



Universidad Nacional del Nordeste Facultad de Ciencias Exactas y Naturales y  
Agrimensura

## INGENIERÍA DEL SOFTWARE II

Año 2024

**Profesora:** Lic.Laura Gomez Solis

**Integrantes:**

Baez, Débora Jazmín - 44744446  
Canteros, Luciana Belen - 44744256  
Chicla Hardy, Alejandro  
Diaz, Bricia Candela

## Contenido

<b>INTRODUCCIÓN.....</b>	<b>2</b>
Breve estado del arte.....	2
Propósito.....	3
Alcance.....	4
Definiciones acrónimos y abreviaturas.....	4
<b>DESCRIPCIÓN GENERAL.....</b>	<b>4</b>
Funcionalidades del producto.....	4
Características del usuario.....	5
Restricciones.....	5
<b>REQUERIMIENTOS ESPECÍFICOS.....</b>	<b>6</b>
Requerimientos funcionales.....	6
Requerimientos no funcionales.....	6
<b>APÉNDICES.....</b>	<b>7</b>
Ciclo de vida - Metodología.....	7
Planificación de actividades.....	7
Herramientas utilizadas.....	8
Análisis de riesgos.....	8
Identificación y análisis.....	8
Planificación.....	9
Evaluación de arquitectura.....	11
<b>HISTORIAS DE USUARIO.....</b>	<b>11</b>
DIAGRAMA ENTIDAD RELACIÓN.....	16
Casos de Uso.....	16
Diagrama de Secuencia.....	16
CONVERSACIÓN.....	18
Contrato de Operación.....	20

# INTRODUCCIÓN

El presente documento es una Especificación de Requisitos de Software perteneciente al desarrollo de un sistema para la gestión de seguridad para barrios privados. Está estructurado según las directivas dadas por el estándar de IEEE 830. Esta especificación está dirigida a los desarrolladores del Sistema y usuarios finales y tiene como objetivo analizar y documentar los requisitos funcionales y no funcionales del futuro sistema.

## Breve estado del arte

En el contexto de seguridad en los barrios privados, se lleva a cabo el registro y control de ingresos y egresos de personas al lugar de forma manual y por personal de seguridad, a partir de esto es que surge la necesidad de automatizar dichas operaciones.

La propuesta es realizar un sistema genérico para la administración de dichas entradas y salidas de individuos, con el objetivo de agilizar la tarea frecuente de control y mejorar la seguridad.

En la actualidad, la seguridad es un tema de gran importancia en la sociedad, y los barrios cerrados no son la excepción. Estos se caracterizan por contar con una seguridad más elevada en comparación con otras zonas urbanas, ya que se encuentran rodeados de muros y cercas, y cuentan con personal de seguridad que controla los accesos. Sin embargo, que la gestión de los movimientos se realice manualmente puede resultar poco eficiente y seguro.

Este funcionamiento puede llevar a la pérdida de tiempo, propiciar errores humanos, generar inconsistencias y demás escenarios que pueden comprometer la seguridad del barrio.

La implementación de un software, puede ser la solución para garantizar un control de acceso más eficiente, seguro y preciso. Este software puede incluir características como identificación, registro de vehículos y alertas, lo que puede mejorar significativamente la seguridad y la eficiencia de la gestión de acceso.

## Propósito

El propósito de la propuesta es la de realizar una aplicación de escritorio para la gestión de accesos y egresos al barrio cerrado en concreto.

El propósito del software es mejorar la seguridad y la eficiencia en la gestión de los ingresos y egresos de personas en los barrios privados. Al automatizar este proceso, el software permite

un control de acceso más preciso y confiable, reduciendo los errores humanos y disminuyendo el riesgo de incidentes de seguridad. Además, el software permite un registro y monitoreo más detallado de los accesos como así también pueden procesar más rápidamente los ingresos y egresos, lo que reduce el tiempo de espera para los residentes y visitantes.

## **Alcance**

**Gestión de visitantes:** La aplicación debe permitir el registro y gestión de visitantes. Los residentes podrán solicitar permisos para sus invitados, proporcionando información como el nombre de los visitantes, la fecha de ingreso y los detalles del vehículo si es necesario. Además, los permisos de ingreso serán notificados a los propietarios quienes podrán aceptar o no a través del software.

**Registro de ingresos y egresos:** La aplicación debe llevar un registro detallado de los ingresos y egresos de personas y vehículos en el barrio cerrado. Esto incluirá la fecha, hora y motivo de la visita, así como la información del vehículo y la identificación de la persona que ingresó o salió.

**Informes**

**Solicitudes y alertas:** La aplicación puede enviar solicitudes y alertas a los residentes sobre el ingreso de visitantes autorizados, así como cualquier evento de seguridad relevante.

## **Definiciones acrónimos y abreviaturas**

ABM - Alta, baja y modificación.

RF : requerimiento funcional

RNF: requerimiento no funcional

**Sprint:** Periodo de tiempo limitado (generalmente de una a cuatro semanas) durante el cual se lleva a cabo un trabajo específico en un proyecto, utilizado en metodologías ágiles de desarrollo de software como Scrum.

## DESCRIPCIÓN GENERAL

### Funcionalidades del producto

- Gestión de ingresos y egresos al barrio
- Gestión de propietarios
- Programar ingresos
- Generar informes

### Características del usuario

<b>Tipo de usuario</b>	<b>Administrador</b>
<b>Habilidades</b>	<b>Conocimientos básicos de computación</b>
<b>Actividades</b>	<b>Ingreso al sistema, ABM de guardias, ABM de propietarios, ABM de propiedades, visualización de historial de ingresos y egresos al barrio</b>

<b>Tipo de usuario</b>	<b>Propietario</b>
<b>Habilidades</b>	<b>Conocimientos básicos de computación</b>
<b>Actividades</b>	<b>Ingreso al sistema, programación de ingresos para invitados, visualización de historial de ingresos y egresos a su nombre.</b>
<b>Tabla 2.3.2</b>  Características del usuario Propietario <b>Tipo</b>  <b>de usuario</b>	<b>Guardia</b>
<b>Habilidades</b>	<b>Conocimientos básicos de computación</b>
<b>Actividades</b>	<b>Ingreso al sistema, confirmación y registro de ingresos y egresos, visualización de historial de ingresos y egresos al barrio.</b>

## **Restricciones**

El sistema será de uso exclusivo para usuarios registrados pertenecientes al barrio, y garantizará la exclusión de personas ajenas al sistema.

El sistema no tendrá un uso excesivo de recursos de hardware en términos de memoria, CPU y ancho de banda de red.

Los tiempos de respuesta no afectarán al funcionamiento normal de la administración del barrio.

El software garantizará la privacidad y seguridad de los datos de los residentes del barrio cerrado, y debe cumplir con las leyes y regulaciones de privacidad y seguridad de datos.

## **REQUERIMIENTOS ESPECÍFICOS**

### **Requerimientos funcionales**

Control de ingresos y egresos

REF #1 El sistema debe permitir al guardia enviar al propietario una solicitud de ingreso en su nombre.

REF #2 El sistema debe permitir al propietario visualizar las solicitudes de ingreso a su nombre.

REF #3 El sistema debe permitir guardar datos del visitante: nombre, apellido, DNI, fecha de ingreso y detalles de vehículo en caso de ser necesario.

REF #4 El sistema debe permitir notificar al propietario que el visitante autorizado ingresó efectivamente.

REF #5 El sistema debe permitir mostrar un informe de movimientos.

REF #6 El sistema debe permitir visualizar los visitantes que actualmente se encuentren dentro del predio.

REF #7 El sistema debe permitir guardar registro de la salida de un visitante.

Programación de visitas

REF #8 El sistema debe permitir al propietario programar anticipadamente la entrada de un visitante.

REF #9 El sistema debe mostrar el listado de visitas programadas.

Administración de usuarios:

REF #10 El sistema deberá permitir crear un usuario.

REF #11 El sistema deberá permitir deshabilitar un usuario.

REF #12 El sistema deberá permitir mostrar un listado de los usuarios registrados.

## Requerimientos no funcionales

De producto:

RNF #1 Las fechas deben registrarse de manera automática.

RNF #2 El sistema debe ser escalable y permitir la adición de nuevas funcionalidades y características en el futuro sin afectar la estabilidad y el rendimiento del sistema existente.

RNF #3 El sistema debe garantizar la seguridad de los datos almacenados y procesados y la privacidad de los usuarios.

RNF #4 El sistema debe ser fácil de usar e intuitivo para los usuarios, incluyendo los guardias de seguridad y los residentes del barrio privado.

RNF #5 El sistema debe contar con un equipo de soporte técnico que pueda brindar asistencia y solucionar problemas de manera rápida y eficiente.

Organizacionales:

RNF #7 La especificación de requerimientos de software se deberá realizar según la norma IEEE 830.

## APÉNDICES

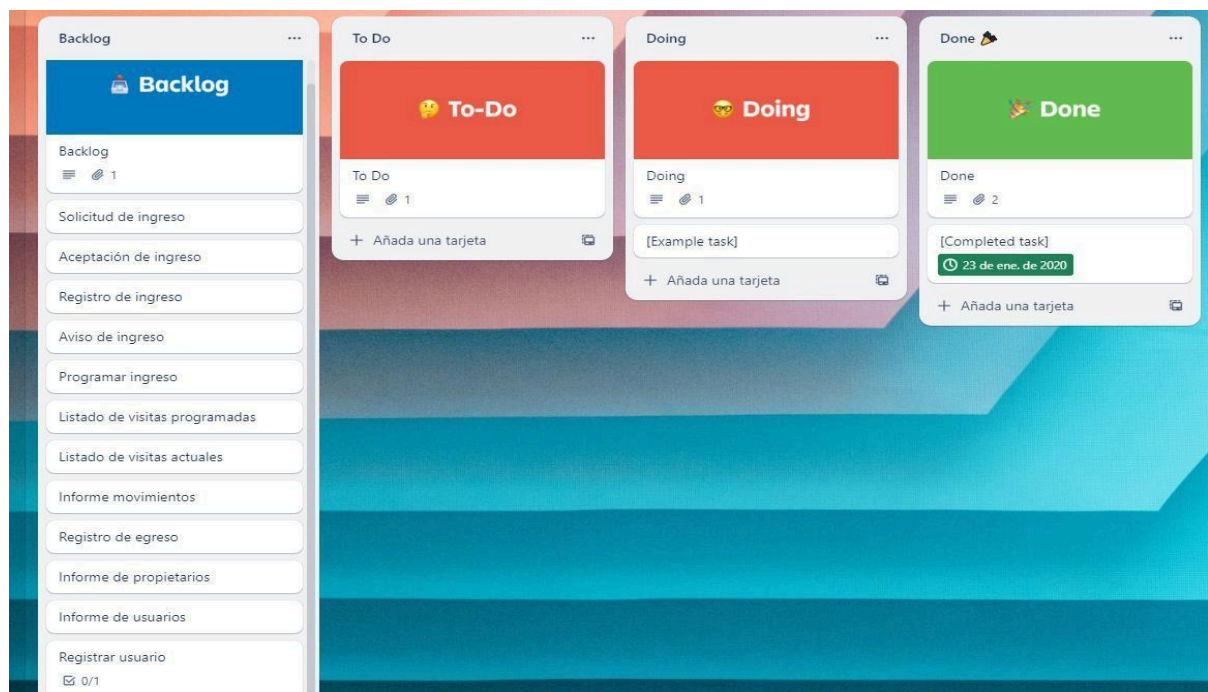
### Ciclo de vida - Metodología

Para el desarrollo del proyecto se tomó la decisión de implementar la metodología “SCRUM” debido a que, como metodología de trabajo puede ofrecer varios beneficios, como la mejora de la comunicación entre los miembros del equipo, la entrega rápida y frecuente de software funcional y la capacidad de responder a las necesidades y requerimientos del cliente de manera efectiva.

Se trabajará en sprints,, en los que se entregan incrementos funcionales del software. Durante cada sprint, el equipo se reúne semanalmente en una reunión de Scrum para revisar el progreso y planificar el trabajo del siguiente

### Planificación de actividades

Todas las tareas a realizar se ubicaron en una pila de producto



## Herramientas utilizadas

Las herramientas utilizadas fueron:

- Discord: para llevar a cabo las reuniones.
- Trello: como organizador de actividades.
- Draw.io: para la creación de diagramas

## Análisis de riesgos

### Identificación y análisis

Descripción	Clasificación	Análisis
El software no cumple con las normas de privacidad y protección de datos, lo que podría exponer la información personal de los residentes y visitantes del barrio cerrado.	Producto / tecnología	Probabilidad: media Incidencia: Alta
El software no proporciona la funcionalidad requerida por los usuarios, lo que reduce su utilidad y valor para el barrio cerrado.	Producto / requerimientos	Probabilidad: baja Incidencia: Alta
El equipo de desarrollo enfrenta problemas de comunicación y coordinación, lo que retrasa el desarrollo y la implementación del software.	Proyecto / personal	Probabilidad: Alta Incidencia: Media
El sistema lleva más tiempo de lo esperado por lo que se retrasan las entregas.	Proyecto / estimación	Probabilidad: Alta Incidencia: Media



El equipo de desarrollo no tiene suficiente experiencia en el desarrollo de software de seguridad, lo que aumenta el riesgo de errores y vulnerabilidades en el software.	Proyecto / personal	Probabilidad: Media Incidencia: Baja
El software puede ser costoso de desarrollar y mantener, lo que puede afectar negativamente la rentabilidad del proyecto.	Negocio / organizacional	Probabilidad: Alta Incidencia: Media
Las fallas en la seguridad del software pueden afectar la confianza del cliente y la reputación de la empresa, lo que puede afectar negativamente las ventas y la adopción del software.	Negocio / organizacional	Probabilidad: Media Incidencia: Alta

## Planificación

Descripción	Gestión
Una vulnerabilidad de seguridad no identificada en el software puede ser explotada por atacantes externos para acceder al barrio cerrado.	Realizar pruebas rigurosas de seguridad en el software antes de implementarlo. Además, mantener actualizado el software con parches y actualizaciones de seguridad para mitigar posibles vulnerabilidades.
El software no cumple con las normas de privacidad y protección de datos, lo que podría exponer la información personal de los residentes y visitantes del barrio cerrado.	Garantizar que el software cumpla con las normas y regulaciones de privacidad y protección de datos. Además, establecer medidas de seguridad adicionales, como la encriptación de datos y el control de acceso a la información.

El software no proporciona la funcionalidad requerida por los usuarios, lo que reduce su utilidad y valor para el barrio cerrado.	Realizar una investigación exhaustiva de los requisitos de los usuarios y asegurarse de que el software proporcione la funcionalidad necesaria. Además, realizar pruebas de usuario y recopilar comentarios y sugerencias para mejorar el software a lo largo del tiempo.
El equipo de desarrollo enfrenta problemas de comunicación y coordinación, lo que retrasa el desarrollo y la implementación del software.	Estrategia de contingencia: Implementar herramientas y metodologías de gestión de proyectos eficaces para fomentar una comunicación clara y una coordinación efectiva entre el equipo de desarrollo y los stakeholders.
El equipo de desarrollo no tiene suficiente experiencia en el desarrollo de software de seguridad, lo que aumenta el riesgo de errores y vulnerabilidades en el software.	Estrategia de contingencia: Contratar o asociarse con expertos en seguridad informática para asesorar al equipo de desarrollo y proporcionar capacitación adicional.
El software puede ser costoso de desarrollar y mantener, lo que puede afectar negativamente la rentabilidad del proyecto.	Realizar una planificación y gestión de proyecto efectivas para minimizar los costos y maximizar la eficiencia en el desarrollo y mantenimiento del software. Además, explorar opciones de financiamiento y reducir la carga financiera.
Las fallas en la seguridad del software pueden afectar la confianza del cliente y la reputación de la empresa, lo que puede afectar negativamente las ventas y la adopción del software.	Realizar pruebas de seguridad exhaustivas en el software y trabajar con expertos en seguridad para garantizar que se aborden las vulnerabilidades y fallas de seguridad. Proporcionar soporte y actualizaciones

## EVALUACIÓN DE ARQUITECTURA

El proyecto requiere la comunicación de algunos componentes en tiempo real, como lo es el visualizado de las solicitudes de ingreso emitidas. Por otro lado, el resto de la aplicación se maneja obteniendo la información necesaria cuando la necesita, es por ello que se hizo el enfoque en dos arquitecturas para el correcto funcionamiento del sistema.

1. Arquitectura cliente/servidor: Para el funcionamiento de los componentes que no necesitarán estar al tanto en todo momento de la actualización de los datos, si no, cuando se lo requiera únicamente. Estos componentes son todos los ABM de la aplicación, inicios de sesión, modificaciones de horarios laborales, etc.
2. Arquitectura Publicador - Suscriptor: Una arquitectura pensada para el manejo de datos en tiempo real entre componentes. Hemos pensado en utilizar el protocolo WebSockets, pero, si bien el sistema hasta el alcance que fue planteado se conformaría con WebSockets al ser una comunicación más directa y eficiente en términos de rendimiento, hemos pensado que en primer lugar, se pueden seguir añadiendo funcionalidades de tiempo real a la aplicación, en segundo lugar, hay varios componentes que requieren estar a la escucha de un mismo evento (Donde toma ventaja la arquitectura Publicador suscriptor) y además la posibilidad de tener tópicos retenidos para garantizar el envío de mensajes a todos los clientes y no dejar que se pierdan en el camino por alguna falla y/o que el cliente haya estado desconectado en el momento que se produjo el mensaje. Y en tercer lugar el bajo acoplamiento que permite esta arquitectura garantiza que en futuras mejoras de la aplicación sea mucho más fácil agregar funcionalidades en tiempo real.

## HISTORIAS DE USUARIO

ID: 1	Solicitud de ingreso
Yo como guardia quiero poder ingresar nombre, apellido y DNI de un solicitante para dar aviso al propietario correspondiente y esperar su autorización.	
Estimación: 5 días	Valor: Alta
Criterios de aceptación:  Todos los campos deben estar completados, de lo contrario mostrar leyenda "campo incompleto"  Todos los datos deben estar en el formato correcto, caso contrario mostrar mensaje de error.  El propietario que se desea visitar debe existir.  En caso de que el propietario no haya generado una solicitud a modo de permiso de ingreso, el guardia deberá esperar la autorización del mismo para aceptar la entrada.	

ID: 2	Aceptación de ingreso	
Yo como propietario quiero poder ver solicitudes de ingreso realizadas a mi nombre		
Estimación: 5 días		Valor: Alta
Criterio de aceptación:  El aviso debe ser emitido por un guardia, de lo contrario no aparecerá en la lista La notificación debe permitir únicamente dos estados (autorizado – no autorizado) La solicitud debe mostrar todos los datos del solicitante.		

ID: 3	Registro de ingreso	
Yo como guardia quiero poder registrar los datos del solicitante y dar aviso de que ha ingresado para que queden guardados en caso de necesitarlos.		
Estimación: 2 días		Valor: Medio
Criterio de aceptación:		
El propietario debe haber autorizado el ingreso caso contrario no permitir levantar la barrera vehicular.		
La fecha del movimiento debe quedar registrada automáticamente		

ID: 4	Aviso de ingreso	
Yo como guardia quiero poder dar aviso de ingreso a un propietario para que esté al tanto de que la persona a la que autorizó ha ingresado.		
Estimación: 2 días		Valor: Medio
Criterio de aceptación: La notificación de ingreso debe ser enviada al propietario tan pronto como la persona autorizada ingrese al barrio cerrado.		

ID: 5	Programar ingreso	
Yo como propietario quiero poder programar con anticipación el ingreso de una persona ingresando la fecha, nombre, apellido y DNI para que el guardia lo autorice automáticamente sin preguntar.		
Estimación: 10 días		Valor: Alta

<p>Criterio de aceptación:</p> <p>Todos los campos deben estar completados, de lo contrario mostrar leyenda “campo incompleto”</p> <p>Todos los datos deben estar en el formato correcto, caso contrario mostrar la leyenda “dato incorrecto”.</p> <p>La fecha de ingreso debe ser igual o mayor que la actual, caso contrario mostrar la leyenda “La fecha de ingreso debe ser igual o mayor que la actual”.</p>
---

ID: 6	Listado de visitas programadas
Yo como guardia quiero poder visualizar las visitas programadas por los propietarios para poder autorizar el ingreso a dichas visitas cuando se presenten en la entrada sin preguntar al propietario.	
Estimación: 5 días	Valor: Media
<p>Criterio de aceptación:</p> <p>La información de las visitas programadas debe ser precisa y actualizada en tiempo real para que el guardia pueda autorizar el ingreso de las visitas en el momento adecuado.</p> <p>La interfaz de visualización de visitas programadas debe ser fácil de usar y comprender para el guardia.</p> <p>La información de las visitas programadas debe incluir el nombre del propietario, el nombre de la visita, la fecha y hora programada, y cualquier otra información relevante necesaria para la autorización.</p>	

ID: 7	Listado de visitas actuales
Yo como guardia quiero poder visualizar las personas no propietarias que se encuentran en este momento dentro del barrio para llevar control.	
Estimación: 2 días	Valor: Media
<p>Criterio de aceptación:</p> <p>En la lista deben aparecer únicamente personas que aún no abandonaron el barrio.</p> <p>Debe haber registros de ingresos previamente, caso contrario mostrar la leyenda “no se hallaron ingresos registrados”.</p> <p>La lista debe actualizarse automáticamente cuando ingresa alguien.</p>	

ID: 8	Registro de egreso
Yo como guardia quiero poder registrar el egreso y detallar una observación para dejar constancia de que la persona abandonó el predio.	
Estimación: 5 días	Valor: Baja
Criterio de aceptación:  La persona que se pretende registrar debe estar en la lista de ingresos.  La observación debe tener menos de 140 caracteres, caso contrario mostrar la leyenda “nro. de caracteres excedidos”.  El registro del tiempo y fecha debe quedar registrado automáticamente.	

ID: 9	Informe movimientos
Yo como dueño del barrio quiero poder visualizar un informe con los movimientos realizados incluyendo nombre, apellido y dni del ingresante, propietario responsable y guardia a cargo, como así también el tipo de movimiento (ingreso – egreso) en el barrio en una determinada fecha para llevar control en caso de ser necesario.	
Estimación: 2 días	Valor: Alta
Criterio de aceptación:  La fecha ingresada debe ser menor o igual a la actual, caso contrario mostrar la leyenda “La fecha ingresada debe ser menor o igual a la actual”.  Se deben haber registrado movimientos previamente, caso contrario mostrar “no se han registrado movimientos a la fecha”.	
ID: 10	Informe de usuarios
Yo como dueño del barrio quiero poder visualizar un informe de los usuarios incluyendo nombre, apellido y DNI para llevar control en caso de ser necesario.	
Estimación: 2 días	Valor: Media
Criterio de aceptación:  Debe haber propietarios cargados, caso contrario mostrar “no se han registrado propietarios”.	

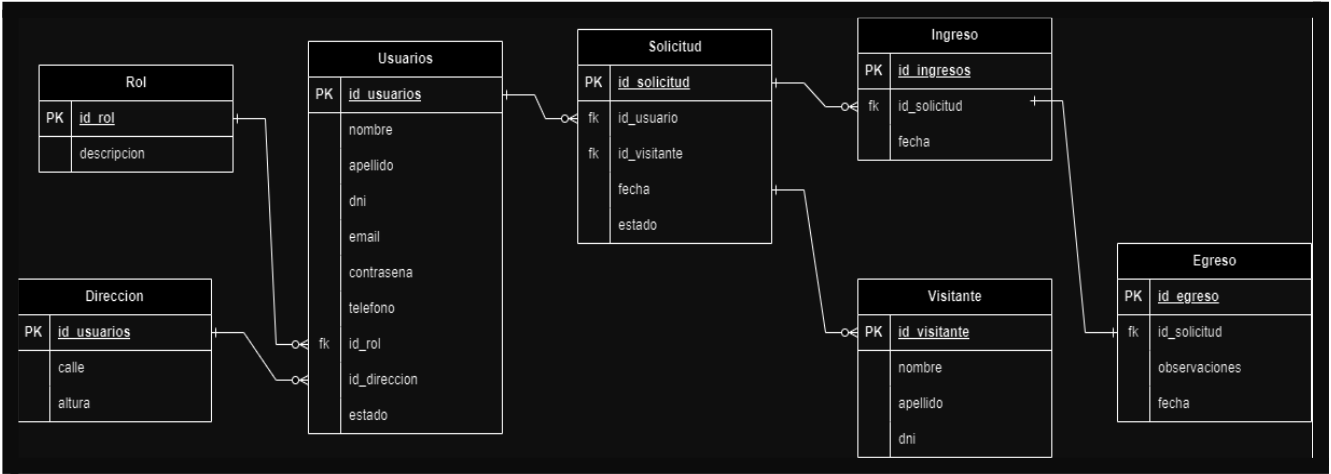
ID: 11	Registrar usuario
Yo como administrador de seguridad, quiero poder crear una cuenta de usuario para los residentes del barrio cerrado proporcionando un nombre de usuario y una contraseña para que el propietario luego pueda acceder a la plataforma de seguridad	

Estimación: 10 días	Valor: Alta
<p>Criterio de aceptación:</p> <p>El sistema debe verificar que el nombre de usuario no esté en uso antes de crear la cuenta de usuario, caso contrario mostrar la leyenda “nombre de usuario en uso”.</p> <p>La contraseña proporcionada por el administrador debe tener un mínimo de 8 caracteres, caso contrario mostrar la leyenda “la contraseña debe tener al menos 8 caracteres”.</p> <p>El sistema debe permitir cambiar una contraseña en cualquier momento después de la creación de la cuenta de usuario.</p>	

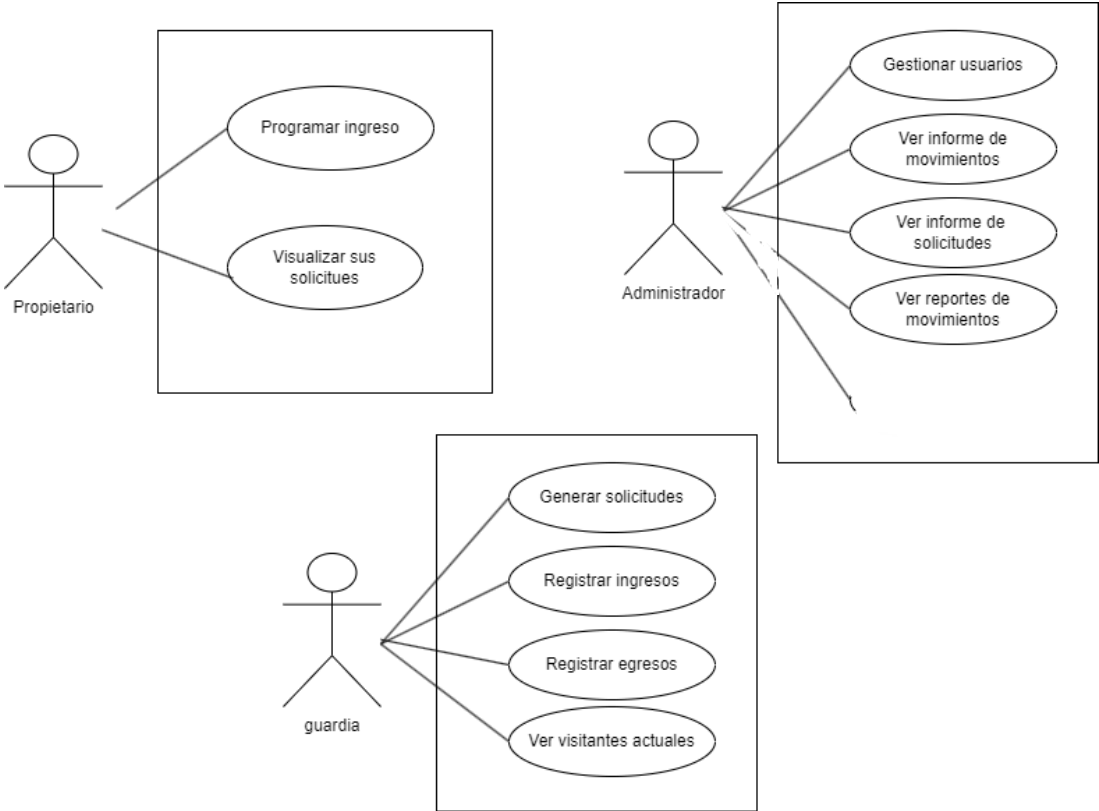
ID:12	Dar de baja usuario
Yo como administrador, quiero poder buscar una cuenta por DNI para poder eliminarla.	
Estimación: 4 días	Valor: Media
<p>Criterio de aceptación:</p> <p>El DNI proporcionado debe pertenecer a algún usuario registrado, caso contrario mostrar leyenda “no se encontraron usuarios”</p> <p>El sistema debe mostrar el usuario que coincida con ese DNI.</p> <p>El sistema debe requerir que se proporcione una confirmación de seguridad antes de eliminar la cuenta de usuario de un usuario encontrado a través de la búsqueda.</p>	

# DIAGRAMAS

## Diagrama entidad relación



## Casos de Uso

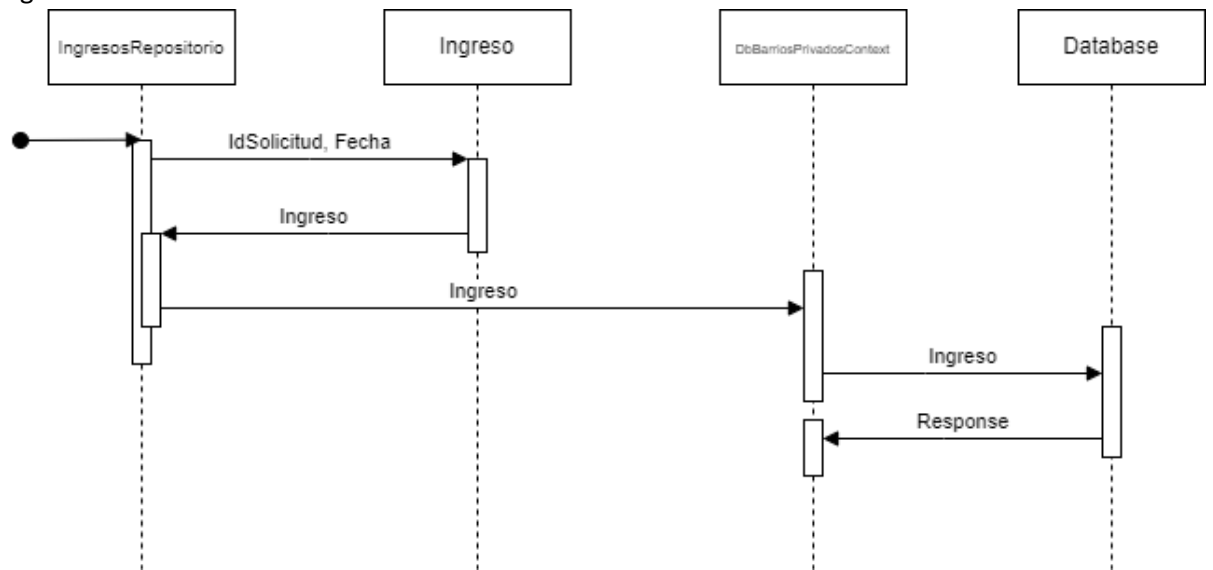




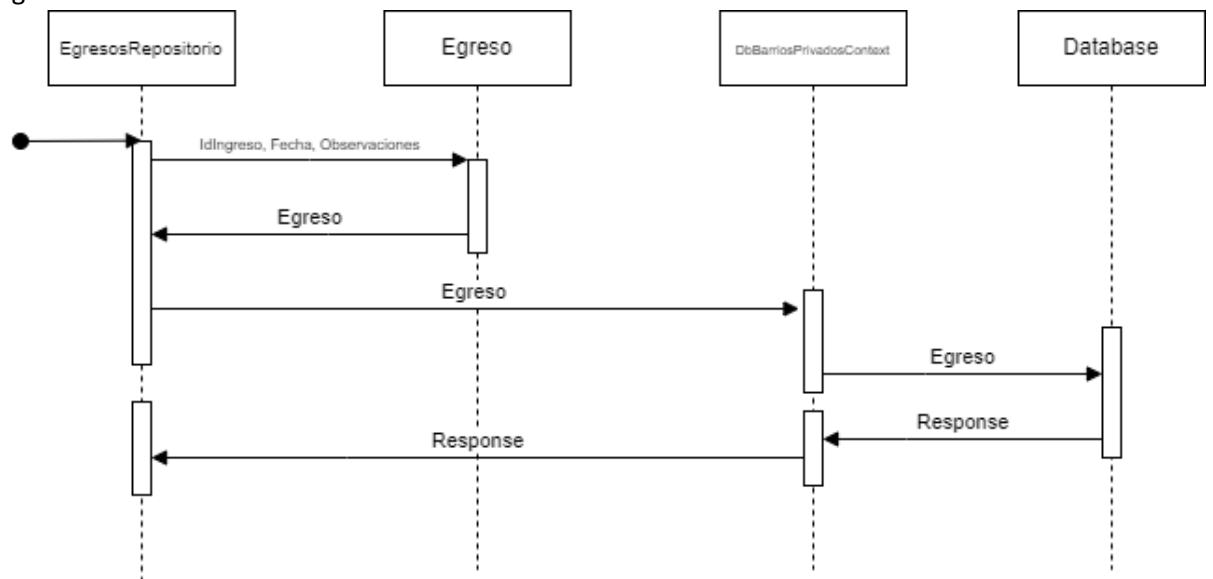
## Diagrama de Secuencia

Funcionalidades:

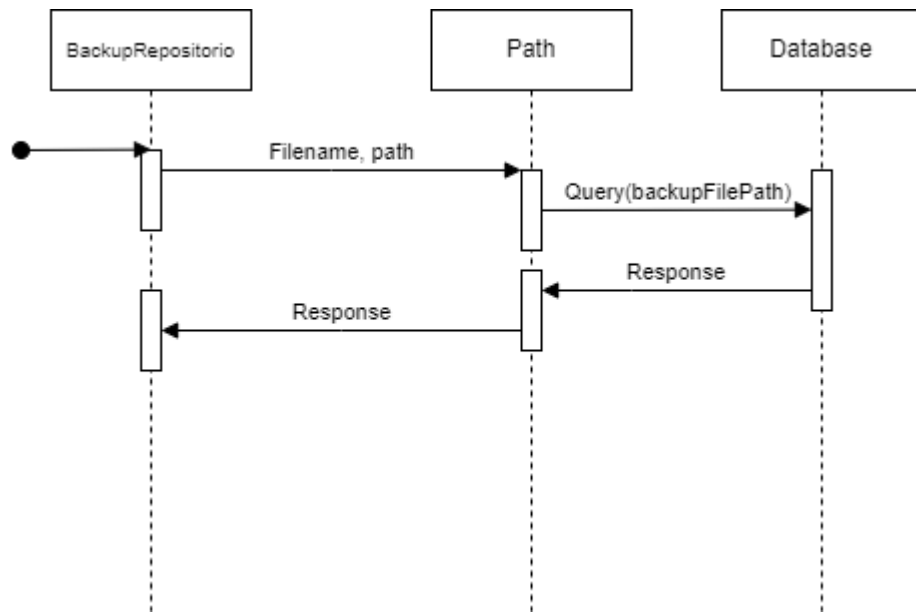
### 1. Ingresos



### 2. Egresos



### 3. Backup de la base de datos



## Conversación

Funcionalidades:

### 1. Petición de Ingreso (Guardia)

Acción	Caso Normal	Caso Alternativo
1. Ingresa un solicitante.		
2. El guardia introduce los datos del solicitante.		
3. El sistema verifica los datos ingresados.	3. 1. Campos completos.	3. 2. Campos incompletos. 3. 3. Muestra leyenda "campo incompleto" 3. 4. Vuelve al paso 2.
4. El sistema verifica que el formato sea correcto.	4. 1. Formato correcto.	4. 2. Formato incorrecto. 4. 3. Muestra leyenda "Dato incorrecto" 4. 4. Vuelve al paso 2.
5. Envía solicitud al propietario.	5. 1. Autorización aceptada.	5. 2. Autorización denegada. 4. 3. Muestra leyenda "Solicitud no autorizada" 4. 4. Fin del caso.
6. Autorización del ingreso.		
7. Realiza aviso de		

ingreso.		
8. Registra ingreso.		
9. Fin del caso de uso.		

## 2. Programar ingreso (Propietario)

<b>Acción</b>	<b>Caso Normal</b>	<b>Caso Alternativo</b>
1. El propietario selecciona la opción "Añadir nueva solicitud."		
2. Proporciona los datos del visitante, (fecha, nombre, apellido y dni)		
3. Se envía la solicitud.		
4. El sistema verifica los datos ingresados.	4. 1. Campos completos.	4. 2. Campos incompletos. 4. 3. Muestra leyenda "campo incompleto" 4. 4. Vuelve al paso 2.
5. El sistema agrega la información del visitante a la tabla de "Mis Solicitudes" con la leyenda "pendiente".		

## 3. Egreso de visitantes (Guardia)

<b>Acción</b>	<b>Caso Normal</b>	<b>Caso Alternativo</b>
1. Un solicitante solicita egresar.		
2. El guardia selecciona el visitante que desea egresar en el registro.	2. 1. El guardia encuentra al visitante y lo selecciona.	2. 2. El guardia no encuentra al visitante. 2. 3. No se encuentra el dato.
3. El guardia añade una observación (opcional).		
4. Se registra el egreso.		
5. Fin del caso de uso.		

## Contrato de Operación

Funcionalidades:

1.

<b>Nombre</b>	Ingreso (IdSolicitud: int, Fecha: date)
<b>Responsabilidades</b>	Registrar los datos de un solicitante y enviar la petición de ingreso al propietario, en caso de ser aceptada, registrar también su ingreso.
<b>Referencias cruzadas</b>	ProgramarIngreso
<b>Excepciones</b>	Si los campos se encuentran incompletos o el formato es incorrecto, indicar que se cometió un error.
<b>Pre-condiciones</b>	El sistema conoce el IdSolicitud.
<b>Post-condiciones</b>	Se creó un nuevo ingreso. El sistema guarda el IdSolicitud, esperando el próximo egreso.

2.

<b>Nombre</b>	Egresos (IdIngreso: int, Fecha: date, Observaciones: varchar)
<b>Responsabilidades</b>	Capturar los datos de egreso del solicitante y registrarlo.
<b>Referencias cruzadas</b>	Ingreso (IdSolicitud: int, Fecha: date)
<b>Excepciones</b>	Si los campos se encuentran incompletos o el formato es incorrecto, indicar que se cometió un error. Si la subida a la base de datos falla, indicar que ocurrió un error.
<b>Pre-condiciones</b>	El sistema conoce el IdIngreso.
<b>Post-condiciones</b>	Se creó un nuevo egreso. El sistema deja de esperar el egreso del IdSolicitud.