CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING

(CS331P) Database Management Systems

B. Tech – Computer Science and Engineering
(Internet of Things)

CHRIST (Deemed to be University),

School of Engineering and Technology,

Kumbalagodu, Bengaluru-560 074

November 2022

CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

Certificate

*This is to certify that Joshua PK Kurian, Fazal Khan, Aniket Das, Dipendra Singh have successfully completed their Mini Project work for (CS331–*Database Management Systems*) in partial fulfilment for the award of Bachelor of Technology in Computer Science and Engineering (Internet of Things) during the year 2022-2023.*

HEAD OF THE DEPARTMENT **FACULTY- IN CHARGE**

EXAMINER 1:

EXAMINER 2:

Name : Fazal Khan - 2162431
Joshua PK Kurian - 2162433
Dipendra Singh - 2162429
Aniket Das-2162401

Examination Center : SoET, CHRIST (Deemed to be University)

Date of Examination : 28/11/2022

ACKNOWLEDGEMENT

With immense pleasure and deep respect, we express our heartfelt gratitude to our revered teacher and guide Mr. Arul V

department of computer science and engineering

for his excellent guidance, constant support, concern, and advice

throughout the period of the project and sustained interest, help and encouragement in the preparation of the project.

TABLE OF CONTENTS

ABSTRACT

The Hotel management System project's purpose is to implement a web-application which helps customers search the availability and prices of various hotel rooms and view their bookings. This project also covers various features like online registration of the users, modifying the customer details and checking price details. In general, this web-application is designed to perform like any other online hotel management website available online.

The Hotel management system consists of two types of users. The customers who access the information provided by the application and the administrator who modifies and updates the information.

The Graphical User Interface (GUI) of our management system was developed using HTML,CSS.

The report contains the details of all the tasks carried out during the entire application development life cycle of the Hotel management Project. This document depicts all the details of the project.

List of Figures Page. No

List of Tables Page. No

List of Screens Page. No

1. INTRODUCTION

1.1 MOTIVATION

Hotel management System is helpful for reserving room for manager and customers. In the latter system all the activities were done manually which were very time consuming and costly. Our Hotel management System deals with the various activities related to the hotel room.

- Room Booking
- Room Availability
- Price Details
- View the bookings

1.2 PROBLEM DEFINITION

If a person wants to reserve a room he/she has to manually go to the hotel and do it but in our Hotel management System we provide an interface to book a room and reserve it prior to the visit. It's our responsibility to keep track of system users, customers, room information and cancellation. The functionality of the Hotel management System is broken into various primary groups.

Customer reservation information and users were added, deleted and updated in the implementation phase to account for the way we decide to implement security. User keeps track of the customer's name, age, room type, room number and customer reservation information link provides a link between the customers reservation information and login table.

A hotel manager can book, cancel, or postpone any reservation for any customer. Rooms are booked through Reservation in which all the details regarding the customer and his room are entered. A unique id is provided for each customer and with the help of which one can print the booking bills etc.

1.3 HOTEL SPECIFICATIONS

Hotel Beach View consists of **4 floors** with a **total of 32 rooms**.The first floor has three deluxe bedrooms and six economy rooms.The second floor has two deluxe bedrooms and seven economy rooms.The third floor has five deluxe bedrooms, two economy rooms and one suite.The fourth floor has four deluxe bedrooms and two suite's. The prices of the rooms are as follows:

Economy:	₹1699
Deluxe:	₹3145
Deluxe plus:	₹4499
Suite:	₹7299

Deluxe plus booking system

Users are also given an option to book deluxe plus rooms where they will be given a double bed and a single bed.The hotel will keep one deluxe and one economy room as booked and these two rooms will be unavailable to others in the system.

1.4 OBJECTIVE OF PROJECT

The main objective of the project is to create a hotel management system where a customer can book a room and get the information regarding the bookings, view different types of room available with different timing for a particular date, price details and cancel the booked services. The various features added to the project provide all the functions to make the task easy to perform.

1.5 LIMITATIONS OF PROJECT

- Cannot be used as a real time application for hotel room booking.(works on local server)
- Payment interface does not exist

2. ANALYSIS

The Hotel management System project is to create a web application which helps the customers to book the room by searching the availability, view prices of various room reservations and view their bookings. This project also covers various features like room services, cancellation of booking etc.

Software's used:

- HTML & CSS Front-end
- PyCharm for connectivity.
- Django Back-end.
- Database in MySQL.

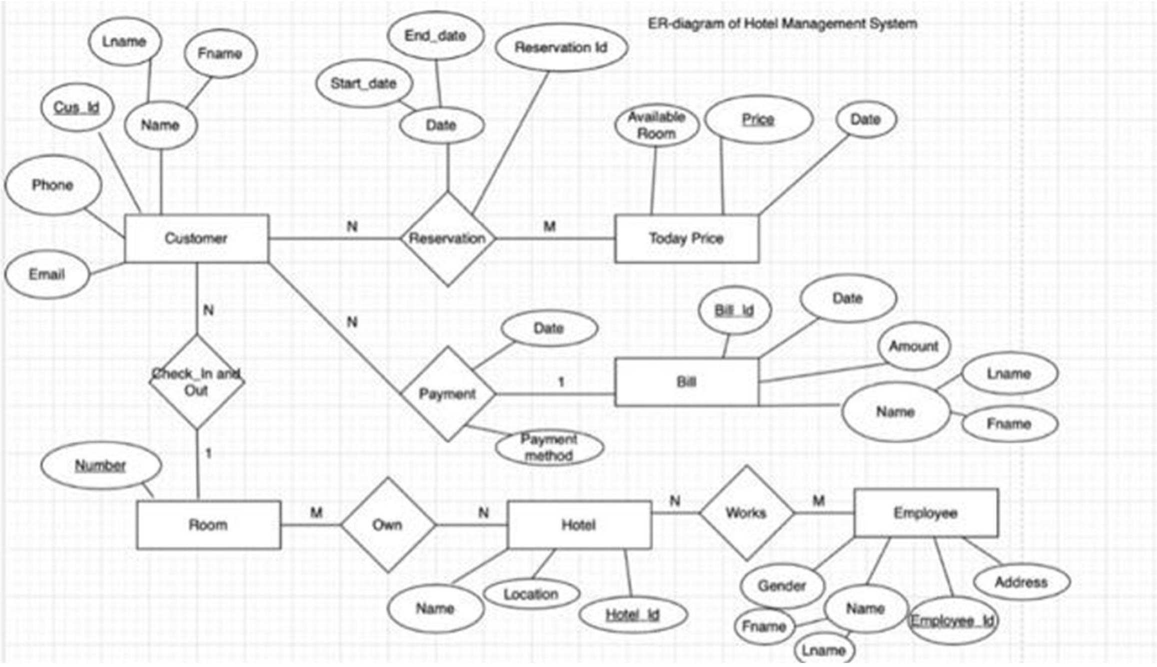
We are using Python for connection between Backend and Frontend.

PyCharm is an Integrated Development Environment used for programming in Python. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems, and supports web development with Django. PyCharm is developed by the Czech company JetBrains.

Django is an open-source framework for backend web applications based on Python — one of the top web development languages. Its main goals are simplicity, flexibility, reliability, and scalability.

MySQL is an open-source relational database management system. Its name is a combination of "My", the name of co-founder Michael Widenius's daughter My, and "SQL", the abbreviation for Structured Query Language.

2.3 CONTENT DIAGRAM OF THE PROJECT



3. DESIGN

3.1 INTRODUCTION

BACK END

The backend is the data access layer that holds room details, reservation, and user information. It helps applications to operate by processing, storing, and transferring their data to and from the customer-facing side. The back end is databases. Databases are organized and structured collections of data. Backend components like the web server, application server, and database which make the frontend interactive. The users can check the room, book the rooms, and manage their bookings using their own devices. Database files are the key source of information into the system.

We are using Django for backend and MySQL for databases.

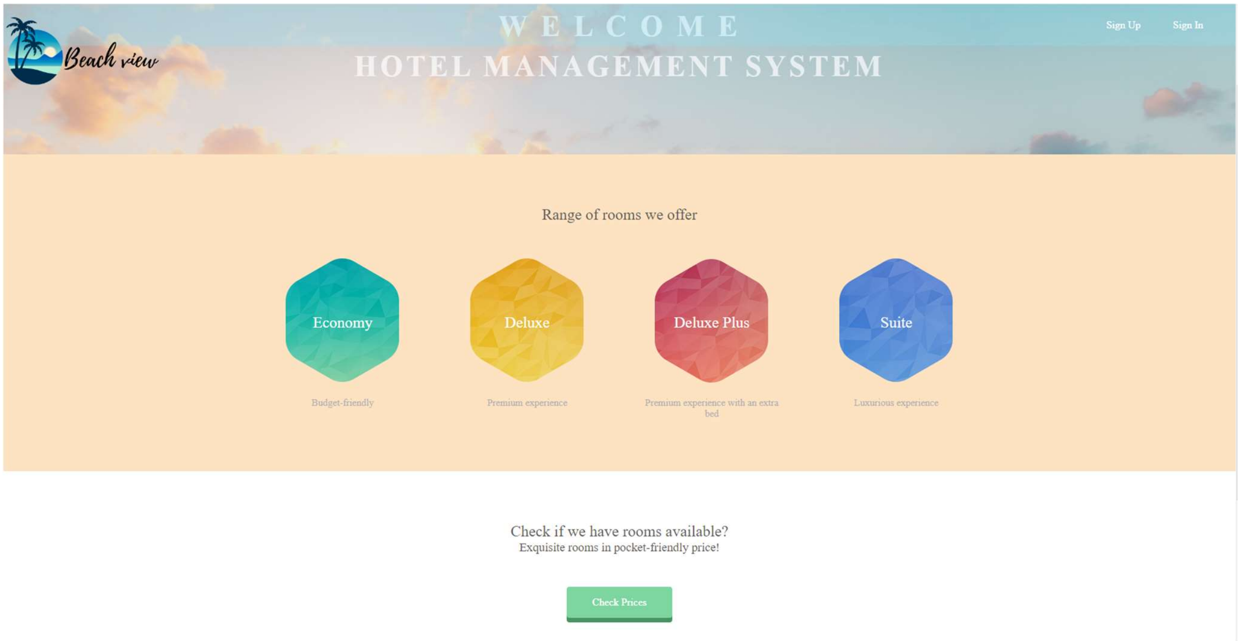
FRONT END

- Front end is done using HTML, CSS (all GUI).

3.2 MODULE DESIGN AND ORGANIZATION

View bookings-In this module we can view the user's reservations.

3.2.1 Home page



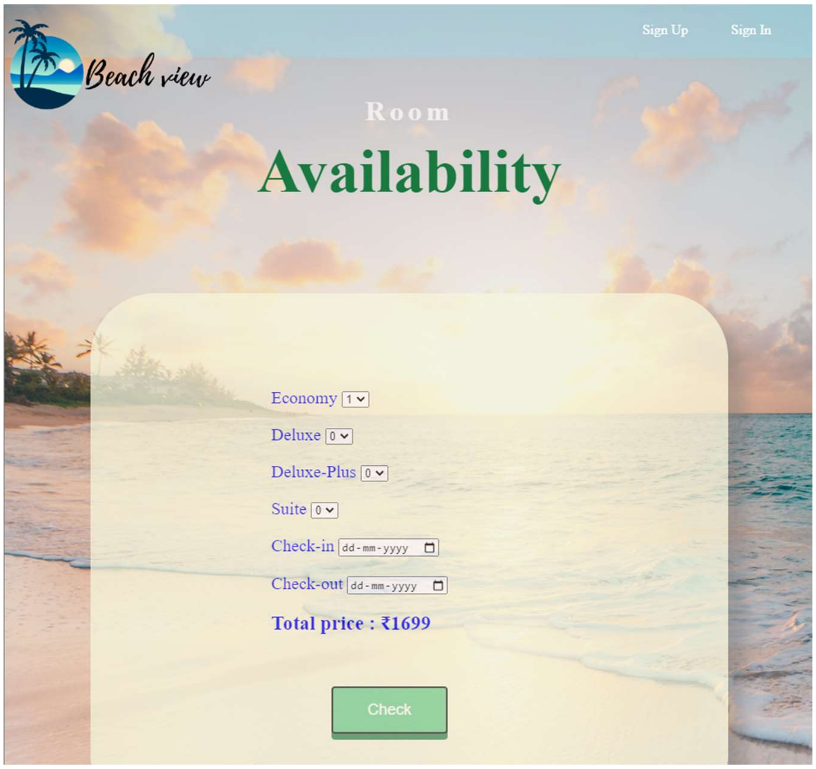
WELCOME

HOTEL MANAGEMENT SYSTEM

Range of rooms we offer

Check if we have rooms available?
Exquisite rooms in pocket-friendly price!

3.2.2 Check Room Availability/ Check Prices



Economy 1

Deluxe 0

Deluxe-Plus 0

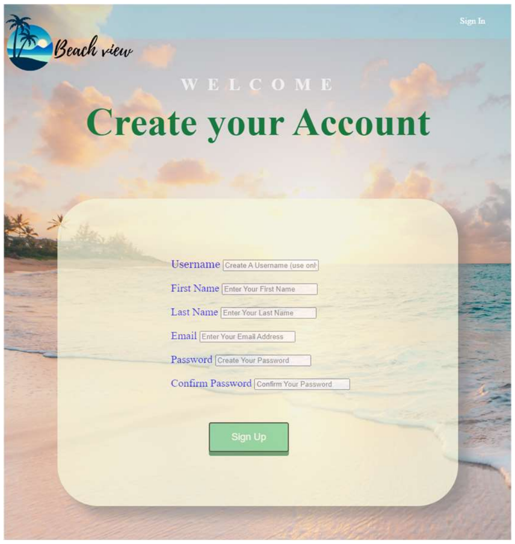
Suite 0

Check-in dd-mm-yyyy

Check-out dd-mm-yyyy

Total price : ₹1699

3.2.3 Sing-up and Sign-in



Username Create A Username (use onl

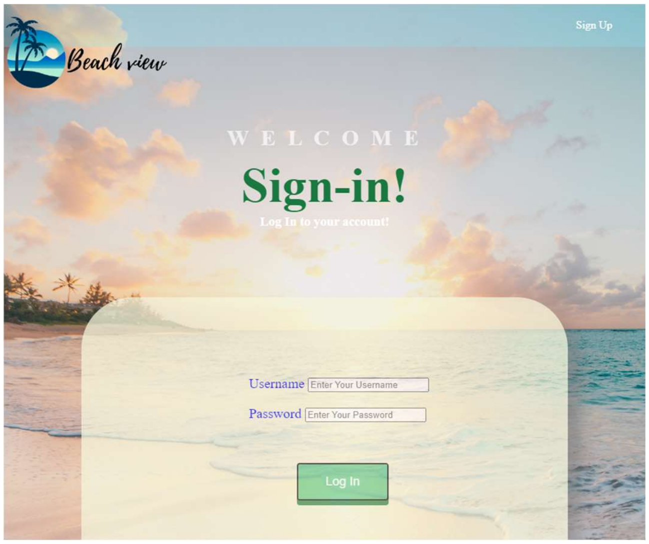
First Name Enter Your First Name

Last Name Enter Your Last Name

Email Enter Your Email Address

Password Create Your Password

Confirm Password Confirm Your Password

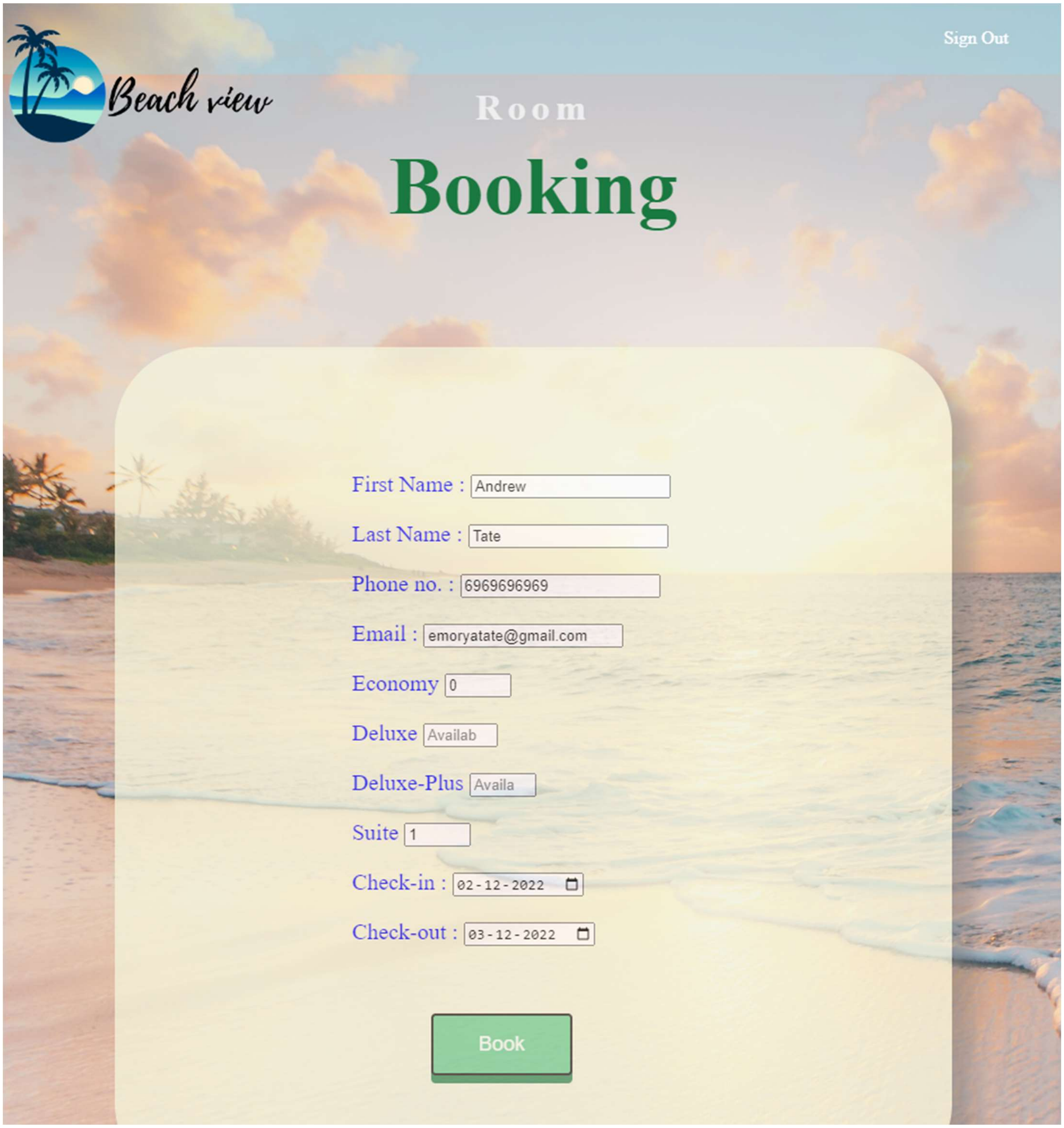


Username Enter Your Username

Password Enter Your Password

3.2.4 Room booking

Here you can give your details.



First Name : Andrew

Last Name : Tate

Phone no. : 6969696969

Email : emoryatate@gmail.com

Economy 0

Deluxe Availab

Deluxe-Plus Availa

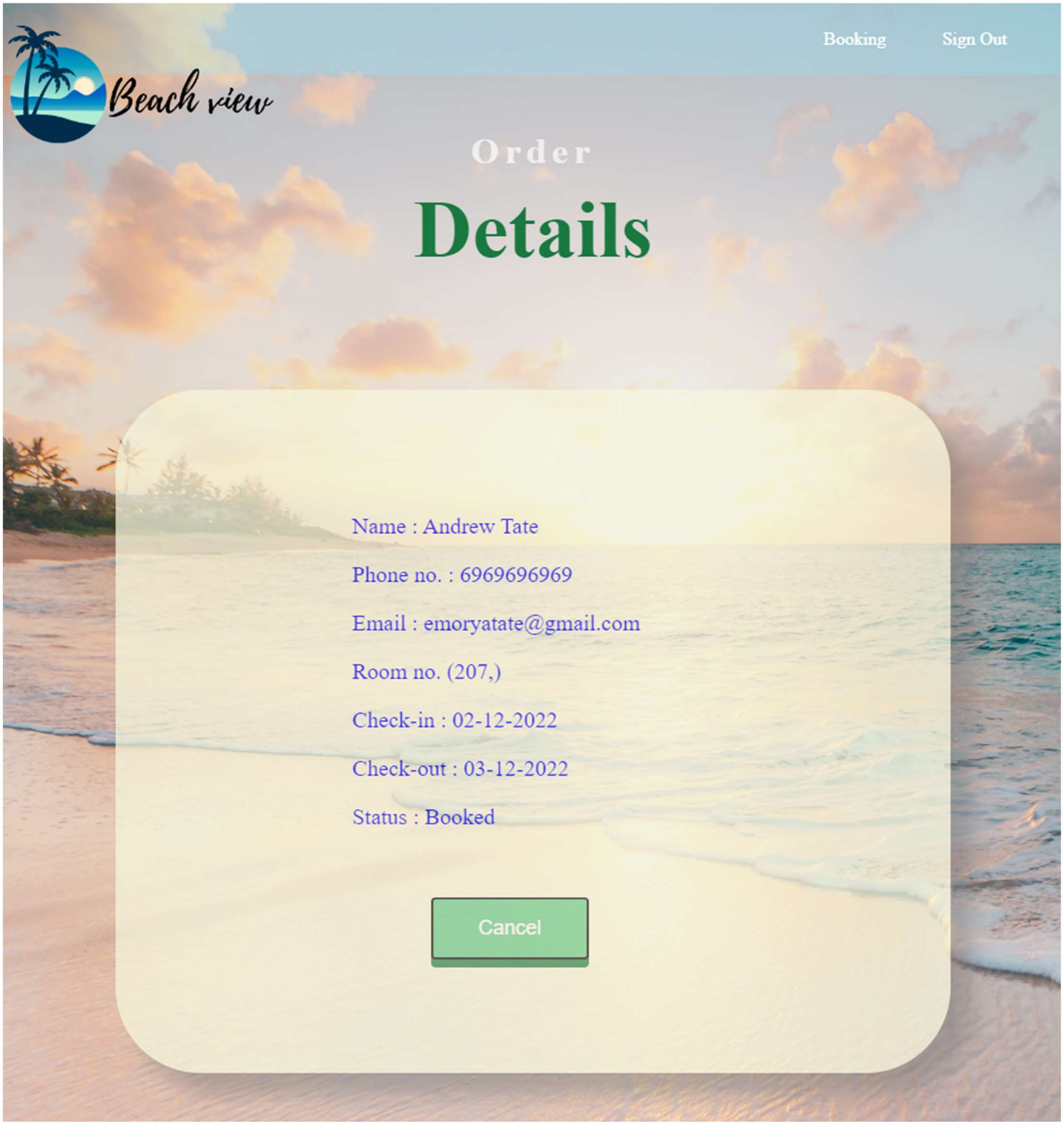
Suite 1

Check-in : 02-12-2022

Check-out : 03-12-2022

3.2.5 View Booking

You can view reservation details here.



Name : Andrew Tate

Phone no. : 6969696969

Email : emoryatate@gmail.com

Room no. (207,)

Check-in : 02-12-2022

Check-out : 03-12-2022

Status : Booked

4. IMPLEMENTATION & TESTING

4.1 METHOD OF IMPLEMENTATION

The front of the application is done using HTML, CSS and the backend using Django server and MySQL database.We have created a HOTEL MANAGEMENT database and all the entities are created inside the database. PyCharm is used to connect the front end and back end.

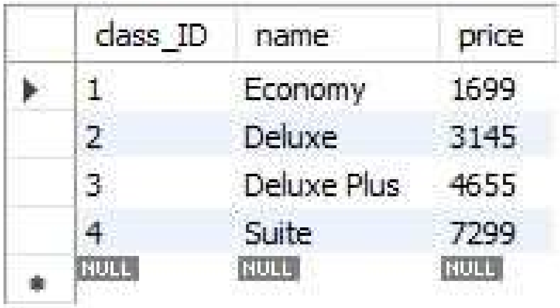
Working:

We are using XAMPP for connection between Backend and Frontend.

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.
We turn on Apache and MySQL

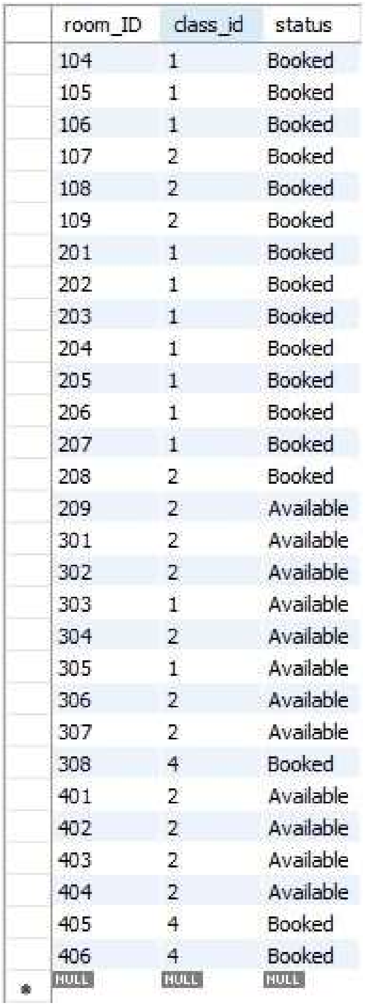
4.1.1 Queries used for creating TABLES

```
CREATE TABLE room_class

(

    class_ID INT NOT NULL PRIMARY KEY,

    name VARCHAR(15),

    price INT

);

INSERT INTO  room_class values(01,'Economy',1699);

INSERT INTO  room_class values(02,'Deluxe',3145);

INSERT INTO  room_class values(03,'Deluxe Plus',4655);

INSERT INTO  room_class values(04,'Suite',7299);
```



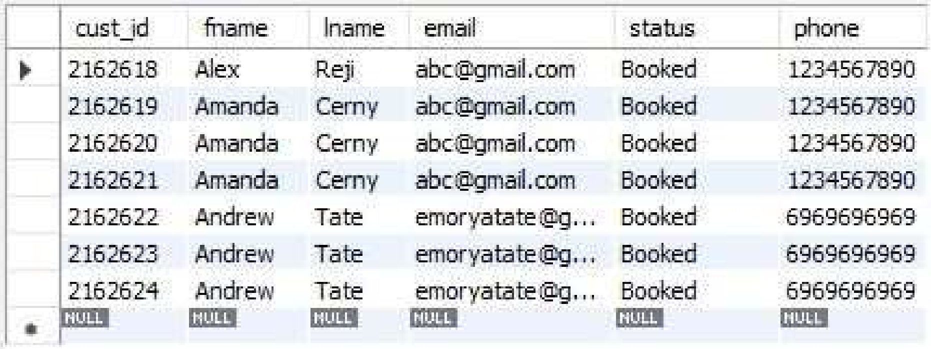
	class_ID	name	price
▶	1	Economy	1699
	2	Deluxe	3145
	3	Deluxe Plus	4655
	4	Suite	7299
*	NULL	NULL	NULL

```
CREATE TABLE room_info

(

    room_ID INT NOT NULL PRIMARY KEY,

    class_id INT,

    status varchar(15)

    CHECK(status in ('Available','Booked','Checked-in','Checked-
    in')),

    FOREIGN KEY (class_id) REFERENCES room_class(class_id)

);



 INSERT INTO  room_info values(101,1,'Available');

 INSERT INTO  room_info values(102,1,'Available');

 INSERT INTO  room_info values(103,1,'Available');

 INSERT INTO  room_info values(104,1,'Available');

                            ||

                            \/

 INSERT INTO  room_info values(406,4,'Available');
```



room_ID	class_id	status
104	1	Booked
105	1	Booked
106	1	Booked
107	2	Booked
108	2	Booked
109	2	Booked
201	1	Booked
202	1	Booked
203	1	Booked
204	1	Booked
205	1	Booked
206	1	Booked
207	1	Booked
208	2	Booked
209	2	Available
301	2	Available
302	2	Available
303	1	Available
304	2	Available
305	1	Available
306	2	Available
307	2	Available
308	4	Booked
401	2	Available
402	2	Available
403	2	Available
404	2	Available
405	4	Booked
406	4	Booked
NULL	NULL	NULL

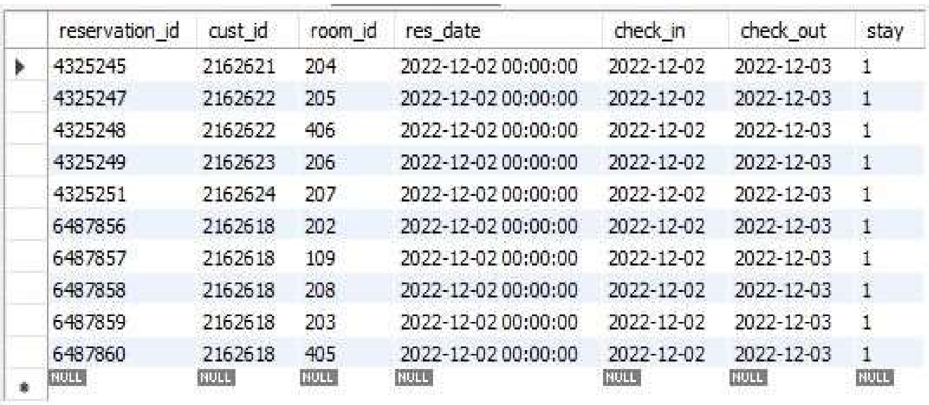
```
 CREATE TABLE CUST_INFO

(

   cust_id int NOT NULL AUTO_INCREMENT=2162400 PRIMARY KEY,

   fname varchar(25),

   lname varchar(25),

   phone int(10),

   email varchar(75),

   status varchar(15)

);
```



cust_id	fname	lname	email	status	phone
2162618	Alex	Reji	abc@gmail.com	Booked	1234567890
2162619	Amanda	Cerny	abc@gmail.com	Booked	1234567890
2162620	Amanda	Cerny	abc@gmail.com	Booked	1234567890
2162621	Amanda	Cerny	abc@gmail.com	Booked	1234567890
2162622	Andrew	Tate	emoryatate@g...	Booked	6969696969
2162623	Andrew	Tate	emoryatate@g...	Booked	6969696969
2162624	Andrew	Tate	emoryatate@g...	Booked	6969696969
NULL	NULL	NULL	NULL	NULL	NULL

```
CREATE TABLE RESERVATION
```

```
(

reservation_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,

cust_id INT ,

room_id INT ,

res_date DATETIME,

check_in DATE,

check_out DATE,

stay DATE,

FOREIGN KEY(cust_id)REFERENCES cust_info(cust_id),

FOREIGN KEY (room_id)REFERENCES room_info(room_id)

);
```



reservation_id	cust_id	room_id	res_date	check_in	check_out	stay
4325245	2162621	204	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
4325247	2162622	205	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
4325248	2162622	406	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
4325249	2162623	206	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
4325251	2162624	207	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
6487856	2162618	202	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
6487857	2162618	109	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
6487858	2162618	208	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
6487859	2162618	203	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
6487860	2162618	405	2022-12-02 00:00:00	2022-12-02	2022-12-03	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL

4.1.2 Queries used for creating VIEWS

```
    CREATE VIEW economy_available AS SELECT room_ID FROM room_info

    WHERE class_id=1 AND status='Available';


    CREATE VIEW deluxe_available AS SELECT room_ID FROM room_info

    WHERE class_id=2 AND status='Available';


    CREATE VIEW suite_available AS SELECT room_ID FROM room_info

    WHERE class_id=4 AND status='Available';


    CREATE VIEW cust_id AS SELECT cust_id FROM cust_info;
```

4.2 TESTING/DEBUGGING

System testing is an important but critical part of a system. Module testing involves the process of testing the logical units of a program individually and integrating the individual modules, to test the overall system.

The objective of module testing is to determine whether the module meets its specifications.

To perform module testing two things are to be considered: the design of test cases and the coordination of the multiple modules. Test cases may be constructed from specification or by analysing the module code.

The testing areas:

1. **Testing for data entry error :**

   Incorrect data entry, leads to incorrect results. Sometimes it may create run time errors. So, data should be checked for validity.

2. **Testing for basic principles :**

   If all processing is in order but display of the screens is disorderly then that must be rectified and such screens can confuse the user. Users will be left wondering on the next actions to be taken. In some of the cases the operator/user has the habit of using a certain key for the option.

3. **Testing for suitable error messages :**

   If the error messages are not suitable phases or incorrect then there can be log off problems to the user who is non-professional will not be able to figure out what exactly the error is.

4. **Testing for suitable response time :**

   In some cases, the results of a particular query may take out a lot of system resources and a longer time to process. In such cases program logic must be rewritten. One major factor, which has influenced the functionality of the system, is the valid data. As and when the master files were completed, the user added valid data and thus the expected results could also be verified.

5. CONCLUSION:

We have successfully developed the desktop application for the Hotel management system. With the help of various tools, we have been able to make an application that would help us to search for a room and book room. We have been successful in our attempt to take care of the needs of users as well as administrators.

Hotel management systems would help each person to view different sorts of rooms available with different services for a particular date. The application makes it convenient for the new customers to login and search for the rooms available. This application helps

User to book the room on convenience without . It makes the entire process online where users can login, search for rooms, book rooms. It also has a facility for common users by logging into the system and seeing the status of bookings.

The project has a very vast scope in future. Project can be updated in the near future as and when required since it is very flexible in terms of expansion.

Future enhancement:

- In future we can add Payment interface.
- Booking cancellation option.

REFERENCES

1. Abraham Silberschatz, Henry F. Korth and S. Sudarshan- “Database System Concepts”, 7th Edition, McGraw-Hill, 2021
2. https://www.geeksforgeeks.org/django-sign-up-and-login-with-confirmation-email-python/
3. https://www.w3schools.com/html/html_css.asp - HTML & CSS
4. https://stackoverflow.com/ - Debugging the code and rectifying errors
5. https://www.youtube.com/ - Video Tutorials