

Proyecto Fin de Ciclo

R. RUÍZ Pastelería dulce y salado



Alberto Galán López
R. Ruíz
Proyecto Fin de Ciclo

ÍNDICE GENERAL

1. INTRODUCCIÓN Y OBJETIVOS	2
2. ESPECIFICACIONES DE REQUISITOS	3
3. PLANIFICACIÓN TEMPORAL Y EVALUACIÓN DE COSTES.....	4
4. TECNOLOGÍAS UTILIZADAS.....	5
5. ANOTACIONES AL DESARROLLO E IMPLEMENTACIÓN.....	6
5.1. ICONO.....	6
5.2. ACTIVITY LOGIN.....	6
5.3. ACTIVITY PRINCIPAL	8
5.3.1. FRAGMENTO CAMARA	9
5.3.2. FRAGMENTO GALERIA	10
5.3.3. FRAGMENTO LISTA DE CANAPES Y OTROS	11
5.3.4. FRAGMENTO DE PEDIDOS	12
5.4. ACTIVITY DETALLE GALERIA	13
5.5. ACTIVITY DETALLE CANAPE U OTRO.....	14
5.6. ACTIVITY AÑADIR CANAPE U OTRO	15
5.7. ACTIVITY DETALLE PEDIDO.....	16
5.8. ACTIVITY AÑADIR PEDIDO.....	17

1. INTRODUCCIÓN Y OBJETIVOS

Mi proyecto de fin de ciclo se basará en el desarrollo de una aplicación móvil para un servicio de catering.

En principio nos centramos en cuales son los objetivos que deberá cumplir dicha aplicación para la futura modernización de la empresa. Para ello es necesario tener una base de datos y apoyarse en las herramientas de Firebase de Google.

Los objetivos específicos de este proyecto son los siguientes:

- I. Tener una lista de Canapés. Esto ayudará a la empresa cuando el cliente pregunte por los productos que se fabriquen en el local, además, de incluir foto, ingredientes y precio por canapé.
- II. Lista de otros productos. En el obrador además de canapés se realizan otro tipo de pastelería y servicios de comida. Por ejemplo: tortilla de patata, empanadas de diversos rellenos, pasteles de todo tipo, mazapán, torrijas, etc.
- III. Subir fotos al Storage. Poder guardar fotos de otros caterings, canapés u otra pastelería dulce y salada, como roscones, tartas saladas etc.
- IV. Galería de fotos. Anteriormente subimos fotos al Storage, entonces también debemos poder ver esas fotos, ya sea para poder descargar en el dispositivo donde este la app, o para enseñar el servicio y productos al cliente en mayor detalle.
- V. Por último, y más importante para la modernización, la lista de pedidos recogidos mediante un formulario completo. Esto permitirá que cualquier persona pueda recoger un pedido en cualquier lugar, sin depender de estar en el local o tener un folleto con todos los tipos de servicios que se realizan.

Con todos estos objetivos, he desarrollado una aplicación que pueda suplir todos ellos, pero también que se pueda seguir desarrollando y mejorando para un futuro. La aplicación puede ser objeto de mejoras, pero hasta que no se realice una explotación no se podrá ver donde hacer dichas mejoras.

2. ESPECIFICACIONES DE REQUISITOS

Para la realización de esta aplicación como dijimos antes usaremos los servicios de Google junto con los permisos de galería, por ello, necesitaremos estos requisitos:

- I. Acceso a una red. Ya sea wifi o datos móviles, esto es necesario para poder acceder a la base de datos (Database) y tenerlo actualizado, por si otro usuario añade canapés nuevos. Además de acceder a la base de datos donde pueden estar los canapés u otros productos, también es importante poder acceder a todas las fotos recogidas en el Storage de Google.
- II. Permisos para acceder a la galería del dispositivo. Esto se debe permitir al ejecutar la aplicación por primera vez, sino no se podrá acceder a ella. Este permiso se usa en el momento de subir imágenes al Storage, dicha imagen antes de subirla se recoge de la galería y se pueden hacer modificaciones en la orientación antes de que se suba a la nube.

3. PLANIFICACIÓN TEMPORAL Y EVALUACIÓN DE COSTES

Planificación temporal de las características que se pretenden desarrollar:

- Pantalla de Inicio, donde se sitúa la base del Authentication de Firebase: **10h.**
- Pantalla principal, con un cajón para poder navegar entre los distintos fragmentos:
 - o Cajón: **4h.**
 - o Fragmento lista de canapés: **2h.**
 - o Fragmento galería de fotos: **2h.**
 - o Fragmento cámara: **5h.**
 - o Fragmento lista pedidos: **2h.**
- Pantallas de detalle:
 - o Pantalla de detalle de un canapé: **2h.**
 - o Pantalla de detalle de una foto: **8h.**
 - o Pantalla de detalle de un pedido: **6h.**
- Pantallas de añadir elemento:
 - o Añadir nuevo canapé: **5h.**
 - o Añadir nuevo pedido: **10h.**

Evaluación de costes

Servicio Firebase: por ser un servicio para solo unos pocos dispositivos, para el servicio interno del catering se usará la versión gratuita que supe todos los objetivos. En un futuro se podrá lanzar una aplicación parecida para ser explotada por los clientes, ya sea para hacer pedidos online, revisar precios u otros servicios.

Android Studio: entorno de desarrollo que usaremos el lenguaje Kotlin, es totalmente gratuito.

4. TECNOLOGÍAS UTILIZADAS

Usaremos esencialmente dos tecnologías, una de ellas será el lenguaje en el que trabajemos y otro será Firebase, herramientas de Google. No usaremos mas para así no depender de otros proveedores y, como dijimos anteriormente, tener un mínimo de coste en la creación del proyecto.

- Vamos a usar el lenguaje de Kotlin, *“Lenguaje de programación de tipado estático que corre sobre la máquina virtual de Java y que también puede ser compilado a código fuente de JavaScript. Es desarrollado principalmente por JetBrains en sus oficinas de San Petersburgo (Rusia)”*.

Por ello, usaremos el entorno de desarrollo de Android Studio que nos permite programar totalmente con este lenguaje y muchas más herramientas.

- Firebase. Es una plataforma ubicada en la nube que dota a los usuarios de una alta calidad de servicios y ventajas:
 - o La sincronización de los datos de los proyectos sin administrar conexiones entre usuarios y sin usar lenguaje complejo.
 - o Se integra fácilmente en cualquier tipo de plataforma ya sea web o aplicaciones móviles, además es compatible con grandes plataformas, como IOS, Android, aplicaciones web, Unity y C++.
 - o Se escala automáticamente a cualquier tipo de aplicación al usar una infraestructura de Google, desde aplicaciones pequeñas a las más potentes.
 - o Creación de proyectos sin necesidad de un servidor. Las herramientas se incluyen en los SDK para los dispositivos móviles y web.

Además de todas estas ventajas, también tiene una gran documentación para los usuarios a la hora de crear sus proyectos, también se puede encontrar otra documentación en Github y en StackOverflow

Junto con estas dos herramientas muy potentes, desarrollaré una aplicación rápida, efectiva y reactiva para todos los usuarios a tiempo real. Y sin la necesidad de tener un servidor fijo y activo durante todo el tiempo. A parte, también hay ejemplos de donde poder ayudarme para sacar ideas o problemas que surjan al mismo tiempo que se programa.

5. ANOTACIONES AL DESARROLLO E IMPLEMENTACIÓN

En este apartado vamos a explicar todo el desarrollo de la aplicación desde que se abre la aplicación hasta la última pantalla implementada (desde ahora, activity o fragmento), deteniéndonos en cada una para observar los objetivos y diseño de esta.

5.1. ICONO

Comenzamos con el icono de la aplicación, una implementación sencilla y rápida que le da estilo y mayor personalidad a la aplicación.



5.2. ACTIVITY LOGIN

Tras pulsar el icono y comenzar la aplicación, lo primero que aparece es aceptar el permiso de acceso a las imágenes de la galería del dispositivo y, posteriormente, tras aceptar debemos pensar en las funcionalidades de esta pantalla.

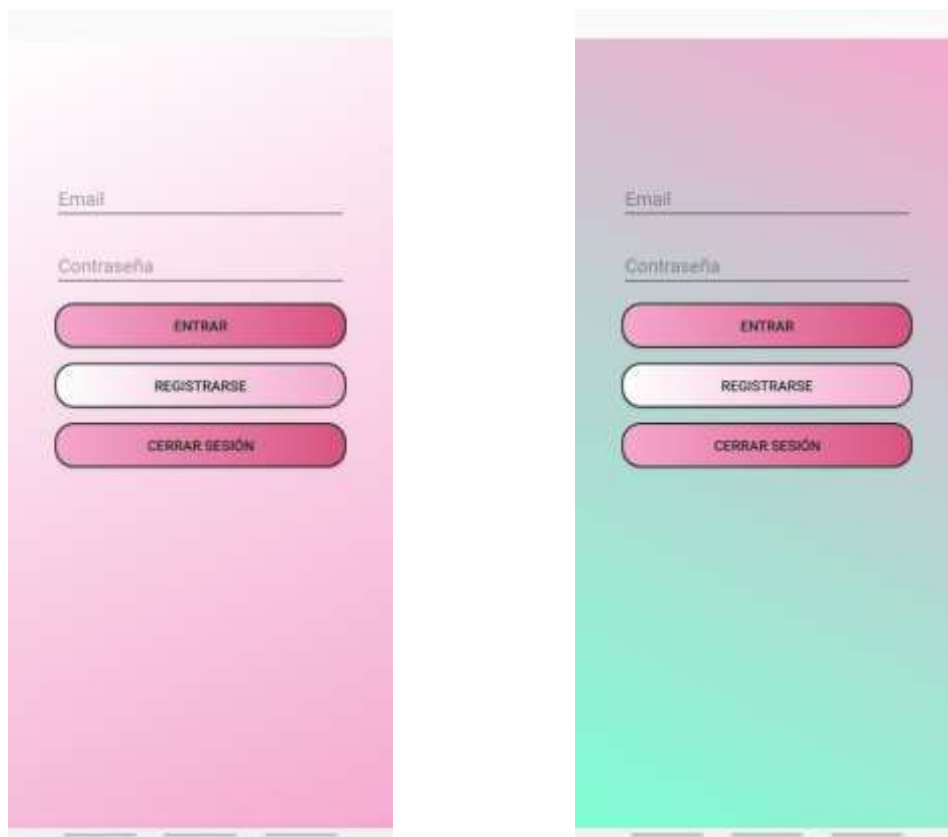
Todas las funcionalidades de esta activity están relacionadas con la herramienta de Authentication de Firebase.

- El primero es que no puedan acceder cualquier persona sin estar registrada. Por el cual, lo primero que debe hacer una persona al encontrarse con la aplicación es formalizar un registro. Este registro debe tener algún tipo de seguridad por el cual el usuario comenzará a escribir en las casillas de texto (EditText) su email y su contraseña - viene especificado en cada una por medio de un indicio – posteriormente, presionará el botón de registrarse que este conllevará a que le aparezca un diálogo a cumplimentar con su nombre en la cuenta que está creando.

Al aceptar dicho diálogo el usuario se ha registrado, pero no puede acceder todavía a la aplicación. Mientras el usuario realiza todas estas acciones le aparecerán pequeños textos (Toast) para indicar el proceso en el que irá y si se ha concluido todos los procesos correctamente. Tras completar el nombre del usuario aparecerá un texto en el que se le indica al usuario que debe acceder al email anteriormente escrito para confirmar su cuenta y así poder acceder. Este medio de confirmación de cuenta es un método de seguridad para que un usuario solo se registre con un email existente y no solo válido.

- Un usuario ya registrado y confirmado el email de confirmación ya puede acceder a la aplicación, de una manera muy sencilla, cumplimenta las casillas de texto con el email con el que se registro y la contraseña de éste. Posteriormente, debe pulsar el botón de entrar y comenzará el uso de la aplicación en sí, dejando atrás esta activity de administrador de usuarios.
- Cualquier usuario que ya haya entrado en la aplicación haciendo el paso anterior, tampoco deberá hacer este proceso todas las veces que entre, ya que el dispositivo mantendrá la sesión iniciada y le reconocerá, pasando perfectamente a la siguiente activity.
- Pensando en que varias personas puedan acceder con distintas cuentas en un mismo terminal, también se dispone de un botón para cerrar la sesión existente. O simplemente para que no acceda directamente la aplicación sin pasar por esta activity.

Respecto al diseño de esta activity encontraremos dos EditText, con sus respectivos indicios de email y contraseña para que pueda cumplimentar el usuario; y tres botones, el primero de entrar, el segundo registrarse y tercero cerrar sesión. También se debe tener en cuenta el fondo de la activity, ya que, es un fondo animado, en el que los colores van cambiando con el paso del tiempo.



5.3. ACTIVITY PRINCIPAL

En esta Activity nos centraremos en la base de navegación entre las distintas herramientas que nos proporciona toda la aplicación, y en lo que se basa la finalidad de la activity. Poder organizar e informatizar toda la producción y recoger los pedidos que se encarguen a una empresa de catering

Para poder navegar entre los distintos fragmentos he incorporado un cajón donde poder mostrar una pequeña cabecera con la imagen corporativa de la empresa, el nombre del usuario y su email con el que se registró, a su vez distintos textos que al pulsar nos lleva a su fragmento correspondiente, como podemos observar en la figura siguiente:



Tenemos 5 distintos fragmentos, aunque en la imagen contemplemos 4, el quinto es parecido a la lista de canapés, pero ofreciendo los otros productos que se ofrecen en el catering.

5.3.1. FRAGMENTO CAMARA

Aquí nos encontramos con la funcionalidad de poder subir imágenes al Storage de Firebase, lo que es lo mismo que una base de datos donde almacenar imágenes, pero esta foto la podremos elegir, mediante un botón de “Seleccionar foto”, de la galería del dispositivo de una manera muy sencilla, ya que, como comentamos al principio pide los permisos necesarios al usuario para realizar esta acción. Posteriormente si la foto no queda al gusto del consumidor puede rotarla pulsando otro botón con la imagen para este fin, un detalle para guardar la imagen en la posición correcta y poder visualizarla con corrección.

Por último, tenemos el botón de “Subir foto” que concluye este fragmento y se usa para subir la imagen que se muestra con un nombre, que se completa en un diálogo tras pulsar dicho botón. Tras aceptar el diálogo con el nombre, aparecerá

un texto en la parte inferior del dispositivo en el que indicará si la acción pudo realizarse con éxito.

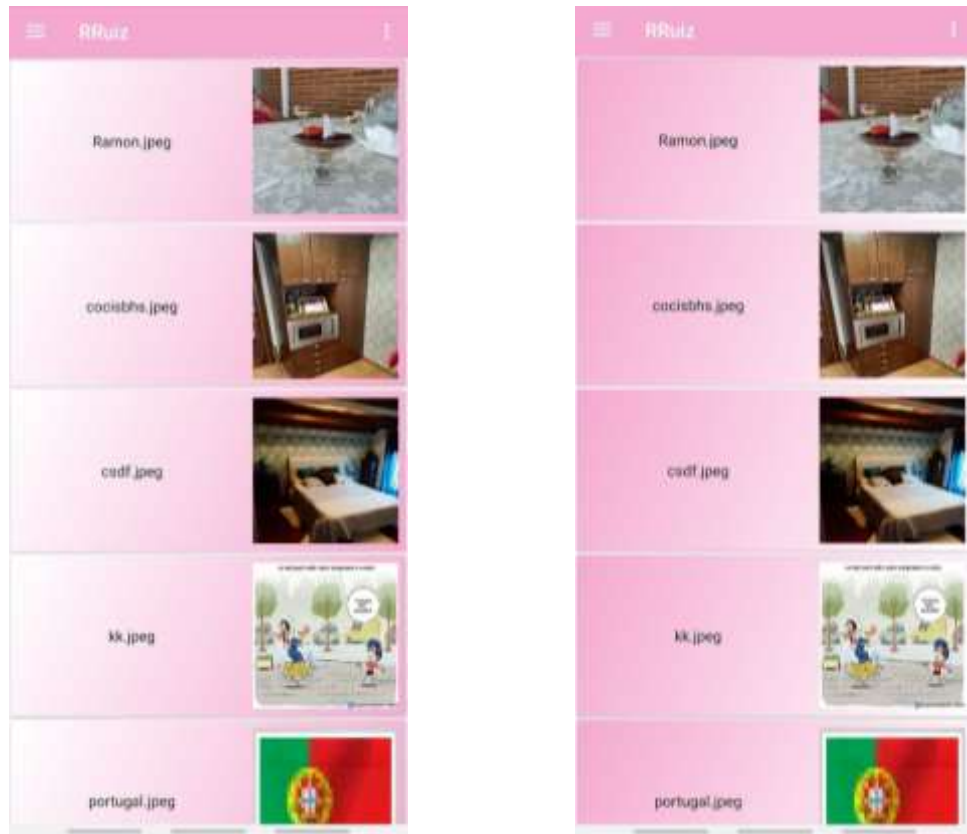
En el diseño de este fragmento nos centramos en un ImageView, una vista donde poder visualizar la imagen que queremos subir. Tiene tres botones, dos de ellos con texto explícito y uno último con una imagen con el símbolo de rotar la imagen en el sentido de las agujas del reloj, tal como mostramos en la siguiente figura:



5.3.2. FRAGMENTO GALERIA

Continuando con los fragmentos nos encontramos con un simple fragmento para poder mostrar todas las imágenes que anteriormente los usuarios de la aplicación subieron por medio del fragmento de la cámara. Aquí aparece un RecyclerView con un diseño propio para ajustarse a las necesidades de éste que describiremos posteriormente. En dicho RecyclerView se podrá seleccionar una imagen de las que se muestra en las filas para poder verla en mas detalle en una activity, que posteriormente comentaremos más en profundidad.

El diseño de este fragmento se basa en las filas del propio RecyclerView y, por el cual, he decidido poner un fondo cambiante con un tono rosa y otro blanco, junto con un TextView donde colocar el nombre de la imagen y un ImageView donde visualizar en pequeño dicha imagen.



5.3.3. FRAGMENTO LISTA DE CANAPES Y OTROS

Aquí describiremos dos fragmentos a la vez, ya que, describirán de la misma manera dos productos distintos. La finalidad de estos fragmentos es mostrar una lista en un RecyclerView donde poder ver todos los canapés y otros guardados en el Firestore (base de datos del servicio de Google), de una manera rápida y sencilla. Además, cuando se pulse en uno específico se ejecutará una nueva activity donde poder ver dicho producto seleccionado en detalle. Asimismo, contiene un botón flotante en la parte inferior derecha donde poder añadir nuevos canapés a esta base de datos. Las activitys de detalle y añadir nuevos productos los detallaremos más adelante.

El diseño es parecido al del fragmento de galería con un fondo animado de dos colores, esta vez rosa y verde agua, con un TextView con el nombre del producto y una ImageView del producto en cuestión.



5.3.4. FRAGMENTO DE PEDIDOS

Para acabar con los fragmentos de la pagina principal tenemos una lista en un RecyclerView de pedidos, donde poder recoger los pedidos y, con un diseño rápido que permite ver en resumen el pedido. Posteriormente al pulsar se podrá ver en detalle, y con un botón flotante en la parte inferior para añadir un pedido nuevo. Dichas dos activitys las explicaremos en detalle más adelante.

Por ultimo el diseño del fragmento se centra en ver el nombre del que hizo el pedido y su número de teléfono en la parte izquierda siendo estos dos TextView. En la parte derecha, de arriba abajo, se encuentra la fecha del pedido, la cantidad de todos los canapés seleccionados (TextViews) y, por último, un CheckBox deshabilitado que únicamente mostrará si es un pedido a recoger en tienda o es para repartir a domicilio, como mostraremos en la siguiente figura:



5.4. ACTIVITY DETALLE GALERIA

En esta activity nos centramos en poder ver la imagen en un tamaño más grande, tras pulsar una fila específica en el fragmento de la galería, con su nombre y un botón para poder ver en detalle dicha foto. Este botón tiene como principal función llamar al navegador del teléfono y acudir a la ruta (url) de la imagen, así poder hacer zoom y moverla libremente, incluso descargarla en nuestro dispositivo para poder verla desde nuestra galería o compartirla por otras aplicaciones.

El diseño que describimos es un ImageView donde ver la imagen con un tamaño notablemente mayor a la del RecyclerView y su respectivo nombre y, por último, el botón de detalle.



5.5. ACTIVITY DETALLE CANAPE U OTRO

Activity con la funcionalidad de informar al usuario con una imagen del producto, su nombre, una breve descripción de éste y su coste por unidad. Una activity realizada con Binding desde que se pulsa en un producto desde su respectivo RecyclerView hasta poder ver en detalle del mismo. Es meramente informativa para el usuario.

Su diseño como describimos anteriormente esta formado por un ImageView y diversos TextView para poder expresar toda la información de la mejor manera posible. Mostramos dos ejemplos del detalle de canapés.



5.6. ACTIVITY AÑADIR CANAPE U OTRO

En esta activity nos centramos en rellenar un formulario para, posteriormente, subirlo a Firestore, en el apartado de canapés u otros. En primer lugar, seleccionar una foto de la galería del teléfono tras pulsar el botón de añadir foto. A continuación, rellenar con un nombre, una breve descripción y el precio de dicho producto, los dos primeros son EditText en el que escribir, pero y el último es un EditText numérico que permite decimales para añadir así un precio mas realista, al poder poner céntimos al precio. Para terminar el formulario pulsar el botón de “Aceptar”. Si se han rellenado todos los anteriores campos, se subirá el nuevo canapé a la lista, se saldrá de la activity y volverá al fragmento de la lista de canapés u otros donde podremos localizar el canapé nuevo recién incluido.



The screenshot shows a mobile application interface for 'R. Ruiz Pastelería dulce y salada'. At the top is a pink header with the text '#R Ruiz'. Below the header is a circular logo with the text 'R. RUIZ' and 'Pastelería dulce y salada'. Under the logo is a pink button labeled 'AÑADIR FOTO'. Below this are three text input fields labeled 'Nombre', 'Descripción', and 'Precio'. At the bottom is a pink button labeled 'ACEPTAR'.

5.7. ACTIVITY DETALLE PEDIDO

Una activity meramente informativa donde mostrar un pedido en detalle y en un tamaño mayor que al RecyclerView. Aquí encontraremos TextViews con el nombre, número de teléfono, y la calle si fuera a domicilio; un CheckBox que estará deshabilitado, pero mostrará si es un pedido a recoger o es a domicilio. Para acabar dos TextViews mostrando cantidad y nombre del canapé que se crearán en código, por el cual, se mostrarán más o menos filas dependiendo de las dimensiones del pedido. Todos estos TextViews estarán enmarcados con un cuadrado rosa para realzar el contenido del pedido.

Mostraremos a continuación dos pedidos de ejemplo.



5.8. ACTIVITY AÑADIR PEDIDO

Esta activity es la mas compleja de toda la aplicación, ya que debe tener en cuenta muchos factores. Todos estos son los necesarios para recoger un pedido con todo detalle:

- Un EditText donde cumplimentar el nombre del cliente.
- Otro EditText de formato numérico para introducir el número de teléfono al que contactar si fuese necesario.
- Un botón que ejecutará una serie de diálogos en los que se podrá seleccionar el día de la entrega o evento y, posteriormente, la hora con unos diálogos cómodos e intuitivos.
- Posteriormente un CheckBox que indica si es a domicilio o es para recoger en tienda, que tiene a su vez la función de habilitar o no el EditText para cumplimentar con la dirección al que desplazarse.
- Ahora toca la parte de la selección de productos, tras pulsar el botón que lo indica, “Añadir productos”, se muestra un diálogo en el que debemos seleccionar si lo que queremos añadir es canapé u otro. Si se eligió

canapé, se ejecutará un diálogo en el que seleccionar la cantidad de canapés, una bandeja completa cuenta con 20 canapés, entonces para la hora de hacer pedidos de una manera más dinámica y rápida se ejecuta este diálogo auxiliar para seleccionar de manera rápida un número inicial de canapés, suelto empezando en 0, media bandeja que son 10 y una bandeja completa de 20 canapés, y posteriormente aparecerá un diálogo de multiselección para escoger entre los distintos tipos de canapés registrados en la aplicación. Si se eligió otro, aparece directamente el diálogo es de multiselección, o sea que, puedes elegir más de un producto a la vez, lo que facilita en no tener que pulsar el botón de añadir producto por cada producto similar a este.

- Tras añadir cualquier producto se crea en código una línea en el que aparecen 2 botones con los cuales podemos incrementar o disminuir la cantidad del producto indicado en el centro de estos, y a su derecha encontramos el nombre del producto seleccionado. Si se desea eliminar un producto de la lista, el método es llegar a cantidad 0 y pulsar el botón de menos, esto ejecutará un diálogo donde preguntará al usuario si está seguro de la eliminación, por si acaso fuera una equivocación, tras aceptar se elimina toda la fila y puede volver a añadirse sin ningún problema.
- Para acabar hay un TextView que se actualiza por cada producto añadido, eliminado o modificado su cantidad para poder aportar un rápido presupuesto al cliente mientras se realiza el pedido.
- Tras cumplimentar todo el formulario con los datos del cliente y los productos que desea, solo falta dar al botón de “Aceptar”, sino tiene todos estos apartados rellenos no deja guardar el pedido en la base de datos. Por último, se cerrará la actividad de creación de un pedido, y nos dejará en el fragmento de la lista de pedidos donde podremos encontrar el pedido que acabamos de registrar.

Todas estas funcionalidades y botones del diseño podremos verlo de una manera más esquemática en la siguiente figura.

RRuiz

Nombre:

Teléfono:

FECHA 18:30 5 de Mayo

☐ DOMICILIO

AÑADIR PRODUCTO

-	0	+	Torta de esadillo con lomo de ahín aromatizado con reducción de Pedro Jiménez y planta
-	10	+	Cucharilla de paté con manzana caramelizada
-	20	+	Nido de hojaldre con huevo frito de codorniz y chistorra

Total: 24,00 €

ACEPTAR

6. CONCLUSIONES Y LINEAS FUTURAS

6.1. CONCLUSIONES

Este proyecto es una simple forma de innovar el proceso de registro de pedidos, así como, disfrutar de opciones como la galería o subir fotos. Consiste en pasar de algo que se hacía en papel a un método más dinámico, pero con todas las ventajas del papel, es decir, poder rectificar en cualquier lugar y de cualquier manera. Además, otra de las ventajas del dispositivo consiste en ofrecer todos los canapés disponibles sin tener que escribir el nombre, impidiendo así que falte algún contenido, como el numero de teléfono, o cometer fallos de escritura que imposibilite su identificación.

También debemos destacar que tener una aplicación de cara a los clientes es mejor, ya que, no solo se tiene un papel con una lista de productos, sino que se tiene la lista con una foto de cada uno, y si fuera necesario hasta una breve descripción de este y su precio. La aplicación también dispone no solo de fotos de canapés sino un apartado exclusivo de fotos donde poder ver algunos servicios, mantelería, mesas, decoración...

En conclusión, un proyecto que puede ofrecer múltiples opciones, con ideas que diremos a continuación, pero que es completamente funcional y disponible para su explotación.

6.2. LINEAS FUTURAS

Para mejorar el proyecto, respecto a su fragmento de cámara, existen dos posibles mejoras. Una de ellas consiste en que, si se registran dos fotos con el mismo nombre, la última se sobrescribe, por lo cual habría que hacer un método para informar al usuario que ese nombre ya existe. La segunda idea consiste en que, una vez elegida la imagen, se pueda dejar en el repositorio de canapés, decoración, catering, que explicamos a continuación.

Siguiendo con el fragmento de la galería se usaría un fragmento intermedio entre el RecyclerView final y la pantalla principal con tres botones para dirigirse a 3 directorios diferentes y así poder clasificar todas las imágenes que se van

subiendo a la plataforma, estas secciones pueden ser canapés, decoración y catering como dijimos anteriormente.

Respecto al fragmento de canapés, se puede incluir en el objeto del canapé mas apartados como alergenicos, tipo (dulce o salado) o a quien va dirigido este canapé (niños o adultos). Todos estos apartados habría que incluirlos en el apartado de creación de canapé como en su detalle.

Para terminar, el fragmento con mayor posibilidad de cambios, el de los pedidos, se puede añadir un registro histórico, donde en su fragmento principal aparezcan solo los pedidos del día actual en adelante, y en el apartado de histórico que aparezcan los anteriores. También se puede añadir, que al seleccionar un pedido este se pueda modificar o incluso eliminar.

7. BIBLIOGRAFIA

Pulido, Francisco (2020). Curso Desarrollo Aplicaciones Multiplataforma Drive.

De las diferentes carpetas de Drive que he utilizado:

- I. Documentación Kotlin.
- II. 5. Retrofit2 Coroutines LiveData MVVM.
- III. 8. ImagePicker.
- IV. 10. RecyclerView para fragmentos.
- V. 15. Firestore.

Google. Documentación Firestore. <https://firebase.google.com/docs?hl=es-419>

(Última vez visitado 25 de mayo de 2020).

Github. <https://github.com/> (Última vez visitado 25 de mayo de 2020).

StackOverflow. <https://stackoverflow.com/> (Última vez visitado 25 de mayo de 2020).