**JOB FINDING SYSTEM**

**Team Members:**

Sarika Reddy Kota

Pallavi Arikatla

Himaja Joginipally

**Story of the Project:**

Be it graduating students or others, will find it difficult while finding and fetching a job in the latest trending technologies. They tend to do a lot of research and get into numerous websites while applying for the jobs of their interest. They get into a confused state on which technology to learn and to pursue their career according to the current market situation. To get some information, they tend to their analysis by reading articles, from different websites, etc.. In order to avoid this exertion, we would like to build a web application which simplifies the difficulty by sorting out the jobs based on our skills and suggesting the technologies based on the current market trend.

The main objective of our application is to provide the current open positions based on company, location, and the technologies with which the user is familiar. It also provides highly paid positions based on the company and the location according to the current market.

**Work Flow:**

This project is to be implemented by using Angular in the front end and backend using express JS and Node JS. We will fetch the jobs data from “Indeed” using web scrapping and then using the Node js with the help of libraries like cheerios and Request. The Request library is used to make an HTTP request from the Node js application and the cheerios are used to get the data from the destination web page. This scrapping data will be sent to the frontend with the help of web API.

**Application Story:**

The project application is a design to help Job seekers to find a suitable job according to their interest and qualification

1.The targeted user of this application is a Person who is in search of a job

2.The application can pick some keywords from the uploaded resume and Can find the related job posting and display them to the user.

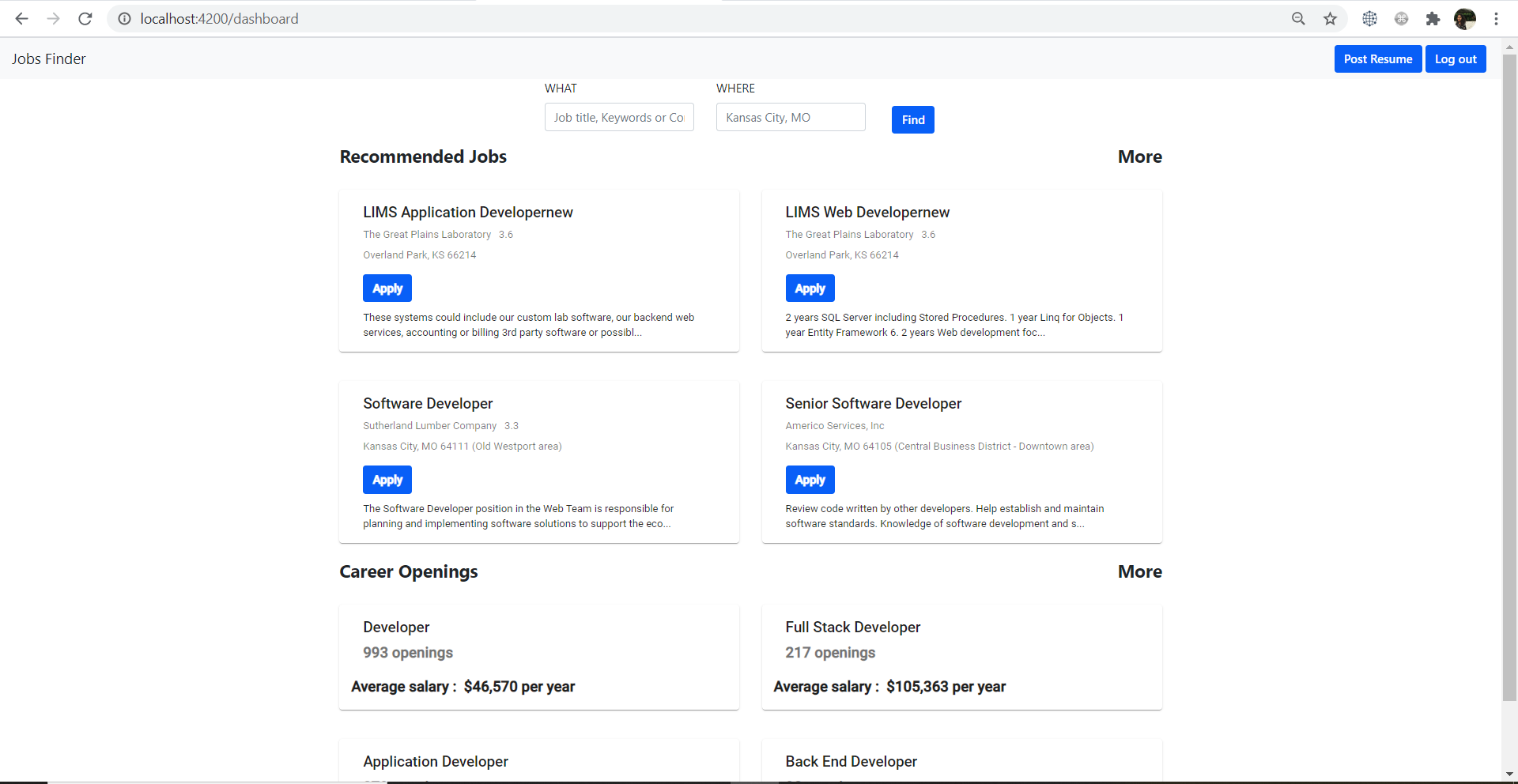
3.The user can use this application when they are in search of a Suitable job, This application is deployed on the web browser.

4.In this application, The user can upload a resume from where some keywords will be picked and Depending on that Suitable job lists, Their location, and Annual pay will be shown. The user will be able to apply for that job if that suits his interest.

5.Indeed website is taken as reference and by using web scraping, Job postings will be displayed to the user and

## **Detail Design of Features (using tools)**

### **User stories:**



## **Testing:**

We have done manual testing:

* We have verified the career advice page which will display the top paid salaries, Companies categorization, and Career opportunities based on the preferred location
* We have successfully tested the functionality of the job description which will display the details of the job.
* We have successfully tested the apply button in job postings which will take you to the actual job portal upon clicking on it.
* We have verified the API by uploading a resume from which some keywords will be picked and based on the Job postings will be shown.
* We have tested the apply button in the job testings which will direct user to the application website.

## **Implementation**

We have implemented the front end using Angular and backend using express JS and Node JS.

### **Front-end**

* In the previous increments, We have implemented the Registration Page, Login Page and Dashboard page.
* In this increment, We have added a component called job description which displays the details about the job in a model.
* We have added an apply button for the job postings.
* On the Careers Advice page, Job postings of the preferred location will be shown.
* Similarly, Openings, salaries and companies will be shown.
* The companies tab will have details about top-rated companies and their career growth.
* Salaries tab displays which show the highest-paid companies based on the location.
* The services like Job, Resume Upload were implemented to get data from the back end and show related information.
* Bootstrap and angular material technologies have been used to accomplish this.
* Forms are implemented using Reactive forms in Angular and validations are done.
* User can redirect to dashboard page whenever he clicks on Enterprise Trajectory.
* UI is completely responsive.

### **Back-end**

### The information on the job posting is fetched by using web scrapping from indeed using Node JS with the help of library cheerios and requests.

### API’S like Job API, Job description API, Openings API, company’s API, Salaries API were used to get related data based on the preferred location.

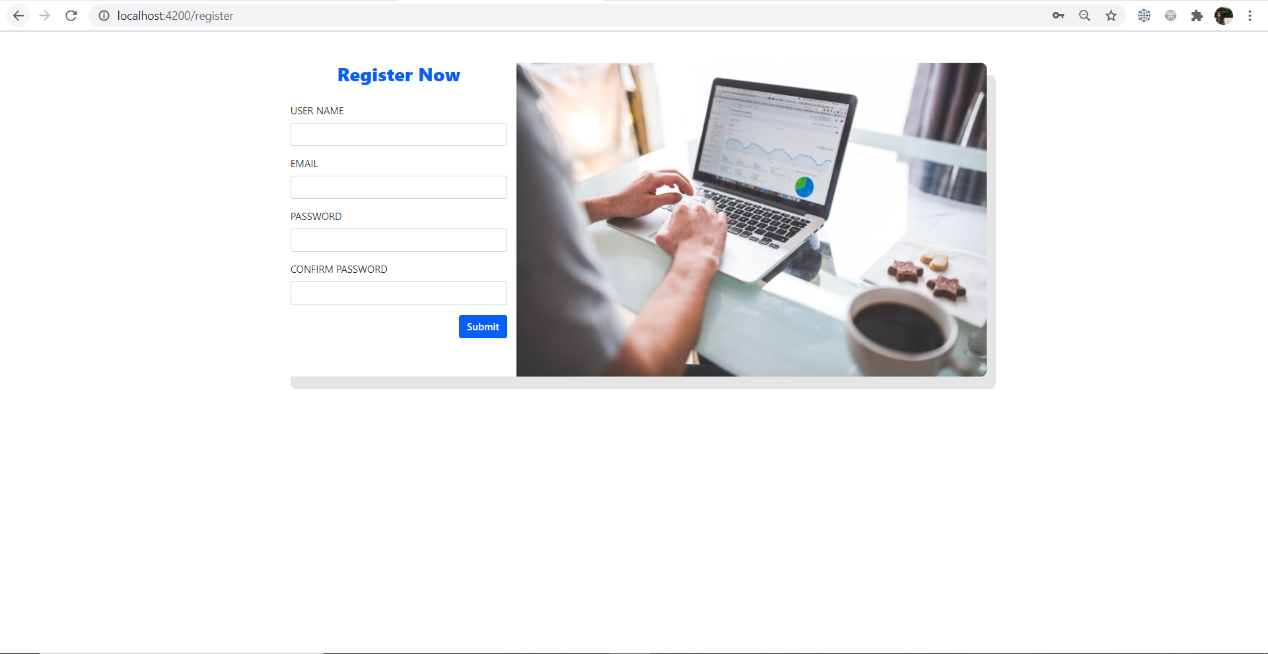
### On the dashboard page, We have created a Job API to get job data and a job description API to get a description of the job from the indeed website.

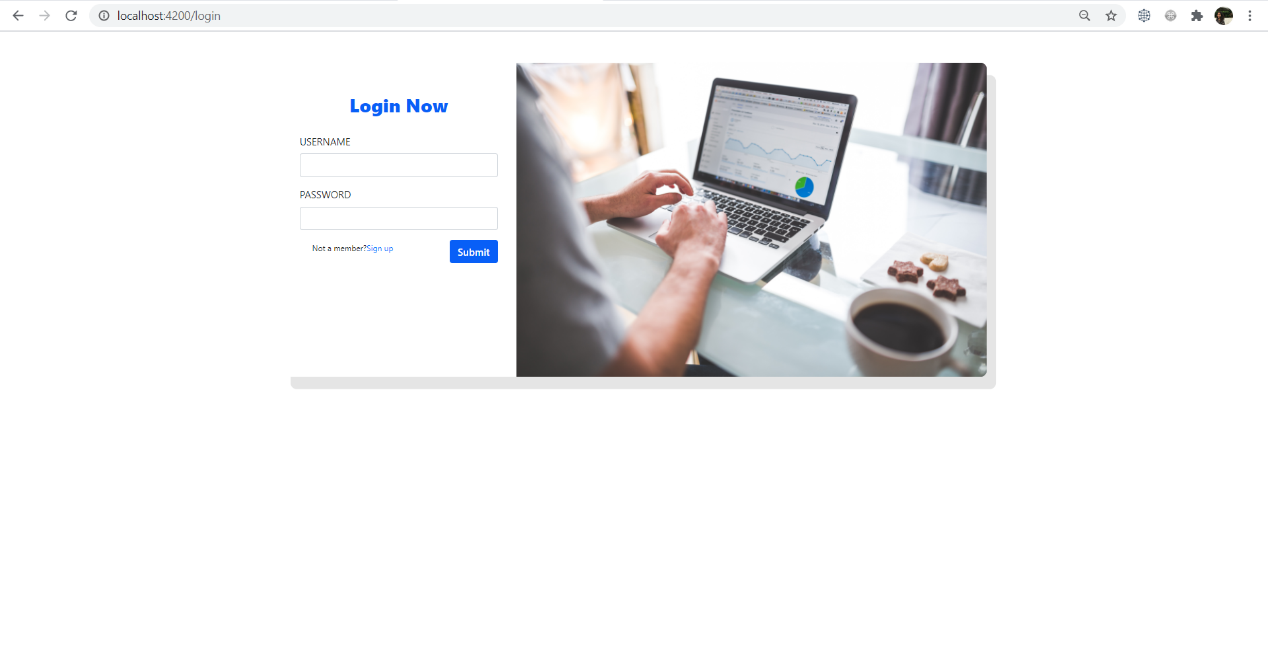
### On the career advice page, Opening API gets the job opening available in the preferred location.

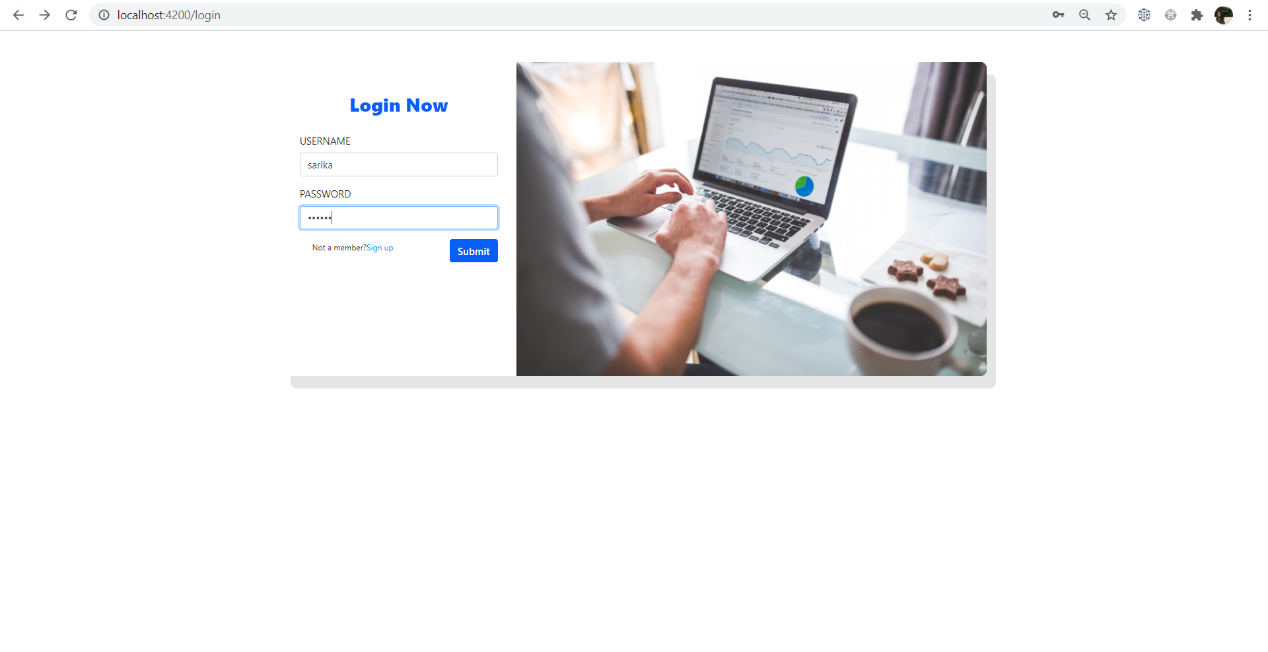
### Salaries API gets the highest-paid companies and respective salaries.

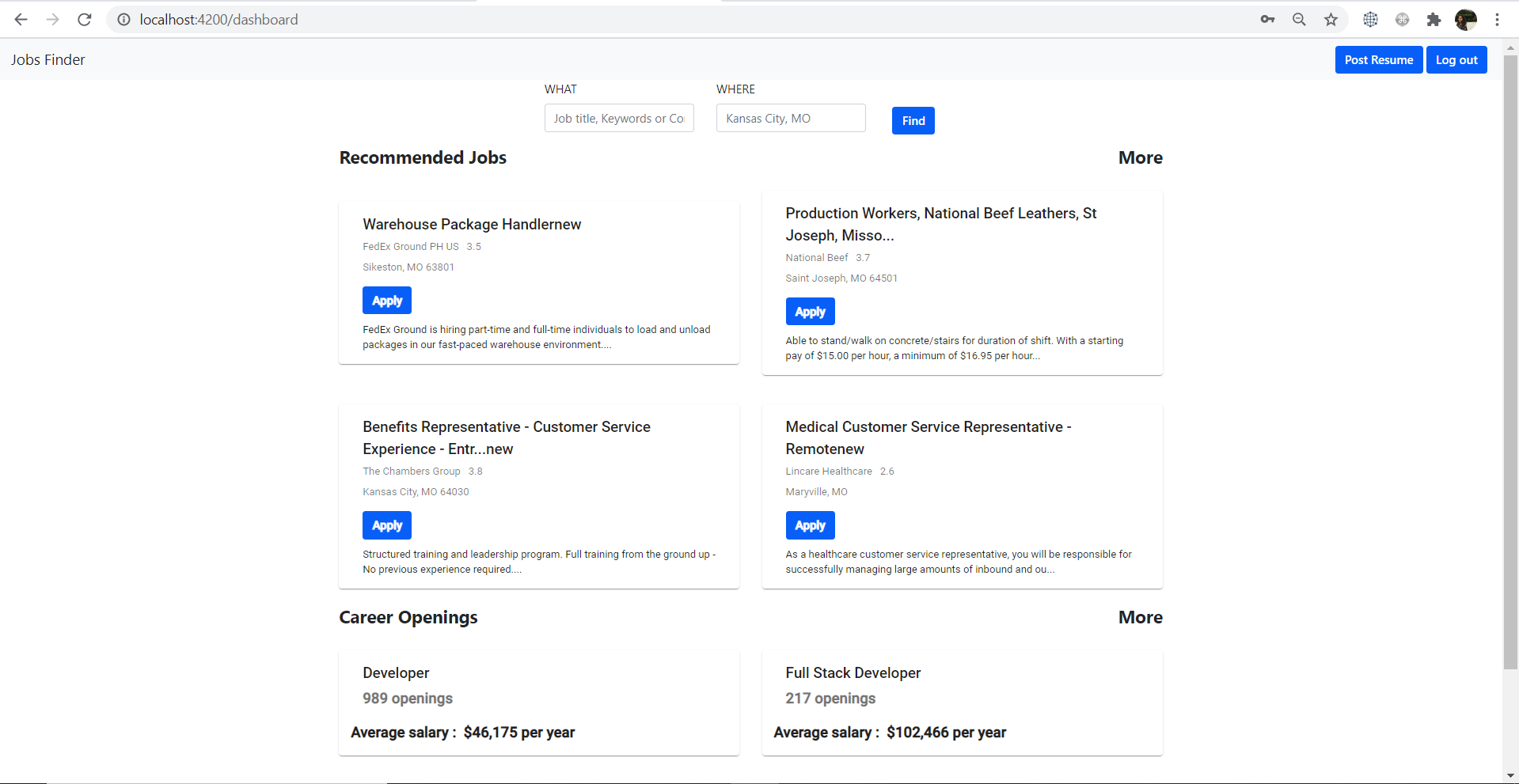
### Companies API gets the companies data in a structured way i.e. which companies are best for career growth and provides the best working environment etc.

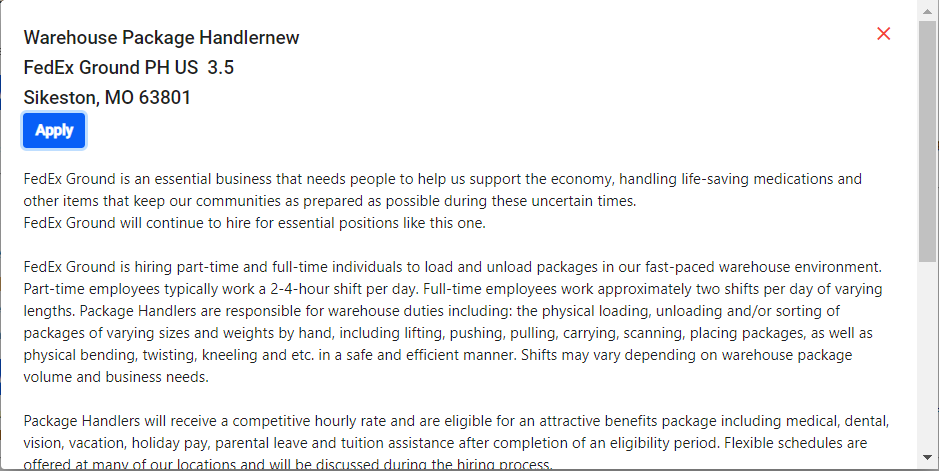
Working Screens:

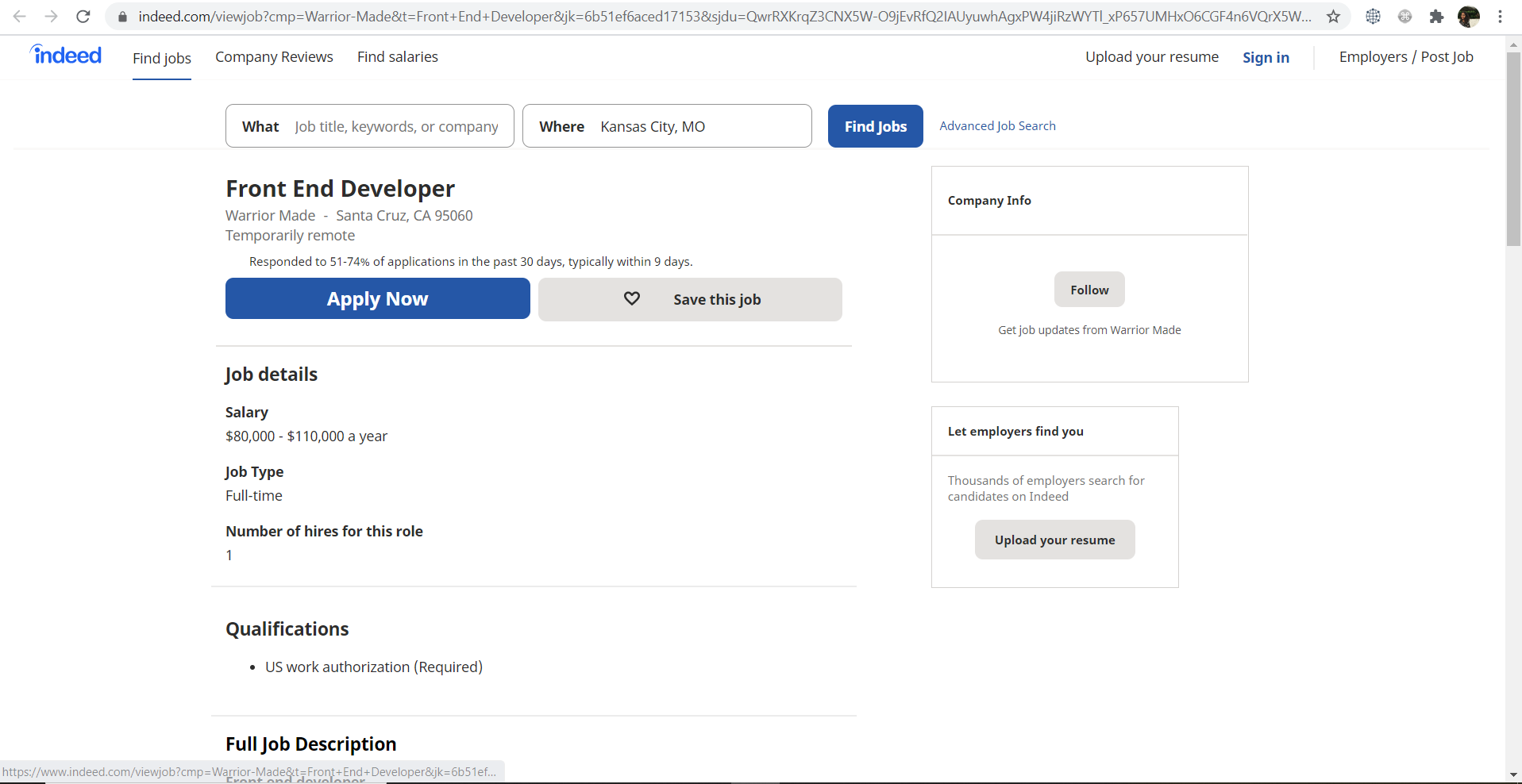


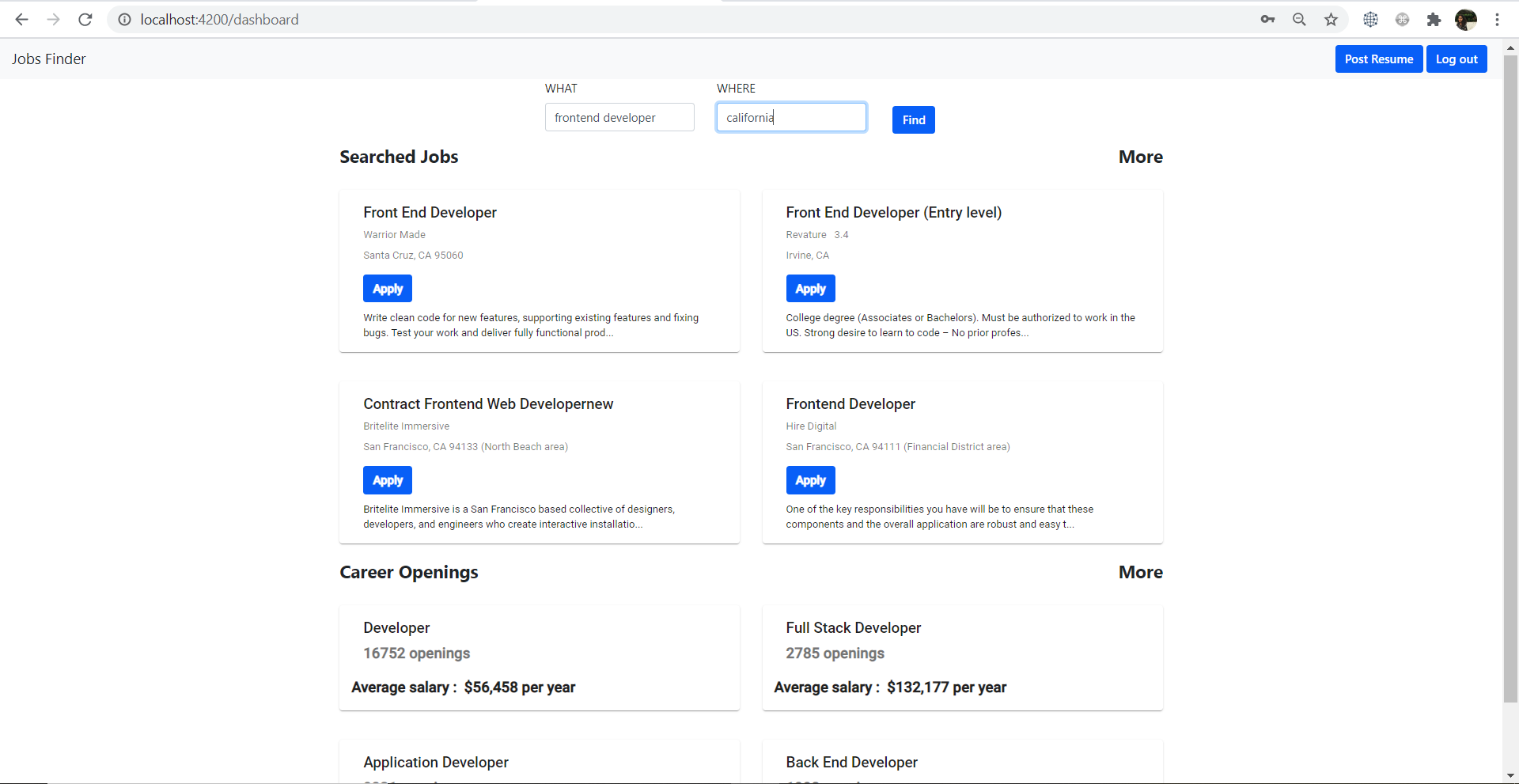


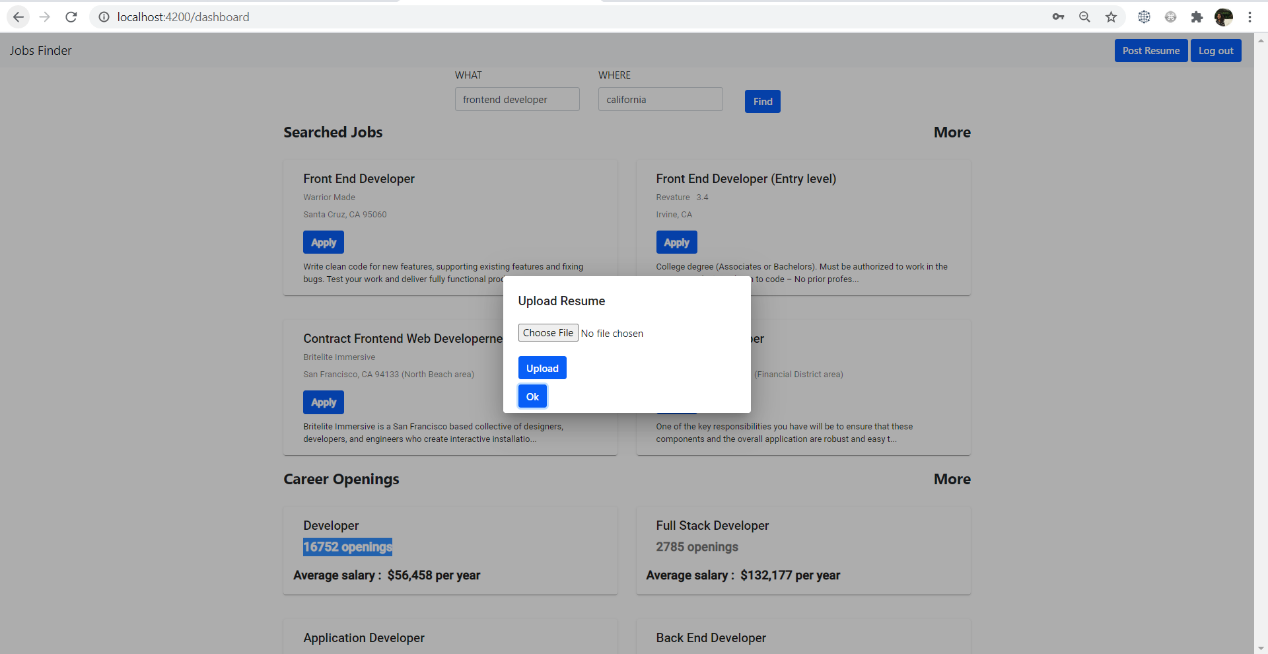


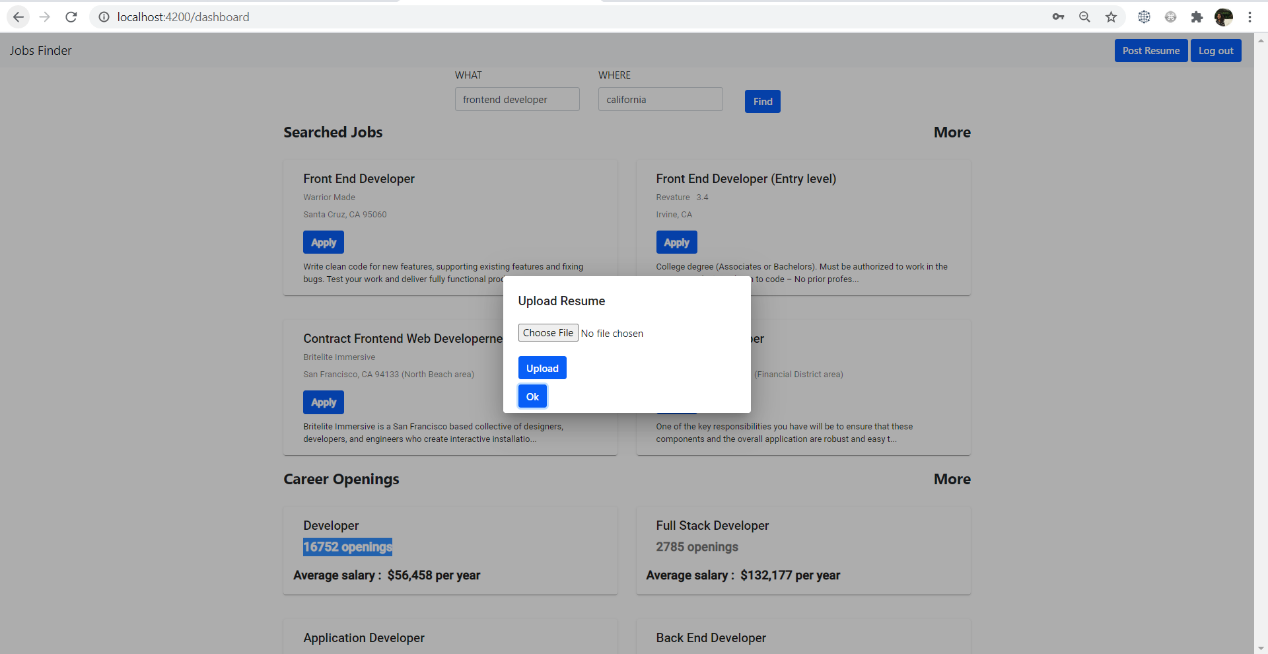


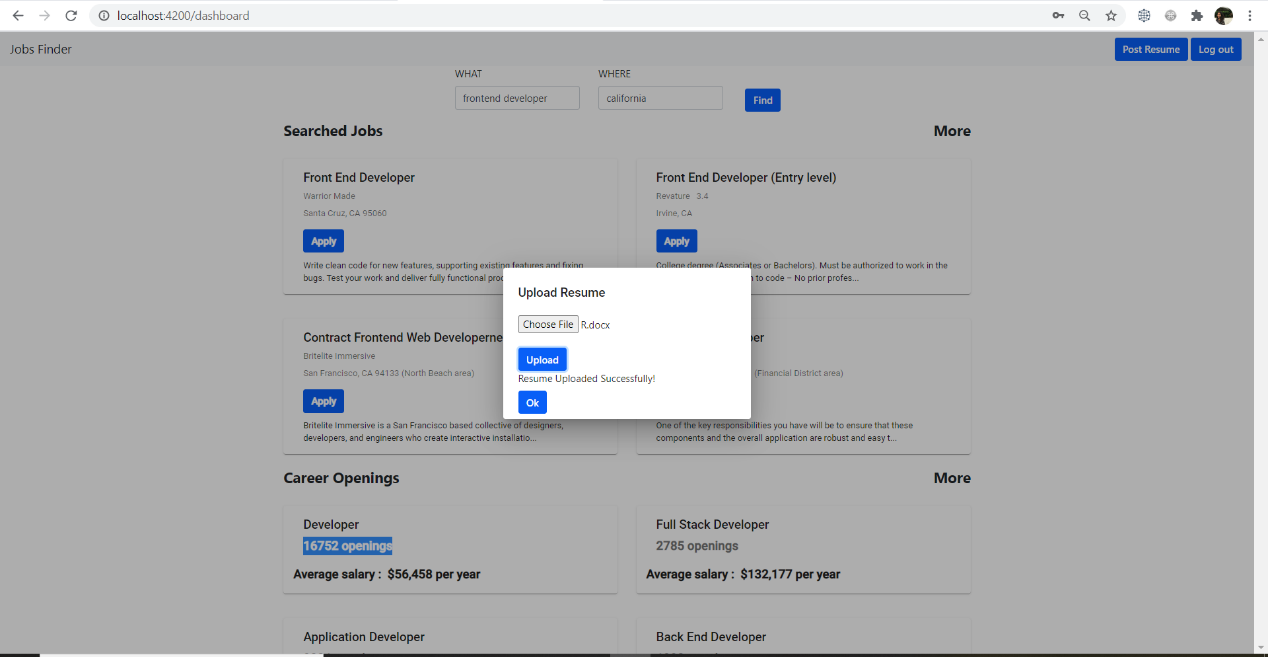


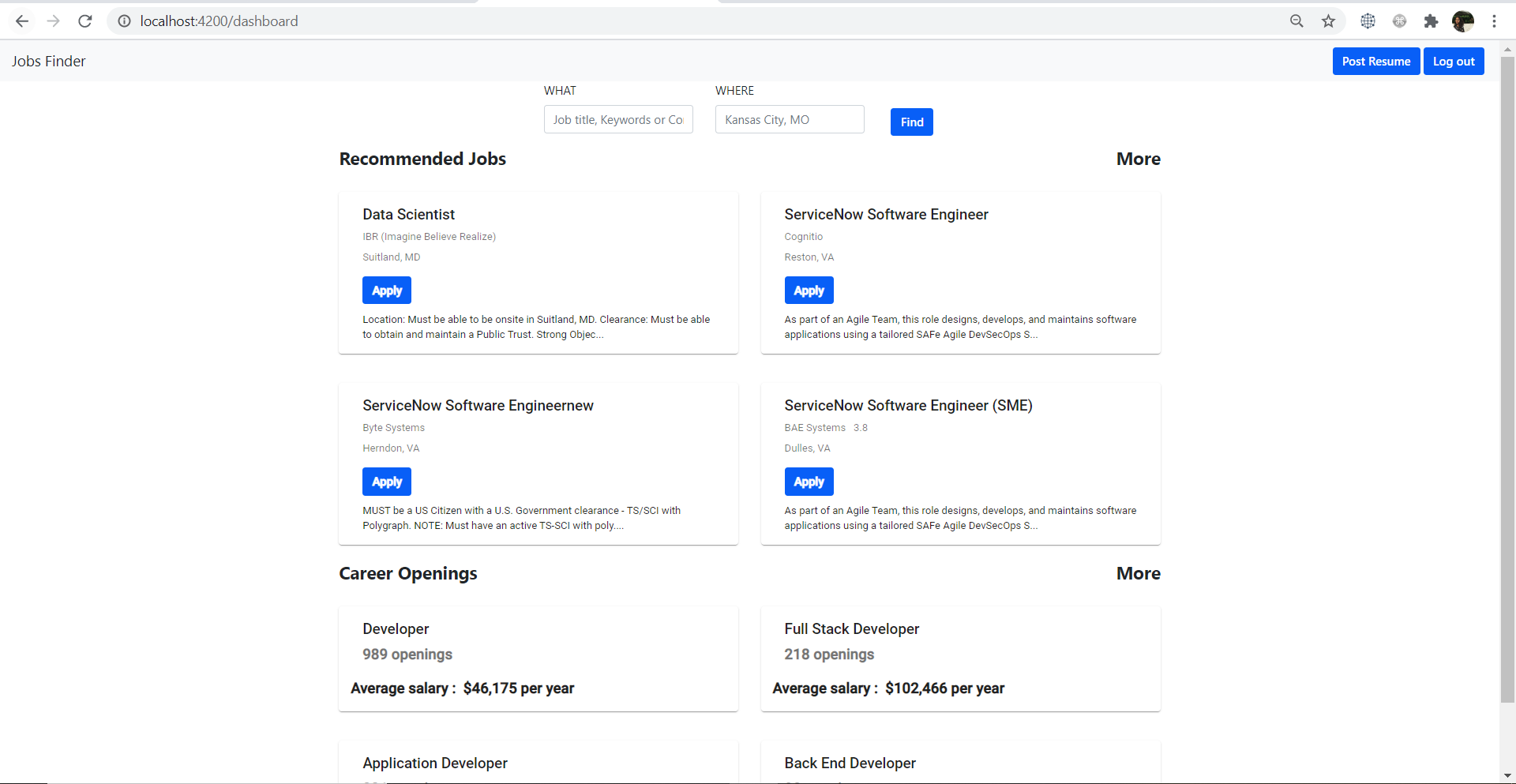












## **Existing Design/Services/REST API**

We have built custom API’s for registration page, login page and dashboard page (Job API, Job Description API), and Career Advice page (openings API, Companies API, Salaries API)

### **Work completed**

### Web scrapping, Web API creation in express- SarikaReddy Kota

### Front-end Angular services, Styling, Responsiveness - Pallavi Arikatla

### Job Description modal, Resume upload, documentation- Himaja Joginipally

**Issues and concerns**

* In order to get the dynamic web scrapping we faced the issue in forming the URL which is overcome by sending the request URL from express which will formed based on the user input.
* Faced issue when trying to manipulate the reference object we faced issue with shallow copy in javascript we resolved by making the deep copy of that object.

## **Bibliography**

* <https://www.indeed.com/career/salaries>
* <https://www.indeed.com/companies?from=gnav-title-webapp>
* <https://www.indeed.com/?from=gnav-acme--discovery-webapp>
* <https://github.com/cheeriojs/cheerio>
* <https://medium.com/bb-tutorials-and-thoughts/angular-understanding-angular-lifecycle-hooks-with-a-sample-project-375a61882478>

Github link:

<https://github.com/SarikaReddyKota/WEB_Project/tree/main>

Video link:

<https://www.youtube.com/watch?v=-nzphSHHPbI&feature=youtu.be>

Presentation link:

<https://github.com/SarikaReddyKota/WEB_Project/blob/main/Fianl%20ppt.pptx>