

INSTITUTO POLITECNICO NACIONAL  
ESCUELA SUPERIOR DE COMPUTO

## **Ingeniería de software**

Documento de requerimientos

Proyecto individual

### **Profesor**

GABRIEL HURTADO AVILÉS

### **Alumno**

Hernandez Diaz Roberto Angel

### **Grupo**

6CV3

## Indice

Introducción .....	3
Objetivos .....	3
Arquitectura del sistema .....	3
Tipos de usuarios dentro del sistema .....	4
Requerimientos funcionales .....	5
Requerimientos no funcionales .....	5
Casos de uso .....	6
Herencias dentro del sistema .....	6
Caso de uso: Login/Register .....	6
Caso de uso: Principal User Sistem .....	9
Caso de uso: Admin Controller.....	11

## Introducción

Dentro de del documento se muestra de forma explicita los lineamientos técnicos necesarios para el desarrollo e implementación del proyecto individual, el cual consiste en una aplicación web orientada a los usuarios y a la consulta de ciertos títulos literarios. Para ello, se integrará una API que permitirá obtener y mostrar dicha información, a su vez integra la posibilidad de personalizar la interfaz mediante temas claro y oscuro.

Esta aplicación también incluye funciones de autenticación y manejo de usuarios, sumando a las sus capacidades para brindar un servicio de búsqueda y consulta de libros.

## Objetivos

- Desarrollo de un sistema de de gestión de control de usuarios, diferenciados por los permisos de modificación de datos sobre otros usuarios dados por administrador y usuario
- Integrar la funcionalidad de una API dada por OpenLibrary hacia la consulta sobre títulos literarios
- Configuración personalizada basada en los temas claros o oscuros del usuario

## Arquitectura del sistema

### 1. Arquitectura General

La aplicación está desarrollada bajo el patrón de arquitectura MVC (Modelo-Vista-Controlador) utilizando ASP.NET Core, separando responsabilidades y facilitando el mantenimiento y escalabilidad de la app.

### 2. Componentes Principales

## Modelo

- Representa la lógica de negocio y la estructura de datos.
- Aquí se definen las clases que representan las entidades del dominio.
- Incluyendo las clases que gestionan el acceso a datos, como el uso de Entity Framework Core para interactuar con la base de datos.

## Vista (

- Son las páginas que ve el usuario, dados por archivos .cshtml.
- Se encargan de presentar los datos y permiten la interacción con el usuario.
- Utilizan Razor para combinar HTML con código C#.
- 

## Controlador

- Gestiona la lógica de flujo entre el modelo y la vista.
- Recibe peticiones del usuario, llama a los modelos necesarios, y retorna la vista correspondiente.
- Cada controlador generalmente corresponde a una entidad o funcionalidad.

## Agregando una capa adicional llamada Services

Servicios: Agregando clases y funcionalidades adicionales por parte de la API que proporciona los datos

## Tipos de usuarios dentro del sistema

- **Usuario regular:**
  - Características: Usuario estándar de la aplicación, contando con un nombre de usuario que sirve como vinculo a su cuenta, una contraseña a usar, además de datos como su primer nombre y apellido, una breve descripción estos, además de una foto de usuario
- Capacidades:
  - Poder ingresar al sistema mediante una autenticación
  - Navegar entre los distintos libros del sistema
  - Modificar la información de su cuenta
- Conocimientos previos:
  - Conocimientos básicos en el uso de la web
- **Administrador:**
  - Características: Usuario administrador de la aplicacion de la aplicación, contando con un nombre de usuario que sirve como vinculo a su cuenta, una contraseña a usar, además de

datos como su primer nombre y apellido, una breve descripción estos, además de una foto de usuario

- Capacidades
  - Poder ingresar al sistema mediante una autenticación
  - Navegar entre los distintos libros del sistema
  - Modificar la información de su cuenta
  - Poder modificar información de otros usuarios del sistema
- Conocimientos previos
  - Conocimientos avanzados en el uso de la web
  - Conocimientos en gestión de sistemas y gestión de usuarios

## Requerimientos funcionales

- Inicio de sesión funcional: El sistema ofrecerá una interfaz donde los usuarios podrán ingresar sus credenciales personales para validar su identidad y obtener acceso a las funcionalidades disponibles según su rol.
- Creación de nuevas cuentas: Será posible registrar usuarios mediante un formulario que capture información esencial, permitiendo así que nuevas personas puedan utilizar la plataforma.
- Gestión de sesión activa: Cada vez que alguien acceda al sistema, se validará si los datos introducidos coinciden con un perfil registrado. Si todo es correcto, se iniciará la sesión y se adaptará la experiencia según el tipo de usuario.
- Visualización de libros disponibles: Se mostrará al usuario un listado con los libros que están en existencia dentro del sistema, incluyendo detalles como nombre, autor.
- Sugerencias personalizadas de lectura Basándose en los gustos o comportamiento del usuario, el sistema podrá recomendar libros que se alineen con sus preferencias, incentivando así la exploración de nuevos títulos.

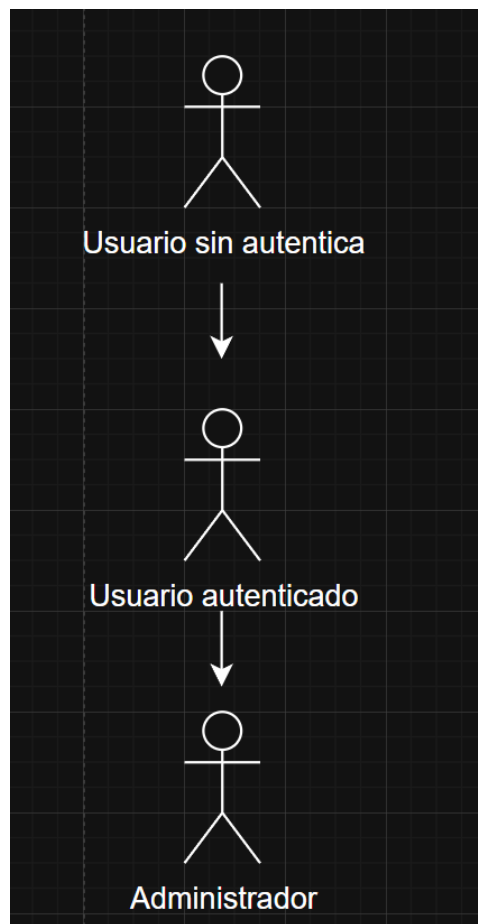
## Requerimientos no funcionales

- Rendimiento y velocidad: La aplicación deberá responder con fluidez, garantizando que las páginas carguen en pocos segundos, incluso cuando múltiples usuarios la estén utilizando de manera simultánea.

- Compatibilidad entre navegadores: El sistema funcionará correctamente en los navegadores web más utilizados como Chrome, Firefox, Edge y Safari, evitando errores o fallas en la presentación de la interfaz.
- Escalabilidad: El diseño del software permitirá incorporar nuevas funcionalidades o aumentar su capacidad conforme crezca la cantidad de usuarios o el volumen de libros registrados, sin afectar el rendimiento.
- Interfaz intuitiva: El diseño de la aplicación debe ser sencillo de entender y utilizar, incluso para personas sin conocimientos técnicos, minimizando la necesidad de capacitaciones.

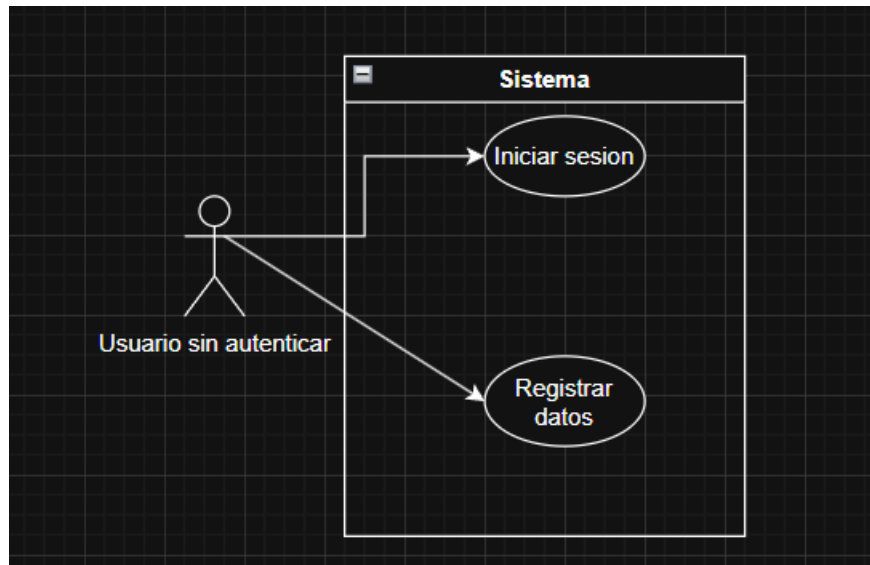
## Casos de uso

### Herencias dentro del sistema



### Caso de uso: Login/Register

#### Diagrama relacionado



## Especificación

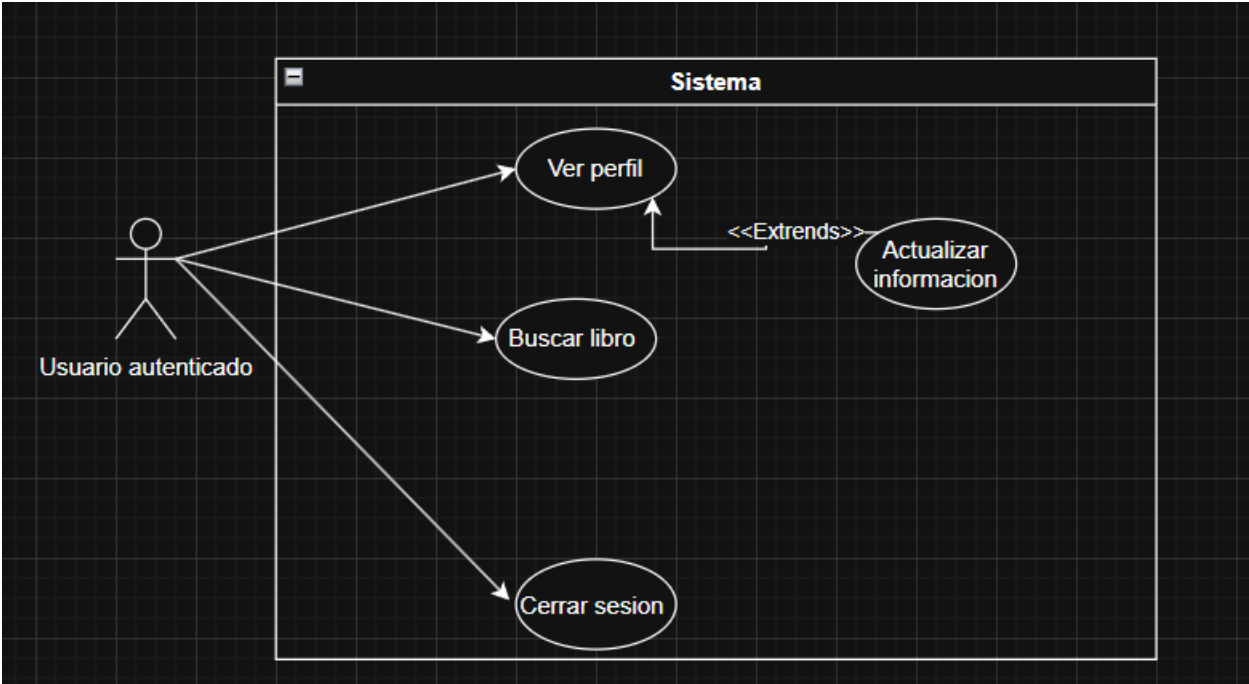
Nombre CU:	Login/Register	
Dependencias	Usuario sin autentificacion y sistema	
Descripción	Permitir que el usuario ingrese al sistema	
Precondición	El usuario debe de tener una cuenta dentro del sistema	
Secuencia normal	Paso	Acción
	1	El usuario selecciona la opción aludida al ingreso del sistema
	2	Sistema solicita llenado de campos
	3	Usuario llena los campos solicitados
	4	Sistema valida la información
	5	Sistema permite dar acceso a la vista principal del usuario
Postcondiciones	El usuario se encuentra en la vista principal	

Flujo alterno	Paso	Acción
	1	El usuario no cuenta con un perfil dentro del sistema
	2	El usuario selecciona la opción aludida al registro de sus datos
	3	El sistema muestra los campos a llenar
	4	El usuario llena los campos solicitados
	5	Sistema valida la información
	6	Sistema retorna a la página principal
	7	El usuario selecciona la opción aludida al ingreso del sistema
Comentarios		
CU que propicia el acceso al sistema		



## Caso de uso: Principal User Sistem

Diagrama relacionado



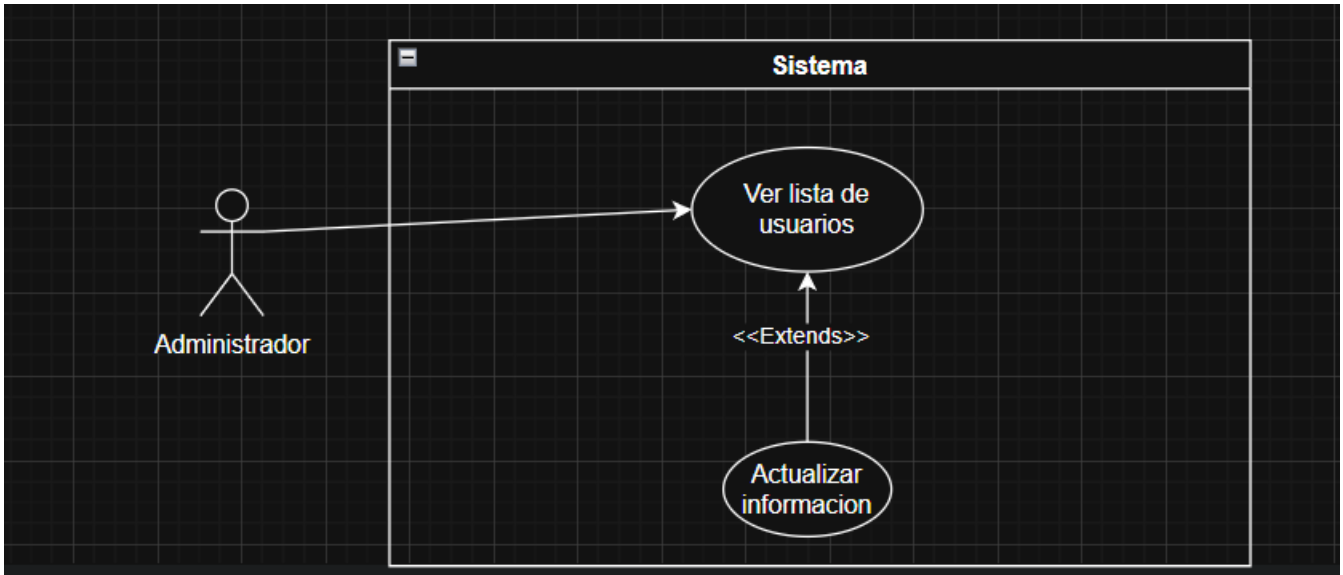
Especificación

Nombre CU:	Principal User Sistem
Dependencias	Usuario autenticado y sistema

Descripción	Ya iniciado el sistema el usuario podrá realizar múltiples acciones, entre ellas la principal de buscar una noticia	
Precondición	El usuario debe de tener una cuenta dentro del sistema y haber accedido con ella al sistema	
Secuencia normal	Paso	Acción
	1	El usuario colocara el titulo o genero de su preferencia dentro de la barra de búsqueda
	2	Sistema retornara una serie de libros basada en la búsqueda del usuario
Postcondiciones	El usuario visualiza los libros dados por el sistema	
Flujo alterno 1	Paso	Acción
	1	El usuario selecciona la opción aludida a la visualización de su perfil
	2	Sistema devuelve la información de la cuenta del perfil
	3	El usuario modifica los datos de su perfil
	5	Sistema valida la información
	6	Sistema retorna a la página principal
Flujo alterno 2	1	El usuario selecciona la opción aludida a salir del sistema
	2	El sistema cierra sesión y retorna a la página de inicio
Comentarios		
CU que da a conocer las acciones dentro del sistema		

## Caso de uso: Admin Controller

Diagrama relacionado



Especificación:

Nombre CU:	Principal Admin Controller
Dependencias	Administrador y sistema
Descripción	Ya iniciado el sistema dentro de la página principal, el administrador podrá entrar a una sección especial para su cargo donde administrara los usuarios

Precondición	El usuario debe de tener una cuenta dentro del sistema y tener el rol de administrador	
Secuencia normal	Paso	Acción
	1	El administrador deberá acceder a la acción aludida a ver usuarios
	2	Sistema retornara una vista donde se listaran todos los usuarios
Postcondiciones	El usuario visualiza los libros dados por el sistema	
Flujo alterno 1	Paso	Acción
	3	El administrador seleccionara la acción aludida a la modificación de un usuario en concreto
	3	Sistema devuelve la información de la cuenta del perfil, así como los campos para ser modificados
	3	El Administrador modifica los datos de su perfil
	5	Sistema valida la información
	6	Sistema retorna a la página de usuarios
Comentarios		
CU que muestra las acciones con privilegios sobre usuarios referentes a la modificación del usuarios		

