

# **BoiChokro**

## *A Hand Book Community*

Software Engineering and Information System Design Lab

Course Code: CSE - 3104

Submitted by: Team OrbitBEE

### **Submitted by:**

Mahmudul Hasan Noman(22CSE001)

Lazmi Rahman Ayman (22CSE004)

Redwanul Haque Rafe (22CSE030)

Tanim Ahmed (22CSE032)

Amin Bhuiyan (22CSE046)

Most. Maliha Akter (22CSE049)

### **Supervised by:**

Md Samsuddoha

Assistant Professor

Department of Computer Science and Engineering

University of Barishal

Submission Date: 5-02-2026

# Table of Contents

<b>1. Introduction.....</b>	<b>4</b>
1.1 Background of the Study .....	4
1.2 Problem Statement.....	5
1.3 Objectives of the System .....	6
1.4 Scope of the Project.....	6
1.5 Environmental & Social Impact .....	7
<b>2. Existing Work .....</b>	<b>8</b>
<b>3. Proposed System (BoiChokro) .....</b>	<b>9</b>
3.1 Overview of BoiChokro.....	9
3.2 Key Features .....	9
3.3 System Objectives and Benefits.....	10
<b>4. Software Requirements Specification (SRS) .....</b>	<b>11</b>
<b>4.1 Inception.....</b>	<b>11</b>
4.1.1 Introduction .....	11
4.1.2 Identifying Stakeholders.....	11
4.1.3 Recognizing Multiple Viewpoints.....	12
4.1.4 Working Towards Collaboration .....	14
<b>4.2 Elicitation .....</b>	<b>16</b>
4.1 Introduction .....	16
4.2 Eliciting Requirements.....	16

4.3 Collaborative Requirements Gathering.....	16
4.4 Quality Function Development.....	17
<b>4.5 User Stories .....</b>	<b>19</b>
4.5.1 User Stories – End User .....	19
4.5.2 User Stories – Library .....	19
4.5.3 User Stories – Administrator .....	20
4.5.4 User Stories – Donor.....	20
4.5.5 User Stories – Payment System .....	20
<b>4.3 Modeling.....</b>	<b>21</b>
<b>4.3 Scenario Based modeling .....</b>	<b>21</b>
4.3.1 Use Case Scenarios and Description .....	21
4.3.2 Use Case Diagram.....	31
4.3.3 Activity & Swimlane Diagram .....	39
<b>4.4 Data Modeling .....</b>	<b>52</b>
4.4.1 Entity Relationship (ER) Diagram .....	52
4.4.2 Schema Diagram .....	59
<b>4.5 Class-Based Modeling.....</b>	<b>60</b>
<b>4.6 Flow-Oriented Modeling .....</b>	<b>63</b>
4.6.1 Data Flow Diagram.....	63
<b>4.7 Behavioral Modeling .....</b>	<b>70</b>
4.7.1 State Transition Diagram.....	70
4.7.2 Sequence Diagram.....	77
<b>5. Implementation.....</b>	<b>81</b>

5.1. Technology Stack .....	82
<b>6. SQA &amp; Testing .....</b>	<b>83</b>
6.1 Test Case Table .....	84
6.2 Testing Tool: Selenium WebDriver .....	85
<b>7. Conclusion &amp; Future Work .....</b>	<b>86</b>
7.1 Conclusion.....	86
7.2 Future work.....	87

## **1. Introduction:**

Technology has evolved and therefore altered how individuals get access to resources and information. Nevertheless, hard copy (physical) books are still used by many readers to study and read even in the digital age. Students, researchers and general readers are very fond of using physical books due to their comfort, no need to use devices and the long-term nature of their use. Nonetheless, buying new books is usually costly and most books are only read once and the rest are not utilized. Meanwhile, there are significant numbers of individuals who are challenged with accessibility to books because of monetary constraints.

In order to solve these problems, a web-based system is suggested, called BoiChokro. BoiChokro mostly targets hard copy book readers whereby users are able to sell and share books through an online system where they can donate and borrow books. Another system feature introduced is library profile where libraries are allowed to enroll and categorize available books. Through the system the users can request to borrow books and librarians can control issuing and returns of books.

Moreover, BoiChokro has a community section where users have a chance to comment on books, like posts, and praise other users comments. This assists in creating an interactive community of readers and promotes sharing of knowledge among readers. BoiChokro also contributes to the environmental sustainability by encouraging the reuse of books, swapping, donation, and borrowing of books thereby lowering the use of paper and saving trees.

### **1.1 Background of the Study :**

Under traditional book distribution systems, the reader normally purchases new books in bookstores or online stores. This process makes the process more expensive to the user and causes an overuse of paper resources. It is common to find many students buying academic books per semester and then do not use them again. Such books go into waste back at home and other students might not be able to afford the same books.

Libraries are also a significant repository of books and most libraries lack an effective digital platform where they can display books they have or process of borrowing and

issuing books in virtual format. Consequently, users tend to be unaware of the books that can be found in local libraries.

Due to the fast development of web technologies, the process of uniting the users, sellers, donors, and libraries on the same platform has become simpler. Book sharing can be made more efficient with the aid of online platforms to remove physical barriers. The context behind this research is to apply the current web technology in designing a central system with users sharing books, donating books to the needy and borrowing books in libraries in an easy and structured way.

BoiChokro is expected to fill this gap by offering a platform where people can sell, exchange, donate and borrow hard copy books, besides offering a community reading space.

## **1.2 Problem Statement:**

The current system of distribution and use of books has a number of issues:

1. New books cannot be afforded by many students and readers.
2. It lacks a specific platform where one can easily sell or exchange used hard copy books with other readers.
3. There has been no organization in the processes of donating books and the donors are usually not aware where the books are most required.
4. Libraries have not gotten a centralized place on the internet where they can show available books and can easily borrow and issue them.
5. There is no right community platform where the readers can discuss books and make recommendations.
6. Books that are not utilized are also a source of waste, which escalates the use of papers that adversely impacts the environment.

These issues have caused several books that can be used to go to waste and many readers are not able to access the books they require. The necessity of a single system that would be able to unite users, libraries, and donors within a single platform is great.

### **1.3 Objectives of the System:**

The key goals of the BoiChokro are:

1. To offer the Internet platform through which customers will be able to sell used books.
2. To allow the user to replace books with other readers.
3. To implement a donation system that will allow donors to request donation and the administration to receive and give books to those in need or organisations.
4. To establish a library profile system using which libraries will be able to register and list books and also manage borrowing and issuing of books using librarians.
5. To enable customers to borrow books of registered libraries via the site.
6. To create a community system in which users have the opportunity to discuss the books, like the post, and comment on the opinion of the other users.
7. To encourage the recycling of books and minimize the level of waste paper usage.

In order to establish a low cost and reachable platform among students and readers.

### **1.4 Scope of the Project:**

The key goals of the BoiChokro are:

1. To offer the Internet platform through which customers will be able to sell used books.
2. To allow the user to replace books with other readers.

3. To implement a donation system that will allow donors to request donation and the administration to receive and give books to those in need or organisations.
4. To establish a library profile system using which libraries will be able to register and list books and also manage borrowing and issuing of books using librarians.
5. To enable customers to borrow books of registered libraries via the site.
6. To create a community system in which users have the opportunity to discuss the books, like the post, and comment on the opinion of the other users.
7. To encourage the recycling of books and minimize the level of waste paper usage.

In order to establish a low cost and reachable platform among students and readers.

### **1.5 Environmental & Social Impact:**

BoiChokro has positive environmental and social impacts as follows:

#### **Environmental Impact:**

BoiChokro can enhance sustainability of the environment by encouraging the use of books rather than buying new ones. By reusing books:

1. The necessity to produce new papers is diminished.
2. A lesser number of trees are felled to paper.
3. Waste of books is also reduced, and there is a more efficient way to utilize the resources.
4. The site creates awareness of environmentally friendly activities and green consumption.

#### **Social Impact:**

BoiChokro has relevant social effect:

1. Aids needy students and readers to access books.
2. Promotes the spirit of giving and assisting others by donations.
3. Helps the libraries to become more visible and easier to borrow.



4. Establishes a good reading society where individuals are able to share knowledge and views.
5. Encourages reading and educating in the society.

## **2. Existing Work**

A tradition of reading and books sharing exists in Bangladesh. informal networks like friends and university notice boards and local community groups. There are many students and readers, who rely on physical bookstores, local libraries, and book. bazaars in order to reach reading materials. In spite of the fact that online markets like general e. On commerce platforms, users can sell or purchase second-hand items including books there. Does not have a specific platform on hard copy book based in Bangladesh. exchange, gift and borrowing in libraries.

Part of the existing platforms and practices involve:

### **Online Marketplaces:**

Second hand books are usually sold by the user procedure of general e-commerce websites or even by social media. groups. These platforms, however, are not specialists in books and do not target the. Categorizing academic textbooks, handling swaps, or other specific needs of readers. facilitating borrowing.

### **Community and Social Media Groups:**

A significant number of book readers in Bangladesh rely on a Facebook group and WhatsApp communities. share, sell, or request books. Although these groups facilitate the connection of users, they fail to offer. organized working procedures, computerized cataloguing of books or database management based tracking, transactions.

### **Library Systems:**

Books are normally handled in traditional libraries in Bangladesh (including university libraries). using hand or simple electronic resources. Most of the smaller community libraries are limited. technological and does not have online catalog systems. The readers have to visit physically. use of these libraries to check on books or to request to borrow them.

### **Donation Initiatives:**

Donation of books in Bangladesh is commonly held through the NGOs, local educators and community volunteers. These programs are targeted at poor students, yet coordination is frequently manual in nature. No digital platform to apply to is centralized. distribute donated books in regions, or manage. Generally, there are piecemeal solutions to book exchange, selling, donation, and borrowing, no one offers one unified platform, which would connect donors, sellers, readers, and libraries within an automated and searchable system. This gap emphasizes the need to have a systematic solution which is specific to the requirements of book readers in.

### **3. Proposed System (BoiChokro)**

The proposed solution, BoiChokro, will fill the gaps that exist by developing a system that fulfills the needs that are missing.

integrated web-based system to promote selling, swapping, donation, library.

borrowing, and community interaction, all geared towards the context of Bangladesh.

#### **3.1 Overview of BoiChokro:**

BoiChokro is a Web-based application through which people can:

**Sell Used Books:** Once a user no longer requires a book, one can sell it to them at a fair price, so that other users can buy the prices cheaply.

**Swap Books:** It allows the user to provide books to swap, which allows book exchange without a monetary transaction.

1. **Donate Books:** Users have the opportunity to request their books donation and the system handles it. administration deals with the gathering and distribution to the needy.
2. **Library Integration:** Registered libraries are capable of uploading the catalogs of their books. Users is able to search books and ask to borrow them. Librarians can issue and administer returns via the system.
3. **Community Feature:** A discussion-style section where the members share book-related topics,
4. promote reads, and comment on posts.

#### **3.2 Key Features:**

1. **User Profiles:**

Users make accounts to follow their lists of books, their history of donations, their requests

to borrow, and community activity.

## 2. Book Listings:

Books are divided (e.g., academic, fiction, non-fiction). In both listings there are details like title, author, condition, price (in case of sale) and availability.

## 3. Swap Requests:

Messaging gives users the option to make swap offers and negotiate with other users.

## 4. Donation Workflow:

The donations are done online by the donors. Administrators process requests, organize pick-up, as well as distribute to schools, community centers or needy students.

## 5. Library Borrowing System:

Libraries have their online catalogues. The users search availability, request borrowing and librarians endorse or handle returns.

## 6. Community Interaction:

Users have the opportunity to make posts with references to books, review, comment, and praise (like). posts -promoting reading in Bangladesh.

### **3.3 System Objectives and Benefits:**

The system is aimed at the following:

1. Improve the access to books among students and readers in Bangladesh.
2. Swap and resell books to reduce the cost of purchasing a book.
3. facilitate the donation system to help poor readers.
4. Provide support libraries with online catalog and borrowing system.
5. Develop a reading community to engage in discussion and learning together.

The combination of these features will transform BoiChokro to a book sharing platform. assistance, and participation in the Bangladeshi environment.

## 5.1 Inception

### 5.1.1 Introduction:

Inception is the beginning phase of requirements engineering. It defines how does a software project get started and what is the scope and nature of the problem to be solved. The goal of the inception phase is to identify concurrence needs and conflict requirements among the stakeholders of a software project.

To establish the ground work we have worked with the following factors related to the inception phases:

1. Identifying Stakeholders
2. Recognizing multiple viewpoints
3. Working towards collaboration
4. Asking the First Questions

### 5.1.2 Identifying Stakeholders:

A stakeholder refers to any individual or group who is directly or indirectly affected by the development, deployment, and operation of the system. Stakeholders include users who interact with the system regularly, as well as organizations and individuals who influence system decisions, maintenance and policy enforcement.

To identify the stakeholders of the BoiChokro – A Hand Book Community system, we analyzed the objectives of the platform and considered the following questions:

- a. Who will use the system regularly?
- b. Who will manage and control system operations?
- c. Who will benefit from the outcomes of the system?
- d. Who will support financial transactions and technical maintenance?
- e. Whose activities will be affected after system deployment?

#### **End Users (Readers / Students / Book Lovers):**

End users are the primary stakeholders of the system. They directly interact with BoiChokro to buy, sell, swap, and donate books within their local community. They also participate in community discussions, use wishlist and alert features, and view environmental impact statistics.

#### **2. Library Authorities (Public Libraries, University Libraries):**

Library authorities use the system to register libraries, manage available books, approve user bookings, maintain lending history, and manage library members. They play a key role in ensuring book availability and organized circulation.

### **3. System Administrator:**

The system administrator is responsible for managing the overall platform. This stakeholder controls user roles, approves book listings and donations, moderates community content, manages access control, and generates system-wide reports.

### **4. Donors (Individuals and Institutions):**

Donors contribute books to the platform for donation purposes. They are interested in a transparent and efficient donation process and expect to see the social and environmental impact of their contributions.

### **5. Educational Institutions (Schools and Community Libraries):**

Educational institutions receive donated books through the system. They benefit from easy coordination with donors and libraries and rely on the system for fair and organized book distribution.

### **6. Payment Service Provider:**

The payment service provider facilitates secure online transactions for book purchases. This stakeholder ensures payment integrity, transaction security, and smooth integration with the platform.

### **7. Development and Maintenance Team:**

Developers are responsible for designing, implementing, maintaining, and upgrading the system. They handle technical issues, system updates, and ensure long-term scalability and reliability.

## **5.1.3 Recognizing Multiple Viewpoints:**

Each stakeholder group has different expectations and concerns regarding the BoiChokro system. Recognizing these viewpoints helps in resolving conflicts and aligning system requirements with stakeholder needs.

### **1.3.1 End User Viewpoints**

- a. Affordable access to physical books within nearby locations.
- b. Easy options to buy, sell, swap, and donate books.

- c. Automated wishlist alerts for desired books.
- d. Safe and interactive community space for discussions and recommendations.
- e. Awareness of environmental impact through visual dashboards.

### **1.3.2 Library Authority Viewpoints**

- a. Ability to register libraries and manage book inventories.
- b. Control over approving user bookings.
- c. Accurate maintenance of lending history.
- d. Increased visibility through geo-location based search.
- e. Access to analytics on book usage and demand.

### **1.3.3 System Administrator Viewpoints**

- a. Full administrative control over users, libraries, and book listings.
- b. Approval authority for new listings and donations.
- c. Tools for moderating community posts and chat content.
- d. Generation of reports and performance analytics.
- e. Enforcement of role-based access control.

### **1.3.4 Donor Viewpoints**

- a. Simple and reliable book donation process.
- b. Transparency regarding where donated books are distributed.
- c. Clear visibility of social and environmental contributions.

### **1.3.5 Educational Institution Viewpoints**

- a. Easy receipt of donated books for students and readers.
- b. Smooth coordination with donors, libraries, and administrators.
- c. Fair and organized distribution of books.

### **1.3.6 Payment Service Provider Viewpoints**

- a. Secure and reliable payment processing.
- b. Smooth integration of payment APIs.
- c. Protection against transaction failures and fraud.

### **1.3.6 Developer's viewpoints:**

- a. Easy to develop.
- b. No ambiguous requirement.

### **5.1.4 Working Towards Collaboration:**

Since stakeholders may have overlapping or conflicting requirements, collaboration is necessary to reach a balanced and practical solution. To achieve this, the following steps were followed:

1. Identification of common and conflicting requirements.
2. Categorization of requirements based on priority.
3. Evaluation of feasibility and system constraints.
4. Finalization of requirements through consensus.

#### **Common Requirements**

- a. User-friendly and efficient system.
- b. Web-based access.
- c. Secure authentication and role-based access.
- d. Easy book discovery and management.

#### **Conflicting Requirements**

- a. Maximum feature set vs. limited budget.
- b. Open access vs. controlled access.
- c. Advanced features vs. system simplicity.

After analysis and prioritization, the final requirements were agreed upon.

#### **Final Requirements**

- a. Secure user registration with email OTP verification.
- b. Role-based access control for users, libraries, and administrators.
- c. Book posting for sale, swap, and donation.
- d. approval mechanism for listings and donations.
- e. Area-based and category-based book search.
- f. Wishlist and automated availability alerts.
- g. Library registration and lending management.
- h. Community discussion and chat system.
- i. Integrated payment system.
- j. Environmental impact tracking and visualization.
- k. Administrative reporting and moderation.

### **Asking the First Questions**

During the inception phase, a set of context-free questions was used to understand the project goals, stakeholder expectations, and system constraints. These questions helped clarify:

- a. The core problem to be solved.
- b. Expected benefits of the system.
- c. Possible alternatives to the proposed solution.
- d. Stakeholder perceptions of success.
- e. Communication effectiveness among stakeholders.

The answers to these questions guided the requirement analysis and shaped the overall system vision.

## **Conclusion**

The inception phase of the BoiChokro project established a clear understanding of the system's purpose, scope, and stakeholders. It helped identify key requirements, resolve conflicting viewpoints, and define a shared vision for the platform. This phase laid a strong foundation for subsequent requirement analysis, design, and development activities, ensuring that BoiChokro effectively addresses community needs, promotes sustainable book reuse, and strengthens local engagement.



## **4.2. Elicitation**

### **4.2.1 Introduction**

Elicitation is a task that helps stakeholders define what is required from a software system. During the elicitation phase, several challenges may arise, such as problems of scope, problems of requirement volatility, and problems of understanding among stakeholders. Identifying and managing these issues at an early stage is essential to ensure that the system meets real user needs. However, elicitation is not an easy task. To overcome these challenges, we conducted the requirements elicitation activities for the BoiChokro (A Hand Book Community) system in a structured, organized, and systematic manner. This approach helped us clearly understand stakeholder expectations and transform them into well-defined system requirements.

### **4.2.2 Eliciting Requirements**

Unlike the inception phase, where a Question and Answer (Q&A) approach is primarily used, the elicitation phase employs a requirements elicitation format that combines elements of problem solving, elaboration, negotiation, and specification. This process requires close cooperation between end users, system administrators, and developers to successfully elicit accurate and complete requirements.

To elicit requirements for the BoiChokro system, we carried out the following four activities:

1. Collaborative Requirements Gathering
2. Quality Function Deployment
3. Usage Scenarios
4. Elicitation Work Products

### **4.2.3 Collaborative Requirements Gathering**

Various approaches to collaborative requirements gathering have been proposed, each utilizing different interaction scenarios. For the BoiChokro project, we followed

a collaborative approach involving discussions and analysis with potential stakeholders.

The following steps were completed during the collaborative requirements gathering process:

- a. Meetings and informal discussions were conducted with potential end users such as students and book readers to understand their expectations from a local book-sharing platform.
- b. Library representatives were consulted to identify their requirements regarding book management, user bookings, lending history, and library membership management.
- c. Discussions were held regarding the challenges faced in acquiring affordable physical books and managing unused books within local communities.
- d. Feedback was collected on community interaction features, donation processes, and environmental impact awareness.
- e. Finally, based on these discussions and observations, the final requirement list for the Boichokro system was selected and refined.

## **4.2.4 Quality Function Development**

### **Normal Requirements**

These are essential system functionalities. Absence of these would make the system unacceptable.

- a. User, Library, and Admin Core Management.
- b. Create and update user personal information.
- c. Secure password hashing.
- d. Role-based access control (Admin, Library, User).
- e. Email OTP verification and password recovery.
- f. Book & Library Management.
- g. Post books for sale, swap, or donation.
- h. Register libraries and available books.
- i. Manage book listings and availability.

- j. Approve new listings and donations.
- k. Manage all user, book, and library data.
- l. Moderate community posts.
- m. Manage access control.
- n. Communication & Notifications.

## **Expected Requirements**

These improve user satisfaction proportionally—the better they work, the happier the users are.

- a. User Experience & Transactions.
- b. View transaction history.
- c. Maintain wishlist.
- d. Automated alerts for book availability.
- e. Track reused and donated books.
- f. Review and rating system.
- g. Library Operations.
- h. Approve user bookings.
- i. Geo-location search for books and libraries.
- j. Maintain lending history.
- k. Manage library membership.
- l. Community Interaction.
- m. Participate in community discussions.
- n. Post requests for books.
- o. Like, comment, and filter community posts.
- p. Report inappropriate content.
- q. Generate reports (Admin).
- r. Message alerts in chat module.

## **Exciting Requirements:**

These are not expected by users but significantly enhance satisfaction and system uniqueness.

- a. Environmental Impact Features.
- b. View environmental impact statistics.
- c. Calculate paper and trees saved.
- d. Visual dashboards showing contribution impact.
- e. Integrated payment system.

### **4.2.5 User Stories:**

#### **1. User Stories – End User**

1. **As a user**, I want to create an account with email verification, so that my account remains secure.
2. **As a user**, I want to buy used books within my locality, so that I can avoid delivery costs.
3. **As a user**, I want to sell my old books I no longer need, so that others can benefit from them.
4. **As a user**, I want to add books to my wishlist, so that I get notified when they become available.
5. **As a user**, I want to chat with other users, so that I can negotiate or discuss books easily.
6. **As a user**, I want to post or see book reviews and recommendations, so that I can help other readers and get help from others.

#### **2. User Stories – Library**

1. **As a library**, I want to register and list available books, so that users can find them easily.
2. **As a library**, I want to approve or reject user bookings, so that books are properly managed.
3. **As a library**, I want to maintain lending history, so that records remain transparent.

4. **As a library**, I want analytics on book usage, so that I can improve library services.

### **3. User Stories – Administrator**

1. **As an admin**, I want to manage users and roles, so that access control is enforced.
2. **As an admin**, I want to approve book listings and donations, so that platform quality is maintained.
3. **As an admin**, I want to moderate community posts, so that inappropriate content is removed.
4. **As an admin**, I want to generate system reports, so that performance and usage can be analyzed.

### **4. User Stories – Donor**

1. **As a donor**, I want to donate books to schools and libraries, so that underprivileged students benefit.
2. **As a donor**, I want to see impact statistics, so that I feel motivated to donate more.

### **5. User Stories – Payment System**

1. **As a payment system**, I want to securely process payments, so that transactions are safe and reliable.

## 4.3 Scenario Based Modeling

### 4.3.1 Use Case Scenarios

Level-0 (System)	Level-1 (Module)	Level-2 (Use Case)	Actors
Book Reuse & Sharing System	Authentication	Sign Up	User, Library
		Email OTP Verification	User, Library
		Sign In	User, Library, Admin
		Sign Out	User, Library, Admin
		Change Password	User, Library, Admin
		Password Recovery	User, Library
	User Management	Create Personal Profile	User

		Update Personal Information	User
		Role-Based Access Control	Admin
	Book Management	Post Book (Sell / Swap / Donate)	User
		Upload Book Details	User
		Edit Book Listing	User, Library

		Delete Book Listing	User, Library
		Approve Book Listings	Admin
	Library Management	Register Library	Library
		Manage Available Books	Library
		Approve User Bookings	Library
		Maintain Lending History	Library
	Search & Transactions	Search Books	User, Library
		Filter Books (Category / Location)	User
		Geo-location Based Search	User
		Request Book	User

	Wishlist & Alerts	Transaction History	User
		Add Book to Wishlist	User
		Book Availability Alerts	User
	Community Module	Post Community Discussion	User
		Request Books	User
		Like / Comment on Posts	User
		Filter Community Posts	User

		Report Inappropriate Content	User
		Moderate Community Posts	Admin
	Chat Module	Send Message	User
		Receive Message Alerts	User
		Real-Time Chat	User
		Chat Moderation	Admin
	Notifications	Dashboard Notifications	User, Library
		Email Notifications	User, Library
	Environmental Impact Dashboard	Track Reused Books	User, Admin
		Track Donated Books	User, Admin
		Calculate Paper Saved	System
		View Contribution Statistics	System
	Payments	Make Payment	User
		Payment Verification	User
		Manage Users	Payment Gateway
	Administration	Manage Libraries	Admin
		Manage Books	Admin
		Generate Reports	Admin
		Manage Access Control	Admin



## 4.3.1 Use Case Description

### 01. Authentication

#### i. Use Case: Sign Up

**Primary Actors:** User, Library

**Goal in Context:** To register a new user or library into the system.

**Preconditions:**

- a. Enters System provides a registration interface.
- b. User/Library is not already registered

**Trigger:** User/Library wants to access the system.

**Main Scenario:**

- a. Actor opens the Sign Up page.
- b. Required details (name, email, role, password).
- c. System validates inputs.
- d. System sends OTP to registered email.
- e. Actor verifies OTP.
- f. System confirms registration.

**Exceptions:**

- a. Invalid OTP
- b. email already registered.

**Priority:** Essential

**When Available:** First increment

#### ii. Use Case: Sign In

**Primary Actors:** User, Library, Admin

**Goal in Context:** To access the system.

**Preconditions:** Actor must be registered.

**Trigger:** Actor wants to log in.

**Main Scenario:**

- a. Actor enters email and password.
- b. System authenticates credentials.
- c. Dashboard is displayed.

**Exceptions:**

- a. Invalid credentials.
- b. Blocked account.

**Priority:** Essential

**When Available:** First increment

### **iii. Use Case: Sign Out**

**Primary Actors:** User, Library, Admin

**Goal in Context:** To securely exit the system.

**Preconditions:** Actor must be logged in.

**Trigger:** Actor selects Sign Out.

**Main Scenario:**

- a. System ends session.
- b. Redirects to home page.

**Priority:** Essential

### **iv. Use Case: Password Recovery / Change Password**

**Primary Actors:** User, Library, Admin

**Goal in Context:** To reset or update password.

**Preconditions:** Actor has a registered email.

**Trigger:** Actor forgets or wants to change password.

**Main Scenario:**

- a. Actor requests password reset/change.
- b. OTP/link sent to email.
- c. Actor sets new password.
- d. System confirms update.

**Exceptions:**

- a. Weak password,
- b. Invalid OTP.

**Priority:** Essential

## **2. User Management**

### **i. Use Case: Create Personal Profile**

**Primary Actors:** User

**Goal in Context:** To create a personal profile.

**Preconditions:** User must be logged in.

**Trigger:** User accesses profile section.

**Main Scenario:**

- a. User enters profile details.
- b. System saves profile.

**Priority:** Expected

### **ii. Use Case: Role-Based Access Control**

**Primary Actors:** Admin

**Goal in Context:** To manage permissions.

**Preconditions:** Admin logged in.

**Main Scenario:**

- a. Admin selects user/library.
- b. Assigns or updates role.
- c. System updates access rights.

**Priority:** Essential

### **3. Book Management**

#### **i. Use Case: Post Book (Sell / Swap / Donate)**

**Primary Actors:** User

**Goal in Context:** To list a book for reuse.

**Preconditions:** User logged in.

**Main Scenario:**

- a. User enters book details and mode (sell/swap/donate).
- b. Uploads images.
- c. Submits listing.
- d. Admin approves listing.

**Exceptions:** Invalid details.

**Priority:** Essential

#### **ii. Use Case: Edit / Delete Book Listing**

**Primary Actors:** User, Library

**Goal in Context:** To maintain accurate listings.

**Preconditions:** Listing exists.

**Main Scenario:**

- a. Actor selects listing.

- b. Updates or deletes details.
- c. System confirms action.

**Priority:** Essential

## **4. Search & Transactions**

### **i. Use Case: Search Books**

**Primary Actors:** User, Library

**Goal in Context:** To find available books. **Preconditions:** System has active listings. **Main Scenario:**

- a. Actor enters keywords or filters.
- b. System displays results.

**Exceptions:** No results found.

**Priority:** Essential

### **ii. Use Case: Request Book**

**Primary Actors:** User

**Goal in Context:** To request a listed book.

**Preconditions:** Book is available.

**Main Scenario:**

- a. User sends request.
- b. Owner/Library receives notification.
- c. Request approved or rejected.

**Priority:** Essential

## **5. Community Module**

### **i. Use Case: Post Community Discussion**

**Primary Actors:** User

**Goal in Context:** To interact with the community.

**Preconditions:** User logged in.

**Main Scenario:**

- a. User creates a post.
- b. Others like/comment.
- c. Admin moderates content if needed.

**Priority:** Expected

## **6. Chat Module**

**i. Use case: real-time chat**

**Primary Actors:**User

**Goal in Context:** To communicate with other users.

**Preconditions:** Both users logged in.

**Main Scenario:**

- a. User sends message.
- b. Receiver gets notification.

**Priority:** Expected

## **7. Environmental Impact Dashboard**

**i. Use Case: Track Reused / Donated Books**

**Primary Actors:** User, Admin

**Goal in Context:** To measure environmental impact.

**Preconditions:** Transactions completed.

**Main Scenario:**

- a. System logs reused/donated books.
- b. Calculates paper saved.
- c. Displays statistics.

**Priority:** Expected

## **8. Payments & Administration**

### **i. Use Case: Make Payment**

**Primary Actors:** User

**Goal in Context:** To complete paid transactions.

**Preconditions:** Payment gateway available.

**Main Scenario:**

- a. User selects payment option.
- b. Payment verified.
- c. Transaction recorded.

**Priority:** Essential

### **ii. Use Case: Manage Users / Libraries / Reports**

**Primary Actors:** Admin

**Goal in Context:** To administer the system.

**Preconditions:** Admin logged in.

**Main Scenario:**

- a. Admin views system data.
- b. Manages users, libraries, books.
- c. Generates reports.

**Priority:** Essential

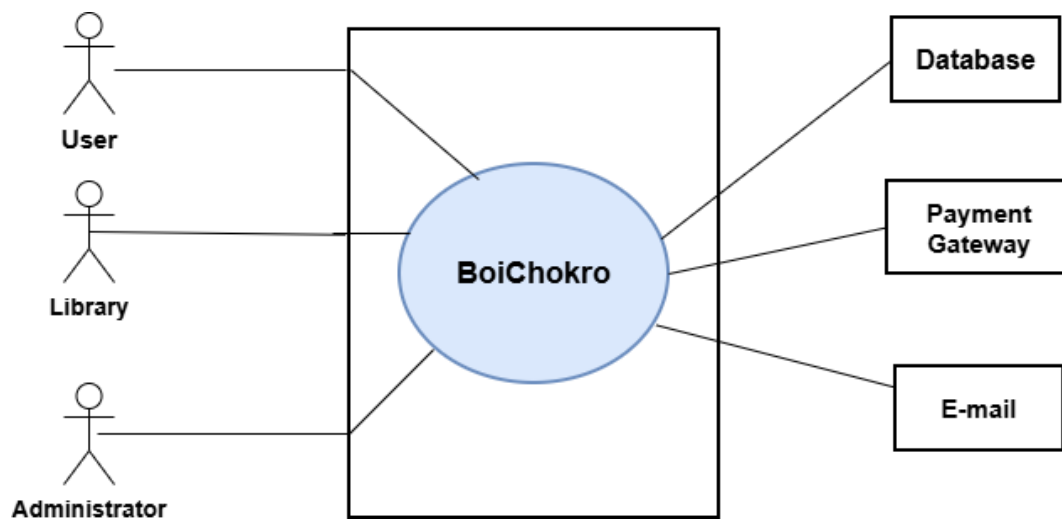
## 4.3.2 Use Case Diagram

**Level:** 0

**Name:** *BoiChokro*

**Primary Actor:** User, Library, Administration

**Secondary Actor:** Database, Payment Gate way, Email



**Figure: Level 0 - BoiChokro**

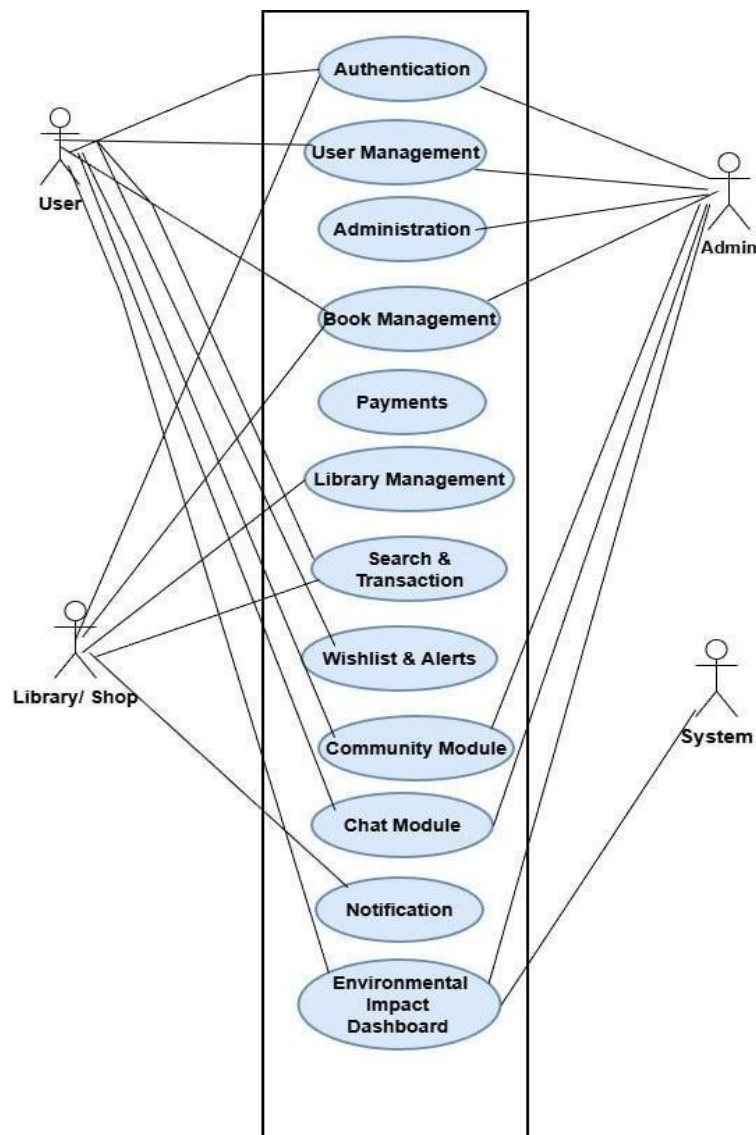


**Level: 1**

**Name:** User-Book-Library management, Administration, Payment etc.

**Primary Actor:** User, Library/Shop, Admin

**Secondary Actor:** System



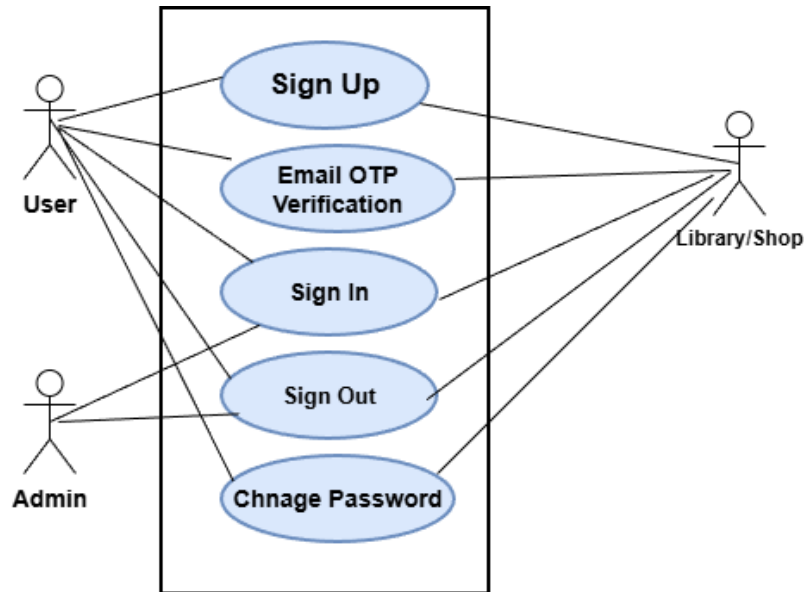
**Figure: Level 1 - BoiChokro**

**Level: 2.1**

**Name:** Authentication

**Primary Actor:** User, Library/Shop, Admin

**Secondary Actor:**

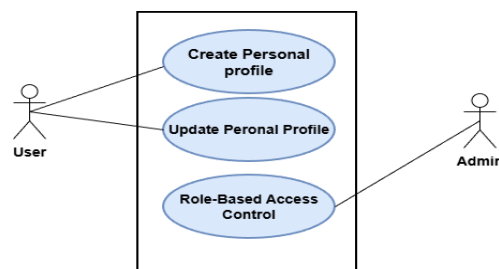


**Figure: Level 2.1 - Authentication**

**Level: 2.2**

**Name:** User management

**Primary Actor:** User, Admin

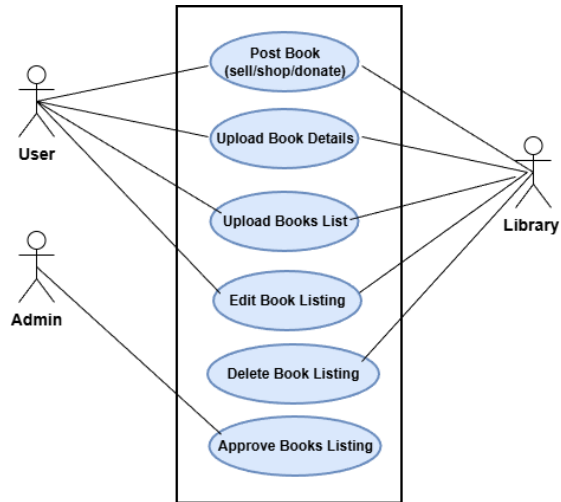


**Figure: Level 2.2 - User Management**

**Level: 2.3**

**Name:** Book management

**Primary Actor:** User, Library, Admin

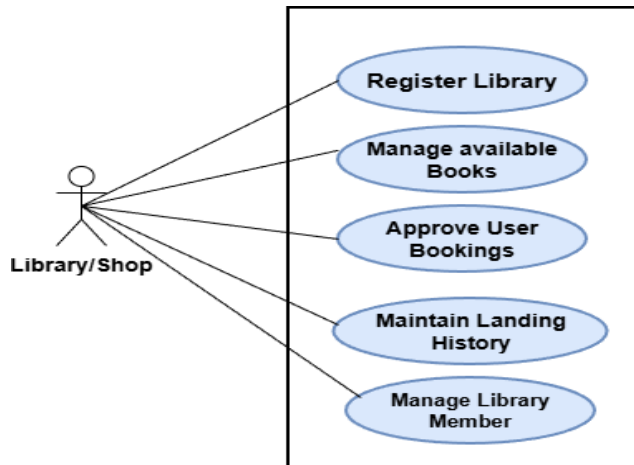


**Figure: Level 2.3 - Book Management**

**Level: 2.4**

**Name:** Library/Shop management

**Primary Actor:** Library/Shop

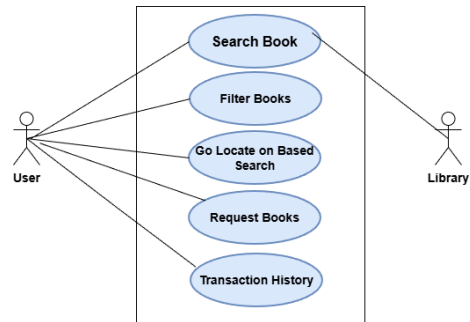


**Figure: Level 2.4 - Library/Shop Management**

**Level: 2.5**

**Name:** Search & transaction

**Primary Actor:** User, Library

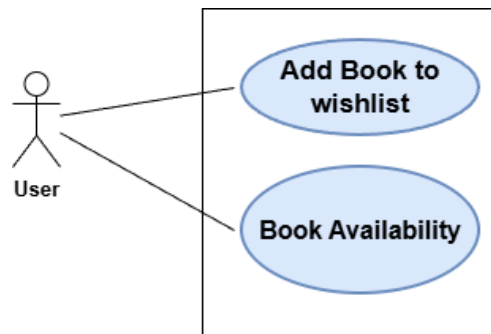


**Figure: Level 2.5 - Search & Transaction**

**Level: 2.6**

**Name:** Wishlist & Alerts

**Primary Actor:** User

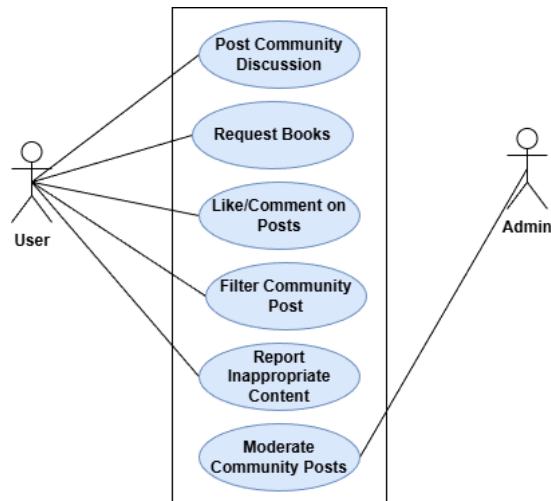


**Figure: Level 2.6 - Wishlist & Alerts**

**Level: 2.7**

**Name:** Community Module

**Primary Actor:** User, Admin

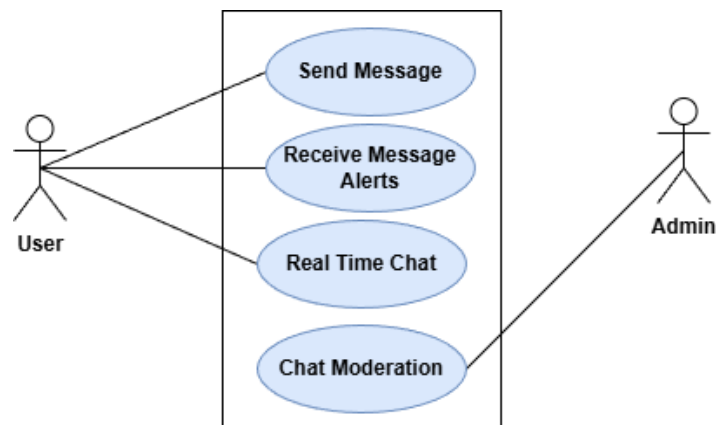


**Figure: Level 2.7 - Community Model**

**Level: 2.8**

**Name:** Chat Module

**Primary Actor:** User, Admin

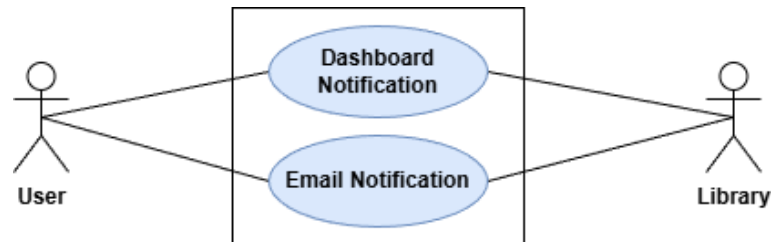


**Figure: Level 2.8 - Chat Module**

**Level:** 2.9

**Name:** Notification Module

**Primary Actor:** User, Library



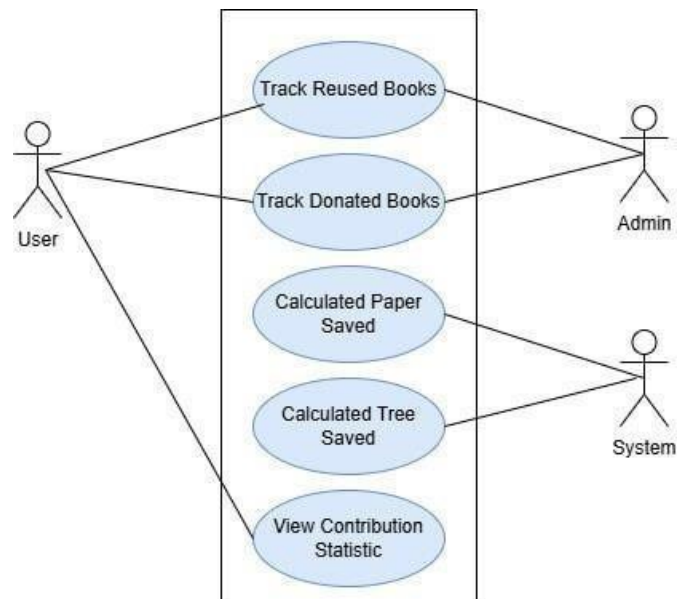
**Figure: Level 2.9 - Notification Module**

**Level:** 2.10

**Name:** Environmental Impact

**Primary Actor:** User, Admin

**Secondary Actor:** System



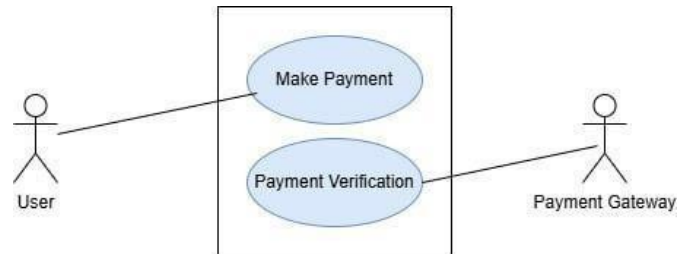
**Figure: Level 2.10 - Environmental Impact Dashboard**

**Level:** 2.11

**Name:** Payments

**Primary Actor:** User

**Secondary Actor:** Payment Gateway

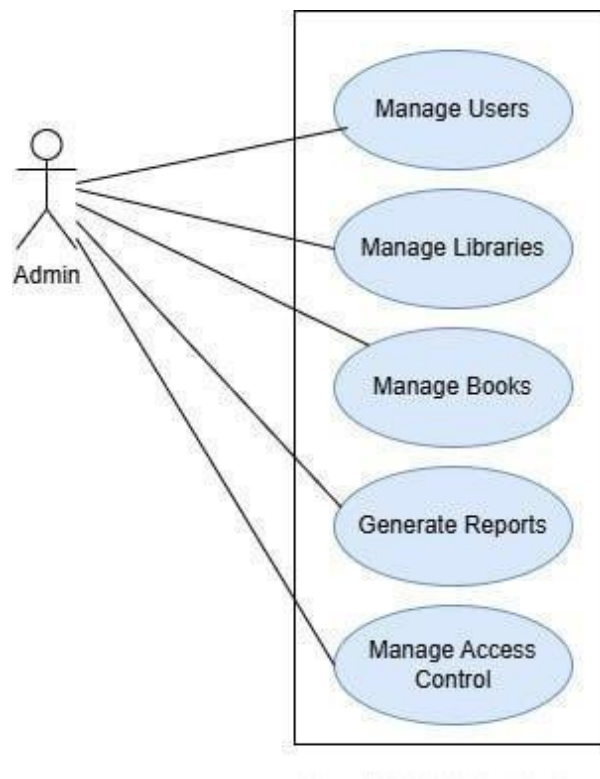


**Figure: Level 2.11 - Payments**

**Level:** 2.12

**Name:** Administration

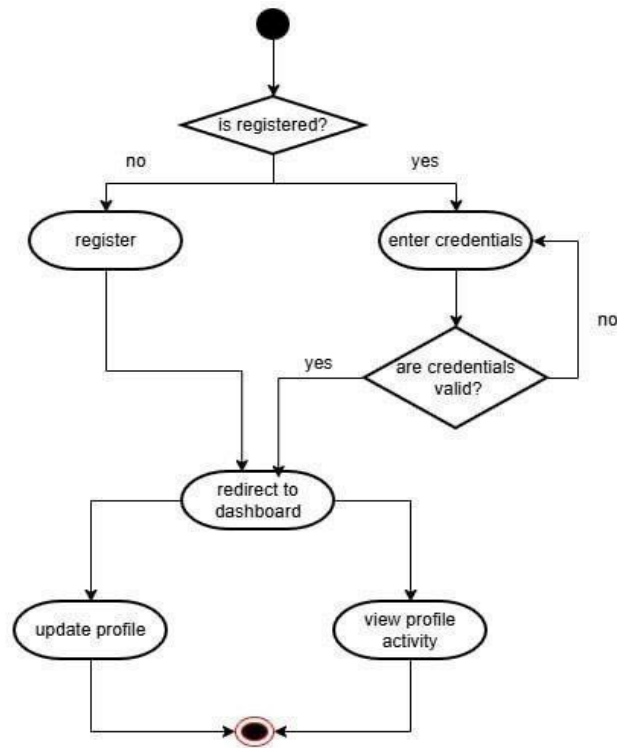
**Primary Actor:** Admin



**Figure: Level 2.12 - Administration**

### 4.3.3 Activity & Swimlane Diagram

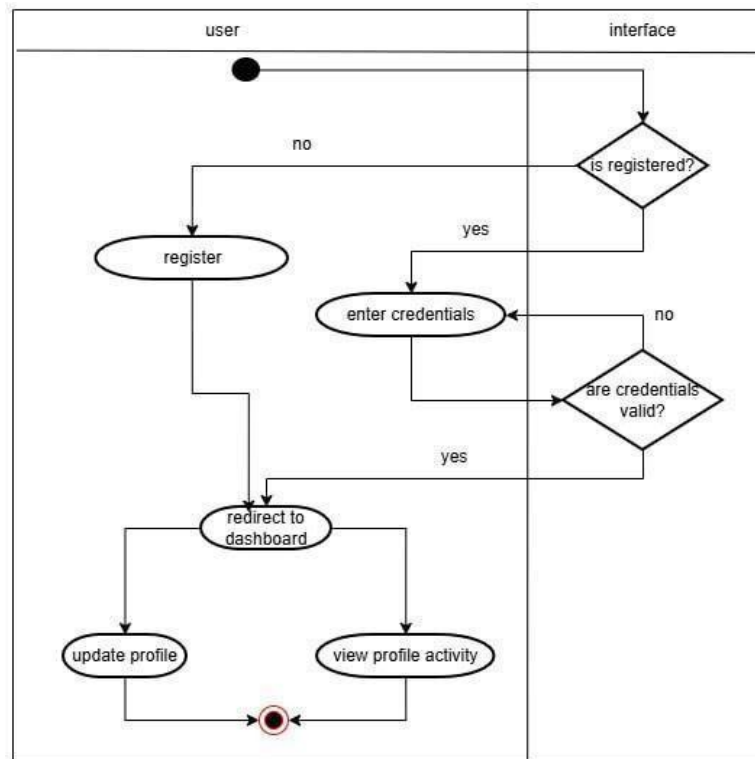
**Use case 1: Authentication**  
Activity Diagram:



**Figure: Authentication**



## Swimlane Diagram:

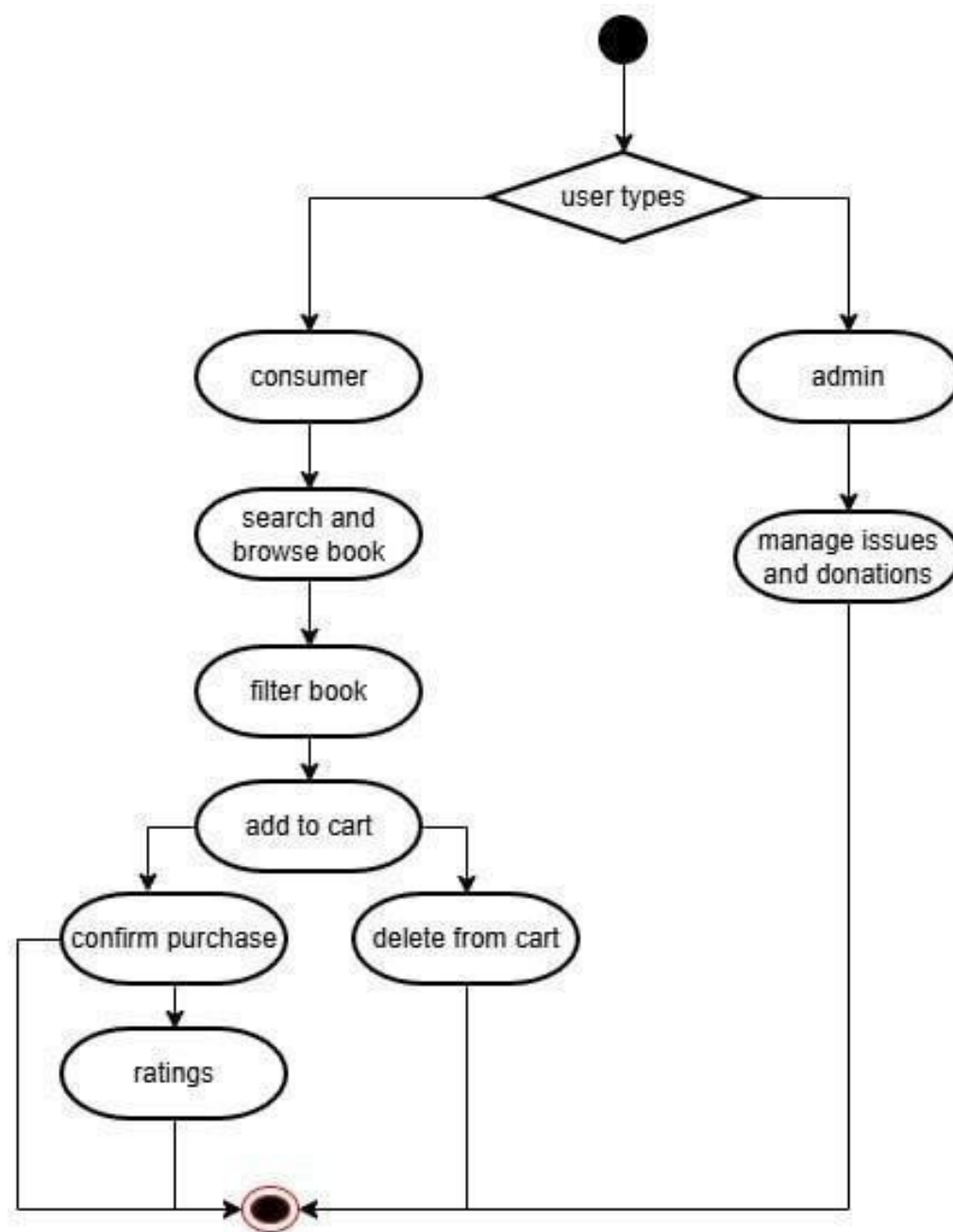


**Figure: Authentication**

The following diagram shows the user log-in process, which runs verification of registration and credential validity and allows access to the dashboard functionality, such as profile updates and activities.

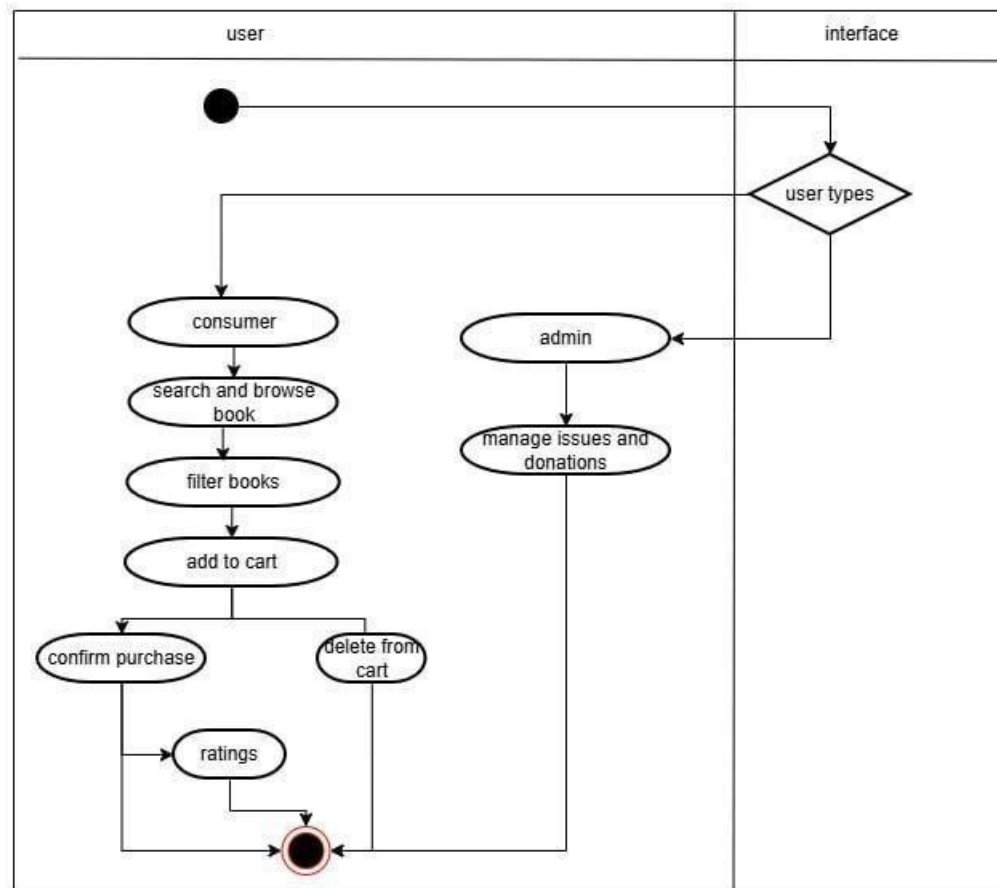
**Name:** User Management

**Reference:** Use Case 2.2



**Figure: User Management**

## Swimlane Diagram:

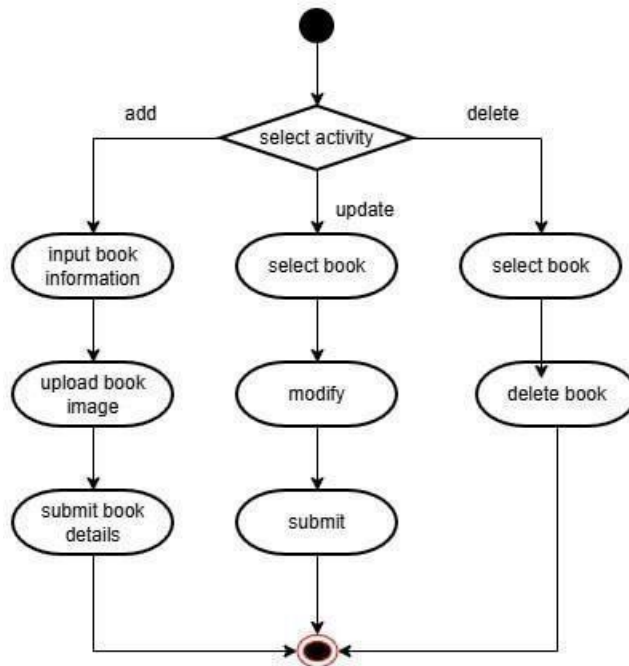


**Figure: User Management**

It differentiates the consumer and the admin role: consumers have the ability to view, filter, and buy books, whereas the admins deal with problems and donation, both culminating in a specific exit point.

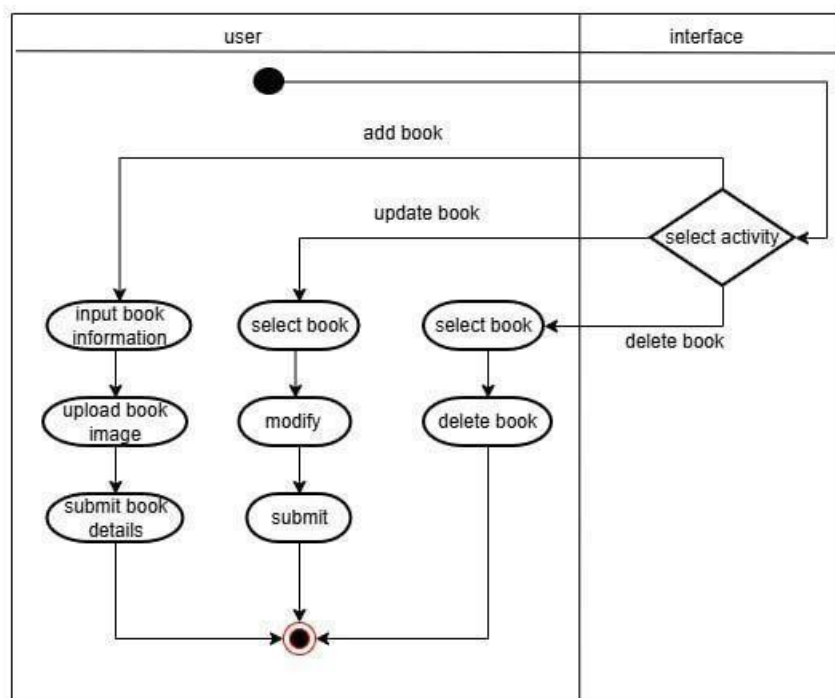
**Name:** Book Management

**Reference:** Use Case 2.3



**Figure: Book Management**

## Swimlane Diagram

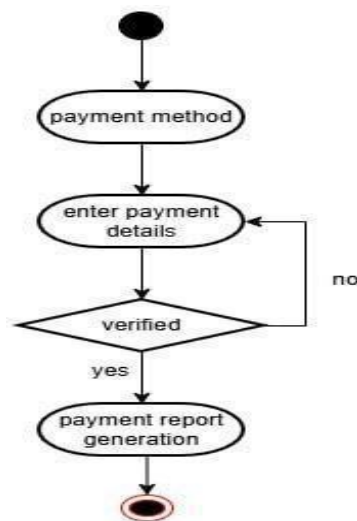


## Figure: Book Management

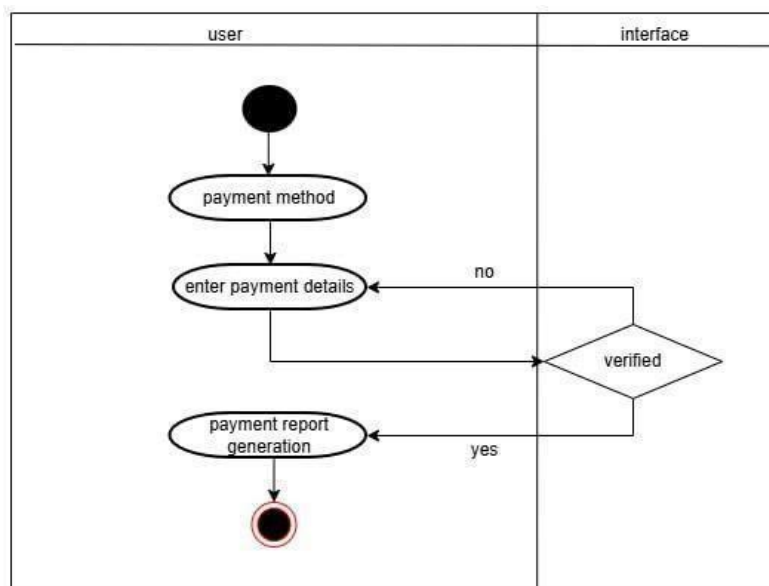
The flow supports the process of adding, updating, or deleting books and lets the user choose an activity and execute specific steps to complete the process that leads to a single process termination.

**Name: Payment**

**Reference: Use Case 2.11**



**Swimlane Diagram:**



This diagram serves to represent the payment workflow, when a user selects a

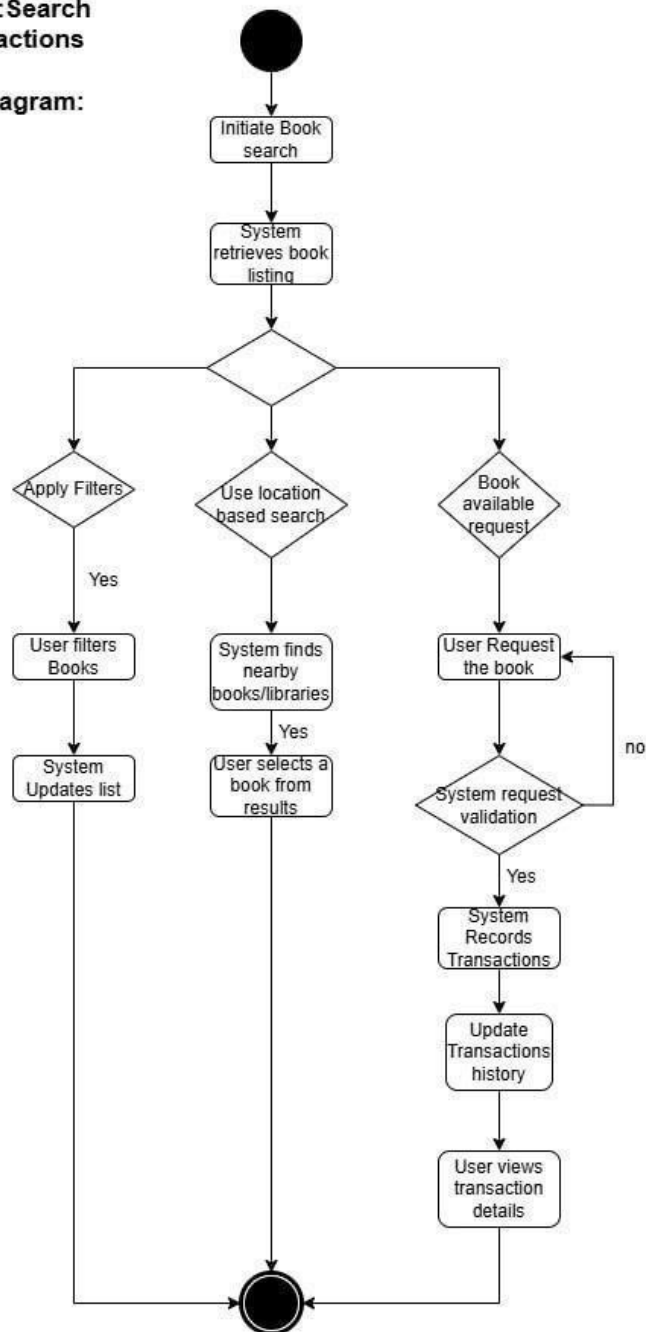
method, fill out the details and goes to report generation in a case of successful verification.

**Name:** Search & Transaction

**Reference:** Use Case 2.5

**Use Case :Search  
and Transactions**

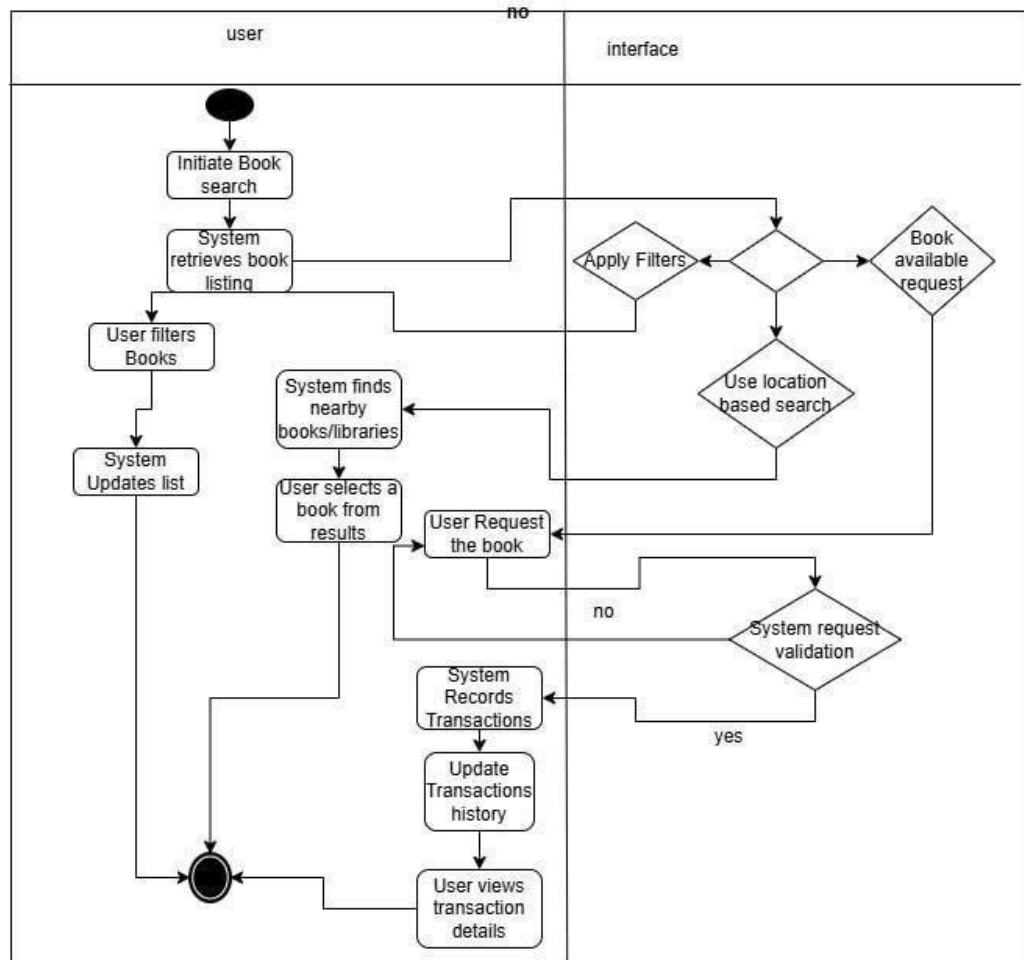
**Activity Diagram:**



## Swimlane Diagram

Use Case  
: Search and  
Transactions

Swimlane  
Diagram:



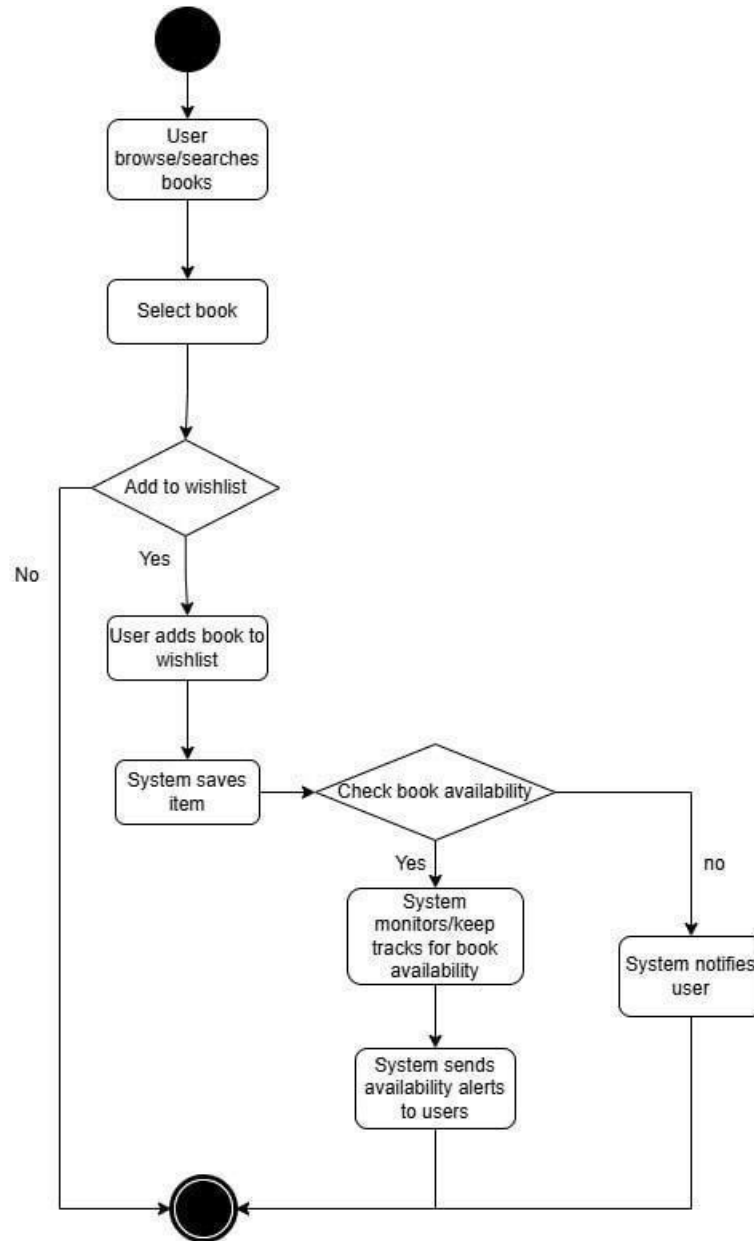
The diagram illustrates the search and transaction process of the book search and transaction, including filtering, location search, availability check, and recording of transactions, and finally viewing of the transaction details.

**Name:** Wishlist & Alerts

**Reference:** Use Case 2.6

**Use Case :Wishlist and Alerts**

**Activity Diagram:**

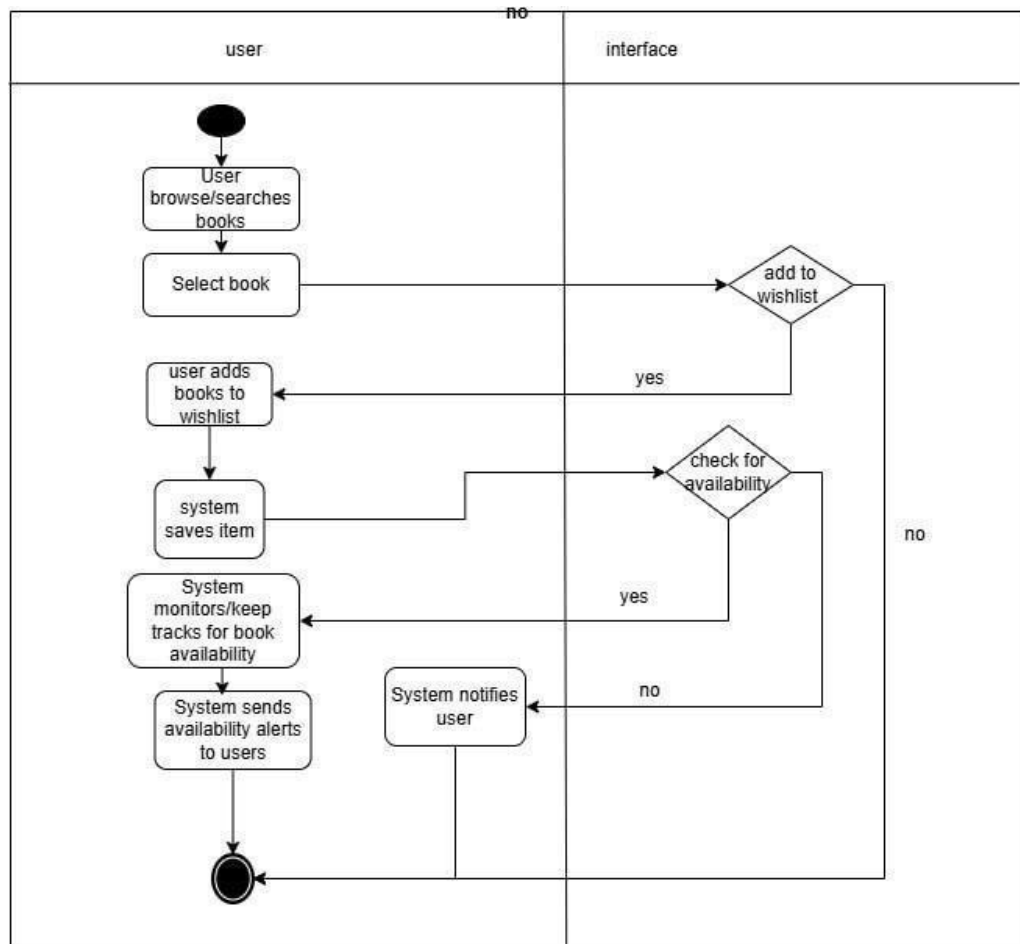




## Swimlane Diagram

Use Case :  
wishlist and  
alerts

Swimlane  
Diagram:



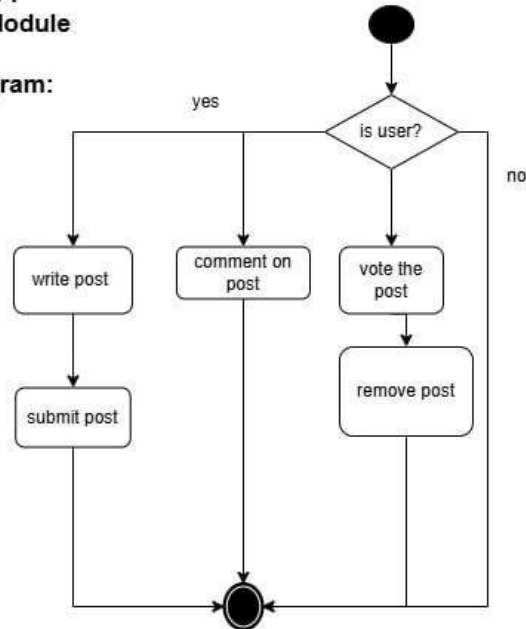
It represents that users add the books to their wishlist, and the system monitors the availability of the book sending alerts or notifications depending on the status of the book.

**Name:** Community Model

**Reference:** Use Case 2.7

**Use Case :**  
**Community Module**

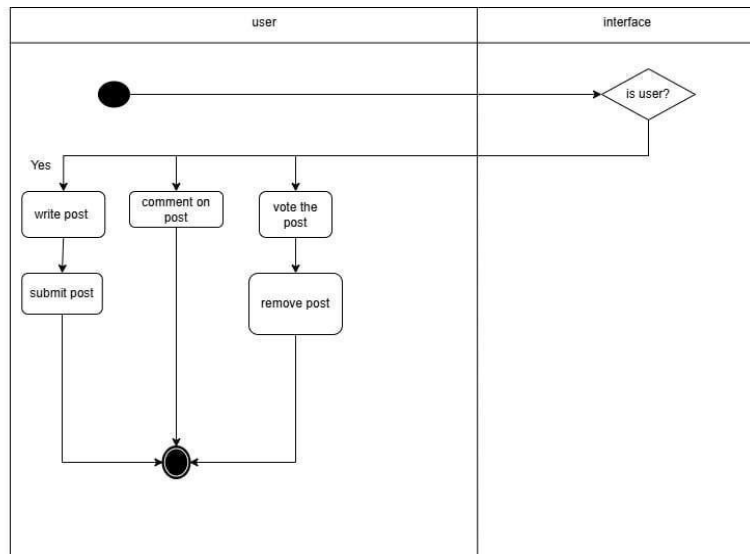
**Activity Diagram:**



## Swimlane Diagram

**Use Case :**  
**community**

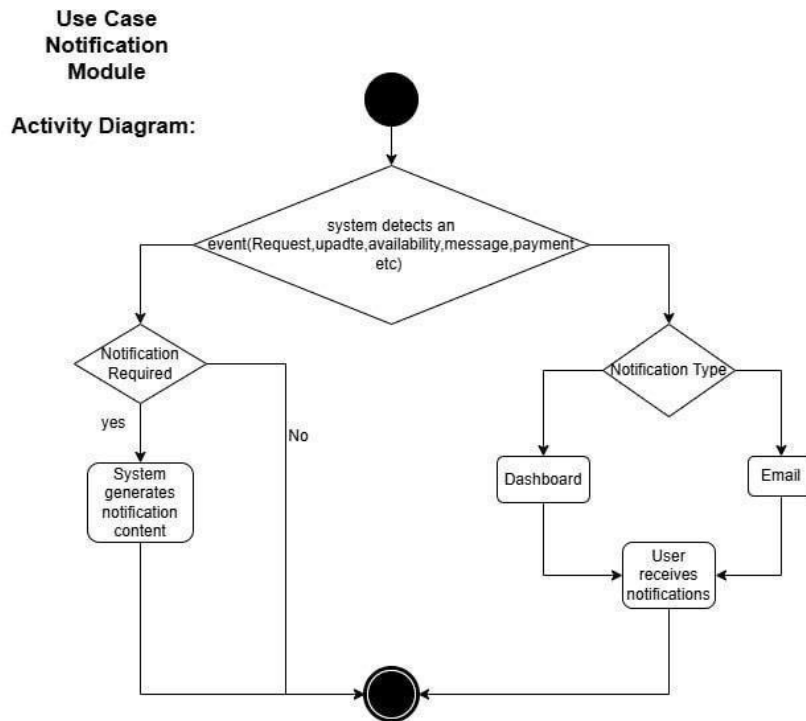
**Swimlane Diagram:**



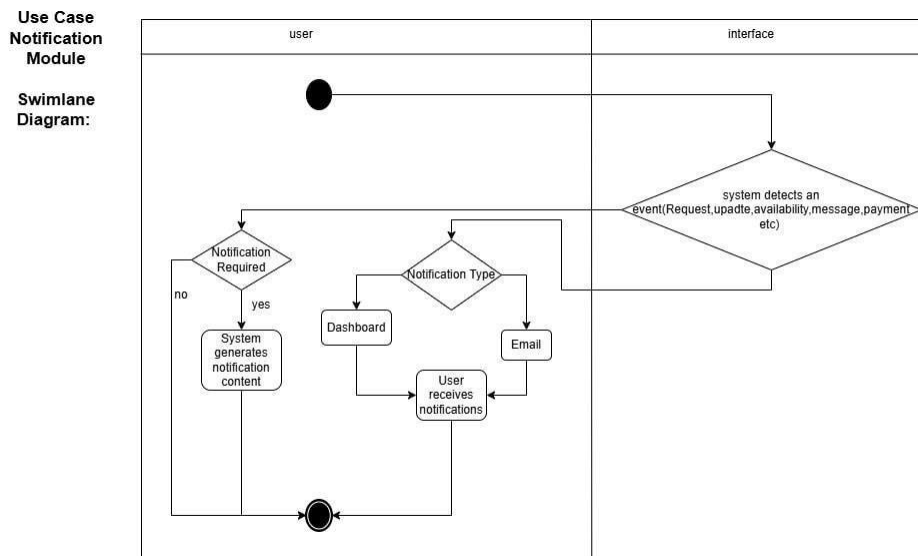
The flow enables authenticated users to write, comment, vote and remove posts and unauthenticated users are unable to interact.

**Name:** Notification Module

## Reference: Use Case 2.9



## Swimlane Diagram:

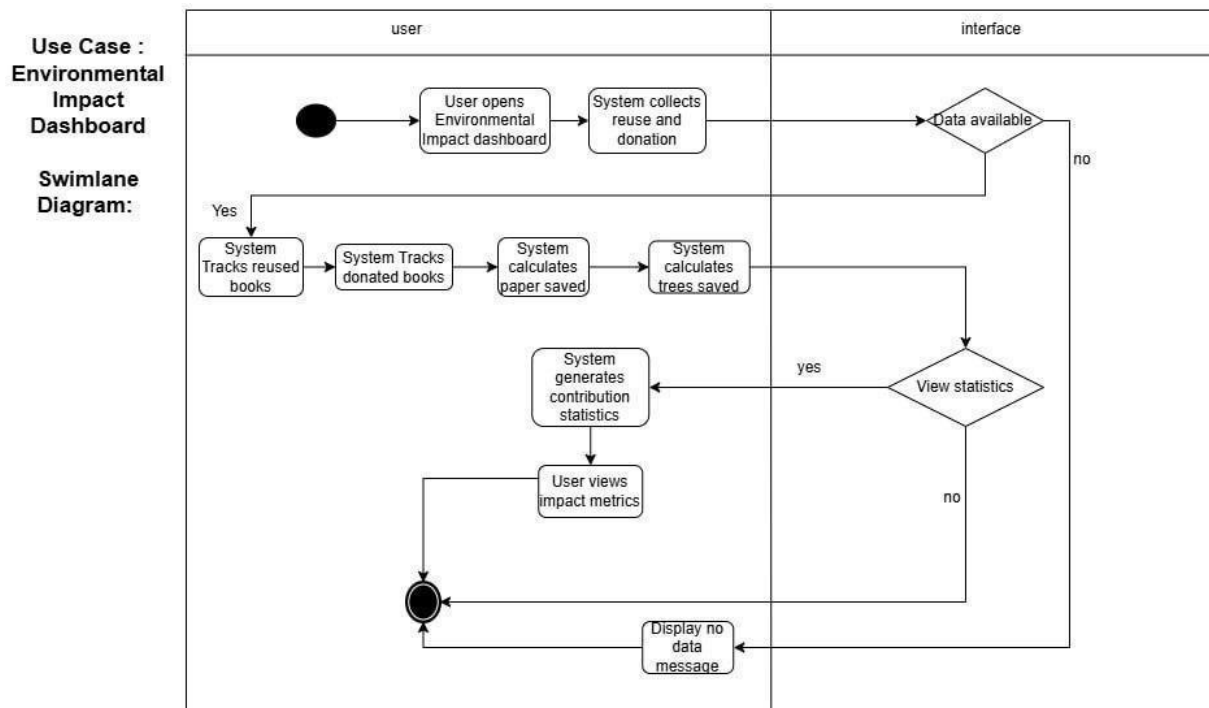


This flow chart describes the process by which the system identifies events, determines whether or not the notifications are required, creates the content, and transmits the content through a dashboard or e-mail.

**Name:** Environmental Impact Dashboard

**Reference:** Use Case 2.10

## Swimlane Diagram:



This diagram enables the system to gather data regarding the reuse and donation data and show the impact metrics to the user in case the data is present.

## 4.4 Data Modeling

### 4.4.1 Entity Relation (ER) Diagram

#### Step 1: Identified Entities

The following entities are identified from the system model:

- a) User
- b) Role
- c) Library
- d) Book
- e) BookListing
- f) Transaction
- g) Payment
- h) Wishlist
- i) CommunityPost
- j) Comment
- k) ChatMessage
- l) Notification
- m) EnvironmentalImpact

#### Step 2: Attributes and Primary Keys of Each Entity

<b>User</b>	<u>user_id</u> name password phone role_id (FK) is_verified, created_at)
<b>Role</b>	<u>role_id</u> role_name
<b>Library</b>	<u>library_id</u> user_id (FK) library_name email

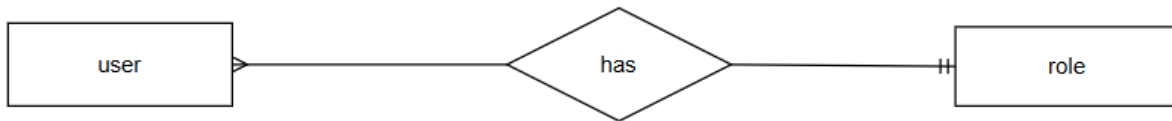
	phone address status created_at
<b>Book</b>	<u>book_id</u> title author category isbn condition
<b>BookListing</b>	<u>listing_id</u> book_id (FK) user_id( FK) library_id( FK) listing_type price availability_status approved_by_admin created_at
<b>Transaction</b>	<u>transaction_id</u> listing_id (FK) buyer_id( FK) transaction_type transaction_date status
<b>Payment</b>	<u>payment_id</u> transaction_id( FK) amount payment_method payment_status payment_date
<b>Wishlist</b>	<u>wishlist_id</u> user_id (FK) book_id( FK) created_at

<b>CommunityPost</b>	<u>post_id</u> user_id (FK) content post_type created_at status
<b>Comment</b>	<u>comment_id</u> post_id(FK) user_id(FK) comment_text created_at
<b>ChatMessage</b>	<u>message_id</u> sender_id(FK) receiver_id(FK) message_text sent_at status
<b>Notification</b>	<u>notification_id</u> user_id(FK) message, notification_type, is_read, created_at
<b>EnvironmentalImpact</b>	<u>impact_id</u> user_id(FK) reused_books donated_books paper_saved trees_saved, last_updated

## Cardinality and Relationship:

### 1. Role – User

- **Cardinality:** One-to-Many (1 : M)



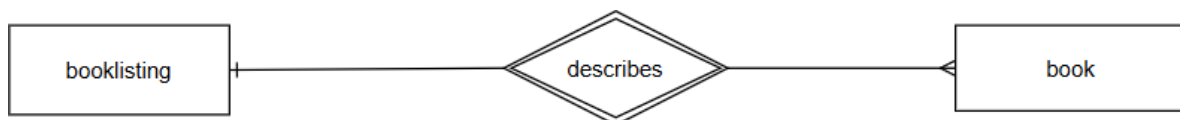
### 2. User – BookListing

- **Cardinality:** One-to-Many (1 : M)



### 3. Book – BookListing

- **Cardinality:** One-to-Many (1 : M)

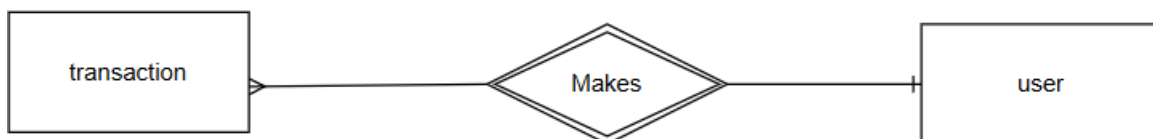


### 4. BookListing – Transaction

- **Cardinality:** One-to-Many (1 : M)

### 5. User – Transaction

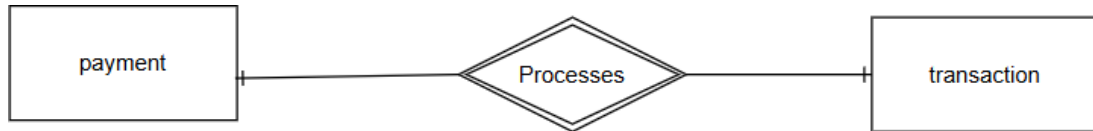
- **Cardinality:** One-to-Many (1 : M)





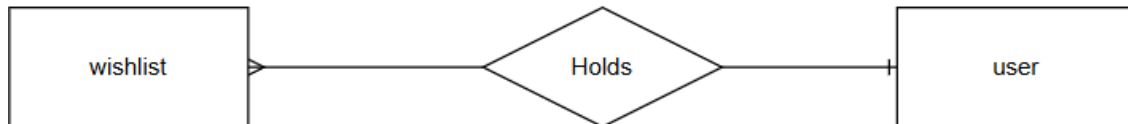
## 6. Transaction – Payment

- **Cardinality:** One-to-One (1 : 1)



## 7. User – Wishlist

- **Cardinality:** One-to-Many (1 : M)



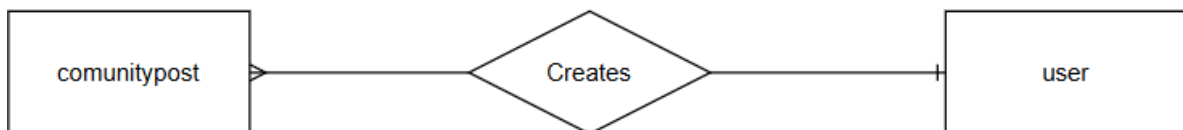
## 8. Wishlist – Book

- **Cardinality:** Many-to-One (M : 1)



## 9. User – CommunityPost

- **Cardinality:** One-to-Many (1 : M)



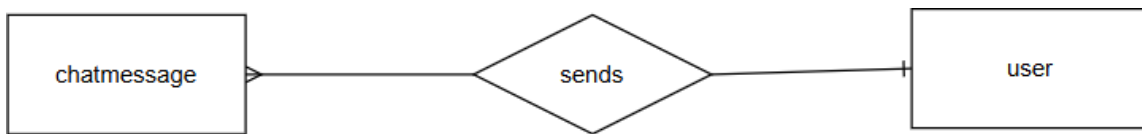
## 10. CommunityPost – Comment

- **Cardinality:** One-to-Many (1 : M)



### 11. User – ChatMessage

- **Cardinality:** One-to-Many (1 : M)



### 12. User – Notification

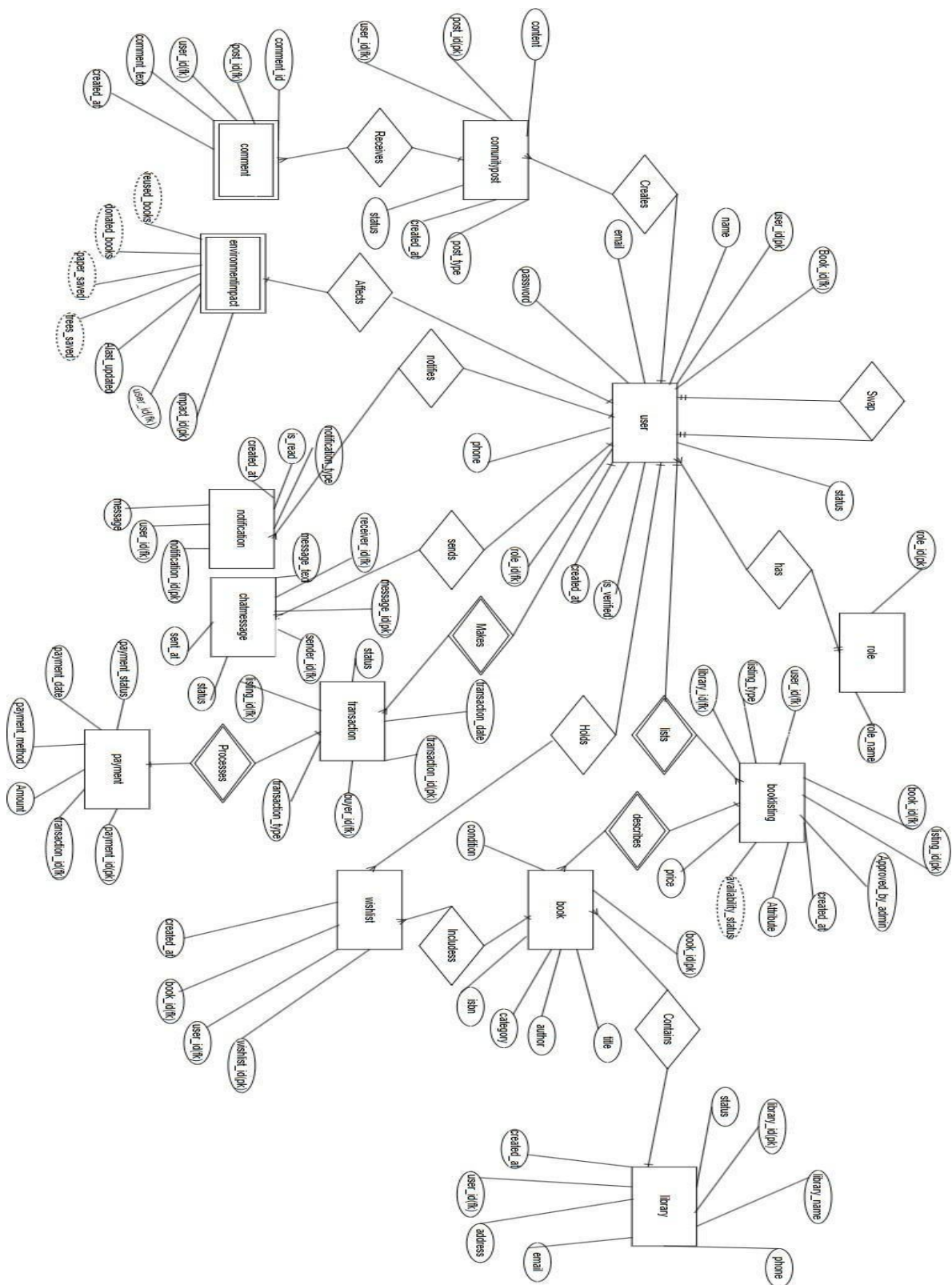
- **Cardinality:** One-to-Many (1 : M)



### 13. User – Environmental impact

- **Cardinality:** One-to-One (1 : 1)





**Figure: ER diagram For BoiChokro**

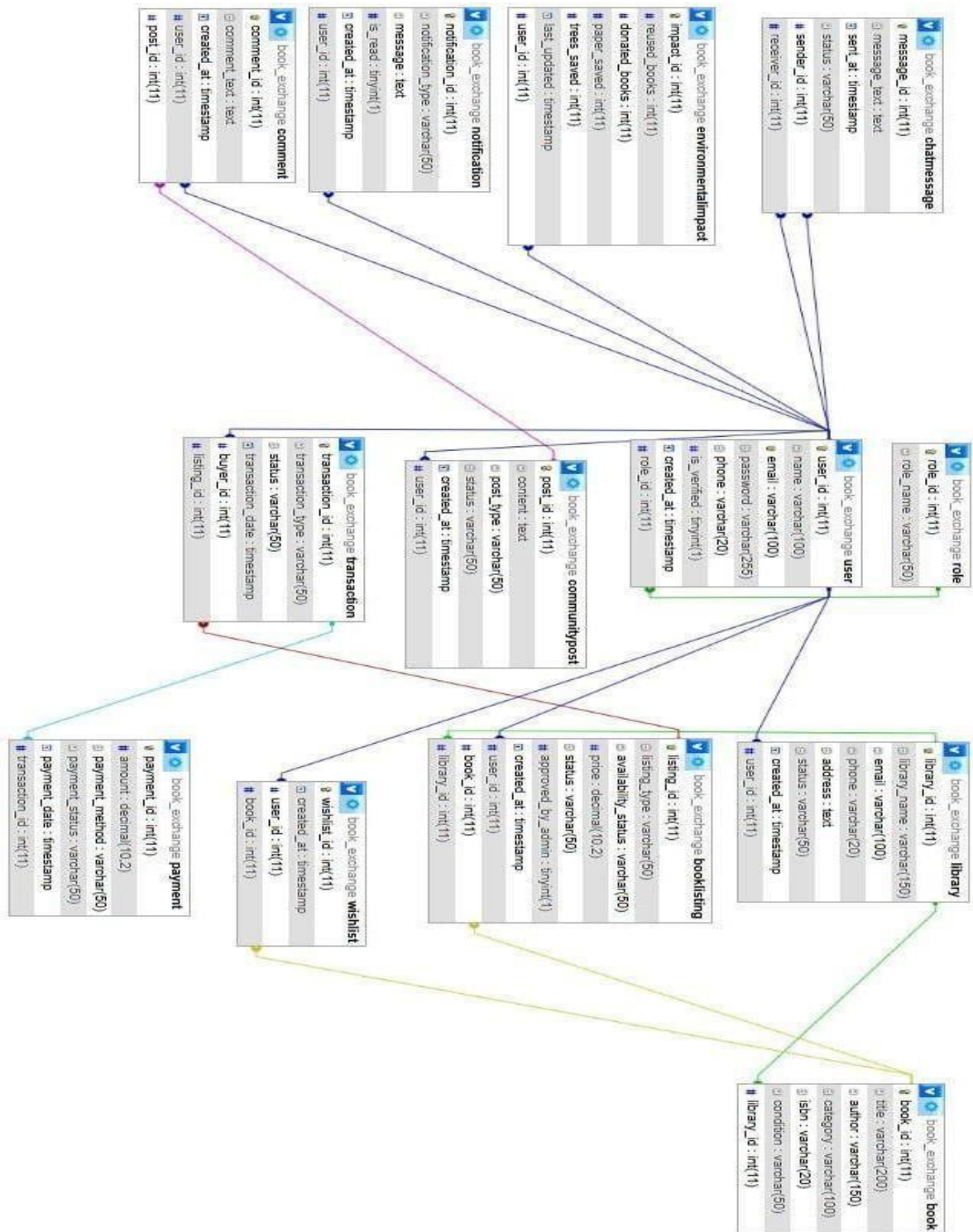


Figure: Schema Diagram for BoiChokro

## 4.5 Class-Based Modeling

Class-based modeling identifies the classes, attributes and operations required to represent the structure of the system. It helps transform requirements into an object-oriented design by identifying system entities and defining their responsibilities.

### 4.5.1 Identify Class and Attributes

#### Step-1: Identifying and categorize all nouns

All nouns were extracted from the problem statement, requirements, and use-case descriptions of the Boichokro system and categorized as follows:

**External Entities :** User, Library, Admin, Payment Gateway, School, Donor.

**Things :** Book, Book Listing, Wishlist, Notification, Message, Review, Rating, Donation, Transaction.

**Occurrences / Events :** Register, Login, Logout, Post Book, Approve Listing, Search, Request Book, Payment, Chat, Donate, Borrow, Return, Generate Report.

**Roles :** Admin, Library Authority, Reader, Student.

**Organizational Units:** Library, Community, Educational Institution.

**Places :** Book Location, Library Location.

**Structures :** System, Server, Database, Web Interface.

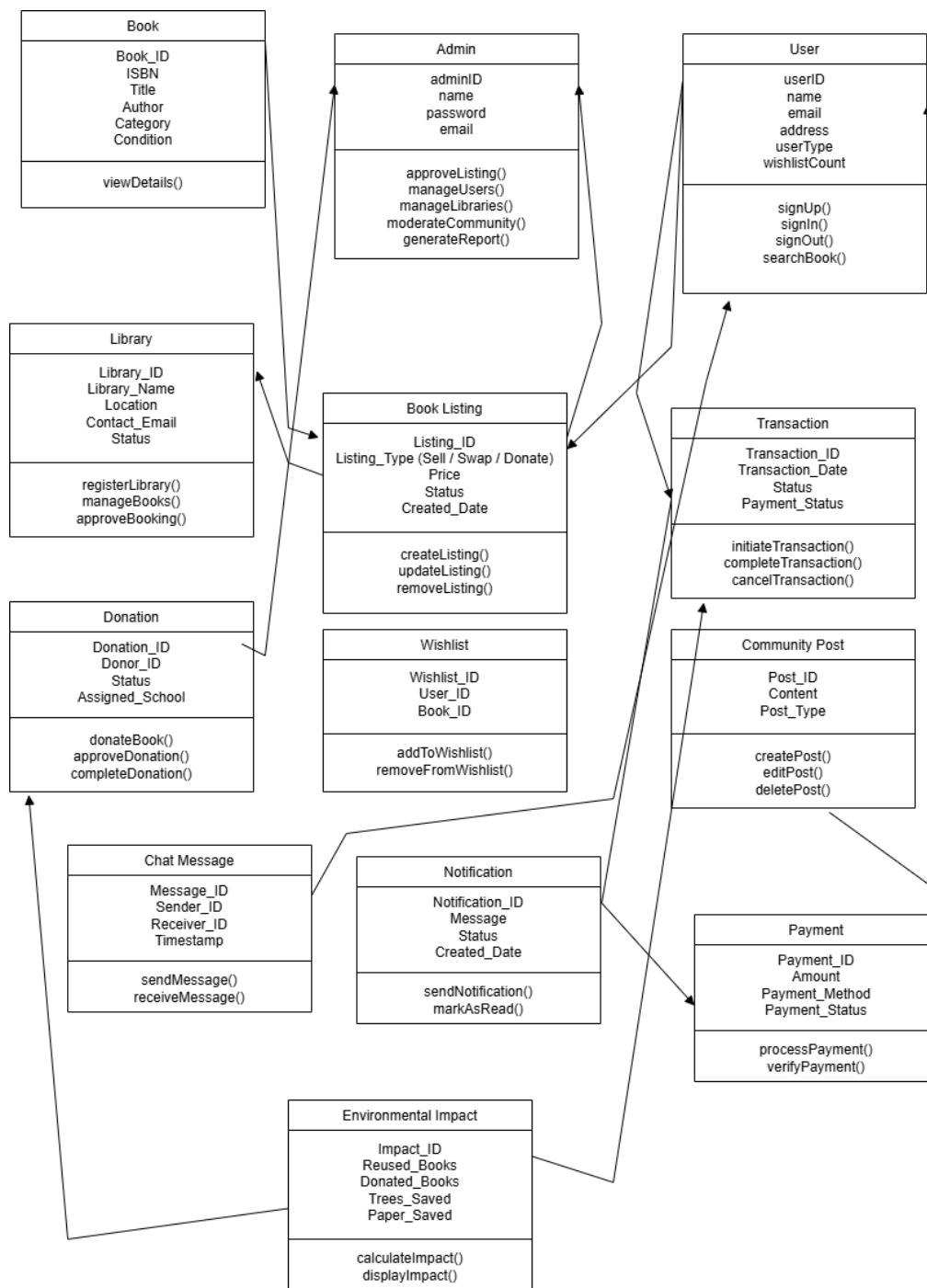
#### Step-2: Selection of Potential Classes

The identified nouns were evaluated using the following selection characteristics:

1. Retained information
2. Needed services
3. Multiple attributes
4. Common attributes
5. Common operations
6. Essential requirements

**Table: Final Class Selection**

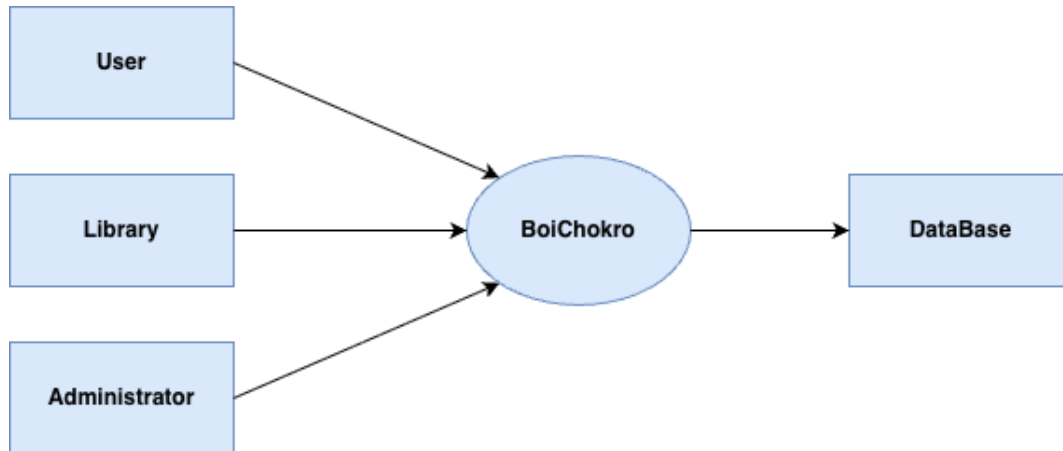
<b>Potential Class</b>	<b>Characteristic Number That Applies</b>
User	Accepted
Admin	Accepted
Library	Accepted
Book	Accepted
Book Listing	Accepted
Transaction	Accepted
Donation	Accepted
Wishlist	Accepted
Community Post	Accepted
Chat Message	Accepted
Notification	Accepted
Payment	Accepted
Environmental Impact	Accepted
Server	Rejected (fails 3)
Interface	Rejected (fails 1,4,5)
Button	Rejected (fails 1,3,5,6)
Location	Rejected (fails 3)
Search	Rejected (fails 3)
Report	Rejected (fails 1,3,6)



**Figure: Class Diagram for Boichokro**

## 4.6. Flow-Oriented Modeling

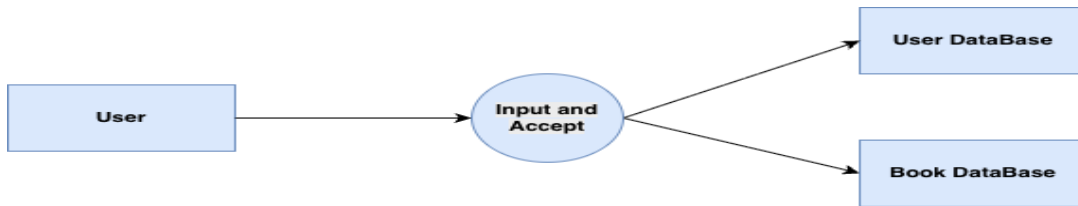
### 4.8.1 Data Flow Diagram



#### Level 0 for BoiChokro

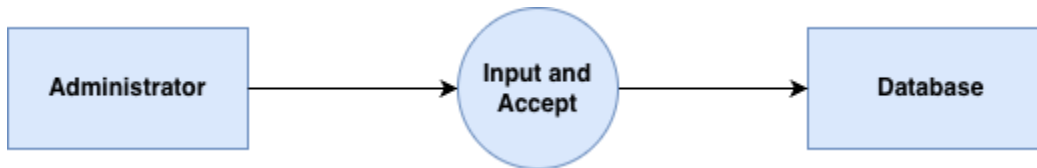
This schematic illustrates the interaction point of users, libraries and administrators with the central BoiChokro system, which receives the inputs of the users and stores the data in a single database





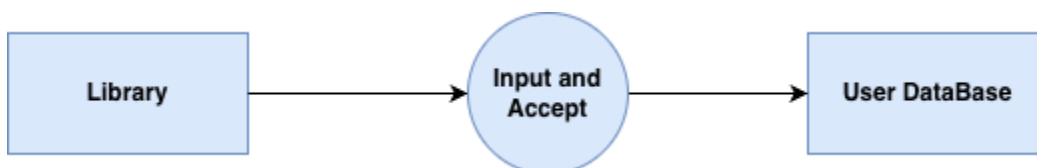
### **Level 1.1(user) For BoiChokro**

It shows the process of accepting user data into the system and directing it to User and Book databases to be stored and processed.



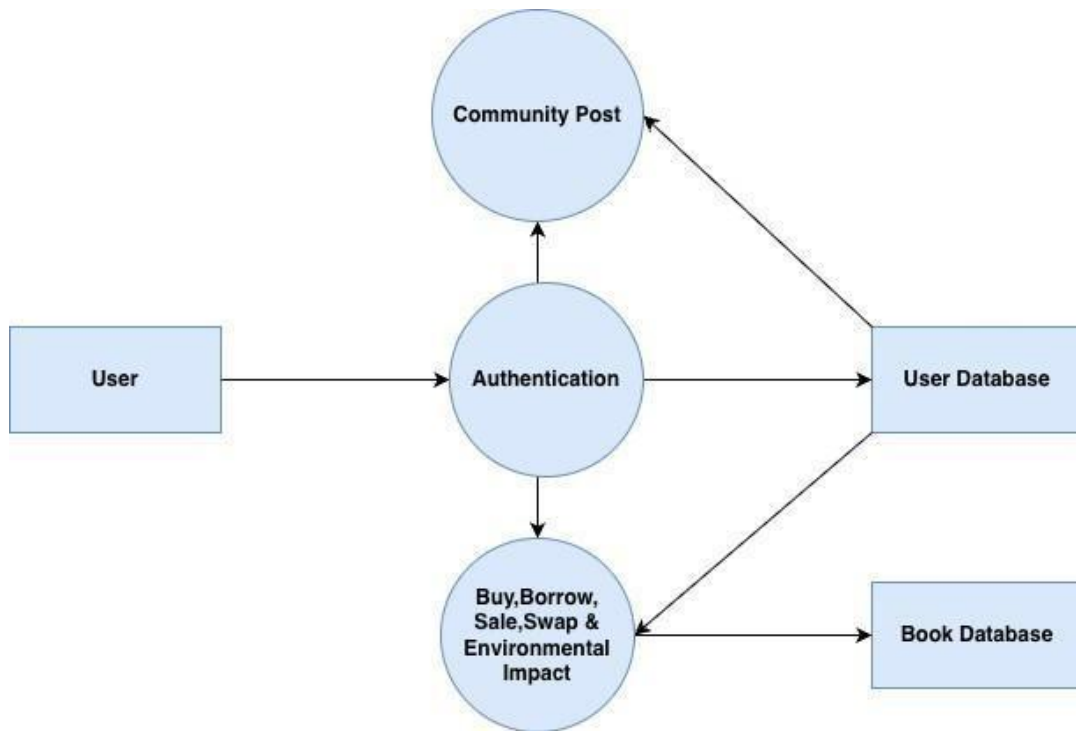
### **Level 1.2(Administrator) For BoiChokro**

The flow indicates the entry of data by the administrators into the system, and the information will further be processed and stored in the central database.



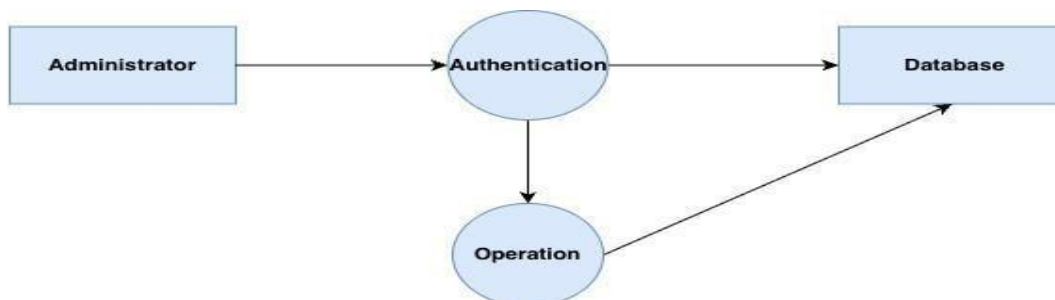
### **Level 1.3 (Library) For BoiChokro**

The diagram describes how the libraries provide their information to the system via the input module and this information is collected by the system and reflected in the User database.



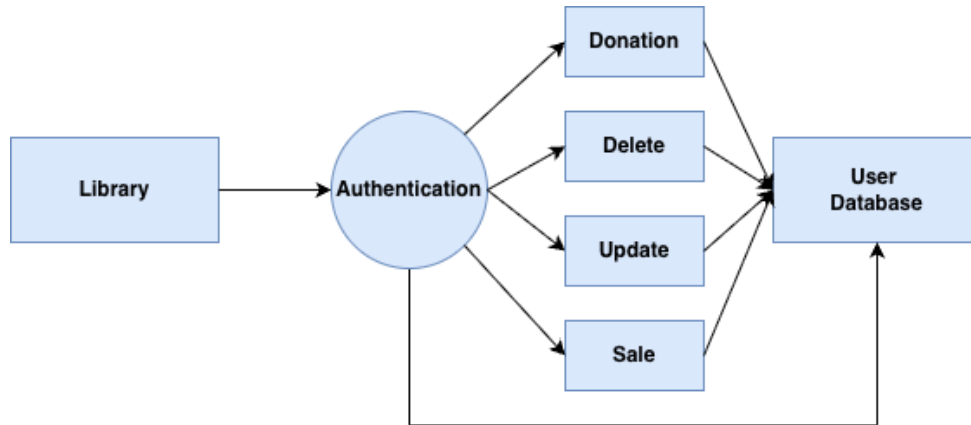
### Level 2.1(User) For BoiChokro

This chart indicates how the registered users will use posts in the community and access the facilities of books such as purchase, loan, and environmental management, and the data flow will be between the user and the book databases.



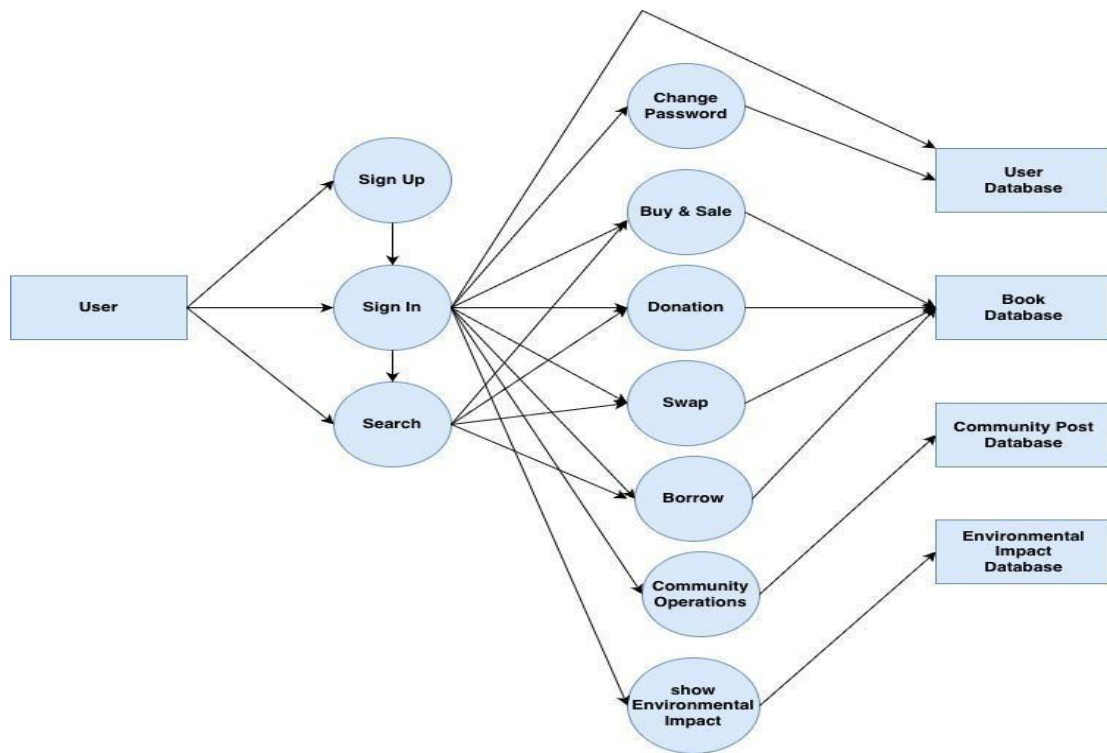
### Level 2.2 (Administrator) For BoiChokro

This describes the type of activity that the administrators perform to gain access to the system database, conducting operations and interacting with it in order to perform administrative activities.



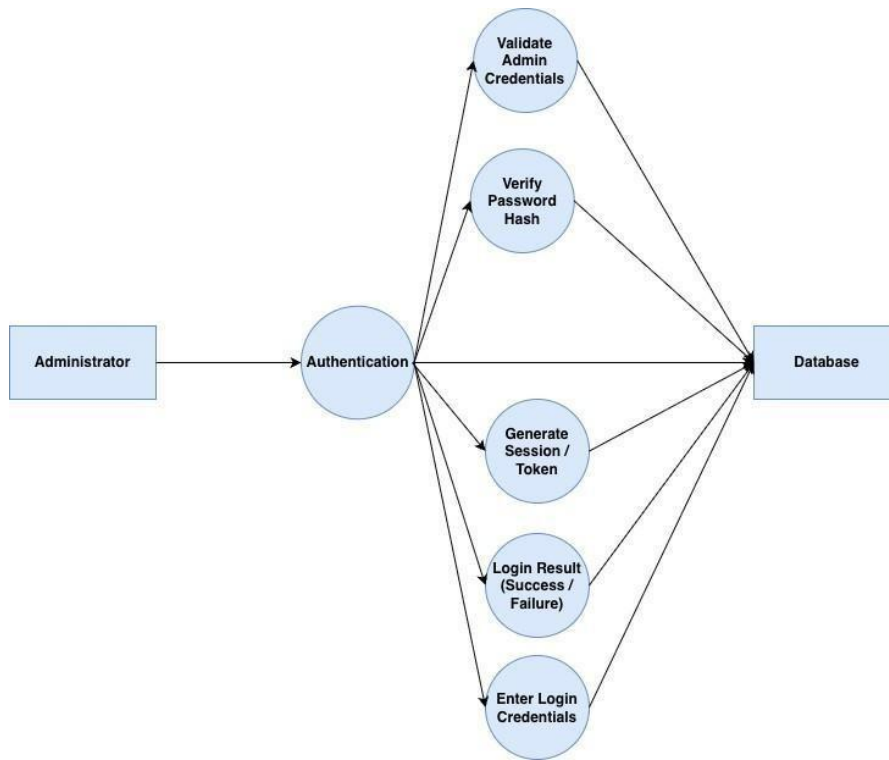
### **Level 2.3 (library) For BoiChokro**

Here, libraries verify and execute operations such as donation, deletion, update and sale, which all update user database.



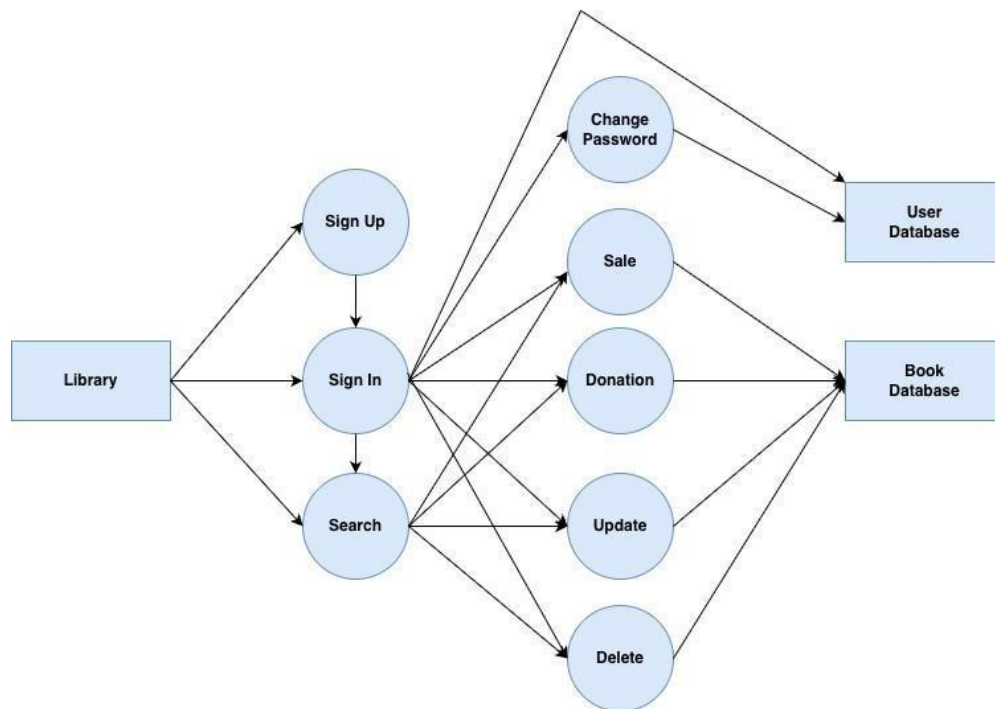
### Level 3.1 (User) For BoiChokro

This chart outlines user interaction during the sign-up and sign-in to accessing functions such as booking transactions, community posts, and environmental impact with the data entering into databases accordingly.



### **Level 3.2 (Administrator) For BoiChokro**

It elaborates on the process of authentication of the administrator, credential validation, password hashing and generation of a session, which are recorded in the system database.



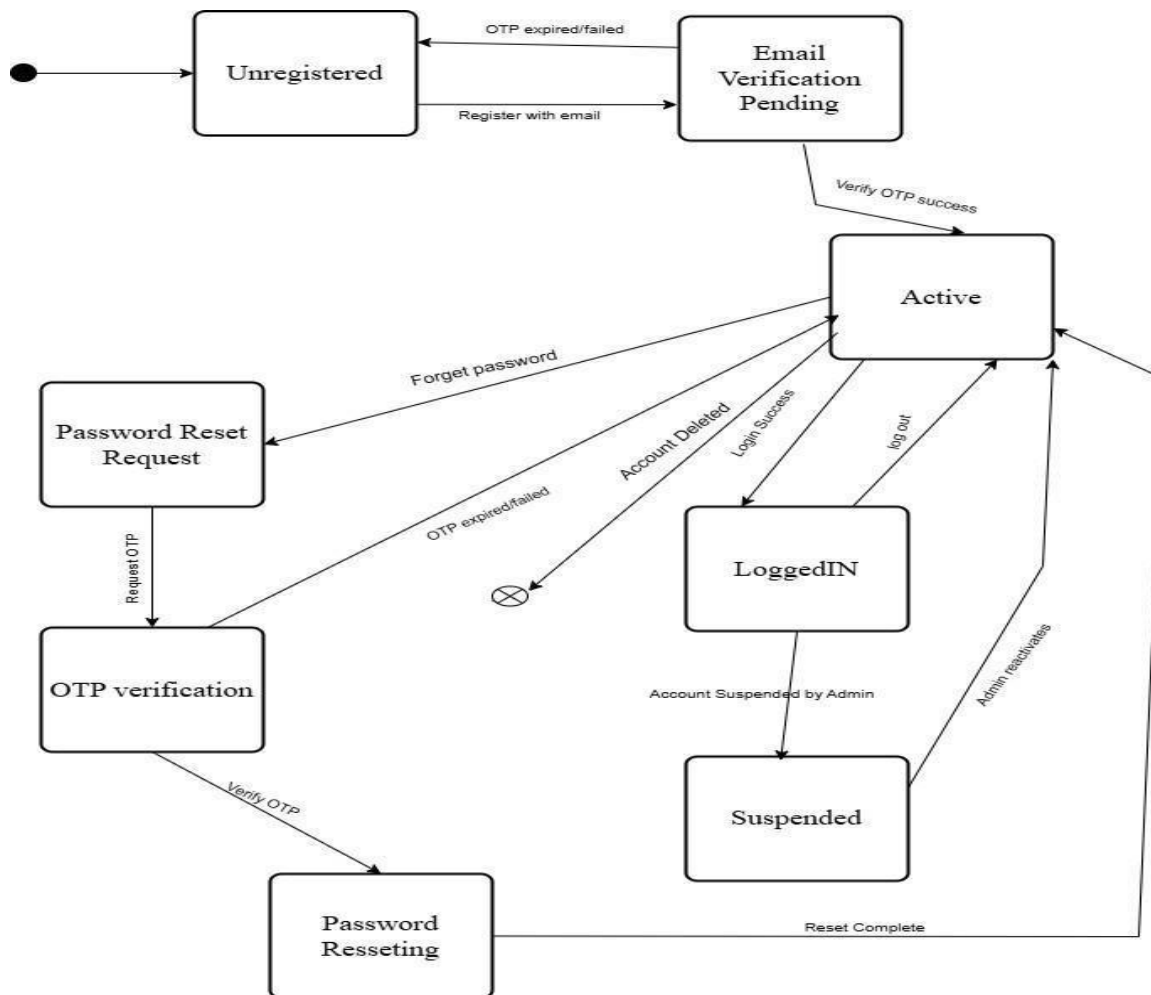
### **Level 3.3 (Library) For BoiChokro**

This flow depicts the operations of authenticating and performing such operations as sale, donation, update, and deletion, which interact with users and book database.

## 4.7. Behavioral Modeling

### 4.7.1 State Transition Diagram

Name: User Authentication

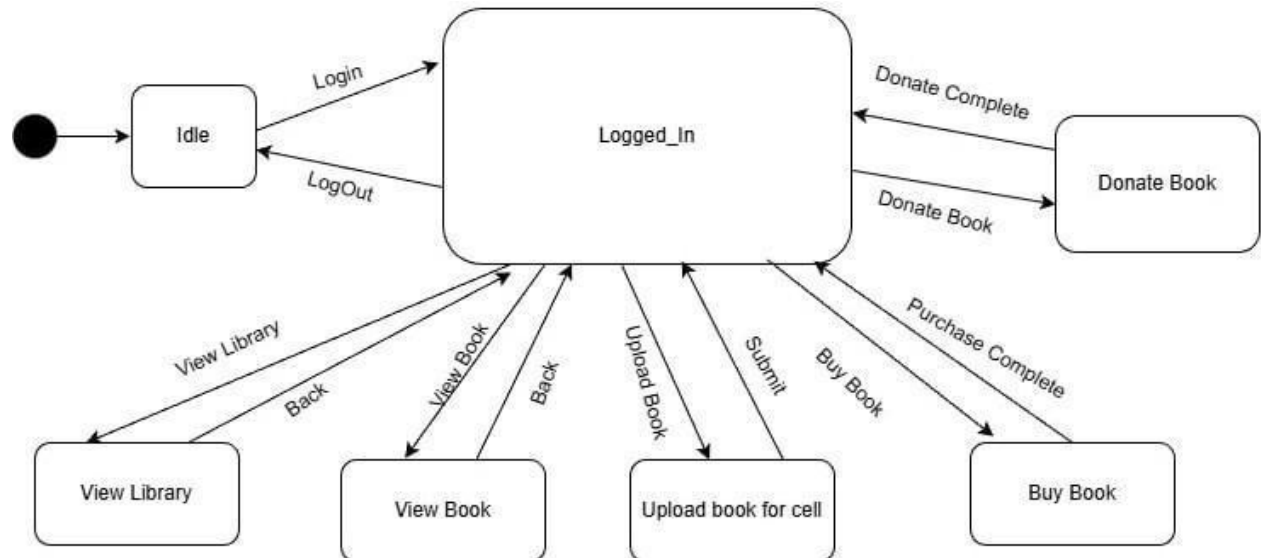


**Figure: User Authentication**

The diagram is an overview of the user states between registration and login, password reset, suspension, and reactivation, with the transitions between states being initiated by actions such as OTP verification, and an abuse decision made by the administration.

**ID: 02**

**Name: User Logged In**



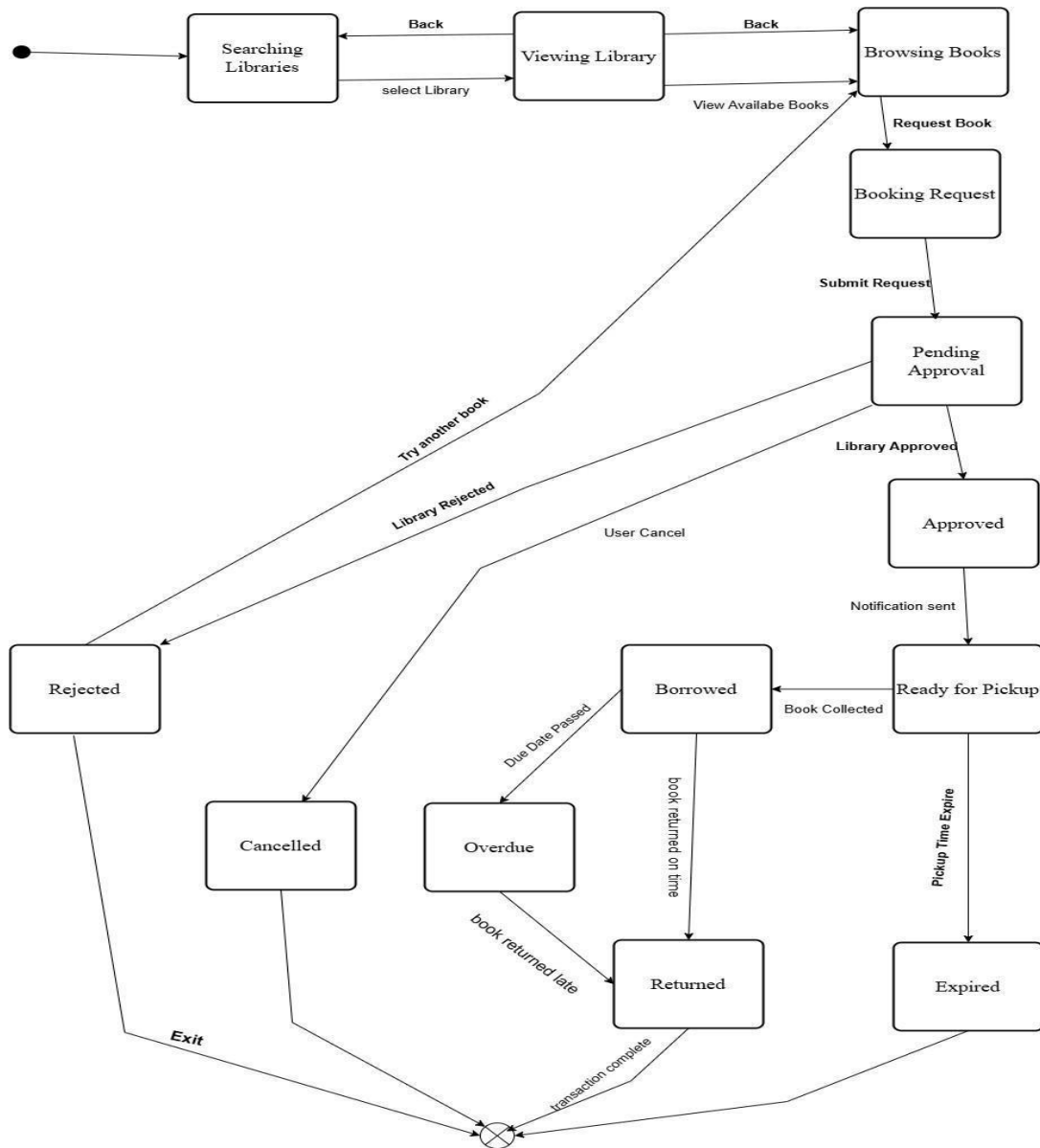
**Figure: User Logged In**

It describes behavior of the user after the login process and the user is able to view, upload, donate, and purchase books, with each process returning to the logged-in state.



**ID: 03**

**Name: Library**

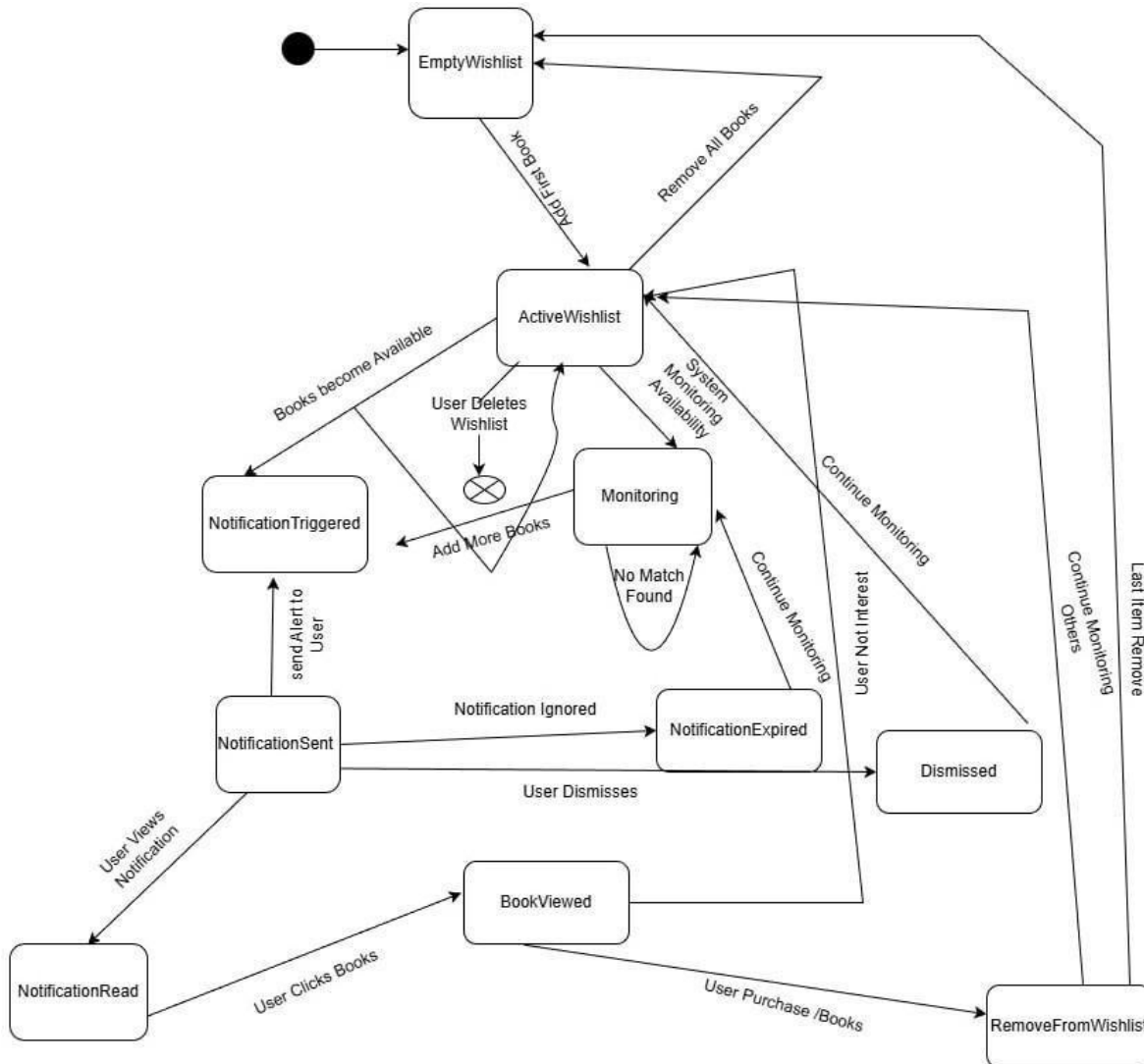


**Figure: Library**

This flow illustrates the way in which the users search libraries, request books and navigate approval, rejection, cancellation, and overdue situations and finally complete the transaction.

**ID: 04**

**Name: Wishlist & Alerts**

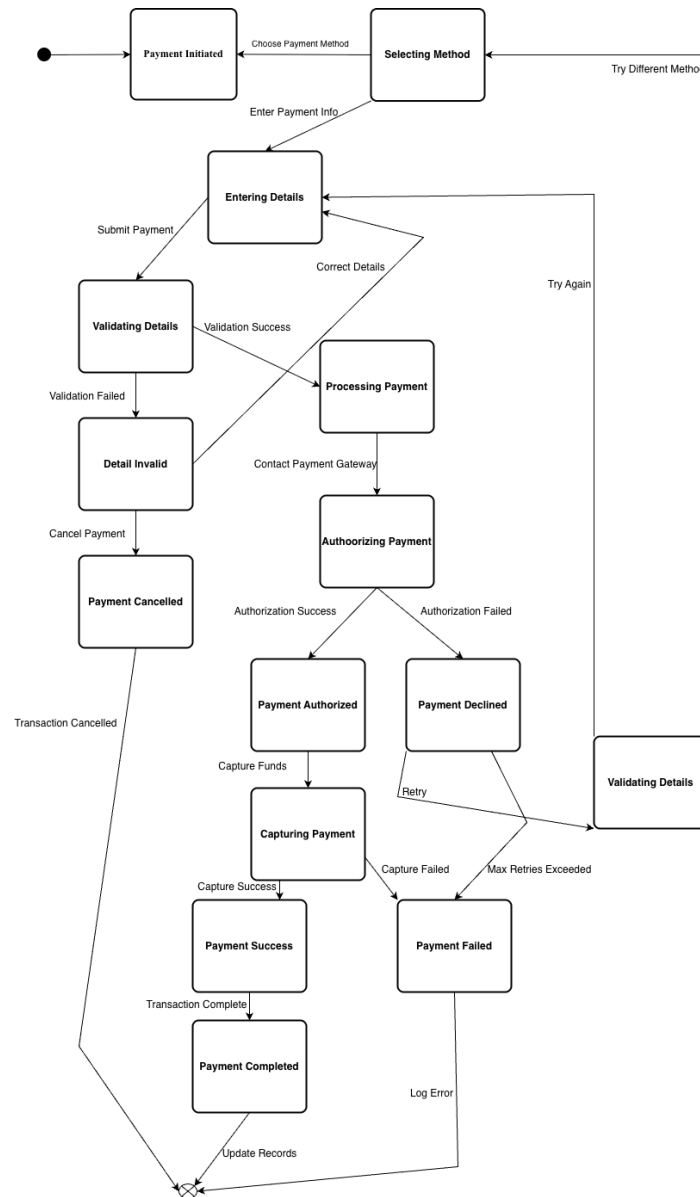


**Figure: Wishlist & Alerts**

This illustration follows the process of the wishlist operation of adding books to the track of the availability, sending a notification, seeing the books, and deleting the items, being dynamic as the user becomes involved in its management

**ID: 05**

**Name: Payments**



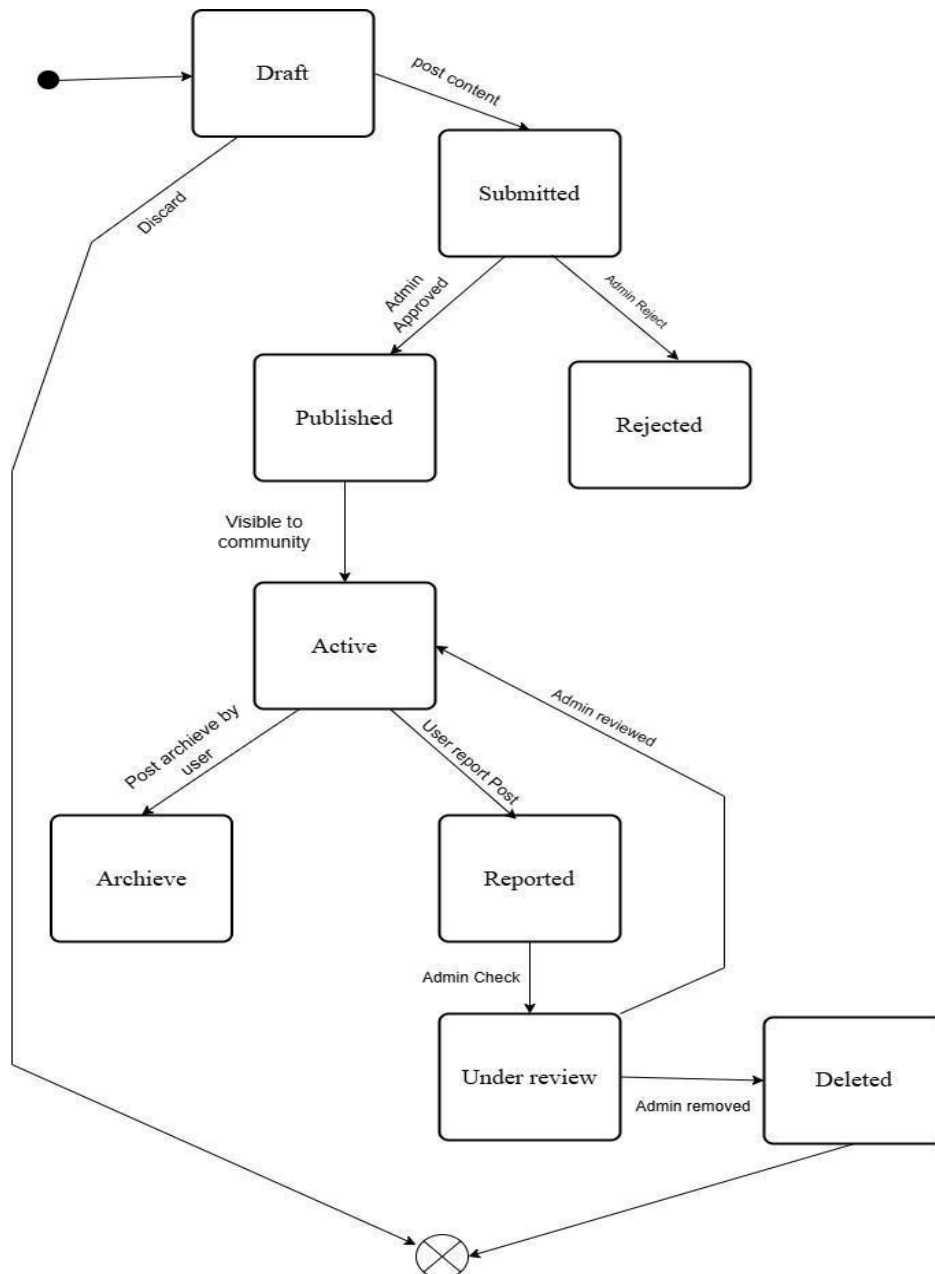
**Figure: Payments**

The diagram covers the entire payment process, including the choice of method, the validation of the detail, the authorization, the capture of the funds, the

completion of the transaction, and the errors and the resilience strategies.

**ID: 06**

**Name: Community Post**



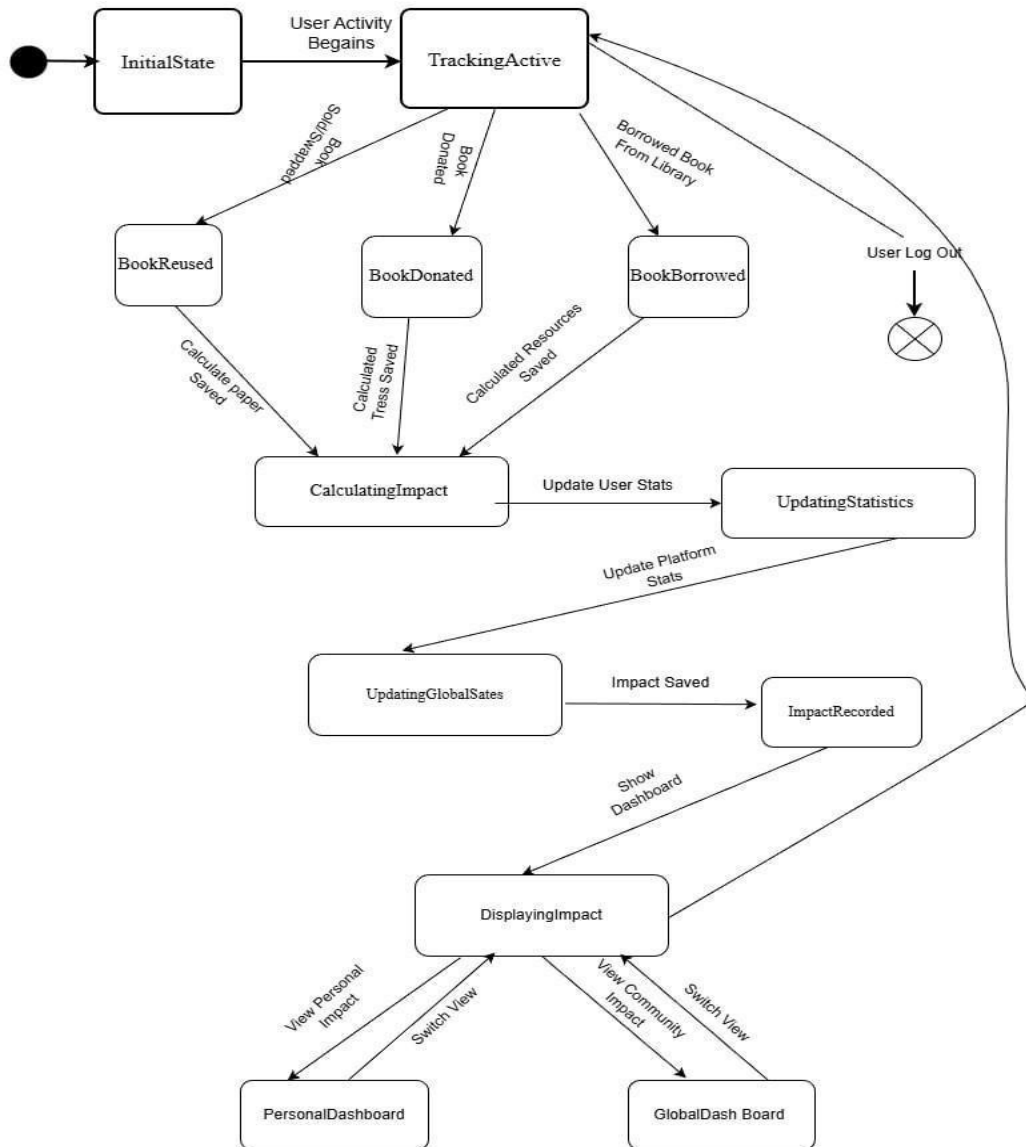
**Figure: Community Post**

This follows the path of a post between draft and publication, through the steps of

approval by the administration, report on the post by users, review and the potential possibility of the result being an archive or the post being deleted.

**ID: 07**

**Name: Environmental Dashboard**

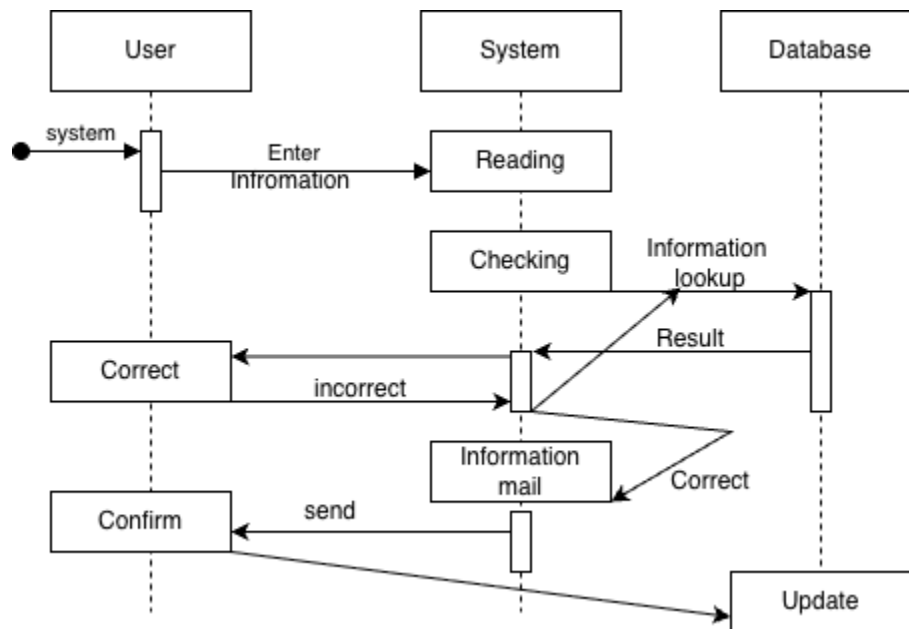


**Figure: Environmental Dashboard**

This flow demonstrates how user actions such as reuse, donation, and borrowing processes are monitored, computed, and figures updated and eventually, personal

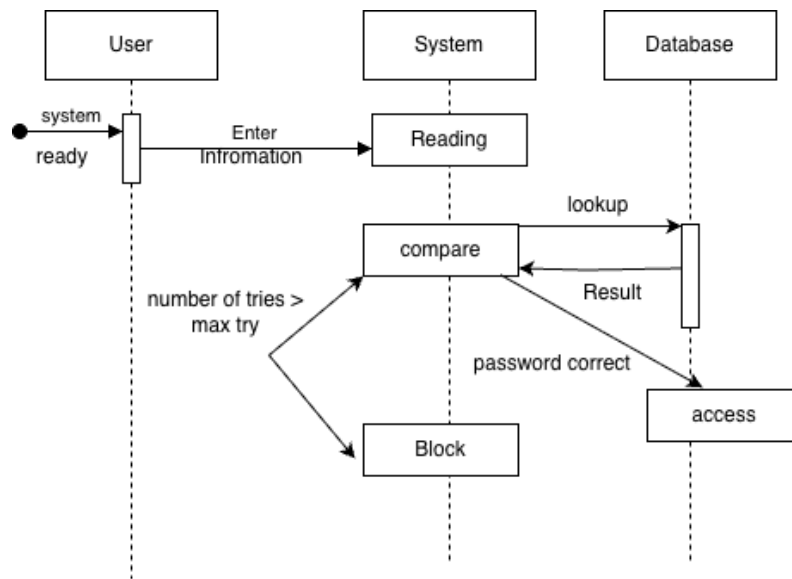
and global dashboard interfaces are displayed.

### 4.7.2 Sequence Diagram

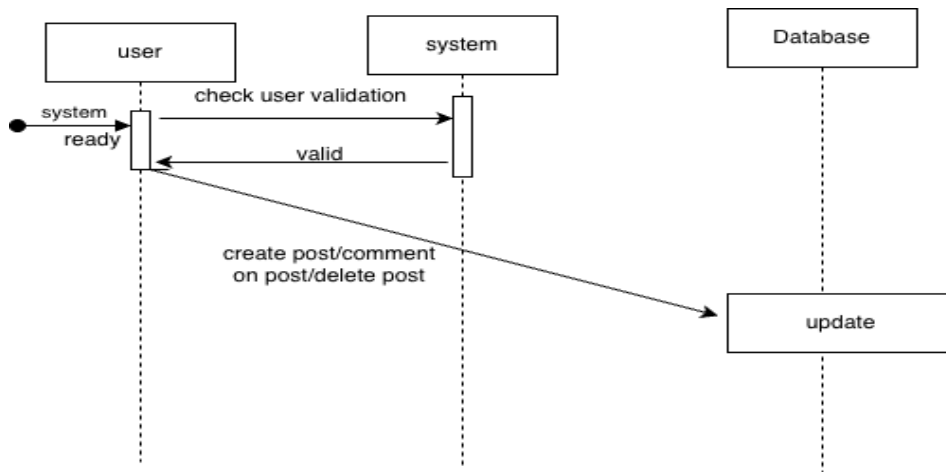


**Figure: Registration**

It demonstrates the procedure in which the user inputs the registration data into the system, the system checks the data against the database and approves or rejects the data before the final procedure of updating the system is complete.



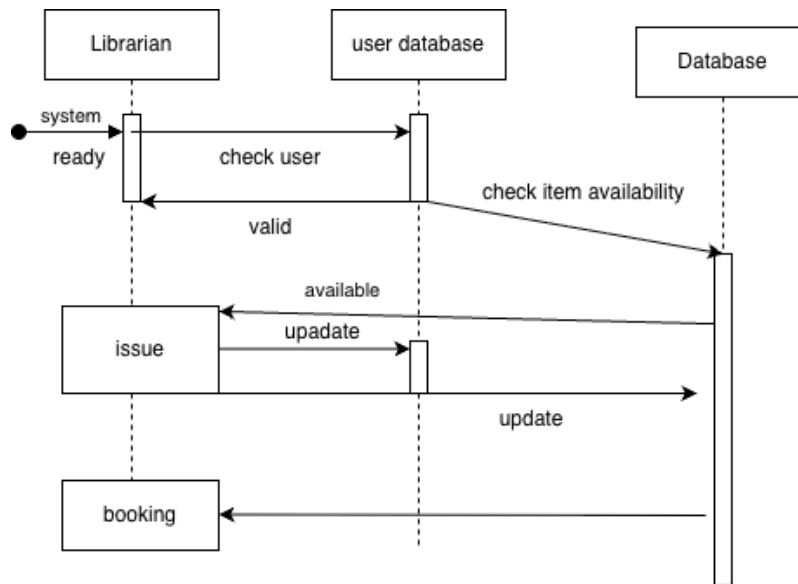
**Figure: Sign In**



**Figure: Community**

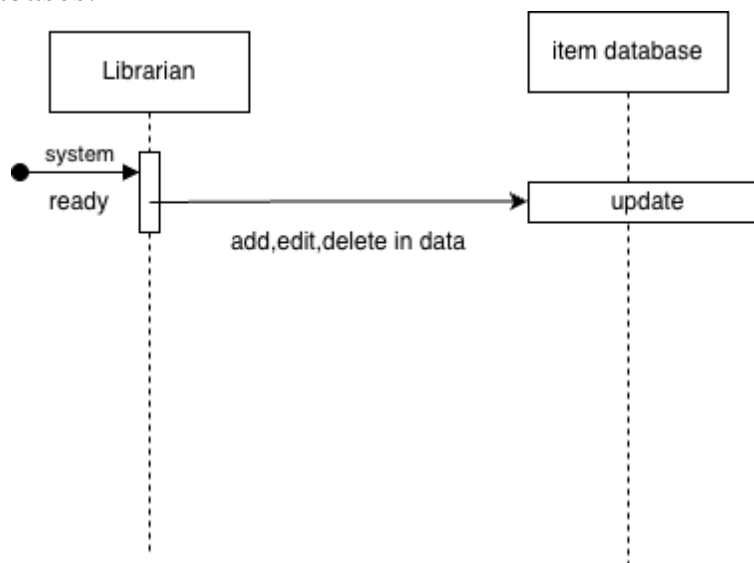
The following sequence diagram shows how the user signs in: the system will read the credentials entered by the user, compare them to the entries in the database and either allow the user in when correct or lock out the user after a self-imposed number of attempts. It entails interactions between the User, System and Database.

It describes the interaction of a verified user with the system to create, comment or delete posts, and the updates will be reflected in the database.



**Figure: Borrow**

This flow will signify the role of the librarian that involves checking availability of users and items, updating records and booking it out by the coordinated exchange of data with databases.

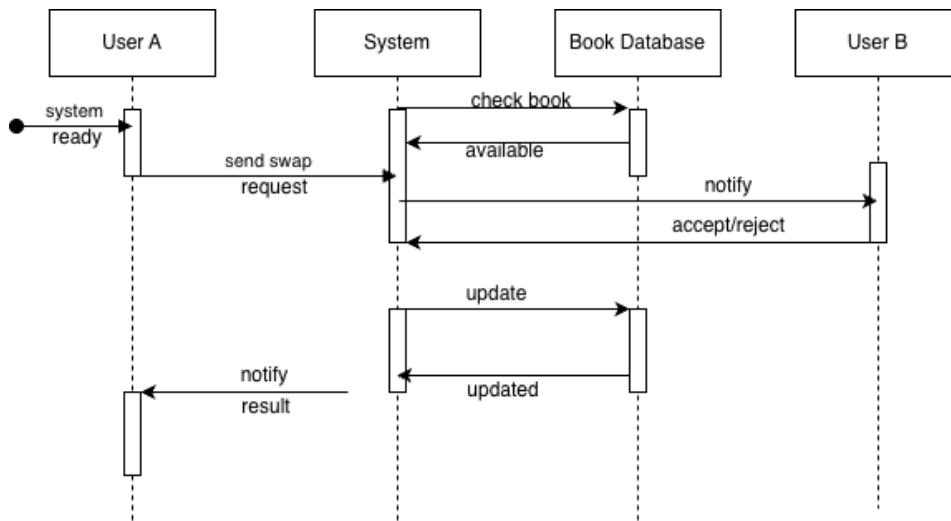


**Figure: Update**

It depicts the process by which the librarian is able to add, edit or delete information which gets processed by the system and will be reflected in the item

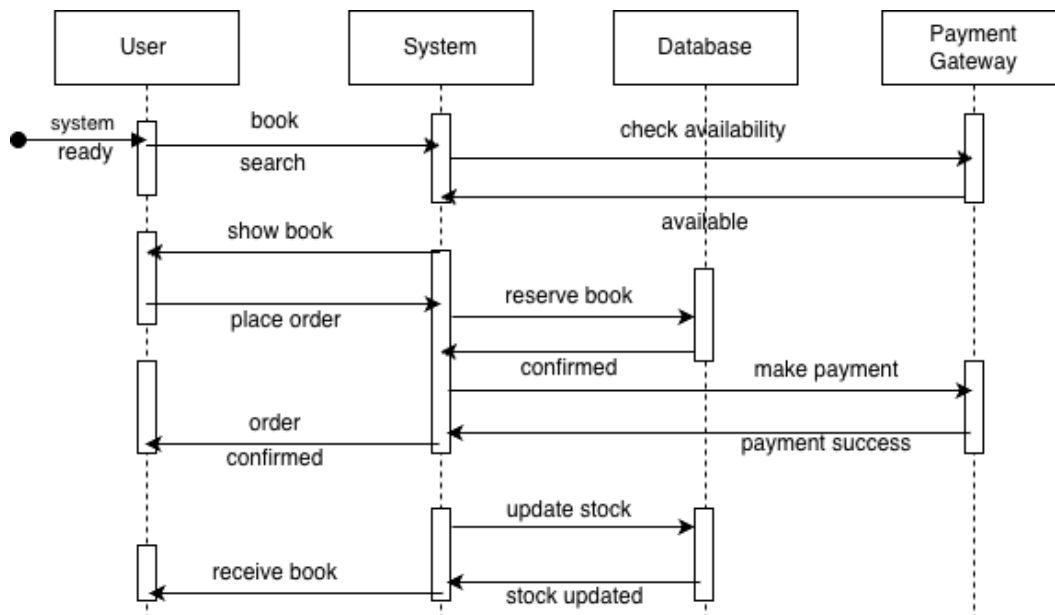


database.



**Figure: Book Swapped**

This diagram illustrates how User A initiates a book swap request, the system checks availability, notifies User B, and updates the database based on User B's response. It highlights the interaction flow between users and system components during a swap.



**Figure: Book Selling**

This diagram shows the process of selling a book, starting from the user's search to

placing an order, checking availability, reserving the book, processing payment, and confirming delivery. It emphasizes coordination between the system, database, and payment gateway.

## **5. Implementation**

The BoiChokro project implementation stage aims at transforming the suggested system design into a complete functioning web based application. This stage will entail the creation of the frontend interface, the implementation of the backend logic, connection to the database, and the establishment of the local server environment. The system is based on client-server architecture with users communicating with the web interface and the server executing the requests and data storage.

BoiChokro uses HTML, CSS and JavaScript to build the frontend and provide a responsive user-friendly interface. They can easily browse books, sell books, trade books, and order to borrow books an existing library through the use of books. Community section gives people an opportunity to make posts, like (appreciate) material and utter remarks about conversations concerning books.

The system backend is developed in PHP that does user authentication, form submission, session management, and communication with MySQL database. Any of the dynamic processes like adding book, handling swap request, donation request, library borrowing request and community interaction are processed on the server side.

The database management system is MySQL, which will contain all the required information, as user profile, book, library, borrowing, donation request, and community post and comment. Structured database tables are used to maintain the relationships between different entities.

To test and develop the system, the system is launched in a local server setup using XAMPP, which is consisting of Apache, MySQL and PHP. This is where developers are free to test features, keep bugs and make sure that the system is working before deploying the project into a live server. The input validation and session control are basic security measures that are put in place to maintain the reliability and safety of the system.

## **5.1 Technology Stack:**

BoiChokro system is created on the basis of the following technology stack:

### **Frontend Technologies:**

#### 1. HTML (Hypertext markup language):

Used to design the structure and format of web pages, such as forms, navigation bars and content sections.

#### 2. CSS (Cascading Style Sheets):

Familiar with designing and styling the web site so as to make it presentable and easy to construct.

#### 3. JavaScript (JS):

Used to provide dynamic features and interactivity like form validation, dynamic content updating and user interaction.

### **Backend Technology:**

#### 1. PHP (Hypertext Preprocessor):

The server-side scripting language used to serve user requests, to manage user sessions, work with the data, and interact with the database.

### **Database:**

#### 1. MySQL:

They are used to maintain and store the data related to users, books, library services, loans, donor requests, and community news.

### **Server Environment:**

#### 1. XAMPP Server:

Apache web server, MySQL database, and PHP are built-in into it and used as a local development environment. It can be used to test and debug the system on a local basis.

## 6. SQA & Testing

Software Quality Assurance (SQA) and Testing are critical processes in the development of the BoiChokro system, which will ensure that the software is reliable, functional, secure, and user-friendly. The main objective of SQA is to ensure that the quality of the software is maintained at all times during the software development process, adhering to standards, processes, and best practices. During the development of this project, SQA involves requirement verification, design review, implementation review, and testing of all the functional modules of the software. The testing process is conducted to detect any errors, bugs, and usability problems in the software before it is delivered. The testing process will ensure that all the features of the BoiChokro system are functioning according to the requirements specified in the software development process. The BoiChokro system is tested using both manual testing and automated testing tools. The manual testing process is conducted to check the user interface, usability, and user experience of the software. The automated testing process is conducted using Selenium WebDriver, which tests critical user scenarios such as login, book posting, swapping, donation application, borrowing books from the library, and interaction with the community members.

The following types of testing are applied in the project:

1. **Unit Testing:** Testing individual components or modules (e.g., login, book upload form).
2. **Integration Testing:** Ensuring proper interaction between frontend, backend, and database.
3. **System Testing:** Testing the complete system as a whole.
4. **Functional Testing:** Verifying that all features work according to requirements.
5. **Usability Testing:** Checking user-friendliness and ease of navigation.
6. **Regression Testing:** Ensuring that new changes do not break existing functionality.

## 6.1 Test Case Table

Test Case ID	Module Name	Test Case Description	Pre-Condition	Test Steps	Expected Result	Actual Result	Status
TC-01	User Registration	Register with valid information	User is on registration page	1. Enter valid name, email, password 2. Click Register	Account is created successfully and user is redirected	As Expected	Pass
TC-02	User Login	Login with valid credentials	User is registered	1. Enter valid email and password 2. Click Login	User logs in and dashboard is displayed	As Expected	Pass
TC-03	Book Sell	Add a book for selling	User is logged in	1. Go to Add Book 2. Enter book details 3. Submit	Book is listed in sell section	As Expected	Pass
TC-04	Book Swap	Request a book swap	User is logged in	1. Select a book 2. Click Swap Request	Swap request is sent successfully	As Expected	Pass
TC-05	Donation System	Apply for book donation	User is logged in	1. Fill donation form 2. Submit application	Donation request is sent to admin	As Expected	Pass

TC-06	Library Borrow	Borrow a book from library	User is logged in, book exists	1. Select library book 2. Click Borrow	Borrow request is sent and recorded	As Expected	Pass
TC-07	Community Post	Create a community post	User is logged in	1. Write post 2. Click Post	Post appears in community feed	As Expected	Pass
TC-08	Comment System	Comment on a post	User is logged in	1. Open a post 2. Write comment 3. Submit	Comment is added successfully	As Expected	Pass
TC-09	Admin Panel	Approve donation request	Admin is logged in	1. Open donation requests 2. Approve one request	Donation request status updated	As Expected	Pass
TC-10	Logout	Logout from system	User is logged in	1. Click Logout	User is logged out and redirected to login page	As Expected	Pass

## 6.2 Testing Tool: Selenium WebDriver

Selenium WebDriver is employed as a major automated test tool for BoiChokro. It facilitates automated control for web browsers and assists in simulating real user interactions for a web application. Selenium is a commonly used test tool for web applications due to its flexibility and cross-browser compatibility.

### Why Selenium WebDriver is Used

1. Supports popular browsers such as Chrome and Firefox

2. Automates repetitive test cases
3. Reduces manual testing effort
4. Improves test accuracy and consistency
5. Helps in regression testing after system updates

### **Testing Activities Using Selenium**

1. Automated testing of login and registration
2. Automated testing of book listing and borrowing flows
3. Automated testing of donation application process
4. Automated testing of community posting and commenting
5. Automated testing of navigation and form validation

### **Advantages**

1. Open-source and free to use
2. Cross-browser support
3. Supports multiple programming languages
4. Suitable for large-scale web application testing

## **7. Conclusion & Future Work**

### **7.1 Conclusion**

The BoiChokro system is a socially effective and innovative solution to the problem.

The obstacles to book readers in Bangladesh. Through the combination of book selling, swapping,

The system has addressed the accessibility and economic challenges by converting donation, as well as library borrowing, into a single platform. The community aspect of BoiChokro

also adds value to the user experience, allowing them to interact with each other, give feedback on different opinions and products and so on.

The adoption of such a system facilitates effective reuse of books which not only benefits the individual users in financial terms but also helps in maintaining the environment by eliminating the need to print books that are not in good condition and conserving of the natural resource. Also, the donation option helps in restoring useful reading resources to the communities and the individuals who require them most and hence promotes literacy and education in socio-economic backgrounds.

## **7.2 Future Work**

Even though the present release of BoiChokro offers an extensive amount of services,

The platform can be further improved by making a number of improvements to increase its usability, reach, and impact.

### **Future work includes:**

#### **1. Mobile application development:**

Creating Android and iOS apps in order to make the platform more affordable to users.  
who are the major users of the mobile devices.

#### **2. Online Payment Integration:**

The adoption of secure online purchasing and selling gateways.

#### **3. Enhanced Search and Recommendation:**

Adoption of machine learning or smart recommendation system to recommend books.  
it depends on the interests of the user and the history of reading.

#### **4. Delivery Support System:**

The addition of logistics capabilities to facilitate tracking of books delivery and coordination of the same sellers and buyers.

#### **5. Regional Language Support:**

Inclusion of the multilingual services (Bangla and English) to reach a wider audience across.  
Bangladesh.



## 6. Analytics Dashboard:

Giving the administrators and librarians analytics dashboards to monitor the usage trends, popular books, donation influence, and community involvement.

BoiChokro can become one of the top online book sharing platforms, education, and society development in Bangladesh. It can in the future with future developments still benefit the readers, fund educational programs, and make a difference.