

---

# Project Report

---

## Mecha Sales System by ellipsis

### Team Members:

Adil Alam	i170127 CS-E
Shaheer Hassan	i170271 CS-E
Omar Ahmed	i170040 CS-E

## Table of Contents

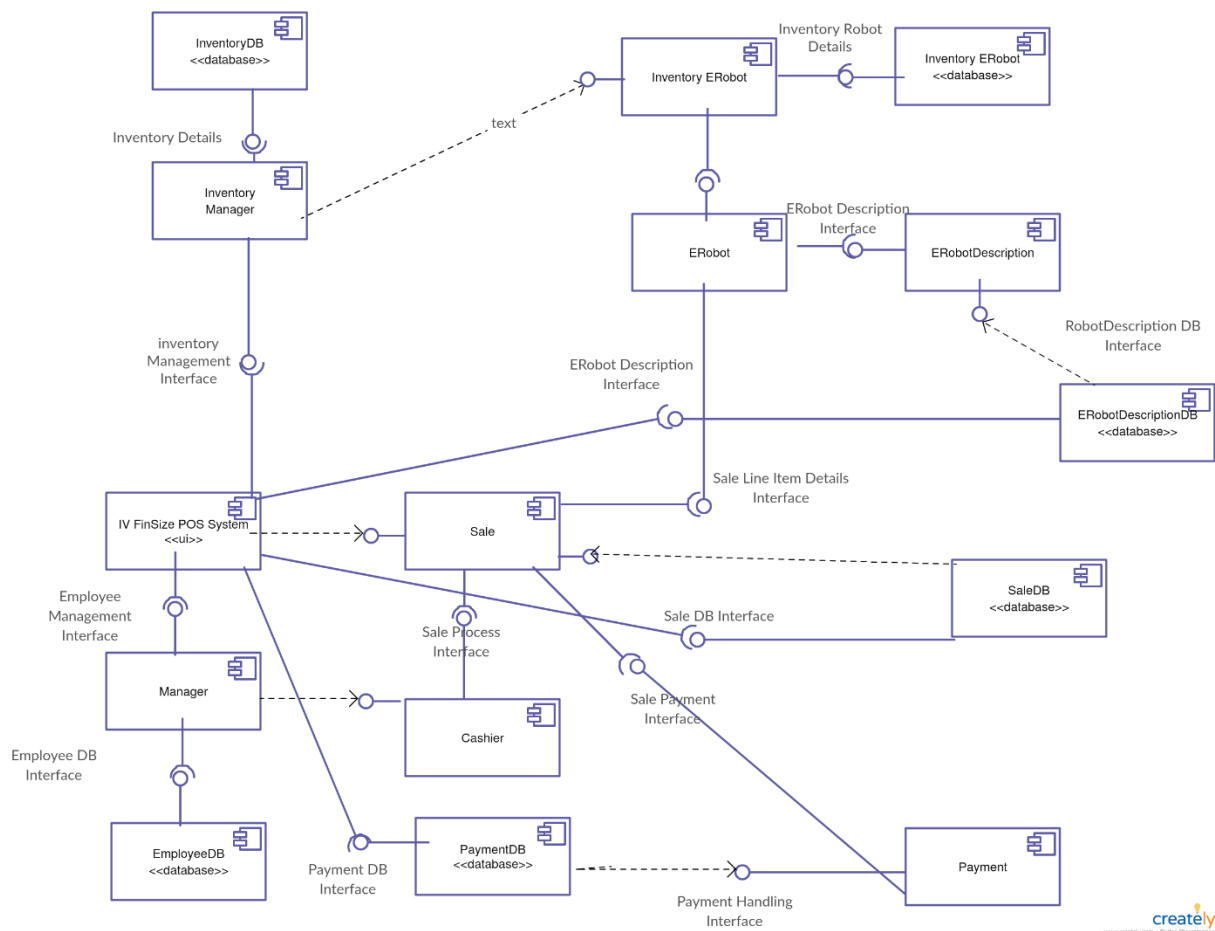
<b>Introduction</b>	<b>2</b>
<b>Component Diagram</b>	<b>3</b>
<b>System Architecture</b>	<b>4</b>
<b>Product Backlog</b>	<b>5</b>
<b>Burndown Chart</b>	<b>16</b>
<b>Trello Snapshots</b>	<b>17</b>
<b>Project GUIs</b>	<b>19</b>
<b>Sprint 1 (UI and Testing snapshots)</b>	<b>19</b>
<b>Sprint 2 (UI and Testing snapshots)</b>	<b>37</b>
<b>Sprint 3 (UI and Testing snapshots)</b>	<b>54</b>

## Introduction

Mecha Sales System under discussion is a real-time implementation of Mecha Sales System for electronic robot stores. It is a desktop application. This system mainly applies in the areas of customer purchase system, staff management, order management, inventory management related to electronic robotics, and generating reports of customers using business intelligence algorithms. Our project is a point of sale system for electronic robots' stores. This system targets the customers and delivery domain of sales and purchases, and it is flexible, reliable to be applied to a variety of robotics in stores. Mecha Sales System is a computer-based solution for electronic robot stores seeking an alternative to compete with market shareholders, providing a fast and efficient way of entering and storing data as well as providing online ordering services to customers.

Mecha Sales System can also be used to automate the process of manually maintaining the records related to the subject of maintaining the stock and cash flow in files. It provides easy management of production, clients, and accounting. It allows them to switch from manual operations to automation, seeking ease, efficiency and providing quality services to their customers. Mecha Sales System can be implemented in all types of stores by expanding the implementation at higher levels such as marts and factory containers.

## Component Diagram



### Components:

- 1) **Sale:** Provide all interface for managing sale and require interface is saleDB.
- 2) **Cashier:** Provide all interface for processing sale and require interface is sale interface.
- 3) **Manager:** Provide all interface for managing employees and require interface is employeeDB.
- 4) **ERobot:** Provide interface for inventory of robots and require interface is ERobotDB.
- 5) **InventoryManager:** Provide interface for managing inventory of robots and require interface is ERobotDB.

### Types of Connectors Used:

- 1) Stream
- 2) Shared Database
- 3) Message Bus

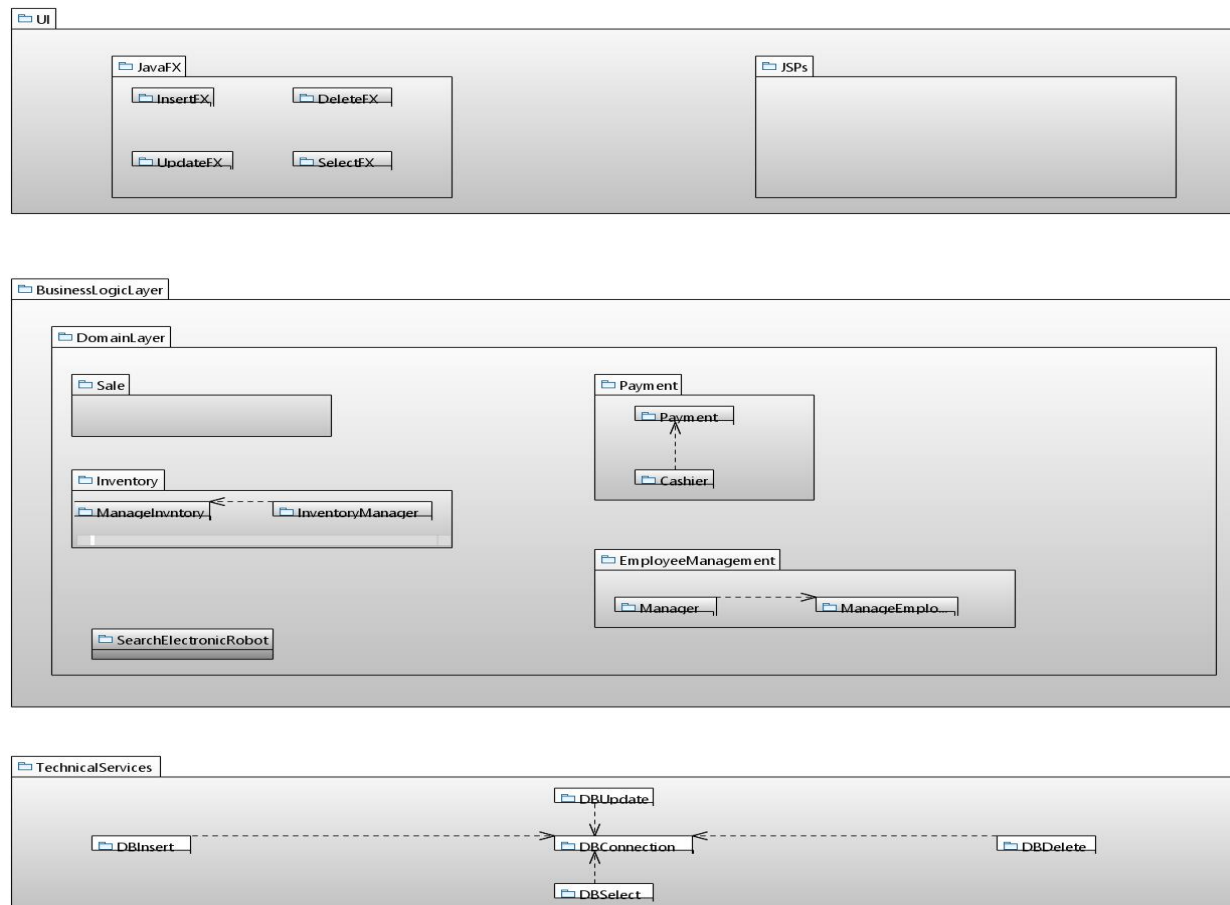
## System Architecture

The System Architecture will be layered Architecture, as it divides the data, interfaces components separately, UI layer deals with user interface like buttons, widgets etc. While domain layer deals with data models, and technical layer deals with database etc.

The main reason to use the Layered architecture is ‘Separation of Concerns’. UI layer have not the logic to make sale or add employee. The responsibility of each layer is separate, no layer interrupts the work of other layer.

User Interface will be function using Java API known as JavaFX major use for desktop app development, as our system does not require online availability so preferably making a desktop app make it portable and to support multiple terminals

### Architecture Diagram:



## Product Backlog

Sprint-user Story	Tasks	Status
<p><b>Sprint 1:</b></p> <p><b>User Story 1: <u>Manage Employee</u></b></p> <p><b>Description:</b>  <i>As a manager, I want to manage employees so that I can keep record of the currently working employee's information.</i></p> <p><b>Acceptance Criteria:</b>            I know I am done with managing employees when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> After being authenticated, I can manage employee's information by pressing the widget "Register Employee", "Edit Employee", "View Employee", and "Delete Employee".</li> <li>- <b>Failure:</b> Error message with "Error occurred" will be pop up.</li> </ul> <p><b>Sub-User Story 1.1: Register Employee</b></p> <p><b>Description:</b>  <i>As a manager, I want to register employees so that I can keep records of the employees and information related to them.</i></p> <p><b>Acceptance Criteria:</b>            I know I am done with registering employees when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> Valid Employee Information provided with the proper format, along with the "Employee Registered" message popped up after pressing the widget "Register Employee".</li> <li>- <b>Failure:</b> Error message with showing "Invalid format" will pop up.</li> </ul>	<p><b><u>1.1 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create Database Table Name Employee</li> <li>- Create UI for registering employees</li> <li>- Write code to enter data in database</li> <li>- Integrate UI with code</li> <li>- Link Register Employee tab with managers tab.</li> </ul>	<p>All the tasks of sub-user story 1.1 are completed within the planned time.</p>

<p><b>Sub-User Story 1.2:</b> Update Employee Details</p> <p><b>Description:</b> <i>As a manager, I want to update employees' details like name, age, grade, etc. so that I can keep updated information on employees.</i></p> <p><b>Acceptance Criteria:</b> I know I am done with updating the details of employees when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> "Employee Details Updated" message pops up after pressing "Update"</li> <li>- <b>Failure:</b> "Error Occurred" message pops after pressing "Update".</li> </ul>	<p><b><u>1.2 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create UI for updating employee details information</li> <li>- Write code for updating employees' information in database.</li> <li>- Link UI with code.</li> <li>- Link Update Employee tab with Managers tab.</li> </ul>	<p>All the tasks of sub-user story 1.2 are completed within the planned time.</p>
<p><b>Sub-User Story 1.3:</b> Remove Employee Account</p> <p><b>Description:</b> <i>As a manager, I want to remove employees' accounts so that I can keep my system updated about current employees and employees that had left.</i></p> <p><b>Acceptance Criteria:</b> I know I am done with Deleting Employees Account when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> "Employee Account Deleted" message pops up after pressing "Delete Account".</li> <li>- <b>Failure:</b> "Employee Account Deleted" message pops up after pressing "Delete Account".</li> </ul>	<p><b><u>1.3 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create UI for deleting employee account.</li> <li>- Write code to update employee state from current to left in database.</li> <li>- Link UI with code.</li> <li>- Link deleting employee account tab with managers tab.</li> </ul>	<p>All the tasks of sub-user story 1.3 are completed within the planned time.</p>
<p><b>Sub-User Story 1.4:</b> View Employee Details</p> <p><b>Description:</b> <i>As a manager, I want to view employee details like name, age, grade so that I am able to know which employee is currently working and information related to them.</i></p> <p><b>Acceptance Criteria:</b> I know I am done with viewing employee details when:</p>	<p><b><u>1.4 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create UI for viewing employee's details.</li> <li>- Write code to query employee's information from database.</li> <li>- Link UI and code.</li> </ul>	<p>All the tasks of sub-user story 1.4 are completed within the planned time.</p>

<ul style="list-style-type: none"> <li>- <b>Success:</b> Valid User Information popped up.</li> <li>- <b>Failure:</b> Invalid information or no information is popped up.</li> </ul>	<ul style="list-style-type: none"> <li>- Link view employees' detail tab with managers tab.</li> </ul>	
<p><b>Sub-User Story 1.5:</b> Employee Login/Authentication</p> <p><b>Description:</b></p> <p><i>As an employee, I want to login to the system so that I can access the system functions like make sales, generate a report, etc. to facilitate the customers.</i></p> <p><b>Acceptance Criteria:</b></p> <p>I know I am done with authentication and get system accessories when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> Valid User Information provided with the proper format, press login button, message with "Authenticated and prevailing Access" will pop up.</li> <li>- <b>Failure:</b> Invalid User name error message pops up, or error message "Cannot Authenticated" pop up.</li> </ul>	<p><b>1.5 Tasks:</b></p> <ul style="list-style-type: none"> <li>- Creating UI for authenticate an employee</li> <li>- Write code to authenticate employees name and password from database</li> <li>- Link UI with code</li> <li>- Link authenticate employee tab with employee tab.</li> </ul>	<p>All the tasks of sub-user story 1.5 are completed within the planned time.</p>
<p><b>User Story 2: <u>Manage Electronic Robots</u></b></p> <p><b>Description:</b></p> <p><i>As an inventory manager, I want to manage electronic robots so that I can keep record of the currently present items in the stock.</i></p> <p><b>Acceptance Criteria:</b></p> <p>I know I am done with managing employees when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> After being authenticated as inventory manager, I can manage electronic robots information by pressing the widget "Add E Robot", "Edit E Robot", "View E Robot", and "Delete E Robot".</li> <li>- <b>Failure:</b> Error message with "Error occurred" will be pop up.</li> </ul> <p><b>Sub-User Story 2.1:</b> Add Electronic Robot</p> <p><b>Description:</b></p>	<p><b>2.1 Tasks:</b></p> <ul style="list-style-type: none"> <li>- Create Database Table Name "Electronic Robot"</li> <li>- Create UI for adding electronic robot.</li> <li>- Write code to add new electronic robot in database.</li> <li>- Link UI with code</li> <li>- Creating UI for Inventory Manager</li> <li>- Link adding new electronic robot tab with inventory manager tab.</li> </ul>	<p>All the tasks of sub-user story 2.1 are completed within the planned time.</p>



<p><i>As an Inventory Manager, I want to add electronic robots so that I can keep records of new electronic robots.</i></p> <p><b>Acceptance Criteria:</b></p> <p>I know I am done with adding electronic robots when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> “Electronic Robots Added” message pops up.</li> <li>- <b>Failure:</b> “Error Occurred” message with details of the error pops up.</li> </ul>		
<p><b>Sub-User Story 2.2:</b> Update Electronic Robot Description</p> <p><b>Description:</b></p> <p><i>As an inventory manager, I want to update the Electronic Robot description so that I can keep a record of Electronic Robots updated.</i></p> <p><b>Acceptance Criteria:</b></p> <p>I know I am done with updating the description of Electronic Robot.</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> Valid User Information provided with the proper format, and upon saving the new information, the “Electronic Robot Description Updated” message will pop up.</li> <li>- <b>Failure:</b> In case of missing any required description like Robot-id an error message with “Electronic Robot Description Not Updated” will pop up.</li> </ul>	<p><b><u>2.2 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create UI for updating Electronic Robot description.</li> <li>- Write code to update Electronic Robot description in database.</li> <li>- Link UI with code.</li> <li>- Link adding new electronic robot tab with inventory manager tab.</li> </ul>	<p>All the tasks of sub-user story 2.2 are completed within the planned time.</p>
<p><b>Sub-User Story 2.3:</b> Remove Electronic robots</p> <p><b>Description:</b></p> <p><i>As an Inventory Manager, I want to remove unwanted or old electronic robots so that I can keep my system updated about current electronic robots.</i></p> <p><b>Acceptance Criteria:</b></p>	<p><b><u>2.3 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create UI for deleting electronic robots.</li> <li>- Write code to deleting electronic robots in database.</li> <li>- Associate code for searching electronic</li> </ul>	<p>All the tasks of sub-user story 2.3 are completed within the planned time.</p>

<p>I know I am done with removing electronic robots when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> Upon deleting the description of the robot, the message with “Electronic Robots Details Deleted” will pop up. The entry of that robot description will also be deleted on that event.</li> <li>- <b>Failure:</b> Upon entering invalid input format - Error message with “Cannot Delete electronic robots” will pop up. Alternatively, may pop up an error message with “electronic robots not found”.</li> </ul>	<p>robots description with UI for deleting electronic robots.</p> <ul style="list-style-type: none"> <li>- Link UI with code.</li> <li>- Link deleting electronic robots tab with inventory manager tab.</li> </ul>	
<p><b>Sub-User Story 2.4:</b> View electronic robots Details <b>Description:</b> <i>As an Inventory Manager, I want to view electronic robot details like name, specifications, price, and a quantity so that I am able to know which electronic robots are currently available and which features they have.</i></p> <p><b>Acceptance Criteria:</b> I know I am done with viewing electronic robots’ details when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> Valid electronic robots Information provided with the proper format press View button, message with “electronic robots Details will pop up.</li> <li>- <b>Failure:</b> Alternatively, may pop up an error message with “Electronic robots not found”.</li> </ul>	<p><b>2.4 Tasks:</b></p> <ul style="list-style-type: none"> <li>- Create UI for viewing electronic robots’ details.</li> <li>- Write code to view electronic robots’ details by getting information from database.</li> <li>- Link UI with code.</li> <li>- Link viewing electronic robots’ details tab with inventory manager tab.</li> </ul>	<p>All the tasks of sub-user story 2.4 are completed within the planned time.</p>
<p><b>User Story 3: <u>Generate Inventory Report</u></b> <b>Description:</b> <i>As an Inventory Manager, I want to see the detailed inventory report which can be the quantities of all the robots in my inventory i.e. stock available, and which of the Robots are low in stocks so that I can see</i></p>	<p><b>Tasks:</b></p> <ul style="list-style-type: none"> <li>- Creating UI for generating inventory report.</li> <li>- Write code to generate inventory report.</li> <li>- Link UI with code.</li> </ul>	<p>All the tasks of user story 3 are completed within the planned time.</p>

<p><i>the stock of robots that are available using a report.</i></p> <p><b>Acceptance Criteria:</b> I know I am done with generating inventory reports:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> After pressing the “generating report” report cross-ponding to complete inventory will pop up.</li> <li>- <b>Failure:</b> Error Message with text pop up “Error Occurred”.</li> </ul>	<ul style="list-style-type: none"> <li>- Link generate inventory report tab with inventory manager tab.</li> </ul>	
<p><b>Sprint 2:</b> <b>User story 4: <u>Process Sale</u></b> <b>Description:</b> <i>As a cashier, I want to process sale so that I can sale robots to customers and keep record of the sale’s information.</i></p> <p><b>Acceptance criteria:</b> I know I am done with Process Sale when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> After being authenticated, I can process sale by pressing the widget “Make Sale”, “Add Sale Line Item”, and “Remove Sale Line Item”, “End Sale”.</li> <li>- <b>Failure:</b> Error message with “Error occurred” will be pop up.</li> </ul> <p><b>Sub-user story 4.1: Make Sale</b> <b>Description:</b> <i>As a cashier, I want to make new sale so that I can manage sale line robots, record sale and manipulate the process of complete sale.</i></p> <p><b>Acceptance criteria:</b> I know I am done with making sale.</p> <ul style="list-style-type: none"> <li>- <b>Success-</b> After pressing button Make New Sale, a new UI of Sale will pop up where I can start recording sale of electronic robots.</li> <li>- <b>Failure-</b>Error Message with text pop up “Cannot Make New Sale”.</li> </ul>	<p><b>4.1 Tasks:</b></p> <ul style="list-style-type: none"> <li>- Creating UI for Make-Sale.</li> <li>- Write code for making new sale.</li> <li>- Link UI with code.</li> <li>- Link make sale tab with cashier’s tab.</li> </ul>	<p>All of the tasks of sub-user story 4.1 are completed within the planned time.</p>

<p><b>Sub-user story 4.2:</b> Add Sale Line Item</p> <p><b>Description:</b>  <i>As a cashier, I want to make new sale line item for each purchased Electronic Robot so that I can record each electronic root price, quantity and description in each sale and calculate subtotal and total.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with making sale line item.</p> <ul style="list-style-type: none"> <li>- <b>Success-</b> After press button Make new line item, Message text with “New Sale Line Item Created” and attached with the current running sale.</li> <li>- <b>Failure-</b>Error Message with text pop up “Cannot Make New Sale Line Item”.</li> </ul>	<p><b><u>4.2 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Creating UI for Sale Line Item.</li> <li>- Write code for making sale line item.</li> <li>- Link UI with code.</li> <li>- Link create sale line item tab with make sale tab.</li> </ul>	<p>All of the tasks of sub-user story 4.2 are completed within the planned time.</p>
<p><b>Sub-user story 4.3:</b> Remove Sale Line Item</p> <p><b>Description:</b>  <i>As a cashier, I want to remove sale line item for each purchased Electronic Robot from a sale that was not purchased by customer so that I can manage each purchase item.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with removing sale line item.</p> <ul style="list-style-type: none"> <li>- <b>Success-</b> After press button remove new line item, Message text with “Sale Line Item Removed” and sale line item is detached from sale and its price is removed from sale.</li> <li>- <b>Failure-</b>Error Message with text pop up “Cannot Remove Sale Line Item”.</li> </ul>	<p><b><u>4.3 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Deleting UI for Sale Line Item.</li> <li>- Write code for deleting sale line item from a sale.</li> <li>- Link UI with code.</li> <li>- Link delete sale line item tab with new sale tab.</li> </ul>	<p>All of the tasks of sub-user story 4.3 are completed within the planned time.</p>

<p><b>Sub-user story 4.4: End Sale</b></p> <p><b>Description:</b>  <i>As a cashier, I want to end sale so that the sale was recorded and I can make new sale.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with ending sale when:</p> <ul style="list-style-type: none"> <li>- <b>Success-</b> After press button End Sale, Message text with “Sale Ended”</li> <li>- <b>Failure-</b> Error Message with text pop up “Unable to End the Sale”.</li> </ul>	<p><b><u>4.4 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Creating UI for Ending Sale.</li> <li>- Writing code for end sale.</li> <li>- Link code with UI</li> <li>- Linking end sale tab with new sale.</li> </ul>	<p>All the tasks of sub-user story 4.4 are completed within the planned time.</p>
<p><b>User story 5: <u>Process Payment</u></b></p> <p><b>Description:</b>  <i>As a cashier, I want to process payment so that I can keep record of the cash in, cash out and returns on payment.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with process payment when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> After being authenticated, I can process payment information by pressing the widget “Make Payment”, “Handle Returns”,</li> <li>- <b>Failure:</b> Error message with “Error occurred” will be pop up.</li> </ul> <p><b>Sub-user story 5.1: Make Payment</b></p> <p><b>Description:</b>  <i>As a cashier, I want to make payment for current sale along with the taxes so that I can record sale payment with taxes.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with making payment.</p> <ul style="list-style-type: none"> <li>- <b>Success-</b> After pressing button make payment total cash message with “total payment” will pop up including cash in and cash out and includes taxes.</li> </ul>	<p><b><u>5.1 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create UI for make payment.</li> <li>- Write code for make payment.</li> <li>- Linking code with UI.</li> <li>- Linking make payment tab with new sale tab.</li> <li>- Handling Cash in and Cash out</li> </ul>	<p>All the tasks of sub-user story 5.1 are completed within the planned time.</p>

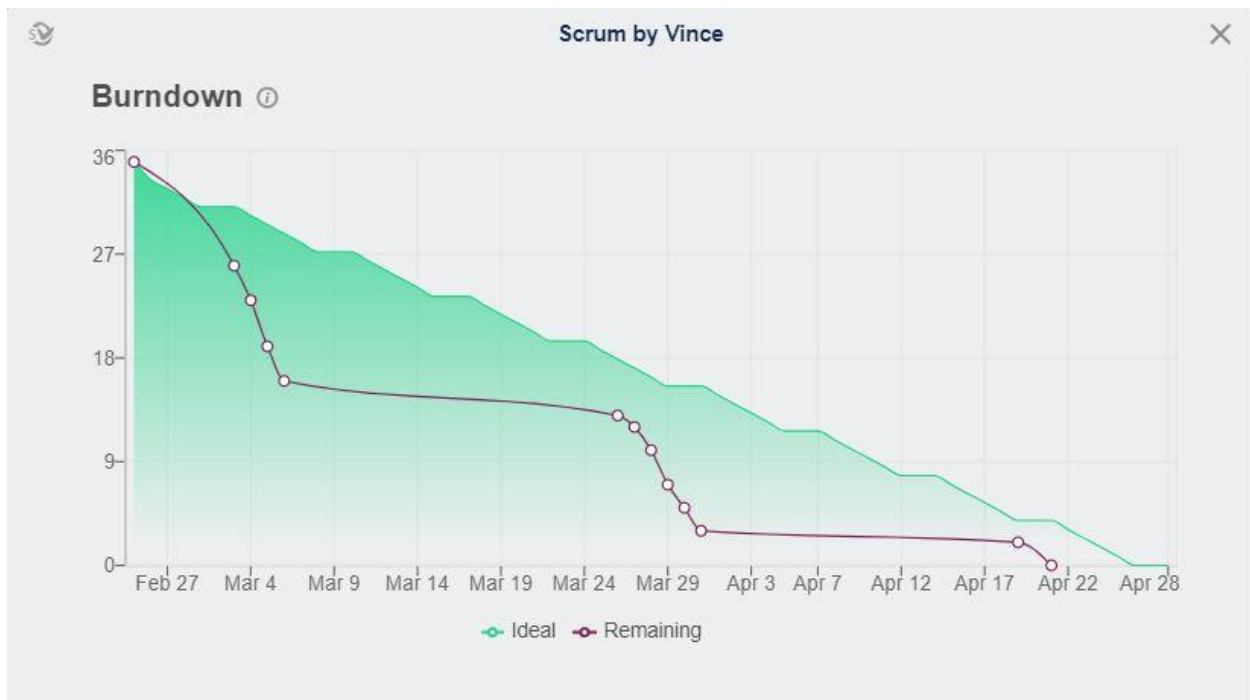
<ul style="list-style-type: none"> <li>- <b>Failure</b>-Error Message with text pop up “Cannot Make Payment” or “unable to Find Returns”.</li> </ul>		
<p><b>Sub-user story 5.2:</b> Handle Returns</p> <p><b>Description:</b>  <i>As a cashier, I want to handle returns on each payment with and without an awareness of discount on that payment.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with making payment.</p> <p><b>Success-</b> After pressing button “payment” message with “returns” will pop up including the amount to be return with applied discount.</p> <p><b>Failure</b>-Error Message with text pop up “unable to Find Returns”.</p>	<p><b><u>5.2 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Create UI for handling returns.</li> <li>- Linking it with payment.</li> </ul>	All the tasks of sub-user story 5.2 are completed within the planned time.
<p><b>User story 6:</b> <u><b>Generate Receipt</b></u></p> <p><b>Description:</b>  <i>As a cashier, I want to generate receipt so that customer can cross check robot name, their quantities and their price.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with generate report when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> After pressing button end sale, receipt with sale will be printed</li> <li>- <b>Failure:</b> Error Message with text pop up “Unable to Generate Receipt”.</li> </ul>	<p><b><u>Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Writing code for generating report.</li> <li>- Link code with end sale</li> </ul>	All the tasks of user story 6 are completed within the planned time.
<p><b>User story 7:</b> <u><b>Generate Sales Report</b></u></p> <p><b>Description:</b>  <i>As a manager, I want to see the detailed summary of sales of my products. I want to see the monthly sales report so that I can use this information to make business decisions.</i></p> <p><b>Acceptance criteria:</b>          I know I am done with generating report.</p>	<p><b><u>Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Creating UI for viewing of monthly/ yearly sales of products.</li> <li>- Writing code for generating report.</li> <li>- Linking UI with the code.</li> </ul>	All the tasks of user story 7 are completed within the planned time.

<ul style="list-style-type: none"> <li>- <b>Success-</b> After pressing button for generating report after selecting monthly or yearly report cross ponding to sales will be pop up.</li> <li>- <b>Failure-</b>Error Message with text pop up “No report Found” or “unable to “Generate Sale Report”.</li> </ul>	<ul style="list-style-type: none"> <li>- Link generate sale report with managers tab.</li> </ul>	
<p><b>Sprint 3:</b></p> <p><b>User story 8: <u>Search Robot Description</u></b></p> <p><b>Description:</b>  <i>As a customer, I want to see the location, price; features of a specific robot in store by giving name or robot-id so that I am able to locate it know its description. I may also want alternative robots that have the required features or functionality.</i></p> <p><b>Acceptance criteria:</b>  I know I am done with searching robot description when:</p> <ul style="list-style-type: none"> <li>- <b>Success:</b> After entering information about electronic robot, details of that robot is displayed.</li> <li>- <b>Failure:</b> Error Message with text pop up “cannot find Robot”.</li> </ul>	<p><b>Tasks:</b></p> <ul style="list-style-type: none"> <li>- Creating UI for searching robot by name or Robot-id</li> <li>- Creating UI for displaying location and features of specific robot.</li> <li>- Writing code for searching robot by the name or Robot-id</li> <li>- Linking UI with the code.</li> <li>- Link “Search Robot description” with cashiers tab.</li> </ul>	<p>All the tasks of user story 8 are completed within the planned time.</p>
<p><b>User story 9: <u>Make Custom Demand</u></b></p> <p><b>Description:</b>  <i>As a Customer, I want to place an order for a robot that is out of stock or want to make a custom robot for myself by providing the description of functionality and features of required robot.</i></p> <p><b>Acceptance criteria:</b>  I know I am done with making demand for custom robots</p> <ul style="list-style-type: none"> <li>- <b>Success-</b> After entering information about custom electronic robot and my own information including phone #, message pop up with text “Custom Demand Saved” and saving customer information.</li> </ul>	<p><b>Tasks:</b></p> <ul style="list-style-type: none"> <li>- Making UI for Make Custom Demand, accessible without authentication for any customer in the store.</li> <li>- Writing code for make custom demand</li> <li>- Linking UI with the code Link “Make Custom Demand” with general view or customers tab.</li> </ul>	<p>All the tasks of user story 9 and sub-user story 9.1 are completed within the planned time.</p>

<ul style="list-style-type: none"> <li>- <b>Failure</b>-Error Message with text pop up “Cannot make demand”, in case of entering any invalid information.</li> </ul> <p><b>Sub user story 9.1:</b> view Customer Demands</p> <p><b>Description:</b>  <i>As an Inventory Manager, I want to see the customer requirement and able to see trends on the customer sight so that become familiar with customer demands and able to communicate to them better by providing them certain products.</i></p> <p><b>Acceptance Criteria:</b>  I know I am done with making demand for custom robots</p> <ul style="list-style-type: none"> <li>- <b>Success</b>- After pressing the Button “View Demands” the new screen will pop up and shows customer details.</li> <li>- <b>Failure</b>-Error Message with text pop up “Cannot show demand”, in case of entering any invalid information.</li> </ul>	<p><b><u>9.1 Tasks:</u></b></p> <ul style="list-style-type: none"> <li>- Making UI for viewing customer demand,</li> <li>- Writing code for viewing custom demand</li> <li>- Linking UI with the code</li> <li>- Link “view customer demand” with inventory manager tab.</li> </ul>	
---	---	--

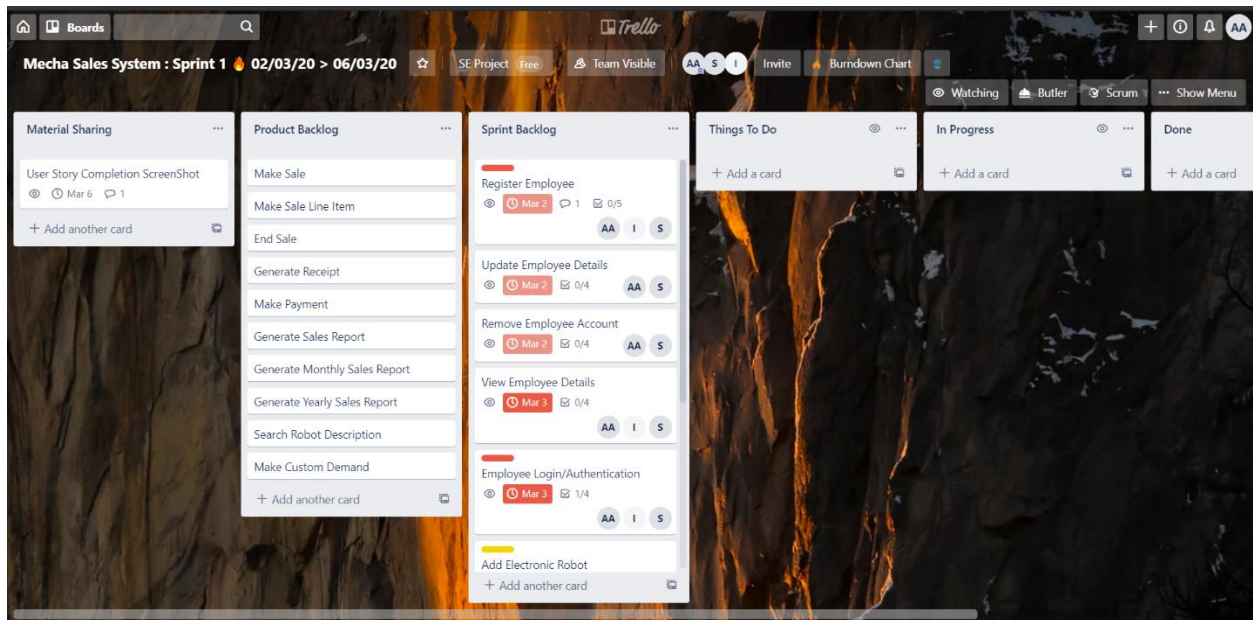


## Product Burndown Chart

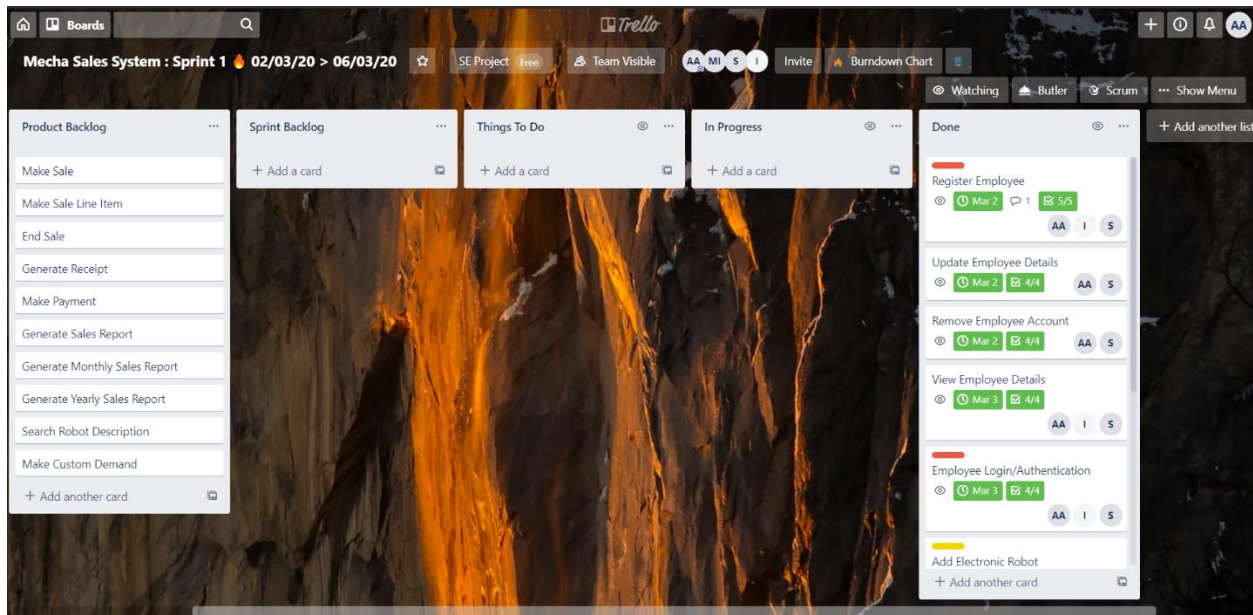


## Trello Snapshots

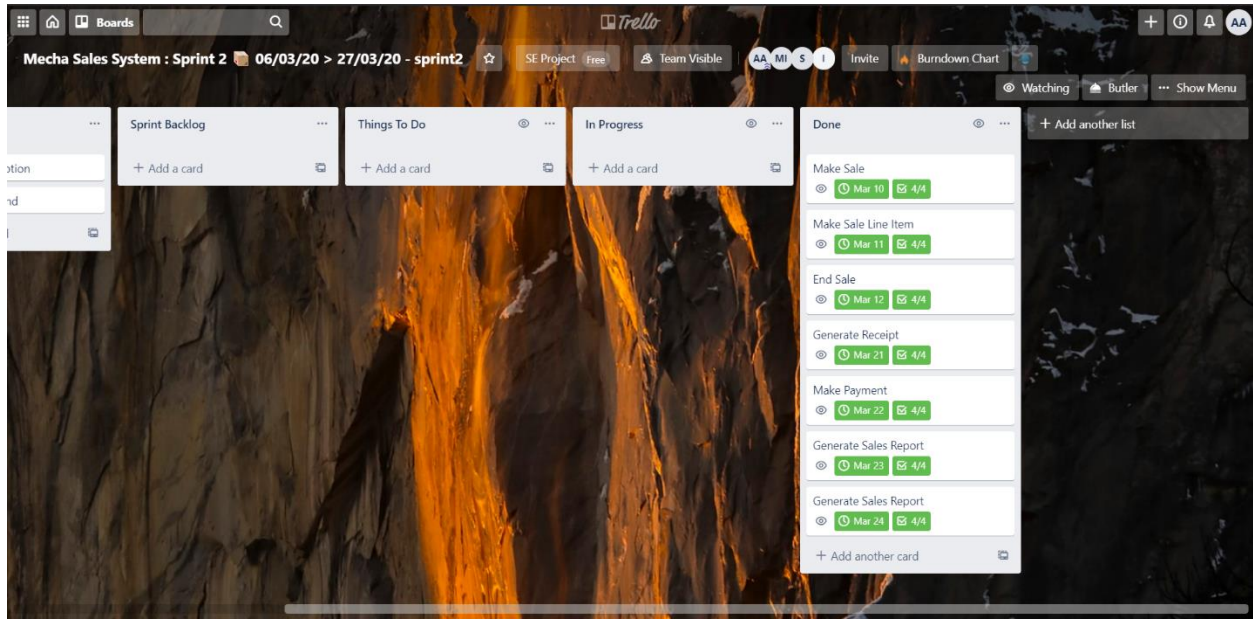
Start:



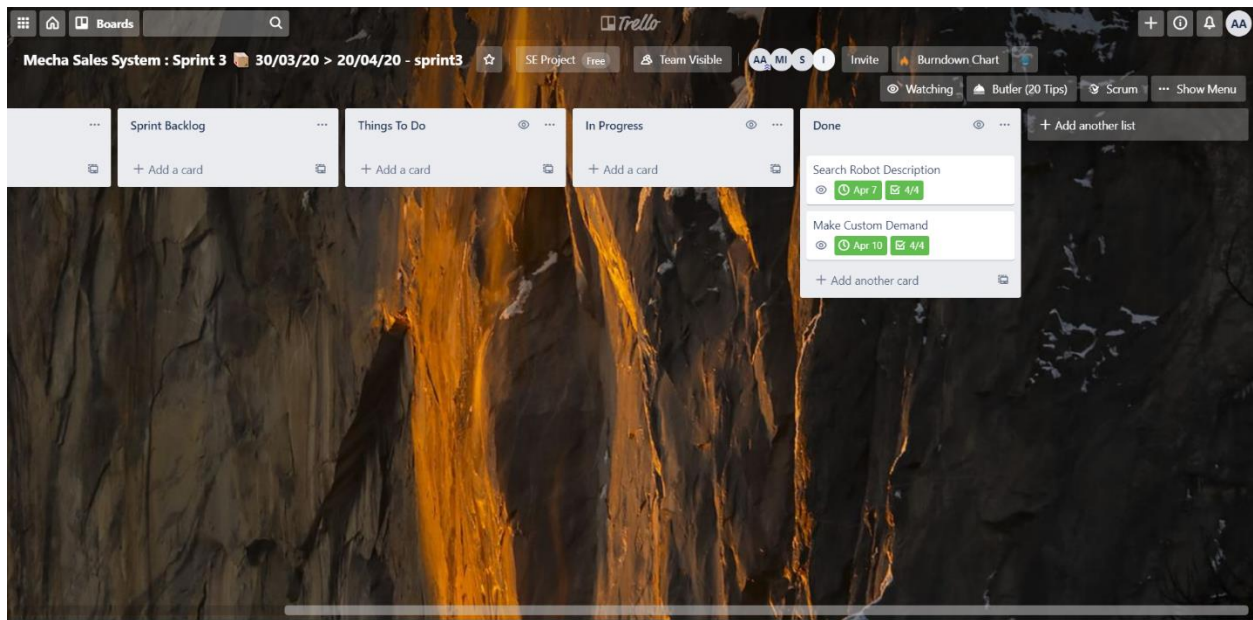
Sprint 1 Completion:



## Sprint 2 Completion:



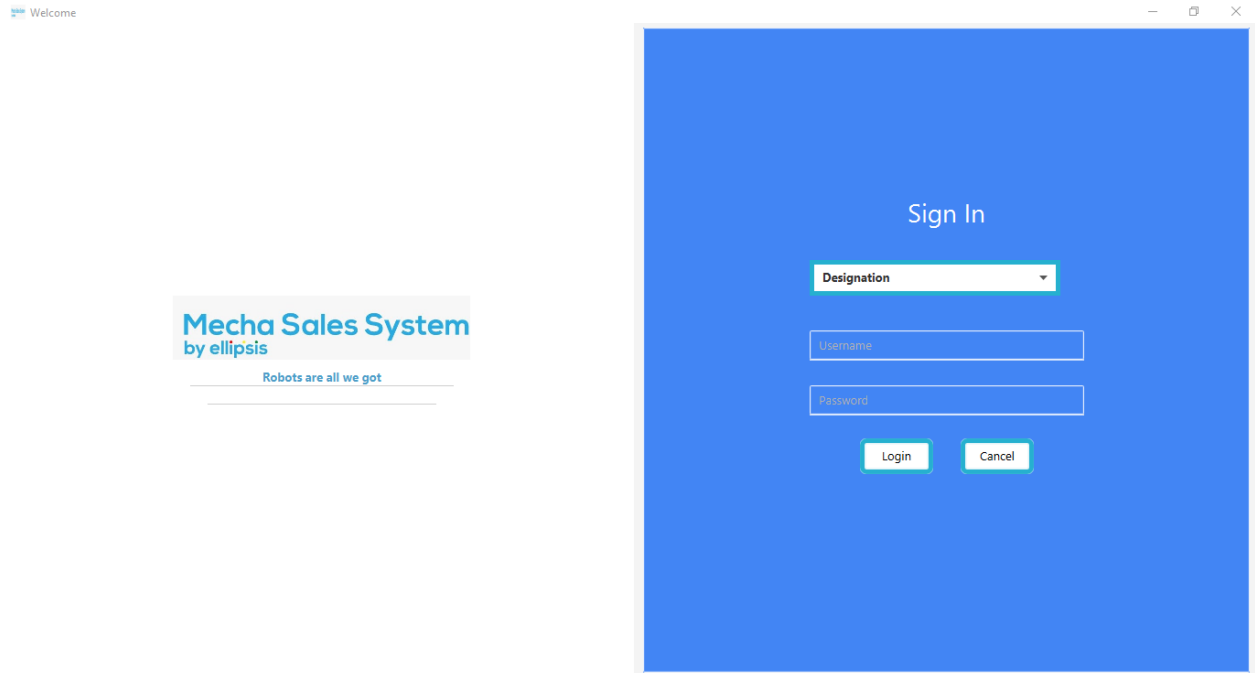
## Sprint 3 Completion:



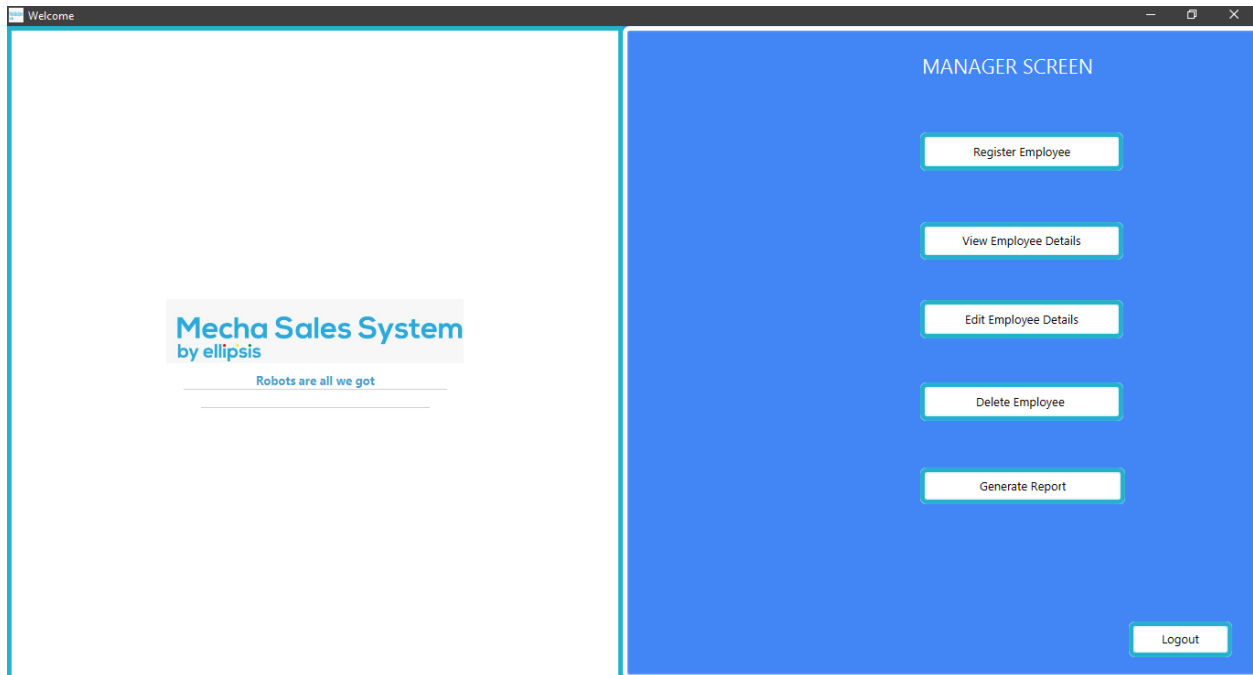
## Project GUIs

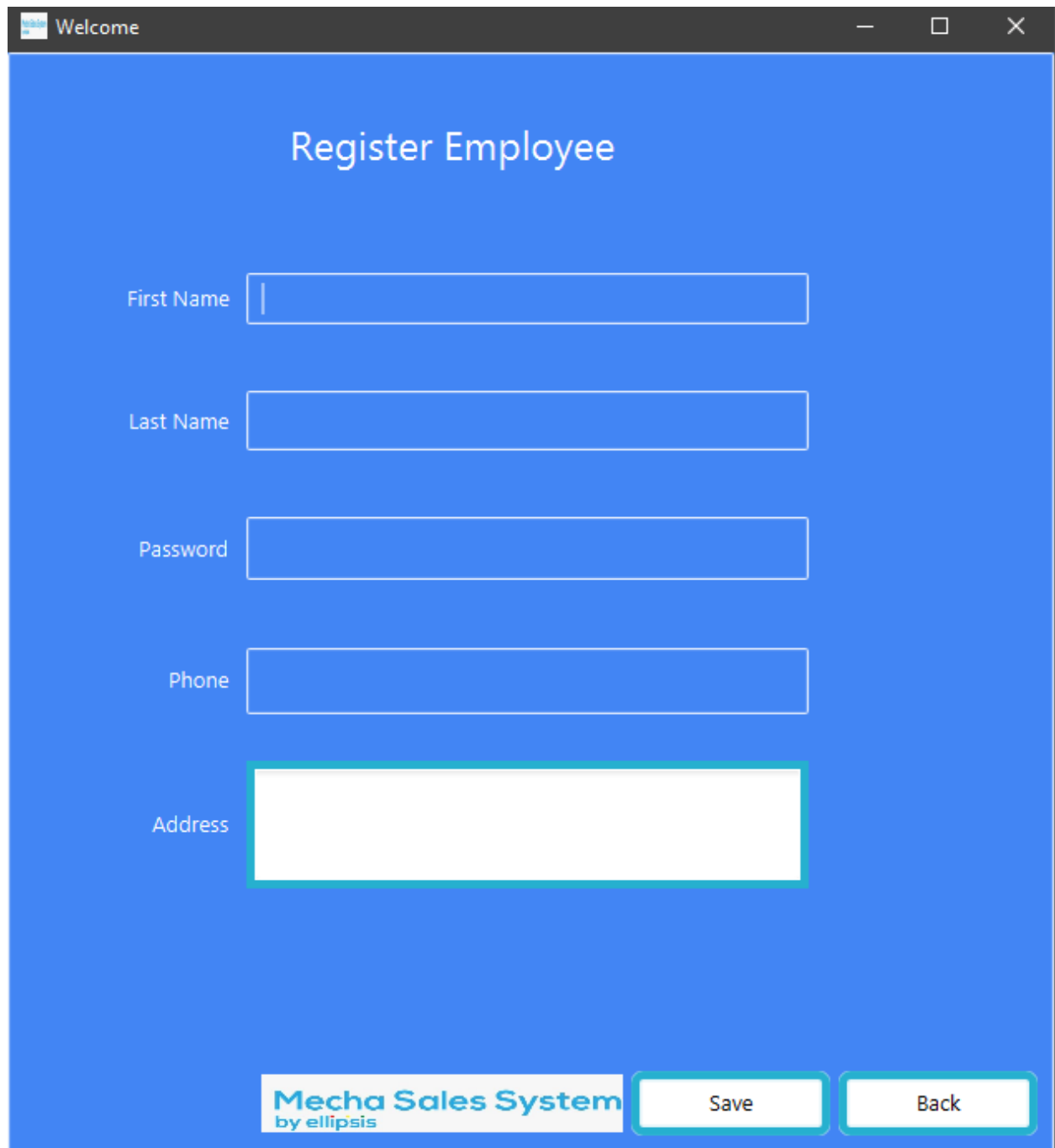
### Sprint 1 (Manager's Module):

#### Employee Authentication UI:



Manager Screen UI:



Register Employee UI:A screenshot of a web application window titled 'Welcome'. The main content area has a blue background and is titled 'Register Employee'. It contains five input fields: 'First Name', 'Last Name', 'Password', 'Phone', and 'Address'. The 'Address' field is highlighted with a green border. At the bottom, there is a logo for 'Mecha Sales System by ellipsis' and two buttons labeled 'Save' and 'Back', both of which are also highlighted with green borders.

Welcome

## Register Employee

First Name

Last Name

Password

Phone

Address

**Mecha Sales System**  
by ellipsis

Save Back

## Update Employee Details UI:

Welcome

### Update Details

Employee ID

Name

Password

Phone

Address

Type

**Mecha Sales System**  
by ellipsis

Delete Employee UI:

Welcome

## Delete Account

Employee ID

Name

Password

Phone

Address

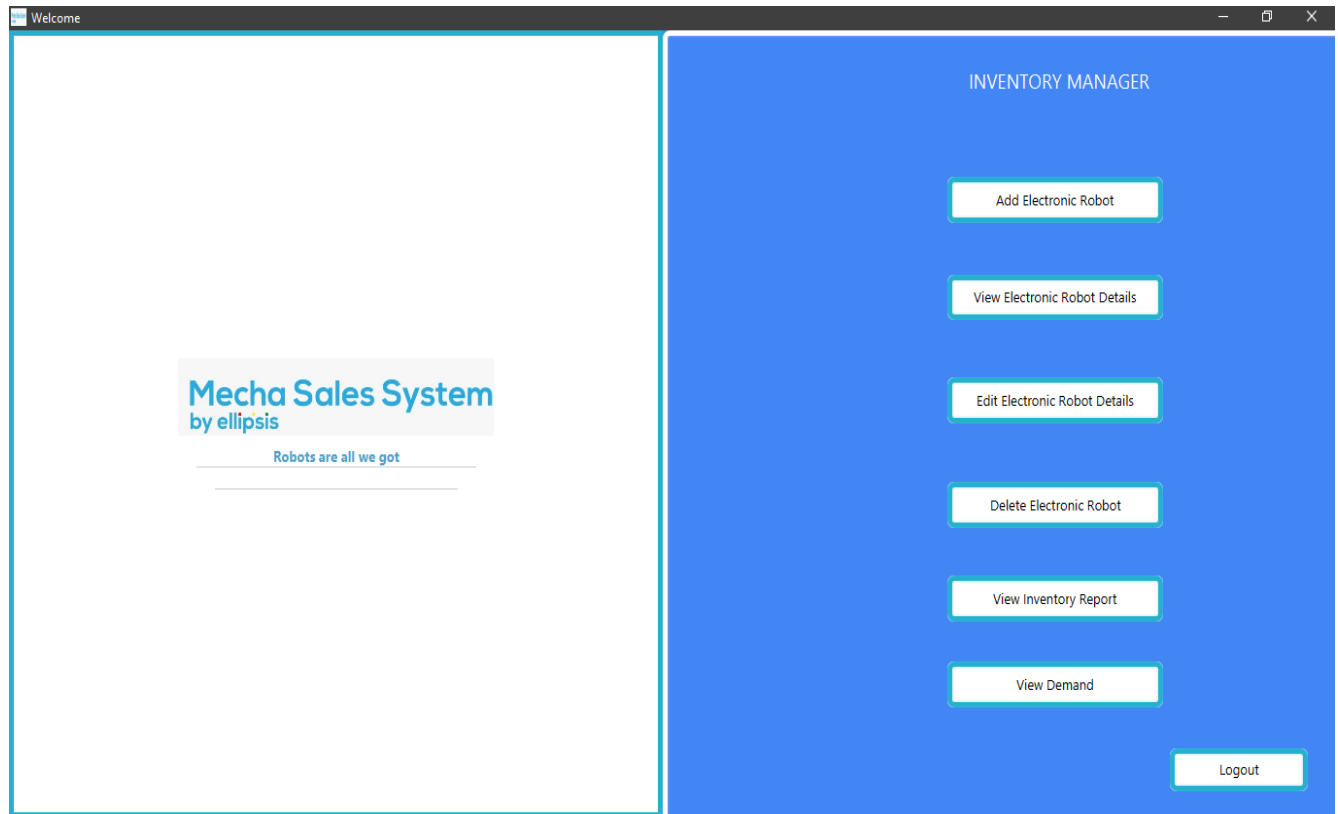
Mecha Sales System  
by ellipsis



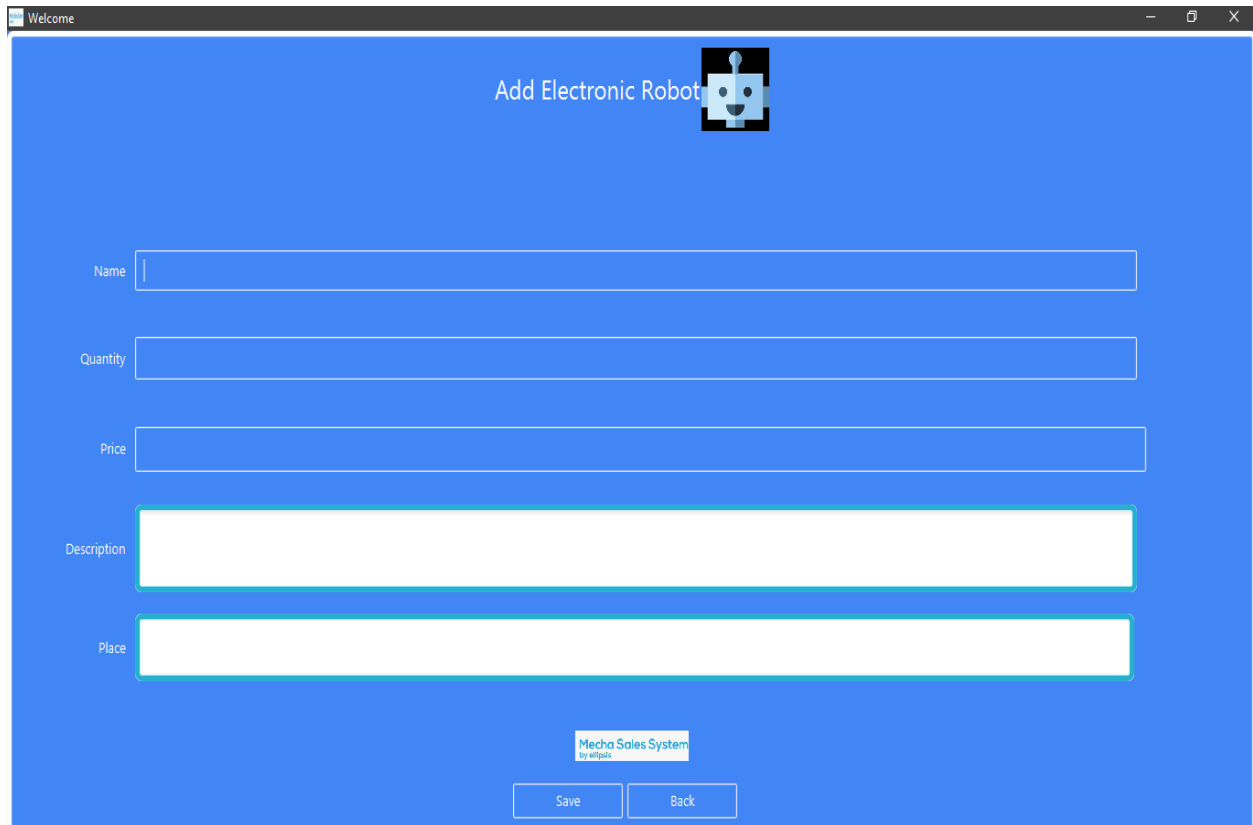
### View Employee Details UI:

[illegible]

Inventory Manager UI:



Add Electronic Robots UI:



Welcome

Add Electronic Robots

Name

Quantity

Price

Description

Place

Mecha Sales System  
by ellipsis

Save Back

Edit E Robot Description UI:

Update Robot Details

Code

Name

Quantity

Price

Description


Place

Mecha Sales System  
by ellipsis

View Electronic Robot Details:

Welcome

View Robot Details



Code

Search

Name

Quantity

Price

Description

Place


Mecha Sales System  
by ellipsis

Back

Delete E Robot Details:

Welcome

Delete E Robot



Code

Search

Name

Quantity

Price

Description

Place

Mecha Sales System  
by ellipsis

Delete

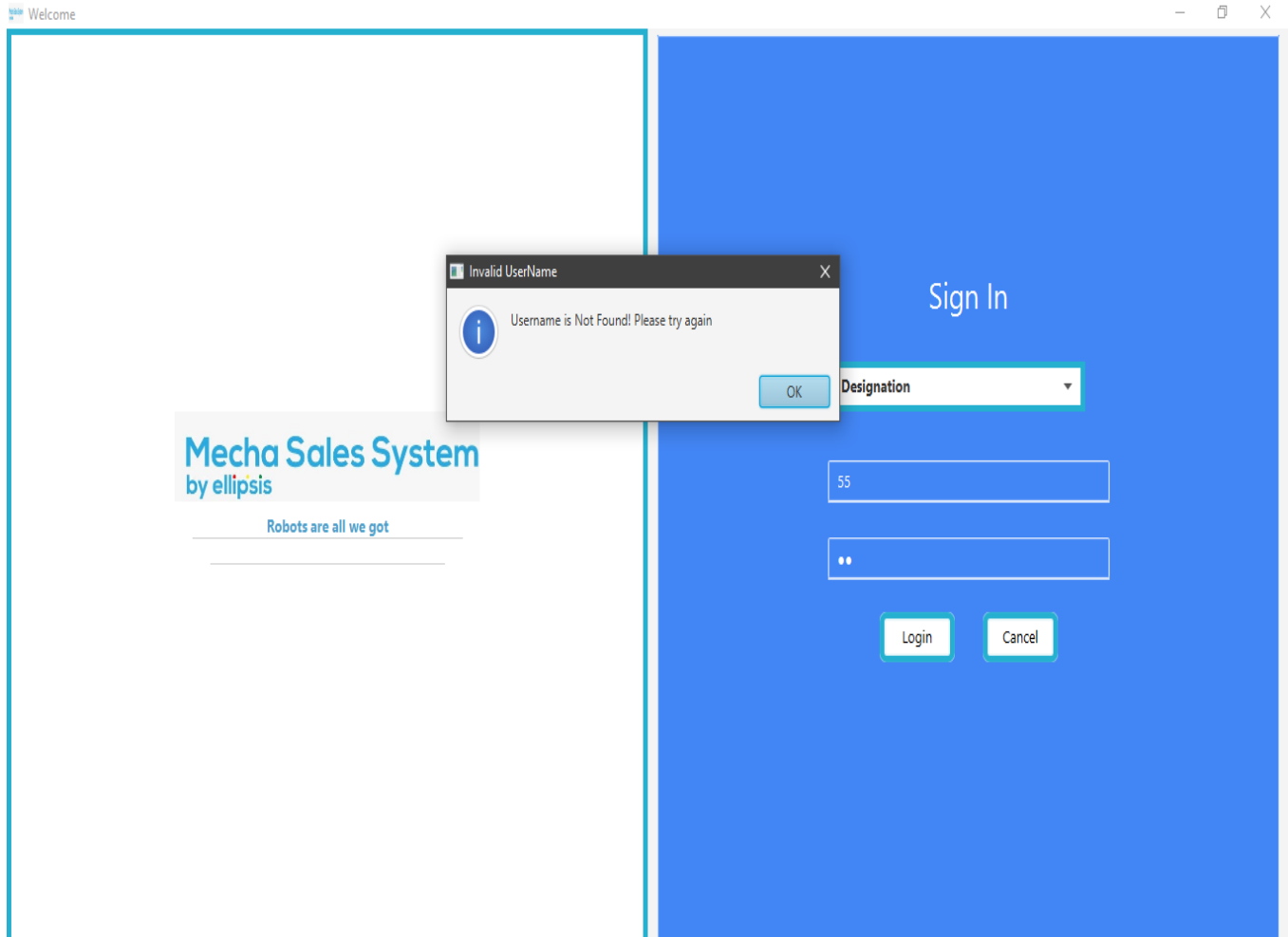
Back

Inventory Report UI:

[illegible]

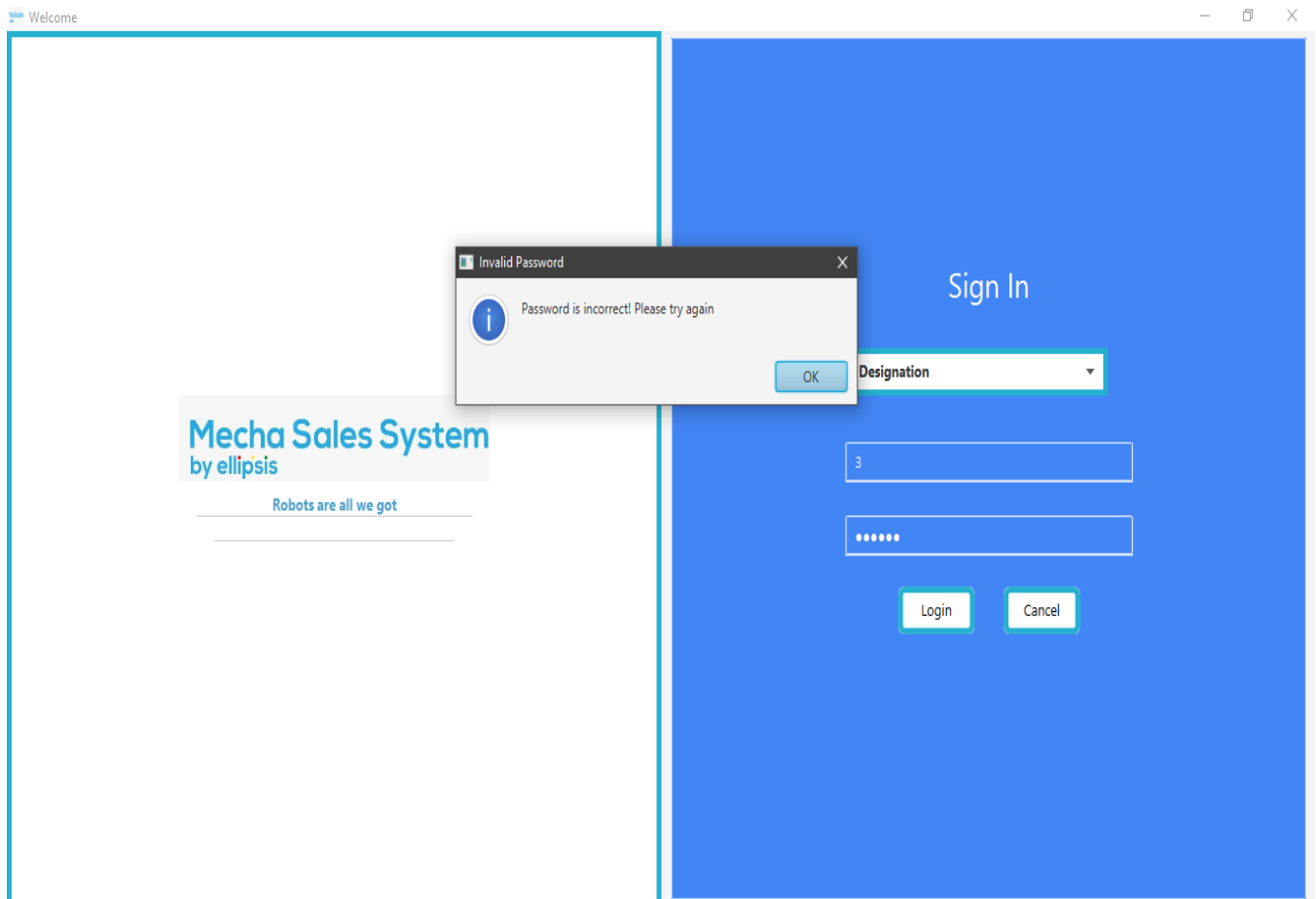
## Sprint 1 Testing (Manager's Module Testing):

### Test Case 1:

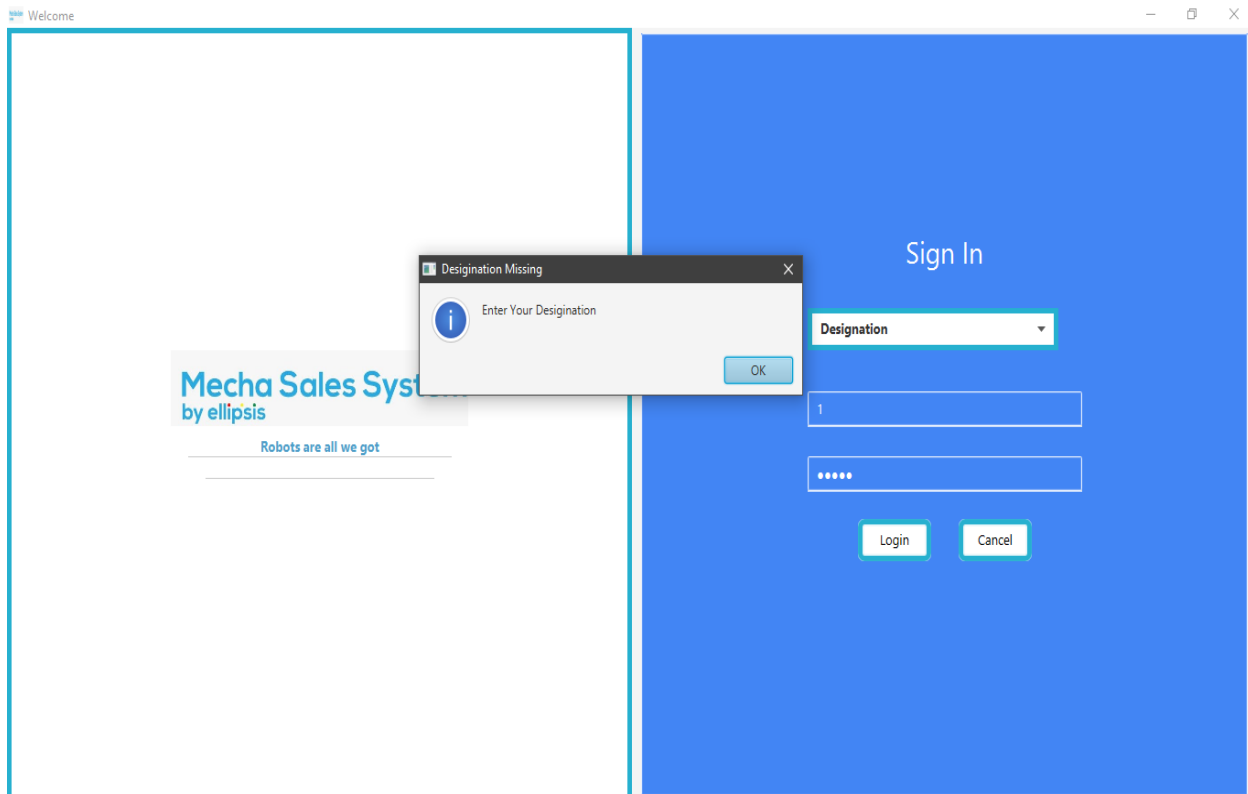




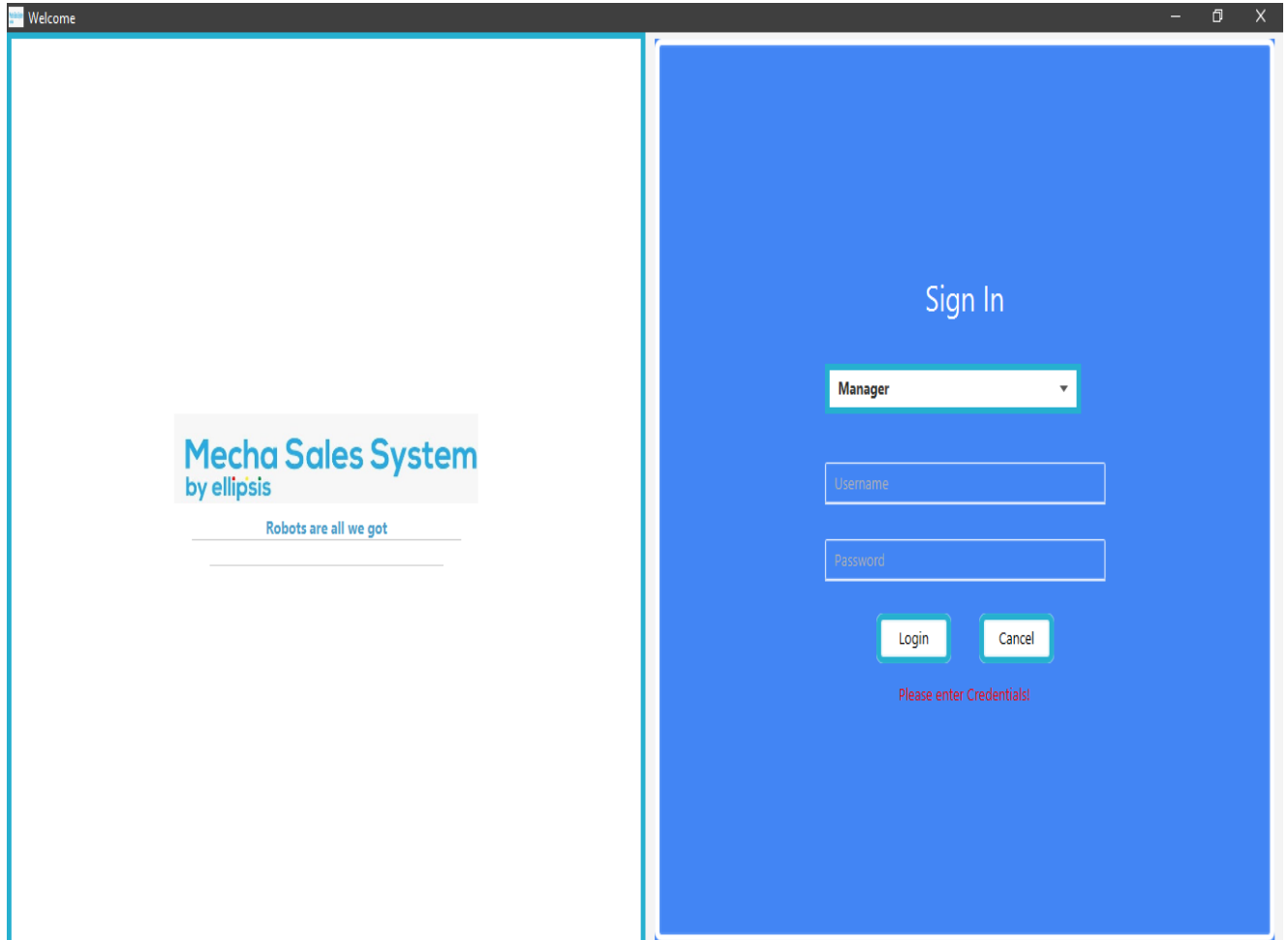
Test Case 2:



**Test Case 3:**



**Test Case 4:**



**Test Case 5:**

Welcome

### Delete Account

Employee ID

Name

Password

Phone

Address


Please enter Employee ID!

Mecha Sales System  
by ellipsis

**Test Case 6:**

Welcome

### Delete E Robot



Code

Name

Quantity

Price

Description

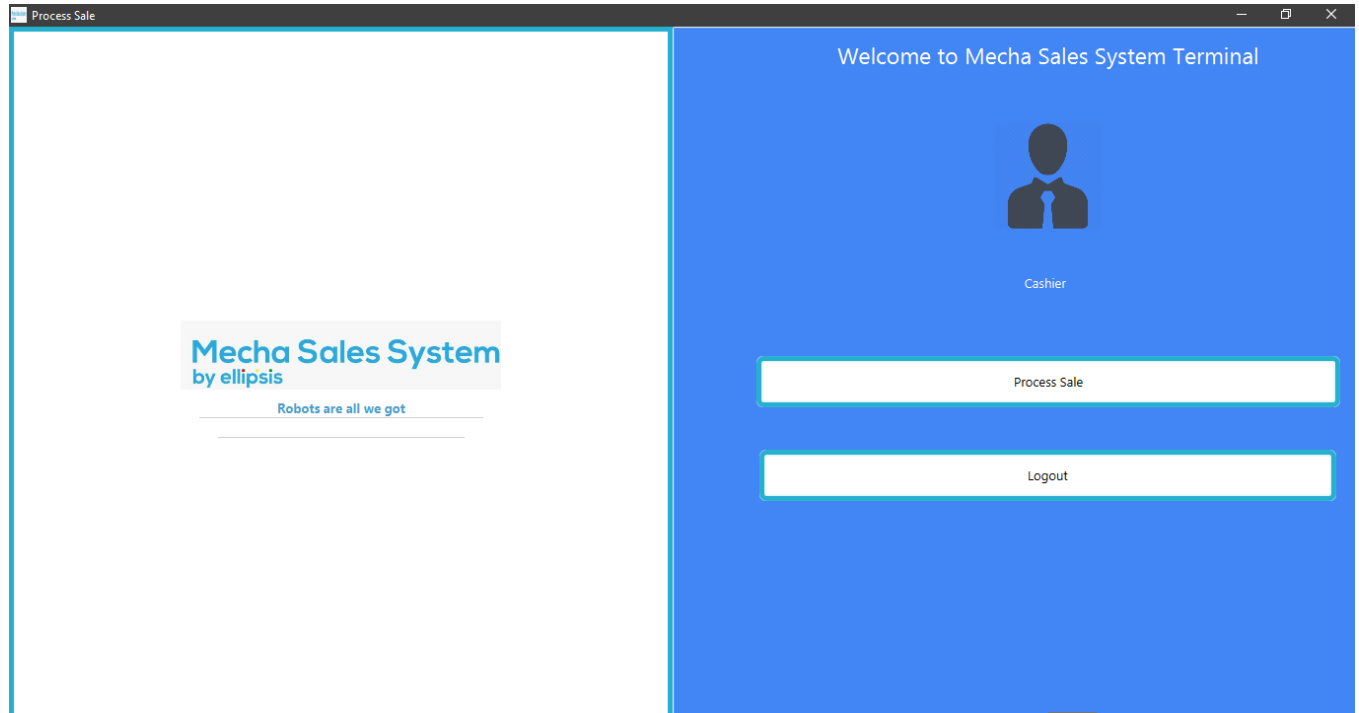
Place

Please enter Robot Code!

Mecha Sales System  
by ellipsis

## Sprint 2 (Sales Module):

### Cashier Screen UI:



### Process Sale UI:

[illegible]

**Add Sale Line Item UI:**

Process Sale

Add Sale Line Item

Code: 1 FindInfo

Name: Drone

Quantity: 5

Price: 4500

Description: Remote control drone

Mecha Sales System  
by ellipsis

Back Add Sales Item



**DB Storage UI:****Sale:**

	sale_code	total_amount	payment
▶	1	2500	3000
	2	1500	2000
	3	1600	2000
	4	0	0
	5	0	0
	6	24000	24000
	7	0	0

**Sale Line Item:**

ID	sale_code	robot_code	robot_name	robot_desc	robot_price	qty	subtotal
1	1	2010	drone	Remote Control Drone	2500	1	2500
2	1	2010	Drone	Remote Control Drone	1500	1	1500
3	1	201	Drone	Remote control drone	4000	5	20000
4	2	201	Drone	Remote control drone	4000	8	16000
5	2	201	Drone	Remote control drone	4000	8	20000
6	3	201	Drone	Remote control drone	4000	6	24000
7	4	201	Drone	Remote control drone	4000	6	24000

### Edit Sale Line Item UI:

Process Sale

Edit Sale Line Item

Code  FindInfo

Name

Quantity

Price


Description

Mecha Sales System  
by ellipsis

Back Edit Sale Line

## Make Payment UI:

Process Sale

Payment 

Total 18000

Net Payment

(-) Discount

Returns

Make Payment

Mecha Sales System  
by ellipsis

**Generate Receipt UI:**

Purchase Details

Receipt

ERobot_Code	Description	Robot Name	Price	Quantity	Sub Total	Var	Sale No
67	2	Testing	testing	67	5	335	
67	5	Alexa	Alexa Robot does pretty amazing stuff			450	4
1800							

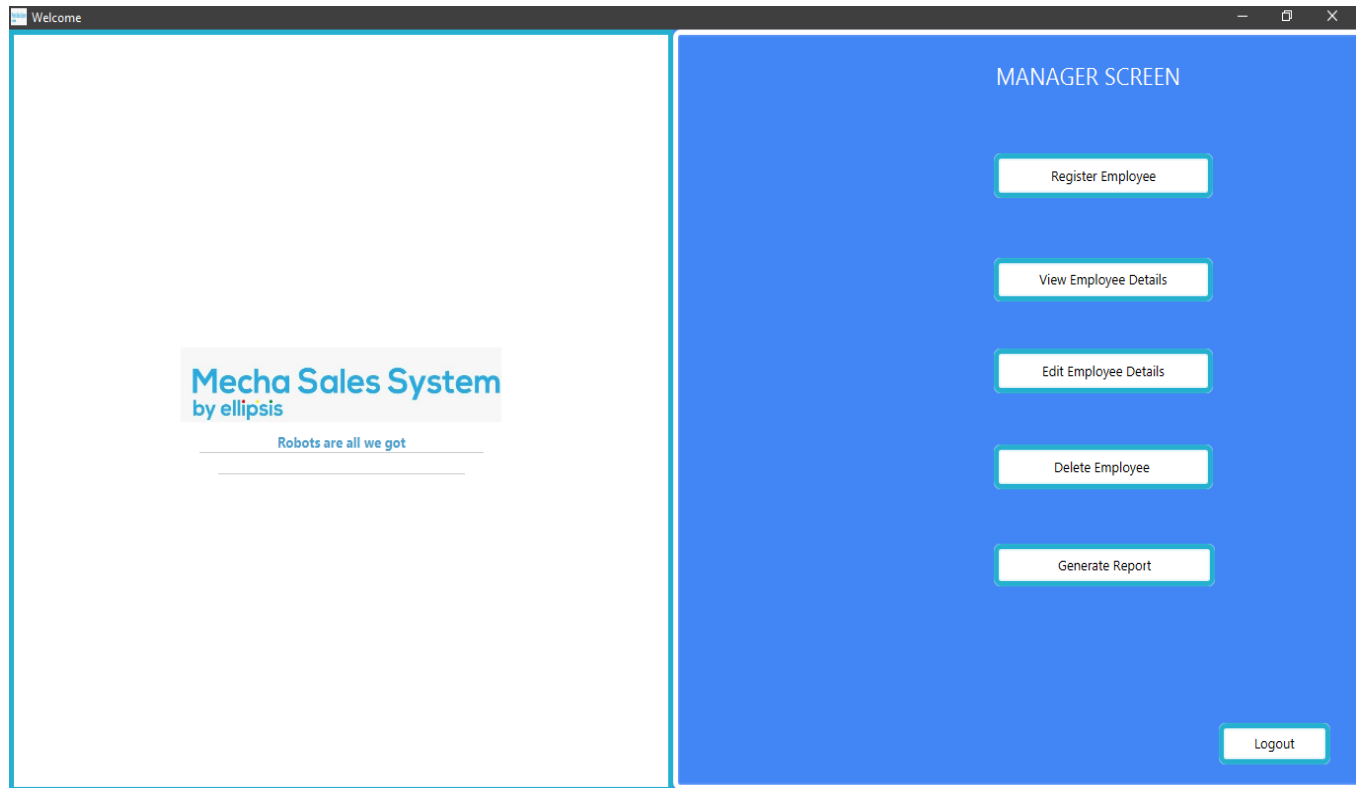
Total = 2135

Payment = 2200

Returns = 65

OK

## Manager Access To Generate Report UI:



**Generate Sale Report UI:**

Welcome


ID	sale_code	robot_code	robot_name	robot_desc	robot_price	qty	subtotal
1	1	2010	drone	Remote Control Drone	2500	1	2500
2	1	2010	Drone	Remote Control Drone	1500	1	1500
3	1	201	Drone	Remote control drone	4000	5	20000
4	2	201	Drone	Remote control drone	4000	8	16000
5	2	201	Drone	Remote control drone	4000	8	20000
6	3	201	Drone	Remote control drone	4000	6	24000
7	4	201	Drone	Remote control drone	4000	6	24000
8	5	209	Toyia	DEIRWE	149	5	745
9	6	201	Drone	Remote control drone	4000	6	24000
10	6	209	Toyia	DEIRWE	149	6	894
11	7	201	Drone	Remote control drone	4000	8	32000
12	8	201	Drone	Remote control drone	4000	5	20000
13	9	201	Drone	Remote control drone	4000	5	20000
14	10	201	Drone	Remote control drone	4000	5	20000
15	11	202	Elephant	HEOSD	42	5	210
16	12	209	Toyia	DEIRWE	149	5	745
17	13	202	Elephant	HEOSD	42	10	420
18	43	3	Elephant	HEOSD	42	4	168
19	44	3	Elephant	HEOSD	42	19	798
20	45	1	Drone	Remote control drone	4500	3	13500
21	47	1	Drone	Remote control drone	4500	3	13500
22	49	1	Drone	Remote control drone	4500	2	9000
23	50	1	Drone	Remote control drone	4500	5	22500
24	51	3	Elephant	HEOSD	42	3	126

Back

Total Sales: 38

Total Quantity: 203

Total Amount: 104258



Sales Report

## Sprint 2 Testing (Sales Module Testing):

### **Test Case 1:**

The screenshot shows a web application window titled "Process Sale". The main content area has a blue background. At the top center, it says "Add Sale Line Item" next to a small robot icon. Below this, there are five input fields: "Code", "Name", "Quantity", "Price", and "Description". The "Code" field has a "Find Info" button to its right. At the bottom center, there is a red error message "Please enter Credentials!". Below the error message is a logo for "Mecha Sales System by ellipsis". At the very bottom, there are two buttons: "Back" and "Add Sales Item".

Process Sale

Add Sale Line Item

Code  Find Info

Name

Quantity

Price

Description

Please enter Credentials!

Mecha Sales System  
by ellipsis

Back Add Sales Item

## **Test Case 2:**

The screenshot shows a web application window titled "Process Sale". The main content area has a blue background and is titled "Edit Sale Line Item" with a small robot icon. It contains several input fields: "Code" (with a "FindInfo" button), "Name", "Quantity", "Price", and "Description". A red error message "Please enter Credentials!" is displayed. The "Mecha Sales System by ellipsis" logo is centered. At the bottom are "Back" and "Edit Sale Line" buttons.

Process Sale

Edit Sale Line Item

Code  FindInfo

Name

Quantity

Price

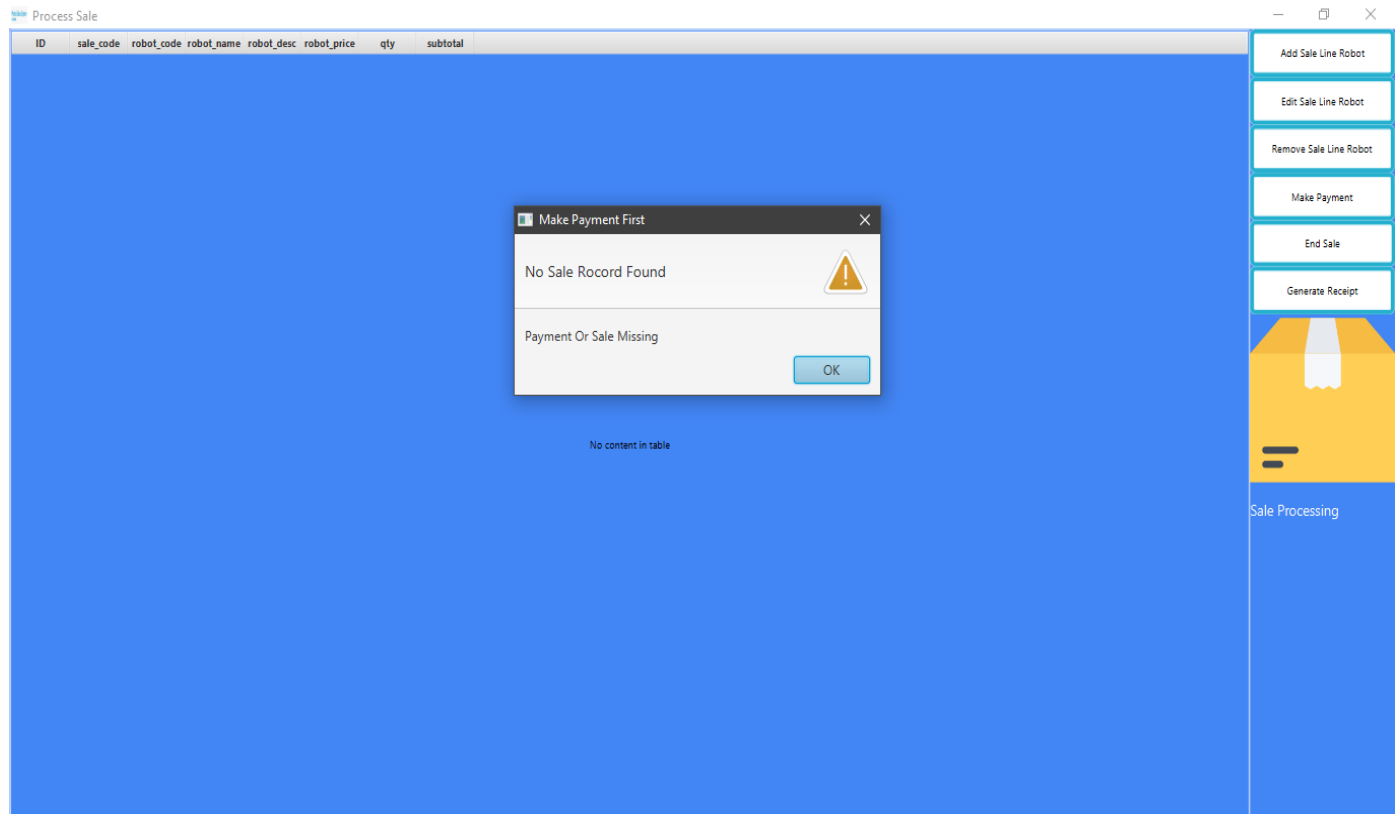
Description

Please enter Credentials!

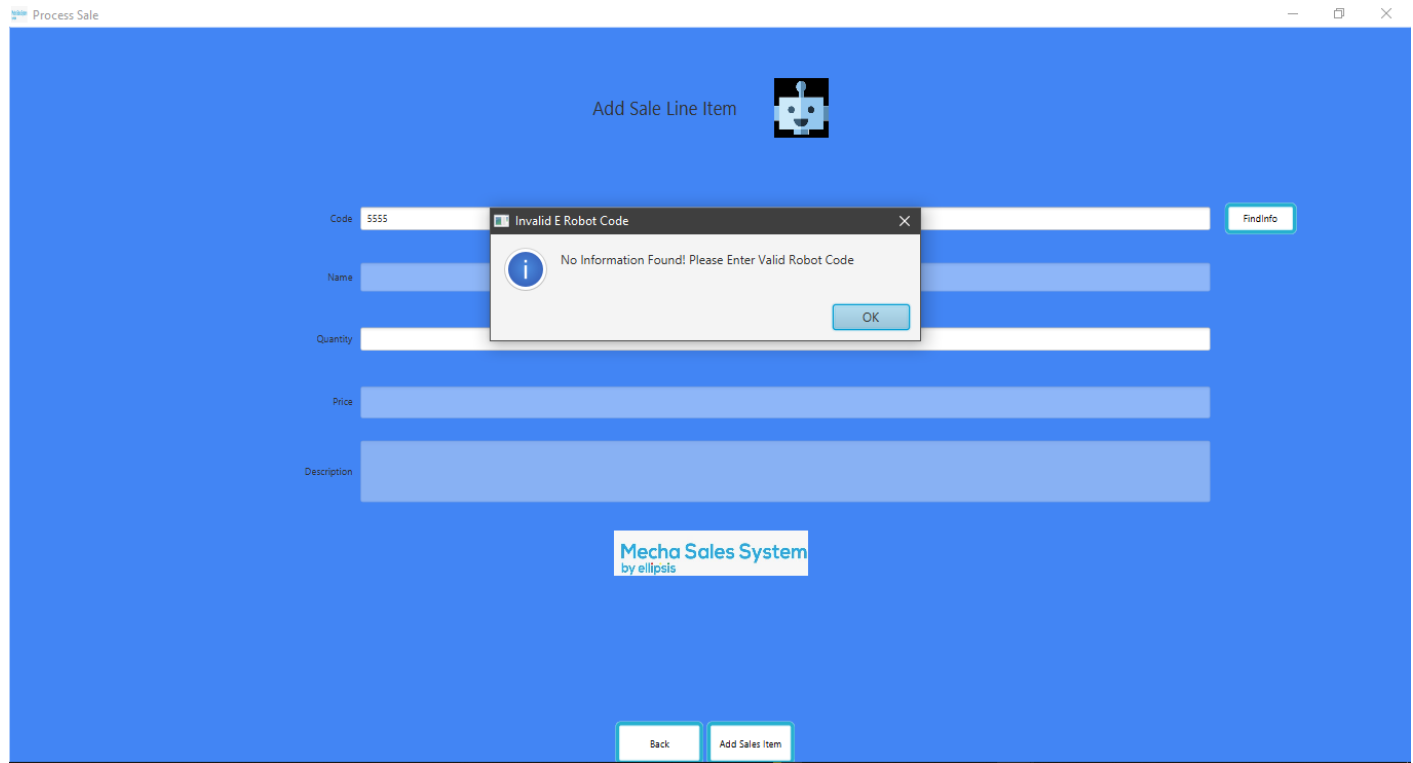
Mecha Sales System  
by ellipsis

Back Edit Sale Line




**Test Case 3:**

**Test Case 4:**



**Test Case 5:**

Process Sale

Payment 

Total 22500

Net Payment 23000

(-) Discount 0

Returns 500

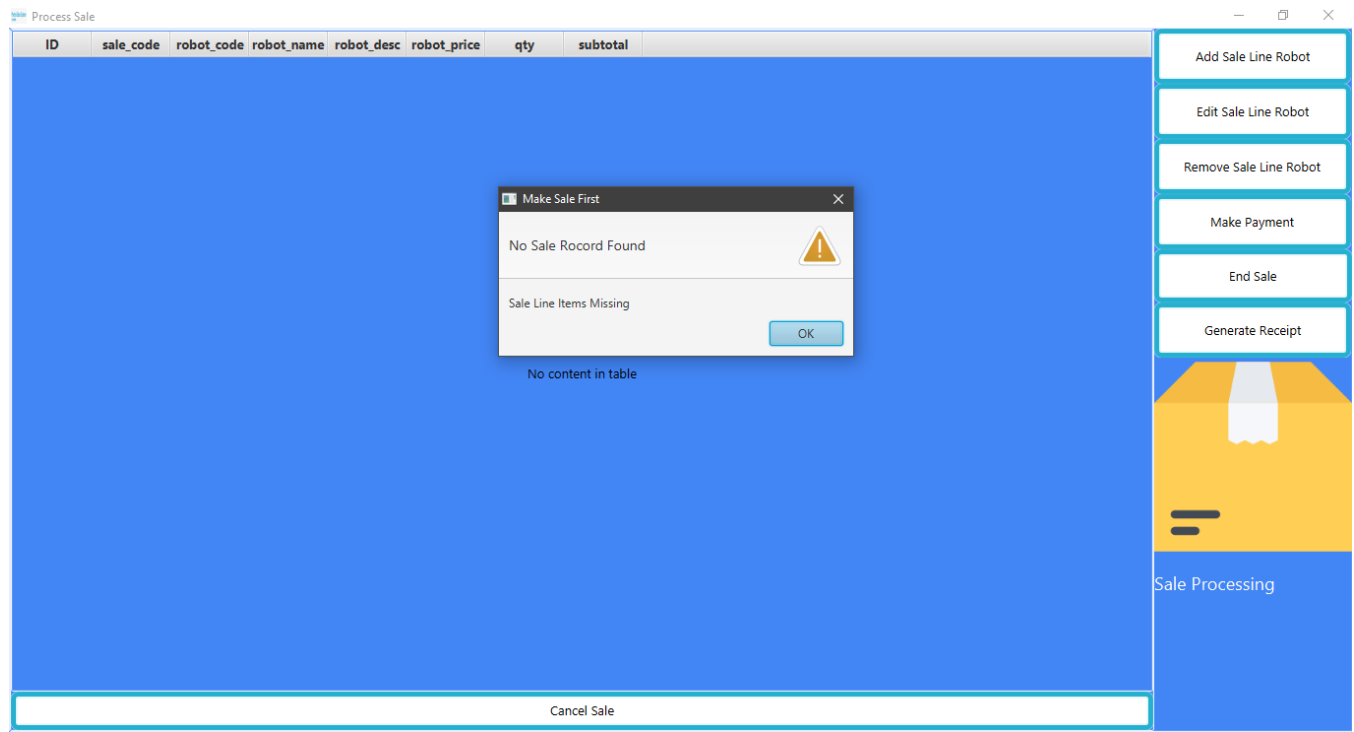
Payment Details

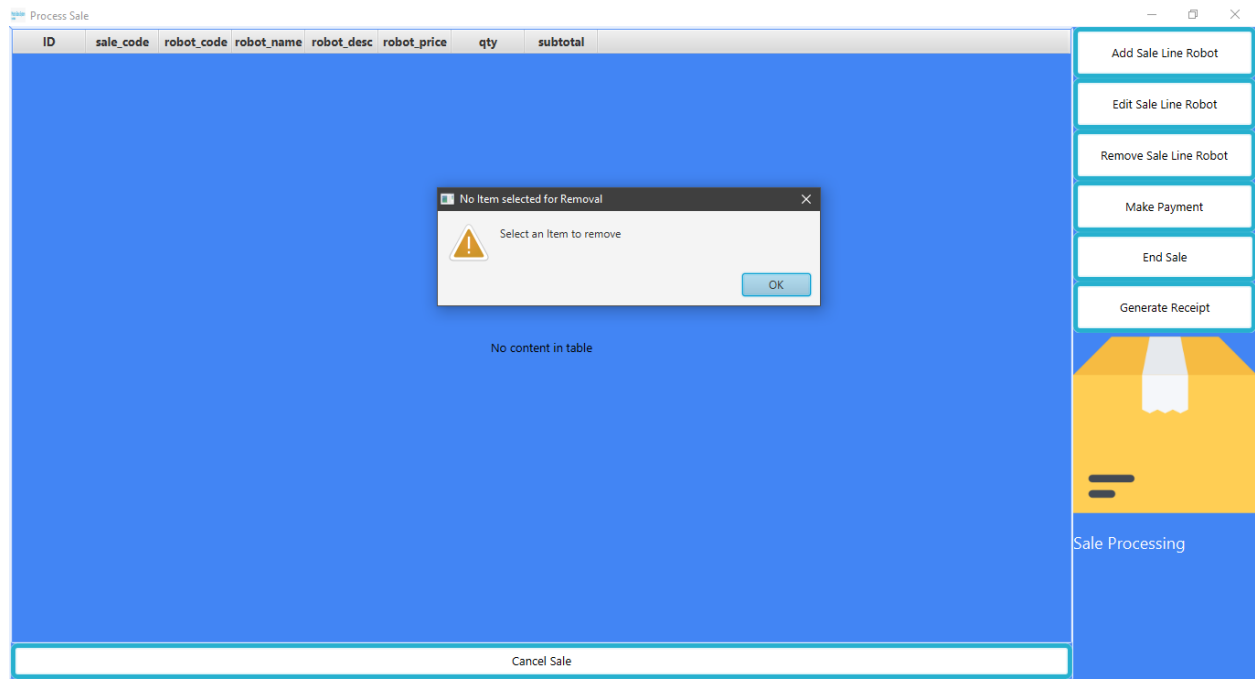
Payment Successfully Done!!

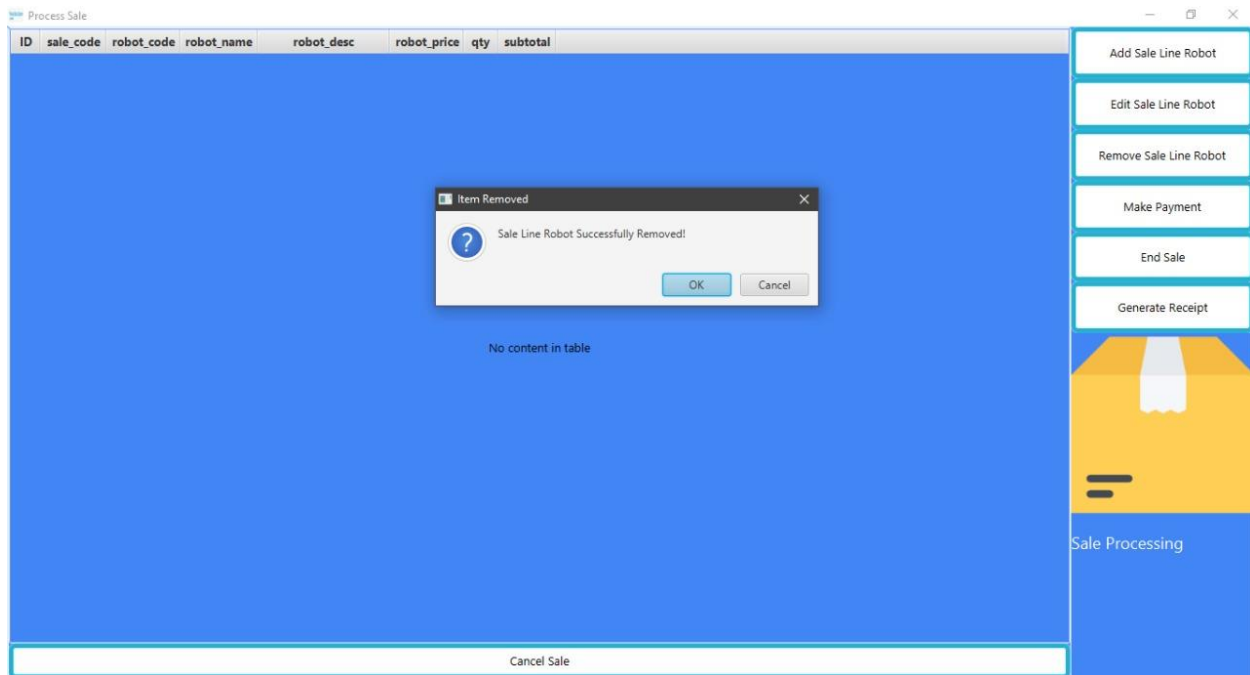
OK

Make Payment

Mecha Sales System  
by ellipsis

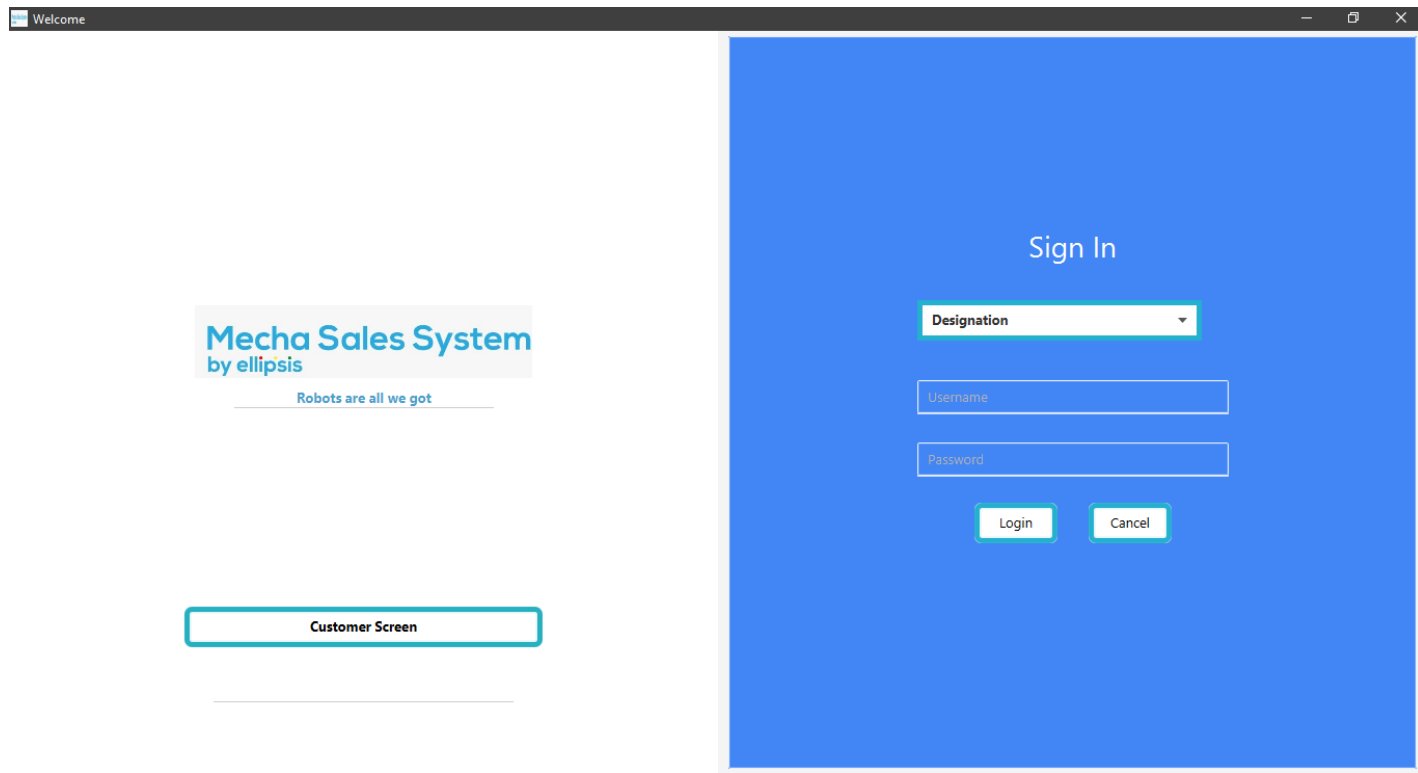
**Test Case 6:**

**Test Case 7:**

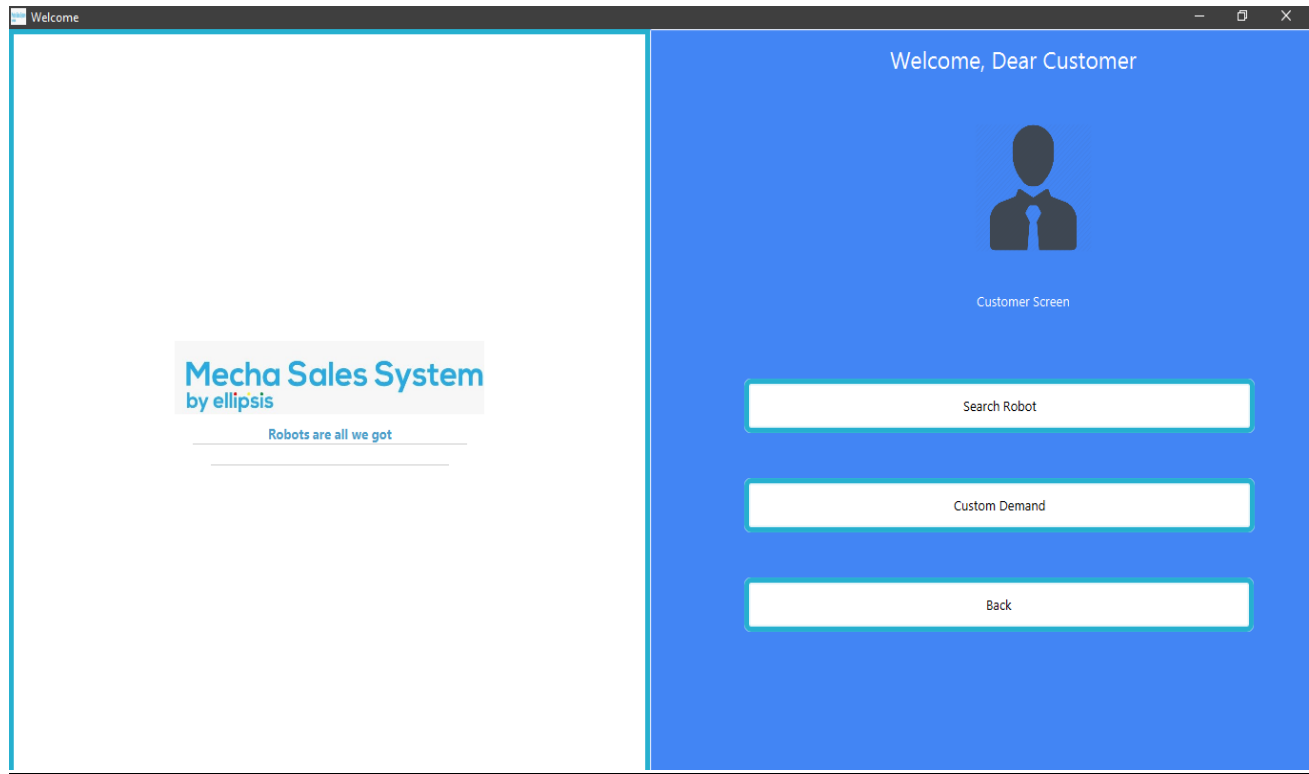
**Test Case 8:**

Sprint 3 (Search and Demand Module):

Customer Screen Button UI

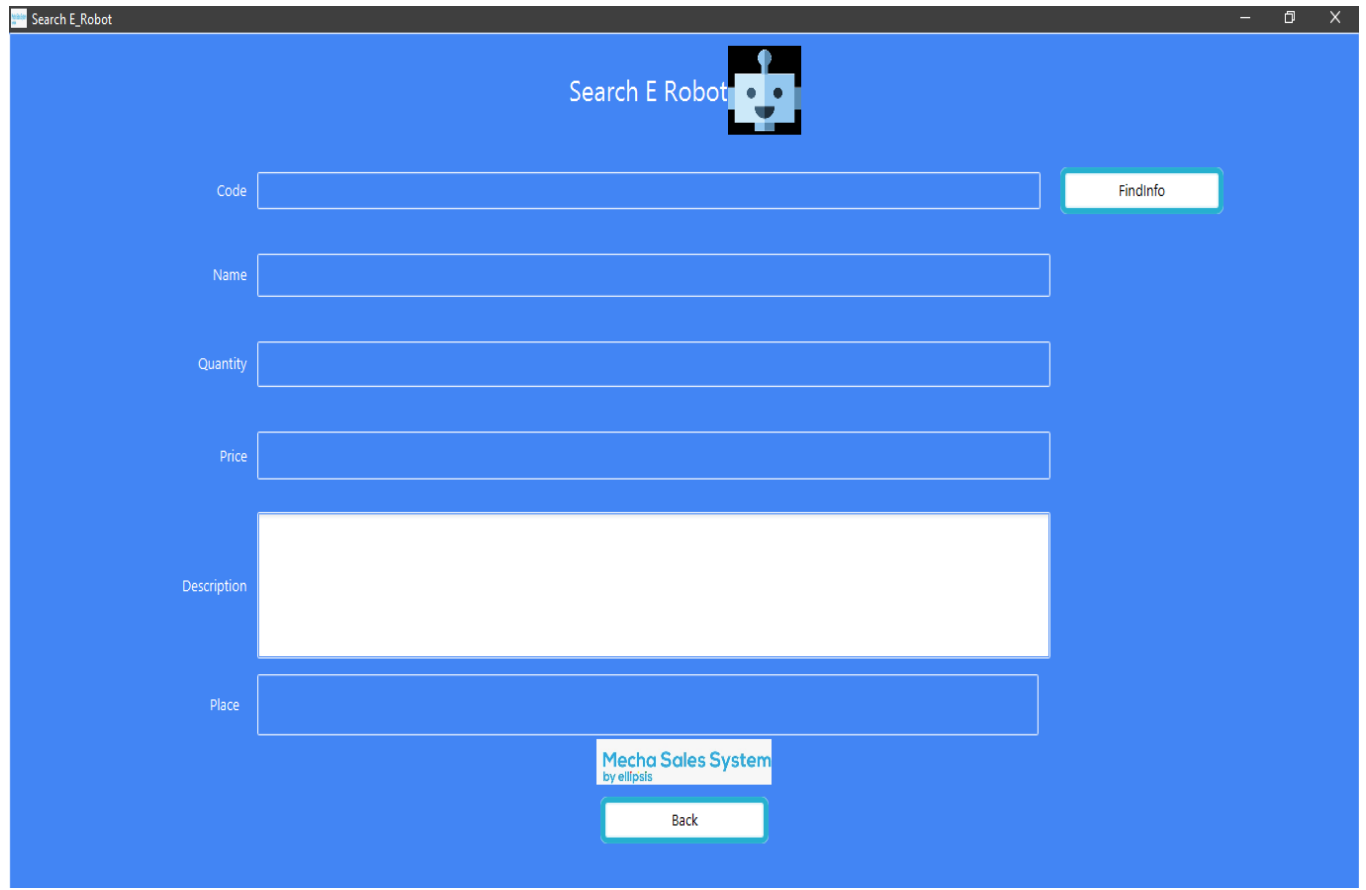


## Customer Screen UI





## Search Robot UI



The screenshot shows a web application window titled "Search E\_Robot". The interface has a blue background. At the top center, it says "Search E Robot" next to a small robot icon. Below this, there are several input fields: "Code", "Name", "Quantity", "Price", "Description" (a larger text area), and "Place". To the right of the "Code" field is a green button labeled "FindInfo". At the bottom center, there is a logo for "Mecha Sales System by ellipsis" and a green button labeled "Back".

Search E Robot

Code

Name

Quantity

Price

Description

Place

FindInfo


Mecha Sales System  
by ellipsis

Back

## Custom Demand UI

Search E\_Robot

Custom Demand



Customer Name

Customer Phone#

Quantity

Item Name

Description

Mecha Sales System  
by ellipsis

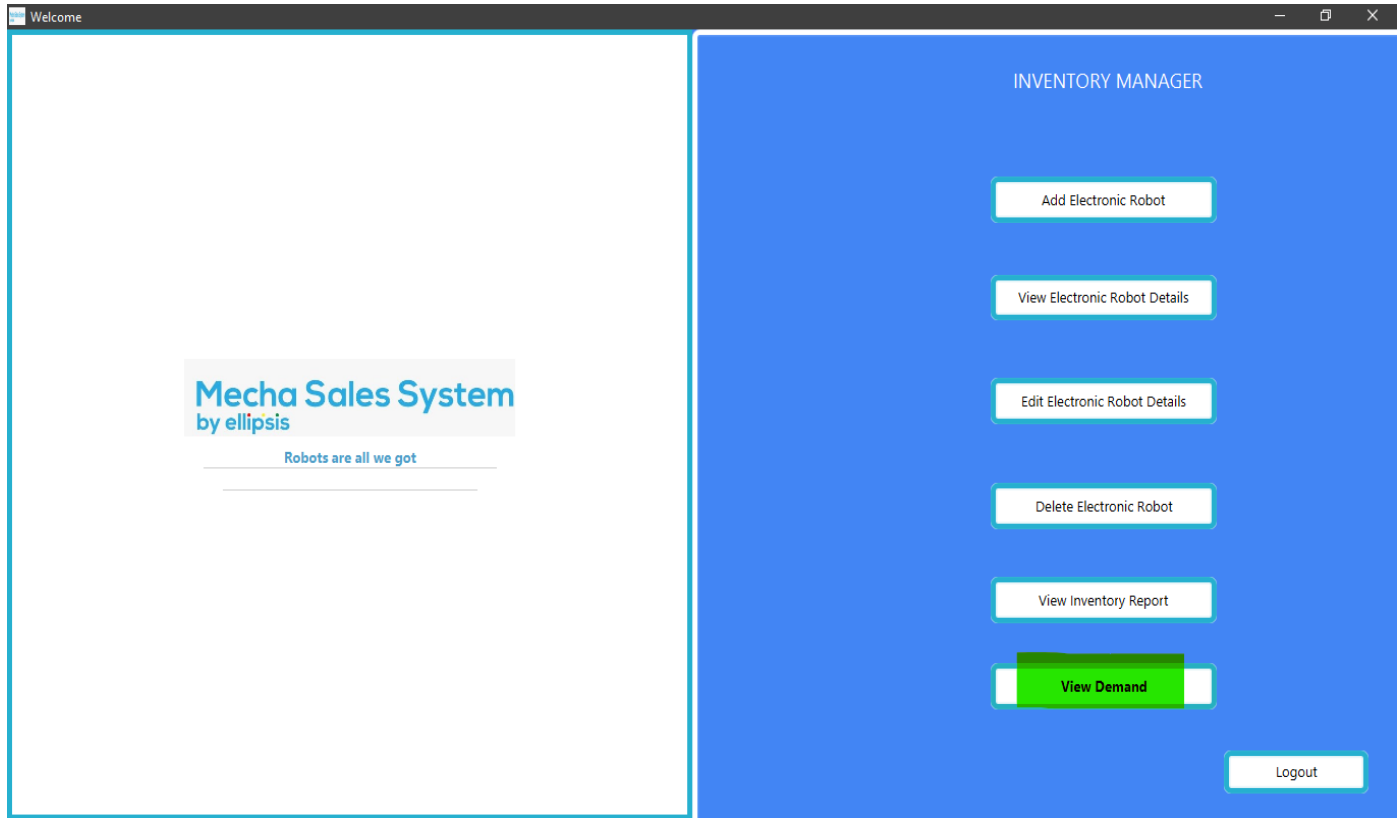
Add Demand

Back

## Backend

	Demand_Code	Demand_Qty	Customer_Name	Customer_Phone_no	Item_Name	Item_Desc
▶	7	5	shaheer	03368768462	car	remote control car
	8	50	adil	090078601	Drone	Remote Control Drone

## Inventory Manager View Demands UI

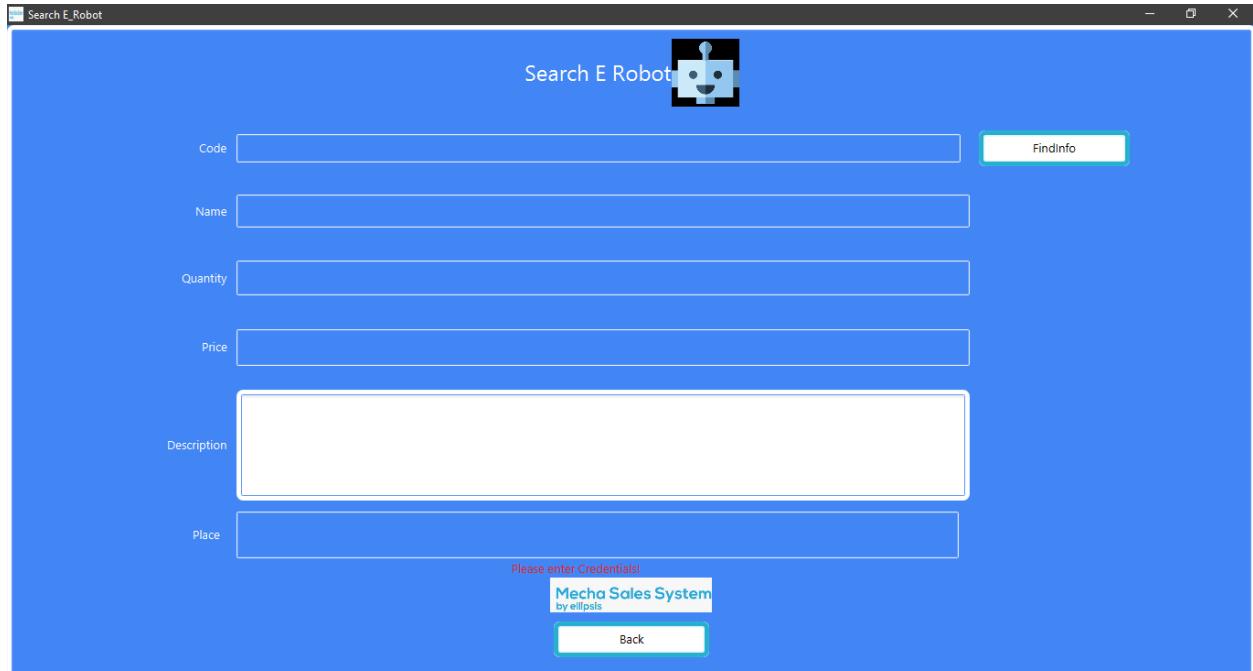


## Demand Report UI

[illegible]

Sprint 3 Testing (Search and Demand Module Testing):

Test Case 1:



The screenshot shows a web application window titled "Search E Robot". The interface has a blue background. At the top center, there is a logo of a robot head and the text "Search E Robot". Below the logo, there are several input fields: "Code", "Name", "Quantity", "Price", "Description" (a larger text area), and "Place". To the right of the "Code" field is a button labeled "FindInfo". At the bottom center, there is a red error message "Please enter Credentials!". Below the error message is a logo for "Mecha Sales System by ellipsis" and a button labeled "Back".

**Test Case 2:**

Search E Robot

Code

Name

Quantity

Price

Description

Place

**Invalid E Robot Code**


No Information Found! Please Enter Valid Robot Code

OK

Mecha Sales System  
by ellipsis

**Test Case 3:**

Custom Demand



Customer Name

Adil Hussain

Customer Phone#

090078601

Quantity

3

Item Name

Auto Mechanical Psycho

Description

Call for description

Mecha Sales System  
by ellipsis

Add Demand

Back


Demand Added

Demand Added

OK

**Test Case 4:**

Custom Demand



Customer Name

Customer Phone#

Quantity

Item Name

Description

Please enter Credentials!

Mecha Sales System

by ellipsis

Add Demand

Back