**Name : Srikanth Bandi**

**M3\_RestaurantBillingSystem\_Utility**

[](https://github.com/srikanthbandi949/M3_RestaurantBillingSystem_Utility/blob/main/6_Images_And_Videos/billingsystem.jpg)

**SDLC Activity Based Learning**

**Folder Structure**

| **Folder** | **Description** |
| --- | --- |
| 1\_Requiremets | Documents detailing Requirements(High level and Low level requirements) |
| 2\_Architecture | Structural and Behavioural design(Flow charts , usecase diagrams) |
| 3\_Implementation | All code and code files |
| 4\_TestPlanAndOutput | Documents with test plan and output |
| 5\_Report | Whole Report of the project |
| 6\_Images\_And\_Videos | Project necessary images and videos will be here |

**Contriutors List and Summary**

| **PS NO.** | **Name** | **Features** | **Number of testcases implemented** | **No\_of testcases passed** | **No\_of testcases failed** |
| --- | --- | --- | --- | --- | --- |
| 99007538 | Srikanth Bandi | Entire Project: Requirements, Architecture, Implementation, Test plan,Report | 2 | 2 | 0 |

**References**

* Javatpopint
* W3 Schools, Geeks for geeks

# Certificates

### 1.Sololearn

### 2.Hackerrank

## Sololearn certificate

* Sololearn java completion certification

## Hackerrank

* Hackerrank java goldmedal screenshot

# Requirements

## Description

* This Project is titled as **RestaurantBillingSystem** and developed in using java language.
* RestaurantBillingSystem is an utility which is used in Restaurants, Hotels etc.., From this system owner can add the food item s and drinks whatever they have in the restaurant. and also they can update the prices of food items at anypoint of time by selecting update food items and also owner has a login page, only owner can login with his credientials and other login page is for customers, where they don't have to enter any credientials just by clicking on the customers case they directly go into customers page. Where they can see different options to display the menu , food order , food update which is used to update the previously ordered food , myorder - where they can see what are the food items they oredr along with the price of each order, bill - where it shows the bill along with the gst and also shows biil individually for each item they order. At the end of the options both owner and customer can see the exit case which is for when ever they want to exit from the page they can simply select that case where iot can simply taken into the loginpage again by exiting the previous page.

# Requirements

# High Level Requirements

| **S.No** | **Description** |
| --- | --- |
| HR-1 | Login -> It is to login into that system |
| HR-2 | Owner -> Owner Portal |
| HR-3 | Customer -> Customer Portal |
| HR-4 | FoodOrder -> Food order to eat |
| HR-5 | Bill -> ill for Food items |

# Low Level Requirements

| **S.No** | **Description** |
| --- | --- |
| LR-1 | Login consists of two parts : 1. Owner, 2. Customer . To logi to owner portal we need to type username and password |
| LR-2 | In Owner portal we have diffent options like: 1. Add Fooditems, 2. Update FoodItems, 3. Remove FoodItems, 4. Display Menu, 5. Exit |
| LR-3 | In Customer Portal we need different requirements like : 1. Display Menu, 2. Order Food Items, 3. Change Order, 4. Cancel Order, 5. Display Order, 6. Pay Bill, 7. Exit |
| LR-4 | It has the option to update food and delete food , and also change order and update order by the customer |
| LR-5 | In the Bill we need different requirements as : 1. Food Bill, 2. GST TAX, 3. Total Amount |

# SWOT Aalysis

|  |  |
| --- | --- |
| Strength | Easy to use and Easy to Choose the available options and you can update and change your fooditems and also update the price of fooditems any time byb the owner |
| Weakness | Need GUI to look more attactive |
| Opportunity | Quickly understand makes more demand in future , and also GUI also implemented in future, Easily used by anyone so common man can also be used easily , so oppurtinities will increase |
| Threat | GUI applications are available so compitition is high and hard , Lots of applicatins are developed |

# 4W's & 1H

#### Who

* everyone now-a-days wants to use the portals to make thier order quick and saves time by not waiting until the waiter comes and asks for our order
* all age group persons can simply use this interface

#### What

* It is a Restaurant food ordering and bill generating system with this we can simply order our food without any one taking and writing our order , and also easily update our food list through the owner portl with ccedientials and generate bill along with GST
* Makes work easier and also simple and saves time

#### When

* When ever we goto restaurant and also hotels for oredering food and eating ths will help and also it generates bill along with GST, When ever we want to change our ordered items we can change through this by simply clicking on change order

#### Where

* We can use this in Food courts, Restaurants, Hotels for oredering and generating bill.

#### How

* When ever we open the application it asks the user to choose whether he/she is Owner/Customer
* After selecting , if it is Owner then it asks Login credientials, if customer then it directly takes to other page and showss some options , a d he/she wants to choose accourding to them what he want to do
* At the end it generates bill along with the GST and display overall amount and shows fooditem and GST amount then Shows Overall amount

# TestPlan

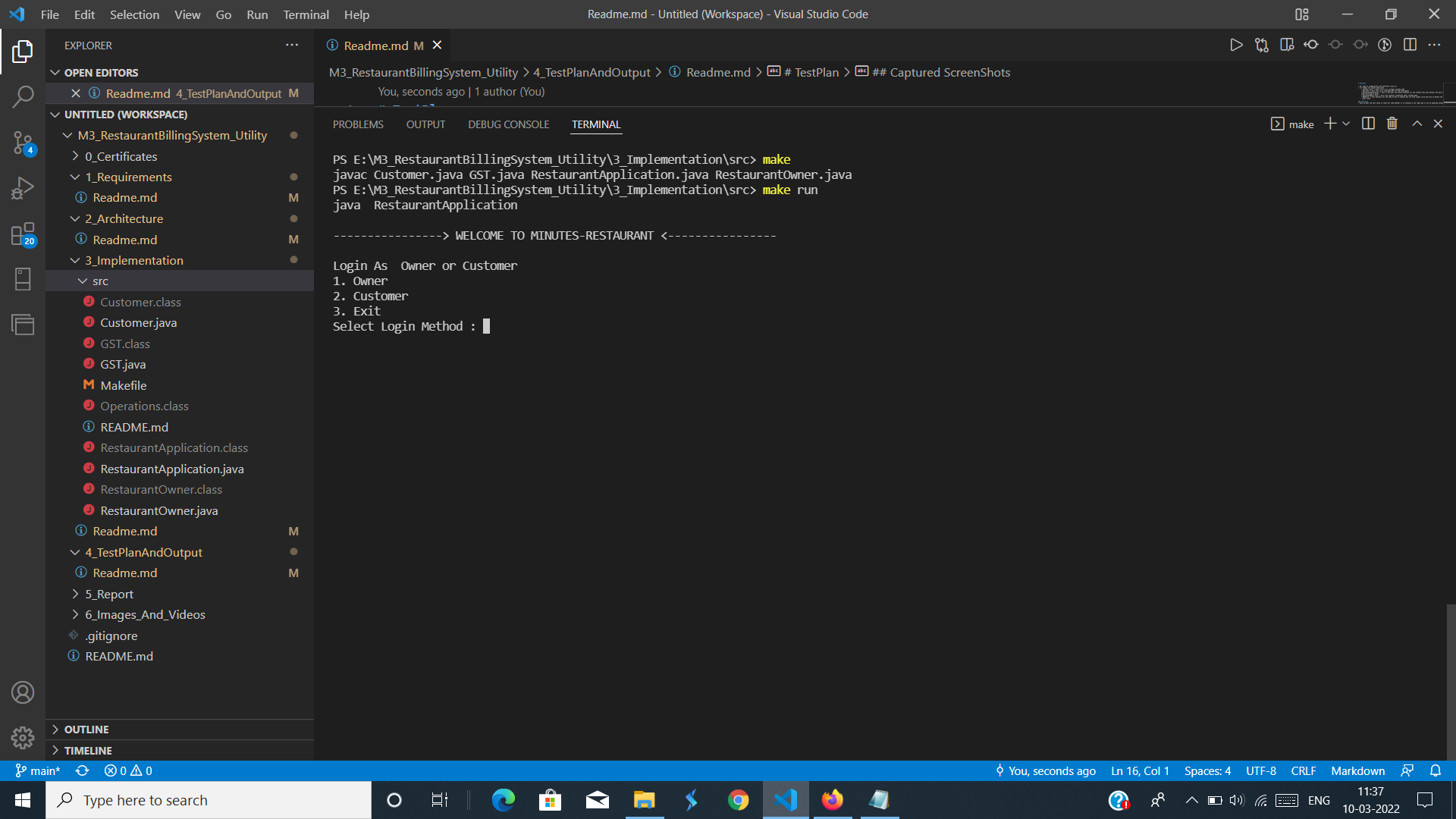
* All code is categeroised into different files as:
* src folder for source code files
  + Customer.java -> this file is for customer related code
  + GST.java -> this file is to calculate GST for the food amount
  + RestaurantApplication -> This is the main file which contains all the linked files and contains the main code and how it starts and how it works
  + RestaurantOwner.java -> this file contains restaurant owner related data
  + Makefile -> this contains all the code to how to compile and run with single click and also to delete the unnecessary files as .class files

### Testcases

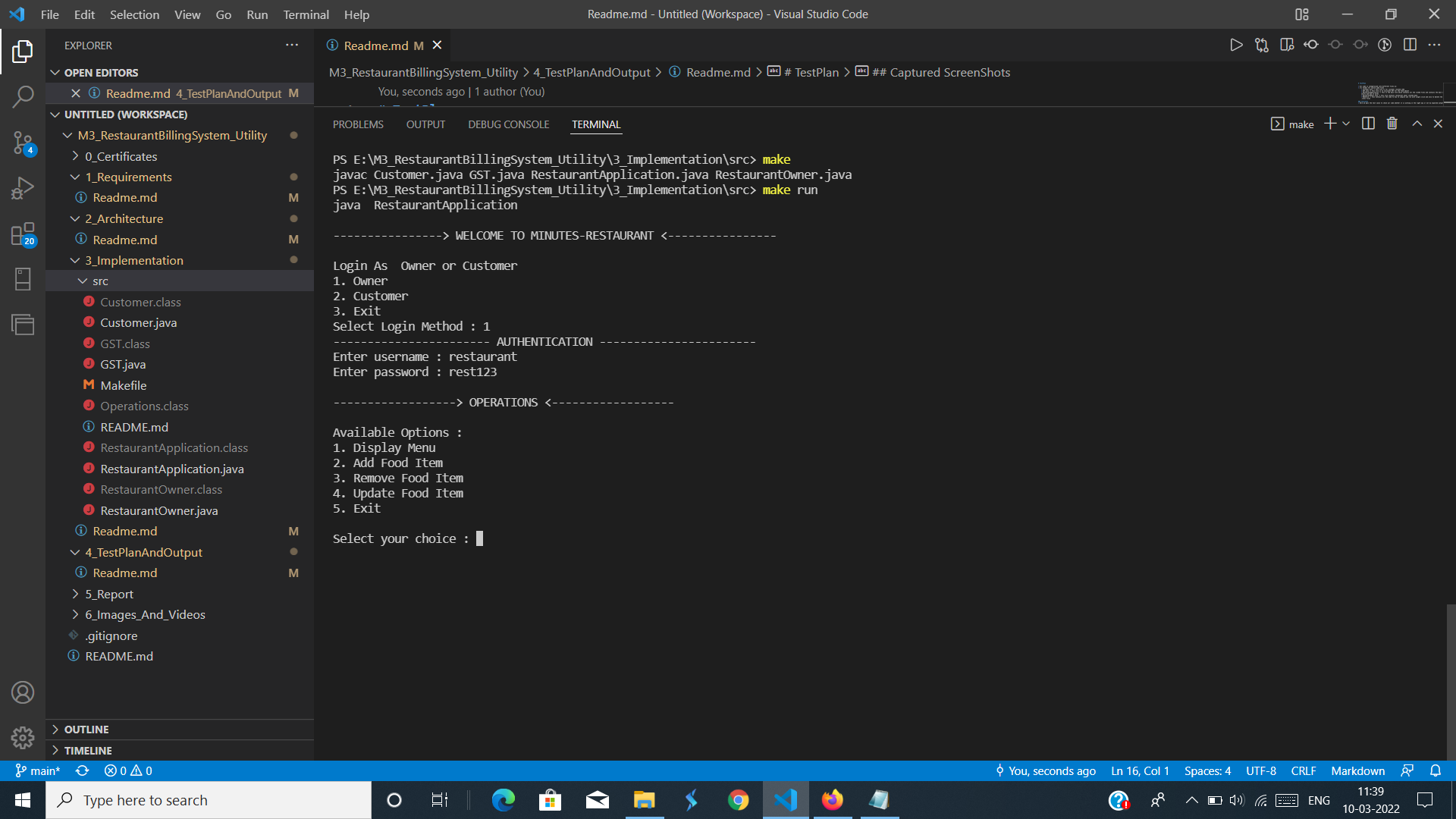
* Here we pass the test cases to check our code whether it is working in the right way or not by expected outputs

## Captured ScreenShots

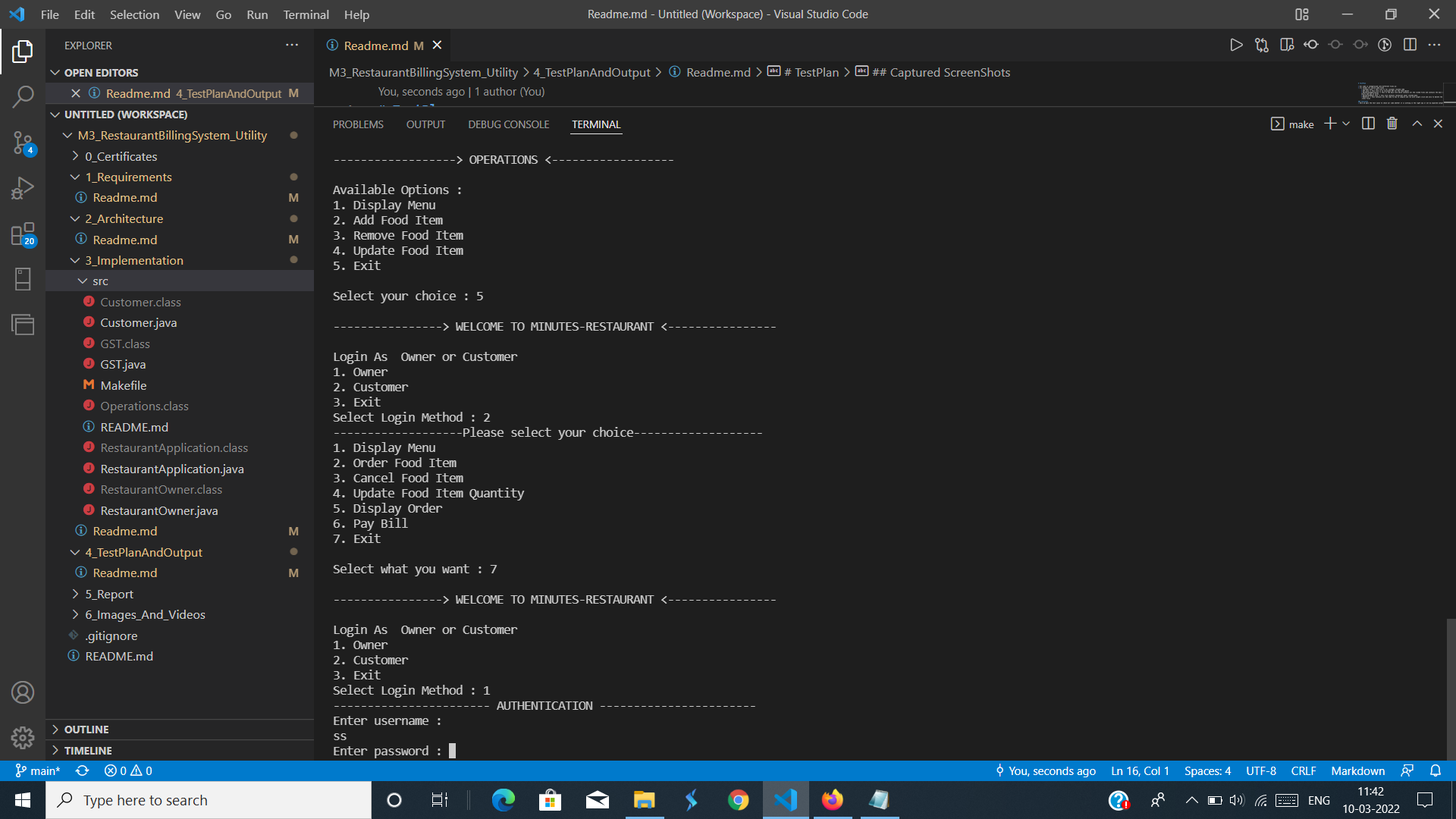
### OpenPage :

[](https://github.com/srikanthbandi949/M3_RestaurantBillingSystem_Utility/blob/main/4_TestPlanAndOutput/openpage.png)

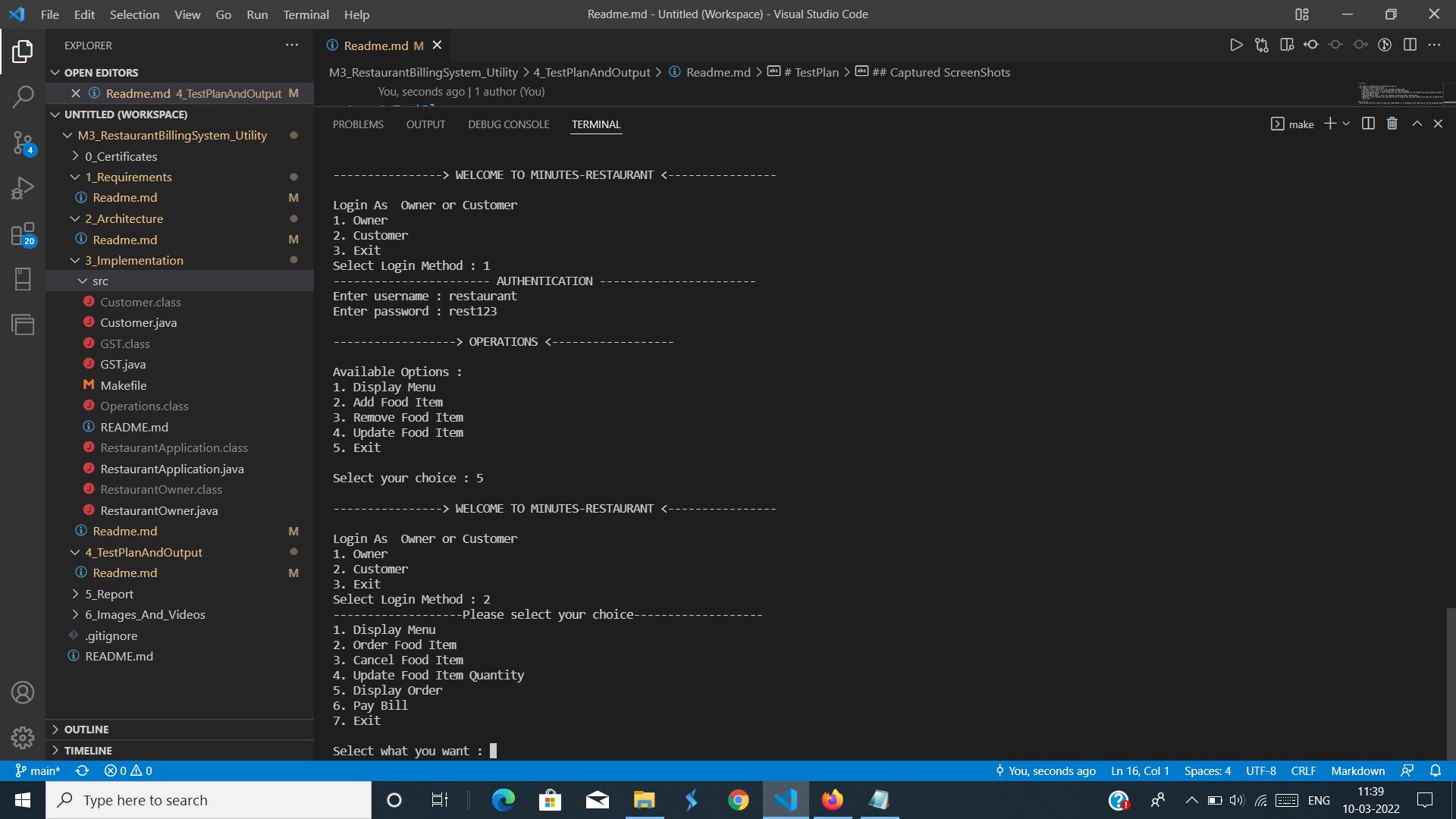
### Owner :

[](https://github.com/srikanthbandi949/M3_RestaurantBillingSystem_Utility/blob/main/4_TestPlanAndOutput/owner.png)

### Owner Credientials :

[](https://github.com/srikanthbandi949/M3_RestaurantBillingSystem_Utility/blob/main/4_TestPlanAndOutput/OwnerCredentials.png)

### Food :

[](https://github.com/srikanthbandi949/M3_RestaurantBillingSystem_Utility/blob/main/4_TestPlanAndOutput/Food.png)

# Table

* ID, Description of the test cases , input values , Expected Output, Actual output

| **ID** | **Description** | **input values** | **Expected output** | **Actual Output** |
| --- | --- | --- | --- | --- |
| 1 | Passing Owner and entering credientials | username: restaurant, password: rest123 | open owner portal | opens owner portal |
| 2 | entering 2 which is customer | 2 | open customer portal | opened customer portal |

**Images\_And\_Videos**

* It contains all the images and videos which are used in the project