



UNIÓN EUROPEA  
Fondo Social Europeo  
El FSE invierte en tu futuro



# PROYECTO DE DESARROLLO DE APLICACIONES MULTIPLATAFORMA GESTIÓN DE CONSULTAS MÉDICAS

CICLO FORMATIVO DE GRADO SUPERIOR  
**DESARROLLO DE APLICACIONES MULTIPLATAFORMA (IFCS02)**

CURSO 2024-2025

Autor/a/es:

**Maximiliano Benjamín Ramos Aguirre**

Tutor/a:

**Federico Banda Sierra**



DEPARTAMENTO DE INFORMÁTICA Y COMUNICACIONES  
**IES LUIS VIVES**

## Resumen

**Gestión de Consultas Médicas** es un proyecto orientado a facilitar la gestión de citas médicas en centros de salud mediante una solución multiplataforma. Se compone de una aplicación móvil para pacientes (desarrollada en Kotlin para Android), un portal web para la gestión de citas por parte del personal sanitario (en React.js) y una API RESTful (en ASP.NET Core) que conecta ambos entornos. Este sistema permite reservar, modificar y cancelar consultas médicas, así como gestionar usuarios y agendas médicas. El objetivo es mejorar la experiencia del paciente, optimizar los recursos del centro médico y contribuir a la digitalización del sector sanitario.

## Abstract

**Medical Appointment Management** is a cross-platform project aimed at improving appointment scheduling in healthcare centers. It includes an Android mobile app for patients (developed in Kotlin), a web portal for healthcare staff (developed in React.js), and a RESTful API (built with ASP.NET Core) that integrates both platforms. The system allows users to book, modify, and cancel appointments, and enables doctors to manage their schedules. The project seeks to enhance patient experience, optimize resources, and support the digital transformation of the healthcare industry.

## Índice

<b>1. INTRODUCCIÓN .....</b>	<b>3</b>
<b>1.1. OBJETIVO.....</b>	<b>3</b>
<b>1.2. ALCANCE .....</b>	<b>3</b>
<b>1.3. JUSTIFICACIÓN.....</b>	<b>3</b>
<b>2. IMPLEMENTACIÓN.....</b>	<b>4</b>
<b>2.1. ANÁLISIS DE LA APLICACIÓN .....</b>	<b>4</b>
2.1.1. REQUISITOS FUNCIONALES DE LA APLICACIÓN .....	4
2.1.2. ANÁLISIS Y SELECCIÓN DE LAS TECNOLOGÍAS.....	4
2.1.3. PLANIFICACIÓN DE LA REALIZACIÓN DEL PROYECTO .....	4
2.1.4. OTROS.....	5
<b>2.2. DISEÑO.....</b>	<b>6</b>
<b>2.3. IMPLEMENTACIÓN.....</b>	<b>6</b>
<b>2.4. IMPLANTACIÓN .....</b>	<b>6</b>
<b>2.5. DOCUMENTACIÓN .....</b>	<b>7</b>
<b>3. CONCLUSIONES .....</b>	<b>7</b>
<b>3.1. RESULTADOS Y DISCUSION .....</b>	<b>7</b>
<b>3.2. TRABAJO FUTURO (OPCIONAL).....</b>	<b>7</b>
<b>4. BIBLIOGRAFÍA.....</b>	<b>7</b>
<b>ANEXOS.....</b>	<b>7</b>

## 1. INTRODUCCIÓN

Este proyecto se ha desarrollado como parte de la formación del CFGS en Desarrollo de Aplicaciones Multiplataforma. Su finalidad es mejorar la organización de citas médicas a través de soluciones digitales accesibles desde distintos dispositivos.

**OBJETIVO**

Lo que pretendéis conseguir con el proyecto, conocimientos que vais a adquirir o en los que vais a profundizar...

### 1.1. ALCANCE

El proyecto se delimitará en las siguientes áreas y funcionalidades, considerando un desarrollo que sea alcanzable en un período de 3 meses:

#### **Funcionalidades Básicas (Obligatorias)**

- **Backend (API RESTful en ASP.NET Core):**
  - Gestión de usuarios (pacientes, médicos, administradores).
  - CRUD de citas médicas y gestión de horarios.
  - Autenticación y autorización (mediante JWT).
  - Conexión a una base de datos para el almacenamiento seguro de datos.
- **Aplicación Móvil (Android con Kotlin):**
  - Registro e inicio de sesión para pacientes.
  - Visualización de disponibilidad de citas.
- **Portal Web de Administración:**
  - Panel de control para la gestión de consultas y agendas médicas.
  - Módulo de administración para registrar y actualizar datos de pacientes y profesionales.
  - Funcionalidades CRUD para servicios y citas.
  - Creación de archivos PDF de citas.

### 1.2. JUSTIFICACIÓN

En el contexto actual, los sistemas de salud presentan carencias en la gestión digital de citas médicas, generando largas esperas y problemas de comunicación entre pacientes y profesionales. Este proyecto propone una solución ágil, intuitiva y moderna que permite:

- Aumentar la calidad del servicio médico.
- Reducir la carga administrativa.
- Mejorar la experiencia del usuario.
- Facilitar la digitalización del sector salud.

A nivel internacional ya existen soluciones similares que han tenido un impacto positivo. Este proyecto se inspira en estas iniciativas, adaptándolas al entorno local con tecnologías modernas.

## 2. IMPLEMENTACIÓN

### 2.1. ANÁLISIS DE LA APLICACIÓN

La aplicación permitirá a los pacientes gestionar sus citas de forma sencilla y a los profesionales médicos visualizar y organizar sus agendas.

#### 2.1.1. Requisitos funcionales de la aplicación

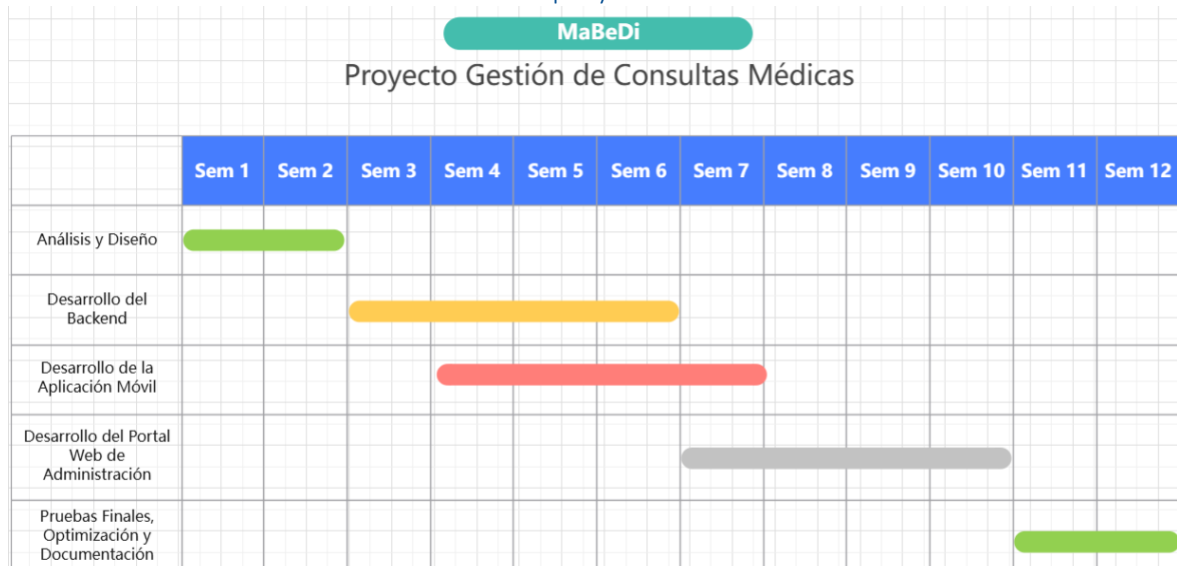
- Registro e inicio de sesión de usuarios.
- Visualización de disponibilidad y horarios.
- Reserva, modificación y cancelación de citas.
- Gestión de usuarios, citas y agendas desde el portal web.
- Notificaciones básicas a los pacientes.

#### 2.1.2. Análisis y selección de las tecnologías

Se han valorado varias tecnologías y se han elegido las siguientes:

- **Backend:** ASP.NET Core (robustez, integración con Entity Framework Core, buen soporte para APIs REST).
- **Base de datos:** PostgreSQL (compatibilidad y fiabilidad).
- **Frontend Web:** React.js (popularidad, rendimiento, componentes reutilizables).
- **App Móvil:** Kotlin (idioma oficial Android, moderno, seguro).
- **Despliegue:** Vercel y Render.

#### 2.1.3. Planificación de la realización del proyecto



#### 2.1.4. Otros

### Presupuesto del Proyecto

Este presupuesto estima los costes necesarios para el desarrollo completo del proyecto durante un período de 12 semanas, considerando recursos humanos, infraestructura tecnológica y otros gastos relacionados.

#### 1. Recursos Humanos

Actividad	Horas estimadas	Tarifa por hora (€)	Coste (€)
Análisis y diseño	40	25	1,000
Desarrollo Backend	80	25	2,000
Desarrollo Frontend	60	25	1,500
Web			
Desarrollo App	60	25	1,500
Móvil			
Testing y corrección	30	20	600
Documentación y entrega	20	20	400
Total Recursos Humanos	290		7,000

#### 2. Infraestructura y Servicios

Concepto	Coste mensual (€)	Meses	Coste total (€)
Hosting backend (Render)	10	3	30
Hosting web (Vercel)	0	3	0
Base de datos PostgreSQL	15	3	45
Servicio email SMTP	5	3	15
Licencias software	0	0	0
Total Infraestructura			90

#### 3. Hardware y Otros Gastos

Concepto	Coste (€)
Amortización PC de desarrollo	125
Materiales y papelería	50
Total Hardware y Otros	175

#### 4. Resumen Final del Presupuesto

Categoría	Coste (€)
Recursos Humanos	7,000
Infraestructura y Servicios	90
Hardware y Otros	175
TOTAL GENERAL	7,265

- Notas

- Las tarifas por hora se han ajustado a niveles de un desarrollador junior o becario.
- Se han utilizado servicios con planes gratuitos o de bajo coste para minimizar gastos.
- El coste humano está valorado para dar una idea profesional, aunque en un TFG es trabajo personal no remunerado.
- Este presupuesto es una aproximación para un desarrollo profesional realista en el tiempo estipulado.

## 2.2. DISEÑO

También tendrá sus propios apartados. A modo de ejemplo podremos servirnos de:

- Prototipado: Diseño de pantallas (no el resultado final), bocetos, Mockups

Mockup of a web interface for a doctor's role. The header shows "Rol: Doctor" and a "Cerrar Sesión" button. Below is a navigation bar with buttons for "Tabla Pacientes", "Formulario Pacientes", "Tabla Doctores", "Formulario Doctores", "Tabla Citas", and "Formula". The main content area is a table with headers: "Nombre", "Telefono", "Correo", "Fecha Nacimiento", and "Dni".

- Diagramas E/R o modelo de datos
- Diagramas de clase genérico (Atributos y métodos principales)
- Casos de uso (El desarrollo de cada caso de uso en anexos)
- Diagramas de flujo de datos: Pueden complementar aquellos casos de uso complejos.
- Base de datos. Estructura de tablas a utilizar, soporte lógico y físico.
- Arquitectura: Tecnologías utilizadas en cada parte del proyecto

## 2.3. IMPLEMENTACIÓN

Se desarrollará una API RESTful en ASP.NET Core con controladores para cada entidad.

La app móvil se conectará al backend mediante llamadas HTTP usando Retrofit.

El portal web utilizará Axios o Fetch para la comunicación con la API.

## 2.4. IMPLANTACIÓN

El despliegue inicial se realizará localmente y, posteriormente, en servidores de Vercel (Portal Web) y Render (Api Service + Base de datos).

El APK será instalado directamente en dispositivos Android de prueba.

## 2.5. DOCUMENTACIÓN

Se entregará:

- Manual técnico (estructura, API endpoints, esquemas de BD).
- Manual de usuario para el portal web y la app.
- Documentación de pruebas y casos de uso.

## 3. CONCLUSIONES

Este proyecto ha permitido aplicar de forma real los conocimientos adquiridos a lo largo del ciclo formativo, desarrollando una solución útil para un problema real. Se ha trabajado en equipo con planificación, adaptando los conocimientos técnicos a necesidades concretas y desarrollando habilidades en backend, frontend y despliegue en la nube.

### 3.1. RESULTADOS Y DISCUSIÓN

A lo largo del desarrollo se han superado diversos retos técnicos, especialmente en la sincronización de agendas y gestión de estados entre frontend y backend. La planificación inicial se ha cumplido en su mayor parte, con pequeños ajustes derivados de la complejidad real en fases intermedias.

### 3.2. TRABAJO FUTURO (OPCIONAL)

Hasta donde se ha llegado y posibles ampliaciones del proyecto (continuación del alcance).

## 4. BIBLIOGRAFÍA

- Microsoft Docs – ASP.NET Core: <https://learn.microsoft.com/aspnet/core>
- Android Developers – Kotlin: <https://developer.android.com/kotlin>
- React Official Website: <https://react.dev>
- PostgreSQL Documentation: <https://www.postgresql.org/docs/>
- Azure Architecture Center: <https://learn.microsoft.com/azure/architecture>
- Figma Design Tool: <https://www.figma.com>

## ANEXOS

Detalle del código y otra documentación que consideréis oportuna.

- I. Diagrama de clases
- II. Modelo E/R
- III. Prototipos de interfaz
- IV. Código fuente relevante