CS257 DATABASE AND INFORMATION SYSTEMS

PROJECT REPORT

GitHub Repository

HOTEL DATABASE MANAGEMENT

T. RAM PREETHAM 210001070
VIVEK BHOJWANI 210001079

Problem Statement

The main objective is to create a database management system for a hotel. We need an organised management system which can easily manage all operations and data of hotel. The following areas of the hotel database will be managed like customer details and their stay details, storing customer details for future reference, accessing staff details,etc.

Solution

We made a Hotel database management system to simplify front desk office tasks, enhance management functions, and improve the experience of customers and guest reservations. With this software, a hotel can improve efficiency and effectively manage operations such as check-ins, check out and confirmation of reservations

Tech Stacks

EJS
CSS
JavaScript
Bootstrap

Frontend

NodeJS
ExpressJS

Backend

MySQL

Database

Features

Features our solution provide are:

Users will be able to browse the site without signing in.

They will be able to see vacancies available and various services provided by the hotel.

To book a room they have to login or sign up.

If user id is not available, action will be aborted

They will be able to book any available room according to their convenience.

If user id is not available, action will be aborted

An option will be given for cancellation aided with authentication

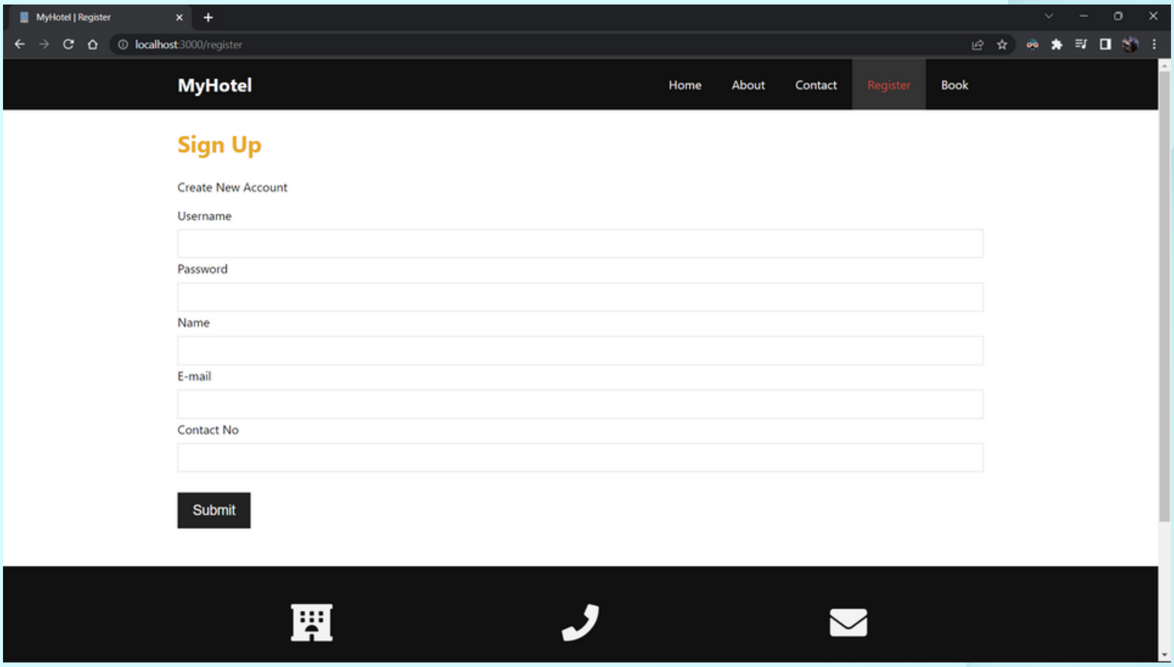
After booking, customers can print/download their confirmation receipt.

Cancellation and booking is aided by authentication.

TFor digital assistance you can contact the hotel through the site.

Features Glimpse

Registration



Sign Up

Create New Account

Username

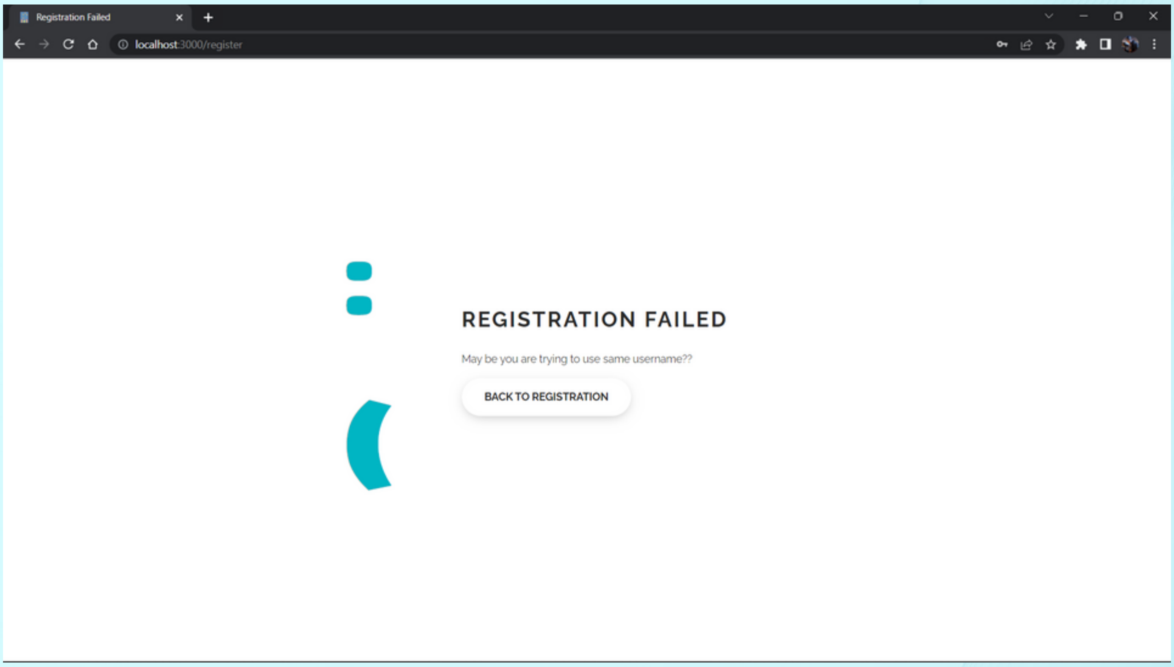
Password

Name

E-mail

Contact No

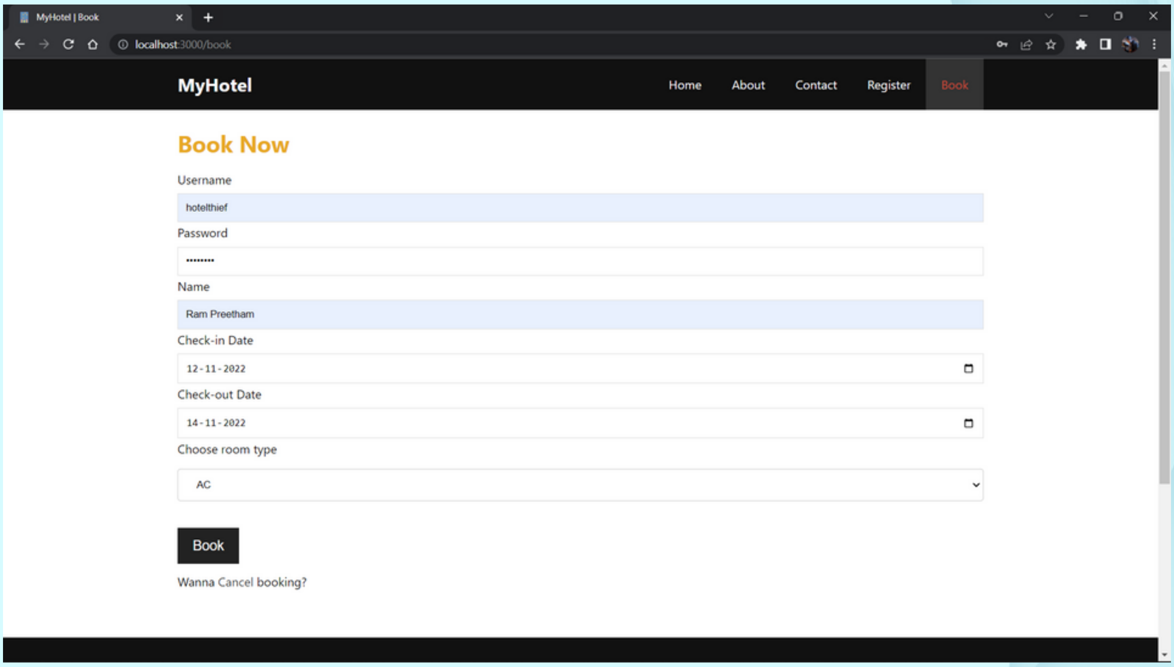
Unique username constraint



REGISTRATION FAILED

May be you are trying to use same username??

Book Rooms



Book Now

Username
hotelthief

Password
••••••••

Name
Ram Preetham

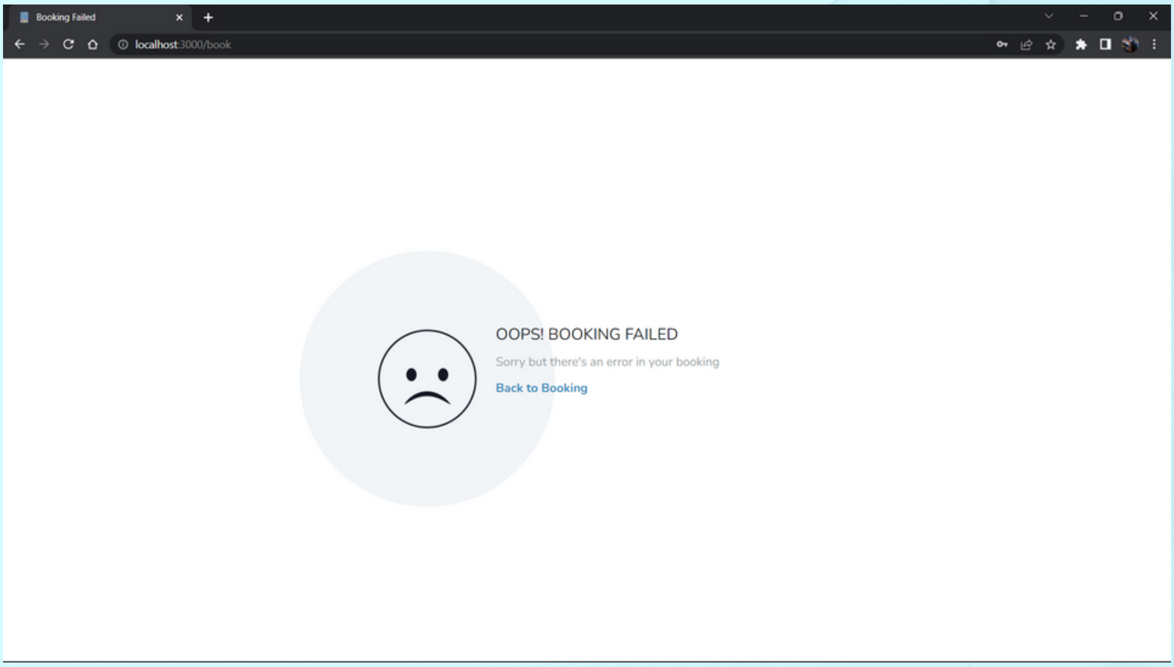
Check-in Date
12-11-2022

Check-out Date
14-11-2022

Choose room type
AC

Wanna Cancel booking?

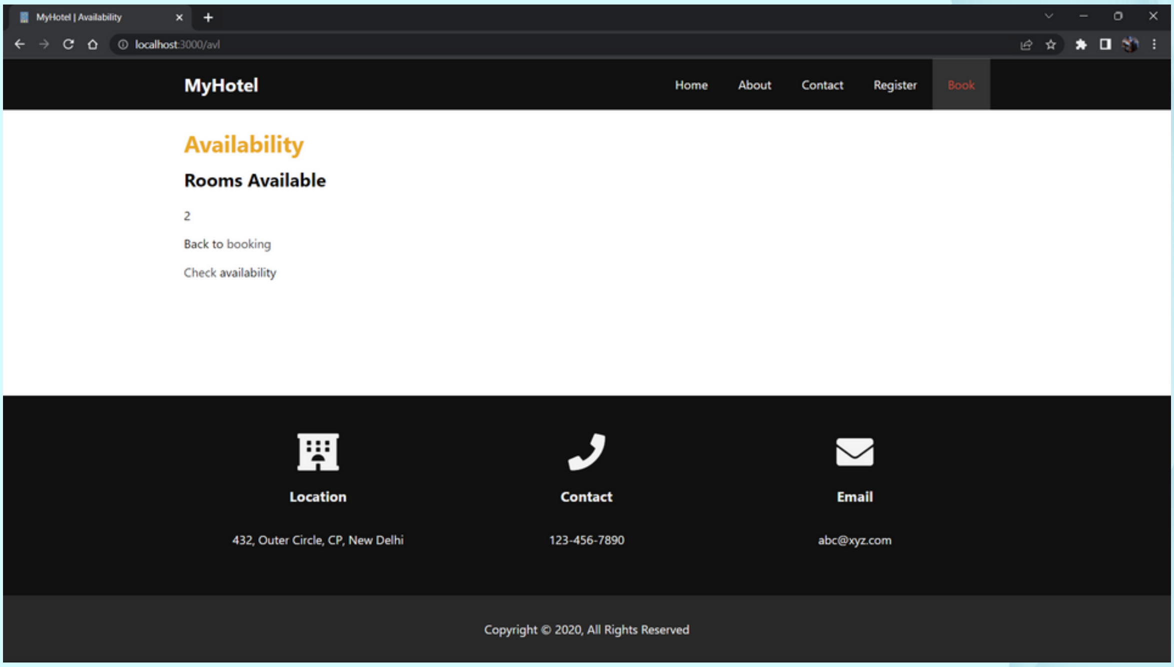
Authentication while booking



OOPS! BOOKING FAILED

Sorry but there's an error in your booking

Check Availability of rooms



Availability

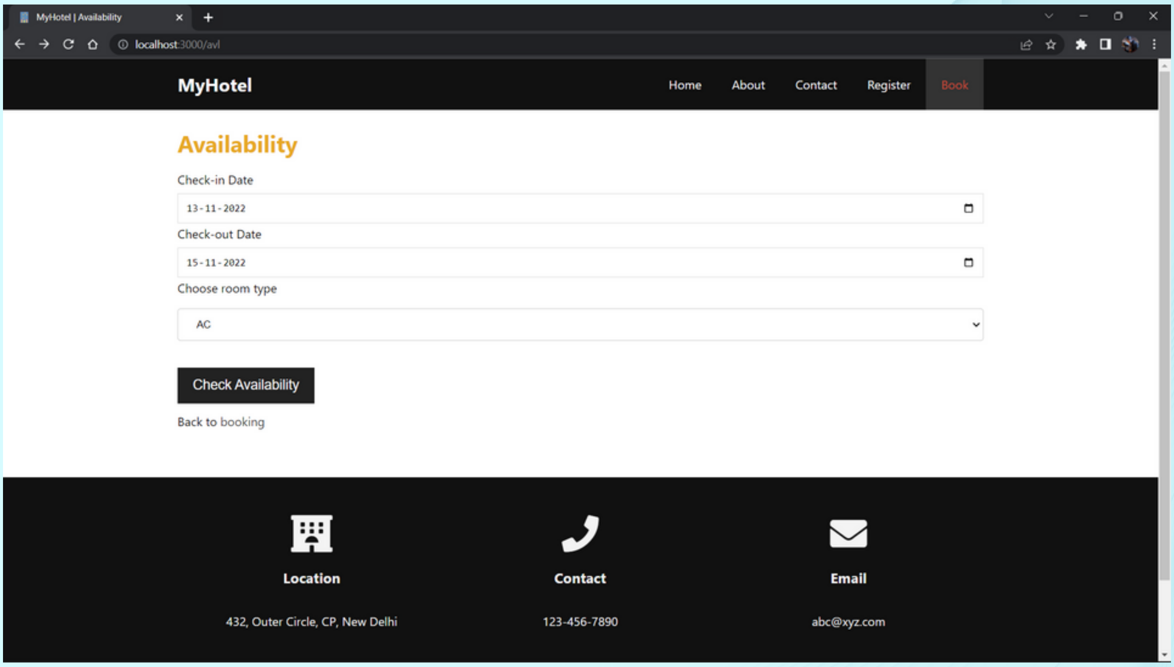
Rooms Available

2

Back to booking

Check availability

Location
432, Outer Circle, CP, New Delhi
Contact
123-456-7890
Email
abc@xyz.com
Copyright © 2020, All Rights Reserved



Availability

Check-in Date

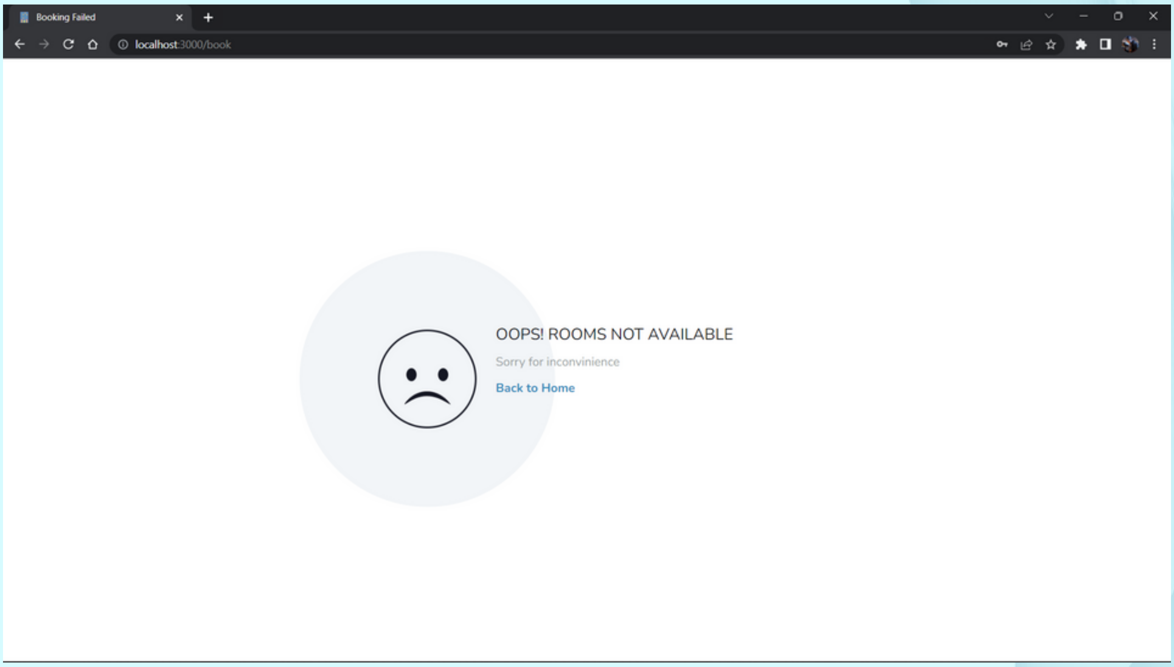
13-11-2022

Check-out Date

15-11-2022

Choose room type

AC

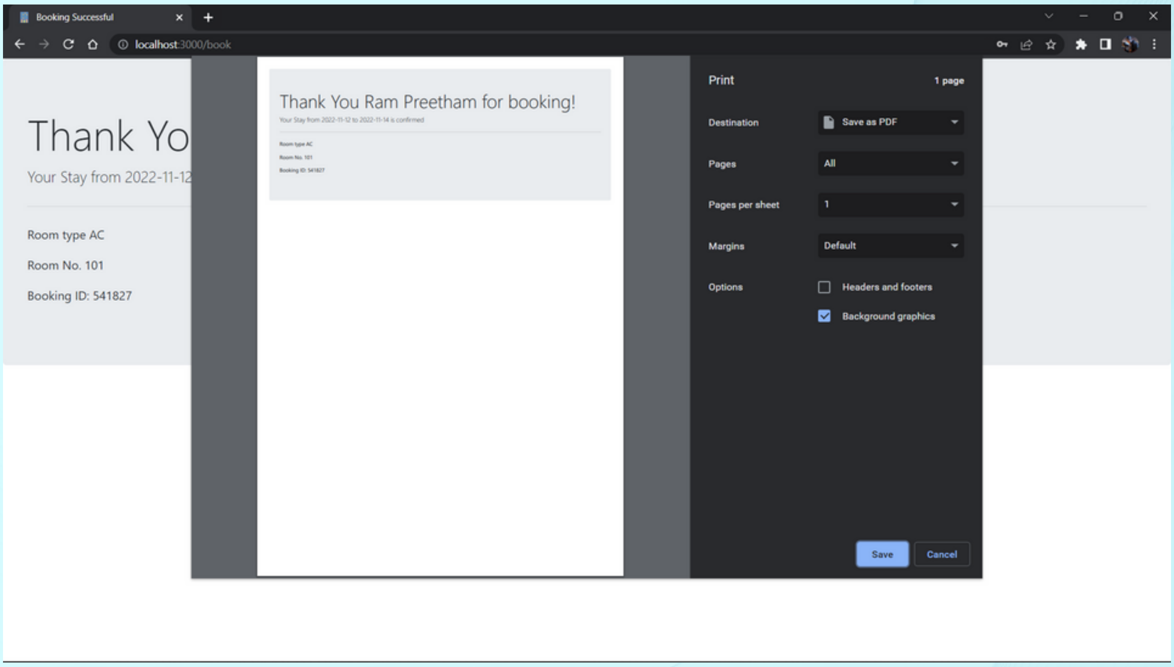


OOPS! ROOMS NOT AVAILABLE

Sorry for inconvinience

Back to Home

Print Confirmation



Thank Yo

Your Stay from 2022-11-12

Room type AC

Room No. 101

Booking ID: 541827

Thank You Ram Preetham for booking!

Your Stay from 2022-11-12 to 2022-11-14 is confirmed

Room type AC

Room No. 101

Booking ID: 541827

Print 1 page

Destination: Save as PDF

Pages: All

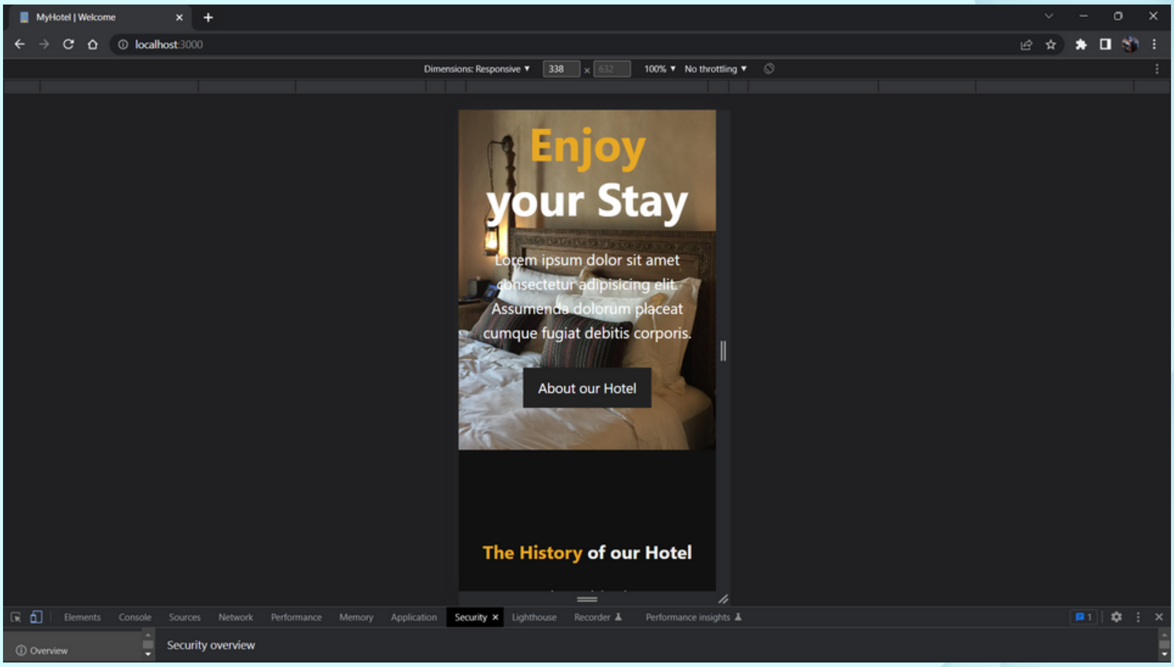
Pages per sheet: 1

Margins: Default

Options: ☐ Headers and footers

☑ Background graphics

Responsive Design



Enjoy your Stay

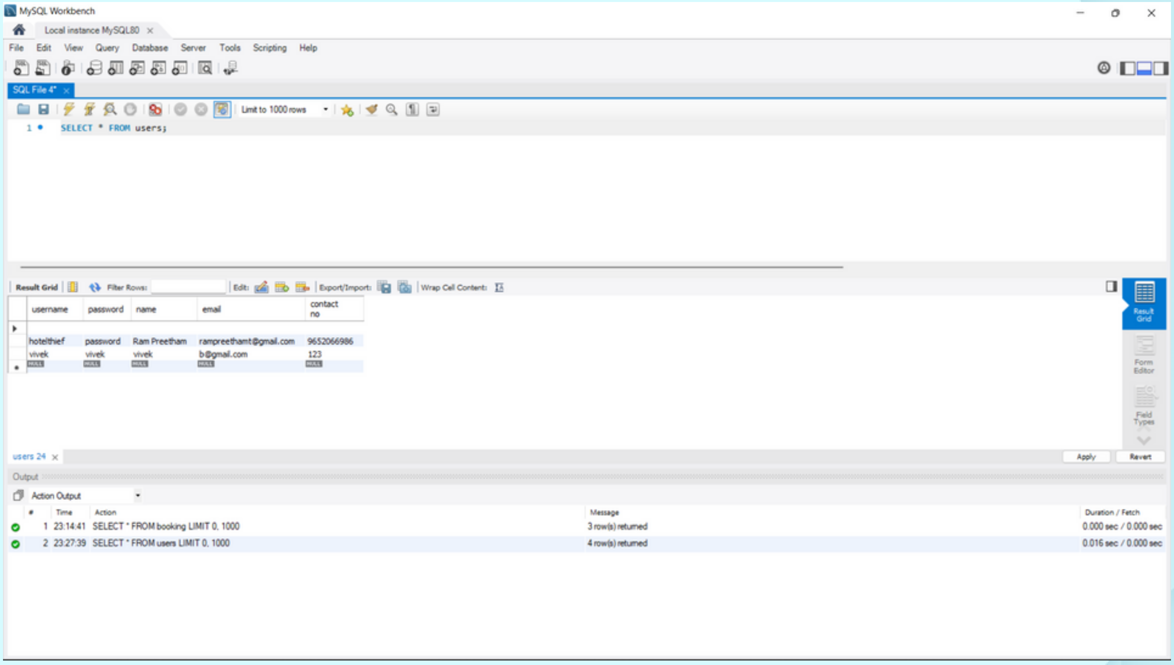
Lorem ipsum dolor sit amet consectetur adipisicing elit. Assumenda dolorum placeat cumque fugiat debitis corporis.

About our Hotel

The History of our Hotel

Security overview

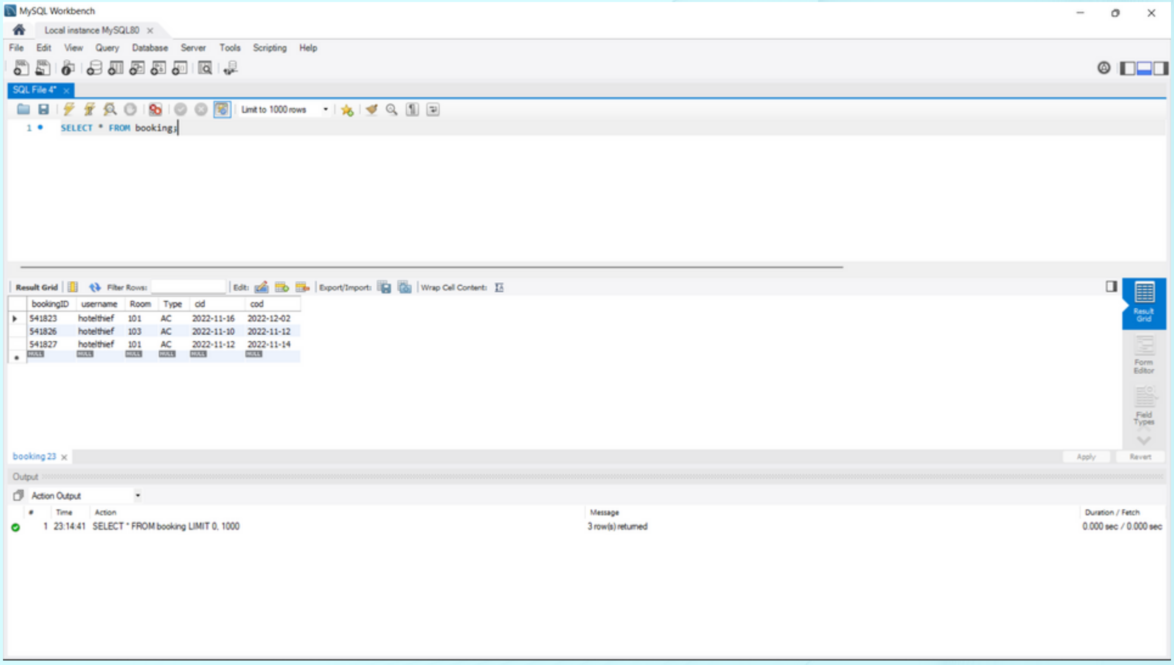
Database updates



```
SELECT * FROM users;
```

username	password	name	email	contact no
hotelthief	password	Ram Preetham	rampreethamt@gmail.com	9652066986
vivek	vivek	vivek	b@gmail.com	123
NULL	NULL	NULL	NULL	NULL

```
#  Time      Action                               Message             Duration / Fetch
1  23:14:41  SELECT * FROM booking LIMIT 0, 1000  3 row(s) returned   0.000 sec / 0.000 sec
2  23:27:39  SELECT * FROM users LIMIT 0, 1000    4 row(s) returned   0.016 sec / 0.000 sec
```



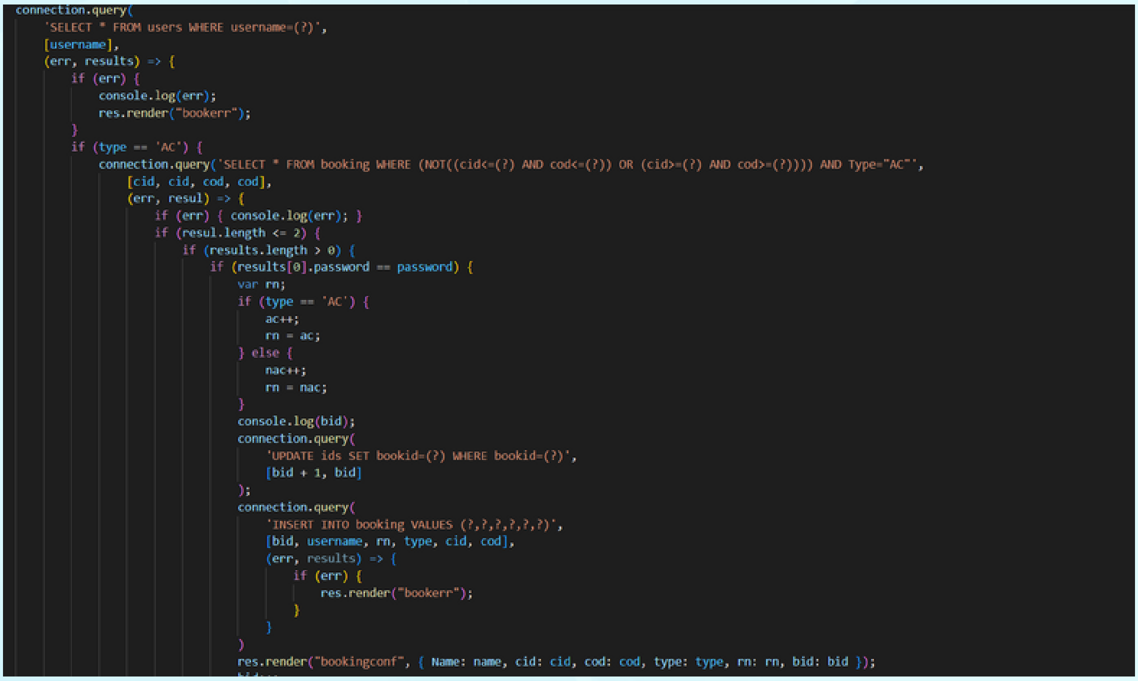
```
SELECT * FROM booking;
```

bookingID	username	Room	Type	cid	cod
541823	hotelthief	101	AC	2022-11-16	2022-12-02
541826	hotelthief	103	AC	2022-11-10	2022-11-12
541827	hotelthief	101	AC	2022-11-12	2022-11-14
NULL	NULL	NULL	NULL	NULL	NULL

```
#  Time      Action                               Message             Duration / Fetch
1  23:14:41  SELECT * FROM booking LIMIT 0, 1000  3 row(s) returned   0.000 sec / 0.000 sec
```

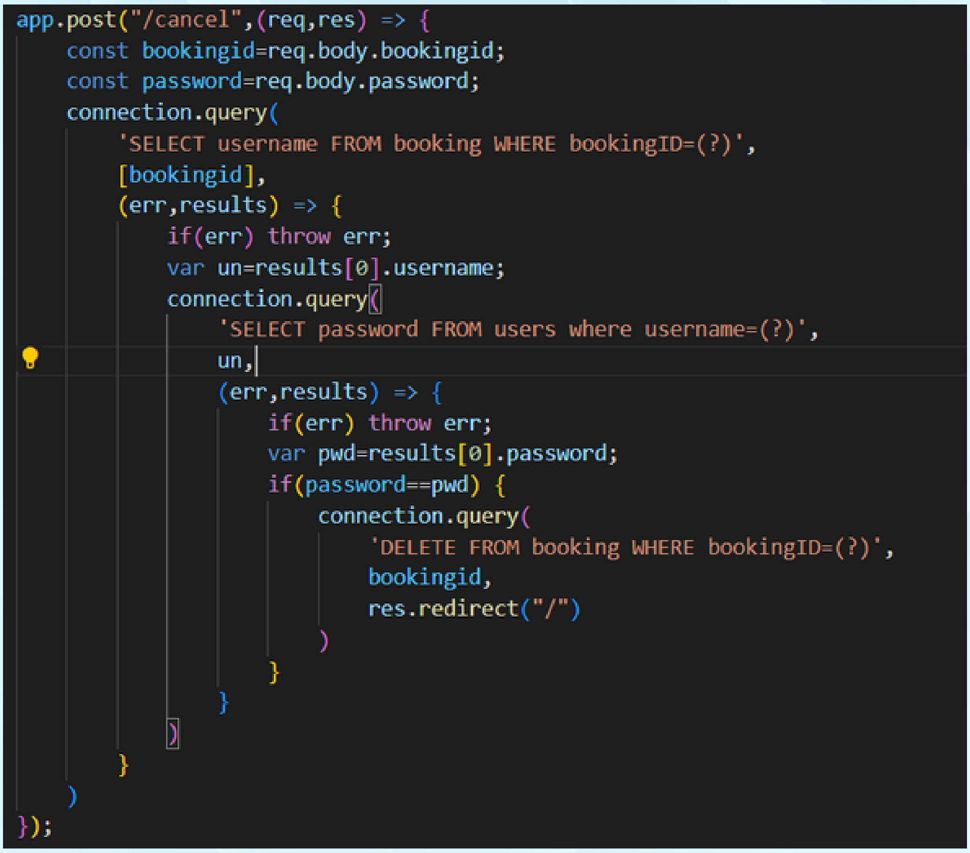
Queries

Booking



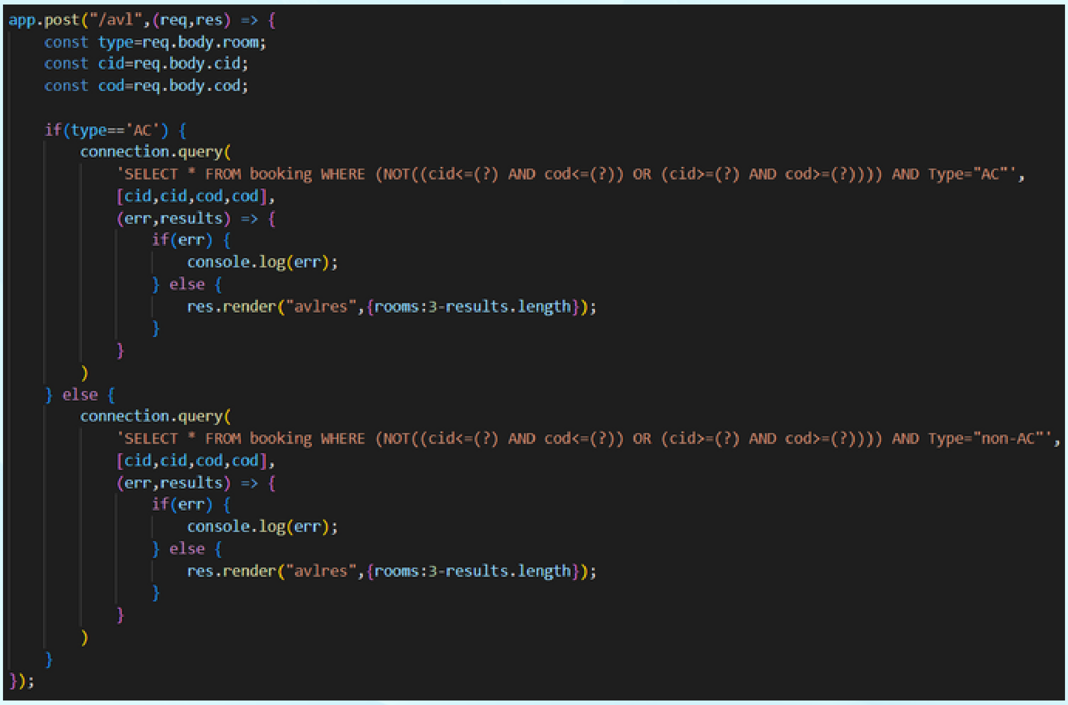
```
connection.query(
    'SELECT * FROM users WHERE username=(?)',
    [username],
    (err, results) => {
        if (err) {
            console.log(err);
            res.render("bookerr");
        }
        if (type == 'AC') {
            connection.query('SELECT * FROM booking WHERE (NOT((cid<=(?) AND cod<=(?)) OR (cid>=(?) AND cod>=(?)))) AND Type="AC"',
                [cid, cid, cod, cod],
                (err, resul) => {
                    if (err) { console.log(err); }
                    if (resul.length <= 2) {
                        if (results.length > 0) {
                            if (results[0].password == password) {
                                var rn;
                                if (type == 'AC') {
                                    ac++;
                                    rn = ac;
                                } else {
                                    nac++;
                                    rn = nac;
                                }
                                console.log(bid);
                                connection.query(
                                    'UPDATE ids SET bookid=(?) WHERE bookid=(?)',
                                    [bid + 1, bid]
                                );
                                connection.query(
                                    'INSERT INTO booking VALUES (?,?,?,?,?,?)',
                                    [bid, username, rn, type, cid, cod],
                                    (err, results) => {
                                        if (err) {
                                            res.render("bookerr");
                                        }
                                    }
                                )
                                res.render("bookingconf", { Name: name, cid: cid, cod: cod, type: type, rn: rn, bid: bid });
```

Cancellation



```
app.post("/cancel",(req,res) => {
    const bookingid=req.body.bookingid;
    const password=req.body.password;
    connection.query(
        'SELECT username FROM booking WHERE bookingID=(?)',
        [bookingid],
        (err,results) => {
            if(err) throw err;
            var un=results[0].username;
            connection.query(
                'SELECT password FROM users where username=(?)',
                un,
                (err,results) => {
                    if(err) throw err;
                    var pwd=results[0].password;
                    if(password==pwd) {
                        connection.query(
                            'DELETE FROM booking WHERE bookingID=(?)',
                            bookingid,
                            res.redirect("/")
                        )
                    }
                }
            )
        }
    )
});
```

Availability



```
app.post("/avl",(req,res) => {
    const type=req.body.room;
    const cid=req.body.cid;
    const cod=req.body.cod;

    if(type=='AC') {
        connection.query(
            'SELECT * FROM booking WHERE (NOT((cid<=(?) AND cod<=(?)) OR (cid>=(?) AND cod>=(?)))) AND Type="AC"',
            [cid,cid,cod,cod],
            (err,results) => {
                if(err) {
                    console.log(err);
                } else {
                    res.render("avlres",{rooms:3-results.length});
                }
            }
        )
    } else {
        connection.query(
            'SELECT * FROM booking WHERE (NOT((cid<=(?) AND cod<=(?)) OR (cid>=(?) AND cod>=(?)))) AND Type="non-AC"',
            [cid,cid,cod,cod],
            (err,results) => {
                if(err) {
                    console.log(err);
                } else {
                    res.render("avlres",{rooms:3-results.length});
                }
            }
        )
    }
});
```

FUTURE UPDATES

MyHotel

- Better user interface and user experience.
- Adding more filtering and searching options.
- Adding a payment gateway.
- Adding google map api to access user location for pickup services.
- Holiday. Management for Staff.
- Sending confirmation using e-mail

Conclusion

To conclude the the description about the project : This project,
developed using nodejs and MySql, is based on a very easy-to-use
booking system which is made after the analysis of the existing system and is flexible for future enhancement. This project was made to handle various room booking activities in the hotel and avoid the problems which occur when carried manually. Identification of the drawbacks of the existing system and improving them in our current project was the main objective of this project.