	UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE		DEPARTAMENTO DE COMPUTACION
	CÓDIGO:		ESPE-DCEX-v1-2023-011

CONTROL DE LECTURA N° 2

DATOS INFORMATIVOS

Apellidos y Nombres: Montachana Aldana Fabricio Mateo

Fecha: 08-12-2024

NRC: 1323

Asignatura: Programación Orientada a Objetos

- **TEMA**

UML

- **Objetivo General:**

- Desarrollar un sistema de gestión de biblioteca eficiente que facilite el control del catálogo de libros, el registro de usuarios, y la administración de transacciones, optimizando los recursos y el tiempo de operación.

- **Objetivos Específicos:**

- **Implementar un sistema de administración de catálogo de libros**

Permitir el registro, consulta, actualización y eliminación de información de libros, incluyendo disponibilidad.

- **Gestionar usuarios y empleados de la biblioteca**

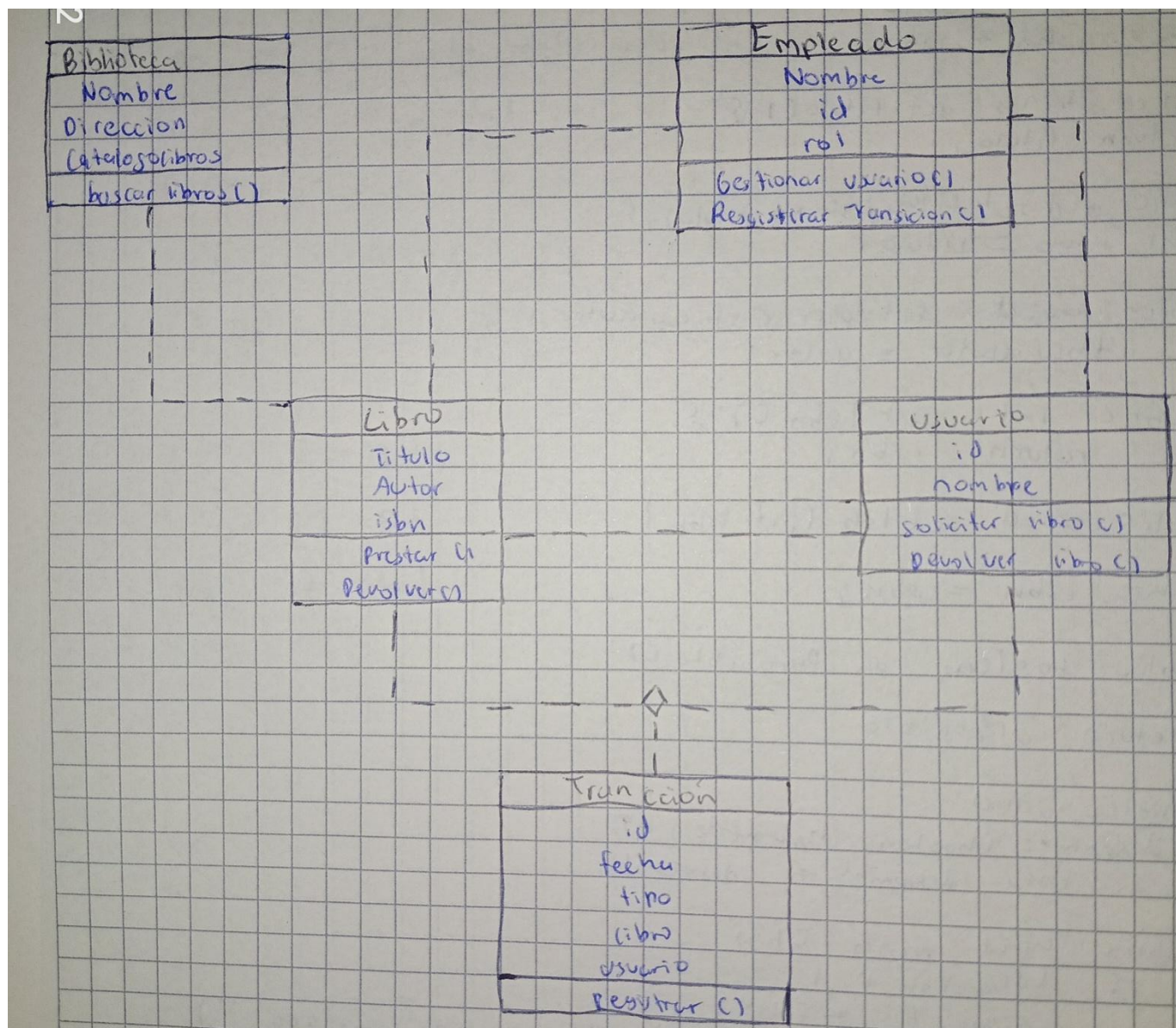
- Proveer funcionalidades para registrar usuarios y empleados, con permisos diferenciados según roles, y mantener un historial de interacciones.

	UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE		DEPARTAMENTO DE COMPUTACION	
	CÓDIGO:		ESPE-DCEX-v1-2023-011	

○ Automatizar el control de transacciones

Diseñar un sistema para registrar préstamos, devoluciones y reservas, asegurando un seguimiento efectivo de los libros en circulación.

1. DESARROLLO



Libro

```
public class Libro {
    private String titulo = "El señor de los anillos";
    private String autor = "J.R.R. Tolkien";
    private int isbn = 100;
    private boolean disponible = true;

    public Libro (String titulo,
    String autor, int isbn) {
        this.titulo = titulo;
        this.autor = autor;
        this.isbn = isbn;
    }

    public String get titulo () {
        return titulo;
    }

    public void set titulo (String titulo) {
        this.titulo = titulo;
    }

    public void set autor (String autor) {
        this.autor = autor;
    }

    public int get isbn () {
        return isbn;
    }

    public void set isbn (int isbn) {
        this.isbn = isbn;
    }

    public boolean is disponible () {
        return disponible;
    }

    public void
    set disponible (boolean disponible) {
        this.disponible = disponible;
    }

    public void prestar () {
        if (disponible) {
            disponible = false;
            System.out.println ("El libro ha sido prestado.");
        } else {
            System.out.println ("El libro no está disponible.");
        }
    }
}
```


Usuario

```
import java.util.*;
import java.util.*;

public class Usuario {
    private String nombre = "Fabricio Nieto";
    private String direccion = "Av. Amambas y Inagrito";
    private List<Libro> libros prestados = new ArrayList<>();

    public Usuario(String nombre, String direccion) {
        this.nombre = nombre;
    }

    public String getDireccion() {
        return direccion;
    }

    public void setDireccion(String direccion) {
        this.direccion = direccion;
    }

    public List<Libro> getLibrosPrestados() {
        return libros prestados;
    }

    public void eliminarLibro(Libro libro) {
        libros prestados.remove(libro);
    }
}
```

Biblioteca

```
import java.util.*;
import java.util.*;

public class Biblioteca {
    private String nombre = "Biblioteca Municipal";
    private List<Libro> catalogo = new ArrayList<>();

    public Biblioteca(String nombre) {
        this.nombre = nombre;
    }

    public String getNombre() {
        return nombre;
    }

    public List<Libro> getCatalogo() {
        return catalogo;
    }

    public void agregarLibro(Libro libro) {
        catalogo.add(libro);
    }

    public void eliminarLibro(Libro libro) {
        catalogo.remove(libro);
    }
}
```


Empleado

```

import java.util.ArrayList;
import java.util.List;

public class Empleado {
    private String nombre = "Ana Gomez";
    private String cargo = "Bibliotecaria";
    private List<Usuario> usuarios = new ArrayList<>();

    public Empleado (String nombre, String cargo) {
        this.nombre = nombre;
        this.cargo = cargo;
    }

    public String getNombre() {
        return nombre;
    }

    public void setNombre (String nombre) {
        this.nombre = nombre;
    }

    public String getCargo() {
        return cargo;
    }

    public void setCargo (String cargo) {
        this.cargo = cargo;
    }

    public List<Usuario>
    getUsuarios () {
        return usuarios;
    }

    public void
    agregar usuario (Usuario usuario)
    {
        usuarios.add (usuario);
    }

    public void
    eliminar Usuario (Usuario usuario)
    {
        usuarios.remove (usuario);
    }
}

```


Transaccion

import java.util.Date;

```
public class Transaccion {
    private Date fecha = new Date (2022,12,7);
    private Usuario usuario;
    private Libro libro;
```


```
    public Transaccion (Usuario usuario, Libro libro) {
        this.usuario = usuario;
        this.libro = libro;
    }
```

```
    public Date get fecha() {
        return fecha;
```

```
}
```

```
    public void set fecha (Date fecha) {
        this.fecha = fecha;
```

```
}
```

	UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE	DEPARTAMENTO DE COMPUTACION	
		CÓDIGO:	ESPE- DCEX-v1-2023- 011

Conclusiones:

- **Optimización de procesos**

La implementación del sistema de gestión de biblioteca facilita el manejo del catálogo, el registro de usuarios y el control de transacciones, reduciendo tiempos y minimizando errores administrativos.

- **Mejora en la experiencia de usuarios**

El sistema proporciona a los usuarios una experiencia más fluida y accesible, con herramientas como consulta de disponibilidad de libros y seguimiento de préstamos.

- **Eficiencia operativa**

Al automatizar las tareas relacionadas con libros y transacciones, los empleados pueden enfocarse en brindar un mejor servicio, incrementando la productividad de la biblioteca.