

UNIVERSIDAD AUTÓNOMA GABRIEL RENÉ MORENO

FACULTAD DE INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN Y

TELECOMUNICACIONES

CARRERA DE INGENIERIA EN REDES Y TELECOMUNICACIONES



"SISTEMA DE GESTIÓN DE PRESTAMOS DE LIBROS EN LA FICCT "

Proyecto : 66

Área : Informática

Categoría : Avanzado

Materia : Estructura de Datos I

Docente : Ing. Msc. Mario Campos Becerra

Integrantes :

- Copa Umacacho Gabriel
- Lizarazu Caballero Hamilton
- Mendez Roman Alex Yamil
- Mendía Parada Enrique
- Mendoza Pacsi Jose Carlos

Contenido

1. Contexto y Antecedentes del Problema	4
2. Planteamiento del Problema	5
3. Objetivos del Proyecto	6
3.1 Objetivo General.....	6
3.2 Objetivos Específico	6
4. Desarrollo o Propuesta Solución	7
4.1 Arquitectura General del Sistema.....	7
4.2 Módulos Funcionales del Sistema.....	8
4.3 Flujo de Funcionamiento	8
4.4 Tecnologías y Estructuras Utilizadas.....	9
4.5 Beneficios del Sistema.....	9
4.6 Resultados Esperados	10
5. Marco Teórico y Metodológico.....	14
6. Conclusiones y Recomendaciones	15
Bibliografía	17

Índice de Figuras

Figura 1 <i>Estructura de paquetes y módulos principales del sistema.</i>	11
Figura 2 <i>Proceso de registro de un nuevo usuario (Alumno).</i>	12
Figura 3 <i>Confirmación de Registro Exitoso y asignación de código.</i>	12
Figura 4 <i>Vista principal del usuario mostrando el menú de opciones y el listado de libros.</i>	13
Figura 5 <i>Confirmación de un préstamo registrado exitosamente.</i>	13
Figura 6 <i>Módulo de visualización y gestión de préstamos del usuario.</i>	14

1. Contexto y Antecedentes del Problema

La Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones (FICCT) de la Universidad Autónoma Gabriel René Moreno, ubicada en Santa Cruz de la Sierra, cumple un papel fundamental en la formación de profesionales del área tecnológica.

Dentro de sus servicios académicos, la biblioteca constituye un pilar esencial para el aprendizaje y la investigación, ya que proporciona acceso a textos, manuales, guías y materiales especializados que contribuyen al desarrollo académico de estudiantes y docentes.

Con el crecimiento constante de la población estudiantil y del cuerpo docente, la demanda de material bibliográfico ha aumentado significativamente. Este incremento ha evidenciado la necesidad de contar con un sistema de gestión más moderno y eficiente que agilice los procesos de préstamo, devolución y control de libros. Actualmente, muchos de estos procedimientos se realizan de forma manual o con herramientas informales, como hojas de cálculo o cuadernos, lo que genera demoras, errores de registro y atención limitada.

En los últimos años, la biblioteca de la FICCT ha enfrentado dificultades administrativas y técnicas en la gestión de sus recursos bibliográficos. El registro de préstamos y devoluciones se realiza, en muchos casos, mediante planillas físicas o archivos simples, ocasionando pérdida de información, errores en los registros, falta de control sobre los ejemplares prestados y escasa capacidad para generar reportes confiables.

La ausencia de un sistema digital centralizado también impide conocer en tiempo real la disponibilidad de libros, el historial de préstamos por usuario y los títulos más solicitados. Esto ralentiza el trabajo del personal encargado y afecta la calidad del servicio a la comunidad universitaria.

Por estas razones, se hace necesaria la implementación de un Sistema Automatizado de Préstamos de Libros que permita optimizar los procesos de registro, control y seguimiento de los materiales disponibles. Este sistema ofrecerá un servicio más rápido, ordenado y confiable, beneficiando directamente a estudiantes, docentes y personal administrativo.

Además, el proyecto tiene un enfoque formativo y tecnológico, ya que se desarrolla utilizando herramientas aprendidas en la carrera, como Java, el entorno de desarrollo NetBeans y la arquitectura MVC (Modelo–Vista–Controlador). Esto demuestra la aplicación práctica de los

conocimientos adquiridos en materias como *Estructura de Datos I* y *Programación Orientada a Objetos*, promoviendo el desarrollo de soluciones reales dentro del entorno universitario.

En resumen, este proyecto busca modernizar la gestión bibliotecaria de la FICCT, mejorando el control del inventario, reduciendo los errores y tiempos de atención, y garantizando un servicio eficiente y transparente que esté a la altura de las necesidades académicas actuales.

2. Planteamiento del Problema

En la biblioteca de la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones (FICCT), el proceso de préstamo y devolución de libros se realiza principalmente de forma manual o mediante registros dispersos en hojas de cálculo.

Esta situación ha generado diversas dificultades en la gestión del inventario bibliográfico, afectando directamente la eficiencia del servicio y la experiencia de los usuarios.

Actualmente, el control de préstamos, devoluciones y disponibilidad de ejemplares depende de la verificación manual de planillas o archivos simples. Esto provoca demoras en la atención, errores de registro y pérdida de información, especialmente en los periodos de mayor demanda estudiantil.

La falta de un sistema automatizado también impide que el personal bibliotecario conozca en tiempo real cuántos ejemplares se encuentran disponibles o prestados, lo que complica la toma de decisiones sobre compras, mantenimiento o reposición de material.

Por otro lado, la ausencia de reportes confiables limita la posibilidad de realizar un seguimiento estadístico del uso de los libros. No es posible, por ejemplo, identificar fácilmente cuáles son los títulos más solicitados o qué usuarios mantienen préstamos vencidos. Esto no solo representa un problema operativo, sino también una barrera para la gestión académica, ya que los datos del inventario no están integrados ni actualizados de forma automática.

El aumento constante del número de estudiantes en la FICCT ha puesto aún más en evidencia estas limitaciones. En la actualidad, los procesos manuales no logran sostener la demanda diaria de préstamos, y los registros físicos tienden a volverse inexactos con el tiempo.

Como consecuencia, se generan confusiones, demoras y pérdida de control sobre los ejemplares, lo que repercute negativamente en el acceso a la información y el aprovechamiento de los recursos bibliográficos.

Frente a esta problemática, se plantea el desarrollo de un Sistema de Préstamos de Libros para la Biblioteca de la FICCT, capaz de automatizar todas las operaciones relacionadas con el registro de usuarios, control de inventario, préstamos, devoluciones y generación de reportes.

El sistema busca centralizar la información en una base de datos unificada, ofreciendo a los bibliotecarios herramientas para realizar consultas rápidas, generar estadísticas precisas y evitar errores humanos. Asimismo, los estudiantes y docentes contarán con un proceso de atención más ágil, transparente y confiable.

Este proyecto no solo responde a una necesidad operativa, sino que también promueve la modernización tecnológica de la facultad, fomentando la digitalización de procesos internos mediante el uso de Java, NetBeans y el patrón MVC, alineados con la formación práctica de los estudiantes de Ingeniería en Sistemas y Redes.

En definitiva, el problema a resolver se resume en la falta de un sistema informático automatizado que permita una gestión eficiente, segura y confiable del préstamo de libros en la FICCT, mejorando la calidad del servicio bibliotecario y el acceso oportuno a la información.

3. Objetivos del Proyecto

3.1 Objetivo General

Desarrollar e implementar un Sistema Automatizado de Préstamos de Libros para la biblioteca de la FICCT (UAGRM), que optimice el registro, control y seguimiento de los préstamos y devoluciones, reduciendo errores, mejorando la atención a los usuarios y fortaleciendo la gestión de los recursos bibliográficos mediante herramientas tecnológicas modernas.

3.2 Objetivos Específico

- Analizar los procesos actuales de préstamo y devolución de libros en la biblioteca de la FICCT, identificando las limitaciones y fallas del sistema manual utilizado.

- Diseñar la arquitectura del sistema bajo el modelo MVC (Modelo–Vista–Controlador), estructurando los módulos y las relaciones entre los componentes principales (usuarios, libros, préstamos y reportes).
- Implementar el sistema utilizando el lenguaje Java en el entorno NetBeans, aplicando principios de programación orientada a objetos y estructuras de datos vistas en la materia.
- Integrar y probar la persistencia de datos mediante archivos serializados o base de datos local, garantizando la confiabilidad, disponibilidad y exactitud de la información.
- Validar y documentar el funcionamiento del sistema mediante pruebas de uso reales, generando evidencias y reportes que demuestren la mejora en los tiempos de atención y en la organización del inventario.

4. Desarrollo o Propuesta Solución

El proyecto “Sistema de Préstamos de Libros para la FICCT” fue desarrollado en el lenguaje Java utilizando el entorno Apache NetBeans IDE 17.

Su principal objetivo es automatizar el control de préstamos, devoluciones y reportes en la biblioteca de la FICCT, mejorando la rapidez, la organización y la confiabilidad de la información.

El sistema se diseñó aplicando la arquitectura MVC (Modelo–Vista–Controlador), lo que permite separar la lógica del negocio, la interfaz de usuario y la gestión de datos, facilitando su mantenimiento y escalabilidad.

4.1 Arquitectura General del Sistema

El sistema está dividido en tres capas principales:

- **Modelo:** Representa la lógica del negocio y los datos. Se encuentra en el paquete biblioteca. Negocio e incluye clases como Alumno, Docente, Bibliotecario, Libro, Préstamo, Nodo y diversas listas (ListaAlumnos, ListaLibros, ListaPrestamo, etc.).

Estas clases definen los atributos y métodos necesarios para manipular la información de los usuarios, los libros y los préstamos.

- **Vista:** Corresponde a la interfaz gráfica del sistema, desarrollada con Java Swing.

Se encuentra en el paquete biblioteca.Presentacion e incluye ventanas como Login, PanelBibliotecario, PanelUsuario, Registro, RegistroLibro, VentanaPrestamo, VistaPrestamosUsuario, VistaEditarPerfil, entre otras.

- **Controlador:** Coordina la comunicación entre las capas de modelo y vista. Gestiona los eventos que ocurren al interactuar con las ventanas, como registrar un préstamo, devolver un libro o generar un reporte.

Además, se implementa la persistencia de datos mediante la serialización de objetos, lo que permite guardar la información en archivos binarios y recuperarla automáticamente al reiniciar el sistema.

4.2 Módulos Funcionales del Sistema

- **Gestión de usuarios:** permite registrar, modificar o eliminar datos de estudiantes, docentes y bibliotecarios, guardando su información y su historial de préstamos.
- **Gestión de libros:** registra el material bibliográfico con sus datos (título, autor, editorial, cantidad de ejemplares) y controla su disponibilidad en tiempo real.
- **Préstamos y devoluciones:** realiza las operaciones de préstamo, devolución y renovación, actualizando automáticamente el inventario y el estado de los libros.
- **Reservas:** administra la cola de espera de los usuarios cuando no hay ejemplares disponibles, asignando automáticamente el siguiente usuario una vez que se devuelve un libro.
- **Reportes:** genera listados y estadísticas sobre libros más solicitados, usuarios con mora, préstamos por periodo académico y control de inventario.

4.3 Flujo de Funcionamiento

- El bibliotecario inicia sesión desde la ventana Login.
- Accede al panel principal (PanelBibliotecario) donde puede registrar usuarios o libros.
- Desde VentanaPrestamo, el sistema permite prestar, devolver o renovar ejemplares, validando disponibilidad y fechas.
- Los datos se guardan mediante serialización y pueden consultarse posteriormente en VistaPrestamosUsuario o en los reportes.

- Finalmente, los reportes se generan automáticamente, permitiendo un control total del inventario y del historial de préstamos.

4.4 Tecnologías y Estructuras Utilizadas

El sistema fue desarrollado utilizando el lenguaje Java dentro del entorno Apache NetBeans 17, haciendo uso de la biblioteca Java Swing para el diseño de la interfaz gráfica. Se aplicó el patrón de arquitectura MVC (Modelo–Vista–Controlador), lo que permitió dividir claramente la lógica del negocio, la interfaz de usuario y la gestión de datos. Gracias a esta estructura, el sistema mantiene una organización clara, facilita la localización de errores y permite realizar futuras modificaciones sin alterar el funcionamiento general.

La persistencia de la información se logró mediante el proceso de serialización de objetos, lo cual garantiza que los datos registrados (usuarios, libros y préstamos) se guarden de manera segura y se recuperen automáticamente cada vez que el sistema se reinicia. Además, se emplearon estructuras de datos como listas, colas y mapas para manejar los diferentes elementos del sistema. Las listas permiten mantener colecciones de objetos como libros o préstamos, las colas administran las reservas por orden de solicitud, y los mapas asocian de forma eficiente los usuarios con sus préstamos activos.

4.5 Beneficios del Sistema

El sistema ofrece múltiples ventajas frente al proceso manual. En primer lugar, reduce significativamente el tiempo que toma registrar un préstamo o devolución, permitiendo atender a más usuarios en menos tiempo. También minimiza los errores comunes de duplicación o pérdida de información, ya que todas las operaciones se realizan de forma automática y validada. Además, proporciona información en tiempo real sobre la disponibilidad de los libros, evitando confusiones y mejorando la organización del inventario.

Otra de las fortalezas del sistema es la posibilidad de generar reportes automáticos con solo unos clics, lo que facilita la toma de decisiones dentro de la biblioteca. De esta manera, los bibliotecarios pueden conocer los títulos más solicitados, los usuarios con mora y los periodos de mayor demanda. Finalmente, el sistema consolida toda la información en una sola plataforma, mejorando el control administrativo y abriendo la posibilidad de futuras ampliaciones, como la conexión con bases de datos externas o el envío de notificaciones automáticas a los usuarios.

4.6 Resultados Esperados

Con la implementación de este sistema, se espera una mejora notable en la eficiencia y calidad del servicio de la biblioteca. El tiempo promedio de atención por préstamo se reducirá de varios minutos a menos de uno, los errores de registro serán prácticamente eliminados, y la generación de reportes será inmediata y precisa. Además, el bibliotecario podrá acceder a la información del inventario de manera rápida y confiable, manteniendo un control actualizado de los ejemplares disponibles.

En conjunto, estos resultados contribuirán a una gestión bibliotecaria más moderna, transparente y alineada con las necesidades actuales de la comunidad universitaria. El proyecto no solo representa un avance tecnológico para la FICCT, sino también un aporte práctico al proceso formativo de los estudiantes que participaron en su desarrollo, demostrando la aplicación real de los conocimientos adquiridos en programación, estructuras de datos y diseño de sistemas.

Figura 1

Estructura de paquetes y módulos principales del sistema.

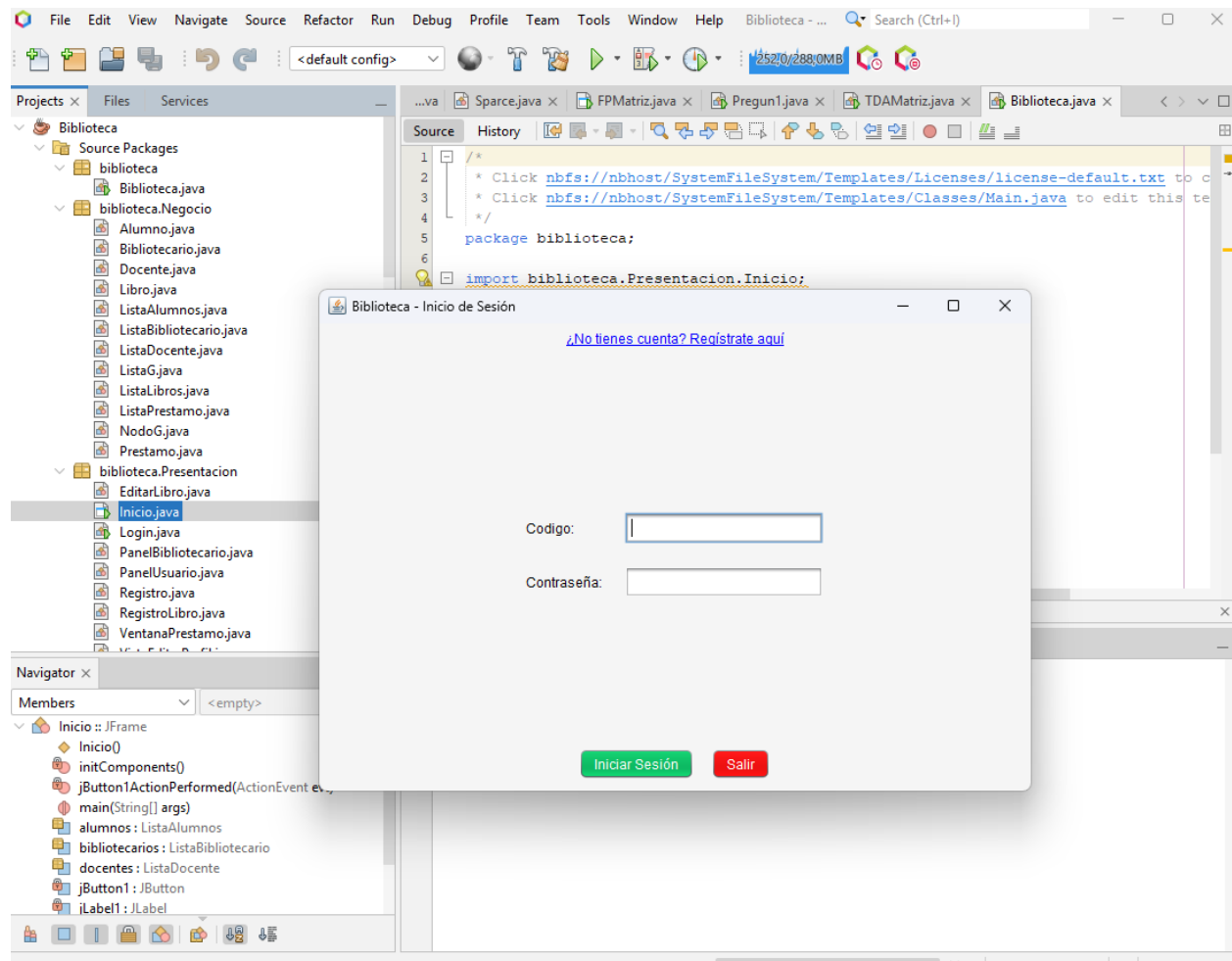


Figura 2

Proceso de registro de un nuevo usuario (Alumno).

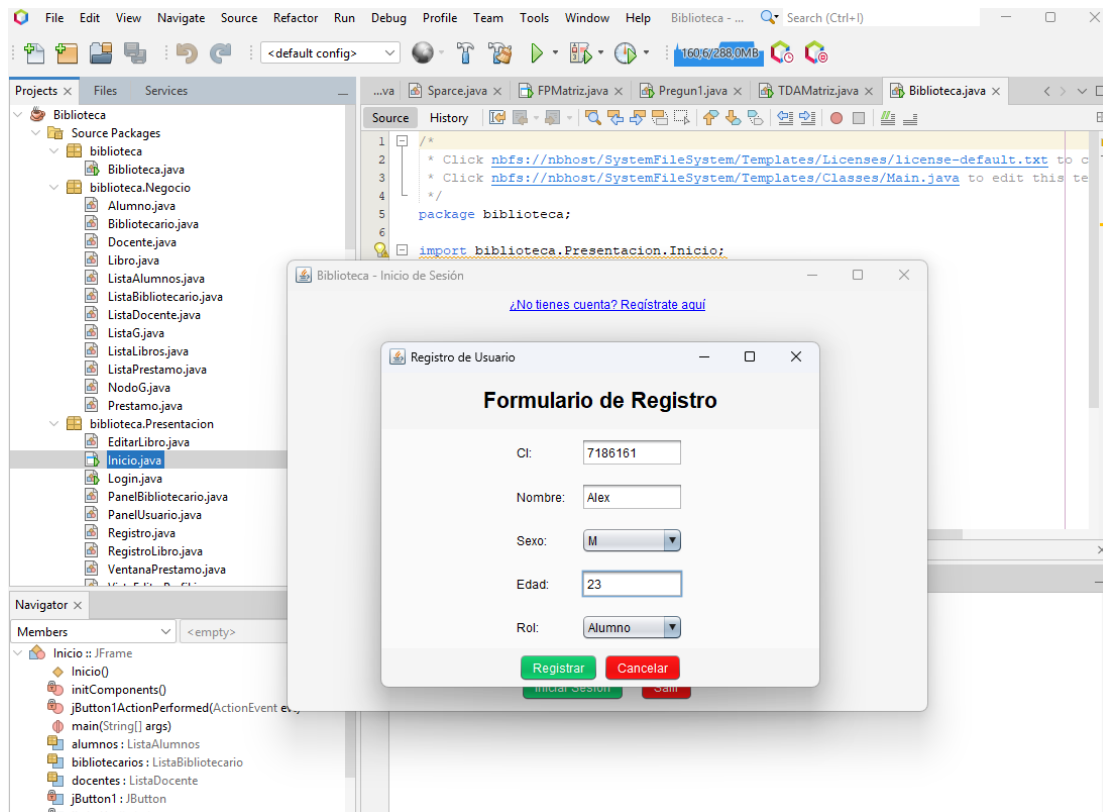


Figura 3

Confirmación de Registro Exitoso y asignación de código.

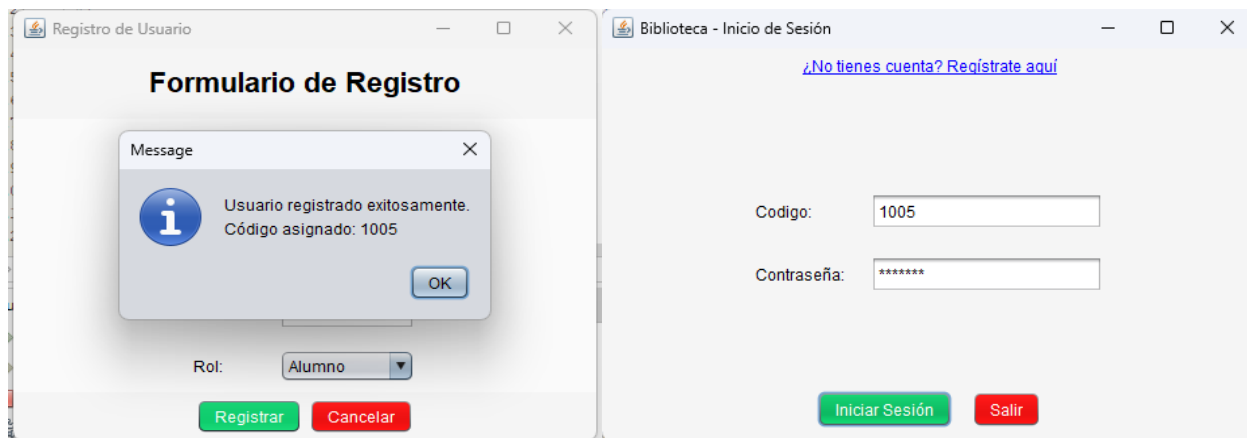


Figura 4

Vista principal del usuario mostrando el menú de opciones y el listado de libros.

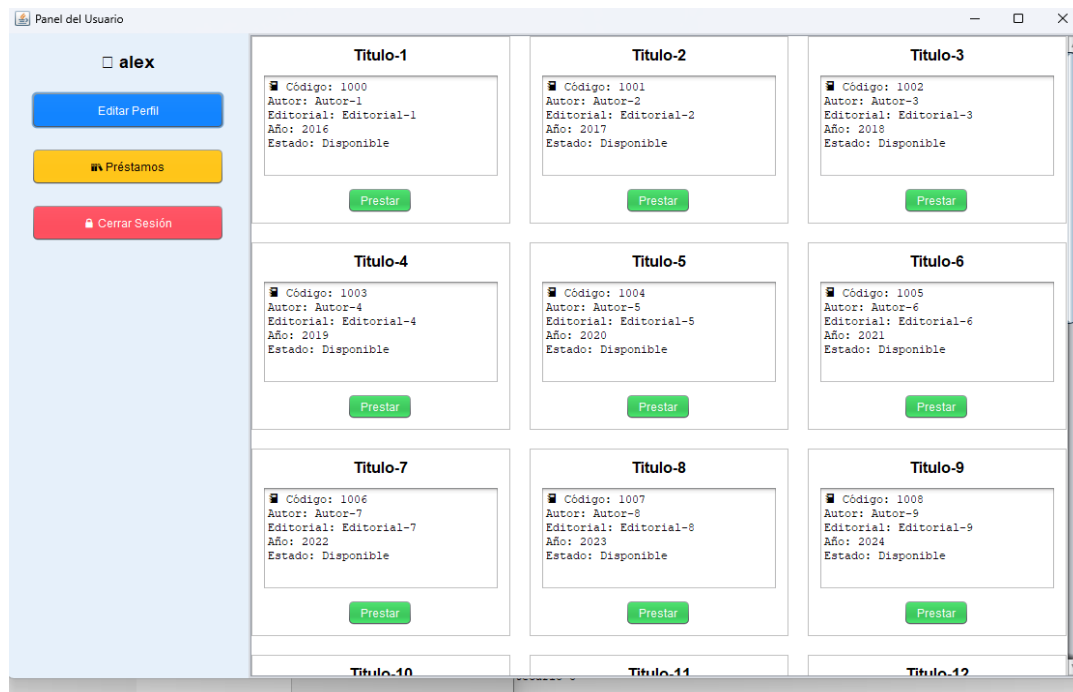


Figura 5

Confirmación de un préstamo registrado exitosamente.

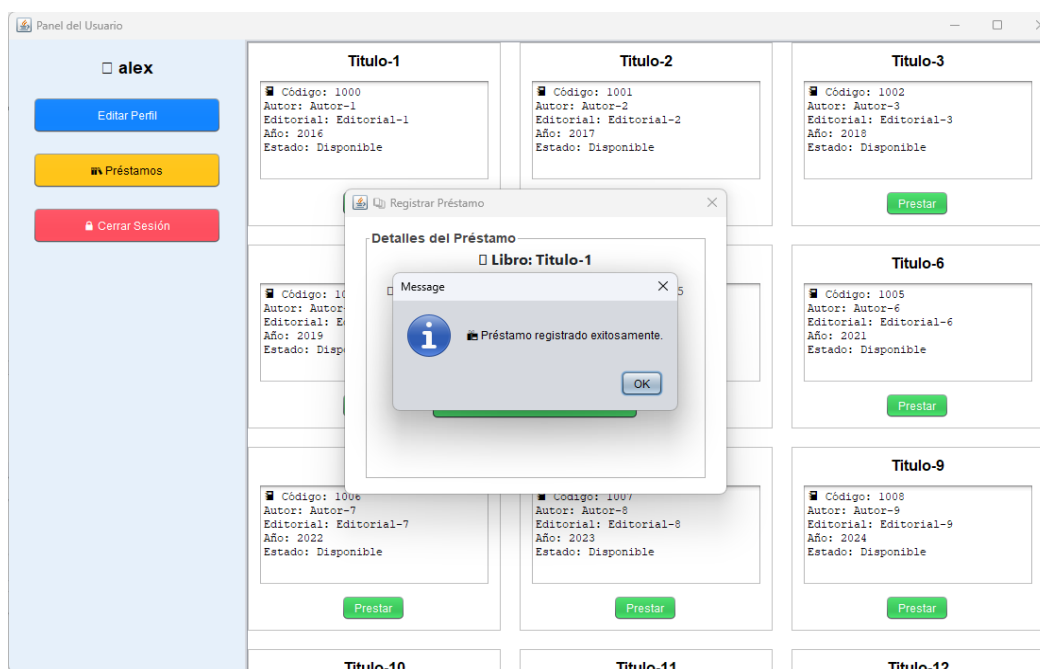
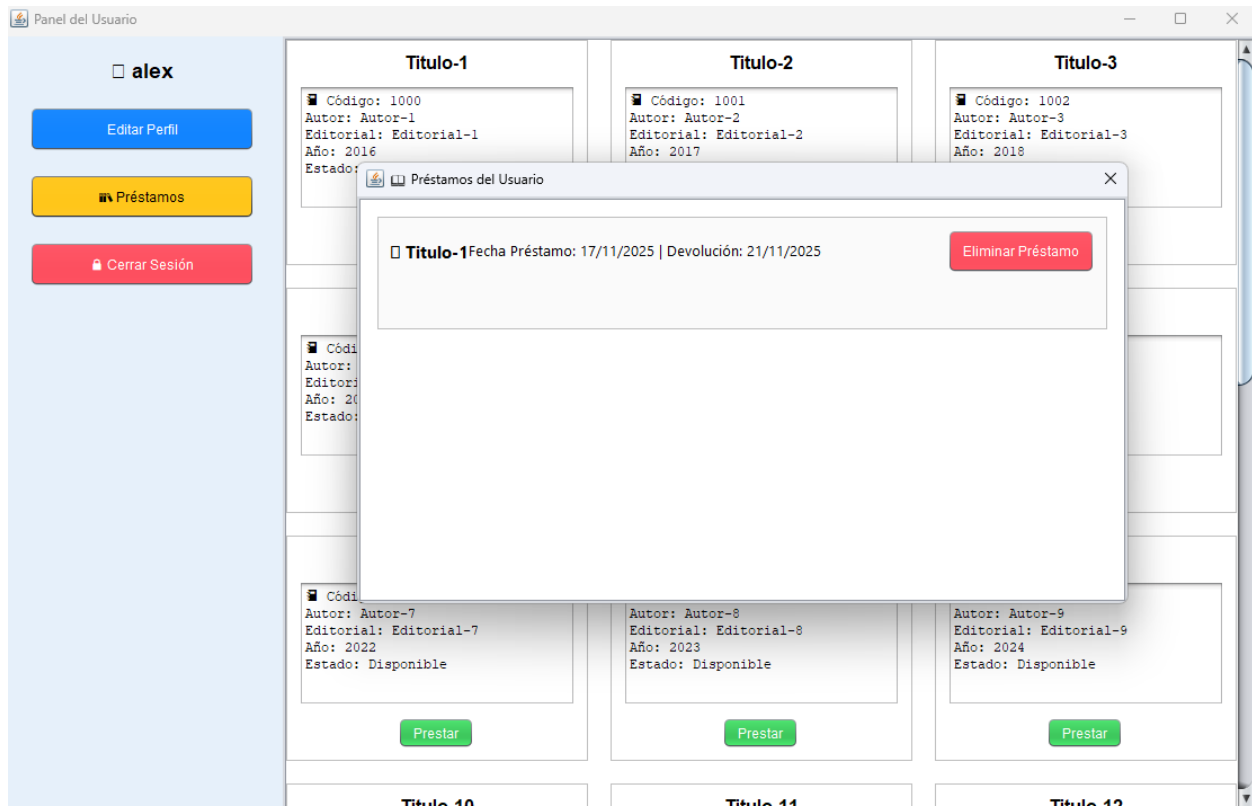


Figura 6

Módulo de visualización y gestión de préstamos del usuario.



5. Marco Teórico y Metodológico

Según (Arias, 2006) el desarrollo del sistema se basa en los principios de la Programación Orientada a Objetos (POO) y en la arquitectura Modelo–Vista–Controlador (MVC). La POO permite representar los elementos del sistema, como libros, usuarios y préstamos, a través de clases y objetos que poseen sus propios atributos y comportamientos. Este enfoque facilita la organización del código, la reutilización de componentes y el mantenimiento general del programa. El modelo MVC, por su parte, divide el sistema en tres partes bien diferenciadas: el Modelo, encargado de la lógica del negocio y las reglas internas del sistema; la Vista, que representa la interfaz gráfica desarrollada en Java Swing y que permite la interacción con el usuario mediante formularios y botones; y el Controlador, que actúa como intermediario entre ambos, procesando las acciones del usuario y actualizando los datos cuando se realizan préstamos, devoluciones o consultas. Esta estructura garantiza un desarrollo ordenado, flexible y fácil de mantener, además de permitir futuras mejoras sin alterar el núcleo del sistema.

En cuanto a las estructuras de datos, se emplearon listas, colas y mapas para lograr un funcionamiento rápido y eficiente. Las listas almacenan la información de libros, usuarios y préstamos activos; las colas organizan las reservas según el orden de solicitud; y los mapas (estructuras parecidas a un “diccionario”) permiten localizar o actualizar información de manera inmediata, sin necesidad de recorrer todas las listas. Gracias a esta implementación, el sistema responde de forma casi instantánea cuando se busca un libro o usuario, mejorando considerablemente la velocidad y el rendimiento general.

La persistencia de datos se implementó mediante serialización de objetos, lo que permite guardar la información registrada en archivos binarios y recuperarla automáticamente al reiniciar el sistema. Este método garantiza la conservación de los datos sin necesidad de usar un servidor de base de datos, lo que resulta práctico y seguro para un entorno académico. Además, deja abierta la posibilidad de migrar el sistema a una base de datos relacional en el futuro, sin modificar su estructura principal ni su arquitectura MVC.

Desde el punto de vista metodológico, el desarrollo se llevó a cabo con un enfoque incremental e iterativo, que consiste en construir el sistema por etapas y realizar pruebas continuas. En una primera fase se analizaron los procesos actuales de la biblioteca y se identificaron los actores principales (bibliotecario, estudiante y docente). Posteriormente se elaboraron los diagramas de casos de uso, las clases y la estructura MVC. En las siguientes etapas se implementaron los módulos de usuarios, libros, préstamos, devoluciones, reservas y reportes, verificando en cada fase su correcto funcionamiento. Finalmente, se realizaron pruebas de validación con datos reales, comprobando la integridad de la información, el control de disponibilidad y la exactitud de los reportes generados.

6. Conclusiones y Recomendaciones

La implementación del sistema automatizado de préstamos de libros para la FICCT permitió demostrar que la digitalización de procesos manuales mejora significativamente la eficiencia y el control dentro de la biblioteca. El uso del lenguaje Java y la arquitectura Modelo–Vista–Controlador (MVC) facilitó la organización del código y la separación de responsabilidades, logrando un sistema estable, modular y fácil de mantener. Con ello, se redujeron los errores humanos y los tiempos de atención, optimizando el flujo de trabajo del personal bibliotecario.

El proyecto cumplió su objetivo principal de ofrecer una herramienta informática que centraliza la información y agiliza las operaciones diarias de registro, préstamo, devolución y generación de reportes. Gracias a la automatización, el bibliotecario puede conocer en tiempo real el estado de los ejemplares, generar reportes de uso y detectar fácilmente usuarios con préstamos vencidos. Además, el sistema contribuye al fortalecimiento académico de los estudiantes que participaron en su desarrollo, ya que les permitió aplicar de forma práctica los conocimientos adquiridos en programación orientada a objetos, estructuras de datos y diseño de software.

Entre los resultados más relevantes se encuentra la notable disminución del tiempo promedio de atención, que pasó de varios minutos en el sistema manual a menos de un minuto con el sistema automatizado. También se evidenció una mejora en la precisión de los registros y en la disponibilidad de información confiable para la toma de decisiones. Estos avances reflejan un impacto positivo tanto en la gestión interna de la biblioteca como en la experiencia de los usuarios finales.

Como recomendaciones, se sugiere implementar una base de datos relacional en futuras versiones, lo que permitiría almacenar la información de forma más robusta y facilitaría la conexión con otros sistemas de la facultad. Asimismo, sería conveniente añadir un módulo de notificaciones automáticas que alerte a los usuarios sobre la fecha de devolución de los libros o sobre nuevos ejemplares disponibles. Finalmente, se recomienda continuar con la mejora de la interfaz gráfica para hacerla aún más intuitiva y adaptable a dispositivos móviles, fortaleciendo la accesibilidad y la modernización del servicio bibliotecario.

En conclusión, el proyecto no solo ofrece una solución tecnológica funcional para la biblioteca de la FICCT, sino que también representa un aporte académico y formativo que evidencia la aplicación de conocimientos teóricos en un entorno real, contribuyendo al desarrollo tecnológico y educativo de la universidad.

Bibliografía

- Arias, F. G. (febrero de 2006). *El Proyecto de Investigacion: Introduccion a la Metodologia Cientifica*. Caracas : Editorial Episteme. Obtenido de https://www.academia.edu/29404107/Fidias_G_Arias_El_Proyecto_de_Investigacion_5ta_Edicion_
- Laudon, K. C. (2016). *Sistemas de Informacion Gerencial*. Mexico: Pearson Educacion. Obtenido de <https://www.pearson.com/es/latin-america/educacionsuperior/laudon.html>
- Mas, V. M. (2015). *Introduccion a la Programacion orientada a Objetos*. febrero, España: Universitat Oberta de Catalunya. Obtenido de <https://openaccess.uoc.edu/server/api/core/bitstreams/e850a312-0435-4674-a055-cbb88f75fb84/content>
- Pinto, J. (septiembre de 2020). *Fundamentos de la Programacion en Java*. Obtenido de https://www.uv.mx/personal/pmartinez/files/2021/03/fundamentosdelaprogramacionenjava_completo2021.pdf