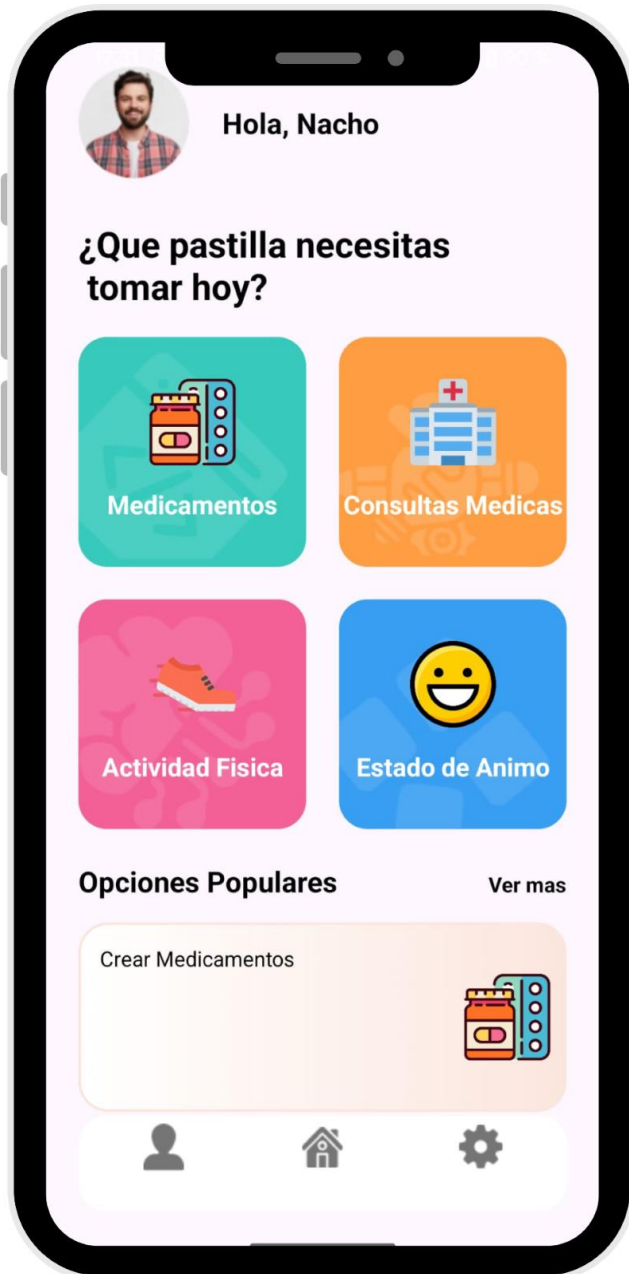


Manual de Despliegue de la Aplicación Android MedMonitorApp



Manual de Configuración y Despliegue de la App Android de MedMonitorApp	4
Introducción	4
1. Requisitos Previos	4
1.1 Hardware	4
1.2 Software	5
2. Configuración Inicial	5
2.1 Clonar el Repositorio	5
2.2 Configuración en Android Studio	5
2.3 Configuración de la URL de la API	5

3. Ejecución de la Aplicación	6
3.1 Selección del Dispositivo de Prueba	6
3.2 Compilar y Ejecutar	6
4. Pruebas de la Aplicación.....	6
4.1 Funcionalidades Clave a Probar	6
5. Resolución de Problemas Comunes	7
6. Configuración Avanzada (Opcional)	7
6.1 Firmar la Aplicación para Producción.....	7
6.2 Configurar ProGuard para Optimización.....	7
7. Verificación Final.....	8
8. Recomendaciones.....	8

Manual de Configuración y Despliegue de la App Android de MedMonitorApp

Introducción

La aplicación Android de **MedMonitorApp** está desarrollada en **Kotlin** y diseñada para interactuar con la API y la base de datos de la aplicación. Este manual guía al desarrollador paso a paso en la configuración, compilación, ejecución y pruebas de la aplicación en dispositivos físicos o emuladores.

1. Requisitos Previos

1.1 Hardware

- **Ordenador:**
 - Procesador: Intel i5/AMD Ryzen 5 o superior.
 - Memoria RAM: Mínimo 8 GB.
 - Espacio libre en disco: Al menos 10 GB.
- **Dispositivo Android (opcional):**
 - Sistema operativo: Android 8.0 (Oreo) o superior.
 - Conexión USB o Wi-Fi.

1.2 Software

1. **Android Studio:** [Descargar aquí](#).
 - a. Versión recomendada: Electric Eel o superior.
2. **Java Development Kit (JDK):** Versión 17 o superior.
3. **Git:** [Descargar aquí](#).
4. **SDK de Android:**
 - a. Niveles de API: 21 (Lollipop) a 33 (Android 13).

2. Configuración Inicial

2.1 Clonar el Repositorio

1. Abre una terminal o consola.
2. Ejecuta el siguiente comando: `git clone https://github.com/Ignaciosgh/TFG.git`
3. Ve al directorio del proyecto: `cd TFG/aplicacion movil`

2.2 Configuración en Android Studio

1. Abre **Android Studio**.
2. Selecciona **"Open an Existing Project"**.
3. Navega a la carpeta `mobile/` dentro del repositorio y haz clic en **OK**.
4. Android Studio descargará automáticamente las dependencias necesarias. Si no sucede automáticamente:
 - a. Haz clic en **File > Sync Project with Gradle Files**.

2.3 Configuración de la URL de la API

1. Abre el archivo `ApiService.kt` en la carpeta `network/`.
2. Busca la constante `BASE_URL`: `const val BASE_URL = "http://<tu-ip>:8080"`
3. Cambia `<tu-ip>` por:
 - a. `localhost` si usas un emulador.
 - b. La IP de tu ordenador si pruebas en un dispositivo físico.
 - i. En Windows: Ejecuta `ipconfig` y busca "Dirección IPv4".


- ii. En Mac/Linux: Ejecuta `ifconfig` o `ip a`.

3. Ejecución de la Aplicación

3.1 Selección del Dispositivo de Prueba

1. **Usar un Emulador:**
 - a. Ve a **Tools > Device Manager**.
 - b. Crea un nuevo emulador con las siguientes configuraciones recomendadas:
 - i. Dispositivo: Pixel 5 o similar.
 - ii. Nivel de API: 30 (Android 11).
 - c. Inicia el emulador.
2. **Usar un Dispositivo Físico:**
 - a. Habilita la **Depuración USB** en tu dispositivo:
 - i. Ve a **Configuración > Opciones de Desarrollador > Depuración USB**.
 - b. Conecta tu dispositivo al ordenador con un cable USB.

3.2 Compilar y Ejecutar

1. Haz clic en el botón **Run** (icono de ) en la barra superior de Android Studio.
2. Selecciona el dispositivo/emulador en la ventana emergente.
3. Espera a que la aplicación se compile y se instale en el dispositivo/emulador.
4. Verifica que la app se abre automáticamente después de la instalación.

4. Pruebas de la Aplicación

4.1 Funcionalidades Clave a Probar

1. **Inicio de Sesión y Registro:**
 - a. Prueba iniciar sesión con credenciales válidas y no válidas.
 - b. Registra un nuevo usuario y verifica que se guarda en la base de datos.
2. **Gestión de Medicamentos:**
 - a. Añade un medicamento y verifica que aparece en la lista.
 - b. Elimina un medicamento y confirma que desaparece.
3. **Notificaciones:**

- a. Programa un recordatorio y verifica que la notificación aparece a la hora correcta.
4. **Historial Médico:**
 - a. Accede al historial y verifica que los datos se muestran correctamente.

5. Resolución de Problemas Comunes

Problema	Causa Posible	Solución
Error: "API no responde"	La URL en <code>ApiService.kt</code> no es correcta o la API no está corriendo.	Verifica la configuración de la URL y asegúrate de que la API está ejecutándose.
El emulador no inicia	Configuración insuficiente del AVD.	Aumenta la memoria RAM asignada al emulador en Device Manager .
Error: "Gradle sync failed"	Dependencias corruptas o no descargadas.	Haz clic en File > Sync Project with Gradle Files o verifica tu conexión a internet.
Dispositivo físico no detectado	Depuración USB no habilitada o controladores no instalados.	Habilita la depuración USB en el dispositivo y descarga los controladores desde Android Studio.

6. Configuración Avanzada (Opcional)

6.1 Firmar la Aplicación para Producción

1. Ve a **Build > Generate Signed Bundle / APK**.
2. Selecciona **APK** y haz clic en **Next**.
3. Crea un nuevo keystore (archivo `.jks`) o usa uno existente:
 - a. Ubicación: Elige una carpeta en tu ordenador.
 - b. Contraseña: Elige una contraseña segura.
4. Completa los campos y selecciona el modo **release**.
5. Haz clic en **Finish**. El APK firmado estará en la carpeta `app/release`.

6.2 Configurar ProGuard para Optimización

- Habilita ProGuard para reducir el tamaño del APK y proteger el código:

- Abre `build.gradle` del módulo `app`.
- Asegúrate de que las líneas siguientes están habilitadas:
`minifyEnabled true`
`shrinkResources true`
`proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'`

7. Verificación Final

1. **Conexión con la API:**
 - a. Prueba todas las funcionalidades dependientes de la API (inicio de sesión, medicamentos, etc.).
2. **Fluidez de la Interfaz:**
 - a. Asegúrate de que no hay retrasos al navegar entre pantallas.
3. **Compatibilidad:**
 - a. Prueba la aplicación en diferentes versiones de Android y tamaños de pantalla.

8. Recomendaciones

1. **Pruebas en Diferentes Dispositivos:**
 - a. Usa dispositivos físicos y emuladores para verificar el comportamiento en diferentes entornos.
2. **Monitoreo de Errores:**
 - a. Usa herramientas como Firebase Crashlytics para detectar errores en producción.
3. **Documentación de Cambios:**
 - a. Documenta cualquier cambio en la configuración del proyecto para facilitar futuras actualizaciones.