

Especificación de Requerimientos del Sistema de Gestión de Incidentes Urbanos

Equipo: Capibaras vs Labubus 2

Integrantes:

- Barrientos Sánchez José Antonio
- Figueroa Barrientos Andrea Valeria
- Morales Chaparro Gael Antonio
- Ortiz Cervantes Leonardo Rafael
- Rivera Lara Sandra Valeria

1. Introducción

1.1 Propósito

El presente documento especifica los requerimientos funcionales y no funcionales de la **Aplicación Web para el Registro y Gestión de Incidentes Urbanos**.

El propósito del proyecto es crear un sistema que permita a los ciudadanos registrar, visualizar y gestionar incidentes urbanos (como baches, luminarias descompuestas, obstáculos en la vía pública, entre otros). Se busca brindar una herramienta que permita una mejor gestión de problemas que aquejan a la comunidad mediante la participación de la ciudadanía y el gobierno, con el fin de mejorar la infraestructura urbana.

1.2 Alcance

El sistema permitirá a los usuarios reportar incidentes urbanos, marcando su ubicación en un mapa interactivo, adjuntando fotografías y proporcionando una breve descripción. Además, cualquier usuario podrá actualizar el estado de un incidente a **"En proceso"** o **"Resuelto"**. La aplicación mostrará todos los incidentes en un mapa categorizados según su tipo y estado.

1.3 Definiciones y Abreviaciones

- **Usuario:** Cualquier persona que utiliza la aplicación para reportar, actualizar o buscar incidentes y comentar en las publicaciones. Así mismo puede hacer reportes hacia otros usuarios.
- **Incidente:** Reporte de un problema urbano (ej.: bache, luminaria, obstáculo).
- **Administrador:** Usuario con permisos especiales.
 - **Permisos especiales:** Permitir borrar publicaciones de incidentes y gestionar cuentas de los usuarios no administradores (poder darlas de baja).
- **Estado:** Situación actual del incidente.
 - **Reportado:** Fue publicado con evidencia (fotos) y hasta el momento ningún ciudadano o autoridad ha publicado un avance en la aplicación.
 - **En proceso:** Algún usuario publicó evidencia (fotos) de que el incidente está siendo tratado.
 - **Resuelto:** Algún usuario publicó evidencia (fotos) de que el incidente ya está completamente arreglado y no genera ningún riesgo o molestia para la ciudadanía.

Tipo de Incidente

- **Automovilístico**
 - Accidentes de tránsito que involucren vehículos o peatones.
 - Vehículos obstruyendo la vía tras un choque.
 - Daños en infraestructura causados por colisiones.
 - **Infraestructura Vial**
 - Baches, grietas o hundimientos en el pavimento.
 - Falta o deterioro de señalización vial y semáforos averiados.
 - Puentes, banquetas o pasos peatonales en mal estado.
 - **Medio Ambiente y Fenómenos Naturales**
 - Árboles caídos o en riesgo de caída.
 - Inundaciones.
 - Fugas de agua, drenaje obstruido o contaminación en espacios públicos.
 - **Arquitectura**
 - Daños en edificios (grietas o estructura interna en mal estado).
 - Daños estructurales en edificaciones de transporte público.
 - **Otros (Manifestaciones, Ferias, Eventos)**
 - Concentraciones de personas por marchas o protestas.
 - Eventos públicos que bloqueen calles o espacios urbanos.
 - Instalaciones temporales como ferias o mercados ambulantes.
-

2. Requerimientos del Sistema

2.1 Requerimientos Funcionales

- **RF1: Registro de Incidentes**
 - Permitir a los usuarios reportar incidentes.
 - **Ubicación:** Los usuarios deben marcar la ubicación del incidente en un mapa interactivo.
 - **Adjuntar evidencia:** Permitir adjuntar una o más fotografías.
 - **Descripción:** Incluir una breve descripción del problema.
- **RF2: Actualización del Estado de Incidentes**
 - Permitir a cualquier usuario actualizar el estado del incidente a **"En proceso"** o **"Resuelto"**.
 - Permitir que el usuario dueño de una publicación de incidente pueda eliminarla solamente cuando esta no se encuentre en el estado de **"Reportado"**.
 - Requerir evidencia fotográfica cuando se marque el incidente como **"En proceso"** o **"Resuelto"**.
 - **IDescripción:** Incluir una breve descripción de la actualización del incidente.
- **RF3: Visualización de Incidentes**
 - Mostrar los incidentes en un mapa interactivo.
 - Permitir filtros por tipo de incidencia, ubicación, estado y filtros personalizados.
- **RF4: Gestión de Usuarios y Roles**
 - Registro, creación y autenticación de usuarios.
 - Gestión de perfiles, incluyendo la asignación de roles (Usuario y Administrador).

- Permitir que los usuarios puedan reportar a otros usuarios con comentarios del por qué.
- Permitir que los administradores puedan dar de baja publicaciones y cuentas de usuarios.

2.2 Requerimientos No Funcionales

- **RNF1: Seguridad**

- Cifrado de contraseñas y datos sensibles.
- Implementación de inicio de sesión con Google y autorización.
- **Uso de protocolo HTTPS** para garantizar la seguridad de la comunicación entre clientes y servidores.
- Uso de contraseñas seguras (mínimo de caracteres requeridos).
- Solo usuarios con cuenta pueden registrar incidentes.

- **RNF2: Rendimiento**

- Soportar aproximadamente 100 usuarios simultáneos.
- Garantizar que las búsquedas y actualizaciones en el sistema se realicen en menos de 2 segundos en promedio.

- **RNF3: Disponibilidad**

- Asegurar una disponibilidad del sistema del 99% o superior.

- **RNF4: Usabilidad**

- Interfaz intuitiva y accesible para los usuarios.
- Diseño responsive para facilitar el uso en dispositivos móviles.

3. Casos de Uso

CU1: Registro de un Incidente

Actores: Usuario

Flujo Principal:

1. El usuario inicia sesión en la aplicación.
2. Selecciona la opción **"Reportar Incidente"**.
3. Marca la ubicación del incidente en el mapa interactivo.
4. Adjunta una o más fotografías.
5. Escribe una breve descripción del problema.
6. Envía el reporte y el sistema registra el incidente con estado **"Reportado"**.

CU2: Actualización del Estado de un Incidente

Actores: Usuario, Administrador

Flujo Principal:

1. El usuario o administrador inicia sesión en la aplicación.
2. Selecciona un incidente previamente reportado.

3. Cambia el estado a **"En proceso"** o **"Resuelto"**.
4. Se solicita adjuntar evidencia fotográfica.
5. El sistema actualiza el estado del incidente.

Flujo Alternativo:

- Si no se adjunta la evidencia al cambiar a "Resuelto", el sistema muestra un mensaje de error y solicita la prueba correspondiente.

CU3: Visualización y Filtro de Incidentes

Actores: Usuario, Administrador

Flujo Principal:

1. El usuario accede a la sección de visualización de incidentes.
2. El sistema despliega un mapa interactivo con todos los incidentes registrados.
3. El usuario aplica filtros (por tipo de incidente y estado).
4. El sistema actualiza la vista con los incidentes que cumplen los criterios de búsqueda.

CU4: Reporte de Incidente Falso o Duplicado

Actores: Usuario, Administrador **Flujo Principal:**

1. El usuario inicia sesión y accede a un incidente.
2. El sistema despliega el mapa interactivo.
3. El usuario selecciona el reporte.
4. Selecciona la opción "Reportar incidente falso o duplicado".
5. Proporciona una justificación o enlace a un incidente similar.
6. El sistema notifica a un administrador para su revisión.
7. Si el administrador confirma que el incidente es falso o duplicado, lo marca como tal y lo elimina o fusiona con otro reporte.

Flujo Alternativo:

- Si el incidente no se considera falso o duplicado, el administrador lo deja activo y notifica al usuario que reportó.

4. Requisitos de Hardware y Software

4.1 Hardware

- **Servidor:**
 - Mínimo 4GB de RAM.
 - Procesador Quad-Core.
 - Almacenamiento en disco SSD para mejorar la velocidad de acceso a la base de datos.

4.2 Software y Tecnologías Recomendadas

- **Frontend:**
 - **React:** Para construir una interfaz de usuario dinámica y responsiva.

- **TypeScript:** Para mejorar la robustez y mantenibilidad del código.
 - **Backend:**
 - **Kotlin:** Para el desarrollo de la API y la lógica del servidor.
 - **Spring Boot:** Framework para facilitar el desarrollo y la configuración del backend.
 - **Base de Datos:**
 - **MongoDB:** Base de datos NoSQL ideal para manejar datos flexibles y escalables relacionados con los incidentes.
 - **Mapas:**
 - **API de Google:** Para integrar mapas interactivos con información geolocalizada precisa.
 - **Gestión de Proyectos:**
 - **Trello:** Para organizar tareas, planificar sprints y gestionar el progreso del proyecto.
-

5. Consideraciones Finales

Este documento establece los requerimientos iniciales para la **Aplicación Web para el Registro y Gestión de Incidentes Urbanos**. Cualquier modificación o ampliación deberá ser documentada y aprobada por los interesados.