SW Engineering CSC648/848 Spring 2021

PipeWave

"The Fast-Track from Student to Professional"

Team Lead/Front End: Jennifer Finaldi,
Team Lead Email: jfinaldi32@gmail.com
Database/Back End: Robert Cacho Ruiz
Front End: Kevin Danh
Back End: Jahir Hernandez
Github: Anthony Nguyen

Section 02, Team 04

Milestone 2-- 3/9/2021

History Table (Revision Summary):

03/09/2021: Document Created

04/01/2021: Revised P1 & P2

04/03/2021: Revised P1 & P2

1. Functional Requirements

Priority 1-

1. *All users* from verified schools or companies shall be able to register and create accounts on the website
2. *All users* shall be able to update their profile information
3. *All users* shall be able to view other's profiles
4. *Students* shall be able to upload resumes to their profile in a pdf format
5. *Students* shall be able to enter any demographics, experience and education to customize their profile
6. *Industry Professionals* shall be able to search for new graduates or students by major and demographics
7. *Industry Professionals* shall be able to register and get alerts for matching student or new graduate profiles
8. *Industry Professionals* shall be able to leave recommendations on students or graduates
9. *Professors* shall be able to rate students on a scale from 1-5
10. *Professors* shall be able to enter recommendations/reviews for students

Priority 2-

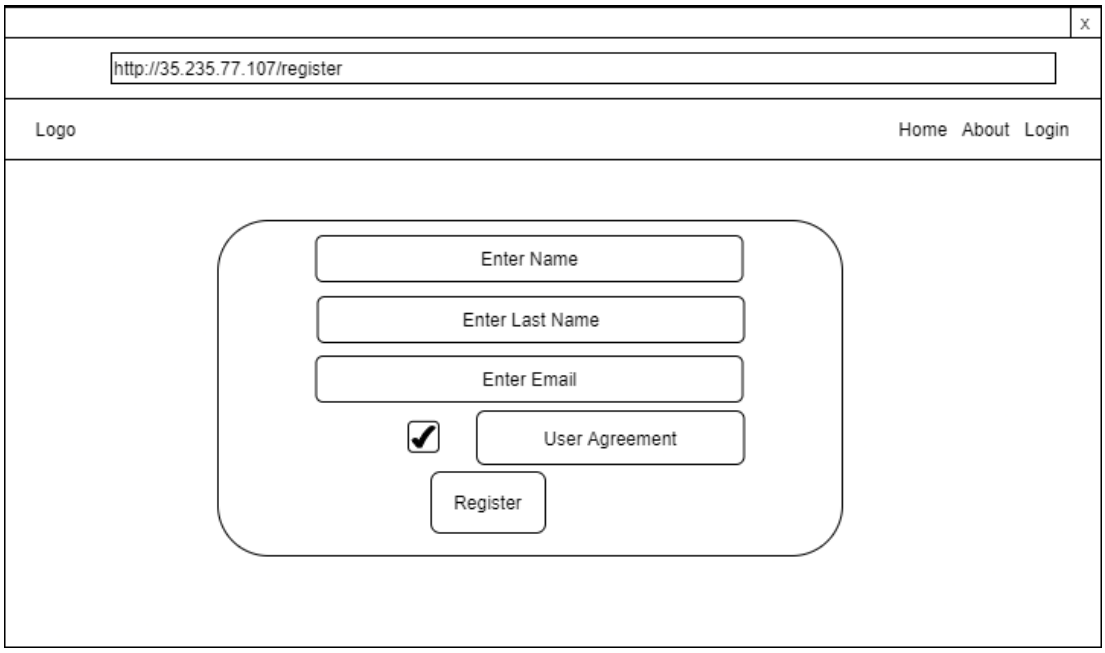
1. *All users* may be able to view reviews left on employer or employee profiles
2. *All users* may have the ability to use an in-site messaging system to communicate with other registered users
3. *Students* may be able to upload videos to their profile in supported video format
4. *Industry Professionals* may be able to follow profiles of candidates that they may be interested in without notifying them
5. *Industry Professionals* may be able to search for professors by demographics or subjects
6. *Industry Professionals* may be able to flag or notify a candidate that they are interested in interviewing through the press of a button

Priority 3-

1. *Students* may be able to follow other students or companies that they are interested in without sending notifications
2. *Students* may be able to leave recommendations for other students
3. *Professors* may be able to enter any demographics, experience and subjects that they teach to customize their profile

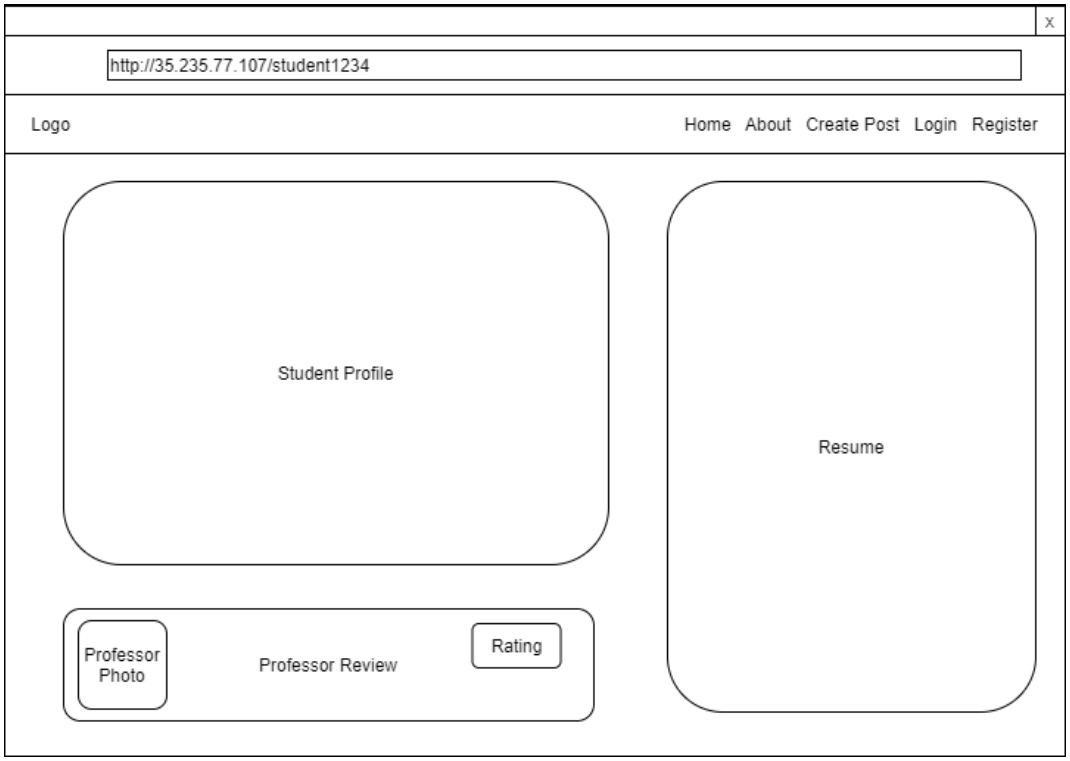
2. UI Mockups and Storyboards (front end)

Main Use Case 1: Recent Graduate Looking for a Job



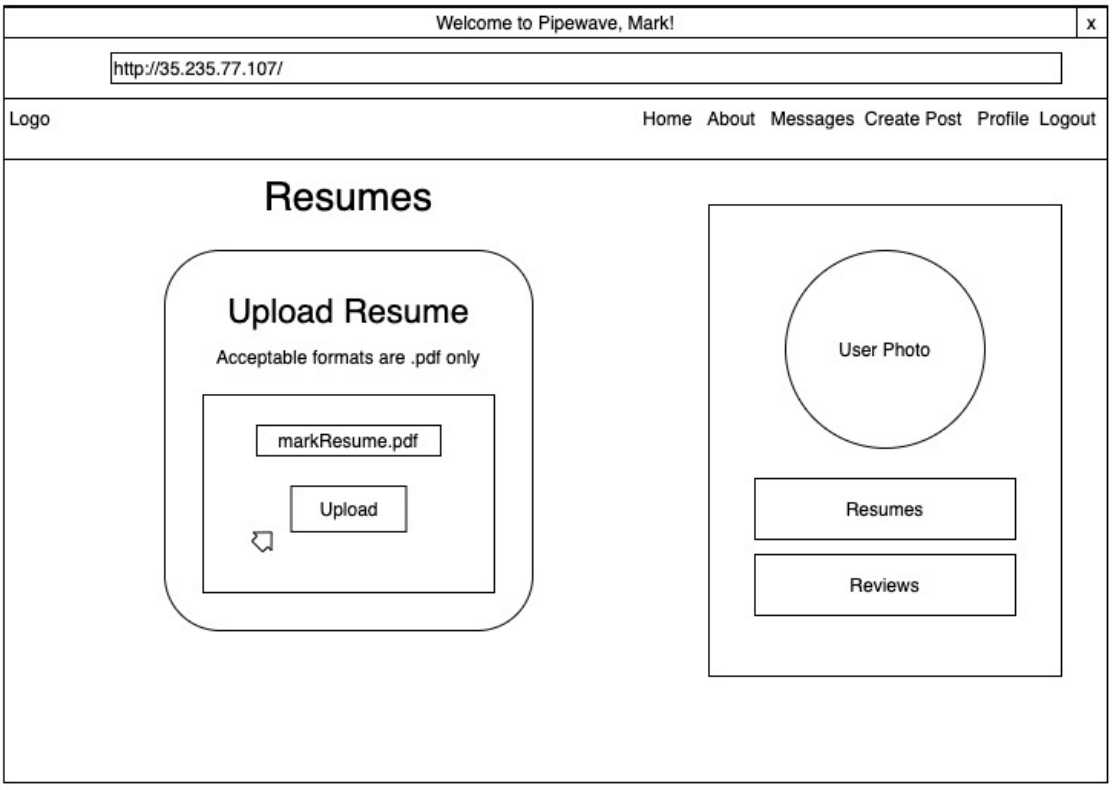
John is new to the website so he enters his information on the registration page and clicks on ‘Register’ to create a profile.

A professor viewing the website gets a notification that John has joined pipewave and is in need of a review.

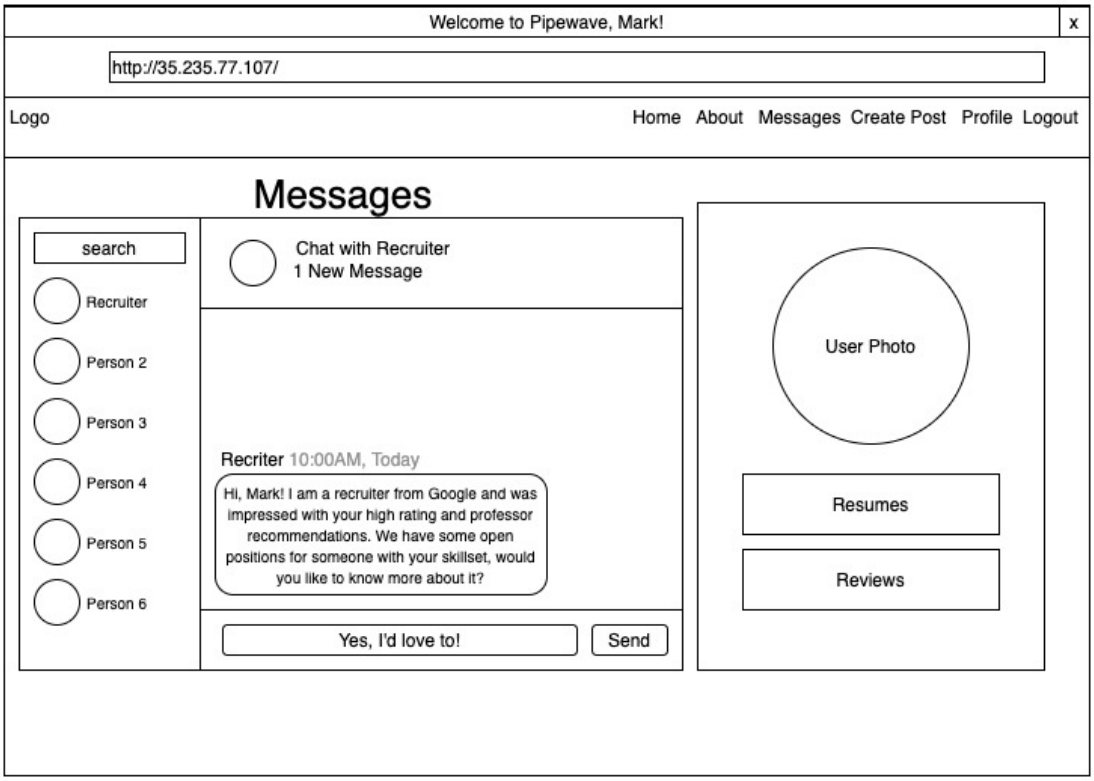


The professor leaves a review which is now displayed on John's profile.

Main Use Case 2: Current student looking for internship

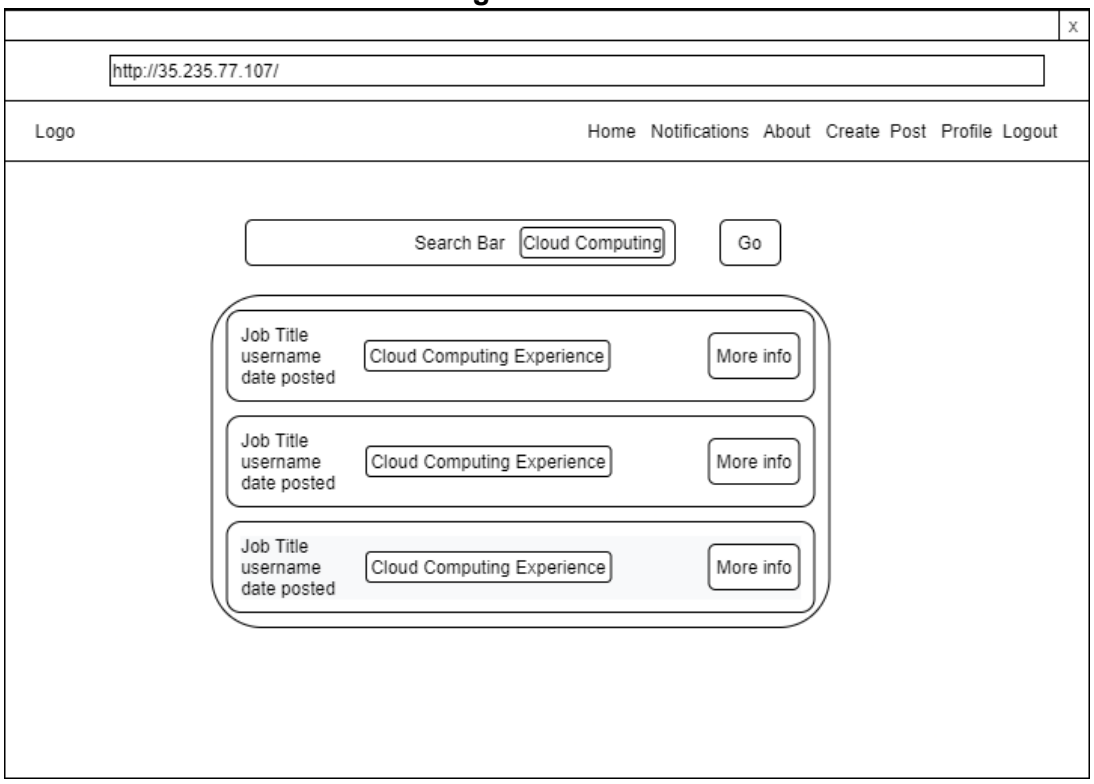


Mark selects a resume from his local device to upload to his profile, then clicks upload

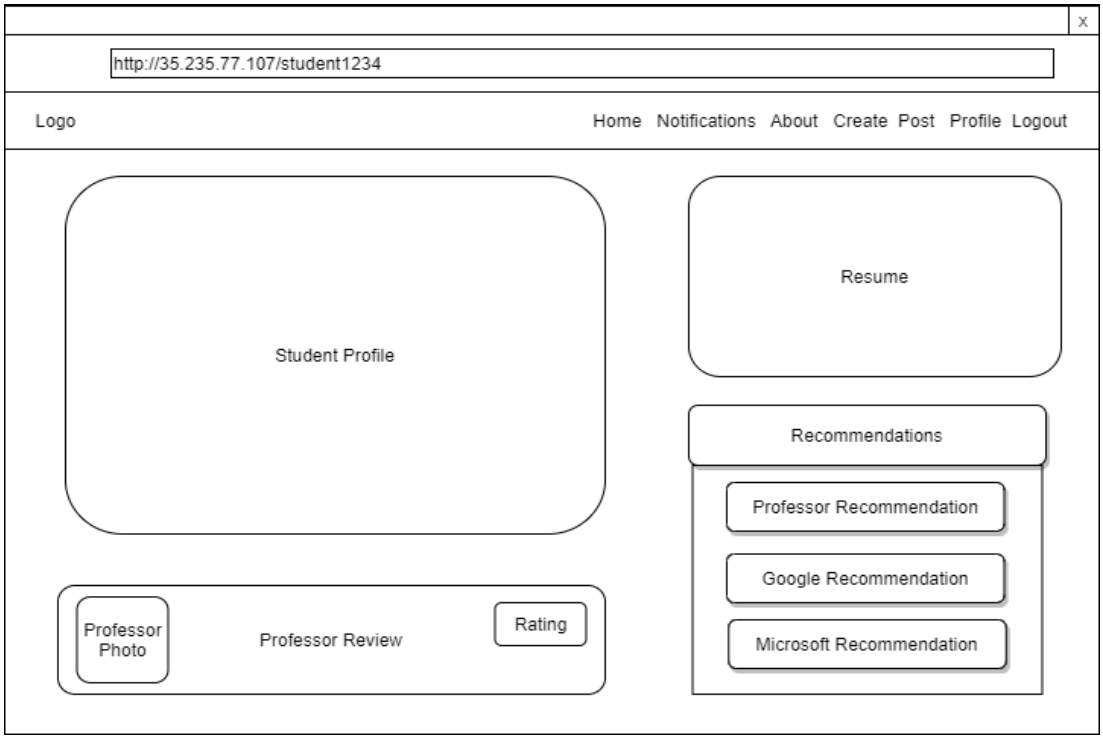


Mark discovers a message from a recruiter with a potential job offer and replies.

Main Use Case 3: Recruiter looking for overall talent

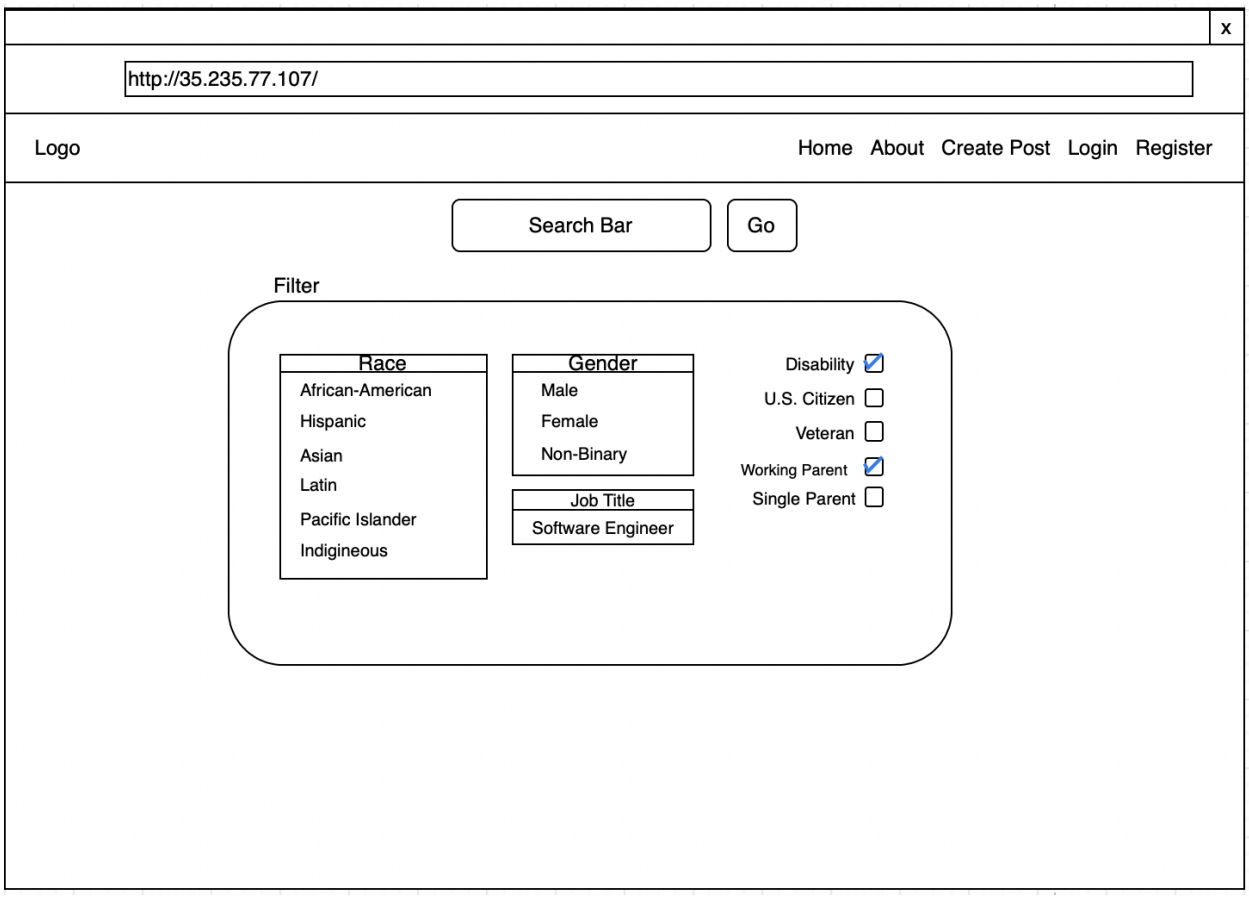


Linda is viewing the website for potential talent and filters based off of cloud computing experience.

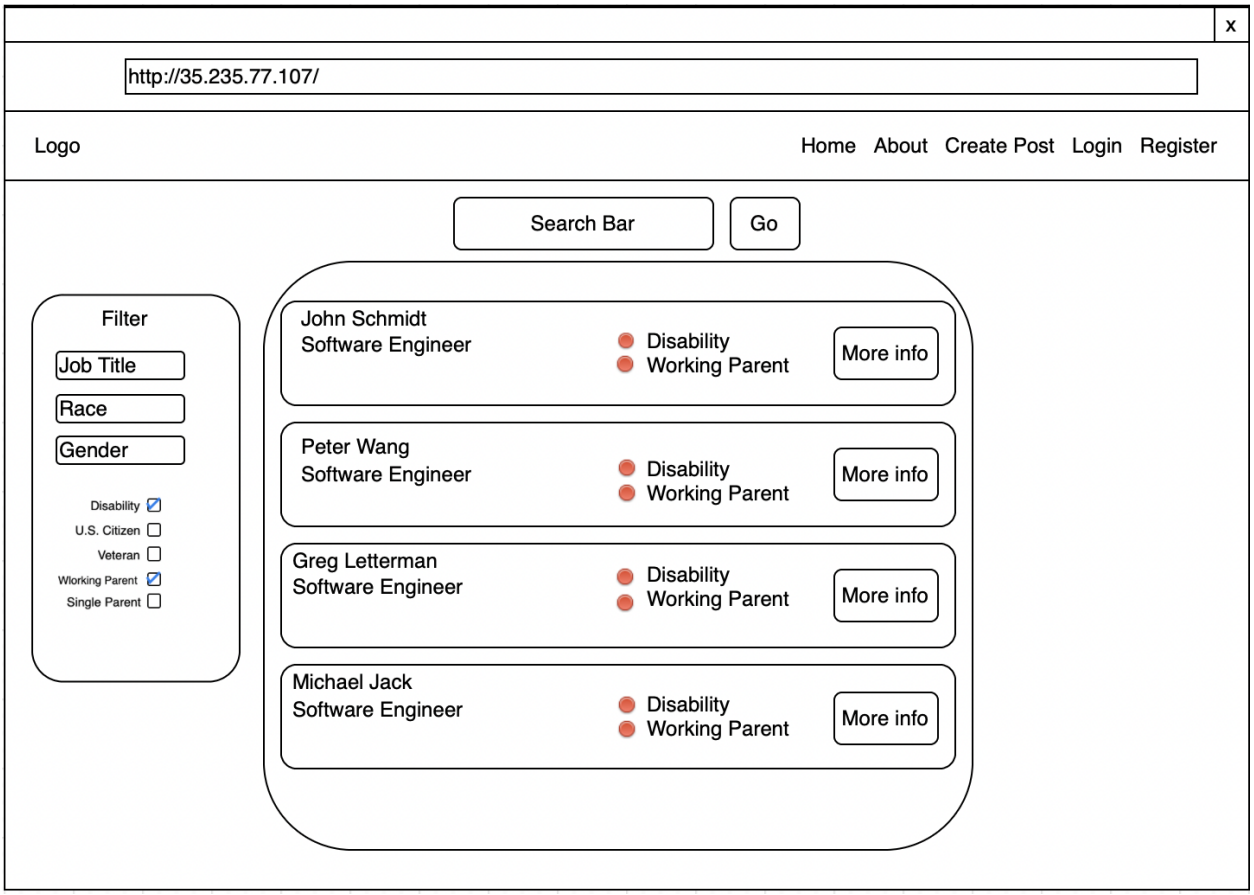


Linda sees the profile she's interested in and clicks the tabs to view to view the company and professor recommendations

Main Use Case 4: Current student looking for internship



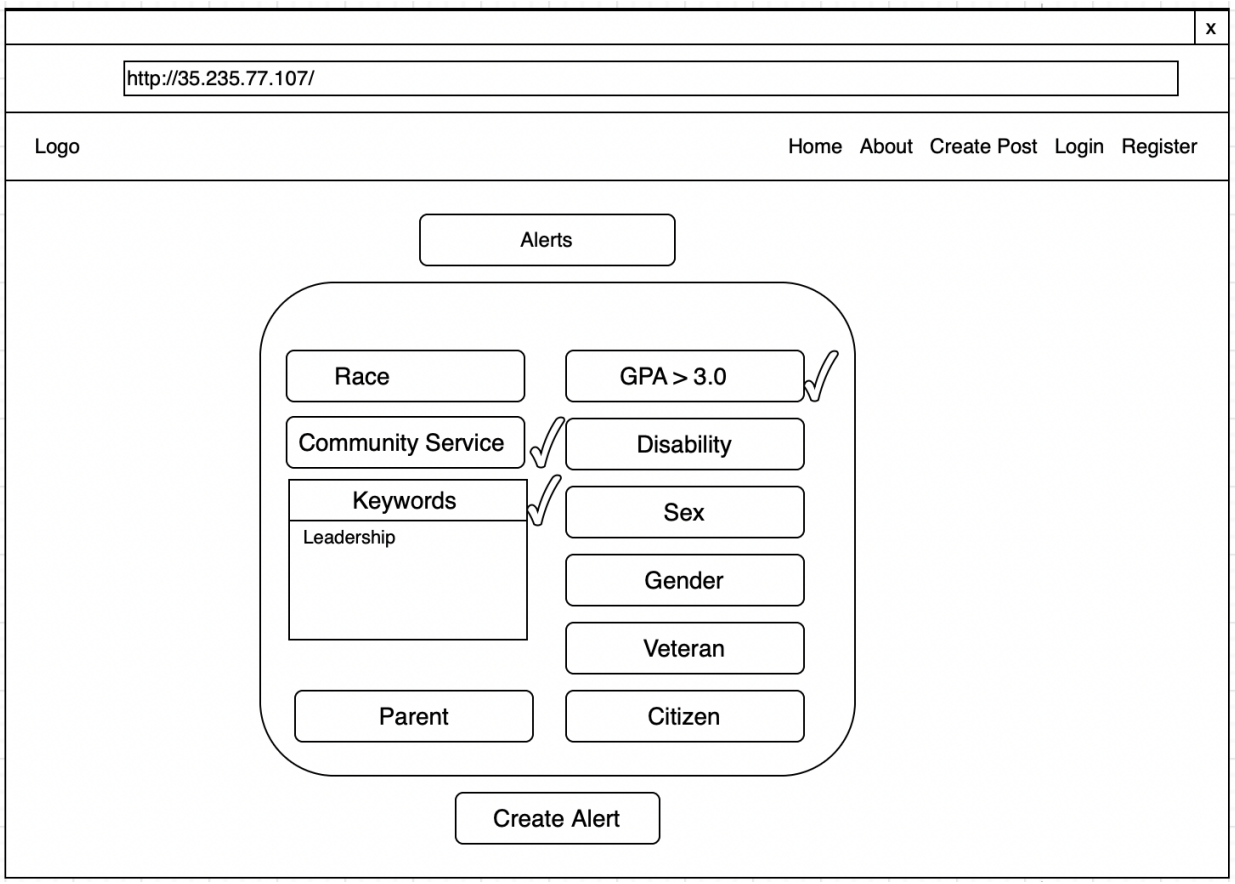
Joseph, an employer, looks to recruit candidate using the search function, filtering by disability and working parent.



Search results show up for Joseph, showing only disabled and working parent employees.

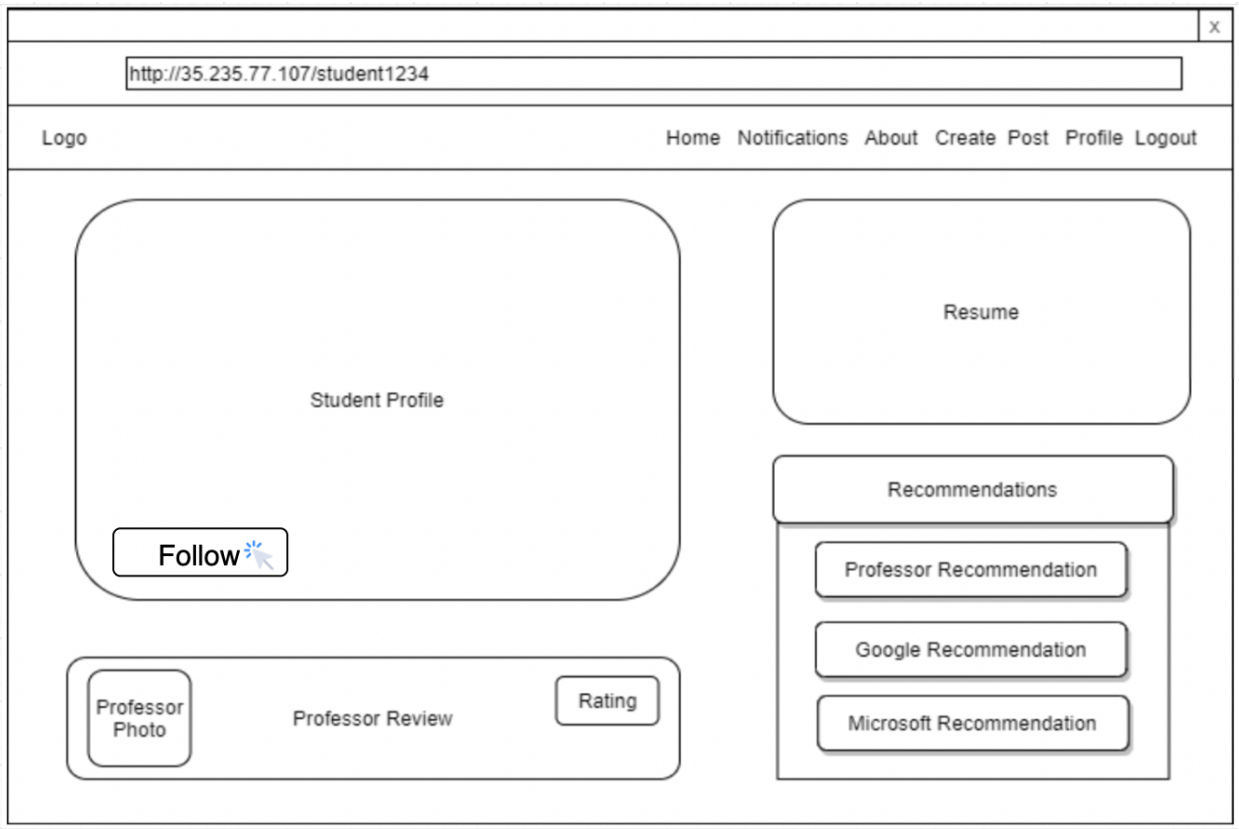
Main Use Case 5: Recruiter Creates Alert Page

Brenda begins to set up an alert page to follow recent graduates based on profile.



When a new candidate is found, Brenda is notified via website message system.

Additionally, Brenda can choose to follow other pages individually.


x
http://35.235.77.107/student1234
Logo
Home
Notifications
About
Create Post
Profile
Logout
Student Profile
Follow
Resume
Recommendations
Professor Recommendation
Google Recommendation
Microsoft Recommendation
Professor Photo
Professor Review
Rating

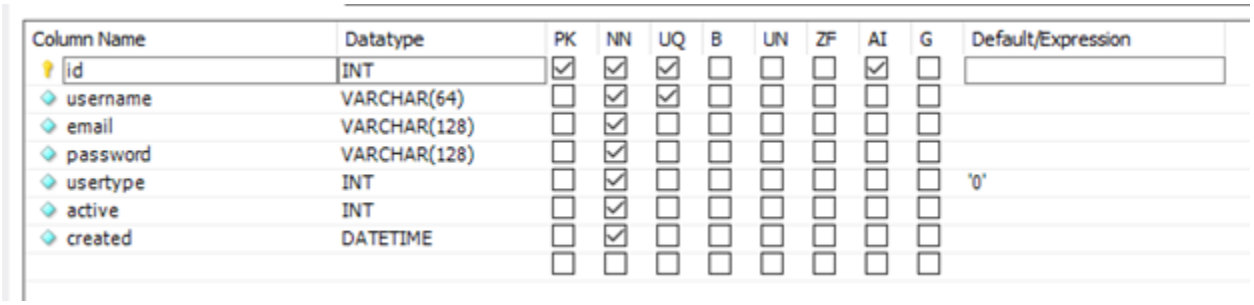
3. High Level Architecture and Database Organization

Database Organization:

Our team uses Mysql2 along with the mysql2 node package.

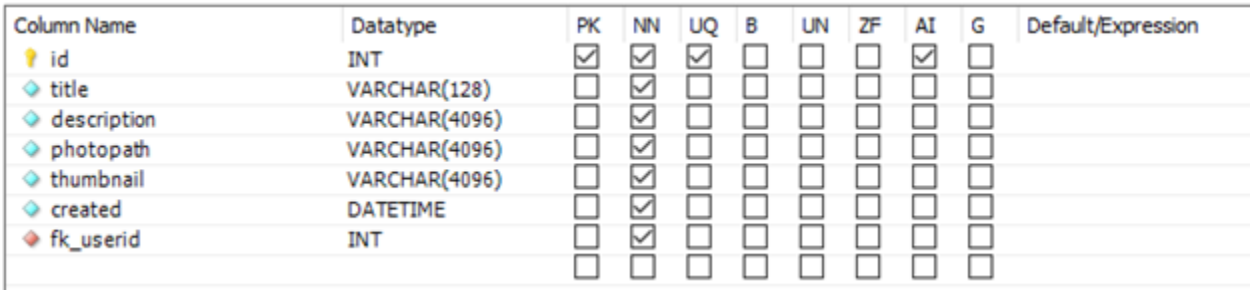
Currently we have a rough sketch of how our database will look.

Users Table



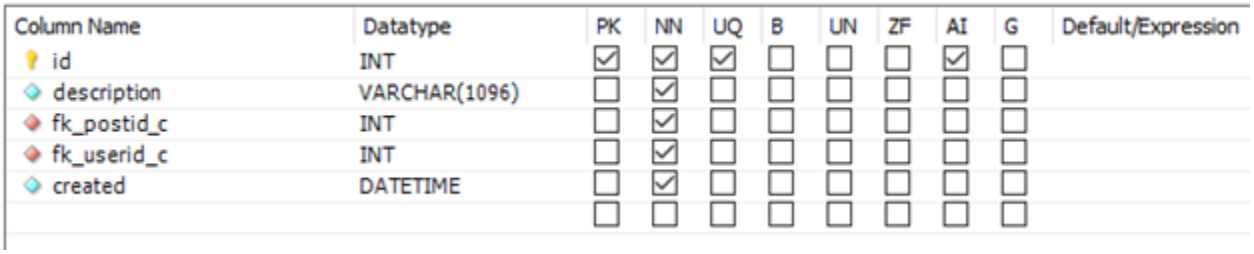
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	INT	☑	☑	☑	☐	☐	☐	☑	☐	
username	VARCHAR(64)	☐	☑	☑	☐	☐	☐	☐	☐	
email	VARCHAR(128)	☐	☑	☐	☐	☐	☐	☐	☐	
password	VARCHAR(128)	☐	☑	☐	☐	☐	☐	☐	☐	
usertype	INT	☐	☑	☐	☐	☐	☐	☐	☐	'0'
active	INT	☐	☑	☐	☐	☐	☐	☐	☐	
created	DATETIME	☐	☑	☐	☐	☐	☐	☐	☐	
		☐	☐	☐	☐	☐	☐	☐	☐	

Posts Table



Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	INT	☑	☑	☑	☐	☐	☐	☑	☐	
title	VARCHAR(128)	☐	☑	☐	☐	☐	☐	☐	☐	
description	VARCHAR(4096)	☐	☑	☐	☐	☐	☐	☐	☐	
photopath	VARCHAR(4096)	☐	☑	☐	☐	☐	☐	☐	☐	
thumbnail	VARCHAR(4096)	☐	☑	☐	☐	☐	☐	☐	☐	
created	DATETIME	☐	☑	☐	☐	☐	☐	☐	☐	
fk_userid	INT	☐	☑	☐	☐	☐	☐	☐	☐	
		☐	☐	☐	☐	☐	☐	☐	☐	

Comments Table



Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	INT	☑	☑	☑	☐	☐	☐	☑	☐	
description	VARCHAR(1096)	☐	☑	☐	☐	☐	☐	☐	☐	
fk_postid_c	INT	☐	☑	☐	☐	☐	☐	☐	☐	
fk_userid_c	INT	☐	☑	☐	☐	☐	☐	☐	☐	
created	DATETIME	☐	☑	☐	☐	☐	☐	☐	☐	
		☐	☐	☐	☐	☐	☐	☐	☐	

Media Storage:

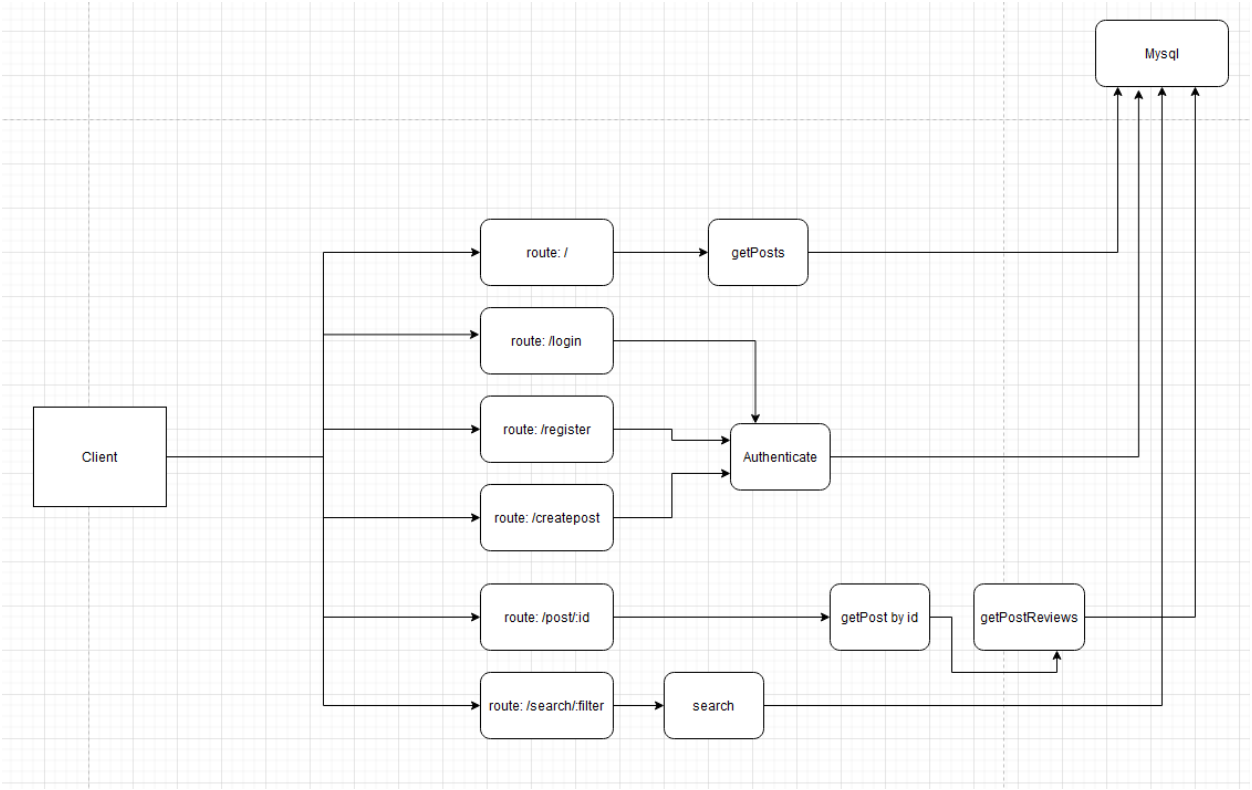
Our files (videos, images, audio) will be stored in file systems for our server.

Search/filter architecture and implementation:
To use search, we concatenate first and then search by using haystack. After users have entered their search request, the results would organize where users scroll down of all the results matching what they were searching for. The database terms would be centralized around the title of other users posts that they have created as well the username of other users.

Our own APIs (if any):
Currently, there's no plan of using our own APIs.

Non-Trivial Algorithms or processes (if any, like ranking, etc):
Utilizing mysql functions to get the reviews, where users are reviewed based on their profile. The review system can be based on a rating of 1-5 stars, as well as a post on their profile by a professor that will display some written feedback.

4. High Level UML Diagrams



5. Key Risks

1. (*legal/content*) Possible legal issues surrounding the search filter system. Allowing people to filter by marginalized groups tends to be a controversial social justice issue that could work against those who disclose any protected classes.

> Solution: It is decided to move forward with this functionality while understanding that it may at some point need to be changed or modified.

2. (*skills*) We have decided upon using Handlebars as a frameworks, however, only one team member from the back-end currently is familiar with it.

> Solution: All of the front-end team will have to study up on how to use it in order to take advantages of the benefits and conveniences of using it.

3. (*skills*) Our current codebase was primarily built by one individual team member and as such, the rest of the team may not understand all of the ways that the code is connected and how to modify it. This creates unnecessary obligation to the individual who built it, as they have to spend time guiding other teammates through the process.

> Solution: The builder may have to meet with the team to explain the code base in detail, to familiarize everyone with the specifics of exactly how it works

4. (*skills*) Some teammates are not familiar enough with web dev in order to make autonomous contributions to the project without guidance or mentoring by another teammate.

> Solution: The solution to part 3 may be able to assist in this. Otherwise, these teammates will have to do some independent study in order to prepare themselves to contribute. Or the teammate most familiar with the project can assign small attainable tasks for them to complete.

5. (*teamwork*) It can be difficult to delegate individual tasks to all members of the group because of the problems listed above in number 3.

> Solution: The teammate most familiar with the code base should be the one to assign these tasks, or effectively communicate or list the precise features that need to be implemented.

6. (*teamwork*) Coordinating meetings and communicating tasks to members can become a problem if some or all teammates don't regularly check our Discord channel for updates or info.

> Solution: Team leader should be responsible for making sure all teammates are regularly checking the channel, and if necessary sending group text messages to alert the team of any updates.

7. (*schedule*) Team members might have personal issues and other committments that will prevent them from working as hard as needed on this project.

> Solution: Members can update the team or team lead whenever anything comes up that may present a problem, so that tasks can be redistributed, or that certain features can be adjusted in terms of priority.

6. Project Management

_____In the planning of Milestone 2, we decided to break up the team into its respective front and back-end teams. The front end team decided to meet and discuss plans for revising and organizing the functional requirements, as well as plan out storyboards and assign each use case. After the front end team planned and distributed tasks, it was to all be completely asynchronously.

The back end team was given autonomy to plan their meeting and work sessions to tackle the high-level architecture and database organization sections, for them to complete asynchronously and independently. We are keeping track of progress through a shared google document, as well as the creation of a Milestone2 channel on our team's Discord channel, in which the team lead listed all of the tasks to complete and their status. It was decided by the team lead against using an external service like Trello for the sake of keeping all of the team's updates, resources, and discussions all in one place which is easy to manage. It is worth noting that a google group was created for Team Pipewave to easily keep track of and share all documents related to this project.

As for future tasks, the team will be dividing the Vertical Prototype milestone into front and back end teams. The back end will be responsible for tweaking and polishing our already existing search prototype, and the front-end will adjust the user interface to be more streamlined and in-line with the wireframe designs in this document.