

Library Management System



Submitted By:

Muhammad Zubair 2020-CS-601
Muhammad Huzzifa 2020-CS-606
Abdul Rehman 2020-CS-624
Muhammad Qasim 2020-CS-643

Submitted To:

Sir. Waqas Ali

Contents

| | | |
|----------|---|----------|
| 1 | MilesStone-1 | 6 |
| 1.1 | Project Porposal | 6 |
| 1.1.1 | Introduction | 6 |
| 1.1.2 | Objective | 6 |
| 1.1.2.1 | Insertion | 6 |
| 1.1.2.2 | Extraction | 6 |
| 1.1.2.3 | Keep-Record | 6 |
| 1.1.2.4 | Searching | 6 |
| 1.1.3 | Approach | 6 |
| 1.1.4 | Requirements | 7 |
| 1.1.4.1 | Functional Requirements | 7 |
| 1.1.4.2 | Non-Functional Requirements | 7 |
| 1.1.5 | Features | 7 |
| 1.1.6 | Tools and Technology | 8 |
| 1.1.7 | Project Work-flow | 8 |
| 1.1.8 | References | 8 |
| 1.2 | Feasibility Report | 9 |
| 1.2.1 | Team Members | 9 |
| 1.2.2 | The Client | 9 |
| 1.2.3 | The task to be Understand | 9 |
| 1.2.4 | Benefits | 9 |
| 1.2.5 | Preliminary requirements analysis | 9 |
| 1.2.6 | Technical Requirement | 10 |
| 1.2.7 | Scope | 10 |
| 1.2.8 | Suggested Deliverable | 11 |
| 1.2.8.1 | Management Deliverable | 11 |
| 1.2.8.2 | Requirement analysis | 11 |
| 1.2.8.3 | Wire-frames | 11 |
| 1.2.8.4 | Source Code | 11 |
| 1.2.9 | Technical Deliver-ability | 11 |
| 1.2.9.1 | Database | 11 |
| 1.2.9.2 | Interactive menu | 11 |
| 1.2.10 | Development Model | 11 |
| 1.2.10.1 | Process visibility | 11 |
| 1.2.10.2 | Separation of tasks | 12 |
| 1.2.10.3 | Quality control | 12 |

| | | |
|----------|---|-----------|
| 1.2.11 | Outline | 12 |
| 1.2.11.1 | Milestone 1 (October 10, 2022): Feasibility Report | 12 |
| 1.2.11.2 | Milestone 2 (October 24, 2022): Progress Report Meeting I | 12 |
| 1.2.11.3 | Milestone 3 (November 21, 2022): Progress Report Meeting II | 12 |
| 1.2.11.4 | Milestone 4 (December 19, 2022) | 12 |
| 1.2.12 | Risk Analysis | 12 |
| 1.2.12.1 | Changing Requirements Risk | 12 |
| 1.2.12.2 | Non-Functional Requirements | 13 |
| 1.2.13 | Conclusion | 13 |
| 2 | MilesStone-2 | 13 |
| 2.1 | Scenario | 13 |
| 2.2 | Case Diagram | 14 |
| 2.3 | USE CASES | 15 |
| 2.3.1 | Login | 15 |
| 2.3.2 | Add new book | 16 |
| 2.3.3 | View Book | 17 |
| 2.3.4 | Delete Book | 18 |
| 2.3.5 | Update Books | 19 |
| 2.3.6 | Add User | 20 |
| 2.3.7 | Delete User | 21 |
| 2.3.8 | Update User Info | 22 |
| 2.3.9 | View User Information | 23 |
| 2.3.10 | Search | 24 |
| 2.3.11 | Issue Book | 25 |
| 2.3.12 | Receiving Return Book | 26 |
| 2.3.13 | Complete Book Information | 27 |
| 2.3.14 | Chart | 28 |
| 2.3.15 | Issue Books Chart | 29 |
| 2.3.16 | Threshold Chart | 30 |
| 2.4 | Wireframe | 31 |
| 2.4.1 | Login | 31 |
| 2.4.2 | Home Page | 31 |
| 2.4.3 | View User | 32 |
| 2.4.4 | Delete User | 32 |

| | | |
|--------|-------------------------------------|----|
| 2.4.5 | Update User | 33 |
| 2.4.6 | Add Book | 33 |
| 2.4.7 | View Book | 34 |
| 2.4.8 | Delete Book | 34 |
| 2.4.9 | Update Book | 35 |
| 2.4.10 | Issue Book | 36 |
| 2.4.11 | Return Book | 37 |
| 2.4.12 | Chart Page | 37 |
| 2.4.13 | Issuance Chart | 38 |
| 2.5 | UI/UX Design | 39 |
| 2.5.1 | Login | 39 |
| 2.5.2 | Home Page | 39 |
| 2.5.3 | View User | 40 |
| 2.5.4 | Delete User | 40 |
| 2.5.5 | Update User | 41 |
| 2.5.6 | Add Book | 41 |
| 2.5.7 | View Book | 42 |
| 2.5.8 | Delete Book | 42 |
| 2.5.9 | Update Book | 43 |
| 2.5.10 | Issue Book | 44 |
| 2.5.11 | Return Book | 45 |
| 2.5.12 | Chart Page | 45 |
| 2.5.13 | Issuance Chart | 46 |
| 2.6 | ERD Diagram | 47 |
| 2.7 | Realtional Schema Diagram | 48 |
| 2.8 | UML Diagram | 49 |
| 2.9 | DFD Model | 50 |

1 MilesStone-1

1.1 Project Porposal

1.1.1 Introduction

The project Library Management System is developed in C-sharp, which mainly focuses on basic operations in library like adding new members, new books, and updating new information, searching books and members to borrow and return books. With the ever-increasing sea of information, this modern function of library is becoming very complex and tedious to handle manually. The librarian has to devote most of his/her time to handing such functions of library. So after conducting the feasibility study we decided to make the manual Library management system to be computerized. We tried our best to make our project which shall provide user friendly interface.

1.1.2 Objective

The software Library Management System has four main objectives.

1.1.2.1 Insertion Insertion to Database,User friendly input screen

1.1.2.2 Extraction Extracting from Database,Attractive output screen

1.1.2.3 Keep-Record Keep in record,borrowed books list and available book list

1.1.2.4 Searching Searching Facility system,Search for books and members

1.1.3 Approach

The project Library Management System is a software for monitoring and controlling the transactions in a library. In our existing system all the transaction of books is done manually. So taking more time for a transaction like borrowing a book or returning it and also for searching of members and books. The main focus of this project is to lessen human effort and encourage efficient record keeping.

1.1.4 Requirements

1.1.4.1 Functional Requirements

- Our software shall provide log-in and sign-up function with the help of carefully designed forms and providing labels in front of each entity.
 - Our software shall provide different menu buttons for books, students, stored information and in last an exit button to exit from app.
 - Our software shall provide different functionality for students to register themselves, so they could borrow and return books.
 - Our System Must be providing users the ability to get detailed information of any of the book.
 - Our System Must be providing user friendly Layout During Registration, Issuance of book and Searching Books

1.1.4.2 Non-Functional Requirements

- Our software must be secure by providing every student a code after register by keep ensuring the integrity of user data.
 - Our software must provide response within 1 to 5 seconds.
 - Our software can be made available to use 24/7, using uninterrupted power supply or any medium like UPS or Generators.
 - Our software must be providing good performance to avoid any Lagging Issue While Performing any Steps.
 - Our software must be recoverable and manageable in case of error by recognizing error and keep tracking of application stable versions.
 - Our software must provide easy and readable text to user, so that it can be made easy to use for End-Users.

1.1.5 Features

To utilize resources in an efficient manner by increasing their productivity through automation. Through our software user can add members, add books, search members/books, update information, borrow and return books in quick time.

- Improve library services.
- Help Librarians with management information.
- Help Librarian in reporting on various operation of library.
- Increase the rate at which tasks are done accurately.

Development of this application is highly economically feasible. Some important features of our software are given below.

- Search for a book.
- Add/Delete book.
- Maintain the database.
- View user details.
- Issue books.
- Return books.

1.1.6 Tools and Technology

- Operating System: Windows XP, Window Vista, Window 7, Window 8, Window 8.1, Window 10.
- Language: C-sharp.
- IDE: Net-Beans IDE.
- Database Management System: MySQL.
- Software: Visual Studio.

1.1.7 Project Work-flow

- Librarian as an admin log-in himself in the software.
- Admin can perform multiple tasks like add Book, add Student, view Data, delete Book, remove Student, and search Data.
- Admin perform these functions from Database.
- If student is registered by admin in database then he/she can borrow or return books.
- If student is not registered in database then he/she has to contact with admin.

1.1.8 References

- <https://www.wikipedia.org>
- <https://www.geeksforgeeks.org/c-sharp-tutorial>
- <https://www.geeksforgeeks.org/mysql-common-mysql-queries>
- <https://www.udemy.com>

1.2 Feasibility Report

1.2.1 Team Members

1. Muhammad Zubair (2020-CS-601) (2020cs601@student.uet.edu.pk)
2. Muhammad Huzzifa (2020-CS-606) (2020cs606@student.uet.edu.pk)
3. Abdul Rehman (2020-CS-624) (2020cs624@student.uet.edu.pk)
4. Muhammad Qasim (2020-CS-643) (2020cs643@student.uet.edu.pk)

1.2.2 The Client

The Client is an educational based institutional organization which is paying a team developer to create Library Management System for clients.

1.2.3 The task to be Understand

In this project, Library Management System is developed where user can perform various functions like adding new members, new books, and updating new information, searching books and members to borrow and return books. With the ever-increasing sea of information, this modern function of library is becoming very complex and tedious to handle manually. The librarian has to devote most of his/her time to handing such functions of library.

1.2.4 Benefits

This application is developed for handling multiple users at a time. Multiple users can run the application at a same time and perform the various option according to his/her choice. This application is also a feature to check records of users (Books and Students) and how many transactions done.

1.2.5 Preliminary requirements analysis

The System needs to fulfill the following functional requirement:

- Our System shall provide sign-up and Log-in functionality to Admin.
- Our System shall provide a functionality of Fast response to different user.
- Our System shall provide a functionality of transition history to different user.
- Our System shall provide a functionality to change code of registered student.

The System needs to fulfill the following non-functional requirement:

- Our System must response up to 1-10 seconds.
- Our System must be secure as it provides login functionality to the different user.
- Our System must be capable of handling 20K register per seconds.
- Our System must be scalable up to 50K request per seconds.
- Our System must be available 24/7 hours with the help of UPS or Generator.
- Our System must be recoverable and manageable in case of errors by recognizing error and keeping track of application stable version.
- Our System must be easy to understand for end-users.
- Our System must ensure data integrity for login and Sign-up functions.

1.2.6 Technical Requirement

- Server System is going to be running on a server. In this case, system itself is acting as server. System itself will provide service to clients. It will fulfill client's requests which are running on same computer and hence acts as a server itself.
- Database Database will be used to store and modify data of users and books. The group members will have access to database. They can add edit or delete tables. Group will be using free databases online.

1.2.7 Scope

To utilize resources in an efficient manner by increasing their productivity through automation. Through our software user can add members, add books, search members/books, update information, borrow and return books in quick time.

- Improve library services.
- Help Librarians with management information.
- Help Librarian in reporting on various operation of library.
- Increase the rate at which tasks are done accurately.

Development of this application is highly economically feasible. Some important features of our software are given below.

- Search for a book.
- Add/Delete book.
- Maintain the database.

- View user details.
- Issue books.
- Return books.

1.2.8 Suggested Deliverable

1.2.8.1 Management Deliverable

1.2.8.2 Requirement analysis Project proposal includes functional and nonfunctional requirements. This ensures that all requirements are included, if client finds any feature missing or miss-communicated then document is modified according to his requirements.

1.2.8.3 Wire-frames Project includes the wire-frames of the website so the client will know how the application is going to look before the system is even created.

1.2.8.4 Source Code Document of completed project with source code concludes the project.

1.2.9 Technical Deliver-ability

1.2.9.1 Database The system uses database to store the data of user's sign-up and data of books.

1.2.9.2 Interactive menu The system will provide and interactive menu.

1.2.10 Development Model

The Project will follow the waterfall model as the requirements of the client are well defined and client knows exactly what they want, that's why waterfall model is best suitable for the system.

1.2.10.1 Process visibility The client is well aware of the stage where the group and project currently lie, because of the regulator meetings and discussions between the group and the client.

1.2.10.2 Separation of tasks The group will concentrate at one are at a time since the group members are limited and the project needs co-ordination so every task will be done parallel to each other and group will coordinate effectively.

1.2.10.3 Quality control A waterfall model allows the group to focus on the given task at a time and helps them to communicate through it, since the time frame is already known and plan of development is designed, a waterfall model will allow the developer to understand and develop better.

1.2.11 Outline

1.2.11.1 Milestone 1 (October 10, 2022): Feasibility Report The group is going to submit and present the feasibility report of the project, that is going to be conclusive and detailed about the requirements and problems, risks of the project development.

1.2.11.2 Milestone 2 (October 24, 2022): Progress Report Meeting I The group will present a progress report that tells the client what is done until now and how the flow is working out for the group.

1.2.11.3 Milestone 3 (November 21, 2022): Progress Report Meeting II The group will present a progress report that tells the client what is done until now and how the flow is working out for the group.

1.2.11.4 Milestone 4 (December 19, 2022) The project will be live as a working functional system and will be presented by the team and then handed over to the client.

1.2.12 Risk Analysis

1.2.12.1 Changing Requirements Risk The client may change the requirements during the development process, which can affect the architecture completely or partially depending on the situation. Solution: To avoid this from occurring, requirements are discussed in advance with the client. Since we are using the waterfall model, which is linear and does not support run time changes to the plan.

1.2.12.2 Non-Functional Requirements Since the group does not have access to high end servers there could be changes to the non-functional requirement fulfillment. Solution: A follow up meeting held to discuss the non-functional requirements

1.2.13 Conclusion

Based on the analysis mentioned above the group concludes that the project is feasible and the group (collectively) is willing to take this project. Since the group is going to use free database servers, and other free online tools, also as a student project there will be no labour cost as well, so the project costs nothing but time. The project will be live as functional system on the deadline provided by the client.

2 MilesStone-2

2.1 Scenario

We will be designing Library Management system. The system will allow admin to log-In into the system. The system will allow admin to add data, delete data and update data of users and books and also view the list of users and books. The system will allow admin to issue books to users and each user can issue maximum three books. The system will allow admin to check the returning of book from the relevant user in case of late then user has to pay fine. The system will allow admin to check the complete details of Issued and returned books.

2.2 Case Diagram

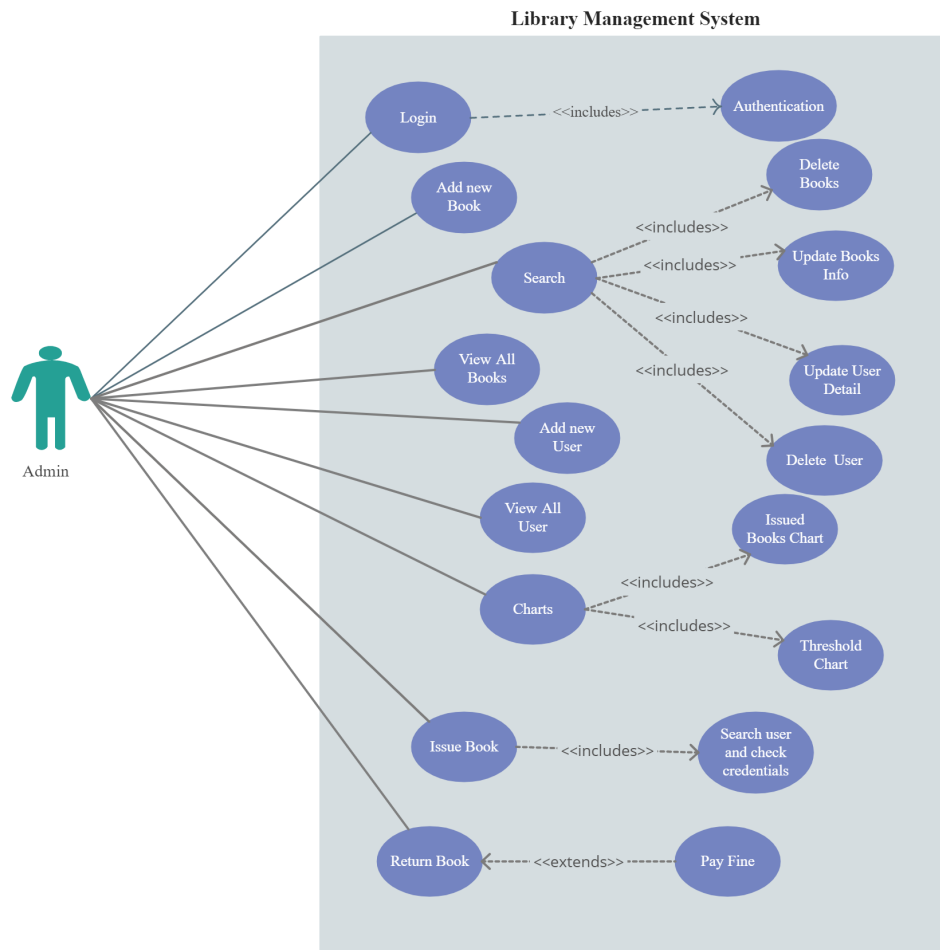


Figure 1: Case Diagram.

2.3 USE CASES

2.3.1 Login

| | |
|------------------------|--|
| Name | Login |
| Goal | Enables a librarian to login successfully |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to handle whole library transaction. Librarian open the website of library management system. |
| Entry Condition | Librarian has to register or must has login id in database to get access. |
| Post Condition | After successfully login admin can perform multiple tasks like register users. Add books Delete books View all information from database |
| Basic Flow | A login page will appear to admin/librarian. Admin has to enter his/her username. Admin has to enter his/her password. Then click on login button. System will compare its login id, if it matches to database then user will enter in website. If login id of admin does not match to database, then wrong username or password box will appear. |
| Alternate Flow | Librarian has to contact with database administrator to generate his login id to access website. |
| Exception | Admin forget his/her password. Admin has to click on forgotten password or reset password. Then try with new password. |

2.3.2 Add new book

| | |
|------------------------|---|
| Name | Add new book |
| Goal | Enables a librarian to add a new book information successfully |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to add new books data in database of library. Librarian open the database and start entering data of books. |
| Entry Condition | Librarian must has authenticated login id in database to access this feature. |
| Post Condition | After successfully adding a new book, admin can perform multiple tasks like view book details. Delete book |
| Basic Flow | Add Book” page will appear to admin/librarian. Admin has to enter book name. Admin has to enter book author name. Admin has to enter book price. Then click on Add button. System will add this information of book to the database. |
| Alternate Flow | No other path. |
| Exception | Admin is entering book detail but it is not shown in database. Admin has to check the connection to server. Rebuilt the connection and then try again to add book. |

2.3.3 View Book

| | |
|------------------------|---|
| Name | View Book |
| Goal | Enables a librarian to view all books successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to check how much books are there in library. Librarian is notified that to generate a report on present books in library. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to add books first to view books. |
| Post Condition | After viewing details of all book's admin can generate a report. Delete any book |
| Basic Flow | Admin has to click on "Books" button first. Then select "View Book" from menu. If admin has entered some books, then books information will display to a new page. At this stage admin can delete any book by selecting book. |
| Alternate Flow | Librarian has to check database to see which books are added. |
| Exception | There is no any book in view list. Admin has to add new book first. Then try again to view books details. |

2.3.4 Delete Book

| | |
|------------------------|---|
| Name | Delete Book |
| Goal | Enables a librarian to delete books successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian want to delete books from the list. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to add books first to delete books. |
| Post Condition | After viewing details of all book's admin can generate a report. Delete any book |
| Basic Flow | Admin has to click on "Books" button first. Then select "View Book" from menu. If admin has entered some books, then books information will display to a new page. At this stage admin can delete any book by selecting book. |
| Alternate Flow | Librarian has to check database to delete any book. |
| Exception | There is no any book in view list. Admin has to add new book first. Then try again to delete books details. |

2.3.5 Update Books

| | |
|------------------------|---|
| Name | Update Book |
| Goal | Enables a librarian to update books successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian want to update books from the list. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to add books first to update books. |
| Post Condition | After viewing details of all book's admin can generate a report. Delete any book |
| Basic Flow | Admin has to click on "Books" button first. Then select "View Book" from menu. If admin has entered some books, then books information will display to a new page. At this stage admin can update any book by selecting book. |
| Alternate Flow | Librarian has to check database to update any book details. |
| Exception | There is no any book in view list. Admin has to add new book first. Then try again to update books details. |

2.3.6 Add User

| | |
|------------------------|--|
| Name | Add User |
| Goal | Enables a librarian to register a user successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to register new users' data in database of library. Librarian is notified that to register all user of institution |
| Entry Condition | Librarian must has authenticated login id in database to access this feature. |
| Post Condition | After successfully adding a new user, admin can perform multiple tasks like view user details. Delete user |
| Basic Flow | “Add User” page will appear to admin/librarian. Admin has to enter user name. Admin has to enter department name. Admin has to enter semester number. Admin has to enter user contact number. Admin has to enter user E-mail. Admin has to enter user registration number. Then click on Add button. System will add this information of user to the database. |
| Alternate Flow | No other path. |
| Exception | Admin is entering book detail but it is not shown in database. Admin has to check the connection to server. Rebuilt the connection and then try again to add new user. |

2.3.7 Delete User

| | |
|------------------------|--|
| Name | Delete User |
| Goal | Enables a librarian to delete a user successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to delete users' data in database of library. |
| Entry Condition | Librarian must has authenticated login id in database to access this feature. |
| Post Condition | After successfully deleting a user, admin can perform multiple tasks like view user details. |
| Basic Flow | “Delete User” page will appear to admin/librarian. Admin has to enter user name. Admin has to enter user id. Then click on delete button. System will delete this information of user to the database. |
| Alternate Flow | Librarian has to check database to delete any user details. |
| Exception | Admin has to check the connection to server. Rebuilt the connection and then try again to delete user. |

2.3.8 Update User Info

| | |
|------------------------|--|
| Name | Update User Info |
| Goal | Enables a librarian to update a user successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to update users' data in database of library. |
| Entry Condition | Librarian must has authenticated login id in database to access this feature. |
| Post Condition | After successfully updating user info, admin can perform multiple tasks like view user details. |
| Basic Flow | “Update User” page will appear to admin/librarian. Admin has to enter user name. Admin has to enter user id. Then click on update button. System will update this information of user to the database. |
| Alternate Flow | Librarian has to check database to update any user details. |
| Exception | Admin has to check the connection to server. Rebuilt the connection and then try again to update user. |

2.3.9 View User Information

| | |
|------------------------|---|
| Name | View User |
| Goal | Enables a librarian to view all user successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to check how much users are there in library. Librarian is notified that to generate a report on present users registered in library. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to add user first to view user. |
| Post Condition | After viewing details of all user's admin can generate a report. Remove user |
| Basic Flow | Admin has to click on "User" button first. Then select "View User" from menu. If admin has entered some users, then users' information will display to a new page. At this stage admin can delete any book by selecting user. |
| Alternate Flow | Librarian has to check database to see which users are added. |
| Exception | There is no any user in view list. Admin has to add new user first. Then try again to view user details. |

2.3.10 Search

| | |
|------------------------|---|
| Name | Search |
| Goal | Enables a librarian to search user or book successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to search the users' data or books data in database of library. |
| Entry Condition | Librarian must has authenticated login id in database to access this feature. |
| Post Condition | After successfully searching a user or a book, admin can perform multiple tasks like view user details. |
| Basic Flow | Admin want to delete or update data of books or users. Admin has to enter book name or user name. Admin has to enter user id or book id. Then click on delete or update data of books or users. System will delete or update this information of user or books from the database. |
| Alternate Flow | Librarian has to check database to see which user and books are added. |
| Exception | Admin has to check the connection to server. Rebuilt the connection and then try again to delete or update data of users or books. |

2.3.11 Issue Book

| | |
|------------------------|--|
| Name | Issue Book |
| Goal | Enables a librarian to issue for user successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to issue books for users who are registered in database. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to add user and some books first to issue books them for user. |
| Post Condition | After issuing books to user admin can track the date and time of issue of book to each user. |
| Basic Flow | Admin has to click on “Issue Book” button first. Now admin has to enter all data of user and book also. But on second side admin search the user id from database. If user id matches with database, then he/she can borrow book. |
| Alternate Flow | No Other Path. |
| Exception | Admin facing errors during issue of book for a user. Register user in database first and then check which book you are trying to issue present in database or not. Then try again to issue book. |

2.3.12 Receiving Return Book

| | |
|------------------------|---|
| Name | Receiving Return Book |
| Goal | Enables a librarian to receive book return by the user successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to receive all issue books for users who are registered in database. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to issue some to books to some users to receive returned book by users. |
| Post Condition | After receiving return books by user admin can track the date and time of return of book to each user. Also tracking this thing that which user has not return the books. |
| Basic Flow | Admin has to click on "Return Book" button first. Now admin has to enter all data of user and book also like user id and book name. Select return date. Then click on return button. |
| Alternate Flow | No Other Path. |
| Exception | Admin facing errors during receiving returned book from user. Check user is registered in database first and then check which book you are trying to receive from user. Then try again. |

2.3.13 Complete Book Information

| | |
|------------------------|--|
| Name | Complete Book Information |
| Goal | Enables a librarian to see all issued books and returned books successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified that to provide list of all issued and return books. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to issue some to books to some users and receive returned book by users. |
| Post Condition | After this use case user will able to organize and understand which types of books are mostly issued by users. Also tracking this thing that which user has not return the books. |
| Basic Flow | Admin has to click on “Complete Books Information” button first. Now admin will be able to see list of all issued books and their issued date. Now admin will be able to see list of all returned books and their returned date. |
| Alternate Flow | No Other Path. |
| Exception | Admin facing errors during viewing complete detail of books. Check connection with server. Then try again. |

2.3.14 Chart

| | |
|------------------------|--|
| Name | Chart |
| Goal | Enables a librarian to see charts successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified to provide chart. |
| Entry Condition | Librarian has to register or must has login id in database to get access. Librarian must have to issue some books to see the chart. |
| Post Condition | After this use case user will able to organize and understand which types of books are mostly issued by users. |
| Basic Flow | Admin has to click on “Charts” button first. Now admin will be able to see list of charts. Now admin will be able to see the required chart according to his choice. |
| Alternate Flow | No Other Path. |
| Exception | Admin has not issued books to user. Check connection with server. Then try again. |

2.3.15 Issue Books Chart

| | |
|------------------------|--|
| Name | Chart |
| Goal | Enables a librarian to see Issue Books chart successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified to provide issued books chart. |
| Entry Condition | Librarian has to register or must have login id in database to get access. Librarian must have to issue some books to see the chart. |
| Post Condition | After this use case user will be able to organize and understand which types of books are mostly issued by users. |
| Basic Flow | Admin has to click on "Charts" button first. Now admin will be able to see list of charts. Now admin click on "Issue Books Chart". Now admin will be able to see the chart of issued books to user. |
| Alternate Flow | No Other Path. |
| Exception | Admin has not issued books to user. Check connection with server. Then try again. |

2.3.16 Threshold Chart

| | |
|------------------------|--|
| Name | Threshold Chart |
| Goal | Enables a librarian to see threshold chart successfully. |
| Actors | Admin/Librarian |
| Trigger | Librarian is notified to provide threshold chart. |
| Entry Condition | Librarian has to register or must have login id in database to get access. Librarian must have added and issued books to users. |
| Post Condition | After this use case user will be able to organize and understand which types of books are mostly short because more users have issued that book. |
| Basic Flow | Admin has to click on "Charts" button first. Now admin will be able to see list of charts. Now admin will click on "Threshold Chart". Now admin will be able to see threshold chart of books. |
| Alternate Flow | No Other Path. |
| Exception | Admin has not added books in database. Check connection with server. Then try again. |

2.4 Wireframe

2.4.1 Login



Figure 2: LOGIN.

2.4.2 Home Page

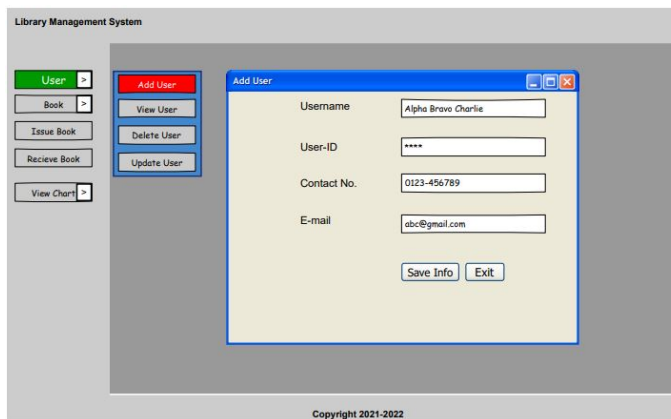


Figure 3: HOME PAGE.

2.4.3 View User

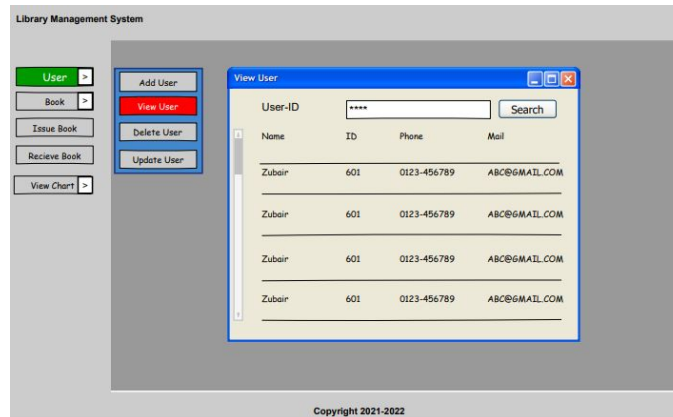


Figure 4: VIEW USER.

2.4.4 Delete User

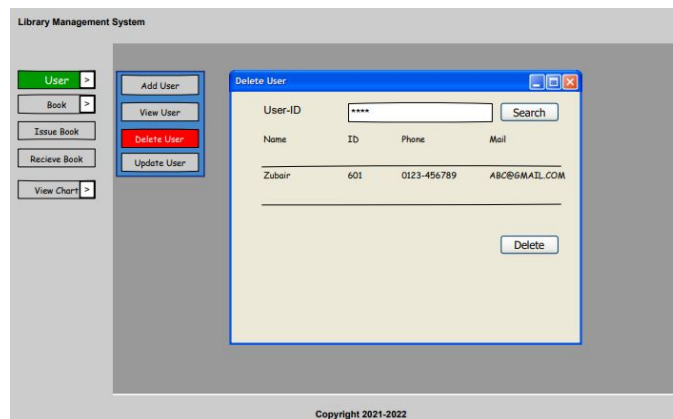


Figure 5: DELETE USER.

2.4.5 Update User

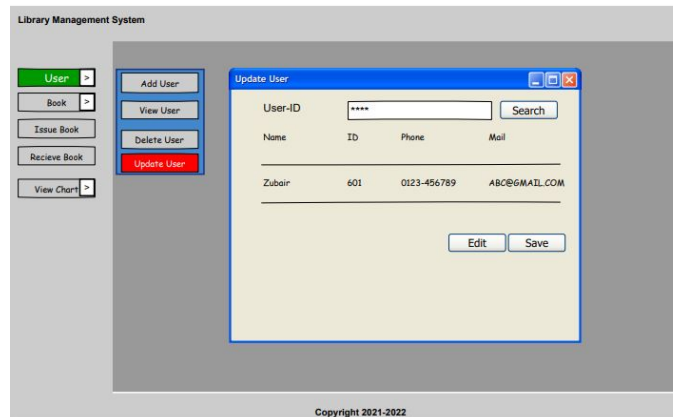


Figure 6: UPDATE USER.

2.4.6 Add Book

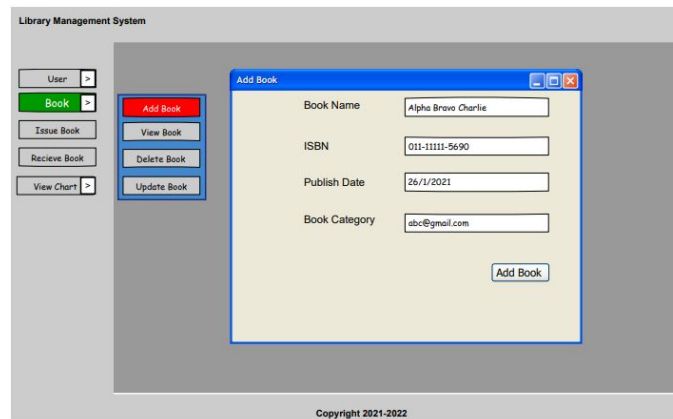


Figure 7: ADD BOOK.

2.4.7 View Book

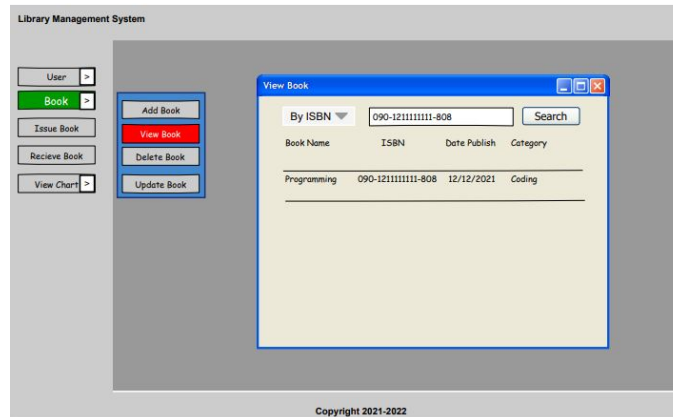


Figure 8: VIEW BOOK.

2.4.8 Delete Book

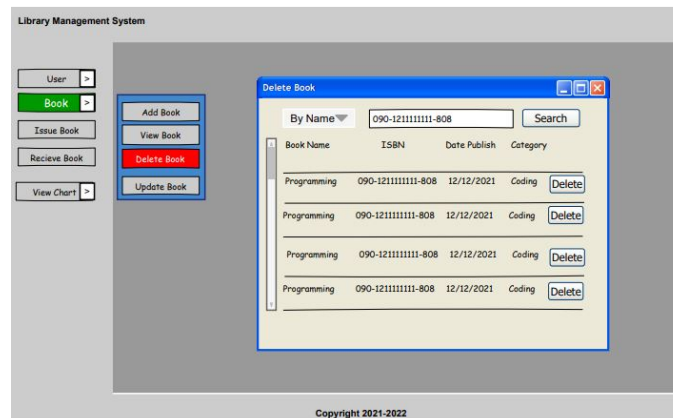


Figure 9: DELETE BOOK.

2.4.9 Update Book

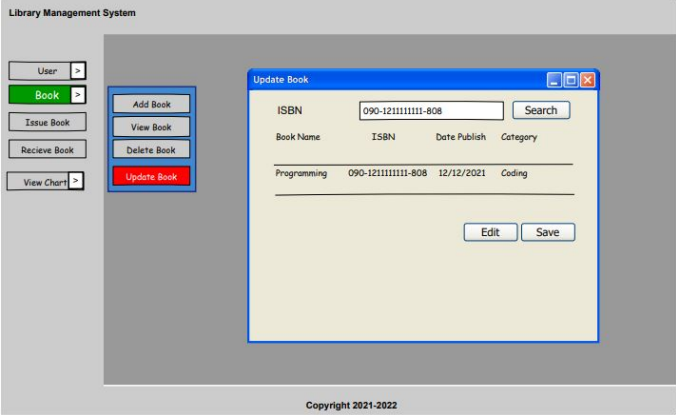


Figure 10: UPDATE BOOK.

2.4.10 Issue Book

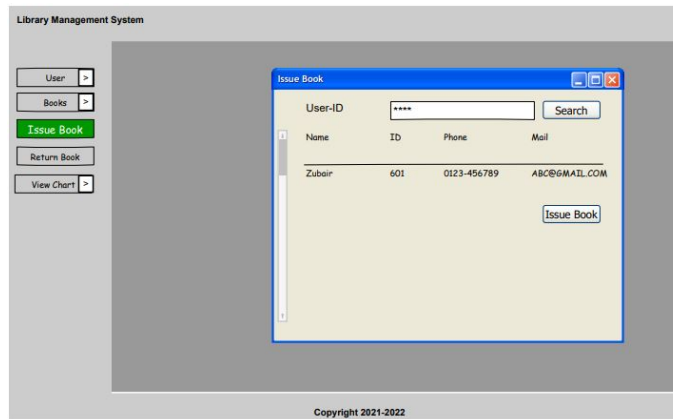


Figure 11: ISSUE BOOK.

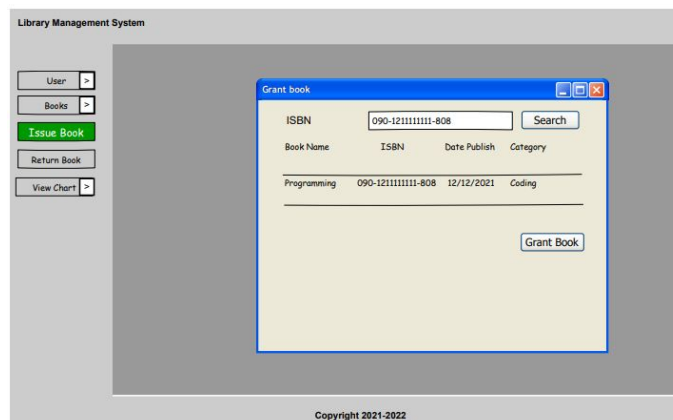


Figure 12: ISSUE BOOK.

2.4.11 Return Book

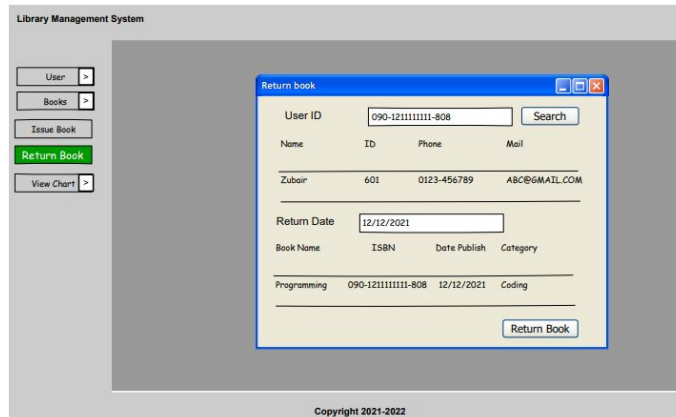


Figure 13: RETURN BOOK.

2.4.12 Chart Page

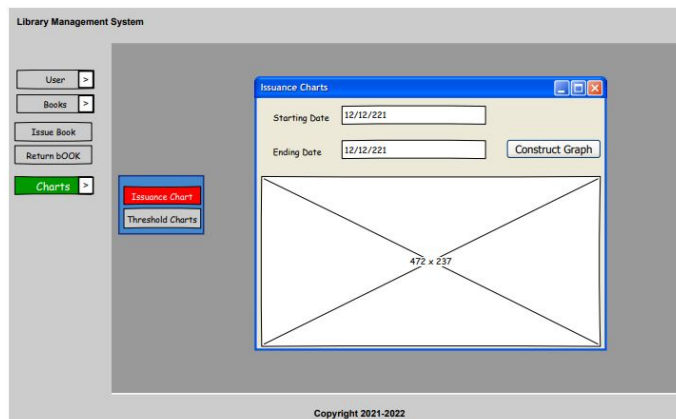


Figure 14: ThRESHOLD CHART.

2.4.13 Issuance Chart

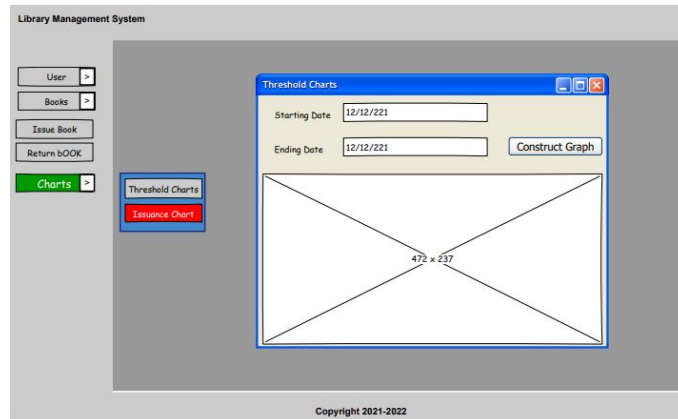


Figure 15: ISSUANCE CHART.

2.5 UI/UX Design

2.5.1 Login

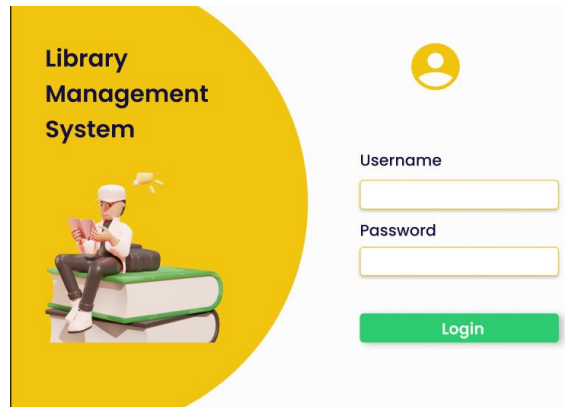


Figure 16: LOGIN.

2.5.2 Home Page

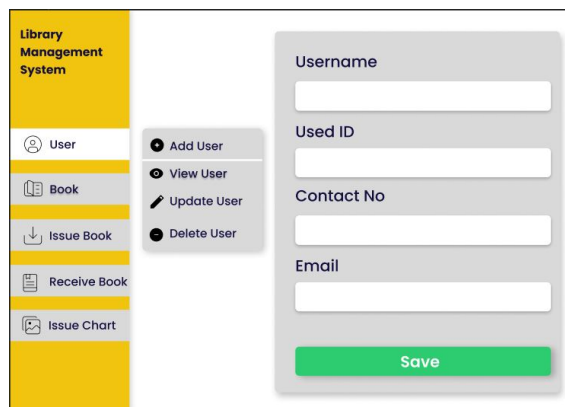


Figure 17: HOME PAGE.

2.5.3 View User

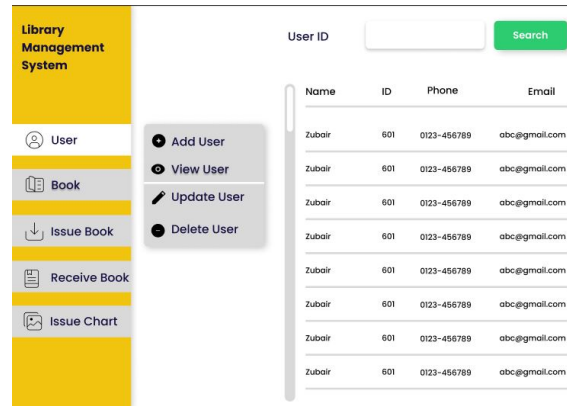


Figure 18: VIEW USER.

2.5.4 Delete User

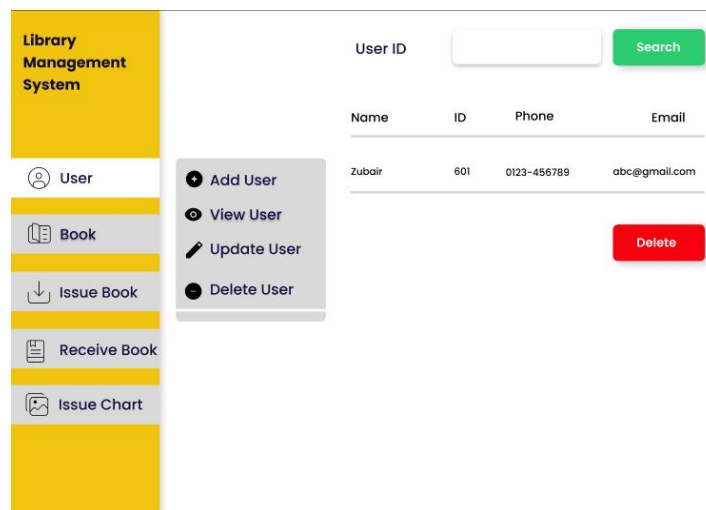


Figure 19: DELETE USER.

2.5.5 Update User

The screenshot shows the 'Update User' interface. On the left is a sidebar with the following menu items: User, Book, Issue Book, Receive Book, and Issue Chart. The 'User' menu item is selected. In the main area, there is a search bar for 'User ID' with a 'Search' button. Below the search bar is a table with the following data:

| Name | ID | Phone | Email |
|--------|-----|-------------|---------------|
| Zubair | 601 | 0123-456789 | abc@gmail.com |

Below the table are two buttons: 'Update' and 'Save'. A floating menu is visible over the table with the following options: Add User, View User, Update User (selected), and Delete User.

Figure 20: UPDATE USER.

2.5.6 Add Book

The screenshot shows the 'Add Book' interface. On the left is a sidebar with the following menu items: User, Book, Issue Book, Receive Book, and Issue Chart. The 'Book' menu item is selected. In the main area, there is a floating menu with the following options: Add Book (selected), View Book, Update Book, and Delete Book. To the right of the floating menu is a form with the following fields: Book Name, ISBN No, Publish Date, and Book Category. Below the form is a green 'Add Book' button.

Figure 21: ADD BOOK.

2.5.7 View Book

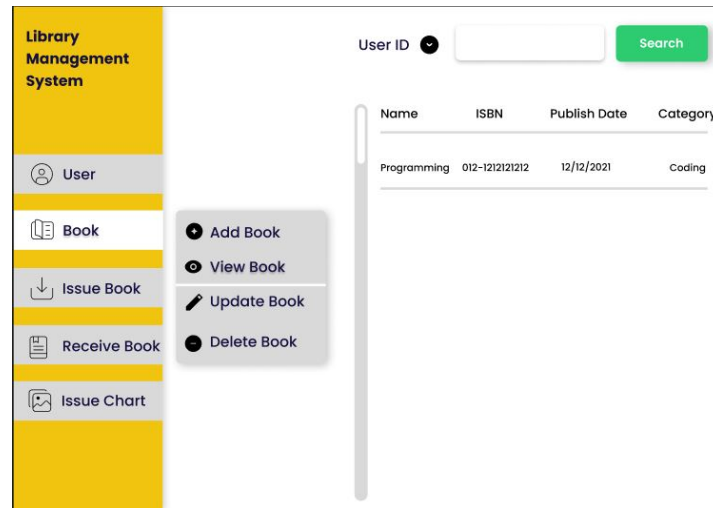


Figure 22: VIEW BOOK.

2.5.8 Delete Book

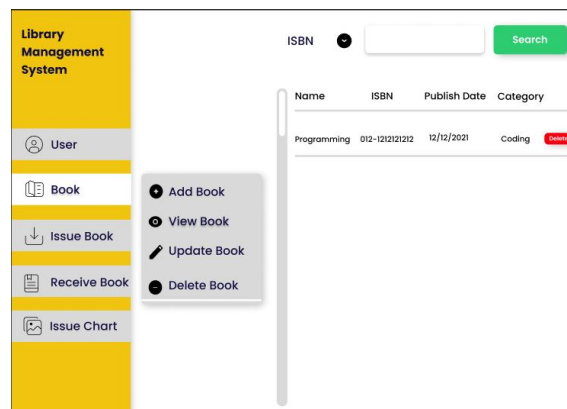


Figure 23: DELETE BOOK.

2.5.9 Update Book

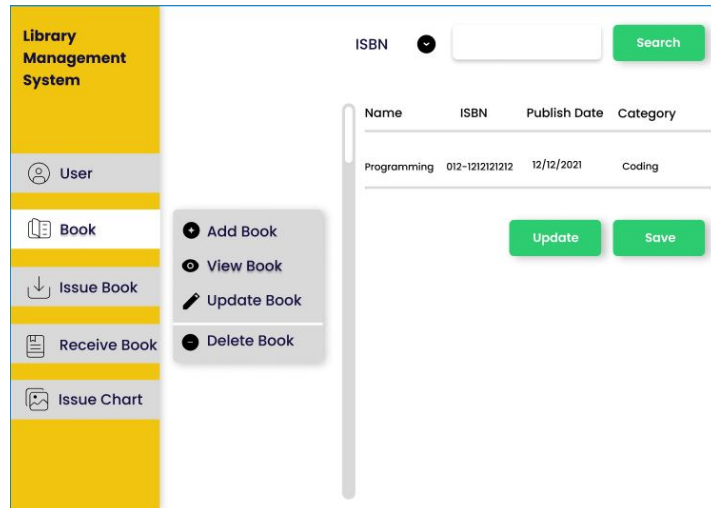


Figure 24: UPDATE BOOK.

2.5.10 Issue Book

Library Management System

User ID Search

| Name | ID | Phone | Email |
|--------|-----|-------------|---------------|
| Zubair | 601 | 0123-456789 | abc@gmail.com |

Issue Book

Figure 25: ISSUE BOOK.

Library Management System

ISBN Search

| Name | ISBN | Publish Date | Category |
|-------------|-------------|--------------|----------|
| Programming | 0123-456789 | 12/12/2021 | Coding |

Grant Book

Figure 26: ISSUE BOOK.

2.5.11 Return Book

| Name | ID | Phone | Mail |
|--------|-----|-------------|---------------|
| Zubair | 601 | 0123-446789 | abc@gmail.com |

| Name | ISBN | Publish Date | Category |
|-------------|-------------|--------------|----------|
| Programming | 0123-456789 | 12/12/2021 | Coding |

Figure 27: RETURN BOOK.

2.5.12 Chart Page

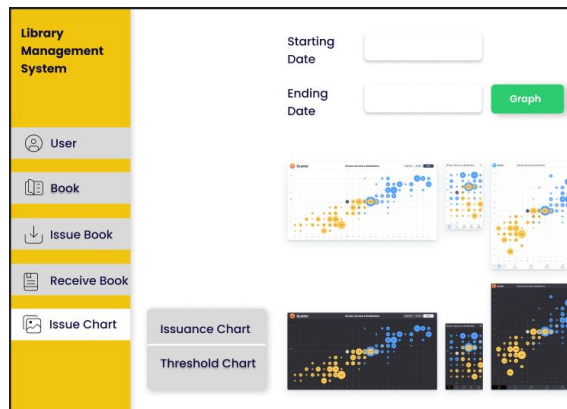


Figure 28: ThRESHOLD CHART.

2.5.13 Issuance Chart

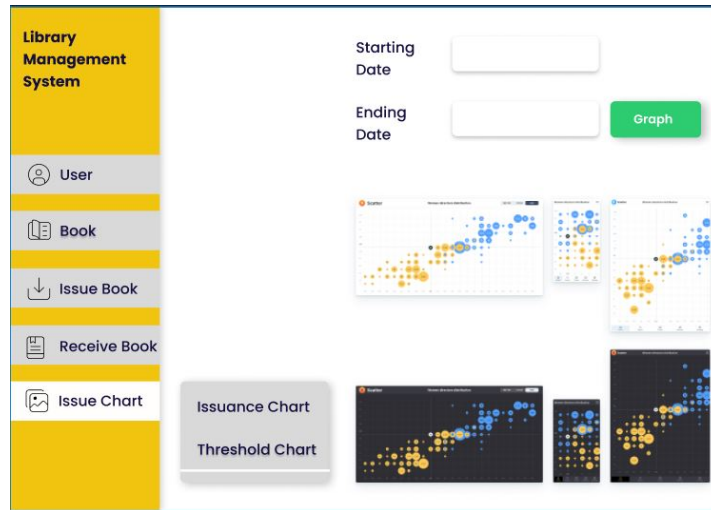


Figure 29: ISSUANCE CHART.

2.6 ERD Diagram

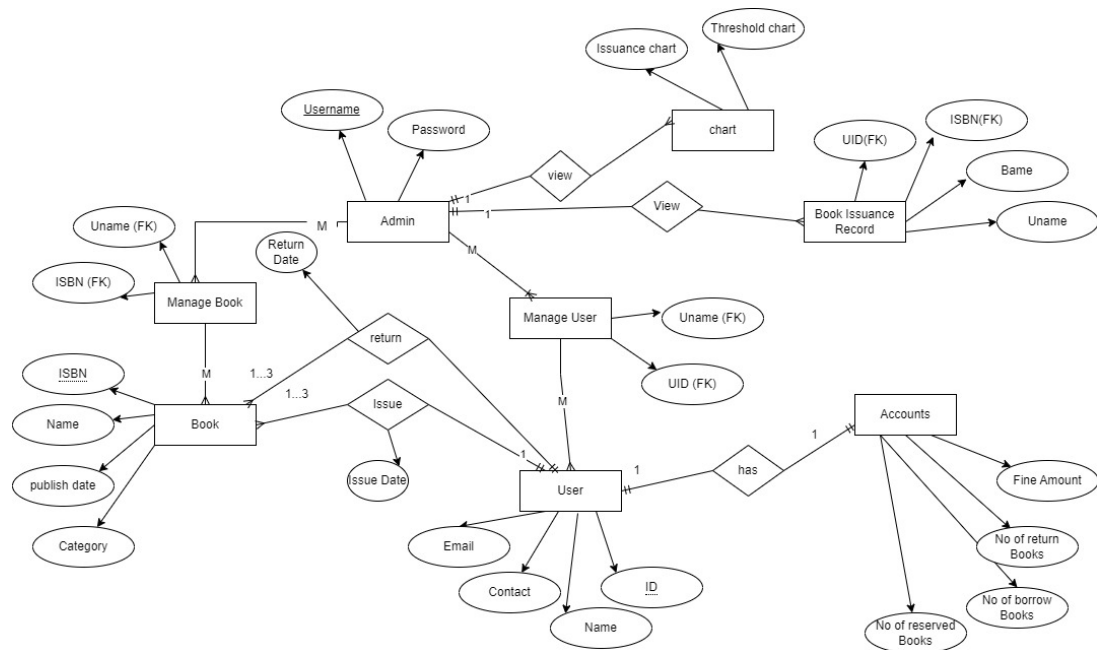


Figure 30: ERD.

2.7 Realtional Schema Diagram

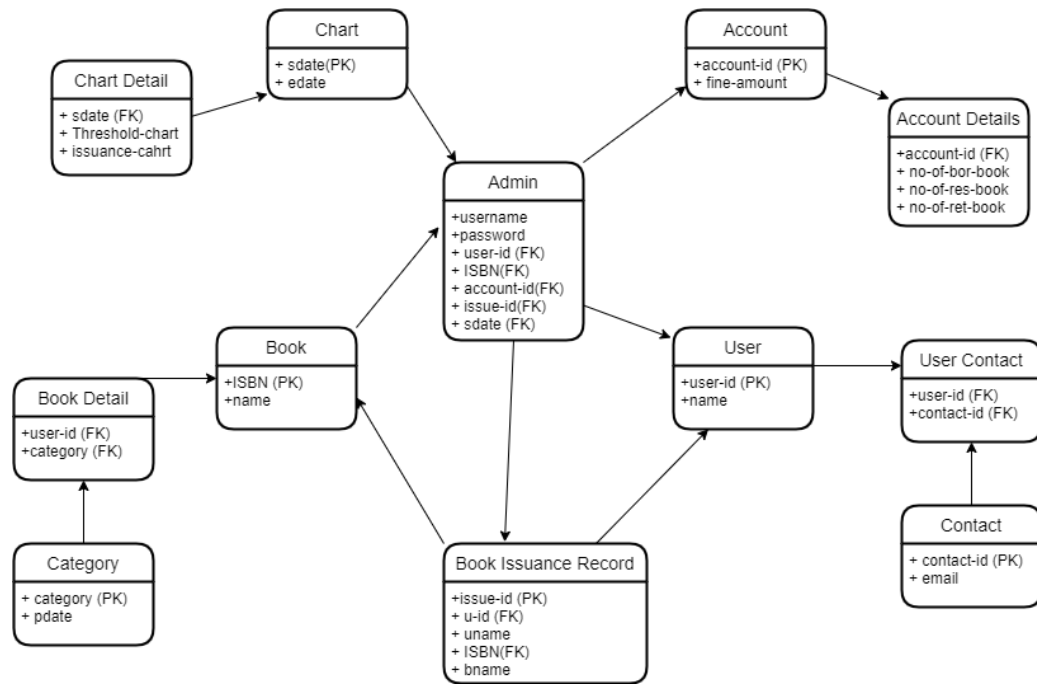


Figure 31: Realtional Schema.

2.8 UML Diagram

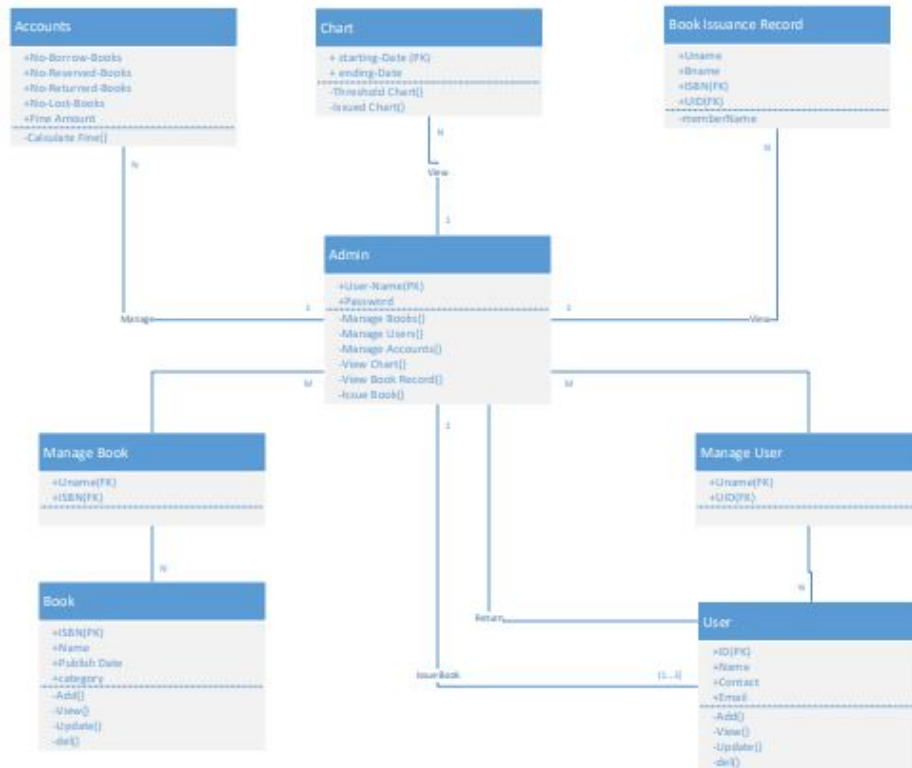


Figure 32: UML.

2.9 DFD Model

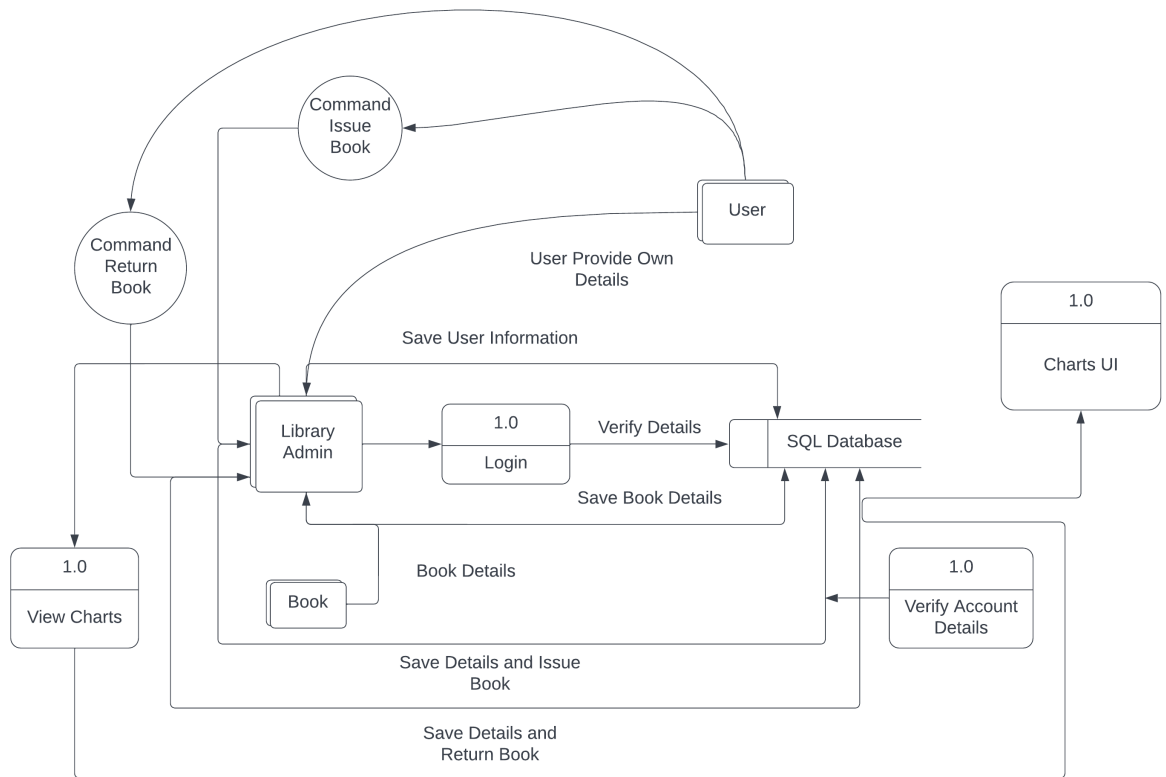


Figure 33: DFD Model.