**Project Report**

**On**

**HOTEL Management WITH**

**COVID PROTOCOLS**

**Bachelor of Technology (Information Technology)**

**(2021)**

**JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY**

**(NOIDA-128)**

**INDEX**

|  |  |
| --- | --- |
| **S.No.** | **Topic** |
| 1. | Introduction |
| 2. | Class Diagram |
| 3. | Code |
| 4. | Results And Discussion |
| 5. | Conclusion |

**INTRODUCTION**

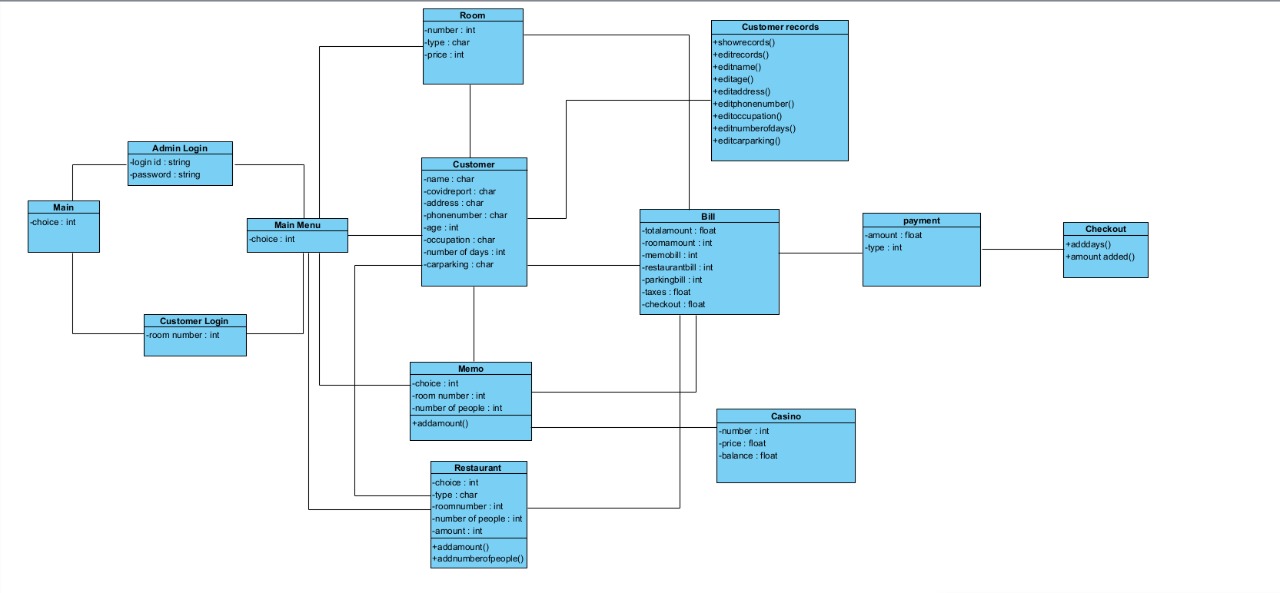
To create a modernized hotel management system which can adapt to the new covid and living norms. The project consists of proper hotel management along with modernized billing system. With proper use of OOPs concepts we aim to create a project which could strike the cord directly between the admin and customer.

The purpose of the project was to create an efficient records management system where the customer records are handled properly. Here the customer records are recorded accordingly with his check in. The rooms have been divided into various sections in accordance with new guidelines and laws. The customer records can be searched along with his room along with his personal information. Records can be modified easily to reduce the risk of mismanagement. We also introduced the concept of an itemized billing system where the total amount of his room charge along with the charges per facilities which is used by the customer during his stay will be shown with proper checkout. It also introduces the concept of in-built facility management system to provide various other facilities provided by the hotel to the customer. The modernized billing system works accordingly with modern taxing system. This program is designed in such a way that it executes perfectly along with customer demands.

**OBJECTIVE:**

To create a modernized Hotel Management System aims to make simpler a staff's interaction with the various modules of the Hotel and ease the process of acquiring information and providing services accordingly with new living norms. The system can be accessed by the admin and customers but the highest priority given to admin who is allocated a login id and password to access. It will also allow cutting the operational costs of the hotel. The system also aims to create a new and modernized billing system adapting itself to the covid norms.

* This project aims to show how efficiently a modernized hotel management can adapt to the new norms that have been created due to pandemic. This system can make the users work less cumbersome, more efficient.
* This system reduces the liability of handling the records work manually, reduces workload.
* We have tried to create an efficient program which aims to reduce redundancy, improves code reusability.
* Proper handling of system records which works efficiently and with sheer precision.
* It aims to reduce risk load and aims to create a user-friendly environment in which user can handle the system easily with accuracy.
* This program uses the various oops concepts to emulate the aims of programmer which is to create a program which reduces the risk of negligence, to make the program precise and more understandable.
* It also reduces the code maintenance and provide better security to the code and increases the portability.

**Class Diagram:**

**SOURCE CODE:**

#include <iostream>

#include <fstream>

#include <stdio.h>

#include <stdlib.h>

#include <cstring>

#include <ctime>

#include <iomanip>

#include <conio.h>

#include <dos.h>

using namespace std;

***//CLASS HOTEL***

class HOTEL

{

private:

    int room\_no;

    int age;

    float total = 0;

    float tax = 0, dis = 0, amt = 0;

    long mem = 0;

    char car;

    char car\_park[10];

    long cp, pay1 = 0, pay2 = 0;

    long ad\_pay = 0;

    char occ[20];

    char name[30],surname[20];

    char address[50];

    char phone[15];

    int days;

    long cost = 0;

    char rtype[30];

    char reptype[20];

    long pay = 0;

    void theatre(int);

    void swim(int);

    void gym(int);

    void breakfast(int);

    void lunch(int);

    void dinner(int);

public:

    void covid();

    void normal();

    void memo();

    void casino();

    void add\_day(int);

    void main\_menu();

    void add();

    void display();

    void rooms();

    void edit();

    int check(int);

    void modify();

    void bill();

    void restaurant();

    void modify\_name(int);

    void modify\_address(int);

    void modify\_phone(int);

    void modify\_age(int);

    void modify\_occ(int);

    void modify\_days(int);

    void rules();

    void admlog();

    void custlog();

} h;

***//FUNCTION MAIN MENU***

void HOTEL::main\_menu()

{

    int choice;

    char z;

    while (choice != 8)

    {

        system("cls");

        cout << "\n\t\t\t +------------------------+";

        cout << "\n\t\t\t |   Hotel Raddison Blu   |";

        cout << "\n\t\t\t +------------------------+";

        cout << "\n-> Main Menu";

        cout << "\n\n1.  Book A Room";

        cout << "\n2.  Customer Information";

        cout << "\n3.  Rooms Allotted";

        cout << "\n4.  Memo";

        cout << "\n5.  Edit Customer Details";

        cout << "\n6.  Order Food from Restaurant";

        cout << "\n7.  Generate Bill";

        cout << "\n8.  Exit";

        cout << "\n\  Enter Your Choice: ";

        cin >> choice;

        switch (choice)

        {

        case 1:

            add();

            break;

        case 2:

            display();

            break;

        case 3:

            rooms();

            break;

        case 4:

            memo();

            break;

        case 5:

            edit();

            break;

        case 6:

            restaurant();

            break;

        case 7:

            bill();

            break;

        case 8: cout << "\nThank You For Visiting!\n\n";

             return;

             break;

        default:

        {

            cout << "\n Wrong choice.";

            cout << "\n Press any key to continue.";

            getch();

        }

        }

    }

}

***//FUNCTION FOR COVID ROOMS***

void HOTEL::covid()

{

    ofstream fout("Record.DAT", ios::app | ios::binary);

    int r, flag;

    cout << "\n\n Enter Rooms from Room No. 101 to 150. ";

    cout << "\n\n ENTER CUSTOMER DETAILS";

    cout << "\n ----------------------";

    cout << "\n Room Number: ";

    cin >> r;

    flag = check(r);

    if (flag == 1)

        cout << "\n Sorry, Room is already booked.\n";

    else

    {

        if (r > 0 && r < 101 || r > 150)

        {

            cout << "\n Sorry, Room does not exist.\n";

            cout << "\n You Must Select Rooms From Room No. 101 to 150. ";

        }

        else

        {

            room\_no = r;

            cout<<" First Name: ";

           cin>>name;

           cout<<" Last Name: ";

           cin>>surname;

            cout << " Address: ";

            cin >> address;

            cout << " Phone Number: ";

            cin >> phone;

            cout << " Age: ";

            cin >> age;

            fflush(stdin);

            cout << " Occupation: ";

            cin >> occ;

            cout << " Number of Days: ";

            cin >> days;

            fflush(stdin);

            cout << " Want Car Parking ? {y/n}:  ";

            cin >> car;

            if (car == 'y' || car == 'Y')

            {

                cp = 30 ;

                strcpy(car\_park, "YES");

            }

            else

            {

                cp = 0;

                strcpy(car\_park, "NO");

            }

            strcpy(reptype, "Positive");

            if (room\_no >= 101 && room\_no <= 150)

            {

                strcpy(rtype, "Covid Room");

                cost = days \* 4000;

                cout << "\n Doctor will be available at 6:00 pm to 8:00 pm. ";

            }

            fout.write((char \*)this, sizeof(HOTEL));

            cout << "\n Room has been booked.";

        }

    }

    cout << "\n Press any key to continue.";

    fout.close();

    getch();

}

***//FUNCTION FOR NORMAL ROOMS***

void HOTEL::normal()

{

    ofstream fout("Record.DAT", ios::app | ios::binary);

    int r, flag;

    cout << "\n\n Enter Rooms from Room No. 1 to 100. ";

    cout << "\n\n ENTER CUSTOMER DETAILS";

    cout << "\n ----------------------";

    cout << "\n Room Number: ";

    cin >> r;

    flag = check(r);

    if (flag == 1)

        cout << "\n Sorry, Room is already booked.\n";

    else

    {

        if (r < 1 || r > 101)

        {

            cout << "\n Sorry, Room does not exist.\n";

            cout << "\n You Must Select Rooms From Room No. 1 to 100. ";

        }

        else

        {

            room\_no = r;

            cout<<" First Name: ";

           cin>>name;

           cout<<" Last Name: ";

           cin>>surname;

            cout << " Address: ";

            cin >> address;

            cout << " Phone Number: ";

            cin >> phone;

            cout << " Age: ";

            cin >> age;

            fflush(stdin);

            cout << " Occupation: ";

            cin >> occ;

            cout << " Number of Days: ";

            cin >> days;

            fflush(stdin);

            cout << " Want Car Parking ? {y/n}: ";

            cin >> car;

            if (car == 'y' || car == 'Y')

            {

                cp = 30;

                strcpy(car\_park, "YES");

            }

            else

            {

                cp = 0;

                strcpy(car\_park, "NO");

            }

            strcpy(reptype, "Negative");

            if (room\_no >= 1 && room\_no <= 50)

            {

                strcpy(rtype, "Ordinary");

                cost = days \* 2000;

            }

            else if (room\_no >= 51 && room\_no <= 80)

            {

                strcpy(rtype, "Deluxe");

                cost = days \* 5000;

            }

            else if (room\_no >= 81 && room\_no <= 100)

            {

                strcpy(rtype, "Royal");

                cost = days \* 8000;

            }

            fout.write((char \*)this, sizeof(HOTEL));

            cout << "\n Room has been booked.";

        }

        cout << "\n Press any key to continue.";

        fout.close();

        getch();

    }

}

***//FUNCTION TO BOOK A ROOM***

void HOTEL::add()

{

    system("cls");

    char c, q, m;

    int r, flag;

    cout << "\n\t\t\t +----------+------------+----------+";

    cout << "\n\t\t\t |  Rooms   |  Room Type |   Price  |";

    cout << "\n\t\t\t +----------+------------+----------+";

    cout << "\n\t\t\t |   1-50   |  Ordinary  |  Rs.2000 |";

    cout << "\n\t\t\t |  51-80   |   Deluxe   |  Rs.5000 |";

    cout << "\n\t\t\t |  81-100  |    Royal   |  Rs.8000 |";

    cout << "\n\t\t\t | 101-150  | Covid Room |  Rs.4000 |";

    cout << "\n\t\t\t +----------+------------+----------+";

    cout << "\n\t\t\t |   Car Parking Per Day is Rs.30   |";

    cout << "\n\t\t\t +----------+------------+----------+";

    cout << "\n\n Do you want to book a room? {Y/N}: ";

    cin >> c;

    if (c == 'y' || c == 'Y')

    {

        cout << "\n Do You Have Covid Report ? {Y/N}: ";

        cin >> q;

        if (q == 'Y' || q == 'y')

        {

            cout << "\n Are you Corona Positive Or Negative? {P/N}: ";

            cin >> m;

            if (m == 'p' | m == 'P')

            {

                covid();

            }

            else

            {

                normal();

            }

        }

        else if (q == 'n' || q == 'N')

        {

            cout << "\n Extra Charge will be apply for the Report...\n Are You Corona Positive or Negative? {P/N}: ";

            cin >> m;

            if (m == 'p' | m == 'P')

            {

                covid();

            }

            else

            {

                normal();

            }

        }

    }

    else

    {

        cout << "\n Press any key to continue.";

        getch();

        main\_menu();

    }

}

***//FUNCTION TO DISPLAY PARTICULAR RECORD***

void HOTEL::display()

{

    system("cls");

    fstream fin("Record.DAT", ios::in | ios::binary);

    int r, flag;

    cout << "\n Enter Room Number: ";

    cin >> r;

    flag = check(r);

    while (fin.read((char \*)this, sizeof(HOTEL)))

    {

        if (room\_no == r)

        {

            system("cls");

            cout<<"\n Customer Details";

            cout<<"\n ----------------";

            cout<<"\n\n Room Number: "<<room\_no;

            cout<<"\n Name: "<<name<<" "<<surname;

            cout<<"\n Address: "<<address;

            cout<<"\n Phone Number: "<<phone;

            cout<<"\n Age: "<<age;

            cout<<"\n Occupation :"<<occ;

            cout<<"\n Staying for "<<"'"<<days<<"'"<<" days.";

            cout<<"\n Car Parking: "<<car\_park;

            cout<<"\n Room Type: "<<rtype;

            cout<<"\n COVID Report: "<<reptype;

            cout<<"\n Total cost of stay: "<<cost;

            flag=1;

            break;

}

    }

    if (flag == 0)

        cout << "\n Room No." << r << " is Vacant.";

    cout << "\n\n Press any key to continue.";

    fin.close();

    getchar();

    getchar();

}

***//FUNCTION TO SHOW ALL RECORDS***

void HOTEL::rooms()

{

    system("cls");

    const char separator = ' ';

    const int NoWidth = 8;

    const int GuestWidth = 13;

    const int AddressWidth = 12;

    const int RoomWidth = 12;

    const int ContactNoWidth = 12;

    const int agewidth = 8;

    fstream fin("Record.DAT", ios::in | ios::binary);

    cout << "\n\t\t\t    LIST OF ROOMS ALLOTED";

    cout << "\n\t\t\t   -----------------------";

    cout << "\n\n+---------+--------------+-------------+-------------+-------------+---------+";

    cout << "\n| Room No.|     Name     |   Address   |  Room Type  | Contact No. |   Age   |";

    cout << "\n+---------+--------------+-------------+-------------+-------------+---------+";

    while (fin.read((char \*)this, sizeof(HOTEL)))

    {

        cout << "\n|" << setw(NoWidth) << setfill(separator) << room\_no << " |";

        cout << setw(GuestWidth) << setfill(separator) << name << " |";

        cout << setw(AddressWidth) << setfill(separator) << address << " |";

        cout << setw(RoomWidth) << setfill(separator) << rtype << " |";

        cout << setw(ContactNoWidth) << setfill(separator) << phone << " |";

        cout << setw(agewidth) << setfill(separator) << age << " |";

    }

    cout << "\n+---------+--------------+-------------+-------------+-------------+---------+";

    cout << "\n Press any key to continue.";

    fin.close();

    getch();

}

***//FUNCTION TO CHECK ROOM AVAILABILITY***

int HOTEL::check(int r)

{

    int flag = 0;

    fstream fin("Record.DAT", ios::in | ios::binary);

    while (fin.read((char \*)this, sizeof(HOTEL)))

    {

        if (room\_no == r)

        {

            flag = 1;

            break;

        }

        else

        {

            if (r > 150)

            {

                flag == 2;

                break;

            }

        }

    }

    fin.close();

    return (flag);

}

***//FUNCTION TO MODIFY CUSTOMER DETAILS***

void HOTEL::edit()

{

    system("cls");

    int choice, r;

    cout << "\n EDIT MENU";

    cout << "\n ---------";

    cout << "\n\n 1. Modify Customer Information.";

    cout << "\n 2. Back.";

    cout << "\n\n Enter your choice: ";

    cin >> choice;

    system("cls");

    switch (choice)

    {

    case 1:

        modify();

        break;

    default:

        cout << "\n Okay, Press any key to open main menu.";

        getch();

        main\_menu();

        break;

    }

    cout << "\n Press any key to continue.";

    getch();

}

***//FUNCTION TO MODIFY CUSTOMER DETAILS***

void HOTEL::modify()

{

    system("cls");

    int ch, r;

    cout << "\n\t +-------------+";

    cout << "\n\t | MODIFY MENU |";

    cout << "\n\t +-------------+";

    cout << "\n\n 1. Modify Name";

    cout << "\n 2. Modify Address";

    cout << "\n 3. Modify Phone Number";

    cout << "\n 4. Modify Age";

    cout << "\n 5. Modify Occupation";

    cout << "\n 6. Modify Number of Days of Stay";

    cout << "\n 7. Back to Main menu.";

    cout << "\n Enter Your Choice: ";

    cin >> ch;

    system("cls");

    cout << "\n Enter Room Number: ";

    cin >> r;

    switch (ch)

    {

    case 1:

        modify\_name(r);

        break;

    case 2:

        modify\_address(r);

        break;

    case 3:

        modify\_phone(r);

        break;

    case 4:

        modify\_age(r);

        break;

    case 5:

        modify\_occ(r);

        break;

    case 6:

        modify\_days(r);

        break;

    default:

        cout << "\n Press Any Key to Continue..";

        getchar();

        break;

    }

}

***//FUNCTION TO MODIFY CUSTOMER NAME***

void HOTEL::modify\_name(int r)

{

    long pos, flag = 0;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << "\n Enter New Name: ";

            cin >> name;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Customer Name has been modified.";

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is vacant.";

    file.close();

    getch();

}

***//FUNCTION TO MODIFY CUSTOMER ADDRESS***

void HOTEL::modify\_address(int r)

{

    long pos, flag = 0;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << "\n Enter New Address: ";

            cin >> address;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Customer Address has been modified.";

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is vacant.";

    file.close();

    getch();

}

***//FUNCTION TO MODIFY CUSTOMER PHONE NUMBER***

void HOTEL::modify\_phone(int r)

{

    long pos, flag = 0;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << "\n Enter New Phone Number: ";

            cin >> phone;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Customer Phone Number has been modified.";

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is vacant.";

    file.close();

    getch();

}

***//FUNCTION TO MODIFY CUSTOMER AGE***

void HOTEL::modify\_age(int r)

{

    long pos, flag = 0;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << "\n Enter New Age: ";

            cin >> age;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Customer Age has been modified.";

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is vacant.";

    file.close();

    getch();

}

***//FUNCTION TO MODIFY CUSTOMER OCCUPATION***

void HOTEL::modify\_occ(int r)

{

    long pos, flag = 0;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << "\n Enter New Occupation: ";

            cin >> occ;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Customer Occupation has been modified.";

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is vacant.";

    file.close();

    getch();

}

***//FUNCTION TO MODIFY CUSTOMER BOOKING DAYS***

void HOTEL::modify\_days(int r)

{

    long pos, flag = 0;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << " Enter New Number of Days of Stay: ";

            cin >> days;

            if (room\_no >= 1 && room\_no <= 50)

            {

                strcpy(rtype, "Ordinary");

                cost = days \* 2000;

            }

            else if (room\_no >= 51 && room\_no <= 80)

            {

                strcpy(rtype, "Deluxe");

                cost = days \* 5000;

            }

            else if (room\_no >= 81 && room\_no <= 100)

            {

                strcpy(rtype, "Royal");

                cost = days \* 8000;

            }

            else if (room\_no >= 101 && room\_no <= 150)

            {

                strcpy(rtype, "Covid Room");

                cost = days \* 4000;

            }

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Customer information is modified.";

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is Vacant.";

    file.close();

    getch();

}

***//FUNCTION TO BOOK MEMO***

void HOTEL::memo()

{

    system("cls");

    int r, ch, num;

    cout << "\n\t\t\t\t Programs:";

    cout << "\n\t\t\t -------------------------- ";

    cout << "\n\t\t +----------+---------------+-----------------+";

    cout << "\n\t\t |  S.No.   |    Program    |   Price/Person  |";

    cout << "\n\t\t +----------+---------------+-----------------+";

    cout << "\n\t\t |     1    |    Theatre    |      Rs.500     |";

    cout << "\n\t\t |     2    | Swimming Pool |      Rs.200     |";

    cout << "\n\t\t |     3    |      Gym      |      Rs.100     |";

    cout << "\n\t\t |     4    |     Casino    |         -       |";

    cout << "\n\t\t |     5    |   Main menu   |         -       |";

    cout << "\n\t\t +----------+---------------+-----------------+";

    cout << "\n Enter your choice: ";

    cin >> ch;

    switch (ch)

    {

    case 1:

        cout << " Enter Room Number: ";

        cin >> r;

        theatre(r);

        break;

    case 2:

        cout << " Enter Room Number: ";

        cin >> r;

        swim(r);

        break;

    case 3:

        cout << " Enter Room Number: ";

        cin >> r;

        gym(r);

        break;

    case 4:

        casino();

        break;

    default:

        cout << "\n Okayy..";

        cout << "\n Press any key to continue...!";

        getch();

        main\_menu();

        break;

    }

}

***//FUNCTION TO BOOK THEATRE***

void HOTEL::theatre(int r)

{

    int num, flag = 0;

    long pos;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << " Enter number of people: ";

            cin >> num;

            cout << "\n Theatre is Booked For " << num << " peoples.\n Hope You will Enjoy!";

            mem = 500 \* num;

            pay1 = pay1 + mem;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Amount added to the bill: Rs. " << mem;

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is Vacant.";

    getch();

    file.close();

}

***//FUNCTION TO BOOK SWIMMING POOL***

void HOTEL::swim(int r)

{

    int num, flag = 0;

    long pos;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << " Enter number of people: ";

            cin >> num;

            cout << "\n Swimming Pool is Booked For " << num << " peoples.\n Hope You will Enjoy!";

            mem = 200 \* num;

            pay1 = pay1 + mem;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Amount added to the bill: Rs. " << mem;

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is vacant.";

    file.close();

    getch();

}

***//FUNCTION TO BOOK GYM***

void HOTEL::gym(int r)

{

    int num, flag = 0;

    long pos;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << " Enter number of people: ";

            cin >> num;

            cout << "\n Gym is Booked For " << num << " peoples.\n Hope You will Enjoy!";

            mem = 100 \* num;

            pay1 = pay1 + mem;

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Amount added to the bill: Rs. " << mem;

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is Vacant.";

    file.close();

    getch();

}

***//FUNCTION OF CASINO RULES***

void HOTEL::rules()

{

    system("cls");

    cout << "\n";

    cout << "\t\tRULES OF THE GAME\n";

    cout << "1. Choose any number between 1 to 10\n";

    cout << "2. If you win you will get 10 times of money you bet\n";

    cout << "3. If you bet on wrong number you will lose your betting amount\n";

    cout << "...............................................................................";

}

***//FUNCTION TO PLAY CASINO***

void HOTEL::casino()

{

    time\_t timer = time(NULL);

    int amount;

    int bettingAmount;

    int guess;

    int dice;

    char choice;

    system("cls");

    cout << "\n\t\t\t\*.\*.\*.\*.\*.\*.\*.\*.\*.\*.\*";

    cout << "\n\t\t\t\*  Casino Carnival  \*";

    cout << "\n\t\t\t\*.\*.\*.\*.\*.\*.\*.\*.\*.\*.\*";

    cout << "\n\n\t\t  Time: " << ctime(&timer);

    cout << "\n\nEnter Total amount to be Deposit to play game: Rs.";

    cin >> amount;

    do

    {

        system("cls");

        rules();

        cout << "\n\nYour current balance is Rs." << amount << "\n";

        do

        {

            cout << "\nYou enter money to bet: Rs.";

            cin >> bettingAmount;

            if (bettingAmount > amount)

                cout << "\nYour betting amount is more than your current balance\n"

                     << "Please Re-enter amount\n ";

        } while (bettingAmount > amount);

        do

        {

            cout << "\nGuess your number to bet between 1 to 10: ";

            cin >> guess;

            if (guess <= 0 || guess > 10)

                cout << "Please check the number!! should be between 1 to 10\n"

                     << "Re-enter data\n ";

        } while (guess <= 0 || guess > 10);

        dice = rand() % 10 + 1;

        if (dice == guess)

        {

            cout << "\nGood Luck!! You won Rs." << bettingAmount \* 10;

            amount = amount + bettingAmount \* 10;

        }

        else

        {

            cout << "\nBad Luck this time !! You lost Rs. " << bettingAmount << "\n";

            amount = amount - bettingAmount;

        }

        cout << "\nThe winning number was : " << dice << "\n";

        cout << "\nYou have Rs. " << amount << "\n";

        if (amount == 0)

        {

            cout << "You have no money to play ";

            break;

        }

        cout << "\nDo you want to play again (y/n)? ";

        cin >> choice;

    }

while (choice == 'Y' || choice == 'y');

    cout << "\nThanks for playing game. Your balance amount is Rs. " << amount;

}

***//FUNCTION TO ORDER FOOD***

void HOTEL::restaurant()

{

    system("cls");

    int r, ch, num;

    cout << "\n\t\t\t\tRESTAURANT MENU:";

    cout << "\n\t\t\t ------------------------------- ";

    cout << "\n\t\t +----------+------------+-----------------+";

    cout << "\n\t\t |  Order   |    Type    |   Price/Person  |";

    cout << "\n\t\t +----------+------------+-----------------+";

    cout << "\n\t\t |     1    |  Breakfast |      Rs.500     |";

    cout << "\n\t\t |     2    |   Lunch    |      Rs.1000    |";

    cout << "\n\t\t |     3    |   Dinner   |      Rs.1200    |";

    cout << "\n\t\t |     4    |    Back    |         -       |";

    cout << "\n\t\t +----------+------------+-----------------+";

    cout << "\n\n Enter your choice: ";

    cin >> ch;

        switch (ch)

    {

    case 1:

        cout << " Enter Room Number: ";

        cin >> r;

        breakfast(r);

        break;

    case 2:

        cout << " Enter Room Number: ";

        cin >> r;

        lunch(r);

        break;

    case 3:

        cout << " Enter Room Number: ";

        cin >> r;

        dinner(r);

        break;

    default:

        cout << "\n Okay... ";

        cout << "\n\n Press any key to continue.";

        getch();

        main\_menu();

    }

}

***//FUNCTION TO ORDER BREAKFAST***

void HOTEL::breakfast(int r)

{

    int num, flag = 0;

    long pos;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << " Enter number of people: ";

            cin >> num;

            pay = 500 \* num;

            pay2 = pay2 + pay;

            cout << "\n Table is Booked For " << num << " peoples.\n ";

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Amount added to the bill: Rs. " << pay;

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is Vacant.";

    getch();

    file.close();

}

***//FUNCTION TO ORDER LUNCH***

void HOTEL::lunch(int r)

{

    int num, flag = 0;

    long pos;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << " Enter number of people: ";

            cin >> num;

            pay = 1000 \* num;

            pay2 = pay2 + pay;

            cout << "\n Table is Booked For " << num << " peoples.\n ";

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Amount added to the bill: Rs. " << pay;

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is vacant.";

    getch();

    file.close();

}

***//FUNCTION TO ORDER DINNER***

void HOTEL::dinner(int r)

{

    int num, flag = 0;

    long pos;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            cout << " Enter number of people: ";

            cin >> num;

            pay = 1200 \* num;

            pay2 = pay2 + pay;

            cout << "\n Table is Booked For " << num << " peoples.\n ";

            file.seekg(pos);

            file.write((char \*)this, sizeof(HOTEL));

            cout << "\n Amount added to the bill: Rs. " << pay;

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Sorry, Room is Vacant.";

    getch();

    file.close();

}

***//FUNCTION TO ADD MORE DAYS***

void HOTEL::add\_day(int r)

{

    int add\_days, flag = 0;

    long pos;

    cout << " Enter number of days You Want to Add: ";

    cin >> add\_days;

    fstream file("Record.DAT", ios::in | ios::out | ios::binary);

    while (!file.eof())

    {

        pos = file.tellg();

        file.read((char \*)this, sizeof(HOTEL));

        if (room\_no == r)

        {

            days = add\_days + days;

            if (room\_no >= 1 && room\_no <= 50)

            {

                ad\_pay = add\_days \* 2000;

                cost = cost + ad\_pay;

            }

            else if (room\_no >= 51 && room\_no <= 80)

            {

                ad\_pay = add\_days \* 5000;

                cost = cost + ad\_pay;

            }

            else if (room\_no >= 81 && room\_no <= 100)

            {

                ad\_pay = add\_days \* 8000;

                cost = cost + ad\_pay;

            }

            else if (room\_no >= 101 && room\_no <= 150)

            {

                ad\_pay = add\_days \* 4000;

                cost = cost + ad\_pay;

            }

        }

        file.seekg(pos);

        file.write((char \*)this, sizeof(HOTEL));

        cout << "\n Amount added to the bill: Rs. " << ad\_pay;

        flag = 1;

        break;

    }

    if (flag == 0)

        cout << "\n Sorry, Room is Vacant.";

    getchar();

    file.close();

}

***//FUNCTION TO PAY BILL***

void HOTEL::bill()

{

    int r,c, flag = 0;

    int a;

    char pm[10];

    time\_t timer = time(NULL);

    char ch;

    cout << "\n Enter Room Number: ";

    cin >> r;

    flag = check(r);

    system("cls");

    ifstream fin("Record.DAT", ios::in | ios::binary);

    ofstream fout("temp.DAT", ios::out | ios::binary);

    while (fin.read((char \*)this, sizeof(HOTEL)))

    {

        if (room\_no == r)

        {

c=cp\*days;

            cout<<"\n Name: "<<name <<" "<<surname;

            cout << "\n You Stay for "<< "'" << days << "'"<< " days";

            cout << "\n Room Bill: Rs." << cost;

            cout << "\n Memo Bill: Rs." << pay1;

            cout << "\n Restaurant Bill: Rs." << pay2;

            cout << "\n Car parking: Rs." << c;

            cout << "\n Your bill is: Rs." << cost + pay1 + pay2 + c;

            tax = (cost + pay1 + pay2) \* 0.18;

            total = tax + cost + pay1 + pay2 + c;

            cout << "\n\t Taxes (18%): Rs." << tax;

            cout << "\n\t Total Bill is: Rs." << total;

            cout << "\n\n Do you want to check out this customer(y/n): ";

            cin >> ch;

            if (ch == 'n' || ch == 'N')

            {

                add\_day(r);

                fout.write((char \*)this, sizeof(HOTEL));

            }

            else

            {

                cout << "\n\n\t\t\t\t Payment Method:";

                cout << "\n\t\t +----------+----------------+-------------+";

                cout << "\n\t\t |   S.No.  |      Type      |   Discount  |";

                cout << "\n\t\t +----------+----------------+-------------+";

                cout << "\n\t\t |     1    |       Cash     |      0%     |";

                cout << "\n\t\t |     2    |   Credit Card  |      10%    |";

                cout << "\n\t\t |     3    | Online Banking |      5%     |";

                cout << "\n\t\t +----------+----------------+-------------+";

                cout << "\n Enter Payment Method: ";

                cin >> a;

                if (a == 1)

                {

                    strcpy(pm, "Cash");

                    amt = total;

                }

                else if (a == 2)

                {

                    strcpy(pm, "Credit Card");

                    dis = total \* 0.1;

                    amt = total - dis;

                }

                else

                {

                    strcpy(pm, "Online Banking");

                    dis = total \* 0.05;

                    amt = total - dis;

                }

                system("cls");

                cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

                cout << "\t\t\t \*        HOTEL RADDISON      \*";

                cout << "\n................................................................................";

                cout << "\t\t\t\tFaridabad-121001";

                cout << "\n\t\t\t\t  Tel-12930932\n";

                cout << "-------------------------------------------------------------------------------";

                cout << "\n\t\t\t\t   Invoice:\n";

                cout << "\n\t\t Time: " << ctime(&timer);

                cout<<"\n Name: "<<name <<" "<<surname;

                cout << "\n Phone Number:    " << phone;

                cout << "\n Room Type:       " << rtype;

                cout << "\n You Stay for "<< "'" << days << "'"<< " days";

                cout << "\n Room Bill:       Rs." << cost;

                cout << "\n Memo Bill:       Rs." << pay1;

                cout << "\n Restaurant Bill: Rs." << pay2;

                cout << "\n Car parking:     Rs." << c;

                cout << "\n Your bill is:    Rs." << cost + pay1 + pay2 + c;

                cout << "\n..........................................................";

                cout << "\n\t Taxes{18%}: Rs." << tax;

                cout << "\n\t Total Bill is: Rs." << total;

                cout << "\n\n Payment Method: " << pm;

                cout << "\n\n You Have To Pay Rs." << amt;

                getch();

                cout << "\n Customer Checked Out.";

                flag = 1;

            }

        }

        else

            fout.write((char \*)this, sizeof(HOTEL));

    }

    fin.close();

    fout.close();

    if (flag == 0)

        cout << "\n Sorry, Room is Vacant.";

    else

    {

        remove("Record.dat");

        rename("temp.dat", "Record.dat");

    }

    getch();

}

***//FUNCTION FOR ADMIN LOGIN***

void HOTEL::admlog()

{

    int u, p;

    system("cls");

    cout << "\n\t\t\t +-----------------------+";

    cout << "\n\t\t\t | ADMINISTRATION LOGIN  |";

    cout << "\n\t\t\t +-----------------------+";

    cout << "\n\n Enter the login details";

    for (int i = 1; i < 3; i++)

    {

        cout << "\n\n Enter Admin ID: ";

        cin >> u;

        cout << " Enter password: ";

        cin >> p;

        fflush(stdin);

        if (u == 1 && p == 2)

        {

            cout << "\n\n Access Granted..!";

            cout << "\n Press any key to continue....!!";

            getch();

            h.main\_menu();

            break;

        }

        else

        {

            cout << "\n Sorry wrong User Id and Password \n Enter the detail again:";

            cout << "\n Chance left: " << 2 - i << "\n";

        }

    }

}

***//FUNCTION FOR CUSTOMER LOGIN***

void HOTEL::custlog()

{

    int r,c, flag = 0;

    int a;

    char pm[10];

    time\_t timer = time(NULL);

    char ch;

    cout << "\n Enter Room Number: ";

    cin >> r;

    flag = check(r);

    ifstream fin("Record.DAT", ios::in | ios::binary);

    ofstream fout("temp.DAT", ios::out | ios::binary);

    while (fin.read((char \*)this, sizeof(HOTEL)))

    {

        if (room\_no == r)

        {

c=cp\*days;

            system("cls");

            cout << "\n  Details";

            cout << "\n ----------------";

            cout << "\n\n Room Number:  " << room\_no;

            cout<<"\n Name: "<<name <<" "<<surname;

            cout << "\n Address:      " << address;

            cout << "\n Phone Number: " << phone;

            cout << "\n Age:          " << age;

            cout << "\n Occupation :  " << occ;

            cout << "\n Car Parking:  " << car\_park;

            cout << "\n Room Type:    " << rtype;

            cout << "\n COVID Report: " << reptype;

            cout << "\n Staying for "<< "' " << days << " '"<< " days.";

            cout << "\n\n Total cost of stay: Rs." << cost;

            cout << "\n Room Bill:          Rs." << cost;

            cout << "\n Memo Bill:          Rs." << pay1;

            cout << "\n Restaurant Bill:    Rs." << pay2;

            cout << "\n Car parking:        Rs." << c;

            cout << "\n Your bill is:       Rs." << cost + pay1 + pay2 + c;

            tax = (cost + pay1 + pay2) \* 0.18;

            total = tax + cost + pay1 + pay2 + c;

            cout << "\n Taxes (18%):        Rs." << tax;

            cout << "\n Total Bill is:      Rs." << total;

            flag = 1;

            break;

        }

    }

    if (flag == 0)

        cout << "\n Room No." << r << " is Vacant.";

    cout << "\n Thanks For using our Portal.!";

    cout << "\n\n Press any key to continue.";

    fin.close();

    getchar();

}

***//MAIN FUNCTION OF PROGRAM***

int main()

{

    int ch;

    cout << "\n\t\tJaypee Institute of Information Technology";

    cout << "\n\t\t\t\tNoida-128";

    cout << "\n\n\t\tSDF-2 Project On Hotel Management with covid protocols.";

    cout << "\n\n\t\t->Developed By:  Rishabh, Shashwat, Kartik, Sushant\n\n";

    cout << "\n\n\t\t\tPress Any key to continue...!";

    getch();

    system("cls");

    cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

    cout << "\t\t\t \*  WELCOME TO HOTEL RADDISON  \*";

    cout << "\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

    cout << "\n\n1.  ADMIN LOGIN";

    cout << "\n2.  CUSTOMER LOGIN";

    cout << "\n3.  EXIT";

    cout << "\n\n Enter your Choice: ";

    cin >> ch;

    switch (ch)

    {

    case 1:

        h.admlog();

        break;

    case 2:

        h.custlog();

        break;

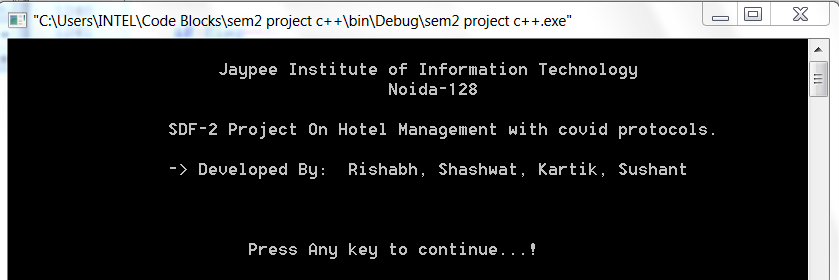
    default:

        cout << "\n Wrong Choice";

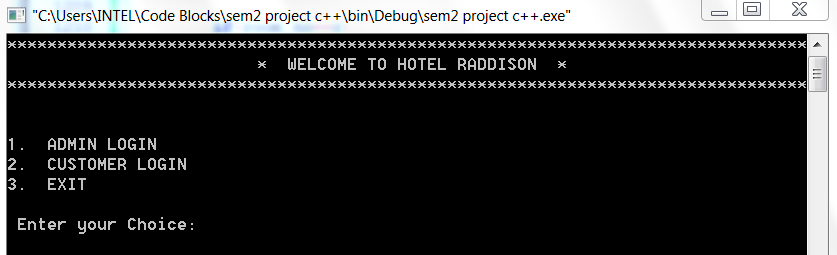
    }

    return 0;

}

**Results :**

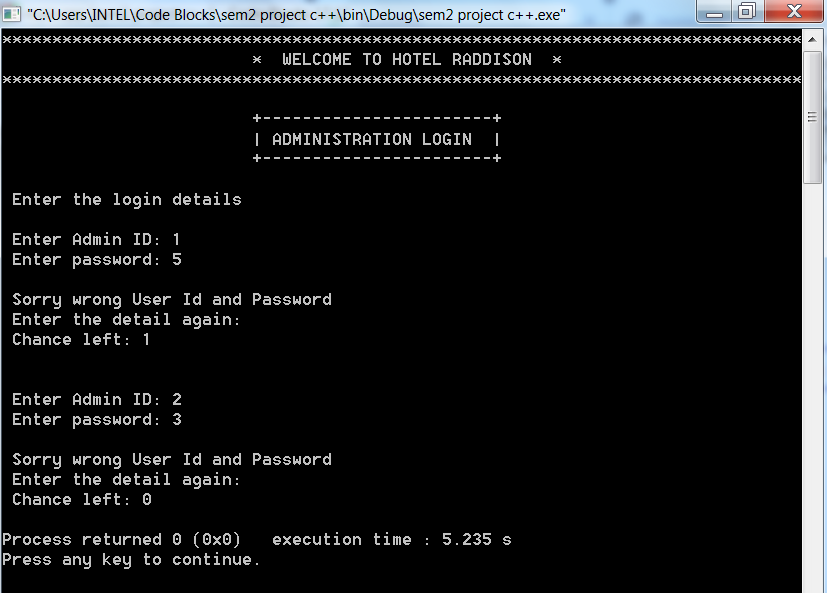
* **Fig1. Welcome Screen**

**WE CHOOSE CHOICE 1i.e.ADMIN LOGIN**

* **Fig2. Login Screen**

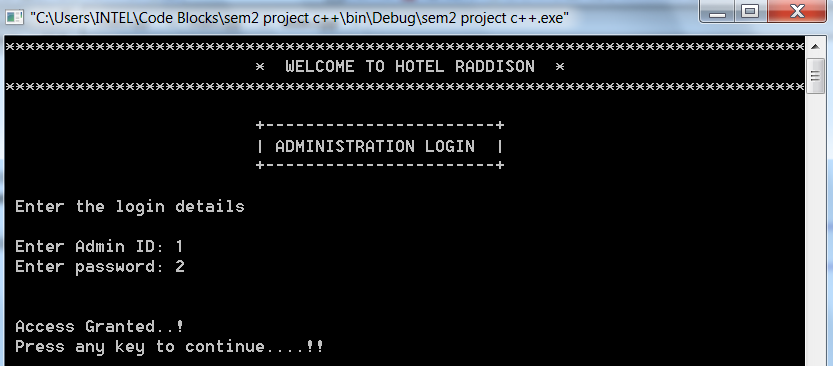
**Login screen for admin and for customers here admin have the highest priority since only**

**admin can add, delete, search, update customer as well as hotel records. On the other hand, availability of rooms can be checked by the customer through customer login.**



* **Fig3. Invalid Login Details screen**

**If admin enters wrong details only 1 more chance will be left otherwise the system will revert back to the login page.**

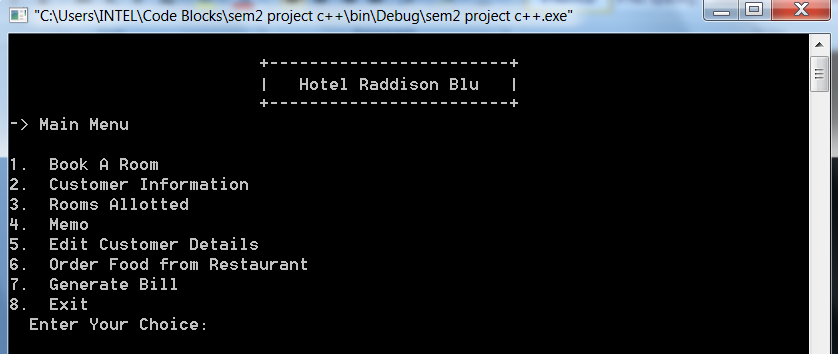


* **Fig4.Admin:-**

**Admin Login ID: 1**

**Password: 2**

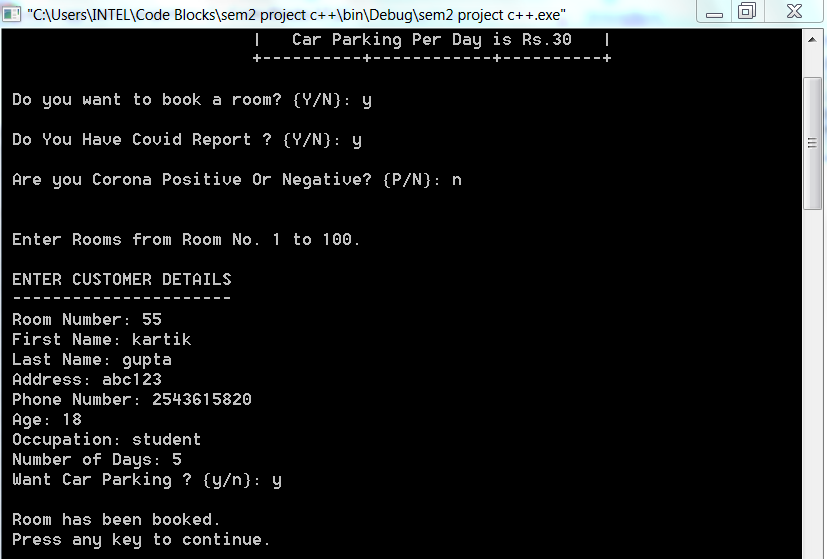
**Login page for the admin. The system can be accessed when the correct login details are entered. Only admin can edit records of hotel and customers.**



* **Fig5. Main Menu Screen**

**Options to handle records**

**After the login details are entered the main menu will open providing various options to the admin through which he can access, add, search, view and delete the various records in the system.**

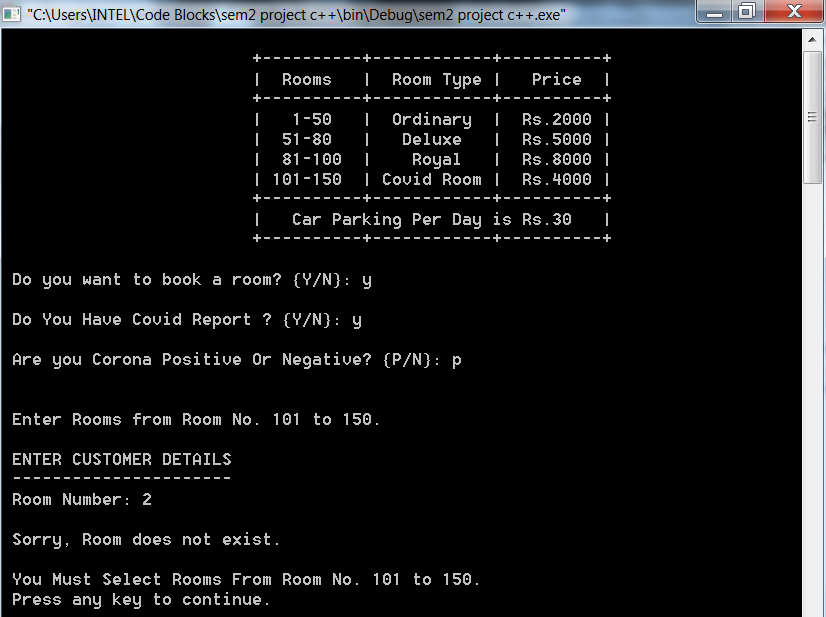
**WE CHOOSE CHOICE 1 i.e. BOOK A ROOM**

* **Fig6. Add Customer**

**Here first admin asked for covid report if customer don’t have report then extra charge will be applied and then admin asked that he/she is corona positive or negative.**

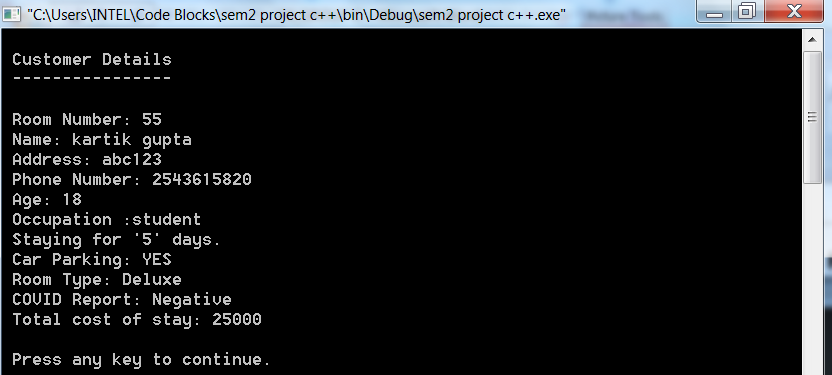
**Accordingly, the room will be allocated to customer.**

**If you enter the choice as 1. The booking screen will open through which customer can book his room by giving his personal information. If the customer is identified as covid positive he will be automatically allocated quarantine room according to the government norms. Since it is now being mandatory to bring a RT-PCR report so as to enter hotel if a customer fails to provide the report certain hotels are provided with testing kits where his reports can be provided to identify whether he is covid positive or negative. Hotel will take a certain charge for the test.**



* **Fig7. Add Room screen (Invalid room number entered)**

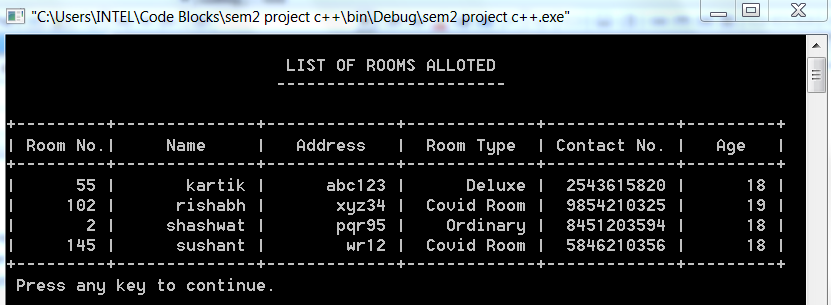
**Here customer is Corona positive and he entered room no. 2 which is available for only non-covid customers. A message will be shown that “Sorry, Room does not exist. You must select rooms from room no.101 to 150.” The same instance will happen if non covid customer tries to book a room from 101 to 150 since rooms 101 to 150 are allotted only for covid patients.**

**WE CHOOSE CHOICE 2 i.e. CUSTOMER RECORD**

* **Fig8. Customer Record**

**To Search and display the details of customer**

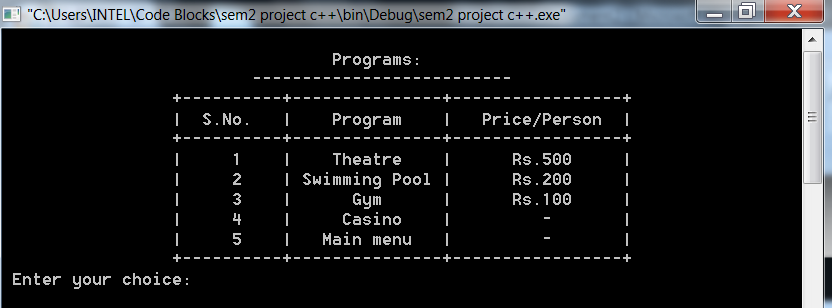
**Through main menu the admin can enter through customer details and can search for a specific room to check whether the room is booked and can also check the record of that customer.**

**WE CHOOSE CHOICE 3 i.e. Room Allocated**

* **Fig9. Rooms Allocated**

**Here we can see all rooms which are already allocated**

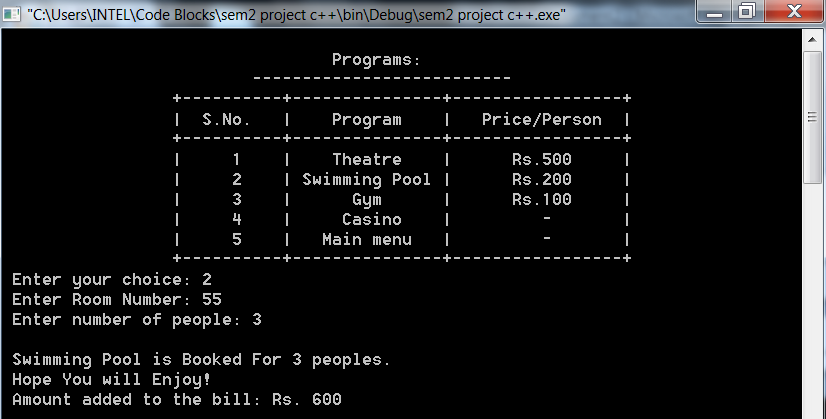
**Through Main menu screen the admin can go to the customer rooms allotted screen where he can check the records for all the rooms that have been allotted in the hotel.**

**WE CHOOSE CHOICE 4 i.e. Memo**

* **Fig10. Memo Screen**

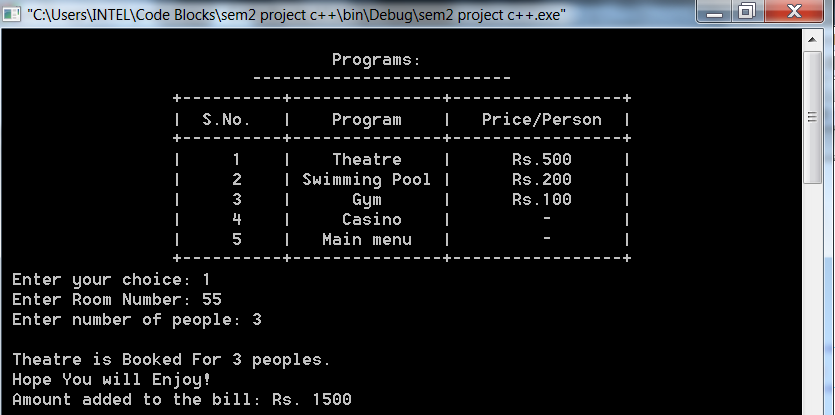
**Here customer can book a memo/program.**

**Through main menu the admin can enter the memo screen through which he can book various programs for the customers according to their choice.**



* **Fig11. Memo Screen**

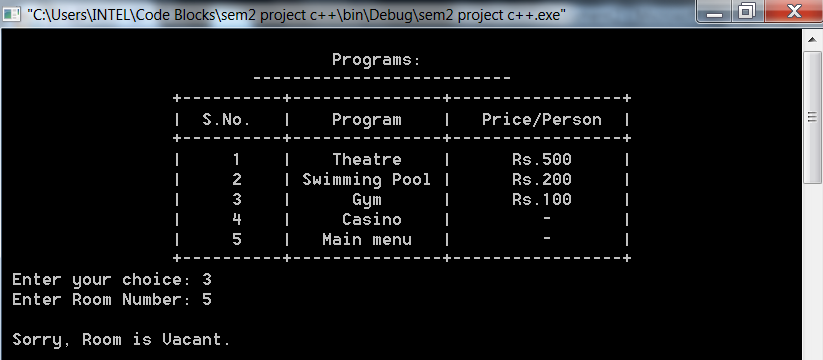
**If a customer staying in room no. 55 wants to book swimming pool along with two others (staying in his room) he will be charged per person and the total charge will be added to his checkout bill.**



* **Fig12. Memo Screen**

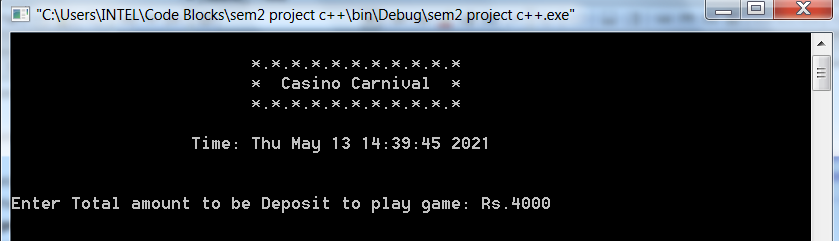
**Here again room No.55 customer book a program 1(Theatre).**

**The previous amount (Rs.600) and current amount (Rs.1500) both will be added to the bill.**



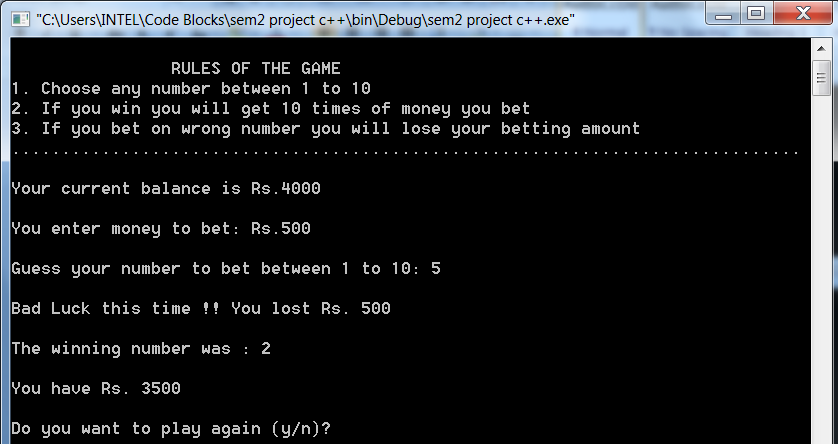
* **Fig13. Memo Screen**

**If a room which is vacant is entered.**



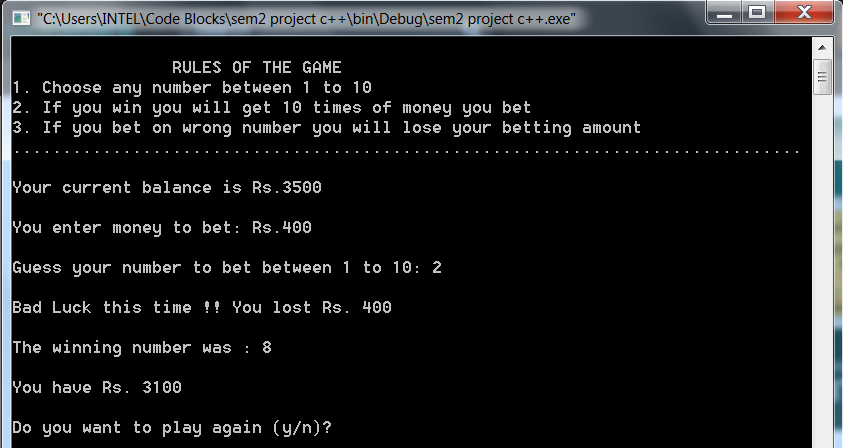
* **Fig14. Casino Carnival**

**If the user wants to play the casino he will be asked to enter his money deposit.**

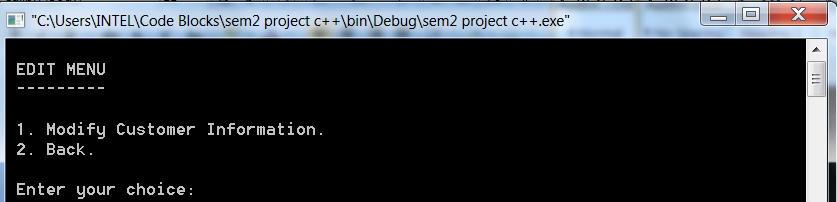


* **Fig15. Casino Screen**

**After the customer adds his balance he will be asked to put a bet and the number on which he wants to put his bet. If the customer has lost, he will be asked if he want to play the casino again.**

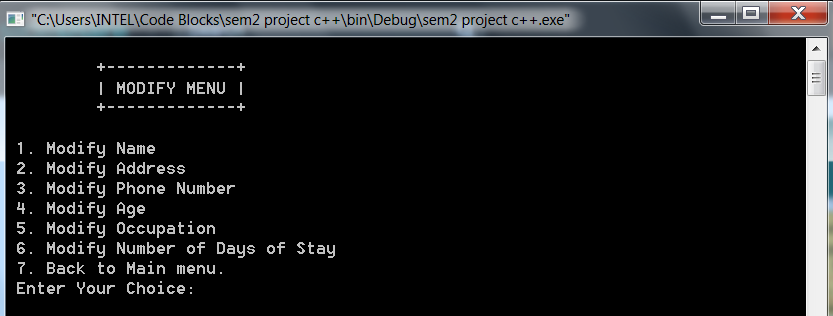


* **Fig16. casino Screen**

**WE CHOOSE CHOICE 5 i.e. EDIT**

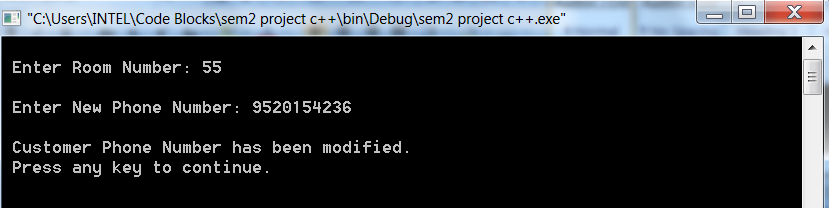
* **Fig17. Edit Menu Screen**

**Through Edit records from main menu the customer records can be edited.**



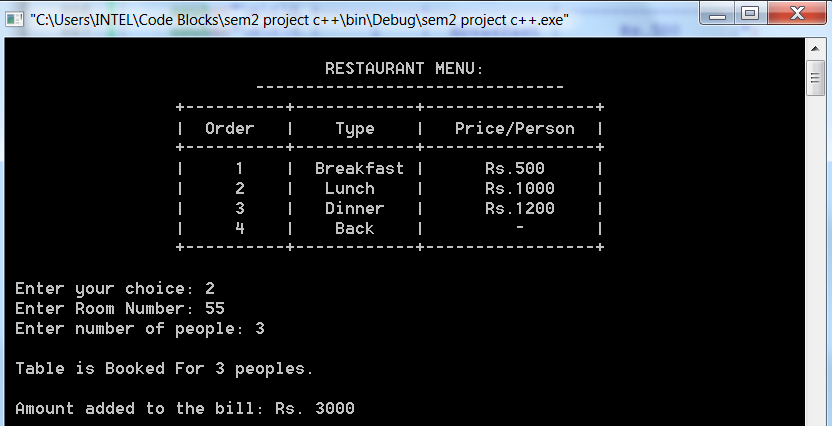
* **Fig18. Modify Menu Screen**

**If the customer wants to modify his records his records.**



* **Fig19. Modify Phone Number Screen**

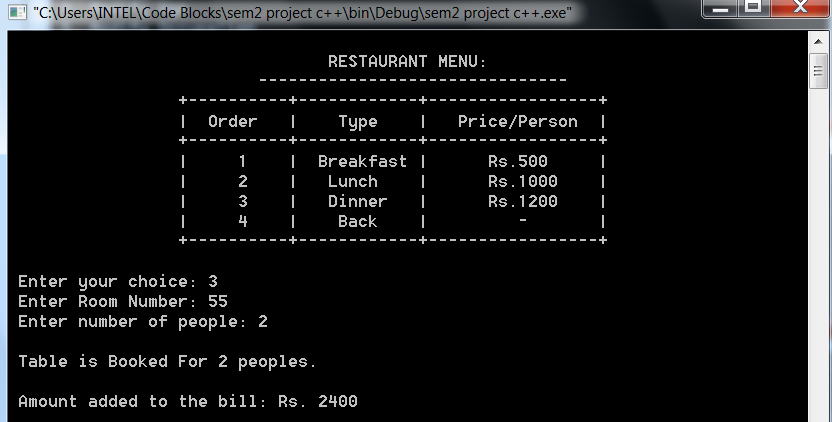
**The customer records for room no. 55 are modified.**

**WE CHOOSE CHOICE 6 i.e. ORDER FOOD** 

* **Fig20. Restaurant menu screen**

**Customer from room no. 55 ordered lunch for three people.**

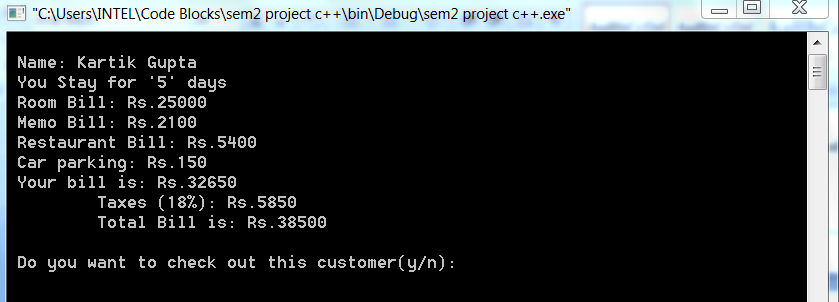
**This amount will be added to the Bill.**



* **Fig21. Restaurant Menu Screen**

**Here again room No.55 customer book Order Food type 3(Dinner)**

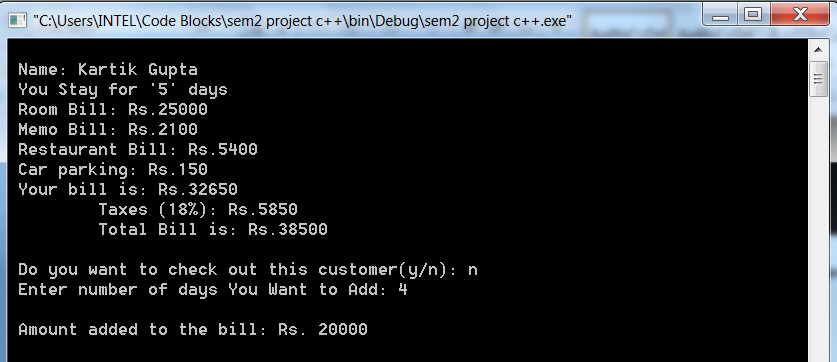
**The previous amount (Rs.3000) and current amount (Rs.1200) both will be added to the bill.**

**WE CHOOSE CHOICE 7 i.e. GENERATE BILL**

* **Fig22. Billing Screen**

**On the billing screen, customers total bill will be shown along with taxes and GST charged by the hotel.**

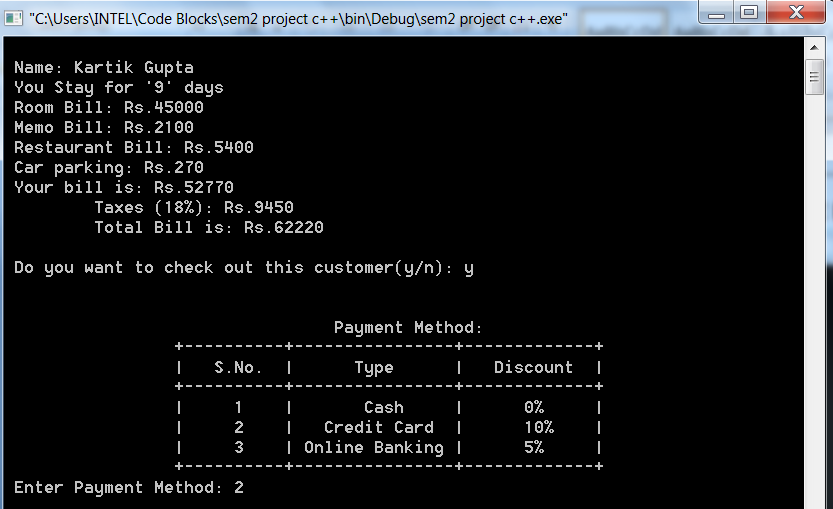
**The customer will be asked if he want to check out.**



* **Fig23. billing Screen**

**If the customer refuses to check out he will be asked to add the number of days that will be added to his stay.**

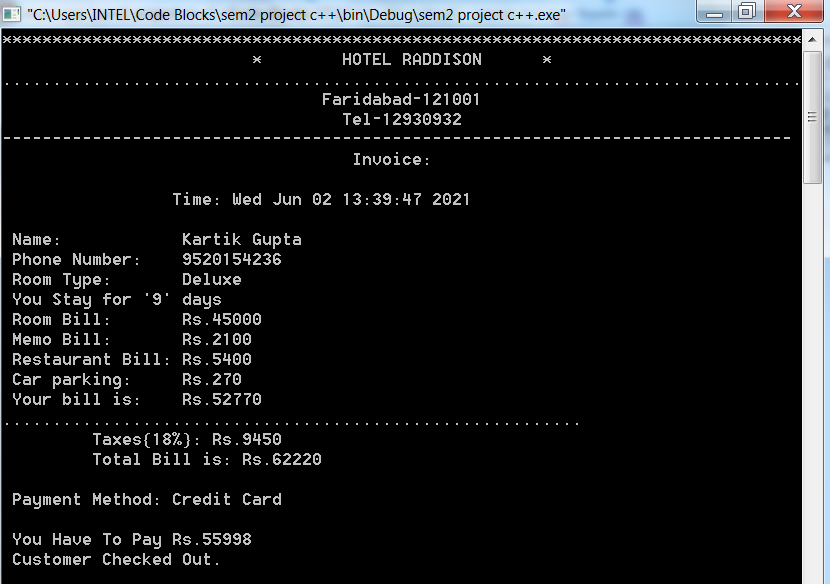
**He will be charged accordingly.**

IF AGAIN **WE CHOOSE CHOICE 7 i.e. GENERATE BILL**

* **Fig24. Billing Screen**

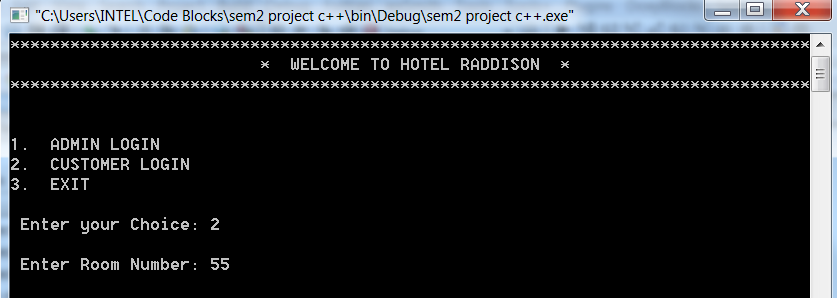
**Then admin ask to enter payment method.**

**If a customer wants to checkout he will be allowed to pay his cash through different payment methods and discount will be provided accordingly.**



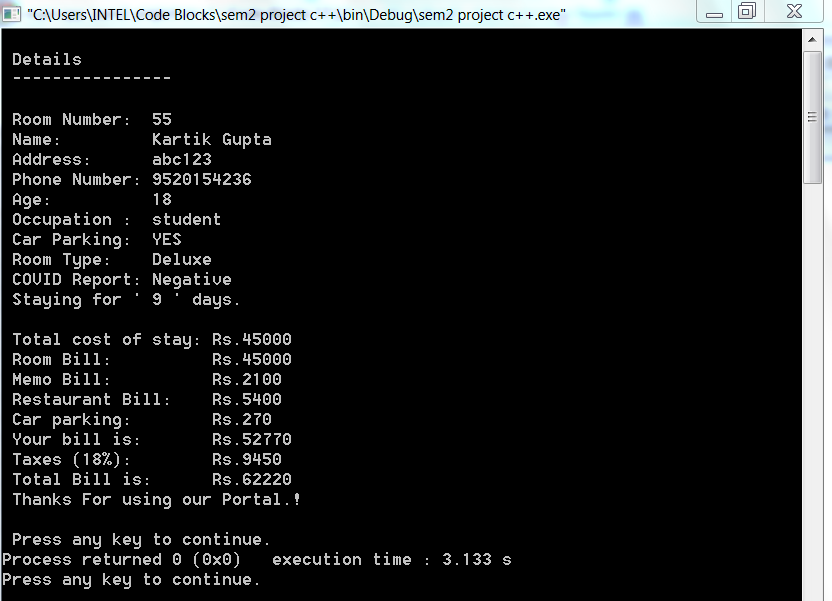
* **Fig25. Invoice**

**Here the customer wants to pay his bill through credit card he will be provided discount and his final bill will be shown.**

AFTER EXITING FROM ADMIN LOGIN USER SELECT **CHOICE 2 i.e. CUSTOMER LOGIN**

* **Fig26. LOGIN Screen**

**If user enters through customer login. The user will be asked the room for which he wants to see the records.**

**Fig27. Customer Portal**

**The Details of existing customer will be displayed.**

**CONCLUSION:**

It was wonderful learning experience for our team. While creating this project we came across the various concepts of OOPS programming. We also how it helps to make our program less cumbersome and easy, to read, write and edit. Along with that we used the concepts of file handling too which makes it easy for the user to edit the records easily. With the help of various concepts of oops programming it became quite easy to execute the project and to adapt it to modern covid norms and guidelines. With help of the various concepts it became easy to execute a modern billing system (itemized) which aims to adapt itself with modern covid times and modern taxing system.

Object-oriented languages provide sophisticated concepts in the simplest possible framework In C++ the framework is not simple and the concepts are obscured. OOP addresses many issues in order to facilitate the production of complex and sophisticated program. OOPS reduces the complexity of the program

The purpose of the project was to change the old notion of the hotel management system and to create a new one which can completely adapt itself with modern times and makes the work easier for the user to handle the system during these difficult times.

**Reference:**

[1] *https://www.codegrepper.com/*

[2] *https://www.guru99.com/cpp-file-read-write-open.html*

[3]*https://www.google.com/search?q=invoice+format&tbm=isch&chips=q:invoice+format,g\_1:gst:p5w-fiAuedg%3D&rlz=1C1CHBF\_enIN913IN913&hl=en&sa=X&ved=2ahUKEwiQwZP3nvjwAhXJhksFHQgjCE4Q4lYoBHoECAEQIA&biw=1007&bih=694#imgrc=2HkuIH4-nKcVwM*

[4] *Computer Science with C++ Class XII by Sumita Aror*