

SENG 513 Group Project Report

Instructor: Lorans Alabood

TA: Majid Askari



Divide 'n' Hire

Group #30: By: Ahad Hamirani, Isaac Lutzko, Reeyan
Maknojiya

Introduction

The objective of our project, “Divide ‘n’ Hire”, is to create a unique and improved take on the hiring process for both hiring managers and job-seekers (i.e., those searching for jobs) in the tech industry. The way in which programmers, designers, and tech specialists look for jobs, and in which companies looking to hire people with these types of skills, has not evolved and can be a very time-consuming journey for both parties - “Divide ‘n’ Hire” aims at solving these issues. Taking inspiration from modern dating apps, “Divide ‘n’ Hire” matches up hiring managers with people looking for tech-related roles through our filtering algorithm which streamlines the recruitment process and makes it more time-efficient. Once a hiring manager matches up with a job-seeker, they are then able to privately chat with the other person so they could both get more relevant information (e.g., the job-seeker could learn more about the job and the hiring manager could learn more about the job-seeker) and then move onto the next stage of the hiring process.

In this report, we will first provide a general background of the usefulness of our application, who our users and stakeholders are, and what our motivation is for this project. Next, we’ll discuss the goals of this project, describe what our users can do with this application, give a brief user manual for use of the application for both of our types of users, and analyze the changes between our mockups and our final project. Then we will prove that “Divide ‘n’ Hire” meets all project requirements, describe the technology we used and why and how we decided to use them, reflect on what we would add or change to the project if we had more time to work on it, and finally talk about what lessons we learned from this project.

Background

The motivation we have for “Divide ‘n’ Hire” began at the beginning of our brainstorming stage when we decided that we wanted a project that was fun to make, unique, useful for possible users, and beneficial for us so that we could learn the many different aspects of developing a web application. Although we came up with many different ideas for the project, we believe that “Divide ‘n’ Hire” meets our motivation criteria best which would make our development process much more enjoyable.

The main reason why we think “Divide ‘n’ Hire” is useful has to do with the currently outdated recruitment process used by people looking for tech jobs and by hiring managers looking to hire people for their tech roles. Although “Divide ‘n’ Hire” currently does not directly help with the interviewing stage of the recruitment process, it does however speed up how long it takes for job-seekers and hiring managers to meet and talk.

For job-seekers, rather than filling out the same boring job applications over and over they can just simply decide if they’re interested in jobs posted on “Divide ‘n’ Hire” by a hiring manager and chat with them later if a hiring manager is interested in them. This can save a lot of time for a job-seeker because they can use that saved time for improving their skills (e.g., learning a new programming language or software) which could make them more attractive as a job candidate to future hiring managers.

And for hiring managers, this process remains largely similar as they still have to look through job-seekers’ profiles and decide if they’re interested in connecting with them or not, but communicating through an instant messaging platform can be quicker and more convenient for them compared to email.

Besides the two types of users of our application, other stakeholders would be companies that employ these hiring managers and possibly the families of job-seekers (as it could give more time around family).

Project Goals

The overall goal of our project was to learn a lot of important web development skills by creating a unique and innovative web application that could catch the eyes of future employers for us.

To start, as a team we were aware that there are a large number of different aspects of web development (UX design, UI, Frontend development, databases, APIs, servers, etc.) and we had previous knowledge and skills with some of these aspects, but how all these different aspects work together to make a complete web application was somewhat unknown to us, so learning how all the pieces fit together was one of our main goals. Another important part of web development we wanted to learn was how single-page applications work and how web frameworks such as React or Vue.js come into play, so these are also parts of modern web development we wanted to become proficient with.

Next, creating a project that was unique, innovative, and we would be proud of, was also part of the overall goal of our project. A unique project that we would be proud of has two main benefits. One, a creative project would be great on a resume as it would showcase our web development skills and creativity to possible future employers. And two, creating a project that isn't a replica of an already established project (e.g., checkers, Facebook, etc.) is more exciting and captivating which makes the overall project more fun and enjoyable.

Project Accomplishments

As previously mentioned, "Divide 'n' Hire" connects hiring managers to job-seekers with a focus on tech-based roles, but what exactly can our two types of users do using our application?

Hiring managers

- Hiring managers that create an account on behalf of their company and sign in to the "Divide 'n' Hire" web application have the ability to add multiple job postings for their company, look through a queue of job-seekers for each job posting they make, decide whether they are interested in each job-seeker or not, and chat with job-seekers they are interested in if the job-seeker is also interested in a job they have posted.
- It is important to note that these queues of job-seekers for each job posting that are shown to hiring managers are different because the job-seekers in each queue have explicitly stated what kind of job type(s) they are interested in and will not appear in a queue if they didn't state they were interested in their profile. This is done through our filtering algorithm.
- An example of this would be if there was a hiring manager at a large company and they were looking to hire people for multiple tech roles, let's say a Full Stack Developer role, a Software Engineer role, and a Data Scientist role. This hiring manager could create an account on behalf of their company and from there could add those roles on their account. From there, they could start by going to the explore page and select the Data Scientist role that they posted, and then look through job-seekers that have stated they are interested in working as a Data Scientist. After reviewing a job-seeker's profile, the hiring manager can

decide whether they are interested in connecting with the job-seeker or not, and they cannot move on to see the next job-seeker until they've made a decision. The hiring manager can then do the same for the Full Stack Developer role and the Software Engineer role that they've posted. If a job-seeker decided that they are interested in a job that this hiring manager has posted and is also interested in that job-seeker, they can then chat and connect through our private messaging component.

Job Seekers

- Job-seekers that create an account and sign in to the application can explore different types of jobs that they're interested in and have been posted by hiring managers, make a decision on whether or not they're interested in a job, and chat with hiring managers that are interested in them if they too are interested in jobs posted by these hiring managers.
- An example of this starts with a person looking to find a new job in either a Front-end developer role or a Full-stack developer role, because they are adequately qualified for either one and are equally interested in both. This person could first by creating an account and fill out all their information for their profile. From there, they could go to the explore page and go through the different Front-end developer or Full-stack developer jobs shown to them and posted by hiring managers. Then they decide whether or not they are interested in the job they're looking through (and similar to hiring managers they cannot move on to the next job until they've made a decision), and if they decide that they're interested in a job and the hiring manager of that job is also interested in the job-seeker, they can chat message each other to move on to the next stage of the hiring process and hopefully set up an interview.

Detailed Project Description

Although there are many similarities between some of the pages that a hiring manager user would see and use compared to a job-seeker, to give a thorough project description we will give a brief user manual for hiring managers, and then one for our job-seekers where we only point out the differences between job-seekers and hiring managers.

Hiring Managers

- To start, hiring managers begin at our landing page (see Figure 1) which we kept fairly simple with a fun, easy-going design, that contains only one button which is the 'Join Now' button. When selected, this button redirects the user to the sign up form page.
- The sign up form page has input text boxes so that the hiring manager can fill out their email, a password, and a confirmed password, as well as a dropdown so that they can decide between setting up their account as a job seeker or as a hiring manager (see Figure 2). From there, the page also has a 'Sign Up' button below the input text boxes so that once valid input is added to the inputs and the dropdown selected, the hiring manager can select that button to move on. It is important to note that if invalid information is entered into the input text boxes or the dropdown left unselected, users are prompted with an appropriate error message and does not let them to move forward.
- Once the hiring manager correctly filled out information on the previous page and selected the 'Sign Up' button to move on, they move on to the next page which lets them fill out their

company name in an input text box and add their company logo which will be used as part as the jobs they post next (see Figure 3). After filling out the appropriate information, hiring managers select the 'Finish' button below the input text box and adding the company logo to move on to the next page which redirects them to the explore page.

- At this point, the Explore page will not contain any job-seeker's profiles to look at (see Figure 4) because they have not added any job postings yet, but there is still however the navbar which contains three options at the top left of the page: Explore (current page you're on at this point), Matches, Jobs, and Logout. To start adding information so that hiring managers can look at job-seekers on the Explore page or see possible matches on the Matches page (which would be empty at this point as well because there would be no job postings at that time), the user can select the 'Jobs' option at the top right of the navbar to go to the Jobs page.
- On the Jobs page, we see the page is empty except for the same navbar from the Explore page (which will stay that way on the rest of the pages) and a '+' button icon - this button allows the hiring manager to move to the Add Jobs page so that they can add a new job posting (see Figure 5).
- Now on the Add Jobs page, hiring managers can fill out the following information for their job posting: the job title, start date, location, and salary which are input textboxes near the top on the right, dropdowns for selecting the job type and the number of years of experience that the hiring manager is looking for which are located on the left of the page, a '+' button on the right where hiring managers can add the skills (programming languages, development skills) they're looking for in a possible applicant, and finally a large input text box at the bottom where hiring managers can fill out the job description (see Figure 6). After the information is filled out, the hiring manager can select the 'Add Job' button at the bottom of the page to finish the process. This brings the hiring manager back to the Jobs page where the recently added job is now shown on the page (see Figure 7). To delete a job here that was added, a user can select the large 'X' button that is inside the boxes that contain a job's information.
- After adding a job posting or two and selecting the Explore option on the top of the navbar, we now see the Explore page that contains your job postings on the left sidebar, and the job-seeker profile cards on the right that are filtered for your job postings with a button bar below the profile cards (see Figure 8). First, to look at job-seeker profiles for each job posting made, a hiring manager can select the title of a job posting on the left sidebar which will show what job-seekers correspond with the job posting they've selected, and this is depicted by the arrow that displays this. The button bar on the bottom contains a green checkmark button which hiring managers can select to show that they are interested in a job-seeker and a red 'X' button which hiring managers can select to show that they are not interested in a job seeker. After selecting one of these buttons, the hiring manager will be shown another job-seeker profile card to look through.
- Now, this Explore page that was just recently discussed is applicable to a wider screen (greater than or equal to 992px in width to be exact) but it is somewhat different for smaller screens to offer a better user experience. With smaller screens we hide the left sidebar and instead just show one box that contains either a job-seekers profile card or the information that would be on the left sidebar for choosing which job-posting to look at (see

Figure 9 and Figure 10). By selecting the left chevron arrow on the left near the top of the page, hiring managers can select that arrow to toggle between looking at a job-seekers profile card or looking at the list of job-postings they've made so that they can decide which one to focus on. This responsiveness for the explore page is also applicable to the matches page as it has a very similar layout.

- So moving to the matches page by selecting the 'Matches' option at the top in the navbar, the hiring manager would now see a left sidebar that contains their job postings and an arrow beside each one, and on the right there is a box that contains their conversations with different job-seekers they've matched with (see Figure 11). To simplify things on the left sidebar, hiring managers can select the name of each job posting or the arrow which would then drop down the names of the appropriate job-seekers that correspond with the job posting. The hiring manager can then select the name of a job-seeker there to see their chat history and then send messages there. On the right in the conversation box for each job-seeker they've matched with, the hiring manager can type a message in the input form at the bottom of the page and can press the orange 'SEND' button to send the message.

Job-Seekers

- Now moving on to our job-seekers, they start at the same landing page and sign up form page as the hiring manager, the only difference with the sign up form page is that when they select 'Job-seeker' in the dropdown they're redirected to a different page than the hiring manager which is where job-seekers create their profile.
- The "create profile" page for job-seekers has a lot of inputs and forms because this allows users to showcase their skills and experiences to hiring managers (see Figures 12a and 12b). It is important to note that we don't have the navbar options here because job-seekers need to fill out their information before moving on, and in the future they can edit their profiles later. At the top of this page, there are three input text boxes for filling out your first name, your last name, and a description about yourself (this textbox is bigger), and to the left of these input text boxes there is a button that allows users to add a profile picture. Below that top section, job-seekers can choose up to three different job types that they are interested in using three dropdowns with the same options. It is important to note that users must at least fill out the information we discussed on this page thus far and can move on to the next page if they so please. Below that there is a 'Skills' section where job-seekers can select the '+' button on the right which opens up a popout where they can select checkboxes that correspond to the skills they have and leave the other checkboxes unchecked. After selecting the appropriate skills, the job-seeker can select the 'Add Skills' button to add these skills to their profile which then closes the popout. Below this 'Skills' section, we have our 'Relevant Experience' section where job-seekers can fill out information regarding previous job experience. For each job experience a job-seeker has, they can fill out the company and title of the job on the left in the input text boxes, then use the two dropdowns to the right of these input textboxes where they can select what type of job it was and how many years they had worked in each particular job, and finally a large job description input text box on the right where the job-seeker can give a brief description of the job. If a job-seeker wants to add more work experiences (i.e., another job they worked at a different company), they can select the '+' button on the top right of the

'Relevant Experience' section which then creates these forms again that you can fill in. Below the 'Relevant Experience' section the job-seeker can go through the 'Socials' section - this is where job-seekers can add links for their LinkedIn and their GitHub so that hiring managers can go to this job-seeker's profiles. For this section, there is a dropdown on the left where the job-seeker can select LinkedIn or GitHub, and to the right of that, there is an input textbox where they can insert the appropriate link there. To add another social platform and link, the job-seeker can select the '+' button on the right of this 'Socials' section which then creates these forms again for them to fill out. The final section of this page is the 'Education' page where job-seekers can add education info that they have such as information about their university degree. This section has two input text boxes, one for entering what education type and one for entering what institution the job-seeker got the education from, and below the input text boxes there are two date input sections, one for start date and one for end date, where users can either fill out the dates via keyboard or they can select the calendar icon in the date input sections which opens a calendar for selecting the date. Finally, after entering all this information for their profile, a job-seeker can then select the 'Finish' button at the very bottom to finalize their profile and move on to the next page which is the Explore page.

- Now the main difference between the Explore page for job-seekers and the Explore page for hiring managers is the absence of the left sidebar as it is not necessary for job-seekers, and rather than having job-seeker information in the main section of the page, there is only job postings that are posted by hiring managers (see Figure 13). The job-seekers can then use the green checkmark button and red 'X' button at the bottom (same as from the hiring manager Explore page) to decide if they are interested in a job posting or not. It is also important to add that the navbar for job-seekers gives these 4 options: Explore, Matches, Profile, and Logout. This navbar is consistent through these pages when logged in for job-seekers.
- After selecting the 'Matches' option at the navbar, job-seekers will see the Matches page which is very similar to the Matches page for hiring managers (see Figure 14). The only difference between the two is that rather than selecting a job title on the left sidebar of the hiring manager Matches page which drops down a list of job-seekers, job-seekers on the Matches page just select a job title on the left sidebar that opens up the chat history between the hiring manager of that selected job and the job-seeker without dropping down a list of job-seekers which obviously is not needed here.
- On the 'Profile' page, job-seekers can view what their profile looks like and what a hiring manager would see (see Figure 15). If a job-seeker would like to edit their profile they can select the pencil button on the top right of the page which transitions them to the 'Edit Profile' page. The 'Edit Profile' page has the exact same fields, design, and functionality as the 'create profile' page, the only difference is that the information in these fields is already filled out from what their profile currently looks like (see Figure 16).

Mockups vs Final Project

There are not many differences between the mockups we made earlier and the final project we have now and a big part of that has to do with the fact that our mockup was very

well-crafted - this also made the development process much simpler. With that being said, there are a few differences between the two sections so we will touch on them now.


First, on our landing page, our sign up page, and on our create profile page for job-seeker for our mockups, we had a nav bar where users could select the options 'Explore', 'Matches', and 'Login', but on our final project we do not have these options on our navbar (see Figure 17 and Figure 1 to notice the navbar). This helps our users because at these pages they should not be able to navigate to different pages before they've either logged in or created an account - this creates a better user experience.






Second, on our mockup when a hiring manager creates an account by selecting the 'Sign Up' button on the sign up page, we open a popup to prompt the user to enter their company name and their company logo (see Figure 18). However, on our final project we don't open a popup, instead, we show a new page that is very similar to the sign up page except for the forms and inputs that a user can enter information into (see Figure 3). This is not a large change but good to note.

Next, an important change we made was including a header and toggle arrow button on the right side of the Explore page for hiring managers (see Figures 19 and 4 to notice the Explore page differences). This is included to support the mobile support requirement of the project that we will discuss shortly and to help users know what they are viewing and where they are on the application.

Finally, the last main change we made was to get rid of the skip button that is on the Explore pages for both hiring managers and job-seekers on our mockups (see Figures 19 and 4). On our final project we only have the green checkmark button and the red 'X' button on the button bar at the bottom of this page. Other than these changes that have been discussed, there are no other major changes that we have made from the mockup stages to the final project.

Project Requirements

Project Requirement	Was the requirement met?	Reasoning for why Divide 'n' Hire meets this requirement
Client Side		The reason as to why Divide 'n' Hire meets this requirement is because it is accessible through a single-page interface. The application has one single HTML file, an index.html file, and the rest of the interface is done through Vue.js files which all use HTML, CSS, and JS technologies. It is also important to note that Vue is a free open-source web framework and our application was testing using both Chrome and Firefox browsers and they worked as expected which meets this requirement.

Mobile Support		Beginning by using a mobile-first development approach, this made this mobile support requirement must easier to meet. With the use of Bootstrap and CSS media queries, our application is responsive and mobile friendly.
Server Side		Most of our application logic such as our filtering algorithm and how we store user information is all done on the server side. As required, our server side is implemented using Node.js and relies heavily on JavaScript combined with communication via websockets with Socket.IO.
Multiple Users Support		Because our application needs both job-seeker and hiring manager users, Divide 'n' Hire depends on having multiple users. Due to the nature of modern platforms, the more the users the better the application, so our application supports multiple simultaneous users. We are able to do this through the use of Node.js and how we store our information on the server side. We've tested this and it's proven that multiple users can use the application at the same time. We've also implemented a very basic user authentication that is completed by users at the login and sign up pages for both hiring managers and job-seekers.
Persistence		With respect to the persistence capabilities of our application, once our servers are running, job-seekers and hiring managers can both create accounts and begin using the application. These users are able to logout and login again and their information such as matches, chat history, etc., will still be there. However, it is important to note that if our servers are stopped, these users' information will be gone as it is stored on the server side of our application and not in a database. This is verified by professor Alabood and TA Askari as being acceptable. The lack of a database will be discussed in the "Future Work" section of this report.
User Interaction		Our application offers two ways of interacting with users. One, job-seekers can like or dislike a job posting that is published by a hiring manager, and a hiring manager can like or dislike a job-seekers profile as previously mentioned. And two, users can privately chat with each other if they have matched up. This interaction was key to our application and was performed using Socket.IO.

Technology Used

There are many different technologies that we've used to create "Divide 'n' Hire" and the main technologies were Vue.js, Node.js, Express.js, Socket.IO, Bootstrap, and JavaScript. We will now discuss why and how we used each one, and then provide brief examples of each of them. It is important to note that these technologies are already used as part of our application

and there isn't much of a way of obtaining these technologies because they are already included in our JSON files and are imported into many of our files instead. With that being said, there are a few small things that need to be done so that someone could run our application.

The first and most important technology we've used is JavaScript. Obviously using JS on both the client and server-side of our application is a requirement so that is why we're using it, it is also important to bring it up because all the other technologies we use depend on JS. How we used JS consists of the main logic aspects of our application, storing our data objects, and making the UI of our application more dynamic. An example of this is how we used JS to write our filtering algorithm which determines the appropriate job-seeker profile card to show to hiring managers on their Explore page and determines the appropriate job postings to show job-seekers on their Explore page. Note that JavaScript is not needed to be installed.

The next technology that we used that was important was Node.js. Node.js is an open-source JavaScript server framework that allows developers to run JavaScript on the server. Node.js was also a requirement but if it wasn't we still would have used it because of how much it helped us with building our server-side and with checking if information is properly being properly sent from the server-side to the client-side. An example of this would be running our Node.js servers so that we can run our application and then test and make sure that our job-seeker data is properly sent to the client-side. When it comes to installing Node.js, a user can follow the link here (<https://nodejs.org/en/download/>) to download the appropriate version for your operating system and that is how someone could obtain Node.js to run our application.

Another crucial technology we used was Socket.IO. We decided to use Socket.IO because it allowed us to have asynchronous communication between the server and clients via WebSockets which is very simple and easy to use. We used Socket.IO for 2 main things. The first way how we used Socket.IO was to send user information from the client-side to the server-side to store. An example of this would be a job-seeker filling out their information on the create profile page and when they select the 'Finish' button to complete that process, the information is sent from the client-side to the server-side with the use of WebSockets. Because this information is stored on the server-side, it can then be used and retrieved so that it can be sent back to the client page when the client requests that information. The second way how we used Socket.IO was for our private messaging functionality (e.g., sending messages to and from hiring managers and job-seekers). Because Node.js should be installed as it was just previously mentioned, to install Socket.IO a user can run the command 'npm install socket.io' in their terminal (Node.js needs to be installed because it has its package manager, npm, which is what is installing socket.io).

The next technology we used was Vue.js which is a front-end JS framework for creating user interfaces, specifically for single-page interfaces. One of the main reasons why we used Vue.js was because it has made developing our front end much easier and it makes creating single-page applications much easier. How we used Vue.js was to create different Vue 'view' pages that contain all the components for all of our pages. Each page consisted of an HTML section, a JS section, and a CSS section - this format made our development easier and more straightforward. An example of this would be looking at the UI of our Explore page for either a hiring manager or a job-seeker. To obtain and use Vue.js to run our application, one would run the command 'npm init vue@latest' in their terminal.

Finally, the last main technology we used in our application was Bootstrap which is a free front-end open-source CSS framework that helps with making web applications responsive. The reason why we chose to use Bootstrap is that it helps greatly when it comes to making a web application responsive and mobile-friendly. How we used it is through Bootstrap snippets to include in the HTML code, specifically in the classes of an HTML element, that is in our Vue.js pages. An example of this would be including the text “col-lg-6” in the class of an HTML element at a specific point. This text means that this HTML element should cover half of the container on the page it is on, and the breakpoint for when an element should be collapse would be at “lg” (or large) which is considered to be 992px in Bootstrap. Bootstrap is not needed to be installed to run our web application.

Future Work

Although we are very proud of what we have created in a limited amount of time and in difficult circumstances regarding group work, there are definitely some aspects that we would add to “Divide ‘n’ Hire” to make it a better project for ourselves and for our users.

The first thing we would change with our application would be to include an API and database so that even if our servers stopped after a user is done using the application, the user could then start up the server again and see all their matches, their profile, etc, and pick up where they left off. If we were given more time we would likely use MongoDB as our database because the way we’re currently storing our information in objects would work well with MongoDB.

Next, we would add a verification aspect for both job-seekers and hiring managers so that they can be more sure that the other user they are interacting with is in fact who they say they are. There are many issues with bots and people impersonating other people on information sharing platforms so adding a proper verification component to our application would instill more trust for our users.

Lastly, we would implement a video/audio call feature into our private messaging with matches functionality. Just like with other platforms such as Facebook, Snapchat, and dating apps, allowing users to connect via a video or audio call can save time for both hiring managers and job-seekers. This would be especially beneficial in the recruitment/hiring process as it would allow hiring managers and job-seekers to carry out a remote job interview through our application.

Lessons Learned

This project has been a lot of fun and there is a ton we have learned from this project, so we’re now going to highlight some of the more notable things we learned from building “Divide ‘n’ Hire”.

The first thing we learned was how great Socket.IO is. Not only is it useful for messaging/chat features in websites but it also can be used to send other data (e.g., input/form data) through websockets on the click of a button. The simplicity of Socket.IO also makes it great and I’m sure we will use it in future projects.

Another key technology we used and benefitted greatly from was Node.js. Node.js speeds up web development dramatically as you can see what changes you are making as work in real time. We also initially started by focusing on using Django for storing our user

information which offers many benefits, but upon reviewing the project requirements we realized we could not because it offers the same offerings that Node.js does and we needed to use Node.js (Node.js is based on JavaScript, Django is based on Python). Even though that meant we needed to be modify our strategy and what technologies we needed to use, we did learn a lot about Node.js and all of it's benefits. From this we also learned the importance of being agile, flexible, and reactive to changes in the development process as things are bound to change.

We also learned the benefits of using web frameworks such as React and Vue.js. Because of the requirement of creating a single-page application, using these frameworks made development much easier and organized as it it offers a great way of breaing up big projects. Through our research we decided to use Vue.js because of it has a smaller learning curve compared to React and we didn't have a whole lot of time to learn these frameworks. Learning and working with Vue.js has been very beneficial and easy to pick up so we would recommend it to others.

When it comes to working in a team there are also other important aspects we learned. First, working with others can be difficult and you cannot always depend on other to communicate, step up, and pull their own weight. This is something that is universal and sometimes happens, but we learned that the best thing to do is to still do your best to include all group members, and try and get them engaged. Even if that is not very successful, we also learned that all you can do is make the most of it, see it as a way of learning more and improving resiliency in difficult situations, and communicate effectively with people to the best of oyur ability. Second, we also learned that completing a big project on a tight time constraint can be daunting, but working together and figuring out who can contribute best to each different part project is key. This translates across the board, not just working on a programming final project, because everyone will have to work with others in the future.

Conclusion

To summarize, we first introduced our web application by discussing the objectives of the project and the major features and functionalities of it. We then described why "Divide 'n' Hire" is useful, who our users are, what motivated us to make this type of project, and what the overall goals of the project was and how we met these goals. After that, we touched on what each type of our users, hiring managers and job-seekers, are able to do using our application, and then gave a brief user manual for how both hiring managers and job-seekers can use our web application, especially for all the different elements we have on each page. From there we explained the main changes between our mockups and the user interface of our final project and then demonstrated how "Divide 'n' Hire" met all project requirements. Next we described which technologies we used, why and how we used them, gave examples of how we used them, and then explained how to use these technologies to run our application. Finally, we discussed what we would add to the project if we had more time to work on the project, and then finished off by explaining the lessons we learned from completing this project.

Creating a web application of this level was an intimidating task and it was difficult to finish but we learned a lot about ideating, designing, developing, and implementing a web application and enjoyed the process as of doing so. "Divide 'n' Hire" is a project that we'll be proud of and will be beneficial in all of our careers going forward.

Appendix

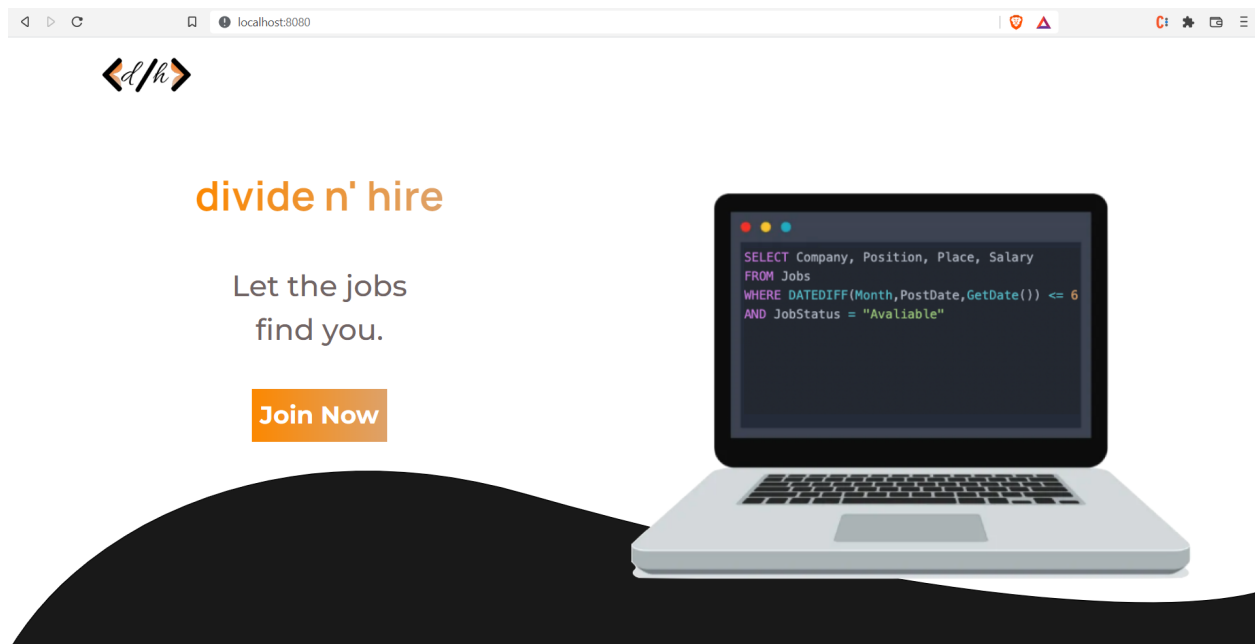


Figure 1: Main landing page for both hiring managers and job-seekers

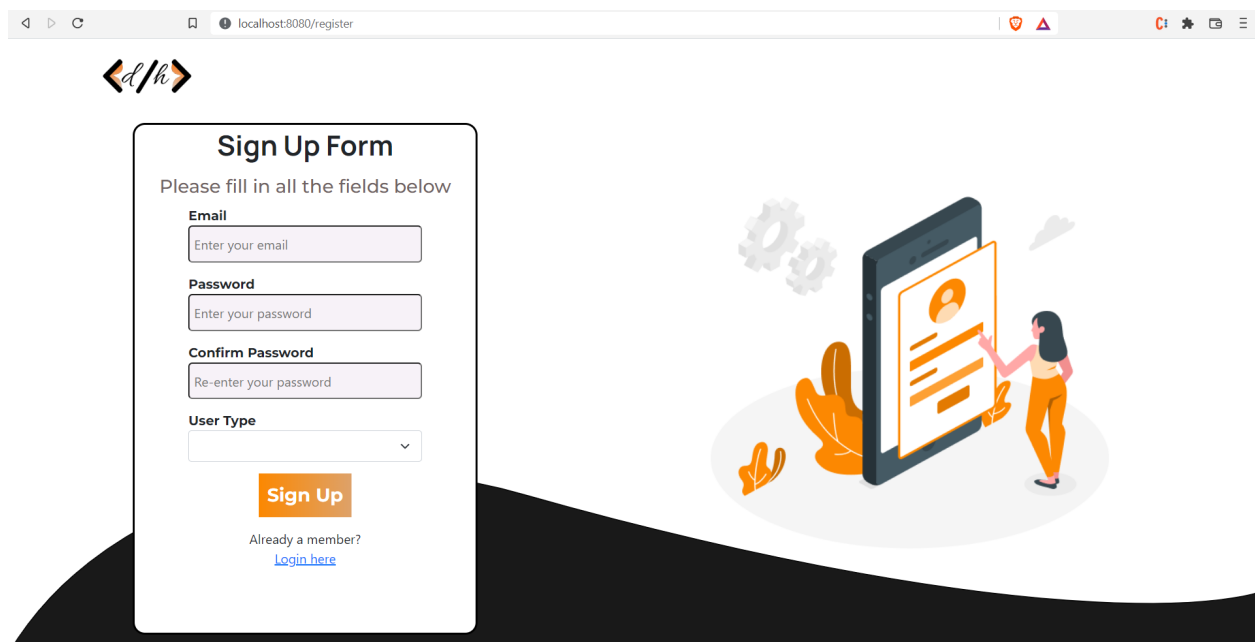


Figure 2: Main sign up page for both hiring managers and job-seekers

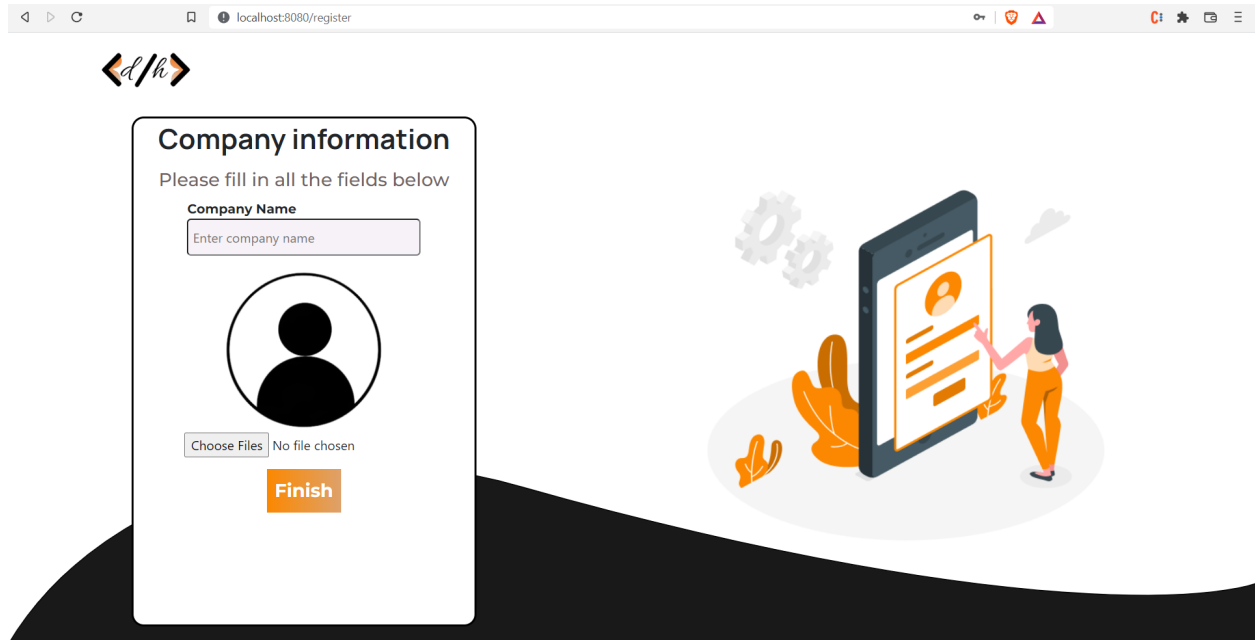


Figure 3: Company information page for hiring managers

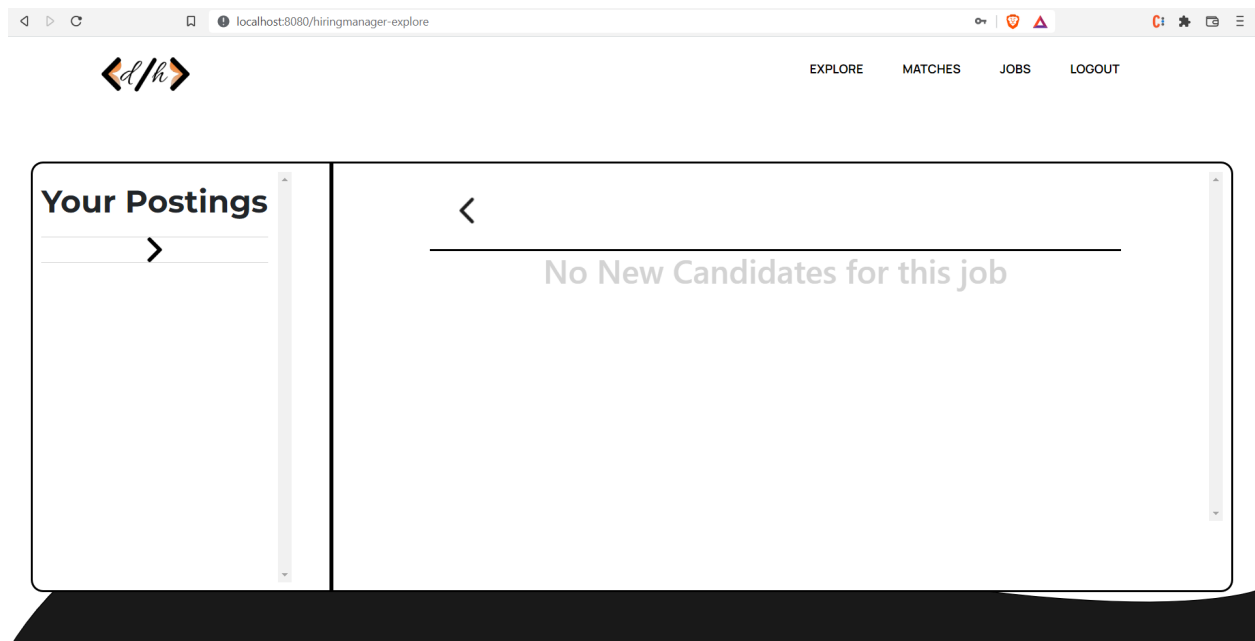


Figure 4: Explore page for hiring managers when a hiring manager has not yet added any job postings

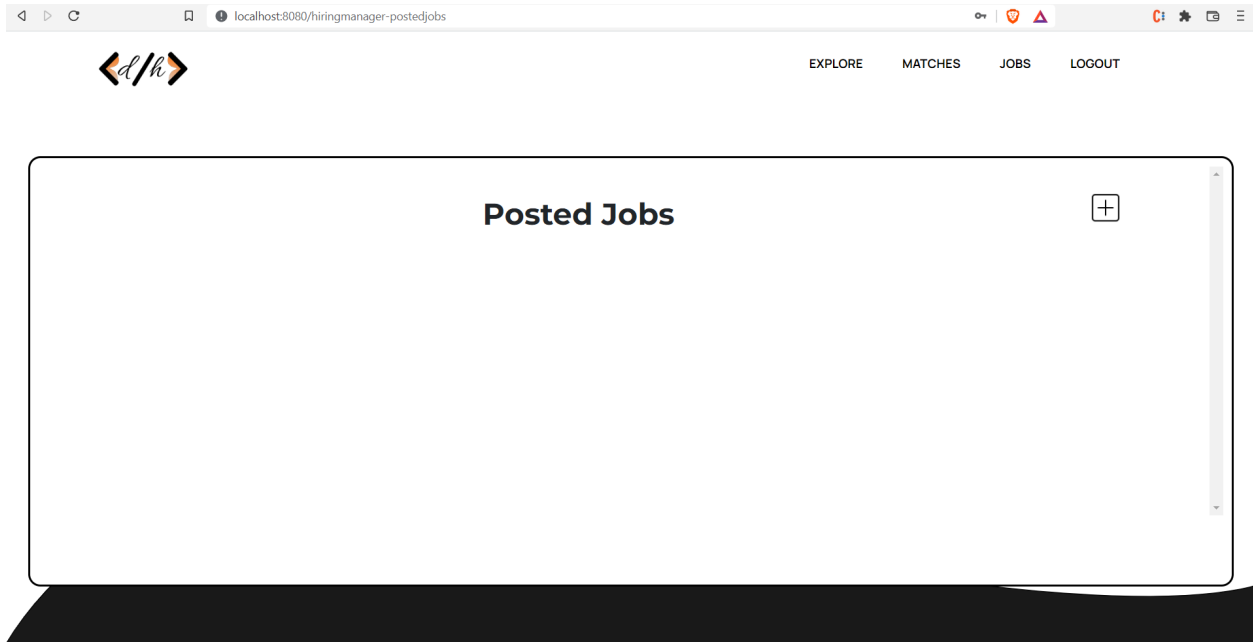


Figure 5: Posted jobs page for hiring managers when a hiring manager has not yet added any job postings

A screenshot of a web browser showing the 'Add a Job' page. The browser's address bar displays 'localhost:8080/hiringmanager-addjob'. The page features a logo on the left and navigation links (EXPLORE, MATCHES, JOBS, LOGOUT) on the right. The main content area is titled 'Add a Job' and contains several input fields: 'Job Type' (a dropdown menu), 'Job Title' (a text input with placeholder 'Enter the job title'), 'Location' (a text input with placeholder 'Enter job location'), 'Years of Experience' (a dropdown menu), 'Start date' (a date input with placeholder 'yyyy-mm-dd' and a calendar icon), and 'Salary' (a text input with placeholder 'Enter estimated yearly salary'). Below these fields is a section titled 'Skills' with a small square button with a plus sign. At the bottom is a large text area for 'Job description' with the placeholder 'Give a brief description of your job'. An orange 'Add Job' button is located at the bottom center of the page. The browser's developer tools are visible at the bottom.

Figure 6: Add jobs page for hiring managers which is where hiring managers can add job postings

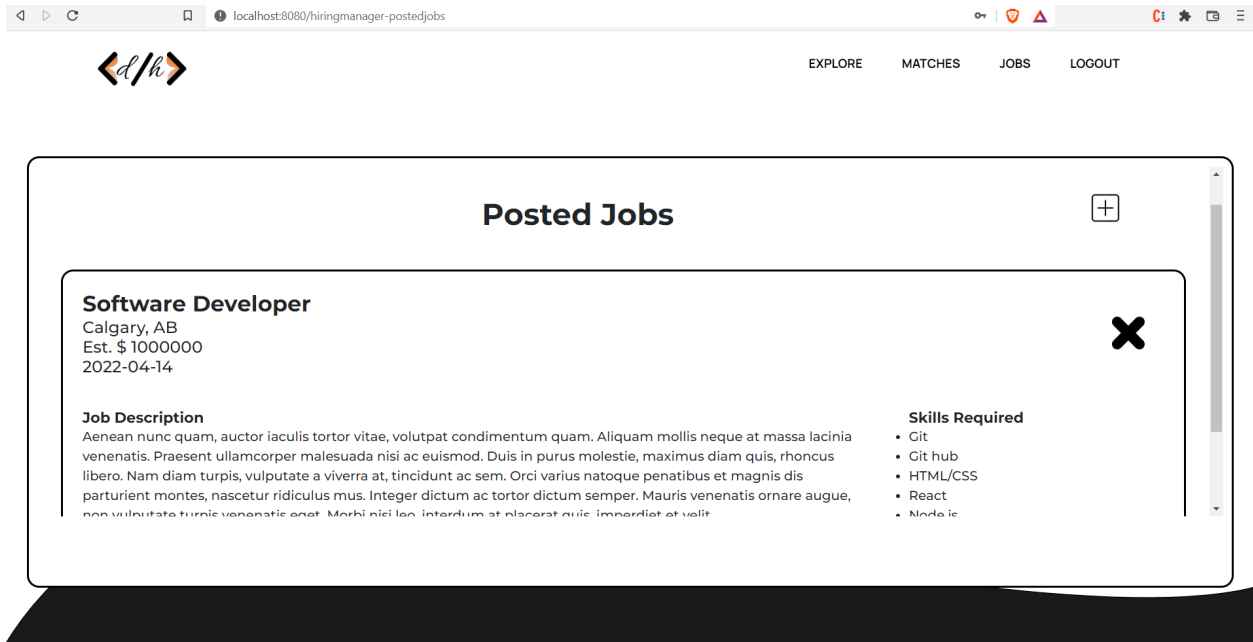


Figure 7: Posted jobs page for hiring managers when a hiring manager has added job postings

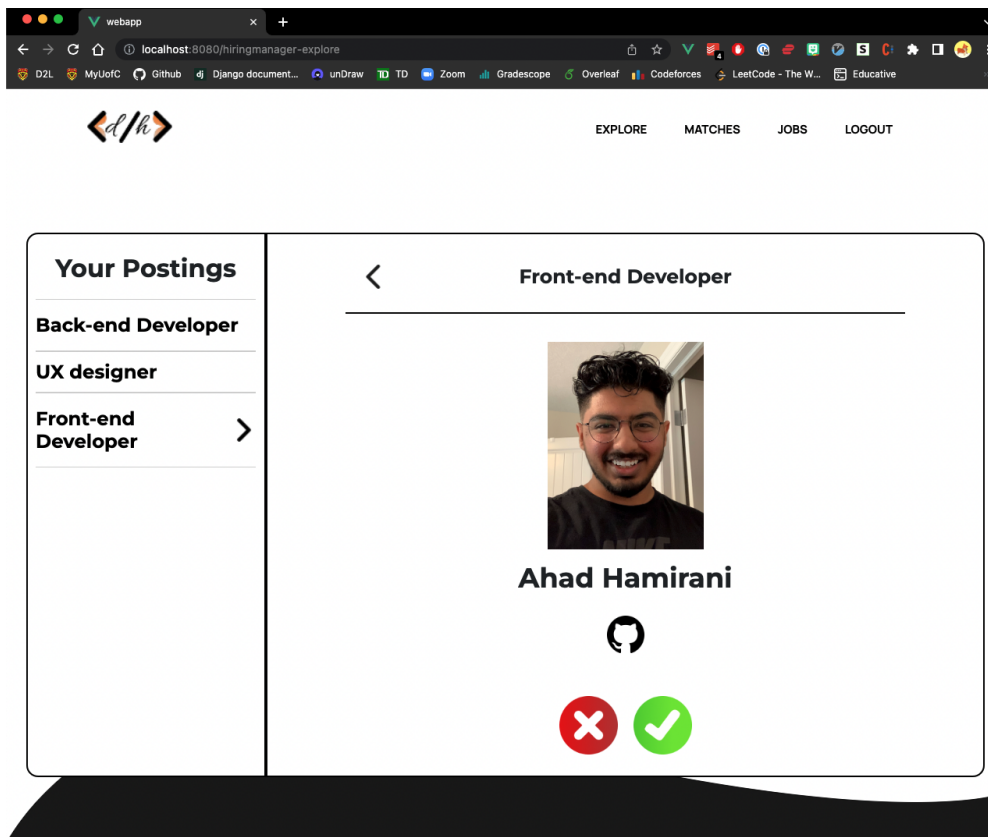


Figure 8: Explore page for hiring managers when a hiring manager has added job postings

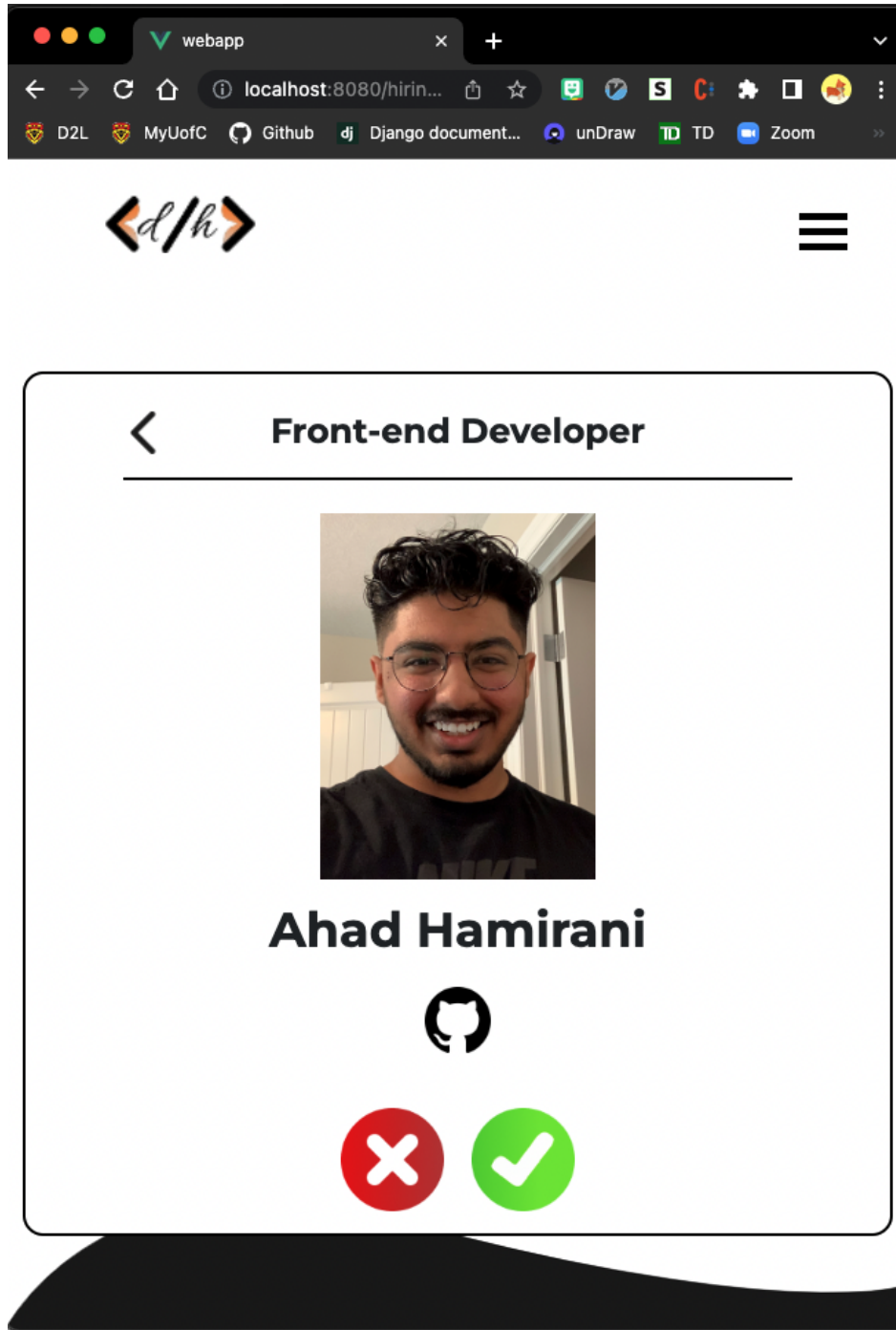


Figure 9: Explore page for hiring managers when screen is narrow and when hiring manager has added job postings and viewing job-seeker profile card

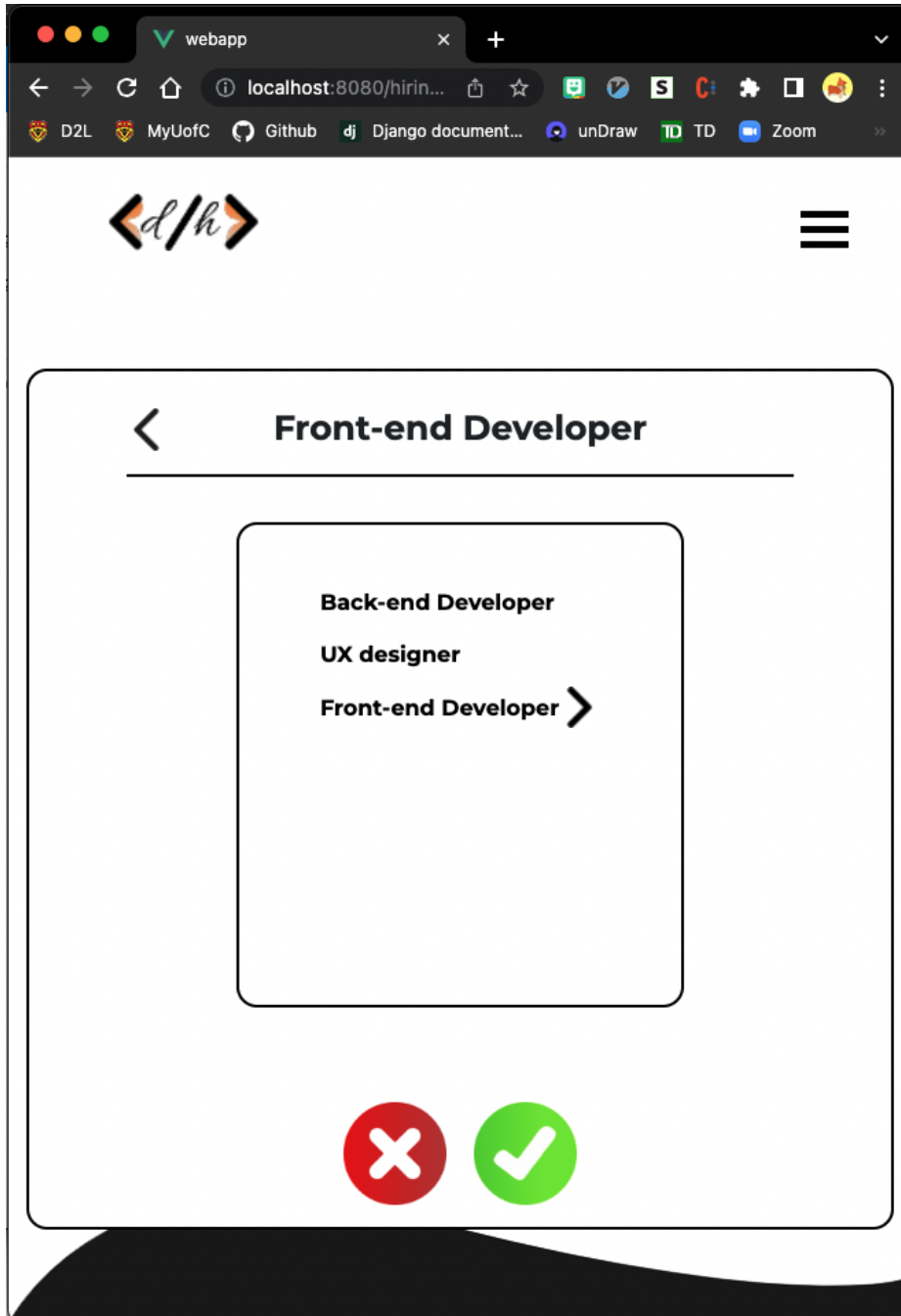


Figure 10: Explore page for hiring managers when screen is narrow and when hiring manager has added job postings and viewing the information that would otherwise be on the left sidebar.

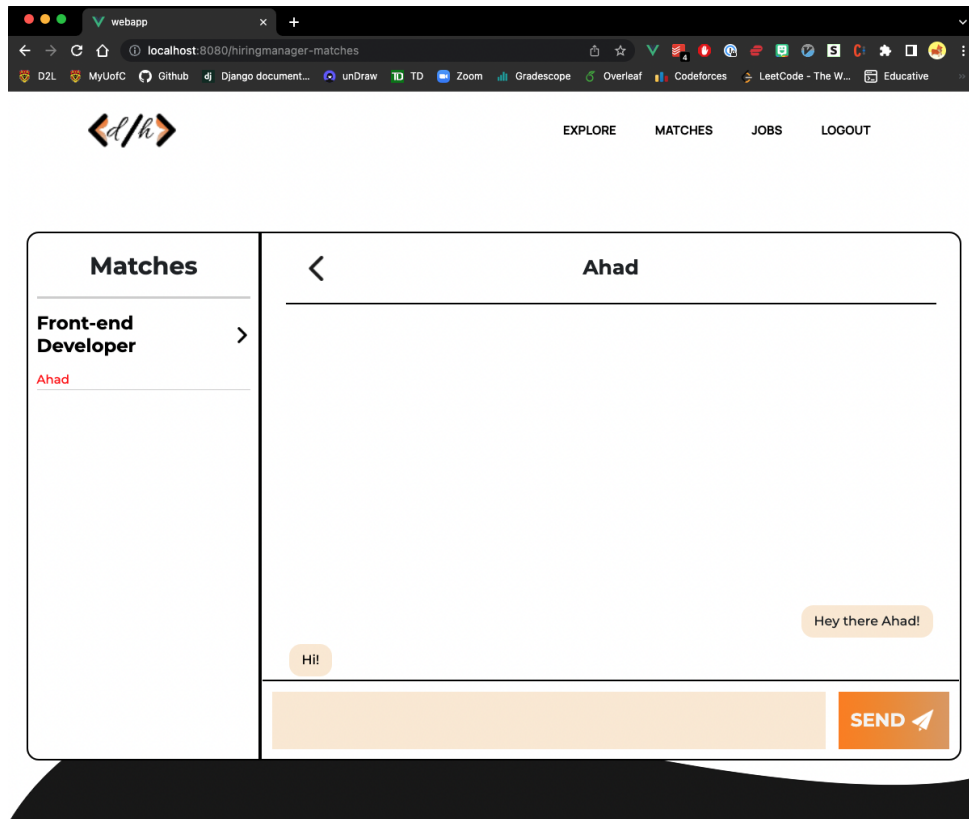


Figure 11: Main matches page for hiring managers for when a job-seeker is selected so that the hiring manager can chat with the job-seeker

The screenshot shows a web browser window with the URL `localhost:8080/createaccount`. The page features a profile creation form. At the top left is a profile picture placeholder with a 'Choose Files' button and the text 'No file chosen'. To the right are input fields for 'First Name' and 'Last Name', and a text area for 'About you'. Below these is a section titled 'What types of jobs are you looking for?' with the instruction 'Choose up to thee different job types'. This section contains three dropdown menus labeled 'Choice 1', 'Choice 2', and 'Choice 3'. Further down are sections for 'Skills' and 'Relevant Experience', each with a plus icon to add more items. The 'Relevant Experience' section includes input fields for 'Company', 'Job Title', 'Job Type', and 'Years Worked', along with a 'Job description' text area.

Figure 12a: Top half of the create profile page for job-seekers

Skills +

Relevant Experience +

Company:

Job Type:

Job description:

Job Title:

Years Worked:

Socials +

Platform:

Link:

Education +

Education Type:

Institution:

Start date:

End date:

Finish

Figure 12b: Bottom half of the create profile page for job-seekers

EXPLORE MATCHES PROFILE LOGOUT

Data Scientist
Netflix
Seattle
Est. \$ 175000
Start date: 2022-04-22

Job Description
The ideal candidate is clearly passionate about new opportunities and has a demonstrated track record of success in delivering new features and products. A commitment to team work, hustle, and strong communication skills (to both business and technical partners) are absolute requirements. Creating reliable, scalable, and high-performance products requires exceptional technical expertise, a sound understanding of the fundamentals of Software Engineering, and practical experience building large-scale distributed systems. This person has thrived and succeeded in delivering high quality technology products/services in a hyper-growth environment where priorities shift fast.

Required Skills
• Git hub
• Git
• SQL

Experience
≥1 years in relevant field

X ✓

Figure 13: Explore page for job-seekers

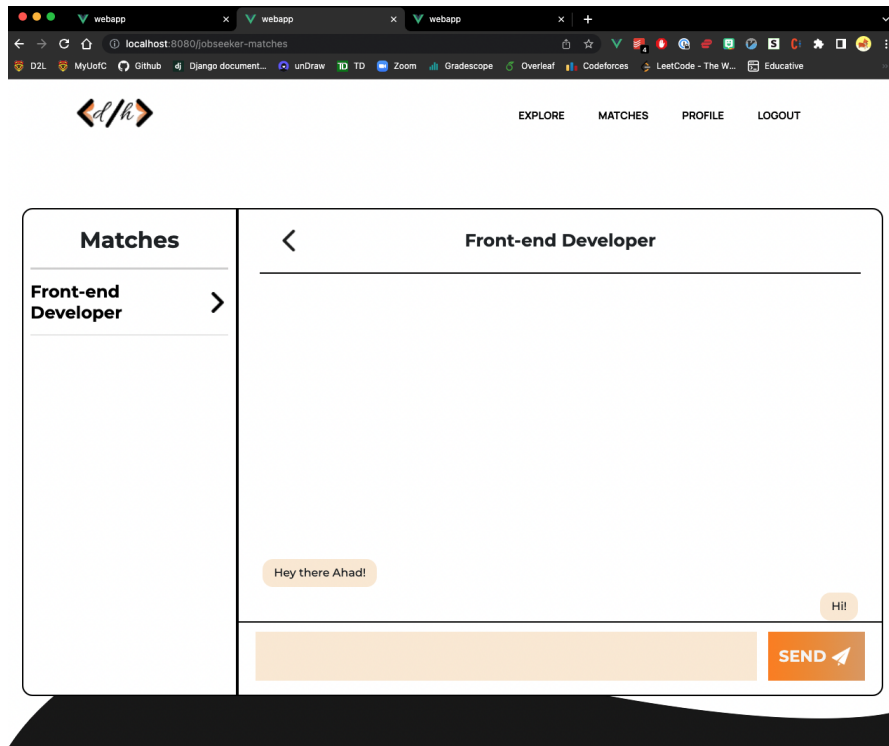


Figure 14: Matches page for job-seekers

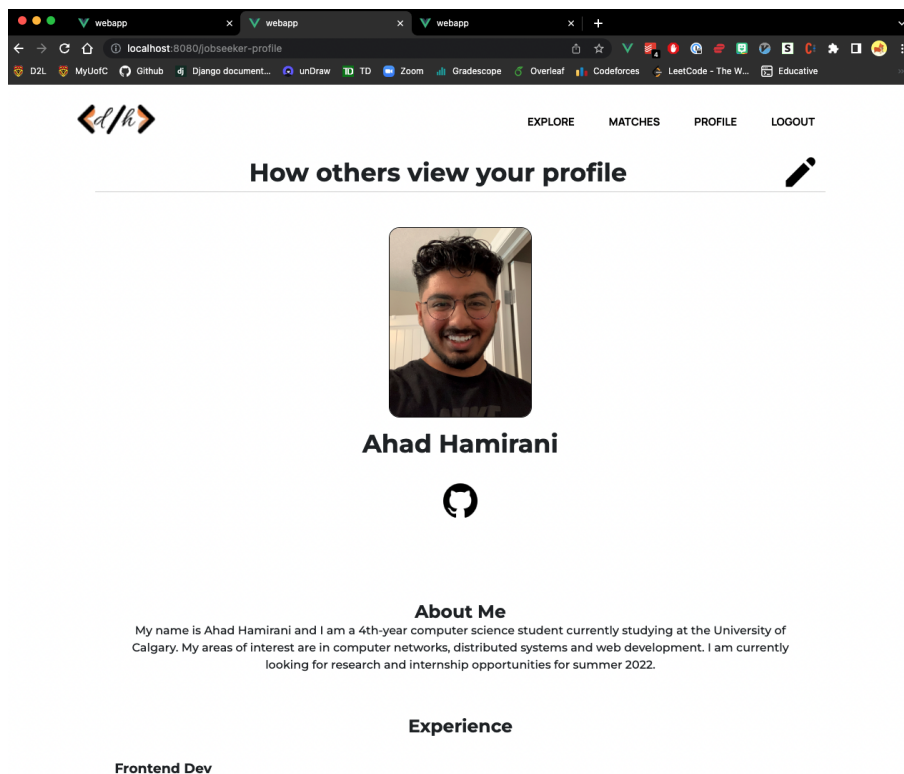


Figure 15: Profile page for job-seeker

EXPLORE MATCHES PROFILE LOGOUT

First Name
Ahad

Last Name
Hamirani

About you
My name is Ahad Hamirani and I am a 4th-year computer si

Choose Files No file chosen

What types of jobs are you looking for?
Choose up to three different job types

Choice 1
Back-end Developer

Choice 2
data scientist

Choice 3
Front-end Developer

Skills
Git hub Git SQL C++

Relevant Experience

Company
BlackBerry QNX

Job Type
Front-end Developer

Job description
Demonstrated a deep understanding of embedded systems and POSIX-I

Job Title
Frontend Dev

Years Worked
1

Socials

Figure 16: Edit profile page for job-seekers

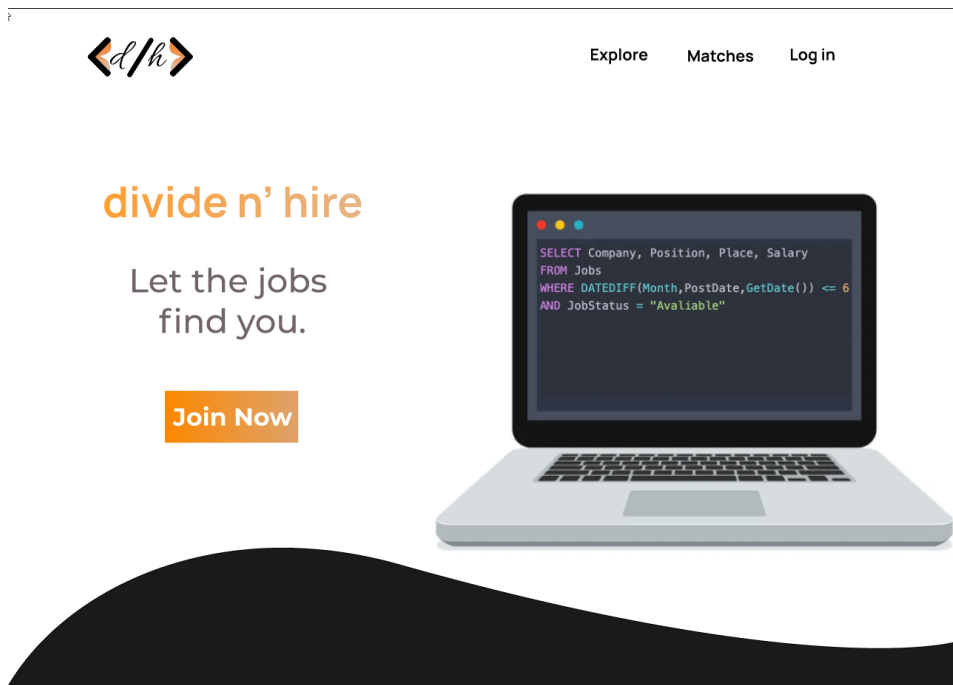


Figure 17: Mockup landing page for both hiring managers and job-seekers

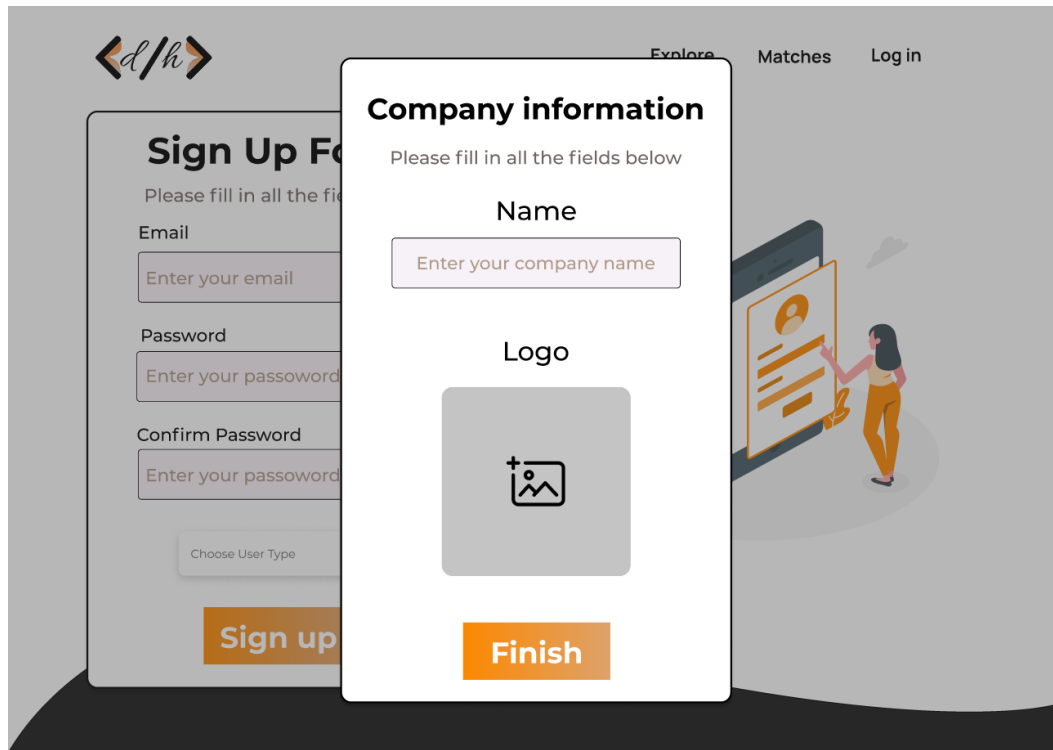


Figure 18: Mockup company information popup page for hiring managers

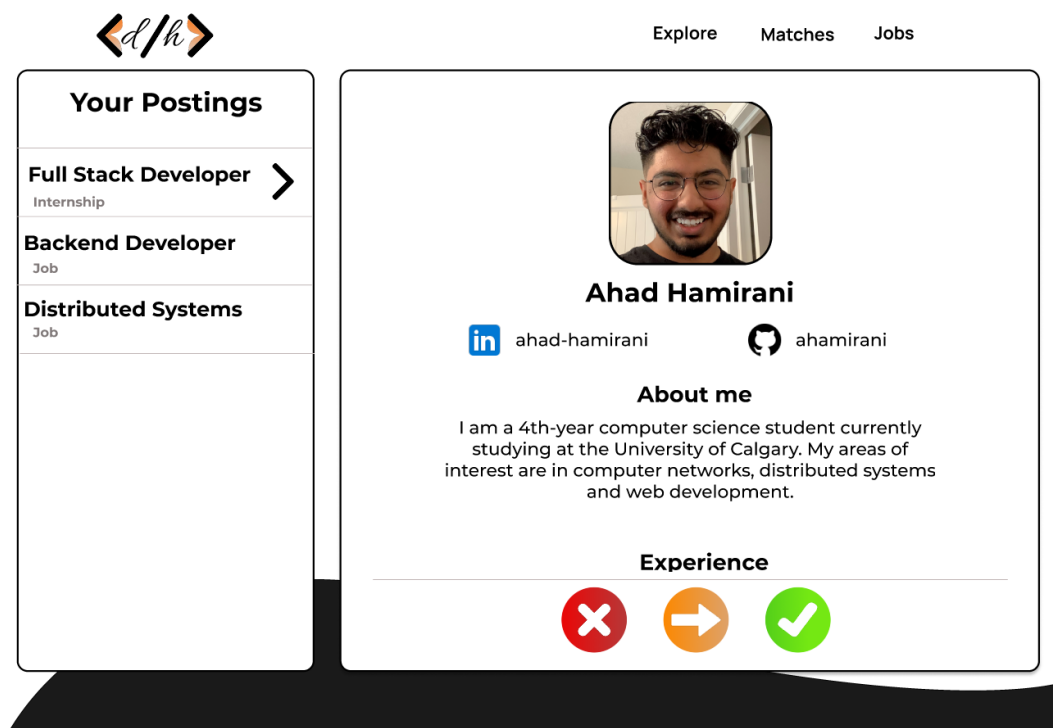


Figure 19: Mockup Explore page for hiring managers