

**DESARROLLO DE SOFTWARE MÓVIL PARA LA GESTIÓN DE LA MOVILIDAD
DE LA COMUNIDAD UNIVERSITARIA HACIA Y DESDE EL CAMPUS.**

JUAN FELIPE RUEDA ROMÁN

JAVIER EDUARDO CASTELLANOS ONTIVEROS

UNIVERSIDAD INDUSTRIAL DE SANTANDER

FACULTAD DE INGENIERIAS FISICO-MECANICAS

ESCUELA DE INGENIERIA DE SISTEMAS E INFORMATICA

BUCARAMANGA

2017

**DESARROLLO DE SOFTWARE MÓVIL PARA LA GESTIÓN DE LA MOVILIDAD
DE LA COMUNIDAD UNIVERSITARIA HACIA Y DESDE EL CAMPUS.**

JUAN FELIPE RUEDA ROMÁN

JAVIER EDUARDO CASTELLANOS ONTIVEROS

**Trabajo de Grado para optar al título de
Ingeniero de Sistemas e Informática**

Director

FERNANDO ANTONIO ROJAS MORALES

Ingeniero de Sistema, Msc

UNIVERSIDAD INDUSTRIAL DE SANTANDER

FACULTAD DE INGENIERIAS FISICO-MECANICAS

ESCUELA DE INGENIERIA DE SISTEMAS E INFORMATICA

BUCARAMANGA

2017

AGRADECIMIENTOS

A mi papá por siempre apoyarme en todas las decisiones que he tomado y por siempre brindarme consejos y escucharme siempre que él podía, por sacar nuestra familia adelante y siempre darnos un buen ejemplo.

A mi mamá por ser siempre que ha sido y siempre un apoyo incondicional en situaciones difíciles y fáciles, por sacarnos adelante y darnos mucho amor.

A mis hermanos por dar apoyo y siempre dándome a entender que siempre tenemos que dar lo mejor de nosotros y que no importe la situación.

A mis amigos, amigas, compañeros, compañeras que siempre he tenido el apoyo de ellos y siempre sacando sonrisas siendo muy importantes para seguir adelante.

Al profesor Fernando Rojas, por brindarme el conocimiento desde 2 semestre y siempre contar con tiempo para apoyarnos.

JAVIER EDUARDO CASTELLANOS ONTIVEROS

AGRADECIMIENTOS

A lo largo del desarrollo de este proyecto muchas personas han de diferentes formas contribuido a la realización del mismo, nombrare algunos de los que con sus aportes contribuyeron de manera más significativa para la culminación de este.

Agradezco a mis compañeros de clase y mi compañero de proyecto que han aportado a mi formación de una u otra forma, a aquellos que nos permitieron dar a conocer nuestro proyecto, a los que nos permitieron forjar vínculos que al final fueron significativos.

A aquellas personas que se interesaron en nuestro proyecto y están dispuestas a apoyarnos para seguir adelante.

A aquellos profesores que con su carisma o sabiduría aportaron a mi formación personal y profesional, en especial a mi director de proyecto, que me alentó a tomar una de las decisiones más difíciles de mi carrera profesional y siempre creyó en mí.

A mi mamá, papá y hermanos que me ha apoyado de todas las formas posibles y me mantuvieron por el camino correcto cuando yo ya no quería seguir, también a mi novia que siempre me ha motivado y ha sido un apoyo incondicional desde el comienzo de la realización de este proyecto.

A todos ustedes y aquellos que también contribuyeron pero que por alguna razón se me escapa nombrar en este momento, gracias.

JUAN FELIPE RUEDA ROMÁN

Tabla de contenido

INTRODUCCION	14
1. PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA	15
2. OBJETIVOS	16
2.1 OBJETIVO GENERAL	16
2.2 OBJETIVOS ESPECÍFICOS	16
3. MARCO DE REFERENCIA	17
3.1 Terminología:	17
3.1.1 Carpooling	17
3.1.2 Data Base as a Service	17
3.1.3 Backend as a Service	18
3.1.4 Web Service.....	18
3.2 Android	18
3.2.1 Android sdk 16	19
3.2.1 Android sdk 25	19
3.3 Lenguajes de programación	20
3.3.1 Java	20
3.3.2 Json.....	20
3.3.3 XML	20
3.3.4 JavaScript.....	21
3.3.5 Node.js	21
3.4 Modelo Vista Controlador	22
3.4.1 Modelo	22

3.4.2	Controlador	22
3.4.3	Vista	23
3.5	Plataformas	23
3.5.1	Firestore	23
3.5.2	Integrated Development Environment (IDE)	23
3.5.3	AndroidStudio	24
3.5.4	GitHub	24
4	ESTADO DEL ARTE	25
4.1	UBER	25
4.2	WHEELS	27
5	METODOLOGÍA	29
6	DESARROLLO DEL PROYECTO	30
6.1	REQUERIMIENTOS INICIALES	30
6.2	PROTOTIPO FUNCIONAL	31
6.3	PRIMER PROTOTIPO	35
6.4	SEGUNDO PROTOTIPO	38
6.5	TERCER PROTOTIPO	42
6.6	CUARTO PROTOTIPO	46
7	CONCLUSIONES	48
8	RECOMENDACIONES EN UN FUTURO	49
	BIBLIOGRAFÍA	50
	ANEXOS	52

Lista de figuras

Figura 1: Pantalla pedir uber de la aplicación UBER. Fuente UBER.....	26
Figura 2: Pantalla principal de Wheels Fuente Wheels	28
Figura 3: primer prototipo: menú y pantalla principal.....	36
Figura 4: Primer prototipo pantalla principal mostrando rutas creadas.....	38
Figura 5: Segundo prototipo módulo perfil de usuario.	40
Figura 6: Tercer prototipo, módulo de una ruta especifica.....	43
Figura 5: Tercer prototipo módulo perfil de usuario.	44

Lista de tarjetas CRC

Tarjeta CRC Buscar ruta.....	31
Tarjeta CRC Usuario.....	32
Tarjeta CRC Pasajero.....	32
Tarjeta CRC Conductor	33
Tarjeta CRC Ruta	33

RESUMEN

TÍTULO: DESARROLLO DE SOFTWARE MÓVIL PARA LA GESTIÓN DE LA MOVILIDAD DE LA COMUNIDAD UNIVERSITARIA HACIA Y DESDE EL CAMPUS.

AUTORES: JUAN FELIPE RUEDA ROMÁN, JAVIER EDUARDO CASTELLANOS ONTIVEROS

PALABRAS CLAVE: Backend as a Service, Android, Firebase, JAVA, XML, carpooling.

DESCRIPCIÓN

Los sistemas de transporte público de la ciudad de Bucaramanga lugar donde se encuentra ubicada la ciudad la Universidad Industrial de Santander son, en algunos casos, deficientes y de mala calidad, debido a esto y aprovechando el hecho de que algunos integrantes de la comunidad universitaria cuentan con medios de transporte privados que no están utilizando en su totalidad, se pensó en desarrollar una aplicación que ayudase a conectar a dichos integrantes de la comunidad y se contribuyera a la solución de este problema a través de esta iniciativa de carpooling.

Dicha aplicación móvil se desarrolló para el sistema operativo Android, que es el más usado en la comunidad universitaria, esta, permite a los usuarios tanto publicar como unirse a rutas que les sean de su conveniencia, además, cuenta con un sistema de calificaciones y reportes al usuario.

La aplicación fue desarrollada haciendo uso del entorno de desarrollo integrado (IDE) Android Studio oficial para el desarrollo de aplicaciones Android, este usa el lenguaje JAVA, y XML, la aplicación también usa Firebase como su Backend as a Service(BaaS).

* Trabajo de grado.

**Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: MSc. Fernando Antonio Rojas Morales.

ABSTRACT

TITLE: DEVELOPMENT OF A MOBILE SOFTWARE FOR THE ARRANGEMENT OF THE MOBILITY OF THE UNIVERSITY COMMUNITY FROM AND TO THE CAMPUS.

AUTHORS: JUAN FELIPE RUEDA ROMÁN, JAVIER EDUARDO CASTELLANOS ONTIVEROS

KEYWORDS: Backend as a Service, Android, Firebase, JAVA, XML, carpooling.

DESCRIPTION

The public transport systems of the city of Bucaramanga place where is located the Industrial University of Santander, are, in some cases, deficient and of poor quality, due to this and taking advantage of the fact that some members of the university community count with private ways of transport that they are not using in their totality, it was thought to develop an application that helped to connect those members of the community and contributed to the solution of this problem through this initiative of carpooling.

This mobile application was developed for the Android operating system, which is the most used in the university community, this allows users to both publish and join routes that are of their convenience, in addition, has a system of qualifications and reports to the user.

The application was developed using Android Studio that is the official Integrated Development Environment (IDE) for Android app development, this uses the JAVA language, and XML, the application also uses Firebase as its Backend as a Service (BaaS).

* Bachelor Thesis.

** Faculty of Physicomechanical Engineering. School of Systems Engineering and Informatics. Director: MSc. Fernando Antonio Rojas Morales.

INTRODUCCION

El transporte público suele ser un problema en la mayoría de las grandes ciudades, malas planificaciones, sobrepoblación, mal manejo de espacios públicos, una infinidad de causas hacen que tanto el transporte público como la movilidad en general sean un problema en las grandes ciudades.

Teniendo en cuenta estas problemáticas presentadas, este proyecto de grado plantea una aplicación móvil que permita a usuarios propietarios de medios de transporte privados, compartir sus vehículos, con otros usuarios interesados en sus rutas, para que de este modo se reduzcan no solo el número de usuarios demandantes de medios de transporte para movilizarse hacia y desde la Universidad, sino también el número de vehículos que no están siendo usados en su totalidad, ya sea 5 o más, el cupo máximo de estos, logrando así una eficiencia mayor en su uso diario, y siendo un alivio en cierta forma a los medios de transporte habituales como también a el alto flujo vehicular de la ciudad.

Se decidió desarrollar esta aplicación para el sistema operativo Android, que es el más usado por la comunidad universitaria en general, usando el IDE Android Studio, y Firebase como nuestro BasS, se plantea una aplicación que permite a los usuarios publicar y ver las rutas en tiempo real, interactuar con los conductores además de otros pasajeros, calificar a los usuarios, generar reportes que ayudan a mantener el orden y control de usuarios.

La aplicación suple la necesidad básica de interacción y comunicación entre los usuarios, pero debe ser mejorada a futuro para que esta permita ver las rutas en un mapa facilitando al usuario encontrar la ruta más adecuada para él, además de implementar un sistema de chat propio que facilite la comunicación entre los usuarios.

En este documento se podrá encontrar los soportes teóricos de nuestra investigación además de los métodos usados para desarrollar esta misma.

1. PLANTEAMIENTO Y JUSTIFICACIÓN DEL PROBLEMA

El transporte público en el área metropolitana de Bucaramanga es ineficiente o de mala calidad, día a día estudiantes como profesores o miembros de la comunidad universitaria en general, usan estos medios para acudir a clases, viéndose perjudicados de una u otra forma en el desempeño normal de sus actividades académicas, además de, en el peor de los casos, se vean forzados a usar medios ilegales de transporte, poniendo en riesgo su seguridad.

Algunos miembros de la comunidad universitaria, cuentan con medios de transporte privados, los cuales no usan en su totalidad, es por esto que se plantea a través del desarrollo de un aplicativo móvil, crear una plataforma que permita la interacción entre estos dos grupos de la comunidad, logrando así suplir la necesidad de un medio de transporte que permite a los miembros de la comunidad desplazarse desde y hacia la Universidad de forma rápida, segura y efectiva.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

Desarrollo de una aplicación móvil para el sistema operativo Android, de tipo carpooling que permita a la comunidad universitaria compartir lugar en sus vehículos para el traslado desde casa a la universidad y viceversa.

2.2 OBJETIVOS ESPECÍFICOS

- Determinar las características de una aplicación móvil tipo carpooling.
- Diseñar e implementar la aplicación en el paradigma de orientación a objetos para el sistema operativo Android.
- Diseñar e implementar una base de datos para la persistencia de objetos de rutas y usuarios.
- Realizar pruebas de funcionalidad de la aplicación.
- Documentar el código de la aplicación utilizando la herramienta javadoc.

3. MARCO DE REFERENCIA

3.1 Terminología:

3.1.1 Carpooling:

Es el intercambio de nuestro coche o de las demás personas lo compartan para que más de una persona viaje en este.

Al compartir un vehículo todas las personas se benefician ya que sus costos en este viaje se ven reducidos ya que no se cobra por una sola persona algunos gastos ya que estos serán divididos entre los participantes del viaje algunos de estos gastos serian combustible, gastos de peaje y ya sea en viajes muy largos, poder tumarse para conducir.

Hacer uso de esta práctica vemos reducido la contaminación que se haría normalmente si no se compartiera el automóvil con otras personas, también notaremos un mejor tráfico que habría en la ciudad, ya que se están disminuyendo el tráfico y también la demanda por parqueaderos ya que se vería reducido este.

En el 2009 el uso compartido en el país de Estados Unidos esta práctica represento aproximadamente 43.5 % de todos los viajes.

3.1.2 Data Base as a Service:

Es un modelo basado en la nube que nos permite elegir un ambiente de base de datos por algunos modelos pre establecidos y utilizando un marco de trabajo de autoservicio. Este servicio se beneficia por ser más ágil y rápido a la hora del despliegue de la base de datos. Para que un servicio sea considerado como DBaaS debe cumplir unos requisitos como disponer del cliente bajo demanda sin que este necesite una instalación de algún software o hardware, que el pago sea solamente por su uso sin necesidad de algún contrato y el proveedor sea el responsable del servicio.

3.1.3 Backend as a Service:

Es un modelo que proporciona a los desarrolladores de aplicaciones una forma de vincularlas a back-end de almacenamiento en la nube lo que proporciona funciones como lo son notificaciones, administración de usuarios y poder integrar nuestra aplicación con los servicios de redes sociales, podemos acceder a todos estos servicios por medio de kits de desarrollo de software personalizados, últimamente este servicio ha ganado mucha popularidad por los desarrolladores de aplicaciones móviles y web's.

3.1.4 Web Service:

Se utiliza para la comunicación entre máquina y máquina más que todo con el idioma entre máquinas como xml y json, este servicio fue diseñado para interoperar una máquina desde otra haciendo el uso de la red. Las entidades competentes y responsables de la arquitectura y reglamentación del web service son OASIS y W3C.

Algunos de los estándares utilizados en este son Protocol, Statick, XML, SOAP, WSDL, entre otros.

3.2 Android

Este sistema operativo fue diseñado para dispositivos con la característica que tuviera pantalla táctil como tabletas, teléfonos, los relojes inteligentes, algunos televisores e inclusive coches que cuentan con estas características, este sistema operativo fue desarrollado por Android Inc. Pero la gran empresa Google respaldaba desde el comienzo a esta empresa para este desarrollo y en el 2005 compro Android y este sistema fue presentado en el 2007. El primer dispositivo con la gran suerte de tener este sistema operativo fue el HTC Dream y fue vendido en el año 2008 en el mes de octubre. El sistema operativo está compuesto de aplicaciones las cuales son ejecutadas en un framework Java las cuales son orientadas a objetos.

El código del sistema operativo Android fue lanzado por el mismo Google con una licencia de código abierto.

Este sistema se basa en el kernel de Linux.

3.2.1 Android sdk 16

Jelly Bean fue presentado el 24 de julio de 2012 en el evento "Breakfast with Sundar Pichai". Actualmente cuenta con el 1.3% de los usuarios que utiliza Android, esta versión implemento funciones en el sistema operativo como el soporte de Bluetooth de baja energía y se incluyeron idiomas para esta versión como hebreo, árabe, afrikáans, amhárico, hindi, suajili y zulú. Tiene implementado un sistema de localización wifi el cual esta activo en segundo plano, en esta versión tuvo mejoras en la seguridad del sistema.

Una nueva forma de escritura como lo fue el autocompletado y facilita poder marcar los números, se implementa Google now como asistente personal.

En cuanto a los desarrolladores, se incorporaron nuevas opciones a la hora de crear apps, mejoras en el análisis de compilación de aplicaciones y APIs. Una mejor seguridad a la hora de utilizar la encriptación WiFi, mejoras en los codificadores de vídeo, monitorización de notificaciones.

3.2.1 Android sdk 25

Android Nougat fue dado a conocer el 18 de mayo de 2016 en el evento Google I/O.

Android Nougat incluyo la característica llamada multi ventana, la cual nos permite usar 2 aplicaciones al mismo tiempo teniendo la pantalla dividida cada una para cada aplicación. Algunos teléfonos inteligentes ya contaban

con este sistema, pero ahora va a ser compatible con cualquier dispositivo que tenga esta versión de Android.

Algunos teléfonos implementaron la realidad virtual pero ahora esta va a ser compatible de forma nativa lo cual con cualquier dispositivo se podrá utilizar realidad virtual.

3.3 Lenguajes de programación:

3.3.1 Java:

Java es un lenguaje de programación orientado a objetos el cual fue diseñado para ser lo más simple posible, el propósito de java es permitir que los desarrolladores una vez escriban su código y este sea ejecutado en un dispositivo lo cual quiere decir una vez el código corre en un dispositivo no se tiene que recompilar. Java es uno de los lenguajes más populares que se están usando actualmente para desarrollar aplicaciones de cliente servidor con reporte aproximadamente de 10 millones de usuarios.

3.3.2 Json:

JSON, o lo que significa JavaScript Object Notation, es utilizado para el intercambio de datos. JSON es un subconjunto de cómo podemos describir objetos de JavaScript, este es considerado como un formato de lenguaje independiente.

Podemos comparar JSON con XML dándole algunas ventajas a JSON como lo que es el intercambio de datos ya que es más sencillo por el analizador sintáctico parser de JSON. En JavaScript, un texto JSON lo podemos analizar muy fácil invocando la función eval(), por esto la comunidad de desarrolladores AJAX ha aceptado a JSON.

3.3.3 XML:

XML, el cual tiene un significado en inglés como “eXtensible Markup Language”, el cual traduce “Lenguaje de Marcado Extensible”, es un meta-lenguaje el cual nos permite utilizar marcas que han sido desarrolladas previamente por el World Wide Web Consortium utilizado para almacenar datos en forma legible. A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones deben comunicarse entre sí o integrar información.

3.3.4 JavaScript:

Javascript es un lenguaje con muchas posibilidades, ya que con este lenguaje podemos crear programas pequeños que luego van a ser insertados en el código de una página web o en programas más grandes, se pueden crear diferentes efectos para que haya más interacción con los usuarios.

Este lenguaje es caracterizado por como se basan las acciones con menos restricciones. Es un sistema que utiliza Windows y X-Windows la mayor parte de este lenguaje se enfoca en describir objetos y funciones que respondan a las acciones que haga el usuario como el movimiento del mouse, oprimir teclas en el teclado entre muchos más.

Por otra parte existen dos tipos de JavaScript el que esta al lado de cliente el cual se ejecuta en el cliente, este se denomina Navigator JavaScript y también existe otro JavaScript que es por parte del servidor, este fue el segundo en salir y se denomina LiveWire JavaScript.

3.3.5 Node.js

Node.js es un entorno JavaScript por parte del servidor, es para la creación de eventos. Node ejecuta JavaScript y se utilizara un motor V8, el cual ha sido desarrollado por Google

para uso de su navegador Chrome. Utilizando el motor V8 permitira a Node realizar un entorno de ejecución por parte del servidor que compila y ejecuta JavaScript a una velocidad muy alta. Además de su velocidad de ejecución de JavaScript, lo más importante está en lo que Node.js hace lo cual es un bucle de eventos para tener gran volumen de clientes y todas sus operaciones se llevan a cabo de forma asíncrona. Node es de código abierto, y se ejecuta en Mac OS X, Windows y Linux.

3.4 Modelo Vista Controlador

3.4.1 Modelo

Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, como consultas o actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación. Envía a la vista aquella parte de la información que en cada momento se le solicita para que esta sea mostrada. Las peticiones de acceso o manipulación de la información se realizan por medio del controlador.

3.4.2 Controlador

Responde a eventos por lo general la hacen los usuarios e invoca peticiones a nuestro modelo cuando se hace alguna solicitud sobre la información. También puede enviar comandos a su vista asociada si se solicita un cambio en la forma en que se presenta el modelo.

3.4.3 Vista

Presenta el modelo en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.

3.5 Plataformas.

3.5.1 Firebase

Firebase es una plataforma fundada en 2011 y comprada posteriormente por Google en 2014, esta nos permite crear y desarrollar aplicaciones para dispositivos móviles de una forma muy rápida y muy buena calidad, esta plataforma tiene como objetivo facilitar la gestión de nuestra app aumentando la base de usuarios y por tanto su monetización.

Esta plataforma se podrá usar tanto para iOS como para Android y para web y ofrece al desarrollador diversas funcionalidades como lo son estadísticas seguridad entre muchas más, que el desarrollador podrá elegir y adaptar a la forma que este necesitando para la app que se encuentre desarrollando.

3.5.2 Integrated Development Environment (IDE)

Un entorno de desarrollo integrado o entorno de desarrollo interactivo, es una aplicación la cual se encarga de proporcionar servicios integrales para facilitar al desarrollador el desarrollo de un software.

Normalmente, un IDE consiste de un editor de código fuente, herramientas de construcción automáticas y un depurador.

3.5.3 AndroidStudio

Android Studio es el IDE oficial por Google para la plataforma Android. Se anuncio el 16 de mayo de 2013 en la conferencia Google I/O, y fue el que reemplazo a Eclipse y se convirtió en el IDE para el desarrollo de aplicaciones para Android. La primera versión estable fue publicada en diciembre de 2014.

Esta herramienta es totalmente libre y se encuentra disponible para su uso para todos los sistemas operativos actuales (Windows, MacOs y Linux.)

3.5.4 GitHub

Es una plataforma de desarrollo colaborativo en la cual podemos alojar proyectos utilizando un sistema de control de versiones, el código se almacena de forma pública, pero se puede hacer de forma privada utilizando la versión paga de github.

Características

- Tenemos un Wiki para cada proyecto
- Nos da la opción de tener un gráfico para ver cómo los desarrolladores progresan en sus repositorios y bifurcaciones del proyecto
- Es como una red social ya que nos brinda que nos pueden seguir personas, y también podremos seguir a otros desarrolladores
- Una herramienta para trabajar entre programadores.
- Gestor de proyectos de estilo Kanban2

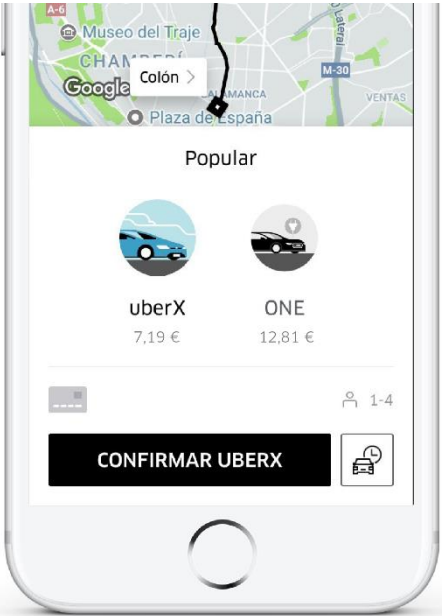
4 ESTADO DEL ARTE

La contaminación de CO por parte de los vehículos ha aumentado drásticamente con el paso de los años, para combatir este problema han ido naciendo diferentes plataformas tanto que ayudan al humano a transportarse como al medio ambiente para disminuir el CO producido por diferentes vehículos innecesariamente, algunas de estas plataformas son:

4.1 UBER:

Uber es líder en el mercado de transportar sus usuarios por conductores particulares algo parecido a un taxi, uber maneja distintos tipos de servicio uberX, uberPOOL, UberBLACK, UberVAN, UberANGEL, UberBICI, UberEATS. En este caso nos concentraremos en uberPOOL ya que este es el mismo servicio que se presta con uberX pero con la posibilidad que un usuario que se dirige a un destino que este dentro de la ruta que escogió el usuario principal pueda unirse a la ruta sin ningún inconveniente y llevarlo, el pago se vería reducido para el cliente principal (el que escogió el destino principal). Por medio de este método todos ganan tanto el usuario principal como los usuarios que se unieron a la ruta y el medio ambiente ya que se redujo la contaminación de CO ya que no se necesitaron 3 o 4 vehículos para llegar a sus destinos, sino solamente uno. Pero Uber no cuenta con estos sistemas en todas las ciudades del país, en Bucaramanga no contamos con la opción de pedir un uberPool, por lo tanto esta aplicación no nos serviría y aun así si lo implementaran esta plataforma se encuentra actualmente con problemas legales y se está viendo afectada al momento de utilizarse y no tendría buen acogida por la ciudadanía.

Figura 1: Pantalla pedir uber de la aplicación UBER. Fuente UBER.

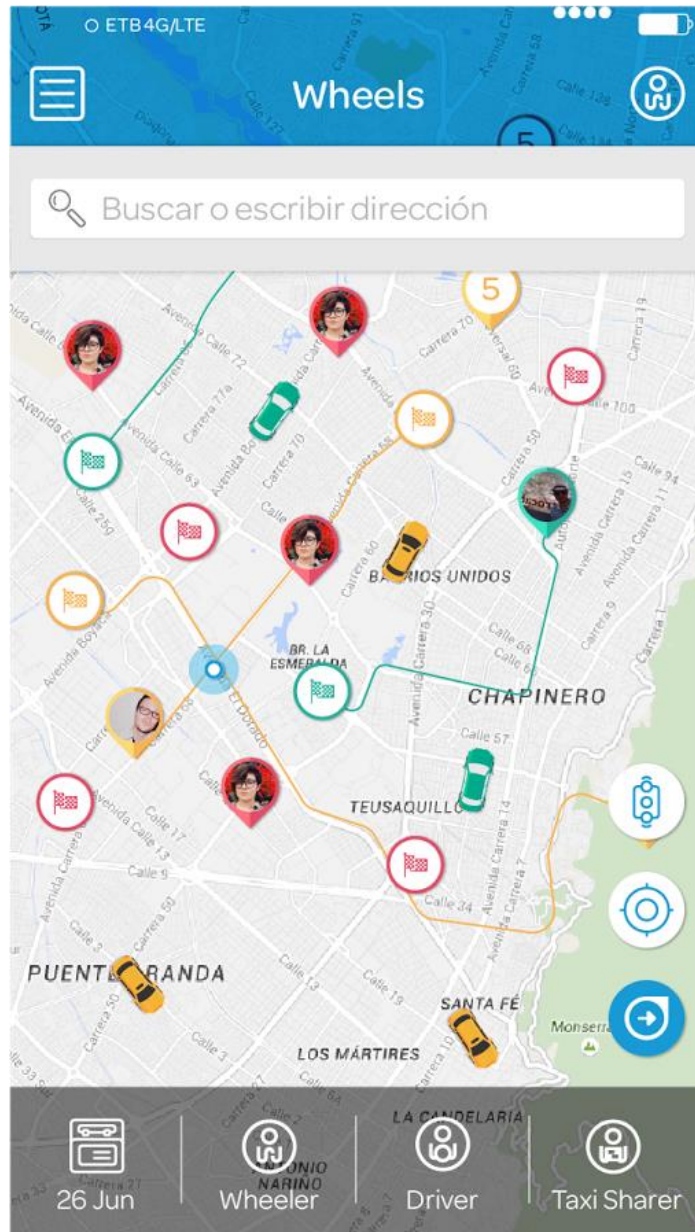


4.2 WHEELS:

Wheels es una plataforma la cual se encarga de compartir vehículos, servicios de taxi o planes que se realicen en bicicleta, en este caso nos enfocaremos en compartir el vehículo o el servicio de taxi. Wheels consiste en definir un grupo de personas o ingresar a grupos de instituciones o empresas, el siguiente paso a realizar en la plataforma es definir la ruta que se está creando si eres conductor o si eres un pasajero esperando a que alguien te de cupo en su ruta, si eres el creador de una ruta en taxi, por último definir con quienes vas a viajar enviando una notificación a los que te llamen la atención para concretar los detalles del viaje y como se encontraran.

Esta aplicación desde nuestro punto de vista tiene falencias en cuanto a usabilidad, una vez la inicias, no se puede encontrar un modo de publicar tu ruta fácilmente e igual las rutas disponibles no se pueden ver de manera fácil lo que hace que los usuarios desistan de usar esta aplicación rápidamente.

Figura 2: Pantalla principal de Wheels Fuente Wheels



5 METODOLOGÍA

Desarrollo evolutivo:

Consiste en una idea de desarrollar una implementación inicial, presentándola a los usuarios y que estos nos den comentarios, y por medio de diferentes versiones ir perfeccionando nuestro sistema hasta que se desarrolla un sistema correcto. Las actividades de especificación, desarrollo y validación se entrelazan en vez de separarse, con una rápida retroalimentación entre estas.

La ventaja de un proceso del software que se basa en un enfoque evolutivo es que la especificación se puede desarrollar de forma creciente. Tan pronto como los usuarios desarrollen un mejor entendimiento de su problema, éste se puede reflejar en el sistema software.

Para sistemas pequeños y de tamaño medio aproximadamente que no pasen de 500.000 líneas de código, el enfoque evolutivo de desarrollo es el mejor. Los problemas del desarrollo evolutivo para sistemas grandes y complejos que su desarrollo toma mucho tiempo, donde diferentes equipos desarrollan distintas partes del sistema este uso no es recomendable para el desarrollo evolutivo. Para establecer una arquitectura al sistema se dificulta ya que se hace difícil integrar las contribuciones de los diferentes equipos. Para sistemas grandes, es recomendable realizar un proceso mixto entre el modelo cascada y el modelo evolutivo ya que puede implicar que se desarrolle un prototipo desechable para resolver algunas dudas que se tengan de la especificación del sistema y se puede reestructurar este sistema ya que las partes del sistema que se encontraban bien diseñadas se pueden desarrollar con el modelo cascada.

6 DESARROLLO DEL PROYECTO

6.1 REQUERIMIENTOS INICIALES

Para determinar los requisitos iniciales que debía satisfacer nuestra aplicación, se decidió dar una mirada al sistema que están usando actualmente los estudiantes y comunidad UIS para trasladarse usando el carro compartido, el grupo llamado “UIS wheels”, donde estudiantes y comunidad UIS en general, interactúan mediante publicaciones y comentarios en estas mismas para llegar a un común acuerdo entre el conductor y los interesados de la ruta de auto compartida.

Los principales requerimientos que esto genera son los siguientes

- Visualizar una lista de las rutas activas en el momento además de permitir una búsqueda entre estas.
- Publicar una ruta con destino y hora.
- Establecer comunicación entre usuarios.

Se plantea que, dentro del alcance de este proyecto, la aplicación cumpla con estos requerimientos básicos, pero a futuro, la aplicación debería responder a requerimientos tales como:

- Poder visualizar las rutas en tiempo real mediante un mapa de la ciudad.
- Implementar un sistema de chat propio de la aplicación.
- Sistemas de calificación entre usuarios.

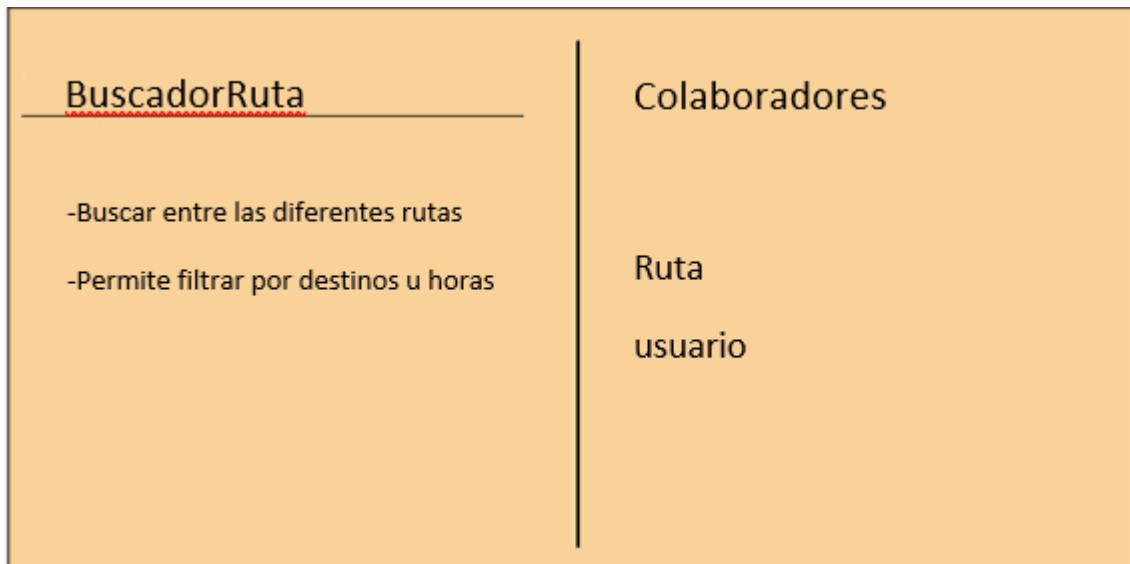
Se decidió también diseñar la aplicación siguiendo los lineamientos de” Material Desing”.

6.2 PROTOTIPO FUNCIONAL

Para el desarrollo del primer prototipo se tomó la realimentación obtenida del prototipo no funcional y se plantearon tarjetas CRC, técnica creada por Kent Beck y Ward Cunningham. Usada para el diseño de software orientado a objetos consiste en dibujar una tarjeta por cada clase u objeto y dividirla en tres zonas, el nombre de la clase, las responsabilidades de esta y sus colaboradores.

Primeras tarjetas CRC

Tarjeta CRC Buscar ruta



Tarjeta CRC Usuario

Usuario	Colaboradores
<ul style="list-style-type: none">-Permite crear nuevos usuarios además del uso del perfil de Facebook para dicho fin-Comunica usuarios entre si-Permite reportar otros usuarios-Permite ver perfiles de otros usuarios-lleva conteo de reportes al usuario	

Tarjeta CRC Pasajero

Pasajero	Colaboradores
<ul style="list-style-type: none">-Permite calificar conductores-Hereda de usuario	usuario

Tarjeta CRC Conductor

Conductor	Colaboradores
<ul style="list-style-type: none">-Permite publicar rutas-Hereda de Usuario-Permite agregar especificaciones del automóvil-Definir cupo del <u>automovil</u>	usuario

Tarjeta CRC Ruta

Ruta	Colaboradores
<ul style="list-style-type: none">-Permite la creación del recorrido por el usuario con un destino, origen, hora y cupo-Visualización en el mapa-Muestra cupos disponibles en rutas afines-Muestra la información del conductor que postea la ruta	usuario

A partir de estas tarjetas se realiza el primer diseño de clases a utilizar en este proyecto, como base para iniciar el desarrollo del primer prototipo que fue evolucionando hasta convertirse en el prototipo final.

6.3 PRIMER PROTOTIPO

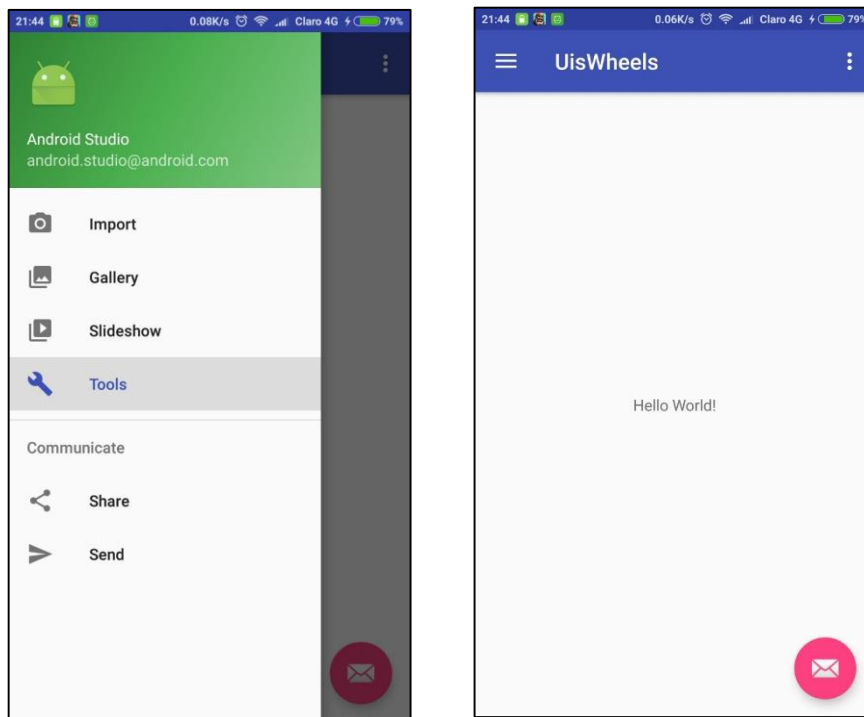
Para el desarrollo del primer prototipo se pensó en dar solución al primer requerimiento, la visualización de las rutas en tiempo real, se investigó por la web la mejor opción a la hora de establecer una base de datos que nos permitiera obtener esta funcionalidad, que fuese de bajo costo, sostenible y se adaptara a las necesidades que teníamos en ese momento.

Lo más usual es adquirir un servicio web (en inglés, web service o web services), con Php, Mysql y Json, pero el tiempo de implementación de estas tecnologías era alto, se siguió investigando y se encontró la alternativa Firebase que es un Backend como servicio, aunque no se tenía conocimiento al respecto y el tiempo de implementación podría llegar a aumentar debido a esto, se decidió optar por esta opción pues ofrecía ventajas tales como:

- Facilidad a la hora de despliegues para Android.
- Al ser una Backend como servicio no se necesita un Servicio Web en primera instancia.
- Costos, Firebase es gratis para nuevos usuarios que tienen consumos moderados de la base de datos.
- Al ser un servicio de google se obtiene confiabilidad y completa integralidad con Android.
- Fácil escalabilidad.
- Ofrece el servicio de base de datos en tiempo real, que es justo lo que se estaba buscando para la solución de este requerimiento.

Una vez decidido usar Firebase como nuestro BaaS, se dispuso a diseñar el prototipo inicial usando Android Studio, se pensó en una interfaz simple siguiendo los parámetros del Material Design se obtuvieron los siguientes resultados:

Figura 3: primer prototipo: menú y pantalla principal



Una vez establecida la pantalla principal de la aplicación se procede a llenar el contenido de esta obteniendo las rutas alojadas en la base de datos en tiempo real, para esto se usa un RecyclerView y se procede a poblarlo en tiempo real con la funcionalidad de la base de datos de Firebase.

La base de datos de Firebase, almacena los datos como como objetos JSON. La base de datos puede conceptualizarse como un árbol JSON alojado en la nube.

Este es un ejemplo de la primera estructura creada para almacenar las rutas:

```
-KxQi4ydvMZnZ8J3wTN7
  ..... activa: "false"
  ..... autor: "Johnny David Hernández"
  ..... cupo: "3"
  ..... descripcion: "calle 105 autopista cra
  ..... destino: "UIS"
  ..... disponible: "3"
  ..... fecha: "27/10/2017"
  ..... hora: "06:29"
  ..... origen: "Coaviconsa PROVENZA"
  ..... userid: "0vA5NwkNIQh2hQs018BM3D0RAH5"
```

Firestore permite a través de una función dotar con un identificador único a un campo de un objeto permitiendo de esta forma, diferenciar todos los objetos creados dentro de otro objeto, en este caso, cada ruta tendrá un identificador único que la diferencia de las otras permitiéndome de esta forma acceder a esta independientemente, también me permite obtener estos identificadores para que en dado caso pueda almacenar un identificador de otro objeto dentro de este mismo objeto, un ejemplo en este caso es si quisiera guardar el identificador del usuario en un campo de esta ruta permitiéndome así saber en cualquier momento quien es el usuario dueño de dicha ruta.

Una vez poblada la base de datos con rutas de prueba, se procede a acceder a la aplicación y encontramos el siguiente resultado:

Figura 4: Primer prototipo pantalla principal mostrando rutas creadas.



Como se puede observar se visualizan las rutas, y los cambios en las rutas se pueden ver en tiempo real

6.4 SEGUNDO PROTOTIPO

Una vez resuelto el requisito principal, se proceden a agregar características a la aplicación tales como un sistema de autenticación que diferencie los usuarios, formas de publicar las rutas, perfiles de usuario, accesos rápidos para las rutas publicadas por cada usuario entre otros. El resultado es una aplicación más funcional que permite el desarrollo normal del usuario.

Para implementar el sistema de autenticación se usó el sistema de autenticación propio de Firebase que “busca facilitar la creación de un sistema de autenticación seguro, a la vez que mejora la experiencia de incorporación y acceso para los usuarios finales. Proporciona una solución de identidad de extremo a extremo, compatible con cuentas de correo electrónico/contraseña, autenticación telefónica, Google, Twitter, Facebook, acceso de GitHub y otros”.

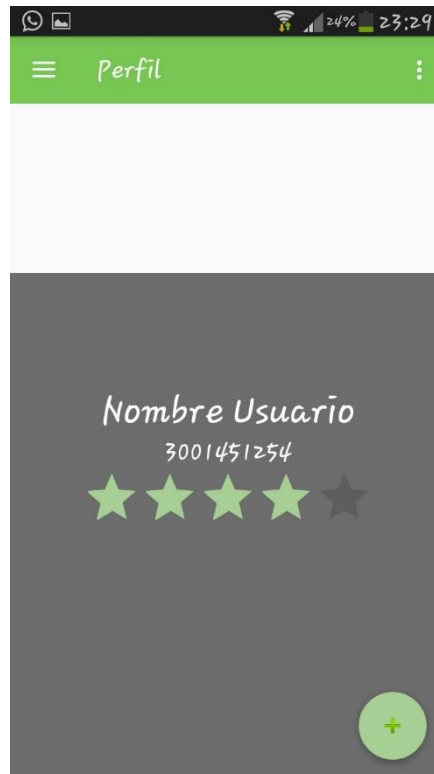
En un principio se implementó el sistema de autenticación simple con correo electrónico y contraseña, pero se planea agregar el método de autenticación con redes sociales a futuro.

Una vez creado el sistema de autenticación para el acceso a los usuarios, implementar un perfil de usuario es el paso a seguir, el sistema de autenticación de Firebase nos otorga un identificador único para cada usuario asociado a su correo electrónico, esta es la clave para identificar a los usuarios que acceden a nuestra aplicación, para almacenar los datos de los usuarios se establece la siguiente estructura de datos JSON:

```
0QYo0pSTPkhSv41c5pAzogeaMuH3 + x
+----- calificacion
----- device_token: "dIocICsPmao:APA91bFm0N11PySxAmRIM1CQZdHoRXGZAI
----- name: "Sebastián Blanc
----- phone_number: "3164595276
A----- profile_pic: "https://firebasestorage.googleapis.com/v0/b/uiis erar
esta información para su edición o simplemente su visualización, además de la
```

creación de una actividad que muestre la información pública y permita a otros usuarios acceder a esta información; el diseño de esta pantalla es el siguiente:

Figura 5: Segundo prototipo módulo perfil de usuario.

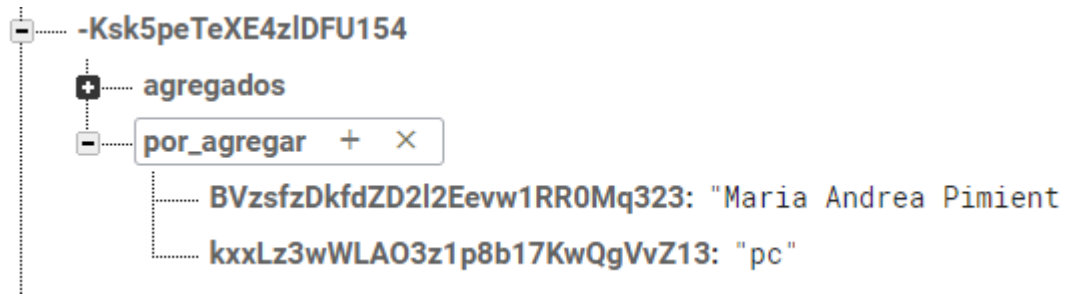


Por el momento la imagen del usuario está vacía pues esta requiere acceder a el módulo de almacenamiento de Firebase y se planeó implementar en versiones futuras.

El procedimiento para crear un panel donde el usuario pudiera ver las rutas que el mismo ha creado fue sencillo puesto que se repitió el mismo procedimiento usado para crear la pantalla principal se filtraron las rutas que fuesen solo del usuario que estaba accediendo en el momento.

Una vez creados los usuarios se procede a añadir un campo a el objeto JSON de la ruta para que cada ruta tenga un campo donde se establece el creador de la ruta,

en este campo se almacena el id del usuario creado por Firebase, y se procede a implementar el botón que permite a un usuario unirse a otra ruta, para esto se decide crear otro objeto JSON que almacene todos los usuarios de cada ruta puesto que permite el acceso a estos datos, la estructura es la siguiente:

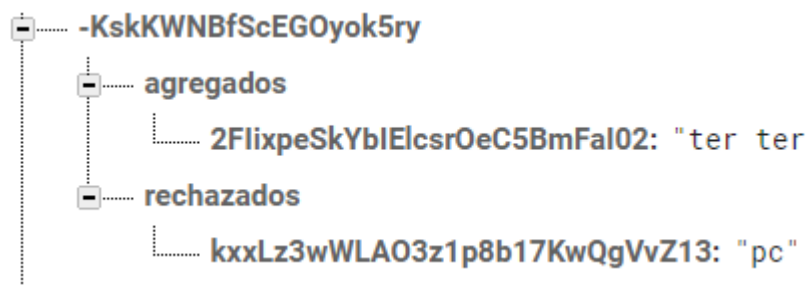


De esta forma se lleva el control de los usuarios que se unieron a dicha ruta.

6.5 TERCER PROTOTIPO

Una vez solucionados los requisitos básicos para un funcionamiento básico de la aplicación se procede a implementar funcionalidades más complejas pero que son fundamentales para una correcta interacción con el usuario.

La primera funcionalidad a implementar es permitir a un dueño de una ruta, aceptar o rechazar a un usuario que solicite unirse a su ruta, además de la posibilidad de expulsar a uno que ya esté en su ruta actualmente, para esto se decide cambiar la estructura del objeto JSON "PasajerosRuta" de la siguiente manera:



De esta manera se facilita el acceso a la información acerca de la ruta para el usuario dueño de la ruta, solo resta proceder a crear una vista de la ruta donde se puedan acceder a estas listas de usuarios, y que el usuario dueño de la ruta disponga de botones que le permitan aceptar, rechazar o expulsar a los usuarios que están interactuando con su ruta, el diseño para esta pantalla es el siguiente:

Figura 6: Tercer prototipo, módulo de una ruta específica.



Como se puede observar el usuario ya fue agregado y ahora puede ser expulsado si el usuario dueño de la ruta así lo desea.

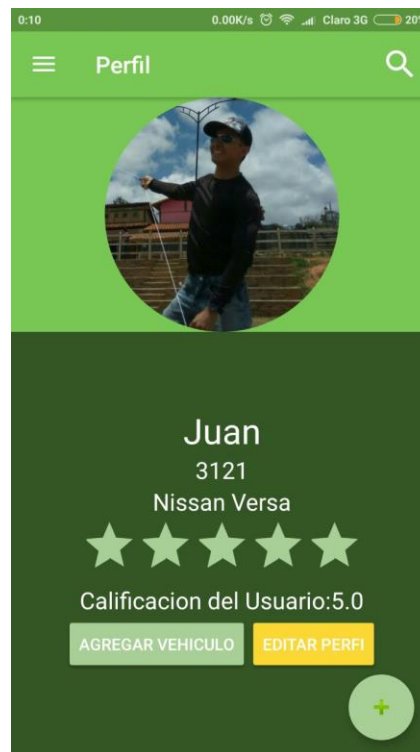
También se implementó un diseño para el perfil de usuario mejorado donde ya se puede visualizar la imagen y se agregaron funcionalidades como la de editar perfil, agregar vehículos, y mostrar calificación del usuario.

Con el nuevo diseño del perfil nuevas funcionalidades son requeridas, como la implementación de un botón que me permita calificar a un usuario, permitiendo así a otros usuarios ver la calificación de un usuario predeterminado y decidir de esta forma si desea interactuar con este a la hora de planificar sus rutas, también se

agregó un campo al objeto “User” de la base de datos donde se almacenan los datos correspondientes a el automóvil del usuario si es que este posee uno.

El diseño del perfil es el siguiente:

Figura 5: Tercer prototipo módulo perfil de usuario.



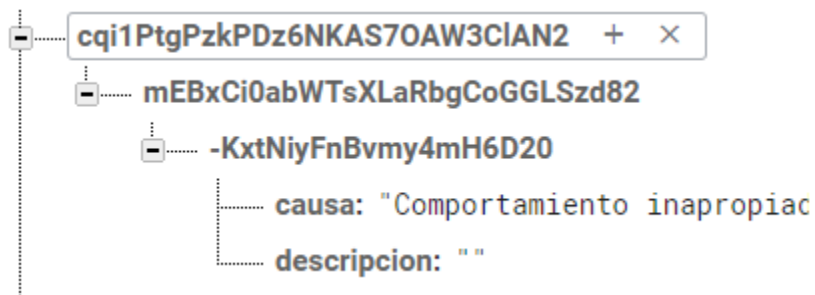
Para el almacenamiento de la imagen de perfil del usuario se hizo uso del sistema de almacenamiento de Firebase el cual mediante un sencillo funcionamiento me permite crear una carpeta la cual hemos llamado “imágenes_de_perfil”, en la cual se almacenan las imágenes con el nombre que les damos y le asigna un link único que es el que usaremos para descargar dicha imagen, este link se almacena en el objeto “user”, en el campo ‘imagen’, de allí se obtiene cada vez que se necesita mostrar dicha imagen.

Android Studio por defecto no permite mostrar imágenes redondeadas, para esto se usó la librería Picasso que es una poderosa librería para obtener y descargar imágenes para Android.

6.6 CUARTO PROTOTIPO

Una de las principales funcionalidades faltantes de la aplicación es el método de búsqueda de las rutas activas, así que se implementó un botón de búsqueda en la pantalla principal que permite a los usuarios buscar entre las rutas más adecuada para realizar su trayecto y de esta manera poder unirse a ella.

También se implementó la funcionalidad de reportar usuario que guarda en un objeto JSON llamado 'Reporte' los reportes realizados a un usuario por otros usuarios otorgando así la posibilidad de banear a usuarios que realicen acciones inapropiadas, la estructura de dicho objeto es la siguiente:



Además, se implementaron funcionalidades para brindar a los usuarios una mejor experiencia usando la aplicación entre estas

- Funcionalidad de reporte de errores mediante correo electrónico.
- Recuperación de contraseña con correo.
- Edición de datos de usuario (nombre, número, automóvil registrado, imagen de perfil, entre otros)
- Notificaciones completamente funcionales que indican acerca de modificaciones en rutas propias o rutas a las que el usuario se está intentando unir.

- Herramienta de Analytics que nos permite ver el comportamiento de los usuarios en tiempo real.
- Se implementó la herramienta de Firebase de testeo.
- Se implementó un módulo de configuración remota que nos permite saber cuándo el usuario está usando una versión desactualizada que no es compatible con el sistema actual.
- La aplicación se encuentra disponible para su descarga en la Play Store de Google pero a la fecha solo está disponible en su versión Alhpa.

7 CONCLUSIONES

- Una aplicación de carpooling como Uiswheels contribuye a aliviar la problemática de movilidad planteada.
- La aplicación móvil cumple con los requisitos necesarios para permitir a los usuarios compartir el automóvil para dirigirse desde y hacia la Universidad.
- Usar Firebase como nuestro servicio Backend, permite que la aplicación sea más ágil y funcional, con su amplia bandeja de herramientas y su facilidad de implementación hacen de UisWheels una aplicación fiable que cumple con las expectativas de los usuarios.

8 RECOMENDACIONES EN UN FUTURO

- Para poder cubrir toda la comunidad universitaria y que la aplicación sea un total éxito se recomienda desarrollar la aplicación para iOS y además una versión web.
- Es recomendable añadir funcionalidades como el uso de mapas para mostrar las rutas y un chat para usuarios.

BIBLIOGRAFÍA

JSON, JSON, [en línea][citado en:18/05/2017] disponible en: <http://json.org/>

DESARROLLO WEB, Modelo vista controlador, [en línea] [citado en: 18/05/2017] disponible en: <https://desarrolloweb.com/articulos/que-es-mvc.html>

FIREBASE, FIREBASE, [en línea] [citado en: 18/05/2017] disponible en: <https://firebase.google.com/?hl=es-419>

ANDROID, Android Studio, [en línea] [citado en: 18/05/2017] disponible en: <https://developer.android.com/studio/index.html?hl=es-419>

ANDROID, ANDROID, [en línea] [citado en: 18/05/2017] disponible en: <https://developer.android.com/index.html>

ANDROID, Android 7.0 Nougat, [en línea] [citado en: 18/05/2017] disponible en: https://www.android.com/intl/es-419_mx/versions/nougat-7-0/

ANDROID, Android Jelly Bean, [en línea] [citado en: 18/05/2017] disponible en: https://www.android.com/intl/es-419_mx/versions/jelly-bean-4-3/

ANDROID, ANDROID, [en línea] [citado en: 18/05/2017] disponible en: https://www.android.com/intl/es_es/

FIREBASE, FIREBASE AUTHENTICATION [en línea] [citado en 25/09/2017] disponible en: <https://firebase.google.com/products/auth/>

PICASSO, FIREBASE AUTHENTICATION [en línea] [citado en 25/09/2017]
disponible en: <http://square.github.io/picasso/>

Ian, Sommerville “Desarrollo evolutivo” en: Ingeniería de software Novena edición,
Editorial Pearson.

Simon Bennett, Steve McRobb, Ray Farmer “Java (lenguaje de programación) en:
Análisis y diseño orientado a objetos de sistemas usando UML Tercera edición,
Editorial McGraw-Hill.

ALONSO RODRIGUEZ ZAMORA “Extensible Markup Language (XML)” en:
Publicación en internet y tecnología xml, Editorial RA-MA.

Ramos Salavert, Isidro; Lozano Pérez, María Dolores “Entorno de desarrollo
integrado (IDE)” en: Ingeniería del software y bases de datos: tendencias actuales,
Universidad de Castilla La Mancha, 2000.

ANEXOS

Para la realización de este cuestionario se usó la herramienta de formularios de Google, la cual permite una fácil creación, visualización y análisis. Presentaremos la estructura del cuestionario, y enseguida resultados de esta.

¿Qué fue lo que más le gustó de la app? *

Texto de respuesta larga

¿Qué fue lo que menos le gustó de la app? *

Texto de respuesta larga

¿Tuvo algún problema usandola? *

Sí

No

Describe su problema

Texto de respuesta larga

¿Crees que le falta alguna funcionalidad a la app?(si es así, describela) *

Texto de respuesta larga

Que metodo de verificacion de usuario te parece más efectivo

- Código de estudiante
- Correo institucional
- Otra...

Si tienes alguna recomendación o comentario escríbelo acá.

Texto de respuesta larga

A continuación serán las respuestas que obtuvimos en nuestra primera encuesta.

Marca temporal	¿Qué fue lo que más le gustó de la app?	¿Qué fue lo que menos le gustó de la app?	¿Tuvo algún problema usandola?	Describe su problema
11/10/2017 21:22:48	La facilidad de uso, es intuitiva	No tener claro los cupos disponibles en cada ruta	No	
23/10/2017 23:00:41	La vista del perfil	Que no hay mapa	Si	No pude especificar que es una moto
26/10/2017 7:02:31	Es concreta, no se generan confusiones en su uso.	Sería mejor con el código de estudiante y contraseña de plataforma, así no se meten personas ajenas a la universidad, y se evitan problemas.	No	
26/10/2017 7:05:21	Es mucho más fácil buscar una ruta en la app, que ponerse a bajar y bajar en el grupo de Facebook buscando alguna que sirva	La descargué hace 15 minutos y ya van 2 veces que que sale "La aplicación ha dejado de funcionar"	Si	No encuentro cómo cancelar un servicio. Me uní a una ruta para probar la app y ahora no soy capaz de cancelar.
26/10/2017 7:15:09	Es sencilla.	De primera mano no dan la opción de registrar el carro.	Si	Cuando iba a publicar la ruta me decía que registrara el carro y no encontraba dónde.
26/10/2017 8:16:57	Interfaz sencilla	No se puede cancelar una ruta	Si	No se puede cancelar una ruta
26/10/2017 8:17:57	la identificación rápida del conductor y su vehículo	no se puede llamar a las personas que se unen al grupo o al conductor que he elegido	No	
26/10/2017 8:24:57	Lo sencillo para ver las rutas y los cupos que quedan	Nada, todo lo vi bien	No	
26/10/2017 8:58:58	El diseño	Todo me gusto	No	
26/10/2017 9:17:50	La facilidad de uso	Se crean las cuentas muy lento, al usar el buscador y colocar algo que no existe; no se sabe si está cargando o no hay coincidencias	Si	La app se ceró al bus car uis en el buscador
26/10/2017 9:26:00	La plataforma de la app	Que no hay manera de desunirse de la ruta si por error se unió o ya no lo necesita	Si	Al entrar al perfil me reporta que la aplicación dejó de responder y se cierra
26/10/2017 9:56:22	La facilidad de publicar la ruta y de buscar las rutas	Por ahora todo bien	No	
26/10/2017 11:45:49	El diseño de la plataforma	La interfaz, como se muestra la información	Si	Falla en algunos momentos y se cierra la aplicación (entendible al estar en esta fase) del resto trabaja muy bien en un Asus ZenFone 3+

¿Crees que le falta alguna funcionalidad a la app?(si es así, descríbela)	Que metodo de verificacion de usuario te parece más efectivo	Si tienes alguna recomendación o comentario escríbelo acá.
Filtrar por lugares del recorrido	Codigo de estudiante	
Un mapa.	Codigo de estudiante	El mapa.
No	Codigo de estudiante	
No se me ocurre nada por ahora	Codigo de estudiante	
No	Considero que no debe existir metodo, simplemente que la aplicación solo se comparta por el grupo o se cree algún codigo de descarga si es posible. He visto papás de estudiantes también transportando gente y es más sencillo.	
Un chat entre conductor y pasajeros para acordar sitio de recogida, hora, etc..	Codigo de estudiante	Además de ver quien es el conductor, ver quienes son los pasajeros también.
si, que tenga la ubicacion en tiempo real. Tambien conocer la identificacion de las personas que se unen a mi ruta	Codigo de estudiante	cual es la capacidad de pasajeros que tiene el medio de transporte. Acordar el punto de encuentro del conductor con los pasajeros. Que se puedan realizar llamadas y enviar mensajes directos.
No	Codigo de estudiante	
Un chat para hablar directamente con las p	Codigo de estudiante	
No creo	Codigo de estudiante	
Una especie de filtro por lugar de salida o llegada para que sea más fácil encontrar las personas que cubran esa ruta y una manera de comunicarse con el conductor	Codigo de estudiante	
No veo como definir los puntos de encuentro con el pasajero, o como comunicarme con ellos		
Poder de unirme de un wheel debido a un contratiempo o algo	Codigo de estudiante	Incluir un chat en vivo de las personas en el wheel , un mapa de la Universidad señalando la ubicación aproximada del vehículo y finalmente una lista de reglas de convivencia y uso de la aplicación

Marca temporal	¿Qué fue lo que más le gustó de la app?	¿Qué fue lo que menos le gustó de la app?	¿Tuvo algún problema usandola?	Describe su problema
26/10/2017 12:21:23	Primero que para poder ingresar a la app se tenga que hacer un registro de usuario y segundo la manera en la que se organizan las rutas tienen los datos necesarios (usuario, teléfono, lugares, cupos, hora), podrían complementar con las sugerencias pero todo va bien. Pero principalmente el emprendimiento.	Los usuarios pueden usar cualquier imagen, mejorar un poco en la veracidad de los datos y así aumentarían la seguridad de la app.	No	
26/10/2017 14:10:19	Lo facil de utilizar	Que se deberían verificar las cuentas ya que ya se encuentran cuentas falsas y no es conveniente por seguridad	Si	Hay un bug al momento de entrar a la seccion "Perfil", lo que provoca que la app se cierre
26/10/2017 16:57:09	buenos patrones de usabilidad, muy intuitiva, facil registro, facil forma de llenar datos del vehiculo, y las rutas, la busqueda, el horario, que se pudiera activar y desactivar la ruta, la forma de contactar, el uso de material design etc.	de momento nada.	No	
26/10/2017 22:56:39	Su diseño y el requerimiento de insertar una imagen	Hasta el momento nada	No	
27/10/2017 16:59:06	Saber cuántos cupos disponibles hay	Nada	No	
29/10/2017 19:06:27	me gusta la presentación que tiene	no entiendo como será la interacción con los pasajeros	Si	el sistema me dió aviso de un error y que se cerraría la app, se cerró y se abrió sola

¿Crees que le falta alguna funcionalidad a la app?(si es así, describela)	Que metodo de verificacion de usuario te parece más efectivo	Si tienes alguna recomendación o comentario escríbelo acá.
Sería de utilidad que le anexaran un mapa (por ejemplo usar google maps, no sé si sea posible), y la persona marque la ruta desde el lugar de inicio hasta el punto final, podría servir igualmente para indicar la ubicación de la persona y saber que rutas van a pasar cerca a donde se encuentra. También podrían crearse "paradas", sitios donde siempre se encuentren los estudiantes para que sea un poco mas cómodo.	Codigo de estudiante	Esta muy buena, pueden seguir mejorándola.
No	Correo institucional	
no, esta bastante completa para un primer lanzamiento.	Codigo de estudiante	
Poder publicar que ruta se necesita y a que hora	Codigo de estudiante	Me gustaría que la imagen que requieren sea una foto de la persona y no una imagen de cualquier tipo, así tendría conocimiento previo del rostro de la persona que me transportará hasta la universidad. Gracias por realizar la app, me gustaría crearán una versión para ios, hay personas que conozco utilizan uis wheels que tienen Iphone, pero entiendo que eso se desarrollará más adelante.
No	Correo institucional	
que se puedan cuadrar rutas permanentes, por ejemplo, todos los lunes a la misma hora por el mismo trayecto, sin tener que poner la fecha. tambien sería bueno que se pueda buscar como usuario y no como transportador una ruta poniendo el sitio de salida o la ubicación gps y la app busque las rutas que salgan mas cercanas	Codigo de estudiante	me parece que van por un excelente camino, de verdad sería muy muy util para todos