

# Reporte Técnico del Proyecto Final

---

## Nombre del Proyecto

MAPIsecurity – Aplicación de seguridad ciudadana con funcionalidades de georreferenciación y reportes.

## Tecnologías Utilizadas

- Lenguaje de programación: C#
- Framework: .NET Framework (WinForms)
- Interfaz gráfica: Windows Forms
- Visualización de mapas: WebView2 + Leaflet.js (mapas interactivos)
- Base de datos: Microsoft SQL Server
- Librerías adicionales:
  - Microsoft.Web.WebView2
  - Newtonsoft.Json

## Estructura Técnica del Proyecto

- FormInicio.cs: Formulario principal para inicio de sesión y registro. Valida usuarios y se conecta con SQL Server.
- FormMenu.cs: Menú principal que navega entre las funcionalidades: mapa, reporte, consulta, y cierre de sesión.
- FormReporte.cs: Permite seleccionar una ubicación en un mapa y guardar un reporte clasificado automáticamente.
- FormConsulta.cs: Muestra los reportes registrados en una tabla y permite verlos en un mapa.
- FormMapa.cs: Muestra un mapa con marcadores de reportes y genera alertas si el usuario está cerca.
- ReporteUbicacion.cs: Clase modelo con las propiedades: Tipo, Descripcion, Latitud, Longitud.

## Conexión a Base de Datos

Se utiliza SqlConnection para conectar a SQL Server con la cadena de conexión:

```
Server=CARLO\SQLEXPRESS01;Database=MAPIsecurity;Trusted_Connection=True;
```

Tablas utilizadas:

- Usuarios (Id, NombreUsuario, Contraseña)
- Reportes (Id, UsuarioId, Tipo, Descripción, Latitud, Longitud, Fecha)

## Mapas y Geolocalización

- Motor de mapas: Leaflet.js
- Renderizado: WebView2 embebido en formularios
- Funcionalidades:
  - Selección de ubicación
  - Marcadores personalizados
  - Alertas por proximidad
  - Comunicación HTML ↔ C# vía postMessage y ExecuteScriptAsync

## Clasificación de Incidentes

Clasificación automática basada en palabras clave (ej: 'robo', 'asalto', 'accidente').  
Se tenía un bloque para integración futura con Gemini AI (comentado).

## APIs utilizadas en el proyecto

Aunque no se usaron APIs REST externas, el proyecto emplea varias APIs internas:

- **WebView2 API:** Para incrustar y comunicar contenido web (Leaflet) con WinForms.
- **SQL Server ADO.NET API:** Para conectarse y operar con la base de datos.
- **System.Media:** Para alertas sonoras.
- **Leaflet.js (JavaScript):** Para visualización de mapas dentro de WebView2.

## Observaciones Técnicas Finales

El proyecto tiene una arquitectura modular y clara separación de responsabilidades.  
Se emplea tecnología apropiada para interacción geográfica, interfaz de usuario y persistencia de datos.