



TÉCNICO SUPERIOR EN DESARROLLO DE
APLICACIONES MULTIPLATAFORMA
Departamento de Informática

PROYECTO



Weenkvan
Manual Técnico

Autor:

Curso Académico:

Índice

1 Introducción	3
1.1 ¿Cómo surge la idea?	3
1.2 Antecedentes	3
1.3 Idea inicial	4
1.4 Evolución de la idea	4
1.5 Investigación de la competencia	4
Yescapa	4
Booking	5
2 Objetivos	6
3 Tecnologías involucradas	7
3.1 Android Studio	7
3.2 Firebase	7
3.3 Java	8
3.4 JSON	8
3.5 Git + GitHub	8
3.6 Dia - Diagram Editor	8
3.7 Visual Paradigm 16.3	8
3.8 Adobe Photoshop 2021	8
3.9 Icon Kitchen	9
4 Proceso de desarrollo	9
4.1.1 Base de datos	9
4.1.2 Diagramas de casos de uso	10
4.1.3 Diagrama de clases	12
4.2.1 GitHub	13
5 Proceso de Despliegue	17
6 Propuesta de mejora o trabajos futuros	18
7 Bibliografía	19

1 Introducción

Weeknvan es una aplicación pensada para dispositivos móviles con sistema operativo Android para la gestión de reservas sobre caravanas, realizadas entre usuarios ofertantes y demandantes del servicio.

1.1 *¿Cómo surge la idea?*

Personalmente, soy una persona aficionada a este tipo de vehículos, y he estado siguiendo en redes sociales a usuarios de autocaravanas y campers. He podido identificar varios problemas:

- El alto precio de los vehículos vendidos por fabricantes y camperizadores.
- El alto precio del alquiler de estos por parte de profesionales.
- La inversión que debe realizar el dueño del vehículo para adaptarlo a sus necesidades, que tardará un tiempo en amortizar.
- La pandemia ha forzado a la gente a buscar nuevas alternativas para viajar alejadas de las muchedumbres a las que estábamos acostumbrados.
- A su vez, la crisis generada por la pandemia ha dejado mermados los bolsillos de mucha gente, lo que hace que busquen otras formas de viajar menos convencionales y más económicas.

Por todo esto, decidí desarrollar una aplicación para ayudar a los usuarios a gestionar reservas de este tipo de vehículos.

1.2 *Antecedentes*

Weeknvan parte del proyecto final presentado para la asignatura de programación multimedia y dispositivos móviles. Esta aplicación utilizaba una base de datos alojada en Firebase, lo cual me obligó a investigar por mi cuenta, y no pude sacarle todo el potencial que podía. No era una aplicación funcional, simplemente era un proyecto para demostrar que podía investigar y utilizar los conocimientos adquiridos para desarrollar una aplicación para dispositivos Android.

Proyecto “Desarrollo de Aplicaciones Multiplataforma”

Título del Proyecto: Weeknvan



Solamente podía leer vehículos, crear y modificar usuarios, y crear reservas. Permitía ver una sola foto de los vehículos, y tenía bugs que podían solucionarse a futuro.

1.3 Idea inicial

La idea inicial consistía en realizar una aplicación para dispositivos con sistema operativo Android, utilizando como lenguaje de programación Java, y desarrollar una API para que la aplicación pudiera gestionar los datos. Pretendía gestionar la API y la base de datos mediante contenedores Docker. Todo esto se alojaría en AWS, o en algún otro proveedor.

1.4 Evolución de la idea

Tras tener la primera tutoría con Javier, llegamos a la conclusión de que lo mejor sería añadirle nuevas funcionalidades a la aplicación, como la posibilidad de subir vehículos con varias fotos, una mejor gestión de las reservas, añadidos estéticos, etc.

1.5 Investigación de la competencia

No existe mucha competencia en el sector del alquiler de autocaravanas, pero no por ello debemos subestimarla. Para ello, he analizado las principales aplicaciones de la competencia, para identificar sus puntos fuertes y sus puntos débiles.

Yescapa

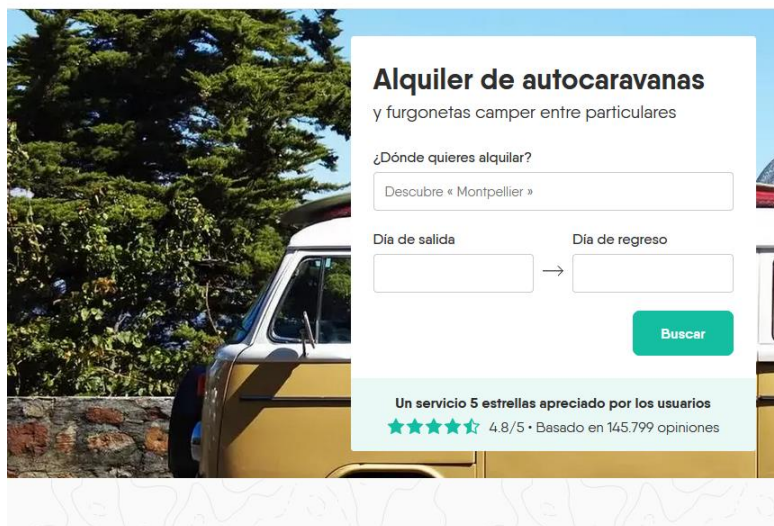


Ilustración 1. Portada – Web Yescapa.



Ilustración 2. Detalles – App Yescapa.

Yescapa es el principal competidor. Podemos ver que no pueden utilizar su servicio los usuarios profesionales. Nos permite la subida de un vehículo con varias fotos, detalles, etc.

Está disponible para Android, iOS y para navegadores web.

Booking

Aunque ahora mismo no sea competencia directa, si que lo es de manera indirecta. Booking es una plataforma referencia en el mercado de las reservas de hoteles. Se fue expandiendo a alojamientos gestionados por particulares, vehículos, vuelos, etc.

No es de extrañar que llegue el momento de que den el salto al mercado de las autocaravanas, por lo tanto, vamos a analizarlo.

Booking no es una plataforma amigable a la hora de anunciar tu alojamiento, por lo que tampoco lo sería a la hora de anunciar un vehículo. Permite subida de varias fotos, varias opciones de precio, precio variable según demanda, etc.

Está disponible para Android, iOS y para navegadores web.

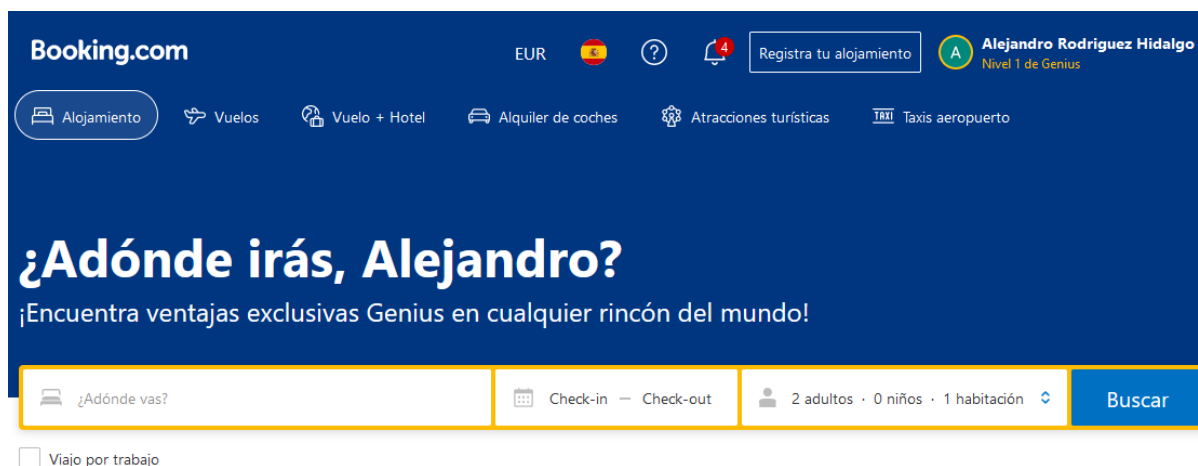
The image shows the Booking.com homepage. At the top, there's a navigation bar with the Booking.com logo, currency (EUR), a language selector (Spanish flag), a help icon, a notification bell with 4 alerts, a 'Registra tu alojamiento' button, and a user profile for 'Alejandro Rodriguez Hidalgo' with a Genius level 1 badge. Below this is a horizontal menu with icons and labels for 'Alojamiento', 'Vuelos', 'Vuelo + Hotel', 'Alquiler de coches', 'Atracciones turísticas', and 'Taxis aeropuerto'. The main heading is '¿Adónde irás, Alejandro?' followed by the subtext '¡Encuentra ventajas exclusivas Genius en cualquier rincón del mundo!'. Below this is a search bar with three input fields: '¿Adónde vas?' (with a house icon), 'Check-in — Check-out' (with a calendar icon), and '2 adultos · 0 niños · 1 habitación' (with a person icon and a dropdown arrow). To the right of these fields is a blue 'Buscar' button. At the bottom left of the search bar area, there is a checkbox labeled 'Viajo por trabajo'.

Ilustración 3. Portada – Web Booking

Utilizaremos lo aprendido en este análisis para reforzar los puntos que vemos débiles en la competencia, así como para tomar ideas sobre la disposición de los elementos, campos que tener en cuenta a la hora de guardar datos, etc.

2 Objetivos

Después de probar algunas de las aplicaciones similares en el mercado, pude ver los pros y los contras de cada una de ellas. Casi todas están orientadas a que los usuarios particulares alquilen sus vehículos. Sin embargo, otras plataformas de funcionamiento similar, como puede ser Booking o Idealista, permite tanto usuarios particulares como profesionales.

Nuestro modelo de negocio sería a comisión con el total del precio acordado entre ambas partes, por lo que nos interesa tener a cuantos más usuarios ofertantes mejor.

Los objetivos, desde un principio, estaban bien definidos:

- Debe de ser una aplicación que funcione con Firebase, para en un futuro, poder escalarla.
- Debe de tener una interfaz gráfica agradable e intuitiva.
- Sus funciones tienen que estar colocadas donde el usuario espera que estén. Un usuario confundido abandonará la aplicación para no volver, y es un lujo que no nos podemos permitir.

3 Tecnologías involucradas

3.1 *Android Studio*

Para la creación esta aplicación hemos utilizado el entorno de desarrollo Android Studio. Este IDE firmado por JetBrains nos permite el desarrollo para dispositivos Android de todo tipo, desde smartphones a radios de coche, pasando por relojes y tablets.

La aplicación ha sido desarrollada utilizando Java como lenguaje de programación. A su vez, las vistas de la aplicación se encuentran en formato XML.

La versión utilizada del IDE es Android Studio Chipmunk 2021.2.1.

3.2 *Firebase*

Firebase es un BaaS (Backend as a Service) de Google, que nos proporciona una API y un Framework para facilitarnos la vida a la hora de elaborar aplicaciones para multitud de plataformas.

Las herramientas de Firebase que hemos utilizado son:

- **Firebase Authenticator:** Es una herramienta utilizada para la autenticación de usuarios mediante correo, teléfono, redes sociales, etc. Nos permite multitud de funcionalidades útiles, como pueden ser la recuperación de contraseña, o el inicio de sesión con una cuenta de Google.
- **Firebase Realtime Database:** Es una herramienta utilizada para guardar bases de datos en formato JSON. Tiene la particularidad de que son bases de datos actualizadas a tiempo real, con lo que nos provee de una serie de métodos y listeners para sacar todo el provecho de esta función.
- **Firebase Storage:** Es un blob de almacenamiento en la nube, muy útil para subir archivos como documentos, imágenes, etc. Es una herramienta muy potente si se usa de manera conjunta a una base de datos, ya que podemos guardar los punteros a nuestras imágenes para recuperarlas más tarde asociadas al objeto.

3.3 Java

Como hemos dicho anteriormente, el lenguaje de programación elegido ha sido Java. Aunque Android Studio nos permite programar las aplicaciones también utilizando el lenguaje Kotlin, he preferido utilizar Java por la soltura adquirida en estos años.

3.4 JSON

Firebase Realtime Database utiliza JSON. No tendremos que tocar código si no queremos, pero nos permite la opción de importar un archivo JSON. En mi caso, he preferido formatear un archivo JSON, por la rapidez con la que se forma, y luego importarlo a la base de datos para que refleje datos significativos de cara a los juegos de pruebas.

3.5 Git + GitHub

El control de versiones se ha realizado mediante Git versión 2.30.0. A su vez, este repositorio local se ha ido sincronizando con la nube de GitHub mediante el gestor de control de versiones integrado en Android Studio.

Para cambios estéticos en el repositorio se ha utilizado el gestor integrado en la propia web de GitHub.

3.6 Dia - Diagram Editor

Para la creación del diagrama entidad-relación de base de datos que luego transformaremos a JSON, hemos utilizado el editor Dia. Es libre, y aunque antiguo, funciona correctamente.

3.7 Visual Paradigm 16.3

Para la creación de los diagramas de casos de usos hemos utilizado Visual Paradigm en su versión de prueba.

3.8 Adobe Photoshop 2021

Para la creación de algunas imágenes, prototipos y demás elementos gráficos hemos utilizado una versión de dudosa procedencia de Adobe Photoshop, la suite gráfica por excelencia.

3.9 Icon Kitchen

Para la creación del icono de la aplicación se ha utilizado la página web Icon Kitchen, la cual facilita mucho el proceso de generar iconos para Android. Te permite descargar un archivo zip con todo el contenido de los iconos en su interior, listo para reemplazar en la ubicación correcta de tu proyecto Android.

También permite generar iconos para iOS, AndroidTV, Play Store, etc.

Esta herramienta está desarrollada por un programador de Firebase (Google).

4 Proceso de desarrollo

4.1. Análisis

Incluir los diagramas que se hayan desarrollado durante el para el desarrollo de la aplicación (diagramas de casos de uso, diagrama de clases, diagrama entidad relación, etc).

Se puede incluir toda la documentación que se estime oportuna.

4.1.1 Base de datos

El diseño de la base de datos se hizo mediante un diagrama Entidad/Relación. Más tarde lo transformaremos a JSON, ya que son las normas que nos impone Firebase Realtime Database.

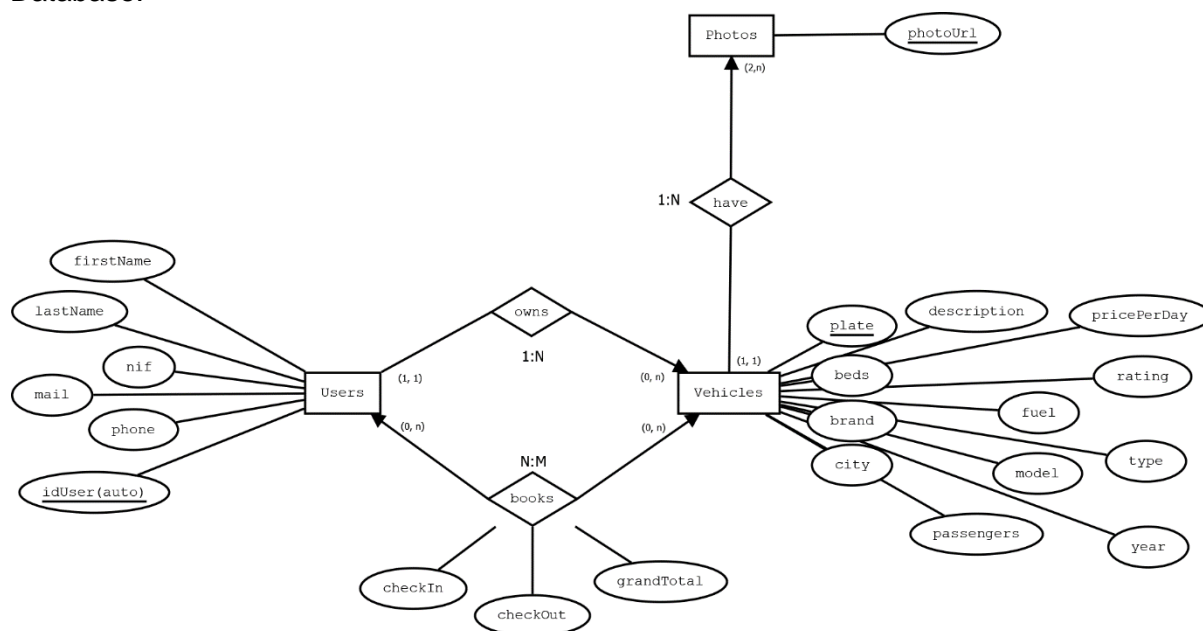


Ilustración 4. Diagrama Entidad/Relación.

Los usuarios guardarán su nombre, apellidos, nif, mail, teléfono y un idUser autogenerado gracias a Firebase Authenticator.

Los vehículos guardarán matrícula, camas, marca, ciudad, descripción, combustible, modelo, pasajeros, precio diario, valoración, tipo y año. También guardarán un array de fotos, las cuales contendrán a su vez la URL de la misma. Un usuario puede poseer un vehículo o no, por lo tanto al realizar la propagación de claves, el vehículo guardará el atributo owner.

La relación intermedia books N:M nos da una tabla intermedia. Esta tabla guardará las claves de usuarios y de vehículos, así como una fecha de entrada y otra de salida. Esta combinación nos permitirá múltiples reservas de un mismo usuario a un mismo vehículo. También guardará el atributo calculado grandTotal, para poder obtenerlo directamente sin tener que consultar el precio diario del vehículo.

4.1.2 Diagramas de casos de uso

En nuestro caso, solo disponemos de un usuario, que dispondrá de dos vistas.

- La de usuario sin vehículo, buscando alquilar uno.
- La de usuario propietario, ofertando su vehículo.

Así, he realizado los siguientes diagramas, con las opciones de cada vista.

En el caso de usuarios, se decidió prescindir de la opción borrar, ya que es algo trivial que no aporta nada académicamente hablando, y restaba tiempo utilizado para otras mejoras más interesantes.

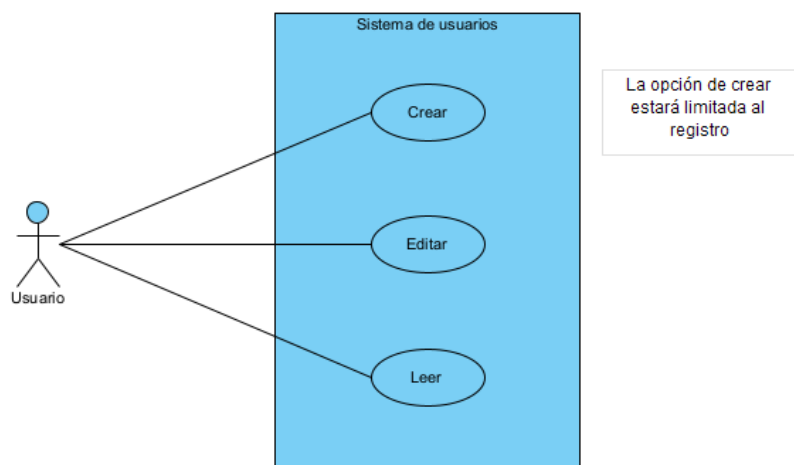


Ilustración 5. Diagrama casos de uso – Usuario.

En el caso del sistema de reservas, se ha prescindido de la opción de modificar, ya que es mucho más rápido borrarla y hacerla de nuevo. De nuevo, se podría haber abordado, pero consumiría un tiempo muy valioso en otras partes de la aplicación.

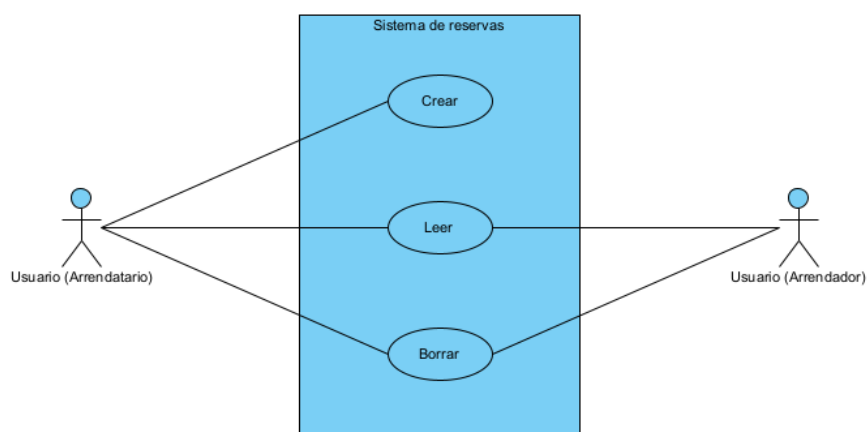


Ilustración 6. Diagrama casos de uso – Reservas.

Por último, en el caso del sistema de vehículos, se ha decidido que los usuarios puedan crear y leer vehículos. Es aquí donde viene una de las partes más complejas de la aplicación, ya que se permite la subida múltiple de imágenes, así como la visualización de estas mediante un carrusel.

Se ha forzado la subida de más de dos fotos, para que el usuario disponga de múltiples fotos (Ejemplo: Interior y exterior). De esta manera, también podremos apreciar el carrusel en nuestras instancias de vehículos.

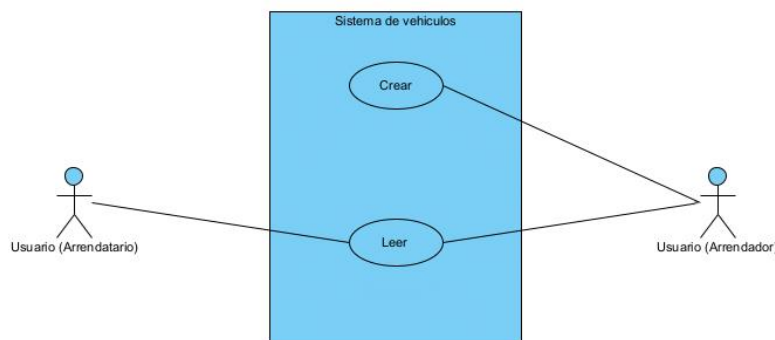


Ilustración 7. Diagrama casos de uso – Vehículos.

4.1.3 Diagrama de clases

El diagrama de clases generado por Visual Paradigm es demasiado extenso como para documentarlo. No se adjuntará dicho diagrama de clases por considerarlo poco útil. Sin embargo, la aplicación estructura dichas clases en distintos paquetes:

- Paquete listadapters, con 6 adaptadores para distintos RecyclerView que maneja la aplicación:
 - ListAdapter para el RV principal.
 - ListAdapterBooking para los RV con fechas.
 - ListAdapterItemTop para el RV horizontal de arriba.
 - ListAdapterItemBottom para el RV horizontal de abajo.
 - ListAdapterUser para el RV de vehículos de usuarios.

- SliderAdapter para el carrusel de la pantalla detalles.
- Paquete login:
 - ActivityMain.
 - LoginFragment.
 - SignUpFragment.
- Paquete main, con el activity BottomNavigationActivity, y los siguientes fragmentos:
 - BookingFragment, que realizará la reserva.
 - DetailsFragment, que mostrará los detalles del vehículo.
 - MyBookingFragment, que mostrará nuestras reservas entrantes y salientes.
 - SearchFragment, que mostrará la pestaña de búsqueda.
 - UserFragment, que mostrará la pestaña de usuario con todas sus opciones.
 - VehiclesFragment, que mostrará el RV principal con los vehículos que se le indiquen.
 - VehiclesUploadFragment, que nos permitirá subir un vehículo a Firebase.
- Paquete objects, que guardará los objetos utilizados por la aplicación:
 - Booking, reserva.
 - ItemBanner, banner de los RV horizontales.
 - User, usuario.
 - Vehicle, vehículo.

4.2. Desarrollo (diagrama de secuencias,...)

4.2.1 GitHub

Para explicar el proceso de desarrollo, vamos a basarnos en los commits producidos en GitHub. No es una documentación totalmente fiel, ya que tengo algunos commits muy extensos, del orden de +400 líneas.

Proyecto “Desarrollo de Aplicaciones Multiplataforma”

Título del Proyecto: Weeknvan



JUNTA DE EXTREMADURA
Consejería de Educación y Empleo

- Día 1: Se realiza el commit inicial y se retoca el proyecto. Se corrigen errores y se crea el README.md.

Commits on May 15, 2022

- Update del README.md
arodriguezh09 committed 29 days ago
- Creamos readme en rama master ...
arodriguezh09 committed 29 days ago
- Arreglado el gradle y el viewbinding del login
arodriguezh09 committed 29 days ago
- Commit inicial del proyecto
arodriguezh09 committed 29 days ago

- Día 2 y 3: Cambios sin importancia en el día 2. Se cambia completamente la estética y funcionalidad del login/signup.

Commits on May 19, 2022

- Merge remote-tracking branch 'origin/master'
arodriguezh09 committed 25 days ago
- Cambios estéticos login/registro + recursos.
arodriguezh09 committed 25 days ago

Commits on May 16, 2022

- Update header ...
arodriguezh09 committed 29 days ago
- Update README.md
arodriguezh09 committed 29 days ago

Proyecto “Desarrollo de Aplicaciones Multiplataforma”

Título del Proyecto: Weeknvan



- Días 4 y 5: Cambios en signup + cambios en distintos cv listadapters.

fragmentos,

Commits on Jun 5, 2022

- Merge remote-tracking branch 'origin/master'
arodriguez09 committed 9 days ago
- Fragmento de búsqueda, cardviews, rv, y listAdapters.
arodriguez09 committed 9 days ago

Commits on May 20, 2022

- Update README.md ...
arodriguez09 committed 25 days ago
- Search item carousel + teoría para marcar y bloquear fechas en el Mat...
arodriguez09 committed 25 days ago
- Signup user push
arodriguez09 committed 25 days ago

- Días 6 y 7: Se cambian CV, adaptan RV, se añade funcionalidad a los mismos en reservas, muchos pequeños cambios.

Commits on Jun 11, 2022

- Añadido rv user, arreglo scroll
arodriguez09 committed 2 days ago
- RV de reservas (in/out) funcional
arodriguez09 committed 3 days ago

Commits on Jun 10, 2022

- Creado RV de reservas complejo
arodriguez09 committed 4 days ago

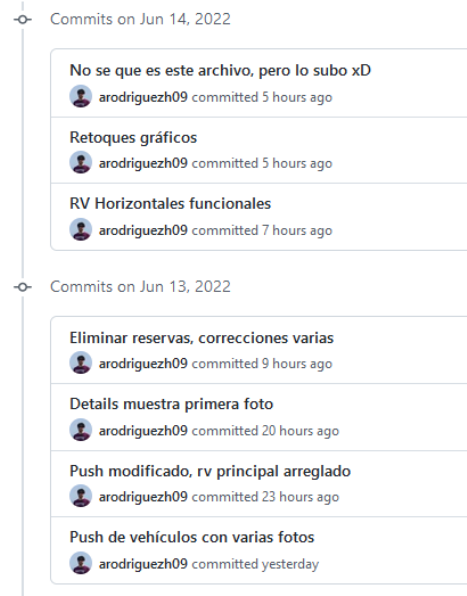
Commits on Jun 9, 2022

- Creado CV de reservas
arodriguez09 committed 4 days ago

Commits on Jun 6, 2022

- Añadido fragmento de creación/edición de vehículos
arodriguez09 committed 8 days ago

- Días 8, y 9: Se termina de tocar los detalles finales, se corrigen push a Firebase, se añade push de varias fotos, se recuperan en un carrusel, se realizan retoques a nivel gráfico.



4.3. Pruebas realizadas

Para la realización de pruebas se ha optado por seguir una tabla con los posibles fallos que puede dar la aplicación, e intentar hacer que falle. Ensayo – error.

Prueba	Superada	Observaciones
Crear usuario – campo blanco	Correcto	Ninguna
Crear reserva pisando otra	Correcto	Ninguna
Crear vehículo – campo blanco	Correcto	Ninguna
Crear vehículo – sin foto	Correcto	Ninguna
Crear vehículo – menos de dos fotos	Correcto	Ninguna
Crear usuario – formato correo	Correcto	Ninguna
Crear usuario – coincidir contraseñas	Correcto	Ninguna
Borrar reserva	Correcto	Ninguna

5 Proceso de Despliegue

Nuestra aplicación tiene la build de la APK subida a GitHub. Bastaría con descargarla en un dispositivo con sistema operativo Android y dar permisos de ejecución a aplicaciones de Orígenes Desconocidos. Según la capa de personalización de nuestro teléfono, se encontrará en un sitio o en otro. La ruta a seguir suele ser:

Ajustes > Seguridad > Orígenes desconocidos / Activar

Nuestra aplicación utiliza la API 21 de Android. Es decir, será compatible con todos aquellos dispositivos con al menos Android 5.0 (Lollipop). Esto nos dará una cobertura sobre un 98.6% de los dispositivos con sistema operativo Android en uso.

Nuestra base de datos está alojada en Firebase. No puedo facilitar los datos de acceso a la misma, ya que no se puede acceder con un dominio de @iesalbarregas.es, y he tenido que utilizar mi correo electrónico personal. Aún así, la aplicación está preparada para funcionar desde el momento en que se instale, no debería haber problema ninguno.

6 Propuesta de mejora o trabajos futuros

Esta aplicación permite muchísimas mejoras, tanto sobre ella misma como trabajando con la base de datos en Firebase.

Mejoras sobre la aplicación existente:

- Podrían pintarse las fechas ocupadas en el calendario.
- Podría incluirse un tema para modo oscuro, muy utilizado actualmente.
- Podría realizarse un CRUD completo por cada uno de los objetos, borrando también la parte que incluyera almacenamiento de imágenes en la nube.
- Sería interesante poder mandar mensajes entre los distintos usuarios, y crear un perfil público para estos, con sus diferentes vehículos.
- Al estar semi implementado un sistema de puntuación de vehículos, podría completarse. Actualmente se abandonó la idea porque tendría que recuperar la información de la BBDD y calcularla en el dispositivo, por lo que no sería una manera segura de hacerlo. Actualmente se añade con un random.
- Mejoras obvias en el diseño.

Mejoras sobre Firebase:

- Creación de otras aplicaciones que hicieran uso de la misma base de datos, siendo compatibles entre distintas plataformas.
- Creación de reglas para que sólo puedan acceder a la BBDD las aplicaciones firmadas.

7 Bibliografía

Documentación Firebase - <https://firebase.google.com/docs/database/>

Documentación Java - <https://docs.oracle.com/javase/7/docs/api/>

Guías de diseño de Material Design - <https://material.io/design/>

Tutoriales Foxandroid - <https://www.youtube.com/c/Foxandroid01/>

Stackoverflow en general - <https://stackoverflow.com/>