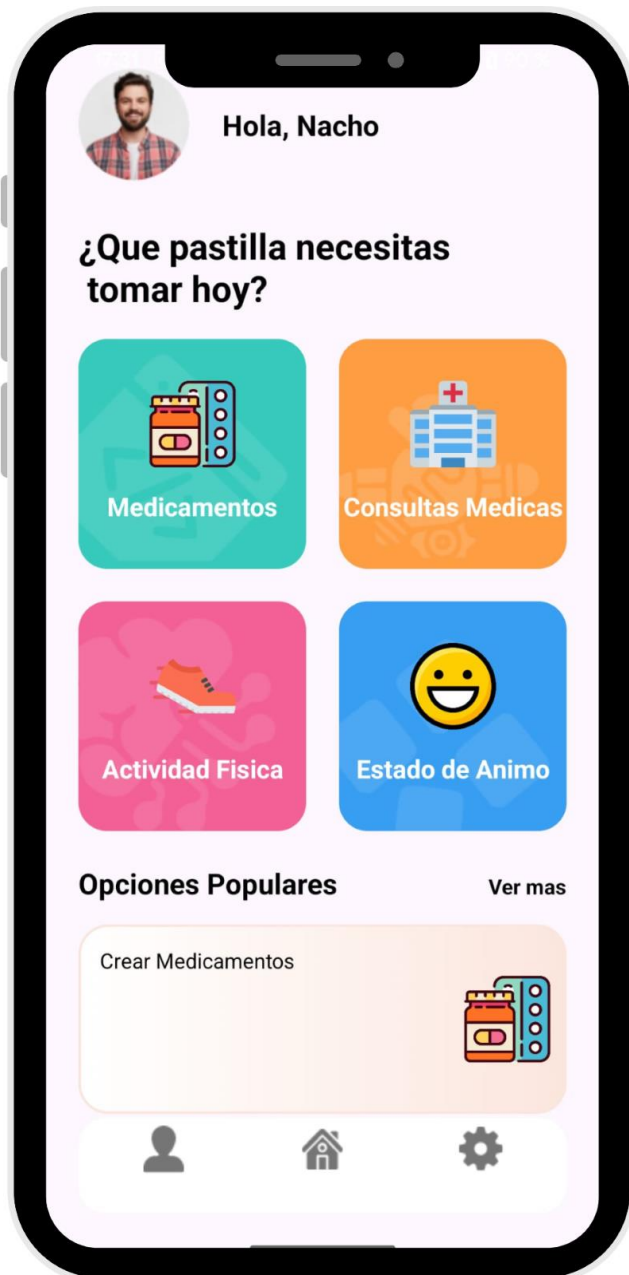


Manual de Despliegue de la API MedMonitorApp



Manual de Configuración y Despliegue de la API de MedMonitorApp	3
Introducción	3
1. Requisitos Previos	4
1.1 Hardware	4
1.2 Software Necesario	4
2. Configuración Inicial	4
2.1 Clonar el Repositorio	4
3. Configuración de la API	4
3.1 Configurar el Acceso a la Base de Datos	4
3.2 Configurar el Puerto del Servidor (Opcional)	5
4. Compilación de la API	5
4.1 Compilación con Maven	5
4.2 Verificar la Compilación	5
5. Ejecución de la API	5
5.1 Ejecutar Localmente	5
5.2 Verificar Endpoints	6
6. Despliegue en Producción (Opcional)	6
6.1 Configuración de un Servidor	6
6.2 Seguridad	6
7. Resolución de Problemas	6
8. Recomendaciones	7

Manual de Configuración y Despliegue de la API de MedMonitorApp

Introducción

La API de **MedMonitorApp** es el componente principal que conecta la base de datos con las aplicaciones móviles y de escritorio. Está desarrollada en **Java** utilizando **Spring Boot**. Este manual guía paso a paso cómo configurarla, compilarla y ejecutarla.

1. Requisitos Previos

1.1 Hardware

- Ordenador con al menos 8 GB de RAM.
- Espacio libre en disco: 2 GB (para dependencias y compilación).

1.2 Software Necesario

1. **Java Development Kit (JDK)** versión 17 o superior.

Descárgalo aquí: <https://www.oracle.com/java/technologies/javase-jdk17-downloads.html>

2. **Maven** (opcional si no usas el wrapper de Maven incluido).
 - a. Descargar: <https://maven.apache.org/download.cgi>
3. **Git** para clonar el repositorio.

Descargar: <https://git-scm.com/>

2. Configuración Inicial

2.1 Clonar el Repositorio

1. Abre una terminal o consola de comandos.
2. Ejecuta: `git clone https://github.com/tu-usuario/medmonitor-repo.git`
3. Accede al directorio de la API: `cd medmonitor-repo/api`

3. Configuración de la API

3.1 Configurar el Acceso a la Base de Datos

1. Abre el archivo `application.properties` en la carpeta `src/main/resources/`.
2. Modifica las credenciales para conectarte a tu base de datos:
`spring.datasource.url=jdbc:mysql://localhost:3306/medmonitor`
`spring.datasource.username=root`

`spring.datasource.password=tu_contraseña`

3. Verifica que el nombre de la base de datos (medmonitor) coincide con el configurado en el servidor MySQL.

3.2 Configurar el Puerto del Servidor (Opcional)

- Por defecto, la API usa el puerto 8080. Si necesitas cambiarlo:
 - En `application.properties`, agrega o edita: `server.port=8081`

4. Compilación de la API

4.1 Compilación con Maven

1. Asegúrate de estar en la carpeta `api/`.
2. Ejecuta el siguiente comando: `./mvnw clean package`
 - a. Esto descargará las dependencias y generará un archivo `.jar` en la carpeta `target/`.

4.2 Verificar la Compilación

1. Al finalizar, asegúrate de que se creó el archivo `api.jar` en `target/`.
2. Si hay errores, revisa los mensajes de error en la consola. Comunes:
 - a. Dependencias no resueltas: Asegúrate de tener conexión a internet.
 - b. Problemas con la versión de Java: Confirma que estás usando JDK 17 o superior.

5. Ejecución de la API

5.1 Ejecutar Localmente

1. En la terminal, ejecuta: `java -jar target/api.jar`
2. Verifica que la API está corriendo:
 - a. Abre un navegador y accede a: <http://localhost:8080>

- b. Deberías ver un mensaje de bienvenida o el endpoint raíz.

5.2 Verificar Endpoints

- Usa herramientas como **Postman** o **curl** para probar los endpoints:
 - **Login:** POST <http://localhost:8080/auth/login>
 - **Consultar medicamentos:** GET <http://localhost:8080/medications>

6. Despliegue en Producción (Opcional)

6.1 Configuración de un Servidor

- Si despliegas la API en un servidor (Linux/Windows):
 - Copia el archivo `api.jar` al servidor.
 - Ejecuta el comando: `nohup java -jar api.jar &`
 - Configura un dominio o subdominio apuntando al servidor (opcional).

6.2 Seguridad

- **Variables de entorno:** No almacenes contraseñas en `application.properties`. Usa:
`spring.datasource.password=${DB_PASSWORD}`
- **Firewall:** Asegúrate de permitir el puerto en el servidor.

7. Resolución de Problemas

Problema	Causa	Solución
Error: "Access denied for user 'root'@"	Credenciales incorrectas para la base de datos.	Verifica el archivo <code>application.properties</code> .
Error: "Port already in use"	El puerto 8080 está siendo usado por otro programa.	Cambia el puerto en <code>application.properties</code> (por ejemplo, a 8081).

Error: "Connection refused to database"	La base de datos no está corriendo o las credenciales son incorrectas.	Verifica que el servidor MySQL está corriendo y que los datos son correctos.
No carga dependencias en Maven	Problemas de conexión a internet.	Asegúrate de tener conexión o verifica el archivo pom.xml.

8. Recomendaciones

1. Pruebas regulares:

- Usa herramientas como Postman para verificar que los endpoints están funcionando.

2. Logs:

- Habilita logs detallados en producción para identificar problemas rápidamente. `logging.level.org.springframework=DEBUG`

3. Documentación de la API:

- Utiliza Swagger o Postman para documentar los endpoints y facilitar su uso.