**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.No** | **CONTENTS** | **SIGN.** |
| **1.** | **Introduction** |  | |
| **2.** | **System Description** |  | |
| **3** | **Architectural Design** |  | |
| **4** | **Project Design** |  | |
| **5** | **Description of Modules** |  | |
| **6** | **FlowChart** |  | |
| **7** | **Program Source Code**  **&**  **Screen Shots** |  | |

**Introduction**

Today, be it a company, institution or a non-governmental organization, for the smooth running of them all, we need a backup of data, so that vital information can be stored, preserved, retrieved and used as per the requirements. Companies employ what they call computer operators who store and handle the database which stores all necessary information concerning the company. What is much needed is a proper interface which may enable us to handle such systems efficiently.

This project entitled “Hotel Management System” has been developed with an aim to help one such organization – a hotel to easily manage and handle the data concerning it. It has the following objectives:

* **Automation:**

The Hotel Management System automates each and every activity of the manual system and increases its throughput. Thus the response time of the system is very less and it works very fast.

* **Accuracy:**

The Hotel Management System provides the user a quick response with very accurate information regarding the users etc. Any details on system in an accurate manner, as and when required.

* **User-Friendly:**

The software Hotel Management System has a very user-friendly interface. Thus the users will feel very easy to work on it. The software provides accuracy along with a pleasant interface.

* **Maintenance Cost:**

The Hotel Management System reduces the cost of maintenance.

**System Description**

Hotel Management System is used to maintain the records regarding the Booking Information of guests (Check in/Check out) ,Dinner Reservation, and Information regarding its employees.

The project implements a complete Hotel Management System procedure which saves all the data concerning a Hotel. It makes some computerized facilities in the Hotel to store records.

Here the main users of the system are the members of the Hotel Employees.

*The project is mainly divided into two modules, namely –*

*Guest – Details such as the name, booking number, phone number, number of rooms, type of rooms etc can stored and retrieve.*

*Staff– details such as contract of the employees , date of joining , address etc can be viewed.*

There are five types of actions performed by this project –

* **Main** – Give user the choice to select between Staff and Guest.
* **Booking Information**– Enables the user to retrieve the information about the customer.
* **Check in**– Enables the user to enter the information of the guest on its arrival.
* **Employee Access**– Enables the use to see the records of the employee(eg-name, contract, designation etc) stored in the hotel database.
* **Reservation**–Customers can book dinner reservations at the restaurant and also enables the user to check already booked reservations .

**Architectural Design**

**STAFF**

* STORES,RETRIEVE,EDITS DETAILS OF Guest at the hotel
* Shows the details regarding the employees of the hotel
* Makes reservation at the restaurant

**DATABASE**

**Hotel**

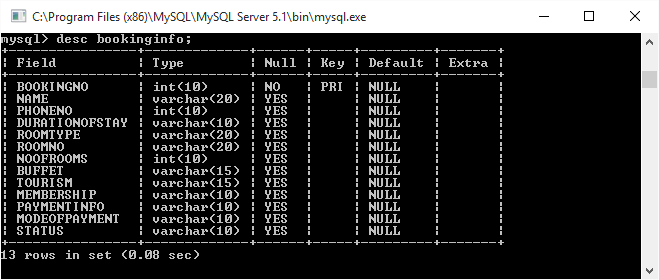
**Project Design Structure**

* **DATABASE**

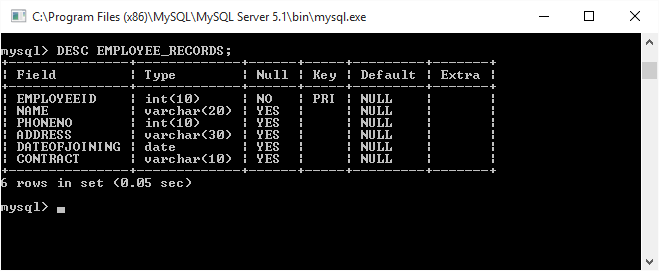
|  |  |
| --- | --- |
| **Name** | **TABLES** |
| **RDS** | BookingInfo |
| Employee\_Records |
| Restaurant |

* **TABLES/RELATIONS**

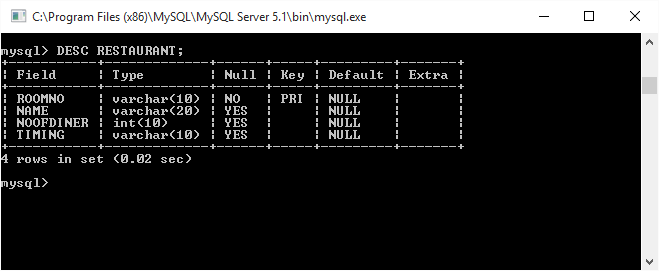
**BookingInfo**



**EMPLOYEE\_RECORDS**



**Restaurant**



**Description of Modules**

As mentioned above, the project is divided broadly into three modules, but being more specific, following are the modules (with their detailed descriptions) –

* **Login Module** - Obviously, this module does nothing but puts up the demand of the username and password of the administrator. (It is to be noted that in case of this project, the username is the name of the employee and password is”123456” in the first login module which is required to access the Management System. The second Login Module is used to give access to specific people only. The user name in the second login module in”Tridentgroups@Gurgaon”and the password is ‘Trident’) Upon correct input of the username and the password, it redirects the administrator to the main page o from where one can easily navigate across the entire project.
* **Record Insertion Module** - The primary function of this project is to store data. To begin with, the injection module handles the insertion of data. Broadly there are Two modules – CheckIn, and Reservation. The injection module is designed in such as way that it keeps the insertion of data ‘clean’.

Using this module, the staff themselves or whoever for the matter is administering the database can add / insert data concerning CheckIn and Reservation.

* **Record Retrieval Module** - This module has been created for the sole purpose of clean removal of existing data, namely the booking Information of the guests(eg-Name,Phoneno,Payment information etc) , dinner reservation in restaurant and employee information(eg-name, address etc.).
* **Viewer Module** - This module has been designed to view data concerning the Booking Information and Dinner Reservation of a given Hotel’s database.

For the sake of convenience and better user-application-interface experience, this module has been categorized into Booking Information, CheckIn and Employee records and has been designed to serve data in a tabular method.

* **Deletion Module** - This module has been designed to delete data concerning the Student, teachers and staff of a given School’s database based on the search criteria put by the user.
* **Search Module** - Easily fathomable, this module helps the administrator to search for a particular guest’s information (eg-name, phone number, payment information, mode of payment etc) or an employee;s information (eg-name, address, phone number)conveniently. All he/she needs to do is enter either the Booking Number Of the guest,or employeeid of the employee . The module will consult the database and the respective table and drag identifiable data, if it exists.

**Source Code & Screenshots**

print(""" \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

^\_^                                              WELCOME TO XYZ HOTEL                                                                         ^\_^

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

""")

import mysql.connector as sqlhotel

mycon=sqlhotel.connect(host="localhost",user="root",password="CLASS12khyati@@")

mycursor=mycon.cursor()

if mycon.is\_connected():

    print("SUCCESSFULLY CONNECTED")

mycursor.execute("CREATE DATABASE IF NOT EXISTS hotel")

mycursor.execute("use hotel")

ROOMRENT=0

rbill=0

lbill=0

total=0

CID=""

def Customer\_Details():

    global CID

    mycursor.execute("create table if not exists CUSTOMER\_DETAILS(CID VARCHAR(20),NAME VARCHAR(30),CONTACT\_NUMBER VARCHAR(50),ADDRESS VARCHAR(30), EMAIL VARCHAR(30))")

    CID = input("Enter Customer Identification Number : ")

    NAME = input("Enter Customer Name : ")

    CONTACT\_NUMBER= input("Enter Customer Contact Number : ")

    ADDRESS= input("Enter Customer Address : ")

    EMAIL= input("Enter Customer Email : ")

    QUERY="INSERT INTO CUSTOMER\_DETAILS VALUES('{}','{}','{}','{}','{}')".format(CID,NAME,CONTACT\_NUMBER,ADDRESS,EMAIL)

    mycursor.execute(QUERY)

    mycon.commit()

    print("RECORDED THE DETAILS OF THE NEW CUSTOMER")

    mycursor.execute("CREATE TABLE IF NOT EXISTS BOOKING\_DETAILS(CID VARCHAR(20),CHECK\_IN varchar(20),CHECK\_OUT varchar(20))")

    CHECK\_IN=input("Enter Customer CheckIN Date [ YYYY-MM-DD ] : ")

    CHECK\_OUT=input(" Enter Customer CheckOUT Date [ YYYY-MM-DD ] : ")

    QUERY= "INSERT INTO BOOKING\_DETAILS VALUES({},'{}','{}')".format(CID,CHECK\_IN,CHECK\_OUT)

    mycursor.execute(QUERY)

    mycon.commit()

    print("RECORDED THE DETAILS OF CUSTOMER COMING IN AND OUT")

def  room\_Rent():

    global CID

    global ROOMRENT

    mycursor.execute("create table if not exists ROOM\_RENT\_DETAILS(CID VARCHAR(20),ROOM\_CHOICE VARCHAR(10),NO\_OF\_DAYS VARCHAR(20),ROOM\_NUMMBER VARCHAR(10),ROOMRENT VARCHAR(25))")

    print ("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* TYPES OF ROOMS AVAILABLE IN OUR HOTEL \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

    print (" 1. type A----> 15000 Rs.")

    print (" 2. type B----> 7000 Rs. ")

    print (" 3. type C---> 5500 Rs. ")

    print (" 4. type D ----> 3500 Rs. ")

    ROOM\_CHOICE =int(input("Enter Option from the above options : "))

    NO\_OF\_DAYS=int(input("Enter No. Of Days : "))

    ROOM\_NUMBER=int(input("Enter Customer Room No : "))

    if ROOM\_CHOICE==1:

        ROOMRENT = NO\_OF\_DAYS \* 15000

        print("type A Room Rent : ",ROOMRENT)

    elif ROOM\_CHOICE ==2:

        ROOMRENT = NO\_OF\_DAYS \* 7000

        print(" type B Room Rent : ",ROOMRENT)

    elif ROOM\_CHOICE ==3:

        ROOMRENT = NO\_OF\_DAYS \* 5500

        print(" type C Room Rent : ",ROOMRENT)

    elif ROOM\_CHOICE==4:

         ROOMRENT= NO\_OF\_DAYS \* 3500

         print(" type D  Room Rent : ",ROOMRENT)

    else:

        print("PLEASE ENTER A OPTION FROM THE ABOVE PROVIDED !!! ")

    QUERY="INSERT INTO ROOM\_RENT\_DETAILS VALUES({},{},{},{},{})".format(CID,ROOM\_CHOICE,NO\_OF\_DAYS,ROOM\_NUMBER,ROOMRENT)

    mycursor.execute(QUERY)

    mycon.commit()

    print("Your Room Has Been Booked For : ",NO\_OF\_DAYS , "Days" )

    print("Your Total Room Rent is : Rs. ",ROOMRENT)

    return ROOMRENT

def restaurant\_menu():

   print("1. Vegetarian Combo ----->Rs.450")

   print("2. Non-Vegetarian Combo ----->Rs750")

   print("3. Vegetarian & Non-Vegetarian Combo ----->RS.1500")

   print("4. Tea ----->Rs.20")

   print("5. Coffee ----->Rs.30")

   print("6. Juice ----->Rs.50")

   print("7. Snacks ----->Rs.100")

   print("8. Desserts ----->Rs.150")

   print("9. Chinese food ----->Rs.100")

   print("10. Sandwich ----->Rs.50")

def order\_food():

    global CID

    global rbill

    mycursor.execute("create table if not exists RESTAURANT(CID VARCHAR(20),FOOD\_ITEM VARCHAR(30),QUANTITY VARCHAR(30),BILL VARCHAR(30))")

    Desired\_dish= int(input("Enter Your food\_item : "))

    Quantity=int(input("Enter Quantity : "))

    if Desired\_dish==1:

            print(" ORDER: Vegetarian Combo ")

            rbill = Quantity \* 450

    elif Desired\_dish==2:

            print(" ORDER: Non-Vegetarian Combo ")

            rbill =Quantity \* 750

    elif Desired\_dish==3:

            print(" ORDER: Vegetarian & Non-Vegetarian Combo ")

            rbill= Quantity \* 1500

    elif Desired\_dish==4:

            print(" ORDER: Tea ")

            rbill= Quantity \* 20

    elif Desired\_dish==5:

            print(" ORDER: Coffee ")

            rbill= Quantity \* 30

    elif Desired\_dish==6:

            print(" ORDER: Juice")

            rbill= Quantity \* 50

    elif Desired\_dish==7:

            print(" ORDER: Snacks")

            rbill= Quantity \* 100

    elif Desired\_dish==8:

            print(" ORDER: Desserts")

            rbill= Quantity \* 150

    elif Desired\_dish==9:

            print(" ORDER: Chinese food")

            rbill= Quantity \* 100

    elif Desired\_dish==10:

            print(" ORDER: Sandwich")

            rbill= Quantity \* 20

    else:

            print("PLEASE ENTER A OPTION FEOM THE MENU")

    mycursor.execute("INSERT INTO RESTAURANT VALUES({},'{}','{}','{}')".format(CID,Desired\_dish,Quantity,rbill))

    mycon.commit()

    print(" Total Bill Amount Is : Rs.",rbill)

    return rbill

def  laundry\_price():

     print("""1.trouser ----->Rs20

2. Shirt ----->Rs20

3. Saree----->Rs50

4.Suit-------->Rs40

5. Pant------> Rs 30

""")

def laundry\_bill():

        global CID

        global lbill

        c=input("do u want to continue:(y/n)")

        while c=="Y" or c=="y":

             laundry=int(input("enter laundry no. "))

             qty=int(input("enter quantity of laundry"))

             if laundry==1:

                   print("1. trouser")

                   lbill=qty\*20

                   print("your laundry bill is: Rs",lbill)

                   return lbill

             elif laundry==2:

                   print("2.Shirt")

                   lbill=qty\*20

                   print("your laundry bill is: Rs",lbill)

                   return lbill

             elif laundry==3:

                    print("3.Saree")

                    lbill=qty\*50

                    print("your laundry bill is: Rs",lbill)

                    return lbill

             elif laundry==4:

                   print('4.Suit')

                   lbill=qty\*40

                   print("your laundry bill is: Rs",lbill)

                   return lbill

             elif laundry==5:

                   print('5.pant')

                   lbill=qty\*30

                   print("your laundry bill is: Rs",lbill)

                   return lbill

             else:

                print("Invalid choice")

             c=input("do u want to continue:(Y/N)")

def Total\_bill():

       global CID

       global total

       global rbill

       global lbill

       global ROOMRENT

       total=ROOMRENT+rbill+lbill

       print("---------------------------------------")

       print(" roomrent        =Rs",ROOMRENT)

       print("restaurant bill=Rs",rbill)

       print("laundry bill      =Rs",lbill)

       print("---------------------------------------")

       print("your total bill is: Rs",total,"^\_^")

def menu():

       print(""" 1. customer details

2. Room Rent

3. Restaurant menu

4.  Restaurant Bill

5. View Laundry Price

6.  Laundry Bill

7. Total Bill

8. Exit""")

       c="y"

       while True:

           try:

               ch=int(input("enter your choice(1-6)"))

           except :

                   exit("\n enter a no.")

           if ch==1:

            Customer\_Details()

           elif ch==2:

             room\_Rent()

           elif ch==3:

              restaurant\_menu()

           elif ch==4:

              order\_food()

           elif ch==5:

               laundry\_price()

           elif ch==6:

              laundry\_bill()

           elif ch==7:

                  Total\_bill()

           elif ch==8:

               break

menu()