HOTEL MANAGEMENT SYSTEM

NEHAL MAHAJAN
YUVRAJ SINGH

CONTENTS

1.INTRODUCTION

1.1 Introduction to the problem area:

The hospitality industry, particularly hotel management, involves multifaceted challenges in efficiently handling customer interactions, reservations, and feedback. Hotel Management System recognizes the need for a robust solution to streamline these processes, ensuring a seamless experience for both customers and hotel administrators.

1.2 Purpose of the Project:

The Hotel Management System aims to address the complexities associated with hotel management by providing an integrated platform. The primary purposes include:

- Simplifying customer registration and room booking processes.
- Implementing a rating and review system to gather valuable customer feedback.
- Allowing customers to select specific facilities for a personalized experience.
- Facilitating booking inquiries, updates, and cancellations.
- Enhancing overall operational efficiency and customer satisfaction.

1.3 Objective of the Project

The primary objectives of our project are:

- Creating an ER diagram and relational model with appropriate primary keys and having minimum redundancy using normalisation (upto BCNF).
- Develop a user-friendly customer registration system and room booking mechanism.
- Establish a comprehensive rating and review system.
- Create a facility selection feature for customers.
- Provide a platform for booking inquiries and booking cancellation process.

1.4 Find and Organize the Information Required (Requirement Analysis):

To achieve the project objectives, a thorough requirement analysis is essential which involves:

- Identifying key functionalities required for customer registration and booking(age should be greater than 18).
- Listing the facilities that customers can select during their stay.
- Outlining the information needed for booking inquiries and updates.
- Determining the criteria and policies for booking cancellations like refund percentage.

2.HARDWARE REQUIREMENT

2.1 Hardware tools:

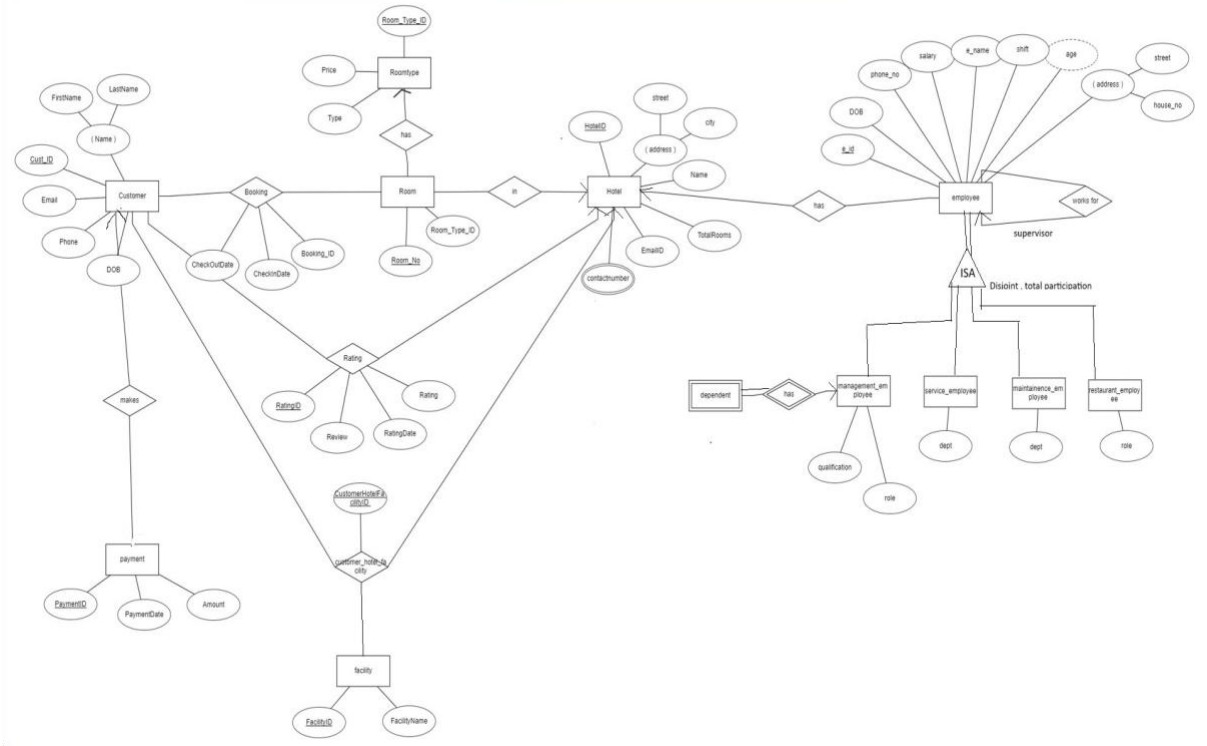
- Laptop/Personal Computer: A standard laptop or personal computer for development purposes.
- Internet Connection: Required for accessing online resources during development and testing.

2.2 Software tools:

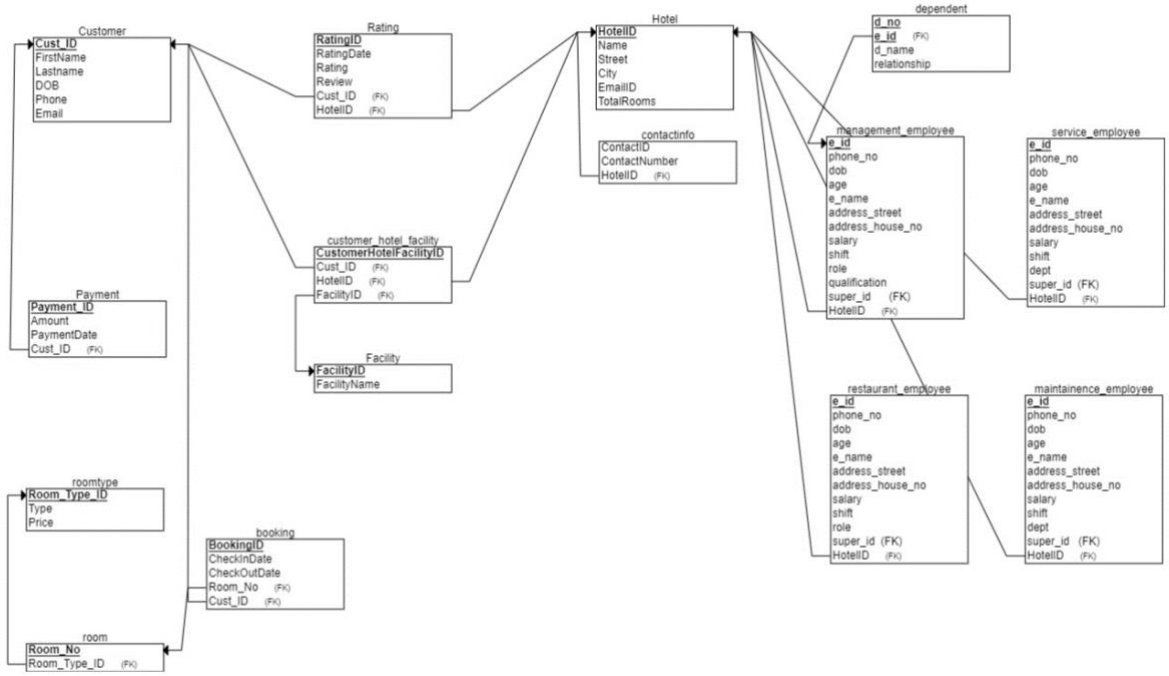
- Operating System: Linux/Unix or Windows Server for the central system.
- Web Server: Apache or Nginx for hosting the web-based components.
- Database Management System: MySQL or another relational database for storing customer and reservation data.
- Programming Language: PHP for server-side scripting to handle dynamic content.
- Client-Side Scripting: HTML, CSS, JavaScript for creating an interactive user interface.
- Development Environment: Code editor like Visual Studio Code, Sublime Text, or any preferred IDE.

3.Design

3.1 ER Model:



3.2 ER to Schema:



3.3 Functional Dependencies:

TABLE 1: CUSTOMER
Cust_ID -> FirstName, LastName, DOB, Phone, Email

TABLE 2: BOOKING
Booking_ID -> CheckInDate, CheckOutDate, room_no, Cust_ID

TABLE 3: HOTEL
HotelID -> Name, Street , City , EmailID, TotalRooms

TABLE 4: RATING
RatingID -> Rating, Review, RatingDate ,Cust_ID, HotelID

TABLE 5: PAYMENT
PaymentID -> Amount, PaymentDate, Cust_ID

TABLE 6: CONTACTINFO
ContactID -> ContactNumber ,HotelID

TABLE 7: FACILITY
FacilityID -> FacilityName

TABLE 8 :ROOM
Room_No -> Room_Type_ID

TABLE 9: ROOMTYPE
Room_Type_ID -> Type , Price

TABLE 10: CUSTOMER_HOTEL_FACILITY
CustomerHotelFacilityID -> Cust_ID, HotelID, FacilityID

TABLE 11: MANAGEMENT_EMPLOYEE
e_id -> e_name, phone_no, dob, age, address_street,
address_house_no, salary, shift, HotelID, super_id, qualification, role

TABLE 12: MAINTAINENCE_EMPLOYEE
e_id -> e_name, phone_no, dob, age, address_street,
address_house_no, salary, shift, HotelID, super_id, dept

TABLE 13: SERVICE_EMPLOYEE
e_id -> e_name, phone_no, dob, age, address_street,
address_house_no, salary, shift, HotelID, super_id, dept

TABLE 14: RESTAURANT_EMPLOYEE
e_id -> e_name, phone_no, dob, age, address_street,
address_house_no, salary, shift, HotelID, super_id, role

TABLE 15: DEPENDENTS
d_no ,management_employee.e_id -> d_name, relationship

3.4 Specify the primary key:

S.No.	TABLE	PRIMARY KEY
1.	CUSTOMER	Cust_ID
2.	BOOKING	Booking_ID
3.	HOTEL	HotelID
4.	CONTACTINFO	ContactID
5.	RATING	RatingID
6.	PAYMENT	PaymentID
7.	ROOM	Room_No
8.	ROOMTYPE	Room_Type_ID
9.	MANAGEMENT_EMPLOYEE	e_id
10.	SERVICE_EMPLOYEE	e_id
11.	RESTAURANT_EMPLOYEE	e_id
12.	MAINTAINANCE_EMPLOYEE	e_id
13.	DEPENDENTS	e_id, d_no
14.	FACILITY	FacilityID
15.	CUSTOMER_HOTEL_FACILITY	CustmerHotelFaclityID

3.5 Set-up the table relationship:

1.Unary relationship

S.NO.	TABLE	RELATIONSHIP
1.	Employee	Works for

2. Binary relationship

S.NO.	TABLE 1	RELATIONSHIP	TABLE 2
1.	Customer	makes	Payment
2.	Customer	booking	Room
3.	Customer	rating	Hotel
4.	Room	has	roomtype
5.	Room	in	Hotel
6.	Hotel	has	employee
7.	Management _employee	has	Dependent (weak entity)

3. Ternary relationship

S.NO	TABLE1	TABLE2	TABLE3	RELATIONSHIP
1.	Customer	Hotel	Facility	Customer_Hotel_Facility

4. Specialisation

Employee has 4 sub-groupings(disjoint and total participation) :
management_employee
maintenance_employee
service_employee
restaurant_employee

3.6 Refine your design by applying normalization rule till BCNF

1. ContactNumber corresponding to the Hotel was a multi-valued attribute . So, a separate table contactinfo was made to ensure 1NF.

2. All tables are normalised upto 1NF as each column in every table has atomic values.

3. All attributes other than primary key are fully dependent on the primary key which ensure that there are no partial dependencies making it 2NF.

4. Room :

Field	Type	Null	Key
Room_No	int	NO	PRI
Room_Type_ID	int	YES	MUL

Roomtype:

Field	Type	Null	Key
Room_Type_ID	int	NO	PRI
Type	varchar(50)	YES	
Price	decimal(10,2)	YES	

Here , if the attributes Type and price were part of room itself and roomtype table didn't exist, then the type and price would have been repeated multiple times in our room table which would have caused redundancy.

Also, in that case :

Room_No -> Room_Type_ID
Room_Type_ID ->Price
Room_Type_ID ->Type

Thus, to avoid such transitive dependencies , two different tables for room and roomtype should be made. This ensures 3NF.

5. A table is in BCNF if, and only if, for every non-trivial functional dependency (X → Y), X is a superkey.
   Hence, all our tables are normalised upto BCNF as they satisfy these conditions.

3.7 Natural join to prove lossless decomposition:

Lossless join decomposition is a decomposition of a relation R into relations R1, R2 such that if we perform a natural join of relation R1 and R2, it will return the original relation R.

Room :

Room_No	Room_Type_ID
101	1
103	1
105	1
108	1
110	1
102	2
106	2
109	2
104	3
107	3

Roomtype:

Room_Type_ID	Type	Price
1	Standard	2000.00
2	Deluxe	3500.00
3	Suite	5000.00

Room JOIN Roomtype:

Room_No	Room_Type_ID	Room_Type_ID	Type	Price
101	1	1	Standard	2000.00
103	1	1	Standard	2000.00
105	1	1	Standard	2000.00
108	1	1	Standard	2000.00
110	1	1	Standard	2000.00
102	2	2	Deluxe	3500.00
106	2	2	Deluxe	3500.00
109	2	2	Deluxe	3500.00
104	3	3	Suite	5000.00
107	3	3	Suite	5000.00

Here , the table we got after natural join of room and roomtype is the same table we had before normalisation of ROOM into room and roomtype. Hence , lossless join decomposition.

4. IMPLEMENTATION

4.1 XAMPP TABLE

TABLE 1: CUSTOMER

Cust_ID	FirstName	LastName	DOB	Phone	Email
1	Nehal	Mahajan	2005-02-28	12345	abcde
2	jhdk	dhwd	2005-04-27	973	hsh
3	ha	ha	2000-04-04	34	shrutivohra@gmail.com
4	yuvraj	singh	2005-01-01	8929944767	yuvraj@gmail.con
5	Shreya	Vohra	2005-01-01	987654321	shreya@gmail.com
6	Kirti	Verma	2001-01-01	7654321	kirtimehra@gmail.com

TABLE 2: BOOKING

Booking_ID	Cust_ID	Room_No	CheckInDate	CheckOutDate
2	2	101	2023-11-12	2023-11-14
4	1	108	2023-05-11	2023-05-13
5	2	108	2023-05-12	2023-05-14
6	2	108	2023-05-13	2023-05-15
7	1	108	2023-05-16	2023-05-18
8	2	102	2023-05-21	2023-05-23

TABLE 3: HOTEL

HotelID	Name	Street	City	EmailID	TotalRooms
1	Delight	Birch street	Delhi	delight@gmail.com	10

TABLE 4: RATING

RatingID	Cust_ID	HotelID	Rating	Review	RatingDate
1	2	1	2.00	clean	2020-03-03
2	1	1	5.00	good	2023-04-04
3	1	1	5.00	nice	2023-09-11
4	3	1	4.80	Delicious food.	2023-11-05
5	4	1	5.00	Best hotel in town.	2023-11-02
6	5	1	4.70	Good place	2023-11-07
7	10	1	4.60	Good hotel, I loved the greenery.	2023-11-11
8	10	1	4.90	I loved it.	2023-11-12
9	14	1	5.00	Best best hotel	2023-11-11
10	15	1	5.00	very very good	2023-11-11

TABLE 5: PAYMENT

PaymentID	Cust_ID	Amount	PaymentDate
1	2	600.00	2020-04-09
2	2	400.00	2015-04-06
3	2	600.00	2014-04-07
4	2	200.00	2020-05-07
5	1	200.00	2013-04-06
6	1	500.00	2017-03-05
7	1	500.00	2023-07-03
8	1	200.00	2023-08-03
9	5	500.00	2023-12-27
10	10	200.00	2023-11-30

TABLE 6: CONTACTINFO

ContactID	HotelID	ContactNumber
1	1	111-222-333
2	1	999-888-777

TABLE 7: FACILITY

FacilityID	FacilityName
1	Airport Shuttle
2	Live Karaoke
3	Swimming Pool
4	Gym
5	Sports Centre
6	Restaurant
7	Kids Play Area
8	Laundry Service
9	Free Wi-Fi

TABLE 8 :ROOM

Room_No	Room_Type_ID
101	1
103	1
105	1
108	1
110	1
102	2
106	2
109	2
104	3
107	3

TABLE 9: ROOMTYPE

Room_Type_ID	Type	Price
1	Standard	2000.00
2	Deluxe	3500.00
3	Suite	5000.00

TABLE 10: CUSTOMER_HOTEL_FACILITY

CustomerHotelFacilityID	Cust_ID	HotelID	FacilityID
1	1	1	1
2	2	1	1
3	2	1	2
4	2	1	3
5	1	1	1
6	1	1	3
7	1	1	4
8	5	1	2
9	5	1	3
10	5	1	6
11	10	1	1
12	10	1	3

TABLE 11: MANAGEMENT_EMPLOYEE

e_id	phone_no	dob	age	e_name	address_street	address_house_no	salary	shift	HotelID	super_id	qualification	role
1	1234567890	1980-01-15	44	John Doe	789 Oak St	Apt 301	60000	Morning	1	NULL	MBA	Manager
2	9876543210	1975-08-22	48	Jane Smith	456 Pine St	Apt 201	55000	Morning	1	1	BSC	Assistant Manager
3	8765432109	1990-03-10	34	Robert Johnson	123 Elm St	Apt 401	48000	Evening	1	NULL	MBA	Supervisor

TABLE 12: MAINTAINENCE_EMPLOYEE

e_id	phone_no	dob	age	e_name	address_street	address_house_no	salary	shift	HotelID	super_id	dept
10	4567890123	1985-09-12	38	David Wilson	789 Oak St	Apt 302	32000	Morning	1	NULL	Maintenance Incharge
11	5678901234	1990-04-25	34	Sophie Taylor	456 Pine St	Apt 202	[illegible]	Morning	1	10	Housekeeping
12	6789012345	1988-07-07	35	Lucas Harris	123 Elm St	Apt 403	[illegible]	Evening	1	10	Housekeeping

TABLE 13: SERVICE_EMPLOYEE

e_id	phone_no	dob	age	e_name	address_street	address_house_no	salary	shift	HotelID	super_id	dept
7	1234567890	1992-12-03	31	Michael Lee	123 Main St	Apt 102	40000	Morning	1	NULL	Services Manager
8	2345678901	1997-07-18	26	Sophia Martinez	456 Pine St	Apt 301	35000	Full Day	1	7	Swimming and Gym
9	3456789012	1993-02-25	31	James Johnson	123 Elm St	Apt 402	21000	Morning	1	7	Gardening

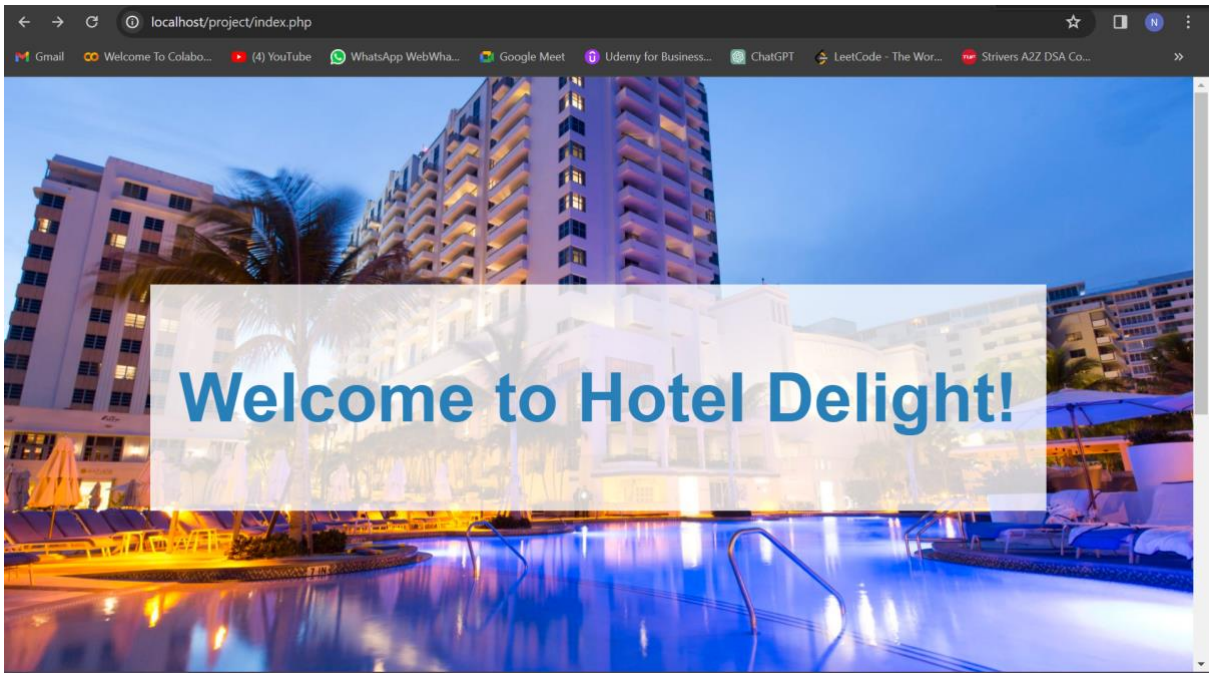
TABLE 14: RESTAURANT_EMPLOYEE

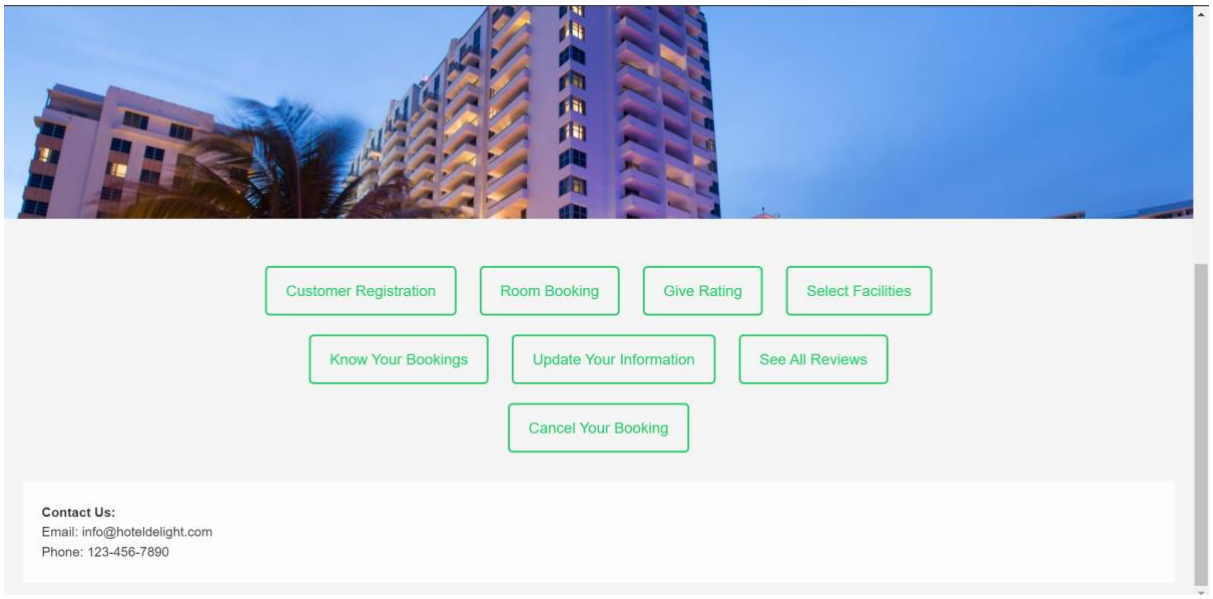
e_id	phone_no	dob	age	e_name	address_street	address_house_no	salary	shift	HotelID	super_id	role
4	2345678901	1995-05-20	28	Emily Davis	123 Main St	Apt 101	26000	Morning	1	NULL	Chef
5	7890123456	1998-11-30	25	William White	456 Elm St	Apt 202	10000	Morning	1	4	Waiter
6	8901234567	2000-07-15	23	Olivia Brown	789 Oak St	Apt 303	25000	Evening	1	NULL	Chef

TABLE 15: DEPENDENTS

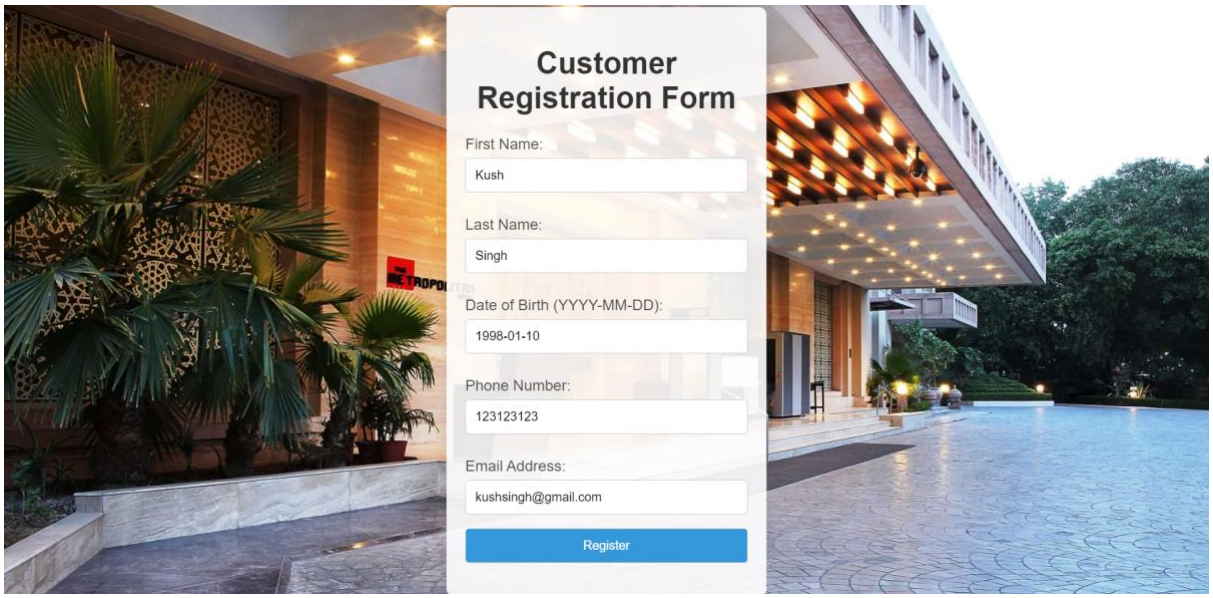
d_no	management_e_id	d_name	relationship
1	1	Sarah Doe	Spouse
1	2	Ethan Smith	Child
1	3	Ava Johnson	Child
2	3	Mia Johnson	Child

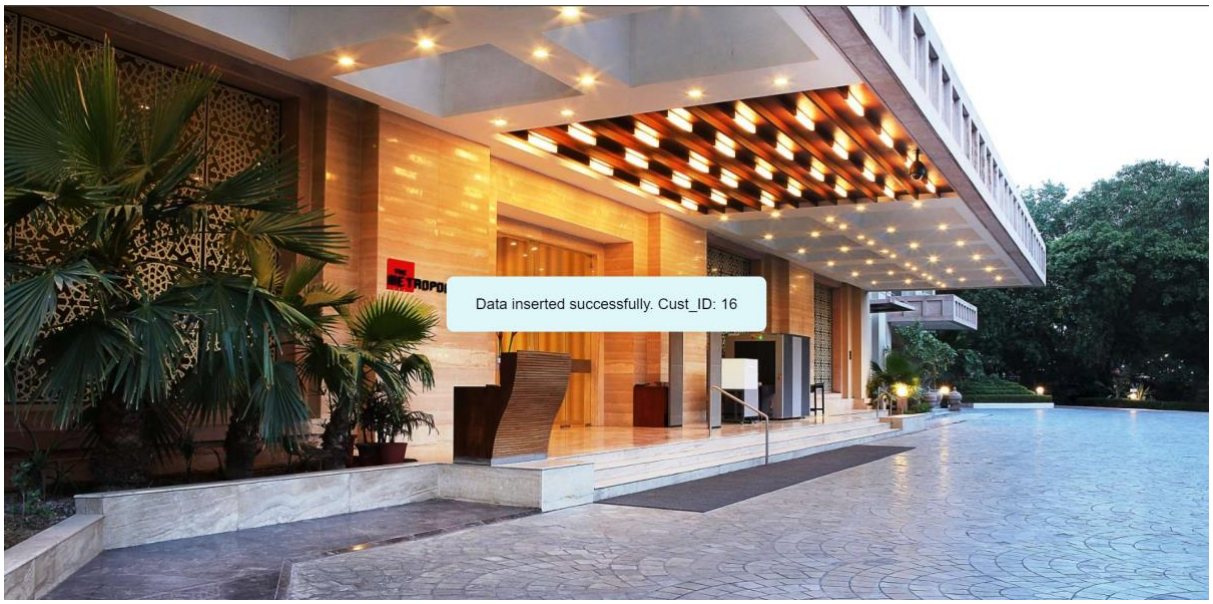
4.2 Snapshots of running of application:



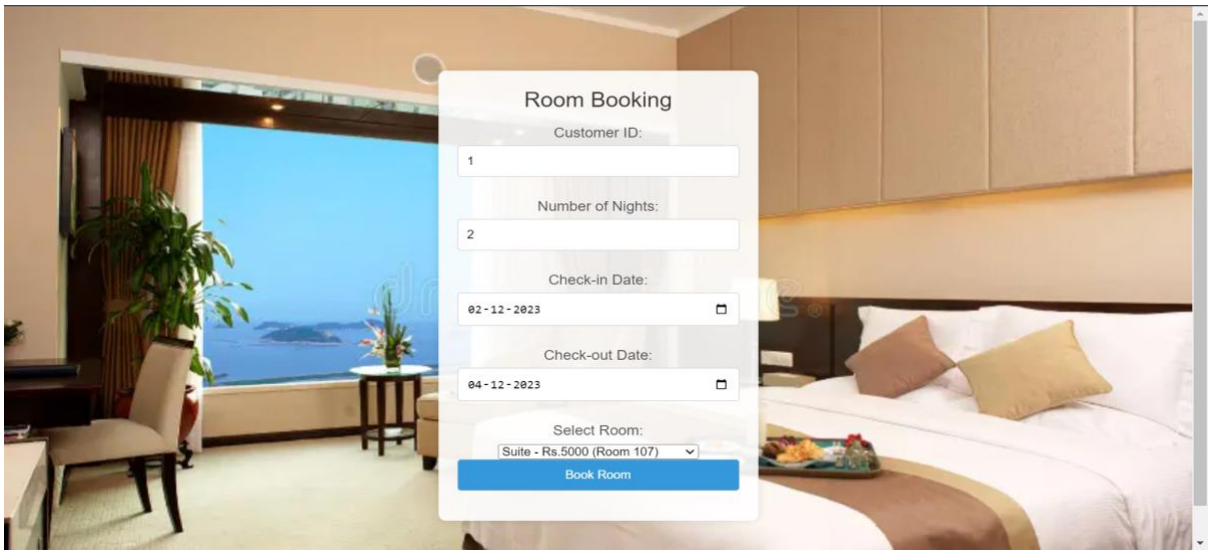


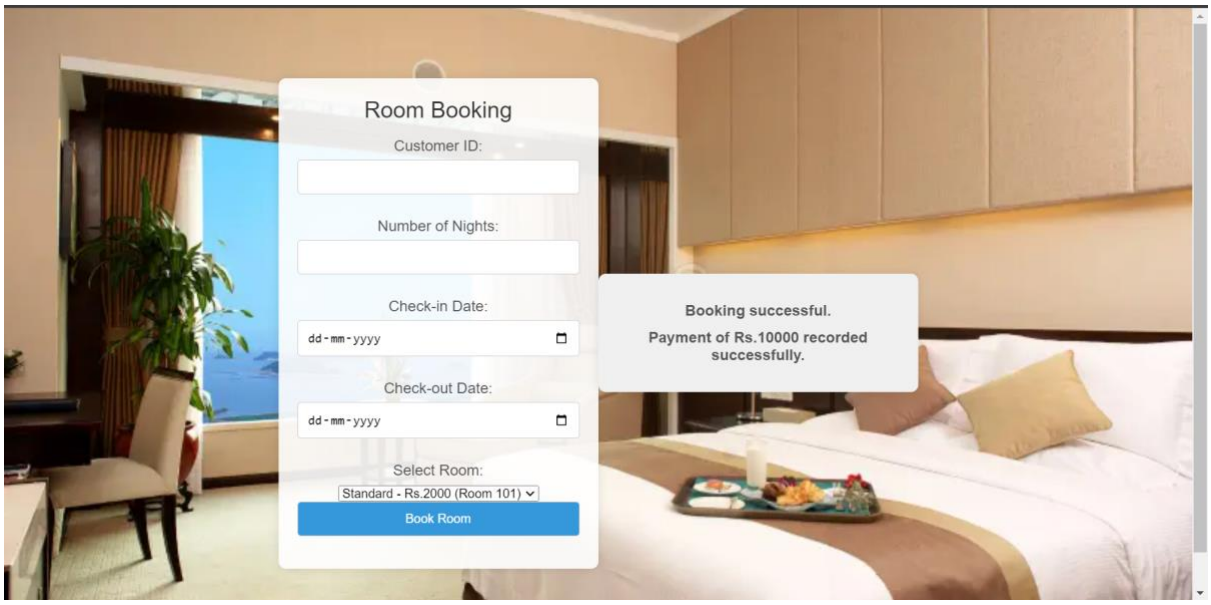
1. Customer Registration



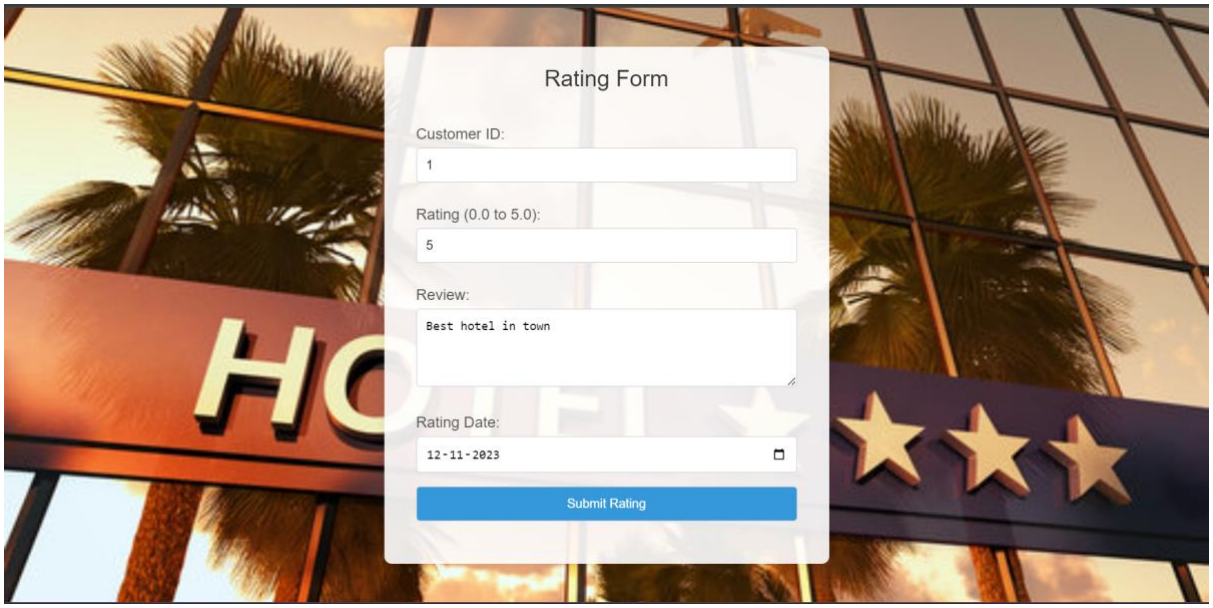


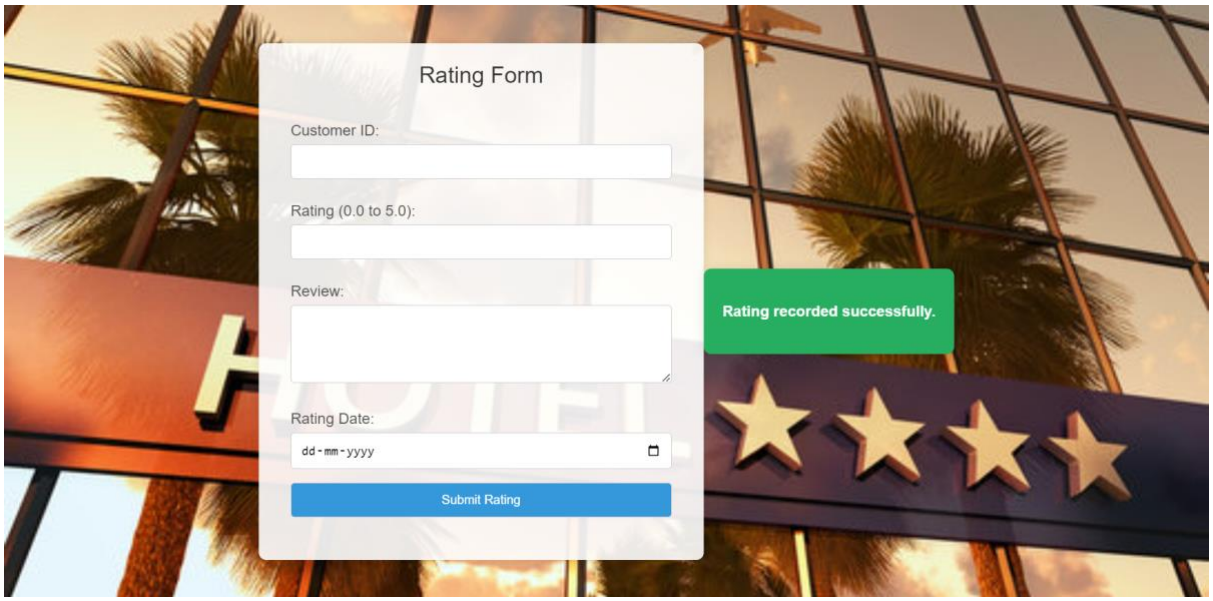
2.Room Booking



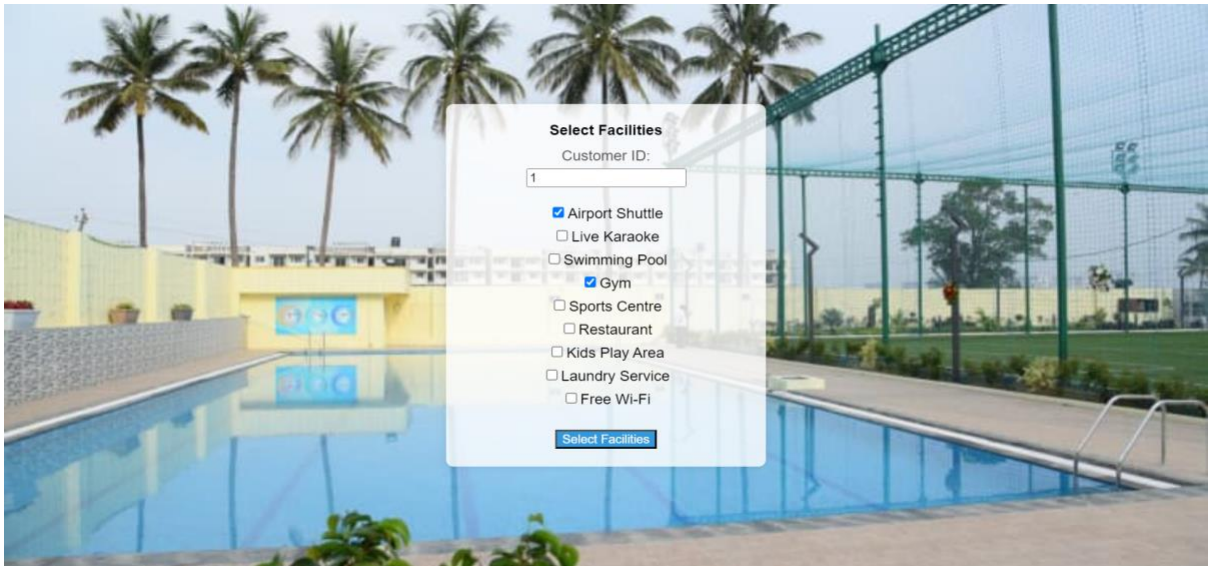


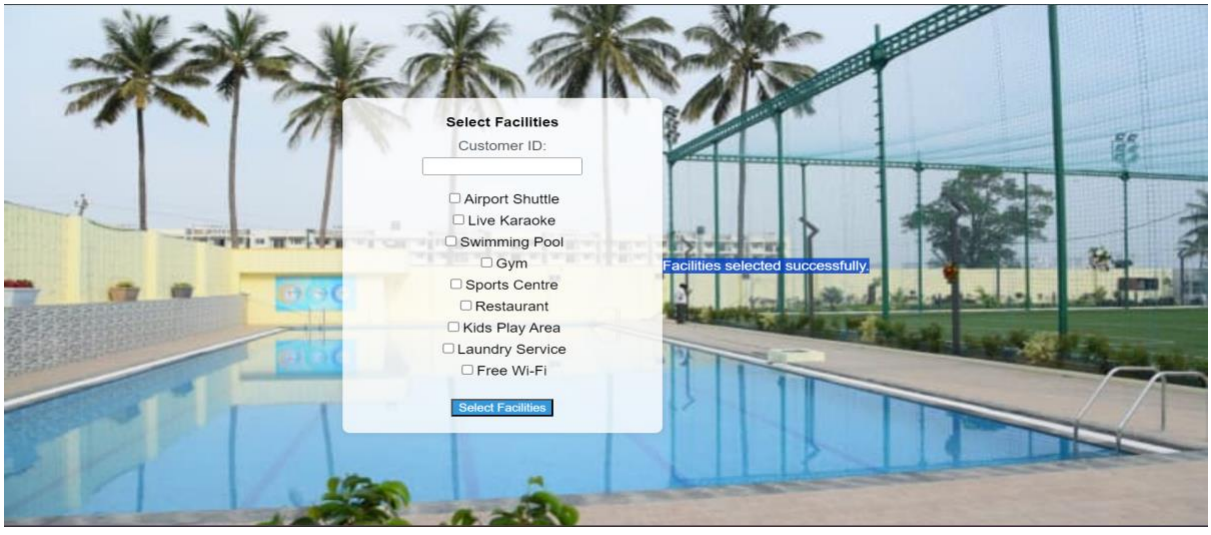
3.Give rating



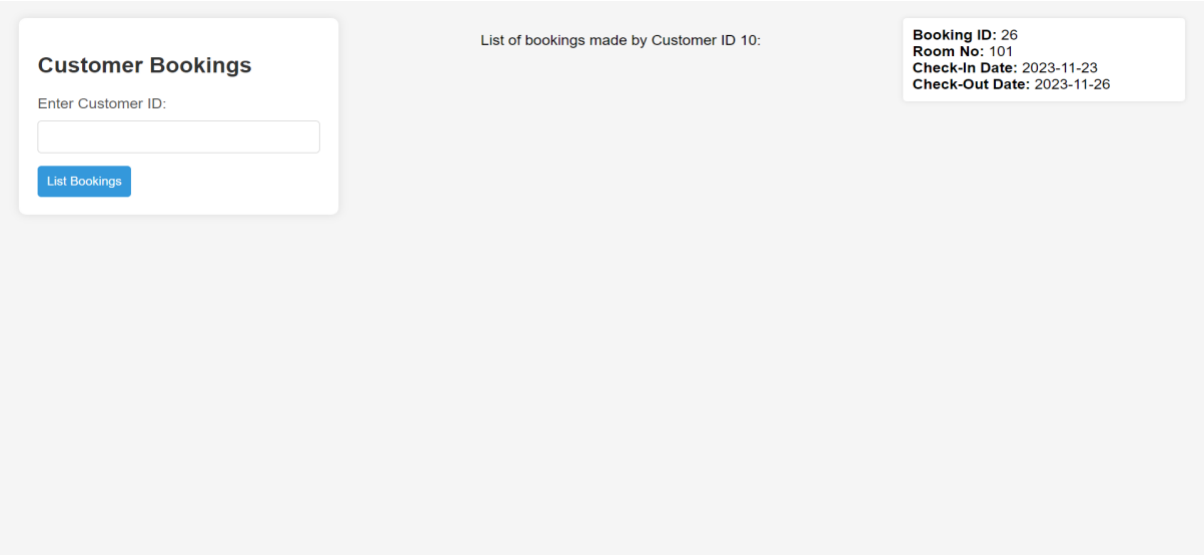


4.Select facilities

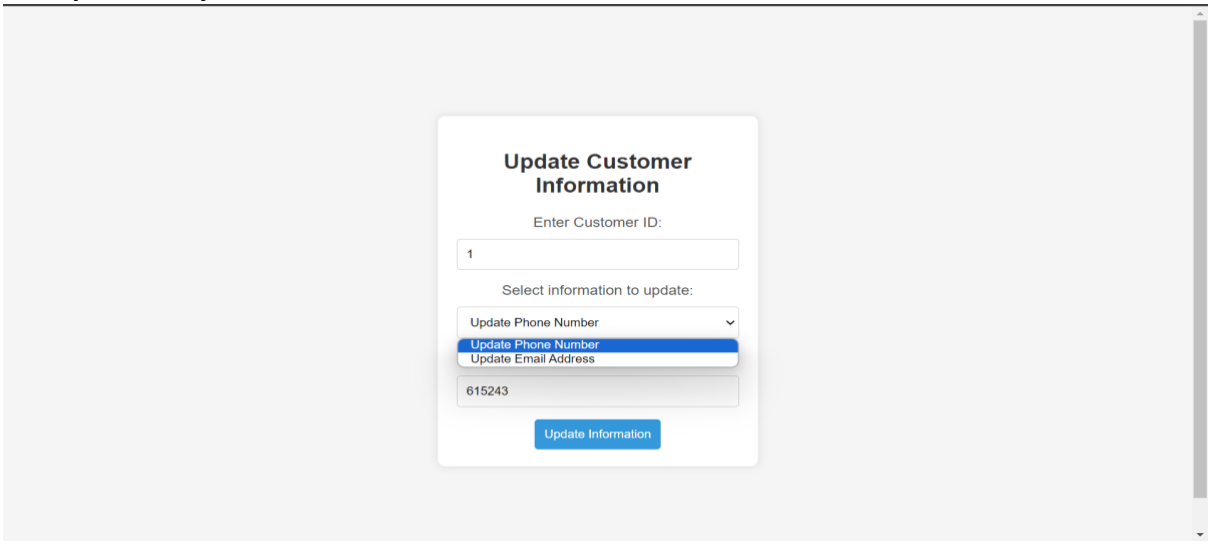


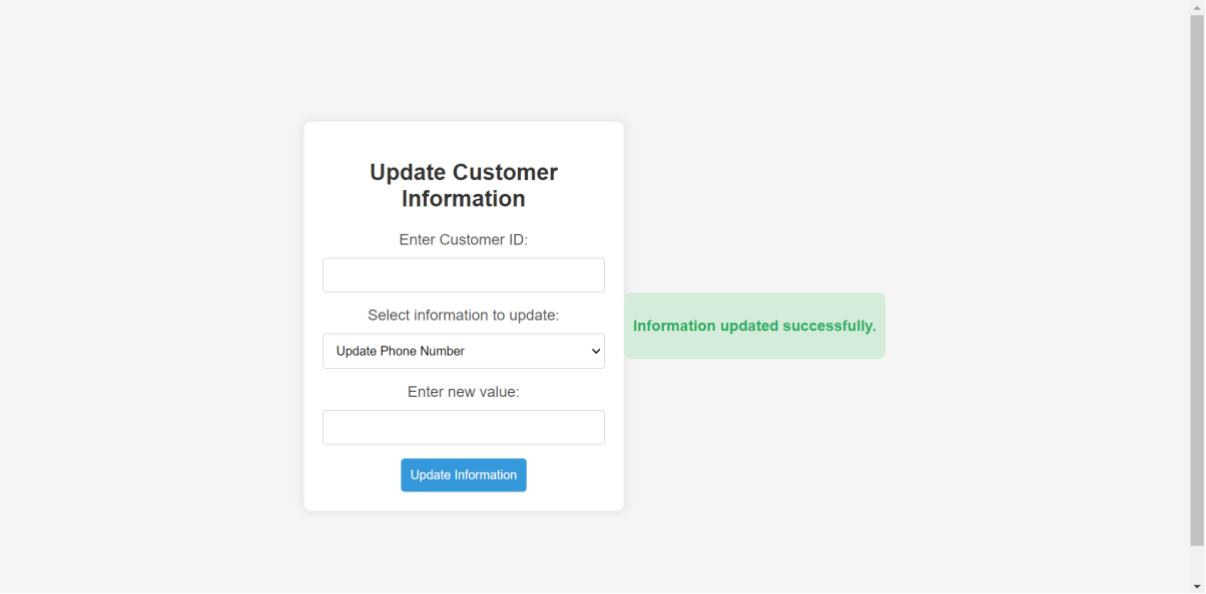


5.Know your booking

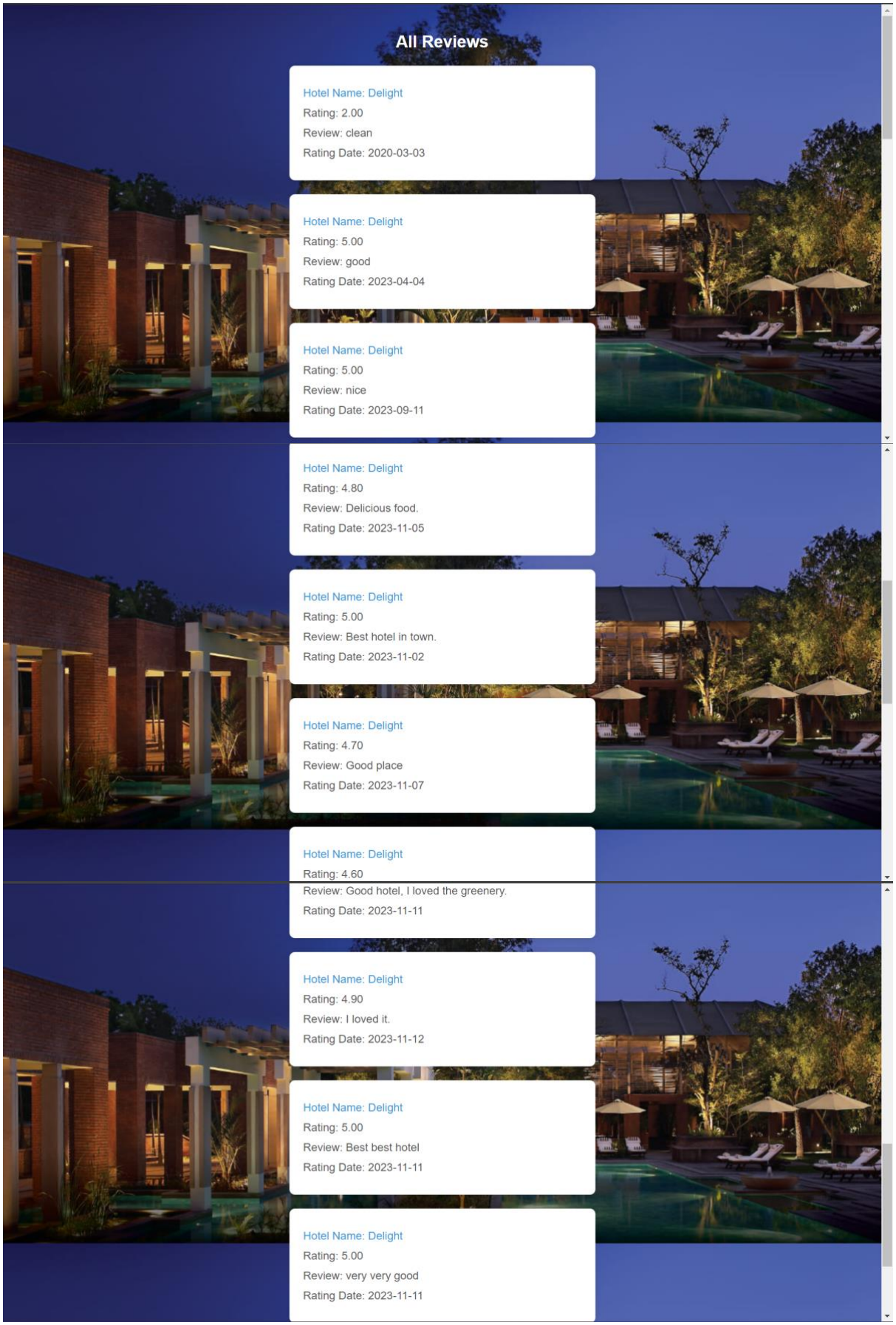


6.Update your information





7.See all reviews



8.Cancel booking