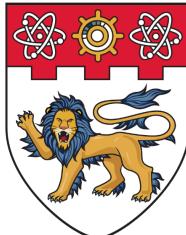


# SC2006 Software Engineering



**NANYANG  
TECHNOLOGICAL  
UNIVERSITY  
SINGAPORE**

## Lab#1 Deliverables

Group Members (Group 45)	
Name	Matriculation No.
Kua Li Min	U2322513H
Hudzaifah Bin Muhammad Taufiq	U2320600F
Cheng Hui Wen	U2322123G
Ng Hoe Ping	U2321991F
Seow Jia Xian Jackson	U2322995F
Chua Ze Ming	U2321797B
Zhang Yuxuan	U2321475L

## Table of Contents

<b>1.0 Project Introduction</b>	<b>3</b>
<b>1.1 Purpose</b>	<b>3</b>
<b>1.2 Intended Audience</b>	<b>3</b>
<b>2.0 Documentation of Functional and Non-functional requirements</b>	<b>4</b>
2.1 Functional Requirements	4
1. User Account Management System	4
2. Review System	6
3. Listing System	6
4. Browsing System	7
5. Payment System	8
2.2 Non-functional requirements	8
<b>3.0 Data Dictionary</b>	<b>10</b>
<b>4.0 Initial Use Case Model</b>	<b>11</b>
4.1 Use Case Diagram	11
4.2 Use Case Description	12
User Account Management System	12
Review System	19
Listing System	22
Browsing System	29
Payment System	39
<b>5.0 UI Mockups</b>	<b>42</b>
User Account Management System	42
Review System	49
Listing System	49
Browsing System	51
Payment System	55

# **1.0 Project Introduction**

## 1.1 Purpose

We want to create a web application to help the community with their daily necessities by looking for rental items or services. This can include but not limited to: rental items like, one-off items that you may not want to buy upfront, for example, power drills for DIY projects; and as a one size fits all service platform to source for services such as plumbing services or dog walking services.

We aim to help both parties, the renter and rentee, by creating a website where they can advertise their services or rental items. The website will also help manage their listings, and also help the filter and sort listing details. We want to create functions that can help minimize the friction for both parties.

## 1.2 Intended Audience

Our intended audience is anyone who wants to rent out items or services or look for these items or services. We are dedicated to supporting low-income families, who may face challenges purchasing items upfront - by providing cheap rental solutions for short-term usage.

We have added negotiable rates for flexibility, allowing users to pay what they feel comfortable. We also recognize that people do not like the inconvenience of travelling often or do not have the capability to do so frequently. So this web application will also take into account the distance from the listing seller as an optional feature. Thus, optimizing productivity whilst reducing time wasted by integrating schedulers into our system.

## **2.0 Documentation of Functional and Non-functional requirements**

### **2.1 Functional Requirements**

1. User Account Management System
  - 1.1. Users must be able to register for a new account
    - 1.1.1. System must check that all requirements are fulfilled before making a new account
      - 1.1.1.1. User must click on the username input text box and enter a username
      - 1.1.1.2. The system will check that the username is not taken
        - 1.1.1.2.1. If it is taken, there will be a red message that says "Username already taken" that will be displayed to the user
      - 1.1.1.3. User must click on the password input text box and enter a password
      - 1.1.1.4. The system will check that the password is at least of 8 characters, with at least 1 number
        - 1.1.1.4.1. If it does not fulfil the password requirements, there will be a red message that says "Password does not fulfil requirements, please have at least 8 characters, with at least 1 number" that will be displayed to the user
      - 1.1.1.5. User must click on the phone number input text box and enter a phone number
      - 1.1.1.6. The system will check that the phone number is all numeric and has exactly 8 numbers
        - 1.1.1.6.1. If it does not fulfil the phone number requirements, there will be a red message that says "Phone number does not fulfil requirements, please have exactly 8 numbers" that will be displayed to the user
    - 1.1.2. System will show a success message to the user when a new user account is created
  - 1.2. User must be able to login to his account
    - 1.2.1. User must click username and password input textbox and enter their username and password and submit
    - 1.2.2. The system must validate the user's username and password
      - 1.2.2.1. If it is not valid, it will show "Invalid" credentials in red, to be displayed to the user
    - 1.2.3. System will show the landing page after the user has logged in
    - 1.2.4. User must be able to reset his password in the case it is forgotten

- 1.2.4.1. System must verify the user details with their username
  - 1.2.4.2. User must click "Forgot password" and enter his password twice, which will follow the requirements in 1.1.1.4
  - 1.2.4.3. System must verify that both passwords are the same, and proceed to change his password
  - 1.2.4.4. System must show a success message to the user when his password is changed
- 1.3. User must be able to view their user account data
- 1.3.1. User must click view user data
  - 1.3.2. System will show the user their username, phone, email, password, rating, listings, rentals, reviews, and bio.
- 1.4. User must be able to manage his account details
- 1.4.1. User must be able to update user details
    - 1.4.1.1. User must click on their username and enter their new username
    - 1.4.1.2. The system will check that the username is not taken
      - 1.4.1.2.1. If it is taken, there will be a red message that says "Username already taken" that will be displayed to the user
    - 1.4.1.3. System must show a success message when the username of the user is updated
    - 1.4.1.4. User must click on their phone number and enter their new phone number
    - 1.4.1.5. The system will check that the phone number is not taken
      - 1.4.1.5.1. If it is taken, there will be a red message that says "Phone number already taken" that will be displayed to the user
    - 1.4.1.6. System must show a success message when the phone number of the user is updated
    - 1.4.1.7. User must click on their email address and enter their new email address
    - 1.4.1.8. The system will check that the email address is not taken
      - 1.4.1.8.1. If it is taken, there will be a red message that says "Email address already taken" that will be displayed to the user
    - 1.4.1.9. System must show a success message when the email address of the user is updated
  - 1.4.2. User must be able to change his password
    - 1.4.2.1. User must click on the password input text box and enter his password twice, which will follow the requirements in 1.1.1.4
    - 1.4.2.2. System must verify that both passwords are the same, and proceed to change his password
    - 1.4.2.3. System must show a success message to the user when his password is changed

### 1.4.3.

## 2. Review System

### 2.1. User must be able to submit a review

- 2.1.1. User must click on "Submit" review on the product listing or service that has been completed
- 2.1.2. User must click on a rating between 1 to 5 stars
- 2.1.3. The user can optionally upload a photo for the review
  - 2.1.3.1. The user must click on "Upload Photo" and upload a photo
  - 2.1.3.2. System must verify that this is a valid photo
- 2.1.4. The user must click on the text box to enter a review of the product listing or service
- 2.1.5. The user must then click "Submit Review"
- 2.1.6. The system must verify that the user has submitted a rating between 1 to 5 stars
- 2.1.7. The system will verify that it has received the review, and show a success message to the user

## 3. Listing System

### 3.1. User must be able to manage his listings

- 3.1.1. User must click on "Create Listing" button on the home page to create a listing
  - 3.1.1.1. The user must enter the details of the items he/she wants to list in the text box
  - 3.1.1.2. The user may or may not upload a photo of their choice
  - 3.1.1.3. The user must enter the available date and time for item collection
  - 3.1.1.4. System will register the item and send the details of the listing to the database
- 3.1.2. User may update the details of their listed items under the "Your Listings" page which is found under account information tab
  - 3.1.2.1. The user must click on the box of the listed item that they want to update
  - 3.1.2.2. User must re-enter/change the details of the listed item if need be
  - 3.1.2.3. User must press the "Submit" button to make the necessary changes
  - 3.1.2.4. System will send the updated details of the listed item to the database to overwrite the old details
- 3.1.3. User may delete their item listings under the "Your Listings" page which is found under account information tab
  - 3.1.3.1. The user must click on the box of the listed item that they want to delete

- 3.1.3.2. User must press the "Delete" button/Trash button to delete the selected item
- 3.1.3.3. The system will display a pop-up box to verify if user wants to delete the selected item
  - 3.1.3.3.1. System will redirect the user to the previous page the user was on if the user clicks on "No" button in the pop-up box
- 3.1.3.4. The system will delete the selected item if the user clicks on the "Yes" button and remove the details from the database

#### 4. Browsing System

- 4.1. User must be able to search listing
  - 4.1.1. User must be able to type in keywords to find items in the search bar
  - 4.1.2. System will redirect the user to the landing page with all relevant items
- 4.2. User must be able to filter listings by applying various filters to find their desired items.
  - 4.2.1. User must be able select one or more categories by ticking the relevant category boxes.
  - 4.2.2. System will redirect users to show items that fall under the categories that user chooses
  - 4.2.3. User must be able to key in price range to filter the items that fall within this range
  - 4.2.4. System will redirect users to show items that fall within this price range
  - 4.2.5. System must have both ascending and descending price orders for users to choose from
  - 4.2.6. System must have various time-measuring units: by hours, by day, by week for users to choose from
  - 4.2.7. System will display the items in categories, price ranges, order of display in terms of different time units once the user make their choices
- 4.3. User must be able to open listing details when clicking on the listing profiles
  - 4.3.1. System must display seller's overall rating in terms of a star rating out of 5 based on customer reviews
  - 4.3.2. System must display past customer reviews on various other listings of that particular seller
  - 4.3.3. User will be able to choose available time slots to arrange a meeting with the seller
  - 4.3.4. User must be able to manage offers
    - 4.3.4.1. User must be able to make an offer
    - 4.3.4.2. User must be able to accept offer if they are satisfied with the
    - 4.3.4.3. User must be able to reject offer if they are not satisfied with the

- 4.3.5. User must be able to review report listing if they realise the listing is a scam
  - 4.3.5.1. System must have an option to report listing
  - 4.3.5.2. System will redirect the user if they want to report listing
  - 4.3.5.3. User must be able to key in the text messages in the description box to describe the suspected scam case they encountered.
  - 4.3.5.4. System will review the report and make necessary refunds arrangement if the report is deemed as truthful
  - 4.3.5.5. System must permanently disable the seller to list items if more than 3 scam reporting cases made in a single month are found to be truthful

## 5. Payment System

- 5.1. User must be able to make payment for selected item via PayPal
  - 5.1.1. User must enter their bank card details on the payment page
  - 5.1.2. User must click on the payment button
  - 5.1.3. System will redirect the user to PayPal for payment processing
  - 5.1.4. PayPal will deduct the balance from the user's account if there is sufficient balance
    - 5.1.4.1. Error message will pop-up if user has insufficient balance in their account
    - 5.1.4.2. User will be redirected back to product page with a pop-up message indicating that the purchase has failed
  - 5.1.5. System will display a message informing users that the purchase has been successful
  - 5.1.6. PayPal will generate a receipt/invoice containing details of the purchase
  - 5.1.7. PayPal will send the receipt/invoice to the user's email
  - 5.1.8. User may also check their purchase history under the account information tab

## 2.2 Non-functional requirements

1. Performance
  - 1.1. The web application shall take within 3 seconds to load in and exit
  - 1.2. The account registration and login page shall take within 3 seconds to load
  - 1.3. Verification for the validity of username, password, phone number shall take within 3 seconds
  - 1.4. Any success message should take 1 second to display
  - 1.5. New account must be updated in the system within 3 seconds
  - 1.6. Any account information updates shall be reflected in the system within 1 seconds

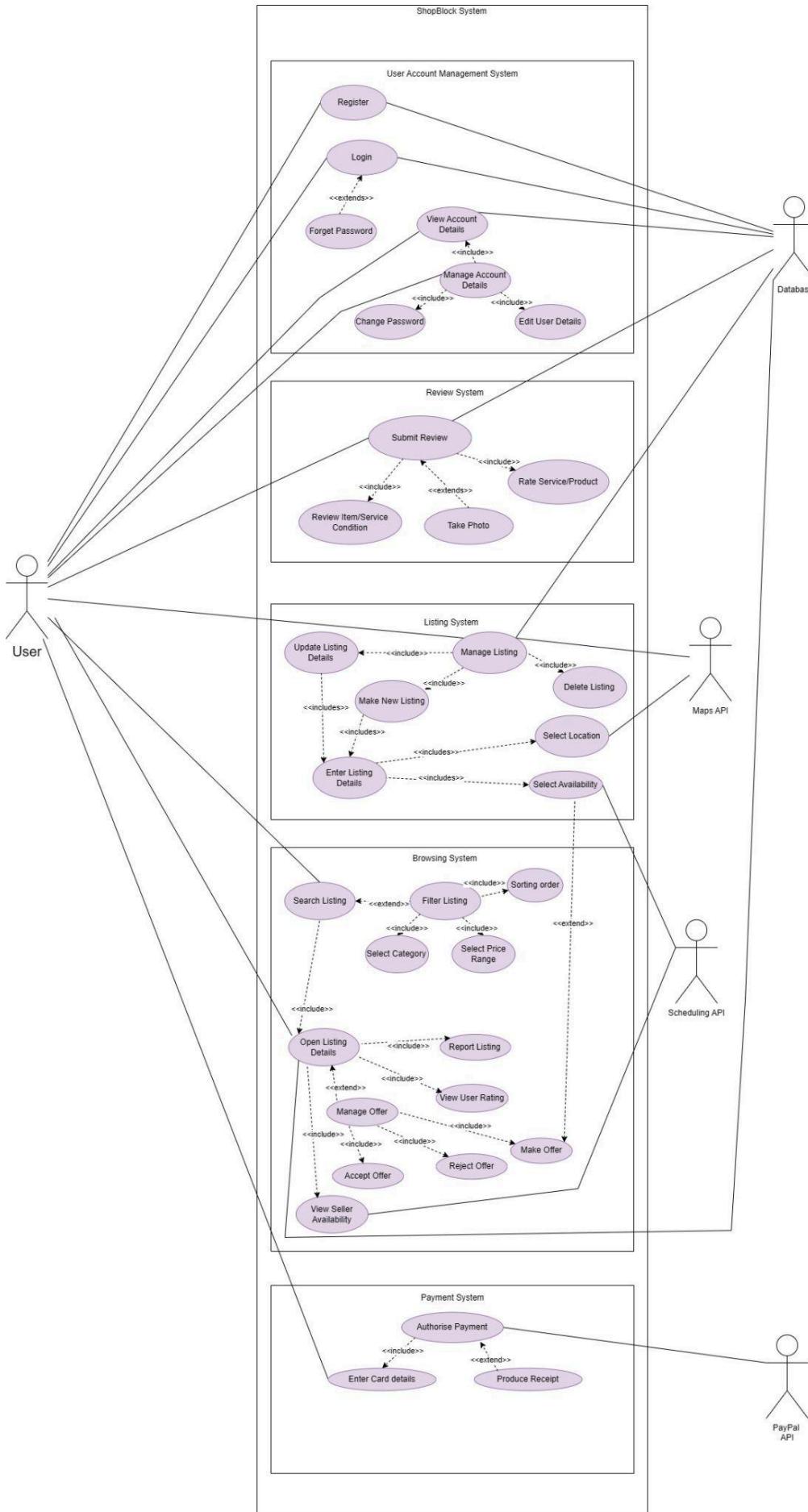
- 1.7. System shall redirect the user to landing page within 1 seconds
  - 1.8. A photo should be uploaded within 5 seconds in the review
  - 1.9. Any review submission shall be stored in the system within 3 seconds
2. Safety Requirements
  - 2.1. User must be able to launch a report of any suspicious listings/profiles to the system
  - 2.2. System to display moving animation upon payment receipts to prevent any cases of fraud between renters and rentees
  - 2.3. Users to agree on a clear rental agreement that outlines the terms and conditions for each transaction
  - 2.4. Developers of the team will perform consistent app maintenance
  - 2.5. Adequate testing will be done by the developers
3. Security Requirements (zeming)
  - 3.1. User must be logged in to make an offer and to schedule an appointment
  - 3.2. The web page will adhere to web security standards and use an updated security stack
  - 3.3. System will use tried and tested industry standards of authenticating via JWT (Json Web Tokens) to authorise logged in users

## 3.0 Data Dictionary

Term	Definition
User	A person who has a registered account and is using the services provided by the application.
Account	Profile of registered user of the application containing their details such as their username, phone number, rating etc...
Job	A service that is rented out by a Renter (see below)
Item	An object that is rented out by a Renter (see below)
Renter	A user on the platform renting our their goods/services
Rentee	A user on the platform looking to rent goods/ services
Listing	An entry within the application posted by a renter. Each listing contains information about the good/service including its description, rental price, availability, location, and any associated media such as images or videos.
Availability	Specific time periods that a renter designates in their listings for when their goods or services are accessible for rental.
Offer	A proposal submitted by a rentee to the renter within the application. The offer includes a rental price and rental period that the rentee wishes to propose.
Rating	A numerical score provided by a user (either a rentee or renter) to assess the quality of their experience during a rental transaction. Ratings are given on a scale of 1 to 5 stars with 1 being the lowest rating and 5 being the highest..
Review	A written evaluation provided by renter/rentee after a rental transaction has been completed. The review allows both parties to share their experiences and feedback about the rental process, including the quality of the good or service, communication, and overall satisfaction with the interaction.

## **4.0 Initial Use Case Model**

### 4.1 Use Case Diagram



## 4.2 Use Case Description

Key:

Frequency 1-5, 1 = Low 5 = Very Frequent

User Account Management System

Use Case ID:	<b>USER_UC_1</b>		
Use Case Name:	<b>Register</b>		
Created By:	<b>Ng Hoe Ping</b>	Last Updated By:	<b>Ng Hoe Ping</b>
Date Created:	<b>1/9/2024</b>	Date Last Updated:	<b>1/9/2024</b>

Actor:	<b>User</b>
Description:	Allows a non-User to create and register a new account.
Preconditions:	<ul style="list-style-type: none"> <li><b>1. User must not be an existing registered user in the system.</b></li> <li><b>2. User must have the necessary personal information to create an account.</b></li> </ul>
Postconditions:	<ul style="list-style-type: none"> <li><b>1. Account details are stored in the database to be retrieved when prompted</b></li> </ul>
Priority:	
Frequency of Use:	<b>1</b>
Flow of Events:	<ul style="list-style-type: none"> <li><b>1. User inputs required information: Username, Phone Number, Password.</b></li> <li><b>2. System validates the information with existing information in the database.</b></li> <li><b>3. System creates the account if no conflicting username/phone number is found.</b></li> <li><b>4. System updates account information in database.</b></li> <li><b>5. System directs the user to the main page.</b></li> </ul>
Alternative Flows:	<p><b>AF-S1-a. User account already exists: Phone number taken.</b></p> <ul style="list-style-type: none"> <li><b>1. System displays “An account with this phone number exists!”</b></li> <li><b>2. System prompts user to “Create New Account”, “Log In” or “Forget Password”</b></li> <li><b>3. If “Create New Account”, then continue from main flow step 1.</b></li> <li><b>4. If “Log In”, then system directs to log in page to continue from Login (USER_UC_2)</b></li> <li><b>5. If “Forget Password”, then system will direct to “forget password page” (USER_UC_3)</b></li> </ul> <p><b>AF-S1-b. Username already taken</b></p>

	<ol style="list-style-type: none"> <li>1. System displays “Username already taken”</li> <li>2. System prompts user to input a different username.</li> <li>3. Continue from main flow step 1.</li> </ol> <p><b>AF-S1-c. Password does not meet requirements.</b></p> <ol style="list-style-type: none"> <li>1. System displays “Password does not meet requirements!”</li> <li>2. System displays “Please input a password of at least 1 special character, mixed case and at least 8 characters long” message.</li> <li>3. Continue from main flow step 1.</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>USER_UC_2</b>		
Use Case Name:	<b>Login</b>		
Created By:	<b>Ng Hoe Ping</b>	Last Updated By:	<b>Ng Hoe Ping</b>
Date Created:	<b>1/9/2024</b>	Date Last Updated:	<b>1/9/2024</b>

Actor:	<b>User</b>
Description:	Allows an existing registered user to log in to the website with their personal information.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be an existing registered user in the database.</li> <li>2. User must have the necessary personal information to log in to the account.</li> </ol>
Postconditions:	
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. User inputs required information: username and password.</li> <li>2. System validates the information with existing information in the database.</li> </ol>

	<p>3. System logs in to the account if the inputted information matches the database.</p> <p>4. System directs the user to the main page.</p>
Alternative Flows:	<p><b>AF-S1-a. Username found but incorrect password</b></p> <ol style="list-style-type: none"> <li>1. System displays “Incorrect Password! Please try again”</li> <li>2. System prompts the user to input the password again.</li> <li>3. Continue to Main Flow Step 1.</li> <li>4. After three unsuccessful attempts, the system prompts the user with “Forget Password?”.</li> <li>5. If User chooses to “Forget Password” User will be directed to the “Forget Password” (USER_UC_3) Page to reset their password</li> </ol> <p><b>AF-S1-b. Ignoring the “Forget Password” prompt</b></p> <ol style="list-style-type: none"> <li>1. System displays “Incorrect Password! Please try again”</li> <li>2. System prompts the user to input the password again.</li> <li>3. Continue to Main Flow Step 1.</li> <li>4. After three unsuccessful attempts, the system prompts the user with “Your account will be locked after two more attempts. Forget password?”.</li> <li>5. If User chooses to ignore the prompt, the user will be allowed to re-attempt to log in to the account.</li> <li>6. If User successfully input the correct corresponding password, the user will be directed to the main page.</li> <li>7. If User has two more failed attempts (total of 5 unsuccessful tries), the user will be locked out from the account.</li> <li>8. The system prompts the user with “Your account has been temporarily locked for 5 mins. Please try again later or reset your password”.</li> <li>9. If User chooses to “Forget Password” User will be directed to the “Forget Password” (USER_UC_3) Page to reset their password</li> <li>10. User can choose to try again later in 5 mins, continue from Main Flow Step 1</li> </ol> <p><b>AF-S1-c. Username not found!</b></p> <ol style="list-style-type: none"> <li>1. System displays “Username not found”</li> <li>2. System prompts the user to input username again.</li> <li>3. Continue from Main Flow Step 1</li> </ol>
Exceptions:	
Includes:	
Extends:	Forget Password (USER_UC_3)
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>USER_UC_3</b>		
Use Case Name:	<b>Forget Password</b>		
Created By:	<b>Ng Hoe Ping</b>	Last Updated By:	<b>Ng Hoe Ping</b>
Date Created:	<b>1/9/2024</b>	Date Last Updated:	<b>1/9/2024</b>

Actor:	<b>User</b>
Description:	Allows an existing registered user to edit their existing password in the database.
Preconditions:	<ol style="list-style-type: none"> <li><b>User must be an existing registered user in the database.</b></li> <li><b>User must have the corresponding email to the registered account.</b></li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li><b>Account details are stored in the database to be retrieved when prompted</b></li> </ol>
Priority:	
Frequency of Use:	<b>1</b>
Flow of Events:	<ol style="list-style-type: none"> <li><b>User inputs required information: Username and Phone Number</b></li> <li><b>System validates the information with existing information in the database.</b></li> <li><b>System prompts user to input new password.</b></li> <li><b>User successfully resets password.</b></li> <li><b>User is directed to the Login page (USER_UC_1)</b></li> </ol>
Alternative Flows:	<p><b>AF-S1-a. Password does not meet requirements.</b></p> <ol style="list-style-type: none"> <li><b>Starting from main flow step 3.</b></li> <li><b>System displays “Please input a password of at least 1 special character, mixed case and at least 8 characters long” message.</b></li> <li><b>User inputs valid password and is direct to the Login Page (USER_UC_1)</b></li> </ol> <p><b>AF-S1-b. Username not found!</b></p> <ol style="list-style-type: none"> <li><b>System displays “Username not found”</b></li> <li><b>System prompts the user to input username again.</b></li> <li><b>Continue from Main Flow Step 1</b></li> </ol> <p><b>AF-S1-c. Phone number does not match!</b></p> <ol style="list-style-type: none"> <li><b>System displays “Phone number does not match”</b></li> <li><b>System prompts the user to input phone number again.</b></li> <li><b>Continue from Main Flow Step 1</b></li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	

Assumptions:	
Notes and Issues:	

Use Case ID:	<b>USER_UC_4</b>		
Use Case Name:	<b>View Account Details</b>		
Created By:	<b>Ng Hoe Ping</b>	Last Updated By:	<b>Ng Hoe Ping</b>
Date Created:	<b>2/9/2024</b>	Date Last Updated:	<b>2/9/2024</b>

Actor:	<b>User</b>
Description:	<b>Allows a user to view their own individual account details</b>
Preconditions:	<b>1. User must be logged in</b>
Postconditions:	
Priority:	
Frequency of Use:	<b>2</b>
Flow of Events:	<ul style="list-style-type: none"> <li><b>1. User selects the “View Account Details” option from the main page.</b></li> <li><b>2. Users are able to see their username, listing details and reviews and phone number.</b></li> <li><b>3. User can choose “Manage Account Details” (USER_UC_5), or choose to go back to the main page.</b></li> </ul>
Alternative Flows:	
Exceptions:	
Includes:	<b>Manage Account Details (USER_UC_5)</b>
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>USER_UC_5</b>		
Use Case Name:	<b>MANAGE ACCOUNT DETAILS</b>		
Created By:	<b>Ng Hoe Ping</b>	Last Updated By:	<b>Ng Hoe Ping</b>
Date Created:	<b>2/9/2024</b>	Date Last Updated:	<b>2/9/2024</b>

Actor:	<b>User</b>
Description:	Allows a user to read/modify their own individual account details.

Preconditions:	<b>1. User must be logged in</b>
Postconditions:	
Priority:	
Frequency of Use:	<b>2</b>
Flow of Events:	<b>1. User can choose to edit their individual data by accessing “Change Password” (USER_UC6) or “Edit User Data” (USER_UC_7)</b>
Alternative Flows:	
Exceptions:	
Includes:	Change Password ( <b>USER_UC_6</b> ) Edit User Data ( <b>USER_UC_7</b> )
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>USER_UC_6</b>		
Use Case Name:	<b>Change Password</b>		
Created By:	<b>Ng Hoe Ping</b>	Last Updated By:	<b>Ng Hoe Ping</b>
Date Created:	<b>2/9/2024</b>	Date Last Updated:	<b>2/9/2024</b>

Actor:	User
Description:	Allows a user to change their existing password
Preconditions:	<b>1. User must be logged in</b>
Postconditions:	<b>1. Account details are stored in the database to be retrieved when prompted</b>
Priority:	
Frequency of Use:	<b>1</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. System prompts the user to input current account password.</li> <li>2. System validates the password with existing information in the database.</li> <li>3. System then prompts the user to input a new password twice.</li> <li>4. User inputs a new password twice</li> <li>5. System prompts “Password successfully changed!”</li> <li>6. System updates the new password, overwriting the existing information in the database.</li> </ol>
Alternative Flows:	AF-S1-a. Current Password does not match

	<ol style="list-style-type: none"> <li>1. Starting from main flow step 2.</li> <li>2. System prompts “Password does not match!”</li> <li>3. Continue from main flow step 1.</li> </ol> <p><b>AF-S1-b. Password does not meet requirements.</b></p> <ol style="list-style-type: none"> <li>1. Starting from main flow step 4.</li> <li>2. System displays “Please input a password of at least 1 special character, mixed case and at least 8 characters long” message.</li> <li>3. User inputs a new password</li> <li>4. System prompts “Password successfully changed!”</li> <li>5. System updates the new password, overwriting the existing information in the database</li> </ol> <p><b>AF-S1-b. “Confirm Password does not match new password”</b></p> <ol style="list-style-type: none"> <li>3. Starting from main flow step 4.</li> <li>4. System displays “Your new password does not match!” message.</li> <li>6. User re-inputs the password.</li> <li>7. System prompts “Password successfully changed!”</li> <li>8. System updates the new password, overwriting the existing information in the database.</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>USER_UC_7</b>		
Use Case Name:	<b>Edit User Details</b>		
Created By:	<b>Ng Hoe Ping</b>	Last Updated By:	<b>Ng Hoe Ping</b>
Date Created:	<b>2/9/2024</b>	Date Last Updated:	<b>2/9/2024</b>

Actor:	<b>User</b>
Description:	Allows a user to read/modify their own individual account details.
Preconditions:	<b>1. User must be logged in</b>
Postconditions:	
Priority:	
Frequency of Use:	<b>1</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. Users can choose to edit their username, phone number, and profile picture.</li> <li>2. After User edit and confirm changes</li> </ol>

	<p>3. System will prompt “Changes saved!”</p> <p>4. System will direct back to “Manage Account Details” (USER_UC_5)</p> <p><b>AF-S1-a. Username has been taken</b></p> <ol style="list-style-type: none"> <li>1. User chooses a new username and confirm changes</li> <li>2. System displays “Username already taken”</li> <li>3. System prompts User to input a different username.</li> <li>4. Continue from main flow step 2.</li> </ol> <p><b>AF-S1-b. Phone number has been taken</b></p> <ol style="list-style-type: none"> <li>1. User chooses a new phone number and confirms changes.</li> <li>2. System displays “An account with this phone number exists!”</li> <li>3. System prompts User to input a new phone number</li> <li>4. Continue from main flow step 2.</li> </ol>
Alternative Flows:	
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

## Review System

Use Case ID:	<b>REVIEW_UC_1</b>		
Use Case Name:	<b>Submit Review</b>		
Created By:	<b>Li Min</b>	Last Updated By:	<b>Li Min</b>
Date Created:	<b>31/8/24</b>	Date Last Updated:	<b>31/8/24</b>

Actor:	<b>User, Database API</b>
Description:	<b>This use case allows a user to submit a review for a service or product. The user provides feedback on the item/service condition and optionally rates the service/item. The user may also choose to take a photo as part of the review.</b>
Preconditions:	<b>1. User must be logged in 2. User must have had rented the product</b>
Postconditions:	<b>1. Feedback on the item/service condition is included in the review</b>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<b>1. The user selects the option to submit a review</b>

	<ol style="list-style-type: none"> <li>2. The system prompts the user to provide a rating out of 5</li> <li>3. The system prompts the user to write some comments</li> <li>4. The system prompts the user to optionally take or upload a photo</li> <li>5. The user submits the review.</li> <li>6. The system saves the review to the database and confirms the submission to the user.</li> </ol>
Alternative Flows:	<p><b>AF-S1-a. User leaves ratings blank</b></p> <ol style="list-style-type: none"> <li>1. System displays “Please select a rating!”</li> <li>2. Continue from main flow step 1</li> </ol> <p><b>AF-S1-b. User leaves comments blank</b></p> <ol style="list-style-type: none"> <li>1. System displays “Please enter some comments!”</li> <li>2. Continue from main flow step 3</li> </ol>
Exceptions:	
Includes:	<b>Review Item/Service Condition (REVIEW_UC_2)</b> <b>Rate Service/Product (REVIEW_UC_3)</b>
Extends:	<b>Take Photo (REVIEW_UC_4)</b>
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>REVIEW_UC_2</b>		
Use Case Name:	<b>Review Item/Service Condition</b>		
Created By:	<b>Li Min</b>	Last Updated By:	<b>Li Min</b>
Date Created:	<b>31/8/24</b>	Date Last Updated:	<b>31/8/24</b>

Actor:	User
Description:	This use case allows the user to provide feedback on the condition of the item or service being reviewed. This is part of the broader "Submit Review" use case.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in</li> <li>2. User must have had rented the product</li> </ol>
Postconditions:	1. Feedback on the item/service condition is included in the review
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. The user is prompted to describe the condition of the item/service.</li> <li>2. The user provides feedback on the condition</li> </ol>
Alternative Flows:	

Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>REVIEW_UC_3</b>		
Use Case Name:	<b>Rate Service/Product</b>		
Created By:	<b>Li Min</b>	Last Updated By:	<b>Li Min</b>
Date Created:	<b>31/8/24</b>	Date Last Updated:	<b>31/8/24</b>

Actor:	<b>User</b>
Description:	<b>This use case allows the user to rate the service or product out of 5 stars. This is included as part of the broader "Submit Review" use case.</b>
Preconditions:	<b>1. User must be logged in 2. User must have had rented the product</b>
Postconditions:	<b>1. Review is stored in the system and available for others to view</b>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<b>1. The system prompts the user to provide a rating out of 5. 2. The user selects a rating.</b>
Alternative Flows:	
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>REVIEW_UC_4</b>		
Use Case Name:	<b>Take Photo</b>		
Created By:	<b>Li Min</b>	Last Updated By:	<b>Li Min</b>
Date Created:	<b>31/8/24</b>	Date Last Updated:	<b>31/8/24</b>

Actor:	User
Description:	<b>This use case allows the user to optionally take or upload a photo as part of the review. This extends the "Submit Review" use case.</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in</li> <li>2. User must have had rented the product</li> </ol>
Postconditions:	1. The photo is included as part of the review.
Priority:	
Frequency of Use:	2
Flow of Events:	<ol style="list-style-type: none"> <li>1. The system prompts the user to take or upload a photo.</li> <li>2. The user takes or selects a photo from their device.</li> <li>3. The system includes the photo in the review.</li> </ol>
Alternative Flows:	<b>AF-S1-a. Photo fails to upload</b> <ol style="list-style-type: none"> <li>1. System displays “Photo failed to upload. Please try again.”</li> <li>2. Continue from main flow step 1</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

## Listing System

Use Case ID:	LIST_UC_1		
Use Case Name:	<b>Manage Listing</b>		
Created By:	Hui Wen	Last Updated By:	Hui Wen
Date Created:	1/9/24	Date Last Updated:	1/9/24

Actor:	User, Maps API, Database	
Description:	<b>Allows user to create, update, or delete a listing.</b>	
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. User must enter all the necessary information.</li> </ol>	
Postconditions:	<ol style="list-style-type: none"> <li>1. The listing is successfully created, updated, or deleted.</li> <li>2. Any changes made to a listing are reflected in the app and visible to all users.</li> </ol>	

	<b>3. The system ensures the listing complies with all necessary validation rules before saving.</b>
Priority:	
Frequency of Use:	<b>4</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. The renter navigates to the "Manage Listings" section in the app.</li> <li>2. The renter selects an option to create a new listing, update an existing listing, or delete a listing.</li> <li>3. The renter provides or modifies details such as the title, description, rental price, availability, location, and images.</li> <li>4. The renter submits the changes.</li> <li>5. The system saves the changes and updates the listing information.</li> <li>6. The renter receives a confirmation that the listing has been successfully managed.</li> </ol>
Alternative Flows:	<p><b>AF-S1-a. User cancels the operation before submission</b></p> <ol style="list-style-type: none"> <li>1. System exits listing page and return to home page, and no changes will be saved.</li> </ol> <p><b>AF-S1-b System encounters an error upon submission(e.g., database connectivity issues)</b></p> <ol style="list-style-type: none"> <li>1. Notifies the user with a prompt telling them to resubmit.</li> </ol>
Exceptions:	
Includes:	Update Listing Details Make New Listing Delete Listing
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>LIST_UC_2</b>		
Use Case Name:	<b>Make New Listing</b>		
Created By:	<b>Hui Wen</b>	Last Updated By:	<b>Hui Wen</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	<b>User</b>
Description:	<b>Allows user to create a new listing to showcase the item/service they want to rent out.</b>

Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. User must enter all the necessary information.</li> </ol>
Postconditions:	1. Listing is stored in the system and available for other users to view
Priority:	
Frequency of Use:	3
Flow of Events:	<ol style="list-style-type: none"> <li>1. User selects create listing option</li> <li>2. System prompts user to input details (Covered in Enter Listing Details)</li> <li>3. User submits listing.</li> <li>4. System saves the listing and displays it on the user's profile. The listing is made available for other users to view.</li> </ol>
Alternative Flows:	<p><b>AF-S1-a.</b> User leaves Title/description/rental price/location blank and tries to submit listing</p> <ol style="list-style-type: none"> <li>1. System displays "Must include Title/description/rental price/location "</li> <li>2. continue from main flow step 2</li> </ol> <p><b>AF-S1-b.</b> User leaves the create listing page before submitting</p> <ol style="list-style-type: none"> <li>1. System prompts user "Are you sure you want to leave and delete listing?"</li> <li>2. If user selects yes, all information entered is deleted and user will return to Manage Listing page</li> <li>3. If user selects no continue main flow from step 2.</li> </ol>
Exceptions:	
Includes:	Enter Listing Details
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<u>LIST_UC_3</u>		
Use Case Name:	<b>Update Listing Details</b>		
Created By:	Hui Wen	Last Updated By:	Hui Wen
Date Created:	1/9/24	Date Last Updated:	1/9/24

Actor:	User
Description:	Allows user to edit their existing listings.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. User must select an existing listing that they had previously created.</li> </ol>
Postconditions:	1. Edits to listing is updated and stored in the system and available for other users to view

Priority:	
Frequency of Use:	3
Flow of Events:	<ol style="list-style-type: none"> <li>1. User selects update listing option</li> <li>2. User edits the information in their listing such as changing the title, description, rating, photo.</li> <li>3. User confirms listing update.</li> <li>4. System saves the edits and displays it on the user's profile. The updated listing is made available for other users to view.</li> </ol>
Alternative Flows:	<p><b>AF-S1-a. User leaves Title/description/rental price/location blank and tries to confirm listing update</b></p> <ol style="list-style-type: none"> <li>1. System displays "Must include Title/description/rental price/location "</li> <li>2. continue from main flow step 2</li> </ol> <p><b>AF-S1-b. User leaves the update listing page before submitting</b></p> <ol style="list-style-type: none"> <li>1. System prompts user "Are you sure you want to leave and delete changes?"</li> <li>2. If user selects yes, all changes are deleted and user will return to ____ page. The listing will not be updated and the original listing remains</li> <li>3. If user selects no continue main flow from step 2.</li> </ol>
Exceptions:	
Includes:	Enter Listing Details
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	LIST_UC_4		
Use Case Name:	Delete Listing		
Created By:	Hui Wen	Last Updated By:	Hui Wen
Date Created:	1/9/24	Date Last Updated:	1/9/24

Actor:	User
Description:	Allows the user to delete their own existing listing.
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. User must select an existing listing that they had previously created.</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. Listing is removed from the system and users are no longer able to view it.</li> </ol>
Priority:	
Frequency of Use:	3

Flow of Events:	<ol style="list-style-type: none"> <li>1. User selects delete listing option</li> <li>2. System prompts user to confirm deletion.</li> <li>3. User confirms listing deletion.</li> <li>4. Listing is deleted from the system and is no longer available for other users to view.</li> </ol>
Alternative Flows:	<b>AF-S1-a. User selects “cancel” when being prompted to confirm deletion</b> <ol style="list-style-type: none"> <li>1. System cancels the deletion, listing is not deleted.</li> <li>2. User is returned to the original listing page.</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>LIST_UC_5</b>		
Use Case Name:	<b>Select Location</b>		
Created By:	<b>Hui Wen</b>	Last Updated By:	<b>Hui Wen</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	<b>User, Maps API</b>
Description:	<b>Allows user to select their location that a rentee can use when deciding to rent.</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in.</li> <li>2. User must allow locations to be used</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. The selected location is stored as part of the listing and is visible to all users</li> </ol>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. System presents a map interface with options to search for locations by entering an address, or using the device’s GPS to determine the current location.</li> <li>2. User enters location</li> <li>3. System asks the user to confirm the selection.</li> <li>4. User reviews the selected location and confirms the location.</li> <li>5. System saves the selected location as part of the rental request.</li> </ol>
Alternative Flows:	<b>AF-S1-a. User Cancels Location Selection:</b> <ol style="list-style-type: none"> <li>1. At any point during the location selection process, the user can cancel the selected location</li> </ol>

	<p>2. The system exits the location selection process and returns the user to the previous screen without saving any location data.</p> <p>3. Continue from main flow 1</p> <p><b>AF-S1-b. Location Not Found:</b></p> <ol style="list-style-type: none"> <li>1. If the system is unable to find the location entered by the user, an error message is displayed which prompts the user to re-enter the address or use the device's GPS location.</li> <li>2. Continue from main flow 1.</li> </ol> <p><b>AF-S1-c. GPS Access Denied:</b></p> <ol style="list-style-type: none"> <li>1. The system allows the user to enter a location manually.</li> <li>2. Continue from main flow 1.</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	LIST_UC_6		
Use Case Name:	Enter Listing Details		
Created By:	Ze Ming	Last Updated By:	Ze Ming
Date Created:	1/9/2024	Date Last Updated:	7/9/2024

Actor:	User
Description:	Enter Listing Details allows user to enter details so that they are able to find related items to borrow
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in</li> <li>2. A listing must have been clicked</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. User selects Listing location</li> <li>2. User selects Availability</li> </ol>
Priority:	
Frequency of Use:	1
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on listing name and inputs listing name</li> <li>2. User clicks on listing details and inputs listing details</li> <li>3. System validates that the input</li> <li>4. User can then select listing location or availability</li> </ol>

Alternative Flows:	<b>AF-S1-a. User cancels enter listing details:</b> 1. User can click back button on the browser tab at any time 2. The system will exit the page and brings the user back to the manage listing page 3. Continue from Manage Listing
Exceptions:	
Includes:	<b>Select Location (LIST_UC_5)</b> <b>Select Availability (LIST_UC_7)</b>
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>LIST_UC_7</b>		
Use Case Name:	<b>Select Availability</b>		
Created By:	<b>Ze Ming</b>	Last Updated By:	<b>Ze Ming</b>
Date Created:	<b>1/9/2024</b>	Date Last Updated:	<b>7/9/2024</b>

Actor:	User
Description:	Select Availability allows the user to select an available time slot from the listing provider to schedule a date and time
Preconditions:	1. User must be logged in 2. A listing must have been clicked 3. User has entered listing details
Postconditions:	
Priority:	
Frequency of Use:	<b>2</b>
Flow of Events:	1. User clicks to select availability 2. System will redirect user to scheduling webpage 3. User can then click on which date and time to schedule the appointment 4. System will then give the user an option to add to their calendar
Alternative Flows:	<b>AF-S1-a. User cancels select availability:</b> 1. User can click back button on the browser tab at any time 2. The system will exit the page and brings the user back to the manage listing page 3. Continue from Manage Listing
Exceptions:	
Includes:	

Extends:	<b>Make Offer (BROWSE_UC_11)</b>
Special Requirements:	
Assumptions:	
Notes and Issues:	

## Browsing System

Use Case ID:	<b>BROWSE_UC_1</b>		
Use Case Name:	<b>Search Listing</b>		
Created By:	<b>Yuxuan</b>	Last Updated By:	<b>Yuxuan</b>
Date Created:	<b>3/9/24</b>	Date Last Updated:	<b>7/9/24</b>

Actor:	<b>User</b>
Description:	<b>The Search Listing function allows the user to search for goods and services currently available on the web app. Users input keywords to search the listings, and the system filters and displays results that match the search criteria.</b>
Preconditions:	
Postconditions:	<b>1. System will display the relevant items or services that match the keywords entered by the user.</b>
Priority:	
Frequency of Use:	<b>5</b>
Flow of Events:	<ol style="list-style-type: none"> <li><b>System prompts the user to enter the keyword into the search bar.</b></li> <li><b>User clicks the search button to initiate the search.</b></li> <li><b>System processes the search request and retrieves listings that match the entered keywords.</b></li> <li><b>System displays a list of relevant listings based on the search results.</b></li> </ol>
Alternative Flows:	<p><b>AF-S1-a. No search results found</b></p> <p><b>Trigger:</b> The system finds no listings that match the entered keywords.</p> <ol style="list-style-type: none"> <li><b>The system displays a message: "No listings found matching your search criteria."</b></li> <li><b>The system suggests related search terms or prompts the user to broaden their search.</b></li> <li><b>Users can modify the keywords and initiate a new search.</b></li> <li><b>continue to main flow step 3</b></li> </ol>
Exceptions:	
Includes:	<b>Open Listing Details ( BROWSE_UC_6)</b>
Extends:	<b>Filter Listing (BROWSE_UC_2)</b>
Special Requirements:	

Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_2</b>		
Use Case Name:	<b>Filter Listing</b>		
Created By:	<b>Yuxuan</b>	Last Updated By:	<b>Yuxuan</b>
Date Created:	<b>3/9/24</b>	Date Last Updated:	<b>7/9/24</b>

Actor:	<b>User</b>
Description:	<b>The filter listing functionality allows users to apply multiple filters to refine the scope of listings and narrow down the search results. This helps users find relevant items or services efficiently based on their specific needs. Available filters may include category, price range, location, availability, and sorting order.</b>
Preconditions:	.
Postconditions:	<ol style="list-style-type: none"> <li><b>The system displays listings that match all the selected filters.</b></li> <li><b>If no listings match the applied filters, the system informs the user accordingly.</b></li> </ol>
Priority:	
Frequency of Use:	<b>4</b>
Flow of Events:	<ol style="list-style-type: none"> <li><b>The system displays a "Filter" button on the web page.</b></li> <li><b>System prompts the user to click the "Filter" button, and the system presents a list of available filter options.</b></li> <li><b>System prompts the user to select multiple filters.</b></li> <li><b>User clicks the "Apply" button to apply the filters.</b></li> <li><b>System processes the filter selections and refreshes the listing page to show only the items or services that meet the selected criteria.</b></li> </ol>
Alternative Flows:	<b>AF-S1-a. Incompatible filters</b> <b>Trigger:</b> The user selects incompatible filters <ol style="list-style-type: none"> <li><b>The system detects conflicting filter criteria.</b></li> <li><b>The system displays a warning message: "Selected filters are incompatible. Please choose again."</b></li> <li><b>The user corrects the filter selection and applies the new filters.</b></li> <li><b>Continues from main flow step 5</b></li> </ol>
Exceptions:	
Includes:	<b>Select Category (BROWSE_UC_3)</b> <b>Select Price Range (BROWSE_UC_4)</b> <b>Sorting order (BROWSE_UC_5)</b>
Extends:	

Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_3</b>		
Use Case Name:	<b>Select Category</b>		
Created By:	<b>Yuxuan</b>	Last Updated By:	<b>Yuxuan</b>
Date Created:	<b>3/9/24</b>	Date Last Updated:	<b>7/9/24</b>

Actor:	User
Description:	Users can select multiple categories from a predefined list to filter and search for relevant items or services. The system will display items or services based on the selected categories, allowing users to narrow down their search.
Preconditions:	
Postconditions:	<ol style="list-style-type: none"> <li>1. The system displays a list of items or services that match the categories selected by the user.</li> <li>2. If no listings match the categories, the system notifies the user and prompts for changes to the search.</li> </ol>
Priority:	
Frequency of Use:	3
Flow of Events:	<ol style="list-style-type: none"> <li>1. System prompts the user to tick multiple category boxes.</li> <li>2. System prompts the user to confirm the category selection by clicking the "Apply" button.</li> <li>3. System retrieves listings that match the selected categories and displays them to the user.</li> <li>4. System display listings that match the filters.</li> </ol>
Alternative Flows:	<p><b>AF-S1-a. No available listings for current filters applied</b></p> <ol style="list-style-type: none"> <li>1. System will display the warning message “No available listing and please enter again” if filters applied have filtered out all listings available</li> <li>2. User must clear the current filters</li> <li>3. Continue from main flow step 1</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_4</b>		
Use Case Name:	<b>Select Price Range</b>		
Created By:	<b>Yuxuan</b>	Last Updated By:	<b>Yuxuan</b>
Date Created:	<b>3/9/24</b>	Date Last Updated:	<b>7/9/24</b>

Actor:	User
Description:	The user specifies a price range for the listings by entering minimum and maximum price values and selecting a relevant time unit (hourly, daily, or weekly). The system will filter and display listings that match the specified price range and time unit.
Preconditions:	
Postconditions:	1. The system displays listings that fall within the specified price range and time unit.
Priority:	
Frequency of Use:	3
Flow of Events:	<ol style="list-style-type: none"> <li>System prompts the user to select the time unit of pricing.</li> <li>System prompts the user to key in minimum and maximum price values</li> <li>User enters minimum and maximum price values.</li> <li>Users click the "Apply" button to apply the filters.</li> <li>The system processes the request and displays listings that match the price range and time unit criteria.</li> <li>System redirects the user to the landing page with all the listings that match the price range filters.</li> </ol>
Alternative Flows:	<p>AF-S1-a. No listing found  <b>Trigger:</b> No listings are available that match the specified price range and time unit.</p> <ol style="list-style-type: none"> <li>The system displays a message: "No listings found for the selected price range and time unit."</li> <li>Users modify the filters to search again.</li> <li>Continue from main flow step 4</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_5</b>
--------------	--------------------

Use Case Name:	<b>Sorting order</b>		
Created By:	<b>Yuxuan</b>	Last Updated By:	<b>Yuxuan</b>
Date Created:	<b>3/9/24</b>	Date Last Updated:	<b>7/9/24</b>

Actor:	<b>User</b>
Description:	<b>System allows users to sort listings based on various sorting metrics, such as distance, price or user rating. Users can choose to sort listings in ascending or descending order based on the selected metric.</b>
Preconditions:	
Postconditions:	<b>1. System will display the listings in sorted order based on the sorting metrics chosen</b>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. System prompts the user to select rate (per hour, day, or week)</li> <li>2. User selects the rate</li> <li>3. System prompts the user to select either ascending or descending</li> <li>4. User enters their choices</li> <li>5. User confirms the selection by clicking "Apply".</li> <li>6. System displays listings sorted according to the selected metrics and order.</li> </ol>
Alternative Flows:	<b>AF-S1-a. No listing to sort</b> <b>Trigger:</b> There are no listings available to sort when the user attempts to apply sorting <ol style="list-style-type: none"> <li>1. The system displays a message: "No listings available to sort."</li> <li>2. System prompts the user to either modify their search criteria or return to the search page to generate listings.</li> <li>3. Continue from main flow step 3</li> </ol>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_6</b>		
Use Case Name:	<b>Open Listing Details</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	<b>User</b>
Description:	<b>Open Listing Details allows the user to view listing details regarding a product</b>
Preconditions:	
Postconditions:	<b>1. User is able to view the details of the product</b>
Priority:	
Frequency of Use:	<b>4</b>
Flow of Events:	<b>1. User clicks on a product 2. System redirects user to a new page displaying listing details of the product as well as actions</b>
Alternative Flows:	
Exceptions:	
Includes:	<b>Report Listing, View User Rating, View Seller Availability</b>
Extends:	<b>Manage Offer</b>
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_7</b>		
Use Case Name:	<b>Report Listing</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	<b>User</b>
Description:	<b>Report Listing allows users to report a listing that is deemed suspicious or illegal</b>
Preconditions:	<b>1. User must have an User account 2. User must have clicked on the listed product that they wish to report</b>
Postconditions:	<b>1. Listed product is removed from the shop and the account associated with it may be banned</b>

Priority:	
Frequency of Use:	<b>1</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on the report button</li> <li>2. Report Listing pop-up window appears</li> <li>3. User enters details on violations/community guidelines that the listed product has invoked</li> <li>4. User submits the information</li> <li>5. System will do a review of the report and log the event</li> <li>6. Review team will remove the listed product and ban the account associated with it if necessary</li> </ol>
Alternative Flows:	<b>AF-S1-a.</b>  <b>User clicks on the X button on the Report Listing pop-up window which causes it to be closed</b>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_8</b>		
Use Case Name:	<b>View Seller Availability</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	User
Description:	<b>View Seller Availability allows users to view when the seller is available to allow collection of the item for the user (customer)</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have clicked on the view availability button on the product page</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. A calendar is displayed. Available dates are shown to the user for collection</li> </ol>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on the view availability button</li> <li>2. System redirects user to a new page</li> <li>3. System retrieves data from the scheduling API to display on the new page</li> <li>4. System displays a calendar with the available dates shown for user to collect the product</li> </ol>
Alternative Flows:	
Exceptions:	

Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_9</b>		
Use Case Name:	<b>View User Ratings</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	User
Description:	<b>View User Ratings allows users (customer) to view the user ratings/reviews of the seller. User ratings will include ratings ranging from 1-5 stars from past deals/transactions completed by the seller</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have an User account</li> <li>2. User must have clicked on the view user ratings button on the product page</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. Past deals completed by the seller with the user ratings and comments is displayed to the user (customer)</li> </ol>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on the view user ratings button</li> <li>2. System redirects the user to the renter's profile page which contains the reviews</li> <li>3. System displays the information to the user</li> </ol>
Alternative Flows:	
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_10</b>		
Use Case Name:	<b>Manage Offer</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>

Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>
---------------	---------------	--------------------	---------------

Actor:	<b>User</b>
Description:	<b>Manage offer allows the user to make/accept/reject offers.</b> <b>Typically, the renter is the only party allowed to accept or reject offers while the rentee is only allowed to make offers.</b>
Preconditions:	<b>2. User must have an User account</b>
Postconditions:	<b>2. User is able to make/accept/reject offers</b> <b>3. Database will be updated with the information</b>
Priority:	
Frequency of Use:	<b>4</b>
Flow of Events:	<b>1. User clicks on a product</b> <b>2. System identifies if user is a renter or rentee</b> <b>3. System will allow user to make offer if user is a rentee; accept/reject offer if user is a renter</b> <b>4. User will make a decision</b> <b>5. System will perform the corresponding tasks according to user's decision</b>
Alternative Flows:	
Exceptions:	
Includes:	<b>Make Offer, Accept Offer, Reject Offer</b>
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_11</b>		
Use Case Name:	<b>Make Offer</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	<b>User</b>
Description:	<b>Make Offer allows the rentee to make an offer to the renter.</b>
Preconditions:	<b>1. User must have an User account</b> <b>2. User must have clicked on the make offer button on the product page</b>
Postconditions:	<b>1. The renter is informed of the offer that was made by the user</b>
Priority:	
Frequency of Use:	<b>3</b>

Flow of Events:	<ol style="list-style-type: none"> <li>1. User sets a price for the renter</li> <li>1. User clicks on the make offer button</li> <li>2. User submits the price to the system</li> <li>3. System informs the renter that an offer has been made</li> </ol>
Alternative Flows:	
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_12</b>		
Use Case Name:	<b>Accept Offer</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	User
Description:	<b>Accept Offer allows the renter to accept an offer that was made by rentee.</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have an User account</li> <li>2. User has received an offer on their product listing</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. The rentee is informed that the offer has been accepted by the renter</li> <li>2. Listing will be marked as sold on the database</li> </ol>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on “Your rentals” on profile icon sidebar</li> <li>2. User (renter) clicks on the product with offers</li> <li>3. User (renter) clicks on the accept offer button</li> <li>4. System will notify the other party (rentee) that the offer has been accepted by the renter</li> </ol>
Alternative Flows:	<b>AF-S1-a.</b>  <b>User clicks on the X button on the accept offer window which causes it to be closed</b>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>BROWSE_UC_13</b>		
Use Case Name:	<b>Reject Offer</b>		
Created By:	<b>Seow Jia Xian</b>	Last Updated By:	<b>Seow Jia Xian</b>
Date Created:	<b>1/9/24</b>	Date Last Updated:	<b>1/9/24</b>

Actor:	<b>User</b>
Description:	<b>Reject Offer allows the renter to reject an offer that was made by the rentee</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must have an User account</li> <li>2. User must have clicked on the reject offer button on their product listing page</li> </ol>
Postconditions:	<ol style="list-style-type: none"> <li>1. The rentee is informed that the offer has been rejected by the seller</li> </ol>
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. User clicks on “Your rentals” on profile icon sidebar</li> <li>2. User (renter) clicks on the product they want to reject offers</li> <li>3. User (renter) clicks on the reject offer button</li> <li>4. System will notify the other party (rentee) that the offer has been rejected by the seller</li> </ol>
Alternative Flows:	<b>AF-S1-a.</b>  <b>User clicks on the X button on the reject offer window which causes it to be closed</b>
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

## Payment System

Use Case ID:	<b>PAY_UC_1</b>		
Use Case Name:	<b>Authorize Payment</b>		
Created By:	<b>Ze Ming</b>	Last Updated By:	<b>Ze Ming</b>
Date Created:	<b>1/9/2024</b>	Date Last Updated:	<b>7/9/2024</b>

Actor:	<b>User, Payment API</b>
Description:	<b>Authorize Payment allows the user to pay for the service</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in</li> <li>2. User must have selected a listing to pay for</li> </ol>
Postconditions:	
Priority:	
Frequency of Use:	<b>2</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. User will select to pay for the listing</li> <li>2. System will redirect the user to the payment gateway</li> <li>3. User will input their payment methods and pay</li> <li>4. System will verify that payment has been made</li> </ol>
Alternative Flows:	<b>AF-S1-a. User cancels authorize payment:</b> <ol style="list-style-type: none"> <li>4. User can click back button on the browser tab at any time</li> <li>5. The system will exit the page and brings the user back to the manage listing page</li> <li>6. Continue from Manage Listing</li> </ol>
Exceptions:	
Includes:	<b>Enter Card Details (PAY_UC_2)</b> <b>Produce Receipt (PAY_UC_3)</b>
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>PAY_UC_2</b>		
Use Case Name:	<b>Enter Card Details</b>		
Created By:	<b>Ze Ming</b>	Last Updated By:	<b>Ze Ming</b>
Date Created:	<b>1/9/2024</b>	Date Last Updated:	<b>7/9/2024</b>

Actor:	<b>User, Payment API</b>
Description:	<b>Authorize Payment allows the user to pay for the service</b>
Preconditions:	<ol style="list-style-type: none"> <li>1. User must be logged in</li> <li>2. User must have selected a listing to pay for</li> </ol>
Postconditions:	
Priority:	
Frequency of Use:	<b>3</b>
Flow of Events:	<ol style="list-style-type: none"> <li>1. System will prompt for payment methods</li> <li>2. User will select to enter card details and enter card details</li> </ol>

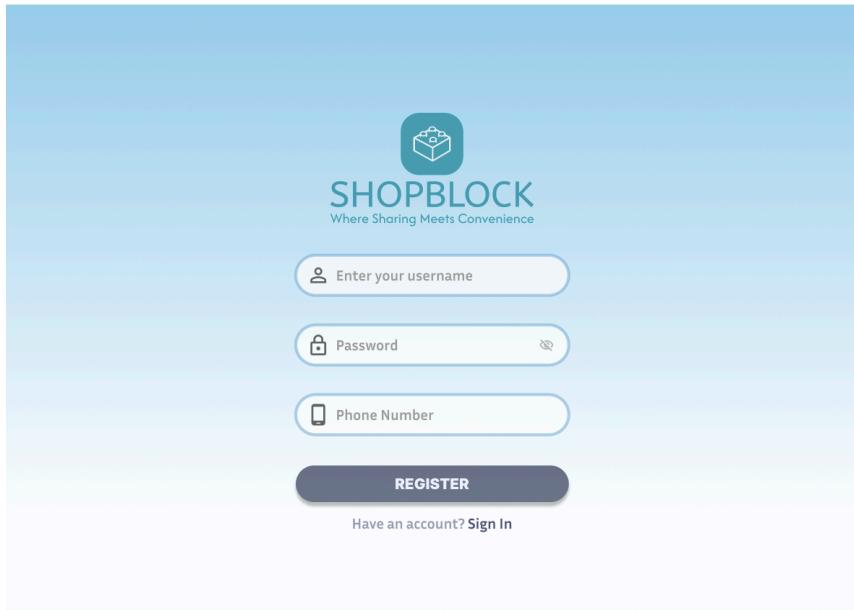
	<b>3. System will verify that the card details are valid and make payment</b>
Alternative Flows:	User clicks the back button on the browser tab, the system will then show the initial payment page
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case ID:	<b>PAY_UC_3</b>		
Use Case Name:	<b>Produce Receipt</b>		
Created By:	<b>Ze Ming</b>	Last Updated By:	<b>Ze Ming</b>
Date Created:	<b>1/9/2024</b>	Date Last Updated:	<b>7/9/2024</b>

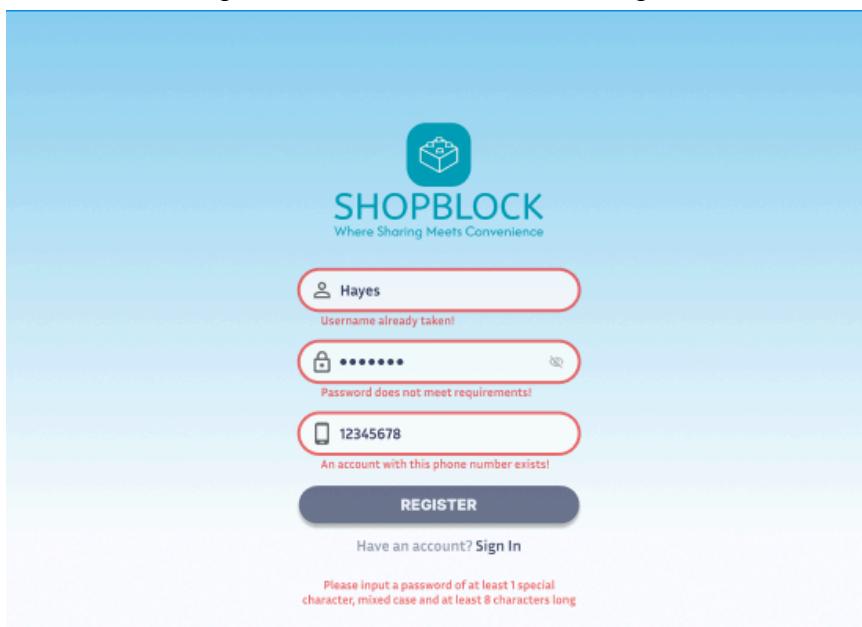
Actor:	<b>User, Payment API</b>
Description:	<b>Authorize Payment allows the user to pay for the service</b>
Preconditions:	<b>1. User must be logged in 2. User must have selected a listing to pay for</b>
Postconditions:	
Priority:	
Frequency of Use:	<b>2</b>
Flow of Events:	<b>1. User logs into the Payment API gateway 2. User looks for the payment transaction to produce a receipt 3. Payment API will produce a receipt for the user</b>
Alternative Flows:	User clicks the back button on the browser tab, the system will then show the initial payment page
Exceptions:	
Includes:	
Extends:	
Special Requirements:	
Assumptions:	
Notes and Issues:	

## **5.0 UI Mockups**

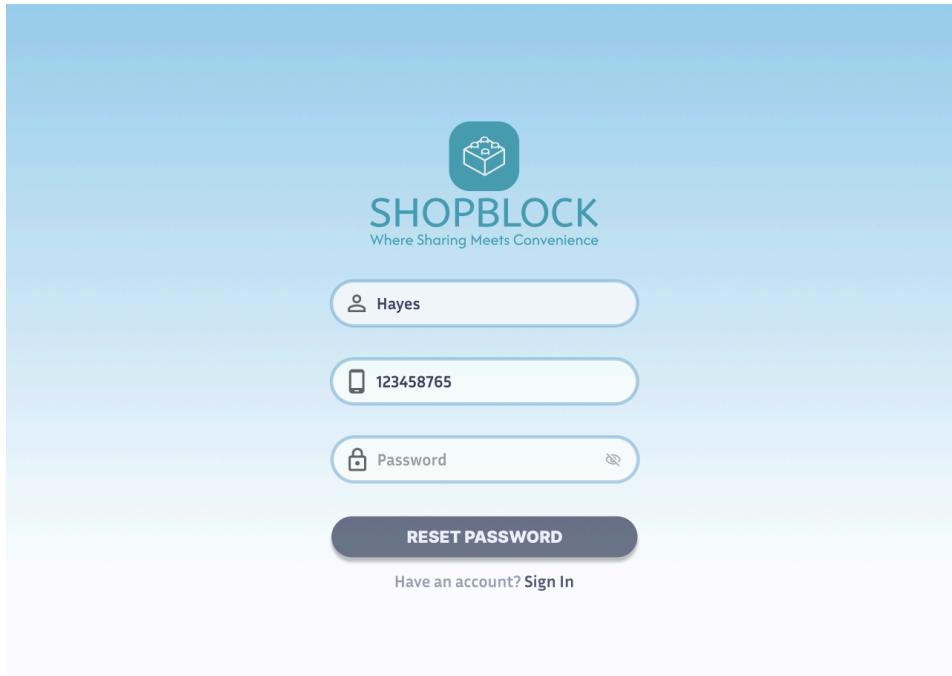
User Account Management System  
USER\_UC\_1 - Register/Sign - up page



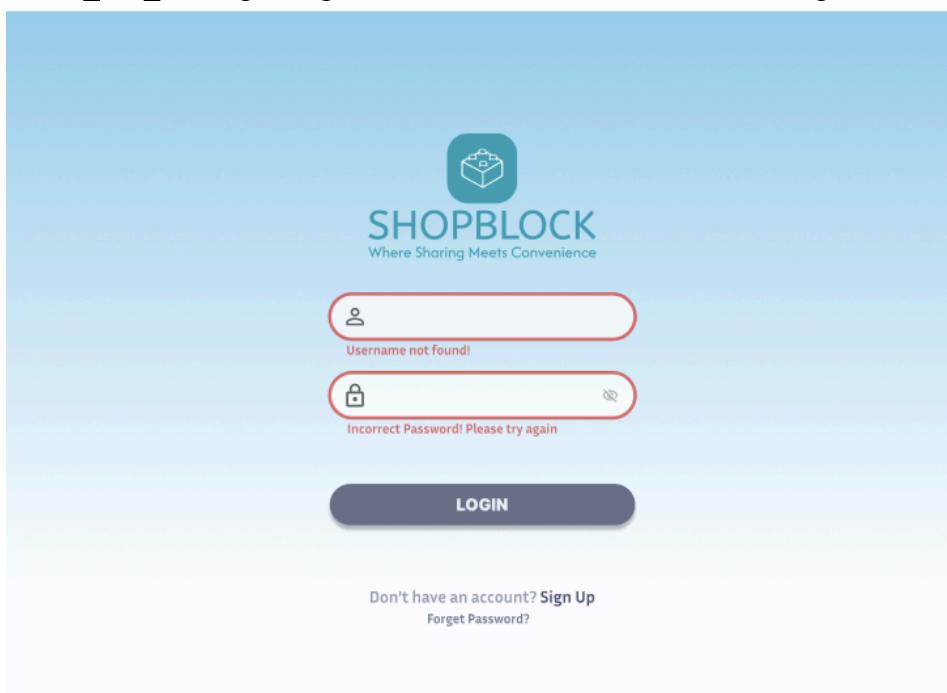
USER\_UC\_1 - Register/Sign - up page - alternative flows: Username already taken/Password does not meet requirements/An account with this phone number exists



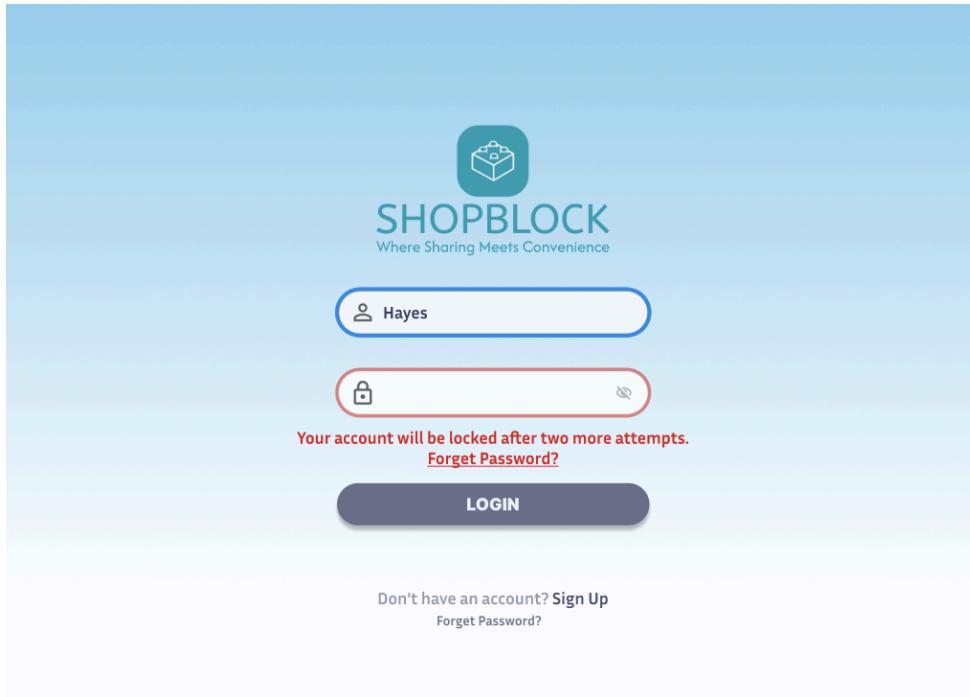
USER\_UC\_2 - Login Page



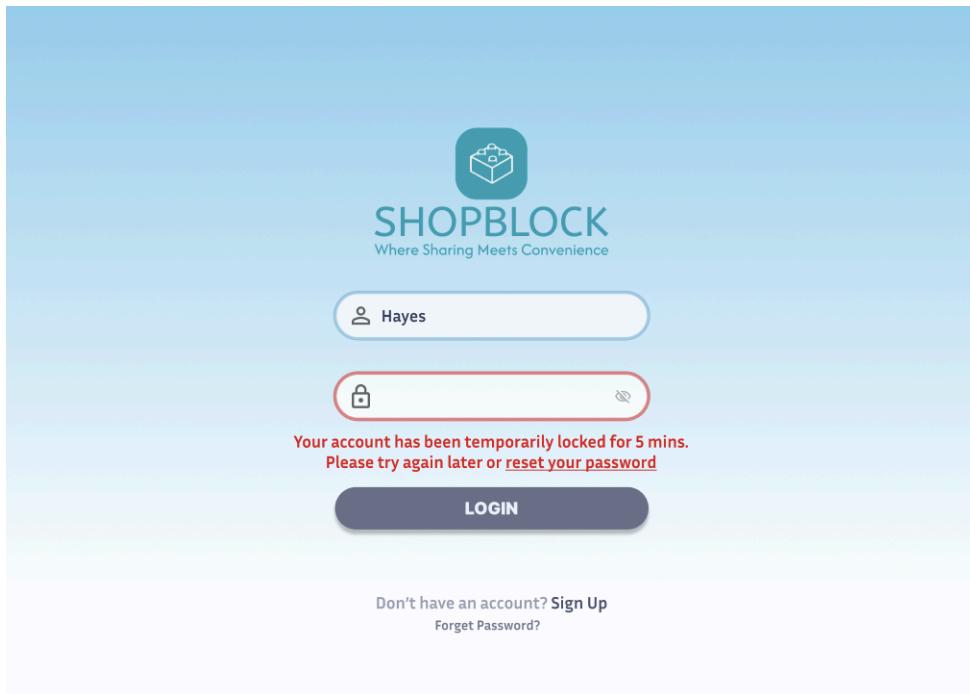
USER\_UC\_2 - Login Page - Username not found *OR* incorrect password



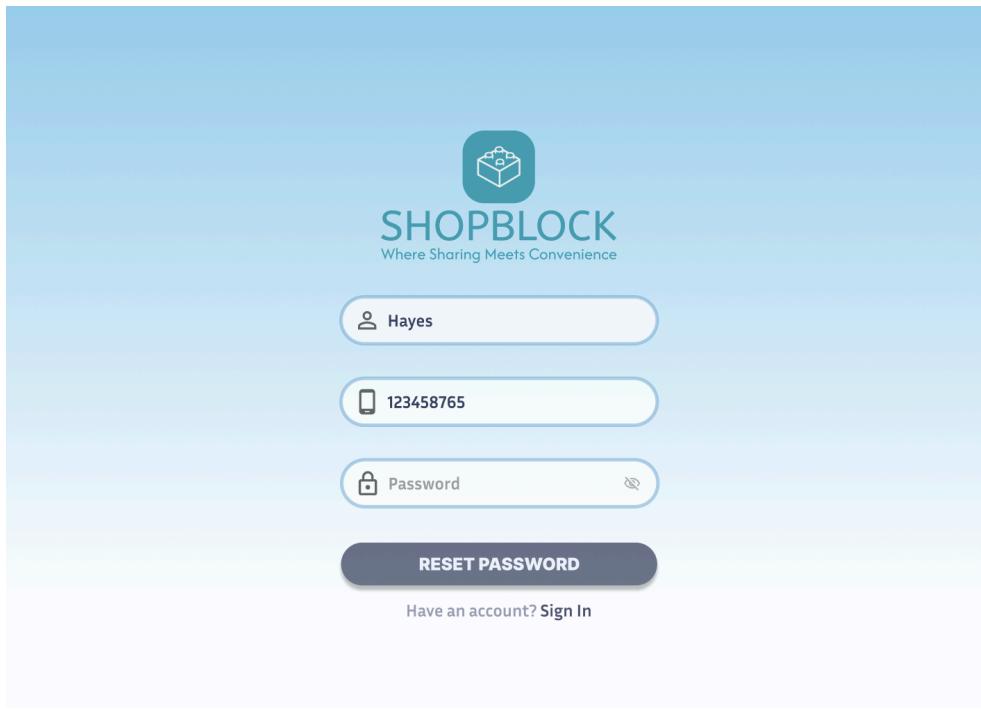
USER\_UC\_2 - Login Page - (after 3 incorrect password)



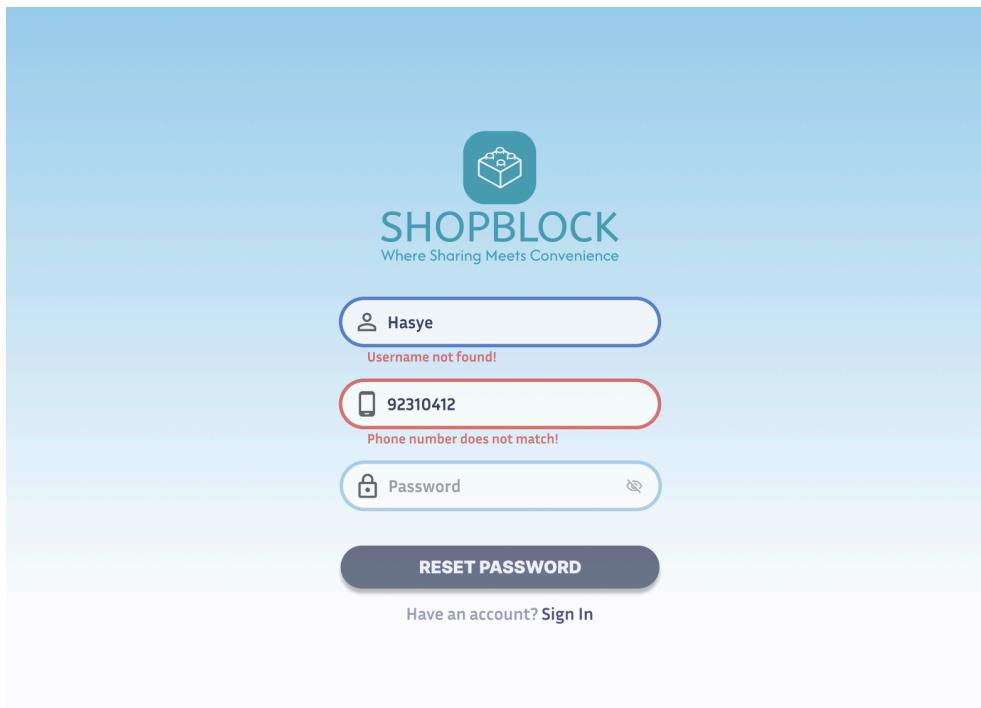
USER\_UC\_2 - Login Page - (account locked after 5 tries)



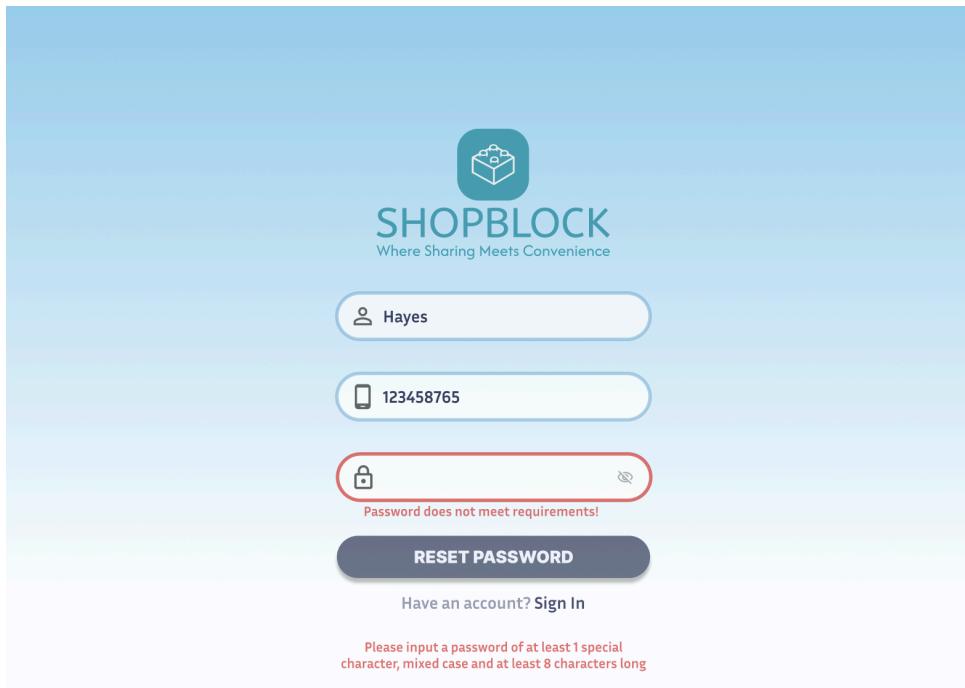
USER\_UC\_3 - Forget password



USER\_UC\_3 - Forget password (Username \*OR\* Phone number does not match)



USER\_UC\_3 - Forget password (Password Requirement)



USER\_UC\_4,5,7 - View Account details, Manage Account Details, Edit User Details

The screenshot displays the ShopBlock user profile for a user named "RonaldoCR7". The header includes the ShopBlock logo, navigation links for Electronics, Services, Supplies, All Categories, and a "GET 20% OFF FIRST RENTAL" offer, along with a user icon and a plus sign for adding items.

**Profile**

Here's my short description

@USER12345  
5.0 ★★★★★ (234)

**About me** [Edit](#)

- 52 Listings
- 104 Rentals

**Username** RonaldoCR7  
**Mobile** +65 86\*\* \*\*\*\*  
 Bishan, Singapore

**Biography**

Hey there! I'm Ronaldo, a marketing professional by day and a sports enthusiast by night. Originally from Brazil and now calling Singapore home, I've embraced the blend of cultures and the vibrant lifestyle this city offers. My weekends are often filled with football matches, hiking, and capturing the beauty of Singapore with my DSLR camera.

**Reviews**

Hazard23 ★★★★★ Review from rentee 1mo  
Great renter to deal with! Punctual! Pleasant transaction! Very nice and friendly renter to deal with, thanks for the product.

AntoineGriezmannnn ★★★★★ Review from rentee 2mo  
Fantastic renter to work with! Punctual and friendly, making for a very pleasant transaction. Thank you for the great product!

USER\_UC\_7 - Edit User Details (Username has been taken)

**SHOPBLOCK**

GeoSearcher FAQ Cart Search User Profile

Electronics Services Supplies All Categories GET 20% OFF FIRST RENTAL +

## Profile

Here's my short description

@RonaldoCR7 5.0 ★★★★★ (234)

**About me** Edit

52 Listings 104 Rentals

**Username:** RonaldoCR7 **Mobile:** +65 86\*\* \*\*\*\* **Location:** Bishan, Singapore

**Biography**

Hey there! I'm Ronaldo, a marketing professional by day and a sports enthusiast by night. Originally from Brazil and now calling Singapore home. I've embraced the blend of cultures and the vibrant lifestyle this island city offers. In my free time, I enjoy football matches, hiking, and photography with my DSLR camera.

**Username already taken** Try Again

Hazard23 ★★★★★ Review from rentee 1mo  
Great renter to deal with! Punctual! Pleasant transaction! Very nice and friendly renter to deal with, thanks for the product.

AntoineGriezmannnn ★★★★★ Review from rentee 2mo  
Fantastic renter to work with! Punctual and friendly, making for a very pleasant transaction. Thank you for the great product!

## USER\_UC\_6 - Change Password

**SHOPBLOCK**  
Where Sharing Meets Convenience

Enter current password

Enter new password

Confirm new password

**RESET PASSWORD**

## Review System

REVIEW\_UC\_1,2,3,4 - Submit Review

The screenshot shows the 'Transaction History' section of the SHOPBLOCK website. A completed transaction for 'Pet Walking Service' by 'Hazard23' is displayed. The service was \$9/hr for 1700 hours from 19/12/2024 to 19/12/2024, totaling \$54. The status is 'COMPLETED'. Below the transaction details is a review form with a 5-star rating and a placeholder 'Write your review here'. Buttons for 'Add photo' and 'Submit' are present.

## Listing System

LIST\_UC\_1 - Mange Listings (Your Listings page)

The screenshot shows the 'Your Listings' section of the SHOPBLOCK website. It displays 62 items found in 'Your Listings'. The left sidebar includes filters for 'Category' (Electronics, Services, Supplies, All categories), 'Rates' (Hourly, Daily, Weekly), and 'Price Range' (with input fields and an 'Enter' button). The main area shows a grid of eight service and product listings with images and descriptions:

- Mahjong Table \$10/Day (Yellow checkmark)
- Pet Walking Services \$20/Hr (Yellow checkmark)
- Housekeeping \$200/Day (Yellow checkmark)
- Electrician \$30/Hr (Green checkmark)
- Clothes Rack \$16/Day (Green checkmark)
- Plumber \$60/Hr (Green checkmark)
- Power Drill \$20/Day (Green checkmark)
- DJ Equipment \$20/Hr (Green checkmark)

Navigation arrows at the bottom indicate the current page is 1 of 1.

## LIST\_UC\_2 - Make New Listing

The screenshot shows the SHOPBLOCK website interface. At the top, there is a navigation bar with links for Electronics, Services, Supplies, All Categories, a GeoSearcher button, a FAQ link, a shopping cart icon, a search bar, and a user profile icon. A promotional banner for "GET 20% OFF FIRST RENTAL" is visible. On the left, a sidebar displays categories like Electronics, Services, Supplies, and All categories, along with filters for Rates (Hourly, Daily, Weekly) and Price Range. Below the sidebar, there are four small preview cards for "Portable AC \$5/Day", "Cleaning Services \$10/Hr", "Ladder \$3/Hr", and "Walk-A-Dog Service \$20/Hr". The main content area features a "Trending Now" section with a "Create Listing" modal. The modal has fields for Title, Description, Price, Unit, Category, and an "Add photo" button. A "Submit" button is located at the bottom right of the modal. The background shows a blurred view of the trending items.

## LIST\_UC\_3 - Update Listing Details / Delete Listing

This screenshot shows the same SHOPBLOCK website interface as the previous one, but with a different modal open. The "Edit Listing" modal is displayed over the "Create Listing" modal. It contains the same fields as the creation modal: Title, Description, Price, Unit, Category, and an "Add photo" button. Additionally, it includes a red "Delete Listing" button. The "Submit" button remains at the bottom right. The background elements, including the sidebar, trending items, and promotional banner, are identical to the first screenshot.

# Browsing System

## Home page

The screenshot shows the ShopBlock homepage. At the top, there's a navigation bar with icons for GeoSearcher, FAQ, Cart, and a User profile. Below the bar, there are category links: Electronics, Services, Supplies, and All Categories. A promotional banner says "GET 20% OFF FIRST RENTAL". On the left, there are filter sections for Category (Electronics, Services, Supplies, All categories), Rates (Hourly, Daily, Weekly, Desc Asc), and Price Range (with an "Enter" button). The main content area is titled "Trending Now" and features eight items with images and descriptions: Poker Table (\$40/Day), Poker Set (\$4/Hr), Monopoly Set (\$10/Day), Louis V Luggage (\$450/Week), Portable AC (\$5/Day), Cleaning Services (\$10/Hr), Ladder (\$3/Hr), and Walk-A-Dog Service (\$20/Hr). A "Filter By" dropdown menu is visible on the right.

Browse\_UC\_1,2,3,4,5 -Search listing - Filter Listing - Select category - Select Price RangeShopBlock Filter Option Showcase

This screenshot is similar to the first one but includes a "Your Listings" and "Your Rentals" overlay on the right side of the "Filter By" dropdown. The rest of the interface and content are identical to the first screenshot, showing the trending items and filtering options.

## Browse\_UC\_6 - Open Listing Details - ShopBlock Single Listing Page (Rentee POV)

The screenshot shows a listing for a "Mahjong Table" in the "Supplies" category. The listing includes a main image of the table, a side image of its legs, and a detailed view of the playing surface. A "View Location" button is present. The listing details show it costs \$10/Day and is posted by a user with a 5.0 rating. A map of the area around Marymount is shown below the listing.

**Mahjong Table** ⓘ (Report Listing)  
S\$10/Day

**Description**  
For those who love modern convenience, our mahjong table comes with optional automatic tile shuffling and dealing functions, saving time and adding a touch of sophistication to your game nights. It also includes built-in drawers for storing tiles and accessories, plus cup holders for added comfort during long sessions.

**@RonaldoCR7** 5.0 ★★★★★ (23)

S\$ 7 Make Offer  
View Availability  
View User Ratings  
View renter offers

**View Location**

**Map:** A map showing the location of the listing near Marymount, with nearby landmarks like Apex Tyres, Good Bites, and Genki Forest marked.

## Browse\_UC\_7 - Report Listing - ShopBlock Single Listing Page (Rentee POV)

The screenshot shows the same Mahjong Table listing as above, but with a "Report Listing" modal open in the center. The modal asks the user to provide reasons and details for reporting the listing. The background listing and map are visible behind the modal.

**Report Listing**

Please provide the reason and details for reporting this listing for violating our community guidelines.

**Submit**

**Mahjong Table** ⓘ  
S\$10/Day

**Description**  
For those who love modern convenience, our mahjong table comes with optional automatic tile shuffling and dealing functions, saving time and adding a touch of sophistication to your game nights. It also includes built-in drawers for storing tiles and accessories, plus cup holders for added comfort during long sessions.

**@RonaldoCR7** 5.0 ★★★★★ (23)

S\$ 7 Make Offer  
View Availability  
View User Ratings  
View renter offers

**View Location**

## Browse\_UC\_8 - View Seller Availability

The screenshot shows a booking interface for 'blockshop' service by 'Jane Doe'. At the top, there's a green header with the service name and a progress bar showing step 1 of 3. Below the header is a section titled 'APPOINTMENT DATE & TIME'. On the left, a calendar for September 2024 is displayed, with the 2nd highlighted. On the right, a 'Timezone' dropdown is set to 'Singapore (+8:00)' with four time options: 11:00 PM, 11:15 PM, 11:30 PM, and 11:45 PM. At the bottom right is a 'NEXT >' button.

## Browse\_UC\_9 - View User Ratings - (Renter POV)

The screenshot shows a user profile page for 'SHOPBLOCK'. The top navigation bar includes categories like Electronics, Services, Supplies, All Categories, and a promotional banner for 'GET 20% OFF FIRST RENTAL'. The main content area features a 'Profile' section with a placeholder photo and a short description. Below it is an 'About me' section with icons for 52 Listings and 104 Rentals. The 'Reviews' section contains two entries: one from 'Hazard23' (5.0 stars) praising punctuality and friendliness, and another from 'AntoineGriezmannnn' (5.0 stars) for a pleasant transaction. The user's username is listed as 'RonaldoCR7'.

## Browse\_UC\_10,11 - Manage Offer - Make Offer - (Rentee POV)

The screenshot shows a product listing for a "Mahjong Table". The top navigation bar includes categories like Electronics, Services, Supplies, and All Categories, along with a promotional banner for "GET 20% OFF FIRST RENTAL". The main content area displays three images of the mahjong table: a top-down view, a side view of the tile racks, and a top-down view with labels for North, South, East, and West. Below the images, the product name "Mahjong Table" is listed with a price of "\$\$10/Day". A "Description" section explains the features of the table. To the right, there is a user profile for "@LebronJ06" with a yellow icon, showing a rating of "No Ratings Yet". Below the profile are buttons for "Accept Offer" (green), "Reject Offer" (red), and "View all offers" (blue).

## Browse\_UC\_10,12,13 - Manage Offer - Accept Offer - Reject Offer (Renter POV)

The screenshot shows a "Listings" section on the left and an "Offers (Mahjong Table)" section on the right. The Listings section includes items like a Monopoly Set (\$10/Day), a Mahjong Table (\$10/Day), a Walk-A-Dog Service (\$20/Hr), and DJ Equipment (\$20/Hr). The Offers section lists two offers for the Mahjong Table: one from "Hazard23" for \$30 (Total \$30) and another from "Hazard256" for \$10 (Total \$10). Each offer has a green checkmark button and a red X button next to it.

# Payment System

## Payment\_UC\_1 - Authorize Payment



**Pay with PayPal**

Enter your email address to get started.

[Forgot email?](#)

**Next**

or

**Pay with Debit or Credit Card**

## Payment\_UC\_2 - Enter Card Details

[Pay now](#)[Pay Later](#)

## Pay with debit or credit card

We don't share your financial details with the merchant.

  
United States

Phone type Mobile	Phone number +1
----------------------	--------------------



Expiration date	CVV
-----------------	-----

### Billing address

First name	Last name
------------	-----------

Shipping same as billing address

[Save info & create your PayPal account](#) 



Shop with confidence



Shop millions of merchants



Save big with great deals

By creating an account, you confirm you're at least 18 years old and agree to the [E-sign Consent](#), [User Agreement](#) and [Privacy Statement](#).

[Create Account & Continue](#)

Have a PayPal account? [Log In](#)

[Français](#) [Español](#) [中文](#) | 

[Policies](#) [Legal](#) [Privacy](#)

© 1999 - 2024



&lt; Back

## Transaction details

 Print

Payment received from John smith

January 30, 2023 at 7:12:47 PM PST

| Transaction ID: 21T2978457491141A

Gross amount

\$428.48 USD

Payment Status: COMPLETED

Issue a refund

Payment Type: Mobile Payment

OK to ship to

JohnSmith

123 E Main St  
Anywhere, State  
UNITED STATES  
Confirmed 

Seller protection

Eligible

Eligible when you ...

Ship to the address on this page.

Save your tracking or shipping info.