Lab 2 Report

Name: Darshil Kapadia

013007280

Git-Repo:

https://github.com/Anuragis/CMPE273-36.git

HomeAway using NodeJS, ReactJS and Kafka

Introduction:

Goals of the system:

- The goal of this system is to widen the understanding of ReactJS and NodeJS. Along with that, its goal is to help understand the state management in react using Redux and to develop enterprise-wide applications which takes care of various operations like managing the information of users, managing the information of owners, managing the information of properties listed by different owners and also taking care of booking done by different travelers.
- In order to develop this application, we will get a wide understanding of different concepts of ReactJS like props, states, Redirect features, form validations and many more.
- State Management, quite important while working with ReactJS, will be done using the help of Redux and in turn learning those Redux's concept of Store, Actions and Reducers.
- This application will also help us in widening the backend i.e NodeJS concepts because of different operations like communication with MongoDB, managing session, encrypting passwords using bcrypt, protecting routes using JWT and Passport, uploading images and many more.
- This application will also help us in getting hands on experience on one of the most useful and emerging technology called Kafka. Kafka basically combines 2 features called "Publish Subscribe and Messaging Queues". We will learn basic communication over here using Kafka Services.

Purpose of the system:

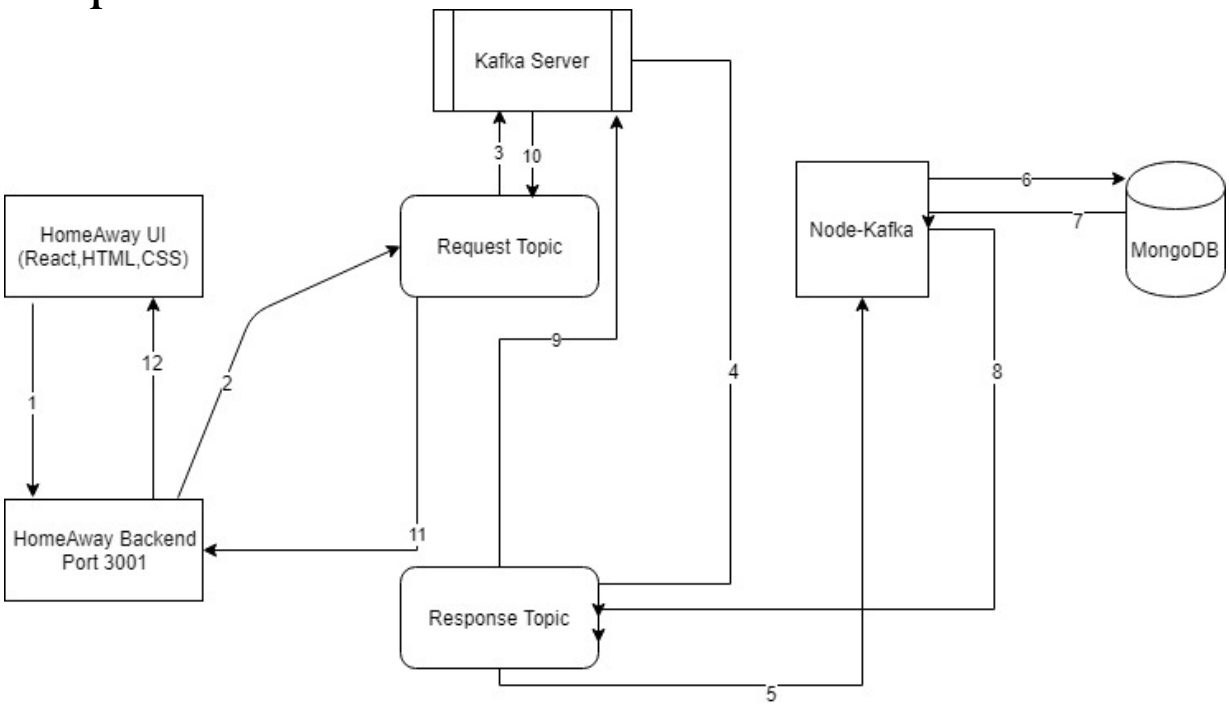
- The purpose of the system is to connect different property listers and the travelers who will access these listed properties.
- The system will allow different owners of properties around the globe to rent their properties on the system and in turn help them to earn through it.
- For travelers, this system will allow them to search for different properties all around the world and booking the one they want to rent.
- This system will take care of booking and listing of properties with proper session management.

System Design:

- The system is a HomeAway application used to achieve the goals of the system mentioned above. For UI, ReactJS and Redux is used and NodeJS handles the server and MongoDB serves as a Database.

- The request received from the client is passed to kafka's request topic. It then gets transferred to kafka server where proper handling of that request is done. Kafka, here, works with mongoDB to fetch and search through data and performs a required operation to send a desired response to the client.
- The response again is sent through one of the topic through which it is fetched and sent to the client.

Complete Flow:



Backend Routes

Routes	Method	Functionality
Index Routes		
/travellogin	POST	- Sends the parameters like email and password and the route authenticates the user. If the credentials are correct, the user is logged in.
/travelsignup	POST	- Sign's up new user and send an acknowledgement on successful creation.
/ownerlogin	POST	- Sends the parameters like email and password and the route authenticates the owner. If the credentials are correct, the owner is logged in.

/ownersignup	POST	- Sign's up the new owner and send an acknowledgement on successful creation.
	Travel Routes	
/:travelid	GET	- Get the details of the travel user having id :travelid
/:travelid	PUT	- Update the details of the travel user having id :travelid
/:travelid/editpassword	PUT	- Edits the password of the user having id :travelid
/:travelid/bookingdetails	GET	- Gets the booking details that the user having :travelid has done.
/:travelid/upload	POST	- Updates the profile pic of the traveler having id :travelid.
/:travelid/question	GET	- Shows all the questions asked by the traveler having id :travelid under their Inbox.
/:travelid/question	POST	- Allows traveler to post questions on properties listed by owners.
	Owner Routes	
/:ownerid	GET	- Get the details of the owner user having id :ownerid
/:ownerid	PUT	- Updates the detail of the owner having id :ownerid.
/:ownerid/editpassword	PUT	- Edits the password of the user having id :ownerid
/:ownerid/property	GET	- Gives all the properties listed by the owner having id :ownerid
/:ownerid/dashboard	GET	- Gives the details about who has booked the properties listed by the owner having id :ownerid
/:ownerid/question	GET	- Shows all the questions asked on the properties listed by the owner having id :ownerid.
/:travelid/question	POST	- Allows owner to post answers on questions asked by travellers.
	Property Routes	
/	POST	- Adds a new property

/:propertyid	GET	- Gets the details of the property having id :propertyid
/:propertid	PUT	- Updates the details of the property.
/search	POST	- Searches for properties with filters provided in the body.
/:propertyid/book	POST	- Books the property having id :propertyid
/:propertyid/upload	POST	- Uploads photos for the property having id :propertyid.

Test Cases and Results

We'll consider different cases and will visit them one by one. The use cases that I'll be discussing are as follows:

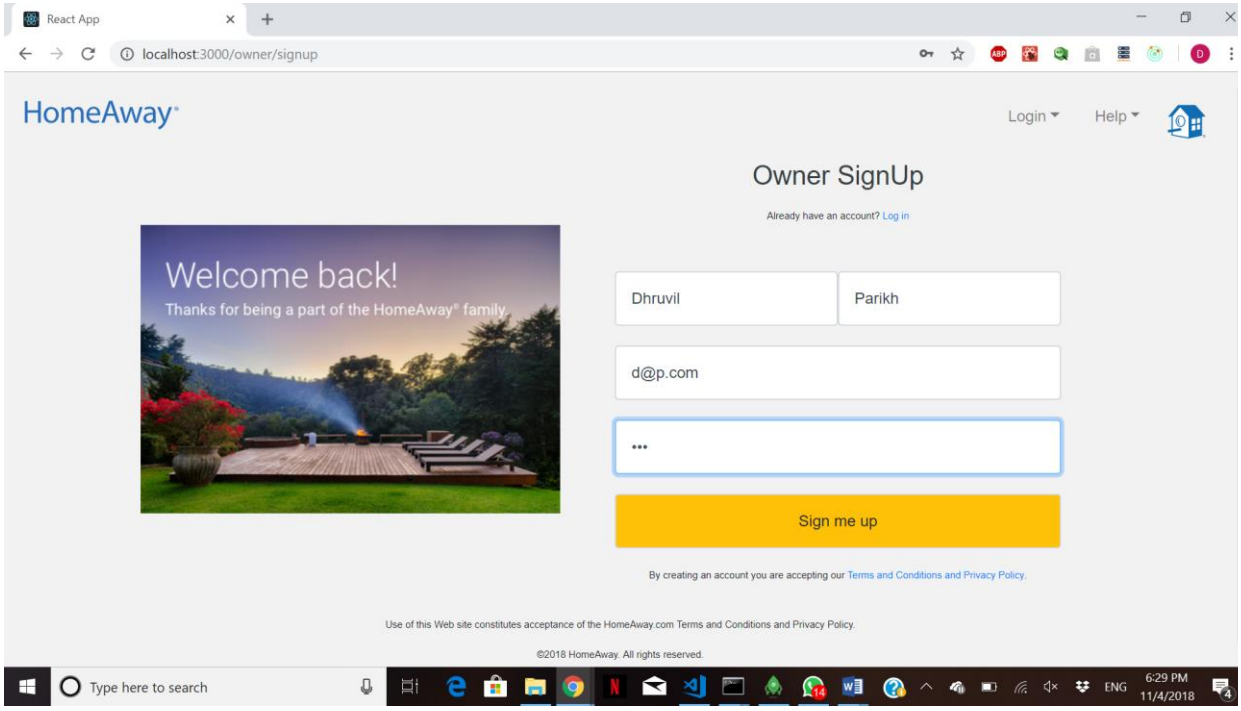
1) **Creating a new owner account.**
2) **Logging in with the newly created owner account.**
3) **Listing a new property from this owner account.**
4) **Displaying the properties listed by the owner.**
5) **Editing property details.**
6) **Editing the owner's detail like billing address, contact no and other things.**
7) **Logging off from the owner account.**
8) **Creating a new travel account.**
9) **Logging in with the newly created travel account.**
10) **Editing the basic details of travel account.**
11) **Searching a property based on "city, arrival, departure and # of guests".**
12) **Displaying the result of the above search and filtering through it.**
13) **Asking Owner the question on any of the property searched.**
14) **Selecting a property from search result and booking it with the travel account.**
15) **Looking at the recent trips that I have booked and filtering through it.**
16) **Logging off from the travel account.**
17) **Answering to the question asked by the traveler.**
18) **Looking at Owner's Dashboard.**

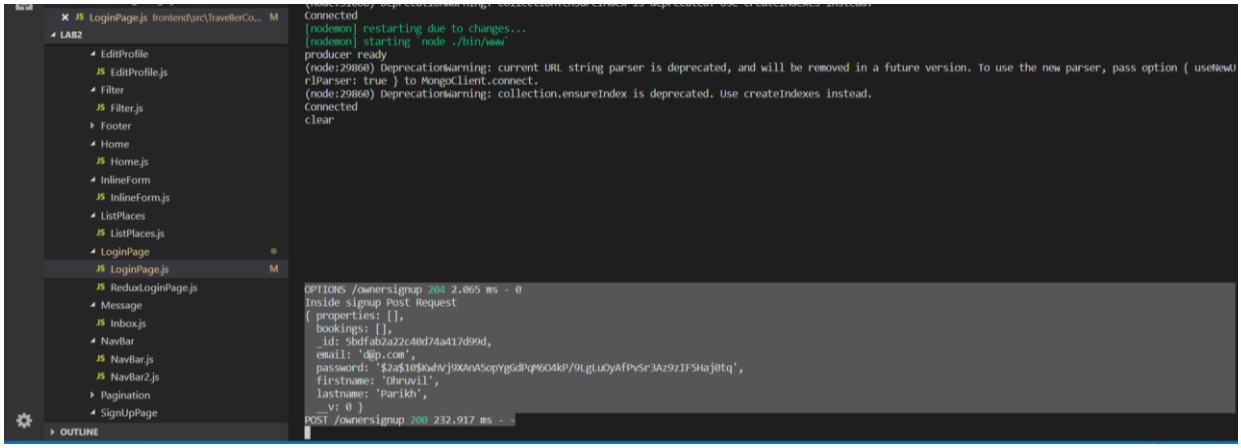
We'll also consider different use cases which will have invalid inputs in form like arrival date is after the departure date and many more. Different use cases that we'll consider for validations are as follows:

1) **Keeping the email/password field empty**
2) **Logging in with incorrect credentials**
3) **Searching the property without location city or arrival date or departure date or guests.**
4) **Editing the information and keeping required fields like name, contact no blank.**

Test Cases:

Creating a new owner account

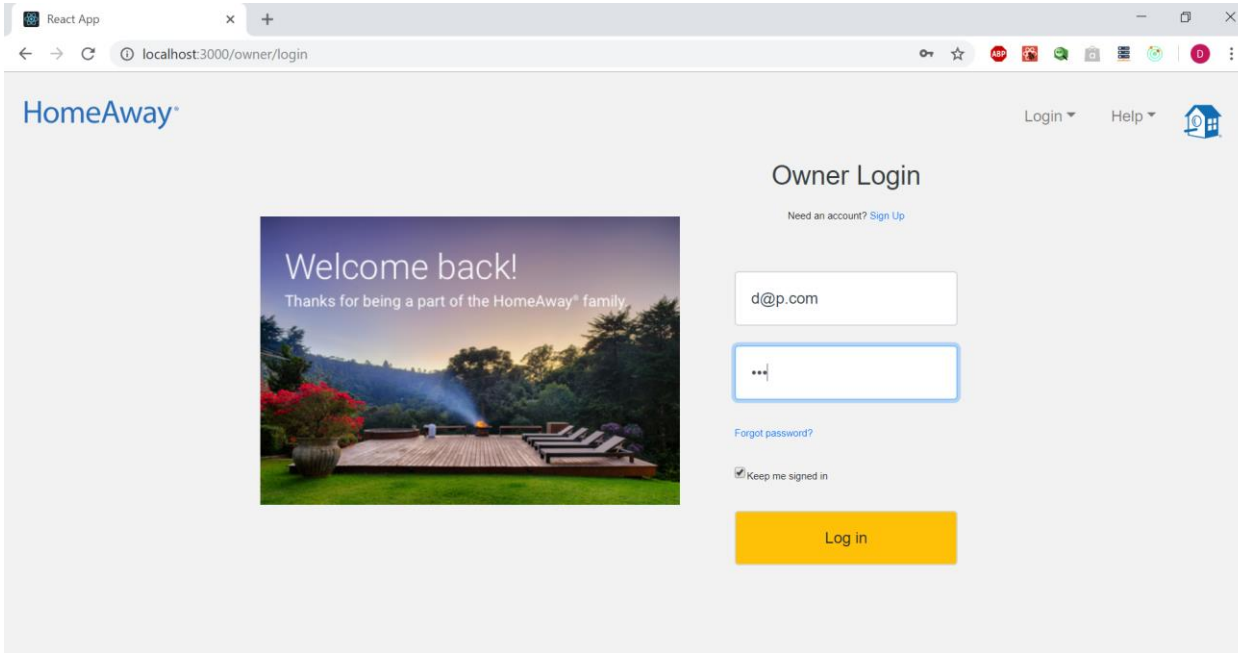


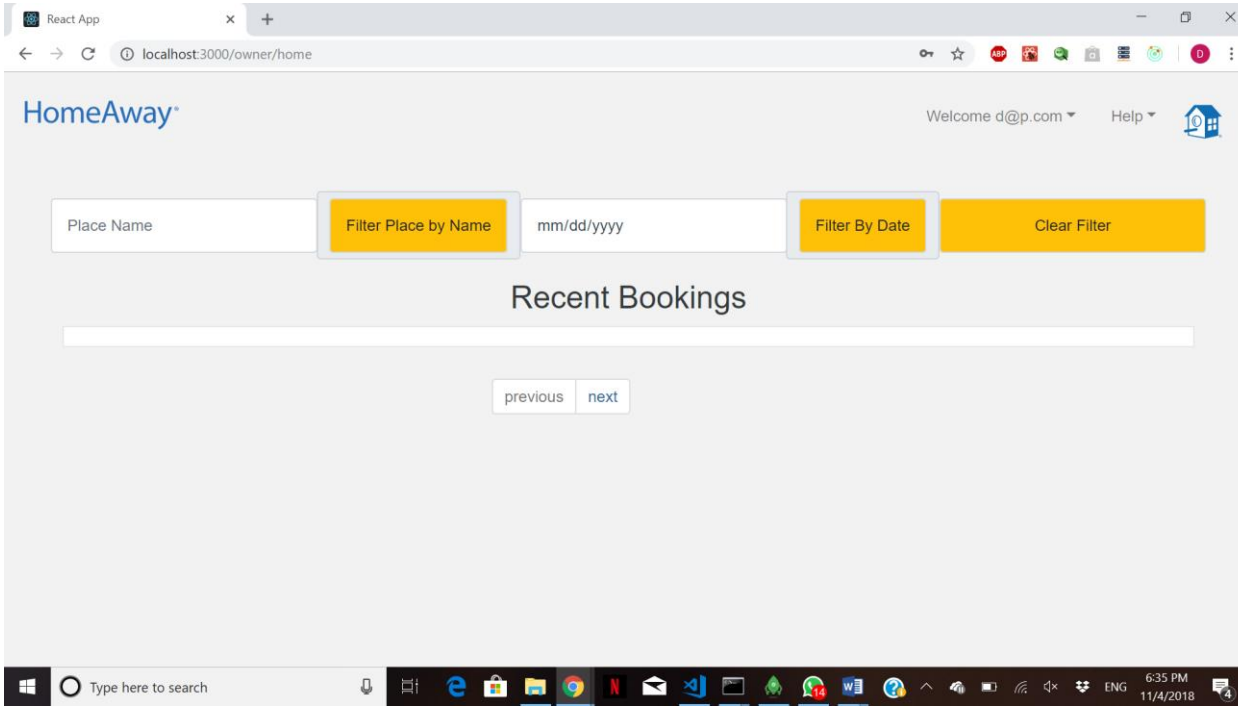


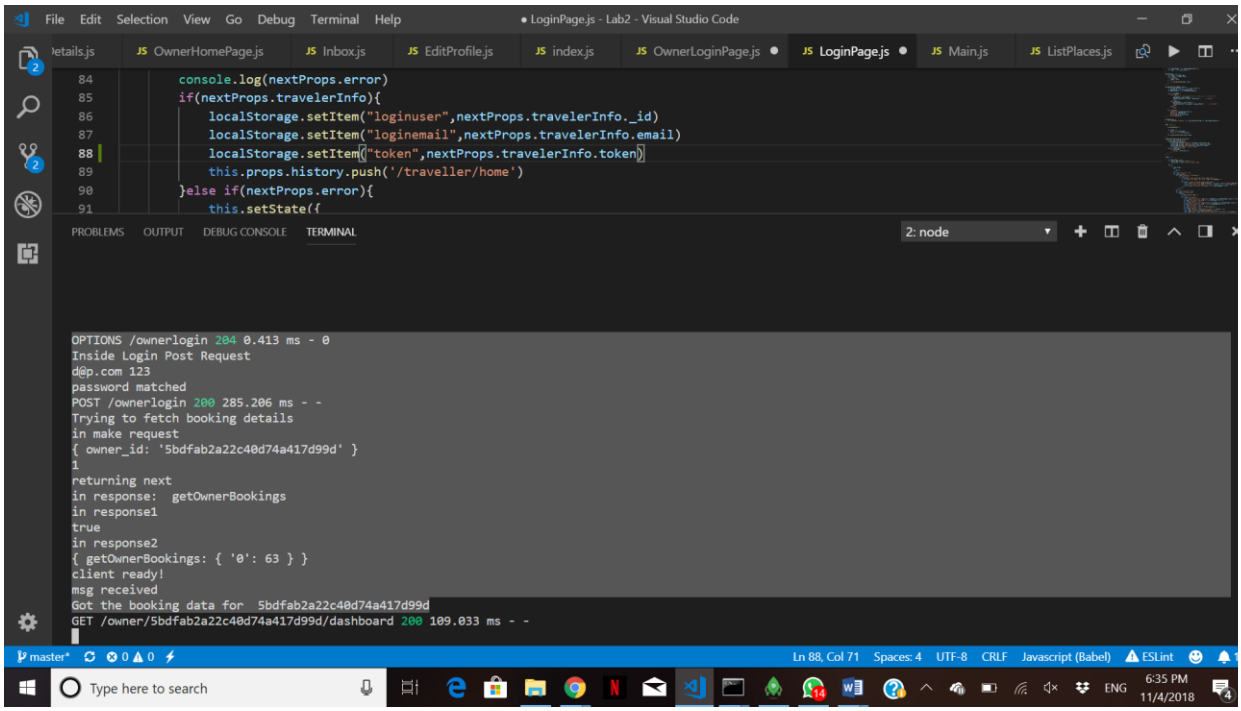
Explanation:

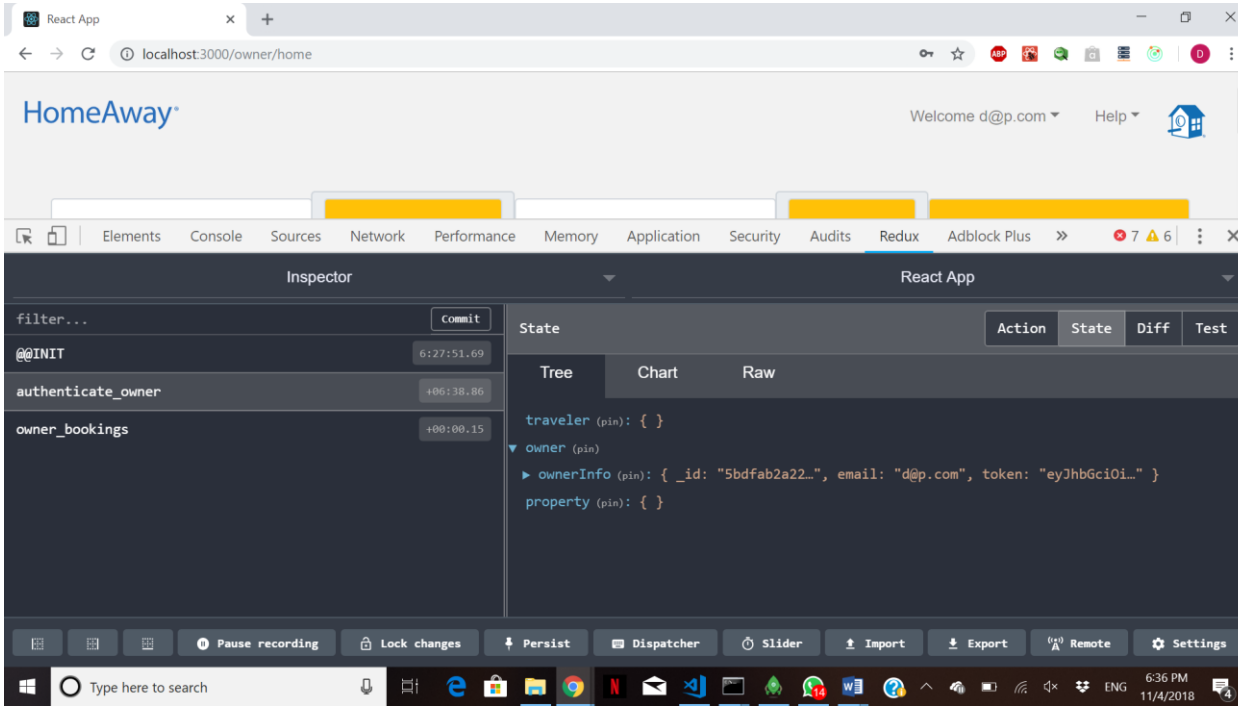
- A new owner will be created named “Dhruvil Parikh” and login credentials of d@p.com and the password encrypted with bcrypt.
- As one can see in the output of the terminal, the post request has been successfully completed and the password too has been stored after being encrypted with bcrypt and the salt cycle running for 10 rounds.

Logging in with the newly created owner account







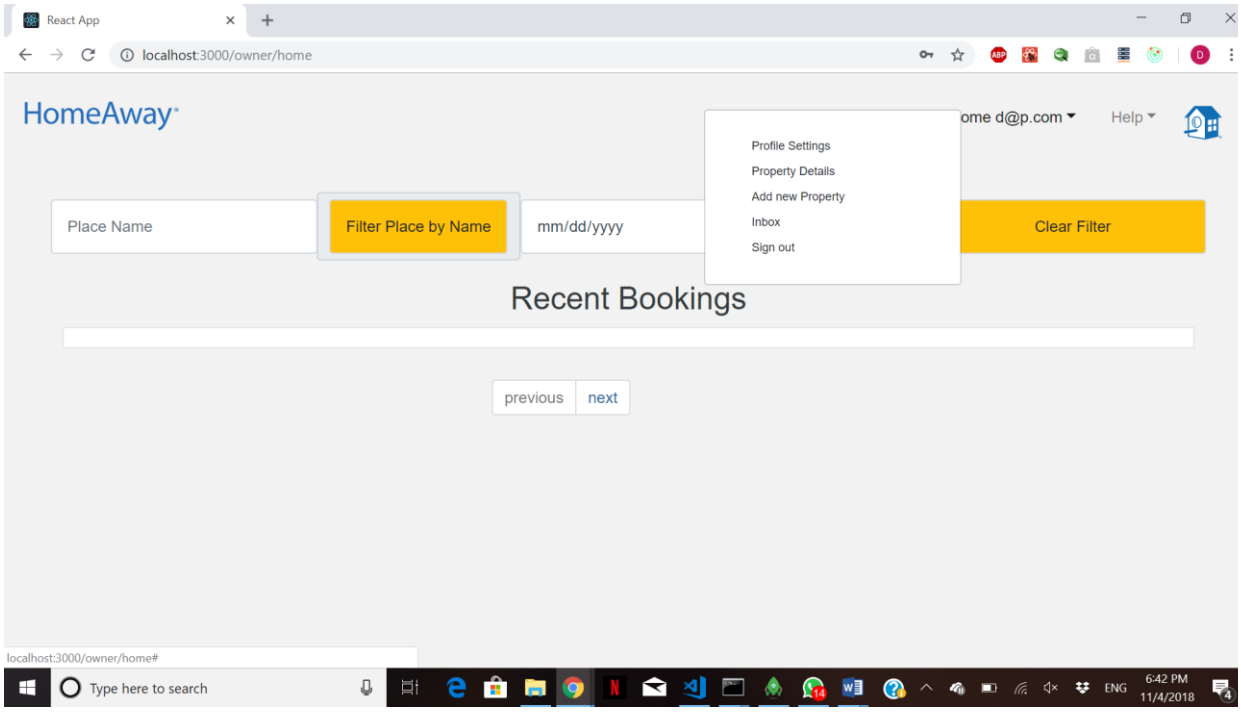


Explanation:

- The owner logged in using his login credentials.
- Notice the navigation bar, when the owner has not logged in and thus getting the option of "Login" only at the top.
- Please note the change once we login, the navigation bar now has welcome d@p.com on top.

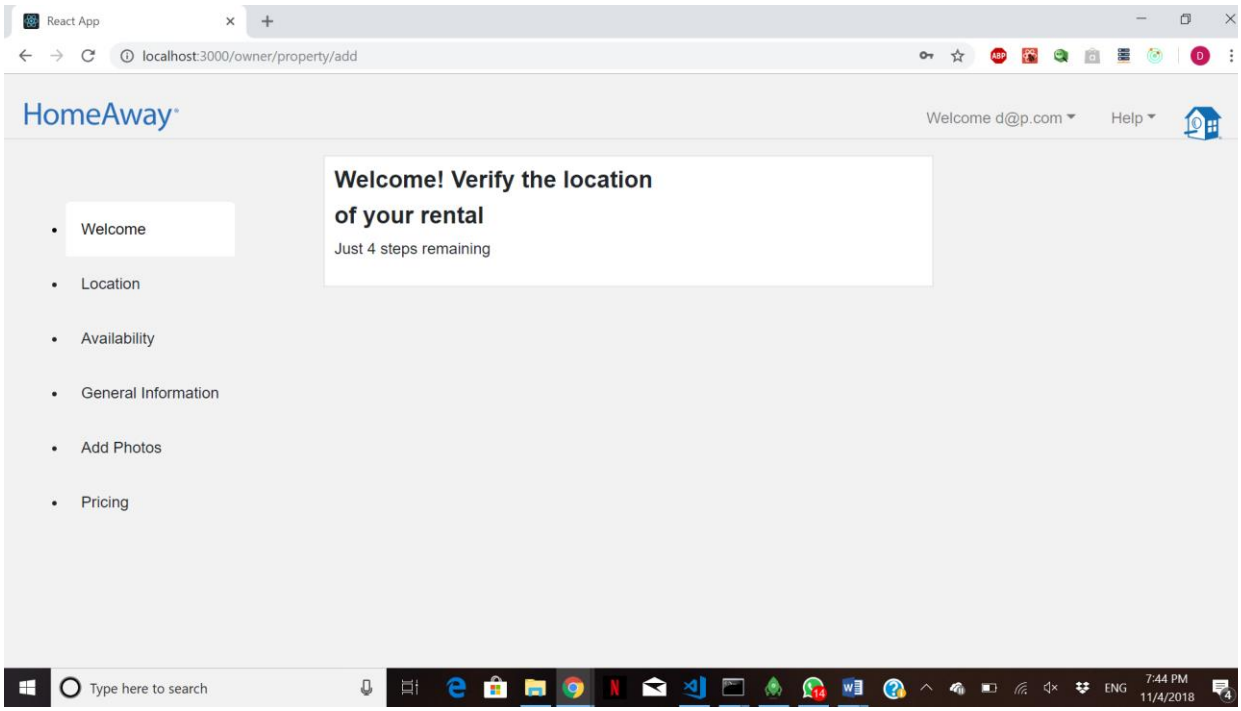
- After logging in, the redux store updates itself with owner information.

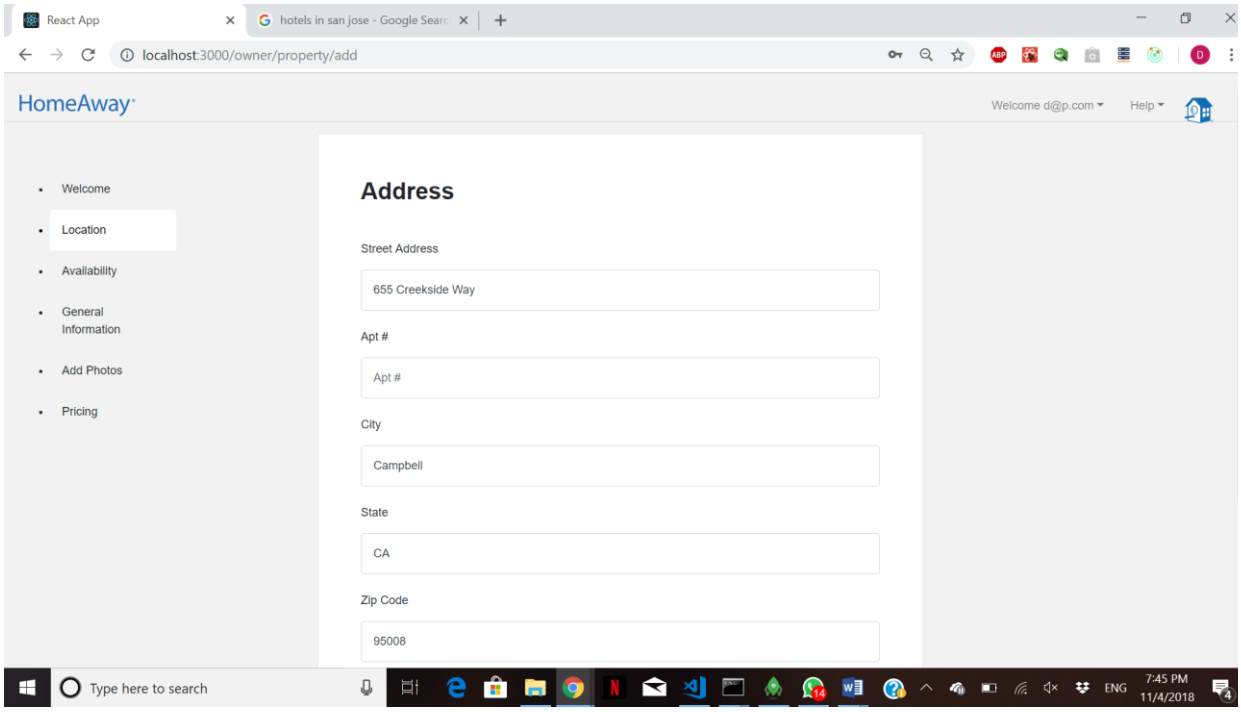
Listing a new property from this owner account

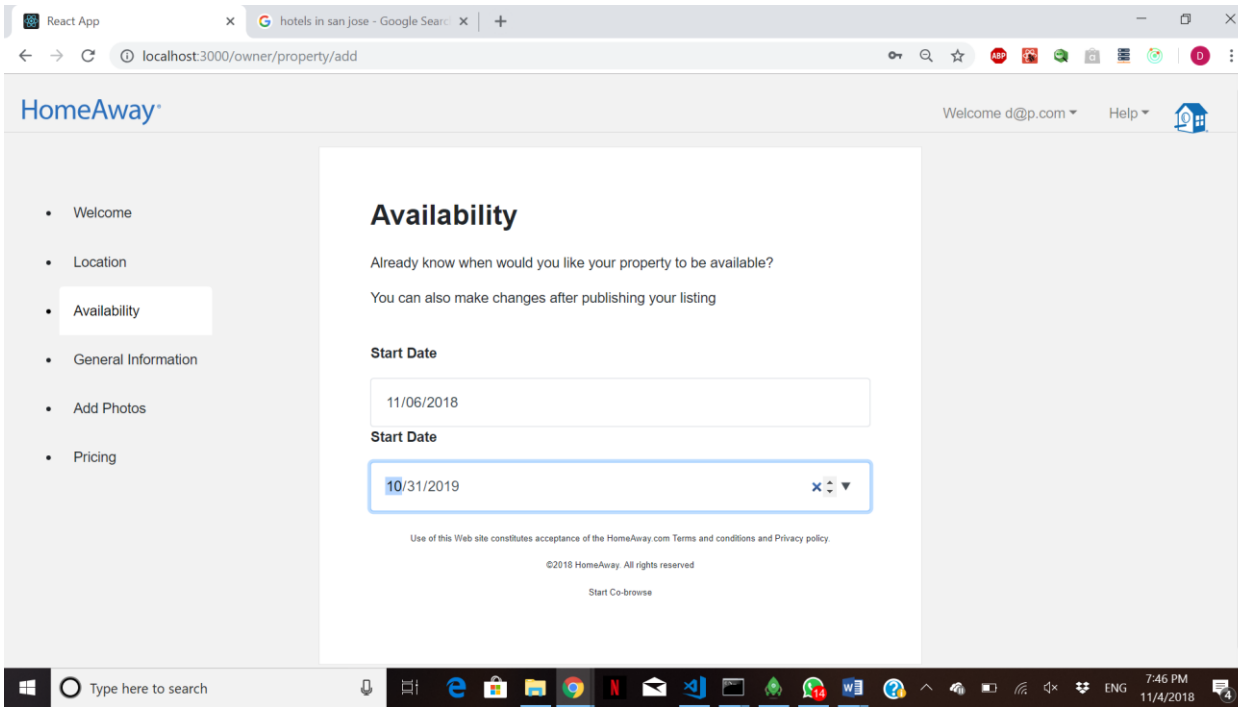


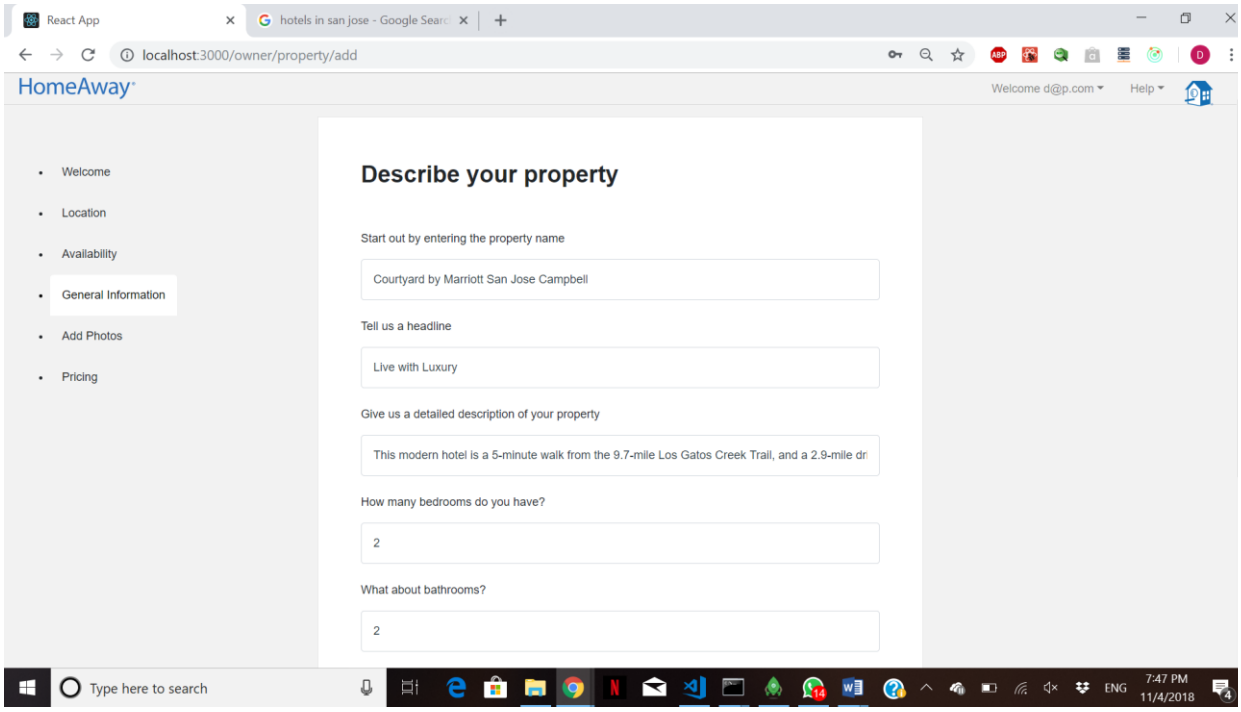
Explanation:

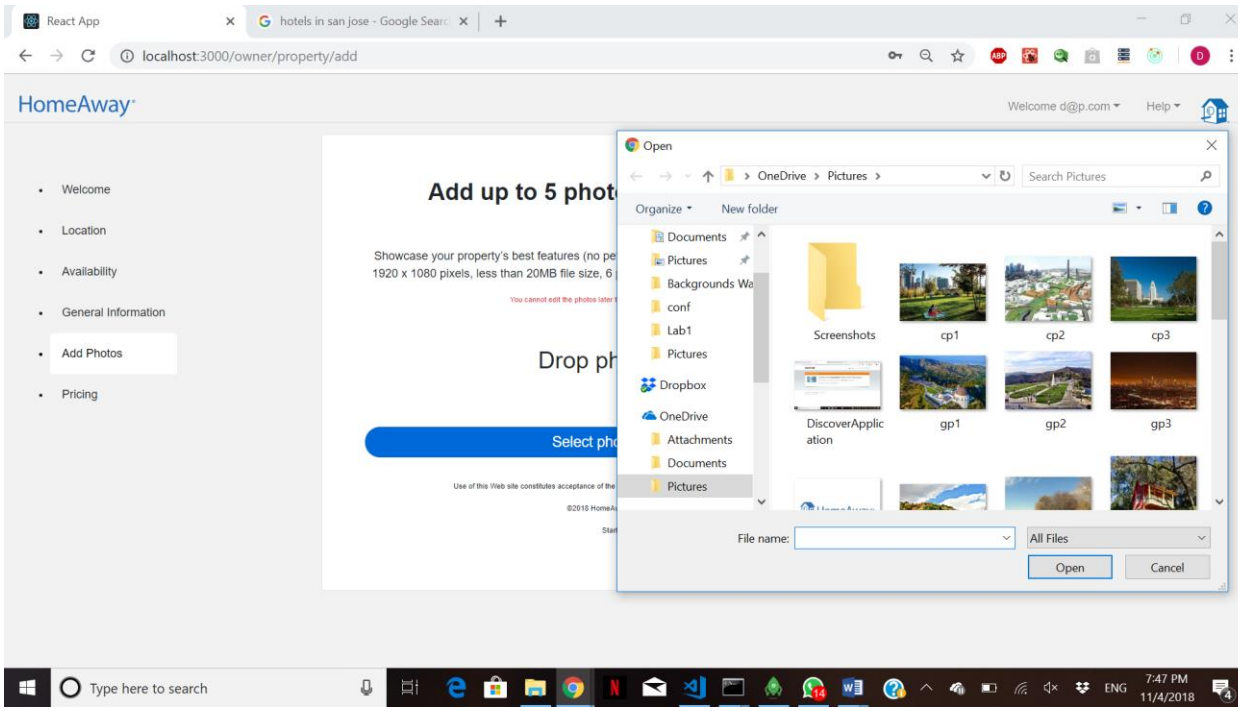
- This is the home screen of the owner.
- As you can see on the navigation bar, the "login" option has changed and the owner now gets access to options like "Add new property, profile settings and many more."
- This screen usually shows the bookings that the travelers have done on the listings that that owner has listed for the rent.
- The screen also allows user to filter through the bookings done on their properties.
- We will be able to see the booking over here after we will create a new traveler who will book this owner's properties.

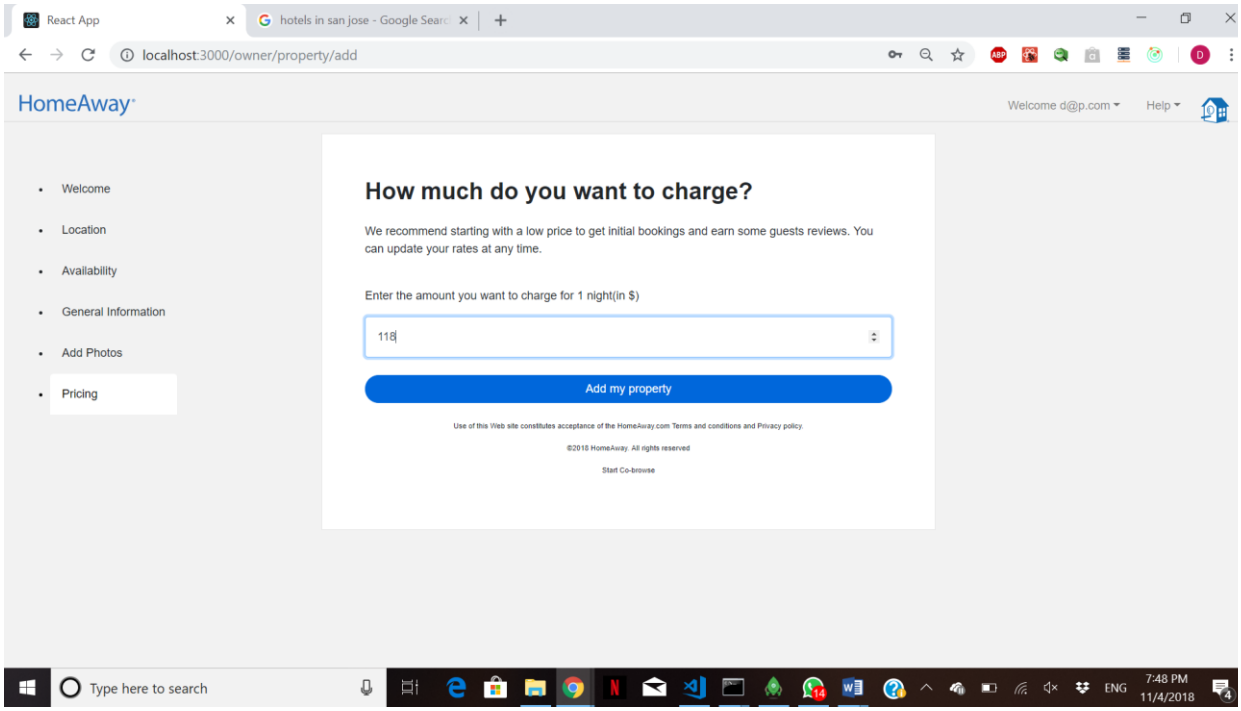

React App
localhost:3000/owner/property/add
HomeAway
Welcome d@p.com
Help
Welcome! Verify the location of your rental
Just 4 steps remaining
Welcome
Location
Availability
General Information
Add Photos
Pricing
Type here to search
ENG
7:44 PM
11/4/2018

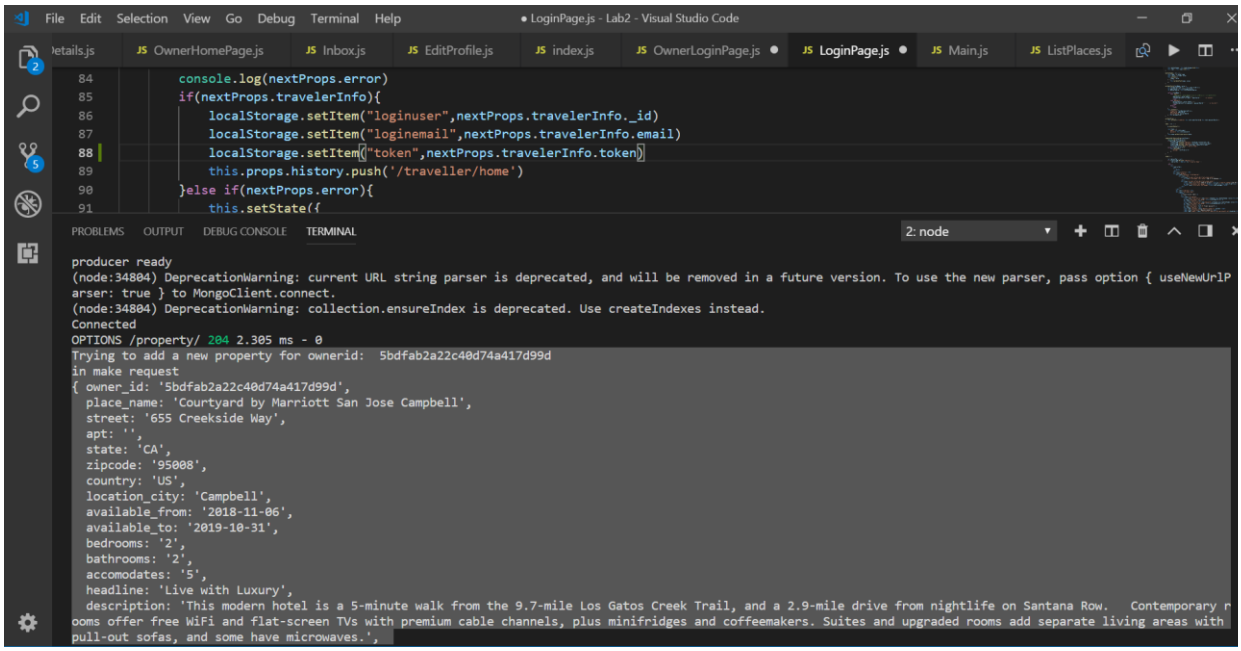

React App
hotels in san jose - Google Searc
localhost:3000/owner/property/add
HomeAway
Welcome d@p.com
Help
Welcome
Location
Availability
General Information
Add Photos
Pricing
Address
Street Address
655 Creekside Way
Apt #
Apt #
City
Campbell
State
CA
Zip Code
95008
Type here to search
ENG
7:45 PM
11/4/2018

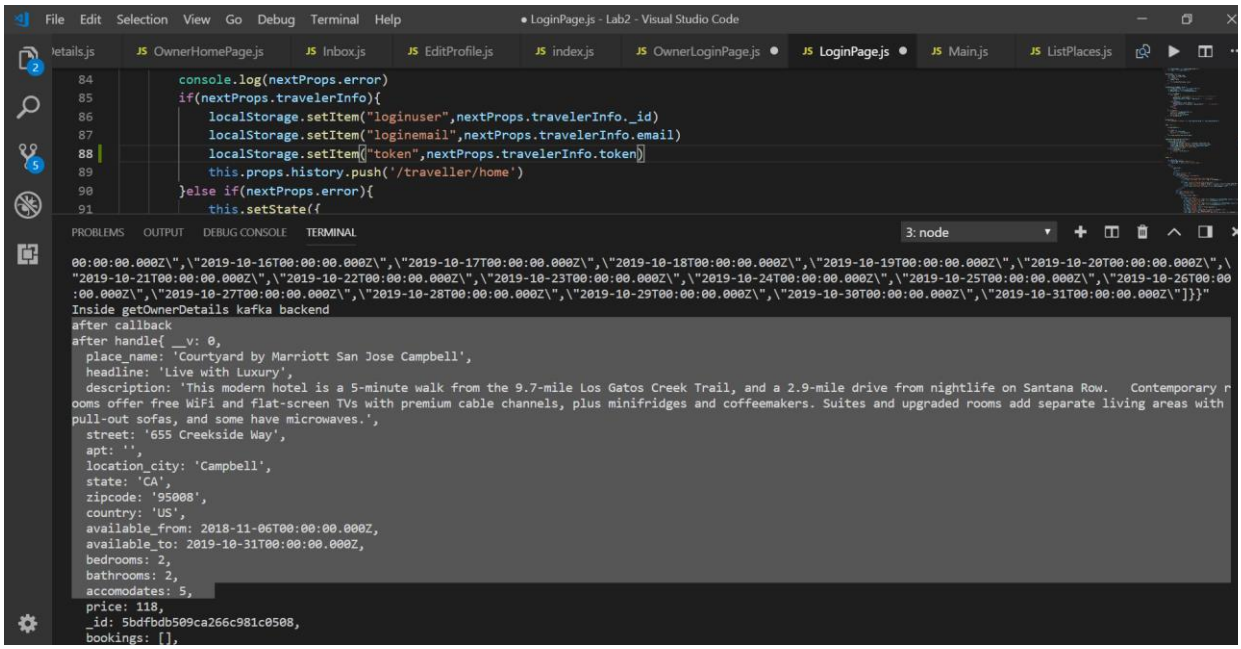

React App
hotels in san jose - Google Searc
localhost:3000/owner/property/add
HomeAway
Welcome d@p.com
Help
Welcome
Location
Availability
General Information
Add Photos
Pricing
Availability
Already know when would you like your property to be available?
You can also make changes after publishing your listing
Start Date
11/06/2018
Start Date
10/31/2019
Use of this Web site constitutes acceptance of the HomeAway.com Terms and conditions and Privacy policy.
©2018 HomeAway. All rights reserved
Start Co-browse
Type here to search
ENG
7:46 PM
11/4/2018

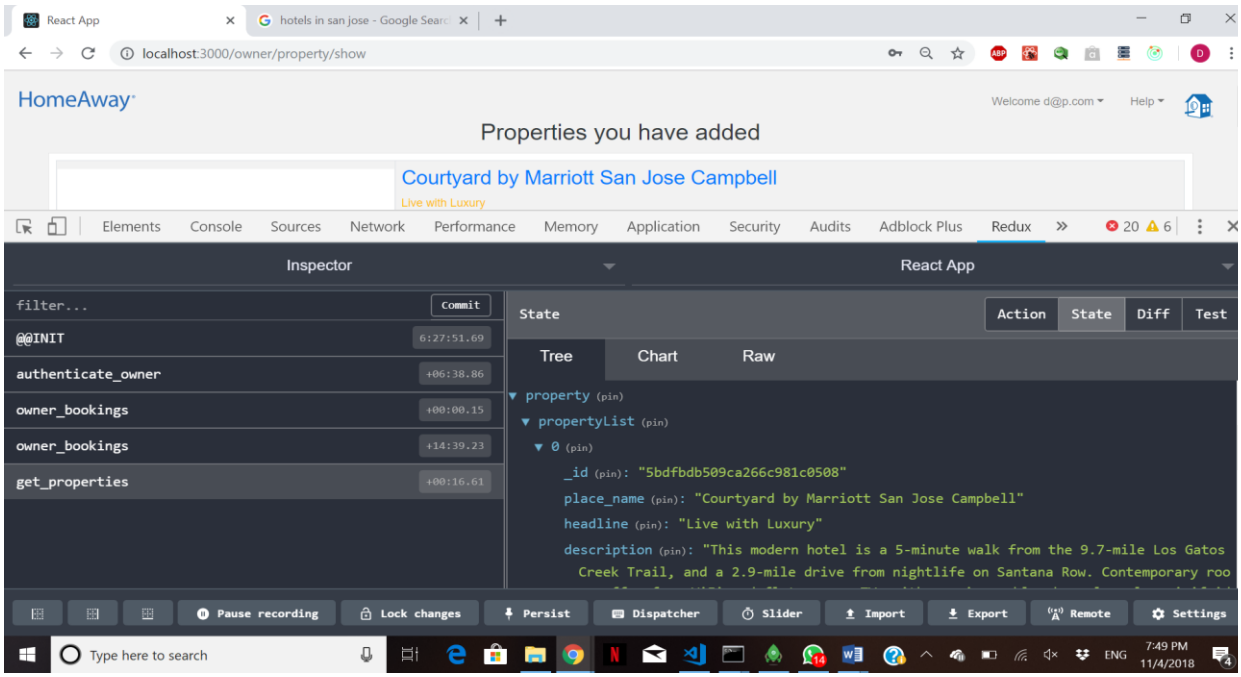

React App
hotels in san jose - Google Searc
localhost:3000/owner/property/add
HomeAway
Welcome d@p.com
Help
Welcome
Location
Availability
General Information
Add Photos
Pricing
Describe your property
Start out by entering the property name
Courtyard by Marriott San Jose Campbell
Tell us a headline
Live with Luxury
Give us a detailed description of your property
This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos Creek Trail, and a 2.9-mile dr
How many bedrooms do you have?
2
What about bathrooms?
2
Type here to search
ENG
7:47 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/owner/property/add
HomeAway
Welcome d@p.com
Help
Welcome
Location
Availability
General Information
Add Photos
Pricing
Add up to 5 phot
Showcase your property's best features (no pe
1920 x 1080 pixels, less than 20MB file size, 6
You cannot edit the photos later
Drop ph
Select pho
Use of this Web site constitutes acceptance of the
©2018 HomeA
Start
Open
OneDrive › Pictures ›
Search Pictures
Organize
New folder
Documents
Pictures
Backgrounds Wa
conf
Lab1
Pictures
Dropbox
OneDrive
Attachments
Documents
Pictures
Screenshots
cp1
cp2
cp3
DiscoverApplic
ation
gp1
gp2
gp3
File name:
All Files
Open
Cancel
Type here to search
ENG
7:47 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/owner/property/add
HomeAway
Welcome d@p.com
Help
Welcome
Location
Availability
General Information
Add Photos
Pricing
How much do you want to charge?
We recommend starting with a low price to get initial bookings and earn some guests reviews. You can update your rates at any time.
Enter the amount you want to charge for 1 night(in $)
118
Add my property
Use of this Web site constitutes acceptance of the HomeAway.com Terms and conditions and Privacy policy.
©2018 HomeAway. All rights reserved
Start Co-browse
Type here to search
ENG
7:48 PM
11/4/2018


File Edit Selection View Go Debug Terminal Help
• LoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
console.log(nextProps.error)
if(nextProps.travelerInfo){
localStorage.setItem("loginuser",nextProps.travelerInfo._id)
localStorage.setItem("loginemail",nextProps.travelerInfo.email)
localStorage.setItem("token",nextProps.travelerInfo.token)
this.props.history.push('/traveller/home')
}else if(nextProps.error){
this.setState({
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
producer ready
(node:34804) DeprecationWarning: current URL string parser is deprecated, and will be removed in a future version. To use the new parser, pass option { useNewUrlP
arser: true } to MongoClient.connect.
(node:34804) DeprecationWarning: collection.ensureIndex is deprecated. Use createIndexes instead.
Connected
OPTIONS /property/ 204 2.305 ms - 0
Trying to add a new property for ownerid: 5bdfab2a22c40d74a417d99d
in make request
{ owner_id: '5bdfab2a22c40d74a417d99d',
place_name: 'Courtyard by Marriott San Jose Campbell',
street: '655 Creekside Way',
apt: '',
state: 'CA',
zipcode: '95008',
country: 'US',
location_city: 'Campbell',
available_from: '2018-11-06',
available_to: '2019-10-31',
bedrooms: '2',
bathrooms: '2',
accomodates: '5',
headline: 'Live with Luxury',
description: 'This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos Creek Trail, and a 2.9-mile drive from nightlife on Santana Row. Contemporary r
ooms offer free WiFi and flat-screen TVs with premium cable channels, plus minifridges and coffeemakers. Suites and upgraded rooms add separate living areas with
pull-out sofas, and some have microwaves.',

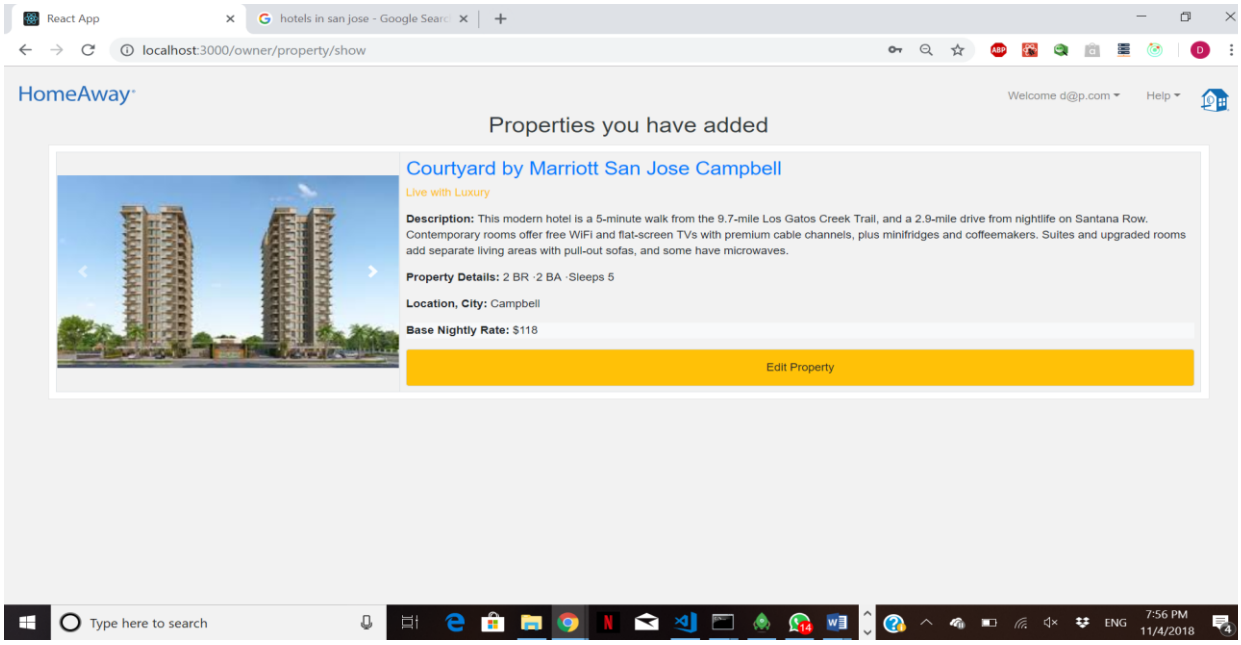

File Edit Selection View Go Debug Terminal Help
• LoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
console.log(nextProps.error)
if(nextProps.travelerInfo){
localStorage.setItem("loginuser",nextProps.travelerInfo._id)
localStorage.setItem("loginemail",nextProps.travelerInfo.email)
localStorage.setItem("token",nextProps.travelerInfo.token)
this.props.history.push('/traveller/home')
}else if(nextProps.error){
this.setState({
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
3: node
00:00:00.000Z\",\"2019-10-16T00:00:00.000Z\",\"2019-10-17T00:00:00.000Z\",\"2019-10-18T00:00:00.000Z\",\"2019-10-19T00:00:00.000Z\",\"2019-10-20T00:00:00.000Z\",\
"2019-10-21T00:00:00.000Z\",\"2019-10-22T00:00:00.000Z\",\"2019-10-23T00:00:00.000Z\",\"2019-10-24T00:00:00.000Z\",\"2019-10-25T00:00:00.000Z\",\"2019-10-26T00:00
:00.000Z\",\"2019-10-27T00:00:00.000Z\",\"2019-10-28T00:00:00.000Z\",\"2019-10-29T00:00:00.000Z\",\"2019-10-30T00:00:00.000Z\",\"2019-10-31T00:00:00.000Z\"]}}"
Inside getOwnerDetails kafka backend
after callback
after handle{ __v: 0,
place_name: 'Courtyard by Marriott San Jose Campbell',
headline: 'Live with Luxury',
description: 'This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos Creek Trail, and a 2.9-mile drive from nightlife on Santana Row. Contemporary r
ooms offer free WiFi and flat-screen TVs with premium cable channels, plus minifridges and coffeemakers. Suites and upgraded rooms add separate living areas with
pull-out sofas, and some have microwaves.',
street: '655 Creekside Way',
apt: '',
location_city: 'Campbell',
state: 'CA',
zipcode: '95008',
country: 'US',
available_from: 2018-11-06T00:00:00.000Z,
available_to: 2019-10-31T00:00:00.000Z,
bedrooms: 2,
bathrooms: 2,
accomodates: 5,
price: 118,
_id: 5bdfbdb509ca266c981c0508,
bookings: [],

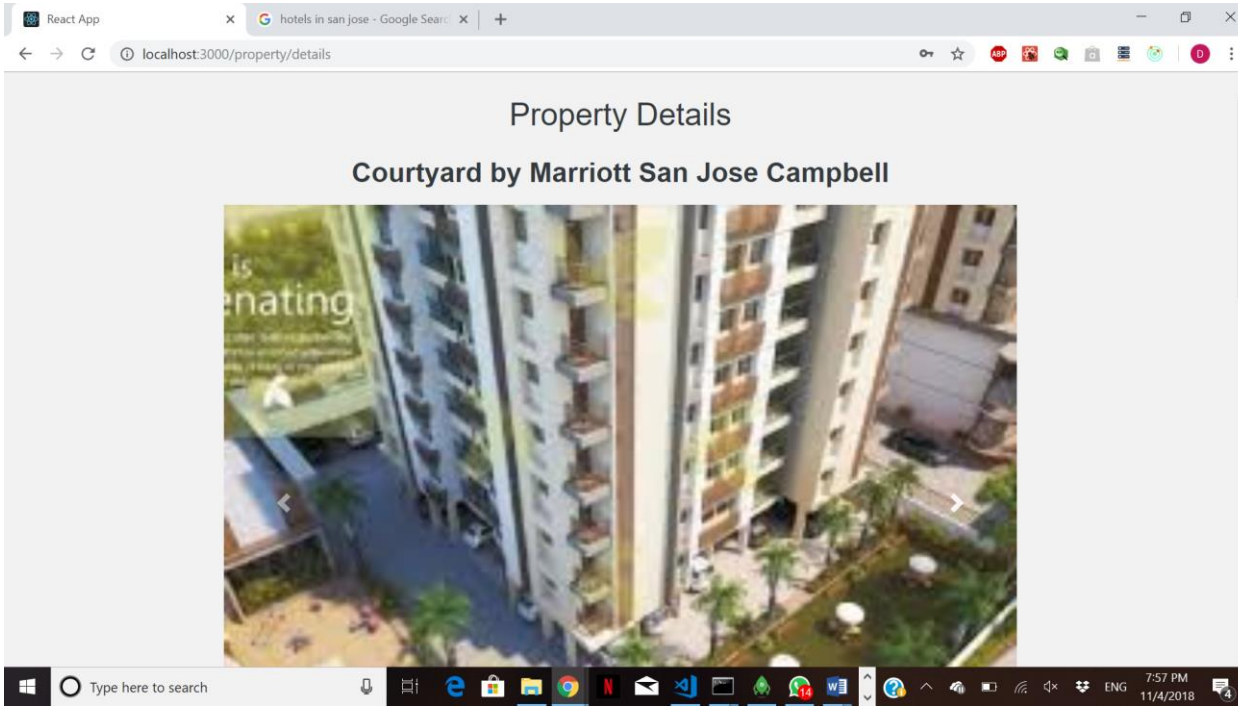


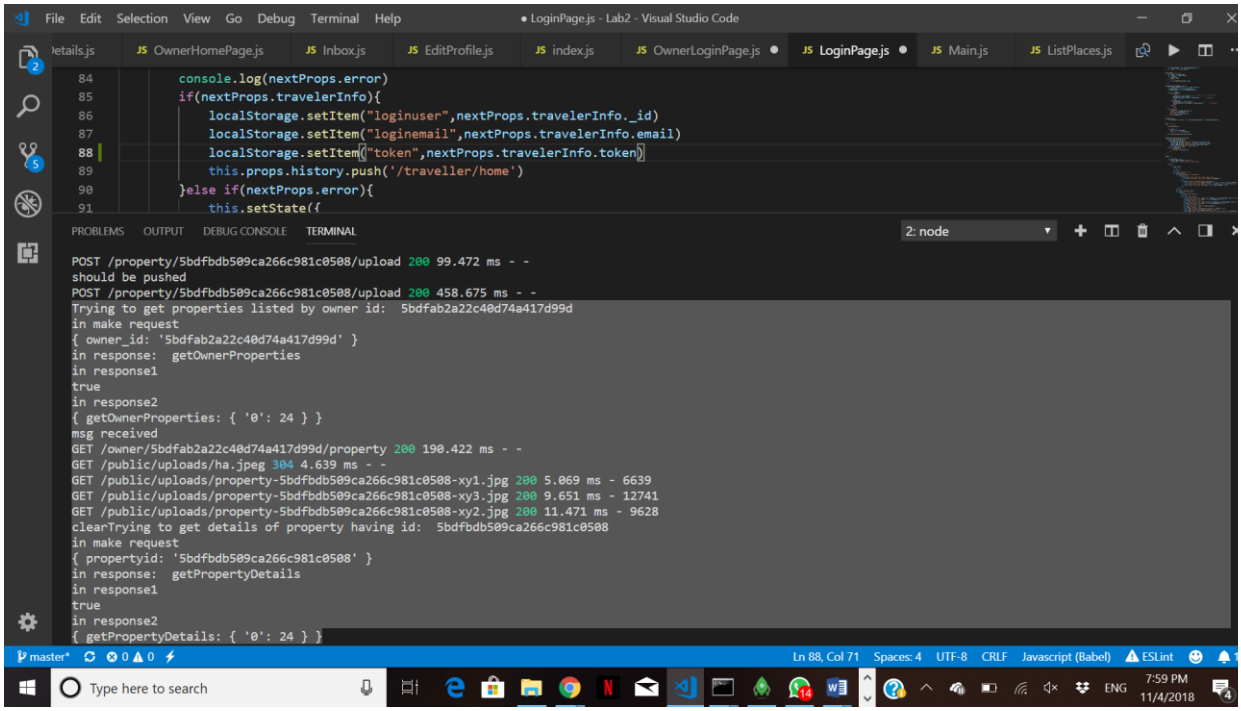
Explanation:

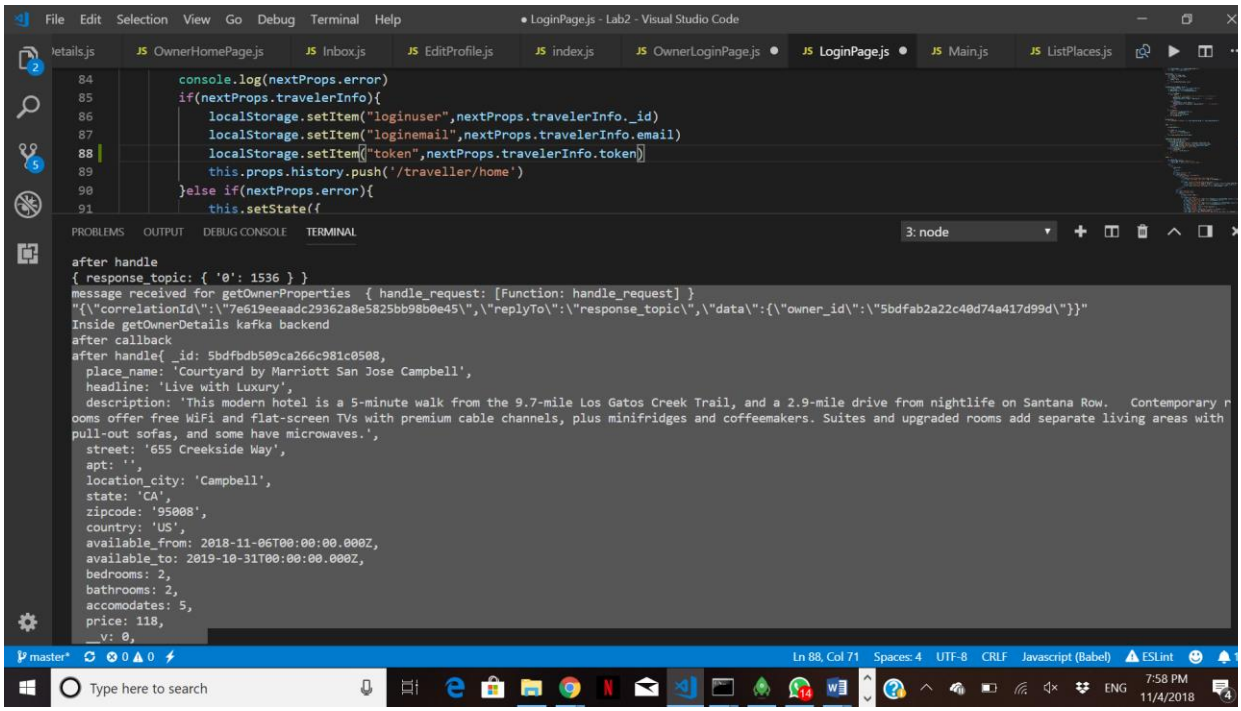
- These screenshots showcases the steps of how owner can add a new property listing.
- The property is located in "Campbell" city.
- The redux store shows that the store has now updated and it has stored the newest property added by the owner.

Displaying the properties listed by the owner




React App
hotels in san jose - Google Searc
localhost:3000/property/details
Property Details
Courtyard by Marriott San Jose Campbell
Type here to search
ENG
7:57 PM
11/4/2018

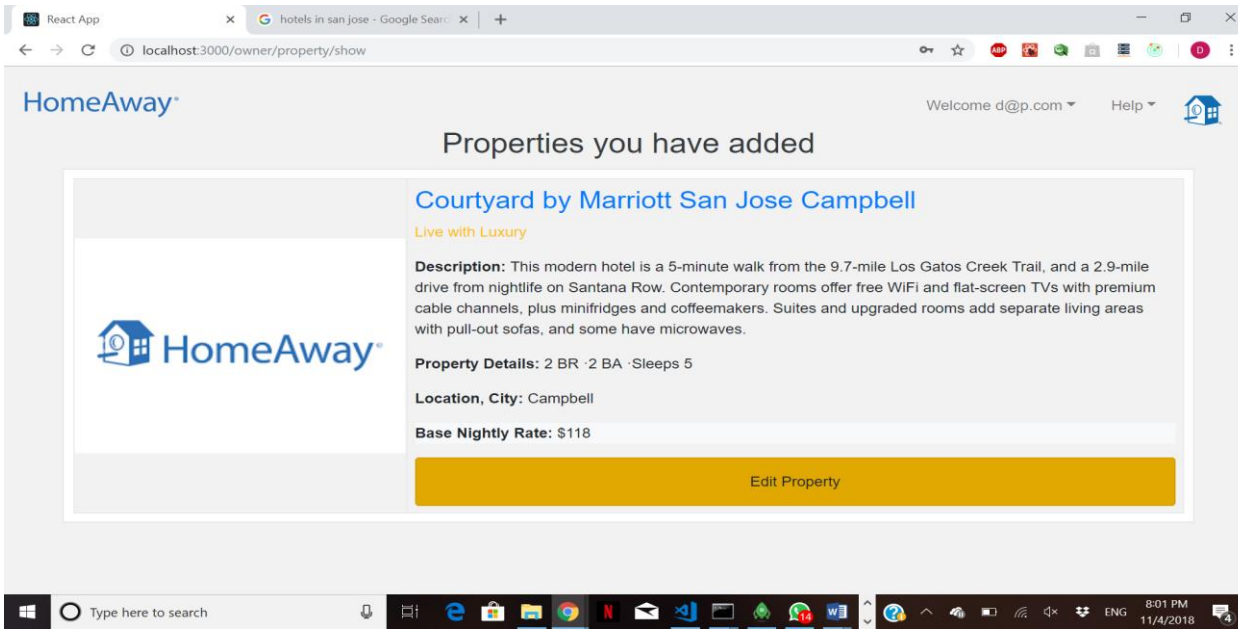

File Edit Selection View Go Debug Terminal Help
• LoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
console.log(nextProps.error)
if(nextProps.travelerInfo){
localStorage.setItem("loginuser",nextProps.travelerInfo._id)
localStorage.setItem("loginemail",nextProps.travelerInfo.email)
localStorage.setItem("token",nextProps.travelerInfo.token)
this.props.history.push('/traveller/home')
}else if(nextProps.error){
this.setState({
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
POST /property/5bdfbdb509ca266c981c0508/upload 200 99.472 ms - -
should be pushed
POST /property/5bdfbdb509ca266c981c0508/upload 200 458.675 ms - -
Trying to get properties listed by owner id: 5bdfab2a22c40d74a417d99d
in make request
{ owner_id: '5bdfab2a22c40d74a417d99d' }
in response: getOwnerProperties
in response1
true
in response2
{ getOwnerProperties: { '0': 24 } }
msg received
GET /owner/5bdfab2a22c40d74a417d99d/property 200 190.422 ms - -
GET /public/uploads/ha.jpeg 304 4.639 ms - -
GET /public/uploads/property-5bdfbdb509ca266c981c0508-xy1.jpg 200 5.069 ms - 6639
GET /public/uploads/property-5bdfbdb509ca266c981c0508-xy3.jpg 200 9.651 ms - 12741
GET /public/uploads/property-5bdfbdb509ca266c981c0508-xy2.jpg 200 11.471 ms - 9628
clearTrying to get details of property having id: 5bdfbdb509ca266c981c0508
in make request
{ propertyid: '5bdfbdb509ca266c981c0508' }
in response: getPropertyDetails
in response1
true
in response2
{ getPropertyDetails: { '0': 24 } }
master*
Ln 88, Col 71
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
7:59 PM
11/4/2018

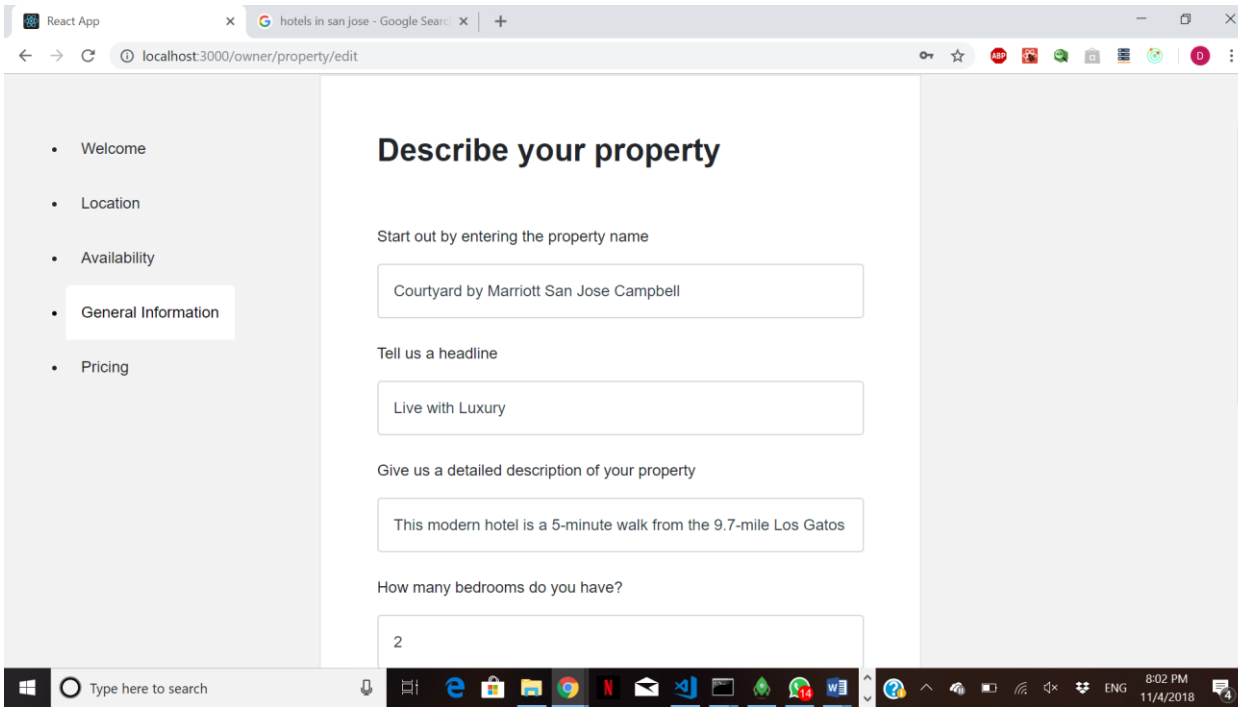


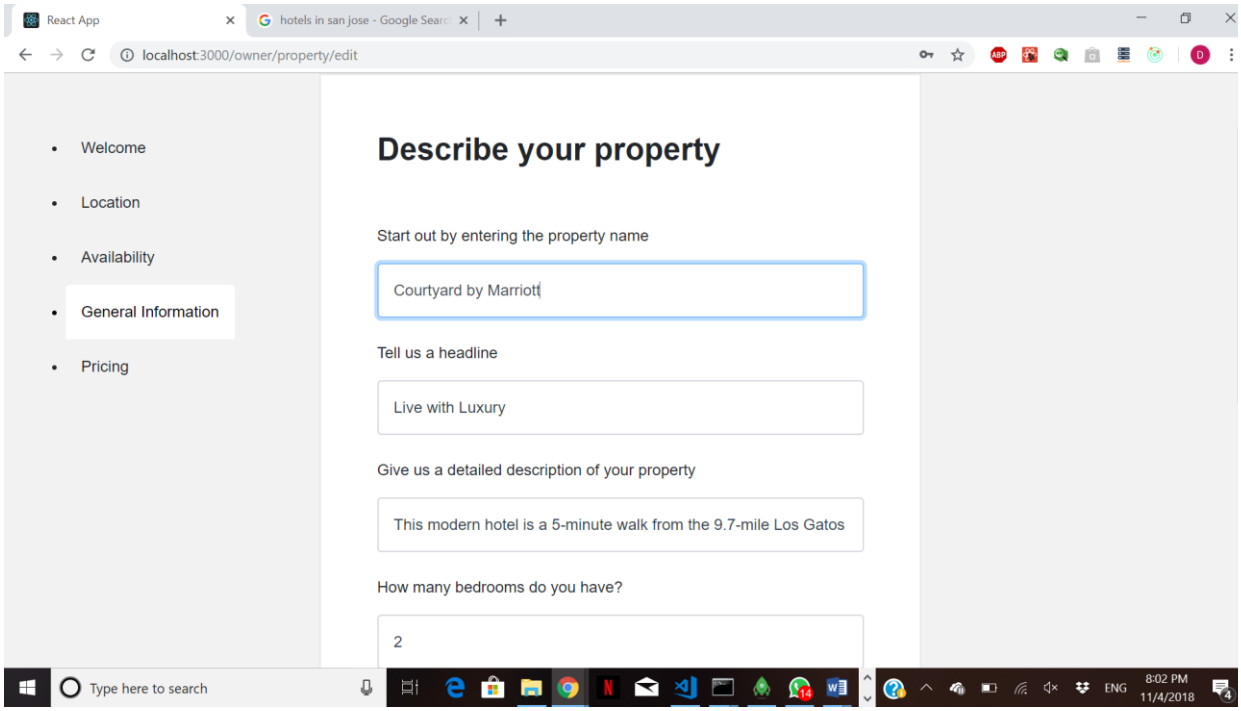
Explanation:

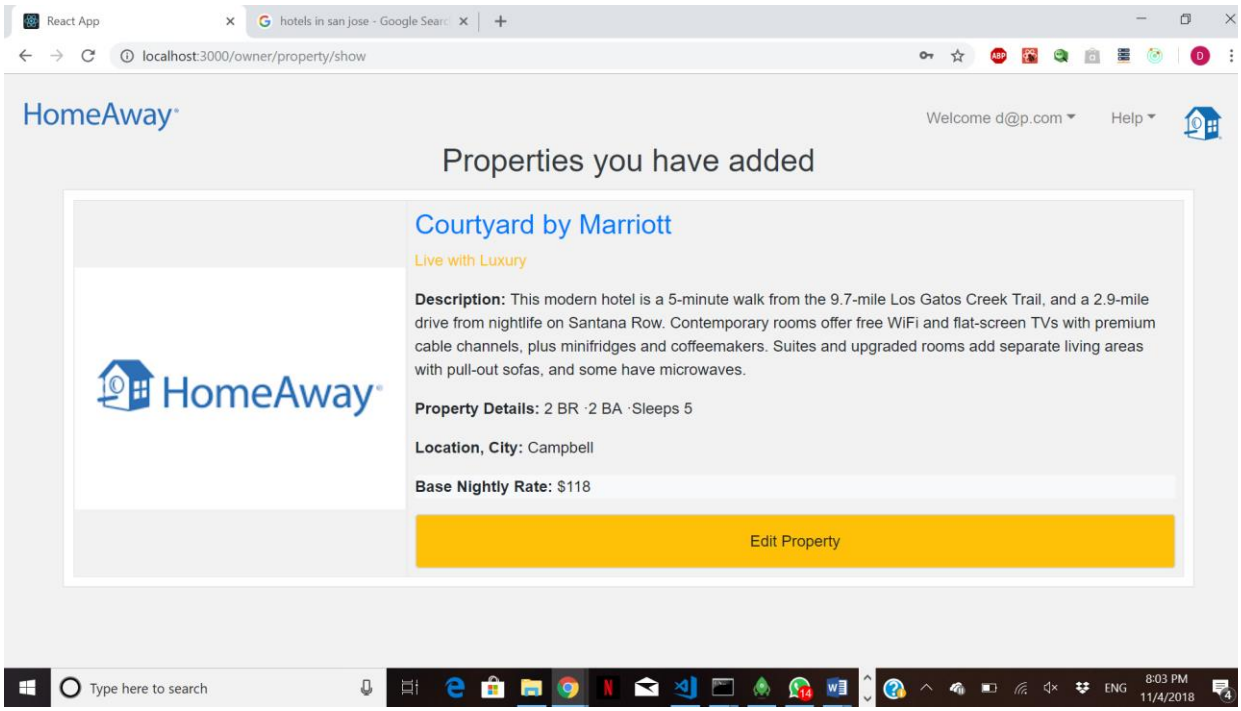
- These screenshots shows the properties listed by the owner.
- On the property list, if we click on the property name, it redirects to a propertyDetails page as shown in the screenshot.
- This request is being processed by "getOwnerProperties" topic in kafka.

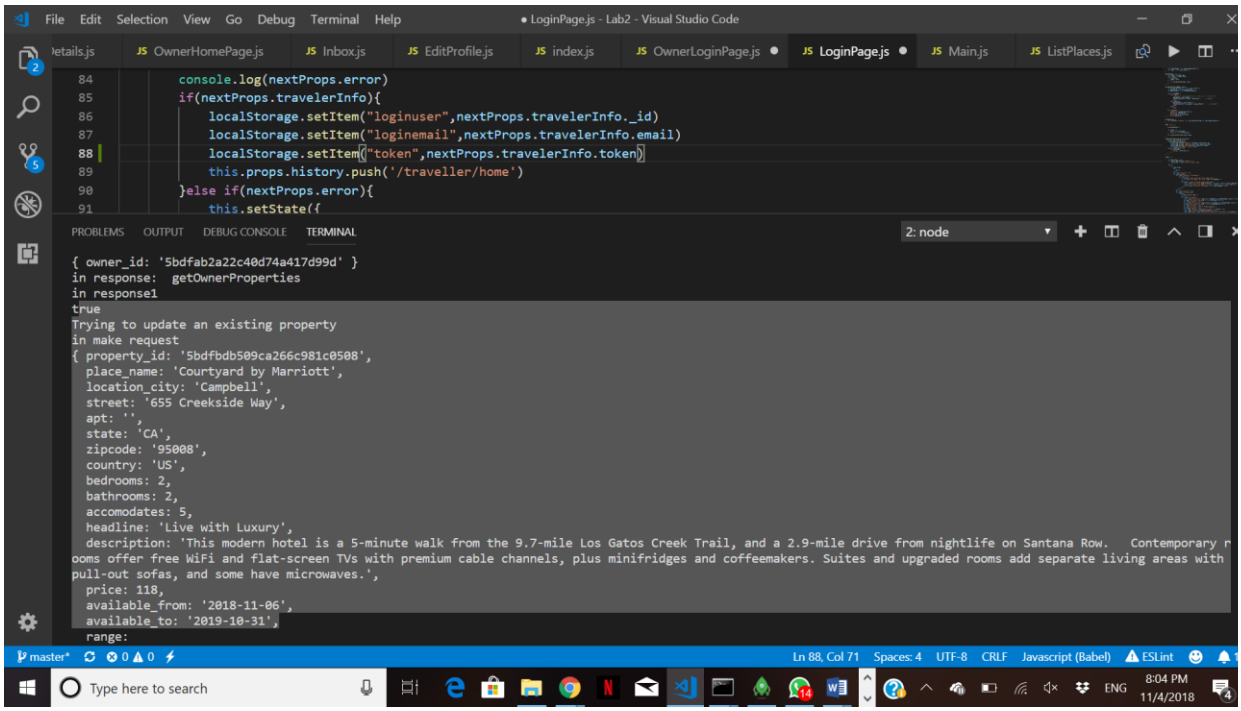
Editing property details

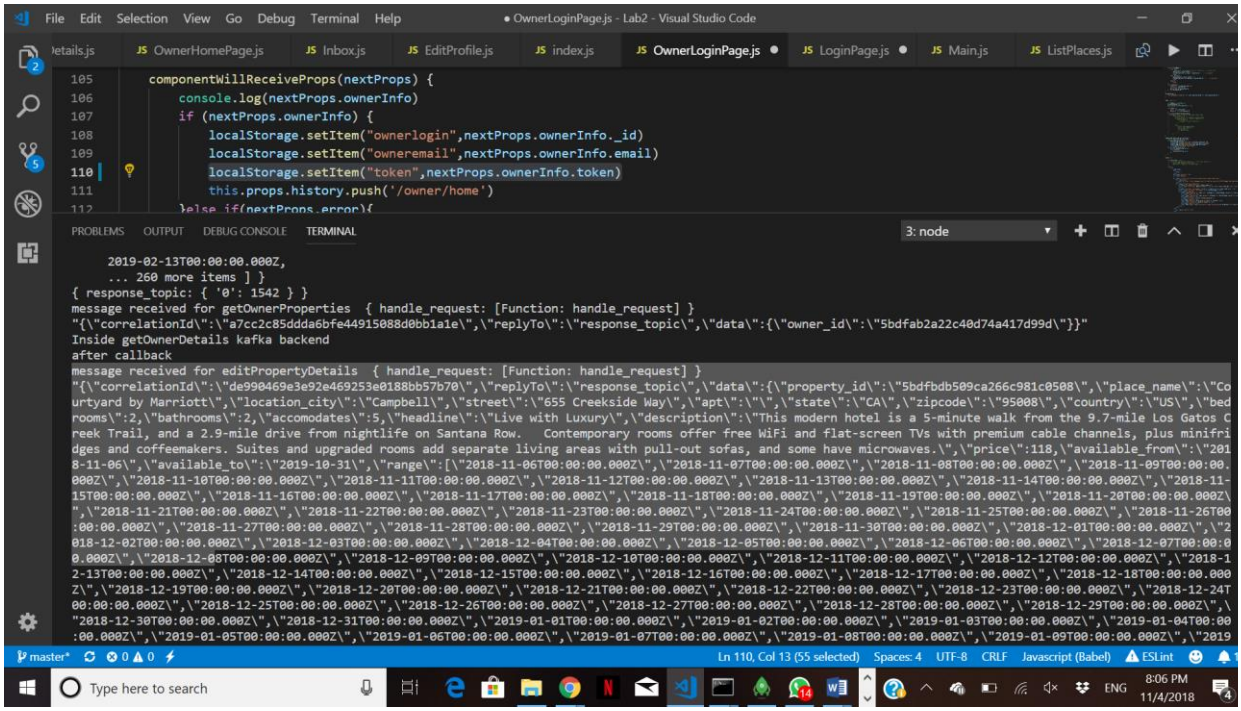


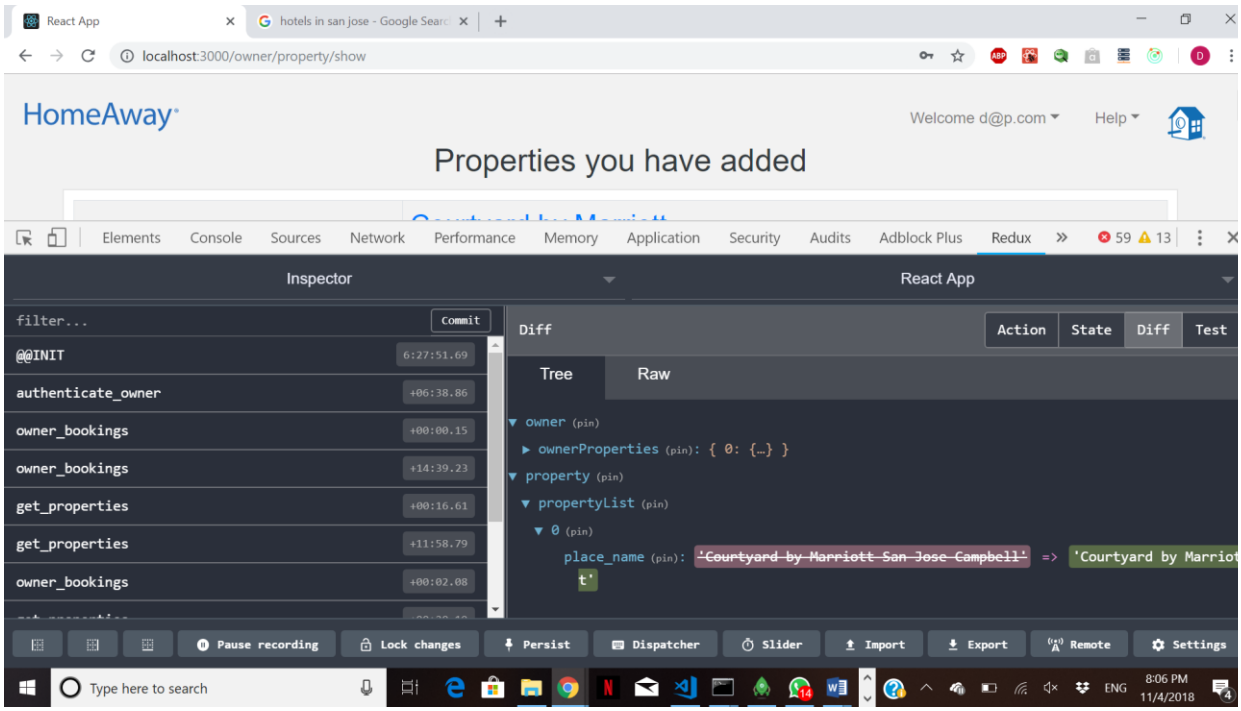

React App
hotels in san jose - Google Searc
localhost:3000/owner/property/edit
Welcome
Location
Availability
General Information
Pricing
Describe your property
Start out by entering the property name
Courtyard by Marriott San Jose Campbell
Tell us a headline
Live with Luxury
Give us a detailed description of your property
This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos
How many bedrooms do you have?
2
Type here to search
ENG
8:02 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/owner/property/edit
Welcome
Location
Availability
General Information
Pricing
Describe your property
Start out by entering the property name
Courtyard by Marriott
Tell us a headline
Live with Luxury
Give us a detailed description of your property
This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos
How many bedrooms do you have?
2
Type here to search
ENG
8:02 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/owner/property/show
HomeAway
Welcome d@p.com
Help
Properties you have added
Courtyard by Marriott
Live with Luxury
HomeAway
Description: This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos Creek Trail, and a 2.9-mile drive from nightlife on Santana Row. Contemporary rooms offer free WiFi and flat-screen TVs with premium cable channels, plus minifridges and coffeemakers. Suites and upgraded rooms add separate living areas with pull-out sofas, and some have microwaves.
Property Details: 2 BR ·2 BA ·Sleeps 5
Location, City: Campbell
Base Nightly Rate: $118
Edit Property
Type here to search
ENG
8:03 PM
11/4/2018


File Edit Selection View Go Debug Terminal Help
• LoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
console.log(nextProps.error)
if(nextProps.travelerInfo){
localStorage.setItem("loginuser",nextProps.travelerInfo._id)
localStorage.setItem("loginemail",nextProps.travelerInfo.email)
localStorage.setItem("token",nextProps.travelerInfo.token)
this.props.history.push('/traveller/home')
}else if(nextProps.error){
this.setState({
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
{ owner_id: '5bdfab2a22c40d74a417d99d' }
in response: getOwnerProperties
in response1
true
Trying to update an existing property
in make request
{ property_id: '5bdfbdb509ca266c981c0508',
place_name: 'Courtyard by Marriott',
location_city: 'Campbell',
street: '655 Creekside Way',
apt: '',
state: 'CA',
zipcode: '95008',
country: 'US',
bedrooms: 2,
bathrooms: 2,
accomodates: 5,
headline: 'Live with Luxury',
description: 'This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos Creek Trail, and a 2.9-mile drive from nightlife on Santana Row. Contemporary r
ooms offer free WiFi and flat-screen TVs with premium cable channels, plus minifridges and coffeemakers. Suites and upgraded rooms add separate living areas with
pull-out sofas, and some have microwaves.',
price: 118,
available_from: '2018-11-06',
available_to: '2019-10-31',
range:
master*
Ln 88, Col 71
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
8:04 PM
11/4/2018

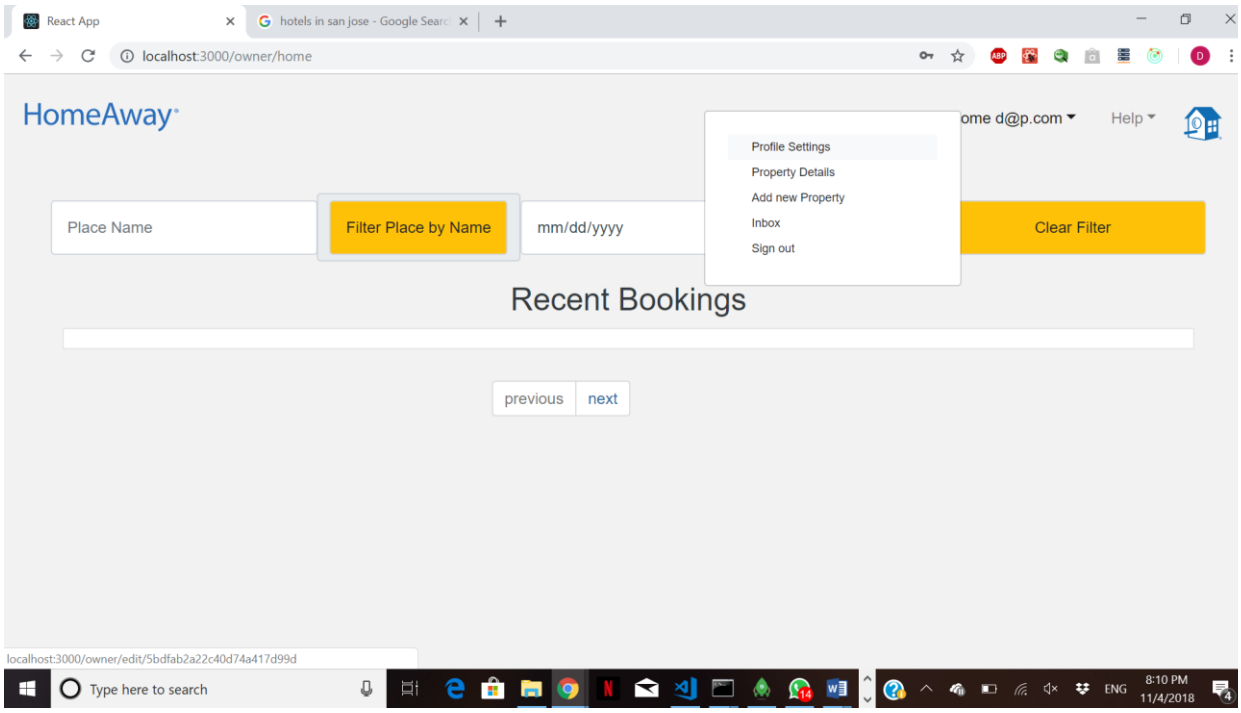

File Edit Selection View Go Debug Terminal Help
• OwnerLoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
}else if(nextProps.error){
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
3: node
2019-02-13T00:00:00.000Z,
... 260 more items] }
{ response_topic: { '0': 1542 } }
message received for getOwnerProperties { handle_request: [Function: handle_request] }
"{\"correlationId\":\"a7cc2c85ddda6bfe44915088d0bb1a1e\",\"replyTo\":\"response_topic\",\"data\":{\"owner_id\":\"5bdfab2a22c40d74a417d99d\"}}"
Inside getOwnerDetails kafka backend
after callback
message received for editPropertyDetails { handle_request: [Function: handle_request] }
"{\"correlationId\":\"de990469e3e92e469253e0188bb57b70\",\"replyTo\":\"response_topic\",\"data\":{\"property_id\":\"5bdfbdb509ca266c981c0508\",\"place_name\":\"Co
urtyard by Marriott\",\"location_city\":\"Campbell\",\"street\":\"655 Creekside Way\",\"apt\":\"\",\"state\":\"CA\",\"zipcode\":\"95008\",\"country\":\"US\",\"bed
rooms\":2,\"bathrooms\":2,\"accomodates\":5,\"headline\":\"Live with Luxury\",\"description\":\"This modern hotel is a 5-minute walk from the 9.7-mile Los Gatos C
reek Trail, and a 2.9-mile drive from nightlife on Santana Row. Contemporary rooms offer free WiFi and flat-screen TVs with premium cable channels, plus minifri
dges and coffeemakers. Suites and upgraded rooms add separate living areas with pull-out sofas, and some have microwaves.\",\"price\":118,\"available_from\":\"201
8-11-06\",\"available_to\":\"2019-10-31\",\"range\":[\"2018-11-06T00:00:00.000Z\",\"2018-11-07T00:00:00.000Z\",\"2018-11-08T00:00:00.000Z\",\"2018-11-09T00:00:00.
000Z\",\"2018-11-10T00:00:00.000Z\",\"2018-11-11T00:00:00.000Z\",\"2018-11-12T00:00:00.000Z\",\"2018-11-13T00:00:00.000Z\",\"2018-11-14T00:00:00.000Z\",\"2018-11-
15T00:00:00.000Z\",\"2018-11-16T00:00:00.000Z\",\"2018-11-17T00:00:00.000Z\",\"2018-11-18T00:00:00.000Z\",\"2018-11-19T00:00:00.000Z\",\"2018-11-20T00:00:00.000Z\
",\"2018-11-21T00:00:00.000Z\",\"2018-11-22T00:00:00.000Z\",\"2018-11-23T00:00:00.000Z\",\"2018-11-24T00:00:00.000Z\",\"2018-11-25T00:00:00.000Z\",\"2018-11-26T00
:00:00.000Z\",\"2018-11-27T00:00:00.000Z\",\"2018-11-28T00:00:00.000Z\",\"2018-11-29T00:00:00.000Z\",\"2018-11-30T00:00:00.000Z\",\"2018-12-01T00:00:00.000Z\",\"2
018-12-02T00:00:00.000Z\",\"2018-12-03T00:00:00.000Z\",\"2018-12-04T00:00:00.000Z\",\"2018-12-05T00:00:00.000Z\",\"2018-12-06T00:00:00.000Z\",\"2018-12-07T00:00:0
0.000Z\",\"2018-12-08T00:00:00.000Z\",\"2018-12-09T00:00:00.000Z\",\"2018-12-10T00:00:00.000Z\",\"2018-12-11T00:00:00.000Z\",\"2018-12-12T00:00:00.000Z\",\"2018-1
2-13T00:00:00.000Z\",\"2018-12-14T00:00:00.000Z\",\"2018-12-15T00:00:00.000Z\",\"2018-12-16T00:00:00.000Z\",\"2018-12-17T00:00:00.000Z\",\"2018-12-18T00:00:00.000
Z\",\"2018-12-19T00:00:00.000Z\",\"2018-12-20T00:00:00.000Z\",\"2018-12-21T00:00:00.000Z\",\"2018-12-22T00:00:00.000Z\",\"2018-12-23T00:00:00.000Z\",\"2018-12-24T
00:00:00.000Z\",\"2018-12-25T00:00:00.000Z\",\"2018-12-26T00:00:00.000Z\",\"2018-12-27T00:00:00.000Z\",\"2018-12-28T00:00:00.000Z\",\"2018-12-29T00:00:00.000Z\",\
"2018-12-30T00:00:00.000Z\",\"2018-12-31T00:00:00.000Z\",\"2019-01-01T00:00:00.000Z\",\"2019-01-02T00:00:00.000Z\",\"2019-01-03T00:00:00.000Z\",\"2019-01-04T00:00
:00.000Z\",\"2019-01-05T00:00:00.000Z\",\"2019-01-06T00:00:00.000Z\",\"2019-01-07T00:00:00.000Z\",\"2019-01-08T00:00:00.000Z\",\"2019-01-09T00:00:00.000Z\",\"2019
master*
Ln 110, Col 13 (55 selected)
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
8:06 PM
11/4/2018

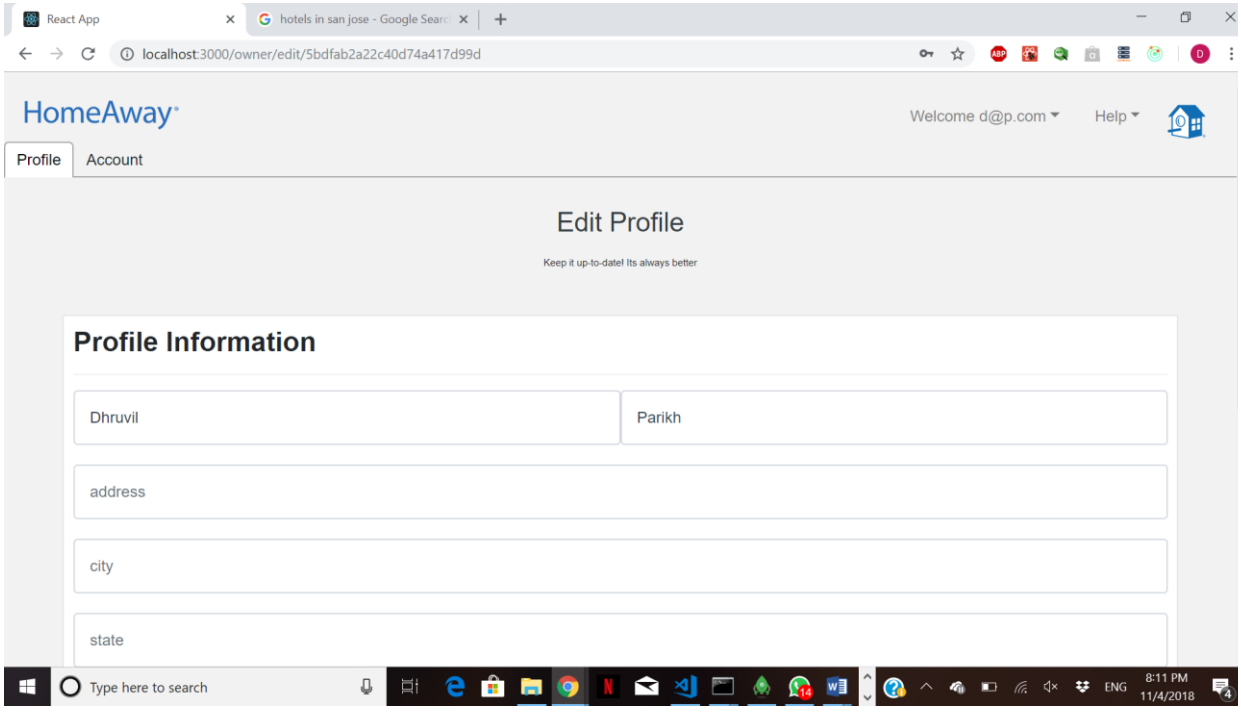


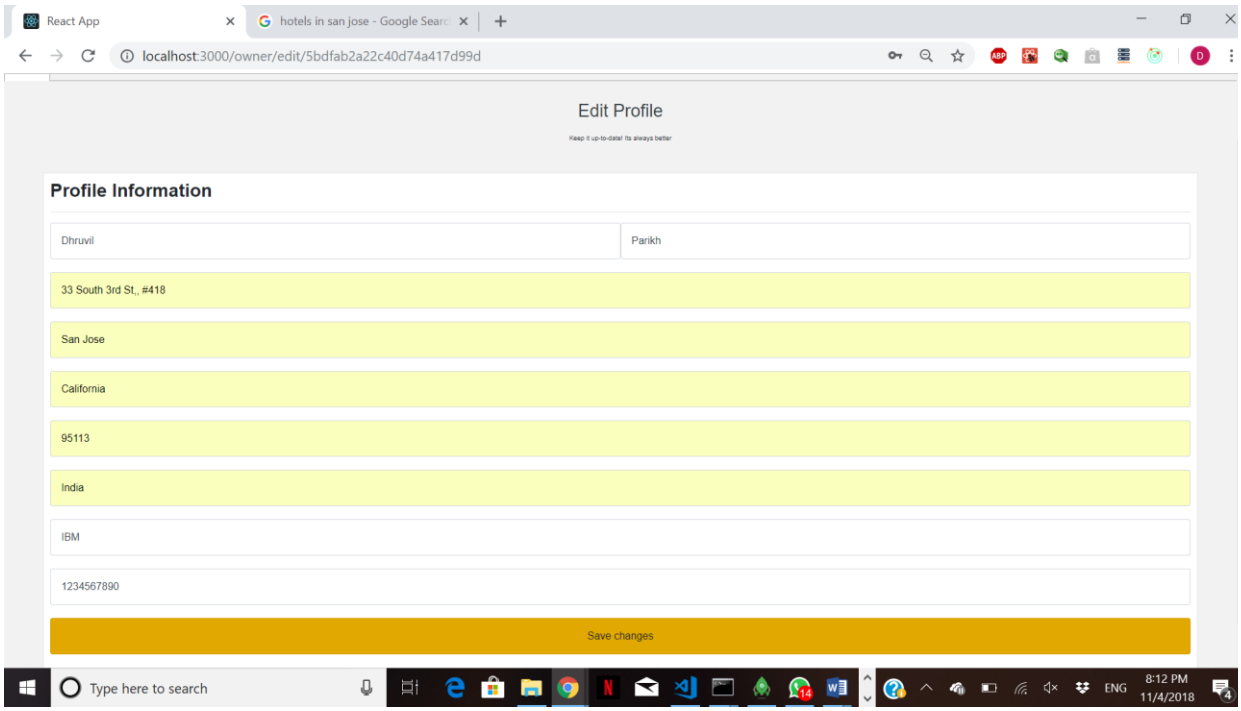
Explanation:

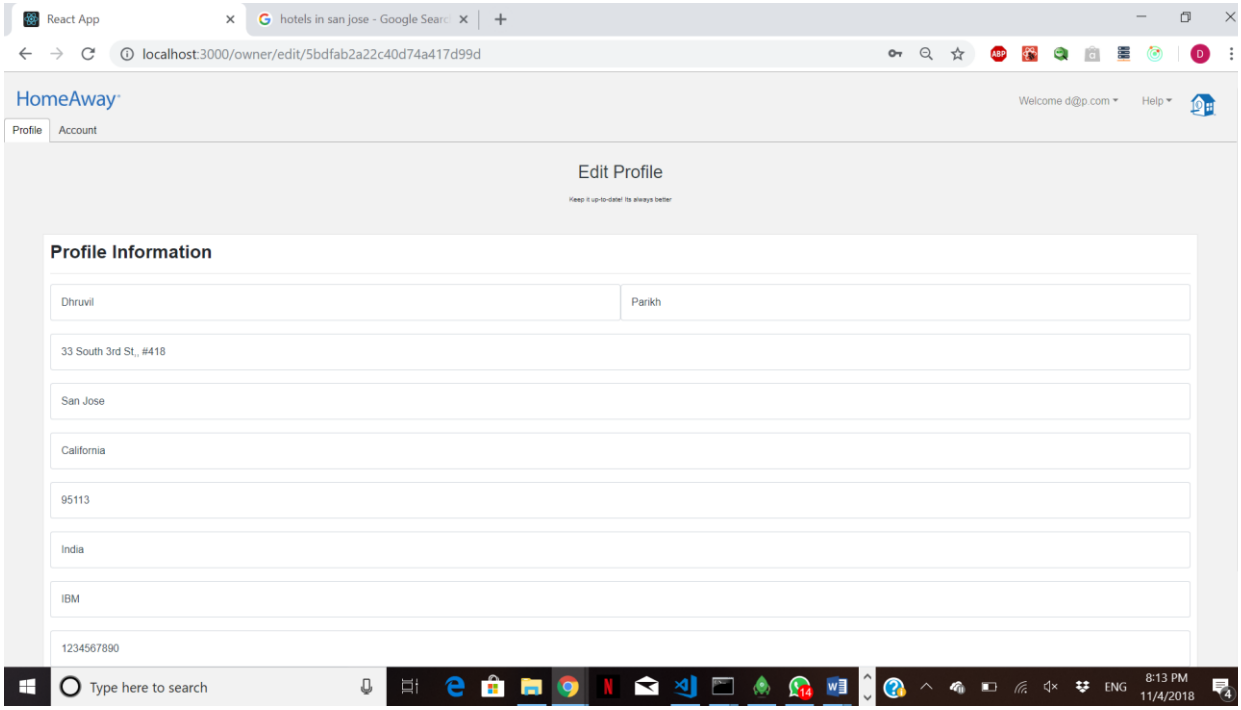
- These screenshots show the steps to edit the property details.
- Here, we are changing the property name from “Courtyard by Marriot San Jose Campbell” to only “Courtyard by Marriot”.
- The redux screenshot also shows us that the property has been correctly edited and the redux store reflects the changing as shown in the screenshot.

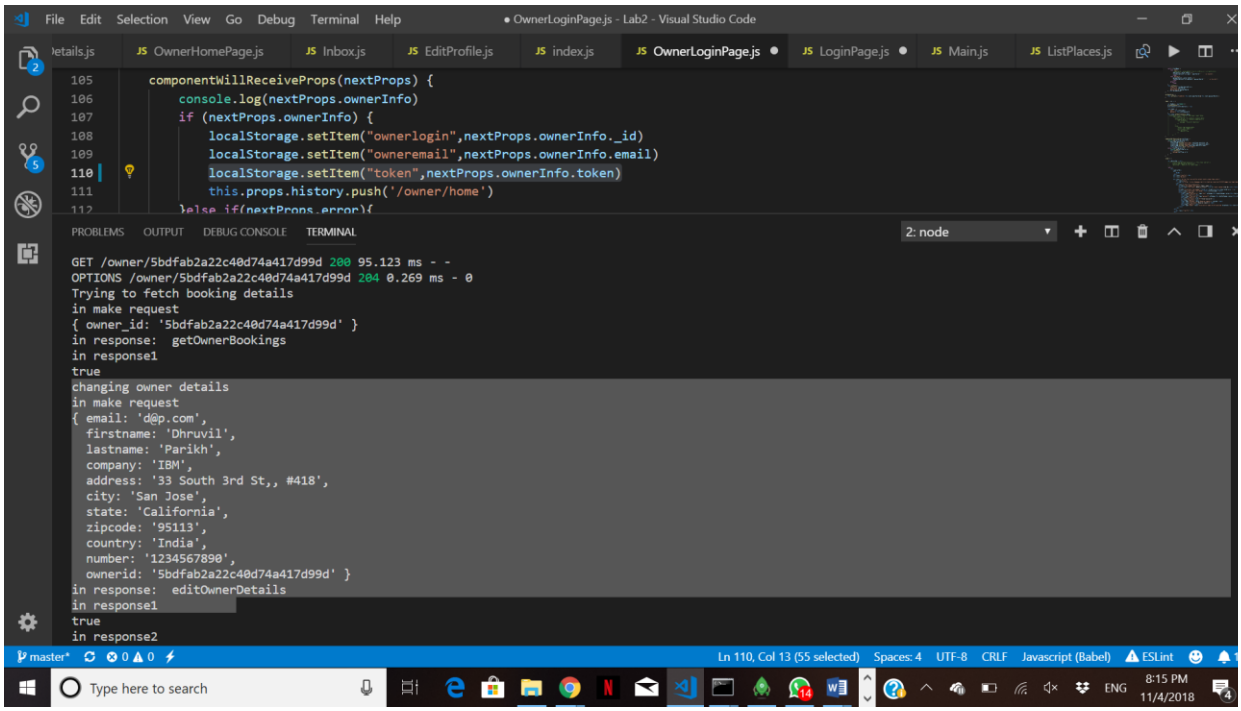
Editing the owner's detail like billing address, contact no and other things

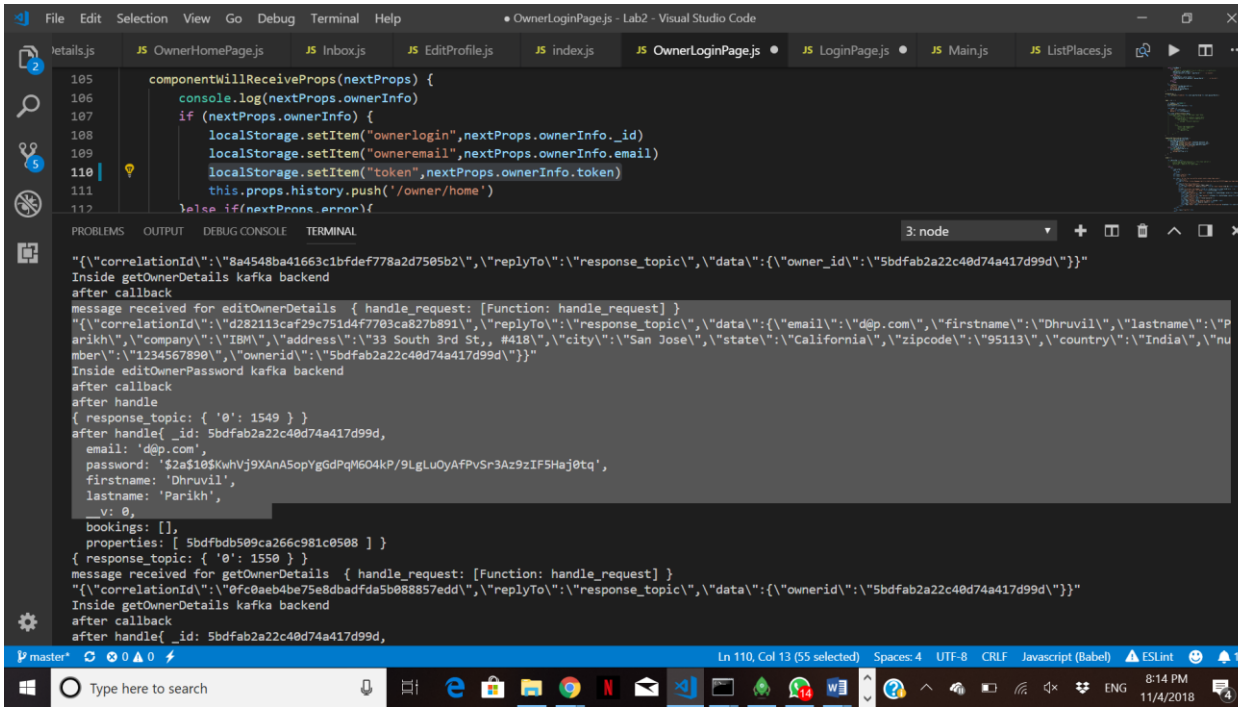


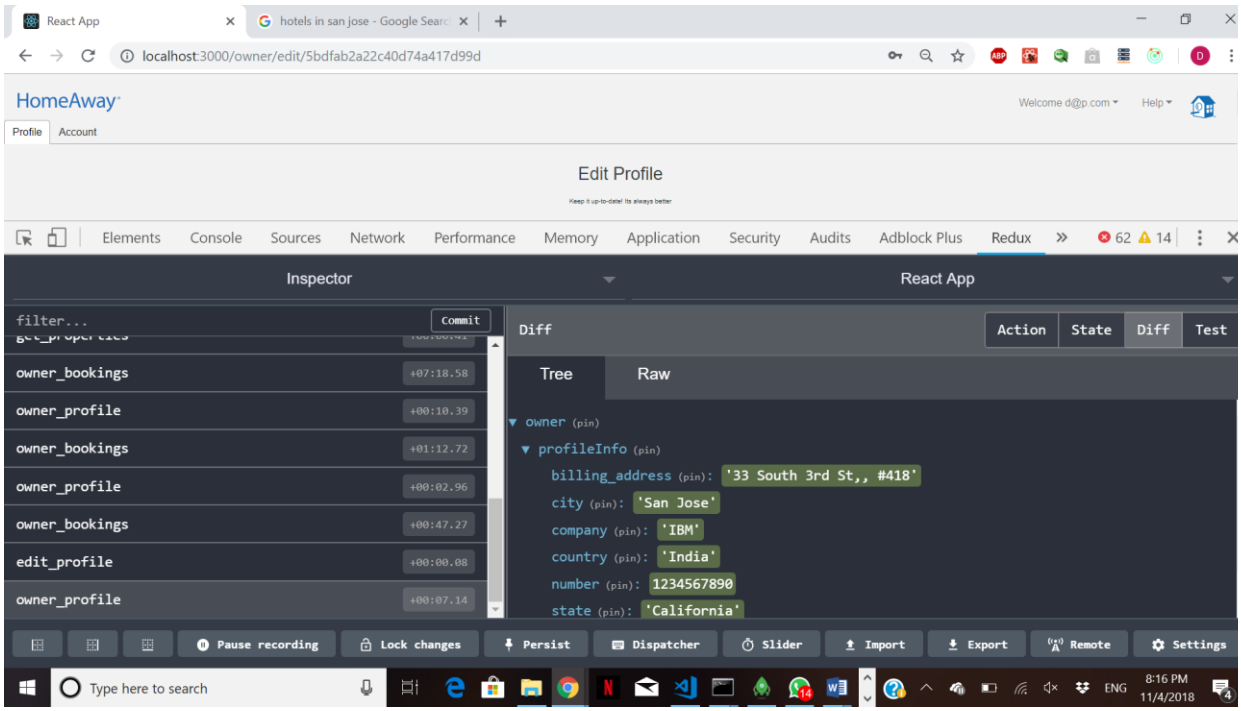



React App
hotels in san jose - Google Searc
localhost:3000/owner/edit/5bdfab2a22c40d74a417d99d
Edit Profile
Keep it up-to-date! Its always better
Profile Information
Dhruvil
Parikh
33 South 3rd St,, #418
San Jose
California
95113
India
IBM
1234567890
Save changes
Type here to search
ENG
8:12 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/owner/edit/5bdfab2a22c40d74a417d99d
HomeAway
Welcome d@p.com
Help
Profile
Account
Edit Profile
Keep it up-to-date! Its always better
Profile Information
Dhruvil
Parikh
33 South 3rd St,, #418
San Jose
California
95113
India
IBM
1234567890
Type here to search
ENG
8:13 PM
11/4/2018


File Edit Selection View Go Debug Terminal Help
• OwnerLoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
GET /owner/5bdfab2a22c40d74a417d99d 200 95.123 ms - -
OPTIONS /owner/5bdfab2a22c40d74a417d99d 204 0.269 ms - 0
Trying to fetch booking details
in make request
{ owner_id: '5bdfab2a22c40d74a417d99d' }
in response: getOwnerBookings
in response1
true
changing owner details
in make request
{ email: 'd@p.com',
firstname: 'Dhruvil',
lastname: 'Parikh',
company: 'IBM',
address: '33 South 3rd St,, #418',
city: 'San Jose',
state: 'California',
zipcode: '95113',
country: 'India',
number: '1234567890',
ownerid: '5bdfab2a22c40d74a417d99d' }
in response: editOwnerDetails
in response1
true
in response2
Ln 110, Col 13 (55 selected) Spaces: 4 UTF-8 CRLF Javascript (Babel) ESLint
Type here to search
ENG
8:15 PM
11/4/2018

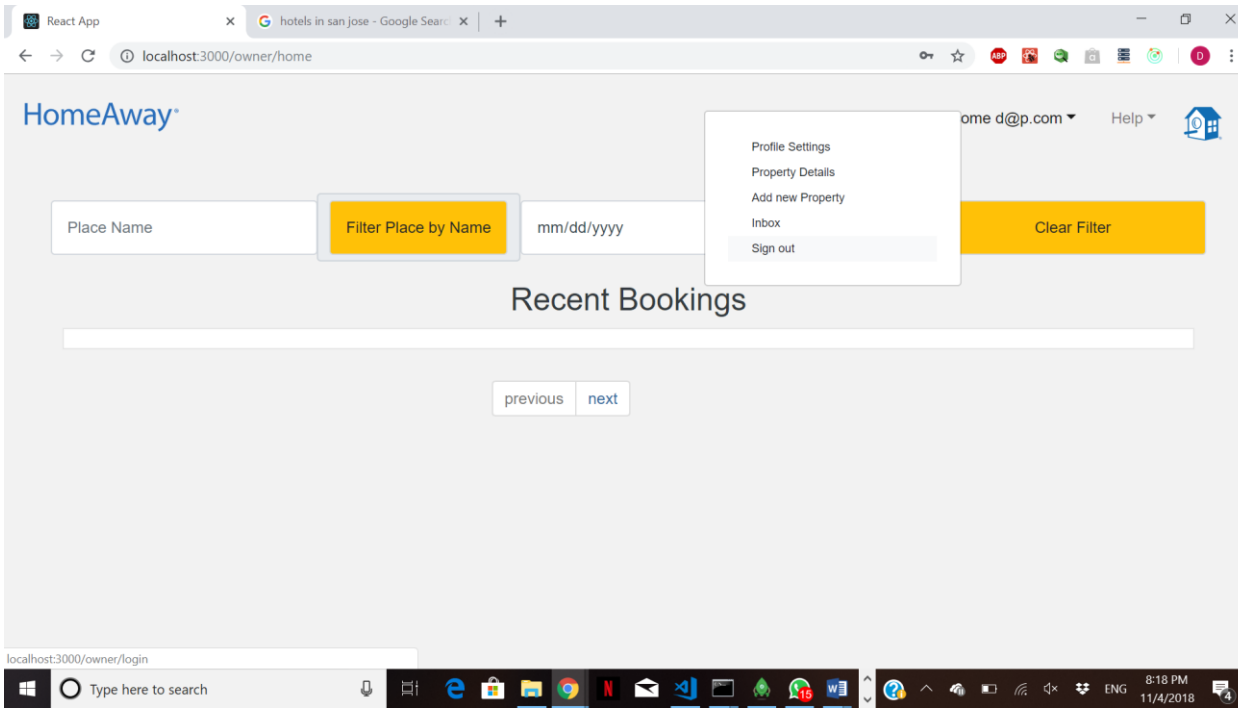

File Edit Selection View Go Debug Terminal Help
• OwnerLoginPage.js - Lab2 - Visual Studio Code
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
3: node
"{\"correlationId\":\"8a4548ba41663c1bfdef778a2d7505b2\",\"replyTo\":\"response_topic\",\"data\":{\"owner_id\":\"5bdfab2a22c40d74a417d99d\"}}"
Inside getOwnerDetails kafka backend
after callback
message received for editOwnerDetails { handle_request: [Function: handle_request] }
"{\"correlationId\":\"d282113caf29c751d4f7703ca827b891\",\"replyTo\":\"response_topic\",\"data\":{\"email\":\"d@p.com\",\"firstname\":\"Dhruvil\",\"lastname\":\"Parikh\",\"company\":\"IBM\",\"address\":\"33 South 3rd St,, #418\",\"city\":\"San Jose\",\"state\":\"California\",\"zipcode\":\"95113\",\"country\":\"India\",\"number\":\"1234567890\",\"ownerid\":\"5bdfab2a22c40d74a417d99d\"}}"
Inside editOwnerPassword kafka backend
after callback
after handle
{ response_topic: { '0': 1549 } }
after handle{ _id: 5bdfab2a22c40d74a417d99d,
email: 'd@p.com',
password: '$2a$10$KwhVj9XAnA5opYgGdPqM6O4kP/9LgLuOyAfPvSr3Az9zIF5Haj0tq',
firstname: 'Dhruvil',
lastname: 'Parikh',
__v: 0,
bookings: [],
properties: [5bdfbdb509ca266c981c0508] }
{ response_topic: { '0': 1550 } }
message received for getOwnerDetails { handle_request: [Function: handle_request] }
"{\"correlationId\":\"0fc0aeb4be75e8dbadfda5b088857edd\",\"replyTo\":\"response_topic\",\"data\":{\"ownerid\":\"5bdfab2a22c40d74a417d99d\"}}"
Inside getOwnerDetails kafka backend
after callback
after handle{ _id: 5bdfab2a22c40d74a417d99d,
Ln 110, Col 13 (55 selected) Spaces: 4 UTF-8 CRLF Javascript (Babel) ESLint
Type here to search
ENG
8:14 PM
11/4/2018

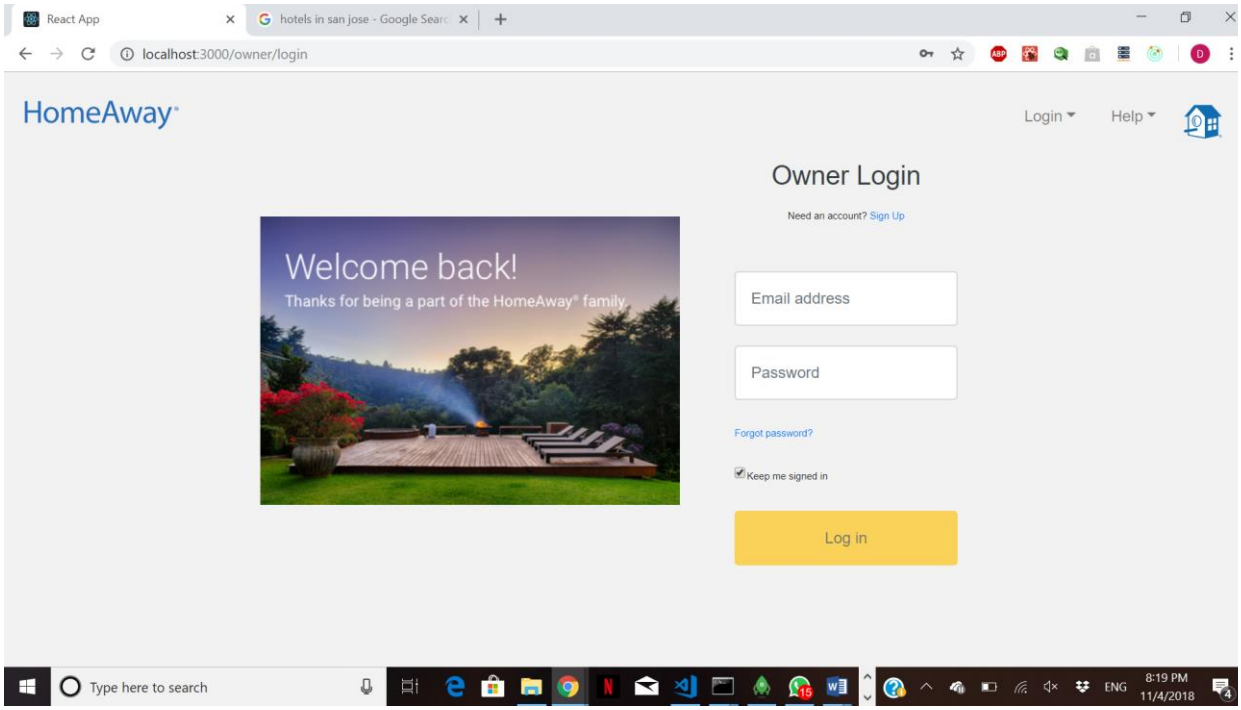


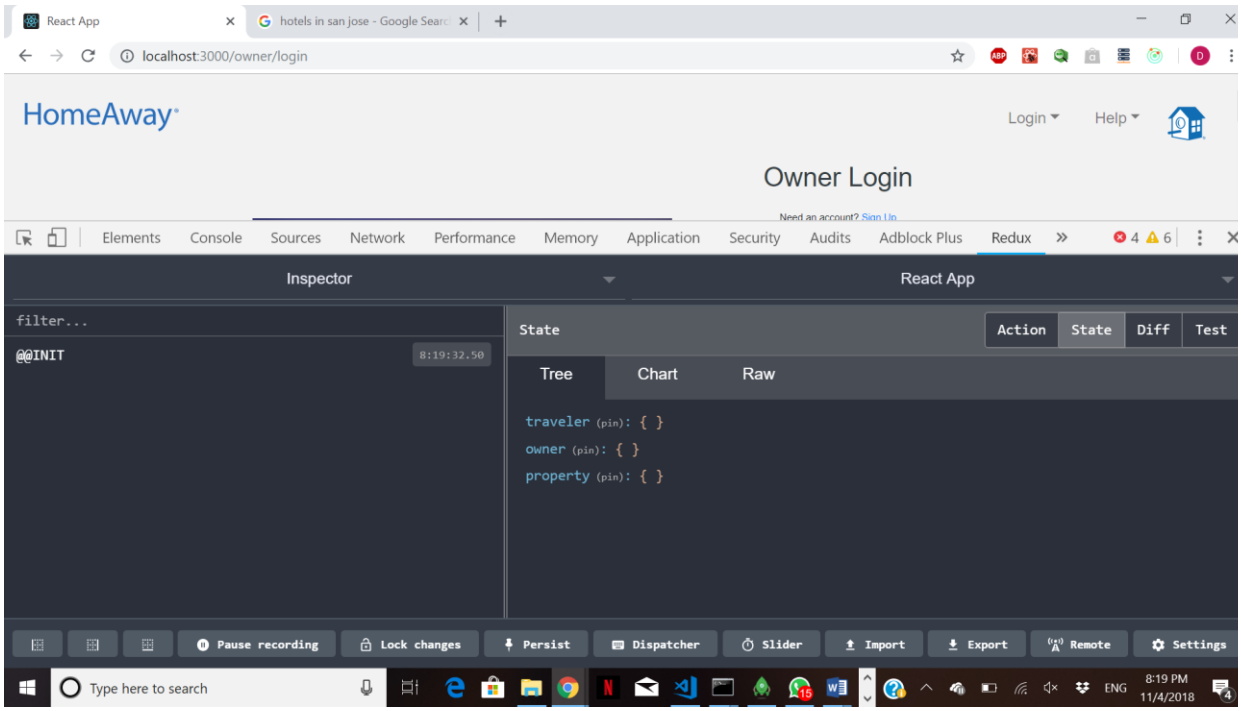
Explanation:

- These steps show how to change the details after creating an account.
- As you can see in the 2nd screenshot, the details which were already provided during the signup are pre-loaded using componentDidMount() lifecycle method.
- We added the basic information which weren't asked while creating the account.
- On saving the changes, the change request is passed through "editOwnerDetails" of the kafka-backend.
- The redux store also shows the difference in store, indicating that the following fields have been added to the owner profile.

Logging off from the owner account



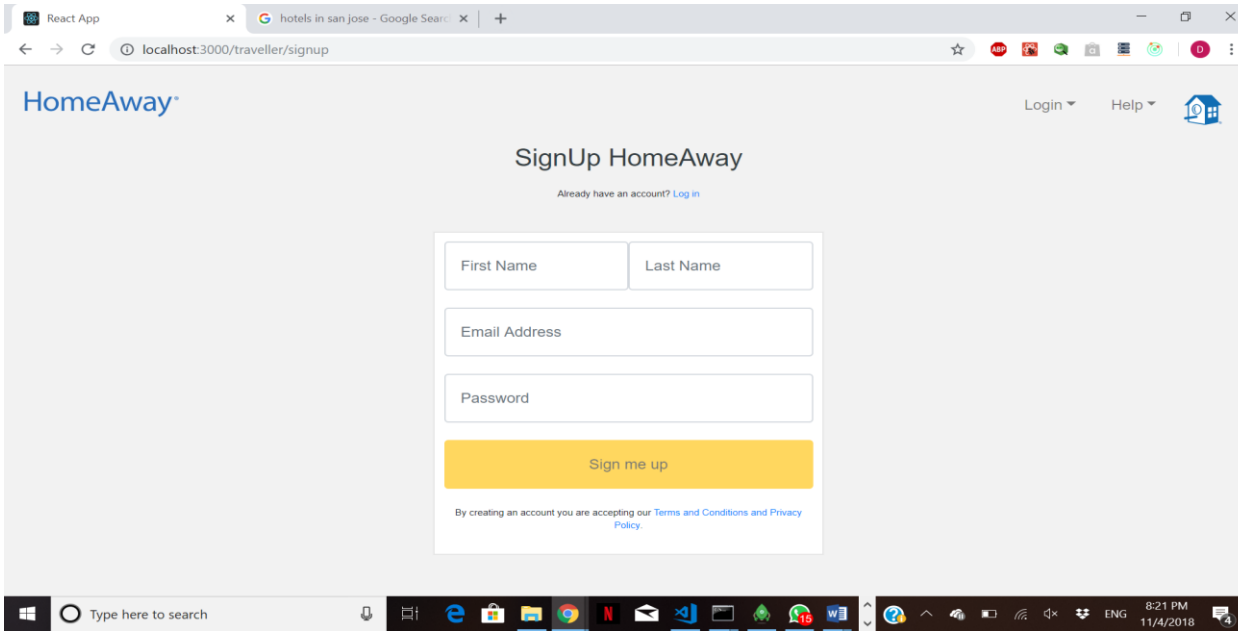


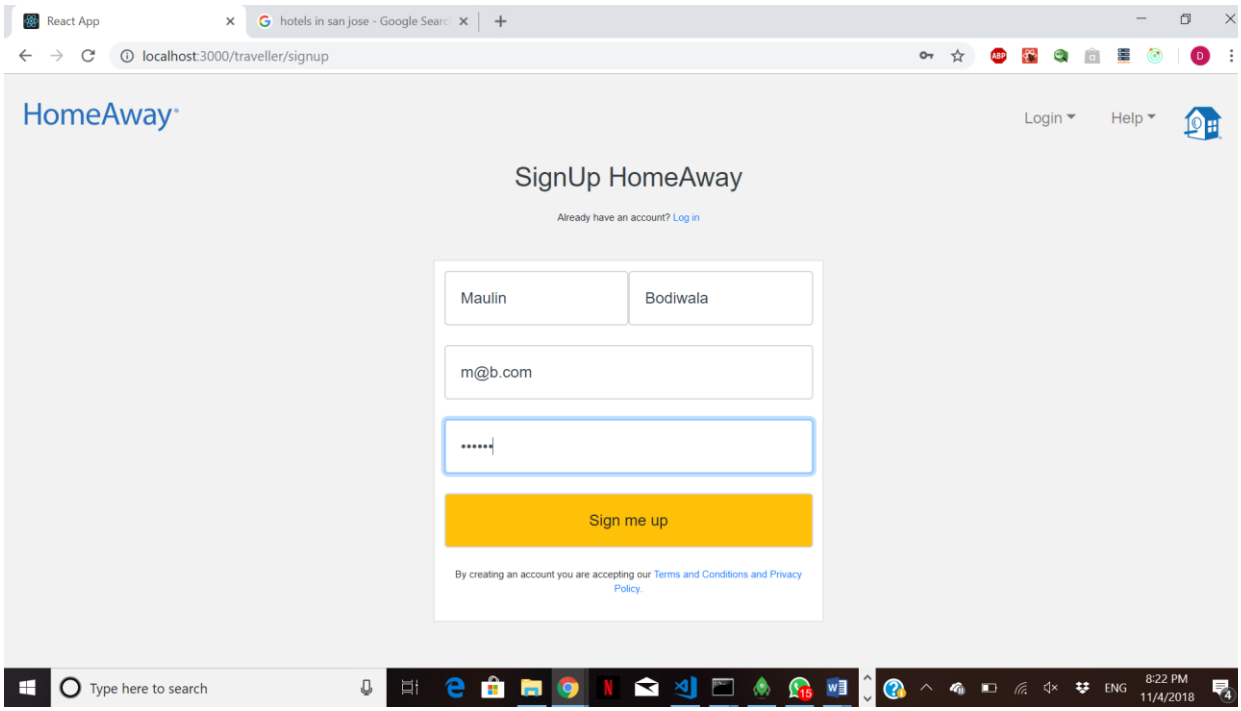


Explanation:

- Once the owner logs out, the owner will be directed to the login page again and the options are gone too.
- Redux store is cleared too.

Creating a new travel account

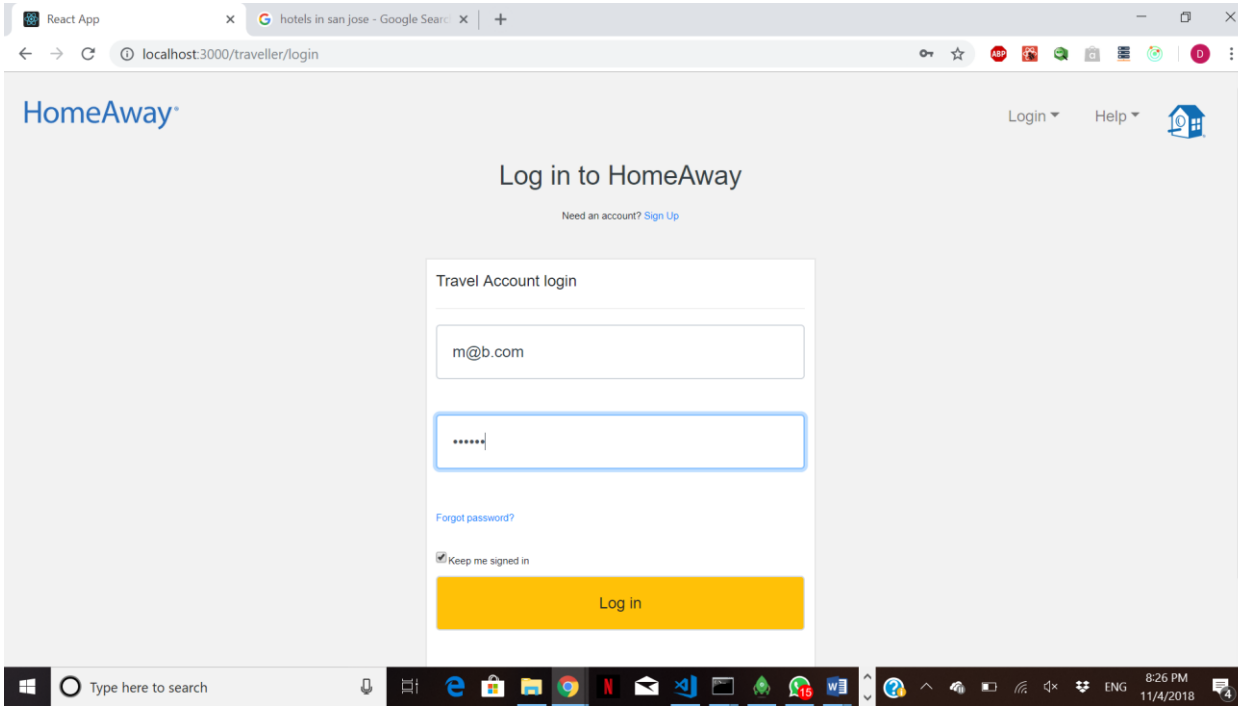


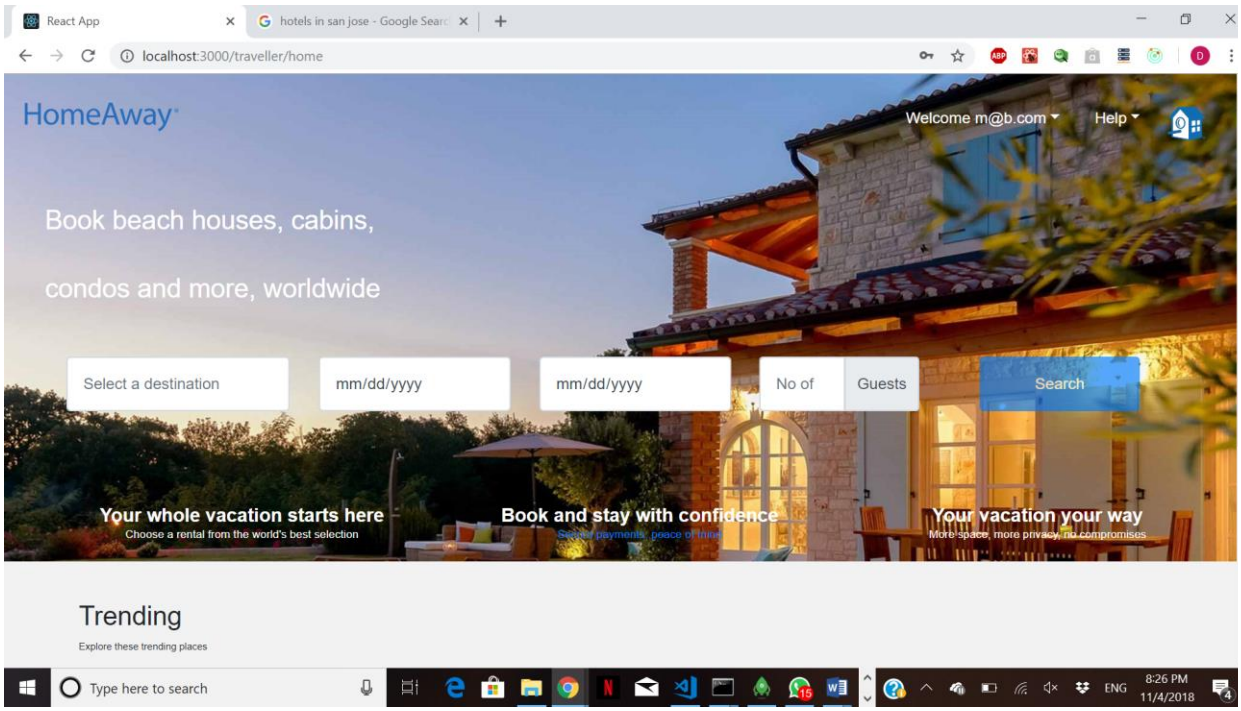


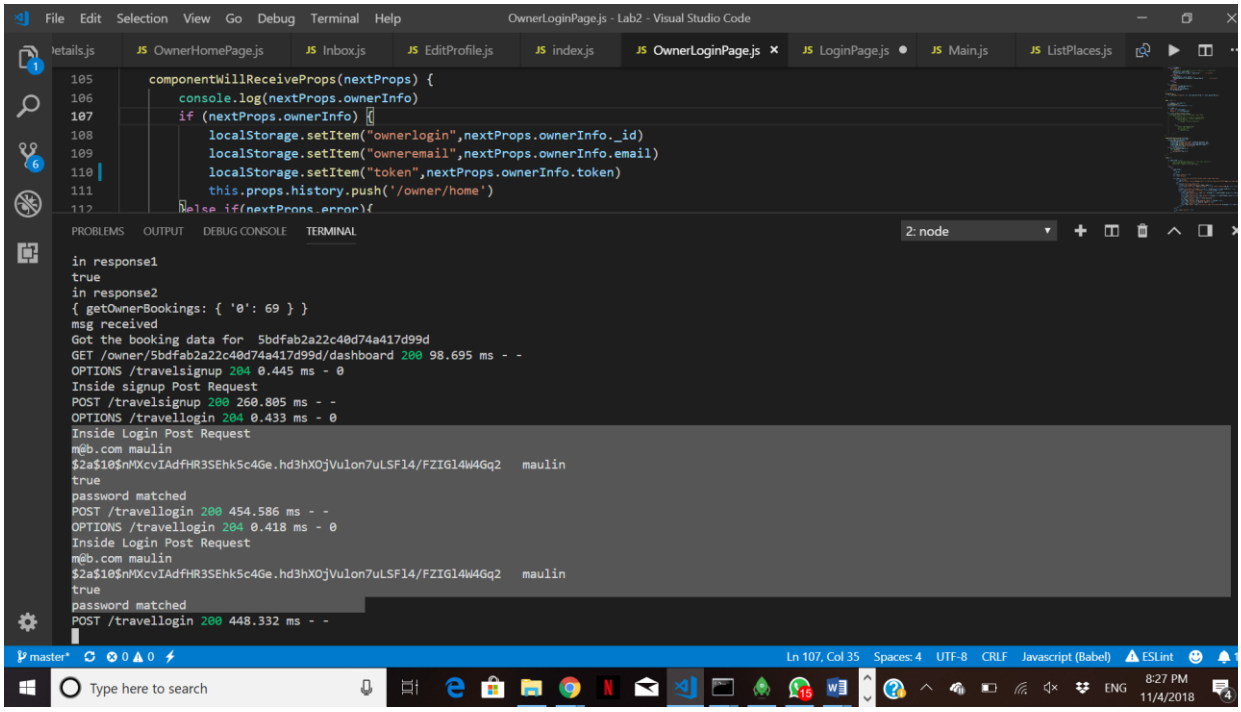
Explanation:

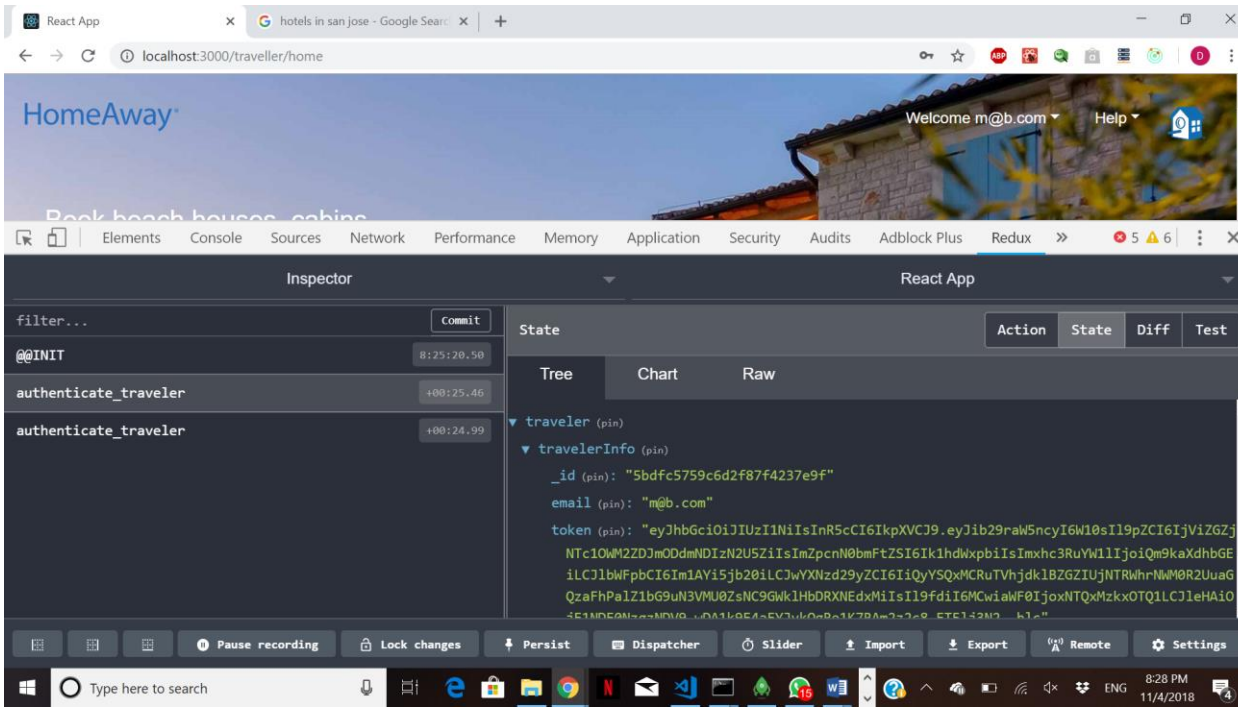
- We created a new travel account with Name: Maulin Bodiwala and Email id: m@b.com and password=maulin which has been encrypted using bcrypt.

Logging in with new travel account



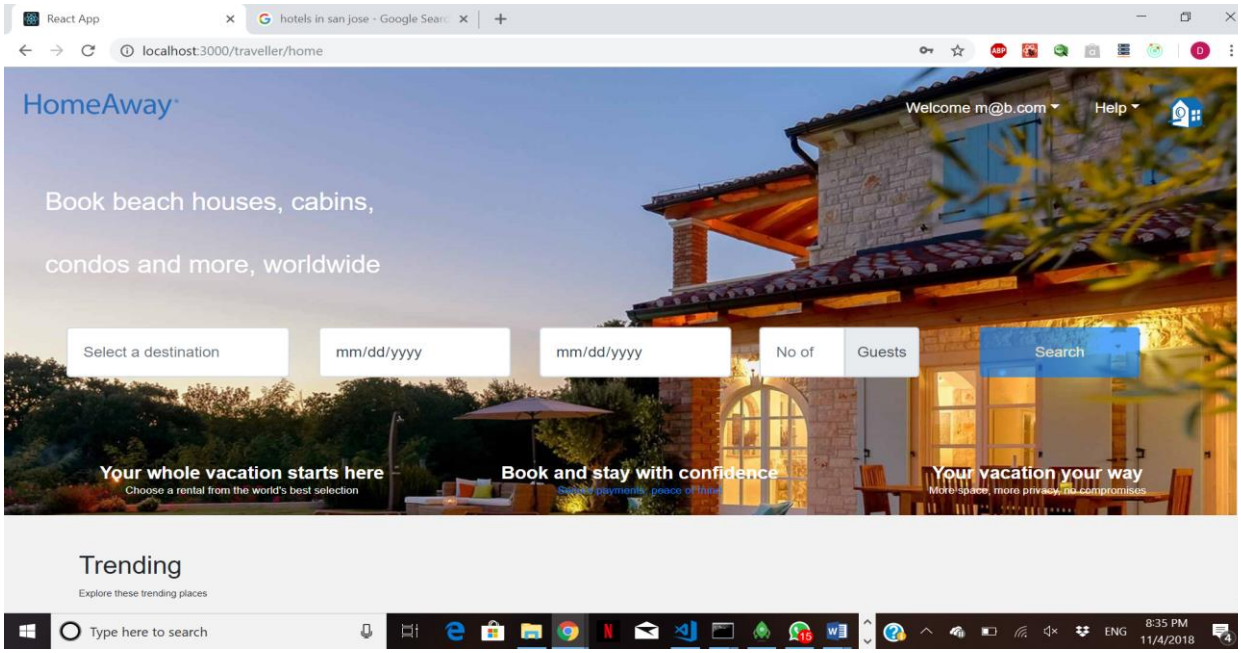

React App
hotels in san jose - Google Searc
localhost:3000/traveller/home
HomeAway
Welcome m@b.com
Help
Book beach houses, cabins,
condos and more, worldwide
Select a destination
mm/dd/yyyy
mm/dd/yyyy
No of
Guests
Search
Your whole vacation starts here
Choose a rental from the world's best selection
Book and stay with confidence
Secure payments, peace of mind
Your vacation your way
More space, more privacy, no compromises
Trending
Explore these trending places
Type here to search
ENG
8:26 PM
11/4/2018

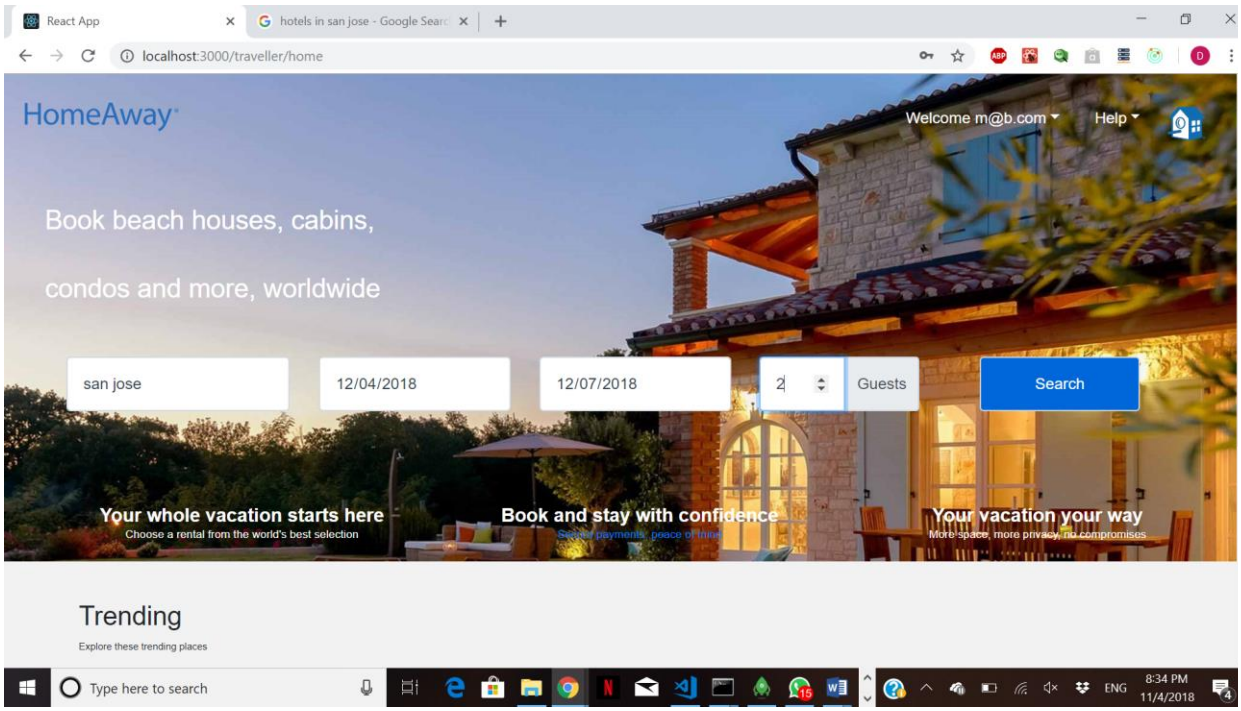

File Edit Selection View Go Debug Terminal Help
OwnerLoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
in response1
true
in response2
{ getOwnerBookings: { '0': 69 } }
msg received
Got the booking data for 5bdfab2a22c40d74a417d99d
GET /owner/5bdfab2a22c40d74a417d99d/dashboard 200 98.695 ms - -
OPTIONS /travelsignup 204 0.445 ms - 0
Inside signup Post Request
POST /travelsignup 200 260.805 ms - -
OPTIONS /travellogin 204 0.433 ms - 0
Inside Login Post Request
m@b.com maulin
$2a$10$nMXcvIAdfHR3SEhk5c4Ge.hd3hXOjVulon7uLSFl4/FZIGl4W4Gq2 maulin
true
password matched
POST /travellogin 200 454.586 ms - -
OPTIONS /travellogin 204 0.418 ms - 0
Inside Login Post Request
m@b.com maulin
$2a$10$nMXcvIAdfHR3SEhk5c4Ge.hd3hXOjVulon7uLSFl4/FZIGl4W4Gq2 maulin
true
password matched
POST /travellogin 200 448.332 ms - -
master*
Ln 107, Col 35
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
8:27 PM
11/4/2018

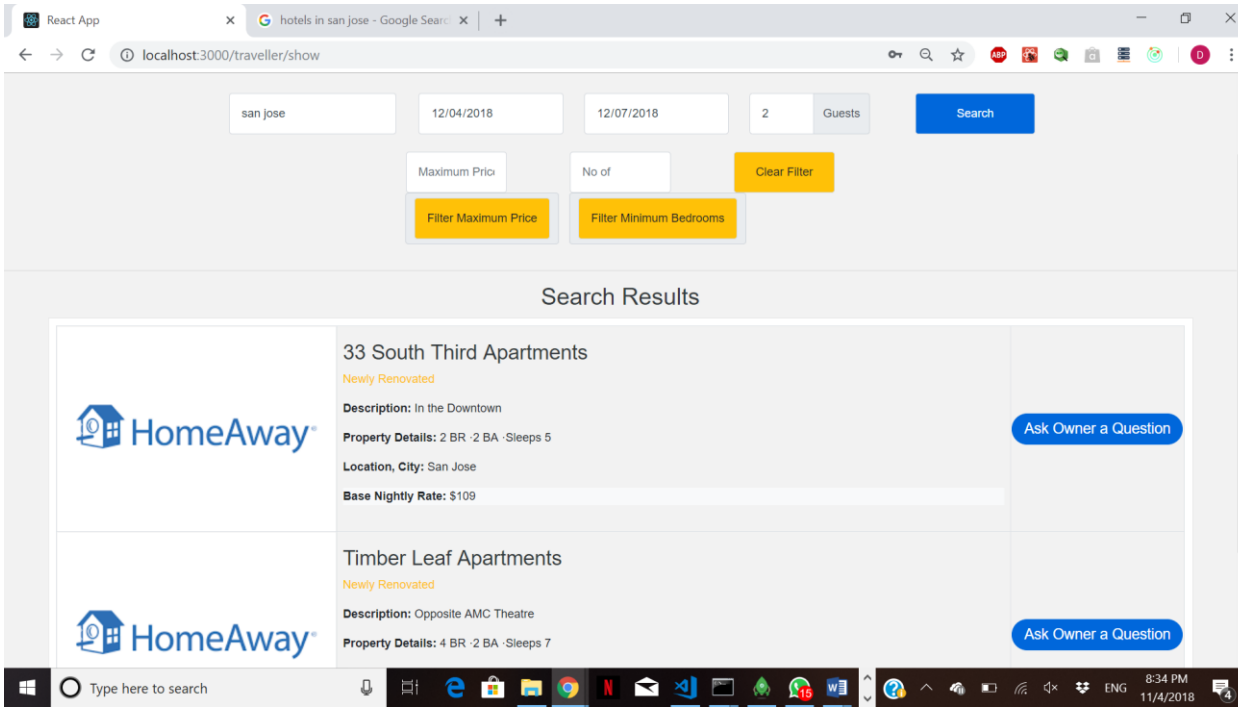


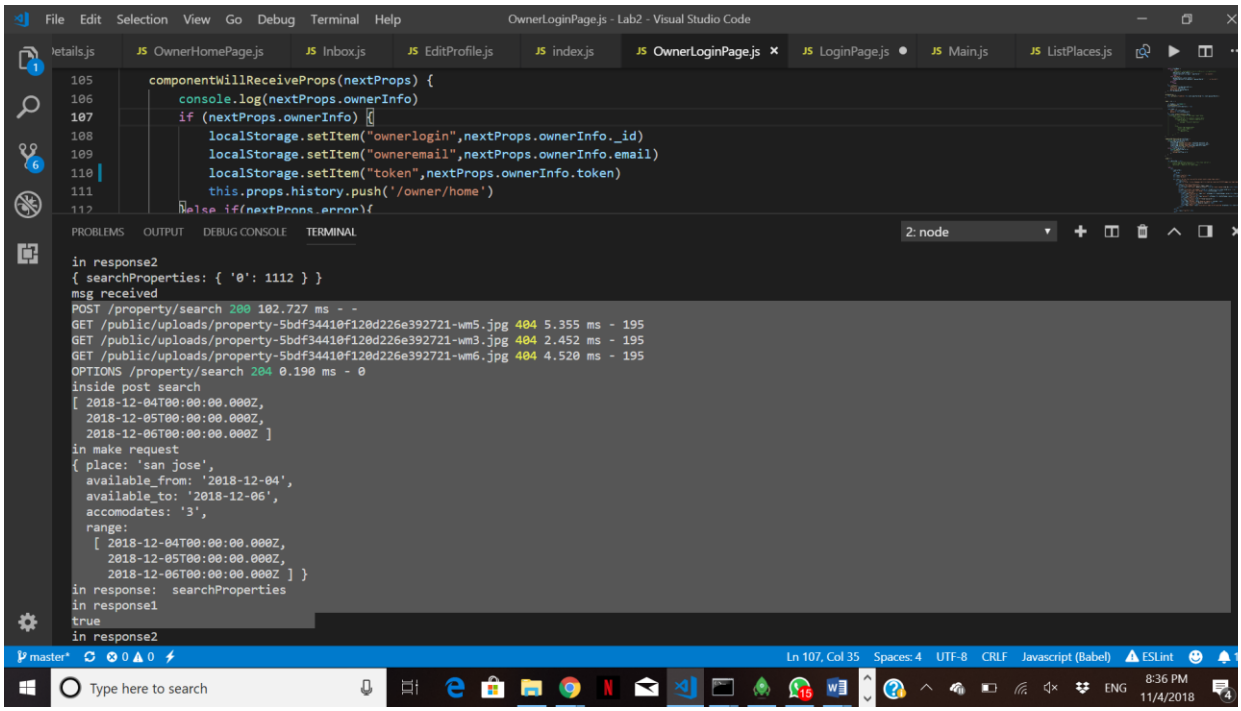
Explanation:
- We have logged in using the login credentials that we just used for signing up.
- The redux store has stored the traveler's information into its store.

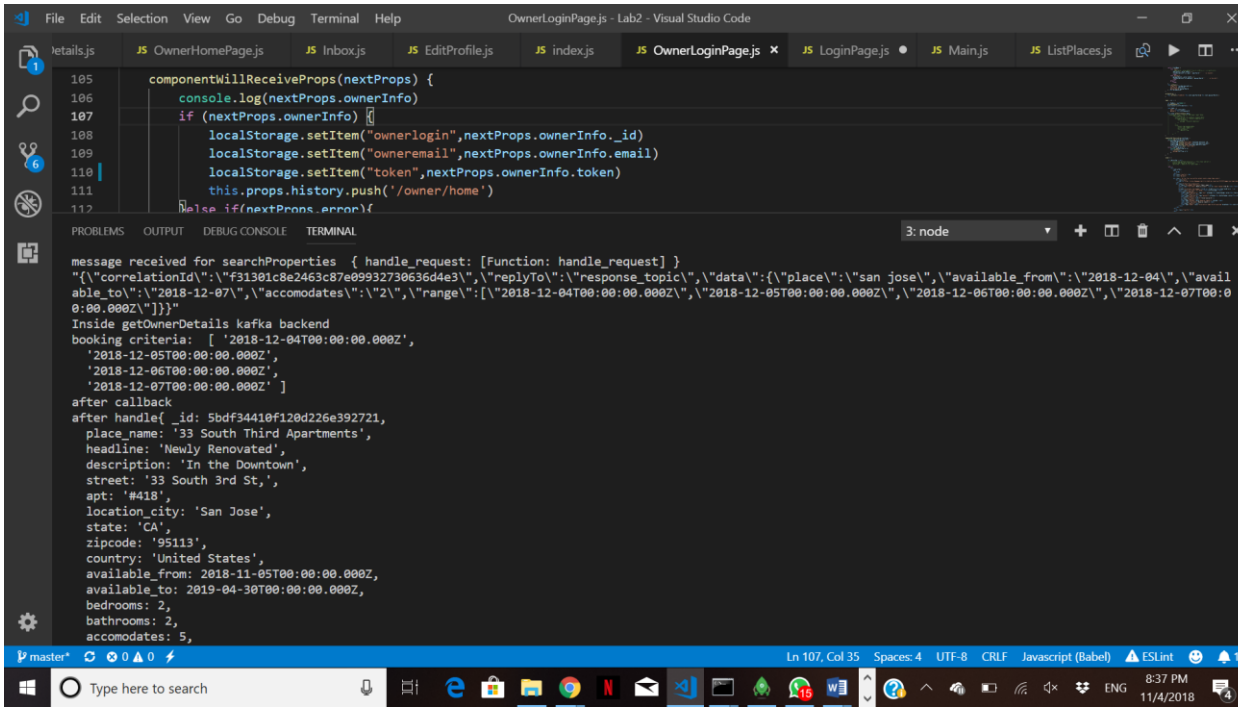
Searching a property based on "city, arrival, departure and # of guests"

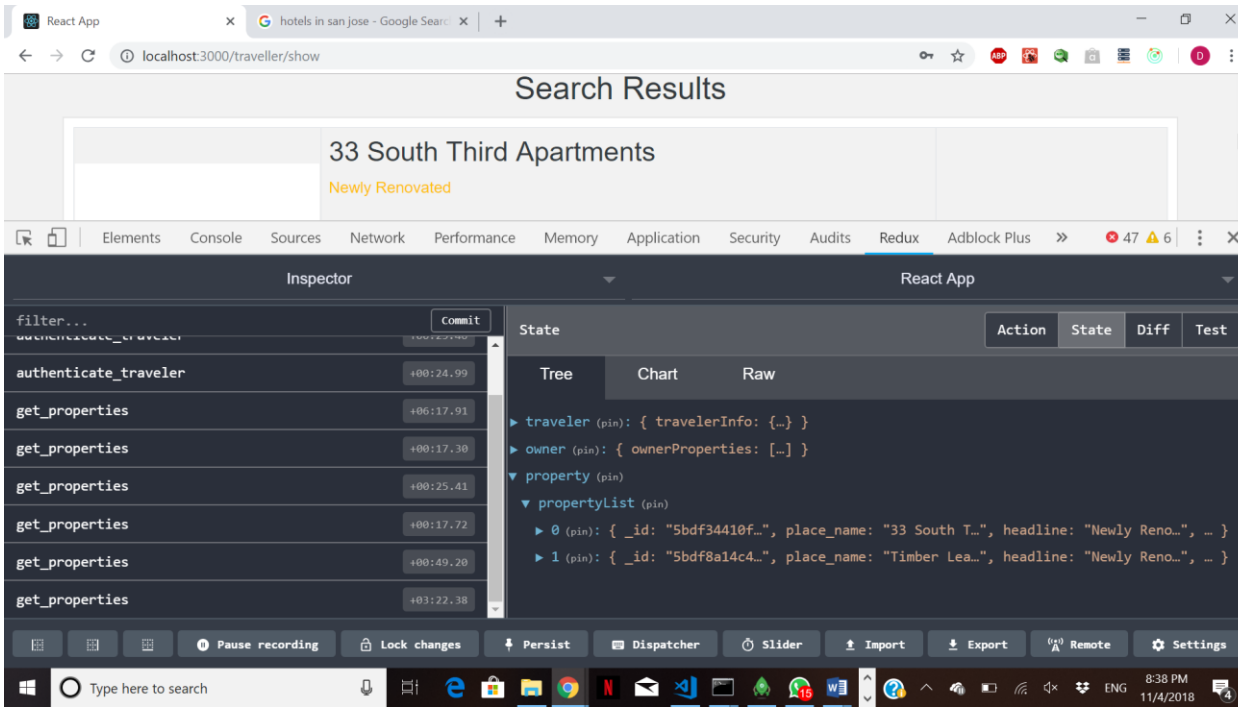



React App
hotels in san jose - Google Searc
localhost:3000/traveller/home
HomeAway
Welcome m@b.com
Help
Book beach houses, cabins,
condos and more, worldwide
san jose
12/04/2018
12/07/2018
2
Guests
Search
Your whole vacation starts here
Choose a rental from the world's best selection
Book and stay with confidence
Your vacation your way
More space, more privacy, no compromises
Trending
Explore these trending places
Type here to search
ENG
8:34 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/traveller/show
san jose
12/04/2018
12/07/2018
2
Guests
Search
Maximum Price
No of
Clear Filter
Filter Maximum Price
Filter Minimum Bedrooms
Search Results
33 South Third Apartments
Newly Renovated
Description: In the Downtown
Property Details: 2 BR ·2 BA ·Sleeps 5
Location, City: San Jose
Base Nightly Rate: $109
Ask Owner a Question
Timber Leaf Apartments
Newly Renovated
Description: Opposite AMC Theatre
Property Details: 4 BR ·2 BA ·Sleeps 7
Ask Owner a Question
Type here to search
ENG
8:34 PM
11/4/2018


File Edit Selection View Go Debug Terminal Help
OwnerLoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
in response2
{ searchProperties: { '0': 1112 } }
msg received
POST /property/search 200 102.727 ms - -
GET /public/uploads/property-5bdf34410f120d226e392721-wm5.jpg 404 5.355 ms - 195
GET /public/uploads/property-5bdf34410f120d226e392721-wm3.jpg 404 2.452 ms - 195
GET /public/uploads/property-5bdf34410f120d226e392721-wm6.jpg 404 4.520 ms - 195
OPTIONS /property/search 204 0.190 ms - 0
inside post search
[2018-12-04T00:00:00.000Z,
2018-12-05T00:00:00.000Z,
2018-12-06T00:00:00.000Z]
in make request
{ place: 'san jose',
available_from: '2018-12-04',
available_to: '2018-12-06',
accomodates: '3',
range:
[2018-12-04T00:00:00.000Z,
2018-12-05T00:00:00.000Z,
2018-12-06T00:00:00.000Z] }
in response: searchProperties
in response1
true
in response2
master*
Ln 107, Col 35
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
8:36 PM
11/4/2018

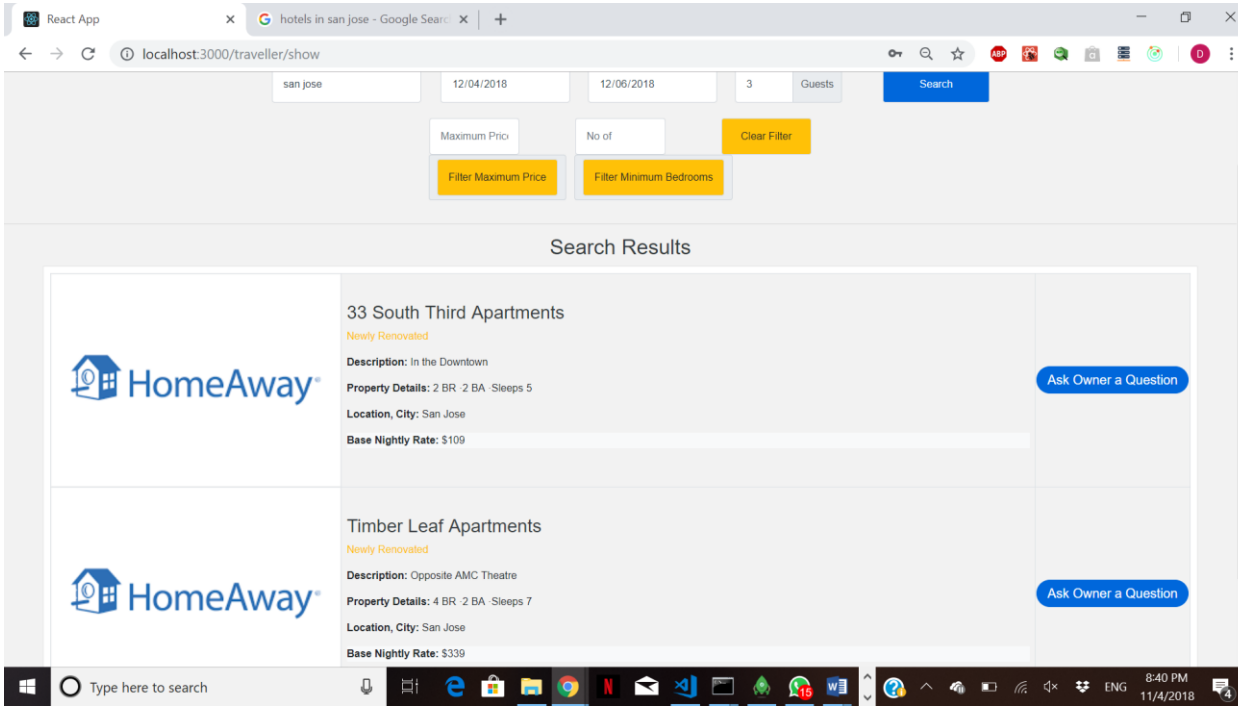

File Edit Selection View Go Debug Terminal Help
OwnerLoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
3: node
message received for searchProperties { handle_request: [Function: handle_request] }
"{\"correlationId\":\"f31301c8e2463c87e09932730636d4e3\",\"replyTo\":\"response_topic\",\"data\":{\"place\":\"san jose\",\"available_from\":\"2018-12-04\",\"available_to\":\"2018-12-07\",\"accomodates\":\"2\",\"range\":[\"2018-12-04T00:00:00.000Z\",\"2018-12-05T00:00:00.000Z\",\"2018-12-06T00:00:00.000Z\",\"2018-12-07T00:00:00.000Z\"]}}"
Inside getOwnerDetails kafka backend
booking criteria: ['2018-12-04T00:00:00.000Z',
'2018-12-05T00:00:00.000Z',
'2018-12-06T00:00:00.000Z',
'2018-12-07T00:00:00.000Z']
after callback
after handle{ _id: 5bdf34410f120d226e392721,
place_name: '33 South Third Apartments',
headline: 'Newly Renovated',
description: 'In the Downtown',
street: '33 South 3rd St,',
apt: '#418',
location_city: 'San Jose',
state: 'CA',
zipcode: '95113',
country: 'United States',
available_from: 2018-11-05T00:00:00.000Z,
available_to: 2019-04-30T00:00:00.000Z,
bedrooms: 2,
bathrooms: 2,
accomodates: 5,
master*
Ln 107, Col 35
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
8:37 PM
11/4/2018

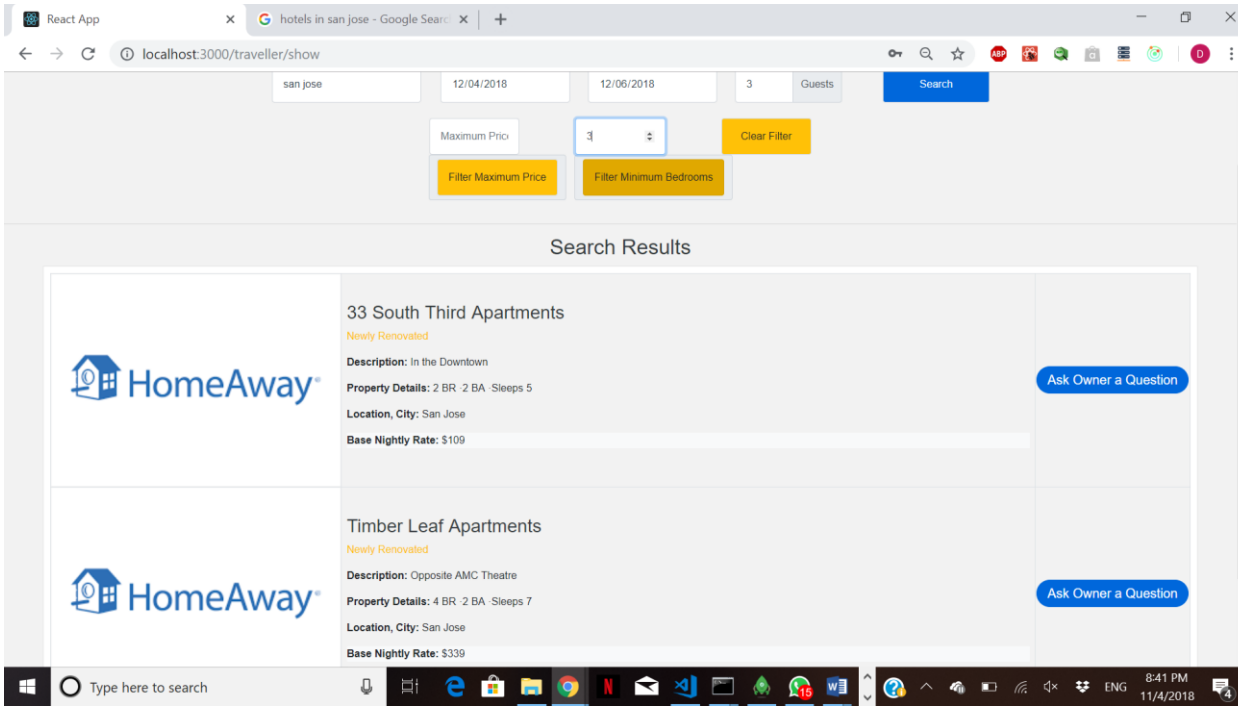


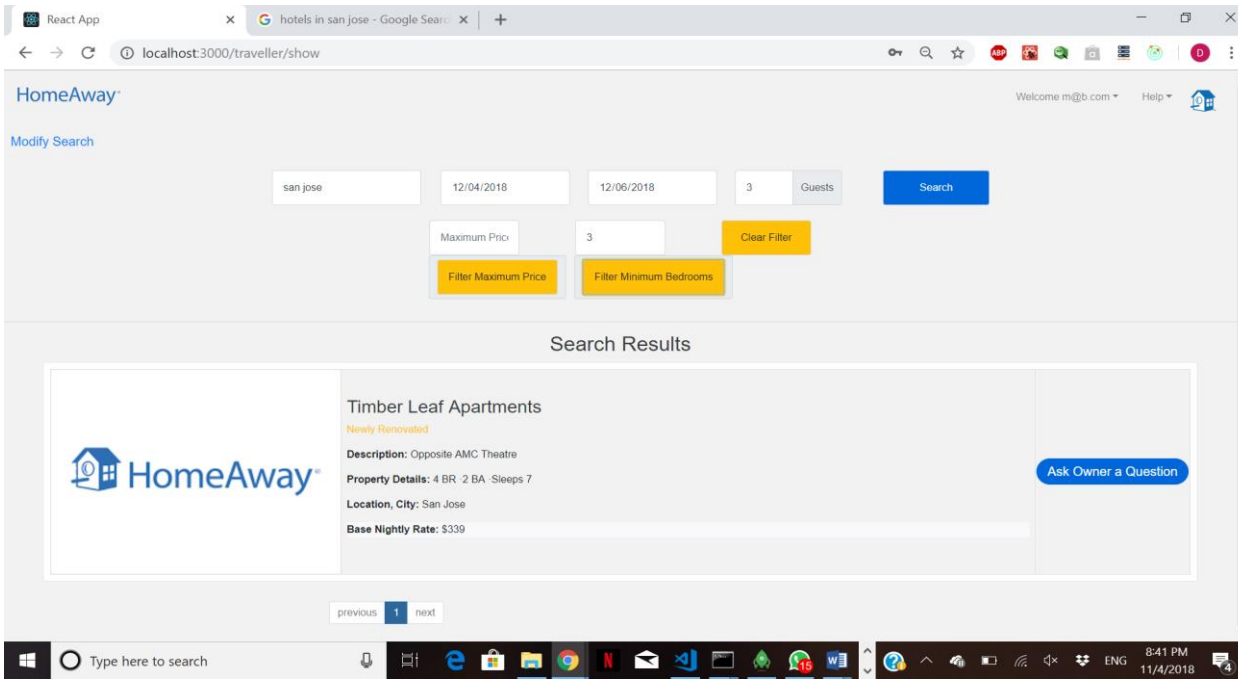
Explanation:

- As you can see in the first screenshot, because we haven't entered all data into the form, the search button is disabled and will not let us access the search button.
- After we provide all the details(as done in the 2nd screenshot), the search button will get enabled and we will get our results.
- The console output shows the topic which is being used to search the properties, as, we have to take care of all the details provided and plus, we have to make sure that the dates provided are not booked already.
- After we click on search, we will be directed to a page which will show the search results. But, notice how the inline form has the search parameters intact even after we have directed to the next page. This was persisted using redux store.
- The redux store also has stored the properties that we just searched.

Displaying the result of the above search and filtering through it



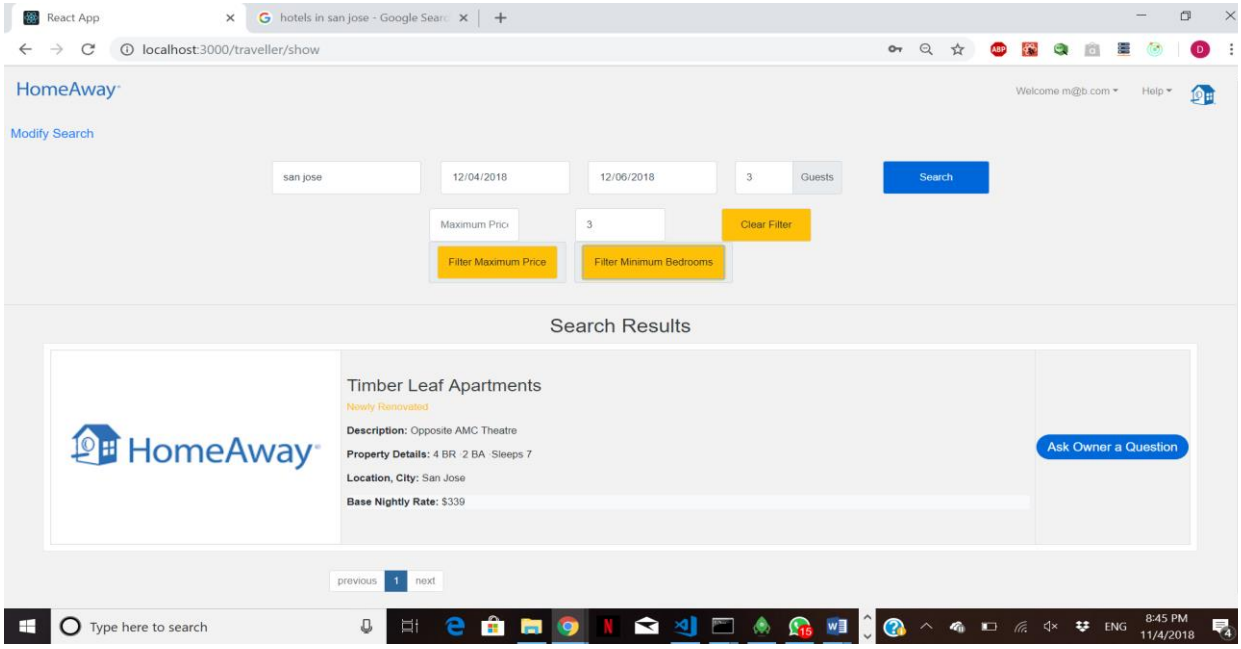


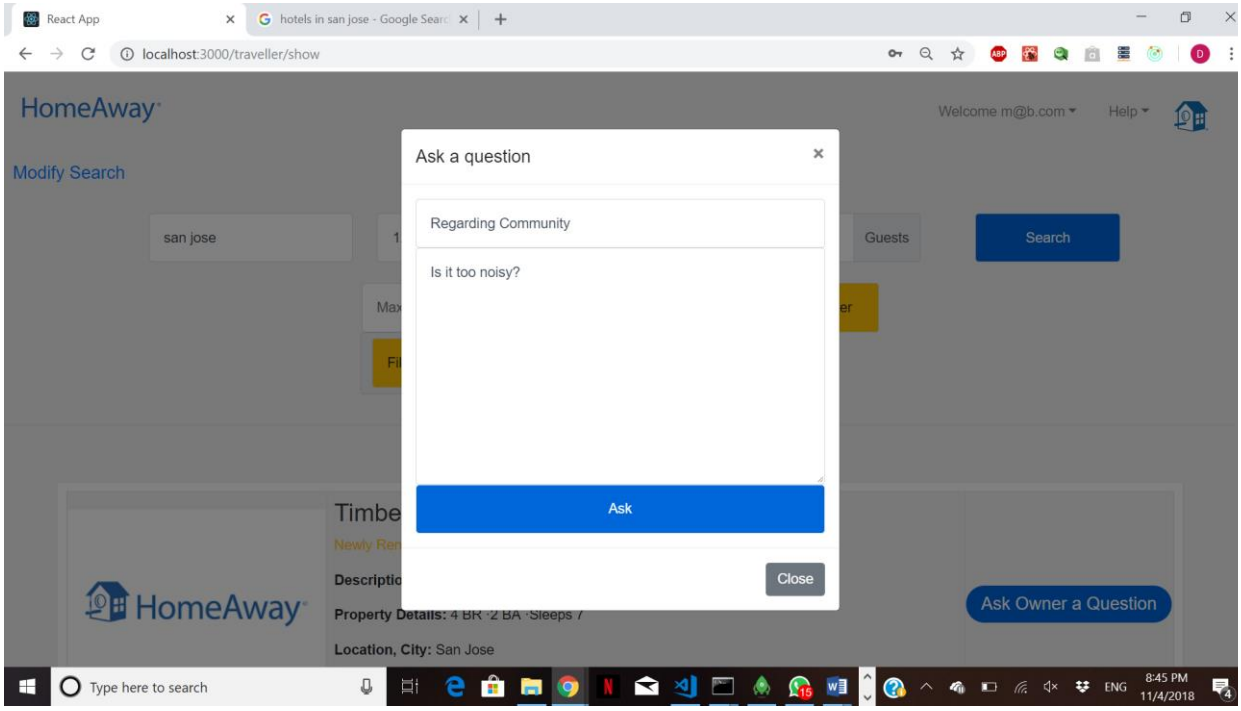


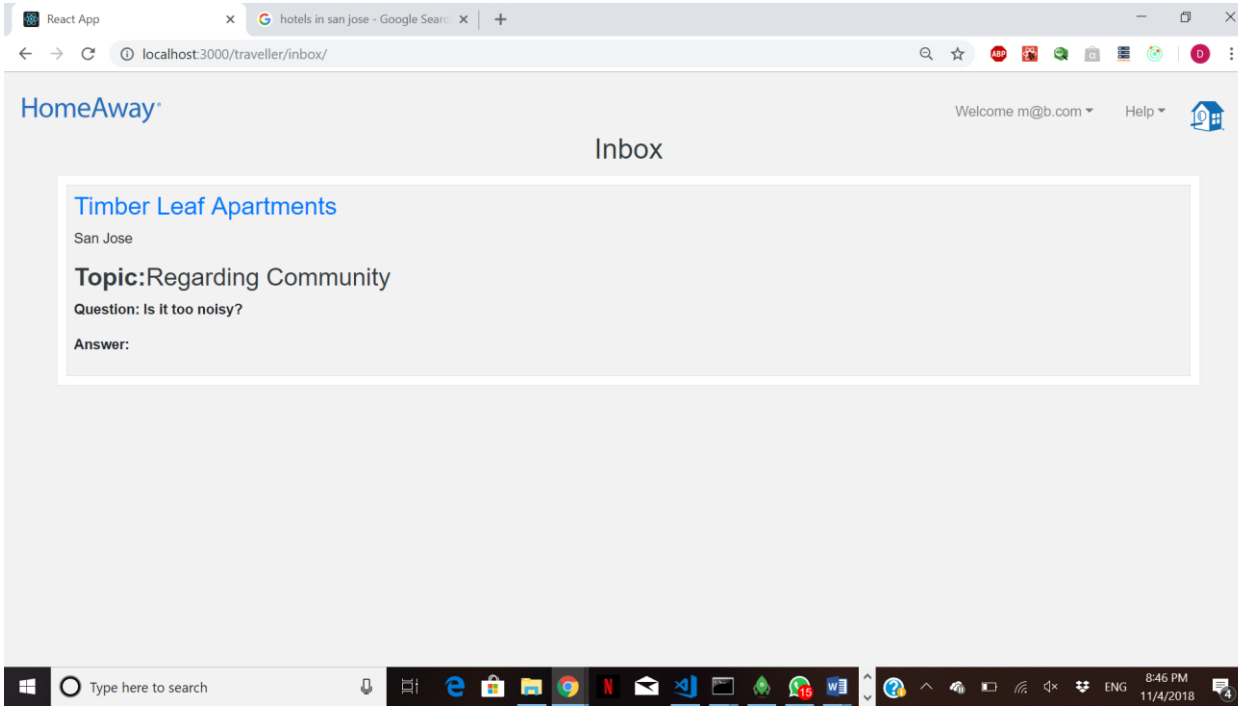
Explanation:

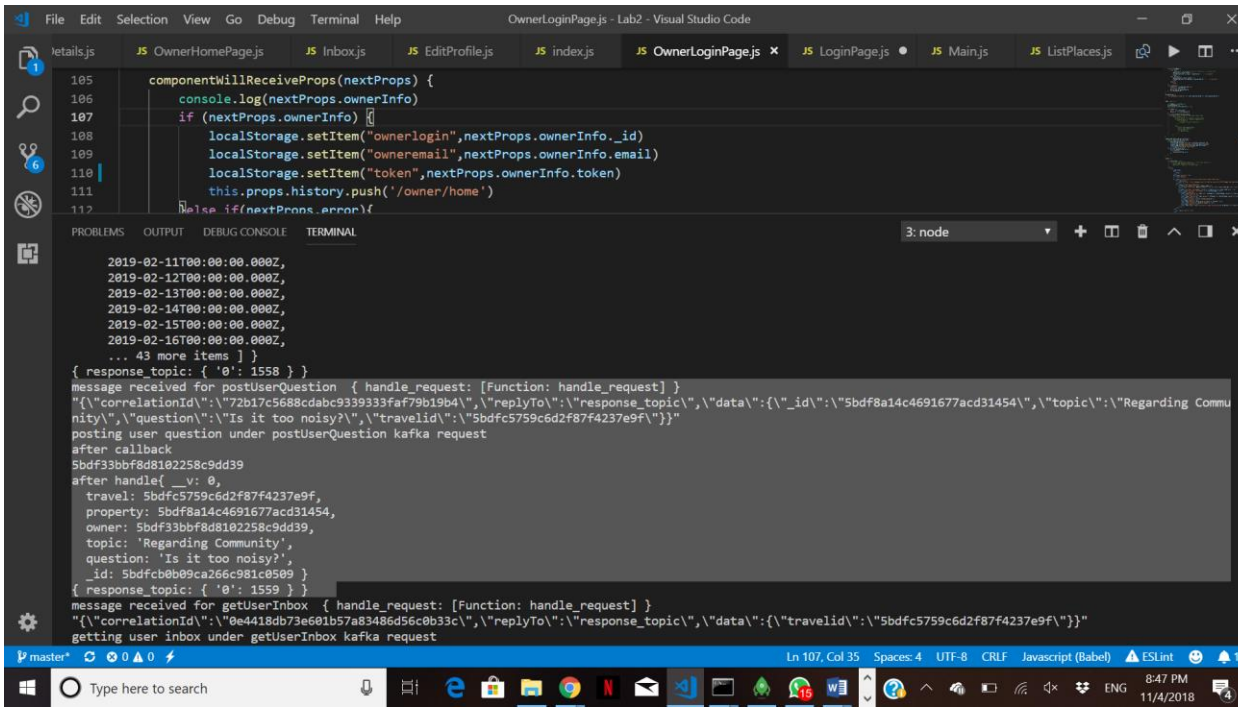
- Here, we are filtering the results to get all the properties which has more than 3 bedrooms.
- As shown in the screenshot, the only property in the searched list having more than 3 bedrooms is Timber Leaf Apartments and thus it is returned.
- The filtering is done by '_' operator imported from lodash library.

Asking Owner the question on any of the property searched




React App
hotels in san jose - Google Searc
localhost:3000/traveller/show
HomeAway
Welcome m@b.com
Help
Modify Search
san jose
Guests
Search
Ask a question
Regarding Community
Is it too noisy?
Ask
Close
Timbe
Newly Ren
Descriptio
Property Details: 4 BR 2 BA Sleeps 7
Location, City: San Jose
HomeAway
Ask Owner a Question
Type here to search
ENG
8:45 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/traveller/inbox/
HomeAway
Welcome m@b.com
Help
Inbox
Timber Leaf Apartments
San Jose
Topic:Regarding Community
Question: Is it too noisy?
Answer:
Type here to search
ENG
8:46 PM
11/4/2018

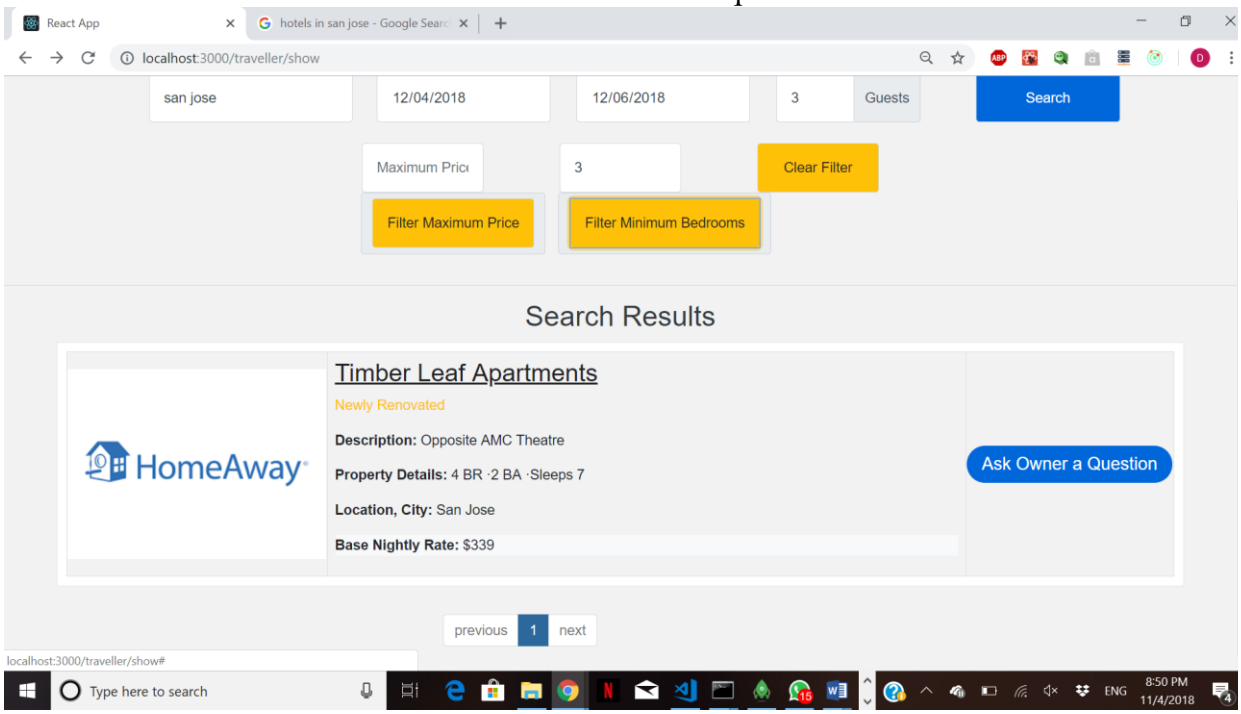


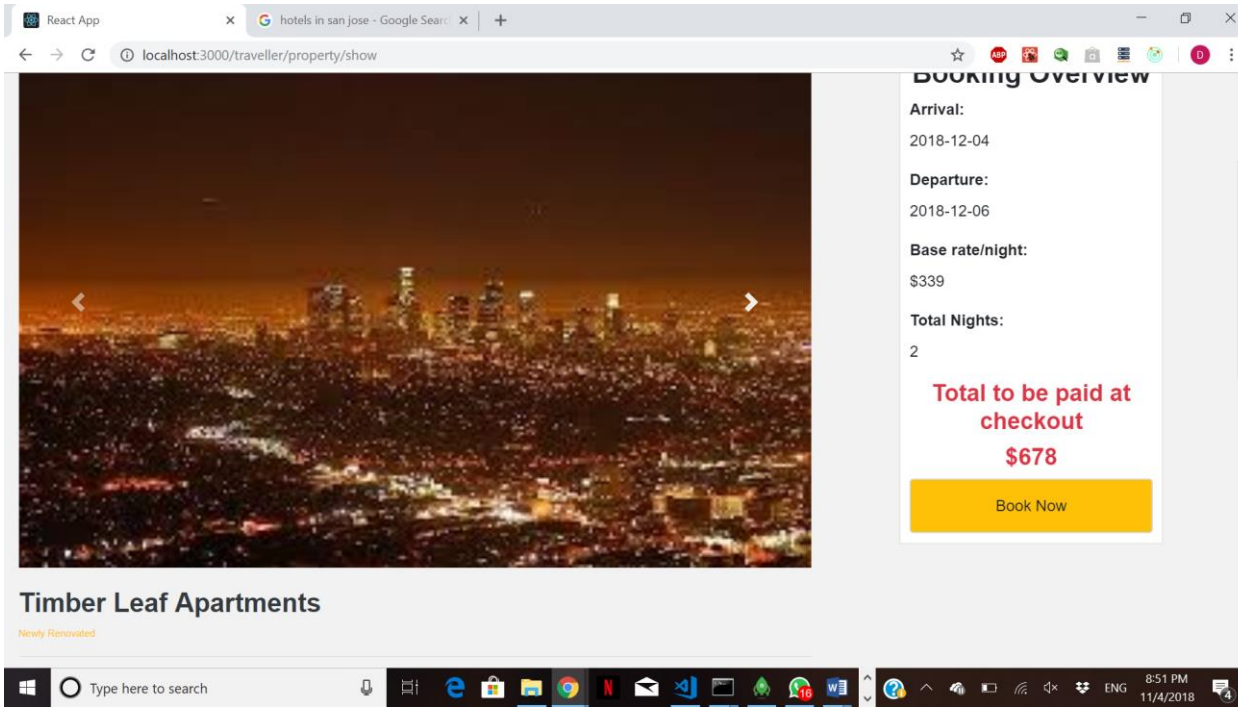
Explanation:

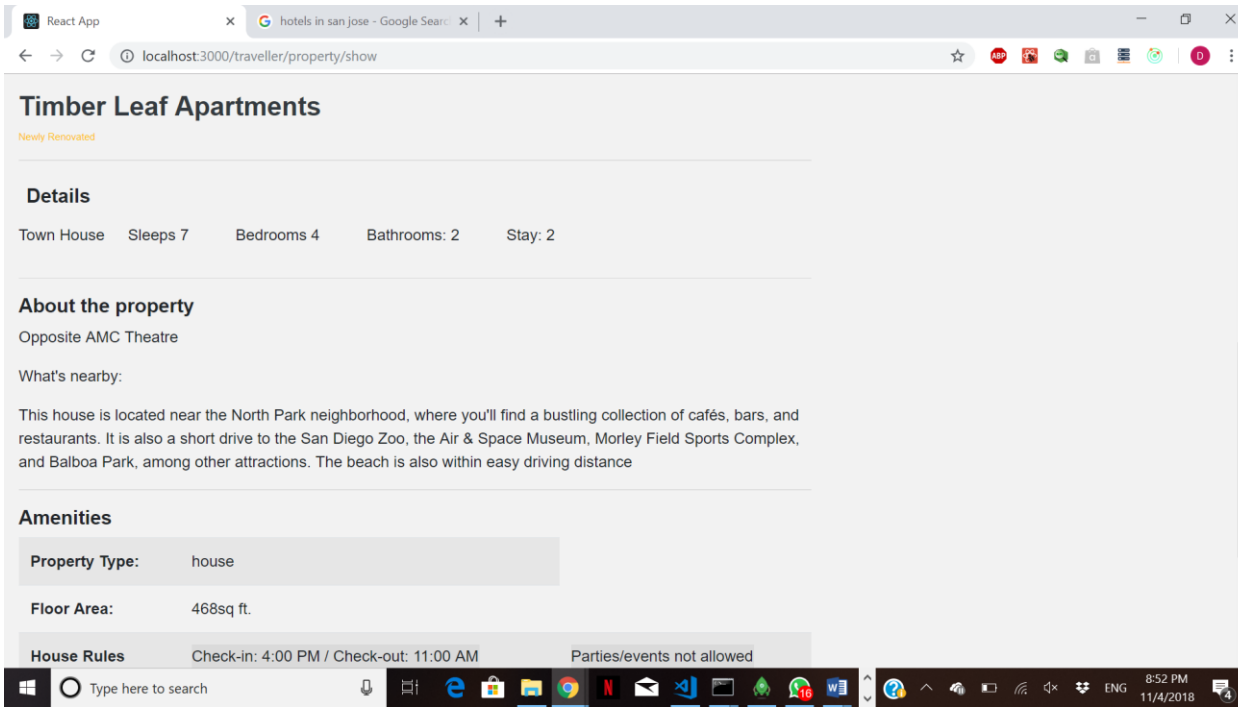
- As shown in the screenshot, we asked owner the question about the community.
- The inbox of the traveler shows that the question has been posted.
- The answer is not there, as for that the owner will have to reply first. We'll do that later.
- The redux store is not storing the questions as those aren't needed to pass to any other component.

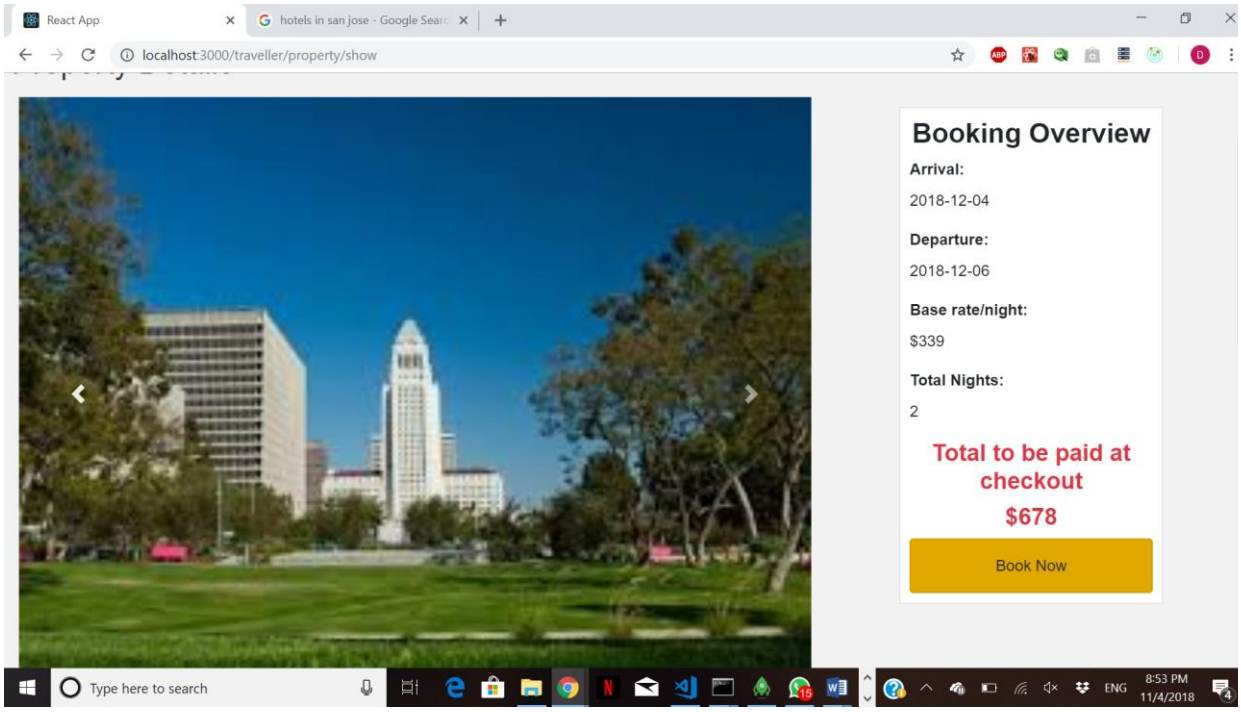
Selecting a property from search result and booking it with the travel account

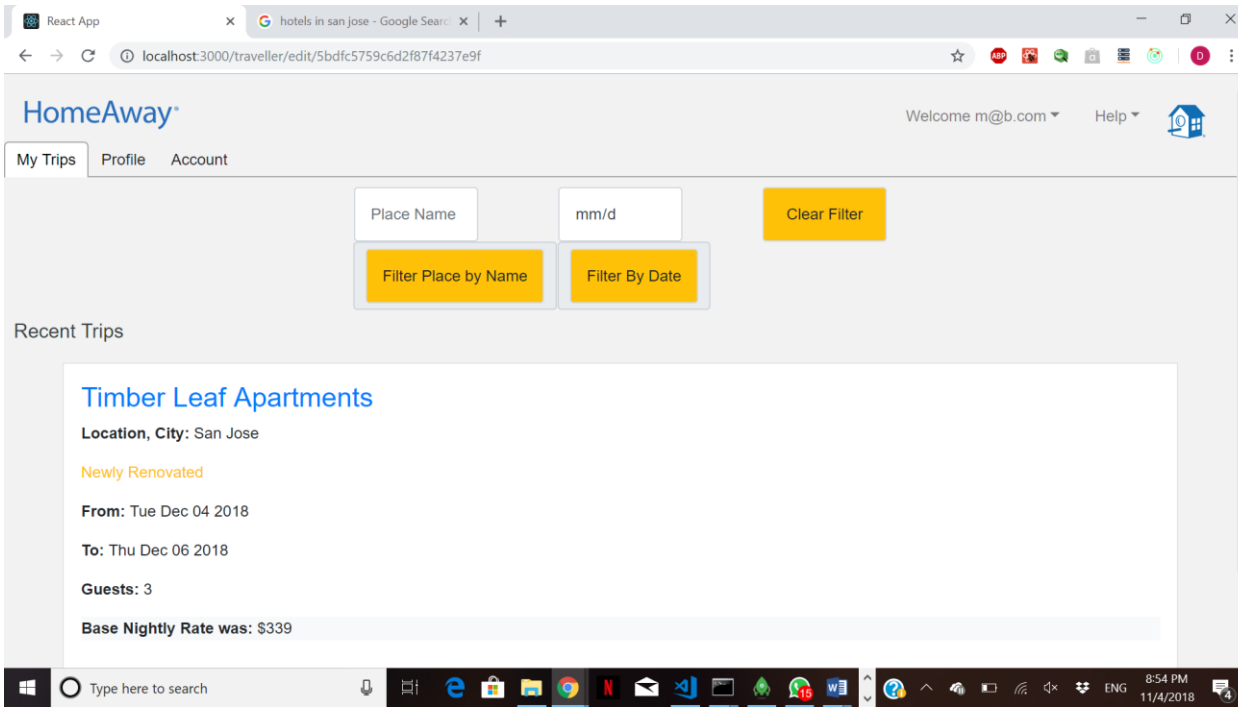
Let's book Timber-Leaf Apartments

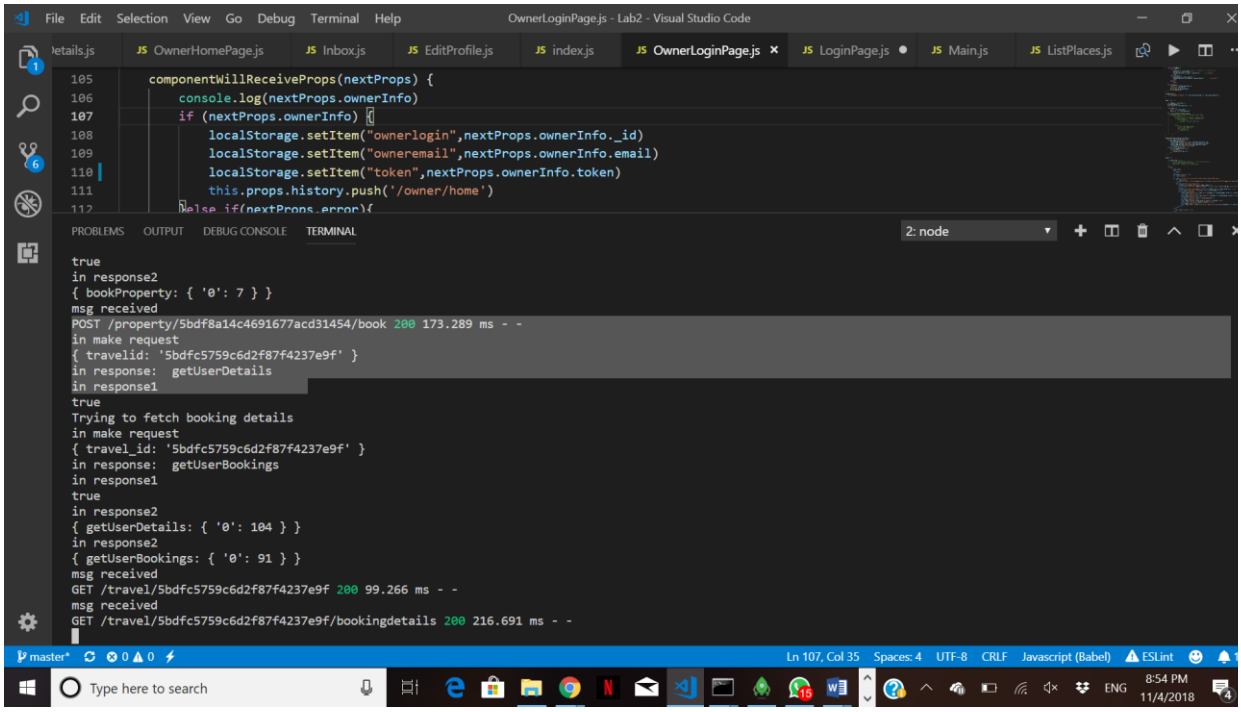


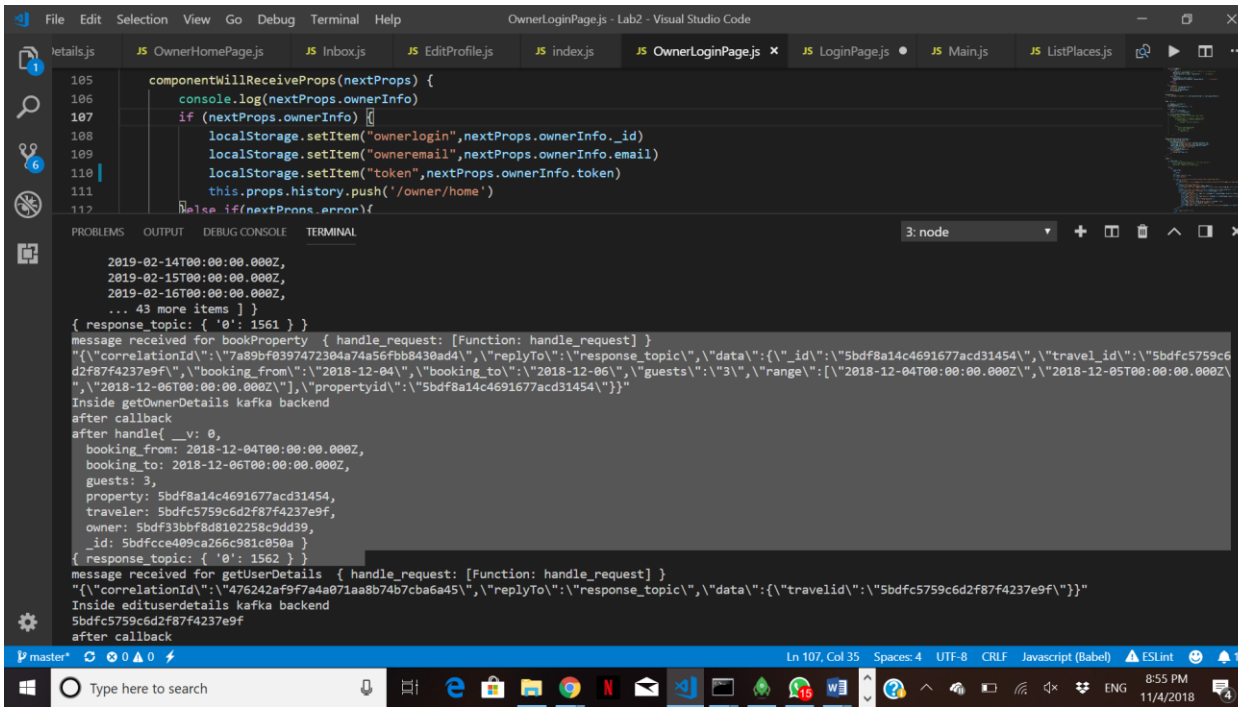


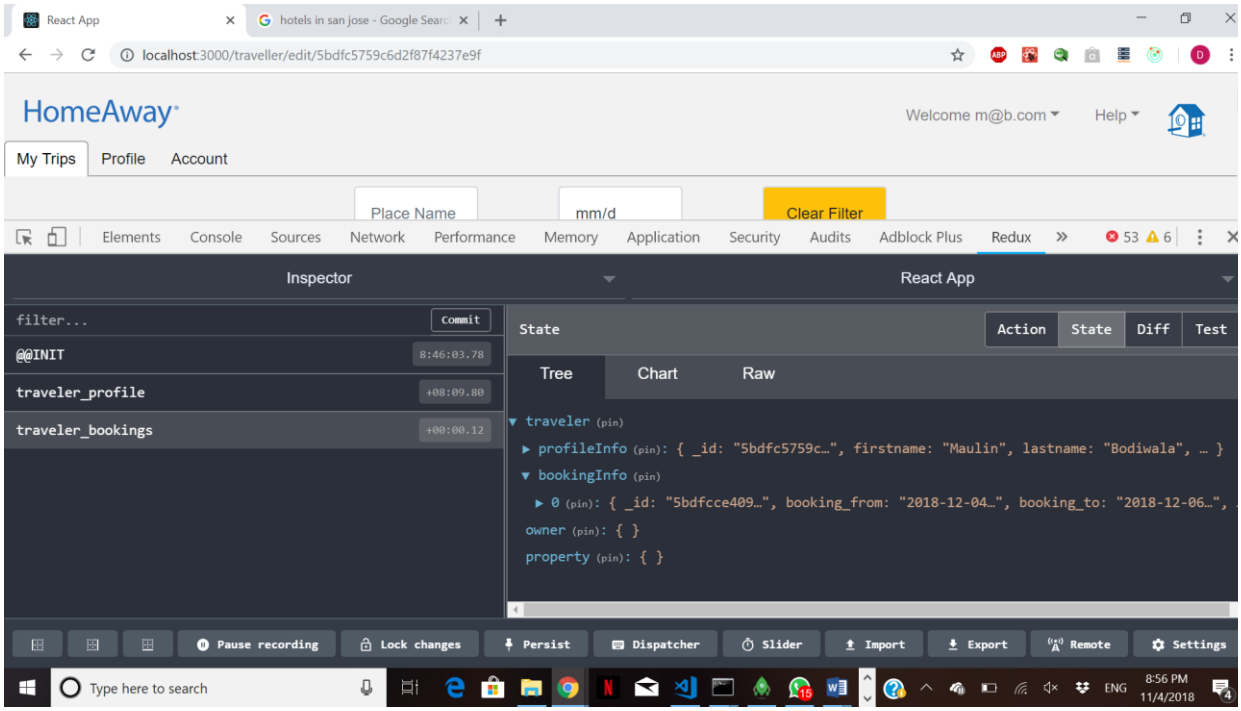

React App
hotels in san jose - Google Searc
localhost:3000/traveller/property/show
Timber Leaf Apartments
Newly Renovated
Details
Town House Sleeps 7 Bedrooms 4 Bathrooms: 2 Stay: 2
About the property
Opposite AMC Theatre
What's nearby:
This house is located near the North Park neighborhood, where you'll find a bustling collection of cafés, bars, and restaurants. It is also a short drive to the San Diego Zoo, the Air & Space Museum, Morley Field Sports Complex, and Balboa Park, among other attractions. The beach is also within easy driving distance
Amenities
Property Type: house
Floor Area: 468sq ft.
House Rules Check-in: 4:00 PM / Check-out: 11:00 AM Parties/events not allowed
Type here to search
ENG
8:52 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/traveller/property/show
Booking Overview
Arrival:
2018-12-04
Departure:
2018-12-06
Base rate/night:
$339
Total Nights:
2
Total to be paid at checkout
$678
Book Now
Type here to search
ENG
8:53 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/traveller/edit/5bdfc5759c6d2f87f4237e9f
HomeAway
Welcome m@b.com
Help
My Trips
Profile
Account
Place Name
mm/d
Clear Filter
Filter Place by Name
Filter By Date
Recent Trips
Timber Leaf Apartments
Location, City: San Jose
Newly Renovated
From: Tue Dec 04 2018
To: Thu Dec 06 2018
Guests: 3
Base Nightly Rate was: $339
Type here to search
ENG
8:54 PM
11/4/2018


File Edit Selection View Go Debug Terminal Help
OwnerLoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
true
in response2
{ bookProperty: { '0': 7 } }
msg received
POST /property/5bdf8a14c4691677acd31454/book 200 173.289 ms - -
in make request
{ travelid: '5bdfc5759c6d2f87f4237e9f' }
in response: getUserDetails
in response1
true
Trying to fetch booking details
in make request
{ travel_id: '5bdfc5759c6d2f87f4237e9f' }
in response: getUserBookings
in response1
true
in response2
{ getUserDetails: { '0': 104 } }
in response2
{ getUserBookings: { '0': 91 } }
msg received
GET /travel/5bdfc5759c6d2f87f4237e9f 200 99.266 ms - -
msg received
GET /travel/5bdfc5759c6d2f87f4237e9f/bookingdetails 200 216.691 ms - -
master*
Ln 107, Col 35
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
8:54 PM
11/4/2018

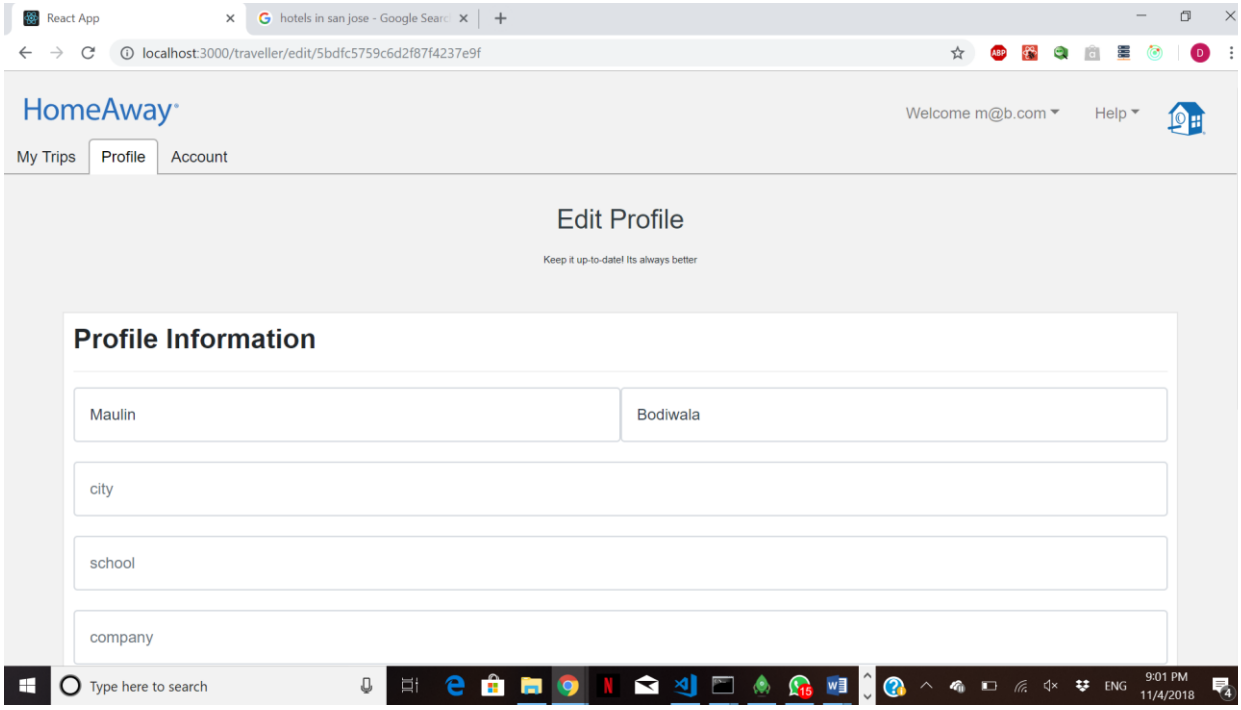


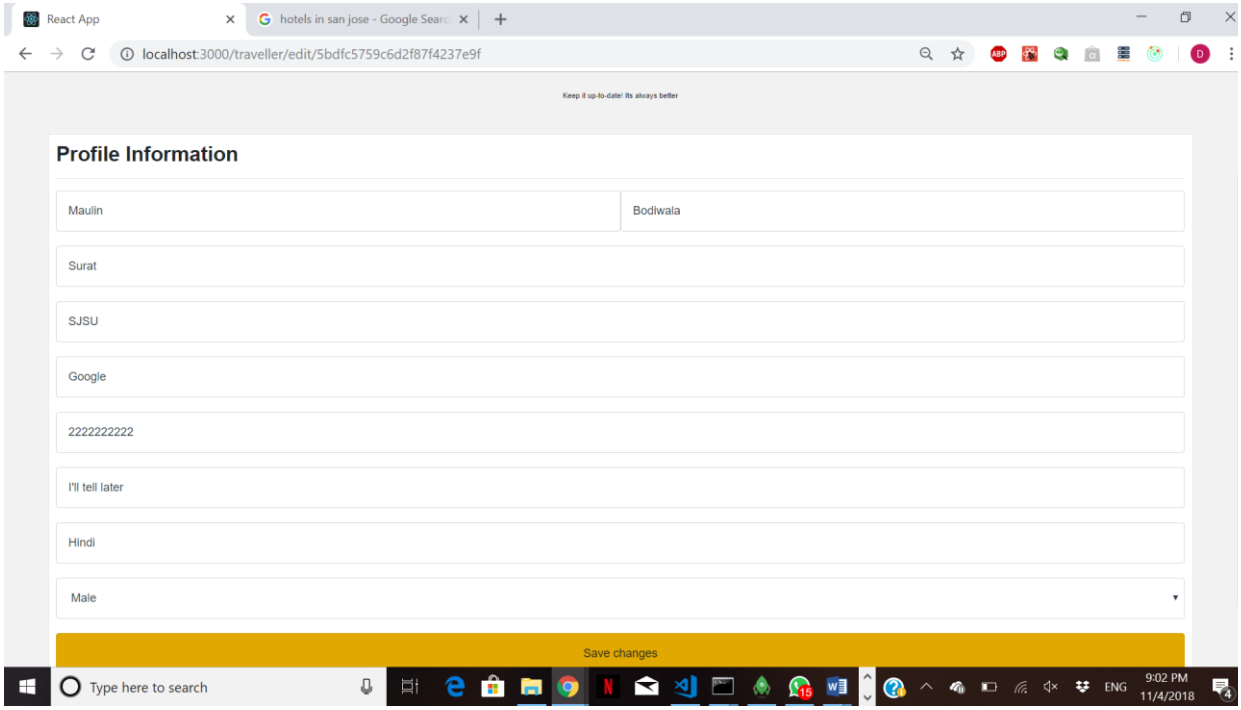


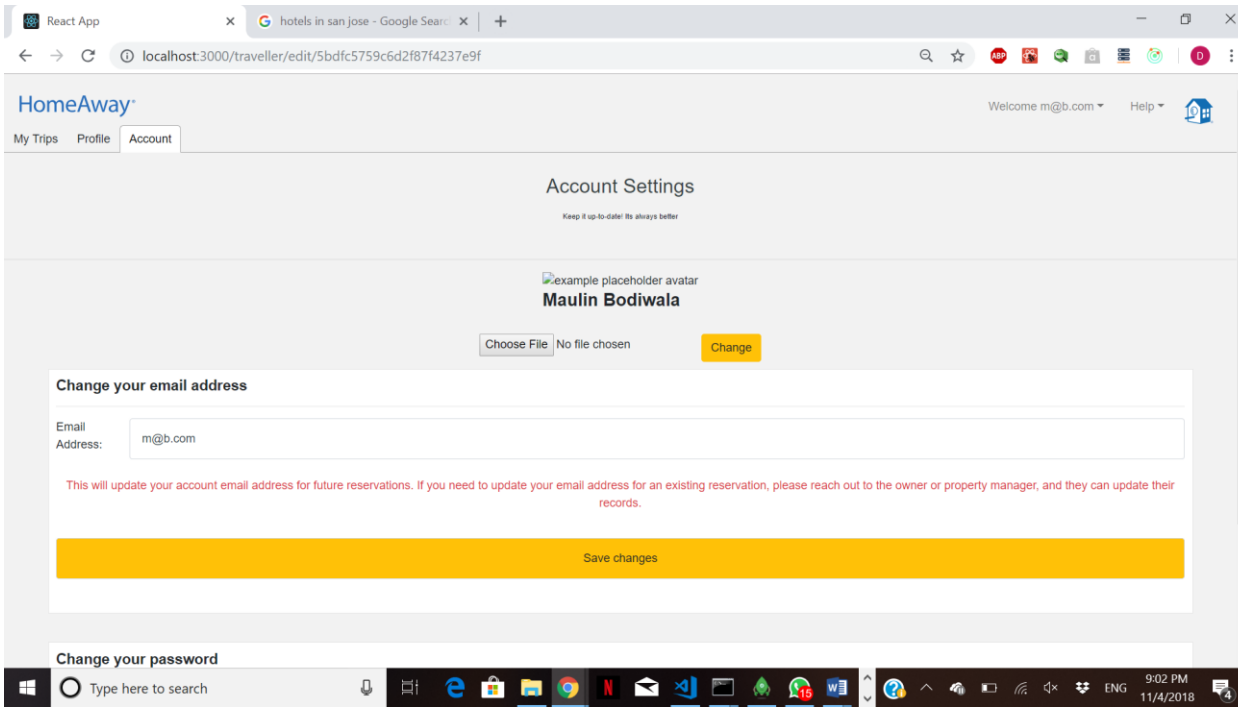
Explanation:

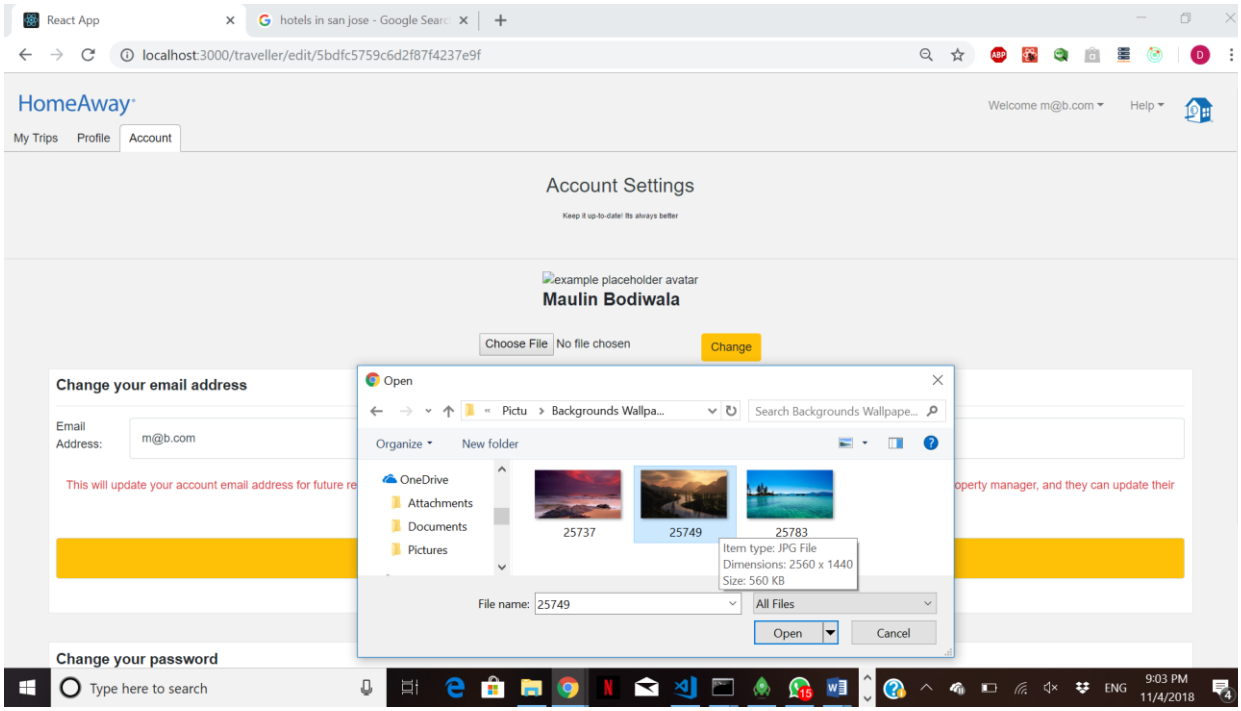
- The above screenshots show the details of the property that we just selected i.e ”Timber Leaf Apartments.”
- On the side, we can re-confirm our booking details and after clicking on book, we book that property for that date and for that user.
- The console output and the redux store will certify that thing.

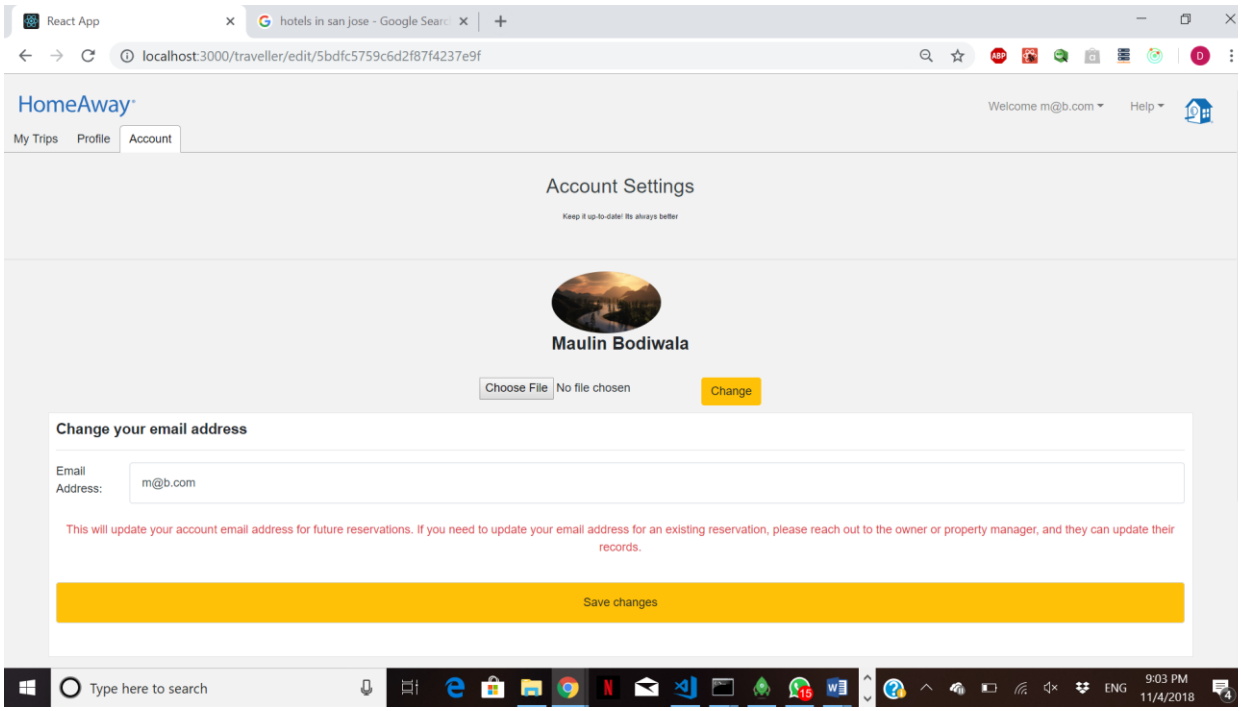
Editing the basic details of travel account

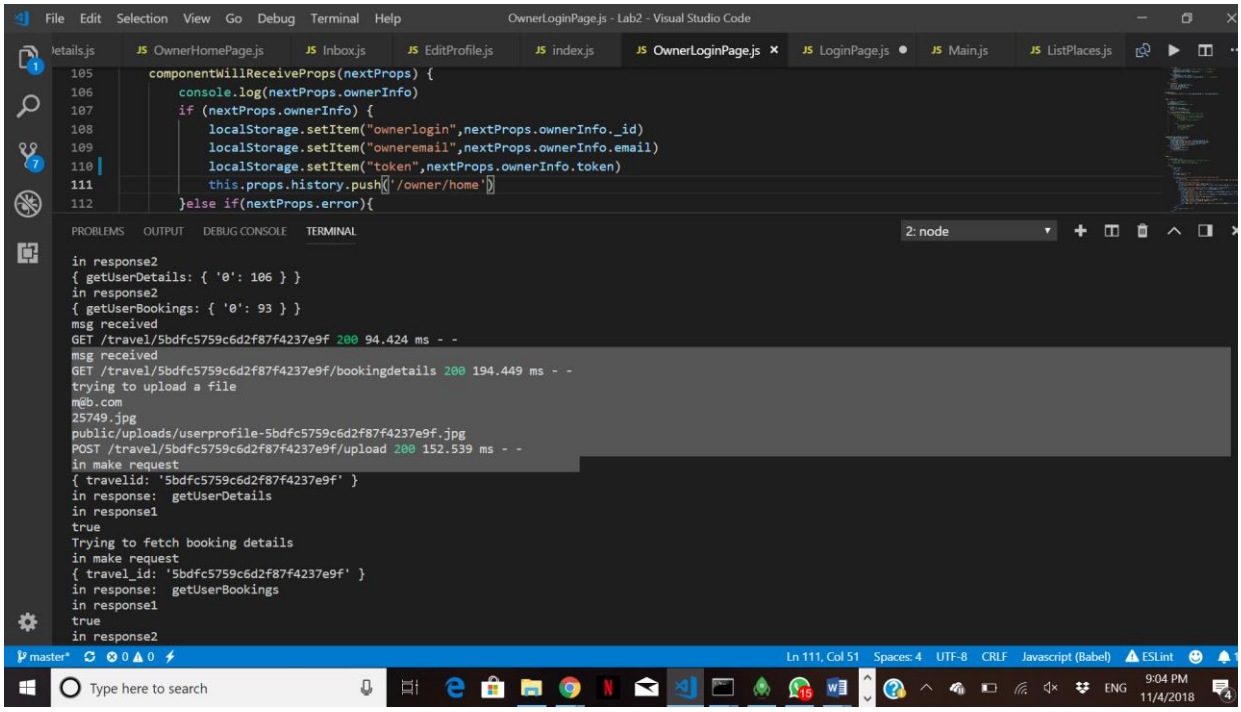


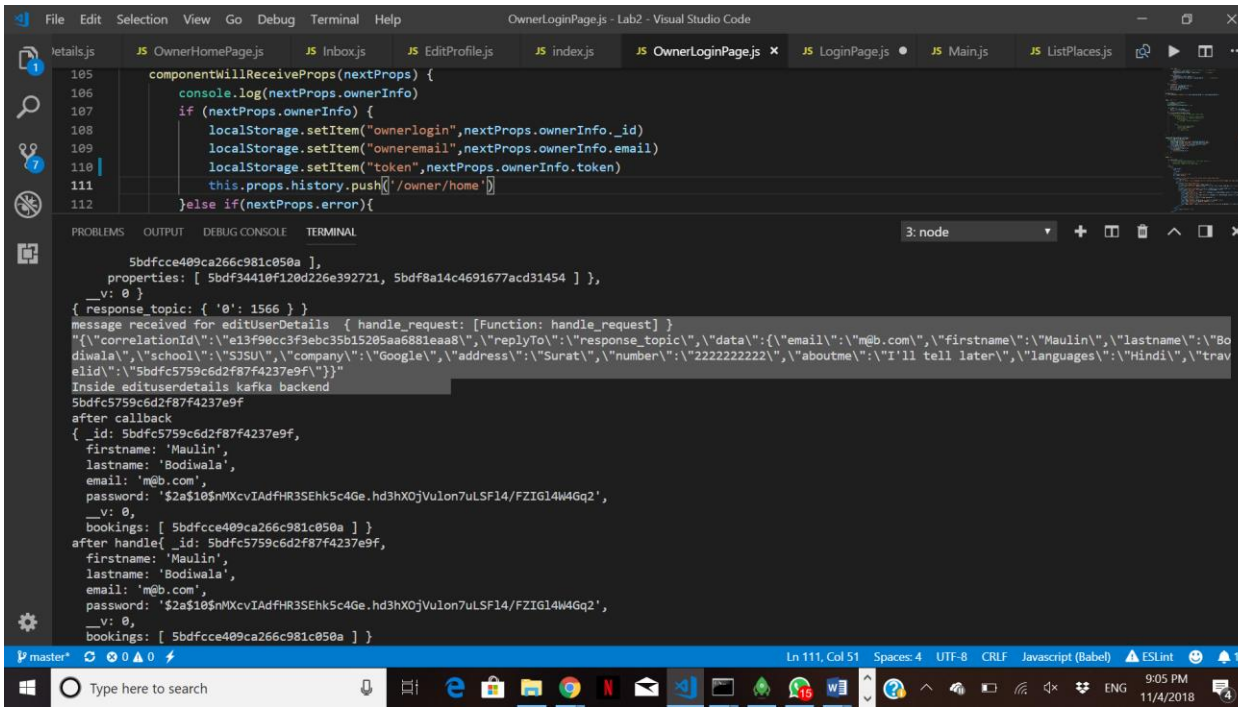


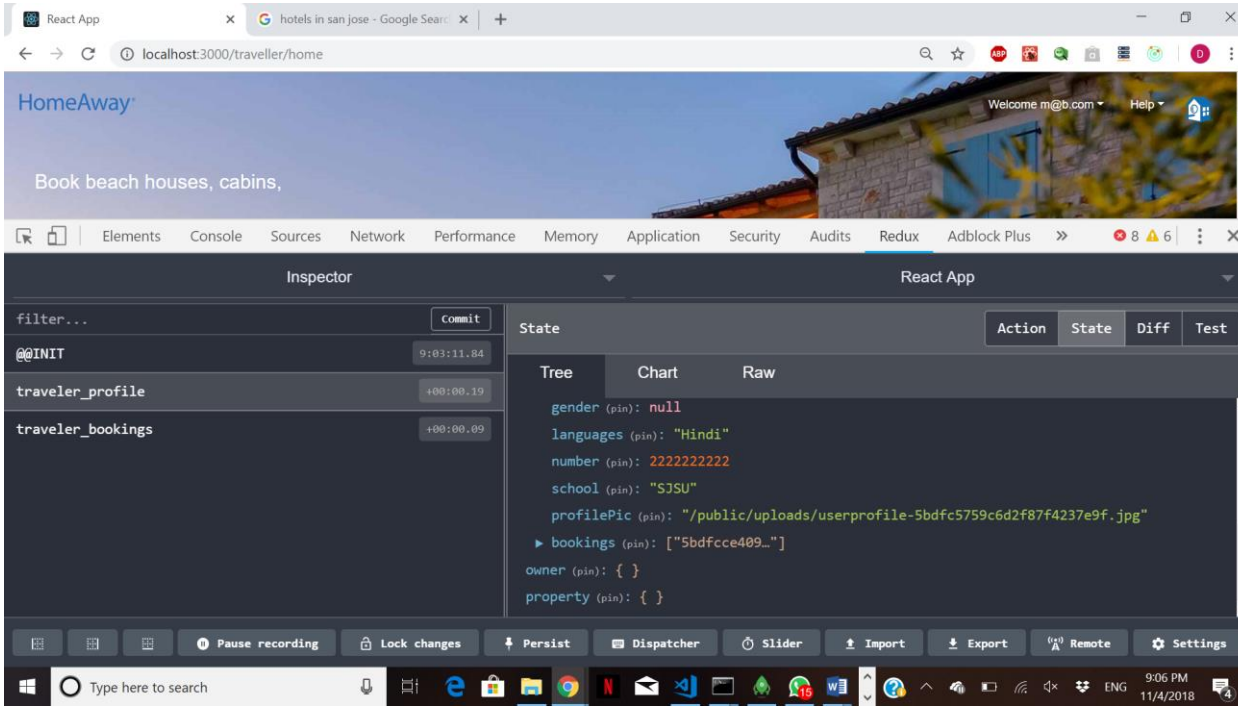

React App
hotels in san jose - Google Searc
localhost:3000/traveller/edit/5bdfc5759c6d2f87f4237e9f
HomeAway
Welcome m@b.com
Help
My Trips
Profile
Account
Account Settings
Keep it up-to-date! Its always better
example placeholder avatar
Maulin Bodiwala
Choose File
No file chosen
Change
Change your email address
Email Address:
m@b.com
This will update your account email address for future reservations. If you need to update your email address for an existing reservation, please reach out to the owner or property manager, and they can update their records.
Save changes
Change your password
Type here to search
ENG
9:02 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/traveller/edit/5bdfc5759c6d2f87f4237e9f
HomeAway
Welcome m@b.com
Help
My Trips
Profile
Account
Account Settings
Keep it up-to-date! Its always better
example placeholder avatar
Maulin Bodiwala
Choose File
No file chosen
Change
Change your email address
Email Address:
m@b.com
Open
Pictu
Backgrounds Wallpa...
Search Backgrounds Wallpape...
Organize
New folder
OneDrive
Attachments
Documents
Pictures
25737
25749
25783
Item type: JPG File
Dimensions: 2560 x 1440
Size: 560 KB
File name:
25749
All Files
Open
Cancel
Change your password
Type here to search
ENG
9:03 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/traveller/edit/5bdfc5759c6d2f87f4237e9f
HomeAway
Welcome m@b.com
Help
My Trips
Profile
Account
Account Settings
Keep it up-to-date! Its always better
Maulin Bodiwala
Choose File
No file chosen
Change
Change your email address
Email Address:
m@b.com
This will update your account email address for future reservations. If you need to update your email address for an existing reservation, please reach out to the owner or property manager, and they can update their records.
Save changes
Type here to search
ENG
9:03 PM
11/4/2018


File Edit Selection View Go Debug Terminal Help
OwnerLoginPage.js - Lab2 - Visual Studio Code
OwnerHomePage.js
Inbox.js
EditProfile.js
index.js
OwnerLoginPage.js
LoginPage.js
Main.js
ListPlaces.js
componentWillReceiveProps(nextProps) {
console.log(nextProps.ownerInfo)
if (nextProps.ownerInfo) {
localStorage.setItem("ownerlogin",nextProps.ownerInfo._id)
localStorage.setItem("owneremail",nextProps.ownerInfo.email)
localStorage.setItem("token",nextProps.ownerInfo.token)
this.props.history.push('/owner/home')
}else if(nextProps.error){
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: node
in response2
{ getUserDetails: { '0': 106 } }
in response2
{ getUserBookings: { '0': 93 } }
msg received
GET /travel/5bdfc5759c6d2f87f4237e9f 200 94.424 ms - -
msg received
GET /travel/5bdfc5759c6d2f87f4237e9f/bookingdetails 200 194.449 ms - -
trying to upload a file
m@b.com
25749.jpg
public/uploads/userprofile-5bdfc5759c6d2f87f4237e9f.jpg
POST /travel/5bdfc5759c6d2f87f4237e9f/upload 200 152.539 ms - -
in make request
{ travelid: '5bdfc5759c6d2f87f4237e9f' }
in response: getUserDetails
in response1
true
Trying to fetch booking details
in make request
{ travel_id: '5bdfc5759c6d2f87f4237e9f' }
in response: getUserBookings
in response1
true
in response2
master*
Ln 111, Col 51
Spaces: 4
UTF-8
CRLF
Javascript (Babel)
ESLint
Type here to search
ENG
9:04 PM
11/4/2018



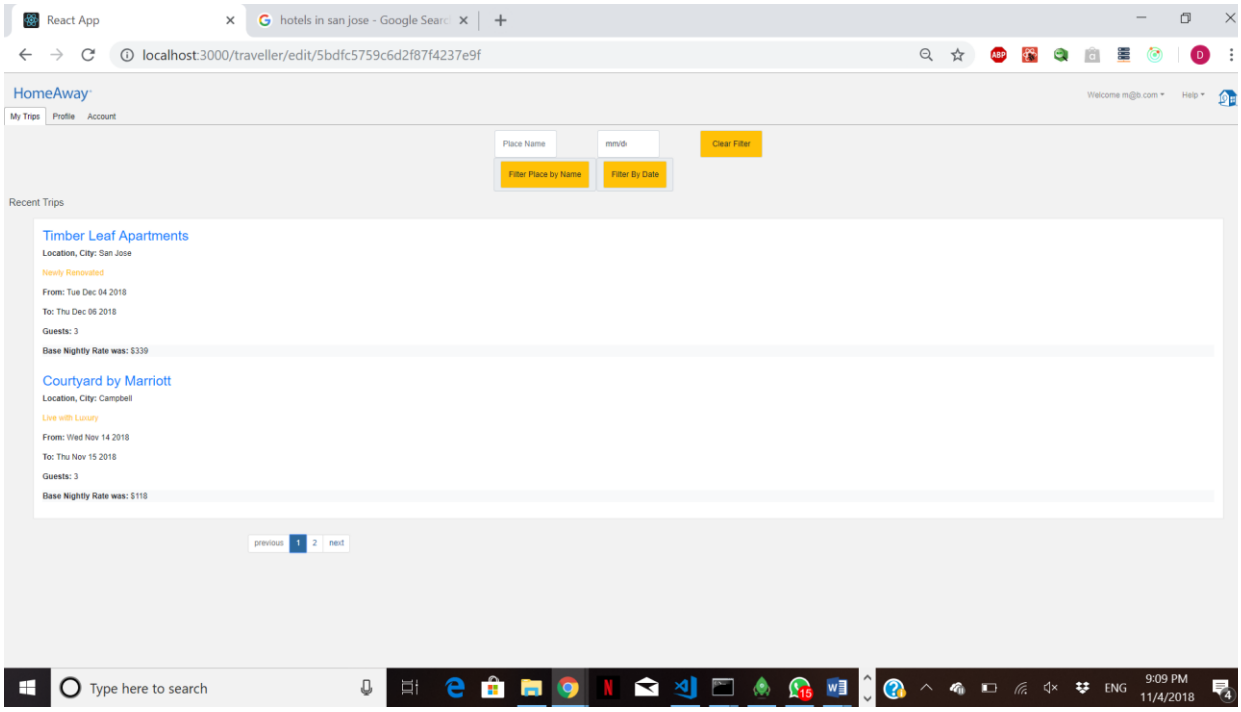


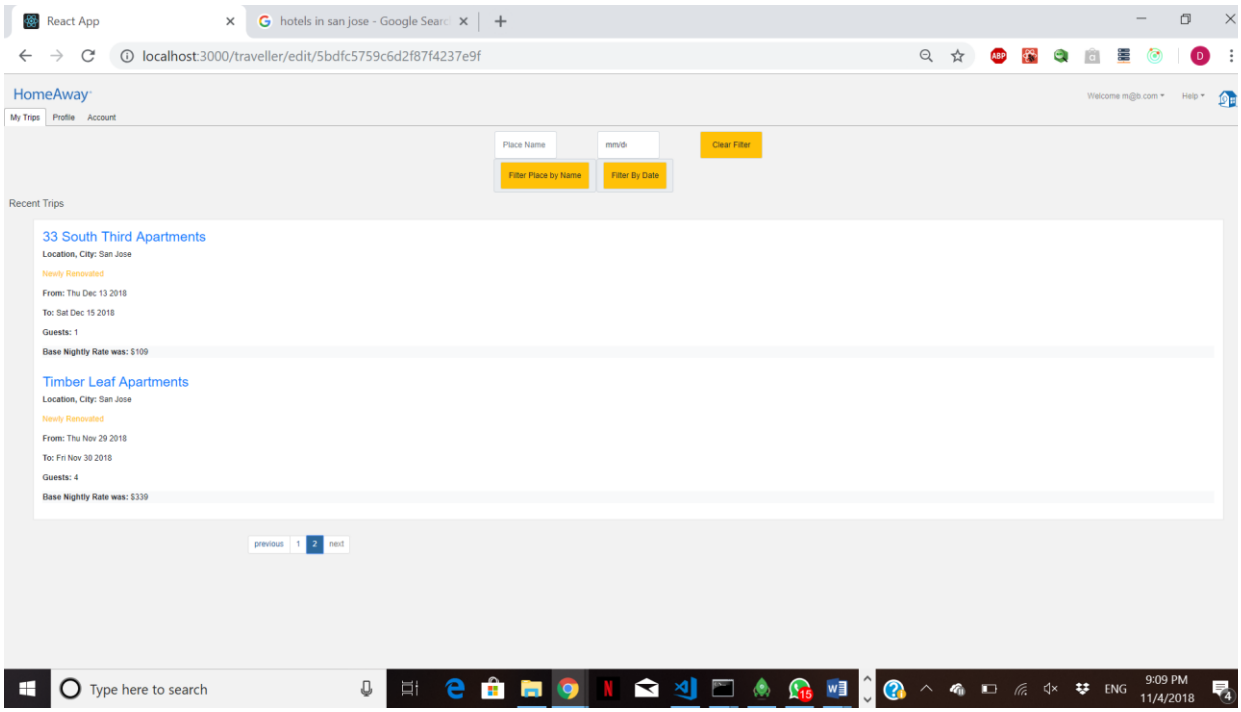
Explanation:

- This section shows how we updated the profile picture of the travel user and edited some details.
- The profile pic is stored on the server side at the location /public/uploads/profile-{travel_id}.jpg

- After that, one screenshot shows the recent changes done on the user profile on the redux store.

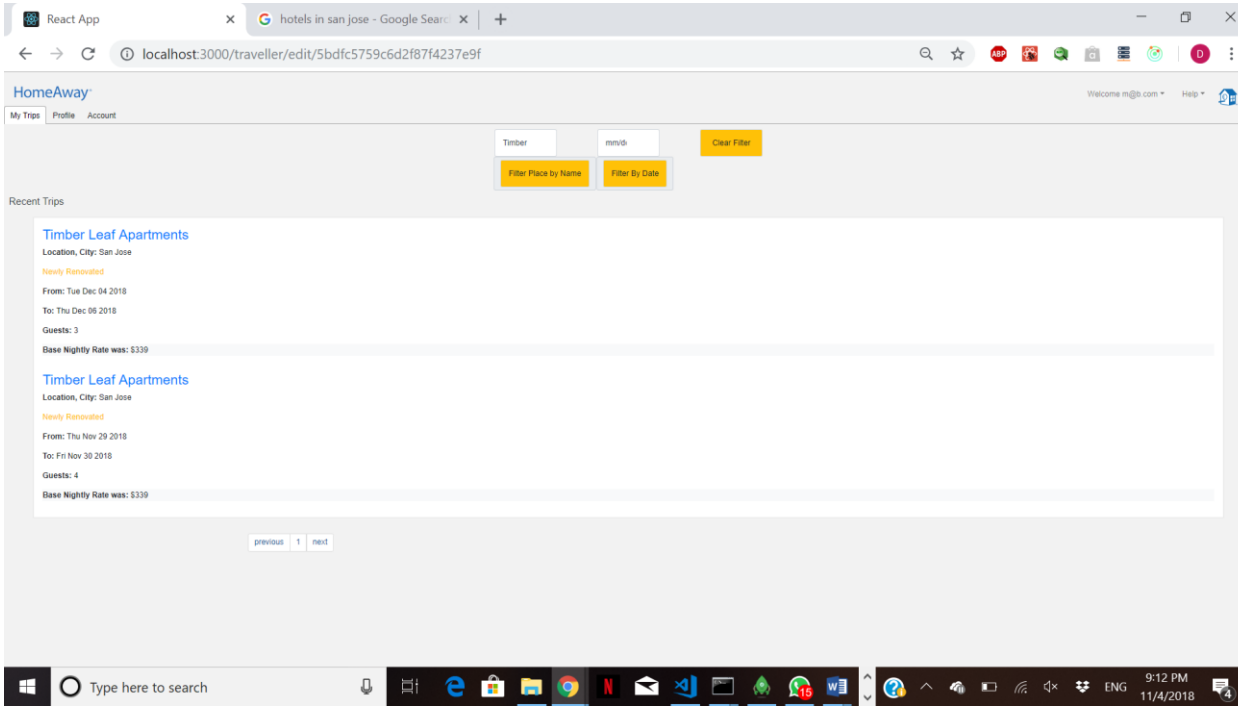
Looking at the recent trips that I have booked and filtering through it





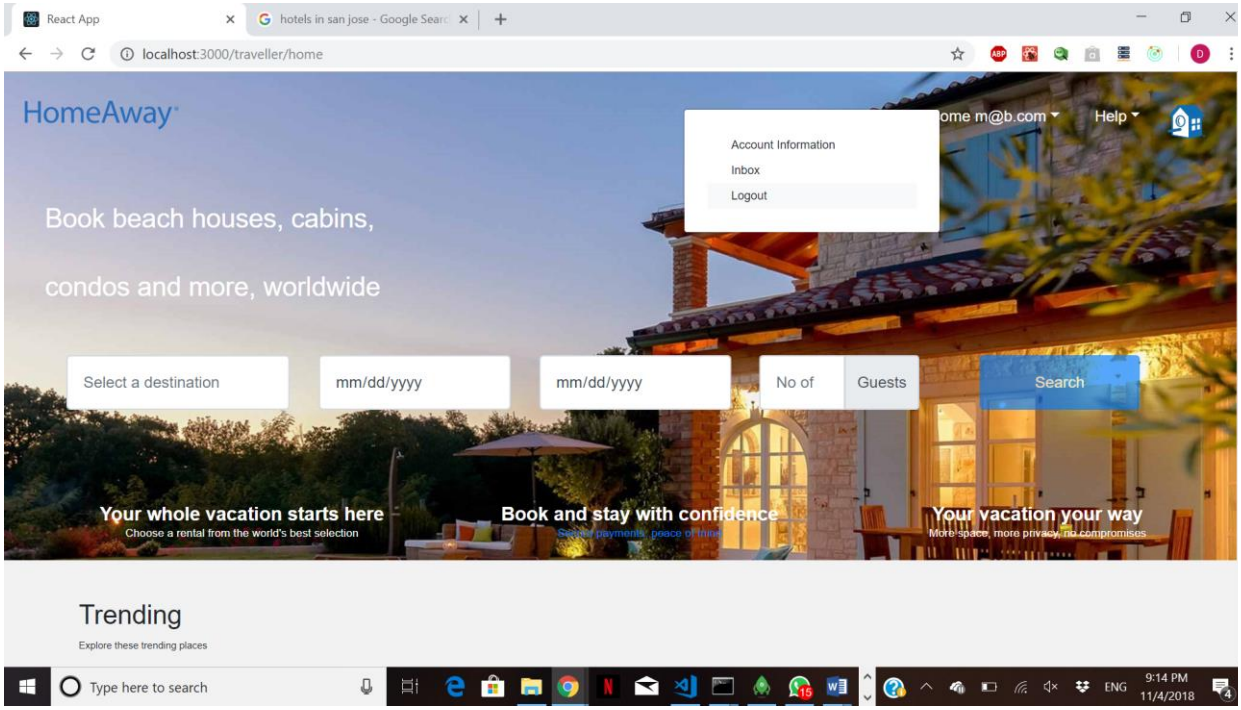
Explanation:
- I booked few other properties to demonstrate pagination and filtering options.
- The screenshot shows that I have booked 4 properties recently, and the bookings are divided into 2 pages as shown in the screenshots.(Pagination size is 2)

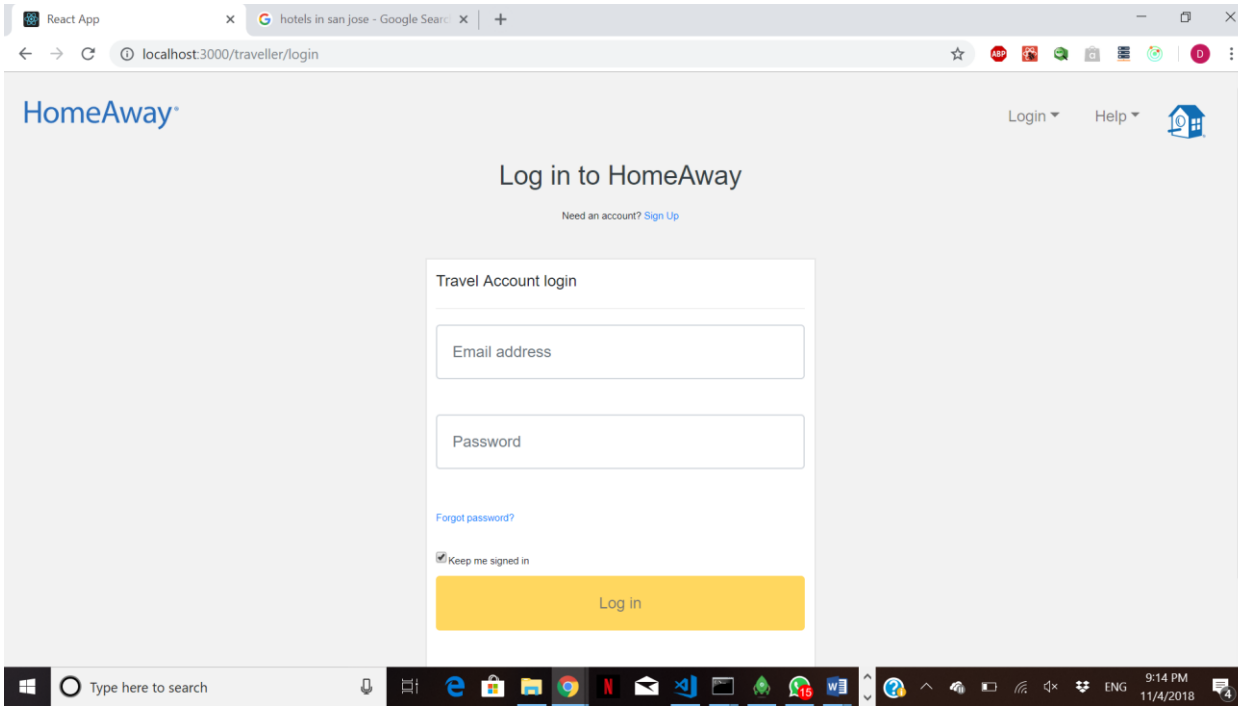
Lets filter through these bookings.



Explanation
- We filtered to see only the bookings I have done for Timber Leaf Apartments. There are only 2 bookings, hence 2 results.

Logging Off



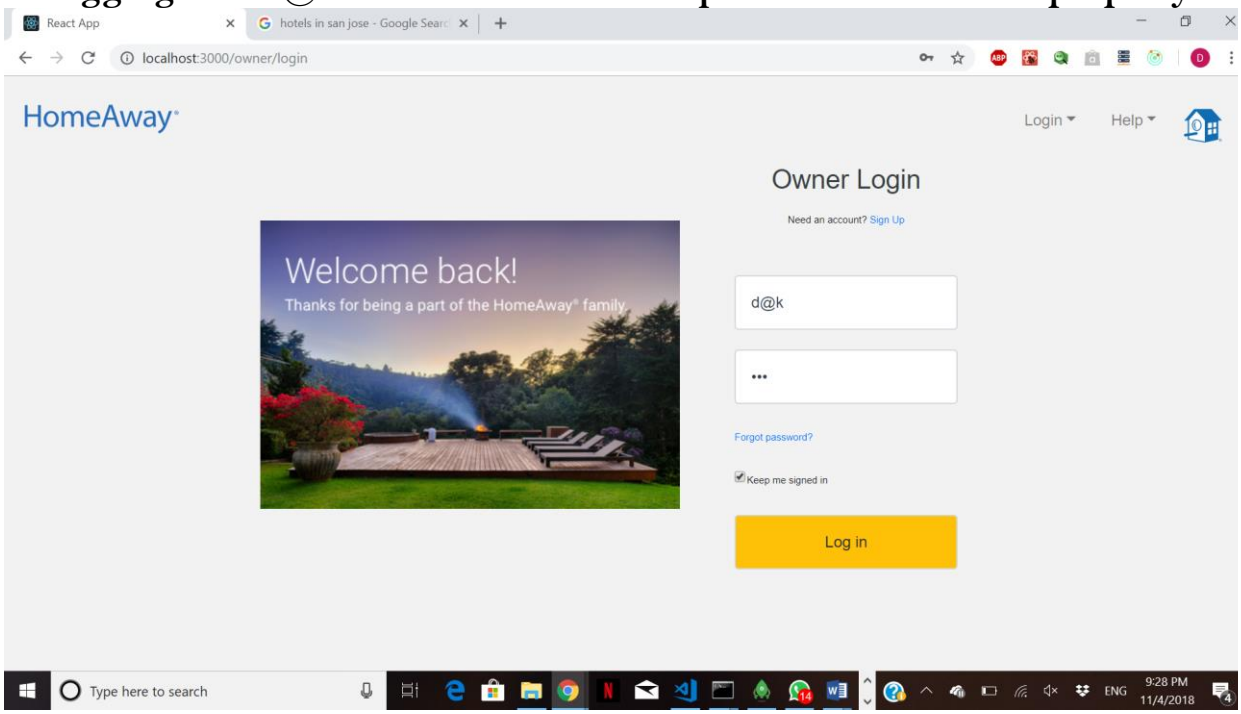


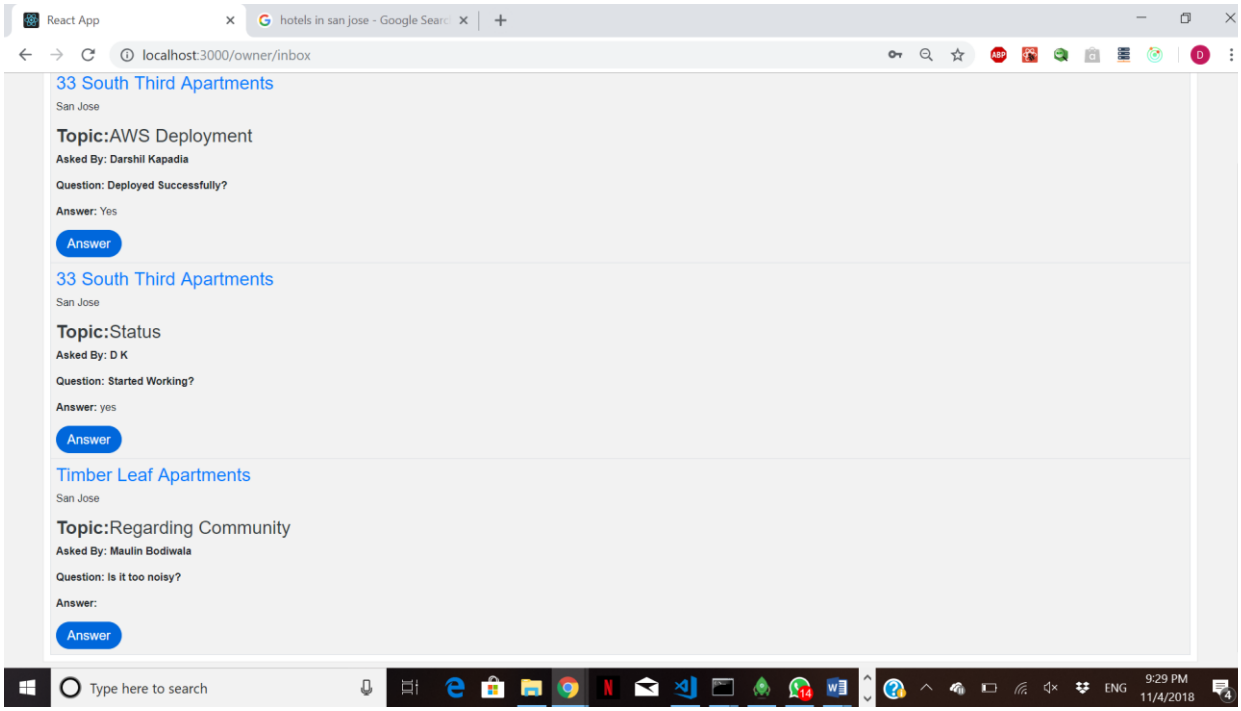
Explanation:

- Demonstrates a simple functioning of logging off.

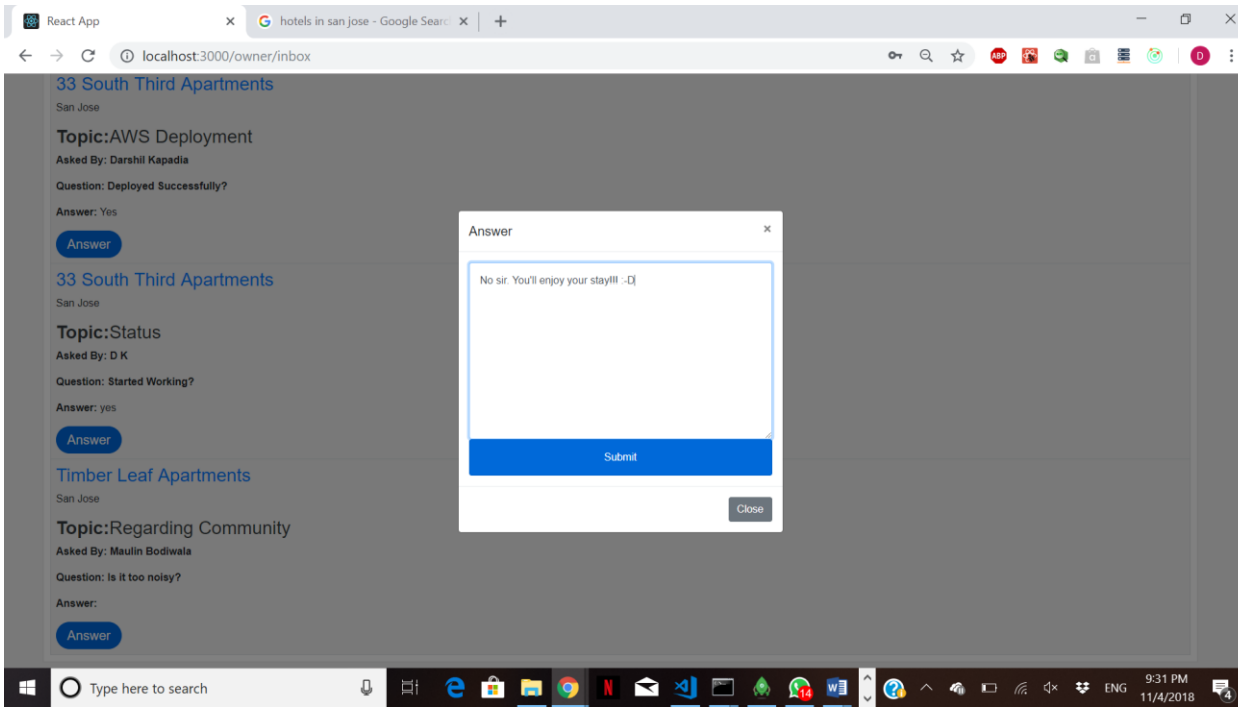
Answering to the question asked by the traveler

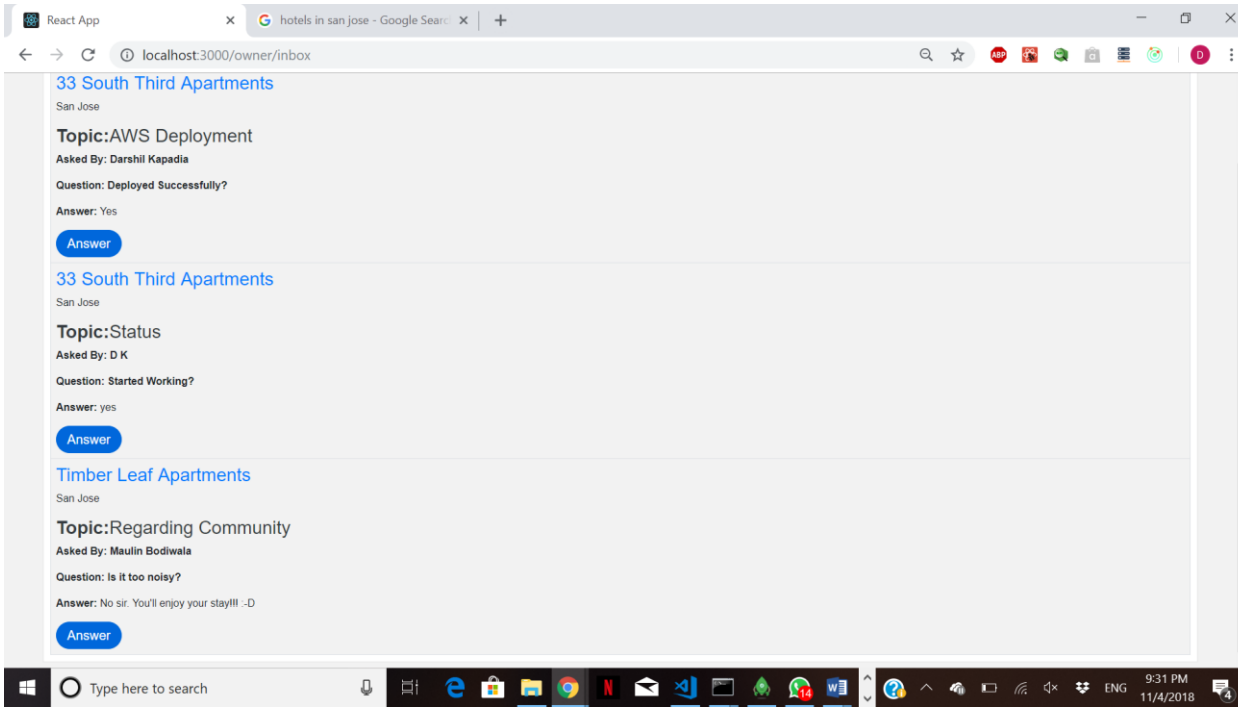
Logging into d@k owner to answer the question asked on his property



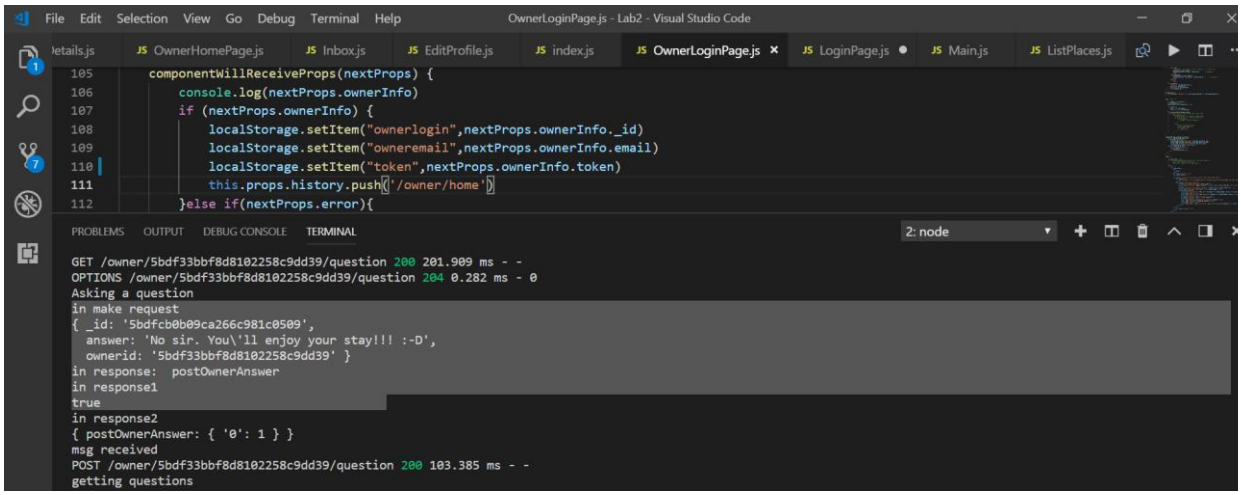


3rd Question is the one asked by our new traveler. Let's answer it


React App
hotels in san jose - Google Searc
localhost:3000/owner/inbox
33 South Third Apartments
San Jose
Topic:AWS Deployment
Asked By: Darshil Kapadia
Question: Deployed Successfully?
Answer: Yes
Answer
33 South Third Apartments
San Jose
Topic:Status
Asked By: D K
Question: Started Working?
Answer: yes
Answer
Timber Leaf Apartments
San Jose
Topic:Regarding Community
Asked By: Maulin Bodiwala
Question: Is it too noisy?
Answer:
Answer
Answer
No sir. You'll enjoy your stay!!! :-D
Submit
Close
Type here to search
ENG
9:31 PM
11/4/2018


React App
hotels in san jose - Google Searc
localhost:3000/owner/inbox
33 South Third Apartments
San Jose
Topic:AWS Deployment
Asked By: Darshil Kapadia
Question: Deployed Successfully?
Answer: Yes
Answer
33 South Third Apartments
San Jose
Topic:Status
Asked By: D K
Question: Started Working?
Answer: yes
Answer
Timber Leaf Apartments
San Jose
Topic:Regarding Community
Asked By: Maulin Bodiwala
Question: Is it too noisy?
Answer: No sir. You'll enjoy your stay!!! :-D
Answer
Type here to search
ENG
9:31 PM
11/4/2018

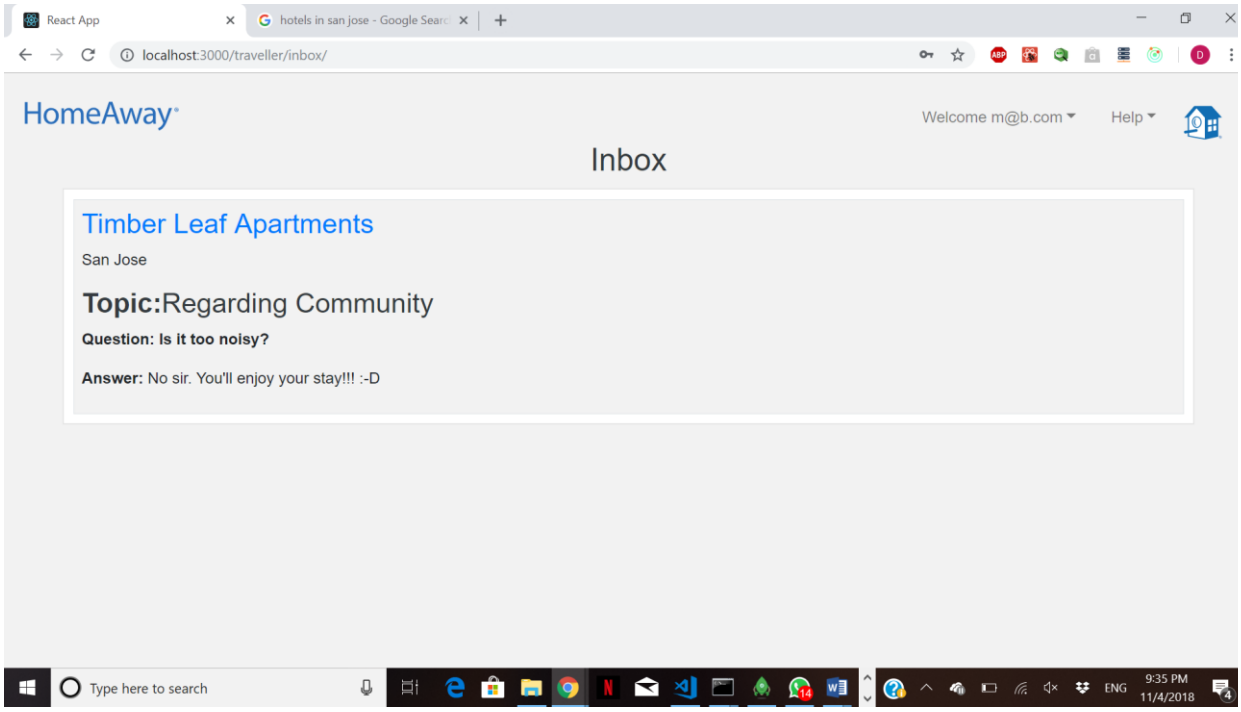
Answered



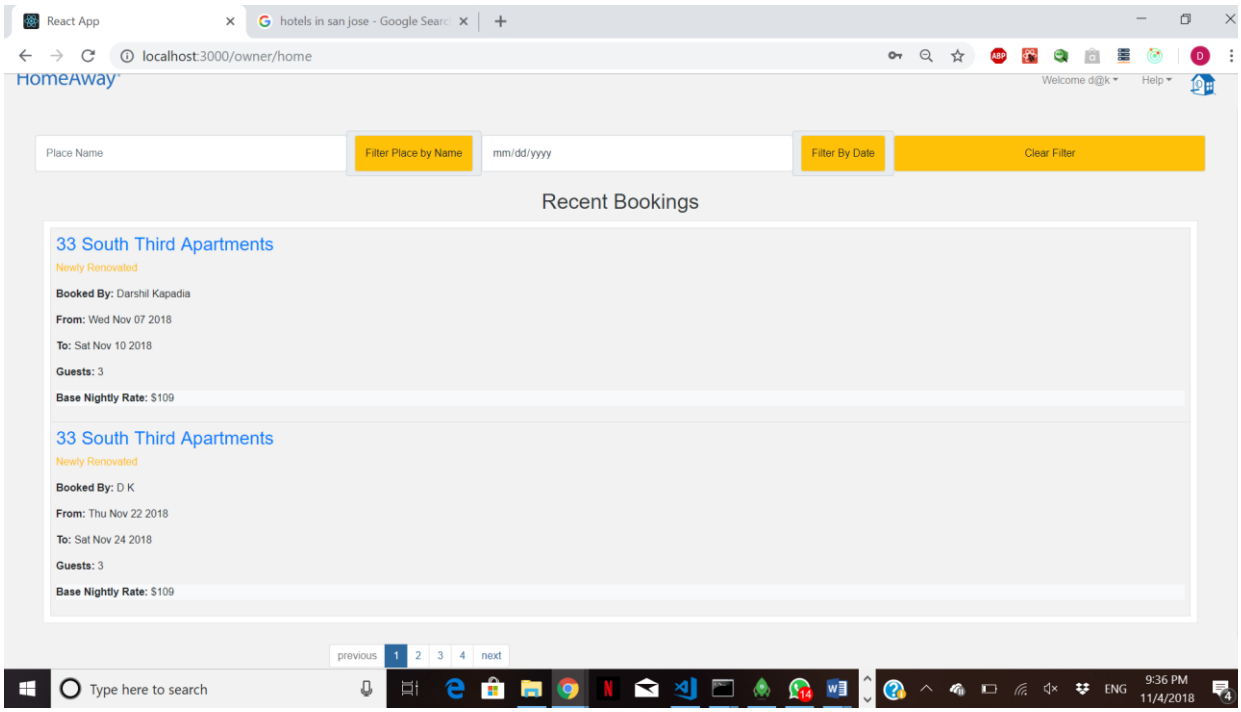
Explanation:

- As shown in the screenshots, we successfully answered the traveler's question. Afterwards, we'll verify whether traveler can see the answer or not?

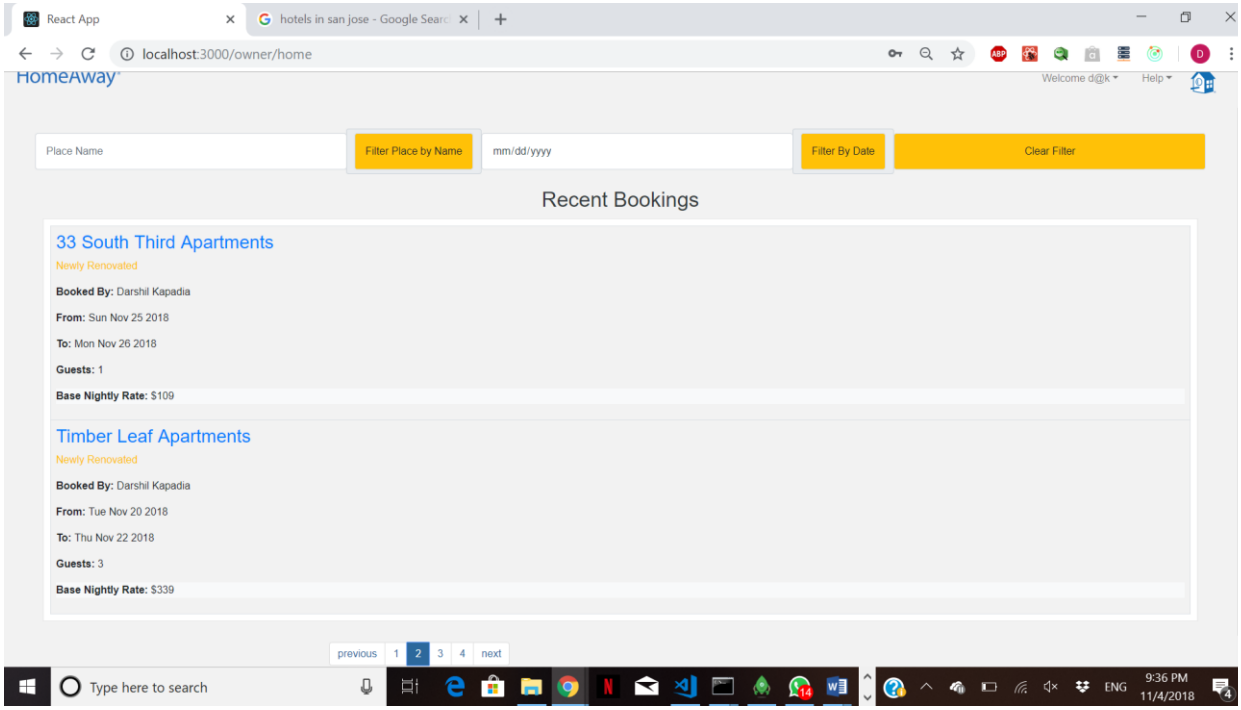
Checking the answer on traveler's side



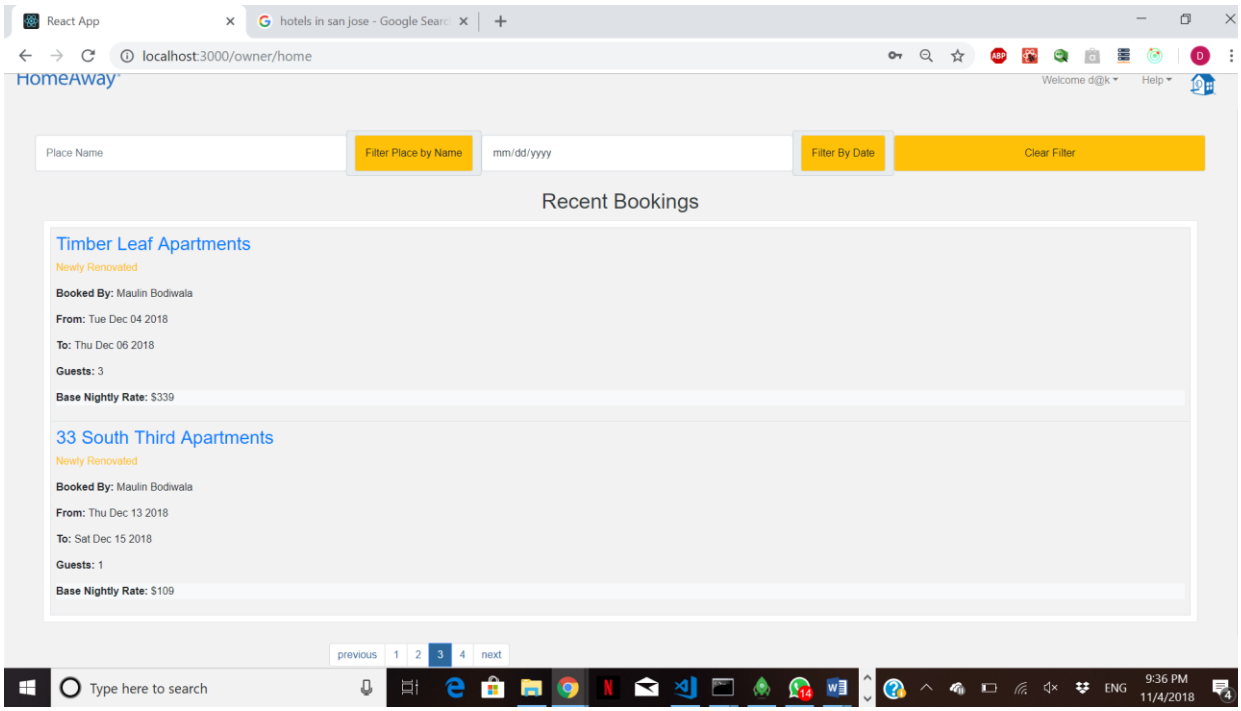
Looking at Owner's Dashboard



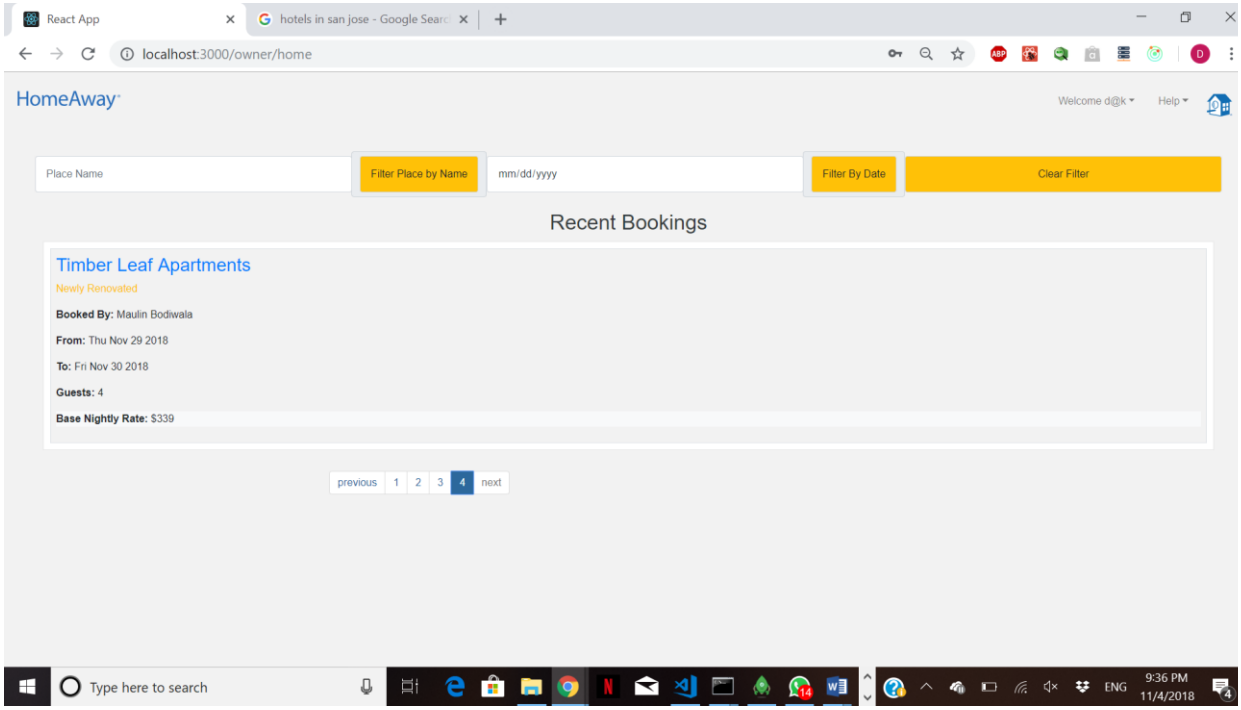
1st Page



2nd page

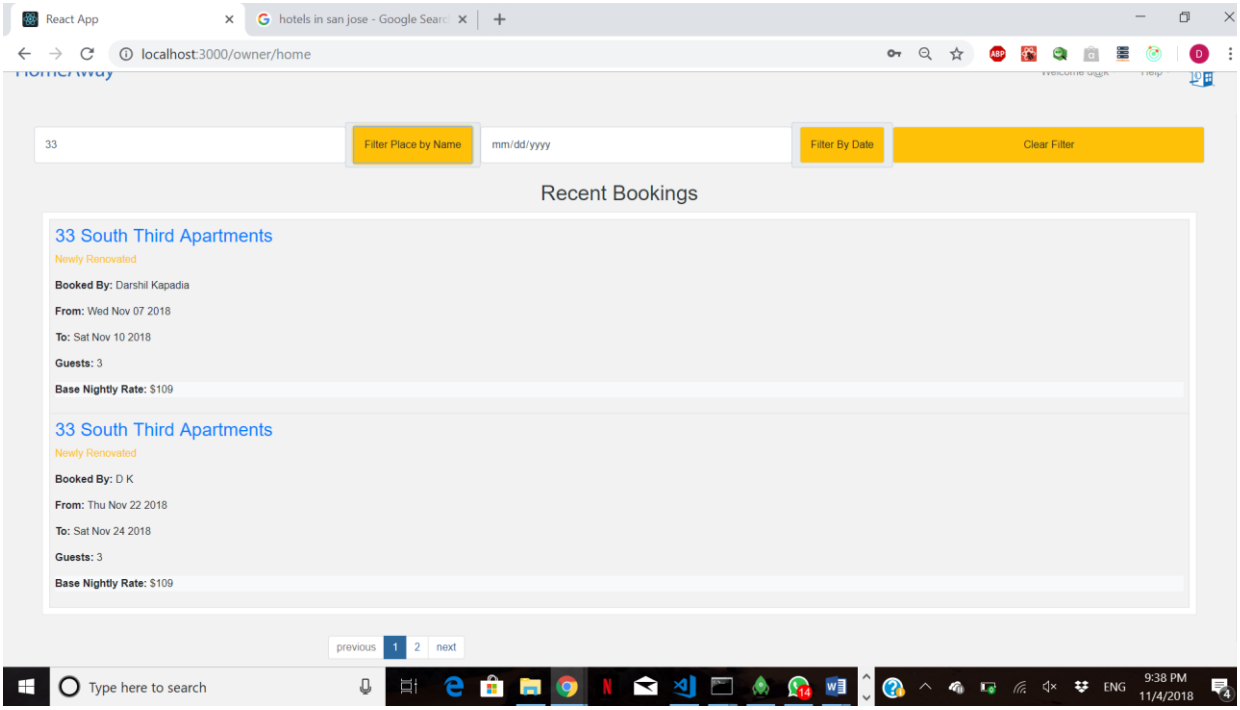


3rd Page

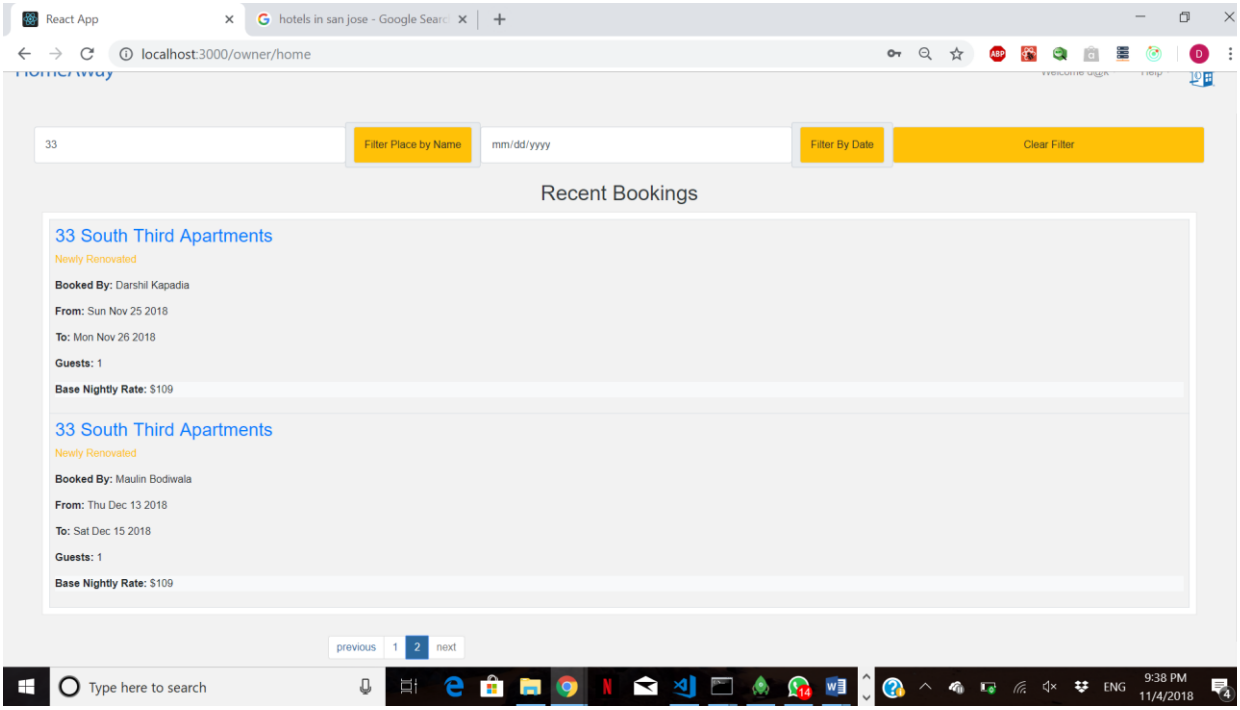


4th Page

Filtering through recent bookings



Results having only 33 South Third Apartment Bookings

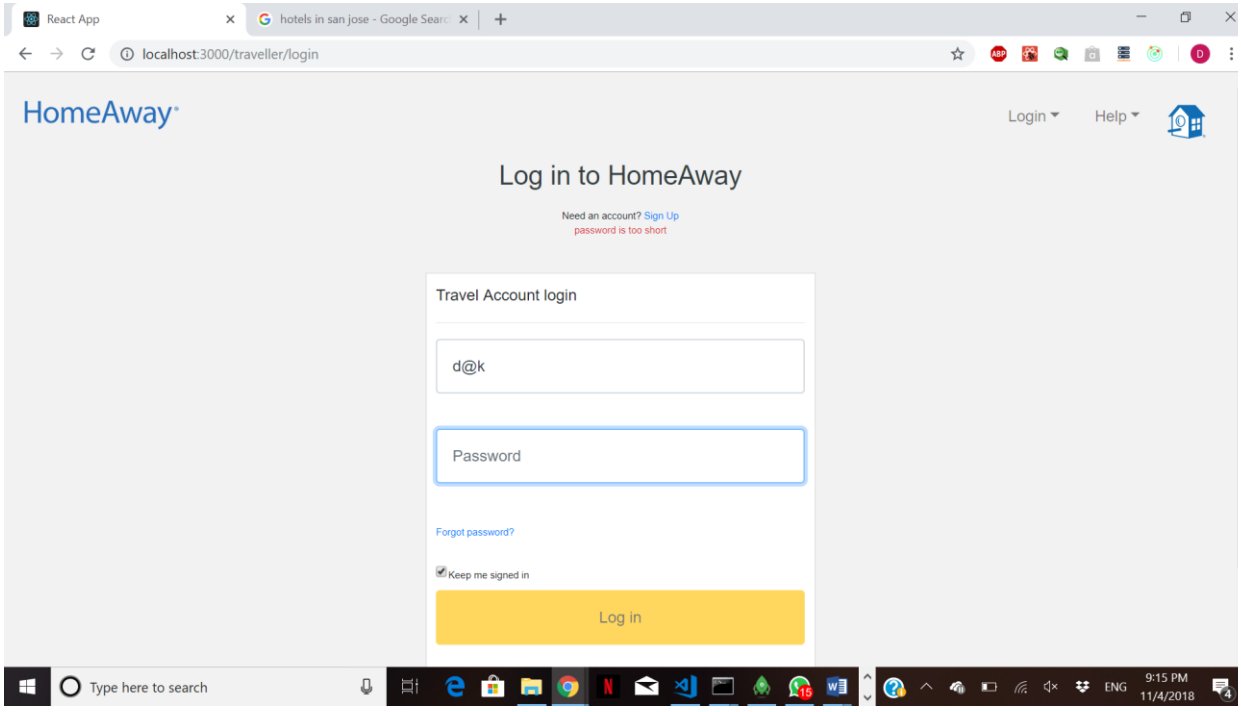


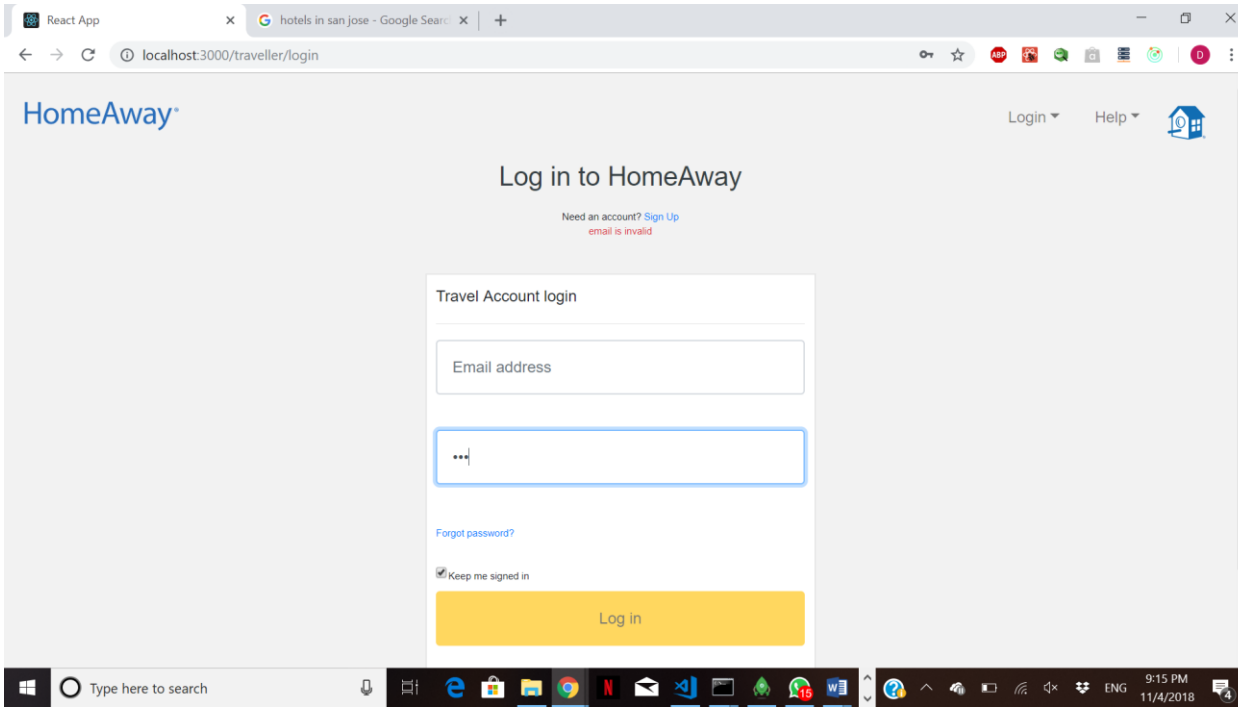
Explanation:

- We filtered to get only the bookings made on property named "33 South Third Apartments"

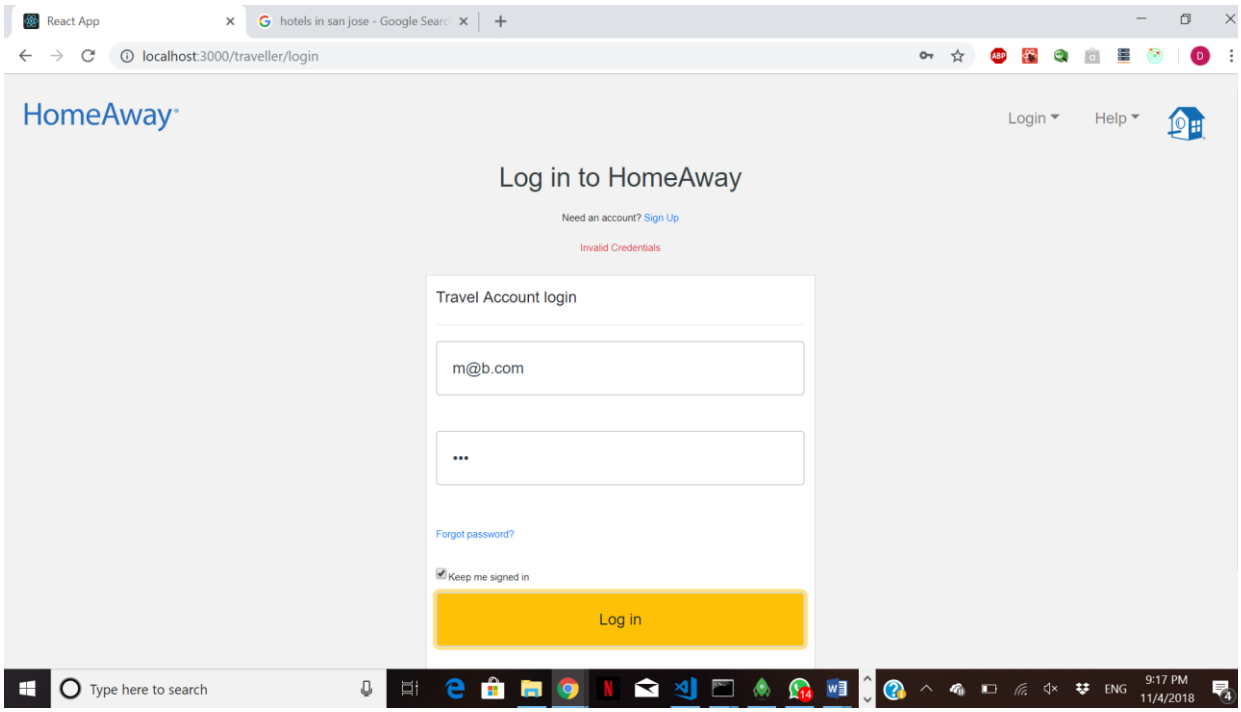
Validation Test-Cases' Results:

Keeping the email/password field empty

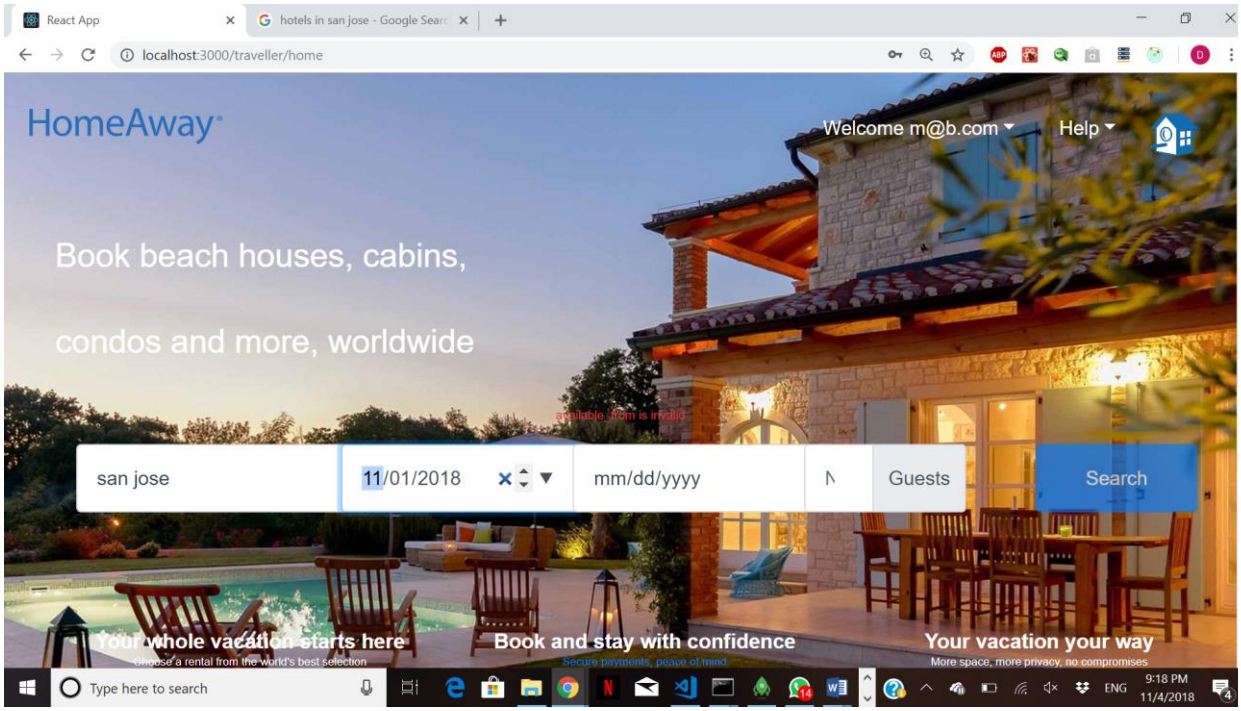


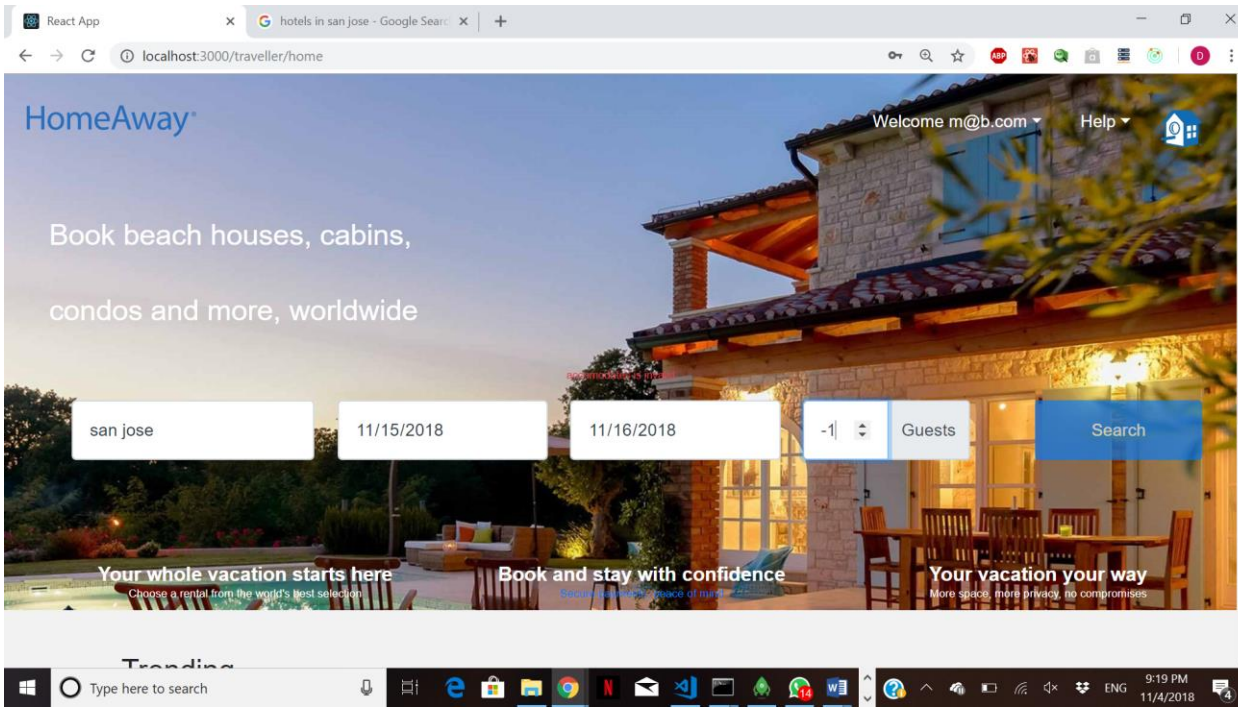


Logging in with incorrect credentials



Searching the property without location city or arrival date or departure date or guests

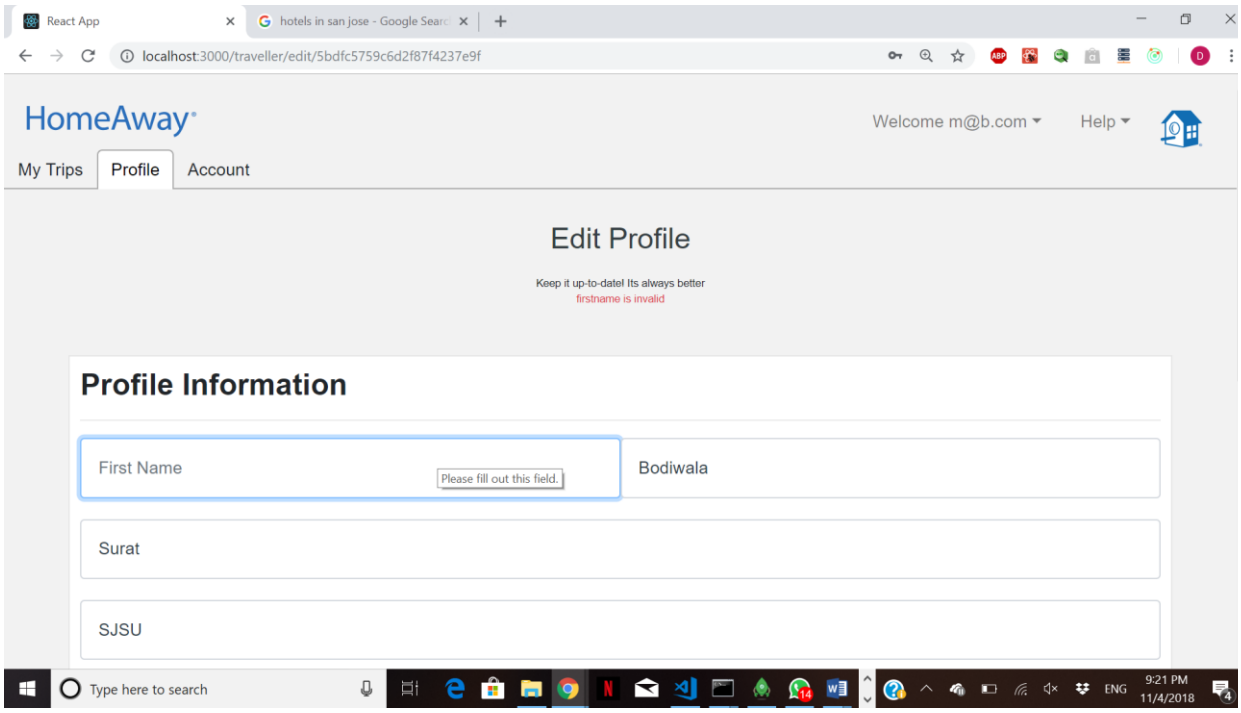


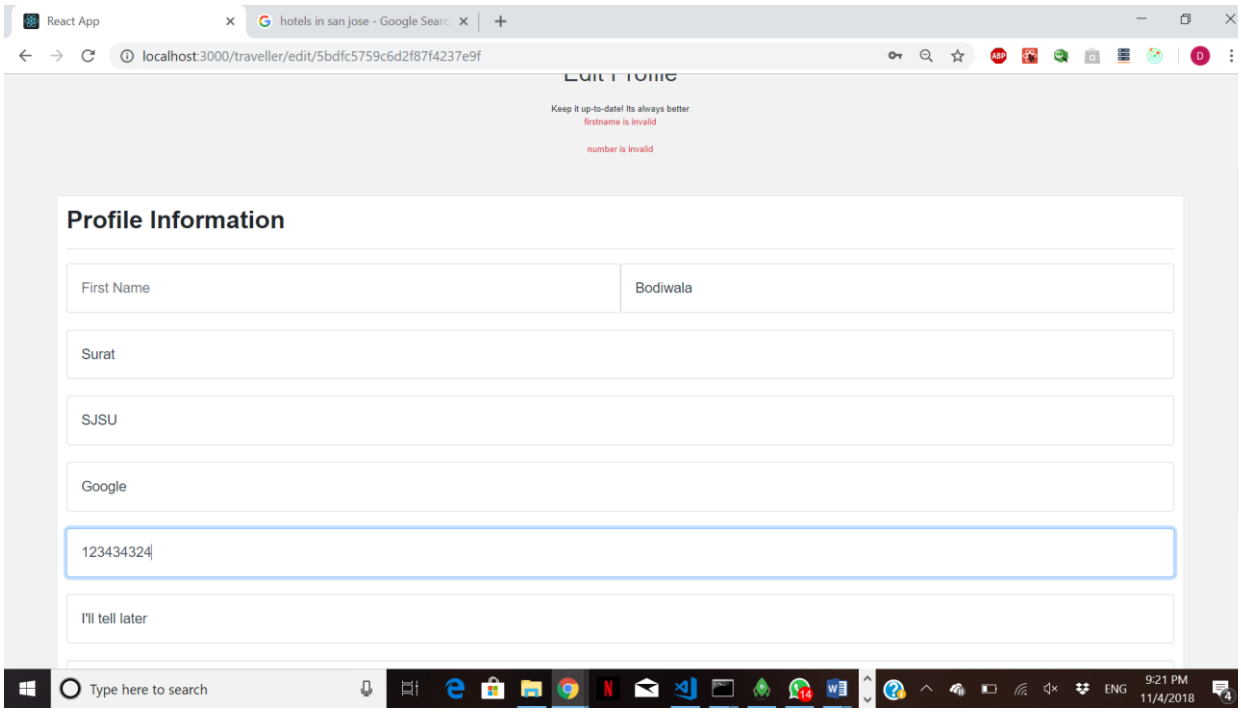


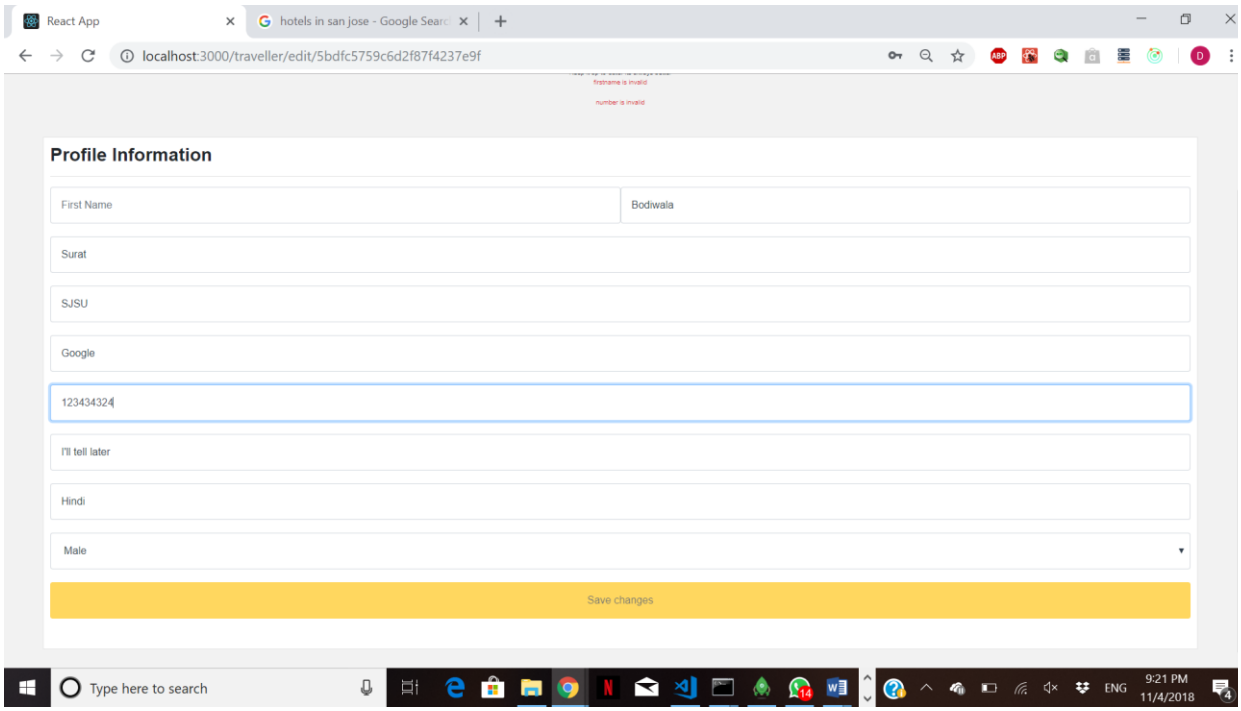
Explanation:

- First screenshot shows that the arrival date is before today's date and will show an error to the user saying "available_from is invalid".
- Second screenshot shows that the guests cannot be negative. We selected -1 as our guests which is never possible and thus it gets reflected.
- In both the cases, please notice the search button. It is disabled throughout. It will only get enabled if all the details are passing the validations.

Editing the information and keeping required fields like name blank and contact no not of 10 digits







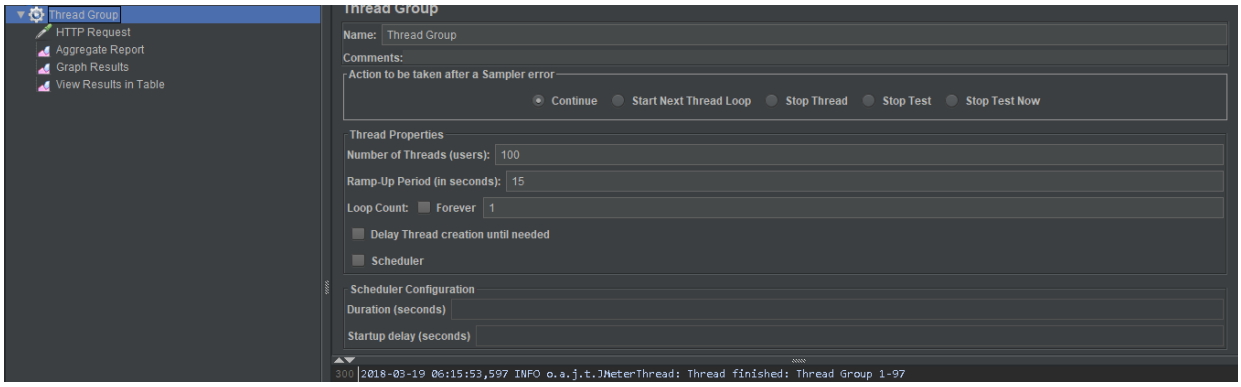
Explanation:

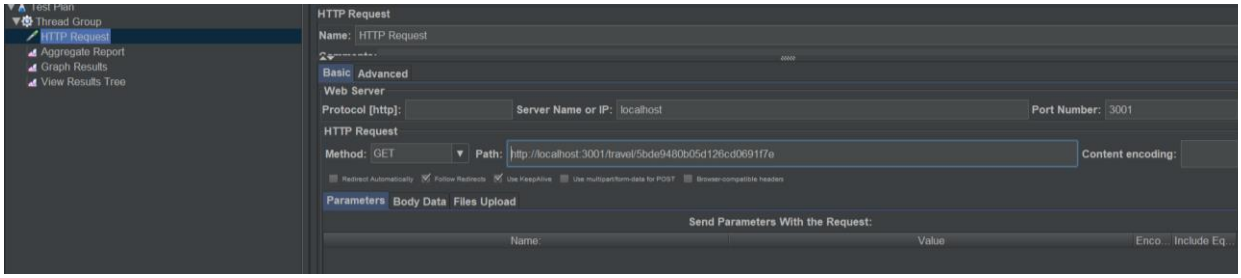
- As it shows, one cannot have empty name in the database and one cannot have more than or less than 10 digits in the contact no.
- Because these validations were proved wrong, the "Save Changes" button is disabled and the user is shown the error above the form.

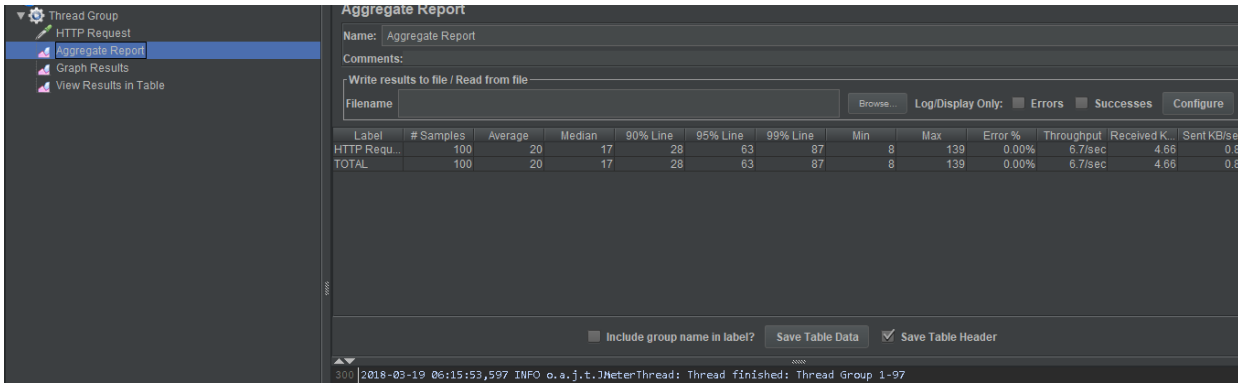
Performance

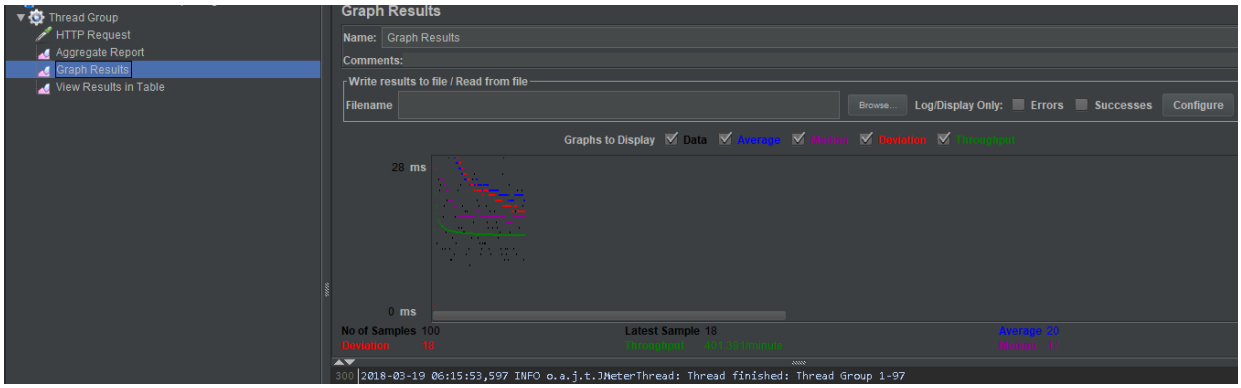
J-Meter (With Connection Pooling)

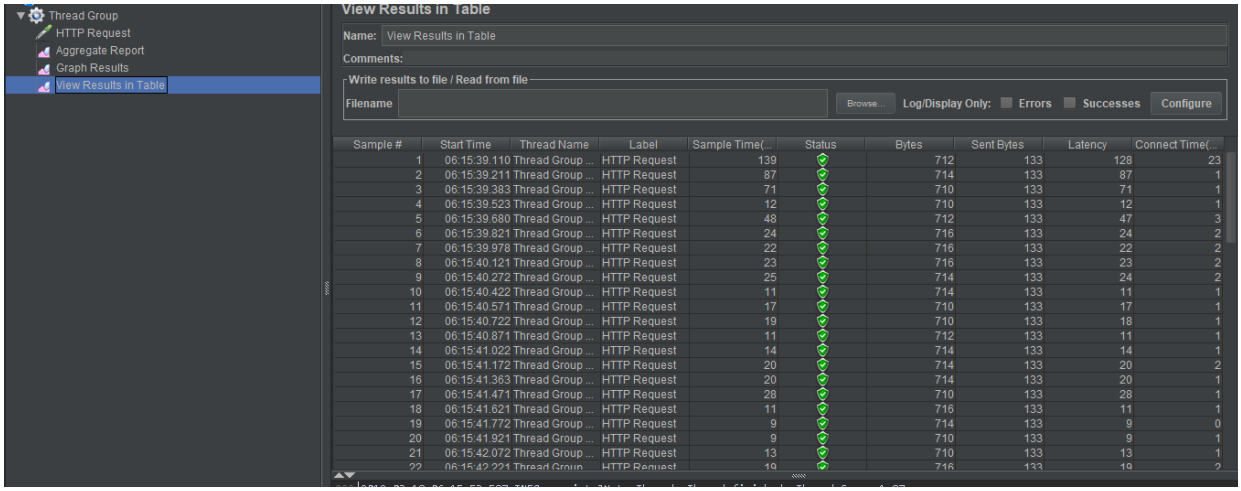
On Invoking 100 Concurrent Users

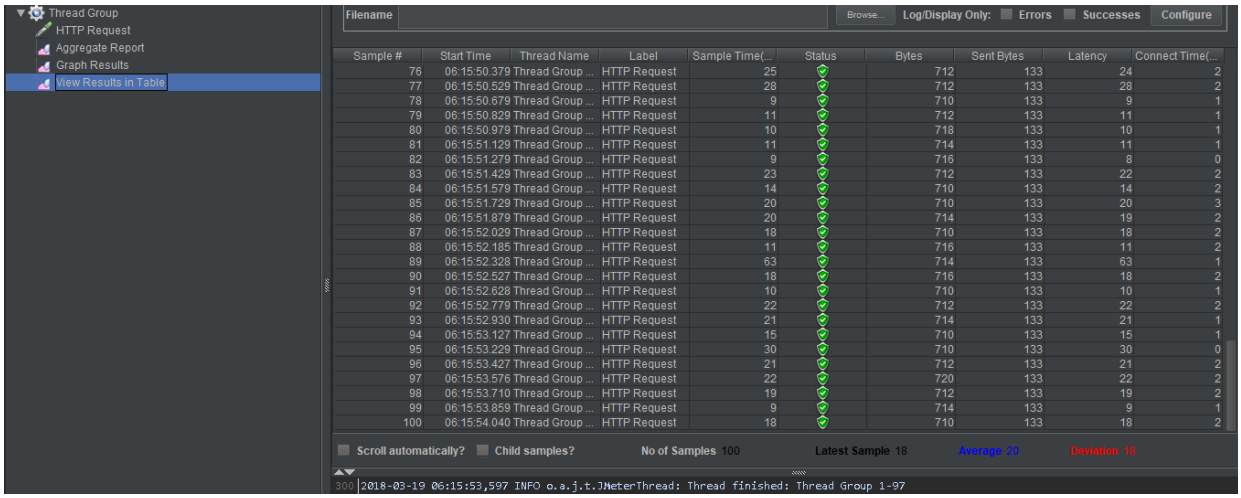




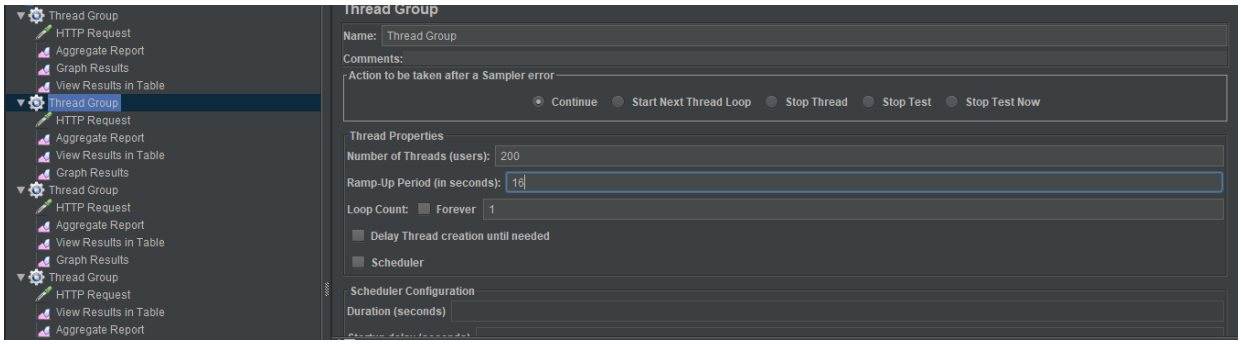


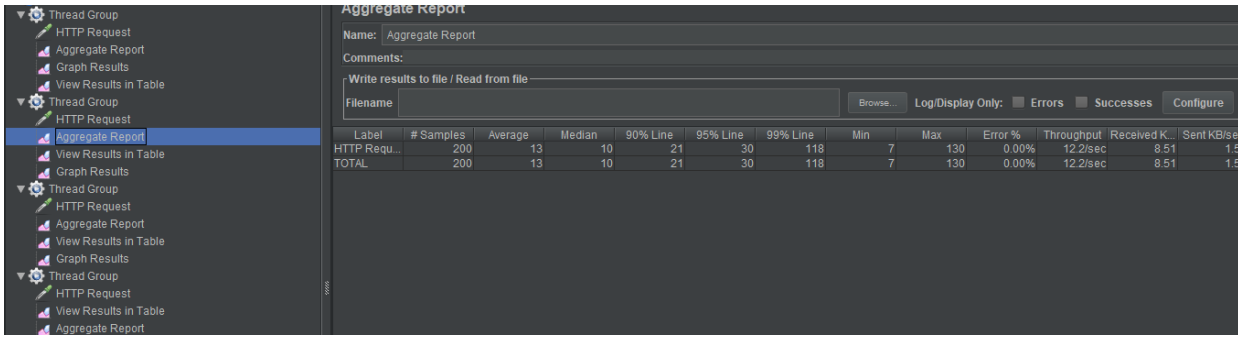


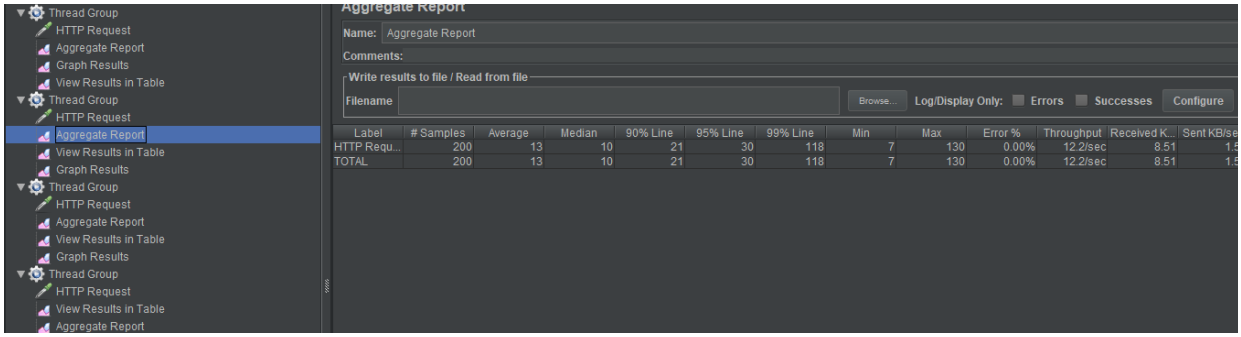


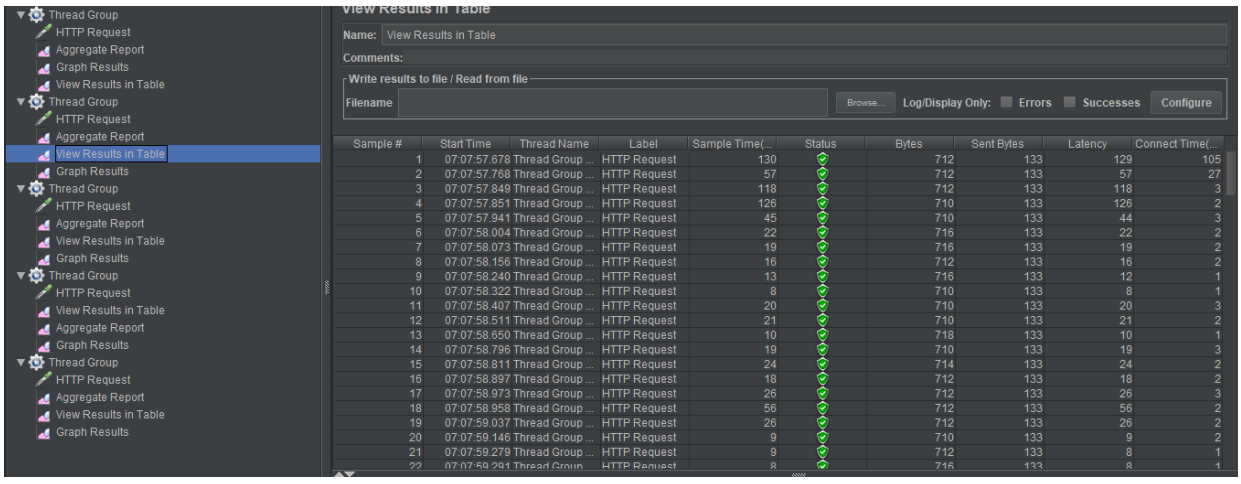


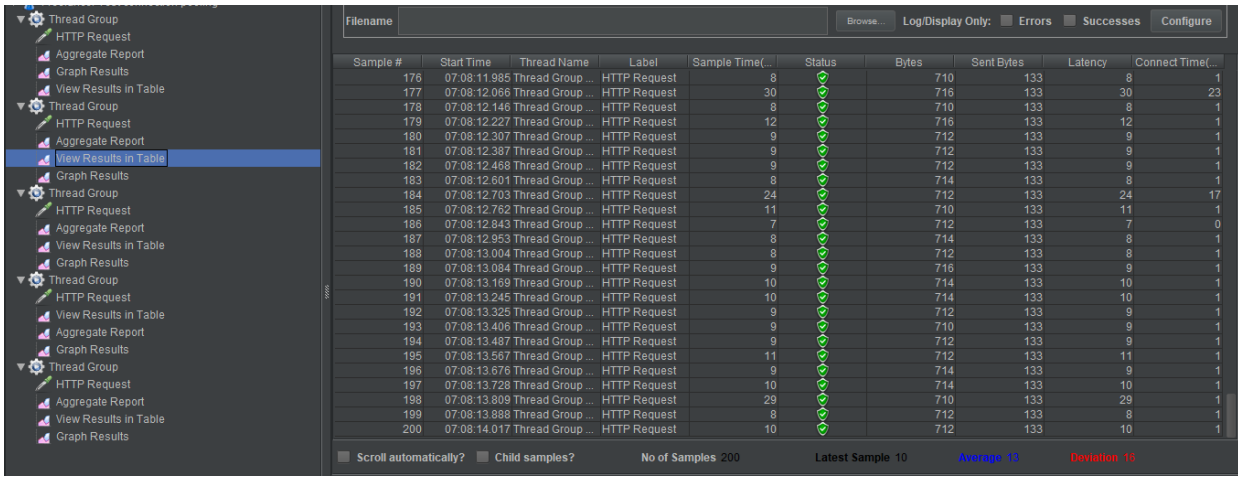
On invoking 200 concurrent users

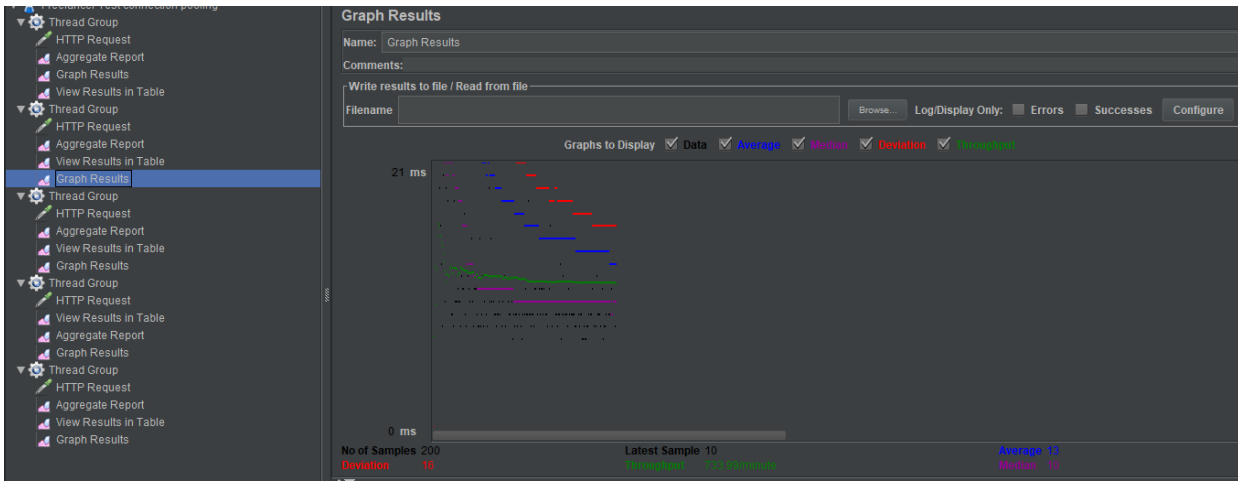

Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
Thread Group
Name: Thread Group
Comments:
Action to be taken after a Sampler error
Continue
Start Next Thread Loop
Stop Thread
Stop Test
Stop Test Now
Thread Properties
Number of Threads (users): 200
Ramp-Up Period (in seconds): 16
Loop Count: Forever 1
Delay Thread creation until needed
Scheduler
Scheduler Configuration
Duration (seconds)


Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
Aggregate Report
Name: Aggregate Report
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Label # Samples Average Median 90% Line 95% Line 99% Line Min Max Error % Throughput Received K... Sent KB/se
HTTP Requ... 200 13 10 21 30 118 7 130 0.00% 12.2/sec 8.51 1.5
TOTAL 200 13 10 21 30 118 7 130 0.00% 12.2/sec 8.51 1.5

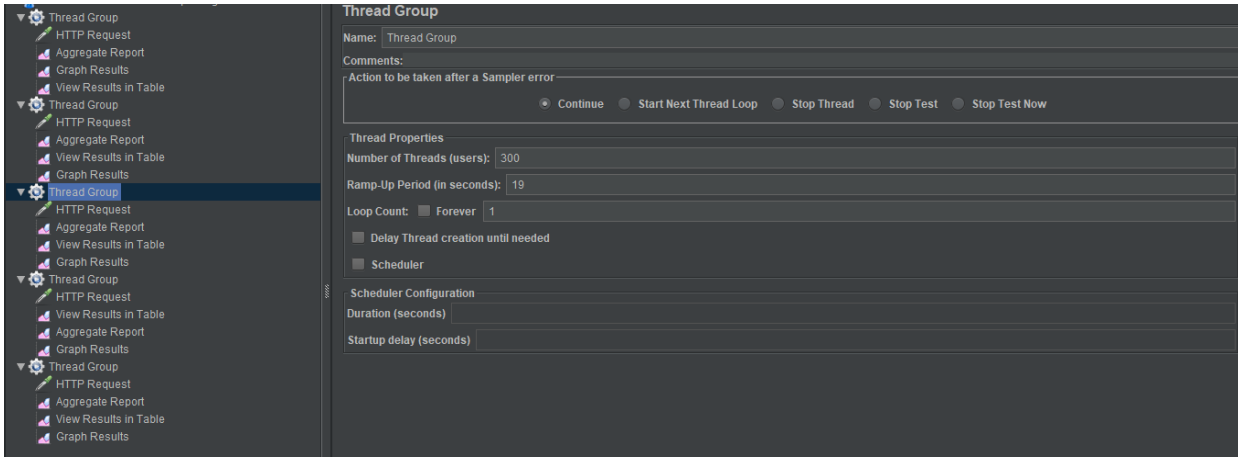

Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
Aggregate Report
Name: Aggregate Report
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Label # Samples Average Median 90% Line 95% Line 99% Line Min Max Error % Throughput Received K... Sent KB/se
HTTP Requ... 200 13 10 21 30 118 7 130 0.00% 12.2/sec 8.51 1.5
TOTAL 200 13 10 21 30 118 7 130 0.00% 12.2/sec 8.51 1.5

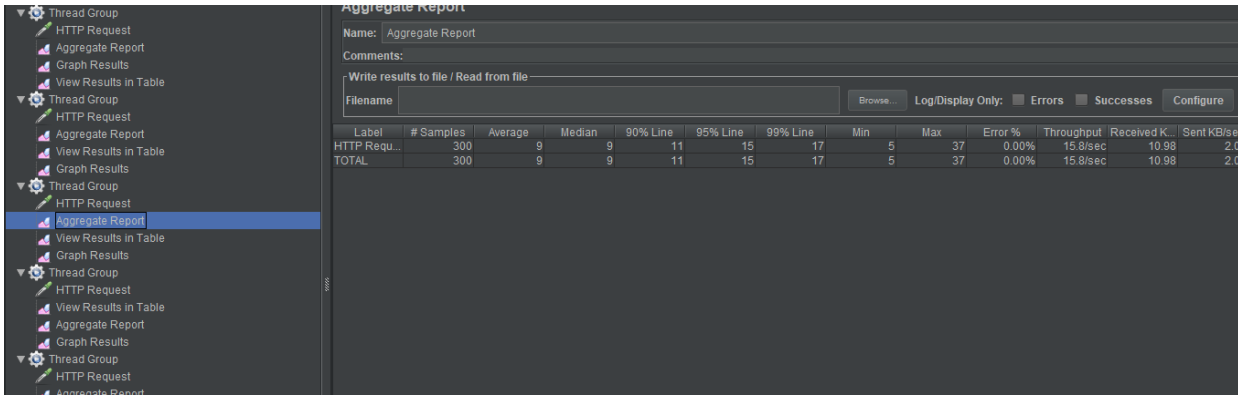

Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
View Results in Table
Name: View Results in Table
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Sample # Start Time Thread Name Label Sample Time(... Status Bytes Sent Bytes Latency Connect Time(...
1 07:07:57.678 Thread Group ... HTTP Request 130 712 133 129 105
2 07:07:57.768 Thread Group ... HTTP Request 57 712 133 57 27
3 07:07:57.849 Thread Group ... HTTP Request 118 712 133 118 3
4 07:07:57.851 Thread Group ... HTTP Request 126 710 133 126 2
5 07:07:57.941 Thread Group ... HTTP Request 45 710 133 44 3
6 07:07:58.004 Thread Group ... HTTP Request 22 716 133 22 2
7 07:07:58.073 Thread Group ... HTTP Request 19 716 133 19 2
8 07:07:58.156 Thread Group ... HTTP Request 16 712 133 16 2
9 07:07:58.240 Thread Group ... HTTP Request 13 716 133 12 1
10 07:07:58.322 Thread Group ... HTTP Request 8 710 133 8 1
11 07:07:58.407 Thread Group ... HTTP Request 20 710 133 20 3
12 07:07:58.511 Thread Group ... HTTP Request 21 710 133 21 2
13 07:07:58.650 Thread Group ... HTTP Request 10 718 133 10 1
14 07:07:58.796 Thread Group ... HTTP Request 19 710 133 19 3
15 07:07:58.811 Thread Group ... HTTP Request 24 714 133 24 2
16 07:07:58.897 Thread Group ... HTTP Request 18 712 133 18 2
17 07:07:58.973 Thread Group ... HTTP Request 26 712 133 26 3
18 07:07:58.958 Thread Group ... HTTP Request 56 712 133 56 2
19 07:07:59.037 Thread Group ... HTTP Request 26 712 133 26 2
20 07:07:59.146 Thread Group ... HTTP Request 9 710 133 9 2
21 07:07:59.279 Thread Group ... HTTP Request 9 712 133 8 1

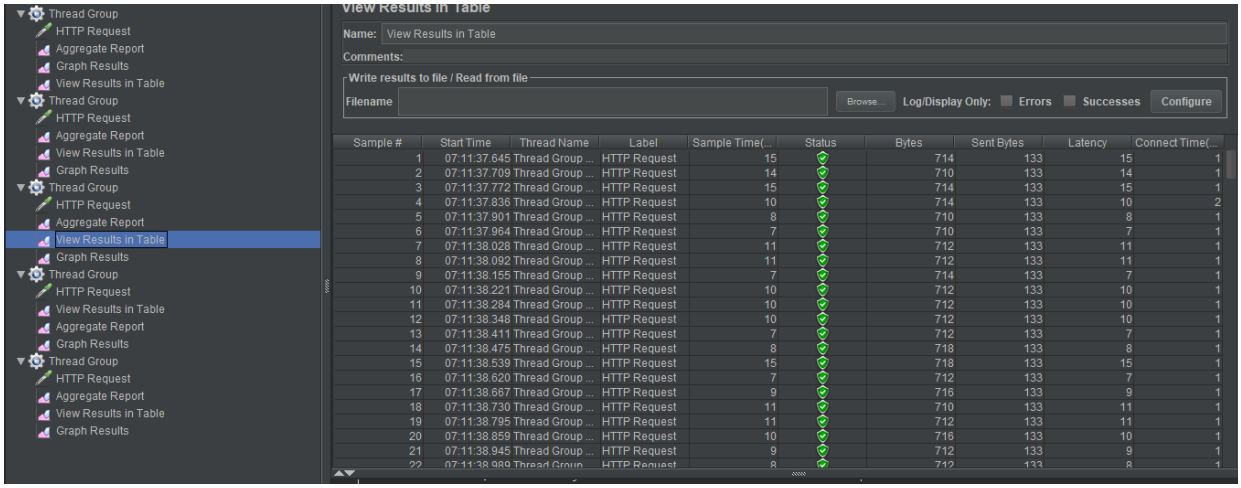


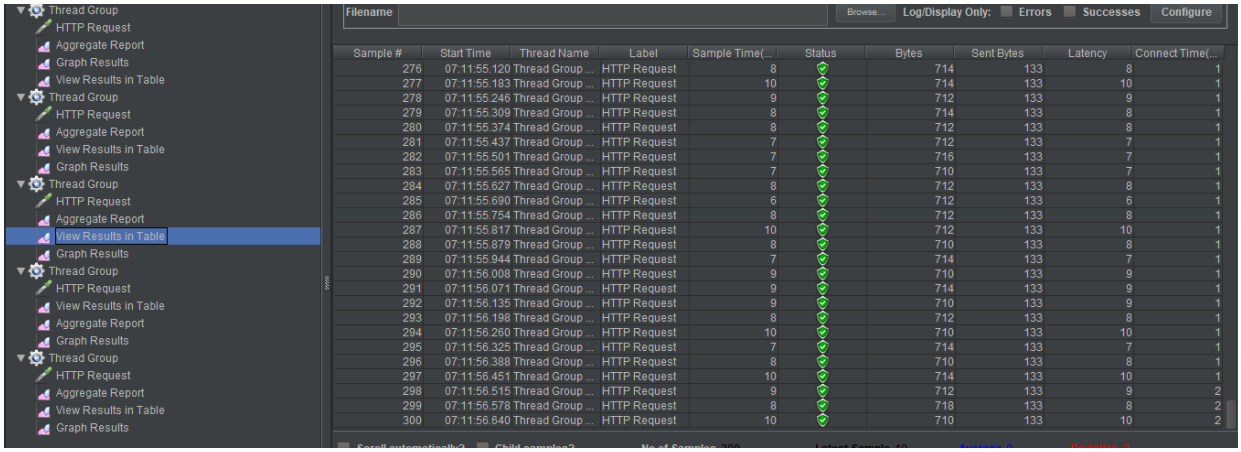


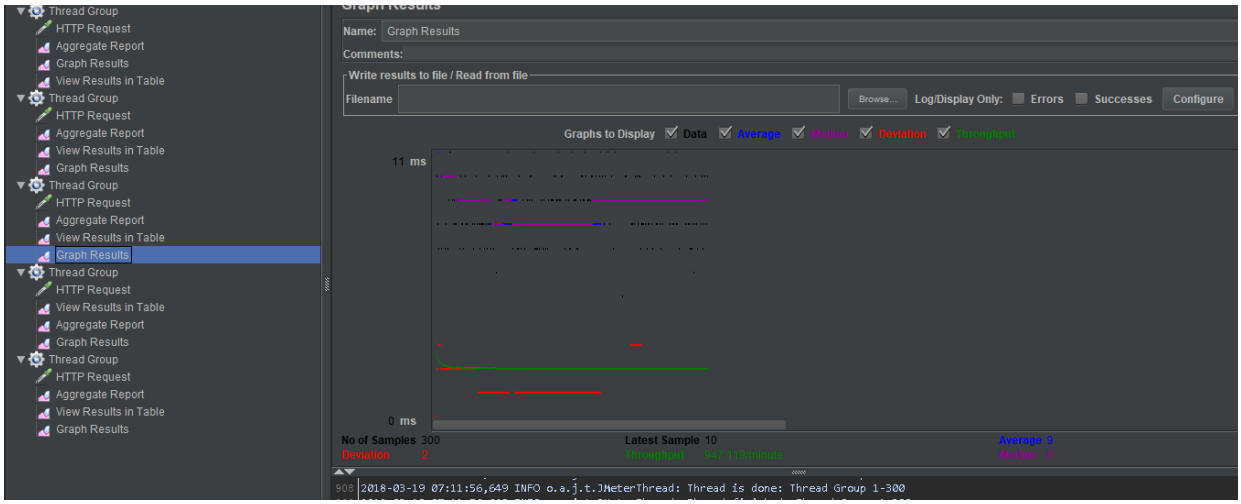
On Invoking 300 Concurrent Users



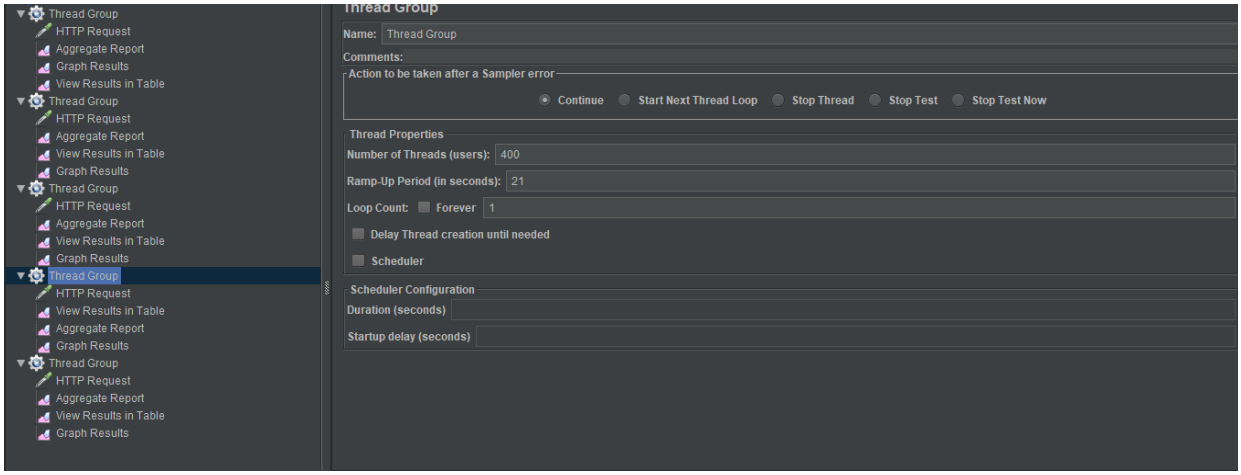

Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
Graph Results
Thread Group
HTTP Request
Aggregate Report
Name: Aggregate Report
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Label # Samples Average Median 90% Line 95% Line 99% Line Min Max Error % Throughput Received K... Sent KB/se
HTTP Requ... 300 9 9 11 15 17 5 37 0.00% 15.8/sec 10.98 2.0
TOTAL 300 9 9 11 15 17 5 37 0.00% 15.8/sec 10.98 2.0

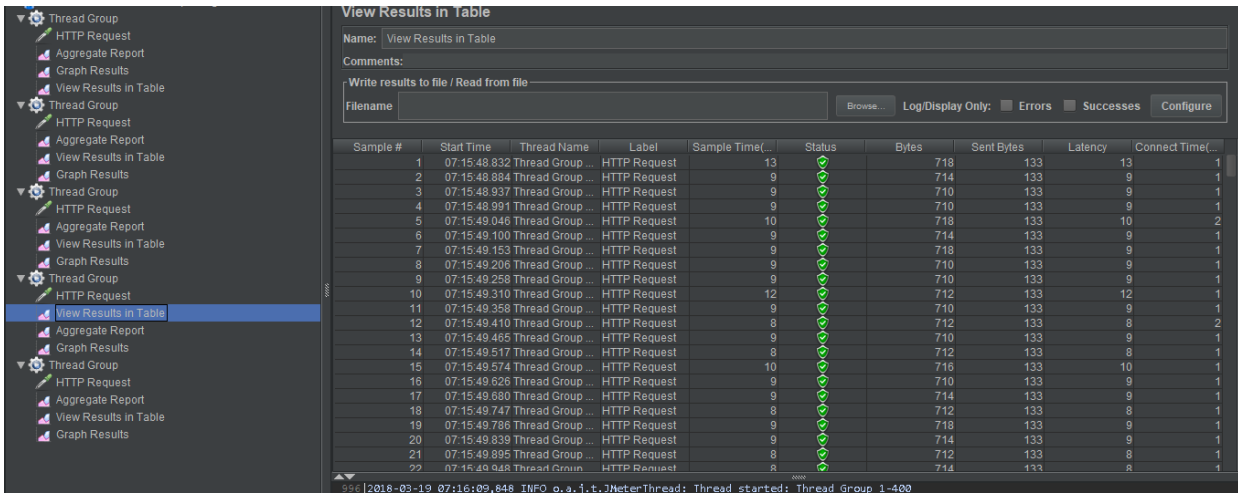

Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
View Results in Table
Name: View Results in Table
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Sample # Start Time Thread Name Label Sample Time(... Status Bytes Sent Bytes Latency Connect Time(...
1 07:11:37.645 Thread Group ... HTTP Request 15 714 133 15 1
2 07:11:37.709 Thread Group ... HTTP Request 14 710 133 14 1
3 07:11:37.772 Thread Group ... HTTP Request 15 714 133 15 1
4 07:11:37.836 Thread Group ... HTTP Request 10 714 133 10 2
5 07:11:37.901 Thread Group ... HTTP Request 8 710 133 8 1
6 07:11:37.964 Thread Group ... HTTP Request 7 710 133 7 1
7 07:11:38.028 Thread Group ... HTTP Request 11 712 133 11 1
8 07:11:38.092 Thread Group ... HTTP Request 11 712 133 11 1
9 07:11:38.155 Thread Group ... HTTP Request 7 714 133 7 1
10 07:11:38.221 Thread Group ... HTTP Request 10 712 133 10 1
11 07:11:38.284 Thread Group ... HTTP Request 10 712 133 10 1
12 07:11:38.348 Thread Group ... HTTP Request 10 712 133 10 1
13 07:11:38.411 Thread Group ... HTTP Request 7 712 133 7 1
14 07:11:38.475 Thread Group ... HTTP Request 8 718 133 8 1
15 07:11:38.539 Thread Group ... HTTP Request 15 718 133 15 1
16 07:11:38.620 Thread Group ... HTTP Request 7 712 133 7 1
17 07:11:38.667 Thread Group ... HTTP Request 9 716 133 9 1
18 07:11:38.730 Thread Group ... HTTP Request 11 710 133 11 1
19 07:11:38.795 Thread Group ... HTTP Request 11 712 133 11 1
20 07:11:38.859 Thread Group ... HTTP Request 10 716 133 10 1
21 07:11:38.945 Thread Group ... HTTP Request 9 712 133 9 1

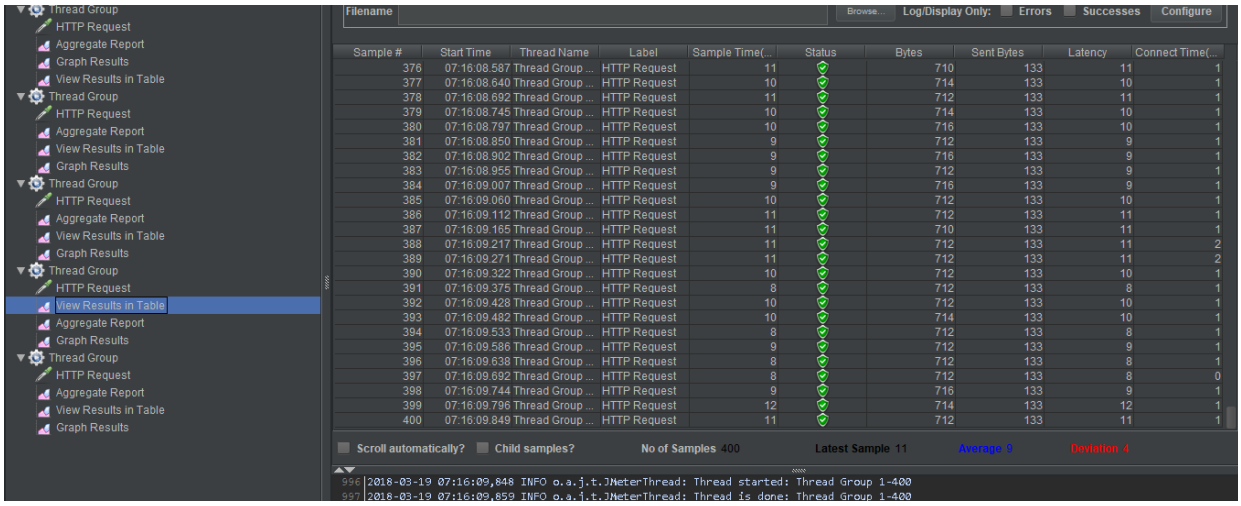

Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Sample # Start Time Thread Name Label Sample Time(... Status Bytes Sent Bytes Latency Connect Time(...
276 07:11:55.120 Thread Group ... HTTP Request 8 714 133 8 1
277 07:11:55.183 Thread Group ... HTTP Request 10 714 133 10 1
278 07:11:55.246 Thread Group ... HTTP Request 9 712 133 9 1
279 07:11:55.309 Thread Group ... HTTP Request 8 714 133 8 1
280 07:11:55.374 Thread Group ... HTTP Request 8 712 133 8 1
281 07:11:55.437 Thread Group ... HTTP Request 7 712 133 7 1
282 07:11:55.501 Thread Group ... HTTP Request 7 716 133 7 1
283 07:11:55.565 Thread Group ... HTTP Request 7 710 133 7 1
284 07:11:55.627 Thread Group ... HTTP Request 8 712 133 8 1
285 07:11:55.690 Thread Group ... HTTP Request 6 712 133 6 1
286 07:11:55.754 Thread Group ... HTTP Request 8 712 133 8 1
287 07:11:55.817 Thread Group ... HTTP Request 10 712 133 10 1
288 07:11:55.879 Thread Group ... HTTP Request 8 710 133 8 1
289 07:11:55.944 Thread Group ... HTTP Request 7 714 133 7 1
290 07:11:56.008 Thread Group ... HTTP Request 9 710 133 9 1
291 07:11:56.071 Thread Group ... HTTP Request 9 714 133 9 1
292 07:11:56.135 Thread Group ... HTTP Request 9 710 133 9 1
293 07:11:56.198 Thread Group ... HTTP Request 8 712 133 8 1
294 07:11:56.260 Thread Group ... HTTP Request 10 710 133 10 1
295 07:11:56.325 Thread Group ... HTTP Request 7 714 133 7 1
296 07:11:56.388 Thread Group ... HTTP Request 8 710 133 8 1
297 07:11:56.451 Thread Group ... HTTP Request 10 714 133 10 1
298 07:11:56.515 Thread Group ... HTTP Request 9 712 133 9 2
299 07:11:56.578 Thread Group ... HTTP Request 8 718 133 8 2
300 07:11:56.640 Thread Group ... HTTP Request 10 710 133 10 2

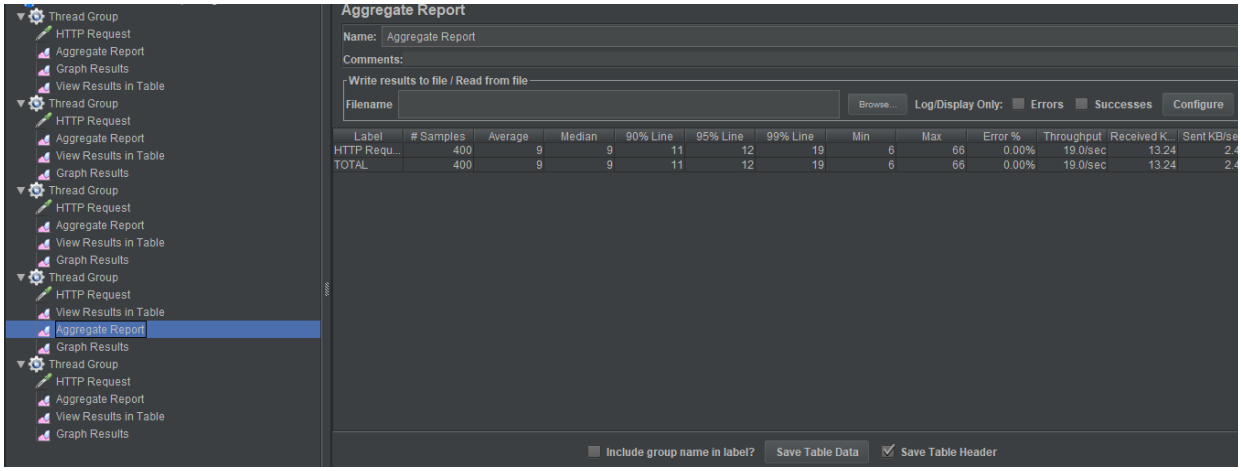


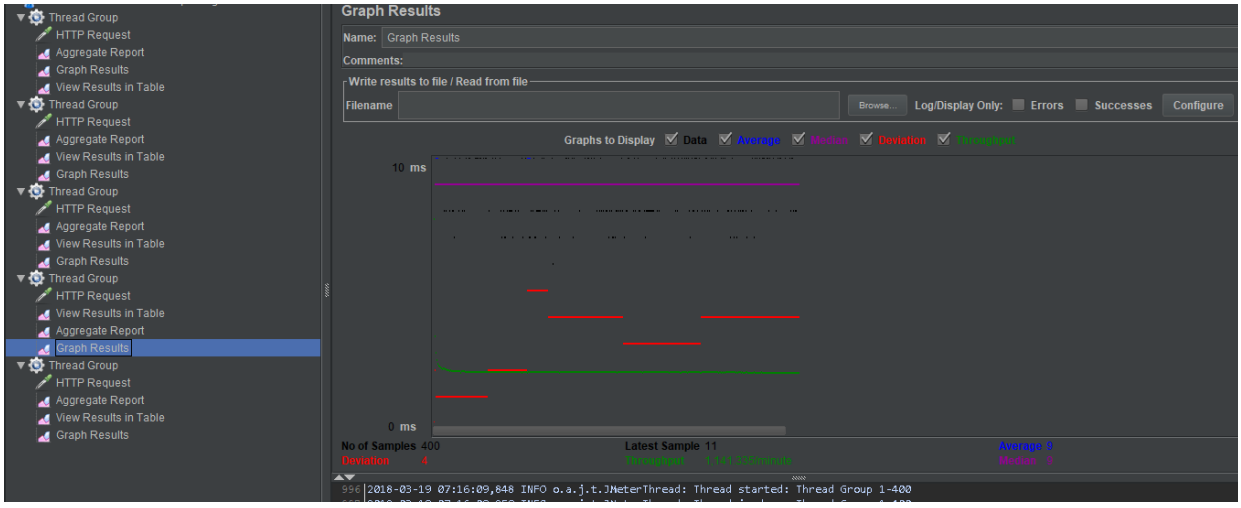
On invoking 400 concurrent users



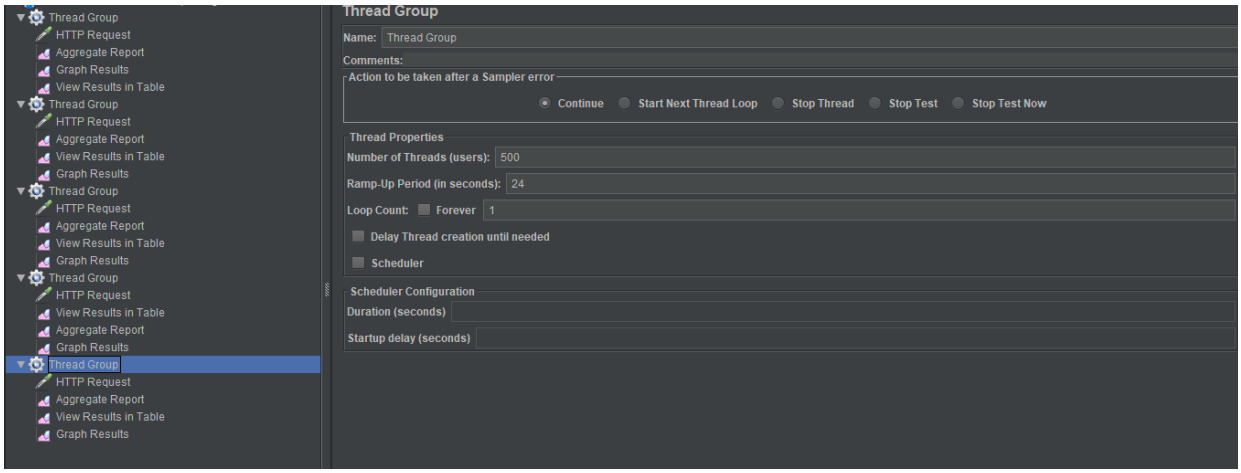


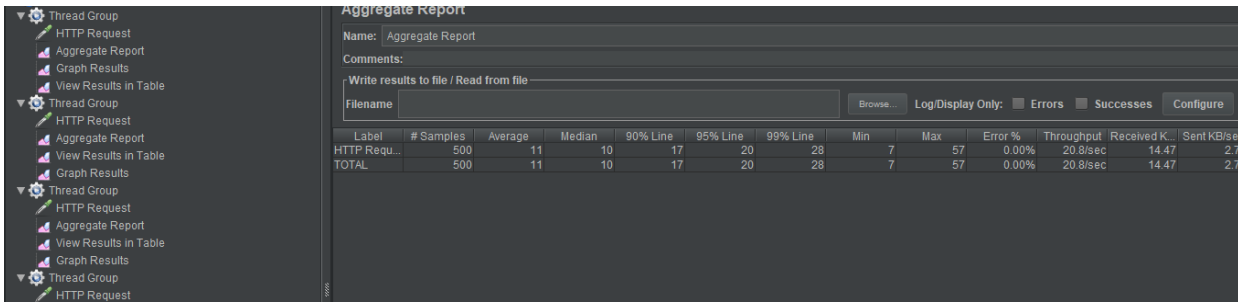


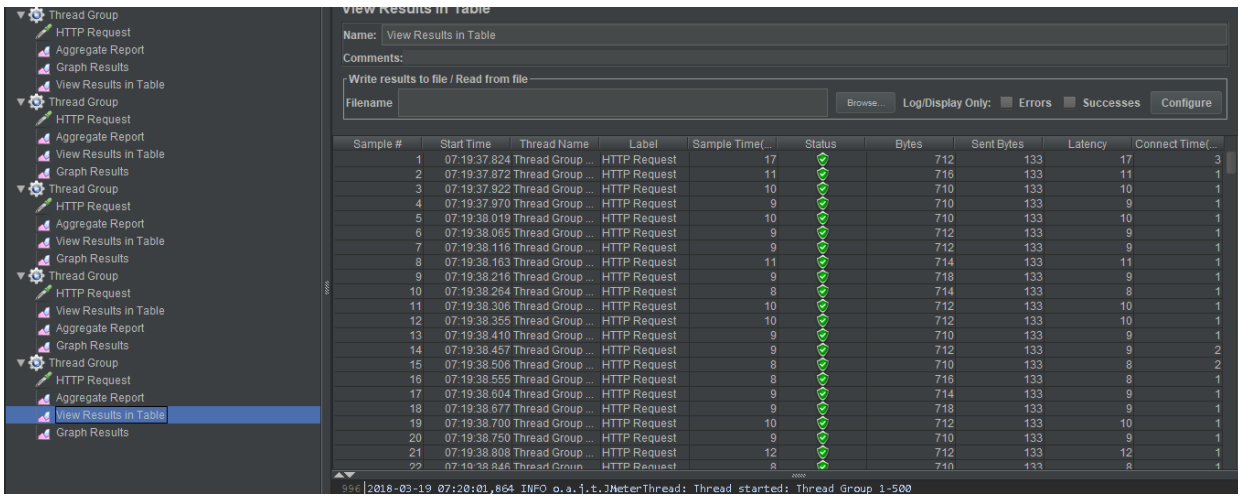


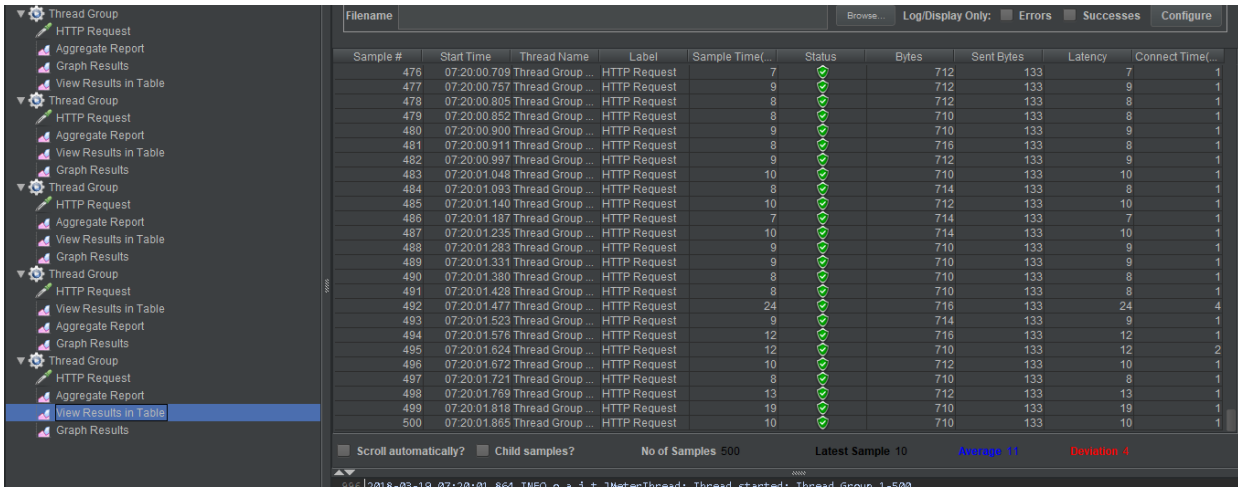


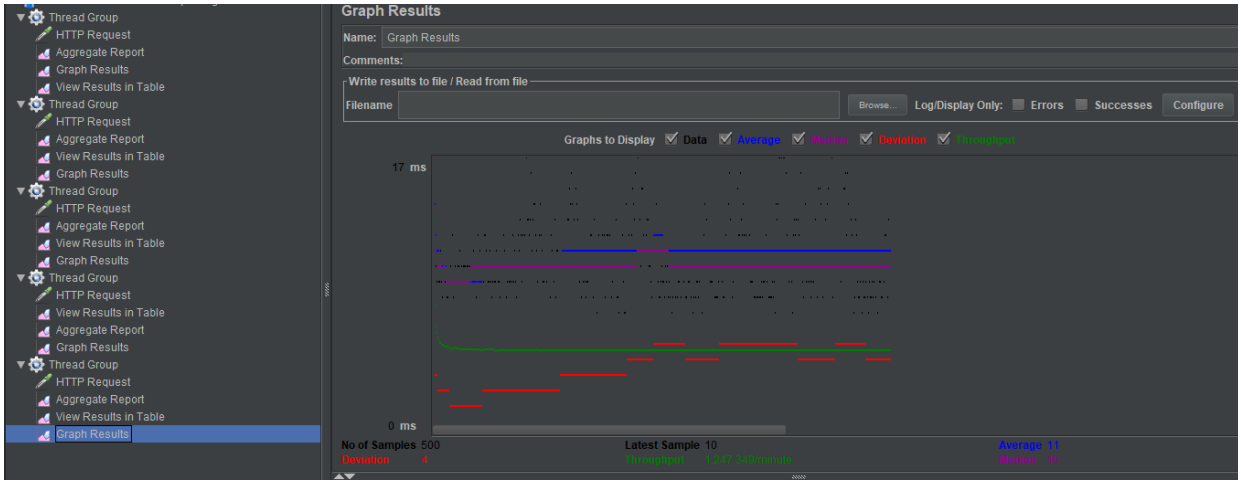
On invoking 500 concurrent user


Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
Name: Thread Group
Comments:
Action to be taken after a Sampler error
Continue
Start Next Thread Loop
Stop Thread
Stop Test
Stop Test Now
Thread Properties
Number of Threads (users): 500
Ramp-Up Period (in seconds): 24
Loop Count: Forever 1
Delay Thread creation until needed
Scheduler
Scheduler Configuration
Duration (seconds)
Startup delay (seconds)

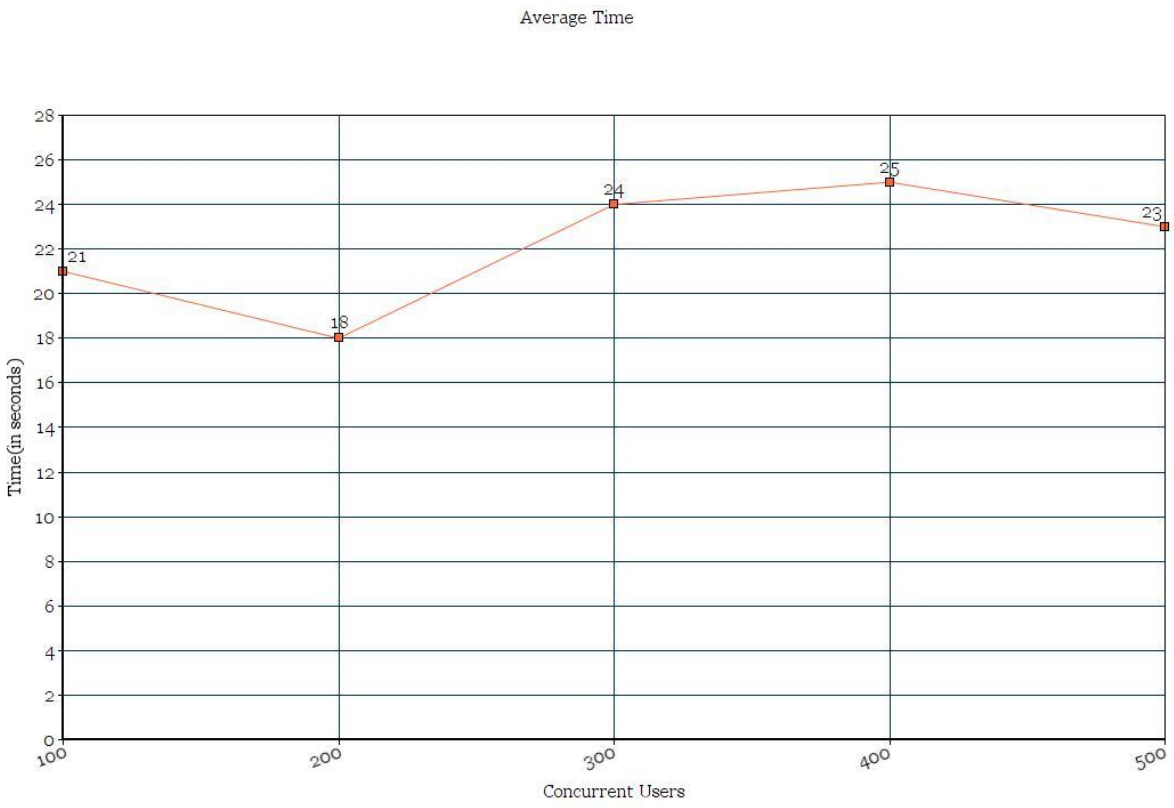

Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
Name: Aggregate Report
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Label # Samples Average Median 90% Line 95% Line 99% Line Min Max Error % Throughput Received K... Sent KB/se
HTTP Requ... 500 11 10 17 20 28 7 57 0.00% 20.8/sec 14.47 2.7
TOTAL 500 11 10 17 20 28 7 57 0.00% 20.8/sec 14.47 2.7


Thread Group
HTTP Request
Aggregate Report
Graph Results
View Results in Table
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Thread Group
HTTP Request
View Results in Table
Aggregate Report
Graph Results
Thread Group
HTTP Request
Aggregate Report
View Results in Table
Graph Results
Name: View Results in Table
Comments:
Write results to file / Read from file
Filename
Browse...
Log/Display Only: Errors Successes
Configure
Sample # Start Time Thread Name Label Sample Time(... Status Bytes Sent Bytes Latency Connect Time(...
1 07:19:37.824 Thread Group ... HTTP Request 17 712 133 17 3
2 07:19:37.872 Thread Group ... HTTP Request 11 716 133 11 1
3 07:19:37.922 Thread Group ... HTTP Request 10 710 133 10 1
4 07:19:37.970 Thread Group ... HTTP Request 9 710 133 9 1
5 07:19:38.019 Thread Group ... HTTP Request 10 710 133 10 1
6 07:19:38.065 Thread Group ... HTTP Request 9 712 133 9 1
7 07:19:38.116 Thread Group ... HTTP Request 9 712 133 9 1
8 07:19:38.163 Thread Group ... HTTP Request 11 714 133 11 1
9 07:19:38.216 Thread Group ... HTTP Request 9 718 133 9 1
10 07:19:38.264 Thread Group ... HTTP Request 8 714 133 8 1
11 07:19:38.306 Thread Group ... HTTP Request 10 712 133 10 1
12 07:19:38.355 Thread Group ... HTTP Request 10 712 133 10 1
13 07:19:38.410 Thread Group ... HTTP Request 9 710 133 9 1
14 07:19:38.457 Thread Group ... HTTP Request 9 712 133 9 2
15 07:19:38.506 Thread Group ... HTTP Request 8 710 133 8 2
16 07:19:38.555 Thread Group ... HTTP Request 8 716 133 8 1
17 07:19:38.604 Thread Group ... HTTP Request 9 714 133 9 1
18 07:19:38.677 Thread Group ... HTTP Request 9 718 133 9 1
19 07:19:38.700 Thread Group ... HTTP Request 10 712 133 10 1
20 07:19:38.750 Thread Group ... HTTP Request 9 710 133 9 1
21 07:19:38.808 Thread Group ... HTTP Request 12 712 133 12 1
22 07:19:38.846 Thread Group ... HTTP Request 8 710 133 8 1
996 2018-03-19 07:20:01,864 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-500





Average Time



Mocha test results for 5 randomly selected calls

```
> mocha


  √ checking if profile is edited (126ms)
  √ checking if profile is fetched
  √ checking if properties are returned (171ms)
  √ checking if answer is posted (93ms)
  √ checking if pic is uploaded (98ms)

  5 passing (519ms)
```

Question and Answers:

Q) Compare passport authentication process with the authentication process used in Lab1.

Ans:

In Lab1 , client sessions were used and nothing was used for storing sessions. Due to this session used to get destroyed as when the browser is closed. Even though this method was able to store user password and username in session however it is not scalable for many servers and all routes of REST API calls has to be authenticated again for user.

In Lab2, express sessions are used with passport local strategy with node js. Sessions are stored in Mongo DB. When user logs in, the session information is stored in mongo DB. After that it initializes passport with local strategy and authenticates the user. Now even if the tabs are closed and the application is open on other tab in same browser, user wont be redirected to login page. Instead he will be directly shown home page. Here Mongo DB is used to check saved username and password. Then it calls serialize method to attach user object which has user authentication information to session. Request is made for user session data with passport user data. Now for all other requests from browser, a session object with user is sent. Passport's initialize method is called every time for a request and user is verified. Passport authentication is independent of the server which receives request. Passport is very useful for acheiving scalability.

Q: Compare performance with and without Kafka. Explain in detail the reason for difference in performance.

Ans:

The performance of the application is observed to be slower without kafka than with kafka. It is happening because kafka guarantees fast message delivery for scalable framework. Message load balancing help for fast retrieval of messages using kafka. Kafka framework has topics, brokers , producer and consumers and zookeeper to manage message delivery and scalability. Zookeeper is sharing the state and is highly available if all brokers fails.

Q: If given an option to implement MySQL and MongoDB both in your application, specify which data of the applications will you store in MongoDB and MySQL respectively

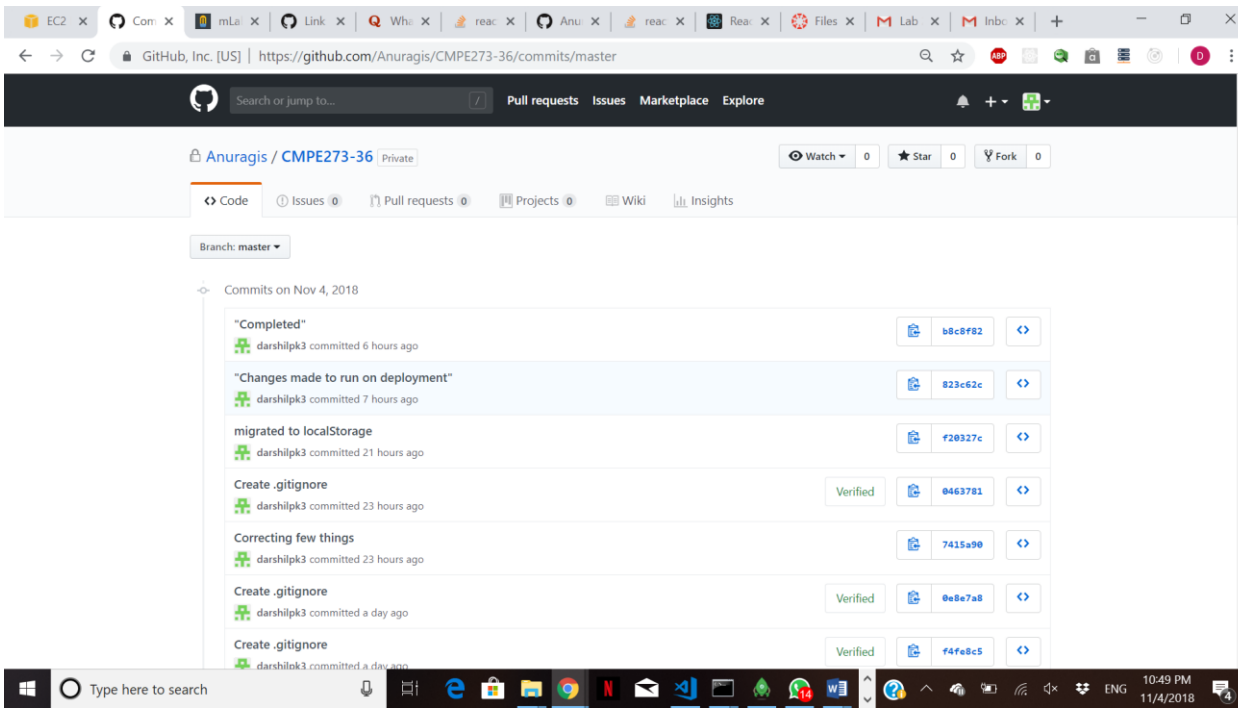
Ans:

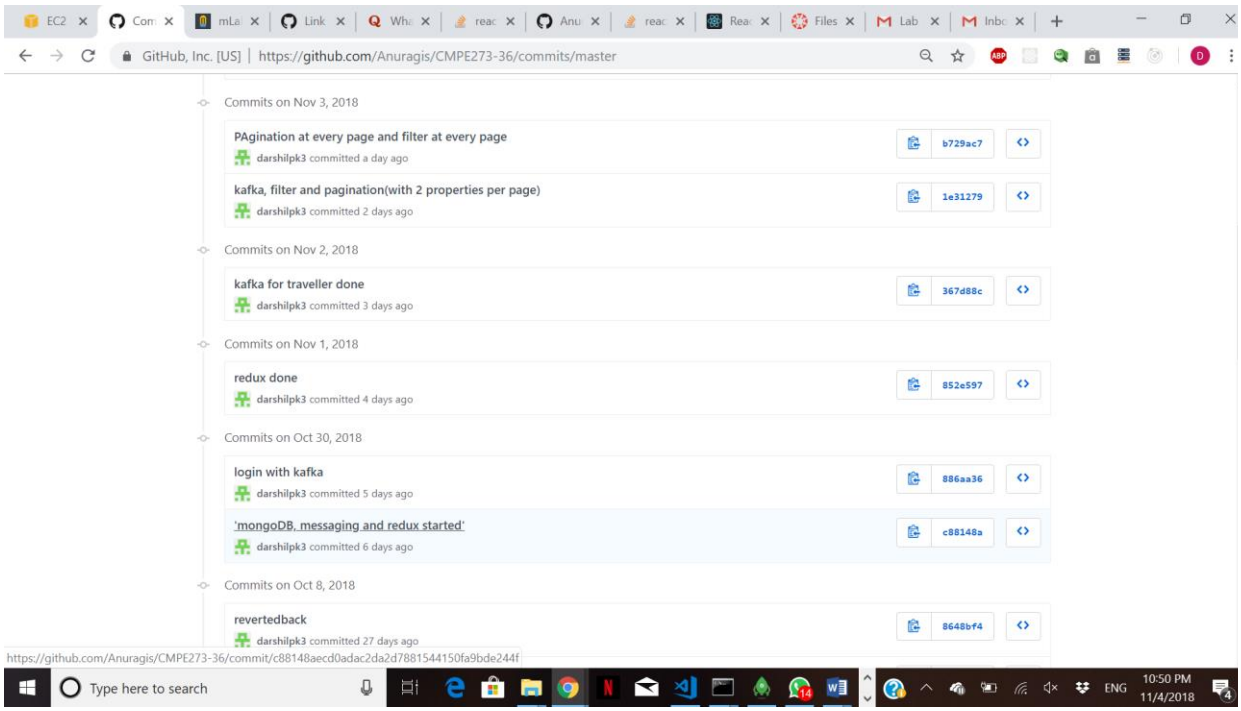
Both the databases, MongoDB and MySQL have their pros and cons.

My SQL is good for highly relational database. MongoDB is not an easy replacement for legacy systems that were built for relational databases.

Freelancer application requires multi-row transactions for several functionalities, thus better suited for a relational database.
On the other hand, MongoDB is suitable for environments with high write load because it supports high insert rate and doesn't have to worry about transaction safety, unlike MySQL which locks up the entire database and cause performance issues during insert operation. As the application contains multiple insertions and updates as user performs actions on mainly project data. Looking at this scenario, project, bid and transaction related data should be in MongoDB and User related data should be in MySQL, but this again creates problem when mapping of project and user data or bid and user data is to be done as data needs to be fetched from both, Mongo and MySQL.
Third and one of the most important factor is of scaling. MongoDB supports built-in replication, sharding, and auto-elections. Using auto-elections, you can set up a secondary database to automatically take over if the primary database fails. Sharding allows for horizontal scaling, which is difficult to implement in MySQL.
But as the application developed was small and need for good relational database is prominent, even with all above advantages of MongoDB I would choose to use MYSQL for storing all databases namely User, Project, Bid and Payment_history and only session data in MongoDB.

Commit History





Github Link: https://github.com/Anuragis/CMPE273-36.git

Extra - Requirement:
Owner with more than 5 bookings: email: d@k, password: 123
Traveler with more than 5 bookings: email: d@k, password:123

AWS account information:
Email: darshilpareshbhai.kapadia@sjsu.edu
Password: Maruti_800