“OBJETOS PERDIDOS”

Un dibujo en blanco y negro

Descripción generada automáticamente con confianza baja

ENTREGA Nº6

Grupo Nº6

Integrantes

|  |  |  |
| --- | --- | --- |
| Legajo | Nombre | E-Mail |
| 27845 | Alexis Juan Carlos Torrente | alexis.torrente@alumnos.frgp.utn.edu.ar |
| 29057 | Francisco Ciarallo | francisco.ciarallo@alumnos.frgp.utn.edu.ar |
| 27853 | Francisco Crestanello | francisco.crestanello@alumnos.frgp.utn.edu.ar |
| 26268 | Gastón Sosa | gaston.sosa@alumnos.frgp.utn.edu.ar |
| 25756 | Ricardo Recofsky | obdulio.adducci@alumnos.frgp.utn.edu.ar |
| 29056 | Sebastián Ré | sebastian.re@alumnos.frgp.utn.edu.ar |

Fecha de entrega: 16/09/2023

Contenido

[**Propuesta: Objetos Perdidos** 3](#_Toc145796260)

[Alcance 3](#_Toc145796261)

[Objetivo mensurable 4](#_Toc145796262)

[Motivo 4](#_Toc145796263)

[Módulos funciones 4](#_Toc145796264)

[Funcionalidad: 4](#_Toc145796265)

[Perfiles 5](#_Toc145796266)

[Relevamiento 5](#_Toc145796267)

[Creación de cuenta 5](#_Toc145796268)

[Tipos de publicaciones en los Foros 6](#_Toc145796269)

[Navegación 6](#_Toc145796270)

[Mensajería interna 7](#_Toc145796271)

[Avisos y Notificaciones 7](#_Toc145796272)

[Seguridad y Privacidad 7](#_Toc145796273)

[Roles de Usuarios de la aplicación 7](#_Toc145796274)

[Mapas 8](#_Toc145796275)

[Reportes 8](#_Toc145796276)

[Comentarios en los perfiles 8](#_Toc145796277)

[Contacto 8](#_Toc145796278)

[Módulos funcionales 9](#_Toc145796279)

[Contribución social 11](#_Toc145796280)

[Plataforma de desarrollo e implementación 11](#_Toc145796281)

[Requerimientos funcionales 12](#_Toc145796282)

[Diagrama de análisis de casos de uso 22](#_Toc145796283)

[Especificación de casos de uso 23](#_Toc145796284)

[Diagramas de Secuencia 33](#_Toc145796285)

[Diagrama de actividades 45](#_Toc145796286)

[Diagrama Entidad Relación 66](#_Toc145796287)

[Prototipo de pantallas 67](#_Toc145796288)

[Informe de avance 91](#_Toc145796289)

# **Propuesta: Objetos Perdidos**

Aplicación móvil que permitirápublicar y encontrar objetos perdidos para devolver o recuperar.

## Alcance

El alcance es para todo público.

Se busca que los vecinos dispongan de una aplicación para su teléfono móvil con mínimamente las diferentes funcionalidades:

* Foro de solicitud de objeto: en donde se posteará el objeto perdido y toda la información relacionada a él (hora, lugar y fecha de cuando se perdió; descripción del objeto).
* Foro de objetos encontrados: en donde se postearía el objeto encontrado en la calle y toda la información relacionada a él (hora, lugar y fecha de cuando se encontró; descripción del objeto).  
  Ambos foros estarían subdivididos en municipios y tanto moderadores como administradores podrán modificar o dar de baja dichas publicaciones si incumplen las normas.
* Sistema de mensajería para que los usuarios se puedan comunicar entre ellos fácilmente, ayudando recuperación de los objetos.
* Sistema de Alta: Un formulario para que los usuarios puedan darse de alta en la aplicación. También habría otro formulario que solo sería para administradores que les permitiría dar de alta a moderadores y/o administradores.
* Perfiles: Los usuarios podrán ver su perfil, y editar algunos datos de éste. También se podrá dejar comentarios en el perfil de usuario.
* Sección de reportes: La aplicación contaría con una sección que mostraría reportes (tipo de objeto más perdido, lugares donde se pierden más objetos, etc.).
* Sección de contacto: En esta sección el usuario podría contactar con el soporte si se le presentara algún problema en la aplicación.

Podría haber una sección en la web con la ubicación del edificio municipal de cada zona, donde se encontraría la Caja de Objetos Perdidos. La idea sería que, la persona que encuentra un objeto en la calle, deje dicho objeto en la caja (preferiblemente en la del municipio donde encontró el objeto, en caso contrario postear en qué caja se dejó).

## Objetivo mensurable

El objetivo sería lograr que los ciudadanos comiencen a utilizar la aplicación con la finalidad de que estos sean capaces de recuperar aquellos objetos o documentos que han perdido en la calle. De alcanzar el objetivo, deberíamos empezar a notar una mayor actividad en las cajas de objetos perdidos y un menor tiempo de recuperación de dichos objetos para las personas que los perdieron. Esto también evitaría el papeleo en caso de pérdida de documentación legal (EJ: DNI).

Se considerará exitosa si alcanza al menos 20 usuarios simultáneos diarios.

## Motivo

Brindarle a la comunidad la posibilidad de poder ayudarse entre sí a recuperar objetos perdidos.

## Módulos funciones

La App contara con 3 perfiles, un módulo de registro, un módulo de alta, baja y modificación de objetos perdidos y encontrados, otro de consulta de objetos, reportes, contacto con soporte, mapa de ubicaciones de la caja de objetos y perfiles.

## Funcionalidad:

* Registrarse.
* Loguearse.
* Dar de alta Objetos Perdidos (ABM).
* Búsqueda de objetos por zona.
* Ponerse en contacto entre Usuarios.
* Cambiar estado de publicación (a “recuperado”).
* Mapa de ubicaciones con la caja de objetos.
* Reportes.

## Perfiles

* **Administrador de Aplicación:**Puedebanear, bloquear o eliminar a usuarios y/o publicaciones denunciadas o que incumplan las normas.
* **Moderadores:** Puede dar de baja publicaciones que incumplan las normas y a usuarios.
* **Usuarios:**Los usuarios pueden buscar o publicar objetos perdidos o encontrados dependiendo el caso.

**Plataforma**

Aplicación móvil, Android.

# Relevamiento

## Creación de cuenta

Al ingresar a la aplicación se mostrará una pantalla en la que se debe introducir usuario y contraseña para los usuarios registrados, para los no registrados habrá una opción de “darse de alta” para registrarse en la aplicación para lo que se le solicitará al usuario los siguientes datos:

* Nombre de usuario
* Contraseña
* Nombre
* Apellido
* Email
* Fecha de nacimiento
* Genero
* Localidad
* Provincia
* DNI (para evitar duplicados)

El sistema deberá asegurar que el nombre de usuario ingresado es válido (sin símbolos especiales, sin nombres obscenos o malintencionados), a su vez que completa el atributo “TipoUsuario” con el valor por defecto (usuario “común”).

## Tipos de publicaciones en los Foros

Se permitirán dos tipos de publicaciones en los foros: Publicar objeto perdido ó publicar objeto encontrado.

Para el caso de los objetos perdidos: se deberá crear una publicación informando:Titulo de publicación, tipo de objeto, imagen (opcional), provincia, localidad, posibles ubicaciones en dónde se perdió el objeto, fecha, rango horario(Opcional) y descripción.

Para el caso de los objetos encontrados: se creará una publicación en donde se ingrese: Titulo de publicación, tipo de objeto, imagen (opcional), provincia, localidad, ubicación en donde se encontró el objeto, fecha, rango horario y descripción.

Las publicaciones, contarán con una caja de comentarios en dónde los usuarios podrán brindar información adicional y/o pedir más detalles del objeto publicado.

## Navegación

Para navegar entre las publicaciones habrá dos secciones separadas para los objetos encontrados y los objetos perdidos. En ambas se podrá filtrar por tipo de objeto, fecha de publicación, partido y localidad. También se podrá filtrar por palabras claves las cuales darán más detalle de los objetos.

## Mensajería interna

Los usuarios podrán contactarse entre ellos a través del sistema de mensajería de la aplicación.Estesistemacontará con una pantalla en la que dos usuarios podrán enviarse mensajes de texto. Para buscar un usuario habrá una pantalla en la que se podrá buscar por nombre de usuario.

## Avisos y Notificaciones

La aplicación puede proporcionar notificaciones y así mantener a usuarios actualizados sobre las nuevas publicaciones.

## Seguridad y Privacidad

Con respecto a la seguridad y privacidad de los datos del usuario, se deberán guardar con técnicas de encriptación para que estos datos no sean de fácil acceso para ciberdelincuentes y así poder tener la información protegida.

## Roles de Usuarios de la aplicación

En adición al usuario común, el sistema contará con un usuario de tipo Moderador, y otro de tipo Administrador.Los moderadores tendrán la responsabilidad de moderar los foros y los usuarios, dándolos de baja a si se considera que se le está dando un uso incorrecto/malintencionado a la aplicación. El usuario Administrador, además de lo mencionado anteriormente, es el único que podrá dar de alta a nuevos Moderadores y Administradores.

## Mapas

En otra pestaña, se encontrará un mapa en donde el usuario podría ver las ubicaciones de los edificios municipales que cuenten con la caja de objetos, en donde un usuario podría dejar los objetos que encontrará para que un usuario pueda ir allí a recupera el objeto que previamente habría perdido.

## Reportes

Habrá una pantalla donde podrán visualizarse distintos reportes: Tipos de objetos más perdidos - localidades con mayor cantidad de objetos perdidos - cantidad de objetos perdidos, separado por tipo, en cada mes.

## Comentarios en los perfiles

En una pestaña aparte, comentarios en los perfiles de los usuarios, lo que podría ayudar a otros usuarios a evaluar la confiabilidad de las personas con las que pueden llegar a interactuar y así también poder mejorar la aplicación en base al feedback del usuario.

## Contacto

En otra pestaña se contaría con un formulario para poder contactarse con el soporte. Esto permitiría que en caso de que un usuario estuviera experimentando problemas con la aplicación, tenga una vía disponible para poder contactarse con los administradores y/o moderadores para poder trabajar en la solución para su problema.

## Módulos funcionales

Panel de "Login y Registro":

* Registro de nuevos usuarios por DNI para evitar duplicados
* Acceso a usuarios registrados

Panel de “Objetos perdidos":

* Se mostrará un listado con los objetos perdidos y la lista tendrá la acción de editar por fila y un botón de "nuevo" para dar de alta nuevos objetos perdidos.

Panel de "Alta de objeto perdido"

* Titulo de la publicación. Máximo 50 caracteres (Obligatoria).
* Tipo de objeto (Obligatoria).
* Campo para agregar una imagen (Opcional).
* Campo para indicar la fecha en la que se perdió (Obligatoria).
* Campo para indicar el rango horario de la pérdidadelobjeto (Opcional).
* Provincia (Obligatorio).
* Localidad(Obligatorio)
* Posibles ubicaciones (Obligatorio).
* Descripción. Máximo 500 caracteres (Obligatorio).

Panel de "Alta de objeto encontrado"

* Titulo de la publicación. Máximo 50 caracteres (Obligatoria).
* Tipo de objeto (Obligatoria).
* Campo para agregar una imagen (Opcional).
* Campo para indicar la fecha en la que se perdió (Obligatoria).
* Campo para indicar el rango horario de la pérdida del objeto (Opcional).
* Provincia (Obligatorio).
* Localidad(Obligatorio)
* Ubicación en donde se encontró el objeto (Obligatorio).
* Descripción. Máximo 500 caracteres (Obligatorio).

Panel de "objetos encontrados":

* Se mostrará un listado con los objetos encontrados y la lista tendrá la acción de editar por fila y un botón de "nuevo" para dar de alta nuevos objetos encontrados.

Panel para "Administradores":

* Se podrán buscar a usuarios para banearlos y/o darlos de baja totalmente del sistema
* Se podrán editar o dar de baja los hilos de los foros así como las publicaciones de los objetos

Panel para "Moderadores":

* Se podrán editar o dar de baja los hilos de los foros así como las publicaciones de los objetos

Panel de "Perfil":

* El usuario tendrá un perfil donde podrá editar sus datos personales, únicamente se tendrá como público su nombre, localidad y correo electrónico. Además de una descripción si este lo desea.
* Dentro de este panel se podrá ver los comentarios de las personas a quienes ayudo
* Desde este panel se podrá acceder a sus publicaciones activas.
* Junto al nombre del usuario se podrá ver una medalla de reconocimiento por haber ayudado (dependiendo la cantidad de ayudas dadas o resueltas).

# Contribución social

Creemos que esta aplicación sería de gran ayuda para la comunidad, puesto que (por lo menos en mi trabajo como empleado de despensa/almacén) es común que cada poco tiempo alguien pierda documentos personales como DNIs, licencias de conducir, tarjetas de discapacidad, etc. y estaría bueno que hubiera alguna forma en la que se pudiera contactar a dicha persona, o por lo menos notificar la perdida/recuperación de dichos objetos.

# Plataforma de desarrollo e implementación

* **Parte gráfica: Adobe XD, InVision Studio**o Drawio
* Software: Android Studio utilizando el lenguaje de programación Java y bases de datos MySql.

# Requerimientos funcionales

**<Requerimiento funcional #1 Registrar usuario>**

Descripción: Las personas que ingresen por primera vez a la aplicación deben crearse un usuario.

Inputs: Nombre de usuario, contraseña, nombre, apellido, email, fecha de nacimiento, genero, localidad, provincia y DNI

Proceso:

* El usuario ingresa nombre de usuario, contraseña, nombre, apellido, email, fecha de nacimiento, genero, localidad, provincia y DNI.
* Se verifica que no exista un usuario ya creado con ese DNI.
* Se registra el usuario en el sistema.

Outputs: Usuario registrado

Error Handling: En caso de que ya exista un usuario con el DNI ingresado, se informara por pantalla.

**<Requerimiento funcional #2 Ingresar a la aplicación>**

Descripción: Para poder utilizar la aplicación se deberá ingresar con un usuario y contraseña.

Inputs: Nombre de usuario y contraseña.

Proceso:

* El usuario ingresa nombre de usuario y contraseña.
* Se verifica que ese nombre de usuario y contraseña coincidan con uno existente en el sistema.
* Se ingresa a la aplicación.

Outputs: Ingreso exitoso

Error Handling: En caso de que no esté registrado ese usuario y contraseña, se informara por pantalla.

**<Requerimiento funcional #3 Alta publicación objeto perdido>**

Descripción: El usuario podrá crear una publicación en el foro el objeto que haya perdido.

Inputs: Titulo de publicación, tipo de objeto, imagen (opcional), fecha, rango horario (opcional), provincia, localidad, posibles ubicaciones, descripción.

Proceso:

* El usuario ingresa los datos de entrada.
* Se verifican que los datos requeridos sean ingresados.
* Se guardan los datos y se crea la publicación.

Outputs: Publicación creada con éxito.

Error Handling: En caso de que el usuario no rellene un campo requerido, se le notificará al usuario.

**<Requerimiento funcional #4Alta publicación objeto encontrado>**

Descripción: El usuario podrá crear una publicación en el foro del objeto que haya encontrado.

Inputs: Titulo de publicación, tipo de objeto, imagen (opcional), fecha, rango horario (Opcional), provincia, localidad, ubicación, descripción.

Proceso:

* El usuario ingresa los datos de entrada.
* Se verifican que los datos requeridos sean ingresados.
* Se guardan los datos y se crea la publicación.

Outputs: Publicación creada con éxito.

Error Handling: En caso de que el usuario no rellene un campo requerido, se le notificará al usuario.

**<Requerimiento funcional #5 Navegar entre publicaciones>**

Descripción: El usuario podrá navegar entre las distintas publicaciones dividida entre dos secciones: Objetos perdidos y objetos encontrados. El usuario podrá aplicar filtros para obtener resultados de búsqueda más precisos.

Inputs: Interacción del usuario con la vista de publicaciones.

Proceso:

* El usuario está en la vista de publicaciones.
* El usuario selecciona una sección de su preferencia y podrá aplicar los filtros si lo desea, tales como tipo de objeto, fecha de publicación, provincia y localidad. También podrá filtrar por palabras clave.

Outputs: Lista de publicaciones con los criterios seleccionados por el usuario.

Error Handling: En caso de que no haya publicaciones con los criterios seleccionados por el usuario, se le notificará.

**<Requerimiento funcional #6 Chatear entre usuarios>**

Descripción: Los usuarios podrán intercambiar mensajes de texto entre sí mediante el sistema de mensajería interna de la aplicación. Un usuario podrá buscar a otro mediante su nombre de usuario para poder comenzar a chatear.

Inputs: Interacción del usuario con la vista de mensajería interna de la aplicación.

Proceso:

* El usuario ingresa en la mensajería interna de la aplicación.
* El usuario selecciona con quien desea chatear. También tendrá un cuadro de texto para buscar a otro usuario por su nombre de usuario para poder comenzar a chatear.
* Se mostrará una pantalla en la que ambos usuarios podrán chatear en tiempo real.

Outputs: Conversación en tiempo real en la pantalla entre ambos usuarios.

Error Handling: En caso de que en la búsqueda por nombre de usuario, el usuario buscado no exista, se le notificará al usuario buscador.

**<Requerimiento funcional #7 Editar perfil>**

Descripción: El usuario podrá entrar a su propio perfil, donde al darle a un lápiz (solo disponible para el dueño del perfil) puede cambiar sus datos personales.

Inputs: Nombre de usuario, nombre, apellido, email, fecha de nacimiento, genero, localidad, provincia, sube.

Proceso:

* El usuario edita el campo que desee cambiar, puede ser nombre de usuario, nombre, apellido, email, fecha de nacimiento, genero, localidad o provincia.
* Se verifica que todos los datos estén cargados.
* Se guardan los datos del usuario en el sistema.

Outputs: Cambios guardados correctamente.

Error Handling:

* En caso de que el usuario no complete un campo solicitado, se le notificará de alguna forma.

**<Requerimiento funcional #8 Comentar perfil>**

Descripción: Un usuario que pudo recuperar su objeto perdido tiene la posibilidad de agradecer a quien lo recupero o le indico sobre este, accediendo al perfil de quien público el objeto encontrado o encontró el objeto de la publicación de perdidos.

Inputs: Comentario.

Proceso:

* El usuario entra al perfil de quien lo ayudo, selecciona la opción de comentar (habilitada solo si recupero su objeto)
* Expresa su comentario y lo pública.
* Se guarda el comentario en el perfil al que se accedió.

Outputs: Comentario publicado.

Error Handling:

* En caso de que el usuario no complete el campo.
* Se escriban insultos.

**<Requerimiento funcional #9 Dar de baja usuario>**

Descripción: El administrador dará de baja o baneará a un usuario.

Inputs: DNI.

Proceso:

* El administrador ingresa el dni del usuario a dar de baja.
* Se verifica que el DNI sea existente.
* Se realiza la baja lógica del usuario.

Outputs: Usuario dado de baja exitosamente.

Error Handling:

* En caso de que el DNI sea inexistente se le avisará al administrador que dicho dni no existe.

**<Requerimiento funcional #10 Editar publicaciones de objetos perdidos>**

Descripción: El moderador o administrador modificará las publicaciones de objetos perdidos

Inputs: Titulo de publicación, tipo de objeto, imagen (opcional), fecha, rango horario (opcional), provincia, localidad, posibles ubicaciones, descripción.

Proceso:

* El moderador/administrador modifica los datos que considera necesarios.
* Se guarda la modificación hecha por el moderador/administrador.

Outputs: Publicación modificada con éxito.

Error Handling:

* En caso de que no se pueda modificar la publicación por un error se le informará a quien esté realizando la acción.

**<Requerimiento funcional #11 Editar publicaciones de objetos encontrados>**

Descripción: El moderador o administrador modificará las publicaciones de objetos encontrados

Inputs: Titulo de publicación, tipo de objeto, imagen (opcional), fecha, rango horario (opcional), provincia, localidad, posibles ubicaciones, descripción.

Proceso:

* El moderador/administrador modifica los datos que considera necesarios.
* Se guarda la modificación hecha por el moderador/administrador.

Outputs: Publicación modificada con éxito.

Error Handling:

* En caso de que no se pueda modificar la publicación por un error se le informará a quien esté realizando la acción.

**<Requerimiento funcional #12 Dar de baja publicación>**

Descripción: El moderador/administrador dará de baja una publicación.

Inputs: Interacción del usuario con la vista de edición de publicación.

Proceso:

* El moderador/administrador ingresa al modo de edición de publicación.
* El moderador/administrador selecciona la opción de dar de baja
* Se realiza la baja lógica de la publicación en cuestión.

Outputs: Publicación dada de baja exitosamente.

Error Handling:

* En caso de que haya alguna falla en el sistema al momento de realizar la interacción, se le hará saber al usuario mediante un mensaje de error.

**<Requerimiento funcional #13 Registrar Moderador o Administrador>**

Descripción: Un usuario administrador da de alta a otro Administrador, o a un usuario de tipo Moderador.

Inputs: Nombre de usuario, contraseña, nombre, apellido, email, fecha de nacimiento, genero, localidad, provincia, DNI y Tipo de usuario.

Proceso:

* El administrador ingresa los datos del usuario a dar de alta: nombre de usuario, contraseña, nombre, apellido, email, fecha de nacimiento, genero, localidad, provincia, DNI y Tipo de usuario.
* Se verifica que no exista un usuario ya creado con ese DNI.
* Se registra el usuario en el sistema.

Outputs: Usuario registrado

Error Handling: En caso de que ya exista un usuario registrado con ese DNI, el sistema procederá a modificar únicamente el atributo de Tipo de usuario.

**<Requerimiento funcional #14Ver mapa>**

Descripción: El usuario podrá ver las ubicaciones de los edificios municipales que cuenten con la caja de objetos, en donde un usuario podría dejar los objetos que encontrará para que un usuario pueda ir allí a recupera el objeto que previamente habría perdido.

Inputs: Interacción del usuario con la vista de ubicaciones de cajas de objetos.

Proceso:

* El Usuario interactúa con la vista ubicaciones de cajas de objetos.
* Se muestran las diferentes ubicaciones de los municipios que cuenten con la caja de objetos.
* Al seleccionar alguna de estas ubicaciones, se le mostrarán más detalles al usuario.

Outputs: Información detallada sobre la ubicación seleccionada.

Error Handling: - En caso de que haya alguna falla en el sistema al momento de realizar la interacción, se le hará saber al usuario mediante un mensaje de error.

**<Requerimiento funcional #15Contactar soporte>**

Descripción: En caso de que un usuario experimente problemas con la aplicación, podrá contactarse con el soporte enviando un mensaje a través de este formulario

Inputs: Nombre, apellido, email, breve resumen del problema y detalle del problema.

Proceso:

* El Usuario rellena el formulario con su nombre, apellido, email, un breve resumen del problema y el detalle del problema.
* Se envía un correo al email de soporte con la información que proporciono el usuario.

Outputs: Correo enviado.

Error Handling: En caso de que haya alguna falla al momento de enviar el correo, se le hará saber al usuario mediante un mensaje de error.

**<Requerimiento funcional #16 Comentar publicación>**

Descripción: Los usuarios podrán realizar comentarios dentro de una publicación para brindar información adicional y/o pedir más detalles del objeto publicado.

Inputs: Comentario.

Proceso:

* El usuario entra a una publicación y selecciona la opción de comentar.
* Escribe su comentario.
* Selecciona la opción de publicar.
* Se guarda el comentario en la publicación.

Outputs: Comentario publicado.

Error Handling: En caso de que haya alguna falla al momento de enviar el comentario, se le hará saber al usuario mediante un mensaje de error.

**<Requerimiento funcional #17 Actualizar publicación a “Recuperado”>**

Descripción: El creador de una publicación podrá actualizar su estado a “Recuperado” indicando que este objeto ya fue entregado a su dueño.

Inputs: Interacción del usuario con la vista de la publicación.

Proceso:

* El usuario selecciona la opción de “Objeto recuperado”.
* El estado de la publicación es modificado a “recuperado”.

Outputs: Publicación actualizada.

Error Handling: En caso de que haya alguna falla al momento de cambiar el estado de la publicación, se le hará saber al usuario mediante un mensaje de error.

**<Requerimiento funcional #18 Reportes>**

Descripción: En esta sección se podrán ver reportes sobre los tipos de objetos más perdidos, localidades con mayor cantidad de objetos perdidos, cantidad de objetos perdidos, separado por tipo, en cada mes.

Inputs: Filtro.

Proceso:

* El usuario selecciona la opción por la cuál quiere obtener un reporte
* Se mostrará el reporte generado por pantalla.

Outputs: Reporte generado con éxito.

Error Handling: En caso de que haya algún error intentando armar el reporte se le informará al usuario con un cartel.

**<Requerimiento funcional #19 Notificaciones >**

Descripción: Esta sección se encargará de enviarle notificaciones al usuario para mantenerlo actualizado sobre nuevas publicaciones.

Inputs: Creación de una publicación.

Proceso:

* Se crea una nueva publicación.
* Se verifica la provincia de la publicación creada y se envía notificaciones a los usuarios que residan en la misma provincia.

Outputs: Notificación al usuario sobre una nueva publicación ha sido creada en su provincia.

Error Handling: En caso de que haya algún error intentando enviar la notificación se le informará al usuario con un cartel.

**<Requerimientos No-Funcionales>**

1 Desempeño: Las técnicas de seguridad aplicadas no deberían de afectar el rendimiento de la aplicación.

2 Confiabilidad: Las técnicas utilizadas deben ser de confianza.

3 Disponibilidad: Al implementar las técnicas de seguridad, ésta no debería de ser algún problema para usuarios comunes.

4 Seguridad:Los datos del usuario, se deberán guardar con técnicas de encriptación.

5 Mantenibilidad: Se deben emplear actualizaciones de seguridad regularmente.

6 Portabilidad: Las actualizaciones de seguridad no deberían de afectar a la portabilidad de la aplicación.

…….

Restricciones

* Los datos personales de los usuarios deben ser encriptados con técnicas de seguridad.

# Diagrama de análisis de casos de uso



# Especificación de casos de uso

|  |
| --- |
| **1. Caso de uso: Registrar usuario** |
| Descripción: Las personas que ingresen por primera vez a la aplicación deben crearse un usuario. |
| Actores: Usuario común |
| **Camino Básico:**  1. El usuario ingresa nombre de usuario, contraseña, nombre, apellido, email, fecha de nacimiento, genero, localidad, provincia y DNI.  2. El sistema valida que los datos ingresados coincidan con los formatos requeridos, por ejemplo que la contraseña contenga números y letras. También verifica que no haya campos vacíos. Si alguno de los campos no cumple los requerimientos lo informara por pantalla.  3. El sistema invoca “VerificarUsuario”. Si el DNI existe, muestra mensaje “Ya existe un usuario con ese DNI”. Si el DNI no existe se registran los datos ymuestra por pantalla el mensaje “Usuario registrado”  4. El actor cierra el caso de uso |
| Precondiciones: --- |
| Postcondiciones: Al usuario le llegara un correo con un número y deberá ingresarlo en un campo para verificar que es su correo electrónico ya que este medio es por el que será notificado de los objetos. |
| Requerimientos especiales/observaciones: La contraseña debe tener mínimo 8 caracteres, letras y números. Nombre y apellido no deben contener números. Todos los campos son obligatorios. La edad no puede ser mayor a 99 años. |

|  |
| --- |
| **2. Caso de uso: Ingresar a la aplicación** |
| Descripción: Las personas ya registradas ingresan a la aplicación |
| Actores: Usuario común, Usuario moderador, Usuario administrador |
| **Camino Básico:**  1. El usuario ingresa su nombre y contraseña  2. El sistema invoca “VerificarUsuario”. Si el usuario y contraseña no existen, muestra por pantalla el mensaje "Usuario y contraseña incorrectos". Si son correctos el sistema deja pasar al usuario al home.  3. El actor cierra el caso de uso. |
| Precondiciones: --- |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: Ante una determinada cantidad de fallos de login el usuario es bloqueado y debe desbloquear la cuenta utilizando el código que le llegará al correo. |

|  |
| --- |
| **3. Caso de uso: Alta publicación objeto perdido** |
| Descripción: Las personas dan de alta una publicación de objeto perdido. |
| Actores: Usuario común |
| **Camino Básico:**  1. El usuario ingresa a la vista de foros y selecciona objetos perdidos.  2. Se le mostrará al usuario el listado de objetos perdidos y éste selecciona la opción de publicar objeto perdido.  3. La aplicación le muestra los campos a completar.  4. El usuario ingresa el Titulo de publicación, tipo de objeto, imagen (opcional), fecha, rango horario (opcional), provincia, localidad, posibles ubicaciones y descripción.  5. El usuario al presionar en aceptar, se verificará que los campos requeridos hayan sido completados y procederá a dar de alta la publicación y notificar al usuario. |
| Precondiciones: El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: Según la provincia que ingrese el usuario, se cargarán sus respectivas localidades para mayor comodidad. Se le permitirá ingresar al usuario hasta un máximo de 2 posibles ubicaciones. Se debe verificar que los campos requeridos hayan sido ingresados por el usuario. |

|  |
| --- |
| **4. Caso de uso: Alta publicación objeto encontrado** |
| Descripción: Las personas dan de alta una publicación de objeto encontrado. |
| Actores: Usuario común |
| **Camino Básico:**  1. El usuario ingresa a la vista de foros y selecciona objetos encontrados.  2. Se le mostrará al usuario el listado de objetos encontrados y éste selecciona la opción de publicar objeto encontrado.  3. La aplicación le muestra los campos a completar.  4. El usuario ingresa el Titulo de publicación, tipo de objeto, imagen (opcional), fecha, rango horario (Opcional), provincia, localidad, ubicación y descripción.  5. El usuario al presionar en aceptar, se verificará que los campos requeridos hayan sido completados y procederá a dar de alta la publicación y notificar al usuario. |
| Precondiciones: El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: Según la provincia que ingrese el usuario, se cargarán sus respectivas localidades para mayor comodidad. Se debe verificar que los campos requeridos hayan sido ingresados por el usuario. |

|  |
| --- |
| **5. Caso de uso: Navegar entre publicaciones** |
| Descripción: Los usuarios podrán navegar por las publicaciones y al seleccionar una se mostraran los datos de la misma. |
| Actores: Usuario común, Usuario moderador, Usuario administrador |
| **Camino Básico:**  1. El actor selecciona una publicación.  2. El sistema invoca “Verificar publicación” y retorna la publicación encontrada.  3. El actor cierra el caso de uso |
| Precondiciones:   * El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: --- |

|  |
| --- |
| **6. Caso de uso: Chatear entre usuarios** |
| Descripción: Los usuarios podrán comunicarse entre ellos mediante mensajes de texto. |
| Actores: Usuario común |
| **Camino Básico:**  1. El usuario selecciona con quien desea chatear o bien buscando al usuario con quien desea chatear ingresando el nombre del usuario en el cuadro de búsqueda y luego presionando el botón de buscar.  2. El sistema mostrará una pantalla en la que ambos usuarios podrán chatear en tiempo real.En caso de que no se pueda acceder al usuario se mostrará el mensaje: "No se pudo encontrar el usuario".  3. El actor cierra el caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: En caso de que en la búsqueda por nombre de usuario, el usuario buscado no exista, se le notificará al usuario buscador. |

|  |
| --- |
| **7. Caso de uso: Editar perfil** |
| Descripción: Cada usuario podrá modificar los datos personales. |
| Actores: Usuario común, Usuario moderador, Usuario administrador |
| **Camino Básico:**  1. Se mostrará por pantalla los datos del usuario para que pueda modificarlos.  2. El actor modifica los datos a excepción del campo de DNI que no puede ser modificado.  3. El sistema valida que los datos ingresados coincidan con los formatos requeridos y que no haya campos vacíos. Si alguno de los campos no cumple los requerimientos lo informara por pantalla.  4. Se registran los nuevos datos yse muestra por pantalla el mensaje “Cambios guardados”.  5. El actor cierra el caso de uso. |
| Precondiciones: El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: La contraseña debe tener mínimo 8 caracteres, letras y números. Nombre y apellido no deben contener números. Todos los campos son obligatorios. |

|  |
| --- |
| **8. Caso de uso: Comentar perfil** |
| Descripción: Cuando un usuario es ayudado, tiene la posibilidad de dejar un comentario en el perfil de quien lo ayudo. |
| Actores: Usuario común. |
| **Camino Básico:**  1. Se ingresa al perfil de quien brindo ayuda.  2. Le da al botón de comentar.  3. Una vez realizado el comentario, se le da al botón “Publicar”  4. Se valida que el comentario no este vacío ni contenga palabras de odio.  5. Se guarda el comentario en base de datos.  6. Aparece el mensaje “Comentario publicado”.  7. Se actualiza automáticamente la página mostrando el comentario publicado. |
| Precondiciones:   * El usuario debe estar logueado. * El usuario debe tener habilitado la opción de comentar en el perfil de quien lo ayudo. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: El comentario debe tener más de 10 caracteres. |

|  |
| --- |
| **9. Caso de uso: Dar de baja Usuario** |
| Descripción: Los administradores/moderadores dan de baja a un usuario. |
| Actores: Usuario Administrador, Usuario Moderador. |
| **Camino Básico:**  1. Se ingresa al panel de administracion.  2. El actor busca en sistema el usuario a partir del DNI o del nombre.  3. El sistema invoca “Verificar Usuario” y retorna el usuario encontrado.  4. Se le da al botón “Dar de baja”  5. Se confirma la acción.  6. Aparece el mensaje “Usuario Eliminado”.  7. El actor cierra el caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. * El usuario debe ser de tipo Administrador/Moderador |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: De no encontrarse el usuario, el sistema se lo hará saber al actor en pantalla. De ocurrir algún problema en la base de datos al momento de buscar o eliminar al usuario, se le informara al usuario por pantalla |

|  |
| --- |
| **10. Caso de uso: Editar publicaciones de objetos perdidos** |
| Descripción: Los administradores/moderadores modifican la información o datos de una publicación. |
| Actores: Usuario Administrador, Usuario Moderador. |
| **Camino Básico:**  1. El actor busca en sistema el foro a partir de su ID.  2. El sistema invoca “Verificar publicación” y retorna la publicación encontrada.  3. Se le da al botón “Editar”.  4. Se editan los datos que el actor considere necesarios.  5. Se confirma la acción.  6. Aparece el mensaje “Publicación modificada”.  7. Se actualiza la página, mostrando la modificación aplicada.  8. El actor cierra el caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. * El usuario debe ser de tipo Administrador/Moderador |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: De no encontrarse la publicación, el sistema se lo hará saber al actor en pantalla. De ocurrir algún problema en la base de datos al momento de buscar o editar la publicación, se le informara al usuario por pantalla |

|  |
| --- |
| **11. Caso de uso: Editar publicaciones de objetos Encontrados** |
| Descripción: Los administradores/moderadores modifican la información o datos de una publicación. |
| Actores: Usuario Administrador, Usuario Moderador. |
| **Camino Básico:**  1. El actor busca en sistema el foro a partir de su ID.  2. El sistema invoca “Verificar publicación” y retorna la publicación encontrada.  3. Se le da al botón “Editar”.  4. Se editan los datos que el actor considere necesarios.  5. Se confirma la acción.  6. Aparece el mensaje “Publicación modificada”.  7. Se actualiza la página, mostrando la modificación aplicada.  8. El actor cierra el caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. * El usuario debe ser de tipo Administrador/Moderador |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: De no encontrarse la publicacion, el sistema se lo hará saber al actor en pantalla. De ocurrir algún problema en la base de datos al momento de buscar o editar la publicacion, se le informara al usuario por pantalla |
| **12. Caso de uso: Dar de baja Publicación** |
| Descripción: Los administradores/moderadores dan de baja una publicación. |
| Actores: Usuario Administrador, Usuario Moderador. |
| **Camino Básico:**  1. El actor busca en sistema el foro a partir de su ID.  2. El sistema invoca “Verificar publicación” y retorna la publicación encontrada.  3. Se le da al botón “Dar de baja”.  4. Se confirma la acción.  5. Aparece el mensaje “Publicación Eliminada”.  6. El actor cierra el caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. * El usuario debe ser de tipo Administrador/Moderador |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: De no encontrarse la publicación, el sistema se lo hará saber al actor en pantalla. De ocurrir algún problema en la base de datos al momento de buscar o eliminar la publicación, se le informara al usuario por pantalla |

|  |
| --- |
| **13. Caso de uso: Registrar Moderador o administrador** |
| Descripción: Un usuario administrador tiene la capacidad de dar de alta a otro usuario Administrador o a otro usuario Moderador |
| Actores: Usuario Administrador. |
| **Camino Básico:**  1. El usuario administrador va a su panel de administración.  2. Busca la opción “Dar de alta Usuario”  3. Completa los siguientes datos del formulario; nombre de usuario, contraseña, nombre y apellido, email, fecha de nacimiento, genero, localidad, provincia, DNI y tipo de usuario.  4. En tipo de usuario debe decidir si se trata de un usuario Administrador o Moderador.  5. Una vez que le da a agregar usuario se validan los datos (como un DNI duplicado).  6. Se da el aviso de que se agregó correctamente al usuario y se limpian los datos del formulario.  7. El actor cierra el formulario / caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. * El usuario debe ser Administrador. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones:La contraseña debe tener mínimo 8 caracteres, letras y números. Nombre y apellido no deben contener números. Todos los campos son obligatorios. |

|  |
| --- |
| **14. Caso de uso: Ver mapa** |
| Descripción: Los usuarios pueden acceder a la pestaña “mapa” donde podrán ver las ubicaciones de los edificios municipales que cuentan con caja de objetos perdidos. Allí pueden dejar o buscar un objeto. |
| Actores: Usuario Común, Usuario Administrador, Usuario Moderador |
| **Camino Básico:**  1. El usuario va a “Ver mapa”.  2. El usuario interactúa con el mapa, pudiendo observar las ubicaciones de los municipios con cajas de objetos.  3. Al seleccionar una ubicación se podrá ver un detalle del lugar.  4. El actor cierra el mapa / caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones:Cualquier usuario puede acceder al mapa, ya que solo se trata de una herramienta para conocer la ubicación de municipios. |

|  |
| --- |
| **15. Caso de uso: Contactar soporte** |
| Descripción: En caso de que ocurra un problema con la aplicación o situación desagradable en la plataforma es posible contactar al soporte. |
| Actores: Usuario Común |
| **Camino Básico:**  1. El usuario va a “Contacto”.  2. En esta parte de la app podrá ver un formulario en el que indicará su nombre, apellido, email y la descripción de su problema.  3. Al completar los datos debe darle al botón “Enviar”.  4. Se envía un correo al email de soporte con la información que proporciono el usuario.  5. Se redirecciona al inicio/default al usuario. |
| Precondiciones:   * El usuario debe estar logueado. |
| Postcondiciones:   * Se envía una respuesta de soporte al usuario que solicito ayuda. |
| Requerimientos especiales/observaciones:La descripción del problema en el formulario debe ser detallada correctamente |

|  |
| --- |
| **16. Caso de uso: Comentar publicación** |
| Descripción: Los usuarios al seleccionar una publicación, luego de los datos de la misma habrá una caja de comentarios en la cual los usuarios podrán dejar sus comentarios. |
| Actores: Usuario Común, Usuario moderador, Usuario administrador |
| **Camino Básico:**  1. El actor selecciona una publicación.  2. En la caja de comentarios, el usuario ingresa en una caja de texto lo que quiera comentar.  3. Al presionar el botón “publicar comentario” se guardará lo que escribió el usuario en la caja de texto en la publicación. |
| Precondiciones:   * El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones:Al presionar el botón “publicar comentario” se debe verificar que haya algo escrito en la caja de texto, caso contrario se le notificará al usuario que debe completar ese campo. Cualquier tipo de usuario puede comentar, ya sea un usuario común para brindar o pedir más información, o un moderador y/o administrador para lo mismo o pedir que se edite algunos campos si es que fuese necesario. La caja de texto debe permitir máximo 200 caracteres. |

|  |
| --- |
| **17. Caso de uso: Actualizar publicación a “Recuperado”** |
| Descripción: El usuario creador de una publicación podrá cambiar el estado a “Recuperado”. |
| Actores: Usuario común |
| **Camino Básico:**  1. El actor selecciona la opción “Objeto recuperado”.  2. El sistema cambia el estado de la publicación a “Recuperado”.  3. El actor cierra el caso de uso. |
| Precondiciones:   * El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: |

|  |
| --- |
| **18. Caso de uso: Reportes** |
| Descripción: En esta sección se podrá ver reportes sobre los objetos perdidos y sus características. |
| Actores: Usuario común, Usuario moderador, Usuario administrador |
| **Camino Básico:**  1. El usuario selecciona la opción por la cual quiere obtener un reporte.  2. El sistema generará un reporte el cuál se lo mostrará a dicho usuario por pantalla3. |
| Precondiciones:   * El usuario debe estar logueado. |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: -- |

|  |
| --- |
| **19. Caso de uso: Notificaciones** |
| Descripción:Esta sección enviara notificaciones sobre nuevas publicaciones a los usuarios |
| Actores: Usuario común, Usuario moderador, Usuario administrador |
| **Camino Básico:**  1. Una nueva publicación es generada por un usuario.  2. El sistema verifica la zona de la publicación creada contra la zona de los usuarios del sistema  3.El sistema envía una notificación a los usuarios sobre la nueva publicación que se encuentra a sus alrededores. |
| Precondiciones: -- |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: -- |

|  |
| --- |
| **20. Verificar Usuario** |
| Descripción: Esta sección recibirá los datos de usuario tanto para el registro, ingreso, baja o modificación del mismo (incluye moderación) |
| Actores: CU 1 – Registrar Cliente, CU 2 – Ingresar a la aplicación, CU 9 - Dar de baja Usuario, CU 13 – Registrar Moderador o Administrador |
| **Camino Básico:**  1. Se reciben las credenciales de un nuevo usuario y se verifica la inexistencia del DNI dentro de la base de datos para permitir el alta  2. Se reciben las credenciales de un usuario registrado y se machean con las existentes de la base de datos, si existe tal combinación se da el ok para el ingreso  3. Se recibe un usuario para darse de baja, se machea los datos recibidos con los de la base y se da el ok.  4. Se recibe un usuario para moderar, se machea los datos recibidos con los de la base y se da el ok |
| Precondiciones: -- |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: Se recibe el tipo de operación que se desea realizar, por ejemplo “A” para alta con los datos que no se pueden repetir en la base como nombre de usuario, DNI y correo electrónico. |

|  |
| --- |
| **21. Verificar Publicación** |
| Descripción: Esta sección recibirá los datos de una publicación para verificar |
| Actores: CU 10 – Editar publicaciones de objetos perdidos, CU 11 - Editar publicaciones de objetos encontrados, CU 12 – Dar de baja publicación , CU 16 – Comentar publicación, CU 17 – Actualizar publicación a “Recuperado” |
| **Camino Básico:**  1. Se recibe el ID de una publicación (sea de objeto perdido o encontrado)  2. Se verifica que la nueva publicación cumpla con las condiciones que deba tener dicha publicación, de ser así se da el Ok.  3. Se verifica que la publicación modificada cumpla con las condiciones que deba tener dicha publicación, de ser así se da el Ok. |
| Precondiciones: -- |
| Postcondiciones: --- |
| Requerimientos especiales/observaciones: |

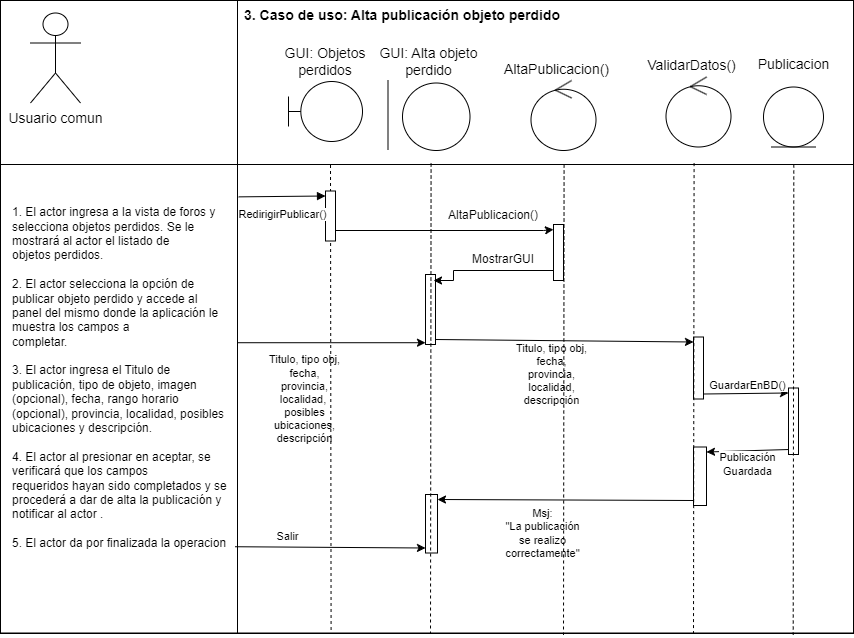
# Diagramas de Secuencia

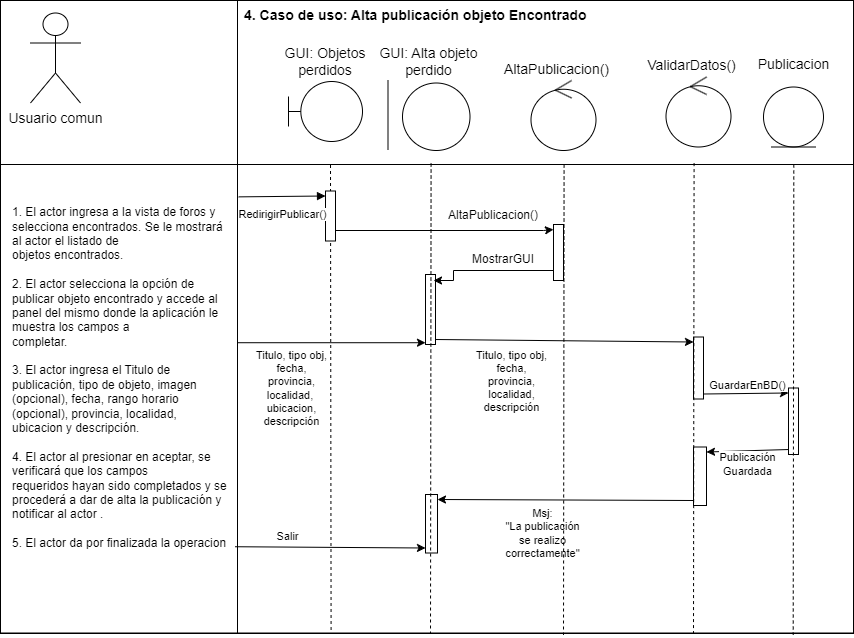
Diagrama

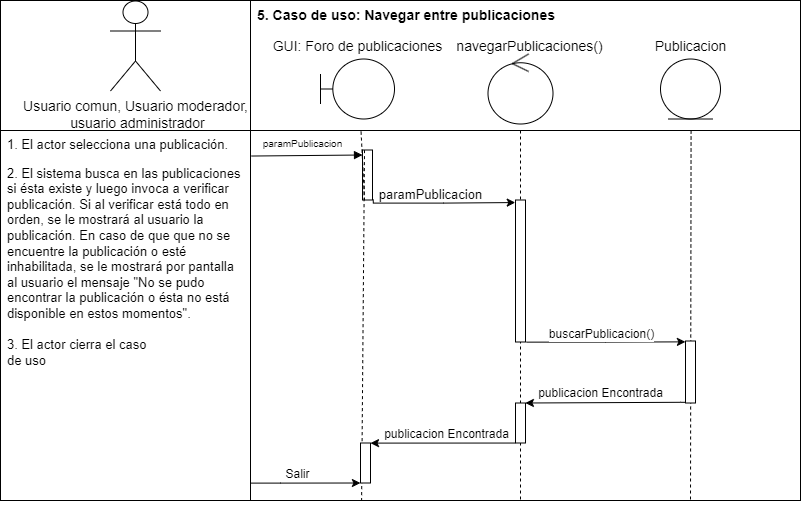
Descripción generada automáticamente

Diagrama

Descripción generada automáticamente







Diagrama

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

Diagrama

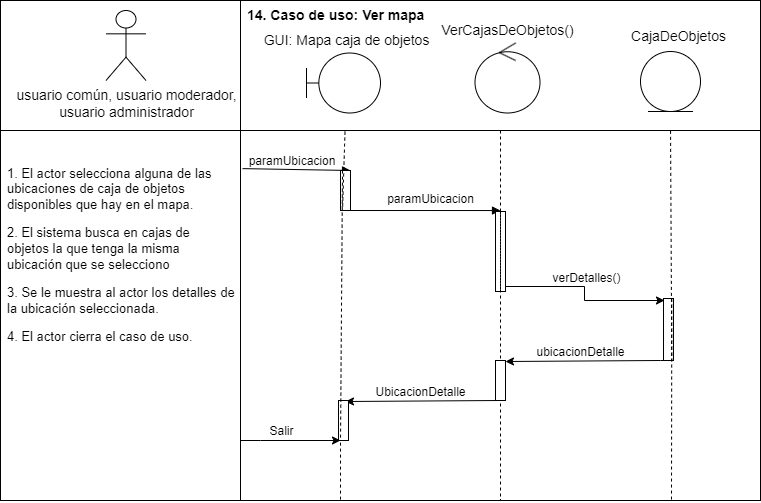
Descripción generada automáticamente

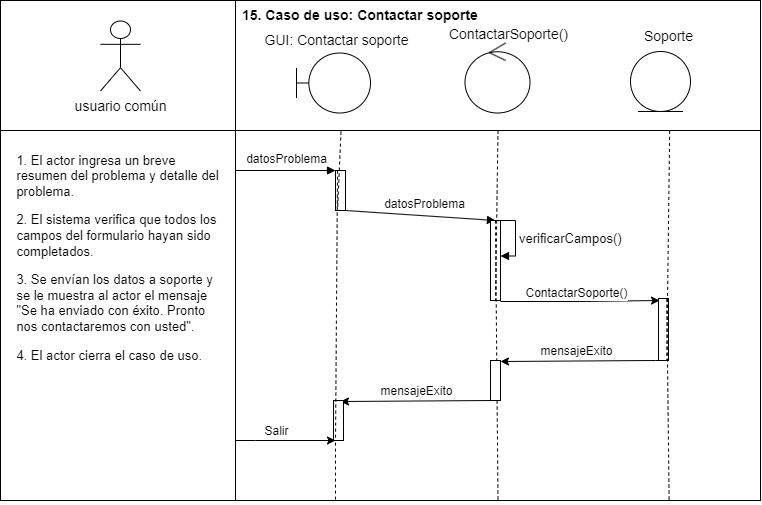
Diagrama

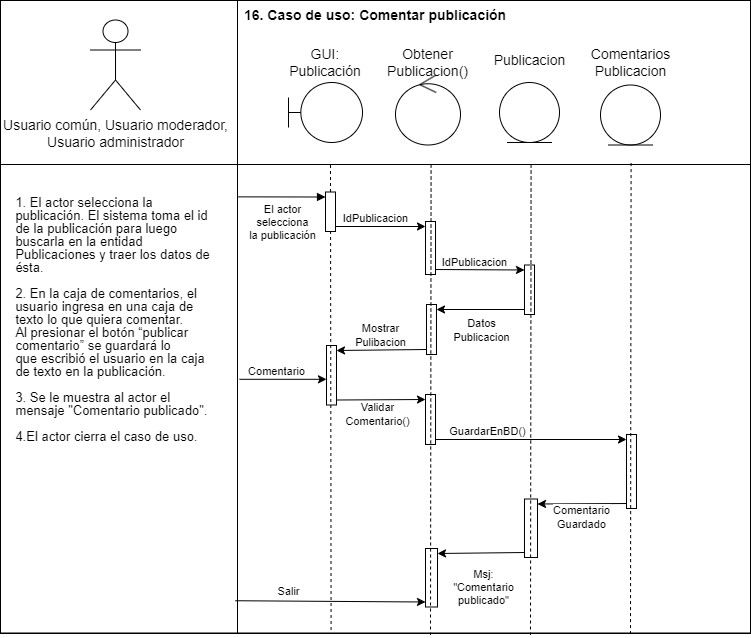
Descripción generada automáticamente

Diagrama

Descripción generada automáticamente







Diagrama

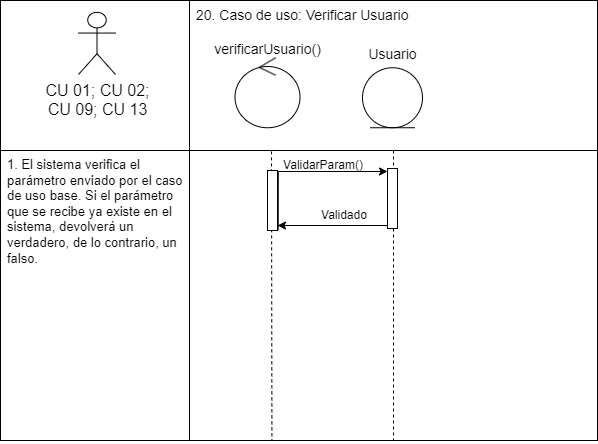
Descripción generada automáticamente

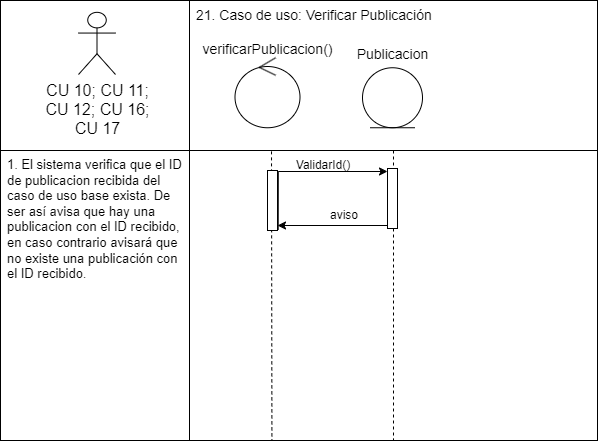
Diagrama

Descripción generada automáticamente

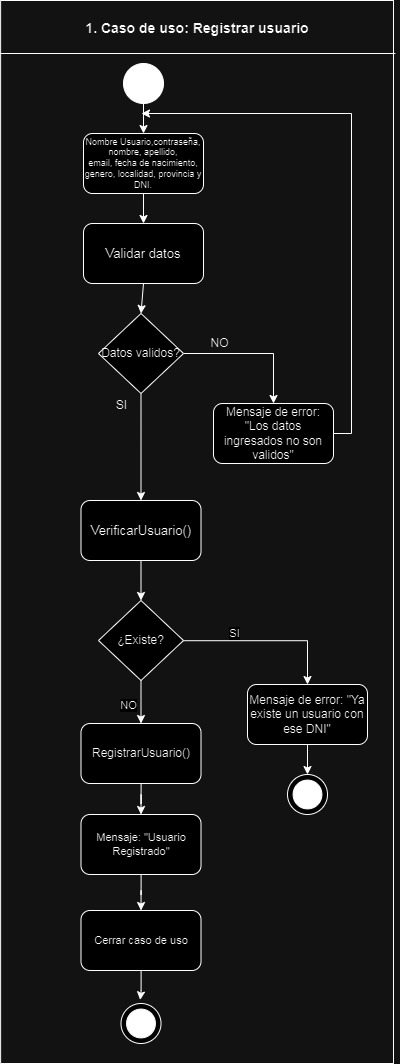
Diagrama

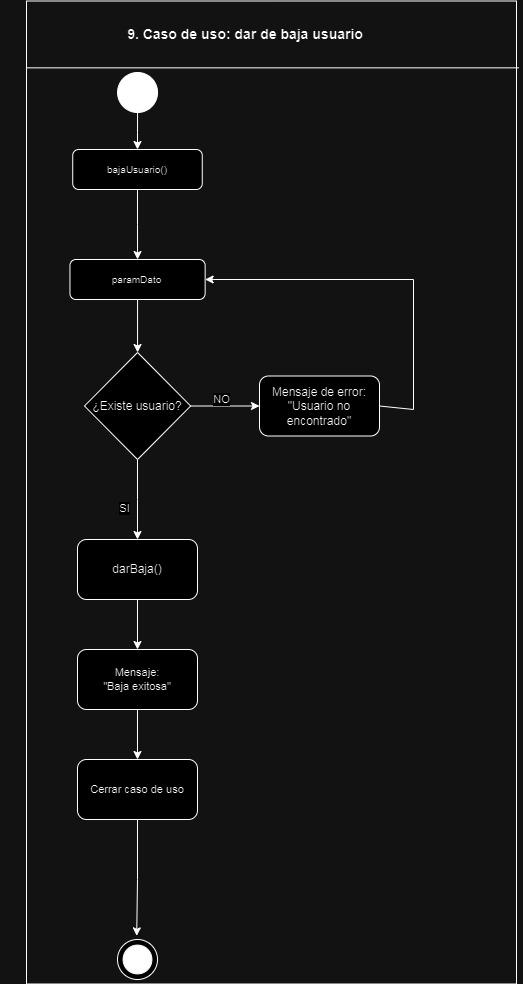
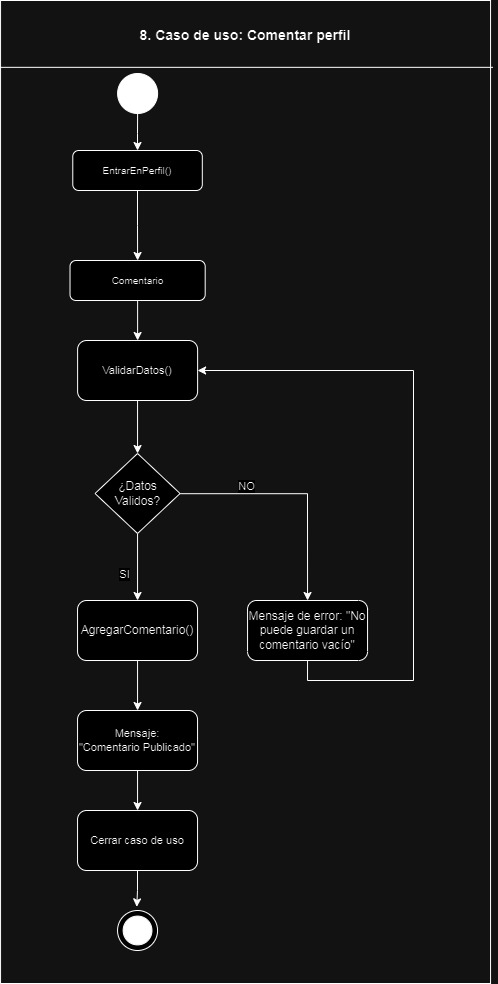
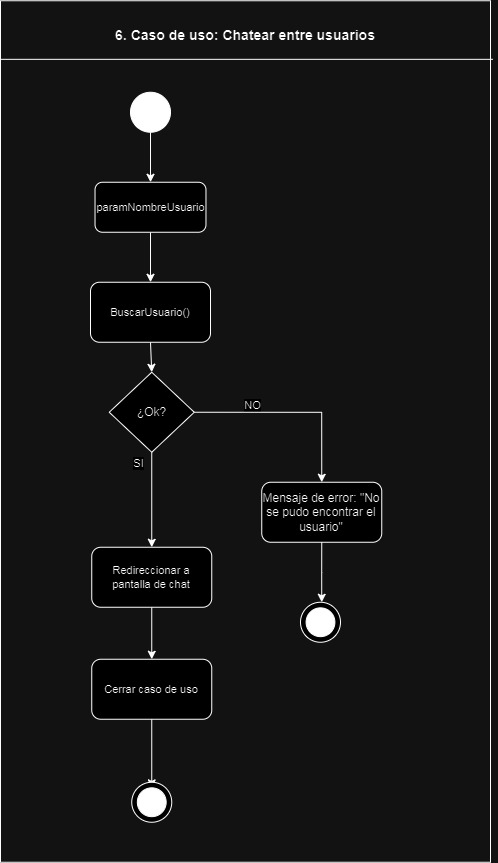
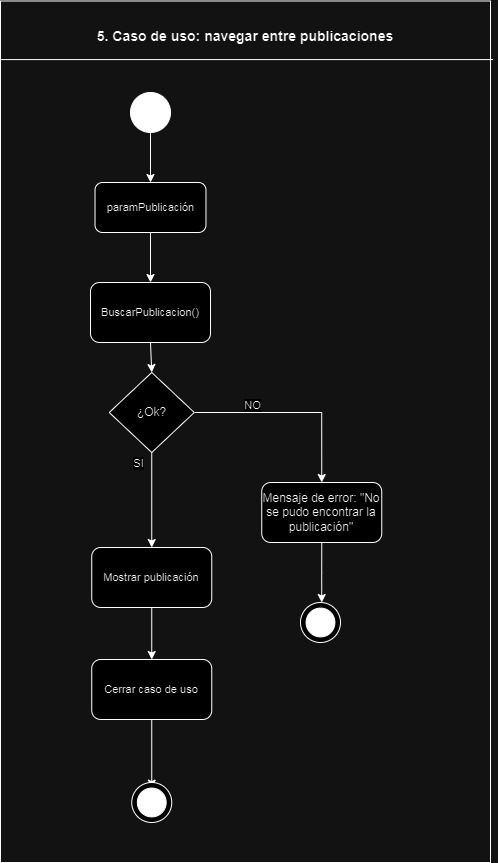
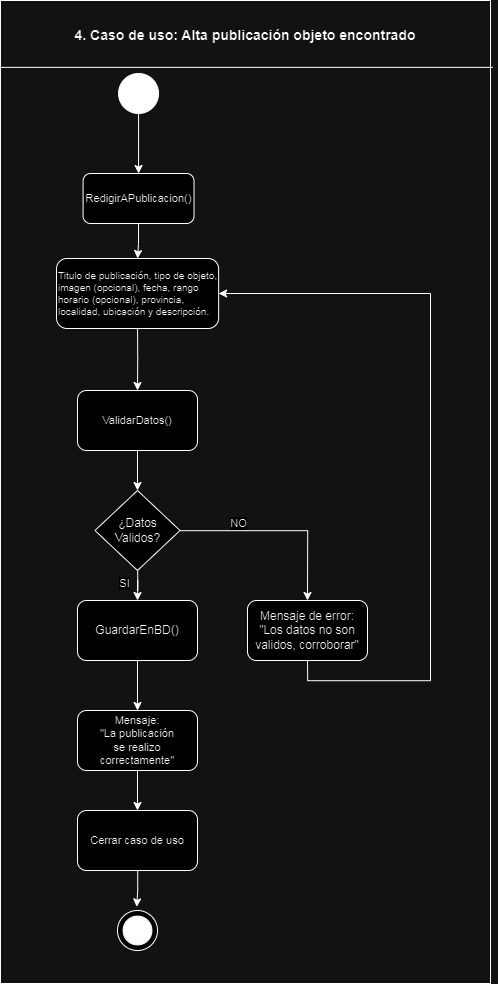
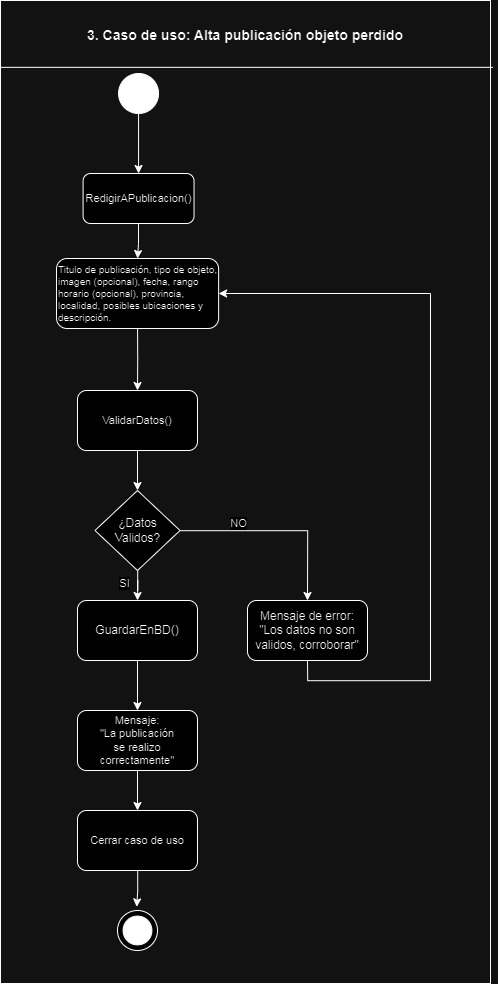
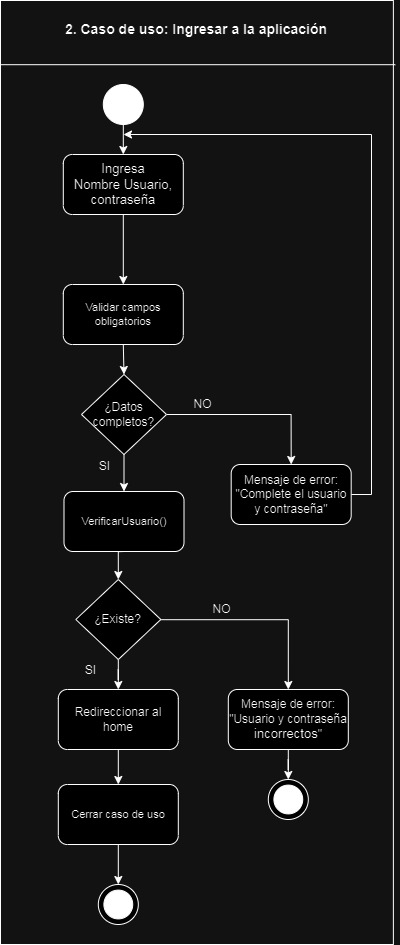
Descripción generada automáticamente

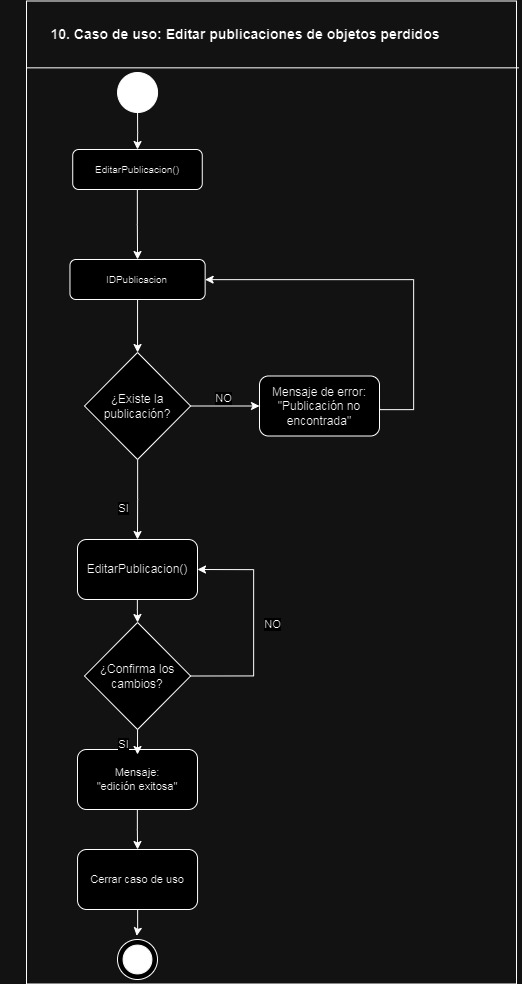


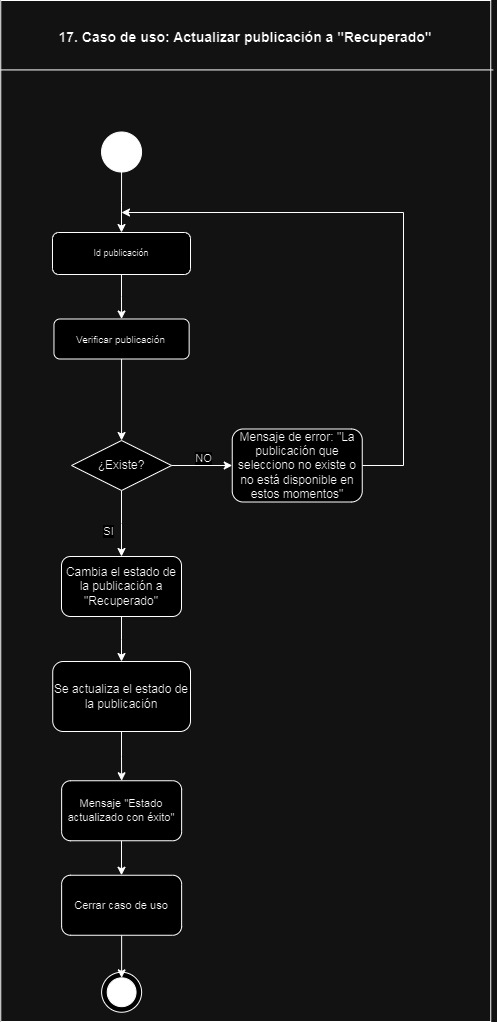
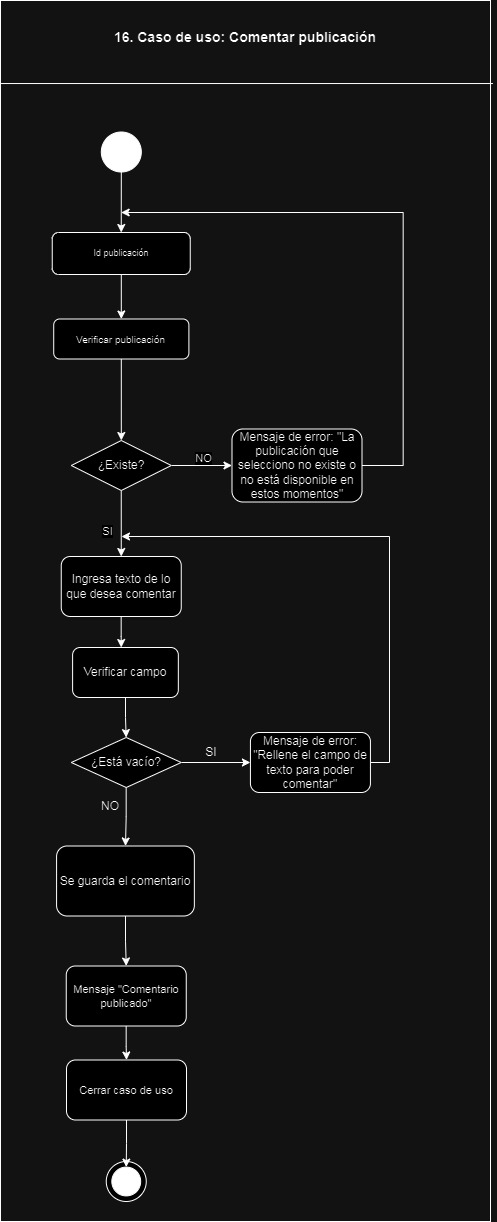
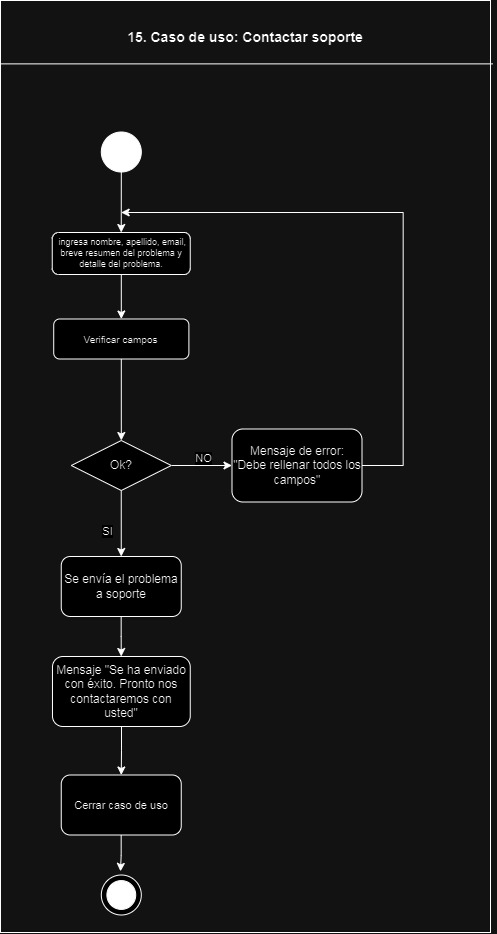
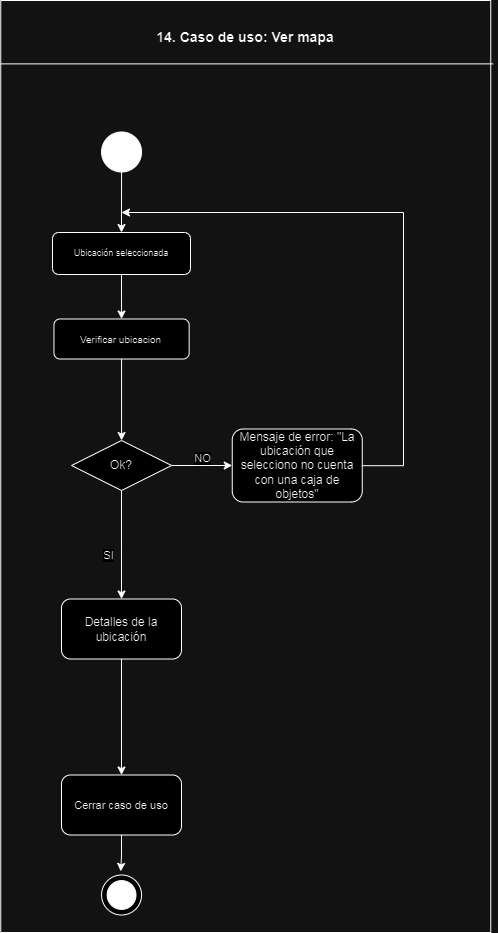
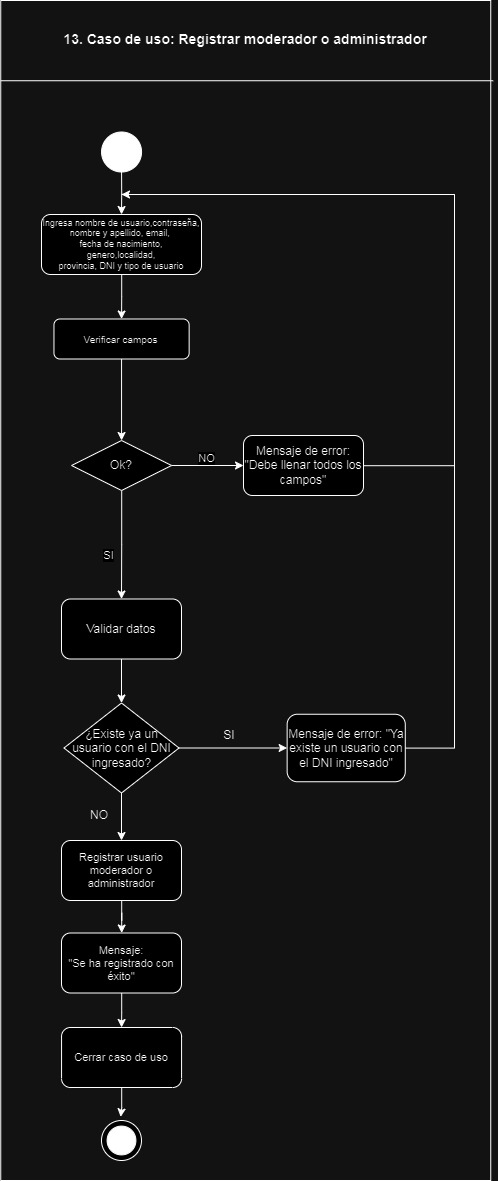
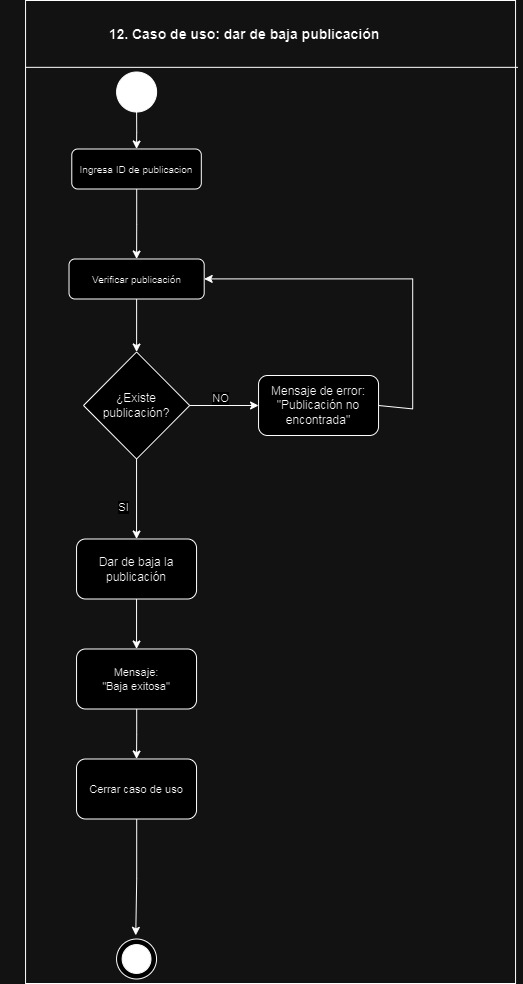
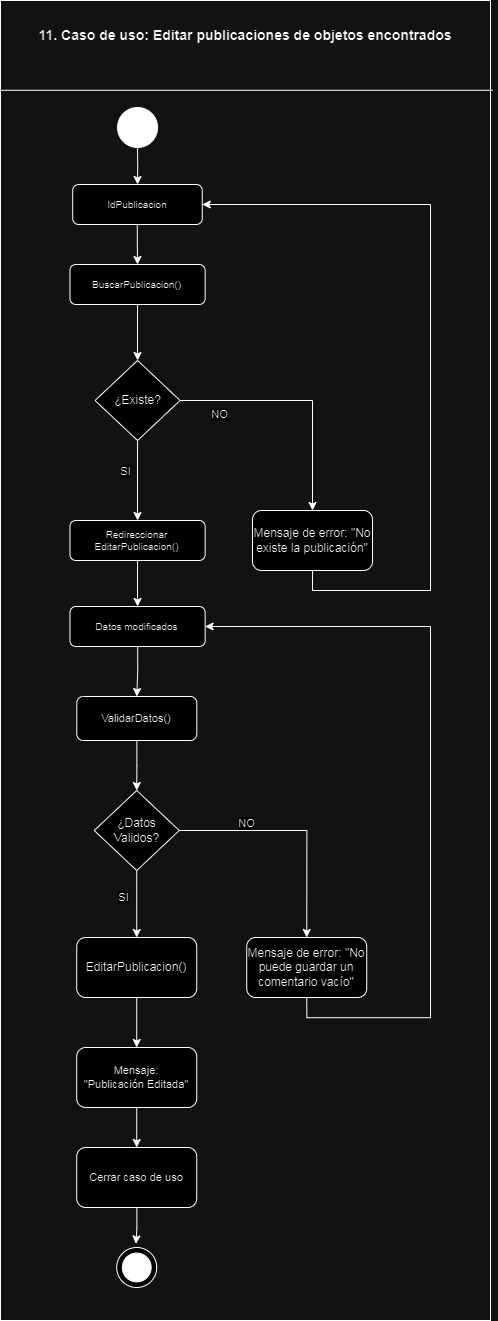


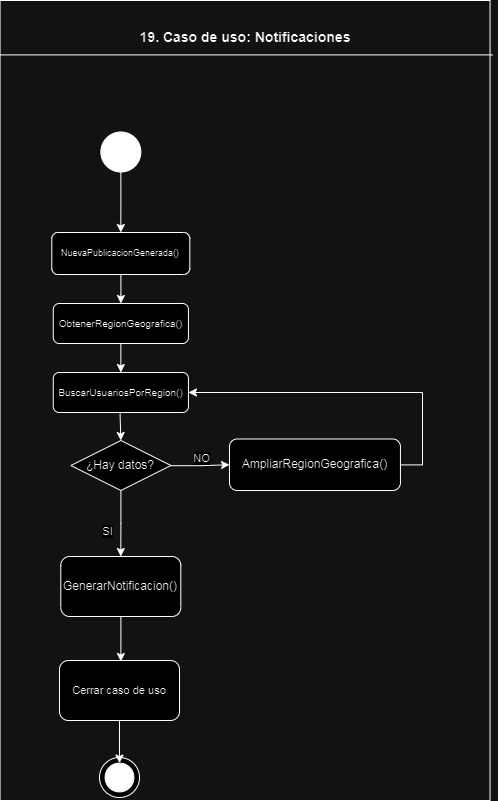
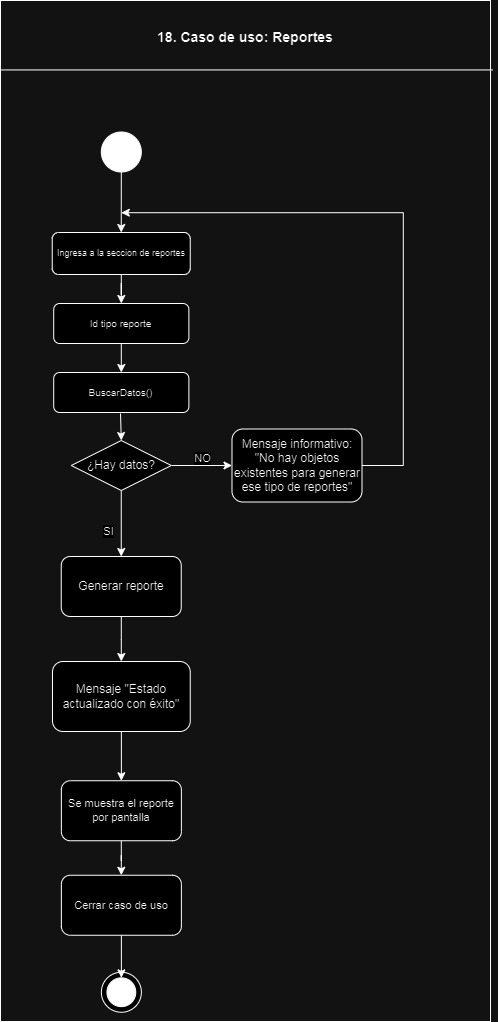
# Diagrama de actividades

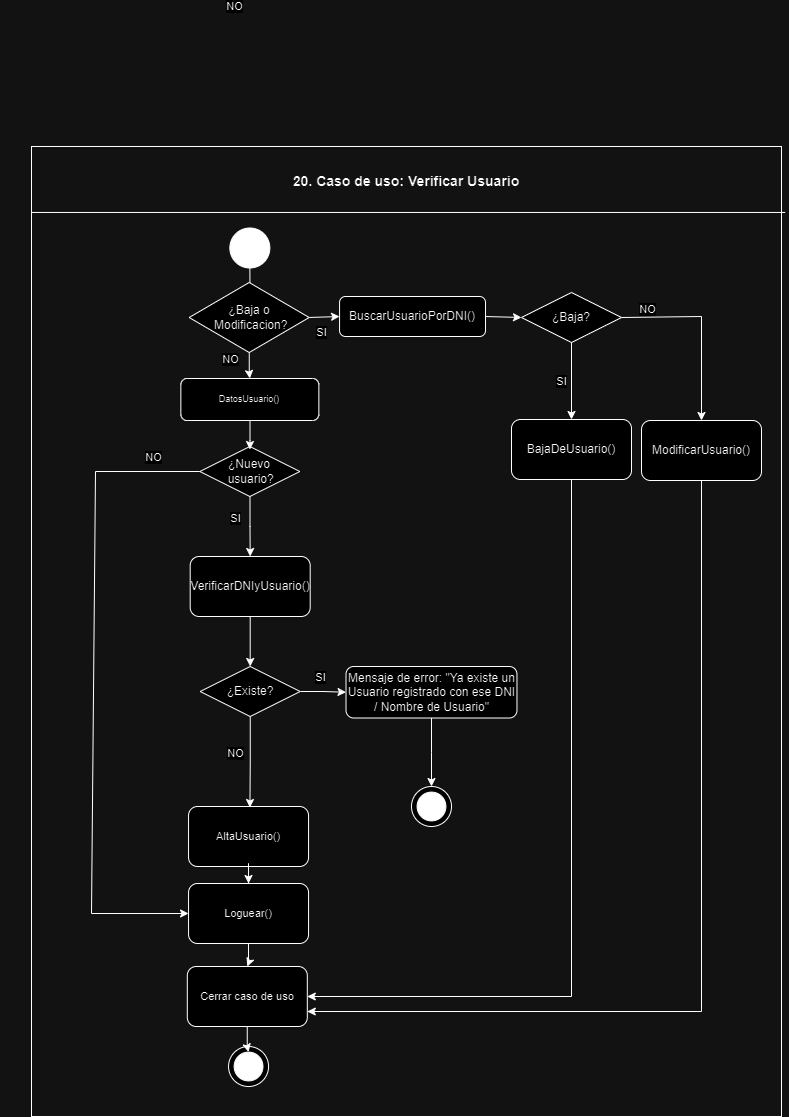


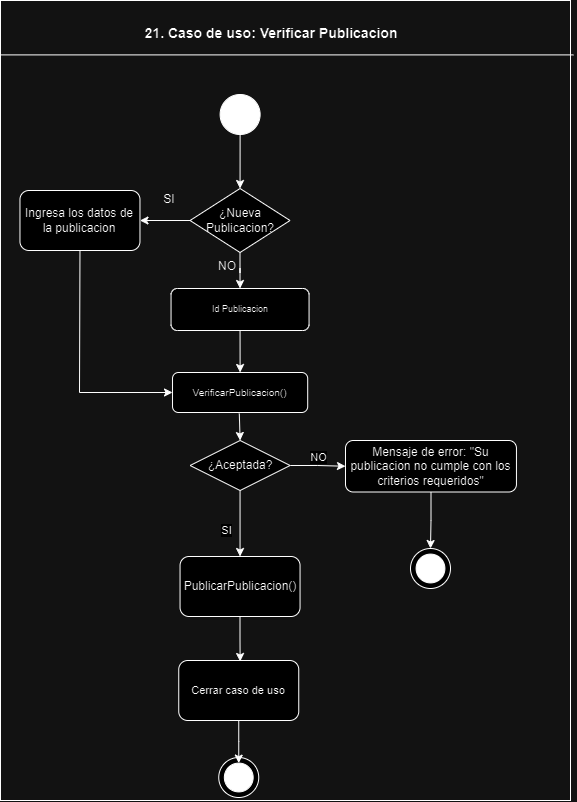




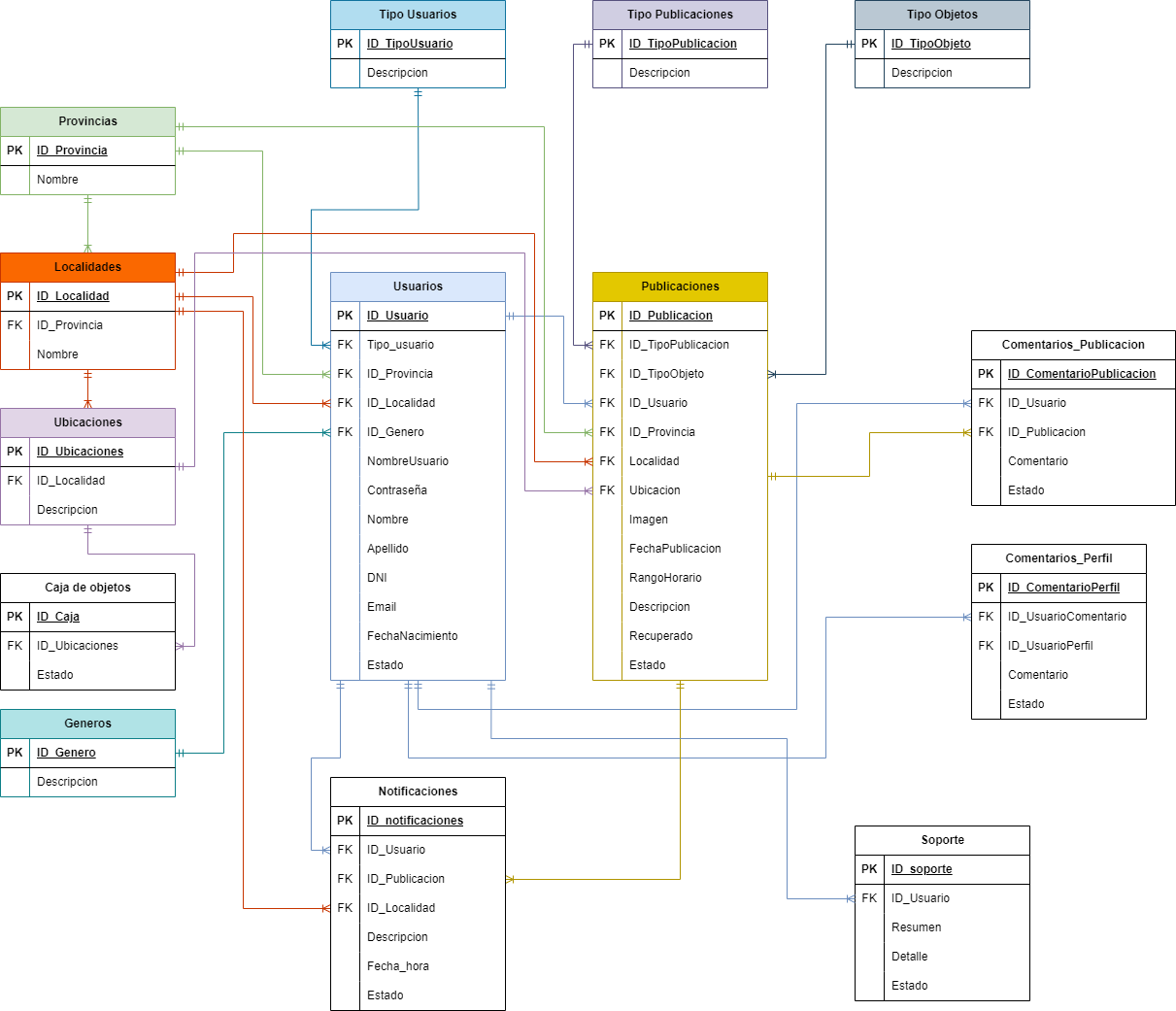






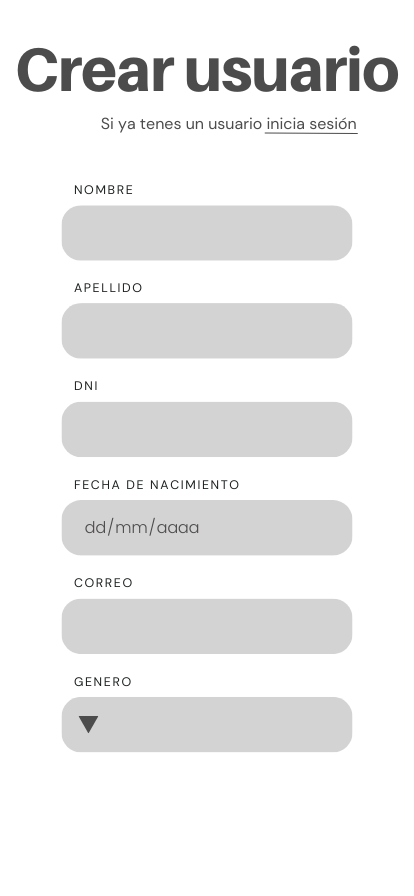


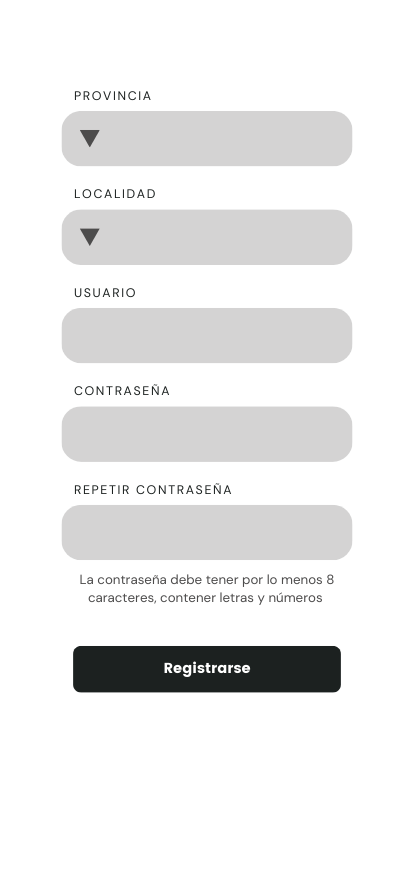
# Diagrama Entidad Relación



# Prototipo de pantallas

Crear usuario

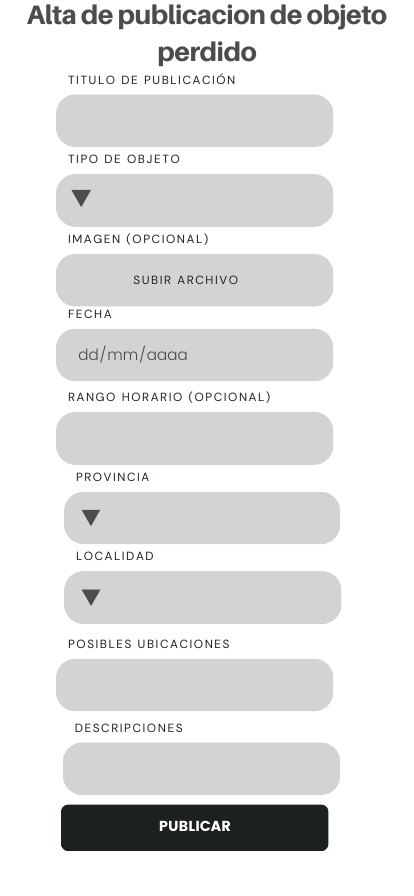




Login



Alta objeto perdido



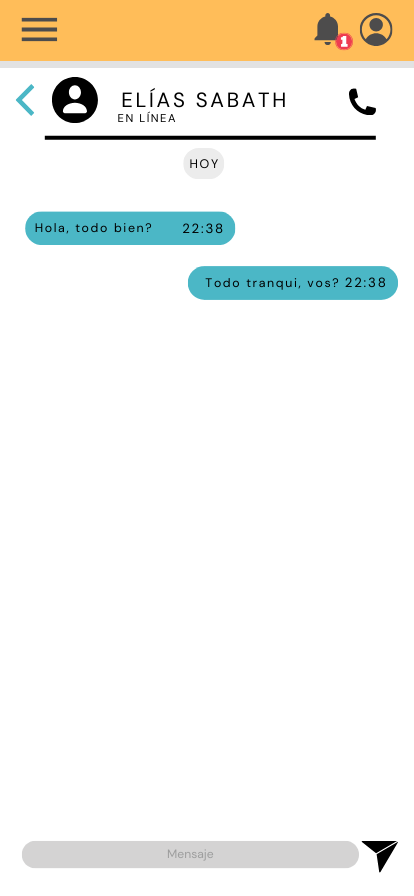
Alta objeto encontrado



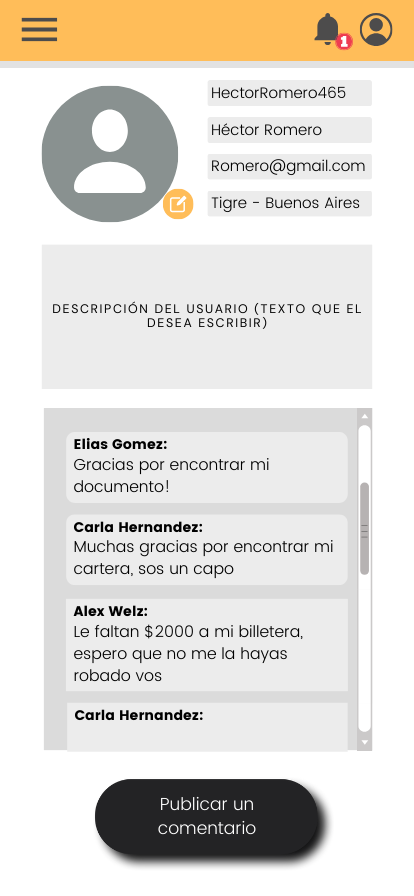
Mensajería interna



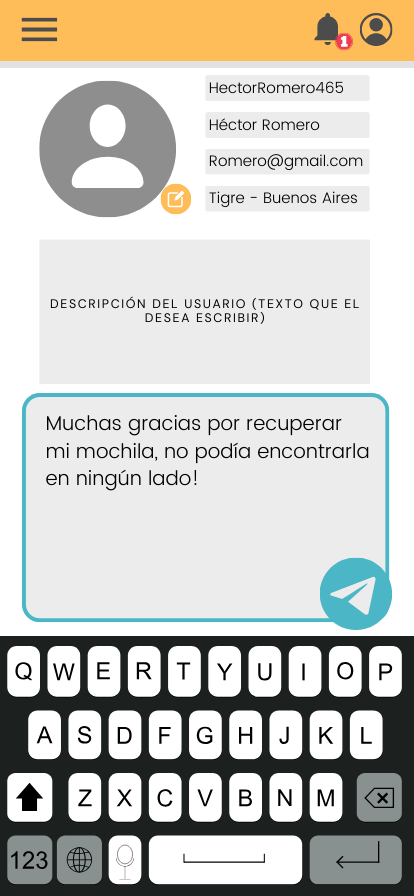
Chat



Perfil usuario



Comentar perfil



Notificaciones



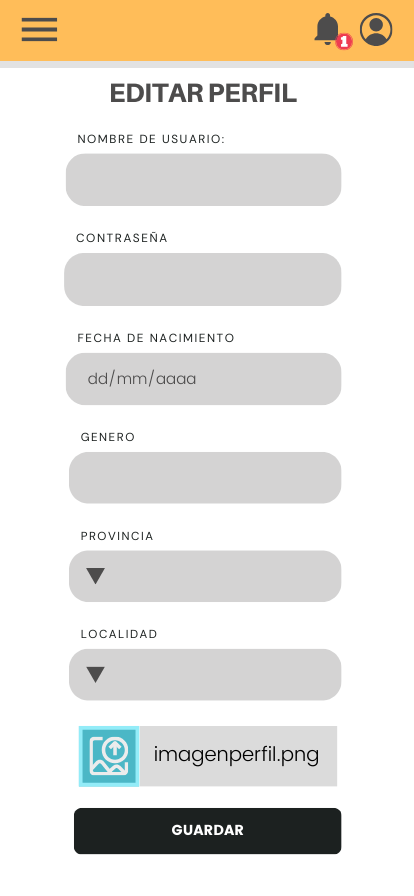
Menú usuario



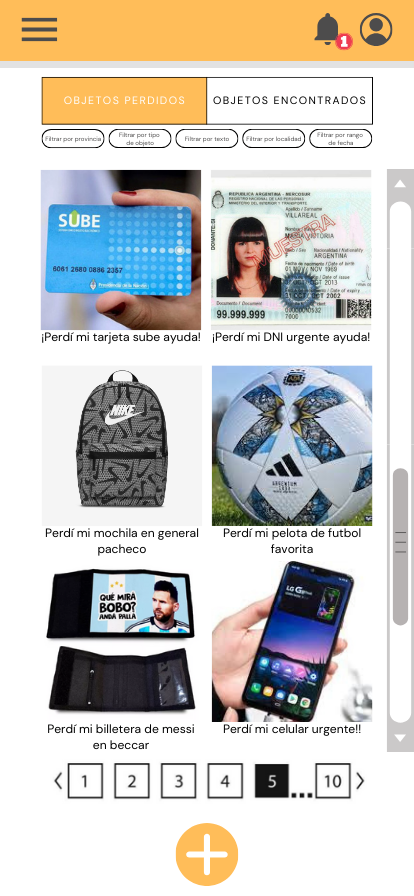
Menú administrador



Editar perfil



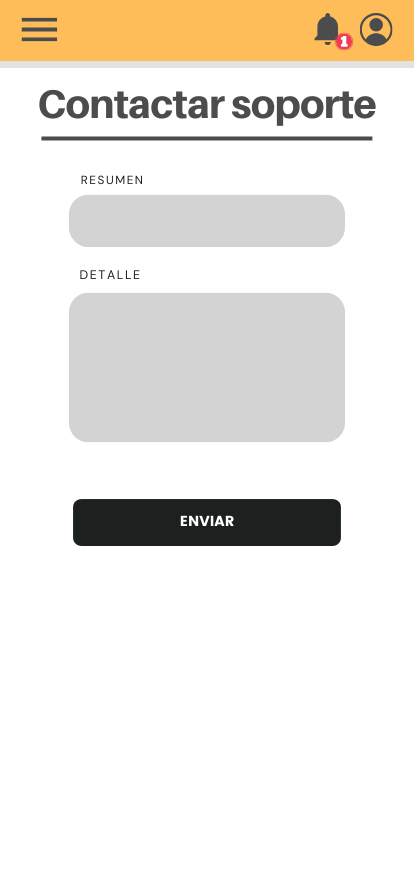
Foro de publicaciones



Editar objeto perdido(misma que la de editar objeto encontrado)



Contactar soporte



Panel de administración



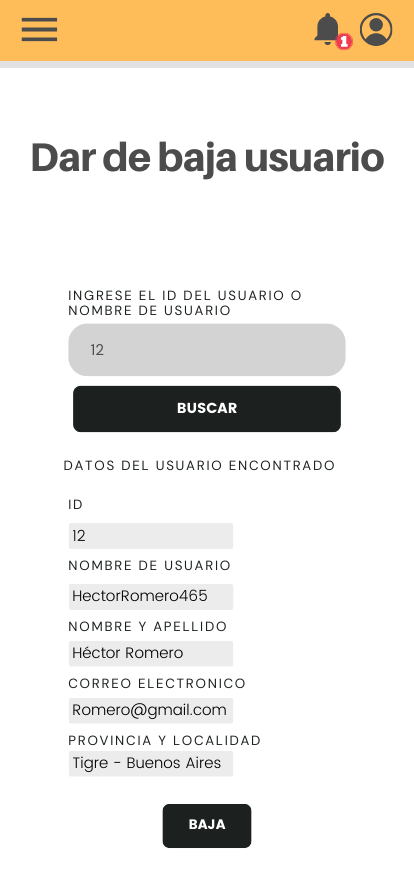
Editar o dar de baja publicación



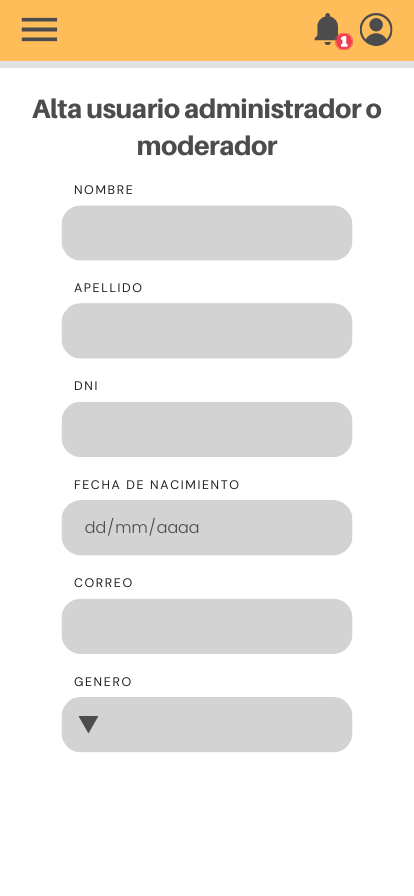
Editar publicación



Dar de baja usuario

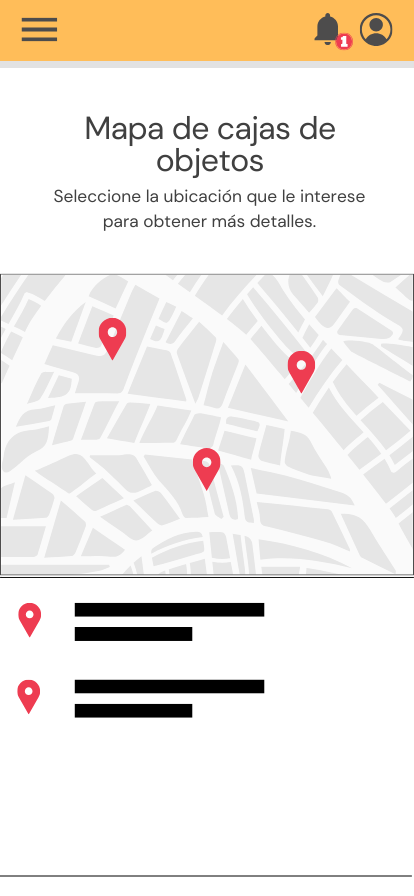


Alta usuario administrador o moderador





Ver mapa



Reportes



# Informe de avance

Fecha 16/09/2023

Objetos Perdidos

Tareas

|  |  |
| --- | --- |
| Entregado | En esta entrega |
| * **Entrega 1:** Propuestas con módulos funcionales, contribución social, plataforma de desarrollo y objeto mensurable: - Animales Perdidos - Objetos Perdidos - EcoTrakcer**ndi** * **Entrega 2**: Se toma la propuesta *Objetos perdidos* y se amplía y desarrollan los módulos funcionales, se detalla información sobre los paneles de la app y su uso. * **Entrega 3**: Construimos las especificaciones de los requerimientos, definiendo casa paso del proceso.  Realizamos un relevamiento consistente con los requerimientos funcionales y no funcionales. * **Entrega 4**: Construimos las especificaciones de los casos de uso * **Entrega 5**: Realizamos los Diagramas de Secuencia y Actividad/Estados * **Entrega 6**: Corregimos los diagramas de secuencia 3 y 4, realizamos el DER y los prototipo de las pantallas. | * Lo primero que hicimos fue realizar las correcciones de los diagramas de secuencia 3 y 4. * Modificamos un poco el diagrama de secuencia y de actividad 15 que serian lo de contactar soporte, un cambio mínimo que sería sus datos de entrada ya que lo veíamos que algunos datos eran innecesarios. * Realizamos el DER con las tablas que consideramos que serian importantes para el funcionamiento del software a construir utilizando la pagina “draw.io”. * Realizamos los prototipos de pantalla de nuestra aplicación móvil con la página “canva”. |

Problemas Presentados

|  |  |
| --- | --- |
| Problema | Detalles |
| * Definir el alcance y tiempo de desarrollo del proyecto. * Definir donde se alojarán los datos de los usuarios en la nube (contratación de servicio externo o servidor propio). * Costear gastos relacionados al alojamiento y mantenimiento del sistema. * Tiempo estimado en la creación de la aplicación. | Se debe definir el alcance de la forma más exacta posible para cumplir los tiempos de desarrollo  Se debe hacer un análisis de costo/beneficio en cuanto al alojamiento de los datos de aplicación y procesamiento de datos de los usuarios  Se debe realizar un análisis en cuanto al costeo de gastos fijos que conlleva contratar servicios de terceros o armar un servidor con gran capacidad de procesamiento.  Se debe considerar el tiempo estimado de la creación de la aplicación. |

Integrantes

|  |  |  |
| --- | --- | --- |
| Legajo | Nombre | E-Mail |
| 27845 | Alexis Juan Carlos Torrente | alexis.torrente@alumnos.frgp.utn.edu.ar |
| 29057 | Francisco Ciarallo | francisco.ciarallo@alumnos.frgp.utn.edu.ar |
| 27853 | Francisco Crestanello | francisco.crestanello@alumnos.frgp.utn.edu.ar |
| 26268 | Gastón Sosa | gaston.sosa@alumnos.frgp.utn.edu.ar |
| 25756 | Ricardo Recofsky | obdulio.adducci@alumnos.frgp.utn.edu.ar |
| 29056 | Sebastián Ré | sebastian.re@alumnos.frgp.utn.edu.ar |