

# RESIDENCIA

## DSA J COMPONENT REVIEW 2&3

19.05.2020

—

MEERA L	18BEC0597
SIDDARTHA MAREEDU	19BCE0871
NAGIREDDY BHARAT REDDY	19BCE0854
SOURAV SUMAN THAKUR	19BCE0879
POTHA GOUNI MADHAV	19BDS0037



## Abstract

In this project Residencia, we aim to create a user interface portal for managing the allotment of rooms in a hotel/resort. Rooms can be booked online, and guests are given the advantage to check the status of the allotment. Also, to make the project more efficient, a control panel will be dedicated to hotel/resort administrators. This software will also include the code to retrieve information dynamically from the database and validate it.

The salient features of this software include

1. The display of available rooms on a calendar,
2. Sending confirmation or regret emails(on allotment or cancellation),
3. The administrator viewing the requests received,
4. Print the request in pdf format.
5. Exclusive guest dashboard

## Aim

This project aims to create a web-based application that allows the guests to book a room in a resort/ hotel and the administrators to handle all the activities online. This project uses HTML, CSS, Bootstrap, JQuery, JQuery UI and javascript for front end designing and responsive webpage, MYSQL as the backend tool to store data and PHP code to access the database dynamically.

## Objective

The principal objective of the project includes exploring various data structures, database management and data processing systems. This software will also include the code to retrieve information dynamically from the database and validate it. The entire design will be user-friendly.

## Scope

The designed software is a great tool and finds numerous practical applications such as booking resorts, serviced apartments or homestays.

## Introduction

Residencia is a resort/hotel management system. It is a web-based application that allows the hotel administrator to handle the room bookings online. The interactive GUI is very convenient and flexible thus enabling the guest to book rooms hassle free. This application is a powerful tool and gives the administrator a privileged control panel to manage room allotment in a very efficient way.

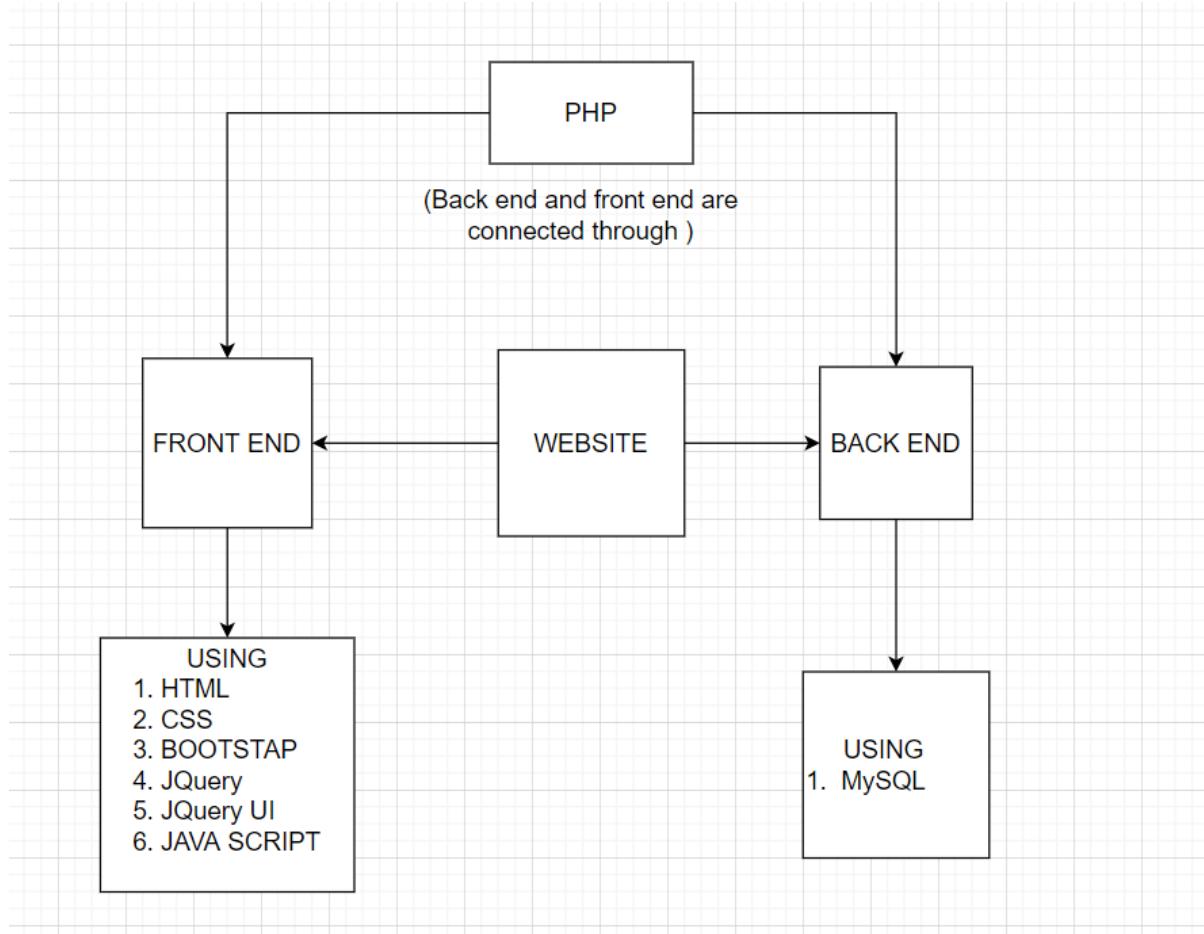
## List Of Modules

1. INDEX PAGE
2. SIGNUP & LOGIN PAGE
3. GUEST DASHBOARD
4. ADMIN PANEL

## Literature Review

The home page will be designed using HTML, CSS and Bootstrap. It consists of three different portals - Login, Sign Up and Admin login leading to respective pages. It also consists of a calendar which displays the availability of room for next 3 months and the guests can choose the preferred dates. There is a user registration form available where new users can create their account by providing required information to the system. The registration form comes with various validation. Confirmation mail is sent to the user as soon as the guest registers. The admin has a separate login with special privileges. The admin has the right to approve or disapprove the user's request for room booking.

## Implementation



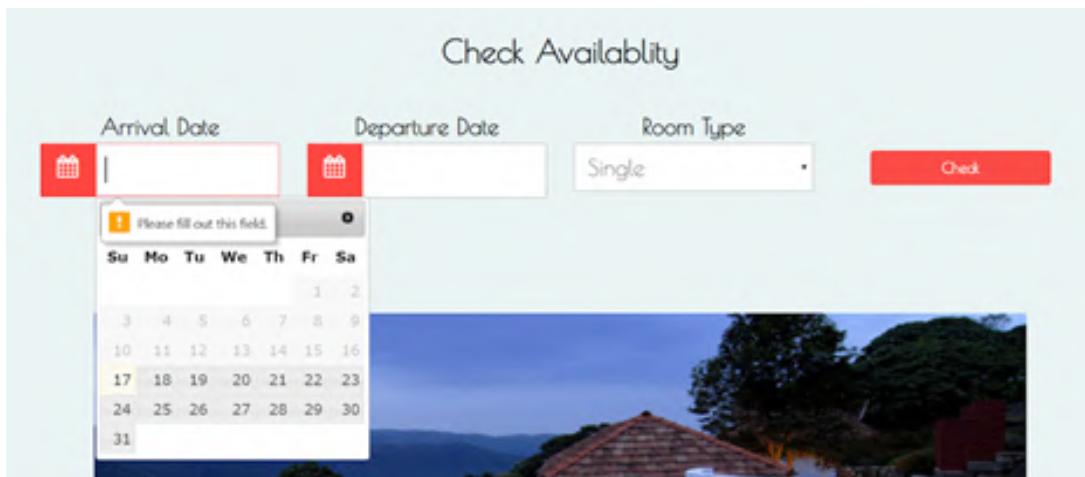
## Module Description

### 1. Index page

The index page or the front page is designed using HTML, CSS, Bootstrap. It consists of a navbar leading to the login and signup pages. Below the navbar lies the check availability column. Upon entering the arrival, departure date and room type, the website runs a check with the database. It returns if there are any rooms available, making the page incredibly user-friendly. The page also features a carousel. The calendar is created using JQuery and JQuery UI plugins.

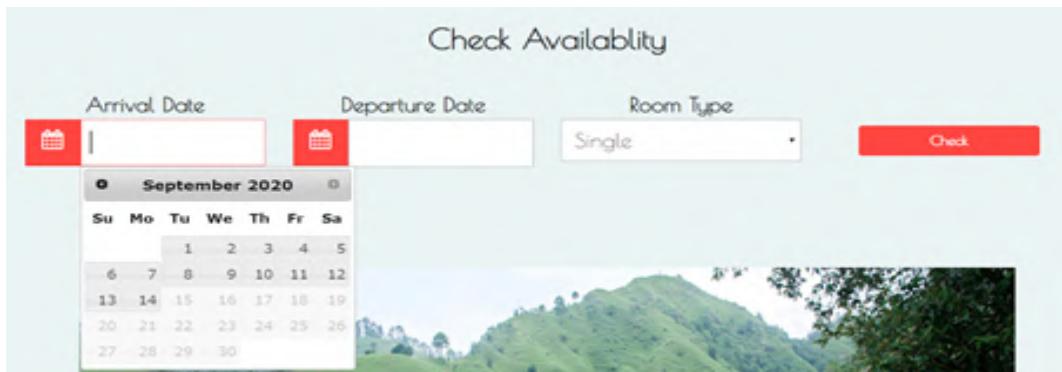
#### VALIDATIONS:

- The calendar shows the current date, and previous dates are disabled.
- The form does not take empty inputs.



The screenshot shows a "Check Availability" form. It includes fields for "Arrival Date" (with a red calendar icon), "Departure Date" (with a red calendar icon), and "Room Type" (a dropdown menu set to "Single"). A red "Check" button is on the right. A tooltip message "Please fill out this field." appears over the "Arrival Date" input field. Below the input fields is a calendar for September 2020, showing days from 1 to 30. The background of the form features a scenic image of a house and trees.

- The website ensures the user does not book for dates further 3 months from the current date.



The screenshot shows the same "Check Availability" form. The "Arrival Date" field is empty and has a red border. The "Departure Date" field contains "September 2020". The "Room Type" dropdown is set to "Single". The "Check" button is red. The calendar for September 2020 is displayed, showing days from 1 to 30. The background image of the house and trees is visible at the bottom.

## BACKEND

- Once the form is submitted (On clicking 'Check Availability'), the PHP code directs the page to MySQL database. A table named 'room\_status' consists of all the bookings done so far. A code runs to see if there are any available rooms for the given dates and displays a message accordingly.

```
C:\ XAMPP for Windows - mysql -u root -p
Meera@LAPTOP-JJDJ0VWW d:\xampp
# mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 291
Server version: 5.6.26 MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use dsadb;
Database changed
mysql> select * from room_status;
+-----+-----+-----+-----+-----+-----+-----+
| RSTATID | GID | STATUS | BOOKED_DATE_FROM | BOOKED_DATE_TO | ROOMID | REASON_FOR_CANCEL |
+-----+-----+-----+-----+-----+-----+-----+
|     1 |   5 | BOOKED | 2020-05-13 | 2020-05-18 |      5 | |
|     2 |   6 | BOOKED | 2020-05-19 | 2020-05-20 |      5 |
|     3 |   7 | BOOKED | 2020-05-22 | 2020-05-25 |      5 |
|     4 |   8 | BOOKED | 2020-05-20 | 2020-05-25 |      5 |
|     5 |   2 | BOOKED | 2020-05-11 | 2020-05-12 |      1 |
|     6 |   3 | BOOKED | 2020-05-11 | 2020-05-12 |      2 |
|     7 |   4 | BOOKED | 2020-05-11 | 2020-05-12 |      3 |
|     8 |   5 | BOOKED | 2020-05-11 | 2020-05-12 |      4 |
|     9 |   6 | BOOKED | 2020-05-11 | 2020-05-12 |      5 |
|    10 |   7 | BOOKED | 2020-05-20 | 2020-05-22 |      1 |
|    11 |   8 | BOOKED | 2020-05-10 | 2020-05-11 |      1 |
|    12 |   9 | BOOKED | 2020-05-10 | 2020-05-11 |      2 |
|    13 |  10 | CANCELLED | 2020-05-23 | 2020-05-23 |      1 | lockdown |
+-----+-----+-----+-----+-----+-----+-----+
```

## 2. SignUp page

Once the user checks the room availability, the user can proceed to the SignUp page. Upon Signing up, a mail is sent to the user's registered email id. This feature is executed using the class 'PHP mailer'.

### VALIDATION:

- All input fields are mandatory.

The screenshot shows a 'Sign Up' form with the following fields and errors:

- Email:** An input field labeled "Enter Email".
- Name:** An input field labeled "Name" with a red error message box containing the text "Please fill out this field." above it.
- Password:** An input field labeled "Password".
- Sign Up:** A red button labeled "Sign Up".
- Link:** A link "Have an account? [Login](#)".

- The same email can not be registered twice.

This email is already registered!!

The screenshot shows a 'Sign Up' form with the following fields and errors:

- Email:** An input field labeled "Enter Email".
- Name:** An input field labeled "Name".
- Password:** An input field labeled "Password".
- Sign Up:** A red button labeled "Sign Up".
- Link:** A link "Have an account? [Login](#)".

A red error message "This email is already registered!!" is displayed below the input field.

- The field 'email' makes sure the given input is of the required format.

# Sign Up

Email

! Please include an '@' in the email address. 'meera' is missing an '@'.

Password

Sign Up

## BACKEND:

Upon form submission, PHP code directs the page to the MySQL database. The inputs are inserted into a table named 'signup'. The table consists of a column named 'Personid' which is auto-incremented and unique to every guest.

```
23 | 20 | BOOKED | 2020-05-20 | 2020-05-21 | 1 |
24 | 21 | CANCELLED | 2020-07-23 | 2020-07-24 | 1 | VIP arrival
25 | 22 | BOOKED | 2020-07-23 | 2020-07-24 | 2 |
26 | 23 | BOOKED | 2020-07-23 | 2020-07-24 | 3 |
27 | 24 | BOOKED | 2020-07-23 | 2020-07-24 | 4 |
28 | 25 | BOOKED | 2020-05-30 | 2020-06-04 | 2 |
30 | 27 | BOOKED | 2020-05-22 | 2020-05-22 | 1 |
31 | 28 | BOOKED | 2020-05-22 | 2020-05-22 | 2 |
32 | 29 | BOOKED | 2020-05-21 | 2020-05-25 | 3 |
33 | 30 | BOOKED | 2020-05-21 | 2020-05-25 | 4 |
34 | 31 | BOOKED | 2020-05-22 | 2020-05-22 | 3 |
35 | 32 | BOOKED | 2020-05-21 | 2020-05-21 | 1 |
36 | 33 | BOOKED | 2020-05-21 | 2020-05-21 | 2 |
37 | 34 | CANCELLED | 2020-05-21 | 2020-05-22 | 4 | lockdown
+
+-----+-----+-----+-----+-----+
36 rows in set (0.00 sec)

mysql> select * from signup;
+-----+-----+-----+-----+
| Personid | Email | Name | Password |
+-----+-----+-----+-----+
| 20 | reachlmeera@gmail.com | Meera_L | qwerty |
| 13 | reachpurnaa@gmail.com | purnaa_L | asdfgh |
| 14 | hotelresidencia30@gmail.com | admin | admin |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>
```

### 3. Login Page

After signing up, the user can login to access his dashboard.

VALIDATION:

- All input fields are mandatory.
- Until username and password are verified, the user can not access the dashboard.

Invalid Credentials!!

The image shows a simple login interface. At the top center, it says "Login". Below that is a label "Email" followed by an empty input field. Below that is a label "Password" followed by another empty input field. At the bottom of the form is a red rectangular button with the word "Login" in white. In the bottom-left corner of the form area, there is a link in red text that reads "New User? Sign Up".

BACKEND:

- The input details(Email and Password) are stored in a temporary variable. These are credentials compared with the 'signup' table in the database. If they match, the user is allowed to access his dashboard.

### 4. User Dashboard

Upon logging in, the user dashboard pops up. This dashboard consists of all the requests he has submitted. The page also has a button named 'New Request', which allows the user to submit a request for booking rooms for his desired dates.

If rooms are available for requested dates, a confirmation mail is sent to the user, and a message is displayed. In case the rooms are booked, a regret mail is sent along with the display of a message.

The room allotment is based on queue data structure (First Come, First Serve).

The user dashboard is based on stack data structure. Newest request pops up first.

#### VALIDATION:

- All input fields are mandatory.
- The field 'phone number' makes sure the given input is of the required format.

Welcome Meera L |

Arrival Date	2020/05/22	Departure Date	2020/05/24
Room Type	Single		
Phone Number	123650C	<input type="text"/> Please match the requested format. <small>(e.g., +91 9876543210)</small>	
Address	<input type="text"/>		
<input type="button" value="Submit"/>			

- Use of session variable for login page. The session variable is destroyed once the user logs out. This ensures that the user has to log in everytime he wants to access the dashboard.

#### BACKEND

- Room availability is checked for the requested dates. Only if rooms are available, the guest details are entered into a database table named 'guestdetails2' else the data is discarded.

```
mysql> select * from guestdetails2;
+-----+-----+-----+-----+-----+-----+
| gid | arrival_date | departure_date | room_type | phone_number | address | Personid |
+-----+-----+-----+-----+-----+-----+
| 2 | 2020-05-11 | 2020-05-12 | single | 9741837315 | #427/B, 14th Cross, 15th Main, Vijayanagar 3rd Stage | 7 |
| 3 | 2020-05-11 | 2020-05-12 | single | 9741837315 | #427/B, 14th Cross, 15th Main, Vijayanagar 3rd Stage | 7 |
| 4 | 2020-05-11 | 2020-05-12 | single | 9741837315 | #427/B, 14th Cross, 15th Main, Vijayanagar 3rd Stage | 7 |
| 5 | 2020-05-11 | 2020-05-12 | single | 9741837315 | #427/B, 14th Cross, 15th Main, Vijayanagar 3rd Stage | 7 |
| 6 | 2020-05-11 | 2020-05-12 | twin | 9741837315 | #427/B, 14th Cross, 15th Main, Vijayanagar 3rd Stage | 7 |
| 7 | 2020-05-20 | 2020-05-22 | single | 0984406858 | Cheluvamba Mansion, Valmiki Rd, Opp. Railway Museum | 10 |
```

## 5. Admin Panel

The admin has a dedicated dashboard. The salient features of admin panel include

- Process Request
- View Available Rooms
- Room Master Maintenance
- Receipt Generation
- Block Room
- Release Room
- View Old Request
- Stayed People List

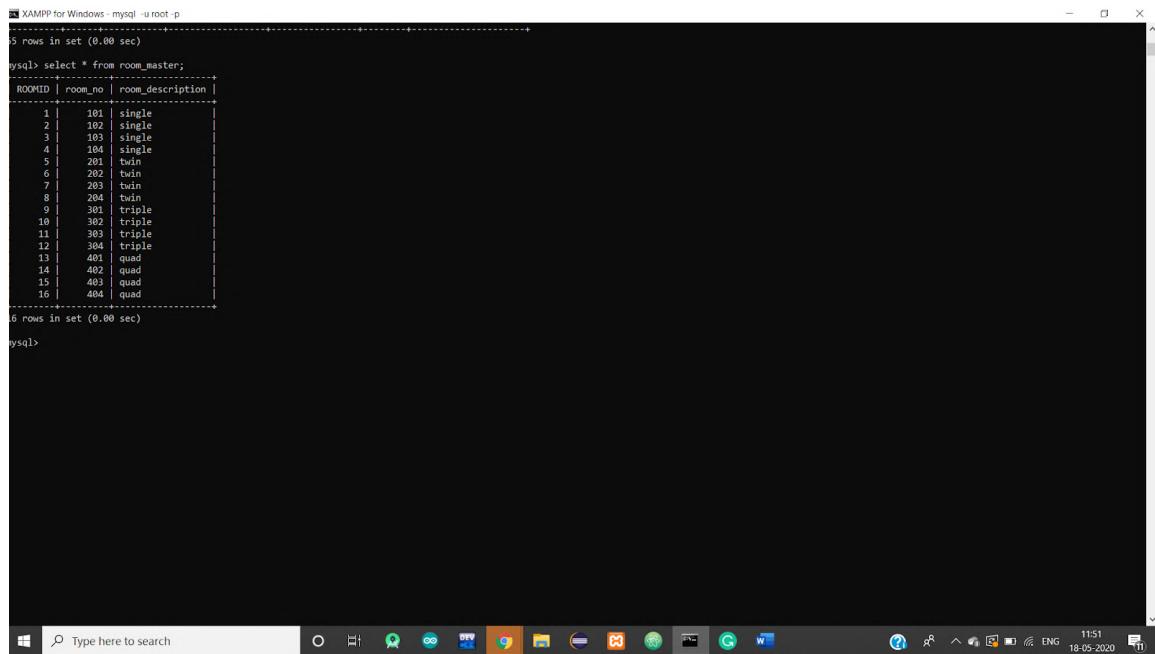
### Process Request

Admin is given the privilege to view all the requests from current date and further dates. The admin can print the guest details in a pdf format. The admin can also cancel the booked rooms. This is updated in the database table named 'room\_status'.

RSTATID	GID	STATUS	BOOKED_DATE_FROM	BOOKED_DATE_TO	ROOMID	REASON_FOR_CANCEL
1	5	BOOKED	2020-05-13	2020-05-18	5	
2	6	BOOKED	2020-05-19	2020-05-20	5	
3	7	BOOKED	2020-05-22	2020-05-25	5	
4	8	BOOKED	2020-05-20	2020-05-25	5	
5	2	BOOKED	2020-05-11	2020-05-12	1	
6	3	BOOKED	2020-05-11	2020-05-12	2	
7	4	BOOKED	2020-05-11	2020-05-12	3	
8	5	BOOKED	2020-05-11	2020-05-12	4	
9	6	BOOKED	2020-05-11	2020-05-12	5	
10	7	BOOKED	2020-05-20	2020-05-22	1	
11	8	BOOKED	2020-05-10	2020-05-11	1	
12	9	BOOKED	2020-05-10	2020-05-11	2	
13	10	CANCELLED	2020-05-23	2020-05-23	1	lockdown
14	11	CANCELLED	2020-05-23	2020-05-23	2	Resort maintenance
15	12	BOOKED	2020-05-13	2020-05-14	1	
16	13	BOOKED	2020-05-12	2020-05-13	5	
17	14	BOOKED	2020-05-30	2020-05-30	1	
18	15	BOOKED	2020-05-29	2020-05-31	2	
19	16	BOOKED	2020-05-29	2020-05-31	3	
20	26	BOOKED	2020-05-22	2020-05-30	6	
21	18	CANCELLED	2020-05-16	2020-05-17	1	
22	19	BOOKED	2020-05-16	2020-05-17	2	
23	20	BOOKED	2020-05-20	2020-05-21	13	
24	21	CANCELLED	2020-07-23	2020-07-24	1	VIP arrival
25	22	BOOKED	2020-07-23	2020-07-24	2	
26	23	BOOKED	2020-07-23	2020-07-24	3	
27	24	BOOKED	2020-07-23	2020-07-24	4	
28	25	BOOKED	2020-05-30	2020-06-04	2	
30	27	BOOKED	2020-05-22	2020-05-22	1	
31	28	BOOKED	2020-05-22	2020-05-22	2	
32	29	BOOKED	2020-05-21	2020-05-25	3	
33	30	BOOKED	2020-05-21	2020-05-25	4	
34	31	BOOKED	2020-05-22	2020-05-22	3	
35	32	BOOKED	2020-05-21	2020-05-21	1	
36	33	BOOKED	2020-05-21	2020-05-21	2	
37	34	CANCELLED	2020-05-21	2020-05-22	4	lockdown
38	35	BOOKED	2020-05-28	2020-05-30	2	
39	36	BOOKED	2020-05-30	2020-05-31	5	
40	37	BOOKED	2020-06-01	2020-06-02	1	
41	38	BOOKED	2020-06-01	2020-06-02	2	
42	39	CANCELLED	2020-06-01	2020-06-02	3	Lockdown
43	40	BOOKED	2020-06-01	2020-06-02	4	
44	41	BOOKED	2020-05-30	2020-05-31	6	
45	42	BOOKED	2020-05-22	2020-05-23	13	
46	43	BOOKED	2020-05-30	2020-05-31	9	

## View Available Rooms

Upon choosing from and to date, the page directs itself showing all the available rooms for the chosen dates. The data is retrieved from the table named 'room\_status' and 'room\_master'.



```
XAMPP for Windows - mysql -u root -p
5 rows in set (0.00 sec)

mysql> select * from room_master;
+-----+-----+-----+
| ROOMID | room_no | room_description |
+-----+-----+-----+
| 1     | 101    | single          |
| 2     | 102    | single          |
| 3     | 103    | single          |
| 4     | 104    | single          |
| 5     | 201    | twin            |
| 6     | 202    | twin            |
| 7     | 203    | twin            |
| 8     | 204    | twin            |
| 9     | 301    | triple          |
| 10    | 302    | triple          |
| 11    | 303    | triple          |
| 12    | 304    | triple          |
| 13    | 401    | quad            |
| 14    | 402    | quad            |
| 15    | 403    | quad            |
| 16    | 404    | quad            |
+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

## Receipt Generation

The admin can generate a receipt for the guests staying. A PHP code runs finding all the guests staying on the current date and previous dates from the database. The number of days of stay is calculated, and hence the price is calculated. The receipt is generated in PDF format. This feature uses 'pdfclass'.

## Block Room

The admin has the right to block rooms for a certain period in case of emergencies/hotel maintenance or any other reason

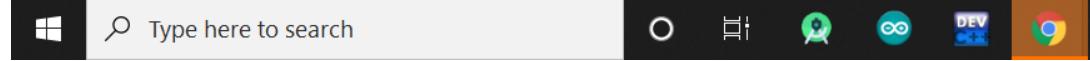
**BACKEND:**

A table named 'blocked\_dates' is created and all the blocked dates are inserted into it.

```
10 rows in set (0.00 sec)

mysql> select * from blocked_dates;
+----+-----+-----+-----+
| bid | from_date | to_date | reason | status |
+----+-----+-----+-----+
| 286 | 2020-05-18 | 2020-05-18 | test | 
| 287 | 2020-09-02 | 2020-09-05 | block | 
| 293 | 2020-09-10 | 2020-09-11 | dasara | 
| 295 | 2020-08-28 | 2020-08-29 | booked | 
+----+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql>
```



## Release Room

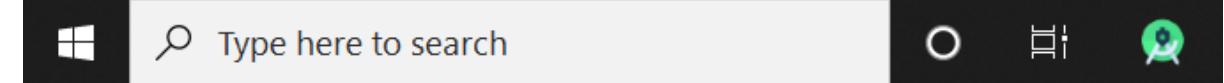
The blocked rooms can be released by the admin by clicking on the link given. This enables the user to book rooms again hassle free.

### BACKEND

The column status from the table 'booked\_dates' changes to 'RELEASED'.

```
mysql> select * from blocked_dates;
+----+-----+-----+-----+
| bid | from_date | to_date | reason | status |
+----+-----+-----+-----+
| 286 | 2020-05-18 | 2020-05-18 | test | Released
| 287 | 2020-09-02 | 2020-09-05 | block | 
| 296 | 2020-08-12 | 2020-08-14 | adsfa | 
| 293 | 2020-09-10 | 2020-09-11 | dasara | 
| 295 | 2020-08-28 | 2020-08-29 | booked | Released
+----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql>
```



## View Old Request

View old request directs to a page consisting of a list of all the requests received by the resort/hotel. This list also consists of cancelled requests. The data is fetched from the database using PHP code.

### BACKEND

```
mysql> select * from room_status;
+-----+-----+-----+-----+-----+-----+-----+
| RSTATID | GID | STATUS | BOOKED_DATE_FROM | BOOKED_DATE_TO | ROOMID | REASON_FOR_CANCEL |
+-----+-----+-----+-----+-----+-----+-----+
| 1 | 5 | BOOKED | 2020-05-13 | 2020-05-18 | 5 | null |
| 2 | 6 | BOOKED | 2020-05-19 | 2020-05-20 | 5 | null |
| 3 | 7 | BOOKED | 2020-05-22 | 2020-05-25 | 5 | null |
| 4 | 8 | BOOKED | 2020-05-20 | 2020-05-25 | 5 | null |
| 5 | 2 | BOOKED | 2020-05-11 | 2020-05-12 | 1 | null |
| 6 | 3 | BOOKED | 2020-05-11 | 2020-05-12 | 2 | null |
| 7 | 4 | BOOKED | 2020-05-11 | 2020-05-12 | 3 | null |
| 8 | 5 | BOOKED | 2020-05-11 | 2020-05-12 | 4 | null |
| 9 | 6 | BOOKED | 2020-05-11 | 2020-05-12 | 5 | null |
| 10 | 7 | BOOKED | 2020-05-20 | 2020-05-22 | 1 | null |
| 11 | 8 | BOOKED | 2020-05-10 | 2020-05-11 | 1 | null |
| 12 | 9 | BOOKED | 2020-05-10 | 2020-05-11 | 2 | null |
| 13 | 10 | CANCELLED | 2020-05-23 | 2020-05-23 | 1 | Lockdown |
| 14 | 11 | CANCELLED | 2020-05-23 | 2020-05-23 | 2 | Resort maintenance |
| 15 | 12 | BOOKED | 2020-05-13 | 2020-05-14 | 1 | null |
| 16 | 13 | BOOKED | 2020-05-12 | 2020-05-13 | 5 | null |
| 17 | 14 | BOOKED | 2020-05-30 | 2020-05-30 | 1 | null |
| 18 | 15 | BOOKED | 2020-05-29 | 2020-05-31 | 2 | null |
| 19 | 16 | BOOKED | 2020-05-29 | 2020-05-31 | 3 | null |
| 20 | 26 | BOOKED | 2020-05-22 | 2020-05-30 | 6 | null |
| 21 | 18 | CANCELLED | 2020-05-16 | 2020-05-17 | 1 | null |
| 22 | 19 | BOOKED | 2020-05-16 | 2020-05-17 | 2 | null |
+-----+-----+-----+-----+-----+-----+-----+
```

```
mysql> select * from signup;
+-----+-----+-----+-----+
| Personid | Email | Name | Password |
+-----+-----+-----+-----+
| 20 | reachlmeera@gmail.com | Meera L | qwerty |
| 21 | harry@gmail.com | Harry | harry |
| 22 | anna@gmail.com | Anna | anna |
| 23 | meghan875@gmail.com | Meghan | meghan |
| 13 | reachpurnaa@gmail.com | purnaa L | asdfgh |
| 14 | hotelresidencia30@gmail.com | admin | admin |
+-----+-----+-----+-----+
6 rows in set (0.00 sec)

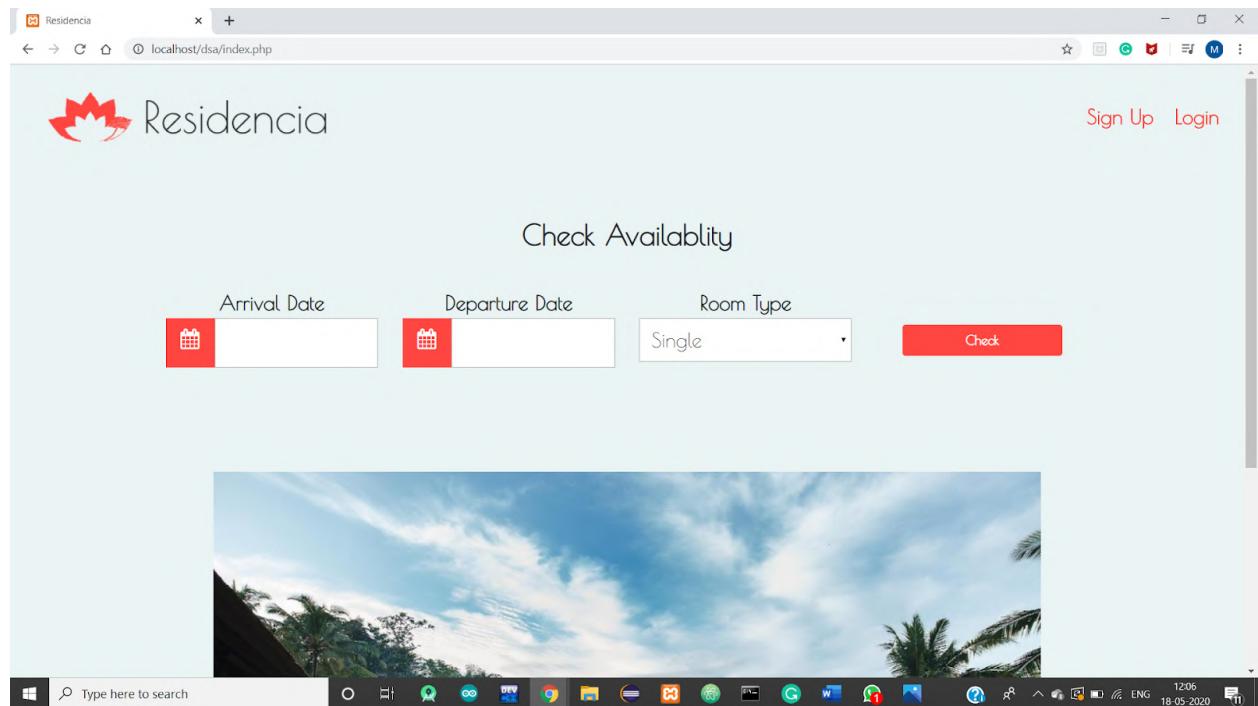
mysql>
```

## Stayed People List

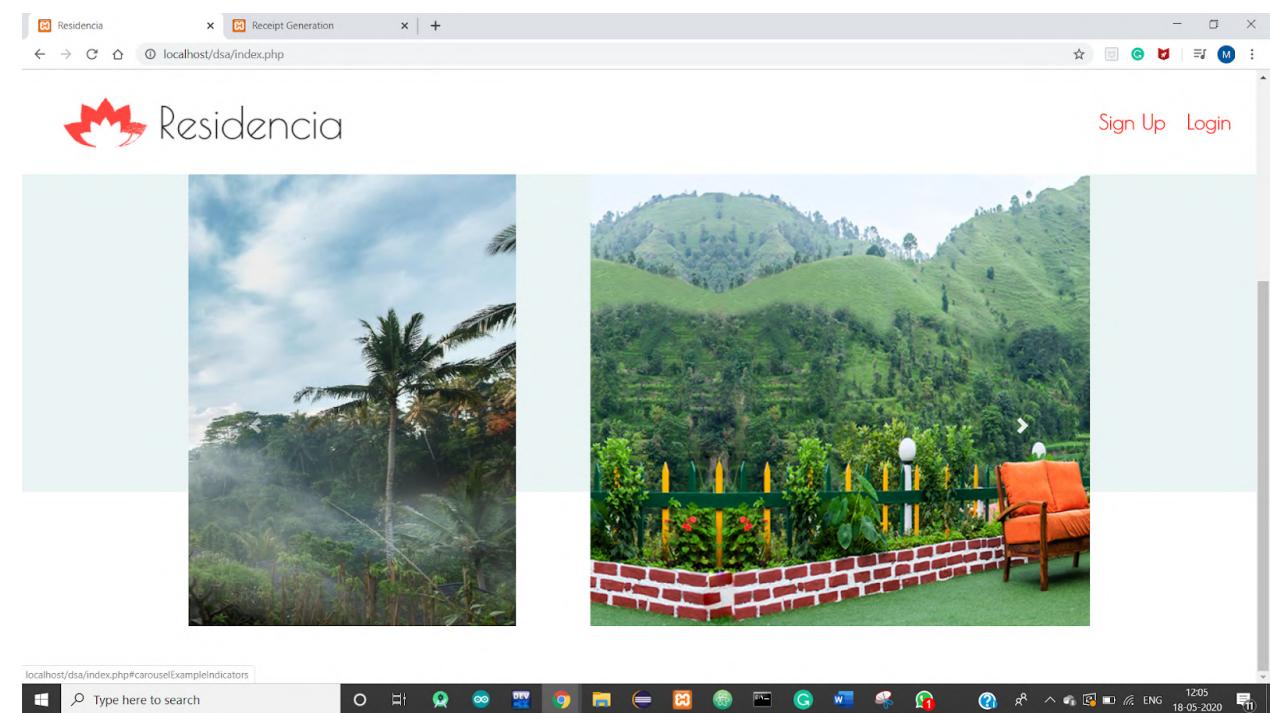
The list includes the history of all people who stayed in the resort/hotel. The list is fetched from the database using PHP code. The list also shows the room occupied by the guest. The table used for the data is 'room\_status' and 'signup'.

## Results and Discussion

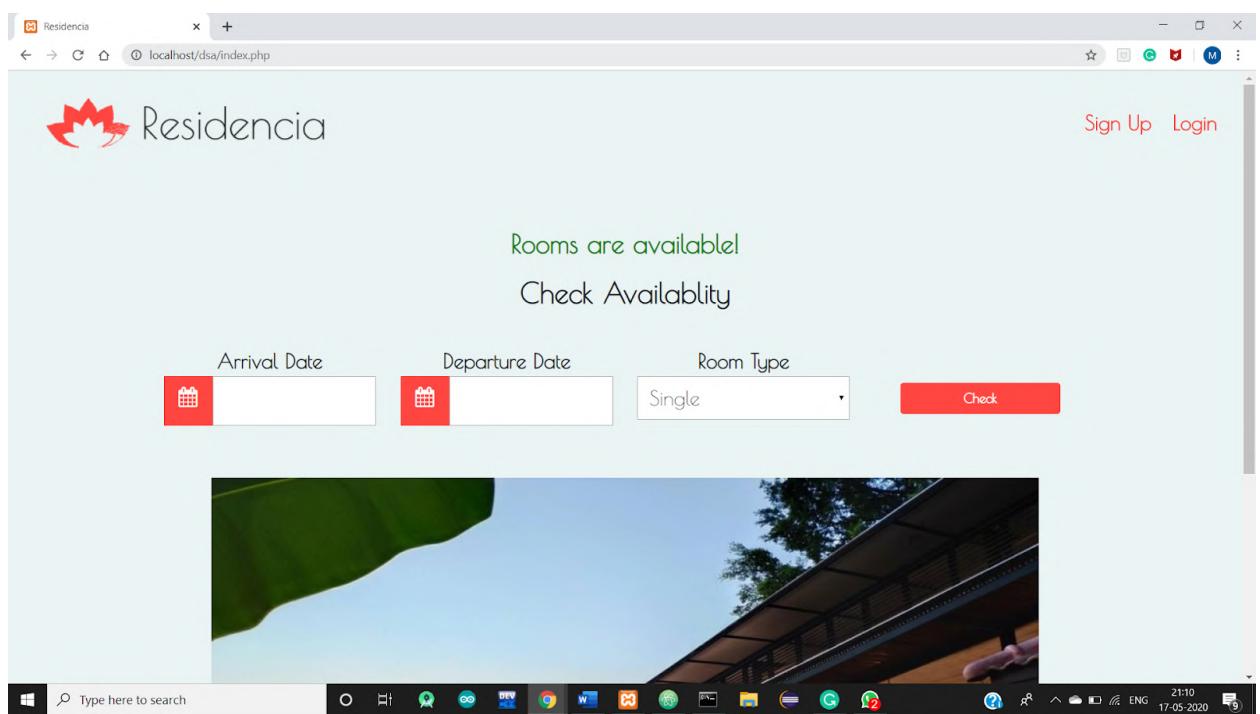
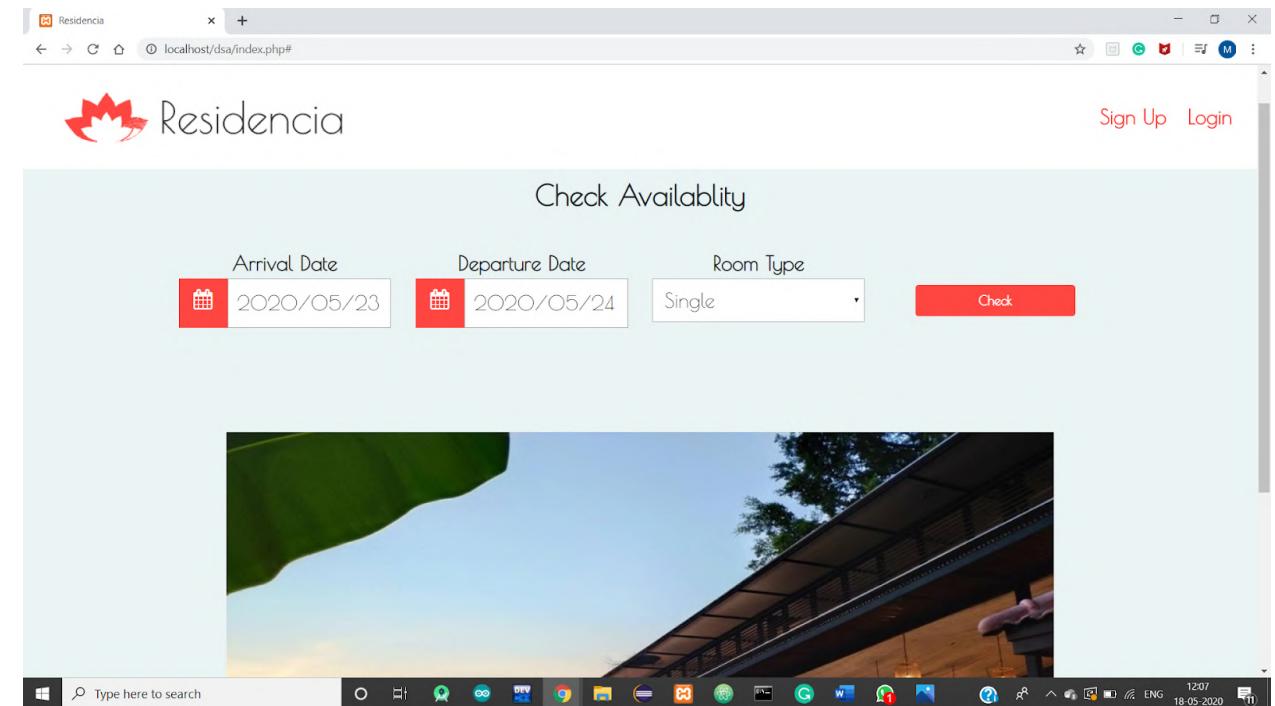
### 1. Index Page



### 2. Carousel - The index page includes carousel, to display stunning images of the resort



### 3. Checking Availability - Displays a datepicker



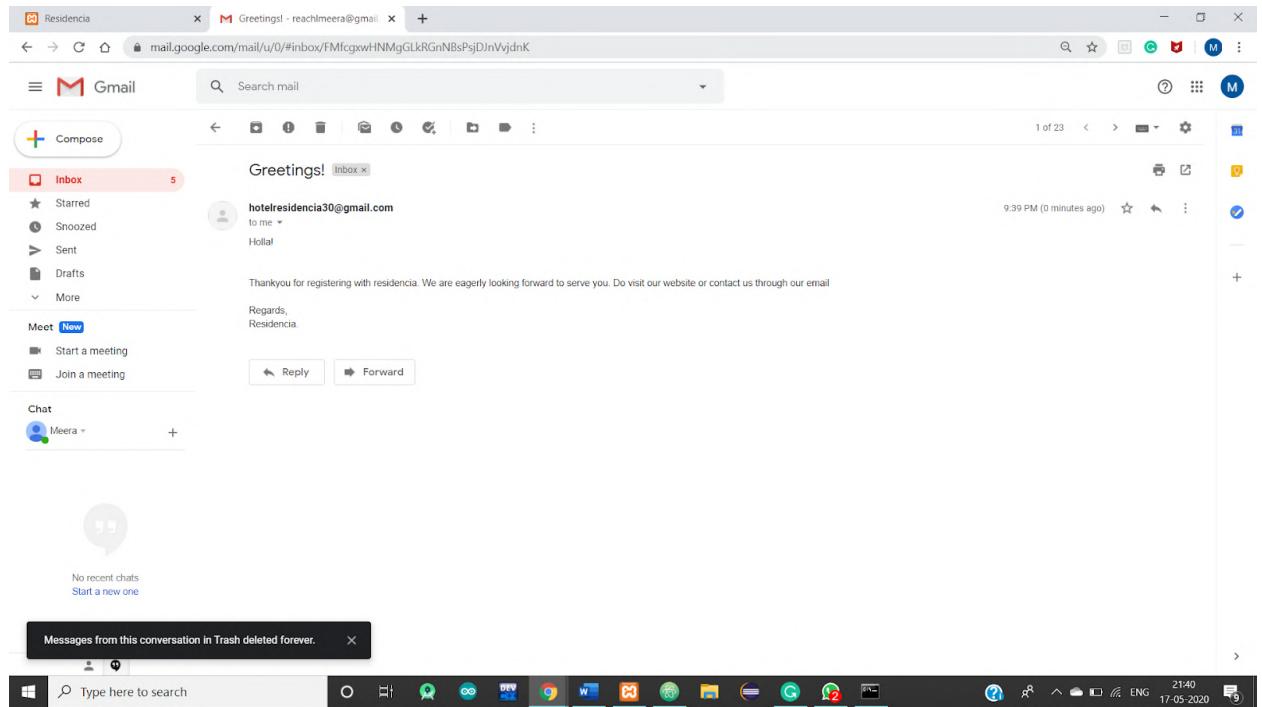
#### 4. Sign Up Page

A screenshot of a web browser window showing the Residencia sign-up page. The title bar says "Residencia" and the address bar shows "localhost/dsa/signup.php". The page features a logo of a red maple leaf and the word "Residencia". A "Sign Up" button is at the top right of a form area. The form fields are labeled "Email", "Name", and "Password". The "Email" field contains "reachlmeera@gmail.com", the "Name" field contains "Meera L", and the "Password" field contains ".....". Below the form is a "Sign Up" button and a link "Have an account? Login". The browser's taskbar at the bottom shows various open tabs and system icons.

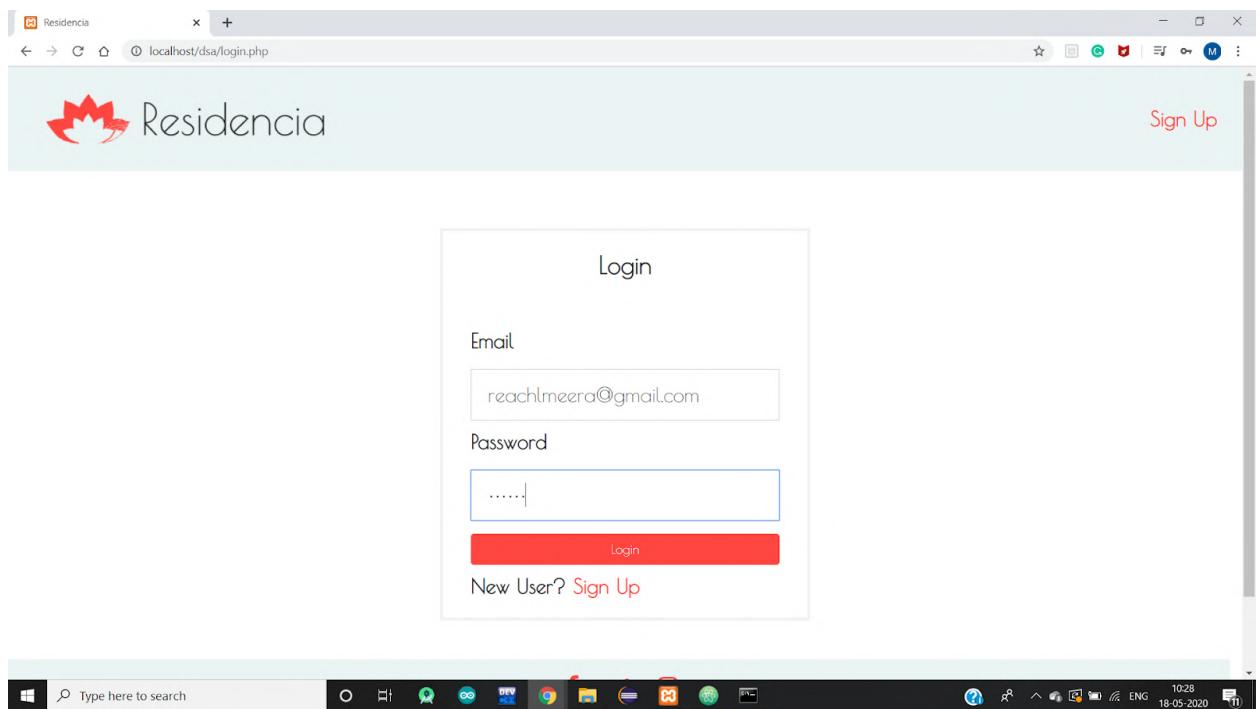
#### 5. After Signing Up

A screenshot of a web browser window showing the Residencia sign-up page after a successful registration. The title bar says "Residencia" and the address bar shows "localhost/dsa/signup.php". The page features a logo of a red maple leaf and the word "Residencia". A message "Thankyou for registering with us !" is displayed above the sign-up form. The sign-up form fields are labeled "Email", "Name", and "Password". The "Email" field has the placeholder "Enter Email", the "Name" field has the placeholder "Name", and the "Password" field has the placeholder "Password". The browser's taskbar at the bottom shows various open tabs and system icons.

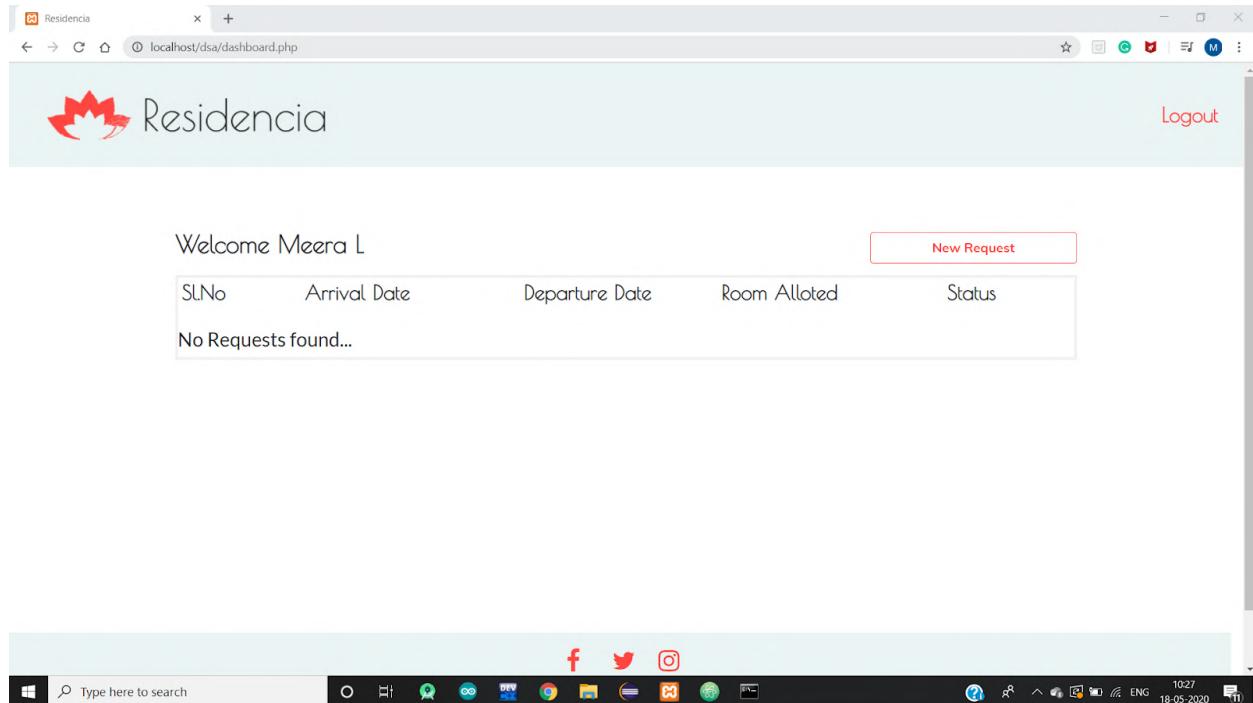
## 6. After signing up, a mail is sent to the user



## 7. Login page

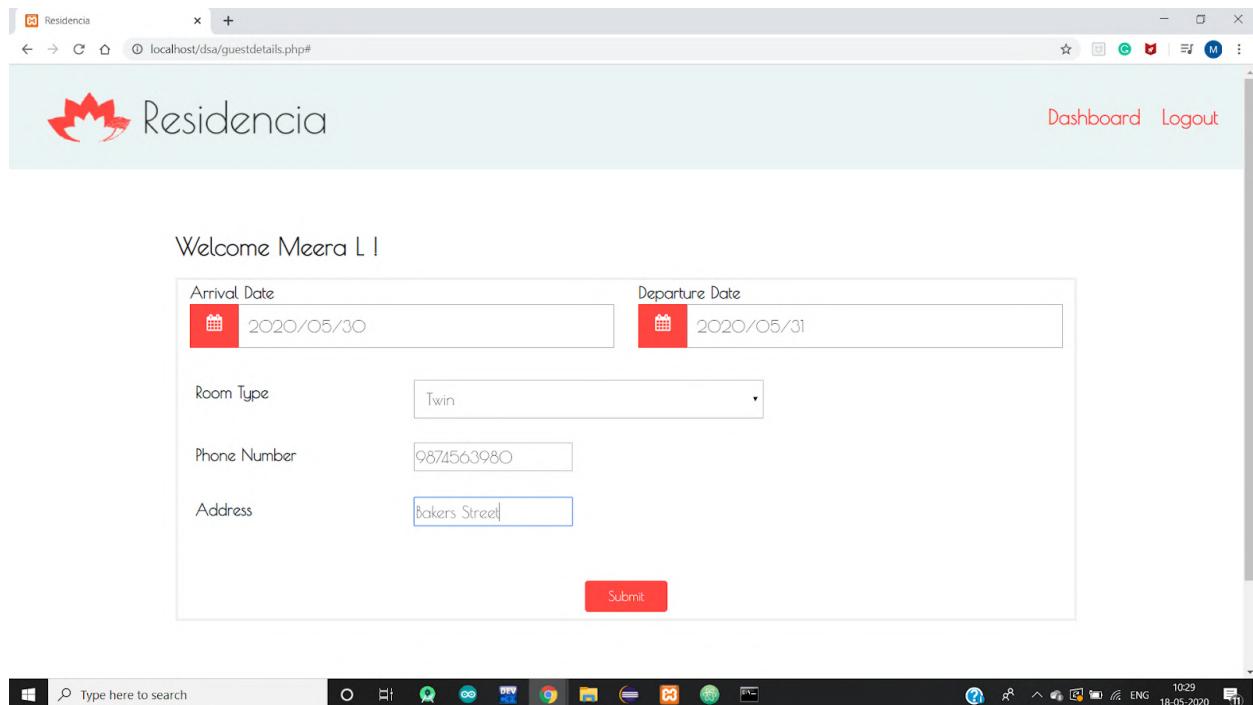


## 8. Dashboard of the user



The screenshot shows a web browser window titled "Residencia" with the URL "localhost/dsa/dashboard.php". The page features a logo of a red maple leaf and the word "Residencia". A "Logout" link is in the top right. A welcome message "Welcome Meera L" is displayed. A "New Request" button is in the top right of the main area. Below it is a table with columns: Sl.No, Arrival Date, Departure Date, Room Alloted, and Status. A message "No Requests found..." is shown in the table body.

## 9. On Clicking New Request



The screenshot shows a web browser window titled "Residencia" with the URL "localhost/dsa/guestdetails.php#". The page features a logo of a red maple leaf and the word "Residencia". A "Dashboard" and "Logout" link are in the top right. A welcome message "Welcome Meera L |" is displayed. The main area contains a form with fields for Arrival Date (set to 2020/05/30), Departure Date (set to 2020/05/31), Room Type (set to Twin), Phone Number (set to 9874563980), and Address (set to Bakers Street). A "Submit" button is at the bottom right.

## 10. After form Submission

- a) If rooms are available

We have received your booking! Kindly check your mail!

Welcome Meera L !

Arrival Date

Departure Date

Room Type

Phone Number

Address

Confirmation mail is sent

Booking Confirmation

hoteleresidencia30@gmail.com to me 10:14 AM (16 minutes ago)

Greetings!  
Your booking has been confirmed. We are eagerly awaiting your presence. Kindly Check your dashboard for more details

Regards,  
Residencia

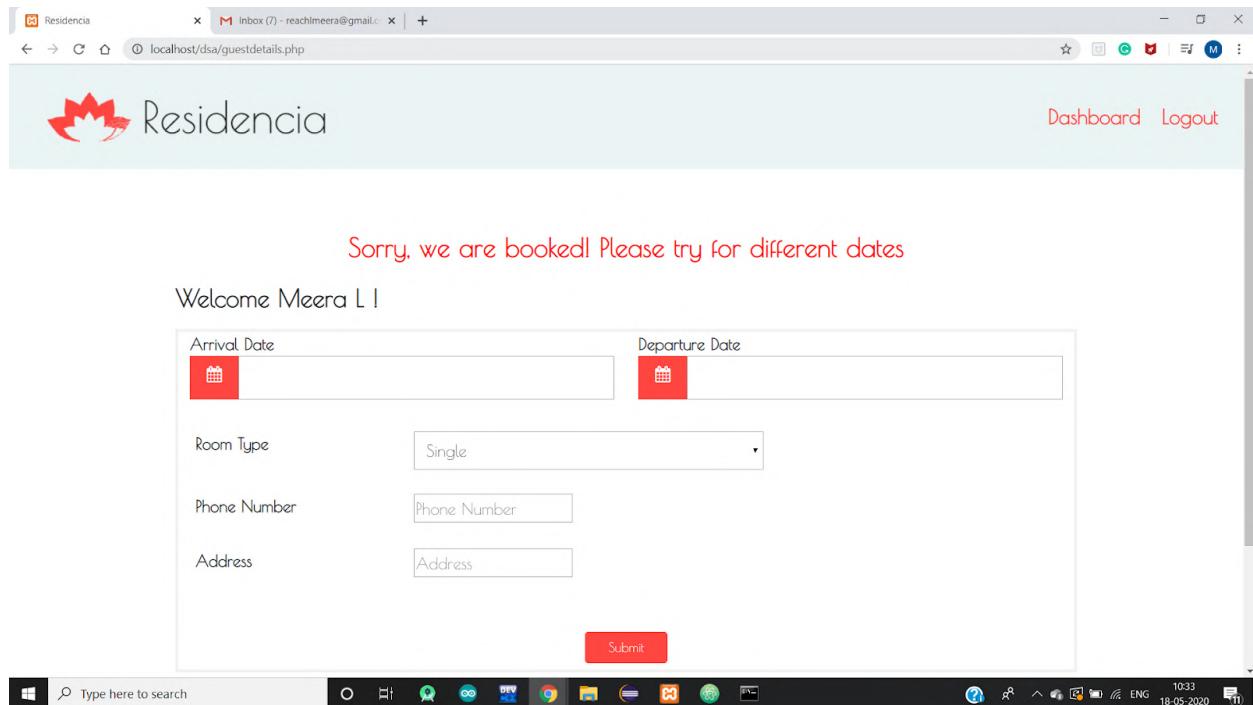
hoteleresidencia30@gmail.com to me 10:29 AM (0 minutes ago)

...

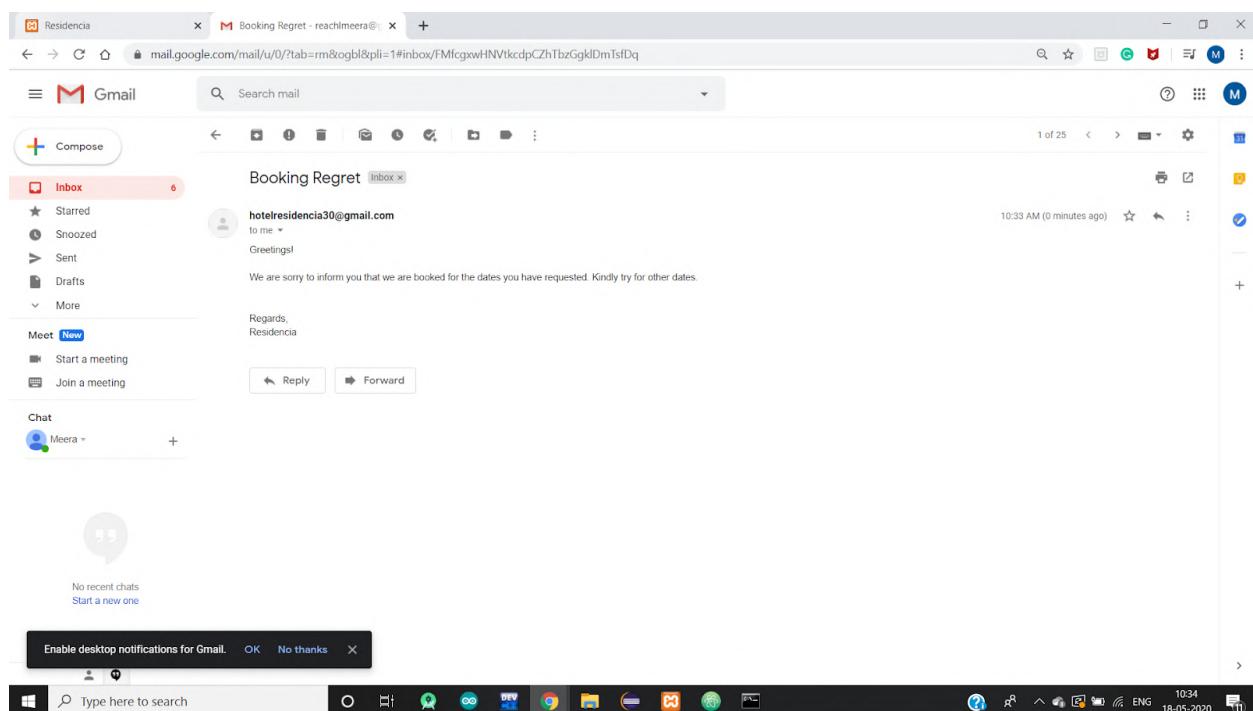
No recent chats  
Start a new one

Enable desktop notifications for Gmail. OK No thanks

b) If rooms are not available



Regret mail is sent





## 11. View of dashboard after submitting request

SL.No	Arrival Date	Departure Date	Room Alloted	Status
1	01-06-2020	02-06-2020	103	BOOKED
2	01-06-2020	02-06-2020	104	BOOKED
3	01-06-2020	02-06-2020	102	BOOKED
4	01-06-2020	02-06-2020	101	BOOKED
5	30-05-2020	31-05-2020	201	BOOKED

## 12. Admin Login

### 13. Admin Dashboard

Welcome Admin !

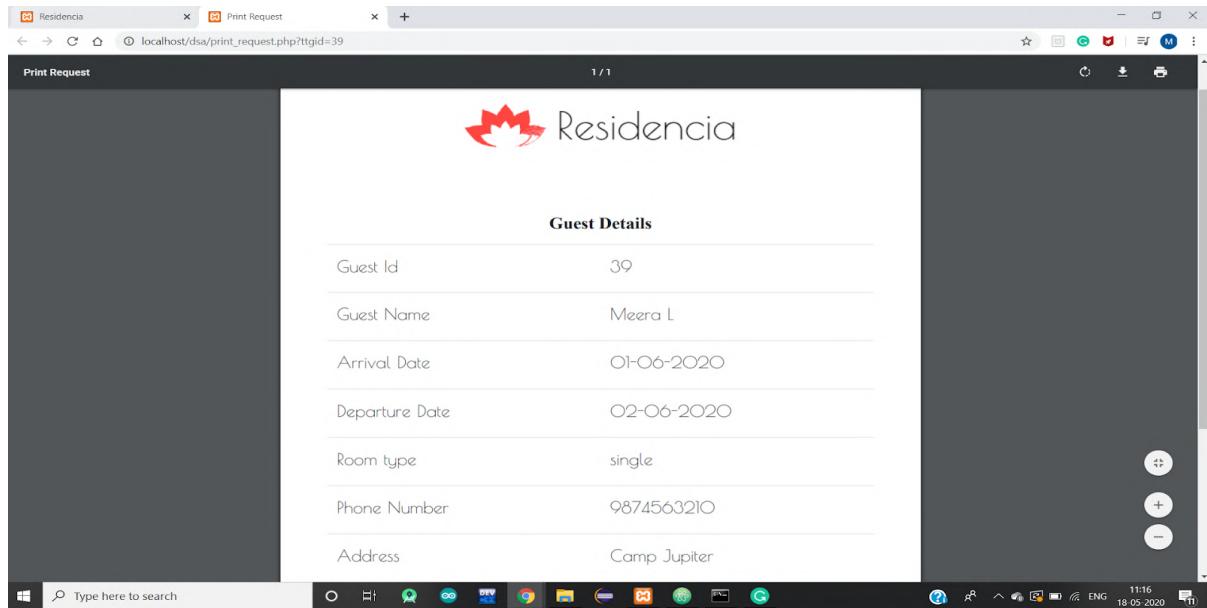
<a href="#">Process Request</a>	<a href="#">View Available Rooms</a>
<a href="#">Room Master Maintenance</a>	<a href="#">Receipt Generation</a>
<a href="#">Block Room</a>	<a href="#">Release Room</a>
<a href="#">View Old Request</a>	<a href="#">Stayed People List</a>

### 14. On clicking 'View Request'

Booking Requests Received

SL No	Guest Id	Name	Arrival Date	Departure Date	Status	View Requests	Cancel
1	39	Meera L	01-06-2020	02-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
2	40	Meera L	01-06-2020	02-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
3	25	purnaa L	30-05-2020	04-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
4	38	Meera L	01-06-2020	02-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
5	37	Meera L	01-06-2020	02-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
6	43	Meghan	30-05-2020	31-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
7	44	Harry	12-06-2020	13-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
8	45	Harry	16-07-2020	17-07-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
9	46	Anna	23-05-2020	24-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
10	47	Anna	29-08-2020	31-08-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
11	42	Anna	22-05-2020	23-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
12	41	Harry	30-05-2020	31-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
13	36	Meera L	30-05-2020	31-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>

a) Viewing the request in pdf form



b) Cancel a booking

The screenshot shows a web browser window for the "Residencia" system. The title bar indicates the URL is "localhost/dsa/cancel\_req.php?ttgid=39". The page header includes the "Residencia" logo, "Dashboard", and "Logout" links. The main content area is titled "Cancelling a booked accommodation". It displays the following details:

Details		
GuestID	:	39
Name	:	Meera L
Arrival Date	:	01-06-2020
Departure Date	:	02-06-2020
Cancel	:	<input checked="" type="radio"/> YES <input type="radio"/> NO
Reason for Cancelling	:	<input type="text" value="Lockdown"/>

At the bottom of the form is a "UPDATE" button.

### 15. After cancelling a request

The screenshot shows a web browser window for 'Residencia' with the URL 'localhost/dsa/viewrequest.php'. The page title is 'Booking Requests Received'. A table lists 13 rows of booking data. The 1st row has a status of 'CANCELLED' in yellow, while all other rows have 'BOOKED' in green. The last column contains links labeled 'Click here' and 'Yes/No'.

Sl. No	Guest Id	Name	Arrival Date	Departure Date	Status	View Requests	Cancel
1	39	Meera L	01-06-2020	02-06-2020	CANCELLED	<a href="#">Click here</a>	<a href="#">CANCELLED</a>
2	40	Meera L	01-06-2020	02-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
3	25	purnaa L	30-05-2020	04-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
4	38	Meera L	01-06-2020	02-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
5	37	Meera L	01-06-2020	02-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
6	43	Meghan	30-05-2020	31-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
7	44	Harry	12-06-2020	13-06-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
8	45	Harry	16-07-2020	17-07-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
9	46	Anna	23-05-2020	24-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
10	47	Anna	29-08-2020	31-08-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
11	42	Anna	22-05-2020	23-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
12	41	Harry	30-05-2020	31-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>
13	36	Meera L	30-05-2020	31-05-2020	BOOKED	<a href="#">Click here</a>	<a href="#">Yes/No</a>



### 16. Viewing Old Requests(including cancelled ones)

The screenshot shows a web browser window for 'Residencia' with the URL 'localhost/dsa/oldrequest.php'. The page title is 'Old Requests'. A table lists 10 rows of booking data. The 9th row has a status of 'CANCELLED' in red, while all other rows have 'BOOKED' in green. The last column contains links labeled 'Click here' and 'Yes/No'.

Sl. No	Guest Id	Name	Arrival Date	Departure Date	Status
1	2		11-05-2020	12-05-2020	BOOKED
2	3		11-05-2020	12-05-2020	BOOKED
3	4		11-05-2020	12-05-2020	BOOKED
4	5		11-05-2020	12-05-2020	BOOKED
5	6		11-05-2020	12-05-2020	BOOKED
6	9		10-05-2020	11-05-2020	BOOKED
7	12		13-05-2020	14-05-2020	BOOKED
8	13		12-05-2020	13-05-2020	BOOKED
9	18		16-05-2020	17-05-2020	CANCELLED
10	19		16-05-2020	17-05-2020	BOOKED



**Receipt Generation**

SL No	Guest Id	Name	Arrival Date	Departure Date	Status	Generate Receipt
2	48	Anna	18-05-2020	18-05-2020	BOOKED	<a href="#">Click here</a>
3	49	Anna	18-05-2020	18-05-2020	BOOKED	<a href="#">Click here</a>
4	50	Anna	18-05-2020	18-05-2020	BOOKED	<a href="#">Click here</a>
5	51	Harry	18-05-2020	18-05-2020	BOOKED	<a href="#">Click here</a>
6	52	Harry	18-05-2020	18-05-2020	BOOKED	<a href="#">Click here</a>
7	53	Meghan	18-05-2020	18-05-2020	BOOKED	<a href="#">Click here</a>

## 18. Printing Receipt in PDF format

**Receipt Generation**

1/1

**Guest Details**

Guest Id	53
Guest Name	Meghan
Arrival Date	18-05-2020
Departure Date	18-05-2020
Room type	quad
Duration of Stay	1 days
Price	INR 3500
Phone Number	8650048630
Address	Bakers Street

19. List of people who stayed along with the room number they stayed

The screenshot shows a web browser window with the title bar "Residencia" and "Receipt Generation". The address bar shows "localhost/dsa/stayedppl.php". The main content area displays a table titled "Booking Requests Received" with the following data:

Sl. No.	Guest Id	Name	Arrival Date	Departure Date	Room type	Room number
2	48	Anna	18-05-2020	18-05-2020	single	101
3	49	Anna	18-05-2020	18-05-2020	twin	201
4	50	Anna	18-05-2020	18-05-2020	triple	301
5	51	Harry	18-05-2020	18-05-2020	twin	202
6	52	Harry	18-05-2020	18-05-2020	triple	302
7	53	Meghan	18-05-2020	18-05-2020	quad	401

20. Blocking rooms

The screenshot shows a web browser window with the title bar "Residencia". The address bar shows "localhost/dsa/block\_dates.php". The main content area displays a form titled "Block Rooms" with the following fields:

- From Date: 2020/09/10
- To Date: 2020/09/12
- Reason: We're booked!
- Block button

At the bottom of the page, there is a footer with social media icons (Facebook, Twitter, Instagram) and an email address: hotelresidencia30@gmail.com. The taskbar at the bottom of the screen shows various application icons.

All rooms blocked on the given dates....

Block Rooms

From Date

To Date

Reason

**Block**

f t i hotelresidencia30@gmail.com

## 21. Releasing the blocked rooms

SL No	From Date	To Date	Reason	Release Block
1	02-09-2020	05-09-2020	We are booked!	<a href="#">Click here</a>
2	12-08-2020	14-08-2020	Resort maintenance	<a href="#">Click here</a>
3	10-09-2020	11-09-2020	dasara	<a href="#">Click here</a>

f t i

The screenshot shows a browser window titled "Residencia" with the URL "localhost/dsa/release\_block.php?bid=287". The page displays a red maple leaf logo and the word "Residencia". On the right, there are "Dashboard" and "Logout" links. A message "All rooms released on the selected dates...." is shown in blue. Below it, a section titled "Blocked dates" contains a table:

SL No	From Date	To Date	Reason	Release Block
1	12-08-2020	14-08-2020	Resort maintenance	<a href="#">Click here</a>
2	10-09-2020	11-09-2020	dasara	<a href="#">Click here</a>

The Windows taskbar at the bottom shows various pinned icons and the date/time "18-05-2020 20:14".

## 22. Viewing Available rooms for selected date

The screenshot shows a browser window titled "Residencia" with the URL "localhost/dsa/availablerooms.php#". The page displays a red maple leaf logo and the word "Residencia". On the right, there are "Dashboard" and "Logout" links. A section titled "Available Rooms" contains input fields for "From Date" (set to "2020/06/01") and "To Date" (set to "2020/06/02"), along with a "View Rooms" button. Below this, there is a footer with social media icons (Facebook, Twitter, Instagram) and an email address "hotelresidencia30@gmail.com". The Windows taskbar at the bottom shows various pinned icons and the date/time "18-05-2020 20:15".

The screenshot shows a web browser window titled "Residencia" with the URL "localhost/dsa/availablerooms.php". The page header includes a logo of a red maple leaf and the word "Residencia", along with "Dashboard" and "Logout" links. The main content area is titled "Available Rooms" and features two date input fields: "From Date" and "To Date", each accompanied by a calendar icon. A red "View Rooms" button is positioned to the right of the date fields. Below these fields, a list of room numbers is displayed: Room No. 301, Room No. 302, Room No. 303, Room No. 304, Room No. 401, Room No. 402, Room No. 403, and Room No. 404. At the bottom of the page, there are social media sharing icons for Facebook, Twitter, and Instagram, and an email address: "hotelresidencia30@gmail.com". The browser's taskbar at the bottom shows various open applications, and the system tray indicates the date and time as "18-05-2020 20:16".



## Time Complexity and its Graphical Representation

- INSERTING DATA INTO MYSQL

```
mysql> CREATE TABLE aaa(
->   id INT AUTO_INCREMENT PRIMARY KEY,
->   name VARCHAR(30) NOT NULL,
->   email VARCHAR(50)
-> );
Query OK, 0 rows affected (0.18 sec)

mysql> INSERT into aaa(id,name,email) values ("1","Annabeth","anna@gmail.com");
Query OK, 1 row affected (0.02 sec)

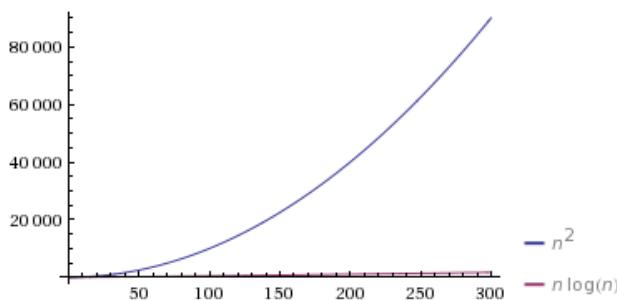
mysql>
```



The time required for inserting a row is determined by the following factors, where the numbers indicate approximate proportions:

- Connecting: (3)
- Sending query to server: (2)
- Parsing query: (2)
- Inserting row: ( $1 \times$  size of row)
- Inserting indexes: ( $1 \times$  number of indexes)
- Closing: (1)

The time complexity for the insertion of indexes is  $O(\log N)$  assuming B-tree indexes.



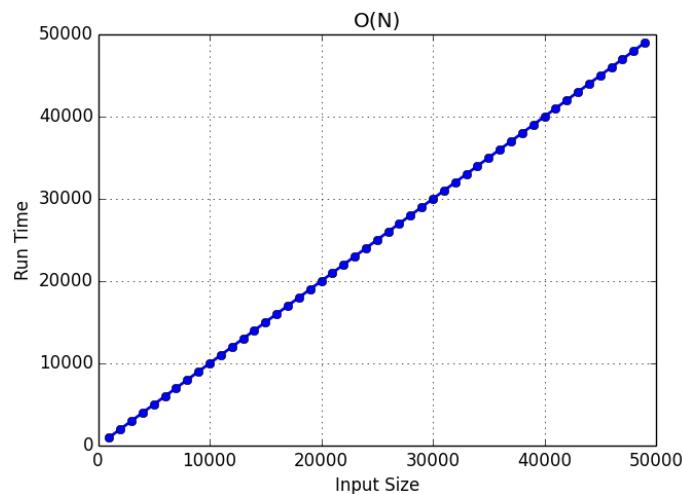
- USE OF SEARCH QUERY IN MYSQL

```
mysql> select * from signup;
+-----+-----+-----+
| Personid | Email           | Name    | Password |
+-----+-----+-----+
|     20   | reachlmeera@gmail.com | Meera L | qwerty   |
|     21   | harry@gmail.com      | Harry   | harry    |
|     22   | anna@gmail.com       | Anna    | anna    |
|     23   | meghan875@gmail.com | Meghan  | meghan   |
|     13   | reachpurnaa@gmail.com | purnaa L | asdfgh  |
|     14   | hotelresidencia30@gmail.com | admin   | admin    |
+-----+-----+-----+
6 rows in set (0.00 sec)

mysql>
```

The time complexity is  $O(n)$

Graph:



## References

1. C Niyizamwiyitira, Performance Evaluation Of Sql And Nosql Database Management Systems In A Cluster, *IJDMS Journal ,Vol 9,No.5 , December 2017 , PP. 1-24*  
DOI : 10.5121/ijdms.2017.9601
2. Elizabeth R. Towell Northern William D. Haseman , Implementation of an Interface to Multiple Databases, *JDM, Vol 6, No. 2, September 27, PP. 33-39*
3. Nihan Kesim, Marek Segot, Implementing an Object - Oriented Deductive Database Using Temporal Reasoning, *JDM, Vol 7, Issue 4, August 29,1999, PP 21-34*
4. Latifur Khan, Dennis McLEod, Cyrus Shahabi, An adaptive probe-Based Technique to Optimize Join Queries in Distributed Internet Databases, *JDM, Vol 12, Issue 4, December 2001, PP. 3-14*
5. Dinesh Batra, Conceptual Data Modelling Patterns: Representation and Validation, *JDM, Volume 16, Issue 2, April 2005, PP. 84-85*
6. Athman Bouguettaya, Zaki Malik, Lori Kroff, A Scalable Middleware for Web Databases, *JDM, Vol 17, Issue 4, March 2003, PP. 20-46*

- 
7. Zhen He, Jerome Darmont, Evaluating the Dynamic Behaviour of Database Applications, *JDM, Vol 16, Issue 2*, July 2006, PP. 21-45
  8. Arijit Sengupta, V Ramesh, Designing Document SQL: An Accessible yet Comprehensive Ad-Hoc Querying Frontend for Query, *JDM, Vol 20, Issue 4*, September 2009, PP. 26-53
  9. Anteneh Ayanso, Paulo B. Goes, A Cost-Based Range Estimation over Relational Databases, *JDM, Vol 20, Issue 3*, May 2009, PP. 3-15
  10. Yingjiu Li, Huiping Guo, A Multiple-Bits Watermark for Relational Data, *JDM, Vol 19, Issue 3*, October 2008, PP. 1-24
  11. Amel Mammar, Regine Laleau, UB2SQL: A Tool for Building Database Applications, *JDM, Vol 17, Issue 1*, September 2006
  12. Mathew N. Smith, Peter J.H. King, A Database Interface for Link Analysis, *Vol 16, Issue 1*, June 2005, PP. 60-74
  13. Nendad Jukic, Susan V Vrbsky, Allen Parrish, Enhancing Database Access Control by Facilitating Non-Key Related Cover Stories, *JDM, Vol 16, Issue 3*, PP. 1-20
  14. R Preethi, P. Sumathi Sumathi, Dynamic Analysis for Data Stream Classification, *IJDS, Vol 3, Issue 1*, March 2017

- 
15. Carl Stephen, Michael T. Vanecek, Data Management Issues in Information Systems, *JDM*, Vol 6 Issue 4, January 27, 1995 (pages 3-12)