OOPD Project – Food Delivery System

Project Title : Food Delivery Management System

Prepared By : Simran, Walmikee Dhopte, Parikshit Sarode, Samiksha Garg

Problem statement

Problems that are faced by the users while using the current Food Delivery system.

The current system of the delivery doesn’t allow users to login from different places.

1. As it is not available so every delivery calls up to find out if the order is available or not. - Too many calls to answer

2. Every time a delivery calls for an order you have to check the racks for the availability of the restaurants. - Too much of physical work.

3. Either the delivery has to call or come to the Food Delivery to reserve the order, but cannot reserve online over the internet.

4. Amounts have to be calculated manually.

5. Deliveries have no idea to track their location.

Project Scope

Project Title : Food Delivery Management System

Prepared By : Simran, Walmikee Dhopte, Parikshit Sarode, Samiksha Garg

Project Justification:

This project is basically updating the manual Food Delivery system into a terminal-based application so that the users can know the details of their account, availability of orders etc., all seamlessly.

Project Characteristics and Requirements:

1. Secured database (food.db).
2. User tracking
3. Track feature
4. Calculating amounts and coupons
5. Checking the authenticity of user.
6. Deliveries will get an update before the due date in order to avoid amounts

Project Management Deliverables: Project plan, making report, developing structure, .whl file creation, statement, Project profiling, UML, DOXYGEN, readme etc.,

Product-related deliverables: Profiling, UML, reports, design documents, working software code for every possible input scenario, test plan, project limitations etc.

Project Success Criteria: Our main goal is to complete this project within allotted dead line. It is necessary to develop a method for capturing the benefits while the Food Delivery management system is being developed, tested, and after it is submitted.

Statement of Work

1. Scope of Work

The terminal biased Food Delivery Management System should manage different types of Food Delivery tasks such as Orders, Customers, Restaurants, Tracking, Rating, Wishlist, Cart and any other resources which the management feels in the future could form a resource. A resource can be categorized to facilitate managing the resources. Deliveries and Delivery Staff can look location from the Food Delivery system. If any order did not reach user in time, we can have a feature for tracing or request cancelling. We also have feature for wishlist and rating and different options to pay.

2. Users

The client for this project is the Food Delivery users who uses this program, this could be both restaurants or customers.

3. Period of Performance

The duration of the project is 1 month. All the tasks are distributed among team members to accomplish the goal of completing the project. All concepts were implemented in the program

- Class.
- Object.
- Method.
- Inheritance.
- Polymorphism.
- Data Abstraction.
- Encapsulation.

4. Deliverables Schedule

Deliverable	**Date Due**
Written report part I: Statement of work and Project Plan	Week 1
Written report part II: Project scope & problem statement/ requirements specification + coding	Week 2
Written report part III: Problem analysis and research report + coding	Week 2
Written report part IV: Prototype implementation or practical experimental work + coding	Week 3
Written report part V: Review of results and statement of findings (including test reports) + coding	Week 4
Written report part VI: UMLs, profiling, reports + coding	Ongoing assessment at online meetings

5. Applicable Standards:

The project “Food Delivery Management System” follows various programming standards and makes uses of all OOPD concepts taught in class. It also some error handling mechanism, in cases user inputs wrong input

6. Acceptance Criteria:

The acceptance criteria for the users are

- The terminal site has to communicate with the database (food.db).
- Users can easily search the order with in less time
- Maintaining and Authentication of the account details of the user.
- Users can track the order, view the order through online etc.

7. Additional Requirements:

Qualified team members are very important for project. Team members have to have a basic knowledge in the following areas:

- A sound knowledge in programming languages such as Python, Java and a knowledge of database (food.db) such as MySQLITE3, SQLITE3 server or Access.

Software Design Document

1. Introduction

Software Design Document gives us the detailed description about the flow of the project.

1.1 Purpose

Software Design Document the modules, interfaces and the interaction between modules and interfaces. It also helps us in giving information about how the system will look the appearance of user interfaces (terminal) and the information view of the system. This really helps in coding too. It provides a direction about how basic control and data structures will be organized.

The UML (here we made Class Diagram) and Documentation is made before programming starts. It describes how the software will be structured, and what functionality will be included. This document forms the basis for all future design and coding.

1.2 Scope

The Food Delivery Management System is being developed for the college as final OOPD project. The important modules that are going to implement in the proposed system.

- Calculating the amounts.
- Reservation of orders.
- Tracking order depends upon i) Order name ii) Restaurant.
- Log in, depends upon user Id.
- Administrator has all the privileges to add, modify and delete the restaurants.
- It has amount calculation mechanism for the delivery.
- There is also feature of minimum order of 100.
- Coupons once used can’t be used again
- One can modify cart as well

1.3 Definitions and Abbreviations

The definitions that are constantly used in the document include the following.
.whl file – making wheel file with proper directory structure for others to import

Unified modelling language - The Unified Modelling Language - UML

2.0 System Overview

The Food Delivery Management System is a terminal-based application which is able to manage different types of Food Delivery resources such as Orders, Customers, Restaurants, Tracking, Rating etc.

3. System Architecture: The Three Tier Architecture:

To develop software for Food Delivery management system the architecture that we are implementing is Three Tier Architecture. The User Interface Layer(terminal), Application Logic Layer and Data Storage Layer together called as Three Tier Architecture.

3.1 User Interface Layer

The User Interface Layer will consist of everything shown directly to the user. This Layer will communicate only with the Application Logic Layer. This Layer will also be responsible for initial validation of any user inputted data. Any validation that requires communication to the Data Storage Layer will however be completed at the Logic Layer.

3.2 Application Logic Layer

The Business logic for our project is in Python. It will be responsible for providing a secure connection to the Data Storage Layer and formatting information received from the SQLITE3 Server for presentation to the User Interface Layer.

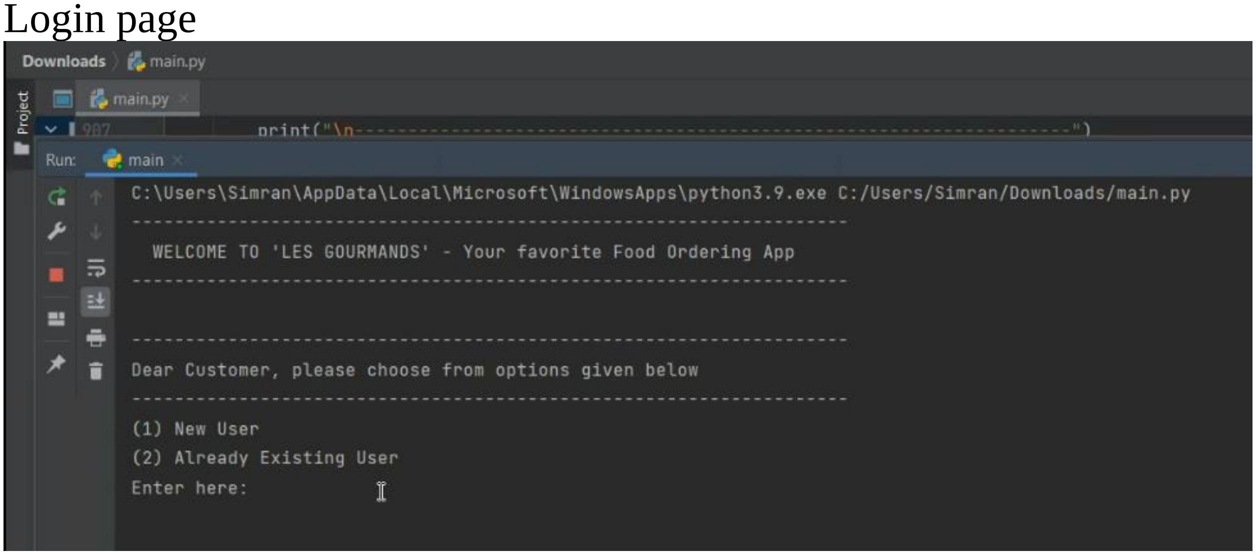
3.3 Data Storage Layer

The Data Storage Layer will record all information required by the Logic and User Interface Layer. All data stored in the database (food.db) safely. Within the Data Storage Layer a collection of SQLITE3 Queries will provide access to the data in a meaningful way.

4.0 Architectural Design

Some screen shots of the Food Delivery system

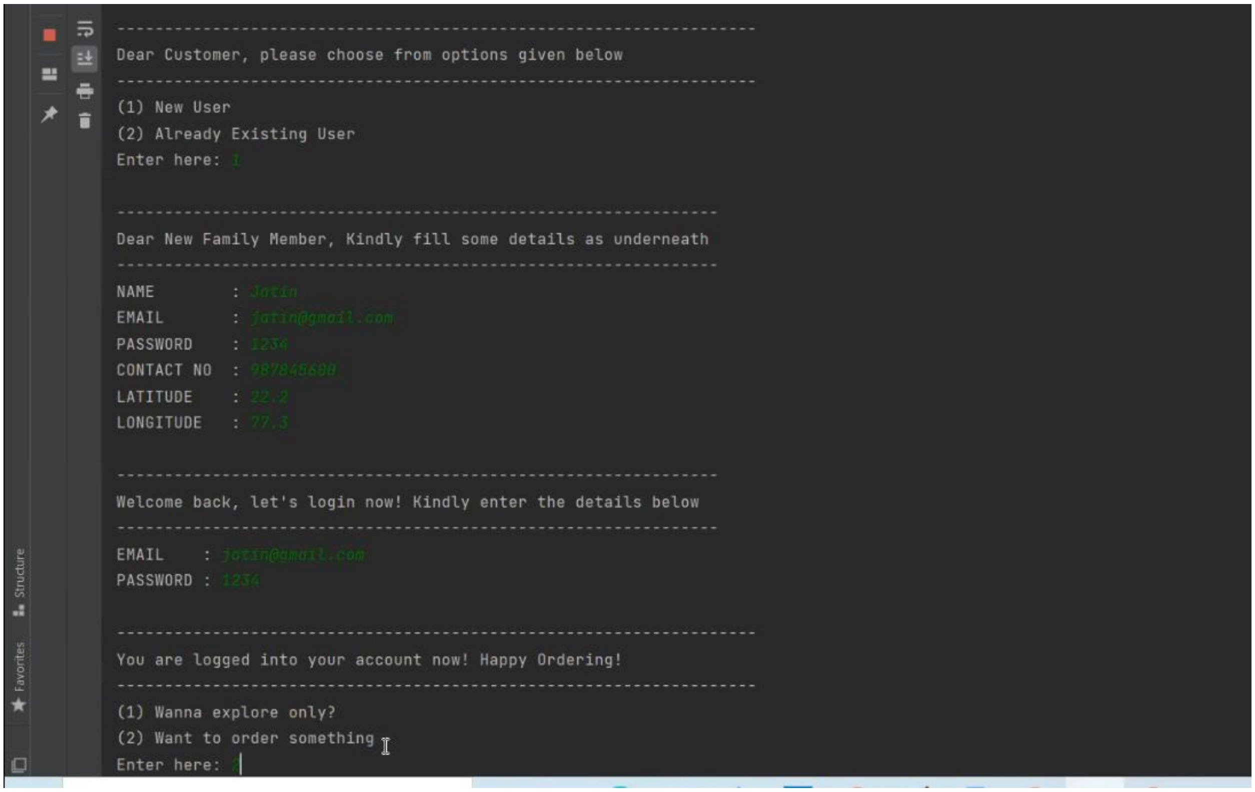
Login page



```
print("\n-------------------------------------------------------------------")
```

```
C:\Users\Simran\AppData\Local\Microsoft\WindowsApps\python3.9.exe C:/Users/Simran/Downloads/main.py
-------------------------------------------------------------------
  WELCOME TO 'LES GOURMANDS' - Your favorite Food Ordering App
-------------------------------------------------------------------


-------------------------------------------------------------------
Dear Customer, please choose from options given below
-------------------------------------------------------------------
(1) New User
(2) Already Existing User
Enter here:
```



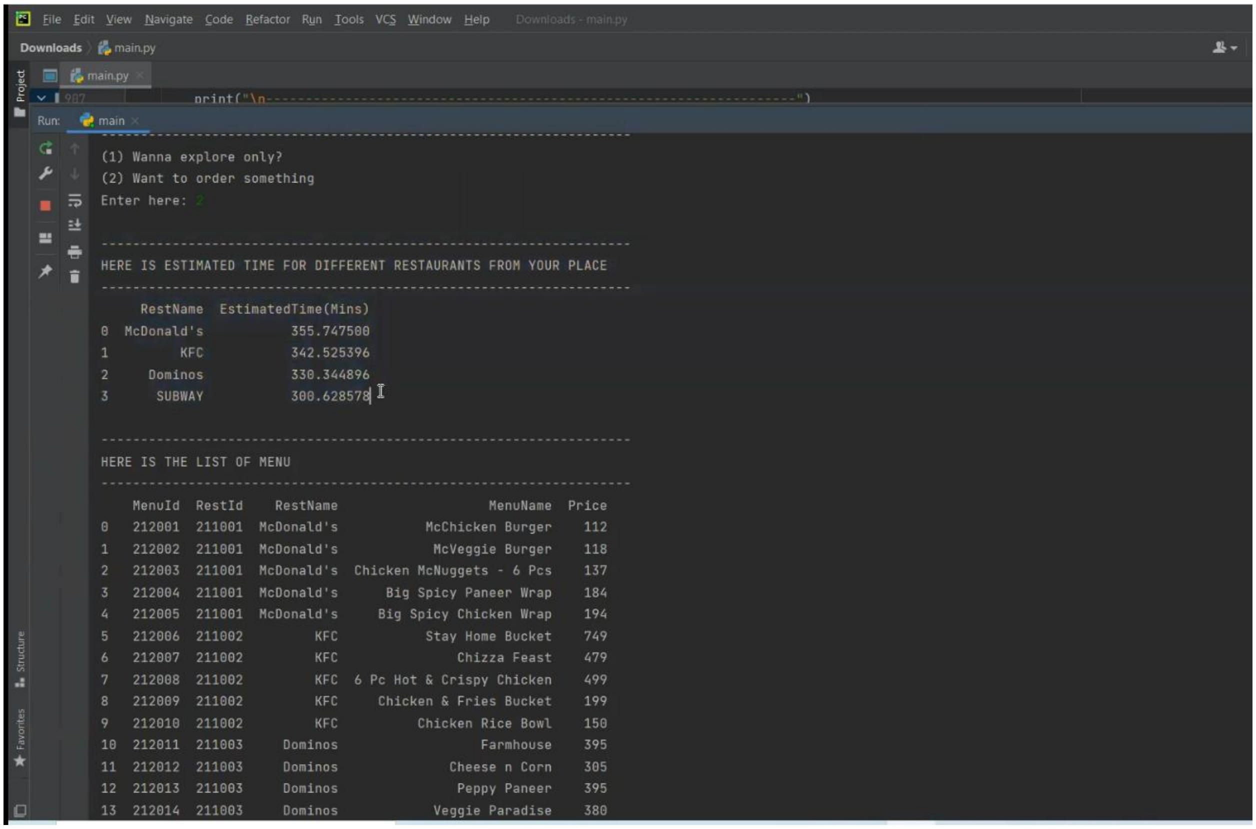
```
-------------------------------------------------------------------
Dear Customer, please choose from options given below
-------------------------------------------------------------------
(1) New User
(2) Already Existing User
Enter here: 1


-------------------------------------------------------------------
Dear New Family Member, Kindly fill some details as underneath
-------------------------------------------------------------------
NAME         : Jatin
EMAIL        : jatin@gmail.com
PASSWORD     : 1234
CONTACT NO   : 987845600
LATITUDE     : 22.2
LONGITUDE    : 77.3


-------------------------------------------------------------------
Welcome back, let's login now! Kindly enter the details below
-------------------------------------------------------------------
EMAIL    : jatin@gmail.com
PASSWORD : 1234


-------------------------------------------------------------------
You are logged into your account now! Happy Ordering!
-------------------------------------------------------------------
(1) Wanna explore only?
(2) Want to order something
Enter here: 2
```

TIME estimates for different restaurants and their menu



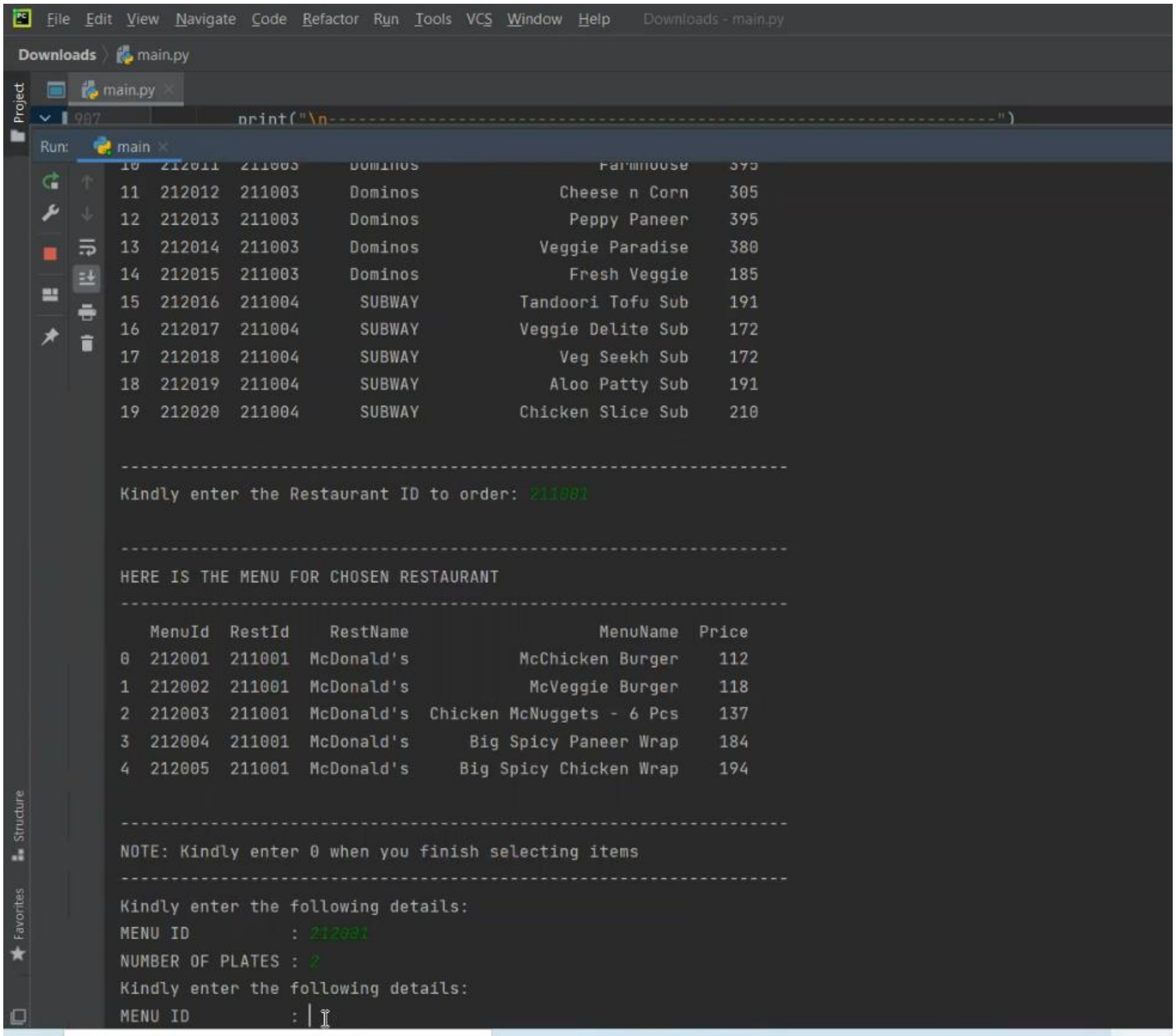
```
print("\n-------------------------------------------------------------------")
```

```
-------------------------------------------------------------------
(1) Wanna explore only?
(2) Want to order something
Enter here: 2

-------------------------------------------------------------------
HERE IS ESTIMATED TIME FOR DIFFERENT RESTAURANTS FROM YOUR PLACE
-------------------------------------------------------------------
     RestName  EstimatedTime(Mins)
0  McDonald's           355.747500
1         KFC           342.525396
2     Dominos           330.344896
3      SUBWAY           300.628578

-------------------------------------------------------------------
HERE IS THE LIST OF MENU
-------------------------------------------------------------------
    MenuId  RestId    RestName                   MenuName  Price
0   212001  211001  McDonald's           McChicken Burger    112
1   212002  211001  McDonald's            McVeggie Burger    118
2   212003  211001  McDonald's  Chicken McNuggets - 6 Pcs    137
3   212004  211001  McDonald's      Big Spicy Paneer Wrap    184
4   212005  211001  McDonald's     Big Spicy Chicken Wrap    194
5   212006  211002         KFC           Stay Home Bucket    749
6   212007  211002         KFC               Chizza Feast    479
7   212008  211002         KFC  6 Pc Hot & Crispy Chicken    499
8   212009  211002         KFC     Chicken & Fries Bucket    199
9   212010  211002         KFC          Chicken Rice Bowl    150
10  212011  211003     Dominos                  Farmhouse    395
11  212012  211003     Dominos              Cheese n Corn    305
12  212013  211003     Dominos               Peppy Paneer    395
13  212014  211003     Dominos            Veggie Paradise    380
```

Menu



File Edit View Navigate Code Refactor Run Tools VCS Window Help Downloads - main.py
Downloads 〉 main.py
main.py
Project
Run: main
Structure
Favorites

```
987          print("\n-------------------------------------------------------------------")
```

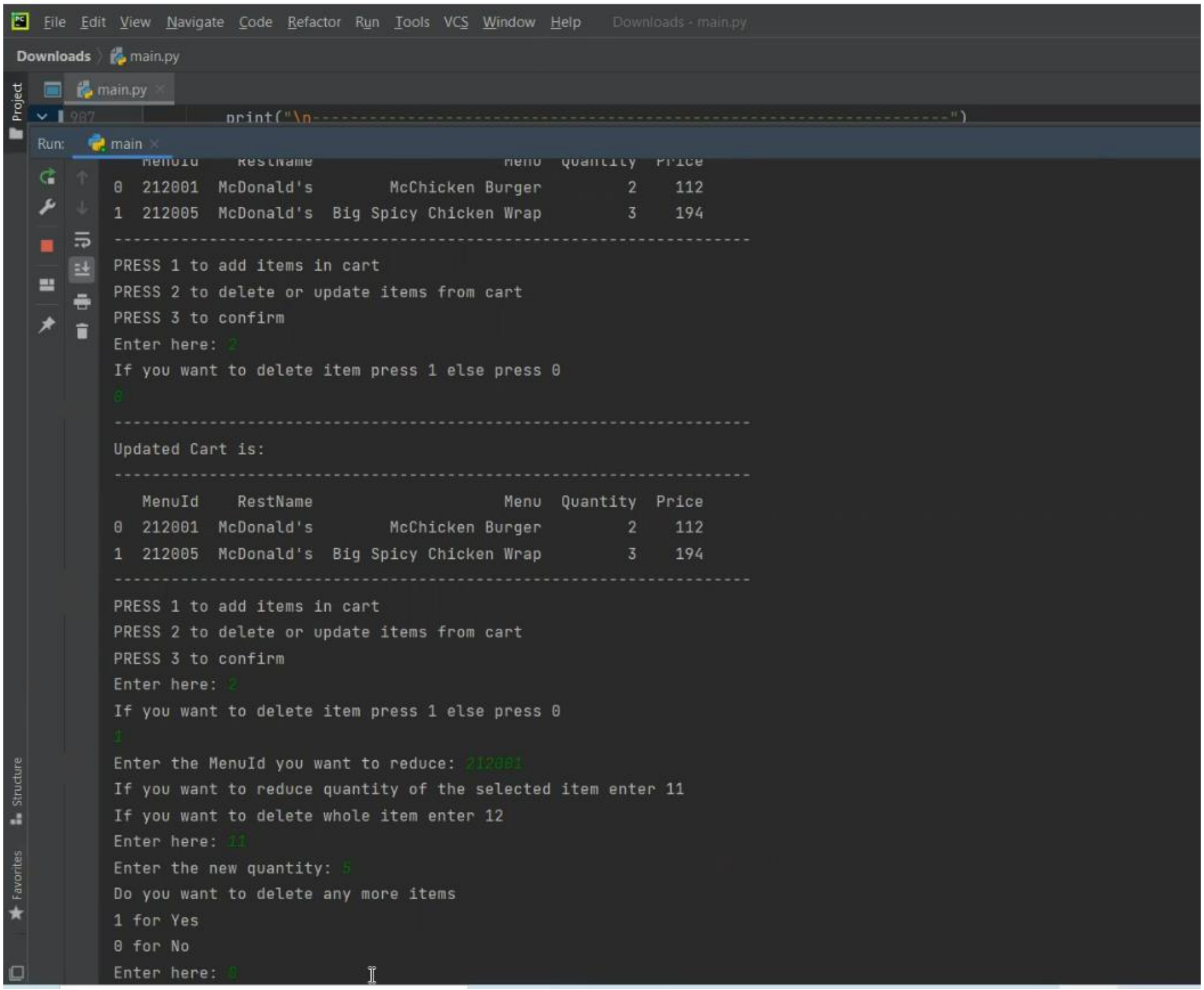
```
10  212011  211003     Dominos              Farmhouse    395
11  212012  211003     Dominos           Cheese n Corn    305
12  212013  211003     Dominos            Peppy Paneer    395
13  212014  211003     Dominos        Veggie Paradise    380
14  212015  211003     Dominos            Fresh Veggie    185
15  212016  211004      SUBWAY      Tandoori Tofu Sub    191
16  212017  211004      SUBWAY      Veggie Delite Sub    172
17  212018  211004      SUBWAY          Veg Seekh Sub    172
18  212019  211004      SUBWAY         Aloo Patty Sub    191
19  212020  211004      SUBWAY     Chicken Slice Sub    210


-------------------------------------------------------------------
Kindly enter the Restaurant ID to order: 211001


-------------------------------------------------------------------
HERE IS THE MENU FOR CHOSEN RESTAURANT
-------------------------------------------------------------------
   MenuId  RestId    RestName                   MenuName  Price
0  212001  211001  McDonald's          McChicken Burger    112
1  212002  211001  McDonald's           McVeggie Burger    118
2  212003  211001  McDonald's  Chicken McNuggets - 6 Pcs    137
3  212004  211001  McDonald's      Big Spicy Paneer Wrap    184
4  212005  211001  McDonald's     Big Spicy Chicken Wrap    194


-------------------------------------------------------------------
NOTE: Kindly enter 0 when you finish selecting items
-------------------------------------------------------------------
Kindly enter the following details:
MENU ID          : 212001
NUMBER OF PLATES : 2
Kindly enter the following details:
MENU ID          : 
```

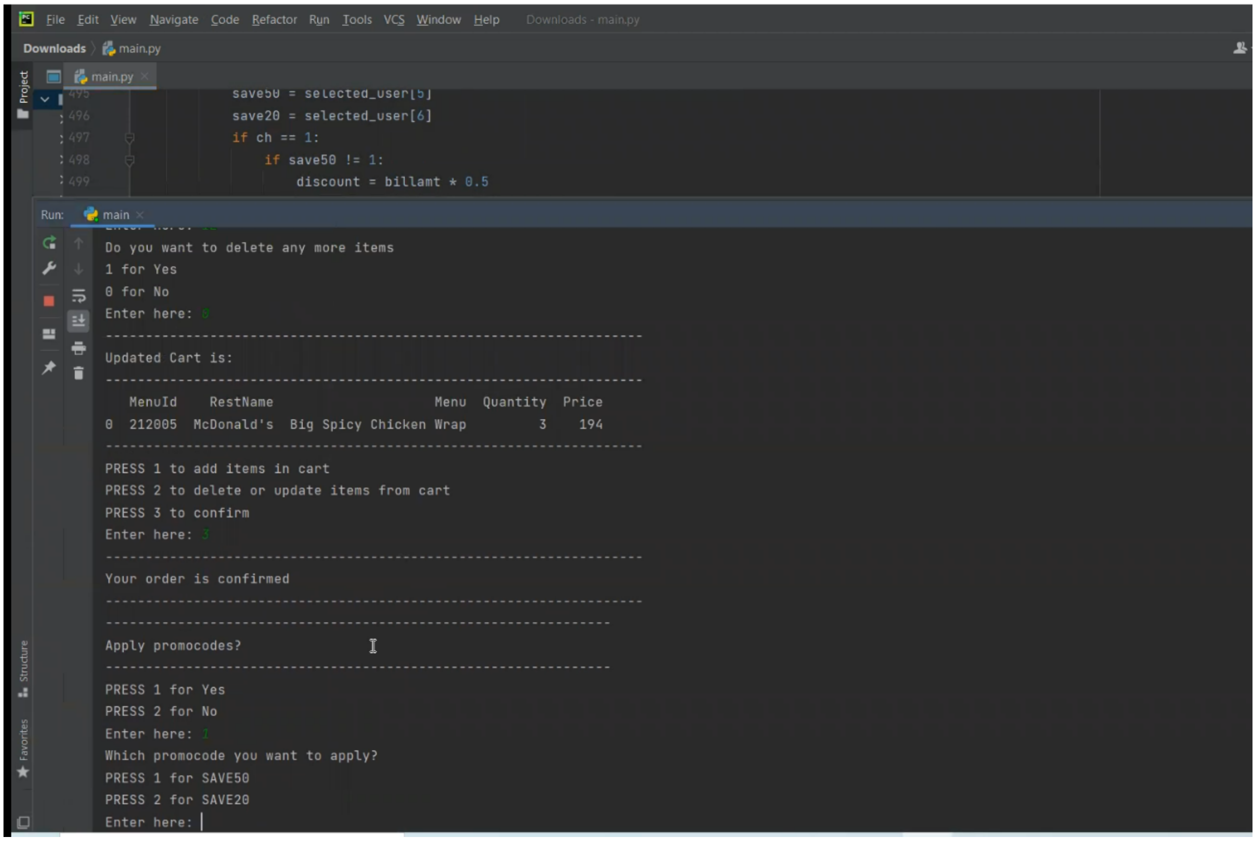
Order Placing and updating



```
print("\n-------------------------------------------------------------------")
```

```
   MenuId    RestName                   Menu  Quantity  Price
0  212001  McDonald's        McChicken Burger         2    112
1  212005  McDonald's  Big Spicy Chicken Wrap         3    194
-------------------------------------------------------------------
PRESS 1 to add items in cart
PRESS 2 to delete or update items from cart
PRESS 3 to confirm
Enter here: 2
If you want to delete item press 1 else press 0
0
-------------------------------------------------------------------
Updated Cart is:
-------------------------------------------------------------------
   MenuId    RestName                   Menu  Quantity  Price
0  212001  McDonald's        McChicken Burger         2    112
1  212005  McDonald's  Big Spicy Chicken Wrap         3    194
-------------------------------------------------------------------
PRESS 1 to add items in cart
PRESS 2 to delete or update items from cart
PRESS 3 to confirm
Enter here: 2
If you want to delete item press 1 else press 0
1
Enter the MenuId you want to reduce: 212001
If you want to reduce quantity of the selected item enter 11
If you want to delete whole item enter 12
Enter here: 11
Enter the new quantity: 5
Do you want to delete any more items
1 for Yes
0 for No
Enter here: 0
```

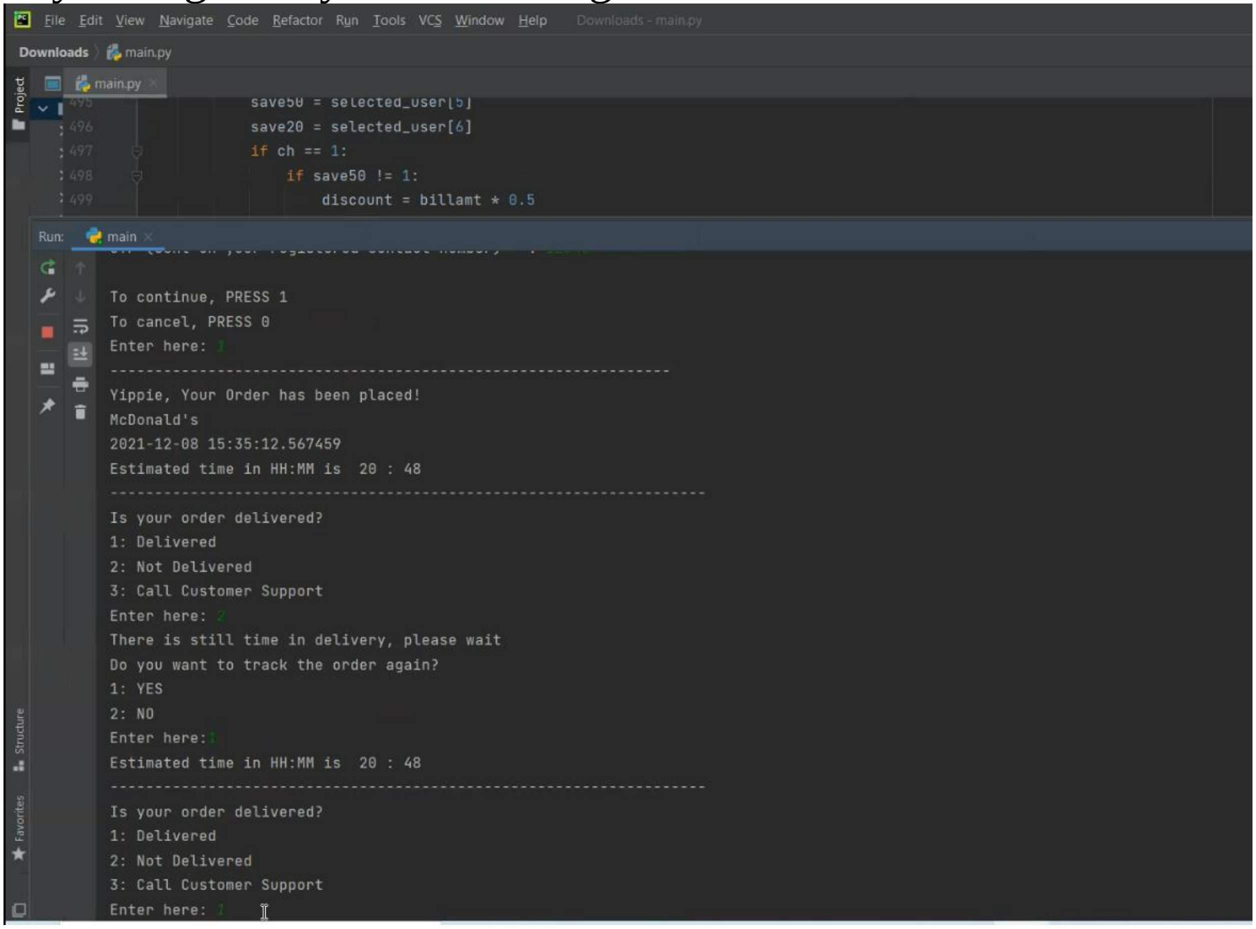
Cart and Promocodes



```python
                save50 = selected_user[5]
                save20 = selected_user[6]
                if ch == 1:
                    if save50 != 1:
                        discount = billamt * 0.5
```

```
Do you want to delete any more items
1 for Yes
0 for No
Enter here: 0
-------------------------------------------------------------------
Updated Cart is:
-------------------------------------------------------------------
   MenuId    RestName                    Menu  Quantity  Price
0  212005  McDonald's  Big Spicy Chicken Wrap         3    194
-------------------------------------------------------------------
PRESS 1 to add items in cart
PRESS 2 to delete or update items from cart
PRESS 3 to confirm
Enter here: 3
-------------------------------------------------------------------
Your order is confirmed
-------------------------------------------------------------------
---------------------------------------------------------------
Apply promocodes?
---------------------------------------------------------------
PRESS 1 for Yes
PRESS 2 for No
Enter here: 1
Which promocode you want to apply?
PRESS 1 for SAVE50
PRESS 2 for SAVE20
Enter here: 
```

Payment gateways and tracking and customer feedback



```
            save50 = selected_user[5]
            save20 = selected_user[6]
            if ch == 1:
                if save50 != 1:
                    discount = billamt * 0.5
```

```

To continue, PRESS 1
To cancel, PRESS 0
Enter here: 1
---------------------------------------------------------------
Yippie, Your Order has been placed!
McDonald's
2021-12-08 15:35:12.567459
Estimated time in HH:MM is  20 : 48
-------------------------------------------------------------------
Is your order delivered?
1: Delivered
2: Not Delivered
3: Call Customer Support
Enter here: 2
There is still time in delivery, please wait
Do you want to track the order again?
1: YES
2: NO
Enter here:1
Estimated time in HH:MM is  20 : 48
-------------------------------------------------------------------
Is your order delivered?
1: Delivered
2: Not Delivered
3: Call Customer Support
Enter here: 1
```

4 . Non Functional Requirements

4.1 . Hardware Interfaces

Server Side:

- Operating System: Windows 9x or above, MAC or UNIX.
- Processor: i3.0 GHz or higher
- RAM: 256 Mb or more
- Hard Drive: 2 GB or more

Client side:

- Operating System: Windows 9x or above, MAC or UNIX.
- Processor: Pentium III or 2.0 GHz or higher.
- RAM: 256 Mb or more

4.2 . Software Interfaces

- Database (food.db): SQLITE3 Server.
- Python support

5 . Other Nonfunctional Requirements

5.1 . Performance Requirements

The proposed system that we are going to develop will be used as the Chief performance system within the different places of the delivery which interact with the delivery staff and delivery's. Therefore, it is expected that the database (food.db) would perform functionally all the requirements that are specified by the delivery.

5.2 . Safety Requirements

The database (food.db) may get crashed at any certain time due to virus or operating system failure. Therefore, it is required to take the database (food.db) backup. Also, the data has to be stored safely.

5.3 . Security Requirements

We are going to develop a secured database (food.db) for the delivery. There are different categories of users namely staff, administrator, Food Delivery staff, deliverys etc . Depending upon the category of user the access rights are decided. It means if the user is an administrator then he can be able to modify the data, delete, append etc . All other users other than Food Delivery staff only have the rights to retrieve the information about database (food.db).

5.4 . Software Quality Attributes

The Quality of the database (food.db) is maintained in such a way so that it can be very user friendly to all the users of the database (food.db).

5.5 Hardware Constraints

The system requires a database (food.db) in order to store persistent data. The database (food.db) should have backup capabilities.

5.6 Software Constraints

The development of the system will be constrained by the availability of required software such as terminal servers, database (food.db) and development tools.

The availability of these tools will be governed by the Delivery.

The most recent versions of software development tools may not be installed at the Delivery.

6. Limitations of Project

Following can be some limitations

1. It can not refund the amount from the restaurant side through the application.
2. Needs to get updated when any changes are to be made
3. OTP is not authenticated

All of them can be improved by adding more functionalities but it was beyond scope of this project.