



# Manual Tecnico



# Descripción del Proyecto

¿Cuál es el nombre de la aplicación?

El nombre de la aplicación es Coffee YAO

Es una aplicación móvil desarrollada en Android Studio, diseñada y dirigida para el público en general. Su objetivo principal es digitalizar la experiencia de compra de una de las bebidas más consumidas a nivel mundial como lo es el café en muchas de sus variantes.

- Explorar el menú completo de productos (café, postres y platillos)
- Realizar pedidos de forma rápida y sencilla
- Personalizar su experiencia mediante favoritos
- Consultar su historial de compras anterior



# Índice

## 1. Descripción del Proyecto

- 1.1 Introducción
- 1.2 Objetivos
- 1.3 Alcances

## 2. Tecnologías y Herramientas

- 2.1 Lenguajes de Programación
- 2.2 Base de Datos
- 2.3 Herramientas de Desarrollo

## 3. Arquitectura del Sistema

- 3.1 Patron de diseño
- 3.2 Instalación de dependencias
- 3.3 Configuración de Firebase
- 3.4 Ejecución de la Aplicación

## 4. Módulos y Funcionalidades

- 4.1 Módulo de Autenticación
- 4.2 Módulo de Productos
- 4.3 Módulo de carrito de Compras
- 4.4 Módulo de Favoritos
- 4.5 Módulo de Perfil e Historial

## 5. Interfaz de Usuario

- 5.1 Flujo de Navegacion
- 5.2 Diseño de Pantallas



# Tecnologías y Herramientas

## Lenguajes de Programación

Kotlin fue el principal y unico lenguaje de programación utilizado en esta aplicación.

## Herramientas de Desarrollo

Android Studio, Android Emulator y Firebase Console fueron las principales herramientas que ayudaron al desarrollo de esta aplicacion para un mejor desarrollo.

## Frameworks y Librerías

Firebase Authentication, ViewPager2 y glide fueron algunas librerías utilizadas en esta app.

## Base de Datos

Firebase Firestore y Firebase Authentication fueron las unicas base de datos utilizadas en nuestra aplicacion para el manejo de datos.

# Arquitectura del Sistema

## Patron de diseño

MVVM (Model-View-ViewModel) - Aunque estamos usando servicios, la estructura sigue principios MVVMSingletons para servicios (AuthService, CarritoService, etc.)  
Navegación por Fragments con BottomNavigationView

## Estructura de Carpetas

```
com.example.coffee/  
├─ activities/      # Activities principales  
├─ fragments/      # Fragments de navegación  
├─ adapters/       # Adaptadores para RecyclerViews  
├─ services/       # Lógica de negocio y Firebase  
└─ models/         # Modelos de datos
```

## Diagrama de Componentes



# Configuración del Entorno

## Requisitos del Sistema

- Android Studio Arctic Fox o superior
- Android SDK API 24 mínimo (Android 7.0)
- Dispositivo o emulador con Android 7.0+
- Conexión a Internet para Firebase

Dependencias principales en build.gradle:

```
dependencies {  
    implementation 'com.google.firebase:firebase-auth-  
ktx:22.1.2'  
    implementation 'com.google.firebase:firebase-firestore-  
ktx:24.7.0'  
    implementation 'com.google.android.material:material:1.9.0'  
    implementation 'androidx.viewpager2:viewpager2:1.0.0'  
    implementation 'com.github.bumptech.glide:glide:4.14.2'  
}
```

## Configuración de Firebase

1. crear proyecto en Firebase Console
2. Agregar aplicación Android (com.example.coffee)
3. Habilitar Autenticación (Email/Password)
4. Habilitar Firestore Database

## Ejecución de la Aplicación

1. Clonar repositorio
2. Abrir Android Studio
3. Sincronizar Proyecto
4. Ejecutar en Emulador

# Estructura de la Base de Datos

## Esquema de Firestore

```
firestore/  
├── usuarios/  
│   ├── {userId}/  
│   │   ├── email: string  
│   │   ├── fechaRegistro: timestamp  
│   │   ├── favoritos/  
│   │   │   ├── {productoId}:  
│   │   │   │   ├── nombre: string  
│   │   │   │   ├── precio: number  
│   │   │   │   ├── categoria: string  
│   │   │   │   └── imagenUrl: string  
│   │   └── pedidos/  
│   │       ├── {pedidoId}:  
│   │       │   ├── fecha: string  
│   │       │   ├── total: number  
│   │       │   ├── estado: string  
│   │       │   └── productos: array  
│   │       │       ├── nombre: string  
│   │       │       ├── cantidad: number  
│   │       │       └── precio: number
```

## Colecciones y Documentos

usuarios: Datos de usuarios registrados

favoritos: Subcolección de productos favoritos por usuario

pedidos: Subcolección de historial de pedidos por usuario



# Módulos y Funcionalidades

## Módulo de Autenticación

- Registro de usuarios - Crear nueva cuenta con email y contraseña
- Inicio de sesión - Acceso con credenciales existentes
- Persistencia de sesión - Usuario permanece logueado
- Cierre de sesión - Cerrar sesión de forma segura

## Módulo de Productos

- Catálogo por categorías - Cafés, postres, platillos
- Detalles de productos - Información completa e ingredientes
- Búsqueda visual - Navegación intuitiva por categorías

## Módulo de Carrito de Compras

- Agregar productos - Desde cualquier pantalla de productos
- Gestión de cantidades - Aumentar/disminuir cantidad de productos
- Cálculo automático - Total se actualiza en tiempo real
- Proceso de checkout - Finalizar compra y guardar en historial

## Módulo de Favoritos

- Agregar/eliminar - Gestión de productos favoritos
- Sincronización - Favoritos guardados en la nube
- Acceso rápido - Lista dedicada de productos favoritos

## Módulo de Perfil e Historial

- Información de usuario - Email y datos básicos
- Historial de pedidos - Lista completa de compras anteriores
- Interfaz por pestañas - Navegación entre perfil e historial



# Interfaces de Usuario

## Flujo de Navegación



## Diseño de Pantallas Principales

- Login/Registro: Formularios simples con validación
- Inicio: Grid de 3 categorías (Cafés, Postres, Platos)
- Productos: Cards con imagen, nombre, precio y acciones
- Carrito: Lista editable con controles de cantidad
- Detalles: Vista completa con ingredientes y botón agregar
- Perfil: Pestañas con información usuario e historial

## Componentes Reutilizables

- ProductosAdapter: Para grids de productos
- CarritoAdapter: Para lista del carrito con controles
- FavoritosAdapter: Para lista de favoritos
- HistorialAdapter: Para historial de pedidos

# Pruebas y Depuración

## Casos de Prueba Principales

- Registro e inicio de sesión de usuarios
- Navegación entre todas las pantallas
- Agregar/eliminar productos del carrito
- Gestión de productos favoritos
- Proceso completo de compra
- Visualización de historial de pedidos

## Manejo de Errores

- Validación de formularios (email, contraseña)
- Manejo de conexión a Internet
- Mensajes de error de Firebase
- Fallback para datos no disponibles

## Logs y Monitoreo

- Logs de debug para seguimiento de operaciones
- Monitoreo de errores en Firebase Console
- Verificación de reglas de seguridad

## Preparación para Producción

- Optimización de imágenes y recursos
- Configuración de ProGuard/R8
- Pruebas en dispositivos reales
- Verificación de reglas de Firestore

## Mejoras Futuras

- Sistema de pagos integrado
- Notificaciones push
- Programa de fidelidad
- Reservas de mesas
- Pedidos para llevar con geolocalización