

Sistema de información bibliotecario

Juan Sebastian Barragan Vigoya

Código: 1710010192

Proyecto de aula

Profesor

Isabel Mahecha

Politecnico Grancolombiano

Facultad de ingeniería

Ingenieria de software 1

2020

## Contenido

1. Descripción del sistema
2. Requerimientos:
  - 2.1. Diagrama de casos de uso
  - 2.2. Especificación de casos de uso
  - 2.3. Requerimientos no funcionales
  - 2.4. Modelo de datos (Diagrama de Clases o E/R)
  - 2.5. Diagramas de objetos
  - 2.6. Diagramas de Secuencia
  - 2.7. Diagramas de estados
3. Diseño
  - 3.1. Diagrama de componentes
  - 3.2. Diagrama de despliegue
  - 3.3. Diseño de la interfaz gráfica (Mockups)
4. Prototipo final
  - 4.1. Instructivo básico de uso con imágenes y explicación de las funciones que tiene cada pantalla.
  - 4.2. Documentación de pruebas
  - 4.3. Trabajo futuro o posibles mejoras
5. Lecciones aprendidas
  - 5.1. Requerimientos
  - 5.2. Desarrollo de software
  - 5.3. Trabajo en grupo
6. Bitácora del proyecto
7. Actas de reuniones
8. Repositorio git

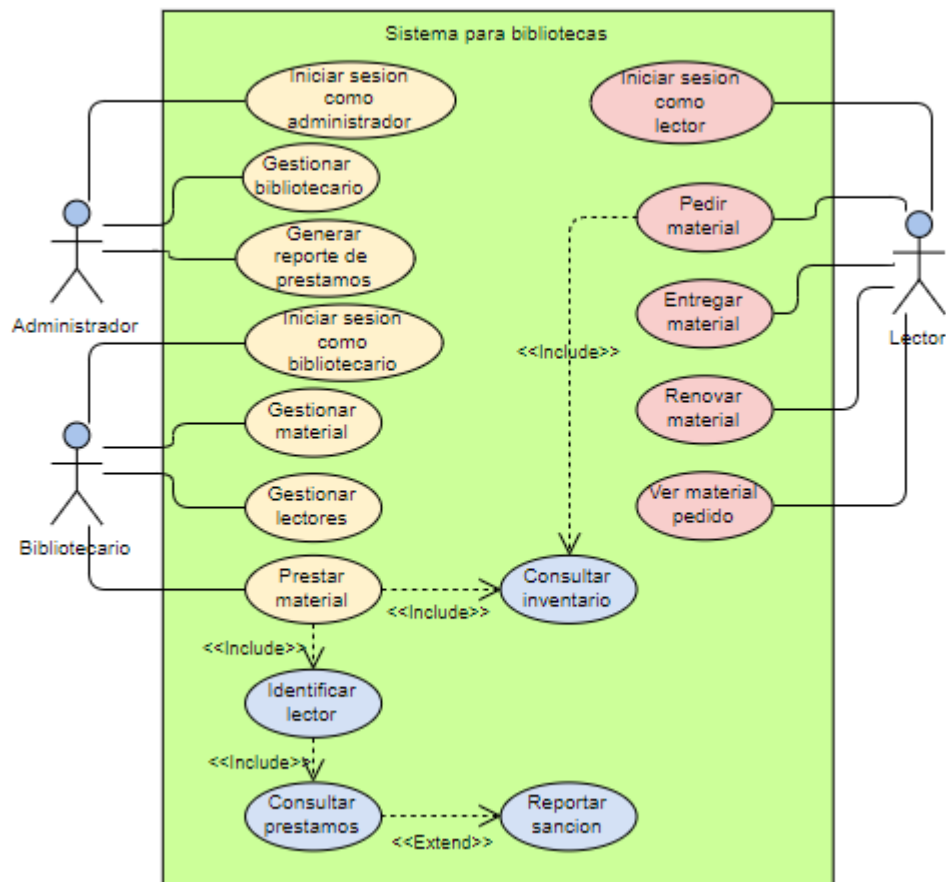
## Contenido

### 1. Descripción del sistema

Requerimientos actualizados

### 2. Requerimientos

#### 2.1. Diagrama de casos de uso



#### 2.2. Especificación de casos de uso

JB-1001	Gestionar usuarios
Versión	1.0 (05/03/2009)
Dependencias	1. Verificar en sistema los usuarios ingresados al sistema 2. Registrar en sistema lectores

Precondición	El administrador debe iniciar sesion para ingresar al sistema	
Descripción	Gestionar usuarios	
Secuencia normal	Paso	Acción
	1	El administrador inicia sesion en el sistema
	2	El administrador da inicio a gestionar usuario
	3	El sistema pide seleccionar si desea añadir, modificar o eliminar usuarios
	4	El sistema pide datos del usuario
	5	El sistema guarda la información en una base de datos
	8	El administrador cierra sesión.
Postcondición	El usuario ya puede ingresar al sistema	
Excepciones	4	Si el usuario a gestionar no existe
Comentarios		

JB-1002	Reservar material	
Versión	1.0 (05/03/2009)	
Dependencias	1. Verificar en sistema el material disponible y prestado 2. Registrar en sistema el material reservado	
Precondición	El lector debe haber iniciado sesion.	
Descripción	Reservar material bibliotecario	
Secuencia normal	Paso	Acción
	1	El usuario inicia sesion en el sistema
	2	El usuario busca el material que quiere
	3	El sistema muestra el material disponible
	4	El usuario elige el material
	5	El sistema sube a la base de datos la reserva
	8	El usuario cierra sesión
Postcondición	El usuario tiene 3 días para recoger el material o se borrara la reserva	
Excepciones	6	El cliente no acepta el precio y no lleva el producto

Comentarios	
-------------	--

JB-1003	Registrar préstamo		
Versión	1.0 (05/03/2009)		
Dependencias	1. Gestionar prestamos 2. Fecha de devolución		
Precondición	El bibliotecario debe iniciar sesión como bibliotecario		
Descripción	El usuario solicita un préstamo de material		
Secuencia normal	Paso	Acción	
	1	El administrador inicia sesión como administrador	
	2	El sistema solicita que se identifique el lector	
	3	El bibliotecario da los datos del lector ya registrado por el administrador al sistema	
	4	El sistema pide datos de identificación del material	
	5	El administrador ingresa datos del material	
	6	El sistema indica la fecha de la entrega del material	
	7	El sistema confirma el préstamo	
	8	El sistema registra el prestamos correctamente	
Postcondición	El usuario se lleva el material y el sistema lo registra		
Excepciones	Paso	Acción	
	3	Si el usuario tiene alguna sanción	
		E.1	El sistema impide realizar el préstamo
		E.2	El administrador informa de la sanción
		E.3	Se cancela el préstamo

## 2.3. Requerimientos funcionales y no funcionales

### Requerimientos funcionales

## Proceso

- El sistema deberá permitir al administrador y lectores registrarse.
- El sistema enviar un correo electrónico cuando se registre algún lector en el sistema de la biblioteca.
- El sistema deberá solicitar una contraseña para que los usuarios puedan interactuar con la aplicación.
- El sistema debe tener una base de datos con la información de los libros e indicando nombre, categoría e id.
- El sistema deberá permitir al administrador gestionar lectores.
- El sistema deberá permitir al administrador añadir, modificar y eliminar cualquier material bibliotecario.
- El sistema deberá permitir al lector consultar todo el material de la base de datos.
- El sistema deberá permitir al lector reservar cualquier material.
- El sistema deberá permitir al lector cancelar la reserva.
- El sistema deberá permitir al administrador prestar material.
- El sistema deberá generar un reporte de cada préstamo indicando idPréstamo, nombreMaterial, idMaterial, fechaDevolucion, nombreLector y idLector.
- El sistema deberá permitir al lector renovar el préstamo.
- El sistema deberá permitir al administrador saber el material prestado y cual está disponible.
- El sistema deberá generar reportes semanales de préstamos de libros.

## Interfaz grafica

- La interfaz gráfica debe estar para móvil con sistema operativo Android 5.0
- En el menú se podrá acceder al inicio ver las categorías los libros y ajustes en la cuenta del lector
- La pantalla del inventario muestra los libros por id nombre y categoría
- En la pantalla de cada libro saldrá si está disponible o no
- Al reservar el libro la pantalla arrojará un mensaje para poder pasar por el libro a la biblioteca
- Al recoger el libro la pantalla arroja mensaje de que el libro está prestado
- La arrojaron mensaje cuando el libro este cercano a la devolución

## Seguridad

- El sistema controlará el acceso y permitirá únicamente a los usuarios autorizados entrar a la aplicación
- Los usuarios entraran al sistema indicando su usuario y contraseña
- El sistema enviara un correo electrónico cuando haya modificaciones de usuario
- El administrador puede modificar los registros del sistema

## Interfaces externas

- El sistema estará implementado bajo la plataforma Android
- El sistema deberá ser compatible para funcionar con Android 5.0 en adelante
- El sistema no requiere de licenciamiento de software adicional para su funcionamiento

## **Requerimientos no funcionales**

### Eficiencia

- La aplicación móvil debe funcionar desde Android 5.0 en adelante
- Cada interacción o funcionalidad de la aplicación con el usuario no debe durar más 10 segundos
- El sistema debe ser capaz de operar adecuadamente con hasta 1000 usuarios concesiones ocurrentes
- La aplicación no consumirá tanta batería
- La aplicación tendrá un peso aproximado de 300 MB
- Cada modificación de datos que haga el usuario debe guardarse en la base de datos aproximadamente en 4 segundos
- La información modificada en el inventario debe reflejarse de forma inmediata para todos los usuarios que acceden al sistema

## Usabilidad

- La aplicación será muy fácil de aprender a usar máximo 2 horas
- La aplicación contará con manuales de usuario debidamente documentados y adaptados en un lenguaje sencillo y claro
- El sistema deberá arrojar procedimientos exitosos
- En caso de error el sistema de enviar mensajes que sean claros e informativos para el usuario
- La aplicación debe tener buenas interfaces gráficas con diseño responsive.

## Seguridad lógica y de datos

- El acceso de datos al sistema sólo se cambiará por el administrador
- El sistema se respaldará los datos almacenados exitosamente en la base de datos
- El sistema se desarrollará bajo las mejores prácticas y recomendaciones de seguridad para disminuir vulnerabilidades en el código
- Se deben realizar respaldos semanales de la base de datos y deberán almacenarse en una ubicación diferente de donde está instalado el sistema
- Todas las comunicaciones del cliente deben estar encriptadas con el algoritmo RSA
- Se deben aplicar las actualizaciones del sistema operativo de forma periódica

## Ambiente de desarrollo

- La aplicación estará desarrollada para dispositivos Android por medio del programa Android Studio

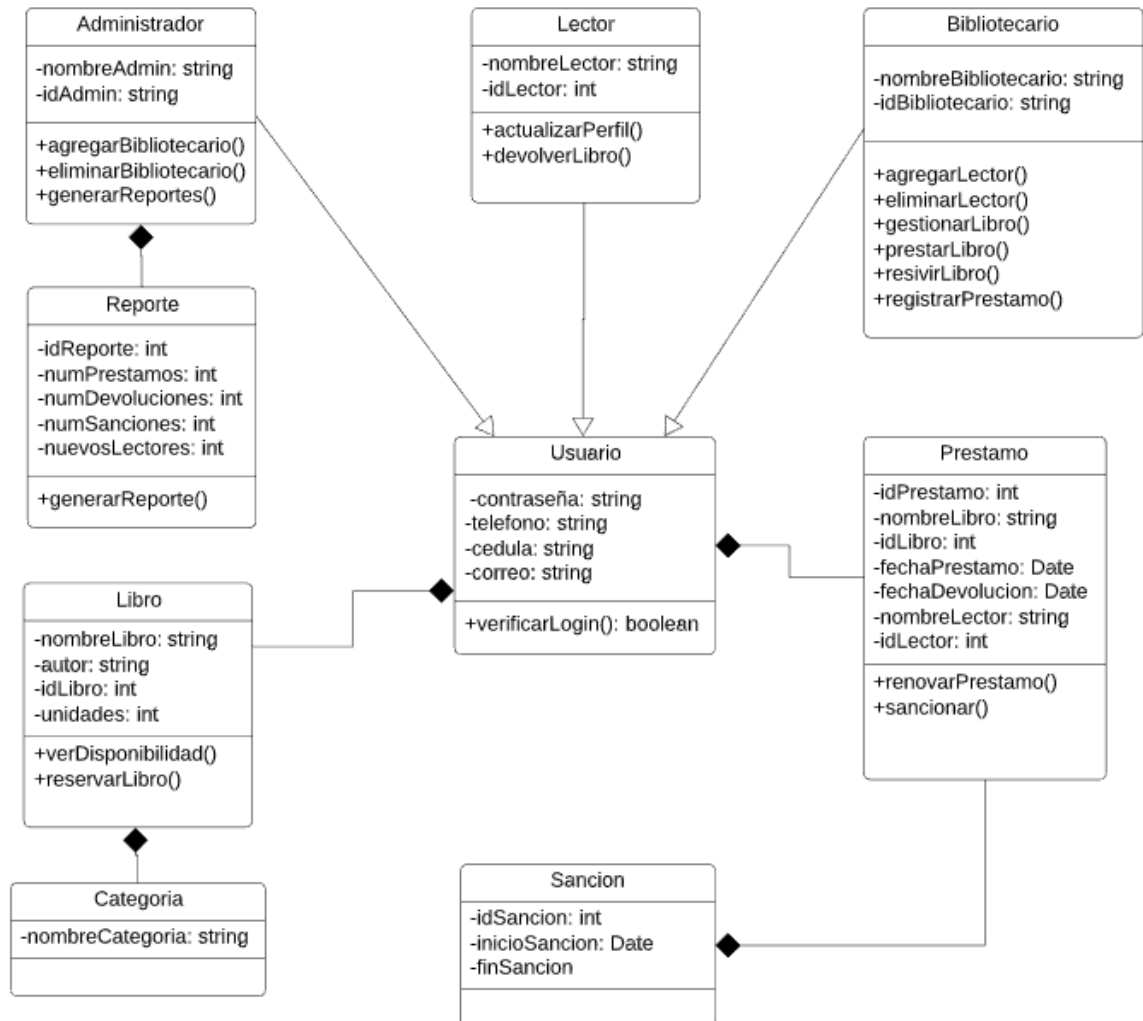


- Lenguaje de programación Java

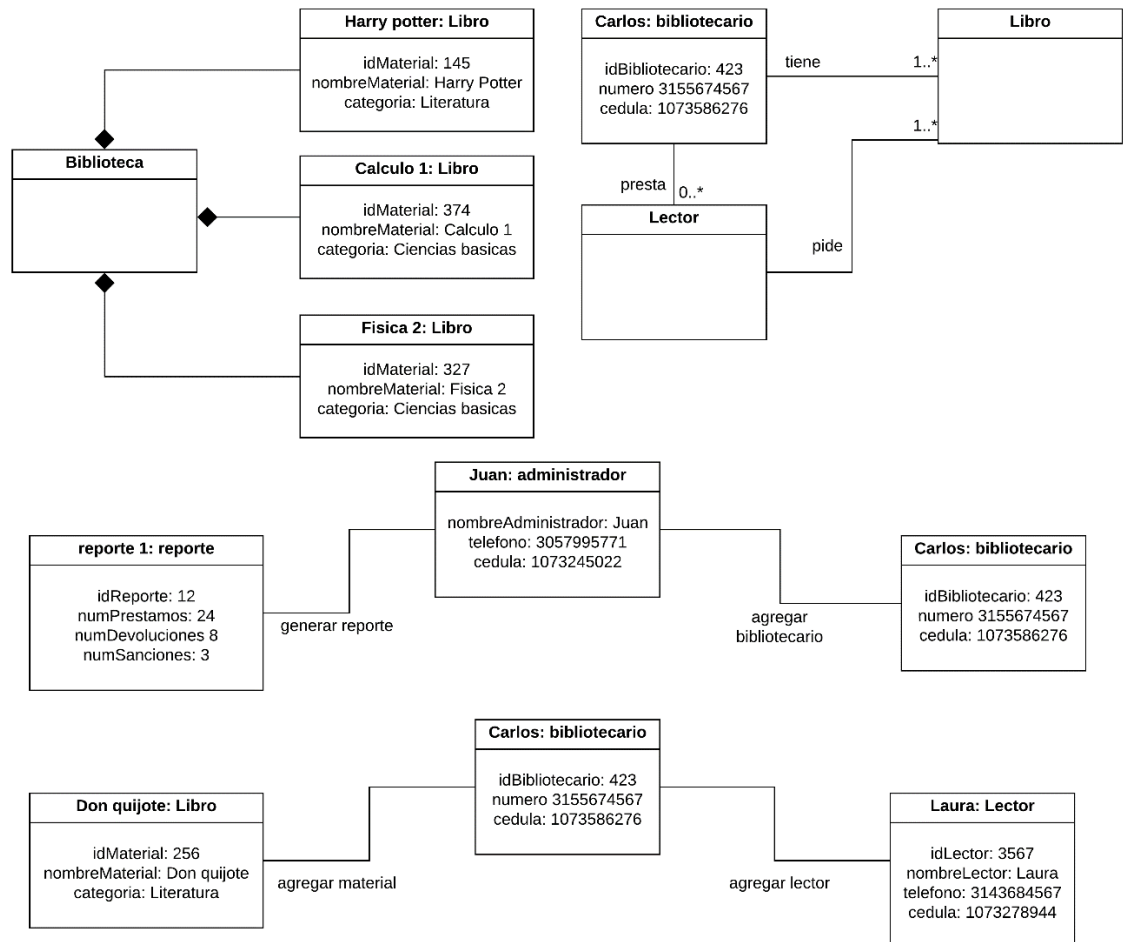
## Dependibilidad

- El sistema debe tener una disponibilidad del 99,99% de las veces en que un usuario intente accederlo.
- El tiempo para iniciar o reiniciar el sistema no podrá ser mayor a 5 minutos.
- La tasa de tiempos de falla del sistema no podrá ser mayor al 0,5% del tiempo de operación total.
- El promedio de duración de fallas no podrá ser mayor a 15 minutos.  
La probabilidad de falla del Sistema no podrá ser mayor a 0,05.

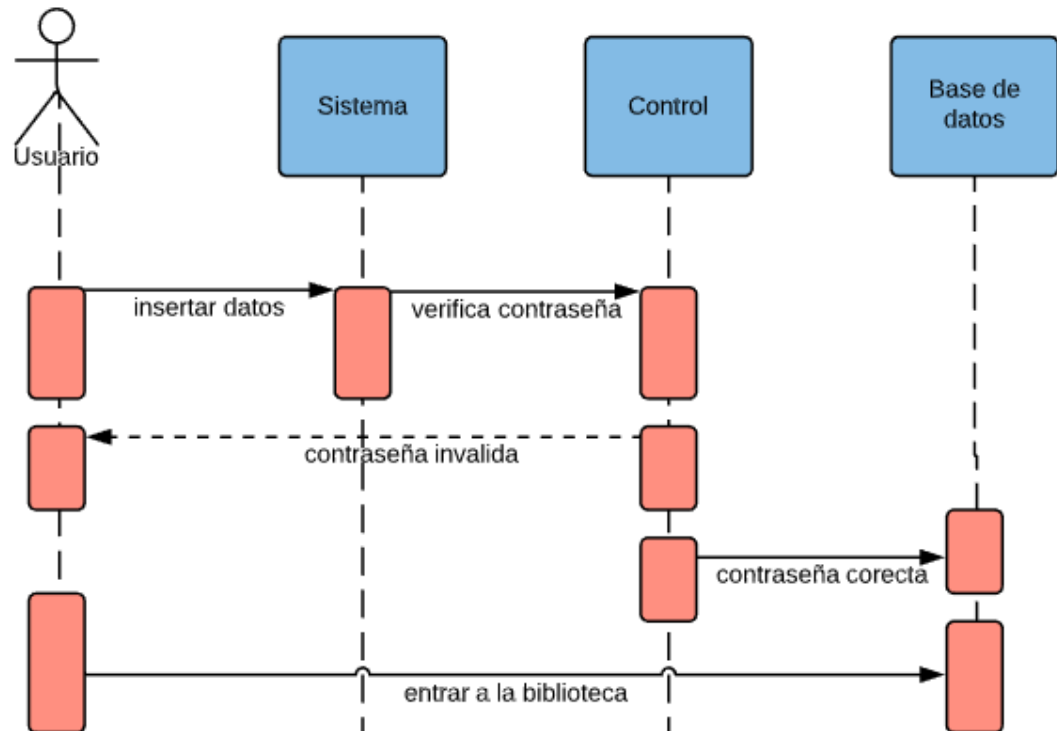
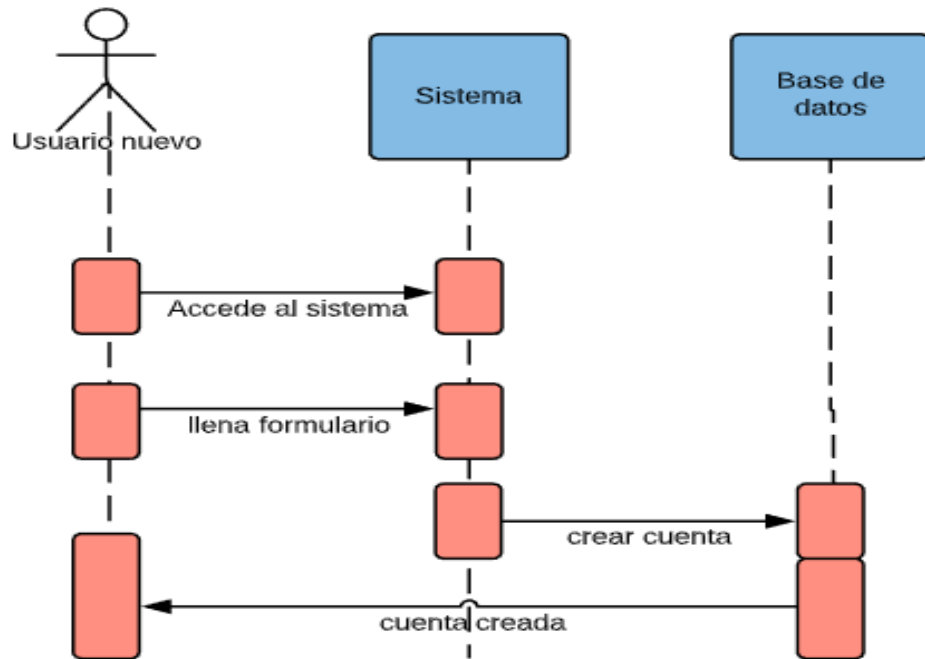
## 2.4. Modelo de datos (Diagrama de Clases o E/R)

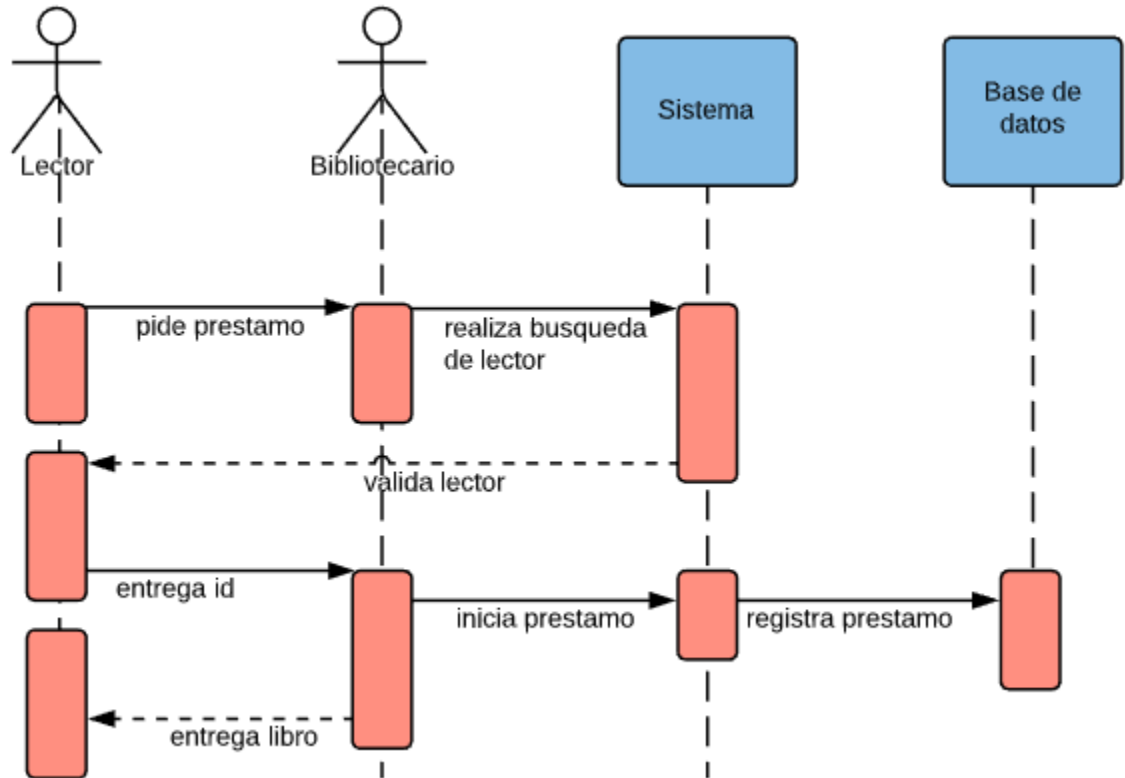
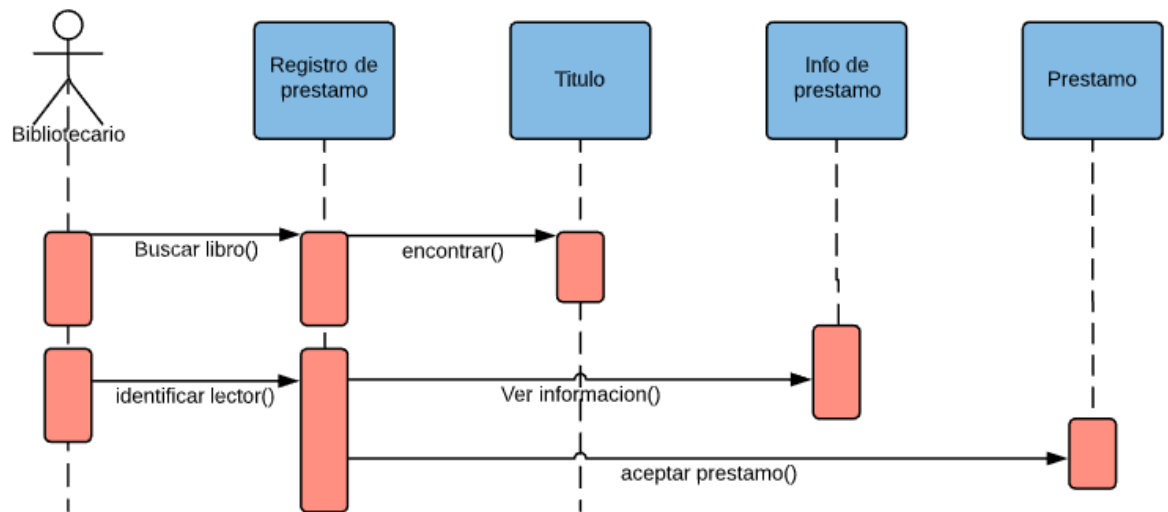


## 2.5. Diagramas de objetos

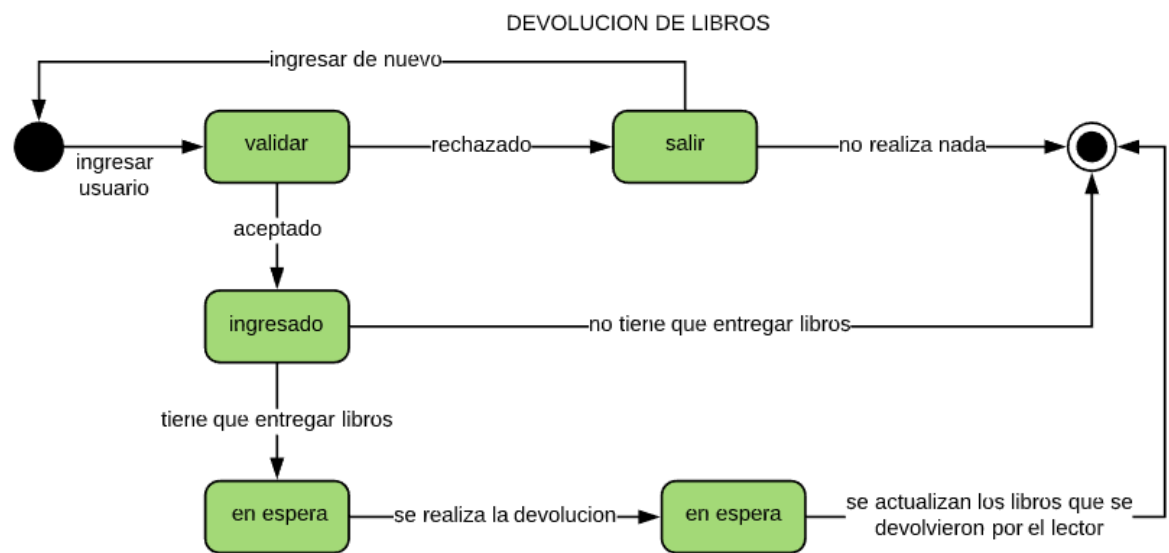
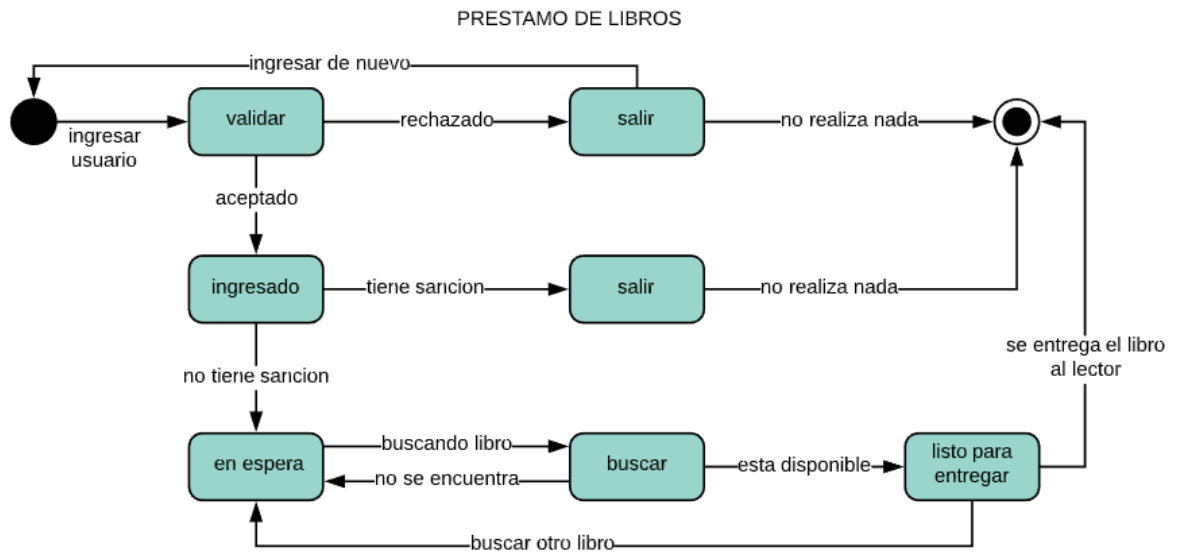


## 2.6. Diagramas de Secuencia





## 2.7. Diagramas de estados

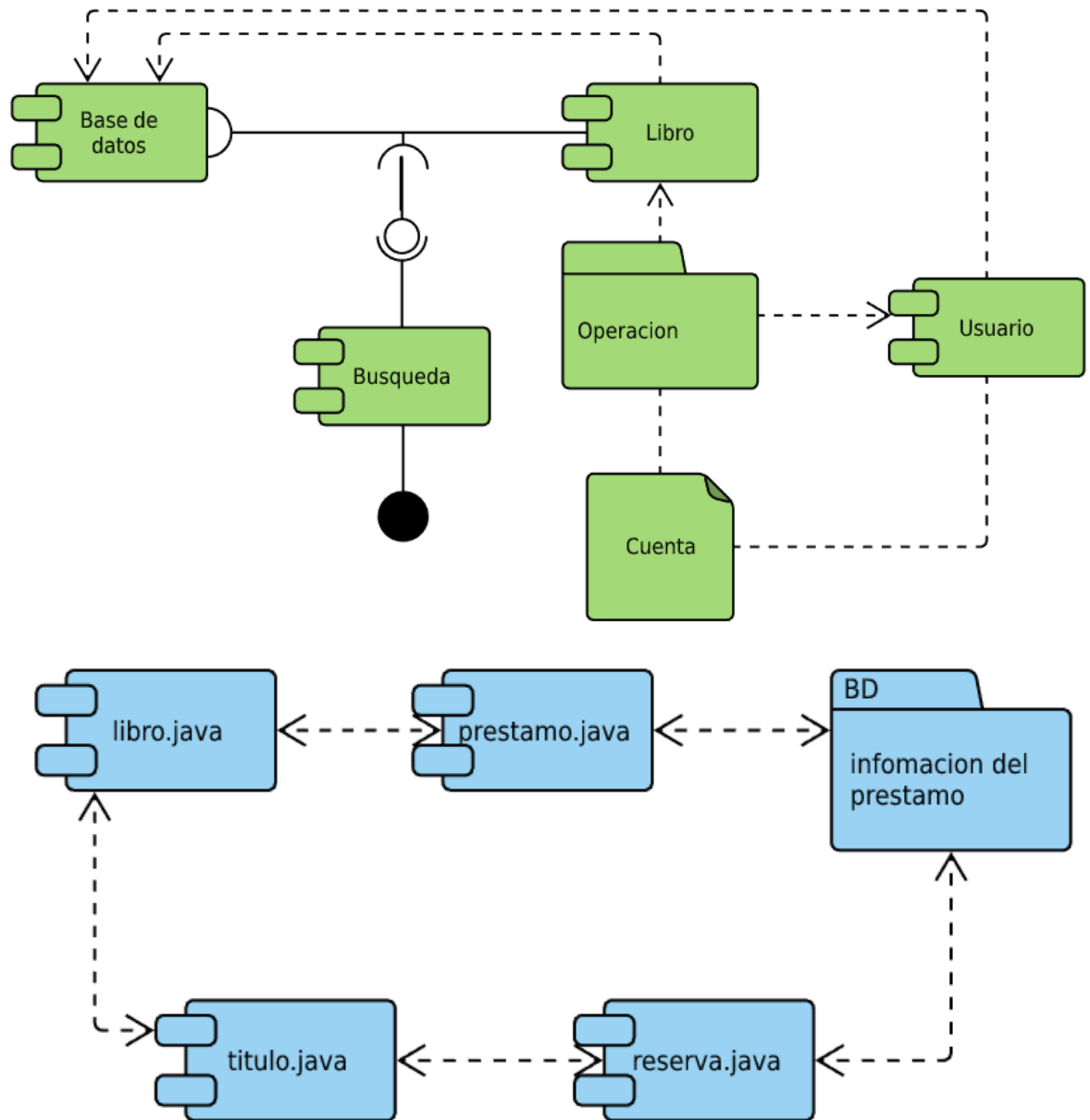


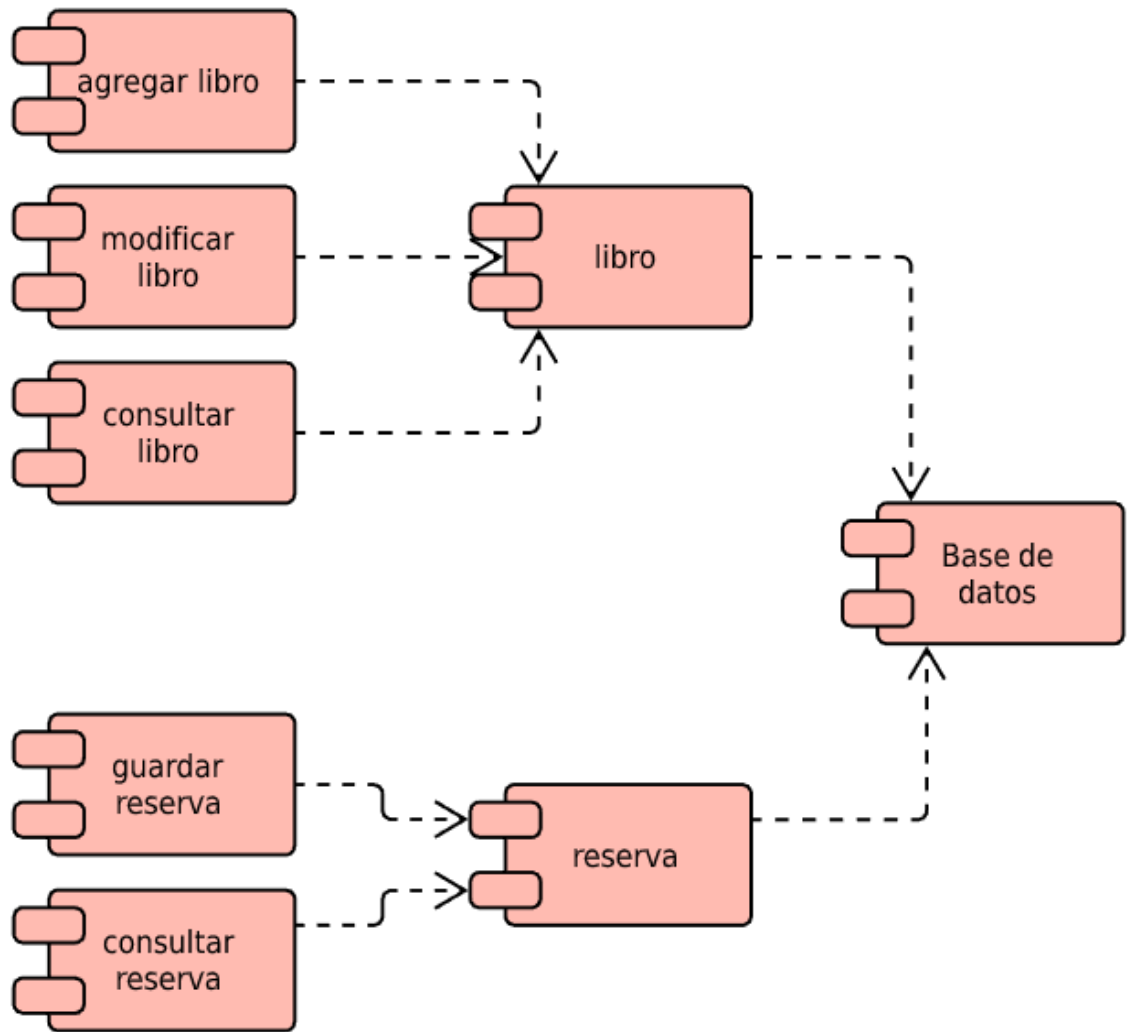
## 2.8. Diagramas de actividades

## 2.9. Diagramas de flujo de información

### 3. Diseño

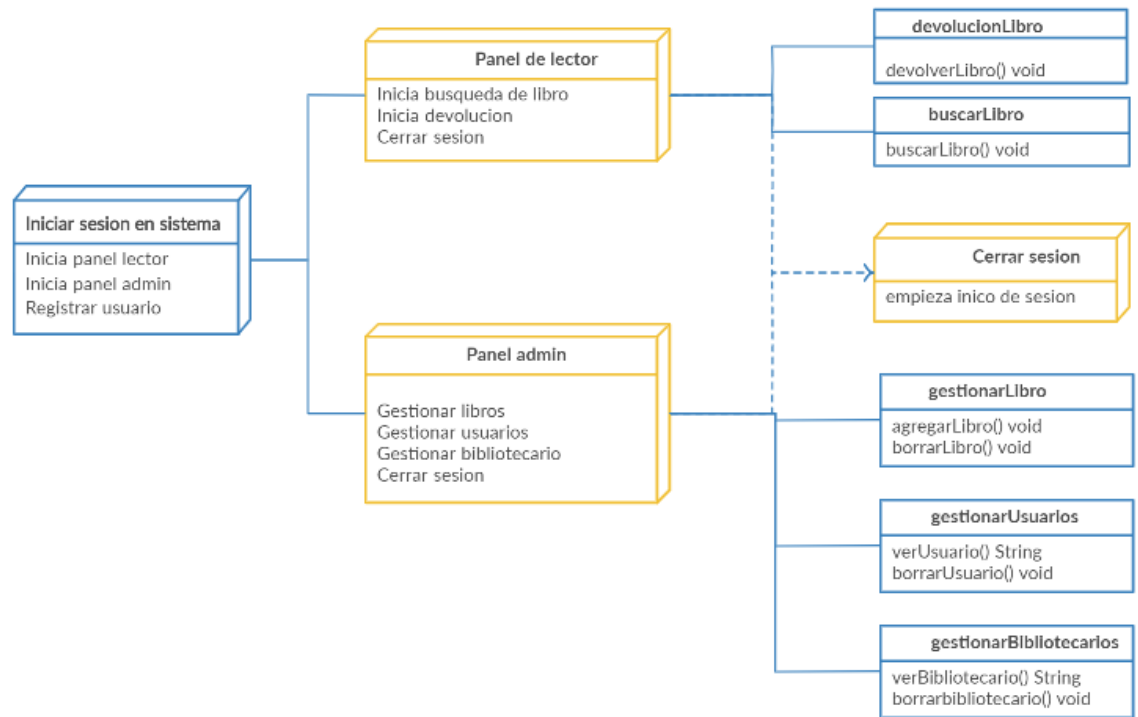
#### 3.1. Diagrama de componentes



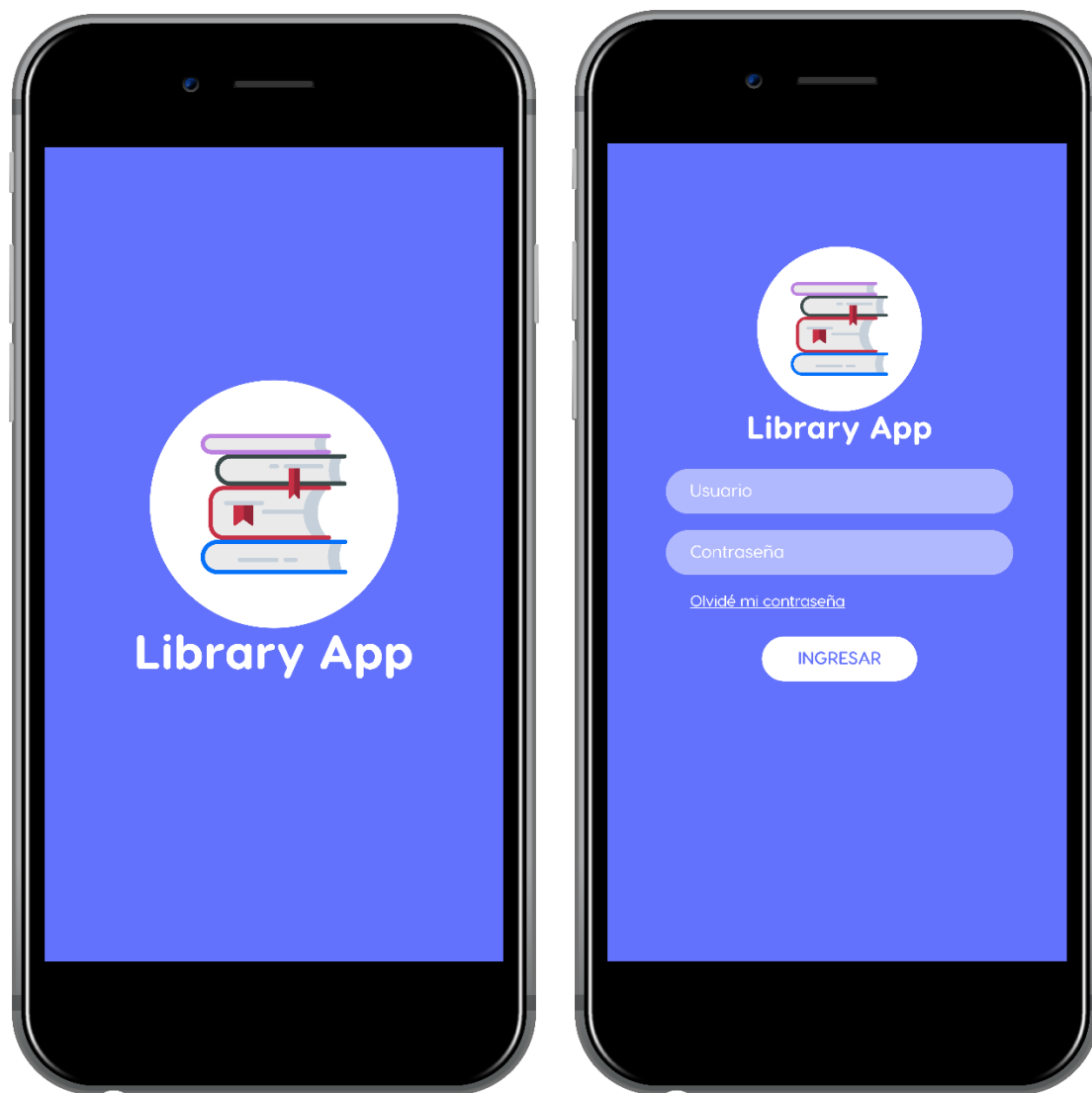


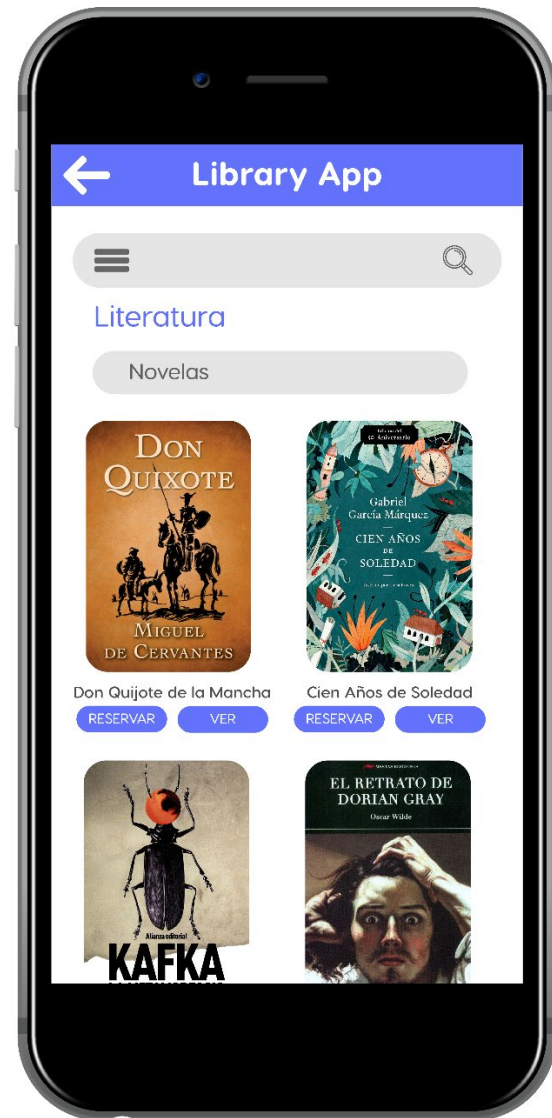
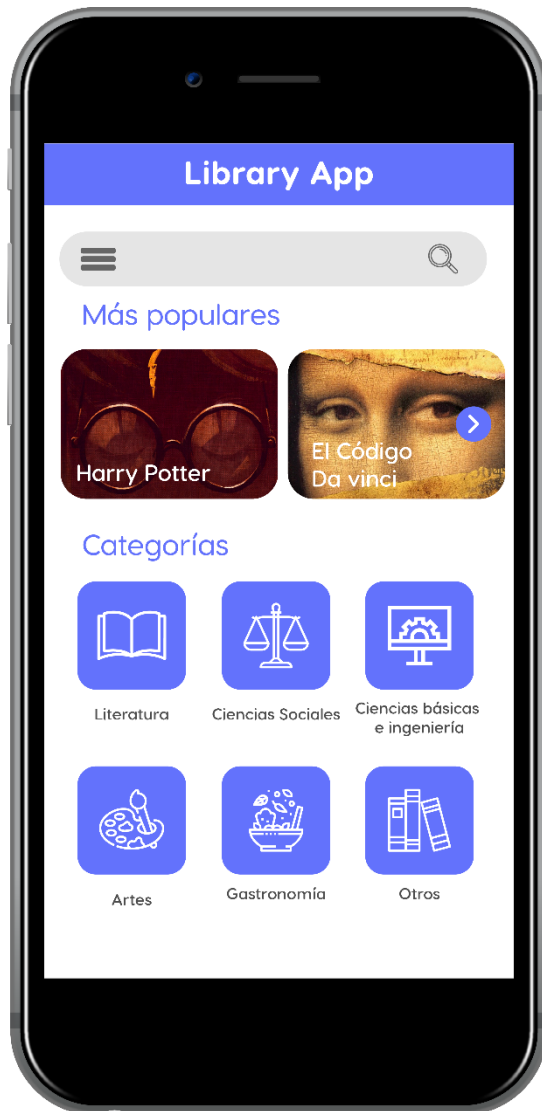


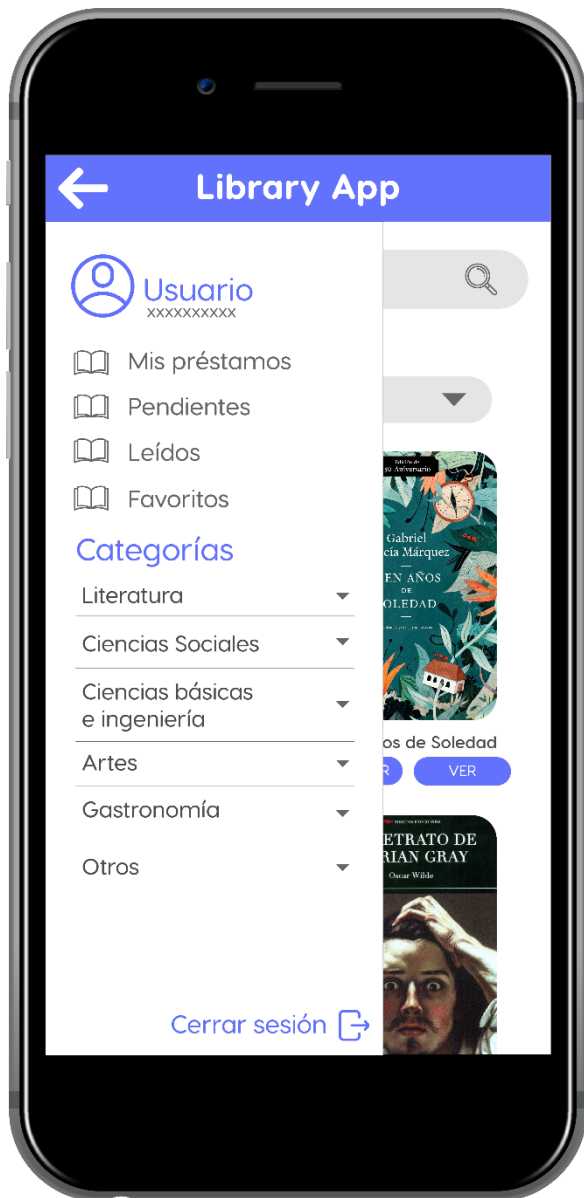
### 3.2. Diagrama de despliegue



### 3.3. Mockups







#### 4. Prototipo final

##### 4.1. Instructivo básico con imágenes y explicación

Este es el splash screen al iniciar la aplicación



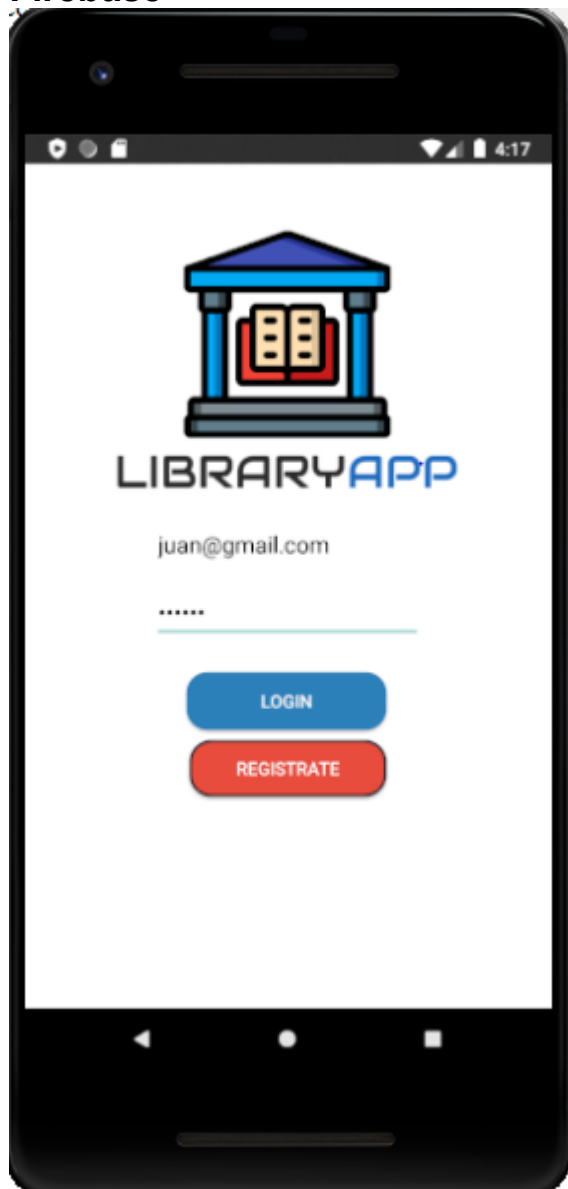
**Esta es la pantalla de registro de nuevos usuarios con correo y contraseña**



**Aquí esta la confirmación de Firebase con el nuevo usuario**

Identificador	Proveedores	Fecha de creación	Inicio de sesión	UID de usuario ↑
juan@gmail.com	✉	28 may. 2020	28 may. 2020	RzAhMQzq63XZVVw0Mlo02NAkn...

**En esta pantalla hacemos login para entrar a la aplicación donde pide correo y contraseña y lo verifica mediante Firebase**

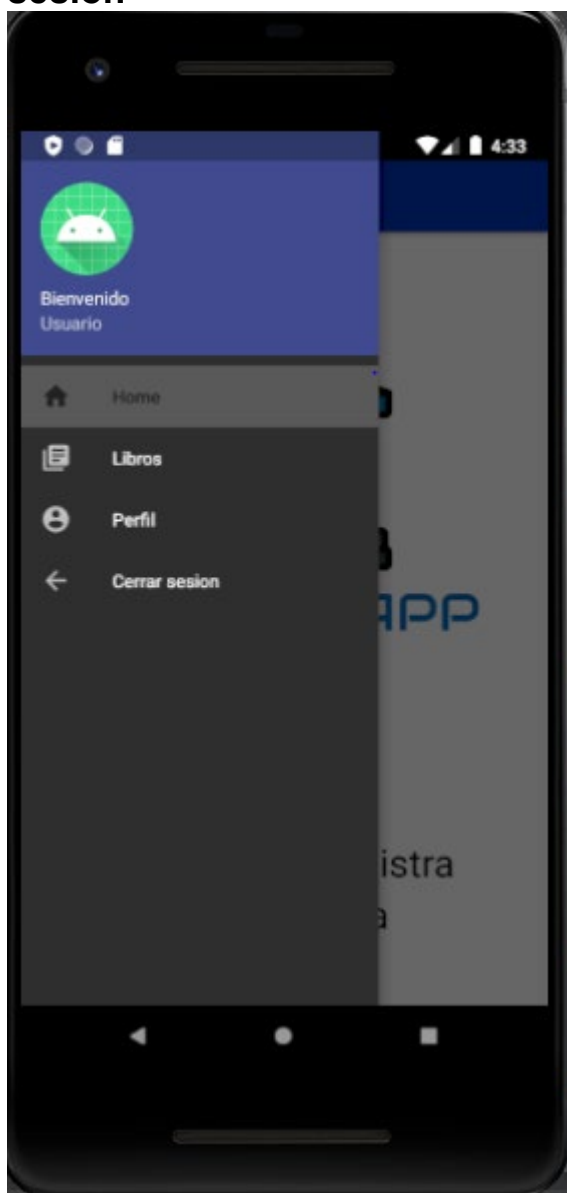


**Esta es la pantalla de inicio que le da la bienvenida al usuario**





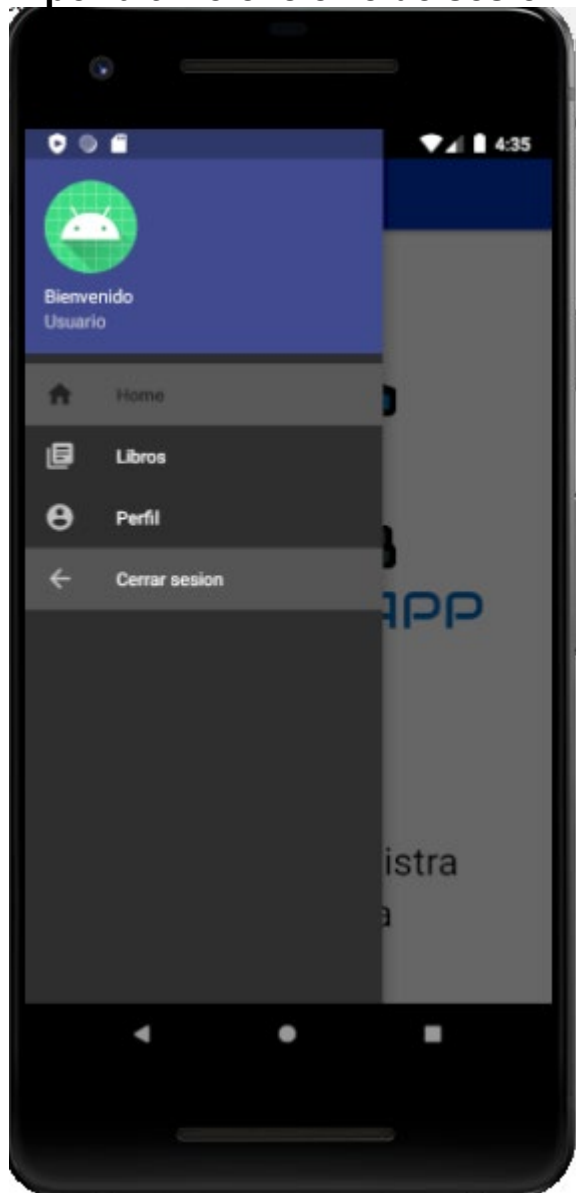
**Este es el menú donde encontraremos el home, los libros disponible, los datos del perfil y por ultimo el cierre de sesion**



Al escoger la opción libros nos lleva a esta pantalla donde encontraremos los libros disponible por la biblioteca para que el usuario pase a recogerlo



Y por ultimo el cierre de sesion



## **4.2. Documentación de pruebas**

Las pruebas que se realizaron fueron en el emulador del mismo Android Studio donde en cada cambio que hacia lo emulaba para saber si quedo bien, así como la implementación con la base de datos Firebase

## **4.3. Trabajo futuro o posibles mejoras**

# **5. Lecciones aprendidas**

## **5.1. Requerimientos**

Aprendí mucho de requerimientos, como los diferentes diagramas que realizamos a lo largo del semestre, los requerimientos funcionales y no funcionales que son de gran importancia a la hora de realizar un proyecto de un sistema de información

## **5.2. Desarrollo de software**

Me gustó mucho empezar a usar el programa Android Studio ya que este será uno de los programas más usados por desarrolladores de aplicaciones móviles, aunque es un poco pesado es una gran herramienta

## **5.3. Trabajo en grupo**

El trabajo en grupo es super importante ya que todos dan su punto de vista y mejoras del sistema a desarrollar. Como este proyecto lo hice solo si me faltó esa ayuda que necesita cada desarrollador como ideas nuevas, ajustes pertinentes y apoyo.

## 6. Actas

En este tercer corte me comprometí a entregar cada entrega y cada adelanto de mi proyecto de aula, en los que se incluye diagramas, requerimientos funcionales y no funcionales y también los prototipos de mi aplicación móvil. Por último subí cada archivo a mi repositorio en GitHub.

## 7. Repositorio git

<https://github.com/jubarragan2/App-biblioteca.git>