

REPORT TITLE

Pizza Store Application

Introduction

Welcome to our Pizza Store Application! This program is designed to streamline the process of managing the Pizza Store. Whether you're a pizza enthusiast looking for a hassle-free way to customize your favorite pizzas or a restaurant owner managing orders efficiently, our Pizza Store Application has got you covered.

How to Run the Program

Prerequisites:

1. Make sure you have Python installed on your system. You can download Python from <https://www.python.org/downloads/>
2. Ensure that the required libraries are installed. You can install them using the following command:
 - **pip install tk**

Running the Program:

1. **Download the Code:** Download the provided Python code and save it to a directory on your computer.
 2. **Navigate to the Directory:** Open a terminal or command prompt and navigate to the directory where you saved the Python code.
 3. **Run the Program:** Execute the following command to run the program:
 - **python presentation_layer.py**
 4. **Explore Different Roles:**
 - Upon running the program, a Tkinter GUI window will appear.
 - You will be presented with two main roles: "Owner" and "Customer."
-

-
- Click on the corresponding buttons to explore functionalities for each role.

Owner Role

Recipe Management:

1. Click on "Recipe Management" to manage pizza recipes.
2. You can add, update, view, and delete recipes.

Inventory Management:

1. Click on "Inventory Management" to manage ingredient inventory.
2. You can add, update, view, and delete ingredients.

Menu Management:

1. Click on "Menu Management" to manage the pizza store menu.
2. You can add, update, view, and delete menu items.

Customer Role

Place an Order:

1. Click on "Customer" to access the customer interface.
2. Explore the menu, add items to the order, and place an order.

Create Your Own Pizza:

1. In the customer interface, you can also create your own custom pizza with a choice of base, sauce, and ingredients.

Data Files

The program uses a data file named **data.json** to store recipes, inventory, and menu data.

Ensure that the data.json file is present in the same directory as the Python code.

The program will load existing data from this file, and modifications will be saved back to it.

By following these instructions, you should be able to run the pizza store management system and explore its functionalities for both the owner and customer roles.

Test Cases

Test Case 1: Owner - Load Recipes

Description:

Retrieves a listview of all the available recipes in the system.

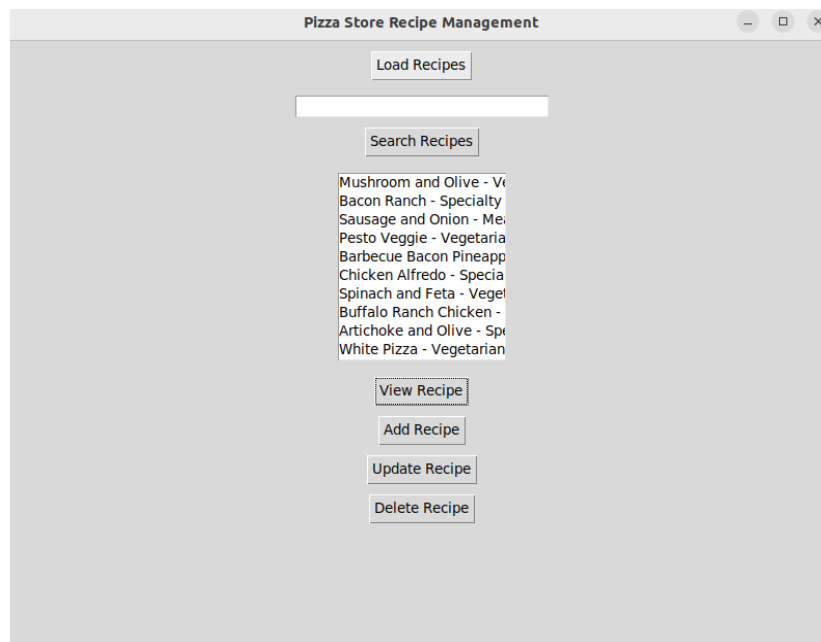
Steps:

- Click "Owner".
- Navigate to "Recipe Management."
- Click on "Load Recipe."

Expected Outcome:

- The system should successfully load a listview of all the available recipes.
- The recipe list should now be visible in the recipe list box.

Actual Outcome:



Test Case 2: Owner - Search Recipe

Description:

Attempt to search a pizza recipe in the system.

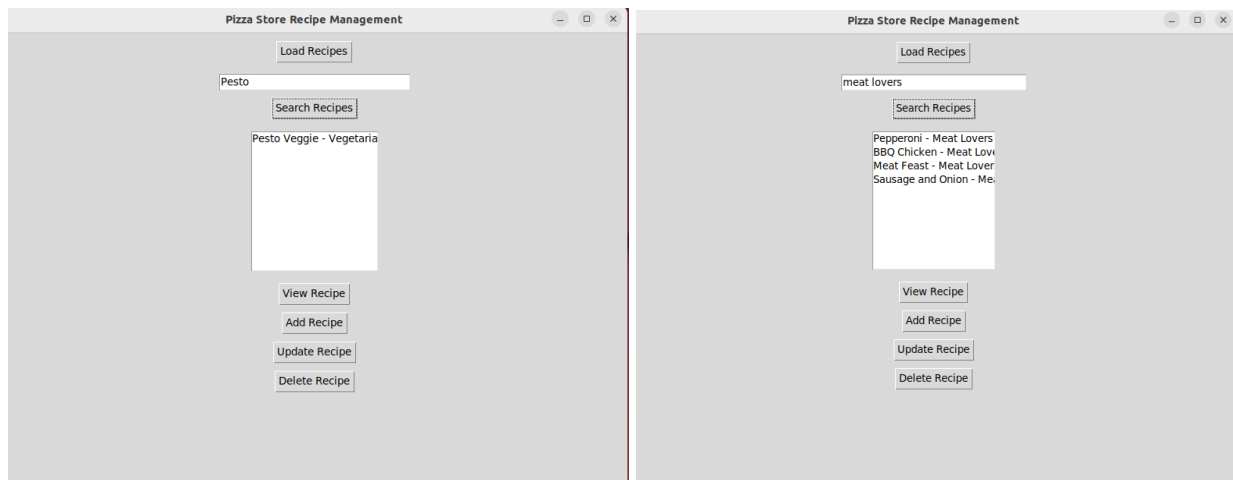
Steps:

- Click "Owner".
- Navigate to "Recipe Management."
- Write recipe name or category in the text field above the "Search Recipes" button.
- Click on "Search Recipes."

Expected Outcome:

- The system should display the searched recipe in the existing listbox.
- The searched recipe should now be visible in the recipe list.

Actual Outcome:



Test Case 3: Owner - Add Recipe

Description:

Attempt to add a new pizza recipe to the system.

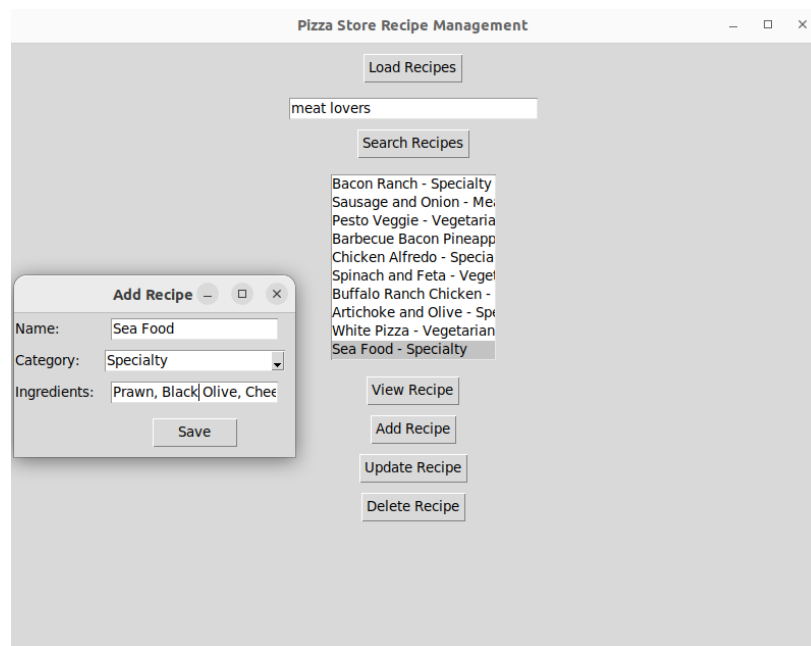
Steps:

- Click "Owner".
- Navigate to "Recipe Management."
- Click on "Add Recipe."
- Enter details for a new pizza recipe (e.g., name, base, sauce, ingredients).
- Click "Save."

Expected Outcome:

- The system should successfully add the new recipe.
- The recipe should now be visible in the recipe list.

Actual Outcome:



Test Case 4: Owner - View Recipe

Description:

View the available recipes in the system.

Steps:

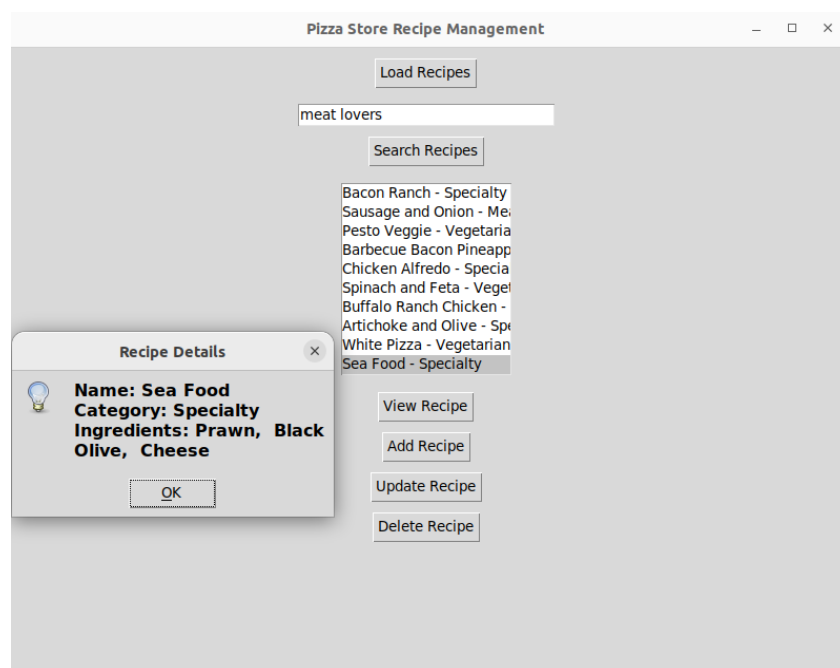
- Click "Owner".

- Navigate to "Recipe Management."
- Select a recipe to view from the listbox.
- Click on "View Recipe."

Expected Outcome:

- The system should successfully display the details of the selected recipe.
- The recipe details should now be visible.

Actual Outcome:



Test Case 5: Owner - Update Recipe

Description:

Update a recipe in the system.

Steps:

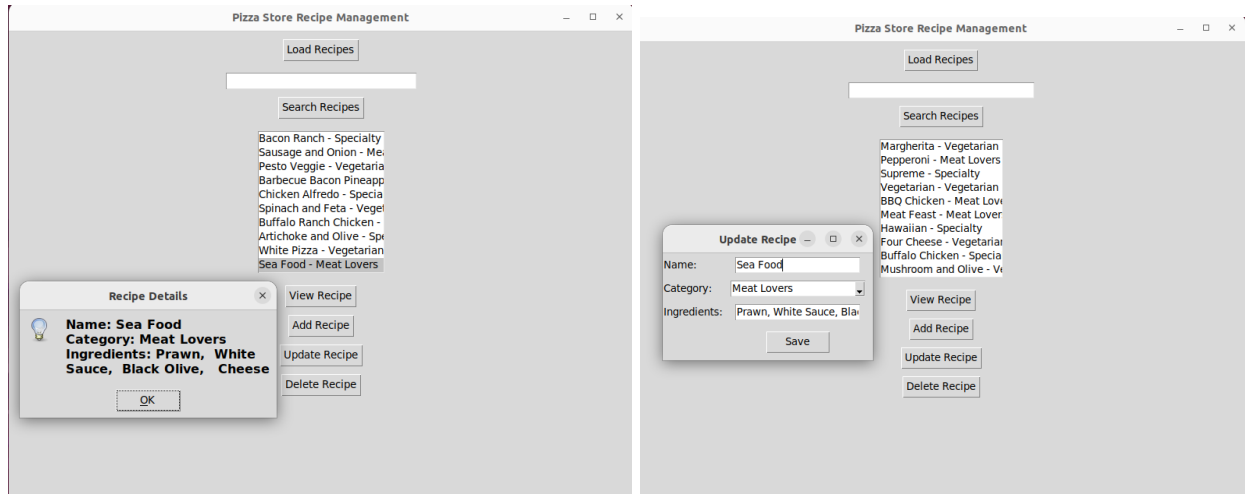
- Click "Owner".
- Navigate to "Recipe Management."
- Select a recipe to update from the listbox.
- Edit details for the selected pizza recipe.

- Click "Save."

Expected Outcome:

- The system should successfully update the details of the selected recipe.
- The recipe details should now be visible.

Actual Outcome:



Test Case 6: Owner - Delete Recipe

Description:

Delete a recipe in the system.

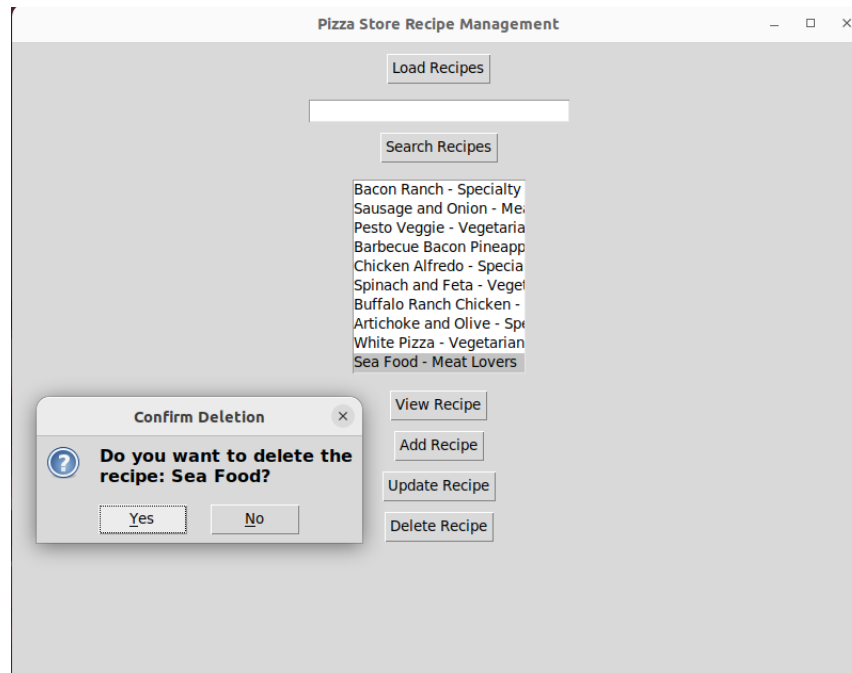
Steps:

- Click "Owner".
- Navigate to "Recipe Management."
- Select a recipe to delete from the listbox.
- Click on "Yes"

Expected Outcome:

- The system should successfully delete the selected recipe.

Actual Outcome:



Test Case 7: Owner - Load Inventory

Description:

Retrieves a table of all the available ingredients in the system.

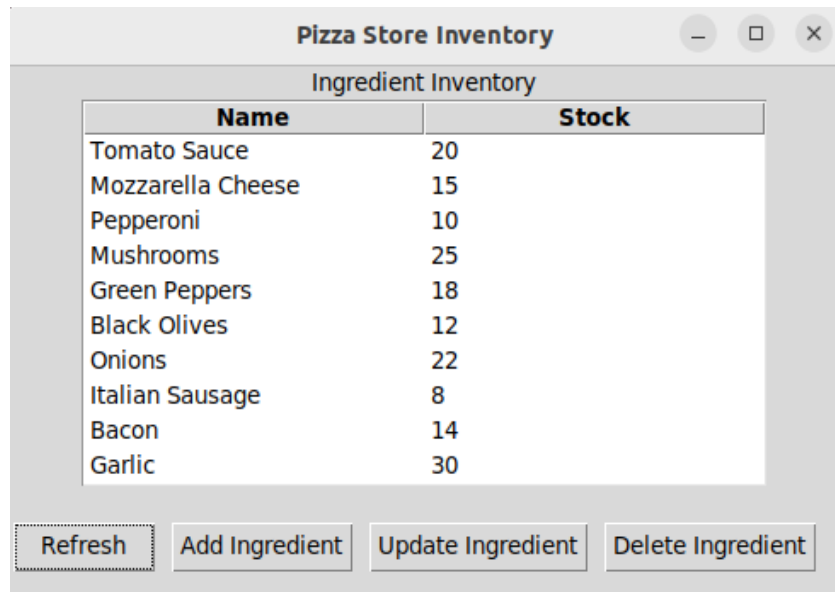
Steps:

- Click "Owner".
- Navigate to "Inventory Management."
- Click on "Refresh"

Expected Outcome:

- The system should successfully load a table containing all ingredients in the inventory.
- The inventory should now be visible in a table.

Actual Outcome:



Ingredient Inventory	
Name	Stock
Tomato Sauce	20
Mozzarella Cheese	15
Pepperoni	10
Mushrooms	25
Green Peppers	18
Black Olives	12
Onions	22
Italian Sausage	8
Bacon	14
Garlic	30

Refresh Add Ingredient Update Ingredient Delete Ingredient

Test Case 8: Owner - Add Inventory

Description:

Attempt to add an inventory to the system.

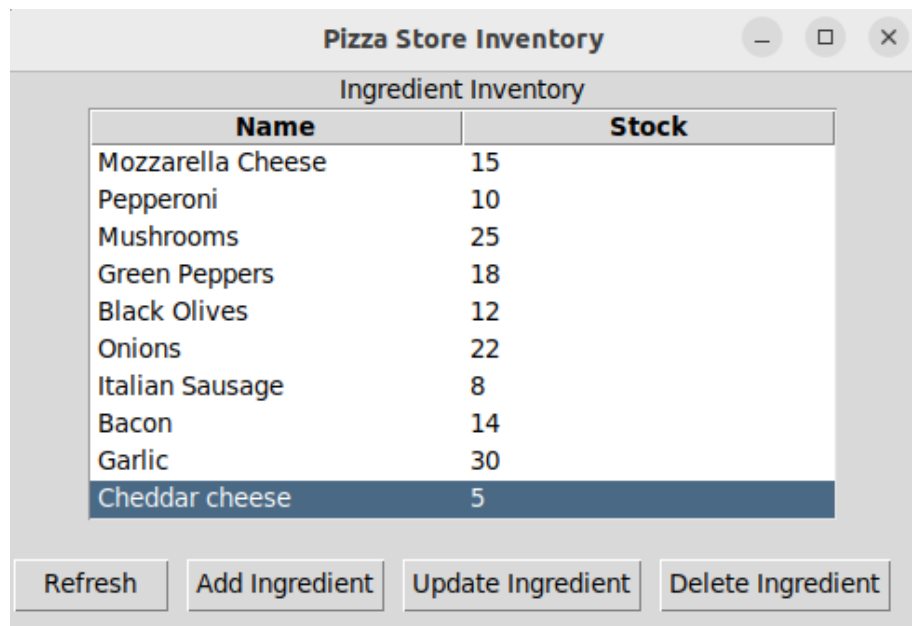
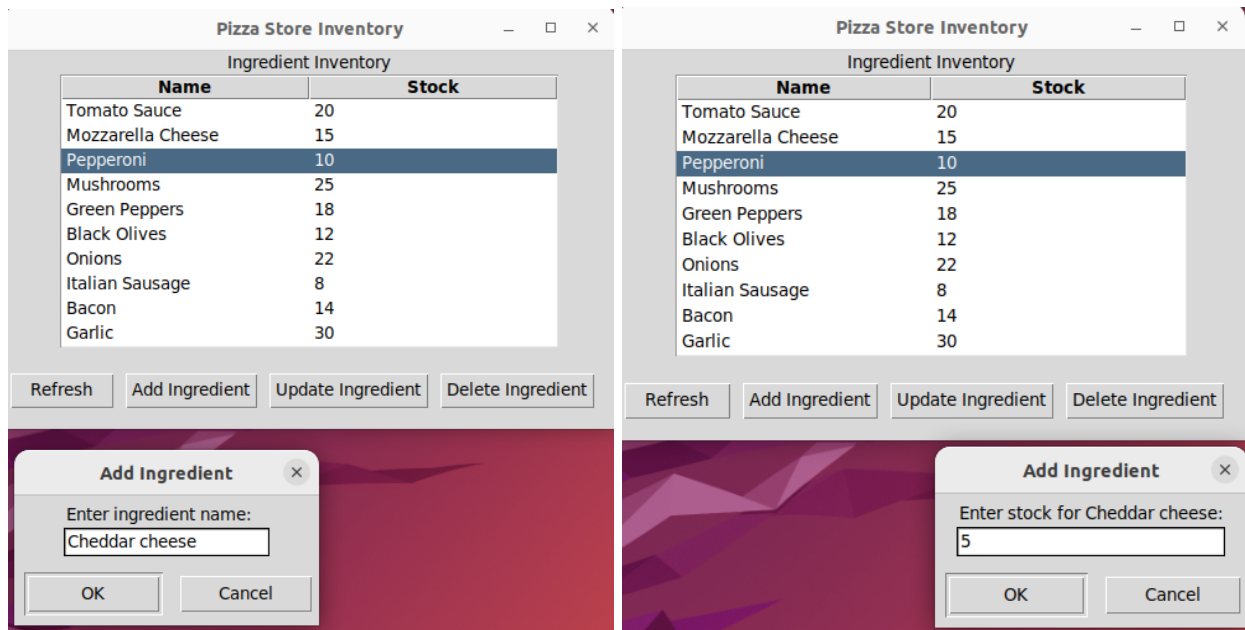
Steps:

- Click "Owner".
- Navigate to "Inventory Management."
- Click on "Add Ingredient."
- Click "Ok."
- Enter the amount for the ingredient.
- Click "Ok."

Expected Outcome:

- The system should successfully add the new ingredient to the inventory.
- The ingredient should now be visible on the Ingredient Inventory table.

Actual Outcome:



Test Case 9: Owner - Update Ingredient

Description:

Update an ingredient in the system.

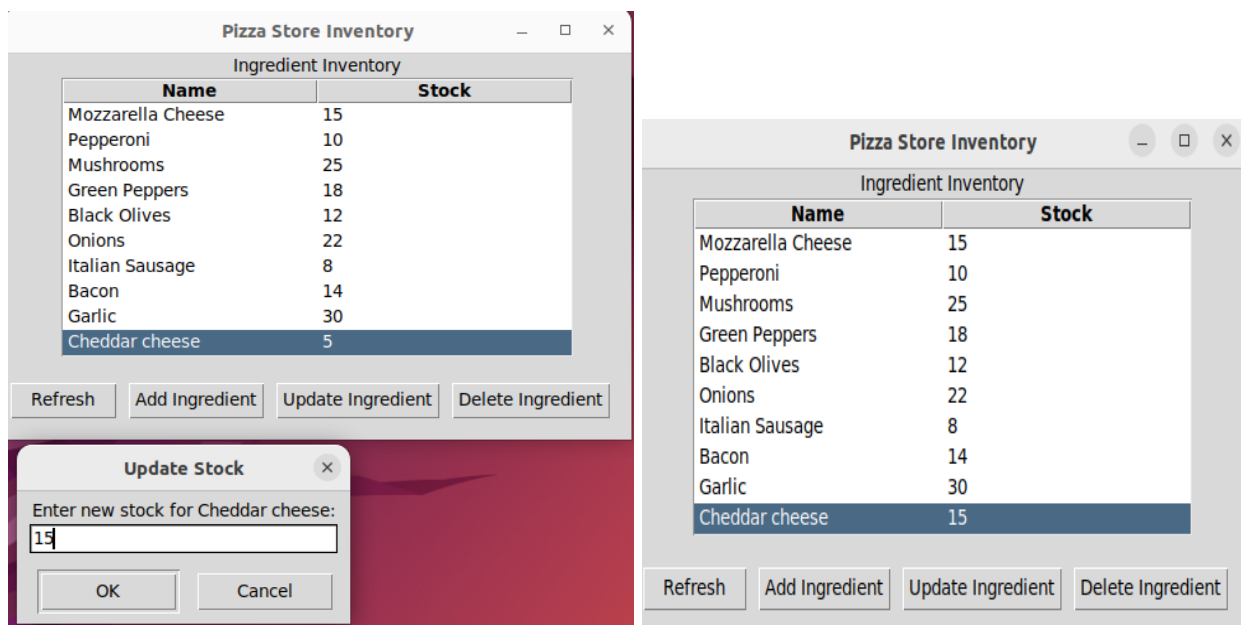
Steps:

- Click "Owner".
- Navigate to "Inventory Management."
- Select an ingredient to update from the table.
- Edit details for the selected ingredient.
- Click "Ok."

Expected Outcome:

- The system should successfully update the stock details of the selected ingredient.
- New stock details for the ingredient should now be visible.

Actual Outcome:



Test Case 10: Owner - Delete Ingredient

Description:

Delete an ingredient in the system.

Steps:

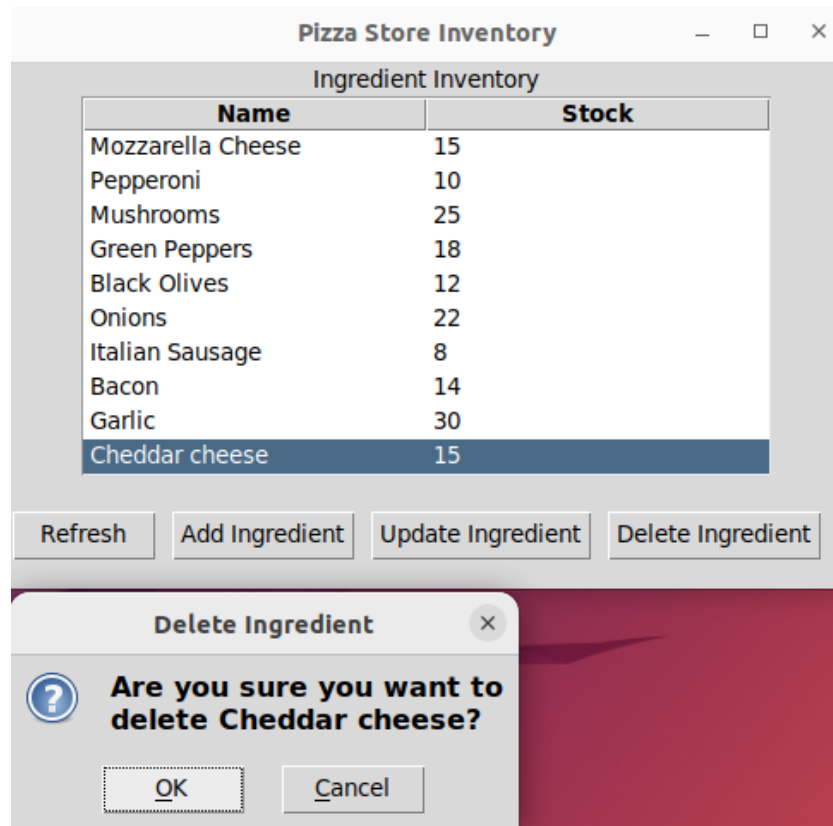
- Click "Owner".
- Navigate to "Inventory Management."

- Select an ingredient to delete from the Ingredient Inventory table.
- Click on "Ok"

Expected Outcome:

- The system should successfully delete the selected ingredient.

Actual Outcome:



Test Case 11: Owner - Load Menu Items

Description:

Retrieves a table of all the available pizzas and side dishes in the system.

Steps:

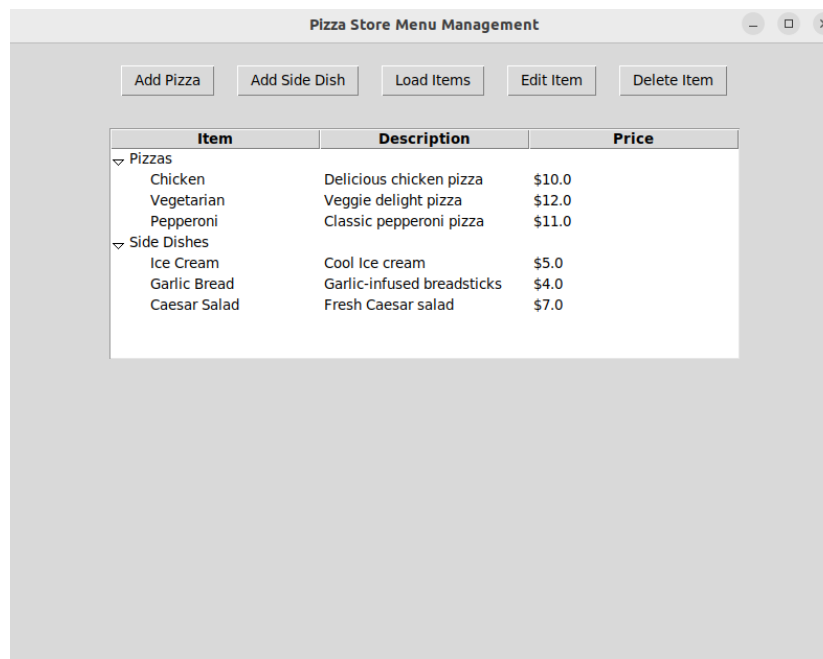
- Click "Owner".
- Navigate to "Menu Management."

-
- Click on "Load Items"

Expected Outcome:

- The system should successfully load a table containing all Pizzas and Side dishes in the menu.
- The menu should now be visible in a table.

Actual Outcome:



Item	Description	Price
▼ Pizzas		
Chicken	Delicious chicken pizza	\$10.0
Vegetarian	Veggie delight pizza	\$12.0
Pepperoni	Classic pepperoni pizza	\$11.0
▼ Side Dishes		
Ice Cream	Cool Ice cream	\$5.0
Garlic Bread	Garlic-infused breadsticks	\$4.0
Caesar Salad	Fresh Caesar salad	\$7.0

Test Case 12: Owner - Add Pizza

Description:

Attempt to add a pizza to the menu.

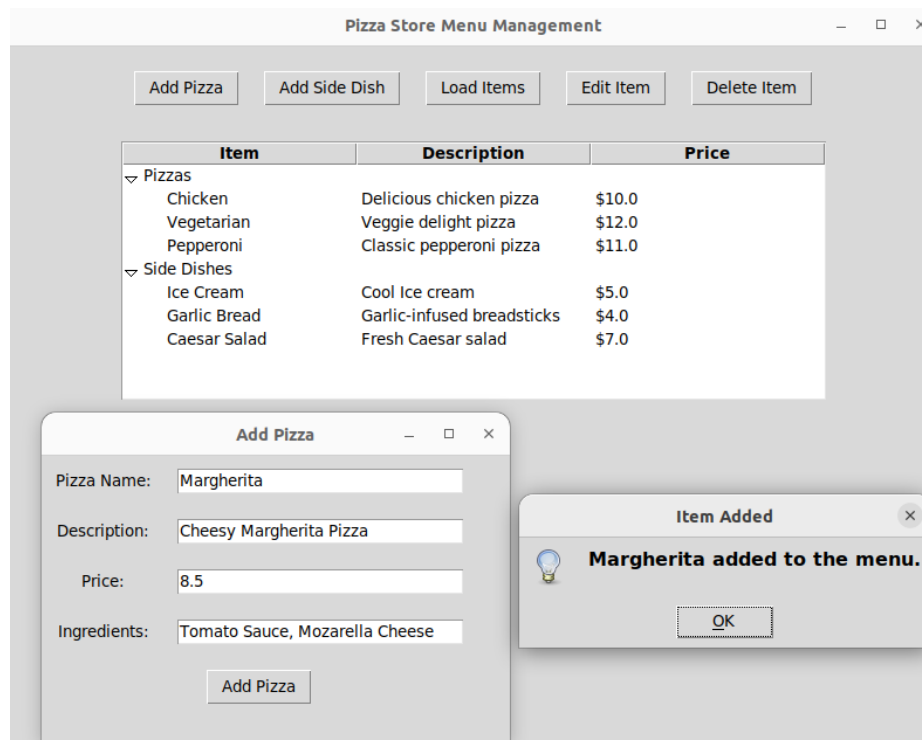
Steps:

- Click "Owner".
- Navigate to "Menu Management."
- Click on "Add Pizza."
- Enter details for a new pizza(e.g., name, description, price, ingredients).
- Click on "Add Pizza."

Expected Outcome:

- The system should successfully add the new pizza to the menu.
- The pizza should now be visible on the table.

Actual Outcome:



Test Case 13: Owner - Add Side Dish

Description:

Attempt to add a side dish to the menu.

Steps:

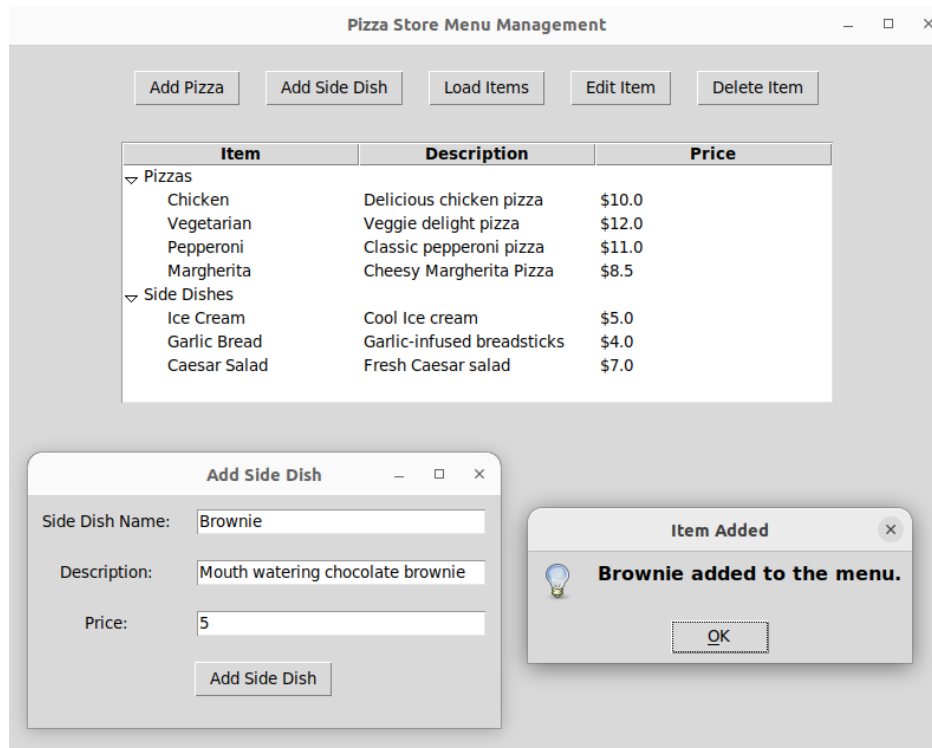
- Click "Owner".
- Navigate to "Menu Management."
- Click on "Add Side Dish."
- Enter details for a new size dish (e.g., name, description, price).

- Click on "Add Side Dish."

Expected Outcome:

- The system should successfully add the new side dish to the menu.
- The side dish should now be visible on the table.

Actual Outcome:



Test Case 14: Owner - Update Item

Description:

Update an item in the menu.

Steps:

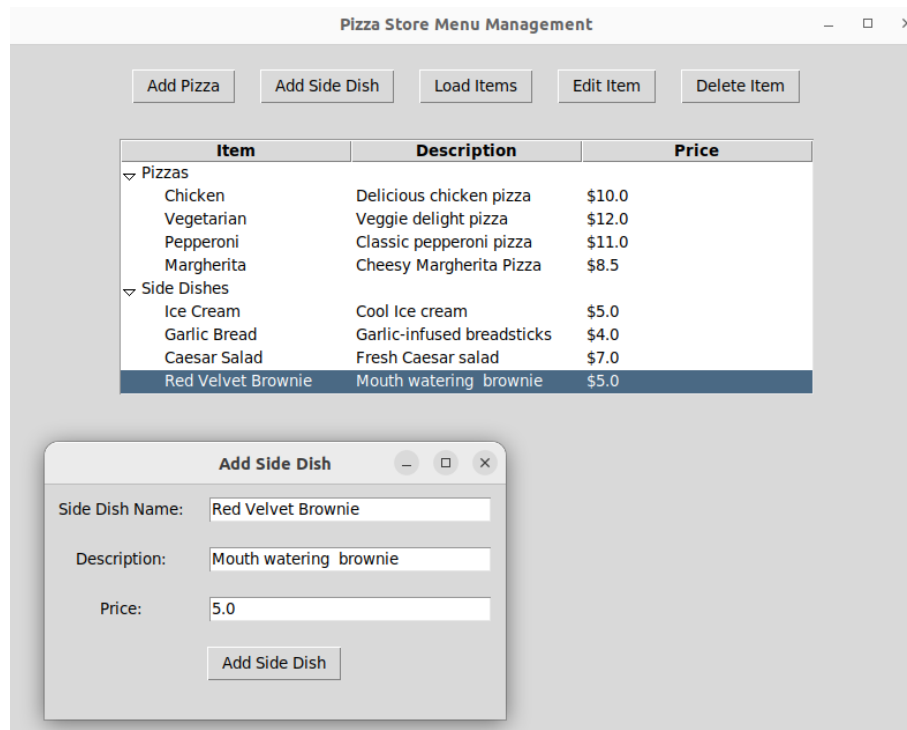
- Click "Owner".
- Navigate to "Menu Management."
- Select an item to update from the table.
- Edit details for the selected item.

- Click "Save."

Expected Outcome:

- The system should successfully update the details of the selected item.
- The item details should now be visible in the menu.

Actual Outcome:



Test Case 15: Owner - Delete Item

Description:

Delete an item from the menu.

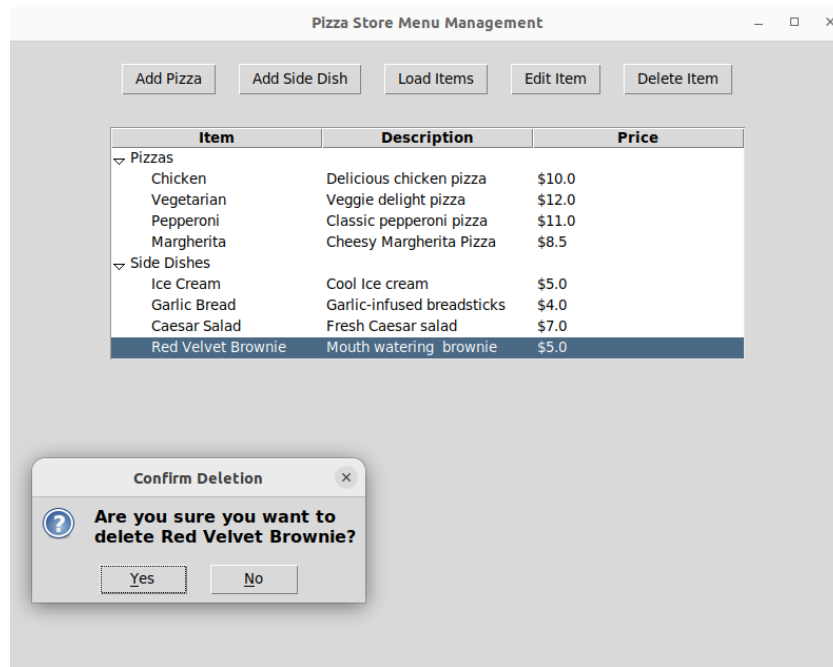
Steps:

- Click "Owner".
- Navigate to "Menu Management."
- Select an item to delete from the table.
- Click on "Ok"

Expected Outcome:

- The system should successfully delete the selected item from the menu.

Actual Outcome:



Test Case 16: Customer - View Menu

Description:

View the menu of Pizza Store Application.

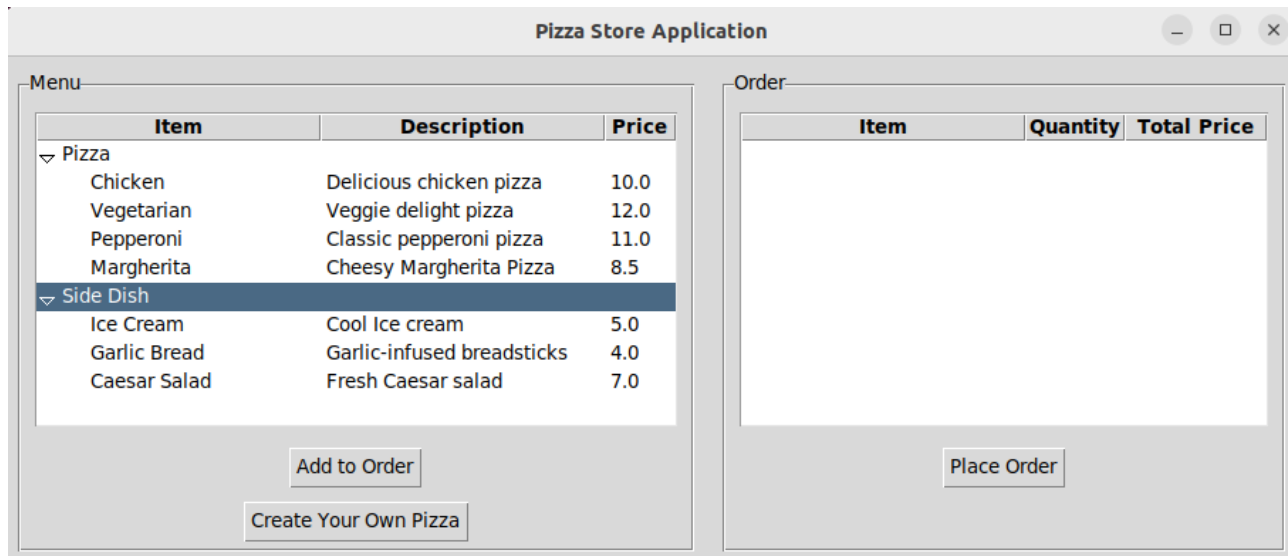
Steps:

- Click "Customer".
- Navigate to "Menu."

Expected Outcome:

- The system should successfully display the menu.
- The menu details should now be visible.

Actual Outcome:



Test Case 16: Customer - Create Customized Pizza

Description:

Create a customized pizza to order.

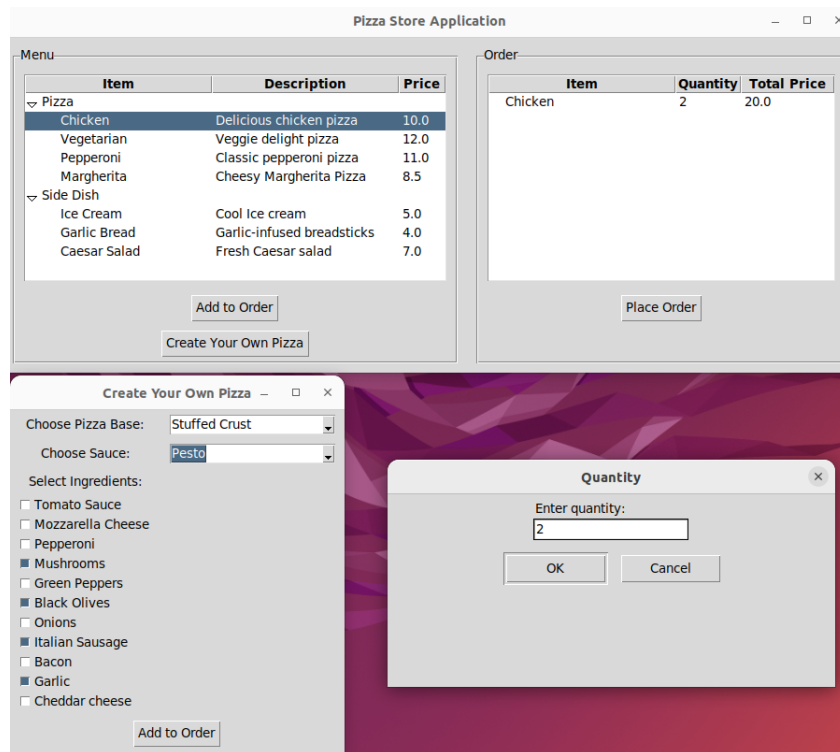
Steps:

- Click "Customer".
- Navigate to "Menu."
- Click on "Create Your Own Pizza"
- Enter the details of the customized pizza
- Select an item from the table.
- Click on "Add to Order"
- Enter the quantity
- Click "Ok".

Expected Outcome:

- The system should successfully add the customized pizza to the order.

Actual Outcome:



Test Case 15: Customer - Create Order

Description:

Create an order.

Steps:

- Click "Customer".
- Navigate to "Menu."
- Select an item from the table.
- Enter the quantity.
- Click on "Add to Order"
- Click on "Place Order"

Expected Outcome:

- The system should successfully place the order.
- The billing details should now be visible.

Actual Outcome:

