SOFE-2800U
CRN-43993
Faculty Of Engineering And Applied science
Web Programming - Final Report
(Fall 2022)

David Hanna: 100828635
Hasan Khan:100820450
Syed Nasir : 100809447
Marwan Alam: 100842087

Introduction

Our group consisted of David Hanna, Hasan Khan, Marwan Alam, and Syed Nasir. Our group's objective was to create a dynamic and open response website that allows clients to be at the forefront of recruitment, by giving them multiple built-in accessible features that make job searching or posting a lot easier. We believe that we have truly accomplished this with JetLinks. With JetLinks it has never been easier for people who are either seeking employment or employees, to find the right person for the job.

Overview

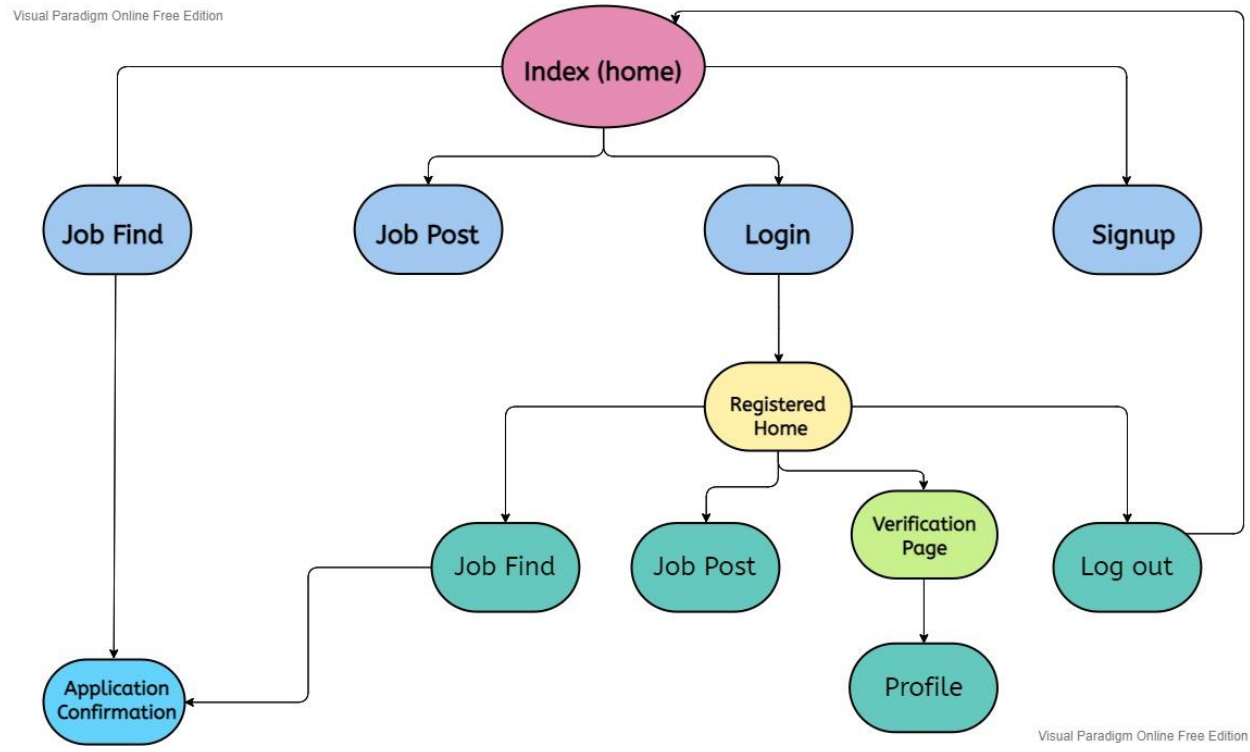
A job recruitment website that can be broken down into 6 parts, such as the Main page, Subpage, Design and styling, Information and content, Menu and navigation, and finally the Header and footer page

Outcomes and Goal

So the primary goal was to help job seekers to find their desired dream job with ease of access. With JetLinks job seekers are able to find their desired job, due to our constantly updating GUI web pages that match industry standards, and ensures the employer that he/she is receiving the fittest person for the position. Another thing that helps JetLinks stand out is the ability to submit a resume or a job without constantly having to sign in. with JetLinks security is never a concern. Our database ensures that all our client's personal information is secure and is not accessible to anyone. A goal that JetLinks is looking to achieve in its promising future is to get all employees or employers to use and build trust with our website when they are looking for further recruitment.

Our secondary goal is to simplify the recruitment process for the recruiters as it helps candidates to recruit and keep track of your information when you apply directly through our website. One way we went through this is we added a page for posting jobs for the recruiters to make it easier for the candidate and the recruiter team by clicking apply and the recruitment team will get a notification that someone has applied to the job ad posted. We did that through a secure database that retrieves the information that was inputted by the candidate, and as for the recruiter we put an outline that helps the recruiter post the required information for the job.

Site Structure



Technical specifications

The webpage can be divided into 2 major sections - Front-end and Back-end.

Front-end

Consists of HTML web pages under 'HBS' templates with content and respectively connected to a CSS styling sheet and designed in such a way that the backend is easily able to connect with the Front-End . The HTML pages are developed until we reach the planned template with help of CSS. All pages were given a similar navigation toolbar at the top as well as the footer at the bottom. Both of them contained links connecting to the rest of web pages from any other webpage. We further strengthen the website with addition of data and information along with constructing in such a way that it has ease of access for both the company and job seekers. In a specific web page, we gave companies the ability to post jobs with the help of a form which is connected to our dedicated backend database where it stores the information given by the client . And respectively rely on another webpage, where the information is being fetched from the database on to it. Thus allowing the job seekers to see the posted job by the companies and are able to quickly apply with a click of a button. Furthermore, A dedicated Sign up page and correspondingly a login page are all interlinked with the robust backend database server. Moreover your profile can be changed once again with the 'profile and setting' web page once logged in, giving users complete control over their account and settings.

Back-End

We have selected Node.js as our primary backend language, along with Express as the framework and MongoDB as the database server. The whole project is run as an API on specified localhost ports. To make communication between HTML pages and the main Application json, we converted our HTML files to HBS extensions, which allowed us a similar functionality as Php.

The application is built to call and post all HTML files as well as Schema Model files for MongoDB. Here's a list of dependencies used by us in Nodejs:

- Nodemon - For faster booting of apps, and real time updates
- Express - Framework to handle node modules
- HBS - HTML extension for integrating JS in HTML code
- Mongoose - Used for communication between the application and MongoDb servers

Limitations

The website wasn't able to provide online chat as it required a whole new API to install to the website which required a lot of time. Also, unfortunately we didn't have the time frame to embed it in, as well as when a candidate applies a form that needs to be filled in order for the recruiter to understand what the applicant information is before hiring them.