IT314: Software Engineering
Group21: Renting System
Lab03: Specifying Functional and Non-Functional Requirements

—---

Use Case Diagram

Actors:

(Primary actors)
- ❖ **Lender**
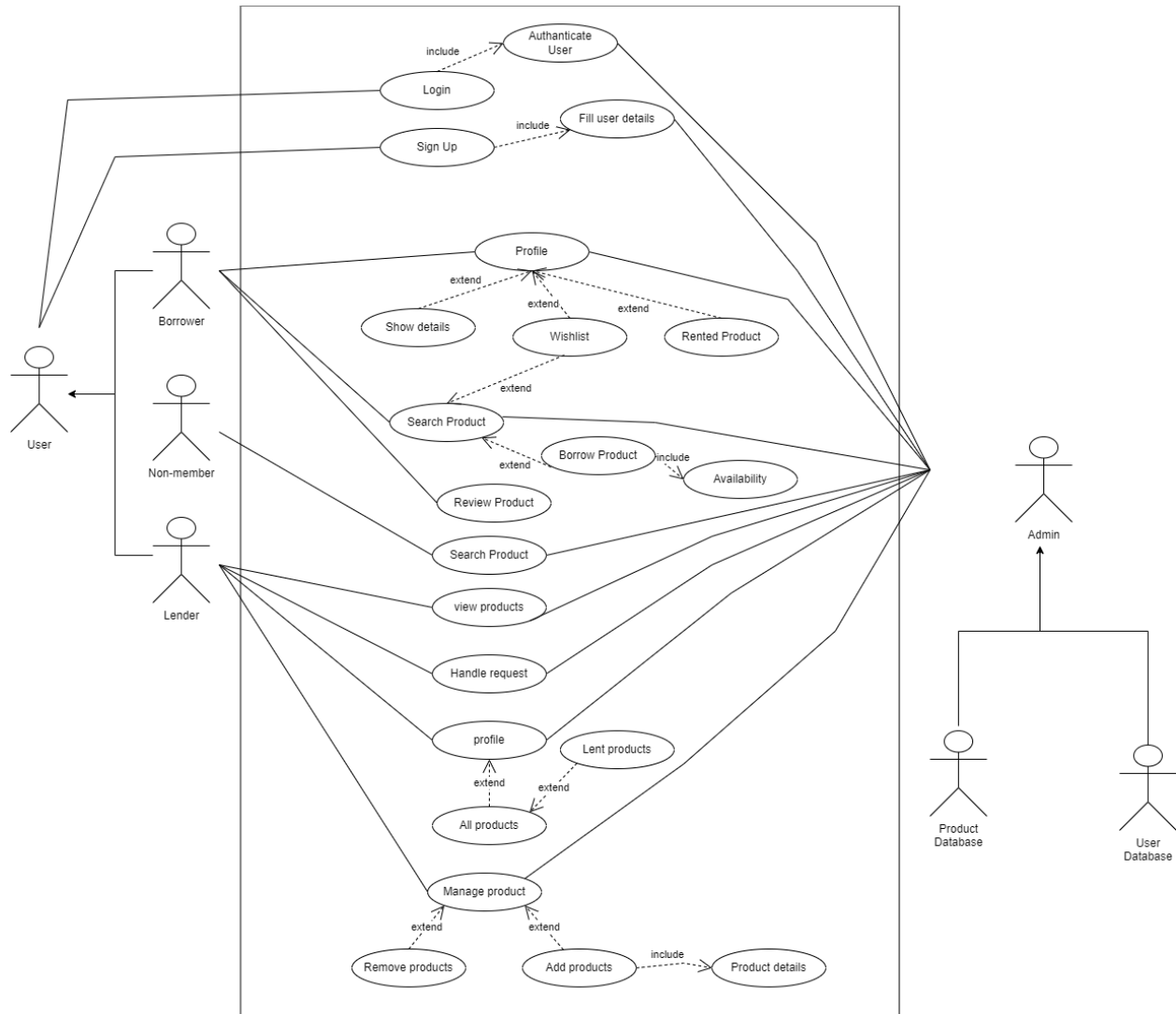- ❖ **Borrower**
- ❖ **Non-member user**

(Secondary Actors)
- ❖ **Admin**
- ❖ **User database**
- ❖ **Products database**

Use cases:
1. **Sign in/Sign up**
2. **Lending a product**
3. **Search a product**
4. **Add a product to wishlist**
5. **Borrow a product**
6. **Payment gateway**

Use cases:-

Title: Sign in/Sign up
Primary Actor: User
Goal: To allow the user to either sign in to an existing account or create a new account to access the system.
Preconditions:
The system is operational.
The user has a device with internet access.

Main Success Scenario:
1. The user opens the system's login page.
2. If the user already has an account, they select the "Sign in" option and enter their username/email and password.
3. The system verifies the user's credentials and logs them in.
4. If the user does not have an account, they select the "Sign up" option and provide the required information, such as their name, email, password, and any other relevant details.
5. The system verifies the user's information and creates a new account for them.
6. The system automatically logs the user into the new account.
7. The user is redirected to the system's main page or the page they were trying to access before being prompted to sign in/up.

Alternate Flows:
3.a If the user enters incorrect login credentials, the system prompts them to try again or reset their password.
5.a If the user provides incomplete or incorrect information during the sign-up process,
   the system prompts them to correct the errors or provide additional information.

Postconditions:
The user is either signed in to the system or has created a new account and is automatically signed in.

Exceptions:
If the system is offline or not functioning properly, the user cannot sign in or sign up

Title: Lending a Product

Primary Actor: Lender

Goal: To allow a lender to lend a product to a borrower.

Preconditions:
- The user(lender) should have an account.
- The user should be logged in into his account.
- The lender owns the product they wish to lend on the website.

Main Success Scenario:
1. Lender opens the webpage to add a product to the website.
2. The lender gives relevant information about the product to the system.
3. After giving all the information, the lender accepts an agreement for lending and the system adds the product to its database.
4. Lender can remove the product after lending the product.

Alternate Flows:
2.a If the lender enters incomplete information about the product the system notifies the lender to enter the full information about the product.

3.a If the lender doesn't agree with the terms the system doesn't allow the product to be added until the lender agrees.
4.a Lender tries to remove products even after the product was borrowed by someone then system notifies the lender that lender can take product back after borrowing period gets over.

Postconditions:

- The lender has lent the product to the borrower.
- The duration of the lending product has been recorded on the website.

Exceptions:

If the system is not functioning or the lender's internet connection is not stable then the product can't be added.

Title: Search a product

Primary Actor: User

Goal: To find and select a desired product for borrowing from a website.

Preconditions:

- The user has a device with internet access.
- The user has an idea about what product he is interested in borrowing.

Main Success Scenario:
The user navigates to the search icon located on the homepage.
The user types the name of the product they want to borrow.
The website displays a list of search results .
The user selects a product he is interested in.
Postconditions:
The user(borrower) has found the product and added it to the wishlist for borrowing.
The user may continue to browse the website to check for other products.
The user completes the process of borrowing.
Alternate Flows:
The user may not be able to find the desired product.
The website doesn't show any results .

Title: Add a product to the wishlist.

Primary Actor: User

Goal: To add currently available/unavailable products to the wishlist to lend it in the future.

Preconditions:

The user has a device with internet access.

The user should be logged in into his account.

The user has an idea about what product he is interested in borrowing.

The product borrower wants is already borrowed by someone else.

Main Success Scenario:

1. The user finds the product.
2. The user adds it into the wishlist.
3. If the product is already borrowed by someone else, the system will ask the user to notify when available.

Alternate Flows:

1.a The user doesn't find any products.

Postconditions:

The user(borrower) has found the product and added it to the wishlist for borrowing.

The user may continue to browse the website to check for other products.

Title: Borrowing a Product

Actors:

Primary Actor: User (the person who needs to borrow a product)

Secondary Actor: Lender (the person who owns the product and is lending it out)

Goal: To allow a lender to lend a product to a borrower.

Preconditions:
- Users should have an account (i.e. signed up) for borrowing any product.
- The user should be logged in into his account.
- The user has a device with internet access.

Main Scenario:
1. User has to select the product which the user wants to borrow.
2. User has to accept terms and conditions to borrow the same product.
3. User has to add the duration according to the lender's timeline for which he/she wants to borrow the product.
4. User get access to lender contact details for further communication (like when to and from where to pick up the borrowed product)
5. User pay for the borrowed product with help of available payment modes
6. When the borrowing period is over, User returns the product to Lender in good condition.
7. Lender inspects the product and confirms that it has been returned in good condition.

Alternative Scenario:

1.a   If the selected product is not available.

1.a.1 Users can add the product to the wishlist.

3.a   Timeline exceeds the lenders' product lending time

3.a.1 User gets an error message to change the borrowing time duration.

6.a.0 User gets notification 2 days prior to the completion of the borrowing
            period.

6.1   User will not return the product after completing the borrowing period

6.a.1 User gets notification that the user has overdue the borrowed product
        and has to pay additional charges according to condition.

6.a.2 Users can give ratings and reviews to borrowed items.

Postconditions:
- The borrower has borrowed the product from the lender.
- The borrower has paid for the product successfully.

Title: Payment gateway

Actors:-

Primary Actor: Borrower

Secondary Actor: Website & Lender

Goal: To complete payment of product by the borrower

Preconditions:

The user is already logged in.

The borrower has found the desired product to borrow.

The product is available for borrowing.

The payment server is functioning properly from both sides.

Main Success Scenario:
1. The user clicks the payment button and leads to the payment process.
2. The user chooses any of the given payment methods.
3. The user enters related details.
4. The user completes the payment process.
5. The user gets the successful payment message.

Alternate Flows:

2.a The user does not select any of the payment methods.

4.a The user does not have sufficient balance.

5.a The transaction failed due to an error.

Postconditions:

The borrower & Lender receives payment messages.

The borrower gets possession of the product.

Non- Functional requirements:

- System should be available 24*7.
  Justification: Users can rent/need a product any time therefore the system should run all the time so that the user can use the system sitting in his/her comfort zone.
- User's sign in data like username and password should not be saved in plain text in the database.
  Justification: As hackers can easily get password and username, we should encrypt it and then store it in the database.
- System should not allow a user to use the system from multiple devices at the same time.
  Justification: If a user forgets to logout from one device then someone else can misuse his account so it is necessary for the user to logout from one device to login from another device.
- The system's performance should be fast and responsive, with minimal downtime and fast page load times.

Justification: To provide users a better experience.

- Interface of the system should be user friendly, such that the user can find out easily all the functionality of the system.
  Justification: Because of the bad interface of the system users will shift to the other platform which is more easy to use and this will directly affect the company's revenue.
- The system should be compatible with a wide range of devices, browsers, and operating systems.
  Justification: we don't know on which device the user is using our system, so no matter what device the user is using, performance of the system and interface should be consistent.
- User's data should be kept confidential and only accessible to authorized users.
  Justification: The system should ensure that the data is protected from unauthorized access.

- The system should backup the data at regular intervals of time to prevent system from power/hardware failures.
  Justification: To prevent data from getting completely lost during any natural calamities/power or hardware failure, the system could backup the data at regular intervals of time.

- System should be able to handle multiple requests from the multiple users as the number of users increases over time.
  Justification: System should be scalable.