

SmartMeal

2º DAM

Anteproyecto TFG

Autor: Hugo González Cuadrado

Fecha: 21/10/2025

Tabla de contenido

- [SmartMeal](#)
 - [2º DAM](#)
 - [Anteproyecto TFG](#)
 - [Tabla de contenido](#)
 - [Definición del problema](#)
 - [Definición de la solución](#)
 - [Planteamiento](#)
 - [Objetivos](#)
 - [Objetivos Específicos](#)
 - [Procedimiento](#)
 - [Entornos de Desarrollo](#)
 - [Lenguajes y Tecnologías](#)
 - [Bases de Datos](#)
 - [Sistemas Operativos Compatibles](#)
 - [Tipo de Dispositivo](#)
 - [Usabilidad de los Diseños Elegidos](#)
-

Definición del problema

En la actualidad, muchas personas buscan mejorar su alimentación y optimizar su tiempo al planificar comidas semanales. Sin embargo, la mayoría de las aplicaciones de cocina disponibles ofrecen recetas de manera genérica, sin adaptarse a las necesidades específicas del usuario, como alergias, preferencias alimenticias, objetivos calóricos o ingredientes disponibles en casa.

Entre las principales dificultades que enfrentan los usuarios destacan:

- Falta de una herramienta que **planifique menús personalizados** de forma automática.
- Dificultad para **generar listas de compra sincronizadas** con los menús.
- Escasa integración con **funciones inteligentes**, como reconocimiento por voz o escaneo de productos.
- Limitadas opciones de sincronización **entre dispositivos o modo offline**.

La ausencia de una aplicación centralizada y adaptativa provoca pérdida de tiempo, poca organización alimentaria y dificultad para mantener una dieta equilibrada. Por tanto, existe la necesidad de una solución tecnológica que simplifique y personalice la gestión alimentaria diaria del usuario.

Definición de la solución

La solución propuesta es **SmartMeal**, una aplicación móvil multiplataforma desarrollada con **Flutter** que permite a los usuarios generar menús semanales equilibrados y automatizar su lista de la compra, integrando funciones de reconocimiento inteligente de productos y almacenamiento en la nube mediante **Firestore**.

Con **SmartMeal**, el usuario podrá:

- Generar menús automáticos adaptados a sus preferencias, alergias y necesidades calóricas.
- Crear listas de la compra sincronizadas con sus menús.
- Escanear alimentos mediante **ML Kit (OCR)** o registrar ingredientes por voz.
- Sincronizar y guardar toda la información en **Firestore**, incluso en modo offline.
- Disfrutar de una interfaz moderna y accesible basada en **Material Design 3**.

En conjunto, SmartMeal ofrecerá una experiencia completa e inteligente para gestionar la alimentación diaria de manera simple, eficiente y personalizada.

Planteamiento

Fase	Descripción
Análisis de requisitos del usuario	Identificación de las necesidades de los usuarios en la planificación de comidas. Definición de las funcionalidades clave: menús inteligentes, lista de compra, reconocimiento por voz e imagen.
Análisis del problema	Estudio de las limitaciones de las apps existentes y de cómo optimizar la experiencia del usuario mediante IA ligera y Firestore.
Diseño	Creación de la arquitectura MVVM , diseño de interfaces adaptadas a Material Design 3 y definición del modelo de datos en Firestore .
Implementación	Desarrollo del frontend con Flutter (Dart) , integración de la lógica nativa con Kotlin y conexión con Firestore y Auth .
Pruebas	Validación de la generación de menús, sincronización en la nube, funcionamiento offline y reconocimiento inteligente.
Despliegue	Generación de APK y documentación final del proyecto para exposición y evaluación del TFG.

Objetivos

El objetivo principal de este proyecto es desarrollar una aplicación móvil multiplataforma que facilite la planificación alimentaria de los usuarios, automatizando la generación de menús, listas de compra y recomendaciones personalizadas mediante inteligencia artificial ligera.

Objetivos Específicos

- **Automatizar la planificación de menús semanales** adaptados a las preferencias del usuario.

- **Integrar reconocimiento de voz e imagen (OCR)** mediante **Google ML Kit**.
 - **Sincronizar los datos del usuario en la nube** usando **Firebase Firestore**, con soporte offline.
 - **Implementar autenticación** con **Firebase Auth** (correo y Google).
 - **Diseñar una interfaz moderna e intuitiva** basada en **Material Design 3**.
 - **Aplicar el patrón arquitectónico MVVM** para mantener una estructura limpia y escalable.
 - **Demostrar la integración de IA ligera con TensorFlow Lite**, para ofrecer sugerencias de menús personalizadas.
-

Procedimiento

Entornos de Desarrollo

- **Android Studio:** entorno principal para la implementación de Flutter y Kotlin.
- **Visual Studio Code:** edición ligera para desarrollo y depuración rápida.
- **Git + GitHub:** control de versiones y gestión del repositorio del TFG.

Lenguajes y Tecnologías

- **Frontend:** Flutter (Dart).
- **Backend y Lógica Nativa:** Kotlin.
- **Base de Datos:** Firebase Firestore (NoSQL).
- **Autenticación:** Firebase Auth.
- **IA Ligera:** ML Kit y TensorFlow Lite.
- **Reconocimiento de voz:** SpeechRecognizer (Android).

Bases de Datos

- **Firebase Firestore (NoSQL):** Almacenamiento y sincronización de datos (usuarios, menús, recetas, lista de compra).
- **Firebase Storage:** para imágenes de recetas y productos.

Sistemas Operativos Compatibles

- **Android** (principal destino).
- **Windows / Linux / macOS** (entornos de desarrollo y prueba).

Tipo de Dispositivo

- Smartphones y tabletas Android.

Usabilidad de los Diseños Elegidos

- Interfaz basada en **Material Design 3**, que proporciona:
 - Colores dinámicos y componentes modernos (NavigationBar, Cards, FABs).
 - Diseño accesible, adaptable y uniforme.
 - Experiencia fluida en dispositivos móviles.

Ejemplo de configuración en Flutter:

```
theme: ThemeData(  
  colorSchemeSeed: Colors.green,  
  useMaterial3: true,  
)
```