Documentación de FutTFG



Desarrollado por: Yurik Franco Castaño

Desarrollo de aplicaciones multiplataforma

24/25

[1. Introducción](#_7es9nx8l6k81)

[2. Arquitectura de la Aplicación](#_e8ywxbxutovh)

[3. Instalación y Configuración](#_4rpa7f8dygv7)

[Requisitos del Sistema:](#_u5bf94fkc28v)

[Dependencias Principales:](#_hwelsna5lrqq)

[4. Estructura del Proyecto](#_eawikd8mrzs)

[5. Funcionalidades Principales](#_a0s3p9itzswu)

[Gestión de Usuarios:](#_gx91xmaxe3ap)

[6. Uso de Firebase](#_t9xivoebbd2z)

[Authentication:](#_eu8o4d7z2qxg)

[Cloud Firestore:](#_n43devrie3ny)

[Storage:](#_q09l52cgbs8v)

[Security Rules:](#_wm0pgko8pscw)

[7. Pantallas y Flujos de Usuario](#_oc6a3b1iw0ub)

[8. Conclusiones](#_lek155g023a3)

# 

# 

# 1. Introducción

FutTFG es una aplicación móvil Android desarrollada en Kotlin que funciona como una plataforma de compra-venta de artículos deportivos, especialmente enfocada en productos relacionados con el fútbol. La aplicación permite a los usuarios registrarse, publicar productos para la venta, buscar artículos, realizar compras y gestionar su perfil de usuario.

# 2. Arquitectura de la Aplicación

Patrón MVVM (Model-View-ViewModel): La aplicación implementa el patrón de arquitectura MVVM, separando claramente la lógica de negocio de la interfaz de usuario.

Componentes de Arquitectura Android: Utiliza LiveData para la observación de datos y ViewModel para la gestión del estado de la UI.

Firebase como Backend: Implementa Firebase como solución backend, utilizando varios de sus servicios principales.

Navigation Component: Utiliza el componente de navegación de Android para la gestión de fragmentos y flujos de navegación.

# 3. Instalación y Configuración

### Requisitos del Sistema:

* Android SDK mínimo: 24 (Android 7.0)
* Android SDK objetivo: 34 (Android 14)
* Java Version: 17
* Kotlin Version: 1.9.0

### Dependencias Principales:

* Firebase BOM: 32.7.4
* AndroidX Core KTX
* Material Design Components
* Picasso: 2.8 (para carga de imágenes)
* Navigation Components
* SwipeRefreshLayout: 1.1.0

# 4. Estructura del Proyecto

La aplicación está organizada en los siguientes paquetes principales:

**adapters:** Contiene los adaptadores para RecyclerViews

**model:** Clases de datos y modelos de negocio

**ui:** Fragmentos y ViewModels organizados por funcionalidad:

**home**

**profile**

**search**

**sell**

**detail**

# 5. Funcionalidades Principales

### Gestión de Usuarios:

* Registro de usuarios
* Inicio de sesión
* Gestión de perfil con foto
* Cierre de sesión
* Gestión de Productos:

Publicación de productos

Categorización (Running, Ropa, Suplementación, etc.)

Búsqueda y filtrado

Visualización detallada

Sistema de Compra-Venta:

Listado de productos

Proceso de compra

Historial de compras

Gestión de productos publicados

# 6. Uso de Firebase

La aplicación utiliza los siguientes servicios de Firebase:

### Authentication:

* Gestión de registro y autenticación de usuarios
* Manejo de sesiones

### Cloud Firestore:

* Almacenamiento de datos de usuarios
* Gestión de productos
* Registro de transacciones

### Storage:

* Almacenamiento de imágenes de perfil
* Almacenamiento de imágenes de productos

### Security Rules:

* Reglas de seguridad para productos
* Protección de datos de usuarios
* Control de acceso a compras

# 7. Pantallas y Flujos de Usuario

**Pantalla de Inicio:**

* Listado de productos disponibles
* Filtrado por categorías
* Búsqueda de productos

**Perfil de Usuario:**

* Información personal
* Foto de perfil
* Historial de compras
* Productos publicados
* Publicación de Productos:
* Formulario de creación
* Selección de categoría
* Carga de imágenes
* Establecimiento de precio

**Detalle de Producto:**

* Información completa
* Opción de compra
* Contacto con vendedor

# 8. Conclusiones

FutTFG es una aplicación robusta y bien estructurada que implementa buenas prácticas de desarrollo Android moderno:

* Utiliza arquitectura MVVM para una clara separación de responsabilidades
* Implementa Firebase como backend, proporcionando una solución escalable y segura
* Incluye características de seguridad y validación de usuarios
* Ofrece una experiencia de usuario fluida y moderna
* Implementa manejo de errores y estados de carga
* Utiliza componentes de Material Design para una interfaz consistente
* Sigue las guías de diseño de Android y las mejores prácticas de desarrollo
* La aplicación está preparada para escalar y puede ser extendida con nuevas funcionalidades manteniendo su estructura actual.