Job Hunter

https://github.com/comp195/Spring2020Project-cannifs-children

Dharak Vasavda | d_vasavda@u.pacific.edu

Nathan Luu | n_luu1@u.pacific.edu

April 30, 2020

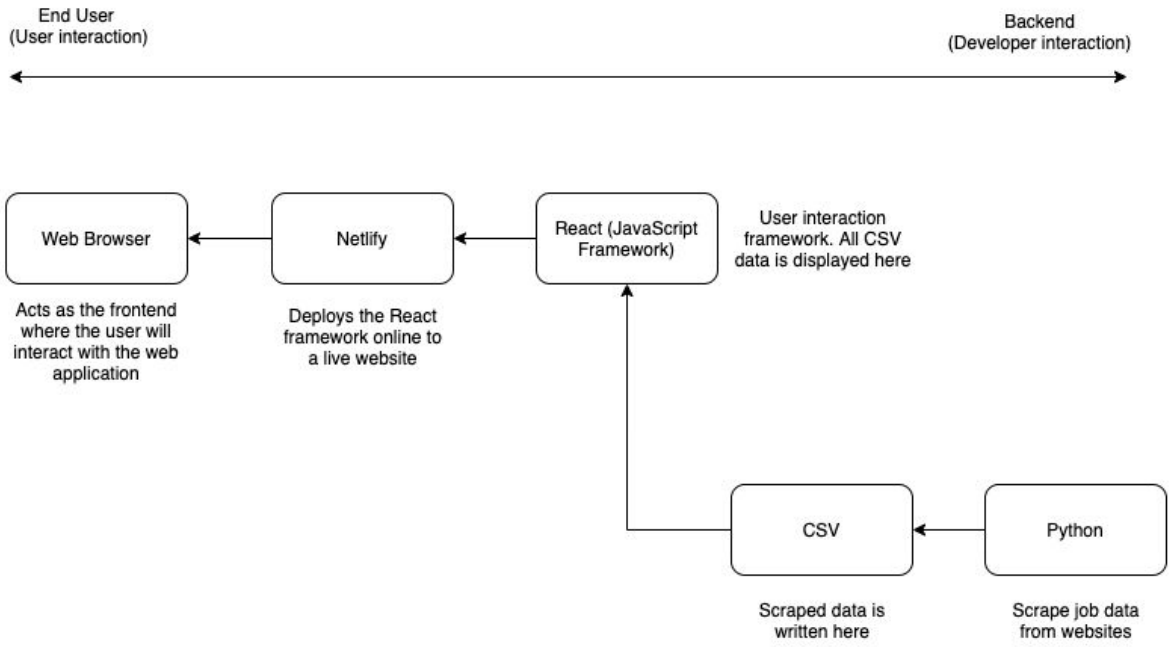
Project Overview

Job Hunter is an online web service which helps folks find jobs according to their given filters. The goal of this service is to help aggregate multiple job-finding websites such as Monster.com and Indeed.com together. This will create simplicity and ease-of-use for the end user to search for a job. To add, Job Hunter is a cross platform tool hosted online and supported by most modern web browsers. The key difference between Job Hunter and other job hunting websites is that Job Hunter will actively scan jobs on multiple Job posting websites and congregate them into one. This will make job hunting easier.

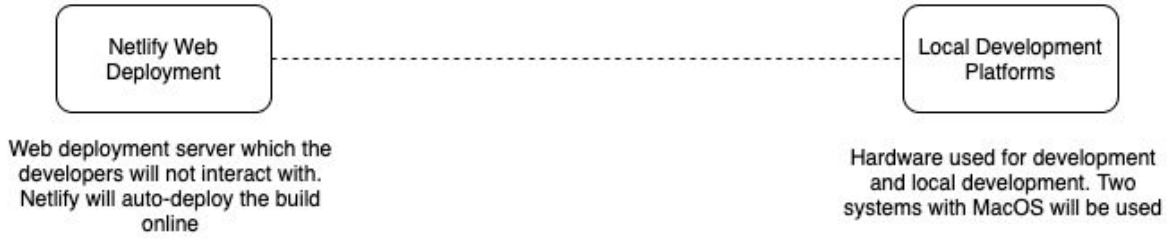
System Architecture

Job Hunter will require a structured architecture. Multiple tools and frameworks will be needed to create a modern UI-friendly and data processing aspect to the application. There will contain a frontend which will be the interaction component for an end user, and the backend which will integrate with CSV, Python, and React. Python will scrape data and then write its data to the CSV file. The React JavaScript library will display the data written to the CSV. Netlify will host all of the React library source on a live webpage.

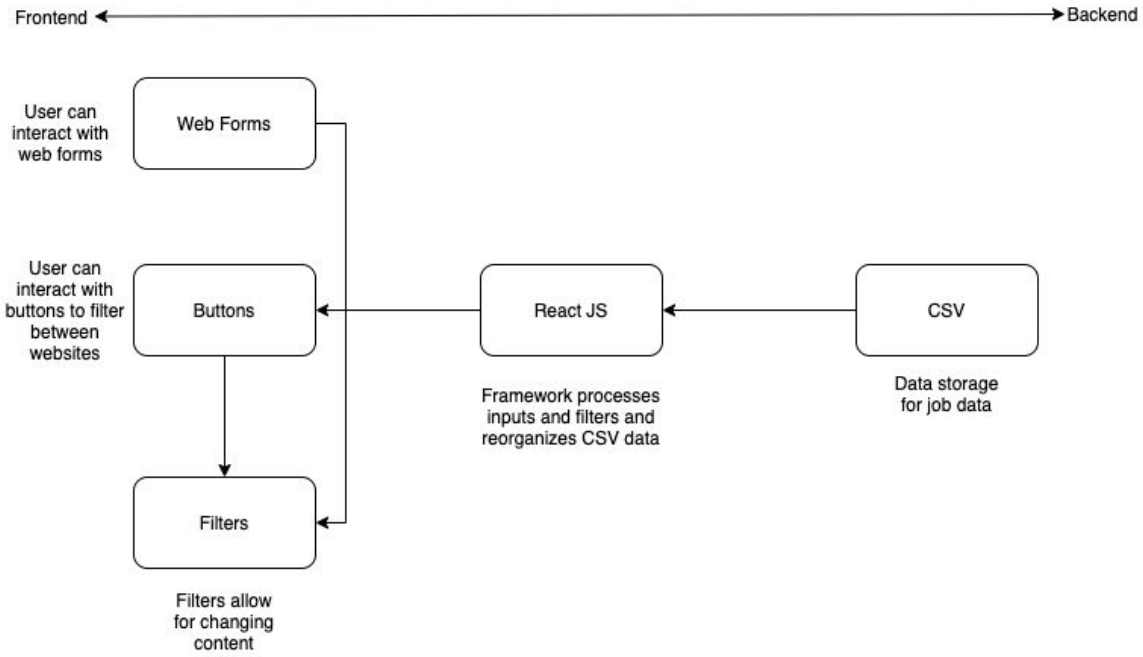
Software Modules



Hardware Components



User Interfaces



Hardware, Software, and System Requirements

Hardware & System Requirements:

Windows:

Windows 7, Windows 8, Windows 8.1, Windows 10 or later

An Intel Pentium 4 processor or later that's SSE2 capable

Mac

OS X Yosemite 10.10 or later

Linux:

64-bit Ubuntu 14.04+, Debian 8+, openSUSE 13.3+, or Fedora Linux 24+

An Intel Pentium 4 processor or later that's SSE2 capable

Software Requirements:

End User - Google Chrome, Safari, Internet Explorer, Firefox, or any web browser compatible with Javascript

Developer - Python >2.7, React (Javascript Framework), npm, yarn

External Interfaces

CSV

CSV will serve as the data storage mechanism. For the job API data pulled over by Python, the job data will need to be stored somewhere. CSV will be the data storage.

Refer to the extensive documentation: https://docs.python.org/3/library/csv.html

React

JavaScript library which will serve as an endpoint for end users to interact with. This includes forms, buttons, animations, and images contained on the webpage.

Refer to the extensive documentation: https://reactjs.org/docs/getting-started.html

Netlify

Netlify will serve as the backend hosting platform. The website will be hosted on this Continuous Delivery platform which will auto-deploy changes to a live website.

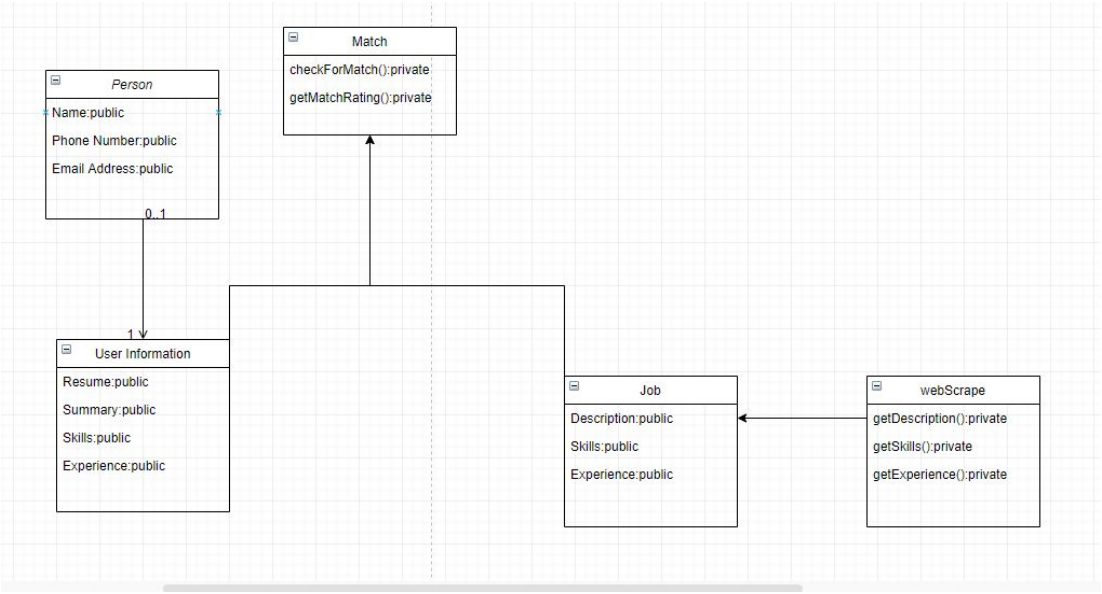
Refer to the extensive documentation: https://www.netlify.com

Software Design

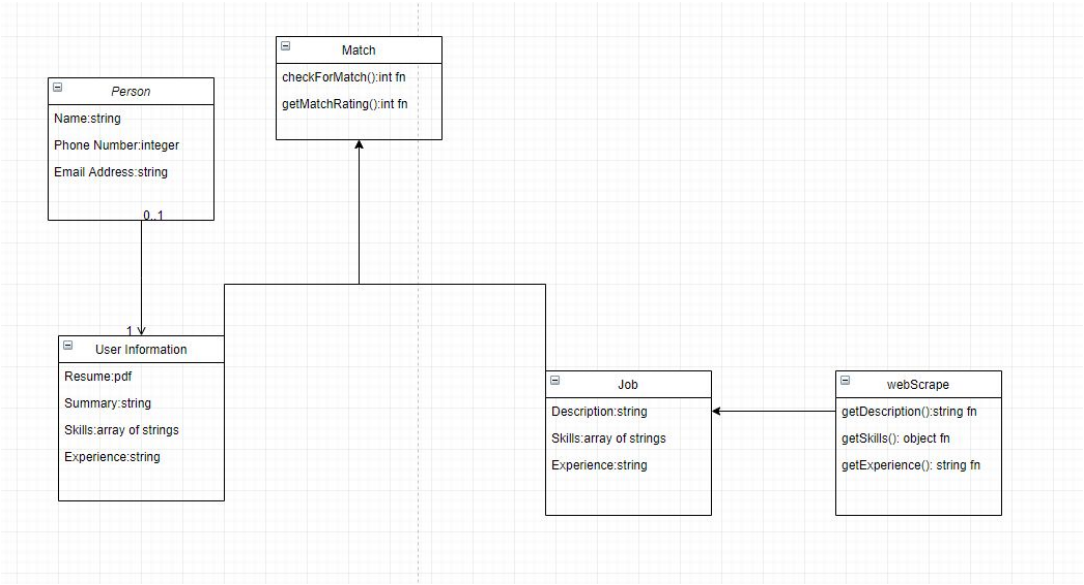
This section gives specifications for all software components, in sufficient detail to begin coding. The suggested documentation defined here assumes an object-oriented design using UML. If the software design does not fit with object-oriented principles (for example, a Prolog based logic program) alternate documentation will need to be determined.

Although this section defines software, it should not be tied to a specific programming language. One of the purposes in using UML for design is to allow for language independent designs.

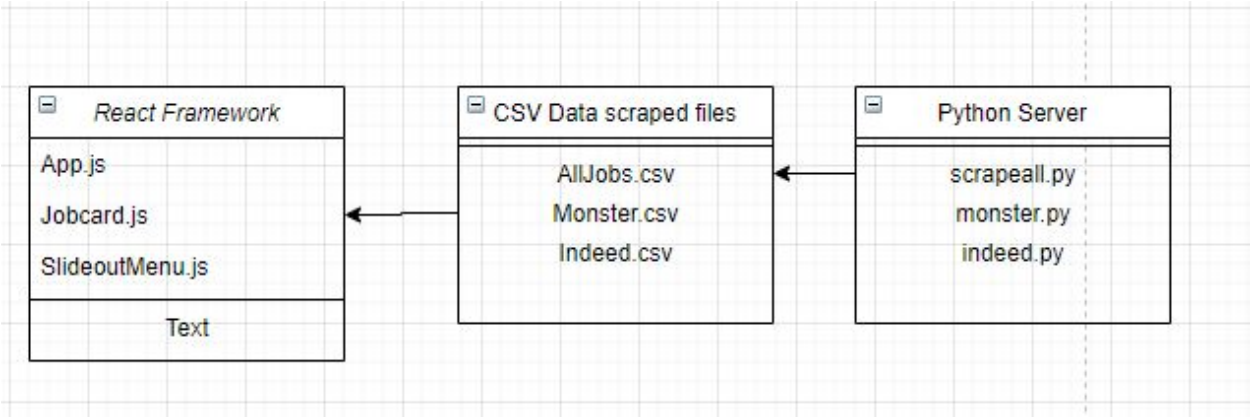
Class Diagram



Class Specifications



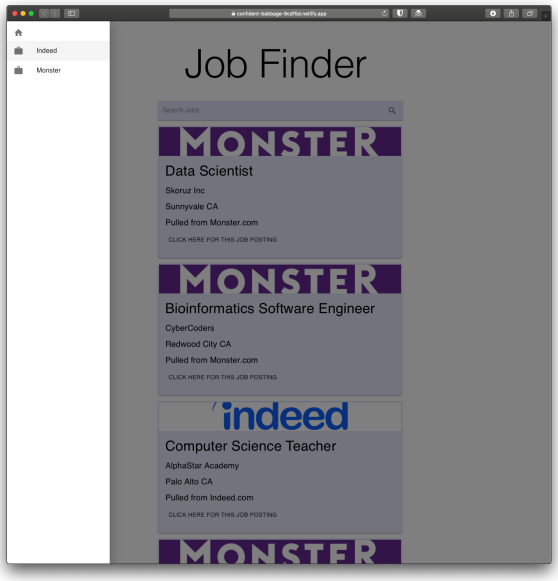
Interaction Diagrams



Design Considerations

Some design considerations to keep in mind are related to consistency. Several job searching websites such as Indeed and Glassdoor all have similarly consistent designs. The two focal points of these sites include a search bar and several buttons which act as filters. Making the search feature apparent will be important, as well as position title and company image. These are crucial in creating friendly job searching UIs.

User Interface Design



In this mockup, it shows the front page where you are allowed to search for any jobs with any filters you want. The second page shows the search query. The last page shows the profile page in which our algorithm will find best matches by searching for similar words between the profile and job description.

Glossary of Terms

References

No references were used for this project other than the linked documentation.