

Documentación 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

Historial de Cambios

Versión	Fecha	Descripción	Autor
1.0.0	25 de marzo	<p>Esta iteración se centró en implementar los casos de uso básicos de usuario, cubriendo el ciclo completo de autenticación y gestión de datos personales para garantizar una experiencia fluida.</p> <p>Casos de usos implementados:</p> <ul style="list-style-type: none">- CU9: Registro de Usuario (Sign Up)- CU10: Obtener Información de Usuario- CU11: Inicio de Sesión (Login)- CU12: Cierre de Sesión (Logout)	Todos los miembros de Capibaras vs Labubu 2

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 un registro de incidente.
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. (CU6)

Flujo Alternativo:

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

CU5: Archivar Incidente Resuelto

Actores: Administrador

Flujo Principal:

1. El administrador inicia sesión en la aplicación.
2. Accede a la lista de incidentes marcados como **"Resuelto"**.
3. Selecciona un incidente y verifica que cumple con los criterios de resolución.
4. Elige la opción **"Archivar Incidente"**.
5. El sistema cambia el estado del incidente a **"Archivado"** y lo oculta del mapa interactivo.

Flujo Alternativo:

- Si el incidente no está completamente resuelto, el administrador puede revertir el estado a **"En proceso"** y notificar al usuario correspondiente.

CU6: Eliminar Incidente Falso o Duplicado

Actores: Administrador

Flujo Principal:

1. El administrador inicia sesión y revisa los incidentes reportados como **"Falso o Duplicado"**.
2. Selecciona un incidente y evalúa la justificación proporcionada.
3. Si confirma que es inválido, elige la opción **"Eliminar Incidente"**.
4. El sistema solicita confirmación y, tras aceptar, elimina permanentemente el registro.
5. Se notificará al usuario que ingresó el incidente y al usuario que lo reportó como **Falso o Duplicado**, indicando que el incidente fue borrado.

Flujo Alternativo:

- Si el incidente es válido, el administrador puede marcarlo como **"Confirmado"** y notificar al usuario que lo reportó como falso.

CU7: Reportar Usuario por Contenido Inadecuado

Actores: Usuario, Administrador

Flujo Principal:

1. El usuario inicia sesión en la aplicación.
2. Accede al mapa interactivo y selecciona un incidente reportado por otro usuario.
3. Si identifica contenido inadecuado, selecciona la opción **"Reportar Usuario"**.
4. El sistema despliega un formulario con:
 - Menú desplegable para seleccionar motivo (**"Contenido indebido"**, **"Spam"**).
 - Campo de comentario (máx. 200 caracteres). El usuario debe llenar obligatoriamente este campo (al menos un carácter).
5. El usuario envía el reporte.
6. El sistema:
 - Registra el reporte.
 - Envía una notificación automática al administrador para su revisión.
 - Muestra una notificación: **"Reporte enviado. Gracias por colaborar"**.

Flujo Alternativo:

- Si cancela el reporte, el sistema regresa a la vista del incidente sin guardar cambios.

CU8: Gestionar Usuarios Reportados

Actores: Administrador

Flujo Principal:

1. El administrador inicia sesión y accede a la sección **"Usuarios Reportados"**.
2. El sistema muestra una lista de usuarios con reportes pendientes, incluyendo:
 - Nombre del usuario reportado.
 - Motivo del reporte.
 - Incidente asociado (con enlace para revisarlo).
3. El administrador selecciona un usuario y evalúa el contenido:
 - **Si es contenido indebido** (por ejemplo, ofensivo o ilegal):
 - Selecciona **"Dar de baja usuario"**.
 - El sistema bloquea al usuario (cuando el usuario inicie sesión verá el motivo por el bloqueo, según CU11-FA3).
 - Se notificará al usuario que reportó al segundo usuario indicando que el segundo usuario fue dado de baja.
 - El sistema guía al administrador para dar de baja la publicación del incidente asociado (CU6).
 - **Si es otro tipo de contenido** (por ejemplo, error honesto):
 - El sistema registra el incidente en la lista de incidentes reportados y redirige al administrador para **"Eliminar publicación"**, siguiendo CU6.
 - Se notificará tanto al usuario reportado como al que hizo el reporte indicando que el incidente fue borrado.

Flujo Alternativo:

- Si el administrador considera que el reporte es infundado, puede marcarlo como **"Rechazado"** y archivar el caso.

CU9: Registro de Usuario

Actores: Usuario (no registrado)

Flujo Principal:

1. El usuario ingresa a la aplicación y selecciona **"Registrarse"**.
2. El sistema muestra un formulario con campos:
 - Correo electrónico.
 - Contraseña.
3. El usuario introduce sus datos y envía el formulario.
4. El sistema valida que:
 - El correo no esté registrado previamente.
5. Si todo es correcto, el sistema:
 - Crea un nuevo usuario en la base de datos.
 - Muestra mensaje: **"Registro exitoso. ¡Bienvenido/a!"**.

Flujos Alternativos:

- **FA1: Correo ya registrado:**
Si el sistema detecta que el correo existe, muestra un mensaje de error.
- **FA2: Error de conexión:**
Si hay fallos de red, muestra: **"Error de conexión. Intenta nuevamente"**.

CU10: Obtener Información de Usuario

Actores: Usuario registrado, Administrador

Precondición: El usuario debe tener una sesión activa con un token válido.

Flujo Principal:

1. El usuario accede a la sección **"My account"**.
2. El sistema solicita el token de autenticación almacenado.
3. El sistema valida el token y busca al usuario en la base de datos.
4. Si la información es encontrada, el sistema muestra:
 - Datos básicos del usuario (nombre, email).
 - Mensaje: **"Información cargada correctamente"**.

Flujos Alternativos:

- **FA1: Token nulo o inválido:**
Si el token es nulo o inválido, muestra un error y redirige al usuario a la pantalla de login.
- **FA2: Usuario no encontrado:**
Si el token es válido pero no corresponde a ningún usuario, muestra un error.
- **FA3: Error de conexión:**
Si hay fallos al consultar la base de datos, muestra un mensaje de error.

CU11: Inicio de Sesión de Usuario

Actores: Usuario registrado

Flujo Principal:

1. El usuario ingresa a la aplicación y selecciona **"Log in"**.
2. El sistema muestra un formulario con campos:
 - Correo
 - Contraseña
3. El usuario introduce sus datos y envía el formulario.
4. El sistema:
 - Busca al usuario en la base de datos.
 - Genera un nuevo token de autenticación.
 - Actualiza el token del usuario.
 - Muestra mensaje de inicio exitoso y permite el acceso a las funcionalidades según su rol.

Flujos Alternativos:

- **FA1: Usuario no encontrado:**
Si el correo o contraseña no coinciden, el sistema muestra error de usuario no encontrado. Se permite al usuario ingresar los datos otra vez.

- **FA2: Error de conexión:**

Si falla la comunicación con la base de datos, el sistema muestra error de conexión.

- **FA3: Usuario bloqueado:**

Si el sistema identifica que el usuario fue bloqueado por reportes indebidos en la app, se le muestra un mensaje explicando por qué fue bloqueado y se limita su acceso a la app.

CU12: Cierre de Sesión (Logout)

Actores: Usuario autenticado

Precondición: El usuario debe tener una sesión activa en el sistema.

Flujo Principal:

1. El usuario selecciona la opción "**Log out**" desde cualquier pantalla de la aplicación.
2. El sistema recibe la solicitud con el token de autenticación actual.
3. El sistema valida el token y busca al usuario asociado.
4. Si el token es válido, el sistema invalida el token actual.
5. Muestra mensaje: "**Sesión cerrada correctamente**" y redirige al usuario a la pantalla de inicio de sesión.

Flujos Alternativos:

- **FA1: Token inválido/usuario no encontrado:**

Si el token no corresponde a ningún usuario activo, el sistema muestra error de sesión no válida. Elimina los datos de sesión local y redirige al login.

- **FA2: Error de conexión:**

Si hay problemas para comunicarse con la base de datos, muestra: "**Error: No se pudo completar la acción. Intente nuevamente**".

Permite reintentar el cierre de sesión.

CU13: Actualización de Datos de Usuario

Actores: Usuario registrado, Administrador

Precondición:

El usuario debe tener una sesión activa con un token válido.

Flujo Principal:

1. El usuario selecciona la sección "**Update user details**".
2. El sistema muestra un formulario editable con sus datos actuales (email, contraseña, etc.).
3. El usuario modifica uno o más campos y confirma los cambios.
4. El sistema:
 - Valida el token de sesión.
 - Verifica que el nuevo email (si se modifica) no esté registrado por otro usuario.
5. Si todo es válido, actualiza los datos en la base de datos.
6. Muestra mensaje: "**Tus datos se han actualizado correctamente**".

Flujos Alternativos:

- **FA1: Email ya en uso:**

Si el nuevo email existe en otro usuario, el sistema muestra: "**Error: Este correo ya está registrado**".

Use otro diferente" y mantiene los datos originales.

- **FA2: Usuario no encontrado:**

Si el token no corresponde a un usuario válido, muestra: "**Error: Sesión inválida. Será redirigido al login**" y cierra la sesión automáticamente.

- **FA3: Error de conexión:**

Si falla la comunicación con la base de datos, muestra: "**Error: No se pudieron guardar los cambios. Intente más tarde**" y no se hacen modificaciones.

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.

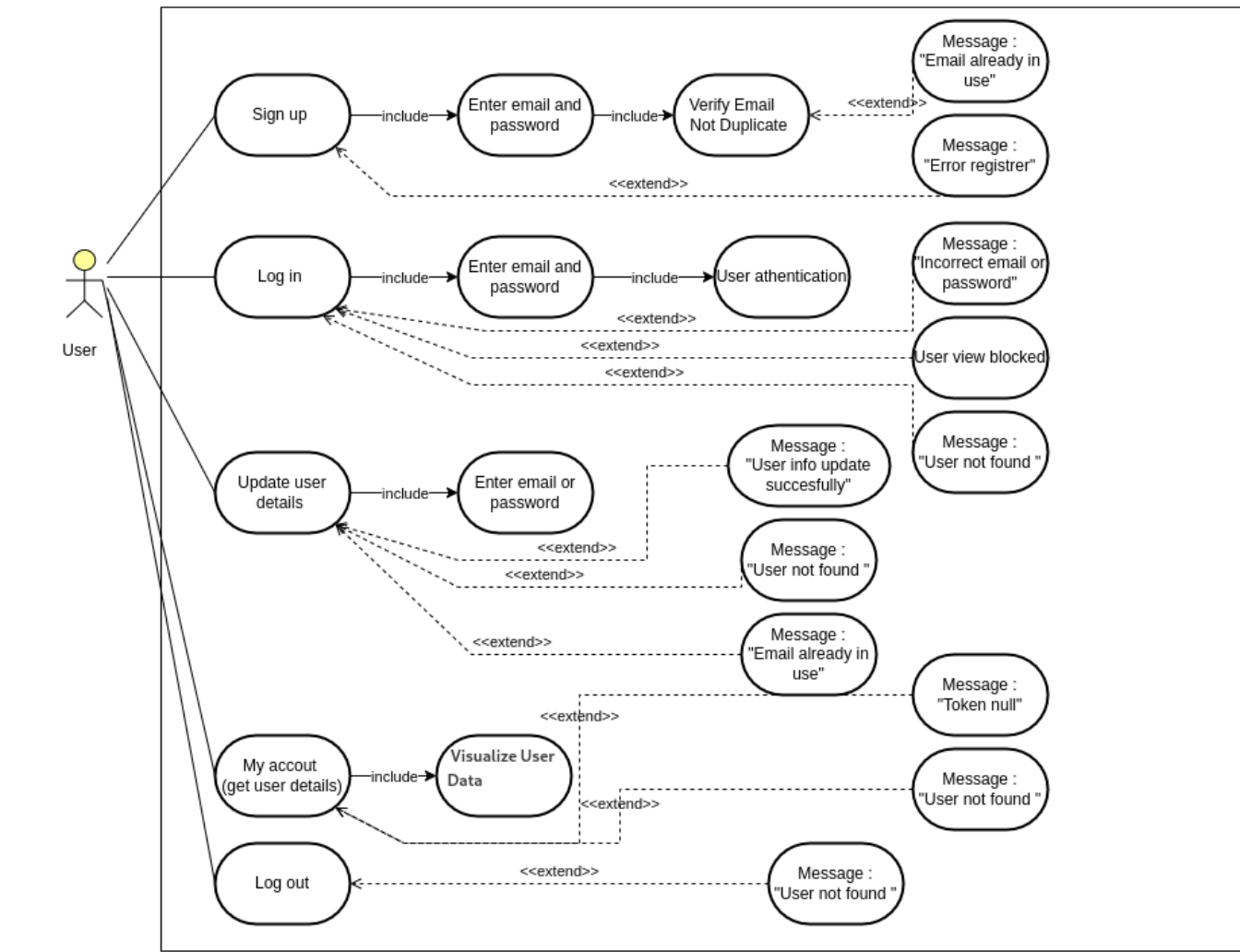
4.2 Software y Tecnologías Recomendadas

- **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.
-

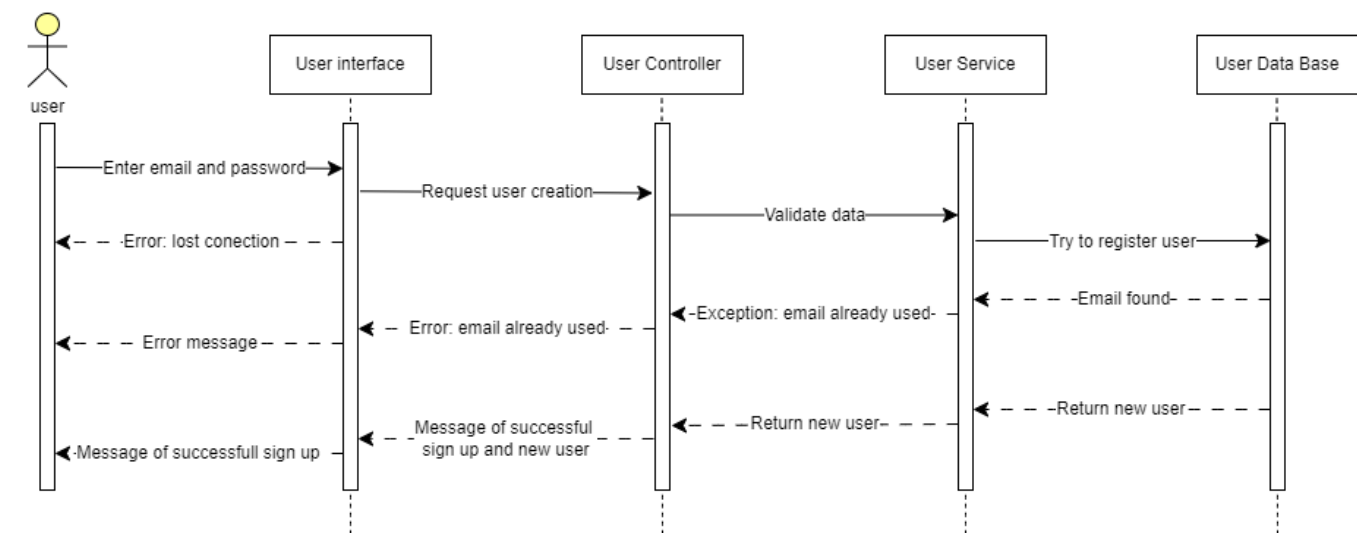
5. Diagramas de Casos de Uso

5.1 Casos de Uso Usuario

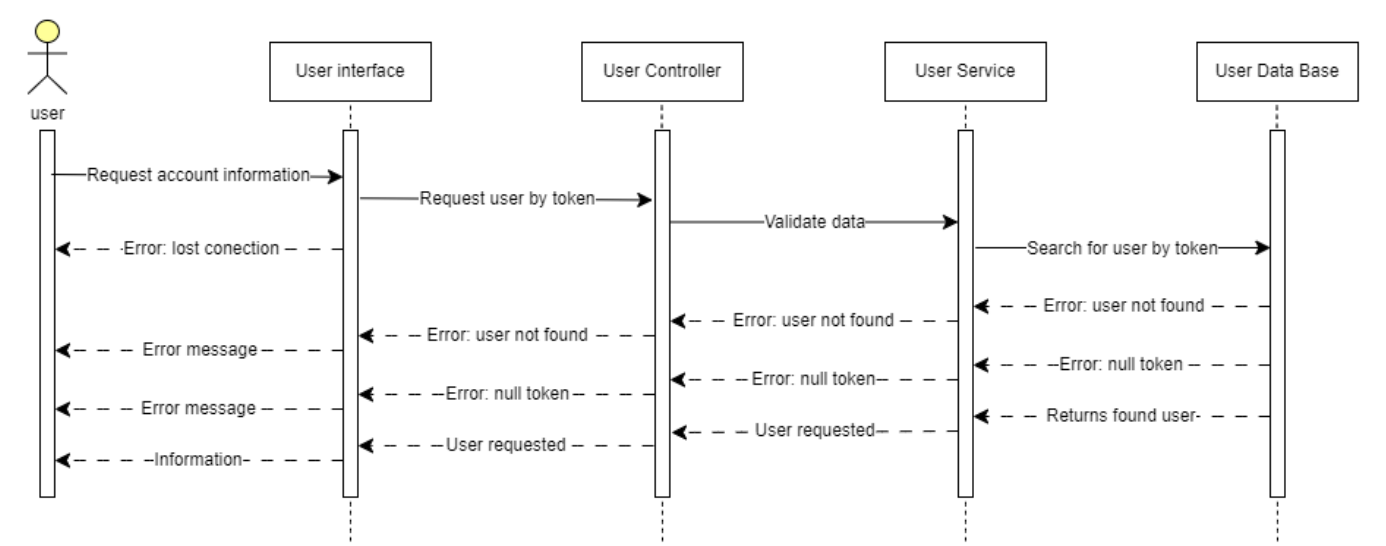
User Functionalities:



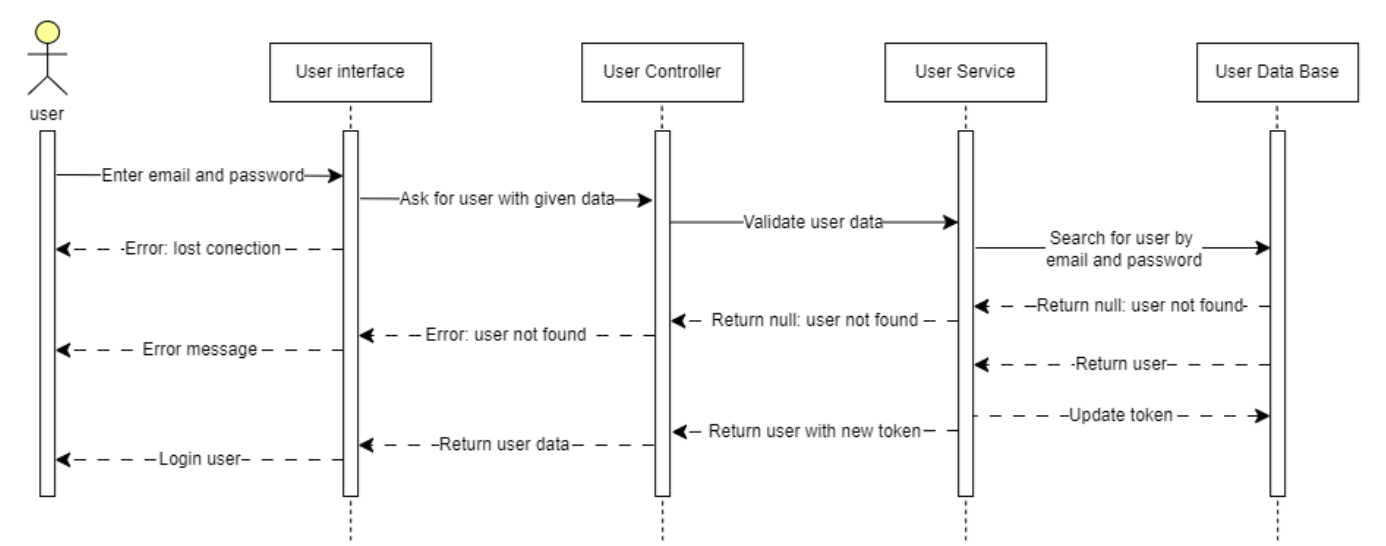
Create User:



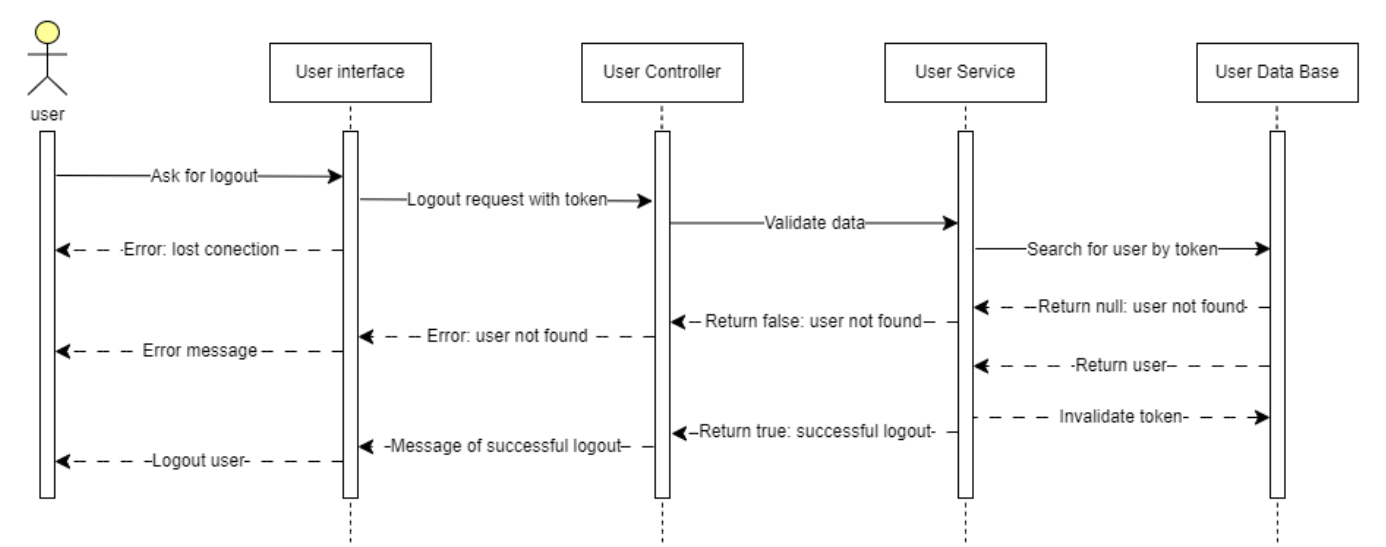
Get User:



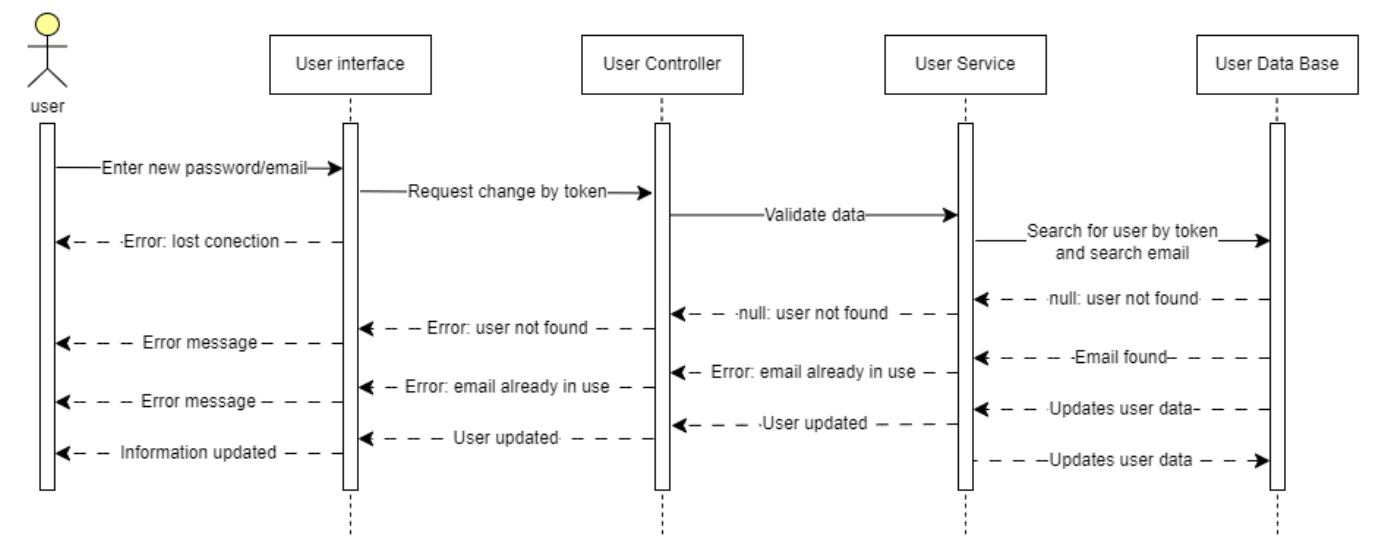
Login User:



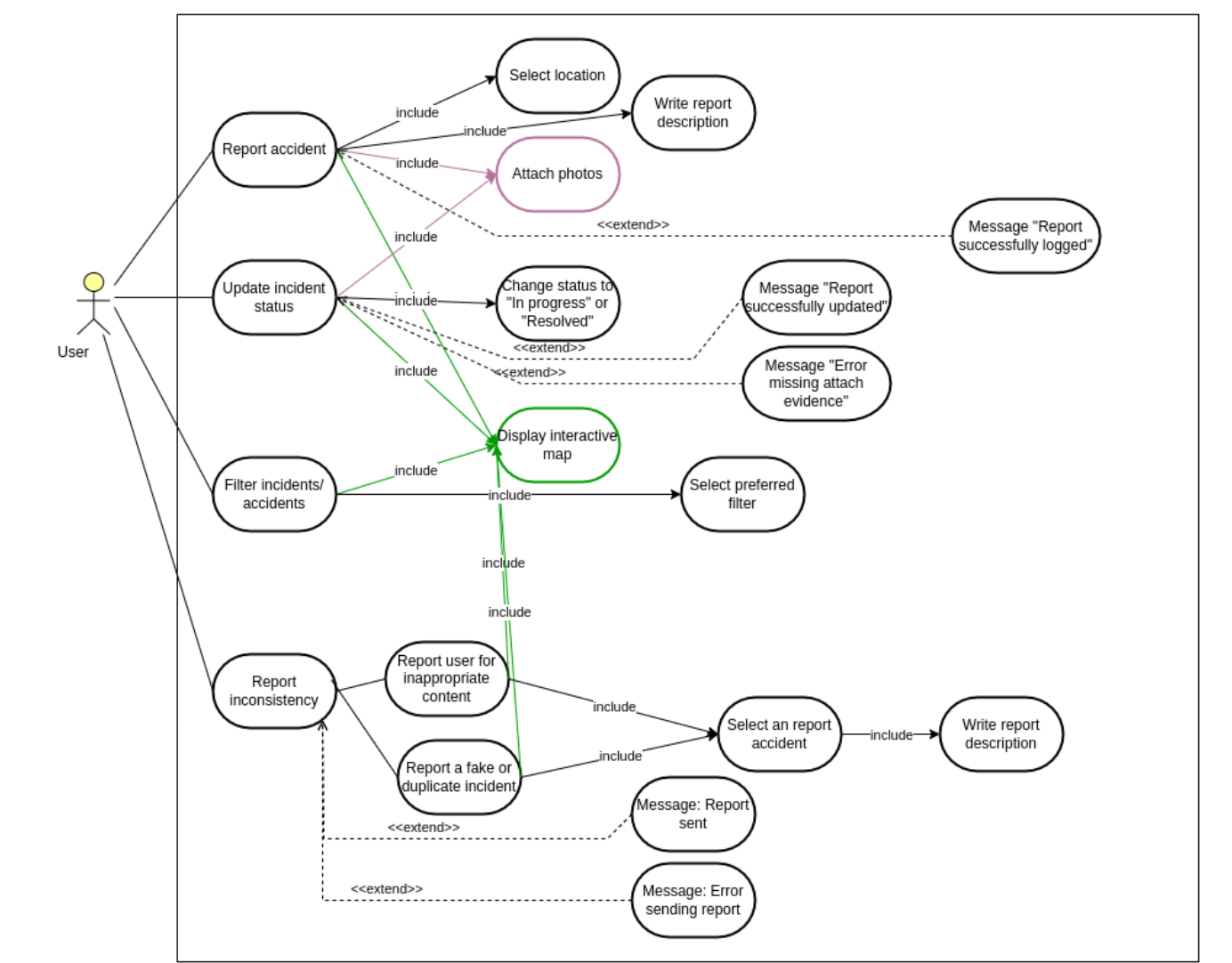
Logout User:



Update User:

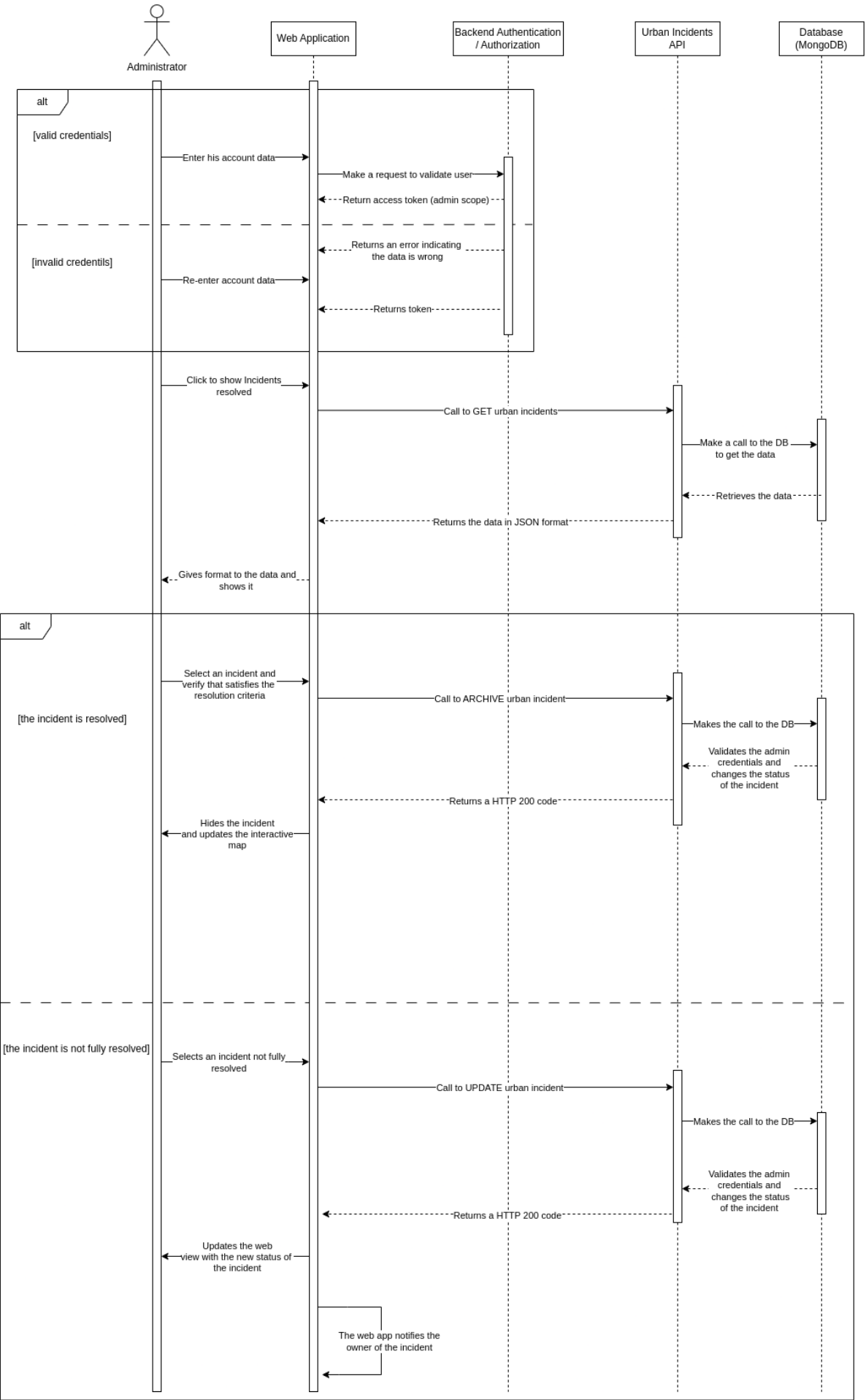


Report Accident Functionalities:

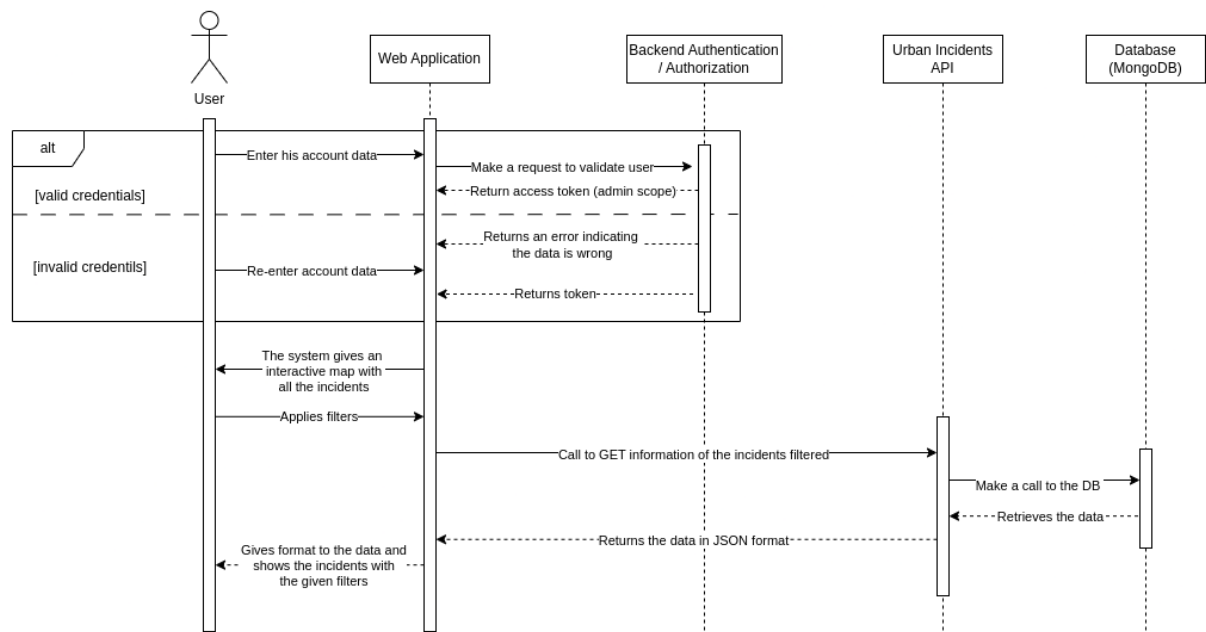


5.2 Casos de Uso Plataforma de Incidentes Urbanos

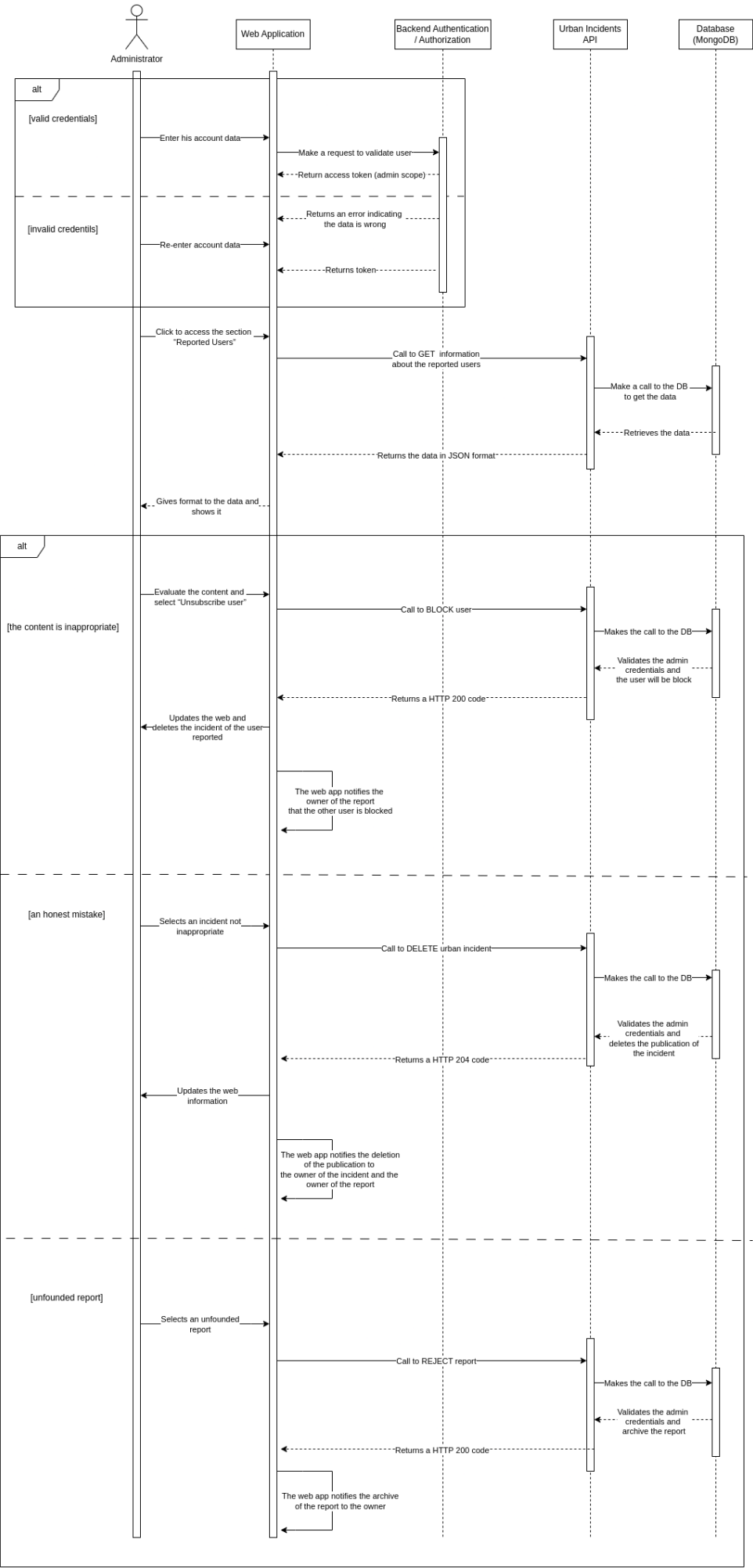
Archive Solved Incident:



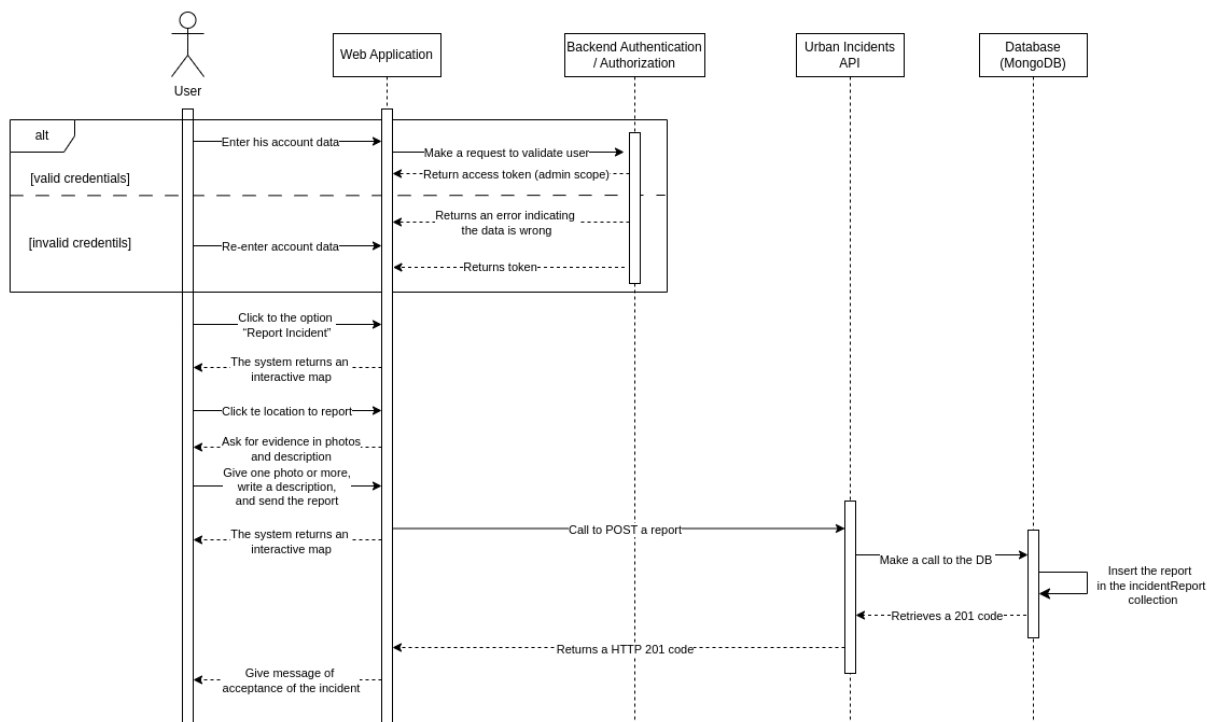
Incident Visualization And Filtering:



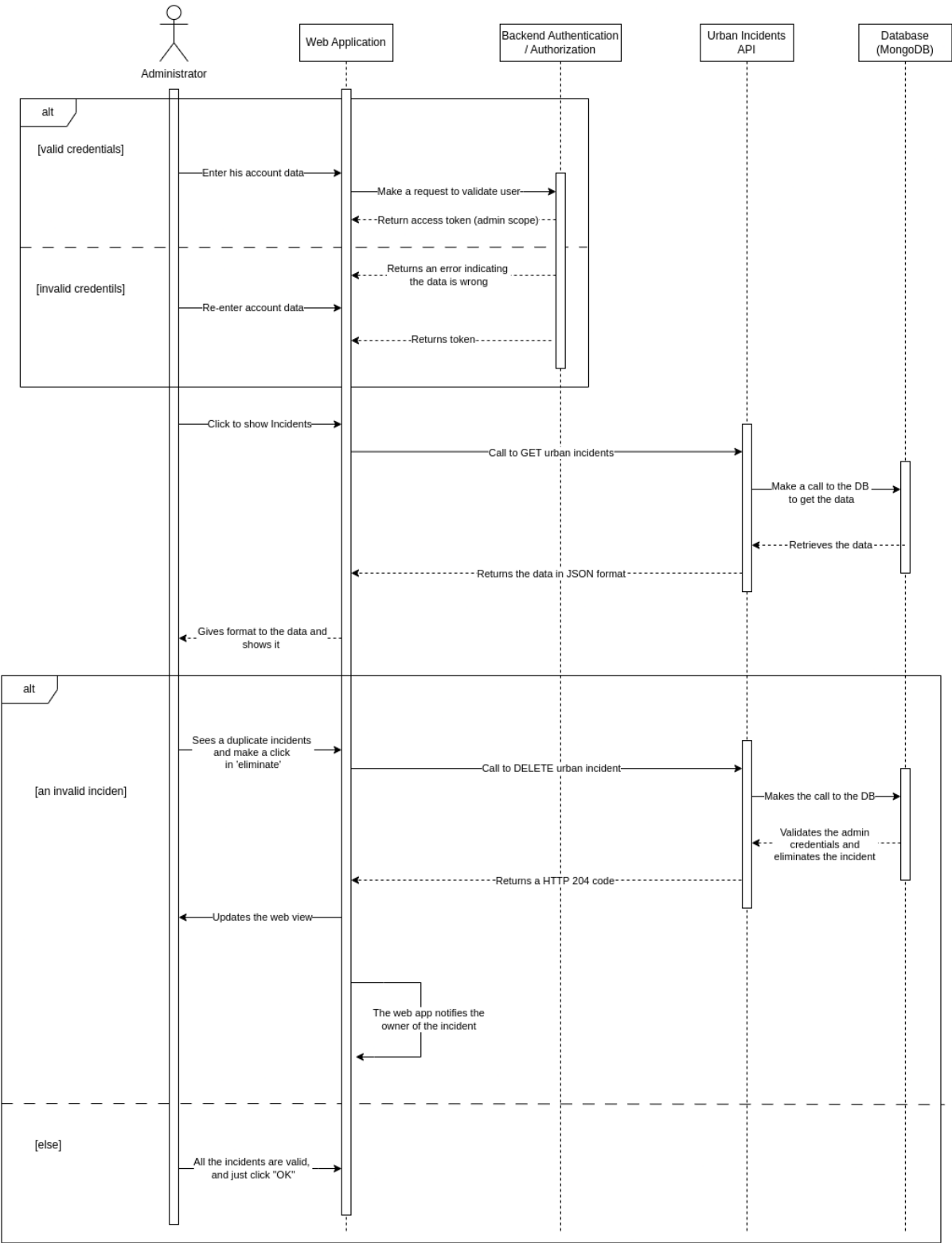
Manage Reported Users:



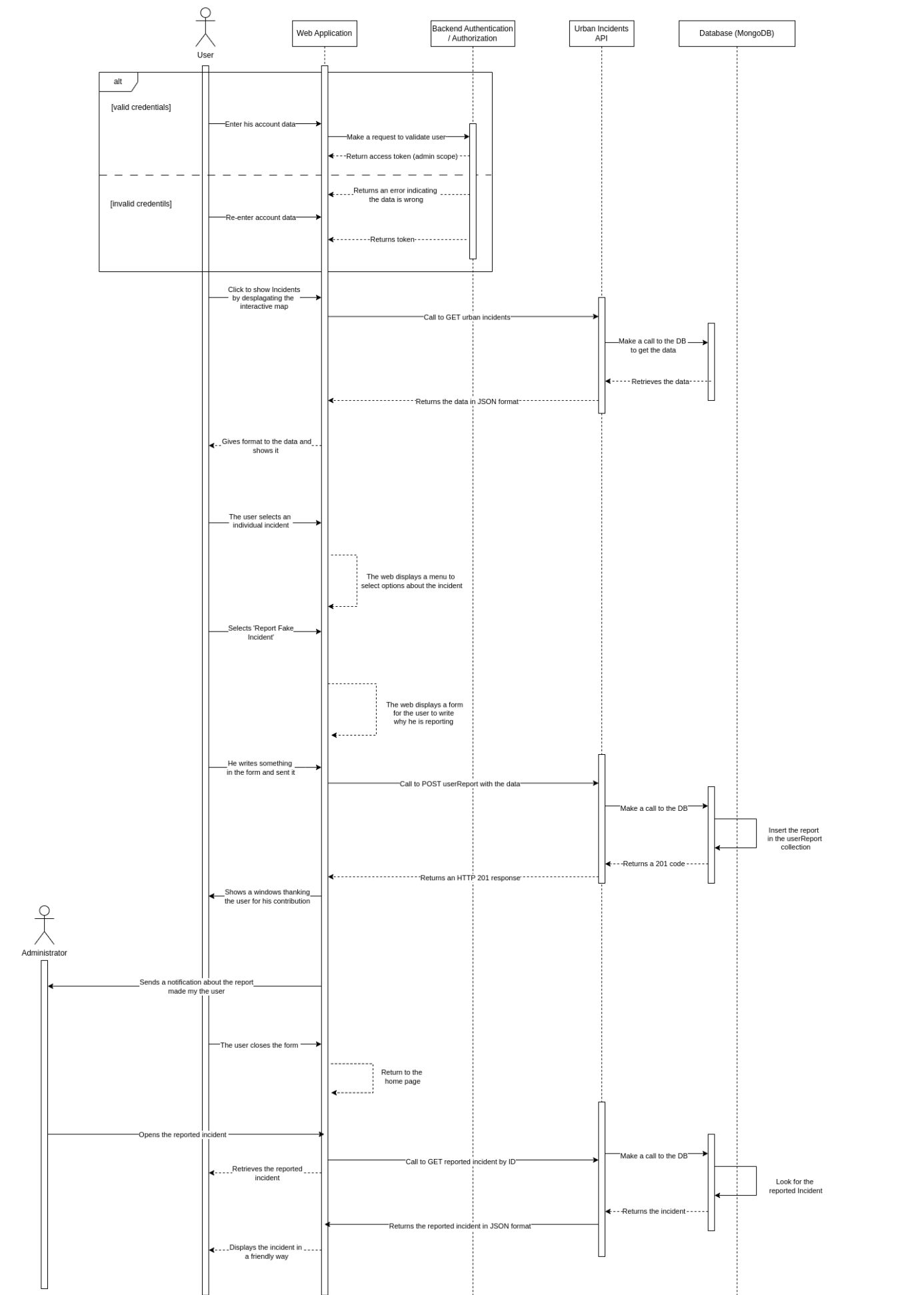
Register An Incident:

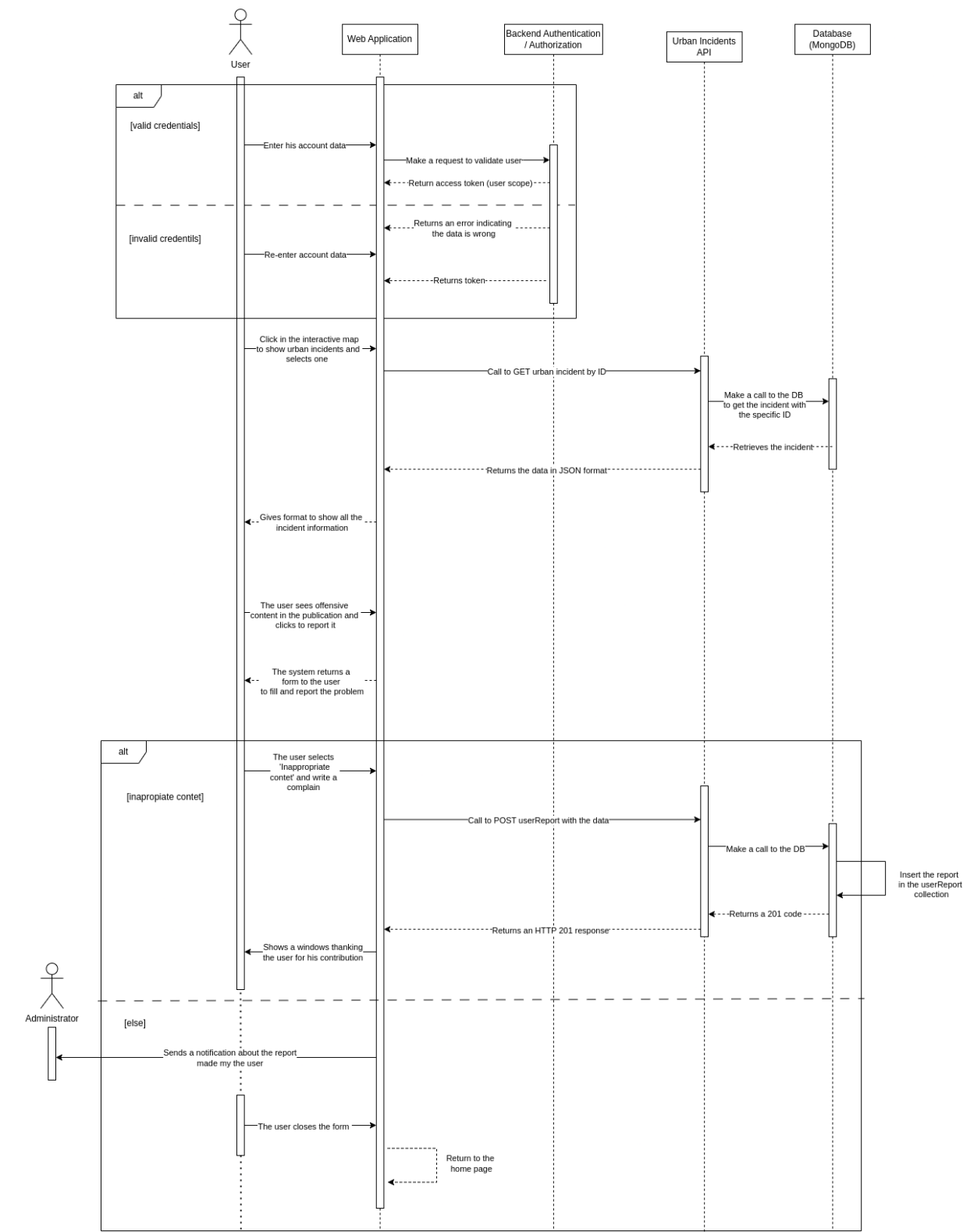


Remove Duplicate Incident:



Report Fake Incident:





Update Incident State:

