



Daniel Campos Sánchez

Proyecto Final - Grado Superior Desarrollo de Aplicaciones Multiplataforma

Nombre de la empresa: MyShopPal -V 1.0
Espartinas, Sevilla
2021

Índice

1.- Visión general	3
1.1.- Objetivos	
1.2.- Especificaciones	
2.- Tecnologías a utilizar	4
3.- Infraestructura	5
3.1.- Elementos Hardware	
3.2.- Elementos Software	
3.3.- Recursos Web	
4.- Documentación	6
4.1.- Sistema Operativo: Android	
4.2.- Lenguaje: Kotlin	
4.3.- Requerimientos funcionales	
4.3.1.- Descripción básica	
4.3.2.- Seguridad	
5.- Pantallas de uso	11
6.- Arquitectura del proyecto	29
7.- Google Firebase	33
7.1.- Firebase Authentication	
7.2.- Cloud Firestore	
7.3.- Storage	



1.- Visión general

Por la situación que estamos viviendo, la compra online es uno de los comercios que más se ha disparado en el último año. Tras las distintas cuarentenas, nos hemos visto obligados a pedir de manera online. Este hecho nos remite a que la mensajería, también ha incrementado este último año, por eso, para este proyecto vamos a crear una aplicación móvil para Android donde cualquier usuario de la plataforma puede subir sus propios productos y venderlos.

Se ha descubierto la facilidad de compra que podemos tener con un solo click, y que nos lo traigan a casa.

El proyecto viene a cubrir esa falta de tecnología de las tiendas, como puede ser un comercio local de nuestra localidad, las cuales no están preparadas para vender online. Y no solo eso, sino que también cualquier particular tiene espacio en la plataforma para vender sus productos, ya sean nuevos o de segunda mano.

Ya no solo es el comercio, sino que también hemos descubierto que se le puede dar una segunda vida a objetos que tenemos en casa y que solo ocupan espacio.

A día de hoy la mayor demanda se encuentra en las compras online, por lo tanto, esto generaría un gran número de ventas para estas pequeñas empresas.

1.1.- Objetivos

La aplicación que se va a desarrollar tendrá las siguientes **funcionalidades**:

1. Pantalla Splash al iniciar la aplicación (con el logo de la empresa).
2. Registro y login de usuario a través de correo electrónico y contraseña.
3. Recuperación de contraseña (si se ha olvidado).
4. Rellenar formulario del usuario con aspectos generales (sexo, telefono, etc...)
5. Consulta de diferentes productos

1.2.- Especificaciones

Hay dos fases en el proyecto, que son:

La principal es el **diseño y la arquitectura del proyecto**, donde se relacionan todas las pantallas, los componentes definidos, clases modelo, estilos, colores, palabras y frases. Se

ha separado todo en carpetas, para un mayor acceso de todos los componentes del proyecto. Se puede ver en el punto **6 - Arquitectura del proyecto**.

La segunda y última, es la **implementación**. Se han utilizado componentes como puede ser el servicio de Firebase de Google, donde nos da acceso a herramientas como una base de datos en la nube, autenticación de usuarios, colecciones de objetos donde guardaremos los datos principales. Todo esto seguido de la documentación que proporciona Google para hacer su correcta implementación. Se podrá ver más detalladamente.

2.- Tecnologías a utilizar

I. Sistema Operativo base

La práctica se va a realizar en un ordenador equipado con Windows 10.

II. IDE

Para realizar aplicaciones de Android, usaremos Android Studio en su última versión. Este será el IDE principal para toda la programación de las pantallas. El lenguaje seleccionado para esta aplicación es Kotlin.

III. Base de datos.

La Base de datos que usaremos para la aplicación será Firebase (base de datos remota)

IV. Ilustraciones

Las ilustraciones, como el logo de la empresa, se realizarán con aplicaciones como Illustrator o Photoshop de Adobe.

V. Otros puntos

Todo depende de cómo vaya el desarrollo de la aplicación, en rasgos principales, creo que lo tengo todo, aplicación desarrollada en Android Studio usando el lenguaje Kotlin, usando como base de datos Firebase. No se si necesitaré otras tecnologías durante el desarrollo, pero creo que estas cubren la parte principal del proyecto.

3.- Infraestructura

Para este proyecto se han utilizado diferentes recursos, tanto infraestructura propia como herramientas de cloud.

3.1.- Elementos Hardware

Componentes	Características técnicas	Tareas
Portatil	- MSI GP60 2PE Leopard - 16Gb de RAM - S.O: Windows 10	Diseño de la aplicación. Redacción de la documentación.
Smartphone Principal	Pixel 3a API 30	Terminal para pruebas, usado como simulador para probar las funcionalidades de la aplicación.
Smartphone secundario	Pixel 4 API R	Terminal para pruebas, usado como simulador para probar las funcionalidades de la aplicación. (En caso de fallo del smartphone principal).

3.2.- Elementos Software

Herramienta	Tareas
Android Studio 3.6	Entorno de desarrollo integrado oficial para la plataforma Android
Adobe Photoshop CC 2019	Realización del logo para la empresa. El logo es usado como icono de la aplicación y aparece durante la ejecución del proyecto.
Mozilla Firefox 89.0	Navegador utilizado para la búsqueda de información, uso de Firebase, búsqueda de vídeos relacionados con componentes de la aplicación

3.3.- Elementos Web

Herramienta	Tareas
Google Firebase	Plataforma para el desarrollo de apps para Android que facilita tareas como la creación o autenticación, base de datos en la nube (Cloud), almacenamiento de datos, configuración remota, etc...

4.- Documentación

4.1.- Sistema Operativo: Android



android

Android es un **sistema operativo móvil** diseñado para dispositivos móviles con pantalla táctil como teléfonos inteligentes o tablets, pero que también lo encontramos en otros dispositivos como relojes inteligentes, televisores o incluso en los sistemas multimedia de algunos modelos de coches. Un sistema operativo desarrollado por Google y basado en el Kernel de Linux y otros softwares de código abierto y que se ha convertido en el principal responsable de la popularización de muchos

dispositivos inteligentes por el hecho de facilitar el uso de una gran cantidad de aplicaciones de forma sencilla.

Google Play es la tienda o plataforma online desarrollada por el gigante buscados y donde podemos encontrar todo tipo de apps para Android. Su app está instalada por defecto en todos los dispositivos Android para que los usuarios tengan un fácil acceso a todo el catálogo de apps de música, juegos, noticias, clima, educación, compras, salud, utilidades, mapas, deporte, bancarias, etc.

Para poder usar la tienda de aplicaciones, es necesario asociar una cuenta de Gmail, pero lo cierto es que, debido a la gran popularidad del sistema operativo de Google, a día de hoy

podemos encontrar otras tiendas de aplicaciones alternativas a Google Play donde encontraremos otras muchas herramientas y aplicaciones.

De cualquier forma, las aplicaciones para Android están comprimidas en formato **APK**, lo que facilita su instalación desde cualquier explorador de archivos en la mayoría de los dispositivos. Eso sí, es posible que antes de nada tengamos que acceder a los ajustes del sistema para activar la opción que nos permite instalar aplicaciones de fuentes desconocidas. Y es que Android viene configurado para que sea desde Google Play desde donde se realice la instalación de todo tipo de aplicaciones.

4.2.- Sistema Operativo: Android



Kotlin es un lenguaje de programación estático de código abierto que admite la programación funcional y orientada a objetos. Proporciona una sintaxis y conceptos similares a los de otros lenguajes, como C#, Java y Scala, entre muchos otros. No pretende ser único, sino que se inspira en décadas de desarrollo del lenguaje. Cuenta con variantes que se orientan a la JVM (Kotlin/JVM), JavaScript (Kotlin/JS) y el código nativo (Kotlin/Native).

Kotlin es administrado por [Kotlin Foundation](#), un grupo creado por [JetBrains](#) y Google, que se ocupa de continuar el desarrollo del lenguaje. Google admite oficialmente Kotlin para el desarrollo de Android, lo cual significa que la documentación y las herramientas de Android están diseñadas para ser compatibles con Kotlin.

Algunas API de Android, como [Android KTX](#), son específicas de Kotlin, pero la mayoría están escritas en Java y se pueden llamar desde Java o Kotlin. La interoperabilidad de Kotlin con Java es fundamental para su crecimiento. Eso quiere decir que puedes llamar al código Java desde Kotlin, y viceversa, y de esa manera aprovechar todas tus bibliotecas de Java existentes. La popularidad de Kotlin da como resultado una experiencia de desarrollo más agradable en Android, aunque continuamos desarrollando el marco de trabajo de Android con Kotlin y Java en la mente.

4.3.- Requerimientos funcionales

4.3.1.- Descripción básica

La aplicación dispondrá de varias funcionalidades que se describen a continuación:

- **Login:** Para poder entrar en la aplicación hace falta tener un correo electrónico registrado. En la pantalla se encuentran dos campos, email y contraseña, si no coincide con alguno de los campos con lo que hay en la base de datos, no se puede acceder a la pantalla principal. Si se ha olvidado la contraseña, se puede recuperar pulsando sobre el texto '**¿Has olvidado tu contraseña?**', nos mostrará una nueva pantalla, donde debemos poner el email al que quiere que se le envíe el correo para restablecer la contraseña del usuario.
- **Registro:** Si no tenemos acceso a la aplicación, es porque debemos crear una cuenta. Al registrarnos, nos aparece un formulario para rellenar diferentes campos. Por ejemplo, nombre, apellido, correo electrónico y contraseña. Una vez aceptemos los términos y condiciones de la aplicación, estaremos registrados. Ahora tendremos acceso desde la pantalla de login con la nueva cuenta.
- **Perfil de usuario:** Si es la primera vez que entramos con el usuario en la aplicación, nos mostrará una pantalla de **Perfil de usuario**, donde rellenaremos más datos del usuario, como telefono, sexo, etc... Si no es la primera vez que entra en la aplicación, esta pantalla no aparecerá, solo cuando accedemos por primera vez.
- **Sesión abierta:** Una vez hemos abierto sesión en la aplicación con tu usuario, no se cerrará la sesión. Esto se debe a que si vemos aplicaciones más conocidas, como Instagram, Wallapop, Amazon, nunca cierran la sesión del usuario para no pedir las credenciales todas las veces que entre en la aplicación.
- **Comprar productos:** La aplicación nos enseña una pantalla con todos los productos que han subido los usuarios a la aplicación. Pueden aparecer productos repetidos, ya que por ejemplo, si buscamos un producto en Amazon posiblemente encontremos el mismo producto vendido por diferentes vendedores, pues lo mismo ocurre aquí. Cada producto tiene una ID que está asociado a un usuario.

- **Tus pedidos:** Todos los productos que hayamos pedido, nos aparecerán en **Tus pedidos**. Pulsando sobre el pedido, podemos observar diferentes detalles necesarios como son los productos que se han pedido, el estado del pedido, dirección a la que se envía, el subtotal del pedido...
- **Direcciones:** Un mismo usuario puede tener varias direcciones asociadas a su cuenta. Principalmente se le dan dos opciones, su casa o oficina, además de tener la opción de otro. A la hora de realizar un pedido tiene que seleccionar una dirección, y si no hay ninguna dirección asociada, debe crearse una.
- **Subir tus productos:** Cada usuario de la aplicación puede ser su propia “**tienda**”. Siguiendo un poco la dinámica de ventas como aplicaciones estilo Wallapop, cada usuario puede vender todos sus productos, y obviamente el mismo no puede comprar sus productos aunque aparezca en la página principal.
- **Ver tus productos vendidos:** Tenemos la opción de ver los productos que hemos subido, tanto la opción de subir como de borrar. Y una pantalla con los productos que se han vendido, para así saber qué productos hemos comprado nosotros a otros compradores.
- **Detalles de tus pedidos:** En **Pedidos** podemos ver todos los pedidos detallados, por ejemplo, si se han comprado dos productos, aparecerá el porte final y la descripción de los dos productos, asociados a un **ID de pedido**, una **fecha**, **estado del pedido**. Los **estados** del pedido van cambiando según pasa el tiempo, es decir, desde que se realizó el pedido hasta que se finaliza esa compra, deben pasar dos horas, mientras podemos ver el cambio en el estado del producto.
- **Cerrar sesión:** Como la sesión se queda guardada aunque cerremos la aplicación, si queremos abrir otra sesión, en **Ajustes**, tenemos la opción de **Logout**, y tras cerrar sesión, nos mandará al **Login principal** donde podemos registrarnos con otra cuenta, pero no con un correo que ya está en los datos de la aplicación.

4.3.2.- Seguridad

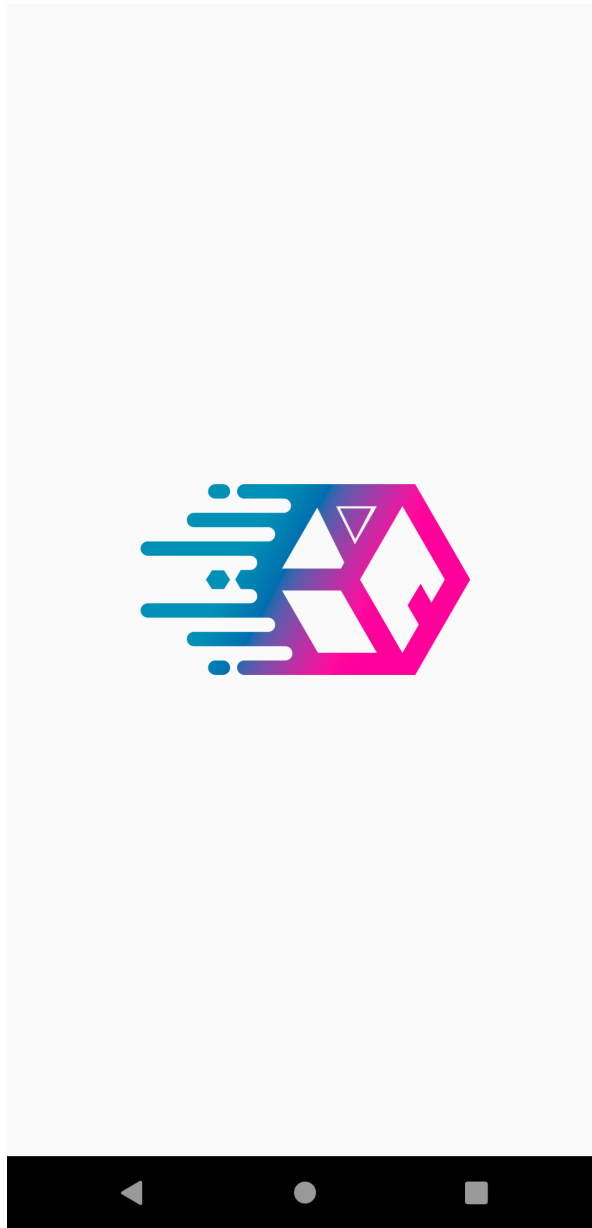
En cualquier proyecto de la actualidad se debe prestar mucha atención en la seguridad, para ello se tiene que disponer de acceso como mínimo con usuario y contraseña. En este caso, se habilitará autenticación mediante correo electrónico y contraseña.

Se utilizará el sistema de autenticación que nos ofrece Google llamado Firebase Authentication. Una vez que el usuario inicie sesión por primera vez, se creará una cuenta de usuario nueva y se la vinculará con las credenciales, es decir, el nombre y la contraseña del usuario con los que el usuario inició sesión.

Esta cuenta nueva se almacenará como parte del proyecto y se puede usar para identificar a un usuario en la app.

5.- Pantallas de uso

Aquí mostraremos las pantallas que se utilizarán durante la aplicación.

Nombre de la pantalla	Imagen
activity_splash.xml	

activity_login.xml



LOGIN


Email

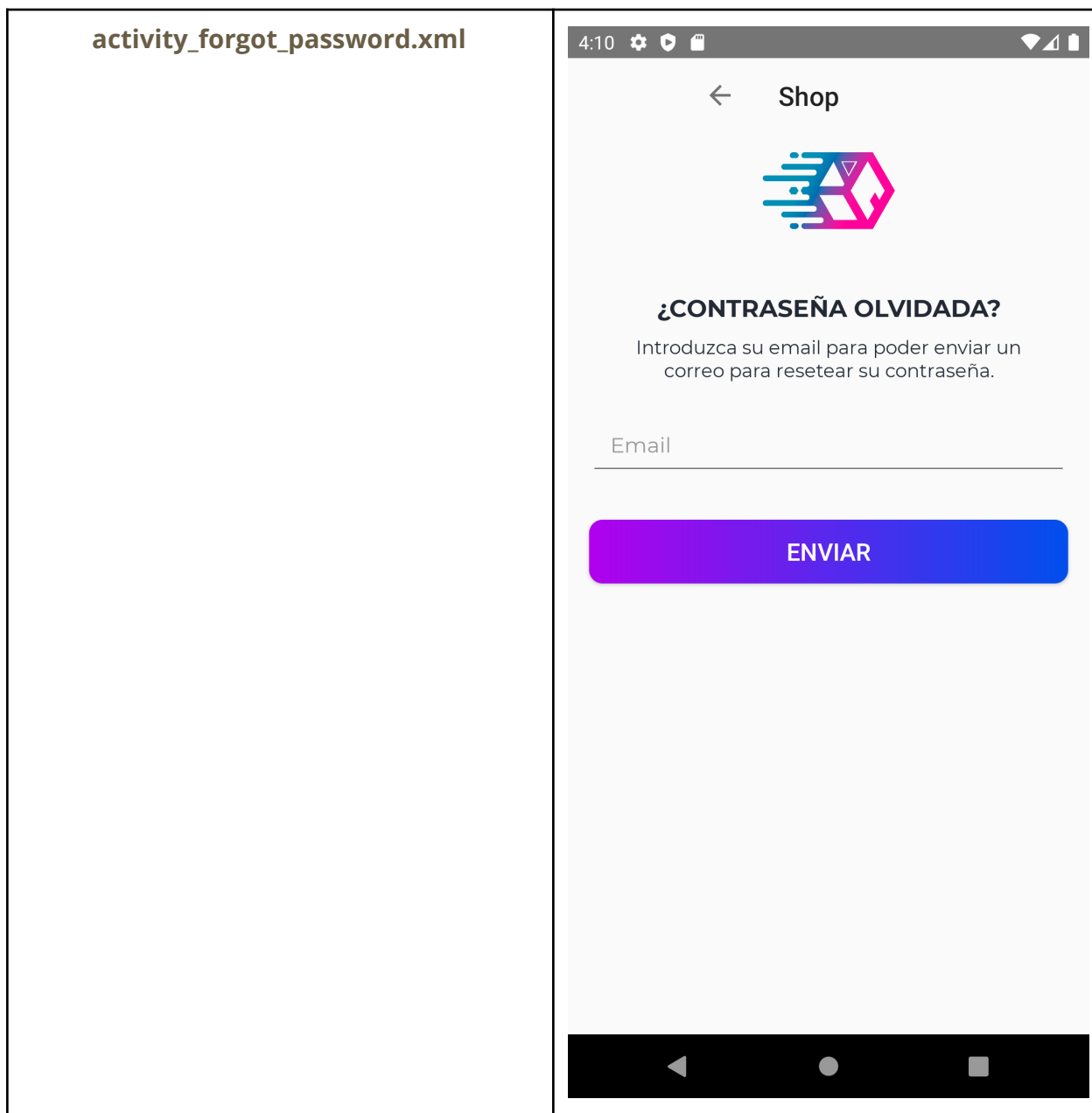
Contraseña

[¿Has olvidado tu contraseña?](#)

ACCEDER

[¿No tienes cuenta? Registrar](#)


activity_register.xml	
	<div><div></div><div>Crear una cuenta</div></div> <div><div>Nombre</div><div>Apellidos</div><div>Email</div><div>Contraseña</div><div>Confirmar contraseña</div></div> <div><div><input type="checkbox"/></div><div>Acepto los terminos y condiciones</div></div> <div><div>REGISTRAR</div></div> <div><div>¿Tienes cuenta?</div><div>Login</div></div>



activity_user_profile.xml

4:10

PERFIL



Nombre

Apellidos

Email

Telefono movil

Hombre



Mujer


Guardar

dashboard_activity.xml

8:15

Dashboard





Ruedas Universal 52mm
\$29







Tabla 8" Yuri Relics
\$57





Ruedas Spitfire 53mm
\$59.80





Ejes Venture V-LIGHT
\$76.9







Dashboard









activity_product_details.xml

8:15



DETALLES DEL PRODUCTO



Tabla 8" Yuri Relics

\$57

Descripción del producto

Marca Almost

Cantidad de existencias


6

AÑADIR AL CARRITO

fragment_products.xml

8:15

Productos



Ruedas Universal 52mm
\$29







Tabla 8" Yuri Relics
\$57




Ruedas Spitfire 53mm
\$59.80




Ejes Venture V-LIGHT
\$76.9




Ejes Independent Stage 11
\$73.2




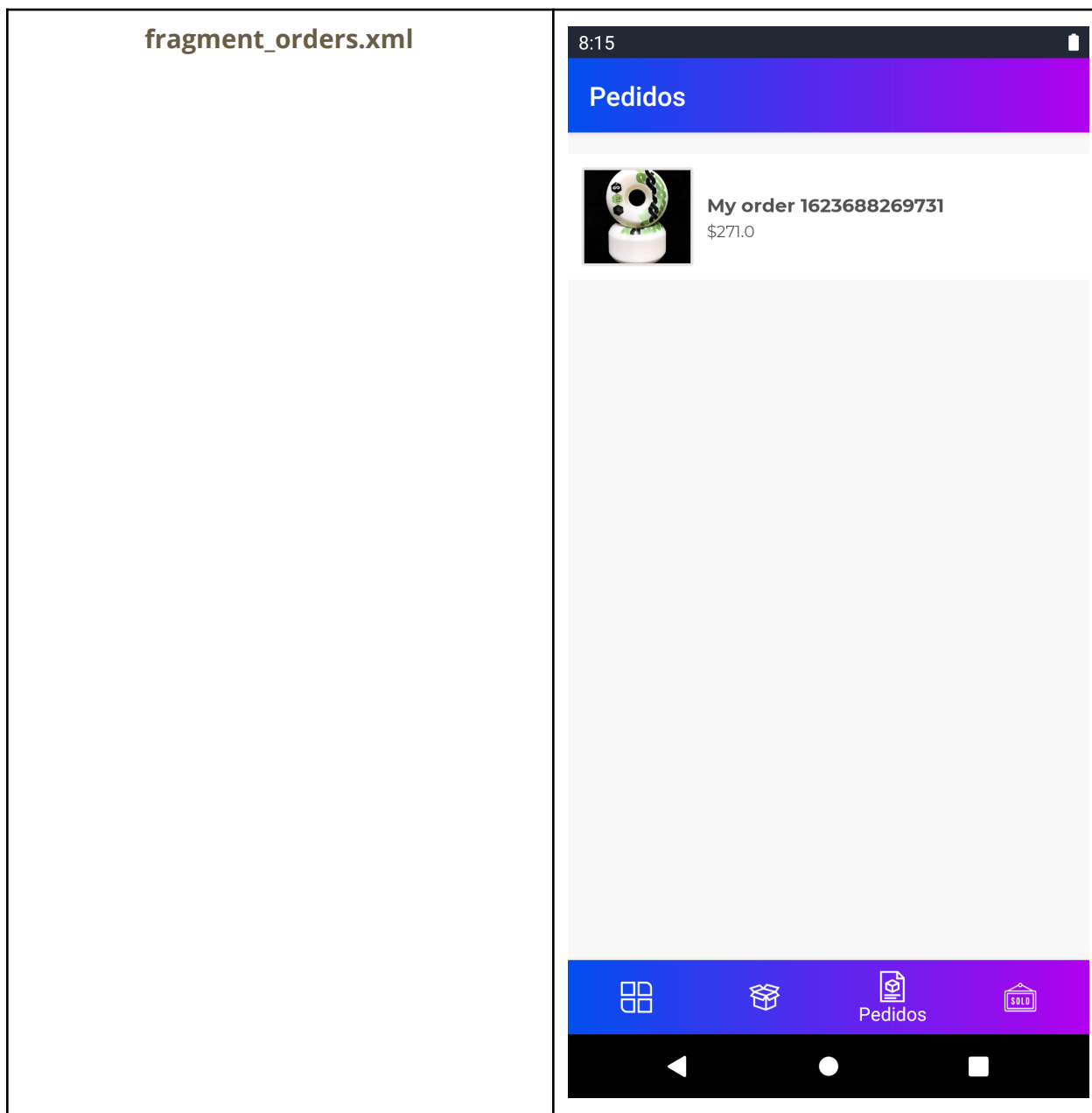
Ejes Imagine Silver



Productos







activity_my_order_details.xml

8:15



LOS DETALLES DE MI PEDIDO

Detalles del pedido

ID de pedido My order 1623688269731

Fecha del pedido 14 Jun 2021 16:31

Estado del pedido **Entregado**

Productos



Ruedas Universal 52mm

\$29

9

Dirección de envío

Home

Daniel Campos

Calle Perú, 41807

n 43

654654654

Artículos Recibidos

Subtotal 261.0

Gastos de envío 10.0

Importe total 271.0

Modo de pago Pago contra reembolso

fragment_sold_products.xml

8:15

Productos vendidos



Tabla 8" Yuri Relics

\$57



Ruedas Universal 52mm

\$29



Ruedas Spitfire 53mm

\$59.80



Productos vendidos

activity_sold_product_details.xml

8:15



DETALLES DEL PRODUCTO VENDIDO

Detalles del pedido

ID de pedido My order 1623698805315

Fecha del pedido 14 Jun 2021 19:26

Productos



Tabla 8" Yuri Relics

\$57

2

Dirección de envío

Home

Daniel Campos

Calle Perú, 41807

n 43

654654654

Artículos Recibidos

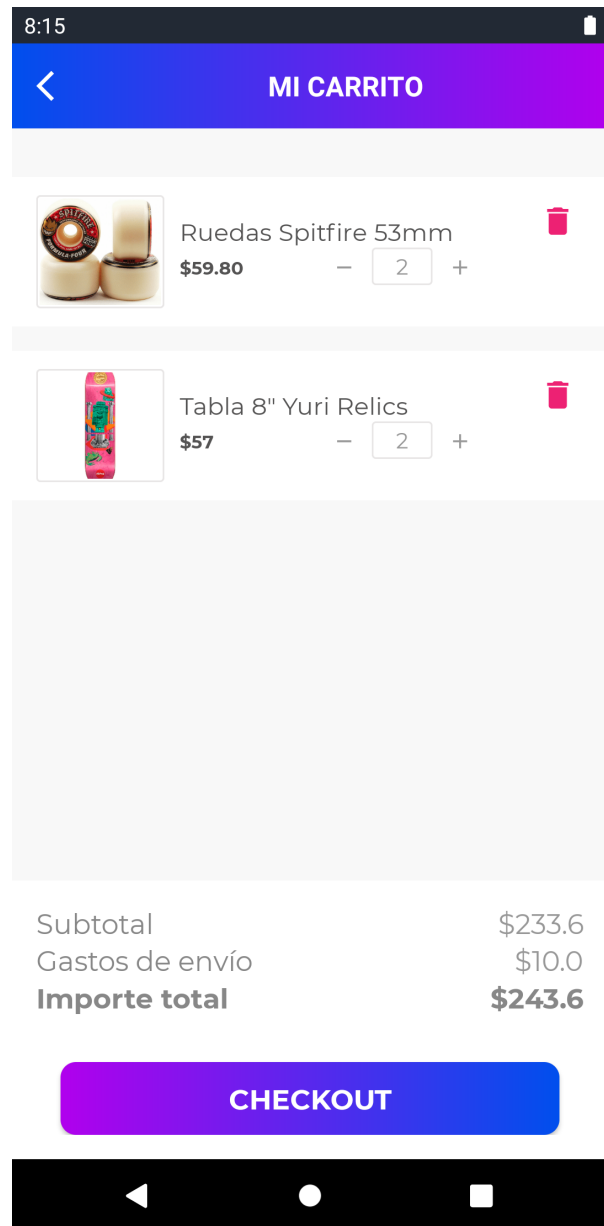
Subtotal 233.6

Gastos de envío 10.0

Importe total 243.6



activity_card_list.xml



activity_checkout.xml

8:15

< CHECKOUT

Productos





Tabla 8" Yuri Relics
\$57

2



Ruedas Spitfire 53mm
\$59.80

2

Dirección seleccionada

Home
Daniel Campos
Calle Perú, 41807
n 43


654654654

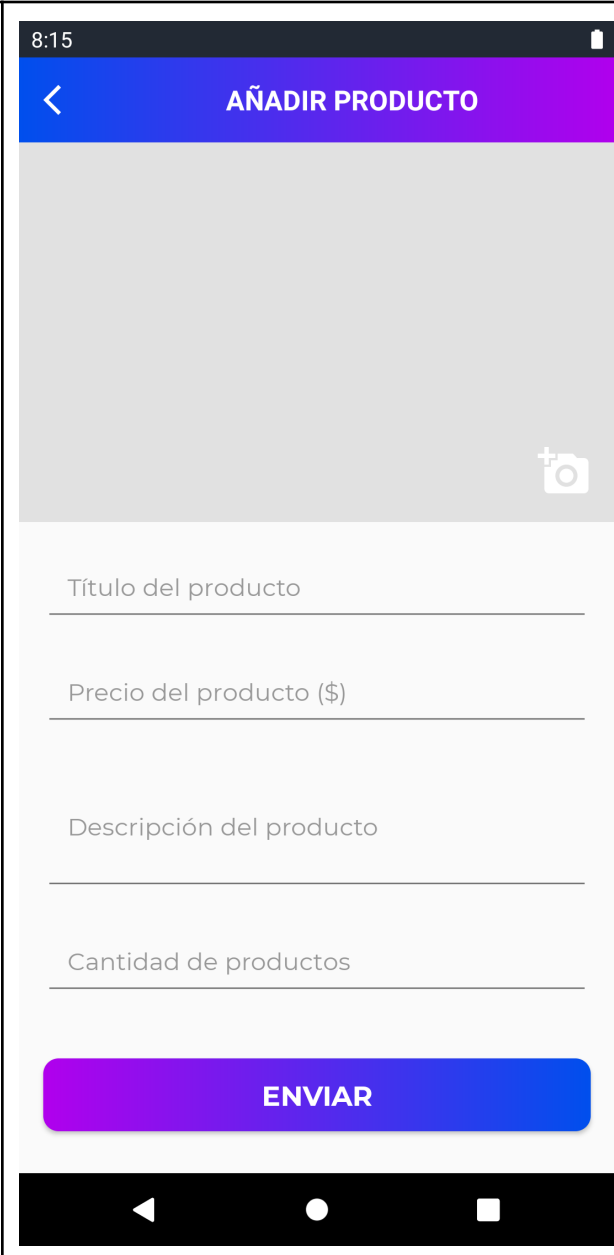
Artículos Recibidos

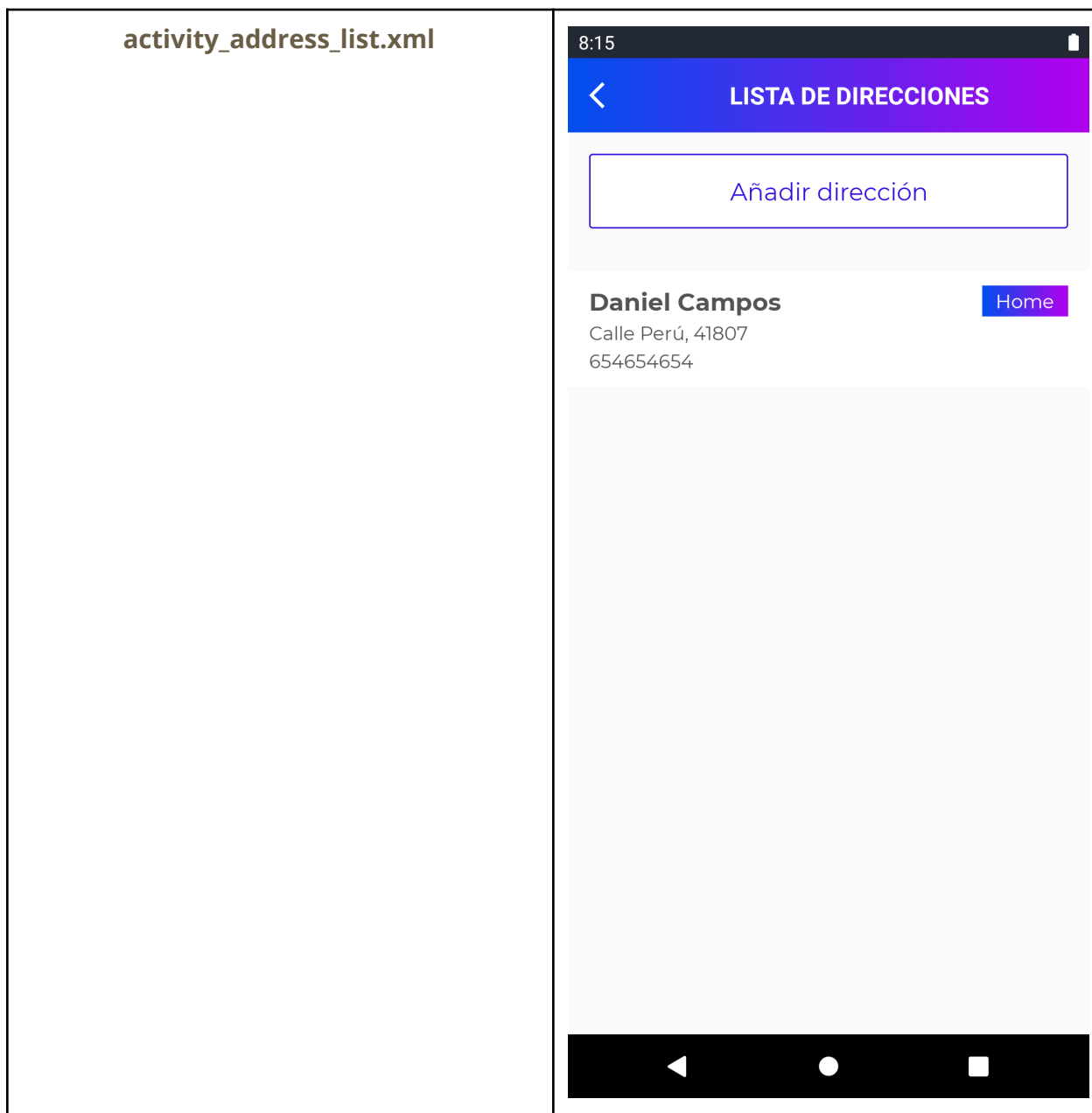
Subtotal \$233.6

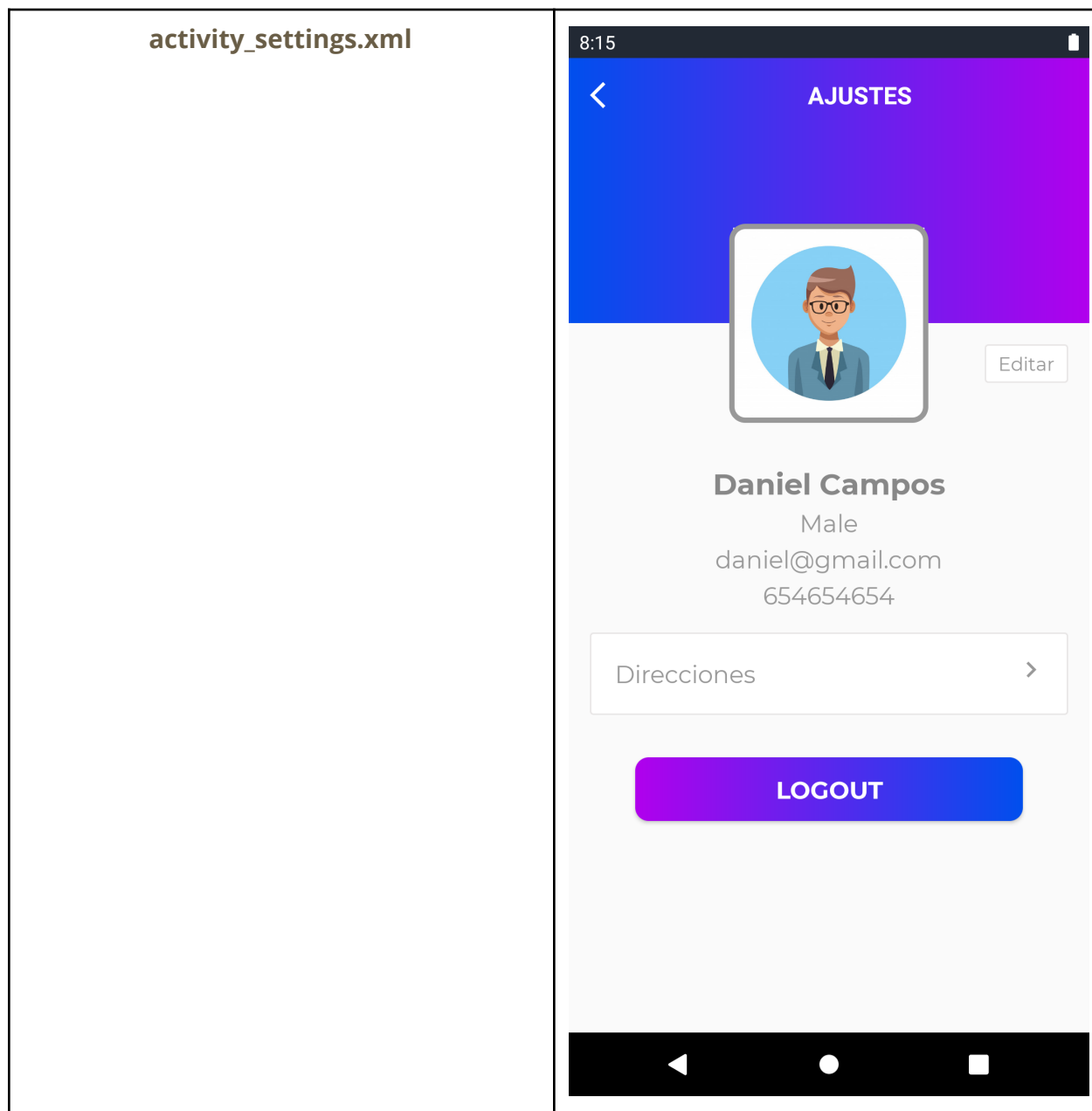
Modo de pago **Pago contra reembolso**

COLOCAR ORDEN



<p>activity_add_product.xml</p>	 <p>The image shows a mobile application interface for adding a product. At the top, there is a status bar with the time 8:15 and a battery icon. Below it is a header bar with a blue-to-purple gradient, containing a back arrow on the left and the text "AÑADIR PRODUCTO" on the right. The main content area has a light gray background. It features a large gray rectangle for a product image, with a camera icon in the bottom right corner. Below the image are four text input fields with labels: "Título del producto", "Precio del producto (\$)", "Descripción del producto", and "Cantidad de productos". At the bottom of the form is a blue button with the text "ENVIAR". The very bottom of the screen shows the Android navigation bar with back, home, and recents icons.</p>
----------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------





activity_add_edit_address.xml

8:15

< AÑADIR DIRECCIÓN

Nombre completo

Número de teléfono

Dirección

Código postal

Nota adicional

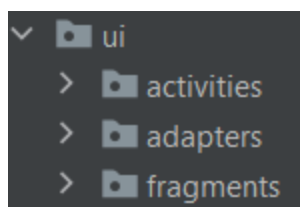
CASA OFICINA OTRO

ENVIAR

6.- Arquitectura del proyecto

Para la arquitectura del proyecto, se ha separado todo por carpetas, para tener un rápido acceso a todos los ficheros necesarios. En el árbol del proyecto, se puede observar

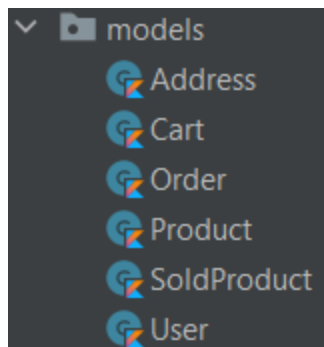
6.1.- UI



Aquí encontramos tres grupos muy importantes en la aplicación:

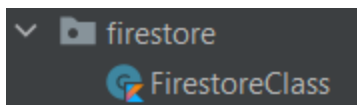
- **Activities:** Todos los activities de la aplicación (que no son más que el código fuente para hacer funcionar el layout que hemos diseñado, como botones, textView, etc...), con todo el código necesario para hacer comprobaciones y conseguir una buena ejecución entre todas las llamadas de los activities y el buen funcionamiento de los botones creados.
- **Adapters:** Todas las clases de este paquete son utilizados para recoger los datos de la base de datos, y mostrarlos en los RecyclerView. Los RecyclerView son los listados de objetos, que previamente cargaremos con un diseño (card_view) y muestra todos los objetos de la lista.
- **Fragments:** Las clases fragment se utilizan para definir las funciones y miembros que utilizaremos en todos los fragmentos. Es decir, establecer en los card_view los datos de los objetos recogidos de la base de datos.

6.2.- Models (Modelos)



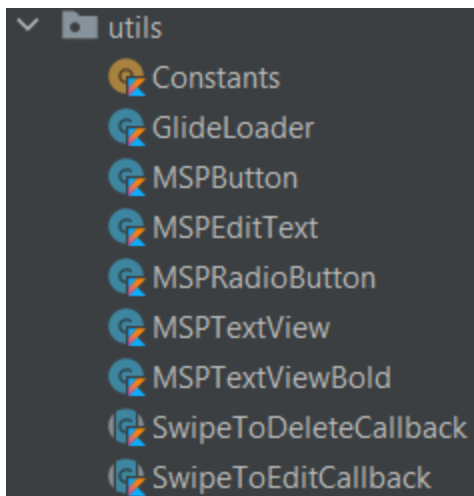
Son las clases de los objetos que vamos a ir creando durante la ejecución del programa, donde se establecen los campos que vamos a utilizar.

6.3.- Firestore



Aparecen todas las funciones relacionadas con la conexión entre aplicación y base de datos. Obtener todos los objetos, actualizar, borrar, nuevos registros de usuarios, login, etc...

6.4.- Utils (Utiles)



En este paquete se encuentran las clases que vamos a utilizar durante la aplicación, desde constantes, como aplicar fuentes descargadas a diferentes botones, textView, ect... Definimos un poquito más detalladamente todo lo que encontramos aquí.

6.4.1.- Constants

Encontramos todas las constantes que aparecen durante toda la aplicación, desde las constantes usadas para la base de datos de firebase, hasta la selección de imágenes que vamos a subir. También utilizamos las constantes para actualizar los datos de las colecciones de la base de datos.

6.4.2.- Glide Loader

Un objeto personalizado para crear una función común para Glide que se puede utilizar en toda la aplicación. Se utiliza para cargar las imágenes que cargamos en la aplicación.

6.4.3.- MSP... Button, EditText, RadioButton, TextView, TextViewBold

Aplicamos el tipo de letra a todos los componentes.

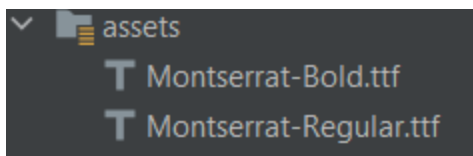
6.4.4.- SwipeToDeleteCallback

Clase abstracta que utilizaremos para la función de borrado.

6.4.5.- SwipeToEditCallback

Clase abstracta que utilizaremos para la función de edición.

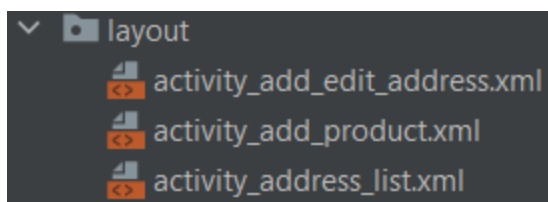
6.5.- Assets



Fuentes para aplicar en el texto de todo lo que diseñamos, en mi caso he escogido, Monserrat Regular y Monserrat Bold.

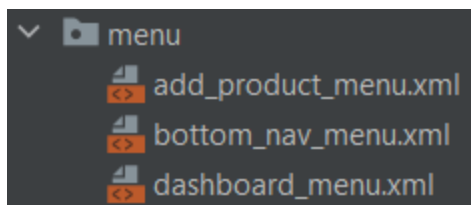
6.6.- Res

6.6.1.- Layout



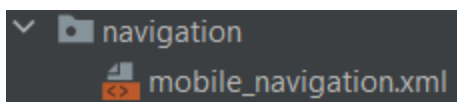
Paquete donde encontramos todos los layout que hemos diseñado para obtener los activities a nuestro gusto. Estos layouts están asociados a un activity, que son los que encontramos en el paquete ui.activities. En el layout solo encontramos el diseño, en los activities, las funciones de cada uno.

6.6.2.- Menú



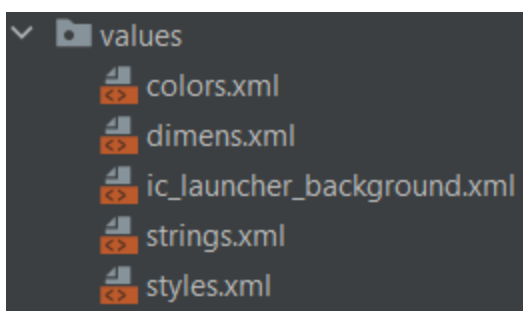
Encontramos los botones que añadiremos en los menús de los dashboard, por ejemplo, añadir un producto, ajustes, cesta de la compra, etc... Todos los iconos necesarios para lo descrito anteriormente.

6.6.3.- Navegation



Fichero con los fragments, asociados a cada fragmento que hemos creado anteriormente. Son cuatro, Dashboard, productos, pedidos y productos vendidos. Para ser más exactos, es la navegación de los cuatro botones que encontraremos en la pantalla principal de la tienda.

6.6.4.- Values



6.6.4.1.- Colors

Todos los colores están predefinidos en este fichero. Son llamados normalmente en las pantallas de layout a la hora de definir qué colores se utilizarán, por ejemplo, en los botones, el color del texto, etc.

6.6.4.2.- Dimens

Todas las dimensiones que serán utilizadas para el tamaño de cualquier tipo de componente del diseño, está aquí escrito. El porqué es muy sencillo, muchos de los valores se van a repetir durante todo el diseño de la aplicación, por ejemplo el tamaño de botones, simplemente con llamar a la dimensión de los botones que tenemos aquí escrita. Así no creamos desproporciones en los mismos elementos que estamos utilizando en el diseño.

6.6.4.3.- Strings

Absolutamente todas las frases, palabras de la aplicación se encuentran aquí definidas. Ocurre lo mismo que con las dimensiones, si necesitamos cambiar un texto en concreto que se repite durante toda la aplicación, simplemente tenemos que buscar el string en concreto y cambiar el valor para que se vea reflejado en todas las llamadas de ese string, de esta manera, no tenemos que ir buscando línea

por línea de código, donde tenemos que cambiar el mensaje. Se pueden crear diferentes ficheros strings, para así, añadir más idiomas, y que nuestra aplicación pueda ser más universal y tenga mayor capacidad de alcance, no solo para un país en concreto, sino para todo el mundo, por ejemplo, teniendo ficheros string en español (atacamos directamente a clientes de España y Latinoamérica) y otro en inglés.

6.6.4.4.- Styles

Definimos los estilos, los estilos que se han definido aquí son principalmente el NoActionBar. Son usados en el Manifest del proyecto, y se aplican los estilos que hemos definido aquí, o los que están ya predefinidos en la aplicación.

7.- Google Firebase



Firebase de Google es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Está disponible para distintas plataformas (iOS, Android y web), con lo que es más rápido trabajar en el desarrollo.

Aunque fue creada en 2011 pasó a ser parte de Google en 2014, comenzando como una base de datos en tiempo real. Sin embargo, se añadieron más y más funciones que, en parte, permitieron agrupar los SDK de productos de Google con distintos fines, facilitando su uso.

Su función esencial es hacer más sencilla la creación de tanto aplicaciones webs como móviles y su desarrollo, procurando que el trabajo sea más rápido, pero sin renunciar a la calidad requerida.

Sus herramientas son variadas y de fácil uso, considerando que su agrupación simplifica las tareas de gestión a una **misma plataforma**. Las finalidades de las mismas se pueden dividir en **cuatro grupos: desarrollo, crecimiento, monetización y análisis**. Es especialmente interesante para que los desarrolladores no necesiten dedicarle tanto tiempo al backend, tanto en cuestiones de desarrollo como de mantenimiento.

7.1.- Firebase Authentication

La identificación de los usuarios de una app es necesaria en la mayoría de los casos si estos quieren acceder a todas sus características.

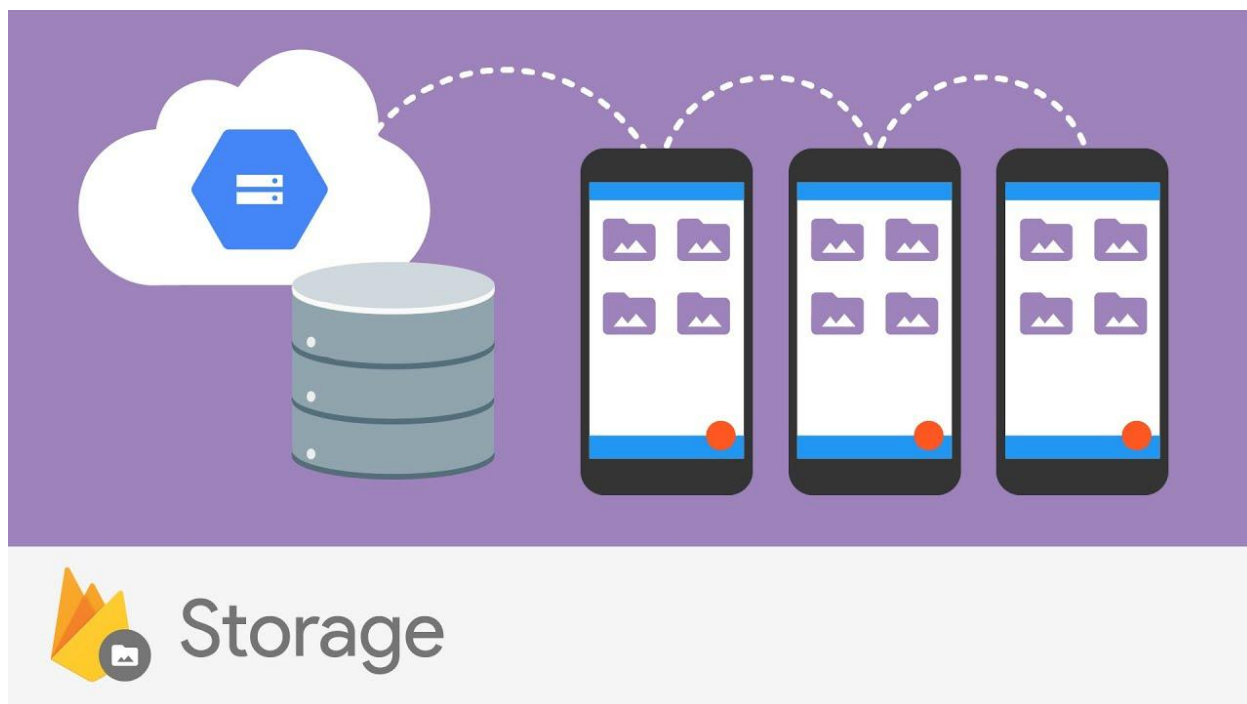
Firebase ofrece un sistema de **autenticación** que permite tanto el **registro** propiamente dicho (mediante email y contraseña) como el acceso utilizando perfiles de otras plataformas externas (por ejemplo, de Facebook, Google o Twitter), **una alternativa muy cómoda** para usuarios reacios a completar el proceso.

Así, este tipo de tareas se ven simplificadas, considerando también que desde aquí se gestionan los accesos y se consigue una mayor seguridad y protección de los datos. Se debe mencionar que Firebase puede guardar en la nube los datos de inicio de sesión con total seguridad, evitando que una persona tenga que identificarse cada vez que abra la aplicación.



Firebase Authentication

7.2.- Cloud Firestore



Firebase cuenta con un sistema de almacenamiento, donde los desarrolladores pueden guardar los **ficheros de sus aplicaciones** (y vinculándolos con referencias a un árbol de ficheros para mejorar el rendimiento de la app) y sincronizarlos. Al igual que la mayoría de herramientas de Firebase, es personalizable mediante determinadas reglas.

Este almacenamiento es de gran ayuda para **tratar archivos de los usuarios** (por ejemplo, fotografías que hayan subido), que se pueden servir de forma más rápida y fácil. También hace la descarga de referencias a ficheros más segura.

7.3.- Storage



Firebase Storage no es más que un servicio en el que se puede almacenar contenido generado por usuarios tales como fotos y vídeos, definición que ellos mismos dan de su producto. El servicio te provee una API simple con la que puedes subir archivos así como descargarlos. También te provee la posibilidad de usar Google Cloud Storage desde el servidor para acceder a estos espacios de almacenamiento, denominados buckets.