Jaypee Institute of Information Technology,Noida

Data Structures & Algorithm

A Mini Project on

Restaurant Management System

Submitted by -

		Submitted to -
Pranati Tiwari	9919102056	Dr. Shruti Jaiswal
Mridnal Garg	9919102032	
Pranjal Shukla	9919102060	
Aryan Raj	991910205211	

ABSTRACT

This report documents the process of designing, developing and testing a software system to be used in a restaurant; usually given the name restaurant management system.

Currently, many restaurants use a paper-based system to communicate between the restaurant and kitchen which can be shown to be one of the least efficient approaches.Even though this approach is implemented in successful profitable restaurants, there are several problems such as Miscommunication caused by handwriting, Unmanageable order logging, Inefficient restaurant-kitchen communication & Difficult order tracking and time management which could be seen as reducing the restaurant's efficiency.

The restaurant management system is a web based application which enables easier communication between all teams within a restaurant by minimising the probability of human errors and Increasing the efficiency of the restaurant thereby increasing the profit.

As we don't have customers in restaurants, we have guests therefore we need to take care of everything to provide them with the best possible experience, and for that if they get to order and update their orders while without personally dealing with waiters, this will surely provide them with the better and personal space oriented experience.

<u>INTRODUCTION</u>

Restaurant Management System is a web-based application. This system is developed to automate day to day activity of a restaurant.

Restaurant is a kind of business that serves people all over world with ready-made food. This system is developed to provide service facility to restaurant and also to the customer. This restaurant management system can be used by employees in a restaurant to handle the clients, their orders and can help them easily place orders and calculate bills of the total order. The services that are provided is food ordering and bill calculation by the customer through the system online, customer information management, menu information management and report.

Main objective builds the system this is to provide easy ordering service by online to the customer which makes it time efficient for the customers as it reduces the waiting time after placing the order because customer can use this app to place their desired order before reaching the restaurant. Each menu item has a name and price.

With this system online, order management will become easier and systematic to replace traditional system where are still using paper.

SYSTEM SPECIFICATIONS

Minimum Hardware Requirements: -

- Pentium-IV(Processor).
- 256 MB Ram
- 512 KB Cache Memory
- Hard disk 8 GB

Implementations: -

- **Programming language:** C++
- **Data Structure used:** Arrays
- File Handling

DATASET DESCRIPTIONS

The dataset for this project is going to be a full-fledged menu display in the customer's section and the database of the guests in the restaurants with the details of the items they have ordered in that restaurant. The orders will be served accordingly,

Menu description is:

Serial No.	Item	Price
1	Paneer Tikka	350
2	Dal Makhani	290
3	Shahi Paneer	320
4	Veg Pulao	250
5	Hakka Noodles	200
6	Pasta	290
7	Butter Naan	65
8	Tandoori Roti	40
9	Missi Roti	50

SOURCE CODE OF RESTAURANT MANAGEMENT SYSTEM

```cpp
#include<iostream>
#include<fstream>
#include<string.h>
#include<conio.h>
#include<cstdio>
#include<stdlib.h>
using namespace std;

struct items{
    char dish[50];
    int key;
    float price;
}r;

class restaurant
{
public:
    void create();  //Add a dish to the menu
    void query();   //Show details at a key value
    void display(); //display menu
    void update();  //update menu items
    void delet();   //delete menu items
    void admin();   //admin portal
};
```

```cpp
class customer:public restaurant
{
public:
    void order();      //Customer menu to order
    void showbill();   //display bill information
    void dish_menu();  // prints Food menu
};

void restaurant::create()
{
    char a;
    int k;
    fstream obj;
    top:
    do {
        obj.open("Item List.txt",ios::in|ios::binary);
        cout<<"Enter the dish key: ";
        cin>>k;
        while(obj.read((char*)&r,sizeof(r)))
            {
                if(r.key==k)
                {
                    cout<<"Key is already exist..."<<endl;
                    obj.close();
                goto top;
                }
            }
        obj.close();
        obj.open("Item List.txt",ios::app|ios::binary);
```

```cpp
            r.key=k;
            cin.ignore();
            cout<<"ENTER THE DISH NAME: ";
            gets(r.dish);
            cout<<"ENTER THE DISH PRICE: ";
            cin>>r.price;
            obj.write((char*)&r,sizeof(r));
            cout<<"Do you want to add any other dish?[y/n]:";
            cin>>a;
            obj.close();
        }
        while(a!='n');
}

void restaurant::display()
{
    int c=0;
    fstream obj;
    obj.open("Item List.txt",ios::in|ios::binary);
    cout<<"\t\t\t\t\t|||||||||||||||||||||||FOOD MENU|||||||||||||||||||||"<<endl;
    cout<<"\t\t\t\t\t Key\t\tDISH NAME\t\tPRICE"<<endl;
    while(obj.read((char*)&r,sizeof(r)))
    {
        cout<<"\t\t\t\t\t "<<r.key<<"\t\t"<<r.dish<<"\t\t\tRs."<<r.price<<endl;
        c++;
    }
    if(c==0)
    {
```

```cpp
            cout<<"List is empty"<<endl;
        }
    cout<<"\t\t\t\t\t||||||||||||||||||||||||||||||||||||||||||||"<<endl;
    obj.close();
}


void restaurant::query()
{
    int a,c=0;
    fstream obj;
    obj.open("Item List.txt",ios::in);
    cout<<"Enter the dish key: ";
    cin>>a;
    while(obj.read((char*)&r,sizeof(r)))
    {
        if(r.key==a)
        {
         cout<<"\t"<<r.key<<"\t\t"<<r.dish<<"\t\t\tRs."<<r.price<<endl;
         c++;
        }
    }
    if(c==0)
    {
        cout<<"Key not found..."<<endl;
    }
    obj.close();
}

void restaurant::update()
```

```cpp
{
    int a,p,c=0;
    fstream obj;
    obj.open("Item List.txt",ios::in|ios::out|ios::binary);
    display();
    cout<<"\nENTER DETAILS TO UPDATE";
    cout<<"\nEnter the dish key: ";
    cin>>a;
    obj.seekg(0);
    while(obj.read((char*)&r,sizeof(r)))
    {
        if(r.key==a)
        {
            cout<<"\nRecord to be updated: "<<endl;
            cout<<"\nKEY: "<<r.key;
            cout<<"\nDISH NAME: "<<r.dish;
            cout<<"\nPRICE: Rs."<<r.price<<endl;
            p = obj.tellg()-(sizeof(r));
            obj.seekp(p);
            cout<<"\nENTER THE DISH KEY: ";
            cin>>r.key;
            cin.ignore();
            cout<<"ENTER THE DISH NAME: ";
            gets(r.dish);
            cout<<"ENTER THE DISH PRICE: ";
            cin>>r.price;
            obj.write((char*)&r,sizeof(r));
            c++;
        }
```

```cpp
		}
	if(c==0)
	{
		cout<<"Key not found..."<<endl;
	}
	obj.close();
}

void restaurant::delet()
{
	int a,c;
	fstream obj,obj1;
	obj.open("Item List.txt",ios::in|ios::binary);
	obj1.open("temp.txt",ios::app|ios::binary);
	cout<<"Enter the dish key: ";
	cin>>a;
	while(obj.read((char*)&r,sizeof(r)))
	{
		if(r.key==a)
		{
			c++;
			cout<<"\t"<<r.key<<"\t\t"<<r.dish<<"\t\t\t"<<r.price<<endl;
			cout<<"Record is deleted"<<endl;
		}
		if(r.key!=a)
		{
			obj1.write((char*)&r,sizeof(r));
		}
```

```cpp
        }
    obj.close();
    obj1.close();
    if(c==0)
        {
            cout<<"not found"<<endl;
        }
    remove("Item List.txt");
    rename("temp.txt","Item List.txt");
}


 void restaurant::admin()
{
char a;
    do{
            cout<<"\n\n\t\t\t\t\t||||||||||||||||||||||||||||"<<endl;
            cout<<"\t\t\t\t\t||||   ADMIN MENU   ||||"<<endl;
            cout<<"\t\t\t\t\t||||||||||||||||||||||||||||"<<endl;
            cout<<"\t\t\t\t\t|                      |"<<endl;
            cout<<"\t\t\t\t\t|  1) ADD DISH         |"<<endl;
            cout<<"\t\t\t\t\t|  2) Display          |"<<endl;
            cout<<"\t\t\t\t\t|  3) QUERY            |"<<endl;
            cout<<"\t\t\t\t\t|  4) UPDATE           |"<<endl;
            cout<<"\t\t\t\t\t|  5) DELETE           |"<<endl;
            cout<<"\t\t\t\t\t|  0) EXIT TO MAIN MANU |"<<endl;
            cout<<"\t\t\t\t\t|                      |"<<endl;
            cout<<"\t\t\t\t\t||||||||||||||||||||||||||||"<<endl;
            cout<<"\n\t\t\t\t\t\tSelect an option: ";
            cin>>a;
```

```
                    switch(a)
                    {
                    case '0':
                        system("CLS");
                        break;
                    case '1':
                        system("CLS");
                        create();
                        break;
                    case '2':
                        system("CLS");
                        display();
                        break;
                    case '3':
                        system("CLS");
                        query();
                        break;
                    case '4':
                        system("CLS");
                        update();
                        break;
                    case '5':
                        system("CLS");
                        delet();
                        break;
                    default: system("CLS");
                    }
            }while(a!='0');
    }
```

```cpp
struct {
char d[50];
float p,amount;
int qty;
}rd;

void customer::order()
{
    int a,c=0;
     char ch;
     float total=0;
    fstream obj,obj1;
    dish_menu();
    obj1.open("bill.txt",ios::out|ios::binary);
     do{
    obj.open("Item List.txt",ios::in|ios::binary);
    cout<<"Enter the dish key:";
    cin>>a;
    while(obj.read((char*)&r,sizeof(r)))
    {
        if(r.key==a)
        {
        c++;
        cout<<"Enter the quantity: ";
        cin>>rd.qty;
        rd.amount=rd.qty*r.price;

cout<<"\t"<<r.dish<<"\t\tRs."<<r.price<<"*"<<rd.qty<<"\t\tRs."<<rd.amoun
t<<endl;
```

```cpp
            strcpy(rd.d,r.dish);
            rd.p=r.price;
            obj1.write((char*)&rd,sizeof(rd));
            total=total+rd.amount;
            }
          }
        if(c==0)
       {
         cout<<"Not found"<<endl;
       }

       cout<<"Do you want to order more [y/n]";
       cin>>ch;
       obj.close();
        }
        while(ch!='n');
       obj1.close();
       cout<<endl;
       system("CLS");
       cout<<"|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||"<<endl;
        cout<<"|||||||||||||||||||||||||||||||||||||||||||||||||||||RESTAURANT  BILL
||||||||||||||||||||||||||||||||||||||||||"<<endl;
       cout<<"|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||"<<endl;
       showbill();
       cout<<"|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||"<<endl;
       cout<<"|||\t\t\t\t\t\t   TOTAL = Rs."<<total<<"\t\t\t\t\t   |||"<<endl;
       cout<<"|||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||"<<endl;
}
```

```cpp
void customer::showbill(){
    fstream obj;
    obj.open("bill.txt",ios::in|ios::binary);
    cout<<"\t\t\tDISH NAME\t\tPRICE\t\tQTY\t\tAMOUNT"<<endl;
    while(obj.read((char*)&rd,sizeof(rd)))
    {

cout<<"\t\t\t"<<rd.d<<"\t\t\tRs."<<rd.p<<"\t\t"<<rd.qty<<"\t\tRs."<<rd.amount<<endl;    }
    obj.close();
}


void customer::dish_menu()
{
    fstream obj;
    obj.open("Item List.txt",ios::in|ios::binary);
    cout<<"\t\t|||||||||||||FOOD MENU|||||||||||||";
    cout<<"\n\t\t|| Key\tDISH NAME\tPRICE  ||"<<endl;
    while(obj.read((char*)&r,sizeof(r)))
    {
        cout<<"\t\t|| "<<r.key<<")\t"<<r.dish<<"\t\tRs."<<r.price<<" ||"<<endl;
    }
    cout<<"\t\t||||||||||||||||||||||||||||||||||||"<<endl<<endl;

    obj.close();
}

void intro(){
    cout<<"\t\t\t\t\t";
```

```cpp
        cout<<"|||||||||||||||||||||||||||||||"<<endl;
        cout<<"\t\t\t\t\t";
        cout<<"|| RESTAURANT MANAGEMENT SYSTEM ||"<<endl;
        cout<<"\t\t\t\t\t";
        cout<<"|||||||||||||||||||||||||||||||||||"<<endl;
        cout<<"\n\t\t\t\t\t";
        cout<<"Project Submitted To:"<<endl;
        cout<<"\t\t\t\t\t";
        cout<<"Dr.Shruti Jaiswal"<<endl;
        cout<<"\n\t\t\t\t\t";
        cout<<"Project Implemented By:"<<endl;
        cout<<"\t\t\t\t\t";
        cout<<"Mridnal Garg  (9919102032)"<<endl;
        cout<<"\t\t\t\t\t";
        cout<<"Pranati Tiwari(9919102056)"<<endl;
        cout<<"\t\t\t\t\t";
        cout<<"Aryan Verma   (9919102052)"<<endl;
        cout<<"\t\t\t\t\t";
        cout<<"Pranjal Shukla(9919102060)"<<endl<<endl;

        system("pause");
        system("cls");
}

int main()
{
    intro();
    char a;
    customer obj;
```

```cpp
    do
    {
        cout<<"\n\n\n\t\t\t\t\t|||||||||||||||||||||"<<endl;
        cout<<"\t\t\t\t\t|                 |"<<endl;
        cout<<"\t\t\t\t\t|   1)Admin       |"<<endl;
        cout<<"\t\t\t\t\t|   2)customer    |"<<endl;
        cout<<"\t\t\t\t\t|   0)Exit        |"<<endl;
        cout<<"\t\t\t\t\t|                 |"<<endl;
        cout<<"\t\t\t\t\t|||||||||||||||||||||"<<endl;
        cout<<"\n\t\t\t\t\tselect an option: ";
        cin>>a;
        switch(a)
        {
        case '0':
            break;
        case '1':
                system("CLS");
                obj.admin();
                break;
        case '2':
                system("CLS");
                obj.order();
                break;
        }
    }while(a!='0');
}
```
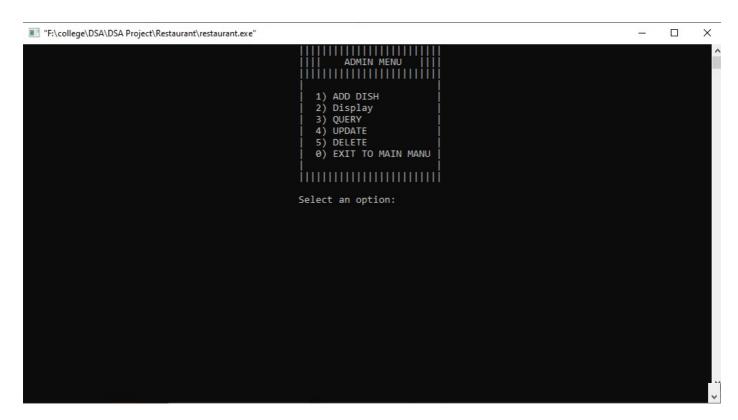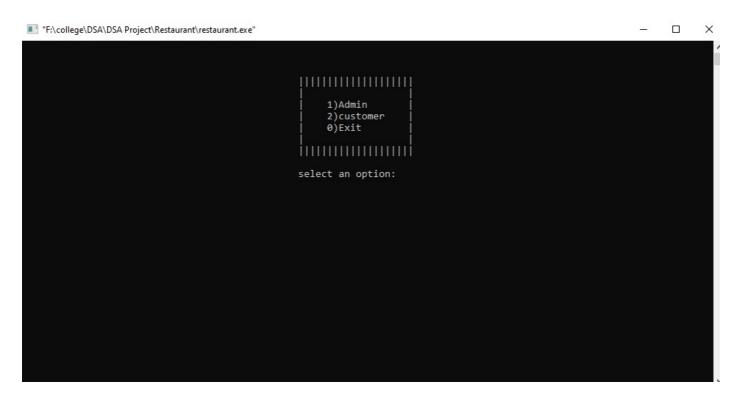
OUTPUT OF RESTAURANT MANAGEMENT SYSTEM.

			ADMIN MENU			
1) ADD DISH
2) Display
3) QUERY
4) UPDATE
5) DELETE
0) EXIT TO MAIN MANU

Select an option:

Fig. Home Page

|||||||||||||||||||
| |
| 1)Admin |
| 2)customer |
| 0)Exit |
| |
|||||||||||||||||||
select an option:

Fig. Main Menu

```
Enter the dish key: 1
ENTER THE DISH NAME: Paneer Tikka
ENTER THE DISH PRICE: 350
Do you want to add any other dish?[y/n]:y
Enter the dish key: 2
ENTER THE DISH NAME: Dal Makhni
ENTER THE DISH PRICE: 290
Do you want to add any other dish?[y/n]:y
Enter the dish key: 3
ENTER THE DISH NAME: Shahi Paneer
ENTER THE DISH PRICE: 320
Do you want to add any other dish?[y/n]:y
Enter the dish key: 4
ENTER THE DISH NAME: Veg Pulao
ENTER THE DISH PRICE: 250
Do you want to add any other dish?[y/n]:y
Enter the dish key: 5
ENTER THE DISH NAME: Hakka Noodles
ENTER THE DISH PRICE: 200
Do you want to add any other dish?[y/n]:y
Enter the dish key: 6
ENTER THE DISH NAME: Pasta
ENTER THE DISH PRICE: 290
Do you want to add any other dish?[y/n]:y
Enter the dish key: 7
ENTER THE DISH NAME: Butter Naan
ENTER THE DISH PRICE: 65
Do you want to add any other dish?[y/n]:y
```
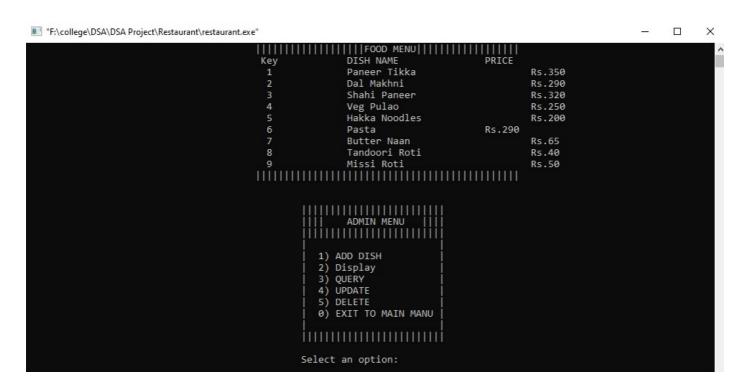
Fig. Entering Dishes in the menu

```
|||||||||||||||||||FOOD MENU||||||||||||||||||||
        Key           DISH NAME              PRICE
        1             Paneer Tikka                    Rs.350
        2             Dal Makhni                      Rs.290
        3             Shahi Paneer                    Rs.320
        4             Veg Pulao                       Rs.250
        5             Hakka Noodles                   Rs.200
        6             Pasta               Rs.290
        7             Butter Naan                     Rs.65
        8             Tandoori Roti                   Rs.40
        9             Missi Roti                      Rs.50
        |||||||||||||||||||||||||||||||||||||||||

                |||||||||||||||||||||||||
                ||||    ADMIN MENU    ||||
                |||||||||||||||||||||||||
                |                       |
                |  1) ADD DISH          |
                |  2) Display           |
                |  3) QUERY             |
                |  4) UPDATE            |
                |  5) DELETE            |
                |  0) EXIT TO MAIN MANU |
                |                       |
                |||||||||||||||||||||||||

                Select an option:
```
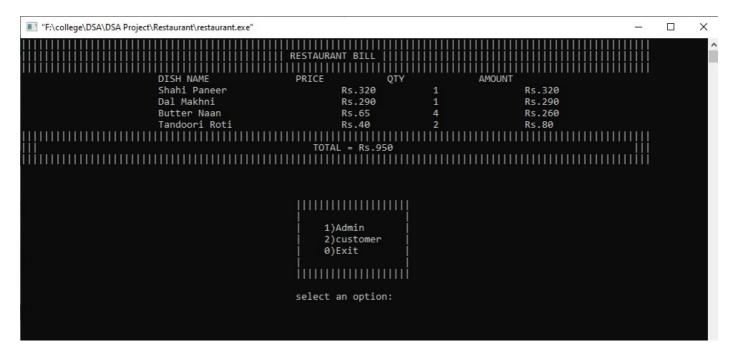
Fig. Food Menu

Fig. Calculated Bill after the final order placed

CONCLUSION

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

- ➢ The Restaurant Management System of the entire system improves efficiency.
- ➢ It provides has a friendly graphical user interface which proves to be better when compared to the existing system.
- ➢ It effectively overcomes the delay in communications.
- ➢ The System has adequate scope for modification in future if it is necessary.