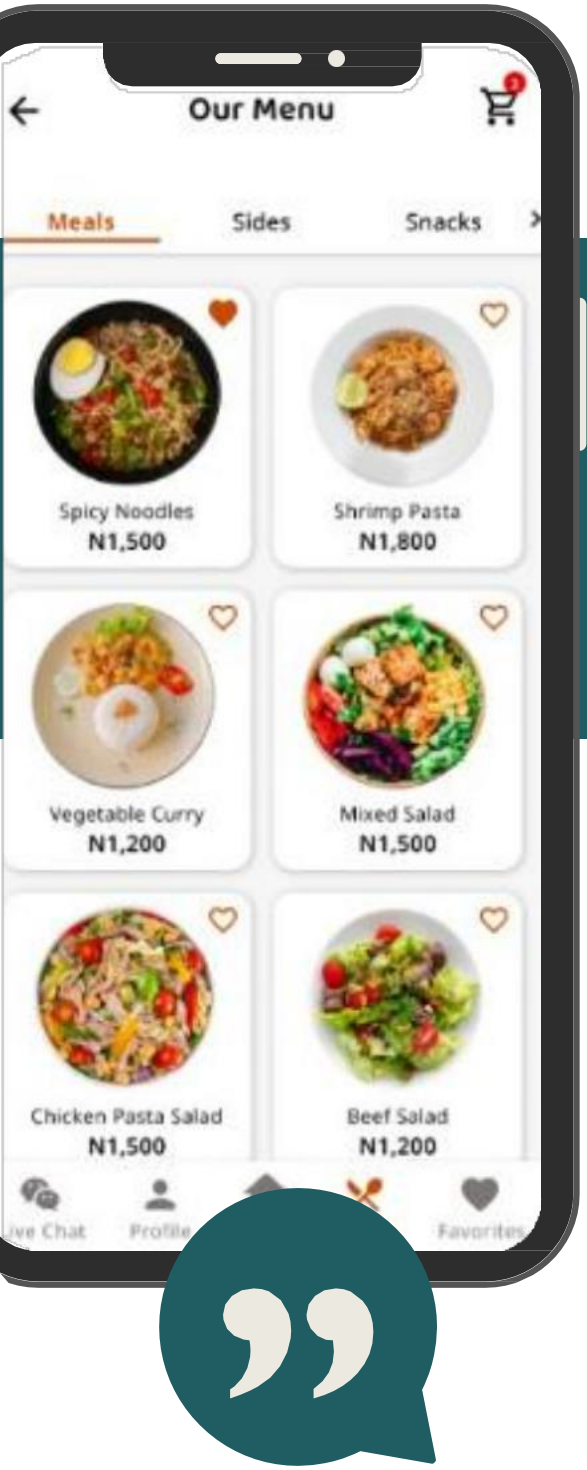


FOOD ORDER AND DELIVERY APPLICATION

OBJECTIVES

To assist owners in obtaining the delivery management procedure' transparency plan and providing customers for a way to place an order at a restaurant over the internet.. Give your clients a high level of information transparency.
And encourage automated task and person allocation.

AUTHORED BY:

- **Shivam Vijayvargiya**

Introduction

- Online Food Ordering System is a user interface. The Online Food Ordering System's main purpose is to maintain track of information such as Item Category, Food, Delivery Address, Order, and Shopping Cart. It keeps track of information about the Item Category, the Customer, the Shopping Cart, and the Item Category. Only the administrator gets access to the project because it is totally built at the administrative level. The project's purpose is to develop software that will cut down on the time spent manually managing Item Category, Food, Customer, and Delivery Address. It saves the Delivery Address, Order, and Shopping Cart information.

Problem Definition

- The technology we recommend is an easy-to-use online meal ordering system for customers. The procedure of taking a customer's order is made easier with this technology. Customers may place orders fast utilizing the online meal ordering system, which generates an online menu. Customers can also use a meal menu to keep track of their orders.

- The next part provides an overview of the Software Requirements Specification developed from the subject Online Food Ordering System. To begin, the document's purpose and intended audience are described. The scope of the project is then specified in the paper, with a special emphasis on what the resulting programmed will do and the benefits that come with it.

Problem Solution

- We have used Python modules such as pandas(series, data frame), matplotlib, pyplot, numpy, Cx_Oracle. This application will give easy access to user to order the food delivery.

- We have connected the python and oracle to get easy access of the data.

Implementation:

1. This code shows how the user will login the application.

```
def first():
    print('~' * 160)
    print('~' * 64, 'FOOD ORDERING AND DELIVERY', '~' * 64)
    print('~' * 160)
    print("\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\tWhat do you want to do?")
    print("\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t1) USER LOGIN ")
    print("\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t2) ORDER FOOD ")
    print("\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t3) GRAPHICS ")
    print("\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t4) EXIT APPLICATION ")
    print('~' * 160)
    print('~' * 160)

####################################### creting connection between python and oracle ###################################

    import cx_Oracle
    import numpy as np
    import pandas as pd
    import matplotlib.pyplot as plt
    connection = cx_Oracle.connect(user="system", password="1234", dsn="localhost/XE")
    cursor = connection.cursor()

    userinput = int(input("Enter your choice(1-4):"))

################################################### USER LOGIN PAGE ################################################

    if userinput == 1:

        def insert_customer(cid, cname, caddress, carea, ccontact):
            cursor.close()
            connection.commit()

        try:
            q = int(input("Enter Customer id:"))
            if(q >= 6001 and q <= 6030):
                print('User aldready exist')

            else:
                w = input("Enter Customer name:")
                e = input("Enter Customer address:")
                r = input("Enter Customer's area:")
                t = input("Enter Customer contact no.:")

                insert_customer(q, 'w', 'e', 'r', t)
                print("Congrats!! You are successfully registered on our application")

            sql = ("select * from customer where cid = :abcd")
            cursor.execute(sql, abcd=q)
            res = cursor.fetchall()
            df = pd.DataFrame(res, columns=['Cid', 'Cname', 'Caddress', 'carea', 'ccontact'])
            print(df)


        except Exception as e:
            print(e)
        first()
```

2. This code shows how the user can get access to the shops.

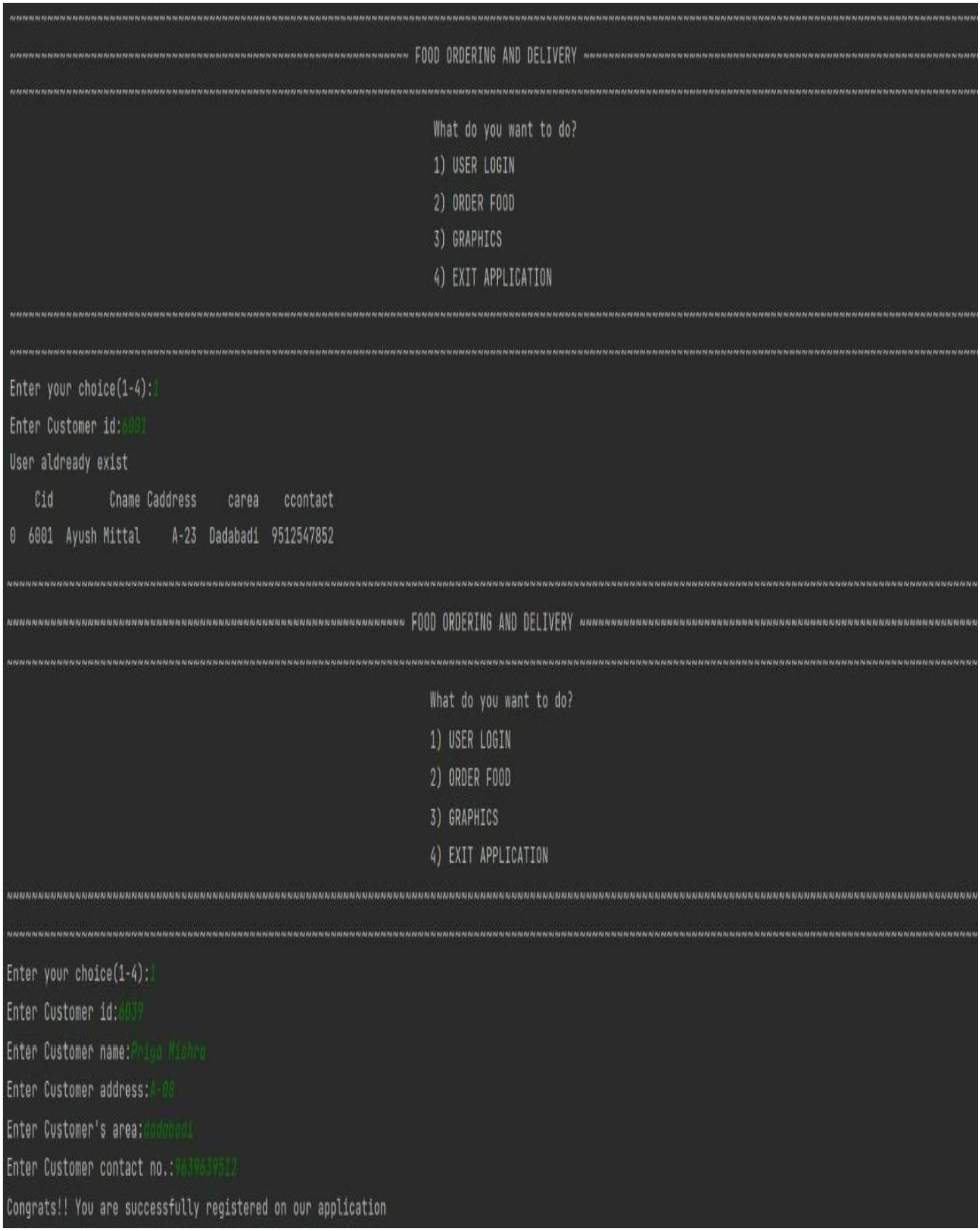
```
57
58
59  ################################################## ORDER FOOD ##################################################
60
61      if userinput == 2:
62          print("Select a shop to see its menu")
63          cursor.execute(
64              """
65              Select * from shop
66              """
67          )
68          res = cursor.fetchall()
69          df = pd.DataFrame(res, columns=['Shop_Id', 'Shop_Name', 'Area', 'Address', 'Rating', 'Contact', 'Did'])
70          print(df)
71
72
73  ################################################## MENU DISPLAY ##################################################
74
75
76
77          shopinput = input("Enter shop name: ")
78          sql = (
79              "select a.shopid, a.shopname, b.pid, b.pname, b.pprice, b.veg_nonveg from menu b, shop a "
80              "where a.shopid = b.shopid AND shopname = :a")
81          cursor.execute(sql, a=shopinput)
82          res = cursor.fetchall()
83          df = pd.DataFrame(res, columns=['Shop_Id', 'Shop_Name', 'pid', 'pname', 'pprice', 'veg_nonveg'])
84          print(df)
85
86
87
88  ################################################## MENU Selecting ##################################################
89
90
91
92          menuinput = input("Enter what do you want to order: ")
93          menuinpu = input("Enter what do you want to order: ")
94          sql = ("select a.shopid, a.shopname, b.pid, b.pname, b.pprice, b.veg_nonveg from menu b, shop a "
95                 "where a.shopid = b.shopid AND shopname = :a AND (pname = :b OR pname = :z)")
96          cursor.execute(sql, a=shopinput, b=menuinput, z = menuinpu)
97          res = cursor.fetchall()
98          df = pd.DataFrame(res, columns=['Shop_Id', 'Shop_Name', 'pid', 'pname', 'pprice', 'veg_nonveg'])
99          print(df)
100
101
102
103         a1 = input("Please enter your customer ID: ")
104
105
106
107         print("Congrats. Your order is successfully placed!!!")
108         print("Your order will be delivered to this address within 30 minitues!!")
109         sql = ("select caddress, carea from customer where cid = :c")
110         cursor.execute(sql, c=a1)
```

3. This code tells how pie chart are drawn.

```
        res = cursor.fetchall()
        df = pd.DataFrame(res, columns=['Address', 'area'])
        print(df)
        first()



########################################################## GRAPHICS ##########################################################



    if userinput == 3:

        print("1. Number of veg and non veg items.")
        print("2. Number of number of shops by area.")


        cursor.execute("select veg_nonveg, count(veg_nonveg) from menu group by veg_nonveg")
        goes = []
        rows = []
        for row in cursor:
            goes.append(row[0])
            rows.append(row[1])
        plt.pie(rows, labels=goes, shadow=True)
        plt.show()



        cursor.execute("select area, count(area) from shop group by area")
        goes = []
        rows = []
        for row in cursor:
            goes.append(row[0])
            rows.append(row[1])
        plt.pie(rows, labels=goes, shadow=True)
        plt.show()
        first()

##################################################### EXIT APPLICATION ###################################################

    if userinput == 4:
        print("Thnak you for using our application")
        exit()

################################################## Calling the entire function ###########################################

first()
```

Result

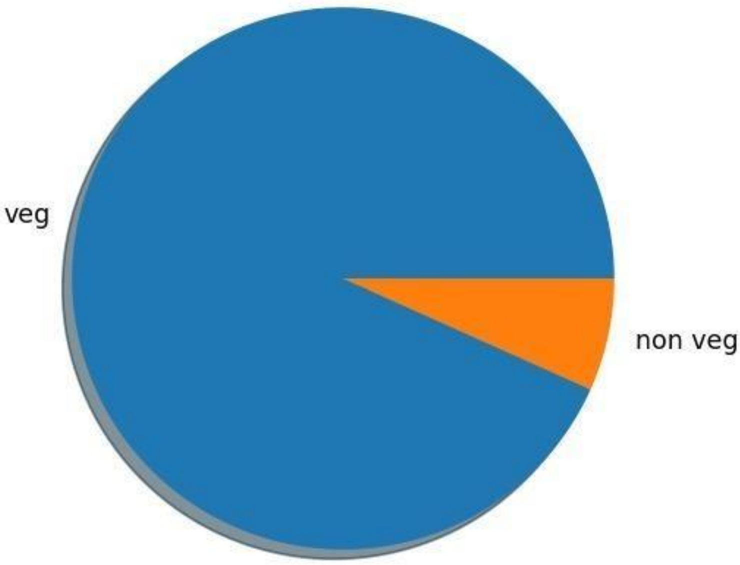
1. This result shows that the user is successfully registered.

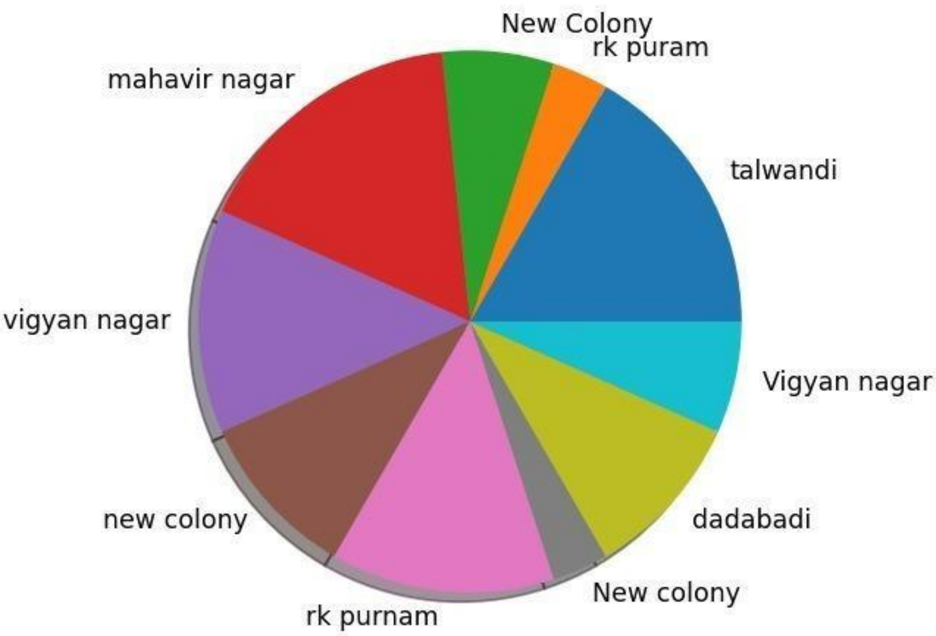


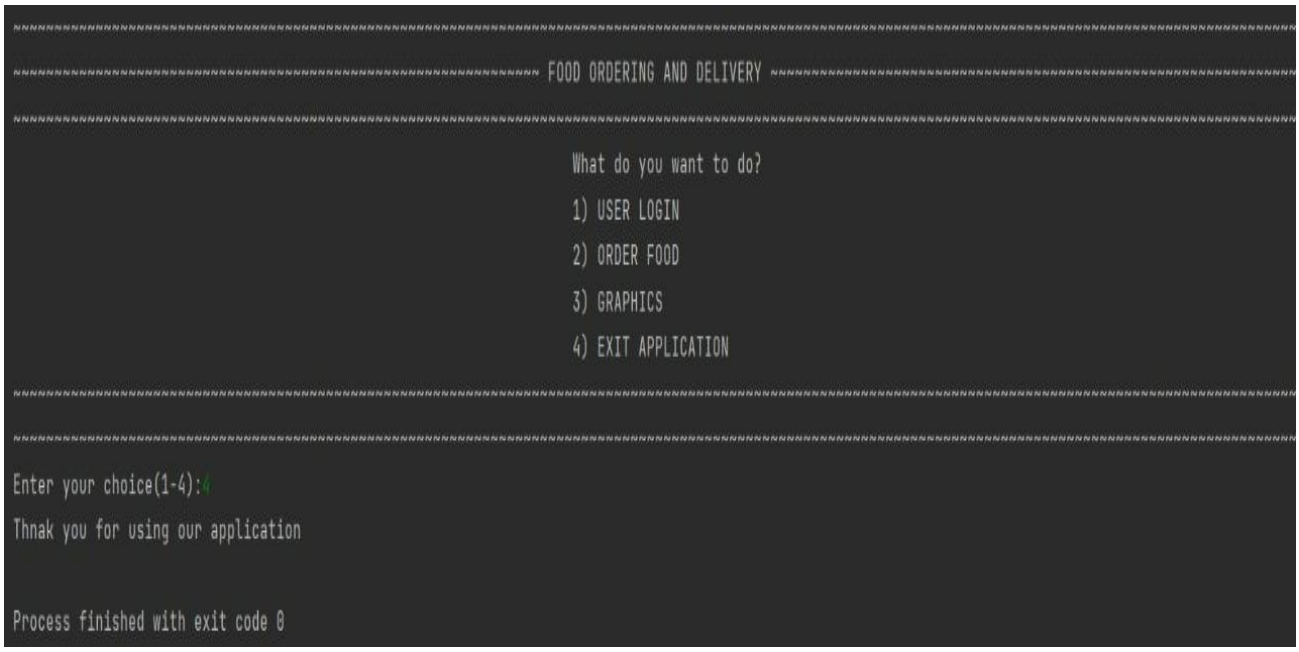
2. This result shows the shops which user can access.

```
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ FOOD ORDERING AND DELIVERY ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
                                              What do you want to do?
                                              1) USER LOGIN
                                              2) ORDER FOOD
                                              3) GRAPHICS
                                              4) EXIT APPLICATION
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter your choice(1-4):2
Select a shop to see its menu
    Shop_Id     Shop_Name          Area  ... Rating     Contact   Did
0      1001       dominos  mahavir nagar  ...    4.2  9874563200  2001
1      1002     macdonalds      talwandi  ...    4.8  6565659874  2002
2      1003        subway   vigyan nagar  ...    4.9  5987452145  2003
3      1004     burgerking      dadabadi  ...    4.2  5959595959  2004
4      1005    buskinrobins    new colony  ...    4.5  7897894545  2005
5      1006      pizzahut        rk puram  ...    3.2  4564569852  2006
6      1007         kanha      talwandi  ...    4.5  7854125478  2007
7      1008           taj   vigyan nagar  ...    4.0  9856985654  2008
8      1009        faasos      dadabadi  ...    3.2  5959595900  2009
9      1010         rawat  mahavir nagar  ...    3.5  9595959565  2010
10     1011      rollshub    new colony  ...    4.2  8974532654  2011
11     1012           DMB      rk purnam  ...    4.0  9856458741  2022
12     1013           LMB  mahavir nagar  ...    4.2  8956254125  2013
13     1014           JMB      dadabadi  ...    3.2  9547896521  2014
14     1015       agrawal      talwandi  ...    5.0  9856555555  2015
15     1016          RJ14      rk purnam  ...    4.2  7897896555  2016
16     1017      Naturals    New Colony  ...    1.2  5252525252  2017
17     1018   Brown Sugar   Vigyan nagar  ...    3.0  8989890000  2018
18     1019           BMB   Vigyan nagar  ...    4.5  9010125478  2019
19     1020        Bhagat     New Colony  ...    4.2  7032578952  2020
20     1021       Mudowen      rk purnam  ...    3.2  9012547852  2021
21     1022     Burgerfarm      talwandi  ...    3.2  9012300044  2022
22     1023       sodhani  mahavir nagar  ...    4.2  8900015632  2023
23     1024    maheshwari     new colony  ...    2.2  8965000147  2024
24     1025          namo   vigyan nagar  ...    2.4  8965451254  2025
25     1026      vegbites  mahavir nagar  ...    2.9  9595959565  2026
26     1027           NBC      rk purnam  ...    4.8  5999987854  2027
27     1028       Thelama      talwandi  ...    4.2  5896541258  2028
28     1029       bakingo   vigyan nagar  ...    4.3  6969598979  2029
29     1030        Eatfit     New colony  ...    4.1  5987453200  2030

[30 rows x 7 columns]
Enter shop name: LMB
   Shop_Id Shop_Name    pid          pname  pprice veg_nonveg
0     1013       LMB  10131  chole bhature      90        veg
1     1013       LMB  10132    raj kachori      60        veg
2     1013       LMB  10133    masala dosa     150        veg
3     1013       LMB  10134          laddu      30        veg
4     1013       LMB  10135     kaju katli     450        veg
Enter what do you want to order: laddu
Enter what do you want to order: raj kachori
   Shop_Id Shop_Name    pid        pname  pprice veg_nonveg
0     1013       LMB  10132  raj kachori      60        veg
1     1013       LMB  10134        laddu      30        veg
Please enter your customer ID: 6001
Congrats. Your order is successfully placed!!!
Your order will be delivered to this address within 30 minitues!!
  Address     area
0    A-23  Dadabadi
```

3. This result shows the pie charts.







Conclusion

This application helps the user to directly interact with the database to insert, retrieve and update the values. The user can specify the data that he needs. He can also view graphics related to his or her data. Based on the result of this project, it helps customer in making order easily. It gives information needed in making order to customer. The Food website application made for restaurant and mess can help restaurant and mess in receiving orders and modifying its data and it is also made for admin so that it helps admin in controlling all the Food system.

References

1. https://www.oracle.com/database/technologies/appdev/python/quickstartpythononprem.html
2. https://www.geeksforgeeks.org/oracle-database-connection-in-python

Git Hub Link:

https://github.com/shivamV21/Python