



Middlesex
University
Mauritius

4/9/2021

CST1510 Programming for Data Communication and Networking

Final Coursework

Avinesh Culloo

(M00776456)



Contents

Introduction.....	2
Server-Side Software System.....	3
Client-side Software System	6
Some screenshots of the GUI	6

Introduction

This project is based on designing and implementing a 'Vending Machine' for sushi. It is a Client/Server-Side software system built in Python to interact with the user on the client-side and the data storage is processed on the server-side. The vending machine under the name of 'ケシャフ sushi' (Keshav in Japanese) will allow the user to carry out the following operations when generating an order.

- 1) Start with a welcome message.
- 2) Choose sushi according to a category and then, he/she will have to enter the quantity the latter wish to purchase.
- 3) A window will display the sushi ID, sushi's name, individual price, and the total price.
- 4) After each order place, the vending machine will allow the following list of options:
 - a. Add Another
 - b. Finish and Pay
 - c. Cancel
- 5) Add Another operation will allow the user to add another sushi to his/her cart.
- 6) Finish and Pay will display the final receipt containing the list of items, individual price, total price and proceed to make the payment.
- 7) Cancel will display a message and go back to the Welcome window.
- 8) The user can make payment either by cash or card. If payment is done by cash, then the change will also be displayed.
- 9) Finally, after payment, a goodbye message will be displayed and redirecting the user back to the welcome message.

Python

Python is a high-level programming language that can be used for the following applications:

- 1) Web Development: Django, Pyramid, Bottle, Tornado, Flask, web2py
- 2) GUI Development: tkinter, PyGObject, PyQt, PySide, Kivy, wxPython
- 3) Scientific and Numeric: SciPy, Pandas, IPython
- 4) Software Development: Buildbot, Trac, Roundup

For this coursework, tkinter was used for the GUI development.

Tkinter

Tkinter is not the only GUI Programming toolkit for python as mentioned above, but it is the most commonly used one. It provides a powerful object-oriented interface to the Tk GUI toolkit.

Tkinter was imported by running the following code as shown in fig1:

fig 1

```
from tkinter import *
```

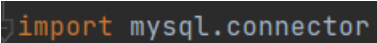
Server-Side Software System

The server will carry out the following operations:

- 1) The server will set up the connection to communicate to the client and also with the database.
- 2) It will retrieve the sushi details from a database stored in MySQL database service into a list.
- 3) The server will send the list to the client.
- 4) After the order is placed at the client-side, the quantity ordered along with the sushi ID will be sent back to the server-side.
- 5) The server will update the quantity in the database according to the sushi ID.

MySQL Database Service

To connect to a database from python, we have to import MySQL library as shown in fig 2 below.

fig 2 

The database will contain 4 tables of the following name:

- 1) categories
Attributes: categoriesID(int), Name (varchar).
- 2) food
Attributes: foodID(int), categoriesID(int), productid(int), price(int), qty(int).
- 3) platters
Attributes: plattersID(int), platterName(varchar), price(int), qty(int).
- 4) products
Attributes: productid(int), Name(varchar).

In order, for the server software system able to extract records from the database, a connection has to be set up.

Fig 3 will show a screenshot of the code to be amended in server.py.

Fig 4 will show a screenshot of how to find the Connection Name, Hostname, Port, Username, Password.

Fig 3

```
HOST = '127.0.0.1' # ← insert the HOST of the database
PORT = 3306 # ← insert the PORT of the database

mydb = mysql.connector.connect(
    user='root', # ← insert the username
    password='*****', # ← insert the user password
    host=HOST,
    database='sushi', # ← insert the database name here
    port=PORT,
    raise_on_warnings=True, # whether warnings should raise exceptions.
)
```

Fig 4

Connection Name: final_Project

Connection Remote Management System Profile

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Parameters SSL Advanced

Hostname: 127.0.0.1 Port: 3306 Name or IP address of the server host - and TCP/IP port.

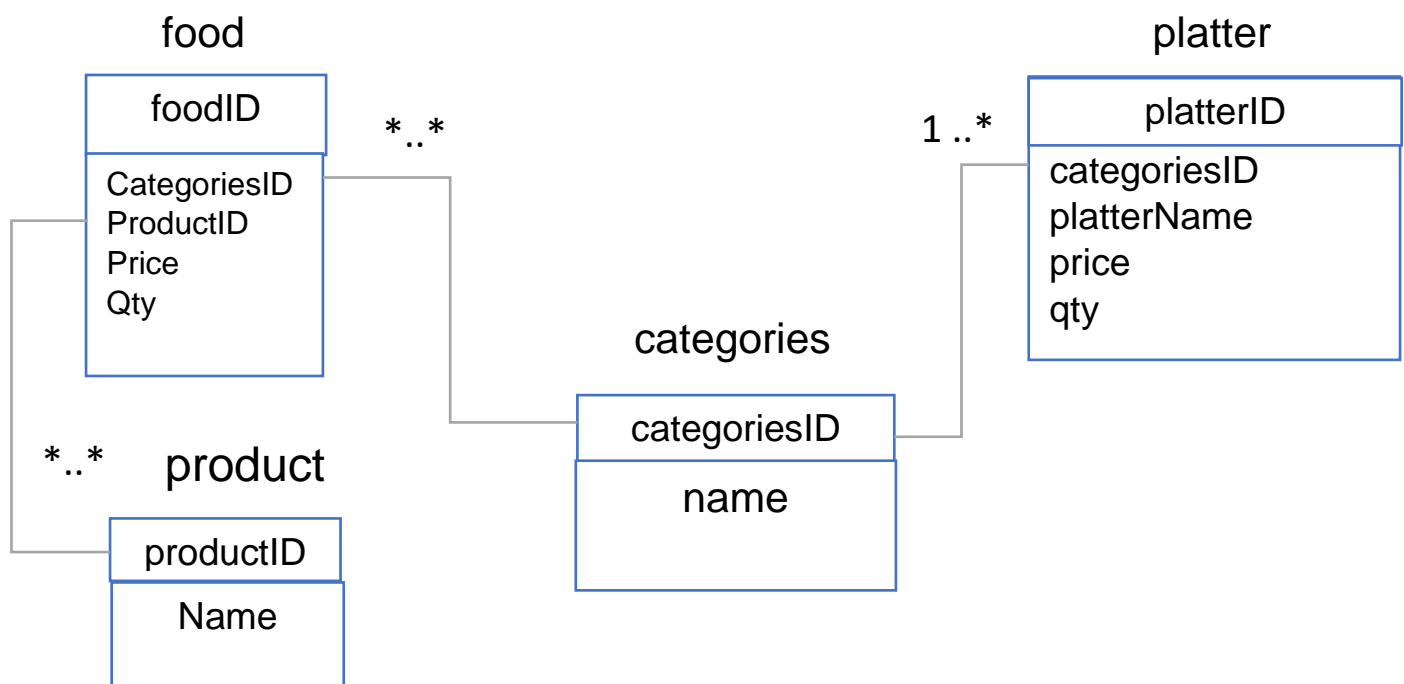
Username: root Name of the user to connect with.

Password: Store in Vault ... Clear The user's password. Will be requested later if it's not set.

Default Schema: The schema to use as default schema. Leave blank to select it later.

Moving forward, the file named 'execution.sql' should be run in the database to create the tables mentioned above.

The UML diagram of the database is shown below.



Socket Programming

Sockets will act as a point-to-point channel of communication between the client and the server. The socket library was imported as shown in fig 5 below.

fig 5

```
import socket
```

Threading in communication

When communication is set up, once it received or sent information from/to the client, the communication will terminate. In order, to keep the communication active, threading was used.

Threading was imported as shown in fig 6 below.

fig 6

```
from _thread import *
```

Client-side Software System

On the client-side, the GUI will be implemented which will allow the user to carry out the following operations:

- 1) Choose a category
- 2) Choose sushi from that specific category and choose the quantity (0 to 10) and place the order.
- 3) The user can choose another sushi.
- 4) The final receipt will be displayed and then, proceeding to pay (either by cash or by card).

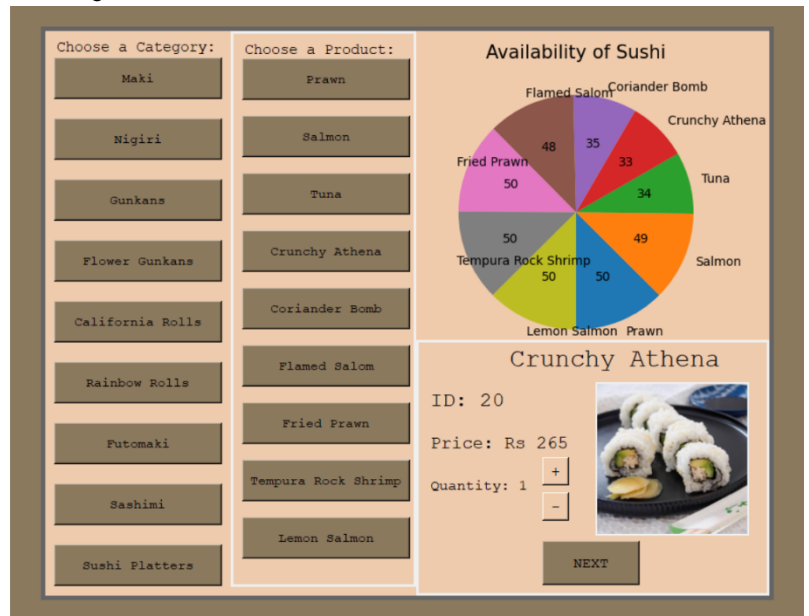
Some screenshots of the GUI

- 1) Fig 7 shows the welcome window.
- 2) Fig 8 shows the menu window

Fig 7



Fig 8



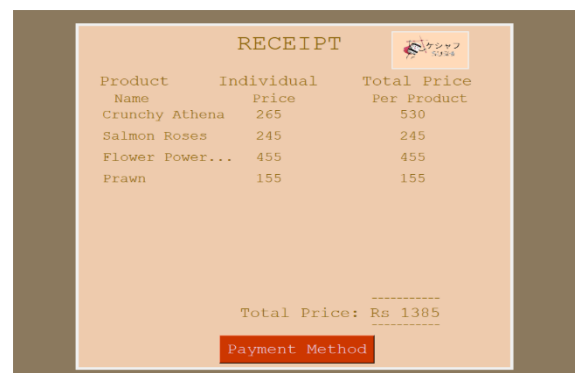
- 3) Fig 9 shows the individual product details window.

Fig 9



- 4) Fig 10 shows the final receipt window.

Fig 10



5) Fig 11 shows the payment method.

Fig 11.A

CASHCARD

2000

Enter your cash

20001000500200100

50251051

CEVALIDATE

Your change: Rs 615

Finish

Fig 11.B

CASHCARD

Name

Avinesh Culloo

Account Number

5368 9501 6424 0096

CVV

788

VALID THRU

08/24

Finish