

Universidad Tecnológica de Xicotepec de Juárez

Profesor: Héctor Valderrábanos Gonzales

Nombre del Proyecto: Alerta Ciudadana

Integrantes del equipo:

Jareni Gomez Juan

Griselda Cabrera Franco

José Daniel Loza Marín

Esaú Vargas Alvares

Abdiel Rivera Gayoso

1.Documento de Requisitos

1.1 Propósito:

Describir en términos del cliente los servicios y restricciones de Alerta Ciudadana, plataforma móvil y web para reporte y análisis de incidentes urbanos.

1.2 Contexto y Objetivos

- Contexto: Falta de canales ágiles para que ciudadanos reporten accidentes, vandalismo o fallas de infraestructura.
- Objetivos:
 - Permitir reportes geolocalizados con foto.
 - Proveer dashboard web para autoridades.
 - Generar estadísticas y alertas automáticas.

1.3 Visión Global del Sistema

- Frontera: App móvil (iOS/Android) + Plataforma web + Backend + Base de datos en nube.
- Interfaces externas:
 - API de mapas (Google / OpenStreetMap)
 - Servicio de notificaciones push / SMS

1.4 Requisitos Funcionales

ID	Descripción	Prioridad	Fuente
RF-001	Enviar reporte con foto, descripción y ubicación exacta.	Alta	Ciudadano
RF-002	Visualizar incidentes en mapa interactivo en tiempo real.	Alta	Ciudadano
RF-003	Validar y clasificar incidentes por tipo y urgencia.	Media	Admin Mun.
RF-005	Gestión de usuarios y permisos en plataforma web.	Media	Admin Mun.

1.5 Requisitos No Funcionales

ID	Descripción	Métrica

RNF-001	Cifrado de extremo a extremo en todas las comunicaciones.	TLS 1.2+
RNF-002	Disponibilidad mínima del 95 % en horas hábiles.	Uptime $\geq 95\%$
RNF-003	Tiempo de respuesta de la API menor a 300 ms en el 90 % de las peticiones.	$p90 \leq 300\text{ ms}$
RNF-004	Escalabilidad para 500 usuarios concurrentes.	Pruebas de carga con 500 threads
RNF-005	Tiempo de carga de la app móvil menor a 2 s en redes 4G.	Medido en campo con dispositivos reales

1.6 Restricciones

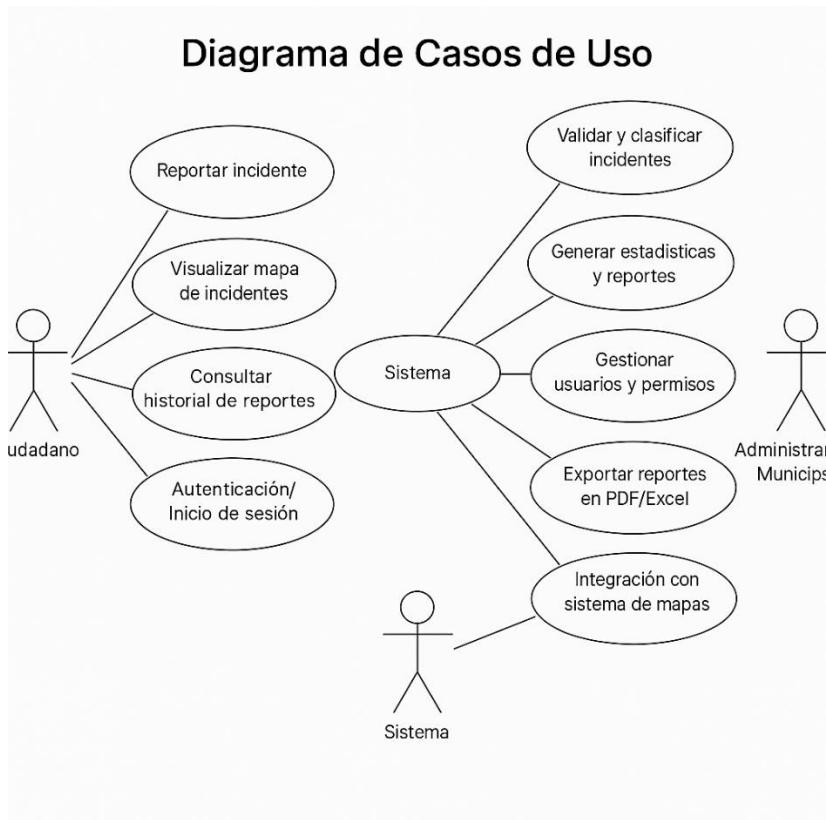
- Operar únicamente en México con geocoding de Google o OSM.
- Piloto inicial en una sola alcaldía.
- Cumplir leyes de protección de datos locales (LGPD).

1.7 Glosario

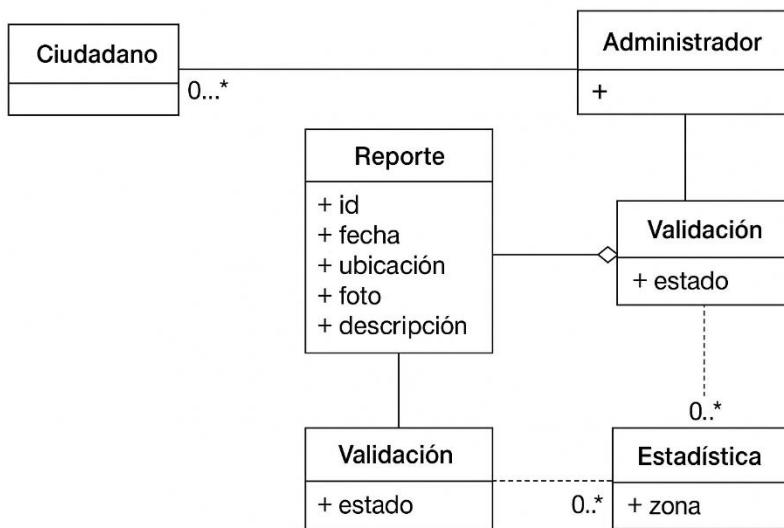
Término	Definición
Incidente	Evento urbano que afecta seguridad o infraestructura.
Ciudadano	Usuario final que reporta incidentes desde la app.
Administrador	Rol con permisos de validación, análisis y exportación de datos.

2. Modelo del Sistema

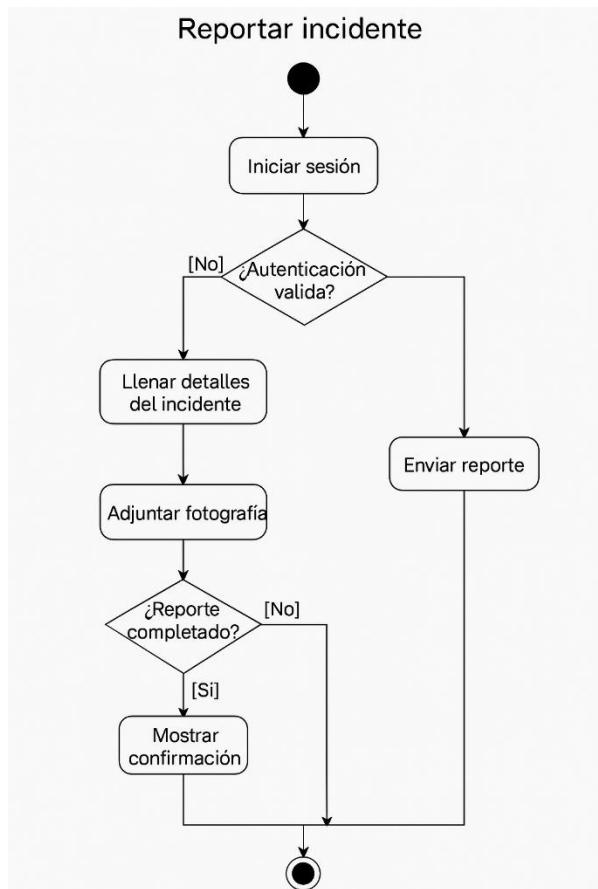
2.1 Diagrama de Casos de Uso



2.2 Diagrama de Clases



2.3 Diagrama de Actividad “Reportar incidente”



3. Especificación de Requisitos (IEEE 830)

3.1 Introducción

- Propósito del documento: Servir de base para diseño y pruebas.
- Alcance del sistema: App + web + backend para reporte y análisis de incidentes.
- Definiciones, acrónimos y abreviaciones: ver sección 1.7.

3.2 Descripción General

- Perspectiva del producto: Sistema independiente con GUI móvil y web.
- Funciones del producto: envío de reportes, visualización de datos, gestión de usuarios.
- Características de los usuarios:
 - Ciudadanos: interacción sencilla.
 - Admins: conocimientos medios de TI.

3.3 Requisitos Específicos

3.3.1 Requisitos Funcionales

RF-001: Autenticación de usuario

- Precondición: El ciudadano ya está registrado en el sistema.
- Flujo Principal:
 - El usuario abre la pantalla de “Inicio de sesión”.
 - Ingresá email y contraseña.
 - El sistema valida credenciales.
 - Si son correctas, concede acceso y redirige al dashboard.
- Flujos Alternos:
 - A1: Si el email no existe → muestra “Usuario no encontrado” → fin.
 - A2: Si la contraseña es incorrecta → muestra “Contraseña inválida” → vuelve a paso 2 (máximo 3 intentos).
- Poscondición: El usuario queda autenticado y con sesión activa.

RF-002: Envío de reporte de incidente

- Precondición: El usuario ha iniciado sesión.
- Flujo Principal:
 - El ciudadano selecciona “Nuevo reporte”.
 - Completa campos: descripción, tipo de incidente, ubicación (mapa o GPS).
 - Adjunta fotografía (opcional).
 - Pulsa “Enviar”.
 - El sistema guarda el reporte y muestra confirmación.
- Flujos Alternos:
 - A1: Si falta descripción o ubicación → alerta “Campos obligatorios” → vuelve a paso 2.

- A2: Si la foto excede peso permitido → muestra “Archivo muy grande” → vuelve a paso 3.
- A3: Si falla conexión al servidor → muestra “Error de red” → opción “Reintentar” o “Guardar en borrador”.
- Poscondición: Se genera un nuevo reporte con estado “Pendiente de validación”.

RF-003: Validación y clasificación de incidentes

- Precondición: El administrador municipal ha iniciado sesión en el portal web.
- Flujo Principal:
 - El admin accede a la lista de reportes pendientes.
 - Selecciona un reporte.
 - Revisa fotografía, descripción y ubicación.
 - Asigna categoría (vandalismo, falla de infraestructura, riesgo) y prioridad.
 - Marca “Validar” para actualizar estado a “Validado”.
 - El sistema notifica al ciudadano.
- Flujos Alternos:
 - A1: Si el reporte es falso/inválido → marca “Rechazar” y guarda observaciones → notifica al usuario.
- Poscondición: El reporte queda con estado “Validado” o “Rechazado” y categoría asignada.

RF-004: Generación y exportación de reportes estadísticos

- Precondición: El administrador está autenticado.
- Flujo Principal:
 - El admin abre sección “Estadísticas”.
 - Selecciona rango de fechas y filtros (zona, categoría).
 - El sistema muestra gráficas y tablas.
 - Pulsa “Exportar” y elige formato (PDF / Excel).
 - El sistema genera y descarga el archivo.
- Flujos Alternos:
 - A1: Si no hay datos en el rango → muestra “No hay datos para esos filtros”.
- Poscondición: Se descarga un archivo con la estadística solicitada.

RF-005: Gestión de usuarios y permisos

- Precondición: El administrador principal ha iniciado sesión y posee rol “Super-Admin”.
- Flujo Principal:
 - Accede a “Usuarios” en el menú.
 - Elige “Crear usuario” / “Editar” / “Eliminar”.
 - Para creación: ingresa nombre, email, rol (admin, analista, lector).

- Guarda cambios.
- El sistema actualiza la lista y notifica acción realizada.
- Flujos Alternos:
 - A1: Si el email ya existe → alerta “Usuario ya registrado” → fin.
 - A2: Si intenta eliminar al último super-admin → bloquea acción con mensaje.
- Poscondición: La lista de usuarios refleja la modificación (nuevo, editado o eliminado).

3.3.2 Requisitos No Funcionales

ID	Categoría	Descripción	Métrica / Validación
RNF-001	Seguridad	Todas las comunicaciones cliente-servidor deben usar TLS 1.2 o superior.	Escaneo de puertos y pruebas SSL
RNF-002	Disponibilidad	La plataforma debe tener un uptime mínimo del 95 % en horas hábiles.	Monitoreo con herramienta de uptime
RNF-003	Rendimiento	El 90 % de las llamadas a la API deben responder en ≤ 300 ms.	Pruebas de carga p90 (JMeter, Locust)
RNF-004	Escalabilidad	Soportar hasta 500 usuarios concurrentes sin degradar el rendimiento.	Pruebas de escalado (stress test)
RNF-005	Usabilidad	Tiempo de aprendizaje para un nuevo usuario ≤ 1 hora.	Test con usuarios reales
RNF-006	Mantención	El código debe estar documentado al 90 % y con cobertura de tests ≥ 80 %.	Revisión de documentación y cobertura
RNF-007	Compatibilidad	Compatible con las últimas 2 versiones de iOS y Android, y navegadores modernos.	Pruebas en dispositivos reales

3.4 Interfaces Externas

- API de Mapas: Endpoints para geocoding y visualización.
- Servicio de Notificaciones: Endpoint para envío de push/SMS.

3.5 Atributos de Calidad

- **Seguridad:** cifrado en tránsito y reposo.
- **Usabilidad:** tiempo de aprendizaje < 1 hora.
- **Mantenibilidad:** código documentado al 90 %.

4. Formato de Validación de Requisitos

ID	Requisito	Método de Validación	Resultado	Observaciones
RF-001	Autenticación de usuario	Pruebas de integración	✓	
RF-002	Envío de reporte con ubicación y foto	Prueba en dispositivo real	✓	Latencia < 2 s en 4G
RF-003	Validación y clasificación de incidentes	Revisión de flujo de trabajo	✓	
RF-004	Exportación de reportes en PDF/Excel	Prueba de export y formatos	✓	PDF generado sin errores
RF-005	Gestión de usuarios y permisos	Caso de prueba de roles	✓	Usuario admin / ciudadano OK
RNF-001	Cifrado TLS 1.2+	Escaneo de puertos	✓	TLS activo, puertos cerrados
RNF-002	Disponibilidad $\geq 95\%$	Monitoreo con uptime robot	✓	Promedio mensual 97 %
RNF-003	p90 API ≤ 300 ms	Pruebas de carga	✓	p90 = 250 ms