SW Engineering CSC 648/848 Fall 2022

The Hiring Guru

Team 2: Binary Brains
Farhan Haider (Team Lead/Backend Lead): shaider1@sfsu.edu
Kenny Leong (Frontend Lead/GitHub Master): kleong2@mail.sfsu.edu
Mamadou Bah (Frontend): mbah1@mail.sfsu.edu
Khushi Khanna (Backend) kkhanna@sfsu.edu
Eric Leow (Frontend): elow@mail.sfsu.edu

Milestone #2

October 20th, 2022

Row	Document Version	Review Date	Description
1	Version 1	9/22/2022	Milestone 2 was assigned, Document was first created with sections to be filled out
2	Version 2	9/26/2022	Went over Milestone 1 feedback and assigned work to revise it to gain filled credit.
3	Version 3	9/28/2022	Went through Milestone 2 together and assigned less high maintenance tasks to different team members to have done.
4	Version 4	10/6/2022	Received Constructive Feedback from the professor and planned out how we were going to further improve our progress
5	Version 5	10/8/2022	Met up for a quick meeting to work on Trello tasks given to us

6	Version 6	10/8/2022	Had tasks assigned to be due 10/14/2022 for the home page, involving both Frontend and Backend Team
7	Version 7	10/15/2022	Individually met with team lead for designated updates and future assignments to complete Milestone.
8	Version 8	10/17/2022	Made sure all team members had completed prior assignments and split up any necessary work that was left for the document
9	Version 9 Final Version	10/19/2022	Had our final M2 Meeting with a check-in and final touches for M2. Spend the evening wrapping everything up
10	Version 10 - V2 Updates	11/09/2022	Updates for M3V2

Table of Contents

1 - Data Definitions V3…….……4

2 - Prioritized Functional Requirements V3…….....7

3 - Use Cases and Storyboard……..…….……18

4 - High Level Database Architecture and Organization…….....33

5 - High Level APIs and Main Algorithm.....37

6 - High Level UML Diagrams 39

7 - High Level Application Network and Deployment Diagrams.....40

8 - Identifying *Actual* Key Risks.....42

9 - Project Management.....…….....44

10 - Detailed list of Contributions.....…….....45

1 - Data Definitions

HR Manager

An HR Manager does the following:

- An HR manager shall be able to be the main actor and the most frequent user of this application, responsible for managing the complete hiring process.
- A HR Manager shall be able to schedule the various steps of the application process
- A HR Manager shall be able to reach out to Candidates throughout the application process
- A HR Manager shall be able to move the Candidate along the recruitment pipeline given positive feedback from Interviewer
- A HR Manager shall be able to be able to engage with communication channels
- A HR Manager shall be able to view hiring insights to find the most suitable Candidate
- A HR Manager shall be responsible for ensuring the onboarding process is completed by the Candidate

Candidate

A Candidate does the following:

- The Candidate is a prominent user within the application, responsible for going through the entire Recruitment Pipeline Process from start to finish
    - Involves being aware of open positions and requirements of each position
    - Involves completing each step of the recruitment process on time
    - Involves remaining in communication during the recruitment process with various other users

Executive

An Executive does the following:

- The Executive entity is an entity that has the same influence as the HR Manager but is one above the HR Manager.
- The Executive shall be able to do all of the tasks that the HR Manager can, such as:
    - An Executive is able/responsible for scheduling the various steps of the application process
    - An Executive is able to reach out to Candidates throughout the application process
    - An Executive moves the Candidate along the recruitment pipeline given positive feedback from Interviewer
    - An Executive is able to be able to engage with communication channels

    - An Executive is able to view hiring insights to find the most suitable Candidate
    - An Executive is responsible for ensuring the onboarding process is completed by the Candidate
- An Executive includes creating referral programs and overseeing the entire recruitment process and managing the progress of the other users within the recruitment pipeline

Tech Recruiter/Marketing

A Tech Recruiter does the following:

- A Tech Recruiter, or the Marketing Team, is the main user that starts the entire Recruitment Pipeline Process within the Hiring Guru, responsible for hiring the Candidate that will undergo the process.
- A Tech Recruiter shall be able to promote the job on various platforms and figure out what is the best platform and marketing strategy to target the appropriate group of Candidates
- A Tech Recruiter shall be responsible for presenting suitable Candidates needed for the job description

Interviewer

An Interviewer does the following:

- An Interviewer is a user who is an employee within the desired company by the general user and interviews them to see if they qualify for the specific role.
- An Interviewer shall be responsible for scoring each of the Candidate's based on the scoresheet provided by the company
- As an Interviewer, it's crucial to understand the skills and experience that are necessary for the job description

Job Application

A Job Application does the following:

- A Job Application shall be created by an Employee
- A Job Application shall be filled out by a Candidate
- A Job Application shall contain profile information of many candidates
- A Job Application shall be able to be modified by an Employee

Employee

An Employee does the following:

- An Employee shall be able to view candidates in the Recruitment Pipelines Process

- An Employee shall be able to interview Candidates
- An employee shall be able to schedule interview of candidates
- An employee shall be able to setup Permissions and Groups
- An authorized employee shall be able to shortlist Applications
- An authorized employee setup Recruitment Pipelines Process
- An authorized employee shall be creating many Job positions
- An authorized employee shall be able to create many Job roles

General User

A General User can do the following:

- A General User is a user who is only exploring the application and has the most basic features accessible to them, most often an unauthenticated user.

Group

A Group can do the following:

- A group can have many employees
- A group can be assigned multiple permissions

Permission

A Permission can do the following:

- A Permission can be assigned to multiple Groups

2 - Prioritized Functional Requirements

P1

Authorized User

Requirement ID	Requirement Statement
FRAUTH001	Authorized user shall be able to log out anytime
FRAUTH002	Authorized user shall be able to create an organization
FRAUTH003	Authorized user shall be able to modify organization detail
FRAUTH004	Authorized user shall be able to invite employees to the organization
FRAUTH005	Authorized user shall be able to create permission groups
FRAUTH006	Authorized user shall be able to add employees to user groups
FRAUTH007	Authorized user shall be able to remove employees from user groups
FRAUTH008	Authorized Users shall be able to create a new role (e.g., QA Engineer, Software Engineer, etc.)
FRAUTH009	Authorized User shall be able to update a role
FRAUTH010	Authorized User should be able to see the list of all roles
FRAUTH011	Authorized User shall be able to remove a role from the system
FRAUTH012	Authorized User shall be able to open a new position for a particular role
FRAUTH013	Authorized User shall be able to close a listed position
FRAUTH014	Authorized User shall be able to see all the applications for a specific position
FRAUTH015	Authorized User shall be able to add referral for a specific position
FRAUTH016	Authorized User shall be able to see all the referrals for a specific position
FRAUTH017	Authorized User shall be able to drop an application
FRAUTH018	Authorized User shall be able to design the hiring process for a position
FRAUTH019	Authorized User shall be able to update the hiring process for a position

FRAUTH020	Authorized User shall be able to see the hiring process for a position
FRAUTH021	Authorized User shall be able to accept an application for a specific position
FRAUTH022	Authorized User shall be able to see a list of all the accepted applications
FRAUTH023	Authorized User shall be able to remove a Candidate from the hiring process
FRAUTH024	Authorized User shall be able to add an Employee
FRAUTH025	Authorized User shall be able to view a list of all Employees
FRAUTH026	Authorized User shall be able to remove an Employee from the company
FRAUTH027	Authorized User shall be able to schedule an interview of the Candidate
FRAUTH028	Authorized User shall be able to send a calendar invite to the Interviewers and the Candidate
FRAUTH029	Authorized User shall be able to see the schedule detail of an interview of the candidate
FRAUTH030	Authorized User shall be able to see the detail of an interview of the candidate
FRAUTH031	Authorized User shall be able to schedule a programming test of the candidate
FRAUTH032	Authorized User shall be able to update the time and date of a programming test
FRAUTH033	Authorized User shall be able to add the feedback/decision of a programming test
FRAUTH034	Authorized User shall be able to attempt a programming test
FRAUTH035	Authorized User should be able to create a job post on LinkedIn
FRAUTH036	Authorized User should be able to create a new advertising campaign on LinkedIn
FRAUTH037	Authorized User should be able to analyze LinkedIn profiles using Artificial Intelligence

FRAUTH038	Authorized User shall be able to send a message to the matching profiles to see if they are interested
FRAUTH039	Authorized User shall be able to add profiles of interest to the recruitment pipeline
FRAUTH030	Authorized User shall be able to like a listed position
FRAUTH041	Authorized User shall be able to update his profile
FRAUTH042	Authorized User shall be able to search for job by the posted time
FRAUTH043	Authorized User shall be able to post the job advertisement on LinkedIn
FRAUTH044	Authorized User shall be able to schedule a programming test for the candidate
FRAUTH045	Authorized User shall be able to schedule an interview of the candidate
FRAUTH046	The system shall send email to the candidate once programming test is scheduled
FRAUTH047	The system shall send email to the candidate once an interview is scheduled
FRAUTH048	Authorized User shall be able to send acceptance email to the candidate
FRAUTH049	Authorized User shall be able to send rejection email to the candidate
FRAUTH050	Authorized User shall be able to send a background check must to a candidate, which must be passed before being able to accept a job offer
FRAUTH051	Authorized User shall be able to review each applicant’s resume and experience prior to interviewing
FRAUTH052	Authorized User shall be able to determine how to score candidate's responses based off of company’s expectations
FRAUTH052	Authorized User shall be able to answer Candidate's questions appropriately
FRAUTH053	Authorized User shall be able to administer situational and personality tests to narrow the list of Candidates
FRAUTH054	Authorized User shall be able to write a job description

FRAUTH055	Authorized User shall be able to ensure that the employment paperwork is correct and complete
FRAUTH056	Authorized User shall be able to ensure that the application should be easy to complete
FRAUTH057	Authorized User shall be able to schedule a phone screen
FRAUTH058	Authorized User shall be able to schedule a skills test
FRAUTH059	Authorized User shall be able to ensure appropriate questions are on the skills test related to the job requirement
FRAUTH060	Authorized User shall be able to schedule a new time for the skills test if Candidate is unable to complete it within the provided time frame
FRAUTH061	Authorized User shall be able to allow for convenient and quick channels of communication for the candidate to ask questions throughout the interview process

General User

Requirement ID	Requirement Statement
FRGENR000	General user shall be able to sign up directly for an individual account
FRGENR001	General user shall be able to signup using Google account
FRGENR002	General user shall be able to signup using a Microsoft account
FRGENR003	General user shall be able to login using Google account
FRGENR004	General user shall be able to login using Microsoft account
FRGENR005	General user shall be able to login using Username and Password
FRGENR006	General user shall be able recover his/her account using email
FRGENR007	General User shall be able to go back to the home screen anytime by clicking on the HiringGuru logo
FRGENR008	Authorized User shall be able to like a listed position
FRGENR009	General User shall be able to see the HiringGuru about page in the page footer

FRGENR010	General User shall be able to see the product features in the home page
FRGENR011	General User shall be able to search for jobs on the home page
FRGENR012	General User shall be able to apply for a job

P2

Authorized User

Requirement ID	Requirement Statement
FRAUTH062	System should automatically add an accepted application to the hiring process
FRAUTH063	System should automatically move an accepted application to the next step when a step is successful
FRAUTH064	System shall automatically notify the Candidate about the success/failure in a step
FRAUTH065	Authorized User shall be able to mark an Employee as an Interviewer
FRAUTH066	Authorized User shall be able to see a list of all Interviewers
FRAUTH067	Authorized User shall be able to remove an Employee from the Interviewer status
FRAUTH068	System should automatically send a calendar invite to the Interviewers and the Candidate
FRAUTH069	Authorized User shall be able to update the details of an interview - date, interviewers etc.
FRAUTH070	Authorized User shall be able to see the results of the programming test of a Candidate
FRAUTH071	Authorized User should be able to create a job post on Handshake
FRAUTH072	Authorized User should be able to update job posts created using the System
FRAUTH073	Authorized User should be able to pause an advertising campaign

FRAUTH074	Authorized User should be able to see statistics of an advertising campaign
FRAUTH075	Authorized User should be able to scrape profiles from LinkedIn given a set of keywords
FRAUTH076	Authorized User shall be able to save the list of matching profiles
FRAUTH077	Authorized User shall be able to ensure the Interviewer understands what skills and experience the job require
FRAUTH078	Authorized User shall be able to ensure the Interviewer has the proper skills to take the interview
FRAUTH079	Authorized User shall be able to create relevant application questions
FRAUTH080	Authorized User shall be able to identify the hiring need
FRAUTH081	Authorized User shall be able to determine if an external hire- go through usual advertising channels
FRAUTH082	Authorized User shall be able to perform a reference check
FRAUTH083	Authorized User shall be able to create a checklist of all of the paperwork related to employment the Candidate needs to fill out after accepting an offer
FRAUTH084	Authorized User shall be able to give Candidate ample time to complete the employment paperwork
FRAUTH085	Authorized User shall be able to, if no candidates meet the hiring criteria, decide whether to restart the hiring process
FRAUTH086	Authorized User shall be able to ask the candidate about their salary expectations
FRAUTH087	Authorized User shall be able to, if they wish to restart the hiring process, decide how to adjust the hiring criteria to yield more candidates
FRAUTH088	Authorized User shall be able to, if they wish to restart the hiring process, decide how to advertise the position to more favorable candidates

FRAUTH089	Authorized User shall be able to, if the candidate is ready to accept the job offer, decide which elements of the offer letter are negotiable
FRAUTH090	Authorized User shall be able to ensure that Artificial Intelligence is able to parse through the candidate's resume to fill out fields in the application
FRAUTH091	Authorized User shall be able to give a certain time frame for the candidate to accept the offer letter
FRAUTH092	Authorized User shall be able to create convenient methods for applicants to be able to withdraw from the application process at any time
FRAUTH093	Authorized User shall be able to provide detailed feedback on what they liked about the candidate's performance but also reservations they might have
FRAUTH112	Authorized User shall be able to create activities for the Candidate to do
FRAUTH113	The application should be able to have a total cost calculator
FRAUTH114	The application should be able to have a digital document process
FRAUTH115	The application should be able to have a workload manager
FRAUTH116	The application should be able to have a candidate portal
FRAUTH117	Screening function for making sure the candidate is cleared
FRAUTH118	Authorized User shall be able to create a database for tracking candidate’s progress
FRAUTH119	Authorized User shall be able to assign role to candidate
FRAUTH120	The application should be able to provide job-specific programs for each candidate
FRAUTH121	The application should be able to grant access to technical tools needed by the candidate
FRAUTH122	The application should be able to have a paid or unpaid boarding process
FRAUTH123	The application should be able to allocate mentor for Candidate
FRAUTH124	The application should be able to create a schedule for the Candidate

FRAUTH125	The application should be able to organize the workspace
FRAUTH126	Introduce the candidate to technologies they will need to learn how to use
FRAUTH127	Authorized User shall be able to prepare orientation with other Candidates
FRAUTH128	The application shall have an administration tools for the interviewer
FRAUTH129	Authorized User shall be able to Take in any comments or complaints that the candidate might have
FRAUTH130	Authorized User shall be able to Organize meetings between the employer and the candidates
FRAUTH131	Manage any health benefits that should be given to the employee
FRAUTH132	Authorized User shall be able to request Employment background screening
FRAUTH133	Authorized User shall be able to Send offer letter to candidate
FRAUTH134	Authorized User shall be able to Email reminders of the next steps in onboarding

General User

Requirement ID	Requirement Statement
FRGENR013	General user shall be able to book a product demo
FRGENR014	General user shall be enable or disable cookies
FRGENR015	General user shall be able to chat with the organization bot

P3

Authorized User

Requirement ID	Requirement Statement

FRAUTH094	System should automatically remove the Candidate from the hiring process if a step is failed
FRAUTH095	Authorized User shall be able to view the detail of an Employee
FRAUTH096	Authorized User shall be able to update the detail of an Employee
FRAUTH097	Authorized User shall be able to allow an interviewer to set their availability time
FRAUTH098	Authorized User shall be able request Candidate availability for an interview
FRAUTH099	System should automatically schedule an interview of the Candidate
FRAUTH100	System should automatically schedule a programming test if it is the current step
FRAUTH101	Authorized User should be able to create a job post on Stackoverflow
FRAUTH102	Authorized User should be able to create a new advertising campaign on YouTube
FRAUTH103	Authorized User should be able to create a new advertising campaign on StackOverflow
FRAUTH104	Authorized User should be able to calculate the budget and allocate money for specific advertising campaigns
FRAUTH105	Authorized User shall be able to calculate budget to see how many open positions are possible to create
FRAUTH106	Authorized User should be able to calculate the budget to allocate for training interviewers on how to conduct an interview
FRAUTH107	Authorized User shall be able to talk about the day-to-day duties and inform the candidate about the work culture
FRAUTH108	Authorized User shall be able to identify if the hire is an external or internal applicant
FRAUTH109	Authorized User shall be able to determine if an internal hire- minimal advertising is needed

FRAUTH110	Authorized User shall be able to design a recruitment pipeline for individual candidates
FRAUTH111	Authorized User shall be able to provide specific feedback as to why they didn't pass that round
FRAUTH135	Authorized User shall be able to prepare new Candidate for training
FRAUTH136	Authorized User shall be able to create a new onboarding program
FRAUTH137	The application should be able to have a calculate candidate hours
FRAUTH138	The application should be able to have a decrease wage
FRAUTH139	The application should be able to have a give raise
FRAUTH140	The application should be able to have a calculate trust level
FRAUTH141	The application should be able to have a survey candidate
FRAUTH142	The application should be able to have a find jobs for the candidate within company
FRAUTH143	The application should be able to have a create presentations
FRAUTH144	The application should be able to host meetings
FRAUTH145	The application should be able to have a way to communicate with other data items
FRAUTH146	Authorized User shall be able to create tasks for candidate
FRAUTH147	The application should be able to have a deadline setter
FRAUTH148	Authorized User shall be able to set goals for the candidate to complete
FRAUTH149	Authorized User shall be able to find candidate function
FRAUTH140	Authorized User shall be able to pause candidate function
FRAUTH141	Authorized User shall be able to remove candidate function
FRAUTH142	The application should be able to have memory allocation for Candidate information

FRAUTH143	The application should be able to organize team dynamics
FRAUTH144	The application should be able to organize team dynamics
FRAUTH145	The application should be able to organize where and how they will be working
FRAUTH146	Authorized User shall be able to do a performance review scheduling
FRAUTH147	Manage space for Candidates
FRAUTH148	Authorized User shall be able to display information needed by the candidate

General User

Requirement ID	Requirement Statement
FRGENR016	General user shall be able to see product pricing
FRGENR017	General User shall be able to see statistics of a job post

3 - Use Cases and Storyboards with UI Mockups

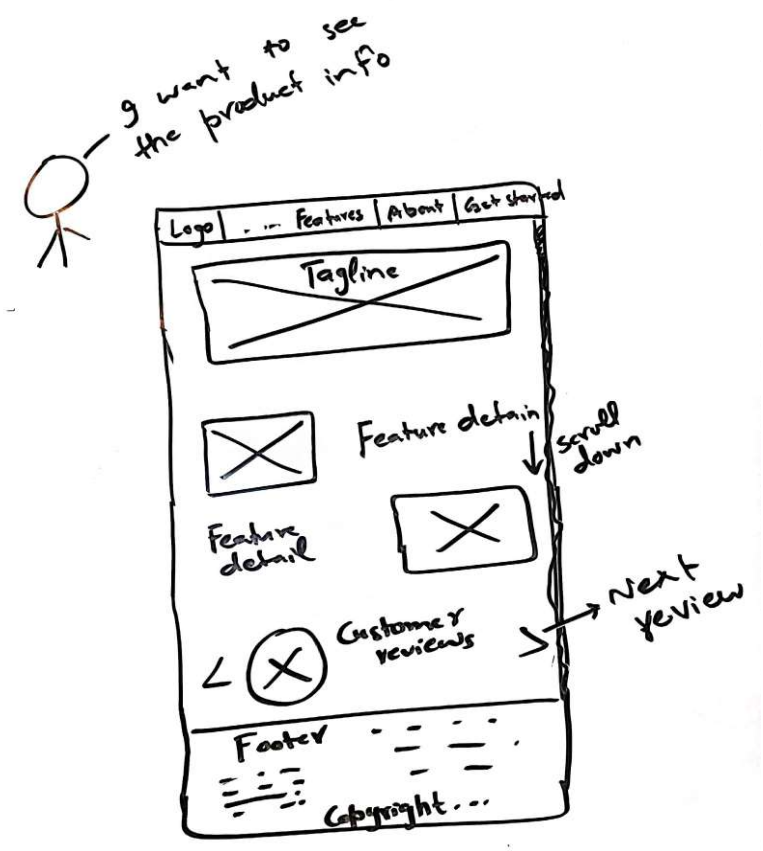
This section provides high-level mockups/storyboards for prioritized use cases.

Use case - See product information

Description

This use case starts when a general user wants to see our product's information which primarily includes the tagline, features, and customer reviews or testimonials. To archive this, the user goes to the *Hiring Guru* landing page, which is the index page of the website. The tagline and logo are visible at the top. The user scrolls down to see various product features which are associated with assistive snapshots of the UI. The user scrolls until he reaches the bottom, where he can see the customer testimonials and reviews. The page footer contains information about copyright and other quick links.

Storyboard

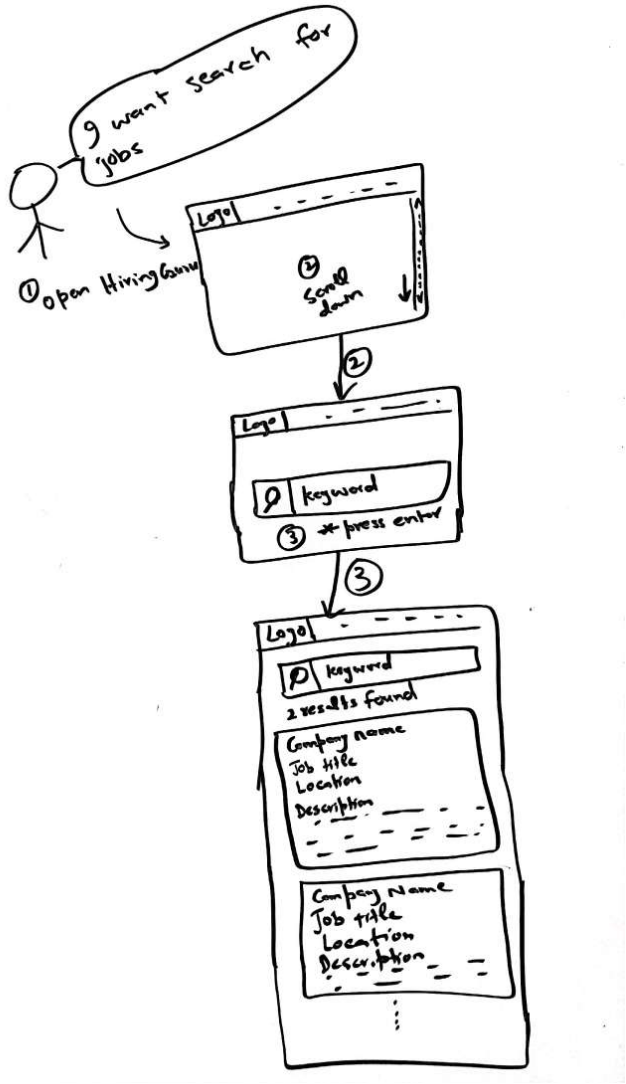


Use Case - Search for jobs

Description

This use case starts when a general user wants to search for jobs listed by companies operating with *Hiring Guru*. To achieve this, the user goes to the *Hiring Guru* main page, which is also the landing page of the application. User scrolls down to the section that displays a search field. The user then types his query into that field and presses the enter button. The system loads a spinner until the search results are being fetched. During this time, the user sees a spinner. Once results are fetched, the user can view a list of jobs along with the company title, job location, and job description.

Storyboard

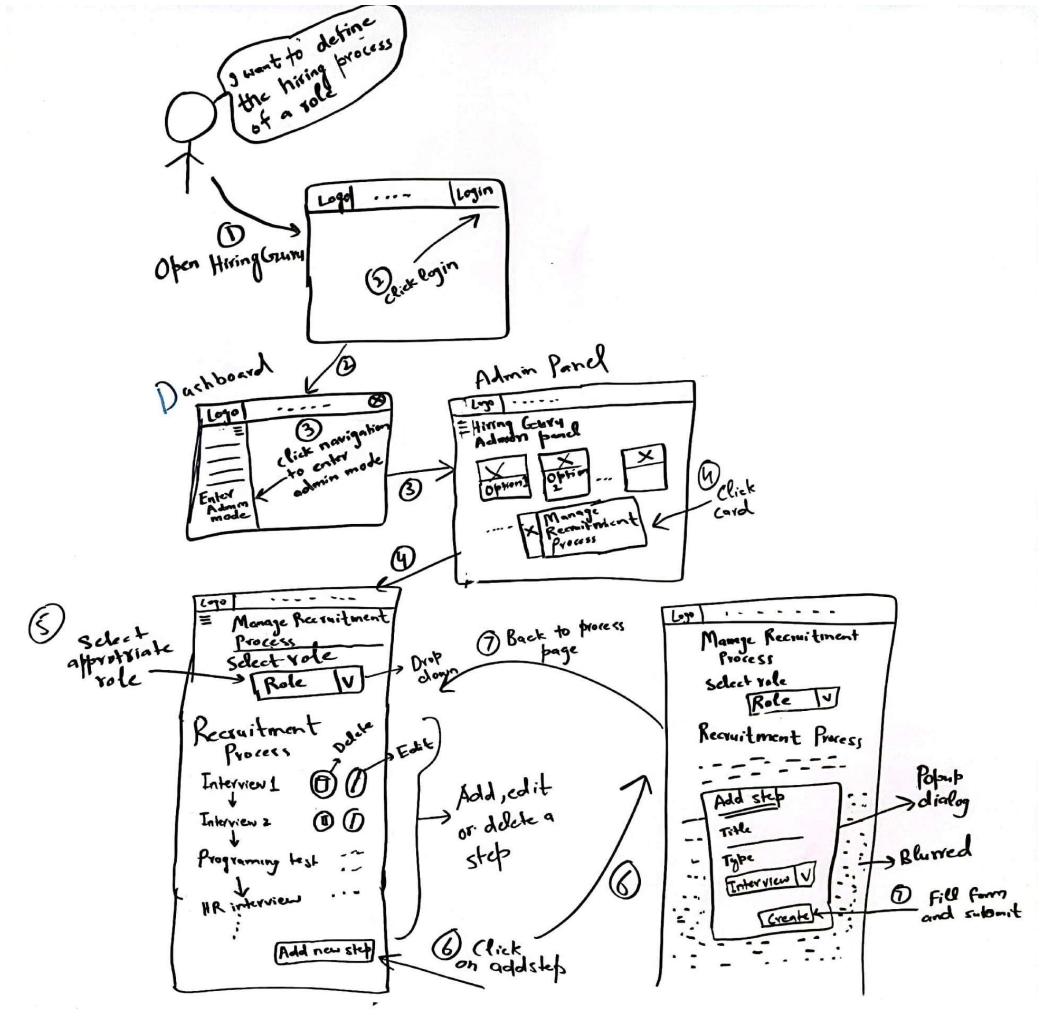


Use Case - Design hiring process for job role

Description

This use case describes the scenario when an Admin user wants to design the hiring process for a particular role (e.g., Software engineer, Graphic Designer). For this, the user first logs in to the system from the landing page with his admin credentials. Then from the side navigation, he goes to the administrative dashboard, which shows various high-level operations like designing the hiring process for a role. Clicking on this card takes the user to the Design Hiring Process page. The user selects a role for which he needs to design the process. The system displays the existing process if present and provides UI for adding new steps, removing an existing step, or updating a step.

Storyboard

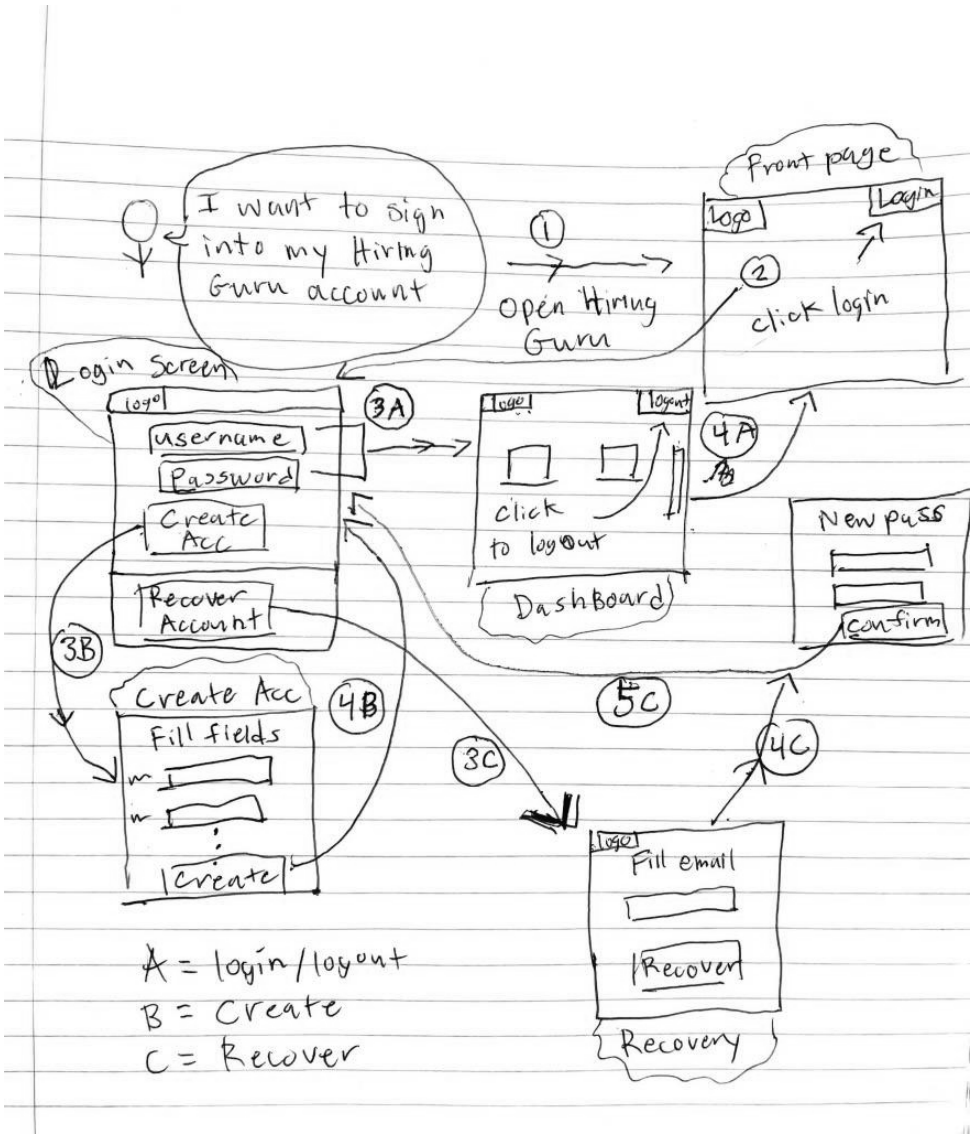


Use Case - Account management

Description

Within this use case, any specific user will be able to do multiple things with their general account for the Hiring Guru. This includes the signup, login, logout, and recover account flow. Upon Hiring Guru, you will be given the prompt to Login or Create an Account. When Creating an Account, the user will need to fill out the required information to have that account created. They can also recover an older account if need be as well. All these various features will be used for users when creating an account.

Storyboard

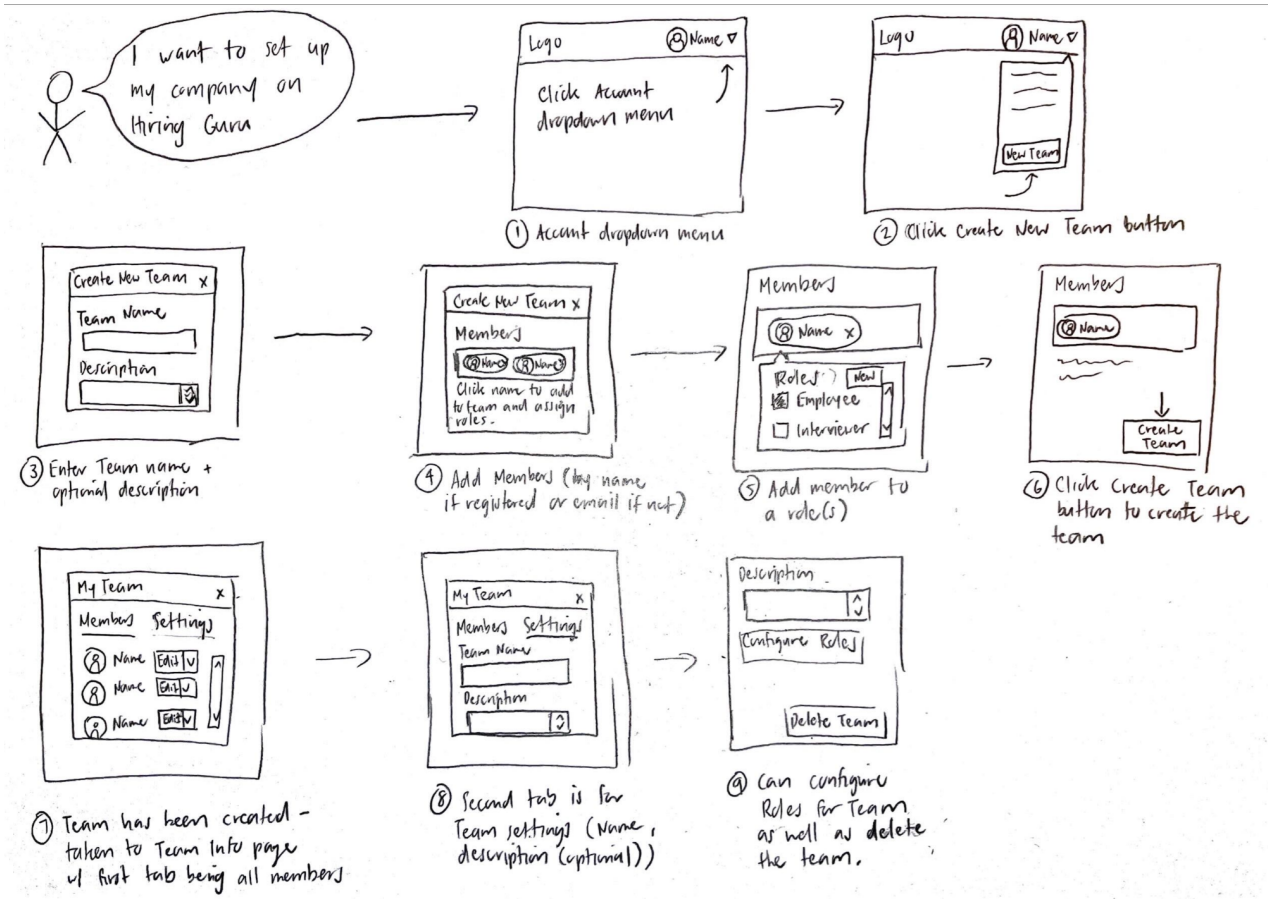


Use Case - Set up a Company

Description

This use case involves creating a new Team or Company within *Hiring Guru*. The idea is an individual will start with an account but will be able to create a Team from their account. This allows the user to set up an organization and allow new or existing members who have an account on the *Hiring Guru* platform to join it as employees. The user will be walked through the options for creating a new Team, and once created, will be able to view the newly-created Team members and their settings.

Storyboard

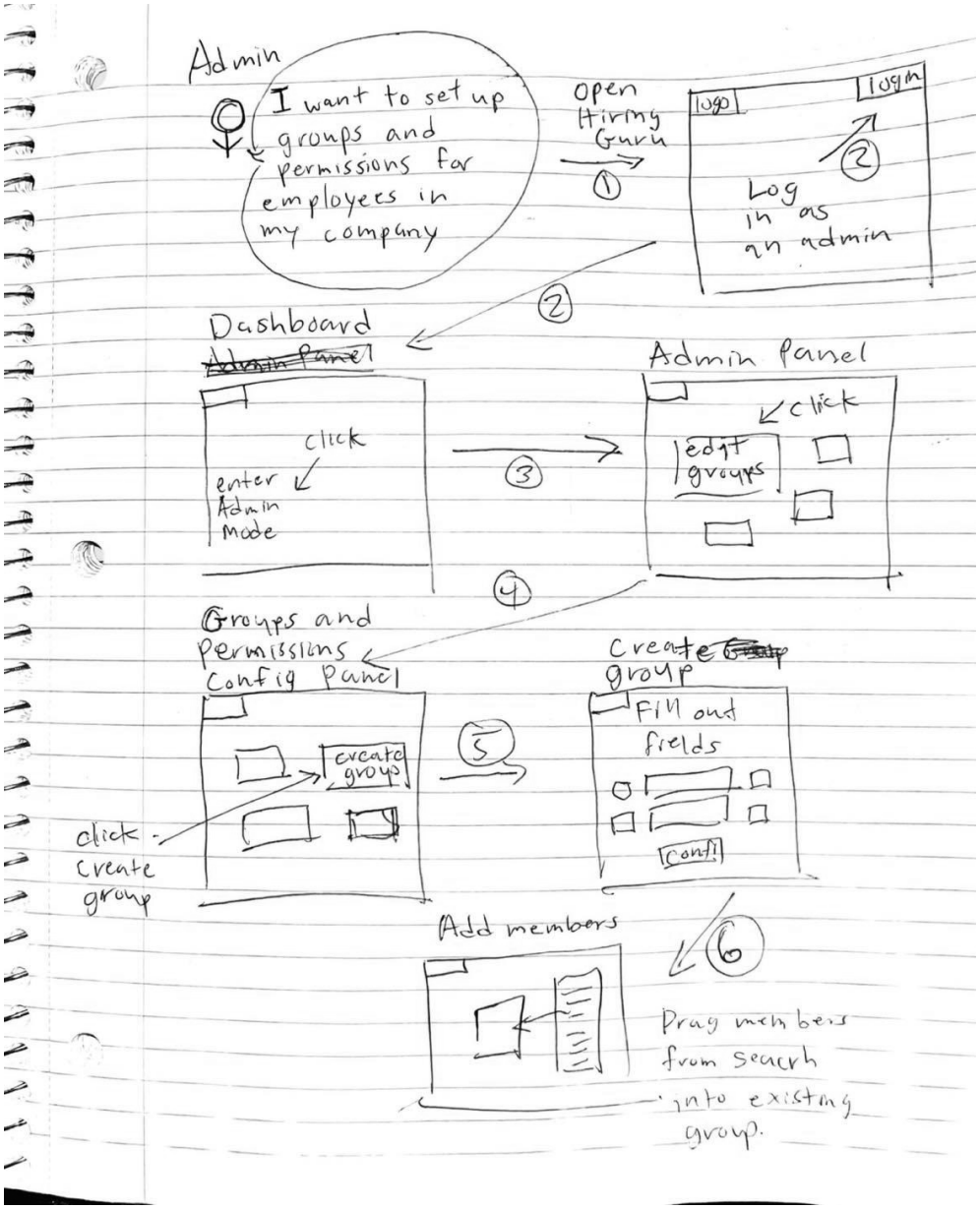


Use Case - Set up user groups and permissions

Description

This is related to the “Set up a company” use case (it comes after that). A company has many employees, and they have various permissions etc. Any authorized user will find a need to create a specific group or specific permissions for certain needs. need to be met. This feature will allow any authorized user to fulfill these needs by swiftly creating a team with the features on the *Hiring Guru*. Some examples will be creating a description for said groups and permissions as well as adding any members that are to be added for these groups.

Storyboard

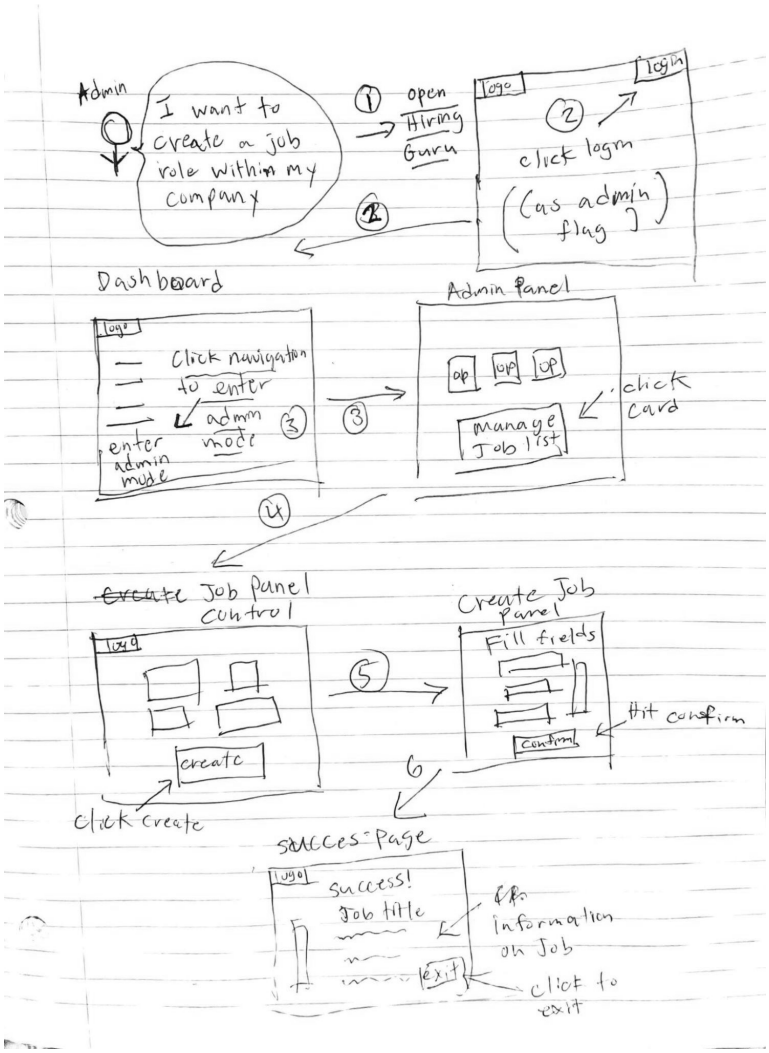


Use Case - Create Job Roles

Description

This use case is about creating a role for the overall company using the Hiring Guru Application. Authorized employees will only be able to do these features where they can create a role and fill out all the necessary requirements for that role. Once the authorized user has completed all the needed sections and filled out the job requirements, then they will be able to submit the job post, which will then be posted on the Hiring Guru application for Candidates and other general users to apply for.

Storyboard

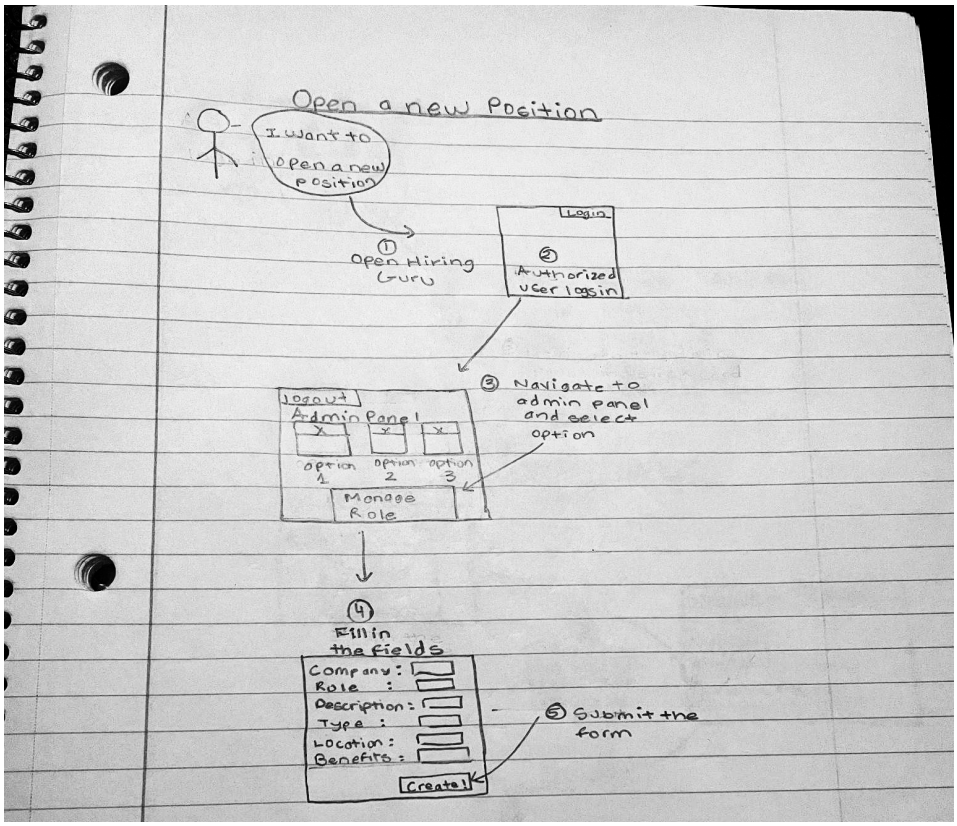


Use Case - Open a new position/job

Description

Whenever a company needs a new position, the Hiring Guru application will have a feature for them to do just that in a very simple and convenient way. Once again, this feature is for authorized users only, where they can log in and select the option to create a few positions. There they can fill out the needed requirements for that position as well as what it entails such as benefits, salary expectations, location, job description, and whatever else is needed. After they complete that, they will then have the option to post that to the Hiring Guru site where other users can apply for said position.

Storyboard

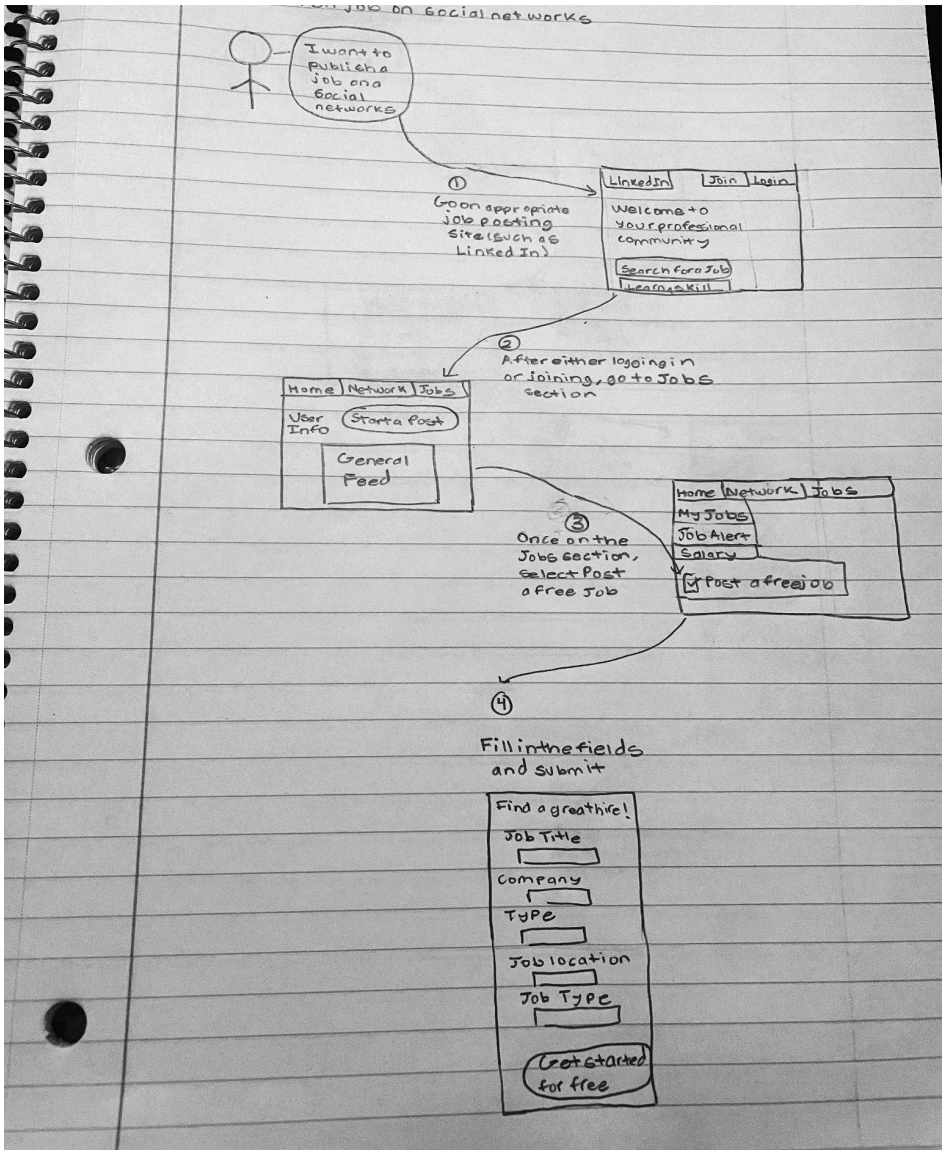


Use Case - Publish job on social networks

Description

Once a job is created, the next thing is to publish it on a social network. This will allow any users from across the Internet to ideally find the job opening and apply for it. After a specific job position is posted, there will be a prompt to promote this job on certain social media platforms. These platforms can include LinkedIn, Twitter, Handshake, Indeed, Instagram, and so much more. In addition, a job link can be generated and copied so any user can share this among other individuals for their own usage.

Storyboard

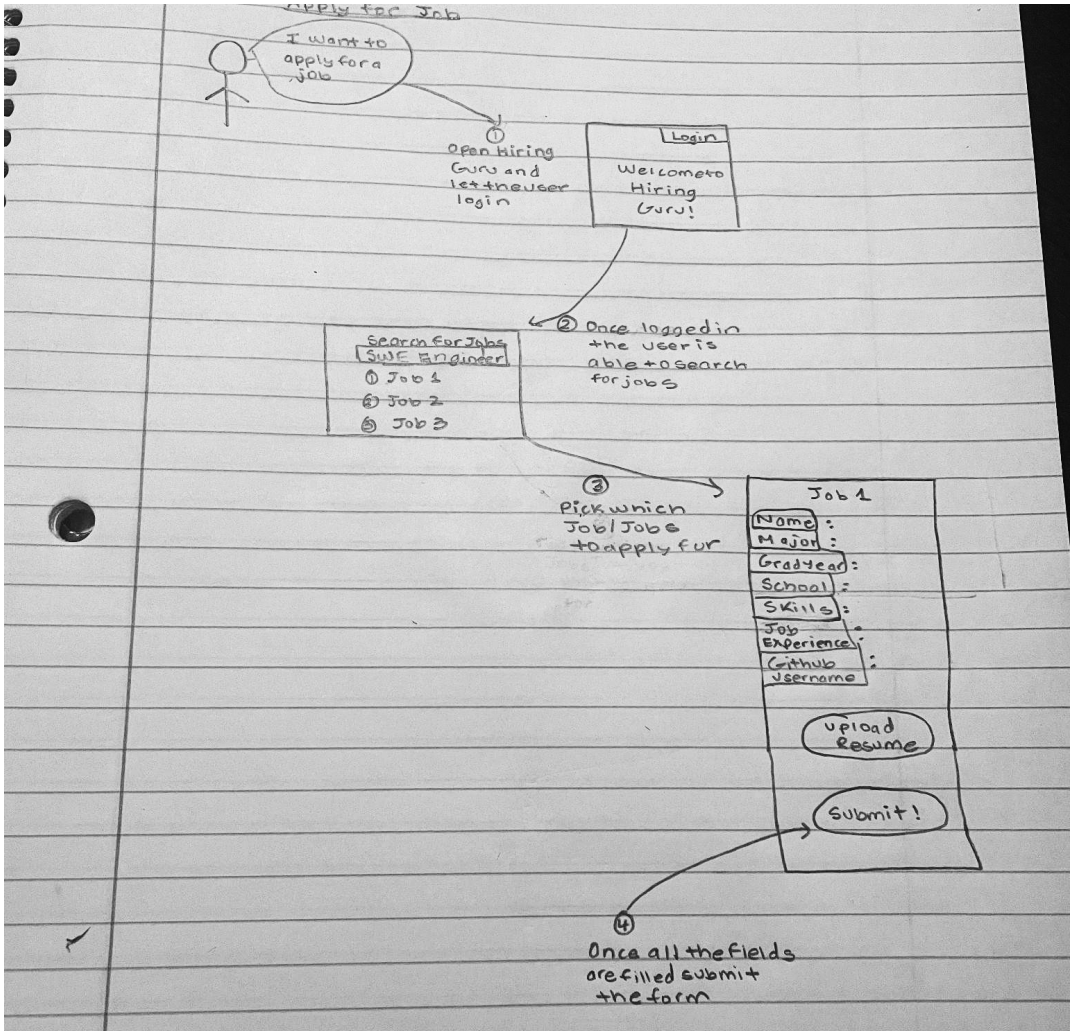


Use Case - Apply for Job

Description

With this use case, a general user is able to apply for a job in the Hiring Guru application. After the user logins, they are able to search for a job role that matches the description of what they want to apply for. An example would be a software engineer. A user is able to see all of the roles that match the description of what they entered while searching for a job, and a user is able to apply for multiple jobs. After selecting a job title, the user is taken to a form in which they are provided many required fields to fill out before they are able to submit the form.

Storyboard

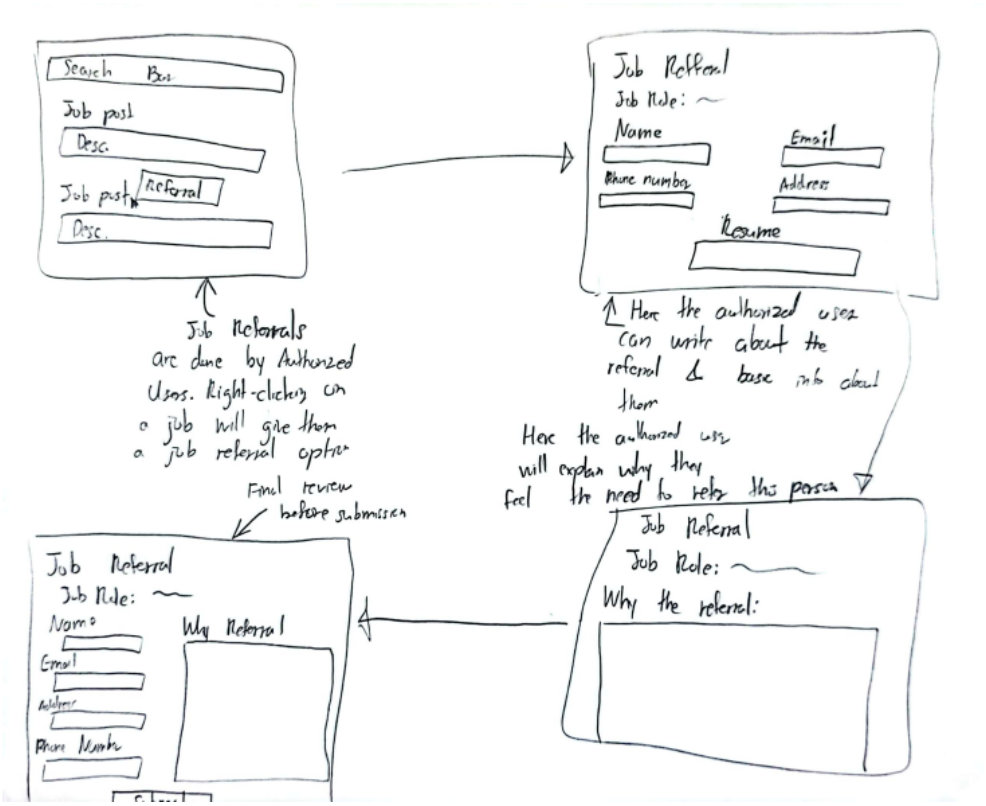


Use Case - Add Referral for Job

Description

Within this use case, an Employee is able to add a referral for a specific job. They can refer any candidate or user that deem to be fit for the role. Upon finding the page with the job post, an authorized user is able to right-click and go to the job referral portion. Here, they can add general information about the user as well as a resume for any other employees to view as well. After that page, they will be given a prompt to write out why they see this specific individual fit for this role. They can write whatever is captivating enough for other employees to agree with them. Once that is completed, they will review all the written information and submit the referral.

Storyboard

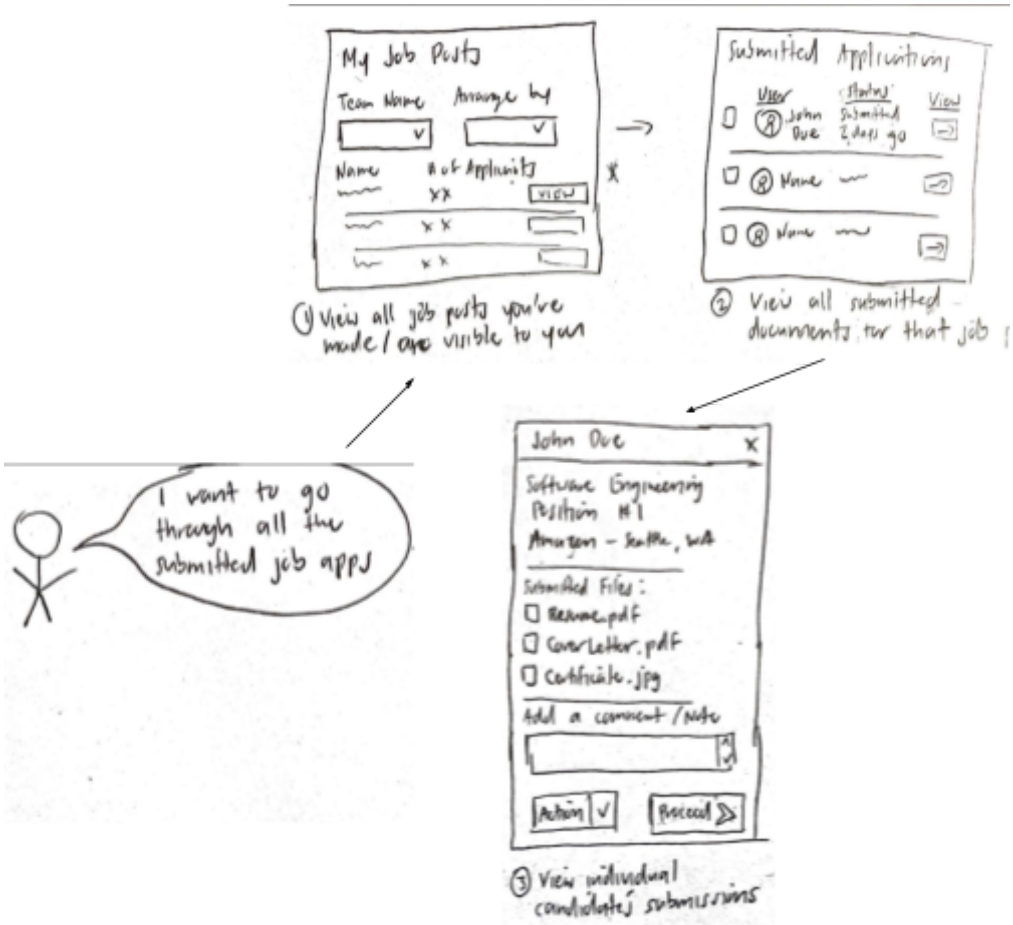


Use Case - Manage job applications

Description

This use case accounts for the user going through submitted job posts for a position where they can accept (send forward through the hiring process) or reject (throw out) applications. The user is able to view all candidates who have made a submission for a chosen position and see a detailed view of a particular candidate's submissions. Within the detailed view, the user can get a closer look at the candidate's submitted files as well as add comments/notes for their personal reference. Once the user has made their decision, the user can choose to contact or accept or reject the candidate.

Storyboard

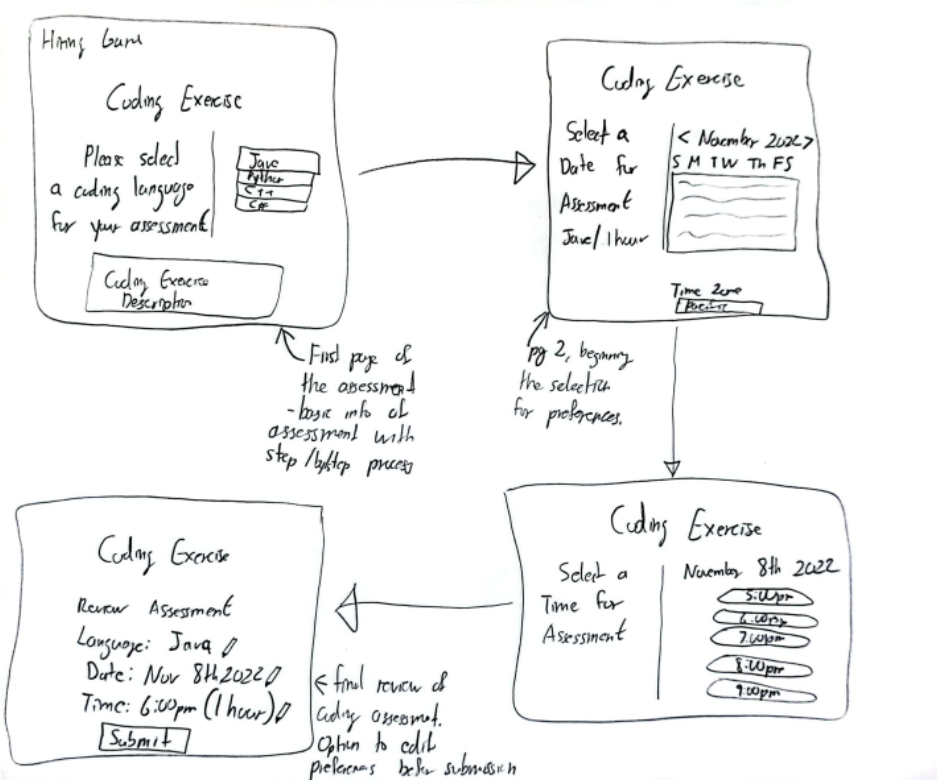


Use Case - Schedule Programming Assessment

Description

When a candidate is in the programming test stage, the system will show an option to schedule a programming test. This will probably include setting a date and time for the test and a test variant (a company usually has multiple variants of programming tests). The system will then use HackerRank API to schedule a test of the candidate (the candidate will receive an email from HackerRank).

Storyboard

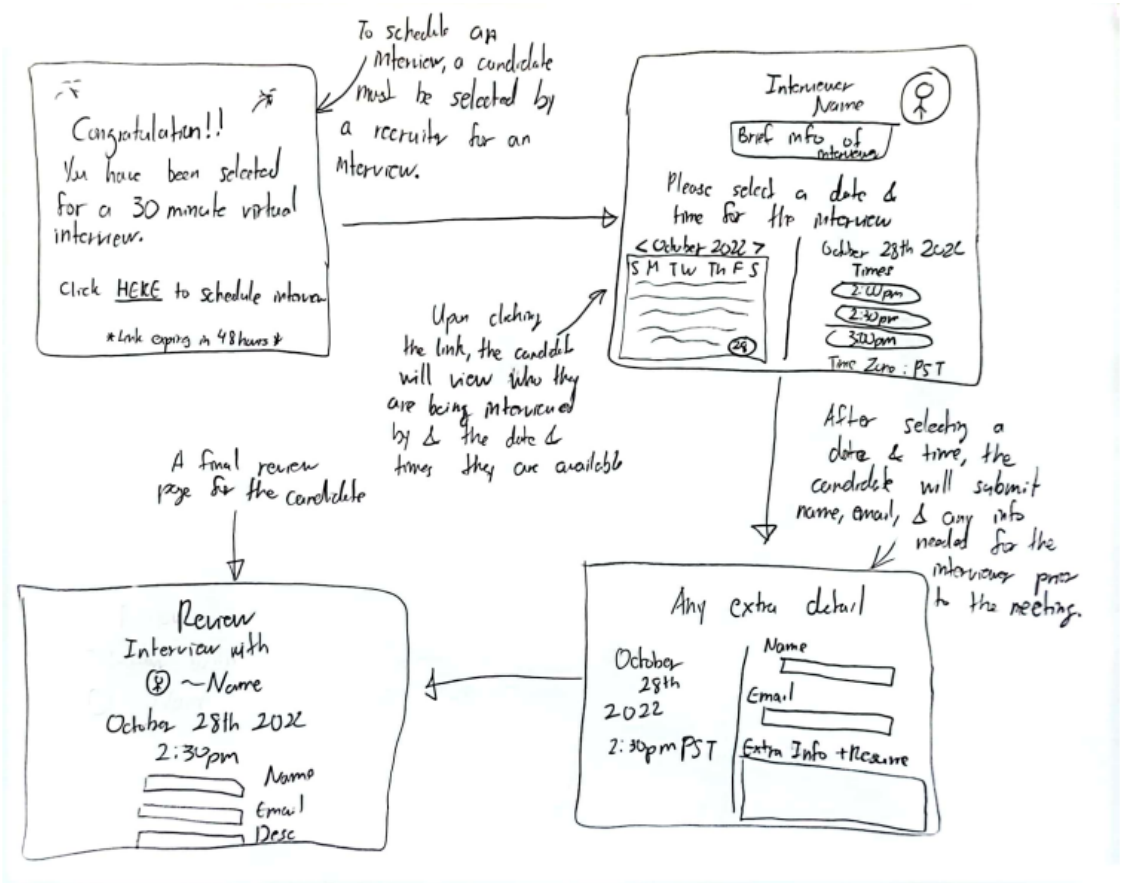


Use Case - Schedule an interview

Description

Once a Candidate is selected for an interview, they will be given a link which will take them to a section of the site where they can view their upcoming interview information. This information will include the individual who will be interviewing them, as well as what their role is within the company. After that, they can select a date and time as well as add any information they want to be known prior to the interview. Upon submission, this will then go back to the Interviewer, who will be notified of the date and time chosen for the interview.

Storyboard

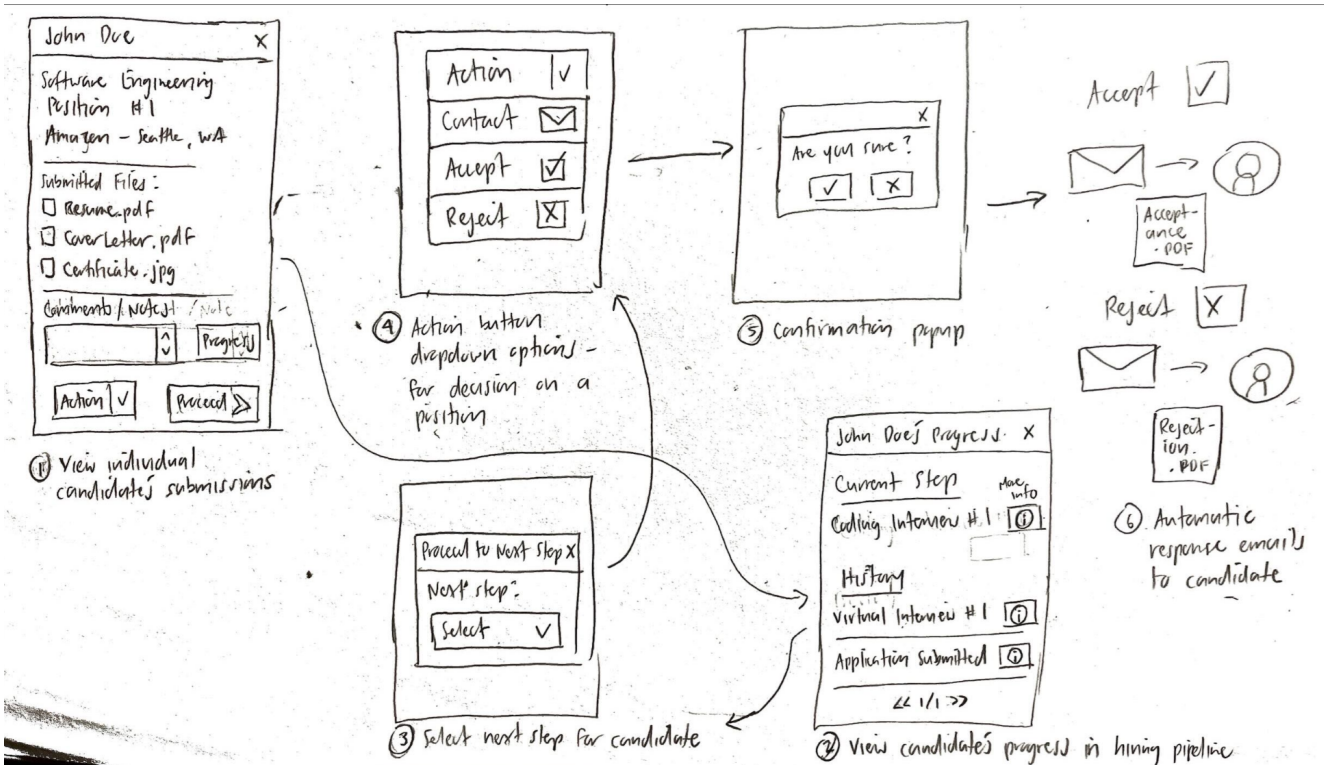


Use Case - Complete Hiring Process

Description

At the last stage of the hiring process, the system shows options to complete the hiring process. Specifically, a button to either reject the application or place a job offer. Rejection will include writing an email, the system will send the email itself. For acceptance, the system will allow uploading a PDF file with a job offer and will email that to the candidate.

Storyboard



4 - High-level database architecture and organization

- Company (Strong Entity)
    - Database Requirements
        - The company shall have at least one employee
        - The company shall be created by one user
    - Entities and Attributes
        - Company_id: key, alphanumeric, multi-value, composite
            - Address
            - Company Size
            - Employees
            - Mission
            - Square Feet
            - Title
            - Description

- Role (Weak Entity)
    - Database Requirements
        - A Job Role shall be assigned to many employees
        - A Job Role shall be created by an Employee
        - A role shall have many positions opened
    - Entities and Attributes
        - Role_id: key, alphanumeric, multi-value, composite
            - Description
            - Title
            - Benefits

- Position (Weak Entity)
    - Database Requirements
        - A Job Position shall be opened for a Role
        - A Job Position shall be created by the authorized Employee
    - Entities and Attributes
        - Position_id: key, alphanumeric, derived

- Candidate(Strong Entity)
    - Database Requirements

        - A Candidate shall be able to view at zero to many job positions
        - A Candidate shall be able to be interviewed by an Employee
        - A Candidate shall be send Applications for multiple job Positions
    - Entities and Attributes
        - Candidate_id: key, alphanumeric, derived
        - Information: alphanumeric, composite
            - First name
            - Last name
            - Date of Birth
            - Email
            - Phone Number
            - Resume

- Employee (Strong Entity)
    - Database Requirements
        - An Employee shall be able to view candidates in the Recruitment Pipelines Process
        - An Employee shall be able to interview Candidates
        - An employee shall be able to schedule interview of candidates
        - An employee shall be able to setup Permissions and Groups
        - An authorized employee shall be able to shortlist Applications
        - An authorized employee setup Recruitment Pipelines Process
        - An authorized employee shall be creating many Job positions
        - An authorized employee shall be able to create many Job roles
    - Entities and Attributes
        - Employee_id: key, alphanumeric, composite
            - Name
            - Role
            - Position
        - Employee number: numeric, derived

- JobApplication (Strong Entity)
    - Database Requirements
        - A job application shall be opened by a candidate for a specific Position
        - A job application shall contain profile information of many candidates
    - Entities and Attributes
        - JobApp_id: key, alphanumeric, derived

- Interview (Weak Entity)
    - Database Requirements
        - An Interview shall be able conducted by at least one Employees
        - An Interview shall have at least one Candidate
    - Entities and Attributes
        - Interview_id: key, alphanumeric, composite
            - Time
            - Date
            - Interviewer
            - Interviewee
            - Location

- Group (Weak Entity)
    - Database Requirements
        - A group can have many employees
        - A group can be assigned multiple permissions
    - Entities and Attributes
        - group_id: key, alphanumeric, derived

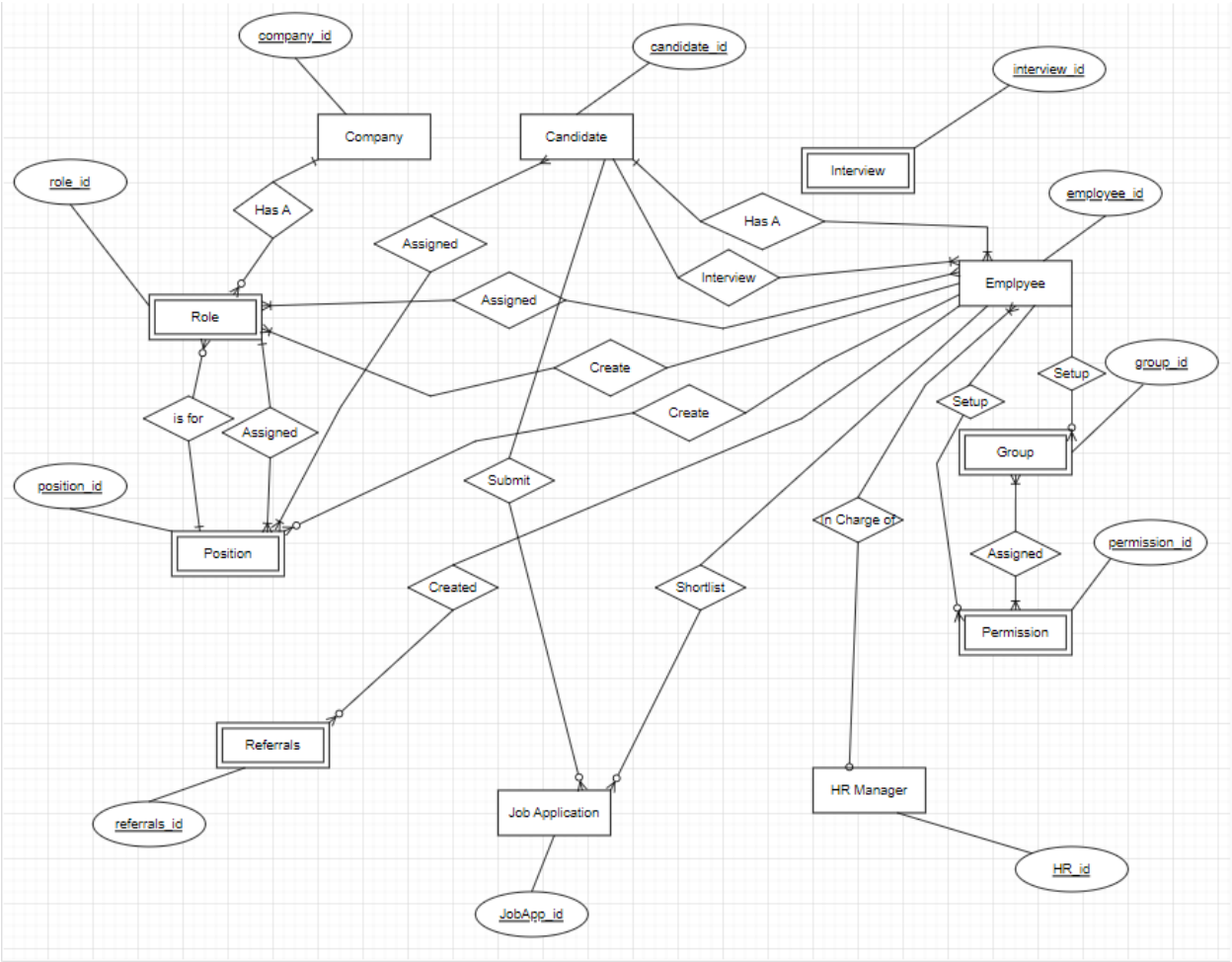
- Permission (Weak Entity)
    - Database Requirements
        - A Permission can be assigned to multiple Groups
    - Entities and Attributes
        - Permission_id: key, alphanumeric, derived

- Referrals (Weak Entity)
    - Database Requirements
        - A Referral shall be created by at least Employees
        - A Referral shall be used for one Candidates
    - Entities and Attributes
        - Referral_id: key, alphanumeric, derived

- HR Manager (Strong Entity)
    - Database Requirements
        - A HR Manager shall be creating many Job positions

        - A HR Manager shall be creating many Job roles
        - A HR Manager shall be one Authorized user
        - A HR Manager shall be in charge of many employees
    - Entities and Attributes
        - HR_id: key, alphanumeric, Composite
            - Role
            - Position
            - Title

Entity Relationship Diagram (ERD)



5 - High Level APIs and Main Algorithms

APIs

The following are the REST resources and their associated APIs:

Company

1. Create a new Company
    - This API creates a new company with the requested parameters : title and description. Both title and description are requested in the Company constructor, and once provided are able to create a new Company.
2. Get the detail of a Company
    - This API gets a certain company once a company id is provided. If a company that matches the id exists, a company gets returned.
3. Get a list of all Company
    - This API gets all companies that exist within the database. Within the API response, the company title, company description, and company ID is returned.
4. Update the details of a Company
    - This API updates an existing company. For this API, an existing company id is requested as a parameter as well as the new company description and title. After this API is sent, the database updates the details of the company of the provided id.
5. Delete a Company
    - This API deletes a company from the database. For this API, a company id is needed to delete the company.

Role

1. Create a new Role
    - This API creates a new role. For the path of this API, a company id is requested in the path, and title,expectations, and benefits are requested as parameters. Title, expectations, benefits, and a Company variable are requested in the Role constructor, and once all are provided a new Role is created.
2. Get the detail of a Role
    - This API gets a certain role once a company id is provided.For the path of this API, a company id is requested in the path. If an existing company id and a corresponding role match the path variable, then a role gets returned, which includes the corresponding company information.
3. Get a list of all Role

    - This API gets all roles that exist within the database. Within the API response, the role title, role expectations,role benefits and role ID is returned.
4. Update the details of a Role
    - This API updates an existing role. For this API, a company id is requested in the path. Additionally, a new role title, expectations, and benefits is requested as a parameter. After the API is run, the role details are updated on the corresponding company id.
5. Delete a Role
    - This API deletes a role from the database. For this API, a company id is needed to delete the role and company details.

Position

1. Create a new Position
    - For this API, the title and experience necessary for the position are requested as parameters. The title and experience necessary are requested in the Referral constructor and once both are provided a new Position is created.
2. Get the detail of a Position
    - For this API, a position id is required in the path of this API. If a position with that id exists, then the position is returned.
3. Get a list of all Position
    - This API gets all of the positions that exist within the database. The response returns the title and experience necessary for the position.
4. Update the details of a Position
    - This API updates the details of existing positions. For this API, a position id is requested in the path. Additionally, the title and experience necessary are requested as parameters.
5. Delete a Position
    - This API deletes a position that exists within the database. For this API, the position id is required in the path of the API.
6. Search all open positions using a query string
    - This API requires a parameter that provides a keyword that the query string parses through and finds positions that match the keyword. The API returns all positions that match the keyword.

Referrals

1. Create a new Referral
    - For this API, the first name of the candidate, the last name of the candidate, the email of the candidate, and the resume are requested as parameters. All of the requested parameters are requested in the Referral constructor.
2. Get the detail of a Referral

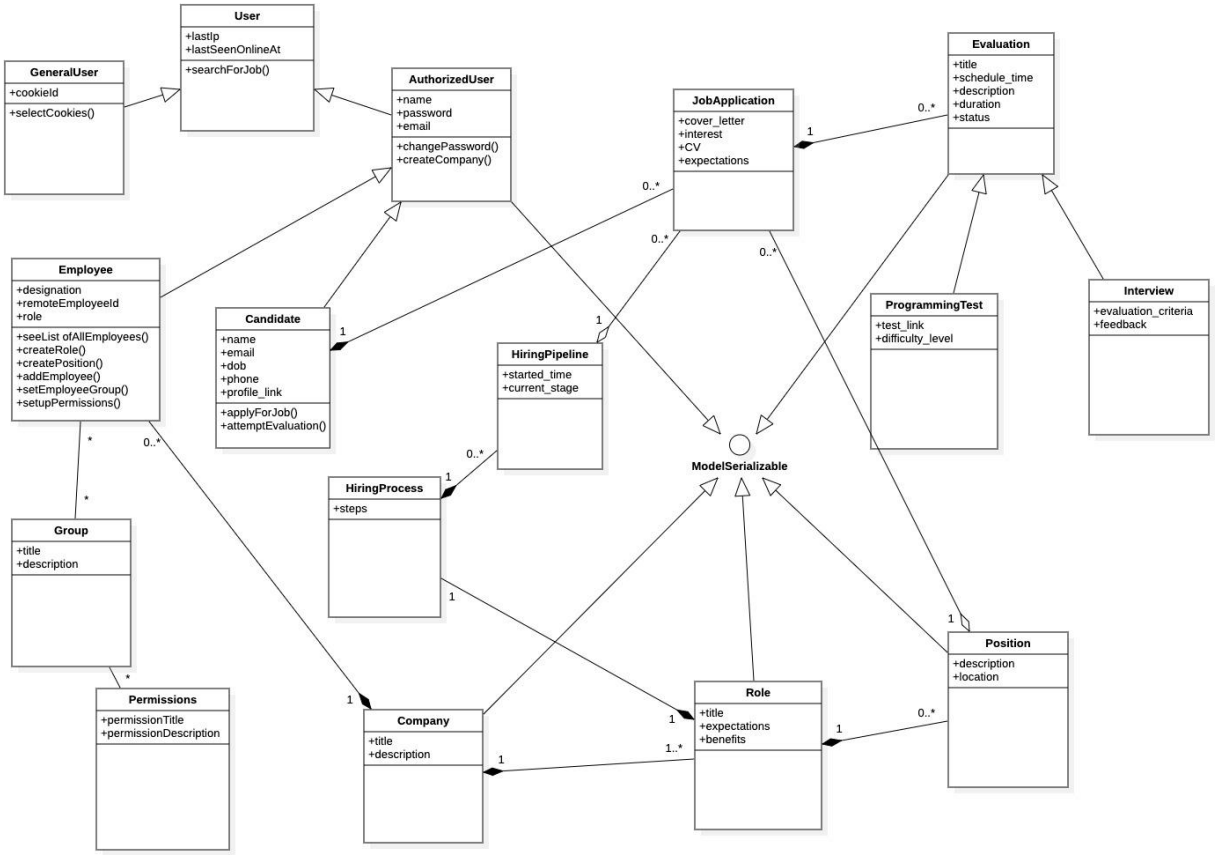
    - For this API, a referral id is required in the path of this API. If a referral with that id exists, then the referral is returned.
3. Get a list of all Referral
    - This API gets all of the referrals that exist within the database. The response returns the first name of the candidate, the last name of the candidate, the email of the candidate, and the resume file name.
4. Update the details of a Referral
    - This API updates the details of existing referral. For this API, a referral id is requested in the path. Additionally, the first name of the candidate, the last name of the candidate, the email of the candidate, and the resume are requested as parameters.
5. Delete a Referral
    - This API deletes a referral from the database. The API takes the referral id as the path variable.

Job Application

1. Create a new Application
    - This API creates a new job. For the path of this API, a role id is requested in the path, and title,location, description, and job type are requested as parameters. Title,location, description, job type, and a Role variable are requested in the Job constructor, and once all are provided a new Job is created.
2. Get the detail of an Application
    - This API gets a certain job once a role id is provided.For the path of this API, a role id is requested in the path. If an existing role id and a corresponding job match the path variable, then a job gets returned, which includes the corresponding role and company information.
3. Get a list of all Applications
    - This API gets all jobs that exist within the database. Within the API response, the job title, job location, job description, job type and job ID is returned. Additionally, the corresponding role and company information is returned as well.
4. Update the details of an Application
    - This API updates an existing job. For this API, a role id is requested in the path. Additionally, a new title,location, description, and job type is requested as a parameter. After the API is run, the job details are updated on the corresponding role id.
5. Delete an Application
    - This API deletes a job from the database. For this API, a role id is needed to delete the job, role, and company details.

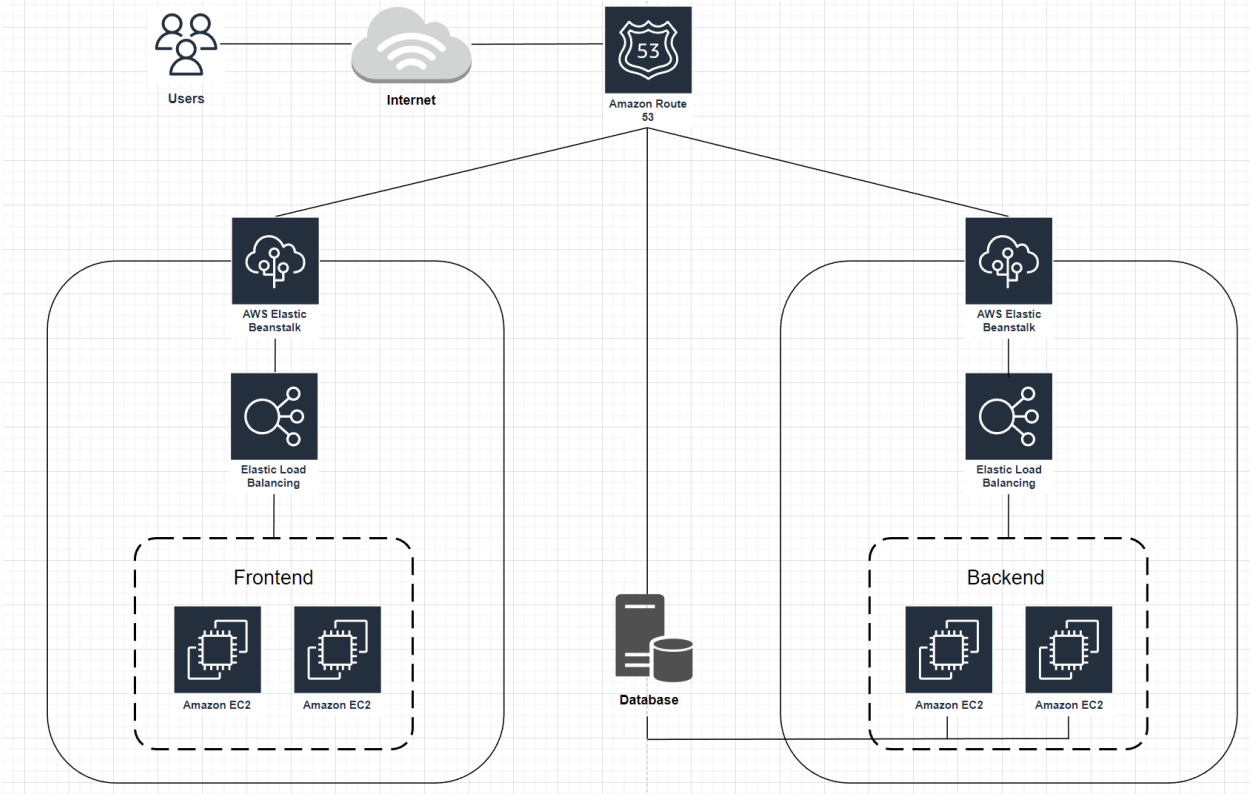
6 - High-Level UML Diagrams

System class diagram

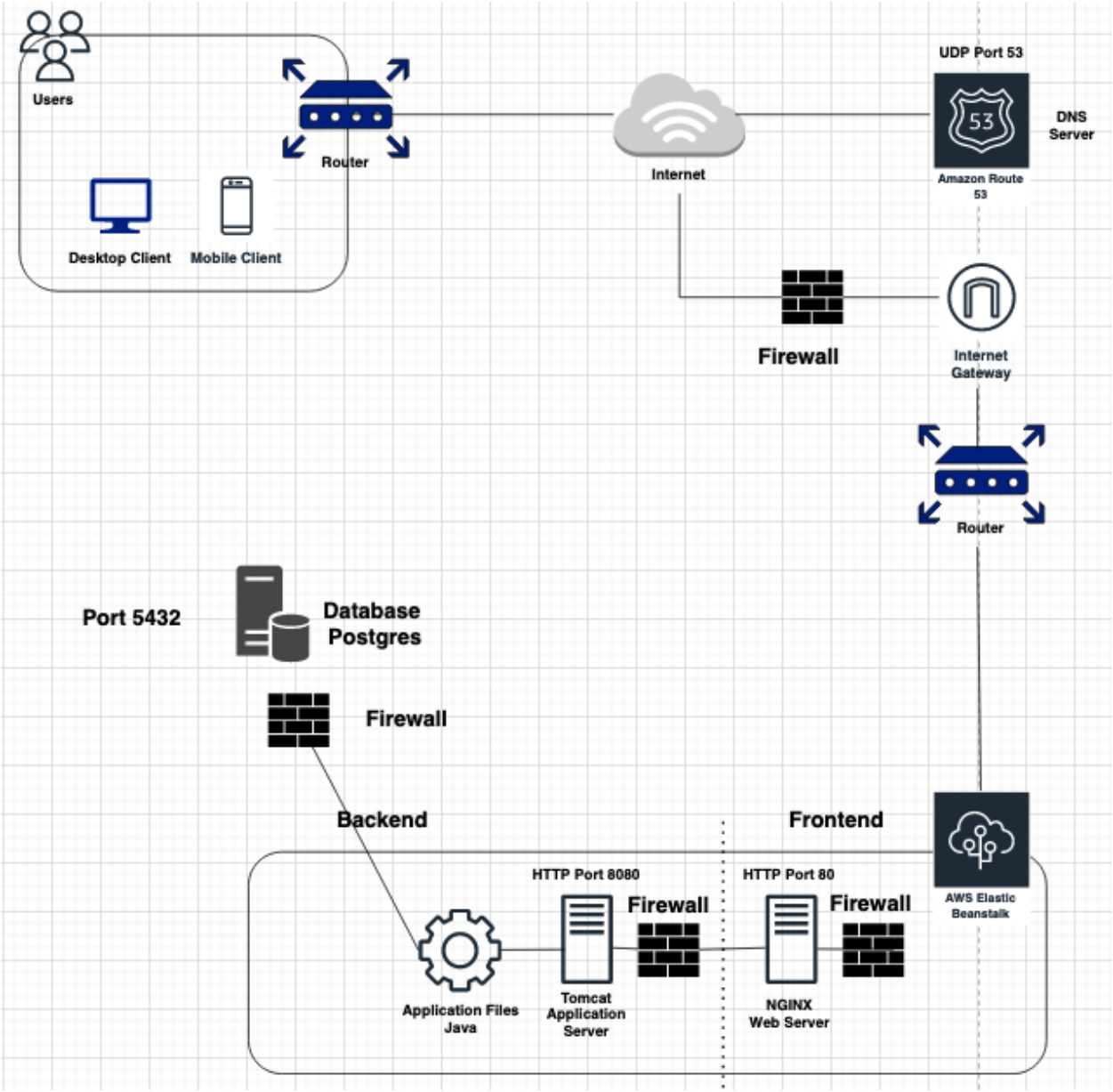


7 - High-Level Application Network and Deployment Diagrams

AWS Elastic Beanstalk Deployment Diagram



Network Diagram



8 - Identify actual key risks for your project at this time

1. **Course Conflicts**
    - Risk: Various high-course assignments and conflicts interfere with the time needed for this project. Many of us are taking high-level courses in addition to this class, such as Intro to Database, UNIX Programming, and even some Grad

School work, to name a few. Every class requires a strong level of commitment and assigns rigorous work to be completed in a timely manner. All this work can become overbearing at times and conflict with the work needed to be done within the Software Engineering Course.

- Solution: Our solution for this project risk is to further improve our time management skills. In the real world, many different aspects of life are going to come into play when it comes to Software Engineering, and this course is a strong example of a real-life scenario. We have to navigate the different course workloads ourselves and make sure we get all the necessary requirements of the milestone done while not compromising other course work

2. Scheduling Conflicts

- Risk: All team members have a number of other commitments and different schedules (commute time, off and on-campus jobs, other class times), which all conflict and make having everyone present in the meeting rather difficult. Our main meeting time is 7:30 PM every Mon and Wed, and even then, that is right when some team members finish meetings, finish class time, and right when they get home. It gives us very little time to decompress and relax at all, which can greatly show within the milestone quality.
- Solution: To solve our scheduling conflicts, we did our very best to find a general time that works for everyone. In addition to that, we also made sure we established a strong base of communication with one another. The reason for this is that when certain things come up, that may conflict with a team member's ability to attend the meeting, we make sure to have detailed meeting notes for them to understand exactly what they missed so we are not wasting any precious time catching them up.

3. Knowledge Gap

- Risk: All the team members have different experiences with software engineering as a whole. Some came from community colleges, some had industry experience, and some only have SFSU experience. It is difficult to find common ground in the technologies and project where everyone feels comfortable in what they are working on
- Solution: The way we have decided to deal with this problem is by making sure all group members have a strong sense of patience and understanding for one another. We all must acknowledge that everyone has different academic experiences, and some may be more beneficial in this project than others. That is why we make sure to do our own side homework to ensure all team members understand the technologies we will be using and for those who already understand the coding language to help teach others, so we are all skilled enough to complete the project.

4. Lack of Team Member

- Risk: At the beginning of the semester, we had a full team of 6 members ready to tackle this entire project. Unfortunately, one of our team members at the time decided to move groups to aid a smaller team in the project. While we hold no resentment towards him and fully understand the reason for their team switch, it now puts the load of the project onto only five of us. This makes the workload rather intense for all of us to complete, so we have to work extra hard to compensate for that.
- Solution: Due to an uneven classroom size, some teams have a small number of individuals compared to others. We once had a full team ready to tackle this project. However, they decided to move to another group needing a member because their skills and experiences better fit their project. This now leaves us with 5 members rather than 6. This put an extra set of pressure on us, but we did our very best to work as a team to do the work. We understand that even though we are down a member, if we put in that extra effort, then we will be able to compensate for the missing team member and get this project done.

9 - Project management

Trello is the main tool we are currently using to keep a unified progression on the project. In addition, we agreed to meet twice a week, on Wednesdays and Mondays, in order to go over what is expected of each team member and update each other on what is going on. We try to

keep the meetings short, around twenty minutes, as all of our schedules are pretty busy. We also use the Trello board for retrospection every week.

Posting our daily status in discord is a requirement that has been made by the team lead. This means that each team member must post every day what they have done, what they will do the next day and any questions or concerns about the project. This really keeps us up to date with each other and helps ensure that we stay connected as a team.

Another important part of our team process is evaluating how we are doing every few weeks. We have anonymous concern meetings where we anonymously post things that we think we are doing well, any things that we could do better at. We then go over them all and discuss how we can improve.

10 - Detailed list of contributions

- Contributions:
    - Kenny Leong

        - Worked on the Landing page - frontend. Completed NavBar, CustomerReview, Tagline, and Footer components
        - Did 3 mockups/storyboards
        - Worked on Application and Network architecture diagrams
        - Helped plan the frontend work
    - Mamadou Bah
        - Worked on the Landing page - frontend. Completed ProductFeature component
        - Did 3 mockups/storyboards
        - Worked on “Identify actual key risks …” section
        - Worked on the high-level database architecture section
        - Formatting and improving the document
    - Khushi Khanna
        - Did 3 mockups/storyboards
        - Worked on Data Definitions section
        - Worked on High-Level APIs section
        - Worked on backend APIs
        - Helped to prioritize the functional requirements
    - Eric Leow
        - Did 3 mockups/storyboards
        - Tried to setup the ORM
        - Worked on the Project management section
        - Helped to prioritize the functional requirements
    - Farhan Haider
        - Did 3 mockups/storyboards
        - Worked on the UML diagram
        - Prioritized functional requirements
        - Completed the JobSearch component on the frontend
        - Planned the overall work
        - Helped other team members with their work
    - Score (Out of 10 - 10 being ideal)
        - Kenny Leong - 9
        - Mamadou Bah - 8

- Khushi Khanna - 8
- Eric Leow - 6