



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Actividad de Aprendizaje Contacto Docente N.º 2 Primer Parcial

Tema de la Actividad: Creación de Objetos y UML

Tipo de Actividad: Diseño y Modelado

Apellidos: Guapulema Pichamba

Nombres: Pamela Inés

NRC: 1322

Análisis y desarrollo de la Actividad

Diseñe 5 objetos diferentes con su correspondiente diagrama UML, asegurándose de mostrar las relaciones entre ellos.

Ejemplo:

- Libro: Representa los libros disponibles en una biblioteca, con atributos como título, autor, isbn, y métodos como prestar() y devolver().
- Usuario: Los usuarios pueden tener una lista de libros prestados. Se relaciona con Libro mediante composición.
- Biblioteca: Administra un catálogo de libros y se relaciona con Libro por asociación.
- Empleado: Gestiona usuarios y libros, y se relaciona con Usuario y Libro por asociación.
- Transacción: Registra préstamos y devoluciones. Se relaciona con Usuario y Libro por agregación.

Relación entre las clases para el diagrama UML

- Asociación. – A contiene al elemento B dentro de su definición.



- Herencia. – B ha heredado elementos de A, se interpreta que B es hijo de A.



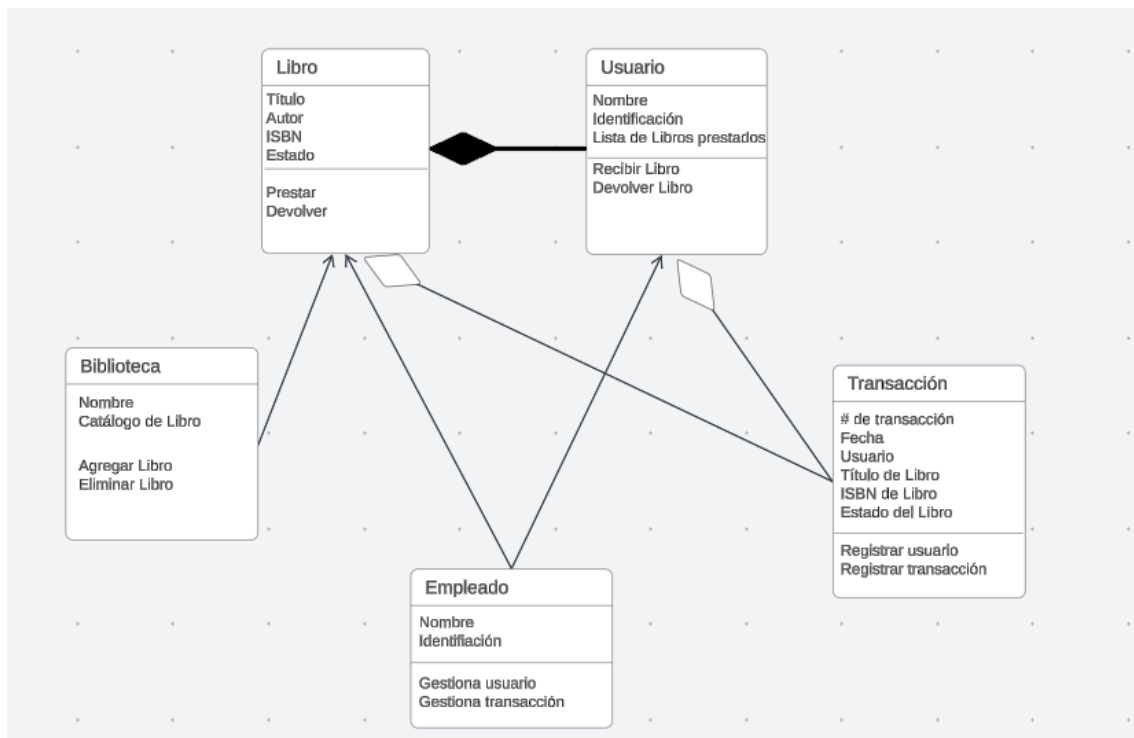
- Agregación. – A contiene varios elementos de B, se puede representar como una lista de datos.



- Composición. – A es totalmente dependiente de B, implica que si A no existe tampoco lo haría B.



Diagrama UML de relación entre clases

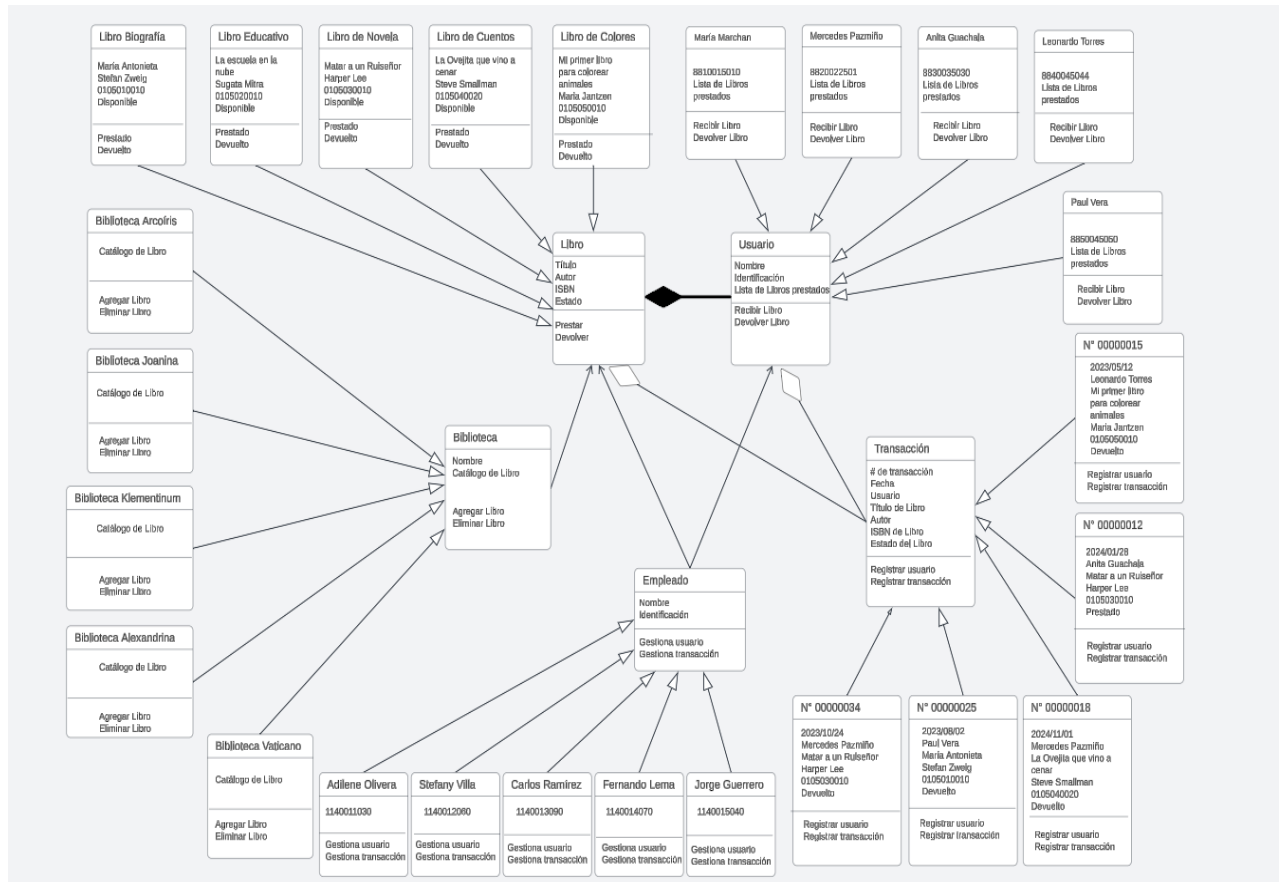


Dentro del diagrama se observa la relación entre clases, Clase Libro empieza siendo disponible en la Clase Biblioteca que tiene un catálogo de la Clase Libro para la Clase Usuario, donde puede pedir prestado o devolverlo, esto es la Clase Transacción que registra estas acciones por la Clase Empleado para la Clase Usuario.

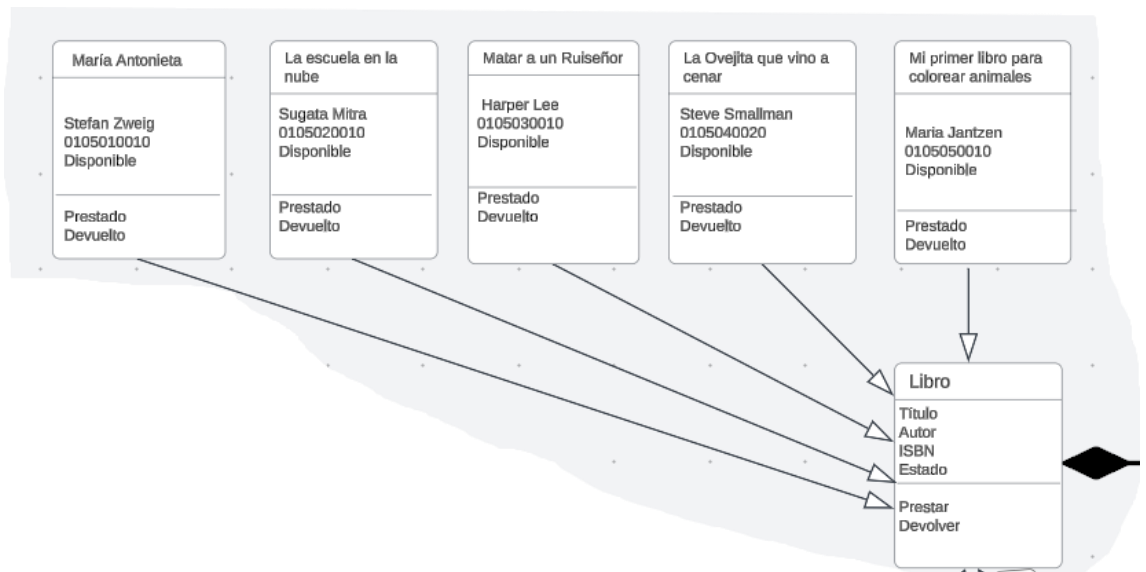
Usuario depende únicamente de Libro, Biblioteca contiene a Libro en su definición como también Empleado contiene a Libro y a Usuario, Transacción depende de varios elementos como Usuario y Libro.

Diagrama UML junto con los 5 objetos por Clase de Herencia

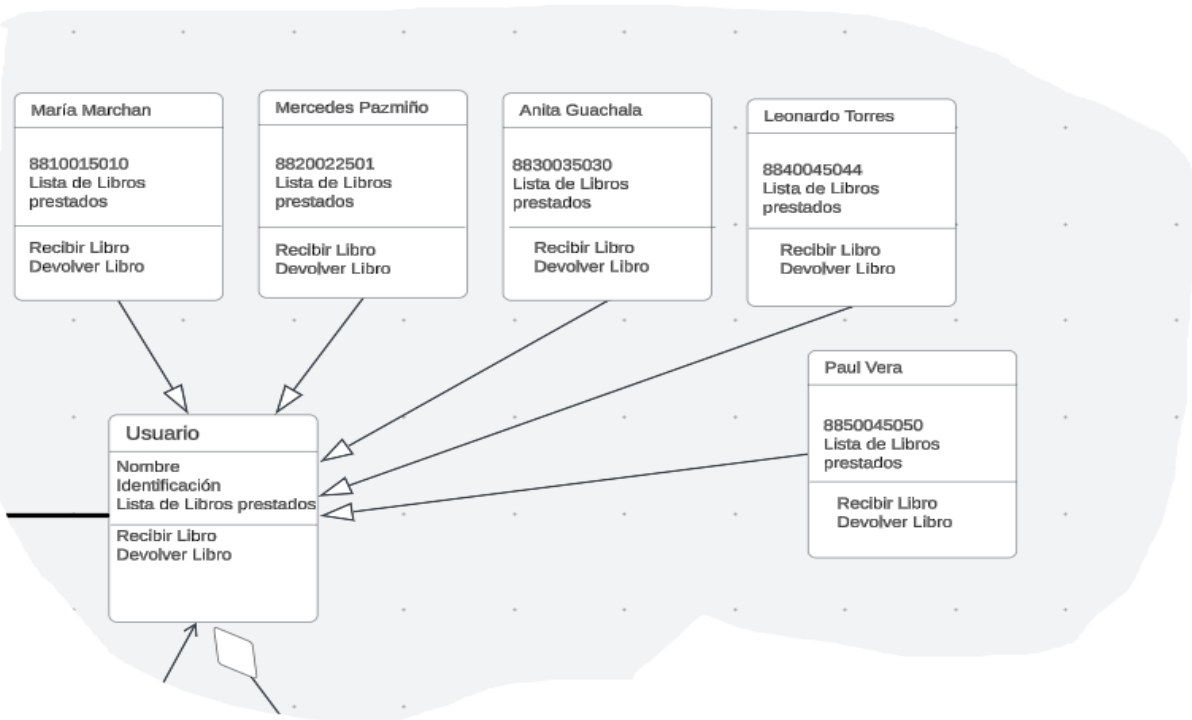
Enlace hacia el diagrama: https://lucid.app/lucidspark/8399b8a7-cc16-4f1d-b54c-a9972ad4a691/edit?view_items=NiNwTgMfeuWj&invitationId=inv_0fc96d41-ce5b-4d9a-8e3e-20fed8ddc109



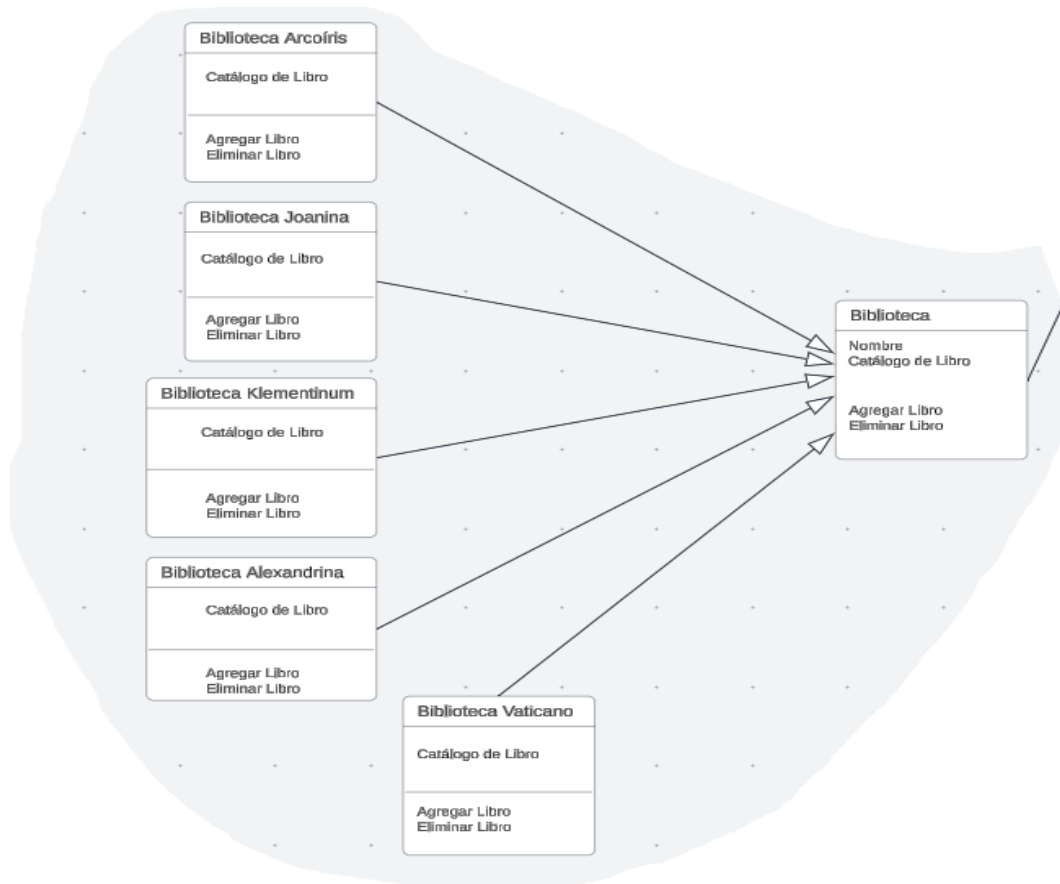
1. Clase Libro



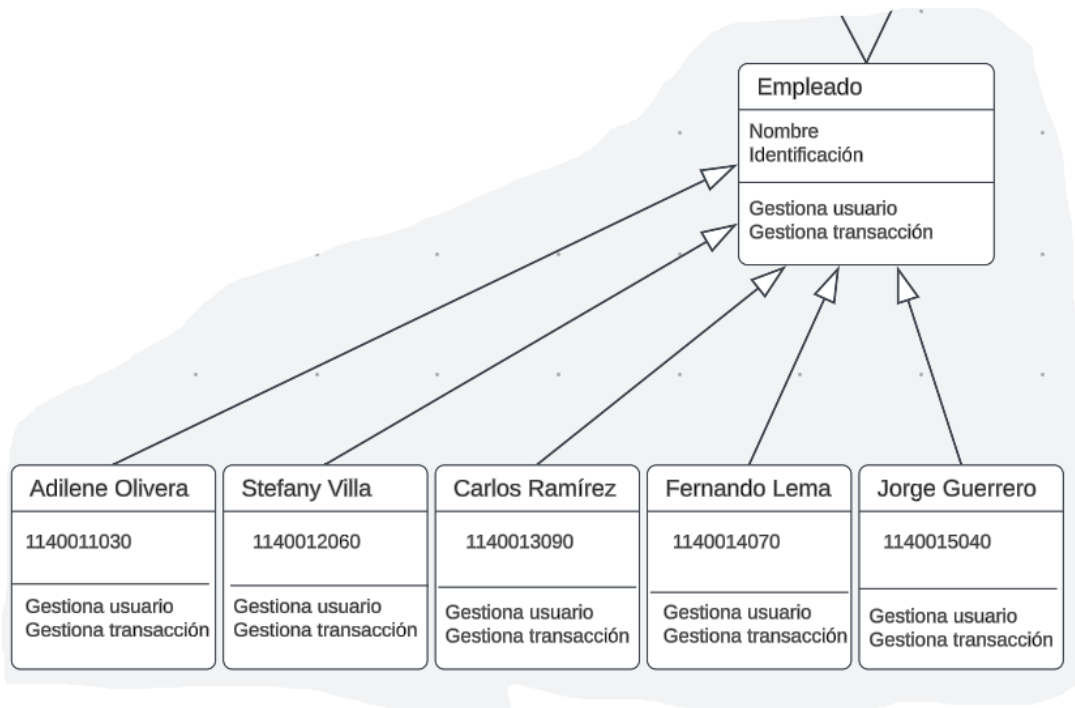
2. Clase Usuario



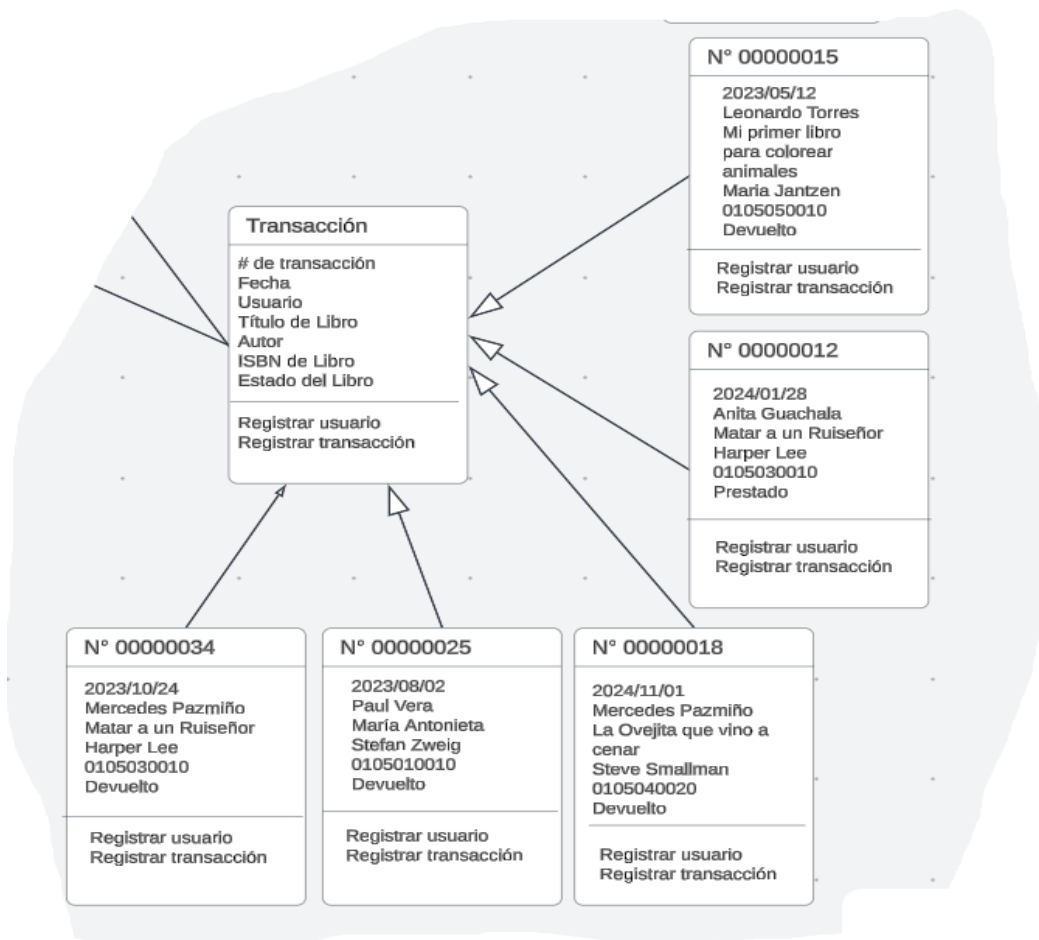
3. Clase Biblioteca



4. Clase Empleado



5. Clase Transacción



Implementación de Código

El diagrama previamente presentado se lo transfirió a código dentro de un simulador Java como por ejemplo online GDB.

Main.java

```
public class Main {  
    public static void main(String[] args) {  
        Libro libro1 = new Libro();  
        libro1.titulo = "Mi primer libro para colorear animales";  
        libro1.autor = "Maria Jantzen";  
        libro1.isbn = 0105050010;  
        libro1.estado = "disponible";  
        libro1.disponible();  
  
        Libro libro2 = new Libro();  
        libro2.titulo = "La Ovejita que vino a cenar";  
        libro2.autor = "Steve Smallman";  
        libro2.isbn = 0105040020;  
        libro2.estado = "disponible";  
        libro2.disponible();  
  
        Libro libro3 = new Libro();  
        libro3.titulo = "Matar a un Ruiseñor";  
        libro3.autor = "Harper Lee";  
        libro3.isbn = 0105030010;  
        libro3.estado = "disponible";  
        libro3.disponible();  
  
        Libro libro4 = new Libro();
```

```
libro4.titulo = "La escuela en la nube";  
libro4.autor = "Sugata Mitra";  
libro4.isbn = 0105020010;  
libro4.estado = "disponible";  
libro4.disponible();
```

```
Libro libro5 = new Libro();  
libro5.titulo = "María Antonieta";  
libro5.autor = "Stefan Zweig";  
libro5.isbn = 0105010010;  
libro5.estado = "disponible";  
libro5.disponible();
```

```
}
```

```
}
```

Libro.java

```
public class Libro { // NOMBRE DE LA CLASE
```

```
    String titulo;    // ATRIBUTO
```

```
    String autor;
```

```
    int isbn;
```

```
    String estado;
```

```
    public void disponible() { // METODO
```

```
        System.out.println("El libro " + titulo + " de " + autor + " con ISBN " + isbn + " su  
estado es " + estado);
```

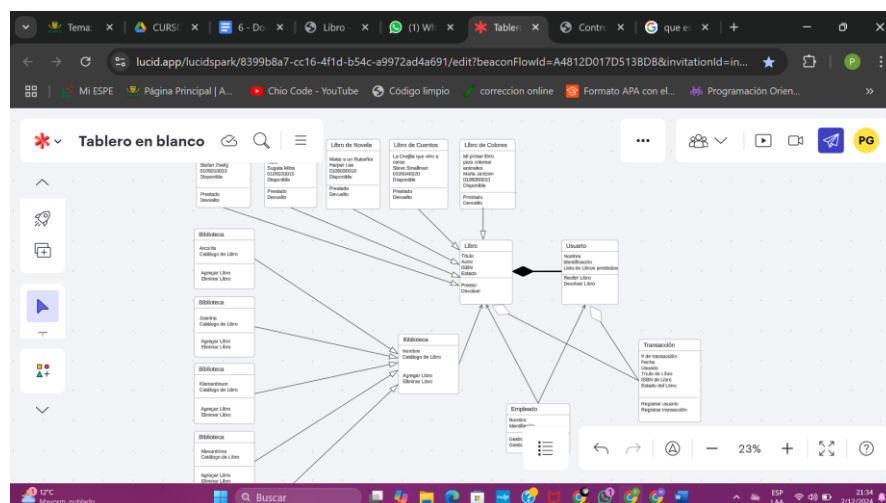
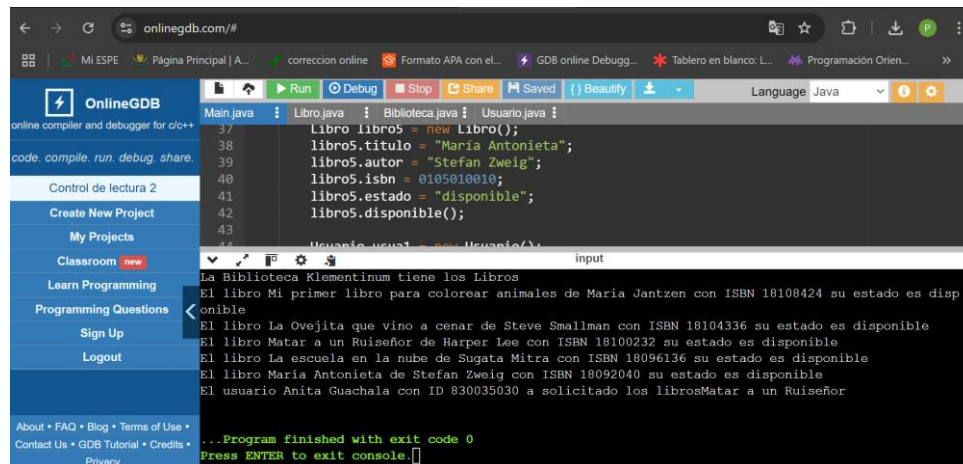
```
    }
```

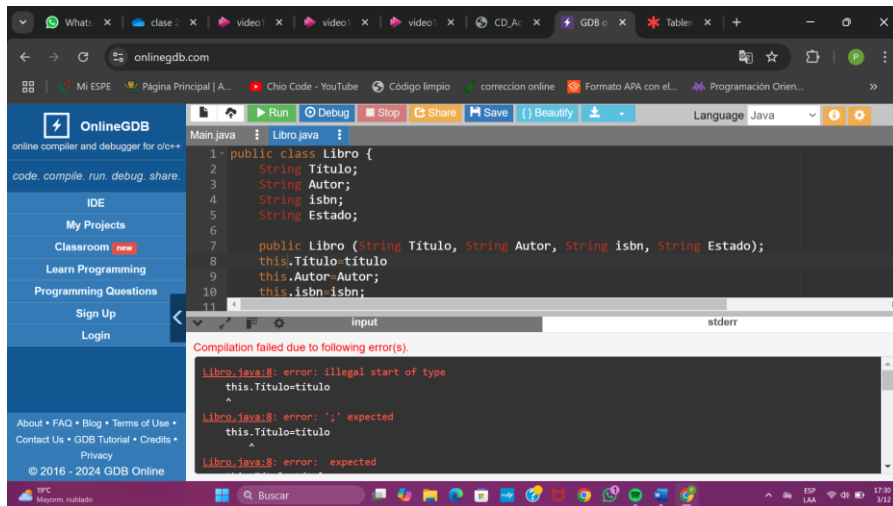
```
}
```

Evidencia de la consola de ejecución

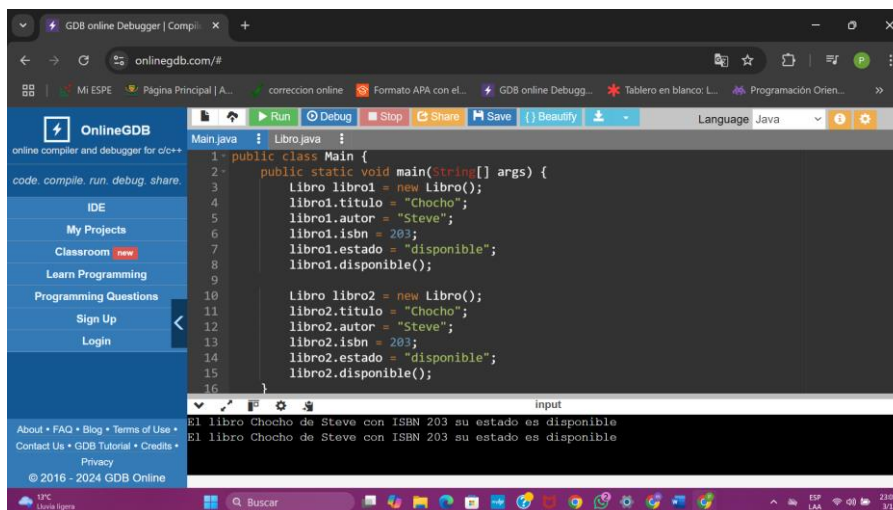
El libro	Mi primer libro para colorear animales de Maria Jantzen con ISBN 18108424	su estado es disponible
El libro	La Ovejita que vino a cenar de Steve Smallman con ISBN 18104336	su estado es disponible
El libro	Matar a un Ruiseñor de Harper Lee con ISBN 18100232	su estado es disponible
El libro	La escuela en la nube de Sugata Mitra con ISBN 18096136	su estado es disponible
El libro	María Antonieta de Stefan Zweig con ISBN 18092040	su estado es disponible

Evidencia de la realización de prueba y error





```
1 public class Libro {
2     String Titulo;
3     String Autor;
4     String isbn;
5     String Estado;
6
7     public Libro (String Titulo, String Autor, String isbn, String Estado);
8     this.Titulo=titulo;
9     this.Autor=Autor;
10    this.isbn=isbn;
11 }
```



```
1 public class Main {
2     public static void main(String[] args) {
3         Libro libro1 = new Libro();
4         libro1.titulo = "Chocho";
5         libro1.autor = "Steve";
6         libro1.isbn = 203;
7         libro1.estado = "disponible";
8         libro1.disponible();
9
10        Libro libro2 = new Libro();
11        libro2.titulo = "Chocho";
12        libro2.autor = "Steve";
13        libro2.isbn = 203;
14        libro2.estado = "disponible";
15        libro2.disponible();
16    }
```

Resumen de resolución

Para realizar la actividad, se analizaron los ejemplos propuestos: Libro, Usuario, Biblioteca, Empleado y Transacción, identificando las clases principales, sus atributos, métodos y relaciones. Esto permitió construir un modelo que representara de manera precisa la dinámica de una biblioteca, enfocándonos en acciones clave como la gestión de préstamos y devoluciones. Se adaptaron los ejemplos iniciales según las necesidades del sistema, asegurando que cada objeto cumpliera un rol específico y que las relaciones reflejaran interacciones reales.

El diseño del diagrama UML fue fundamental para visualizar de forma estructurada las conexiones entre las clases. Se utilizaron relaciones como asociación, composición y agregación para representar correctamente las interacciones. Por ejemplo, se mostró cómo un usuario puede tener una lista de libros prestados mediante composición, mientras que las transacciones registran préstamos y devoluciones utilizando agregación.

Este diagrama sirvió como base para la implementación en código, se realizaron pruebas para validar el correcto funcionamiento del sistema y confirmar que cumpliera con los requisitos establecidos.

Referencias:

W3Schools.com. (s. f.). https://www.w3schools.com/java/java_variables_multiple.asp
<https://docs.google.com/document/d/1CzDYZtQtsptWUmFEWqjhsRKXmPHsUwoz/edit>
https://docs.google.com/document/d/1S5DxQ3LoI5-NiaPyLS-m4m0Wms96GzASa-1hfJ2X_TI/edit?tab=t.0
https://drive.google.com/drive/folders/1UHU_nWzqZGK64fS2HvVLCi8v6O7M2UhP