


# IT TEXTHEOS

Nº	Nombre	Apellidos	Correo
1	Alejandro	Ortega Ordoñez	<a href="mailto:aortegaordonez@uma.es">aortegaordonez@uma.es</a>
2	Eduardo Noni	Stojadinovic	<a href="mailto:edunosto@uma.es">edunosto@uma.es</a>
3	Iván	Aguilar García	<a href="mailto:ivanaguilargarcia@uma.es">ivanaguilargarcia@uma.es</a>
4	Jonathan	Ruiz Oliva	<a href="mailto:jonathanruizoliva@uma.es">jonathanruizoliva@uma.es</a>
5	José Francisco	Villena Páez	<a href="mailto:jfvillenapaez@uma.es">jfvillenapaez@uma.es</a>
6	Miguel Ángel	Caballero Serrano	<a href="mailto:miguelangelcs22@uma.es">miguelangelcs22@uma.es</a>

Nombre del grupo: TexTheos
Titulación: Informática
Grupo de la titulación: 2ºA
Grupo reducido: G1
Número de grupo: 06
Repositorio GitHub: <a href="#">GitHub</a>
Espacio Trello: <a href="#">Trello</a>
Memoria del proyecto:  Biblioteca Online

1. Introducción.....	3
2. Roles.....	5
3. Riesgos.....	6
4. Planificación.....	9
5. Requisitos.....	12
Requisitos Funcionales:.....	12
Requisitos No Funcionales:.....	21
6. Casos de uso.....	27
7. Modelo del Dominio.....	40
8. Diagramas de secuencia.....	43
9. Arquitectura Software.....	56
Implementación y pruebas.....	56
10. Herramientas software.....	58

# 1. Introducción

Nuestro proyecto estará basado en la visualización de libros de forma online y/o presencial, de forma que el cliente pueda obtener el libro desde la comodidad de su hogar, ya sea descargándolo o desde la aplicación, o que obtenga sus libros de forma física yendo a unos de nuestros puntos de venta.

Para formar una gran comunidad de lectores, nuestros clientes podrán reunirse si así lo desean en ciertos puntos de encuentro para leer o discutir sobre sus libros favoritos. Dichas reuniones se concordarán en la aplicación realizando una reserva, aunque siempre habrá lectores que quieran compartir sus experiencias sobre su historial de lectura.

Nuestra aplicación tendrá la posibilidad de creación de usuarios para marcar sus favoritos, creación de listas, escribir reseñas y la posibilidad de encontrar nuevos usuarios para compartir libros o listas. Además podrán realizar una reserva para obtener los libros de forma física más tarde, asimismo estará la posibilidad de reservar plazas para hacer una reunión. El usuario podrá suscribirse a nuestro plan para obtener mejor calidad de los libros así como mayor gama de libros, tanto más antiguos como más actuales. Desde la aplicación, el usuario podrá ver una muestra del libro antes de comprarlo. Al igual que los usuarios puedan escribir reseñas, existirán algunos usuarios con prioridad en sus reseñas, ya que estos serán clientes que hayan consumido una gran cantidad de libros y que serán sometidos a un pequeño examen para que puedan obtener dichos privilegios, además contrataremos expertos para que hagan sus propias reseñas.

También obtendremos información de éxito y ventas sobre los libros para que el cliente tenga la máxima información sobre el producto. Nuestra aplicación recopilará datos no personales del usuario para brindarle un catálogo de libros lo más adecuado a sus necesidades y gustos.

Nuestros usuarios tendrán la posibilidad de contratar de menús u obtener descuentos por puntos a través de métodos como:

- Programa de lealtad: Los clientes pueden acumular puntos por cada libro que alquilen o compren, y así poder canjearlos en descuentos o productos.
- Descuentos por volumen: Ofrecer descuentos a los clientes que alquilen o compren una gran cantidad de libros a la vez.

- Descuentos por invitación: Los clientes reciben un descuento cuando invitan a un amigo que se registra y realiza una compra.
- Descuentos de temporada: Ofrecer descuentos en ciertos momentos del año, como durante las vacaciones o el inicio del año escolar.
- Descuentos para estudiantes: Los estudiantes reciben un descuento al presentar una identificación válida.
- Cupones: Los clientes pueden recibir cupones para descuentos en futuras compras.
- Descuentos para miembros: Los clientes que pagan una cuota de membresía anual reciben descuentos en todas sus compras.
- Funcionalidad de chat en vivo o soporte al cliente para resolver dudas o problemas de los usuarios.

## 2. Roles

- **Líderes del Proyecto (Miguel Angel C.S. y Eduardo N.S.):** Serán responsables de la planificación general del proyecto, asignación de tareas, seguimiento del progreso y comunicación con el equipo. También tomarán decisiones sobre el desarrollo del software asegurándose de que se cumplan los objetivos propuestos y que se implementen las ideas propuestas.

- **Analistas de Requisitos (Miguel Angel C.S. y José F.V.P.):** Se encargarán de recopilar, analizar y documentar los requisitos del cliente. Junto con el equipo de desarrollo garantizarán que el producto final cumpla con las expectativas del cliente.

- **Users Interface (Jonathan R.O. y Alejandro O.O.):** Diseñarán la interfaz de usuario del sistema, asegurándose de que sea intuitiva y fácil de usar. Colaborarán mayoritariamente con los analistas de requisitos y programadores para garantizar una buena experiencia de usuario.

- **Programadores (Jonathan R.O., Eduardo N.S. y Iván A.G.):** Desarrollarán el software siguiendo los requisitos y el diseño establecido. Serán responsables de escribir código limpio y bien documentado.

- **Testers (José F.V.P., Alejandro O.O., Iván A.G.):** Realizarán pruebas exhaustivas del software para identificar y reportar cualquier error o defecto. Junto con los programadores solucionarán los problemas detectados y garantizarán la mayor calidad posible del software final.

- **Project Manager (Miguel Angel C.S.):** Supervisará el progreso del proyecto, gestionará los recursos y resolverá cualquier problema que surja durante el desarrollo. Se asegurará de que el proyecto se entregue a tiempo.

- **Representante del Cliente (Eduardo N.S.):** Actuará de intermediario entre el equipo de desarrollo y el cliente, transmitiendo sus necesidades y expectativas al equipo y proporcionando retroalimentación sobre el progreso del proyecto.

### 3. Riesgos

#### 1. Riesgo Tecnológico

Descripción: Problemas que podrían surgir durante la integración de las diferentes tecnologías utilizadas en el desarrollo del proyecto, como incompatibilidades entre herramientas, problemas de configuración o dificultades en la implementación del sistema.

Probabilidad: Moderada, dado que el proyecto implica el uso de múltiples herramientas y tecnologías.

Efectos: Serios, ya que podría retrasar significativamente el desarrollo del proyecto y afectar la calidad del software final.

Estrategia para mitigarlo: Los programadores y testers deberán de realizar pruebas de forma regular durante todo el ciclo de desarrollo y resolver cualquier problema de manera eficiente.

#### 2. Cambios en los Requisitos

Descripción: Solicitud por parte del cliente de cambios frecuentes en los requisitos del proyecto. Pueden surgir debido a una comprensión incompleta de sus restricciones o a cambios en sus necesidades.

Probabilidad: Alta, ya que es común que los requisitos evolucionen a lo largo del tiempo a medida que el cliente adquiere una mejor comprensión de sus necesidades.

Efectos: Serios, dado que pueden afectar a la planificación del proyecto, aumentar los costes y el tiempo de desarrollo, y comprometer la calidad del software final.

Estrategia para mitigarlo: Establecer una comunicación clara y abierta con el cliente desde el principio del proyecto mediante una metodología ágil que

permita adaptarse rápidamente a los cambios y mantener un proceso de desarrollo que facilite la incorporación de nuevos requisitos.

### 3. Rendimiento del Sistema

Descripción: El sistema no puede soportar la carga esperada de usuarios o no cumple con los requisitos de rendimiento establecidos.

Probabilidad: Moderada, ya que el rendimiento puede verse afectado por factores como el tamaño de la base de datos, la complejidad de las consultas o la infraestructura de red.

Efectos: Catastróficos, debido a que podría resultar en una mala experiencia para los usuarios, pérdida de clientes y daño a la reputación de la biblioteca.

Estrategia para mitigarlo: El equipo de desarrollo realizará pruebas exhaustivas de rendimiento durante todo el ciclo de desarrollo identificando y resolviendo los cuellos de botella de rendimiento de manera constante, optimizando el código e identificar cualquier problema de rendimiento en tiempo real.

### 4. Riesgo de Seguridad de Datos

Descripción: La seguridad de los datos del sistema se ve comprometida debido a vulnerabilidades en el software o a acciones de terceros.

Probabilidad: Alta y las amenazas a la seguridad pueden provenir de una variedad de fuentes, incluidos hackers, malware y errores humanos.

Efectos: Catastróficos, ya que la pérdida de datos sensibles puede tener graves consecuencias legales y financieras para la biblioteca, así como dañar su reputación y la confianza de los usuarios.

Estrategia para mitigarlo: El equipo de desarrollo implementará prácticas de seguridad sólidas, como encriptación de datos, autenticación de usuarios y control de acceso. Además, se realizarán parches de seguridad de manera regular.

## 5. Riesgo de Recursos Insuficientes

Descripción: El proyecto no cuenta con los recursos adecuados, como personal, tiempo o presupuesto, para completarse con éxito.

Probabilidad: Moderada, debido a que la disponibilidad de recursos puede verse afectada por factores como cambios en la disponibilidad del personal o restricciones en el presupuesto.

Efectos: Serios, ya que la falta de recursos puede llevar a retrasos en el proyecto, aumento de los costes y afectar a la calidad del software final.

Estrategia para mitigarlo: Planificación cuidadosa de los recursos desde el principio del proyecto, identificando claramente los recursos necesarios y serán asignados de manera eficiente. Además, deberemos de mantener una comunicación abierta con todas las partes implicadas para gestionar expectativas y resolver cualquier problema de recursos.

## 6. Riesgo de Cambios en la Tecnología

Descripción: Cambios significativos en la tecnología o en las prácticas de desarrollo durante el transcurso del proyecto.

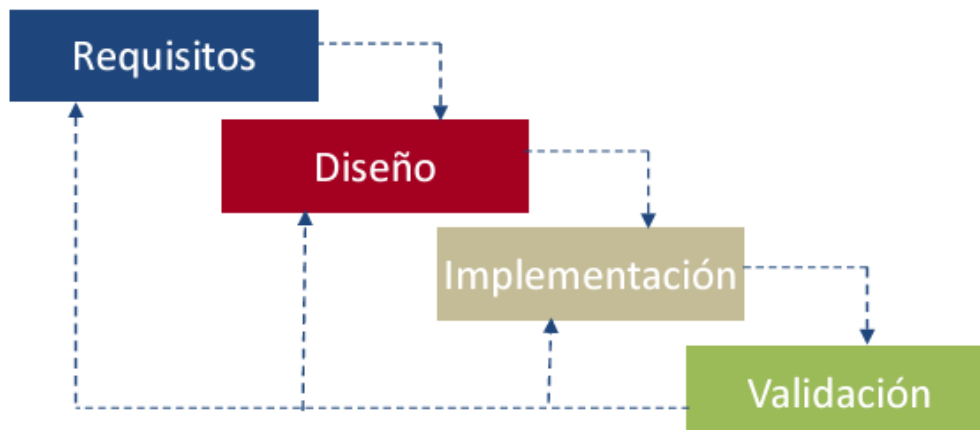
Probabilidad: Moderada, ya que las tecnologías evolucionan constantemente, y es posible que surjan nuevos enfoques que nos puedan afectar al desarrollo del proyecto y nos interese cambiar.

Efectos: Tolerables/Serio, dado que si se producen cambios en la tecnología surgirá la necesidad de adaptarse a las nuevas necesidades o entornos de programación y requerirá conocimiento previo que puede que no poseamos inicialmente. Sin embargo, estos efectos pueden gestionarse con una planificación adecuada.

Estrategia para mitigarlo: El equipo de desarrollo se mantendrá actualizado sobre las tendencias tecnológicas y las mejores prácticas actuales y estará preparado para adaptarse a cambios si es necesario.

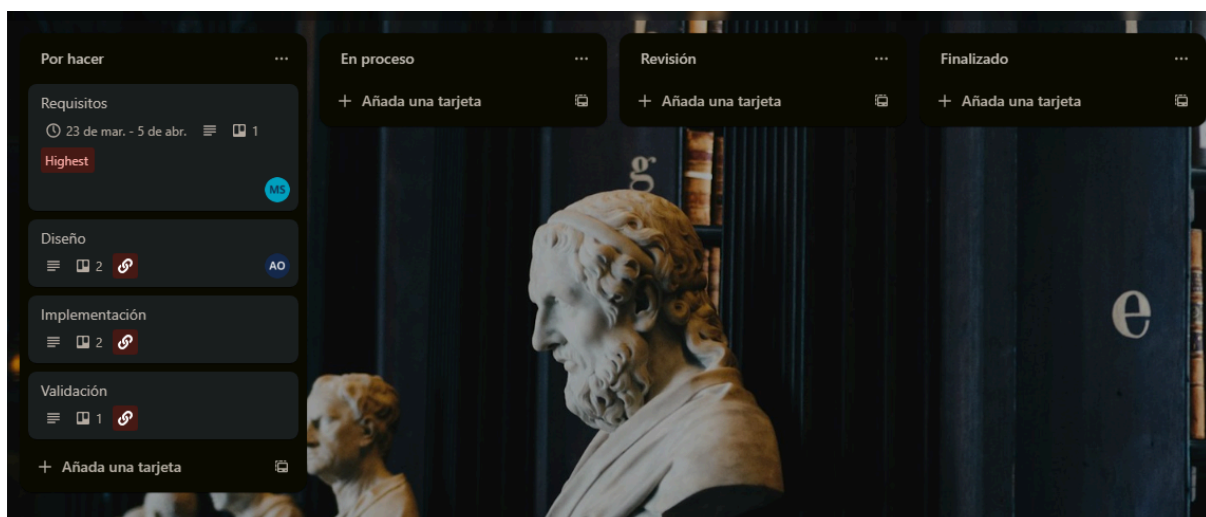


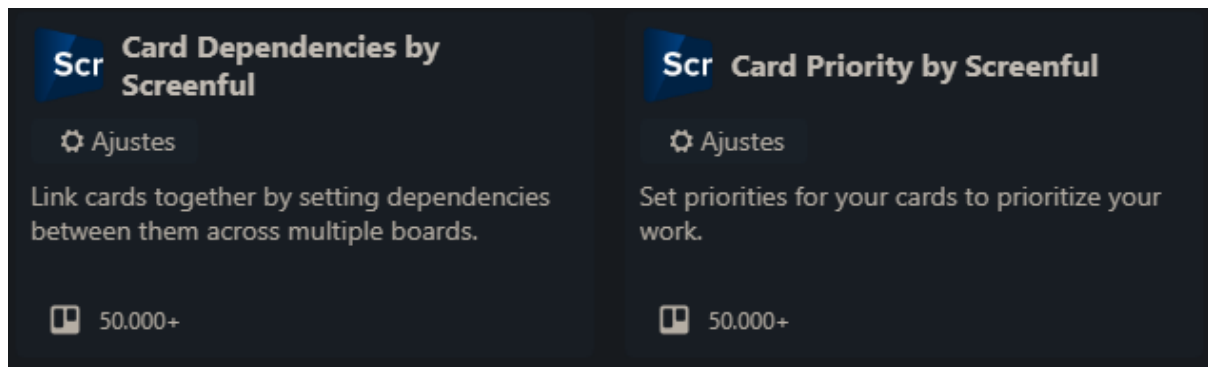
## 4. Planificación



El modelo en cascada, es la propuesta de un enfoque metodológico para hacer un software que consiste en ordenar de forma lineal las distintas etapas que debes de seguir al momento de desarrollar tu software. Este modelo ha sido seleccionado por nosotros debido a que es un modelo muy sencillo para aprender a hacer software. Otro motivo por el que lo hemos elegido es porque al analizar las desventajas que se nos pueden presentar por utilizar este modelo no nos son demasiado problemáticas dada la situación del proyecto, entre algunas desventajas del proyecto encontramos:

- El cliente a veces no conoce todos los requisitos al inicio.
- Los miembros del equipo deben esperar a que terminen tareas precedentes.
- El cliente debe tener paciencia.
- Poco flexible.
- Los proyectos reales raramente siguen el flujo secuencial del modelo.





- Requisitos:

Los requisitos son las especificaciones y características que el sistema debe cumplir para satisfacer las necesidades y requerimientos del cliente. Se dividen principalmente en requisitos funcionales, es decir, son los requisitos que describen las funciones y características específicas que el sistema debe realizar. Incluyen acciones que el sistema debe permitir a los usuarios realizar o procesos que debe automatizar.

En cambio, los no funcionales son los requisitos que describen las características de calidad y restricciones de rendimiento que el sistema debe cumplir. No se refieren a una función específica del sistema, sino a cómo debe funcionar el sistema en términos de rendimiento, seguridad, usabilidad, etc.

- Diseño:

#Estructura:

- Página de inicio/registro de sesión: El cliente registra un usuario con su contraseña para iniciar sesión y poder acceder más tarde. Debe tener un usuario único.
- Buscador: Los usuarios pueden buscar el libro por título, escritor, año y categoría.
- Categorías: Los libros se ordenan por categorías y etiquetas.
- Creación de etiquetas y favoritos: El usuario puede crear y guardar etiquetas, además de la etiqueta favoritos que viene por defecto
- Foro de discusiones en la pestaña del libro: Los usuarios pueden comentar los libros que han leído.

- Comentarios dentro de la comunidad: Los usuarios pueden interactuar entre sí dentro de nuestro foro.
- Opción de compra: Los usuarios pueden descargar los libros después de comprarlos.
- Alquilar: Los usuarios pueden alquilar el libro durante cierto tiempo para leerlo por un precio moderado.

#Cómo se desarrollan los elementos:

El programa inicia con el inicio de sesión/registro, es intermediario entre los usuarios y el programa para que la gente no pueda entrar en usuarios ajenos; también las demás opciones que te da el programa.

Dentro de la aplicación principal tenemos el buscador que se relaciona directamente con los libros, también tenemos las configuraciones de la cuenta, las categorías que se relaciona con los libros y la sección del foro de la comunidad que se relaciona con otros usuarios, además de la sección de compra y alquiler de libros. En la sección de libros existe relación con una sección de relacionados, guardar en alguna categoría, comentarios sobre el libro por parte de otros usuarios además de opción para comprar o alquilar.

- Implementación:

Utilizaremos para crear nuestro software dos elementos fundamentales los cuales son los siguientes:

- Lenguaje de programación: Java
- IDE: IntelliJ

La razón por la que vamos a usar estas dos es simple. La primera es el lenguaje de Java porque este es el lenguaje de programación que mejor manejan nuestros programadores y es un buen lenguaje para aplicaciones de escritorio. La segunda que usaremos es IntelliJ ya que este es un IDE que los programadores también manejan con agilidad, también permite la capacidad de implementar una interfaz gráfica.

- Validación:

En la fase de validación comprobaremos el funcionamiento adecuado del software, veremos si cumple todos los requisitos propuestos por el cliente y controlaremos la correcta realización de las tareas anteriores. Para ello ejecutaremos tests y dejaremos que el cliente pruebe las funcionalidades de la aplicación.

## 5. Requisitos

### Requisitos Funcionales:

#### - Creación de Usuarios:

Descripción: Los usuarios podrán crear cuentas en la plataforma para acceder a todas las funcionalidades.

Historia de Usuario: Como usuario nuevo, quiero poder registrarme en la plataforma ingresando mi información básica (nombre, correo electrónico, contraseña) para acceder a las funcionalidades de la plataforma.

Criterio de Aceptación: El sistema debe permitir al usuario completar el formulario de registro y enviar la información. Se debe verificar que la cuenta se cree correctamente y que el usuario pueda iniciar sesión con las credenciales proporcionadas.

#### - Reserva y alquiler de Libros:

Descripción: Los usuarios podrán reservar las ediciones físicas de los libros de las ubicaciones ofrecidas por la aplicación, también podrá alquilar tanto la edición física como la digital de un libro por un tiempo

Historia de Usuario: Como usuario de la plataforma, quiero poder reservar las ediciones en físico en alguna de las tiendas que tenga cerca y quiero poder alquilar estos o sus versiones digitales en la aplicación.

**Criterio de Aceptación:** El sistema debe poder ofrecer la posibilidad de reservar los libros en sus versiones físicas en alguna de las ubicaciones ofrecidas por la aplicación así como su alquiler tanto en ediciones físicas en estas o sus versiones digitales en la propia aplicación de la tienda.

- Reserva de plaza para reuniones:

**Descripción:** Los usuarios serán capaces de reservar plazas dadas por la aplicación en diferentes lugares para que varios usuarios puedan reunirse.

**Historia de Usuario:** quiero poder reservar algún lugar que me ofrezca esta aplicación para así poder reunirme con mis amigos o compañeros y hablar sobre los libros que poseemos.

**Criterio de Aceptación:** El sistema ofrecerá diferentes opciones de lugares para que los usuarios puedan reservar plazas en esa ubicación para reunirse junto a otros usuarios, la aplicación mostrará cuántas plazas dispone ese lugar así como si están reservadas ya por otro usuario y quien las ha reservado.

- Notificaciones :

**Descripción:** La aplicación enviará una serie de importantes avisos sobre diferentes temas relacionados con la misma

**Historia de Usuario:** como usuario de la aplicación, me gustaría estar al tanto de las ofertas, libros nuevos de ciertos autores entre otros tipos de información que yo considere interesante

**Criterio de Aceptación:** La aplicación deberá avisar al usuario a través de correos electrónicos (notificaciones en dispositivos móviles) de diferentes temas (ofertas, libros de autores, etc).

- Visualización de Muestra y detalles de Libros:

Descripción: Los usuarios tendrán acceso a la sinopsis del libro propuesta por

su autor así como a una pequeña parte del primer capítulo para que puedan decidir qué hacer con el libro.

Historia de Usuario: me gustaría ser capaz de ver una sinopsis del libro más allá de tan solo el título para saber que tipo de historia voy a leer, también me gustaría leer parte del primer capítulo para ver si me parece una buena idea tenerla

Criterio de Aceptación: La aplicación debe poder ofrecer una sinopsis del libro así como la capacidad de acceder a leer una primera parte del capítulo.

- Historial de actividades:

Descripción: Los usuarios pueden acceder a su historial de libros o pagos

Historia de Usuario: como usuario de la aplicación, quiero ser capaz de acceder a los libros que he visto o leído y también a mi historial de compras realizadas en la aplicación

Criterio de Aceptación: La aplicación debe ser capaz de mostrar el historial de libros leídos o que está leyendo el usuario así como las compras que ha realizado en la aplicación.

- Creación de Listas de Lectura (con etiquetas):

Descripción: Los usuarios podrán crear sus propias listas de libros e incluso añadirles sus propias etiquetas personalizadas para estos.

Historia de Usuario: quiero poder organizar de la manera que desee los libros que tengo en la aplicación para así tenerlo todo más organizado

Criterio de aceptación: La aplicación debe poseer la capacidad de permitir al usuario de crear sus propias listas personales privadas así como la de darle a estos la posibilidad de poner etiquetas personalizadas a los libros que posean.

- Sincronización de lectura:

Descripción: La aplicación guardará la última página del libro que este haya leído.

Historia de Usuario: como usuario de la aplicación me gustaría que se guardase la última página del libro que haya leído para así no tener que buscarlo más adelante cuando vuelva a leerlo

Criterio de Aceptación: La aplicación debe ser capaz de guardar la última página que el usuario haya leído en el libro indicado.

- Estadísticas sobre lecturas del usuario:

Descripción: Los usuarios podrán acceder a unas estadísticas sobre cuánto tiempo llevan leyendo usando la aplicación, que tipos de libros y cuales libros más ha leído el usuario.

Historia de Usuario: quiero poder ser capaz de acceder a las estadísticas para poder estar atento al tiempo de lectura que tengo con esta y que tipos de libros leo más

Criterio de Aceptación: La aplicación debe poseer la capacidad de mostrarle al usuario de manera clara y concisa las estadísticas de los libros que lee y el tiempo de lectura que pasa en la aplicación junto a qué tipos de libro lee.

- Escritura de Reseñas de Libros:

Descripción: Permite a los usuarios expresar sus comentarios y críticas sobre sus lecturas. Compartiendo experiencias, y recomendaciones.

Historia de Usuario: Como usuario, quiero poder escribir reseñas sobre los libros que he leído para compartir mis opiniones y recomendaciones con otros usuarios.

**Criterios de Aceptación:** Los usuarios pueden escribir reseñas y calificar los libros.

**- Suscripción a Planes Premium:**

**Descripción:** Contenido exclusivo mediante pago a cambio de ventajas en descuentos en alquileres y reserva de libros o cupones en próximas compras. Así como servicios adicionales que no están disponibles para el resto de usuarios.

**Historia de Usuario:** Como usuario interesado en acceder a contenido exclusivo, deseo poder suscribirme a planes premium para obtener descuentos en alquileres, reservas y cupones en próximas compras.

**Criterios de Aceptación:** La suscripción a planes premium está disponible y proporciona beneficios claros y exclusivos.

**- Búsqueda Avanzada de Libros:**

**Descripción:** Permite la realización de búsquedas específicas y detalladas de libros que se ajusten a los intereses y preferencias del usuario. Filtrando por título, autor, género, fecha de publicación y calificación.

**Historia de Usuario:** Como usuario, quiero realizar búsquedas avanzadas de libros para encontrar obras que se ajusten a mis intereses y preferencias.

**Criterios de Aceptación:** La búsqueda avanzada de libros permite a los usuarios filtrar resultados de manera efectiva.

**- Gestión de Perfil de Usuario:**

**Descripción:** Controlando y personalizando la información personal del usuario, preferencias de lectura, notificaciones y privacidad, etc.

**Historia de Usuario:** Como usuario, quiero gestionar mi perfil personalizado para controlar mi información, preferencias y privacidad en la plataforma.

**Criterios de Aceptación:** Los usuarios pueden gestionar su perfil, incluyendo preferencias y privacidad.



**-Gestión de Biblioteca Personal:**

Descripción: El usuario organiza su colección de libros de manera eficiente y personalizada. Agregando nuevos libros a su lista y organizarlos por categorías, etiquetas o favoritos.

Historia de Usuario: Como usuario, quiero organizar mi biblioteca personal de manera eficiente, agregando nuevos libros y categorizándolos según mis preferencias.

Criterios de Aceptación: La gestión de la biblioteca personal es intuitiva y permite una organización eficiente.

**- Integración de Redes Sociales:**

Descripción: Permite conectar la cuenta del usuario con su perfil en otras redes sociales. Permitiendo compartir reseñas, recomendaciones y otras actividades.

Historia de Usuario: Como usuario, deseo poder conectar mi cuenta de la plataforma con perfiles en otras redes sociales para compartir contenido y actividades relacionadas con la lectura.

Criterios de Aceptación: La integración con redes sociales facilita el intercambio de contenido.

**- Sistema de Recomendación de Libros:**

Descripción: Utiliza un algoritmo de inteligencia artificial para recomendar en base a búsquedas previas libros de género similar de su interés.

Historia de Usuario: Como usuario, quiero recibir recomendaciones personalizadas de libros basadas en mis intereses y búsquedas anteriores.

Criterios de Aceptación: El sistema de recomendación ofrece sugerencias relevantes y personalizadas.

- Inicio de sesión de usuarios:

Descripción: Permite el acceso a la cuenta personal de un usuario ya registrado mediante su usuario y contraseña.

Historia de Usuario: Como usuario registrado, deseo poder iniciar sesión en mi cuenta de manera segura y sencilla.

Criterios de Aceptación: El inicio de sesión de usuarios es seguro.

- Disponibilidad de pago:

Descripción: Procesamiento de transacciones de pago de manera segura. Permitiendo realizar compras de libros, suscripciones al plan premium etc.

Historia de Usuario: Como usuario, quiero poder realizar transacciones de pago de forma segura para comprar libros o suscribirme a planes premium.

Criterios de Aceptación: Las transacciones de pago se procesan de forma segura y confiable.

- Posibilidad de subir libros de un usuario:

Descripción: Ofrece la posibilidad al usuario de contribuir con su propio contenido a la plataforma. Permitiendo la carga de archivos digitales como PDF o ePub.

Historia de Usuario: Como usuario, deseo tener la capacidad de subir mis propios libros a la plataforma para compartir mi contenido con otros usuarios.

Criterios de Aceptación: Los usuarios pueden subir libros digitales a la plataforma con facilidad y sin errores.

- Lectura de libros online

Descripción: Los usuarios podrán leer libros en la aplicación formato digital.

Historia de usuario: Como usuario, puedo leer los libros a través de la aplicación.

Criterio de aceptación: El sistema permite mostrar al usuario el contenido de los libros.

- Se podrá descargar cualquier libro desde la aplicación.

Descripción: Los usuarios podrán leer en formato pdf después de descargarlos desde la aplicación.

Historia de usuario: Como usuario, puedo leer los libros en formato pdf fuera de la aplicación después de descargarlos.

Criterio de aceptación: El sistema permite al usuario descargar los libros en formato pdf.

- Prioridad entre usuarios

Descripción: En la sección de reseñas, los comentarios de los usuarios pueden tener distinta prioridad en el orden en el que se muestran.

Historia de usuario: Como usuario, quiero que las reseñas sean útiles y se muestren comentarios que aporten a la comunidad.

Criterio de aceptación: El sistema muestra reseñas de usuarios bien valorados.

- Lista de amigos

Descripción: En la lista de amigos aparecen los usuarios que haya añadido previamente el usuario.

Historia de usuario: Como usuario, quiero poder contactar con las personas que tengan los mismos gustos que yo.

Criterio de aceptación: El sistema aporta una lista donde añadir a otros usuarios.

- Compartir libros entre amigos

Descripción: Los usuarios pueden compartir los libros para así poder leer el mismo libro.

Historia de usuario: Como usuario, quiero compartir los libros que tengo con mis amigos.

Criterio de aceptación: El sistema permite compartir libros con otros usuarios.

- Foro público donde poder compartir puntos de vista con otros usuarios  
Descripción: La aplicación tendrá una sección de comunidad donde compartir libros y chatear con otros usuarios.

Historia de usuario: Como usuario, quiero socializar con la comunidad, para así conocer personas con los mismos gustos que yo.

Criterio de aceptación: El sistema permite socializar entre usuarios.

- Obtención de puntos por registro de amigos en sus compras  
Descripción: Si un usuario invita a un amigo a la aplicación y éste se une, el usuario recibe descuentos, si además, este amigo compra un libro por recomendación del usuario, el usuario gana descuentos igualmente.

Historia de usuario: Como usuario, quiero descuentos en la aplicación.

Criterio de aceptación: El sistema proporciona descuentos.

- Soporte al cliente

Descripción: La aplicación da la posibilidad de avisar de errores y reportes.

Historia de usuario: Como cliente, quiero poder reportar fallos, y usuarios cuando sea necesario.

Criterio de aceptación: El sistema proporciona atención al cliente.

- Descuentos en función de la época del año

Descripción: La aplicación proporciona descuentos en los días festivos.

Historia de usuario: Como cliente, quiero tener descuentos en los días festivos.

Criterio de aceptación: El sistema proporciona descuentos según las festividades.

- Los clientes pueden acumular puntos por cada libro que alquilen o compren, y así poder canjearlos en descuentos o productos.

Descripción: La aplicación proporciona puntos para usar en las compras.

Historia de usuario: Como cliente quiero descuentos por comprar

Criterio de aceptación: El sistema da puntos para usar en las compras

- Se podrá ofrecer descuentos a los clientes que alquilen o compren una gran cantidad de libros a la vez.

Descripción: Por la compra donde se gaste más de 50 € se hará descuento al usuario.

Historia de usuario: Como usuario quiero descuentos por comprar de una vez varios productos.

Criterio de aceptación: El sistema da descuentos por la compra de varios productos.

- Los clientes pueden recibir cupones para descuentos en futuras compras.

Descripción: Los cupones se pueden acumular para aumentar el descuento de estos.

Historia de usuario: Como usuario quiero descuentos semanales o diarios.

Criterio de aceptación: El sistema proporciona descuentos en forma de cupones por ingresar un día en específico y reclamarlo.

### Requisitos No Funcionales:

- Que los libros en la base de datos tengan id:

Descripción: Los libros se organizarán en la base de datos a través de una ID y eso permitirá conectar los libros con los géneros y usuarios.

Historia de usuario: Como cliente quiero saber que libros tengo y a qué géneros pertenecen los libros(Fantasía, tragedia,...).

Criterio de aceptación: La base de datos proporciona los ids a los libros

**- Seguridad de la Información del Usuario:**

Descripción: Adoptar las medidas necesarias para manejar y proteger los datos personales de los usuarios de nuestra aplicación.

Historia de usuario: Como usuario quiero que mis datos se guarden de forma segura.

Criterio de aceptación: El sistema guardará de forma segura la información personal proporcionada por los usuarios.

**- Distintos métodos de pago:**

Descripción: Proporcionar a los usuarios diferentes formas de efectuar sus pagos en nuestra aplicación.

Historia de usuario: Como usuario quiero tener diferentes métodos para efectuar mis compras.

Criterio de aceptación: El sistema proporciona diferentes formas de pagar

**- Rendimiento de la Plataforma:**

Descripción: Que la plataforma sea rápida, fácil de usar y con pocos fallos.

Historia de Usuario: Quiero que la aplicación no falle cada dos por tres.

Criterio de aceptación: El sistema proporciona el mínimo número de fallos.

**- Restauración de datos:**

Descripción: Hacer copias de seguridad para mantener a salvo los datos y libros de usuarios, y a su vez poder recuperarlos si se da el caso.

Historia de usuario: Quiero mantener mis libros y datos si cambio de dispositivo.

Criterio de aceptación: El sistema creará copias de seguridad para mantener los datos.

**- Disponibilidad y Tiempo de Actividad:**

Descripción: El usuario podrá ver que libros posee, cuanto tiempo pasa en la aplicación, y cuánto tiempo pasa con los libros.

Historia de usuario: Quiero saber cuanto tiempo paso en la aplicación o si he leído mucho estos días.

Criterio de aceptación: El sistema proporcionará gráficas o de maneras similares los datos respectivos con el tiempo que el usuario pasa en la aplicación,...

#### - Experiencia de Usuario:

Descripción: El usuario tendrá a su disposición diferentes herramientas para mejorar su experiencia en nuestra aplicación, como diferentes mejoras de accesibilidad y control parental

Historia de usuario: Como usuario quiero ser capaz de tener una experiencia agradable.

Criterio de aceptación: El sistema proporcionará herramientas de control parental y mejora de calidad de vida para personas con minusvalía.

#### - Cumplimiento de restricciones legales

Descripción: Garantizar que el sistema cumpla con todas las restricciones legales y regulaciones pertinentes relacionadas con la protección de datos y derechos de autor.

Historia de Usuario: Como usuario de la plataforma, quiero tener la seguridad de que mis datos personales y el contenido que consumo cumplen con las leyes y regulaciones.

Criterios de Aceptación: El sistema garantiza la privacidad y seguridad de los datos del usuario de acuerdo con las leyes de protección de datos pertinentes y se implementan medidas para garantizar el cumplimiento de los derechos de autor y otras regulaciones.

#### - Recuperar contraseña

Descripción: Permite a los usuarios restablecer su contraseña en caso de olvido o pérdida.

Historia de Usuario: Como usuario de la plataforma, si olvido mi contraseña, quiero poder recuperarla fácilmente para poder acceder a mi cuenta.

Criterios de Aceptación: Los usuarios pueden restablecer su contraseña utilizando un proceso seguro de verificación de identidad.

#### - Escalabilidad:

Descripción: Diseñar el sistema para que pueda crecer y adaptarse a medida que aumenta el número de usuarios y libros, sin afectar negativamente a la experiencia del usuario.

Historia de Usuario: Como administrador del sistema, quiero asegurarme de que la plataforma pueda manejar un aumento en el número de usuarios y contenido sin comprometer el rendimiento.

Criterios de Aceptación: El sistema puede manejar un aumento significativo en el número de usuarios concurrentes sin experimentar una degradación del rendimiento, usando arquitecturas distribuidas y el aprovisionamiento automático de recursos.

#### - Mantenibilidad:

Descripción: Facilitar la actualización y el mantenimiento del sistema para garantizar su funcionamiento continuo a lo largo del tiempo, mediante la modularización del código y la documentación adecuada.

Historia de usuario: Como usuario quiero que la aplicación prevalezca a lo largo de los años para poder disfrutar de mis libros.

Criterios de Aceptación: El sistema proporcionará actualizaciones y mejoras para mantenerse a lo largo de los años.

#### - Internacionalización y localización:

Descripción: La aplicación puede traducir el contenido del libro como la aplicación en sí.

Historia de usuario: Como usuario quiero poder entender lo que estoy leyendo.



Criterio de aceptación: El sistema traduce el contenido y se adapta a los idiomas y regiones.

- Tolerancia a fallos:

Descripción: La aplicación trata los fallos que puedan surgir.

Historia de usuario: Como usuario quiero que la aplicación no tenga fallos constantemente.

Criterio de aceptación: El sistema proporciona medidas contra fallos.

- Eficiencia energética:

Descripción: La aplicación trata de minimizar el consumo y aprovechar la eficiencia máxima posible.

Historia de usuario: Como cliente quiero que la aplicación no me consuma todos los recursos de mi dispositivo.

Criterio de aceptación: El sistema trata de aprovechar los recursos sin consumir excesivamente.

- Compatibilidad:

Descripción: La aplicación puede usarse en otros dispositivos

Historia de usuario: Como usuario quiero poder usar la aplicación en todos mis dispositivos.

Criterio de aceptación: El sistema da soporte de compatibilidad para poder usarse en distintos dispositivos.



## 6. Casos de uso

### 1. CU1. Crear libro

- Contexto de uso: Un administrador, cuando quiera, puede crear un libro en la plataforma.

- Precondiciones y activación: El administrador está conectado a la plataforma, la plataforma posee un opción especial para administradores para crear un libro y posee un libro para añadir.

- Garantías de éxito / Postcondición: Se crea un nuevo libro en el sistema.

- Escenario principal:

La plataforma tiene un botón visible solo para los administradores para agregar un libro a la plataforma, el sistema muestra un formulario para crear un nuevo libro, se rellena, se finaliza el formulario y el libro se publica en el sistema y muestra un mensaje de éxito.

- Escenarios alternativos: El libro no se ha creado correctamente y se muestra un mensaje al administrador.

### 2. CU2. Editar libro

- Contexto de uso: Un administrador quiere o necesita modificar información de un libro existente en el sistema.

- Precondiciones y activación: El administrador está conectado al sistema y accede a la opción de edición de libros.

- Garantías de éxito / Postcondición: La información del libro modificada se muestra correctamente en el sistema.

- Escenario principal:

El administrador selecciona el libro a modificar, el sistema muestra el formulario con la información existente del libro, el administrador modifica los campos necesarios, pulsa el botón de guardar cambios y el sistema actualiza la información del libro mostrando un mensaje de éxito.

- Escenarios alternativos: Los cambios no se guardan correctamente y se muestra un mensaje de error.

### 3. CU3. Eliminar libro

- Contexto de uso: Un administrador quiere o necesita eliminar un libro de la plataforma.

- Precondiciones y activación: El administrador está conectado al sistema y accede a la opción de eliminar libro

- Garantías de éxito / Postcondición: El libro seleccionado se elimina correctamente del sistema.

- Escenario principal:

El administrador selecciona el libro a eliminar, el sistema muestra la confirmación de eliminación, el administrador confirma la eliminación y el sistema borra de la base de datos el libro y muestra un mensaje de confirmación.

- Escenarios alternativos: El libro no se elimina correctamente y se muestra un mensaje de error.

#### 4. CU4. Buscar libro

- Contexto de uso: Un usuario quiere buscar un libro concreto en la plataforma.

- Precondiciones y activación: El usuario está conectado en el sistema y accede al buscador.

- Garantías de éxito / Postcondición: El usuario encuentra el libro deseado o una lista de resultados relevantes.

- Escenario principal:

El usuario ingresa en el buscador el título, autor o palabras claves del libro, el sistema muestra los resultados.

- Escenarios alternativos: La búsqueda no muestra resultados o la conexión a la base de datos falla.

#### 5. CU5. Ver detalles del libro

- Contexto de uso: Un usuario desea ver información detallada sobre un libro o leerlo en sí..

- Precondiciones y activación: El usuario está conectado al sistema y accede a un libro para leerlo o a “detalles” para ver la información.

- Garantías de éxito / Postcondición: El usuario visualiza correctamente los el contenido y detalles del libro y el libro existe en la plataforma.

- Escenario principal:

El usuario encuentra y selecciona el libro que desea leer y el sistema muestra una página con el texto del libro y la información detallada del libro (título, autor, sinopsis...).

- Escenarios alternativos: La página del libro o la de detalles no cargan correctamente o la información está incompleta.

#### 6. CU6. Añadir libro a lista de favoritos

- Contexto de uso: Cuando un usuario desea añadir un libro a su lista de favoritos previamente creada para acceder luego más fácilmente.
- Precondiciones y activación: El usuario debe tener acceso al sistema y lista de favoritos en este caso a la que desea añadir el libro
- Garantías de éxito / Postcondición: El libro se agrega correctamente a la lista de favoritos del usuario.
- Escenario principal:
  1. El usuario elige la opción de añadir a favoritos mientras visualiza por pantalla los detalles del libro.
  2. El sistema añade el libro a la lista de favoritos del usuario y muestra un mensaje de éxito por pantalla.
- Escenarios alternativos: El libro no se agrega correctamente a favoritos y se muestra por pantalla un mensaje de error.

#### 7. CU7. Crear lista/etiqueta de libros

- Contexto de uso: Cuando un usuario desea crear una lista o etiqueta personalizada de libros.
- Precondiciones y activación: El usuario está conectado en el sistema y accede a la opción de crear una nueva lista/etiqueta.
- Garantías de éxito / Postcondición: Se crea una nueva lista/etiqueta de libros en el sistema.
- Escenario principal:
  1. El usuario accede a la opción de crear una nueva lista.
  2. El sistema muestra por pantalla un formulario simple para que el usuario escriba el nombre y una breve descripción opcional de la lista/etiqueta.
  3. El usuario completa el formulario simple y confirma la creación de la lista.
  4. El sistema crea la nueva lista y muestra un mensaje de éxito.
- Escenarios alternativos: La lista no se crea correctamente y se muestra un mensaje de error.

#### 8. CU8. Agregar libro a lista

- Contexto de uso: Cuando un usuario quiere añadir un libro a la lista personalizada que ha creado.
- Precondiciones y activación: El usuario está conectado en el sistema y ha creado al menos una lista y visualiza los detalles del libro.

- Garantías de éxito / Postcondición: El libro se añade correctamente a la lista seleccionada por el usuario.

- Escenario principal:

1. El usuario elige la opción de “agregar a lista” mientras visualiza los detalles del libro.

2. El sistema muestra las listas creadas por el usuario.

3. El usuario elige la lista a la que quiere añadir el libro.

4. El sistema añade el libro a la lista seleccionada y muestra un mensaje de éxito.

- Escenarios alternativos: El libro no se añade correctamente a la lista y se muestra un mensaje de error.

## 9. CU9. Eliminar libro de lista

- Contexto de uso: Cuando un usuario quiere eliminar un libro de una lista personalizada.

- Precondiciones y activación: El usuario está conectado en el sistema, ha creado al menos una lista y visualiza los detalles del libro dentro de la lista.

- Garantías de éxito / Postcondición: El libro se elimina correctamente de la lista que el usuario ha elegido

- Escenario principal:

1. El usuario selecciona la opción de “eliminar libro” mientras visualiza los detalles del libro dentro de la lista.

2. El sistema muestra un mensaje para verificar la eliminación.

3. El usuario confirma la eliminación del libro de la lista .

4. El sistema elimina el libro de la lista y muestra un mensaje de éxito.

- Escenarios alternativos: El libro no se elimina correctamente de la lista y se muestra un mensaje de error.

## 10. CU10. Reservar libro en punto de venta

- Contexto de uso: Cuando un usuario quiere reservar un libro para recogerlo en un punto de venta físico.

- Precondiciones y activación: El usuario está conectado al sistema, visualiza los detalles del libro y accede a la opción de reserva en punto de venta.

- Garantías de éxito / Postcondición: El libro se reserva correctamente en el punto de venta seleccionado por el usuario.

- Escenario principal:

1. El usuario elige la opción de “reservar” en punto de venta mientras visualiza los detalles del libro.

2. El sistema muestra una lista de puntos de venta disponibles.

3. El usuario elige el punto de venta deseado.

4. El sistema confirma la reserva del libro en el punto de venta y muestra un mensaje de éxito.

- Escenarios alternativos: La reserva no se realiza correctamente y se muestra un mensaje de error.

#### 11. CU11. Reservar plaza para reunión

- Contexto de uso: Ocurre cuando un usuario quiere reservar un espacio para hacer una reunión con otros lectores.

- Precondiciones y activación: El interesado se va a la sección del sistema para hacer una reunión y hace una reserva.

- Garantías de éxito / Postcondición: El usuario consigue la reserva de su plaza.

- Escenario principal:

- 1.El usuario va a la sección de reserva de sitios y lo pulsa.

2. El sistema muestra una lista de sitios disponibles.

- 3.El usuario coje el sitio que más quiera.

4. El sistema hace la reserva y muestra un mensaje de éxito.

- Escenarios alternativos: No quedan reservas en este momento y la aplicación muestra un mensaje de no haberse podido reservar.

#### 12. CU12. Crear reseña de libro

- Contexto de uso: Ocurre cuando el usuario quiere poner una reseña de un libro sea para bien o mal.

- Precondiciones y activación: El usuario se encuentra en el sistema visualizando el libro al que le quiere poner la reseña.

- Garantías de éxito / Postcondición: Se guarda la reseña bien y se queda en el libro.

- Escenario principal:

1. El usuario va al libro que quiere poner la reseña y le da al botón de hacer reseña.

2. El sistema muestra un formulario para que el usuario ingrese su reseña.

3. El usuario escribe la reseña y la guarda.

4. El sistema guarda la reseña y ahora se puede ver en la pestaña de reseñas.

- Escenarios alternativos: La reseña no se guarda por algún motivo y se muestra un mensaje de error.

### 13. CU13. Ver reseñas de libro

- Contexto de uso: Cuando un usuario quiere ver las reseñas que tiene un libro en concreto.

- Precondiciones y activación: El usuario está en el sistema y va a la pestaña del libro y le da a ver reseñas.

- Garantías de éxito / Postcondición: El usuario ve las reseñas del libro.

- Escenario principal:

1. El usuario le da al botón de ver reseñas en la página principal del libro.

2. El sistema muestra las reseñas de todos los usuarios que han hecho reseñas al libro.

- Escenarios alternativos: No hay reseñas que ver o no tiene conexión a internet.

### 14. CU14. Suscribirse a plan premium

- Contexto de uso: Cuando un usuario quiere tener más ventajas en el sistema.

- Precondiciones y activación: El usuario se encuentra en el sistema y le da a conseguir plan premium.

- Garantías de éxito / Postcondición: El usuario ahora posee el plan premium.

- Escenario principal:

1. El usuario le da a conseguir el plan premium.

2. El sistema muestra los detalles que tiene cada plan y las formas de como pagarlo.

3. El usuario elige el plan que más quiere y la forma en la que desea pagar.

4. El sistema hace la suscripción y muestra un mensaje de todo correcto.

- Escenarios alternativos: La suscripción falla y muestra un mensaje de error.

### 15. CU15. Obtener muestra de libro

- Contexto de uso: Ocurre cuando el usuario quiere ver una muestra del libro para saber si comprarlo o no.

- Precondiciones y activación: El usuario va al libro que quiere en el sistema y le da a la pestaña de muestra.



- Garantías de éxito / Postcondición: El usuario puede leer un fragmento del libro.

- Escenario principal:

1. El usuario le da a ver muestra en un libro concreto.
2. El sistema muestra unas páginas del libro.

- Escenarios alternativos: La muestra se visualiza bien o no se encuentra para ese libro.

#### 16. CU16. Compartir lista de libros

- Contexto de uso: Cuando un usuario quiere compartir una lista personalizada de libros con otros usuarios.

- Precondiciones y activación: El usuario está usando el sistema y pulsa la opción de compartir lista.

- Garantías de éxito / Postcondición: La lista se comparte correctamente con los otros usuarios seleccionados.

- Escenario principal:

1. El usuario usa la opción de compartir lista.
2. El sistema muestra algunas opciones para compartir la lista.
3. El usuario elige la forma de compartir la lista y selecciona los usuarios a los que compartir la lista.
4. El sistema envía la lista a los usuarios seleccionados.

- Escenario alternativo: Ocurre un fallo en el proceso y se muestra un mensaje de error.

#### 17. CU17. Acumular puntos por compra

- Contexto de uso: Cuando un usuario compra un libro, acumula puntos por ello.

- Precondiciones y activación: El usuario está usando el sistema y compra un libro.

- Garantías de éxito / Postcondición: El usuario recibe los puntos, que se suman a su total.

- Escenario principal:

1. El usuario compra un libro.
2. El sistema suma los puntos de la compra al total del usuario.

- Escenario alternativo: Ocurre un error en la suma de puntos y aparece un mensaje de error.

#### 18. CU18. Canjear puntos por descuento

- Contexto de uso: Cuando un usuario canjea sus puntos por posibles descuentos.

- Precondiciones y activación: El usuario está usando el sistema y pulsa en canjear puntos.

- Garantías de éxito / Postcondición: Se aplica el descuento correspondiente por el canjeo de puntos.

- Escenario principal:

1. El usuario entra en la opción de canjear sus puntos.

2. El sistema muestra los distintos descuentos y el usuario elige alguno.

3. El usuario efectúa la compra con el descuento aplicado.

- Escenario alternativo: Ocurre un error en la aplicación del descuento y salta un mensaje de error.

#### 19. CU19. Obtener descuento por invitación

- Contexto de uso: Cuando a un usuario se le otorga un descuento por invitar a un amigo y que este realice una compra.

- Precondiciones y activación: El usuario usa la opción de invitar amigos, y el amigo efectúa una compra.

- Garantías de éxito / Postcondición: El usuario recibe dicho descuento por invitar al amigo.

- Escenario principal:

1. El usuario invita a su amigo a usar el sistema mediante la herramienta de invitar amigos.

2. El amigo se registra en el sistema y efectúa una compra.

3. El sistema otorga al usuario inicial un descuento.

- Escenario alternativo: Ocurre un fallo y no se entrega el descuento correctamente.

#### 20. CU20. Obtener descuento de temporada

- Contexto de uso: Cuando se otorga un descuento al usuario por comprar en una época determinada.

- Precondiciones y activación: Se compra en una época predefinida (Días festivos, inicio de curso, etc.).

- Garantías de éxito / Postcondición: Se recibe correctamente el descuento en la fecha definida.

- Escenario principal:

1. El usuario compra durante la época de descuento.
2. El sistema pone un descuento en los productos seleccionados.
- Escenario alternativo: Ocurre un error en la aplicación del descuento por épocas.

#### 21. CU21. Obtener descuento para estudiantes

- Contexto de uso: Un estudiante recibe un descuento al presentar una identificación válida.
- Precondiciones y activación: El usuario realiza una compra y presenta una identificación como estudiante que sea verdadera.
- Garantías de éxito / Postcondición: El usuario recibe correctamente el descuento tras verificar que sea estudiante.
- Escenario principal:
  1. El usuario hace una compra y da unos datos de su cuenta para que se verifique que es estudiante.
  2. El sistema verifica la identificación y aplica automáticamente el descuento.
- Escenarios alternativos: La identificación del estudiante no es válida y el descuento no se aplica y se da un mensaje de error.

#### 22. CU22. Obtener descuento mediante cupón

- Contexto de uso: El usuario recibe un cupón para obtener descuentos en sus futuras compras.
- Precondiciones y activación: El usuario tiene un cupón válido para descuentos.
- Garantías de éxito / Postcondición: El usuario aplica correctamente el descuento obtenido mediante el cupón.
- Escenario principal:
  1. El usuario accede a la opción de aplicar cupón durante el proceso de compra.
  2. El usuario escribe su cupón.
  3. El sistema valida el cupón y aplica automáticamente el descuento.
- Escenarios alternativos: El cupón no es válido o ya ha sido utilizado y el descuento no se aplica y da un error.

#### 23. CU23. Obtener descuento para miembros

- Contexto de uso: Cuando un usuario que es miembro recibe descuentos en todas las compras que hace.

- Precondiciones y activación: El usuario es miembro pagado verdaderamente.

- Garantías de éxito / Postcondición: El usuario recibe el correspondiente descuento a aplicar en el producto que elija.

- Escenario principal:

1. El usuario realiza una compra estando logueado como miembro.

2. El sistema verifica la veracidad de que exista un miembro logueado como indica y aplica el descuento al producto.

- Escenarios alternativos: No es miembro o no lo verifica adecuadamente y se produce un error.

#### 24. CU24. Iniciar chat en vivo con soporte al cliente

- Contexto de uso: Cuando un usuario necesita ayuda o tiene preguntas e inicia un chat al soporte al cliente.

- Precondiciones y activación: El usuario está conectado en el sistema y accede a la opción de chat en vivo.

- Garantías de éxito / Postcondición: El usuario inicia correctamente el chat y el soporte al cliente le responde a su consulta.

- Escenario principal:

1. El usuario inicia un chat en vivo con soporte.

2. El sistema conecta al usuario con una persona de soporte disponible en ese momento.

3. El usuario hace su pregunta o solicitud de ayuda y el agente de soporte responde en tiempo real.

- Escenarios alternativos: No hay agentes de soporte disponibles en ese momento y el usuario recibe un mensaje de error.

#### 25. CU25. Realizar prueba de rendimiento del sistema

- Contexto de uso: Cuando los programadores y testers necesitan evaluar el rendimiento del sistema.

- Precondiciones y activación: Tenemos acceso al sistema y a herramientas para realizar pruebas de rendimiento.

- Garantías de éxito / Postcondición: Se obtienen datos precisos sobre el rendimiento del sistema y posibles cuellos de botella.

- Escenario principal:

1. Se hace una prueba de rendimiento del sistema.
2. Analizamos los resultados de la prueba e identificamos áreas de mejora o posibles problemas de rendimiento.
  - Escenarios alternativos: La prueba de rendimiento no se ejecuta correctamente.

#### 26. CU26. Realizar parche de seguridad

- Contexto de uso: Cuando el equipo de desarrollo necesita aplicar un parche de seguridad para proteger el sistema contra vulnerabilidades.
- Precondiciones y activación: El grupo tiene acceso a los parches de seguridad para cuando sea necesario activarlos en caso de vulnerabilidad.
- Garantías de éxito / Postcondición: Se aplica correctamente el parche y se soluciona la vulnerabilidad o la comprobación.
- Escenario principal:

El grupo encuentra una vulnerabilidad en el sistema y por tanto deciden aplicar el parche de seguridad en el sistema para solucionarlo, en el caso previsto, el parche soluciona la vulnerabilidad.
- Escenario alternativo: En un caso alternativo, el parche falla o no logra arreglar la vulnerabilidad.

#### 27. CU27. Actualizar tecnologías utilizadas en el proyecto

- Contexto de uso: Cuando el grupo decide actualizar las tecnologías utilizadas en el proyecto para mejorar el rendimiento o la seguridad.
- Precondiciones y activación: El grupo contempla las nuevas tecnologías actuales y decide realizar una actualización.
- Garantías de éxito / Postcondición: Se implementan correctamente las nuevas tecnologías y se mejora el sistema.
- Escenario principal:

El grupo decide actualizar el sistema con las nuevas actualizaciones y deciden migrar a estas nuevas tras una exhaustiva investigación, tras haber conseguido trasladar y adaptar el sistema, realizan unas pruebas para probar que la migración no haya traído problemas y se haya actualizado con éxito el sistema.
- Escenario alternativo: La migración o las pruebas fallarían demostrando que no se ha actualizado correctamente el sistema.

## 28. CU28. Obtener información de éxito y ventas sobre los libros

- Contexto de uso: Cuando el grupo necesita obtener datos sobre la popularidad y ventas de los libros en la plataforma.

- Precondiciones y activación: El grupo posee una herramientas de análisis de datos y al sistema.

- Garantías de éxito / Postcondición: Se obtiene información precisa sobre el éxito y las ventas de los libros.

- Escenario principal:

El grupo tras notar la necesidad de comprobar el éxito y las ventas de libros deciden realizar un análisis de datos del sistema con la herramienta que poseen, indican unos filtros búsqueda y activan la herramienta y si esta consigue dar los datos de manera correcta e interpretables se consideraría un éxito.

- Escenario alternativo: el caso de que la herramienta no recopila bien los datos indicados o la forma en las que los proporcione no sean interpretables para el grupo.

## 29. CU29. Recopilar datos de búsquedas del usuario

- Contexto de uso: Cuando el sistema recopila datos no personales del usuario para otorgar recomendaciones personalizadas.

- Precondiciones y activación: El usuario utiliza el sistema y acepta los términos de privacidad.

- Garantías de éxito / Postcondición: Se recopilan y utilizan correctamente los datos no personales del usuario para mejorar la experiencia del usuario.

- Escenario principal:

El usuario navega a través de la aplicación, buscando y leyendo diferentes libros, el sistema se encarga de recopilar los libros que le interesan al usuario así como sus opiniones dejadas o los libros marcados como interesantes, una vez recopilados estos datos, la aplicación le mostraría al usuario diferentes recomendaciones, sobre todo libros, donde se consideraría un éxito la mejora de la experiencia del usuario.

- Escenario alternativo: El sistema no recopila bien los datos o los utiliza de manera inadecuada pudiendo llegar a empeorar la experiencia de usuario.

## 30. CU30. Gestionar recursos para el proyecto

- Contexto de uso: Cuando el grupo necesita asignar y gestionar recursos para el proyecto, como personal, tiempo y disponibilidad.

- Precondiciones y activación: El grupo tiene acceso a herramientas de gestión de proyectos y recursos.

- Garantías de éxito / Postcondición: Se asignan correctamente los recursos necesarios y se gestiona eficientemente su uso.

- Escenario principal:

El grupo evalúa los recursos que necesita el proyecto y los asigna en función de las capacidades de los miembros del equipo. Se considera un éxito en el caso de que los recursos hayan sido asignados correctamente y el desarrollo haya mejorado con creces.

-Escenario alternativo: el que no se asignan los recursos correctamente y afecta negativamente a la gestión de recursos.

### 31. CU31: Iniciar sesión

- Contexto de uso: Cuando el usuario quiere registrarse/iniciar sesión en la plataforma

- Precondiciones y activación: El usuario tiene acceso al sistema o tiene una cuenta creada en la plataforma.

- Garantías de éxito / Postcondición: El inicio de sesión es correcto.

- Escenario principal:

El usuario escribe el nombre del usuario y la contraseña, comprueba que lo escrito anteriormente sea correcto, y muestra un mensaje de éxito, luego el sistema le da acceso a la cuenta seleccionada por el usuario.

- Escenarios alternativos: Si la cuenta no se encuentra en el sistema , o se ha ingresado incorrectamente, se muestra un mensaje de error.

### 32. CU32: Usuario vende libro

- Contexto de uso: Cuando el usuario sube sus propios libros al sistema y los vende.

- Precondiciones y activación: El usuario accede a la opción de publicar libro.

- Garantías de éxito / Postcondición: El libro se sube correctamente al sistema.

- Escenario principal:

1. El usuario entra en la opción de subir el libro.

2. El usuario sube el libro, rellenando toda la información relevante del mismo, como su información o precio.

3. El sistema valida el libro subido y lo añade a la base de datos.

- Escenarios alternativos: Ocurre algún problema en la subida del libro al sistema y se notifica con un mensaje de error.

### 33. CU33: Usuario alquila, reserva o compra libro

- Contexto de uso: El usuario quiere alquilar, reservar o comprar un libro existente en la plataforma o en físico.

- Precondiciones y activación: El usuario accede a la opción de compra o alquiler dentro de la descripción del libro y le da a la opción de comprar o alquilar. El usuario va a un punto de venta y compra el libro disponible físicamente.

- Garantías de éxito / Postcondición: El usuario posee el libro físicamente o se muestra en su lista de comprados en su perfil, siendo capaz de leer el libro.

- Escenario principal:

1. El usuario busca el libro deseado y existente en la plataforma
2. El usuario pulsa la opción de comprar/alquilar/reservar el libro
3. El usuario realiza el método de pago
4. El usuario elige la opción de recogerlo en físico o obtenerlo en su biblioteca personal en la plataforma.
5. El usuario obtiene un mensaje de confirmación de que el libro ha sido obtenido exitosamente.

- Escenarios alternativos: Método de pago rechazado o no existente, libro no disponible. El libro no aparece en la lista de comprados de su perfil.

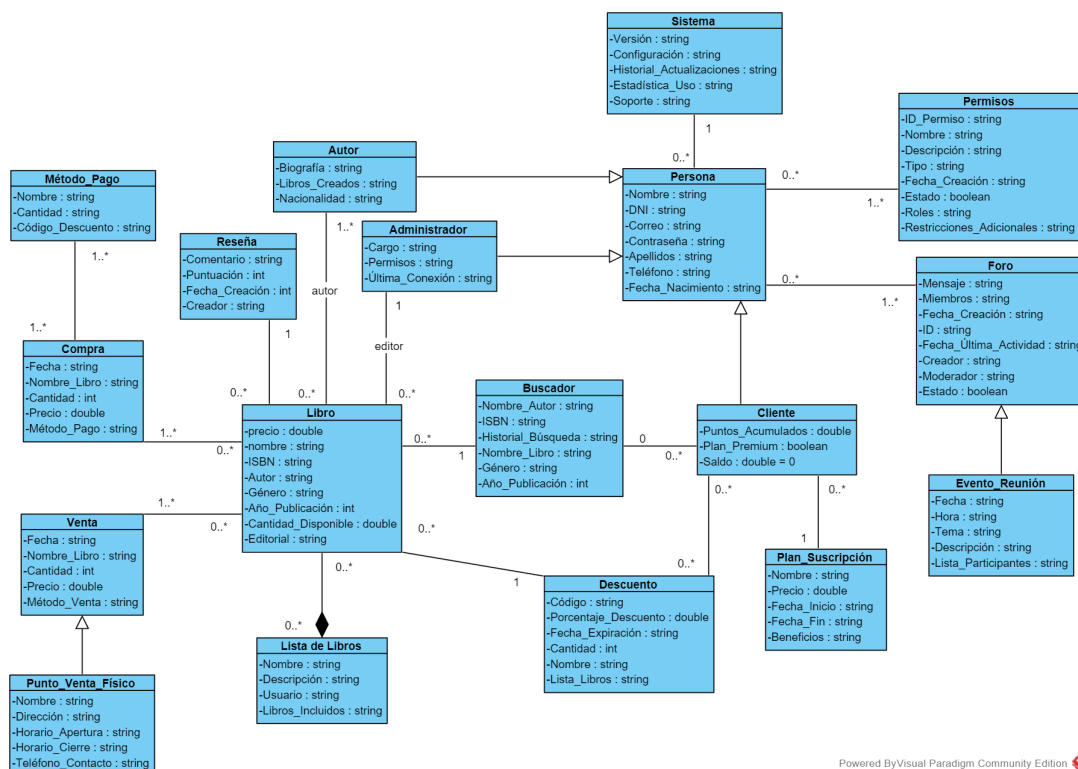
## 7. Modelo del Dominio

- **Libro:** Un libro del sistema.
  - Asociación con Autor: Uno o muchos autores pueden crear uno o muchos libros.
  - Asociación con Administrador: Un administrador puede editar uno o muchos libros.
- **Persona:** Una persona que usa la aplicación.
- **Cliente:** Un cliente de la aplicación.
  - Generalización con Persona: un cliente es una persona.
- **Autor:** Un autor de un libro.



- Generalización con Persona: un autor es una persona.
- **Administrador:** Un administrador del sistema.
  - Generalización con Persona: un administrador es una persona.
- **Permisos:** Los permisos que tendrán las personas en la aplicación.
  - Asociación con Persona: Una persona puede tener uno o muchos permisos.
- **Reseña:** Una reseña de un libro.
  - Asociación con Libro: Un libro puede tener una o muchas reseñas.
- **Lista de Libros:** Una lista personalizada de libros hecha por los usuarios.
  - Composición con Libro: Muchas listas pueden estar compuesta de muchos libros.
- **Venta:** Una venta física de un libro.
  - Asociación con Libro: Muchos libros tienen muchas ventas.
- **Compra:** Una compra online de un libro.
  - Asociación con Libro: Muchos libros tienen muchas compras.
- **Descuento:** Un descuento aplicado a la compra de un libro.
  - Asociación con Cliente: Uno o muchos clientes pueden tener uno o muchos descuentos.
  - Asociación con Libro: Uno o muchos libros pueden tener un descuento.
- **Plan\_Suscripción:** Una suscripción adquirible por los usuarios.
  - Asociación con Cliente: Uno o muchos clientes pueden tener una suscripción.
- **Evento\_Reunión:** Una reunión organizada por los usuarios.
  - Generalización con Foro: El evento de reunión se crea en el foro.
- **Foro:** Foro de usuarios online.
  - Asociación con Persona: Muchos foros tienen muchas personas.

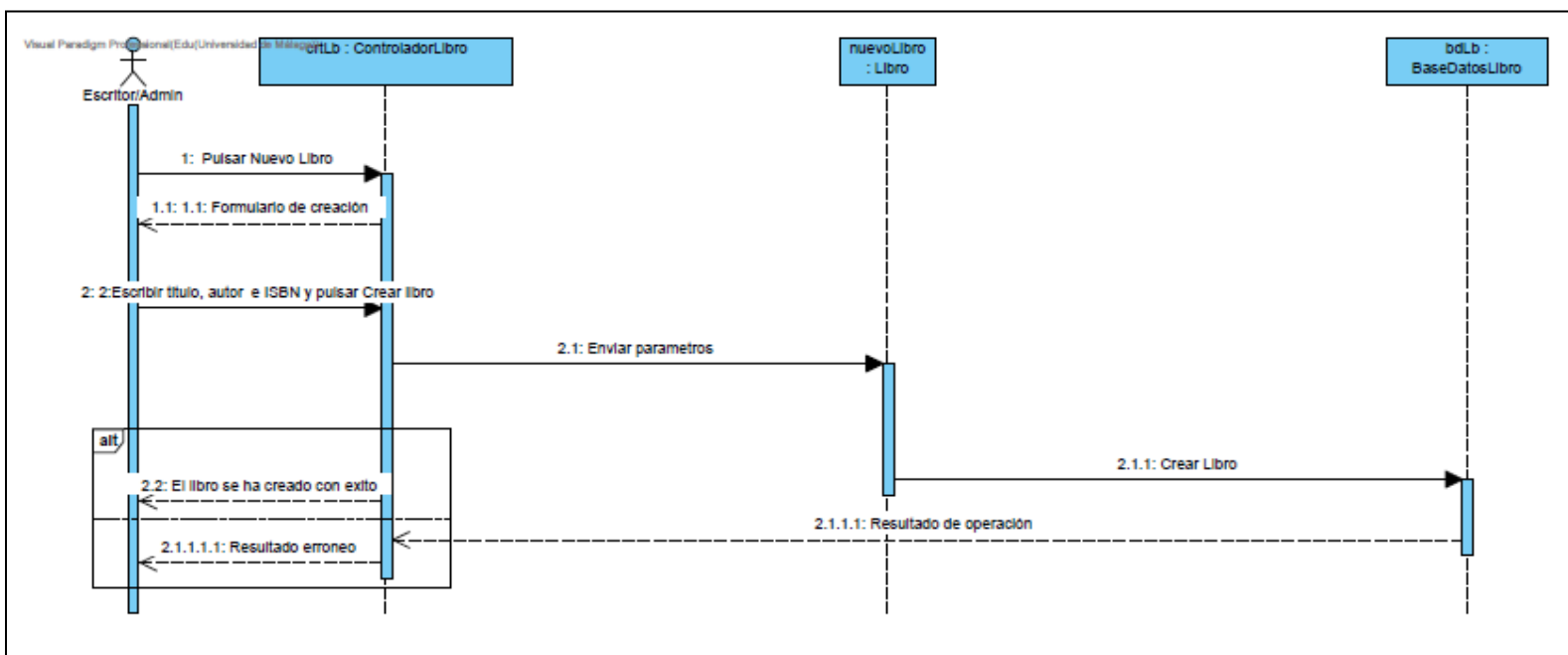
- **Punto\_Venta\_Físico**: Lugar físico de recogida de libros.
  - Generalización con Venta: El punto de venta físico existe por las ventas.
- **Método\_Pago**: Forma de pagar al efectuar una compra.
  - Asociación con Compra: Muchas compras tienen muchos métodos de pago.
- **Buscador**: El buscador de libros de la aplicación.
  - Asociación con Cliente: Muchos clientes pueden usar un buscador.
  - Asociación con Libro: Un buscador puede encontrar muchos libros.
- **Sistema**: El sistema del servicio.
  - Asociación con Persona: Muchas personas pueden usar el sistema.



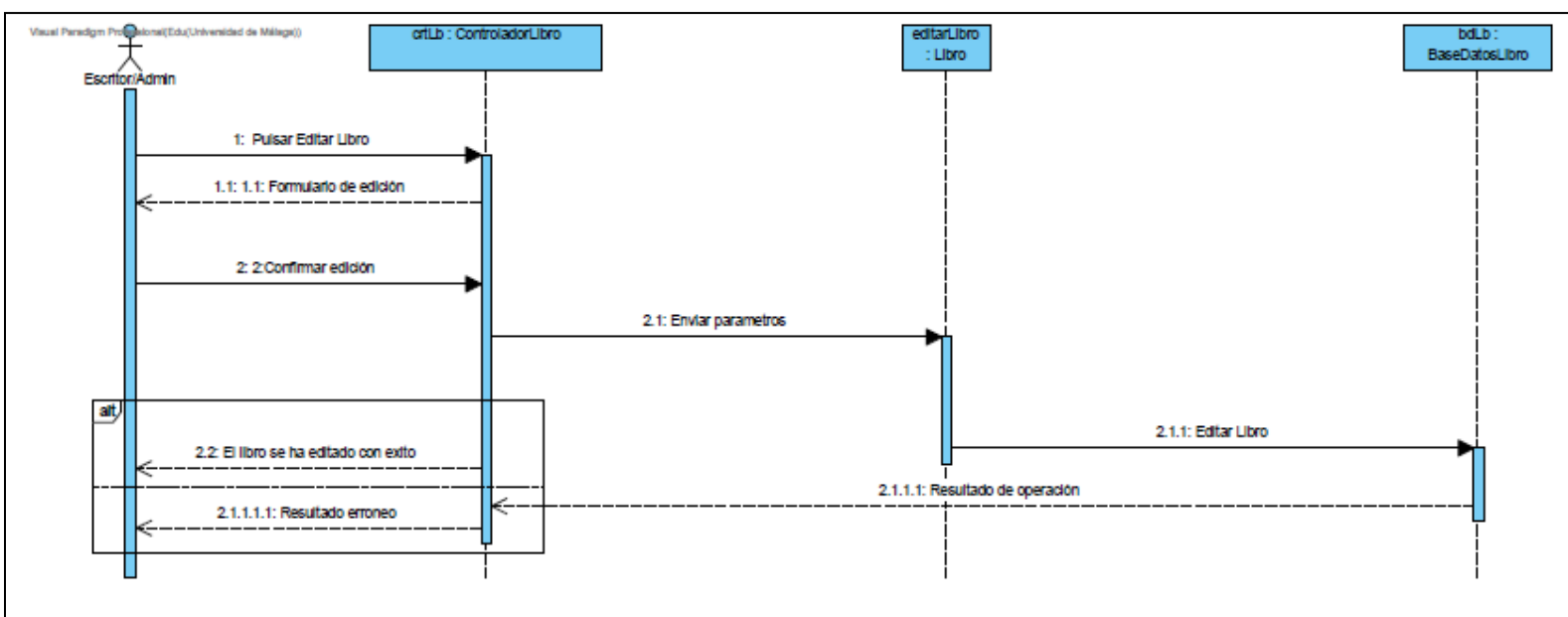
Powered By Visual Paradigm Community Edition

## 8. Diagramas de secuencia

- **Crear Libro:** El administrador pulsa en crear Libro y se le devuelve un formulario con los datos a rellenar.  
Se rellenan los datos y se envían a la función nuevoLibro.  
Entonces se comunica con la base de datos y ésta devuelve el resultado de la operación al controlador.  
Se devuelve al usuario si la operación se ha completado con éxito o no.



- **Editar Libro:** El administrador pulsa en Editar Libro y se le devuelve un formulario con los datos a editar.  
Se modifican los datos necesarios y se envían a la función editarLibro.  
Entonces se comunica con la base de datos y ésta devuelve el resultado de la operación al controlador.  
Se devuelve al usuario si la operación se ha completado con éxito o no.

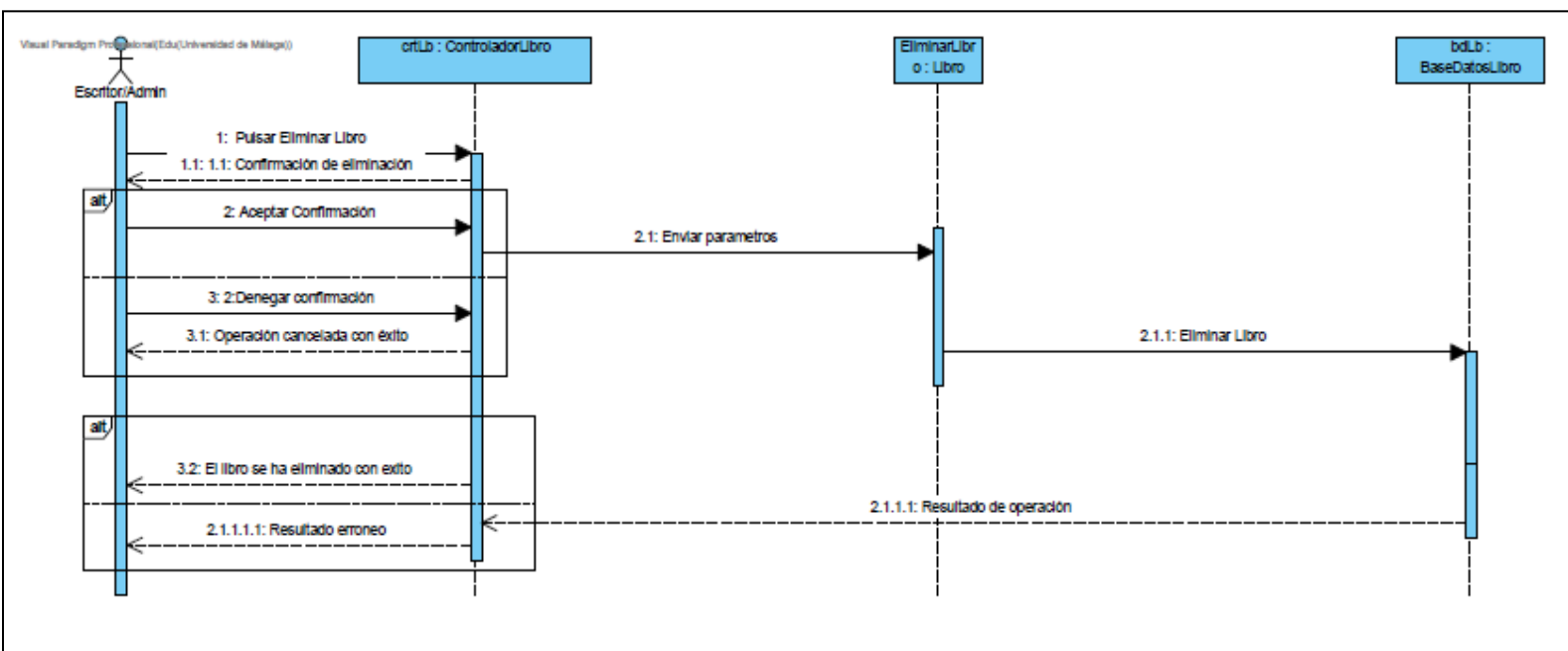


- **Eliminar Libro:** El administrador pulsa en Eliminar Libro y el controlador responde con una confirmación de eliminación.

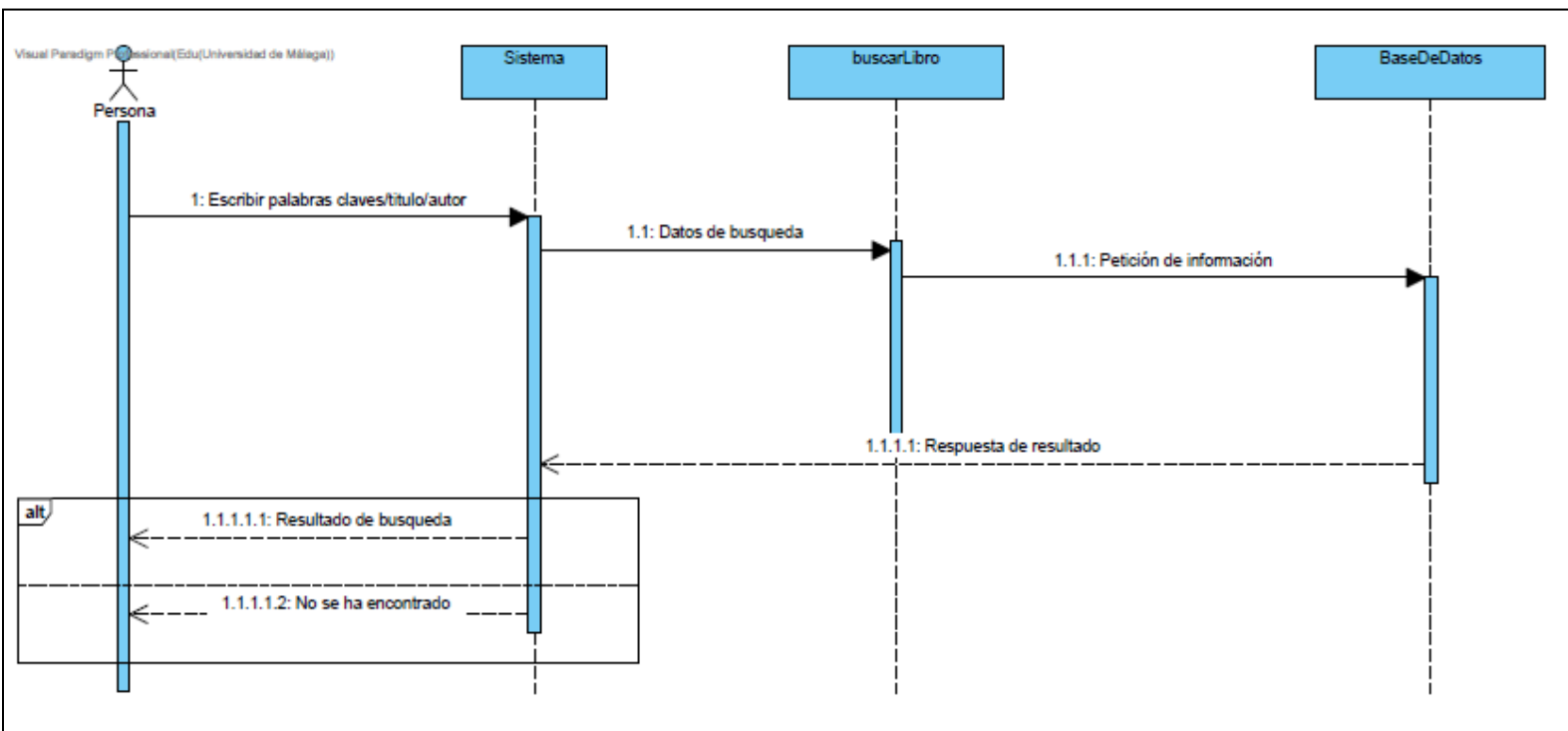
El administrador puede aceptar la confirmación para que el controlador envíe una petición a la función eliminarLibro o puede denegar la confirmación y finalizar la operación.

Entonces si se aceptó la confirmación la función se comunica con la base de datos y ésta devuelve el resultado de la operación al controlador.

Se devuelve al usuario si la operación se ha completado con éxito o no.



- **Buscar Libro:** Una persona escribe título, autor o palabras clave del libro a buscar y el sistema envía los datos a la función buscarLibro. La función pide esta información a la base de datos, que devuelve el resultado de la búsqueda al sistema. El sistema muestra una lista de libros al usuario si la operación ha sido exitosa o un mensaje: "No se han encontrado resultados" en otro caso.

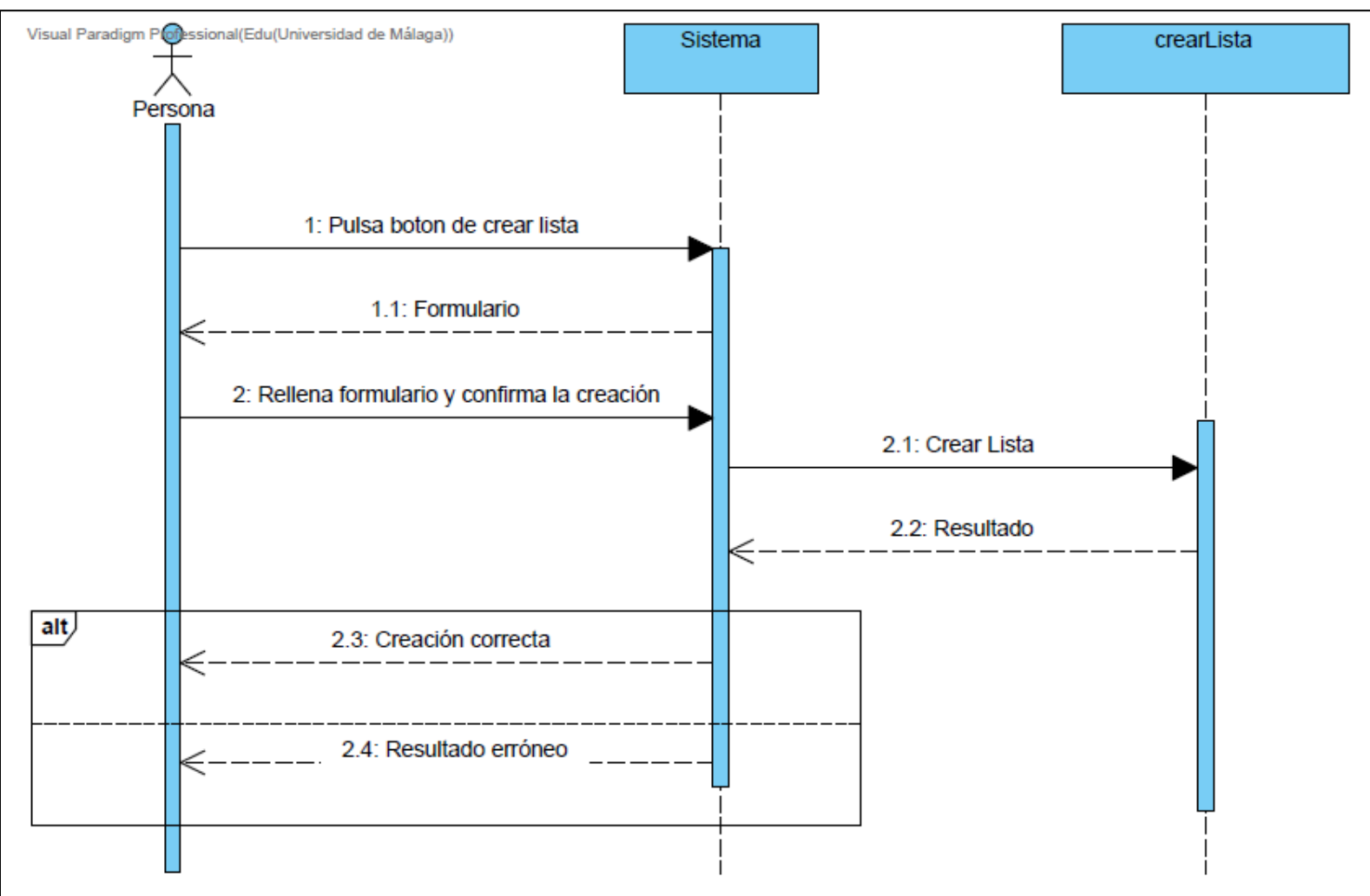


- **Crear Lista:** Una persona pulsa el botón Crear Lista y el sistema devuelve un formulario con datos a rellenar.

La persona rellena el formulario y confirma la creación de la lista

El sistema llama a la función crearLista y la función devuelve el resultado de la operación.

El sistema devuelve a la persona si la operación ha sido exitosa o no.

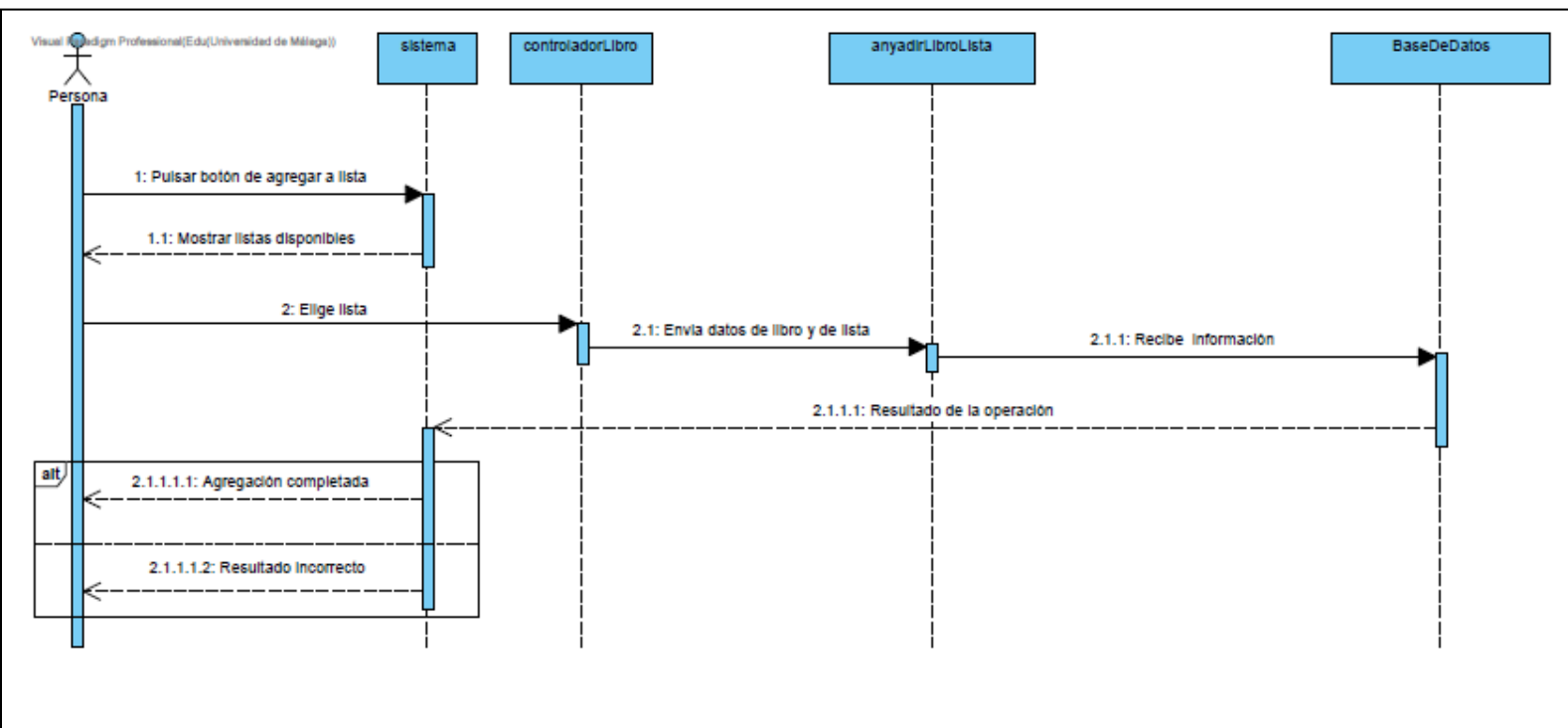


- **Añadir Libro a la Lista:** Una persona pulsa el botón Agregar a Lista y el sistema devuelve las listas disponibles.

La persona elige la lista que desee y el controlador envía los datos a la función `anyadirLibroLista`.

La función manda la información a la base de datos que devuelve el resultado al sistema.

El sistema devuelve a la persona si la operación fue correcta o no.



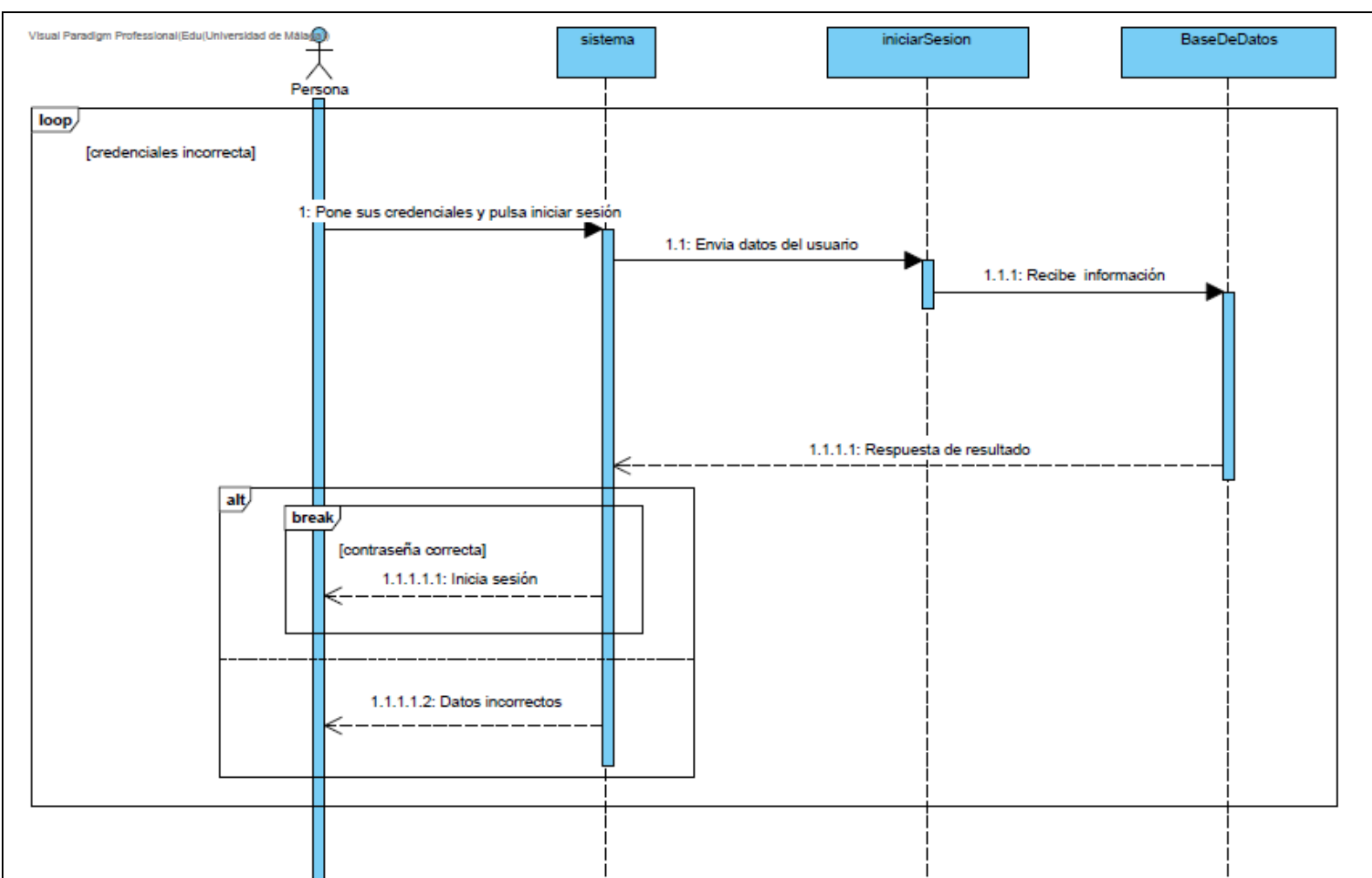
- **Iniciar sesión:** Una persona introduce sus credenciales y pulsa el botón Iniciar sesión, enviando sus datos al sistema.

El sistema envía los datos del usuario a la función `iniciarSesión`, que se comunica con la base de datos.

La base de datos devuelve al sistema el resultado de la operación al sistema, que responde al usuario si el inicio de sesión fue correcto o si los datos fueron erróneos.

Mientras que las credenciales sean incorrectas, el sistema pedirá de nuevo que introduzca sus credenciales.





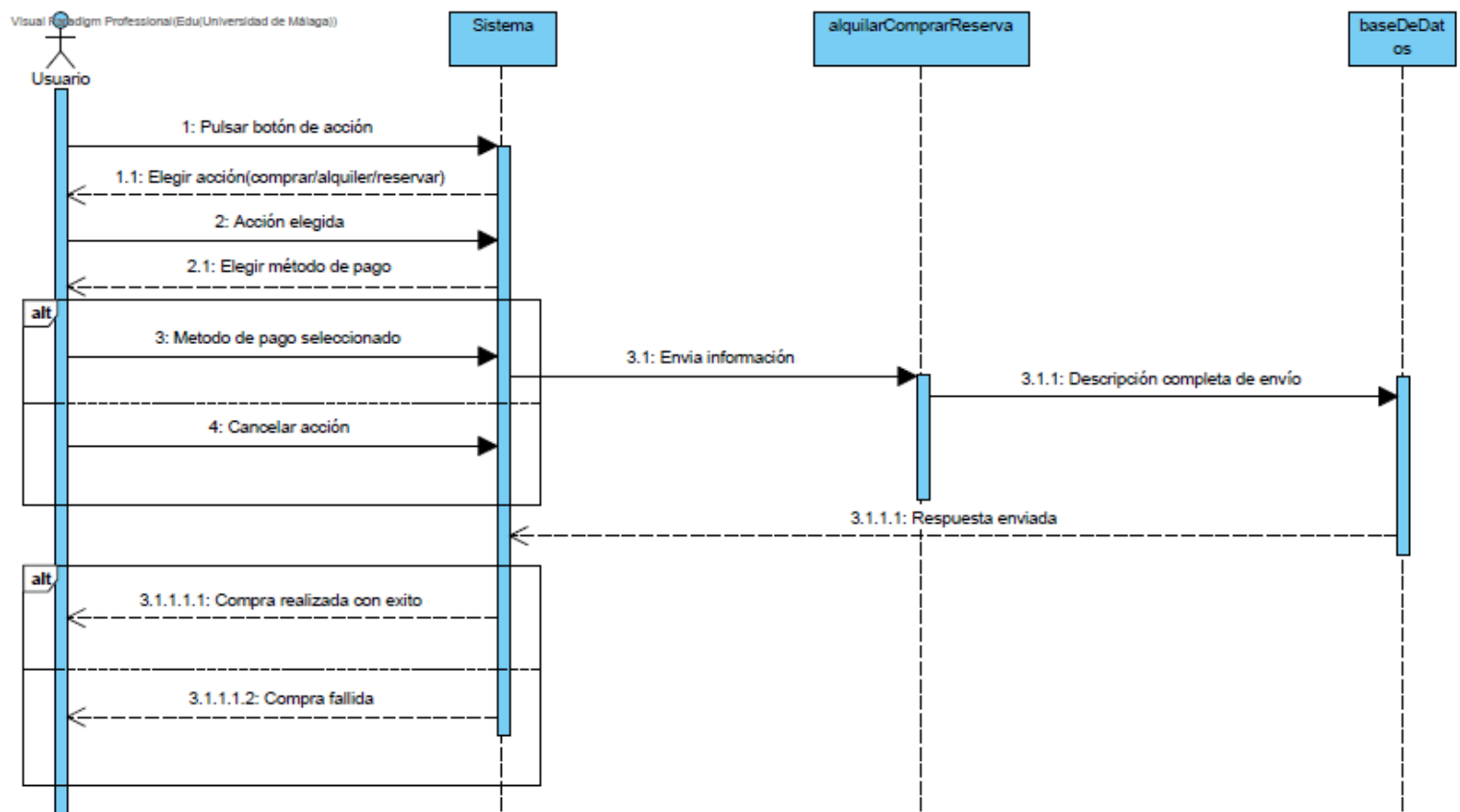
- **Usuario alquila, reserva o compra libro:** El usuario pulsa comprar/reservar/alquilar libro y se envía al sistema.

El sistema responde enviando una petición de confirmación para la elección de la acción a realizar. El usuario la selecciona y recibe los distintos métodos de pago disponibles. Posteriormente accede a uno de ellos o cancela la operación.

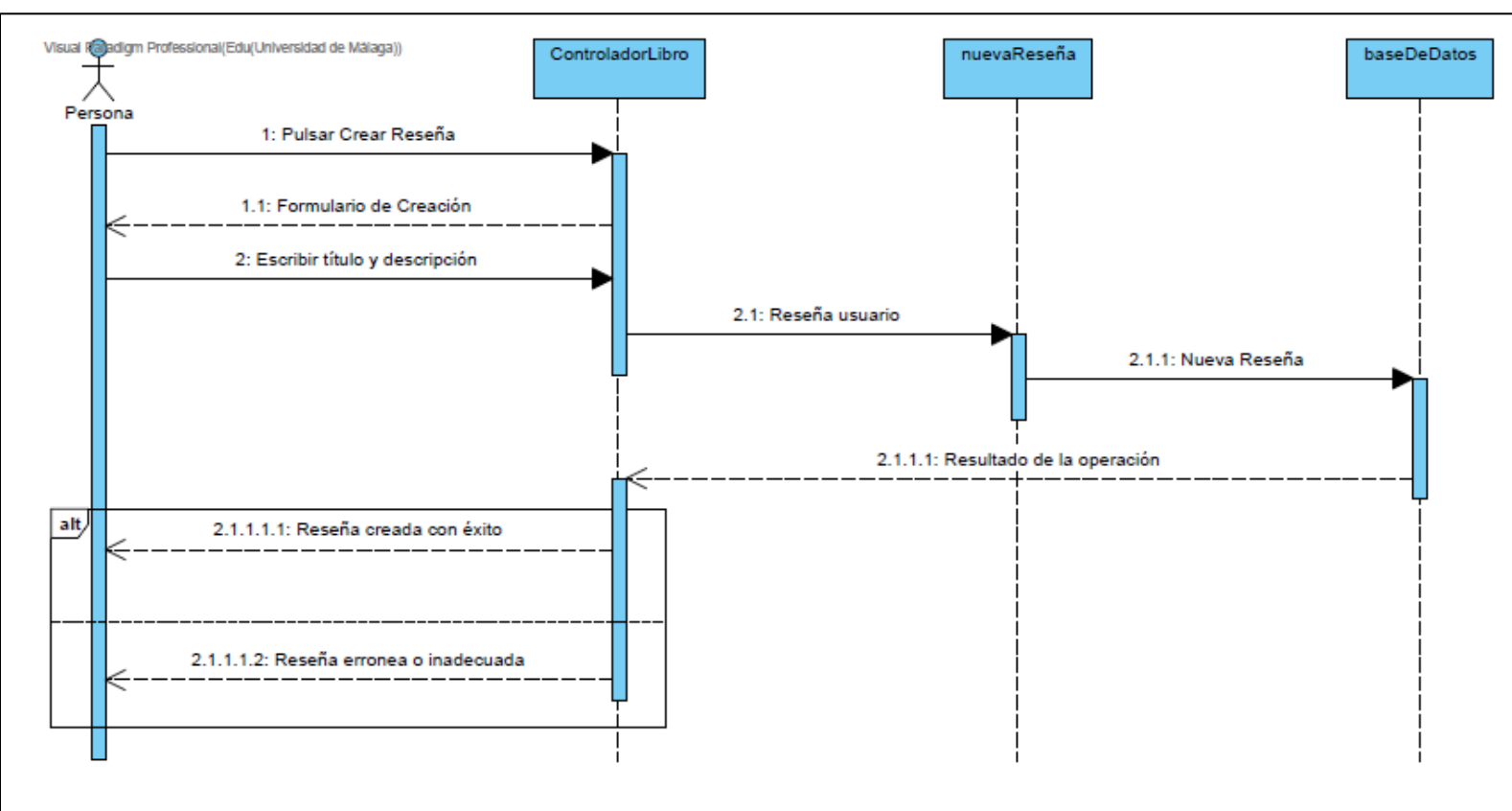
El controlador llama a la función correspondiente, que se comunica con la base de datos.

El Sistema, a través de la función “alquilarComprarReservar” envía la petición a la base de Datos.

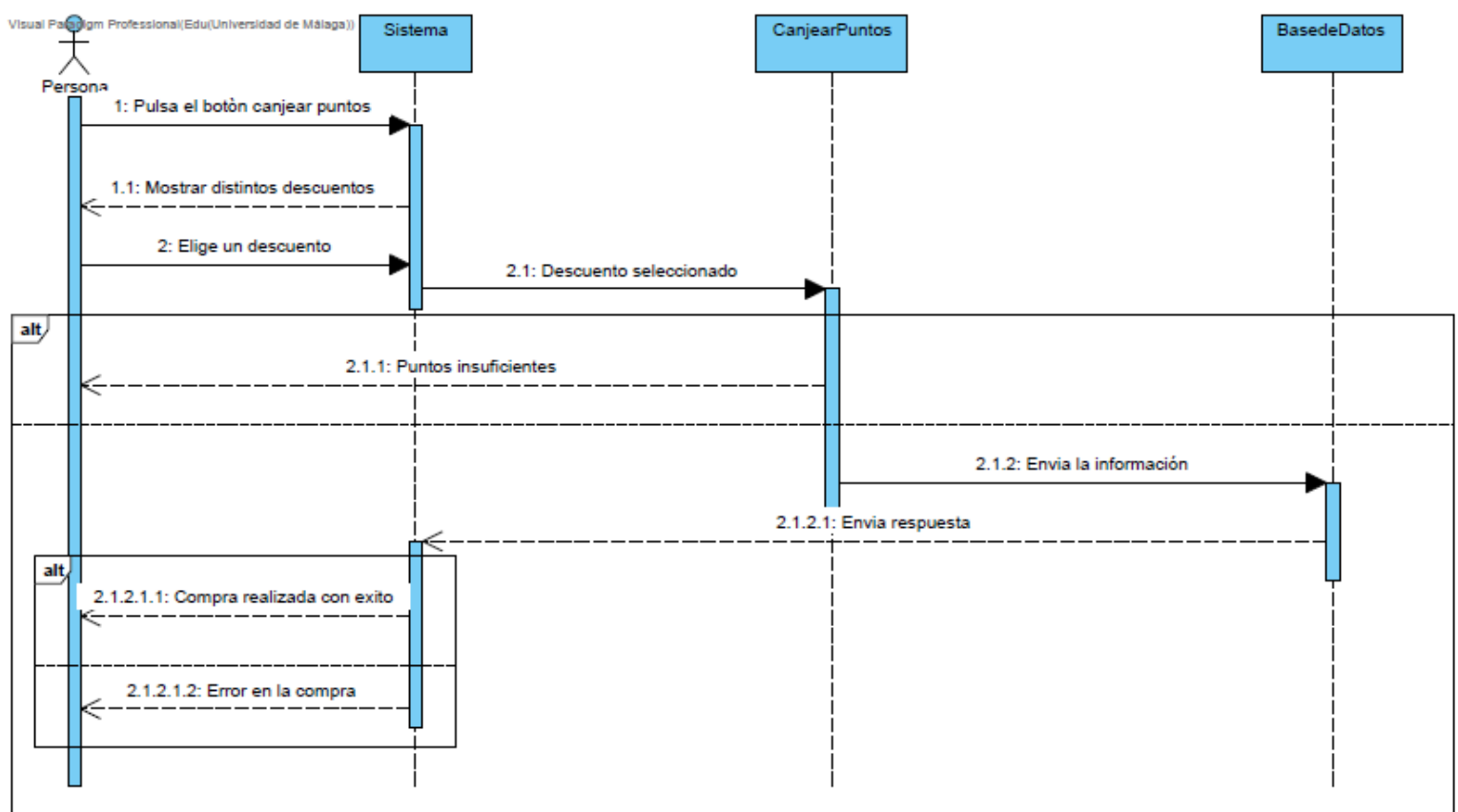
Finalmente, esta devuelve el resultado de la operación al sistema, que comunicará a la persona si el proceso ha sido exitoso o no.



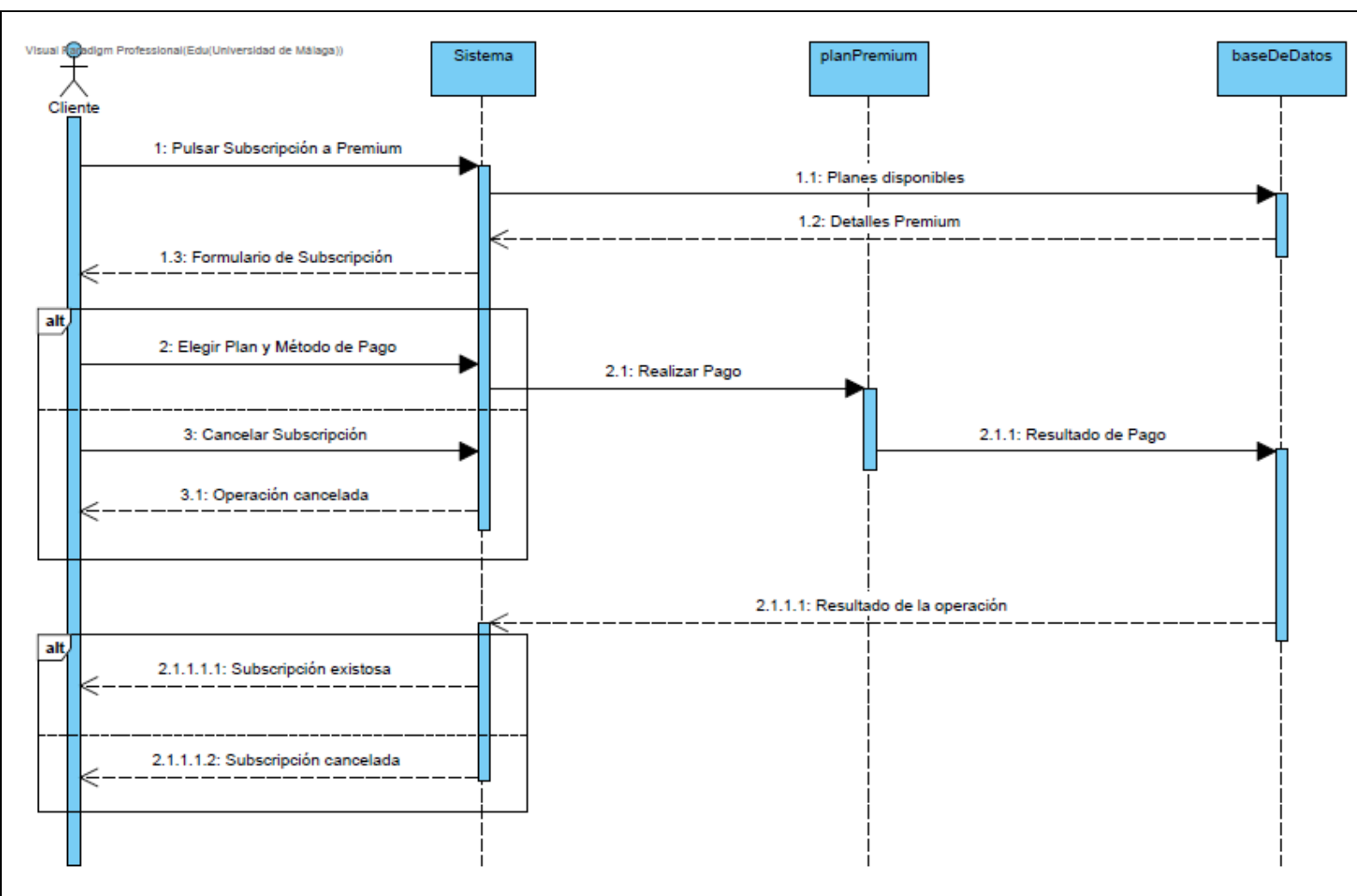
- **Crear reseña:** La persona pulsa en el botón crear reseña y el sistema responde con un formulario de reseña. La persona rellena el formulario con el título y descripción de la reseña y esta información se manda al sistema. El sistema llama a la función nuevaReseña, que manda su información a la base de datos. La base de datos devuelve el resultado de la operación al sistema. El sistema devuelve al usuario si la reseña fue creada con éxito o no.



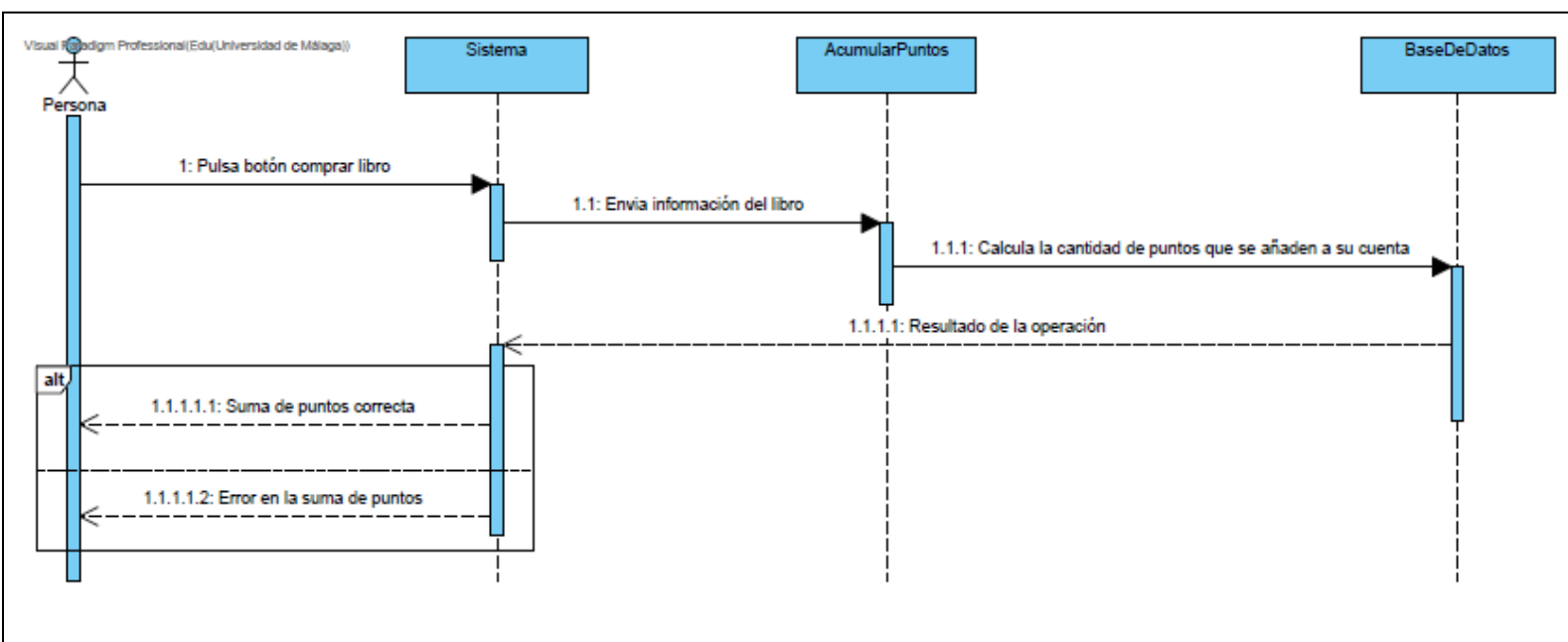
- **Canjear puntos de descuento:** Una persona pulsa el botón canjear puntos y el sistema muestra los distintos descuentos. La persona elige un descuento y el sistema llama a la función canjearPuntos. La función envía la información a la base de datos si el usuario tiene suficientes puntos. Si el usuario no tiene suficientes puntos, se devuelve al mismo un mensaje de error. Si la operación sigue adelante, la base de datos devuelve su respuesta al sistema. Si el resultado es correcto, se devuelve un mensaje de éxito al usuario. Si no, se devuelve un mensaje de error.



- **Suscribirse a plan premium:** Una persona pulsa en Suscripción Premium y el sistema devuelve un formulario de suscripción. La persona lo rellena y lo envía al sistema, que pregunta a la base de datos por los planes disponibles. La base de datos responde con los planes y sus detalles al sistema, que los envía al usuario. La persona puede elegir un plan y un método de pago o bien puede elegir cancelar la operación de pago. Si opta por la primera opción, el sistema se comunica con la función Pagar para realizar el pago. La función comunica a la base de datos el resultado del pago. La base de datos devuelve al sistema el resultado de la operación, y el sistema comunica a la persona si la suscripción fue exitosa o no. Si opta por la segunda, el sistema cancela la suscripción y devuelve un mensaje de éxito o error al usuario.



- **Acumular puntos:** La persona pulsa en comprar libro, y el sistema contacta con la función acumularPuntos. La función informa a la base de datos de los puntos a dar y ésta devuelve el resultado de la operación al sistema. El sistema devuelve al usuario si los puntos se han añadido correctamente o ha habido un error en la operación.



## 9. Arquitectura Software

Hemos escogido el estilo de arquitectura Orientada a Objetos, en el que tenemos diferentes objetos (o clases), como Crear, Login, Inicio o Register, entre otras, que se relacionan entre sí mediante llamadas.

Es un modelo de arquitectura adecuado porque la aplicación va a ser ejecutada en distintas máquinas de los usuarios. Además, es un estilo fácil de implementar, mantener y modificar, y ofrece la posibilidad de utilizar interfaces que nos facilitan la creación del programa. También es un modelo que hemos usado en otros proyectos, por lo que nos facilita su entendimiento e implementación.

Sin embargo, también nos ha dificultado la implementación de distintas clases, como Login y CrearLista, por la colaboración entre las clases.

### Implementación y pruebas

La implementación se ha realizado en el lenguaje Java con el IDE de Eclipse, al igual que las pruebas unitarias y de integración utilizando los Frameworks de Junit 5 y Mockito 3.12.4.

Se han implementado 6 casos de usos:

- CU1: Crear libro, mediante la clase Crear.java
- CU7: Crear lista, mediante la clase CrearLista.java
- CU8: Añadir libro a lista, mediante la clase verDetalles.java
- CU5: Ver detalles del libro, también mediante la clase verDetalles.java
- CU3: Eliminar libro, mediante la clase Inicio.java e InicioUsuario.java\*
- CU31: Iniciar Sesión, mediante la clase Login.java y Register.java

\*Inicio.java e InicioUsuario.java es la interfaz que muestra los libros, con las opciones de visualización y eliminación del libro si se está en el modo de Administrador, ejecutándose Inicio.java si en el Inicio de sesión se escribe en el apartado de correo electrónico cualquier usuario con ...@textheos.es. Solo se



muestra la opción de visualización si en el Inicio de sesión se escribe cualquier correo electrónico que no acabe en ...@*textheos.es*, ejecutándose así la clase de InicioUsuario.java.

Actualmente los usuarios creados son “[usuario@gmail.com](mailto:usuario@gmail.com)” con contraseña “0000” y “[admin@textheos.es](mailto:admin@textheos.es)” con la misma contraseña, “0000”.

Se pide por favor que **no se modifique de ninguna manera los .txt** del proyecto, si es el caso, podría desmoronar completamente el funcionamiento del código, ya que estos leen de cierta manera los .txt. En el caso de que ocurriera un accidente en el .txt puede eliminarlo y se generará de nuevo pero sin la información original.

Es necesario que el proyecto de java se ejecute en el IDE de Eclipse o Visual Studio Code (no hay que importar ninguna librería para que funcione, ya están importadas en el proyecto) para que su funcionamiento sea el correcto.

Las 18 pruebas unitarias realizadas son sobre las clases de Inicio de sesión.java incluyendo el Register.java, CrearLista.java y Crear.java.

Todas las pruebas se realizan ejecutando la clase de Alltest.java dentro de la carpeta “tests” que a su vez está dentro de la carpeta “src”.

## 10. Herramientas software

Comunicación		
	Trello	
	Discord	
	WhatsApp	
	Miro	
Desarrollo		
	IntelliJ	
	GitHub	
	GitKraken / GitHub Desktop	