

**CAPSTONE PROJECT REPORT**

**Software Requirement Specification**

– Da Nang, June 2025 –

# Table of Contents

[**Table of Contents 2**](#_7z3k3q21g1nd)

[**I. Record of Changes 3**](#_tdgqz6z5wucc)

[II. Software Requirement Specification 4](#)

[1. Product Overview 4](#_l7urj7g6p6dh)

[2. User Requirements 5](#_ef7rx557bat2)

[2.1 Actors 5](#_l4ahokc2bnxs)

[2.2 Use Cases 6](#_26a4d5apkna7)

[2.2.1 Diagram(s) 6](#_2ure8wtw2rmf)

[2.2.2 Descriptions 8](#)

[2.2.2.1 Account Management 9](#_865p6nn6ifob)

[2.2.2.2 User Management 12](#_gw05ipmj56a4)

[2.2.2.3 Motorbike Management 15](#_b7spvrf7zpda)

[2.2.2.4 Website Management 19](#_ilb0j0769g6g)

[2.2.2.5 Bike Rental Management 22](#_gzggqzryrwal)

[2.2.2.6 Reporting and Statistics 25](#_7as684y5xb35)

[2.2.2.7 Support and Feedback 26](#_w8j9lb5svd1k)

[2.2.2.8 Motorbike Browsing and Renting 29](#_pnxilm13zcus)

[2.2.2.9 Commenting 35](#_i50hbr58yncy)

[3. Functional Requirements 36](#_iv6xiltt3rui)

[3.1 System Functional Overview 36](#_d0m3pu37kmxq)

[3.1.1 Screens Flow 36](#)

[3.1.2 Screen Descriptions 36](#)

[3.1.3 Screen Authorization 37](#)

[3.1.4 Non-Screen Functions 37](#)

[3.1.5 Entity Relationship Diagram 39](#)

[3.2 Account Management 40](#_3tdm7xqmxrtk)

[3.2.1 Sign In 40](#_cblwi3aagooz)

[3.2.2 Sign up 41](#_4isbnqv0e0ra)

[3.2.3 User Profile 42](#_haapch)

[3.4 Support and Feedback 43](#_pjje418jmnqi)

[3.4.1 Chat Window 43](#_miveh5e15wg2)

[3.5 Motorbike browsing and renting 44](#_1bzw5zti7d6n)

[4. User Management 45](#_u294y237f6kq)

[5. Motorbike Management 46](#_mhwwwjwp1p3h)

[6. Website Management 46](#_lzvfme6fvkc7)

[7. Bike Rental Management 46](#_hu4rf1wxvxwz)

[8. Reporting and Statistics 48](#_nlel0le68sl6)

[4. Non-Functional Requirements 48](#_hpunscyiq234)

[4.1 External Interfaces 48](#_d2vk3dphwgto)

[4.1.1 Hardware 48](#_v0g4sklqnwq)

[4.1.2 Software 48](#_8oquaztew5uq)

[4.2 Quality Attributes 48](#_f3p4idlynpkr)

[4.2.1 Usability 48](#_tvlc3fvpgww2)

[4.2.2 Reliability 48](#_ly63dq5qcn6o)

[4.2.3 Performance 48](#_3yb8cue7b1jh)

[4.2.4 Security 49](#)

[5. Requirement Appendix 49](#)

[5.1 Business Rules 49](#_kjpquwy8787g)

[5.2 Application Messages List 51](#)

# I. Record of Changes

| **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- |
| 04/06 | A | HieuNM | Initial commit. |
| 06/06 | M | HieuNM | Rewrote details of Use case 24, 25, 26, 29, 30, and 31.  Reformatted use case forms for consistency.  Correctly appended business rules to multiple use cases. |
| 07/06 | M | BinBK | Modified [use case diagram](#_2ure8wtw2rmf). |
| A | Everyone | Added UI descriptions and screens for each [feature](#_iv6xiltt3rui). |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

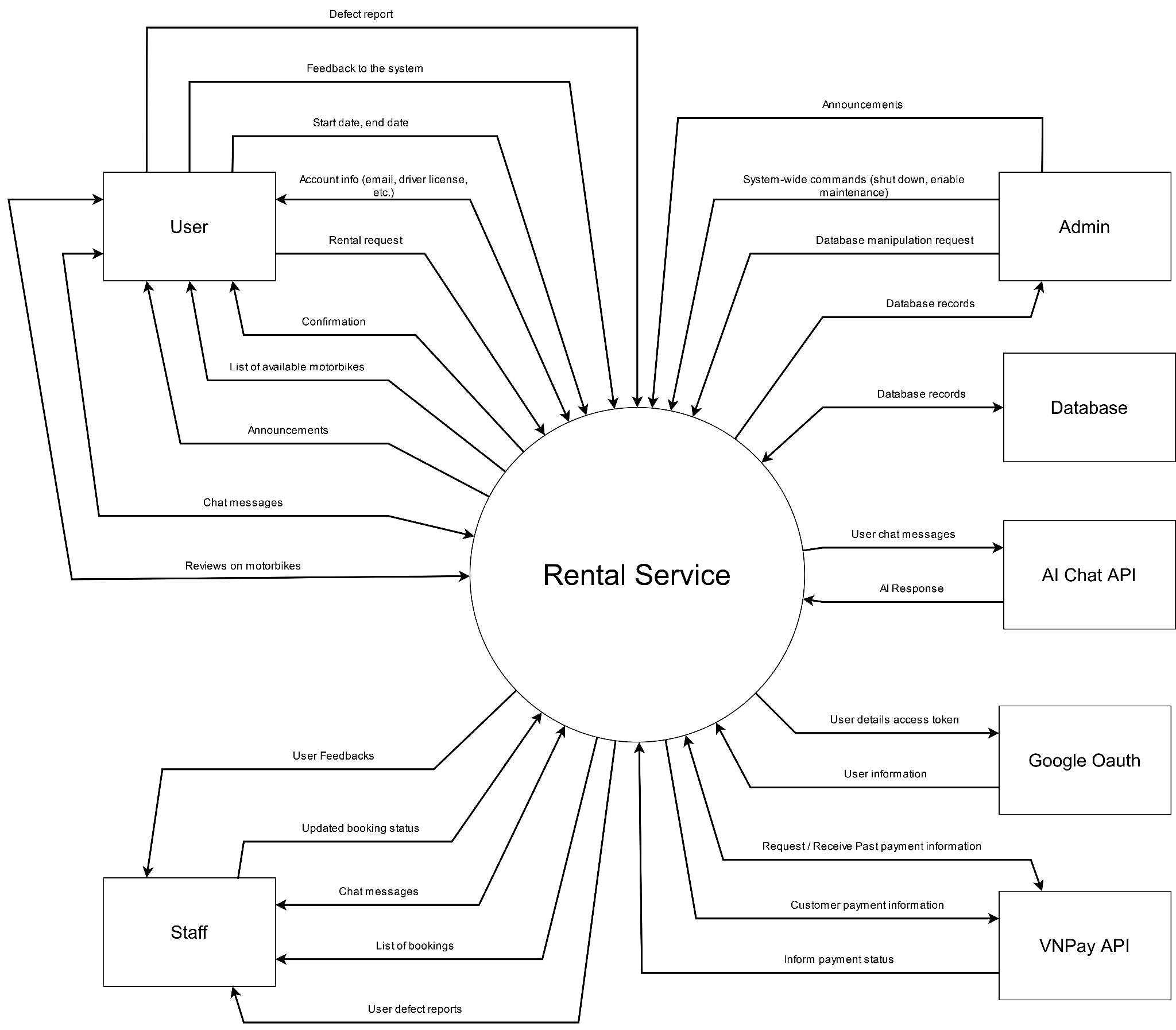
\*A - Added M - Modified D - Deleted

# II. Software Requirement Specification

## 1. Product Overview

[vroomvroom.click](http://vroomvroom.click) is a web application for self-drive motorbike and car rental services in major cities such as Hanoi, Da Nang, and Ho Chi Minh City. The platform is designed to simplify the vehicle rental experience by enabling users to browse available vehicles, create rental requests, and manage their bookings through a user-friendly interface.

The primary goal is to demonstrate a functional rental system that is fast, accessible, and transparent, addressing common pain points in the current vehicle rental process in Vietnam. While not intended for production deployment, this prototype showcases the key functionalities of a rental platform from both the user and administrative perspectives.



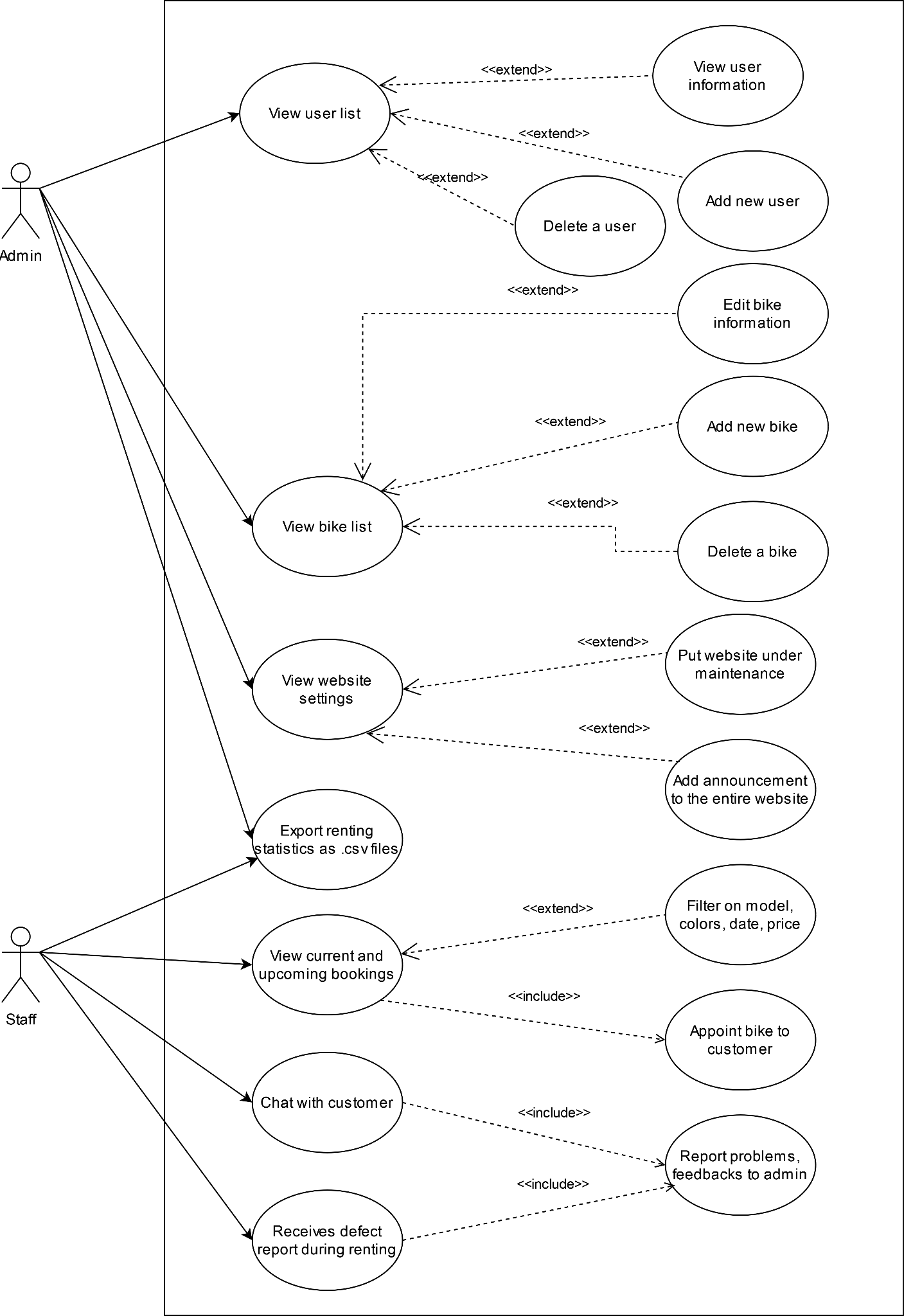
## 2. User Requirements

### 2.1 Actors

| **#** | **Actor** | **Description** |
| --- | --- | --- |
| 1 | Admin | The system administrator can log in to the admin panel, manage motorbikes (CRUD operations), manage customer and staff accounts, handle booking details and status updates, generate reports and statistics on revenue and bookings, and approve bookings if needed. |
| 2 | Staff | The operational staff is responsible for confirming bookings, preparing motorbikes (cleaning, handing over, receiving), assisting customers at pickup points, and managing bike statuses (available, rented, maintenance). |
| 3 | User/Customer[[1]](#footnote-0) | The end user can register, log in, search for motorbikes (by type, price, location, weather), book bikes online, make payments via VNPay, track booking status and history, use Google Maps for bike and pickup/drop-off locations, and interact with a support chatbot. |

### 2.2 Use Cases

#### 2.2.1 Diagram(s)

**

#### 2.2.2 Descriptions

| ID | Feature | Use Case | Use Case Description |
| --- | --- | --- | --- |
| 01 | Account Management | Register | A new user registers for an account to use the rental service. |
| 02 | Login | A user logs into the system with their existing account. |
| 03 | View Account | Registered users view their account information. |
| 04 | Edit Information | Registered users edit their personal information (name, address, etc.). |
| 05 | Request Password Reset Email | Registered users request a password reset email if they forget their password. |
| 06 | User Management | View User List | Admin views the list of all users in the system. |
| 07 | View User Information | Admin views detailed information about a specific user. |
| 08 | Edit User Information | Admin edits user information to the system. |
| 09 | Delete a User | Admin deletes a user from the system. |
| 10 | Motorbike Management | View Motorbike List | Admin views the list of all motorbikes in the system. |
| 11 | Edit Motorbike Information | Admin edits the information of a motorbike (model, price, condition, etc.). |
| 12 | Add New Motorbike | Admin adds a new motorbike to the system. |
| 13 | Delete bike | Admin deletes existing motorbikes. |
| 14 | Filter Motorbikes | Staff filters the motorbike list by model, color, date, or price. |
| 15 | Website Management | View website settings | Admin views the settings of the website in the system. |
| 16 | Put website under maintenance | Admin puts the website into maintenance mode for updates or fixes. |
| 17 | Add announcement to the entire website | Admin displays a banner announcement in the landing page/home page about deals, maintenance time, etc. |
| 18 | Bike Rental Management | View current and upcoming rentings | Allows staff to view current and upcoming rentings of a specific location under that staff management. |
| 19 | Update rent status | Allows staff to update status of a renting (cancelled, renting, returned, extended). |
| 20 | Appoint bike to customer | Allows staff to appoint the appropriate bike for the customer. |
| 21 | View current rented motorbike | Allows customers to view current renting details (motorbike, status, time left, etc). |
| 22 | Cancel rent | Allows customers to cancel current rent or return bike earlier than expected (terms and conditions apply). |
| 23 | Reporting and Statistics | Export Renting Statistics | Admin or staff exports renting statistics (bookings, revenue) as CSV files. |
| 24 | Support and Feedback | Chat | Customers can open a chat session with a staff to answer questions, report problems, etc. |
| 25 | Feedback | Customers can give feedback to anything related to bike rental services through a form. |
| 26 | Report problem during rent | Staff receive reports about defective motorbikes from customers for handling. |
| 27 | Motorbike Browsing and Renting | Browse available motorbikes | Customer browses a list of all available motorbikes that they can rent. |
| 28 | View details of a motorbike | Customer selects a specific bike to view its details, manufacturer, color, model, etc.and rent extra items besides the bike, such as helmets, glasses, raincoats, etc. |
| 29 | Rent a motorbike | Customer rents a motorbike after selecting renting time and specific motorbike. |
| 30 | Pay for reservation | Customer pays a down payment. |
| 31 | View reservation summary | Customers can view their reservation summary and choose to apply discounts. |
| 32 | Select rent duration and location | Customer can choose start date, end date and location to rent motorbikes. |
| 33 | Commenting | Leave a comment on a motorbike | Customers can leave a comment on a specific bike and rate them to share their experiences. |

##### 

##### 2.2.2.1 Account Management

###### 2.2.2.1.1 Register

| UC ID and Name: | **UC-01 Register** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | User | Secondary Actors: |  |
| Trigger: | User click the Register button in the page header or  User click Sign up here link in the login page | | |
| Description: | This use case allow user to sign up a new account | | |
| Preconditions: | None | | |
| Postconditions: | POST-1: User register successfully  POST-2: A new user account has been saved in the database | | |
| Normal Flow: | **1.0**  1.0.1. User enter Full name, email address, date of birth, address, password and confirm password  1.0.2. System validates the inputs: Checks that all required fields are filled. Verifies that the email format is valid. Verifies that the password meets complexity requirements  1.0.3. If validation passes, the system creates a new user account.  1.0.4. The user successfully signs up. | | |
| Alternative Flows: | None | | |
| Exceptions: | **1.0.E1: Validation not passed. Displays error message.** - User leaves required fields blank: "Please fill out all required fields."  - Invalid email format: "Please enter a valid email address."  - Password doesn't meet complexity rules: "Password must be at least 8 characters and include letters and numbers."  - Email is already in use: "An account with this email already exists." | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | None | | |
| Other Information: | None | | |
| Assumptions: | None | | |

###### 2.2.2.1.2 Login

| UC ID and Name: | **UC-02 Login** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | User, Admin, Staff | Secondary Actors: |  |
| Trigger: | User clicks Login button from the page header | | |
| Description: | As a user, I want to be able to log into the system so that I can use the system’s authenticated features and access my personalized account. | | |
| Preconditions: | PRE-1: User account has been created & authorized | | |
| Postconditions: | POST-1: User logs in the system successfully  POST-2: The system tracked successful login into the Activity Log | | |
| Normal Flow: | **2.0**  2.0.1. User accesses the User Login screen  2.0.2. User types in the login details or choose other login options (see 4.2.1)  2.0.3. User clicks the Login button  2.0.4. System validates the login details  2.0.5. System allows user to access  2.0.6. System tracks user’s success login to the Activity Log  2.0.7. System directs user to Landing Page | | |
| Alternative Flows: | **2.1 Google Login**  2.1.1. User chooses to login system using Google account  2.1.2. System redirects the user to the Google’s Login screen  2.1.3. User types in the Google account details and chooses to login  2.1.4. Google validates user’s login information successfully and redirect him/her back to the system  2.1.5. Return to step 5 of normal flow. | | |
| Exceptions: | **2.0.E1 System can’t authenticate the user:** Occurs at step 2.0.4, the error message screen is shown to the user | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR3 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

###### 

###### 2.2.2.1.3 View Account

| UC ID and Name: | **UC-03 View Account** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | User | Secondary Actors: | Staff, Admin |
| Trigger: | When user click the Profile bar in the pop up window after clicking the profile icon | | |
| Description: | This use case allows user to view their account information | | |
| Preconditions: | PRE-1: The user already logged in | | |
| Postconditions: | POST-1: The account information is displayed to the user. | | |
| Normal Flow: | **3.0**  3.0.1. User click the profile icon  3.0.2. Use choose the Profile bar in the pop up window  3.0.3. System will retrieve information based on user’s id from database  3.0.4. System display user’s information | | |
| Alternative Flows: | None | | |
| Exceptions: | **3.0.E1 Account not found**: Occurs at step 3.0.2, the system displays an error if the account does not exist or has been deleted. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR1 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

###### 2.2.2.1.4 Edit Account Information

| UC ID and Name: | **UC-04 Edit Account Information** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | User | Secondary Actors: | Staff, Admin |
| Trigger: | When user click the My profile bar in the pop up window after clicking the profile icon | | |
| Description: | This use case allow user to update their account information | | |
| Preconditions: | PRE-1: The user already logged in | | |
| Postconditions: | POST-1: The user’s account information is updated | | |
| Normal Flow: | **4.0**  4.0.1. User click the profile icon  4.0.2. Use choose the My Profile bar in the pop up window  4.0.3. System will retrieve information based on user’s id from database  4.0.4. System display user’s current information  4.0.5. User click the Edit icon to start editing information  4.0.6. User choose which information to update in input bars and modify the information  4.0.7. User click Save button  4.0.8. System validates the input  4.0.9. If validation passes, system updated new information of user account in database  4.0.10. System displays a notice that user’s information has been updated | | |
| Alternative Flows: | **4.1 Validation fails**  - If validation fails, the system displays error messages highlighting the invalid fields and requests correction.  - Return to step 4.0.4. | | |
| Exceptions: | **4.0.E1 Account not found**: Occurs at step 4.0.2, the system displays an error if the account does not exist or has been deleted.  **4.0.E2 Validation Error**: Occurs at step 4.0.7, user submits invalid data, leading to system displaying error messages highlighting the invalid fields that require correction. | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR1, BR3 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

###### 2.2.2.1.5 Request Reset Password

| UC ID and Name: | **UC-05 Reset Password** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | User | Secondary Actors: | Staff, Admin |
| Trigger: | Users click the link Forget your password | | |
| Description: | This use case allow user to reset their account’s password if forgotten | | |
| Preconditions: | None | | |
| Postconditions: | POST-1: The user’s account password is reset | | |
| Normal Flow: | **5.0**  5.0.1. User clicks the link Forget your password in Login page  5.0.2. System direct user to Reset password page  5.0.3. User enter their email and click Confirm button  5.0.4. System send a password reset email to user’s email  5.0.5. User click a link in the mail, which directing them to a page where they can enter their new password | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR3 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

##### 2.2.2.2 User Management

###### 2.2.2.2.1 View User List

| UC ID and Name: | **UC-06 View User List** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Dinh Duy | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | System |
| Trigger: | An admin indicates that they want to see the list of registered users | | |
| Description: | An admin clicks on the user management section in the admin panel to be directed to the user list page, where they can see all registered users of the system. | | |
| Preconditions: | PRE-1. The admin logged into the system.  PRE-2. The user account has admin privileges. | | |
| Postconditions: | POST-1. Admin can see the list of users. | | |
| Normal Flow: | **6.0**  6.0.1. The Admin navigates to the User Management page.  6.0.2. The System retrieves the list of all users from the data store.  6.0.3. The System displays a list of users, typically including key information like name, email, and role. | | |
| Alternative Flows: | None | | |
| Exceptions: | **6.0.E1 System error loading user list:**  - Occur in step 6.0.1 or 6.0.2.  - The system displays an error message | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR1, BR2 | | |
| Other Information: | None | | |
| Assumptions: | The system has access to data store | | |

###### 2.2.2.2.2 View User Information

| UC ID and Name: | **UC-07 View User Information** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Dinh Duy | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | System |
| Trigger: | The Admin selects a specific user from the user list to view. | | |
| Description: | This use case describes how an Admin can view detailed information for a specific user account. | | |
| Preconditions: | PRE-1. The admin is viewing the user list (Use case 5 has been executed)  PRE-2. The admin selects a valid user from the list | | |
| Postconditions: | POST-1. Admin can see the detailed information of the selected user. | | |
| Normal Flow: | **7.0**  7.0.1. The Admin selects a user from the user list.  7.0.2. The System retrieves the detailed information for the selected user from the data store.  7.0.3. The System displays the detailed user information, which may include all available fields related to the user account, in an editable format (e.g., within a dialog or form). | | |
| Alternative Flows: | None | | |
| Exceptions: | **7.0.E1 User not found:**  - Occurs during step 7.0.1 if the selected user is no longer in the data store.  - The system displays a message indicating that the user could not be found.  **7.0.E2 System error loading user not found:**  - Occurs during step 7.0.1 or 7.0.2.  - The system displays a message indicating that the user information could not be displayed. | | |
| Priority: | High | | |
| Frequency of Use: | Moderate | | |
| Business Rules: | BR1, BR2 | | |
| Other Information: | Ensure sensitive user information be handled securely. | | |
| Assumptions: | The system can retrieve detailed information for a specific user. | | |

###### 2.2.2.2.3 Edit User Information

| UC ID and Name: | **UC-08 Edit User Information** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Dinh Duy | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | System |
| Trigger: | The Admin selects a specific user from the user list to edit. | | |
| Description: | This use case describes how an Admin can view detailed information for a specific user account and modify that information. | | |
| Preconditions: | PRE-1. The admin is viewing the user list (Use case 5 has been executed)  PRE-2. The admin selects a valid user from the list | | |
| Postconditions: | POST-1. Admin can see the detailed information of the selected user.  POST-2: User information is updated in the system if the Admin chooses to edit. | | |
| Normal Flow: | **8.0**  8.0.1. The Admin selects a user from the user list.  8.0.2. The System retrieves the detailed information for the selected user from the data store.  8.0.3. The System displays the detailed user information, which may include all available fields related to the user account, in an editable format (e.g., within a dialog or form).  8.0.4. The Admin can choose to close the detail view without making changes, or modify the user details.  8.0.5. If the Admin chooses to modify the user details:  8.0.5.1. The Admin makes changes to the user information in the provided form fields.  8.0.5.2. The Admin submits the updated user details.  8.0.5.3. The System validates the input.  8.0.5.4. If validation is successful, the System updates the user account with the new details in the data store.  8.0.5.5. The System refreshes the user list (either automatically or upon Admin action) to reflect the changes.  8.0.5.6. The System displays a success message confirming the update.  8.0.5.7. The System closes the detail view or form.  8.0.6. If the Admin closes the detail view without making changes, the use case ends with no changes to the user data. | | |
| Alternative Flows: | None | | |
| Exceptions: | **8.0.E1 User not found:**  - Occurs during step 8.0.2 if the selected user is no longer in the data store.  - The system displays a message indicating that the user could not be found.  **8.0.E2 System error loading user not found:**  - Occurs during step 8.0.2 or 8.0.3.  - The system displays a message indicating that the user information could not be displayed.  **8.0.E3 Invalid Input during User Editing:**  - Occurs during step 8.0.5.3.  - The System displays validation errors next to the input fields that failed validation.  - The System keeps the form open with the entered data, allowing the Admin to correct the input.  - The use case continues from step 8.0.5.2.  **8.0.E4 System Error during User Update:**  - Occurs during step 8.0.5.4.  - The System displays an error message indicating that the user could not be updated due to a system error. | | |
| Priority: | High | | |
| Frequency of Use: | Moderate | | |
| Business Rules: | BR1, BR2 | | |
| Other Information: | Ensure sensitive user information be handled securely. | | |
| Assumptions: | The system can retrieve detailed information for a specific user. | | |

###### 2.2.2.2.4 Delete a User

| UC ID and Name: | **UC-09 Delete a User** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Dinh Duy | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | System |
| Trigger: | The Admin initiates the deletion of a specific user from the user list. | | |
| Description: | This use case describes how an Admin can permanently delete a user account from the system. | | |
| Preconditions: | PRE-1. The admin is viewing the user list (Use case 5 has been executed)  PRE-2. The admin selects a valid user from the list to delete. | | |
| Postconditions: | POST-1: The selected user account is removed from the system.  POST-2: The Admin is presented with the updated user list (without the deleted user).  POST-3: A success message is displayed. | | |
| Normal Flow: | **9.0**  9.0.1. The Admin selects a user to delete from the user list.  9.0.2. The System displays a confirmation dialog, asking the Admin to confirm the deletion.  9.0.3. The Admin confirms the deletion.  9.0.4. The System initiates the process to delete the user account from the data store.  9.0.5. The System removes the user from the list displayed to the Admin.  9.0.6. The System displays a success message confirming the user has been deleted. | | |
| Alternative Flows: | **9.1 The admin canceled the user deletion.**  9.1.1. The admin cancels the user deletion in the confirmation dialog.  9.1.2. The system closes the confirmation dialog. | | |
| Exceptions: | **9.0.E1 User not found during deletion:**  - Occurs during step 9.0.3 if the selected user account is no longer in the data store (deleted by another admin concurrently).  - The system displays a message indicating that the user could not be found for deletetion. | | |
| Priority: | High | | |
| Frequency of Use: | Low to Moderate | | |
| Business Rules: | BR2, BR17, BR18 | | |
| Other Information: | Ensure the deletion process is permanent and irreversible from the Admin interface. | | |
| Assumptions: | The system has the capability to permanently remove a user account from the data store. | | |

### 

##### 2.2.2.3 Motorbike Management

###### 2.2.2.3.1 View Motorbike List

| UC ID and Name: | **UC-10 View Motorbike List** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Trigger: | Admin selects “View Motorbike List” from the management dashboard | | |
| Description: | As an admin, I want to view the list of all motorbikes in the system to manage inventory effectively. | | |
| Preconditions: | PRE-1: Admin is logged into the system with valid credentials.  PRE-2: Motorbike data exists in the system. | | |
| Postconditions: | POST-1: Admin views the complete list of motorbikes. | | |
| Normal Flow | **10.0**  10.0.1. Admin navigates to the Motorbike Management section.  10.0.2. Admin selects “View Motorbike List.”  10.0.3. System retrieves and displays the list of all motorbikes (including model, price, condition, etc.).  10.0.4. System logs the activity in the Activity Log. | | |
| Alternative Flows: | None | | |
| Exceptions: | **10.0.E1 No motorbikes exist in the system.**  - Occurs at step 10.0.3.  - System displays a message: “No motorbikes available”. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Daily | | |
| Business Rules: | BR2, BR20 | | |
| Other Information: | None | | |
| Assumptions: | Motorbike data is accurate and up-to-date | | |

###### 2.2.2.3.2 Edit Motorbike Information

| UC ID and Name: | **UC-11 Edit Motorbike Information** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Trigger: | Admin selects “Edit” on a specific motorbike from the motorbike list | | |
| Description: | As an admin, I want to edit motorbike information (model, price, condition, etc.) to keep the inventory updated. | | |
| Preconditions: | PRE-1: Admin is logged into the system with valid credentials.  PRE-2: The motorbike exists in the system. | | |
| Postconditions: | POST-1: Motorbike information is updated successfully.  POST-2: System logs the update in the Activity Log. | | |
| Normal Flow | **11.0**  11.0.1. Admin navigates to the Motorbike Management section and selects “View Motorbike List.”  11.0.2. Admin selects a motorbike and clicks “Edit.”  11.0.3. System displays the motorbike’s current details in an editable form.  11.0.4. Admin updates the details (e.g., model, price, condition).  11.0.5. Admin clicks “Save.”  11.0.6. System validates the input.  11.0.7. System updates the motorbike information in the database.  11.0.8. System logs the update in the Activity Log.  11.0.9. System redirects Admin to the motorbike list. | | |
| Alternative Flows: | None | | |
| Exceptions: | **11.0.E1: Invalid input data (e.g., negative price).**  **-** Occurs at step 11.0.6.  - System displays an error message: “Invalid input data.”  - Admin corrects the input and resubmits or cancels the action. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Weekly | | |
| Business Rules: | BR2, BR21 | | |
| Other Information: | None | | |
| Assumptions: | Admin has the authority to edit all motorbike details | | |

###### 2.2.2.3.3 Add New Motorbike

| UC ID and Name: | **UC-12 Add New Motorbike** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Trigger: | Admin selects “Add New Motorbike” from the Motorbike Management dashboard | | |
| Description: | As an admin, I want to add a new motorbike to the system to expand the inventory | | |
| Preconditions: | PRE-1: Admin is logged into the system with valid credentials | | |
| Postconditions: | POST-1: A new motorbike is added to the system. | | |
| Normal Flow | **12.0**  12.0.1. Admin navigates to the Motorbike Management section.  12.0.2. Admin selects “Add New Motorbike.”  12.0.3. System displays a form to input motorbike details (model, price, condition, etc.).  12.0.4. Admin fills in the details and clicks “Save.”  12.0.5. System validates the input (see 12.0.E1).  12.0.6. System adds the motorbike to the database.  12.0.7. System logs the addition in the Activity Log.  12.0.8. System redirects Admin to the motorbike list. | | |
| Alternative Flows: | None | | |
| Exceptions: | **12.0.E1: Invalid input data (e.g., missing model).**  - Occurs at step 12.0.5.  - System displays an error message: “Please provide all required fields.”  - Admin corrects the input and resubmits or cancels the action. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Monthly | | |
| Business Rules: | BR2, BR20 | | |
| Other Information: | None | | |
| Assumptions: | All required fields are clearly marked in the form. | | |

###### 2.2.2.3.4 Delete Motorbike

| UC ID and Name: | **UC-13 Delete Motorbike** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Trigger: | Admin selects “Delete” on a specific motorbike from the motorbike list. | | |
| Description: | As an admin, I want to delete a motorbike from the system to remove outdated or unsellable inventory | | |
| Preconditions: | PRE-1: Admin is logged into the system with valid credentials.  PRE-2: The motorbike exists in the system and is not currently rented. | | |
| Postconditions: | POST-1: The motorbike is removed from the system.  POST-2: System logs the deletion in the Activity Log | | |
| Normal Flow | **13.0**  13.0.1. Admin navigates to the Motorbike Management section and selects “View Motorbike List.”  13.0.2. Admin selects a motorbike and clicks “Delete.”  13.0.3. System prompts for confirmation: “Are you sure you want to delete this motorbike?”  13.0.4. Admin confirms the deletion.  13.0.5. System removes the motorbike from the database.  13.0.6. System logs the deletion in the Activity Log.  13.0.7. System redirects Admin to the motorbike list | | |
| Alternative Flows: | None | | |
| Exceptions: | **13.0.E1: Motorbike is currently rented.**  **-** Occurs at step 13.0.5.  - System displays an error message: “Cannot delete a motorbike that is currently rented.”  - Admin is redirected to the motorbike list | | |
| Priority: | Must Have | | |
| Frequency of Use: | Occasionally | | |
| Business Rules: | BR2, BR17 | | |
| Other Information: | None | | |
| Assumptions: | Admin understands the impact of deleting a motorbike. | | |

###### 2.2.2.3.5 Filter Motorbikes

| UC ID and Name: | **UC-14 Filter Motorbikes** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Staff | Secondary Actors: | None |
| Trigger: | Staff selects the “Filter” option from the motorbike list. | | |
| Description: | As a staff member, I want to filter the motorbike list by model, color, date, or price to find specific motorbikes quickly | | |
| Preconditions: | PRE-1: Staff is logged into the system with valid credentials.  PRE-2: Motorbike data exists in the system. | | |
| Postconditions: | POST-1: System displays the filtered motorbike list.  POST-2: System logs the filter activity in the Activity Log | | |
| Normal Flow | **14.0**  14.0.1. Staff navigates to the Motorbike Management section and selects “View Motorbike List.”  14.0.2. Staff selects the “Filter” option.  14.0.3. System displays filter options (model, color, date, price).  14.0.4. Staff inputs filter criteria and clicks “Apply.”  14.0.5. System validates the criteria (see 14.0.E1).  14.0.6. System displays the filtered motorbike list. | | |
| Alternative Flows: | None | | |
| Exceptions: | **14.0.E1: No motorbikes match the filter criteria.**  - Occurs at step 14.0.6.  - System displays a message: “No motorbikes match the selected criteria.”  - Staff clears the filter or applies new criteria. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Daily | | |
| Business Rules: | BR5, BR6 | | |
| Other Information: | None | | |
| Assumptions: | Filter options are intuitive and user-friendly. | | |

##### 2.2.2.4 Website Management

###### 2.2.2.4.1 View website settings

| ID and Name: | **UC-15 View website settings** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 04/06/25 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Description: | The admin accesses the website settings to review current configurations, such as site status, themes, or operational parameters. The use case concludes with the admin viewing the settings displayed on the screen. | | |
| Trigger: | Admin selects the "Website Settings" option from the Admin Panel dashboard. | | |
| Preconditions: | PRE-1: Admin’s identity has been authenticated.  PRE-2: Admin has the necessary permissions to access website settings.  PRE-3: The Admin Panel is operational and accessible. | | |
| Postconditions: | POST-1: The website settings are displayed to the admin.  POST-2: No changes are made to the settings unless further actions are taken. | | |
| Normal Flow: | **15.0**  15.0.1. Admin navigates to the "Website Settings" section in the Admin/Staff Panel.  15.0.2. System retrieves and displays the current website settings (e.g., site status, themes, operational hours).  15.0.3. Admin reviews the settings.  15.0.4. Admin exits the settings view. | | |
| Alternative Flows: | **No alternate flows for this use case.** | | |
| Exceptions: | **15.0.E1: System fails to retrieve settings due to database connectivity issues.**  - Occurs at step 15.0.2.  - System displays an error message: "Unable to load settings. Please try again later."  - Admin is returned to the dashboard. | | |
| Priority: | High (core functionality for website management). | | |
| Frequency of Use: | Approximately 5 times per week, depending on admin’s need to review settings. | | |
| Business Rules: | BR2 | | |
| Other Information: | 1. Quality Attributes: The system should load settings within 2 seconds under normal conditions. 2. Associated Functional Requirements: System must log all admin actions for audit purposes. 3. Failure Handling: If the use case fails (e.g., due to network issues), no changes are made to the system state, and the admin is returned to the dashboard. No durable state changes occur in this use case. | | |
| Assumptions: | The admin is familiar with the Admin Panel interface.  Website settings are stored in a centralized database accessible by the system. | | |

###### 2.2.2.4.2 Put website under maintenance

| ID and Name: | **UC-16 Put website under maintenance** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 04/06/25 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Description: | The admin initiates maintenance mode to temporarily disable user access to the website for updates or repairs. The use case concludes with the website being in maintenance mode and users being notified of the status. | | |
| Trigger: | Admin selects the "Put website under maintenance" option while viewing website settings. | | |
| Preconditions: | PRE-1: Admin’s identity has been authenticated.  PRE-2: Admin has the necessary permissions to modify website settings.  PRE-3: The website is currently operational (not already in maintenance mode).  PRE-4: Use Case 14 (View Website Settings) has been initiated. | | |
| Postconditions: | POST-1: The website is in maintenance mode.  POST-2: A maintenance notification is displayed to all users attempting to access the website.  POST-3: The website settings are updated to reflect the maintenance status. | | |
| Normal Flow: | **16.0**  16.0.1. Admin selects "Put website under maintenance" from the website settings view.  16.0.2. System prompts the admin to confirm the action and optionally specify a maintenance message and duration.  16.0.3. Admin confirms the action and submits the details.  16.0.4. System updates the website status to maintenance mode.  16.0.5. System displays a confirmation message to the admin: "Website is now in maintenance mode."  16.0.6. System ensures all users see a maintenance notification when accessing the website. | | |
| Alternative Flows: | **16.1: Admin cancels the maintenance action.**  16.1.1 Branches at step 3 of Normal Flow (16.0.3).  16.1.2 Admin selects "Cancel" instead of confirming.  16.1.3 System returns the admin to the website settings view without making changes.  16.1.4 Rejoins Normal Flow at step 1 of Use Case 1 (View Website Settings). | | |
| Exceptions: | **16.0.E1: System fails to update the website status due to a server error.**  - Occurs at step 4 of Normal Flow (16.0).  - System displays an error message: "Failed to enable maintenance mode. Please try again."  - Website remains operational, and admin is returned to the settings view. | | |
| Priority: | Medium (important for system updates but not a daily operation). | | |
| Frequency of Use | Approximately 1-2 times per month, depending on maintenance needs. | | |
| Business Rules: | BR2, BR15 | | |
| Other Information: | 1. Quality Attributes: The transition to maintenance mode should complete within 5 seconds. 2. Associated Functional Requirements: System must log all website status changes. 3. Failure Handling: If the use case fails, the website remains operational (no state change occurs). If a partial update occurs, the system rolls back the status to ensure consistency. The change is durable once maintenance mode is successfully enabled. | | |
| Assumptions | - Users are automatically logged out when maintenance mode is enabled.  - The admin can specify a maintenance message, though this is optional. | | |

###### 2.2.2.4.3 Add website announcement

| ID and Name: | **UC-17 Add website announcement** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 04/06/25 |
| Primary Actor: | Admin | Secondary Actors: | None |
| Description: | The admin creates and publishes an announcement visible to all website users, such as for promotions or service updates. The use case concludes with the announcement being live on the website. | | |
| Trigger: | Admin selects the "Add announcement to the entire website" option while viewing website settings. | | |
| Preconditions: | PRE-1: Admin’s identity has been authenticated.  PRE-2: Admin has the necessary permissions to modify website settings.  PRE-3: The website is operational and not in maintenance mode.  PRE-4: Use Case 1 (View Website Settings) has been initiated. | | |
| Postconditions: | POST-1: The announcement is published and visible to all website users.  POST-2: The website settings are updated to include the new announcement. | | |
| Normal Flow: | **17.0**  17.0.1. Admin selects "Add announcement to the entire website" from the website settings view.  17.0.2. System displays a form for the admin to enter the announcement details (e.g., title, message, duration).  17.0.3. Admin fills out the form and submits the announcement.  17.0.4. System validates the announcement details (e.g., checks for empty fields).  17.0.5. System publishes the announcement to the website.  17.0.6. System displays a confirmation message to the admin: "Announcement has been published."  17.0.7. Announcement is visible to all users on the website. | | |
| Alternative Flows: | **17.1: Admin cancels the announcement creation.**  17.1.1. Branches at step 3 of Normal Flow (17.0).  17.1.2. Admin selects "Cancel" instead of submitting.  17.1.3. System discards the form data and returns the admin to the website settings view.  17.1.4. Rejoins Normal Flow at step 1 of UC-15 (View Website Settings). | | |
| Exceptions: | **17.0.E1: Validation fails due to missing required fields.**  - Occurs at step 17.0.4.  - System displays an error message: "Please fill in all required fields."  - Admin is prompted to correct the form and resubmit.  **17.0.E2: System fails to publish the announcement due to a server error.**  - Occurs at step 17.0.5.  - System displays an error message: "Failed to publish announcement. Please try again."  - No announcement is published, and admin is returned to the settings view. | | |
| Priority: | Medium (useful for communication but not critical to core operations). | | |
| Frequency of Use | Approximately 2-3 times per week, depending on the need for announcements. | | |
| Business Rules: | BR2, BR15 | | |
| Other Information: | - Quality Attributes: The announcement should be published within 3 seconds.  - Associated Functional Requirements: System must log all announcement publications.  - Failure Handling: If the use case fails, no announcement is published (no state change occurs). If a partial update occurs, the system rolls back to ensure no incomplete announcement is displayed. The change is durable once the announcement is successfully published. | | |
| Assumptions | - The announcement form includes fields for title, message, and duration.  - Announcements are automatically removed after the specified duration, handled by a separate system process. | | |

##### 2.2.2.5 Bike Rental Management

###### 2.2.2.5.1 View Current and Upcoming Rentings

| UC ID and Name: | **UC-18 View Current and Upcoming Rentings** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Staff | Secondary Actors: | None |
| Trigger: | Staff selects “Rental Management” from the Staff dashboard. | | |
| Description: | As a staff member, I want to view current and upcoming rentings for a specific location to manage operations effectively. | | |
| Preconditions: | PRE-1: Staff is logged into the system with valid credentials.  PRE-2: Rental data exists for the staff’s managed location | | |
| Postconditions: | POST-1: Staff views the list of current and upcoming rentings.  POST-2: System logs the activity in the Activity Log | | |
| Normal Flow | **18.0**  18.0.1. Staff navigates to the Rental Management section.  18.0.2. System retrieves and displays current and upcoming rentings for the staff’s location.  18.0.3. System logs the activity in the Activity Log. | | |
| Alternative Flows: | None | | |
| Exceptions: | **18.0.E1: No rentings exist for the location.**  **-** Occurs at step 18.0.3.  - System displays a message: “No current or upcoming rentings available”.  - Staff is redirected to the Bike Rental Management dashboard. | | |
| Priority: | Must Have | | |
| Frequency of Use: | High | | |
| Business Rules: | BR6,BR7,BR29 | | |
| Other Information: | None | | |
| Assumptions: | Staff is assigned to a specific location | | |

##### 

###### 2.2.2.5.2 Update Rent Status

| UC ID and Name: | **UC-19 Update Rent Status** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Staff | Secondary Actors: | None |
| Trigger: | Staff selects “Update Status” for a specific renting from the rentings list. | | |
| Description: | As a staff member, I want to update the status of a renting (cancelled, renting, returned, extended) to reflect its current state | | |
| Preconditions: | PRE-1.Staff is logged into the system with valid credentials.  PRE-2.The renting exists in the system. | | |
| Postconditions: | POST-1.Renting status is updated successfully.  POST-2.System logs the update in the Activity Log. | | |
| Normal Flow | **19.0**  19.0.1. Staff navigates to the Bike Rental Management section and selects “View Rentings.”  19.0.2. Staff selects a location and clicks “Update Status.”  19.0.3. System displays status options (cancelled, renting, returned, extended).  19.0.4. Staff selects a new status and clicks “Save.”  19.0.5. System validates the status change (see 19.0.E1).  19.0.6. System updates the renting status in the database.  19.0.7. System logs the update in the Activity Log.  19.0.8. System redirects Staff to the rentings list. | | |
| Alternative Flows: | None | | |
| Exceptions: | **19.0.E1: Invalid status transition (e.g., extending a cancelled rent).**  - Occurs at step 19.0.5.  - System displays an error message: “Invalid status transition.”  - Staff selects a valid status or cancels the action. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Daily | | |
| Business Rules: | BR6,BR7 | | |
| Other Information: | None | | |
| Assumptions: | Status transitions follow predefined rules. | | |

###### 2.2.2.5.3 Appoint Bike to Customer

| UC ID and Name: | **UC-20 Appoint Bike to Customer** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Staff | Secondary Actors: | Customer |
| Trigger: | Staff selects “Appoint Bike” for a customer’s rental request. | | |
| Description: | As a staff member, I want to appoint an appropriate motorbike to a customer to fulfill their rental request | | |
| Preconditions: | PRE-1: Staff is logged into the system with valid credentials.  PRE-2: Customer has a valid rental request.  PRE-3: Available motorbikes exist in the system | | |
| Postconditions: | POST-1: A motorbike is appointed to the customer.  POST-2: System logs the appointment in the Activity Log. | | |
| Normal Flow | **20.0**  20.0.1. Staff navigates to the Bike Rental Management section and selects “View Current and Upcoming Rentings” (UC-18).  20.0.2. Staff selects a customer’s rental request and clicks “Appoint Bike.”  20.0.3. System displays a list of available motorbikes.  20.0.4. Staff selects a motorbike and clicks “Confirm.”  20.0.5. System assigns the motorbike to the customer’s rental.  20.0.6. System logs the appointment in the Activity Log.  20.0.7. System redirects Staff to the rentings list. | | |
| Alternative Flows: | None | | |
| Exceptions: | **20.0.E1: No motorbikes are available.**  - Occurs at step 20.0.5.  - System displays a message: “No motorbikes available for appointment.”  - Staff is redirected to the rentings list. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Daily | | |
| Business Rules: | BR7 | | |
| Other Information: | None | | |
| Assumptions: | Customer’s rental request is valid and approved. | | |

###### 2.2.2.5.4 View Current Rented Motorbike

| UC ID and Name: | **UC-21 View Current Rented Motorbike** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Customer | Secondary Actors: | Customer |
| Trigger: | customer selects “View Current Rental” from their account dashboard. | | |
| Description: | As a customer, I want to view details of my current motorbike rental (motorbike, status, time left, etc.) to stay informed. | | |
| Preconditions: | PRE-1: Customer is logged into the system with valid credentials.  PRE-2: Customer has an active rental. | | |
| Postconditions: | POST-1: Customer views the rental details.  POST-2: System logs the activity in the Activity Log. | | |
| Normal Flow | **21.0**  21.0.1. Customers navigate to their account dashboard.  21.0.2. Customer selects “View Current Rental.”  21.0.3. System retrieves and displays the rental details (motorbike, status, time left, etc.).  21.0.4. System logs the activity in the Activity Log. | | |
| Alternative Flows: | None | | |
| Exceptions: | **21.0.E1: Customer has no active rental.**  - Occurs at step 21.0.3.  - System displays a message: “No active rentals found.”  - Customer is redirected to the account dashboard. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Daily | | |
| Business Rules: | BR8 | | |
| Other Information: | None | | |
| Assumptions: | Rental details are accurate and up-to-date | | |

###### 2.2.2.5.5 Cancel Rent

| UC ID and Name: | **UC-22 Cancel Rent** | | |
| --- | --- | --- | --- |
| Created By: | Truong Ngoc Viet | Date Created: | 04/06/2025 |
| Primary Actor: | Customer | Secondary Actors: | None |
| Trigger: | Customer selects “Cancel Rent” from the current rental details. | | |
| Description: | As a customer, I want to cancel my current rental or return the motorbike earlier than expected, subject to terms and conditions | | |
| Preconditions: | PRE-1: Customer is logged into the system with valid credentials.  PRE-2: Customer has an active rental.  PRE-3: Cancellation complies with terms and conditions. | | |
| Postconditions: | POST-1: The rental is cancelled or marked as returned.  POST-2: System logs the cancellation in the Activity Log. | | |
| Normal Flow | **22.0**  22.0.1. Customer navigates to their account dashboard and selects “View Current Rental.”  22.0.2. Customer selects “Cancel Rent.”  22.0.3. System prompts for confirmation: “Are you sure you want to cancel this rental?”  22.0.4. Customer confirms the cancellation.  22.0.5. System validates the cancellation request.  22.0.6. System updates the rental status to “cancelled” or “returned.”  22.0.7. System logs the cancellation in the Activity Log.  22.0.8. System redirects Customer to the account dashboard. | | |
| Alternative Flows: | None | | |
| Exceptions: | **22.0.E1: Cancellation violates terms and conditions (e.g., too late to cancel).**  - Occurs at step 22.0.5.  - System displays an error message: “Cancellation not allowed per terms and conditions.”  - Customer is redirected to the rental details. | | |
| Priority: | Must Have | | |
| Frequency of Use: | Daily | | |
| Business Rules: | BR9, BR10 | | |
| Other Information: | None | | |
| Assumptions: | Terms and conditions are clearly communicated to the customer. | | |

### 

##### 2.2.2.6 Reporting and Statistics

###### 2.2.2.6.1 Export Statistics as CSV files

| UC ID and Name: | **UC-23 Export Statistics as CSV files** | | |
| --- | --- | --- | --- |
| Created By: | Bui Khac Bin | Date Created: | 04/06/2025 |
| Primary Actor: | Staff, Admin | Secondary Actors: | None |
| Trigger: | A staff or the admin indicates that they want to export statistics as CSV files. | | |
| Description: | The initiator goes to their managing site, selects the range of dates they want statistics of, then export them to CSV files which they can download. | | |
| Preconditions: | PRE-1. The initiator must have the proper privileges, in this case, they have to be either a staff or an admin. | | |
| Postconditions: | POST-1. The statistics are gathered by the System, then sent back to the initiator. | | |
| Normal Flow: | **23.0**  23.0.1. The staff or admin accesses their privileged managing site.  23.0.2. They choose to view statistics.  23.0.3. They choose a specific time span, then clicks on the button indicating they want to export as CSV.  23.0.4. The System gathers statistics of the required time span and sends them back to the staff or admin. | | |
| Alternative Flows: | None | | |
| Exceptions: | **23.0.E1. The required time span is invalid, the starting date is after the ending date.**  - Occurs at step 23.0.3.  - System displays that the required time span is invalid, then prompts the initiator to try again. | | |
| Priority: | High | | |
| Frequency of Use: | Each staff exports statistics once every week. The admin exports statistics every two days. | | |
| Business Rules: | BR12, BR13 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

###### 

##### 2.2.2.7 Support and Feedback

###### 2.2.2.7.1 Chat

| UC ID and Name: | **UC-24 Chat** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 06/06/2025 |
| Primary Actor: | User, Staff | Secondary Actors: | None |
| Trigger: | - Users click the chat icon in page header.  - Staffs click the Support Customer with chat icon . | | |
| Description: | - The customer initiates a chat session to ask questions, report issues, or seek assistance.  - A staff or an AI chatbot is assigned to this chat session to answer customer’s questions.  - The use case concludes with the customer either resolving their query through the chat or escalating the issue for further handling by staff. | | |
| Preconditions: | PRE-1: Customer/Staff identity has been authenticated (logged in).  PRE-2: The website is operational and not in maintenance mode.  PRE-3: At least one staff member is available to respond, or an AI chatbot is enabled. | | |
| Postconditions: | POST-1: A chat session is created and logged in the system.  POST-2: The customer’s query is addressed, or the issue is escalated to staff for further action. | | |
| Normal Flow: | **24.0**  24.0.1. Customer clicks the "Chat with Support" button on the website.  24.0.2. System opens a chat window and prompts the customer to choose between AI or staff support (see Alternate flow 24.1 and 24.2). | | |
| Alternative Flows: | **24.1 User chat with staff**  24.1.1. System assigns an available staff member and notifies the customer: "You are now connected with Staff Member X."  24.1.2. Customer types their query or issue (e.g., "My rented motorbike has a flat tire").  24.1.3. System delivers the message to the staff member.  24.1.4. Staff responds (e.g., "Thank you for reporting. I’ll escalate this to our team.").  24.1.5. Customer and staff continue the conversation as needed.  24.1.6. Customer or staff ends the chat session.  24.1.7. System logs the chat session and marks it as closed.  **24.2 User chat with AI**  24.2.1. System connects the customer to an AI chatbot.  24.2.2. AI responds to the customer’s query based on predefined responses (e.g., "Please describe the issue with your motorbike.").  24.2.3. If AI cannot resolve the query, it offers to escalate to staff, rejoining at step 24.1.1. | | |
| Exceptions: | **24.1.E1: No staff member is available to chat.**  - Occurs at step 24.1.1.  - System displays a message: "No staff available. Please try again later or use AI support."  - Customer can switch to AI or exit the chat.  **24.1.E2, 24.2.E1: Network connectivity is lost during the chat.**  - Occurs at any step after step 24.1.1 or step 24.2.1.  - System displays: "Connection lost. Please reconnect."  - Chat session is saved, and customer can resume once reconnected. | | |
| Priority: | High (critical for customer support and satisfaction). | | |
| Frequency of Use: | Approximately 10 times per day, based on expected customer inquiries and issues during rentals. | | |
| Business Rules: | BR2, BR22 | | |
| Other Information: | - Quality Attributes: Chat messages should be delivered within 1 second under normal conditions.  - Associated Functional Requirements: System must support real-time chat with message persistence.  - Failure Handling: If the use case fails (e.g., due to network issues), the chat session is saved up to the last successful message. No durable state changes occur until the chat is closed, at which point the session is logged. If the session is interrupted, the customer can resume the chat later. | | |
| Assumptions: | - The system supports both AI and staff-driven chat.  - Staff members are trained to handle customer queries and escalate issues as needed. | | |

###### 2.2.2.7.2 Feedback

| UC ID and Name: | **UC-25 Feedback** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 06/06/2025 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | Customer selects the "Give Feedback" option from the website’s support or dashboard page. | | |
| Description: | The customer submits feedback about their experience with the bike rental service through a form. The use case concludes with the feedback being recorded in the system for review by staff or admin. | | |
| Preconditions: | PRE-1: Customer’s identity has been authenticated (logged in).  PRE-2: The website is operational and not in maintenance mode. | | |
| Postconditions: | POST-1: The customer’s feedback is recorded in the system.  POST-2: The customer receives a confirmation of submission. | | |
| Normal Flow: | **25.0**  25.0.1. Customer navigates to the "Give Feedback" section on the website.  25.0.2. System displays a feedback form with fields for rating (1-5 stars) and comments.  25.0.3. Customer enters a rating and optional comments (e.g., "Great service, but the bike was a bit noisy").  25.0.4. Customer clicks "Submit Feedback".  25.0.5. System validates the form (ensures rating is provided).  25.0.6. System saves the feedback in the database.  25.0.7. System displays a confirmation message: "Thank you for your feedback!" | | |
| Alternative Flows: | **25.1 Customer cancels feedback submission.**  - Branches at step 25.0.4.  25.1.1. Customer clicks "Cancel" instead of "Submit Feedback".  25.1.2. System discards the form data and returns the customer to the previous page. | | |
| Exceptions: | **25.0.E1: Form validation fails due to missing rating.**  - Occurs at step 25.0.5.  - System displays an error message: "Please provide a rating."  - Customer is prompted to correct the form and resubmit.  **25.0.E2: System fails to save feedback due to database error.**  - Occurs at step 25.0.6.  - System displays an error message: "Failed to submit feedback. Please try again."  - Feedback is not saved, and customer can retry. | | |
| Priority: | Medium (important for gathering customer insights but not critical to core operations). | | |
| Frequency of Use: | Approximately 10 times per day, based on customer engagement. | | |
| Business Rules: | BR23, BR24 | | |
| Other Information: | - Quality Attributes: Feedback submission should complete within 2 seconds.  - Failure Handling: If the use case fails, no feedback is saved (no state change occurs). The customer can retry submission. The change is durable once feedback is successfully saved. | | |
| Assumptions: | - The feedback form is accessible from multiple pages (e.g., dashboard, support page).  - Feedback is reviewed periodically by staff or admin. | | |

###### 2.2.2.7.3 Report problem during rent

| UC ID and Name: | **UC-26 Report problem during rent** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 06/06/2025 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | Customer selects the "Report Problem" option from their current rental details page or during a chat session. | | |
| Description: | The customer reports a problem with their rented motorbike (e.g., a defect). The use case concludes with the problem being logged as a support ticket and staff being notified for handling. | | |
| Preconditions: | PRE-1: Customer’s identity has been authenticated (logged in).  PRE-2: The customer has an active rental in progress.  PRE-3: The website is operational and not in maintenance mode. | | |
| Postconditions: | POST-1: A support ticket is created with the problem details.  POST-2: Staff are notified of the reported problem.  POST-3: The customer receives confirmation of the report submission. | | |
| Normal Flow: | **26.0**  26.0.1. Customer navigates to their current rental details page and clicks "Report Problem".  26.0.2. System displays a form with fields for problem description and urgency level (Low, Medium, High).  26.0.3. Customer fills out the form (e.g., "Flat tire, High urgency").  26.0.4. Customer submits the form.  26.0.5. System validates the form (ensures description is provided).  26.0.6. System creates a support ticket linked to the rental and customer.  26.0.7. System notifies available staff via an internal alert (e.g., email or dashboard notification).  26.0.8. System displays a confirmation to the customer: "Problem reported. Our team will assist you shortly." | | |
| Alternative Flows: | **26.1 Customer reports the problem via chat instead of the form.**  - Branches at step 26.0.1.  26.1.1. Customer initiates a chat session (Use Case 24) and reports the problem during the conversation.  26.1.2. Staff logs the problem as a support ticket on behalf of the customer.  26.1.3. Rejoins Normal Flow at step 26.0.7. | | |
| Exceptions: | **26.0.E1: Form validation fails due to missing description.**  - Occurs at step 26.0.5.  - System displays: "Please provide a problem description."  - Customer corrects the form and resubmits.  **26.0.E2: System fails to notify staff due to notification service downtime.**  - Occurs at step 26.0.7.  - System logs the ticket but displays: "Problem reported, but staff notification failed. Please contact support if not addressed soon."  - Staff can still see the ticket in the system manually. | | |
| Priority: | High | | |
| Frequency of Use: | Approximately 5 times per day, based on expected rental issues. | | |
| Business Rules: | BR6, BR25, BR26 | | |
| Other Information: | - Quality Attributes: Ticket creation and notification should complete within 3 seconds.  - Failure Handling: If the use case fails, the ticket is either not created (rolled back) or created but not notified (staff can still see it in the system). The change is durable once the ticket is created. | | |
| Assumptions: | - The customer can only report problems for active rentals.  - Staff have a dashboard to view and manage support tickets. | | |

##### 2.2.2.8 Motorbike Browsing and Renting

###### 2.2.2.8.1 Browse available motorbikes

| UC ID and Name: | **UC-27 Browse available motorbikes** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | User clicks the Browse Bikes buttons after selecting start date, and end date in the landing page | | |
| Description: | This use case allows users to view models available during the rent duration the user has selected before. Users can view general information about models: model name, rating, rent price per day, brief description… . Users can filter by type and cylinder volume. Users can reset filters or reset duration. | | |
| Preconditions: | PRE-1: User already selected the start date, end date in the landing page | | |
| Postconditions: | POST-1: User can see all models available during rent duration | | |
| Normal Flow: | **27.0**  27.0.1. User queries a list of available models during rent duration the user has already selected before.  27.0.2. System sends back a list of models.  27.0.3. System displays a list of models and rental details.  27.0.4. Users scroll down to see more models.  27.0.5. User selects a specific model.  27.0.6. System redirects to a detailed page of that model (see [UC-28](#_ut25pdfrartp)). | | |
| Alternative Flows: | **27.1 User filters motorbikes**  - Branches at step 27.0.4.  27.1.1. User filters by types, cylinder volume, etc.  27.1.2. System processes filters and displays appropriate models.  27.1.3. Rejoins Normal Flow at step 27.0.5.  **27.2 User resets filters**  - Branches after step 27.1.1.  27.2.1. System clears all filters and displays all available models.  27.2.2. Rejoins Normal Flow at step 27.0.3. | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR20 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

###### 2.2.2.8.2 View details of a motorbike

| UC ID and Name: | **UC-28 View details of a motorbike** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | User clicks the View Details button when viewing list of available models. | | |
| Description: | This use case allows users to view details of a model and rent if already logged in. | | |
| Preconditions: | PRE-1: Users already selected rental dates and are viewing list of available models | | |
| Postconditions: | POST-1: The model id will be sent to system  POST-2: System will send back details information of the model | | |
| Normal Flow: | **28.0**  28.0.1. System displays detail information of a model (name, rating, description, features, number of bikes available) | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | High | | |
| Frequency of Use: | High | | |
| Business Rules: | BR-31 | | |
| Other Information: | None | | |
| Assumptions: | None | | |

###### 2.2.2.8.3 Rent a motorbike

| UC ID and Name: | **UC-29 Rent a motorbike** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 06/06/2025 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | Customer clicks "Rent Now" on the motorbike details page after selecting a motorbike. | | |
| Description: | The customer rents a motorbike by specifying the rental duration and location, leading to the creation of a rental reservation. The use case concludes with the customer being directed to the payment step. | | |
| Preconditions: | PRE-1: Customer’s identity has been authenticated (logged in).  PRE-2: Customer has a verified driver’s license.  PRE-3: The selected motorbike is available for the requested time period.  PRE-4: The website is operational and not in maintenance mode. | | |
| Postconditions: | POST-1: A rental reservation is created in the system with a "Pending Payment" status.  POST-2: The customer is redirected to the payment step (Use Case 30). | | |
| Normal Flow: | **29.0**  29.0.1. Customer clicks "Rent Now" on the motorbike details page.  29.0.2. System displays a rental form with fields for start date, end date, location, and optional extras (e.g., helmet, raincoat).  29.0.3. Customer fills out the form (e.g., "Start: 2025-06-07, End: 2025-06-09, Location: Ha Noi Center").  29.0.4. Customer submits the form.  29.0.5. System checks motorbike availability for the selected dates and location.  29.0.6. System calculates the total cost, including extras.  29.0.7. System creates a rental reservation with "Pending Payment" status.  29.0.8. System redirects the customer to the payment page (Use Case 30). | | |
| Alternative Flows: | **29.1 Customer cancels the rental process.**  - Branches at step 29.0.4.  29.1.1. Customer clicks "Cancel" instead of submitting.  29.1.2. System discards the form data and returns the customer to the motorbike details page. | | |
| Exceptions: | **29.0.E1: Motorbike is no longer available due to a concurrent booking.**  - Occurs at step 29.0.5.  - System displays: "Sorry, this motorbike is no longer available. Please choose another."  - Customer is returned to the motorbike browsing page.  **29.0.E2: System fails to create the reservation due to database error.**  - Occurs at step 29.0.7.  - System displays: "Failed to create reservation. Please try again."  - No reservation is created, and customers can retry.  **29.0.E3: Customer has not uploaded a valid driver license.**  - Occurs at step 29.0.1.  - System displays: “Please upload and verify your driver license before renting a motorbike”.  - The “Rent now” button is blocked, forcing the customer to upload and verify their license. | | |
| Priority: | High (core functionality of the rental service). | | |
| Frequency of Use: | Approximately 100 times per day, based on expected rental volume. | | |
| Business Rules: | BR2, BR4, BR9 | | |
| Other Information: | - Quality Attributes: Reservation creation should complete within 3 seconds.  - Failure Handling: If the use case fails, no reservation is created (rolled back). The customer can retry. The change is durable once the reservation is created. | | |
| Assumptions: | - The rental form includes validation for date ranges (e.g., end date must be after start date).  - The system supports multiple rental locations. | | |

###### 2.2.2.8.4 Pay for reservation

| UC ID and Name: | **UC-30 Pay for reservation** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 06/06/2025 |
| Primary Actor: | Customer | Secondary Actors: | Third-Party Payment Provider |
| Trigger: | Customer is redirected to the payment page after creating a rental reservation (see [UC-29](#_pvfazrdypecx)). | | |
| Description: | The customer pays a down payment for their motorbike rental reservation using a third-party payment provider. The use case concludes with the payment being confirmed and the reservation status updated. | | |
| Preconditions: | PRE-1: Customer’s identity has been authenticated (logged in).  PRE-2: A rental reservation exists with "Pending Payment" status (from Use Case 29).  PRE-3: The website is operational and not in maintenance mode.  PRE-4: The third-party payment provider is accessible. | | |
| Postconditions: | POST-1: The rental reservation status is updated to "Reserved".  POST-2: A payment record is created in the system.  POST-3: The customer is redirected to the reservation summary page (see [UC-31](#_pi65sjc91dap)). | | |
| Normal Flow: | **30.0**  30.0.1. Customer is redirected to the payment page after creating a reservation.  30.0.2. System displays the total amount and down payment required (e.g., "Total: $100, Down Payment: $20").  30.0.3. System loads the third-party payment form (e.g., card details, billing info).  30.0.4. Customer enters payment details and submits the payment.  30.0.5. System sends the payment request to the third-party payment provider.  30.0.6. Payment provider confirms the transaction as successful.  30.0.7. System updates the reservation status to "Reserved".  30.0.8. System creates a payment record in the database.  30.0.9. System redirects the customer to the reservation summary page (see [UC-31](#_pi65sjc91dap)). | | |
| Alternative Flows: | **30.1 Customer cancels the payment process.**  - Branches at step 30.0.4.  30.1.1. Customer clicks "Cancel" instead of submitting payment.  30.1.2. System marks the reservation as "Cancelled" and returns the customer to the motorbike browsing page. | | |
| Exceptions: | **30.0.E1: Payment fails due to invalid card details or insufficient balance.**  - Occurs at step 30.0.6.  - System displays: "Payment failed. Please check your card details or balance and try again."  - Customer can retry with corrected details or after they have enough balance.  **30.0.E2: Payment provider is unavailable.**  - Occurs at step 30.0.5.  - System displays: "Payment service unavailable. Please try again later."  - Reservation remains in "Pending Payment" status, and customer can retry. | | |
| Priority: | High (essential for completing rentals). | | |
| Frequency of Use: | Approximately 100 times per day, aligned with rental reservations. | | |
| Business Rules: | BR27, BR28, BR29 | | |
| Other Information: | - Quality Attributes: Payment processing should complete within 5 seconds.  - Failure Handling: If the use case fails, the reservation remains in "Pending Payment" (no state change). If payment partially processes but fails, the transaction is rolled back by the payment provider. The change is durable once the reservation status is updated. | | |
| Assumptions: | - The down payment is a percentage of the total rental cost.  - The third-party payment provider handles card validation and fraud checks. | | |

###### 2.2.2.8.5 View reservation summary

| UC ID and Name: | **UC-31 View reservation summary** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Minh Hieu | Date Created: | 06/06/2025 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | Customer completes payment for a reservation (see [UC-30](#_w4tkmlk4h1jz)). | | |
| Description: | The customer views a summary of their rental reservation, including motorbike details, dates, location, and cost, with an option to apply a discount. The use case concludes with the customer confirming the reservation details. | | |
| Preconditions: | PRE-1: Customer’s identity has been authenticated (logged in).  PRE-2: The customer has a reservation with "Reserved" status (from [UC-30](#_w4tkmlk4h1jz)).  PRE-3: The website is operational and not in maintenance mode. | | |
| Postconditions: | POST-1: The customer has viewed the reservation summary.  POST-2: If a discount is applied, the reservation cost is updated in the system. | | |
| Normal Flow: | **31.0**  31.0.1. Customer is redirected to the reservation summary page after payment (or navigates to "My Reservations").  31.0.2. System displays the reservation summary: motorbike details (model, color), dates, location, total cost, and down payment status.  31.0.3. System shows an option to "Apply Discount" if applicable.  31.0.4. Customer reviews the summary and clicks "Confirm Reservation".  31.0.5. System confirms the reservation details and displays: "Reservation Confirmed!" | | |
| Alternative Flows: | **31.1 Customer applies a discount.**  - Branches at step 30.0.4.  31.1.1. Customer clicks "Apply Discount" before confirming.  31.1.2. System prompts for a discount code.  31.1.3. Customer enters a discount code (e.g., "RENT10").  31.1.4. System validates the code and applies the discount (e.g., 10% off).  31.1.5. System updates the total cost and displays the new amount.  31.1.6. Rejoins Normal Flow at step 30.0.4 (customer confirms the updated reservation). | | |
| Exceptions: | **31.1.E1: Invalid discount code entered.**  - Occurs at step 31.1.4 during code validation.  - System displays: "Invalid discount code. Please try again."  - Customers can retry or proceed without a discount.  **31.0.E1: System fails to load reservation details due to database error.**  - Occurs at step 31.0.2.  - System displays: "Failed to load reservation. Please try again."  - Customer is returned to the dashboard. | | |
| Priority: | High (necessary for completing the rental process). | | |
| Frequency of Use: | Approximately 100 times per day, aligned with rental reservations. | | |
| Business Rules: | BR27, BR28, BR29, BR30 | | |
| Other Information: | - Quality Attributes: Reservation summary should load within 2 seconds.  - Failure Handling: If the use case fails, no changes are made to the reservation. If a discount is applied but the use case fails afterward, the discount is rolled back. The change is durable once the discount is applied. | | |
| Assumptions: | - Discount codes are managed by admins and stored in a separate system table.  - The customer can only apply one discount per reservation. | | |

###### 2.2.2.8.6 Select rent duration and location

| UC ID and Name: | **UC-32 Select rent duration** | | |
| --- | --- | --- | --- |
| Created By: | Nguyen Tran Phu Quoc | Date Created: | 04/06/2025 |
| Primary Actor: | Customer | Secondary Actors: | None |
| Trigger: | Customers fill out the rental form during the "Rent a Motorbike" process (see [UC-29](#_pvfazrdypecx)). | | |
| Description: | The customer specifies the start date, end date, and location for renting a motorbike as part of the rental process. This use case is embedded within [Use Case 29](#_pvfazrdypecx) and concludes with the rental details being saved for further processing. | | |
| Preconditions: | PRE-1: Customer’s identity has been authenticated (logged in).  PRE-2: Customer has a verified driver’s license.  PRE-3: The customer has selected a motorbike to rent ([Use Case 29](#_pvfazrdypecx) has been initiated).  PRE-4: The website is operational and not in maintenance mode. | | |
| Postconditions: | POST-1: The rental duration and location are saved as part of the rental reservation process.  POST-2: The customer proceeds to the next step in the rental process (cost calculation in [Use Case 29](#_pvfazrdypecx)). | | |
| Normal Flow: | **32.0**  32.0.1. Customer is presented with the rental form during the "Rent a Motorbike" process (step 29.0.2 of [Use Case 29](#_pvfazrdypecx)).  32.0.2. System displays fields for start date, end date  32.0.3. Customer selects the start date (e.g., 2025-06-07), end date (e.g., 2025-06-09)  32.0.4. Customer submits the form as part of the rental process.  32.0.5. System validates the dates (e.g., end date must be after start date, start/end date must be in the future).  32.0.6. System saves the duration as part of the rental reservation data.  32.0.7. System proceeds to the next step in [Use Case 29](#_pvfazrdypecx) (availability check and cost calculation). | | |
| Alternative Flows: | None | | |
| Exceptions: | **32.0.E1: End date before start date.**  - Occurs at step 32.0.5.  - System displays: "End date must be after start date. Please correct."  - Customer corrects the dates and resubmits.  **32.0.E2: Start/end date is in the past.**  - Occurs at step 32.0.5.  - System displays: "Start/end date must be in the future. Please correct."  - Customer corrects the date and resubmits. | | |
| Priority: | High (integral part of the rental process). | | |
| Frequency of Use: | Approximately 100 times per day, aligned with rental reservations. | | |
| Business Rules: | BR31 | | |
| Other Information: | - Quality Attributes: Date and location validation should complete within 1 second.  - Failure Handling: If the use case fails, the rental process (Use Case 29) does not proceed, and no reservation is created. Validation errors are handled interactively with the customer. | | |
| Assumptions: | - The system provides a dropdown of available rental locations.  - Date fields use a calendar picker for user-friendly selection. | | |

##### 2.2.2.9 Commenting

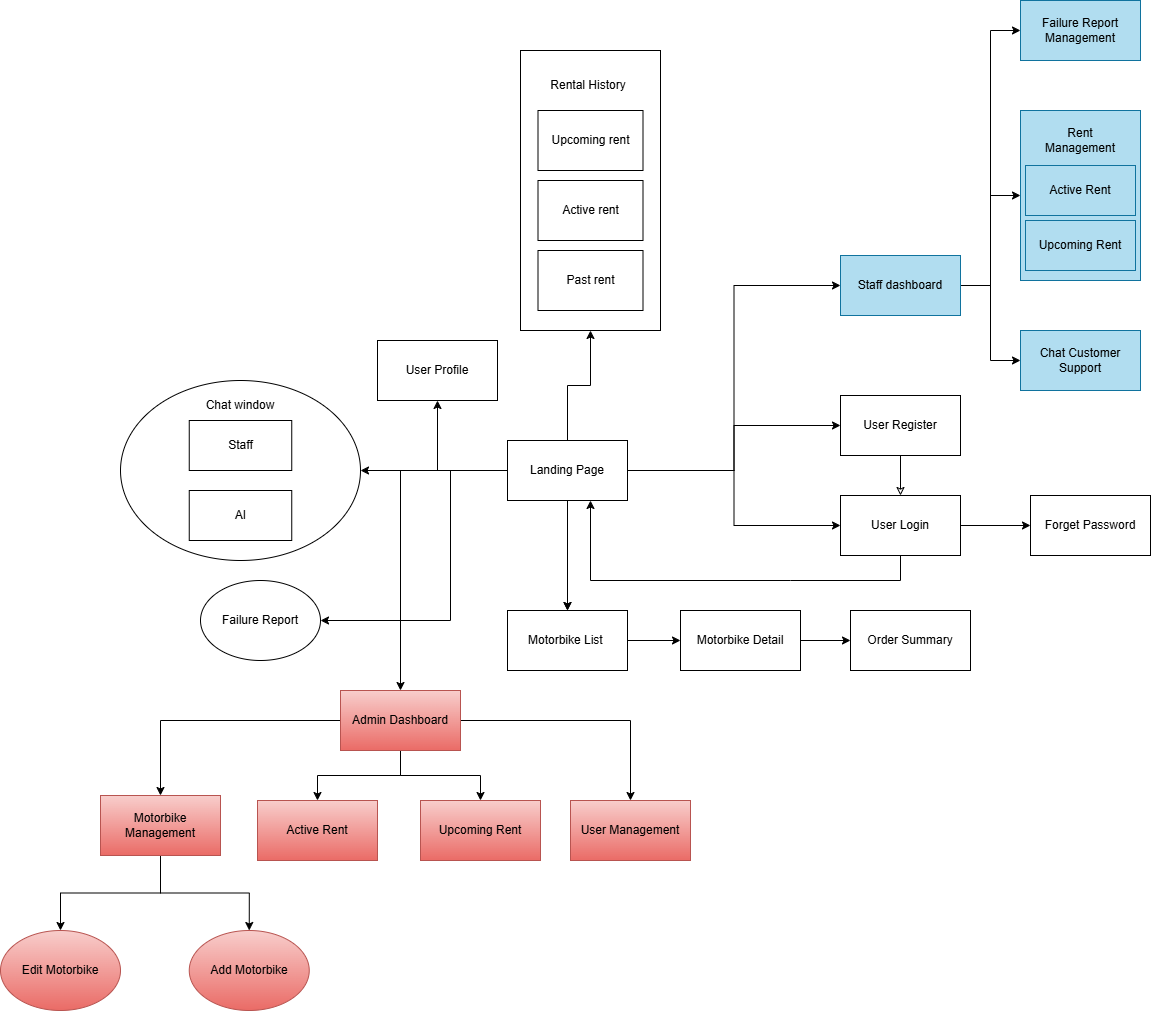
###### 2.2.2.9.1 Commenting

| UC ID and Name: | **UC-33 Commenting** | | |
| --- | --- | --- | --- |
| Created By: | Bui Khac Bin | Date Created: | 04/06/2025 |
| Primary Actor: | User | Secondary Actors: | None |
| Trigger: | A registered user indicates that they want to leave a comment on a specific motorbike model. | | |
| Description: | A registered user selects a model, it may be from the list of available models, the list of models they have rented in the past, or a direct link, then scroll down to the comment section, then leave a comment. | | |
| Preconditions: | PRE-1. The user is registered.  PRE-2. The user has rented the model they want to comment on in the past. | | |
| Postconditions: | POST-1. The comment is registered and logged in the system. Other users can view it. | | |
| Normal Flow: | **33.0**  33.0.1. User queries a list of available models.  33.0.2. System sends back a list of models.  33.0.3. User selects a model from the list.  33.0.4. User goes to the comment section of that model.  33.0.5. User leaves a comment.  33.0.6. System processes the comment and logs it. | | |
| Alternative Flows: | None | | |
| Exceptions: | None | | |
| Priority: | Low | | |
| Frequency of Use: | Under 10 users per model per day. | | |
| Business Rules: | BR13, BR14 | | |
| Other Information: | None | | |
| Assumptions: | At least 5 percent of users who rented a model comment on it. | | |

## 3. Functional Requirements

### 3.1 System Functional Overview

#### 3.1.1 Screens Flow



#### 3.1.2 Screen Descriptions

| **#** | **Feature** | **Screen** | **Description** |
| --- | --- | --- | --- |
| 1 | Account Management | User Profile | Display user information |
| 2 | Motorbike browsing and renting | Landing Page | Where renters choose the rent dates, location |
| 3 | Motorbike browsing and renting | Motorbike List | Display available motorbikes, search, and filter options |
| Motorbike Detail | Detail the motorbikes’ information and rent options: helmets, rain coats,... . |
| Order summary | Summary of the rental (motorbike’s details, rent dates, rent options, total price) |
| 4 | Motorbike rental management | Upcoming rent | Users view their upcoming rent |
| Active rent | Users view their active rent |
| Past rent | Users view their completed rent |
| 5 | Account Management | Login page | Where the user logs in to their account |
| Register page | Where the user registers their account |
| Forget password | Where the user recovers their password |
| 6 | Motorbike management | Edit motorbike | Edit the details of the motorbikes |
| Add motorbike | Add more motorbikes |
| 7 | Motorbike rental management | Upcoming rent | Admin view all upcoming rentals |
| Active rent | Admin view all active rentals |
| 8 | User management | User management | Admin can see registered users’ account information and take actions on their accounts |
| 9 | Admin dashboard | Dashboard overview | See metrics of the system: User registration trends, Rental volume trends |
| 10 | Support and Feedback | Chat Customer Support | Staff can see all the messages from the customers and chat with customer |
| 11 | Support and Feedback | Failure Report Management | Staff can see all the report sent from customers located in the city where staff is working |
| 12 | Support and Feedback | Chat window | Customer can chat with staff or AI for support |

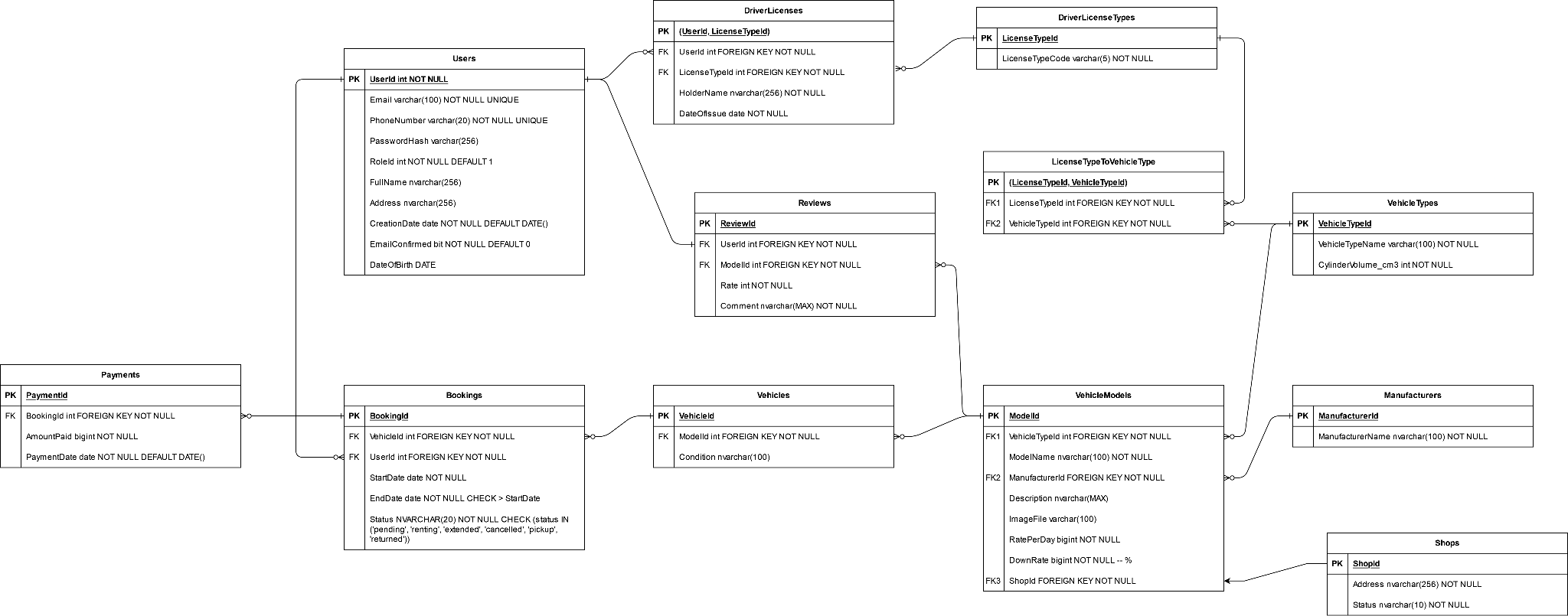
#### 3.1.3 Screen Authorization

| **Screen** | **Customer** | **Staff** | **Admin** |
| --- | --- | --- | --- |
| Landing Page | x |  |  |
| User Profile | x |  |  |
| Rental History | x |  |  |
| Chat Window | x |  |  |
| User Login | x | x | x |
| User Register | x | x | x |
| Forget Password | x | x | x |
| Bike List | x |  |  |
| Bike Detail | x |  |  |
| Order Summary | x |  |  |
| Failure Report | x |  |  |
| Staff Dashboard |  | x |  |
| Staff Rental Management |  | x |  |
| Chat Customer Support |  | x |  |
| Failure Report Management |  | x |  |
| Admin Dashboard |  |  | x |
| Bike Management |  |  | x |
| Active Rental |  |  | x |
| Upcoming Rental |  |  | x |
| User Management |  |  | x |

#### 3.1.4 Non-Screen Functions

| **#** | **Feature** | **System Function** | **Description** |
| --- | --- | --- | --- |
| 1 | Rentals statistic | Export rental statistics to files | Export rental statistics to csv file |
| 2 | Data Processing and Validation | OCR Data Extraction and Validation | Process OCR results from driver’s licenses and IDs to extract relevant fields (e.g., name, license number, date of birth, expiry date). Clean and validate extracted data (e.g., ensure the license number format matches expected patterns, check if the license is expired). Cross-check with existing user records to prevent duplicates. |
| 3 | User Profile Creation/Updating | Create or update user profiles in the database using validated OCR data. Map OCR-extracted fields to database fields, handle partial data (e.g., if OCR fails to read the expiry date), and flag incomplete profiles for manual review by staff. |
| 4 | Data Normalization | Normalize OCR-extracted data (e.g., convert dates to ISO format, standardize name capitalization) and ensure compatibility with database schemas or third-party APIs. |
| 5 | Integration with Google Maps API | Distance and Route Calculation | Use Google Maps Directions API to compute routes, factoring in real-time traffic data. Cache results to reduce API calls and improve performance. |
| 6 | Geofencing for Rental Zones | Use Google Maps Geocoding and Geometry APIs to set boundaries (e.g., city limits) and validate if a customer’s location is within the allowed zone during rental or return. |
| 7 | Location-Based Recommendations | Use Google Maps Places API to find the closest rental stations and cross-reference with available motorbike inventory in the database. |
| 8 | Payment Processing | Payment Initiation and Confirmation | Generate a payment request with rental details (e.g., amount, customer ID), send it to the payment provider, and handle the callback to confirm success or failure. Update the rental status in the database upon successful payment. |
| 9 | Payment Retry Logic | Implement a retry mechanism (e.g., retry up to 3 times with exponential backoff) for transient failures (e.g., network issues). Log failures and notify the customer/staff if all retries fail. |
| 10 | Refund Processing | Calculate the refund amount based on cancellation policies, initiate a refund request via the payment API, and update the transaction record in the database. |
| 11 | Rental Management and Scheduling | Inventory Availability Check | Query the database to compare requested rental dates with existing bookings, considering buffer times for maintenance or cleaning. Return available motorbikes matching the customer’s criteria (e.g., model, location). |
| 12 | Booking Conflict Resolution | Prevent double-booking by locking motorbike records during reservation attempts. If a conflict arises (e.g., two users request the same bike simultaneously), prioritize based on timestamps or user priority. |
| 13 | Rental Status Updates | Transition rental statuses (e.g., from "Reserved" to "Active" at pickup time, or to "Completed" at return time). Handle overdue rentals by flagging them and notifying staff. |
| 14 | Notification and Communication | Automated Email Notifications | Integrate with an email service (Resend) to send messages based on rental events. Use templates for consistency. |
| 15 | Announcement Scheduling | Based on the duration set in the "Add announcement to the entire website" use case, automatically publish and later remove announcements from the database and website. |
| 16 | Staff Alerts for Issues | Trigger alerts (e.g., via email or internal dashboard notifications) when predefined conditions are met (e.g., a customer reports a defect during rental). |
| 17 | Security and Compliance | Data Encryption for Sensitive Information | Use encryption standards (e.g., AES-256) to store sensitive data in the database and during transmission. |
| 18 | Audit Logging | Record user actions (e.g., rental bookings, payment attempts) and system events (e.g., maintenance mode activation) with timestamps, user IDs, and outcomes. Store logs securely for compliance. |
| 19 | Rate Limiting for API Calls | Implement rate limiting for OCR, Google Maps, and payment API calls to avoid exceeding quotas or being flagged for abuse. Cache frequent requests where possible. |
| 20 | Maintenance and Monitoring | Scheduled Maintenance Checks | Based on rental history or mileage data, flag motorbikes that need maintenance (e.g., after 10 rentals or 500 km). Remove them from availability until cleared by staff. |
| 21 | API Health Monitoring | Periodically ping APIs to check availability and latency. Log failures and notify the development team if an API is down. Fallback to manual processes if needed. |
| 22 | Database Cleanup | Periodically archive completed rentals older than a year, remove expired announcements, and clean up temporary data (e.g., failed payment logs after 30 days). |

#### 3.1.5 Entity Relationship Diagram



#### 

| **No** | **Table** | **Description** |
| --- | --- | --- |
| 01 | Users | Stores user data such as name, email. password, etc. |
| 02 | DriverLicenses | Stores driver license information acquired from image scanning. |
| 03 | DriverLicenseTypes | Stores types of driver licenses, such as “A1”, “A2”, etc. |
| 04 | VehicleTypes | Store types of vehicles such as electric bike, 50cc bikes, etc. |
| 05 | LicenseTypeVehicleType | Establishes a many-to-many connection between VehicleTypes and LicenseTypes. This is because a license can be applied to multiple types of vehicle, a type of vehicle can be allowed by many types of licenses. |
| 06 | Manufacturers | Stores details of manufacturers such as Honda, etc. |
| 07 | Shops | Stores details of shops, location and status, the website allows users to choose from where their vehicle will be from, and there are more than one option. |
| 08 | Reviews | Stores reviews of users on a specific model. |
| 09 | VehicleModels | Stores models of vehicles, two identical vehicles of a model. |
| 10 | Vehicles | Stores individual vehicle, one physical vehicle is a record, |
| 11 | Bookings | Stores booking records, one record corresponds to one user and one physical vehicle. |
| 12 | Payments | Stores payment information of bookings. |

### 3.2 Account Management

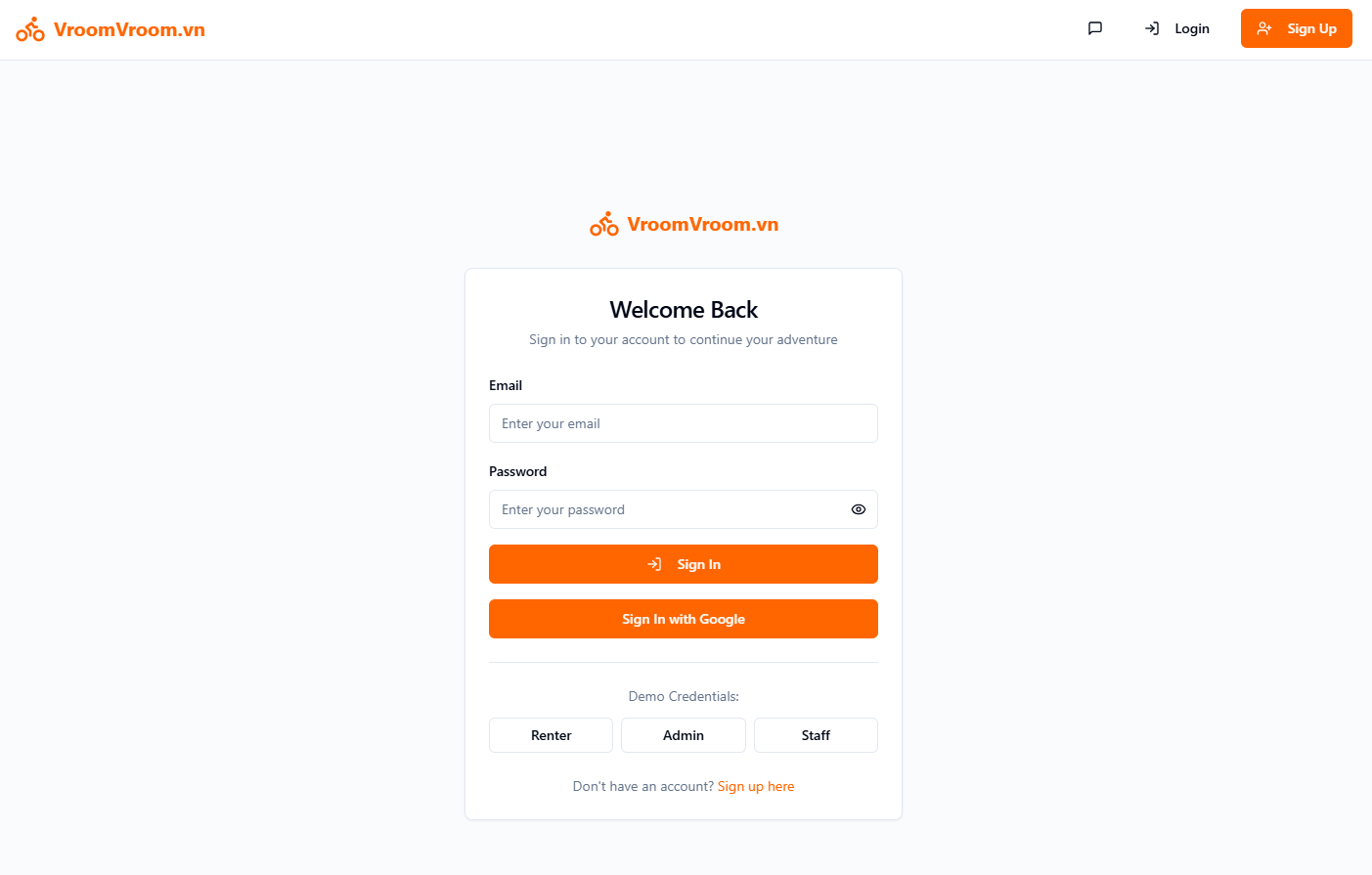
##### 3.2.1 Login Page

Function trigger:

1. User clicks on the "Sign In" button from the HomePage or accesses the page via the “Login” link on the Sign Up page.

2. Users can also log in using Google accounts via corresponding buttons.

● Screen Layout:



**Figure 12. Sign In Screen**

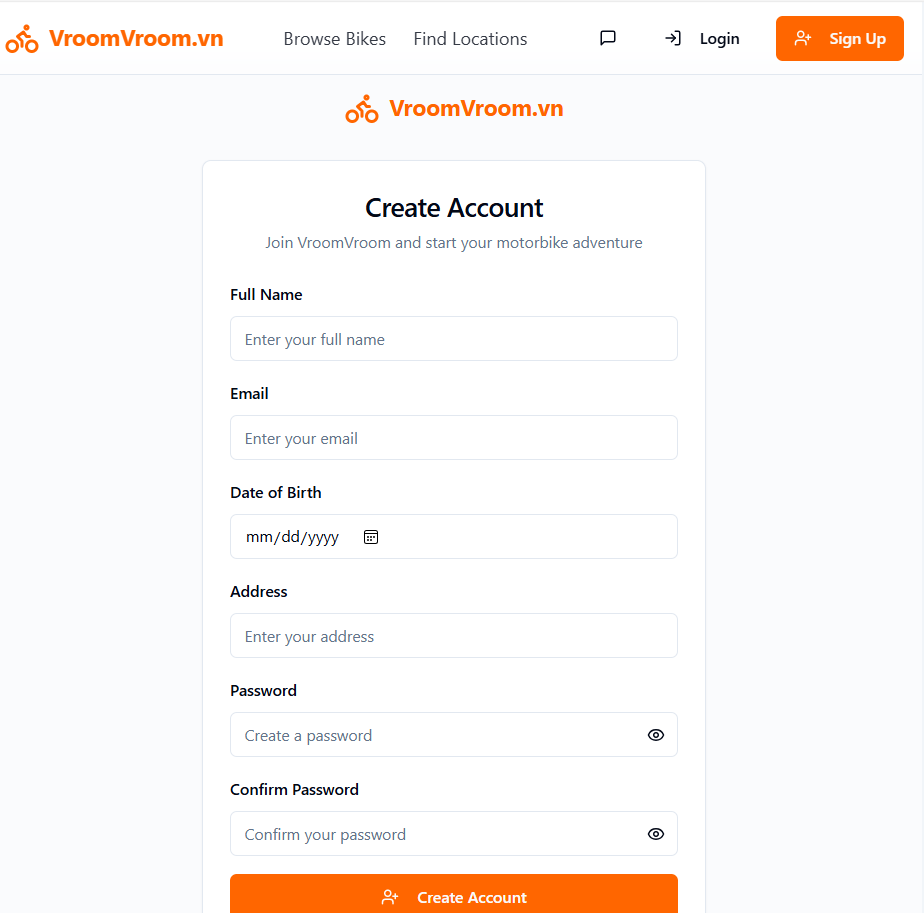
Function detail:

1. The system checks for the presence of required login information (username and password).
2. It then verifies the account existence and password correctness.
3. If login credentials are correct, the system logs the user in and redirects to the dashboard.
4. If login credentials are incorrect, an appropriate error message is displayed.

##### 3.2.2 Register Page

* Function trigger: User clicks on the **"Sign Up"** button from the **HomePage** or accesses via the **“Register”** link on the **Sign In** page.
* Function detail:
  + **Validation:**
    - All fields are required. The system checks if the email is valid and if the password meets certain complexity requirements.
    - Email address and password must be in a valid format.
    - If any field is left empty, the system displays appropriate error messages and does not enable the "Sign Up" button.
  + **Functionalities:**
    - The system checks if all required registration information is filled, and validates the email and password.
    - The system verifies if the username or email is already registered in the system.
    - If all requirements are met, the system allows the user to create a new account and automatically assigns the User role.
    - If the fields are incorrect or incomplete, the system asks the user to correct the information before proceeding.

● Screen Layout:

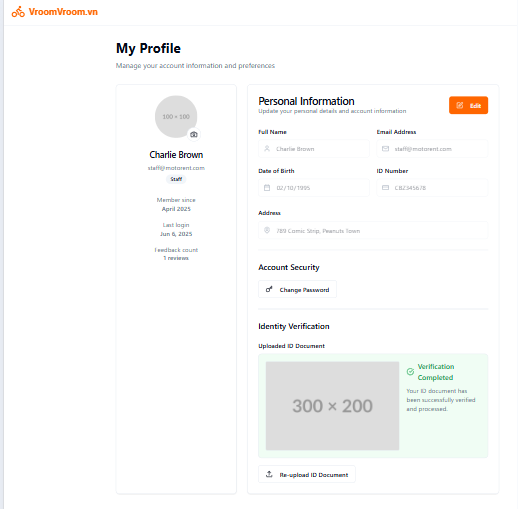


##### 3.2.3 User Profile

* Function trigger: The "View Profile" function is triggered when the user logins successfully. If a user wants to see their profile, they can click on the avatar in the left corner.
* Function detail:

- Functionalities: This screen displays the user's profile information including their name, username, email, phone, address, profile image. They can edit their profile information.and upload a driver's license photo

● Screen Layout



##### 3.2.4 Forgot password

● Function trigger: User click on “Change PassWord” button in their profile page.

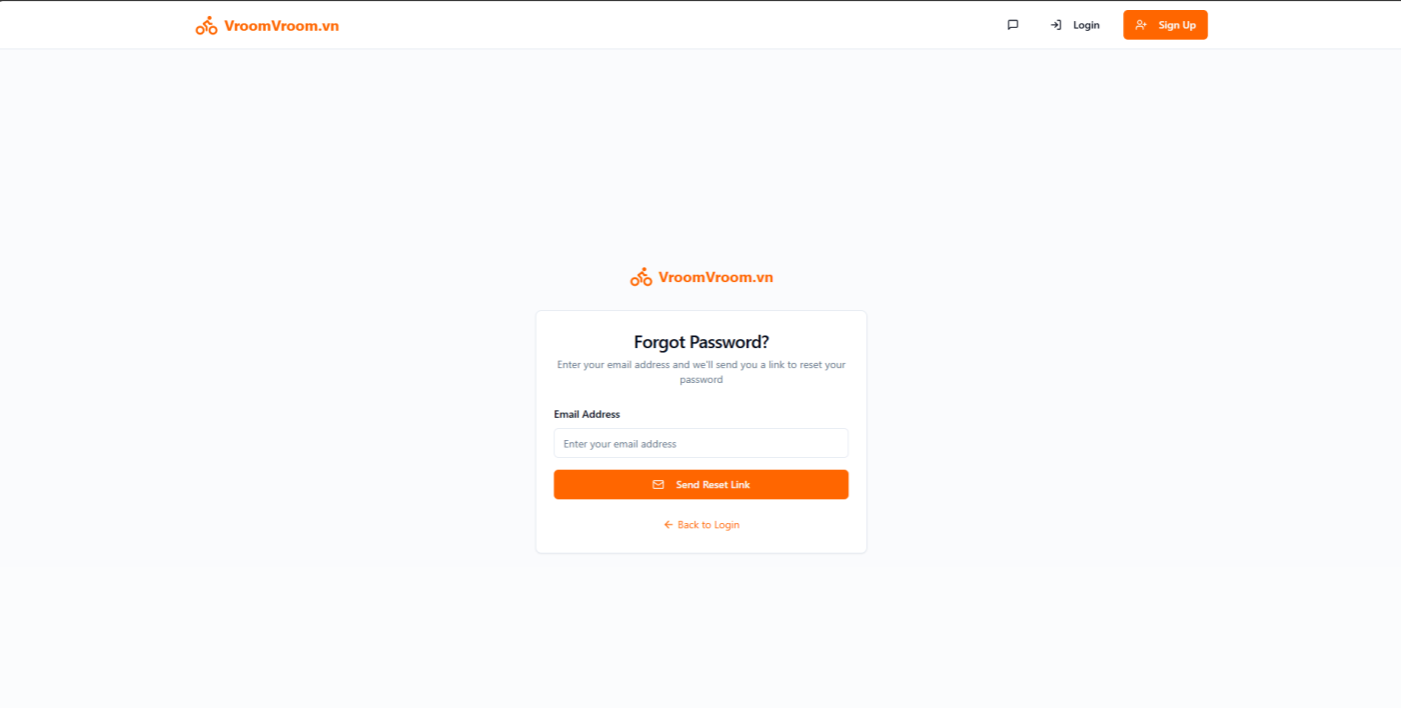
* Function Details:

- Functionalities: This page allows users to change their current password by entering

their old and new passwords. An error message is displayed if the current password

entered is incorrect or if the new passwords do not match.

Screen Layout:



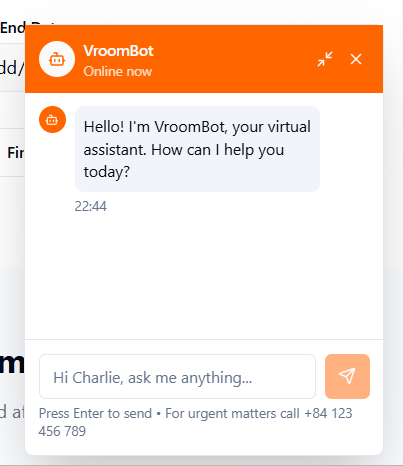
### 3.4 Support and Feedback

##### 3.4.1 Chat Window

**Function trigger:** The “Chat Customer Support” function is triggered when the user clicks on the “Support” button or icon in the navigation bar or homepage. This action opens a chat box at the bottom corner of the screen.

**Function detail:**

* **Functionalities:** This function enables users to:  
  + Chat directly with customer support in real-time
  + Ask questions and receive automated or human support
  + Receive typing indicators and message status (e.g., sent, seen)
* Screen Layout:

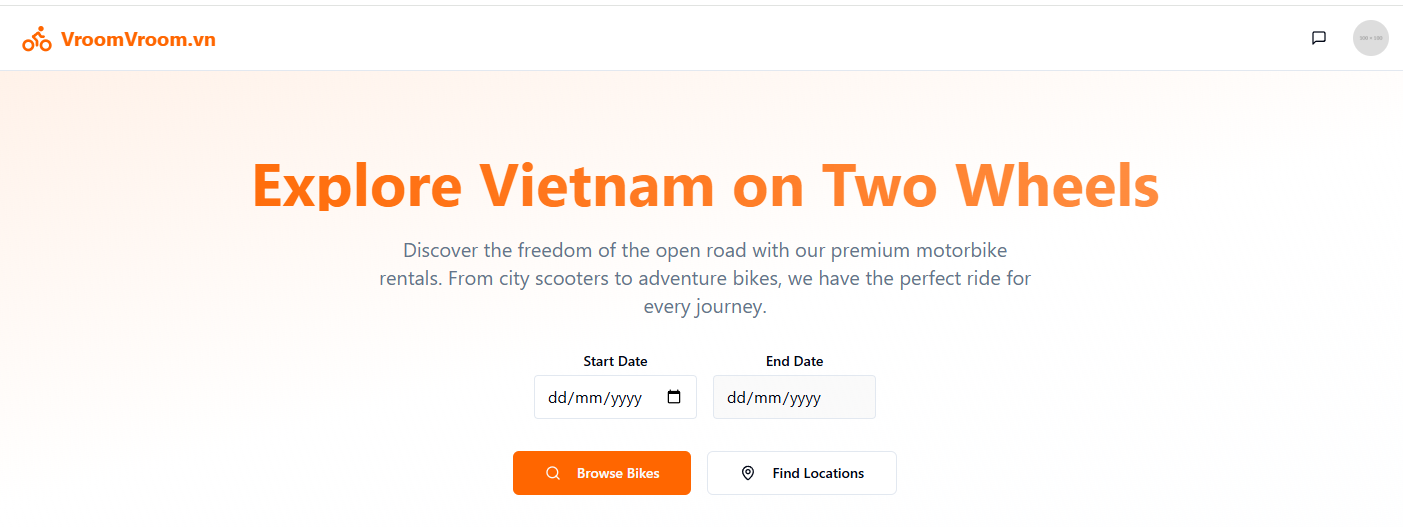


### **3.**5Motorbike browsing and renting

3.5.1 Select rent duration

- Function trigger: User only need to access the landing page of website

- Screen layout:



- Function details:

+ In this landing page, users can select the start date and end date to browse available motorbikes.

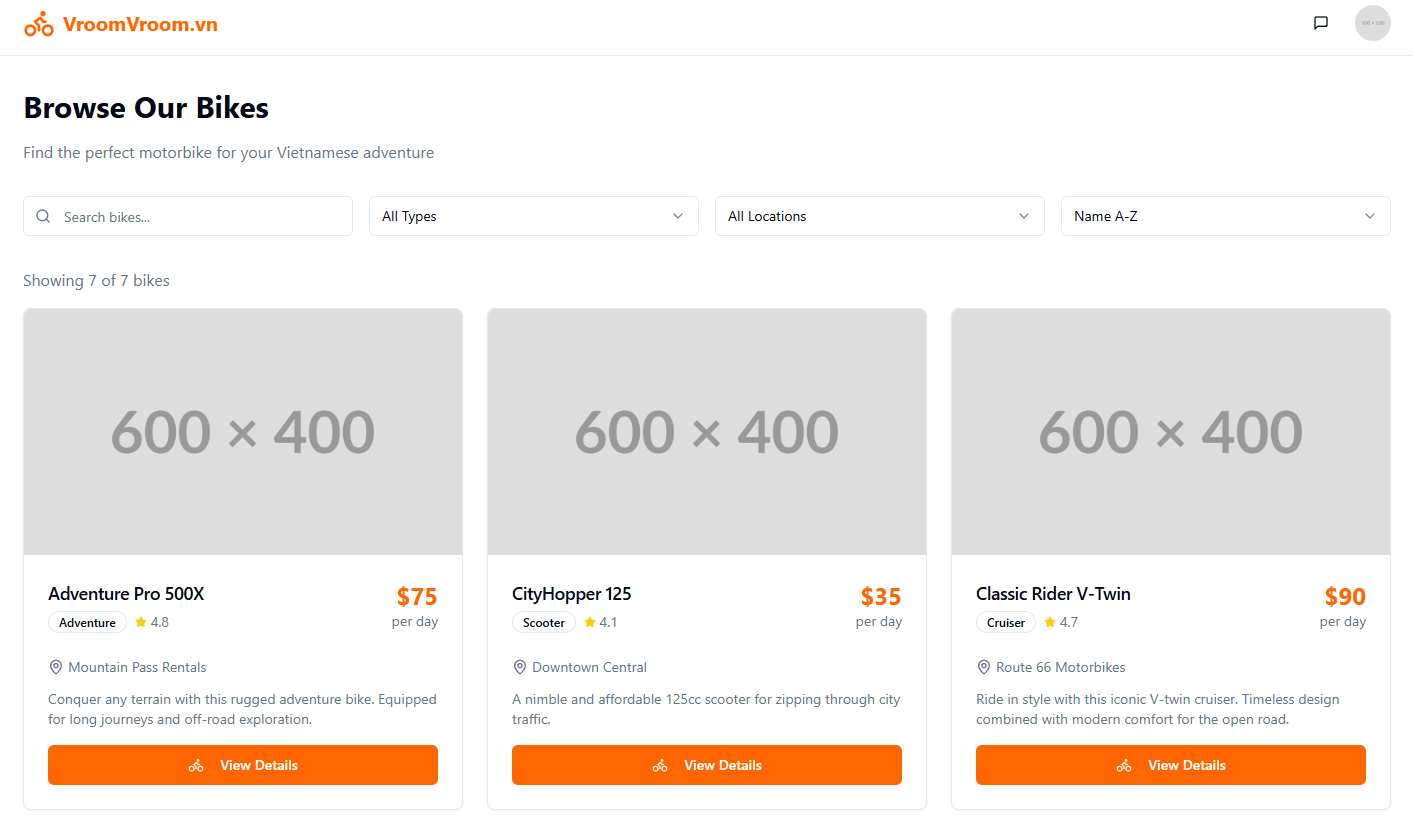
+ System validates the dates (e.g., end date must be after start date, start/end date must be in the future). System displays: "End date must be after start date. Please correct." if end date before start date. System displays: "Start/end date must be in the future. Please correct." if start/end date is in the past

+ Users can only select end date when they already select their start date

3.5.2 Browse available motorbikes

- Function trigger: User clicks Browse Bikes button in the landing page after selecting start date and end date.

- Screen layout:



- Function details:

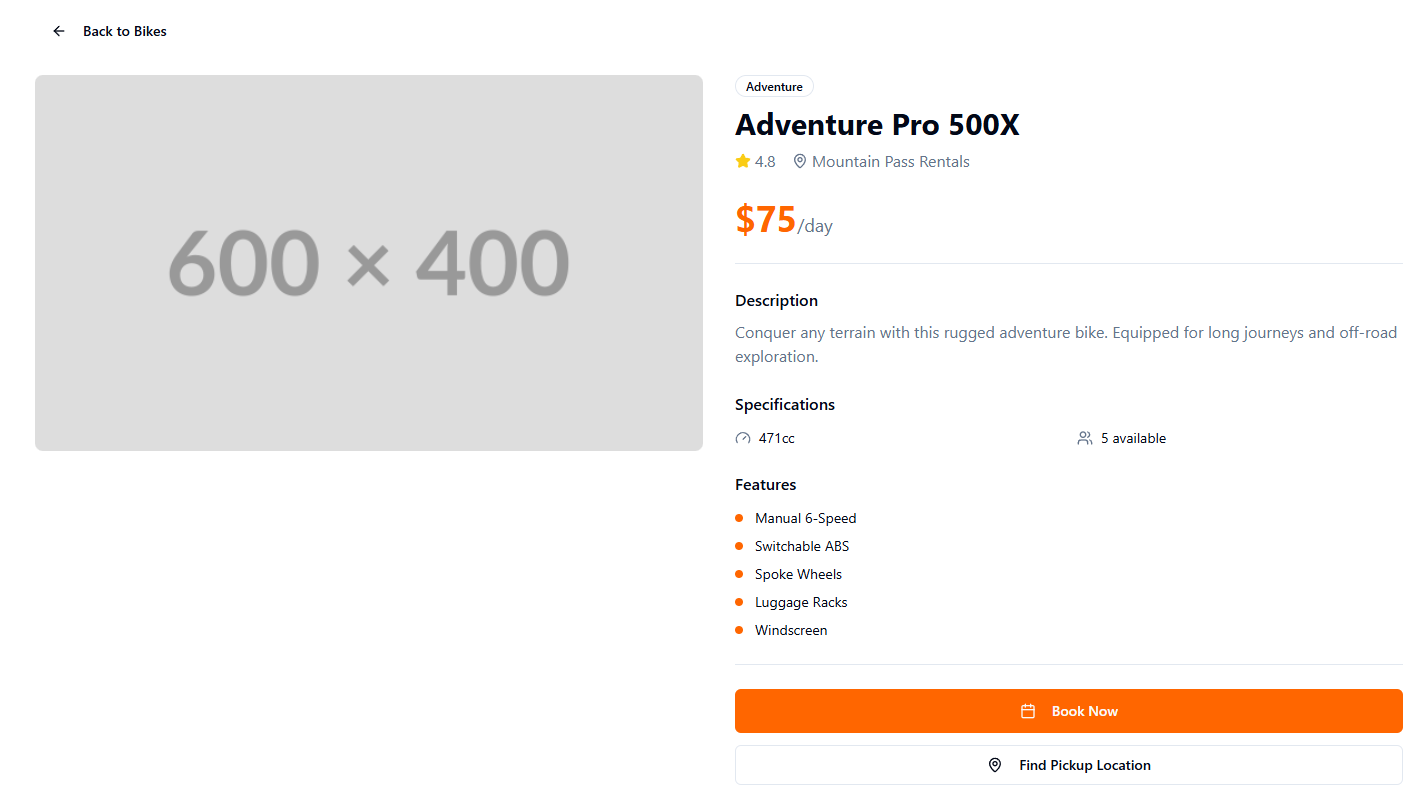
+ In this page users can view available motorbike models during the rent duration the user selected in the landing page.

+ Users can filter by clicking dropdowns or search a specific bike by entering their name in the Search bikes input bar.

3.5.3 View details of a motorbike

- Function trigger: User clicks View Details button when browsing available motorbikes during rent duration

- Screen layout:



- Function details:

+ In this page users can view information of a specific model: description, cylinder volume, features

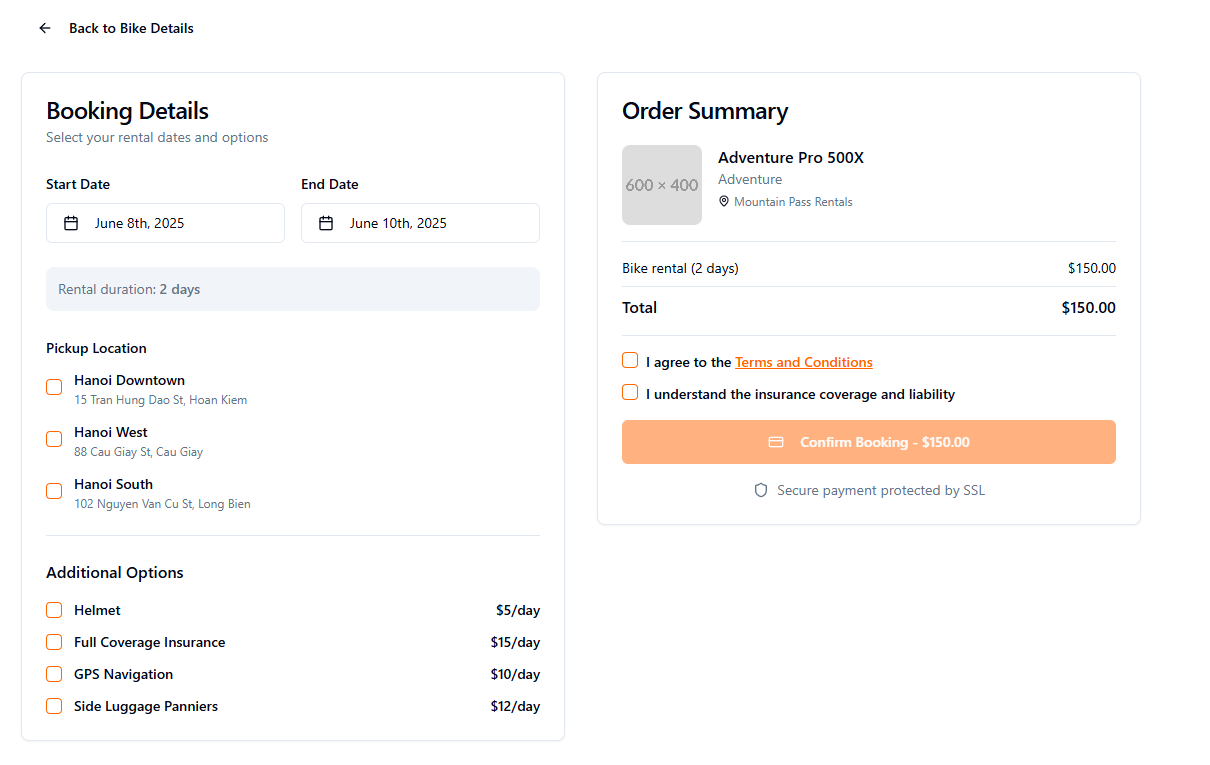
+ Validation:

If the user has not uploaded a valid driver license, the system displays: “Please upload and verify your driver license before renting a motorbike” and the “Rent now” button is blocked, forcing the customer to upload and verify their license.

3.5.4 Rent a motorbike

- Function trigger: User clicks Book Now button when view detail of a model

- Screen layout:



- Function details:

+ In this page, users can reselect their start date and end date. The pickup location is the location where the model is available, and will be changed according to the start date and end date.

+ Users will choose a pickup location and can choose additional options.

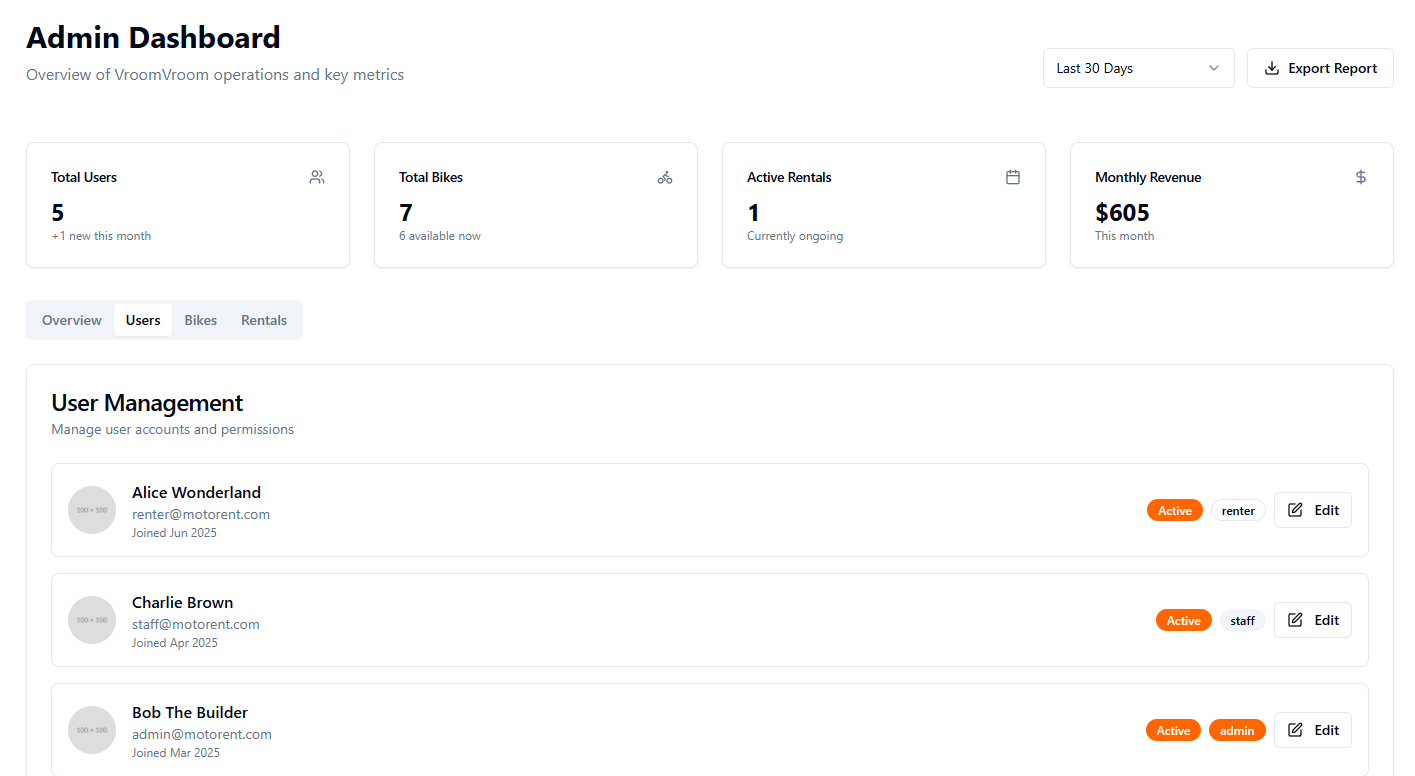
+ Users have to click I agree to Terms and Condition checkbox to be able to click the Confirm Booking button.

##### 4. User Management

4.1 View User List

- Function trigger: Admin click User tab in the Admin Dashboard page

- Screen layout:



- Function details:

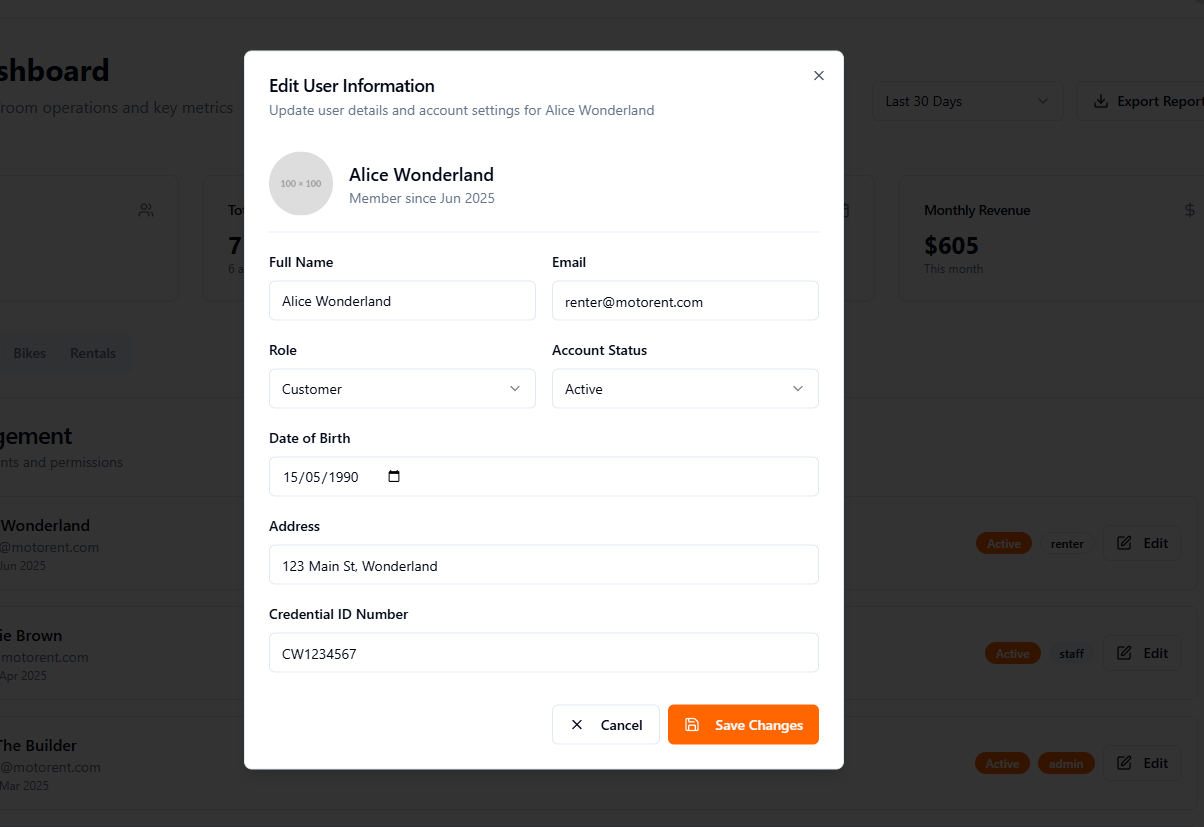
+ Validation: The current user is authorized as admin

+ In this page, the admin can view a list of users: staff and renters and their information. Admin can edit users’ information

4.2 View and Edit User Information

- Function trigger: Admin click Edit button in each Customer card in the User Management page

- Screen layout:



- Function details:

+ Validation: The current user is authorized as admin

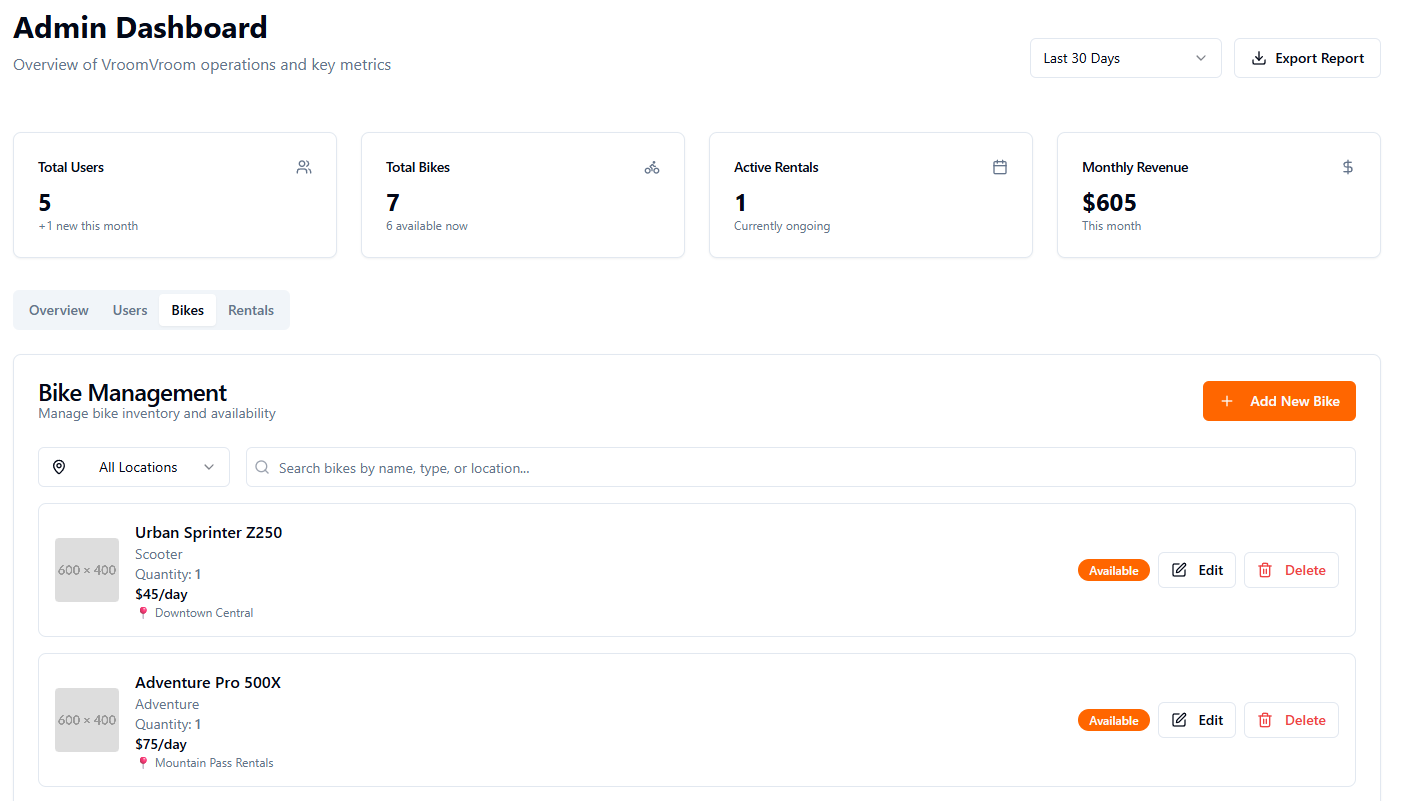
+ In this popup window, admin can view the information of a specific user of the website. The admin can only change the role of the user: Customer, Staff and set the account status: Active, Inactive.

##### 5. Motorbike Management

5.1 View motorbike list

- Function trigger: Admin click Bikes tab in the Admin Dashboard page

- Screen layout:



- Function details:

+ Validation: The current user is authorized as admin

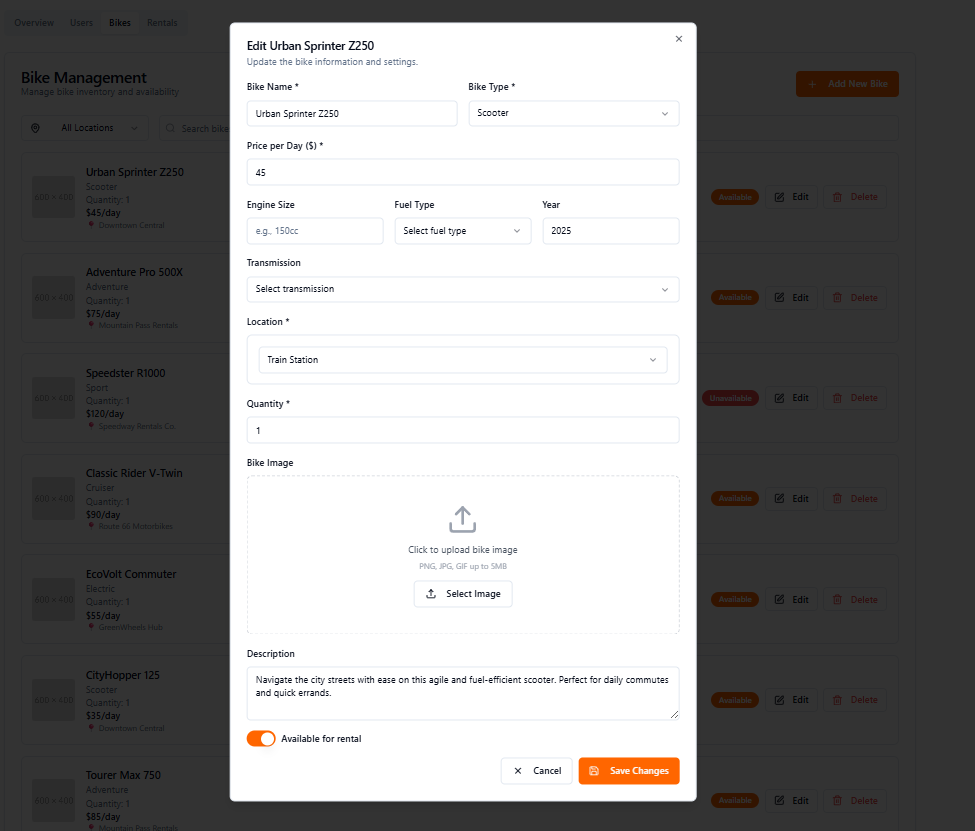
+ In this popup window, admin can view the information of models available in all locations. The admin can filter by clicking the All Locations dropdown list to select which specific location they want to filter the models on. The admin can search for a specific model by entering their name in the search bar.

+ Admin can add a new model, edit or delete a model.

5.2 Edit motorbike

- Function trigger: Admin clicks the Edit button in the Bike Management page

- Screen layout:



- Function details:

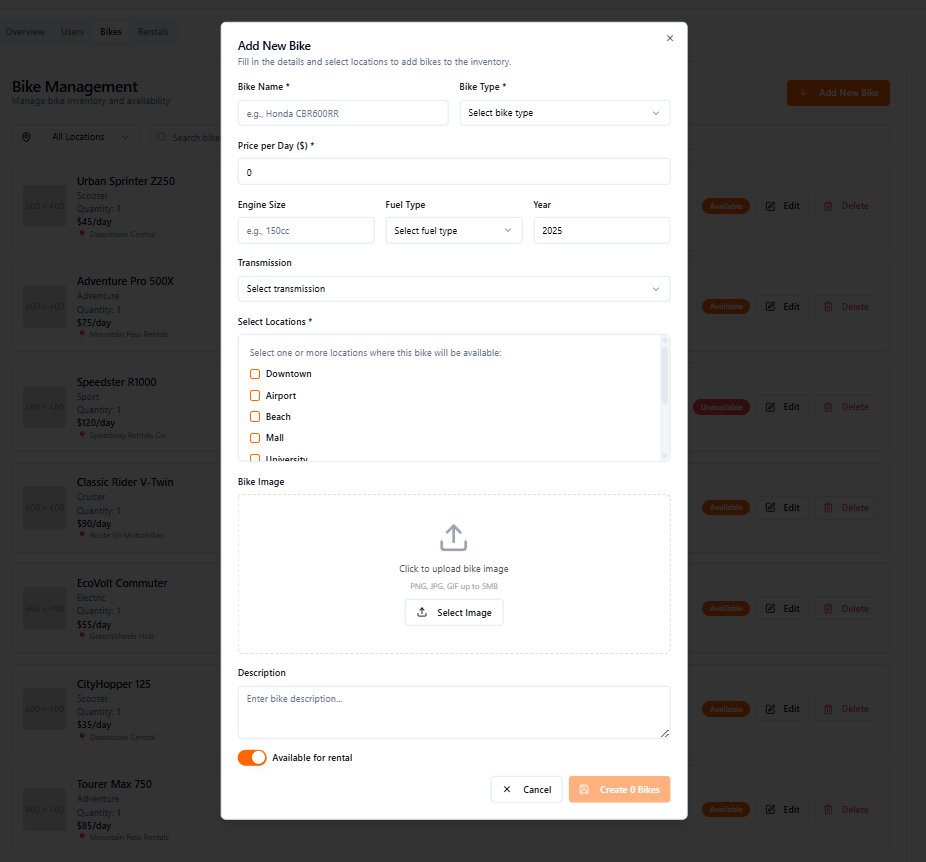
+ Validation: The current user is authorized as admin

+ In this popup screen, admin can view the information of a specific model. The admin can update information about the model.

5.3 Add New Motorbike

- Function trigger: Admin clicks the Add New Bike button in the Bike Management page

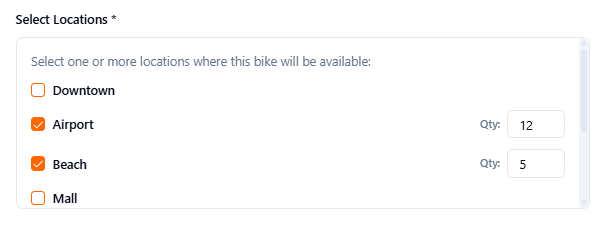
- Screen layout:



- Function details:

+ Validation: The current user is authorized as admin

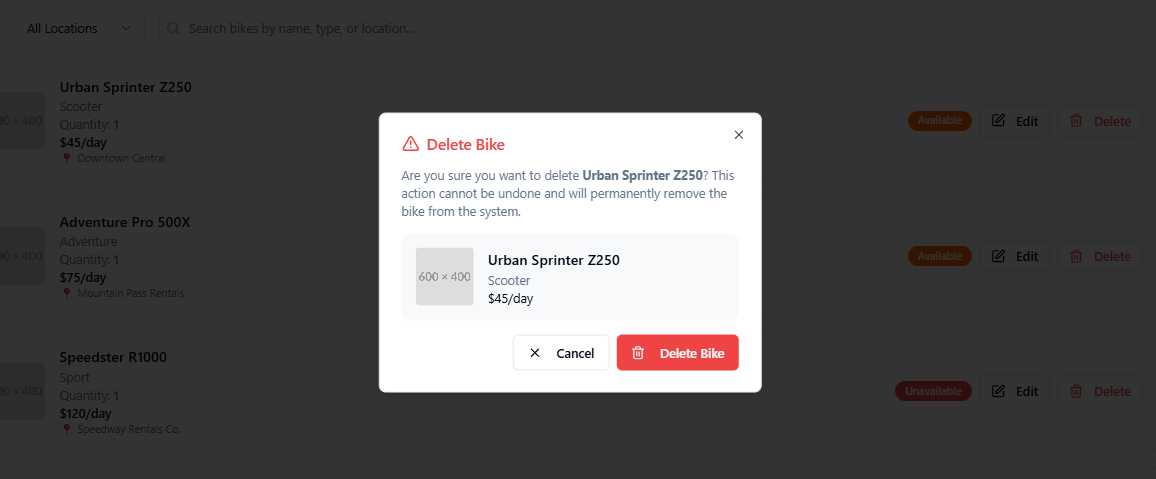
+ In this popup window, admin can add a new model. When selecting locations, admin can modify the quantity in each location.



5.3 Add New Motorbike

- Function trigger: Admin clicks the Delete button in the Bike Management page

- Screen layout:



- Function details:

+ Validation: The current user is authorized as admin

+ In this popup window, admin can add a new model. When selecting locations, admin can modify the quantity in each location.

##### 6. Website Management

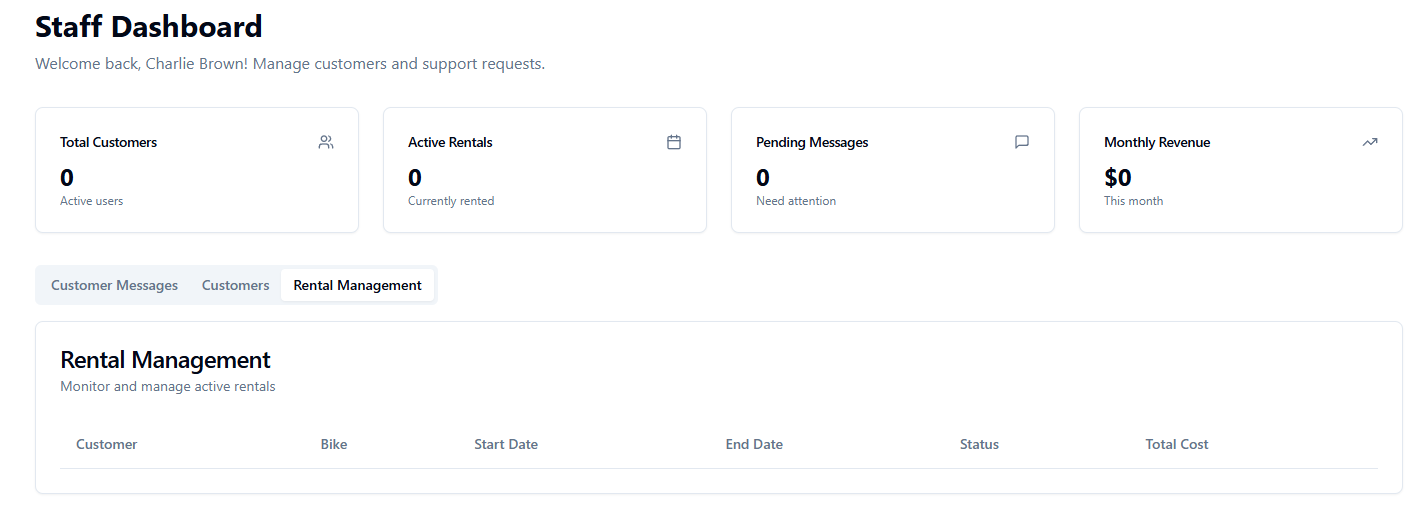
##### 7. Bike Rental Management

7.1 View Current and Upcoming Rentings

- Function trigger: Staff click Rental Management tab in the Staff dashboard

- Function description: Staff views the list of current and upcoming rentings

- Screen layout: Not complete



- Function detail:

+ System retrieves and displays current and upcoming rentings for the staff’s location

7.2 Update Rent Status

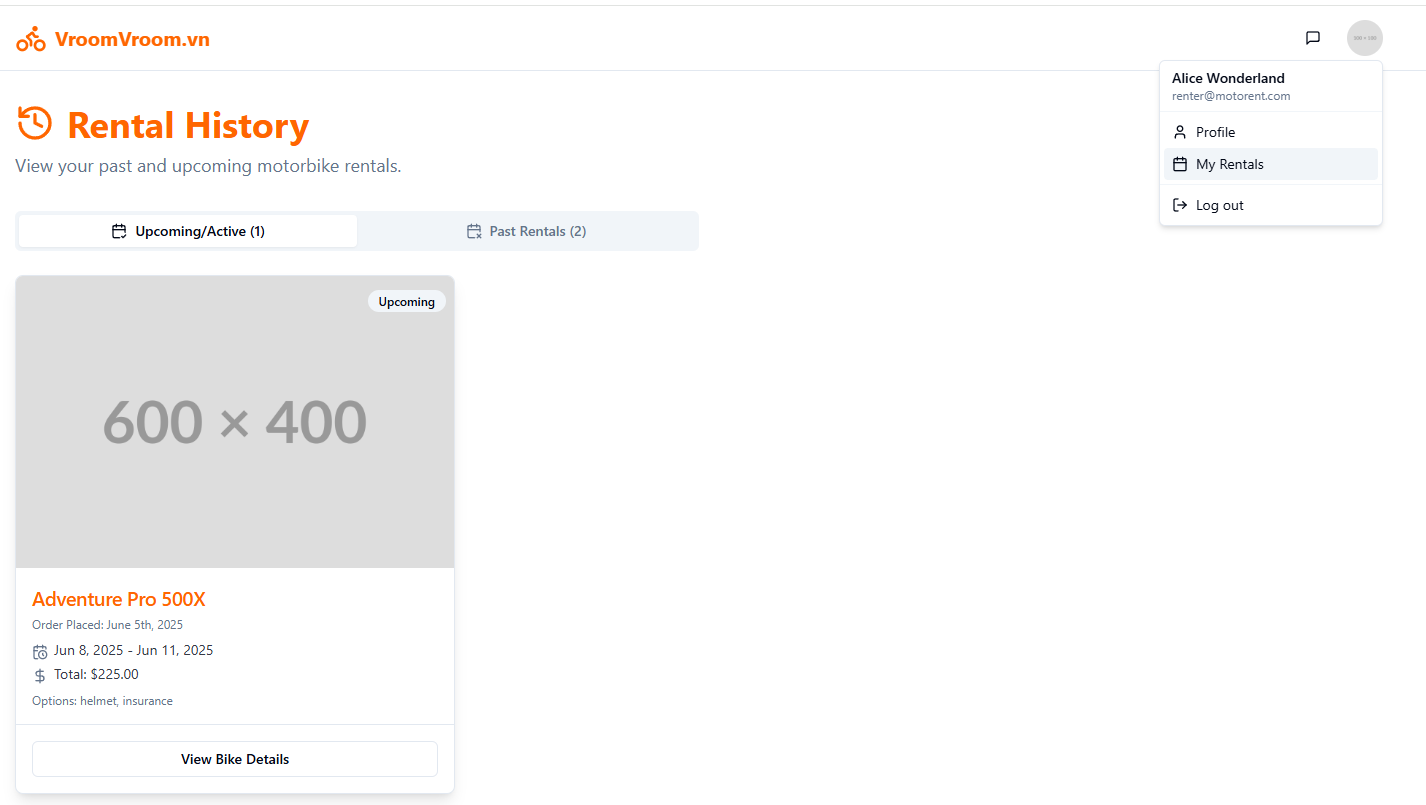
7.3 Appoint Bike to Customer

7.4 View Rented Motorbike

- Function trigger: User clicks My profile bar in the pop up window when clicking the profile icon

- Function description: Users can view upcoming, active or past rentals

- Screen layout:



- Function details:

+ System retrieves and displays the rental details of the user: motorbike model, order date, start date, end date, total money, and optional rental: helmet, insurance, … .

+ If the user has no active rental, the system displays a message: “No active rentals found.”

7.5 Cancel Rent

##### 8. Reporting and Statistics

8.1 Export Statistics as CSV files

## 4. Non-Functional Requirements

### 4.1 External Interfaces

#### 4.1.1 Hardware

The system must be accessible with 1 Gigabyte of RAM, a dual core processor, and integrated graphics.

Performance should be reasonable with 2 Gigabytes of RAM and a quad core processor.

Users must be able to capture images from any of their devices, the system must not impose any hardware restrictions on capturing.

#### 4.1.2 Software

The system must support common browsers, such as Google Chrome (v.128+), FireFox (v.128+), Safari (v.17.0+), Opera (v.115+), …

### 4.2 Quality Attributes

#### 4.2.1 Usability

The UI must be intuitive and easy to use, it should take no more than 5 minutes for a new user to register and log in with their account. The UI navigation of the normal flow of renting motorbikes must not take more than 15 minutes.

#### 4.2.2 Reliability

The system must maintain at least 99% uptime in a year, maintenance excluded.

The Mean Time Between Failures must be above 720 hours.

The Mean Time to Repair must not exceed 24 hours.

There must not be more than 3 bugs every 10,000 lines of code, in the first month since uptime.

There must not be a single critical bug in the system.

#### 4.2.3 Performance

The front-end of the system must fully load in 5 seconds or lower with a 4G connection or equivalent.

Every API call must not take more than 500 millisecond, or 1 seconds for large data retrieval / resource heavy calls.

The system should be able to handle at least 500 concurrent users without having diminishes in performance and response time.

#### 4.2.4 Security

Sensitive user data such as full name, password, phone number, email must be encrypted with a strong algorithm, such as SHA-256 or better.

All data must be transferred under HTTPS, enabled with a valid SSL certificate.

## 5. Requirement Appendix

### 5.1 Business Rules

| **ID** | **Rule Definition** | **Rule Description** |
| --- | --- | --- |
| BR1 | User information | Only display email, full name, date of birth, address, and credential id |
| BR2 | Authenticated Users | Only users with correct permissions can access privileged pages. |
| BR3 | Information Validation | User information fields must be valid (address must exist, phone number must be valid, etc). |
| BR4 | Driver License Validation | Driver license information must be pulled from an image through OCR (users can still manually edit them after OCR), and any users with a valid license will be able to rent a motorbike. |
| BR5 | Filter Validation | Filter criteria must be valid (e.g., price range must be positive, date must be in a valid format). |
| BR6 | Location Restriction | Staff can only view rentings for their assigned location. |
| BR7 | Status Transition | Status changes must follow valid transitions (e.g., renting → returned or extended, but not cancelled → extended). |
| BR8 | Data Access | Customers can only view details of their own active rentals. |
| BR9 | Bike Availability | Only available motorbikes (not rented or reserved) can be appointed to a customer. |
| BR10 | Cancellation Policy | Rentals can only be cancelled or returned early if allowed by the system’s terms and conditions (e.g., within a specific time frame). |
| BR11 | Ability to select time span | The staff or admin can select the range of dates they can export statistics from. |
| BR12 | Security and data integrity | The requested statistics are not stored on the server as a file, they are sent directly to the staff or admin. |
| BR13 | User registration verification | Users must be registered and logged into their account. |
| BR14 | User rental history verification | Users must have rented the model they want to comment on in the past. |
| BR15 | Announcement Visibility | Announcements must be visible to all users on all pages. |
| BR16 | Email address | Email address can not be changed |
| BR17 | Deletion Restriction | A motorbike cannot be deleted if it is currently rented or reserved. |
| BR18 | User Role Restrictions | A user with the role of admin cannot be removed. |
| BR19 | Soft deletion | Set the account status to be deactivated. |
| BR20 | Data Display | The motorbike list must display model, price, condition, and availability status. |
| BR21 | Input Validation | Price must be a positive number, and condition must be selected from predefined options (e.g., New, Used, Damaged). |
| BR22 | Chat Logging | Chat sessions must be logged for audit and quality assurance for at most 7 days, which after the time frame must be deleted from the database for security and privacy purposes. |
| BR23 | Feedback Form Validation | Feedback must include a rating, but comments are optional. |
| BR24 | Feedback Logging | Feedback must be logged for review and further analysis to improve the service. |
| BR25 | Problem Report Condition | Problem reports must be linked to an active rental. |
| BR26 | Report Notification | Staff must be notified immediately of new support tickets. |
| BR27 | Payment Check | A user must have a valid payment method (bank account, credit card, e-wallet, etc) to process down payment and rent motorbikes. |
| BR28 | Down Payment | A down payment is required to confirm a reservation. |
| BR29 | Transaction Logging | All transactions must be logged and users must be able to see past rentings with their transactions. |
| BR30 | Discount Usage | Discounts can only be applied at checkout page and before confirmation. |
| BR31 | Date Validation | Rental date range must be valid (end date must be after start date, both start and end date must be in the future). |

### 5.2 Application Messages List

| **#** | **Message code** | **Message Type** | **Context** | **Content** |
| --- | --- | --- | --- | --- |
| 1 | MSG01 | In line | There is not any search result | *No search results.* |
| 2 | MSG02 | In red, under the text box | Input-required fields are empty | *The \* field is required.* |
| 3 | MSG03 | Toast message | Updating asset(s) information successfully | *Update asset(s) successfully.* |
| 4 | MSG04 | Toast message | Adding new asset successfully | *Add asset successfully.* |
| 5 | MSG05 | Toast message | Confirming email of asset hand-over is sent successfully | *A confirmation email has been sent to {email\_address}.* |
| 6 | MSG06 | Toast message | Resetting asset information successfully | *Return asset(s) successfully.* |
| 7 | MSG07 | Toast message | Deleting asset information successfully | *Delete asset(s) successfully.* |
| 8 | MSG08 | In red, under the text box | Input value length > max length | *Exceed max length of {max\_length}.* |
| 9 | MSG09 | In line | Username or password is not correct when clicking sign-in | *Incorrect user name or password. Please check again.* |
| 10 | MSG10 | In line | Database connection error | Database connection error. |

1. Throughout this report, the terms “User” and “Customer” are used interchangeably, and they bear the same definition as a normal actor using the core services of the website, including but not limited to renting a motorbike and giving feedback. [↑](#footnote-ref-0)