

# *clientDocs*

## Client folder :

*In package.json add proxy (ipv4 or local host) with port of server (5000)*

```
    "extends": [  
      "react-app",  
      "react-app/jest"  
    ],  
  },  
  "proxy": "http://192.168.1.34:5000",  
  "browserslist": {  
    "production": [  
      ">0.2%",  
      "last 1 month",  
      "Firefox ESR"  
    ]  
  }  
}
```

> **npm start** in the client folder to start app in the browser.  
*navigate* to <http://localhost:3000> or <http://yourIPV4:3000> to start client

1. Login page : url : [http://your\\_ipv4:3000/login](http://your_ipv4:3000/login)  
\* use ipv4 or local host accordingly \*



The login form is centered on the page. It features a background collage of travel-related icons including the Eiffel Tower, Big Ben, the Statue of Liberty, a red car, a yellow taxi, a blue train, a globe, a smartphone with a map, a passport, and a suitcase. The word "Login" is written in a large, bold, black font across the top of the form area. Below it, there are two input fields: "Email" with the text "malikvedanshu757@outlook.com" and "Password" with a series of dots. A black "Submit" button is positioned below the password field. At the bottom of the form, there are two links: "New user ? [Register](#)" and "Forgot Password/Username ? [Click here](#)".

Login

Email malikvedanshu757@outlook.com

Password .....

Submit

New user ? [Register](#)

Forgot Password/Username ? [Click here](#)

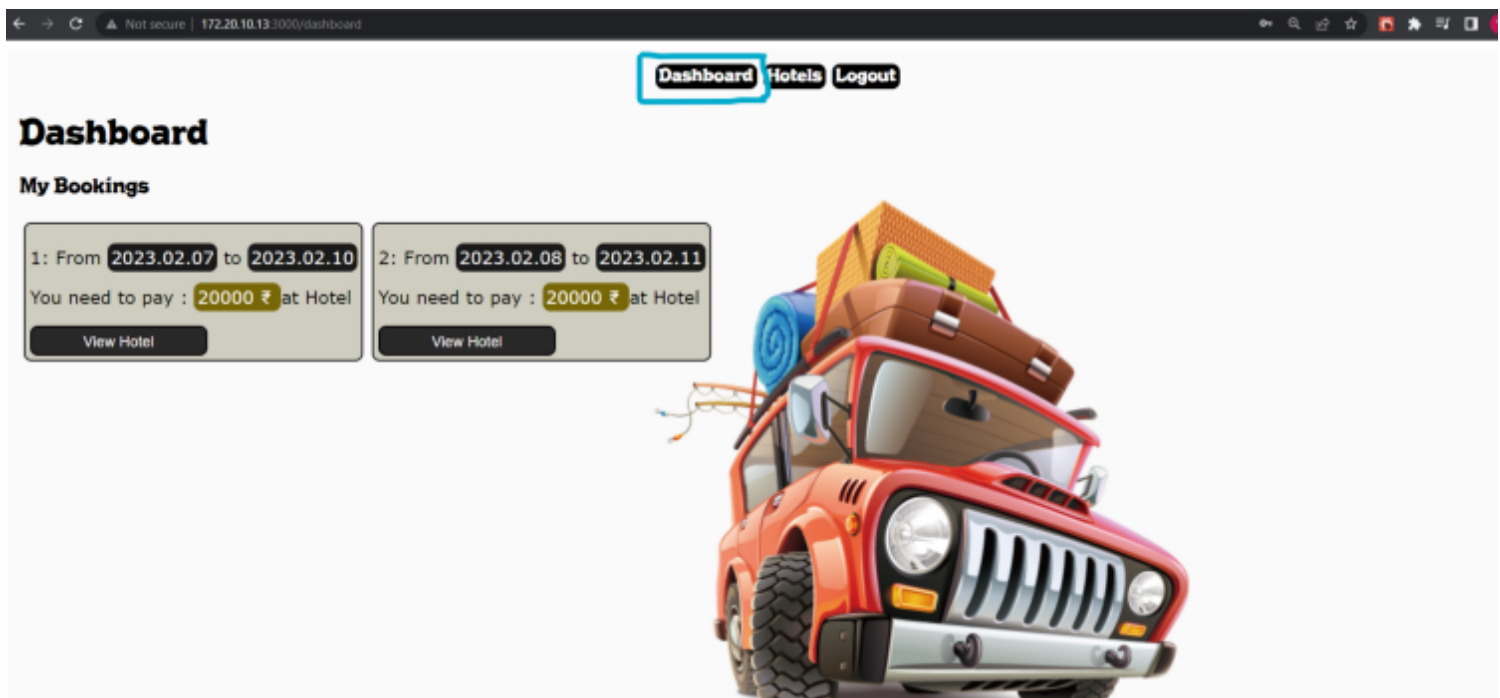
## 2. Dashboard:

URL : <http://localhost:3000/dashboard>

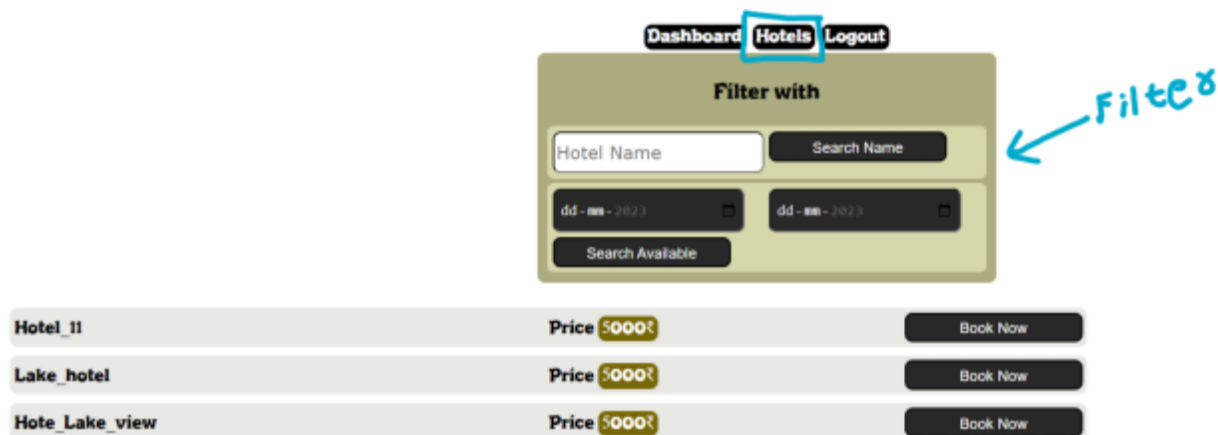
Dashboard shows all bookings so far.

Above which there are 3 buttons to navigate,

- Dashboard : to view all hotels
- Bookings : to view all hotels and book them



3. URL : `_:3000/hotels`  
Shows all hotels,
- Enter a string in name and click on Search Name to show all hotels containing that string
  - Select a checkin date and checkout date and click on Search Available will show all hotels that are available to book.
  - Book now button contains name with the hotel ids, to navigate to a particular hotel



4. Click on Book now will navigate you to : `http://_ipv4_/viewhotel/:hotelId`

*note: hotel id is the id in mongodb database collection :*  
all-hotels

172.20.10.13:3000/viewhotel/63e0c16fb2f2439c0631c817

[Dashboard](#) [Hotels](#) [Logout](#)

**Hotel 11**

**Price : 5000 Rupees**



[Book](#)

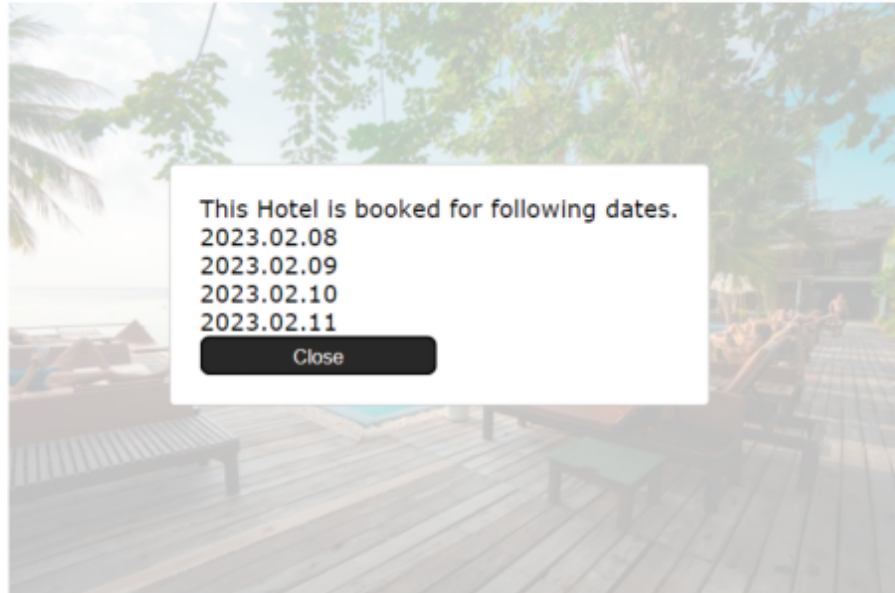
[View booking history](#)

5. Click “View booking” history button will open a modal which will show all the dates on which the hotel is already booked.

[Dashboard](#) [Hotels](#) [Logout](#)

## Hotel 11

Price : 5000 Rupees



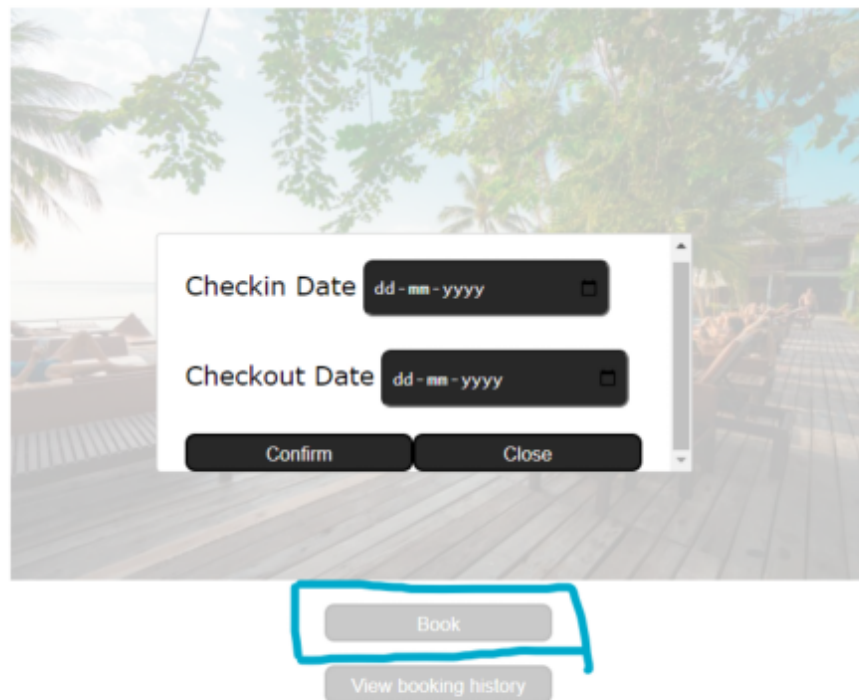
[Book](#)

[View booking history](#)

6. Click on the “book” will open a modal with the a option to select **checkin** and **checkout** dates
7. Click “**confirm**” to confirm Booking dates.
8. A info will show for 2 seconds if booking is successfully completed or not.

**Hotel 11**

**Price : 5000 Rupees**

A screenshot of a hotel booking interface. The background is a blurred image of a tropical beach with palm trees and a wooden deck. Overlaid on this is a white rectangular form with rounded corners. Inside the form, there are two date selection fields: 'Checkin Date' and 'Checkout Date', both with a placeholder 'dd-mm-yyyy' and a small calendar icon to the right. Below these fields are two buttons: 'Confirm' and 'Close'. Below the form, outside the white overlay, are two more buttons: 'Book' and 'View booking history'. The 'Book' button is highlighted with a blue hand-drawn rectangle.

Checkin Date dd-mm-yyyy

Checkout Date dd-mm-yyyy

Confirm Close

Book

View booking history

10. You can navigate to dashboard to see booking history.