SW Engineering CSC648/848 Spring 2021

648-02 | Team 02 | ProSpector

Cameron Cirini (Team Lead) ccirini@mail.sfsu.edu,
Franklin Arevalo (Frontend Lead), Faisa Jama (GitHub Master),
Tony Cao (Backend Lead), Zhuojun He (Backend Support)

“Milestone 2”

03/09/21

Date Submitted	Date Revised
03/09/21	N/A

1. Functional Requirements - prioritized:

Req ID	Priority	Function	Description
F1	**1**	**Students** shall be able to **sign in**.	If users of our application have a registered account with us, they shall be allowed to sign in to their profile and access our services.
F2	**1**	**Students** shall be able to **sign up**.	If users of our application do not have a registered account with us, they shall be able to register themselves by providing necessary information. After, they shall be about to sign in and access our services.
F4	**1**	**Students** can **update** their profile information.	If users who are looking for a job use our application and they have a registered account with us, they shall be able to update their profile information at any time without restrictions.
F6	**1**	**Students** shall **receive alerts** to get ready for interviews.	If users who are looking for a job use our application and they have a registered account, they shall be able to receive alerts when they are matched with a potential employer to get ready for their interviews.
F7	**1**	**Professors** shall have the ability to **rate students** in a scale from 1-5 fashion (being 5 being the highest ratings that implies knowledgeable, responsible, teamwork, leadership, committed to success, etc).	College professors of registered candidates shall have the ability to rate their students for knowledge, responsibility, teamwork, leadership, committed to success, etc).

F8	**1**	**Professors** shall have the ability to **enter recommendations**.	College professors of registered candidates shall have the ability to write any recommendations for candidates.
F9	**1**	Tech **Companies** shall have the ability to **register** to the portal.	Companies using our services to search for potential hires shall be able to register themselves onto your application.
F10	**1**	Tech **Companies** shall have the ability to filter candidates with **advanced search**.	Registered companies of our application shall have the ability to search for candidates with the option for advanced search options based on major and demographics.
F12	**1**	Tech **Companies** shall have the ability to **view applications**.	Registered companies of our application shall have the ability to view candidate applications.
F13	**1**	Tech **Companies** shall have the ability to **download resumes**.	Registered companies of our application shall have the ability to download candidate resumes.
F14	**1**	Tech **Companies** shall **receive alerts** for any and all profiles that match their job listings.	Registered companies of our application shall be alerted if and when a match has been made by our algorithm.
F15	**1**	Tech **Companies** can get more than one alert for the job listed.	Registered companies of our application shall get more than one alert for a job listed if there are multiple matches.

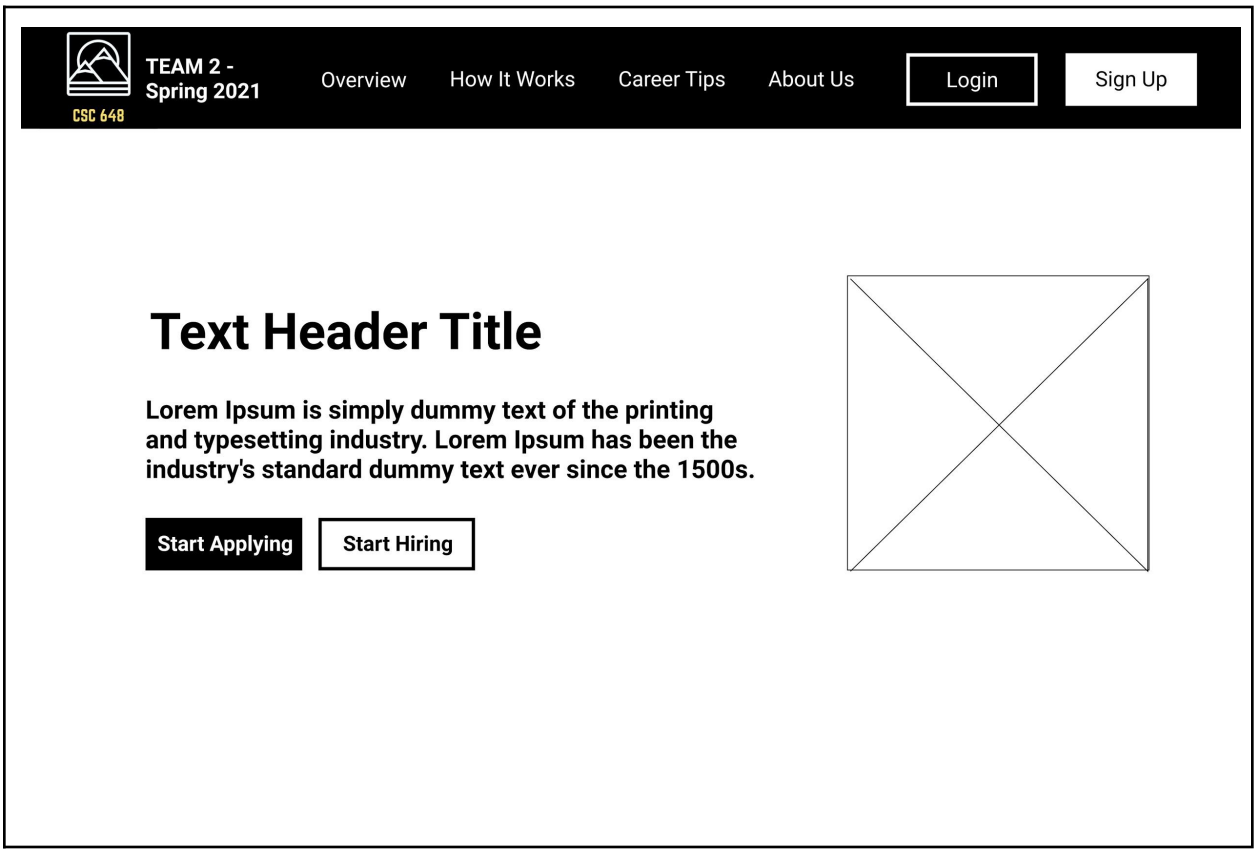
F20	**1**	All **Users** shall have the ability to **delete their profile**.	All registered users of our application shall have the ability to delete their profile and close their account at any time.
F11	**1**	**Advanced search** can be done on major and demographics.	Registered companies of our application using our advanced searches shall be able to search based on major and demographics.
F17	**1**	**Admin** shall have the ability to **view all profiles**.	Admins shall have the ability to view all profiles registered on our application from candidates to professors to companies.
F18	**1**	**Admin** shall have the ability to **view all job listings**.	Admins shall have the ability to view all job postings made by registered companies.
NF1	**1**	**Application** shall be developed.	Application shall be developed, tested and deployed using tools and servers approved by Class CTO and as agreed in M0
NF2	**1**	**Application** shall work on desktop/laptop.	Application shall be optimized for standard desktop/laptop browsers e.g. must render correctly on the two latest versions of two major browsers
NF4	**1**	**Data** shall be stored in database	Data shall be stored in the team's chosen database technology on the team's deployment server.
NF6	**1**	**Application** shall protect the privacy of the users	Privacy of users shall be protected, and all privacy policies will be appropriately communicated to the users.
NF7	**1**	**Application** shall be displayed in english	The language used shall be English.
NF8	**1**	**Application** shall be used by anyone	Application shall be very easy to use and intuitive.

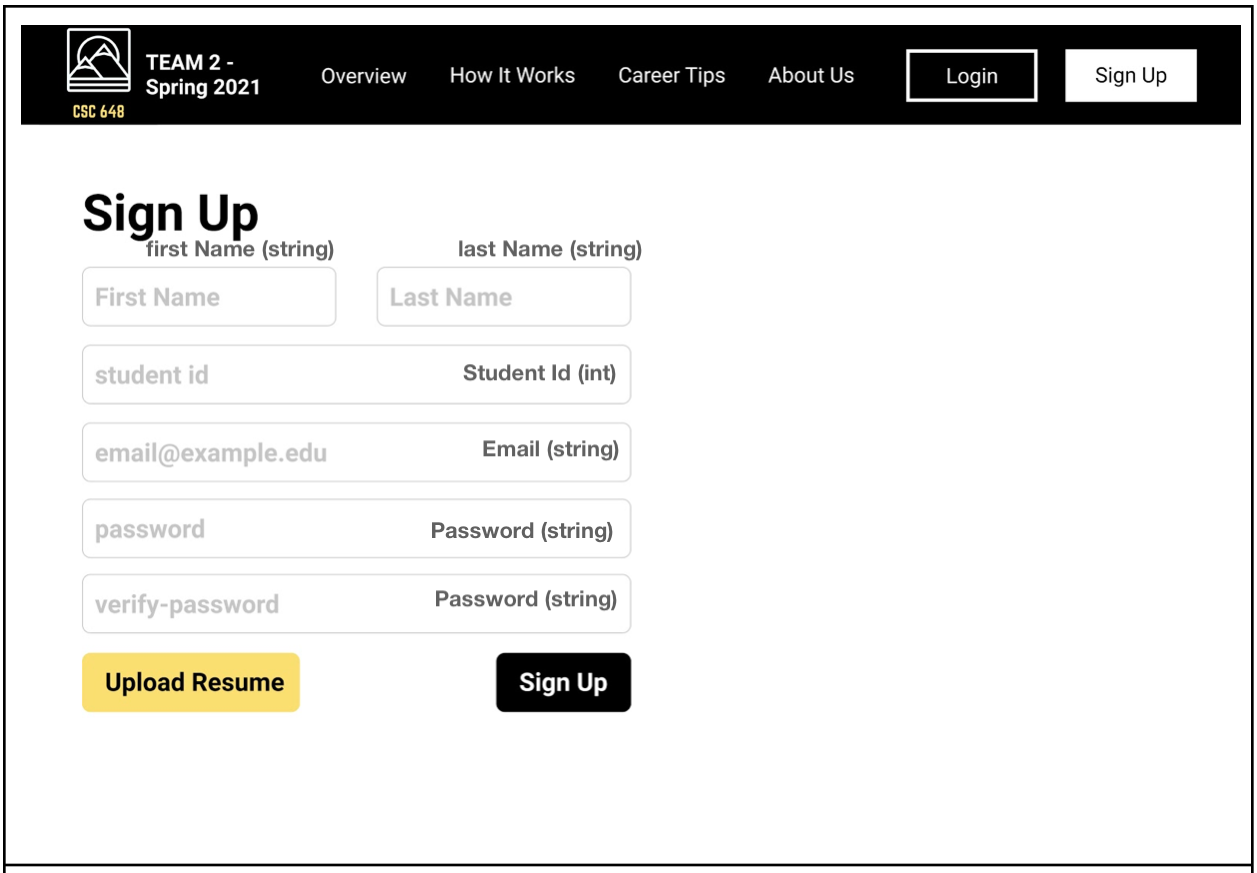
NF12	**1**	**Site** shall be secured.	Site security: basic best practices shall be applied (as covered in the class)
NF14	**1**	**Site** shall display “SFSU Software Engineering Project CSC 648-848, Spring 2021”.	The website shall prominently display the following exact text on all pages "SFSU Software Engineering Project CSC 648-848, Spring 2021. For Demonstration Only” at the top of the WWW page. (Important so not to confuse this with a real application)
F5	**2**	**Students** can **retrieve their password** if forgotten.	If users who are looking for a job use our application and they have a registered account with us and they have forgotten their account password, they shall be able to request to either change their password or retrieve their old password at any time without restrictions.
F16	**2**	**Admin** shall have the ability to approve **company registration**.	Companies who register to use our services will need to be approved by our admins.
F19	**2**	A **Verification email** shall be sent to the user whenever he/she registers for the first time.	Users of our application registering for the first time shall get a verification email sent to them.
NF3	**2**	**Application** shall work on mobile devices	Selected application functions must render well on mobile devices
NF5	**2**	**Application** shall be accessed by 100 users at a time	No more than 100 concurrent users shall be accessing the application at any time

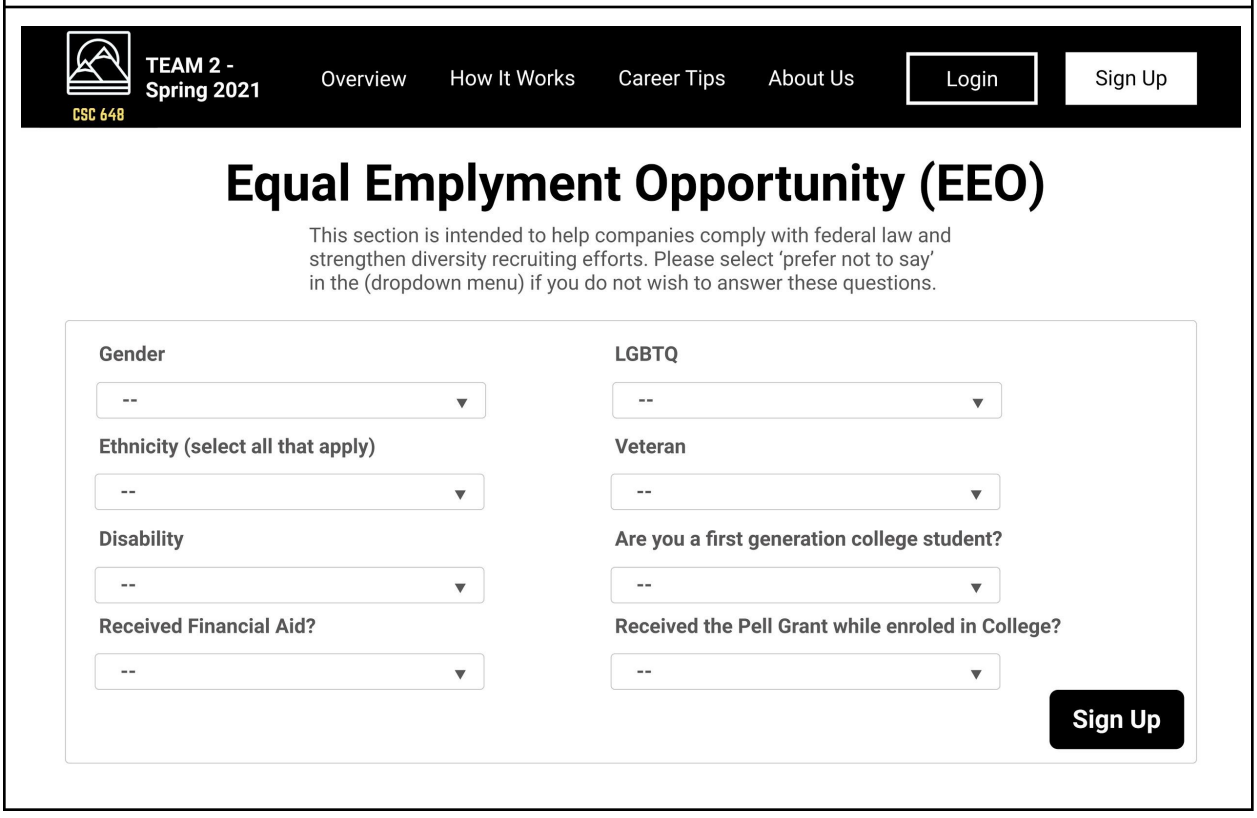
NF9	**2**	**Application** shall use Google Maps	Google maps and analytics shall be added
NF10	**2**	**Application** shall use webmail	No email clients shall be allowed. You shall use webmail.
NF11	**2**	All Functions in the **application** shall be free for all users.	Pay functionality, if any (e.g. paying for goods and services) shall not be implemented nor simulated in UI.
NF13	**2**	**Application** shall be made by the modern SE process.	Modern SE processes and practices shall be used as specified in the class, including collaborative and continuous SW development
F3	**3**	**Students** shall be able to **upload resumes and videos** and enter any demographics, experience and education for their profile.	If users who are looking for a job use our application and they have a registered account with us, they shall be able to upload their resumes, personalized videos, and/or enter any demographics, experiences and education to their profile.

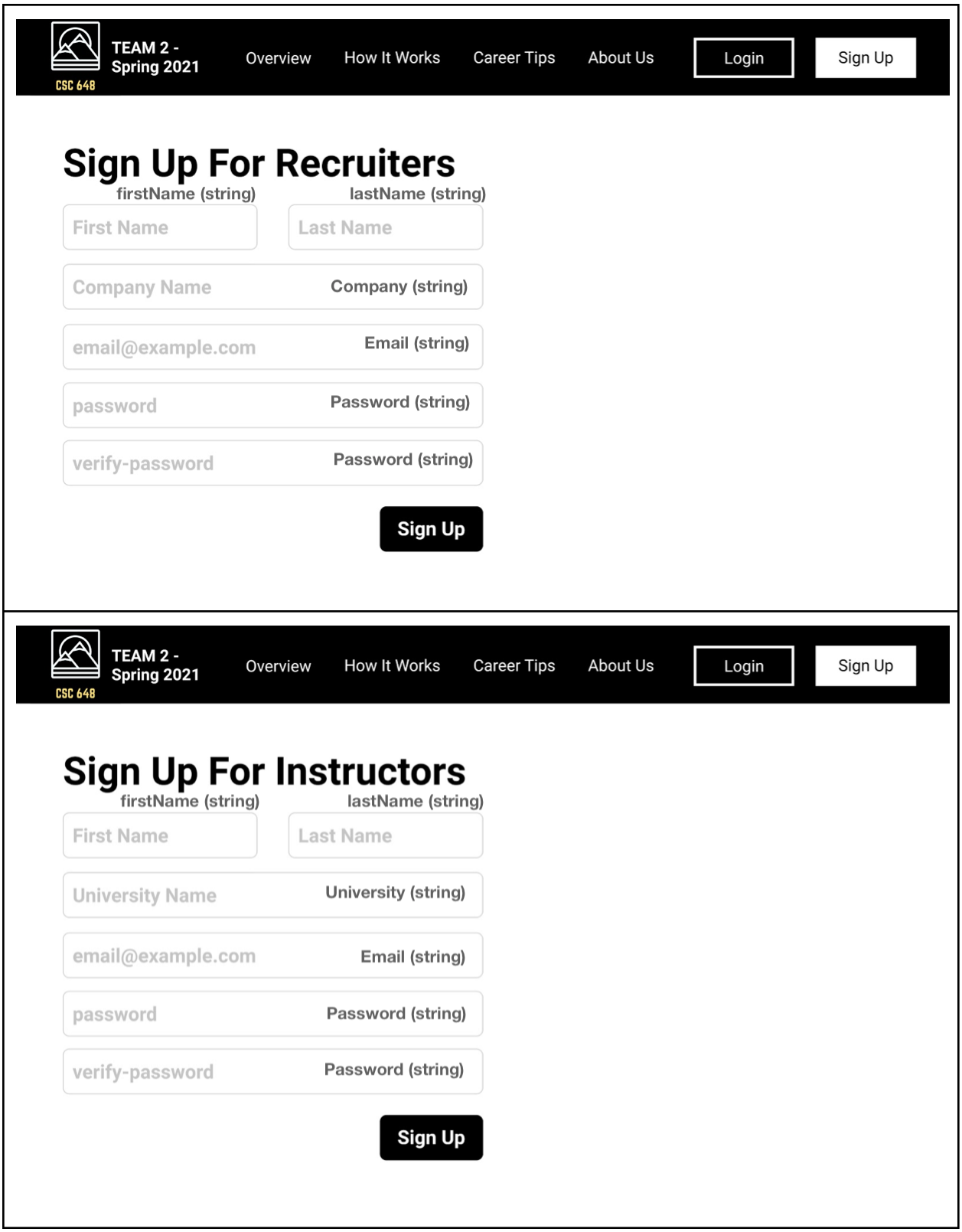
2. UI Mockups and Storyboards (high level only)

1. Once on the Start page, the users shall click the sign up button. On the sign up page, the user shall provide their email, desired password, and any other needed information based on whether they are a student, recruiter, or instructor. Once completed, they have successfully signed up.

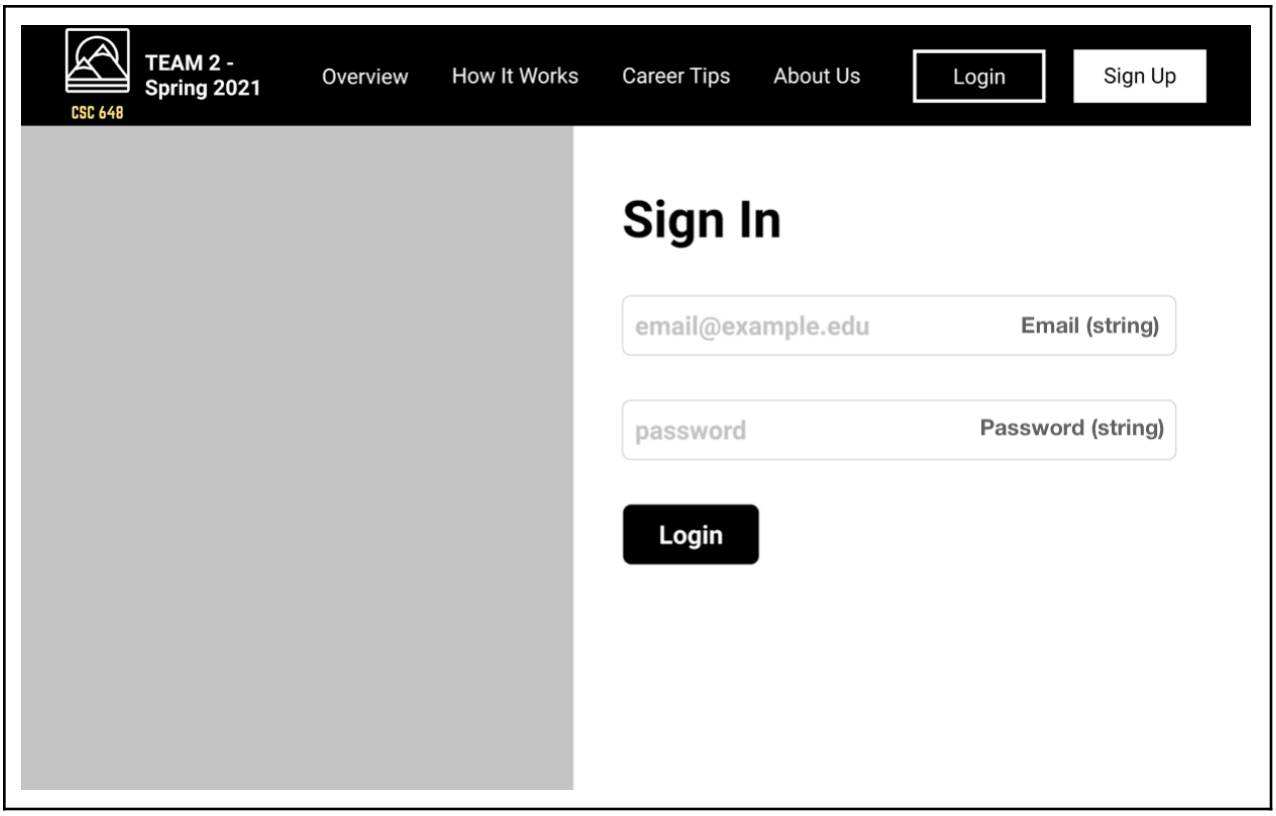


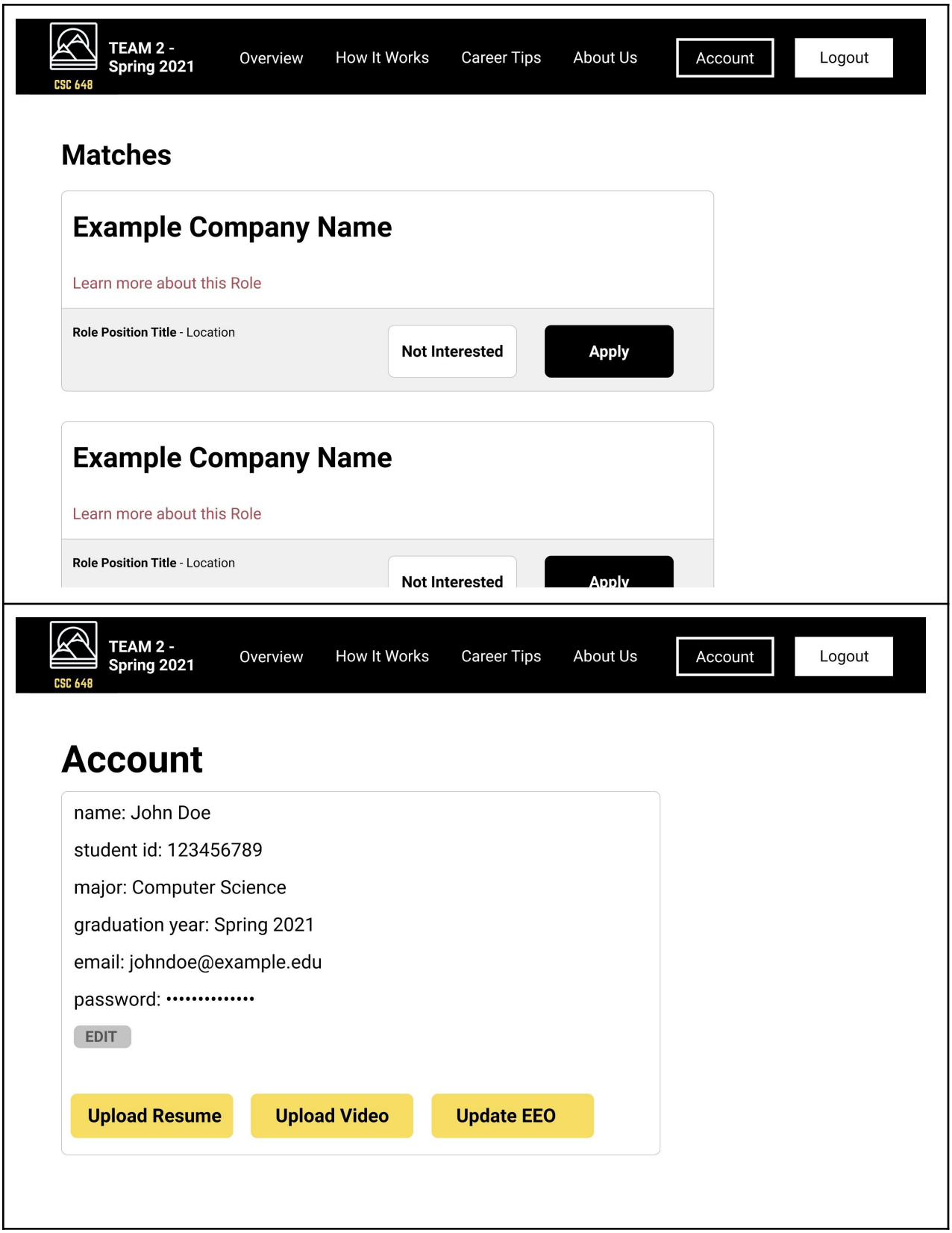

TEAM 2 -
Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Login
Sign Up
Sign Up
first Name (string)
last Name (string)
First Name
Last Name
student id
Student Id (int)
email@example.edu
Email (string)
password
Password (string)
verify-password
Password (string)
Upload Resume
Sign Up


TEAM 2 -
Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Login
Sign Up
Equal Emplyment Opportunity (EEO)
This section is intended to help companies comply with federal law and strengthen diversity recruiting efforts. Please select 'prefer not to say' in the (dropdown menu) if you do not wish to answer these questions.
Gender
--
LGBTQ
--
Ethnicity (select all that apply)
--
Veteran
--
Disability
--
Are you a first generation college student?
--
Received Financial Aid?
--
Received the Pell Grant while enroled in College?
--
Sign Up

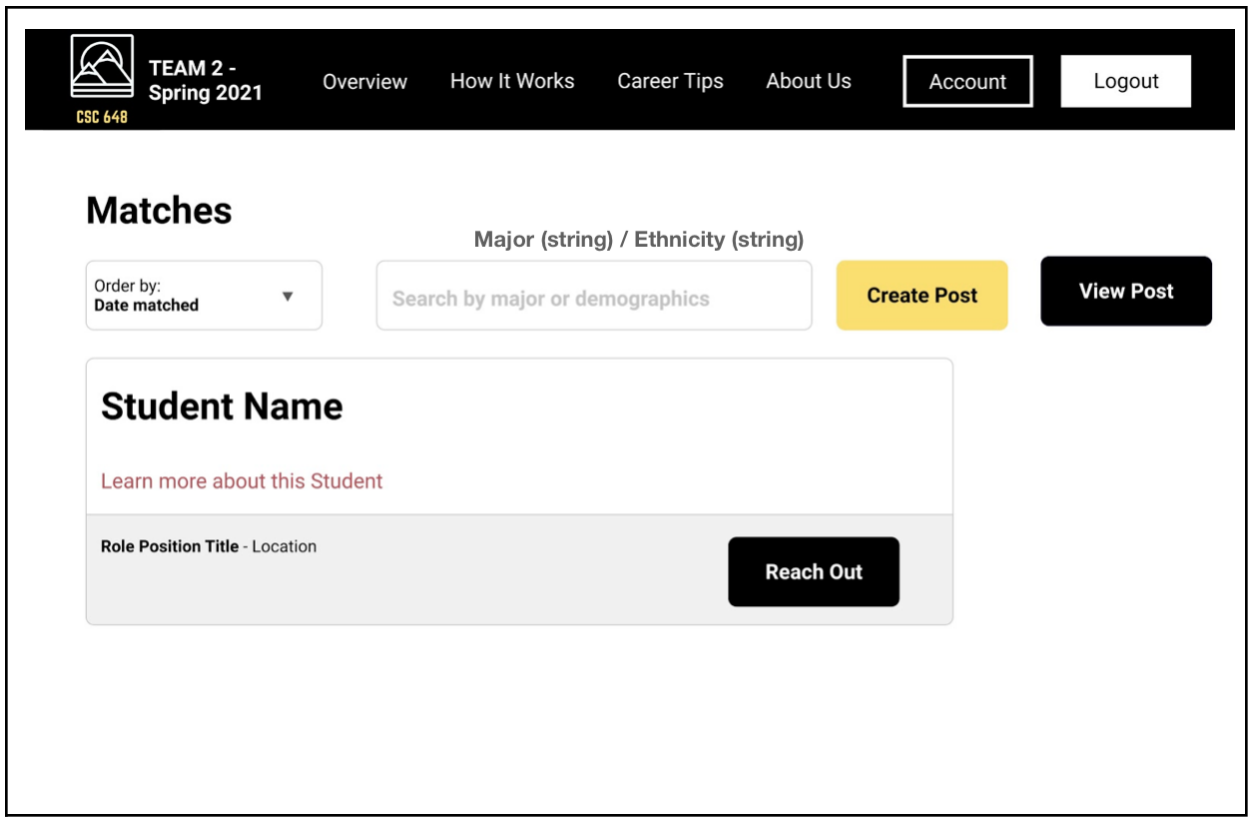

TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Login
Sign Up
Sign Up For Recruiters
firstName (string)
lastName (string)
First Name
Last Name
Company Name
Company (string)
email@example.com
Email (string)
password
Password (string)
verify-password
Password (string)
Sign Up
TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Login
Sign Up
Sign Up For Instructors
firstName (string)
lastName (string)
First Name
Last Name
University Name
University (string)
email@example.com
Email (string)
password
Password (string)
verify-password
Password (string)
Sign Up

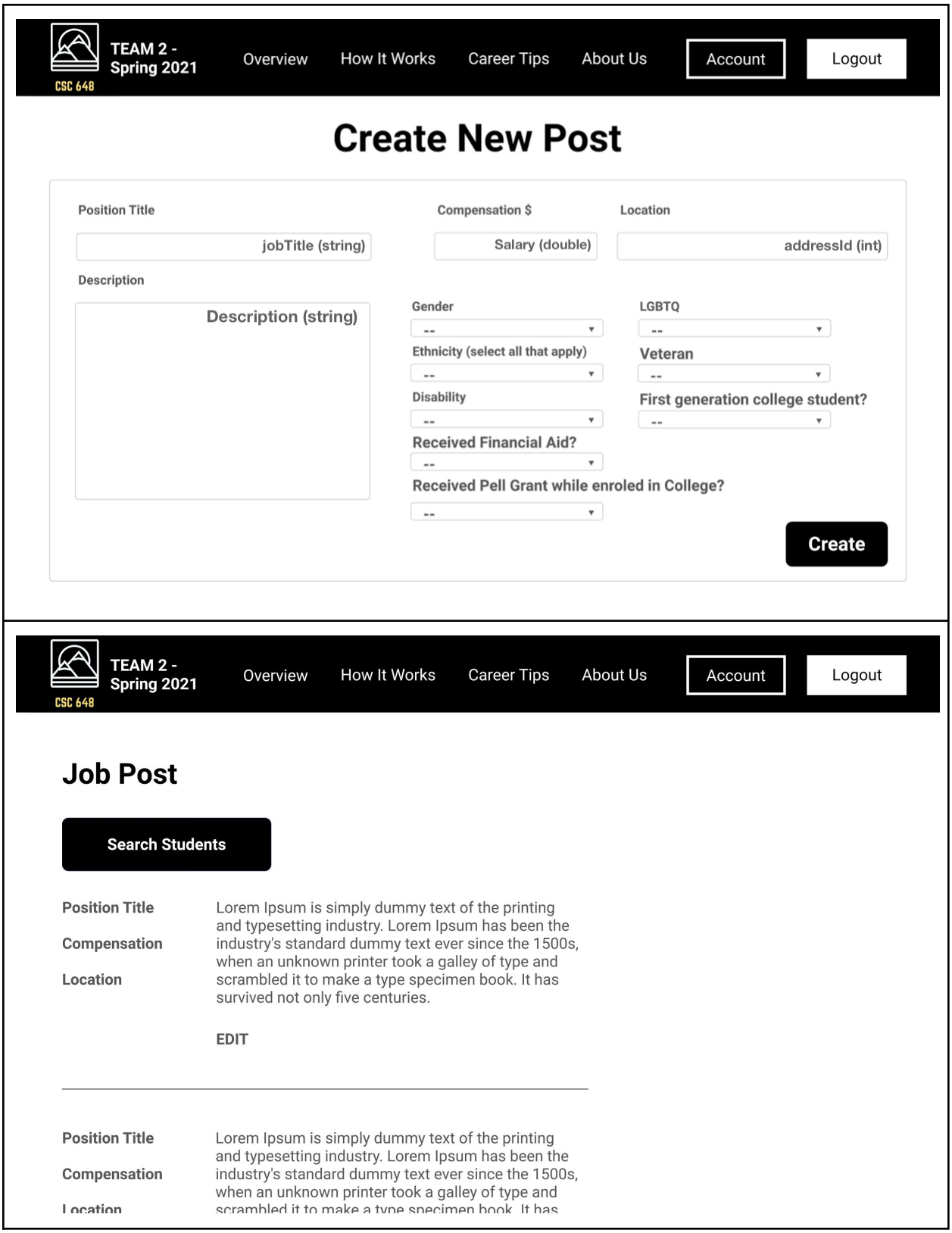
2. Once the student clicks the login button, the student shall sign in. The student shall be directed to their home page. The student shall press the account button to be directed to their profile, and have the choice to upload resumes and videos.



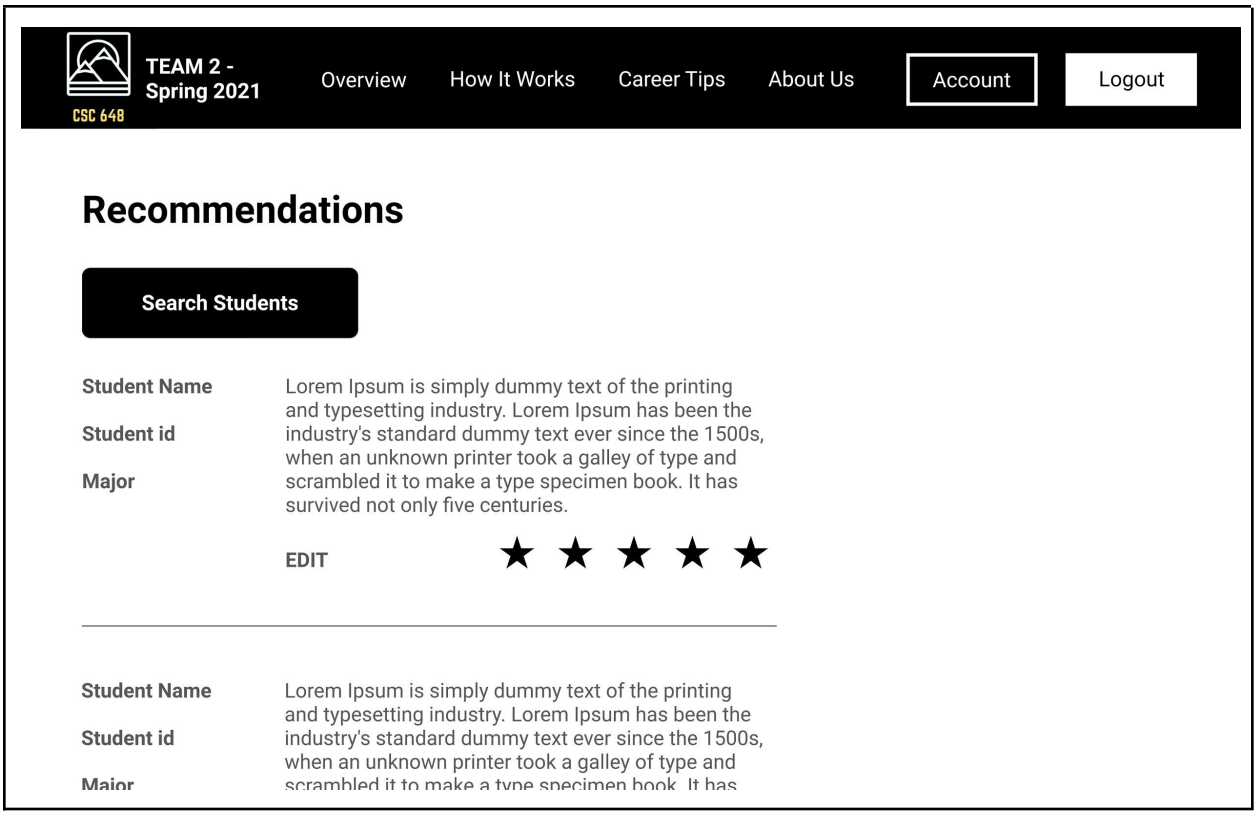

TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Matches
Example Company Name
Learn more about this Role
Role Position Title - Location
Not Interested
Apply
Example Company Name
Learn more about this Role
Role Position Title - Location
Not Interested
Apply
TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Account
name: John Doe
student id: 123456789
major: Computer Science
graduation year: Spring 2021
email: johndoe@example.edu
password: ••••••••••••••
EDIT
Upload Resume
Upload Video
Update EEO

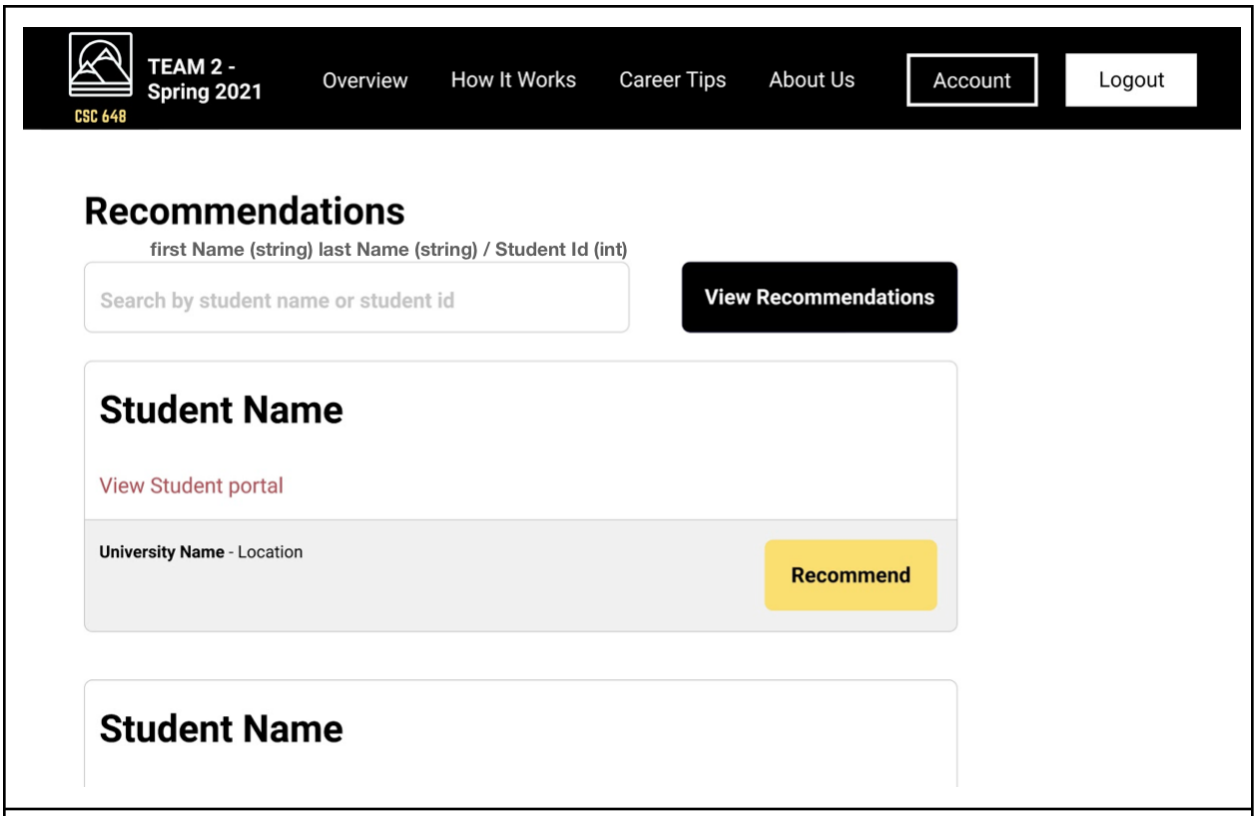
3. Once the recruiter is signed in, the recruiter shall click the button to create a new job post. The recruiter shall add information about the job, and their qualification in demographics, experience and education. Once the recruiter presses the create button, they shall be directed to the list of their job post.

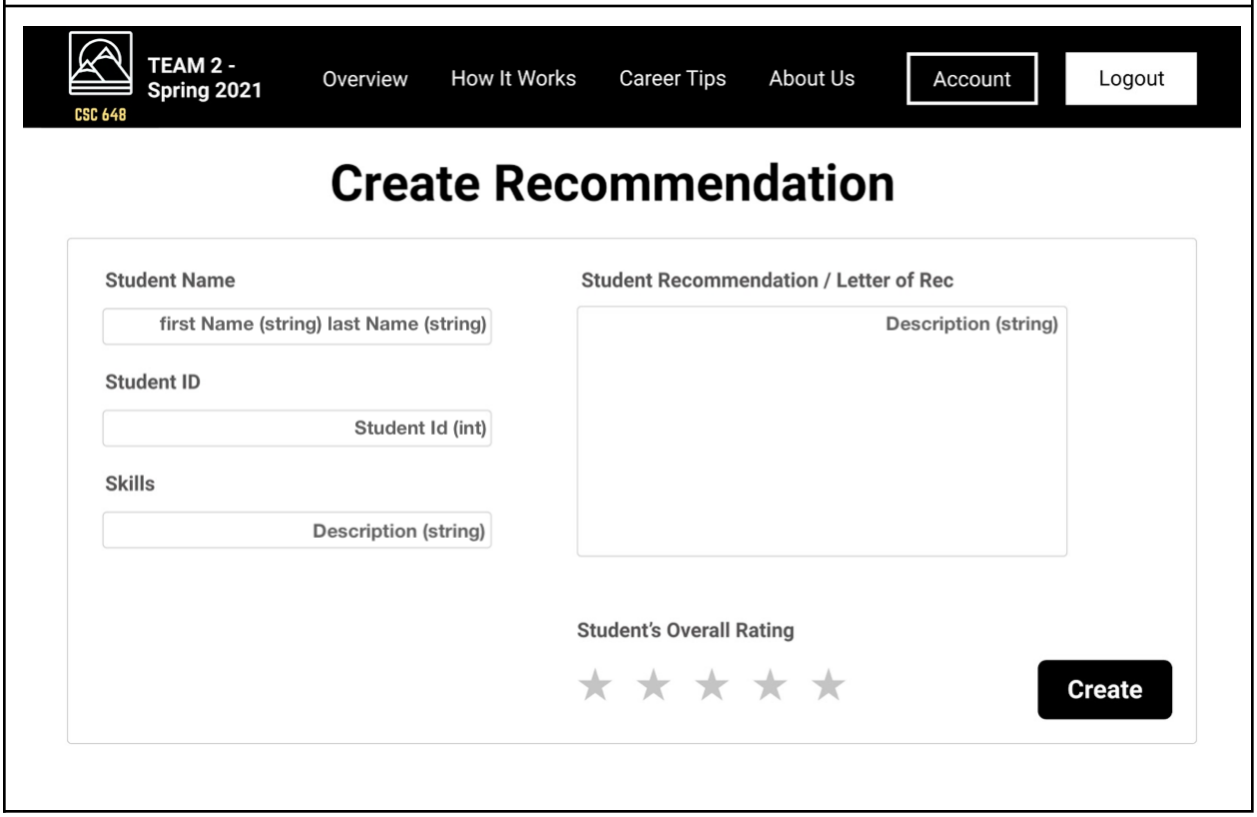



TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Create New Post
Position Title
jobTitle (string)
Compensation $
Salary (double)
Location
addressId (int)
Description
Description (string)
Gender
LGBTQ
Ethnicity (select all that apply)
Veteran
Disability
First generation college student?
Received Financial Aid?
Received Pell Grant while enroled in College?
Create
TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Job Post
Search Students
Position Title
Compensation
Location
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries.
EDIT
Position Title
Compensation
Location
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and

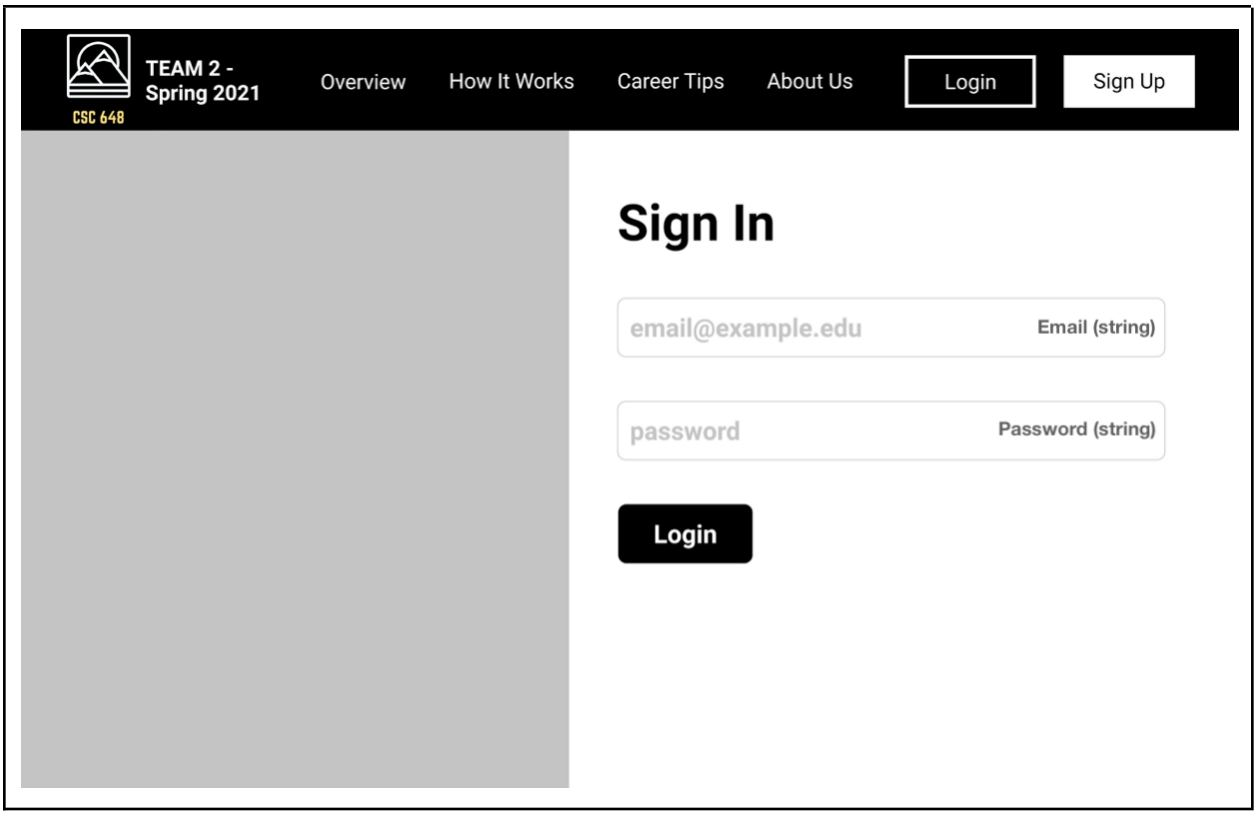
4. Once the professor is signed in, they shall be able to click the search button. The professor shall be directed to the search page, where they shall find the student they recommend. Once the professor presses the recommend button on the student's profile, they shall be moved to the create recommendation page. The professor shall be asked to rate in a scale from 1-5 fashion.

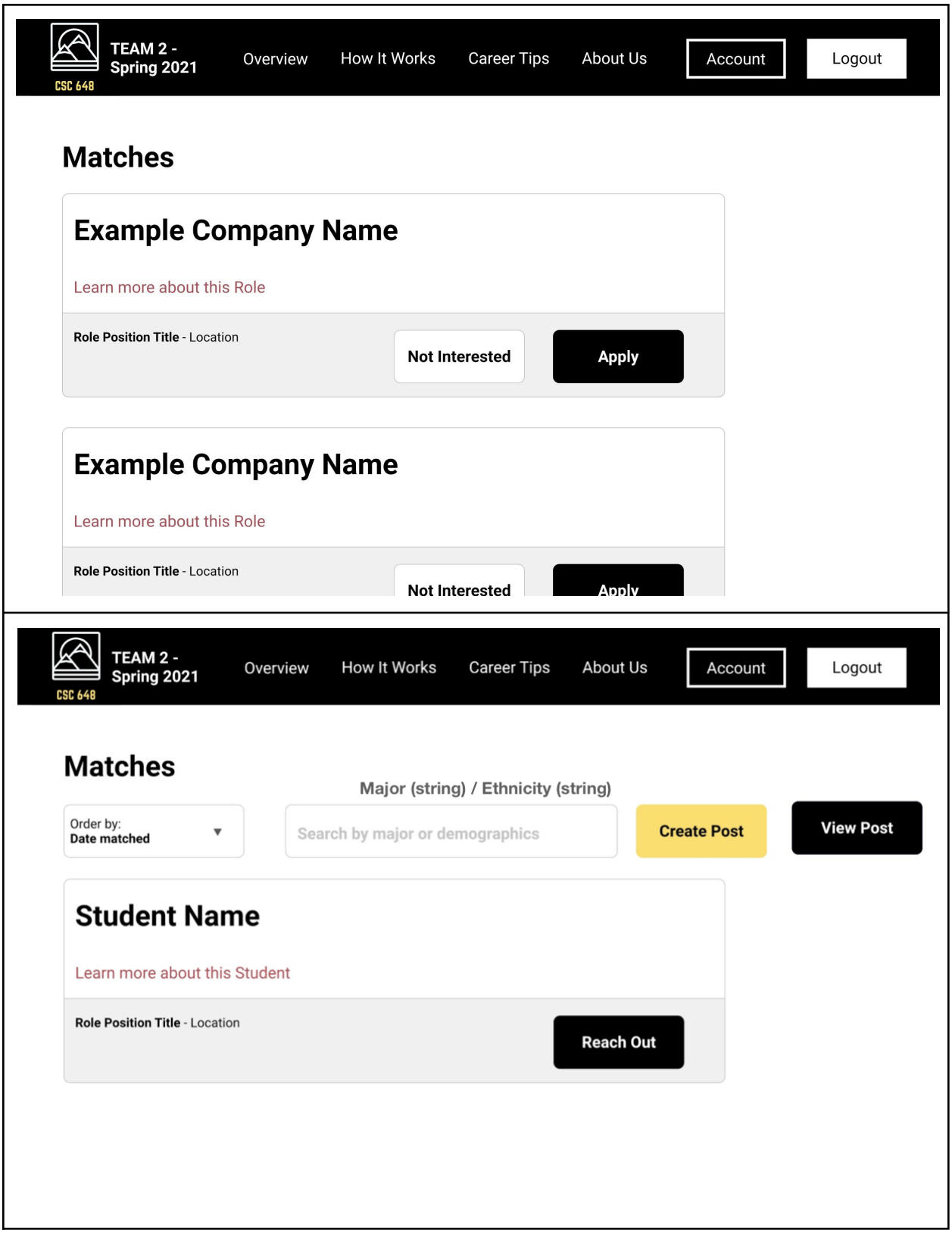



TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Recommendations
first Name (string) last Name (string) / Student Id (int)
Search by student name or student id
View Recommendations
Student Name
View Student portal
University Name - Location
Recommend
Student Name


TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Create Recommendation
Student Name
first Name (string) last Name (string)
Student ID
Student Id (int)
Skills
Description (string)
Student Recommendation / Letter of Rec
Description (string)
Student's Overall Rating
Create

5. Once the user clicks the login button, they shall sign in. When signed in, They shall see the matches on their homepage. The user shall be able to accept or decline the offer.




TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Matches
Example Company Name
Learn more about this Role
Role Position Title - Location
Not Interested
Apply
Example Company Name
Learn more about this Role
Role Position Title - Location
Not Interested
Apply
TEAM 2 - Spring 2021
CSC 648
Overview
How It Works
Career Tips
About Us
Account
Logout
Matches
Major (string) / Ethnicity (string)
Order by:
Date matched
Search by major or demographics
Create Post
View Post
Student Name
Learn more about this Student
Role Position Title - Location
Reach Out

3. High level Architecture, Database Organization

List of main data items and entities:

Student Table

userId	int
email	string
password	string
studentId	int
firstName	string
lastName	string
addressId	int
ethnicity	string
major	string
gender	string
aggregateRating	double
veteranStatus	enum
lgbtqStatus	enum
financialAidStatus	enum
disabilityStatus	enum
firstGeneration	enum

Recommendation Table

recommendationId	int
studentSFSUId	int
professsorId	int
rating	int
recommendationText	string

Recruiter Table

userId	int
email	string
password	string
companyName	string
description	string
addressID	int
websiteLink	string

Professor Table

userId	int
email	string
password	string
firstName	string
lastName	string
university	string

Address Table

addressId	int
streetName	string
aptNum	int
city	string
state	string
zipCode	int

Image Table

id	int
type	string
name	string
userID	int
data	longblog
createAt	datetime
updateAt	datetime

recruiter_savedSearch Table

recruiterUserId	int
textContent	string
dropDownOption	string

Media Storage:

Multimedia will be stored in a file system liked manner as of right now. We will do our best to avoid storing media into our database as BLOBS. We will look into possibly using CDNs but due to fund restrictions, that will most likely not be an option.

Search/filter architecture and implementation:

We will implement the API composition pattern for the search/filter architecture. Front end will send a request to the back end where the request will hit an API composer. From there, our composer's job is to query our databases based on a certain unique ID for a specific item, if we have multiple databases, the composer's job is to query all the databases and combine the results to then send back to our front end.

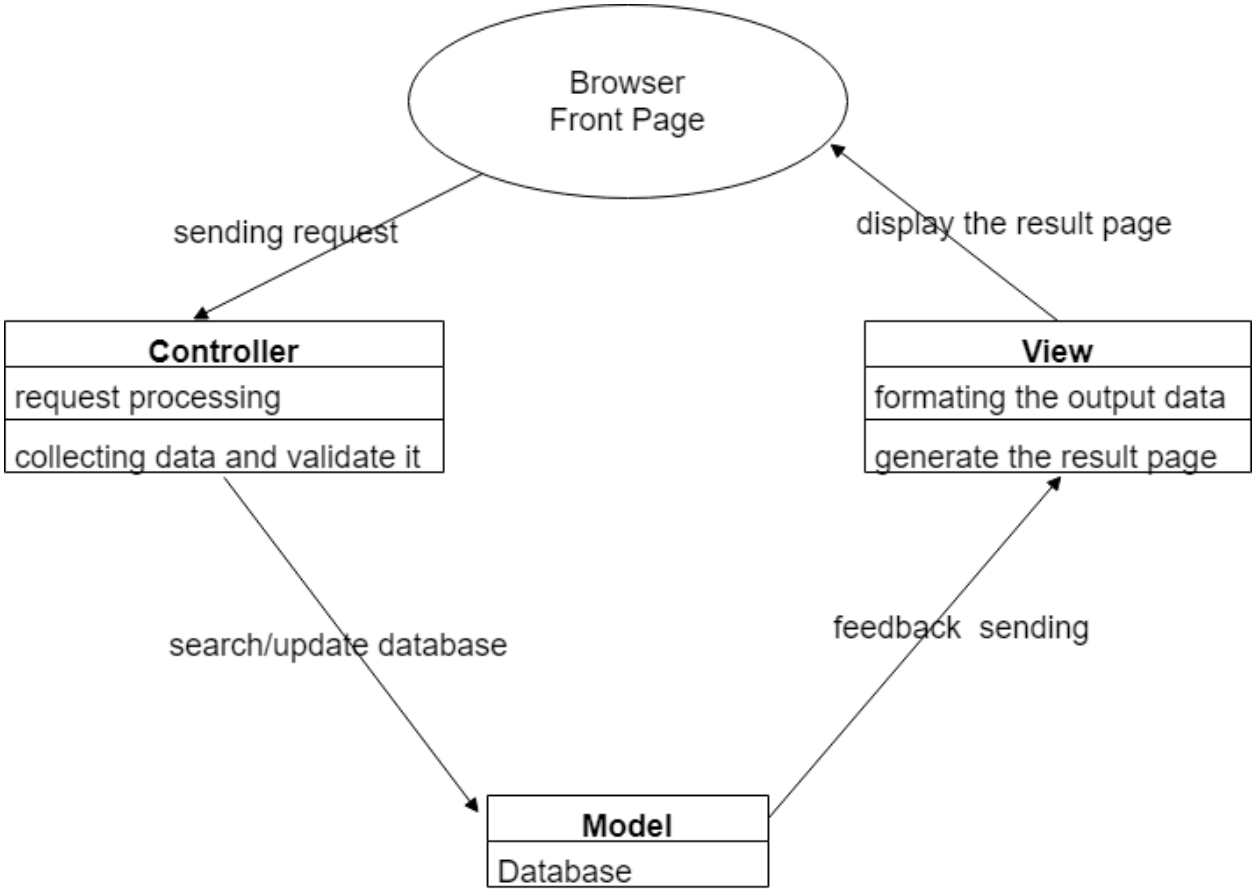
Your own APIs (if any):

1. ComposerAPI()
    a. Process request, query database(s), prepare/return response to front end

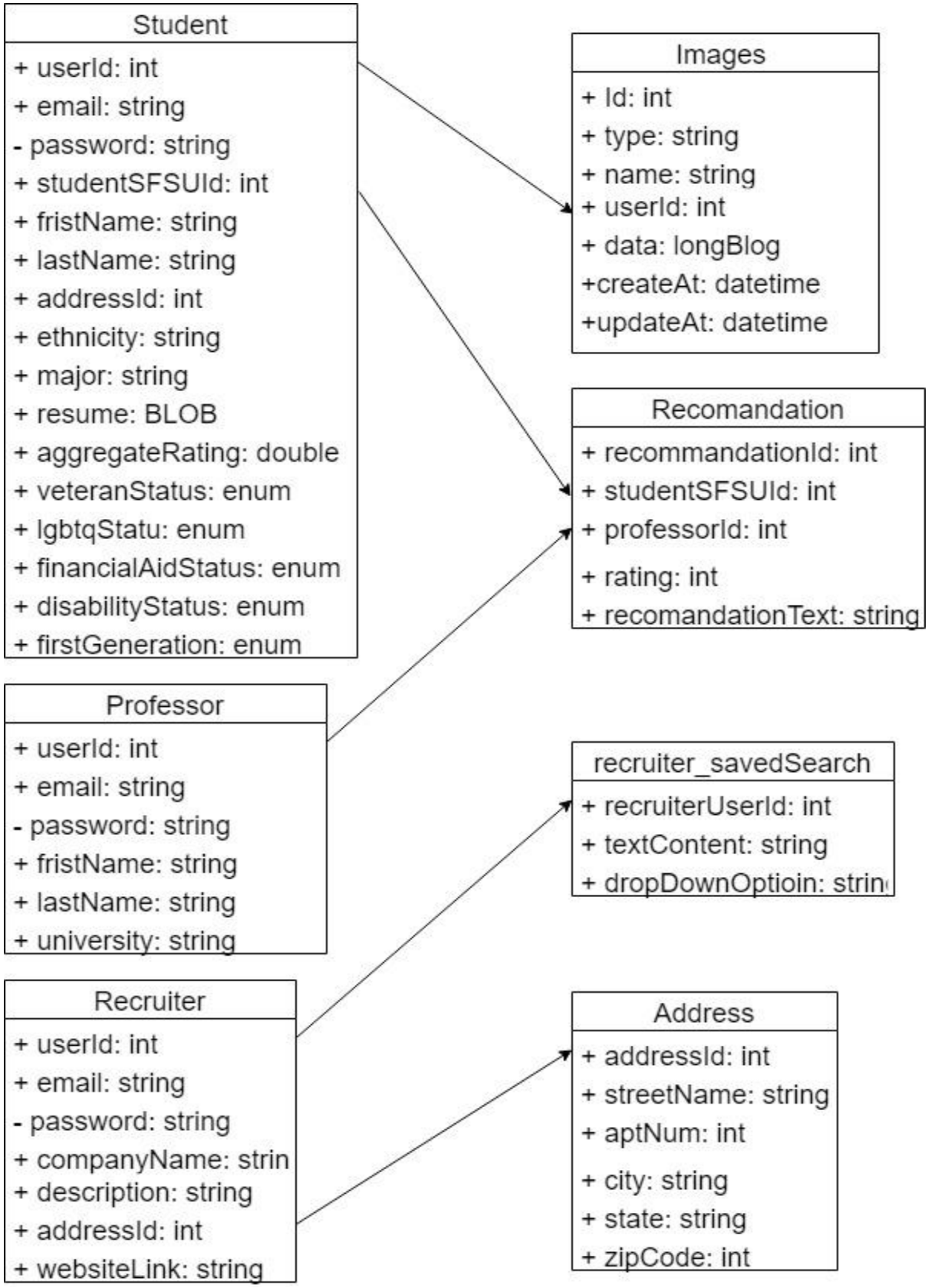
Below is a list of the technologies used in Team02's software stack, it remains unchanged.

Server Host: AWS t2.micro instance
Operating System: Ubuntu 20.10
Server Database: MySQL 8.0.23
Web Server: NGINX 1.19.6
Server-Side Language: JavaScript
Additional Technologies:
  Web Framework: React
  IDE: VSCode
  Web Analytics: Google Analytics
  SSL Cert: Lets Encrypt w/Cert Bot
  SASS: 3.6.4
  React Bootstrap: 1.5.0
  NodeJS: 15.6.0
  ExpressJS: 4.17.1

We will use the Model View Controller architecture. The front page will collect inputs from the users and then send them to the controller. There are a bunch of node.js which will handle different requests processing, data collection and validation etc. And it also gives direction on how to manipulate our database. Database is stored alone in the model component. And it sends out feedback to the view section, based on the result from the request processed. And finally, the view section generates the result page and sends it to the browser again.



4. High Level UML Diagrams



5 .Identify *actual* key risks for your project at this time

Identify *only actual and specific* risks in your current work such as (list those that apply:

- *skills* risks (do you have the right skills),
    - Our group members are not experts with the JavaScript programming language or the JavaScript libraries being used in this project.
    - Our group members have minimal experience with some of the technologies on our stack like AWS and/or NGINX.
    - FIX: We are all devoted to learning and improving ourselves as Software Engineers therefore, we will put in the time to learn these technologies needed to build applications.
- *schedule* risks (can you make it given what you committed and the resources),
    - Our group consists of full time students so there is a high risk in our ability to allocate time to work on the project on our own time and also not being able to align our schedule to meet up throughout the week to discuss the project.
    - FIX: We have all agreed to meet outside of class time for 30 minutes to an hour each week, however, we will communicate with each other if there is a change of plans and try to reschedule for that week.
- *technical* risks (any technical unknowns to solve):
    - Our group is currently unsure if our EC2 server will be sufficient to run our database, Node.js w/ Express, & React.js applications simultaneously. Memory optimization specifically may be necessary.
    - FIX: We will test our EC2 server as we develop our application and adjust our server as we go to make sure our app will run smoothly.
- *teamwork* risks (any issues related to teamwork):
    - There is a chance we might step on each others' toes as we work on the project throughout the semester due to the overwhelming amount of course work.
    - FIX: We have to keep in mind that we are all students and step into each others' shoes in order to be sympathetic while communicating with each other so we do not accidentally offend anyone.
- *legal/content* risks (can you obtain content/SW you need legally with proper licensing, copyright).
    - If the product were to be launched publicly, our team would have to take into consideration the legal ramifications of allowing user generated content on our site. Including creating and maintaining a system to remove false information as outlined in California Assembly Bill 35

- There is also potential for nefarious use of the application, including but not limited to identity theft, illegal job postings, and improper use of our product's search functionalities to discriminate in the hiring process outside of the parameters set by Affirmative Action on both the state and federal levels.
- As our project is educational by nature, and only a proof of concept not intended for public use, these potential risks are only theoretical and will not significantly impact development.

6. Project management

Discord and Asana are the main tools our team is using for the duration of this project. Tasks are delegated on Asana, these tasks pertain to milestone requirements, revisions, future preparations. Each major task has a detailed description and deadline. Discord is used to keep a log of conversations related to both specific tasks and big-picture items. Having a searchable chat log with well defined channels allows the team to review information at their own pace if they are not able to interact in real time. Weekly meetings are hosted on Discord, where each team member is to report on the progress of the tasks delegated to them, and discuss ways to improve upon the components when necessary. Larger architectural conversations that involve the whole team are held during these meetings to ensure team cohesion. Any new tasks delegated during these meetings are promptly added to Asana, and proper references and resources are recorded in their corresponding Discord channels. This process, alongside well documented code, allows for each team to operate fairly independently while encouraging cross-discipline communication on big-picture items.