

## **Ejercicio: Sistema de Reserva de Libros en una Biblioteca**

**Contexto:** Una biblioteca moderna quiere ofrecer a sus usuarios una aplicación que facilite el proceso de reserva de libros. La idea es que tanto los usuarios como los administradores de la biblioteca tengan acceso a las funciones esenciales para consultar, reservar y gestionar libros.

La biblioteca cuenta con un catálogo de libros de diferentes géneros y autores, y no todos los libros están disponibles en todo momento. Para evitar que los usuarios se desplacen en vano hasta la biblioteca, el sistema permite que cada usuario consulte la disponibilidad de los libros de manera virtual. Los usuarios podrán ver el estado de un libro y, si está disponible, hacer una reserva para asegurarse de que estará disponible cuando lo necesiten.

Los administradores son los encargados de mantener actualizado el catálogo de libros, añadir nuevas adquisiciones, modificar la información de los libros existentes y eliminar aquellos que ya no están en circulación. Además, los administradores deben aprobar o cancelar las reservas realizadas por los usuarios, asegurándose de que se respete el orden de las reservas y de que cada usuario reciba confirmación del estado de su solicitud.

### **Escenarios de Uso Comunes**

#### **1. Usuario buscando un libro:**

- María, una usuaria, accede al sistema desde su dispositivo. Busca el libro "Cien años de soledad" y verifica si está disponible. Si el libro está en estado "disponible", María podrá realizar la reserva y el sistema notificará al administrador, quien debe aprobarla para que quede confirmada.

#### **2. Administrador gestionando el catálogo:**

- Omar, el administrador, se encarga de revisar las reservas pendientes cada día. Además, él debe actualizar el catálogo cuando llega un nuevo lote de libros. Hoy ha añadido tres nuevos libros y ha cambiado el estado de "Cien años de soledad" a "reservado" porque María ha hecho una reserva.

#### **3. Reserva pendiente y notificación:**

- María realiza una reserva, pero el sistema la coloca en estado "pendiente". Una vez que Omar aprueba la reserva, el sistema notifica a

María que puede recoger el libro en la biblioteca en los próximos tres días. Si por algún motivo Omar necesita cancelar la reserva, María también recibiría una notificación explicando el cambio.

## **Funcionalidades del Sistema**

Para facilitar estos escenarios, el sistema debe contar con las siguientes funciones:

### **1. Consultar Disponibilidad de Libros:**

- Los usuarios pueden buscar libros y ver información básica (título, autor, género) y el estado actual (disponible o reservado).

### **2. Reservar un Libro:**

- Si el libro está disponible, los usuarios pueden realizar una reserva. Esta acción lo cambia a "pendiente" hasta que el administrador lo apruebe o cancele.

### **3. Gestionar el Catálogo de Libros:**

- Los administradores pueden añadir nuevos libros, modificar los existentes (actualizar detalles o estado) y eliminar libros que ya no están disponibles en la biblioteca.

### **4. Aprobar o Cancelar Reservas:**

- Los administradores revisan las reservas pendientes y pueden aprobarlas (cambiando el estado del libro a "reservado") o cancelarlas.

### **5. Notificaciones a Usuarios:**

- Cada cambio en el estado de una reserva debe generar una notificación automática para informar al usuario.

## **Instrucciones**

### **1. Diagrama de Clases**

**Objetivo:** Modelar los elementos principales del sistema y las relaciones entre ellos.

#### **1. Identifica las Clases Principales:**

- TIPS: Piensa en los roles involucrados (¿quiénes interactúan con el sistema?) y los elementos sobre los que actúan (¿qué elementos gestionan o usan?).
- TIPS: ¿Qué objetos necesita el sistema para funcionar correctamente? ¿Qué roles principales existen en el sistema y qué objetos manejan?

#### **2. Define los Atributos y Métodos:**

- TIPS: Imagina qué información necesita cada clase para cumplir su función en el sistema.
- TIPS: ¿Qué datos debe almacenar cada clase? ¿Qué acciones debe poder realizar cada clase?
- TIPS: Los métodos suelen ser acciones que cada objeto puede realizar en el sistema o sobre otros objetos

#### **3. Establece las Relaciones:**

- TIPS: Piensa en cómo interactúan o dependen unas clases de otras.
- TIPS: ¿Cuál clase necesita estar conectada con otra para que el sistema funcione? ¿Qué relaciones lógicas existen entre estas clases?

## **2. Diagrama de Casos de Uso**

**Objetivo:** Mostrar las interacciones entre los actores (usuarios y administradores) y el sistema.

### **1. Identifica los Actores:**

- TIPS: Reflexiona sobre quiénes interactúan con el sistema y qué roles cumplen.
- TIPS: ¿Qué tipos de usuarios existen en el sistema? ¿Quiénes realizan acciones o toman decisiones en él?

### **2. Define los Casos de Uso:**

- TIPS: Piensa en qué acciones son necesarias para que el sistema cumpla su propósito.
- TIPS: Pregunta: ¿Qué acciones importantes deben realizar cada uno de los actores? ¿Qué situaciones se dan en el sistema en las que los actores necesitan realizar una acción?

### **3. Agrega Detalles de Cada Caso de Uso:**

- TIPS: Describe cada caso de uso en términos generales, pensando en los pasos que se siguen para cumplir la acción.
- TIPS: Pista: Algunos casos de uso pueden depender de otros o necesitar pasos adicionales (por ejemplo, una acción que depende de otra para completarse).