Contents

[Test Plan Introduction 3](#_Toc103243827)

[Test Strategy 3](#_Toc103243828)

[Test Objectives 3](#_Toc103243829)

[Test Assumptions 3](#_Toc103243830)

[Scope of Testing 3](#_Toc103243831)

[Front End Testing 3](#_Toc103243832)

[Unit Testing 3](#_Toc103243833)

[Defect Severity Categorisation 4](#_Toc103243834)

[Front End Testing for User Stories 5](#_Toc103243835)

[Sprint 1 5](#_Toc103243836)

[User Story #1 5](#_Toc103243837)

[User Story #2 6](#_Toc103243838)

[User Story #13 7](#_Toc103243839)

[User Story #14 8](#_Toc103243840)

[User Story #20 9](#_Toc103243841)

[User Story #21 10](#_Toc103243842)

[User Story #35 11](#_Toc103243843)

[User Story #36 12](#_Toc103243844)

[Sprint 2 13](#_Toc103243845)

[User Story #3 and #8 13](#_Toc103243846)

[User Story #4 and #9 14](#_Toc103243847)

[User Story #5 and #10 15](#_Toc103243848)

[User Story #6 and #11 16](#_Toc103243849)

[User Story #7 and #12 17](#_Toc103243850)

[User Story #27 18](#_Toc103243851)

[Sprint 3 19](#_Toc103243852)

[User Story #28 19](#_Toc103243853)

[User Story #29 19](#_Toc103243854)

[User Story #30 20](#_Toc103243855)

[User Story #31 21](#_Toc103243856)

[User Story #32 22](#_Toc103243857)

[User Story #33 22](#_Toc103243858)

[User Story #425 23](#_Toc103243859)

[User Story #34 24](#_Toc103243860)

[User Story #37 25](#_Toc103243861)

[User Story #38 25](#_Toc103243862)

[User Story #39 25](#_Toc103243863)

[User Story #40 26](#_Toc103243864)

[User Story #41 26](#_Toc103243865)

[User Story #42 26](#_Toc103243866)

[User Story #43 27](#_Toc103243867)

[User Story #44 27](#_Toc103243868)

[User Story #45 27](#_Toc103243869)

[Backlog 28](#_Toc103243870)

[User Story #15 28](#_Toc103243871)

[User Story #16 29](#_Toc103243872)

[User Story #17 30](#_Toc103243873)

[User Story #18 30](#_Toc103243874)

[User Story #19 30](#_Toc103243875)

[User Story #420 31](#_Toc103243876)

[User Story #421 32](#_Toc103243877)

[User Story #422 33](#_Toc103243878)

[User Story #423 33](#_Toc103243879)

[User Story #424 33](#_Toc103243880)

[User Story #22 34](#_Toc103243881)

[User Story #23 34](#_Toc103243882)

[User Story #24 35](#_Toc103243883)

[User Story #25 35](#_Toc103243884)

[User Story #26 36](#_Toc103243885)

[Unit Testing for Entities 37](#_Toc103243886)

[Staff class 37](#_Toc103243887)

[User\_Admin class 38](#_Toc103243888)

[Customer class 38](#_Toc103243889)

[Cart class 38](#_Toc103243890)

[Restaurant\_Owner class 39](#_Toc103243891)

[Menu\_Item class 39](#_Toc103243892)

[Restaurant\_Manager class 40](#_Toc103243893)

# Test Plan Introduction

The objective of this test plan is to show how testing will be done for the Hopium program.

# Test Strategy

## Test Objectives

The objective of testing is to ensure that the program works as intended and that there are no critical errors that will affect the end user.

## Test Assumptions

Key Assumptions

* Testing Data will include a small amount of data for initial testing and a larger production like amount of data for final testing
* Performance will not be considered for testing
* Any defects will be reported to the developer team to fix

## Scope of Testing

### Front End Testing

**Purpose:** Front end testing will focus on what the end user will see and experience as they use the program.

**Scope:** Testing will be mainly focused on GUI elements

**Method:** Testing will be done manually on the program’s GUI based on what the user will do for each user story. Results will be compared to expected results and any bugs will be reported to the programming team

**Frequency of Tests:** These tests will be done whenever a new feature is implemented into the program to test the new functionality and to make sure it doesn’t not break existing code.

### Unit Testing

**Purpose:** Testing will be performed to check the functions of the program. Testing is done by feeding the functions inputs and checking the output of the functions.

**Scope:** Testing will mainly be done on the Controller and Entity classes to check if they can retrieve the correct information from the database.

**Method:** Tests will be mainly performed using Junit 5 scripts to run automated tests on a per class basis. Any errors will be reported to the programming team

**Frequency of Tests:** These tests can be run more often than Front End Testing, so this test can be run whenever code is adjusted or new functionality is added.

# Defect Severity Categorisation

Defects will be given a severity level to show the severity of the defect and the priority in fixing the defect.

|  |  |  |
| --- | --- | --- |
| Severity | Priority | Impact |
| Critical | 1 | Defect that can crash the program |
| High | 2 | Defect that disables a function of the program |
| Medium | 3 | Defect that disables a function of the program but has a workaround |
| Low | 4 | Defect that has minimal impact on functionality or purely cosmetic |

# Front End Testing for User Stories

## Sprint 1

### User Story #1

As a User Admin, I want to login so that I can manage user accounts

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Starting page is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 1.1 | Login with valid User Admin credentials | Test if user admin can login with valid credentials |
| 1.2 | Login with invalid User Admin credentials | Test if user admin can login without valid credentials |
| 1.3 | Login with valid User Admin credentials but wrong profile | Test if user admin can login if choosing the wrong profile |

|  |  |
| --- | --- |
| **Test Case 1.1 – Login with valid User Admin credentials** | |
| **Steps** | 1. Click ‘Staff’ button 2. Type in valid credentials (user\_admin1 for both fields) 3. Select ‘User Admin’ in profile dropdown menu 4. Click ‘Login’ button |
| **Expected Results** | Logged in and User Admin homepage displayed |

|  |  |
| --- | --- |
| **Test Case 1.2 – Login with invalid User Admin credentials** | |
| **Steps** | 1. Click ‘Staff’ button 2. Type in invalid credentials (user\_admin0 for both fields) 3. Select ‘User Admin’ in profile dropdown menu 4. Click ‘Login’ button |
| **Expected Results** | Failed Login |

|  |  |
| --- | --- |
| **Test Case 1.3 – Login with valid User Admin credentials but wrong profile** | |
| **Steps** | 1. Click ‘Staff’ button 2. Type in valid credentials (user\_admin1 for both fields) 3. Select wrong profile in the profile dropdown menu (Restaurant Manager selected) 4. Click ‘Login’ button |
| **Expected Results** | Failed Login |

### 

### User Story #2

As a User Admin, I want to logout so that other people can login and for security purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as User Admin 3. User Admin Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 2.1 | Log out from the User Admin account | Test if the user admin can log out of the account |

|  |  |
| --- | --- |
| **Test Case 2.1 – Log out from the User Admin account** | |
| **Steps** | 1. While in the User Admin homepage, click the log out button |
| **Expected Results** | Logged out and returned to the login page |

### 

### User Story #13

As a Restaurant Manager, I want to login so that I can create/modify menu items, coupons, etc.

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 13.1 | Login with valid Restaurant Manager credentials | Test if restaurant manager can login with valid credentials |
| 13.2 | Login with invalid Restaurant Manager credentials | Test if restaurant manager can login without valid credentials |
| 13.3 | Login with valid Restaurant Manager credentials but wrong profile | Test if restaurant manager can login if choosing the wrong profile |

|  |  |
| --- | --- |
| **Test Case 13.1 – Login with valid Restaurant Manager credentials** | |
| **Steps** | 1. Start Application 2. Click 'Staff' button 3. Type in valid username and password (manager1 for both fields) 4. Select Restaurant Manager in the profile login dropdown menu 5. Click 'Login' button |
| **Expected Results** | Logged in and Restaurant Manager homepage displayed |

|  |  |
| --- | --- |
| **Test Case 13.2 – Login with invalid Restaurant Manager credentials** | |
| **Steps** | 1. Start Application 2. Click 'Staff' button 3. Type in invalid username and password (manager0 for both fields) 4. Select Restaurant Manager in the profile login dropdown menu 5. Click 'Login' button |
| **Expected Results** | Failed Login |

|  |  |
| --- | --- |
| **Test Case 13.3 – Login with valid Restaurant Manager credentials but wrong profile** | |
| **Steps** | 1. Start Application 2. Click 'Staff' button 3. Type in valid username and password (manager1 for both fields) 4. Select wrong profile in the profile login dropdown menu (User Admin selected) 5. Click 'Login' button |
| **Expected Results** | Failed Login |

### User Story #14

As a Restaurant Manager, I want to logout so that other people can login and for security purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 14.1 | Log out from the Restaurant Manager account | Test if the restaurant manager can log out of the account |

|  |  |
| --- | --- |
| **Test Case 14.1 – Log out from the Restaurant Manager account** | |
| **Steps** | 1. While in the Restaurant Manager homepage, click the log out button |
| **Expected Results** | Logged out and returned to the login page |

### User Story #20

As a Restaurant Staff, I want to login so that I can manage and fulfil orders

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Starting page is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 20.1 | Login with valid Restaurant Staff credentials | Test if restaurant staff can login with valid credentials |
| 20.2 | Login with invalid Restaurant Staff credentials | Test if restaurant staff can login without valid credentials |
| 20.3 | Login with valid Restaurant Staff credentials but wrong profile | Test if restaurant staff can login if choosing the wrong profile |

|  |  |
| --- | --- |
| **Test Case 20.1 – Login with valid Restaurant Staff credentials** | |
| **Steps** | 1. Start application 2. Click ‘Staff’ button 3. Type in valid credentials (user1 for both fields) 4. Select ‘Restaurant Staff’ in profile dropdown menu 5. Click ‘Login’ button |
| **Expected Results** | Logged in and Restaurant Staff homepage displayed |

|  |  |
| --- | --- |
| **Test Case 20.2 – Login with invalid Restaurant Staff credentials** | |
| **Steps** | 1. Start application 2. Click ‘Staff’ button 3. Type in invalid credentials (user0 for both fields) 4. Select ‘Restaurant Staff’ in profile dropdown menu 5. Click ‘Login’ button |
| **Expected Results** | Failed Login |

|  |  |
| --- | --- |
| **Test Case 20.3 – Login with valid Restaurant Staff credentials but wrong profile** | |
| **Steps** | 1. Start application 2. Click ‘Staff’ button 3. Type in valid credentials (user1 for both fields) 4. Select wrong profile in the profile dropdown menu (User Admin selected) 5. Click ‘Login’ button |
| **Expected Results** | Failed Login |

### User Story #21

As a Restaurant Staff, I want to logout so that other people can login and for security purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Staff 3. Restaurant Staff Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 21.1 | Log out from the Restaurant Staff account | Test if the restaurant staff can log out of the account |

|  |  |
| --- | --- |
| **Test Case 21.1 – Log out from the Restaurant Staff account** | |
| **Steps** | 1. While in the Restaurant Staff homepage, click the log out button |
| **Expected Results** | Logged out and returned to the login page |

### User Story #35

As a Restaurant Owner, I want to login to my account, so that I can view and manage the data

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Starting page is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 35.1 | Login with valid Restaurant Owner credentials | Test if restaurant owner can login with valid credentials |
| 35.2 | Login with invalid Restaurant Owner credentials | Test if restaurant owner can login without valid credentials |
| 35.3 | Login with valid Restaurant Owner credentials but wrong profile | Test if restaurant owner can login if choosing the wrong profile |

|  |  |
| --- | --- |
| **Test Case 35.1 – Login with valid credentials** | |
| **Steps** | 1. Start application 2. Click ‘Staff’ button 3. Type in valid credentials (owner1 for both fields) 4. Select ‘Restaurant Owner’ in profile dropdown menu 5. Click ‘Login’ button |
| **Expected Results** | Logged in and Restaurant Owner homepage displayed |

|  |  |
| --- | --- |
| **Test Case 35.2 – Login with invalid credentials** | |
| **Steps** | 1. Start application 2. Click ‘Staff’ button 3. Type in invalid credentials (owner0 for both fields) 4. Select ‘Restaurant Owner’ in profile dropdown menu 5. Click ‘Login’ button |
| **Expected Results** | Failed Login |

|  |  |
| --- | --- |
| **Test Case 35.3 – Login with valid credentials but wrong profile** | |
| **Steps** | 1. Start application 2. Click ‘Staff’ button 3. Type in valid credentials (owner1 for both fields) 4. Select wrong profile in the profile dropdown menu (User Admin selected) 5. Click ‘Login’ button |
| **Expected Results** | Failed Login |

### User Story #36

As a Restaurant Owner, I want to logout of my account, so that other people can login and for security purposes.

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 36.1 | Log out from the Restaurant Owner account | Test if the restaurant owner can log out of the account |

|  |  |
| --- | --- |
| **Test Case 36.1 – Log out from the Restaurant Owner account** | |
| **Steps** | 1. While in the Restaurant Owner homepage, click the log out button |
| **Expected Results** | Logged out and returned to the login page |

## Sprint 2

### User Story #3 and #8

As a User Admin, I want to create a user account so that I can create an account for a new employee As a User Admin, I want to create a user profile so that I can create a profile for a new employee

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as User Admin 3. User Admin Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 3.1 | Create with new credentials | Test if the program can create a new account |
| 3.2 | Create with existing credentials | Test if the program can create a duplicate account |
| 3.3 | Create with missing credentials | Test if the program will warn the user of empty fields |

|  |  |
| --- | --- |
| **Test Case 3.1 – Create with new credentials** | |
| **Steps** | 1. Click the ‘Create’ button 2. Type in new user account details 3. Click the ‘Create Account’ button |
| **Expected Results** | The program will show a window telling the user a new account is created |

|  |  |
| --- | --- |
| **Test Case 3.2 – Create with existing credentials** | |
| **Steps** | 1. Click the ‘Create’ button 2. Type in duplicate user account details 3. Click the ‘Create Account’ button |
| **Expected Results** | The program will show a window telling the user that the account already exists |

|  |  |
| --- | --- |
| **Test Case 3.3 – Create with missing credentials** | |
| **Steps** | 1. Click the ‘Create’ button 2. Type only username or password or neither 3. Click the ‘Create Account’ button |
| **Expected Results** | The program will show a window telling the user that fields should not be left empty |

### User Story #4 and #9

As a User Admin, I want to edit user accounts, so that I can change and update the user account information

As a User Admin, I want to edit user profiles, so that I can change and update the user profile information

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as User Admin 3. User Admin Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 4.1 | Edit User Account | Test if the program can edit a user’s account details |
| 4.2 | Edit without selecting user | Test if the program can warn that no user is selected |
| 4.3 | Edit with empty fields | Test if the program can warn that there are empty fields |

|  |  |
| --- | --- |
| **Test Case 4.1 – Edit User Account** | |
| **Steps** | 1. Click ‘Edit’ button 2. Select a User Account 3. Modify the details in the fields 4. Click ‘Edit Changes’ button |
| **Expected Results** | The program will show a window telling the user that the edits are saved |

|  |  |
| --- | --- |
| **Test Case 4.2 – Edit details without selecting user** | |
| **Steps** | 1. Click ‘Edit’ button 2. Modify the details without selecting user 3. Click ‘Edit Changes’ button |
| **Expected Results** | The program will warn the user that the account does not exist |

|  |  |
| --- | --- |
| **Test Case 4.3 – Edit with empty fields** | |
| **Steps** | 1. Click ‘Edit’ button 2. Select a User Account 3. Clear the username, password, user profile fields 4. Click ‘Edit Changes’ button |
| **Expected Results** | The program will warn the user that there are empty fields |

### User Story #5 and #10

As a User Admin, I want to be able to search for user accounts so that I can look for specific user accounts

As a User Admin, I want to be able to search for user profiles so that I can look for specific user profiles

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as User Admin 3. User Admin Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 5.1 | Search for existing user by Username | Test if the program can search for an existing user by username |
| 5.2 | Search for non-existent user by Username | Test if the program can tell the admin the user does not exist by username |
| 5.3 | Search for existing user by User Profile | Test if the program can search for an existing user by user profile |
| 5.4 | Search for non-existent user by User Profile | Test if the program can tell the admin the user does not exist by user profile |

|  |  |
| --- | --- |
| **Test Case 5.1 – Search for existing user by Username** | |
| **Steps** | 1. Click ‘View’ button 2. Type an existing username 3. Click ‘Search by Username’ button |
| **Expected Results** | The program will show the list of user accounts that match the criteria |

|  |  |
| --- | --- |
| **Test Case 5.2 – Search for non-existent user by Username** | |
| **Steps** | 1. Click ‘View’ button 2. Type a non-existing username 3. Click ‘Search by Username’ button |
| **Expected Results** | The program will show an empty list |

|  |  |
| --- | --- |
| **Test Case 5.3 – Search for existing user by User Profile** | |
| **Steps** | 1. Click ‘View’ button 2. Type an existing user profile 3. Click ‘Search by User Profile’ button |
| **Expected Results** | The program will show the list of user accounts that match the criteria |

|  |  |
| --- | --- |
| **Test Case 5.4 – Search for non-existent user by User Profile** | |
| **Steps** | 1. Click ‘View’ button 2. Type a non-existing user profile 3. Click ‘Search by User Profile’ button |
| **Expected Results** | The program will show an empty list |

### User Story #6 and #11

As a User Admin, I want to view user accounts so that I can retrieve user account information

As a User Admin, I want to view user profiles so that I can retrieve user profile information

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as User Admin 3. User Admin Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 6.1 | View user account information | Test if the program can display user account details |

|  |  |
| --- | --- |
| **Test Case 6.1 – View all accounts** | |
| **Steps** | 1. Click ‘View’ button |
| **Expected Results** | The program will show the list of user accounts |

### User Story #7 and #12

As a User Admin, I want to be able to suspend user accounts, so that I can suspend their user account if they leave the company

As a User Admin, I want to be able to suspend user profiles, so that I can suspend their user profile if they leave the company

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as User Admin 3. User Admin Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 7.1 | Suspend user | Test if the program can suspend an active user |
| 7.2 | Unsuspend user | Test if the program can un-suspend an inactive user |

|  |  |
| --- | --- |
| **Test Case 7.1 – Suspend user** | |
| **Steps** | 1. Click ‘Suspend’ button 2. Select an active user 3. Click ‘Suspend/Un-suspend Account’ button |
| **Expected Results** | The program will show that the user account is suspended |

|  |  |
| --- | --- |
| **Test Case 7.2 – Un-suspend user** | |
| **Steps** | 1. Click ‘Suspend’ button 2. Select an inactive user 3. Click ‘Suspend/Un-suspend Account’ button |
| **Expected Results** | The program will show that the user account is un-suspended |

|  |  |
| --- | --- |
| **Test Case 7.3 – Suspend/Un-suspend without selecting user** | |
| **Steps** | 1. Click ‘Suspend’ button 2. Click ‘Suspend/Un-suspend Account’ button |
| **Expected Results** | The program will tell the user that no user is selected |

### User Story #27

As a Customer, I want to Insert Table Number, so that I can start placing order under my table number for identification

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Starting page is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 27.1 | Insert table number | Test if the program will allow the customer to start placing orders |
| 27.2 | Leave table field empty | Test if the program will warn the user that the table number field is empty |

|  |  |
| --- | --- |
| **Test Case 27.1 – Insert table number** | |
| **Steps** | 1. Click the ‘Customer’ button 2. Enter the table number in the table number field 3. Click ‘View Menu’ button |
| **Expected Results** | The program will open the customer home page |

|  |  |
| --- | --- |
| **Test Case 27.1 – Insert table number** | |
| **Steps** | 1. Click ‘Customer’ button 2. Leave the table number field empty 3. Click ‘View Menu’ button |
| **Expected Results** | The program will warn the user that no table number is entered. |

## Sprint 3

### User Story #28

As a Customer, I want to browse the menu, so that I can choose what menu items I want

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 28.1 | View Menu | Test if the program will allow the customer to view the menu |

|  |  |
| --- | --- |
| **Test Case 28.1 – View Menu** | |
| **Steps** | 1. Click ‘View Menu’ button |
| **Expected Results** | The program will display the menu |

### 

### User Story #29

As a Customer, I want to add menu items to cart, so that I can order menu items

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 29.1 | Add menu item to cart | Test if the program will allow the customer to add items to cart |
| 29.2 | Add menu item to cart with 0 quantity | Test if the program will tell the user that quantity cannot be 0 |
| 29.3 | Add menu item to cart without selecting an item | Test if the program will tell the user that no item is selected |
| 29.4 | Add menu item with blank quantity | Test if the program will tell the user that quantity is blank |

|  |  |
| --- | --- |
| **Test Case 29.1 – Add menu item to cart** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Select menu item 3. Change the quantity to desired amount 4. Click ‘Add To Cart’ button |
| **Expected Results** | The program will display a message telling the customer the item is added to cart |

|  |  |
| --- | --- |
| **Test Case 29.2 – Add menu item to cart with 0 quantity** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Select menu item 3. Change the quantity to 0 4. Click ‘Add To Cart’ button |
| **Expected Results** | The program will display a message telling the customer that the quantity cannot be 0 |

|  |  |
| --- | --- |
| **Test Case 29.3 – Add menu item to cart without selecting and item** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Click ‘Add To Cart’ button |
| **Expected Results** | The program will display a message telling the customer no menu item is selected |

|  |  |
| --- | --- |
| **Test Case 29.4 – Add menu item to cart with blank quantity** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Select menu item 3. Change the quantity to blank 4. Click ‘Add To Cart’ button |
| **Expected Results** | The program will display a message telling the customer that the quantity cannot be blank |

### User Story #30

As a Customer, I want to edit cart items, so that I can change the quantity of the items in my cart

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 30.1 | Edit cart item quantity | Test if the program will edit the cart item quantity |
| 30.2 | Edit cart item without selecting item | Test if the program will warn the user that no item is selected |
| 30.3 | Edit cart item with quantity of 0 | Test if the program will remove the cart item when quantity is 0 |

|  |  |
| --- | --- |
| **Test Case 30.1 – Edit cart item quantity** | |
| **Steps** | 1. Click ‘Edit Cart’ button 2. Select menu item 3. Type in new item quantity 4. Click ‘Update Cart’ button |
| **Expected Results** | The program will display a message telling the customer the item quantity is updated |

|  |  |
| --- | --- |
| **Test Case 30.2 – Edit cart item without selecting item** | |
| **Steps** | 1. Click ‘Edit Cart’ button 2. Click ‘Update Cart’ button |
| **Expected Results** | The program will display a message telling the customer that no item is selected |

|  |  |
| --- | --- |
| **Test Case 30.3 – Edit cart item quantity** | |
| **Steps** | 1. Click ‘Edit Cart’ button 2. Select menu item 3. Type in quantity as 0 4. Click ‘Update Cart’ button |
| **Expected Results** | The program will display a message telling the customer the item quantity is updated and the item is removed from the cart |

### User Story #31

As a Customer, I want to search different categories of menu items (main course, drinks etc), so that I can easily find the type of food I wish to order.

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 31.1 | Categorize menu items by pasta items | Test if the program will show only pasta items |
| 31.2 | Categorize menu items by pizza items | Test if the program will show only pizza items |
| 31.3 | Categorize menu items by baked rice items | Test if the program will show only baked rice items |
| 31.4 | Categorize menu items by drinks | Test if the program will show only drink items |

|  |  |
| --- | --- |
| **Test Case 31.1 – Categorize menu items by pasta items** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Click ‘Pasta’ button |
| **Expected Results** | The program will display only pasta items |

|  |  |
| --- | --- |
| **Test Case 31.2 – Categorize menu items by pizza items** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Click ‘Pizza’ button |
| **Expected Results** | The program will display only pizza items |

|  |  |
| --- | --- |
| **Test Case 31.3 – Categorize menu items by baked rice items** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Click ‘Baked Rice’ button |
| **Expected Results** | The program will display only baked rice items |

|  |  |
| --- | --- |
| **Test Case 31.4 – Categorize menu items by drinks** | |
| **Steps** | 1. Click ‘View Menu’ button 2. Click ‘Drinks’ button |
| **Expected Results** | The program will display only drink items |

### User Story #32

As a Customer, I want to view cart items, so that I can verify the items that I want to order

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 32.1 | View cart items | Test if the program will allow the customer to view the cart |

|  |  |
| --- | --- |
| **Test Case 32.1 – View cart items** | |
| **Steps** | 1. Click ‘View Cart’ button |
| **Expected Results** | The program will display a list of all items in the cart |

### User Story #33

As a Customer, I want to delete cart items, so that I can remove items from cart which I have incorrectly added/do not want anymore

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 33.1 | Delete cart items | Test if the program will allow the customer to delete cart items |
| 33.2 | Delete cart item without selecting item | Test if the program will warn the user that no item is selected |

|  |  |
| --- | --- |
| **Test Case 33.1 – Delete cart item** | |
| **Steps** | 1. Click ‘Edit Cart’ button 2. Select menu item 3. Click ‘Delete Item’ button |
| **Expected Results** | The program will display a message telling the customer the item quantity is updated |

|  |  |
| --- | --- |
| **Test Case 33.2 – Delete cart item without selecting item** | |
| **Steps** | 1. Click ‘Edit Cart’ button 2. Click ‘Delete Item’ button |
| **Expected Results** | The program will display a message telling the customer that no item is selected |

### User Story #425

As a Customer, I want to add/apply coupon at payment so that I pay a discounted price

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 425.1 | Add/Apply coupon at payment | Test if the program will allow the customer to add a coupon for a discount |
| 425.2 | Add/Apply invalid coupon at payment | Test if the program will tell the customer the coupon is invalid |

|  |  |
| --- | --- |
| **Test Case 425.1 – Add/Apply coupon at payment** | |
| **Steps** | 1. Click ‘Payment’ button 2. Enter coupon code 3. Click ‘Input Coupon’ button |
| **Expected Results** | The program will show the user a message that the coupon is applied |

|  |  |
| --- | --- |
| **Test Case 425.2 – Add/Apply coupon at payment** | |
| **Steps** | 1. Click ‘Payment’ button 2. Enter invalid coupon code 3. Click ‘Input Coupon’ button |
| **Expected Results** | The program will show the user a message that the coupon is invalid |

### User Story #34

As a Customer, I want to do e-payment, so that I can make payment from the table

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. Customer has entered Table Number 3. Customer Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 34.1 | Perform e-payment | Test if the program will allow the customer make e-payment |
| 34.2 | Perform e-payment with no phone number | Test if the program will tell the user there is no phone number entered |
| 34.3 | Perform e-payment with invalid phone number | Test if the program will tell the user the phone number is invalid |

|  |  |
| --- | --- |
| **Test Case 34.1 – Perform e-payment** | |
| **Steps** | 1. Click ‘Payment’ button 2. Enter phone number 3. Click ‘Make Payment’ button |
| **Expected Results** | The program will show the user a message that payment has been completed |

|  |  |
| --- | --- |
| **Test Case 34.2 – Perform e-payment with no phone number** | |
| **Steps** | 1. Click ‘Payment’ button 2. Click ‘Make Payment’ button |
| **Expected Results** | The program will show the user a message that there is not phone number entered |

|  |  |
| --- | --- |
| **Test Case 34.3 – Perform e-payment with invalid phone number** | |
| **Steps** | 1. Click ‘Payment’ button 2. Enter invalid phone number 3. Click ‘Make Payment’ button |
| **Expected Results** | The program will show the user a message that the phone number is invalid |

### User Story #37

As a Restaurant Owner, I want to generate daily average spending per visit, so that I can view customer spending patterns for the day for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 37.1 | Generate Daily Average Spending per visit | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 37.1 – Generate Daily Average Spending per visit** | |
| **Steps** | 1. Click the ‘Average Spend Per Visit’ radio button. 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the average spend for the day in the left column |

### User Story #38

As a Restaurant Owner, I want to generate weekly average spending per visit, so that I can view spending patterns for the week for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 38.1 | Generate Weekly Average Spending per visit | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 38.1 – Generate Weekly Average Spending per visit** | |
| **Steps** | 1. Click the ‘Average Spend Per Visit’ radio button. 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the average spend for the week in the middle column |

### 

### User Story #39

As a Restaurant Owner, I want to generate monthly average spending per visit, so that I can view spending patterns for the month for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 39.1 | Generate Monthly Average Spending per visit | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 39.1 – Generate Monthly Average Spending per visit** | |
| **Steps** | 1. Click the ‘Average Spend Per Visit’ radio button. 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the average spend for the month on the right column |

### User Story #40

As a Restaurant Owner, I want to generate daily frequency/patterns of visits, so that I can view customer visiting frequency/patterns for the day for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 40.1 | Generate Daily Frequency/Patterns of Visits | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 40.1 – Generate Daily Frequency/Patterns of Visits** | |
| **Steps** | 1. Click the ‘Frequency/Patterns of Visits’ radio button 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the frequency/patterns of visits for the day in the left column |

### 

### User Story #41

As a Restaurant Owner, I want to generate weekly frequency/patterns of visits, so that I can view customer visiting frequency/patterns for the week for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 41.1 | Generate Weekly Frequency/Patterns of Visits | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 41.1 – Generate Weekly Frequency/Patterns of Visits** | |
| **Steps** | 1. Click the ‘Frequency/Patterns of Visits’ radio button 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the frequency/patterns of visits for the week in the middle column |

### User Story #42

As a Restaurant Owner, I want to generate monthly frequency/patterns of visits, so that I can view customer visiting frequency/patterns for the month for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 42.1 | Generate Monthly Frequency/Patterns of Visits | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 42.1 – Generate Monthly Frequency/Patterns of Visits** | |
| **Steps** | 1. Click the ‘Frequency/Patterns of Visits’ radio button 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the frequency/patterns of visits for the month on the right column |

### User Story #43

As a Restaurant Owner, I want to generate daily dish/drink preference, so that I can view customer dish/drink preference for the day for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 43.1 | Generate Daily Dish/Drink Preference | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 43.1 – Generate Daily Dish/Drink Preference** | |
| **Steps** | 1. Click the ‘Dish/Drink Preference’ radio button 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the dish/drink preference for the day in the left column |

### User Story #44

As a Restaurant Owner, I want to generate weekly dish/drink preference, so that I can view customer dish/drink preference for the week for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 44.1 | Generate Weekly Dish/Drink Preference | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 44.1 – Generate Weekly Dish/Drink Preference** | |
| **Steps** | 1. Click the ‘Dish/Drink Preference’ radio button 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the dish/drink preference for the week in the middle column |

### 

### User Story #45

As a Restaurant Owner, I want to generate monthly dish/drink preference, so that I can view customer dish/drink preference for the month for marketing purposes

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Owner 3. Restaurant Owner Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 45.1 | Generate Monthly Dish/Drink Preference | Test if the program will generate the correct data |

|  |  |
| --- | --- |
| **Test Case 45.1 – Generate Monthly Dish/Drink Preference** | |
| **Steps** | 1. Click the ‘Dish/Drink Preference’ radio button 2. Click ‘Generate Data’ button |
| **Expected Results** | The program will display the dish/drink preference for the month on the right column |

## Backlog

### User Story #15

As a Restaurant Manager, I want to create menu items so that I can add new menu listings

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 15.1 | Add new menu listing | Test if the program will add a new menu item |
| 15.2 | Add duplicate menu listing | Test if the program will warn the user that the menu item already exists |
| 15.3 | Add new menu listing with empty fields | Test if the program will warn the user that fields are empty |
| 15.4 | Add new menu listing with price of 0 | Test if the program will warn the user that the price cannot be 0 |

|  |  |
| --- | --- |
| **Test Case 15.1 – Add new menu listing** | |
| **Steps** | 1. Click the ‘Create’ button 2. Click the ‘Menu Items’ button 3. Input food name and price 4. Select food category 5. Click the ‘Create Menu Item’ button |
| **Expected Results** | The program will display a message that a new item is added |

|  |  |
| --- | --- |
| **Test Case 15.2 – Add duplicate menu listing** | |
| **Steps** | 1. Click the ‘Create’ button 2. Click the ‘Menu Items’ button 3. Enter an existing menu item name and price 4. Click the ‘Create Menu Item’ button |
| **Expected Results** | The program will display a message that the menu item already exists |

|  |  |
| --- | --- |
| **Test Case 15.3 – Add new menu listing with empty fields** | |
| **Steps** | 1. Click the ‘Create’ button 2. Click the ‘Menu Items’ button 3. Click the ‘Create Menu Item’ button |
| **Expected Results** | The program will display a message that fields cannot be left empty |

|  |  |
| --- | --- |
| **Test Case 15.3 – Add new menu listing with price of 0** | |
| **Steps** | 1. Click the ‘Create’ button 2. Click the ‘Menu Items’ button 3. Enter menu name and price of 0 4. Click the ‘Create Menu Item’ button |
| **Expected Results** | The program will display a message that fields cannot be left empty |

### User Story #16

As a Restaurant Manager, I want to edit menu items so that I can keep the menu items/prices up to date

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 16.1 | Edit menu listing | Test if the program will edit menu item |
| 16.2 | Edit menu listing with empty fields | Test if the program will warn the user that fields are empty |
| 16.3 | Edit menu listing with price of 0 | Test if the program will warn the user that the price is 0 |

|  |  |
| --- | --- |
| **Test Case 16.1 – Edit menu listing** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the ‘Menu Items’ button 3. Select a menu item 4. Change the fields 5. Click the ‘Edit Menu Item’ button |
| **Expected Results** | The program will display a message that the menu listing is edited |

|  |  |
| --- | --- |
| **Test Case 16.2 – Edit menu listing with empty fields** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the ‘Menu Items’ button 3. Select a menu item 4. Delete the text in the fields 5. Click the ‘Edit Menu Item’ button |
| **Expected Results** | The program will display a message that fields cannot be left empty |

|  |  |
| --- | --- |
| **Test Case 16.3 – Edit menu listing with price of 0** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the ‘Menu Items’ button 3. Select a menu item 4. Input price as 0 5. Click the ‘Edit Menu Item’ button |
| **Expected Results** | The program will display a message that price cannot be 0 |

### User Story #17

As a Restaurant Manager, I want to search menu items so that I can look for specific menu items

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 17.1 | Search by menu item name | Test if the program will return correct search results |
| 17.2 | Search by menu item name with empty field | Test if the program will tell the user that the field is empty |

|  |  |
| --- | --- |
| **Test Case 17.1 – Search by menu item name** | |
| **Steps** | 1. Click the ‘Search’ button 2. Enter a menu item name in the search box 3. Click the ‘Search by Menu Item Name’ button |
| **Expected Results** | The program will display a table with the search results |

|  |  |
| --- | --- |
| **Test Case 17.2 – Search by menu item name with empty field** | |
| **Steps** | 1. Click the ‘Search’ button 2. Click the ‘Search by Menu Item Name’ button |
| **Expected Results** | The program will display a message telling the user fields cannot be left empty |

### User Story #18

As a Restaurant Manager, I want to view menu items so that I can view details of the menu item

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 18.1 | View all menu items | Test if the program will display all menu items |

|  |  |
| --- | --- |
| **Test Case 18.1 – View all menu items** | |
| **Steps** | 1. Click the ‘View’ button 2. Click the ‘View All Menu Items’ button |
| **Expected Results** | The program will display a table with all menu items |

### User Story #19

As a Restaurant Manager, I want to delete menu items so that I can remove menu items that are no longer available

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 19.1 | Delete menu item | Test if the program will delete menu items |
| 19.2 | Delete menu item still in orders | Test if the program will prevent the deletion of menu item still in orders |

|  |  |
| --- | --- |
| **Test Case 19.1 – Delete menu item** | |
| **Steps** | 1. Click the ‘Delete’ button 2. Click the ‘Menu Items’ button 3. Click menu item to be deleted 4. Click ‘Delete Menu Item’ button |
| **Expected Results** | The program will display a message that the menu item is deleted |

|  |  |
| --- | --- |
| **Test Case 19.2 – Delete menu item still in orders** | |
| **Steps** | 1. Click the ‘Delete’ button 2. Click the ‘Menu Items’ button 3. Click menu item still in existing orders 4. Click ‘Delete Menu Item’ button |
| **Expected Results** | The program will display a message that the item cannot be deleted |

### User Story #420

As a Restaurant Manager, I want to create coupons that customers can use for discounts

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 420.1 | Add new coupon | Test if the program will add a new coupon |
| 420.2 | Add duplicate coupon | Test if the program will warn the user that the coupon already exists |
| 420.3 | Add new coupon with empty fields | Test if the program will warn the user that fields are empty |

|  |  |
| --- | --- |
| **Test Case 420.1 – Add new coupon** | |
| **Steps** | 1. Click the ‘Create’ button 2. Click the ‘Coupon’ button 3. Input coupon name and discount 4. Click the ‘Create Coupon’ button |
| **Expected Results** | The program will display a message that a new coupon is added |

|  |  |
| --- | --- |
| **Test Case 420.2 – Add duplicate coupon** | |
| **Steps** | 1. Click the ‘Create’ button 2. Click the ‘Coupon’ button 3. Enter an existing coupon name and discount 4. Click the ‘Create Coupon’ button |
| **Expected Results** | The program will display a message that the coupon already exists |

|  |  |
| --- | --- |
| **Test Case 420.3 – Add new coupon with empty fields** | |
| **Steps** | 1. Click the ‘Create’ button 2. Click the ‘Coupon’ button 3. Click the ‘Create Coupon’ button |
| **Expected Results** | The program will display a message that fields cannot be left empty |

### User Story #421

As a Restaurant Manager, I want to be able to edit the coupons so that coupons are up to date with latest discounts

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 421.1 | Edit coupon | Test if the program will edit coupon |
| 421.2 | Edit coupon with empty fields | Test if the program will warn the user that fields are empty |
| 421.3 | Edit coupon with discount value of > 1 | Test if the program will warn the user that the discount value cannot be more than 1 |
| 421.4 | Edit coupon with discount value of 0 | Test if the program will warn the user that the discount value cannot be 0 |

|  |  |
| --- | --- |
| **Test Case 421.1 – Edit coupon** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the ‘Coupon’ button 3. Select a coupon 4. Change the fields 5. Click the ‘Edit Coupon’ button |
| **Expected Results** | The program will display a message that the coupon is edited |

|  |  |
| --- | --- |
| **Test Case 421.2 – Edit coupon with empty fields** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the ‘Coupon’ button 3. Select a coupon 4. Delete the text in the fields 5. Click the ‘Edit Coupon’ button |
| **Expected Results** | The program will display a message that fields cannot be left empty |

|  |  |
| --- | --- |
| **Test Case 421.3 – Edit coupon with discount value of > 1** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the ‘Coupon’ button 3. Select a coupon 4. Set Discount Value as 1.1 5. Click the ‘Edit Coupon’ button |
| **Expected Results** | The program will display a message that discount value cannot be more than 1 |

|  |  |
| --- | --- |
| **Test Case 421.3 – Edit coupon with discount value of 0** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the ‘Coupon’ button 3. Select a coupon 4. Set Discount Value as 0 5. Click the ‘Edit Coupon’ button |
| **Expected Results** | The program will display a message that discount value cannot be 0 |

### User Story #422

As a Restaurant Manager, I want to be able to view coupons so that I can view existing coupon discounts

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 422.1 | View all coupons | Test if the program will display all coupons |

|  |  |
| --- | --- |
| **Test Case 422.1 – Search by coupon name** | |
| **Steps** | 1. Click the ‘View’ button 2. Click the ‘View All Coupons’ button |
| **Expected Results** | The program will display a table with all coupons |

### User Story #423

As a Restaurant Manager, I want to be able to delete coupons that are no longer available

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 423.1 | Delete coupon | Test if the program will delete coupons |

|  |  |
| --- | --- |
| **Test Case 423.1 – Search by coupon name** | |
| **Steps** | 1. Click the ‘Delete’ button 2. Click the ‘Coupons’ button 3. Click coupon to be deleted 4. Click ‘Delete Coupon’ button |
| **Expected Results** | The program will display a message that the coupon is deleted |

### User Story #424

As a Restaurant Manager, I want to be able to search coupons so that I look for specific coupons

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Manager 3. Restaurant Manager Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 424.1 | Search by coupon name | Test if the program will return correct search results |
| 424.2 | Search by coupon name with empty field | Test if the program will tell the user that the field is empty |

|  |  |
| --- | --- |
| **Test Case 424.1 – Search by coupon name** | |
| **Steps** | 1. Click the ‘Search’ button 2. Enter a coupon name in the search box 3. Click the ‘Search by Coupon Name’ button |
| **Expected Results** | The program will display a table with the search results |

|  |  |
| --- | --- |
| **Test Case 424.2 – Search by coupon name with empty field** | |
| **Steps** | 1. Click the ‘Search’ button 2. Click the ‘Search by Coupon Name’ button |
| **Expected Results** | The program will display a message telling the user fields cannot be left empty |

### User Story #22

As a Restaurant Staff, I want to edit order status so that I can change the order status to fulfilled/delivered

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Staff 3. Restaurant Staff Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 22.1 | Fulfill order | Test if the program will fulfill orders |
| 22.2 | Fulfill already fulfilled orders | Test if the program will tell the user that the order is already fulfilled |

|  |  |
| --- | --- |
| **Test Case 22.1 – Fulfill order** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the order to fulfill 3. Click the ‘Fulfill Order’ button |
| **Expected Results** | The program will display a message that the order is fulfilled |

|  |  |
| --- | --- |
| **Test Case 22.2 – Fulfill already fulfilled orders** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click an already fulfilled order 3. Click the ‘Fulfill Order’ button |
| **Expected Results** | The program will display a message that the order is already fulfilled |

### User Story #23

As a Restaurant Staff, I want to edit orders so that I can assist customers who wish to edit their orders after it is sent to the kitchen

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Staff 3. Restaurant Staff Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 23.1 | Edit order | Test if the program will edit orders |
| 23.2 | Edit order with quantity of 0 | Test if the program will tell the user that the quantity cannot be 0 |

|  |  |
| --- | --- |
| **Test Case 23.1 – Edit order** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the order to edit 3. Change the quantity to desired amount 4. Click the ‘Edit Order’ button |
| **Expected Results** | The program will display a message that the order is edited |

|  |  |
| --- | --- |
| **Test Case 23.2 – Edit order with quantity of 0** | |
| **Steps** | 1. Click the ‘Edit’ button 2. Click the order to edit 3. Change the quantity to 0 4. Click the ‘Edit Order’ button |
| **Expected Results** | The program will display a message that the quantity cannot be 0 |

### User Story #24

As a Restaurant Staff, I want to search orders by table number, so that I am able to access the customer’s orders.

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Staff 3. Restaurant Staff Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 24.1 | Search order by table number | Test if the program will show orders for a table |
| 24.2 | Search order with empty table number | Test if the program will tell the user that the table number field cannot be empty |

|  |  |
| --- | --- |
| **Test Case 24.1 – Search order by table number** | |
| **Steps** | 1. Click the ‘Search’ button 2. Enter a table number to search 3. Click the ‘Search Order by Table Number’ button |
| **Expected Results** | The program will display a table of orders for the table number |

|  |  |
| --- | --- |
| **Test Case 24.2 – Search order with empty table number** | |
| **Steps** | 1. Click the ‘Search’ button 2. Click the ‘Search Order by Table Number’ button |
| **Expected Results** | The program will display a message that table number is invalid |

### User Story #25

As a Restaurant Staff, I want to view orders, so that I can view order details

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Staff 3. Restaurant Staff Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 25.1 | View all orders | Test if the program will display all orders |

|  |  |
| --- | --- |
| **Test Case 25.1 – Search by coupon name** | |
| **Steps** | 1. Click the ‘View’ button |
| **Expected Results** | The program will display a table of all orders |

### User Story #26

As a Restaurant Staff, I want to delete orders, so that I can remove invalid/cancelled orders

|  |
| --- |
| **Pre-requisites** |
| 1. Program is running 2. User is logged in as Restaurant Staff 3. Restaurant Staff Homepage is displayed |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 26.1 | Delete orders | Test if the program will delete orders |
| 26.2 | Delete order without selecting order | Test if the program will tell the user that no order is selected |

|  |  |
| --- | --- |
| **Test Case 26.1 – Search order by table number** | |
| **Steps** | 1. Click the ‘Delete’ button 2. Select order to be deleted 3. Click the ‘Delete Order’ button |
| **Expected Results** | The program will display a message that the order is deleted |

|  |  |
| --- | --- |
| **Test Case 26.2 – Search order with empty table number** | |
| **Steps** | 1. Click the ‘Delete’ button 2. Click the ‘Delete Order’ button |
| **Expected Results** | The program will display a message that no order is selected |

# Unit Testing for Entities

## Staff class

The Staff Class is responsible for communicating with the database and retrieving the relevant information from the database whenever the program requires it.

Junit Test file name : StaffTest.java

**List of Unit Test cases for Staff class (Updated 9/5/22)**

|  |  |  |  |
| --- | --- | --- | --- |
| Test ID | Test Description | Tested Function | Expected Result |
| Staff\_1 | Login with valid User Admin credentials | login() | True |
| Staff\_2 | Login with invalid User Admin credentials | login () | False |
| Staff\_3 | Login with valid User Admin but wrong profile | login () | False |
|  |  |  |  |
| Staff\_4 | Login with valid Manager credentials | login () | True |
| Staff\_5 | Login with invalid Manager credentials | login () | False |
| Staff\_6 | Login with valid Manager but wrong profile | login () | False |
|  |  |  |  |
| Staff\_7 | Login with valid Staff credentials | login () | True |
| Staff\_8 | Login with invalid Staff credentials | login () | False |
| Staff\_9 | Login with valid Staff but wrong profile | login () | False |
|  |  |  |  |
| Staff\_10 | Login with valid Owner credentials | login () | True |
| Staff\_11 | Login with invalid Owner credentials | login () | False |
| Staff\_12 | Login with valid Owner but wrong profile | login () | False |

## User\_Admin class

The User\_Admin class is responsible for User Admin functions such as creating and editing accounts and retrieving user info for viewing.

Junit Test file name : User\_AdminTest.java

**List of Unit Test cases for User\_Admin class (Updated 9/5/22)**

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Test Case | Tested Function | Expected Result |
| User\_Admin\_1 | Create new account | createAccount() | No exceptions |
| User\_Admin\_2 | Edit an account | editAccount() | No exceptions |
| User\_Admin\_3 | Get information from database | getUserInfoFromDB() | No exceptions |
| User\_Admin\_4 | Search for accounts | searchAccount() | No exceptions |
| User\_Admin\_5 | Suspend an account | suspendAccount() | No exceptions |

## Customer class

9/5/22 – For now all functions in customer class are just calling the Cart class functions. Therefore, testing the Cart class is sufficient.

## Cart class

The Cart class is responsible for Cart functions such as adding and removing items from the customer’s cart.

Junit Test file name : CartTest.java

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Test Case | Tested Function | Expected Result |
| Cart\_1 | Get transaction id from cart object | getTransaction\_idTest() | No exceptions |
| Cart\_2 | Generate transaction id for table number | setTransaction\_idTest() | No exceptions |
| Cart\_3 | View items in cart | viewCart() | No exceptions |
| Cart\_4 | Add item to cart | addToCart() | No exceptions |
| Cart\_5 | Edit quantity in cart | editCart() | No exceptions |
| Cart\_6 | Delete item from cart | deleteFromCart() | No exceptions |
| Cart\_7 | Get total price of cart | getTotalPrice() | Float Value |
| Cart\_8 | Apply discount on price | applyDiscount() | True |
| Cart\_9 | Make payment | makePayment() | True |

## Restaurant\_Owner class

The Restaurant Owner class is responsible for functions such as generating daily, weekly and monthly reports.

Junit Test file name : Restaurant\_OwnerTest.java

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Test Case | Tested Function | Expected Result |
| Restaurant\_Owner\_1 | Get daily spending report | dailySpending() | String |
| Restaurant\_Owner\_2 | Get weekly spending report | weeklySpending() | String |
| Restaurant\_Owner\_3 | Get monthly spending report | monthlySpending() | String |
| Restaurant\_Owner\_4 | Get daily frequency report | dailyFrequency() | String |
| Restaurant\_Owner\_5 | Get weekly frequency report | weeklyFrequency() | String |
| Restaurant\_Owner\_6 | Get monthly frequency report | monthlyFrequency() | String |
| Restaurant\_Owner\_7 | Get daily preference report | dailyPreference() | String |
| Restaurant\_Owner\_8 | Get weekly preference report | weeklyPreference() | String |
| Restaurant\_Owner\_9 | Get monthly preference report | monthlyPreference() | String |
| Restaurant\_Owner\_10 | Get all reports in a 2D array | getReport() | String [] [] |

## Menu\_Item class

The Menu Item class is responsible for displaying the menu to the customer

Junit Test file name : Menu\_ItemsTest.java

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Test Case | Tested Function | Expected Result |
| Menu\_Item\_1 | Get all menu items | viewMenu() | String [] [] |
| Menu\_Item\_2 | Get pasta menu items | viewMenu() | String [] [] |
| Menu\_Item\_3 | Get pizza menu items | viewMenu() | String [] [] |
| Menu\_Item\_4 | Get baked rice menu items | viewMenu() | String [] [] |
| Menu\_Item\_5 | Get drink menu items | viewMenu() | String [] [] |

## Restaurant\_Manager class

The restaurant manager class is responsible for functions such as creating, editing, searching and deleting menu items/coupons

Junit Test file name : Restaurant\_ManagerTest.java

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Test Case | Tested Function | Expected Result |
| Restaurant\_Manager\_1 | Get transaction id from cart object | checkMenuItemExistence() | True |
| Restaurant\_Manager\_2 | Generate transaction id for table number | createMenuItems() | True |
| Restaurant\_Manager\_3 | View items in cart | checkCouponExistence() | True |
| Restaurant\_Manager\_4 | Add item to cart | createCoupon() | True |
| Restaurant\_Manager\_5 | Edit quantity in cart | edditMenuItems() | True |
| Restaurant\_Manager\_6 | Delete item from cart | searchMenuItems() | True |
| Restaurant\_Manager\_7 | Get total price of cart | viewMenuItems() | True |
| Restaurant\_Manager\_8 | Apply discount on price | deleteMenuItems() | True |

|  |  |  |
| --- | --- | --- |
| **ID** | **Test Case** | **Purpose** |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |

|  |  |
| --- | --- |
| **Test Case** | |
| **Steps** |  |
| **Expected Results** |  |

|  |  |  |
| --- | --- | --- |
| ID | Test Case | Tested Function |
|  |  |  |

|  |  |
| --- | --- |
| **Test Case** | |
| **Expected Results** |  |
| **Actual Results** |  |
| **Pass/Fail** |  |