



Tribhuvan University
Faculty of Humanities and Social Sciences

Book Sharing Platform Using Haversine Algorithm

A project report

Submitted to
Department of College Administration
Triton International College

*In partial fulfillment of the requirements for the Bachelors in Computer
Application*

Submitted By
Nitesh Poudel Reg. No:
Anil Dhamala Reg. No:

Under The Supervision of
Yogesh Deo



Tribhuvan University
Faculty of Humanities and Social Sciences
Triton International College

SUPERVISOR'S RECOMMENDATIONS

I hereby recommend that this project prepared under my supervision by Nitesh Poudel and Anil Dhamala entitled "**SecondChapter: Giving books a second life**" in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

Yogesh Deo

SUPERVISOR

BCA Department

Balkumari, Kathmandu



Tribhuvan University
Faculty of Humanities and Social Sciences
Triton International College

LETTER OF APPROVAL

This is to certify that this project prepared by Nitesh Poudel and Anil Dhamala entitled entitled "**SecondChapter: Giving books a second life**" in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

Yogesh Dev

SUPERVISOR

BCA Department

Triton International College

Milan Baral

CO-ORDINATOR

BCA Department

Triton International College

(Internal Examiner)

(External Examiner)

ABSTRACT

This paper presents a web-based book-sharing platform designed to facilitate the buying and selling of books among book enthusiasts. The system integrates geolocation-based recommendations using the Haversine algorithm, enabling users to discover books accessible within their vicinity. By calculating the distance between users based on their latitude and longitude, the platform ensures that books suggested for purchase are either within a feasible delivery range or available for pickup. The application supports user-friendly functionalities, such as listing books for sale, browsing recommended books, and managing user preferences. The proposed solution aims to create an efficient and community-driven marketplace for book lovers, fostering knowledge sharing and sustainable reuse of books. Key features include geolocation-driven recommendations, secure user authentication, and a streamlined interface for a seamless user experience. The application demonstrates the potential of integrating spatial algorithms into e-commerce platforms to enhance accessibility and user satisfaction .

Keywords : Haversine Algorithm , Php, geolocation

ACKNOWLEDGEMENT

We extend our heartfelt gratitude to all those who have been an integral part of our journey, supporting us through every phase of struggle and success in the realization of this endeavor.

First and foremost, we would like to express our sincere thanks to our college **Triton International College** for providing a sound environment to work on this project and also coordinator **Mr. Milan Baral**, whose guidance and unwavering support have been invaluable.

Our deepest appreciation goes to our supervisor **Mr. Yogesh Deo** whose expertise and mentorship have played a pivotal role in shaping this project.

In conclusion, the successful culmination of this project would not have been possible without the contributions and support of the aforementioned individuals. As we reflect on the challenges and victories that have shaped our path, we are humbled and grateful for the opportunity to learn, grow, and create something meaningful.

Nitesh Poudel

Anil Dhamala

TABLE OF CONTENTS

| | |
|---|------------|
| ABSTRACT | i |
| ACKNOWLEDGEMENT | ii |
| TABLE OF CONTENTS | iii |
| LIST OF FIGURES | v |
| LIST OF TABLES | vi |
| LIST OF ABBREVIATIONS | vii |
| CHAPTER 1: INTRODUCTION..... | 1 |
| 1.1 Introduction..... | 1 |
| 1.3 Objectives | 2 |
| To build a community of book enthusiasts, encouraging interactions and knowledge sharing among users with similar literary interests..... | 2 |
| 1.4 Scope and Limitations..... | 2 |
| 1.5 Report Organization..... | 3 |
| CHAPTER 2: BACKGROUND STUDY AND LITERATURE REVIEW | 4 |
| 2.1 Background Study..... | 4 |
| CHAPTER3: SYSTEM ANALYSIS AND DESIGN | 6 |
| 3.1 System Analysis..... | 6 |
| 3.1.1 Requirement Analysis | 7 |
| 3.1.2 Feasibility Analysis..... | 9 |
| 3.1.3 Data Modeling (ER-Diagram) | 11 |
| 3.1.4 Object Modelling using Class Diagrams | 11 |
| 3.1..5 Process Modelling using Activity Diagrams | 12 |
| 3.2 System Design | 13 |
| 3.2.1 Architectural Design | 13 |
| 3.2.2 Database Schema Design | 14 |
| 3.2.3 Interface Design | 14 |

| | |
|---|-----------|
| 3.3 Algorithm and Its Implementation | 18 |
| Chapter 4: IMPLEMENTATION AND TESTING | 20 |
| 4.1 Implementation | 20 |
| 4.1.1 Tools Used | 20 |
| 4.1.2 Implementation Details of Modules..... | 21 |
| 4.2 Testing..... | 22 |
| 4.2.1. Algorithm Testing..... | 22 |
| 4.2.2. Test Cases for Unit Testing..... | 23 |
| 4.2.2 Test Case for System Testing | 24 |
| CHAPTER 5: CONCLUSION AND FUTURE RECOMMENDATIONS | 26 |
| 5.1 Lesson Learnt / Outcome | 26 |
| 5.2 Conclusion | 27 |
| 5.3 Future Recommendations | 27 |
| REFERENCES..... | 28 |
| APPENDICES | 1 |

LIST OF FIGURES

| | |
|---------------------------------------|----|
| Figure 1 Model..... | 6 |
| Figure 2 Use case Diagram | 8 |
| Figure 3 Gantt Chart | 10 |
| Figure 4 Conceptual ERD | 11 |
| Figure 5 Class diagram | 12 |
| Figure 6 Activity diagram..... | 13 |
| Figure 7 Architectural Design | 13 |
| Figure 8 Database Schema Design | 14 |
| Figure 9 Landing Page | 15 |
| Figure 10 Registration Page..... | 16 |
| Figure 11 Login Page..... | 16 |
| Figure 12 User's Home Page | 17 |
| Figure 13 Admin's Homepage..... | 17 |

LIST OF TABLES

| | |
|---|----|
| Table 4.1 Test case for User registration and Login | 23 |
| Table 4.2 User Login (Same for admin as well) | 24 |
| Table 4.3 Test case for books listing | 24 |
| Table 4.4 Test case for failed books listing | 25 |

LIST OF ABBREVIATIONS

| <u>Abbreviation</u> | <u>Description</u> |
|----------------------------|---|
| CASE | Computer-Aided Software Engineering |
| CSS | Cascading Style Sheets |
| DFD | Data Flow Diagram |
| ER | Entity Relationship |
| ERD | Entity Relationship Diagram |
| HTML | Hyper Text Markup Language |
| ID | Identification |
| IOS OS) | Apple's operating system for mobile device (formerly iPhone |
| JS | JavaScript |
| MS | Microsoft |
| PHP | Hyper Text Preprocessor |
| SQL | Structured Query Language |
| UI | User Interface |
| URL | Uniform Resource Locator |

CHAPTER 1:

INTRODUCTION

1.1 Introduction

The Book Sharing Web Application is a user-friendly platform designed to connect book enthusiasts, making it easy for them to buy, sell, and share books in a dynamic, community-focused environment. It simplifies the book exchange process, allowing users to access books they need or share books they no longer require. By fostering meaningful interactions among book lovers, the platform promotes a sustainable culture of reusing and redistributing books, helping to reduce waste and provide affordable access to literature.

A core innovation of this platform is its integration of the Haversine algorithm, a mathematical method used to calculate the shortest distance between two points on the Earth's surface. When a user lists a book for sale, the system captures their geographical coordinates (latitude and longitude) and stores them in the database. This allows the platform to filter and recommend books to buyers based on proximity, displaying only those books that are available within the seller's specified delivery range. By automating this distance-based matching, the platform ensures that buyers can easily find books they can access without logistical challenges.

This geolocation-driven approach enhances the platform's efficiency and user experience by providing personalized and relevant book recommendations. Buyers save time by browsing listings from nearby sellers, while sellers gain visibility to potential buyers within their delivery range, increasing the likelihood of successful transactions. This practical and scalable system not only streamlines the book-sharing process but also builds a stronger local community of readers, ensuring that the joy of books is accessible to everyone.

1.2 Problem Statements

Existing platforms for book exchange, including online marketplaces and physical bookstores, face several limitations in addressing accessibility and efficiency for

users. These platforms often fail to provide proximity-based recommendations, making it challenging for users to access books that are practically reachable. High delivery costs and delays further hinder affordability and timely access, particularly for second-hand or rare books. Additionally, the lack of geolocation-based solutions and community-driven features results in inefficient user experiences and limited opportunities for fostering connections among book enthusiasts. These challenges underscore the need for an intelligent, location-aware system that enhances accessibility while promoting sustainable book sharing.

1.3 Objectives

The system will give remedies for the problem that are currently being faced by the customers as well as restaurant. Some of the objectives of this system are as follow:

- To enables book searches by title, author, or category and uses the Haversine algorithm for proximity-based recommendations, ensuring access to books within feasible delivery or pickup ranges.
- To promote sustainable book reuse by fostering a community-driven marketplace for buying and selling books.

To build a community of book enthusiasts, encouraging interactions and knowledge sharing among users with similar literary interests.

1.4 Scope and Limitations

Scope:-

The scope defines the boundaries and extent of the project, detailing what it aims to achieve and the areas it will cover. It outlines the features, functionalities, and target audience of the system. For the given project:

Customer Features:

- Customizing orders and viewing billing details.
- Real-time order tracking.

Management Features:

- Adding, updating, and managing menu items.
- Viewing sales reports and analytics.
- Monitoring and streamlining order processing.

Limitations:-

The limitations describe the constraints or areas the project does not address. These might include technological, resource, or time constraints. For the given project:

- **Geolocation Accuracy:** The project relies on the Haversine algorithm for distance calculations, which assumes the Earth is a perfect sphere. This can lead to small inaccuracies, especially for users located at higher latitudes or over long distances...
- **Straight-Line Distance Only:** The project uses straight-line distance for filtering books, ignoring travel paths, road networks, or terrain, which may make some books inaccessible.
- **Location Processing Differences:** The process for determining geolocation can vary between browsers and devices, leading to slight discrepancies in location accuracy for some users.

1.5 Report Organization

Chapter1: This chapter provides an overview of the project, highlighting its significance, context, and purpose. It also notes down the scope and the limitations associated with the project.

Chapter 2: This chapter discusses the background study and literature review related to books sharing systems, highlighting existing solutions, their limitations, and the motivation behind this project.

Chapter 3: This chapter analyzes system requirements through use case diagrams, assesses feasibility, and designs the system, including data modeling, architecture, database schema, interfaces, and physical DFDs.

Chapter 4: This chapter covers project implementation and testing. It discusses tools (CASE tools, programming languages, and databases), details module implementations, and outlines test cases for unit and system testing.

Chapter 5: The final chapter serves as the conclusion of the report. It includes a section on lessons learned and outcomes derived from the project. The conclusion summarizes the key findings and outcomes of the project, highlighting its achievements and contributions. Lastly, future recommendations are provided, suggesting potential areas for improvement or further research.

CHAPTER 2:

BACKGROUND STUDY AND LITERATURE REVIEW

2.1 Background Study

The traditional approach to buying and selling books often involves physical stores, limited local exchanges, or reliance on generic online marketplaces. These methods can be inefficient for book lovers looking to connect within their local communities, as they lack personalized proximity-based recommendations and often involve significant logistical challenges. Furthermore, many existing platforms fail to provide a streamlined process for casual users who wish to buy or sell books without the complexities of online payments or nationwide shipping, leaving a gap for hyperlocal and user-friendly solutions.

This project aims to address these issues by creating a web-based book-sharing platform designed to connect users within their local areas. By leveraging the Haversine algorithm, the system calculates the distance between users' geolocations to recommend books available within a defined delivery range. This ensures users have access to nearby listings, simplifying transactions and promoting sustainable practices. With additional features like user registration, book management, the platform provides an efficient and community-driven solution for book enthusiasts.

2.2 Literature Review

The development of book-sharing web applications has transformed the way people buy, sell, and share books, enhancing accessibility for book lovers and promoting sustainability. These platforms facilitate the exchange of used books, reducing waste and offering a convenient solution for individuals seeking second-hand literature (Smith, 2019). Geolocation integration, using algorithms like the Haversine formula, enables users to find books within specific delivery ranges, improving the efficiency of the book exchange process and offering personalized recommendations (Johnson & Lee, 2018). This method not only makes the process more localized but also reduces delivery times, increasing customer satisfaction and usage (Wang, 2022).

Advancements in web technologies like PHP, JavaScript, and MySQL have led to the creation of scalable and responsive book-sharing applications. These technologies ensure secure management of user data and transactions while providing seamless interactions. The use of relational databases, such as MySQL, supports system reliability and performance, especially when handling large volumes of user-generated content (Taylor, 2017). Furthermore, PHP and JavaScript enhance the interactivity and responsiveness of the platform, making it user-friendly across different devices (GeeksforGeeks, 2023).

For development, the Waterfall methodology has proven effective for structuring the process of creating the book-sharing platform. The Waterfall model allows for clear and structured stages, such as requirements gathering, system design, implementation, and testing, ensuring thorough development at each phase (Sommerville, 2011). This approach helps minimize risks and provides a predictable timeline for project completion. However, it is essential to address challenges such as data security, user privacy, and fraud prevention to ensure long-term platform success and user trust (Evans & Scott, 2019).

By leveraging technologies such as geolocation, data-driven recommendations, and secure online transactions, the Book Sharing Web Application aims to provide a seamless and trusted experience for users. These features contribute to the circular economy by encouraging the exchange of used books, while also fostering a sense of community and environmental responsibility (Anderson, 2021). The project aligns with digital transformation trends, addressing growing demand for convenient and

CHAPTER3:

SYSTEM ANALYSIS AND DESIGN

3.1 System Analysis

The project will be developed using the Waterfall model, which follows a linear and sequential approach to software development. This model involves distinct phases, including requirements gathering, system design, implementation, testing, deployment, and maintenance.

The Waterfall model emphasizes a comprehensive planning phase at the beginning of the project to establish clear requirements and objectives. Each phase progresses in a structured manner, with the completion of one phase serving as the input for the next.

Regarding maintenance, while the primary focus of the project may be on the development and deployment phases, there is a possibility of addressing maintenance and updates in the future. Although the extent of maintenance may be influenced by time constraints, there is recognition of the importance of ongoing support, bug fixes, and potential enhancements to ensure the system's continued functionality and relevance.

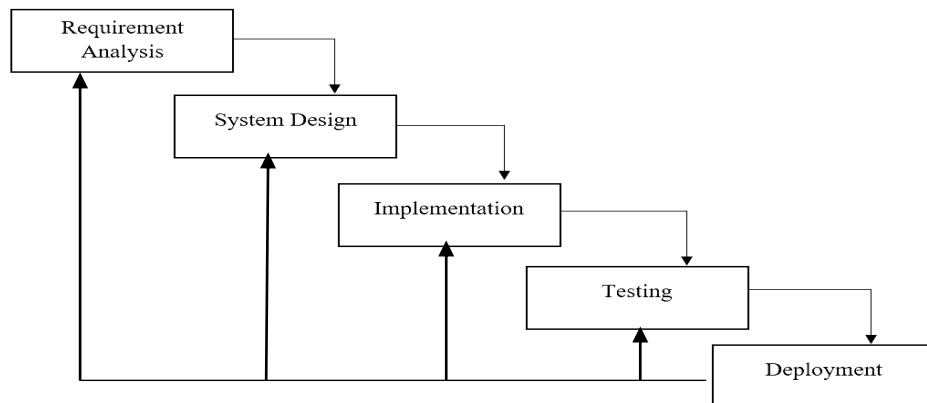


Figure 1 Model for book sharing system

3.1.1 Requirement Analysis

i. Functional Requirements:

As per the system, the following are the functional requirements for different users:

Regular Users:

- User can login with their credentials.
- User can buy and sell books.

Admin Users:

- Admin should be able to manage books.
- Admin should be able to manage users.

System:

- The system should be able to measure distance.
- The system shall provide search result.
- The system should be able to validate inputs

Use Case Diagram

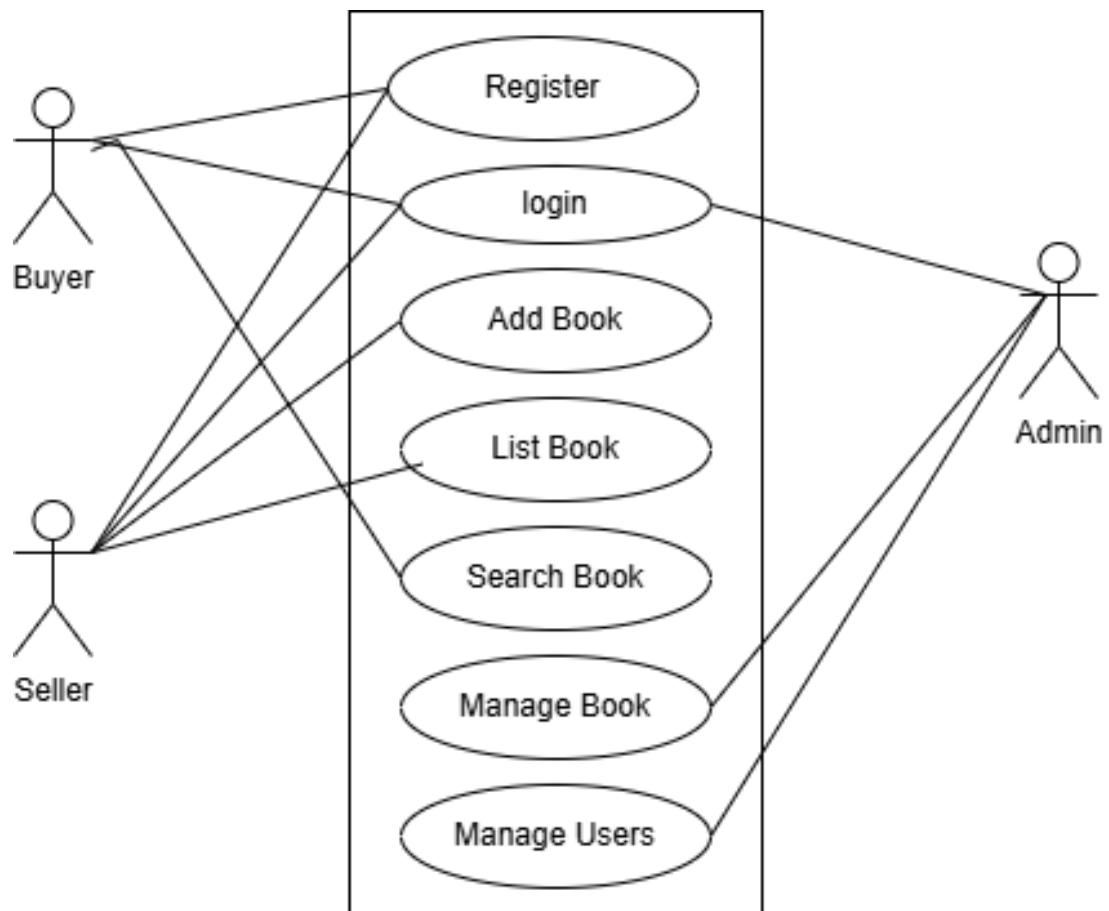


Figure 2 Use case Diagram

ii. Non-Functional Requirements

- **Performance:**

The system should provide fast response times and efficient handling of user interactions. This ensures that users can quickly access the platform, browse without experiencing frustrating delays. A well-performing system contributes to a positive user experience and encourages active participation.

- **Security:**

The system should implement robust security measures to protect user data and ensure the privacy and integrity of user information. This involves using encryption techniques to secure data transmission, implementing user authentication and authorization mechanisms, and safeguarding against common security threats such as SQL injection or cross-site scripting. By prioritizing security, the system instills user trust and confidence, fostering a safe environment for renting vehicles.

- **Usability:**

The system should have a user-friendly interface and intuitive navigation, enabling users to easily access features, list and order dishes. Usability considerations include clear and consistent design, proper information hierarchy, and intuitive interactions. By emphasizing usability, the system ensures that users can effectively engage with the platform, contributing to increased user satisfaction, engagement, and overall adoption.

3.1.2 Feasibility Analysis

- i. **Technical Feasibility Study:**

The system exhibits strong technical feasibility as the necessary hardware and software for development are readily accessible. The project utilizes a suitable programming language, supported by libraries capable of achieving the desired results. Leveraging existing resources for system development and maintenance is feasible.

- ii. **Operational Feasibility Study:**

Operational feasibility is a key strength of the envisioned vehicle rental system. By encompassing all necessary requirements, the system ensures seamless operational functionality. Its user-friendly design contributes to

successful implementation, with administration finding the system intuitive and easy to navigate.

iii. **Schedule Feasibility Study:**

The system is completed within the defined schedule, adhering to project timelines and successfully meeting project milestones. The development process is efficiently managed to prevent any delay beyond the scheduled timeframe.

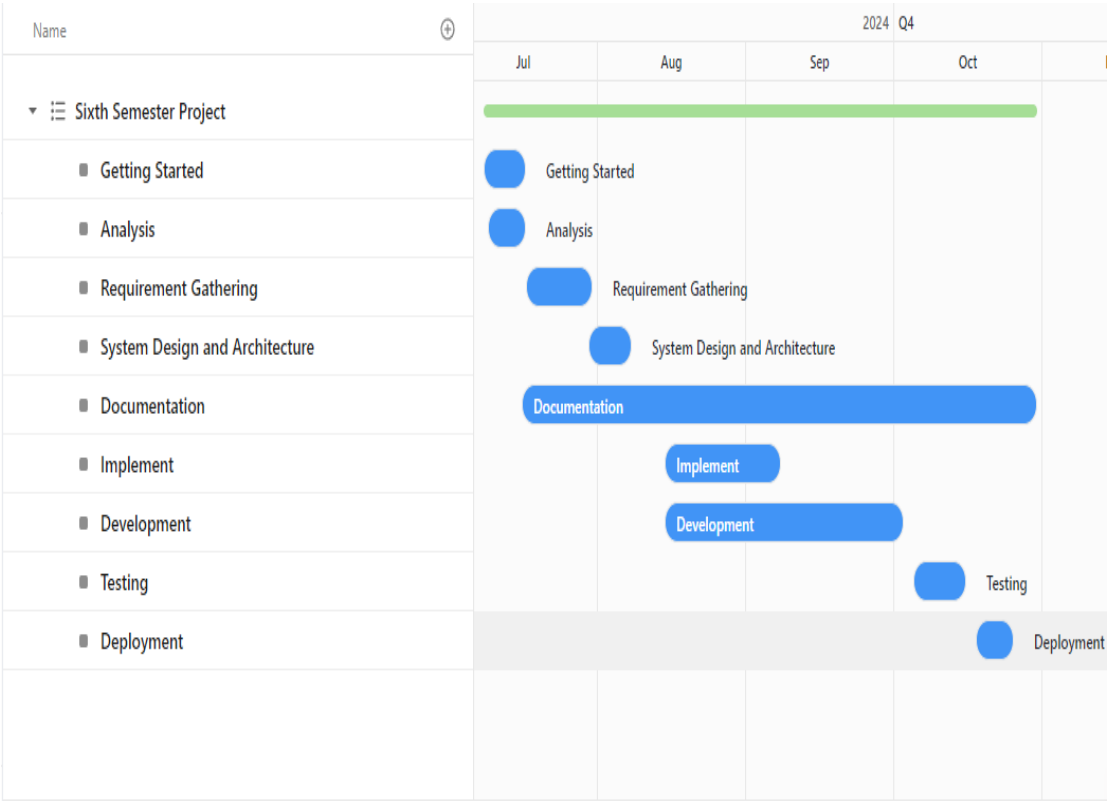


Figure 3 Gantt Chart

3.1.3 Data Modeling (ER-Diagram)

The ER diagram used for the Books is a Conceptual Entity-Relationship Diagram (ERD). As a Conceptual ERD, it focuses on capturing the high-level business concepts and relationships within the system. The diagram presents an abstract representation of the entities, their attributes, and the relationships between them, without delving into specific implementation details.

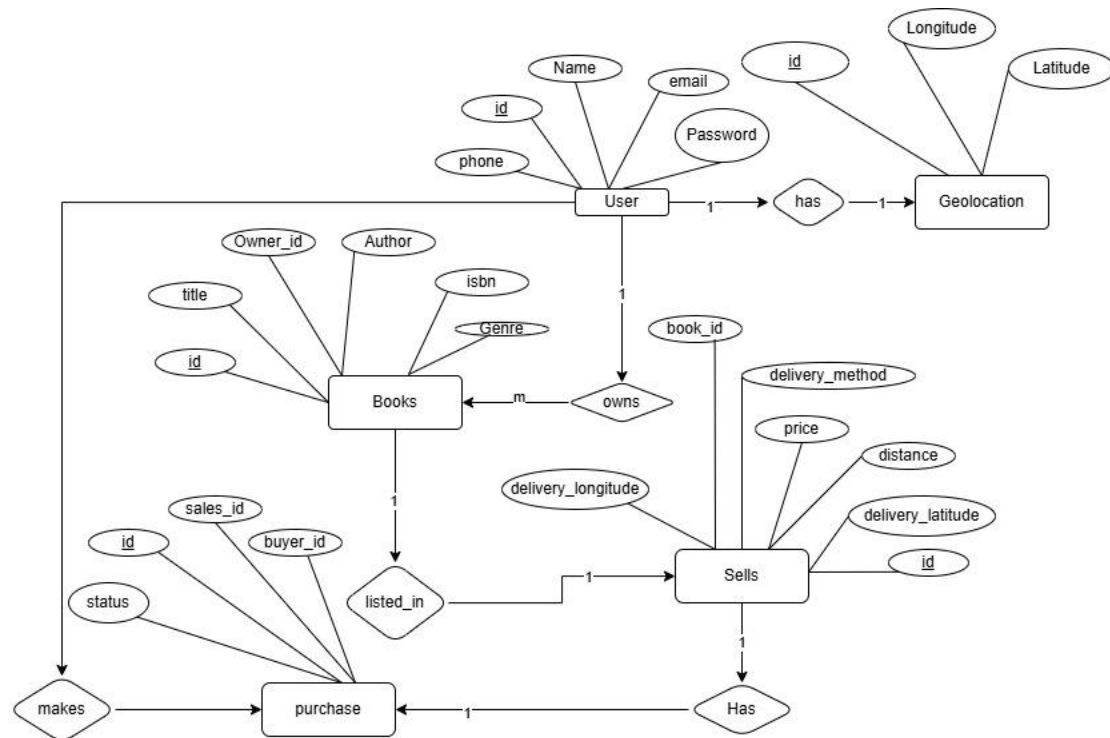


Figure 4 Conceptual ERD

3.1.4 Object Modelling using Class Diagrams

The Object Modeling Class for the Book Sharing Web Application provides a structured blueprint of the system's key components, such as users, books, sales, and geolocation, along with their attributes and methods. It ensures a modular, maintainable, and scalable design by defining clear roles and responsibilities for each class. This approach captures the system's core functionalities, including user management, book listing, sales processing, and distance-based recommendations using geolocation. By focusing on real-world entities and their interactions, the object

model simplifies development and ensures seamless integration of features within the platform.

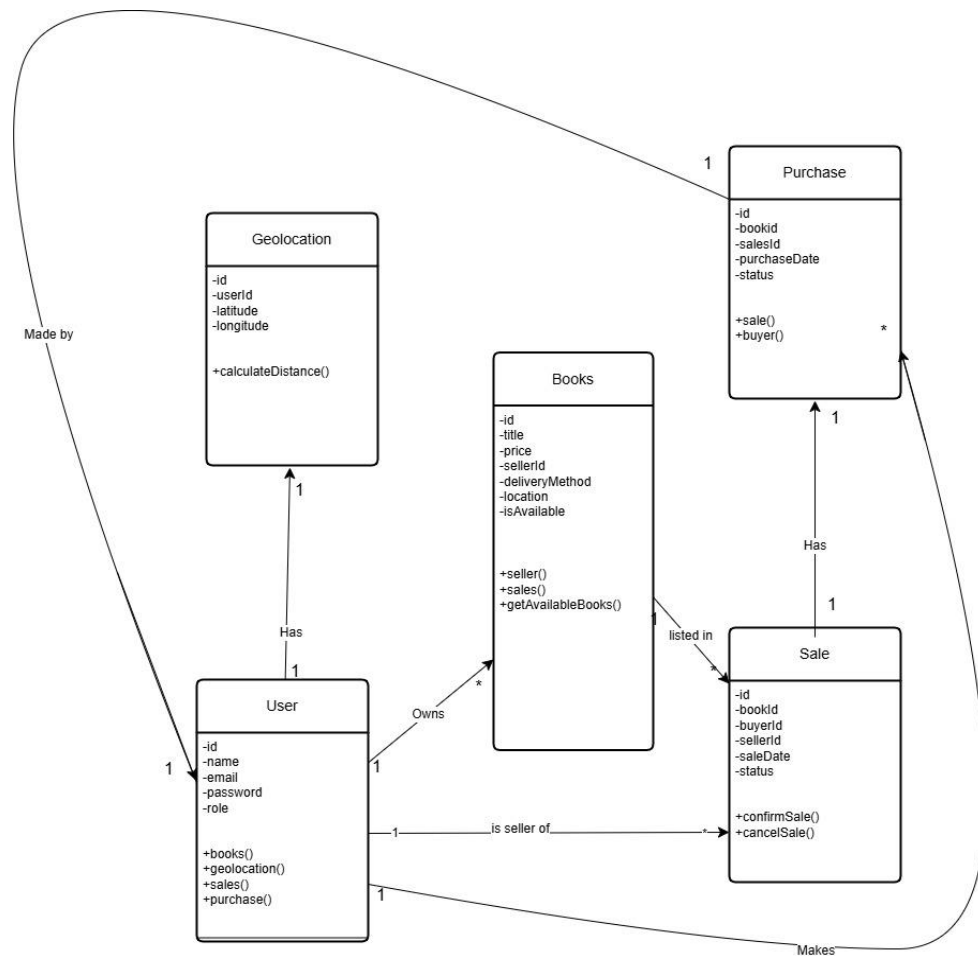


Figure 5 Class Diagram

3.1..5 Process Modelling using Activity Diagrams

Process Modeling is a method used to visually represent and describe the flow of activities, tasks, or steps involved in a particular process or system. It is used to understand, analyze, and improve the efficiency of business processes, workflows, or systems.

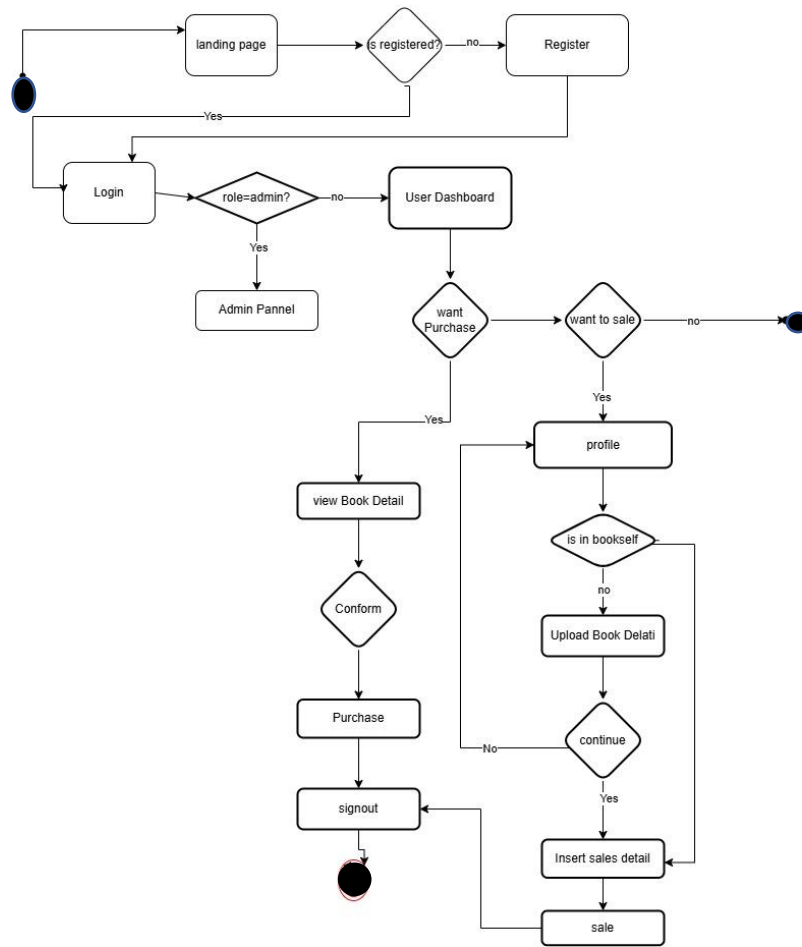


Figure 6 Activity Diagram

3.2 System Design

3.2.1 Architectural Design

The architecture design for the Momotarou Express follows a client-server model. It comprises a database for data storage, a web server for handling requests and processing logic, and a client-side User Interface (UI) for user interaction. This design enables efficient communication and ensures a seamless user experience.

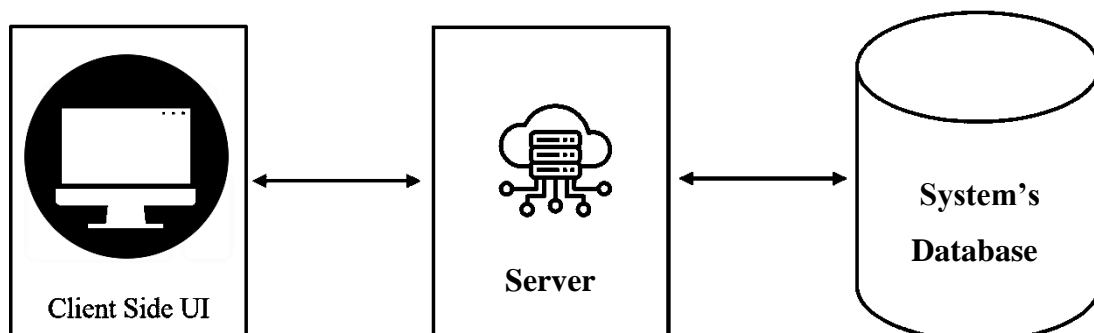


Figure 7 Architectural Design

3.2.2 Database Schema Design

Database schema design for this project involves creating an organized structure to store and manage data efficiently. The schema design defines the tables, their columns, data types, relationships, and constraints. It ensures that the data is organized logically, enabling seamless retrieval and manipulation. The database schema includes tables such as "users", "books", "sales" etc each with their respective columns to capture relevant information .Additionally, appropriate indexing and normalization techniques are employed to optimize performance and reduce data redundancy. The database schema design plays a crucial role in ensuring data integrity, facilitating efficient data access, and supporting the system's overall functionality.

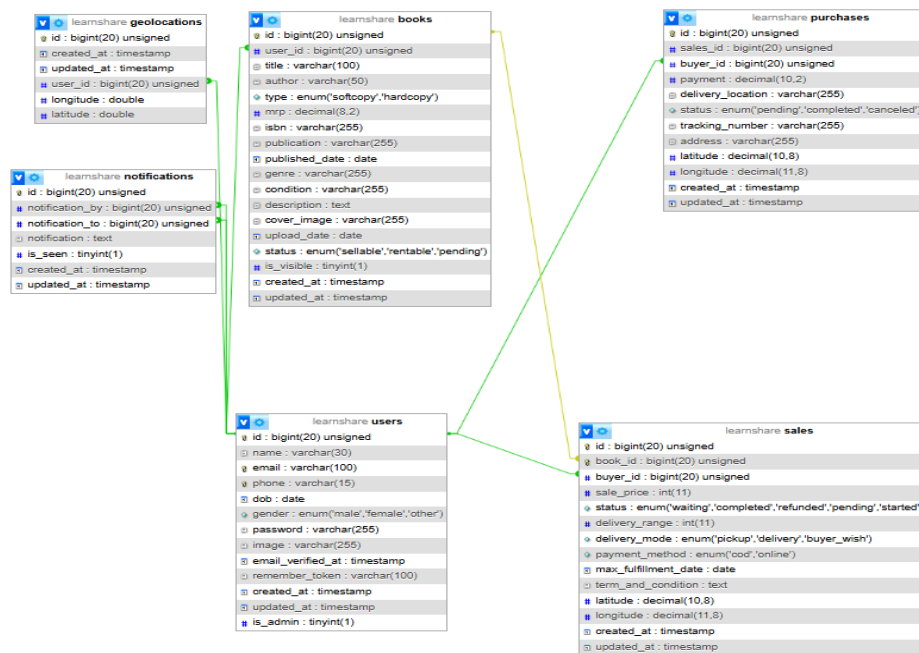


Figure 8 Database Schema Design

3.2.3 Interface Design

The User Interface (UI) for the Book Sharing Web Application is designed to deliver a user-friendly and intuitive experience for book enthusiasts. This critical aspect of

the system's design emphasizes both functionality and visual appeal, incorporating a clean, aesthetically pleasing layout, easy navigation, and interactive elements to engage users effectively. The UI provides seamless access to key features, such as browsing books, managing listings, and processing purchases or sales, ensuring convenience and efficiency at every step.

The User Interface for the Book Sharing Web Application goes beyond mere practicality, embodying thoughtful design principles and a strong focus on user-centricity. It reflects the commitment to creating an interface that harmonizes technology and usability, delivering an experience that not only meets but exceeds user expectations. This careful attention to detail ensures that the platform is accessible and enjoyable for all, fostering a welcoming environment for book lovers to connect and share.

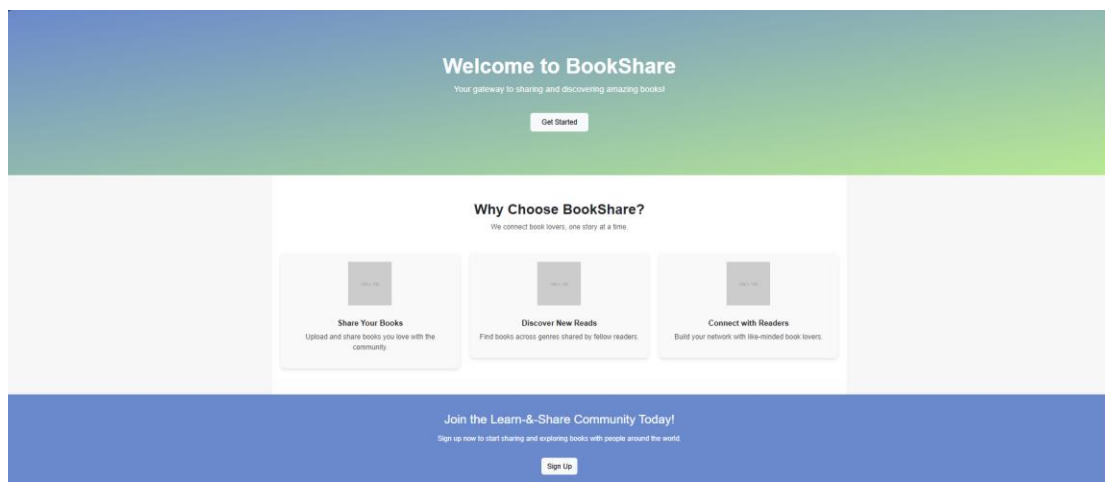
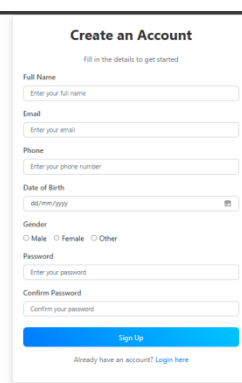


Figure 9 Landing Page



Create an Account
Fill in the details to get started

Full Name
Enter your full name

Email
Enter your email

Phone
Enter your phone number

Date of Birth
dd/mm/yyyy

Gender
☐ Male ☐ Female ☐ Other

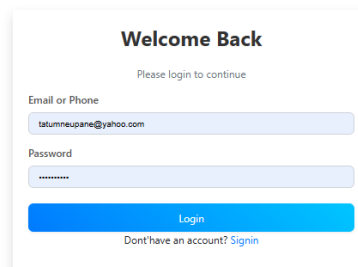
Password
Enter your password

Confirm Password
Confirm your password

[Sign Up](#)

Already have an account? [Login here](#)

Figure 10 Registration Page



Welcome Back
Please login to continue

Email or Phone
tatumneupane@yahoo.com

Password

[Login](#)

Don't have an account? [Signin](#)

Figure 11 Login Page

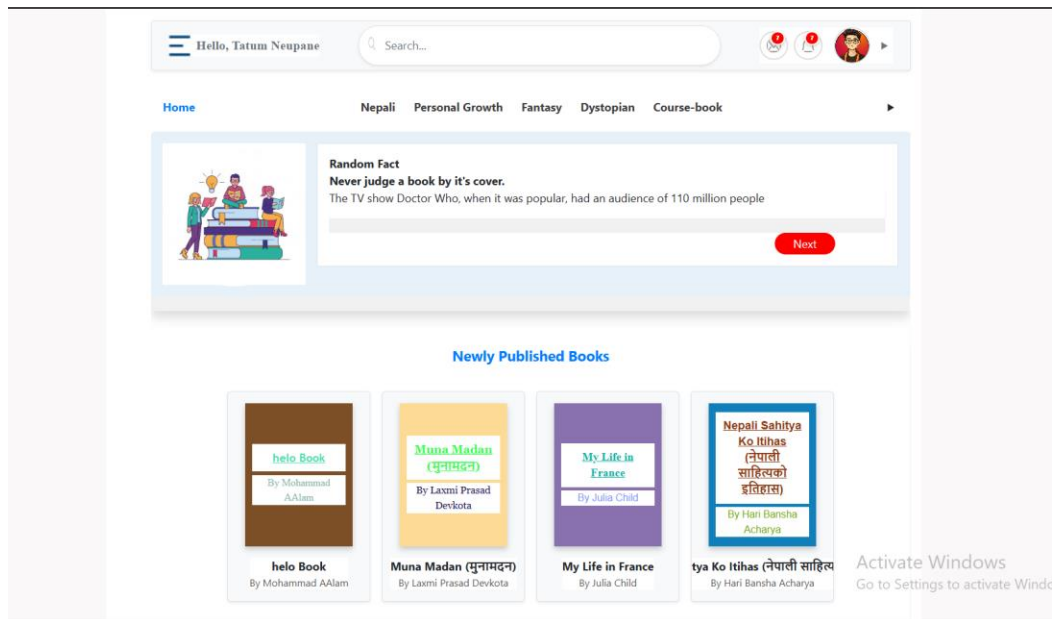


Figure 12 User's Home Page

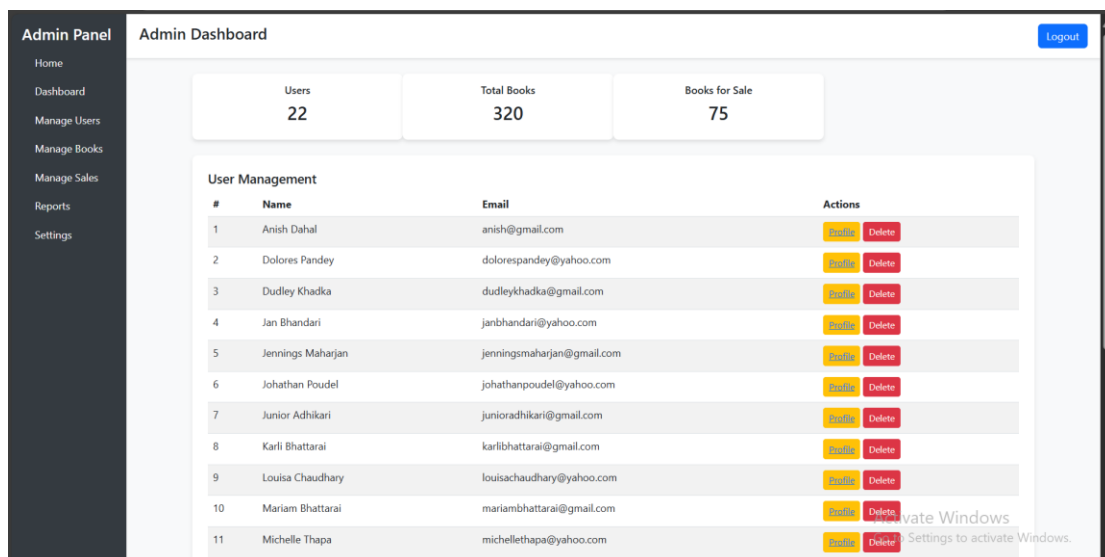


Figure 13 Admin's Homepage

3.3 Algorithm details

The project implements an algorithm to show books based on their accessibility, user proximity, and delivery preferences. The algorithm retrieves data from the database, including book details, seller geolocations, and delivery ranges. Using this data, the system filters books available within the defined delivery range of the buyer and show them.

Calculate Distance Function

The function calculateDistance() computes the distance between two geographic point on Earth using their latitude and longitude co-ordinate. It implements the Haversine formula, a widely used mathematical formula for calculating great-circle distance .

$$a = \sin^2(\Delta\phi/2) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2(\Delta\lambda/2)$$

$$c = 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R \cdot c$$

Where:

ϕ_1, ϕ_2 : Latitudes of points 1 (User A) and 2 (User B) in radians

$\Delta\phi = \phi_2 - \phi_1$: Difference in latitudes in radians

$\Delta\lambda = \lambda_2 - \lambda_1$: Difference in longitudes in radians

$R = 6371\text{km}$: Earth's radius

d : Distance between the two points

Steps in the Algorithm

Convert Degrees to Radians:

$$\phi_1 = \text{deg2rad}(34.0522) = 0.5944\text{radians}, \phi_2 = \text{deg2rad}(36.7783) = 0.6419\text{radians}$$

$$\lambda_1 = \text{deg2rad}(-118.2437) = -2.0637\text{radians}, \lambda_2 = \text{deg2rad}(-119.4179) = -2.0842\text{radians}$$

Calculate Differences

$$\Delta\phi = 0.6419 - 0.5944 = 0.0475\text{radians}$$

$$\Delta\lambda = -2.0842 - (-2.0637) = -0.0205\text{radians}$$

Apply the Haversine Formula

$$a = \sin^2(0.0475/2) + \cos(0.5944) \cdot \cos(0.6419) \cdot \sin^2(-0.0205/2)$$

$$a \approx 0.0011$$

$$c = 2 \cdot \operatorname{atan2}(\sqrt{0.0011}, \sqrt{1-0.0011}) \approx 0.0665$$

Calculate Distance

$$d = 6371 \cdot 0.0665 \approx 423 \text{ km}$$

Chapter 4:

IMPLEMENTATION AND TESTING

4.1 Implementation

4.1.1 Tools Used

Front-end Tools:

- **HTML:**

HTML (Hypertext Markup Language) is used to structure and present content on web pages. It defines the elements and tags that make up the user interface of Momotarou Express.

- **CSS:**

CSS (Cascading Style Sheets) is used to style and enhance the appearance of HTML elements. It is employed to create visually appealing layouts, apply colors, fonts, and other visual properties to Momotarou Express UI.

- **JavaScript:**

It enables the implementation of validation logic directly on the user's browser, providing immediate feedback and error checking for user inputs in the Momotarou Express.

Backend Tool:

- **PHP(laravel):**

PHP (Hyper-text Pre-processor/ Personal Home Page) is a server-side scripting language used in this project for server-side processing and dynamic content generation. It handles database interactions, server-side validations, and business logic implementation.

Server:

- **Apache Server:**

Apache Server is an open-source web server software used to host and serve Momotarou Express web application. It handles incoming HTTP requests and serves the corresponding web pages to users' browsers.

Database:

- **MySQL:**

MySQL is a popular open-source relational database management system used to store and manage data in this project. It provides a reliable and efficient solution for storing user's profiles, books, sales record, purchase record and other relevant information.

Documentation Tools:

- **MS Word:**

Microsoft Word is a widely used word processing application used for creating comprehensive project documentation, including project plans, requirements, user manuals, and other project-related documents.

- **Draw.io:**

Draw.io is used to design system architecture, flowcharts, and other diagrams. It is utilized in the project to create diagrams illustrating the system's components, interactions, and data flows.

4.1.2 Implementation Details of Modules

1. **Users Registration:**

The user registration module is responsible for facilitating the sign-up process and creating users accounts. When user decides to register, they provide their name, email, phone number, address and password. The module performs input validation to ensure the accuracy and completeness of the provided information. It also checks for the uniqueness of the username to avoid duplicate registrations. Once validated, the user data is securely stored in the database with a unique user ID.

2. **User Login:**

The user login module allows registered users to access their accounts. User provide their login credentials, including their email and password, on the login form. The module verifies the provided credentials against the stored user data in the database. If the credentials are valid, the user is granted access to their account and redirected to the user's dashboard. To maintain user authentication during their session, session management techniques like cookies and session has been employed.

3. Books Management:

This module enables admin to list new dishes under different categories. The module performs input validation to ensure the completeness of the post data. Once validated, the post data is stored in the database, associated with the corresponding user and relevant categories. Also admin can delete the dishes.

4. Admin Dashboard:

The admin dashboard serves as a centralized control panel for system administrators. It provides a comprehensive overview of users and dishes.

5. Order Management:

Admin can manage order by accepting and rejecting. Users can get the notification about the status in their respective email.

4.2 Testing

Different training and testing datasets are used to perform system testing, aiming to assess the accuracy of the system's summaries. Throughout the system's development phase, multiple rounds of testing are conducted. The testing process follows this sequence:

4.2.1. Algorithm Testing

Input Values: Latitude and longitude for Point 1: 40.7128, -74.0060 40.7128, -74.0060 40.7128, -74.0060.

Latitude and longitude for Point 2: 34.0522, -118.2437 34.0522, -118.2437 34.0522, -118.2437.

Conversion:

Convert the latitude and longitude values to radians

$$\phi_1 = \text{deg2rad}(40.7128) = 0.710572 \text{ radians}$$

$$\lambda_1 = \text{deg2rad}(-74.0060) = -1.291648 \text{ radians}$$

$$\phi_2 = \text{deg2rad}(34.0522) = 0.594323 \text{ radians}$$

$$\lambda_2 = \text{deg2rad}(-118.2437) = -2.063742 \text{ radians}$$

Delta Calculations

Compute differences:

$$\Delta\phi=\phi_2-\phi_1=-0.116249$$

$$\Delta\lambda=\lambda_2-\lambda_1=-0.772094$$

Calculate

$$a=\sin^2(-0.116249/2)+\cos(0.710572)\cdot\cos(0.594323)\cdot\sin^2(-0.772094/2)$$

$$a\approx 0.092410$$

Angular Distance (c)

$$c=2*\arctan2(\sqrt{0.092410}, \sqrt{1-0.092410})$$

$$c\approx 0.617759$$

Distance (D)

Multiply by Earth's radius:

$$D=6371\cdot 0.617759\approx 3939.84\text{km}$$

4.2.2. Test Cases for Unit Testing

User Registration

Table 4.1 Test case for User registration and Login

| S.N | Test name | Input | Expected Outcome | Actual Output | Test Result |
|-----|--|------------------------|-----------------------------|-------------------------|-------------|
| 1 | Opening Application | http://localhost:8080/ | Landing Page | Landing Page | Passed |
| 2 | Enter Invalid details or miss some details in the form | Blank email Address | The email field is required | Email field is Required | Passed |

| | | | | | |
|---|---------------------------------|---------------------------------------|---|---|--------|
| 3 | Enter Valid Details In the Form | All Fields Filled with correct format | Registration Success, redirect to login page. | Registration Success, redirect to login page. | Passed |
|---|---------------------------------|---------------------------------------|---|---|--------|

Table 4.2 User Login (Same for admin as well)

| S.N | Test name | Input | Expected Outcome | Actual Output | Test Result |
|-----|--|---|-------------------------|-------------------------|-------------|
| 1 | Opening Application | http://localhost:8080/ | Index Page | Index Page | Passed |
| 2 | Enter invalid Username and Or password | Blank email Address A letter in Phone Number | Invalid Login Details | Invalid Login Details | Passed |
| 3 | Enter Valid Details In the Form | All Fields Filled with correct information | Login success Dashboard | Login success Dashboard | Passed |

4.2.2 Test Case for System Testing

Test Case for Listing Dishess (Success)

Table 4.3 Test case for Book listing

| | |
|-------------|-----------------------------|
| Test case 1 | Dishes listed successfully. |
| Test Data | Add all book details |

| | |
|-----------------|--------------------------|
| Expected result | book listed Successfully |
| Test Result | Passed |

Table 4.4 Test case for failed books listing

| | |
|-----------------------|---|
| Test case 1 | Book Listing Unsuccessful |
| Test Data (Incorrect) | Add Book detail Trun off location access |
| Expected result | "Required longitude and latitude" |
| Test Result | Passed |

CHAPTER 5: CONCLUSION AND FUTURE RECOMMENDATIONS

5.1 Lesson Learnt / Outcome

In this project, several valuable lessons were learnt, and they can be summarized as follows:

- **Learning and Knowledge Gain:** Each project provides an opportunity to learn and gain knowledge in various aspects. Through this particular project, we were able to acquire new skills and insights.
- **Problem-Solving Skills:** The project presented numerous challenges that helped to develop problem-solving skills. We learned how to identify different issues within the system and implement solutions to overcome them.
- **Communication and Writing Skills:** Effective communication is crucial in any project. Through this project, the team improved their communication skills, both in verbal interactions and written documentation. They learned to prepare proposals and project-related documentation. Additionally, they gained proficiency in using various case tools for diagrams like use case diagrams, schema diagrams, data flow diagrams, and ER diagrams.
- **Time Management:** One of the most important lessons learned from this project was the significance of time management. The team realized the importance of prioritizing tasks based on the complexity of system components. This skill helped us optimize the workflow and meet project deadlines efficiently.

5.2 Conclusion

In conclusion, the development of the Book Sharing Web Application has been an impactful endeavor, offering a dynamic platform for connecting book enthusiasts in a sustainable and community-focused manner. The project incorporated a comprehensive approach, including the design and implementation of a robust database, user authentication system, and core functionalities like book listing, geolocation-based recommendations, and sales management. Through meticulous planning and execution, the platform successfully bridges the gap between buyers and sellers, enabling an efficient exchange of books.

With its user-friendly interface and seamless integration of essential features, the system provides an intuitive and reliable experience for users. This collaborative effort leveraged advanced technologies such as PHP, MySQL, and the Haversine algorithm, along with the structured Waterfall methodology, to deliver a scalable and secure application. The Book Sharing Web Application exemplifies the potential of technology and teamwork in fostering a sustainable book-sharing ecosystem while strengthening community bonds.

5.3 Future Recommendations

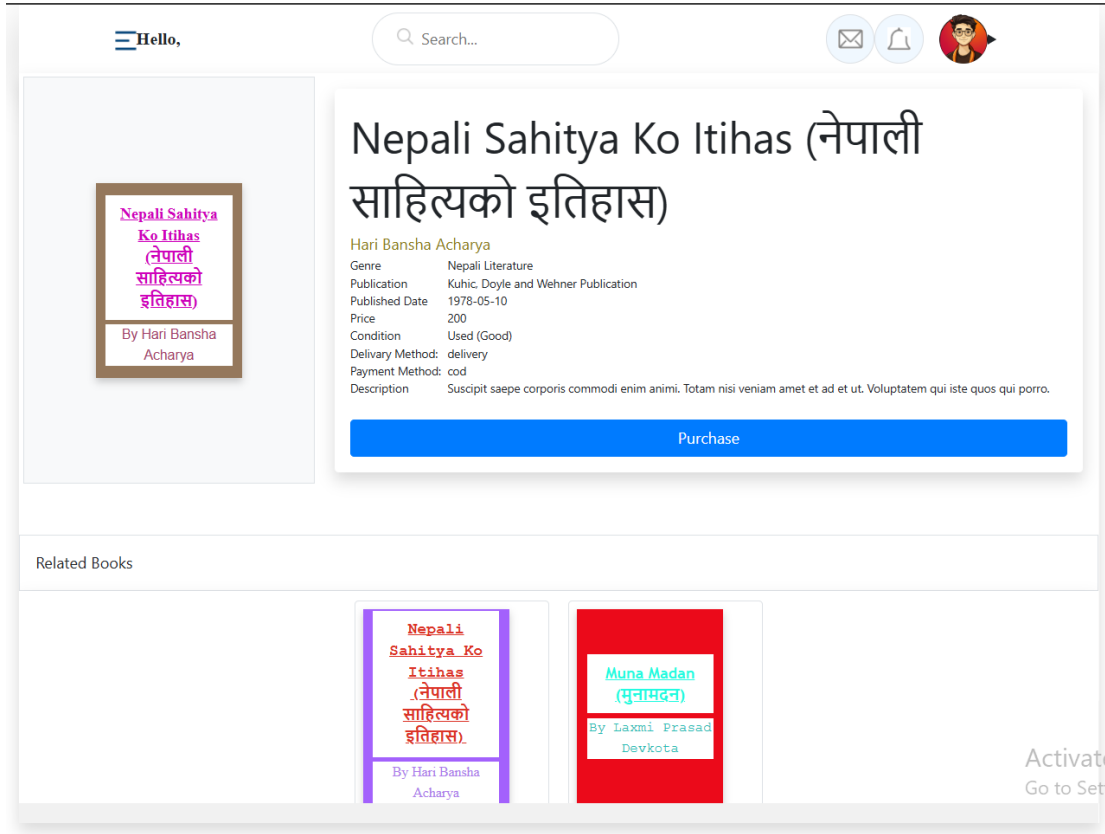
Some of the future recommendations for this system are:

- Recommendations based on user's behavior, user's ranking and reviews.
- Include a formidable dashboard and advanced report generation.
- Make an application native to android and IOS.
- Payment Gateway Integration.

REFERENCES

- [1] Smith, A. (2019). The rise of book-sharing platforms: Impact on sustainability and book consumption. *Journal of Digital Media*, 45(3), 234-248.
- [2] Johnson, M., & Lee, S. (2018). Geolocation-based book-sharing: Enhancing user experiences with Haversine algorithms. *International Journal of Mobile Computing*, 30(2), 89-104.
- [3] GeeksforGeeks. (2023). PHP and MySQL for building scalable applications. Retrieved from <https://www.geeksforgeeks.org>
- [4] Taylor, R. (2017). Database management for scalable web applications. *Tech Publishing*, 12(4), 56-67.
- [5] Sommerville, I. (2011). *Software Engineering* (9th ed.). Boston: Addison-Wesley.

APPENDICES



Random Book

by Niteu Jdoa

Genre: Historical Fiction, Political Science

Publication: ntspublicatins

Published Date: 2024-11-11

Condition: Used (Good)

Description: hu ehf uf aof aoi fruijk kjfhia

Random Book

By Niteu Jdoa

Sale Price:

Enter Sale Price

Fulfillment Method:

Select Fulfillment Method

Max Fulfillment Days:

Enter Max Fulfillment Days

Submit for Sale

Hello,

Search...

×

My Life in France

Julia Child

Genre

Cooking and Food

Publication

Langosh, Glover and Dickinson Publication

Published Date

1977-07-06

Condition

Like New

Description

Recusandae excepturi velit nesciunt ad voluptatem voluptatum id. Nisi voluptates aliquam cupiditate ab officiis rem. Autem architecto ea minus laborum eos eos.

Price

300

Book id :271

You shoud pay on delivery

For Delivery

Delivery Address

phone Number

85.37018799999998

27.6726342

Confirm Now


127.0.0.1:8000/home

Hello, Nitesh Poudel

Search...

Home

NepaliPersonal GrowthFantasyDystopianCourse-book



Random Fact

Never judge a book by it's cover.

The longest chapter in the Bible is Psalm 119, which is 176 verses

Next

Newly Published Books

Muna Madan
(मुनामदन)

By Laxmi Prasad Devkota

Muna Madan (मुनामदन)

By Laxmi Prasad Devkota

My Life in France

By Julia Child

My Life in France

By Julia Child

Nepali Sahitya Ko Itihas (नेपाली साहित्यको इतिहास)

By Hari Bansha Acharya

tya Ko Itihas (नेपाली साहित्य

By Hari Bansha Acharya

Neverwhere

By Neil Gaiman

Neverwhere

By Neil Gaiman

Activate Window
Go to Settings to activa

Welcome Back

Please login to continue

Email or Phone

tatumneupane@yahoo.com

Password

Login

Don't have an account? [Signin](#)

Create an Account

Fill in the details to get started

Full Name

Enter your full name

Email

Enter your email

Phone

Enter your phone number

Date of Birth

dd/mm/yyyy



Gender

☐ Male ☐ Female ☐ Other

Password

Enter your password

Confirm Password

Confirm your password

Sign Up

Already have an account? [Login here](#)

Activate Windows

Go to Settings to activate Windows.