas las necesidades de información que tendrá que resolver el software, sin tener en cuenta por el momento los medios técnicos.

5.1. INTRODUCCIÓN

Con el presente documento se pretende formalizar todas las funcionalidades que debe soportar el software de gestión de compartición de plazas de vehículos para viajar y los usuarios que van a hacer uso de él.

5.2. DESCRIPCIÓN GENERAL

Se trata de crear una aplicación móvil que facilite el contacto entre personas que comparten coche –de hora en adelante, los conductores- con cierto número de plazas disponibles a la hora de desplazarse a un punto y las personas interesadas en ocupar las plazas libres que pone el conductor a su disposición.

El sistema ofrecerá a los usuarios la posibilidad de conocer todos los itinerarios que realizarán los conductores a lo largo de la semana durante todo el año y en especial, aquellos trayectos cuyo origen se encuentren próximos a la ubicación del usuario. También ofrecerá la posibilidad de mostrar la posición de los ofertantes en un mapa.

El usuario podrá establecer sus preferencias tales como configurar su ubicación y establecer su radio de búsqueda. También podrá añadir rutas favoritas. Las rutas favoritas serán aquellas que tienen por origen su ubicación u otra diferente (teniendo en cuenta su radio de cobertura) y como destino la que establezca como destino favorito.

Si el usuario no establece ninguna ruta favorita, se mostrarán todas las rutas ofertadas respecto a su ubicación. Si establece al menos una, se mostrarán todas las ofertas que cumplan ese criterio y opcionalmente podrá ver todas como en el caso anterior. Al establecer al menos una ruta favorita podrá recibir notificaciones de nuevas ofertas que coincidan con su ruta favorita.

La ubicación del usuario puede ser establecida de forma manual y automática. De forma manual será proporcionada por el usuario utilizando los servicios de ubicación (como por ejemplo Google Maps) mientras que la automática será delegada en el dispositivo.

El usuario opcionalmente podrá ampliar el radio de búsqueda respecto de su posición. Esto proporcionará al usuario la comodidad de encontrar posibles ofertantes cerca de su ubicación. También podrá ampliar el radio de búsqueda en el destino.

El usuario opcionalmente podrá buscar aquellas rutas que le puedan interesar con independencia de su ubicación. El usuario que quiera compartir su coche (de ahora en adelante, el conductor) podrá hacerlo en cualquier momento y a través de un pequeño asistente. El conductor podrá añadir su coche en el que especificará sus características y establecer el número de plazas que comparte.

El conductor podrá establecer sus rutas (que podrán ser más de una) a distintos lugares y podrá especificar su frecuencia (diaria, una vez por semana, días de la semana) y establecer por cuánto tiempo la mantendrá, luego deberá establecer su naturaleza, es decir si la ruta será periódica o puntual. Si es puntual, no se podrá lógicamente establecer su frecuencia. Al crear la ruta también tendrá que dar cuenta de la hora que tiene previsto salir y su precio. Al crear la ruta, define un punto de origen o salida y otro de destino o llegada. El punto de origen puede ser su ubicación (si en sus preferencias está su ubicación configurada en manual) o bien, elegir otra alternativa a modo de punto de encuentro con los usuarios que se han apuntado al viaje en la fecha indicada.

El conductor podrá establecer las restricciones de su vehículo (si permite fumar, mascotas o si pone música mientras viaja). El usuario interesado podrá apuntarse a la ruta de un conductor para una fecha en particular. El pasajero también podrá dejar una plaza libre si cancela su asistencia.

El p que ha participado en un viaje (el pasajero) podrá dar su opinión sobre su experiencia y valorar al conductor con una puntuación de 1 a 5. El usuario podrá consultar sus rutas favoritas, sus rutas realizadas, sus mensajes y sus valoraciones realizadas (a conductores) y las valoraciones de los conductores.

El conductor podrá adicionalmente consultar sus rutas programadas y las valoraciones de los usuarios.

El sistema dispondrá de un subsistema de mensajería que permitirá la comunicación entre el conductor y los interesados en su ruta. Esto facilitará que si un usuario apuntado al viaje no puede asistir, pueda avisar al conductor y de esta forma, el último pueda dejar la plaza libre.

Sólo podrán acceder al servicio los usuarios registrados. El procedimiento de registro del usuario se realizará a través de un asistente la primera vez que ejecuta la aplicación. El asistente permitirá además recopilar los datos personales del usuario, establecer sus preferencias, establecer sus rutas favoritas y adicionalmente podrá añadir sus datos como conductor, iniciando el asistente de nuevo conductor.

5.3. REQUISITOS FUNCIONALES DEL SISTEMA

5.3.1. FUNCIONALIDADES DEL ASISTENTE DE REGISTRO DE USUARIO

El asistente se iniciará en el idioma que se encuentre configurado el dispositivo del usuario. Tendrá como objeto principal registrar al usuario en el sistema. También tendrá como objetivos los siguientes aspectos:

- Verificar dispositivo, para garantizar que la aplicación sea utilizada por un único usuario y comprobar que es el que solicita el registro. Para verificar su dispositivo, será necesario solicitarle su número de teléfono. El país por defecto será el que se obtenga del dispositivo y se podrá seleccionar otro que no corresponda con el del dispositivo. Se realizará a través del sistema de red móvil mediante SMS.
- Obtener país, obtiene el país del dispositivo.
- Seleccionar país, opcionalmente podrá seleccionar otro país diferente, lo que conllevará seleccionarlo de una lista de países.
- Listar países, se listarán todos los países que estén soportados y en el idioma del dispositivo (disponibles español e inglés). Opcionalmente se puede buscar el país de la lista.
- Buscar país, mientras el usuario escribe el país, la lista se va acotando con el nombre que va escribiendo.
- Confirmar dispositivo, cuando se recibe confirmación a través del sistema de red móvil durante la recepción del SMS de verificación para garantizar que el dispositivo es quien dice ser. En cuanto se recibe la confirmación, el usuario introducirá sus datos personales, establecerá sus preferencias, podrá crear sus rutas favoritas y finalmente, el usuario podrá optar por iniciar el asistente de nuevo conductor si desea compartir su coche. La confirmación del dispositivo conllevará registrar al usuario en el sistema y con objeto de identificarle en posteriores usos, se almacenarán sus datos en el dispositivo y en el servidor.
- Registrar dispositivo usuario, se almacenarán los datos del usuario en el dispositivo para futuras comprobaciones y se creará una nueva cuenta en el servidor con los datos del usuario y su dispositivo.
- Introducir datos personales, el usuario introducirá sus datos personales. Una vez introducidos y el usuario decida pasar al siguiente paso del asistente, se actualizarán en el servidor.
- Actualizar datos personales servidor, se agregarán los datos personales del usuario a los datos de la cuenta del usuario en el servidor.

- Establecer las preferencias del usuario, entre las que podrá seleccionar el idioma si prefiere uno distinto del que tenga configurado en el dispositivo, establecer su ubicación y el radio de búsqueda. Al iniciarse, se obtendrán las coordenadas terrestres proporcionadas por la red telefónica y se mostrará en pantalla la primera ubicación que se obtenga de la lista. Si el usuario selecciona posicionarse mediante red telefónica, podrá seleccionar otras ubicaciones alternativas.
- Seleccionar idioma, el usuario podrá seleccionar un idioma diferente al del dispositivo que se mostrará en una lista.
- Listar idiomas, se mostrarán en una lista los idiomas disponibles que el usuario podrá seleccionar.
- Establecer ubicación, permite al usuario establecer dónde se encuentra. podrá ser establecida de forma manual o automática a través del dispositivo. Una vez establecida (ya sea manual o automática) se listarán aquellas ubicaciones que coincidan con el criterio seleccionado; si es manual, se listarán aquellas con el nombre del lugar o la dirección introducida y si es automática, aquellas ubicaciones que aparezcan en función de las coordenadas terrestres obtenidas a través de los servicios de ubicación del dispositivo.
- Introducir ubicación manualmente, si el usuario opta por este método, deberá introducir una dirección, un lugar o similar y pulsará un botón que permitirá listar las ubicaciones que coincidan con el nombre de la dirección o lugar introducidos.
- Listar ubicaciones, se listarán todas las ubicaciones que coincidan con la dirección o lugar introducida por el usuario.
- Seleccionar ubicación, permitirá seleccionar aquella ubicación de la lista de ubicaciones que el usuario considere que se aproxima a su posición.
- Obtener posición automáticamente, permitirá utilizar los servicios de ubicación del dispositivo, concretamente del sistema GPS o de la red telefónica. Se obtendrán las coordenadas terrestres de la posición del usuario y se listarán las ubicaciones que sean provistas por los servicios de ubicación en función de las coordenadas terrestres obtenidas.
- Obtener posición red, los servicios de ubicación proveerán las coordenadas terrestres que serán facilitados a través de la red telefónica. Una vez obtenidas se listarán aquellas ubicaciones o lugares que se encuentren en la zona y el usuario podrá seleccionar aquel que más le convenga.
- Obtener posición GPS, funcionará igual que el anterior solo que las coordenadas terrestres serán provistas por el dispositivo GPS controlado por los servicios de ubicación del dispositivo.
- Establecer el radio de búsqueda, permitirá ampliar el margen de búsqueda tanto en el origen como en el destino.
- Añadir ruta favorita, esto permitirá al usuario introducir aquellos trayectos que le interesen estableciendo el punto de origen y destino. Eso le proporcionará recibir notificaciones de conductores que hayan establecido rutas que coincidan con las favoritas dentro del radio de búsqueda establecido. Opcionalmente podrá acceder al asistente del conductor o bien iniciar la aplicación.
- Establecer punto de origen, en principio el punto de origen es la posición del usuario pero se podrá cambiar estableciendo una ubicación manual alternativa que podrá ser seleccionada.
- Establecer punto de destino, de la misma forma que el origen, seleccionará la ubicación de destino.
- Añadir vehículo, permite al conductor añadir su vehículo. Podrá añadir una foto, seleccionar la marca y el modelo de su coche y el número de plazas que comparte. También podrá especificar si en su vehículo permite fumar, mascotas o escuchar música (o la radio).

- Añadir marca de vehículo, a través de esta funcionalidad se podrá especificar la marca del vehículo a través de una lista de marcas disponibles.
- Listar marcas de vehículos, se visualizarán las marcas de vehículos que dispone el sistema.
- Seleccionar marca, permite que el conductor elija la marca de su vehículo de la lista de marcas de vehículos.
- Añadir modelo de vehículo, se podrá especificar el modelo de vehículo del conducto de la marca previamente seleccionada.
- Listar modelos de marca de vehículo, se listarán los modelos de vehículo que soporta el sistema de la marca seleccionada.
- Seleccionar modelo, el conductor seleccionará el modelo de su vehículo de la lista.
- Añadir foto del vehículo, podrá opcionalmente agregar una foto del vehículo.
- Añadir rutas favoritas, el conductor a través del asistente puede añadir una o varias rutas preferidas indicando su punto de origen, su destino, el precio de la plaza disponible y la frecuencia.
- Establecer punto de origen, el punto de origen puede ser su ubicación (si ésta es manual) como por ejemplo si es su domicilio o un punto de encuentro que haya establecido como base para quedar con más personas. Si opta por otro lugar, podrá introducir el nombre de la ubicación y seleccionar el lugar que más le convenga de una lista.
- Establecer punto de destino, similar al anterior pero sin embargo no tiene opción de poder establecer su destino como su ubicación.
- Establecer la frecuencia del viaje, permite establecer el periodo o la regularidad del viaje estableciendo las fechas en las que viajará, la frecuencia con la que viaja; semanal o diaria y si es semanal, podrá especificar los días (de lunes a domingo).
- Listar ubicaciones, se listarán las ubicaciones que coincidan con el nombre de la ubicación introducida por el conductor.
- Seleccionar ubicación, el conductor seleccionará una de las ubicaciones que aparezcan en la lista.
- Mostrar error, cualquier operación que requiera una consulta al servidor de bases de datos y se produzca un error durante las operaciones que lo requieran se mostrará un error en pantalla.

5.3.1. FUNCIONALIDADES DE LA APLICACIÓN

A continuación se describirán las funcionalidades de la aplicación:

- Iniciar aplicación, pese a que esta funcionalidad es obvia, tiene sentido identificarla, puesto que en el inicio de la aplicación se debe comprobar el dispositivo del usuario.
- Comprobar dispositivo del usuario, permitirá conocer si el usuario que inicia la aplicación es un nuevo usuario o es ya un usuario existente. Si es un usuario existente, debe comprobar adicionalmente si su dispositivo se encuentra dado de alta en el sistema, de lo contrario, se iniciará el asistente de verificación del dispositivo. Esta comprobación se realizará contrastando el identificador del dispositivo con la que existe en la cuenta del usuario en el servidor. Si coinciden, la aplicación continuará su ejecución normal mostrando las rutas disponibles en base a sus rutas favoritas. El usuario que inicie la aplicación por primera vez no dispone de información en el dispositivo, por lo que se iniciará el asistente de registro de usuario.

- Visualizar rutas conductores, se mostrarán las rutas ofertadas por los conductores (los viajes que realizarán los conductores en una lista) que coincidan con las rutas favoritas del usuario (registrado) teniendo en cuenta su vigencia, es decir, aquellas rutas que sigan vigentes en el día que se muestran, es decir, que al menos no hayan tenido lugar y por tanto estén disponibles. Si están en curso ya no se mostrarán. Se mostrará en la cabecera de la lista la fecha (día, mes y año) y el usuario podrá cambiar la fecha de consulta para poder apuntarse cualquier otro día que le pueda convenir. El usuario podrá apuntarse a un viaje y el sistema enviará una notificación al conductor. Por defecto se muestran los viajes disponibles a partir de la fecha del sistema. Debe considerarse como la pantalla principal de la aplicación y desde ella se podrán (a través de opciones de menús) gestionar los viajes de los usuarios, sus mensajes, sus rutas favoritas, cambiar sus preferencias (entre otras opciones podrán compartir su coche si no lo realizaron al principio mediante el asistente de nuevo conductor). El usuario podrá recibir notificaciones de nuevas rutas que vayan incorporando los conductores y coincidan con sus favoritas. Opcionalmente podrá buscar otras rutas que no dependan de su ubicación aprovechando la funcionalidad *Buscar rutas ofertadas*.
- Visualizar detalles ruta, permitirá al usuario conocer los detalles de la ruta y las valoraciones sobre el conductor.
- Mostrar mapa de ofertas, permitirá visualizar las ofertas en un mapa, mostrando el origen de las rutas.
- Modificar fecha de consulta, permitirá al usuario (registrado) poder cambiar la fecha de la consulta de aquellas rutas disponibles que coincidan con sus rutas favoritas a partir de esa fecha.
- Apuntarse a un viaje, permitirá al usuario apuntarse a un viaje seleccionándolo de la lista de viajes disponibles. Esto conllevará que el conductor sea notificado mediante el sistema de notificaciones de que el usuario se ha apuntado al viaje.
- Gestionar viajes, permitirá al usuario desapuntarse de un viaje y consultar sus viajes en los que se ha apuntado así como aquellos en los que ya ha realizado y por tanto podrá valorar al conductor.
- Consultar viajes apuntados, podrá consultar la relación de viajes en los que se ha apuntado y aún no ha realizado.
- Desapuntarse de un viaje, conlleva desapuntarse por parte de un usuario ya apuntado e informará al conductor mediante una notificación a través del sistema de notificaciones. Para desapuntarse deberá consultar sus viajes en los que se ha apuntado y seleccionar la opción correspondiente.
- Consultar viajes realizados, Podrá consultar los viajes en los que ha participado y por tanto tendrá opción a comentar y valorar al conductor del viaje que seleccione.
- Gestionar mensajes, mediante el sistema de mensajería el usuario se podrá poner en contacto con el conductor enviándole un mensaje para hacerle cualquier pregunta o explicarle los motivos por los que no pudo asistir (si el conductor le pregunta) o simplemente tiene que hacerle alguna consulta.
- Enviar mensajes, tanto el usuario como el conductor podrán enviarse un mensaje.
- Leer mensajes, el usuario podrá leer sus mensajes que haya intercambiado con los conductores. Opcionalmente podrá contestar a los mensajes.
- Recibir mensajes, el sistema de mensajería comprobará si el usuario tiene mensajes pendientes de recibir y si los tiene, será notificado por el sistema de notificaciones y opcionalmente podrá leerlos.
- Responder a mensaje, el usuario (registrado o conductor) podrá responder a un mensaje que consulte y al hacerlo lo enviará a su remitente (el conductor)
- Notificar recepción mensaje, el sistema de notificaciones avisará al usuario cuando le llegue un mensaje de un conductor a través del sistema de mensajería y opcionalmente podrá consultarlo.

- Cambiar preferencias, esta funcionalidad está descrita en el asistente de nuevo usuario, concretamente *Establecer las preferencias del usuario*, con la diferencia de que el usuario al acceder, se cargarán sus preferencias y en vez de pasar al paso siguiente podrá
- Cargar preferencias usuario, esta funcionalidad permitirá obtener las preferencias del usuario. Puede iniciar el asistente de nuevo conductor para añadir su vehículo y agregar sus rutas (si es un usuario registrado que no tiene vehículo). Si se trata de un conductor, adicionalmente podrá modificar los datos de su vehículo.
- Iniciar asistente de nuevo conductor, dentro de las preferencias podrá iniciar el asistente de nuevo conductor e incorporar su vehículo y sus rutas.
- Modificar vehículo, permite al conductor modificar la foto, la marca y el modelo de vehículo. Se podrá realizar desde la funcionalidad de cambiar preferencias. Para ello cargará los datos del vehículo para ser modificados. Empleará la funcionalidad de *Añadir vehículo* utilizada en el asistente de nuevo conductor
- Cargar datos vehículo, obtendrá los datos del vehículo del conductor si el usuario opta por modificar los datos de su vehículo.
- Gestionar rutas favoritas, el usuario, al acceder a esta funcionalidad, podrá obtener un listado de todas sus rutas favoritas. Opcionalmente podrá eliminar o modificar una existente. También podrá dar de alta una nueva.
- Añadir ruta favorita, permitirá al usuario establecer una ruta de su conveniencia. Esto conllevará que por cada ruta favorita que agregue podrá recibir notificaciones de rutas nuevas creadas por los conductores que se encuentren en el radio de búsqueda del origen y del destino. El lugar de origen de búsqueda corresponde a la posición actual del usuario, que puede ser fija (establecida manualmente) o automática, obtenida por los servicios de ubicación del dispositivo. Si está establecida en manual, podrá recibir notificaciones de nuevas rutas disponibles que se encuentren en el radio de búsqueda establecido (tanto en origen como destino).
- Obtener posición, al añadir una ruta favorita, se obtendrá la posición del usuario. Ésta corresponderá a la que tenga configurada en sus preferencias, que puede ser establecida manualmente y ser fija, o bien puede ser automática y variable a través de los servicios de ubicación.
- Eliminar ruta favorita, el usuario puede eliminar una ruta de sus favoritas.
- Buscar rutas ofertadas, el usuario puede alternativamente buscar una o varias rutas ofertadas por los conductores entre dos puntos diferentes (y su origen diferente a su ubicación) y puede opcionalmente apuntarse. Estas rutas serán facilitadas por el servidor.
- Establecer punto de origen, permite establecer el punto de origen de la ruta, tanto por parte de un usuario a la hora de buscar ofertas de conductores de una ruta en particular como al publicar una ruta ofertada por un conductor o simplemente a la hora de añadir una ruta.
- Establecer punto de destino, permite establecer el punto de destino de un usuario en desplazarse a ese punto tanto desde su ubicación como desde otro punto distinto, así como en la funcionalidad anterior
- Recibir notificación nueva ruta disponible, permitirá dar a conocer al usuario de aquellas rutas nuevas disponibles ofertadas por los conductores que se ajusten a las rutas favoritas del usuario (teniendo en cuenta sus preferencias). El usuario será notificado por el sistema de notificaciones.
- Listar ubicaciones, permitirá listar las ubicaciones que coincidan con el criterio establecido por el usuario

- Seleccionar ubicación, como puede devolver más de una, el usuario podrá elegir la que más le convenga, que internamente contendrá las coordenadas terrestres del lugar seleccionado.
- Gestionar viajes, el conductor podrá añadir sus trayectos en las fechas que tenga previstas, establecer sus características y publicarlos. También podrá modificar las fechas o cancelar el viaje.
- Cancelar viaje, si se cancela el viaje y existen personas apuntadas, se enviará un mensaje automáticamente a las personas para informarles el motivo de la cancelación y permitir así dar respuesta a los usuarios afectados. El conductor también podrá consultar sus viajes (o rutas) planificados.
- Modificar fechas, permitirá trasladar una fecha de un viaje a otro día sólo si es puntual, lo que implicará la cancelación de un viaje en el que los interesados ya se han apuntado y deberá informar del motivo por el que modifica la fecha.
- Publicar viajes, permite ofertar una nueva ruta por parte del conductor a los usuarios interesados y será registrada en el servidor. El usuario establecerá los puntos de origen y destino del trayecto; la frecuencia, es decir, si lo realiza puntualmente o diariamente/semanalmente entre un intervalo de fechas; si es semanal, puede establecer los días de la semana en los que va a viajar; además puede establecer el precio de la plaza que oferta

5.4. REQUISITOS NO FUNCIONALES DEL SISTEMA

A continuación vamos a describir las restricciones del sistema sin entrar todavía en los detalles tecnológicos y del entorno.

5.4.1. Requisitos de Arquitectura

- Los datos de la aplicación se almacenarán en un sistema gestor de bases de datos en el servidor a través de un servicio web.
- Cada operación que sea necesaria realizar y que tenga que ver con actividades de la base de datos del servidor será realizada mediante solicitudes a través del protocolo http.

5.4.2. Requisitos de Seguridad

- El usuario sólo puede darse de alta a través de su teléfono móvil.
- La única vía que tiene el usuario para acceder al sistema es registrándose como usuario mediante el asistente de registro en el que entre otros datos, se obtendrá su identificador de dispositivo para realizar futuras comprobaciones.
- Cada vez que el usuario inicie la aplicación, se obtendrá su identificador de dispositivo y se comparará con el que existe en el sistema y si son diferentes, se iniciará el sistema de verificación de dispositivo cuyo fin será el de actualizar su dispositivo de forma segura. Esto puede suceder cuando un usuario cambia de dispositivo, instala la aplicación y ha copiado la carpeta de datos de la aplicación en su nuevo dispositivo.
- Los datos que se envíen al servidor y se reciban serán cifrados y no será posible acceder a la base de datos directamente.

5.4.3. Requisitos de Disponibilidad

La aplicación estará sólo disponible para teléfonos móviles, es decir, aquellos dispositivos que disponen de un identificador IMEI. La aplicación no estará disponible para tablets. Se tiene previsto que la aplicación no esté disponible si se produce un error de conexión (o no existe conexión) a Internet. Esto es debido a que al iniciarse la aplicación, necesita comprobar el dispositivo, por requisitos de seguridad.

En este apartado vamos a describir el comportamiento del sistema en caso de que no exista conexión.

- Durante el proceso de verificación de usuario, al recibir la confirmación por SMS en el dispositivo, se procederá a mostrar el error en pantalla y la única operación disponible será aceptar la situación y salir. Esto permitirá que al volver a iniciar la aplicación, se pueda iniciar el asistente en la pantalla de introducir sus datos personales pero previamente habrá creado la cuenta del usuario. Durante este proceso existirá un tiempo de espera (timeout) que delimitará el tiempo de respuesta de recepción del SMS y si se supera, se podrá reintentar sin restricciones.
- Al iniciar la aplicación, si se produce un error de conexión o el usuario no se encuentra conectado, la aplicación no estará disponible mostrándolo en pantalla.
- Durante cualquier fase del asistente de registro de usuario, si se produce un error deberá indicar el motivo del error, dejando al usuario la única opción de aceptar y salir. Será por tanto necesario almacenar en un soporte de fichero la fase del asistente en que se encuentra con objeto de iniciar la aplicación en el punto donde se había quedado sin completar.
- Sin embargo, si se produce un error en el asistente del conductor (el usuario ha optado por agregar su vehículo a continuación de finalizar el asistente) no se realizará ninguna comprobación y directamente se accederá a la pantalla principal.

5.4.4. Requisitos de Información

La información que deberá tenerse en cuenta por parte del sistema es la siguiente:

5.4.4.1. Información del usuario

- Identificador de usuario, será el número de teléfono
- Identificador de dispositivo, será la dirección MAC del dispositivo. Permitirá conocer si ha cambiado de dispositivo.
- Fotografía del usuario

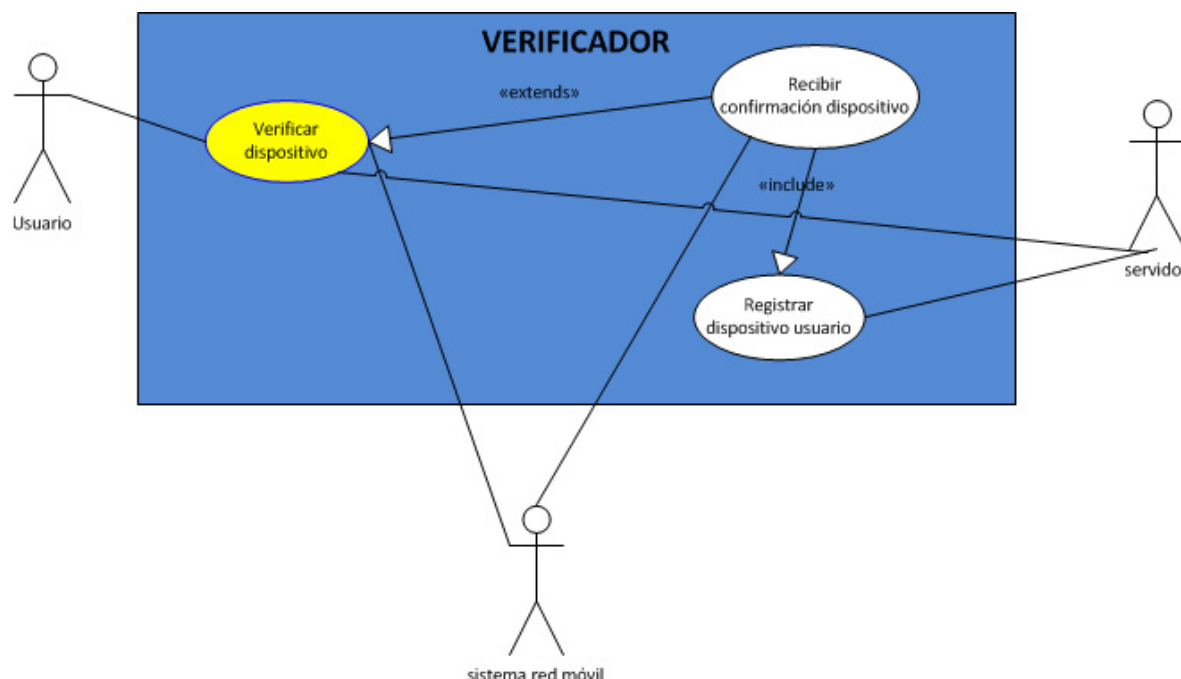
5.4.4.2. Información del vehículo

- Marca y modelo de coche, serán los identificadores propios que denominen la marca y el modelo del vehículo.
- Indicadores que establezcan si se puede fumar, se admiten animales o se puede escuchar música (o la radio).
- Fotografía del vehículo

5.4.4.3. Información del mensaje

- El código de mensaje será único y posiblemente una combinación entre la fecha del sistema y el usuario que lo envía.

El sistema se apoyará en un subsistema que tendrá como funcionalidad principal la de verificar el dispositivo del usuario, lo que permitirá comportarse de forma distinta en función del resultado de la verificación:



De los diagramas expuestos podemos pasar a detallar qué actores intervienen en el sistema y qué roles van a desempeñar en él y los distintos casos de uso.

6.1. Actores

- Usuario, es aquella persona que inicia la aplicación. Si la inicia por primera vez, se iniciará el asistente de registro de usuario que permitirá dar de alta al usuario en el sistema y configurar la aplicación en un instante. Adicionalmente el usuario tendrá que verificar su dispositivo -a través del sistema de red móvil mediante la recepción de un SMS- si éste ha cambiado de terminal y ha realizado una copia de los datos en el nuevo dispositivo. Los datos del usuario se almacenarán adicionalmente en el servidor.
- Usuario registrado, es aquel usuario que ha verificado su dispositivo correctamente y aún no se ha apuntado a ningún viaje ofrecido por un conductor. Puede ponerse en contacto con un conductor a través de la aplicación mediante el sistema de mensajería. También podrá cambiar sus preferencias en todo momento y puede añadir su vehículo por si en un momento dado desea compartir su coche en un viaje. También podrá gestionar sus rutas favoritas para poder recibir notificaciones sobre nuevas ofertas de viajes cuyos trayectos coincidan con sus favoritas. Podrá apuntarse a un viaje (ofertado en base a sus rutas favoritas) del conductor que más le interese. Opcionalmente podrá buscar otros viajes ofertados por otros conductores que ofrezcan otras rutas diferentes a sus favoritas.
- Pasajero, es aquel usuario registrado que al menos se ha apuntado una vez a un viaje, lo cual le permitirá gestionar sus viajes a través de la aplicación. Podrá consultar sus viajes realizados en los que adicionalmente le permitirá valorar al conductor con el que han viajado y también los viajes pendientes de realizar a los que opcionalmente se puede desapuntar lo que conllevará notificar al conductor indicando los motivos de la cancelación.

- Conductor, ofrece un viaje con su vehículo entre dos puntos en una fecha determinada y lo hará público. Podrá cancelar un viaje de una fecha concreta y si existen pasajeros apuntados, se enviará una notificación de la cancelación explicando los motivos. Opcionalmente podrá activar el seguimiento de su viaje en el momento de iniciarse para que familiares o allegados puedan consultar por donde van.
- Sistema red móvil, proveerá al sistema del servicio SMS para la verificación
- Servidor, donde se almacenarán los datos del sistema necesarios para su correcto funcionamiento.
- Sistema de notificación, permitirá informar a los usuarios de nuevas rutas ofrecidas por los conductores y que coincidan con sus favoritas. También notificarán a los usuarios de las posibles cancelaciones que se puedan producir tanto por parte de un conductor como de los pasajeros.
- Servicios de ubicación, proveerán de la información necesaria de geolocalización ya sea a través de servicios web como servicios ofrecidos por el dispositivo (mediante red, GPS).

6.2. Inventario de casos de uso por sistema y actor

A continuación vamos a enumerar la relación de casos de uso de cada sistema y los actores que intervienen. Las abreviaturas que aparecen en el inventario corresponden a los siguientes sistemas:

- SYS_VRF: Sistema verificador de dispositivo.
- APP: Es el sistema principal o aplicación.

Código	Descripción	Sistema	Actores
UCSYS_VRFU01	Verificar dispositivo	SYS_VRF	Usuario, Sistema red móvil, Servidor
UCSYS_VRFU02	Recibir confirmación dispositivo	SYS_VRF	Sistema red móvil
UCSYS_VRFU03	Registrar dispositivo usuario	SYS_VRF	Servidor
UCAPP01	Mostrar asistente	APP	Usuario registrado, Servidor
UCAPP02	Establecer preferencias	APP	Usuario registrado
UCAPP03	Establecer vehículo	APP	Usuario registrado, Servidor
UCAPP04	Gestionar rutas favoritas	APP	Usuario registrado
UCAPP05	Agregar ruta favorita	APP	Usuario registrado
UCAPP06	Eliminar ruta favorita	APP	Usuario registrado
UCAPP07	Recibir notificaciones	APP	Usuario registrado, Sistema de notificaciones, Servidor
UCAPP08	Visualizar viajes disponibles	APP	Usuario registrado, Servidor
UCAPP09	Enviar mensaje	APP	Usuario registrado, Servidor
UCAPP10	Buscar viajes ofertados	APP	Usuario registrado, Servidor, Servicios de Ubicación
UCAPP11	Apuntarse a un viaje	APP	Usuario registrado, Servidor
UCAPP12	Gestionar mensajes	APP	Usuario registrado, Sistema de notificaciones
UCAPP13	Enviar notificación	APP	Usuario registrado, Conductor, Servidor, Sistema de notificaciones
UCAPP14	Gestionar viajes	APP	Pasajero, Servidor
UCAPP15	Valorar conductor	APP	Pasajero, Servidor
UCAPP16	Desapuntar viaje	APP	Pasajero, Servidor
UCAPP17	Organizar viajes	APP	Conductor, Servidor
UCAPP18	Publicar viaje	APP	Conductor, Servidor

UCAPP19	Gestionar seguimiento	APP	Conductor, Servidor, Servicios de ubicación.
UCAPP20	Cancelar viaje	APP	Conductor, Servidor
UCAPP21	Consultar viajes realizados	APP	Conductor

6.3. Especificación de casos de uso

6.3.1. Verificar dispositivo

Identificador	UCSYS_VRFU01
Nombre	Verificar dispositivo
Resumen	Identifica el dispositivo del usuario para poder acceder al sistema
Actor(es)	Usuario, Sistema red móvil, Servidor
Precondiciones	Nuevo usuario, usuario registrado en el sistema o usuario validado pero no registrado.
Postcondiciones	El usuario está dado de alta en el sistema. El dispositivo es válido o ha sido verificado correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. El caso de uso comienza cuando el usuario inicia la aplicación 2. El sistema validará el dispositivo en el sistema 3. Una vez validado el dispositivo, se guardarán los cambios y finalizará el caso de uso.
Flujos alternativos	<ul style="list-style-type: none"> • Nuevo usuario: <ul style="list-style-type: none"> - En el paso 2, si el usuario no existe en el sistema se pedirá en pantalla que introduzca su número de teléfono para validar su dispositivo. - Una vez introducido el nº de teléfono (el cual servirá para identificar al usuario en el sistema) podrá solicitar la verificación y quedará a la espera de recibir la confirmación de que su dispositivo es válido. - Existirá un tiempo de espera para recibir la confirmación. Si se supera, se mostrará en pantalla dicha situación permitiendo al usuario reintentar la verificación del dispositivo. - Alternativamente, el usuario puede cancelar la verificación lo que conllevará la finalización del caso de uso. - En el paso 3, se ha recibido confirmación del dispositivo y se extenderá el caso de uso UCAPP01 (Mostrar asistente). • Usuario registrado, nuevo dispositivo: <ul style="list-style-type: none"> - Se producen los mismos pasos que el usuario no registrado pero con la diferencia que los datos del usuario existen en el sistema y sin embargo los del dispositivo no y debe ser verificado. - En el paso 3, se ha recibido confirmación del nuevo dispositivo y se extenderá el caso de uso UCAPP10 (Visualizar viajes disponibles). • Usuario no registrado pero validado (verificado su nº de teléfono): <ul style="list-style-type: none"> - En el paso 2, puede ocurrir también que en una verificación previa se haya producido un error y por tanto, el dispositivo se ha verificado pero no se ha registrado correctamente en el sistema. Por tanto en este punto no será necesario verificar el número de teléfono del usuario y por consiguiente, se enviarán los datos al servidor para registrar el dispositivo del usuario, por lo que extenderá el caso de uso UCSYS_VRFU03 (Registrar dispositivo usuario).

	<ul style="list-style-type: none"> • Usuario registrado pero no ha completado el asistente: <ul style="list-style-type: none"> - En el paso 2, el usuario registrado no ha finalizado el asistente donde debe completar sus datos personales y sus preferencias por lo que iniciará el caso de uso UCAPP01 (Mostrar asistente).
Excepciones	Si se produce algún error a la hora de comprobar el dispositivo en el sistema (al acceder al servidor, se produce un error) se mostrará el mensaje de error correspondiente y finalizará el caso de uso.
Extensiones	Extiende a UCSYS_VRFU02 (Recibir confirmación dispositivo), UCSYS_VRFU03 (Registrar dispositivo usuario), UCAPP01 (Mostrar asistente) y UCAPP10 (Visualizar viajes disponibles).

6.3.2. Recibir confirmación de dispositivo

Identificador	UCSYS_VRFU02
Nombre	Recibir confirmación dispositivo
Resumen	El sistema queda a la espera de la confirmación.
Actor(es)	Sistema red móvil
Precondiciones	El usuario debe haber iniciado el caso de uso UCSYS_VRFU01 (Verificar dispositivo).
Postcondiciones	Validación de dispositivo correcta.
Flujo normal	<ol style="list-style-type: none"> 1. Se recibe la confirmación del dispositivo 2. Se inicia el caso de uso UCSYS_VRFU03 (Registrar dispositivo usuario)
Inclusiones	UCSYS_VRFU03 (Registrar dispositivo usuario)

6.3.3. Registrar dispositivo usuario

Identificador	UCSYS_VRFU03
Nombre	Registrar dispositivo usuario.
Resumen	Se almacenará la información del usuario en el sistema para identificaciones futuras.
Actor(es)	Servidor
Precondiciones	Se debe haber iniciado el caso de uso UCSYS_VRFU02 , es decir, debe haber recibido confirmación de que el dispositivo es válido.
Postcondiciones	Los datos del usuario han sido almacenados en el sistema correctamente
Flujo normal	<ol style="list-style-type: none"> 1. Se almacenará el identificador de usuario en el dispositivo 2. Se almacenará el identificador de usuario y dispositivo en el servidor
Excepciones	Puede ocurrir un error al registrar los datos en el servidor. En esta situación no es posible continuar por lo que finalizará el caso de uso mostrando la causa del error y finalizará su ejecución; pero se indicará al sistema que el registro queda pendiente.

6.3.4. Mostrar asistente

Identificador	UCAPP01
Nombre	Mostrar asistente
Resumen	Se recopilará la información del usuario registrado sobre sus datos personales y sus preferencias.
Actor(es)	Usuario registrado, Servidor
Precondiciones	Se debe haber iniciado el caso de uso UCSYS_VRFU01 (Verificar dispositivo). No existen datos personales del usuario registrado ni sus preferencias.
Postcondiciones	Los datos personales del usuario y sus preferencias se han almacenado en el sistema.
Flujo normal	<ol style="list-style-type: none"> 1. Se solicitará en pantalla que el usuario introduzca sus datos personales (tales como una foto, nombre completo edad y sexo). 2. Al pulsar la opción correspondiente, se validarán los datos personales. 3. Se almacenarán los datos personales en el sistema. 4. Se iniciará el caso de uso UCAPP02 (Establecer preferencias)
Flujos alternativos	<ul style="list-style-type: none"> • Validación datos personales: - En el paso 2, al pulsar la opción correspondiente para pasar al siguiente paso, se comprobarán que los datos que se consideren obligatorios hayan sido introducidos. Si no lo han sido, no se podrá iniciar el caso de uso UCAPP02 y se volverá al paso 1.
Excepciones	Si se produce un error en el paso 3, se mostrará en pantalla notificando la situación de error y volverá al paso 1 (previa aceptación de la situación por parte del usuario).
Inclusiones	UCAPP02 (Establecer preferencias)
Extensiones	UCSYS_VRFU01 (Verificar dispositivo)

6.3.5. Establecer preferencias

Identificador	UCAPP02
Nombre	Establecer preferencias
Resumen	El usuario podrá cambiar las preferencias tales como establecer su posición de forma manual o automática, el radio de cobertura para buscar nuevas rutas basadas en su posición y establecer las características de su vehículo.
Actor(es)	Usuario registrado
Precondiciones	Ser iniciado por el usuario o a través del asistente, mediante el caso de uso UCAPP01 (Mostrar Asistente).
Postcondiciones	Las preferencias del usuario se han establecido correctamente
Flujo normal	<ol style="list-style-type: none"> 1. Se obtendrán las preferencias del usuario 2. Se mostrará en una parte de la pantalla la ubicación del usuario en base a sus preferencias. 3. Se mostrará también qué método de posicionamiento se está utilizando (manual o automático). 4. Se mostrarán los radios de búsqueda respecto del origen y destino de un trayecto o ruta. 5. El usuario podrá cambiar el tipo de posicionamiento. Como opciones de posicionamiento estarán disponibles la opción de posicionamiento manual y automático. Se apoyará en servicios de ubicación provistos por el sistema para su posicionamiento correcto. 6. El usuario podrá cambiar los radios de búsqueda. 7. Existirá la opción de añadir un vehículo que extenderá el caso de uso UCAPP03 (Establecer vehículo). 8. Existirá la opción de añadir trayectos (o rutas) favoritos
Flujos alternativos	<ul style="list-style-type: none"> • Desde el asistente: <ul style="list-style-type: none"> - En el paso 1, se obtendrán las preferencias por defecto - En el paso 2, se obtendrá la ubicación de forma automática. - Si el usuario opta por finalizar el caso de uso, se extenderá automáticamente el caso de uso UCAPP04 (Gestionar rutas favoritas). • Desde la aplicación: <ul style="list-style-type: none"> - En el paso 2, se obtendrá la ubicación de la forma que tenga establecida el usuario. Si la tiene establecida en manual, habilitará la opción de cambiar la ubicación donde se habilitará un espacio donde el usuario pueda introducir su ubicación. Igualmente ocurrirá con la automática. • En ambos: <ul style="list-style-type: none"> - En el paso 5, si el usuario selecciona manual, se habilitará una opción para que pueda cambiar la ubicación manualmente. Si selecciona automática, podrá adicionalmente seleccionar si prefiere GPS o red para su posicionamiento.

Excepciones	Se podría producir un error relacionado con el envío de información al servidor. Si se produce tal circunstancia, deberá informarse al usuario de esa situación.
Extensiones	UCAPP03 (Establecer vehículo), UCAPP04 (Gestionar rutas favoritas)

6.3.6. Establecer vehículo

Identificador	UCAPP03
Nombre	Establecer vehículo
Resumen	Permite al usuario especificar las características de su vehículo y sus restricciones a la hora de viajar. También permite cambiar de vehículo.
Actor(es)	Usuario registrado, Servidor
Precondiciones	Debe ser iniciado por el caso de uso UCAPP02 (Establecer preferencias)
Postcondiciones	El vehículo se ha incorporado al sistema correctamente.
Flujo normal	<p>Al iniciarse el caso de uso se cargarán los datos del vehículo.</p> <p>En pantalla el usuario podrá optar por:</p> <ol style="list-style-type: none"> 1. Incorporar la foto de su vehículo 2. Seleccionar la marca presionando el espacio correspondiente y seleccionar de una lista. 3. Seleccionar el modelo de la marca seleccionada previamente, presionando el espacio correspondiente y seleccionar de una lista. 4. Especificar el nº de plazas máximo a compartir. 5. Especificar sus restricciones, es decir, si permite fumar, animales o si pone música mientras viaja. 6. El usuario opta por salir y finaliza el caso de uso.
Flujos alternativos	<p>Si no existen los datos del vehículo se mostrará la pantalla predeterminada.</p> <p>Al salir, si el usuario ha incorporado parcialmente los datos del vehículo será notificado de tal situación. Puede optar por cancelar e incorporar el resto de datos o bien; podrá continuar. Si el usuario ha incorporado los datos totalmente; al salir, serán enviados al servidor.</p>
Excepciones	Se podría producir un error relacionado con el envío de información al servidor. Si se produce tal circunstancia, deberá informarse al usuario de esa situación.

6.3.7. Gestionar rutas favoritas

Identificador	UCAPP04
Nombre	Gestionar rutas favoritas
Resumen	Permitirá al usuario conocer cuáles son sus rutas o trayectos favoritos. Opcionalmente recibirá notificaciones de viajes disponibles que coincidan con sus rutas favoritas.
Actor(es)	Usuario registrado
Precondiciones	Debe ser iniciado por el caso de uso UCAPP02 (Establecer preferencias)
Postcondiciones	Se han listado las rutas favoritas del usuario correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. Se mostrarán las rutas favoritas del usuario 2. El usuario podrá optar por agregar una ruta favorita extendiendo el caso de uso UCAPP05 (Agregar ruta favorita) 3. Opcionalmente podrá eliminar una ruta de sus favoritas extendiendo el caso de uso UCAPP06 (Eliminar ruta favorita). 4. Recibirá opcionalmente notificaciones de nuevos viajes disponibles por parte del sistema de notificaciones y extenderá el caso de uso UCAPP07 (Recibir notificaciones). 5. Si el usuario opta por salir,
Flujos alternativos	Tanto si se opta por el paso 2 como por el paso 3, al finalizar, se volverá al paso 1. Si opta por el paso 2 y agrega una nueva ruta
Extensiones	UCAPP05 (Agregar ruta favorita), UCAPP06 (Cancelar ruta favorita), UCAPP07 (Recibir notificaciones)

6.3.8. Agregar ruta favorita

Identificador	UCAPP05
Nombre	Agregar ruta favorita
Resumen	Permite al usuario definir una nueva ruta favorita. Por cada ruta favorita que agregue recibirá una notificación de viaje disponible que coincida con su criterio y sus preferencias de búsqueda.
Actor(es)	Usuario registrado
Precondiciones	Debe haber sido iniciado por el caso de uso UCAPP04 (Gestionar rutas favoritas)
Postcondiciones	La nueva ruta ha sido dada de alta correctamente en el sistema y está listo para recibir notificaciones de viajes disponibles con esa ruta.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario seleccionará un punto de origen. Por defecto será su posición actual. 2. Opcionalmente su posición de origen puede ser otra diferente que no sea su posición actual y podrá escribir en pantalla su nombre y podrán mostrarse aquellas localizaciones que coincidan con la que ha introducido y podrá seleccionar una de ellas. 3. El usuario introduce un punto de destino. Se comportará de la misma forma que en punto anterior. 4. El usuario pulsa la opción de agregar y finaliza el caso de uso.
Flujos alternativos	<p>Si el usuario opta por el paso 4, deberá validarse lo siguiente:</p> <ul style="list-style-type: none"> - Si ha seleccionado su posición actual, el sistema deberá comprobar que se ha introducido un destino. Si no ha sido introducido, se mostrará en pantalla un aviso de que falta por introducir el destino. - Si ha seleccionado otra posición diferente a la actual deberá comprobar que ha introducido el punto de origen. Si no ha sido introducido, se mostrará en pantalla un aviso de que falta por introducirlo.
Extensión	UCAPP07 (Recibir notificaciones)

6.3.9. Eliminar ruta favorita

Identificador	UCAPP06
Nombre	Eliminar ruta favorita
Resumen	El usuario podrá eliminar una ruta de sus favoritas.
Actor(es)	Usuario registrado
Precondiciones	Debe haberse iniciado desde el caso de uso UCAPP04 (Gestionar rutas favoritas)
Postcondiciones	La ruta se ha eliminado del sistema correctamente
Flujo normal	<ol style="list-style-type: none"> 1. El usuario ha seleccionado previamente una ruta de sus favoritas a través del caso de uso UCAPP04. 2. Se le pedirá confirmación en pantalla de eliminación. Puede optar por aceptar o cancelar la operación. 3. Finaliza el caso de uso.
Flujos alternativos	En el paso 2, si el usuario opta por aceptar, se eliminará la ruta favorita del usuario del sistema.

6.3.10. Recibir notificaciones

Identificador	UCAPP07
Nombre	Recibir notificaciones
Resumen	El sistema notificará al usuario de nuevos viajes disponibles. También será notificado un pasajero si un conductor cancela el viaje al que se ha apuntado al igual que un conductor recibirá notificaciones de los pasajeros que se hayan desapuntado de uno de sus viajes planificados. Por último, notificará al usuario de la recepción de nuevos mensajes.
Actor(es)	Usuario registrado, Sistema de notificaciones, Servidor
Precondiciones	Debe existir conexión a Internet
Postcondiciones	Enviar un mensaje (contestación, si se trata de la recepción de un mensaje), consultar el viaje objeto de la cancelación o bien, consultar el nuevo viaje disponible.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema informará al usuario de la existencia de : <ol style="list-style-type: none"> 1.1. Nuevo mensaje de un usuario 1.2. Nuevo viaje disponible planificado por el conductor 1.3. Cancelación de viaje planificado por el conductor 1.4. Cancelación de viaje por parte de un usuario apuntado a un viaje. 1.5. Nuevo viaje disponible. 2. El usuario consulta el contenido de la notificación iniciando el caso de uso correspondiente. Opcionalmente podrá enviar un mensaje en respuesta a la notificación de mensaje o cancelación.

	<p>3. El sistema eliminará la notificación del servidor.</p> <p>4. Finaliza el caso de uso</p>
Flujos alternativos	<p>El usuario consulta la notificación:</p> <ul style="list-style-type: none"> - Si su naturaleza corresponde al paso 1.1 y el destinatario es un usuario o pasajero el sistema iniciará el caso de uso UCAPP12 (Gestionar mensajes). - Si su naturaleza corresponde al paso 1.1 y el destinatario es un conductor iniciará el caso de uso UCAPP18 (Organizar viajes). - Si existe una cancelación de un viaje planificado por un conductor, el destinatario recibirá un mensaje con los motivos de la cancelación iniciando el caso de uso UCAPP12 (Gestionar mensajes). - Si un usuario decide desapuntarse de un viaje de un conductor (paso 1.4), el conductor recibirá un mensaje con los motivos de la cancelación y se iniciará el caso de uso UCAPP18 (Organizar viajes). - Si el usuario consulta la notificación correspondiente al paso 1.5 se iniciará el caso de uso UCAPP08 (Visualizar viajes disponibles).
Extensiones	<p>UCAPP12 (Gestionar mensajes), UCAPP18 (Organizar viajes), UCAPP08 (Visualizar viajes disponibles).</p>

6.3.11. Visualizar viajes disponibles

Identificador	UCAPP08
Nombre	Visualizar viajes disponibles
Resumen	Muestra una lista de viajes disponibles planificados por los conductores en el día y en adelante. El usuario podrá seleccionar uno de su conveniencia y apuntarse. Alternativamente podrá enviar un mensaje al conductor para hacerle consultas. Adicionalmente podrá buscar otras ofertas de viajes diferentes a sus preferencias.
Actor(es)	Usuario registrado, Servidor
Precondiciones	El dispositivo debe haber sido verificado previamente y por tanto haber sido iniciado (la primera vez) por el caso de uso UCSYS_VRFU01 (Verificar dispositivo).
Postcondiciones	Se mostrarán los viajes planificados en el día, en la semana o en el mes por los conductores.
Flujo normal	<ol style="list-style-type: none"> 1. Se listarán los viajes disponibles en el día indicando el conductor, edad, sus restricciones, la fecha y la hora, los puntos de origen y destino y el número de plazas disponibles. Los viajes disponibles se almacenarán en el dispositivo con objeto de identificar los nuevos. 2. El usuario podrá seleccionar un viaje de la lista para obtener los detalles. 3. En la pantalla de detalles del viaje podrá apuntarse mediante el botón correspondiente. 4. También podrá ponerse en contacto con el conductor mediante el envío de mensajes presionando el botón correspondiente. 5. Existirá la opción de buscar viajes alternativos. 6. Si el usuario opta por salir se finalizará el caso de uso.
Flujos alternativos	<ul style="list-style-type: none"> - En el paso 1 y 2, no se listarán aquellos viajes cuyas plazas disponibles queden cubiertas. Tampoco se listarán si el viaje está en curso o ha finalizado. - Si el usuario opta por el paso 3, se iniciará el caso de uso UCAPP11 (Apuntarse a un viaje). - Si el usuario opta por el paso 4, se iniciará el caso de uso UCAPP12 (Gestionar mensajes). - Si el usuario opta por el paso 5, se iniciará el caso de uso UCAPP10 (Buscar viajes ofertados).
Extensiones	UCAPP10 (Buscar viajes ofertados), UCAPP11 (Apuntarse a un viaje), UCAPP12 (Gestionar mensajes)

6.3.12. Enviar Mensaje

Identificador	UCAPP09
Nombre	Enviar mensaje
Resumen	El usuario puede enviar un mensaje o responder a otro mensaje.
Actor(es)	Usuario registrado, Servidor
Precondiciones	Debe haber sido iniciado por el caso de uso UCAPP12 (Gestionar mensajes). El flujo de mensajes entre usuarios es de usuario (registrado, pasajero) a conductor y viceversa.
Postcondiciones	El usuario ha enviado un mensaje individual o en base a una respuesta de un mensaje recibido.
Flujo normal	<ol style="list-style-type: none"> 1. Se recopilará la información del viaje y el conductor seleccionados en el caso de uso UCAPP12 (Gestionar mensajes). 2. Se enviará la información al servidor. 3. Finalizará el caso de uso.
Excepciones	Si en el paso 2 ocurre un error se informará al usuario que la operación no se ha podido realizar, dando la opción de reintentar la operación. Si opta por reintentar, se ejecutará el paso 2.

6.3.13. Buscar viajes ofertados

Identificador	UCAPP10
Nombre	Buscar viajes ofertados
Resumen	El usuario puede enviar un mensaje o responder a otro mensaje.
Actor(es)	Usuario registrado, Servidor, Servicios de ubicación
Precondiciones	Debe haber sido iniciado por el caso de uso UCAPP08 (Visualizar viajes disponibles).
Postcondiciones	Se han obtenido los viajes disponibles entre dos puntos.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario introduce un origen y un destino del cual pretende obtener la información. 2. Al pulsar el botón de búsqueda se mostrarán todos los viajes disponibles entre esos dos puntos. 3. El resto de funcionalidades para cada uno de los resultados es equivalente al caso de uso UCAPP08 (Visualizar viajes disponibles). 4. Finaliza el caso de uso

6.3.14. Apuntarse a un viaje

Identificador	UCAPP11
Nombre	Apuntarse a un viaje
Resumen	El usuario puede apuntarse a un viaje planificado por un conductor.
Actor(es)	Usuario registrado, Servidor
Precondiciones	Debe haber sido iniciado por el caso de uso UCAPP08 (Visualizar viajes disponibles).
Postcondiciones	El usuario se ha apuntado a un viaje.
Flujo normal	<ol style="list-style-type: none"> Se recopilará la información del viaje y el conductor seleccionados en el caso de uso UCAPP08 (Visualizar viajes disponibles). Se enviará la información al servidor. Iniciará el caso de uso UCAPP13 (Enviar notificación) Finalizará el caso de uso.
Flujos alternativos	Si en el paso 2 ocurre un error saltará al paso 4.
Excepciones	Si en el paso 2 ocurre un error se informará al usuario que la operación no se ha podido realizar, dando la opción de reintentar la operación. Si opta por reintentar, se ejecutará el paso 2.
Inclusiones	UCAPP13 (Enviar notificación)

6.3.15. Gestionar mensajes

Identificador	UCAPP12
Nombre	Gestionar mensajes
Resumen	Permite consultar los mensajes que ha enviado o recibido de un usuario. Los mensajes se enviarán de usuario/pasajero a conductor. También puede responder
Actor(es)	Usuario registrado, Sistema de notificaciones
Precondiciones	<ul style="list-style-type: none"> El usuario ha enviado o recibido al menos un mensaje o, El usuario ha recibido una notificación de nuevo mensaje o, El usuario ha seleccionado un viaje disponible y ha optado por contactar con el conductor
Postcondiciones	Se han listado los mensajes del usuario correctamente
Flujo normal	<ol style="list-style-type: none"> El sistema lista todos los mensajes de un viaje en concreto (en orden cronológico) en los que haya participado el usuario encabezados por conductor y viaje indicando el número de mensajes propios. Opcionalmente muestra un nuevo mensaje de otro usuario al recibir una notificación.

	<ol style="list-style-type: none"> Opcionalmente el usuario podrá seleccionar un mensaje presionando sobre un viaje y se listarán todos los mensajes entre el usuario y el conductor que hayan intercambiado en ese viaje. En la cabecera existirá los datos mínimos del viaje y el conductor. Existirá una entrada de texto y donde podrá contestar al mensaje. Opcionalmente podrá eliminar todos los mensajes de un viaje que ya no le interese. Al presionar la opción correspondiente de salir, finalizará el caso de uso.
Flujo alternativo	<p>Recepción de mensaje:</p> <p>Si el caso de uso ha sido iniciado por el caso de uso UCAPP07 (Recibir notificaciones) se dan dos posibilidades:</p> <ul style="list-style-type: none"> Mensaje de conductor recibido Cancelación de viaje (por parte de un conductor) <p>En ambos casos se mostrará directamente el viaje sobre el que se refiere el mensaje y la lista de mensajes listados en orden cronológico.</p> <p>Envío de mensaje:</p> <ul style="list-style-type: none"> Si el usuario opta por contestar a un mensaje (paso 3) del conductor iniciará el caso de uso UCAPP09 (Enviar mensaje). Si el usuario ha decidido contactar con el conductor desde el caso de uso UCAPP08 (Visualizar viajes disponibles), se iniciará el caso de uso en el paso 3 seleccionando el viaje y conductor procedente del caso de uso UCAPP08, es decir, cargará los datos del viaje y el conductor y mostrará sus mensajes. Al introducir el texto del mensaje y pulsar el botón enviar se iniciará el caso de uso UCAPP09 (Enviar mensaje).
Exclusión	UCAPP09 (Enviar mensaje)

6.3.16. Enviar notificación

Identificador	UCAPP13
Nombre	Enviar notificación
Resumen	Tendrá lugar cuando un usuario se apunte o cancele un viaje, o bien el conductor publique o cancele un viaje.
Actor(es)	Usuario registrado, Conductor, Servidor, Sistema de notificaciones
Precondiciones	Puede ser iniciado por los siguientes casos de uso: UCAPP11 (Apuntarse a un viaje), UCAPP16 (Desapuntar viaje), UCAPP18 (Publicar viaje), UCAPP20 (Cancelar viaje).
Postcondiciones	La notificación se ha enviado correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. Recopilará la información a enviar. 2. Envía la información al servidor 3. La notificación queda almacenada en el servidor a la espera de ser recibida por el destinatario. 4. Finaliza el caso de uso.
Flujos alternativos	Si se produce un error en los pasos 2 y 3 saltará al paso 4.
Excepciones	Si se produce un fallo al enviar los datos al servidor, no se mostrará mensaje alguno de error.
Extensiones	UCAPP11 (Apuntarse a un viaje), UCAPP16 (Desapuntar viaje), UCAPP18 (Publicar viaje), UCAPP20 (Cancelar viaje).

6.3.17. Gestionar viajes

Identificador	UCAPP14
Nombre	Gestionar viajes
Resumen	Permite organizar los viajes del pasajero.
Actor(es)	Pasajero, Servidor
Precondiciones	Ninguna
Postcondiciones	Ninguna
Flujo normal	<p>Se puede acceder mediante la opción “Mis viajes”</p> <ol style="list-style-type: none"> 1. Se obtendrán los datos de todos los viajes del usuario y se listarán en pantalla. 2. Por cada viaje en la lista se muestra el conductor, el trayecto, el precio, un indicador de estado (realizado, apuntado) y podrá ampliar la información. 3. El usuario (pasajero) finaliza el caso de uso pulsando el botón correspondiente.

Flujos alternativos	<p>En el paso 2, si el usuario selecciona un viaje y accede a sus detalles podrá optar a:</p> <ul style="list-style-type: none"> • Como pasajero que se ha apuntado a un viaje: <ul style="list-style-type: none"> - Desapuntar viaje: Iniciará el caso de uso UCAPP16. • Como pasajero que ha realizado al menos un viaje: <ul style="list-style-type: none"> - Valorar al conductor: Iniciará el caso de uso UCAPP15. <p>Por tanto, para aquellos viajes cuyo estado sea “realizado”(y no “apuntado”) podrá valorar adicionalmente al conductor. Para aquellos viajes en los que su estado es “apuntado” podrá desapuntarse.</p>
Extensiones	UCAPP15 (Valorar conductor), UCAPP16 (Desapuntar viaje)

6.3.18. Valorar conductor

Identificador	UCAPP15
Nombre	Valorar conductor
Resumen	Permite hacer una valoración de un conductor por parte de un pasajero que ha realizado un viaje con él.
Actor(es)	Pasajero, Servidor
Precondiciones	Debe haber sido iniciado por el caso de uso UCAPP14 (Gestionar viajes). El estado del viaje que ha seleccionado previamente debe ser “realizado”.
Postcondiciones	Se ha realizado la valoración del conductor correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario pulsará el botón correspondiente en la pantalla de detalles del viaje. 2. Existirá una entrada de texto donde puede introducir sus comentarios con un límite de caracteres en la entrada. 3. También podrá puntuar al conductor del 1 al 5. 4. El sistema enviará los datos al servidor para ser públicos en cuanto el usuario pulse la opción de publicar valoración. 5. El caso de uso finalizará en cuanto el usuario opte por salir de la pantalla mediante el procedimiento correspondiente.
Excepciones	Si en el paso 4 ocurre un error se informará al usuario que la operación no se ha podido realizar, dando la opción de reintentar la operación. Si opta por reintentar, se ejecutará el paso 4 de nuevo.

6.3.19. Desapuntar viaje

Identificador	UCAPP16
Nombre	Desapuntar viaje
Resumen	Permite al pasajero desapuntarse de un viaje al que se ha apuntado previamente. Deberá informar al conductor de los motivos por los que anula su participación.
Actor(es)	Pasajero, Servidor
Precondiciones	Debe ser iniciado por el caso de uso UCAPP14 (Gestionar viajes)
Postcondiciones	El pasajero ha cancelado el viaje correctamente
Flujo normal	<ol style="list-style-type: none"> 1. El pasajero ha optado por cancelar su viaje, seleccionándolo previamente a través del caso de uso UCAPP14. 2. Se pedirá en pantalla que introduzca el motivo de la cancelación. 3. El usuario podrá optar a cancelar el viaje presionando el botón de confirmación. 4. Se preguntará en pantalla la confirmación de cancelación 5. Se eliminará el registro del usuario en el servidor del viaje correspondiente. 6. Se iniciará el caso de uso UCAPP13 (Enviar notificación) avisando al conductor que el usuario se desapunta del viaje. 7. Finaliza el caso de uso.
Flujos alternativos	<p>La entrada del motivo de la cancelación es obligatoria luego si no la introduce se informará al usuario de esa situación y que introduzca el motivo de la cancelación.</p> <p>En el paso 3, no continuará al siguiente paso hasta que el usuario no confirme la cancelación.</p>
Excepciones	Si en el paso 5 ocurre un error se informará al usuario que la operación no se ha podido realizar, dando la opción de reintentar la operación. Si opta por reintentar, se ejecutará el paso 5 de nuevo.
Inclusiones	UCAPP13 (Enviar notificación)

6.3.20. Organizar viajes

Identificador	UCAPP17
Nombre	Organizar viajes
Resumen	Permite gestionar los viajes del conductor.
Actor(es)	Conductor, Servidor
Precondiciones	Solo será accesible si el usuario ha agregado al sistema su vehículo.
Postcondiciones	Los viajes planificados se han mostrado correctamente.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario accederá a través de la opción “Mis planes de viaje” 2. El sistema obtendrá los datos necesarios de los viajes planificados por el conductor en el día (si tiene al menos uno) 3. Se mostrarán en una lista ordenada cronológicamente por la fecha del viaje mostrando sus características tales como el tipo de viaje, si es puntual o periódico, si es periódico, en qué días y a qué hora va a tener lugar, el trayecto a realizar. Serán los datos maestros de los viajes. Si mantiene pulsado un elemento de la lista podrá optar por cancelar o modificar sus características. 4. El usuario al pulsar sobre uno de sus viajes se mostrará un resumen de sus características y además podrá acceder al detalle de los próximos viajes (sus fechas programadas, sólo en el caso de que sea periódico) mostrados en una lista donde podrá seleccionar directamente presionando sobre el viaje las siguientes opciones: <ol style="list-style-type: none"> 4.1. Cancelar viaje 4.2. Marcar opción activar seguimiento (necesitará confirmación por parte del conductor) 5. El usuario podrá publicar un nuevo viaje mediante la pulsación del botón correspondiente iniciando el caso de uso UCAPP18 (Publicar viaje). 6. Si el usuario desea salir, finalizará el caso de uso mediante la pulsación del elemento correspondiente.
Flujos alternativos	<p>En el paso 2:</p> <ul style="list-style-type: none"> - Si no tiene ningún viaje programado para el día, mostrará los próximos viajes (si existen) <p>En el paso 3:</p> <ul style="list-style-type: none"> - Si el conductor activa el seguimiento en alguno de sus viajes planificados en el paso 4, se mostrará un indicador o símbolo destinado a informarlo, dando cuenta al usuario qué viajes está dando información a los usuarios. - Si mantiene pulsado el viaje podrá optar por cancelar todos las fechas programadas por lo que deberá iniciar el caso de uso UCAPP20 (Cancelar viaje) <p>En el paso 4:</p> <ul style="list-style-type: none"> - El usuario podrá volver al paso 2 si pulsa el elemento correspondiente a la opción de salir.

	<ul style="list-style-type: none"> - Paso 4.1. Cancelar viaje: Inicialá el caso de uso UCAPP20. Para ello deberá seleccionar previamente un viaje de la lista de viajes planificados y seleccionar uno de sus viajes programados (si tiene más de uno o es de ida y vuelta). - Paso 4.2. Activar/desactivar seguimiento: Se preguntará al conductor si confirma la operación. Si la confirma, iniciará el caso de uso UCAPP19 (Gestionar seguimiento). <p>Alternativamente el usuario podrá consultar su histórico de viajes realizados.</p>
Extensiones	UCAPP18 (Publicar viaje), UCAPP19 (Gestionar seguimiento), UCAPP20 (Cancelar viaje), UCAPP22 (Consultar viajes realizados).

6.3.21. Publicar viaje

Identificador	UCAPP18
Nombre	Publicar viaje
Resumen	Permite al conductor publicar su ruta y por tanto hacerla disponible para aquellos pasajeros que pueda interesarles viajar con él.
Actor(es)	Conductor, Servidor
Precondiciones	Debe haber sido iniciado por el caso de uso UCAPP17 (Organizar viajes)
Postcondiciones	Se ha almacenado un nuevo viaje disponible en el sistema.
Flujo normal	<ol style="list-style-type: none"> 1. Se le pedirá al usuario que seleccione el tipo de viaje (ida o ida y vuelta) que va a realizar y podrá indicar si es puntual o periódico. 2. Establecerá el punto de salida y el punto de llegada y se apoyará en los servicios de ubicación para ello. 3. Establecerá la fecha y la hora que tendrá lugar el viaje. 4. Por defecto el viaje será sólo de ida. Opcionalmente podrá ser de ida y vuelta. 5. El usuario indicará el nº de plazas disponibles 6. El usuario introducirá el precio de la plaza 7. El usuario podrá añadir un comentario o descripción sobre el viaje. 8. El usuario pulsará el botón publicar y generará la información necesaria (tanto del viaje como de las fechas programadas) será enviada al servidor y finalizará el caso de uso.
Flujos alternativos	<p>En el paso 1:</p> <ul style="list-style-type: none"> - Si el usuario opta por seleccionar un viaje puntual podrá indicar la fecha de inicio y la hora - Si el usuario opta por seleccionar un viaje periódico, podrá indicar la fecha de inicio y la fecha de fin del periodo en el que viajará. Si marca la opción de viaje de ida y vuelta podrá adicionalmente establecer la frecuencia, que puede ser diaria entre el periodo o semanal, que permitirá elegir los días de la semana que tendrán lugar. También podrá elegir una hora de inicio y de fin.

Excepciones	Si en el paso 8 ocurre un error se informará al usuario que la operación no se ha podido realizar, dando la opción de reintentar la operación. Si opta por reintentar, se ejecutará el paso 8 de nuevo.
-------------	---

6.3.22. Gestionar seguimiento

Identificador	UCAPP19
Nombre	Gestionar seguimiento
Resumen	Permite a un conductor activar el seguimiento de su viaje para que los usuarios interesados puedan conocer por dónde va. Este caso de uso será especialmente útil cuando el conductor establece el punto de origen de un viaje en un lugar no cercano a donde se encuentra permitiendo a los pasajeros conocer por dónde va.
Actor(es)	Conductor, Servidor, Servicios de ubicación.
Precondiciones	Debe ser iniciado por el caso de uso UCAPP17 (Organizar viajes).
Postcondiciones	El conductor podrá activar o desactivar el seguimiento de su posición.
Flujo normal	<ol style="list-style-type: none"> 1. El sistema recopilará todos los pasajeros que se encuentren apuntados al viaje del cual se informará sobre su seguimiento. 2. Notificar a los pasajeros sobre el nuevo estado de seguimiento. 3. Finaliza el caso de uso
Inclusiones	UCAPP13 (Enviar notificación)

6.3.23. Cancelar viaje

Identificador	UCAPP20
Nombre	Cancelar viaje
Resumen	Permite cancelar completamente un viaje o específicamente en una fecha concreta.
Actor(es)	Conductor, Servidor
Precondiciones	Debe ser iniciado por el caso de uso UCAPP14 (Gestionar viajes). El conductor ha seleccionado previamente uno de sus viajes favoritos y al acceder a los detalles puede ver la lista de los viajes planificados tanto los de ida como los de vuelta.
Postcondiciones	El viaje ha sido cancelado por completo o uno de sus días (si es periódico).
Flujo normal	<ol style="list-style-type: none"> 1. Se solicitará al usuario que introduzca el motivo de la cancelación 2. El sistema esperará a que el usuario pulse el botón “proceder”. 3. Se pedirá confirmación de la cancelación por parte del usuario. 4. Se envía una notificación a los pasajeros que se hayan apuntado al viaje o a alguna de sus fechas programadas. 5. Se elimina toda la información del viaje

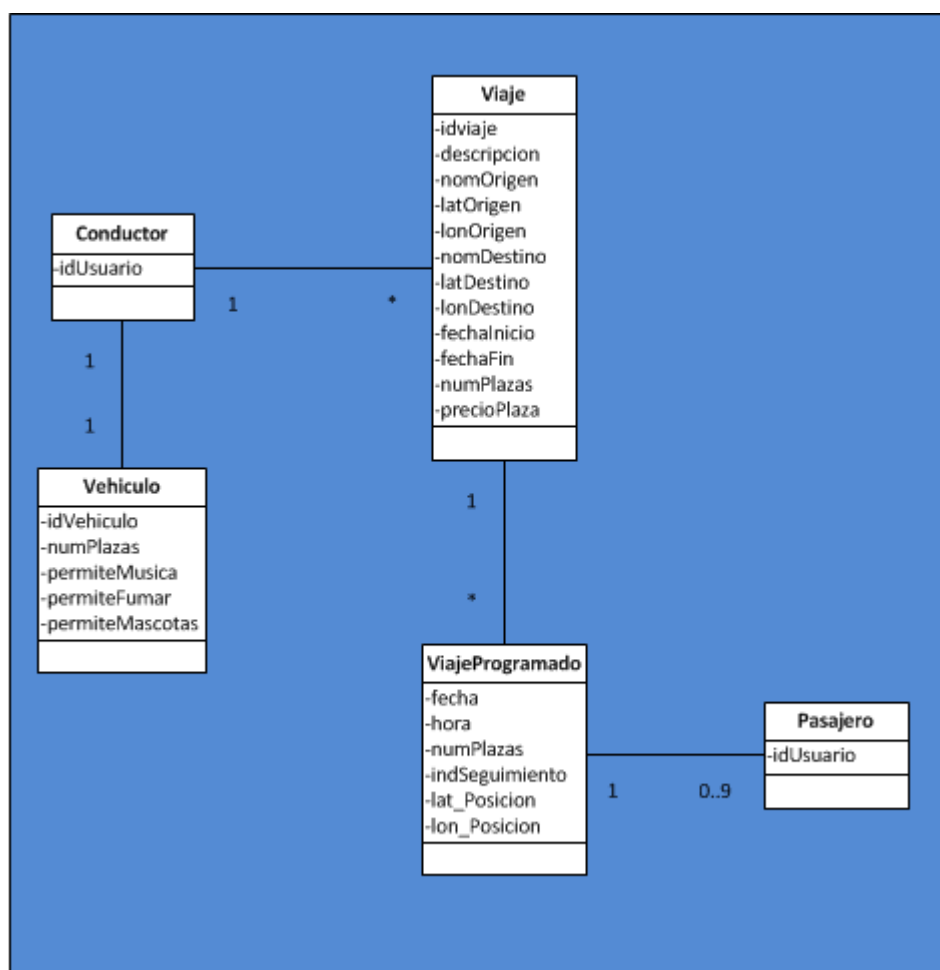
	6. Finalizará el caso de uso
Flujos alternativos	<p>En el paso 3:</p> <ul style="list-style-type: none"> - Se le preguntará al usuario si está seguro de continuar y si pulsa que no, entonces vuelve al paso 2. <p>En el paso 4:</p> <ul style="list-style-type: none"> - Si el usuario ha cancelado un viaje en una fecha concreta que tenía programada y no un viaje por completo, sólo será eliminada la información de esa fecha en particular y se enviará notificación a los usuarios afectados.
Excepciones	Si en el paso 5 ocurre un error se informará al usuario que la operación no se ha podido realizar, dando la opción de reintentar la operación. Si opta por reintentar, se ejecutará el paso 5 de nuevo.
Inclusiones	UCAPP13 (Enviar notificación)

6.3.24. Consultar viajes realizados

Identificador	UCAPP22
Nombre	Consultar viajes realizados
Resumen	Permitirá adicionalmente
Actor(es)	Conductor
Precondiciones	Debe haber sido iniciado por el caso de uso UCAPP14 (Gestionar viajes)
Postcondiciones	Se han listado todos los viajes organizados por el conductor.
Flujo normal	<ol style="list-style-type: none"> 1. Se mostrará una lista de los viajes organizados (que no estén en curso) por el conductor. 2. Se mostrarán las características de cada viaje 3. El usuario podrá seleccionar uno de los viajes de la lista para ampliar la información. 4. El usuario puede optar por salir y por tanto, se finalizará el caso de uso.
Flujos alternativos	<p>Paso 3:</p> <p>Si el usuario opta por consultar los detalles, se mostrará una cabecera con los datos del viaje y a continuación una lista de todos los viaje realizados en distintas fechas programadas al igual que en el caso de uso UCAPP14 Gestionar viajes.</p>

6.4. Entidades del dominio

A continuación vamos a enumerar las clases más importantes identificadas en el dominio del problema. La representación gráfica del modelo del dominio es la siguiente:



Las principales entidades son:

- Conductor, que representa al usuario que viaja y ofrece su vehículo para compartir un número determinado de plazas de su vehículo con otros usuarios.
- Vehículo, que representa al vehículo del conductor.
- Viaje, organizado por el conductor en el que indica su punto de salida, su punto de llegada, la fecha y la hora de salida, el precio de la plaza del vehículo, el número de plazas disponibles (se admitirán como máximo nueve).
- Viaje programado, el conductor para un viaje puede programar varias fechas de desplazamiento o días alternativos de modo que los pasajeros puedan apuntarse (hasta un máximo de nueve).
- Pasajero, representa al usuario que se apunta a un viaje que ha planificado (programado) un conductor para una fecha concreta.

7. DISEÑO DEL SISTEMA

7.1. ARQUITECTURA DEL SISTEMA

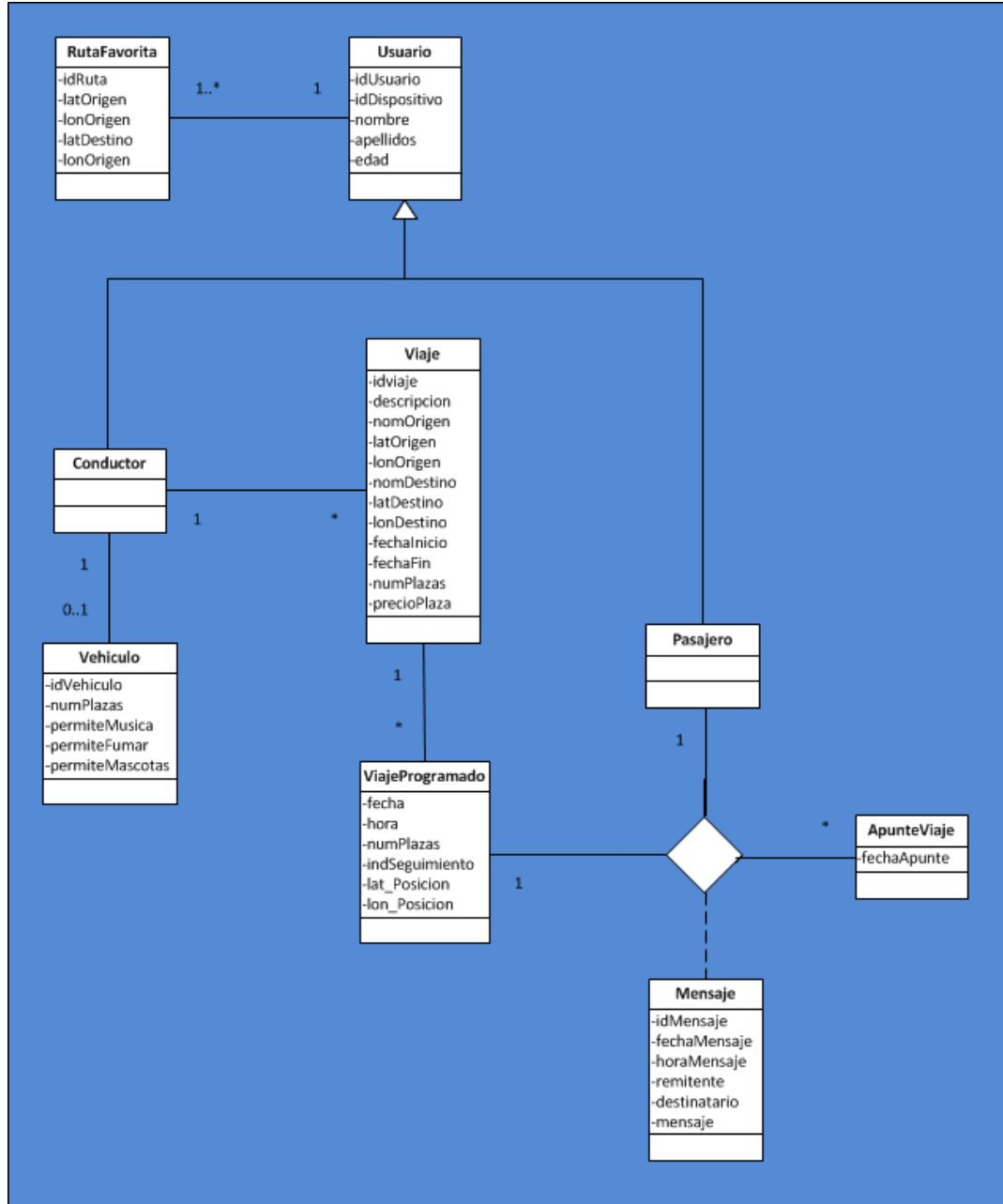
7.2. DISEÑO ESTÁTICO

El siguiente diagrama recoge las entidades principales del sistema. Cabe destacar que el modelo de dominio se ha ampliado de tal manera que se recogen los detalles de cómo los pasajeros se apuntan a los viajes y contempla el sistema de mensajería.

Los pasajeros se apuntarán a los viajes planificados por el conductor que al menos debe existir uno por cada viaje y sería la manera más simple de definir un viaje puntual. Si el usuario decide establecer un viaje rutinario entre semana escogiendo los días que viajará, el pasajero se apuntará (a través de la entidad ApunteViaje) a cada viaje programado (hasta que se cubran las plazas del vehículo) representado por la entidad ViajeProgramado.

Cabe destacar también que el mensaje será el nexo de comunicación entre las entidades principales como se puede observar en el diagrama. Sin embargo como se ha especificado en la toma de requisitos, no existirá la posibilidad de comunicarse entre los pasajeros puesto la finalidad del software no es la de mensajería sino de proveer de una herramienta que facilite la comunicación entre el conductor y los pasajeros.

También se ha incorporado la entidad RutaFavorita que representa aquellos viajes que el usuario estaría interesado en realizar.

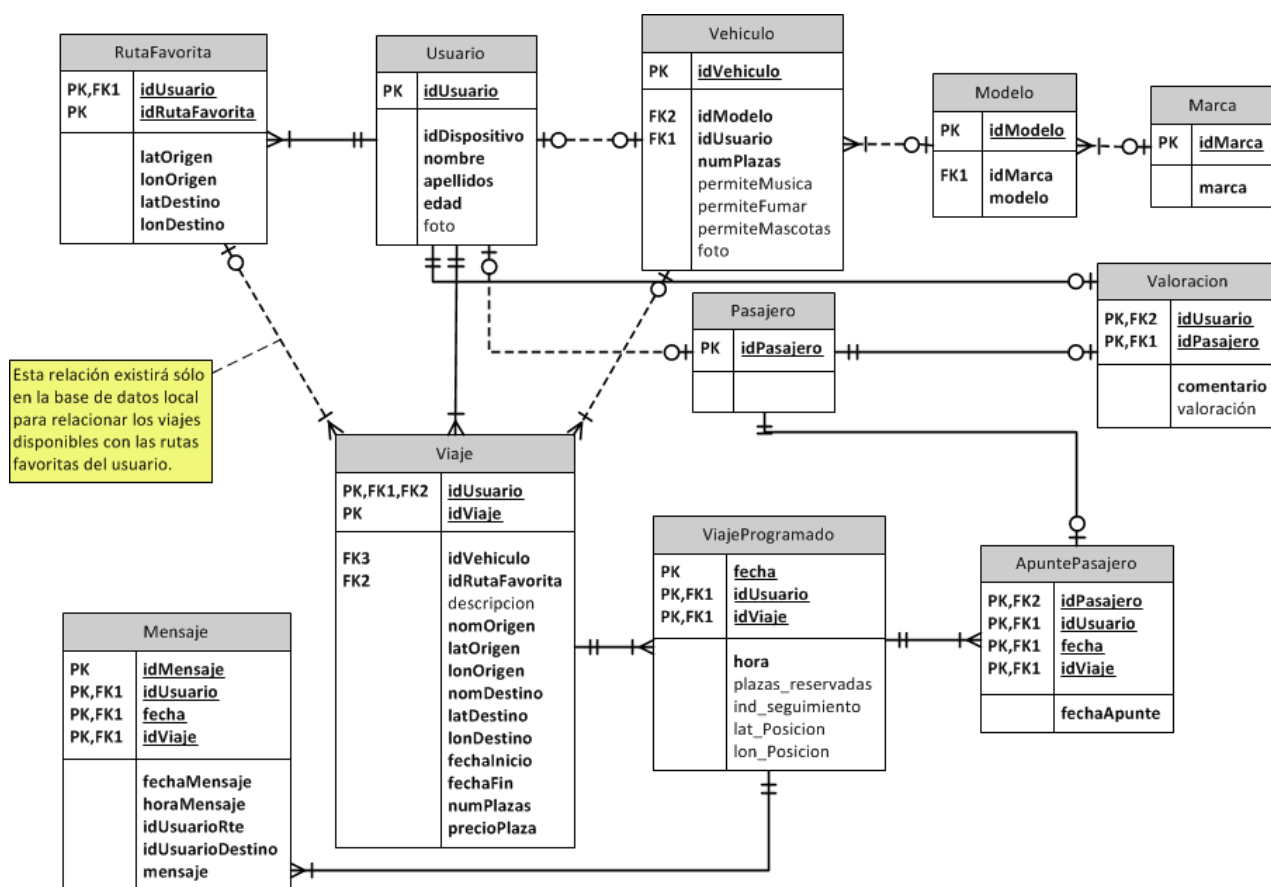


7.3. DISEÑO DE LA PERSISTENCIA

A continuación se describa las entidades principales que formarán parte de la base de datos del sistema. En el dispositivo del usuario existirá una réplica parecida a la que existe en el servidor. Sin embargo, la entidad rutas favoritas será de ámbito privado y por tanto no existirá en la base de datos del servidor.

7.3.1. MODELO RELACIONAL

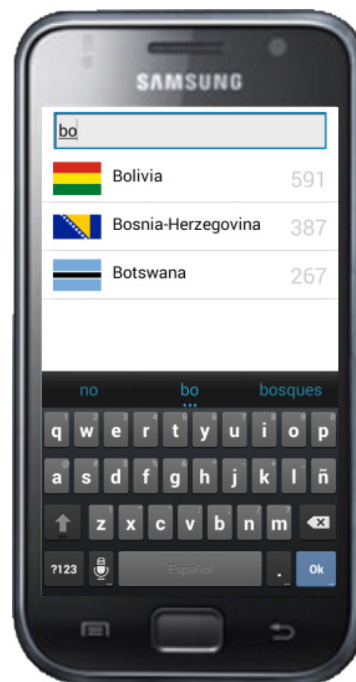
A fin de hacer más cómoda la lectura del diagrama de entidad-relación, se ha definido la entidad Pasajero aunque en realidad dicha entidad no va a ser necesaria puesto que en la entidad ApuntePasajero, el atributo idPasajero ya se interpreta que es el identificador de usuario. De esta forma se puede interpretar correctamente las relaciones entre las entidades y sus atributos.

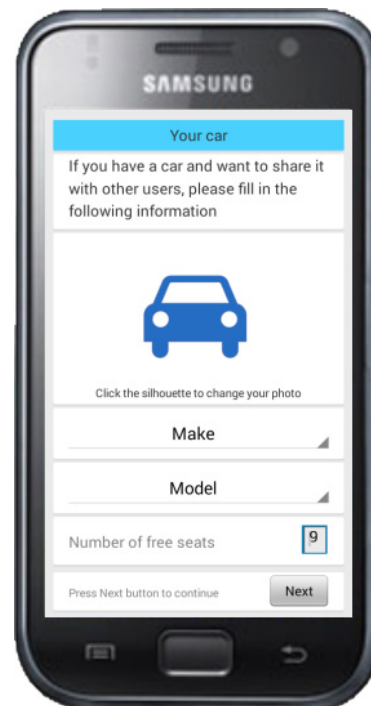
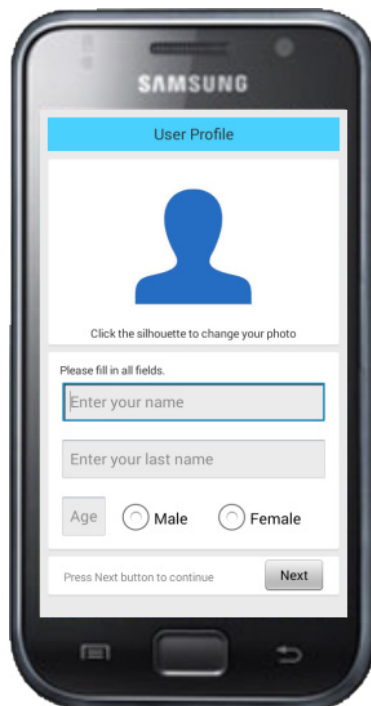
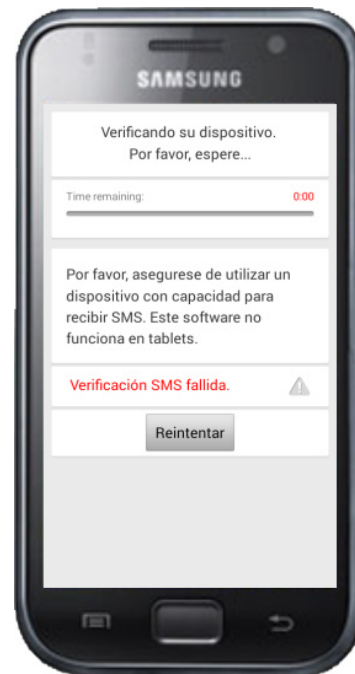


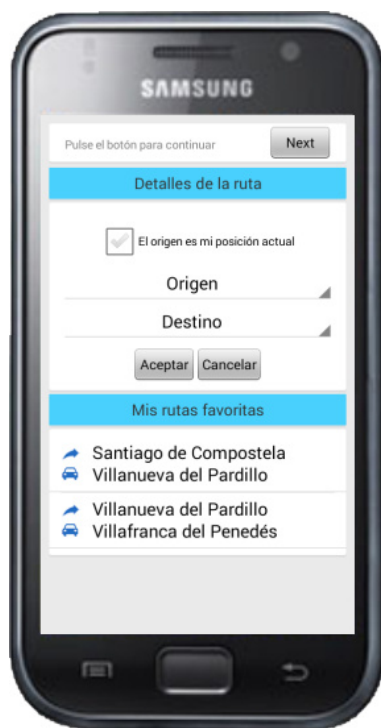
7.4. DISEÑO DE LAS INTERFACES DE USUARIO

7.4.1. Prototipos del asistente

Al iniciar la aplicación por primera vez, se iniciará un asistente para el registro del usuario y la configuración de la aplicación para empezar a funcionar:







7.5. DECISIONES TECNOLÓGICAS

7.5.1. Lenguajes de Desarrollo

El lenguaje utilizado para el desarrollo del software del dispositivo será java a través del SDK de android a través del kit adt-bundle que proporciona la plataforma mediante el IDE eclipse.

El lenguaje utilizado para desarrollar el intercambio de datos entre los clientes y el servidor es php 5.

7.5.2. Soporte de bases de datos local

El dispositivo utilizará un soporte de almacenamiento de datos SQLite donde se almacenará toda la información de carácter privado.

7.5.3. Soporte de bases de datos remota

El soporte de bases de datos utilizado es MySQL 5 en un sistema Linux.

7.5.4. Acceso a datos

El acceso a los datos del sistema por parte del dispositivo se realizará mediante una pasarela web a través del protocolo http.

7.5.5. Soporte de comunicaciones entre clientes y servidor

Para el intercambio de datos entre los clientes y el servidor se realizará mediante servicios web basados en JSON con codificación url BASE64 y a su vez serán privados mediante su ofuscación a través de la encriptación de los datos mediante el algoritmo AES/CBC. Los servicios de ubicación que proveen las herramientas necesarias para la ubicación correcta del usuario así como la búsqueda de posiciones geográficas son provistas en formato

8. IMPLEMENTACIÓN

Vamos a detallar el resultado de la implementación de cada una de las unidades funcionales del proyecto. Dada la envergadura del proyecto y algunas dificultades técnicas no ha sido posible completar el 100% de las funcionalidades. Sin embargo, se ha seguido el diseño planificado y expondremos a continuación cuáles serán los pasos a seguir para acometer las funcionalidades faltantes en una segunda fase.

8.1. HITOS CONSEGUIDOS

- Se ha completado el asistente: La gestión de los usuarios se llevará a cabo a través del asistente, los cuales podrán adicionalmente incorporar su vehículo, definir las rutas que realizarán e identificarse al resto de usuarios como conductores. El resto de usuarios podrá determinar cuáles son sus rutas favoritas, es decir, podrán definir aquellas rutas en las que estén interesados y establecer en que radio de búsqueda en kilómetros desean obtener los resultados. De esta forma se consigue aplicar un filtro de búsqueda para aquellas rutas que se encuentren cercanas a un punto de partida y destino establecidos que detallaremos a continuación. Mediante el servicio web `checkUserDevice.php` se realizará la consulta al servidor y se comprobará si el usuario existe y si el identificador IMEI del dispositivo que está ejecutando la aplicación coincide con el registrado en el sistema. Si no coincide o el usuario no existe, se iniciará el asistente.
- La aplicación es funcional al 100%: Los usuarios pueden agregar sus rutas favoritas y pueden recibir las notificaciones de los viajes ofertados por los conductores. También pueden ponerse en contacto con los conductores a través del servicio de mensajería para consultarles cuestiones relacionadas con algún viaje en particular. Tanto los pasajeros como los conductores también pueden consultar los viajes que han realizado y aquellos en los que se han apuntado para una fecha concreta. Si un pasajero no puede realizar el viaje deberá notificarlo al conductor utilizando para ello el sistema de mensajería. En todo momento si un usuario decide compartir su vehículo, podrá darlo de alta en el sistema y planificar/publicar sus viajes.
- El sistema de mensajería: Está gestionado por el servicio `NotificationService`. Permite poner en contacto a los usuarios y pasajeros con el conductor. El servicio se inicia durante el arranque del dispositivo. Cada 20 segundos comprueba los mensajes que deberán enviarse a sus destinatarios que se encontrarán almacenados en el fichero de base de datos local de SQLite `userdata.db`, a través del servicio web `checkMessages.php`. Cada 40 segundos comprobará a través del mismo servicio web si tiene mensajes como destinatario a través del mismo servicio web. Se ha desarrollado pero no se ha implementado en el software finalmente la posibilidad de notificar acuses de recibo de los mensajes, es decir, informar al emisor que el receptor ha recibido el mensaje.

- El sistema de notificaciones: También es gestionado por el servicio NotificationService. Cada 60 segundos consulta al servidor a través del servicio web checkJourneys.php si existen viajes disponibles

8.2. HITOS A ACOMETER EN UNA SEGUNDA FASE

- Aunque el sistema lo soporta, no se ha implementado la posibilidad de establecer viajes periódicos. La solución precisamente se ha diseñado para que un usuario en el caso de viajes periódicos pueda seleccionar el día que mejor le convenga. La forma de implementarlo es sencilla, puesto que simplemente habría que incorporar en el servicio addJourney.php una nueva funcionalidad que permitiera crear tantas entradas en la tabla smc_planned_journeys como días existieran en el intervalo de fechas de inicio y fin de viaje. El esquema de funcionamiento sería el mismo, sólo que en vez de aparecer una sola entrada en la pantalla de detalles del viaje, aparecerían todas las nuevas entradas generadas. Este planteamiento dota a la aplicación de fiabilidad de uso al usuario puesto que no tiene que leer una lista interminable de viajes como pasa en las aplicaciones actuales. Tampoco se ha podido establecer si el viaje es de sólo ida o ida/vuelta. Podría incorporarse esta funcionalidad mediante la modificación de la estructura de la tabla smc_journeys incorporando una nueva columna que indicara el identificador de viaje maestro de tal manera que se pudieran mostrar agrupados distintos puntos de origen y destino para un mismo viaje.
- Pese a que está integrada y es funcional la opción del idioma, no están asociados todos los controles al diccionario de datos incorporado en el fichero de base de datos local, por lo que habría que crear nuevas entradas para todos los literales faltantes y asociarlos a sus respectivos componentes.
- Tampoco ha dado tiempo a implementar la funcionalidad de valorar al conductor por parte de un pasajero, lo cual dotaría a la aplicación de una mayor calidad del servicio puesto que podría
- A través de la pantalla de “próximos viajes del conductor”, el conductor puede cancelar el viaje. Faltaría implementar en la clase gestora de base de datos local LocalDataManager una función SQL que permita insertar el mensaje que llegaría a todos los viajeros apuntados indicando el motivo de la cancelación y a continuación enviar a través del servicio existente o incluso uno nuevo que actualice la entrada en la tabla smc_journeys para que su estado sea cancelado y a su vez modificar el servicio web checkJourneys.php para que sólo liste aquellos viajes cuyo estado sea activo.
- Otra pequeña funcionalidad faltante es la de la eliminación de una ruta favorita por parte de un usuario que simplemente se resolvería implementando una función en la clase gestora de base de datos local LocalDataManager para que realice la eliminación de la entrada en la tabla smc_user_favroutes para el identificador de ruta seleccionado en la pantalla de rutas favoritas del usuario.
- Otra pieza clave para que la aplicación sea mucho más atractiva es dotarla de la posibilidad de que el conductor pueda iniciar el seguimiento del viaje que pone

en marcha para que los viajeros puedan consultar por dónde va. Esto sería posible mediante la creación de un nuevo servicio web que fuera accedido por el servicio de notificaciones de la aplicación para enviarle los datos de su posición cada 60 segundos.

- Tampoco se ha podido implementar la búsqueda de nuevas rutas independientes de las favoritas. La implementación es relativamente sencilla puesto que se puede reutilizar el servicio existente de buscar rutas favoritas enviándole como parámetros las coordenadas de origen y destino del viaje que deseamos obtener resultados.

8.3. FUTURAS MEJORAS A INCORPORAR

Personalmente pienso que el sistema puede dar mucho más de sí. Por ejemplo, sería importante que a la hora de visualizar los viajes disponibles se listaran en función de la distancia respecto del origen, es decir, que se listen primero aquellos viajes cuyo punto de origen es más cercano y se indique la distancia al origen en la lista.

Otra mejora razonable sería que en la lista de viajes disponibles se visualizara con una marca o icono similar para indicar al usuario que se ha apuntado a dicho viaje. Aunque ya lo puede hacer directamente a través de la sección de “Mis viajes (Mis viajes realizados/Mis próximos viajes/Todos mis viajes)”.

Otra posibilidad de mejora podría ser la de dar opción al conductor de crear/planificar un viaje pero no hacerlo visible al resto de usuario hasta publicarlo (actualmente planificar/crear y publicar se realizan de forma simultánea).

8.4. INVENTARIO DE COMPONENTES SOFTWARE

Vamos a resumir los componentes software que se han desarrollado y una pequeña descripción de lo que hace cada uno:

8.4.1. PAQUETES

A continuación se muestra la relación de los distintos paquetes del proyecto y sus características:

Paquete	Descripción
com.prototype.sharemycar	Paquete principal
com.prototype.sharemycar.data	Contiene las estructuras de datos de la información
com.prototype.sharemycar.adapters	Contiene los adaptadores necesarios para los componentes ListView de las distintas pantallas.
com.prototype.sharemycar.assistant	Contiene todas las pantallas relacionadas con el asistente.
com.prototype.sharemycar.app	Contiene todas las pantallas relacionadas con la aplicación.

com.prototype.sharemycar.managers	Contiene todas las clases gestoras utilizadas por el asistente y la aplicación.
com.prototype.sharemycar.services	Contiene los servicios (que se ejecutarán en el dispositivo) utilizados por la aplicación.

8.4.2. COMPONENTES

Componente	Paquete	Descripción
ErrorActivity.java	com.prototype.sharemycar	Pantalla de visualización de errores.
MainActivity.java	com.prototype.sharemycar	Pantalla encargada de comprobar el dispositivo, la conectividad y de iniciar el asistente si es necesario.
*.java	com.prototype.sharemycar.adapters	Contiene todas las clases que gestionan los adaptadores necesarios para visualizar la información a través de los componentes listview de las distintas pantallas.
*.java	com.prototype.sharemycar.data	Contiene todas las clases necesarias para las estructuras de información de la aplicación.
ChooseCar.java	com.prototype.sharemycar.assistant	Pantalla que permite seleccionar tanto la marca como el modelo de un vehículo.
ChooseCountry.java	com.prototype.sharemycar.assistant	Pantalla que permite seleccionar un país para obtener su prefijo telefónico internacional.
ChooseLanguage.java	com.prototype.sharemycar.assistant	Pantalla que permite seleccionar y establecer el idioma de la aplicación.
ChooseLocation.java	com.prototype.sharemycar.assistant	Pantalla que permite seleccionar manualmente una ubicación mediante los servicios de Google Maps.
DriverSettings.java	com.prototype.sharemycar.assistant	Pantalla que establece los datos del vehículo del conductor.
ManageDriverJourneys.java	com.prototype.sharemycar.assistant	Pantalla que permite

		planificar/publicar un viaje al conductor
PhoneNumber.java	com.prototype.sharemycar.assistant	Pantalla de verificación de dispositivo del usuario.
SearchOptions.java	com.prototype.sharemycar.assistant	Pantalla de opciones de búsqueda del usuario
Terms.java	com.prototype.sharemycar.assistant	Pantalla que se inicia cuando se inicia el asistente. Supuestamente sirve para que el usuario lea las condiciones del servicio y le satisfacen, las acepte y pueda continuar.
UserFavRoutes.java	com.prototype.sharemycar.assistant	Pantalla donde el usuario puede introducir sus rutas favoritas.
UserProfile.java	com.prototype.sharemycar.assistant	Pantalla de perfil de usuario, debe introducir su nombre y opcionalmenet una foto que será enviada al servidor.
VerifyUser.java	com.prototype.sharemycar.assistant	Pantalla que estará a la espera de recibir un SMS a través de un broadcast receiver del sistema para validar al usuario.
AvailableJourneys.java	com.prototype.sharemycar.app	Pantalla encargada de visualizar los viajes disponibles.
ChatPlannedJourney.java	com.prototype.sharemycar.app	Pantalla encargada de mostrar los mensajes entre el conductor y los usuarios/pasajeros. También se utiliza para introducir el motivo de la cancelación de un viaje por parte de un pasajero.
EnrollAvailablePlannedJourneys.java	com.prototype.sharemycar.app	Pantalla que permite a un usuario apuntarse a un viaje de un conductor. Opcionalmente puede desapuntarse (una vez apuntado).
JourneyDetailsForDriver.java	com.prototype.sharemycar.app	El conductor puede consultar qué pasajeros se han apuntado a un viaje y opcionalmente puede cancelarlo (funcionalidad no disponible).
JourneyDetailsForUser.java	com.prototype.sharemycar.app	Permite conocer los detalles del viaje al que un pasajero se ha apuntado y puede optar por contactar con el conductor o

		cancelar el viaje.
MyChatConversations.java	com.prototype.sharemycar.app	Muestra todas las conversaciones en las que ha intervenido tanto el conductor como el usuario/pasajero en un viaje concreto.
MyChats.java	com.prototype.sharemycar.app	Muestra todos los chats en los que el usuario ha participado. Obtendrá el detalle de las conversaciones en la pantalla anterior.
MyJourneys.java	com.prototype.sharemycar.app	Muestra los viajes realizados por el usuario como pasajero, los próximos o ambos.
MyJourneysAsDriver.java	com.prototype.sharemycar.app	Muestra los viajes realizados por el conductor, los próximos o ambos.
SelectChatAvailablePlanned Journeys.java	com.prototype.sharemycar.app	Permite a un usuario/pasajero consultar sobre un viaje al conductor.
AppLocationManager.java	com.prototype.sharemycar.managers	Clase gestora encargada de obtener los datos de la ubicación provistas por los servicios de Google Maps.
AppManager.java	com.prototype.sharemycar.managers	Clase gestora encargada de iniciar las pantallas que requiera la aplicación.
AutoLocationManager.java	com.prototype.sharemycar.managers	Encargada de obtener los datos geográficos via GPS o servicios de red.
DeviceManager.java	com.prototype.sharemycar.managers	Encargada e gestionar todas las funciones relativas al dispositivo como por ejemplo obtener el IMEI.
DownloadFileManager.java	com.prototype.sharemycar.managers	Encargada de descargar ficheros necesarios del servidor como por ejemplo las fotos de los usuarios, vehículos, etc...
DriverSettingsManager.java	com.prototype.sharemycar.managers	Encargada de gestionar todo lo concerniente al conductor y enviarlo al servidor.
FileManager.java	com.prototype.sharemycar.managers	Encargada de gestionar los

		archivos locales del dispositivo, como por ejemplo el fichero de base de datos, los ficheros de las fotos de los usuarios, vehículos, etc...
JourneyDetailsManager.java	com.prototype.sharemycar.managers	Encargada de gestionar todo lo relevante a los datos de los viajes.
LocalDataManager.java	com.prototype.sharemycar.managers	Encargada de gestionar todo lo relevante respecto al acceso a los datos necesarios y realizar las operaciones oportunas las necesarios en el fichero de base de datos local.
LogManager.java	com.prototype.sharemycar.managers	Encargada de visualizar la información necesaria de trazas de log para ayudar en la depuración de errores.
RemoteDataManager.java	com.prototype.sharemycar.managers	Encargada de gestionar todas aquellas operaciones necesarias en el servidor de base de datos remoto.
SecurityManager.java	com.prototype.sharemycar.managers	Encargada de identificar el dispositivo del usuario, y responsable de la encriptación de los datos durante la transmisión.
Translator.java	com.prototype.sharemycar.managers	Encargada del diccionario de datos de literales para visualizar la aplicación en varios idiomas.
UploadFileManager.java	com.prototype.sharemycar.managers	Encargada de remitir los ficheros de imágenes correspondientes al servidor.
UserJourneysManager.java	com.prototype.sharemycar.managers	Encargada de gestionar los viajes de un usuario y su interacción con el servidor.
UserMessagesManager.java	com.prototype.sharemycar.managers	Encargada de gestionar todo lo relacionado con el sistema de mensajería (envío y recepción de mensajes)
UserSettingsManager.java	com.prototype.sharemycar.managers	Encargada de gestionar las preferencias del usuario y enviarlas al servidor remoto.
UtilManager.java	com.prototype.sharemycar.managers	Encargada de realizar operaciones necesarias por distintos

		componentes tales como conversión de fechas, lectura de imágenes, etc...
BootStartServiceReceiver.java	com.prototype.sharemycar.services	Servicio encargado de iniciar el servicio de notificaciones durante el arranque del dispositivo.
NotificationService.java	com.prototype.sharemycar.services	Servicio encargado de recibir de notificaciones de nuevos mensajes o viajes disponibles en base a las rutas favoritas del usuario.
SMSTHandlerService.java	com.prototype.sharemycar.services	Encargado de enviar un SMS al propio n° de teléfono del dispositivo.

8.4.3. SERVICIOS WEB

A continuación se muestra la relación de los distintos servicios web que interactúan con la aplicación:

Servicio	Descripción
addJourney.php	Incorpora un nuevo viaje en el servidor de base de datos
addUserCar.php	Incorpora el vehículo de un usuario
checkJourneys.php	Consulta los viajes disponibles y permite al usuario apuntarse o cancelarlo
checkMessages.php	Consulta los mensajes para notificar a los usuarios.
checkMyChats.php	Obtiene los datos necesarios para la pantalla de consulta de mensajes así como las consultas realizadas en diferentes viajes por parte de un usuario.
checkMyJourneys.php	Obtiene toda la información relacionada con los viajes realizados por o con el conductor.
checkUserDevice.php	Encargado de validar el dispositivo del usuario.

8.5. CONCLUSIÓN

El objetivo de este proyecto ha sido demostrar que los dispositivos móviles pueden ser herramientas indispensables para las personas al igual que lo fueron los ordenadores en su día pero con el valor añadido de la movilidad ofreciendo la posibilidad de acceder a los diferentes servicios disponibles desde cualquier lugar y en cualquier momento.

El desarrollo orientado a plataformas móviles es el futuro: Hoy en día existen 20 más de veinte millones de smartphones en España y el número sigue creciendo. Esta cifra es superior a la de ordenadores personales y sin embargo operan como tales. La ventaja principal respecto del ordenador es que siempre va con su propietario a todas partes y actualmente disponen de la capacidad de conexión a Internet lo que hace de ellos una herramienta fundamental para poder poner en práctica múltiples soluciones y en particular, el caso que nos ocupa.

Este proyecto no sólo ha tratado de demostrar que se pueden construir soluciones software muy útiles para el usuario a través de unas herramientas de desarrollo sino que también puede servir su diseño para construir otro tipo de aplicaciones que puedan satisfacer requisitos parecidos como por ejemplo la gestión de eventos deportivos, culturales, alquiler de vehículos particulares, etc... basados en la ubicación del usuario y a través de su propio servicio de mensajería.

La intención de desarrollar esta aplicación ha sido no sólo de codificar una demo, sino de construir una aplicación completa en la que se ha hecho hincapié en su usabilidad y con el objetivo de poder ser utilizada por usuarios en la vida real a través de Google Play.

Para concluir, el desarrollo de aplicaciones en este tipo de plataformas ha supuesto un reto personal puesto que he tenido experiencia en el desarrollo de aplicaciones plataformas híbridas (concretamente videojuegos a través de C++, para iOS y android) pero nada de experiencia en el ámbito del sdk de android y lo cierto es que a pesar de la carga de trabajo del proyecto he conseguido aprender todo lo necesario para acometer la solución, al menos aquellas funcionalidades que he considerado fundamentales. El resto de funcionalidades que no se han podido incorporar pienso que podrían llevarse a cabo en cuestión de aproximadamente de dos meses, un mes para la integración de las nuevas funcionalidades y otro mes dedicarlo a las pruebas y ajustes antes de su paso a producción.

8.6. NOTAS SOBRE EL SOFTWARE

El software se distribuye con la opción de SMS deshabilitada ya que el operador cargará el coste del envío del SMS cada vez que se pruebe. Si se desea probar su funcionamiento se debe editar el fichero y descomentar la siguiente línea en el servicio SMSHandlerService.java en el método onStartCommand().

Descomentar; //sendVerificationSMS();

A continuación en VerifyUser.java en el método addListeners:

```
Comentar testVerificationOk();  
Descomentar //initTimeoutTimer();
```

Para las pruebas he creado dos sitios web externos de servicio gratuito, dado que no son muy fiables sobre todo cuando los servicios están activos por lo que he decidido cambiar los tiempos para evitar problemas. Sin embargo, en el caso de utilizar un servidor u otro o incluso uno propio, los pasos son los siguientes:

En la aplicación, modificar AppManager.getServerName():

```
String sServerName = "url_servidor"
```

Para utilizar un servidor u otro:

- Hostinger:

```
String sServerName = "http://sharemycar.esy.es/";
```

- 000WebHost:

```
String sServerName = "http://sharemycar.hostei.com/";
```

- Servidor local:

```
String sServerName = "http://192.168.X.Y:8080/";
```

En los servicios web utilizar la conexión necesaria cambiando en el método getServerConnection():

- Hostinger:

```
$db_server = getConnectionExternalServerConnectionES();
```

- 000WebHost:

```
$db_server = getConnectionExternalServerConnection();
```

- Servidor local:

```
$db_server = getConnectionLocalServerConnection();
```