Feasibility Study

# Technical feasibility

--------------------------------------

The technical feasibility study assesses whether the technology, tools, and resources required to build and operate the electronic library system are available, reliable, and cost-effective. The main components of this study include the software and hardware requirements, development tools, and potential technical challenges.

### **.1Software Requirements**

* **Backend Development**: The system will require a robust backend to handle user requests, book uploads, downloads, and other interactions.
* **Frontend Development**: The frontend should be user-friendly, accessible, and responsive.
* **Database Management**: To efficiently manage and organize data related to users, books, and transactions.

### **.2Hardware Requirements**

* **Server Infrastructure**: The electronic library system will require robust server infrastructure to support continuous access and high traffic.
* **Storage**: Sufficient storage capacity is essential to accommodate a growing collection of books and other digital resources.

### .3 **Development Tools**

* **Integrated Development Environment (IDE)**: Recommended options include **Visual Studio Code** for efficient coding, testing, and debugging.
* **Version Control**: Using **Git** for version control will help manage and track code changes throughout the project.
* **Testing and Deployment**: Tools like **GitHub Actions** can automate testing and deployment processes, ensuring smoother updates and bug fixes.

# Schedule feasibility

-------------------------------------

|  |  |  |  |
| --- | --- | --- | --- |
| Phase | Task Description | Estimated Duration | Timeline |
| 1. Planning | - Define project requirements and goals | 1 week | Day 1 - Day 7 |
|  | - Conduct feasibility studies (technical, economic) | 1 week | Day 8 - Day 14 |
|  | - Develop a project plan | 1 week | Day 15 - Day 21 |
| 2. Design | - Design system architecture | 1 week | Day 22 - Day 28 |
|  | - Create wireframes and UI designs | 2 weeks | Day 29 - Day 42 |
|  | - Finalize database design | 1 week | Day 43 - Day 49 |
| 3. Development | - Set up development environment | 1 week | Day 50 - Day 56 |
|  | - Develop front-end interface | 3 weeks | Day 57 - Day 77 |
|  | - Develop back-end functionalities | 3 weeks | Day 78 - Day 98 |
|  | - Integrate database with the application | 2 weeks | Day 99 - Day 113 |
| 4. Testing | - Perform unit testing on individual modules | 2 weeks | Day 114 - Day 128 |
|  | - Conduct integration testing | 1 week | Day 129 - Day 135 |
|  | - User acceptance testing (UAT) | 1 week | Day 136 - Day 142 |
| 5. Deployment | - Deploy system on the server | 1 week | Day 143 - Day 149 |
|  | - Set up user access and permissions | 1 week | Day 150 - Day 156 |
| 6. Maintenance | - Monitor system performance | Ongoing | Day 157 - ongoing |
|  | |  | | --- | | - Fix bugs and update features as needed |  |  | | --- | |  | | Ongoing |  |

# Organizational feasibility

------------------------------------------------

1. **Alignment with Organizational Goals**

Assess how the electronic library system aligns with the organization's mission to provide accessible resources and support digital transformation.

2. **Management Support**

Ensure there is strong support from management or key stakeholders.

3. **Resource Availability**

* **Human Resources**: Evaluate whether you have the necessary staff or need to bring in external developers, designers, or IT support.
* **Technical Resources**: Assess if your existing infrastructure (servers, databases, networks) can support an online library system or if additional technology investments are required.
* **Training Requirements**: Determine the level of training necessary for current staff to operate, manage, and support the system.

4. **User Readiness**

Gauge the readiness of intended users (students, staff, or general users) for the new system.

.

# Economic feasibility

--------------------------------------

### 1. ****Initial Costs****

#### A. ****Software Development:****

* **System Development Cost:** the cost of developing the system with a development team and designers is around $10,000.

#### B. ****Infrastructure:****

* **Hosting Servers:** The cost of hosting the servers could be $200 per month.
* **Cloud Services:** $100 per month for services like Amazon Web Services (AWS).

#### C. ****Security & Data Protection:****

* **SSL and Data Protection:** The cost for implementing security measures is approximately $500.

#### D. ****Software Licensing:****

* **Licenses for Software Tools:** If you're using paid software (like MySQL or certain UI libraries), this could be around $500 annually.

### 2. ****Operating Costs****

#### A. ****Monthly Hosting & Maintenance:****

* Hosting servers = $200 per month.
* Cloud services = $100 per month.
* System Maintenance = $300 per month (for technical support team).

#### B. ****Marketing & Advertising:****

* **Advertising Budget:** $500 per month for online advertising (Google, Facebook, etc.).

#### C. ****Administrative Costs:****

* **Employee Salaries (content management, customer support, etc.):** $1,000 per month.

### 3. ****Potential Revenue Sources****

#### A. ****Subscription or Payment for Downloads:****

* Let's assume a monthly subscription fee of $10 per user.
* If you have 500 users per month, monthly revenue from subscriptions = 500 × $10 = $5,000.

#### B. ****Advertising Revenue:****

* You might generate around $200 per month from ads placed on the platform.

### 4. ****Financial Analysis****

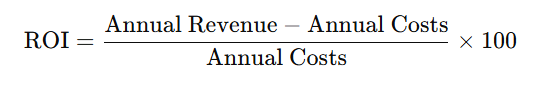
#### A. ****Annual Costs:****

1. **Initial Costs:**
   * System Development Cost = $10,000.
   * Security & Data Protection = $500.
   * Software Licenses = $500.
   * **Total Initial Costs** = $10,000 + $500 + $500 = $11,000.
2. **Annual Operating Costs:**
   * Hosting Servers = $200 × 12 = $2,400.
   * Cloud Services = $100 × 12 = $1,200.
   * System Maintenance = $300 × 12 = $3,600.
   * Marketing & Advertising = $500 × 12 = $6,000.
   * Employee Salaries = $1,000 × 12 = $12,000.
   * **Total Operating Costs** = $2,400 + $1,200 + $3,600 + $6,000 + $12,000 = $25,200.

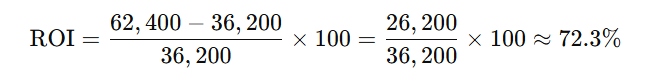
#### B. ****Annual Revenue:****

* **Subscription Revenue:** $5,000 × 12 = $60,000.
* **Advertising Revenue:** $200 × 12 = $2,400.
* **Total Annual Revenue** = $60,000 + $2,400 = $62,400.

#### C. ****Return on Investment (ROI):****



Calculating the ROI:



#### D. ****Payback Period:****

To calculate the payback period, divide the initial costs by the net annual revenue:

* **Net Annual Revenue** = Annual Revenue - Annual Operating Costs = $62,400 - $25,200 = $37,200.



Software Requirements Specification (SRS) for E-Library System

# Introduction

In today’s fast-paced digital world, the need for platforms that provide quick and reliable access to information has become essential. Electronic libraries play a crucial role in facilitating access to a wide range of knowledge sources, such as books, articles, and research materials, without requiring users to visit traditional libraries physically.

This system aims to provide a comprehensive online library that allows users to browse, read, and download books. The system enables users to access a growing collection of books available online, making the process of searching for and selecting the materials they need quick and accurate. With a simple interface and a high-quality user experience, the system strives to meet the needs of readers of all interests and educational levels.

### ****Purpose****

The purpose of this online library system is to create a centralized digital platform where users can easily access, read, and download a wide range of books. This system is designed to support students, researchers, and casual readers by providing a convenient alternative to physical libraries, offering accessible knowledge resources anytime and anywhere.

The primary goals of this system include:

1. **Simplifying Access to Knowledge**: By offering a centralized platform, users can find and access books in various categories without geographical constraints.
2. **Encouraging Digital Reading**: Promote reading and learning by providing easy access to books in digital format.
3. **Enhancing Educational Support**: Serve as a resource hub for students and educators seeking reference material, academic books, and other learning resources.

------------------------------------------------------------------------------------------------------------------------------------------

### ****Scope****

#### ****Product Identification****

The software product developed is named **E-Library System**. This system is designed to serve as an online library, providing functionalities for users to browse, search, read, and download books digitally. Core modules include **Book Management**, **User Management**, **Search and Filter**, **Reading Interface**, **Download Manager**, and **Admin Panel** for system management.

#### ****Product Functions****

The E-Library System will:

* Allow users to search and browse a diverse range of digital books by categories, titles, authors, or keywords.
* Offer an online reading interface for users to access books directly within the system.
* Enable users to download books for offline access.
* Support advanced search and filtering options to make finding resources quick and efficient.
* Provide an admin panel for managing user access, uploading books, and organizing content.

#### ****Application of the Software****

The E-Library System is intended to support educational institutions, libraries, and individual users by providing an accessible, well-organized digital library platform. It will serve students, educators, researchers, and general readers by offering a seamless way to access a broad library collection from any location.

1. **Benefits**:
   * **Easy Access**: Users can access books from any device with internet access, eliminating geographical restrictions.
   * **On-Demand Content**: Provides users with immediate access to reading materials without wait times.
   * **Improved Learning and Research**: Allows students and researchers to find reliable reference material quickly and efficiently.

### ------------------------------------------------------------------------------------------------------------------------------

### ****Definitions, Acronyms, and Abbreviations****

* **E-Library System**: The software application developed to provide a digital library experience where users can browse, read, and download books online.
* **Admin Panel**: A part of the system interface specifically for administrators to manage users and books.
* **User Interface (UI)**: The part of the system where users interact with the library, search for books, and access reading and download options.
* **DBMS (Database Management System)**: Software that manages and organizes data within the library system, such as user accounts, book information, and access logs.
* **Search Module**: A component of the system that enables users to find books using various criteria, such as title, author, category, or keywords.
* **ISBN (International Standard Book Number)**: A unique identifier for books, used for cataloging and searching within the system.
* **PDF (Portable Document Format)**: A file format used for digital books, allowing users to read and download content in a standardized, readable format.
* **Authentication**: The process of verifying a user's identity to grant them access to specific functionalities within the system, such as reading or downloading books.

# General Description

#### Product Perspective

The E-Library System is a web-based application designed to provide users with convenient access to a wide range of digital books. It is positioned within the context of various online reading platforms, digital libraries, and educational resources. Unlike traditional libraries, which require physical presence, the E-Library System offers:

* **Accessibility**: Users can access the library from any location with an internet connection, facilitating learning and reading at their convenience.
* **User-Centric Design**: The system emphasizes user experience, allowing for a more engaging and efficient interaction compared to other digital library offerings.
* **Integration Potential**: The system can be integrated with third-party services for enhanced features, such as user authentication and analytics.

#### Product Functions

The E-Library System will perform the following key functions:

1. **User Management**: Facilitate user registration, login, and profile management.
2. **Book Browsing and Searching**: Allow users to browse books by categories and perform keyword searches to find specific titles or authors.
3. **Reading and Downloading**: Provide an online reading interface and allow users to download books in various formats for offline access.
4. **Feedback and Rating System**: Enable users to rate and provide feedback on books, enhancing community engagement and content quality.
5. **Administration Tools**: Offer administrative capabilities for managing users and the library's book collection.

#### User Characteristics

The eventual users of the E-Library System include:

1. **Students**: Typically tech-savvy individuals seeking access to educational materials and research resources.
2. **Educators**: Teachers and professors who require reference materials and want to recommend resources to students.
3. **Researchers**: Users looking for specific studies or literature in their fields of interest.
4. **General Readers**: Individuals interested in reading for leisure, who may have varying levels of digital literacy.

These user characteristics will influence the system's design, particularly in terms of usability, accessibility features, and the need for clear navigation paths.

#### General Constraints

The following general constraints will affect the design and implementation of the E-Library System:

1. **Technical Constraints**: The system must be compatible with multiple web browsers and mobile devices to ensure wide accessibility.
2. **Legal and Regulatory Constraints**: The system must comply with copyright laws and digital content licensing requirements.
3. **Performance Constraints**: The system should handle a large volume of concurrent users while maintaining quick response times for searches and downloads.
4. **Security Constraints**: Data protection regulations must be followed to ensure user data is secure from unauthorized access.

#### Assumptions and Dependencies

The following assumptions and dependencies affect the requirements stated in the SRS:

1. **Operating System Availability**: It is assumed that users will have access to modern operating systems (Windows, macOS, Linux, iOS, Android) that support the web application.
2. **Internet Connectivity**: The system assumes that users will have reliable internet access, which is critical for accessing digital content.
3. **Device Compatibility**: It is assumed that users will utilize devices capable of running contemporary web browsers with JavaScript and multimedia support.
4. **Content Availability**: The successful deployment of the E-Library System relies on the availability of a diverse range of digital books and materials for inclusion in the library.

# External Interface Requirements

## **User Interfaces**

The User Interfaces (UI) of the E-Library System will be designed to provide an intuitive and engaging experience for users across different devices. The following outlines the key aspects of the user interfaces:

#### **1. General Design Principles**

* **Responsive Design**: The UI will adjust to various screen sizes (desktops, tablets, and smartphones), ensuring accessibility and usability on all devices.
* **Consistency**: Visual elements such as colors, fonts, and button styles will be consistent throughout the application to enhance usability and familiarity.
* **Accessibility**: The design will adhere to accessibility standards to ensure that users with disabilities can navigate and interact with the system effectively.

#### **2. Main Interface Components**

* **Home Page**:
  + A welcoming layout featuring a search bar prominently positioned for quick access to book searches.
  + Categories or genres displayed as tiles or a dropdown menu for easy browsing of available content.
  + Links to user account options (e.g., login, registration) clearly visible.
* **Search Interface**:
  + A dedicated search bar allowing users to input keywords, titles, or author names.
  + Advanced search filters enabling users to narrow results by category, publication date, or format.
* **User Profile Dashboard**:
  + A personalized area where users can view their reading history, saved books, and preferences.
  + Options to update account details, change passwords, and manage reading preferences.
* **Admin Panel**:
  + A secure interface for administrators to manage user accounts, add or update book listings, and generate reports on system usage.
  + Features for monitoring system performance and user activity.

#### **3. Interaction Elements**

* **Buttons and Links**:
  + Clearly labeled buttons for actions (e.g., "Search," "Download," "Read Now") to facilitate easy navigation.
  + Links to external resources or related books prominently displayed to enhance user engagement.

#### **4. Error Handling**

* **Error Messages**:
  + User-friendly error messages that clearly explain the issue and suggest corrective actions (e.g., "Book not found. Please try a different search term.").

## **Hardware Interfaces**

The E-Library System primarily operates as a web-based application, but it may interact with various hardware components to ensure functionality and enhance user experience. The following outlines the key hardware interfaces:

#### **1. Server Requirements**

* **Web Server**:
  + The application will be hosted on a web server that meets the following specifications:
    - Minimum of 8 GB RAM and 4 CPU cores for optimal performance.
    - Sufficient storage (at least 500 GB) to accommodate the database of digital books and user data.
    - Support for HTTPS to ensure secure data transmission between users and the server.
* **Database Server**:
  + A dedicated database server may be utilized, with requirements including:
    - Minimum of 16 GB RAM for efficient data retrieval and management.
    - Robust storage capacity (1 TB or more) to handle growing data needs and backups.
    - Regular backups and data redundancy solutions to protect against data loss.

#### **2. Client Device Requirements**

* **User Devices**:
  + The system is designed to be accessible from various devices, including:
    - **Desktops and Laptops**: Must run on operating systems such as Windows, macOS, or Linux with modern web browsers.
    - **Tablets and Smartphones**: Must support iOS and Android platforms, capable of running current versions of popular web browsers (e.g., Chrome, Safari).

#### **3.Network Requirements**

* **Internet Connectivity**:
  + A stable internet connection is required for both the server and end-users, with minimum recommended speeds:
    - **For Users**: At least 2 Mbps for browsing and reading content without interruptions.
    - **For Server**: A minimum of 10 Mbps upload/download speed to handle multiple concurrent users and ensure fast content delivery.

#### **4. Security Hardware (Optional)**

* **Firewall and Security Appliances**:
  + The system may utilize dedicated hardware firewalls and intrusion detection systems (IDS) to enhance security measures against unauthorized access and cyber threats.

## -------------------------------------------------------------------------------------------------------

## **Software Interfaces**

This section outlines the software interfaces that the E-Library System will interact with, detailing the relationships between the system and other software components, libraries, and third-party services.

#### **Database Management System (DBMS)**

We implement the system using HTML , CSS and PHP

* **DBMS**:
  + The system will use a relational database management system (RDBMS), such as **MySQL** for storing user data, book metadata, and system logs.
  + The database will be accessed via an Object-Relational Mapping (ORM) library to facilitate interactions between the application and the database.

### --------------------------------------------------------------------------------------------------------

## **Communications Interfaces**

This section describes the communications interfaces that the E-Library System will utilize to interact with users and other systems. It outlines the protocols and methods for data exchange, ensuring effective communication between components.

#### **1. Network Protocols**

* **HTTP/HTTPS**:
  + The E-Library System will primarily use the **HTTP** protocol for communication between the client and server, with **HTTPS** implemented for secure data transmission. This ensures that user information and sensitive data are encrypted during transit.

#### **2. Email Communication**

* **SMTP Protocol**:
  + The system will utilize the **Simple Mail Transfer Protocol (SMTP)** for sending email notifications to users. This may include:
    - Account verification emails during the registration process.
    - Password reset links and confirmations for user actions.
    - Notifications about new book additions or system updates.

#### **3. Logging and Monitoring**

* **Logging Interfaces**:
  + The system will implement logging mechanisms to record system events, user actions, and error messages. This will aid in monitoring system performance and debugging issues.
  + Logs will be stored locally or sent to a remote logging service for analysis, ensuring compliance with data retention policies.

User Functional Requirements

### ****For Regular Users:****

1. **Account Management**:
   * Users must be able to create new accounts by providing personal details (e.g., name, email, and password).
   * Users should be able to log in and log out of the system securely.
2. **Book Search and Retrieval**:
   * Users must be able to search for books using keywords such as title, author, or genre.
   * Users should have the option to apply filters like genre, author, publication year, and availability.
   * The system must provide search suggestions (auto-complete) as the user types.
3. **Book Requests**:
   * Users must be able to request books that are unavailable or require approval for access.
   * Users should be able to provide additional information or preferences (e.g., format: physical or digital).
4. **View Book Details**:
   * Users should be able to view detailed information about a selected book, including title, author, genre, language, publication year, and availability status.
5. **Notifications**:
   * Users must receive notifications about the status of their book requests (e.g., approved, rejected, or pending).
6. **Messaging**:
   * Users must be able to view messages sent by the admin, such as updates about book availability or system announcements.

### ****For Admins:****

1. **User Management**:
   * Admins must be able to add new users to the system.
   * Admins should be able to remove inactive or unauthorized users.
2. **Book Management**:
   * Admins must be able to add new books to the system, providing details like title, author, genre, language, and publication year.
   * Admins should be able to remove outdated or irrelevant books from the library.
3. **Request Management**:
   * Admins should be able to view and approve/reject book requests made by users.
   * Admins must be able to provide feedback or reasoning for rejected requests.
4. **Messaging**:
   * Admins must be able to send messages to users to inform them about system updates, request statuses, or book availability.
5. **Search and Monitoring**:
   * Admins must have access to search functionality to find specific users, books, or requests in the system.

**User Non-functional Requirements:**

1. **Usability**:
   * The system should be easy to use for users of various ages and educational backgrounds.
   * The user interface should be intuitive, with guidance for new users to facilitate the registration and system usage process.
2. **Performance**:
   * The system should respond to requests within a few seconds.
   * Users should be able to search for and download books quickly without delays.
3. **Security**:
   * There should be an authentication system to protect users' accounts.
   * Users' personal data and book information should be securely encrypted.
4. **Availability**:
   * The system should be available 99% of the time, with contingency plans in case of any downtime.
5. **Scalability**:
   * The system should be able to handle a large number of users and books without affecting performance.
6. **Reliability**:
   * The system must be reliable, and all functions should work properly without errors.
7. **Compatibility**:
   * The system should be compatible with various browsers and devices (such as smartphones and computers).

System Functional Requirements

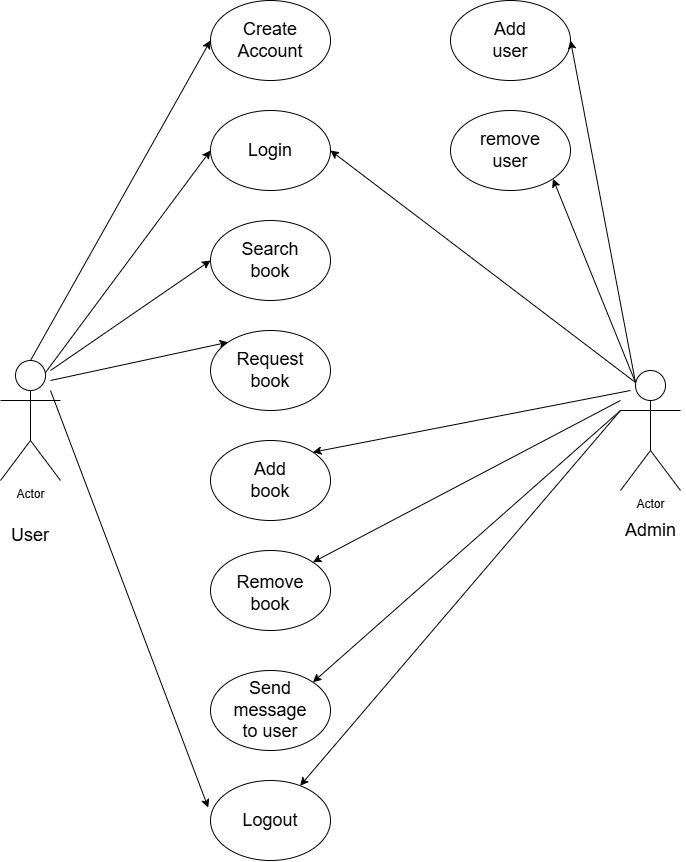
### ****System Requirements:****

1. **Authentication and Authorization**:
   * The system must validate user credentials (e.g., username and password) during login to ensure only authorized users can access their respective functionalities.
   * New users should be able to register by providing personal details such as name, email, and password.
2. **Book Management**:
   * **Add Book**: The system should allow admins to upload new books with details like title, author, category, and file (if downloadable).
   * **Remove Book**: Admins can delete books from the system, ensuring they are no longer accessible to users.
   * **Search Book**: Users can search for books using various filters such as title, author, or category. The search should be fast and provide accurate results.
3. **User Management**:
   * **Add User**: Admins can add new users manually, possibly specifying roles (e.g., regular user or librarian).
   * **Remove User**: Admins can remove inactive or unauthorized users from the system to maintain security.
4. **Request Management**:
   * **Request Book**: Users can submit requests for specific books, possibly with reasons or notes for the admin.
   * **Send Message to User**: Admins can notify users about the status of their requests, overdue books, or any updates via a messaging system integrated into the platform.
5. **Session Management**:
   * The system must allow users to log out securely, ensuring their session is terminated and sensitive data is cleared from the browser or device.

**System Non-functional Requirements:**

1. **Performance**:
   * The system should be able to handle high loads of requests simultaneously.
   * It should support multiple concurrent operations without significant degradation in speed or performance.
2. **Security**:
   * The system must have strong security measures to prevent unauthorized access.
   * Sensitive data, such as personal information and payment data, should be encrypted.
3. **Availability**:
   * The system should be available 24/7.
   * There should be regular backups of data to prevent loss in case of any issues.
4. **Maintainability**:
   * The system should be easy to maintain and update with minimal downtime.
   * It should be flexible enough to allow new features to be added without significantly affecting the current system.
5. **Compatibility**:
   * The system should be compatible with other systems, such as payment systems or inventory management systems.
6. **Scalability**:
   * The system should support an increase in scale (more users or more data) without negatively impacting performance.
   * The system should be expandable to include more features in the future.
7. **Reliability**:
   * The system must run continuously without interruptions.
   * There should be mechanisms to detect and fix errors quickly.

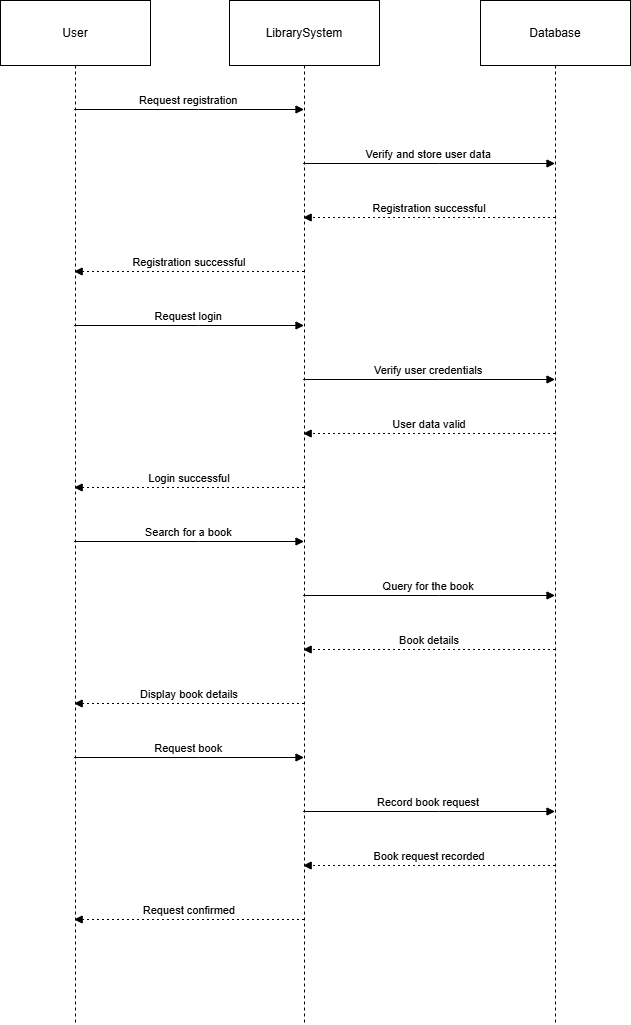
Use Cases Diagram



Class Diagram

#### C:\Users\Nancy\Desktop\class.png

Sequence Diagram



State-Transition Diagram

