

Documentación Setup - Panel General Remotos

Resumen del Proyecto

Panel General Remotos es un sistema de dashboard para el seguimiento de llamadas y rendimiento de remotos, con integración a Google Sheets.

Arquitectura

- **Backend:** .NET 8 Web API con SignalR
- **Frontend:** Angular 18 con Material Design
- **Integración:** Google Sheets API
- **Tiempo Real:** SignalR para actualizaciones automáticas

Prerrequisitos Instalados

Herramientas Base

1. **.NET SDK 8.0.403** - Para desarrollo backend
2. **Node.js v20.17.0** - Para desarrollo frontend
3. **npm v10.8.2** - Gestor de paquetes (incluido con Node.js)
4. **Visual Studio Code** - IDE principal de desarrollo

Especificaciones del Sistema

- **SO:** Windows 64-bit
- **Procesador:** Intel Core i3-1005G1 @ 1.20GHz
- **RAM:** 12.0 GB
- **Almacenamiento:** 238 GB disponibles

Proceso de Setup Realizado

1. Instalación de Herramientas

.NET SDK

```
bash
```

Link de descarga usado:

<https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/sdk-8.0.403-windows-x64-installer>

Verificación:

`dotnet --version`

Resultado: 8.0.403

Node.js

bash

Link de descarga usado:

<https://nodejs.org> (versión LTS)

Verificación:

`node --version` # v20.17.0

`npm --version` # 10.8.2

Visual Studio Code

bash

Link de descarga usado:

<https://code.visualstudio.com/sha/download?build=stable&os=win32-x64-user>

Opciones marcadas durante instalación:

- ☒ Add to **PATH**
- ☒ Register Code as editor **for** supported **file** types
- ☒ Add "Open with Code" action to context menus

2. Creación del Proyecto

Script de Setup Ejecutado

powershell

Ubicación del proyecto:

C:\Dev\panel-general-remotos

Comando ejecutado:

.\CreateProject.ps1

Estructura Creada

```
C:\Dev\panel-general-remotos\  
├── backend/                # .NET 8 Web API  
│   ├── src/PanelGeneralRemotos.Api/ # Proyecto principal API  
│   └── PanelGeneralRemotos.sln      # Solución .NET  
├── frontend/              # Angular 18  
│   └── panel-remotos-web/      # Aplicación Angular  
├── docs/                  # Documentación  
├── scripts/               # Scripts de automatización  
│   └── start-dev.bat         # Script para iniciar desarrollo  
├── .gitignore             # Control de versiones  
└── README.md              # Documentación principal
```

3. Configuración Backend (.NET)

Proyectos Creados

```
bash  
  
# Solución principal  
dotnet new sln -n PanelGeneralRemotos  
  
# Proyecto API  
dotnet new webapi -n PanelGeneralRemotos.Api -o src\PanelGeneralRemotos.Api  
dotnet sln add src\PanelGeneralRemotos.Api\PanelGeneralRemotos.Api.csproj
```

Paquetes NuGet Instalados

```
xml  
  
<PackageReference Include="Microsoft.AspNetCore.SignalR" />  
<PackageReference Include="Microsoft.AspNetCore.Cors" />  
<PackageReference Include="Swashbuckle.AspNetCore" />
```

Verificación

```
bash  
  
dotnet build  
# Resultado: Compilación correcta (0 errores, 0 advertencias)
```

4. Configuración Frontend (Angular)

Aplicación Creada

```
bash  
  
ng new panel-remotos-web --routing=true --style=scss --skip-git=true
```

Configuraciones Seleccionadas

- **Telemetría:** No (N)
- **SSR/SSG:** No (N)
- **Routing:** Sí
- **Estilos:** SCSS

Dependencias Adicionales

```
bash  
  
npm install @angular/material @angular/cdk @angular/animations  
npm install @microsoft/signalr  
npm install date-fns lodash-es
```

Verificación

```
bash  
  
npm list --depth=0  
# Resultado: Todas las dependencias instaladas correctamente
```

5. Visual Studio Code

Workspace Configurado

```
json
```

```
{
  "folders": [
    {
      "name": "Backend (.NET)",
      "path": "./backend"
    },
    {
      "name": "Frontend (Angular)",
      "path": "./frontend/panel-remotos-web"
    },
    {
      "name": "Documentation",
      "path": "./docs"
    }
  ]
}
```

Extensiones Recomendadas

- **C# Dev Kit** - Para desarrollo .NET
- **Angular Language Service** - Para Angular
- **TypeScript Importer** - Para TypeScript
- **Prettier** - Para formateo de código

URLs de Desarrollo

Una vez iniciados los servidores con `.\scripts\start-dev.bat`:

Servicio	URL	Descripción
Backend API	http://localhost:5000	API .NET principal
Swagger UI	http://localhost:5000/swagger	Documentación interactiva API
Frontend	http://localhost:4200	Aplicación Angular

Archivos de Configuración Creados

.gitignore

```
node_modules/  
dist/  
bin/  
obj/  
*.log  
.env  
credentials/  
.vs/  
.vscode/
```

scripts/start-dev.bat

```
batch  
  
@echo off  
echo =====  
echo  PANEL GENERAL REMOTOS - DEV  
echo  =====  
echo  Iniciando Backend (.NET API)...  
start "Backend API" cmd /k "cd /d C:\Dev\panel-general-remotos\backend && dotnet run --project src\PanelGeneralRe  
  
echo  Esperando 5 segundos...  
timeout /t 5 /nobreak >nul  
  
echo  Iniciando Frontend (Angular)...  
start "Frontend Angular" cmd /k "cd /d C:\Dev\panel-general-remotos\frontend\panel-remotos-web && ng serve"
```

Estado Actual del Proyecto

Completado

- ☒ Instalación de todas las herramientas necesarias
- ☒ Creación de estructura del proyecto
- ☒ Configuración de backend .NET con API básica
- ☒ Configuración de frontend Angular con routing
- ☒ Setup de Visual Studio Code con workspace
- ☒ Scripts de automatización para desarrollo
- ☒ Verificación de que todo compila y funciona

Próximos Pasos

- ☐ Configurar conexión con Google Sheets API
 - ☐ Implementar componentes del dashboard
 - ☐ Configurar SignalR para tiempo real
 - ☐ Crear modelos de datos para llamadas y rendimiento
 - ☐ Implementar autenticación/autorización
 - ☐ Diseñar UI/UX del dashboard
-

Comandos Útiles

Desarrollo Diario

powershell

Iniciar proyecto completo

cd C:\Dev\panel-general-remotos

.\scripts\start-dev.bat

Abrir en VS Code

code .

Build backend

cd backend

dotnet build

Instalar nuevas dependencias frontend

cd frontend\panel-remotos-web

npm install [package-name]

Gestión de Proyecto

powershell

Ver estructura

tree /f

Verificar estado de herramientas

dotnet --version

node --version

npm --version

Contacto y Documentación

- **Proyecto:** Panel General Remotos
 - **Ubicación:** C:\Dev\panel-general-remotos
 - **Fecha Setup:** 11 de Agosto 2025
 - **Documentación Técnica:** Ver README.md en el proyecto
-

Troubleshooting

Problemas Comunes

1. **Puerto ocupado:** Cambiar puertos en appsettings.json y angular.json
2. **Node modules:** Ejecutar npm install en carpeta frontend
3. **Build errors:** Ejecutar dotnet clean y dotnet build en backend
4. **VS Code no reconoce comandos:** Reinstalar extensiones recomendadas

Logs Importantes

- **Backend:** Consola de .NET muestra logs detallados
 - **Frontend:** Consola del navegador (F12) para errores Angular
 - **Build:** Mensajes de compilación en terminal
-

Documento generado el 11 de Agosto 2025 - Setup completado exitosamente 