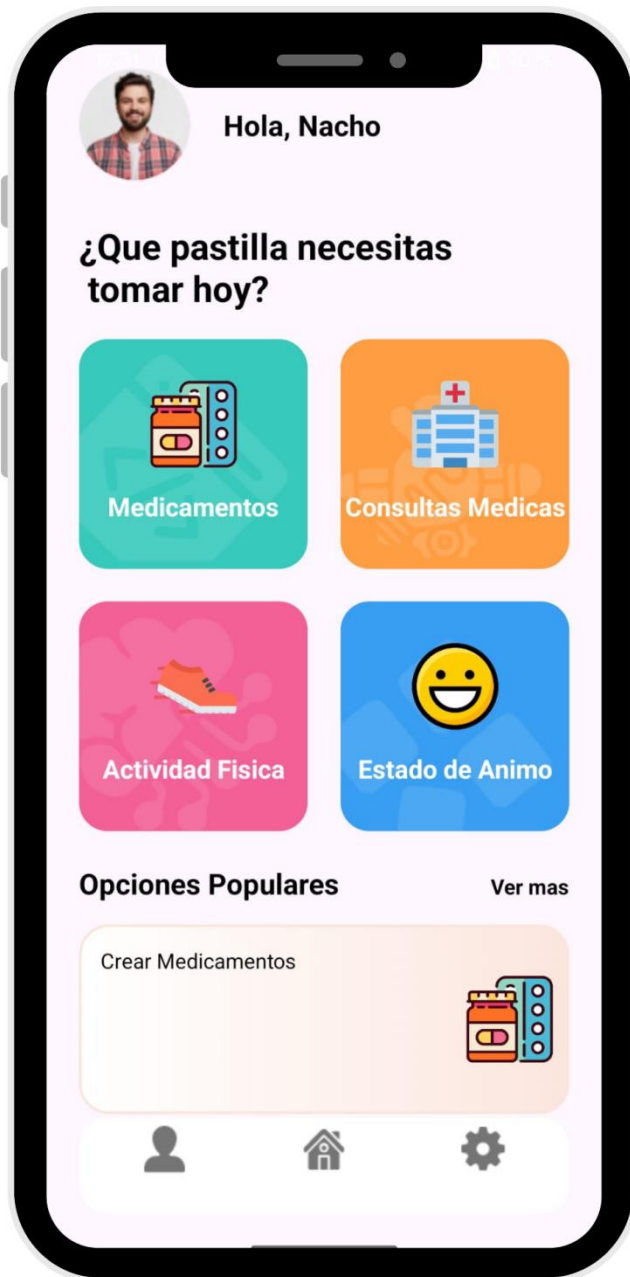


# Manual de Despliegue

## MedMonitorApp



1. Introducción .....	3
2. Requisitos Previos .....	3
2.1 Hardware.....	3
2.2 Software .....	3
3. Despliegue Paso a Paso .....	4
3.1 Clonar el Repositorio .....	4
3.2 Configuración de la Base de Datos.....	4
Paso 1: Instalar MySQL.....	4
Paso 2: Crear la Base de Datos .....	4
Paso 3: (Opcional) Cargar Datos Iniciales .....	4
3.3 Configuración y Despliegue de la API .....	5
Paso 1: Configurar la Conexión a la Base de Datos .....	5
Paso 2: Compilar la API .....	5
Paso 3: Ejecutar la API.....	5
3.4 Despliegue de la Aplicación Móvil .....	5
Paso 1: Abrir el Proyecto.....	5
Paso 2: Configurar la API .....	5
Paso 3: Ejecutar la App.....	6
3.5 Despliegue de la Aplicación para Windows.....	6
Paso 1: Abrir el Proyecto.....	6
Paso 2: Configurar la Conexión .....	6
Paso 3: Compilar y Ejecutar .....	6
4. Pruebas Finales.....	6
5. Solución de Problemas .....	7

# 1. Introducción

MedMonitorApp es un sistema compuesto por:

1. Una base de datos MySQL para almacenar información.
2. Una API desarrollada en Java que conecta la base de datos con las aplicaciones.
3. Una aplicación móvil en Kotlin.
4. Una aplicación de escritorio en C# para gestionar la información desde Windows.

Este manual detalla cómo instalar y configurar cada componente del sistema.

## 2. Requisitos Previos

### 2.1 Hardware

- **Ordenador:**
  - Procesador: Intel i5 o superior / AMD Ryzen 5 o superior.
  - RAM: Mínimo 8 GB.
  - Disco duro: 20 GB de espacio libre.
- **Dispositivo móvil Android:**
  - Android 8.0 (Oreo) o superior.
  - Conectividad USB o Wi-Fi para pruebas.

### 2.2 Software

Descarga e instala los siguientes programas antes de comenzar:

1. **Java Development Kit (JDK)** versión 17 o superior: [Descargar aquí](#).
2. **MySQL Community Server**: [Descargar aquí](#).
3. **Android Studio**: [Descargar aquí](#).
4. **Visual Studio 2022**: [Descargar aquí](#).
5. **Git**: [Descargar aquí](#).
6. **Postman** (opcional, para probar la API): [Descargar aquí](#).

## 3. Despliegue Paso a Paso

### 3.1 Clonar el Repositorio

1. **Abrir Git Bash o una consola:**
  - a. En Windows: Usa CMD, PowerShell o Git Bash.
  - b. En Mac/Linux: Usa Terminal.
2. **Clonar el repositorio:** `git clone https://github.com/Ignaciosgh/TFG.git`
3. **Acceder al directorio del proyecto:** `cd TFG`

### 3.2 Configuración de la Base de Datos

#### *Paso 1: Instalar MySQL*

1. Descarga e instala MySQL Server desde el enlace proporcionado.
2. Durante la instalación:
  - a. **Tipo de configuración:** Elige "Developer Default".
  - b. **Usuario administrador:** Usa root.
  - c. **Contraseña:** Elige una contraseña segura y recuérdala.

#### *Paso 2: Crear la Base de Datos*

1. **Abrir MySQL Workbench** o la línea de comandos de MySQL.
2. Ejecutar el script SQL para crear la base de datos:
  - a. Ve al archivo `database/schema.sql` en el repositorio clonado.
  - b. Copia y pega el contenido en MySQL Workbench o usa el siguiente comando: `mysql -u root -p < ruta/completa/a/schema.sql`

#### *Paso 3: (Opcional) Cargar Datos Iniciales*

- Si el proyecto incluye un archivo `seed.sql`, ejecútalo de la misma manera:  
`mysql -u root -p < ruta/completa/a/seed.sql`

### 3.3 Configuración y Despliegue de la API

#### ***Paso 1: Configurar la Conexión a la Base de Datos***

1. Ve al archivo `application.properties` en `api/src/main/resources/`.
2. Modifica las credenciales de la base de datos:  
`spring.datasource.url=jdbc:mysql://localhost:3306/medmonitor`  
`spring.datasource.username=root`  
`spring.datasource.password=tu_contraseña`

#### ***Paso 2: Compilar la API***

1. Abre una terminal.
2. Ve a la carpeta `api/`: `cd api/`
3. Ejecuta el siguiente comando: `./mvnw clean package`  
Esto generará un archivo `.jar` en la carpeta `target/`.

#### ***Paso 3: Ejecutar la API***

1. Inicia la API con: `java -jar target/api.jar`
2. Abre un navegador y verifica que está corriendo:
  - a. URL: <http://localhost:8080>.

### 3.4 Despliegue de la Aplicación Móvil

#### ***Paso 1: Abrir el Proyecto***


1. Abre Android Studio.
2. Selecciona "Open an Existing Project" y navega a `mobile/`.

#### ***Paso 2: Configurar la API***

1. Abre el archivo `ApiService.kt`: `const val BASE_URL = "http://<tu-ip>:8080"`
  - a. Cambia `<tu-ip>` por `localhost` si usas un emulador.

- b. Si usas un dispositivo físico, obtén la IP de tu ordenador con:
  - i. Windows: `ipconfig` (buscar "Dirección IPv4").
  - ii. Mac/Linux: `ifconfig` o `ip a`.

### ***Paso 3: Ejecutar la App***

1. Conecta un dispositivo Android o inicia un emulador.
2. Haz clic en **Run** (ícono de  en Android Studio).
3. La aplicación se instalará y abrirá.

## **3.5 Despliegue de la Aplicación para Windows**

### ***Paso 1: Abrir el Proyecto***

1. Abre Visual Studio.
2. Selecciona "Open a Project or Solution" y navega a `windows/`.

### ***Paso 2: Configurar la Conexión***

1. Abre `App.config`.
2. Cambia la URL de la API: `<appSettings>`

```
<add key="ApiUrl" value="http://<tu-ip>:8080"/>
</appSettings>
```

### ***Paso 3: Compilar y Ejecutar***

1. Haz clic en `Build > Build Solution`.
2. Ejecuta la aplicación con `Debug > Start Debugging`.

## **4. Pruebas Finales**

1. **Base de Datos:**
  - a. Verifica que las tablas están creadas.
  - b. Prueba insertar y consultar datos.
2. **API:**
  - a. Usa Postman para probar endpoints como `/medications` o `/medications/create`.
3. **App Móvil y Windows:**

- a. Prueba añadir un medicamento y verifica que aparece en la base de datos.
- b. Asegúrate de que las funcionalidades CRUD funcionan correctamente.

## 5. Solución de Problemas

Problema	Causa Posible	Solución
<b>Error: "Database connection failed"</b>	La base de datos no está corriendo o las credenciales son incorrectas.	Verifica que MySQL está activo y revisa <code>application.properties</code> .
<b>"API no responde"</b>	La API no está corriendo o el puerto está ocupado.	Revisa que el comando <code>java -jar</code> se ejecutó correctamente.
<b>"App no conecta a la API"</b>	El dispositivo móvil y la API no están en la misma red.	Verifica la conexión Wi-Fi o ajusta la URL base en <code>ApiService.kt</code> .