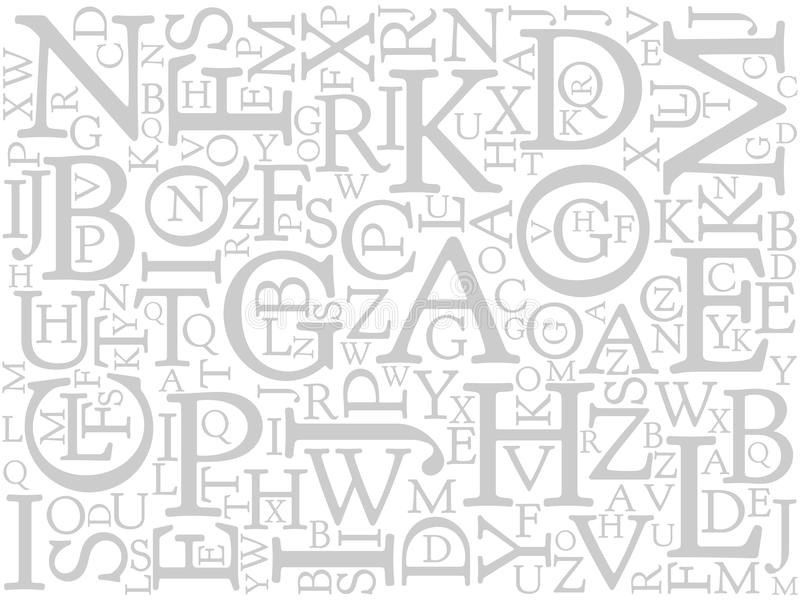
**Programación Avanzada**

**Trabajo integrador POO**





**Profesora:** Edith Schamberger

**Alumnos**: Franco Bravo, Marcos Alejandro Sena Ferreiro,

Costas Daniela Noemi, Federico Alexis Lambrech

**2023**

**15 de mayo**

**Biblioteca Virtual**

Contenido

[Análisis de la aplicación: 3](#_Toc134870354)

[Estrategia implementada: 3](#_Toc134870355)

[Decisiones de diseño establecidas: 3](#_Toc134870356)

[El diagrama de clases del programa sería el siguiente: 3](#_Toc134870357)

[Descripción de clases/objetos: 4](#_Toc134870358)

[Clase Usuario: 4](#_Toc134870359)

[Clase Libro: 5](#_Toc134870360)

[Clase Calificación: 5](#_Toc134870361)

[Clase Reserva: 5](#_Toc134870362)

[Anexo con el código fuente: 5](#_Toc134870363)

# Análisis de la aplicación:

El programa es una simulación de una biblioteca, donde se pueden reservar y prestar libros, devolverlos, cancelar reservas y calificar los libros.

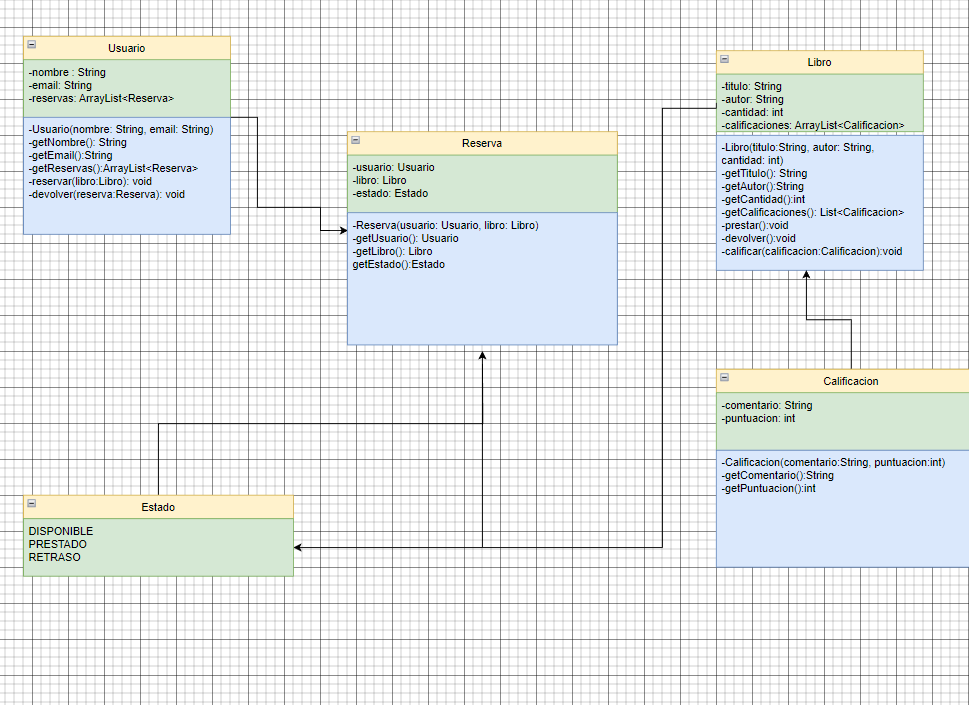
# Estrategia implementada:

Se uso el lenguaje de programación Java para desarrollar el programa y se ha hecho uso de clases y objetos para modelar la biblioteca, los usuarios y los libros.

# Decisiones de diseño establecidas:

Se utilizo enumeración Estado para representar el estado de los libros (disponible, prestado, retraso) y el método reservar de la clase Usuario para agregar una nueva reserva a su lista de reservas. También se ocupo la clase Reserva para representar la reserva de un libro por parte de un usuario y el método devolver de la clase Reserva para permitir la devolución del libro prestado.

# El diagrama de clases del programa sería el siguiente:



# Descripción de clases/objetos:

# Clase Usuario:

Representa a un usuario de la biblioteca. Tiene un nombre, un email y una lista de reservas. Esta clase contiene los métodos getNombre, getEmail, getReservas, reservar y devolver.

# Clase Libro:

Representa a un libro de la biblioteca. Tiene un título, un autor, una cantidad y una lista de calificaciones. Esta clase contiene los métodos getTitulo, getAutor, getCantidad, prestar, devolver y calificar.

## Clase Calificación:

Representa a una calificación que se ha dado a un libro. Tiene un comentario y una puntuación. Esta clase contiene los métodos getComentario y getPuntuacion.

## Clase Reserva:

Representa una reserva de un libro por parte de un usuario. Tiene un usuario, un libro y un estado (disponible, prestado, retraso). Esta clase contiene los métodos getUsuario, getLibro, getEstado, devolver, cancelar y calificar.

# Anexo con el código fuente:

import java.util.ArrayList;  
import java.util.List;  
  
// Enumeración que representa los posibles estados de un libro en la biblioteca  
enum Estado {  
 *DISPONIBLE*, *PRESTADO*, *RETRASO*}  
  
// Clase Usuario que representa a un usuario de la biblioteca  
class Usuario {  
 private final String nombre;  
 private final String email;  
 private final List<Reserva> reservas;  
  
 public Usuario(String nombre, String email) {  
 this.nombre = nombre;  
 this.email = email;  
 this.reservas = new ArrayList<>(); // Lista de reservas realizadas por el usuario  
 }  
  
 public String getNombre() {  
 return nombre;  
 }  
  
 public String getEmail() {  
 return email;  
 }  
  
 public List<Reserva> getReservas() {  
 return reservas;  
 }  
  
 public void reservar(Libro libro) {  
 Reserva reserva = new Reserva(this, libro); // Crear una nueva reserva asociada al usuario y al libro  
 reservas.add(reserva); // Agregar la reserva a la lista de reservas del usuario  
 }  
  
 public void devolver(Reserva reserva) {  
 reserva.devolver(); // Llamar al método devolver de la reserva  
 reservas.remove(reserva); // Eliminar la reserva de la lista de reservas del usuario  
 }  
}  
  
// Clase Libro que representa un libro en la biblioteca  
class Libro {  
 private final String titulo;  
 private final String autor;  
 private int cantidad;  
 private final List<Calificacion> calificaciones;  
  
 public Libro(String titulo, String autor, int cantidad) {  
 this.titulo = titulo;  
 this.autor = autor;  
 this.cantidad = cantidad;  
 this.calificaciones = new ArrayList<>(); // Lista de calificaciones realizadas para el libro  
 }  
  
 public String getTitulo() {  
 return titulo;  
 }  
  
 public String getAutor() {  
 return autor;  
 }  
  
 public int getCantidad() {  
 return cantidad;  
 }  
  
 public List<Calificacion> getCalificaciones() {  
 return calificaciones;  
 }  
  
 public void prestar() {  
 if (cantidad > 0) {  
 cantidad--; // Decrementar la cantidad de libros disponibles al prestar uno  
 }  
 }  
  
 public void devolver() {  
 cantidad++; // Incrementar la cantidad de libros disponibles al devolver uno  
 }  
  
 public void calificar(Calificacion calificacion) {  
 calificaciones.add(calificacion); // Agregar la calificación a la lista de calificaciones del libro  
 }  
}  
  
// Clase Calificacion que representa una calificación realizada por un usuario para un libro  
class Calificacion {  
 private final String comentario;  
 private final int puntuacion;  
  
 public Calificacion(String comentario, int puntuacion) {  
 this.comentario = comentario;  
 this.puntuacion = puntuacion;  
 }  
  
 public String getComentario() {  
 return comentario;  
 }  
  
 public int getPuntuacion() {  
 return puntuacion;  
 }  
}  
  
// Clase Reserva que representa una reserva realizada por un usuario para un libro  
class Reserva {  
 private final Usuario usuario;  
 private final Libro libro;  
 private Estado estado;  
  
 public Reserva(Usuario usuario, Libro libro) {  
 this.usuario = usuario;  
 this.libro = libro;  
 this.estado = Estado.*PRESTADO*;  
 libro.prestar(); // Llamar al método prestar del libro al realizar la reserva  
 }  
}