**TSE2101   
Individual Report**

**for**

Nursery Plant Shopping System

**Version <1.0>**

**Tutorial Section: TT3L**

**Group No.: Group 3**

|  |  |  |
| --- | --- | --- |
| **LIM CAI QING** | **1211102753** |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

| **Date:** | **11/02/2024** |
| --- | --- |

**Table of Contents**

[**1 System Overview 4**](#_2et92p0)

[1.1 Description 4](#_5ufsn0x0oelf)

[1.2 Use Cases 4](#_2s6lmszumj6)

[1.3 Assumptions and Dependencies 5](#_vw3zftqno856)

[**2 Requirements 5**](#_ql6fabufnoo)

[2.1 Use Case Diagram 5](#_4d34og8)

[2.2 Class Diagram 6](#_ngmtlbu1vbzj)

[2.3 State Diagrams 7](#_24dc9lpy7gw3)

[2.4 Data Flow Diagrams 9](#_5d1vyso308yv)

[**3 Design 11**](#_jb2c8anlhatl)

[3.1 Use Cases 11](#_35nkun2)

[3.2 Data Dictionary 33](#_8ukbkfocc18o)

[3.3 Data Structures 43](#_l0lfgqf0ga8k)

[3.4 Subsystem Architecture 49](#_cnpk8wswy9cc)

[3.5 Subsystem Screens 50](#_jj1mrx5x87c)

[3.6 Subsystem Components 59](#_5bj0vf2s9rkx)

[**4 Implementation 73**](#_4i7ojhp)

[4.1 Development Environment 73](#_b99i6j28b1k7)

[4.2 Main Program Codes 74](#_n88dfjvi1s2u)

[4.3 Sample Screens 113](#_tmjs0eexzqlq)

[**5 Testing 125**](#_2bn6wsx)

[5.1 Test Data 125](#_fq2gshc33l7m)

[5.2 Acceptance Test 129](#_3dhb5ec0ooj1)

[5.3 Test Results 131](#_fzal7f8ffrfu)

[**6 Conclusion 170**](#_pcp3ixi8raxn)

[6.1 Project Achievements 170](#_49x2ik5)

[6.2 Quality Assurance 170](#_lg5wgkgrthiz)

[6.3 Problems Encountered 170](#_47a5nexa4dr9)

[6.4 Remarks/Comments 171](#_n3txylgfgwdy)

**Revisions**

| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| Version 1.0 | Lim Cai Qing | This version delineates the comprehensive functionalities and user interactions within a Nursery Plant Shopping System, with a specific focus on the customer perspective. As the point person for the customer module, it's my responsibility to present the full breadth of the system's capabilities which are devised to enhance the online plant shopping experience. The content encompasses an array of diagrams, such as use case, class, state, and data flow diagrams, elucidating the intricate relationships and processes within the system from a customer's viewpoint. Additionally, it includes exhaustive use case descriptions accompanied by sequence diagrams to visualize the customer's journey. The database design is detailed with a complete data dictionary, ensuring a robust back-end structure that supports the front-end functionalities. The architectural and interface designs are crafted with the end-user in mind, ensuring ease of navigation and interaction. Subsystem components are broken down into detailed activity diagrams, providing a granular view of each operational facet. For the implementation phase, the report outlines the development environment, the primary codebase, and sample system interfaces. Testing procedures are thoroughly documented, including test data, acceptance criteria, and results, to underscore the system's reliability. The report culminates with a reflective section that discusses project milestones, quality assurance measures, challenges faced, and final thoughts on the project's execution. | 11/02/2024 |

# **System Overview**

## Description

The Nursery Plant Shopping System serves as a platform that facilitates interactions among four primary user roles: administrator, guests, customers, and delivery men. I am in charge of the customers.The Nursery Plant Shopping System offers customers a seamless online experience for selecting and purchasing a diverse range of nursery plants, complemented by features to manage their profiles and security settings. Customers can navigate through the platform to add or remove plants from their shopping cart, search for plants by name, complete purchases with secure payment options, and engage with the community by leaving reviews and ratings. This intuitive system simplifies the entire process, from browsing to order management, tailored to gardening enthusiasts' needs.

## Use Cases

Table 1.1 provides an overview of the use cases for a customer in the Nursery Plant Shopping System.A customer is a registered user who has access to various functionalities within the system. The key features available to a customer include the ability to change their password, edit profile details, and manage their shopping experience. Customers can search for plants by name, add or remove plants from their shopping cart, and view the items they are interested in purchasing. Additionally, customers can complete purchases, manage orders, and leave reviews and ratings for plants they have bought.

Table 1.1: Customer’s Use Cases

| Actor | Use Cases |
| --- | --- |
| Customer | A customer changes their password |
| A customer can edit profile details. |
| A customer shops plants.   * View added plants in the shopping cart. (sub-use case) * Add plants to the shopping cart. (sub-use case) * Delete plants from the shopping cart. (sub-use case) |
| A customer searches a plant by plant name. |
| A customer completes purchases and manages orders.   * Make payment. (sub-use case) * View order details of ongoing orders. (sub-use case) |
| A customer leaves reviews and ratings. |
| A customer manages a wishlist. |

## Assumptions and Dependencies

In this project, there are several assumptions that we have made:

1. In the context of the payment process, all customers have previously stored their required payment details within the system. When a customer selects a payment method, such as FPX(Internet Banking) or Touch 'n Go Wallet, the system will display a confirmation message for successful payment, eliminating the need to redirect to a third-party payment gateway.
2. For the shipping fee calculation process, it is assumed that accurate state information of customers is available, and the predetermined shipping fee rates for West Malaysia (RM7.00) and East Malaysia (RM12.00) are correctly implemented within the system.

# Requirements

## Use Case Diagram

Figure 2.1 is a use-case diagram for a Nursery Plant Shopping System, depicting the interactions between the Customer and the system.The customer can shop for plants, add or remove them from the cart, complete a purchase and manage orders, including making payments and viewing order details. They can also leave reviews and ratings, manage their wishlist, edit their profile, and change their password. The diagram visually represents the system's functionalities available to the customer.

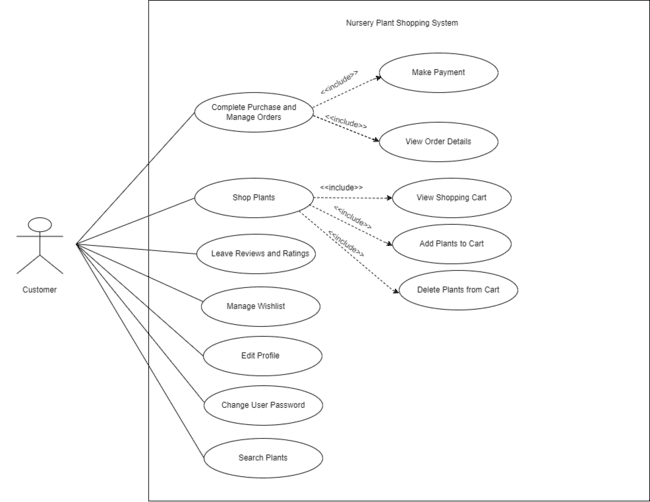
**

Figure 2.1: Nursery Plant Shopping System Customer’s Use Case Diagram

## Class Diagram

Figure 2.2 presents the class diagram for a Nursery Plant Shopping System from the customer's viewpoint, showcasing the `Customer` class with personal and authentication attributes and methods for profile and shopping cart management. It is connected to key classes like `User`, indicating inheritance, `ShoppingCart`, and `Wishlist` for managing shopping activities, `Plant` for the items on sale, `Review` for feedback, and `Order` with `OrderItem` for purchase tracking, all linked with appropriate relationships and multiplicities to depict the system's object-oriented structure for a comprehensive customer shopping experience.

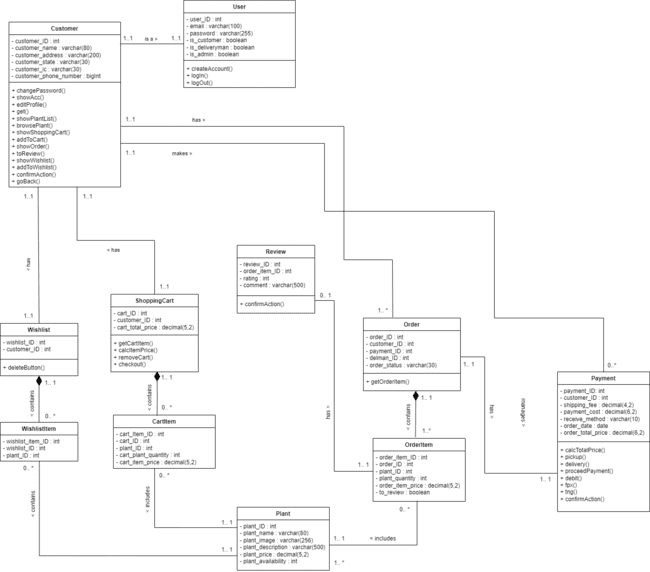


Figure 2.2: Nursery Plant Shopping System Customer’s Class Diagram

## State Diagrams

Figure 2.3 illustrates the State Transition Diagram for a Nursery Plant Shopping System. The user will begin with the “TLET Nursery Plant Shopping System Screen” as the initial state, from which users can navigate to the “Login Screen”. Upon successful login, the system transitions to different home screens based on the user's role which are “Administrator Home Screen” for admin login, “Customer Home Screen” for customer login, and “Delivery Man Home Screen” for delivery personnel login. Each home screen provides a “Logout” transition leading back to the “Login Screen”.

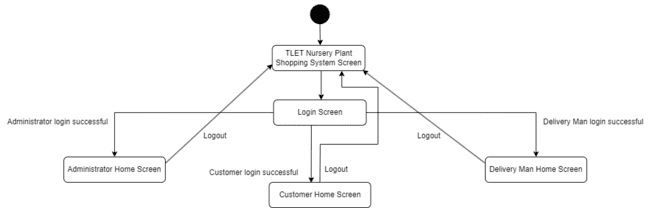


Figure 2.3: State Transition Diagram for Nursery Plant Shopping System

### 2.3.1 Customer State Transition Diagram

Figure 2.4 illustrates the State Transition Diagram for Customers within the TLET Nursery Plant Shopping System. Beginning at the “TLET Nursery Plant Shopping System Screen”, a customer can access the “Login Screen” and proceed to the “Customer Home Screen” upon successful login. From there, the customers have various selections to proceed such as selecting the “Plant List section” which will lead to the “Plant List Screen” and the customers can “Search plant name”, “Add to cart”, or “Add to wishlist”. The “Account Setting section” leads to the “Account Setting Screen” where “Change Password” and “Edit Profile” options are available. The “Wishlist section” takes the customer to the “Wishlist Screen”, and the “Shopping Cart section” to the “Shopping Cart Screen”, where customers can “View added plant”, “Adjust plant quantity”, or “Delete plants”. Following these actions, the “Checkout button” navigates to the “Checkout Screen”. From here, the process diverges based on the chosen method of receiving the order. If the customer selects “Pickup”, they are taken directly to the “Make Payment Screen”. Alternatively, if the customer selects the “Delivery button”, they are led to the “Delivery Details Screen”, where, after checking the necessary details, the customer can click the “Proceed to Payment button” to reach the “Make Payment Screen”. After completing payment, customers will be directed to the “Plant List Screen”. Moreover, customers can review their purchases by clicking on the “My Orders section” which will direct them to the “My Orders Screen”. Customers can submit reviews by pressing the “To Review section” which leads to the “To Review Screen”.

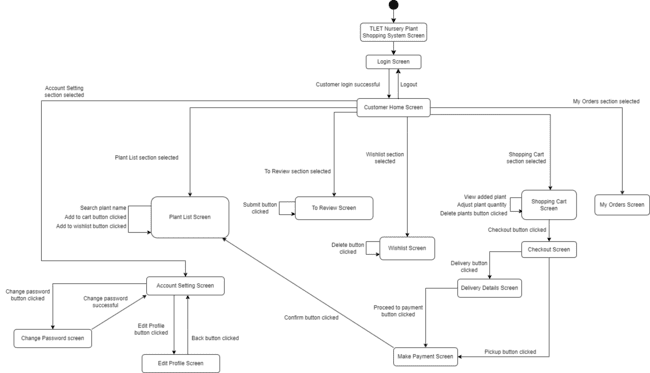


Figure 2.4: State Transition Diagram for Customer

## Data Flow Diagrams

Figure 2.5 illustrates a data flow diagram for a Nursery Plant Shopping System, highlighting the intricate processes and data exchanges between different entities such as the customer, guest, delivery man, and administrator. Central to the system are databases for users, plants, cart items, wishlists, payments, reviews, deliveries, and orders, which interact with various functions like creating accounts, editing profiles, managing shopping carts and wishlists, searching and updating plant information, processing payments, and handling order deliveries. The diagram meticulously maps out the flow of data through the system, detailing the inputs and outputs of each process, and showcasing the system's capability to handle both customer and administrative operations within a nursery plant retail environment.

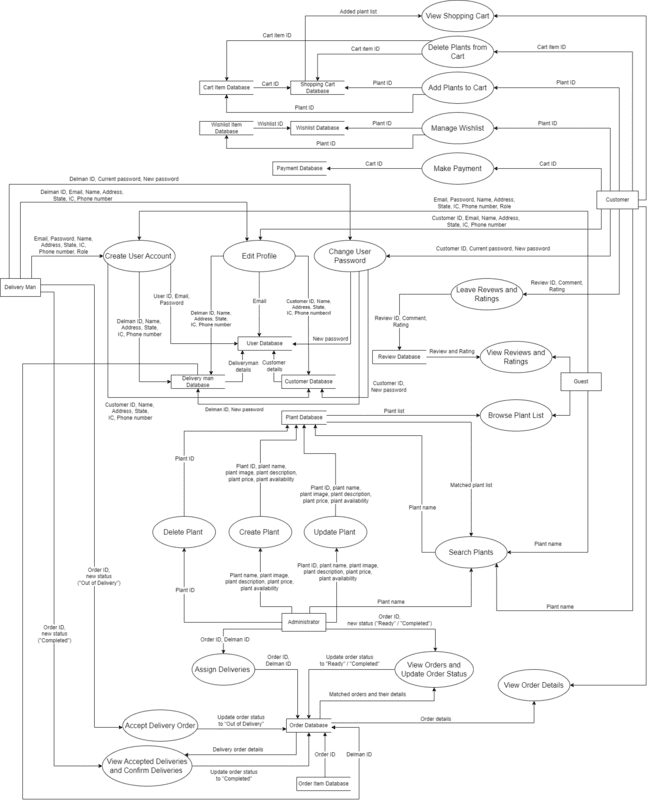


Figure 2.5: Data Flow Diagram for Nursery Plant Shopping System

# Design

## Use Cases

### 3.1.1 A customer changes their password.

Table 3.1: A Customer Changes Their Password Use Case Description

| Use Case Name: | A customer changes their password. |
| --- | --- |
| Description: | A customer decides to update their account security by changing their password. Upon clicking the “Change Password” button within the “Account Settings”, the customer is directed to the “Change Password” page. The customer must enter their old password, choose a new password, and then re-enter the new password for confirmation. The new password creation is guided by four rules: it must not be too similar to the user's personal information, must be at least 8 characters long, cannot be a commonly used password, and cannot be entirely numeric. Once the fields are completed, the customer can submit the new password by clicking the blue "Confirm" button. The system updates the account, and confirms the successful password update. |
| Primary Actor: | Customer |
| Precondition | * The customer is logged into the system. * The customer desires to change their current password. |
| Postcondition | * The customer’s password is successfully changed, and the customer can continue using the system with the new password. |
| Main Success Scenario: | S01: The customer navigates to the “Account Settings” section within the system.  S02: The system displays account settings, including detailed information such as full name, email, phone number, address, state, and IC No, along with the “Change Password” button.  S03:The customer clicks on the “Change Password” button.  S04:The system directs the customer to the “Change Password” page, featuring fields for entering the old password, new password and re-enter the new password for confirmation .  S05:The customer enter their existing password and create a new one, adhering to specified security requirements.  S06:The customer enters their old password for verification and crafts a new password, ensuring it aligns with the four outlined security rules: not resembling personal info, at least 8 characters in length, not a common password, and not purely numerical.  S07:Following the password creation, the customer confirms the new password by re-entering it in the confirmation field.  S06:Upon completing the fields, the customer proceeds by clicking the “Confirm” button to submit the new password.  S07:The system validates the old password, checks the new password against security standards, and, if all criteria are met, processes the password change.  S08: The system updates the customer’s account.  S09:The customer is then redirected back to the ‘Account Settings’ page by displays a confirmation message that the password has been changed and updated. |
| Alternative Scenario: | - |
| Exception Scenario: | A01: @S06 The old password entered is incorrect.  A01.07: The system displays an error message indicating that the entered old password was entered incorrectly.  A01.08: The customer re-enters the old password.  A01.09: RESUME @S05.  A02: @S06 The new password does not meet four outlined security rules.  A02.07: The system displays an error message indicating that the chosen password does not meet the outlined security rules.  A02.08: The customer enters a new password.  A02.09: RESUME @S05. |

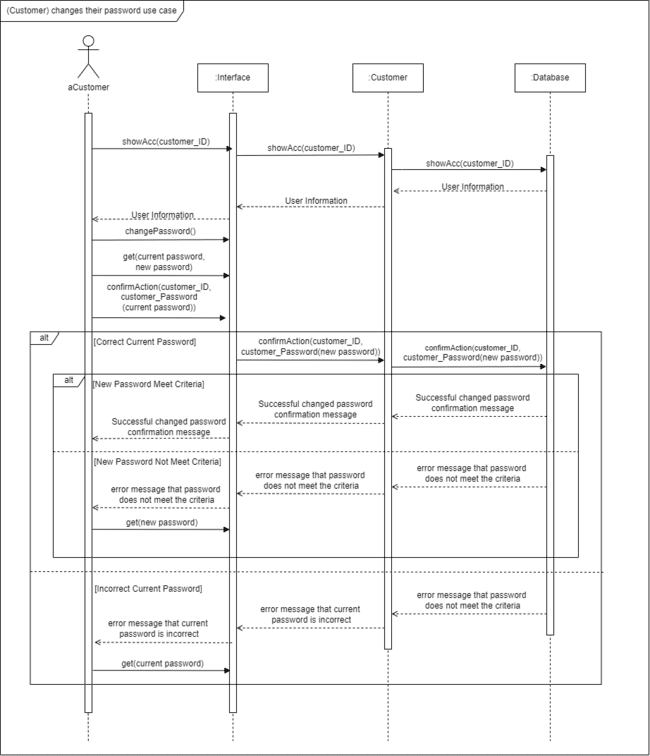


Figure 3.1: A Customer Changes Password Sequence Diagram

### 3.1.2 A customer edits profile details.

Table 3.2: A Customer Edits Profile Details Use Case Description

| Use Case Name: | A customer edits profile details |
| --- | --- |
| Description: | When a customer wishes to revise their profile information, they navigate to the “Account Settings” section of their account. Upon selecting the “Edit Profile” option, they are presented with a form that allows them to update personal details. The customer can make necessary changes and finalize the update by clicking the “Confirm” button. |
| Primary Actor: | Customer |
| Precondition | * The customer is logged into the system. * The customer desires to change their profile details. |
| Postcondition | * The customer’s profile details have been successfully updated in the system. |
| Main Success Scenario: | S01: The customer navigates to the “Account Settings” section within the system.  S02: The system displays account settings, including detailed information such as full name, email, phone number, address, state, and IC No, along with the “Edit Profile” button.  S03: The customer clicks on the “Edit Profile” button.  S04: The system directs the customer to the “Edit Profile” page, featuring fields to modify the customer's personal information.  S05: The customer modifies the necessary information in the fields provided.  S06: The customer submits the updates by clicking the “Confirm” button.  S07: The system validates and saves the updated information.  S08: The system confirms the updates with a message to the customer. |
| Alternative Scenario: | A01: @S04 The customer decides to cancel the update.  A01.05: The customer clicks the “Back” button without saving changes.  A01.06: The customer is then redirected back to the ‘Account Settings’ page. |
| Exception Scenario: | A02: @S05The system detects the invalid email format.  A02.06: The system displays an error message next to the email field, prompting the customer to correct the information.  A02.07: The customer corrects the information.  A02.08: RESUME @S06.  A02: @S05 The system detects the full name that does not meet the specified length requirements.  A02.06: The system then responds by displaying an error message, informing the customer that the full name must be between 10 and 50 characters long.  A02.07: The customer notices the error message and corrects their full name to adhere to the required character length.  A02.08: RESUME @S06.  A02: @S05 The system detects the phone number entered does not meet the specified format requirements.  A02.06: The system then responds by displaying an error message, informing the customer that the phone number should start with "60" and be followed by 9 or 10 digits.  A02.07: The customer notices the error message and corrects their phone number to adhere to the specified format requirements.  A02.08: RESUME @S06.  A02: @S05 The system detects the invalid state.  A02.06: The system then responds by displaying an error message, informing the customer enter a valid state.  A02.07: The customer notices the error message and enter a valid state.  A02.08: RESUME @S06.  A02: @S05 The system detects the IC number entered does not consist of 12 digits.  A02.06: The system then responds by displaying an error message, informing the customer that the phone number should must consist of 12 digits.  A02.07: The customer notices the error message and corrects their IC number.  A02.08: RESUME @S06. |

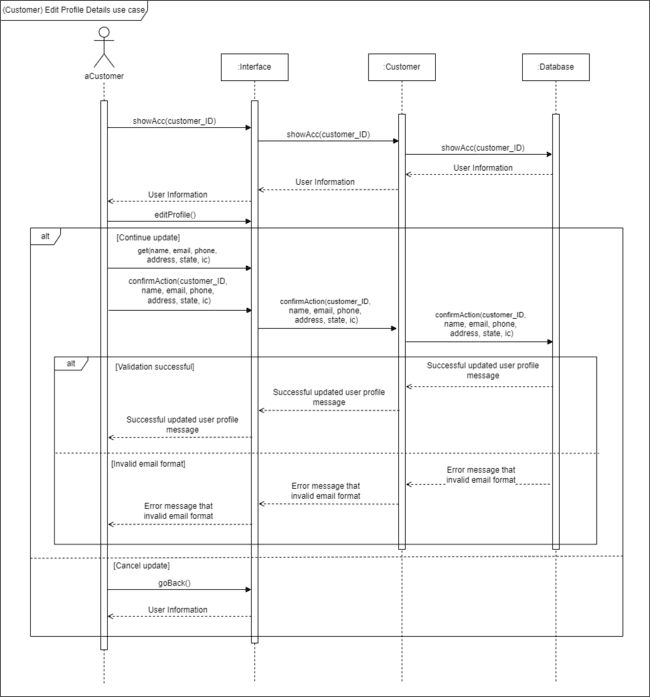


Figure 3.2: A Customer Edits Profile Details Sequence Diagram

### 3.1.3 A customer shops plants.

Table 3.3: A Customer Shops Plants Use Case Description

| Use Case Name: | A customer views added plants in the shopping cart. |
| --- | --- |
| Description: | Customers can view the plants they have added to their shopping cart, along with the plant image, plant name, plant ID, total price and quantity by navigating to the “Shopping Cart” section. |
| Primary Actor: | Customer |
| Precondition | * The customer is logged into the system. |
| Postcondition | * The customer has successfully viewed the plants added to the shopping cart and their details. |
| Main Success Scenario: | S01: The customer navigates to the “Shopping Cart” section within the system.  S02: The system displays a list of plants with detailed information including plant ID, a thumbnail image of the plant, the plant name, plant ID, total price, and quantity. |
| Alternative Scenario: | - |
| Exception Scenario: | A01: @S01 The shopping cart is empty.  A01.02: The system displays a message indicating that the shopping cart is empty.  A01.03: END |
| Use Case Name: | A customer adds plants to the shopping cart. |
| Description: | The customer navigates to the “Plant List” section within the system, where they can view each plant's ID, name, image, reviews, ratings, description, price, and availability. When they find a plant they wish to buy, they can select it and add it to their shopping cart with a simple click of the "Add to Cart" button. The system immediately updates to reflect the addition. |
| Primary Actor: | Customer |
| Precondition | * The customer is logged into the system. |
| Postcondition | * The customer has successfully added the selected plants to their shopping cart. |
| Main Success Scenario: | S01: The customer navigates to the “Plant List” section within the system.  S02: The system displays a list of plants, each entry containing detailed information including the plant ID, plant name, plant image, reviews, rating, description, price, and availability.  S03: The customer chooses the plant they are interested in purchasing.  S04: The customer clicks the “Add to Cart” button for the chosen plants.  S05: The system processes the addition and confirms that the selected plants have been added to the customer's shopping cart. |
| Alternative Scenario: | A01: @S03 If the customer changes their mind and decides not to add the selected plants to the cart.  A01.04: Resume@S01 |
| Exception Scenario: | - |
| Use Case Name: | A customer deletes plants from the shopping cart. |
| Description: | Customers can easily manage their shopping cart. They see a list of their chosen plants, each with the plant’s picture, name, ID, price, and quantity. If they want to remove something out of the cart, They just click the "Remove" button for the chosen plants. The system will remove and update the cart. |
| Primary Actor: | Customer |
| Precondition | * The customer is logged into the system. * The customer has previously added plants to their shopping cart. |
| Postcondition | * The customer has successfully removed the selected plants from their shopping cart. |
| Main Success Scenario: | S01: The customer navigates to the “Shopping Cart” section within the system.  S02: The system displays a list of plants in the cart, including detailed information such as plant ID, a thumbnail image of the plant, the plant name, total price, and quantity.  S03: The customer identifies the plants they wish to remove from the cart.  S04: The customer clicks the “Remove” button for the chosen plants.  S05: The system processes the deletion and confirms that the selected plants have been removed from the customer's shopping cart. |
| Alternative Scenario: | A01: @S03 The customer decides not to remove any plants.  A01.04:RESUME@S01 |
| Exception Scenario: | - |

## 

Figure 3.3: Sub-use cases for Shop Plants Use Case

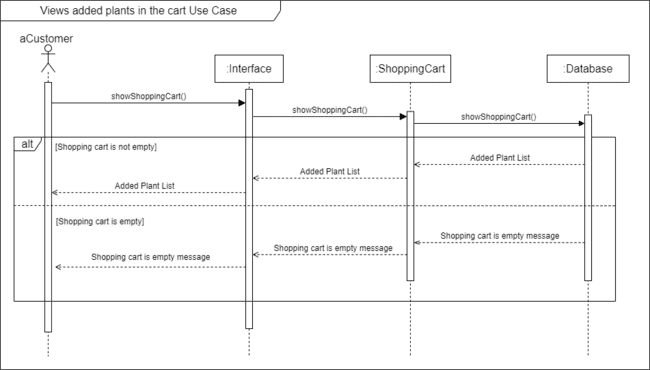


Figure 3.4:View Added Plant in the Cart Sequence Diagram

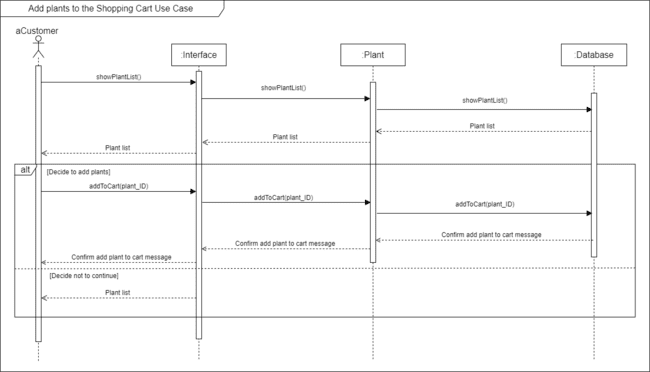


Figure 3.5:Add Plant to the Shopping Cart Sequence Diagram

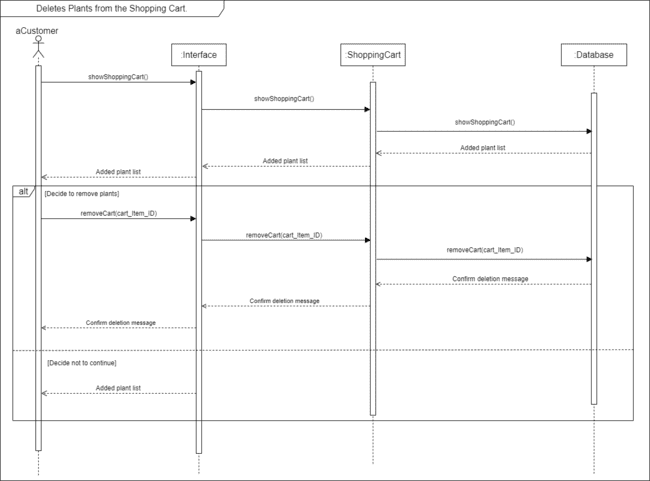


Figure 3.6: Delete Plant from the Shopping Cart Sequence Diagram

### 3.1.4 A customer searches a plant by plant name.

Table 3.4: A Customer Searches A Plant by Plant Name Use Case Description

| Use Case Name: | A customer searches a plant by plant name. |
| --- | --- |
| Description: | In the “Plant List” section, customers can perform a search by entering plant names into the search bar and clicking the “Search” button. The system processes the search query and lists matching plants, showing their ID, image, name, rating, description, price, availability, and reviews. |
| Primary Actor: | Customer |
| Precondition | * The customer is logged into the system. |
| Postcondition | * The customer has successfully searched for and viewed information about a plant by its name. |
| Main Success Scenario: | S01: The customer navigates to the “Plant List” section within the system.  S02: The system displays a list of plants along with a search interface that includes a text input field for the search term.  S03: The customer enters the name of the plant they are interested in into the search bar.  S04: The customer clicks on the “Search” button.  S05: The system processes the search query and displays the results that match the entered plant name.  S06: For each matching result, the system shows detailed information including the plant ID, a thumbnail image of the plant, the plant name, its rating, description, price, availability, and reviews. |
| Alternative Scenario: | - |
| Exception Scenario: | A01: @S05 No matching plants are found based on the entered criteria.  A01.06: The system displays a message indicating that no plants match the entered criteria.  A01.07: The customer refines the search by adjusting the plant name.  A01.08: RESUME @S03 |

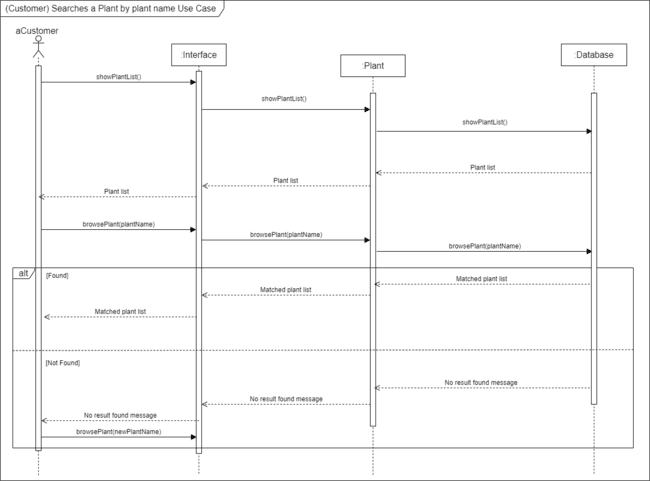


Figure 3.7: Search a Plant by Plant Name Sequence Diagram

### 3.1.5 A customer completes purchases and manages orders.

Table 3.5: A Customer Completes Purchases and Manages Orders Use Case Description

| Use Case Name: | A customer makes a payment. |
| --- | --- |
| Description: | The customer finalizes their plant purchase by navigating to the “Shopping Cart” section and reviewing their selected items, including names, images, and prices. Opting to check out, they are presented with delivery and pickup options. Selecting delivery, they review their details on the following page and proceed to payment. By selecting pickup, they will be informed to pick up from the shop within 14 days and proceed to the payment page. On the payment page, the customer selects their preferred method, and confirms the transaction. Once the payment is processed, the system confirms the successful order. |
| Primary Actor: | Customer |
| Precondition | * The customer is logged into the system. * The customer has previously added plants to their shopping cart. |
| Postcondition | * The customer has completed the payment process and the system has generated a new order. |
| Main Success Scenario: | S01: The customer navigates to the “Shopping Cart” section within the system.  S02: The system displays a list of plants in the cart, including detailed information such as plant ID, a thumbnail image of the plant, the plant name, total price, and quantity.  S03: The customer clicks the “Checkout” button for the chosen plants.  S04: The customer is directed to the “Checkout” page.  S05: The system displays 2 options which are delivery and pickup.  S06: The customer chooses delivery as an option.  S07: The customer is directed to the “Delivery Details” page.  S08: The system displays a list of customer info including detailed information such as full name, delivery address, plant price, shipping fee, and total cost.  S09: The customer clicks the “Proceed to payment” button to make the payment.  S10: The customer is directed to the “Make Payment” page.  S11: The system displays a list of payment info including detailed information such as receiving item via pickup/delivery, total cost(including shipping fee if applicable), and payment method.  S12: The customer clicks the “Credit/Debit” button to make the payment.  S13: The customer clicks the “Confirm” button to confirm the transaction.  S14: The system processes the payment and generates a new order, confirming the successful transaction to the customer.  S15: The customer is redirected back to the Plant List page. |
| Alternative Scenario: | A01: @S12 The customer selects FPX (Internet Banking) as the payment method.  A01.13: Resume@S13.  A02: @S12 the customer clicks the Touch `N Go E-wallet to make the payment.  A02.13: Resume@S13. |
| Exception Scenario: | A03: @S01 If the shopping cart is empty.  A03.02: The system indicates the customer with a message that the cart is empty.  A03.03: When the customer clicks the "Checkout" button, the message box appears. If the customer then selects the "Go to Plant List" button, they are redirected to the plant list page to select items for purchase.  A03.04:Resume@S01 |
| Use Case Name: | A customer views order details of ongoing orders. |
| Description: | Customers can view the current status of their orders by visiting the “My Orders” section after logging into their account. The system presents a detailed list of all orders in progress, including the order ID, the names, quantities, and prices of the plants ordered, the order date, the total cost, and the status of each order. |
| Precondition | * The customer is logged into the system. * The customer has placed one or more orders that are currently being processed. |
| Postcondition | * The customer has successfully viewed the details of their ongoing plant order and can take appropriate actions based on the information obtained. |
| Main Success Scenario: | S01: The customer navigates to the “My Orders” section within the system.  S02: The system displays a list of ongoing orders with detailed information for each ongoing order, including orderID, plant name, quantity, price, order date, total price, and order status. |
| Alternative Scenario: | - |
| Exception Scenario: | A01: @S01 If there are no ongoing orders to display.  A01.02: The system shows a message indicating there are no current orders.  A01.03: END |

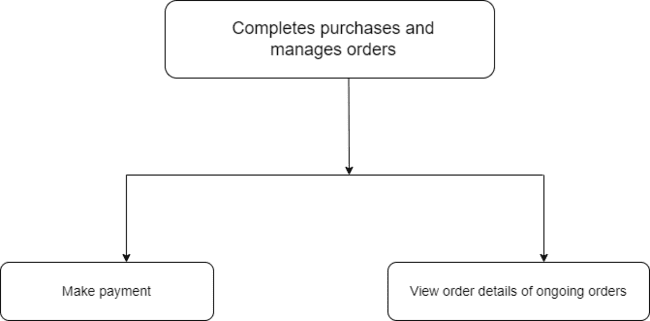


Figure 3.8: Sub-use cases for Complete Purchases and Manage Orders Use Case

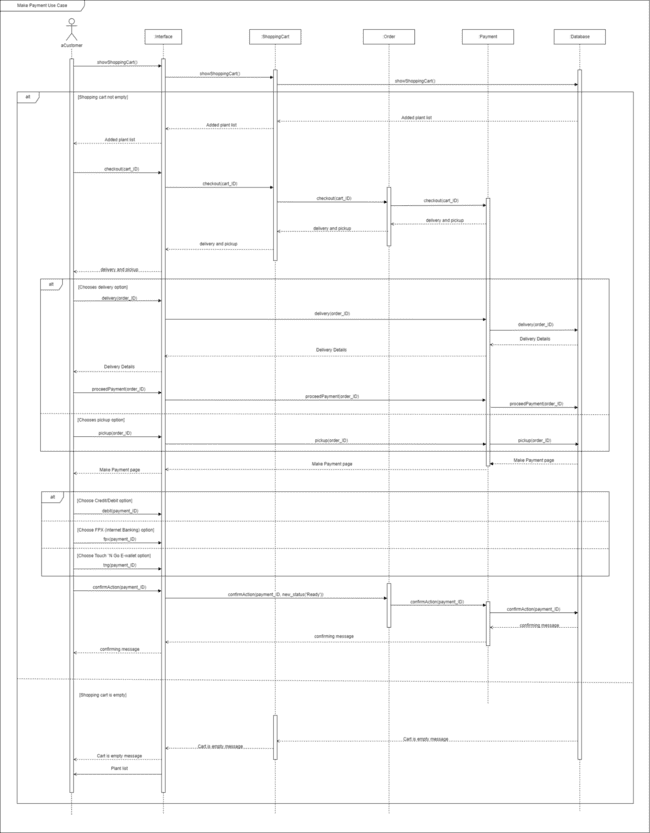


Figure 3.9: Make Payment Sequence Diagram

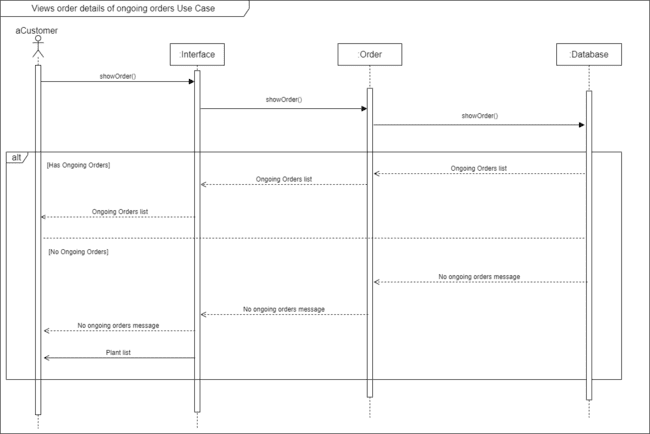


Figure 3.10: View Order Details of Ongoing Orders Sequence Diagram

### 3.1.6 A customer leaves reviews and ratings

Table 3.6: A Customer Leaves Reviews and Ratings Use Case Description

| Use Case Name: | A customer leaves reviews and ratings. |
| --- | --- |
| Description: | The customer navigates to the “To Review” section of their account to provide feedback on their plant purchases. The system shows each plant awaiting review with an image and fields for rating and written feedback. Customers rate the plants on a 1-5 star scale and add comments before submitting their reviews. |
| Primary Actor: | * Customer |
| Precondition | * The customer is logged into the system. * The customer has made purchases and has items available to review in their account. |
| Postcondition | * The customer has submitted their reviews and ratings for the plants they purchased. |
| Main Success Scenario: | S01: The customer navigates to the “To Review” section within the system. S02: The system displays the info of review with detailed information including plant name, rating, review, and plant image.  S03: For each plant listed, the customer is presented with a dropdown menu to select a rating and a text box to write a review for the plant displayed.  S04: The customer selects a star rating from the "Rate the plant" dropdown menu and writes their feedback in the review text box .  S05: After completing the review and rating, the customer submits their feedback by clicking the "Submit" button. |
| Alternative Scenario: | - |
| Exception Scenario: | A01: @S01 If there are no items to review..  A01.02: The system shows a message indicating there are no items to review.  A01.03: END  A01: @S04 The customer tries to submit without choosing a rating.  A01.05:The system displays a message indicating that the rating is required.  A01.06: Resume @S04 |

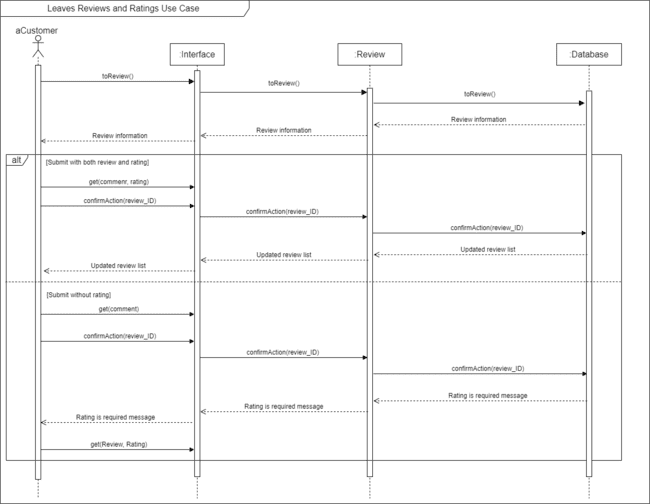


Figure 3.11: Leaves Reviews and Ratings Sequence Diagram

### 3.1.7 A customer manages their wishlist.

Table 3.7: A Customer Manages A Wishlist Use Case Description

| Use Case Name: | A customer manages their wishlist. |
| --- | --- |
| Description: | The customers can view all the plants they have marked for potential future purchase. Each listed item includes the plant's name, an image, and its price. |
| Primary Actor: | * Customer |
| Precondition | * The customer is logged into the system. * The customer has added one or more plants to their Wishlist. |
| Postcondition | * The customer has successfully updated their wishlist |
| Main Success Scenario: | S01: The customer navigates to the “Wishlist” section within the system  S02: The system displays the info of the wishlist with detailed information including plant name, plant image, and price.  S03: To remove a plant from their Wishlist, the customer clicks the “Delete” button next to the item they wish to remove. |
| Alternative Scenario: | - |
| Exception Scenario: | A01: @S01 The customer has not added any plants to the wishlist.  A01.02:The system displays a message indicating that the wishlist is empty.  A01.03:The customer clicks the "Go to Plant List" button, they are redirected to the plant list page to select items to the wishlist.  A01.04:RESUME @S01 |

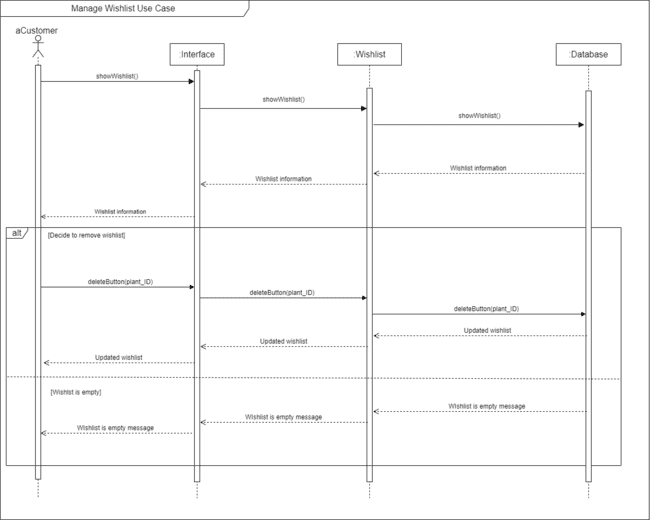


Figure 3.12: Manage Wishlist Sequence Diagram

## Data Dictionary

The User Table Data Dictionary, shown in Table 3.8, is a key component for organizing user information in the Nursery Plant Shopping System. It details essential attributes like user unique identifier(user\_ID), which uniquely identifies each user, user email (email), and user password (password) for login credentials, and role-defining booleans (is\_customer, is\_deliveryman, and is\_admin) to distinguish between customers, delivery personnel, and administrators. The user\_ID serves as the primary key, ensuring that each record is unique and easily retrievable.

Table 3.8: User Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| User | user\_ID | user ID | int | 1 | 1-9999 | Y | PK | - |
| email | user email | varchar(100) | eelin@gmail.com | - | Y | - | - |
| password | user password | varchar(255) | 123456 | - | Y | - | - |
| is\_customer | is customer | boolean | True | - | Y | - | - |
| is\_deliveryman | is deliveryman | boolean | True | - | Y | - | - |
| is\_admin | is administrator | boolean | True | - | Y | - | - |

The Customer Table Data Dictionary shown as Table 3.9 provides a comprehensive overview of the attributes and specifications for managing customer records within the database. It includes essential details such as the customer's unique identifier (customer\_ID), their name (customer\_Name), address (customer\_Address), state (customer\_State), identification card number (customer\_IC), and phone number (customer\_phone\_number). These attributes are essential for customer account management and interactions within the system. The data dictionary also identifies customer\_ID as the primary key (PK) for efficient record referencing within the table.

Table 3.9: Customer Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Customer | customer\_ID | customer ID | int | 91000 | 91000-94999 | Y | PK | - |
| customer\_Name | customer name | varchar(80) | Chin Yu Feng | - | Y | - | - |
| customer\_Address | customer address | varchar(200) | 20, Taman Maju, Section 3/2a, Cheras | - | Y | - | - |
| customer\_State | customer state | varchar(30) | Selangor | - | Y | - | - |
| customer\_IC | customer’s identification card number | varchar(30) | 040202016450 | - | Y | - | - |
| customer\_phone\_number | customer phone number | bigInt | 60128457865 | - | Y | - | - |

The Plant Table Data Dictionary shown as Table 3.10 serves as a comprehensive reference guide for managing plant records within the database. It encompasses key attributes such as the plant's unique identifier (plant\_ID), its name (plant\_Name), image (plant\_Image), description (plant\_Description), price (plant\_Price), and available amount (plant\_Availability). These attributes are crucial for tracking and managing plant inventory within the system. The “plant\_Image” attribute stores the file path of the plant image as a string, using varchar(256) and provides an example of the stored value (folder/image1.jpg) for reference. The data dictionary identifies plant\_ID as the primary key (PK), facilitating efficient record referencing within the table.

Table 3.10: Plant Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Plant | plant\_ID | plant ID | int | 10000 | 10000-19999 | Y | PK | - |
| plant\_Name | plant name | varchar(80) | Aloe Vera | - | Y | - | - |
| plant\_Image | plant image | varchar(256) | folder/image1.jpg | - | Y | - | - |
| plant\_Description | plant description | varchar(500) | Aloe vera is a succulent plant species of the genus Aloe. The plant is stemless with thick, greenish, fleshy leaves that can fan out from the plant’s central stem. The margin of the leaf is serrated with small teeth. | - | Y | - | - |
| plant\_Price | plant price | decimal(5,2) | 10.00 | - | Y | - | - |
| plant\_Availability | plant available amount | int | 100 | 1-9999 | Y | - | - |

The ShoppingCart Table Data Dictionary shown as Table 3.11 provides a comprehensive overview of the attributes and specifications for managing shopping cart records within the system. It contains attributes such as the shopping cart's unique identifier (cart\_ID), linked to a customer's unique identifier (customer\_ID), and the total cost of items in the shopping cart (cart\_total\_price). These attributes are important for tracking and managing customers' shopping activities. The data dictionary designates cart\_ID as the primary key (PK), and identifies customer\_ID as a foreign key (FK) referencing the “Customer” table, enabling efficient association with customer profiles.

Table 3.11: Shopping Cart Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ShoppingCart | cart\_ID | shopping cart ID | int | 20000 | 20000-29999 | Y | PK | - |
| customer\_ID | customer ID | int | 91000 | 91000-94999 | Y | FK | Customer |
| cart\_total\_price | shopping cart total price | decimal(5,2) | 120.50 | 0.00-999.99 | Y | - | - |

The Cart Item Table Data Dictionary shown as Table 3.12 provides an overview for managing individual items within a shopping cart. It contains attributes such as the cart item's unique identifier (cart\_Item\_ID), the associated shopping cart's identifier (cart\_ID), the plant's identifier (plant\_ID), the quantity of the plant in the cart (cart\_Plant\_Quantity), and the price of the cart item (cart\_Item\_Price). These attributes are crucial for tracking and managing the specific items customers have added to their shopping carts. The data dictionary designates cart\_Item\_ID as the primary key (PK), and identifies cart\_ID as a foreign key (FK) referencing the “ShoppingCart” table, linking between shopping cart items and their respective carts. Similarly, the “plant\_ID” attribute is a foreign key (FK) referencing the “Plant” table, connecting each cart item to a specific plant in the system.

Table 3.12: Cart Item Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CartItem | cart\_Item\_ID | cart item ID | int | 30000 | 30000-39999 | Y | PK | - |
| cart\_ID | cart ID | int | 20000 | 20000-29999 | Y | FK | ShoppingCart |
| plant\_ID | plant ID | int | 10000 | 10000-19999 | Y | FK | Plant |
| cart\_Plant\_Quantity | plant quantity | int | 5 | 1-50 | Y | - | - |
| cart\_Item\_Price | cart item price | decimal(5,2) | 100.00 | 0.00-999.99 | Y | - | - |

The Wishlist Table Data Dictionary shown as Table 3.13 provides an overview for managing wishlists associated with customer profiles. It encompasses key attributes, such as the wishlist's unique identifier (wishlist\_ID), and the customer's identifier (customer\_ID). These attributes are used for tracking and managing customer wishlists within the system. The data dictionary designates wishlist\_ID as the primary key (PK), and identifies customer\_ID as a foreign key (FK) referencing the “Customer” table, establishing a direct link between wishlists and individual customers.

Table 3.13: Wishlist Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Wishlist | wishlist\_ID | wishlist ID | int | 40000 | 40000-49999 | Y | PK | - |
| customer\_ID | customer ID | int | 91000 | 91000-94999 | Y | FK | Customer |

The Wishlist Item Table Data Dictionary shown as Table 3.14 provides an overview for managing individual items in customer wishlists. This table features key attributes like the wishlist item’s unique identifier (wishlist\_item\_ID), the identifier for the corresponding wishlist (wishlist\_ID), and the identifier of the plant associated with each item (plant\_ID). These attributes are crucial for tracking and managing the specific items customers have added to their wishlists. The data dictionary designates wishlist\_Item\_ID as the primary key (PK), and identifies wishlist\_ID as a foreign key (FK) referencing the "Wishlist" table, which solidifies the connection between wishlist items and their respective wishlists. Similarly, the plant\_ID, also a foreign key (FK), refers to the "Plant" table, tying each wishlist item to a particular plant.

Table 3.14: Wishlist Item Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| WishlistItem | wishlist\_item\_ID | wishlist item ID | int | 50000 | 50000-59999 | Y | PK | - |
| wishlist\_ID | wishlist ID | int | 40000 | 40000-49999 | Y | FK | Wishlist |
| plant\_ID | plant ID | int | 10000 | 10000-19999 | Y | FK | Plant |

The Payment Table Data Dictionary shown as Table 3.15 provides an overview for managing payment transactions within the system. It contains attributes such as the payment's unique identifier (payment\_ID), the associated customer's identifier (customer\_ID), shipping fee (shipping\_Fee), total cost for the payment (payment\_Cost), the method for receiving the order (receive\_Method), the date of order placement (order\_Date), and the order's total price (order\_total\_price). These attributes are crucial for tracking and managing payment-related information and order details within the system. The data dictionary designates payment\_ID as the primary key (PK), and identifies customer\_ID as a foreign key (FK) referencing the “Customer” table, creating a direct link between payments and individual customers.

Table 3.15: Payment Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Payment | payment\_ID | payment ID | int | 60000 | 60000-69999 | Y | PK | - |
| customer\_ID | customer ID | int | 91000 | 91000-94999 | Y | FK | Customer |
| shipping\_Fee | shipping fee | decimal(4,2) | 5.00 | 0.00-99.99 | Y | - | - |
| payment\_Cost | total cost for payment | decimal(6,2) | 200.00 | 0.00-9999.99 | Y | - | - |
| receive\_Method | method to receive order | varchar(10) | Delivery | - | Y | - | - |
| order\_Date | date of order placement | date | YYYY-MM-DD | - | Y | - | - |
| order\_total\_price | order total price | decimal(6,2) | 200.00 | 0.00-9999.99 | Y | - | - |

The Order Table Data Dictionary shown as Table 3.16 serves as a reference guide for managing customer orders within the system. It encompasses essential attributes such as the order's unique identifier (order\_ID), the associated customer's identifier (customer\_ID), payment identifier (payment\_ID), delivery man identifier (delman\_ID), and the order status (order\_Status). These attributes are fundamental for tracking and managing customer orders and related order details. The data dictionary designates order\_ID as the primary key (PK), and identifies customer\_ID as a foreign key (FK) referencing the “Customer” table, creating a direct link between orders and individual customers. Similarly, payment\_ID and delman\_ID are foreign keys (FK) referencing the “Payment” and “DeliveryMan” tables, respectively, establishing connections with payment and delivery information.

Table 3.16: Order Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Order | order\_ID | order ID | int | 70000 | 70000-79999 | Y | PK | - |
| customer\_ID | customer ID | int | 91000 | 91000-94999 | Y | FK | Customer |
| payment\_ID | payment ID | int | 60000 | 60000-69999 | Y | FK | Payment |
| delman\_ID | delivery man ID | int | 90006 | 90006-90999 | Y | FK | DeliveryMan |
| order\_Status | order status | varchar(30) | Ready | - | Y | - | - |

The Order Item Table Data Dictionary shown as Table 3.17 gives an overview for managing individual items within customer orders in the system. It contains attributes such as the order item's unique identifier (order\_Item\_ID), the associated order's identifier (order\_ID), the plant's identifier (plant\_ID), the quantity of the plant in the order (plant\_Quantity), the price of the order item (order\_Item\_Price), and whether it needs to be reviewed (to\_Review). These attributes are used for tracking and managing detailed information about items within customer orders. The data dictionary designates order\_Item\_ID as the primary key (PK), and identifies order\_ID as a foreign key (FK) referencing the “Order” table, establishing a direct link between order items and their respective orders. Similarly, the “plant\_ID” attribute is a foreign key (FK) referencing the “Plant” table, connecting each order item to a specific plant in the system.

Table 3.17: Order Item Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| OrderItem | order\_Item\_ID | order Item ID | int | 80000 | 80000-89999 | Y | PK | - |
| order\_ID | order ID | int | 70000 | 70000-79999 | Y | FK | Order |
| plant\_ID | plant ID | int | 10000 | 10000-19999 | Y | FK | Plant |
| plant\_Quantity | plant Quantity | int | 5 | 1-50 | Y | - | - |
| order\_Item\_Price | order item price | decimal(5,2) | 100.00 | 0.00-999.99 | Y | - | - |
| to\_Review | to review | boolean | True | - | Y | - | - |

The Review Table Data Dictionary shown as Table 3.18 gives an overview for managing customer reviews within the system. It contains attributes such as the review's unique identifier (review\_ID), the identifier of the ordered item (order\_Item\_ID), the rating provided by the customer (rating), and any comments or feedback left in the review (comment). These attributes are crucial for tracking and managing customer feedback and their opinions about specific orders and items. The data dictionary designates review\_ID as the primary key (PK), and identifies order\_Item\_ID as a foreign key (FK) referencing the “OrderItem” table, creating a direct connection between reviews and specific order items.

Table 3.18: Review Table Data Dictionary

| Table Name | Attribute Name | Contents | Type | Format | Range | Required | PK/ FK | FK Referenced Table |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Review | review\_ID | review ID | int | 91000 | 91000-99999 | Y | PK | - |
| order\_Item\_ID | order Item ID | int | 80000 | 80000-89999 | Y | FK | OrderItem |
| rating | plant Rating | int | 5 | 1-5 | Y | - | - |
| comment | comment | varchar(500) | Receive in good condition. Looks very beautiful | - | Y | - | - |

## Data Structures

### User Array

Table 3.19 presents the User Array Data Structures, a fundamental element for managing user data within the Nursery Plant Shopping System. This array holds critical information for distinguishing between different user roles and their credentials. The user ID is an integer that uniquely identifies each user. The user email and user password are stored as varchar(100) and varchar(255) specifically for security and authentication purposes. The boolean flags (is customer, is delivery man, and is administrator) are to specify the user's role within the system. These attributes ensure that user data is neatly categorized and accessible, simplifying the process of identifying and managing users' interactions and responsibilities.

Table 3.19: User Array Data Structures

| Attribute | Data Type |
| --- | --- |
| user ID | int |
| user email | varchar(100) |
| user password | varchar(255) |
| is customer | boolean |
| is deliveryman | boolean |
| is administrator | boolean |

### Customer Array

Table 3.20 below shows Customer Array Data Structures. The Customer array serves as a fundamental data structure for managing crucial customer information within a system. The Customer ID, represented as an integer, acts as a unique identifier for each customer. Customer Name, a varchar(80), stores customer names, enhancing user recognition and presentation. Customer Address and Customer State, both varchars (200) and (30) respectively, record customer contact details and location information for order delivery. The Customer's Identification Card Number, a varchar(30), stores unique identification data, and the bigInt data type is used for the Customer Phone Number, enabling the storage of contact numbers.

Table 3.20: Customer Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Customer ID | int |
| Customer Name | varchar(80) |
| Customer address | varchar(200) |
| Customer state | varchar(30) |
| Customer’s identification card number | varchar(30) |
| Customer phone number | bigInt |

### Plant Array

Table 3.21 presents the Plant Array Data Structures. The Plant array serves as a fundamental data structure for organizing and managing essential information about plants within the system. The Plant ID, represented as an integer, provides a unique identifier for each plant, enabling precise tracking and reference. Plant Name, a varchar(80), offers a recognizable label for each plant, simplifying user identification. Plant Image, stored as a varchar(256), visually represents the plant through images. The Plant Description, with a varchar(500), provides valuable insights into each plant's characteristics, aiding users in making informed decisions. The Plant Price, formatted as a decimal(5,2), denotes the cost of each plant. Plant Available Amount, an integer, tracks the current quantity of each plant in stock, enabling inventory control.

Table 3.21: Plant Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Plant ID | int |
| Plant name | varchar(80) |
| Plant image | varchar(256) |
| Plant description | varchar(500) |
| Plant price | decimal(5,2) |
| Plant available amount | int |

### ShoppingCart Array

Table 3.22 presents the Shopping Cart Array Data Structures. The Shopping Cart array serves as an important data structure for efficiently managing shopping cart information within the system. The Shopping Cart ID, represented as an integer, provides a unique identifier for each cart, facilitating precise tracking and reference. Customer ID, also an integer, links each shopping cart to a specific customer, ensuring accurate association with individual shoppers. Shopping Cart Total Price, formatted as a decimal(5,2), records the cumulative cost of items within the cart.

Table 3.22: Shopping Cart Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Shopping cart ID | int |
| Customer ID | int |
| Shopping cart total price | decimal(5,2) |

### CartItem Array

Table 3.23 presents the Cart Item Array Data Structures. The Cart Item array serves as a foundational data structure for efficiently managing individual items within shopping carts in the system. Cart Item ID, represented as an integer, assigns a unique identifier to each cart item, ensuring precise tracking and reference. Cart ID, an integer, links each cart item to a specific shopping cart, facilitating accurate association with the corresponding cart. Plant ID, also an integer, records the unique identifier of the plant added to the cart, allowing for precise cart item selection. Plant Quantity, also an integer, captures the quantity of the plant in the cart, aiding in inventory management. Cart Item Price, formatted as a decimal(5,2), represents the cost of each cart item, enabling accurate pricing calculations.

Table 3.23: Cart Item Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Cart item ID | int |
| Cart ID | int |
| Plant ID | int |
| Plant quantity | int |
| Cart item price | decimal(5,2) |

### Wishlist Array

Table 3.24 presents the Wishlist Array Data Structures. The Wishlist array serves as a core data structure for managing customer wishlists within the system. Wishlist ID, represented as an integer, assigns a unique identifier to each wishlist, allowing for precise tracking and reference. Customer ID, also an integer, links each wishlist to a specific customer, ensuring accurate association with individual users.

Table 3.24: Wishlist Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Wishlist ID | int |
| Customer ID | int |

### WishlistItem Array

Table 3.25 presents the Wishlist Item Array Data Structures. The Wishlist Item array serves as a foundational data structure for efficiently managing individual items within wishlists in the system. Wishlist Item ID, represented as an integer, assigns a unique identifier to each wishlist item, ensuring precise tracking and reference. Wishlist ID, an integer, links each wishlist item to a specific wishlist, facilitating accurate association with the corresponding wishlist. Plant ID, also an integer, records the unique identifier of the plant added to the wishlist, facilitating user-specific plant selection and wishlist management.

Table 3.25: Wishlist Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Wishlist item ID | int |
| Wishlist ID | int |
| Plant ID | int |

### Payment Array

Table 3.26 presents the Payment Array Data Structures. The Payment array serves as a vital data structure for managing payment transactions within the system. Payment ID, an integer, assigns a unique identifier to each payment, ensuring precise tracking and reference. Customer ID, also an integer, links each payment to a specific customer, facilitating accurate association with corresponding customers. Shipping Fee, formatted as a decimal(4,2), captures the cost of shipping associated with the payment. Total Cost for Payment, a decimal(6,2), represents the overall cost encompassing the order and shipping fees. Method to Receive Order, a varchar(10), records the chosen method for receiving the order. Date of Order Placement, in date format, denotes the precise date when the order was placed, aiding in order scheduling. Order Total Price, formatted as a decimal(6,2), represents the total cost of the order, including products and excluding shipping fee.

Table 3.26: Payment Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Payment ID | int |
| Customer ID | int |
| Shipping fee | decimal(4,2) |
| Total cost for payment | decimal(6,2) |
| Method to receive order | varchar(10) |
| Date of order placement | date |
| Order total price | decimal(6,2) |

### Order Array

Table 3.27 presents the Order Array Data Structures. The Order array serves as a pivotal data structure for managing customer orders within the system. Order ID, represented as an integer, assigns a unique identifier to each order, facilitating precise tracking and reference. Customer ID, also an integer, links each order to a specific customer, ensuring accurate association with individual users. Payment ID, an integer, connects each order to a corresponding payment. Delivery Man ID, an integer, associates orders with specific delivery personnel. Order Status, a varchar(30), captures the current status of each order, which can be “waiting”, “ready”, “out of delivery” or “completed”.

Table 3.27: Order Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Order ID | int |
| Customer ID | int |
| Payment ID | int |
| Delivery man ID | int |
| Order status | varchar(30) |

### OrderItem Array

Table 3.28 presents the Order Item Array Data Structures. The Order Item array serves as a foundational data structure for efficiently managing individual items within customer orders in the system. Order Item ID, represented as an integer, assigns a unique identifier to each order item. Order ID, also an integer, links each order item to a specific order, facilitating accurate association with corresponding orders. Plant ID, an integer, records the unique identifier of the plant included in the order. Plant Quantity, also an integer, captures the quantity of the plant in the order, facilitating inventory management. Order Item Price, formatted as a decimal(5,2), represents the cost of each order item, enabling accurate pricing calculations. To Review, a boolean, indicates whether the order item requires review, streamlining customer review management within the system.

Table 3.28: Order Item Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Order Item ID | int |
| Order ID | int |
| Plant ID | int |
| Plant Quantity | int |
| Order item price | decimal(5,2) |
| To review | boolean |

### Review Array

Table 3.29 presents the Review Item Array Data Structures. The Review array serves as a comprehensive data structure for managing customer feedback and reviews within the system. Review ID, represented as an integer, assigns a unique identifier to each review. Order Item ID, also an integer, associates reviews with specific order items, enabling detailed feedback on purchased products. Plant Rating, an integer ranging from 1 to 5, with 1 indicating the lowest rating and 5 denoting the highest rating, allowing customers to rate the plant's quality. Comment, a varchar(500), allows customers to provide additional feedback about their experience, facilitating in-depth insights for product improvement and customer satisfaction within the system.

Table 3.29: Review Array Data Structures

| Attribute | Data Type |
| --- | --- |
| Review ID | int |
| Order Item ID | int |
| Plant Rating | int |
| Comment | varchar(500) |

## Subsystem Architecture

Figure 3.13 below is the Component-based Architecture Diagram for the Customer Subsystem. Figure 3.13 provides a clear visualization of the customer-oriented features within the system. Registered customers can personalize their experience through “Account Settings”, with options to “Edit Profile” and “Change Password”. Customers can browse plants through “Plant List” and efficiently find specific plants using the “Search Plant by Plant Name” feature and add desired plants to their wishlist or shopping cart. Customers can also view and track their orders in “My Orders” and submit feedback via the “To Review” component. Additionally, the “Wishlist” component allows customers to view their saved preferred plants for future consideration, while the “Shopping Cart” component enables them to “Add Plants”, “Adjust Plant Quantity”, and proceed to “Make Payment” for their selection of plants. This subsystem is designed to provide a seamless and user-friendly shopping experience, allowing customers to manage their interactions with the Nursery Plant Shopping System efficiently.

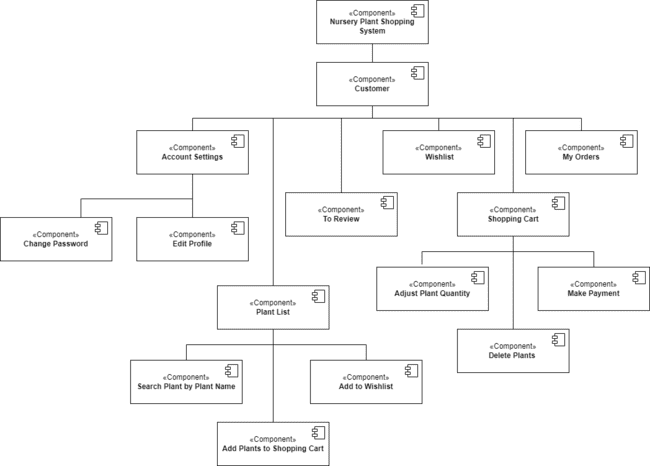


Figure 3.13: Component-based Architecture Diagram for Customer Subsystem

## Subsystem Screens

**3.5.1 Customer Home Screen**

Figure 3.14 shows 'Customer Home Screen', which displays the main interface for registered users on the TLET Nursery Plant Shopping System. This dashboard provides customers with a variety of options including adding items to their shopping cart or wishlist, accessing their orders, reviewing past purchases, managing account settings, and logging out. The central part of the screen highlights a plant's detailed information such as ID, name, image, description, price, and plant availability. Customers can also read existing reviews and ratings for the plants listed.

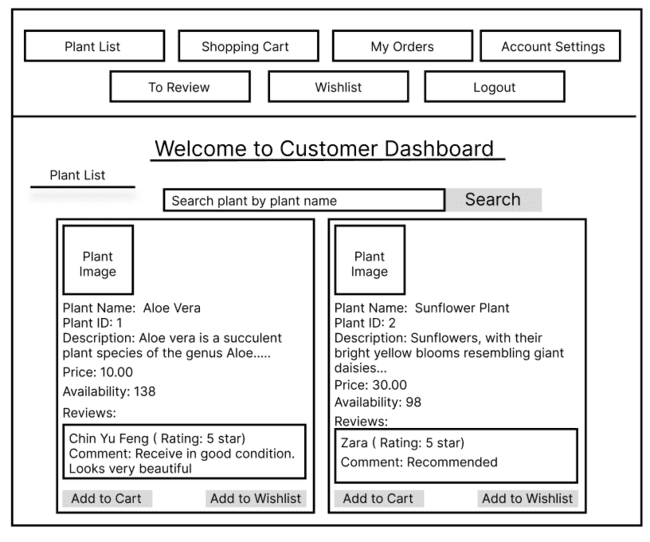
****

Figure 3.14: Customer Home Screen

**3.5.2 Customer Changes Their Password Screen**

Figure 3.16 shows the "Change Password Screen", which presents the interface for customers to update their password on the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 3.14, where the customer selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 3.15, where the customer's details are listed. By selecting the 'Change Password' button, the customer is directed to the interface shown in Figure 3.16, where they are required to enter their old password followed by the new password they wish to set. After submitting the new password by clicking 'Confirm', the system verifies the new password. If the update is successful, a confirmation message is displayed, and the customer is taken back to the "Account Settings Screen" shown in Figure 3.15, completing the password change process.

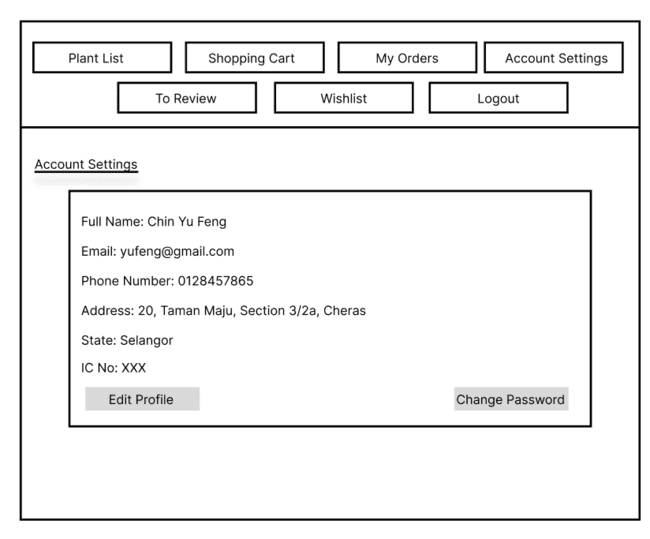
****

Figure 3.15: Account Settings Screen

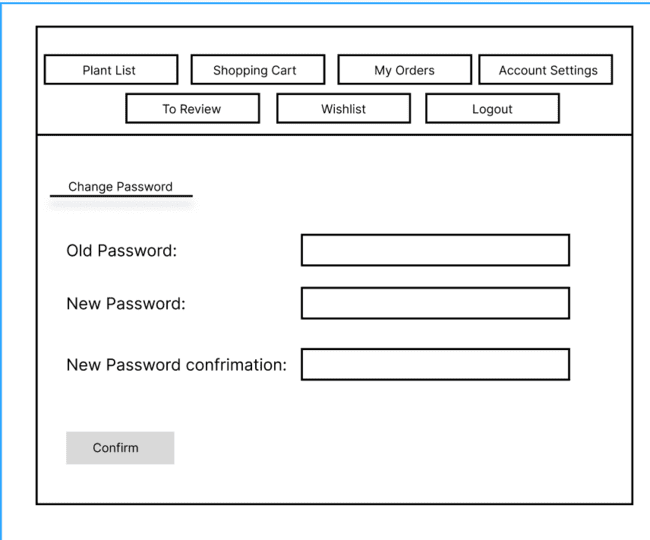
****

Figure 3.16: Change Password Screen

**3.5.3 Customer Can Edit Profile Details Screen**

Figure 3.17 shows 'Edit Profile Details Screen', where customers can update their personal information on the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 3.14, where the customer selects the 'Account Settings' button. This action takes them to the "Account Settings Screen'' displayed in Figure 3.15, where the customer's details are listed. By selecting the 'Edit Profile' button, the customer is directed to the interface shown in Figure 3.17, featuring fields to modify the customer's personal information. Changes are submitted with the 'Confirm' button, and the system displays a confirmation message. If customers choose not to update their personal information, clicking the 'Back' button redirects them to the Account Settings screen in Figure 3.15 without making any updates.

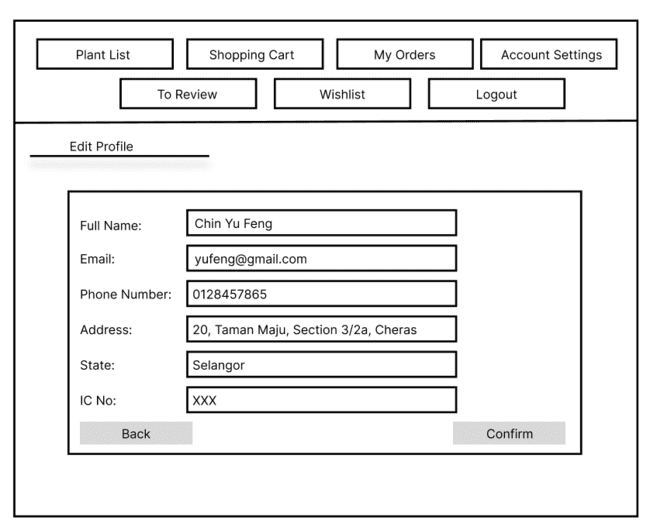
****

Figure 3.17: Edit Profile Details Screen

**3.5.4 Customer Views Added Plants In The Shopping Cart Screen**

Figure 3.18 shows 'Views Added Plants In The Shopping Cart Screen', which displays the shopping cart interface of the TLET Nursery Plant Shopping System. This is where the customers can view plants they have added to their shopping cart. The process is initiated from the "Customer Home Screen" as shown in Figure 3.14, where the customer selects the 'Shopping Cart' button. This action takes them to the "Views Added Plants In The Shopping Cart Screen" displayed in Figure 3.18, where the cart lists the selected plants with details like plant name, ID, image, and total price. Quantities can be adjusted via increase and decrease buttons. Customers can choose to either remove items or proceed to checkout.

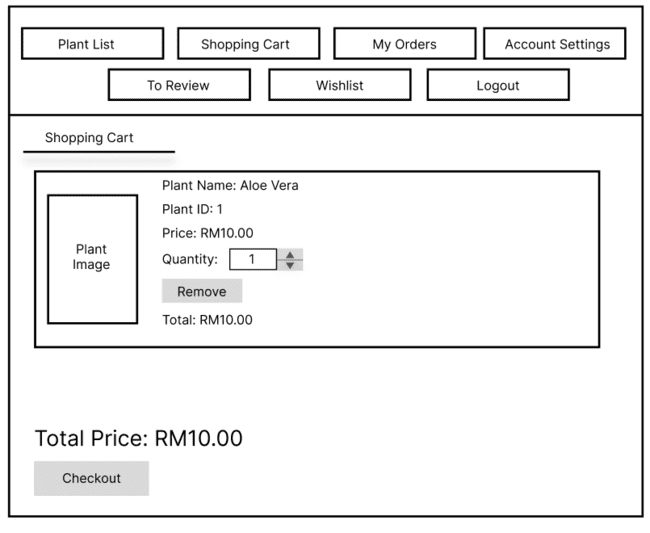
****

Figure 3.18: Views Added Plants In The Shopping Cart Screen

**3.5.5 Customer Adds Plants To The Shopping Cart Screen**

Figure 3.19 shows 'Adds Plants To The Shopping Cart Screen', which presents how customers can select and add plants to their shopping cart in the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 3.14, where the customer selects the 'Plant List' button. This action takes them to the "Adds Plants To The Shopping Cart Screen" displayed in Figure 3.19 where plants are displayed with images, ID, descriptions, prices,availability and reviews. Customers can then add their chosen plants to the shopping cart using the 'Add to Cart' button, with the system updating the cart accordingly. If a customer decides not to add the selected plants to the cart, they will remain on the plant list page as shown in Figure 3.19 to continue browsing.

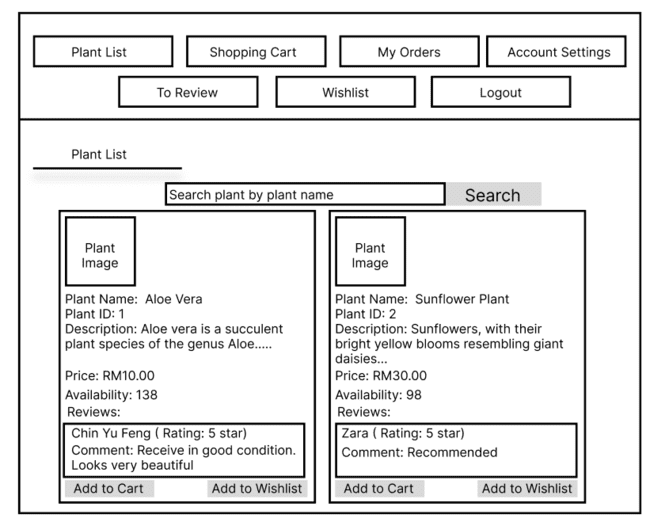
****

Figure 3.19: Adds Plants To The Shopping Cart Screen

**3.5.6 Customer Deletes Plants From The Shopping Cart Screen**

Figure 3.20 shows 'Deletes Plants From The Shopping Cart Screen', which displays the shopping cart interface of the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 3.14, where the customer selects the 'Shopping Cart' button. This action takes them to the "Deletes Plants From The Shopping Cart Screen" displayed in Figure 3.20 (same as Figure 3.18), where the cart lists the selected plants with details like plant name, ID, image, and total price. Customers can remove any unwanted plants by clicking the 'Remove' button. The system then updates to reflect the removal of these plants. Customers who do not wish to delete items can continue shopping or proceed to checkout from this screen in Figure 3.20.

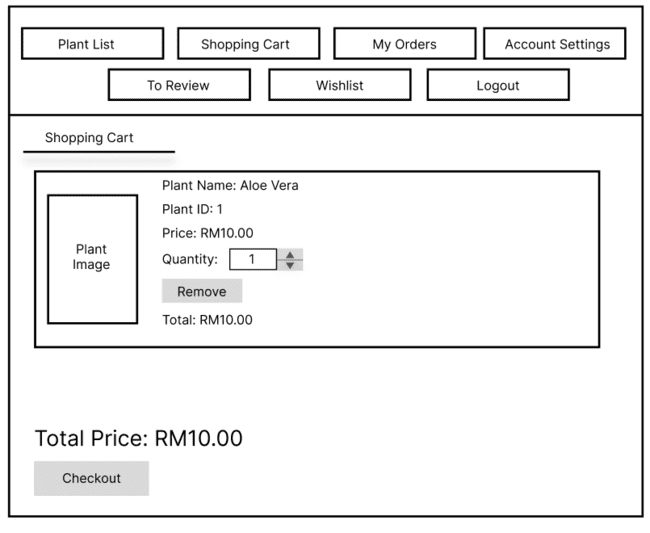
****

Figure 3.20: Deletes Plants From The Shopping Cart Screen

**3.5.7 Customer Searches a Plant by Plant Name Screen**

Figure 3.21 displays 'Searches a Plant by Plant Name Screen', which illustrates the search feature within the TLET Nursery Plant Shopping System that customers can use to find specific plants. The process is initiated from the "Customer Home Screen" as shown in Figure 3.14, where the customer selects the 'Plant List' button. This action takes them to the "Searches a Plant by Plant Name Screen" displayed in Figure 3.21 where plants are displayed with images, ID, descriptions, prices, availability,and reviews. Customers enter a plant name in the search bar and upon clicking the 'Search' button, the system displays the relevant plant listings that fit the search term.

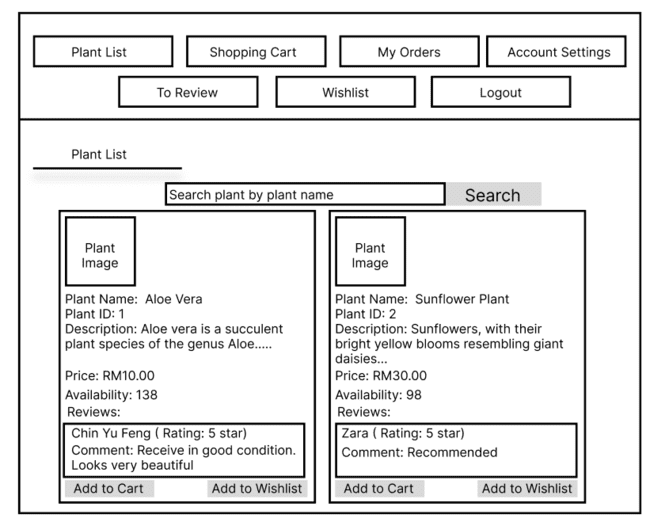
****

Figure 3.21: Searches a Plant by Plant Name Screen

**3.5.8 Customer Makes a Payment Screen**

Figure 3.24 shows the 'Make Payment Screen', in which customers are presented with an interface for completing their plant purchases via various payment methods. This process begins on the 'Customer Home Screen' depicted in Figure 3.14, where customers can select the 'Shopping Cart' to review their selections. This leads them to the 'Views Added Plants In The Shopping Cart Screen' shown in Figure 3.18, where they can choose to proceed by hitting the 'Checkout' button. This action navigates them to the 'Checkout Screen' in Figure 3.22, offering delivery and pickup options. If pickup is the chosen method, the system instructs to collect the order within 14 days at the TLET Company,Jalan Raja,Subang Jaya,Selangor before directing them to the 'Make Payment' interface in Figure 3.24. If the delivery method is selected, the system takes customers to the 'Delivery Details Screen' in Figure 3.23 which consolidates their order details including full name, delivery address, and costs. Clicking the 'Proceed to Payment' button leads them to the 'Make Payment Screen' in Figure 3.24.The system outlines payment options and costs, including any shipping fees. Customers can use the 'Credit/Debit' option to finalize the transaction. Upon confirming the payment, the system processes the order, confirming the successful transaction. For alternative payment methods, The system supports various methods, including FPX (Internet Banking), and the Touch 'N Go E-Wallet, ensuring a seamless checkout process and providing the customer with a confirmation of their successful order. After payment, customers are directed back to the 'Plant List' screen, as indicated in Figure 3.19, to continue shopping or review their orders.

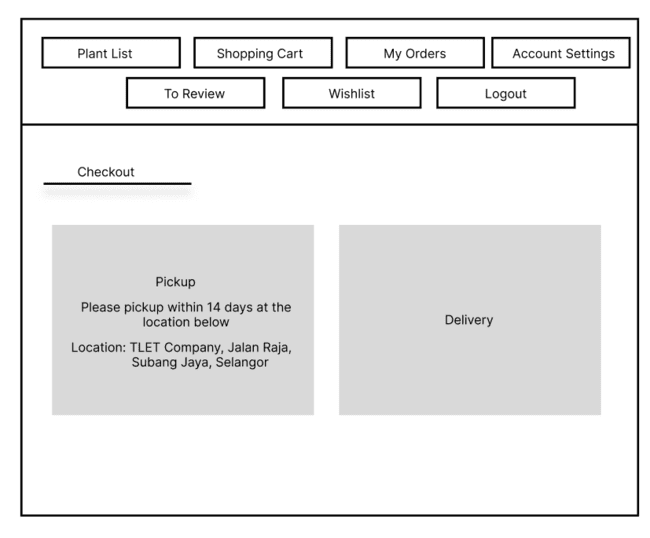
****

Figure 3.22: Checkout Screen

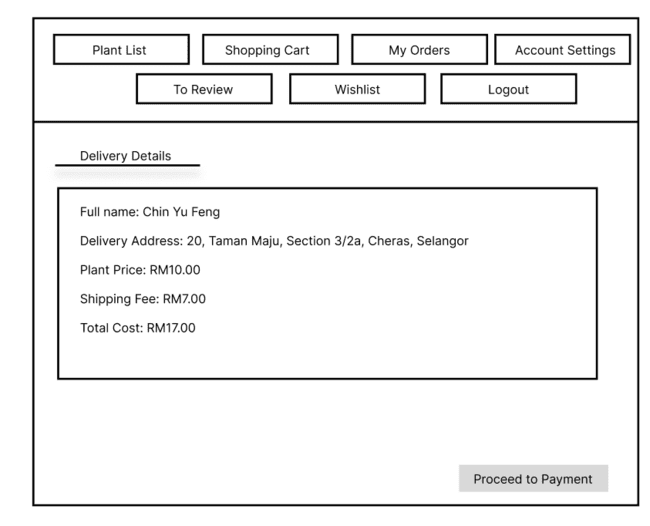
****

Figure 3.23: Delivery Details Screen

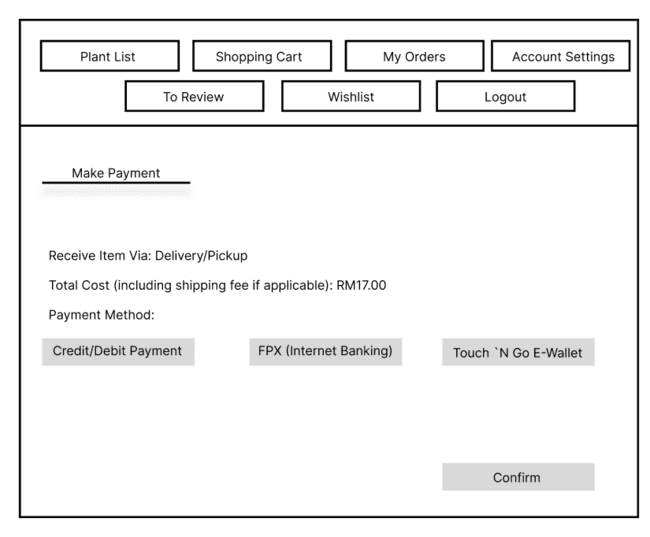
****

Figure 3.24: Make Payment Screen

**3.5.9 Customer Views Order Details of Ongoing Orders Screen**

Figure 3.25 displays the "Views Order Details of Ongoing Orders Screen" in the TLET Nursery Plant Shopping System, a user interface that allows customers to monitor the status of their current orders. This feature is accessed from the "Customer Home Screen", which is referenced in Figure 3.14. By clicking the 'My Orders' button, customers are led to the screen shown in Figure 3.25, where they are presented with a detailed listing of their ongoing orders. For each order, the system provides comprehensive information, including the order ID, names of the plants ordered, the quantity of each plant, price per plant, the total price for the order, the order date, and the current status of the order.

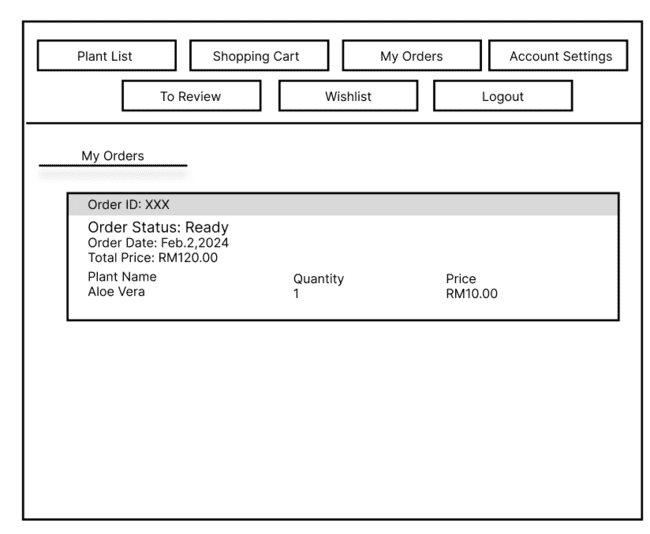
****

Figure 3.25: Views Order Details of Ongoing Orders Screen

**3.5.10 Customer Leaves Reviews and Ratings Screen**

Figure 3.26 shows the "Leaves Reviews and Ratings Screen" of the TLET Nursery Plant Shopping System, where customers can leave feedback on their purchased plants. This feature is accessed from the "Customer Home Screen", which is referenced in Figure 3.14. By clicking the 'To Review' button, customers are led to the screen shown in Figure 3.26, which is presented with a list of the plants they have purchased, each accompanied by an image, a rating system, and a space for written feedback. Customers can assign a rating from 1 to 5 stars and provide comments on their experience with the plant. After filling in their feedback, customers can submit their reviews by clicking the 'Submit' button.

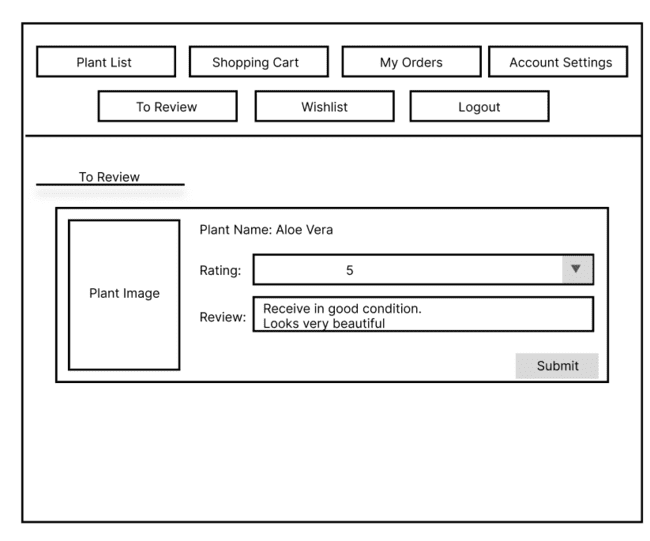
****

Figure 3.26: Leaves Reviews and Ratings Screen

**3.5.11 Customer Manages a Wishlist Screen**

Figure 3.27 presents the "Manages a Wishlist Screen" within the TLET Nursery Plant Shopping System where customers save plants they are considering for future purchases. This feature is accessed from the "Customer Home Screen", which is referenced in Figure 3.14. By clicking the 'Wishlist' button, customers are led to the screen shown in Figure 3.27. It displays each plant that the customer has added to their Wishlist, including a plant image, the plant name, and its price. For convenience, customers can update their Wishlist by removing items they no longer wish to consider; this is done by clicking the "Delete" button next to the respective plant.

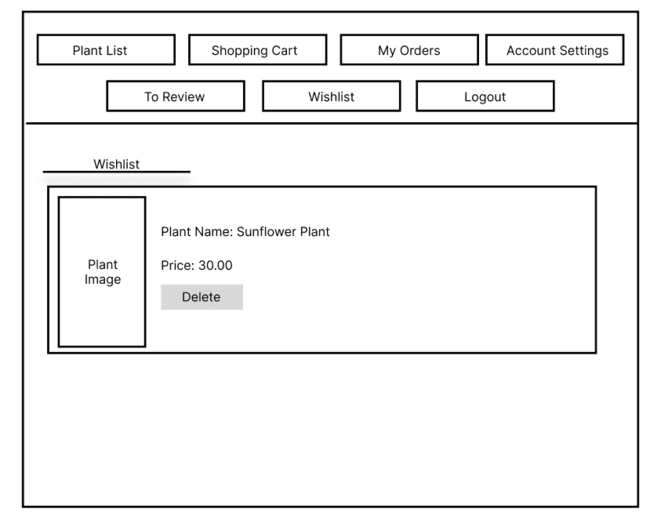
****

Figure 3.27: Manages a Wishlist Screen

## Subsystem Components

Table 3.30 outlines various use cases for a customer in a plant shopping system. The customer has the ability to change their password and edit profile details, ensuring their account remains secure and up-to-date. Shopping functionalities include viewing, adding, and deleting plants from the shopping cart, which are categorized as sub-use cases. Additionally, customers can search for specific plants by name. Post-selection activities involve making payments and viewing details of ongoing orders, again noted as sub-use cases. Lastly, the system allows customers to leave reviews and ratings for plants, enabling feedback and influencing future customer choices.

Table 3.30: Actor and Respective Use Cases

| Actor | Use Cases |
| --- | --- |
| Customer | A customer changes their password |
| A customer can edit profile details. |
| A customer shops plants.   * View added plants in the shopping cart. (sub-use case) * Add plants to the shopping cart. (sub-use case) * Delete plants from the shopping cart. (sub-use case) |
| A customer searches a plant by plant name. |
| A customer completes purchases and manages orders.   * Make payment. (sub-use case) * View order details of ongoing orders. (sub-use case) |
| A customer leaves reviews and ratings. |

### 3.6.1 A customer changes their password.

Figure 3.28 shows the "Change Password Activity Diagram" for the TLET Nursery Plant Shopping System, outlining the steps a customer follows to update their account password. Starting from the "Customer Home Screen," the customer navigates to the 'Account Settings' where they can access the 'Change Password' option. Upon selection, they're prompted to enter their current password and key in a new one. The system checks if the current password is correct and the new one meets the security criteria. If all conditions are met, the password is updated, and a confirmation message is displayed. The process concludes by redirecting them back to their account settings. If the current password is incorrect, the system displays an error message, and the customer is prompted to re-enter the current password. If the customer’s entered new password does not meet the security requirements, the system displays an error message and a customer has to enter a new password.

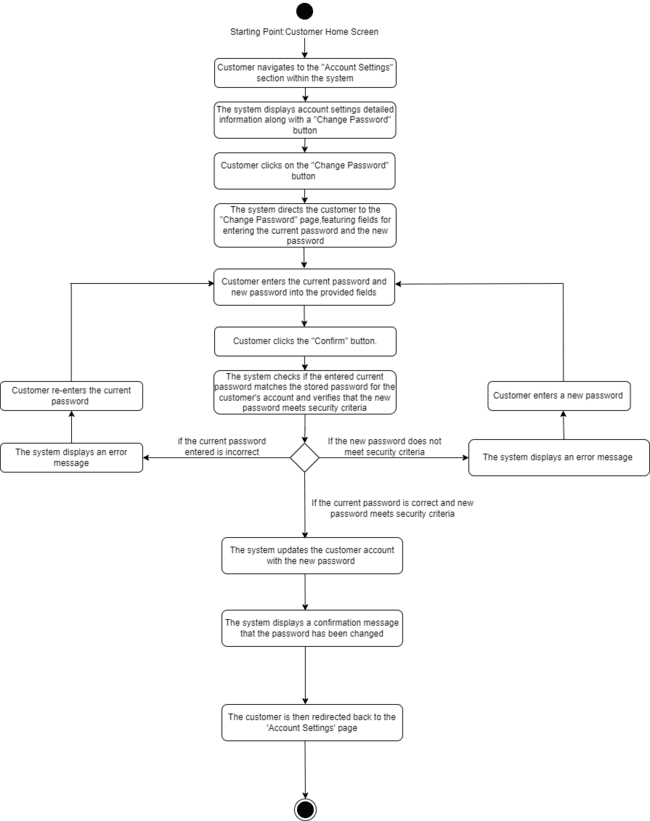


Figure 3.28 Change Password Activity Diagram

### 3.6.2 A customer can edit profile details.

The activity diagram depicted in Figure 3.29 outlines the sequence of actions a customer takes to update their profile information within the system. The process commences at the "Customer Home Screen," where the customer moves to the "Account Settings" section. Here, the system presents detailed account information and an "Edit Profile" button. Once the customer clicks this button, they're directed to a page where they can modify their personal details. The customer can either decide to cancel the update and go back or proceed to alter the necessary information in the provided fields. After submitting the updates, the system conducts a validation process. If any discrepancies, such as an invalid email format, are detected, the system prompts the customer to make corrections. Once the information is validated and saved, the system acknowledges the successful update with a confirmation message, completing the profile editing workflow.

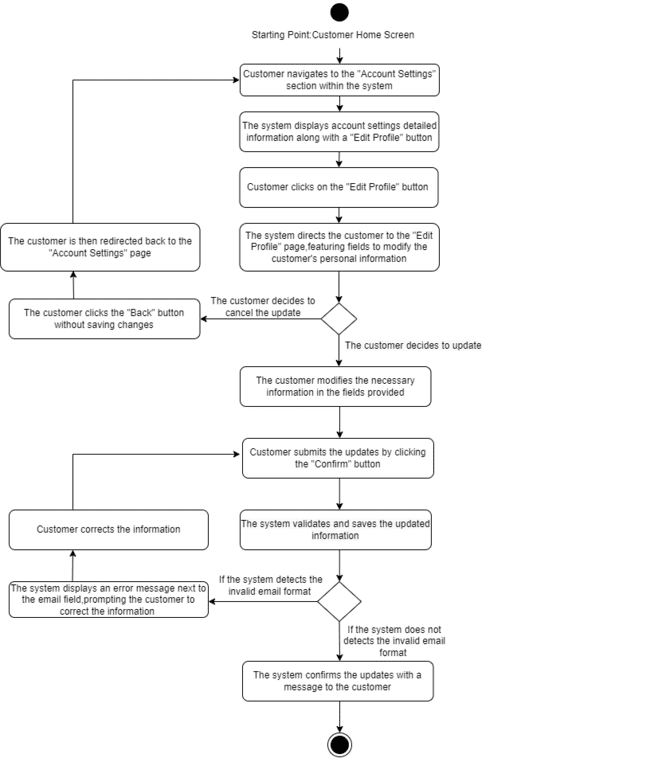


Figure 3.29 Edit Profile Details Activity Diagram

### 

### 

### 3.6.3 A customer shops plants.

Figure 3.30 shows a use case diagram titled "Sub-use cases for Shop Plant Use Case". Figure 3.30 breaks down the main activity of shopping for plants into three distinct actions a user can perform such as viewing added plants in the shopping cart, adding plants to the shopping cart, and deleting plants from the shopping cart. Each sub-use case represents a different interaction that customers can have with the shopping cart feature of a plant nursery shopping system.

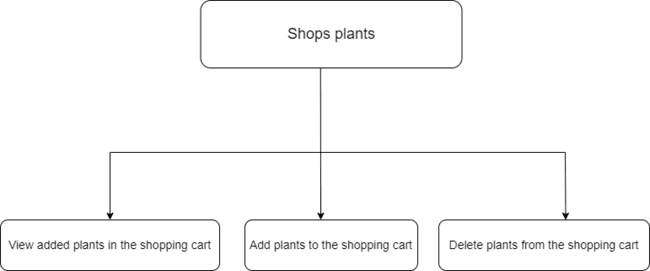


Figure 3.30: Sub-use cases for Shop Plant Use Case

### 3.6.3.1 A customer views added plants in the shopping cart

Figure 3.31 is an activity diagram titled "View Added Plant in the Cart Activity Diagram," illustrating the flow of actions a customer takes to view plants added to the shopping cart within a shopping system. Starting from the "Customer Home Screen," the customer navigates to the "Shopping Cart" section. If the cart is empty, the system displays a message stating the shopping cart is empty. Conversely, if the cart contains items, the system shows a list of added plants, complete with details such as the plant ID, image, name, total price, and quantity.

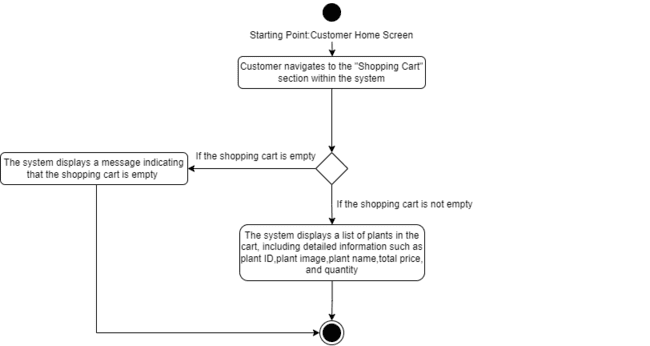


Figure 3.31: View Added Plant in the Cart Activity Diagram

### 3.6.3.2 A customer adds plants to the shopping cart

Figure 3.32 shows "Add Plant to the Shopping Cart Activity Diagram" and illustrates the customer's process for adding items to the shopping cart. The activity begins on the "Customer Home Screen," where the customer navigates to the "Plant List" section. Here, the system displays a list of plants with detailed information including images, reviews, ratings, descriptions, prices, and availability. Customers select plants they are interested in and have the option to add them to their cart. If they change their mind, they can choose not to proceed. However, if they decide to add the plants, they click the "Add to Cart" button. The system then processes the request and confirms that the plants have been successfully added to the cart.

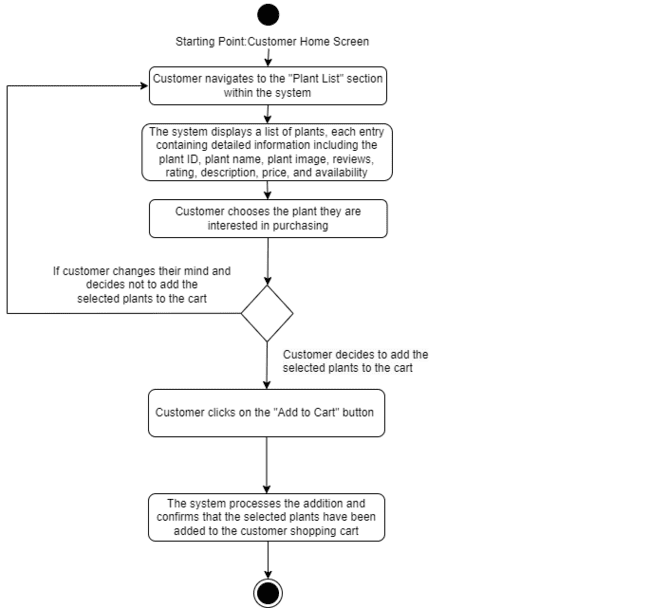


Figure 3.32: Add Plant to the Shopping Cart Activity Diagram

### 3.6.3.3 A customer deletes plants from the shopping cart

Figure 3.33, titled "Delete Plant from the Shopping Cart Activity Diagram," illustrates the process for a customer to remove items from their shopping cart. The sequence begins at the "Customer Home Screen," where the customer proceeds to the "Shopping Cart" section. The system then presents a list of all the plants currently in the cart, including details like plant ID, image, name, total price, and quantity. The customer can identify and select the plants they wish to remove. Upon deciding to remove certain plants, the customer clicks the "Remove" button associated with those items. The system then processes the removal and confirms that the selected plants have been successfully deleted from the shopping cart.

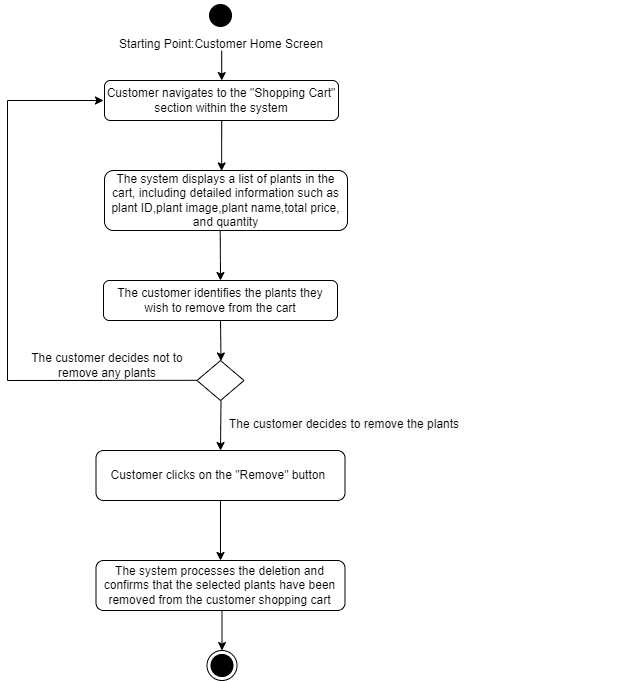


Figure 3.33: Delete Plant from the Shopping Cart Activity Diagram

### 

### 3.6.4 A customer searches a plant by plant name

Figure 3.34 is titled "Search a Plant by Plant Name Activity Diagram" and it details the customer's process of searching for a specific plant within the system. The activity starts on the "Customer Home Screen," where the customer moves to the "Plant List" section. The system then presents a searchable list of plants and a text input field.The customer can enter the plant's name they wish to find and initiate the search by clicking the "Search" button. If the entered name does not match any plants, the system indicates no results and prompts the customer to refine their search. Conversely, if matches are found, the system displays detailed information for each plant that fits the search criteria.

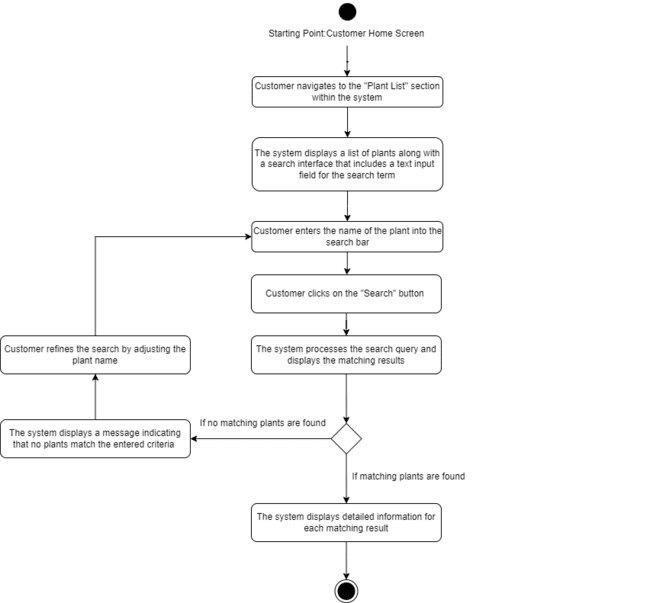


Figure 3.34:Search a Plant by Plant Name Activity Diagram

### 

### 3.6.5 A customer completes purchases and manages orders

Figure 3.35 presents a use case diagram with the main activity "Completes purchases and manages orders". This activity is broken down into two sub-use cases: "Make payment" and "View order details of ongoing orders". The diagram shows that the main use case involves not only the finalization of purchases through payment but also the management of orders by viewing their details as they are being processed.

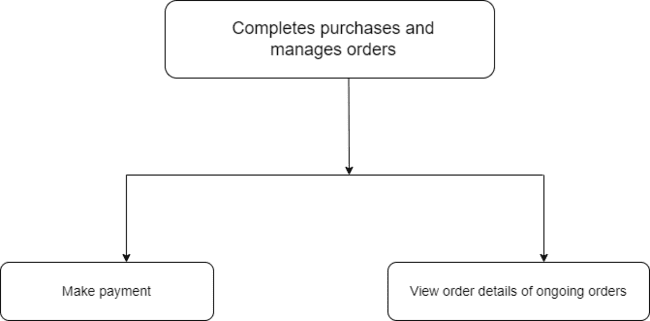


Figure 3.35:Sub-use cases for Complete Purchases and Manage Orders Use Case

#### 3.6.5.1 A customer makes payment

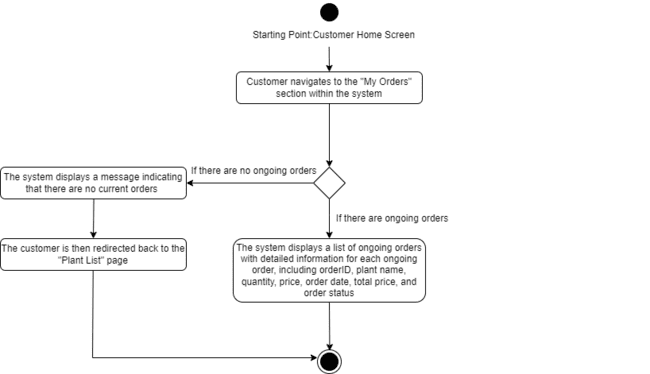
#### Figure 3.36 represents the sequence of steps a customer takes to complete a purchase in the TLET Nursery Plant Shopping System. The activity begins from the Customer Home Screen where the logged-in customer accesses their Shopping Cart, filled with their plant selections. If the shopping cart is empty, a message stating it is empty will be displayed. After reviewing the items and their details, the customer proceeds to checkout, opting for either delivery or pickup. When choosing delivery, they will be directed to the “Delivery Details” page to confirm their personal information, delivery details, and order details. Then, they will move to the payment section after clicking on the “Proceed to payment” button, where they select a payment method—credit/debit card, internet banking, or e-wallet. Upon confirming the payment method and transaction, the system validates and processes the transaction, subsequently confirming the successful order creation. The diagram concludes with the customer being redirected back to the Plant List page, symbolizing the completion of the purchase. If the customer chooses to pick up, the system will inform the customer to pick up the order within 14 days from the specified location. Then, the customer will be directed to the “Make Payment page” where they choose their payment method. The same process as the delivery option will then be followed.

#### 

#### Figure 3.36: Make Payment Activity Diagram

#### 3.6.5.2 A customer view order details of ongoing orders

Figure 3.37 labeled as "View Order Details of Ongoing Orders Activity Diagram" outlines the process for a customer to view the status of their current orders within a shopping system. The starting point is the "Customer Home Screen," from which the customer navigates to the "My Orders" section. Here, the system checks for ongoing orders. If none are present, a message informs the customer that there are no current orders, and the customer is redirected back to the "Plant List" page. If ongoing orders exist, the system displays a detailed list including order ID, plant name, quantity, price, order date, total price, and order status, allowing the customer to track the progress of their purchases.

Figure 3.37: View Order Details of Ongoing Orders Activity Diagram

#### 3.6.6 A customer leaves reviews and ratings

#### Figure 3.38 shows the "Leaves Reviews and Ratings Activity Diagram," depicting the steps a customer follows to leave feedback to the plant. Starting from the Customer Home Screen, the customer proceeds to the "To Review'' section. The system then presents a list of items eligible for review, each accompanied by details such as plant name, current rating, customer's review, and an image of the plant. Customers fill out the review form with comments and assign a rating from 1 to 5 stars. If a customer attempts to submit feedback without assigning a rating, the system prompts them to provide one. Once the review is complete and a rating is selected, the customer submits their feedback by clicking the "Submit" button, finalizing the review process.

#### 

#### Figure 3.38:Leaves Reviews and Ratings Activity Diagram

#### 3.6.7 A customer manage wishlist

Figure 3.39 shows the “Manage Wishlist Activity Diagram" illustrating the process a customer follows to manage their wishlist in the system. Beginning at the Customer Home Screen, the customer moves to the "Wishlist" section. If the wishlist is empty, the system notifies the customer accordingly. Otherwise, the system displays the wishlist items, complete with plant names, images, and prices. The customer can then decide to remove any plant from the wishlist by clicking the "Delete" button associated with the plant, effectively managing their wishlist preferences.

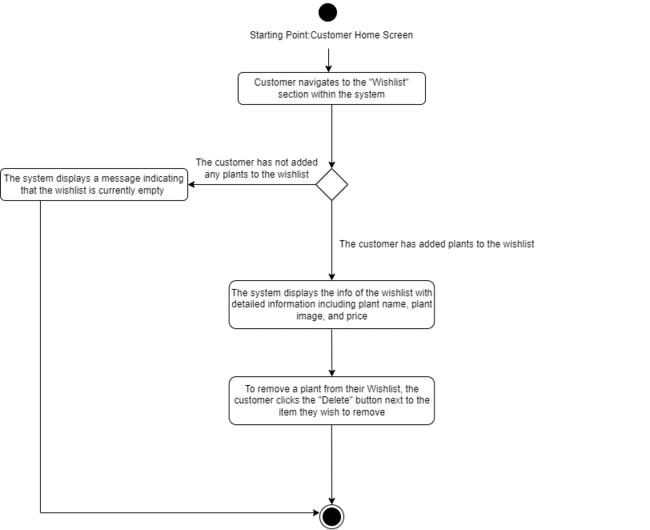


Figure 3.39:Manage Wishlist Activity Diagram

# **Implementation**

## Development Environment

Figure 4.1 displays the project explorer of a Django application tailored for the customer side of a Nursery Plant Shopping System. The directory structure reveals the project folder labeled 'TruePItSys', housing the Django application 'app' and encompassing subdirectories like 'migrations' for database evolution, 'static' for storing static files such as CSS and JavaScript, and 'templates' for HTML files. Within 'templates', there's a hierarchy of HTML files corresponding to various customer interfaces, such as account settings, shopping cart, order details, and payment processes, indicating a comprehensive e-commerce platform. This structure is typical of a Django project, where 'models.py' defines data models, 'views.py' contains business logic to handle HTTP requests, and 'urls.py' maps URLs to their respective views.

|  |  |  |
| --- | --- | --- |

Figure 4.1: Nursery Plant Shopping System (Customer’s) Project Explorer

## Main Program Codes

For the customer’s interaction with the TLET Nursery Plant Shopping System, key functionalities are encapsulated within the `views.py` file found in the Django application structure. This essential script contains the necessary request handling to enable user-specific operations. It supports a variety of customer actions such as changing password details (`change\_password.html`), editing profile information (`edit\_profile.html`), and navigating the shopping cart to add (`cart.html`) or remove plants. Additionally, it facilitates the search for specific plants (`plant\_list.html`), processing payments (`make\_payment.html`), and viewing the details of ongoing orders (`my\_orders.html`). Post-purchase, customers can leave reviews and ratings (`to\_review.html`) and manage their wishlist (`wishlist.html`). Each of these customer-centered actions is coordinated through the corresponding HTML templates and the logic in `views.py`, ensuring a seamless e-commerce experience for plant selection, purchase, and post-purchase activities.

**4.2.1 Customer dashboard**

Figure 4.3 and 4.2 of the TruePItSys Django project reveal the customer-facing aspects of the Nursery Plant Shopping System, where Figure 4.3 serves as a feature-rich presentation layer allowing users to interact with their dashboard, and Figure 4.2 operates as the logic hub, managing user authentication, plant data retrieval, review aggregation, and search functionalities. Together, they constitute a cohesive module facilitating customer interactions from the visual interface to the underlying data processes.

|  |
| --- |

Figure 4.2:customer\_dashboard/view.py/app

|  |
| --- |

Figure 4.3: customer\_dashboard.html/templates/app

**4.2.2 A customer changes their password**

Figure 4.4 and 4.5 present the backend view functions of the Nursery Plant Shopping System's Django application, dedicated to customer account management. Specifically, Figure 4.4 allows customers to view and manage their profile details, while Figure 4.5 provides the functionality for customers to update their passwords securely, both requiring user authentication. Paired with the Figure 4.6 template, which features a clean Bootstrap interface for password updates, these views form a secure and user-friendly system for customers to maintain their account integrity and personal information within the shopping platform.

| Figure 4.4: account\_settings/view.py/app    Figure 4.5: change\_password/view.py/app |
| --- |

|  |
| --- |

Figure 4.6: change\_password.html/templates/app

**4.2.3 A customer can edit profile details.**

In the Nursery Plant Shopping System, the customer's interaction with their profile is managed through a series of views and templates, as shown in Figures 4.4, 4.5, and 4.8. The `account\_settings` function retrieves the logged-in customer's profile details and displays them on the `account\_settings.html` page, while the `update\_profile` function processes profile edits submitted via the `edit\_profile.html` form, performing validations on fields such as full name, email, phone number, address, state, and IC number. These are rendered in a user-friendly interface with Bootstrap styling, ensuring customers can easily manage their personal information and account settings within the system.

|  |
| --- |

Figure 4.7:update\_profile/view.py/app

|  |
| --- |

Figure 4.8: edit\_profile.html/templates/app

**4.2.4 Customer Views Added Plants In The Shopping Cart**

In the Nursery Plant Shopping System, the customer's shopping cart experience is streamlined through several backend functions and a front-end display, as depicted in Figures 4.9 to 4.13. The `add\_to\_cart` view adds selected plants to the customer's shopping cart, ensuring stock availability, while the `view\_cart` view presents the cart with the items and total price. Should customers wish to remove items, the `remove\_from\_cart` function facilitates this, and the `update\_cart\_item` allows for adjustment of plant quantities. The cart's frontend, shown in `cart.html`, provides customers with an interactive list of added plants, prices, and quantities, complete with options to update quantities inline, remove items, or proceed to checkout, reinforcing a seamless shopping experience.

| Figure 4.9: add\_to\_cart/view.py/app    Figure 4.10: view\_cart/view.py/app    Figure 4.11:remove\_from\_cart/view.py/app    Figure 4.12:update\_cart\_item/view.py/app |
| --- |

|  |
| --- |

Figure 4.13: cart.html/templates/app

**4.2.5 Customer Adds Plants To The Shopping Cart**

The customer's interaction with the shopping cart within the Nursery Plant Shopping System is encapsulated in Figures 4.9 to 4.13, where the `add\_to\_cart` function integrates seamlessly into the system's backend, allowing customers to select and add plants to their shopping cart with immediate feedback on stock levels and confirmation messages. The `view\_cart` function then displays these selections in a structured list, showing detailed information and images of the plants chosen, prices, and an option to modify the quantities or remove items directly from the cart, culminating in a user-friendly shopping experience as represented in the `cart.html` template.

**4.2.6 Customer Deletes Plants From The Shopping Cart**

In the Nursery Plant Shopping System, the feature for a customer to delete plants from the shopping cart is depicted in Figures 4.9 to 4.13, specifically through the `remove\_from\_cart` function in the `view.py` module. This backend function, identified as customer deletes plants from the shopping cart, is triggered when a customer selects the delete option next to a plant in their shopping cart, as illustrated in the `cart.html` template. Upon activation, the function locates the plant item within the database through its unique ID, removes it from the customer's shopping cart, and updates the cart's status, providing real-time feedback and maintaining the accuracy of the cart's contents and the total price.

**4.2.7 Customer Searches a Plant by Plant Name**

In the Nursery Plant Shopping System, the `plant\_list\_view` function within `view.py` in Figure 4.14 and the `plant\_list.html` template in Figure 4.15 collaborate to enable customers to search for plants by name. The view function processes the search query submitted through the form in the template, filters the plant objects based on the name entered, and attaches associated reviews to each plant. If a match is found, the results are dynamically updated, displaying the relevant plant details along with user reviews on the template page, which is designed with a clean, user-friendly interface featuring a search bar for input and plant cards for results. If no matches are found, the system informs the user with a message indicating the absence of matching plants, ensuring a responsive and informative search experience.

|  |
| --- |

Figure 4.14:plant\_list\_view/view.py/app

|  |
| --- |

Figure 4.15: plant\_list.html/templates/app

**4.2.8 Customer Makes a Payment**

The customer journey for making payments within the system begins with the checkout process Figure 4.16, where the cart's contents and total price are confirmed. The customer then proceeds to detail their delivery preferences and address, with shipping fees calculated based on their location Figure 4.17. The pivotal step of making the payment is facilitated by the `make\_payment` function Figure 4.18, which presents the total cost including any shipping fees and offers multiple payment methods like Credit/Debit Card, FPX, or E-Wallet. The checkout process is visually represented in a straightforward form, guiding the customer through the pickup or delivery selection Figure 4.19, and subsequently to the delivery details confirmation page Figure 4.20. Finally, the payment page Figure 4.21 concludes the process, ensuring all financial transactions are securely handled, and the customer is informed of the success of their payment, keeping the user experience smooth and intuitive throughout.

| Figure 4.16:checkout/view.py/app    Figure 4.17:delivery\_details/view.py/app      Figure 4.18:make\_payment/view.py/app |
| --- |

|  |
| --- |

Figure 4.19: checkout.html/templates/app

|  |
| --- |

Figure 4.20: delivery\_details.html/templates/app

|  |
| --- |

Figure 4.21:make\_payment.html/templates/app

**4.2.9 Customer Views Order Details of Ongoing Orders**

The feature allowing customers to view the details of their ongoing orders is depicted in Figure 4.22 and Figure 4.23. The back-end functionality, defined in the `my\_orders` view Figure 4.22, fetches order data for the logged-in customer, leveraging Django's ORM capabilities to efficiently prefetch related `order\_items` and `payment` information to minimize database hits. The front-end representation Figure 4.23 provides a user-friendly interface displaying each order's status, date, total price, and a detailed breakdown of purchased items in a tabular format within a responsive card layout. This design ensures customers have easy access to all relevant details of their orders, enhancing transparency and trust in the service provided by the application.

|  |
| --- |

Figure 4.22:my\_orders/view.py/app

|  |
| --- |

Figure 4.23: my\_orders.html/templates/app

**4.2.10 Customer Leaves Reviews and Ratings**

Figures 4.24 and 4.25, along with Figure 4.26, outline the customer review process. Customers are prompted to leave reviews for items marked as 'Completed' and not yet reviewed through the `to\_review` view, which collates these items into a list, ensuring each plant is reviewed only once. The `submit\_review` view then handles the posting of these reviews, saving the customer's feedback and updating the status of the item to reflect that it has been reviewed. The associated template Figure 4.26 provides an intuitive interface, where customers can see their past purchases that require reviews, rate them, and submit their comments via a form. This process not only enables customers to share their experience with the products but also aids in enhancing the service by providing valuable feedback to the system.

| Figure 4.24:to\_review/view.py/app    Figure 4.25:submit\_review/view.py/app |
| --- |

|  |
| --- |

Figure 4.26: to\_review.html/templates/app

**4.2.11 Customer Manages a Wishlist**

The customer's wishlist management in the web application is delineated across Figures 4.27 to 4.30. Customers can add desired plants to their wishlist using the `add\_to\_wishlist` function, which checks for a customer profile before appending items to the wishlist. The `view\_wishlist` function and the corresponding template Figure 4.30 display the current wishlist items, allowing customers to visualize their saved plants. Should the customer decide to remove an item, the `remove\_from\_wishlist` function facilitates this, ensuring the item is deleted from the user's wishlist. This suite of features provides an interactive and personalized experience for the customer, enabling them to curate a list of favored items for potential future purchase.

| Figure 4.27: add\_to\_wishlist/view.py/app    Figure 4.28: view\_wishlist/view.py/app    Figure 4.29: remove\_from\_wishlist/view.py/app |
| --- |

|  |
| --- |

Figure 4.30: wishlist.html/templates/app

## Sample Screens

**4.3.1 Customer Home Screen**

Figure 4.31 shows 'Customer Home Screen', which displays the main interface for registered users on the TLET Nursery Plant Shopping System. This dashboard provides customers with a variety of options including adding items to their shopping cart or wishlist, accessing their orders, reviewing past purchases, managing account settings, and logging out. The central part of the screen highlights a plant's detailed information such as ID, name, image, description, price, and plant availability. Customers can also read existing reviews and ratings for the plants listed.

| Figure 4.31: Customer Home Screen |
| --- |

**4.3.2 Customer Changes Their Password Screen**

Figure 4.33 shows the "Change Password Screen", which presents the interface for customers to update their password on the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 4.31, where the customer selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 4.32, where the customer's details are listed. By selecting the 'Change Password' button, the customer is directed to the interface shown in Figure 5.33, where they are required to enter their old password followed by the new password they wish to set. After submitting the new password by clicking 'Confirm', the system verifies the new password. If the update is successful, a confirmation message is displayed, and the customer is taken back to the "Account Settings Screen" shown in Figure 4.32, completing the password change process.

| Figure 4.32: Account Settings Screen    Figure 4.33: Change Password Screen |
| --- |

**4.3.3 Customer Can Edit Profile Details Screen**

Figure 4.34 shows 'Edit Profile Details Screen', where customers can update their personal information on the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 4.31, where the customer selects the 'Account Settings' button. This action takes them to the "Account Settings Screen'' displayed in Figure 4.32, where the customer's details are listed. By selecting the 'Edit Profile' button, the customer is directed to the interface shown in Figure 4.34, featuring fields to modify the customer's personal information. Changes are submitted with the 'Confirm' button, and the system displays a confirmation message. If customers choose not to update their personal information, clicking the 'Back' button redirects them to the Account Settings screen in Figure 4.32 without making any updates.

| Figure 4.34: Edit Profile Details Screen |
| --- |

**4.3.4 Customer Views Added Plants In The Shopping Cart Screen**

Figure 4.35 shows 'Views Added Plants In The Shopping Cart Screen', which displays the shopping cart interface of the TLET Nursery Plant Shopping System. This is where the customers can view plants they have added to their shopping cart. The process is initiated from the "Customer Home Screen" as shown in Figure 4.31, where the customer selects the 'Shopping Cart' button. This action takes them to the "Views Added Plants In The Shopping Cart Screen" displayed in Figure 4.35, where the cart lists the selected plants with details like plant name, ID, image, and total price. Quantities can be adjusted via increase and decrease buttons.

| Figure 4.35:Views Added Plants In The Shopping Cart Screen |
| --- |

**4.3.5 Customer Adds Plants To The Shopping Cart Screen**

Figure 4.36 shows 'Adds Plants To The Shopping Cart Screen', which presents how customers can select and add plants to their shopping cart in the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 4.31, where the customer selects the 'Plant List' button. This action takes them to the "Adds Plants To The Shopping Cart Screen" displayed in Figure 4.36 where plants are displayed with images, ID, descriptions, prices, and availability. Customers can then add their chosen plants to the shopping cart using the 'Add to Cart' button, with the system updating the cart accordingly. If a customer decides not to add the selected plants to the cart, they will remain on the plant list page as shown in Figure 4.36 to continue browsing.

| Figure 4.36: Adds Plants To The Shopping Cart Screen |
| --- |

**4.3.6 Customer Deletes Plants From The Shopping Cart Screen**

Figure 4.37 shows 'Deletes Plants From The Shopping Cart Screen', which displays the shopping cart interface of the TLET Nursery Plant Shopping System. The process is initiated from the "Customer Home Screen" as shown in Figure 4.31, where the customer selects the 'Shopping Cart' button. This action takes them to the "Deletes Plants From The Shopping Cart Screen" displayed in Figure 4.37 (same as Figure 4.35), where the cart lists the selected plants with details like plant name, ID, image, and total price. Customers can remove any unwanted plants by clicking the 'Remove' button. The system then updates to reflect the removal of these plants. Customers who do not wish to delete items can continue shopping or proceed to checkout from this screen in Figure 4.37.

| Figure 4.37: Deletes Plants From The Shopping Cart Screen |
| --- |

**4.3.7 Customer Searches a Plant by Plant Name Screen**

Figure 4.38 displays 'Searches a Plant by Plant Name Screen', which illustrates the search feature within the TLET Nursery Plant Shopping System that customers can use to find specific plants. The process is initiated from the "Customer Home Screen" as shown in Figure 4.31, where the customer selects the 'Plant List' button. This action takes them to the "Searches a Plant by Plant Name Screen" displayed in Figure 4.38 where plants are displayed with images, ID, descriptions, prices, and availability. Customers enter a plant name in the search bar and upon clicking the 'Search' button, the system displays the relevant plant listings that fit the search term.

| Figure 4.38: Searches a Plant by Plant Name Screen |
| --- |

**4.3.8 Customer Makes a Payment Screen**

Figure 4.39 shows the 'Make Payment Screen', in which customers are presented with an interface for completing their plant purchases via various payment methods. This process begins on the 'Customer Home Screen' depicted in Figure 4.31, where customers can select the 'Shopping Cart' to review their selections. This leads them to the 'Views Added Plants In The Shopping Cart Screen' shown in Figure 4.35, where they can choose to proceed by hitting the 'Checkout' button. This action navigates them to the 'Checkout Screen' in Figure 4.39, offering delivery and pickup options. If pickup is the chosen method, the system instructs to collect the order within 14 days at the TLET Company,Jalan Raja,Subang Jaya,Selangor before directing them to the 'Make Payment' interface in Figure 4.41. If the delivery method is selected, the system takes customers to the 'Delivery Details Screen' in Figure 4.40 which consolidates their order details including full name, delivery address, and costs. Clicking the 'Proceed to Payment' button leads them to the 'Make Payment Screen' in Figure 4.41. The system outlines payment options and costs, including any shipping fees. Customers can use the 'Credit/Debit' option to finalize the transaction. Upon confirming the payment, the system processes the order, confirming the successful transaction. For alternative payment methods, The system supports various methods, including FPX (Internet Banking), and the Touch 'N Go E-Wallet, ensuring a seamless checkout process and providing the customer with a confirmation of their successful order. After payment, customers are directed back to the 'Plant List' screen, as indicated in Figure 4.3.6, to continue shopping or review their orders.

| Figure 4.39: Checkout Screen    Figure 4.40: Delivery Details Screen    Figure 4.41: Make Payment Screen |
| --- |

**4.3.9 Customer Views Order Details of Ongoing Orders Screen**

Figure 4.42 displays the "Views Order Details of Ongoing Orders Screen" in the TLET Nursery Plant Shopping System, a user interface that allows customers to monitor the status of their current orders. This feature is accessed from the "Customer Home Screen", which is referenced in Figure 4.31. By clicking the 'My Orders' button, customers are led to the screen shown in Figure 4.42, where they are presented with a detailed listing of their ongoing orders. For each order, the system provides comprehensive information, including the order ID, names of the plants ordered, the quantity of each plant, price per plant, the total price for the order, the order date, and the current status of the order.

| Figure 4.42: Views Order Details of Ongoing Orders Screen |
| --- |

**4.3.10 Customer Leaves Reviews and Ratings Screen**

Figure 4.43 shows the "Leaves Reviews and Ratings Screen" of the TLET Nursery Plant Shopping System, where customers can leave feedback on their purchased plants. This feature is accessed from the "Customer Home Screen", which is referenced in Figure 4.31. By clicking the 'To Review' button, customers are led to the screen shown in Figure 4.43, which is presented with a list of the plants they have purchased, each accompanied by an image, a rating system, and a space for written feedback. Customers can assign a rating from 1 to 5 stars and provide comments on their experience with the plant. After filling in their feedback, customers can submit their reviews by clicking the 'Submit' button.

| Figure 4.43: Leaves Reviews and Ratings Screen |
| --- |

**4.3.11 Customer Manages a Wishlist Screen**

Figure 4.44 presents the "Manages a Wishlist Screen" within the TLET Nursery Plant Shopping System where customers save plants they are considering for future purchases. This feature is accessed from the "Customer Home Screen", which is referenced in Figure 4.31. By clicking the 'Wishlist' button, customers are led to the screen shown in Figure 4.44.It displays each plant that the customer has added to their Wishlist, including a plant image, the plant name, and its price. For convenience, customers can update their Wishlist by removing items they no longer wish to consider; this is done by clicking the "Delete" button next to the respective plant.

| Figure 4.44: Manages a Wishlist Screen |
| --- |

# **Testing**

## Test Data

Table 5.1 illustrates a series of test cases for the customer password update functionality in a system. Each row represents a unique scenario designed to evaluate the system's ability to enforce password security protocols upon update requests by a customer named Amber Chia. The password policy dictates that a new password must not be overly similar to the user's personal information, it should contain a minimum of 8 characters, it must not be commonly used or simple passwords, and it must include a combination of characters rather than being purely numeric. Highlighted entries in the table specify the particular test data being validated. The objective of these tests is to confirm that the system effectively recognizes and rejects passwords that do not comply with the established security requirements, thereby safeguarding user accounts against potential vulnerabilities.

Table 5.1: Customer Changes Their Password Test Data

| **Customer Name** | **Customer Email** | **Customer Old Password** | **Customer New Password** |
| --- | --- | --- | --- |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | chiaamber@gmail.com |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | chia33 |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | qwertyuiop |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | 123456 |
| Amber Chia | chiaamber@gmail.com | amber\*\*33 | soya\_milk |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | soya\_milk |

Table 5.2 illustrates a series of test cases for a customer profile editing process, with highlighted entries indicating errors due to non-compliance with data validation rules: a name that doesn't meet the required character length, an email lacking the '@' symbol , a phone number not prefixed with '60', a state abbreviation that is not recognized by the system, and an IC number formatted with more than the required 12 digits, ensuring the system accurately prompts users to correct their information.

Table 5.2: Customer Edits Profile Details Test Data

| **Customer Name** | **Customer Email** | **Customer Phone Number** | **Customer Address** | **Customer State** | **Customer IC** |
| --- | --- | --- | --- | --- | --- |
| Amber | chiaamber@gmail.com | 601123498766 | 35,Jalan BPU6 Taman Indah Puchong | Selangor | 040202106450 |
| AmberChia0202 | chiaambergmail.com | 601123498766 | 35,Jalan BPU6 Taman Indah Puchong | Selangor | 040202106450 |
| AmberChia0202 | chiaamber@gmail.com | 011234987665 | 35,Jalan BPU6 Taman Indah Puchong | Selangor | 040202106450 |
| AmberChia0202 | chiaamber@gmail.com | 601123498766 | Jalan Muzium, 88300 Kota Kinabalu, Sabah | Sa | 040202106450 |
| AmberChia0202 | chiaamber@gmail.com | 601123498765 | Jalan Muzium, 88300 Kota Kinabalu, Sabah | Sabah | 040202106450mm |
| AmberChia0202 | chiaamber@gmail.com | 601123498765 | Jalan Muzium, 88300 Kota Kinabalu, Sabah | Sabah | 040202106450 |

Table 5.3 which is a representation of a shopping cart interface from a test data perspective. It shows a list of plant items that a customer, identified by their email, has added to their shopping cart. Each entry in the list specifies the plant's name, a unique plant ID, the unit price, and the quantity selected. The table calculates the total for each item by multiplying the price by the quantity and also displays a cumulative total price.

Table 5.3: Customer View Added Plants In The Shopping Cart Test Data

| **Customer Email** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Total(price\*quantity)** | **Total Price** |
| --- | --- | --- | --- | --- | --- | --- |
| chiaamber@gmail.com | Sunflower Plant | 2 | RM30.00 | 1 | RM30.00 | RM150.00 |
|  | Adenium obesum | 6 | RM30.00 | 3 | RM90.00 | RM150.00 |
|  | Cactus | 3 | RM15.00 | 2 | RM30.00 | RM150.00 |

Table 5.4, titled "Customer Adds Plants To The Shopping Cart Test Data," lists plant products added to a shopping cart by a customer with entries for plant name, ID, price, and quantity, along with calculated individual totals and a grand total price. The entry for 'Ruellia Pink' is highlighted, potentially indicating a specific test case for adding items to the cart.

Table 5.4: Customer Add Plants To The Shopping Cart Test Data

| **Customer Email** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Total(price\*quantity)** | **Total Price** |
| --- | --- | --- | --- | --- | --- | --- |
| chiaamber@gmail.com | Sunflower Plant | 2 | RM30.00 | 1 | RM30.00 | RM160.00 |
|  | Adenium obesum | 6 | RM30.00 | 3 | RM90.00 | RM160.00 |
|  | Cactus | 3 | RM15.00 | 2 | RM30.00 | RM160.00 |
|  | Ruellia Pink | 8 | RM10.00 | 1 | RM10.00 | RM160.00 |

Table 5.5, titled "Customer Deletes Plants From The Shopping Cart Test Data," shows a similar setup but likely represents the state of the shopping cart after the customer has removed some items. This table shows adjusted quantities and a new grand total price, reflecting the changes made to the cart.

Table 5.5: Customer Deletes Plants From The Shopping Cart Test Data

| **Customer Email** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Total(price\*quantity)** | **Total Price** |
| --- | --- | --- | --- | --- | --- | --- |
| chiaamber@gmail.com | Sunflower Plant | 2 | RM30.00 | 1 | RM30.00 | RM130.00 |
|  | Adenium obesum | 6 | RM30.00 | 3 | RM90.00 | RM130.00 |
|  | Ruellia Pink | 8 | RM10.00 | 1 | RM10.00 | RM130.00 |

Table 5.6 titled "Customer Searches A Plant By Plant Name Test Data". It is structured to showcase a search result interface where customers can find detailed information on plants. The table includes columns for Plant Name, Plant ID, Plant Description, Price, Availability, and Reviews. Two plants are listed: "Aloe Vera" and "Sunflower Plant", each with a corresponding unique ID, a detailed description, a listed price, the number of units available in stock, and a customer review. The 'Aloe Vera' entry is priced at RM10.00, has 144 units available, and includes a positive review noting a discrepancy in pot size but overall healthy plants. The 'Sunflower Plant' entry, highlighted, suggests a test case scenario with a price of RM30.00, 99 units available, and a review mentioning affordable prices. This table is likely used to validate the search functionality and the display of plant details on the platform.

Table 5.6: Customer Searches A Plant By Plant Name Test Data

| **Plant Name** | **Plant ID** | **Plant Description** | **Price** | **Availability** | **Reviews** |
| --- | --- | --- | --- | --- | --- |
| Aloe Vera | 1 | Aloe vera is a succulent plant species of the genus Aloe. The plant is stemless with thick, greenish, fleshy leaves that can fan out from the plant’s central stem. The margin of the leaf is serrated with small teeth. | RM10.00 | 144 | Good!  Received a different pot size than expected, but the plants were healthy. |
| Sunflower Plant | 2 | Sunflowers, with their bright yellow blooms resembling giant daisies, are easy to grow as heat-tolerant, pest-resistant, and fast-growing annual plants native to North America. | RM30.00 | 99 | Affordable prices |

Table 5.7, which is titled "Customer Makes A Payment Test Data" and appears to document various test cases for customer purchases within the system. The table includes columns for Customer Email, Customer State, Plant Name, Plant ID, Price, Quantity, Receive Method, Shipping Fee, Total Price (calculated as Price times Quantity), and Total Cost (which includes the shipping fee if applicable).Highlighted rows indicate specific test data entries for verification purposes. Some items are marked for pickup with no shipping fee, while others are to be delivered with corresponding shipping fees. The table ensures that the total costs are correctly calculated, taking into account whether a shipping fee should be added based on the chosen receive method (pickup or delivery). It's a comprehensive test to ensure the accuracy of payment and shipping fee calculations in the system's checkout process.

Table 5.7: Customer Makes A Payment Test Data

| **Customer Email** | **Customer state** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Receive Method** | **Shipping fee** | **Total**  **(price\*quantity)** | **Plant Price** | **Total Cost(include shipping fee if applicable)** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| chiaamber@gmail.com | Sabah | Sunflower Plant | 2 | RM30.00 | 1 | Pickup | - | RM30.00 | RM130.00 | RM130.00 |
|  | Sabah | Adenium obesum | 6 | RM30.00 | 3 | Pickup | - | RM90.00 | RM130.00 | RM130.00 |
|  | Sabah | Ruellia Pink | 8 | RM10.00 | 1 | Pickup | - | RM10.00 | RM130.00 | RM130.00 |
|  | Sabah | Aloe Vera | 1 | RM10.00 | 2 | Delivery | RM12.00 | RM20.00 | RM30.00 | RM42.00 |
|  | Sabah | Ruellia Pink | 8 | RM10.00 | 1 | Delivery | RM12.00 | RM10.00 | RM30.00 | RM42.00 |
|  | Selangor | Camellia Azalea | 9 | RM15.00 | 1 | Pickup | - | RM15.00 | RM15.00 | RM15.00 |
|  | Selangor | Cactus | 3 | RM15.00 | 2 | Delivery | RM7.00 | RM30.00 | RM80.00 | RM87.00 |
|  | Selangor | African Daisy | 7 | RM25.00 | 2 | Delivery | RM7.00 | RM50.00 | RM80.00 | RM87.00 |

Table 5.8, which is titled "Customer Views The Order Details Of Ongoing Orders Test Data." This table is likely used to validate the order details page.It lists several orders with columns for Order ID, Order Status, Order Date, Total Price, Plant Name, Quantity, and Price (calculated as plant price times quantity).The orders are displayed with varied statuses such as 'Waiting', 'Ready', 'Completed', and 'Out of Delivery', and are all dated Feb.7,2024. The table breaks down each order by individual plants, showing the quantity ordered and the total price for each type of plant. This setup allows the testing of the order detail view to ensure that the status updates are accurate and that the price calculations are correctly reflected for each order.

Table 5.8: Customer Views The Order Details Of Ongoing Orders Test Data

| **Order ID** | **Order Status** | **Order Date** | **Total Price** | **Plant Name** | **Quantity** | **Price(plant price\* quantity)** |
| --- | --- | --- | --- | --- | --- | --- |
| 56 | Waiting | Feb.7,2024 | RM130.00 | Sunflower Plant | 1 | RM30.00 |
|  |  | Feb.7,2024 | RM130.00 | Adenium obesum | 3 | RM90.00 |
|  |  | Feb.7,2024 | RM130.00 | Ruellia Pink | 1 | RM10.00 |
| 57 | Ready | Feb.7,2024 | RM42.00 | Aloe Vera | 2 | RM20.00 |
|  |  | Feb.7,2024 | RM42.00 | Ruellia Pink | 1 | RM10.00 |
| 58 | Completed | Feb.7,2024 | RM15.00 | Camellia Azalea | 1 | RM15.00 |
| 59 | Out of Delivery | Feb.7,2024 | RM87.00 | Cactus | 2 | RM30.00 |
|  |  | Feb.7,2024 | RM87.00 | African Daisy | 2 | RM50.00 |

Table 5.9 is titled "Customer Leaves Reviews And Ratings Test Data" and contains columns for Plant Name, Rating, and Review. It has a single entry for a plant called 'Camellia Azalea', which has received a 1-star rating and a review stating "Plants weren't as big as pictured". This table seems designed to test the functionality of submitting and displaying reviews and ratings on the platform.

Table 5.9: Customer Leaves Reviews And Ratings Test Data

| **Plant Name** | **Rating** | **Review** |
| --- | --- | --- |
| Camellia Azalea | 1 star | Plants weren't as big as pictured |

Table 5.10 is titled "Customer Manages The Wishlist Test Data" and lists the names of plants along with their prices, indicating the items a customer has added to their wishlist. It includes 'Cactus' priced at RM15.00 and 'Live Rose Plant' priced at RM35.00. This table likely serves to verify that the wishlist feature of the platform accurately records and reflects the customer's selections and their respective prices.

Table 5.10: Customer Manages The Wishlist Test Data

| **Plant Name** | **Price** |
| --- | --- |
| Cactus | RM15.00 |
| Live Rose Plant | RM35.00 |

## Acceptance Test

Module: Customer

Developer: Lim Cai Qing

Table 5.11: Customer Acceptance Test

| **Criterial** | **Fullfield?** | **Remarks** |
| --- | --- | --- |
| Customer changes their password |  |  |
| Customer edits profile details |  |  |
| Customer views added plants in the shopping cart |  |  |
| Customer adds plants to the shopping cart |  |  |
| Customer deletes plants from the shopping cart |  |  |
| Customer searches a plant by plant name |  |  |
| Customer makes a payment |  |  |
| Customer views the order details of ongoing orders |  |  |
| Customer leaves reviews and ratings |  |  |
| Customer manages the wishlist |  |  |

Date tested:

% Completed:

Test by:

Verified by:

## Test Results

Customer Changes Their Password Test Result:

| Figure 5.1 displays an error during a password change because the new password closely matches the customer's email. The user is asked to create a password that isn't too similar to their personal details. | | | |
| --- | --- | --- | --- |
| **Customer Name** | **Customer Email** | **Customer Old Password** | **Customer New Password** |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | chiaamber@gmail.com |
| Figure 5.1: Password Too Similiar To The Personal Information Test Result | | | |
| Figure 5.2 shows an error message for a password that's too short, asking for at least 8 characters. | | | |
| **Customer Name** | **Customer Email** | **Customer Old Password** | **Customer New Password** |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | chia33 |
| Figure 5.2: Password Too Short Test Result | | | |
| Figure 5.3 displays a password update attempt being rejected because the new password is a commonly used one. The system suggests choosing a less common password. | | | |
| **Customer Name** | **Customer Email** | **Customer Old Password** | **Customer New Password** |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | qwertyuiop |
| Figure 5.3: Password Too Common Test Result | | | |
| Figure 5.4 shows an error due to a password consisting only of numbers, which is not allowed. The system requires the password to be more complex. | | | |
| **Customer Name** | **Customer Email** | **Customer Old Password** | **Customer New Password** |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | 123456 |
| Figure 5.4: Password Is Entirely Numeric Test Result | | | |
| Figure 5.5 shows a password change error because the old password was entered incorrectly. The user is prompted to enter the correct old password. | | | |
| **Customer Name** | **Customer Email** | **Customer Old Password** | **Customer New Password** |
| Amber Chia | chiaamber@gmail.com | amber\*\*33 | soya\_milk |
| Figure 5.5: Old Password Was Entered Incorrectly Test Result | | | |
| Figure 5.6 displays a confirmation message that the customer's password has been successfully updated in the account settings section. | | | |
| **Customer Name** | **Customer Email** | **Customer Old Password** | **Customer New Password** |
| Amber Chia | chiaamber@gmail.com | amber\*\*333 | soya\_milk |
| Figure 5.6:Customer Successfully Changed Password | | | |

Customer Edits Profile Details Test Result:

| Figure 5.7 indicates an error in the profile editing form due to the full name not meeting the required character length; it must be between 10 and 50 characters. | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Customer Name** | **Customer Email** | **Customer Phone Number** | **Customer Address** | **Customer State** | **Customer IC** |
| Amber | chiaamber@gmail.com | 601123498766 | 35,Jalan BPU6 Taman Indah Puchong | Selangor | 040202106450 |
| Figure 5.7: Full Name Must Be Between 10 and 50 Characters Test Result | | | | | |
| Figure 5.8 shows the test result when a customer tries to edit their profile without ‘@’ in their email. An error message will pop up asking the customer to include ‘@’ in their email. | | | | | |
| **Customer Name** | **Customer Email** | **Customer Phone Number** | **Customer Address** | **Customer State** | **Customer IC** |
| AmberChia0202 | chiaambergmail.com | 601123498766 | 35,Jalan BPU6 Taman Indah Puchong | Selangor | 040202106450 |
| Figure 5.8: Email Does Not Include ‘@’ Test Result | | | | | |
| Figure 5.9 shows an error message regarding an invalid phone number format in the profile editing section, indicating that the phone number should start with 60 and be followed by 9 or 10 digits. | | | | | |
| **Customer Name** | **Customer Email** | **Customer Phone Number** | **Customer Address** | **Customer State** | **Customer IC** |
| AmberChia0202 | chiaamber@gmail.com | 6011234987665abc | 35,Jalan BPU6 Taman Indah Puchong | Selangor | 040202106450 |
| Figure 5.9: Invalid Phone Number Format Test Result | | | | | |
| Figure 5.10 shows an error message in the profile editing form due to an improperly entered state abbreviation. The system prompts for a valid state entry. | | | | | |
| **Customer Name** | **Customer Email** | **Customer Phone Number** | **Customer Address** | **Customer State** | **Customer IC** |
| AmberChia0202 | chiaamber@gmail.com | 601123498765 | Jalan Muzium, 88300 Kota Kinabalu, Sabah | Sa | 040202106450 |
| Figure 5.10: Invalid State Entry Test Result | | | | | |
| Figure 5.11 shows an error message for an incorrectly formatted IC number in the profile editing form, indicating that the IC number must consist of 12 digits. | | | | | |
| **Customer Name** | **Customer Email** | **Customer Phone Number** | **Customer Address** | **Customer State** | **Customer IC** |
| AmberChia0202 | chiaamber@gmail.com | 601123498765 | Jalan Muzium, 88300 Kota Kinabalu, Sabah | Sabah | 040202106450mm |
| Figure 5.11: IC Number Format Error Test Result | | | | | |
| Figure 5.12 displays a confirmation message that the customer's profile details have been successfully updated, including the full name, email, phone number, address, state, and IC number. | | | | | |
| **Customer Name** | **Customer Email** | **Customer Phone Number** | **Customer Address** | **Customer State** | **Customer IC** |
| AmberChia0202 | chiaamber@gmail.com | 601123498765 | Jalan Muzium, 88300 Kota Kinabalu, Sabah | Sabah | 040202106452 |
| Figure 5.12: Successful Profile Update Confirmation | | | | | |

Customer View Added Plants In The Shopping Cart Test Result:

| Figure 5.13 presents the 'Shopping Cart' section of the system. It shows a message that the shopping cart is currently empty with a total price of RM0, indicating that no plants have been added to the cart by the customer. The system is prepared to guide customers through the process of adding items to their cart and proceeding to checkout when they are ready. | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Customer Email** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Total(price\*quantity)** | **Total Price** |
| chiaamber@gmail.com | - | - | - | - | - | - |

| Figure 5.13: Empty Shopping Cart Display Test Result | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Figure 5.14 shows the shopping cart interface with a list of plants added by a customer. Each plant item includes a thumbnail image, the plant's name, ID, price, and quantity, with an individual total for each item and a combined total price at the bottom. | | | | | | |
| **Customer Email** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Total(price\*quantity)** | **Total Price** |
| chiaamber@gmail.com | Sunflower Plant | 2 | RM30.00 | 1 | RM30.00 | RM150.00 |
|  | Adenium obesum | 6 | RM30.00 | 3 | RM90.00 | RM150.00 |
|  | Cactus | 3 | RM15.00 | 2 | RM30.00 | RM150.00 |
| Figure 5.14: Customer View Added Plants In The Shopping Cart Test Result | | | | | | |

Customer Add Plants To The Shopping Cart Test Result:

| Figure 5.15 illustrates a customer's shopping cart after adding plants. It includes each plant's thumbnail image, name, ID, price, and quantity, with an individual total price for each plant and the total cost of the cart displayed at the bottom.The shopping cart reflects the customer's selections from the 'Plant List' section, confirming successful addition to the cart. | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Customer Email** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Total(price\*quantity)** | **Total Price** |
| chiaamber@gmail.com | Sunflower Plant | 2 | RM30.00 | 1 | RM30.00 | RM160.00 |
|  | Adenium obesum | 6 | RM30.00 | 3 | RM90.00 | RM160.00 |
|  | Cactus | 3 | RM15.00 | 2 | RM30.00 | RM160.00 |
|  | Ruellia Pink | 8 | RM10.00 | 1 | RM10.00 | RM160.00 |
| Figure 5.15 Customer Add Plants To The Shopping Cart Test Result | | | | | | |

Customer Deletes Plants From The Shopping Car Test Result:

| Figure 5.16 illustrates the shopping cart interface after the customer has removed selected plants. It displays the remaining plants in the cart, each with its thumbnail image, name, ID, price, and quantity. The total price of the cart is updated to reflect the removed items. This confirms that the customer has successfully managed their shopping cart by removing selected plants. | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Customer Email** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Total(price\*quantity)** | **Total Price** |
| chiaamber@gmail.com | Sunflower Plant | 2 | RM30.00 | 1 | RM30.00 | RM130.00 |
|  | Adenium obesum | 6 | RM30.00 | 3 | RM90.00 | RM130.00 |
|  | Ruellia Pink | 8 | RM10.00 | 1 | RM10.00 | RM130.00 |
| Figure 5.16: Customer Deletes Plants From The Shopping Cart Test Result | | | | | | |

Customer Searches A Plant By Plant Name Test Result:

| Figure 5.17 illustrates the 'Plant List' section of the system when a search yields no results. The customer has attempted to search for a plant using the term 'purple flower', and the system has processed the query. However, it has not found any matching entries within the database. A message is displayed to inform the customer that "No plants match the entered criteria", which is an indication of the system's response in the event of an unsuccessful search. This test result confirms the system's capability to handle searches where no existing database entries meet the search parameters, ensuring the customer is appropriately notified. | | | | | |
| --- | --- | --- | --- | --- | --- |
| Figure 5.17: No Match Found in Plant Search Test Result | | | | | |
| Figure 5.18 demonstrates a successful search result within the 'Plant List' section of the system for the plant name "Aloe Vera". The system displays the plant's ID, description, price, availability, and reviews after the customer performs the search. The plant ID is listed as "1", indicating it is the first entry in the system's database. The description provides details about the Aloe Vera being a succulent species, its physical characteristics, and the serrated margin of its leaves. The price is set at RM10.00, with an availability of 144 units, and the customer review highlights a positive experience with a note about receiving a plant in a different pot size than expected. This test result confirms the system's functionality in retrieving and displaying detailed information for a plant based on the entered name. | | | | | |
| **Plant Name** | **Plant ID** | **Plant Description** | **Price** | **Availability** | **Reviews** |
| Aloe Vera | 1 | Aloe vera is a succulent plant species of the genus Aloe. The plant is stemless with thick, greenish, fleshy leaves that can fan out from the plant’s central stem. The margin of the leaf is serrated with small teeth. | RM10.00 | 144 | Good!  Received a different pot size than expected, but the plants were healthy. |
| Figure 5.18: Successful Plant Search by Name - Aloe Vera | | | | | |
| Figure 5.19 captures the successful search for the "Sunflower Plant" in the 'Plant List' section. After the search is conducted, the system exhibits the corresponding entry with details including the plant name, ID "2", a thorough description that describes the Sunflowers as bright yellow blooms, the price at RM30.00, and the available quantity of 99 units. The review section cites an affordable price point, indicating customer satisfaction with the value proposition. This search result validates the system's effective display of pertinent plant information to the customer following a search by plant name. | | | | | |
| Sunflower Plant | 2 | Sunflowers, with their bright yellow blooms resembling giant daisies, are easy to grow as heat-tolerant, pest-resistant, and fast-growing annual plants native to North America. | RM30.00 | 99 | Affordable prices |
| Figure 5.19: Successful Plant Search by Name - Sunflower Plant | | | | | |

Customer Makes A Payment Test Result:

| Figure 5.20 depicts a shopping cart with specific plants selected, their prices, and the aggregated total cost. Proceeding to Figure 5.21, the customer is given checkout options and opts for pickup, which does not incur shipping fees. Figure 5.22 offers payment choices, with the customer selecting credit/debit card for this transaction, as highlighted by the absence of a shipping fee due to the pickup option. Finally, Figure 5.23 displays a successful payment confirmation, complete with an order ID, and navigates the customer back to the Plant List page, concluding the purchase sequence. | | | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Customer Email** | **Customer state** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Receive Method** | **Shipping fee** | **Total**  **(price\*quantity)** | **Plant Price** | **Total Cost(include shipping fee if applicable)** |
| chiaamber@gmail.com | Sabah | Sunflower Plant | 2 | RM30.00 | 1 | Pickup | - | RM30.00 | RM130.00 | RM130.00 |
|  | Sabah | Adenium obesum | 6 | RM30.00 | 3 | Pickup | - | RM90.00 | RM130.00 | RM130.00 |
|  | Sabah | Ruellia Pink | 8 | RM10.00 | 1 | Pickup | - | RM10.00 | RM130.00 | RM130.00 |
| Figure 5.20: Customer Shopping Cart Overview    Figure 5.21: Checkout Options    Figure 5.22: Payment Selection    Figure 5.23: Order Confirmation | | | | | | | | | | |
| In Figure 5.24, the customer's shopping cart displays Aloe Vera and Ruellia Pink plants, with a total item cost of RM30.00. Figure 5.25 offers checkout options, and the customer selects delivery, which incurs a RM12.00 shipping fee due to the destination being Sabah, as detailed in Figure 5.26. The delivery details show the address and total cost including the specific shipping fee. Figure 5.27 shows the payment selection screen, where the customer chooses to pay with a credit/debit card, and the total cost reflects the RM12.00 shipping fee for Sabah. Finally, Figure 5.28 confirms the order with a successful payment and order ID, concluding the transaction and returning the customer to the Plant List page. | | | | | | | | | | |
| **Customer Email** | **Customer state** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Receive Method** | **Shipping fee** | **Total**  **(price\*quantity)** | **Plant Price** | **Total Cost(include shipping fee if applicable)** |
| chiaamber@gmail.com | Sabah | Aloe Vera | 1 | RM10.00 | 2 | Delivery | RM12.00 | RM20.00 | RM30.00 | RM42.00 |
|  | Sabah | Ruellia Pink | 8 | RM10.00 | 1 | Delivery | RM12.00 | RM10.00 | RM30.00 | RM42.00 |
| Figure 5.24: Customer Shopping Cart Overview    Figure 5.25: Checkout Options    Figure 5.26: Delivery Details    Figure 5.27: Payment Selection    Figure 5.28: Order Confirmation | | | | | | | | | | |
| Figure 5.29 shows the customer's shopping cart with a single Camellia Azalea plant, priced at RM15.00, ready for checkout. In Figure 5.30, the customer is at the checkout stage, selecting 'Pickup' as the receive method, which eliminates any shipping fee as the customer is in Selangor and not in the higher-fee regions of Sabah or Sarawak. Proceeding to Figure 5.31, the total cost at the payment selection reflects only the cost of the plant since the pickup option does not require a shipping fee. Lastly, Figure 5.32 confirms the successful processing of the order with the display of an order ID, signifying the completion of the purchase and the return to the Plant List page. | | | | | | | | | | |
| **Customer Email** | **Customer state** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Receive Method** | **Shipping fee** | **Total**  **(price\*quantity)** | **Plant Price** | **Total Cost(include shipping fee if applicable)** |
| chiaamber@gmail.com | Selangor | Camellia Azalea | 9 | RM15.00 | 1 | Pickup | - | RM15.00 | RM15.00 | RM15.00 |
| Figure 5.29: Customer Shopping Cart Overview    Figure 5.30: Checkout Options    Figure 5.31: Payment Selection    Figure 5.32: Order Confirmation | | | | | | | | | | |
| Figure 5.33 presents the customer's shopping cart with a selection of Cactuses and African Daisies, totaling RM80.00. In Figure 5.34, the customer is at the checkout stage with options for pickup or delivery. The customer chooses delivery to Selangor, which comes with a lower shipping fee of RM7.00, as shown in Figure 5.35. The delivery details confirm the shipping fee and the total cost. In Figure 5.36, the customer proceeds to the payment selection screen where the total cost reflects the plant price plus the Selangor-specific shipping fee. Finally, Figure 5.37 displays the order confirmation with the successful processing of payment and order ID, closing the transaction cycle and navigating the customer back to the Plant List page. | | | | | | | | | | |
| **Customer Email** | **Customer state** | **Plant Name** | **Plant ID** | **Price** | **Quantity** | **Receive Method** | **Shipping fee** | **Total**  **(price\*quantity)** | **Plant Price** | **Total Cost(include shipping fee if applicable)** |
| chiaamber@gmail.com | Selangor | Cactus | 3 | RM15.00 | 2 | Delivery | RM7.00 | RM30.00 | RM80.00 | RM87.00 |
|  | Selangor | African Daisy | 7 | RM25.00 | 2 | Delivery | RM7.00 | RM50.00 | RM80.00 | RM87.00 |
| Figure 5.33: Customer Shopping Cart Overview    Figure 5.34: Checkout Options    Figure 5.35: Delivery Details    Figure 5.36: Payment Selection    Figure 5.37: Order Confirmation | | | | | | | | | | |

Customer Views The Order Details Of Ongoing Orders Test Result:

| Figure 5.38 displays a user interface within a customer account showing that there are no ongoing orders. The message "You have no ongoing orders." is clearly presented, indicating that either all previous orders have been fulfilled or that the customer has not placed any orders yet. | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Order ID** | **Order Status** | **Order Date** | **Total Price** | **Plant Name** | **Quantity** | **Price(plant price\* quantity)** |
| - | - | - | - | - | - | - |

| Figure 5.38: No Ongoing Orders Display Test Result | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| Figure 5.39 displays a customer's order details within the "My Orders" section of the online plant store. The order, dated February 7, 2024, is listed with an "Order Status: Waiting" and includes a total price of RM130.00. The customer has purchased various plants—Sunflower Plant, Adenium obesum, and Ruellia Pink—in specific quantities, with the individual prices contributing to the total order cost. This view allows the customer to track the progress and details of their orders that are currently being processed. | | | | | | |
| **Order ID** | **Order Status** | **Order Date** | **Total Price** | **Plant Name** | **Quantity** | **Price(plant price\* quantity)** |
| 56 | Waiting | Feb.7,2024 | RM130.00 | Sunflower Plant | 1 | RM30.00 |
|  |  | Feb.7,2024 | RM130.00 | Adenium obesum | 3 | RM90.00 |
|  |  | Feb.7,2024 | RM130.00 | Ruellia Pink | 1 | RM10.00 |
| Figure 5.39: Customer Views The Order Details Of Ongoing Orders Test Result | | | | | | |
| Figure 5.40 illustrates customer's order overview, revealing that Order ID 57, placed on February 7, 2024, is now ready. The total price of the order is RM42.00, which includes 2 units of Aloe Vera at RM20.00 and 1 unit of Ruellia Pink at RM10.00. | | | | | | |
| **Order ID** | **Order Status** | **Order Date** | **Total Price** | **Plant Name** | **Quantity** | **Price(plant price\* quantity)** |
| 57 | Ready | Feb.7,2024 | RM42.00 | Aloe Vera | 2 | RM20.00 |
|  |  | Feb.7,2024 | RM42.00 | Ruellia Pink | 1 | RM10.00 |
| Figure 5.40: Customer Views The Order Details Of Ongoing Orders Test Result | | | | | | |
| Figure 5.41 illustrates the details of a completed order from a customer's perspective. The order, identified by Order ID 58, is marked as 'Completed' and includes the purchase of a Camellia Azalea. The record shows a total price of RM15.00 for the single item ordered and confirms the order date as February 7, 2024, completing the customer's purchase journey. | | | | | | |
| **Order ID** | **Order Status** | **Order Date** | **Total Price** | **Plant Name** | **Quantity** | **Price(plant price\* quantity)** |
| 58 | Completed | Feb.7,2024 | RM15.00 | Camellia Azalea | 1 | RM15.00 |
| Figure 5.41: Customer Views The Order Details Of Ongoing Orders Test Result | | | | | | |
| Figure 5.42 displays an order summary for Order ID 59, which is currently marked as "Out Of Delivery", indicating that the items have been dispatched. The order, dated February 7, 2024, includes 2 quantities each of Cactus and African Daisy plants, priced at RM30.00 and RM50.00 respectively, totaling RM87.00 for the entire order. | | | | | | |
| **Order ID** | **Order Status** | **Order Date** | **Total Price** | **Plant Name** | **Quantity** | **Price(plant price\* quantity)** |
| 59 | Out Of Delivery | Feb.7,2024 | RM87.00 | Cactus | 2 | RM30.00 |
|  |  | Feb.7,2024 | RM87.00 | African Daisy | 2 | RM50.00 |
| Figure 5.42: Customer Views The Order Details Of Ongoing Orders Test Result | | | | | | |

Customer Leaves Reviews And Ratings Test Result:

| Figure 5.43 illustrates the "To Review" section within a customer’s account, indicating that there are currently no items available to review. The message "You have no items to review." suggests that either all previous purchases have already been reviewed or that recently completed orders have not yet been listed for review. | | |
| --- | --- | --- |
| Figure 5.43: No Review Display Test Result | | |
| Figure 5.44 displays the review interface where a customer has given a one-star rating to the Camellia Azalea, noting the plant size was not as advertised. The review is ready for submission.  Figure 5.45 confirms that the customer's review has been successfully submitted, as indicated by a notification message.In Figure 5.46, the submitted review is visible on the plant listing page, showing the customer's username, their one-star rating, and the comment that the Camellia Azalea was not as big as pictured. | | |
| **Plant Name** | **Rating** | **Review** |
| Camellia Azalea | 1 star | Plants weren't as big as pictured |
| Figure 5.44: Customer Leaves Reviews And Ratings Test Result    Figure 5.45: Confirmation of review    Figure 5.46: Customer review publish at plant list page | | |

Customer Manages The Wishlis Test Result:

| Figure 5.47 displays a user interface for a customer's wishlist feature within the system. It shows that the customer's wishlist is currently empty, with a message clearly stating "Your wishlist is empty." There is a button labeled "Go to Plant List," suggesting that the customer can navigate back to the selection of plants available for purchase and potentially add items to their wishlist. | |
| --- | --- |
| Figure 5.47: Empty Wishlist Display Test Result | |
| Figure 5.48 shows a customer's wishlist.It contains two entries: a "Cactus" priced at RM15.00 and a "Live Rose Plant" priced at RM35.00. Accompanying the text are images of the respective plants, giving the customer a visual confirmation of what they have added to their wishlist. Each plant entry has a "Delete" option, allowing the customer to remove the item from the wishlist if they decide against purchasing it. | |
| **Plant Name** | **Price** |
| Cactus | RM15.00 |
| Live Rose Plant | RM35.00 |
| Figure 5.48: Customer Manages The Wishlis Test Result | |

# Conclusion

## Project Achievements

Our nursery plant shopping system is now fully operational, serving administrators, customers, guests, and delivery man. Despite encountering some deviations from our original schedule, we managed to complete and submit the project on time, delivering a functional platform. My primary focus was on developing customer functionalities, including password changes, profile editing, managing shopping cart items, making payments, and interacting with orders, reviews, and wishlists. Through this project, I refined my skills in Django framework development, refreshed my Python knowledge, and gained proficiency in Bootstrap for frontend design. These enhanced skills have better equipped me for future system development projects, reducing potential challenges and time constraints. Our system boasts an intuitive user interface, improved search functionality, and simplified registration processes, ensuring a seamless experience for all users. Additionally, the integration of customer reviews and ratings encourages community engagement and informed decision-making, in line with our initial assumptions, and enhances our market appeal.

## Quality Assurance

In our project for a nursery plant shopping system, we focused on delivering top-quality software by taking several important steps. Firstly, we carefully planned and reviewed each part of the system to catch any issues early on. We also created a detailed project plan and used methods to manage changes effectively. After finishing a section, we all checked it together to fix any mistakes and make improvements. Throughout development, we made sure to include test data and conduct thorough reviews to meet our software requirements. Communication was key, with constant reminders and note-taking to address any problems quickly. We prioritized user satisfaction by creating test plans, ensuring all features worked smoothly, and maintaining an easy-to-use website. By following Django's guidelines and using its tools, we kept our code clean and secure. We also paid close attention to performance, making sure the site remained fast and reliable.

## Problems Encountered

During the development of our nursery plant shopping system, we faced several challenges, particularly in learning the Django framework and Python from scratch, which required extensive research and testing to understand. The lack of experience with Django and Bootstrap caused delays in our schedule, compounded by the pressure of other assignments. Despite initially planning for additional features like sorting and filtering, time constraints forced us to prioritize essential tasks. Effective teamwork was also crucial, requiring improved communication and organization to stay on track. Despite these hurdles, each obstacle presented a valuable learning opportunity and contributed to the project's eventual success.

## Remarks/Comments

At the beginning, I was really stressed because I had never heard of the Django framework before, and the due date was also tight. So, I had to stay up every night to study the Django framework, watch tutorials, and understand it. I did consider giving up because it was really rushed, and I had other assignments to do as well. But I'm very thankful that I didn't give up and finally managed to complete it. Of course, I also learned that good communication with teammates is crucial because we had some disagreements during the process. However, I knew that compromise was necessary, and we all had to give a little to achieve our goals. Although the process was very tough, it was also a valuable learning experience.