Increment – 4

Property and Rental Management Application.

Authored by: Bhuvan Chennoju
Hemanth Sai Bandi
Manjusha Namburi
Shravya Sri Kanchi

Story-Problem Statement

In the beginning of the fall, guy named Rajesh moved from India to Kansas city for pursuing masters at University of Missouri-Kansas city. It's a new place and new country with nothing like home, and finding a place for himself was a daunting task for him. He has talked with local landlords, searched for apartments nearby, searched for the roommates in college, So after extensive search with his home journey, he ended upon on a website called mac apartments. This place is amazing, with multiple features to select and wide range of houses and apartments in the area. Although most of the details are provided, this site is long with a limited range, and prices are high for renting. So we have come too thought of building a web application template to mimic the website with room finder, tenant, and administrative people prospective.

References: https://www.macapartments.com/
https://www.zillow.com/

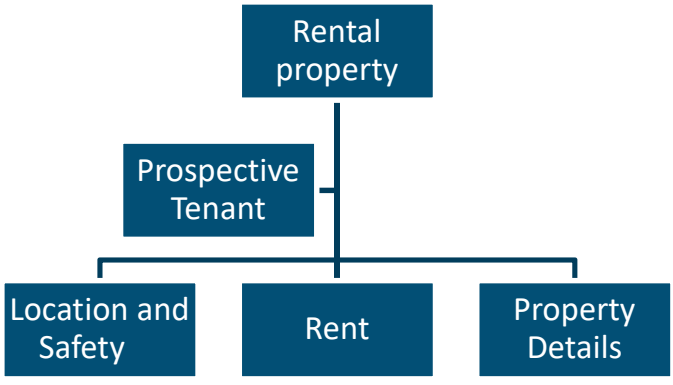
Let's Wear the thinking Cap:

“How to find a better home? What should it offer? How easy is it to get it? So many constraints!!”

Finding a new home is a lot of work! Get a suitable space, with comfortable mat area, bathrooms and most important thing affordability.

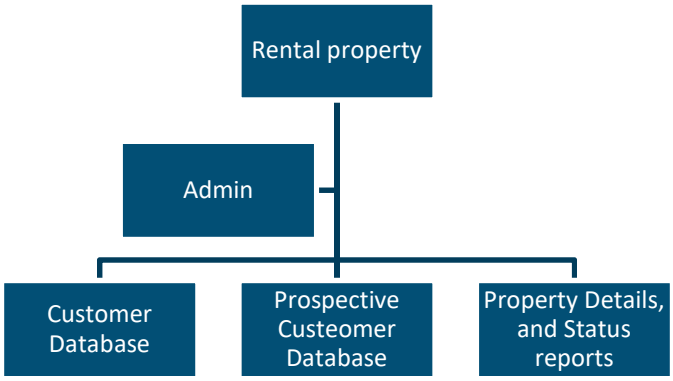
Thinking cap let’s explore what thing one need to make a better choice. Few things are mentioned in hexagonal wheel of thoughts. Let’s decide on what factors one look in a new home, it could be rental price, location the property, Safety of the property, and other details.

The user of the application could be anyone, student, working professional. Everyone has the different requirement in the searching the rental place. So what kind of features do we need to include in the application form new prospective customer side.



These three things any new room searching guy can look in a rental place.

“But what could an administrative people need in the other side of the application? This could be anything”

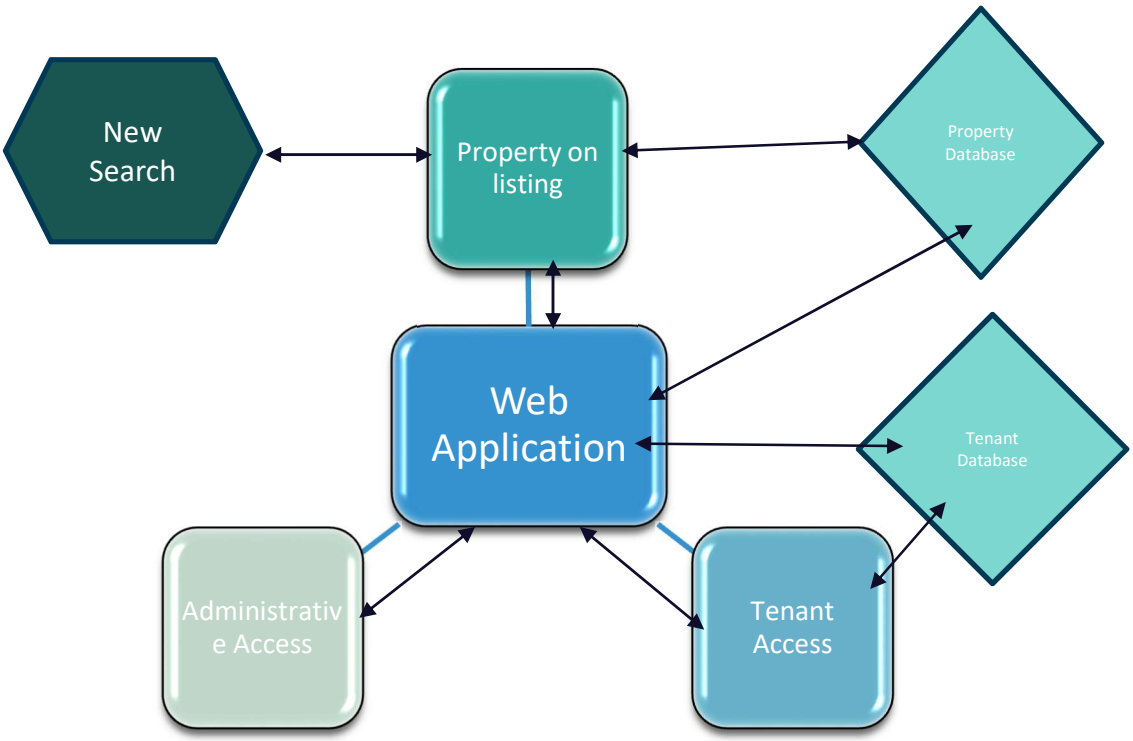


From the other side of the application admin staff need few things as shown in above flow chart. These things include both the parties’ databases, Contact information from the new customers, property status, and reports. By adding these webpages could help the admin staff to easy access to the application.

Solution to the Problem and Work Flow:

- We are going to create a work flow of the application with both tenant, and admin staff webpages
- Make easy available apartment listing with filtering in a given area with number of bedrooms, date of availability, and search button.
- Once filtered the apartments, we can get to application where a customer interest expression form. Once filling that it will send a request for customer login with admin staff in admin dashboard.
- In the Admin Dashboard, all the access control of customers, and data there.

Work Flow Map:



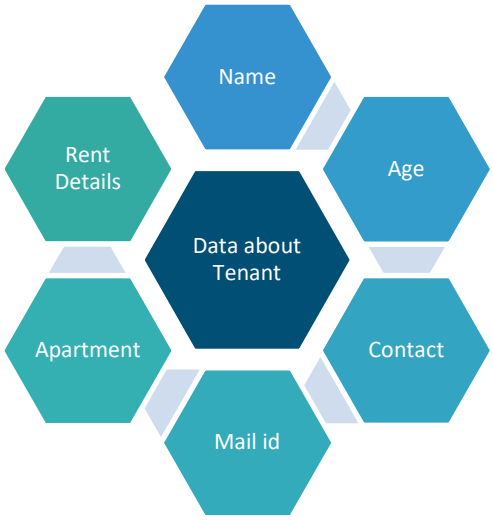
Technical Stack:

- **React -** For front-end website design we have used the React framework
- **Express and Node js –** For back-end design we have used the express and node js with axios to connect with front-end and back-end
- **Mongodb –** We have used the mongodb database to store the data.
- **Others –** Nodemailer to send mail to customer from admin access.

Data:

- In this application, we mainly have two kinds of data. First one is for the tenant data, which could be used by admin people to map a property with the tenant. Second one is for the property listing.
- Both the data will be stored in the mongodb database in the local machine, once the server is up and running, this data will be fetched to the website.
- Most of the data collection is manual and populated data with simple inputs.

Details on Type of Data



Source of Property Data:

- Property data has been collected from the Zillow, and mac apartment's sites.
- Some data has been collected with the manual scraping for few site with Beautiful soups with python
- Tried to collect data with Mashvisor API to get the rental properties data to fill the placeholders.

Source of Tenant Data:

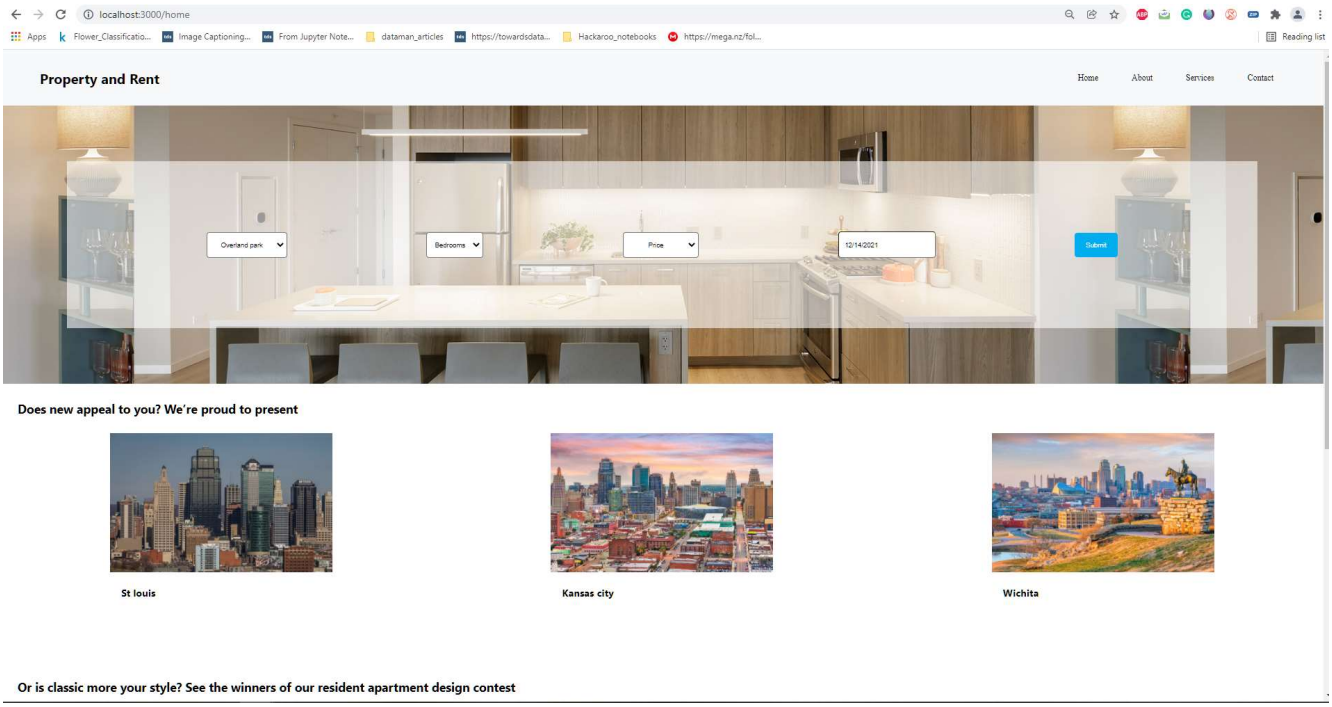
- Tenant Data is manually filled with hand full of collection form random names and addresses
- Realty mole is other API helped to some extent.

Features of the Web Application:

- Home Page
- Property Search
- Interest form
- Admin Dashboard
- Customer Dashboard

1. Home Page

Home page is the first page of the web application, and it shows some information with cites. In this page, we have put the search bar to select the number of bedrooms, available date for the place. Once search is initiated all the information about the available properties shows in the pages.



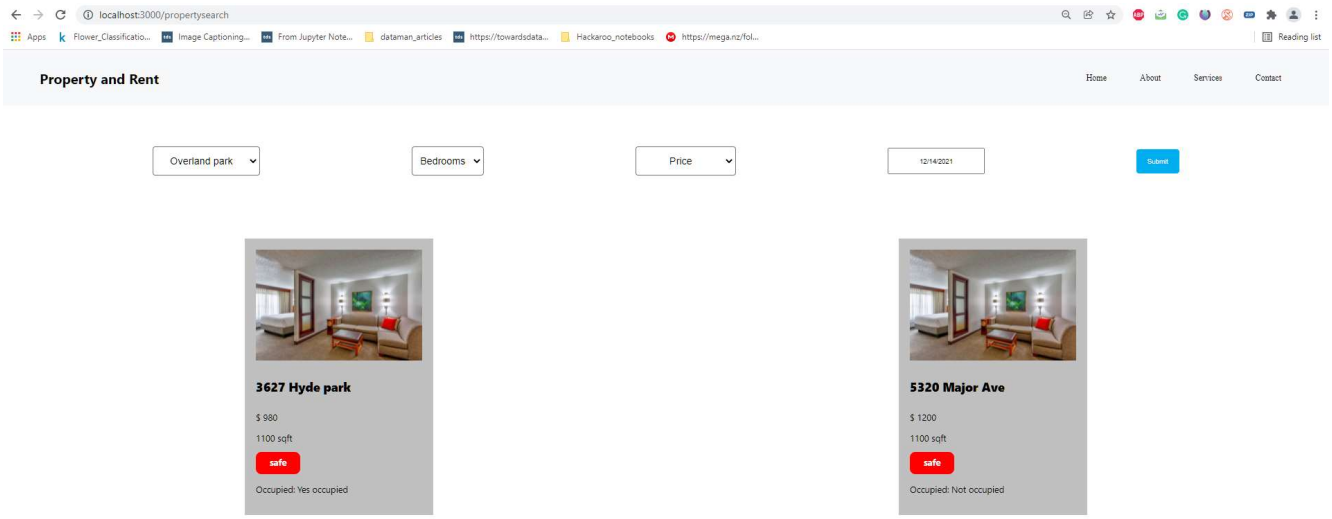
Code Snippet:

```
        <div className="homeSearchContainer">
            {/* {console.log(areaStreetData, 'areaStreetData')} */}
            <div className="homeSearchInsideBox">
                { (loadingArea) ? (
                    ''
                ) : (
                    (areaStreetData) ? (
                        <select name="street" onChange={(e) => handleChange('homesearch', e)} type="text">
                       { areaStreetData.map((v,i) => {
                            return <option key={i} value={v.name}>{v.name}</option>
                        }) }
                         </select>
                    ) : (
                        ''
                    )
                )}
                <select name="bedrooms" type="text" onChange={(e) => handleChange('homesearch', e)}>
                    <option value="--">Bedrooms</option>
                    <option value="0">Studio</option>
                    <option value="1">1+</option>
                    <option value="2">2+</option>
                    <option value="3">3+</option>
                </select>
                <select name="price" type="text" onChange={(e) => handleChange('homesearch', e)}>
                    <option value="--">Price</option>
                    <option value="600">Less than $600</option>
                    <option value="1200">Less than $1200</option>
                    <option value="2000">Less than $2000</option>
                    <option value="3000">Less than $3000</option>
                </select>
                <div>
                <DatePicker name="date" className="datepickerr" selected={homeSearchData.date} onChange={(e) => handleChange('date', e)} />
                </div>
                <button type="button" className="submitButtonHome" onClick={handlesubmit}>Submit</button>
            </div>
        </div>
```

This above code snippet is the helper for creating the filter in the property search

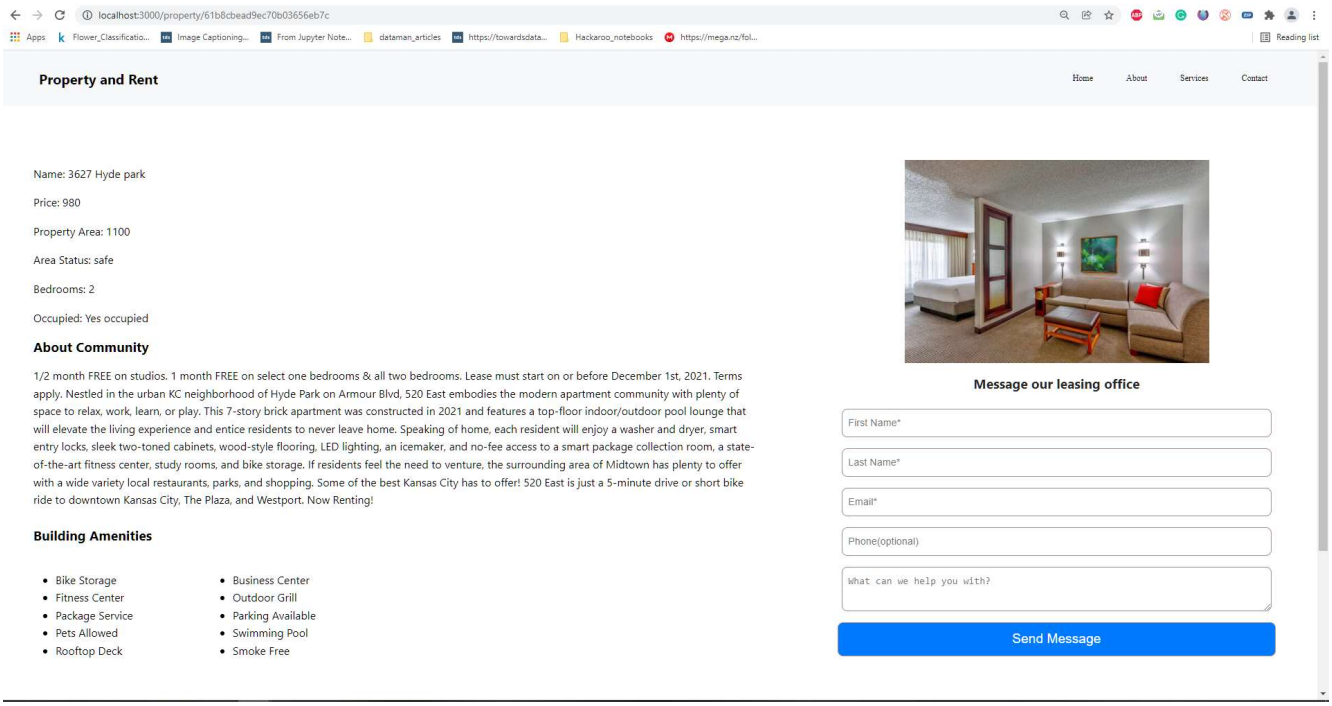
2. Property Search:

Upon the property selection, based on the data available in the database required apartments fit the description will show up.



3. Property Details and Interest Form

Once property is choose from the filtered search it will lead to the webpage with the property details and interest form. This Form will lead to backend data storage to mongodb database for customer database.



Message our leasing office

Bhuvan

Chennoju

bhuvankumarchennoju@gmail.com

Phone(optional)

What can we help you with?

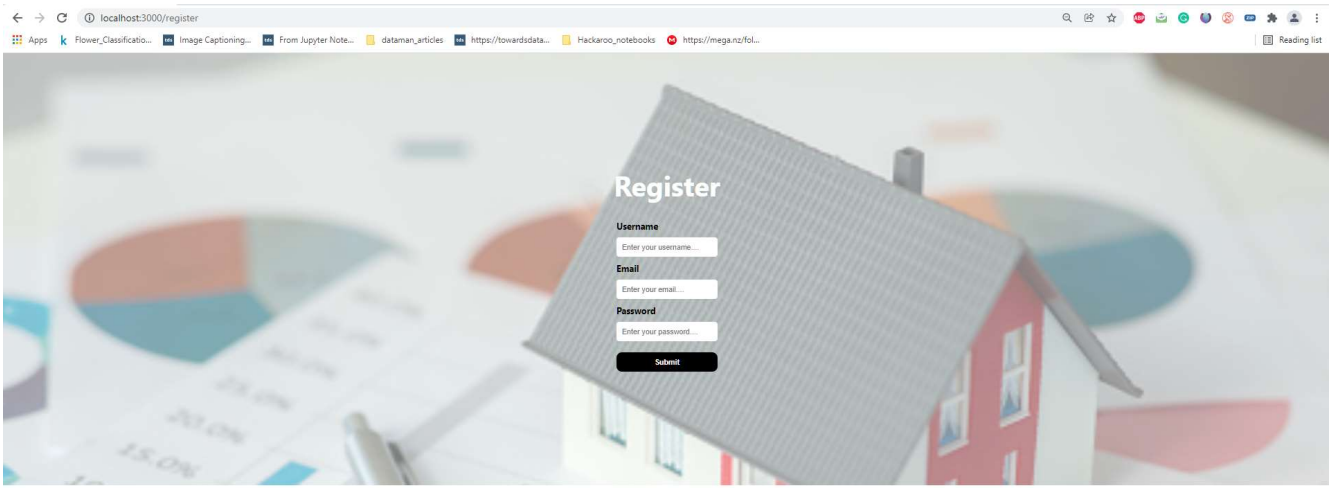
Send Message

Code Snippet:

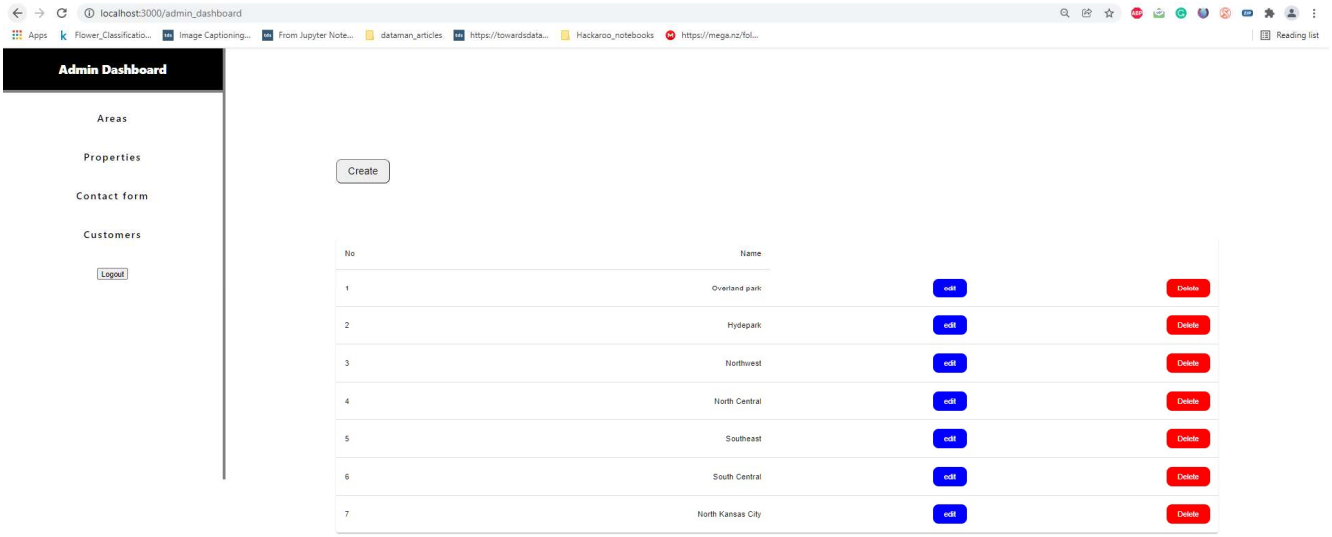
```
const mongoose = require('mongoose');

const Schema = mongoose.Schema;

const contactFormSchema = new Schema({
    firstName: String,
    lastName: String,
    email: String,
    phone: String,
    message: String,
    propertyId: String,
})

const contactform = mongoose.model('contactform', contactFormSchema, 'contactforms')

module.exports = contactform;
```

4. Admin Dashboard

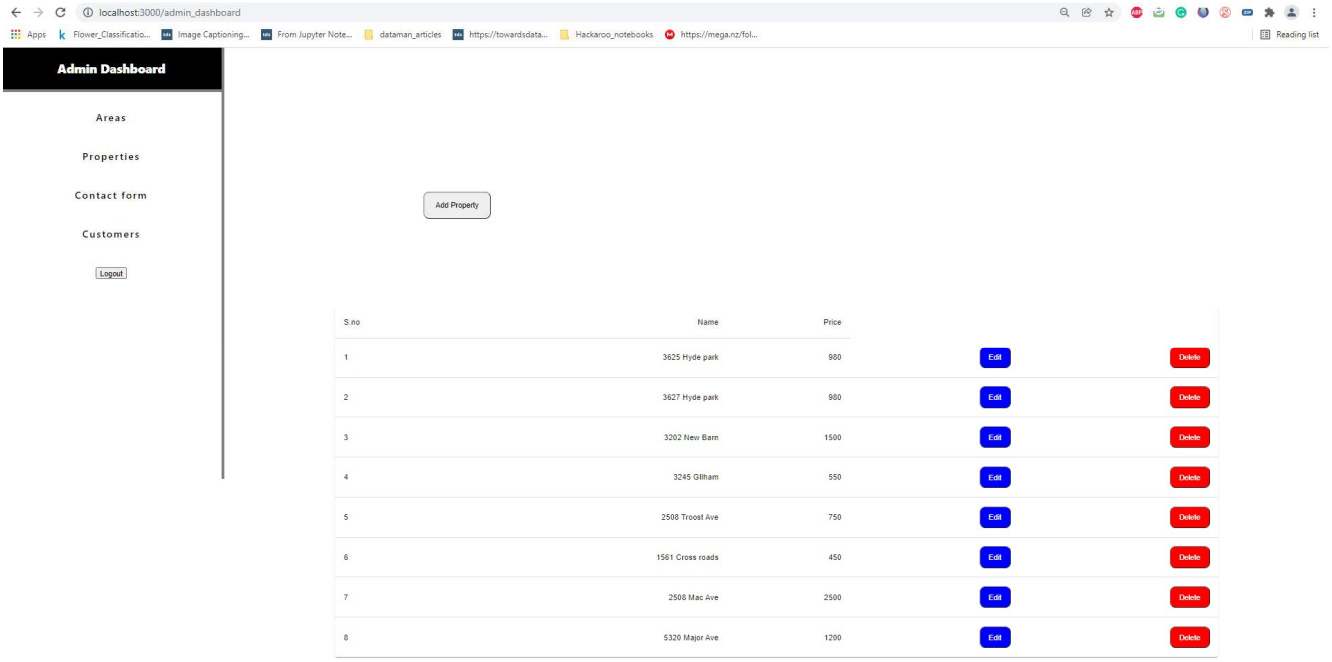
For the admin dashboard, first we have to register with valid credentials, once we register, it lead to login page. From where one can log into admin dashboard, where an access to areas, properties, customers, customer forms can be found.



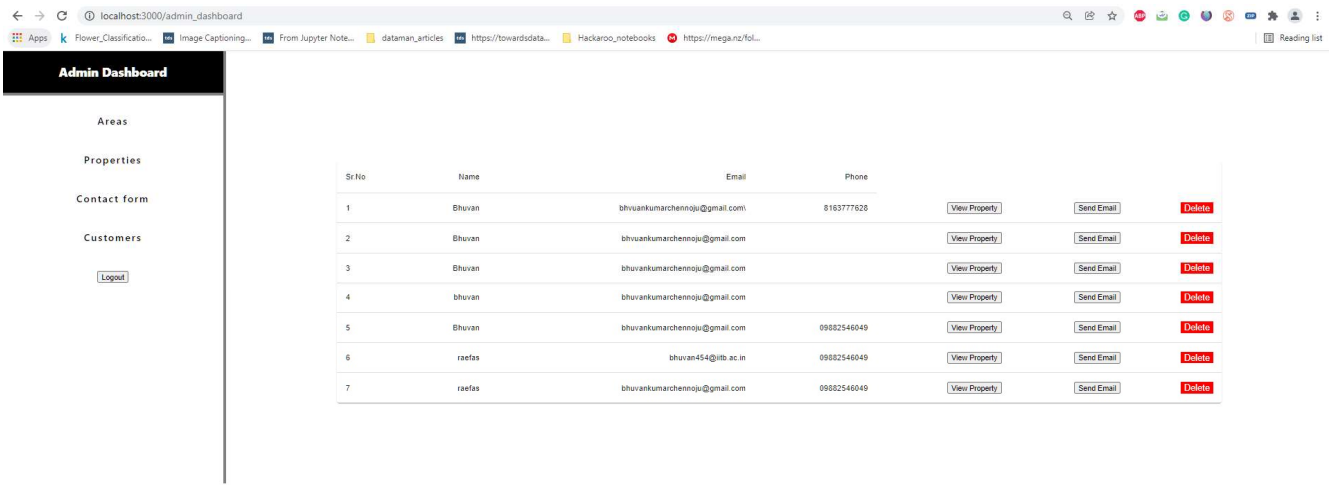
Areas: This is a feature where areas in the given region can be filled into the system.



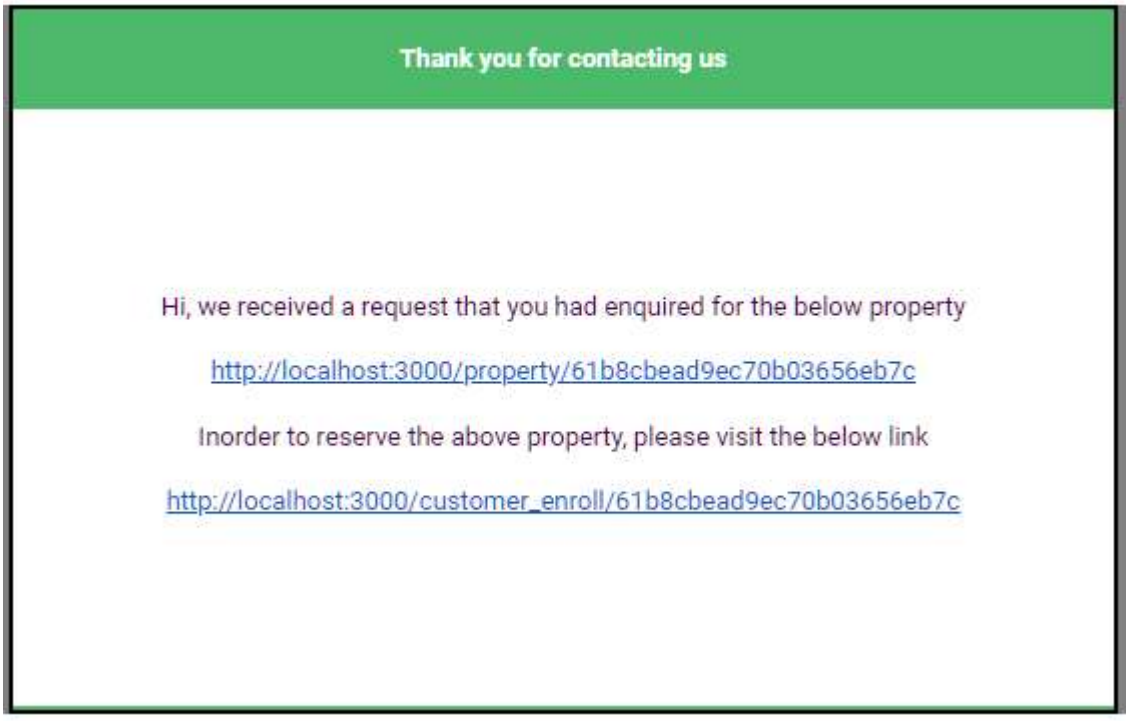
Properties: This is a feature where areas in the given region can be filled into the system.



Contact Form: All the contact forms from the property description will be stored here, from where admin can provide access to customers to become tenants.
From here with send email option a mail with registration link will be send to customer, with this link one can register to become customer.



Mail Received By Customer:



Code snippet:

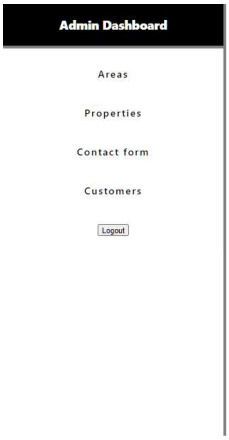
Nodemailer import:

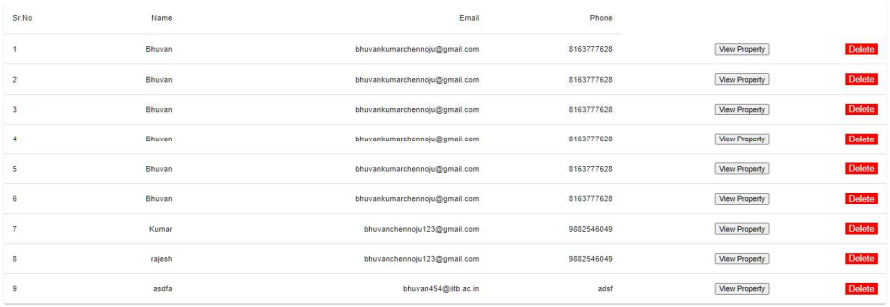
```
const express = require('express');
const router = express.Router();
var nodemailer = require('nodemailer');
const jwt = require('jsonwebtoken');
const Area = require('../models/areaModel');
const Contactform = require('../models/contactFormModel');
const Property = require('../models/propertyModel');
const User = require('../models/userModel');
```

Email Sending Code:

```
    var mailOptions = {
      from: 'webdevumkc@gmail.com',
      to: req.body.email,
      subject: 'Response to enquiry about property',
      text: 'Test',
      html: emailBody
    };

    transporter.sendMail(mailOptions, function(error, info){
      if (error) {
        console.log(error);
      } else {
        console.log('Email sent: ' + info.response);
      }
    });
})
```

Customers: Once the Customers register with the admin sent link, there detail will come to the customer database.

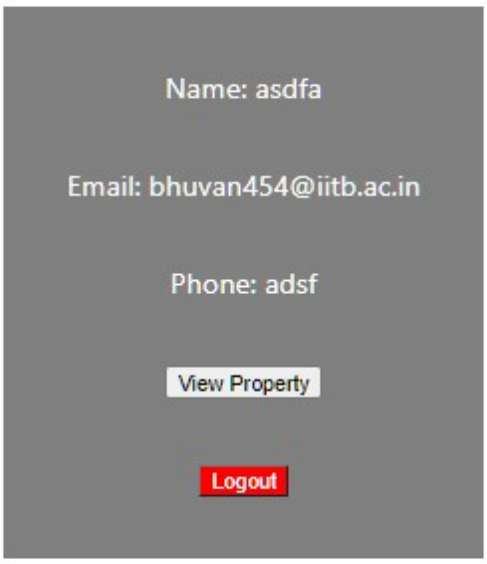




Sr.No	Name	Email	Phone		
1	Bhuvan	bhuvankumarchennoju@gmail.com	8163777628	View Property	Delete
2	Bhuvan	bhuvankumarchennoju@gmail.com	8163777628	View Property	Delete
3	Bhuvan	bhuvankumarchennoju@gmail.com	8163777628	View Property	Delete
4	Bhuvan	bhuvankumarchennoju@gmail.com	8163777628	View Property	Delete
5	Bhuvan	bhuvankumarchennoju@gmail.com	8163777628	View Property	Delete
6	Bhuvan	bhuvankumarchennoju@gmail.com	8163777628	View Property	Delete
7	Kumar	bhuvanchennoju123@gmail.com	9882546049	View Property	Delete
8	rajesh	bhuvanchennoju123@gmail.com	9882546049	View Property	Delete
9	asdfa	bhuvan454@iitb.ac.in	adsf	View Property	Delete

5. Customer Dashboard:

This is a unique customer dashboard with the login credentials, where customer details are mapped the property they are took for rent.



Challenges:

- Integrating API to automate the data filling.
- Creating an interactive UI with both Admin and Tenant dashboard logins for user friendly application.
- Few responsiveness issues when started developing the webpage, because of layout input.

Sharing of the Work:

Admin and Customer Dashboard : Bhuvan Kumar Chennoju
Property Search : Shravya Sri Kanchi
Property Details and Contact Form: Lakshmi Manjusha Namburi
Data collection & Testing: Hemanth Sai Bandi

What can we improve in future?

- API integration
- Interactive UI
- More Data addition

Github link:
https://github.com/bhuvan454/Web_and_mobile/tree/master/Project

Video Link:
https://youtu.be/pKIvhwvOjuA

Presentation Link:
https://github.com/bhuvan454/Web_and_mobile/blob/master/Project/Property%20and%20%20Rental%20%20management%20Web%20application.pptx

Code doc link:
https://github.com/bhuvan454/Web_and_mobile/blob/master/Project/code_file.docx

References:

- https://reactjs.org/docs/getting-started.html
- https://www.mongodb.com/languages/mern-stack-tutorial
- https://github.com/microrealestate/microrealestate
- https://realestatewealthnetwork.com/vacant-house-data-feed/