

PICKACLOTH

MEMORIA DEL PROYECTO DE FIN DE GRADO

Frida Abella Fernández

Jaime Aranda Congil

Guillermo Gabard Delgado

1. RESUMEN

Este proyecto final que hemos creado combina una aplicación Android y una aplicación web centradas en torno al objetivo de poder subir información y visualizarla en tiempo real desde una base de datos en la nube, en nuestro caso Firebase. Creemos que es un tipo de aplicación bastante polivalente, cuya estructura y funcionalidad podría ser útil para el almacenamiento, modificación y visualización digital de datos, algo que hoy en día se hace fundamental y facilita enormemente la productividad de cualquier tarea. De modo que podría limitarse a ser una aplicación de uso personal o bien interno de una empresa, a modo de “biblioteca”, archivo o base de datos visualizada de cualquier elemento. Hemos decidido, sin embargo, abordar el uso de estos datos fundamentalmente desde una perspectiva de negocio, en la cual no sólo haya un administrador sino también una serie de clientes de cuyas acciones dependa este flujo de información.

La idea surge de observar la amplia presencia y cada vez mayor importancia que el e-commerce tiene y la necesidad imperante, para negocios de cualquier tipo, de buscar una salida digital a la gestión y comercialización de su stock de producto. Temáticamente, abordamos este negocio digital desde la perspectiva de un negocio del sector de la ropa, dada la experiencia personal, interés y éxito que éste tiene en la actualidad. Sin embargo sería perfectamente aplicable a distintos tipos de negocio.

Aunque somos conscientes de la amplia competencia que existe para nuestro negocio y modelo de aplicación, la idea era poder acercarnos y conocer desde dentro el funcionamiento de este tipo de aplicaciones, trabajar con una base de datos en la nube y poder entender las necesidades fundamentales que derivan de este tipo de actividades online, fundamentalmente desde dos tipos de perspectivas: usuario y administrador, en función de las cuales hay distintas funcionalidades u objetivos. Se cubren así las dos partes fundamentales de un negocio.

Por otra parte, hemos querido hacer una doble aplicación. Si bien originalmente pensamos que la aplicación móvil sería específicamente utilizada por empleados, para que pudieran fotografiar el producto desde el dispositivo móvil y moverse por el almacén, finalmente resulta ser también ampliamente útil para el perfil de

cliente. Ambas aplicaciones están pensadas para cumplir originalmente con las dos funciones.

2. OBJETIVOS

Como decíamos, los dos roles o usuarios fundamentales tienen unas necesidades específicas diferentes que se intentan cubrir con nuestras dos aplicaciones.

- Administrador (empresa/vendedor):
 - o Dar de alta el stock de producto que va a ofrecer
 - o Visualizar y gestionar sus pedidos (aceptar, notificar como procesados, etc.)
- Usuario (cliente).
 - o Visualizar los productos disponibles ofertados
 - o Añadirlos a un carrito/borrar el carrito
 - o Realizar una orden o simulación de compra.

3. MÓDULOS FORMATIVOS APLICADOS EN EL TFG

Ha sido una de nuestros grandes objetivos el poder aplicar el mayor número de conocimientos adquiridos a lo largo del curso (incluidas nuestras prácticas en empresa o FCT) para poder retornos a ponerlos en práctica en conjunto y además en un proyecto que, de perfeccionarse, podría tener continuidad.

1. Programación

En el desarrollo tanto de la página web como de la aplicación se ha dado funcionalidad dinámica a los elementos y al tratamiento de la información a partir de nociones programación orientada a objetos. En concreto, se ha utilizado Java para implementar la aplicación de Android.

2. Base de datos

Pese a que el curso ha incidido especialmente en las bases de datos SQL nosotros hemos optado por una base de datos no relacional o NoSQL, dada su menor

rigidez que los modelos entidad-relación (las estructuras y relación entre los datos no serán tan rígidas), su escalabilidad y carácter descentralizado, y que no necesita de grandes máquinas por lo que no necesita mayor presupuesto. Nos hemos decantado por Firebase porque además de cumplir con estos requisitos, está alojada en la nube, es gratuita y nos permite trabajar con imágenes gracias a su Storage. Gran parte de su conocimiento, por tanto, hemos tenido que adquirirlo por nuestra cuenta. Sin embargo las nociones generales de interacción con una base de datos sí han sido importantes. Dentro de Firebase hemos usado la RealtimeDatabase y el Storage, que guarda las imágenes (agrupadas en carpetas en función de nuestro código) y se asocian a la Database a partir de una Uri.

3. Entornos de desarrollo

Este módulo nos ha servido en los conocimientos que aporta de cara al trabajo en equipo, gracias al uso de git, y de cara al “debug” para corregir errores de funcionamiento en las aplicaciones.

4. Lenguajes de Marcas

El diseño y desarrollo del front-end ha utilizado en buena medida los conocimientos adquiridos en el módulo, como los lenguajes de marcas y etiquetas HTML, CSS (versiones como SCSS). En Android, los “layouts” o diseños de las *activities* son ficheros XML. Por otra parte, Firebase trabaja con JSON, por lo que es necesario.

5. Diseño de interfaces

La funcionalidad del diseño de la web se ha hecho con Typsecript, una evolución de Javascript con grandes semejanzas a éste.

Por otra parte, hemos intentado tener en cuenta alguno de los conceptos de UX/UI, en lo que se refiere a diseño y usabilidad, para poder hacer las interfaces más agradables e intuitivas, más allá de su mera funcionalidad.

6. Desarrollo de aplicaciones móvil y multiplataforma

Probablemente la que más peso haya tenido en este proyecto, dado que toda una aplicación entera ha sido creada en Android con Java, para lo cual los conocimientos previos han sido fundamentales.

7. Formación en centros de trabajo

Las prácticas en empresa o FCT han llevado a algunos componentes del equipo a formarse y trabajar de manera práctica en proyectos que utilizan Angular como framework, por lo que decidimos aprovechar la ocasión para profundizar en esos conocimientos y aplicarlos en la creación de la aplicación web.

8. Empresa e iniciativa emprendedora

Aunque nuestro producto cuenta con una amplia competencia y no se ha realizado un estudio de mercado exhaustivo, sí se han tenido en cuenta factores como la situación y tendencias del mercado, la viabilidad económica, etc.

4. HERRAMIENTAS/LENGUAJES UTILIZADOS

4.1. Android Studio + Java

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de apps para Android y está basado en IntelliJ IDEA, que ha sido nuestro editor de código para la aplicación móvil. Hemos utilizado **Java** en lugar de Kotlin para programar, dado que era el lenguaje que conocíamos.

Cada módulo de app contiene las siguientes carpetas:

- manifests: contiene el archivo AndroidManifest.xml.
- java: contiene los archivos de código fuente Java, incluido el código de prueba de JUnit.
- res: contiene todos los recursos sin código, como diseños XML, strings de IU e imágenes de mapa de bits.

De entre las ventajas de Android Studio, han sido fundamentales para nosotros:

- Su sistema de compilado flexible basado en Gradle. Gracias a ello hemos podido implementar numerosas dependencias fundamentales para muchas de las funcionalidades de nuestra aplicación (conexión con la base de datos de Firebase y transacciones con ésta, herramientas accesorias para tratar las imágenes como Picasso o un “cropper”, etc.).
- Su emulador, que nos ha permitido ir observando el funcionamiento de la aplicación en ejecución.
- La compatibilidad integrada con servicios en la nube de Google como Firebase.

4.2. Firebase

Hemos elegido que nuestras aplicaciones estén conectadas y sincronicen sus datos con Realtime Database, la base de datos original de Firebase, una solución eficiente para apps para dispositivos móviles que necesitan estados sincronizados

entre los clientes en tiempo real. Algunas de sus ventajas que hemos encontrado con respecto a otros tipos de bases de datos son las siguientes:

- SDK centrados en el cliente. Esto significa que no hay necesidad de servidores que implementar ni mantener.
- Actualizaciones/sincronización con cada cliente conectado en tiempo real.
- Gratuita (en nuestro nivel de uso).
- Se sincronizan automáticamente y en tiempo real los datos.
- Para un proyecto de este tamaño no es necesaria una gran complejidad de datos conectados y dependientes entre sí como en una SQL.

Por otra parte, y con respecto a la otra base de datos de Firebase (Cloud Storage, más nueva), elegimos esta porque:

- Sincroniza datos con consultas básicas (nuestra app no tiene transacciones o búsquedas avanzadas).
- Son pocos GB de datos que cambian constantemente.
- Los datos se estructuran como un árbol JSON simple, en lugar de documentos organizados en colecciones. Nuestros datos son simples y fáciles de organizar. Las transacciones son atómicas en un subárbol de datos específico
- Tiene una disponibilidad/garantía de tiempo estimada en el 99.999% (extremadamente alta).
- Tiene soporte sin conexión para clientes Android.

Por otra parte, hemos escogido Cloud Storage para almacenar nuestros archivos de imágenes asociados a los productos que suben los clientes de las aplicaciones. Estos archivos se almacenan en un bucket de GoogleCloudStorage y son accesibles a través de Firebase y Google Cloud. Trabajar con Storage implica y permite:

- Integrar los SDK de firebase para Cloud Storage
- Crear una referencia o ruta al archivo (de tipo Uri)
- Subir o descargar en tipos nativos en la memoria o disco

- Proteger los archivos al estar integrada con Firebase Authentication (que usamos en nuestro proyecto).
- Gran escalabilidad y operaciones robustas de carga/descarga independientemente de la calidad de red.

Como hemos mencionado, utilizamos FirebaseAuth para la identificación de los usuarios, lo cual nos ha permitido que la app los guarde en la nube de forma segura y ejecute transacciones en función de esa identificación y autenticidad. Admite la autenticación mediante contraseñas, números de teléfono, proveedores de identidad federada populares, como Google, Facebook y Twitter, etc. Nos hemos limitado a usar contraseñas y números.

4.3. Angular + Typescript + Visual Studio Code

Para la realización de la aplicación web se ha utilizado el IDE VisualStudioCode como editor de código y entorno de desarrollo, Angular como framework y Typescript como lenguaje fundamental para los componentes de este framework, junto a HTML y SCSS. Se ha elegido este framework por las siguientes ventajas:

- La gran popularidad y demanda que tiene hoy en día para el desarrollo web
 - Porque ofrece más funcionalidades de serie que una biblioteca
 - Porque usa Typescript, que es un lenguaje de gran consistencia (la sintaxis y manera de hacer las cosas es la misma siempre, añadiendo coherencia)
 - Por ser una SPA (Single Web Application) y su dinamismo
 - Porque sus componentes permiten reutilización amplia y ágil
-
- La gran popularidad y demanda que tiene hoy en día para el desarrollo web
 - Por ser una SPA (Single Web Application)
 - Porque ofrece más funcionalidades de serie que una biblioteca
 - Porque usa Typescript, que es un lenguaje de gran consistencia (la sintaxis y manera de hacer las cosas es la misma siempre, añadiendo coherencia)

4.3. GitHub

Es una plataforma de desarrollo colaborativo que sirve para alojar el código de los proyectos o aplicaciones, de tal manera que diferentes usuarios puedan acceder a ellas, descargarlas o colaborar con su desarrollo a través de ramificaciones que parten de un código en común sin alterarlo, por lo que se facilita el trabajo en común sin errores y un resultado homogéneo.

5. FASES DEL PROYECTO

5.1. La idea

La formulación de la idea llegó desde el momento en que tuvimos que plantear un anteproyecto y pensar en alguna aplicación que resultase atractiva a nuestros intereses y a los de mucha población, y por otra parte de gran demanda o popularidad en el mercado: aplicaciones móviles y web para e-commerce. Dada la versatilidad del e-commerce, nuestras aplicaciones podrían servir para diferentes contextos o sectores, por lo que no dudamos en que era polivalente y apenas barajamos otras ideas.

5.2. El perfil de usuario o cliente

Creemos que es una aplicación transversal a diferentes tipos o naturalezas de negocio y por tanto a diferentes tipos de características personales. Sin embargo, realizando un User Research general, podemos llegar a la conclusión de que nuestro perfil de cliente y usuario podrían encajar a la perfección con el de los componentes de nuestro equipo: una persona joven, indiferente su sexo, fundamentalmente joven (entre los 18 y los 35 o 40 años), con facilidad y manejo con las tecnologías (un uso exhaustivo de éstas para numerosos aspectos de su vida, incluidas, especialmente, las compras o consumo online), la pasión por la moda (en nuestro caso) o cualquier otro objeto de consumo concreto, y/o asimismo la necesidad, voluntad o ilusión por generar un modelo de negocio independiente, a su cargo, en línea con sus intereses y en paralelo (adicional) a otra fuente de ingresos más estable o bien a modo de startup o empresa, como dedicación full-time, que consista en un e-commerce.

5.3. Wireframes y mock-ups

A continuación presentamos los prototipos que creamos para las dos aplicaciones de cara a establecer, en las fases iniciales y a medida que avanzábamos en las ideas sobre funcionalidades, en, la estructura general, arquitectura de la información, visualización de los datos, apariencia general y casos de uso o experiencia del usuario a través de ellas.

- Aplicación Android

- Aplicación web (Angular)



Pantalla login



Pantalla inicial

Desktop - 1

Login Principal Agregar producto Ver productos

AGREGAR PRODUCTO

Nombre

Categoría

Descripción

Precio

Precio venta

Cantidad

Imagen

Atrás Subir

FOOTER

Pantalla agregar productos

Desktop - 1

Login Principal Agregar producto Ver productos

Productos



FOOTER

Visualizar productos

5.3. Casos de uso

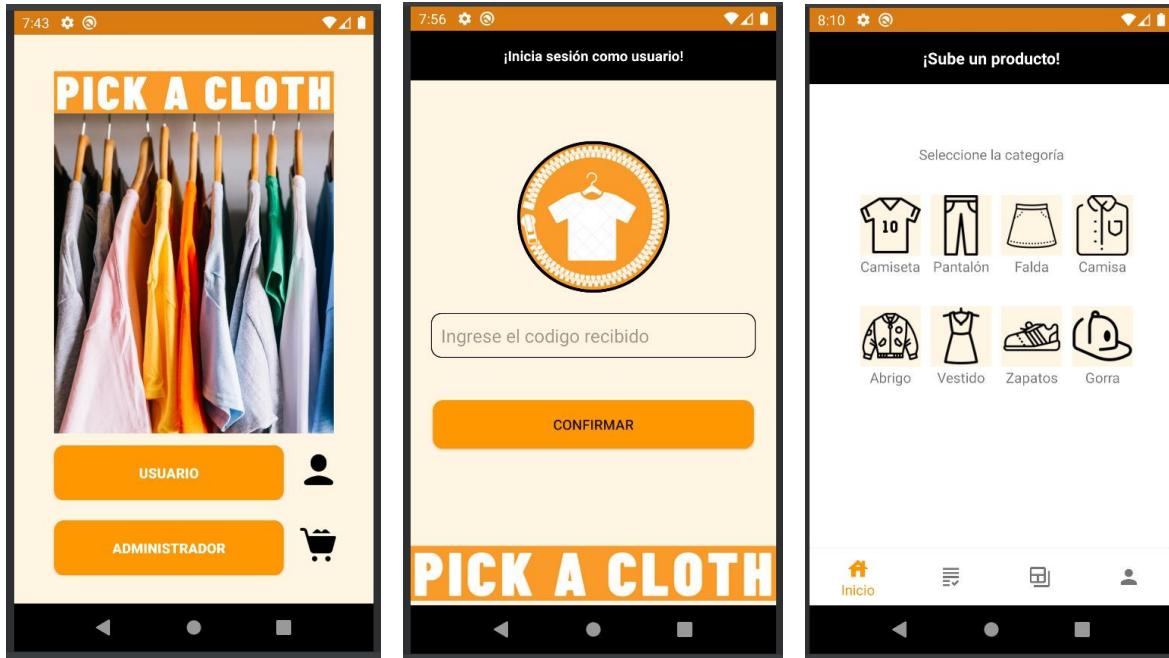
Desde un primer momento nos hemos planteado el elaborar dos aplicaciones, una nativa para Android y una web. Originalmente pensamos en la aplicación móvil para facilitar las tareas diarias de un administrador o gestor de producto/almacén, que necesite ir fotografiando los diferentes productos situados a lo largo del almacén sin estar condicionado a sentarse en un escritorio, y que a su vez su labor de subida de productos a la base de datos tenga una sincronización inmediata con los clientes, que podrán ver productos disponibles, de modo que la productividad de la empresa crezca al recortarse los tiempos. Después fuimos conscientes de la necesidad de implementar una parte para el cliente, que permitiese a la misma empresa, a través de una única aplicación, que sus clientes realicen compras o pedidos de los mismos productos que la empresa ofrece y muestra en tiempo real.

- Aplicación Android

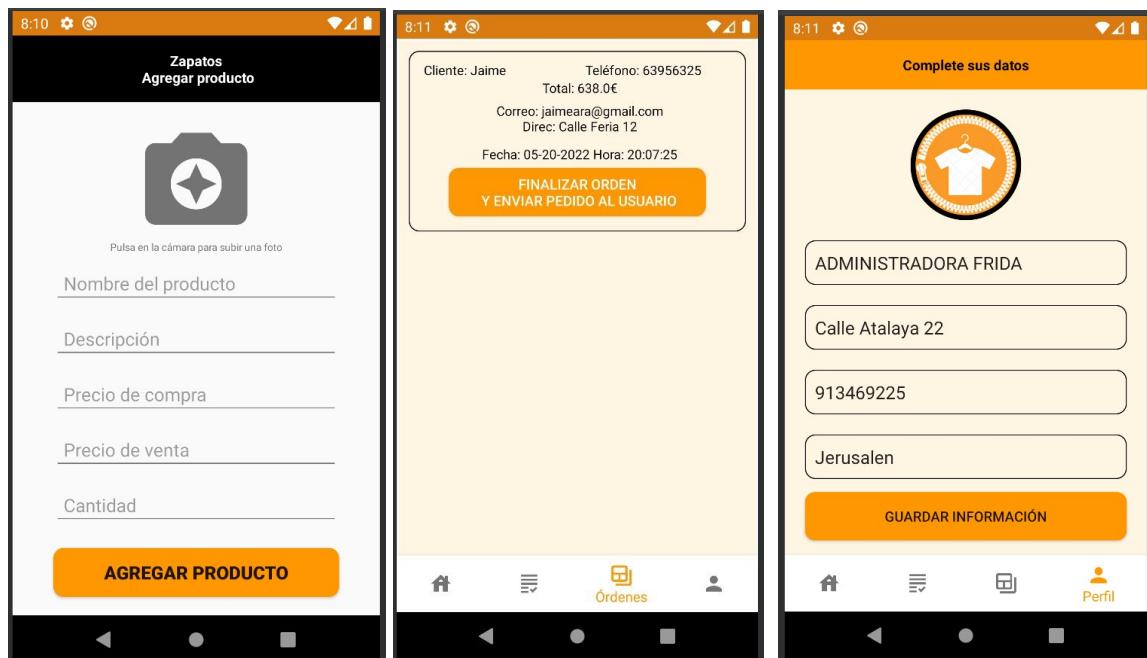
En el caso de la aplicación para Android, un caso de uso normal es el siguiente:

1. El usuario se encontrará con una pantalla de Login, que le permitirá elegir entre rol Administrador o rol Usuario. Por facilidad y comodidad, hemos creado de antemano una serie de usuarios válidos. En función del botón que seleccione, podrán suceder dos cosas o abrirse dos tipos de funcionalidades o utilidades diferentes de la aplicación:
 2. En caso de ser administrador:
 - a) Si es la primera vez que entra, se entra en una pantalla en la que podrá introducir sus datos, lo cual le da de alta y asocia unos datos de perfil en la base de datos.
 - b) Si el usuario no está loggeado (no sesión iniciada), se accede a un Login específico para administrador en que se pedirá un número de teléfono al que se le enviará un código y si ese número y password están en base de datos se loggea.
 - c) Una vez está iniciado en la sesión, estará en su pantalla principal, desde la que desde un menú a pie de página podrá elegir las siguientes acciones:
 - o **Inicio** (subir producto). Puede subir un producto, seleccionando previamente una categoría y a continuación, en otra pantalla, agregar los datos e imagen de un producto de esa categoría.
 - o **Inventario**. Lista todos los productos subidos en la base de datos. (Sin implementar todavía).

- **Órdenes.** Donde se listan las órdenes de compra de los usuarios.
- **Mi Perfil.** Podrá modificar sus datos personales de perfil.



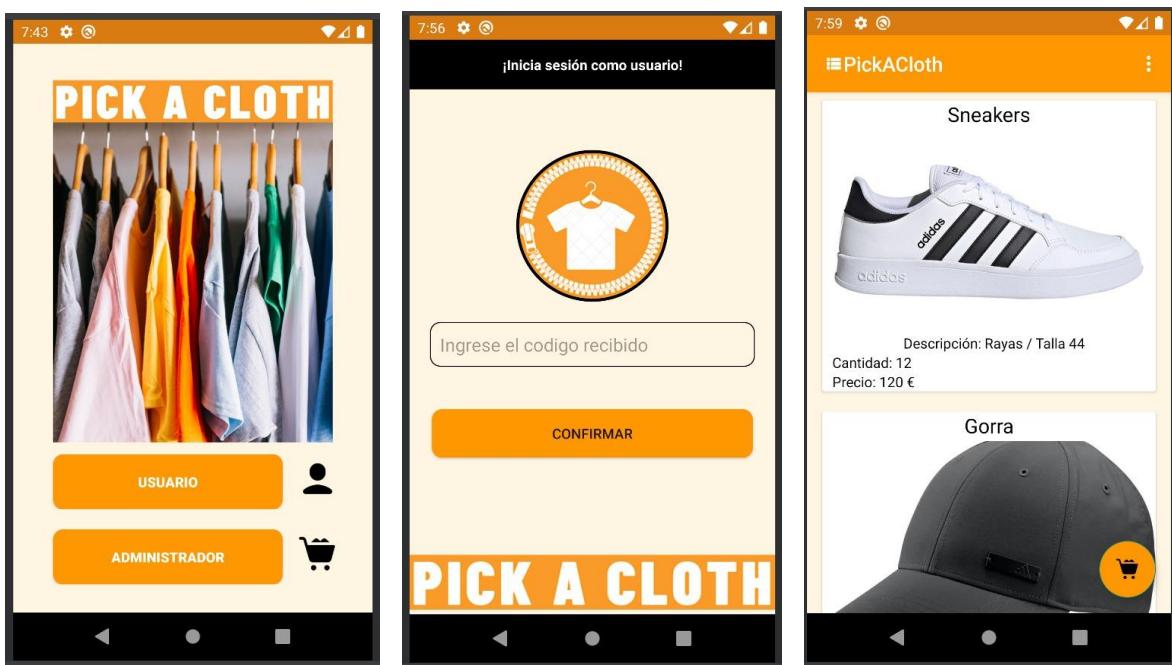
Pantalla bienvenida, inicio de sesión. selección categorías (inicial)



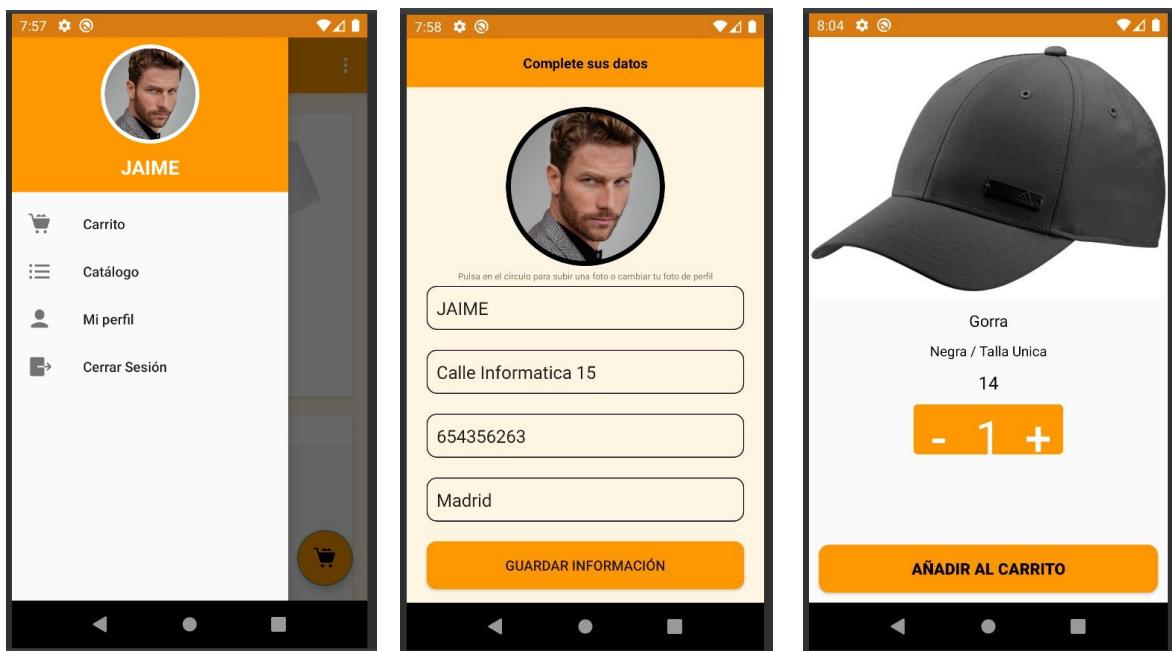
Pantalla subir producto, listado de pedidos y cambiar Mi Perfil

3. Por otra parte, si es un usuario de tipo cliente:

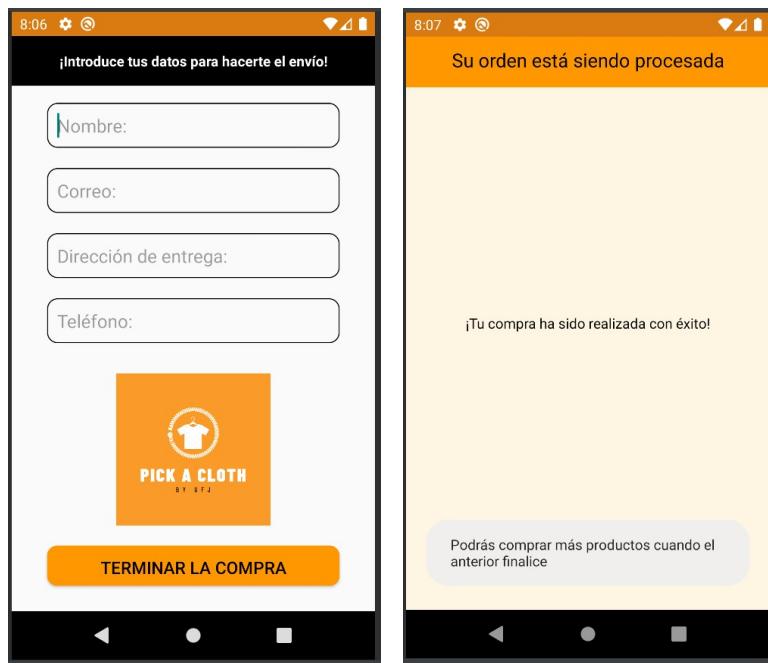
- a) En caso de no estar loggeado previamente, le llevará a un login en que se pedirá un número de teléfono al que se le enviará un código y si ese número y password están en base de datos se loggea.
- b) Una vez está iniciado en la sesión, si es la primera vez que entra, irá a una pantalla en la que podrá ponerse una foto (que podrá coger de su teléfono y modificar) y datos de perfil.
- c) Por lo demás, siempre irá a su pantalla principal, en la que se muestra todo el catálogo de productos de la base de datos. De ésta, desde un menú que se desliza desde la izquierda, donde se muestran nombre y foto de perfil y podrá elegir las siguientes acciones:
 - o **Carrito.** Se muestran todos los productos (con sus cantidades, precios, importe total) seleccionados para comprar. El usuario puede borrar el carrito (que eliminará también el carrito de la base de datos) o bien confirmar el pedido/orden (botón Continuar). Al presionar continuar, pondrá sus datos personales para confirmar el pedido y la orden de envío. Al presionar botón “Terminar la compra”, se le informará de su éxito, y se generará una Orden de compra en la base de datos que a su vez el administrador podrá visualizar. Hasta que el vendedor/administrador no acepte su pedido, el usuario no podrá añadir más productos al carrito: solo 1 pedido por usuario a la vez.
 - o **Carrito.** Se muestran todos los productos (con sus cantidades, precios, importe total) seleccionados para comprar. El usuario puede borrar el carrito (que eliminará también el carrito de la base de datos) o bien confirmar el pedido/orden (botón Continuar). Al presionar continuar, pondrá sus datos personales para confirmar el pedido y la orden de envío. Al presionar botón “Terminar la compra”, se le informará de su éxito, y se generará una Orden de compra en la base de datos que a su vez el administrador podrá visualizar. Hasta que el vendedor/administrador no acepte su pedido, el usuario no podrá añadir más productos al carrito: solo 1 pedido por usuario a la vez.
 - o **Mi perfil.** Podrá modificar sus datos personales y foto de perfil.
 - o **Cerrar sesión.** Sign-out del usuario, obligándole después a repetir el proceso de sign-in.



Pantalla bienvenida, login de usuario, catálogo de productos (inicio)



Menú lateral de usuario, Mi perfil, pestaña para añadir producto



Confirmar pedido y aviso de que ha sido realizado

- **Aplicación web (Angular)**

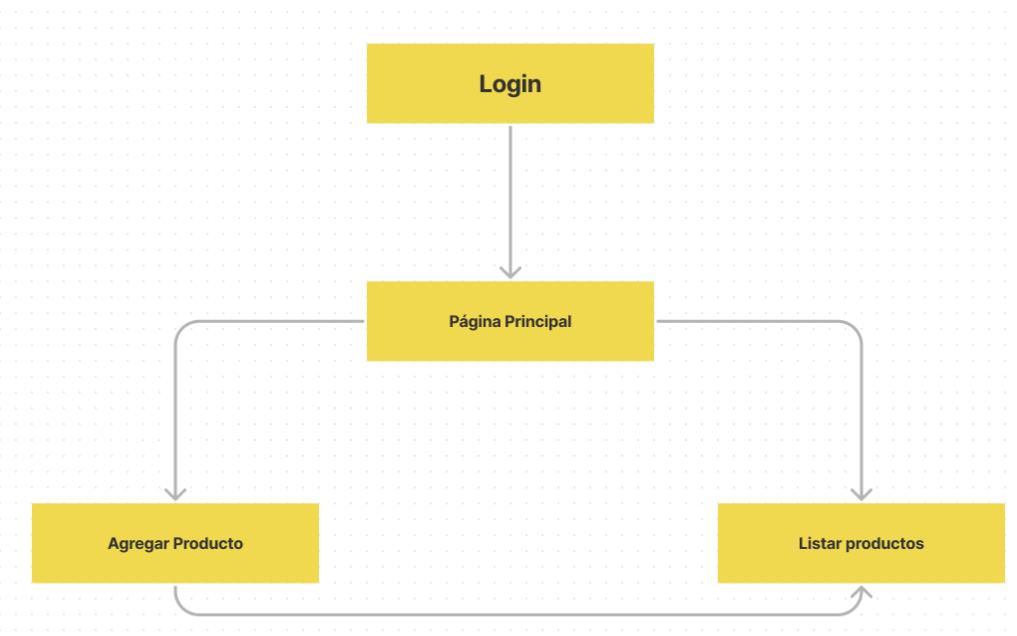
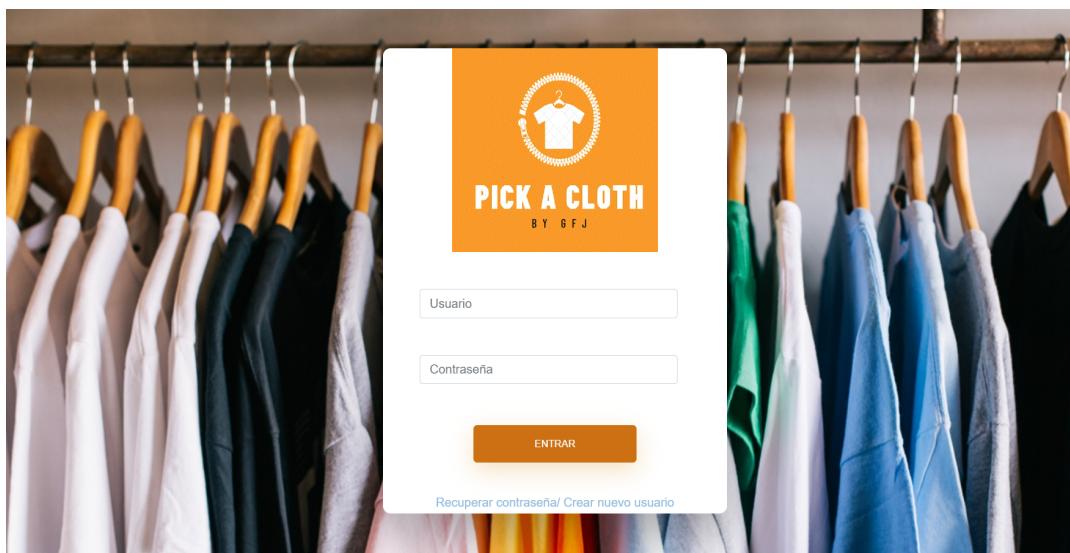


Diagrama de casos de uso



Pantalla de login

Navbar Login Principal Agregar Producto Ver Productos

The main page of the application. It features a large orange header with the "PICK A CLOTH" logo and "BY G F J" text. Below the header is a large white "PICK A CLOTH" title. Underneath the title are two dark blue buttons labeled "Agregar Producto" and "Visualizar". At the very bottom of the page is a thin grey footer bar with the copyright notice "© 2022 Copyright EDIX".

Pantalla inicial

Navbar Login Principal Agregar Producto Ver Productos

AGREGAR PRODUCTO



*Nombre

*Categorías

*Descripción

*Precio

*Precio venta

*Cantidad

*Imagen
 Ninguno archivo selec.

© 2022 Copyright: EDIX

Pantalla agregar producto

Navbar Login Principal Agregar Producto Ver Productos

Productos



Camiseta
Nombre: Blanca / Talla M
Categoría: Camiseta
Precio compra : 15€
Precio Venta: 20€
Cantidad: 5



Pantalla visualizar producto

6. BASE DE DATOS



Tabla **Admin** (administradores). Recoge los datos de administrador que se añaden/modifican en el primer login y en “Perfil”. Campos: ciudad, dirección, nombre, teléfono, uid (user id) que se asigna como identificador único de ese registro.



Desde Firebase (en el navegador) también se podrán añadir, modificar y eliminar registros.



Tabla Usuarios. En ella se guardan los distintos usuarios con sus datos, incluida una imagen que está en el Storage. Campos: ciudad, dirección, nombre, teléfono, imagen y uid (user id) que se asigna como identificador único de ese registro.

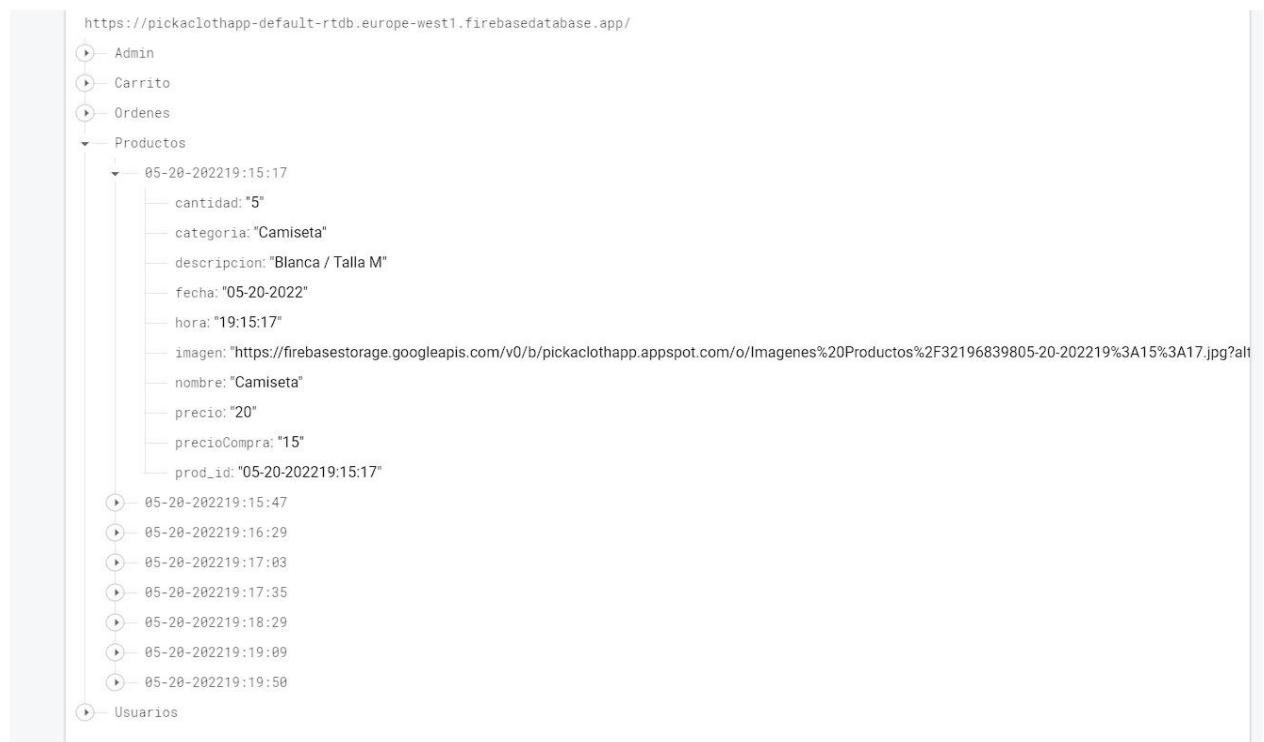


Tabla Productos, donde se van añadiendo los productos que sube el administrador con sus datos y foto (que va al Storage). Cada registro tiene los siguientes campos: cantidad, categoría, descripción, fecha, hora, imagen, nombre, precio, precio de compra, ID de producto. El último se asigna como identificador.



Tabla **Órdenes** (pedidos). El administrador podrá ver desde la base de datos los pedidos u órdenes de sus clientes. De momento, la aplicación solo acepta una orden a la vez por cada usuario, hasta que el administrador la resuelva (acepte/envíe) y el usuario pueda hacer otra.



Tabla Carrito. En Usuario compra se podrán observar los productos que cada usuario tiene añadidos a su carrito.

7. CONCLUSIONES MEJORAS DEL PROYECTO

De manera general, estamos satisfechos con el resultado de nuestras aplicaciones, teniendo en cuenta que hemos conseguido implementar las dos funciones principales que nos habíamos propuesto (subir y mostrar productos) y hemos añadido algunas más a la app de Android. Somos conscientes de que tanto en diseño como en funcionalidad ambas aplicaciones son mejorables. Sin embargo, también nos alegra el haber podido profundizar y avanzar mucho más en conocimientos previos (Android) y en otros completamente nuevos (Angular), lo cual ha supuesto todo un reto dado el corto periodo de tiempo para desarrollar.

Hay una serie de funcionalidades que, debido a esta falta de tiempo, no hemos podido implementar y que consideramos que mejorarían la usabilidad y utilidad de la aplicación. Esperamos poder en un futuro dedicarnos a trabajar en ellas. Son las siguientes:

- Que no haya un registro de usuarios, es decir, opción a crear usuarios desde la aplicación. De momento, solo podrán hacer login aquellos usuarios que hayan sido previamente creados desde Firebase RealtimeDatabase (en el navegador).
- Un inventario o lista actualizada en la que los productos que hayan sido “comprados/pedidos” por clientes, vayan desapareciendo (en la cantidad correspondiente) de la base de datos o inventario/stock. De momento, aunque el administrador confirme la orden no cambia la cantidad de producto.
- Que el usuario pueda buscar por nombre los productos en un buscador, en lugar de recorrer toda la lista, o bien listar/filtrar por categorías.
- Que no se puedan borrar elementos individuales del carrito, es decir, uno por uno. En su lugar, solo se puede borrar el carrito entero, con todos sus elementos.
- De momento, la aplicación solo acepta una orden a la vez por cada usuario, hasta que el administrador la resuelva (acepte/envíe) y el usuario pueda hacer otra. En un futuro, se buscaría ampliar a que el número de órdenes por usuario pueda aumentar, aunque no hayan sido aceptadas por el administrador.

Esperamos poder seguir desarrollando y profundizando en un futuro cercano.

8. MIEMBROS DEL EQUIPO

Frida Abella Fernández

Jaime Aranda Congil

Guillermo Gabard Delgado