

**CENG 241 TERM PROJECT**

Restaurant Management System

**Team Members:**

Sultan Gürbüz 202211032

Cevdet Alp Özgün 202211406

Şevval Kaymazlı 202211039

İlker Feza Çoban 202211403

Content:

[1.Introduction: 2](#_Toc155462199)

[2.Purpose of the Program 3](#_Toc155462200)

3.Design of The Program

[4.Testing Results & explanation 6](#_Toc155462202)

[7](#_Toc155462203)

[Admin Interface: 8](#_Toc155462204)

[Employee Interface: 11](#_Toc155462205)

[EXIT 13](#_Toc155462206)

[5. Conclusion: 14](#_Toc155462207)

[5.1 Review & Comment: 14](#_Toc155462208)

[5.2 Suggestions: 14](#_Toc155462209)

# 

# 1.Introduction:

* Program Language: C++
* Main Program Name: Restaurant Management System
* Compiler: Visual Studio and VS Code

# 2.Purpose of the Program

Goal: We aim to help both the employees and the owners of a restaurant by streamlining the process of handling their respective tasks.

We allow,

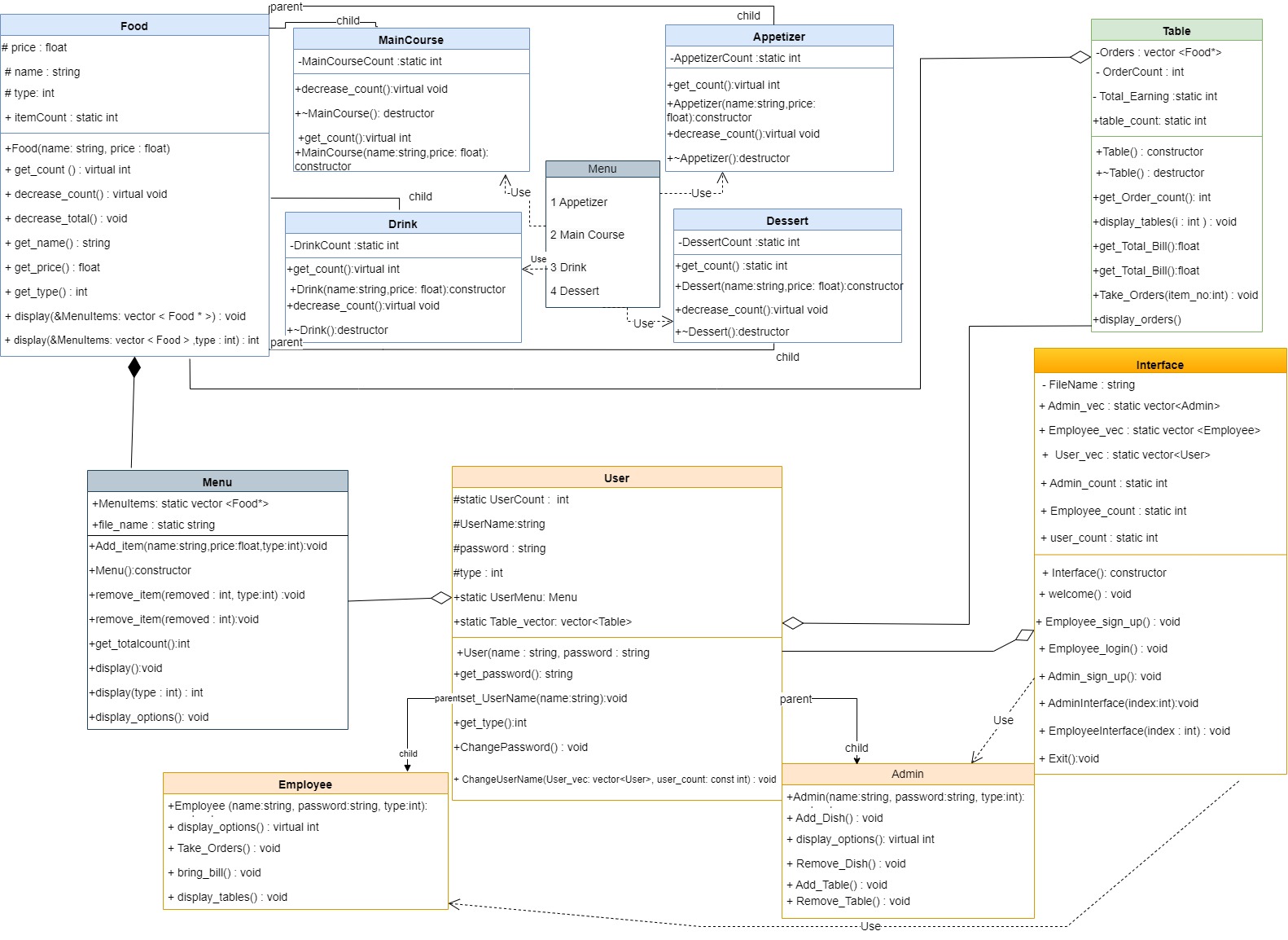
* Employees to take orders for each table separately,
* Display all tables in order to help customers to get an available one faster,
* Calculate and display the total bill for each table,
* Change their account details as they please,
* Allow admins(owners) to add and remove dishes from the menu without causing issues with previously ordered dishes,
* Adding and removing tables as they see fit if no customer is currently using the table that admin decided to remove.
* The menu and employee information is taken from a user-given file.

In the context of this restaurant's operations, admins, who are the owners of the restaurant, can manage the food menu by adding items with their names and prices, and removing the items from the menu. Also, if they want to make a change in the restaurant, they are allowed to whether to add or remove the tables, depending on the customers' situations. As well, the aim of that is to facilitate the workload of the employees and admins. Additionally, they are allowed to change their account details as they please, such as their username and password.

On the other side, the employees can take orders from the customers using the given numbers on the menu one by one and write down the orders in the customer's table list. Furthermore, they can display all tables along their status (in use or available) in order to help customers to get an available one more quickly. Also, they can calculate and display the total bill for each table. Similar to the admins, employees have the privilege to change their account details.

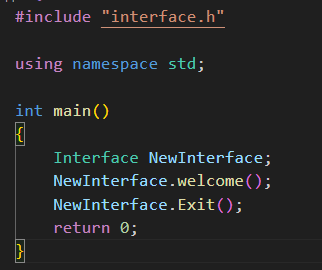
In addition to these functionalities, the "users" and the "menu" files are taken from the users at the beginning of the program, in order to function properly. They can input the files by following the provided guide. (The user file contains information about the admins and employees, while the menu file contains the dish properties such as their name and price)

To use this program, every user must have a unique username and a password. Once signing up for the program, users can reuse it by simply logging in.

lts: **3.Design of The Program**

# 3.Design of The Program

# 4.Testing Results

First, user must upload the database text files.

Inside of constructor of Interface class, taking the user

information and menu from the text files, and fill the Users,

Employee and Admin vector before start.

Then the welcome() function is used for login and sign up for users.

metin, yazı tipi, ekran görüntüsü, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Syntax of Users.txt

For login you need a username and password. The program takes the users Information from the Users text file.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturulduAnd you have only three attempts for sign in, in case of forget your password, you cannot use the program.

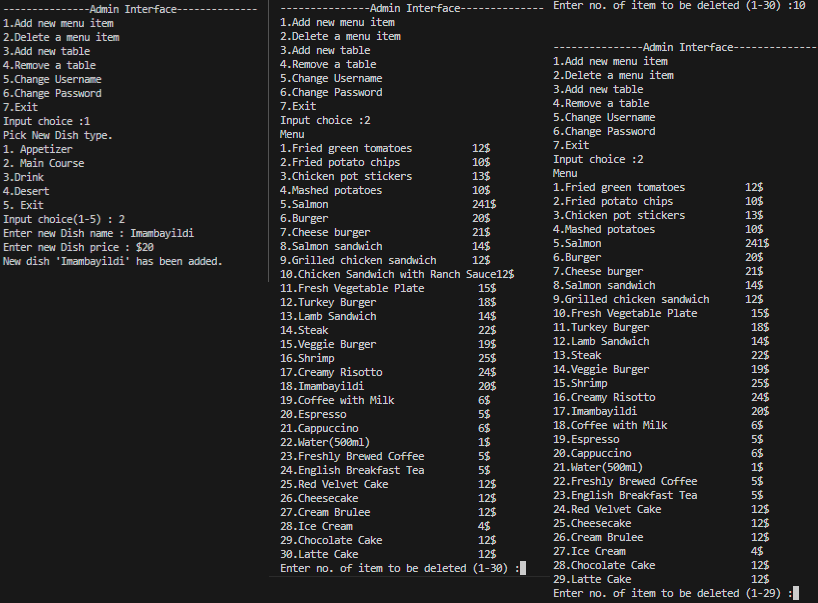
Users can sign up if the username is unique.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

# metin, ekran görüntüsü, yazı tipi içeren bir resim Açıklama otomatik olarak oluşturuldu

# Admin Interface:

-Adding and Removing menu items:

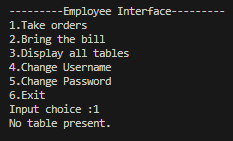
Admin can add a new menu item and can also delete an item from the menu.

For example, Admin deleted the 10th item of the menu. It automatically decreases the number of items.

-Admin can add a new table to the restaurant.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

In the beginning program, it has no table for employee to taking orders, so admin must add table.

If there is no table in the restaurant, employee cannot take orders.

If an employee wants to take order, Admin must add table to the restaurant.

metin, ekran görüntüsü, yazılım, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazılım içeren bir resim

Açıklama otomatik olarak oluşturuldu

If the table is in use, Admin cannot remove the table.

Changing username is easy:

You just enter your new username, and it will change immediately.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldumetin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

After signing out you can use your new username to login.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

If you try to change your username that something already in used, program will not allow to do it.

If you say no, the previous menu will be open.

If you say yes and write an unused username, it will work.

metin, ekran görüntüsü, yazı tipi, menü içeren bir resim

Açıklama otomatik olarak oluşturulduIt is different for change password. It requires to check old password before changing it for safety.

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

If you say no:

If you say yes:

# Employee Interface:

-Take Orders

metin, ekran görüntüsü, yazılım, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

If the menu information is not existed, this message is shown.

metin, ekran görüntüsü, menü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Table orders stored in the table vector which is in the User class. Stored Information can be reused until the user exit the program.

-Display all tables:

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

Employee can see all the information about tables, their order, availability.

metin, ekran görüntüsü, yazı tipi, tasarım içeren bir resim

Açıklama otomatik olarak oluşturuldu-Calculate Bill

According to user’s choice, the function can calculate the bill for the table they choose.

# EXIT

metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

After exiting the programme, the changes in the menu and user information will be updated.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

In the figure sultan added,

Alp changed to sultanG after exiting the programme.

# 5. Conclusion:

This project provides a restaurant management system which allows users to serve, keep tracking the customers' interactions and streamline the operational administration. We have implemented these by using C++ language and following object-oriented principles. Also, the project process is strengthened our understanding in object-oriented programming.

## 5.1 Review & Comment:

This restaurant management system project almost archives all the goals outlined in the project proposal, for instances: creating menu, managing tables, placing and tracking orders, bill calculation for each table, displaying menu and orders, presenting bills and clearing tables.

Addition to these, we have added account management for every user to sign up and login, they can change their usernames and passwords, also admins can add or remove tables and menu items.

However, we did not make an option for changing price for foods. We thought that it would be unnecessary to add this function.

There might be more functionalities beyond these, but we intentionally kept the program simple and focused on fundamental operations essential for a restaurant.

## 5.2 Suggestions:

To enhance the program, it would be great to implement an interface or a database for data tracking or creating a customer section for them to use this program. We have expanded our knowledge and insights beside the C++ language, such as SQL and QT, which might enhance the program much more, if we would have use them.