

Informe de las prácticas de experimentación y aplicación de los aprendizajes

1. Datos Informativos:

Facultad:	<i>CIENCIAS ADMINISTRATIVAS GESTIÓN EMPRESARIAL E INFORMÁTICA</i>
Carrera:	<i>SOFTWARE</i>
Asignatura:	<i>PROGRAMACION ORIENTADA A OBJETOS</i>
Ciclo:	<i>2</i>
Docente:	<i>ING. MÓNICA BONILLA</i>
Título de la práctica:	<i>INTEGRACIÓN DE OBJETOS MEDIANTE APLICACIONES DE TEXTO</i>
No. de práctica:	<i>3</i>
Escenario o ambiente de aprendizaje de la practica	<i>En casa</i>
No. de horas:	<i>4</i>
Fecha:	<i>16/12/2024</i>
Estudiantes:	<i>Ariel Alejandro Calderón</i>
Calificación	

2. Introducción:

La práctica realizada tiene como propósito entender la implementación y gestión de objetos en el lenguaje Java, aplicando conceptos fundamentales de la POO.

3. Objetivo de la práctica:

Realizar la integración de objetos en la programación orientada a objetos.

4. Descripción del desarrollo de la práctica:

Enunciado del problema: Se requiere de un sistema para abordar los siguientes problemas clave:

- Es necesario mantener un registro estructurado de los libros disponibles, incluyendo detalles como título, autor, ISBN, año de publicación y estado (disponible o prestado). La falta de automatización en esta tarea puede causar pérdidas o inconsistencias en el inventario.
- Las bibliotecas necesitan registrar y gestionar información de usuarios (nombre, ID único, contacto, dirección), garantizando un acceso rápido y confiable a estos datos para atender solicitudes de préstamos y consultas.
- Es fundamental llevar un seguimiento eficiente de los préstamos activos, identificando la relación entre un libro prestado, el usuario asociado y el estado del préstamo (activo o finalizado). La falta de un control centralizado puede derivar en disputas por devoluciones tardías o libros extraviados.
- Tanto los usuarios como los administradores necesitan herramientas rápidas para consultar libros por ISBN y usuarios por ID único. Esto garantiza agilidad en los procesos de búsqueda dentro de la biblioteca.
- Los administradores requieren una interfaz para realizar operaciones clave, como agregar libros al inventario, registrar nuevos usuarios y supervisar el flujo de préstamos y devoluciones.

Diagrama de Casos de Uso - Biblioteca

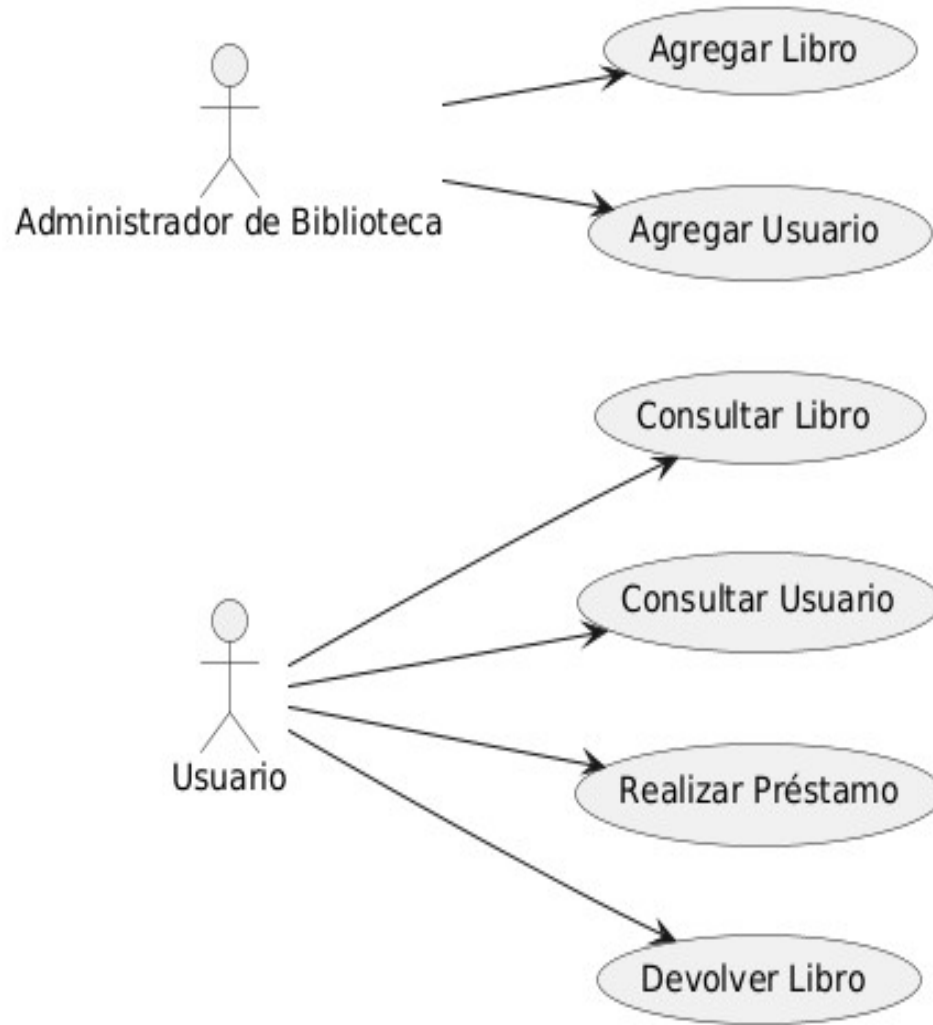
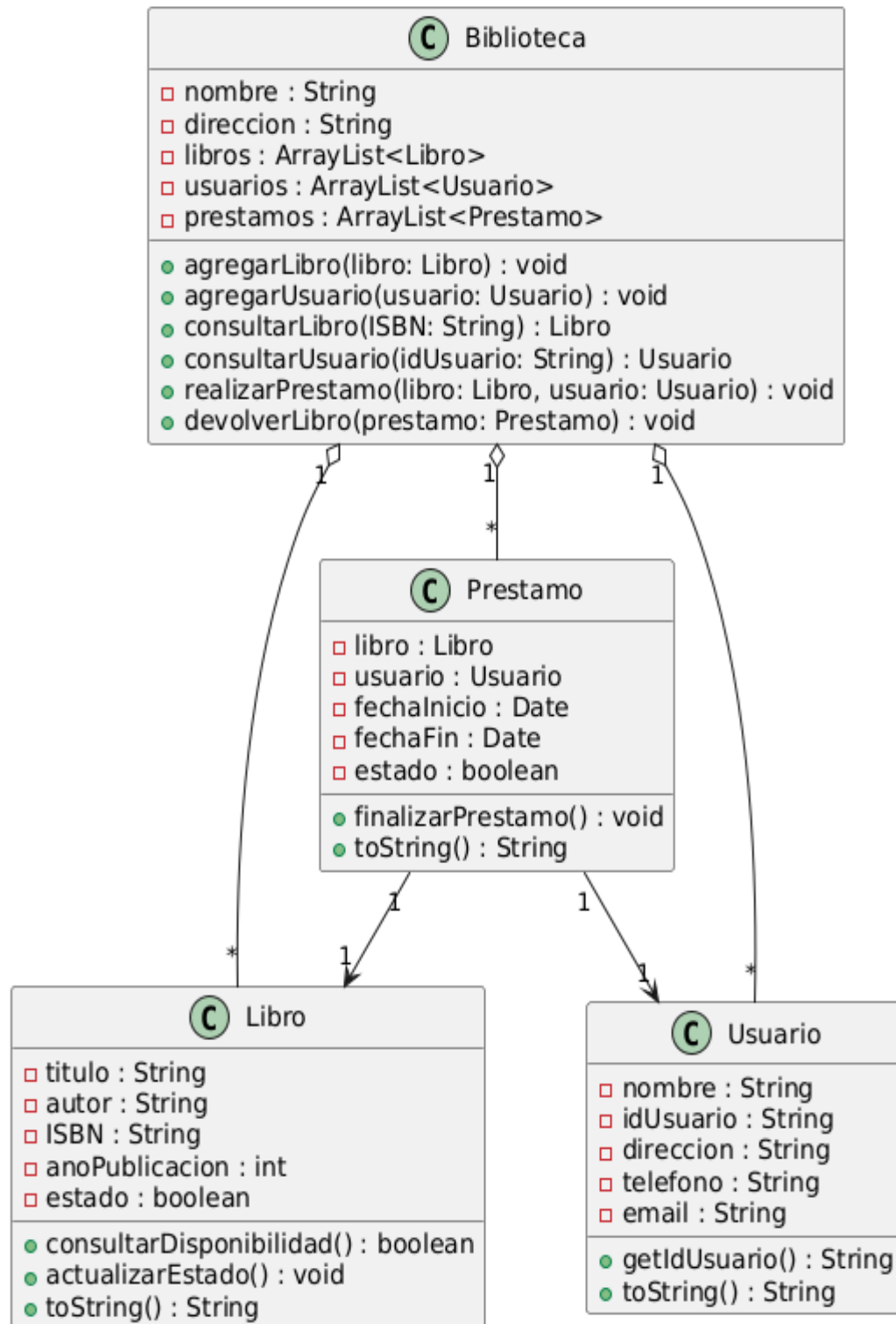


Diagrama de Clases - Biblioteca



5. Metodología:

La metodología seguida fue de aprendizaje activo, donde se ejemplificó ejemplos de diagrams UML para la gestión de objetos en programación orientada a objetos.

6. Resultados obtenidos:

Ejemplificar la integración de objetos mediante aplicaciones de texto.

7. Conclusiones:

La implementación de objetos en programación orientada a objetos es una habilidad fundamental para desarrollar sistemas. A través de esta práctica se aprendió a cómo diagramar sistemas en código usando la conversión de UML.

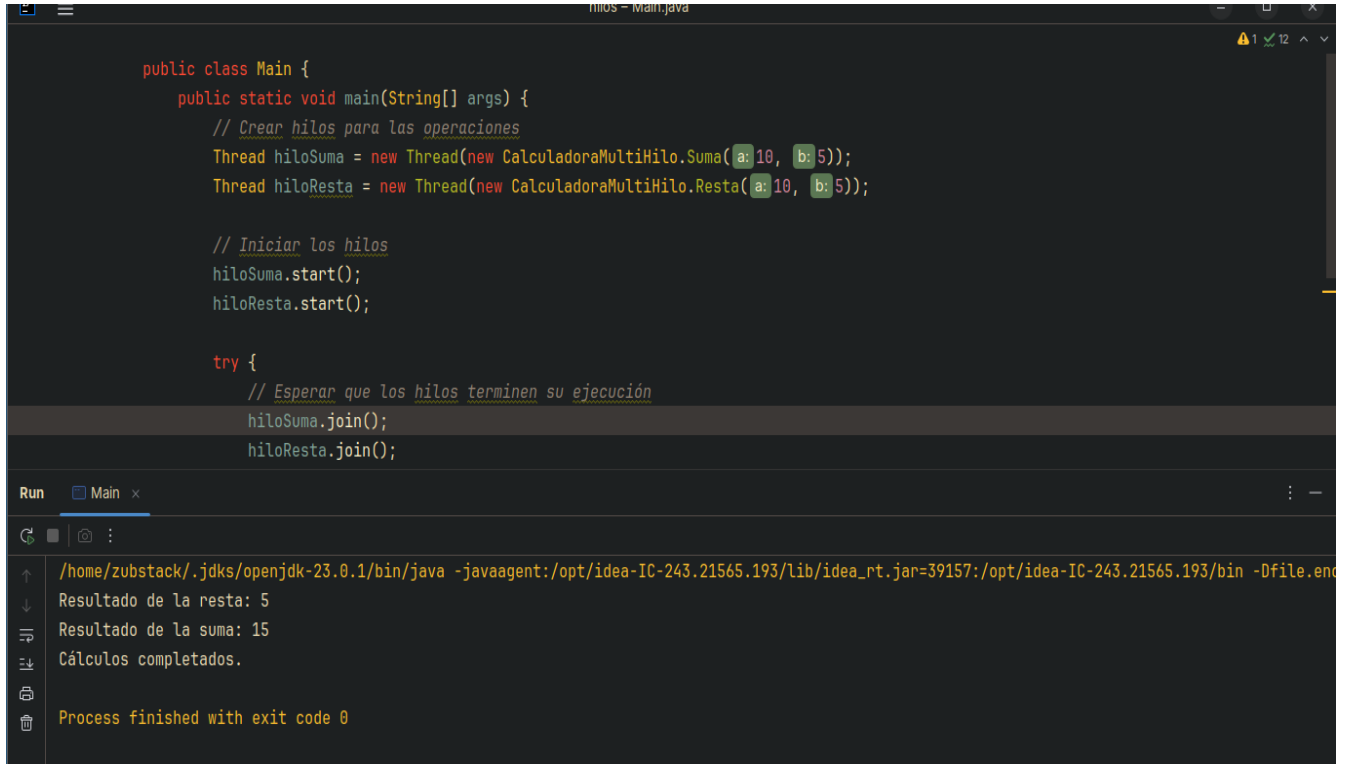
8. Recomendaciones:

- Se recomienda trabajar paso a paso para convertir código fuente de algún sistema en una representación más visual. Para ello, se empieza primero por conseguir una idea clara de cada uno de los objetos y funcionalidades que participan en el sistema y sus relaciones.
- Se aconseja usar herramientas de software para plasmar diagramas UML como son Lucid chart o Plant UML.

9. Bibliografía:

- Eckel, B. (2006). *Thinking in Java* (4th ed.). Pearson Education.
- Bloch, J. (2008). *Effective Java* (2nd ed.). Addison-Wesley.
- Oracle Corporation. (2024). *Concurrency in Java*
<https://docs.oracle.com/javase/tutorial/essential/concurrency/>.

10.Anexos:



```
public class Main {
    public static void main(String[] args) {
        // Crear hilos para las operaciones
        Thread hiloSuma = new Thread(new CalculadoraMultiHilo.Suma(a: 10, b: 5));
        Thread hiloResta = new Thread(new CalculadoraMultiHilo.Resta(a: 10, b: 5));

        // Iniciar los hilos
        hiloSuma.start();
        hiloResta.start();

        try {
            // Esperar que los hilos terminen su ejecución
            hiloSuma.join();
            hiloResta.join();
        }
    }
}
```

Run Main x

/home/zubstack/.jdk/openjdk-23.0.1/bin/java -javaagent:/opt/idea-IC-243.21565.193/lib/idea_rt.jar=39157:/opt/idea-IC-243.21565.193/bin -Dfile.encoding=UTF-8

Resultado de la resta: 5
Resultado de la suma: 15
Cálculos completados.

Process finished with exit code 0

Anexo 1: Ejecucion de codigo ejemplo