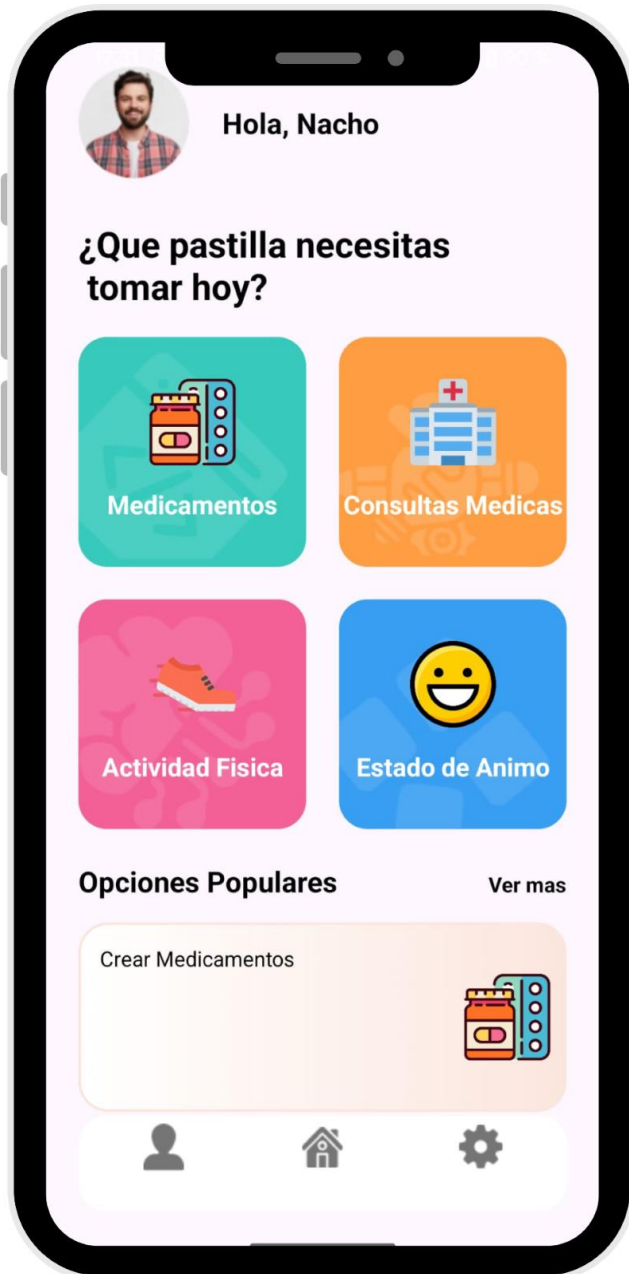


Manual de Despliegue de la Aplicación Windows MedMonitorApp



Manual de Configuración y Despliegue de la Aplicación Windows de MedMonitorApp ..	4
Introducción	4
1. Requisitos Previos	4
1.1 Hardware	4
1.2 Software	4
2. Configuración Inicial.....	5
2.1 Clonar el Repositorio	5
2.2 Abrir el Proyecto en Visual Studio.....	5
3. Configuración de la Conexión a la API	5

4. Compilación del Proyecto	6
4.1 Verificar la Configuración	6
4.2 Compilar.....	6
5. Ejecución de la Aplicación	6
5.1 Ejecutar en Modo Debug	6
5.2 Crear el Ejecutable	6
6. Pruebas de Funcionalidades	7
6.1 Funcionalidades Clave a Probar	7
6.2 Manejo de Errores.....	7
7. Resolución de Problemas	7
8. Despliegue Final.....	8
8.1 Crear el Instalador (Opcional).....	8
8.2 Probar el Instalador	8
9. Recomendaciones.....	8

Manual de Configuración y Despliegue de la Aplicación Windows de MedMonitorApp

Introducción

La aplicación Windows de **MedMonitorApp** está desarrollada en **C#** utilizando **Windows Forms**. Su propósito es ofrecer una interfaz gráfica para gestionar datos y conectarse a la API del sistema. Este manual proporciona instrucciones paso a paso para configurarla, compilarla y ejecutarla.

1. Requisitos Previos

1.1 Hardware

- Ordenador con sistema operativo Windows 10 o superior.
- Procesador: Intel i3/AMD Ryzen 3 o superior.
- RAM: Mínimo 4 GB.
- Espacio libre en disco: 2 GB.

1.2 Software

1. **Visual Studio 2022** (Community, Professional o Enterprise).

Descargar aquí: <https://visualstudio.microsoft.com/>

- a. Durante la instalación, asegúrate de seleccionar la **carga de trabajo para desarrollo de aplicaciones de escritorio en .NET**.
2. **.NET Framework** o **.NET Core Runtime** (según el proyecto).
 - a. Visual Studio lo instala automáticamente con la carga de trabajo mencionada.
 3. **Git** para clonar el repositorio.

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

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. Accede al directorio del proyecto de la aplicación Windows: `cd TFG/aplicacion windows`

2.2 Abrir el Proyecto en Visual Studio

1. Abre **Visual Studio**.
2. Selecciona "**Open a Project or Solution**".
3. Navega a la carpeta windows/ dentro del repositorio y selecciona el archivo de solución (.sln).
4. Visual Studio cargará el proyecto. Si aparece un mensaje para restaurar paquetes NuGet, haz clic en **Restore**.

3. Configuración de la Conexión a la API

1. Abre el archivo de configuración de la aplicación (App.config o su equivalente).
2. Busca la clave ApiUrl y ajusta la URL de la API: `<appSettings>`
`<add key="ApiUrl" value="http://<tu-ip>:8080"/>`
`</appSettings>`
 - a. Cambia <tu-ip> por:
 - i. localhost si la API está en el mismo ordenador.
 - ii. La dirección IP de la máquina donde corre la API, si está en otro equipo.
3. Guarda los cambios.

4. Compilación del Proyecto

4.1 Verificar la Configuración

1. Asegúrate de que la **configuración del proyecto** está en modo Debug para pruebas.
 - a. En la barra superior de Visual Studio, selecciona Debug en lugar de Release.
2. Asegúrate de que la plataforma está configurada correctamente:
 - a. Usa x86 para sistemas de 32 bits o x64 para sistemas de 64 bits.

4.2 Compilar

1. Haz clic en **Build > Build Solution** o presiona Ctrl+Shift+B.
2. Verifica que no haya errores de compilación.
3. Si hay errores:
 - a. Asegúrate de que todas las dependencias están restauradas (menú **Tools > NuGet Package Manager > Manage NuGet Packages**).
 - b. Revisa los mensajes de error en la ventana de salida.

5. Ejecución de la Aplicación

5.1 Ejecutar en Modo Debug

1. Haz clic en **Debug > Start Debugging** o presiona F5.
2. La aplicación se abrirá. Prueba las funcionalidades clave:
 - a. Inicio de sesión.
 - b. Gestión de medicamentos.
 - c. Visualización del historial.

5.2 Crear el Ejecutable

1. Cambia la configuración a Release en la barra superior.
2. Ve a **Build > Build Solution**.
3. El ejecutable (.exe) se generará en la carpeta bin/Release.

6. Pruebas de Funcionalidades

6.1 Funcionalidades Clave a Probar

1. **Conexión con la API:**
 - a. Asegúrate de que los datos mostrados (como medicamentos o usuarios) se sincronizan correctamente con la base de datos a través de la API.
2. **CRUD de Medicamentos y otras Tablas:**
 - a. Crear, leer, actualizar y eliminar medicamentos.
3. **Validación de Datos:**
 - a. Intenta ingresar datos incorrectos o vacíos para verificar los mensajes de error.

6.2 Manejo de Errores

- Asegúrate de que la aplicación muestra mensajes claros si:
 - No puede conectarse a la API.
 - Hay un error en las operaciones CRUD.

7. Resolución de Problemas

Problema	Causa Posible	Solución
Error: "No se puede conectar a la API"	La URL en App.config es incorrecta o la API no está corriendo.	Verifica la URL de la API y que el servidor de la API está ejecutándose.
Error: "No se encuentra ensamblado"	Falta un paquete NuGet necesario.	Restaura los paquetes en Tools > NuGet Package Manager > Restore NuGet Packages .
La aplicación se cierra al iniciar	Excepción no controlada en el código.	Usa el modo Debug para identificar la línea del código que genera el error.
No se genera el archivo .exe	Error durante la compilación.	Revisa los mensajes en la consola de salida y verifica que todas las referencias del proyecto están correctas.

8. Despliegue Final

8.1 Crear el Instalador (Opcional)

1. Usa herramientas como **Inno Setup** o el asistente de instalación de Visual Studio.
2. Configura los archivos necesarios:
 - a. El ejecutable (.exe) generado en bin/Release.
 - b. Archivos adicionales como App.config si es necesario.
3. Genera el instalador.

8.2 Probar el Instalador

- Instala la aplicación en un equipo diferente al de desarrollo para asegurarte de que funciona correctamente.

9. Recomendaciones

1. **Documenta Cambios en la Configuración:**
 - a. Anota cualquier cambio realizado en App.config o en las referencias del proyecto.
2. **Pruebas en Distintos Entornos:**
 - a. Asegúrate de probar la aplicación en versiones de Windows diferentes.
3. **Mantenimiento del Código:**
 - a. Revisa regularmente las dependencias de NuGet para mantenerlas actualizadas.