Visvesvaraya Technological UniversityBelagavi-590 018, Karnataka

A Mini Project Report on

"HOTEL MANAGEMENT DATABASE"

Mini Project Report submitted in partial fulfilment of the requirement for the DBMS Laboratory with Mini Project [18CSL58]

Bachelor of Engineering

in

Computer Science and Engineering

Submitted by

Rohit Joshi [1JT19CS074]

Shravan K G [1JT19CS086]

Department of Computer Science and Engineering

Jyothy Institute of Technology Tataguni, Bengaluru-560082

Jyothy Institute of Technology

Tataguni, Bengaluru-560082 Department of Computer Science and Engineering

CERTIFICATE

Certified that the mini project work entitled **"HOTEL MANAGEMENT DATABASE"** carried out by **Rohit Joshi [1JT19CS074] and Shravan K G [1JT19CS086]** bonafide students of Jyothy Institute of Technology, in partial fulfilment for the award of **Bachelor of Engineering** in **Computer Science and Engineering** department of the **Visvesvaraya Technological University, Belagavi** during the year **2021-2022**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

Mrs. Nikitha S
Guide, Asst. Professor
Dept. of CSE

Dr. Prabhanjan S
Professor & HOD
Dept. of CSE

External Viva Examiner

Signature with Date :1.

2.

ACKNOWLEDGEMENT

Firstly, we are very grateful to this esteemed institution **"Jyothy Institute of Technology**"for providing us an opportunity to complete our project.

We express our sincere thanks to our **Principal Dr. Gopalakrishna K** for providing us with adequate facilities to undertake this project.

We would like to thank **Dr. Prabhanjan S, Professor and Head of Computer Science** and Engineering Department for providing for his valuable support.

We would like to thank our guides **Mrs. Nikitha S, Assistant Professor** for their keeninterest and guidance in preparing this work.

Finally, we would thank all our friends who have helped us directly or indirectly in thisproject.

Rohit Joshi [1JT19CS074]
Shravan K G [1JT19CS086]

ABSTRACT

The system aims at the maintenance and management of the different Hotels that are available in the different parts of the world. It mainly takes care of the Hotel management at the core area of the database. The system provides the information regarding the different Hotels that are available and their status specific to availability.
The guests can register themselves with the required information that is expected by the system. Each registered guest can raise a request for the unit bookings. The Guests are scheduled with the information of the availability of the units for they have requested the time.

The application is designed to make the existing system more reliable, fast and easy for all, provides a methodical way of managing large databases. For this application we used the backend as SQL to store the data which is used in the application and for the user interface we have used python.

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

1. INTRODUCTION

1.1 Introduction to DBMS

A database is simply an organized collection of related data, typically stored on disk, and accessible by many concurrent users, it is a logically coherent collection of data with some inherent meaning, representing some aspect of real world and which is designed, built and populated with data for a specific purpose.

Databases are managed by a Database Management System(DBMS) which is a collection of programs that enables user to create and maintain a database.

Advantages of DBMS:

1. Redundancy is controlled.
2. Unauthorized access is restricted.
3. Providing multiple user interfaces.
4. Enforcing integrity constraints.
5. Providing backup and recovery.

1.2 Introduction to SQL

Structured Query Language (SQL), is a language used to request data from a databasewhich includes database creation, deletion, retrieval of required tables and even manipulation of data held in a relational database management system.
SQL is considered as a Non-Procedural or a High level language in which the expected result or operation is given without the specific details about how to accomplish the task. So,SQL is a declarative language.
Therefore, SQL is designed at a higher conceptual level of operation than procedural languages as procedural languages includes only the information about opening and closingtables, loading and searching indexes, or flushing buffers and writing data to file systems, but the lower level logical and physical operations are not specified in SQL.

1.3 Introduction to Hotel Management Database

The project, Hotel Management System is a desktop application that allows the hotel manager to handle all hotel activities. Interactive GUI and the ability to manage various hotel bookings and rooms make this system very flexible and convenient.
Hotel management project provides room booking, staff management and other necessary hotel management features. The system allows the manager to post available rooms in the system. Receptionist can view and book room for customers. Other hotel services can also be viewed by the customers and can book them too. The system is hence useful for both customers and managers to portable manage the hotel activities.

CHAPTER 2
DESIGN

Theory of ER Diagram

The Entity–Relationship model (ER model) describes the structure of a database with thehelp of a diagram, which is known as **Entity Relationship Diagram (ER Diagram)**

An **Entity Relationship Diagram (ERD)** shows the relationships of entity sets stored in adatabase. An entity in this context is an object, a component of data.

An entity set is a collection of similar entities. These entities can have attributes that defineits properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of database.

ER diagrams are used to sketch out the design of a database.

ENTITIES

An entity is an 'object' in the real world with an independent existence and an entity type defines a collection (or set) of entities that have the same attributes. Each entity type in thedatabase is described by its name and attributes.

An entity type is represented in ER diagrams as a rectangular box enclosing the entity typename.

RELATIONSHIPS

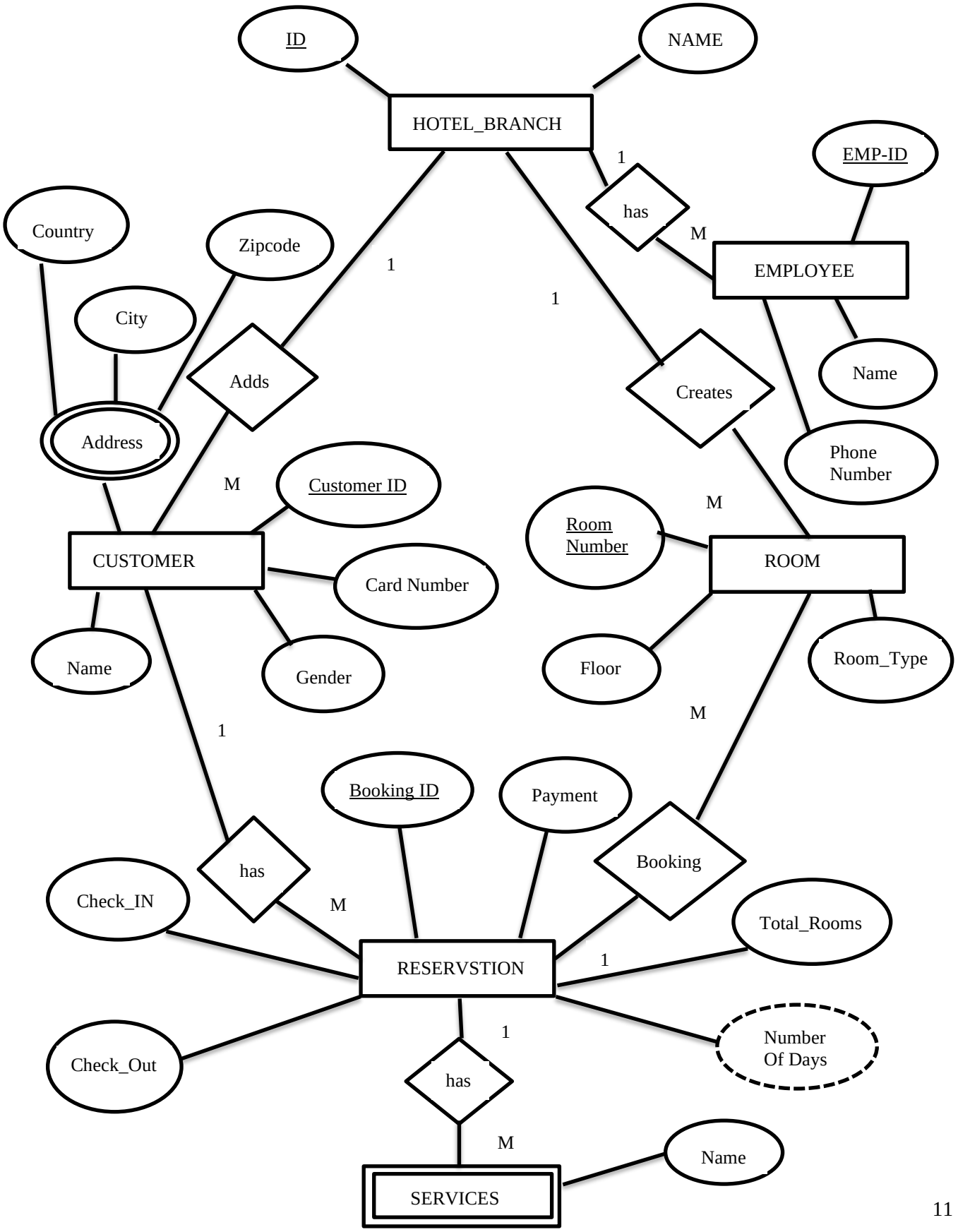
A relationship among two or more entities represents an association among the entities andwhenever an attribute of one entity refers to another entity, there exists a relationship between the two entities. In a relationship, a foreign key of one table refers the primary key of the other table and it is represented by diamond shape in ER diagram.

ATTRIBUTES

An attribute represents some property of interest that further describes an entity and thecolumn header of the table shows the attributes. Each attribute in a table has a certain domain which allows it to accept a certain ‘set of values’ only.

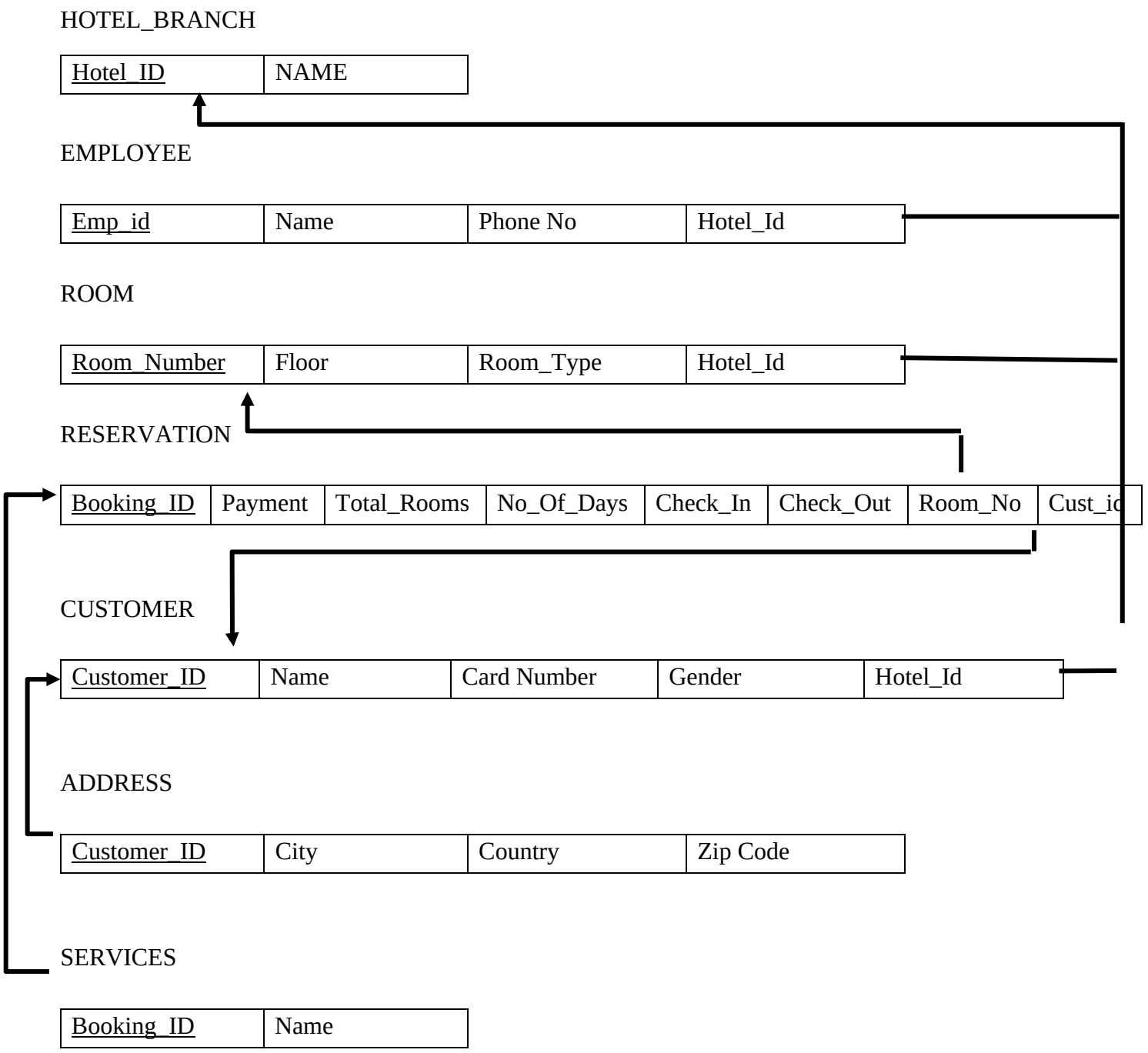
The attribute values, of each entity, will define its characteristics in the table and isrepresented by oval in the ER diagram.

ER DIAGRAM


ID
NAME
HOTEL_BRANCH
1
has
M
EMP-ID
EMPLOYEE
Country
Zipcode
City
1
1
Adds
Creates
Name
Address
Phone
Number
M
Customer ID
M
Room
Number
CUSTOMER
Card Number
ROOM
Name
Gender
Floor
Room_Type
M
1
Booking ID
Payment
has
Booking
Check_IN
M
Total_Rooms
RESERVSTION
1
1
Number
Of Days
Check_Out
has
M
Name
SERVICES

n

SCHEMA DIAGRAM



List of Tables

CHAPTER 3
IMPLEMENTATION

Create Table Commands:

Hotel Branch Table

```
CREATE TABLE `hotel` (  `ID` int(11) NOT NULL, `Name`varchar(45) DEFAULT NULL, PRIMARY KEY (`ID`)) ;
```

Customer Table

```
CREATE TABLE `customer` (  `ID` int(11) NOT NULL, `Name` varchar(100) DEFAULT NULL,`Card Number` varchar(100) DEFAULT NULL, `Gender` varchar(10) DEFAULT NULL, `hotel_id` int(11) DEFAULT NULL, PRIMARY KEY (`ID`),  KEY `hotel_id` (`hotel_id`),CONSTRAINT `customer_ibfk_1` FOREIGN KEY (`hotel_id`) REFERENCES `hotel` (`ID`) ON DELETE CASCADE) ;
```

Address Table

```
CREATE TABLE `address` (`Customer_ID` int(11) DEFAULT NULL,`Country` varchar(100) DEFAULT NULL, `City` varchar(100) DEFAULT NULL,`ZipCode` varchar(100) DEFAULT NULL,  `hotel_id` int(11) DEFAULT NULL,  KEY `hotel_id` (`hotel_id`),  KEY `address_ibfk_2` (`Customer_ID`), CONSTRAINT `address_ibfk_1` FOREIGN KEY (`hotel_id`) REFERENCES `hotel` (`ID`) ON DELETE CASCADE, CONSTRAINT `address_ibfk_2` FOREIGN KEY (`Customer_ID`) REFERENCES `customer` (`ID`) ON DELETE CASCADE ON UPDATE NO ACTION)
```

Reservation Table

```
CREATE TABLE `booking` (`Booking_ID` int(11) NOT NULL, `Customer_ID` int(11) DEFAULT NULL, `hotel_id` int(11) DEFAULT NULL,  `Check_in_date` varchar(100) DEFAULT NULL,  `Check_out_date` varchar(100) DEFAULT NULL, `Total_Rooms` varchar(100) DEFAULT NULL,`Payment` varchar(100) DEFAULT NULL, `Room_Number` varchar(100) DEFAULT NULL,`No_of_Days` varchar(45) DEFAULT NULL, PRIMARY KEY (`Booking_ID`),  KEY `Customer_ID` (`Customer_ID`),KEY `hotel_id` (`hotel_id`),  CONSTRAINT `booking_ibfk_1` FOREIGN KEY (`Customer_ID`) REFERENCES `customer` (`ID`) ON DELETE CASCADE, CONSTRAINT `booking_ibfk_2` FOREIGN KEY (`hotel_id`) REFERENCES `hotel` (`ID`) ON DELETE CASCADE);
```

Room Details Table

```
CREATE TABLE `details` (`floor` int(11) DEFAULT NULL, `RoomNumber` varchar(45) NOT NULL, `RoomType` varchar(45) NOT NULL, `Booked` varchar(45) DEFAULT NULL, `hotel_id` varchar(45) DEFAULT NULL,  PRIMARY KEY (`RoomNumber`), UNIQUE KEY `RoomNumber` (`RoomNumber`)) ;
```

Employee Table

```
CREATE TABLE `employee` ( `ID` int(11) NOT NULL, `Name` varchar(50) DEFAULT NULL, `Gender` varchar(10) DEFAULT NULL, `Phone_Number` varchar(45) DEFAULT NULL, `hotel_id` int(11) DEFAULT NULL,  PRIMARY KEY (`ID`),  KEY `hotel_id` (`hotel_id`), CONSTRAINT `employee_ibfk_1` FOREIGN KEY (`hotel_id`) REFERENCES `hotel` (`ID`) ON DELETE CASCADE) ;
```

Services Table

```
CREATE TABLE `services` (`Booking_ID` int(11) DEFAULT NULL,`Name` varchar(100) DEFAULT NULL,  KEY `services_ibfk_1` (`Booking_ID`));
```

INSERTING VALUES

ADDRESS TABLE:

```
INSERT INTO ADDRESS VALUES(3363,'INDIA','BANGALORE',560011,1);
INSERT INTO ADDRESS VALUES(2296,'INDIA','BANGALORE',560011,1);
INSERT INTO ADDRESS VALUES(1904,'INDIA','SAGARA',65200,3);
INSERT INTO ADDRESS VALUES(1565,'USA','MANHATTAN',560050,3);
INSERT INTO ADDRESS VALUES(6733,'INDIA','KARKAL',756015,3);
```

Customer_ID	Country	City	ZipCode	hotel_id
3363	India	Bangalore	560011	1
2296	India	Bangalore	560011	1
1904	India	Sagara	65200	3
1565	USA	Manhattan	560050	3
6733	India	Karkal	756015	3

RESERVATIONS TABLE:

```
INSERT INTO BOOKING VALUES(1099,3363,1,26-JAN-2022,27-JAN-
2022,4,2200.00,100,1);
INSERT INTO BOOKING VALUES(1120,6733,1,28-JAN-2022,29-JAN-
2022,4,2750.00,105,1);
INSERT INTO BOOKING VALUES(6399,3363,1,12-DEC-2017,22-DEC-
2017,15,22000.00,128,10);
INSERT INTO BOOKING VALUES(6601,6733,1,28-JAN-2022,30-JAN-
2022,2,3300.00,208,2);
INSERT INTO BOOKING VALUES(8028,1565,1,27-JAN-2022,31-JAN-
2022,1,11000.00,104,4);
INSERT INTO BOOKING VALUES(9137,1565,1,27-JAN-2022,28-FEB-
2022,2,88000.00,200,32);
```

Booking_ID	Customer_ID	hotel_id	Check_in_date	Check_out_date	Total_Rooms	Payment	Room_Number	No_of_Days
1099	3363	1	26/01/2022	27/01/2022	4	Rs.2200.00	100	1
1120	6733	3	28/01/2022	29/01/2022	4	Rs.2750.00	105	1
6399	3363	1	12/12/2017	22/12/2017	15	Rs.22000.00	28	10
6601	6733	3	28/01/2022	30/01/2022	2	Rs.3300.00	208	2
8028	1565	3	27/01/2022	31/01/2022	1	Rs.11000.00	104	4
9137	1565	3	27/01/2022	28/02/2022	2	Rs.88000.00	200	32

CUSTOMER TABLE:

```
INSERT INTO CUSTOMER VALUES(1565,'RAKSHITH',784512345,'MALE',3);
INSERT INTO CUSTOMER VALUES(1904,'SHRAVAN',4784,'MALE',3);
INSERT INTO CUSTOMER VALUES(2296,'SHRAVAN',789456123,'MALE',1);
INSERT INTO CUSTOMER VALUES(3363,'ROHIT',87945,'MALE',1);
INSERT INTO CUSTOMER VALUES(6733,'KEERTHI,123456789123,'MALE',3);
```

ID	Name	Card Number	Gender	hotel_id
1565	Rakshith	784512345	Male	3
1904	Shravan	4784	Male	3
2296	Shravan	789456123	Male	1
3363	Rohit	87945	Male	1
6733	Keerthi	123456789123	Male	3

ROOM DETAILS TABLE:

```
INSERT INTO DETAILS TABLE(1,100,’LUXURY’,’TRUE’,2);
INSERT INTO DETAILS TABLE(1,104,’DOUBLE’,’TRUE’,1);
INSERT INTO DETAILS TABLE(1,105,’DOUBLE’,’TRUE’,3);
INSERT INTO DETAILS TABLE(1,206,’SINGLE’,’FALSE’,1);
INSERT INTO DETAILS TABLE(1,107,’SINGLE’,’FALSE’,1);
INSERT INTO DETAILS TABLE(1,108,’SINGLE’,’TRUE’,3);
INSERT INTO DETAILS TABLE(1,109,’LUXURY’,’FALSE’,3);
INSERT INTO DETAILS TABLE(1,300,’SINGLE’,’FALSE’,3);
```

floor	RoomNumber	RoomType	Booked	hotel_id
1	100	Luxury	True	2
1	104	Double	True	1
1	105	Double	True	3
2	206	Single	False	1
2	207	Single	False	1
2	208	Single	True	3
2	209	Luxury	False	3
3	300	Single	False	3

HOTEL TABLE:

```
INSERT INTO HOTEL VALUES(1,’HOTEL1’);
INSERT INTO HOTEL VALUES(2,’HOTEL2’);
INSERT INTO HOTEL VALUES(3,’HOTEL3’);
```

ID	Name
1	Hotel1
2	Hotel2
3	Hotel3

EMPLOYEE TABLE:

```
INSERT INTO EMPLOYEE VALUES(2427,’MANOJ’,7845126578,’MALE’1);
```

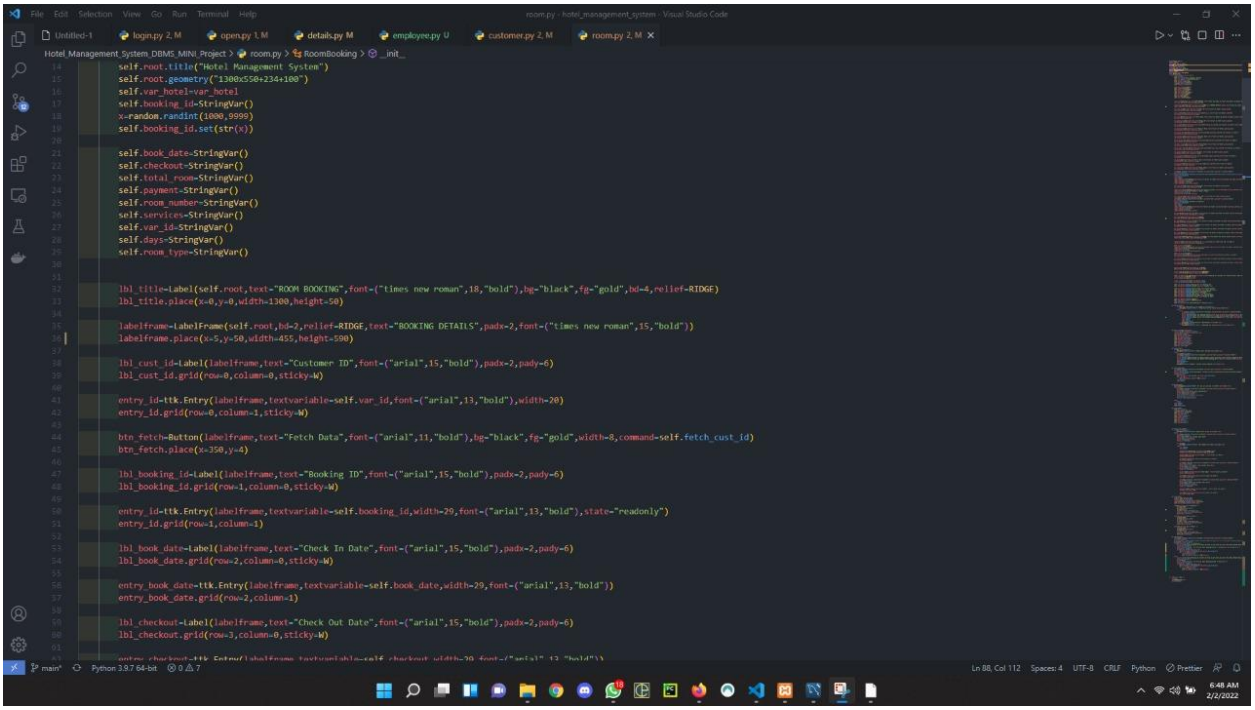
ID	Name	Gender	Phone_Number	hotel_id
2427	Manoj	7845126578	Male	1

SERVICES TABLE:

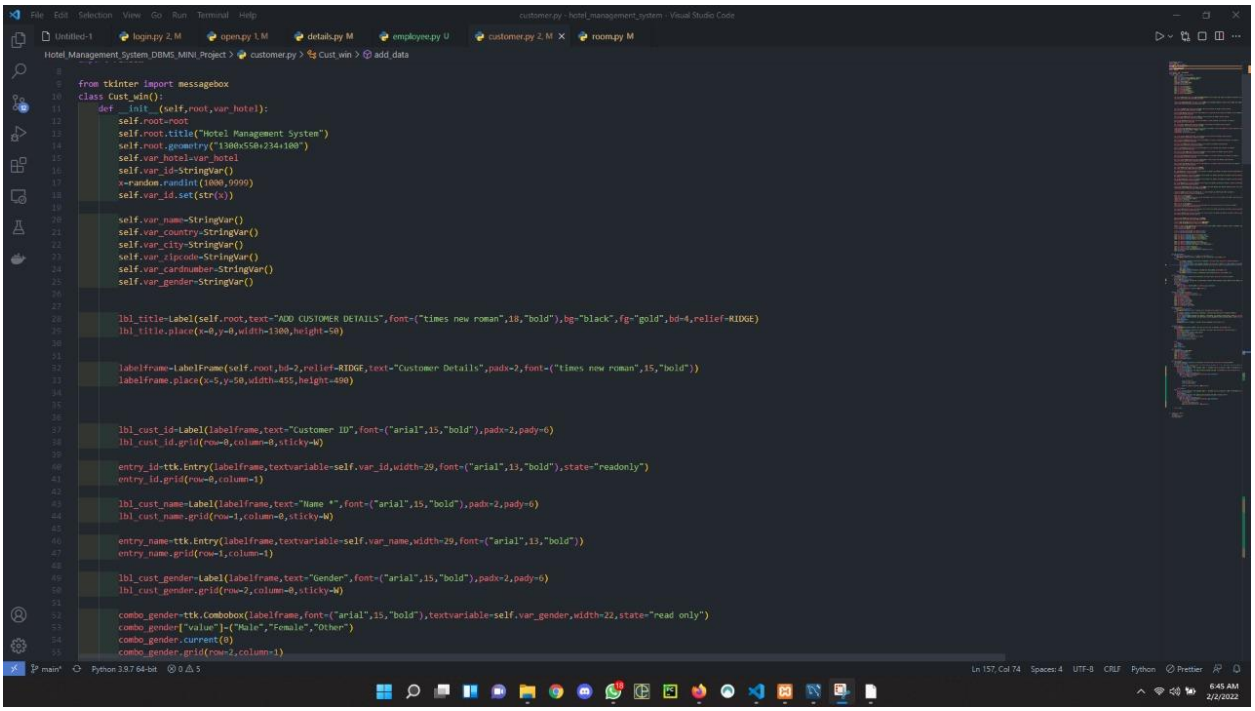
```
INSERT INTO SERVICES VALUES(6399,’BREAKFAST’);
INSERT INTO SERVICES VALUES(1099,’BREAKFAST’);
INSERT INTO SERVICES VALUES(8028,’DINNER’);
INSERT INTO SERVICES VALUES(9137,’DINNER’);
INSERT INTO SERVICES VALUES(1120,’DINNER’);
INSERT INTO SERVICES VALUES(6601,’MEALS’);
```

Booking_ID	Name
6399	Breakfast
1099	Breakfast
8028	Dinner
9137	Dinner
1120	Dinner
6601	Meals

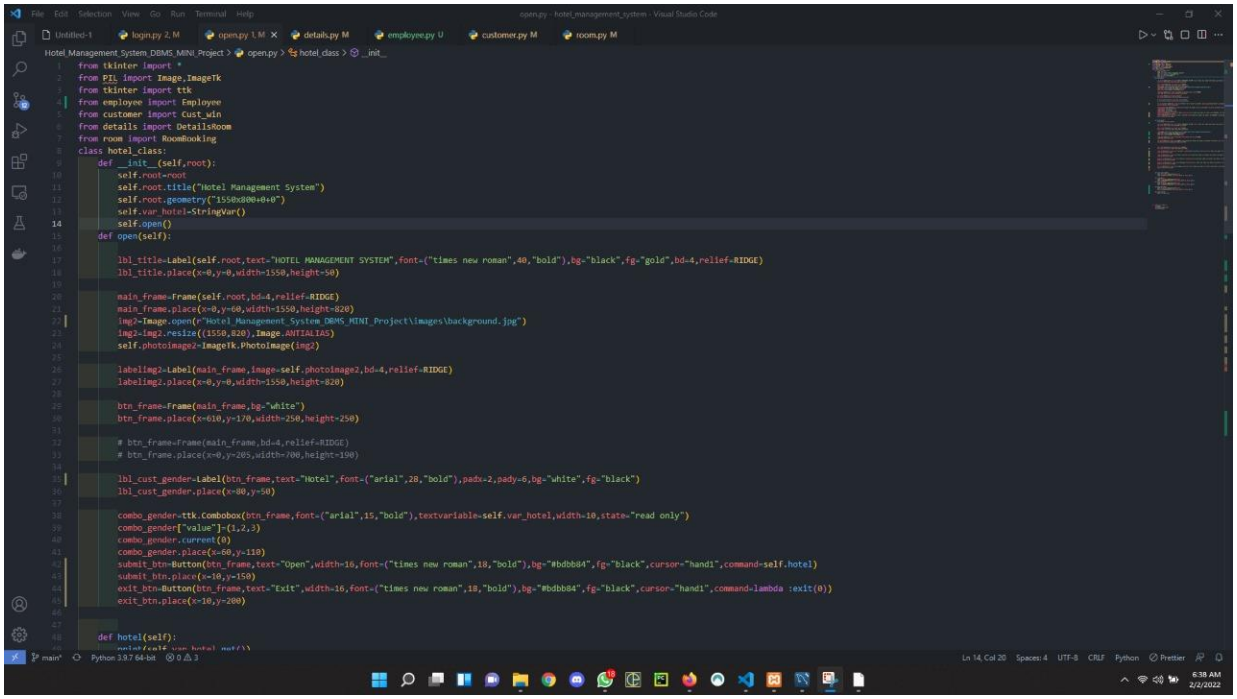
GUI Implementation



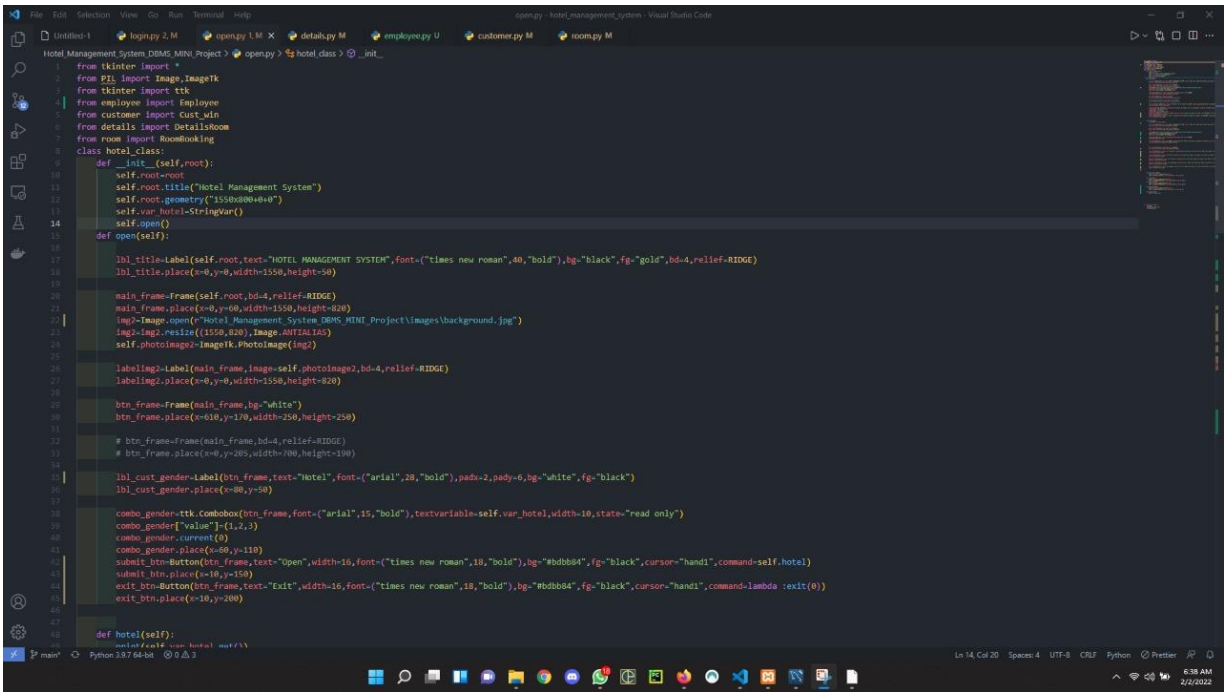
```
        self.root.title("Hotel Management System")
        self.root.geometry("1300x550+234+100")
        self.var_hotel=var_hotel
        self.booking_id=StringVar()
        x=random.randint(1000,9999)
        self.booking_id.set(str(x))

        self.book_date=StringVar()
        self.checkout=StringVar()
        self.total_room=StringVar()
        self.payment=StringVar()
        self.room_number=StringVar()
        self.services=StringVar()
        self.var_id=StringVar()
        self.days=StringVar()
        self.room_type=StringVar()


        lbl_title=Label(self.root,text="ROOM BOOKING",font=("times new roman",18,"bold"),bg="black",fg="gold",bd=4,relief=RIDGE)
        lbl_title.place(x=0,y=0,width=1300,height=50)

        labelframe=LabelFrame(self.root,bd=2,relief=RIDGE,text="BOOKING DETAILS",padx=2,font=("times new roman",15,"bold"))
        labelframe.place(x=5,y=50,width=455,height=590)

        lbl_cust_id=Label(labelframe,text="Customer ID",font=("arial",15,"bold"),padx=2,pady=6)
        lbl_cust_id.grid(row=0,column=0,sticky=W)

        entry_id=ttk.Entry(labelframe,textvariable=self.var_id,font=("arial",13,"bold"),width=20)
        entry_id.grid(row=0,column=1,sticky=W)

        btn_fetch=Button(labelframe,text="Fetch Data",font=("arial",11,"bold"),bg="black",fg="gold",width=8,command=self.fetch_cust_id)
        btn_fetch.place(x=350,y=4)

        lbl_booking_id=Label(labelframe,text="Booking ID",font=("arial",15,"bold"),padx=2,pady=6)
        lbl_booking_id.grid(row=1,column=0,sticky=W)

        entry_id=ttk.Entry(labelframe,textvariable=self.booking_id,width=29,font=("arial",13,"bold"),state="readonly")
        entry_id.grid(row=1,column=1)

        lbl_book_date=Label(labelframe,text="Check In Date",font=("arial",15,"bold"),padx=2,pady=6)
        lbl_book_date.grid(row=2,column=0,sticky=W)

        entry_book_date=ttk.Entry(labelframe,textvariable=self.book_date,width=29,font=("arial",13,"bold"))
        entry_book_date.grid(row=2,column=1)

        lbl_checkout=Label(labelframe,text="Check Out Date",font=("arial",15,"bold"),padx=2,pady=6)
        lbl_checkout.grid(row=3,column=0,sticky=W)
```



```
from tkinter import messagebox
class Cust_win():
    def __init__(self,root,var_hotel):
        self.root=root
        self.root.title("Hotel Management System")
        self.root.geometry("1300x550+234+100")
        self.var_hotel=var_hotel
        self.var_id=StringVar()
        x=random.randint(1000,9999)
        self.var_id.set(str(x))

        self.var_name=StringVar()
        self.var_country=StringVar()
        self.var_city=StringVar()
        self.var_zipcode=StringVar()
        self.var_cardnumber=StringVar()
        self.var_gender=StringVar()


        lbl_title=Label(self.root,text="ADD CUSTOMER DETAILS",font=("times new roman",18,"bold"),bg="black",fg="gold",bd=4,relief=RIDGE)
        lbl_title.place(x=0,y=0,width=1300,height=50)


        labelframe=LabelFrame(self.root,bd=2,relief=RIDGE,text="Customer Details",padx=2,font=("times new roman",15,"bold"))
        labelframe.place(x=5,y=50,width=455,height=490)



        lbl_cust_id=Label(labelframe,text="Customer ID",font=("arial",15,"bold"),padx=2,pady=6)
        lbl_cust_id.grid(row=0,column=0,sticky=W)

        entry_id=ttk.Entry(labelframe,textvariable=self.var_id,width=29,font=("arial",13,"bold"),state="readonly")
        entry_id.grid(row=0,column=1)

        lbl_cust_name=Label(labelframe,text="Name *",font=("arial",15,"bold"),padx=2,pady=6)
        lbl_cust_name.grid(row=1,column=0,sticky=W)

        entry_name=ttk.Entry(labelframe,textvariable=self.var_name,width=29,font=("arial",13,"bold"))
        entry_name.grid(row=1,column=1)

        lbl_cust_gender=Label(labelframe,text="Gender",font=("arial",15,"bold"),padx=2,pady=6)
        lbl_cust_gender.grid(row=2,column=0,sticky=W)

        combo_gender=ttk.Combobox(labelframe,font=("arial",15,"bold"),textvariable=self.var_gender,width=22,state="read only")
        combo_gender["value"]=("Male","Female","Other")
        combo_gender.current(0)
        combo_gender.grid(row=2,column=1)
```



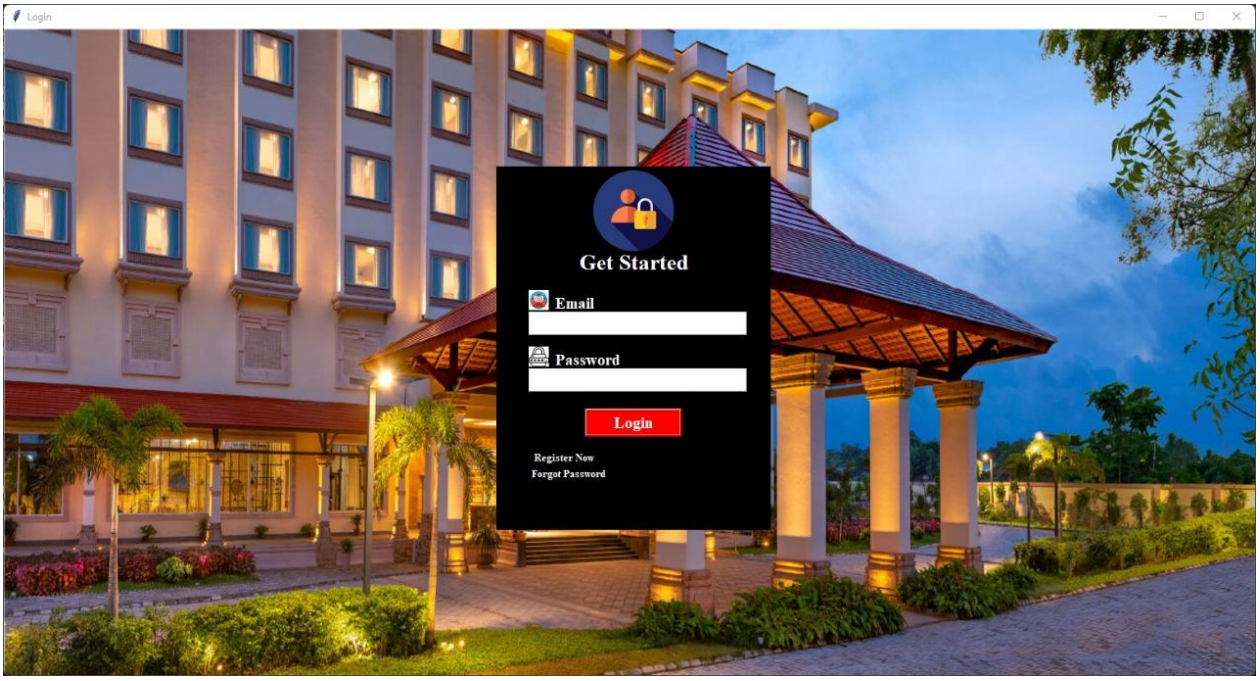
```python
from tkinter import *
from PIL import Image,ImageTk
from tkinter import ttk
from employee import Employee
from customer import Cust_win
from details import DetailsRoom
from room import RoomBooking
class hotel_class:
    def __init__(self,root):
        self.root=root
        self.root.title("Hotel Management System")
        self.root.geometry("1550x800+0+0")
        self.var_hotel=StringVar()
        self.open()
    def open(self):

        lbl_title=Label(self.root,text="HOTEL MANAGEMENT SYSTEM",font=("times new roman",40,"bold"),bg="black",fg="gold",bd=4,relief=RIDGE)
        lbl_title.place(x=0,y=0,width=1550,height=50)

        main_frame=Frame(self.root,bd=4,relief=RIDGE)
        main_frame.place(x=0,y=60,width=1550,height=820)
        img2=Image.open(r"Hotel_Management_System_DBMS_MINI_Project\images\background.jpg")
        img2=img2.resize((1550,820),Image.ANTIALIAS)
        self.photoimage2=ImageTk.PhotoImage(img2)

        labelImg2=Label(main_frame,image=self.photoimage2,bd=4,relief=RIDGE)
        labelImg2.place(x=0,y=0,width=1550,height=820)

        btn_frame=Frame(main_frame,bg="white")
        btn_frame.place(x=610,y=170,width=250,height=250)

        # btn_frame=Frame(main_frame,bd=4,relief=RIDGE)
        # btn_frame.place(x=0,y=205,width=700,height=190)

        lbl_cust_gender=Label(btn_frame,text="Hotel",font=("arial",28,"bold"),padx=2,pady=6,bg="white",fg="black")
        lbl_cust_gender.place(x=80,y=50)

        combo_gender=ttk.Combobox(btn_frame,font=("arial",15,"bold"),textvariable=self.var_hotel,width=10,state="read only")
        combo_gender["value"]=(1,2,3)
        combo_gender.current(0)
        combo_gender.place(x=60,y=110)
        submit_btn=Button(btn_frame,text="Open",width=16,font=("times new roman",18,"bold"),bg="#bdbb84",fg="black",cursor="hand1",command=self.hotel)
        submit_btn.place(x=10,y=150)
        exit_btn=Button(btn_frame,text="Exit",width=16,font=("times new roman",18,"bold"),bg="#bdbb84",fg="black",cursor="hand1",command=lambda :exit(0))
        exit_btn.place(x=10,y=200)


    def hotel(self):
```



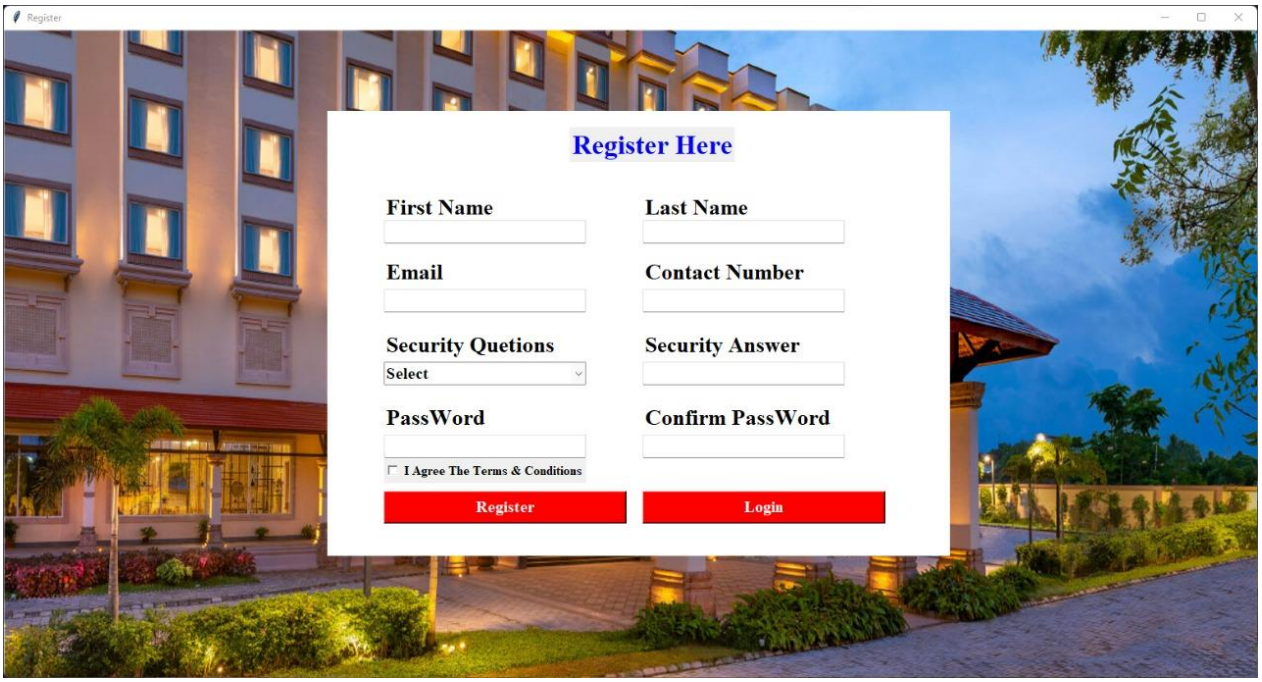

```python
from tkinter import *
from PIL import Image,ImageTk
from tkinter import ttk
from employee import Employee
from customer import Cust_win
from details import DetailsRoom
from room import RoomBooking
class hotel_class:
    def __init__(self,root):
        self.root=root
        self.root.title("Hotel Management System")
        self.root.geometry("1550x800+0+0")
        self.var_hotel=StringVar()
        self.open()
    def open(self):

        lbl_title=Label(self.root,text="HOTEL MANAGEMENT SYSTEM",font=("times new roman",40,"bold"),bg="black",fg="gold",bd=4,relief=RIDGE)
        lbl_title.place(x=0,y=0,width=1550,height=50)

        main_frame=Frame(self.root,bd=4,relief=RIDGE)
        main_frame.place(x=0,y=60,width=1550,height=820)
        img2=Image.open(r"Hotel_Management_System_DBMS_MINI_Project\images\background.jpg")
        img2=img2.resize((1550,820),Image.ANTIALIAS)
        self.photoimage2=ImageTk.PhotoImage(img2)

        labelImg2=Label(main_frame,image=self.photoimage2,bd=4,relief=RIDGE)
        labelImg2.place(x=0,y=0,width=1550,height=820)

        btn_frame=Frame(main_frame,bg="white")
        btn_frame.place(x=610,y=170,width=250,height=250)

        # btn_frame=Frame(main_frame,bd=4,relief=RIDGE)
        # btn_frame.place(x=0,y=205,width=700,height=190)

        lbl_cust_gender=Label(btn_frame,text="Hotel",font=("arial",28,"bold"),padx=2,pady=6,bg="white",fg="black")
        lbl_cust_gender.place(x=80,y=50)

        combo_gender=ttk.Combobox(btn_frame,font=("arial",15,"bold"),textvariable=self.var_hotel,width=10,state="read only")
        combo_gender["value"]=(1,2,3)
        combo_gender.current(0)
        combo_gender.place(x=60,y=110)
        submit_btn=Button(btn_frame,text="Open",width=16,font=("times new roman",18,"bold"),bg="#bdbb84",fg="black",cursor="hand1",command=self.hotel)
        submit_btn.place(x=10,y=150)
        exit_btn=Button(btn_frame,text="Exit",width=16,font=("times new roman",18,"bold"),bg="#bdbb84",fg="black",cursor="hand1",command=lambda :exit(0))
        exit_btn.place(x=10,y=200)


    def hotel(self):
```

CHAPTER 4
RESULTS AND SNAPSHOTS

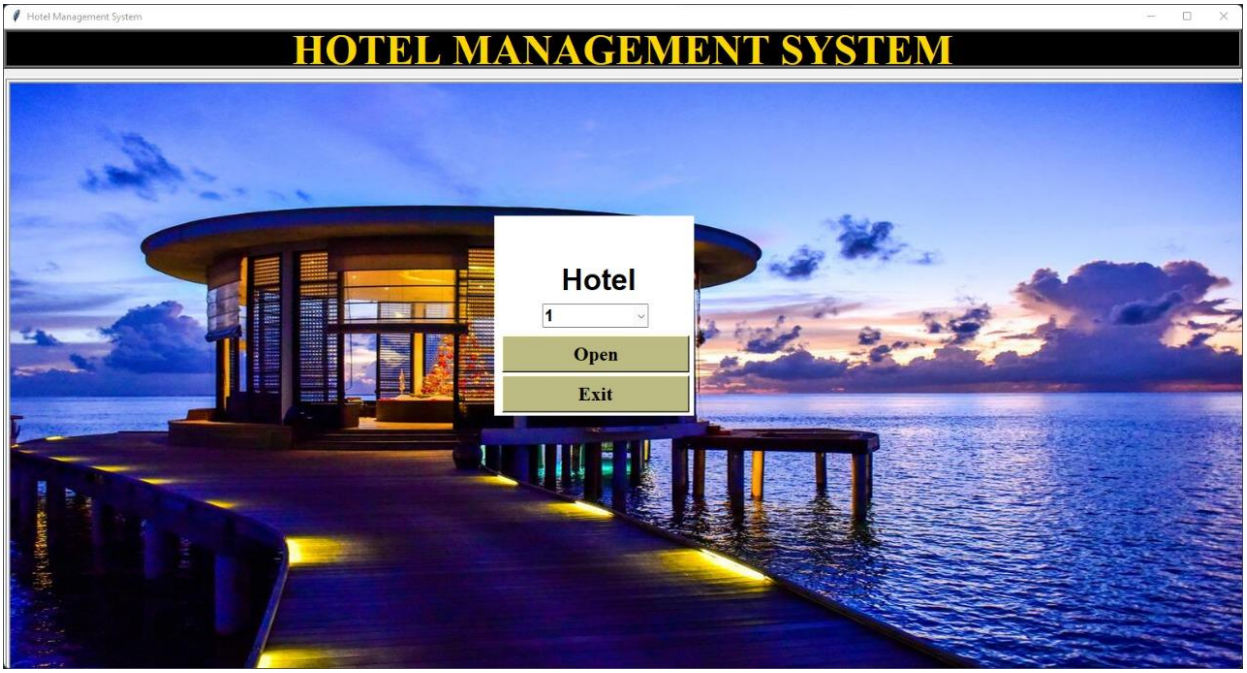
LOGIN PAGE



REGISTER PAGE

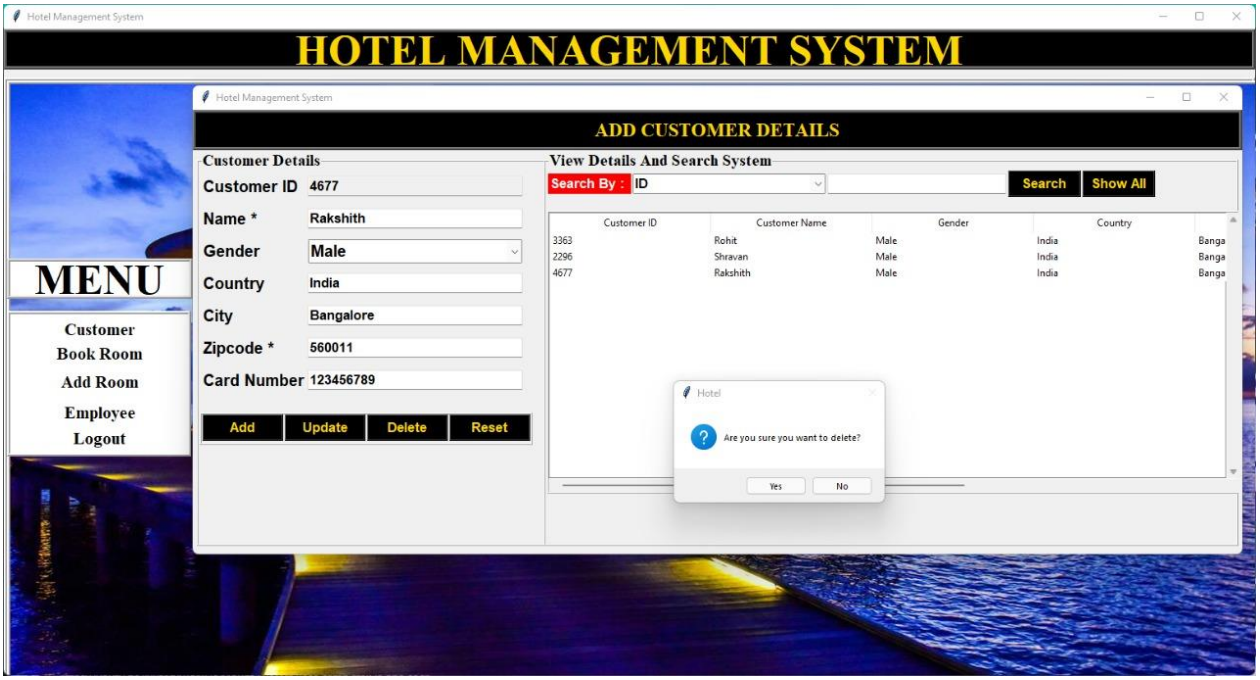


MAIN PAGE

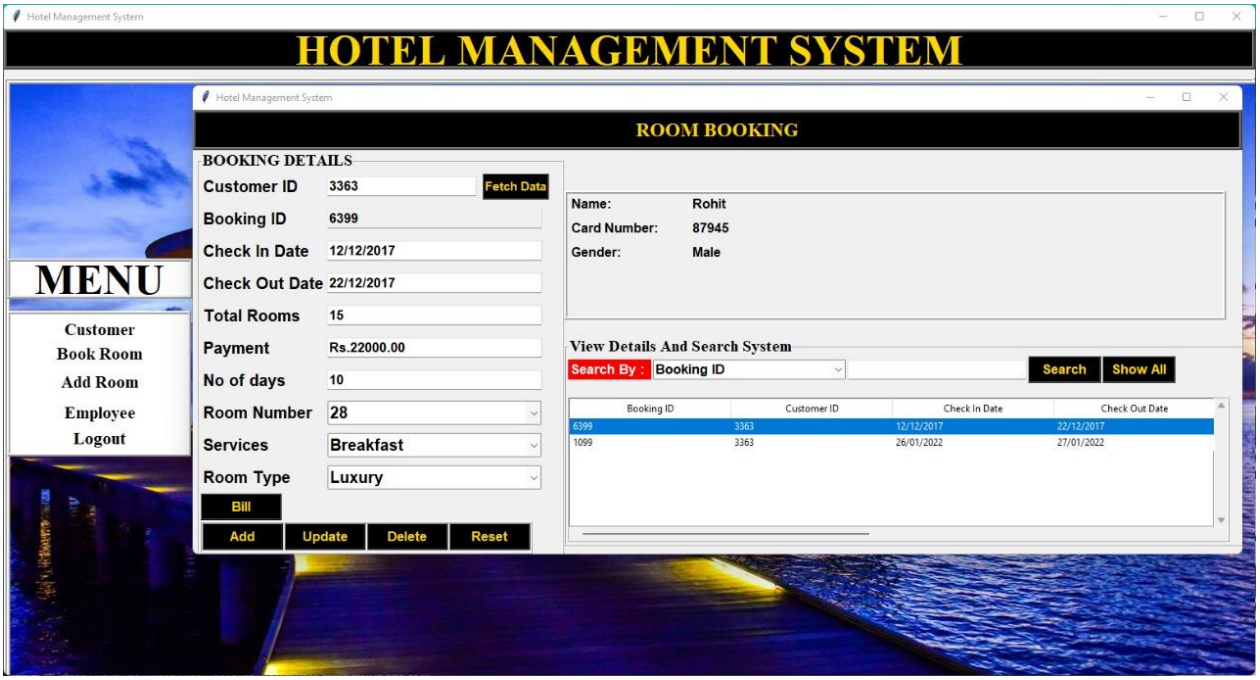

Hotel Management System
HOTEL MANAGEMENT SYSTEM
Hotel
1
Open
Exit


Hotel Management System
HOTEL MANAGEMENT SYSTEM
MENU
Customer
Room
Details
Employee
Logout

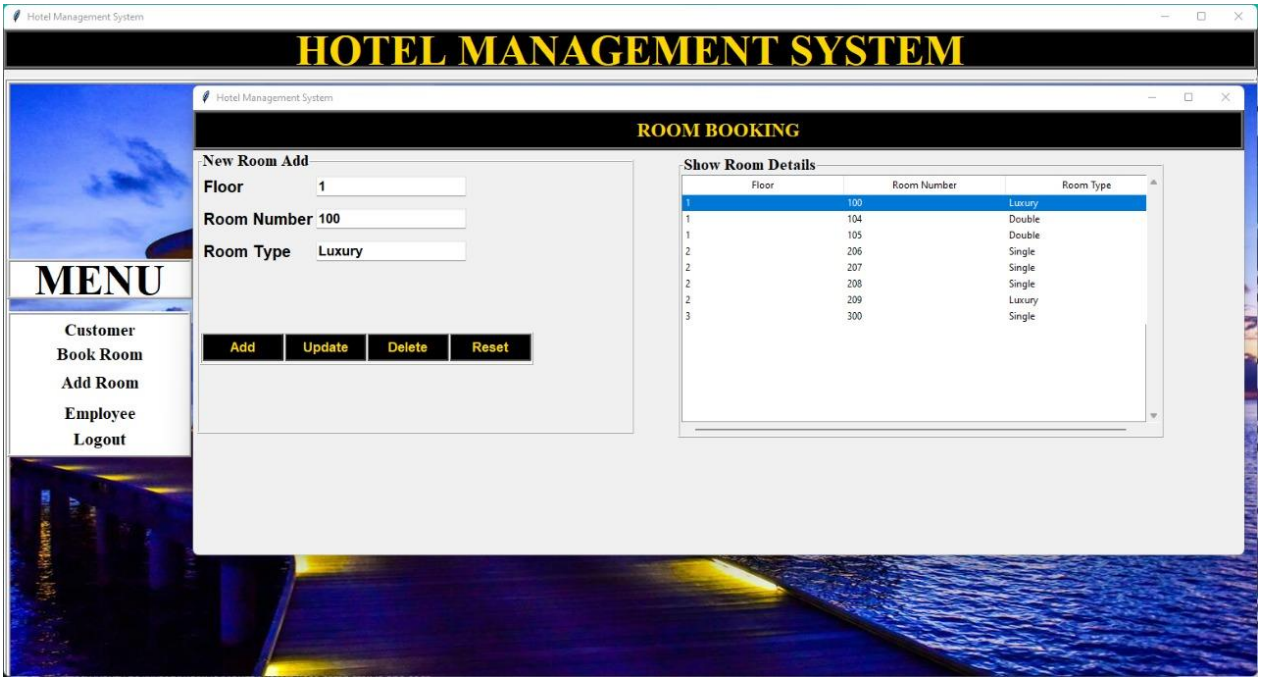
CUSTOMER DETAILS



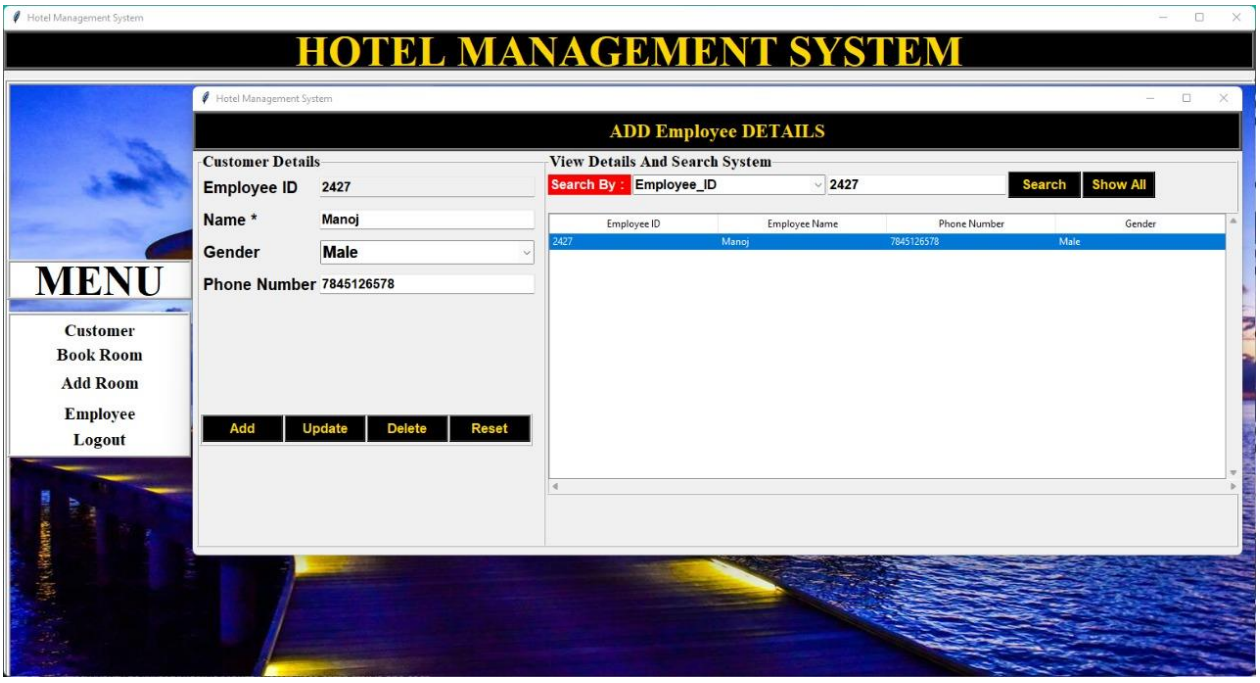
ROOM BOOKING DETAILS



ADDING ROOM



EMPLOYEE DETAILS



CONCLUSION

The project “Hotel Management System” is aimed to develop to maintain the day-to-day state of admission/Vacation of Residents, List of Workers , payment details etc. Main objective of this project is to provide solution for hotel to manage most there work using computerized process. This software application will help admin to handle customers information, room allocation details, payment details, billing information.etc

The existing system is a manually maintained system. All the Hotel records are to be maintained for the details of each customers, Fee details, Room Allocation , Attendance etc. All these details are entered and retrieved manually,because of this there are many disadvantages like Time Consuming ,updating process, inaccuracy of data.For avoiding this we introduced or proposed a new system in proposed system the computerized version of the existing system. provides easy and quick access over the data.

References

Python
https://www.python.org/

Tkinter

https://docs.python.org/3/library/tkinter.html

MySQL Database https://www.mysql.com/downloads/