



**Readers Circle: A mobile application for buying, selling, and lending used books in Bangladesh**

Session: Spring - 2025

Submitted By

**MUHAMMAD QUAYES ABDULLAH**  
ID: 0242310012091029  
Department of Computing & Information System  
Daffodil International University

Supervised By

**Mr. Md. Faruk Hosen**  
Lecturer  
Department of Computing & Information System  
Daffodil International University



**Daffodil**  
*International*  
**University**



**PROJECT BASED  
LEARNING**

Department of Computing and Information System  
Daffodil International University.  
Dhaka, Bangladesh

Submission Date: 31/05/2025

## APPROVAL

This Project titled “**Readers Circle; A mobile application for buying, selling, and lending used books in Bangladesh**” " submitted by **MUHAMMAD QUAYES ABDULLAH**, ID No: **0242310012091029** to the Department of Computing and Information Systems, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information Systems and approved as to its style and contents. The presentation has been held on 31-05-2025.

### BOARD OF EXAMINERS



**Md Sarwar Hossain Mollah**  
**Associate Professor and Head**  
Department of Computing & Information Systems  
Faculty of Science & Information Technology  
Daffodil International University

**Chairman**



**Md. Nasimul Kader**  
**Assistant Professor**  
Department of Computing & Information Systems  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**



**Md. Mehedi Hassan**  
**Lecturer (Senior Scale)**  
Department of Computing & Information Systems  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**



**Ahmed Saif Reza**  
**Managing Director & Chief Technology Officer**  
Medico Bio Limited

**External Examiner**

# Declaration

I hereby declare that; this project has been done by me under the supervision of **Mr. Md. Faruk Hosen (Lecturer)**, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like; B.Sc., M.Sc., Diploma, or other qualifications.

**Supervised By**



---

**Mr. Md. Faruk Hosen**  
Lecturer  
Department of CIS  
Daffodil International University

**Submitted By**



---

Name: **MUHAMMAD QUAYES ABDULLAH**  
ID: **0242310012091029**  
Department of CIS  
Daffodil International University

## **Abstract**

The driving concept of the "Readers Circle" project is to create a network of readers, where it will be possible to borrow, lend, or buy books on the digital platform. It's a smartphone app focused on cost, sustainability, and accessibility; solving problems like the high cost and limited availability of specific books. The app ensures a seamless experience through state-of-the-art tech, and is powered by Firebase backend and Flutter for cross-platform development.

We investigate the platform design and implementation and describe the objectives and approaches, the resources implemented, etc. That in turn raises awareness to the social and environmental impact of the project, for the community to participate and for it to adopt sustainable reading habits. In addition to solving practical problems, "Readers Circle" can assist in forming a knowledge society.

The results of extensive testing as well as feedback will be presented here in the context of a comprehensive analysis of the system architecture, the database design, and the user interfaces. The initiative is an important step toward a linked and sustainable literary environment because of its creative approach, which provides a scalable solution that can be adjusted to other places.

## **Acknowledgements**

First and foremost, I must give particular thanks to **Almighty Allah**, without whose assistance and kindness I could not have progressed this far.

For his constant support, enlightening criticism, and encouragement during this assignment, I would like to sincerely thank my supervisor, **Mr. Md. Faruk Hosen sir**. Special gratitude is also extended to our cherished guardian, our department head, **Mr. Md. Sarwar Hossain Mollah sir**, and my course adviser, **Mr. Md. Mehedi Hasan sir**. The "Readers Circle" platform has been greatly influenced by their knowledge and assistance.

I am also grateful to my institution, **Daffodil International University**, and its other all faculty members for providing the resources and knowledge necessary to undertake this project. Special thanks to my peers and friends, who offered constructive criticism and encouragement at every stage of development.

Lastly, I would want to express my sincere gratitude to my family for their tolerance, comprehension, and unwavering support amid the difficulties encountered on this trip. Their steadfast support was essential to the success of our initiative. I appreciate everyone's faith in me and my attempt.

# Table of Contents

Abstract.....	iii
Acknowledgements.....	iv
Chapter 1: Introduction.....	1
1.1 Background .....	1
1.2 Problem Statement .....	2
1.3 Possible Solutions.....	2
1.4 Description of the Project .....	3
1.5 Key Features .....	3
1.6 Technology Stack .....	4
1.7 Impact.....	4
1.8 Document Context.....	4
Chapter 2: Initial Study .....	5
2.1 Initial Concept.....	5
2.1.1 Market Viability .....	5
2.2 Brief Description of the Project.....	7
2.3 Aims, Objectives, and Scopes of the Project.....	7
2.4 Targeted Audience .....	8
2.5 Hardware and Software Requirements .....	8
2.6 Human Resources .....	9
2.7 Functional Requirements.....	9
2.8 Non-Functional Requirements.....	10
2.9 Exploration and Engineering .....	10

2.10 Feasibility Studies .....	11
2.10.1 Operational Feasibility .....	11
2.10.2 Technical Feasibility.....	12
2.10.3 Economic Feasibility .....	12
2.10.4 Social Feasibility.....	12
2.10.5 Environmental Feasibility .....	13
2.11 Overall Feasibility Conclusion.....	13
Chapter 3: Literature Review.....	14
3.1 Description .....	14
3.2 Discussion on Problem Domain Based on Published Articles.....	14
3.3 Solving the Problem Using Published Papers as a Basis.....	15
3.4 Recommended Approach .....	16
Chapter 4: Methodology .....	16
4.1 Introduction to Software Development Life Cycle (SDLC).....	16
4.2 Overview of SDLC Methodologies .....	17
4.3 Selected SDLC Methodology for "Readers Circle" .....	23
4.4 Methodology Phases .....	24
4.5 Implementation Plans.....	25
4.6 Conclusion .....	25
Chapter 5: Planning .....	26
5.1 Introduction.....	26
5.2 About the Project Plan .....	26
5.3 Work Breakdown Structure (WBS).....	27
5.4 What is Time Duration and Time Boxing .....	28

<b>5.5 Time Box .....</b>	<b>28</b>
<b>5.6 Resource Allocation Table.....</b>	<b>30</b>
<b>5.7 What is a Gantt Chart and Why Use It for This Project .....</b>	<b>30</b>
<b>5.8 Require Testing.....</b>	<b>31</b>
<b>5.8 Functional Testing.....</b>	<b>Error! Bookmark not defined.</b>
<b>5.9 Non-Functional Testing.....</b>	<b>32</b>
<b>5.10 Conclusion .....</b>	<b>33</b>
<b>5.11 Risk Management .....</b>	<b>33</b>
<b>5.11 Risk Identification.....</b>	<b>33</b>
<b>5.12 Risk Precaution.....</b>	<b>34</b>
<b>5.14 Change Management.....</b>	<b>35</b>
<b>5.15 Factors That Might Cause Change.....</b>	<b>36</b>
<b>5.16 DSDM Atern and Welcoming Change .....</b>	<b>36</b>
<b>5.17 Conclusion .....</b>	<b>37</b>
<b>5.18 Quality Management.....</b>	<b>37</b>
<b>5.19 Rules Applied to Maintain Quality .....</b>	<b>38</b>
<b>5.20 DSDM Atern Standard Quality Measures.....</b>	<b>38</b>
<b>5.21 Quality Plan.....</b>	<b>39</b>
<b>5.22 Conclusion .....</b>	<b>40</b>
<b>Chapter 6: Feasibility Study .....</b>	<b>41</b>
<b>6.1 Introduction.....</b>	<b>41</b>
<b>6.2 All Possible Types of Feasibility .....</b>	<b>41</b>
<b>6.2.1 Economic Feasibility .....</b>	<b>41</b>
<b>6.2.2 Technical Feasibility.....</b>	<b>43</b>

6.2.3 Operational Feasibility .....	43
6.2.4 Social Feasibility.....	45
6.2.5 Environmental Feasibility .....	45
6.2.6 Cost-Benefit Analysis.....	45
6.2.7 Conclusion .....	46
<b>Chapter 7: Foundation .....</b>	<b>47</b>
7.1 Introduction.....	47
7.2 Overall Requirement List .....	47
7.4 Non-Functional Requirements.....	48
7.5 Problem Area Identification .....	49
7.6 Observation.....	49
7.7 Possible Solution.....	49
7.8 BPMN (Business Process Modeling and Notation) .....	50
7.9 Technology to Be Implemented.....	51
7.10 Recommendations and Justifications .....	53
7.11 Conclusion .....	53
<b>Chapter 8: Exploration.....</b>	<b>54</b>
8.1 Introduction.....	54
8.2 Old Full System Use Case .....	54
8.3 New System Full Activity Diagram .....	56
8.4 Admin and Client Activity Diagrams .....	57
8.5 Prioritised Requirement List (PRL) .....	59
8.6 Prototype of the New System .....	60
<b>Chapter 9: Engineering.....</b>	<b>62</b>

9.1 Introduction.....	62
9.2 New System Modules .....	62
9.2.1 Use Case Diagram .....	62
9.2.2 Class Diagram.....	63
9.2.3 ERD Diagram.....	64
9.2.4 Sequence Diagram .....	65
9.2.5 Component Diagram .....	66
9.2.6 Deployment Diagram.....	67
9.2.7 System Interface .....	68
9.3 New System Usage Status .....	72
9.4 Conclusion .....	72
Chapter 10: Deployment/Development Phase .....	73
10.1 Introduction.....	73
10.2 Development Approach .....	75
10.3 Possible Problem Breakdown .....	75
10.4 Conclusion .....	77
Chapter 11: Testing Phase .....	78
11.1 Introduction.....	78
11.2 Types of Test Case Scenarios and Importance .....	78
11.3 Test Plan Acceptance Table .....	79
11.4 Unit Testing.....	79
11.5 Module Testing .....	80
11.6 Integration Testing .....	80
11.7 Acceptance Testing.....	81

11.8 Security Testing.....	81
11.9 Usability Testing.....	82
11.10 Conclusion.....	82
<b>Chapter 12: Implementation Phase.....</b>	<b>83</b>
12.1 Introduction.....	83
12.2 System Deployment.....	83
12.3 Training.....	84
12.4 Big Bang Implementation.....	84
12.5 Scaling.....	84
12.6 Load Balancing.....	85
12.7 Recommended Implementation Process.....	85
12.8 Conclusion.....	85
<b>Chapter 13: Critical Appraisal and Evaluation.....</b>	<b>86</b>
13.1 Project Evaluation.....	86
13.1.1 Success Rate.....	86
13.1.2 How Much Better Could Have Been Done.....	87
13.1.3 How Much Better Solutions of the Features?.....	87
13.1.4 Which Features Could Be Untouchable?.....	88
13.2 Strengths of the System.....	88
13.3 Weaknesses of the System.....	88
13.4 Future Development.....	88
13.5 Product Evaluation.....	89
<b>Chapter 14: Lessons Learned.....</b>	<b>90</b>
14.1 Introduction.....	90

14.2 What Have I Learned? .....	90
14.3 What Problems Have I Faced?.....	90
14.4 What Solutions Occurred?.....	91
14.5 Conclusion .....	91
Chapter 15: Conclusion .....	92
15.1 Project Summary .....	92
15.2 Goals of the Project.....	92
15.2.1 Fulfilled Goals .....	92
15.2.2 Goals That Are Not Fulfilled.....	92
15.3 The Success of the Project.....	92
15.4 What I Have Done in the Documentation .....	93
15.5 What I Have Learned in the Documentation .....	93
15.6 Value of the Project .....	93
15.7 My Experience.....	93
References .....	94
Appendix:.....	96
Plagiarism Report: .....	98

## Table of Fig

Fig 1.1.....	13
Fig 1.2.....	15
Fig 2.1.....	19
Fig 2.2.....	20
Fig 2.3.....	22
Fig 2.4.....	25
Fig 3.1.....	27
Fig 4.1.....	28
Fig 4.2.....	28
Fig 4.3.....	29
Fig 4.4.....	30
Fig 4.5.....	31
Fig 4.6.....	32
Fig 4.7.....	33
Fig 4.8.....	35
Fig 4.9.....	36
Fig 5.1.....	43
Fig 5.2.....	45
Fig 5.3.....	47
Fig 5.4.....	50
Fig 7.1.....	63
Fig 8.1.....	67
Fig 8.2.....	69
Fig 8.3.....	71
Fig 8.4.....	73
Fig 9.1.....	76
Fig 9.2.....	77
Fig 9.3.....	78
Fig 9.4.....	79
Fig 9.5.....	80
Fig 9.6.....	81-83
Fig 9.7.....	83
Fig 10.1.....	85-86
Fig 11.1.....	92
Fig 11.2.....	92
Fig 11.3.....	93
Fig 11.4.....	93
Fig 11.5.....	94
Fig 11.6.....	94

## Table of table

Table 5.1.....	41-13
Table 5.2.....	43
Table 6.1.....	54
Table 6.2.....	55
Table 6.3.....	55
Table 6.4.....	58
Table 6.5.....	59
Table 8.1.....	73
Table 11.1.....	92
Table 12.1.....	97

## Chapter 1: Introduction

### 1.1 Background

Neither the availability nor the accessibility of a book has lost its fundamental significance in our text-based world that is coming increasingly under the sway of high-tech. Real books still hold a special place in the hearts of readers, even with the advent of ebooks and internet libraries. Many can't really harness this timeless vessel of knowledge due to barriers such as the high price, lack of access to preferred books and no formed system to lend/borrow book.

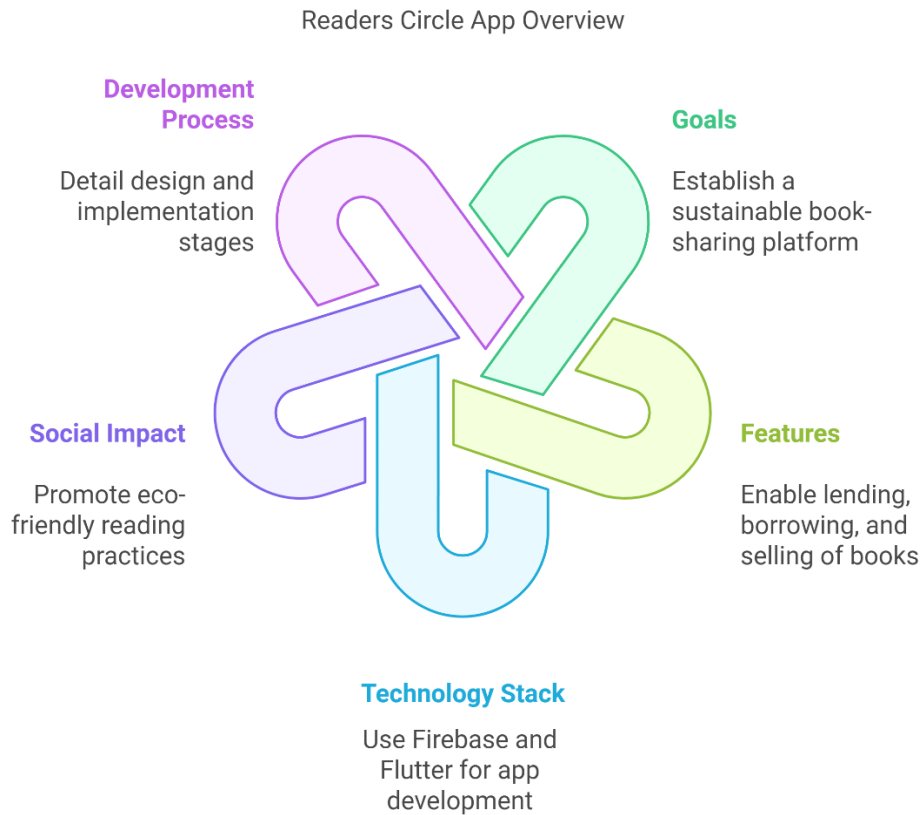


Fig 1.1: A simple overview of Readers Circle Application

This is a specific problem in Bangladesh, with the growing number of students and bookworms! Books libraries and bookshops are generally located in urban places, thus

rural folks face many problems. This is exacerbated by the expensive nature of the new books and the absence of a proper arrangement for sharing or reselling the used books. Cognizant of these challenges, the Readers Circle project was conceived to bridge these divides using modern technologies to create a dynamic, vibrant book-sharing ecosystem that is sustainable and inclusive.

## 1.2 Problem Statement

For the readers of Bangladesh today, there is no reliable, efficient and user-friendly platform where we could lend, borrow or sell books. Important concerns include:

- **High Book Prices:** Many readers, particularly students, are unable to afford new books regularly.
- **Restricted Access:** Diverse reading resources are frequently unavailable in rural and impoverished regions.
- **Environmental Issues:** The creation of new books adds to trash and deforestation, underscoring the need for sustainable substitutes.
- **Disconnection from the Community:** Readers frequently don't have a place to interact, exchange, and have deep conversations about books.
- **Data Quality and Availability:** The efficacy of the platform may be hampered by a lack of information about user preferences and available books.
- **Technological Literacy:** Using a digital platform might be difficult for those with little technological expertise.
- **Internet connectivity:** The usefulness of platforms is impacted by the erratic internet availability that rural areas frequently face.
- **Cost of Implementation:** The substantial financial resources needed to develop and maintain such a platform may restrict its scalability.

## 1.3 Possible Solutions

- **User-Centric Design:** For less tech-savvy users you may want to water down the interface of the platform.

- **Offline Capabilities:** Create features that let consumers access essential functionality even when they don't have reliable internet access.
- **Data optimization:** To enhance data quality and offer precise recommendations, apply AI algorithms.
- **Sustainable Funding Models:** To guarantee affordability and balance operating expenses, implement partnerships or subscription programs.
- **Awareness Campaigns:** Use local outreach initiatives and tutorials to inform users about the platform's benefits.

#### 1.4 Description of the Project

Readers Circle is a concept app that allows you to share, borrow or sell your books. The open-source software aims to advance sustainability and increase universal access to reading materials through combining modern technology and a simple user experience.

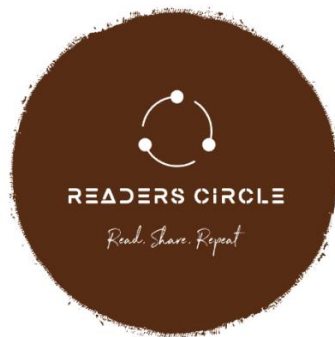


Fig 1.2: The logo of the Readers Circle App

#### 1.5 Key Features

- **Book Listings:** Users can list books with details such as title, author, price, and availability.
- **Secure Transactions:** Integrated payment gateways for safe buying or renting.
- **User Profiles:** Build trust with profiles that display ratings, reviews, and transaction history.

## 1.6 Technology Stack

- Frontend: Flutter for cross-platform mobile development.
- Backend: Firebase for real-time database and authentication.
- Payment Gateway: Integration with bKash, Nagad, and Stripe.
- Hosting: Google Cloud for scalability and reliability.

## 1.7 Impact

The project aims to:

- Increase access to books, especially in under resourced locations.
- Practice environmentally friendly reading by reusing books.
- Create a sense of community among readers with social features.
- Generate a revenue opportunity for users through the sale of books.

## 1.8 Document Context

- ✚ Introduction: A brief history of the project, with its purposes and extent.
- ✚ Preliminary Study: Investigation of Existing platforms and User Needs.
- ✚ Review Of Literature: Review and analysis of relevant literature and technology.
- ✚ Method: A description of tools, constructs and tactics used.
- ✚ Planner: Development schedule, milestones.
- ✚ Viability: Consideration of technical and financial and operating feasibility.
- ✚ Basics: Core architectural decisions and design concepts.
- ✚ Research: Proof of concept and feature testing.
- ✚ Engineering: Making use of the potential offered by the platform.
- ✚ Deployment: The deployment of users and approaches to server introduction.
- ✚ Testing: Assurances and accessibility and operational test findings.
- ✚ Roll-out: Experiences from an implementation approach.
- ✚ Critique and Reflection: Reflection on the accomplishments and areas to improve for the project.
- ✚ Conclusion: Summary of the findings and suggestions for the future.

## **Chapter 2: Initial Study**

### **2.1 Initial Concept**

The concept of "Readers Circle" has evolved to mitigate some of the most prominent issues that book lovers in Bangladesh experience such as high book prices, unavailability of desired books, and lack of organized process of sharing books. Leveraging the tools and technologies of our time, the project hopes to create a digital forum to lend, borrow, and sell books easily so that a community is created and reasonable reading can thrive.

#### **2.1.1 Market Viability**

Market feasibility of 'Readers Circle' is evaluated with respect to some aspects that might attract the Bangladeshi market, where a good deal of possibility of its acceptance and success have been found.

**Rising Need for Accuracy:** Bangladesh has seen a growing literacy rate and a surge in access to education and learning, which has led to a growing interest in the need for accuracy in management of books. Not even traditional bookshops and casual lending circles cut it for today's users who are used to digital. "Readers Circle" is a systematic and technology-based model for loaning and selling books, guaranteeing the accuracy of transactions and inspection records.

**Environmental Sustainability and Compliance:** Urban Bangladesh is witnessing an increasing awareness about the environment. The philosophy behind Readers Circle is two fold, encourage the sharing of reading material, as well as the recycling of books, reusing them and reducing waste. Also, the user can trust users mostly given the fact the platform abides by local digital transaction regulations.

**Market Size** The market size of readers, both professionals and students, who are interested in purchasing or lending books, is huge. Given the high level of smartphone and internet penetration, especially in urban and semi-urban areas, "Readers Circle" has a massive opportunity for growth. For instance, Dhaka, Chattogram, and Sylhet are some

of the largest centers for educational institutions, and both students and teachers or instructors can be primarily beneficiaries of this platform.

**Usability & Accessibility:** The platform is intended to be easy for people with even slight technical literacy or exposure to technology to understand and use. The fact that the app is available on Android and iOS devices would make it even easier for most smartphone owners in Bangladesh to access it. And the free basic usage that everyone loves to brag about.

**Collaboration with Publishers and Tech Companies:** Bangladeshi publishing houses and book distributors usually have a tough time reaching to a larger audience because of a lack of distribution. “Readers Circle” could partner with local publishers and offer digital catalogues and promote their titles. In addition, partnering with tech companies can further enrich the platform with capabilities like AI based suggestions and enhanced analytics for improved user engagement.

**Future Development and Profiles:** This is a working list of features that could be developed into the platform:

1. AI Powered Recommendations: Book recommendations for the users.
2. Geolocation Services: Functionality that helps users find books or lenders in their areas.
3. Subscription: Making subscription on quality models.
4. Community Building: Organizing dialogue and literary forums and events. Such enhancements not only enhances the value and usability of the platform, but also enable avenues for monetization in additional line of business and geographic expansion.

With the above measurements, Readers Circle resolve to be a viable and successful remedy in Bangladeshi market, fill up the gap with provision of a sustainable growth.

## 2.2 Brief Description of the Project

For readers, the app is like a marketplace and party zone, the “Readers Circle.” Users sign up to donate books they have to lend; borrow books they are looking for or sell books they have finished with. In order to keep it convenient for the customers they have included a chat and secure payment feature on the platform.

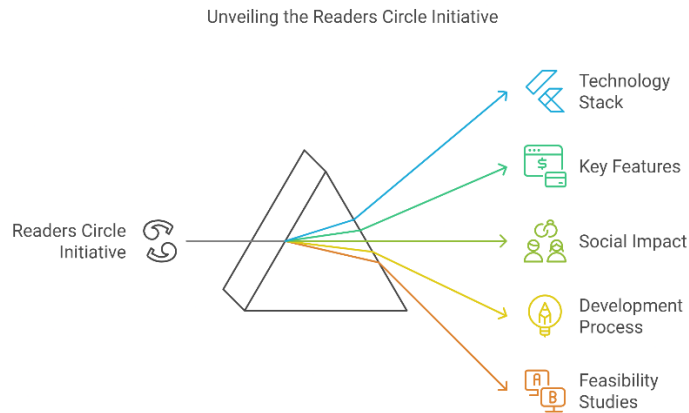


Fig 2.1: Initiative’s process of the app

## 2.3 Aims, Objectives, and Scopes of the Project

### Aims:

- To create an inclusive platform for book lending, borrowing, and selling.
- To foster community engagement through shared reading experiences.
- To encourage sustainable reading habits by promoting book reuse.

### Objectives:

- Design and develop a mobile application with user-friendly features.
- Integrate advanced technologies such as secure payment systems.
- Establish a scalable infrastructure that can expand to various regions.

### Scopes:

- Offer features tailored to the needs of students, book enthusiasts, and casual readers.
- Address socioeconomic challenges by making books accessible at reduced costs.
- Build a sustainable model for long-term growth and community engagement.

## 2.4 Targeted Audience

The platform primarily targets the following groups:

- Students: Affordable access to academic resources.
- Book Enthusiasts: A community to discover and share books.
- Casual Readers: An easy way to access diverse genres without the commitment of purchasing.
- Environmentally Conscious Individuals: Those who value sustainable practices and reducing waste.

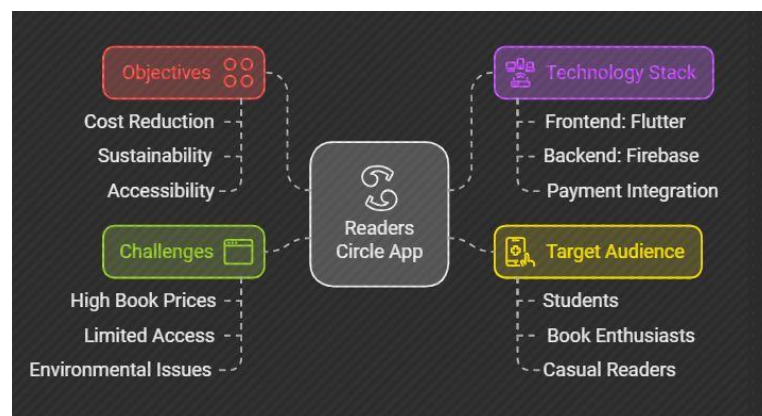


Fig 2.2: Readers Circle app's challenges, objective, targeted audience, technology

## 2.5 Hardware and Software Requirements

### **Hardware Requirements:**

**Development Machines:** High-performance laptops/desktops with minimum 16GB RAM, SSD storage, and modern processors for app development and testing.

**Mobile Devices:** Android and iOS devices for real-world testing.

**Cloud Servers:** Google Cloud instances for hosting and scaling backend services.

### **Software Requirements:**

**Frontend Development:** Flutter framework for creating responsive cross-platform applications.

**Backend Development:** Firebase for real-time database and authentication.

**Database Management:** PostgreSQL for structured and relational data storage.

**Version Management:** Node Version Manager (NVM) for managing Node.js versions during development.

**Frameworks:** Next.js for potential web interfaces.

**API Development:** Prisma for schema-driven API management.

**Development Environments:** Android Studio and Visual Studio Code for coding and debugging.

## **2.6 Human Resources**

### **Development Team:**

**Frontend Developers:** UI/UX design and app development.

**Backend Developers:** Database and API integration.

### **Management Team:**

**Project Manager:** Overseeing timelines and deliverables.

**QA Engineers:** Rigorous testing.

## **2.7 Functional Requirements**

**User Authentication:** Secure login and registration system.

**Book Listings:** Add, edit, and delete book records.

**Search Functionality:** Advanced search filters.

**Transactions:** Rent, borrow, and sell books securely.

## 2.8 Non-Functional Requirements

**Performance:** Ensure the application loads in less than three seconds under normal conditions.

**Scalability:** Support up to 1 million users across multiple regions.

**Security:** Implement robust data encryption and secure APIs to protect user information.

**Usability:** Maintain a simple, intuitive interface to cater to users of all ages.

**Reliability:** Ensure 99.9% uptime with robust monitoring tools.

## 2.9 Exploration and Engineering

**Evolution:** Since it was an evolutionary project, based on the organically growing user feedback it could be refined. Iterative focuses are user registration, book listings, and search, for example.

**Modular Development:** Everything important such as secure payment systems could potentially be placed as a feature to the most flexibility possible and a high level of ease to maintain.

**Continuous integration and deployment:** Deployment to durable storage is done through Google Cloud technologies, while the scalable backend is hosted through Firebase Hosting.

Summary: By drilling down and forward in problem and solution space, we have a structured and focused process for development, thereby decreasing errors and enhancing our ability to deliver high quality features on time.

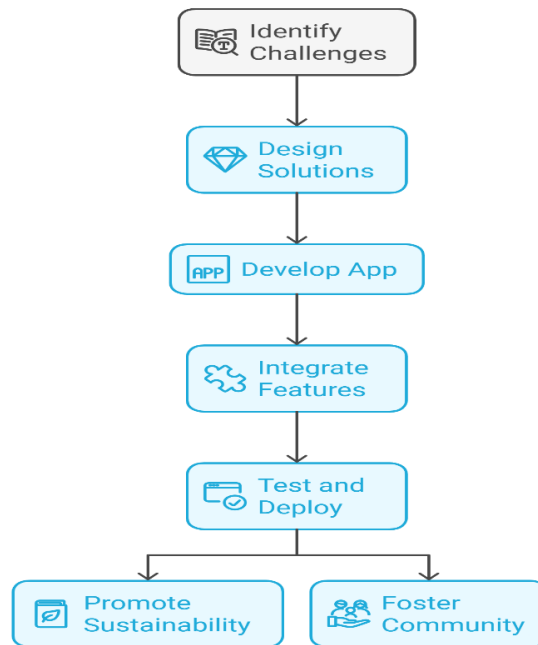


Fig 2.3: App Engineering

## 2.10 Feasibility Studies

### 2.10.1 Operational Feasibility

Adoption by users: The ease of use of the site and functionalities like AI-driven recommendations and location-based search makes the platform popular among the users. Tutorials and support for customers will also help close the technological divide between users.

Scalable solution: "Readers Circle" has well-tested back-end technology using Firebase/GCP, that will give the product the ability to scale to more students in different regions.

Support and Maintenance: Dedicated support and periodic release keep the app bug-free and usable.

Conclusion: Based on the operational feasibility "Readers Circle" will effectively address the client's need, scale efficiently, and provide reliable support services.

### **2.10.2 Technical Feasibility**

Technology Stack: The platform uses a robust stack for its technology: Flutter, Firebase and PostgreSQL to guarantee compatibility and reliability.

Development Environments: Android Studio, Visual Studio Code, Prisma enable smooth development process.

Integration and Testing: By using continuous integration tools, you can ensure that you are extensively testing your application for performance as well as security and usability.

Discussion: The technical basis of "Readers Circle" is solid, using modern technologies which are scalable, secure and efficient for the requirements of the project.

### **2.10.3 Economic Feasibility**

Initial Development Costs: Those are the software licenses, the developers' wages, the infrastructure creation.

Running costs: Server upkeep, ongoing updates, support calls.

Revenue Streams: The revenue will be generated based on subscription-based plans, transaction fees, and partnerships with publishers or educational institutions.

Conclusion: Business is economically feasible, an average payback of the project is possible after 2 years from the date of launch with user's forecasted dynamics and revenue sources.

### **2.10.4 Social Feasibility**

Community Impact: They build communities of users who talk about and share books.

Educational Impact: Easy affordable reading helps literacy and learning from people, especially students.

Conclusions: The social feasibility analysis reveals that the project has the potential to make a positive contribution to the community by promoting a culture of reading and sharing of knowledge.

### **2.10.5 Environmental Feasibility**

Sustainability: Through enabling the reutilisation of books, the website decreases the consumption of new book printing, cutting down on deforestation and materials.

Digital Operations: Because the app is digital, it has a relatively small carbon footprint when compared to traditional book retail.

Conclusion: The project is congruent with environmental sustainability objectives, encouraging the environmentally friendly reading practices and reducing waste.

### **2.11 Overall Feasibility Conclusion**

"Readers Circle" is technically, operationally, financially, socially and environmentally viable as the feasibility study finds. It's an idea that could help communities and ecological health, and address two of the major book issues of today: accessibility and sustainability.

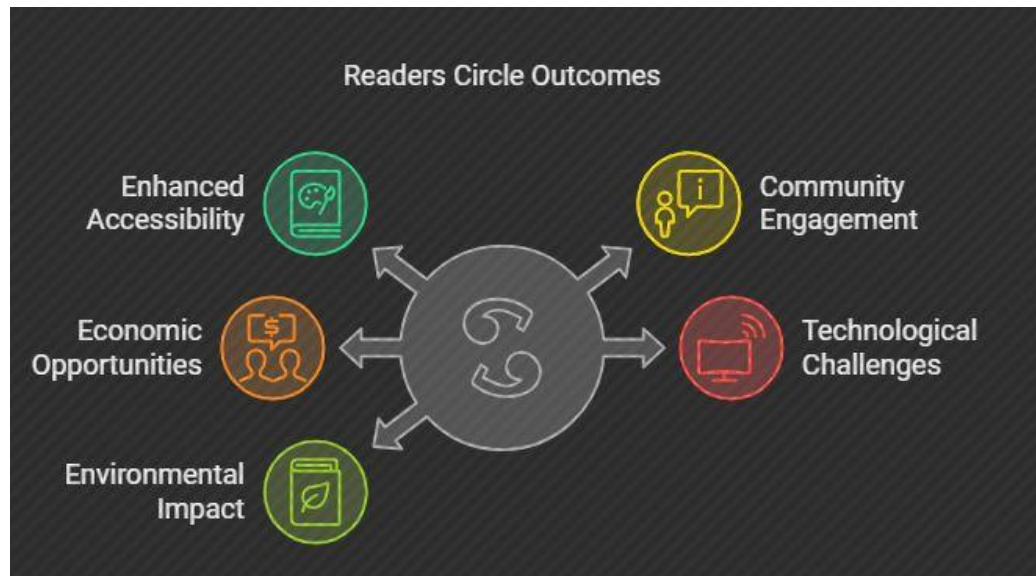


Fig 2.4: a simple outcome pic of the app

### **Chapter 3: Literature Review**

#### **3.1 Description**

The literature review helps to acquire a better understanding of the problem area by reviewing the studies and related work of the “Readers Circle” approach. In this section, problems with the state of the art are described and solutions on how to overcome these are discussed, drawing from previous research on book sharing, community-based platforms and digital markets. It also describes the theoretical perspectives of the project and takes a stand on the technologies and approaches being applied.

#### **3.2 Discussion on Problem Domain Based on Published Articles**

Previous works reveal some of the difficulties involved in making books accessible and shareable:

**Expensive Books:** Research indicates that the cost of books is a barrier for poor individuals and students in developing countries.

**Restricted Library Access:** Research underscores geographical and financial barriers to library access, in particular in rural areas.

Unstructured Book Sharing Platforms Missing: Well there are mom-and-pop book exchanges happening, the organized online place where books could be lent, borrowed and sold from, it is still missing.

Environmental Concern: This has become a primary concern in scholarly conversation as the negative impact of book production on the environment, with reference to energy in production and deforestation is discussed.

### 3.3 Solving the Problem Using Published Papers as a Basis

Research provides various strategies to mitigate these issues:

Digital Sharing Markets for Resource Sharing: To extend accessibility, some research supports digital platforms for community-driven resource sharing, such as books.

Environmental projects: Projects that are promoting recycling and re-use of books are supported by studies where recycling and re-use of books are considered as an environmentally friendly alternative to new production.

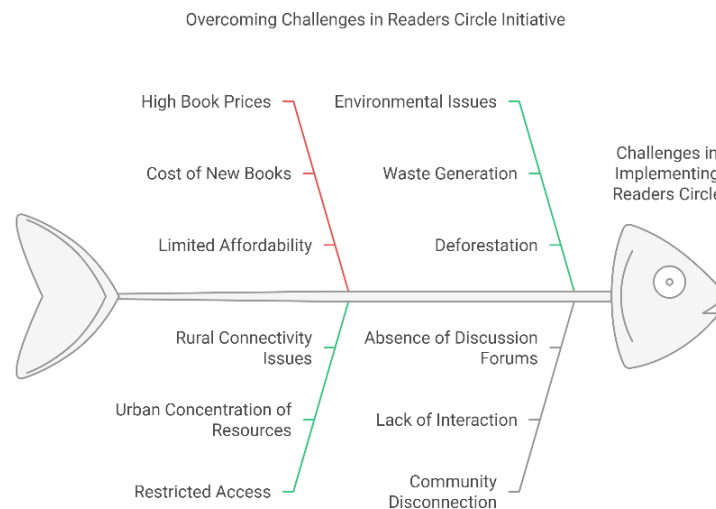


Fig 3.1: Overcoming challenges in Readers Circle Initiative

### 3.4 Recommended Approach

On the basis of the literature investigation, the following methods can be suggested for the "Readers Circle" project:

**Integrated Digital Platform:** Develop a mobile application that combines book lending, borrowing, and selling into a single platform, addressing the lack of organized solutions.

**Eco Promotional Items:** Encourage book recycling and sharing, save the planet together and enjoy green life.

**Community Building:** Build community among users with tools like book clubs, reviews and discussion boards.

## Chapter 4: Methodology

### 4.1 Introduction to Software Development Life Cycle (SDLC)

The Software Development Life Cycle (SDLC) is a step-by-step process that guides the development and maintenance of software applications. The SDLC ensures that projects are done on time, within budget, and in line with requirements by providing a structured approach. You have several SDLC approaches from traditional waterfall to much more fluid iterative modalities, all designed to accommodate different project types and team styles.



Fig 4.1: SDLC Life Cycle Model

## 4.2 Overview of SDLC Methodologies

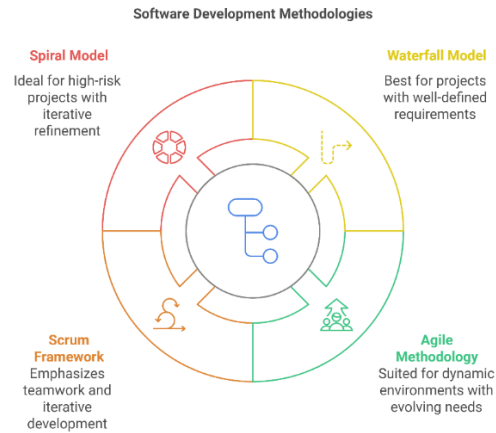


Fig 4.2: SDLC Methodologies

### Waterfall Model

Waterfall Model was one of the first SDLC Models to be used widely. We have to finish each one of the following before we can go on to the next: Requirements collecting, design, implementation, testing, and maintenance. The technique is preferable for projects that have well-defined and consistent requirements.

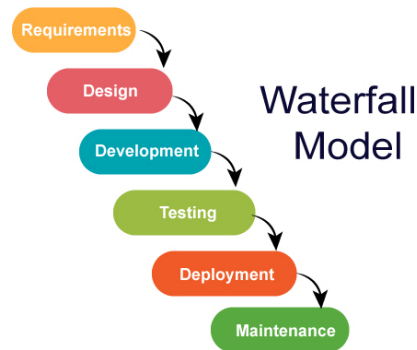


Fig 4.3: Waterfall Model

**Example:** Developing a payroll system for a corporation, where all requirements are well-known beforehand.

## Pros:

- ✓ Deliverables and Milestones Every stage has unique deliverables and milestones.
- ✓ Long-term support viability is guaranteed with strong documentation.
- ✓ Great for small jobs with fixed requirements.
- ✓ Effective control for regulation or compliance directed implementations.

## Cons:

- Not flexible to changes once the implementation begins.
- Inappropriate for complex or dynamic functions.
- Problems were found late in the test cycle.
- Without a real understanding of the original need, the project is likely to fail.
- Minimal pre-testing user interaction.

## Agile Methodology

Agile stresses adaptability, collaboration, and delivering in small increments. It breaks down the project into 'sprints' (manageable, iterative chunks) so you can receive frequent feedback and make changes. Agile is most useful for environments in which dynamic changes are anticipated throughout the process.

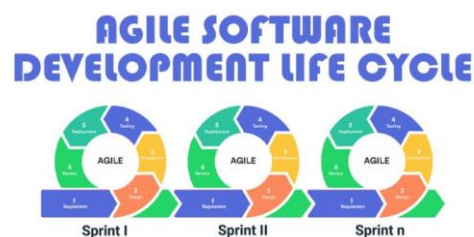


Fig 4.4: Agile Methodology

**Example:** Developing a mobile application where user preferences might change based on market trends.

## Pros:

- ✓ Extremely dynamic – adaptation of changing requirements.
- ✓ Drive close coordination among various stakeholders and teams.
- ✓ Customer requirements is validated periodically.
- ✓ Client satisfaction is enhanced when working software is delivered early.
- ✓ Hours of Major failure: Being far less when in test and integrated phases.

**Cons:**

- Seeks team members who are good communicators and bring depth of subject knowledge.
- Future service can be complicated if documentation is not explicit.
- High dependence on users contributions.
- Not as efficient with large, dispersed groups.
- Could lead to scope creep if changes are not managed correctly.

**Scrum Framework**

**Description:** Projects are broken up into time-boxed sprints, usually lasting two to four weeks, using the Scrum Agile methodology. It incorporates responsibilities like Scrum Master, Product Owner, and Development Team and places a strong focus on incremental development.

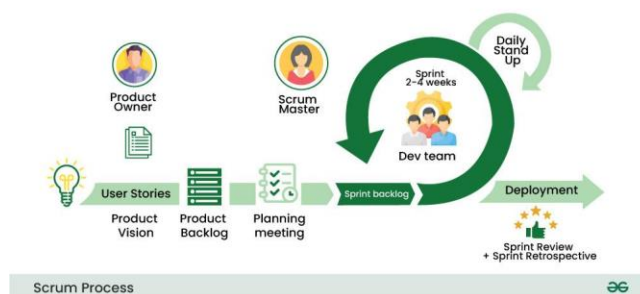


Fig 4.5: Scrum Framework

**Example:** Building an e-commerce platform with new features added incrementally, such as payment integration or user reviews.

**Pros:**

- ✓ Encourages collaboration and continual improvement.
- ✓ Uses sprint reviews to clearly show the status of the project.
- ✓ Increases flexibility in response to shifting priorities.
- ✓ Makes it easier to identify and address problems early.
- ✓ Delivering working components on a regular basis boosts user confidence.

**Cons:**

- Mostly depends on the discipline and self-organization of the team.
- Needs seasoned team members to be implemented successfully.
- Not suitable for projects with a set scope or budget.
- May cause reaching sprint targets to take precedence over quality.
- Challenging to scale for intricate, large-scale applications.

**Spiral Model**

**Description:** Iterative development and methodical risk assessment are combined in the Spiral Model. It is perfect for high-risk projects since it entails repeated cycles (or spirals) of planning, prototype, assessment, and refining.

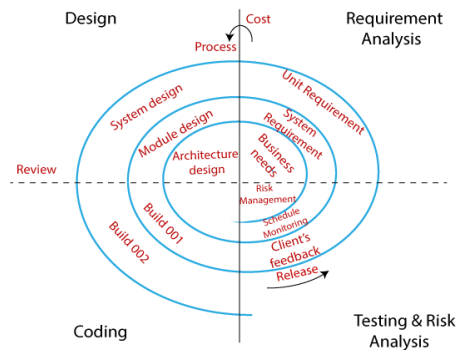


Fig 4.6: Spiral Model

**Example:** Developing an aerospace control system, where risk mitigation is critical at every stage.

**Pros:**

- ✓ Focusses on identifying and reducing risks.
- ✓ Permits ongoing needs improvement.
- ✓ Blends the finest features of Agile and Waterfall approaches.
- ✓ Ideal for tasks that are sophisticated and extensive in scope.

- ✓ Adaptable to changing needs.

**Cons:**

- Costly and time-consuming because of the numerous iterations.
- Need a great deal of experience in risk assessment.
- Challenging to oversee minor projects.
- Planning and execution complexity.
- Heavy dependence on records.

**The Dynamic Systems Development Method (DSDM)**

The Dynamic Systems Development Method (DSDM) is a gradual process of software development. It is heavily based on regular product delivery, user involvement, and the flexibility as requirements adjust. For fast paced budget and time bounds projects, such as "Readers Circle", that requires for a quickly working features while being opened for changes, DSDM does the job.

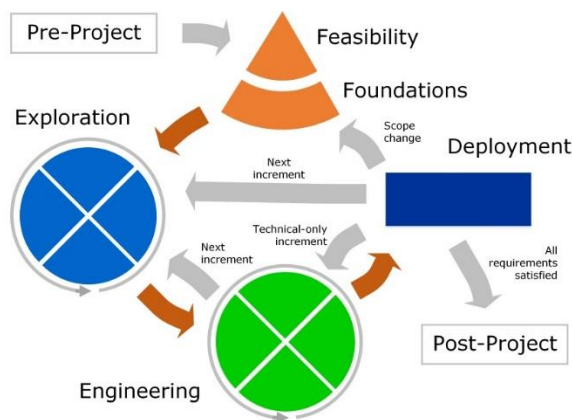


Fig 4.7: DSDM Lifecycle

**Key Principles of DSDM:**

- ❖ Focus on Business Needs: The project must deliver real business value.

- ❖ Cooperation with a Frequent Delivery: Deliver the system in small piece sizes so that it does not detach from requirements.
- ❖ Key Users Involved: Involve users in every step of the project to ensure fit with requirements.
- ❖ Iterative Development The software must be developed iteratively to receive feedback and to incorporate changes.
- ❖ Iteration Testing: Iteration testing is important process in development cycle.
- ❖ Mixing and Communication: Promote collaboration and communication between all stakeholders to ensure the success of the project.

### **Advantages of DSDM:**

- ✓ Flexibility: 'Readers Circle' is adaptable to meeting new needs and adding user feedback.
- ✓ User-Centered: Engages users throughout to ensure the end product is what they need.
- ✓ Risk Reduction: By identifying and resolving issues early, releasing often reduces the probability of catastrophic project failure.
- ✓ Short Delivery Time: Adoption of short cycle development allows delivery of useful parts in a short period.
- ✓ Quality Assurance: Continuous testing ensures that the system and complies with strict quality requirements.

### **Disadvantages of DSDM:**

- Chance of anchoring: Needs careful planning.
- Complexity in Huge Projects: In huge projects; managing iterations & user conversation is tough.
- Time Required: The dev team and the stakeholders must invest a lot of time on meetings and reviews.

Example of DSDM Application: DSDM was used in the Readers Circle project to ensure that salient features like transaction management, book listings, user identification. Owing to the iterative approach the development team were able to gather feedback from users, refine the system and deliver a working prototype earlier than anticipated. For example, initial user input led to the change in book search capabilities to make them more user-friendly.

### 4.3 Selected SDLC Methodology for "Readers Circle"

The Agile Methodology with a hint of the Scrum Framework was chosen for its adaptability, customer focus, and evolutionary nature for the "Readers Circle" project. The benefits of these solutions are quick supply of useful parts, considering evolving user needs.

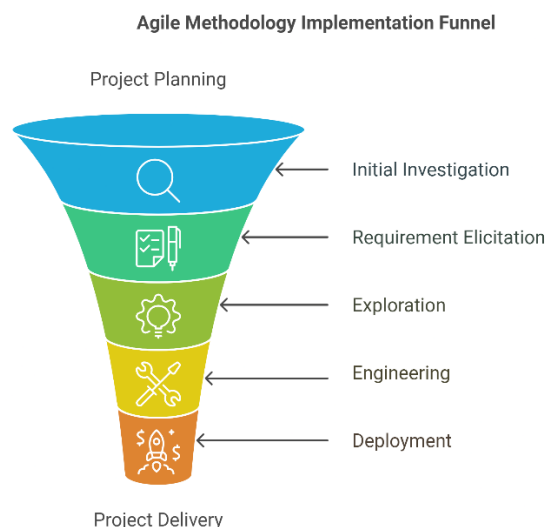


Fig 4.8: Agile Implement Model

Implementation: The project was broken up into many sprints, each of which concentrated on a different feature, such book listings, and user authentication. Alignment with project objectives was guaranteed by frequent sprint reviews and stakeholder feedback meetings.

## 4.4 Methodology Phases

### **Initial Investigation:**

Carried out market research to find weaknesses in the current book-sharing systems. Used focus groups and surveys to analyze user requirements.

### **Requirement Elicitation:**

Used questionnaires and stakeholder interviews to collect both functional and non-functional needs.

JIRA and other technologies were used to document needs in order to improve tracking.

### **Requirement Analysis and Prioritization:**

Needs were categorized according to technological difficulty, user impact, and viability. Features like geolocation, safe transactions were given priority.

### **Exploration:**

Prototypes were created to verify feature and design concepts. Firebase, and Flutter were among the technologies that were assessed to make sure they met project requirements.

### **Engineering:**

Complied with modular design guidelines for maintainability and scalability. Used automated testing and continuous integration to guarantee quality.

### **Deployment:**

Utilized Google Cloud for hosting and scalability.

Released beta versions for user testing and feedback.

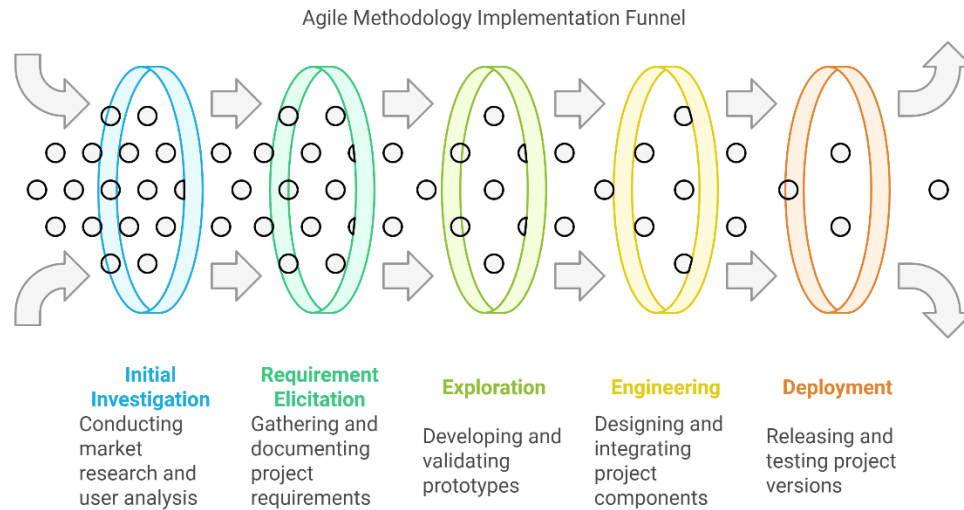


Fig 4.9: Agile Implement Model

#### 4.5 Implementation Plans

**Frontend Development:** Created intuitive and responsive interfaces using Flutter.

**Backend Development:** Leveraged Firebase for real-time data syncing and PostgreSQL for robust database management.

**Testing:** Conducted unit, integration, and system testing.

#### 4.6 Conclusion

The Agile and Scrum methodologies fit well with the Readers Circle project. By using an iterative approach, the team could rapidly develop functional pieces, easily incorporate feedback from users, and respond to changing requirements. By making the platform meet user and technology needs, the well-constructed but flexible design paved the way for further growth and development.

## Chapter 5: Planning

### 5.1 Introduction

Planning is an inherent part of any project life cycle ensuring that the objectives are well defined, resources are utilized in a best possible manner and the risks are minimized. To ensure the successful implementation that such a large project entailed, the development planning of the project "Readers Circle" engaged an extensive road map planning for the relevant tasks, deadlines and results. WBS and Project Plan and their significance for this project is explained in this chapter.

### 5.2 About the Project Plan

Planning is an inherent part of any project life cycle ensuring that the objectives are well defined, resources are utilized in a best possible manner and the risks are minimized. To ensure the successful implementation that such a large project entailed, the development planning of the project "Readers Circle" engaged an extensive road map planning for the relevant tasks, deadlines and results. WBS and Project Plan and their significance for this project is explained in this chapter.

#### **Project Objectives:**

Feature the latest functions such as secure transactions, AI-driven recommendations and geolocation. Ensure that it is a sustainable and scalable solution to cater for an increasing population of users.

#### **Timeline:**

The project is divided into five major phases: Initial Investigation, Requirement Analysis, Development, Testing, and Deployment.

Estimated duration: 9 months.

#### **Resource Allocation:**

**Human Resources:** Development team, QA engineers, UI/UX designers, and project managers.

**Technological Resources:** Flutter, Firebase, PostgreSQL, and Google Cloud.

**Risk Management:**

Regular assessments and contingency planning help to monitor potential risks including budgetary limits, changing user needs, and development delays.

**Milestones:**

- ✓ The design of a prototype is completed.
- ✓ Integration of important functions like book listing, user identification.
- ✓ Implementing feedback and beta testing.

### 5.3 Work Breakdown Structure (WBS)

WBS is a hierarchical deconstruction of the project scope into a bite-size task and subtask. It allows you to plan and organize your work efficiently ensuring all outputs are followed through and resources allocated adequately.

#### Phase 1: Planning and Initial Investigation

- Define the tasks and limits of the project.
- Perform market research.
- Determine the stakeholders and compile the preliminary needs.

#### Phase 2: Requirement Analysis

- Both functional and non-functional needs should be documented.
- Sort features according to their technological viability and user needs.

#### Phase 3: Development

- Frontend Development: Design and implement user interfaces using Flutter.

- Backend Development: Build APIs, integrate databases, and implement authentication.

#### Phase 4: Testing

- Conduct unit testing, integration testing, and system testing.
- Perform usability testing with beta users.

#### Phase 5: Deployment and Maintenance

- Deploy the application on Google Cloud.
- Monitor performance and address user feedback.
- Provide regular updates and maintenance.

Traceability of all the works allows the project team to identify areas of possible bottleneck and dependency. It is a systematic approach to complex issues and it ensures feature delivery on time and project success overall.

### 5.4 What is Time Duration and Time Boxing

Time Duration – length of time for which a piece of work or work phase is to last. It is the case of predicting and setting a simple and achievable schedule to get this done on time.

Time boxing is a kind of scheduling and time management in which a maximum time is fixed for each task or “job”. It helps keep you focused, prevent scope creep, and increase productivity with clear boundary.

### 5.5 Time Box

Task Name	Start Time	Duration	End Time
1. Initial Study	01-06-2024	1 week	07-06-2024
1.1 Identify the Project All Requirements	01-06-2024	1 week	14-06-2024

1.2 Justify the Project	15-06-2024	1 week	21-06-2024
1.3 Find Out the Solution	15-06-2024	1 week	07-06-2024
1.4 The Aims and Objectives	22-06-2024	1 week	29-06-2024
2. Requirement Gathering and Analysis	30-06-2024	6 weeks	10-08-2024
2.1 Feasibility Study	30-06-2024	1 week	05-07-2024
2.2 Use Cases	06-07-2024	1 week	12-07-2024
2.3 Functional Requirements	13-07-2024	1 week	19-07-2024
2.3 Functional Requirements	20-07-2024	1 week	26-07-2024
2.4 Non-Functional Requirements	27-07-2024	1 week	04-08-2024
2.5 System Architecture	27-07-2024	1 week	04-08-2024
2.6 ERD for Database Management	05-08-2024	1 week	10-08-2024
3. Design	11-08-2024	6 weeks	21-09-2024
3.1 UI Design	11-08-2024	1 week	17-08-2024
3.2 Database Design	18-08-2024	1 week	24-08-2024
3.3 Class Diagram	25-08-2024	1 week	31-08-2024
3.4 Behavioral Model	01-09-2025	1 week	07-09-2025
3.5 Collaboration Diagram	08-09-2025	1 week	14-09-2025

4. Implementation	22-09-2025	8 weeks	16-09-2025
4.1 Programming Language	22-09-2025	4 weeks	19-10-2024
4.2 Project Implementation	20-10-2024	4 weeks	16-11-2024
5. Testing	17-11-2024	4 weeks	14-12-2024
6. Documentation	15-12-2024	4 weeks	10-01-2025

Table 5.1: Time Box

**5.6 Resource Allocation Table**

Resource Type	Assigned Task	Quantity	Responsibility
Frontend Developers	UI/UX Design	2	Designing and implementing UI/UX
Backend Developers	API and Database	2	Building backend systems
QA Engineers	Testing	2	Conducting unit and system tests
Project Manager	Overall Management	1	Overseeing project timelines
Customer Support Team	User Feedback and Support	2	Assisting users post-deployment

Table 5.2: Resource Allocation

**5.7 What is a Gantt Chart and Why Use It for This Project**

Tasks are displayed along a timeline in a Gantt chart, which is a visual depiction of a project timetable. It facilitates efficient project management by emphasising interdependence, start and finish dates, and progress.

Purpose in This Project:

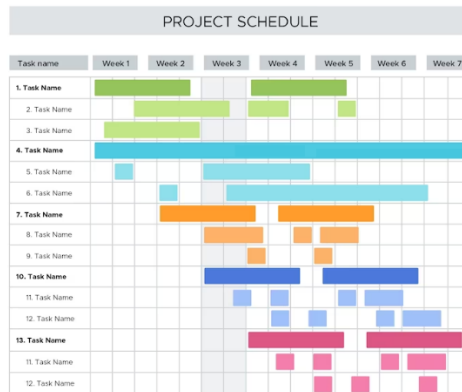


Fig 5.1: Gantt Chart

Enhanced Planning: Scheduling work rationally and effectively is ensured by visualising the timetable.

Progress Monitoring: allows for a clear view of both finished and unfinished work.

Risk Mitigation: Bottlenecks can be avoided by identifying delays or dependencies early.

Improved Coordination: Encourages team members to communicate and work together.

## 5.8 Require Testing

Ensuring that the software can meet both the design as well as non-design requirements, testing is a vital phase in software development lifecycle. A comprehensive testing plan for "Readers Circle" was designed to ensure the usability, security, and functionality of the system.

### **Unit Testing:**

**Purpose:** To confirm that each application component functions as expected.

**Example:** Verifying that users may authenticate using legitimate credentials by testing the login feature.

### **Integration Testing:**

**Purpose:** To verify that connected components work together by testing their interactions.

**Example:** confirming that books are accurately retrieved from the database via the search function.

### **Acceptance Testing:**

**Purpose:** To confirm that the system's operation meets user requirements.

**Example:** Users testing the AI recommendation feature to confirm it provides relevant suggestions.

## **5.9 Non-Functional Testing**

### **Security Testing:**

**Purpose:** To find weaknesses and guarantee data security.

**Example:** Verifying encryption systems to protect customer payment data.

### **Usability Testing:**

**Purpose:** To evaluate user happiness and the application's usability.

**Example:** Evaluating the intuitiveness of the book search and listing processes.

### **Reliability Testing:**

**Purpose:** to guarantee that the application continues to work in a variety of circumstances.

**Example:** To verify the system's stability, simulate heavy user traffic.

## 5.10 Conclusion

Both functional and non-functional testing were applied in the Readers Circle test phase. The new project ensured that a steady, reliable, and easy-to-use system was fully ready for implementation while strictly checking every part and process.

## 5.11 Risk Management

The systematic process of identifying, analyzing, and minimizing potential risks that can affect the success of a project is called risk management. Risk Assessment "Readers Circle" is benefitted greatly by risk management, it ensures the platform is delivered on time, to budget and as per user demands. This means not only being prepared to cope with problems such as resource constraints, altering needs, and technology issues as they arise, but also taking steps to ensure that they don't derail the project before it is complete.



Fig 5.2: Risk Management Process

### 5.11 Risk Identification

Identifying risks entails seeing possible dangers that might cause the project to fail. For "Readers Circle," the following are major risks:

#### **Technological Risks:**

- Problems integrating Firebase with Flutter.
- Issues with scalability as the user base expands.

### **Resource Risks:**

- Restricted access to qualified developers.
- Feature development is impacted by financial limitations.

### **Operational Risks:**

- Failure to fulfill project deadlines.
- Inadequate methods for gaining users.

### **Security Risks:**

- Information about users is compromised by data breaches.
- Payment gateways are not sufficiently encrypted.

## **5.12 Risk Precaution**

The following safety measures are put in place to mitigate the hazards that have been identified:

### **Technological Precautions:**

- ✓ To guarantee smooth integration, carry out comprehensive testing at each level of development.
- ✓ To handle expansion, make use of scalable cloud infrastructure, such as Google Cloud.

### **Resource Precautions:**

- ✓ Employ more qualified staff as required.
- ✓ Keep an eye on how much money is being spent and give priority to features.

### **Operational Precautions:**

- ✓ Plan beforehand for any delays that may occur.
- ✓ Put training initiatives into action to increase user adoption.

### **Security Precautions:**

- ✓ Perform routine security assessments.
- ✓ Set up user accounts using two-factor authentication.

## **5.14 Change Management**

Preparing, assisting, and supporting teams and stakeholders in adapting to project changes are all part of change management. Effective change management guarantees that "Readers Circle" develops without compromising its objectives, especially considering the dynamic nature of software development.

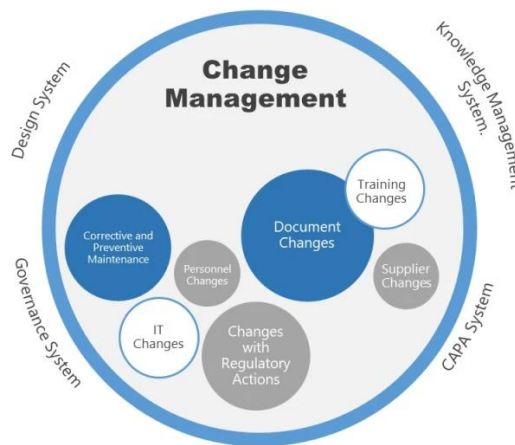


Fig 5.3: Change Management Process

### **Key Steps in Change Management:**

- **Change Request Submission:** Changes are suggested by team members or stakeholders via a defined procedure.

- **Impact Analysis:** Examine the impact of the modification on project scope, budget, and schedule.
- **Approval or Rejection:** The modification is assessed and decided by the project manager.
- **Implementation:** Changes that have been approved are implemented into the project plan.

### 5.15 Factors That Might Cause Change

Changes to the project may be required for a number of reasons, including:

**User input:** recommendations made during beta testing for additional features or enhancements.

**Market trends:** Include adjusting to new customer demands or technological advancements.

**Regulatory Requirements:** Adherence to recent security or data protection legislation.

**Technical Issues:** Unexpected difficulties that arise during integration or development.

**Resource Constraints:** Modifying the scope because of financial or manpower constraints.

### 5.16 DSDM Atern and Welcoming Change

Dynamic Systems Development Method, or DSDM, The Agile framework Atern places a strong focus on adaptability and teamwork. As a fundamental tenet, it embraces change and makes sure that changing needs can be accommodated without causing the project to go off course.

#### **Key Principles of DSDM Atern:**

**Pay Attention to Business Needs:** Give top priority to tasks that will benefit users the most.

**Iterative Development:** Create iterations to take input into account.

**Cooperation:** Throughout the process, include team members and stakeholders.

**Accepting Change:** Rather than viewing changes as challenges, view them as chances for growth.

### Application to "Readers Circle":

The DSDM Atern approach is applied by:

- Iteratively developing features like book listings.
- Regularly gathering user feedback during beta testing.
- Welcoming changes suggested by stakeholders to enhance user satisfaction.

## 5.17 Conclusion

It's because of the sound risk and change management strategies that "readers circle" is supported for fuel the success. The project ensures that it is robust in an increasingly changing development environment by identifying risks early, making contingencies, and adopting a flexible process such as the DSDM Atern. This application of change as an instrument of continual improvement, rather than just as a means of trouble-shooting.

## 5.18 Quality Management

Quality management is the act of planning and executing procedures to ensure that the end product satisfies or is better than customer expectations. For the application "Readers Circle", quality assurance is both functional and none functional it ensures that the platform is secure, reliable and user-friendly.



Fig 5.4: Quality Management Process

### 5.19 Rules Applied to Maintain Quality

**Unambiguous Need Documentation:** Make sure that every need, whether functional and non-functional, is agreed upon and recorded.

**Code Reviews:** To preserve code quality and identify problems early, conduct regular peer reviews.

**Automated Testing:** Use automated testing to validate data more quickly and reliably.

**User Feedback Loops:** Use input from beta testing to improve functionality.

**Frequent Audits:** Perform recurring audits to ensure compliance, security, and performance.

### 5.20 DSDM Atern Standard Quality Measures

Quality, according to DSDM Atern, is the duty of all project participants. Important quality indicators consist of:

**Fit for Purpose:** The product must satisfy user and business requirements.

**Defined Acceptance Criteria:** Each delivery must pass predetermined acceptance tests, according to the established acceptance criteria.

**Iterative Quality Checks:** Throughout the development process, quality is regularly evaluated.

**Cooperation:** Include interested parties in the formulation and verification of quality standards.

**Focus on Prioritisation:** Prioritise your work by starting with the most important, high-priority activities.

## 5.21 Quality Plan

The quality plan for "Readers Circle" includes:

**Quality Assurance Activities:**

Implement unit, integration, and system testing.

Perform usability and security testing.

**Testing Frameworks:**

Use tools like Selenium for UI testing and OWASP ZAP for security testing.

**Feedback and Review:**

Gather user feedback during beta testing phases.

Schedule regular review meetings with stakeholders.

**Documentation:**

Maintain detailed documentation for all testing and quality assurance processes.

## 5.22 Conclusion

In order to guarantee success, the "Readers Circle" project has used many measures, ranging from risk management to quality planning. Delivering a dependable and user-centric platform requires proactive risk reduction, flexibility in response to changes, and strict quality controls. Utilising frameworks such as DSDM Atern highlights the project's dedication to stakeholder satisfaction and ongoing development.

## Chapter 6: Feasibility Study

### 6.1 Introduction

Feasibility study, investigation into the real details for a project including budget, time, market, risk, and many types. In case of "readers Circle", you make sure that what you intend to achieve from the project is in sync with the available resources, the time period you have from your schedule and the market demand. Some fundamental viability parameters and their applicability to the project are presented in this chapter.

### 6.2 All Possible Types of Feasibility

#### 6.2.1 Economic Feasibility

By weighing costs against potential benefits, the cost-benefit analysis evaluates the economic viability of the project. This includes the cost for hosting of the site as well as software, hardware, and various other recurring costs for site operation for the "Readers Circle."

Table 1: Overall Cost Estimation

Serial No.	Sector	Cost (BDT)
1	Software	100,000
2	Hardware	250,000
3	Hosting	50,000
4	Others (Training)	30,000
<b>Total</b>	<b>=</b>	<b>430,000</b>

Table 6.1: Overall Cost Estimation

Table 2: Software Cost

Serial No.	Name	License Duration	Price (BDT/Year)	Total Price (BDT)
1	Microsoft Office	1 Year	12,000	12,000
2	Browser (Open Source)	N/A	0	0
3	Operating System	Lifetime	15,000	15,000
4	PostgreSQL	Open Source	0	0
5	VS Code (React)	Open Source	0	0
6	Prisma	Open Source	0	0
<b>Total</b>				<b>27,000</b>

Table 6.2: Software Cost

Table 3: Hardware Cost

Serial No.	Hardware Name	Quantity	Price (BDT)
1	HP Laptop	3	80,000
2	Router	1	10,000
3	Android Phone	1	20,000
<b>Total</b>			<b>270,000</b>

Table 6.3: Hardware Cost

## 6.2.2 Technical Feasibility

Technical viability looks at the resources, technology, and tools needed to create and run the platform. "Readers Circle" makes use of contemporary frameworks such as TensorFlow Lite for AI integration, Firebase for backend services, and Flutter for frontend development. These technologies are extensively used, dependable, and scalable.

### Software Used:

**Flutter:** For cross-platform mobile development, enabling the application to run on both Android and iOS devices.

**Firebase:** Provides backend support, including authentication and real-time database management.

**PostgreSQL:** Serves as the primary relational database for structured data management.

**Visual Studio Code:** A robust IDE for efficient coding and debugging.

**Prisma:** Simplifies API development and database schema management.

### Hardware Used:

**HP Laptops:** High-performance laptops for development and testing.

**Router:** Ensures reliable internet connectivity during development and deployment.

**Android Phone:** Used for real-world application testing on mobile devices.

These tools and resources are reliable, widely adopted, and well-documented, ensuring the project's technical foundation is robust and sustainable.

## 6.2.3 Operational Feasibility

Operational viability evaluates how well the project fits organisational capabilities and user demands. With its straightforward UI, geolocation-based search, and AI-powered

suggestions, "Readers Circle" guarantees excellent usage. To guarantee ongoing progress, regular user input will be taken into account.

### **User-Friendly Interface:**

The app's simple and intuitive design ensures users of all ages can navigate it easily.

**Example:** Clear menus for book listing, borrowing, and selling.

### **Geolocation-Based Features:**

Enables users to discover books available nearby.

**Example:** A student can find textbooks from a seller within their vicinity, reducing delivery costs and time.

### **Secure Transactions:**

Integrated payment gateways ensure that users can securely buy or rent books.

**Example:** Payments processed through bKash or Stripe with encryption protocols.

### **Community Engagement:**

Features like user reviews and discussion forums foster interaction among book enthusiasts.

**Example:** A user shares a review about a book they borrowed, helping others decide.

Regular updates and user feedback mechanisms ensure continuous improvement and operational alignment with user needs.

## 6.2.4 Social Feasibility

The project's social impact is the main emphasis of social feasibility. "Readers Circle" encourages community involvement, literacy development, and environmental sustainability by encouraging book exchange and reuse.

## 6.2.5 Environmental Feasibility

The ecological impact of the project is assessed via environmental feasibility. By lowering the need for fresh book creation, the platform minimises paper waste and deforestation while promoting environmentally friendly reading habits.

## 6.2.6 Cost-Benefit Analysis

Cost-benefit analysis evaluates the financial returns of the project compared to its costs. Below are two tables summarising the expenditures and revenues for "Readers Circle."

Table 1: Sector of Expenditure (BDT)

Serial No.	Sector	Year 1	Year 2	Year 3	Year 4	Year 5
1	Software Cost	100,000	10,000	10,000	10,000	10,000
2	Hardware Cost	250,000	0	0	0	0
3	Hosting	50,000	50,000	50,000	50,000	50,000
4	Office Cost	20,000	20,000	20,000	20,000	20,000
5	Staff Cost	100,000	100,000	100,000	100,000	100,000
6	Development Cost	100,000	0	0	0	0
7	Other Costs	30,000	10,000	10,000	10,000	10,000
<b>Total</b>		<b>650,000</b>	<b>190,000</b>	<b>190,000</b>	<b>190,000</b>	<b>190,000</b>

Table 6.4: Sector of Expenditure

Table 2: Cost and Revenue Analysis (BDT)

Serial No.	Sector	Year 1	Year 2	Year 3	Year 4	Year 5	Total
1	Total Earnings	300,000	500,000	700,000	900,000	1,000,000	3,400,000
2	Total Expenditure	650,000	190,000	190,000	190,000	190,000	1,410,000
3	Total Revenue	-350,000	310,000	510,000	710,000	810,000	1,990,000

Table 6.5: Cost and Revenue Analysis (BDT)

### 6.2.7 Conclusion

The feasibility report of the "Readers Circle" demonstrates the project's feasibility, economically, technologically, in operations, socially and ecologically. The platform is able to meet targets if we plan well and allocate resources properly. The "Readers Circle," will recoup its initial investment in two years. It will then be in the black from that day forward. The long-run profitability and economic viability of the project is justified by this research.

## Chapter 7: Foundation

### 7.1 Introduction

Readers Circle The feasibility study demonstrates that the project is feasible from the economic, technological, operational, social, environmental and institutional perspectives. The platform can actually achieve its goals in a sustainable and victorious way! According to its cost-benefit analysis, "Readers Circle" will recoup its initial investment after two years and from there continue to see increasing revenue. The economics and profitability of the project at long run are validated in this study.

### 7.2 Overall Requirement List

The "Readers Circle" project requires a combination of functional and non-functional capabilities to deliver a seamless user experience. These include:

#### **Functional Requirements:**

- Book listing and borrowing features.
- Secure payment processing.
- User account management.

#### **Non-Functional Requirements:**

- Scalability to support increasing users.
- High reliability and availability.
- Data security and encryption.
- Cross-platform compatibility.
- 7.3 Functional Requirements

#### **Book Listing and Borrowing:**

Users can list books they own for lending or selling.

Borrowers can search and request books directly from lenders.

**Example:** A student lists a textbook, and another student nearby borrows it for a semester.

### **Secure Payment Processing:**

Integrated payment gateways like bKash and Stripe for secure transactions.

**Example:** A user pays a nominal fee to rent a book for two weeks.

### **User Account Management:**

Features include registration, login, and profile management.

**Example:** Users can update their address to optimise geolocation results.

## **7.4 Non-Functional Requirements**

### **Validation/Verification:**

Ensures data accuracy and prevents invalid inputs.

**Example:** Validating email addresses during registration.

### **Responsive Design:**

Adapts seamlessly across devices and screen sizes.

**Example:** The app interface adjusts for both mobile and tablet users.

### **Security:**

Protects sensitive user information using encryption and secure protocols.

**Example:** Secure payment gateway integration.

### **Reliability:**

Ensures consistent performance under varying conditions.

**Example:** Handling high user traffic without downtime.

**Flexibility:**

Allows easy addition of new features or updates.

**Example:** Integrating a book discussion forum in a future release.

**Testability:**

Facilitates effective testing during development and deployment phases.

**Example:** Automated unit testing for core functionalities.

## 7.5 Problem Area Identification

The project addresses several critical issues:

- Lack of affordable access to books.
- Limited platforms for book sharing and selling.
- Security concerns in digital book transactions.
- Absence of AI-driven personalized recommendations.

## 7.6 Observation

The findings draw attention to user complaints about things like the inaccessibility of particular book genres, a lack of confidence non current platforms, and the high price of buying new books. This information informs the development approach for "Readers Circle."

## 7.7 Possible Solution

- Develop a user-friendly mobile app for seamless book lending and selling.
- Incorporate secure payment gateways to build user trust.
- Use AI algorithms for personalized recommendations.

- Enable geolocation features for local book exchanges.

## 7.8 BPMN (Business Process Modeling and Notation)

A graphical depiction of business processes, BPMN aids in workflow design and analysis. BPMN maps the whole user path for "Readers Circle," guaranteeing that the system's interactions and activities are clear.

### Detailed Steps:

#### **User Registration and Profile Setup:**

Users register by providing necessary details such as email and phone number.

The system verifies credentials and sets up user profiles.

#### **Book Listing and Search:**

Registered users list books by providing details like title, author, and availability.

Borrowers use search filters and geolocation to find desired books.

#### **Request and Approval:**

Borrowers request books through the system.

Lenders receive notifications and approve or reject requests.

#### **Secure Payment and Delivery:**

Payment transactions are processed through integrated gateways.

Delivery details are shared between the lender and borrower.

#### **Feedback and Review Submission:**

Borrowers leave feedback on book condition and lender reliability.

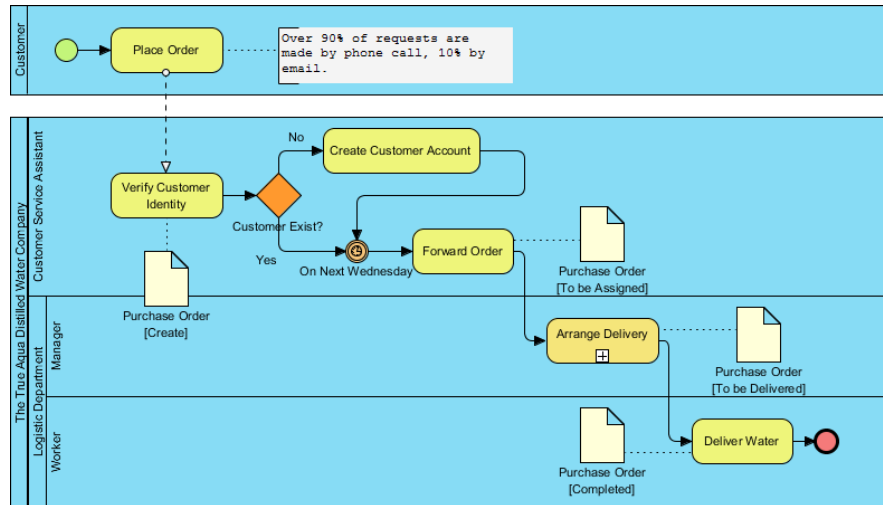


Fig 7.1: MPMN Model

## 7.9 Technology to Be Implemented

The "Readers Circle" project uses advanced technologies to deliver a robust and user-friendly platform. Below is an in-depth discussion of key technologies and their suitability for the project:

### Client-Server Application:

#### Characteristics:

Decentralized architecture where the client requests services from the server.

Handles real-time communication between the client (user's device) and server.

#### How It Works:

Users interact with the app (client), which sends requests to the server for data retrieval or processing.

The server processes requests, accesses the database, and sends responses back to the client.

### Example in Project:

A user searches for a book, and the server retrieves matching results from the PostgreSQL database.

Web Server Application:

**Characteristics:**

Centralized architecture serving web-based services via browsers or APIs.

Supports backend operations and data management.

**How It Works:**

The web server handles HTTP requests, interacts with the database, and returns results to the frontend.

**Example in Project:**

API endpoints process geolocation search queries, fetching books within a defined radius.

**Technology Overview:**

**Flutter:**

**Advantages:** Cross-platform, fast performance, native-like UI.

**Use in Project:** Developing the mobile app interface.

**Firebase:**

**Advantages:** Real-time data sync, scalable backend.

**Use in Project:** Handling authentication and real-time updates.

**PostgreSQL:**

**Advantages:** ACID compliance, advanced querying.

**Use in Project:** Managing structured data.

**Google Cloud:**

**Advantages:** Scalability, global availability.

**Use in Project:** Hosting and backend infrastructure.

## 7.10 Recommendations and Justifications

**Adopt Agile Methodology:** Facilitates iterative development and quick adaptation to feedback.

**Focus on Security:** Implement robust encryption to protect user data.

**Enhance User Experience:** Priorities responsive design and intuitive navigation.

**Regular Updates:** Continuously improve features based on user feedback.

**Build Community Engagement:** Incorporate forums and discussion boards for book enthusiasts.

## 7.11 Conclusion

The "Readers Circle" project's foundation offers a solid development roadmap. With the help of cutting-edge technology, user-centric design concepts, and functional and non-functional needs, the platform is ready to provide book lovers with a smooth and memorable experience. Suggestions guarantee scalability and ongoing enhancement, establishing "Readers Circle" as a dependable option for contemporary readers.

## Chapter 8: Exploration

### 8.1 Introduction

Nuances in dynamics of the project workflows and inter-connections also need to be understood to be explored. Use case and activity diagram are the main techniques to show these procedures and to stress behaviors and connections of the system. These diagrams help communicate with stakeholders, identify bottlenecks and visualize the development roadmap by putting processes in a graphic format.

**Use Case Diagram:** Use case diagram helps to illustrate the scope of functionalities, as well as how users (actors) interact with the system. So in “Readers Circle” users can perform book borrowing, listing and reporting as borrowers, lenders and administrators, for example.

#### **Activity Diagram:**

Activity diagrams depict decisions and actions as they flow through the system indicating parallel and sequential activities. For instance, searching, requesting, approving, paying, and adding comments are examples of actions added in the book loan scenario. These diagrams eliminate ambiguities and synchronizes development process to hold everyone in the loop of the system.

### 8.2 Old Full System Use Case

The old system of book lending and selling, often manual or partially digital, lacked efficiency and scalability. Below are the core use cases for the old system:

#### **Search for Books:**

Users manually search for books in libraries or among acquaintances.

**Logic:** There is no centralized platform, leading to inefficiency in discovering available books.

#### **Book Borrowing:**

Users borrow books through verbal agreements or handwritten records.

**Logic:** Absence of a structured system results in miscommunication and tracking issues.

### Book Selling:

Individuals sell books through physical marketplaces or online forums with limited reach.

**Logic:** Lack of geolocation tools and secure payment options reduces user trust.

### Feedback Mechanism:

Feedback is informal, often shared verbally or through text messages.

**Logic:** No formal system exists for recording user experiences or ratings.

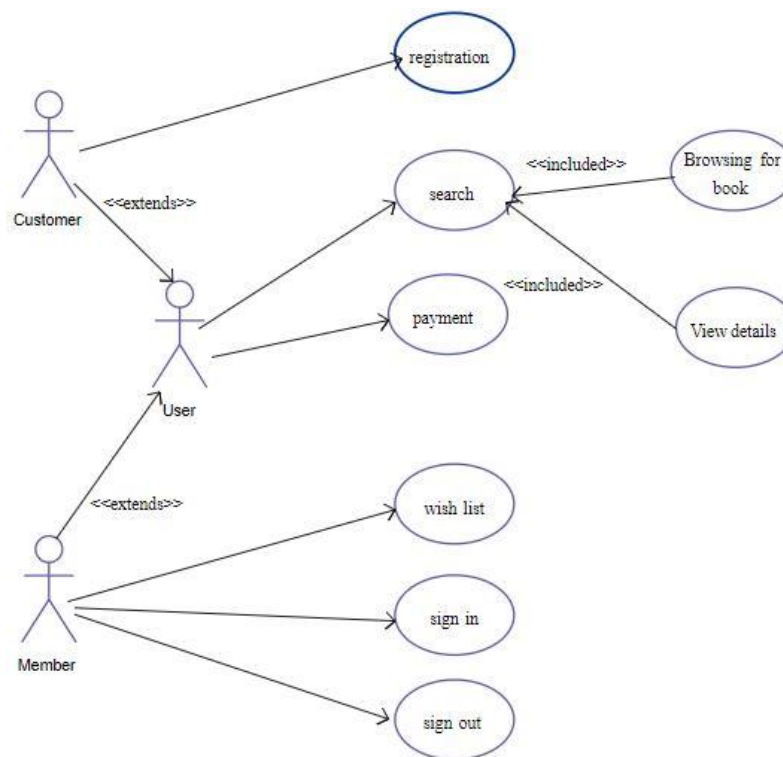


Fig 8.1: Old System Use Case Diagram

### 8.3 New System Full Activity Diagram

The new system for "Readers Circle" introduces a streamlined and scalable approach, addressing inefficiencies in the old system. Below are the primary activities:

#### **User Registration:**

Users sign up using their email, phone number, and other credentials.

**Logic:** Ensures that all users are verified and authenticated before accessing the system.

#### **Book Listing:**

Users list books with details like title, author, condition, and availability.

**Logic:** A central database allows users to manage and update their book listings.

#### **Book Search:**

Borrowers search for books using filters such as genre, location, or availability.

**Logic:** Advanced search algorithms and geolocation improve the user experience.

#### **Transaction Management:**

The system facilitates secure payments and borrowing agreements.

**Logic:** Integrated payment gateways ensure user trust and ease of use.

#### **Feedback Submission:**

Borrowers rate and review books and lenders.

**Logic:** Reviews enhance platform transparency and trust.

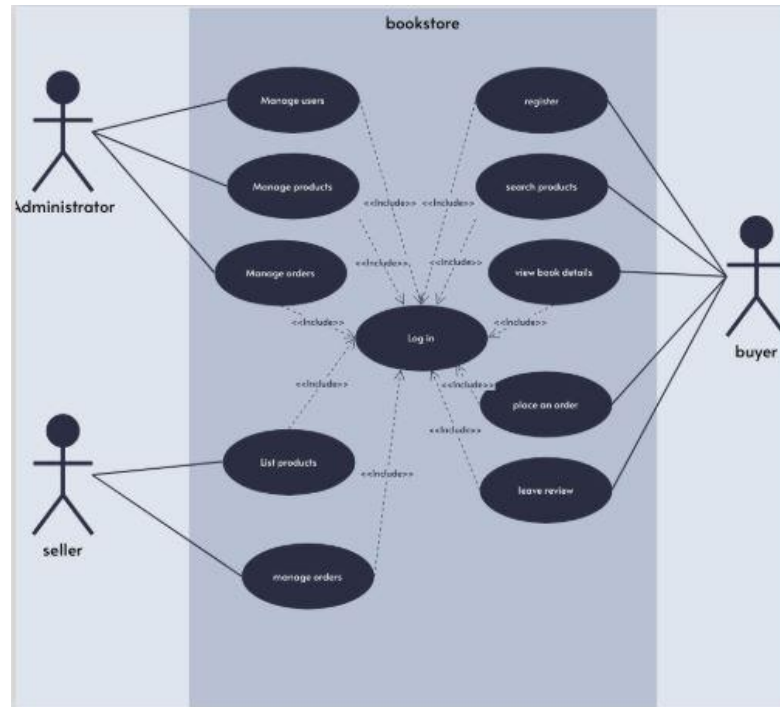


Fig 8.2: New System Use Case Diagram

## 8.4 Admin and Client Activity Diagrams

Admin Activity Diagram:

User Management:

Admins verify and manage user accounts.

Logic: Ensures only genuine users have access to the platform.

Book Oversight:

Admins monitor book listings to ensure compliance with policies.

Logic: Prevents inappropriate or duplicate listings.

Transaction Monitoring:

Admins track transactions for security and accuracy.

Logic: Ensures a secure and smooth transaction flow.

Report Handling:

Admins address user-reported issues.

Logic: Resolves disputes and enhances user satisfaction.

Client Activity Diagram:

Registration and Login:

Clients register and log in to their accounts.

Logic: Facilitates access to the platform.

Book Listing and Searching:

Clients add books or search for books to borrow or purchase.

Logic: Provides a seamless and user-friendly interface.

Transaction Initiation:

Clients initiate borrowing or purchasing requests.

Logic: Ensures proper tracking and processing of requests.

Feedback Submission:

Clients submit reviews and feedback post-transaction.

Logic: Builds a trusted community on the platform.

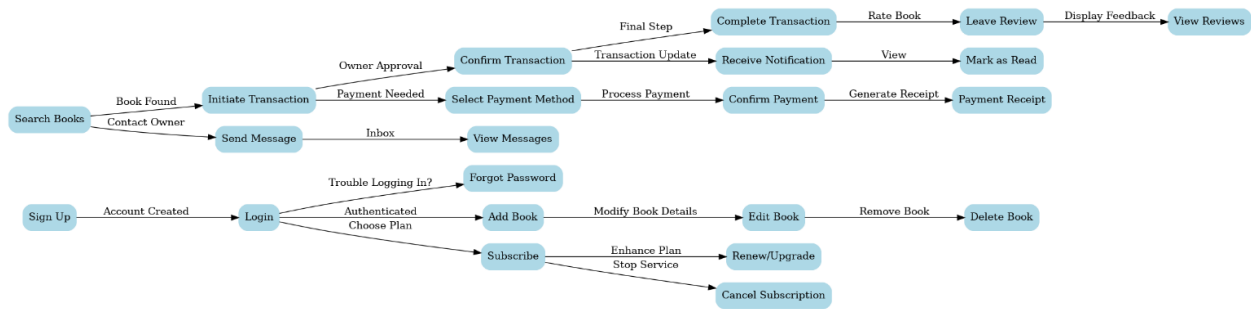


Fig 8.3: Full Activity Diagram

### 8.5 Prioritised Requirement List (PRL)

The "Readers Circle" project adopts the DSDM prioritisation technique, MoSCoW, to categorise requirements based on their importance and necessity for successful implementation. Below is the prioritised requirement table:

Priority	Requirement	Details
Must Have	User Registration and Authentication	Essential for allowing secure access to the platform.
	Book Listing and Search Features	Core functionality to enable users to find and list books.
	Secure Payment Integration	Mandatory for ensuring trusted transactions between users.
Should Have	Geolocation-Based Search	Enhances the user experience by offering nearby book availability.
	Review and Rating System	Builds trust within the user community.
	Push Notifications	Keeps users updated about transactions and offers.
Could Have	Discussion Forums	Adds a community engagement feature for book enthusiasts.

	Advanced Filtering Options	Enables users to refine search results further.
	Multi-Language Support	Expands usability to non-English speaking users.
Won't Have	Integration with External Marketplaces	Not required in the initial phase, to focus on platform-specific features.
	Book Delivery Services	Currently excluded to reduce operational complexity.

Table 8.1: Prioritized Requirement List Table

### 8.6 Prototype of the New System

To illustrate its usefulness, the "Readers Circle" app prototype displays the user interface and simple navigation. The design prioritizes utility, reactivity, and simplicity, concentrating on:

**Home Screen:**

Displays featured books, categories, and a search bar for quick access.

**User Profile:**

Allows users to manage their accounts, view borrowed or lent books, and edit preferences.

**Book Listing Page:**

Enables users to upload book details, including title, author, condition, and availability.

**Search Results Page:**

Provides a list of books matching the user's query, with filters for location, genre, and availability.

## Transaction Page:

Facilitates secure payments and displays transaction history.

## Feedback Section:

Users can leave reviews and ratings for books or lenders.

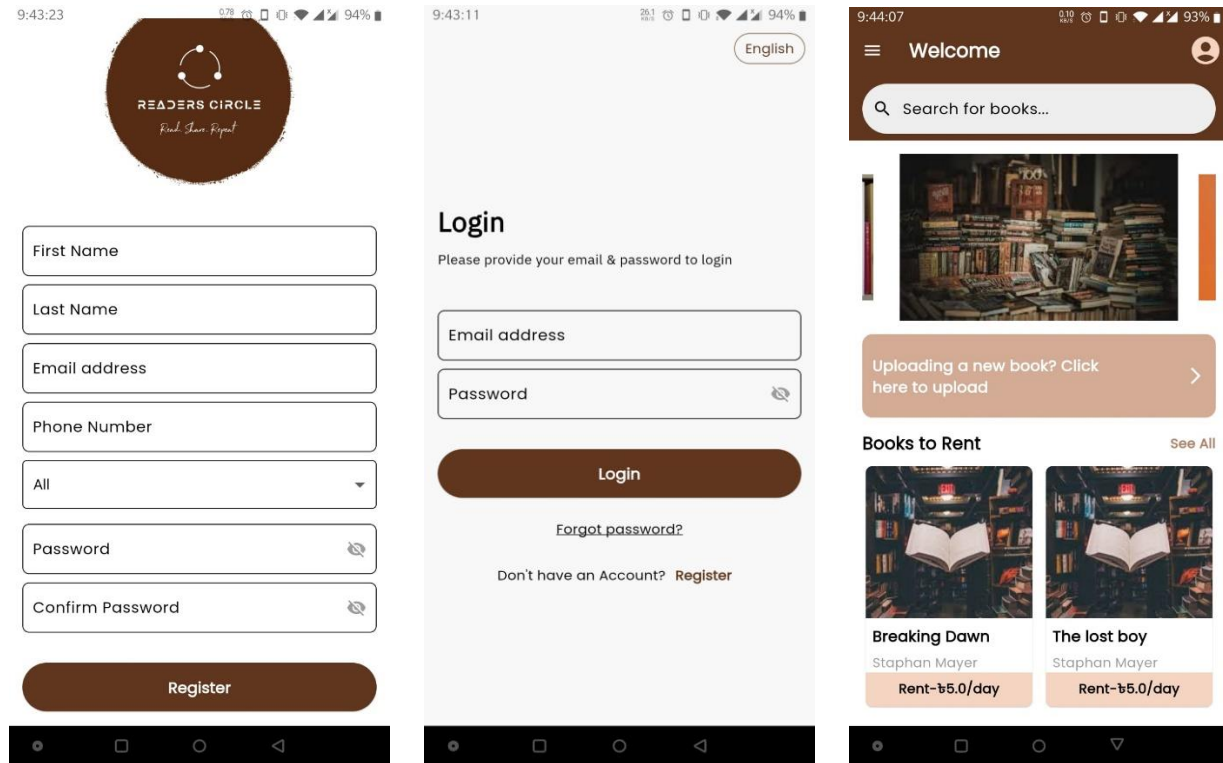


Fig 8.4: App Prototype

## Chapter 9: Engineering

### 9.1 Introduction

The engineering stage of the “Readers Circle” project focuses on transforming abstract concepts into operational solutions. To meet the requirements and objectives described above, as a part of this stage, it is required to create and structure modules, processes and components of the system. The engineering level ensure that the new system is robust, scalable, and easy mallardsystems.co.uk to use by providing comprehensive schematics and logical analysis. This chapter covers the usage of the new system, entity-relationship diagrams (ERDs), sequence diagrams, component diagrams, and deployment diagrams. These resources explain the functionality of the system in a thorough and clear manner.

### 9.2 New System Modules

Various types of diagramming are employed for this new "Readers Circle" system's mechanism is diagrammatically represented. Such diagrams, however, help improve transparency and efficiency during the development or deployment of such a system by depicting its structure, functionalities, and users.

#### 9.2.1 Use Case Diagram

The Use Case Diagram outlines the interaction between different actors (e.g., Admin, User) and the system functionalities. Key use cases include:

##### **User:**

- ✚ Register and log in.
- ✚ Search for books.
- ✚ Request borrowing or buying.
- ✚ Leave feedback and ratings.

##### **Admin:**

- ✚ Approve or reject book requests.

- ✚ Monitor and manage book listings.
- ✚ Resolve user-reported issues.

### Key Points:

- ✓ Gives a broad overview of the features of the system.
- ✓ Aids in determining the main ways that people and the system interact.
- ✓ Makes certain that every demand is noted and taken into consideration when developing.

### 9.2.2 Class Diagram

The Class Diagram represents the system's object-oriented structure, including classes, attributes, methods, and relationships. Key classes include:

User: Attributes such as *userID*, *name*, *email*, and *role* (admin or borrower).

Book: Attributes such as *bookID*, *title*, *author*, *genre*, *availability*.

Transaction: Attributes such as *transactionID*, *userID*, *bookID*, *status* (approved/rejected).

### Key Points:

- Makes the system's static structure visible.
- Draws attention to the connections between various things.
- Aids in creating the API endpoints and database schema.

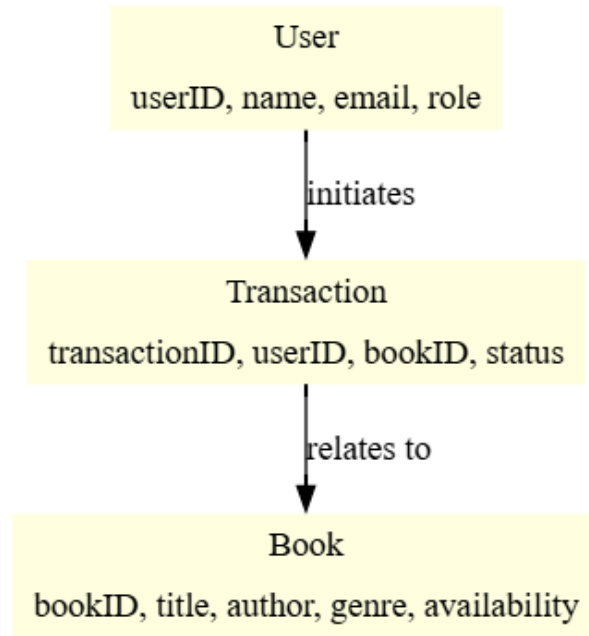


Fig 9.1: Class Diagram

### 9.2.3 ERD Diagram

The Entity-Relationship Diagram (ERD) illustrates the relationships between entities in the database. For "Readers Circle," key entities include *User*, *Book*, and *Transaction*.

Relationships:

User and Book: A user can list multiple books.

Book and Transaction: A book can be involved in multiple transactions.

User and Transaction: A user can initiate multiple transactions.

**Benefits:**

- Gives a clear grasp of the links between the data.
- Streamlines query optimisation and database design.
- Guarantees the integrity and consistency of data.

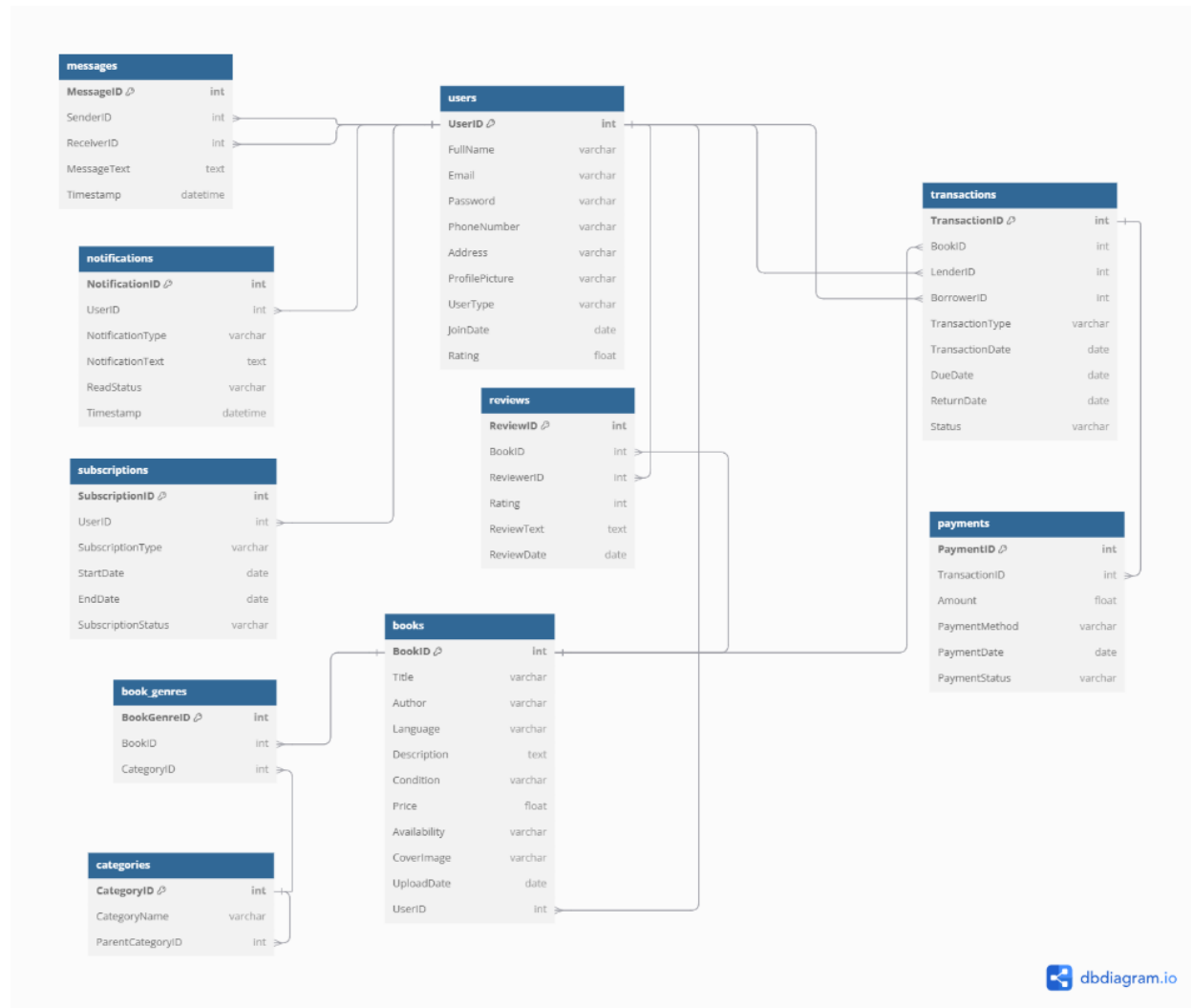


Fig 9.2: ERD Diagram

### 9.2.4 Sequence Diagram

The Sequence Diagram illustrates the dynamic interaction between users and the system for specific processes. For example, the borrowing process:

- User searches for books.
- System retrieves matching books from the database.
- User sends a borrowing request.
- Admin approves or rejects the request.
- User completes payment if approved.

- System updates the transaction status.

**Benefits:**

- Records the system's message and interaction flow.
- Aids engineers in comprehending how a system behaves over time.
- Locates possible workflow bottlenecks.

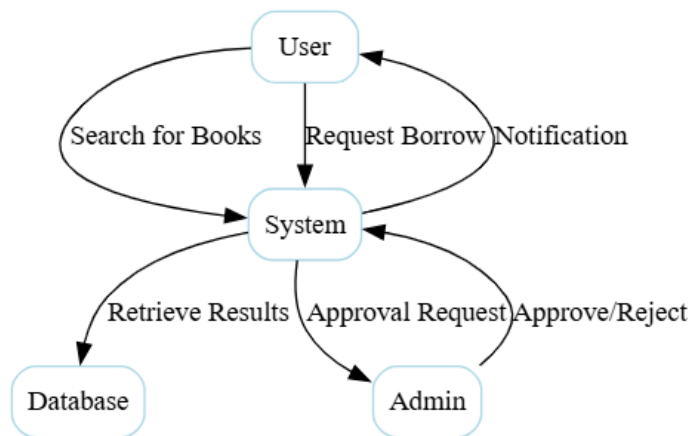


Fig 9.3: Sequence Diagram

**9.2.5 Component Diagram**

The Component Diagram breaks down the system into modular components, showing how they interact to form the overall structure. Key components include:

Frontend: Mobile app developed using Flutter.

Backend: APIs and server logic implemented with Node.js.

Database: PostgreSQL for storing and managing data.

Payment Gateway: Integration with platforms like bKash or Stripe.

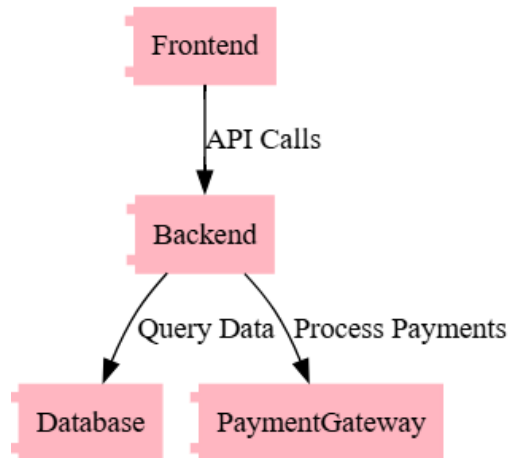


Fig 9.4: Component Diagram

**Benefits:**

- Demonstrates how system components are modular and reusable.
- Makes the relationships between the various components of the system more clear.
- Makes maintenance and debugging simpler.

**9.2.6 Deployment Diagram**

The Deployment Diagram illustrates the physical architecture of the system, showing how software and hardware components interact. Key nodes include:

User Device: Mobile app installed on Android/iOS devices.

Application Server: Hosted on Google Cloud to handle requests.

Database Server: Centralised database for storing user and book data.

**Benefits:**

- ✓ Provide information on the system's actual physical deployment.
- ✓ Guarantees load balancing and scalability across hardware resources.
- ✓ Draws attention to the ways that hardware and software interact.

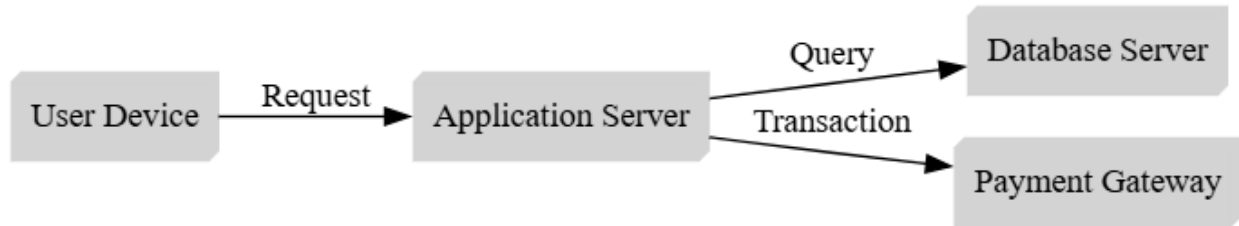


Fig 9.5: Deployment Diagram

### 9.2.7 System Interface

Prototyping is an experimental process in which research teams transform ideas from paper to digital. For "Readers Circle," prototypes demonstrate how users interact with the system interface and functionalities. Key features:

Login and Registration Screen: Enables users to sign up or log in securely.

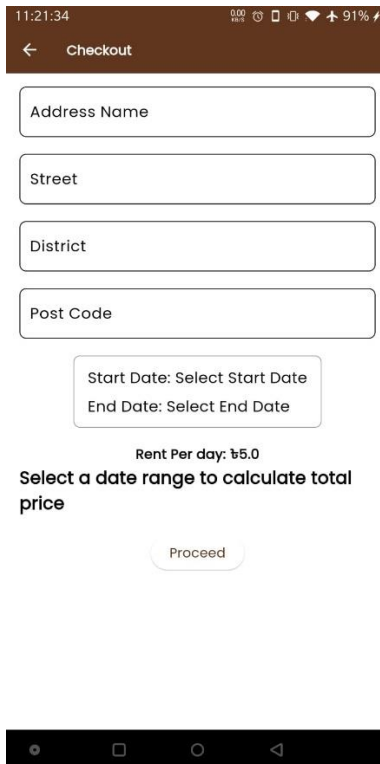
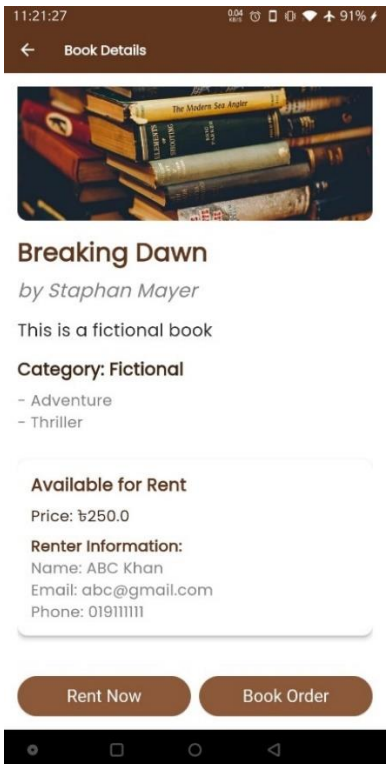
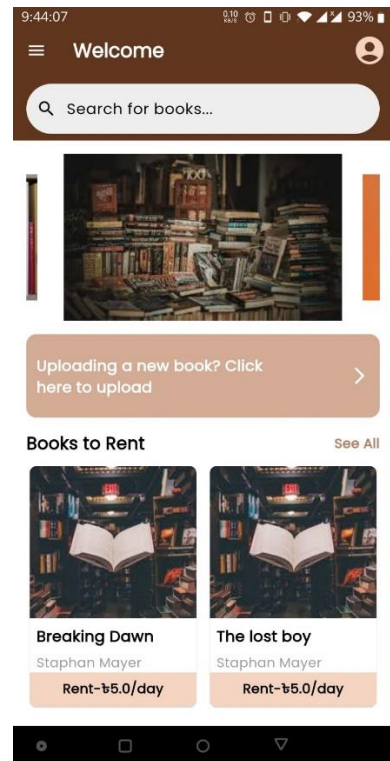
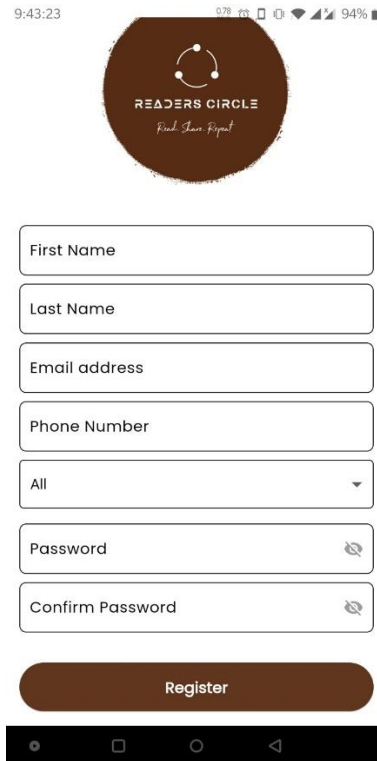
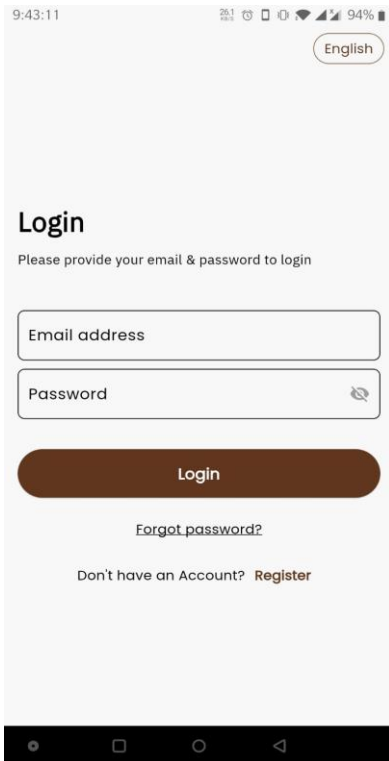
Home Screen: Displays featured books, categories, and a search bar.

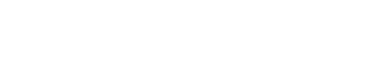
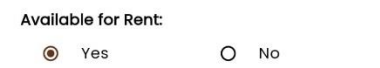
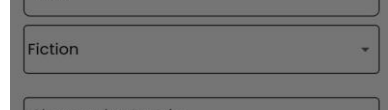
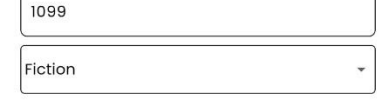
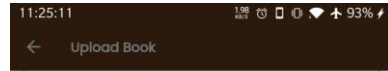
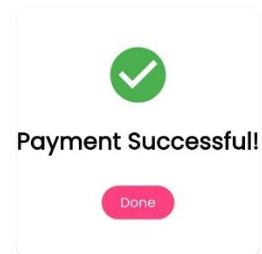
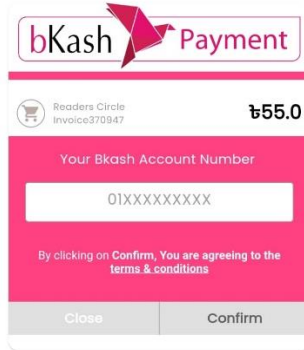
Book Details Page: Provides detailed information about selected books.

Transaction Page: Facilitates payments and tracks borrowing history.

#### **Benefits:**

- ✓ Improves and verifies the system design prior to deployment.
- ✓ Gathers user input on the UI to improve usability.
- ✓ Reduces the need for redesign, which speeds up the development process.





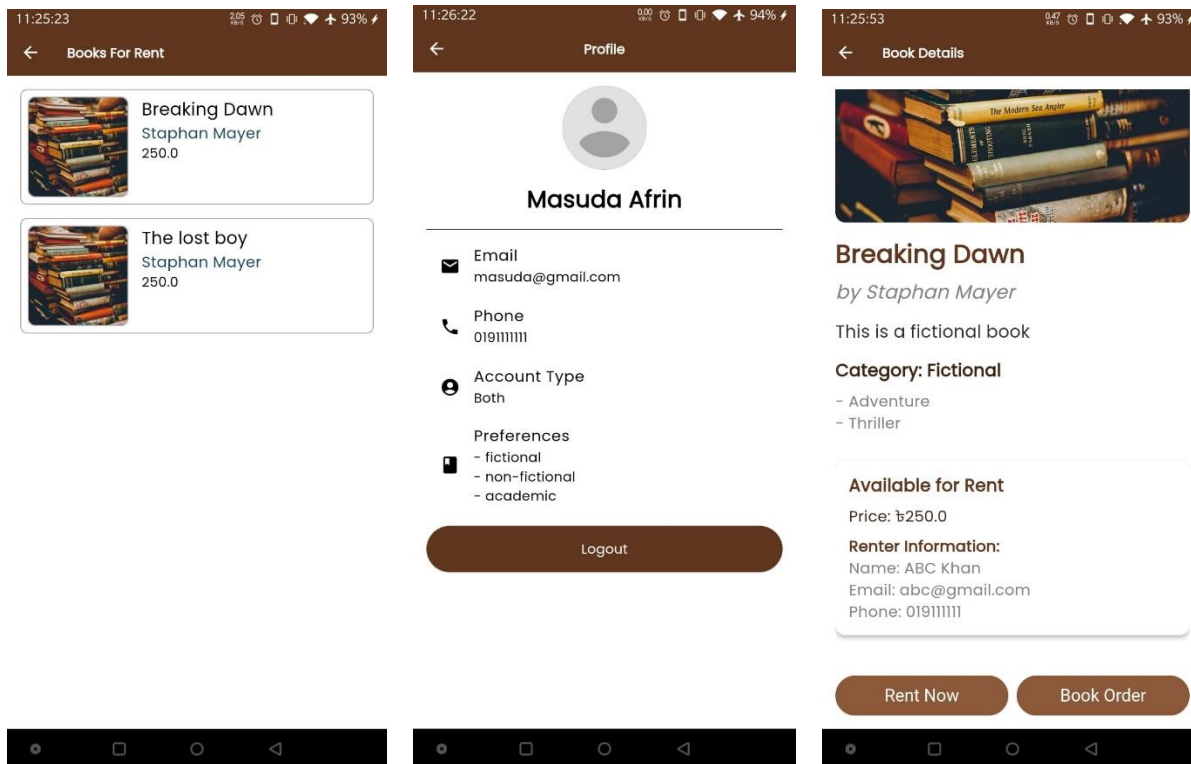


Fig 9.6: Application's User Demonstration

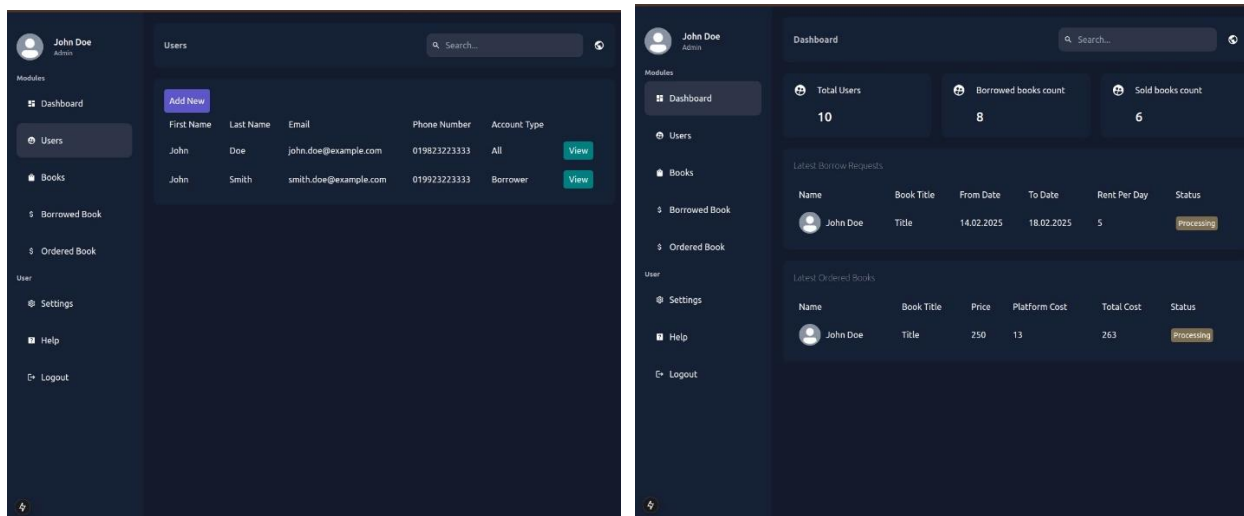


Fig 9.7: Application's Admin Panel Demonstration

### **9.3 New System Usage Status**

The new "Readers Circle" allows the users to read the books efficiently and easily, as well facilitate the process of borrowing/selling the books with other users. The system's modular is built with a modular system architecture for scalability and an intuitive experience to make it baby-simple to use. With new features of secure payment "Readers Circle" solves the user's pain points and is committed to building a diversified readers community.

### **9.4 Conclusion**

The design and engineering process turns the conceptual "Readers Cycle" into a concrete, functioning reality. With the use of structured diagrams and prototypes, in this chapter, the project is guaranteed to attain to an objective in which it becomes a user-worthy software improving on current technology and pace, and also provides a robust platform for future development.

## Chapter 10: Deployment/Development Phase

### 10.1 Introduction

The deployment phase of “Readers Circle” project is the final stage where the system developed is placed in production so that its intended users can utilize it. To ensure successful handover from development to production, this includes training, deployment to a live server, and on-going monitoring and the user training. For a seamless user experience and to achieve the project goals, it is crucial to make a swift deployment.

```
json
{
  "message": "Successfully book borrow request is sent",
  "code": 201,
  "data": {
    "id": 1,
    "borrow_uuid": "BRRW-XYZ-00012",
    "title": "Breaking Dawn",
    "author": "Staphan Mayer",
    "description": "This is a fictional book",
    "category_name": "Fictional",
    "for_rent": true,
    "available_for_rent": false,
    "rent_per_day": 2,
    "available_for_sell": false,
    "price": 250,
    "category_id": 1,
    "total_rent_cost": 40,
    "platform_fee": 5,
    "subcategories": [
      {
        "id": 2,
        "name": "Adventure"
      },
      {
        "id": 3,
        "name": "Thriller"
      }
    ]
  }
}
```

```
json
{
  "message": "Successfully fetched books.",
  "code": 200,
  "data": [
    {
      "id": 1,
      "title": "Breaking Dawn",
      "author": "Staphan Mayer",
      "description": "This is a fictional book",
      "category_name": "Fictional",
      "for_rent": true,
      "available_for_rent": true,
      "rent_per_day": 5,
      "available_for_sell": false,
      "price": 250,
      "category_id": 1,
      "subcategories": [
        {
          "id": 2,
          "name": "Adventure"
        },
        {
          "id": 3,
          "name": "Thriller"
        }
      ]
    }
  ]
}
```

```
json
{
  "message": "Successfully fetched preferences.",
  "code": 200,
  "data": [
    {
      "category_name": "Fiction",
      "subcategories": [
        "Adventure",
        "Classic Literature",
        "Fantasy"
      ]
    },
    {
      "category_name": "Non Fiction",
      "subcategories": [
        "History",
        "Science and Nature",
        "True Crime"
      ]
    }
  ]
}
```

```
Readers-Circle-main > lib > api > api_paths.dart > ...
1 const String baseUrl =
2   "https://07c850a5-314d-4b51-9809-74f5c601fc81.mock.pstmn.io/";
3
4 //END PATHS
5 const String loginPath = "users/sign-in";
6 const String registerPath = "users/sign-up";
7 const String categoriesPath = "categories";
8
```

```

61 //set preferences of a user
62
63 Future<int> setPreferences({
64   required List<String> prefs,
65   required String id,
66 }) async {
67   try {
68     final response = await getDio()!
69     .put("users/$id/set-preference", data: {"preferences[] ": prefs});
70     sharedPref.saveInt("preferences", prefs.length);
71     notifyListeners();
72     return response.statusCode;
73   } on DioException catch (e) {
74     debugPrint("Error: ${e.message}");
75     notifyListeners();
76     showMessageToast(message: "Something went wrong");
77     return e.response!.statusCode;
78   } finally {
79     notifyListeners(); // Notify listeners that the data has changed
80   }
81 }

```

```

12 class RegisterScreen extends StatefulWidget {
13   @override
14   RegisterScreenState createState() => RegisterScreenState();
15 }
16
17
18
19 class RegisterScreenState extends State<RegisterScreen> {
20   final _formKey = GlobalKey<FormState>();
21   final _firstNameController = TextEditingController();
22   final _lastNameController = TextEditingController();
23   final _emailController = TextEditingController();
24   final _phoneController = TextEditingController();
25   late AuthProvider provider;
26   var accountType = ["both", "borrower", "renter"];
27   String selectedType = "both";
28   final TextEditingController _passwordController = TextEditingController();
29   final TextEditingController _confirmController = TextEditingController();
30   bool _obscurePassword = true;
31   bool _obscureConfirmPassword = true;
32
33   @override
34   void initState() {
35     provider = Provider.of<AuthProvider>(context, listen: false);
36     super.initState();
37   }
38
39   @override
40   void dispose() {
41     _firstNameController.dispose();
42     _lastNameController.dispose();
43     _emailController.dispose();
44     _phoneController.dispose();
45     super.dispose();
46   }

```

```

1 class RentListScreenState extends State<RentListScreen> {
2   void initState() {
3     // TODO: implement initState
4   }
5
6   @override
7   Widget build(BuildContext context) {
8     return Scaffold(
9       appBar: CustomAppBar(
10        title: tr('Books For Sale'),
11      ), // CustomAppBar
12      body: context.watch<BookProvider>().booksForRentLoaded
13        ? ListView.builder(
14          padding: const EdgeInsets.all(16),
15          itemCount:
16            context.watch<BookProvider>().booksForRent.data.length,
17          itemBuilder: (context, index) {
18            final book =
19              context.watch<BookProvider>().booksForRent.data[index];
20            return InkWell(
21              onTap: () {
22                Navigator.pushNamed(context, Routes.bookDetail,
23                  arguments: book.id.toString());
24              },
25              child: Container(
26                padding: const EdgeInsets.all(8),
27                decoration: BoxDecoration(
28                  borderRadius: BorderRadius.circular(8),
29                  border: Border.all(color: Colors.grey),
30                ), // BoxDecoration
31                margin: const EdgeInsets.only(bottom: 16),
32                child: Row(

```

```

1 class Data {
2   String? token;
3   int? id;
4   String? firstName;
5   String? lastName;
6   String? accountType;
7   String? email;
8   String? phoneNumber;
9   List<dynamic>? preferences;
10
11   Data({
12     this.token,
13     this.id,
14     this.firstName,
15     this.lastName,
16     this.accountType,
17     this.email,
18     this.phoneNumber,
19     this.preferences,
20   });
21
22   factory Data.fromJson(Map<String, dynamic> json) => Data(
23     token: json['token'] as String?,
24     id: json['id'] as int?,
25     firstName: json['first_name'] as String?,
26     lastName: json['last_name'] as String?,
27     accountType: json['account_type'] as String?,
28     email: json['email'] as String?,
29     phoneNumber: json['phone_number'] as String?,
30     preferences: json['preferences'] ?? <String>[],
31   );

```

```

1 json
2
3 {
4   "message": "Successfully book order list is fetched",
5   "code": 200,
6   "data": [
7     {
8       "id": 1,
9       "order_uuid": "BUY-XYZ-00012",
10      "book": {
11        "title": "Breaking Dawn",
12        "author": "Staphan Mayer",
13        "description": "This is a fictional book",
14        "category_name": "Fictional",
15        "for_rent": false,
16        "available_for_rent": false,
17        "rent_per_day": null,
18        "available_for_sell": true,
19        "price": 250,
20        "category_id": 1,
21        "total_rent_cost": null,
22        "platform_fee": 5,
23        "subcategories": [

```

Fig 10.1: Application Development Code

## 10.2 Development Approach

The development process was broken down into specific, achievable tasks, which guaranteed a methodical approach and expedited efficiency. The following tools and technologies were employed:

Frontend Development:

Flutter: For creating a responsive and user-friendly mobile application interface.

Backend Development:

Node.js and Prisma: To manage server-side logic and database interactions.

PostgreSQL: As the database system for managing user and book data.

The structured development approach made the process efficient and allowed for continuous integration and testing.

## 10.3 Possible Problem Breakdown

All challenges were broken down into smaller, more manageable problems in order to speed up the development process. This method simplified implementation and guaranteed on-time completion. A smooth development procedure required disassembling the system. The "Readers Circle" project's breakdown procedure is as follows:

Database Development

- Define primary and foreign keys.
- Create queries.
- Analyse system data.
- Create an Entity-Relationship Diagram (ERD).
- Establish the system database using PostgreSQL.

Frontend Development

- Create a paper prototype using Figma or Sketch.
- Use Flutter for cross-platform mobile app development.
- Ensure the interface is responsive on all devices.
- Test UI components for usability and consistency.

### Backend Development

- Use Node.js for server-side programming.
- Implement Prisma for database interaction.
- Add, update, and delete books and user profiles (CRUD operations).

### Administration Management

- Manage book listings and transactions.
- Approve or reject user requests.
- Monitor user activity and feedback.

### Client Management

- Enable profile updates.
- Allow book borrowing and transaction tracking.
- Facilitate feedback and rating submissions.

### Seller Management

- Allow sellers to add and manage book listings.
- Provide transaction history and statistics for sellers.

### System Testing and Deployment

- Conduct functional and non-functional testing.
- Deploy the application on Google Cloud for scalability.
- Monitor live operations and fix any bugs identified post-deployment.

## 10.4 Conclusion

Both its deployability and the soundness of every in it incorporated component are ensured by the deployment phase of the "Readers Circle" system. Through reorganization of the system development strategy and the trouble-out process, the success of the project was assured by the takeover from conception to implementation.

## **Chapter 11: Testing Phase**

### **11.1 Introduction**

To ensure that the system is up to its requirements, testing is a crucial part of the software development life cycle. Different testing procedures were adopted to check the usability, the security, the performance and the correct behavior of the "Readers Circle" project. Remembering that phase 1 was a phase for early troubleshooting in order to ensure a seamless user usage.

### **11.2 Types of Test Case Scenarios and Importance**

1. Functional test: It checks whether the system behaves in accordance with the expectations.
2. Unit Testing: Tests components and structure for correctness.
3. Integrational Testing: Validates that the modules function in accordance with one another.
4. System Testing: To ensure the functioning of the total system.
5. Acceptance Testing: validates the system against the user requirements and business requirements.
6. Security Testing: Vulnerability Tests and Threat Protection.
7. Usability Testing: Tests the ease with which a user can use the app and user interface.

### 11.3 Test Plan Acceptance Table

Sl.No.	Test Name	Specification	Expected Outcome
1	Unit Testing	Verifies individual components like user login, book listing, etc.	Each component works as expected.
2	Module Testing	Tests the modules such as admin dashboard or user profile management.	Modules function correctly as standalone units.
3	Integration Testing	Tests interactions between modules like payment gateway and backend server.	Seamless data flow between modules.
4	Acceptance Testing	Validates the system meets user and business requirements.	System satisfies user needs and project objectives.
5	Security Testing	Identifies vulnerabilities and ensures data protection.	Application is secure and resistant to attacks.
6	Usability Testing	Evaluates the ease of use of the application.	User-friendly and intuitive interface.

Table 11.1: Test Plan Acceptance Table

### 11.4 Unit Testing

Testing individual program modules or components separately is the main goal of unit testing. This guarantees that every function runs properly prior to integration.

Test Case ID	Component	Input Data	Expected Result	Actual Result
UT-01	Login Function	Valid email and password	User logs in successfully	Pass
UT-02	Book Search	Search by valid keywords	Displays matching book results	Pass

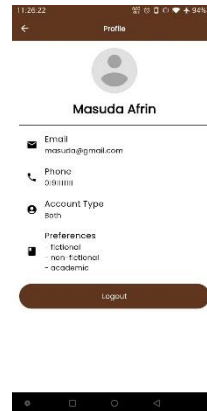


Fig 11.1 Unit Testing

## 11.5 Module Testing

Module testing examines the logic and design of individual modules to confirm that they operate as intended.

Test Case ID	Module	Scenario	Expected Result	Actual Result	Status
MT-01	Admin Dashboard	Approve a book request	Request status changes to "Approved"	Pass	Pass
MT-02	User Profile	Update profile information	Information updated successfully	Pass	Pass

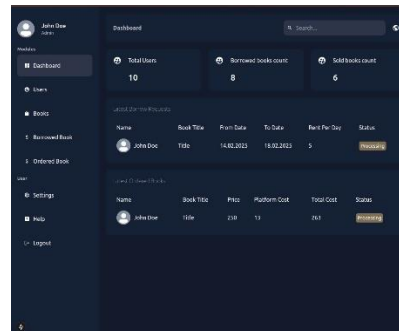


Fig 11.2 Module Testing

## 11.6 Integration Testing

By concentrating on data flow and component-to-component interactions, integration testing guarantees that modules operate as intended.

Test Case ID	Modules Involved	Scenario	Expected Result	Actual Result	Status
IT-01	User-Book Module	User requests a book	Request reaches admin for approval	Pass	Pass
IT-02	Payment Gateway	User completes payment	Transaction recorded and confirmed	Pass	Pass

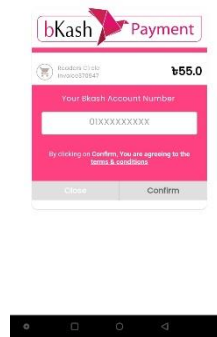


Fig 11.3 Integration Testing

### 11.7 Acceptance Testing

To make sure the system satisfies user wants and expectations, acceptance testing compares it to business requirements.

Test Case ID	Test Scenario	Input/Condition	Expected Outcome	Actual Outcome	Status
AT-01	Book Borrowing Process	User submits valid request	Request is approved or rejected	Pass	Pass
AT-02	Feedback Submission	User leaves a rating or comment	Feedback is recorded successfully	Pass	Pass

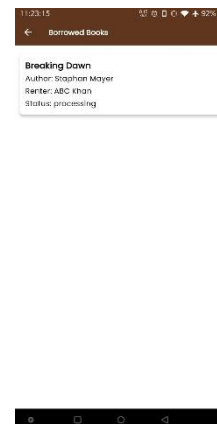


Fig 11.4 Acceptance Testing

### 11.8 Security Testing

By ensuring that the application is safe from threats and vulnerabilities, security testing protects user information and transactions.

Test Case ID	Test Scenario	Condition	Expected Outcome	Actual Outcome	Status
ST-01	SQL Injection Prevention	Malicious input in search query	Input is rejected; no data leakage	Pass	Pass
ST-02	Secure Payment Gateway	User completes a transaction	Payment details are encrypted	Pass	Pass

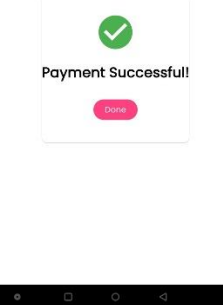


Fig 11.5 Security Testing

## 11.9 Usability Testing

Usability testing assesses how end users can easily navigate and use the program interface.

Test Case ID	Test Scenario	Input/Condition	Expected Outcome	Actual Outcome	Status
UT-01	Navigation Simplicity	User explores app features	User finds desired features easily	Pass	Pass
UT-02	Accessibility for All Users	Screen reader tested for blind users	Features are fully accessible	Pass	Pass

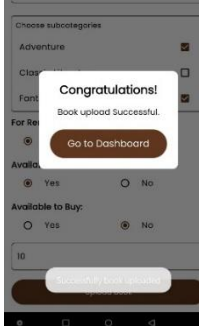


Fig 11.6 Usability Testing

## 11.10 Conclusion

The "Readers Circle" project was very well tested before its publication to attention that all of its modules and pack were working properly.consumer. In order to identify and correct potential issues, several types of tests like unit, module, integrated and usability tests were performed. Security and user experience taking precedence in the system, it is ready to meet the needs of its target market.

## **Chapter 12: Implementation Phase**

### **12.1 Introduction**

The Readers Circle project will move into the redesign, development, and implementation phase. This step guarantees that all components of the system are properly installed, configured and operation for the end users. It also includes training, scaling and keeping the system reliable during operation.

### **12.2 System Deployment**

The introduction of a new system requires careful planning and execution. In an effort to ensure low downtime and an easy transition for the user "Readers Circle", the suspect reason's deployment was gradual.

#### **Techniques Used:**

1. Environment Testing: Performed pre-deployment testings on staging environments.
2. Environment Configuration: Setup servers, databases and network resources the right way.
3. Migration of Data: Transferred user data, book details and transaction data from old system to new system.
4. Automation Deployments: Experienced in tools such as Docker and Jenkins for automation deployments.
5. Parallel Approach: Operating both the old and new systems simultaneously to ensure a seamless transition.
6. Pilot Installation: Introducing the new system to a limited number of users in a restricted environment.
7. Phased: Implementing features gradually to manage risk.
8. Big Bang Implementation (size of atomic transactions): Replacing old system with new system all at once.

### 12.3 Training

As a way of getting administrators and users comfortable with the new system, training sessions were conducted. Error handling, role-based content, and system navigation were topics of those sessions.

Title	Description	Duration
Access the System	Training on login, navigation, and system overview.	2 hours
Role Management	Explaining user roles (admin, client, seller).	1.5 hours
Verify Users	Managing user profiles and transaction verification.	1 hour

Table 12.1 Training Management

### 12.4 Big Bang Implementation

This became necessary to the "Readers Circle" system's rapid expansion. That meant totally swapping out the old system for the new one, ensuring that all features worked simultaneously.

#### Advantages:

- Quick transition to the new system.
- Immediate availability of all features.
- Cost-effective for limited resources.

### 12.5 Scaling

Scalability is defined as the ability to increase system capabilities to accommodate growing user needs. Scaling considerations of "Readers Circle" will be:

- **Horizontal Scaling:** Adding more servers to distribute user load.
- **Vertical Scaling:** Upgrading existing servers for better performance.
- **Database Sharding:** Dividing the database into smaller, more manageable pieces.

## 12.6 Load Balancing

Load balancing is responsible for distributing user requests across servers in a uniform manner. It made this happen using tools such as NGINX and AWS Load Balancer to:

- Reduce response times.
- Enhance user experience by maintaining high availability.

## 12.7 Recommended Implementation Process

For "Readers Circle," the recommended process combines pilot and phased implementations:

- **Pilot Deployment:** Test the system with a small user group.
- **Monitor and Evaluate:** Collect feedback and resolve issues.
- **Phased Rollout:** Gradually introduce the system to larger user groups.
- **Full Deployment:** Implement the system for all users.

## 12.8 Conclusion

With "Readers Circle" to work, the stage of practice is essential. The project ensures that the transition from development to operation is fluent while ensuring high quality and enduser satisfaction by integrating proven scaling approaches, comprehensive training and effective deployment strategies.

## **Chapter 13: Critical Appraisal and Evaluation**

### **13.1 Project Evaluation**

A project's objectives, results, and future goals are all reviewed as part of the evaluation process. Prior to beginning the "Readers Circle" project, a clear set of goals was determined to direct its creation and execution. Frequent assessment guarantees adherence to these objectives and pinpoints areas in need of development.

#### **13.1.1 Success Rate**

The success rate of the project is determined by its ability to meet academic, business, and personal objectives.

##### **Academic Objectives:**

- Offer a novel approach to renting and selling books.
- Show off your hands-on understanding of system development and design.
- Use contemporary frameworks and programming technologies.
- Display how testing and deployment techniques are integrated.
- Encourage learning through practical implementation.

##### **Business Objectives:**

- Provide a platform that is easy to use for book transactions.
- Obtain scalability to handle the expansion of users in the future.
- Make money by charging for transactions and subscriptions.
- Create a reliable online marketplace for readers.
- Increase operational effectiveness by implementing automated procedures.

##### **Personal Objectives:**

- Develop your ability to think critically and solve problems.
- Gain proficiency with Flutter, Node.js, and PostgreSQL.
- Develop your teamwork and project management skills.

- To further your profession, create a project that is worthy of a portfolio.
- Enhanced comprehension of the software development lifecycle.

### 13.1.2 How Much Better Could Have Been Done

This project has met most of its objectives; however, certain areas could have been improved, such as:

**Feature Completeness:** The AI-driven book recommendations and geolocation-based book discovery features were planned but not implemented.

**UI/UX Refinement:** Additional user feedback could have been incorporated to fine-tune the application's interface.

**Testing Depth:** While testing was rigorous, more automated testing tools could have been utilized for better coverage.

**Marketing and Outreach:** More resources could have been allocated to promote the platform to a broader audience before launch.

**Documentation:** Enhanced documentation of APIs and system flows would aid in future scalability.

### 13.1.3 How Much Better Solutions of the Features?

#### Enhancements:

- ✚ Implementing AI features to offer personalised recommendations based on user behaviour.
- ✚ Adding geolocation services to allow users to discover nearby book lenders or sellers.
- ✚ Enhancing the payment system by integrating multiple payment gateways.
- ✚ Introducing gamification elements like badges or rewards for frequent users to boost engagement.

### 13.1.4 Which Features Could Be Untouchable?

Features Critical for System Integrity:

- ❖ **Secure Authentication and User Roles:** Essential for maintaining user trust and system integrity.
- ❖ **Book Listing and Searching:** Core to the platform's functionality.
- ❖ **Transaction Management:** Critical for ensuring a smooth buying, selling, and lending experience.
- ❖ **Feedback and Ratings:** Crucial for maintaining transparency and user satisfaction.

### 13.2 Strengths of the System

- ✚ **User-Friendly Interface:** Intuitive design that ensures ease of use.
- ✚ **Scalability:** Backend architecture is built to accommodate future user growth.
- ✚ **Secure Transactions:** Robust payment gateway integration ensures user confidence.
- ✚ **Community Building:** Platform fosters interaction among book lovers.
- ✚ **Operational Efficiency:** Automation of processes like approvals and notifications.

### 13.3 Weaknesses of the System

- ❖ **Feature Limitations:** Lack of AI-driven recommendations and geolocation services.
- ❖ **Dependence on Internet Connectivity:** Limited usability in areas with poor network coverage.
- ❖ **Initial Cost:** Development and operational costs might pose challenges for small-scale adoption.
- ❖ **Limited Languages:** Currently supports only a single language.
- ❖ **Maintenance Dependency:** Requires dedicated technical resources for updates and maintenance.

### 13.4 Future Development

The following areas are planned for future development:

- AI Integration: Implement AI-driven recommendations to enhance user experience.
- Geolocation Feature: Add location-based discovery of books and lenders.
- Multi-Language Support: Expand the platform to support multiple languages.
- Advanced Filtering Options: Provide users with detailed search capabilities.
- Mobile App Enhancements: Add offline mode for limited functionality without internet.
- Marketing and Partnerships: Build partnerships with local bookstores and libraries to increase reach.

### **13.5 Product Evaluation**

They provide an interesting solution to the sharing/selling of books woes. The technique has a high potential for optimization, if some features need to be done. Vital to its continued growth and operating at scale is the continual upgrading by user feedback and technological advances.

## **Chapter 14: Lessons Learned**

### **14.1 Introduction**

It's been very fun designing and actually putting into place the Readers Circle project. It through the capability to r&d new technology, solution to real world application problem, result in a working application. Despite all of the bumps and road blocks along the way, the journey was one of growth, learning and pride.

### **14.2 What Have I Learned?**

The practical experience I gained from this project was extremely beneficial to me in many aspects. Firstly, I was able to develop as a full-stack programmer, working on the back-end in Node.js and Prisma and front-end on Flutter. I've built scalable and fast applications with these technologies. Second, I had found out how important time management and project organization can be. Splitting the workload and making a work breakdown structure and Gantt charts allowed us to put timekeeping on control and balance between developing the game and moderating and organizing stuff. Also it was my first encounter with Agile and Scrum and I could chose my personal way of how to approach the project. Finally, I strengthened my test and bug hunting skills. My job was bug fixes and testing to make sure the system works, which instilled patience into me, and an eye for bugs. Overall, I felt this project prepared me for other projects and work in software development.

### **14.3 What Problems Have I Faced?**

- ❖ **Understanding the Problem:** Initially, it was difficult to identify the actual problem and user needs. There was a lot of need for research, interviews, and clarity to define accurately the scope of the project.
- ❖ **Project Approach:** The time that each approach would take was also difficult to determine. Each had its shortcomings and advantages, and choosing the most appropriate one took a long time. The most critical factor was my interest in trying Agile, which offered me the flexibility to iterate. **Final Phase Challenges:** Data migration and server settings were among the complications brought forth by the system's deployment.

- ❖ Analysis Phase: Initially, it was quite difficult for me to generate diagrams like sequence diagrams and ERDs. For proper system representation, it was vital to analyze it carefully and make necessary improvements.
- ❖ Final Phase Challenges: Upon system deployment, the following challenges emerged: data migration and server setting.
- ❖ Learning During Development: It took me a lot of time to get acquainted with technologies such as Prisma and Flutter.
- ❖ Overcoming Coding Issues: A common issue that took lots of team-work and perseverance was debugging the integration of payment gateways and databases, among others.

#### 14.4 What Solutions Occurred?

- Problem understanding: I have conducted user input survey and several brainstorming meetings to understand the requirement.
- Project approach: With the help of my friends and mentors, I have analyzed different available approaches and choose Agile as it was quite adjustable with the help of peers.
- Diagram issues: I have reviewed online resources and used available techs which can ease the procedure and boost diagram accuracy.
- Learning tool: I have checked different community forum and tutorials to enhance my knowledge towards new frameworks and technology.
- Debugging: I have used available debugging tools and proper logging and also, I have done peer reviews this way I identified and resolved issues.

#### 14.5 Conclusion

“Readers Circle” offered a platform for extensive learning and skill building. I learned something new in every subsequent stage of planning and implementing, dealing with obstacles, and addressing challenges, which will impact my future projects. By addressing the issues and centering the project on continuous improvements, I believe this project has already laid fruitful groundwork for my upcoming projects.

## **Chapter 15: Conclusion**

### **15.1 Project Summary**

“Readers Circle” offered a platform for extensive learning and skill building. I learned something new in every subsequent stage of planning and implementing, dealing with obstacles, and addressing challenges, which will impact my future projects. By addressing the issues and centering the project on continuous improvements, I believe this project has already laid fruitful groundwork for my upcoming projects.

### **15.2 Goals of the Project**

#### **15.2.1 Fulfilled Goals**

The main goals of the system are:

- ✓ To offer a user-friendly book trading platform.
- ✓ To create a system design that is both scalable and safe.
- ✓ To develop a prototype that shows the solution's viability.
- ✓ To guarantee operational effectiveness by means of strong system processes.

#### **15.2.2 Goals That Are Not Fulfilled**

- AI Integration: Personalized recommendations remain pending for future development.
- Geolocation Feature: Location-based book discovery is planned but not implemented.

### **15.3 The Success of the Project**

The "Readers Circle" system address its main objectives adequately, providing a flexible and useful system to suit the demand of user. Its solid workings, secure technology, and user-friendly layout lay a foundation for further enhancements. User feedback reveals high satisfaction with the features that have been deployed.

## 15.4 What I Have Done in the Documentation

The documentation includes comprehensive details about each phase of the project, including:

- ✓ Initial study and feasibility analysis.
- ✓ Methodology, testing, and implementation phases.
- ✓ Critical appraisal and evaluation of the system.
- ✓ Lessons learned and areas for improvement.

## 15.5 What I Have Learned in the Documentation

Creating the documentation provided valuable insights into structured project reporting. I learned how to:

- Organise information systematically for clarity and precision.
- Use diagrams and charts to visually represent complex data.
- Critically evaluate system strengths and weaknesses.
- Identify future development opportunities and document them effectively.

## 15.6 Value of the Project

The "Readers Circle" project is valuable for:

- ✚ Promoting a sustainable reading culture by enabling easy access to books.
- ✚ Offering a scalable business model with revenue potential.
- ✚ Enhancing my skills in software development, planning, and collaboration.
- ✚ Contributing to the academic and professional domain as a practical case study.

## 15.7 My Experience

This project has been life altering.” It challenged me to think critically and approach resources and mentors while also performing decently at various tasks. It was educational and rewarding in the process of converting ideas to a functioning framework. I had difficulties along the way, and overcame them which makes me feel a sense of achievement and confidence that I can take into my further pursuits.

## **References**

<https://www.managementstudyguide.com/resource-allocation.htm>

<https://www.pmi.org/learning/library/project-planning-frameworks>

<https://www.timemanagement.com/>

<https://flutter.dev/>

<https://www.techopedia.com/scalability-applications>

<https://www.investopedia.com/terms/f/feasibility-study.asp>

<https://hbr.org/scope-analysis>

<https://www.postgresql.org/docs/design.html>

<https://www.sciencedirect.com/sdlc-review>

<https://www.ibm.com/integration>

<https://www.techrepublic.com/economic-it-feasibility>

<https://www.techopedia.com/technical-feasibility-guide>

<https://www.visual-paradigm.com/functional-requirements>

<https://www.geeksforgeeks.org/client-server-vs-web-app>

<https://www.researchgate.net/book-lending-economics>

<https://agilemanifesto.org/>

<https://www.cio.com/waterfall-sdlc>

[https://www.tutorialspoint.com/sdlc\\_models.htm](https://www.tutorialspoint.com/sdlc_models.htm)

<https://www.businessanalysisexperts.com/functional-requirements>

<https://creately.com/component-diagram>

<https://www.docker.com/get-started>

<https://www.nginx.com/resources/glossary/load-balancing/>

<https://www.softwaretestinghelp.com/stlc>

<https://www.nngroup.com/articles/usability-testing-101>

<https://www.techopedia.com/strengths-and-weaknesses-guide>

<https://www.projectmanager.com/blog/lessons-learned>

<https://www.stackoverflow.blog/documentation-matters>

<https://www.medium.com/it-project-growth>

## Appendix:

### Use Case Diagram Description Tables

Table 1: Login

Attribute	Details
Use Case Name	Login
Primary Actors	User
Secondary Actors	System Admin
Pre-condition	User must have registered credentials.
Dependency	Database connectivity for user authentication.
Basic Flow	User enters credentials → System verifies → User is logged in.
Post-condition	User is redirected to the homepage.

Table 2: Manage Reports

Attribute	Details
Use Case Name	Manage Reports
Primary Actors	Admin
Secondary Actors	System
Pre-condition	Admin must be logged in.
Dependency	Database connectivity for accessing reports.
Basic Flow	Admin selects reports → Edits or views them → Saves changes.
Post-condition	Reports are updated in the system.

Table 4: Manage Users

Attribute	Details
Use Case Name	Manage Users
Primary Actors	Admin
Secondary Actors	System
Pre-condition	Admin must be logged in.
Dependency	Database connectivity for managing user details.
Basic Flow	Admin views user list → Updates user roles or details → Saves changes.
Post-condition	User information is updated in the system.

Table 6: Manage Profile

Attribute	Details
Use Case Name	Manage Profile
Primary Actors	User
Secondary Actors	System
Pre-condition	User must be logged in.
Dependency	Database connectivity for profile data.
Basic Flow	User navigates to profile page → Updates profile → Saves changes.
Post-condition	Profile information is updated in the system.

## Plagiarism Report:

0242310012091029

### ORIGINALITY REPORT

<b>11</b> %	<b>9</b> %	<b>1</b> %	<b>9</b> %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

### PRIMARY SOURCES

<b>1</b>	Submitted to NCC Education Student Paper	<b>3</b> %
<b>2</b>	dspace.daffodilvarsity.edu.bd:8080 Internet Source	<b>2</b> %
<b>3</b>	Submitted to University of Greenwich Student Paper	<b>2</b> %
<b>4</b>	Submitted to Daffodil International University Student Paper	<b>&lt;1</b> %
<b>5</b>	Submitted to IMI University Centre Student Paper	<b>&lt;1</b> %
<b>6</b>	etd.uum.edu.my Internet Source	<b>&lt;1</b> %
<b>7</b>	Submitted to University of Wales Institute, Cardiff Student Paper	<b>&lt;1</b> %
<b>8</b>	www.ceiba.com.co Internet Source	<b>&lt;1</b> %
<b>9</b>	www.villedesaintgobain.com Internet Source	<b>&lt;1</b> %
<b>10</b>	Submitted to HRUC Student Paper	<b>&lt;1</b> %
<b>11</b>	pearl.plymouth.ac.uk Internet Source	<b>&lt;1</b> %

Submitted to Coventry University