



# **Software Requirement Engineering**

## **Project Report**

### **Submitted to :**

Dr. Muhammad Adnan

### **Group Members:**

1. Ali Hassan      Sap Id -----
2. Sufyan Ashraf      Sap Id -----

Section 4B

*Riphah School of Computing & Innovation*

*Raiwind Campus*

## **Table of content**

### **1. Project description**

1.1 Project scope

1.2 Major features

### **2. Stackholders**

### **3. Elicitation**

3.1. Interviews

3.2. Observations

3.3. Brain Storming

### **3.4 Prototyping**

### **4. Set of requirements**

4.1 Functional requirements

4.2 Non Functional requirements

4.3 Constraint requirements

4.4 Doman requirements

### **5. Implementation of Analysis techniques**

### **6. Moscow prioritization techniques**

### **7. RTM**

### **8. Graphical Models of requirement**

### **9. Activity Diagram**

### **10. Sequence Diagram**

### **11. Class diagram**

### **12. Validations**

# Project Name :

## “ Borrow Me “

### 1. Project Scope

Borrow Me is an innovative online platform facilitating the borrowing and lending of items within communities. Users can easily lend out underutilized items or borrow what they need temporarily, promoting resource sharing and reducing consumption. With user-friendly listings, robust search, secure messaging, and a rating system, Borrow Me fosters trust and convenience. It addresses the common problem of idle possessions and encourages a sustainable lifestyle. Whether it's tools, books, or recreational gear, Borrow Me connects borrowers with lenders seamlessly. By maximizing resource use, users save money and contribute to environmental conservation. Borrow Me empowers individuals to make meaningful connections and share within their communities. Its intuitive interface and comprehensive features revolutionize ownership norms. Borrow Me heralds a future of collaborative consumption, transforming how people engage with possessions and each other.

#### 1.1 Major Features

- **User Authentication:**
  1. Secure registration and login for users.
  2. Password recovery and account management.
- **Profile Management:**
  1. Users can create, view, and edit their profiles.
  2. Profile includes personal information, lending/borrowing history, and ratings.
- **Item Listing:**
  1. Lenders can list items with descriptions, images, and prices.
  2. Items are categorized for easy navigation.
- **Item Searching:**
  1. Robust search functionality for borrowers to find items.
  2. Filters based on category, location, availability, and price.
- **Item Detail Page:**
  1. Detailed information for each listed item.
  2. Includes availability, rental terms, images, and borrowing options.
- **Cart Management:**

1. Users can add items to a cart for borrowing.
  2. Ability to remove items from the cart before confirming the borrow.
- **Secure Messaging:**
    1. Lenders and borrowers can communicate through a secure messaging system.
    2. Ensures privacy and security of user communications.
  - **Secure Payments:**
    1. Integration with a secure payment gateway for handling transactions.
    2. Supports multiple payment methods.
  - **Feedback and Rating System:**
    1. Users can provide feedback and rate items they have borrowed.
    2. Ratings help build trust within the community.
  - **History Tracking:**
    1. Users can track their lending and borrowing history.
    2. Provides detailed records of past transactions.

## 2 . Stakeholder Group and Role/Responsibility

- **Users :** Lenders , Borrowers
- **System Administrators ,Customer Support Teams**
- **Local Organizations ,Environmental Advocacy Groups**
- **Angel Investors**
- **Corporate Sponsors**
- **Project Manager, Developers ,UX/UI Designers, Development Team**
- **Legal Advisors**
- **Marketing Strategists**
- **Payment Gateway Providers**
- **Messaging Service Providers ,Hosting Providers ,Data Protection Authorities**

## 3.Elicitation Plan

Elicitation is the process of gathering requirements and information from stakeholders to understand their needs, expectations, and constraints regarding a software project or system. It involves various techniques and methods to extract relevant data and insights that will guide the development process. Here's a detailed description of elicitation:

The elicitation techniques we are using:

- Interviews
- Observation
- Brainstorming
- Prototyping

## **3.1. Interview**

1.1.1. Interview technique is commonly used for elicitation because it allows for in-depth exploration of a topic or subject matter. Here are some reasons why we use the interview technique:

- Clarification
- Rich Data Collection:
- Build Rapport:
- Uncover Unanticipated Insights

### **3.1.2. What motivated you to consider using a platform like Borrow Me for borrowing and lending items?**

Answer: "I'm drawn to Borrow Me because it offers a convenient solution to a common problem – the need for occasional items without the burden of ownership. It aligns with my values of sustainability and community collaboration."

### **3.1.3. What is your budget?**

Answer: 10 to 12 laks

### **3.1.4. Do you want a login System?**

Answer: Yes, user will use there ID's to login.

### **3.1.5. How a person borrow item/product?**

Answer: To borrow an item through Borrow Me, users sign up, search for the desired item, initiate a request, and communicate with the lender to agree on terms. Once terms are set, they arrange pickup or delivery, use the item, and return it after the agreed period. Feedback is exchanged, fostering trust and accountability within the community.

### **3.1.6. How a person lend item/product?**

Answer: To lend an item on Borrow Me, create a listing with details and availability, Setting Availability and Terms , Upload Photos, Providing Additional Information ,respond to borrow requests, and arrange pickup or delivery. Hand over the item, track the borrowing period, and receive it back. Leave feedback to build trust within the community.

### **3.1.7. Is user edit there profile?**

Answer: Yes, User can edit there profile

### **3.1.8. How user Authenticate there account?**

Answer: User authenticate there account by using Phone no. and email

**3.1.9. How long can a person borrow a item?**

Answer: The maximum amount of time a person can borrow a item for is 3 month and it also depends on rapport of both parties.

**3.1.10. Do you have any preferences regarding the platform's interface design and usability?**

Answer: Yes , I preference OLX but this just for sale not for borrow & lend.

**3.1.11. What you want payment method?**

Answer: All online payment method like as Bank transfers , Easypaisa , Fonepay , JazzCash , Keenu Pakistan , Payoneer , MasterCard , MoneyGram etc.

**3.1.12. Is there any discount & voucher ?**

Answer: Yes , but it depend on lender & special events .

**3.1.13. What you want to availability of software?**

At 24/7 hours but it never down at day time.

**3.1.14. What types of items would you be most interested in borrowing or lending through Borrow Me?**

Answer: Tools, office items, Household products like furniture – basically anything that I don't need to own year-round but might need occasionally.

**3.1.15. What process if item/product damage?**

Answer : It should optional , either user repair it or pay money.

**3.1.16. Is there any notification on over due date to return item/product?**

Answer : Yes , Of course , user get notification before 2 day of fix date.

**3.1.17. What method are used searching item/product?**

Answer: First user select category than search by Name and some specifications .

**3.1.18. Which device support your software?**

Answer: On each smart device like Android , Apple or PC etc

**3.1.19. What criteria If person fails to achive his/her reserve item/product for borrowing ?**

Answer: He/She must pay fine 200 per day .

### **3.1.20. What happen if a person who lending item/product have get bad rating 5 time?**

Answer: After 3 time bad rating System send a warning notification but on 5 time bad rating his/her account blocked

## **3.2. Observation**

Observation is utilized to directly witness and understand how individuals interact with products, services, or environments in real-life settings. It provides valuable insights into behavior, context, and user experiences, aiding in the validation of data and identification of improvement opportunities.

### **3.2.1. User Authentication and Profile Management:**

- Users must be able to log in using their phone number and email.
- They should have the ability to edit their profiles.

### **3.2.2. Item Borrowing and Lending:**

- Borrowers can search for items, initiate requests, and communicate with lenders.
- Lenders create listings, set availability and terms, respond to requests, and arrange pickup or delivery.
- Borrowing duration is flexible, with a maximum limit of 3 months, depending on mutual agreement.
- Users should receive notifications before the due date for returning borrowed items.
- Options for users to repair damaged items themselves or pay for damages.

### **3.2.3. Interface Design and Usability:**

- Users prefer an interface similar to OLX, emphasizing simplicity and ease of use.

### **3.2.4. Payment Methods and Discounts:**

- Support for various online payment methods like bank transfers, mobile wallets, and international payment gateways.
- Discounts and vouchers may be available, depending on lender discretion and special events.

### **3.2.5. Availability and Support:**

- The platform should be available 24/7, with minimal downtime during the day.
- Accessible on all smart devices, including smartphones, tablets, and PCs.

### 3.2.6. Search and Recommendation:

- Users can search for items by category, name, and specifications.

### 3.2.7. Penalties and Account Management:

- Penalties imposed on borrowers for failing to return reserved items and a warning system for lenders receiving bad ratings, with account suspension after repeated offenses.

## 3.3. Brainstorming

During a brainstorming session, stakeholders could generate ideas such as:

2.3.1. Gamification elements to encourage active participation and positive behavior within the Borrow Me community.

- ✓ **Points System:** Users could earn points for various actions such as listing items, borrowing items, leaving reviews, and referring friends to the platform
- ✓ **Leaderboards:** Borrow Me could feature leaderboards that showcase top borrowers, lenders, and contributors within the community.
- ✓ **Achievement Badges:** Users could earn badges for reaching certain milestones or completing specific tasks
- ✓ **Virtual Currency:** Borrow Me could introduce a virtual currency system where users earn or purchase credits that can be used to unlock premium features, access exclusive items, or participate in auctions.
- ✓ **Leveling Up:** Users could progress through different levels based on their activity and engagement within the platform. Each level could unlock new privileges or benefits, providing users with a sense of accomplishment and progression.
- ✓ **Social Interactions:** Borrow Me could integrate social features such as liking, commenting, and sharing on listings.

2.3.2. Integration with social media platforms to expand user reach and facilitate sharing among existing networks.

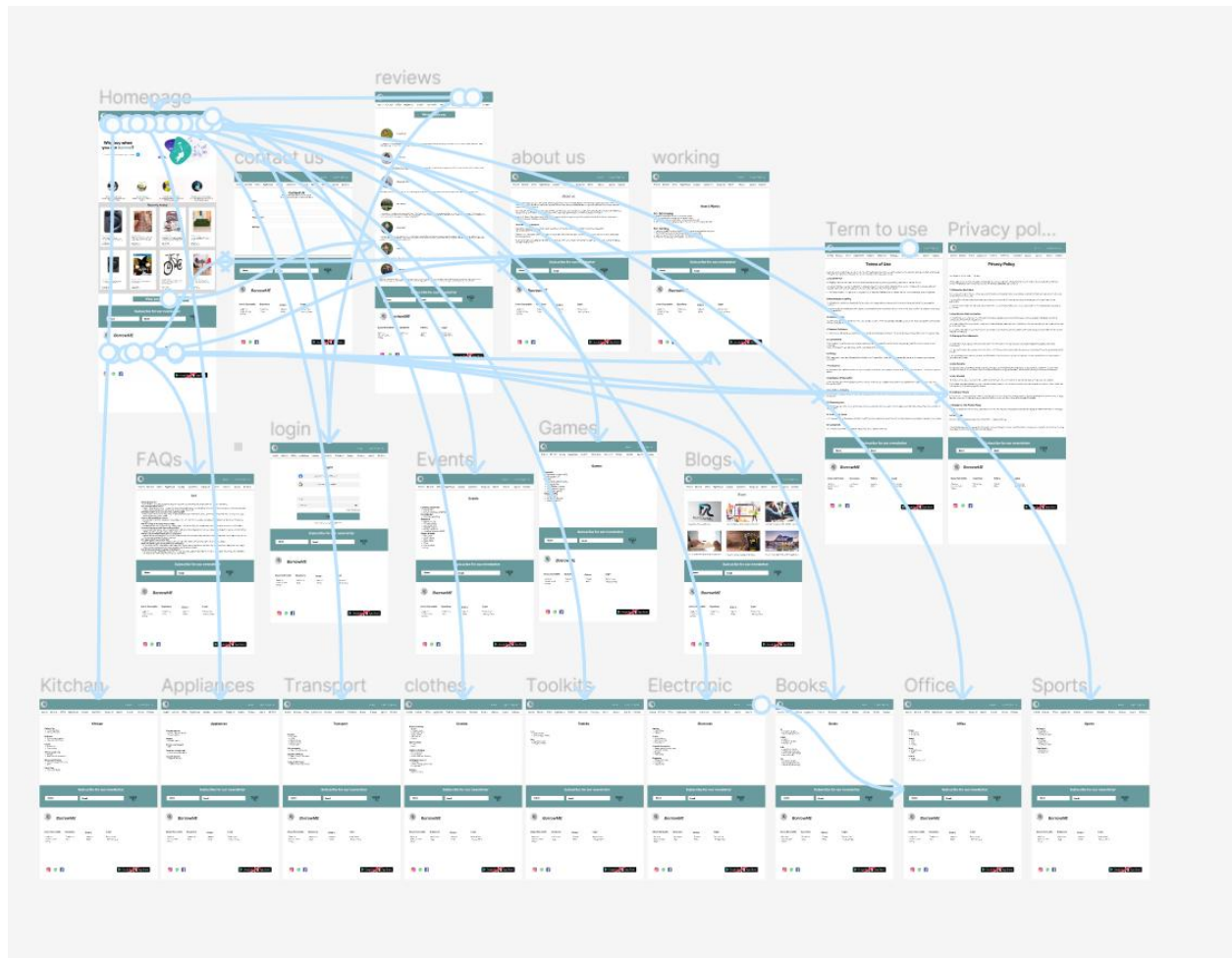
2.3.3. Partnerships with local businesses or organizations to offer exclusive discounts or perks to Borrow Me users.

2.3.4. Implementing a feature for users to request specific items they need but can't find listed, fostering a sense of community collaboration.

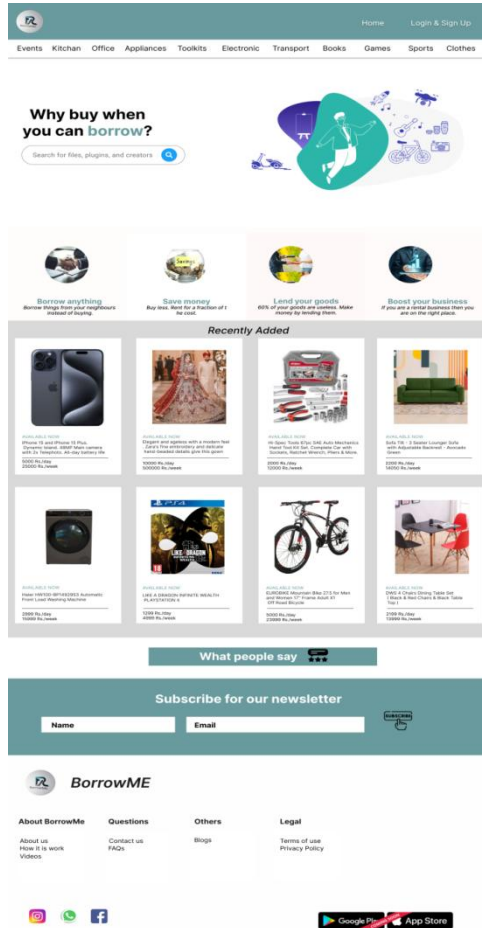
2.3.5. Hosting offline events or meetups for Borrow Me users to connect in person and exchange items directly.



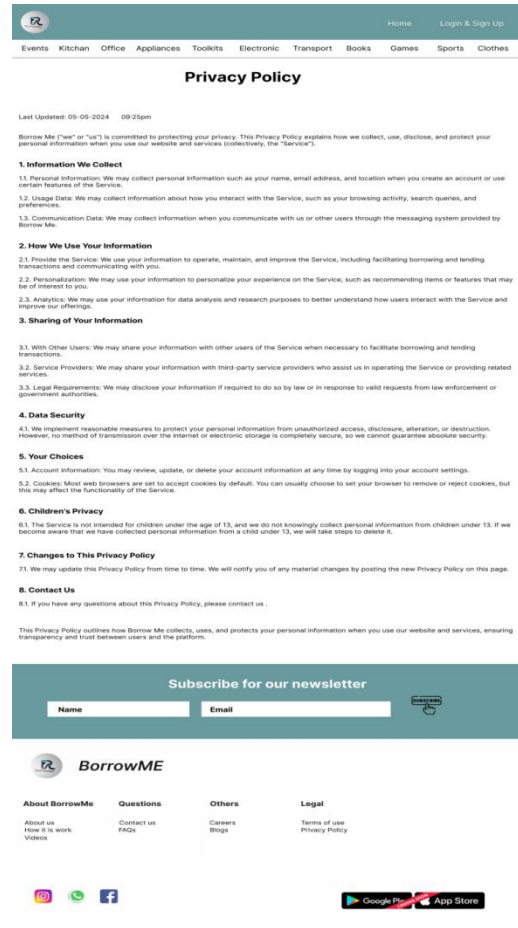
## 3.4. Prototyping

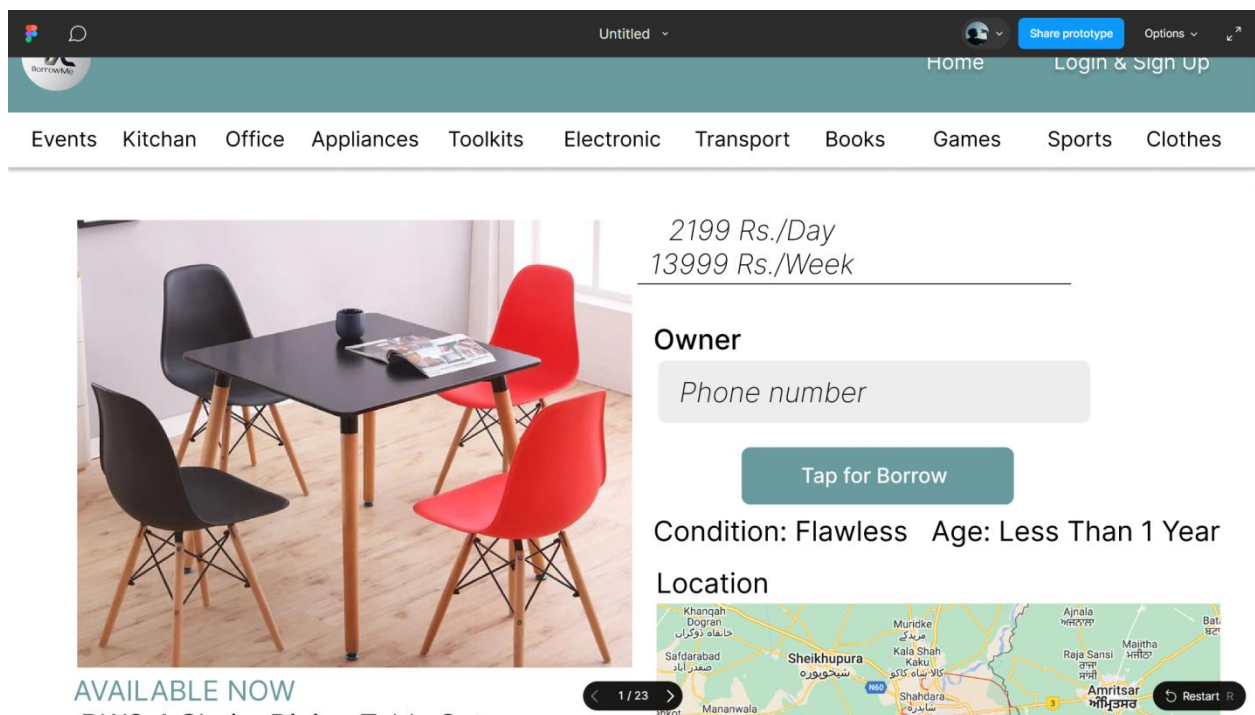
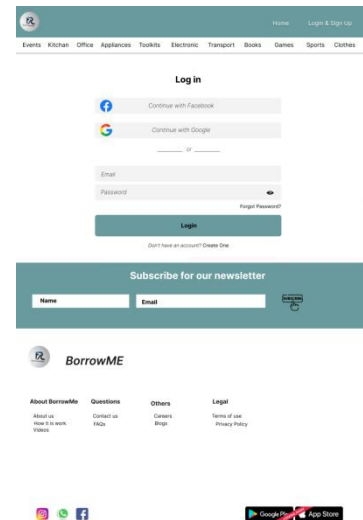
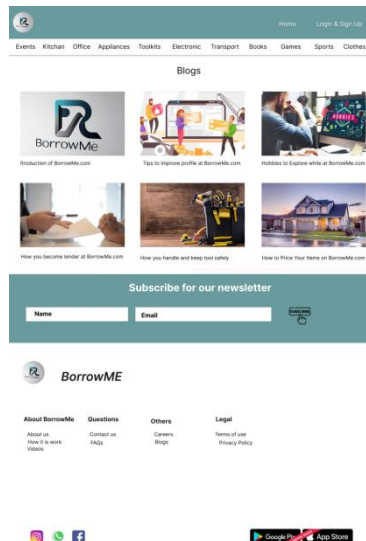


Homepage



## Privacy Policy





## 5. Set of Functional, Non-Functional, Constraint & Domain Requirements

## 5.1 Functional Requirements

| Functional Req. #ID | Function Name       | Functional Requirement Description  |
|---------------------|---------------------|---|
| FR1                 | User Authentication | Users can register and log in securely.   |
| FR2                 | Profile Management  | Users can create and edit their profiles.   |
| FR3                 | Secure Payments     | The website handles secure payments.  |
| FR4                 | Item Searching      | Users browse and search for items to borrow.  |
| FR5                 | History Tracking    | Users track their lending & borrowing history.  |
| FR6                 | Cart Management     | Users can add and remove items from their cart.                                       |
| FR7                 | Direct Borrowing    | Users borrow items directly from the website.   |
| FR8                 | Feedback System     | Users can provide feedback on borrowed items.   |
| FR9                 | Item Detail Page    | Each item has a detailed page with availability, rental terms, and borrowing options. |
| FR10                | Secure Messaging    | Lenders and borrowers can communicate through secure messaging.                       |
| FR11                | Item Listing        | Lenders can list items with descriptions, images, and prices.                         |

## 5.2 Non-Functional Requirements

| NF Req. #ID | Function Name                | NF Req. Description   |
|-------------|------------------------------|---|
| NFR1        | Performance Efficiency       | Fast and responsive.  |
| NFR2        | Scalability                  | Performs well under increased workload.                                       |
| NFR3        | Reliability                  | Operates without breaking down.   |
| NFR4        | Security                     | Ensures safe and secure data, information, and payment processing.            |
| NFR5        | Maintainability              | Easy to maintain and upgrade.   |
| NFR6        | Accessibility                | Accessible to users, including those with disabilities.                       |
| NFR7        | Cross-Platform Compatibility | Works on different browsers and devices, including mobile phones and tablets. |
| NFR8        | Usability                    | Easy to use and helps users find what they need.                              |
| NFR9        | Compliance                   | Adheres to relevant laws and regulations.                                     |

## 5.3 Constraint Requirements

| Constraint Req. ID | Function Name    | Constraint Requirement Description   |
|--------------------|------------------|--|
| CON 1              | Budget           | Costs must not exceed 10 to 12 lakhs.  |
| CON 2              | Time Constraint  | Develop and launch within a specified timeframe.                                   |
| CON 3              | Technology Stack | Use technologies (React, Node.js, MongoDB).  |
| CON 4              | Payment Gateway  | Integrate specified payment methods (Easypaisa, JazzCash, Bank(MCB, HBL, Allied)). |

## 5.4 Domain Requirements

| Domain Req. ID | Function Name       | Domain-Specific Requirement Description          |
|----------------|---------------------|--|
| DOM 1          | Industry Relevance  | Geared towards industry relevance                |
| DOM 2          | Expertise alignment | Users deal with product owner                    |
| DOM 3          | Agreement Expert    | Approve user agreement with correct & originally |

## 6. Implementation of Analysis Techniques

### Checklist

| Requirement Id | Complete | Non-Ambiguous | Testable | Necessary | Confliction |
|----------------|----------|---------------|----------|-----------|-------------|
| FR 1           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 2           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 3           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 4           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 5           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 6           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 7           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 8           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 9           | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 10          | ✓        | ✓             | ✓        | ✓         | ✗           |
| FR 11          | ✓        | ✓             | ✓        | ✓         | ✗           |

The provided checklist is designed to ensure the effective implementation of analysis techniques in a web project. It covers eleven functional requirements (FR 1 to FR 11) and evaluates each

requirement based on five criteria: completeness, clarity (non-ambiguity), testability, necessity, and absence of conflict.

Each requirement is marked as complete, non-ambiguous, testable, and necessary, indicating they are well-defined, clear, verifiable through testing, and essential for the project. Importantly, none of the requirements have any conflicts, which suggests they are consistent with each other and do not pose any implementation issues.

## 7. MoSCoW prioritization technique

The MoSCoW prioritization technique helps categorize features into four groups: Must have, Should have, Could have, and Won't have (this time). Let's apply this technique to the use cases in your project.

| MoSCoW Method   |  |   |  |
|---|--|---|--|
| BorrowMe website  |  |   |  |
| Must Have   | Should Have  | Could Have  | Won't Have   |
| Login : Essential for user authentication and security.             | Add item : Important for adding new items to lend.               | Feedback : Useful for maintaining quality and trust.        | Penalties : Can be deferred to a later release as it adds complexity |
| Register : Necessary for creating new user accounts.                | Remove item : Important for managing available items.            | Check feedback : Nice for lenders to see borrower feedback. |  |
| Authentication :Critical for verifying user identity.               | Edit profile :Important for keeping user information up to date. | Save to cart : : Enhances the user experience.              |  |
| Search Item:: Core functionality for finding items to borrow.       | Update item : Important for tracking the borrowing process.      |   |  |
| Borrow item: Core functionality for borrowing process.              |  |   |  |
| Make payment: Necessary for completing financial transactions.      |  |   |  |
| Notification: Essential for informing users about important events. |  |   |  |
| Return item: Essential for completing the borrowing cycle.          |  |   |  |

## 8. Requirement Traceability Matrix (RTM)

| Req. ID | FR 1 | FR 2 | FR 3 | FR 4 | FR 5 | FR 6 | FR 7 | FR 8 | FR 9 | FR10 | FR11 |
|---------|------|------|------|------|------|------|------|------|------|------|------|
| FR 1    | *    | *    |      |      |      |      |      |      |      |      | *    |
| FR 2    |      | *    |      |      |      |      |      |      |      | *    |      |
| FR 3    |      | *    | *    |      |      |      |      |      |      |      | *    |
| FR 4    |      |      |      | *    |      |      |      |      | *    |      |      |
| FR 5    |      |      |      | *    | *    |      |      |      |      |      | *    |
| FR 6    |      |      |      |      |      | *    | *    |      |      |      | *    |
| FR 7    |      |      |      |      |      |      | *    |      |      | *    |      |
| FR 8    |      |      | *    |      |      |      | *    | *    |      |      |      |
| FR 9    |      |      |      | *    |      |      |      | *    | *    |      |      |
| F 10    |      | *    | *    |      |      |      |      |      | *    | *    |      |
| FR 11   | *    |      |      | *    | *    |      |      |      |      |      | *    |

### Mapping Descriptions:

#### **FR1: User Authentication**

Mapped to: FR2 (Profile Management), FR11 (Item Listing)

#### **FR2: Profile Management**

Mapped to: FR1 (User Authentication), FR10 (Secure Messaging)

#### **FR3: Secure Payments**

Mapped to: FR2 (Profile Management), FR11 (Item Listing)

#### **Item Searching**

Mapped to: FR5 (History Tracking), FR9 (Item Detail Page)

#### **FR5: History Tracking**

Mapped to: FR4 (Item Searching), FR11 (Item Listing)

#### **FR6: Cart Management**

Mapped to: FR7 (Direct Borrowing), FR11 (Item Listing)

#### **FR7: Direct Borrowing**

Mapped to: FR6 (Cart Management), FR10 (Secure Messaging)

#### **FR8: Feedback System**

Mapped to: FR3 (Secure Payments), FR7 (Direct Borrowing)

### FR9: Item Detail Page

Mapped to: FR4 (Item Searching), FR8 (Feedback System)

### FR10: Secure Messaging

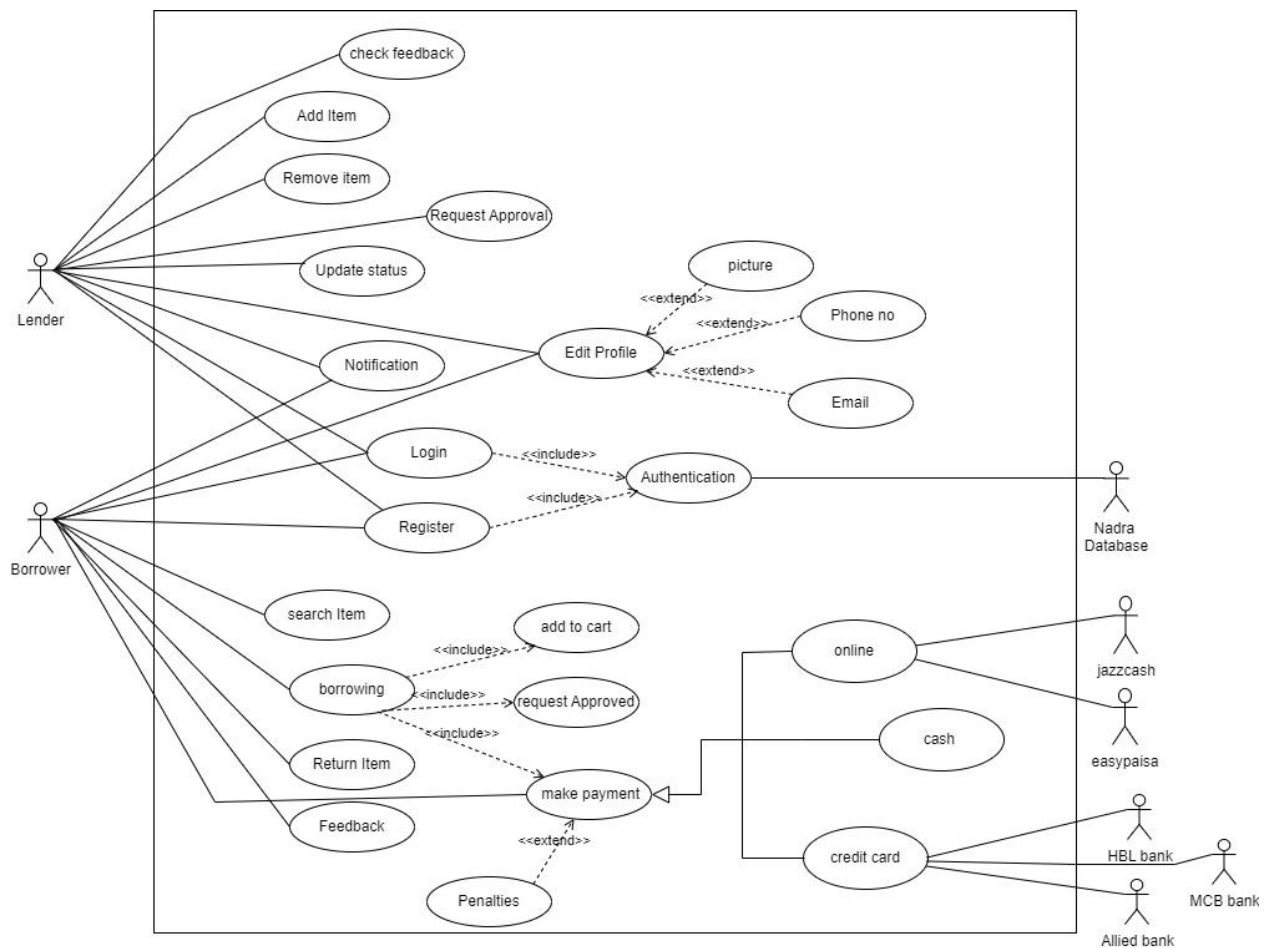
Mapped to: FR2 (Profile Management), FR9 (Item Detail Page)

### FR11: Item Listing

Mapped to: FR1 (User Authentication), FR5 (History Tracking), FR6 (Cart Management)

## 9. Graphical Models of Requirements

### 9.1 Use case Diagram





## 9.2 Use case

|                 |   |
|-----------------|---|
| Use Case        | Login   |
| Description     | The borrower or lender logs into the system.  |
| <b>Actors</b>   | Borrower, Lender  |
| <b>Goals</b>    | Authenticate the user to access the system  |
| Pre-Conditions  | The user must be registered in the system.  |
| Post-Conditions | The user is logged in and can access their account.   |
| Trigger         | The user attempts to log in.  |
| Scenario        | <ul style="list-style-type: none"><li>• The user navigates to the login page.</li><li>• The user enters their credentials (username and password).</li><li>• The system verifies the credentials.</li><li>• If valid, the user is logged in.</li><li>• If invalid, an error message is displayed.</li></ul> |

## 9.3 Use case

|                 |  |
|-----------------|--|
| Use Case        | Register   |
| Description     | A new borrower or lender registers in the system.  |
| <b>Actors</b>   | Borrower, Lender   |
| <b>Goals</b>    | Create a new user account.   |
| Pre-Conditions  | none   |
| Post-Conditions | A new user account is created.   |
| Trigger         | The user selects the option to register.   |
| Scenario        | <ul style="list-style-type: none"><li>• The user navigates to the registration page.</li><li>• The user enters their personal details.</li><li>• The system verifies the information and creates a new account.</li><li>• The user is notified of successful registration.</li></ul> |

## 9.4 Use case

|                 |   |
|-----------------|---|
| Use Case        | Edit Profile  |
| Description     | The user updates their profile information.   |
| <b>Actors</b>   | Borrower, Lender  |
| <b>Goals</b>    | Update user information.  |
| Pre-Conditions  | The user must be logged in.   |
| Post-Conditions | The profile information is updated.   |
| Trigger         | he user selects the option to edit their profile.   |
| Scenario        | <ul style="list-style-type: none"><li>• The user navigates to the edit profile page.</li><li>• The user updates their information (picture, phone number, email).</li></ul> |

|  |   |
|--|---|
|  | <ul style="list-style-type: none"> <li>• The system saves the updated information.</li> </ul> |
|--|---|

## 9.5 Use case

|                 |   |
|-----------------|---|
| Use Case        | Search Item   |
| Description     | The borrower searches for items to borrow.  |
| <b>Actors</b>   | Borrower  |
| <b>Goals</b>    | Find items available for borrowing.   |
| Pre-Conditions  | The user must be logged in.   |
| Post-Conditions | A list of items matching the search criteria is displayed.  |
| Trigger         | he user initiates a search.   |
| Scenario        | <ul style="list-style-type: none"> <li>• The user navigates to the search page.</li> <li>• The user enters search criteria.</li> <li>• The system displays matching items.</li> </ul> |

## 9.6 Use case

|                 |   |
|-----------------|---|
| Use Case        | Add Item  |
| Description     | The lender adds a new item to the system.   |
| <b>Actors</b>   | Lender  |
| <b>Goals</b>    | The lender selects the option to add an item.   |
| Pre-Conditions  | The user must be logged in.   |
| Post-Conditions | The new item is added to the system.  |
| Trigger         | The lender selects the option to add an item.   |
| Scenario        | <ul style="list-style-type: none"> <li>• The lender navigates to the add item page.</li> <li>• The lender enters item details.</li> <li>• The system saves the new item.</li> </ul> |

## 9.7 Use case

|                 |   |
|-----------------|---|
| Use Case        | Remove Item   |
| Description     | The lender removes an item from the system.   |
| <b>Actors</b>   | Lender  |
| <b>Goals</b>    | Make an item unavailable for borrowing.   |
| Pre-Conditions  | The user must be logged in.   |
| Post-Conditions | The item is removed from the system.  |
| Trigger         | The lender selects the option to remove an item.  |
| Scenario        | <ul style="list-style-type: none"> <li>• The lender navigates to the remove item page.</li> <li>• The lender selects the item to remove.</li> <li>• The system removes the item.</li> </ul> |

## 9.8 Use case

|                 |  |
|-----------------|--|
| Use Case        | Borrow Item  |
| Description     | The borrower initiates borrowing an item.  |
| <b>Actors</b>   | Borrower   |
| <b>Goals</b>    | Borrow an item from the lender.  |
| Pre-Conditions  | The user must be logged in.  |
| Post-Conditions | The borrowing request is created.  |
| Trigger         | The borrower selects an item to borrow.  |
| Scenario        | <ul style="list-style-type: none"><li>• The borrower searches for items.</li><li>• The borrower selects an item to borrow.</li><li>• The system creates a borrowing request.</li></ul> |

## 9.9 Use case

|                 |  |
|-----------------|--|
| Use Case        | Return Item  |
| Description     | The borrower returns a borrowed item.  |
| <b>Actors</b>   | Borrower   |
| <b>Goals</b>    | Return a borrowed item to the lender.  |
| Pre-Conditions  | The item must be borrowed.   |
| Post-Conditions | The item is returned, and the borrowing status is updated.   |
| Trigger         | The borrower selects the option to return an item.   |
| Scenario        | <ul style="list-style-type: none"><li>• The borrower navigates to their borrowed items.</li><li>• The borrower selects the item to return.</li><li>• The system updates the item's status to returned.</li></ul> |

## 9.9 Use case

|                 |  |
|-----------------|--|
| Use Case        | Make Payment   |
| Description     | The borrower makes a payment for a borrowed item.  |
| <b>Actors</b>   | Borrower   |
| <b>Goals</b>    | Complete the payment process.  |
| Pre-Conditions  | The item must be borrowed and require payment.   |
| Post-Conditions | The payment is processed.  |
| Trigger         | The borrower initiates the payment process.  |
| Scenario        | <ul style="list-style-type: none"><li>• The borrower selects an item to make a payment.</li><li>• The borrower chooses a payment method (online, cash, credit card).</li><li>• The system processes the payment.</li></ul> |

## 9.10 Use case

|                 |  |
|-----------------|--|
| Use Case        | Request Approval   |
| Description     | The lender approves or rejects a borrowing request.  |
| <b>Actors</b>   | Lender   |
| <b>Goals</b>    | Manage borrowing requests.   |
| Pre-Conditions  | There must be a pending borrowing request.   |
| Post-Conditions | The request is approved or rejected.   |
| Trigger         | A borrowing request is received.   |
| Scenario        | <ul style="list-style-type: none"> <li>• The lender receives a notification of a borrowing request.</li> <li>• The lender reviews the request.</li> <li>• The lender approves or rejects the request.</li> </ul> |

### 9.11 Use case

|                 |   |
|-----------------|---|
| Use Case        | Notification  |
| Description     | The system sends notifications to the user.   |
| <b>Actors</b>   | Borrower, Lender  |
| <b>Goals</b>    | Inform users of important events.   |
| Pre-Conditions  | Action perform  |
| Post-Conditions | The user receives a notification  |
| Trigger         | An event triggers a notification.   |
| Scenario        | <ul style="list-style-type: none"> <li>• An event occurs (e.g., borrowing request, approval, item return).</li> <li>• The system sends a notification to the user.</li> </ul> |

### 9.12 Use case

|                 |  |
|-----------------|--|
| Use Case        | Feedback /Check Feedback   |
| Description     | The borrower leaves feedback for the lender. The lender checks feedback from borrowers.  |
| <b>Actors</b>   | Borrower, Lender   |
| <b>Goals</b>    | Provide feedback & View feedback   |
| Pre-Conditions  | Lending and borrowing  |
| Post-Conditions | Feedback submit  |
| Trigger         | Check and send feedback  |
| Scenario        | <ul style="list-style-type: none"> <li>• The borrower navigates to the feedback page.</li> <li>• The borrower submits feedback.</li> <li>• The system saves the feedback.</li> <li>• The lender navigates to the feedback page.</li> <li>• The lender reviews feedback left by borrowers.</li> </ul> |

### 9.13 Use case

|               |   |
|---------------|---|
| Use Case      | Update Status                                     |
| Description   | The lender updates the status of a borrowed item. |
| <b>Actors</b> | Lender  |

|                 |   |
|-----------------|---|
| <b>Goals</b>    | Manage the status of borrowed items.  |
| Pre-Conditions  | The item must be borrowed and others  |
| Post-Conditions | The item's status is updated  |
| Trigger         | The lender selects the option to update the status.   |
| Scenario        | <ul style="list-style-type: none"> <li>• The lender navigates to the borrowed items page.</li> <li>• The lender updates the status of an item.</li> <li>• The system saves the updated status.</li> </ul> |

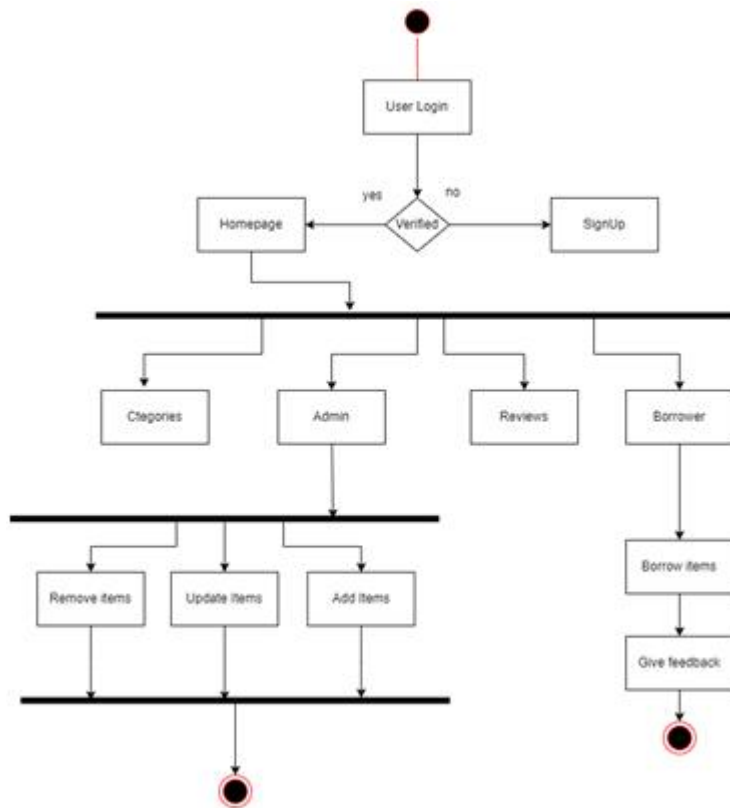
### 9.14 Use case

|                 |   |
|-----------------|---|
| Use Case        | Add to Cart   |
| Description     | The borrower adds an item to their cart.  |
| <b>Actors</b>   | Borrower  |
| <b>Goals</b>    | Prepare to borrow an item.  |
| Pre-Conditions  | The user must be logged in.   |
| Post-Conditions | The item is added to the cart.  |
| Trigger         | borrower selects an item to add to the cart.  |
| Scenario        | <ul style="list-style-type: none"> <li>• The borrower searches for items.</li> <li>• The borrower selects an item to add to the cart.</li> <li>• The system adds the item to the cart.</li> </ul> |

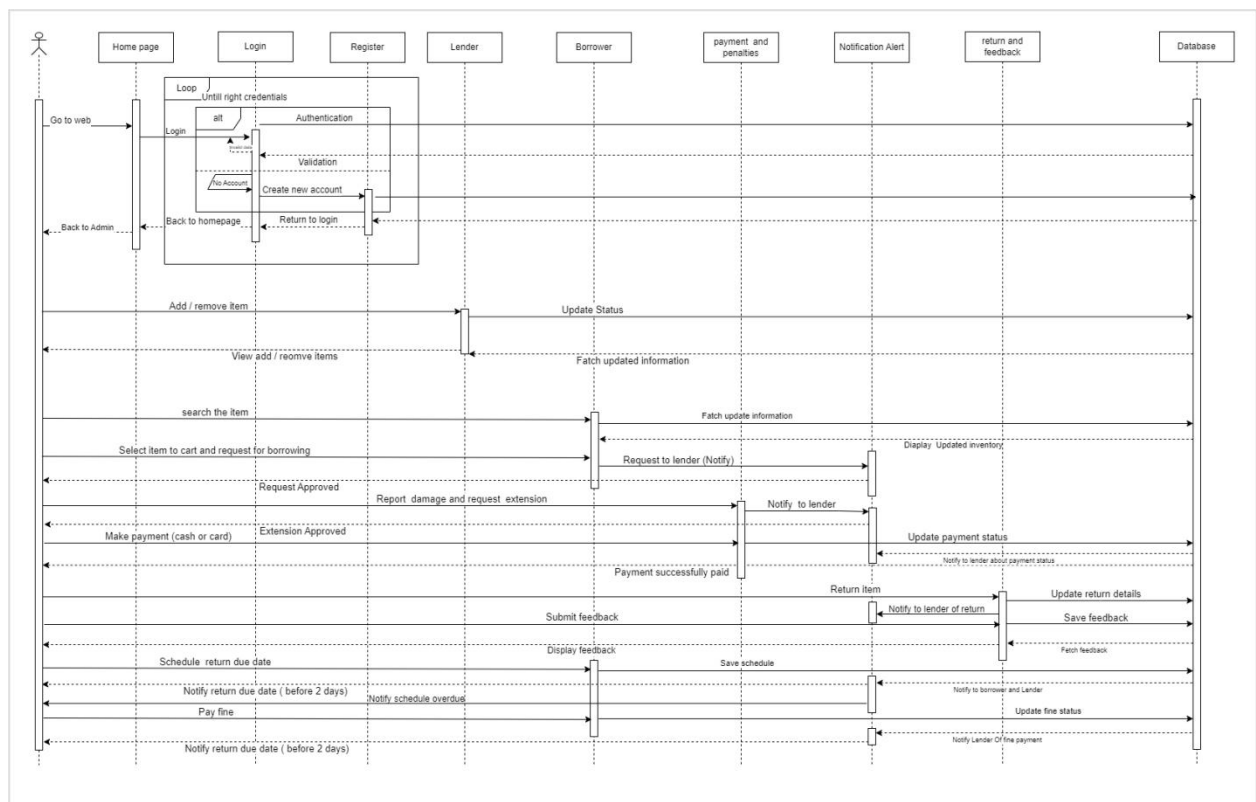
### 9.15 Use case

|                 |  |
|-----------------|--|
| Use Case        | Penalties  |
| Description     | The borrower incurs penalties for late return or damage.   |
| <b>Actors</b>   | Borrower   |
| <b>Goals</b>    | Penalize the borrower for late return or damage.   |
| Pre-Conditions  | The item is returned late or damaged.  |
| Post-Conditions | Penalties are applied.   |
| Trigger         | The item is returned late or damaged.  |
| Scenario        | <ul style="list-style-type: none"> <li>• The borrower returns the item late or damaged.</li> <li>• The system calculates penalties.</li> <li>• The borrower is notified of the penalties.</li> </ul> |

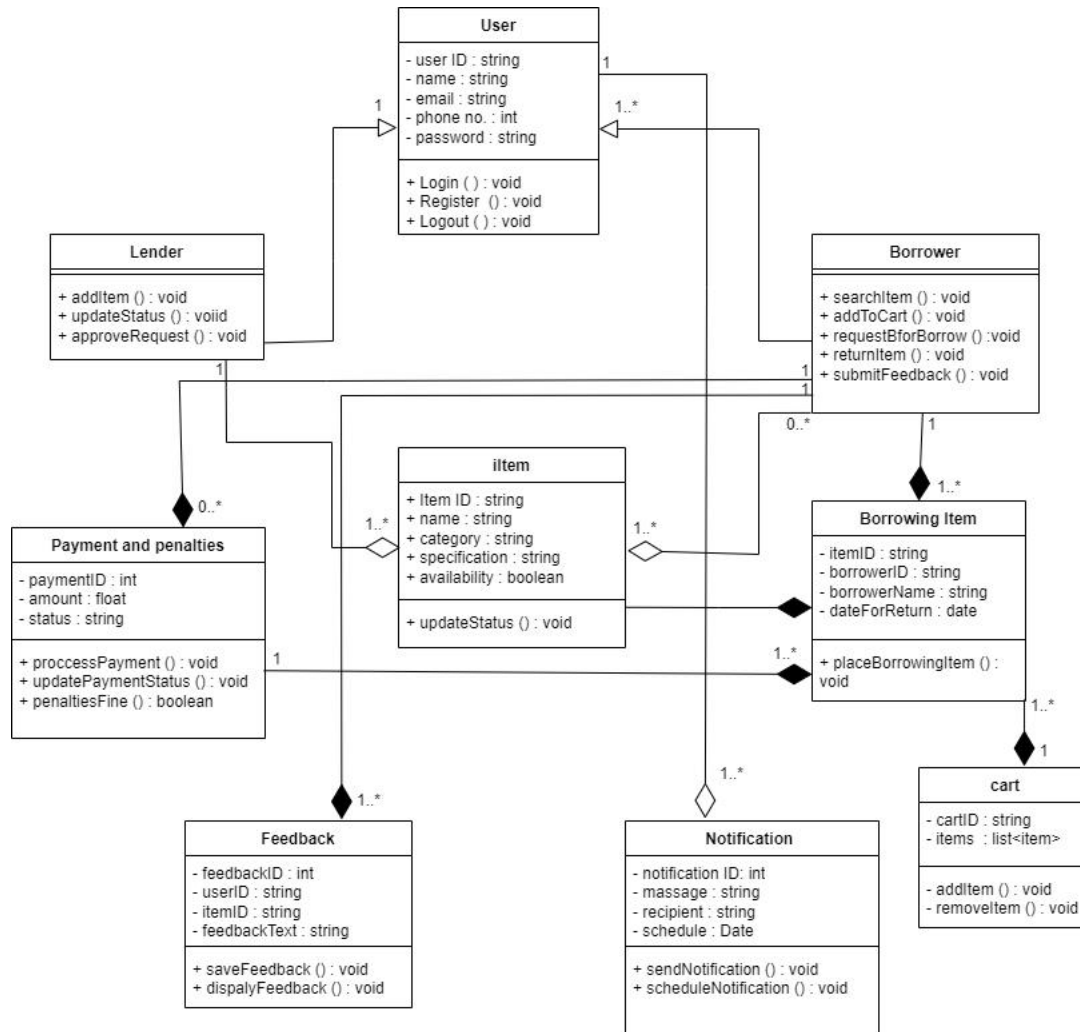
## Activity diagram



## 10. Sequence Diagram



## 11. Class Diagram



## 12. Validation Section

### **Inspection and Peer Reviews:**

Systematic examinations of requirement documents by team members and stakeholders help identify inconsistencies and missing elements. For example, reviewing user registration and login requirements ensures they are complete and clear, catching potential issues early.

### **Review Meetings:**

Regular meetings with stakeholders, such as bi-weekly sessions, allow for feedback and alignment with business goals. Discussing order placement and tracking requirements during these meetings ensures critical functionalities meet user needs.

### **Prototyping:**

Developing prototypes of key components, such as the product catalog and shopping cart, gathers early user feedback and allows for iterative improvements. This ensures the final product is user-friendly and meets expectations.

### **Comprehensive Testing:**

Various types of testing ensure the system meets requirements. Unit testing validates individual components, integration testing checks component interactions, system testing covers end-to-end functionality, and User Acceptance Testing (UAT) confirms the system meets user needs.

### **Formal methods**

ensures the correctness and reliability.

These are based on mathematical models refers to the process of defining system requirements using precise, mathematical language. It involves:

#### **1. Defining Requirements Precisely:**

Clearly specifying what each requirement entails in a detailed, unambiguous manner.

#### **2. Mathematical Rigor:**

Using formal method like VDM, or others to describe behaviors and constraints mathematically. Another formal method is the finite state machines which are commonly presented in tabular form.