

TSE2101 Final Report

for

Nursery Plant Shopping System

Version <3.0>

Tutorial Section: TT3L

Group No.: Group 3

EMILY PHANG RU YING	1211102687
TEO YU JIE	1211102751
LIM CAI QING	1211102753
TOH EE LIN	1221303203

Date: 11/02/2024

Contents

1 Project Management.....	4
1.1 Team Members.....	4
1.2 Project Plan.....	4
2 System Overview.....	7
2.1 Description.....	7
2.2 Actors.....	7
2.3 Assumptions and Dependencies.....	8
2.4 Use Case Diagram.....	9
3 Requirements.....	10
3.1 Class Diagrams.....	10
3.2 State Diagrams.....	11
3.3 Data Flow Diagrams.....	16
4 Design.....	19
4.1 Data Dictionary.....	19
4.2 Data Structures.....	29
4.3 Software Architecture.....	36
4.4 Main Screens.....	42
4.5 Main Components.....	65
4.6 Deployment Diagram.....	66
5 Implementation.....	69
5.1 Development Environment.....	69
5.2 Software Integration.....	70
5.3 Database.....	83
6 Testing.....	92
6.1 Testing Strategy.....	92
6.2 Test Data.....	92
6.3 Acceptance Tests.....	114
7 Sample Screens.....	117
7.1 Main Screens.....	117
7.2 Administrator.....	118
7.3 Guest.....	124
7.4 Customer.....	129
7.5 Delivery Man.....	141
8 Conclusion.....	147
8.1 Completion of Software.....	147
8.2 Software Quality Assurance.....	147
8.3 Group Collaboration.....	147
8.4 Problems Encountered.....	148
8.5 Remarks/Comments.....	148

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Version 1.0	Emily Phang Ru Ying Teo Yu Jie Lim Cai Qing Toh Ee Lin	This version outlines the basic requirements for the nursery plant shopping system which includes the use case diagram with detailed description by using activity diagrams with scenario and class diagram with explanation.	02/12/2023
Version 2.0	Emily Phang Ru Ying Teo Yu Jie Lim Cai Qing Toh Ee Lin	This version includes an improved use case diagram with use case descriptions followed by sequence diagrams. It also includes the development of the database design with a data dictionary, data structures as well as a class diagram. Architecture design is also provided with the three-tier architecture and subsystems explained in detail. More detailed state transition diagrams, interface design, activity diagrams, and deployment diagrams are provided to further explain the flow of our nursery plant shopping system.	12/01/2024
Version 3.0	Emily Phang Ru Ying Teo Yu Jie Lim Cai Qing Toh Ee Lin	This version includes the latest project management and system overview, with clearer use case diagrams. It also includes the updated class diagram, state diagrams, data flow diagram, data dictionary, data structures and architecture design. The implementation section covers the development environment and database structure, while the testing section details the test data and results. Besides, this version showcases user experience through sample screens for each actor, and concludes with a brief summary of our project's completion, challenges, and insights.	11/02/2024

1 Project Management

1.1 Team Members

Table 1.1 outlines the assignments of team members to specific actors within the project. Each team member is designated to fulfill a role corresponding to a particular actor involved in the system.

Table 1.1: Team Member Assignments for Each Actor

Name	Actor/Processes
Toh Ee Lin	Administrator
Teo Yu Jie	Guest
Lim Cai Qing	Customer
Emily Phang Ru Ying	Delivery Man

1.2 Project Plan

Table 1.2 shows the timeline for the TLET Nursery Plant Shopping System Project. It outlines the start and end dates for each stage of the project, from planning and analysis to development, testing, and report writing.

Table 1.2: Project Timeline for TLET Nursery Plant Shopping System Project

Tasks	Start Date	End Date
Project Planning	1/11/2023	10/11/2023
Requirement Analysis	1/11/2023	10/11/2023
Use Case Diagram	11/11/2023	22/11/2023
Class Diagram	23/11/2023	2/12/2023
Data Design	3/12/2023	7/12/2023
Architecture Design	8/12/2023	18/12/2023
Interface Design	19/12/2023	26/12/2023
Component Design	27/12/2023	4/1/2023
Deployment Design	5/1/2023	12/1/2023

Software Development	13/1/2023	1/2/2023
Software Testing	2/2/2023	7/2/2023
Writing Final Report	8/2/2023	11/2/2023

Figure 1.1 below shows the Gantt Chart for TLET Nursery Plant Shopping System Project. Our nursery plant shopping system project begins on the 1st of November 2023 and ends on 11th of February 2024, spanning a total of 103 days. This project is divided into three distinct parts: Project Part 1 is aimed for completion by the 2nd of December 2023; Project Part 2 by the 12th of January 2024; and Project Part 3 by the project's end date, 11th of February 2024.

Project Part 1 is dedicated to the initial stages of development, focusing on foundational design elements such as the use case diagram to outline the system's functionality and activity diagrams to map out the sequence of actions taken by each actor within the system. This phase also includes creating class diagrams to define the main entities and their interactions within the system, ensuring a comprehensive understanding of the system architecture.

In Project Part 2, we dig into the more detailed parts of the system design. We work on the data design to build a database that fits what the system needs, and the architecture design, which shows how the whole system is put together. For the interface design, we create draft screenshots of what the system will look like to the user, whereas for the component design, we incorporate the activity diagrams that illustrate the flow of the system's processes and interactions. Lastly, we plan the deployment design, which is our roadmap for launching the system for actual use. This step-by-step process is crucial for preparing our system to go live.

Project Part 3 is the phase where we turn our plan into reality by developing the software. This involves writing code to create all the features we carefully designed earlier. Once development is underway, we shift our attention to comprehensive software testing, ensuring everything works correctly and reliably. The last task in this phase is to write the final report, which provides a detailed account of the methods employed throughout the project, the problems encountered, and our final thoughts on the project's execution.

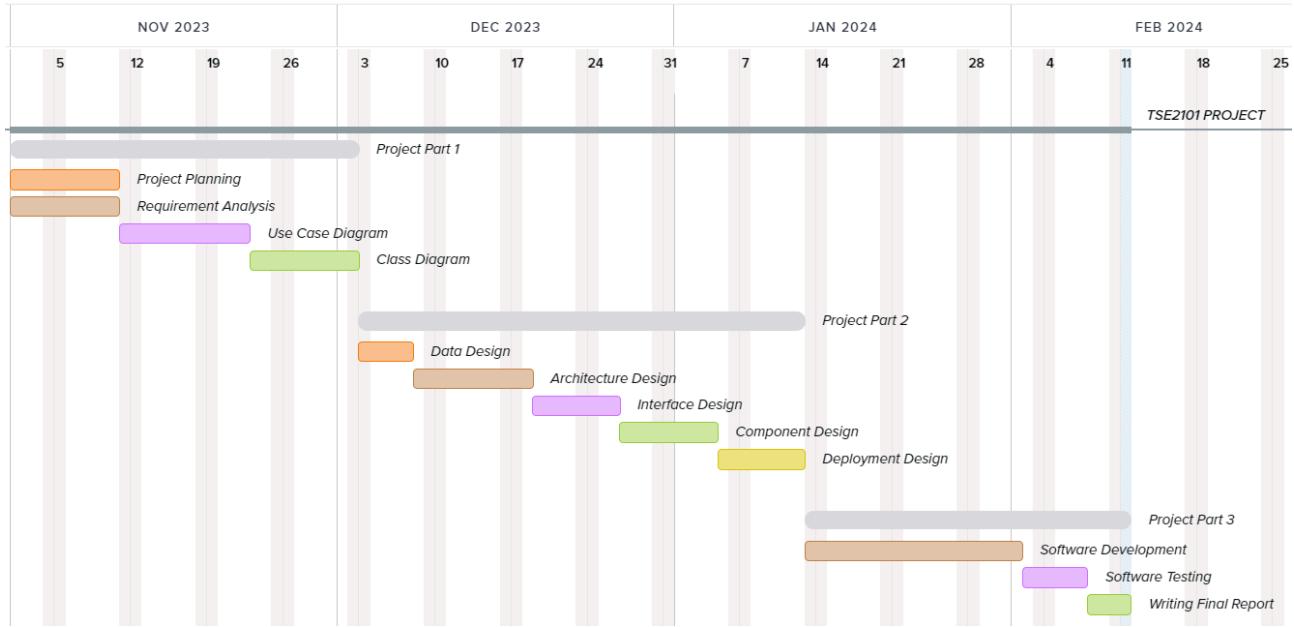


Figure 1.1: Gantt Chart for TLET Nursery Plant Shopping System Project

2 System Overview

2.1 Description

The Nursery Plant Shopping System serves as a platform that facilitates interactions among four primary user roles: administrator, guests, customers, and delivery men. Administrators can manage plant inventory, oversee customer orders, and coordinate with delivery men to ensure a smooth and timely delivery process. Guests, granted the simplicity of searching for plants by name and viewing reviews, can effortlessly transition into customers or delivery men by creating their accounts. For customers, the system provides an intuitive and secure platform for browsing, searching, purchasing a diverse range of nursery plants, leaving reviews, and managing wishlists. Customers can also personalize their experience by editing profile details and changing passwords. Additionally, delivery men use the system to streamline order processing, from accepting delivery orders to confirming successful deliveries, making sure orders reach customers smoothly. This system combines plant management, order processing, customer engagement, and delivery logistics, ensuring a fluid and user-friendly experience for all parties involved.

2.2 Actors

Table 2.1 provides an overview of the various actors within the system and their corresponding use cases in the TLET Nursery Plant Shopping System. The actors include Administrators, responsible for overseeing the plants, orders, and deliveries; Guest, able to explore the plants, create accounts, and view reviews of plants; Customers, engaging in account management, plant shopping, order transactions, and leaving reviews; and Delivery Men, involved in the delivery order process.

Table 2.1: Actors and Respective Use Cases

Actor	Use Cases
Administrator	An administrator manages the plants. <ul style="list-style-type: none"> • Create a plant. (sub-use case) • Update a plant. (sub-use case) • Delete a plant. (sub-use case)
	An administrator searches a plant by plant name.
	An administrator views order details and updates order status.
	An administrator assigns deliveries to delivery men.
Guest	A guest browses the plant list.
	A guest searches a plant by plant name.
	A guest creates a user account.
	A guest views reviews and ratings.

Customer	A customer changes their password
	A customer can edit profile details.
	A customer shops plants. <ul style="list-style-type: none"> • 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. <ul style="list-style-type: none"> • 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.
Delivery Man	A delivery man creates a user account.
	A delivery man changes their password.
	A delivery man can edit profile details.
	A delivery man accepts a delivery order.
	A delivery man views accepted deliveries and confirms delivery.

2.3 Assumptions and Dependencies

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

1. The nursery plant shopping system is designed to have only one administrator responsible for its management and operation.
2. Regardless of the number of times a delivery order is declined, eventually, there will be a deliveryman who accepts and successfully completes the order.
3. 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.
4. 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.

5. Customers are always expected to check their order status to see if their order has been delivered and out for delivery.

2.4 Use Case Diagram

Figure 2.1 is a use case diagram for a Nursery Plant Shopping System, depicting the interactions between the actors and the system. There are four types of actors: Customer, Administrator, Delivery Man, and Guest. The Customer can browse and shop for plants, complete purchases, manage orders, leave reviews, manage wishlists, and change their account password. The Administrator has broader permissions, such as managing the plants, viewing and managing orders, as well as assigning deliveries to delivery men. The Delivery Man's role is focused on managing delivery orders whereas the Guest is able to browse plants, view reviews, and become a customer or a delivery man by creating the respective account.



Figure 2.1: Nursery Plant Shopping System Use Case Diagram

3 Requirements

3.1 Class Diagrams

Figure 3.1 shows the nursery plant shopping system class diagram. The system contains several classes with specific attributes and methods, forming a network of interactions primarily for a nursery plant shopping platform.

The User class is the foundation for the system's role-based functionality, with attributes including user_id, email, password, and role indicators (is_customer, is_deliveryman, is_admin). Extending from the User class is the Administrator class, which inherits User attributes and includes methods tailored for administrative tasks. The Administrator is associated with the Plant class to manage the plant and with the Order class to oversee order processing. The Order class is having a one-to-many relationship with the Orderitem class and this means a single order can contain multiple order items.

The Customer class, derived from User, is connected to the ShoppingCart class through a one-to-one relationship, enabling customers to add, manage, and review cart items before proceeding to checkout. ShoppingCart class contains CartItem class, which represents the individual items in a shopping cart. Besides, Plant Class is linked to CartItem class which means that each cart item is linked to a specific plant. Both Review class and Wishlist class are also linked to Customer class to provide functionality such as leaving reviews for different plants and adding plants that they are interested in purchasing into their wishlist. In this case, Wishlist class contains WishlistItem class, with each item related back to a specific plant.

The DeliveryMan class, also an extension of the User class, is linked to the Order class, facilitating the management of deliveries. The DeliveryMan class has methods such as acceptOrder() and doneDelivery(), which are essential for the execution and completion of the delivery process. Moreover, Payment class is associated with the Order class, ensuring that each order has a corresponding payment record for financial tracking.

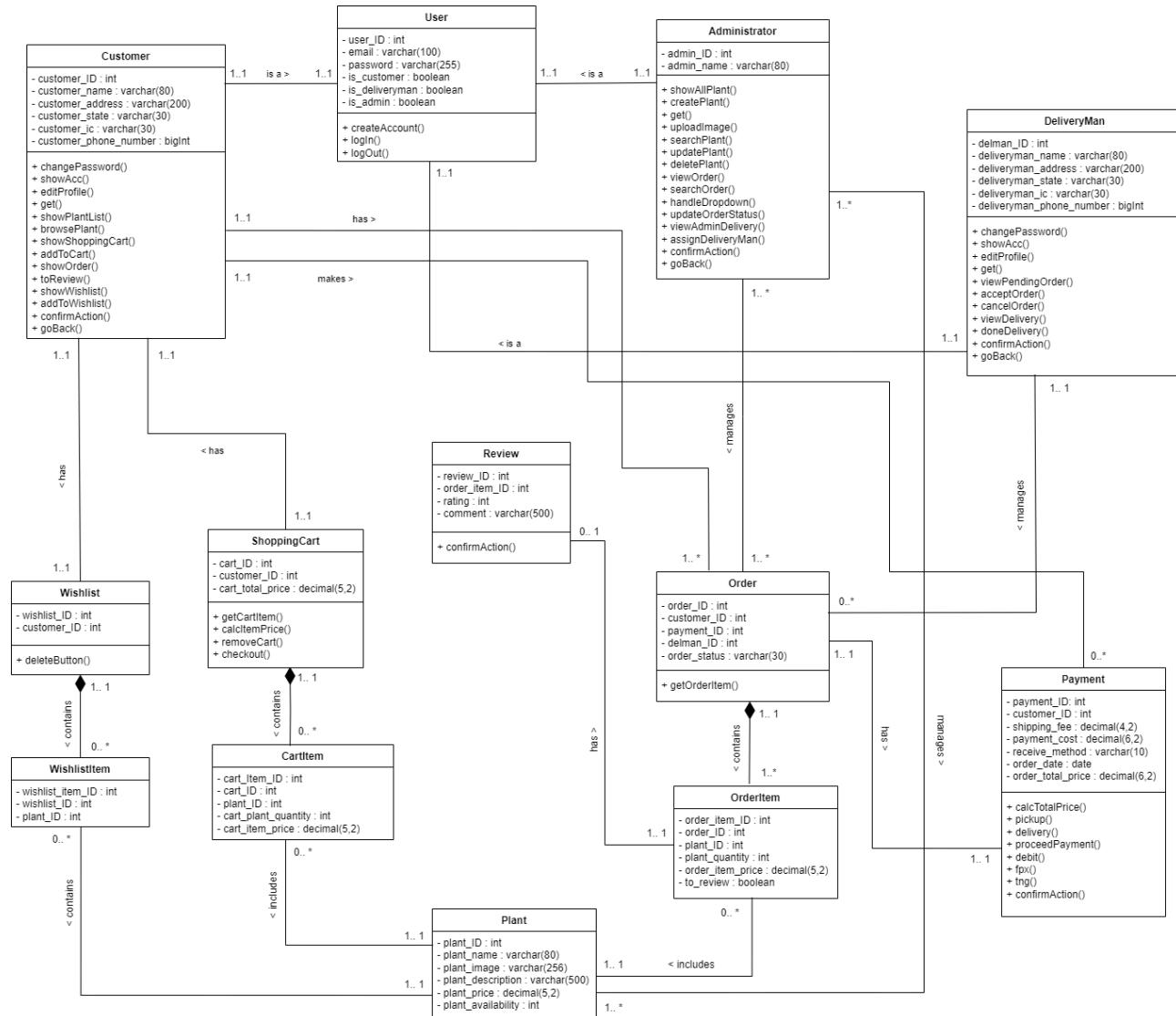


Figure 3.1: Nursery Plant Shopping System Class Diagram

3.2 State Diagrams

Figure 3.2 illustrates the State 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”.

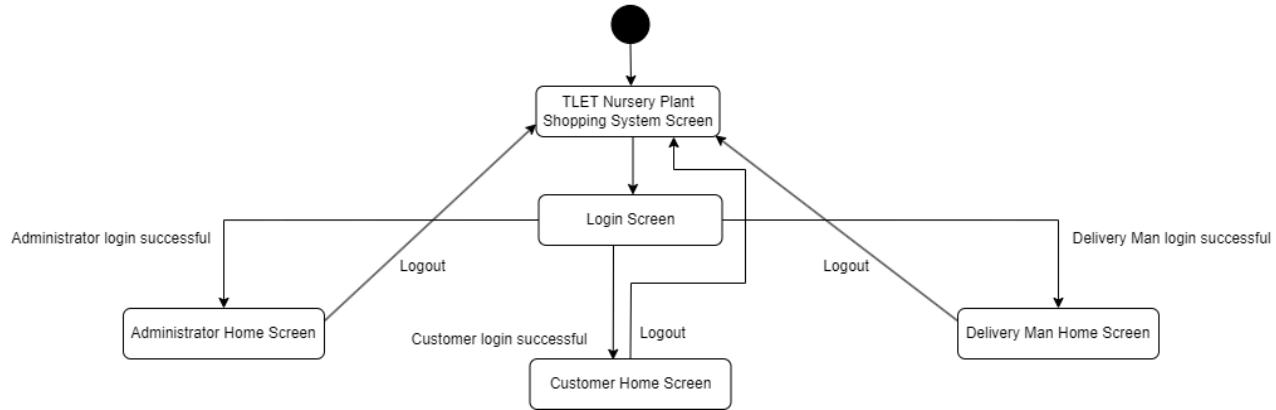


Figure 3.2: State Diagram for Nursery Plant Shopping System

3.2.1 Administrator State Diagram

Figure 3.3 illustrates the State Diagram for an administrator. Starting at the “Login Screen”, a successful login brings the administrator to the “Administrator Home Screen”, where the administrator has the option to manage plants, orders, or deliveries. The “Plant Management Screen” is reached via selecting “Plant Management section”, allowing the administrator to create new plants, search for existing plants, or delete plants. The administrator can create a new plant by clicking the “Create New Plant button” and the system will lead the administrator to the “Create New Plant Screen”. Furthermore, the administrator can edit plant details, which transitions them to the “Update Plant Screen”. In addition, the administrator can select the “Order Management” or “Delivery Management” sections to navigate to respective screens for managing orders or assigning deliveries.

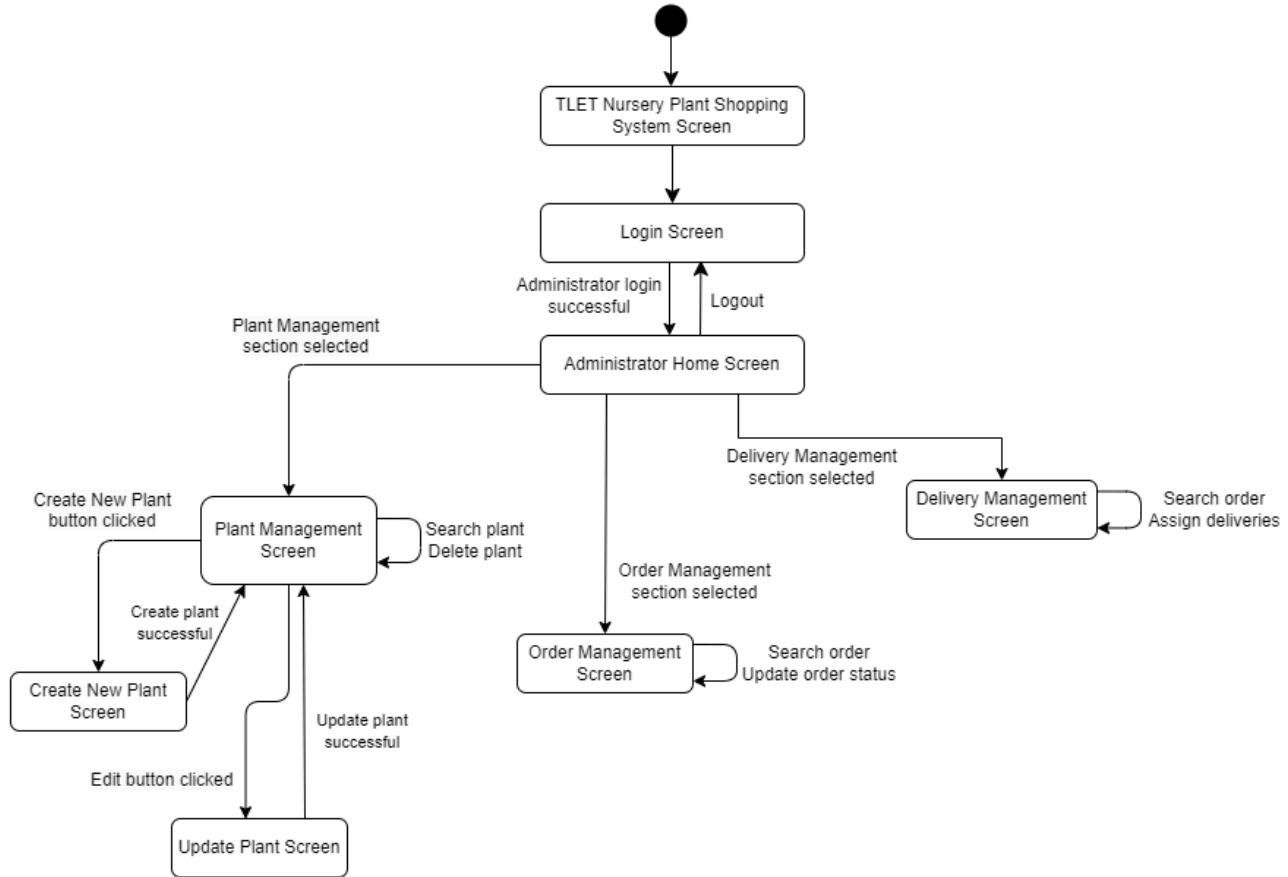


Figure 3.3: State Diagram for Administrator

3.2.2 Guest State Diagram

Figure 3.4 showcases the State Diagram for a guest in the Nursery Plant Shopping System. Beginning with the “TLET Nursery Plant Shopping System Screen”, a guest can either choose to register by selecting the “Signup section” or login by selecting the “Login section”. If the signup option is chosen, the guest is directed to the Registration Form Screen to complete their signup process. Upon successful registration, the guest is taken to the “Login Screen”. Conversely, if the login option is selected, the guest is taken to the “Login Screen” to enter their credentials. After entering the credentials, the system routes them to either the “Customer Home Screen” or the “Delivery Man Home Screen”, depending on their role within the system.

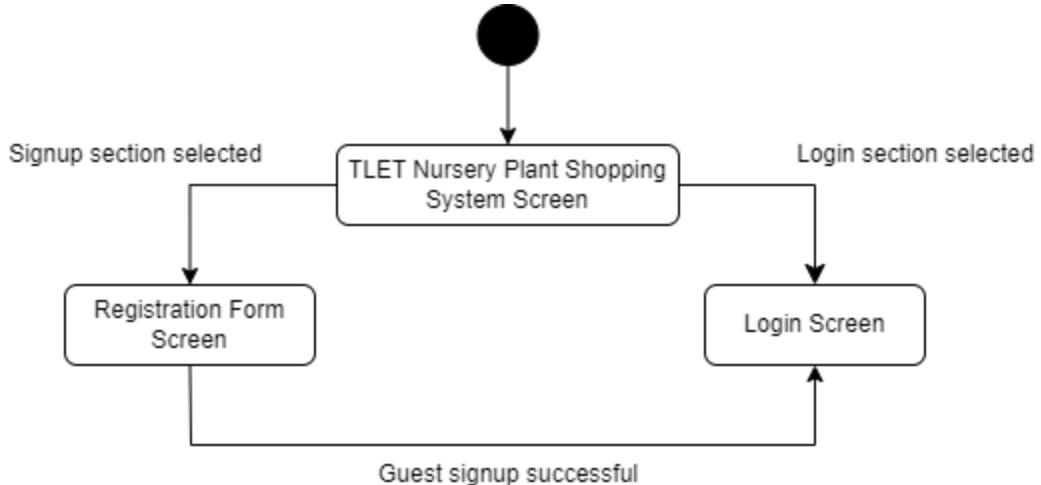


Figure 3.4: State Diagram for Guest

3.2.3 Customer State Diagram

Figure 3.5 illustrates the State 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”.

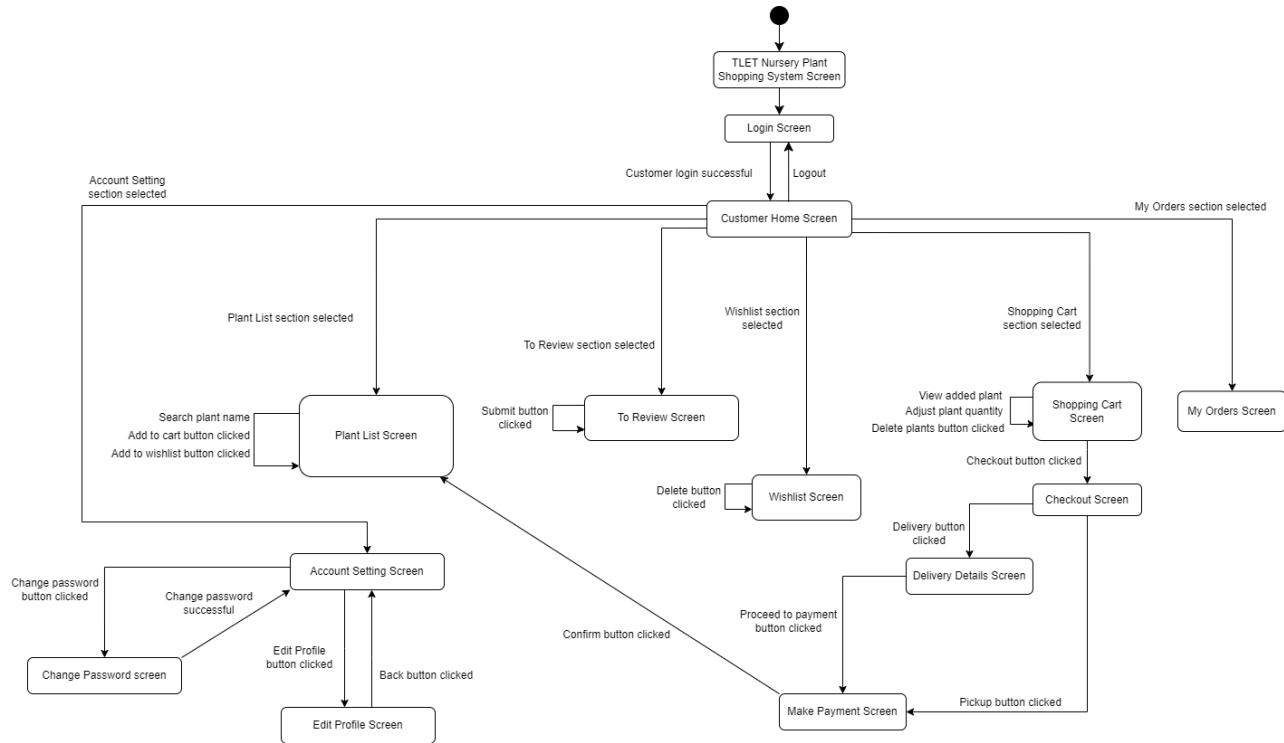


Figure 3.5: State Transition Diagram for Customer

3.2.4 Delivery Man State Diagram

Figure 3.6 presents the "State Diagram for Delivery Man" in the TLET Nursery Plant Shopping System. It begins at the "TLET Nursery Plant Shopping System Screen" and transitions to the "Login Screen", where upon successful login, the Delivery Man reaches the "Delivery Man Home Screen". After a successful login, the delivery man can navigate through various system states. Selecting the "Pending Orders section" takes the delivery man to a "Pending Order screen" where they can accept or reject orders. They can also navigate to the "Accepted Deliveries screen" to view the orders that are accepted by clicking the "Accepted Deliveries section". After successfully delivering the order, the Delivery Man chooses the respective delivery order and clicks the "Confirm Delivery" button at the "Accepted Deliveries screen". The delivery man can also access "Account Settings" and be directed to the "Account Settings Screen" for personal information and security settings, including password changes. Successful password updates return them to the "Account Settings Screen".

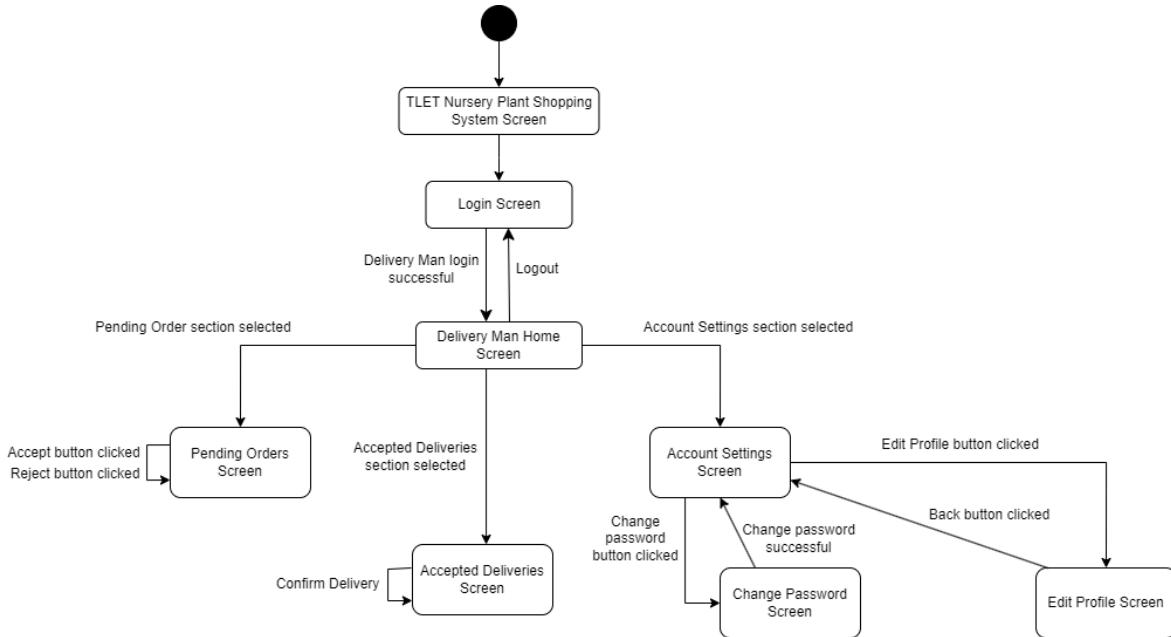


Figure 3.6: State Diagram for Delivery Man

3.3 Data Flow Diagrams

Figure 3.7 illustrates the Data Flow Diagram for a comprehensive Nursery Plant Shopping System that encompasses interactions across various roles including Administrators, Customers, Guests, and Delivery man, each interacting with several dedicated databases.

For Customers, the journey begins with the creation of a user account where they submit details such as email, password, name, address, state, IC, and phone number, which are then stored in the User Database and Customer Database. Once registered, they can proceed to edit their profiles and change passwords as needed. Shopping functionalities are user-friendly, in which they enable the Customers to add plants to a cart, manage a wishlist, or view their shopping cart, with all actions interfacing with the respective Cart and Wishlist Databases. Upon selection, payments are processed through interactions with the Payment Database and after buying, Customers are encouraged to share reviews and ratings, which will be stored in the Review Database.

Guests, who are not registered within the system, are not left out. They can browse the Plant List, which pulls data from the Plant Database, allowing them to search, view available plants, and check reviews from other customers. However, for purchase and further interactions, account creation is necessary.

On the other hand, the Administrator is in charge of the management of the Plant Database. They input plant data, including ID, name, description, image, price, and availability, and have the authority to create, update, or delete plant information, ensuring that the database remains up-to-date with the latest inventory details. They are also responsible for managing the Order Database, which includes viewing order details and updating their status. Additionally, the

Administrator holds the responsibility of assigning delivery orders to specific delivery men within the nursery plant shopping system.

On the logistics front, the system has a dedicated Delivery Man Database. Delivery men receive assignments where the Administrator links the delman ID to the appropriate order. Once an order is matched to a delivery man, it appears on their pending order page for efficient management. Delivery men can also view accepted deliveries and confirm completion, updating the order status to "Completed" in the Order Database, which marks the end of the delivery process.

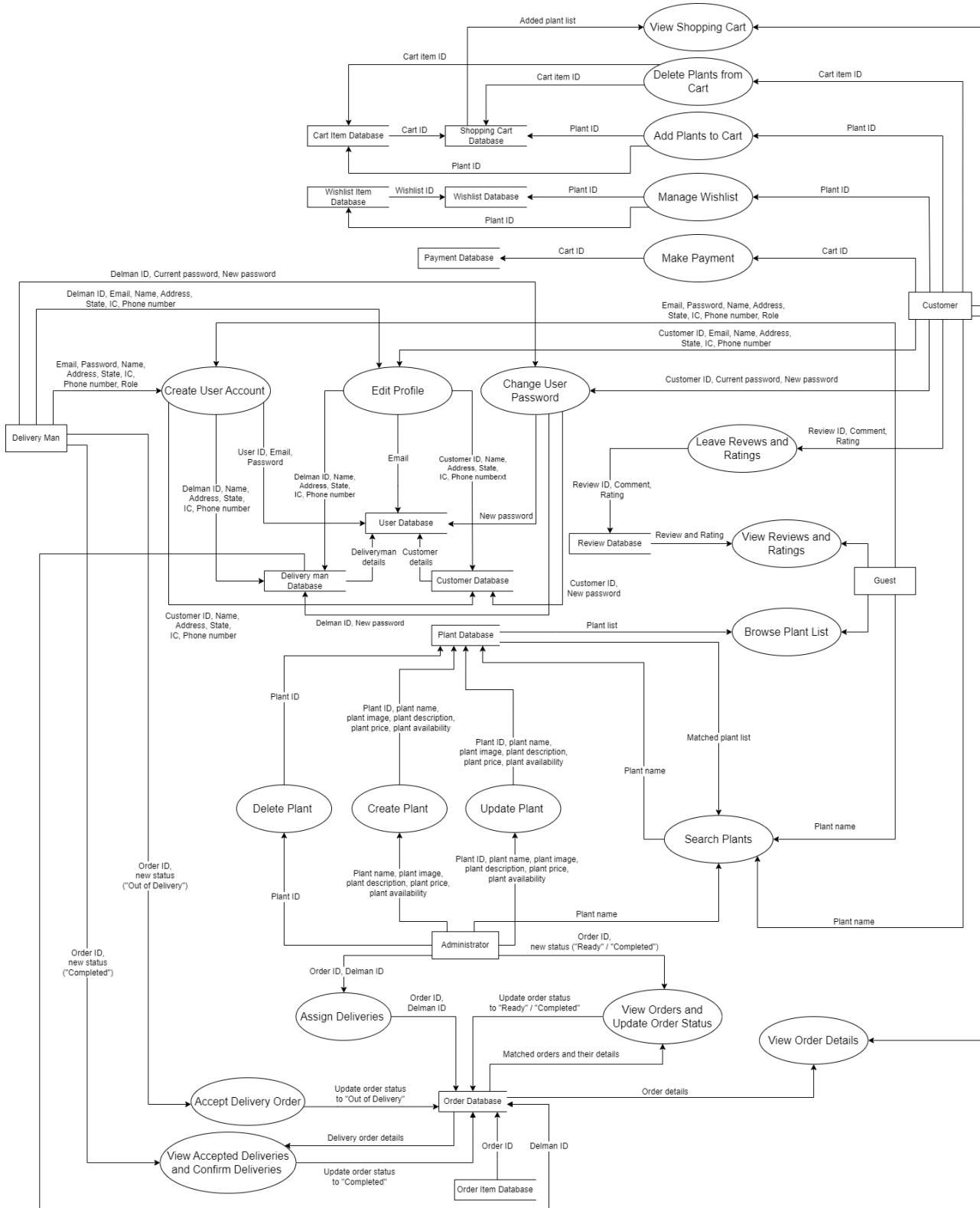


Figure 3.7: Data Flow Diagram for Nursery Plant Shopping System

4 Design

4.1 Data Dictionary

The User Table Data Dictionary, shown in Table 4.1, 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 4.1: 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 Administrator Table Data Dictionary shown as Table 4.2 provides a comprehensive overview of the attributes and specifications for managing administrator records within the database. It includes essential details such as the administrator's unique identifier (admin_ID) and their name (admin_Name). These attributes are essential for user authentication and management within the system. The data dictionary also identifies admin_ID as the primary key (PK), serving as a crucial reference point for this table's records.

Table 4.2: Administrator Table Data Dictionary

Table Name	Attribute Name	Contents	Type	Format	Range	Required	PK/FK	FK Referenced Table
Administrator	admin_ID	administrator ID	int	90000	90000-90005	Y	PK	-
	admin_Name	administrator name	varchar(80)	Ee Lin	-	Y	-	-

The Customer Table Data Dictionary shown as Table 4.3 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 4.3: 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	varchar(30)	040202016450	-	Y	-	-

		identification card number						
	customer_phone_number	customer phone number	bigInt	60128457865	-	Y	-	-

The DeliveryMan Table Data Dictionary shown as Table 4.4 serves as a comprehensive reference guide for managing delivery man records within the database. It contains attributes such as the delivery man's unique identifier (delman_ID), their name (delman_Name), address (delman_Address), state (delman_State), identification card number (delman_IC), and phone number (delman_phone_number). These attributes are crucial for tracking and managing delivery personnel within the system. The data dictionary also designates delman_ID as the primary key (PK), facilitating efficient record referencing within the table.

Table 4.4: Delivery Man Table Data Dictionary

Table Name	Attribute Name	Contents	Type	Format	Range	Required	PK/FK	FK Referenced Table
DeliveryMan	delman_ID	delivery man ID	int	90006	90006-90999	Y	PK	-
	delman_Name	delivery man name	varchar(80)	Siti Fatimah	-	Y	-	-
	delman_Address	delivery man address	varchar(200)	35, Jalan BPU6 Taman Indah, Puchong	-	Y	-	-
	delman_State	delivery man state	varchar(30)	Selangor	-	Y	-	-
	delman_IC	delivery man's identification	varchar(30)	920504018888	-	Y	-	-

		card number						
	delman_phone_number	delivery man phone number	bigInt	60147894163	-	Y	-	-

The Plant Table Data Dictionary shown as Table 4.5 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 4.5: 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	-	Y	-	-

				margin of the leaf is serrated with small teeth.				
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 4.6 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 4.6: 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 4.7 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 4.7: 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 4.8 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 4.8: 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 4.9 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 4.9: 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 4.10 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 4.10: 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 4.11 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 4.11: 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 4.12 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 4.12: 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 4.13 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 4.13: 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	-	-

4.2 Data Structures

4.2.1 User Array

Table 4.14 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 deliveryman, 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 4.14: 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

4.2.2 Administrator Array

Table 4.15 below shows Administrator Array Data Structures. It serves as a fundamental element for managing administrator information within a system. The Administrator ID, represented as an integer, serves as a primary unique identifier for each administrator, facilitating precise user identification and access control. The Administrator Name, stored as a varchar(80), allows for the storage of administrator names, ensuring user-friendly recognition and presentation.

Table 4.15: Administrator Array Data Structures

Attribute	Data Type
Administrator ID	int
Administrator Name	varchar(80)

4.2.3 Customer Array

Table 4.16 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 4.16: 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

4.2.4 DeliveryMan Array

Table 4.17 presents the Delivery Man Array Data Structures. The Delivery Man array is used for efficient management of essential data regarding delivery personnel within the system. The Delivery Man ID, represented as an integer, assigns a distinctive identifier to each delivery man. Delivery Man Name, as a varchar(80), facilitates easy user identification and improves the presentation of their profiles. Moreover, Delivery Man Address and Delivery Man State, both varchars (200) and (30) respectively, collect contact details and location information, enriching the understanding of delivery personnel's operational context. The Delivery Man's Identification Card Number, a varchar(30), stores unique identification data crucial for comprehensive identity verification. Additionally, the use of the bigint data type for the Delivery Man Phone Number accommodates the storage of contact information, ensuring efficient communication within the delivery team.

Table 4.17: Delivery Man Array Data Structures

Attribute	Data Type
Delivery man ID	int
Delivery man name	varchar(80)

Delivery man address	varchar(200)
Delivery man state	varchar(30)
Delivery man's identification card number	varchar(30)
Delivery man phone number	bigInt

4.2.5 Plant Array

Table 4.18 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 4.18: 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

4.2.6 ShoppingCart Array

Table 4.19 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 4.19: Shopping Cart Array Data Structures

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

4.2.7 CartItem Array

Table 4.20 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 4.20: 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)

4.2.8 Wishlist Array

Table 4.21 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 4.21: Wishlist Array Data Structures

Attribute	Data Type
Wishlist ID	int
Customer ID	int

4.2.9 WishlistItem Array

Table 4.22 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 4.22: Wishlist Array Data Structures

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

4.2.10 Payment Array

Table 4.23 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 4.23: 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)

4.2.11 Order Array

Table 4.24 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 4.24: Order Array Data Structures

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

4.2.12 OrderItem Array

Table 4.25 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 4.25: 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

4.2.13 Review Array

Table 4.26 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 4.26: Review Array Data Structures

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

4.3 Software Architecture

4.3.1 Three-Tier Architecture

Figure 4.1 below is a Three-Tier Architecture for a Nursery Plant Shopping System. Figure 4.1 shows a structured approach, which helps in dividing the system into three distinct layers: the Presentation layer (Front End), the Business Logic layer (Back End), and the Data layer. All the three layers communicate with a Database and Third-Party Services.

The client side is where the end-user interaction happens. In this case, it is through a web browser. The browser sends an HTTP request to the web server and receives an HTML response to display the web pages.

The web server consists of both the front end and back end. The front end is the presentation layer that the user directly interacts with. It includes various pages such as Dashboard, Plant Page, Order Page, Delivery Page, Shopping Cart, Account Setting, Review Page, Payment Page, and Wishlist Page. On the other hand, the back end is the business logic layer, where the application's core functionality resides. It manages plants, deliveries, orders, payment calculation, and reviews. The data layer is responsible for data management and includes functions such as create, search, update and delete data entries.

These layers interact with a centralized Database that stores all the relevant data. Besides, these layers integrate with Third Party Services for payment processing, including credit/debit card transactions, internet banking, and eWallet services. This three-tier architecture is important for ensuring that the Nursery Plant Shopping System can handle user interactions, process business logic, and manage data securely.

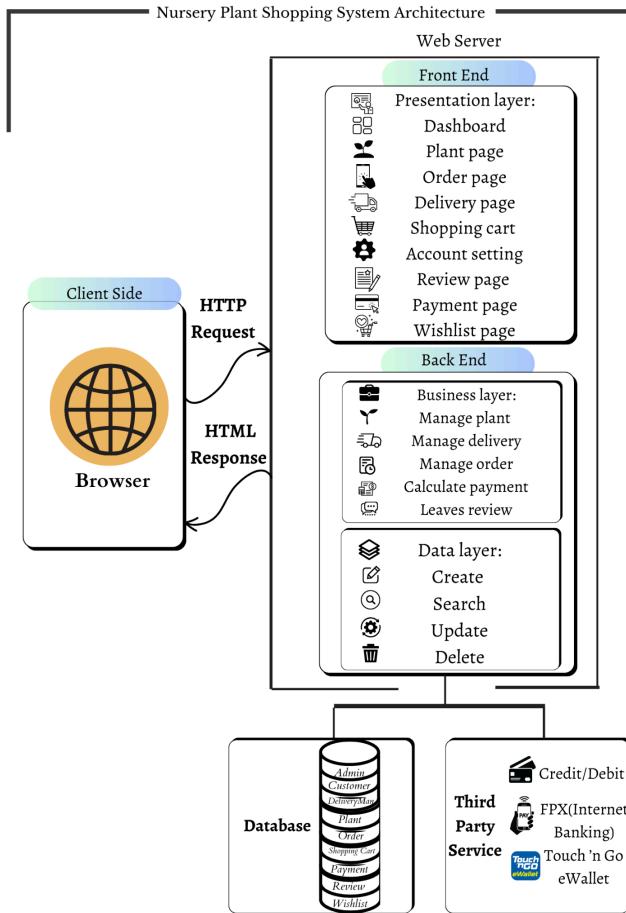


Figure 4.1: Three-Tier Architecture for Nursery Plant Shopping System

4.3.2 Component-based Architecture

Figure 4.2 below is the Component-based Architecture Diagram for the Nursery Plant Shopping System. The Component-based Architecture Diagram presents a modular approach to application design, categorizing functionalities into discrete components based on user roles. There are four user roles in this system: Administrator, Guest, Customer, and Delivery Man. Administrators have components for managing orders, deliveries, and the plants. Guests are provided with components to create accounts, browse plants and view plant reviews. Customers have a suite of components to manage their profiles, orders, and shopping activities, including wishlists, carts and leave reviews. Delivery personnel are equipped with components for handling delivery orders and managing their delivery tasks. Further details of each subsystem are explained in the following sections.

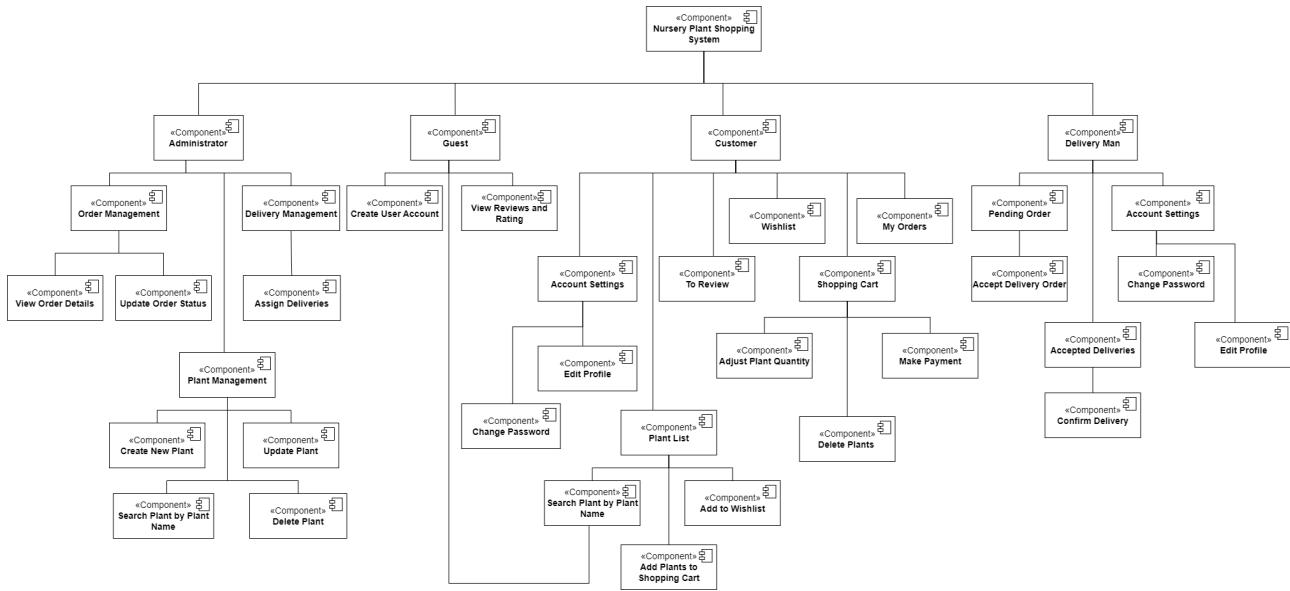


Figure 4.2: Component-based Architecture Diagram for Nursery Plant Shopping System

4.3.3 Administrator Subsystem

Figure 4.3 below is the Component-based Architecture Diagram for the Admin Subsystem. Figure 4.3 illustrates the specific functionalities available to the administrative role within the system. This includes the Order Management component, which allows the administrator to view the details of customer orders and update the order statuses as needed; the Delivery Management component, through which the administrator can assign deliveries to delivery men; and the Plant Management component, which enables the administrator to create new plants and update plants, as well as search for specific plants by name and remove plant from the plant list.

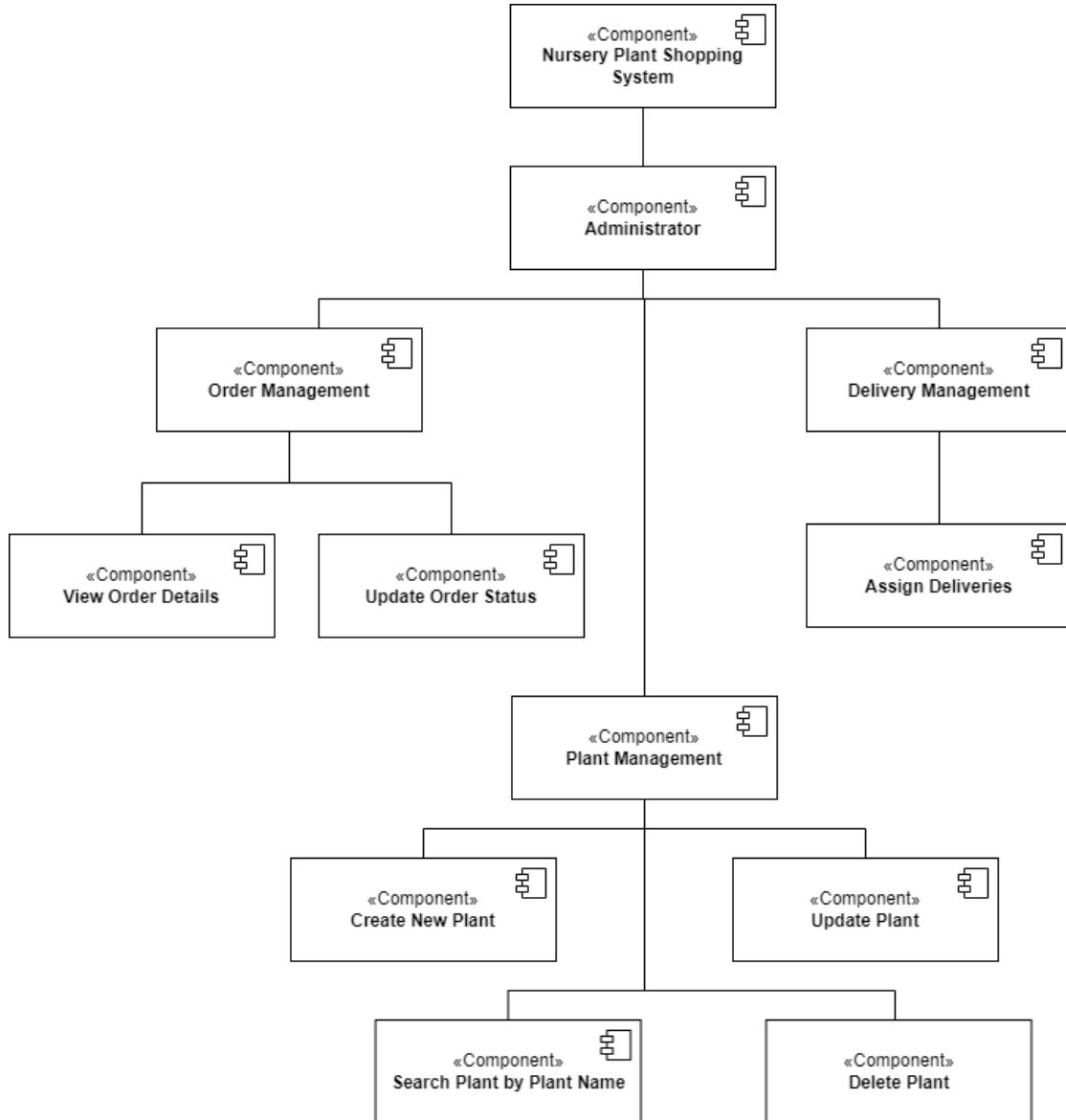


Figure 4.3: Component-based Architecture Diagram for Administrator Subsystem

4.3.4 Guest Subsystem

Figure 4.4 below is the Component-based Architecture Diagram for the Guest Subsystem. Figure 4.4 maps out the specific interactions and functionalities available to guests (users who are not logged in). Guests are allowed to create a new user account, in which they sign up to become a customer or a delivery man. Besides, guests can browse the plants and search a plant by plant name to easily find specific plants they are interested in. Guests can also view reviews and ratings to know the quality and popularity of the plants offered. This set of components ensures that even without an account, guests can engage with the system, assess its offerings, and initiate the registration process, providing a user-friendly and accessible platform for potential customers.

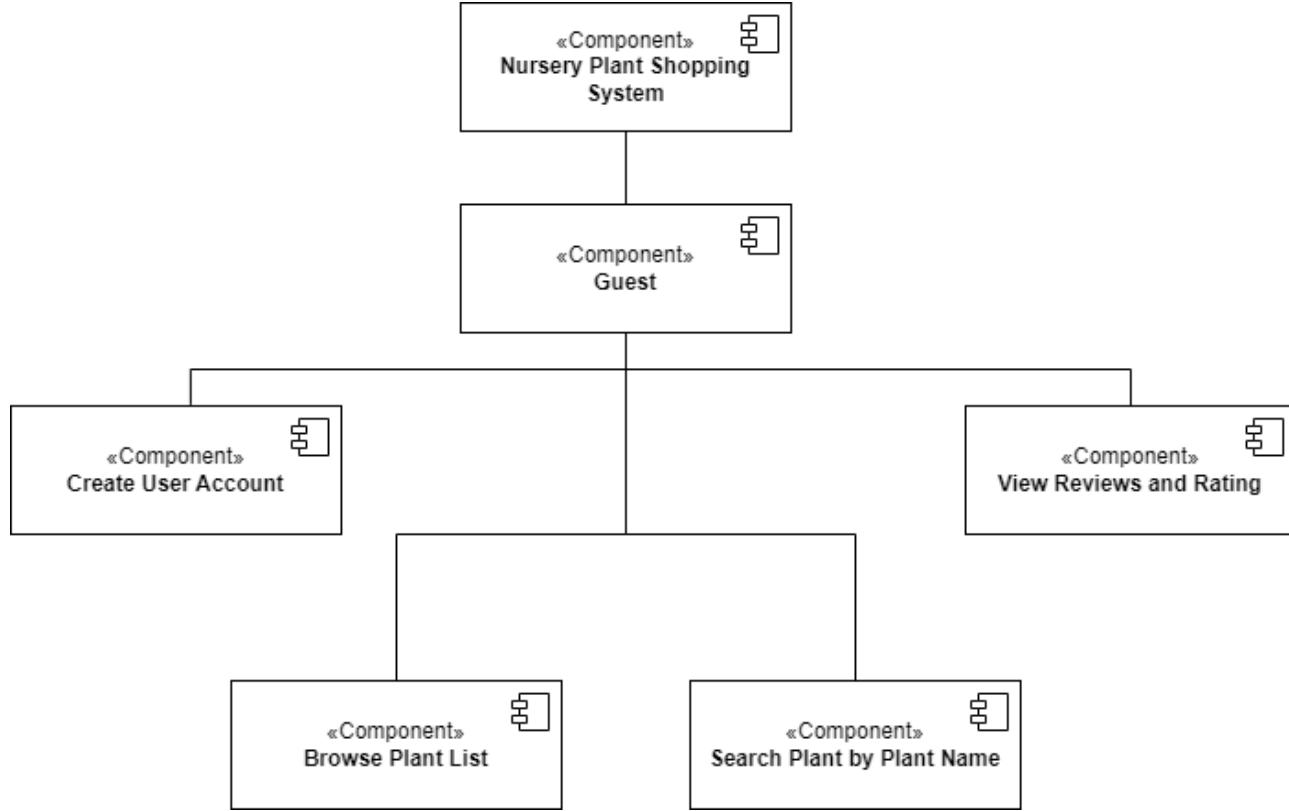


Figure 4.4: Component-based Architecture Diagram for Guest Subsystem

4.3.5 Customer Subsystem

Figure 4.5 below is the Component-based Architecture Diagram for the Customer Subsystem. Figure 4.5 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.

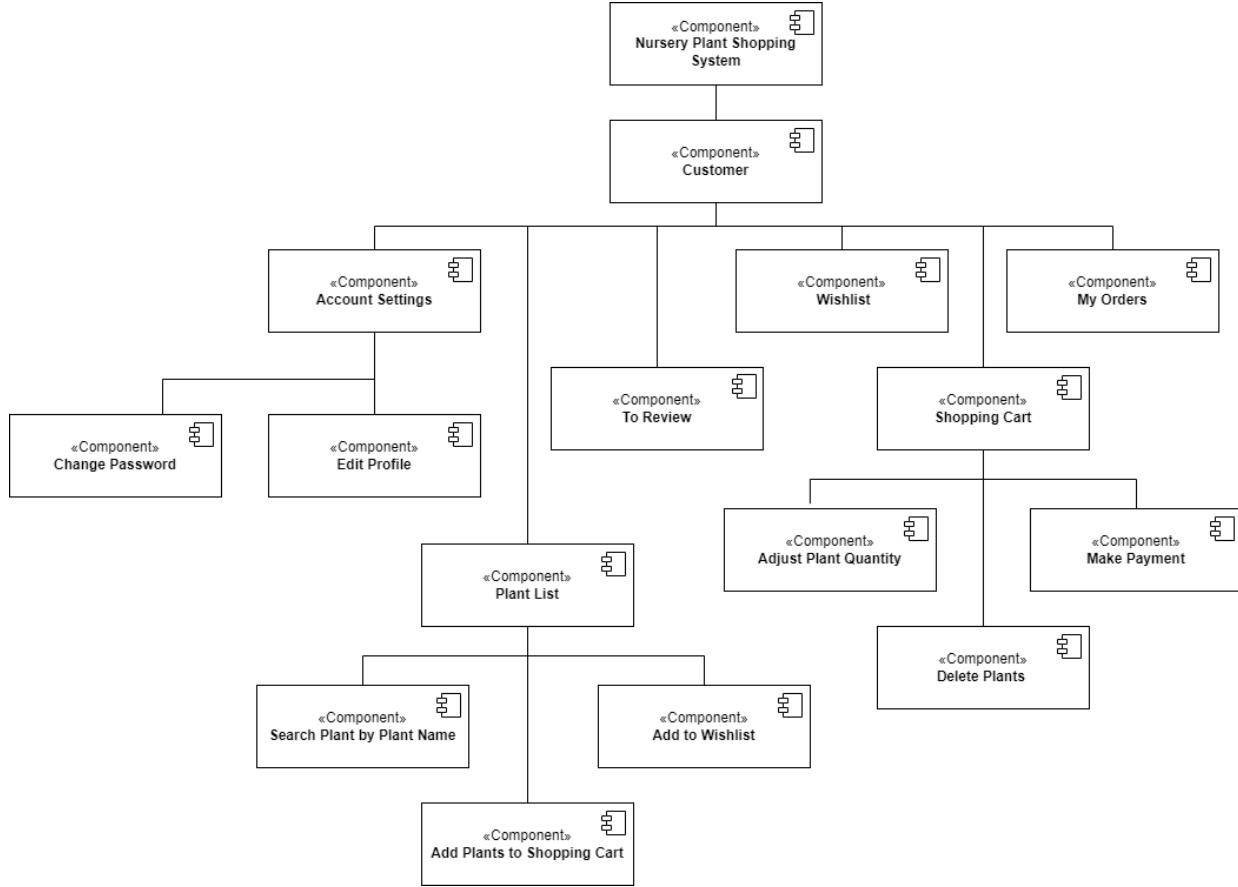


Figure 4.5: Component-based Architecture Diagram for Customer Subsystem

4.3.6 Delivery Man Subsystem

Figure 4.6 below is the Component-based Architecture Diagram for the Delivery Man Subsystem. Figure 4.6 illustrates the functional areas tailored for the delivery personnel. This subsystem allows a delivery man to manage their delivery operations effectively, with components for viewing “Pending Orders” and taking action to “Accept Delivery Order”. After an order has been accepted by the delivery man, that order will be transferred to the “Accepted Deliveries” list for active delivery tracking. The delivery man can “Confirm Delivery” once the plants reach the customers. Besides, personal account management is facilitated through “Account Settings”, which includes options to “Change Password” and “Edit Profile”, ensuring that delivery men can also keep their personal account information up-to-date. This targeted subsystem is used to streamline the delivery process and provide delivery men with the necessary tools to perform their roles efficiently in the plant shopping system.

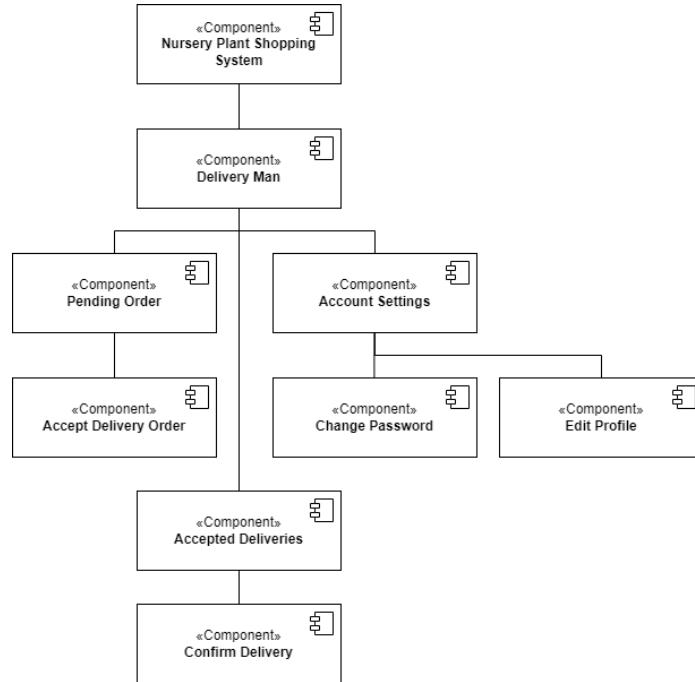


Figure 4.6: Component-based Architecture Diagram for Delivery Man Subsystem

4.4 Main Screens

Figure 4.7 displays the main screen of the "TLET Nursery Plant Shopping System". It includes options for users to sign up or log in, a search bar, and a featured plant listing with its ID, name, image, description, price, and availability. Below the plant information, there's a review button for users to click and scroll through customer ratings and comments.

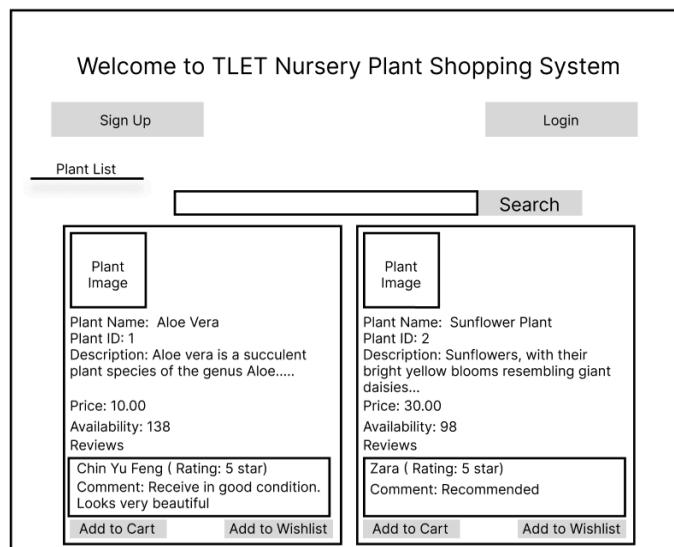


Figure 4.7: TLET Nursery Plant Shopping System Main Screen

4.4.1 Nursery Plant Shopping System Login Page

Figure 4.8 shows the login page of the TLET Nursery Plant Shopping System, created to be easy to use for administrators, customers, and delivery man. To log in, users just need to enter their email and password and click the 'Submit' button. New users can quickly get started by clicking the "Register here" link, which takes them to a Register page as shown in Figure 4.9. The Register page provides fields for users to fill in their personal details such as full name, password, email, address, state, IC No, phone number, and role to create an account. After completing the registration form, users can click on the 'Submit' button to create an account. If the user already has an account, they can click on the 'Login here' which will direct them to the 'Login page' shown in Figure 4.8.

The screenshot shows a 'Login' form. At the top, it says 'Login' and asks the user to 'Please fill in your email and password.' Below this are two input fields: 'Email Address' and 'Password'. A large 'Submit' button is centered below the password field. At the bottom of the form, there is a link 'Don't have an account? [Register here](#)'.

Figure 4.8: TLET Nursery Plant Shopping System Login Screen

The screenshot shows a 'Register' form. It starts with a note 'Please fill this form to create an account.' followed by several input fields: 'Full Name', 'Password', 'Email', 'Address', 'State', 'IC Number', 'Phone Number', and 'Role'. The 'Role' field is a dropdown menu. A large 'Register' button is at the bottom. At the very bottom, there is a link 'Already have an account? [Login here](#)'.

Figure 4.9: TLET Nursery Plant Shopping System Register Screen

4.4.2 Administrator Subsystem Screens

4.4.2.1 Administrator Home Screen

Upon logging in, the administrator is greeted by the home screen as seen in Figure 4.10. The “Administrator Home Screen” features a welcoming message at the center of the screen, saying “Welcome to Admin Dashboard”, so administrator knows he/she is in the right place. There is a navigation bar at the top of the “Administrator Home Screen” that neatly organizes essential management functions into tabs. These tabs include “Plant Management” for managing the plants, “Order Management” for tracking and updating the status of customer orders, “Delivery Management” for assigning delivery orders to delivery men, and “Logout” for administrator to safely exit the system and log out of his/her accounts.

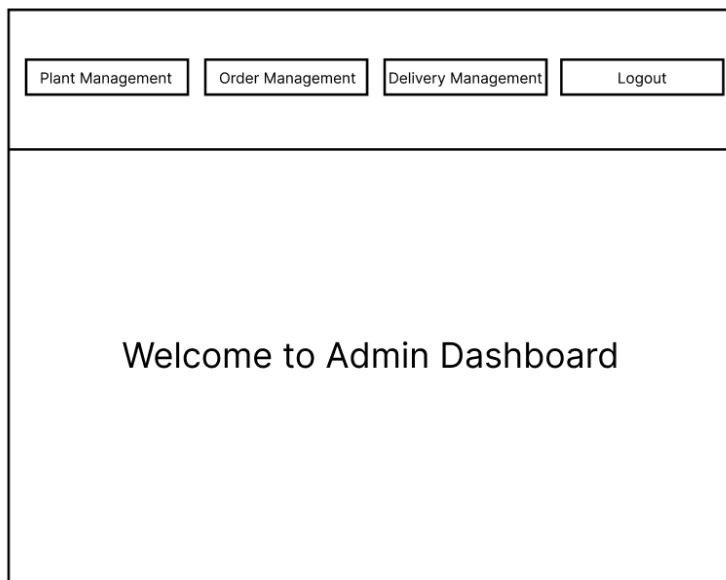


Figure 4.10: Administrator Home Screen

4.4.2.2 Administrator Creates a Plant Screen

Figure 4.11 shows the “Plant Management Screen” which shows the existing plant list when “Plant Management” is selected. Upon clicking “Create New Plant”, the administrator is taken to the “Create New Plant Screen” shown in Figure 4.12 to input the plant's name, description, quantity, and price, and to upload an image to create a new plant to add to the plant list. The system then verifies the new entry's uniqueness, assigns an ID, and confirms its addition to the list.

The screenshot shows a web-based application interface. At the top, there is a navigation bar with four buttons: "Plant Management", "Order Management", "Delivery Management", and "Logout". Below the navigation bar, the title "Plant Management" is displayed. A search bar contains the placeholder "Search Plant by Plant Name" and a "Search" button. Below the search bar is a "Create New Plant" button. The main content area displays a single plant entry in a card format. The card includes a "Plant Image" placeholder, the "Plant ID: 10000", "Plant Name: Sunburst Alocasia", "Description: Tropical plant with arrow-shaped leaves, vibrant green hues, and striking yellow veins.", "Price: 60.00", and "Availability: 100". Below the card are two buttons: "Edit" and "Delete". A "Reviews" section follows, showing a single review: "5 star by Chin Yu Feng" and the comment "Receive in good condition. Looks very beautiful".

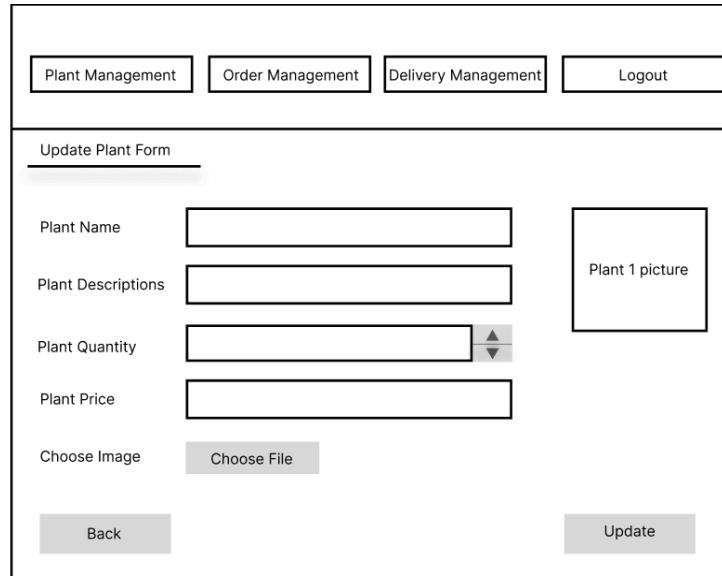
Figure 4.11: Plant Management Screen

The screenshot shows a "Create New Plant Form". At the top, there is a navigation bar with four buttons: "Plant Management", "Order Management", "Delivery Management", and "Logout". Below the navigation bar, the title "Create New Plant Form" is displayed. The form consists of several input fields: "Enter Plant Name" (text input), "Enter Plant Descriptions" (text input), "Enter Plant Quantity" (text input with up/down arrows), "Enter Plant Price" (text input), "Choose Image" (button), and "Choose File" (button). At the bottom of the form are two buttons: "Back" and "Confirm".

Figure 4.12: Create New Plant Screen

4.4.2.3 Administrator Updates a Plant Screen

Figure 4.13 shows the interface for updating a plant in the TLET Nursery system. This process begins at the “Administrator Home Screen”, shown in Figure 4.10, where “Plant Management” is selected, leading to the “Plant Management Screen” displayed in Figure 4.11, which shows the existing plant list. Clicking “Edit” on a plant takes the admin to the “Update Plant Screen” of Figure 4.13, where they can modify the plant's name, description, quantity, price, and image. After changes are made, hitting “Update” saves them and a confirmation is shown. If no edits are needed, clicking “Back” returns the admin to the “Plant Management Screen” seen in Figure 4.11.

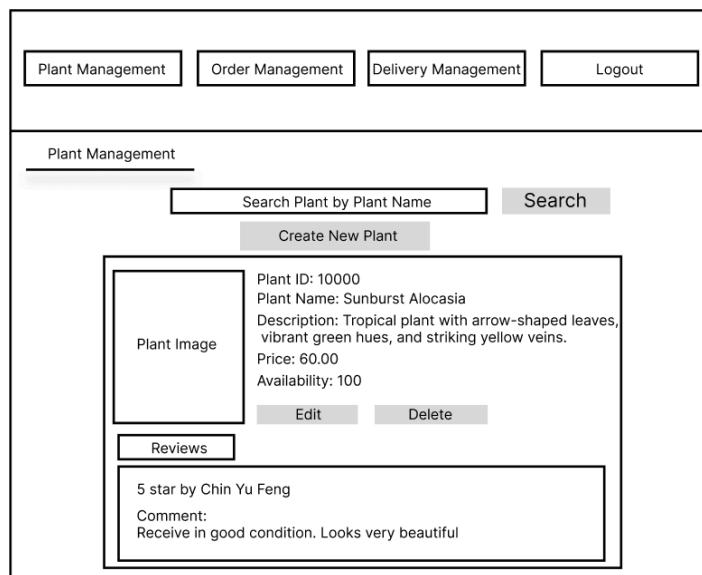


The Update Plant Screen is a form for modifying plant details. It features a header with 'Plant Management', 'Order Management', 'Delivery Management', and 'Logout' buttons. Below the header is a title 'Update Plant Form'. The form includes fields for 'Plant Name' (text input), 'Plant Descriptions' (text input), 'Plant Quantity' (text input with up/down arrows), 'Plant Price' (text input), and 'Choose Image' (button). To the right of these fields is a placeholder box labeled 'Plant 1 picture'. At the bottom are 'Back' and 'Update' buttons.

Figure 4.13: Update Plant Screen

4.4.2.4 Administrator Deletes a Plant Screen

Figure 4.14 shows the “Delete Plant Screen” where an administrator can remove a plant from the TLET Nursery Plant Shopping System. This process begins at the “Administrator Home Screen”, shown in Figure 4.10, where “Plant Management” is selected, leading to the “Plant Management Screen” displayed in Figure 4.14 (same as Figure 4.11), where the administrator is presented with the existing plant list. To delete a plant, the administrator clicks the “Delete” button associated with the plant, which triggers a confirmation prompt. After confirming the deletion, the plant is removed from the system, the plant list is refreshed, and a confirmation message is displayed to indicate the successful removal of the plant.



The Delete Plant Screen displays a list of plants. At the top are 'Plant Management', 'Order Management', 'Delivery Management', and 'Logout' buttons. Below is a search bar with 'Search Plant by Plant Name' and a 'Search' button. A 'Create New Plant' button is also present. The main area shows a plant entry for 'Sunburst Alocasia' with details: Plant ID: 10000, Plant Name: Sunburst Alocasia, Description: Tropical plant with arrow-shaped leaves, vibrant green hues, and striking yellow veins, Price: 60.00, Availability: 100. Below the details are 'Edit' and 'Delete' buttons. A 'Reviews' section shows a single review: 5 star by Chin Yu Feng with the comment: Receive in good condition. Looks very beautiful.

Figure 4.14: Delete Plant Screen

4.4.2.5 Administrator Searches Plant by Plant Name Screen

Figure 4.15 illustrates the "Search Plant by Plant Name Screen" of the TLET Nursery Plant Shopping System. This is where an administrator can look for specific plants within the system. The process begins at the "Administrator Home Screen" referenced in Figure 4.10 by selecting the "Plant Management" option. This action takes the user to the "Plant Management Screen" shown in Figure 4.15 (same as Figure 4.11), which lists all the plants currently in the list. The administrator can enter the name of a plant into the search field and press the "Search" button. The system will then find and display any plants that match the entered search terms.

Figure 4.15: Searches Plant by Plant Name Screen

4.4.2.6 Administrator Views Order Details and Updates Order Status Screen

Figure 4.16 shows "Views Order Details and Updates Order Status Screen" where an administrator can view and manage order statuses within the TLET Nursery Plant Shopping System. The process begins at the "Administrator Home Screen" referenced in Figure 4.10 by selecting the "Order Management" option. This action takes the user to the "Order Management Screen" shown in Figure 4.16, where the system displays a list of ongoing orders with detailed information for each ongoing order. The administrator can update an order's status from the status drop-down menu to "Ready" when it's prepared for pickup or delivery, or to "Completed" after the order is picked up. The update is confirmed by clicking the "Update Status" button, which prompts the system to refresh the order list with the new status. An orderID search feature is also available, enabling the administrator to find specific orders and update their statuses efficiently.

The screenshot shows the 'Order Management' screen. At the top, there are four buttons: 'Plant Management', 'Order Management' (which is highlighted in blue), 'Delivery Management', and 'Logout'. Below the buttons, the title 'Order Management' is displayed. A search bar with the placeholder 'Search by OrderID' and a 'Search' button are present. The main content area displays order details for OrderID 60000. The details are as follows:

OrderID: 60000	CustomerID: 1
Full Name: Chin Yu Feng	Phone Number: 0128457865
Email: yufeng@gmail.com	Address: 20, Taman Maju, Section 3/2a, Cheras
State: Selangor	Order Date: 2/1/2024
Total Plants: 2	Delivery Method: Delivery
Total Price: 120.00	

At the bottom of the content area, there are two buttons: 'Waiting' and 'Update Status'.

Figure 4.16: Views Order Details and Updates Order Status Screen

4.4.2.7 Administrator Assigns Deliveries to Delivery Man Screen

Figure 4.17 shows the “Assigns Deliveries to Delivery Man Screen” where an administrator can assign delivery tasks to the delivery man in the TLET Nursery Plant Shopping System. The process begins at the “Administrator Home Screen” referenced in Figure 4.10 by selecting the “Delivery Management” option. This action takes the user to the “Delivery Management Screen” shown in Figure 4.17, where the system displays a list of ongoing orders scheduled for delivery, along with their details. The administrator can then allocate each delivery to a delivery man from a drop-down menu. The administrator clicks on the “Assign” button to assign the selected delivery to the delivery man. The system updates the “Assign To” status and returns the updated list of ongoing orders scheduled for delivery. Additionally, the administrator can use the OrderID search function to locate specific orders, simplifying the assignment process.

The screenshot shows the 'Delivery Management' screen. At the top, there are four buttons: 'Plant Management', 'Order Management', 'Delivery Management' (which is highlighted in blue), and 'Logout'. Below the buttons, the title 'Delivery Management' is displayed. A search bar with the placeholder 'Search by OrderID' and a 'Search' button are present. The main content area displays order details for OrderID 60000. The details are as follows:

OrderID: 60000	CustomerID: 1
Full Name: Chin Yu Feng	Phone Number: 0128457865
Email: yufeng@gmail.com	Address: 20, Taman Maju, Section 3/2a, Cheras
State: Selangor	Order Date: 2/1/2024
Total Plants: 2	Delivery Method: Delivery
Total Price: 120.00	

Below the details, there is a dropdown menu containing the name 'Siti Fatimah' and an 'Assign' button.

Figure 4.17: Assigns Deliveries to Delivery Man Screen

4.4.3 Guest Subsystem Screens

4.4.3.1 Guest Home Screen

Figure 4.18 shows 'Guest Home Screen' which is the "TLET Nursery Plant Shopping System Screen". It includes options for guests to sign up or log in, a search bar, and a featured plant listing with its ID, name, image, description, price, and availability. Below the plant information, there's a review button for guests to click and scroll through customer ratings and comments.

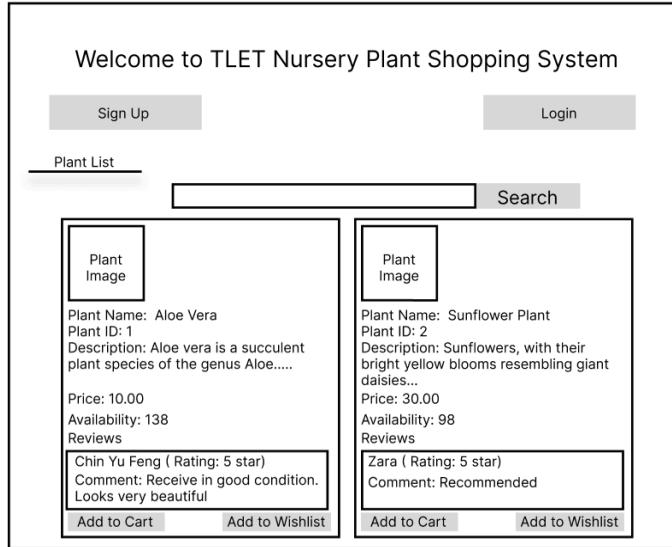


Figure 4.18: Guest Home Screen

4.4.3.2 Guest Searches a Plant by Plant Name Screen

Figure 4.19 displays the "Search Plant by Plant Name Screen" of the TLET Nursery Plant Shopping System, designed for guest users. On this screen, guests can search for plants by typing the name of the plant into the search bar and then clicking the 'Search' button. The system responds by providing detailed information about the plants that match the search criteria, including an image, description, price, availability, and review.

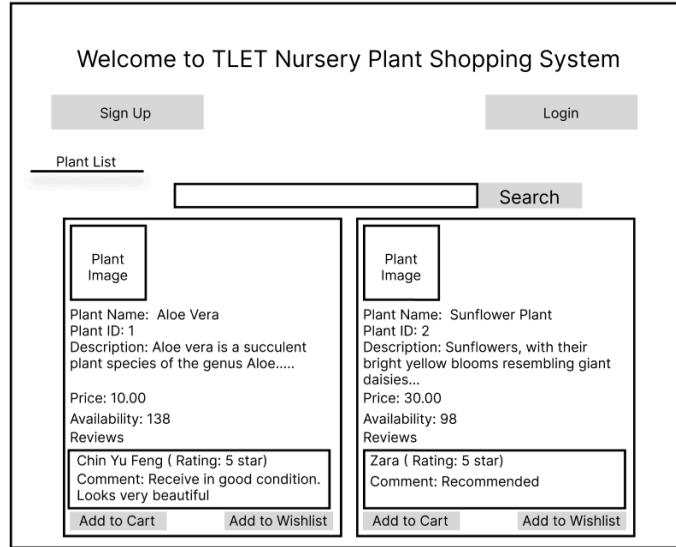
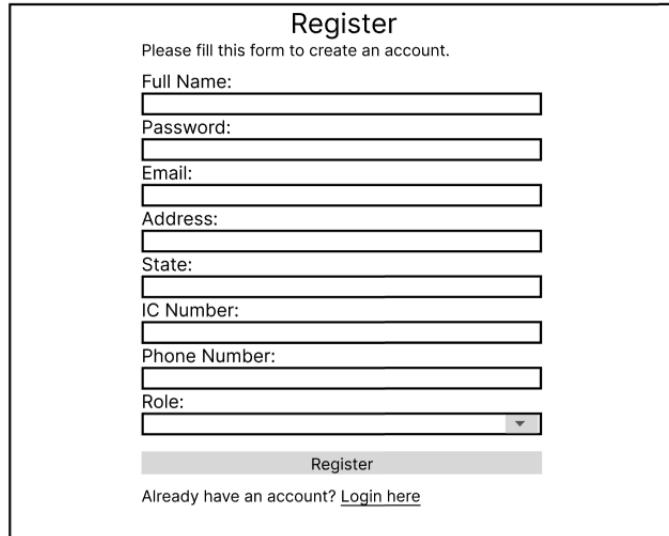


Figure 4.19: Searches Plant by Plant Name Screen

4.4.3.3 Guest Creates a User Account Screen

Figure 4.20 shows the 'Creates User Account Screen', which illustrates the process for guests to create a new account on the TLET Nursery Plant Shopping System. The process begins at the "Guest Home Screen" referenced in Figure 4.18 by clicking the 'Sign Up' button. This action takes the guest to the "Creates User Account Screen" shown in Figure 4.20 where guests fill out the form with personal and contact details. Upon clicking the "Submit" button, the system validates the input and saves the new user details to persistent storage. Successful registration is confirmed with a message. Customers or delivery men who have an account can quickly get started by clicking the "Login here" link, which takes them to a Login page in Figure 4.8.



The image shows a registration form titled 'Register'. It includes fields for Full Name, Password, Email, Address, State, IC Number, Phone Number, and Role. There is a 'Register' button and a link to 'Login here'.

Register	
Please fill this form to create an account.	
Full Name:	<input type="text"/>
Password:	<input type="password"/>
Email:	<input type="text"/>
Address:	<input type="text"/>
State:	<input type="text"/>
IC Number:	<input type="text"/>
Phone Number:	<input type="text"/>
Role:	<input type="text"/>
<input type="button" value="Register"/>	
Already have an account? Login here	

Figure 4.20: Create User Account Screen

4.4.3.4 Guest Views Reviews and Ratings Screen

Figure 4.21 shows 'Views Reviews and Ratings Screen' where guests can view customer feedback on plants in the TLET Nursery Plant Shopping System. Guests can browse or search through the list of plants, and upon interest in a particular plant, they can click on the 'Reviews' button. Once clicked, the reviews will pop up on the screen, allowing the customer to read through the feedback and ratings provided by previous buyers. This helps guests make informed decisions by considering the experiences and satisfaction of others who have purchased the plant.

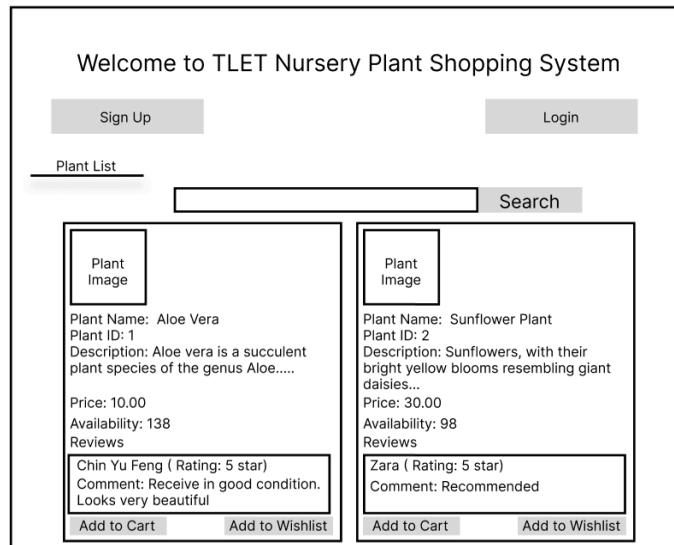


Figure 4.21: Views Reviews and Ratings Screen

4.4.4 Customer Subsystem Screens

4.4.4.1 Customer Home Screen

Figure 4.22 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.

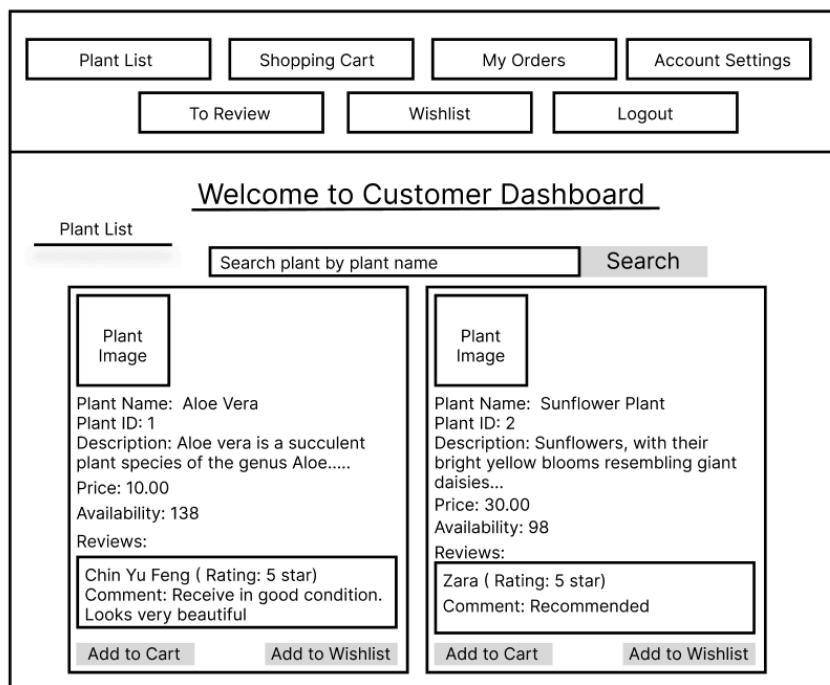


Figure 4.22: Customer Home Screen

4.4.4.2 Customer Changes Their Password Screen

Figure 4.24 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 4.22, where the customer selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 4.23, where the customer's details are listed. By selecting the 'Change Password' button, the customer is directed to the interface shown in Figure 4.24, where they are required to enter their current 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 4.23, completing the password change process.

The screenshot shows the 'Account Settings' screen. At the top, there is a navigation bar with buttons for 'Plant List', 'Shopping Cart', 'My Orders', 'Account Settings', 'To Review', 'Wishlist', and 'Logout'. Below the navigation bar, the title 'Account Settings' is underlined. A large rectangular area displays the customer's profile information: Full Name: Chin Yu Feng, Email: yufeng@gmail.com, Phone Number: 0128457865, Address: 20, Taman Maju, Section 3/2a, Cheras, State: Selangor, and IC No: XXX. At the bottom of this section are two buttons: 'Edit Profile' and 'Change Password'.

Figure 4.23: Account Settings Screen

The screenshot shows the 'Change Password' screen. At the top, there is a navigation bar with buttons for 'Plant List', 'Shopping Cart', 'My Orders', 'Account Settings', 'To Review', 'Wishlist', and 'Logout'. Below the navigation bar, the title 'Change Password' is underlined. The screen contains three input fields: 'Old Password' (empty), 'New Password' (empty), and 'New Password confirmation' (empty). At the bottom is a single button labeled 'Confirm'.

Figure 4.24: Change Password Screen

4.4.4.3 Customer Can Edit Profile Details Screen

Figure 4.25 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 4.22, where the customer selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 4.23, where the customer's details are listed. By selecting the 'Edit Profile' button, the customer is directed to the interface shown in Figure 4.25, 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 4.23 without making any updates.

The screenshot shows a user interface for editing a profile. At the top, there is a navigation bar with buttons for 'Plant List', 'Shopping Cart', 'My Orders', 'Account Settings', 'To Review', 'Wishlist', and 'Logout'. Below the navigation bar, the title 'Edit Profile' is displayed. The main area contains a form with the following fields:

- Full Name: Chin Yu Feng
- Email: yufeng@gmail.com
- Phone Number: 0128457865
- Address: 20, Taman Maju, Section 3/2a, Cheras
- State: Selangor
- IC No: XXX

At the bottom of the form are two buttons: 'Back' and 'Confirm'.

Figure 4.25: Edit Profile Details Screen

4.4.4.4 Customer Views Added Plants In The Shopping Cart Screen

Figure 4.26 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 4.22, 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 4.26, 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.

The screenshot shows a user interface for viewing the shopping cart. At the top, there is a navigation bar with buttons for 'Plant List', 'Shopping Cart', 'My Orders', 'Account Settings', 'To Review', 'Wishlist', and 'Logout'. Below the navigation bar, the title 'Shopping Cart' is displayed. The main area contains a list of plants currently in the cart:

	Plant Name: Aloe Vera
	Plant ID: 1
	Price: RM10.00
	Quantity: <input type="text" value="1"/> ▲ ▼
	Remove
	Total Price: RM10.00

At the bottom of the cart list, there is a 'Checkout' button.

Figure 4.26: Views Added Plants In The Shopping Cart Screen

4.4.4.5 Customer Adds Plants To The Shopping Cart Screen

Figure 4.27 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 4.22, where the customer selects the 'Plant List' button. This action takes them to the "Adds Plants To The Shopping Cart Screen" displayed in Figure 4.27 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 4.27 to continue browsing.

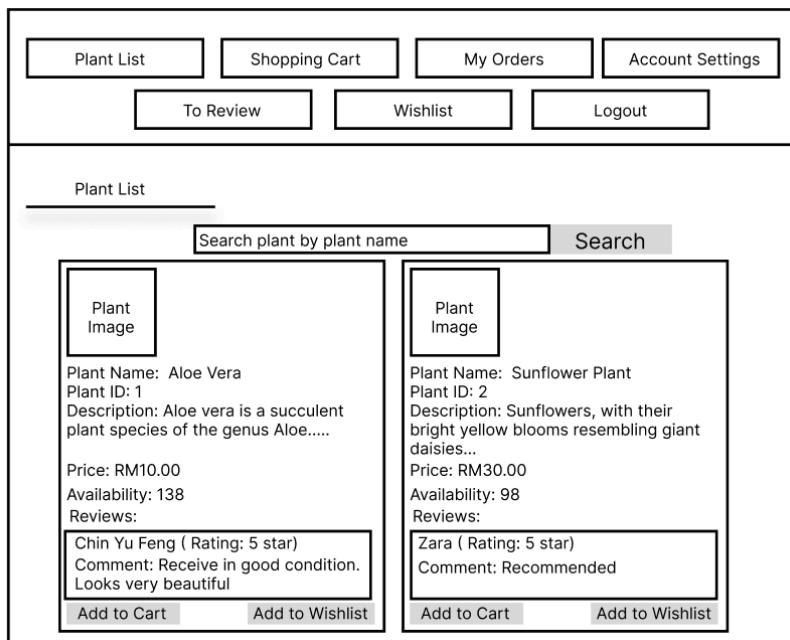


Figure 4.27: Adds Plants To The Shopping Cart Screen

4.4.4.6 Customer Deletes Plants From The Shopping Cart Screen

Figure 4.28 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 4.22, where the customer selects the 'Shopping Cart' button. This action takes them to the "Deletes Plants From The Shopping Cart Screen" displayed in Figure 4.28 (same as Figure 4.26), 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 4.28.

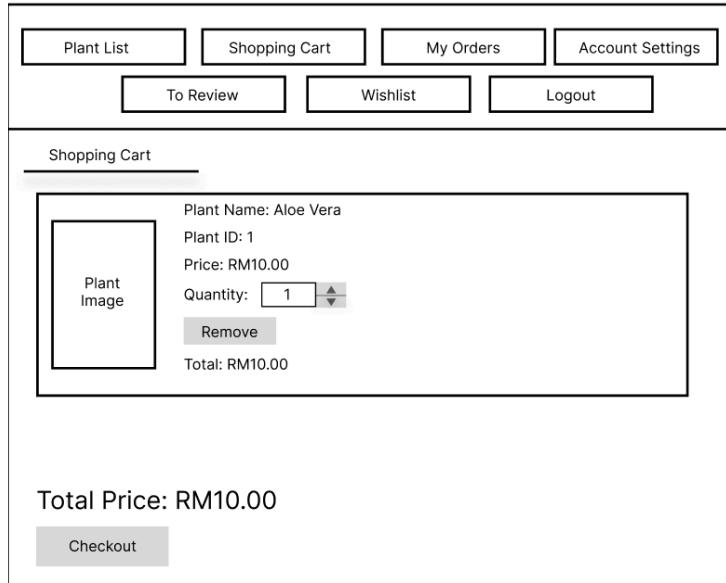


Figure 4.28: Deletes Plants From The Shopping Cart Screen

4.4.4.7 Customer Searches a Plant by Plant Name Screen

Figure 4.29 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 4.22, where the customer selects the 'Plant List' button. This action takes them to the "Searches a Plant by Plant Name Screen" displayed in Figure 4.29 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.

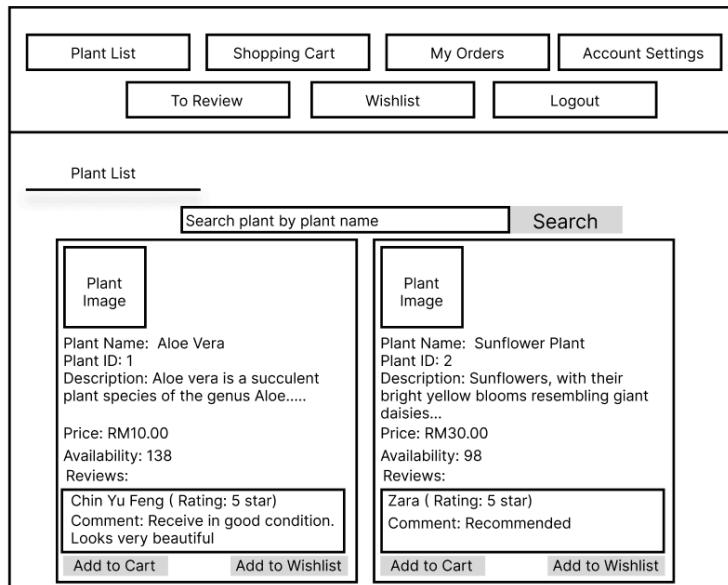


Figure 4.29: Searches a Plant by Plant Name Screen

4.4.4.8 Customer Makes a Payment Screen

Figure 4.32 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 4.22, 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 4.26, where they can choose to proceed by hitting the 'Checkout' button. This action navigates them to the 'Checkout Screen' in Figure 4.30, offering delivery and pickup options. If pickup is the chosen method, the system instructs to collect the order within 14 days at the specified location before directing them to the 'Make Payment' interface in Figure 4.32. If the delivery method is selected, the system takes customers to the 'Delivery Details Screen' (Figure 4.31) 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 4.32. 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 4.27, to continue shopping or review their orders.

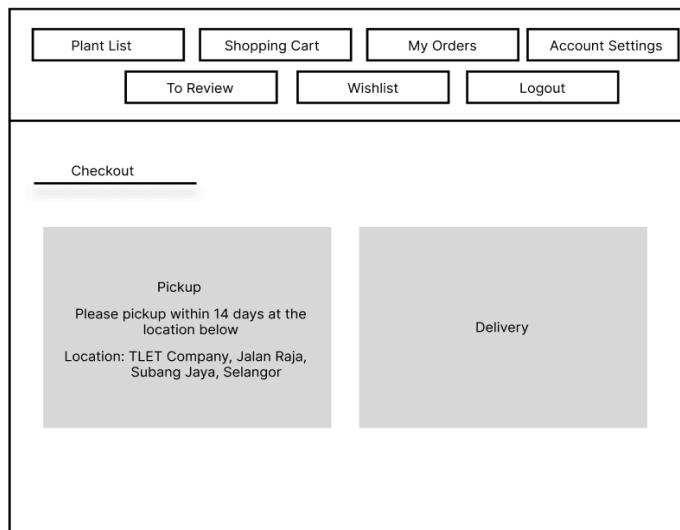
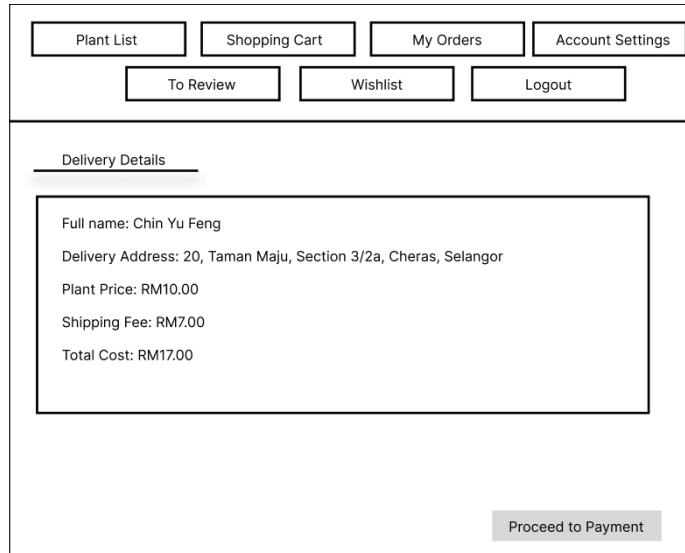


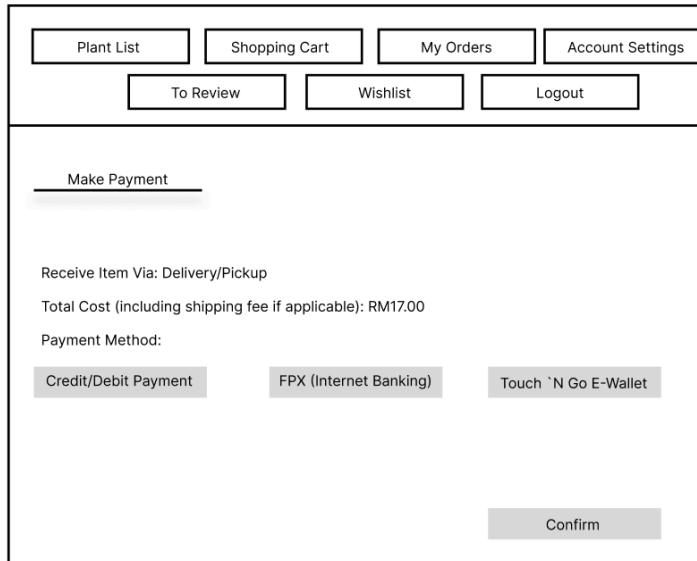
Figure 4.30: Checkout Screen



The screenshot shows a user interface for a plant shopping system. At the top, there is a navigation bar with buttons for 'Plant List', 'Shopping Cart', 'My Orders', 'Account Settings', 'To Review', 'Wishlist', and 'Logout'. Below the navigation bar, the title 'Delivery Details' is displayed. A large rectangular box contains the following information:
 Full name: Chin Yu Feng
 Delivery Address: 20, Taman Maju, Section 3/2a, Cheras, Selangor
 Plant Price: RM10.00
 Shipping Fee: RM7.00
 Total Cost: RM17.00

In the bottom right corner of the main content area, there is a button labeled 'Proceed to Payment'.

Figure 4.31: Delivery Details Screen



The screenshot shows a user interface for making a payment. At the top, there is a navigation bar with buttons for 'Plant List', 'Shopping Cart', 'My Orders', 'Account Settings', 'To Review', 'Wishlist', and 'Logout'. Below the navigation bar, the title 'Make Payment' is displayed. A large rectangular box contains the following information:
 Receive Item Via: Delivery/Pickup
 Total Cost (including shipping fee if applicable): RM17.00
 Payment Method:
 Credit/Debit Payment FPX (Internet Banking) Touch 'N Go E-Wallet

In the bottom right corner of the main content area, there is a button labeled 'Confirm'.

Figure 4.32: Make Payment Screen

4.4.4.9 Customer Views Order Details of Ongoing Orders Screen

Figure 4.33 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 4.22. By clicking the 'My Orders' button, customers are led to the screen shown in Figure 4.33, 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.

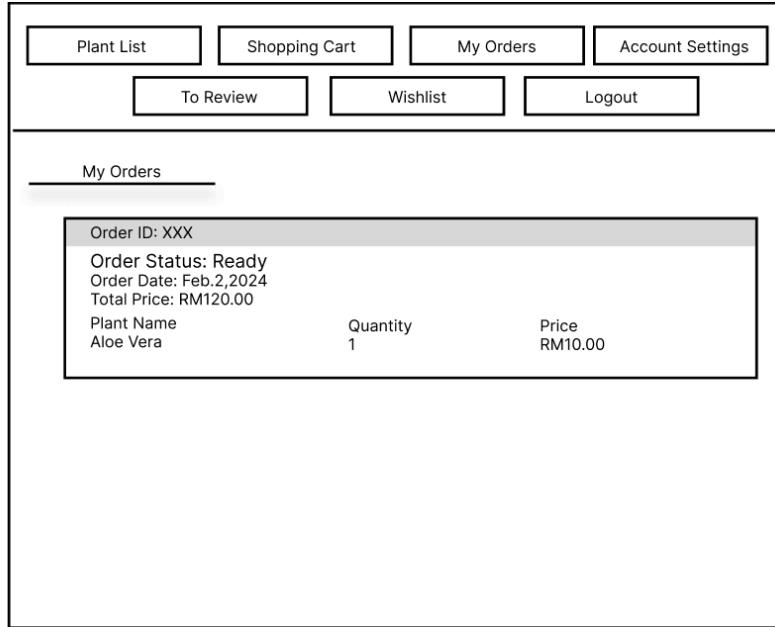


Figure 4.33: Views Order Details of Ongoing Orders Screen

4.4.4.10 Customer Leaves Reviews and Ratings Screen

Figure 4.34 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 4.22. By clicking the 'To Review' button, customers are led to the screen shown in Figure 4.34, 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.

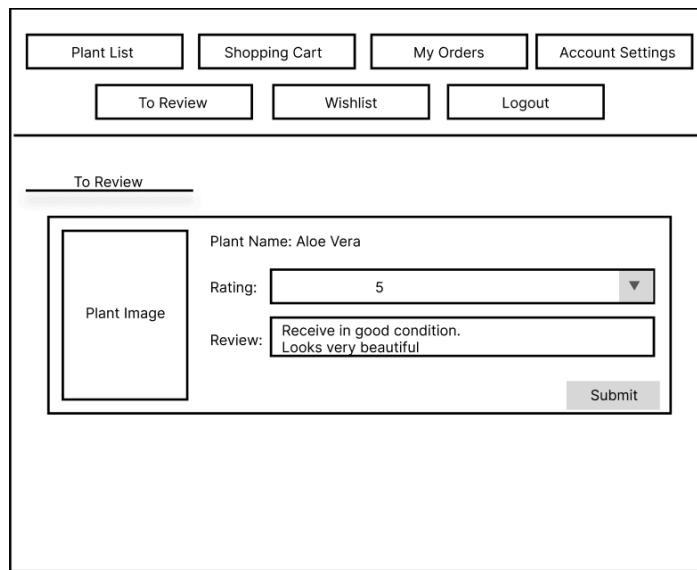


Figure 4.34: Leaves Reviews and Ratings Screen

4.4.4.11 Customer Manages a Wishlist Screen

Figure 4.35 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 4.22. By clicking the 'Wishlist' button, customers are led to the screen shown in Figure 4.35. 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.

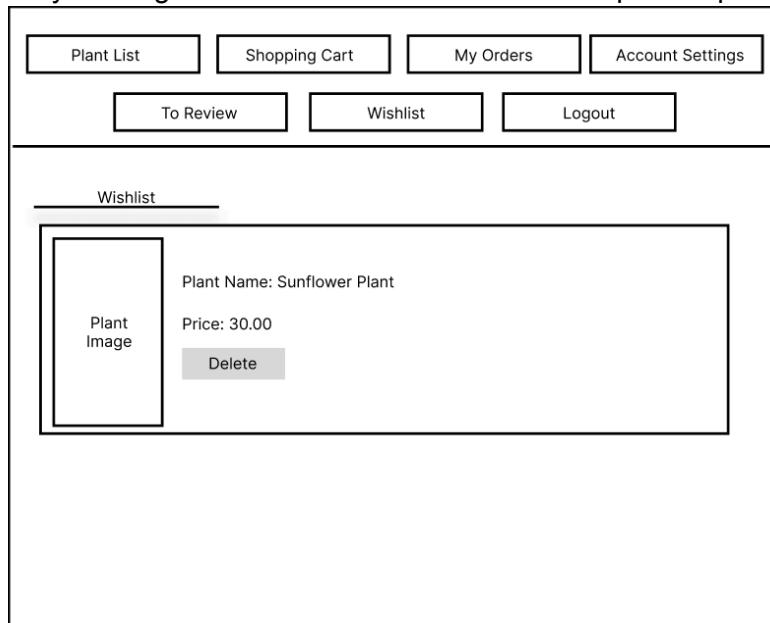


Figure 4.35: Manages a Wishlist Screen

4.4.5 Delivery Man Subsystem Screens

4.4.5.1 Delivery Man Home Screen

Figure 4.36 displays the "Delivery Man Home Screen" of the TLET Nursery Plant Shopping System. Upon login, the delivery man will be presented with a home screen (Figure 4.36) featuring a navigation bar with options for 'Pending Order', 'Accepted Deliveries', 'Account Settings', and a 'Logout' button. Below the navigation, a welcoming message marks the entry to the delivery man dashboard.



Figure 4.36: Delivery Man Home Screen

4.4.5.2 Delivery Man Creates a User Account Screen

Figure 4.37 shows the 'Create User Account Screen', which illustrates the process for delivery men to create a new account on the TLET Nursery Plant Shopping System. Upon accessing the system, the delivery man will first encounter the guest home screen in Figure 4.18, which is also the "TLET Nursery Plant Shopping System Screen" before creating a user account. The delivery man can proceed by clicking on the "Sign Up" button and this action takes the delivery man to the 'Creates User Account Screen' shown in Figure 4.37, where the delivery man fills out the form with personal and contact details. Upon clicking the "Submit" button, the system validates the input and saves the new user details to persistent storage. Successful registration is confirmed with a message. Delivery men who have an account can quickly get started by clicking the "Login" button, which takes them to a Login page in Figure 4.8.

 A wireframe diagram of a registration form titled 'Register'. The form instructions state: 'Please fill this form to create an account.' It contains ten input fields with labels: 'Full Name', 'Password', 'Email', 'Address', 'State', 'IC Number', 'Phone Number', and 'Role'. The 'Role' field is a dropdown menu. Below the form is a 'Register' button and a link 'Already have an account? [Login here](#)'.

Figure 4.37: Create User Account Screen

4.4.5.3 Delivery Changes Their Password Screen

Figure 4.39 shows the "Change Password Screen", which presents the interface for delivery man to update their password on the TLET Nursery Plant Shopping System. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 4.36, where the delivery man selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 4.38, where the delivery man details are listed. By selecting the 'Change Password' button, the delivery man is directed to the screen shown in Figure 4.39, where they are required to enter their current 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 delivery man is taken back to the "Account Settings Screen" shown in Figure 4.38, completing the password change process.

The screenshot shows the 'Account Settings' screen. At the top, there are four buttons: 'Pending Order', 'Accepted Deliveries', 'Account Settings' (which is highlighted), and 'Logout'. Below the buttons, the title 'Account Settings' is underlined. A large rectangular area contains the following user information:
 Full Name: Siti Fatimah
 Email: sitifatimah@gmail.com
 Phone Number: 0147894163
 Address: XXX
 State: Selangor
 IC No: 920504018888
 At the bottom of this information block are two buttons: 'Edit Profile' and 'Change Password'.

Figure 4.38: Account Settings Screen

The screenshot shows the 'Change Password' screen. At the top, there are four buttons: 'Pending Order', 'Accepted Deliveries', 'Account Settings' (highlighted), and 'Logout'. Below the buttons, the title 'Change Password' is underlined. The screen has two input fields: 'Current Password' and 'New Password', each with a corresponding empty text box. In the bottom right corner of the main input area is a 'Confirm' button.

Figure 4.39: Change Password Screen

4.4.5.4 Delivery Man Can Edit Profile Details Screen

Figure 4.40 shows 'Edit Profile Details Screen', where a delivery man can update their personal information on the TLET Nursery Plant Shopping System. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 4.36, where the delivery man selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 4.38, where the delivery man's details are listed. By selecting the 'Edit Profile' button, the delivery man is directed to the screen shown in Figure 4.40, featuring fields to modify the delivery man's personal information. Changes are submitted with the 'Confirm' button, and the system displays a confirmation message. If the delivery man chooses not to update the personal information, clicking the 'Back' button redirects them to the "Account Settings Screen" in Figure 4.38, without making any updates.

Full Name:	Siti Fatimah
Email:	sitifatimah@gmail.com
Phone Number:	0147894163
Address:	XXX
State:	Selangor
IC No:	920504018888

Figure 4.40: Edit Profile Screen

4.4.5.5 Delivery Man Accepts a Delivery Order Screen

Figure 4.41 shows the "Accepts a Delivery Order Screen" from the TLET Nursery Plant Shopping System, used by the delivery man to manage incoming orders. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 4.36, where the delivery man selects the "Pending Orders" button. This action takes them to the "Accepts a Delivery Order Screen" displayed in Figure 4.41. On this screen, the delivery man can view all pending orders with detailed information including the Delivery Order ID, customer names, customer IDs, contact numbers, and delivery addresses. They have the option to either accept or reject these orders directly from this interface. Once an order is accepted, it is marked as out of delivery. If the Delivery Man clicks the "Reject" button, the system removes the delivery order from the list of pending orders.

The screenshot shows a web-based application interface for delivery management. At the top, there is a navigation bar with four buttons: 'Pending Order', 'Accepted Deliveries', 'Account Settings', and 'Logout'. Below the navigation bar, the title 'Pending Orders' is displayed. Under this title, there are two separate boxes, each representing a pending delivery order. Each box contains the following information:

- Delivery Order ID: XXX
- Customer Name: Chin Yu Feng
- Customer ID: 1
- Phone Number: 0128457865
- Delivery address: 20, Taman Maju, Section 3/2a, Cheras, Selangor

Below each box are two buttons: 'Reject' on the left and 'Accept' on the right.

Figure 4.41: Accepts a Delivery Order Screen

4.4.5.6 Delivery Views Accepted Deliveries and Confirm Delivery Screen

Figure 4.42 presents the "Accepts a Deliveries and Confirm Delivery Screen" of the TLET Nursery Plant Shopping System. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 4.36, where the delivery man selects the 'Accepted Deliveries' button. This action takes them to the "Accepts a Deliveries and Confirm Delivery Screen" displayed in Figure 4.42 which allows delivery men to view a list of all plant deliveries they have accepted, complete with details such as delivery order ID, customer names, phone numbers, and delivery addresses. After the successful delivery of an order, the delivery man can confirm the completion of delivery by clicking the "Confirm Delivery" button provided alongside each order. This confirmation updates the order's status to "Completed" in the system's database, ensuring that the delivery process is accurately tracked.

The screenshot shows a web-based application interface for delivery management. At the top, there is a navigation bar with four buttons: 'Pending Order', 'Accepted Deliveries', 'Account Settings', and 'Logout'. Below the navigation bar, the title 'Accepted Deliveries' is displayed. Under this title, there are two separate boxes, each representing an accepted delivery order. Each box contains the following information:

- Delivery OrderID: XXX
- Customer Name: Chin Yu Feng
- Phone Number: 0128457865
- Delivery address: 20, Taman Maju, Section 3/2a, Cheras, Selangor

Below each box is a single button labeled 'Confirm Delivery'.

Figure 4.42: Accepts a Deliveries and Confirm Delivery Screen

4.5 Main Components

Table 4.27 below outlines the subsystems and their corresponding components within the TLET Nursery Plant Shopping System. The Administrator subsystem includes functionalities such as managing the plants, overseeing order details, updating order status, and assigning deliveries. The Guest subsystem enables browsing the plant list, creating user accounts, and accessing reviews. Moreover, the Customer Subsystem encompasses a range of functionalities such as managing accounts, shopping for plants, viewing ongoing order details, and leaving reviews. Lastly, the Delivery Man subsystem involves managing accounts, accepting delivery orders, and confirming deliveries. Each subsystem is designed to cater to specific user roles, facilitating efficient operation of the Nursery Plant Shopping System.

Table 4.27: Subsystems and Components in the TLET Nursery Plant Shopping System

Subsystem	Components
Administrator	An administrator manages the plants. <ul style="list-style-type: none"> • Create a plant. (sub-use case) • Update a plant. (sub-use case) • Delete a plant. (sub-use case)
	An administrator searches a plant by plant name.
	An administrator views order details and updates order status.
	An administrator assigns deliveries to delivery men.
Guest	A guest browses the plant list.
	A guest searches a plant by plant name.
	A guest creates a user account.
	A guest views reviews and ratings
Customer	A customer changes their password
	A customer can edit profile details.
	A customer shops plants. <ul style="list-style-type: none"> • 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. <ul style="list-style-type: none"> • 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.
Delivery Man	A delivery man creates a user account.
	A delivery man changes their password.
	A delivery man can edit profile details.
	A delivery man accepts a delivery order.
	A delivery man views accepted deliveries and confirms delivery.

4.6 Deployment Diagram

4.6.1 Administrator Deployment Diagram

Figure 4.43 depicts a deployment configuration for a Nursery Plant Shopping System, facilitated through an Administrator PC equipped with a browser. The core components include Plant Management, Delivery Management, and Order Management, each providing essential functions within the system. Plant Management allows for creating new plants, updating plant existing records, searching for plants by name, and deleting plants, all interacting with a Plant Database. Delivery Management handles the assignment of deliveries, supported by an Order Database that is linked to the Delivery Man, Payment Database, and Order Item Database. The Payment Database is there to handle money-related details. Order Management includes viewing order details and updating order statuses, relying on Order and Order Item Databases. The Order Item Database is linked to the Plant Database to show plant details for the plants ordered for each order item.

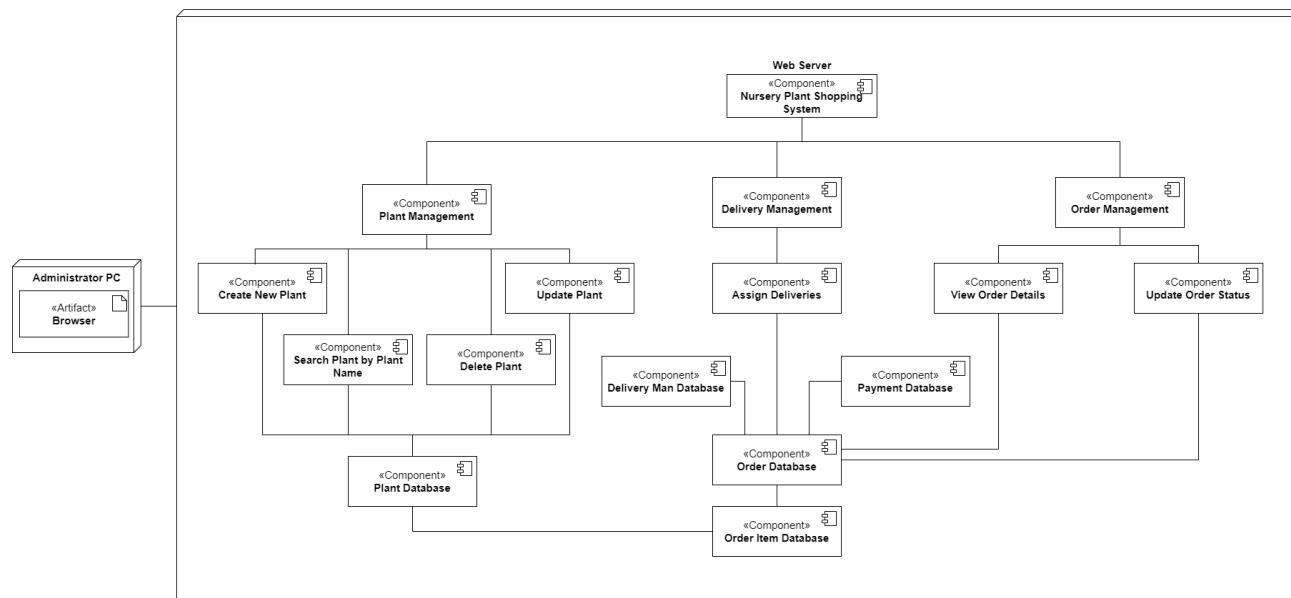


Figure 4.43: Administrator Deployment Diagram

4.6.2 Guest Deployment Diagram

Figure 4.44 shows a deployment diagram for guest users within the Nursery Plant Shopping System, accessed through a Guest PC with a browser. This diagram emphasizes user interaction components such as 'Search Plant by Plant Name' linked to a Plant Database, 'Create User Account' functionality linked to the Delivery Man Database and Customer Database, and 'View Reviews and Rating', which connects to a Review Database. Additionally, the Delivery Man and Customer Databases are interconnected with the User Database. In conclusion, guest users have the ability to browse plants, register for an account, and view reviews and ratings.

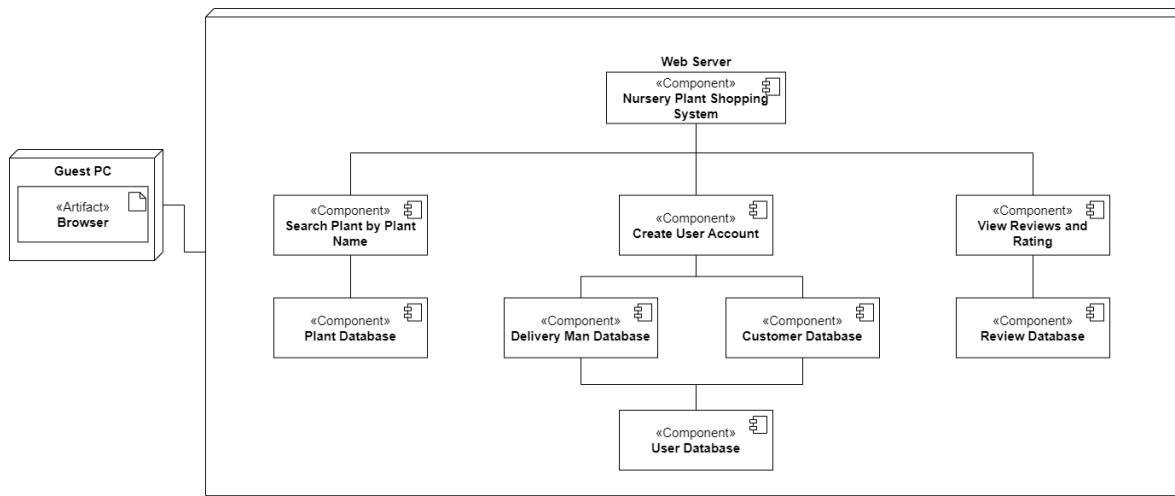


Figure 4.44: Guest Deployment Diagram

4.6.3 Customer Deployment Diagram

Figure 4.45 portrays a sophisticated nursery plant shopping system where a Customer PC with a "Browser" interfaces with a central Web Server for the Nursery Plant Shopping System. The server hosts an array of components for user engagement and functionality which includes 'Account Settings' with features to 'Edit Profile' and 'Change Password' for user account interactions. It interacts with both Customer Database and User Database to store user credentials. The shopping functionality enables customers to browse the 'Plant List' to search for specific plants, add plants to a shopping cart, and add to a wishlist. Each action is supported by the corresponding databases which are Plant Database, Shopping Cart database, Cart Item database, Wishlist database, and WishlistItem Database. The Plant Database holds inventory information; the Wishlist Database and WishlistItem database tracks user preferences; the Shopping Cart Database and Cart Item Database manage active shopping sessions. Within the 'Shopping Cart' where each component is linked to the Shopping Cart Database and Cart Item Database, the system allows customers to 'Adjust Plant Quantity', 'Delete Plants', and 'Make Payment' through a Payment Database. After purchase, customers can view 'My Orders' and submit reviews in 'To Review', with data managed by Order and Review Databases. The Order Database and Order Item Database archive purchase details; the Review Database collects user reviews and the Payment Database secures transaction records.

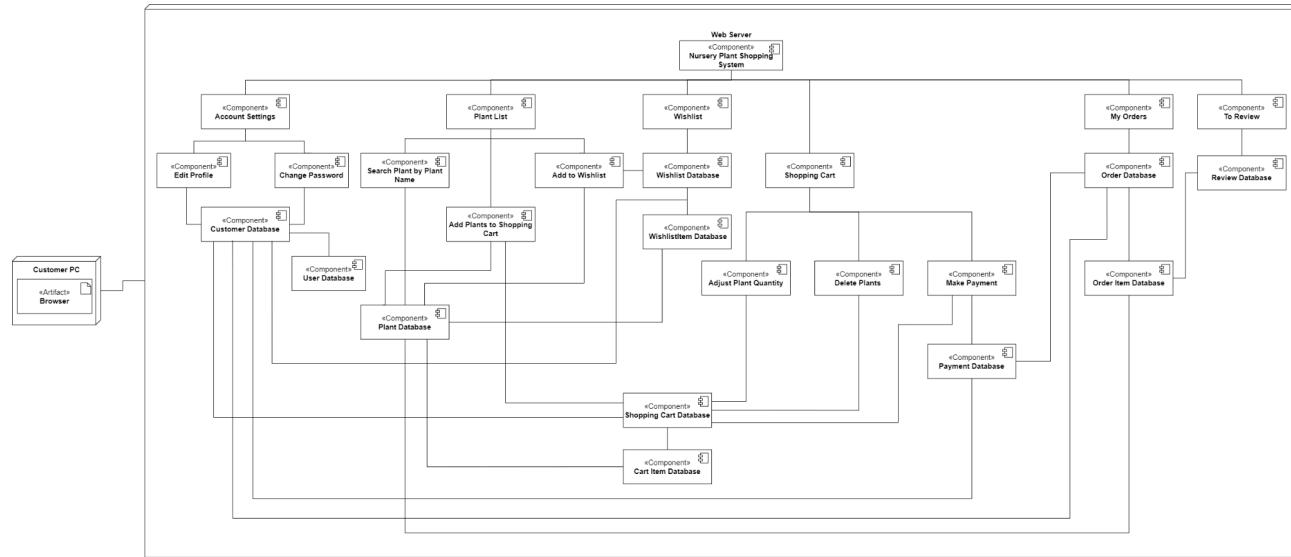


Figure 4.45: Customer Deployment Diagram

4.6.4 Delivery Man Deployment Diagram

Figure 4.46 illustrates the deployment diagram for delivery man within the Nursery Plant Shopping System. The system is accessed through a Delivery Man PC with a browser and facilitates the management of delivery operations. It includes components for handling 'Pending Order' and 'Accepted Deliveries' and features for 'Accept Delivery Order' and 'Confirm Delivery', interfacing with an Order Database and an Order Item Database. Additionally, there are 'Account Settings' with 'Change Password' and 'Edit Profile' options, which interact with a Delivery Man Database.

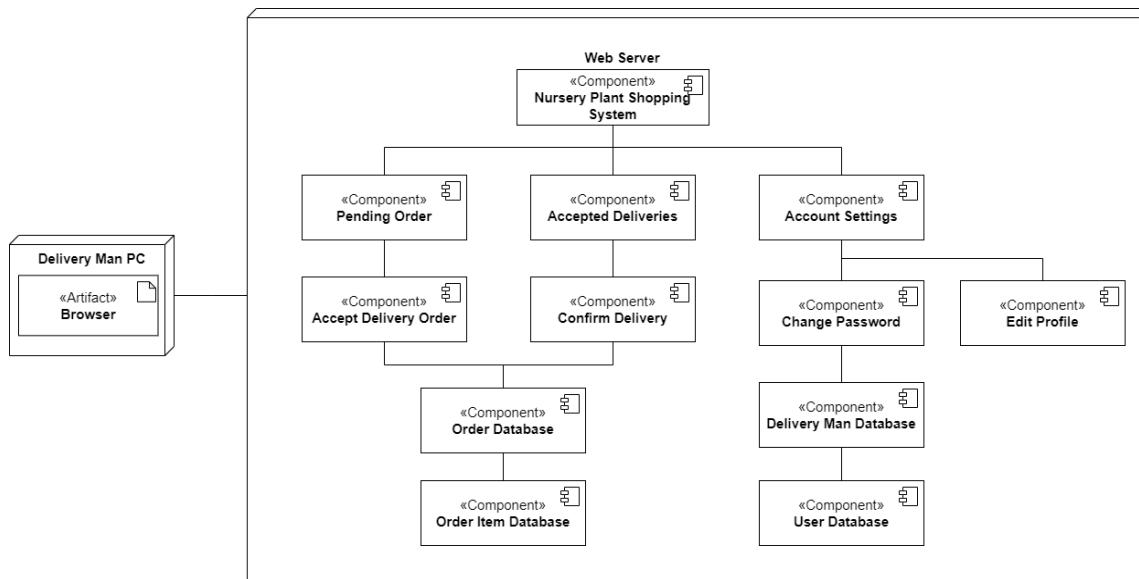


Figure 4.46: Delivery Man Deployment Diagram

5 Implementation

5.1 Development Environment

Figure 5.1 shows a screenshot of the project folder content for the TLET Nursery Plant Shopping System, developed using Django, a high-level Python web framework. Within the project directory, the “app” directory contains standard Django components. For example, “migrations”, “static” which contains the CSS file, “templates” which contains the HTML templates that are used to render various web pages, and some Python files. “admin.py” configures Django’s admin interface for data management via a web-based UI, while “backends.py” provides custom authentication methods for user logins. Besides, in “models.py”, data structures and storage methods are defined. “views.py” handles HTTP requests and generates responses, determining what users see on the website. “urls.py” maps URL patterns to corresponding view functions, directing incoming requests to the appropriate code for processing. Additionally, the “TruePltSys” directory includes the “media” folder for storing plant images, along with core Django files like “settings.py”, “urls.py”, “wsgi.py”, and “asgi.py”, which configure the overall behavior of the Django project. The “db.sqlite3” file serves as the SQLite database file used by Django to store data, while the “manage.py” script is a command-line utility that allows us to interact with this Django project.

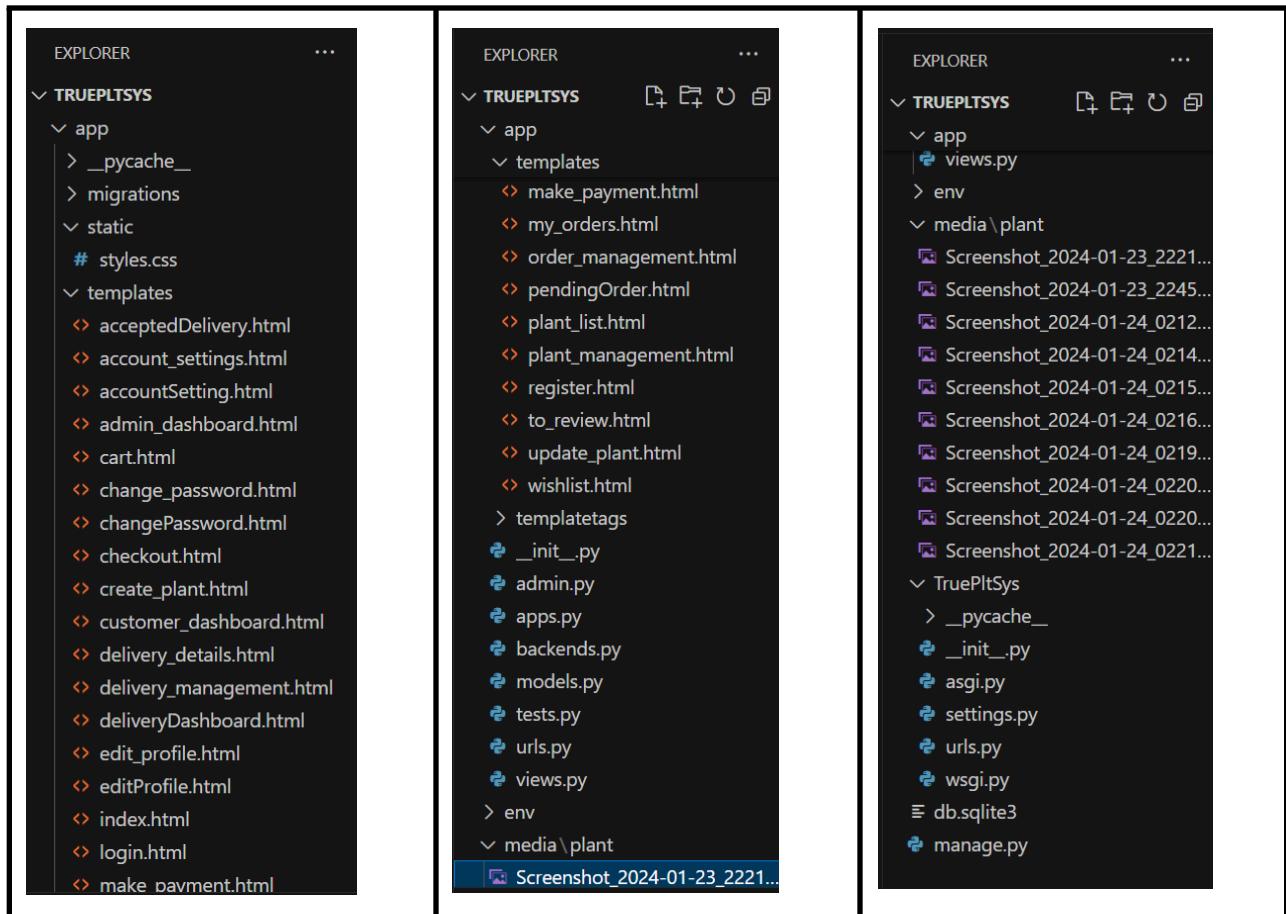


Figure 5.1: Screenshot of the Project Folder Content for the TLET Nursery Plant Shopping System

5.2 Software Integration

Figure 5.2 presents the models.py file for the TLET Nursery Plant Shopping System, developed using Django, a Python web framework. This file defines essential database models for system functionality, encompassing user types (Administrator, Customer, Delivery Man), plants, shopping carts, cart items, wishlists, wishlist items, payments, orders, order items, and reviews. Each model encapsulates specific attributes and behaviours crucial for managing user accounts, plant inventory, shopping transactions, and order fulfillment within the application.

```

from django.contrib.auth.models import AbstractUser
from django.db import models
from django.core.validators import MinValueValidator, MaxValueValidator
from django.db.models.signals import post_save, post_delete
from django.dispatch import receiver
# Create your models here.

# by TOH EE LIN
class User(AbstractUser):
    is_customer = models.BooleanField(default=False)
    is_deliveryman = models.BooleanField(default=False)
    is_admin = models.BooleanField(default=False)
    email = models.EmailField(unique=True)

    USERNAME_FIELD = 'email'
    REQUIRED_FIELDS = []      # Remove 'email' from here since it's the USERNAME_FIELD

# by TOH EE LIN
class Admin(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE, primary_key=True, related_name='admin_profile')

# by TOH EE LIN
class Customer(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE, primary_key=True, related_name='customer_profile')
    customer_name = models.CharField(max_length=80, default='Unknown')
    customer_address = models.CharField(max_length=200)
    customer_state = models.CharField(max_length=30)
    customer_ic = models.CharField(max_length=30, unique=True)
    customer_phone_number = models.BigIntegerField()

    def __str__(self):
        return self.user.email      # Using the email field from the related User object

# by TOH EE LIN
class DeliveryMan(models.Model):
    user = models.OneToOneField(User, on_delete=models.CASCADE, primary_key=True, related_name='deliveryman_profile')
    deliveryman_name = models.CharField(max_length=80, default='Unknown')
    deliveryman_address = models.CharField(max_length=200)
    deliveryman_state = models.CharField(max_length=30)
    deliveryman_ic = models.CharField(max_length=30, unique=True)
    deliveryman_phone_number = models.BigIntegerField()

    def __str__(self):
        return self.user.email

# by TOH EE LIN
class Plant(models.Model):
    plant_name = models.CharField(max_length=80)
    plant_image = models.ImageField(upload_to='plant/')
    plant_description = models.CharField(max_length=500)
    plant_price = models.DecimalField(max_digits=5, decimal_places=2)
    plant_availability = models.PositiveIntegerField(validators=[MinValueValidator(1), MaxValueValidator(999)])

# by TOH EE LIN
class ShoppingCart(models.Model):
    customer = models.ForeignKey(Customer, on_delete=models.CASCADE, related_name='shopping_carts')
    cart_total_price = models.DecimalField(max_digits=5, decimal_places=2)

```

```

def get_total_cart_price(self):
    return sum(item.get_total_item_price() for item in self.cart_items.all())
def update_total_price(self):
    self.cart_total_price = self.get_total_cart_price()
    self.save()

# by TOH EE LIN
class CartItem(models.Model):
    cart = models.ForeignKey(ShoppingCart, on_delete=models.CASCADE, related_name='cart_items')
    plant = models.ForeignKey(Plant, on_delete=models.CASCADE, related_name='cart_items')
    cart_plant_quantity = models.PositiveIntegerField(default=1, validators=[MinValueValidator(1), MaxValueValidator(50)])
    cart_item_price = models.DecimalField(max_digits=5, decimal_places=2)

    def get_total_item_price(self):
        return self.cart_plant_quantity * self.plant.plant_price

# by TOH EE LIN
class Wishlist(models.Model):
    customer = models.ForeignKey(Customer, on_delete=models.CASCADE, related_name='wishlist')

# by TOH EE LIN
class WishlistItem(models.Model):
    wishlist = models.ForeignKey(Wishlist, on_delete=models.CASCADE, related_name='wishlist_items')
    plant = models.ForeignKey(Plant, on_delete=models.CASCADE, related_name='wishlist_items')

# by TOH EE LIN
class Payment(models.Model):
    customer = models.ForeignKey(Customer, on_delete=models.CASCADE, related_name='payments')
    shipping_fee = models.DecimalField(max_digits=4, decimal_places=2, validators=[MinValueValidator(0.00), MaxValueValidator(99.99)])
    payment_cost = models.DecimalField(max_digits=6, decimal_places=2, validators=[MinValueValidator(0.00), MaxValueValidator(9999.99)])
    receive_method = models.CharField(max_length=10)
    order_date = models.DateField()
    order_total_price = models.DecimalField(max_digits=6, decimal_places=2, validators=[MinValueValidator(0.00), MaxValueValidator(9999.99)])

# by TOH EE LTN
class Order(models.Model):
    customer = models.ForeignKey(Customer, on_delete=models.CASCADE, related_name='orders')
    payment = models.ForeignKey(Payment, on_delete=models.CASCADE, related_name='orders')
    delman = models.ForeignKey(DeliveryMan, on_delete=models.CASCADE, related_name='orders', null=True, blank=True)
    order_status = models.CharField(max_length=30)

# by TOH EE LIN
class OrderItem(models.Model):
    order = models.ForeignKey(Order, on_delete=models.CASCADE, related_name='order_items')
    plant = models.ForeignKey(Plant, on_delete=models.CASCADE, related_name='order_items')
    plant_quantity = models.PositiveIntegerField(validators=[MinValueValidator(1), MaxValueValidator(50)])
    order_item_price = models.DecimalField(max_digits=5, decimal_places=2)
    to_review = models.BooleanField(default=True)

    def get_total_price(self):
        return self.plant_quantity * self.plant.plant_price

# by TOH EE LIN
class Review(models.Model):
    order_item = models.ForeignKey(OrderItem, on_delete=models.CASCADE, related_name='reviews')
    rating = models.IntegerField(validators=[MinValueValidator(1), MaxValueValidator(5)])
    comment = models.TextField(max_length=500)

@receiver(post_save, sender=CartItem)
def update_cart_total_on_save(sender, instance, **kwargs):
    instance.cart.update_total_price()

@receiver(post_delete, sender=CartItem)
def update_cart_total_on_delete(sender, instance, **kwargs):
    instance.cart.update_total_price()

```

Figure 5.2: app/models.py

Figure 5.3 displays the urls.py file for the TLET Nursery Plant Shopping System. This file maps URL patterns to corresponding views, facilitating navigation and functionality within the application. It organizes URLs into categories such as general user interaction, administrator tasks, delivery man operations, and customer actions. The file also incorporates settings for serving media files during development.

```

from django.urls import path
from . import views
from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    # General
    path("", views.home, name="home"),
    path("user/login/", views.loginPage, name="loginPage"),
    path("user/signup/", views.registrationPage, name="registrationPage"),
    path("user/accountSetting/", views.accountSetting, name = 'accountSetting'),
    path("logout/", views.logout, name="logout"),
    # Administrator
    path("adm/dashboard/", views.adminDashboard, name="adminDashboard"),
    path("adm/plant/manage/", views.plantManagement, name="plantManagement"),
    path("adm/plant/create/", views.createPlant, name="createPlant"),
    path("adm/plant/update/<int:plant_id>", views.updatePlant, name="updatePlant"),
    path("adm/plant/delete/<int:plant_id>", views.deletePlant, name="deletePlant"),
    path("adm/order/", views.orderManagement, name="orderManagement"),
    path("adm/order/update_status/<int:order_id>", views.update_order_status, name='update_order_status'),
    path("adm/delivery/", views.deliveryManagement, name="deliveryManagement"),
    path("adm/delivery/assign_delman/<int:order_id>", views.assign_delivery, name='assign_delivery'),
    # Delivery man
    path('del/changePassword/', views.changePassword, name = 'changePassword'),
    path('del/editProfile/', views.editProfile, name = 'editProfile'),
    path('del/deliveryDashboard/', views.deliveryDashboard, name = 'deliveryDashboard'),
    path('del/pendingOrder/', views.pendingOrder, name = 'pendingOrder'),
    path('del/acceptedDelivery/', views.acceptedDelivery, name = 'acceptedDelivery'),
    # Check data
    path('checkData/', views.check_existing_data, name = 'checkData'),
    path('checkEditProfile/', views.check_existing_data_editProfile, name = 'checkEditProfile'),
    path('customer_dashboard/', views.customer_dashboard, name='customer_dashboard'),
    path('plant_list/', views.plant_list_view, name='plant_list'),
    path('add_to_cart/<int:plant_id>', views.add_to_cart, name='add_to_cart'),
    path('cart/', views.view_cart, name='view_cart'),
    path('remove_from_cart/<int:item_id>', views.remove_from_cart, name='remove_from_cart'),
    path('update_cart_item/<int:item_id>', views.update_cart_item, name='update_cart_item'),
    path('wishlist/', views.view_wishlist, name='view_wishlist'),
    path('add_to_wishlist/<int:plant_id>', views.add_to_wishlist, name='add_to_wishlist'),
    path('remove_from_wishlist/<int:item_id>', views.remove_from_wishlist, name='remove_from_wishlist'),
    path('to_review/', views.to_review, name='to_review'),
    path('submit_review/<int:order_item_id>', views.submit_review, name='submit_review'),
    path('checkout/', views.checkout, name='checkout'),
    path('delivery_details/', views.delivery_details, name='delivery_details'),
    path('make_payment/', views.make_payment, name='make_payment'),
    path('my_orders/', views.my_orders, name='my_orders'),
    path('account_settings/', views.account_settings, name='account_settings'),
    path('profile/edit/', views.update_profile, name='edit_profile'), # For displaying the form
    path('profile/update/', views.update_profile, name='update_profile'), # For handling the form submission
    #path('update_profile/', views.update_profile, name='update_profile'),
    path('change_password/', views.change_password, name='change_password'),
]

# Add the following to serve media files during development
if settings.DEBUG:
    urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)

```

Figure 5.3: app/urls.py

Figure 5.4 illustrates the admin.py file for the TLET Nursery Plant Shopping System. This file customizes the Django admin interface by registering various models. Each registered model is accompanied by a corresponding admin class, defining how model data is displayed in the Django admin interface. Customizations include defining list display fields, creating custom methods to retrieve related data, and enhancing user interaction within the Django admin interface. This Django admin site serves as a powerful tool for the administrator to manage database records, and maintain system integrity effortlessly.

```
from django.contrib import admin
from django.contrib.auth.admin import UserAdmin
from .models import User, Admin, Customer, DeliveryMan, Plant, ShoppingCart, CartItem, Wishlist, WishlistItem, Payment, Order
from .models import OrderItem, Review

# Register your models here.

# by TOH EE LIN
class CustomUserAdmin(UserAdmin):
    model = User
    # Add custom fields to list_display to show up in the admin list view.
    list_display = ('username', 'email', 'is_customer', 'is_deliveryman', 'is_admin', 'is_staff', 'is_active')
    # Add custom fields to fieldsets to include them in the admin form view.
    fieldsets = UserAdmin.fieldsets + (
        None, {'fields': ('is_customer', 'is_deliveryman', 'is_admin')}),
    )

# by TOH EE LIN
class DjangoAdmin(admin.ModelAdmin):
    list_display = ('get_user_email',)

    def get_user_email(self, obj):
        return obj.user.email
    get_user_email.short_description = 'Email'

# by TOH EE LIN
class CustomerAdmin(admin.ModelAdmin):
    list_display = ('customer_name', 'get_user_email', 'customer_state')

    def get_user_email(self, obj):
        return obj.user.email
    get_user_email.short_description = 'Email'    # Sets the column header

# by TOH EE LIN
class DelmanAdmin(admin.ModelAdmin):
    list_display = ('deliveryman_name', 'get_user_email', 'deliveryman_phone_number')

    def get_user_email(self, obj):
        return obj.user.email
    get_user_email.short_description = 'Email'

# by TOH EE LIN
class PlantAdmin(admin.ModelAdmin):
    list_display = ('plant_name', 'plant_price', 'plant_availability')

# by TOH EE LIN
class CartAdmin(admin.ModelAdmin):
    list_display = ('get_customer_name', 'get_customer_email', 'cart_total_price')

    def get_customer_email(self, obj):
        return obj.customer.user.email
    get_customer_email.short_description = 'Email'
```

```
def get_customer_name(self, obj):
    return obj.customer.customer_name
get_customer_name.short_description = 'Customer Name'

# by TOH EE LIN
class CartItemAdmin(admin.ModelAdmin):
    list_display = ('get_customer_name', 'get_customer_email', 'get_plant','cart_plant_quantity', 'get_cartitem_price')

    def get_customer_email(self, obj):
        return obj.cart.customer.user.email
    get_customer_email.short_description = 'Email'

    def get_customer_name(self, obj):
        return obj.cart.customer.customer_name
    get_customer_name.short_description = 'Customer Name'

    def get_plant(self, obj):
        return obj.plant.plant_name
    get_plant.short_description = 'Plant Name'

    #def get_cartitem_price(self, obj):
    #    #return obj.cart_item_price
    #get_cartitem_price.short_description = 'Item Price (x Quantity)'
    def get_cartitem_price(self, obj):
        return obj.cart_plant_quantity * obj.plant.plant_price
    get_cartitem_price.short_description = 'Item Price (x Quantity)'

# by TOH EE LIN
class WishlistAdmin(admin.ModelAdmin):
    list_display = ('get_customer_name', 'get_customer_email')

    def get_customer_email(self, obj):
        return obj.customer.user.email
    get_customer_email.short_description = 'Email'

    def get_customer_name(self, obj):
        return obj.customer.customer_name
    get_customer_name.short_description = 'Customer Name'

# by TOH EE LIN
class WishlistItemAdmin(admin.ModelAdmin):
    list_display = ('get_customer_name', 'get_customer_email', 'get_plant')

    def get_customer_email(self, obj):
        return obj.wishlist.customer.user.email
    get_customer_email.short_description = 'Email'

    def get_customer_name(self, obj):
        return obj.wishlist.customer.customer_name
    get_customer_name.short_description = 'Customer Name'

    def get_plant(self, obj):
        return obj.plant.plant_name
    get_plant.short_description = 'Plant Name'
```

```

# by TOH EE LIN
class PaymentAdmin(admin.ModelAdmin):
    list_display = ('get_customer_name', 'get_customer_email', 'payment_cost', 'receive_method')

    def get_customer_email(self, obj):
        return obj.customer.user.email
    get_customer_email.short_description = 'Email'

    def get_customer_name(self, obj):
        return obj.customer.customer_name
    get_customer_name.short_description = 'Customer Name'

# by TOH EE LIN
class OrderAdmin(admin.ModelAdmin):
    list_display = ('get_customer_name', 'get_customer_email', 'get_payment_cost', 'get_order_date', 'get_receivemethod',
                   'order_status', 'get_deliveryman_name')

    def get_customer_email(self, obj):
        return obj.customer.user.email
    get_customer_email.short_description = 'Email'

    def get_customer_name(self, obj):
        return obj.customer.customer_name
    get_customer_name.short_description = 'Customer Name'

    def get_payment_cost(self, obj):
        return obj.payment.payment_cost
    get_payment_cost.short_description = 'Payment Cost'

    def get_receivemethod(self, obj):
        return obj.payment.receive_method
    get_receivemethod.short_description = 'Receive Method'

    def get_order_date(self, obj):
        return obj.payment.order_date
    get_order_date.short_description = 'Order Date'

    def get_deliveryman_name(self, obj):
        try:
            return obj.delman.deliveryman_name
        except:
            return None
    get_deliveryman_name.short_description = 'Deliveryman Name'

# by TOH EE LIN
class OrderItemAdmin(admin.ModelAdmin):
    list_display = ('get_customer_name', 'get_customer_email', 'get_plant', 'plant_quantity', 'get_orderitem_total_price')

    def get_customer_email(self, obj):
        return obj.order.customer.user.email    # path: OrderItem -> Order -> Customer -> User
    get_customer_email.short_description = 'Email'

    def get_customer_name(self, obj):
        return obj.order.customer.customer_name
    get_customer_name.short_description = 'Customer Name'

    def get_plant(self, obj):
        return obj.plant.plant_name
    get_plant.short_description = 'Plant Name'

    def get_orderitem_total_price(self, obj):
        return obj.get_total_price()
    get_orderitem_total_price.short_description = 'Total Price'

# by TOH EE LIN
class ReviewAdmin(admin.ModelAdmin):
    list_display = ('get_plant', 'rating', 'comment')

    def get_plant(self, obj):
        return obj.order_item.plant.plant_name
    get_plant.short_description = 'Plant Name'

```

```
# by TOH EE LIN
admin.site.register(User, CustomUserAdmin)
admin.site.register(Admin, DjangoAdmin)
admin.site.register(Customer, CustomerAdmin)
admin.site.register(DeliveryMan, DelmanAdmin)
admin.site.register(Plant, PlantAdmin)
admin.site.register(ShoppingCart, CartAdmin)
admin.site.register(CartItem, CartItemAdmin)
admin.site.register(Wishlist, WishlistAdmin)
admin.site.register(WishlistItem, WishlistItemAdmin)
admin.site.register(Payment, PaymentAdmin)
admin.site.register(Order, OrderAdmin)
admin.site.register(OrderItem, OrderItemAdmin)
admin.site.register(Review, ReviewAdmin)
```

Figure 5.4: app/admin.py

Figure 5.5 shows the backends.py file for the TLET Nursery Plant Shopping System. This file defines an authentication backend named EmailBackend, which extends Django's ModelBackend. The EmailBackend provides functionality to authenticate users using their email address instead of the traditional username. Additionally, it includes methods to authenticate users and retrieve user instances based on their email or user ID. This customization enhances the authentication process within the system, allowing users to log in using their email credentials.

```
from django.contrib.auth.backends import ModelBackend
from django.contrib.auth import get_user_model

User = get_user_model()

# by TOH EE LIN
class EmailBackend(ModelBackend):
    def authenticate(self, request, email=None, username=None, password=None, **kwargs):
        # To support Django Admin login, (auto convert username to email)
        if username:
            email = username

        try:
            user = User.objects.get(email__exact=email)
            if user.check_password(password):
                return user
        except User.DoesNotExist:
            return None

    def get_user(self, user_id):
        try:
            return User.objects.get(pk=user_id)
        except User.DoesNotExist:
            return None
```

Figure 5.5: app/backends.py

Figure 5.6 illustrates the settings.py file for the TLET Nursery Plant Shopping System. This file contains configurations such as specifying the database engine as SQLite3, defining authentication backends for user login, setting up URL paths for login and media files, as well as specifying the custom user model for the application. These settings collectively define the behaviour and functionality of the TLET Nursery Plant Shopping System project within the Django framework.

```
"""
Django settings for TruePltSys project.

Generated by 'django-admin startproject' using Django 5.0.1.

For more information on this file, see
https://docs.djangoproject.com/en/5.0/topics/settings/

For the full list of settings and their values, see
https://docs.djangoproject.com/en/5.0/ref/settings/
"""

import os
from pathlib import Path

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/5.0/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'django-insecure-42qg%4mh4kkgh1t8!_ay&i1_hk)+7tm0wfcj0hb567^dhwm$'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'app',
    'bootstrap5',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'TruePltSys.urls'
```

```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

WSGI_APPLICATION = 'TruePltSys.wsgi.application'

# Database
# https://docs.djangoproject.com/en/5.0/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}

# Password validation
# https://docs.djangoproject.com/en/5.0/ref/settings/#auth-password-validators

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]

# Internationalization
# https://docs.djangoproject.com/en/5.0/topics/i18n/

LANGUAGE_CODE = 'en-us'

TIME_ZONE = 'UTC'
```

```
USE_I18N = True

USE_TZ = True

# Static files (CSS, JavaScript, Images)
# https://docs.djangoproject.com/en/5.0/howto/static-files/

STATIC_URL = '/static/'

# Default primary key field type
# https://docs.djangoproject.com/en/5.0/ref/settings/#default-auto-field

DEFAULT_AUTO_FIELD = 'django.db.models.BigAutoField'

AUTH_USER_MODEL = "app.User"

AUTHENTICATION_BACKENDS = ['app.backends.EmailBackend']

LOGIN_URL = "loginPage"

# Base url to serve media files
MEDIA_URL = '/media/'
# Path where media is stored
MEDIA_ROOT = os.path.join(BASE_DIR, 'media/')
```

Figure 5.6: TruePltSys/settings.py

Figure 5.7 shows the registrationPage function within the views.py file. This code snippet handles the registration process when a user submits their details through the registration form. It validates the form fields, creates a new user account with the provided information, and assigns a role-based profile, either as a customer or a delivery man. Upon successful registration, the user is automatically logged in and redirected to the corresponding dashboard, while an alert flag is set to welcome them. In case of errors, such as duplicate email or IC numbers, the system displays an error message for the user to correct the information.

```
UserModel = get_user_model()

def registrationPage(request):
    request.session['show_alert'] = False # After successfully registered, will show message

    if request.method == 'POST':
        full_name = request.POST.get('fullName')
        password = request.POST.get('password')
        email = request.POST.get('email')
        address = request.POST.get('address')
        state = request.POST.get('state')
        ic_number = request.POST.get('icNumber')
        phone_number = request.POST.get('phoneNumber')
        role = request.POST.get('role')

        try:
            with transaction.atomic():
                user = UserModel.objects.create(
                    username=email.split('@')[0], # for username
                    email=email,
                    is_customer=(role == 'customer'),
                    is_deliveryman=(role == 'delivery_man')
                )

                user.set_password(password)
                user.save()

# Depending on the role, create the profile
if role == 'customer':

    Customer.objects.create( # Save inside customer table
        user=user,
        customer_name=full_name,
        customer_address=address,
        customer_state=state,
        customer_ic=ic_number,
        customer_phone_number=phone_number
    )

    login(request, user)
    request.session['user_email'] = user.email
    request.session['show_alert'] = True
    return redirect('customer_dashboard')
```

```
    elif role == 'delivery_man':  
  
        DeliveryMan.objects.create( # Save inside delivery man table  
            user=user,  
            deliveryman_name=full_name,  
            deliveryman_address=address,  
            deliveryman_state=state,  
            deliveryman_ic=ic_number,  
            deliveryman_phone_number=phone_number  
        )  
  
        # login(request, user)  
        request.session['user_email'] = user.email  
        request.session['show_alert'] = True  
        return redirect('deliveryDashboard')  
  
    else:  
        invalid_messages = 'Invalid role selected.'  
        return render(request, 'register.html', {'messages': invalid_messages})  
  
except Exception as e:  
    # If any error occurs during the user creation, roll back the transaction.  
    messages.error(request, f'An error occurred: {e}')  
    return render(request, 'register.html')  
  
# If it's not a POST request or something went wrong, display the registration form again  
return render(request, 'register.html')
```

Figure 5.7: app/views.py/registrationPage

Figure 5.8 illustrates the LoginPage function within the views.py file, showcasing the login functionality. When a user submits their credentials via the login form, this view function authenticates the user and determines their role using attributes such as "is_admin", "is_deliveryman", and "is_customer", which are set during account creation. Based on the user's role, the system redirects them to the corresponding dashboard using the "redirect" function with the appropriate URL name. This role-based redirection ensures that each user is directed to the relevant subsystems, including administrator subsystem, delivery man subsystem, or customer subsystem.

```
def loginPage(request):
    messages = None # Initial
    if request.method == 'POST':
        user_email = request.POST.get('email')
        user_password = request.POST.get('password')
        user = authenticate(request, email=user_email, password=user_password)
        if user is not None:
            login(request, user)
            # Redirect to different dashboards based on the user's role
            if user.is_admin:
                request.session['user_email'] = user.email # Determine user
                return redirect('adminDashboard')

            elif user.is_deliveryman:
                request.session['user_email'] = user.email
                return redirect('deliveryDashboard')

            elif user.is_customer:
                request.session['user_email'] = user.email
                return redirect('customer_dashboard') # Replace with with customer de URL

            else:
                messages = "You do not have the permission to login."

        else:
            messages = "Invalid email or password."

    return render(request, 'login.html', {"messages": messages})
```

Figure 5.8: app/views.py/loginPage

5.3 Database

5.3.1 User Database

Figure 5.3.1 presents a User Database that is integral to our nursery plant shopping system, detailing the authentication and role designation of each participant. It contains unique identifiers (ID), secured hashed passwords, and usernames for individual login credentials. The database categorizes users into administrator, customers, and delivery man, enabling role-based access to the system. Additionally, email addresses are also recorded for communication purposes.

	id	password	username	is_active	is_customer	is_deliveryman	is_admin	email
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	pbkdf2_sha256\$600000\$2UUDs1XYiK...	Eelin	1	0	0	1	eelin@gmail.com
2	39	pbkdf2_sha256\$600000\$ywzygIx1pY...	yufeng	1	1	0	0	yufeng@gmail.com
3	40	pbkdf2_sha256\$600000\$TWwXueclUl...	zara8061	1	1	0	0	zara8061@gmail.com
4	41	pbkdf2_sha256\$600000\$xNAcq7kqwR...	patel	1	1	0	0	patel@gmail.com
5	42	pbkdf2_sha256\$600000\$2ZkvbWDhR...	sitifatimah	1	0	1	0	sitifatimah@gmail.com
6	43	pbkdf2_sha256\$600000\$6TihqjqaKLQ...	varna1957	1	0	1	0	varna1957@gmail.com
7	44	pbkdf2_sha256\$600000\$7YINIvC5h...	wilsonphang	1	1	0	0	wilsonphang@gmail.com
8	45	pbkdf2_sha256\$600000\$y1e9nTBcoE...	chiaamber	1	1	0	0	chiaamber@gmail.com
9	46	pbkdf2_sha256\$600000\$0PXviqaHHt...	zzahirr	1	1	0	0	zahir5024@gmail.com
10	50	pbkdf2_sha256\$600000\$bfdB7W8X2...	farahh	1	1	0	0	farahbeauty@gmail.com
11	51	pbkdf2_sha256\$600000\$4nXqqASSVF...	JohnYu	1	0	1	0	Yujohn0808@gmail.com
12	52	pbkdf2_sha256\$600000\$rU1ns4dnm...	PWTeh	1	0	1	0	won9595@gmail.com
13	53	pbkdf2_sha256\$600000\$jYx816JJjcEv...	MhmdSankar	1	0	1	0	sankar@gmail.com
14	54	pbkdf2_sha256\$600000\$SYnnitemxO...	MhmdAsh	1	0	1	0	Ashrof5555@gmail.com
15	55	pbkdf2_sha256\$600000\$3wz2Zx1fGg...	jj	1	0	1	0	jeffkuan@gmail.com
16	56	pbkdf2_sha256\$600000\$ROlVzAhASd...	faizz	1	0	1	0	faizul0000@gmail.com
17	57	pbkdf2_sha256\$600000\$azQ6z2UOk...	sRoslan	1	0	1	0	syedroslan@gmail.com
18	58	pbkdf2_sha256\$600000\$eoI9MgRrw...	kana	1	0	1	0	kanavan@gmail.com
19	59	pbkdf2_sha256\$600000\$sBWvjV7IACf...	aqil	1	1	0	0	aqilah1111@gmail.com
20	60	pbkdf2_sha256\$600000\$ncAL3KGaYp...	graceC	1	1	0	0	gracechua@gmail.com
21	61	pbkdf2_sha256\$600000\$X7ZqRYFh4h...	ziliyiC	1	1	0	0	ziliyi0505@gmail.com

Figure 5.3.1: User Database

5.3.2 Administrator Database

Figure 5.3.2 shows a simplified view of the Administrator Database for our nursery plant shopping system. The database appears to include user_id, which corresponds to the unique identifier of an admin user from the main User Database in Figure 5.3.1.

user_id
Filter
1
1

Figure 5.3.2: Administrator Database

5.3.3 Customer Database

Figure 5.3.3 illustrates the Customer Database utilized by our nursery plant system. This database includes user_id that links to the general User Database for identification purposes. The customer_address field contains specific location details where our nursery products would be delivered. The customer_state column provides regional information of our customer base and for planning logistics. Additionally, each customer is associated with a unique customer_ic, a form of identification number, and a customer_phone_number for direct communication. Lastly, the database records the full customer_name, which personalizes the customer experience.

user_id	customer_address	customer_state	customer_ic	customer_phone_number	customer_name
Filter	Filter	Filter	Filter	Filter	Filter
1	39 20, Taman Maju, Section 3/2a, Cheras	Selangor	040202106450	60128457865	Chin Yu Feng
2	40 Gua Musang, 18300, Kelantan	Kelantan	710806031252	60112382536	Zara
3	41 1, Lebuh Tunku Kudin 1, 11700 ...	Penang	991219073328	60173328497	Visnu Patel
4	44 Jalan Muzium, 88300 Kota Kinabalu, ...	Sabah	090921013789	60128378358	Wilson Phang
5	45 1-d, 2nd Floor, Jalan Air Jernih, 2030...	Terengganu	830830116523	601123498765	Amber Chia
6	46 80, Jalan Kepong, 86200 Simpang ...	Johor	951220015890	60199887766	Zahir
7	50 Tanjung Malai, 07000, Kedah	Kedah	850930022020	60127802367	Farah
8	59 Jalan Pulau Gadong, 75250 Malacca	Malacca	990203044977	601123905642	Aqilah
9	60 Jln Pisang Keling, 97000 Bintulu, ...	Sarawak	860806058878	60127717890	Grace Chua
10	61 Kampung Bachang, 75350 Malacca	Malacca	050202043390	60192314587	Ziiyi Chin

Figure 5.3.3: Customer Database

5.3.4 DeliveryMan Database

Figure 5.3.4 displays the Delivery Man Database of our nursery plant shopping system, which is specifically organized to manage the details of the personnel responsible for the delivery of plants to customers. The database holds a user_id for each delivery person, which serves as a link to their profile in the main User Database. It also lists the deliveryman_address and deliveryman_state, which are crucial for assigning delivery tasks based on their locations and for logistical planning. Each delivery person has a designated deliveryman_ic, a personal identification number, and a deliveryman_phone_number to facilitate prompt and efficient communication. The deliveryman_name field is included to identify each delivery agent personally, which is vital for customers to recognize the delivery staff.

	user_id	deliveryman_address	deliveryman_state	deliveryman_ic	deliveryman_phone_number	deliveryman_name
	Filter	Filter	Filter	Filter	Filter	Filter
1	42	35, Jalan BPU6 Taman Indah, Puchong	Selangor	920504018888	60147894163	Siti Fatimah
2	43	66, Lorong 9, Lembah Dato Harun, ...	Sabah	860504083327	60127783043	Varna
3	51	77, Jalan Bukit Aman 3/4H, PJU5, ...	Johor	900312019088	60190567892	John Yu
4	52	AA, Lot 3435N, Jubilee Park, 30450, ...	Perak	821030128788	60169023459	Teh Pu Won
5	53	Lot 3, Jalan 5/72, Puncak Utara, 0623...	Kedah	770921023066	601123907899	Mohamed Sankar
6	54	No. 3-4, Jalan Sasaran 7, Seksyen 21...	Kelantan	810321039055	601133279065	Muhammad Ashrof
7	55	No. 8, Lorong 3/500, Puncak Gembir...	Sarawak	750123133344	60126329037	Jeff
8	56	Lot 63, Lorong 8T, Seri Keramat, 013...	Perlis	900430093221	60138788880	Faizul
9	57	14, Jalan 3F, PJU8, 10172 Tanjung ...	Penang	960628078903	60172229030	Syed Roslan
10	58	No. 3, Jalan 3B, Seri Putra, 63166 ...	Selangor	890730109056	60168907890	Kanavan

Figure 5.3.4: DeliveryMan Database

5.3.5 Plant Database

Figure 5.3.5 showcases the Plant Database for our nursery plant shopping system. The 'id' column assigns a unique identifier to each plant. 'Plant_name' lists the common names of the plants, such as Aloe Vera, Sunflower Plant, and Cactus, among others. The 'plant_image' column appears to contain paths to images of each plant, providing a visual reference. 'Plant_description' offers brief descriptions, highlighting key characteristics and care instructions for each species. 'Plant_price' indicates the cost per unit, which is essential for sales and inventory management. Lastly, 'plant_availability' shows the quantity of each plant in stock.

	id	plant_name	plant_image	plant_description	plant_price	plant_availability
	Filter	Filter	Filter	Filter	Filter	Filter
1	1	Aloe Vera	plant/...	Aloe vera is a succulent plant species of the ...	10	120
2	2	Sunflower Plant	plant/...	Sunflowers, with their bright yellow blooms ...	30	86
3	3	Cactus	plant/...	Cactus is a diverse group of succulent plants ...	15	55
4	4	Live Rose Plant	plant/...	Roses, with their unique combination of thor...	35	243
5	5	Hibiscus Plant	plant/...	The hibiscus plant is an annual or perennial ...	25	70
6	6	Adenium obesum	plant/...	Adenium obesum commonly called desert ros...	30	128
7	7	African Daisy	plant/...	African Daisy, known for their cheerful ...	25	149
8	8	Ruellia Pink	plant/...	Ruellia Pink, a vibrant plant variety, thrives in...	10	88
9	9	Camellia Azalea	plant/...	Camellia Azalea is valued for its beautiful lar...	15	135
10	10	Sansevieria SP	plant/...	Sansevieria SP is smaller in size but its ...	15	50

Figure 5.3.5: Plant Database

5.3.6 ShoppingCart Database

Figure 5.3.6 exhibits the ShoppingCart Database from our nursery plant shopping system. This database includes an 'id' that serves as a unique identifier for each customer's shopping cart session. The 'cart_total_price' column reflects the total price of the items added to the cart, crucial

for checkout and payment processing. The 'customer_id' column links each shopping cart to a specific customer in the User Database, ensuring that each cart's contents and total value are accurately associated with the right customer.

	id	cart_total_price	customer_id
	Filter	Filter	Filter
1	48	50	39
2	49	270	40
3	50	595	41
4	51	50	44
5	52	25	46
6	53	270	45
7	54	105	50
8	55	45	59
9	56	30	60
10	57	275	61

Figure 5.3.6: ShoppingCart Database

5.3.7 CartItem Database

Figure 5.3.7 represents the CartItem Database of our nursery plant shopping system that tracks individual items within a customer's shopping cart. The 'id' is a unique identifier for each entry. The 'cart_plant_quantity' column specifies the quantity of a particular plant chosen by the customer, while the 'cart_item_price' reflects the total price for the specified quantity of that plant. The 'plant_id' column links to the specific plant in the Plant Database, and the 'cart_id' associates the item with a particular customer's shopping cart from the ShoppingCart Database.

	id	cart_plant_quantity	cart_item_price	plant_id	cart_id
	Filter	Filter	Filter	Filter	Filter
1	72	5	50	1	48
2	73	9	270	2	49
3	74	5	150	2	50
4	75	7	245	4	50
5	76	8	200	5	50
6	77	2	50	5	51
7	78	1	25	7	52
8	79	12	120	8	53
9	80	10	150	10	53
10	81	3	45	10	54
11	82	2	60	6	54
12	83	3	45	9	55
13	84	2	30	9	56
14	85	15	225	10	57
15	86	5	50	1	57

Figure 5.3.7: CartItem Database

5.3.8 Wishlist Database

Figure 5.3.8 depicts the Wishlist Database of our nursery plant shopping system, which allows customers to save their favorite plants. The 'id' column is a unique identifier for each wishlist entry, whereas the 'customer_id' links the wishlist item to a specific customer in the User Database.

	id	customer_id
	Filter	Filter
1	10	39
2	11	40
3	12	41
4	13	44
5	14	45
6	15	46
7	16	50
8	17	59
9	18	60
10	19	61

Figure 5.3.8: Wishlist Database

5.3.9 WishlistItem Database

Figure 5.3.9 portrays the WishlistItem Database, which details the items customers have saved to their wishlists. The 'id' serves as a unique identifier for each entry in the wishlist. 'Plant_id' corresponds to the specific plants that customers have shown interest in, which is linked to the Plant Database. The 'wishlist_id' associates the plant with a particular customer's wishlist from the Wishlist Database.

	id	plant_id	wishlist_id
	Filter	Filter	Filter
1	15	1	10
2	16	2	11
3	17	3	11
4	18	4	12
5	19	5	13
6	20	4	13
7	21	4	15
8	22	5	15
9	23	9	14
10	24	10	16
11	25	2	17
12	26	5	17
13	27	6	18
14	28	6	19
15	29	7	19

Figure 5.3.9: WishlistItem Database

5.3.10 Payment Database

Figure 5.3.10 displays the Payment Database for our nursery plant shopping system to record and manage all financial transactions related to customer orders. The 'id' represents a unique identifier for each payment record. 'Shipping_fee' details the costs associated with the delivery of plants to the customer, while 'payment_cost' could denote the total amount paid, which include product prices and shipping fees. 'Receive_method' specifies how the customer chooses to receive their order, whether by delivery or pickup. 'Order_date' is a timestamp of when the order was placed. 'Order_total_price' gives the total cost of the order before adding shipping fees, and 'customer_id' associates each payment record with a specific customer from the User Database.

	id	shipping_fee	payment_cost	receive_method	order_date	order_total_price	customer_id
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	42	7	57	Delivery	2024-02-05	50	39
2	43	0	270	Pickup	2024-02-05	270	40
3	44	0	595	Pickup	2024-02-06	595	41
4	45	12	62	Delivery	2024-02-07	50	44
5	46	7	277	Delivery	2024-02-07	270	45
6	48	0	25	Pickup	2024-02-08	25	46
7	49	7	112	Delivery	2024-02-09	105	50
8	50	0	45	Pickup	2024-02-09	45	59
9	51	12	42	Delivery	2024-02-09	30	60
10	52	7	282	Delivery	2024-02-10	275	61

Figure 5.3.10: Payment Database

5.3.11 Order Database

Figure 5.3.11 depicts the Order Database for our nursery plant shopping system. Each record is uniquely identified by an 'id' and displays the 'order_status' to indicate its progression which is Completed, Ready, or Out Of Delivery. The 'payment_id' links to the corresponding financial transaction, while the 'customer_id' associates the order with the customer's profile from the User Database. The 'delman_id' column is reserved for the identifier of the assigned delivery personnel, with 'NULL' entries indicating that no delivery person has yet been assigned to the order.

	id	order_status	payment_id	customer_id	delman_id
	Filter	Filter	Filter	Filter	Filter
1	42	Completed	42	39	42
2	43	Completed	43	40	NULL
3	44	Completed	44	41	NULL
4	45	Completed	45	44	43
5	46	Completed	46	45	51
6	48	Completed	48	46	NULL
7	49	Ready	49	50	53
8	50	Ready	50	59	NULL
9	51	Out Of Delivery	51	60	55
10	52	Out Of Delivery	52	61	56

Figure 5.3.11: Order Database

5.3.12 OrderItem Database

Figure 5.3.12 shows the OrderItem Database, which is an integral part of the nursery plant shopping system, capturing the details of each plant item within an order. This database records the 'id' as a unique identifier for each order item, 'plant_quantity' for the number of plants ordered, and 'order_item_price' reflecting the total price for those plant quantities. The 'to_review' column indicates whether an item has been assigned for review. The 'order_id' links the item to a specific order in the Order Database, and 'plant_id' associates it with a particular plant in the Plant Database.

	id	plant_quantity	order_item_price	to_review	order_id	plant_id
	Filter	Filter	Filter	Filter	Filter	Filter
1	51	5		10 0	42	1
2	52	9		30 0	43	2
3	53	5		30 0	44	2
4	54	7		35 0	44	4
5	55	8		25 0	44	5
6	56	2		25 0	45	5
7	57	12		10 0	46	8
8	58	10		15 0	46	10
9	60	1		25 0	48	7
10	61	3		15 1	49	10
11	62	2		30 1	49	6
12	63	3		15 1	50	9
13	64	2		15 1	51	9
14	65	15		15 1	52	10
15	66	5		10 1	52	1

Figure 5.3.12: OrderItem Database

5.3.13 Review Database

Figure 5.3.13 exhibits the Review Database to capture customer feedback on individual orders. The database comprises columns for 'id', serving as a unique identifier for each review; 'rating', which is a numerical representation of customer satisfaction; 'comment', containing detailed customer feedback about their order; and 'order_item_id', which links the review to a specific item in the OrderItem Database. This allows for precise tracking of customer sentiment and product performance.

			comment	order_item_id
	id	rating		
	Filter	Filter	Filter	Filter
1	19	5	Receive in good condition. Looks very beautiful	51
2	20	5	Love this plant shop—easy online browsing, and my plants arrived looking fabulous.	52
3	21	5	Quick delivery, got some real beauties, and the prices were a pleasant surprise	53
4	22	1	Found a few damaged leaves on one plant and the delivery was a bit delayed.	54
5	23	5	Wallet-friendly and my new plant babies are thriving, couldn't be happier	55
6	24	3	Prices are a tad high, but the unique plant varieties and their exceptional condition upon arrival justified the cost	56
7	25	3	Delivery took longer than expected, but the healthy plants made it worth the wait.	60
8	26	3	Plants weren't as big as pictured	57
9	27	4	Received a different pot size than expected, but the plants were healthy.	58

Figure 5.3.13: Review Database

6 Testing

6.1 Testing Strategy

In this web application, sandwich testing is utilized for integration testing. Initially, the main classes are established, and modules for each actor, such as administrator, guest, customer, and delivery man, are created and thoroughly tested with appropriate data.

Once all user modules are fully tested and integrated, the integration process begins with the guest and customer modules. After successfully integrating these modules, they are then combined with the administrator's modules, which handle order management and assignment of delivery tasks. Subsequently, the delivery man modules are integrated to ensure smooth order fulfillment and delivery processes. Finally, comprehensive testing using test data is conducted to evaluate the functionality of the integrated nursery plant shopping system.

6.2 Test Data

6.2.1 Administrator Test Data Set

Table 6.2.1.1 below shows the Administrator Credentials, including the administrator's email, first name, last name, and password. This table contains the necessary credentials for logging in as an administrator.

Table 6.2.1.1: Administrator Credentials

Administrator Email	Administrator First Name	Administrator Last Name	Administrator Password
eelin@gmail.com	Toh	Ee Lin	123456

Table 6.2.1.2 below shows the Create Plant Test Data. This table provides test data for creating plant records, encompassing both positive outcomes and scenarios that trigger error messages. It includes details such as the plant name, plant description, plant quantity, plant price, and plant image for various plant entries.

Table 6.2.1.2: Create Plant Test Data

Plant Name	Plant Description	Plant Quantity	Plant Price	Plant Image
Money Tree	Characterised by its unique tree-like plant with a stout trunk. It has been said to bring good fortune and success.	80	35.00	
Cactus	Cactus is a diverse group of succulent plants known for their unique and often unusual appearance. It typically has fleshy, water-storing stems that are adapted to reduce water loss.	15	55.00	
??12	Characterised by its unique tree-like plant with a stout trunk. It has been said to bring good fortune and success.	80	35.00	
Jade Plant	Rounded green fleshy leaves with red lining believed to bring wealth and good fortune	0	29.00	

Jade Plant	Rounded green fleshy leaves with red lining believed to bring wealth and good fortune	50	0.00	
Jade Plant	Rounded green fleshy leaves with red lining believed to bring wealth and good fortune.	50	29.00	(blank)

Table 6.2.1.3 below shows the Update Plant Test Data. This table showcases test data for updating plant records, encompassing various scenarios that may yield either successful updates or trigger error messages. It includes details such as the plant name, plant description, plant quantity, plant price, and plant image.

Table 6.2.1.3: Update Plant Test Data

Plant Name	Plant Description	Plant Quantity	Plant Price	Plant Image
aloe vera	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.	50	20.00	
Cactus	Cactus is a diverse group of succulent plants known for their unique and often unusual appearance. It typically has fleshy, water-storing stems that are adapted to reduce water loss.	15	55.00	

120	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.	50	20.00	
Aloe Vera	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.	50	0.00	

Table 6.2.1.4 below shows the Delete Plant Test Data. In this table, the entry "Money Tree" represents the specific plant test data that is intended for deletion. Upon clicking the "Delete" button associated with this plant, it serves as a verification step to check whether the "Money Tree" plant record remains within the system or has been effectively removed. This serves as a test scenario for evaluating the deletion functionality.

Table 6.2.1.4: Delete Plant Test Data

Plant Name
Money Tree

Table 6.2.1.5 below shows the Search Plant Test Data. In this table, a variety of plant names are presented to be entered into the search bar. These plant names are the test data for searching plants, which encompass both valid inputs that yield search results and invalid inputs that trigger error messages.

Table 6.2.1.5: Search Plant Test Data

Plant Name
sunflower
h

P
na
(blank)
/
20

Table 6.2.1.6 below shows the Search Order Test Data. In this table, various order IDs are provided to be entered into the search bar. These order IDs are the test data for searching orders, which encompassing both valid inputs that yield successful searches and invalid inputs that trigger error messages.

Table 6.2.1.6: Search Order Test Data

Order ID
43
20
(blank)
a
43w
/

Table 6.2.1.7 below shows the Update Order Status Test Data. This table presents test data for updating order statuses, specifically testing the functionality of the nursery plant shopping system's order status update feature. It includes order IDs and the corresponding statuses to which they should be updated.

Table 6.2.1.7: Update Order Status Test Data

Order ID	Order Status (Update to)
43	Ready
44	Ready
44	Completed

45	Ready
46	Ready

Table 6.2.1.8 below shows the Search Delivery Order Test Data. In this table, various order IDs are provided to be entered into the search bar. These order IDs are the test data for searching delivery orders, which encompassing both valid inputs that yield successful searches and invalid inputs that trigger error messages.

Table 6.2.1.8: Search Delivery Order Test Data

Order ID
45
80
(blank)
zz
45a
@

Table 6.2.1.9 below presents the Assign Delivery Order Test Data. This table includes test data for assigning delivery orders to specific delivery men. It provides order IDs and the corresponding names of delivery men to whom the delivery orders should be assigned. These entries serve as test cases to evaluate the functionality of the nursery plant shopping system in assigning delivery orders to delivery men.

Table 6.2.1.9: Assign Delivery Order Test Data

Order ID	Delivery Man Name (Assign to)
45	Siti Fatimah
46	Varna

6.2.2 Guest Test Data Set

Table 6.2.2.1 is structured to display test data for a plant search feature by plant name. It lists five rows with the columns "Plant ID," "Plant Name," "Plant Price," and "Plant Availability." Notably, the "Lily" plant has a dash for both its ID and its availability, indicating that it is not present in the database or inventory. In contrast, the table includes entries for "Aloe Vera," "Sunflower Plant," "Cactus," and "Live Rose Plant," with their respective prices and availability numbers.

Search Plant by Plant Name Test Data:

Plant Credential:

Table 6.2.2.1: Plant List Test Data

Plant ID	Plant Name	Plant Price	Plant Availability
-	Lily	-	-
1	Aloe Vera	10.00	138
2	Sunflower Plant	30.00	98
3	Cactus	15.00	54
4	Live Rose Plant	35.00	250

Table 6.2.2.2 and 6.2.2.3 show the test data for creating a customer and delivery man's user account. The guest is not able to create their account if they have numbers in their name. They can't create their account if their email address does not have '@'. Lastly, they are not able to create an account if their IC and phone number have characters.

Creates User Account Test Data:

Customer:

Table 6.2.2.2: Create Customer Account Test Data

Customer	Customer	Customer	Customer	Customer State	Customer IC	Customer

Name	Password	Email	Address			Phone Number
Chin Yu Feng12	123	yufeng@gmail.com	20, Taman Maju, Section 3/2a, Cheras	Selangor	040202106450	60128457865
Zara	zara**88	zara8061@gmail.com	Gua Musang, 18300, Kelantan	Kelantan	710806031252	60112382536
Visnu Patel	visnu_9999	patel@gmail.com	1, Lebuh Tunku Kudin 1, 11700 Jelutong, Pulau Pinang	Penang	991219073328w	6017332849w7
Alicia Lim	lim#3030	alicia1210@gmail.com	Jalan Lengkongan, Kampung Melayu Kulai, 81000 Kulai, Johor	Johor	650121013277	60143001623

Delivery Man:

Table 6.2.2.3: Create Delivery Man Account Test Data

Delivery Man Name	Delivery Man Password	Delivery Man Email	Delivery Man Address	Delivery Man State	Delivery Man IC	Delivery Man Phone Number
Siti Fatimah	qwerty156	sitifatimah@gmail.com	35, Jalan BPU6 Taman Indah,	Selangor	920504018888	60147894163

			Puchong			
Varna	vv@ @3323	varna1957@gma il.com	66, Lorong 9, Lembah Dato Harun, 90013 Tanjung Aru, Sabah	Sabah	860504083327	60127783043
John Yu	dd3@ea961	Yujohn0808@g mail.com	77, Jalan Bukit Aman 3/4H, PJU5, 82301 Kota Iskandar, Johor Darul Ta'zim	Johor	900312019088	60190567892
Teh Pu Won	teh@ !6687	won9595@gmail .com	AA, Lot3435N, Jubilee Park, 30450, Jalan Jubilee, Taman Jubilee, 30300 Ipoh, Perak	Perak	821030128788	60169023459

Table 6.2.2.4 is organized into "Plant Name," "Rating," and "Comment," and lists various plants along with their corresponding customer ratings and comments. This table serves to display the feedback and ratings given by customers for different plants in the shopping system, demonstrating the review functionality of the system.

View Reviews and Ratings Test Data:

Table 6.2.2.4: View Reviews and Ratings Test Data

Plant Name	Rating	Comment

Hibiscus Plant	-	-
Aloe Vera	4	Recommended!!
Sunflower Plant	5	The flowers are beautiful. Thankyou!
Cactus	5	Good service! Thankyou seller ~
Live Rose Plant	5	Beautiful !!

6.2.3 Customer Test Data

Table 6.2.3.1 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 6.2.3.1: Customer Changes Their Password Test Data

Customer Name	Customer Email	Customer Old Password	Customer New Password
Amber Chia	chiaamber@gmail.co m	amber**333	chiaamber@gmail.co m
Amber Chia	chiaamber@gmail.co m	amber**333	chia33
Amber Chia	chiaamber@gmail.co m	amber**333	qwertyuiop
Amber Chia	chiaamber@gmail.co m	amber**333	123456
Amber Chia	chiaamber@gmail.co m	amber**33	soya_milk
Amber Chia	chiaamber@gmail.co m	amber**333	soya_milk

Table 6.2.3.2 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 6.2.3.2: Customer Edits Profile Details Test Data

Customer Name	Customer Email	Customer Phone Number	Customer Address	Customer State	Customer IC
Amber	chiaamber@g mail.com	60112349876 6	35,Jalan BPU6 Taman Indah Puchong	Selangor	040202106450
AmberChia02 02	chiaambergm ail.com	60112349876 6	35,Jalan BPU6 Taman Indah Puchong	Selangor	040202106450
AmberChia02 02	chiaamber@g mail.com	01123498766 5	35,Jalan BPU6 Taman Indah Puchong	Selangor	040202106450
AmberChia02 02	chiaamber@g mail.com	60112349876 6	Jalan Muzium, 88300 Kota Kinabalu, Sabah	Sa	040202106450
AmberChia02 02	chiaamber@g mail.com	60112349876 5	Jalan Muzium, 88300 Kota Kinabalu, Sabah	Sabah	040202106450 mm
AmberChia02 02	chiaamber@g mail.com	60112349876 5	Jalan Muzium, 88300 Kota Kinabalu, Sabah	Sabah	040202106450

Table 6.2.3.3 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 6.2.3.3: 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.co m	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 6.2.3.4, 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 6.2.3.4: 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 6.2.3.5, 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 6.2.3.5: 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 6.3.2.6 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 6.2.3.6: 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 6.2.3.7, 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 6.2.3.7: Customer Makes A Payment Test Data

Custo	Custo	Plant	Plant	Price	Quan	Recei	Shipp	Total	Plant	Total
-------	-------	-------	-------	-------	------	-------	-------	-------	-------	-------

Customer Email	Customer State	Name	ID	Quantity	Pickup Method	Delivery Fee	(Price * Quantity)	Total Price	Cost(include shipping fee if applicable)
chiamember@gmail.com	Sabah	Sunflower Plant	2	RM30.00	1	Pickup	-	RM30.00	RM130.00
	Sabah	Adenium obesum	6	RM30.00	3	Pickup	-	RM90.00	RM130.00
	Sabah	Ruellia Pink	8	RM10.00	1	Pickup	-	RM10.00	RM130.00
	Sabah	Aloe Vera	1	RM10.00	2	Delivery	RM12.00	RM20.00	RM30.00
	Sabah	Ruellia Pink	8	RM10.00	1	Delivery	RM12.00	RM10.00	RM30.00
	Selangor	Camellia Azalea	9	RM15.00	1	Pickup	-	RM15.00	RM15.00
	Selangor	Cacti	3	RM15.00	2	Delivery	RM7.00	RM30.00	RM80.00
	Selangor	African Daisy	7	RM25.00	2	Delivery	RM7.00	RM50.00	RM80.00

Table 6.3.2.8, 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 6.2.3.8: 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 6.2.3.9 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 6.2.3.9: Customer Leaves Reviews And Ratings Test Data

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

Table 6.2.3.10 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 6.2.3.10: Customer Manages The Wishlist Test Data

Plant Name	Price
Cactus	RM15.00
Live Rose Plant	RM35.00

6.2.4 Delivery Man Test Data

Table 6.2.4.1 shows the test data for creating a delivery man user account. The delivery man is not able to create an account if they have numbers in their name. A delivery man's password must be at least 8 characters long and contain at least one alphabet. They can't create an account if their email address does not have '@'. Lastly, they are not able to create an account if their IC and phone number have characters.

Table 6.2.4.1: Create User Account Test Data

Delivery Man Name	Delivery Man Password	Delivery Man Email	Delivery Man Address	Delivery Man State	Delivery Man IC	Delivery Man Phone Number
Faizul Ahmad123	faizul@34**	faizul0000@gmail.com	Lot 63, Lorong 8T, Seri Keramat, 01311 Beseri, Perlis	Perlis	900430093221	60138788880
Faizul Ahmad	faizul@34**	faizul0000@gmail.com	Lot 63, Lorong 8T, Seri Keramat, 01311 Beseri, Perlis	Perlis	900430093221	60138788880
Faizul Ahmad	faizul@34**	faizul0000@gmail.com	Lot 63, Lorong 8T, Seri Keramat, 01311 Beseri, Perlis	Perlis	900430093221my	60138788880
Faizul Ahmad	faizul@34**	faizul0000@gmail.com	Lot 63, Lorong 8T, Seri Keramat, 01311 Beseri, Perlis	Perlis	900430093221	60138788880abc
Faizul Ahmad	123	faizul0000@gmail.com	Lot 63, Lorong 8T, Seri Keramat, 01311 Beseri, Perlis	Perlis	900430093221	60138788880

Faizul Ahmad	faizul@34**	faizul0000@gmail.com	Lot 63, Lorong 8T, Seri Keramat, 01311 Beseri, Perlis	Perlis	900430093221	60138788880
--------------	-------------	----------------------	--	--------	--------------	-------------

Table 6.2.4.2 shows the test data for changing a delivery man's password. The delivery man's new password must be at least 8 characters long and contain at least one alphabet. A delivery man's current password must be correct in order to change to the new password. A delivery man's new password cannot be the same as the current password.

Table 6.2.4.2: Change User Password Test Data

Delivery Man Name	Delivery Man Email	Delivery Man Current Password	Delivery Man New Password
Faizul Ahmad	faizul0000@gmail.com	faizul@34**	1234
Faizul Ahmad	faizul0000@gmail.com	faizul@34	faizul@34**12
Faizul Ahmad	faizul0000@gmail.com	faizul@34**	faizul@34**
Faizul Ahmad	faizul0000@gmail.com	faizul@34**	faizul@34**12

Table 6.2.4.3 shows the test data for editing a delivery man's user profile. The delivery man is not able to edit their profile if they have numbers in their name. They can't edit their profile if their email address does not have '@'. Lastly, they are not able to edit their profile if their IC and phone number have characters.

Table 6.2.4.3: Edit User Profile Test Data

Delivery Man Name	Delivery Man Email	Delivery Man Phone Number	Delivery Man Address	Delivery Man State	Delivery Man IC
Faizul Ahmad345	faizul0000@gmail.com	60138788880	Lot 63, Lorong 8T, Seri Keramat, 01311 Besi, Perlis	Perlis	900430093221
Faizul Ahmad	faizul0000@gmail.com	60138788880	Lot 63, Lorong 8T, Seri Keramat, 01311 Besi, Perlis	Perlis	900430093221
Faizul Ahmad	faizul0000@gmail.com	60138788880abc	Lot 63, Lorong 8T, Seri Keramat, 01311 Besi, Perlis	Perlis	900430093221
Faizul Ahmad	faizul0000@gmail.com	60138788880	Lot 63, Lorong 8T, Seri Keramat, 01311 Besi, Perlis	Perlis	900430093221my
Faizul Mirul	faizul123@gmail.com	60138788000	14, Jalan 3F, PJU8, 10172 Tanjung Bungah, Penang	Penang	900430093221

Table 6.2.4.4 shows the test data for accepting delivery order. If the delivery man is new and has not been assigned any delivery orders, the pending order section will be empty. A delivery man who has pending orders can accept or reject the order. If a delivery man accepts a pending order, the order will move to the accepted deliveries section. If a delivery man rejects a pending order, the order will be returned to the admin and the admin will have to assign the order again.

Table 6.2.4.4: Accept Delivery Order Test Data

Delivery Man Name	Delivery Man Email	Delivery Man Password	Has Pending Order	Delivery Order ID	Customer Name	Accept/Reject Order	Order Status
Faizul Mirul	faizul123@g mail.com	faizul@34**12	No	-	-	-	-
Siti Fatimah	sitifatimah@g mail.com	qwerty156	Yes	48	Chin Yu Feng	Accept	Ready
Siti Fatimah	sitifatimah@g mail.com	qwerty156	Yes	50	Zara	Reject	Ready

Table 6.2.4.5 shows the test data for viewing accepted deliveries and confirm delivery. If the delivery man is new and has not accepted any delivery orders, the accepted deliveries section will be empty. A delivery man who has accepted deliveries can confirm delivery after they have successfully delivered the order. If a delivery man confirms delivery for an order, the order status in the database will be updated to Completed.

Table 6.2.4.5: View Accepted Deliveries and Confirms Delivery Test Data

Delivery Man Name	Delivery Man Email	Delivery Man Password	Has Accepted Deliveries	Delivery Order ID	Customer Name	Order Status
Faizul Mirul	faizul123@gmai l.com	faizul@34**12	No	-	-	-
Siti Fatimah	sitifatimah@gma il.com	qwerty156	Yes	48	Chin Yu Feng	Out For Delivery

6.3 Acceptance Tests

6.3.1 Administrator Acceptance Test

Table 6.3.1.1 shows the administrator acceptance test with each of its use cases. This table outlines the specific tasks that an administrator should be able to perform within the TLET Nursery Plant Shopping System.

Module: Administrator

Developer: Toh Ee Lin

Table 6.3.1.1: Administrator Acceptance Test

Criteria	Fulfilled?	Remarks
An administrator creates a plant.		
An administrator updates a plant.		
An administrator deletes a plant.		
An administrator searches a plant by plant name.		
An administrator views order details and updates order status.		
An administrator assigns deliveries to delivery men.		

Date tested:

% Completed:

Test by:

Verified by:

6.3.2 Guest Acceptance Test

Module: Guest

Developer: Teo Yu Jie

Table 6.3.2.1: Guest's Acceptance Test

Criteria	Fulfilled?	Remarks

A guest browses the plant list.		
A guest searches a plant by plant name.		
A guest creates a user account.		
A guest views reviews and ratings.		

Date tested:

% Completed:

Test by:

Verified by:

6.3.3 Customer Acceptance Test

Module: Customer

Developer: Lim Cai Qing

Table 6.3.3.1: Customer Acceptance Test

Criteria	Fulfilled?	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:

6.3.4 Delivery Man Acceptance Test

Table 6.3.4.1 shows the delivery man acceptance test with each of its use cases.

Module: Delivery Man

Developer: Emily Phang Ru Ying

Table 6.3.4.1: Delivery Man Acceptance Test

Criteria	Fulfilled?	Remarks
A delivery man creates a user account.		
A delivery man changes their password.		
A delivery man can edit profile details.		
A delivery man accepts a delivery order.		
A delivery man views accepted deliveries and confirms delivery.		

Date tested:

% Completed:

Test by:

Verified by:

7 Sample Screens

7.1 Main Screens

Figure 7.1.1 presents the Home Screen of the TLET Nursery Plant Shopping System, designed for visitors. It offers a user-friendly layout with clear options for logging in or registering. Below, users can utilize a search bar to find plants by name, making navigation easy. Additionally, the screen displays visually appealing plant images, providing visitors with a preview of the diverse selection available for exploration.

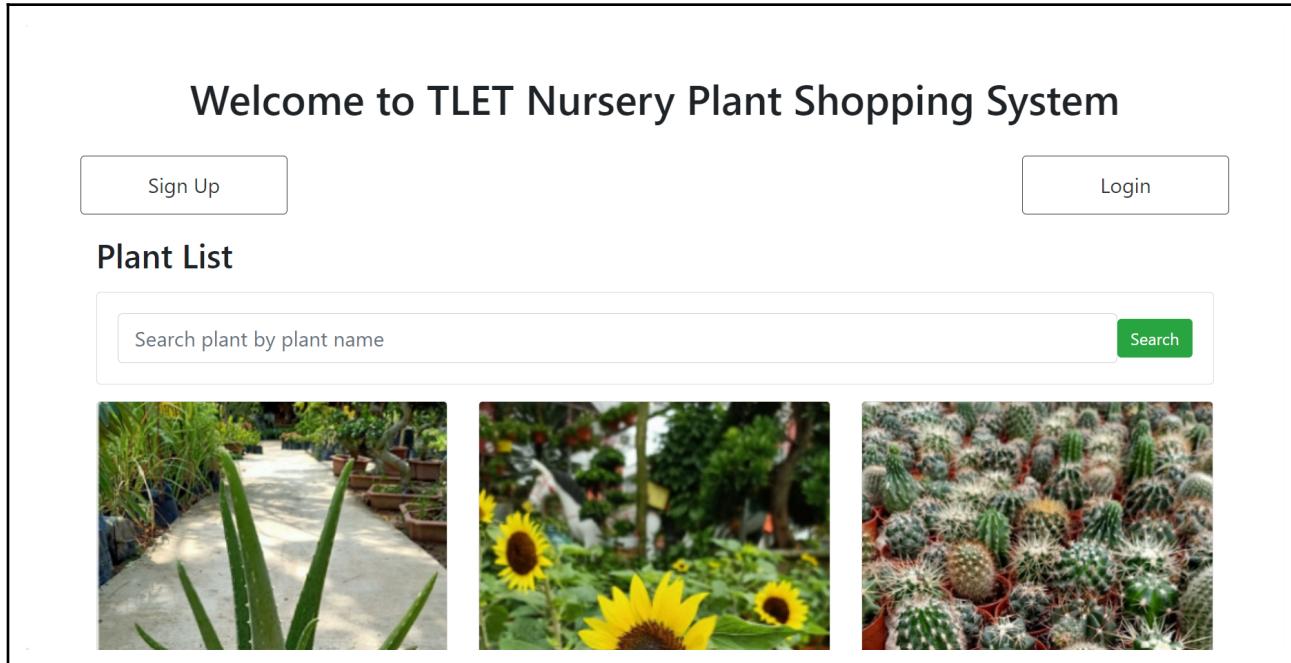


Figure 7.1.1: Home Screen

Figure 7.1.2 shows the Login Screen of the TLET Nursery Plant Shopping System. The screen prompts users to enter their email address and password into designated fields. Upon inputting the required details, users can proceed by clicking the “Submit” button to log in. Alternatively, for those without an account, there is an option to register by selecting the “Register here” link located beneath the login form.

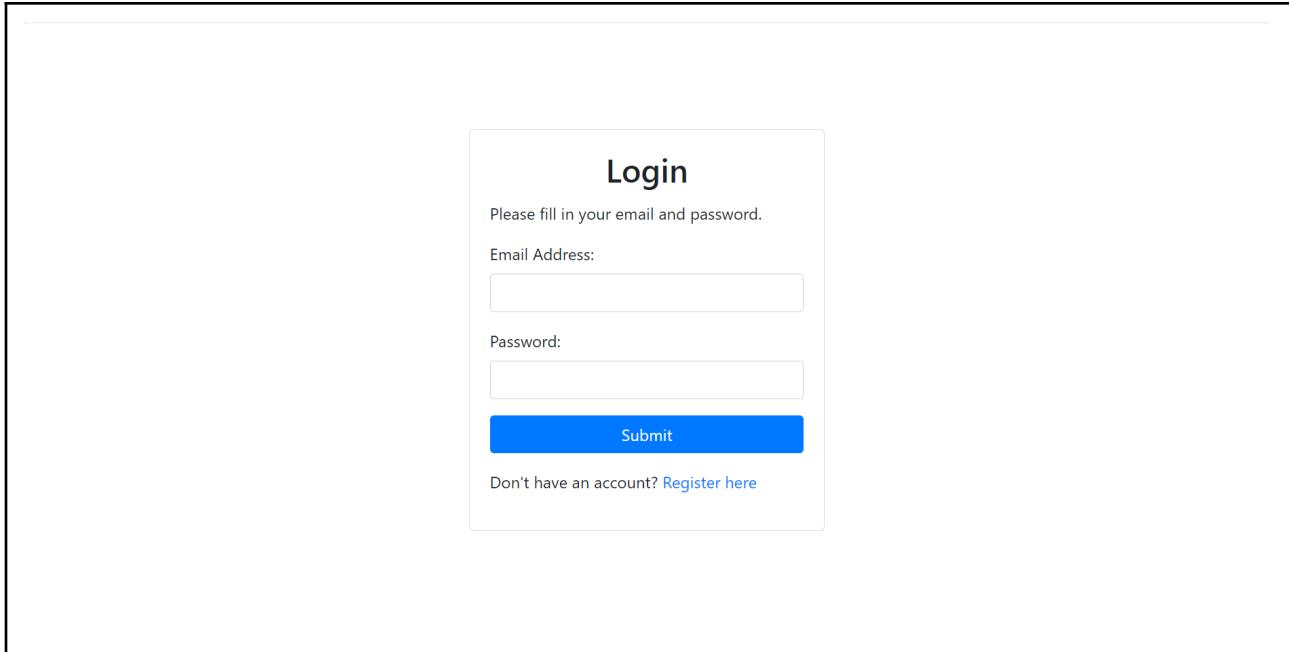


Figure 7.1.2: Login Screen

7.2 Administrator

7.2.1 Administrator Home Screen

Figure 7.2.1 below shows the “Administrator Home Screen”. The “Administrator Home Screen” features a welcoming message at the center of the screen, saying “Welcome to Admin Dashboard”, so administrator know he/she is in the right place. There is a navigation bar at the top of the Administrator Home Screen that neatly organizes essential management functions into tabs. These tabs include “Plant Management” for managing the plants, “Order Management” for tracking and updating the status of customer orders, “Delivery Management” for assigning delivery orders to delivery men, and “Logout” for administrator to safely exit the system and log out of his/her accounts.

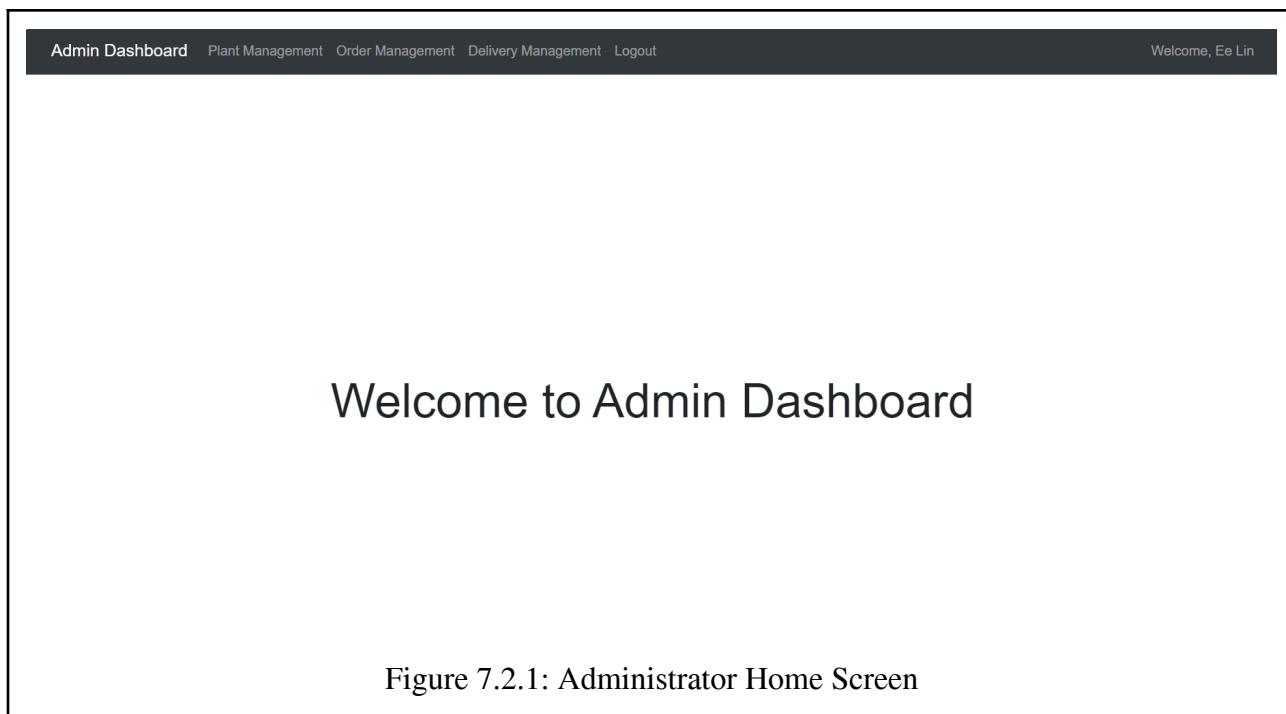
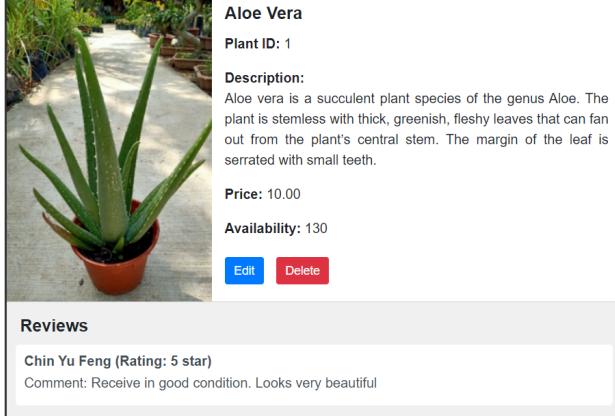


Figure 7.2.1: Administrator Home Screen

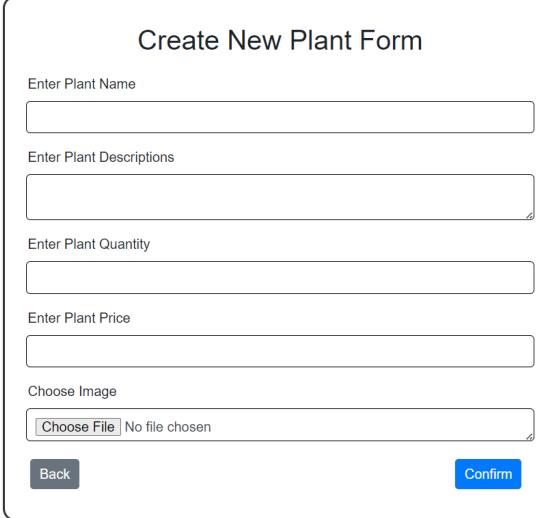
7.2.2 An administrator creates a plant

Figure 7.2.2 below shows the “Plant Management Screen”. After clicking on the “Create New Plant” button, the administrator will be redirected to the “Create New Plant Screen” (Figure 7.2.3). From here, the administrator can fill in the vital information about a new plant, such as its name, detailed descriptions, quantity available for sale, and price. The administrator can also upload a picture of the plant to make the plant information more accurate and eye-catching. Once the details are filled in, the administrator can click the “Confirm” button to submit the form or click the “Back” button to cancel the addition of new plant and return to “Plant Management Screen”.



The screenshot shows a web-based plant management system. At the top, there is a navigation bar with links: Admin Dashboard, Plant Management, Order Management, Delivery Management, and Logout. Below the navigation bar is a search bar labeled "Search Plant by Plant Name" with a "Search" button. A green "Create New Plant" button is also present. The main content area displays a potted Aloe Vera plant with its details: **Aloe Vera**, **Plant ID: 1**. The **Description** is: "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." The **Price** is listed as **10.00**. The **Availability** is **130**. There are "Edit" and "Delete" buttons at the bottom. Below the plant details, there is a section titled "Reviews" which contains a single entry: "Chin Yu Feng (Rating: 5 star) Comment: Receive in good condition. Looks very beautiful".

Figure 7.2.2: Plant Management Screen



The screenshot shows a "Create New Plant Form" dialog box. It contains fields for entering plant details: "Enter Plant Name" (with an empty input field), "Enter Plant Descriptions" (with an empty input field), "Enter Plant Quantity" (with an empty input field), "Enter Plant Price" (with an empty input field), and "Choose Image" (with a file input field showing "Choose File | No file chosen"). At the bottom of the form are two buttons: "Back" and "Confirm".

Figure 7.2.3: Create New Plant Screen

7.2.3 An administrator updates a plant

After clicking the "Edit" button of a particular plant in "Plant Management Screen" (Figure 7.2.2), the administrator will be redirected to the "Update Plant Screen" (Figure 7.2.4), which pre-populates with that plant's current details. From here, the administrator can update any piece

of information, whether it is the plant's name, descriptive text, quantity in stock, plant price or the plant image. Upon making the desired changes, the administrator can save the updates by clicking the "Update" button or choose to go back to "Plant Management Screen" without saving changes using the "Back" button.

Admin Dashboard Plant Management Order Management Delivery Management Logout

Update Plant Form

Enter Plant Name

Enter Plant Descriptions

Enter Plant Quantity

Enter Plant Price

Choose Image
 No file chosen

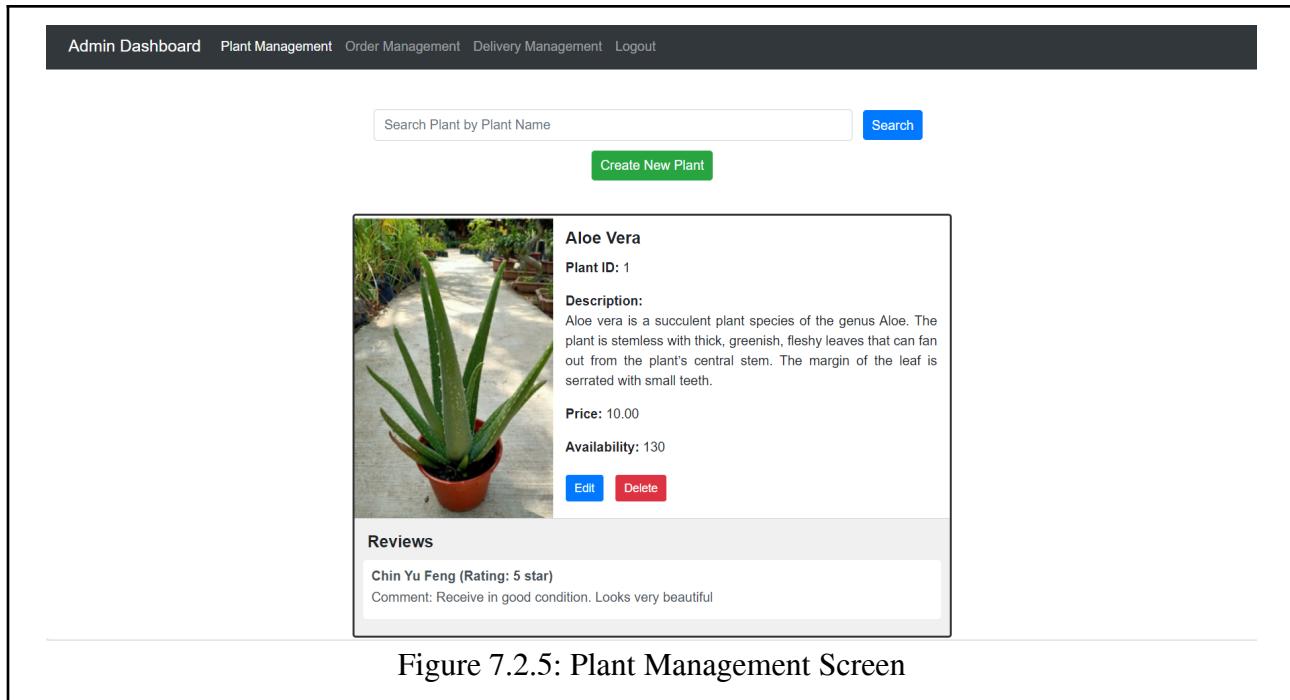
[Back](#) [Update](#)



Figure 7.2.4: Update Plant Screen

7.2.4 An administrator deletes a plant

Figure 7.2.5 below shows the "Plant Management Screen" (same as Figure 7.2.2). From here, the administrator can click on the "Delete" button of a particular plant to delete the plant.



The screenshot shows a web-based plant management system. At the top, there is a dark navigation bar with links: Admin Dashboard, Plant Management, Order Management, Delivery Management, and Logout. Below the navigation bar is a search bar labeled "Search Plant by Plant Name" with a "Search" button next to it. A green "Create New Plant" button is located below the search bar. The main content area displays a potted Aloe Vera plant. The plant's details are listed as follows:

- Name:** Aloe Vera
- Plant ID:** 1
- Description:** 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.
- Price:** 10.00
- Availability:** 130

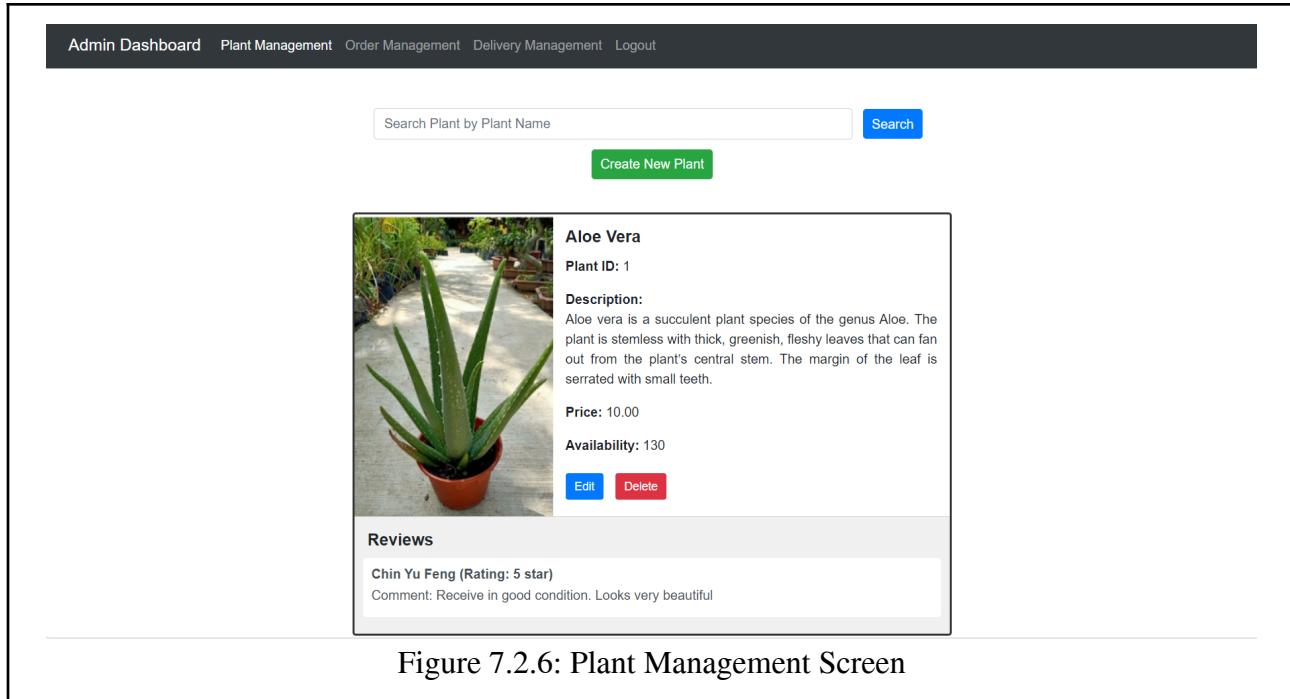
Below the plant details are two buttons: "Edit" (blue) and "Delete" (red). Under the heading "Reviews", there is one entry:

- Chin Yu Feng (Rating: 5 star)**
- Comment: Receive in good condition. Looks very beautiful

Figure 7.2.5: Plant Management Screen

7.2.5 An administrator searches plant

Figure 7.2.6 below shows the “Plant Management Screen” (same as Figure 7.2.2). From here, the administrator can enter a plant name in the search bar and click on the “Search” button to search for matching plants.



The screenshot shows the same web-based plant management system as Figure 7.2.5. The navigation bar, search bar, and "Create New Plant" button are identical. The main content area now displays a search result for "Aloe Vera". The results are shown in a card format:

- Name:** Aloe Vera
- Plant ID:** 1
- Description:** 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.
- Price:** 10.00
- Availability:** 130

Below the plant details are two buttons: "Edit" (blue) and "Delete" (red). Under the heading "Reviews", there is one entry:

- Chin Yu Feng (Rating: 5 star)**
- Comment: Receive in good condition. Looks very beautiful

Figure 7.2.6: Plant Management Screen

7.2.6 An administrator views order details and updates order status

Figure 7.2.7 below shows the “Order Management Screen”. The screen presents each order separately in boxes with key details like order ID, customer’s full name, email, address, and phone number. The number of plants ordered and the total price are also displayed, along with the order date and receive method. From here, the administrator can enter an order ID in the search bar and click on the “Search” button to search for the matching order. The administrator also can select the appropriate order status from the dropdown menu and click on the “Update Status” button to update the selected status for the order.

The screenshot displays the Order Management Screen of a web application. At the top, there is a navigation bar with links: Admin Dashboard, Plant Management, Order Management (which is the active page), Delivery Management, and Logout. Below the navigation bar is a search bar labeled "Search by Order ID" with a "Search" button. The main content area contains two separate order boxes. Each box displays the following information:

OrderID: 44	UserID: 44
Full Name: Zara	Phone Number: 60112382536
Email: zara8061@gmail.com	Address: Gua Musang, 18300, Kelantan
State: Kelantan	Order Date: Feb. 4, 2024
Total Plants: 9	Receive Method: Pickup
Total Price: 270.00	

Below the first order box is a dropdown menu showing "Waiting" and a blue "Update Status" button. The second order box follows a similar structure:

OrderID: 45	UserID: 39
Full Name: Visnu Patel	Phone Number: 60173328497
Email: patel@gmail.com	Address: 1, Lebuh Tunku Kudin 1, 11700 Jelutong, Pulau Pinang
State: Penang	Order Date: Feb. 4, 2024
Total Plants: 20	

Below the second order box is a dropdown menu showing "Waiting" and a blue "Update Status" button. The caption "Figure 7.2.7: Order Management Screen" is centered at the bottom of the screenshot.

7.2.7 An administrator assigns deliveries

Figure 7.2.8 below shows the “Delivery Management Screen”. The screen presents each delivery order separately in boxes with key details like order ID, customer’s full name, email, address, and phone number. The number of plants ordered and the total price are also displayed, along with the order date and receive method. From here, the administrator can enter an order ID in the search bar and click on the “Search” button to search for the matching delivery order. The administrator also can select the delivery man name from the dropdown menu and click on the “Assign” button to assign the delivery order to the selected delivery man.

The screenshot shows a web-based delivery management interface. At the top, there is a navigation bar with links: Admin Dashboard, Plant Management, Order Management, Delivery Management, and Logout. Below the navigation bar is a search bar labeled "Search by Order ID" with a "Search" button. The main content area displays an order record with the following details:

OrderID: 46	UserID: 46
Full Name: Wilson Phang	Phone Number: 60128378358
Email: wilsonphang@gmail.com	Address: Jalan Muzium, 88300 Kota Kinabalu, Sabah
State: Sabah	Order Date: Feb. 5, 2024
Total Plants:	Receive Method: Delivery
Total Price: 62.00	

Below the details, there is a dropdown menu showing "Siti Fatimah" and an "Assign" button.

Figure 7.2.8: Delivery Management Screen

7.3 Guest

7.3.1 Guest main screen.

Figure 7.3.1 depicts the Guest Main Screen of the TLET Nursery Plant Shopping System, designed as the entry point for visitors. Upon arrival, guests are welcomed by a straightforward and clean interface, offering them a choice to either log in to an existing account or sign up for a new one. Below the welcoming message, the interface presents a search bar inviting guests to look for plants by name, enabling a quick and convenient way to navigate the plant selection. The lower section of the screen showcases a visually appealing array of plant images, hinting at the variety of plants available for browsing, which likely serves both as an aesthetic feature and a teaser of the plant catalog they can explore.

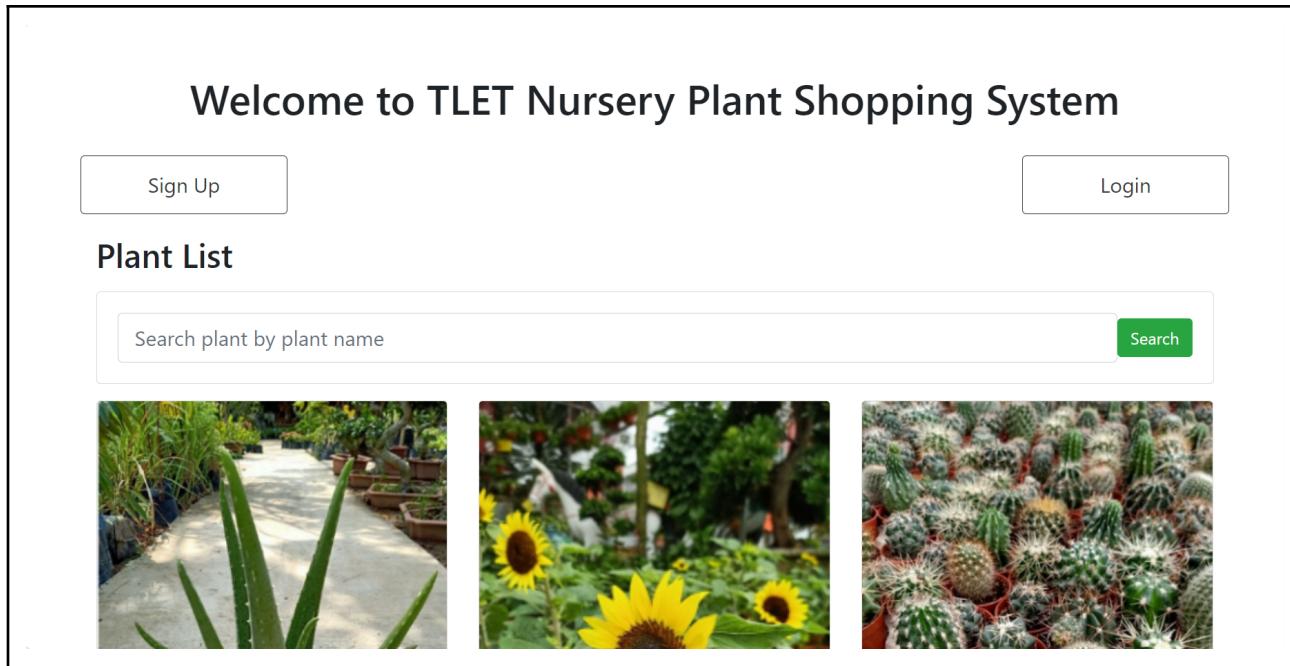


Figure 7.3.1: Guest Main Screen

7.3.2 A guest browses the plant list.

Figure 7.3.2 presents the interface where a guest browses the plant list in the TLET Nursery Plant Shopping System. This screen is neatly organized to showcase a variety of plants, each displayed with an image, the common name of the plant, a unique identification number, and a brief description highlighting its characteristics. For example, the Aloe Vera is described as a stemless succulent with thick, greenish leaves, while the Sunflower Plant is noted for its heat tolerance and bright yellow blooms. A collection of cacti is also displayed, described as having a unique and often unusual appearance. This visual catalog allows guests to effortlessly scroll through the nursery's offerings.

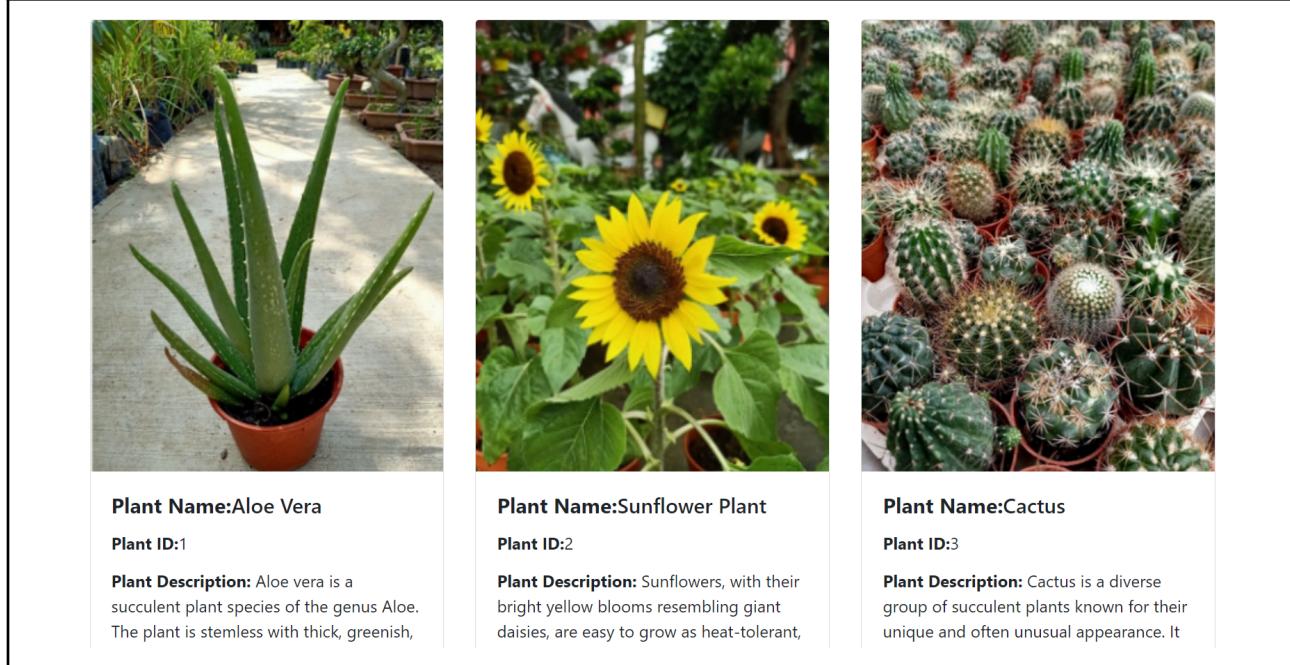


Figure 7.3.2: Guest Browses Plant List Screen

7.3.3 A guest searches a plant by plant name.

Figure 7.3.3 illustrates the "Guest Searches a Plant by Plant Name Screen" on the TLET Nursery Plant Shopping System, where guests can efficiently find specific plants they are interested in. In this scenario, a guest has used the search function to look up "Aloe Vera" by entering the plant name into the search bar. The system responds by displaying the relevant plant, providing a clear image and the plant's name beneath it. This streamlined search feature enhances the user experience by allowing for quick and direct access to a plant's details, without the need to navigate through the entire plant list.

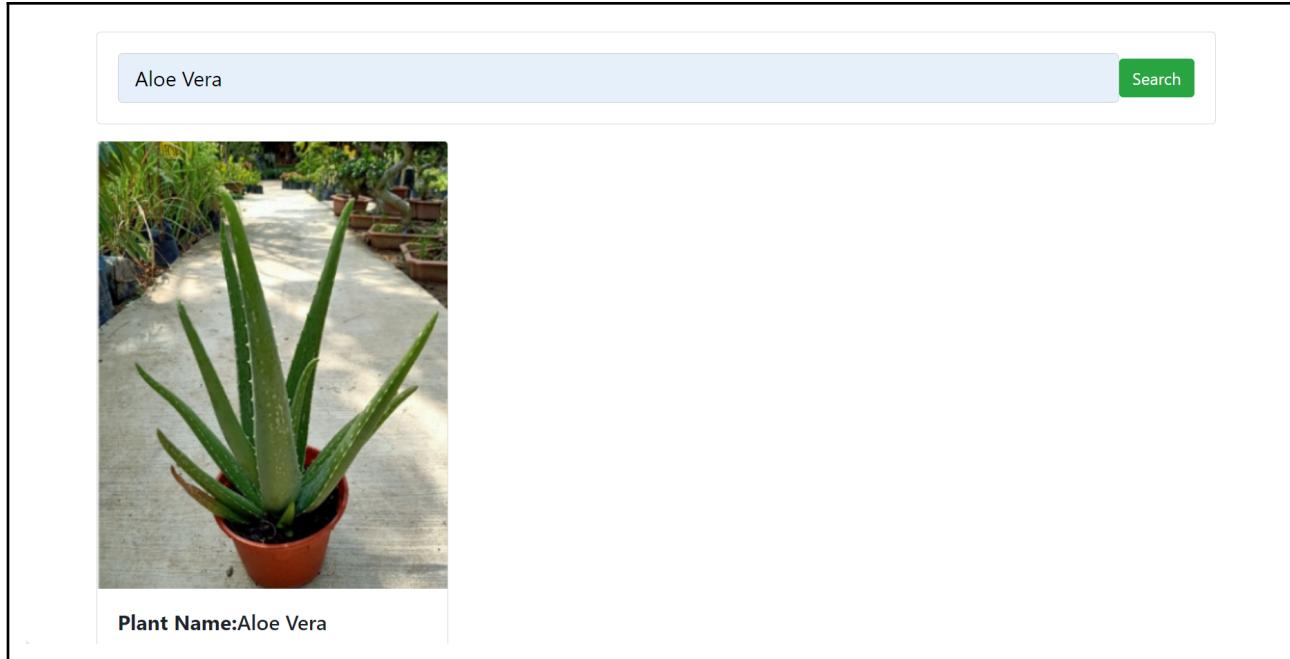


Figure 7.3.3: Guest Searches a Plant by Plant Name Screen

7.3.4 A guest creates a user account.

Figure 7.3.4 displays the "Guest Creates a User Account Screen" from the TLET Nursery Plant Shopping System, a straightforward interface where guests can sign up to become registered users. The screen features a registration form asking for essential information such as full name, password, email address, physical address, state, IC number, and phone number. The guests can also make a decision to become a customer or delivery man by selecting the role on the dropdown box field. Each field is clearly labeled and organized, ensuring that guests can easily fill in their details. This form serves as the first step for guests to create their own account, which will allow them to access additional features of the system, such as purchasing plants or tracking their orders. There is a link to the registration form, ensuring a seamless transition between registering and logging in.

Figure 7.3.5 showcases a login interface for guests attempting to access a user account. The form requests the user to input their email address and password into the respective fields. After entering the necessary information, the user can press the 'Submit' button to log in. If the user does not have an account, there is an option to register by clicking the 'Register here' link provided below the login form.

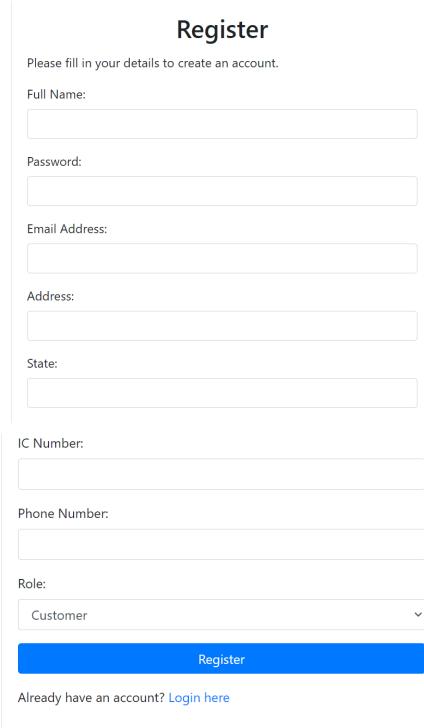
Guest creates a user account.	 <p>The screenshot shows a registration form titled "Register". The instructions say "Please fill in your details to create an account." Below are fields for Full Name, Password, Email Address, Address, State, IC Number, Phone Number, and Role (set to "Customer"). A blue "Register" button is at the bottom, and a link "Already have an account? Login here" is at the bottom right.</p>
-------------------------------	--

Figure 7.3.4: Guest Creates a Account Screen

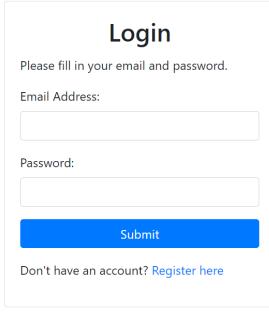
Guest login user account.	 <p>The screenshot shows a login form titled "Login". The instructions say "Please fill in your email and password." Below are fields for Email Address and Password, and a blue "Submit" button. A link "Don't have an account? Register here" is at the bottom right.</p>
---------------------------	---

Figure 7.3.5: Guest Login Screen

7.3.5 A guest views reviews and ratings.

Figure 7.3.6 illustrates a use case where a guest views reviews and ratings on the TLET Nursery Plant Shopping System. After searching for specific plants, the guest is presented with a detailed view of them. For each plant, the system displays a clear image, name, detailed description, price, stock availability, and customer reviews with ratings. This enables the guest to make informed decisions based on the experiences and satisfaction of previous buyers, as seen with the positive reviews for the plants by the customer.

 <p>Plant Name:Aloe Vera Plant ID:1 Plant Description: 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. Price: RM10.00 Availability: 138 Reviews: <ul style="list-style-type: none"> Zara (Rating: 4 star) Comment: Recommended!! </p> <p style="text-align: center;">Add to Cart Add to Wishlist</p>	 <p>Plant Name:Sunflower Plant Plant ID:2 Plant Description: 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. Price: RM30.00 Availability: 98 Reviews: <ul style="list-style-type: none"> Chin Yu Feng (Rating: 5 star) Comment: The flower is beautiful. Thankyou! </p> <p style="text-align: center;">Add to Cart Add to Wishlist</p>	 <p>Plant Name:Cactus Plant ID:3 Plant Description: Cactus is a diverse group of succulent plants known for their unique and often unusual appearance. It typically has fleshy, water-storing stems that are adapted to reduce water loss. Price: RM15.00 Availability: 54 Reviews: <ul style="list-style-type: none"> Zara (Rating: 5 star) Comment: Good service! Thankyou seller ~ </p> <p style="text-align: center;">Add to Cart Add to Wishlist</p>
---	---	---

Figure 7.3.6: Guest Views Reviews and Rating Screen

7.4 Customer

7.4.1 Customer Home Screen

Figure 7.4.1 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.

Plant List Shopping Cart My Orders Account Settings To Review Wishlist Logout Welcome, LIM CAI QING

Welcome to Customer Dashboard

Plant List

Search plant by plant name








Plant Name:Aloe Vera
Plant ID:1

Plant Description: 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.

Price: RM10.00

Availability: 140

Reviews:

- LIM CAI QING (Rating: 5 star)
Comment: good!
- LIM CAI QING (Rating: 2 star)
Comment: Received a different pot size than expected, but the plants were healthy.

Plant Name:Sunflower Plant
Plant ID:2

Plant Description: 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.

Price: RM30.00

Availability: 97

Reviews:

- LIM CAI QING (Rating: 3 star)
Comment: Affordable prices

Plant Name:Cactus
Plant ID:3

Plant Description: Cactus is a diverse group of succulent plants known for their unique and often unusual appearance. It typically has fleshy, water-storing stems that are adapted to reduce water loss.

Price: RM15.00

Availability: 49

Reviews:

- No reviews yet.

Figure 7.4.1: Customer Home Screen

7.4.2 Customer Changes Their Password Screen

Figure 7.4.2 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 7.4.1, 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 7.4.3, 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 7.4.2, completing the password change process.

The screenshot shows the 'Account Settings' page. At the top, there is a navigation bar with links: Plant List, Shopping Cart, My Orders, Account Settings, To Review, Wishlist, and Logout. Below the navigation bar, the title 'Account Settings' is centered. Underneath the title, there is a section titled 'Profile Details' containing the following information:

- Full Name:** LIM CAI QING
- Email:** caiqinglim0327@gmail.com
- Phone Number:** 60143001623
- Address:** 35, Jalan BPU6 Taman Indah, Puchong
- State:** Terengganu
- IC No:** 700918100931

At the bottom of this section are two buttons: 'Edit Profile' (blue) and 'Change Password' (blue).

Figure 7.4.2: Account Settings Screen

The screenshot shows the 'Change Password' page. At the top, there is a navigation bar with links: Plant List, Shopping Cart, My Orders, Account Settings, To Review, Wishlist, and Logout. Below the navigation bar, the title 'Change Password' is centered. The page contains the following fields:

- 'Old password:' followed by an input field.
- 'New password:' followed by an input field.
- A list of password requirements:
 - Your password can't be too similar to your other personal information.
 - Your password must contain at least 8 characters.
 - Your password can't be a commonly used password.
 - Your password can't be entirely numeric.
- 'New password confirmation:' followed by an input field.

At the bottom of the page is a blue 'Confirm' button.

Figure 7.4.3: Change Password Screen

7.4.3 Customer Can Edit Profile Details Screen

Figure 7.4.4 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 7.4.1, where the customer selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 7.4.2, where the customer's details are listed. By selecting the 'Edit Profile' button, the customer is directed to the interface shown in Figure 7.4.4, 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 7.4.2 without making any updates.

The screenshot shows a web-based application interface titled 'Edit Profile'. At the top, there is a navigation bar with links: Plant List, Shopping Cart, My Orders, Account Settings, To Review, Wishlist, and Logout. Below the navigation bar, the main content area has a title 'Edit Profile'. It contains several input fields for personal information:

- Full Name: LIM CAI QING
- Email: caiqinglim0327@gmail.com
- Phone Number: 60143001623
- Address: 35, Jalan BPU6 Taman Indah, Puchong
- State: Terengganu
- IC No: 700918100931

At the bottom left is a blue 'Back' button, and at the bottom right is a blue 'Confirm' button.

Figure 7.4.4: Edit Profile Details Screen

7.4.4 Customer Views Added Plants In The Shopping Cart Screen

Figure 7.4.5 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 7.4.1, 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 7.4.5, 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.

The screenshot shows a shopping cart interface. At the top, there's a navigation bar with links: Plant List, Shopping Cart, My Orders, Account Settings, To Review, Wishlist, and Logout. Below the navigation bar, the title "Shopping Cart" is centered. On the left, there's a photograph of a sunflower plant. To its right, product details are listed: Plant Name :Sunflower Plant, Plant ID: 2, Price :RM30.00, and Quantity : 1. A "Remove" button is located below the quantity input field. At the bottom of the cart summary, it says "Total: RM30.00". Outside the main cart box, the text "Total Price: RM30.00" is displayed, along with a blue "Checkout" button.

Figure 7.4.5: Views Added Plants In The Shopping Cart Screen

7.4.5 Customer Adds Plants To The Shopping Cart Screen

Figure 7.4.6 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 7.4.1, where the customer selects the 'Plant List' button. This action takes them to the "Adds Plants To The Shopping Cart Screen" displayed in Figure 7.4.6 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 7.4.6 to continue browsing.

			
Plant Name: Live Rose Plant	Plant Name: Hibiscus Plant	Plant Name: Adenium obesum	
Plant ID: 4	Plant ID: 5	Plant ID: 6	
Plant Description: Roses, with their unique combination of thorny stems and fragrant blossoms, are often prized as a symbol of achievement, completion, and perfection: After having met the long stems with the thorns, you are rewarded with a flower of great beauty.	Plant Description: The hibiscus plant is an annual or perennial herbaceous plant with trumpet-shaped flowers. Its flowers can reach nearly 10 inches in diameter at maturity and it is a perfect plant for warm climates.	Plant Description: Adenium obesum commonly called desert rose features a swollen woody stem holding lush blossoms of vibrant pink, coral, or red hues colour. It is a widely used ornamental outdoor plant.	
Price: RM35.00	Price: RM25.00	Price: RM30.00	
Availability: 250	Availability: 80	Availability: 130	
Reviews:	Reviews:	Reviews:	
• No reviews yet.	• No reviews yet.	• No reviews yet.	
Add to cart	Add to Wishlist	Add to cart	Add to Wishlist

Figure 7.4.6: Adds Plants To The Shopping Cart Screen

7.4.6 Customer Deletes Plants From The Shopping Cart Screen

Figure 7.4.7 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 7.4.1, where the customer selects the 'Shopping Cart' button. This action takes them to the "Deletes Plants From The Shopping Cart Screen" displayed in Figure 7.4.7 (same as Figure 7.4.5), 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 7.4.7.

The screenshot shows a shopping cart interface. At the top, there is a navigation bar with links: Plant List, Shopping Cart, My Orders, Account Settings, To Review, Wishlist, and Logout. Below the navigation bar, the title "Shopping Cart" is displayed. A large image of a sunflower plant is shown. To its right, the product details are listed: Plant Name :Sunflower Plant, Plant ID: 2, Price :RM30.00, and a Quantity input field set to 1. A red "Remove" button is located below the quantity field. At the bottom of the cart area, it says Total: RM30.00. Outside the main cart box, the text "Total Price: RM30.00" is displayed, followed by a blue "Checkout" button.

Figure 7.4.7: Deletes Plants From The Shopping Cart Screen

7.4.7 Customer Searches a Plant by Plant Name Screen

Figure 7.4.8 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 7.4.1, where the customer selects the 'Plant List' button. This action takes them to the "Searches a Plant by Plant Name Screen" displayed in Figure 7.4.8 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.

Plant List

Cactus X Search




Plant Name:Cactus

Plant ID:3

Plant Description: Cactus is a diverse group of succulent plants known for their unique and often unusual appearance. It typically has fleshy, water-storing stems that are adapted to reduce water loss.

Price: RM15.00

Availability: 49

Reviews:

- No reviews yet.

Add to cart Add to Wishlist

Figure 7.4.8: Searches a Plant by Plant Name Screen

7.4.8 Customer Makes a Payment Screen

Figure 7.4.9 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 7.4.1, 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 7.4.9, 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 7.4.11. 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 7.4.6, to continue shopping or review their orders.

Plant List Shopping Cart My Orders Account Settings To Review Wishlist Logout

Checkout

Pickup - Please pickup within 14 days at the location below:

Location: TLET Company, Jalan Raja, Subang Jaya, Selangor

Delivery

Continue

Figure 7.4.9: Checkout Screen

Plant List Shopping Cart My Orders Account Settings To Review Wishlist Logout

Delivery Details

Full name: LIM CAI QING
Delivery Address: 35, Jalan BPU6 Taman Indah, Puchong
Plant Price: RM30.00
Shipping Fee: RM7.00
Total Cost: RM37.00

Proceed to Payment

Figure 7.4.10: Delivery Details Screen

The screenshot shows a 'Make Payment' screen with the following details:

- Header navigation bar: Plant List, Shopping Cart, My Orders, Account Settings, To Review, Wishlist, Logout.
- Section title: Make Payment
- Text: Receive Item Via: Pickup
- Text: Total Cost (including shipping fee if applicable): RM30.00
- Section: Payment Method:
 - Credit/Debit Card
 - FPX (Internet Banking)
 - Touch 'N Go E-Wallet
- Button: Confirm

Figure 7.4.11: Make Payment Screen

7.4.9 Customer Views Order Details of Ongoing Orders Screen

Figure 7.4.12 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 7.4.1. By clicking the 'My Orders' button, customers are led to the screen shown in Figure 7.4.12, 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.

The screenshot shows the 'My Orders' screen with the following details:

Order ID	Order Status	Order Date	Total Price	Plant Name	Quantity	Price
42	Waiting	Feb. 5, 2024	RM17.00	Aloe Vera	1	RM10.00
43	Completed	Feb. 5, 2024	RM10.00	Aloe Vera	1	RM10.00

Figure 7.4.12: Views Order Details of Ongoing Orders Screen

7.4.10 Customer Leaves Reviews and Ratings Screen

Figure 7.4.13 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 7.4.1. By clicking the 'To Review' button, customers are led to the screen shown in Figure 7.4.13, 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.

The screenshot shows a web page titled 'To Review'. At the top, there is a navigation bar with links: Plant List, Shopping Cart, My Orders, Account Settings, To Review, Wishlist, and Logout. Below the navigation bar, the title 'To Review' is displayed. On the left, there is an image of an Aloe Vera plant in a pot. To the right of the image, the plant's name is listed as 'Plant Name: Aloe Vera'. Below this, there is a 'Rating:' section with a dropdown menu containing the placeholder 'Rate the plant'. Underneath the rating section is a 'Review:' section with a text input field containing the placeholder 'Write your review here...'. At the bottom right of the form is a blue 'Submit' button.

Figure 7.4.13: Leaves Reviews and Ratings Screen

7.4.11 Customer Manages a Wishlist Screen

Figure 7.4.14 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 7.4.1. By clicking the 'Wishlist' button, customers are led to the screen shown in Figure 7.4.14. 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.

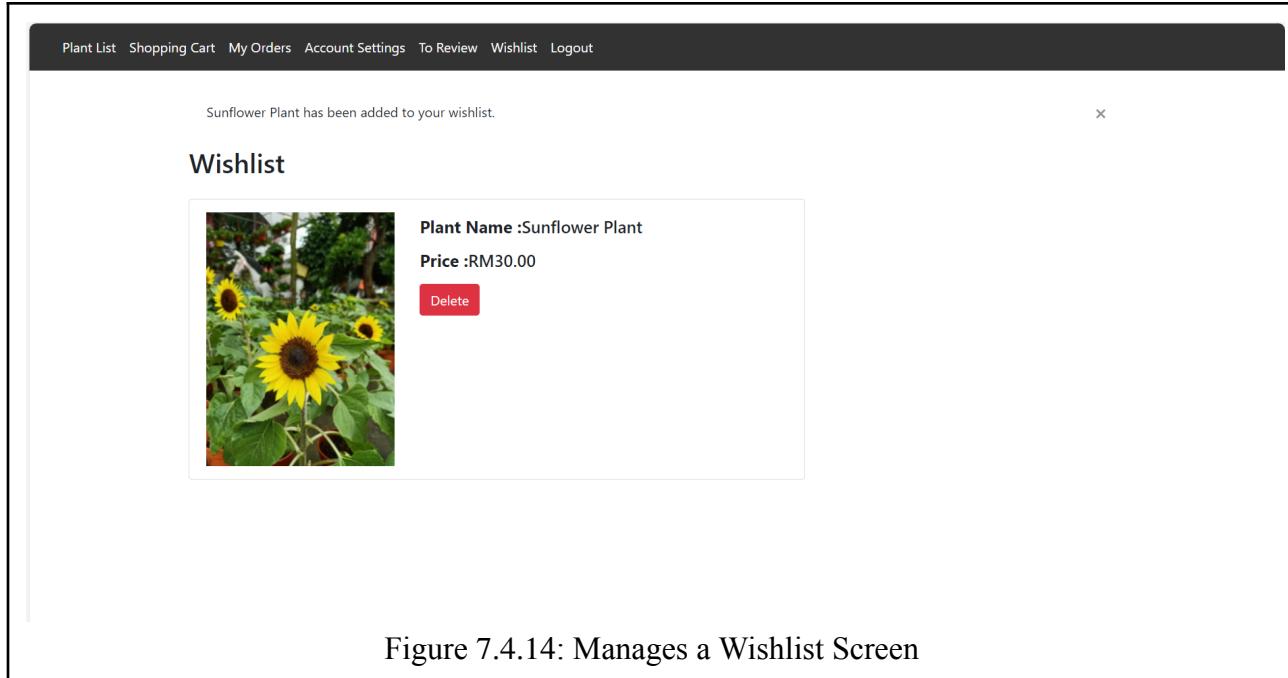
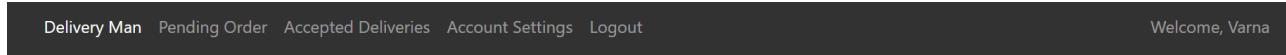


Figure 7.4.14: Manages a Wishlist Screen

7.5 Delivery Man

7.5.1 Delivery Man Home Screen

Figure 7.5.1 displays the "Delivery Man Home Screen" of the TLET Nursery Plant Shopping System. Upon login, the delivery man will be presented with a home screen shown in Figure 7.5.1 featuring a navigation bar with options for 'Pending Order', 'Accepted Deliveries', 'Account Settings', 'Logout' button and a message displaying "Welcome, username". Below the navigation, a welcoming message marks the entry to the delivery man dashboard.



Welcome to Delivery Man Dashboard

Figure 7.5.1: Delivery Man Home Screen

7.5.2 Delivery Man Creates a User Account Screen

Figure 7.5.3 shows the 'Create User Account Screen', which illustrates the process for delivery men to create a new account on the TLET Nursery Plant Shopping System. Upon accessing the system, the delivery man will first encounter the guest home screen in Figure 7.5.2, which is also the "TLET Nursery Plant Shopping System Screen" before creating a user account. The delivery man can proceed by clicking on the "Sign Up" button and this action takes the delivery man to the 'Creates User Account Screen' shown in Figure 7.5.3, where the delivery man fills out the form with personal and contact details. Upon clicking the "Submit" button, the system validates the input and saves the new user details to persistent storage. Successful registration is confirmed with a message. Delivery men who have an account can quickly get started by clicking the "Login" button, which takes them to a Login page in Figure 7.5.4.

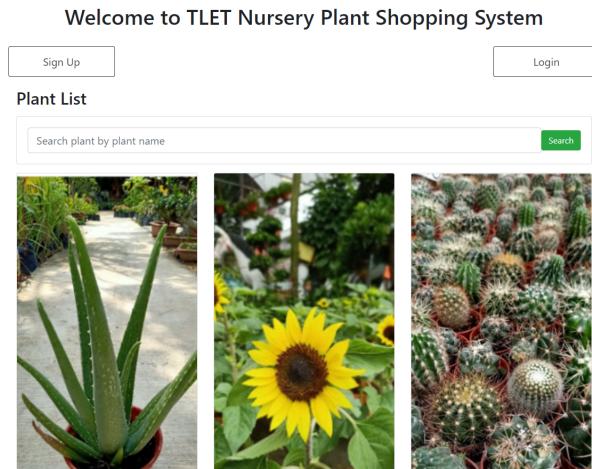


Figure 7.5.2: Guest Home Screen

Register

Please fill in your details to create an account.

Full Name:

Password:

Email Address:

Address:

State:

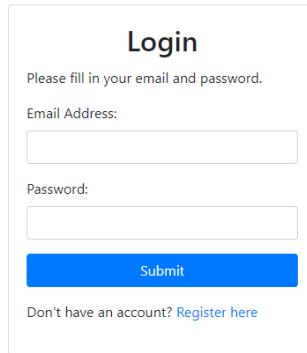
IC Number:

Phone Number:

Role:

Already have an account? [Login here](#)

Figure 7.5.3: Create User Account Screen



The image shows a login form titled "Login". It instructs the user to "Please fill in your email and password." There are two input fields: "Email Address:" and "Password:", each with a corresponding input box. Below the input boxes is a blue "Submit" button. At the bottom of the form, there is a link "Don't have an account? [Register here](#)".

Figure 7.5.4: TLET Nursery Plant Shopping System Login Screen

7.5.3 Delivery Changes Their Password Screen

Figure 7.5.6 shows the "Change Password Screen", which presents the interface for delivery man to update their password on the TLET Nursery Plant Shopping System. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 7.5.1, where the delivery man selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 7.5.5, where the delivery man details are listed. By selecting the 'Change Password' button, the delivery man is directed to the screen shown in Figure 7.5.6, where they are required to enter their current 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 delivery man is taken back to the "Account Settings Screen" shown in Figure 7.5.5, completing the password change process.

Full Name: Siti Fatimah
 Email: sitifatimah@gmail.com
 Phone Number: 60147894163
 Address: 35, Jalan BPU6 Taman Indah, Puchong
 State: Selangor
 IC No: 920504018888

Edit Profile Change Password

Figure 7.5.5: Account Settings Screen

Current Password:

New Password:

Confirm

Figure 7.5.6: Change Password Screen

7.5.4 Delivery Man Can Edit Profile Details Screen

Figure 7.5.7 shows 'Edit Profile Details Screen', where a delivery man can update their personal information on the TLET Nursery Plant Shopping System. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 7.5.1, where the delivery man selects the 'Account Settings' button. This action takes them to the "Account Settings Screen" displayed in Figure 7.5.5, where the delivery man's details are listed. By selecting the 'Edit Profile' button, the delivery man is directed to the screen shown in Figure 7.5.7, featuring fields to modify the delivery man's personal information. Changes are submitted with the 'Confirm' button, and the system displays a confirmation message. If the delivery man chooses not to update the personal

information, clicking the 'Back' button redirects them to the "Account Settings Screen" in Figure 7.5.5, without making any updates.

The screenshot shows the 'Edit Profile' screen. At the top, there is a navigation bar with links: Delivery Man, Pending Order, Accepted Deliveries, Account Settings, and Logout. Below the navigation bar, the title 'Edit Profile' is centered. The main area contains a form with the following fields:

- Full Name: Siti Fatimah
- Email: sitifatimah@gmail.com
- Phone Number: 60147894163
- Address: 35, Jalan BPU6 Taman Indah, Puchong
- State: Selangor
- IC Number: 920504018888

At the bottom of the form are two buttons: 'Back' and 'Confirm'.

Figure 7.5.7: Edit Profile Screen

7.5.5 Delivery Man Accepts a Delivery Order Screen

Figure 7.5.8 shows the "Accepts a Delivery Order Screen" from the TLET Nursery Plant Shopping System, used by the delivery man to manage incoming orders. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 7.5.1, where the delivery man selects the "Pending Orders" button. This action takes them to the "Accepts a Delivery Order Screen" displayed in Figure 7.5.8. On this screen, the delivery man can view all pending orders with detailed information including the Delivery Order ID, customer names, customer IDs, contact numbers, and delivery addresses. They have the option to either accept or reject these orders directly from this interface. Once an order is accepted, it is marked as out of delivery. If the Delivery Man clicks the "Reject" button, the system removes the delivery order from the list of pending orders.

The screenshot shows a "Pending Orders" section with three items:

- Delivery Order ID: 48**
Customer Name: Chin Yu Feng
Customer ID: 40
Phone Number: 60128457865
Delivery address: 20, Taman Maju, Section 3/2a, Cheras
- Delivery Order ID: 50**
Customer Name: Zara
Customer ID: 41
Phone Number: 60112382536
Delivery address: Gua Musang, 18300, Kelantan
- Delivery Order ID: 51**
Customer Name: Zara
Customer ID: 41
Phone Number: 60112382536
Delivery address: Gua Musang, 18300, Kelantan

Figure 7.5.8: Accepts a Delivery Order Screen

7.5.6 Delivery Views Accepted Deliveries and Confirm Delivery Screen

Figure 7.5.9 presents the "Accepts a Deliveries and Confirm Delivery Screen" of the TLET Nursery Plant Shopping System. The process is initiated from the "Delivery Man Home Screen" as shown in Figure 7.5.1, where the delivery man selects the 'Accepted Deliveries' button. This action takes them to the "Accepts a Deliveries and Confirm Delivery Screen" displayed in Figure 7.5.9 which allows delivery men to view a list of all plant deliveries they have accepted, complete with details such as delivery order ID, customer names, phone numbers, and delivery addresses. After the successful delivery of an order, the delivery man can confirm the completion of delivery by clicking the "Confirm Delivery" button provided alongside each order. This confirmation updates the order's status to "Completed" in the system's database, ensuring that the delivery process is accurately tracked.

The screenshot shows an "Accepted Deliveries" section with one item:

- Delivery OrderID: 46**
Customer Name: Zara
Phone Number: 60112382536
Delivery address: Gua Musang, 18300, Kelantan

Figure 7.5.9: Accepts a Deliveries and Confirm Delivery Screen

8 Conclusion

8.1 Completion of Software

We have successfully completed our nursery plant shopping system through cohesive cooperation among our team members, addressing all possible error detections thoroughly. For Guests, we ensured that the browsing of plant lists, and account creation process were smooth and user-friendly. This foundational step allows Guests to seamlessly transition into Customers upon account creation. For Customers, the requirements were rigorously tested to ensure that all needs are met before proceeding with the integration.

For Administrators, who manage the plants and orders, their specific requirements underwent extensive testing with numerous test data to guarantee that all functionalities are up to standard before integration. Similarly, delivery men's functionalities were also validated to ensure that they meet all necessary requirements before their module's integration into the system. Therefore, our nursery plant shopping system is fully completed, with all actor-specific requirements thoroughly tested and fulfilled.

8.2 Software Quality Assurance

To guarantee high software quality, our team starts by thoroughly understanding the project requirements and identifying any challenges we need to overcome. We then focus on resolving any issues during the design phase to improve the overall quality of our software.

At the beginning of the project, we create a detailed plan that outlines how we will manage quality and handle any changes that may arise. Throughout the development process, we perform several tests and reviews with diverse data sets to ensure our software meets all initial requirements.

Additionally, we keep a precise record of any issues we find. This allows us to quickly fix them and ensures that our final product aligns with what was originally planned. We also maintain open lines of communication within our team, sharing updates and flagging any issues as they arise. This team effort helps us overcome obstacles efficiently and keeps our software on track.

8.3 Group Collaboration

We conducted weekly meetings to discuss our project and monitor each member's progress, ensuring everyone contributed effectively to their assigned tasks. Besides, we assisted each other in debugging the system, conducting code research, understanding syntax, and reviewing each other's work.

During the development of our nursery plant shopping system, it became evident that successful collaboration among team members was crucial for achieving a comprehensive and flawless execution of the project.

8.4 Problems Encountered

Throughout the project development process, we encountered several significant challenges, particularly in the coding aspects, which required a lot of research and learning. Even though the Django framework came with a lot of documentation, we found it was not enough to fully grasp the syntax and how everything worked. Since it was our first time using Django, Python, HTML, CSS and Bootstrap, the lack of prior experience posed a big challenge to us. We had to spend extra time studying and familiarizing ourselves with these languages and frameworks, which slowed down our progress.

We faced two big issues during the development of our project. Firstly, we struggled with time limitations when trying to add a feature that would show only the most recent review under each plant details box, with the option to view earlier reviews upon clicking. Secondly, despite our best efforts, we encountered obstacles in integrating a third-party payment gateway into our system. As a result, the current state of our system only allows for the display of a success message when customers select a payment method, without the actual processing of payments.

8.5 Remarks/Comments

This assignment proved challenging as it was our first time completing a Python project. We were not familiar with the basics of the language, and it took us some time to adjust to its syntax. We had to do a lot of research on Python, Django, HTML, CSS, and Bootstrap to understand how to use them.

Despite these challenges, this project gave us valuable experience in web application development, which will benefit us in the future. Through this process, we learned that spending too much time on one problem could mess up our initial plan. Therefore, it is crucial to solve problems quickly and stay adaptable.

Additionally, we realised the importance of teamwork. Teamwork helped us tackle issues more effectively, bringing together different viewpoints and ideas, which saved time and made solving problems easier.