



**2025**

Diseño e implementación de una aplicación móvil como herramienta tecnológica para la  
prevención de la violencia en Totonicapán, Guatemala.

## **MANUAL TECNICO**

Sistema y aplicación “SOS sin miedo”

**QUETZALTENANGO**

**V.0.1**

Para más información, por favor contactase

Danny Francisco Sapón Tax

(502) 5574 3737

[dsapont@miumg.edu.gt](mailto:dsapont@miumg.edu.gt)

## Contenido

Acerca del proyecto.....	5
1.    Introducción.....	6
2.    Diagramas UML: .....	6
Diagrama de secuencias .....	13
Diagrama de Actividades .....	14
Diagrama de Estado .....	15
3.    Diccionario de datos .....	18
4.    Dependencias:.....	37
Dependencias para la aplicación móvil.....	37
Para administrador Web .....	39
5.    Arquitectura general del sistema .....	41
Componentes principales .....	41
6.    Instalación del sistema web .....	43
a.    Clonar el repositorio .....	43
b.    Configurar entorno .....	43
c.    Instalar dependencias.....	43
d.    Migrar base de datos.....	43

e.	Generar clave de aplicación.....	43
f.	Configurar acceso público .....	44
7.	Estructura de carpetas (Laravel API).....	44
8.	Base de datos (MySQL) .....	44
9.	APIs y endpoints principales .....	45
10.	Integraciones externas .....	45
11.	Aplicación móvil (Flutter) .....	46
12.	Seguridad y control de acceso .....	46
13.	Logs y depuración .....	47
14.	Despliegue .....	47
15.	Respaldo y mantenimiento .....	47
16.	Escalabilidad y futuras mejoras .....	48
17.	Glosario .....	48
18.	Créditos técnicos .....	51

### **Acerca del proyecto**

Aplicación móvil desarrollada en *Flutter* con el lenguaje *Dart*, que permite reportar hechos de violencia para personas de todas las edades. Además, cuenta con una plataforma web desarrollada en *Laravel* con *PHP*, que facilita la gestión y derivación de los casos reportados desde la aplicación móvil.

## 1. Introducción

En el presente documento se describe los aspectos internos, arquitectónicos y funcionales del sistema SOS Sin Miedo, que combina una plataforma web institucional y una aplicación móvil ciudadana, conectadas mediante una API REST desarrollada en Laravel.

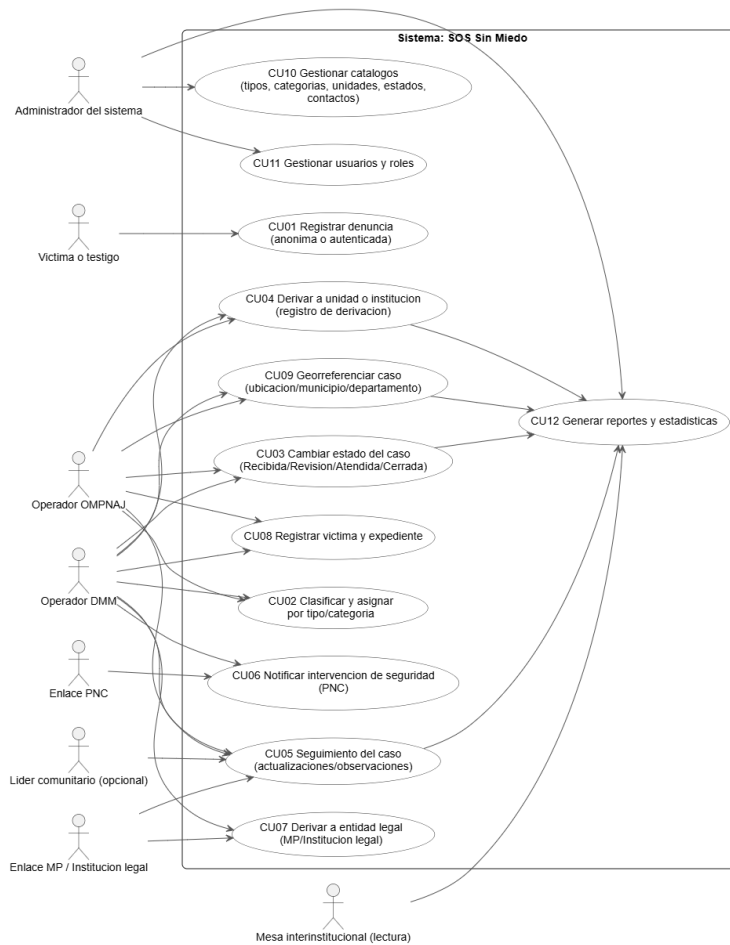
### Objetivo técnico:

Proporcionar a personal técnico y desarrolladores la información necesaria para:

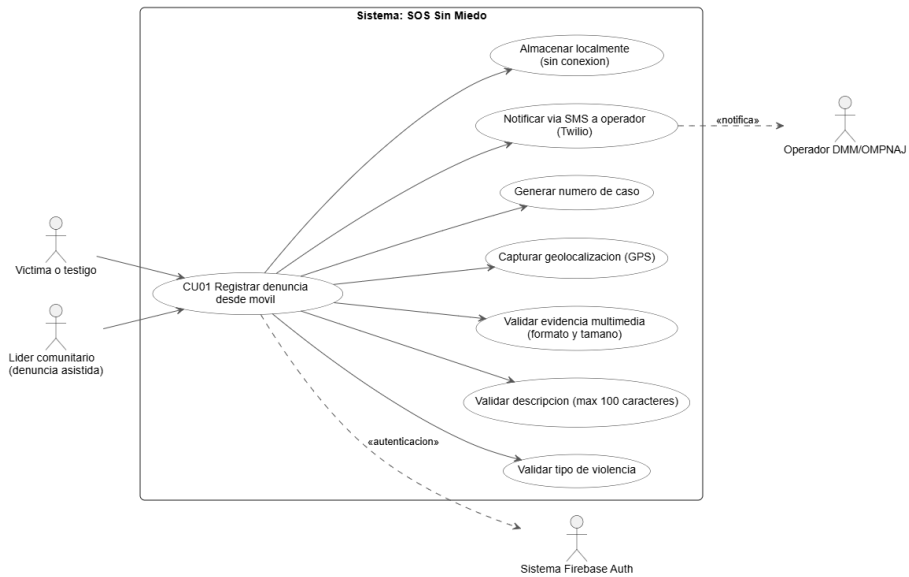
- Instalar, configurar, mantener y escalar el sistema.
- Comprender el flujo completo backend y frontend.
- Integrar nuevos módulos o mejorar los ya existentes.

## 2. Diagramas UML:

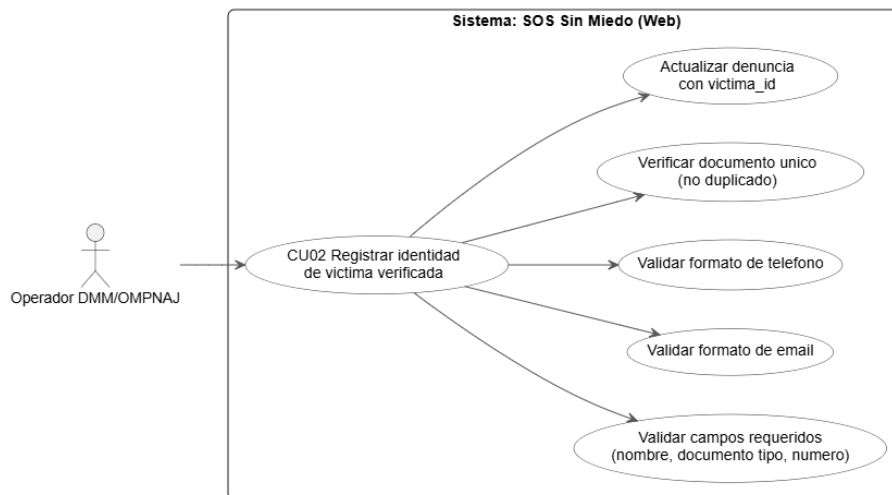
### Diagrama de caso de uso general



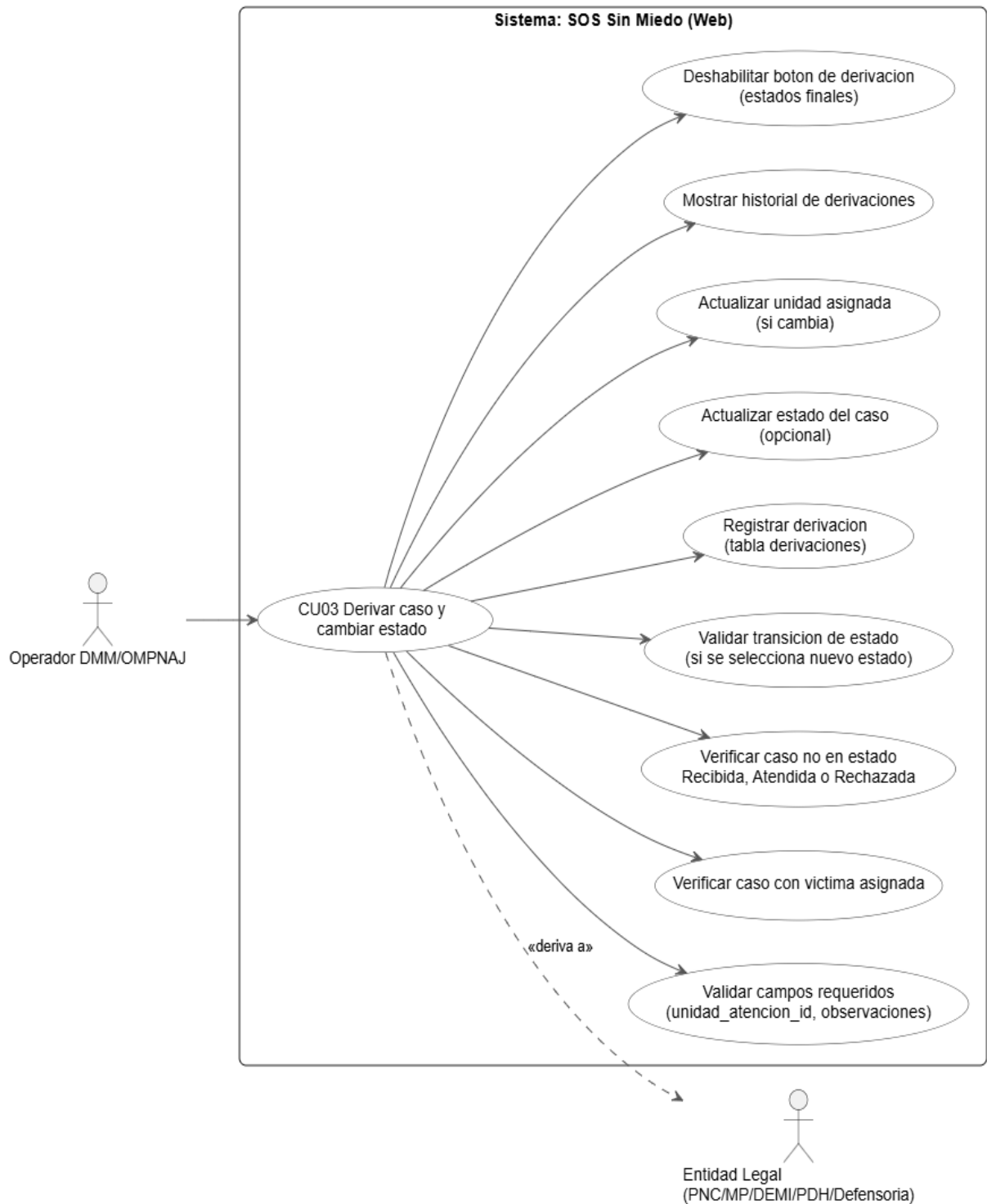
## Diagrama de caso de uso, registro de denuncia



## Diagrama de caso de uso, registro de victima

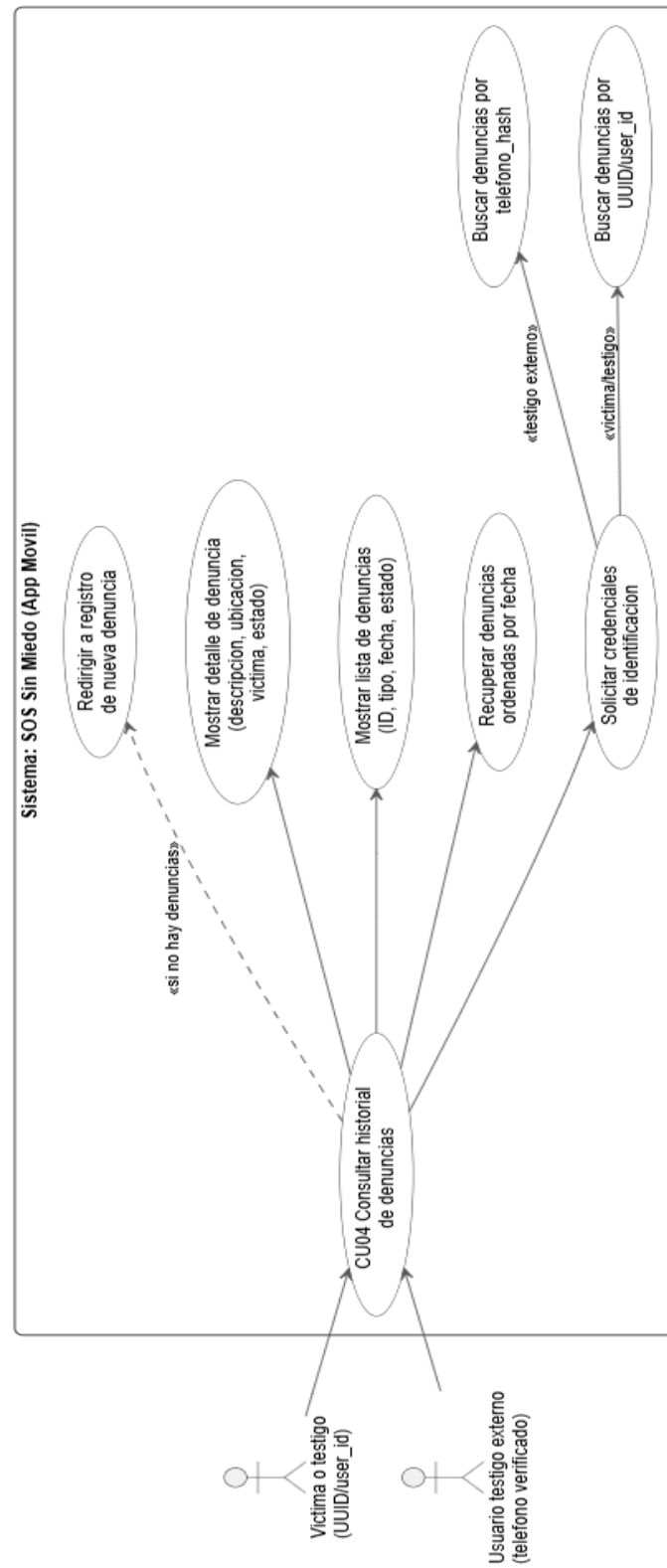


## Diagrama de caso de uso, derivaciones y cambio de estado

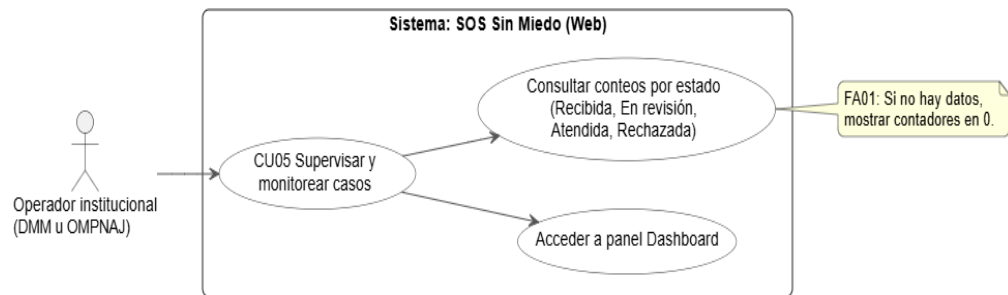




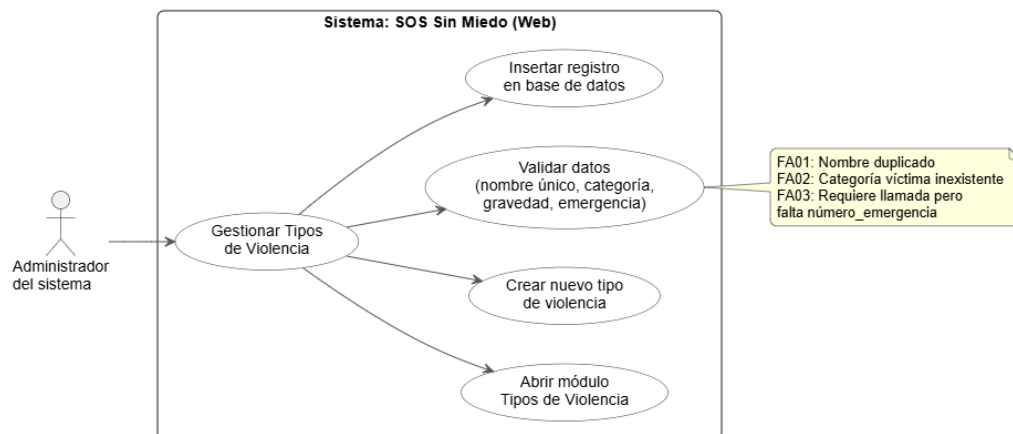
## Diagrama de caso de uso, consulta de historial de denuncias



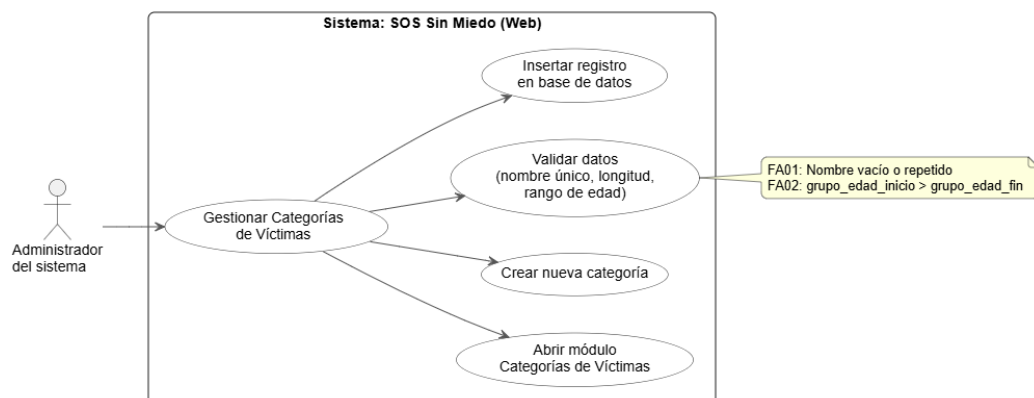
## Diagrama de caso de uso, supervisar y monitorear denuncias



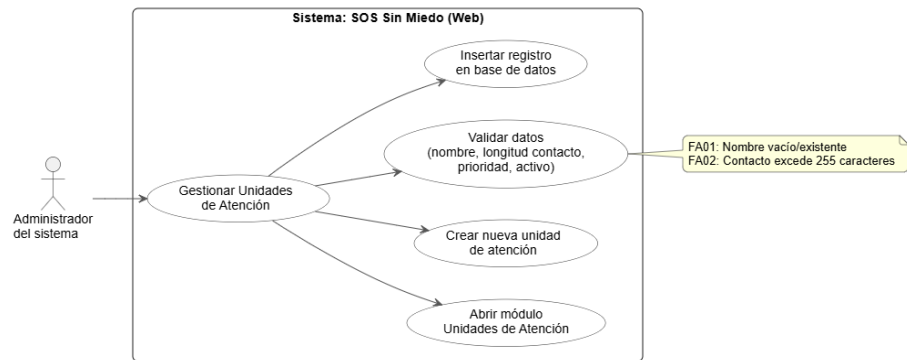
## Diagrama de caso de uso, tipos de violencia



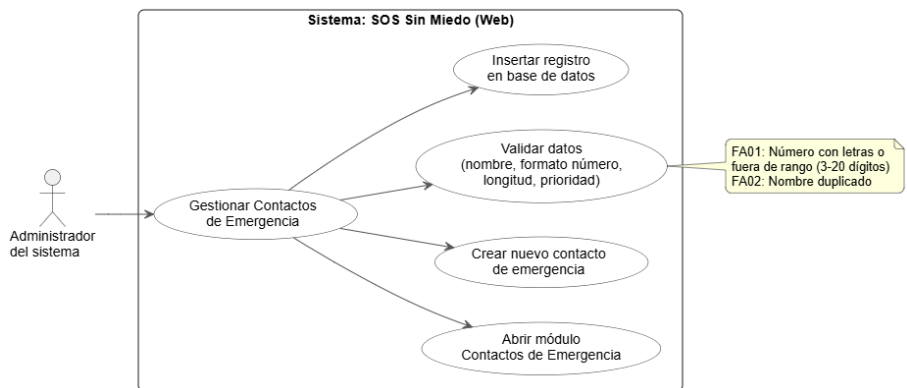
## Diagrama de caso de uso, categoría de víctimas



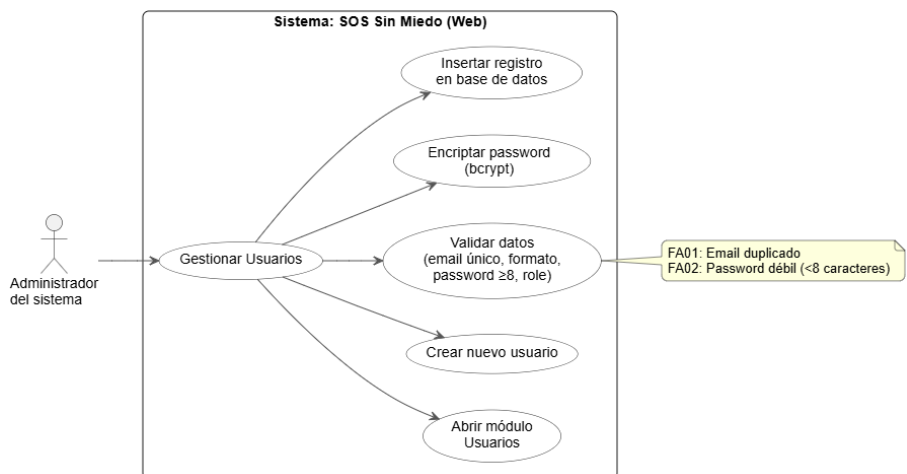
## Diagrama de caso de uso, unidades de atención



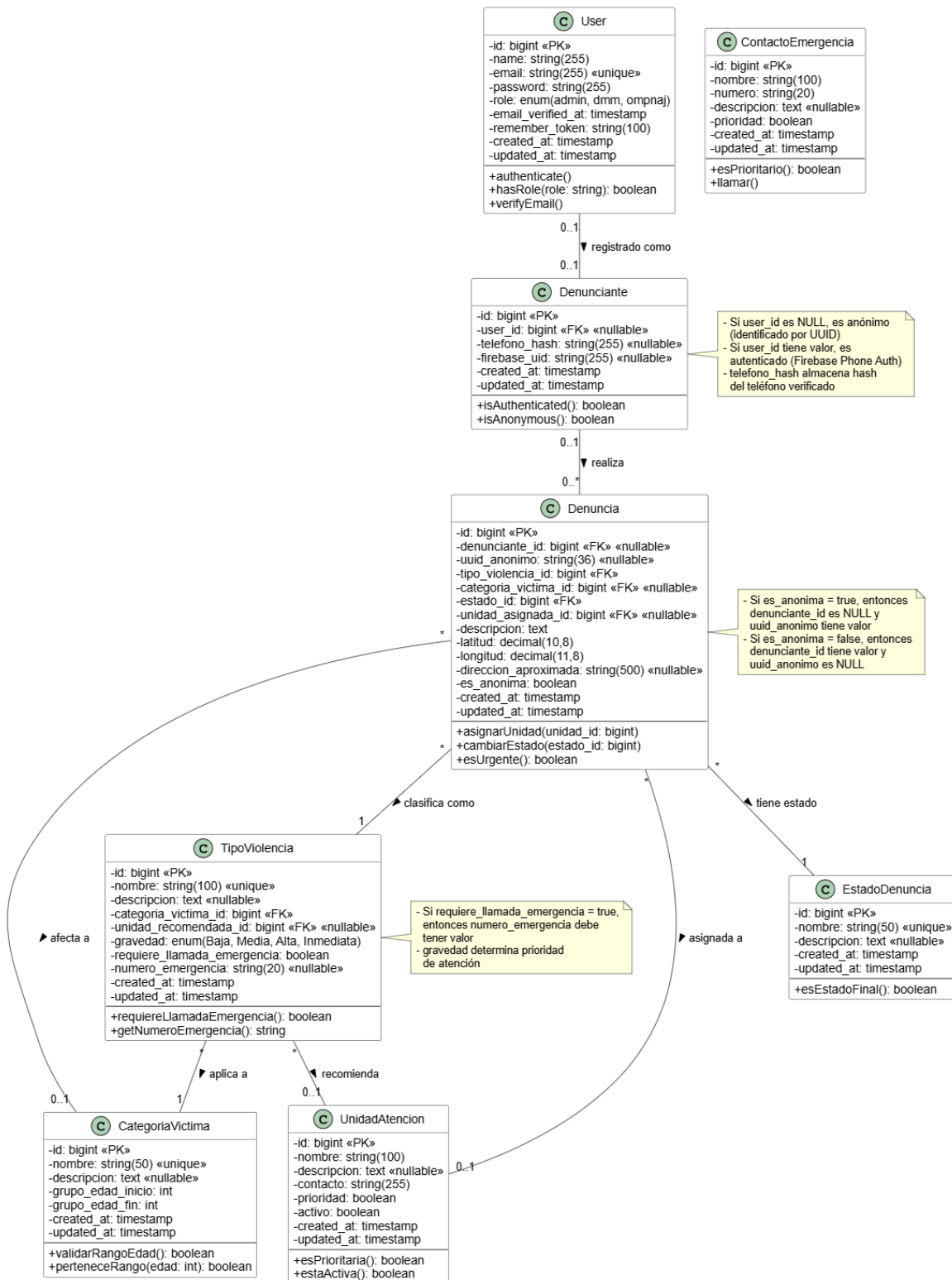
## Diagrama de caso de uso, contactos de emergencia



## Diagrama de caso de uso, gestión de usuarios web

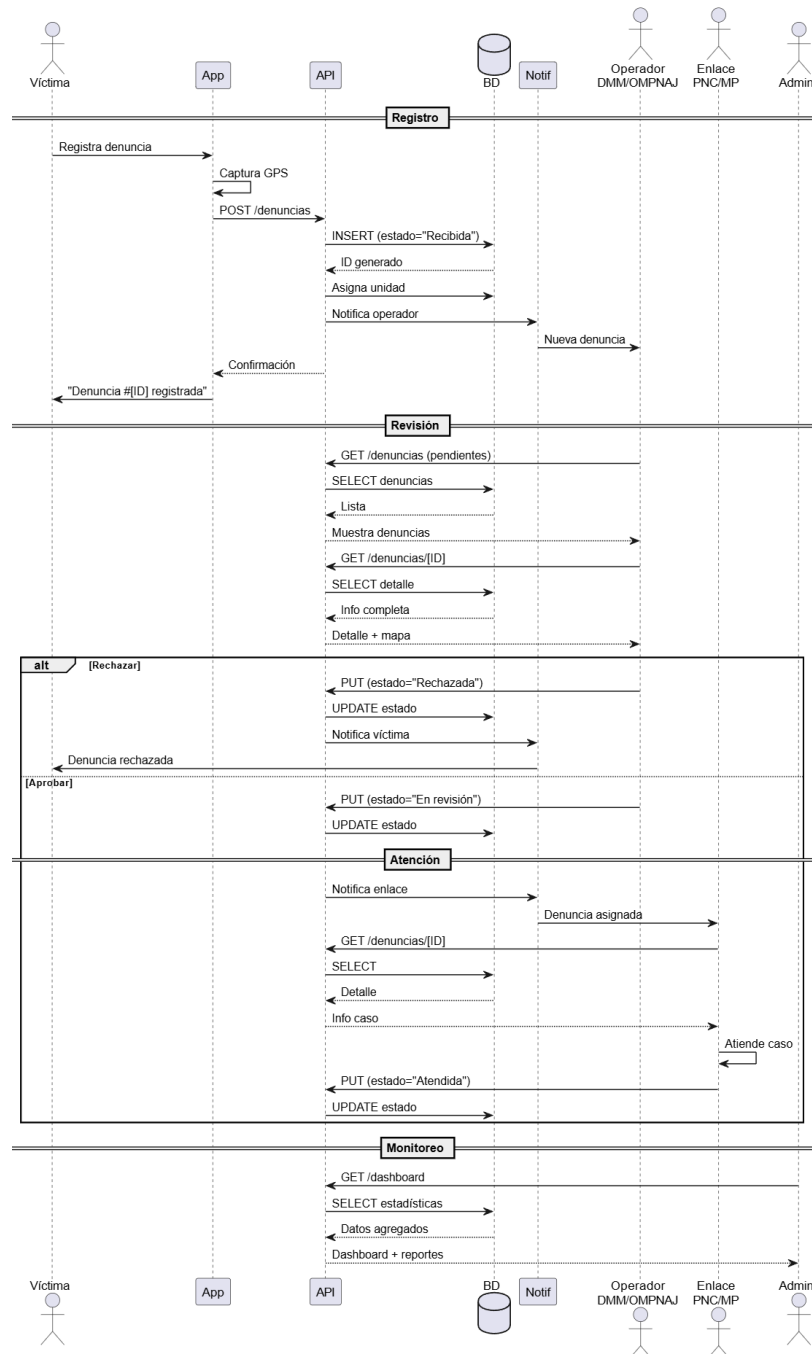


## Diagrama de Clases

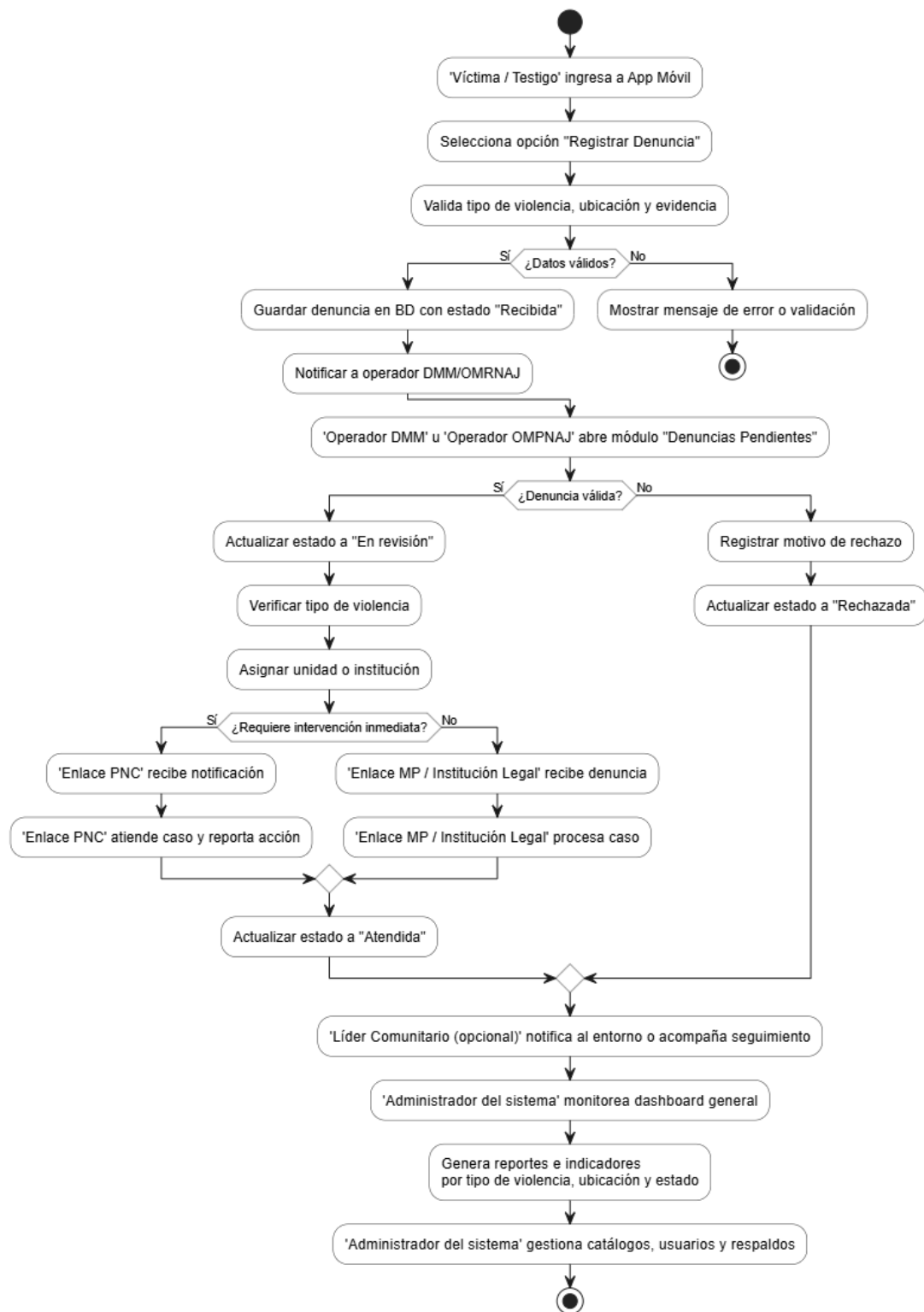


## Diagramas de Comportamiento

### Diagrama de secuencias



## Diagrama de Actividades



## Diagrama de Estado

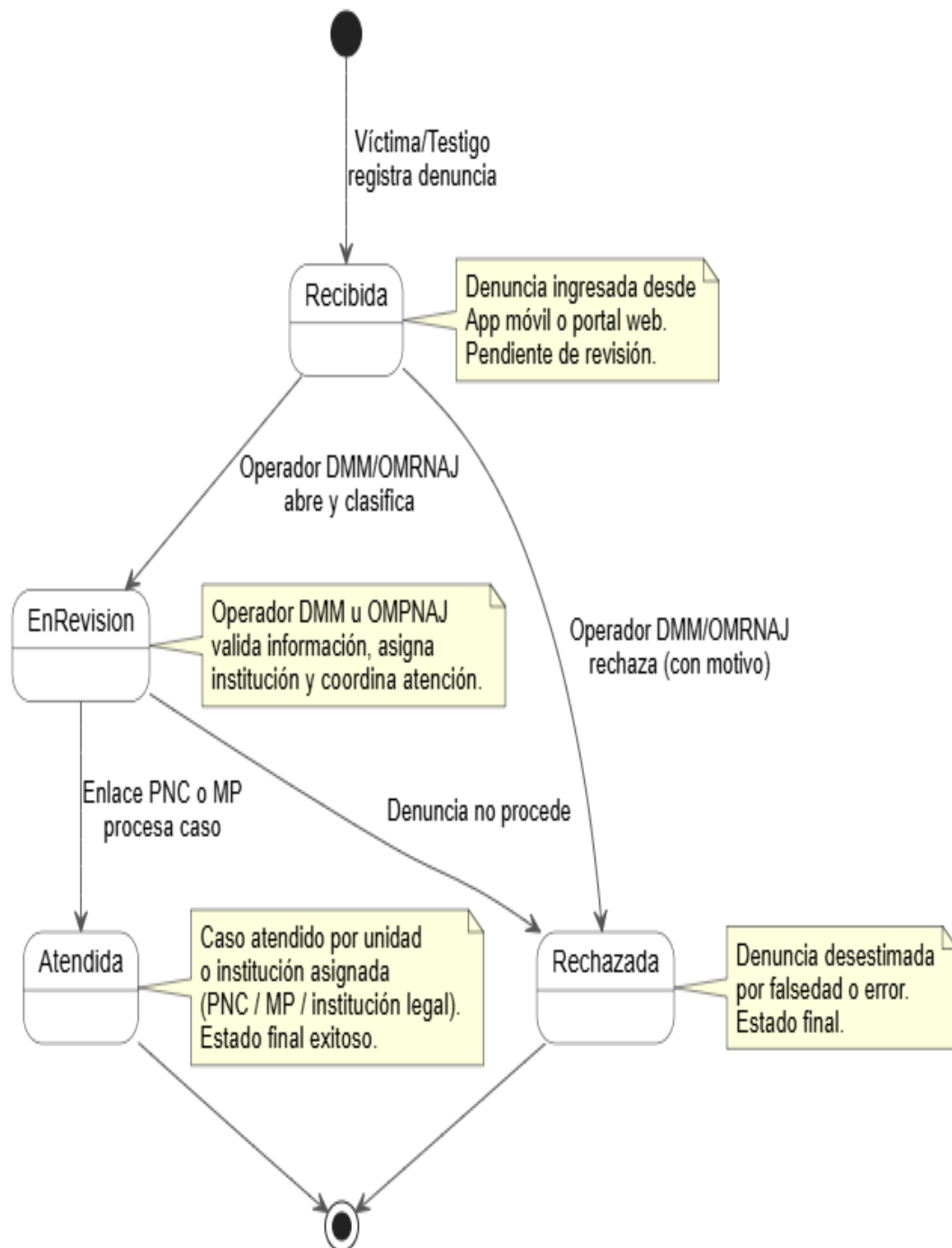


Diagrama de Componentes

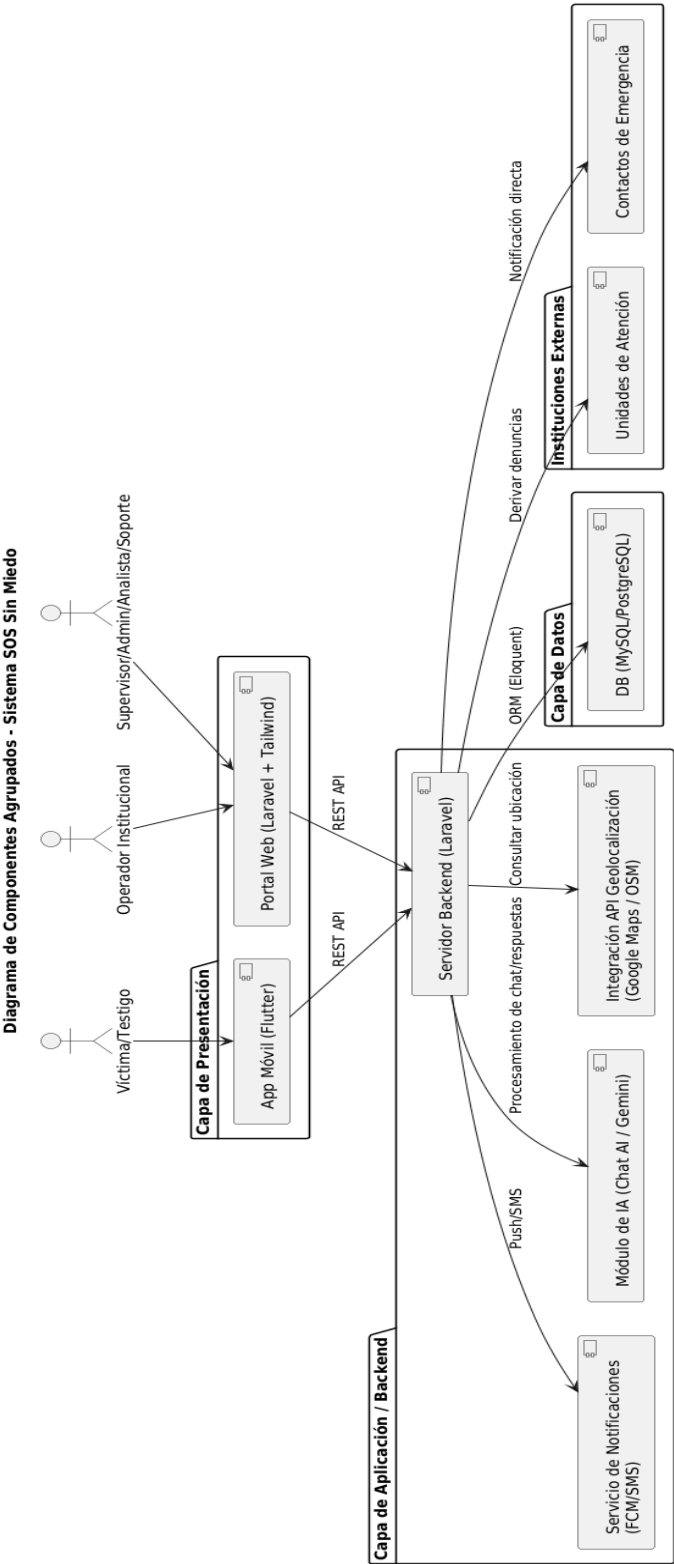
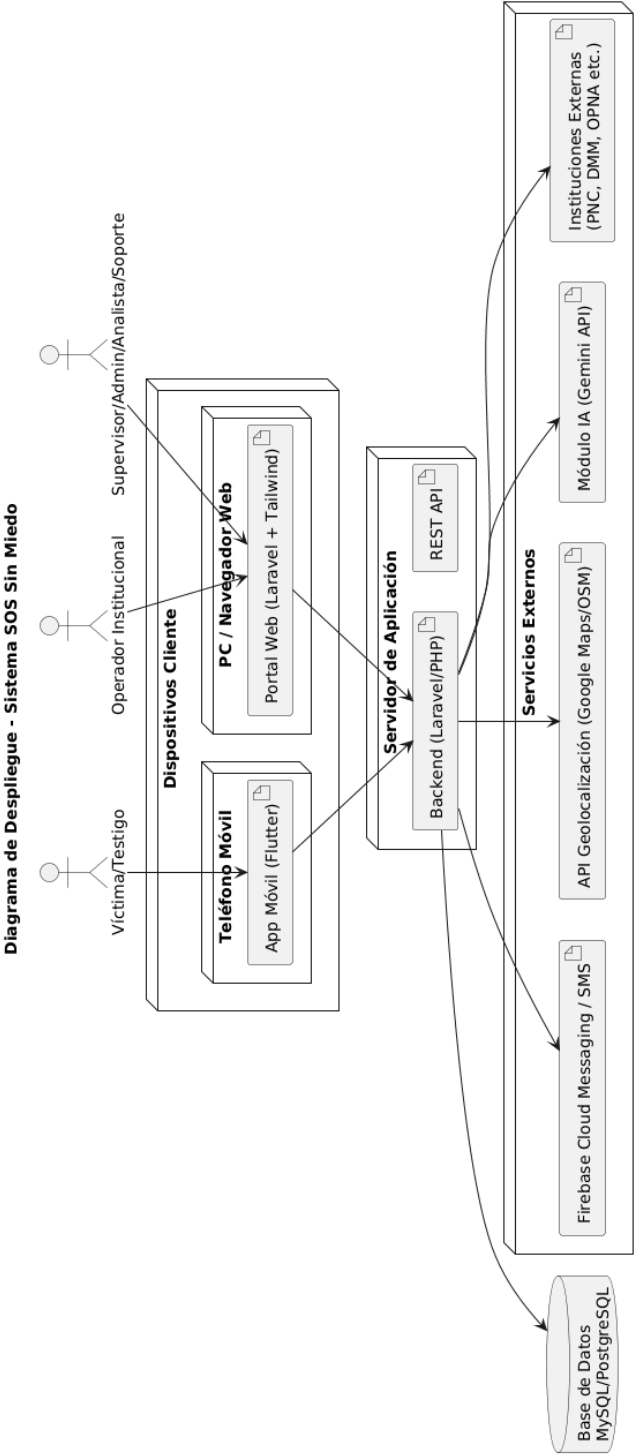




Diagrama de Despliegue



### 3. Diccionario de datos

El sistema fue creado para gestionar transacciones asociadas a la administración de denuncias por violencia en su diferente manifestación, abarcando:

#### Categoría víctimas

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment )	Identificador único de la categoría
<b>nombre</b>	50	Carácter	Nombre de la categoría (Mujer, Niño(a), Adolescente, Joven, Persona Mayor)
<b>descripcion</b>	-	Texto	Descripción de la categoría
<b>grupo_edad_inicio</b>	-	Entero	Edad mínima del grupo
<b>grupo_edad_fin</b>	-	Entero	Edad máxima del grupo
<b>activo</b>	-	Entero	Indica si la categoría está activa
<b>created_at</b>	-	Timestamp	Fecha de creación (CURRENT_TIMESTAMP)

Campo	Tamaño	Tipo de Dato	Descripción
<b>updated_at</b>	-	Timestamp	Fecha de actualización (CURRENT_TIMESTAMP)
Campo clave: <b>id</b>			<b>Relaciones:</b> Tipos de violencia

#### Logs de chats, usando el asistente por inteligencia artificial

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero grande (Auto Increment)	Identificador único del registro de chat
<b>user_id</b>	36	Carácter	Identificador del usuario
<b>user_uuid</b>	36	Carácter	UUID del usuario
<b>conversation_id</b>	50	Carácter	Identificador de la conversación (default: 'default')
<b>prompt</b>	-	Texto	Mensaje/pregunta del usuario
<b>response</b>	-	Texto largo	Respuesta del sistema

Campo	Tamaño	Tipo de Dato	Descripción
<b>usage</b>	-	Texto largo	Información de uso/estadísticas
<b>created_at</b>	-	Timestamp	Fecha de creación
<b>updated_at</b>	-	Timestamp	Fecha de actualización
Campo clave: <b>id</b>			<b>Relaciones:</b> Usuarios

#### Contactos de emergencia

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment)	Identificador único del contacto
<b>nombre</b>	100	Carácter	Nombre del contacto de emergencia
<b>numero</b>	20	Carácter	Número telefónico de emergencia
<b>descripcion</b>	-	Texto	Descripción del contacto
<b>prioridad</b>	-	Entero pequeño	Nivel de prioridad (default: 0)

Campo	Tamaño	Tipo de Dato	Descripción
<b>created_at</b>	-	Timestamp	Fecha de creación (CURRENT_TIMESTAMP)
<b>updated_at</b>	-	Timestamp	Fecha de actualización (CURRENT_TIMESTAMP)
Campo clave: <b>id</b>			

## Denunciantes

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment)	Identificador único del denunciante
<b>telefono</b>	15	Carácter	Hash del número telefónico (bcrypt)
<b>uid_firebase</b>	128	Carácter	Identificador de Firebase
<b>pais_codigo</b>	2	Carácter	Código del país (GT, MX, entre otros)
<b>ultimos_digitos</b>	4	Carácter	Últimos dígitos para mostrar: ••••5555

Campo	Tamaño	Tipo de Dato	Descripción
<b>ultimo_login_at</b>	-	Timestamp	Fecha del último login
<b>bloqueado</b>	-	Entero pequeño	Indica si está bloqueado (default: 0)
<b>created_at</b>	-	Timestamp	Fecha de creación (now())
<b>updated_at</b>	-	Timestamp	Fecha de actualización (now())
Campo clave: <b>id</b>			<b>Relaciones:</b> Denuncias

## Denuncias

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment)	Identificador único de la denuncia
<b>denunciante_id</b>	-	Entero	Identificador del denunciante
<b>tipo_violencia_id</b>	-	Entero	Identificador del tipo de violencia

<b>Campo</b>	<b>Tamaño</b>	<b>Tipo de Dato</b>	<b>Descripción</b>
<b>ubicacion_id</b>	-	Entero	Identificador de la ubicación
<b>lat</b>	10,7	Decima 1	Latitud geográfica
<b>lng</b>	10,7	Decima 1	Longitud geográfica
<b>descripcion</b>	-	Texto	Descripción del incidente
<b>anonima</b>	-	Entero pequeño	Indica si es anónima (default: 1)
<b>estado_id</b>	-	Entero	Estado de la denuncia (default: 1)
<b>fecha_incidente</b>	-	DateTi me	Fecha del incidente
<b>registrado_por</b>	-	Entero grande	Usuario que registró la denuncia
<b>numero_caso</b>	20	Carácte r	Número único del caso

Campo	Tamaño	Tipo de Dato	Descripción
<b>unidad_asignada_id</b>	-	Entero	Unidad asignada para atención
<b>categoria_victima_id</b>	-	Entero	Categoría de la víctima
<b>edad_victima</b>	-	Entero	Edad de la víctima
<b>genero_victima</b>	-	Enumerado	Género de la víctima
<b>sms_status</b>	20	Carácter	Estado del SMS enviado
<b>sms_provider_message_sid</b>	64	Carácter	ID del mensaje del proveedor SMS
<b>sms_sent_at</b>	-	DateTime	Fecha de envío del SMS
<b>created_at</b>	-	Timestamp	Fecha de creación (CURRENT_TIMESTAMP)
<b>updated_at</b>	-	Timestamp	Fecha de actualización (CURRENT_TIMESTAMP)
Campo clave: <b>id</b>			<b>Relaciones:</b> Denunciantes, Tipos de violencia,



Campo	Tamaño	Tipo de Dato	Descripción
			Ubicaciones, Estados denuncia, Unidades atención, Categorías víctimas, Usuarios

### Departamentos

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment)	Identificador único del departamento
<b>nombre</b>	100	Carácter	Nombre del departamento
<b>codigo</b>	10	Carácter	Código del departamento
Campo clave: <b>id</b>			<b>Relaciones:</b> Municipios, Ubicaciones

### Derivaciones

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto	Identificador único de la derivación

Campo	Tamaño	Tipo de Dato	Descripción
		Increment )	
<b>denuncia_id</b>	-	Entero	Identificador de la denuncia
<b>unidad_atencion_id</b>	-	Entero	Identificador de la unidad de atención
<b>observaciones</b>	-	Texto	Observaciones de la derivación
<b>derivado_por</b>	-	Entero grande	Usuario que realizó la derivación
<b>created_at</b>	-	Timestamp	Fecha de creación (CURRENT_TIMESTAMP)
<b>updated_at</b>	-	Timestamp	Fecha de actualización
Campo clave: <b>id</b>			<b>Relaciones:</b> Denuncias, Unidades atención, Usuarios

### Estados denuncia

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment)	Identificador único del estado
<b>nombre</b>	30	Carácter	Nombre del estado (Recibida, En revisión, Atendida)
<b>descripcion</b>	-	Texto	Descripción del estado
<b>color</b>	7	Carácter	Color representativo (default: '#0000FF')
<b>activo</b>	-	Entero	Indica si el estado está activo
<b>creado_en</b>	-	Timestamp	Fecha de creación (CURRENT_TIMESTAMP)
Campo clave: <b>id</b>			<b>Relaciones:</b> Denuncias

## Municipios

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto	Identificador único del municipio

Campo	Tamaño	Tipo de Dato	Descripción
		Increment )	
<b>nombre</b>	100	Carácter	Nombre del municipio
<b>codigo</b>	10	Carácter	Código del municipio
<b>departamento_id</b>	-	Entero	Identificador del departamento
Campo clave: <b>id</b>			<b>Relaciones:</b> Departamentos , Ubicaciones

#### Tokens por reinicio de contraseña

Campo	Tamaño	Tipo de Dato	Descripción
<b>email</b>	255	Carácter	Dirección de correo electrónico
<b>token</b>	255	Carácter	Token de restablecimiento
<b>created_at</b>	-	Timestamp	Fecha de creación
Campo clave: <b>email</b>			

### Tokens por acceso a sistema desde móviles

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero grande (Auto Increment)	Identificador único del token
<b>tokenable_type</b>	255	Carácter	Tipo de entidad asociada al token
<b>tokenable_id</b>	-	Entero grande	ID de la entidad asociada al token
<b>name</b>	255	Carácter	Nombre del token
<b>token</b>	64	Carácter	Token de acceso
<b>abilities</b>	-	Texto	Habilidades/permisos del token
<b>last_used_at</b>	-	Timestamp	Fecha del último uso
<b>expires_at</b>	-	Timestamp	Fecha de expiración
<b>created_at</b>	-	Timestamp	Fecha de creación
<b>updated_at</b>	-	Timestamp	Fecha de actualización
Campo clave: <b>id</b>			

## Sesiones

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	255	Carácter	Identificador de sesión
<b>user_id</b>	-	Entero grande	Identificador del usuario
<b>ip_address</b>	45	Carácter	Dirección IP del usuario
<b>user_agent</b>	-	Texto	Agente de usuario/navegador
<b>payload</b>	-	Texto largo	Datos de la sesión
<b>last_activity</b>	-	Entero	Última actividad registrada
Campo clave: <b>id</b>			<b>Relaciones:</b> Usuarios

## Tipos de violencia

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment)	Identificador único del tipo de violencia

<b>Campo</b>	<b>Tamaño</b>	<b>Tipo de Dato</b>	<b>Descripción</b>
<b>nombre</b>	100	Carácter	Nombre del tipo de violencia
<b>descripcion</b>	-	Texto	Descripción del tipo de violencia
<b>categoria_victima_id</b>	-	Entero	Categoría de víctima asociada
<b>unidad_recomendada_id</b>	-	Entero	Unidad recomendada para atención (DMM, OMPNAJ)
<b>gravedad</b>	-	Enumerado	Nivel de gravedad (default: 'Media')
<b>requiere_llamada_emergencia</b>	-	Entero pequeño	Indica si requiere llamada de emergencia (default: 0)
<b>numero_emergencia</b>	20	Carácter	Número de emergencia (Ej: 110, 122)
<b>mensaje_emergencia</b>	-	Texto	Mensaje al usuario si es emergencia
<b>activo</b>	-	Entero	Indica si el tipo está activo

Campo	Tamaño	Tipo de Dato	Descripción
<b>created_at</b>	-	Timesta mp	Fecha de creación (CURRENT_TIMEST AMP)
<b>updated_at</b>	-	Timesta mp	Fecha de actualización (CURRENT_TIMEST AMP)
Campo clave: <b>id</b>			<b>Relaciones:</b> Categorías víctimas, Unidades atención

## Ubicaciones

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment )	Identificador único de la ubicación
<b>lat</b>	10,8	Decimal	Latitud geográfica
<b>lng</b>	11,8	Decimal	Longitud geográfica



Campo	Tamaño	Tipo de Dato	Descripción
<b>precision_metros</b>	-	Flotante	Precisión de la ubicación en metros
<b>direccion</b>	150	Carácter	Dirección física
<b>municipio_id</b>	-	Entero	Identificador del municipio
<b>departamento_id</b>	-	Entero	Identificador del departamento
<b>created_at</b>	-	Timestamp	Fecha de creación (CURRENT_TIMESTAMP)
Campo clave: <b>id</b>			<b>Relaciones:</b> Municipios, Departamentos, Denuncias

### Unidades de atención

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero (Auto Increment)	Identificador único de la unidad

<b>Campo</b>	<b>Tamaño</b>	<b>Tipo de Dato</b>	<b>Descripción</b>
<b>nombre</b>	100	Carácter	Nombre de la unidad (DMM, OPNA, PNC y otros afines)
<b>descripcion</b>	-	Texto	Descripción de la unidad
<b>contacto</b>	-	Texto	Información de contacto (teléfono, correo, dirección)
<b>prioridad</b>	-	Entero pequeño	Prioridad (1 si responde en emergencia, default: 1)
<b>activo</b>	-	Entero pequeño	Indica si la unidad está activa (default: 1)
<b>created_at</b>	-	Timestamp	Fecha de creación (CURRENT_TIMESTAMP)
<b>updated_at</b>	-	Timestamp	Fecha de actualización (CURRENT_TIMESTAMP)
Campo clave: <b>id</b>			<b>Relaciones:</b> Denuncias, Derivaciones, Tipos de violencia

## Usuarios

Campo	Tamaño	Tipo de Dato	Descripción
<b>id</b>	-	Entero grande (Auto Increment)	Identificador único del usuario
<b>name</b>	255	Carácter	Nombre del usuario
<b>email</b>	255	Carácter	Correo electrónico del usuario
<b>email_verified_at</b>	-	Timestamp	Fecha de verificación de correo
<b>password</b>	255	Carácter	Contraseña del usuario
<b>remember_token</b>	100	Carácter	Token de recordatorio
<b>role</b>	255	Carácter	Rol del usuario (default: 'sin_rol')
<b>created_at</b>	-	Timestamp	Fecha de creación
<b>updated_at</b>	-	Timestamp	Fecha de actualización
Campo clave: <b>id</b>			<b>Relaciones:</b> Denuncias, Derivaciones, Víctimas, Sessions, Chat logs

## Victimas

<b>Campo</b>	<b>Tamaño</b>	<b>Tipo de Dato</b>	<b>Descripción</b>
<b>id</b>	-	Entero (Auto Increment)	Identificador único de la víctima
<b>denuncia_id</b>	-	Entero	Identificador de la denuncia asociada
<b>nombre_completo</b>	150	Carácter	Nombre completo de la víctima
<b>documento_tipo</b>	30	Carácter	Tipo de documento (DPI y Pasaporte)
<b>documento_numero</b>	50	Carácter	Número del documento
<b>telefono</b>	30	Carácter	Teléfono de la víctima
<b>email</b>	100	Carácter	Correo electrónico de la víctima
<b>direccion</b>	200	Carácter	Dirección de la víctima
<b>contacto_emergencia</b>	150	Carácter	Nombre y teléfono de contacto de emergencia

<b>Campo</b>	<b>Tamaño</b>	<b>Tipo de Dato</b>	<b>Descripción</b>
<b>notas_confidenciales</b>	-	Texto	Notas confidenciales sobre la víctima
<b>registrado_por</b>	-	Entero grande	Usuario que registró la víctima
<b>actualizado_por</b>	-	Entero grande	Usuario que actualizó la víctima
<b>created_at</b>	-	Timestamp	Fecha de creación
<b>updated_at</b>	-	Timestamp	Fecha de actualización
<b>Campo clave: id</b>			Relaciones: Denuncias, Usuarios

#### 4. Dependencias:

##### Dependencias para la aplicación móvil

<b>Paquete</b>	<b>Versión</b>	<b>Descripción técnica</b>	<b>Uso dentro del proyecto</b>
flutter	SDK	Framework principal para construir la aplicación móvil multiplataforma.	Base de toda la app. Proporciona los widgets y librerías nativas para Android e iOS.
http	^1.2.0	Cliente HTTP estándar de Dart para realizar peticiones REST.	Se usa para comunicación inicial o llamadas básicas a la API Laravel.
dio	^5.7.0	Cliente HTTP avanzado con	Principal herramienta para conectar con la

Paquete	Versión	Descripción técnica	Uso dentro del proyecto
		interceptores, manejo de errores y cabeceras personalizadas.	API Laravel (envío de denuncias, consultas y autenticación).
geolocator	^14.0.2	Permite obtener la localización del dispositivo con alta precisión.	Captura coordenadas GPS (latitud y longitud) al realizar una denuncia.
url_launcher	^6.3.2	Abre enlaces desde la app (teléfonos, correos, direcciones web).	Abre números de emergencia o recursos web desde la sección de “Ayuda”.
firebase_core	^4.2.0	Librería base para inicializar todos los servicios de Firebase.	Integración obligatoria para login con número telefónico.
firebase_auth	^6.1.1	Implementa la autenticación Firebase vía SMS.	Permite a los usuarios autenticarse mediante token SMS para las denuncias autenticadas.
provider	^6.0.5	Paquete de manejo de estado basado en el patrón Provider.	Gestiona el estado global de controladores (AuthController, ReportController, etc.).
flutter_secure_storage	^9.0.0	Permite guardar datos sensibles cifrados en el dispositivo.	Guarda de forma segura el token del usuario (Bearer Token Laravel Sanctum).
flutter_launcher_icons	^0.14.4	Personaliza el ícono y splash de la app.	Genera los íconos oficiales de la aplicación (SOS Sin Miedo).
cupertino_icons	^1.0.8	Paquete de íconos estilo iOS (complementario al Material).	Mejora visual y compatibilidad entre plataformas (Android/iOS).

## Para administrador Web

### Dependencias principales (Laravel Core)

Estas vienen instaladas por defecto al crear un proyecto Laravel:

Paquete	Descripción
"laravel/framework"	Núcleo del framework Laravel. Incluye Eloquent, Routing, Queues, Validation, etc.
"laravel/sanctum"	Manejo de autenticación por Token (usado en la app móvil para proteger tus endpoints).
"laravel/tinker"	Consola interactiva para desarrollo y depuración (PHP REPL).
"fruitcake/laravel-cors"	Habilita CORS para permitir peticiones desde la app Flutter o el frontend.
"guzzlehttp/guzzle"	Cliente HTTP usado para enviar peticiones externas (lo utilizas en Twilio).

### Dependencias adicionales instaladas por configuración del proyecto

Estas no son propias de Laravel, pero las agregamos tú y yo para funcionalidades específicas del sistema:

Paquete	Uso dentro del proyecto
"twilio/sdk"	Comunicación con la API de Twilio para enviar SMS de alerta.
"kreat/laravel-firebase"	Validación de tokens de Firebase (para login desde la app móvil).
"spatie/laravel-permission"	Asignación de roles y permisos (usuarios de la plataforma web).
"barryvdh/laravel-debugbar" ( <i>en desarrollo</i> )	Depuración visual de consultas, rutas y performance en entorno local.
"laravel/ui" o "laravel/breeze" ( <i>dependiendo del setup web</i> )	Generadores de scaffolding para login/register/reset con autenticación por correo.
"intervention/image" ( <i>opcional</i> )	Procesamiento de imágenes (si planeas permitir subir evidencias o fotos).
"barryvdh/laravel-dompdf" ( <i>opcional</i> )	Generación de reportes PDF o descargas institucionales.

## Dependencias de desarrollo

Estas ayudan durante el desarrollo, pero no son necesarias en producción:

Paquete	Uso
"fakerphp/faker"	Generación de datos falsos para pruebas (usuarios, denuncias, etc.).
"phpunit/phpunit"	Framework de pruebas para Laravel.
"mockery/mockery"	Mocking para test unitarios.
"nunomaduro/collision"	Mejora el manejo visual de errores en la terminal.
"spatie/laravel-ignition"	Mejora la detección y visualización de errores durante desarrollo.

## Dependencias NPM (para el panel web con Tailwind CSS)

En el frontend de Laravel (panel de administración o portal web), usas las siguientes dependencias Node:

Paquete	Descripción
"tailwindcss"	Framework CSS utilizado en todas las vistas del panel.
"autoprefixer"	Añade compatibilidad automática para navegadores.
"postcss"	Procesador de CSS para Tailwind.
"vite"	Compilador moderno (Laravel Mix ya fue reemplazado).
"alpinejs"	Librería ligera para interactividad en el portal web.

Se instala con:

- npm install
- npm run build



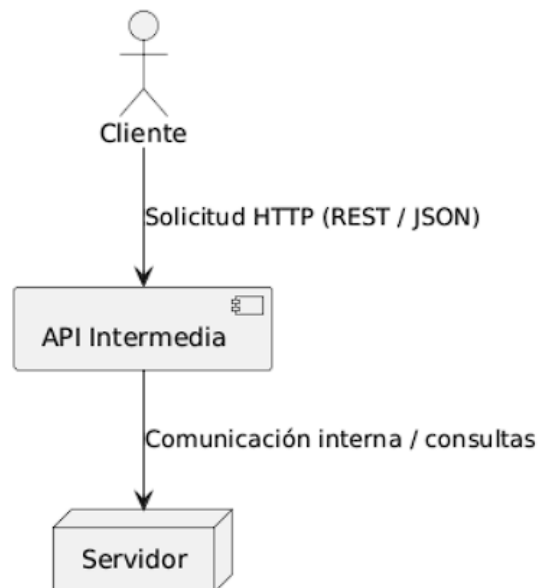
## 5. Arquitectura general del sistema

### Componentes principales

Componente	Descripción
<b>Frontend Web (Laravel Blade / Tailwind)</b>	Panel para operadores e instituciones. Permite login con correo, validación, gestión de denuncias y usuarios.
<b>Backend (Laravel API + Sanctum)</b>	API REST que gestiona autenticaciones, registros de denuncias, alertas SMS y consultas.
<b>Aplicación móvil (Flutter)</b>	App Android que permite al ciudadano enviar denuncias (autenticadas o anónimas).
<b>Autenticación Firebase</b>	Uso de Firebase Auth para validación vía número telefónico en la app.
<b>SMS Alerts (Twilio)</b>	Servicio externo que envía alertas por SMS a unidades de atención.
<b>Base de datos MySQL</b>	Persistencia de entidades principales: denuncias, usuarios, tipos de violencia, unidades, etc.

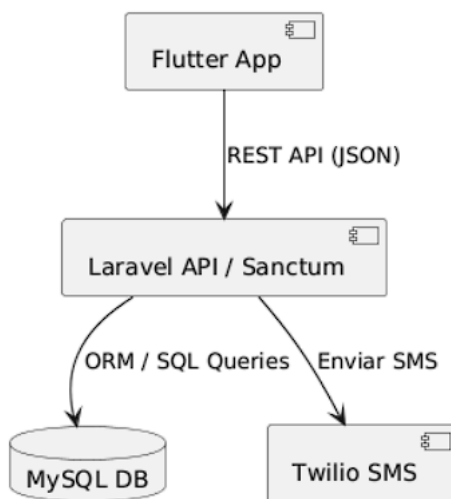
### Patrón de arquitectura:

#### Modelo Cliente - Servidor con API Intermedia



## Diagrama general (resumen):

### Arquitectura General - Flutter + Laravel + MySQL + Twilio



## Requisitos técnicos

### Servidor Backend (Laravel)

- PHP 8.2 o superior
- Composer 2
- MySQL 8.0 o MariaDB
- Node.js (para compilar Tailwind en el panel web) en este caso se compila en local todo y se sube ya compilado dado que se usa un servidor compartido, al usar un servidor dedicado, se rewliza la compilación en el servidor
- Extensiones requeridas: pdo, json, mbstring, openssl

### Configuración recomendada

Entorno	Valor
Servidor web	Nginx
Sistema operativo	Windows Server
Hosting	VPSm o server Compartido con mínimo 4 GB RAM
Certificado SSL	Obligatorio (para API segura en móviles)

## 6. Instalación del sistema web

### a. Clonar el repositorio

```
git clone https://github.com/DannyS12/sos\_sinmiedo.git  
cd sos_sinmiedo
```

### b. Configurar entorno

Editar .env con las credenciales necesarias:

```
APP_NAME="SOS Sin Miedo"  
APP_ENV=production  
APP_URL=https://sossinmiedo.midominio.com  
  
DB_CONNECTION=mysql  
DB_HOST=127.0.0.1  
DB_PORT=3306  
DB_DATABASE=sos_db  
DB_USERNAME=root  
DB_PASSWORD=tu_password  
  
TWILIO_ACCOUNT_SID=xxxxxxxxxxxxx  
TWILIO_AUTH_TOKEN=xxxxxxxxxxxxx  
TWILIO_PHONE_NUMBER=+502XXXXXXX  
  
SANCTUM_STATEFUL_DOMAINS=sistemas502.com
```

### c. Instalar dependencias

```
composer install  
npm install && npm run build
```

### d. Migrar base de datos

```
php artisan migrate --seed
```

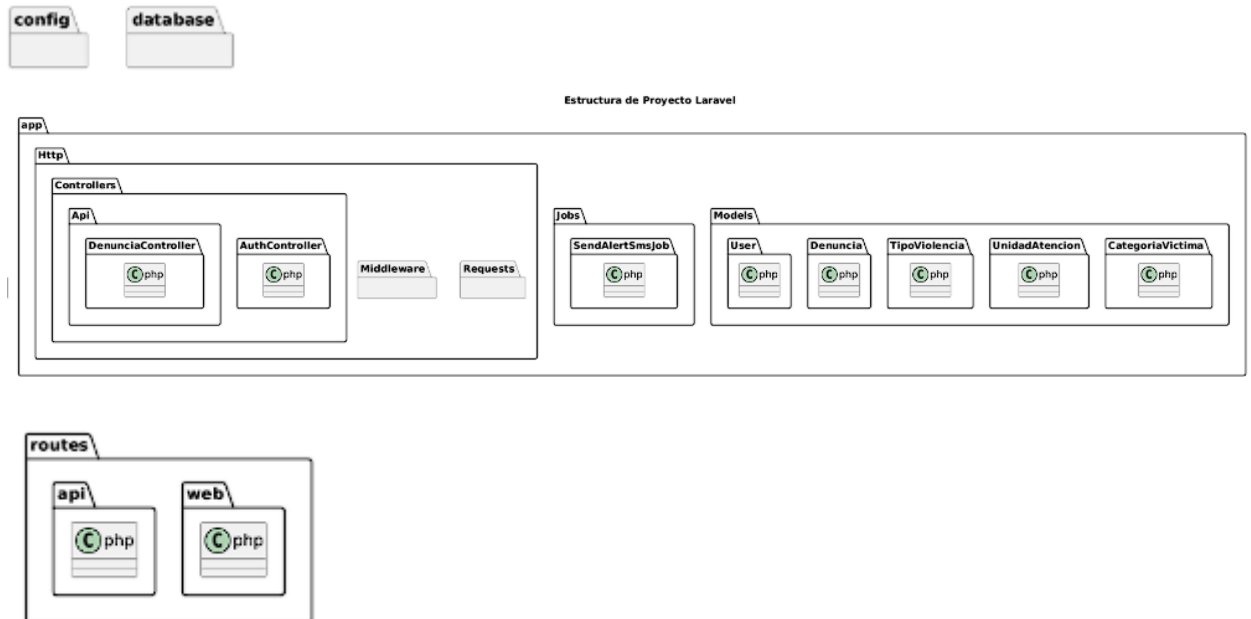
### e. Generar clave de aplicación

```
php artisan key:generate
```

## f. Configurar acceso público

Asegúrate de que el servidor apunte a la carpeta /public.

## 7. Estructura de carpetas (Laravel API)



## 8. Base de datos (MySQL)

### Tablas principales

Tabla	Descripción
<b>users</b>	Usuarios de la plataforma (web y app autenticada).
<b>roles</b>	Define permisos y niveles de acceso.
<b>denuncias</b>	Registros de las denuncias realizadas.
<b>tipos_de_violencia</b>	Catálogo de tipos de violencia.
<b>unidades_de_atencion</b>	Instituciones o unidades receptoras de casos.
<b>categorias_victimas</b>	Agrupaciones por rango etario o condición.
<b>ubicaciones</b>	Coordenadas geográficas de las denuncias.

### Relaciones principales:

User (1) --- (N) Denuncia  
Denuncia (N) --- (1) TipoViolencia  
Denuncia (N) --- (1) UnidadAtencion

Denuncia (N) --- (1) CategoriaVictima  
Denuncia (1) --- (1) Ubicacion

## 9. APIs y endpoints principales

Método	Ruta	Descripción	Auth
POST	/api/denuncias/anonima	Registrar denuncia anónima	No
POST	/api/denuncias	Registrar denuncia autenticada	Sí
GET	/api/denuncias/mis-denuncias	Listar denuncias del usuario autenticado	Sí
GET	/api/tipos-violencia	Catálogo de tipos de violencia	No
GET	/api/unidades	Listado de unidades de atención	Sí
POST	/api/login	Login web (correo/contraseña)	No
POST	/api/register	Registro web + validación de correo	No
POST	/api/password/reset	Solicitud para restaurar contraseña	No

Las peticiones autenticadas se manejan con Laravel Sanctum (Bearer Token).

## 10. Integraciones externas

### Firebase Authentication (Flutter App)

- Firebase se usa solo para el flujo de autenticación vía número telefónico.
- Una vez validado el token Firebase en el backend, Laravel genera un token Sanctum.

### Twilio

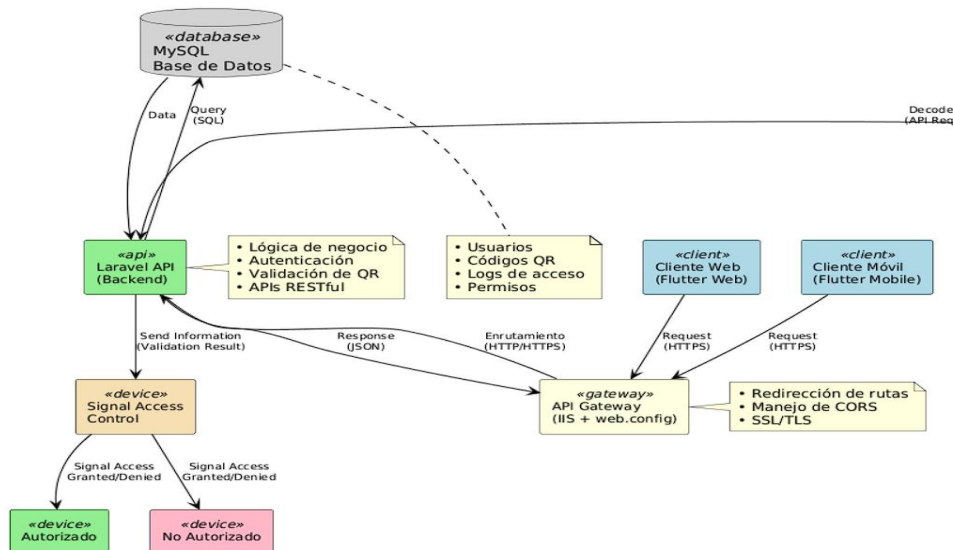
- Se usa para enviar SMS automáticos con el módulo SendAlertSmsJob.

- El envío se dispara tras el guardado exitoso de una denuncia.
- Los errores (por número inválido, etc.) se registran en storage/logs/laravel.log.

## 11. Aplicación móvil (Flutter)

- Clona desde la misma dirección que la versión web  
[https://github.com/DannyS12/sos\\_sinmiedo.git](https://github.com/DannyS12/sos_sinmiedo.git)

### Estructura general



### Funcionalidades principales

- Inicio de sesión (Firebase Phone Auth)
- Denuncia anónima o autenticada
- Envío a endpoints Laravel vía Dio
- Consulta de denuncias personales (/mis-denuncias)
- Ver detalle en modal con ubicación y estado

## 12. Seguridad y control de acceso

- **Laravel Sanctum** maneja tokens personales para app y web.
- **Roles y permisos** asignados a nivel de usuario en roles y role\_user.
- Todas las rutas críticas (/denuncias, /mis-denuncias) están protegidas por middleware auth:sanctum.
- **Validación del formato Twilio (E.164)** evita fallos de SMS.
- Se recomienda activar HTTPS obligatorio para API y app.

### 13. Logs y depuración

Ubicación	Descripción
storage/logs/laravel.log	Errores de Laravel o Twilio
flutter_logs/ (opcional)	Registro de errores en la app
php artisan tinker	Pruebas de endpoints
php artisan queue:work	Procesamiento de SMS en background (en caso de cola real)

### 14. Despliegue

#### Backend Laravel

- Subir código a servidor (Plesk / VPS / Hostinger.)
- Configurar dominio y SSL
- Asegurar que DocumentRoot → /public
- Ejecutar:

```
composer install --no-dev
php artisan migrate --force
php artisan config:cache
php artisan route:cache
```

#### Aplicación móvil

1. Configurar archivo lib/core/config/app\_config.dart con URL base (ej: <https://sossinmiedo.midominio.com/public/api>).
2. Compilar versión release:

```
flutter build apk --release
```

3. Firma y distribución (Play Store / instalación directa).

### 15. Respaldo y mantenimiento

Proceso	Frecuencia	Herramienta
Backup de base de datos	Diario	mysqldump / cron
Logs del servidor	Semanal	logrotate o limpieza manual
Actualización de dependencias	Mensual	composer update
Certificados SSL	Anual	certbot o panel hosting

## 16. Escalabilidad y futuras mejoras

- Implementación de **notificaciones push** (Firebase Messaging).
- Integración de **mapas interactivos** para ver ubicación de denuncias.
- Panel analítico para estadísticas de casos.
- Modo offline para denuncias sin conexión.
- Migración de colas SMS Twilio a **Redis Queue Worker**.

## 17. Glosario

### Términos generales del sistema

Término	Definición	Contexto
Sistema	Conjunto de componentes (web, móvil, backend, BD) que interactúan para cumplir los objetivos.	Se refiere a la solución completa: portal web + app móvil + API.
Aplicación Móvil (App)	Software diseñado para dispositivos móviles (smartphones, tablets).	La app Flutter que usan los ciudadanos para denunciar.
Portal Web	Interfaz de usuario accesible vía navegador web.	La plataforma Laravel para operadores y administradores.
Backend	Parte del sistema que procesa la lógica de negocio, gestiona la BD y la API.	El servidor Laravel que recibe denuncias, envía SMS y gestiona datos.
Frontend	Interfaz de usuario con la que interactúa el usuario final.	La app Flutter y las vistas Blade/Tailwind del portal web.
API (Application Programming Interface)	Conjunto de reglas que permite a diferentes programas comunicarse entre sí.	La API REST de Laravel que conecta la app móvil con el backend.
Base de Datos (BD)	Colección organizada de información estructurada.	MySQL, donde se almacenan denuncias, usuarios, tipos de violencia, etc.
Middleware	Software que actúa como puente entre sistemas operativos o aplicaciones.	En Laravel, filtra peticiones (ej. auth:sanctum para proteger rutas).
Endpoint	URL específica de una API a la que se envía una petición.	Ej. /api/denuncias/anonima o /api/denuncias/mis-denuncias.
Token	Cadena de caracteres que representa una credencial de autenticación.	Usado por Firebase (SMS) y Laravel Sanctum (Bearer Token).



<b>Término</b>	<b>Definición</b>	<b>Contexto</b>
Geolocalización (GPS)	Determinación de la posición geográfica de un objeto o persona.	Coordenadas (latitud/longitud) adjuntas a cada denuncia.
Logs	Registros de eventos, errores o actividades del sistema.	Archivos donde se guardan errores de Laravel, Twilio, etc. (storage/logs/laravel.log).

## **Términos de usuario y autenticación**

<b>Término</b>	<b>Definición</b>	<b>Contexto</b>
Usuario Ciudadano	Persona que utiliza la aplicación móvil para realizar denuncias.	Accede por número telefónico o de forma anónima.
Usuario Institucional	Personal autorizado que gestiona denuncias desde el portal web.	Accede por correo electrónico y tiene un rol asignado.
Autenticación	Proceso de verificar la identidad de un usuario.	Login con Firebase (móvil) o correo/contraseña (web).
Autorización	Proceso de determinar qué acciones puede realizar un usuario autenticado.	Definido por los roles y permisos en el portal web.
Perfil de Usuario	Conjunto de atributos y permisos asociados a una cuenta de usuario.	Determina las funcionalidades accesibles en el portal web (ej. Administrador, Operador).
Denunciante	Persona que realiza una denuncia.	Puede ser anónimo o un usuario autenticado.
Token de Verificación (SMS)	Código numérico enviado por SMS para confirmar la identidad.	Usado por Firebase para el login móvil.
Token de Acceso (Bearer Token)	Credencial generada por el backend para acceder a recursos protegidos.	Generado por Laravel Sanctum, usado por la app móvil.
Cerrar Sesión (Logout)	Finalizar la sesión activa de un usuario.	Disponible en la app móvil y el portal web.

## Términos de denuncias y gestión de casos

<b>Término</b>	<b>Definición</b>	<b>Contexto</b>
Denuncia	Reporte formal de un incidente o acto de violencia.	El registro principal del sistema, creado desde la app o el portal.
Denuncia Anónima	Reporte sin identificación del denunciante.	Permite a los usuarios enviar información sin revelar su identidad.
Denuncia Autenticada	Reporte vinculado a la cuenta de un usuario registrado.	Permite al usuario hacer seguimiento de su caso.
Número de Caso	Identificador único asignado a cada denuncia.	Facilita el seguimiento y la referencia de un reporte.
Tipo de Violencia	Clasificación del incidente reportado (ej. física, psicológica).	Campo obligatorio en el formulario de denuncia.
Estado de la Denuncia	Fase actual en la que se encuentra el proceso de una denuncia.	Ej. Pendiente, En Proceso, Cerrado.
Unidad de Atención	Institución o departamento responsable de gestionar un tipo de denuncia.	Asignada automáticamente según el tipo de violencia y/o ubicación.
Categoría de Víctima	Clasificación de la persona afectada (ej. menor, adolescente, adulto).	Ayuda a las unidades a priorizar y especializar la atención.
Evidencia	Material de apoyo (fotos, videos) adjunto a una denuncia.	Opcional en el formulario de denuncia.
Alerta SMS	Mensaje de texto automático enviado a contactos de emergencia.	Disparado por Twilio al registrar una denuncia.
<b>Mapa Geográfico</b>	Herramienta visual integrada en el sistema que representa espacialmente las denuncias mediante marcadores en un mapa digital, permitiendo la localización y análisis geoespacial de los incidentes reportados.	Utilizado cuando se necesita visualizar los casos en el mapa
Ruta de Denuncia	Flujo o pasos que sigue una denuncia desde su creación hasta su resolución.	Información disponible en la sección de ayuda.

## Términos de tecnologías e integraciones

Término	Definición	Contexto
Flutter	Framework de Google para desarrollar apps móviles multiplataforma.	Usado para construir la aplicación móvil de SOS Sin Miedo.
Laravel	Framework PHP para el desarrollo de aplicaciones web y APIs.	Usado para el backend y el portal web.
Firebase	Plataforma de Google para desarrollo de apps (autenticación, bases de datos, etc.).	Usado específicamente para la autenticación por número telefónico (Firebase Auth).
Twilio	Plataforma de comunicación en la nube para envío de SMS, llamadas, etc.	Usado para enviar las alertas SMS a las unidades de atención.
MySQL	Sistema de gestión de bases de datos relacionales.	Base de datos principal del sistema.
Sanctum	Paquete de autenticación de Laravel para APIs y SPAs.	Genera y valida los tokens de acceso para la app móvil.
Dio	Cliente HTTP para Dart/Flutter con características avanzadas.	Usado en la app Flutter para todas las peticiones a la API Laravel.
Provider	Patrón de gestión de estado para Flutter.	Organiza y comparte la lógica de negocio (controladores) en la app.
Tailwind CSS	Framework CSS utilitario para diseño rápido de interfaces.	Usado en el frontend del portal web.

## Glosario de acrónimos

Acrónimo	Significado
API	Application Programming Interface
BD	Base de Datos
GPS	Global Positioning System
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
SMS	Short Message Service
URL	Uniform Resource Locator
UX	User Experience
UI	User Interface

## 18. Créditos técnicos

Área	Responsable
Arquitectura e implementación Laravel	Danny Francisco Sapón Tax
Aplicación móvil Flutter	Danny Francisco Sapón Tax
Integraciones Firebase y Twilio	Danny Francisco Sapón Tax
Diseño de base de datos y API REST	Danny Francisco Sapón Tax
Infraestructura y despliegue	Danny Francisco Sapón Tax