StudentSurveys

*Senior Project Proposal*

Ahmed Kaleem Chaudhry

CSC489: Senior Project

February 13, 2023

Table of Contents

Summary 3

Significance 3

Required Tools and Availability 4

Demonstration Plans 4

Qualifications 4

Project Specifications 5

Technical Details 6

Timeline 7

Future Enhancements 9

### **Summary**

*StudentSurveys* is a web application that aims to help students at University choose courses, and professors by providing information and reviews from other students. The application requires users to sign up for an account and password to access its core functionalities, such as searching for professors and courses, reading and writing reviews, viewing overall professor ratings, and exploring an "About us" page. Additional features include pagination– which helps users move back and forth on the web pages with ease, a profile page that allows users to view their own profile, a bookmark feature which lets users save professor and course reviews for future reference, and account settings page that lets users change or retrieve forgotten passwords, ensuring that their information is secure. The application serves as a useful reference tool for students in their course selection process.

**Significance**

The project is based on the concepts covered in an online Web Development course, where I learned the fundamental building blocks of web programming starting from HTML, CSS, and JavaScript. The course then progressed to intermediate level concepts of Frontend, including ReactJs and other frontend frameworks such as Jquery and Bootstrap. In the second half of the course, I was introduced to backend development with an embedded database, covering technologies such as NodeJS, Express, Apis, and MongoDb. The course culminated in a final project that was a full-stack application, providing me with a comprehensive understanding of web development architecture and flow.

Similarly, *StudentSurveys* is a full-stack application that will build upon the concepts covered in the course, with added complexity and functionality. Further technical details about the project can be found in the relevant section.

**Required Tools and Availability**

*StudentSurveys* will utilize both hardware and software components. I will use my personal Mac book Air for the development. Virtual studio code will be the main text editor for writing the code. Some of the main technologies that will be used for the development are HTML, CSS, JavaScript, ReactJS, NodeJS, Express, and MongoDB – basically a Mern stack. All of this is available on my laptop, and I anticipate no problem in their accessibility.

# **Demonstration Plans**

All the work mentioned in the timeline will be presented on the due date of checkpoint meeting. The project can be easily displayed through my laptop. All the required software is installed. The only thing required would be an HDMI cord to present the project on the screen, and it could be easily accessed in the room. To make sure that everything goes according to plan, I will test my presentation before the actual meeting.

# **Qualifications**

I have acquired a good amount of experience in web development, which I initially gained through two 0.5 credit independent study courses that I undertook under the guidance of Professor Brian Howard during the winter and spring of last year. Although I continued to work on web development occasionally, I have made significant progress and honed my skills during this winter break. I am now confident in my abilities and eager to pursue this project.

### **Project Specification**

This project aims to assist students in making informed decisions during their course selection process. As a student, I personally experienced the struggle of choosing courses to fulfil my distribution requirements. Often, students at liberal arts colleges are not familiar with the content and teaching style of courses, leading to dissatisfaction with their chosen areas of study. With this application, students can gain insight into courses and professors before committing to a selection, ensuring a fulfilling and enjoyable educational experience.

* *Functional Specification:*
  + The web application will consist of two types of users:
    - General
    - Administrator
* *General User:*
  + Users can signup and login to the app
  + Browse professors, schools, and courses available on the app
  + Users can check the professors' overall rating and the courses they teach
  + They will be able to submit a review for the professor
  + Upvote or downvote the existing reviews of the professors
  + Reply to the existing review
  + Bookmark the course for future reference
  + Can update their account settings, like password etc.
  + Can see Dashboard to check their profile stats
* *Admin User:*
  + Can see Dashboard to check the overall stats of the app
  + Add/Edit/Delete professors from the app
  + Add/Edit/Delete courses of the professor
  + Add/Edit/Delete schools from the app
  + Add/Edit/Delete users from the app

### **Technical Details**

* *Frontend:*
  + I will be using React JS for the frontend of the app
  + CSS for styling the app
  + Redux toolkit for global state management within the app
  + React hooks for example useEffect, useState
  + I will be using material ui for ui components to speed up the process
  + Some of the packages that I am going to use includes:
    - React Router DOM: For making routes in react
    - Redux Toolkit: For state management
    - Axios: For making api calls
* *Backend:*
  + Node js will be the backbone of the app
  + Some of the packages that I am going to use include:
    - ExpressJS: For web server
    - JsonWebToken: For generating unique tokens
    - CORS: To enable cors in the apis
    - Mongoose: An ORM for mongodb to work seamlessly with mongoDB
  + Deployment on web server
* *Database:*
  + MongoDB will be the database for the app
  + We will be using [this](https://www.mongodb.com/) service to host the database and connecting with it remotely
  + Will follow the best practices for database like IP whitelisting to secure it

### **Timeline**

* *Checkpoint One:*
  + I will demonstrate some basic apis for the app using postman
  + Authorization apis
    - Signup and login apis for users
    - Signup and login apis for admin
    - Encrypted password/ hashing functions
  + Profile apis
    - View professor profile with the courses they teach
    - Fetch & Update users’ profile
    - Deleting user profiles
    - Deactivating user profiles
    - Add/Edit/Delete professors from the app
    - Add/Edit/Delete courses of the professor
    - Add/Edit/Delete schools from the app
    - Add/Edit/Delete users from the app
    - Total reviews made by the user displayed on the dashboard
  + Search functionality
    - Search by professor name
    - Search by course name
    - Search by school name
* *Checkpoint Two:*
  + Demonstrating the remaining apis:
    - Reviews
      * Viewing existing reviews
      * Adding new reviews
      * Upvote / Downvote reviews
      * Responding to existing reviews by general user
      * Deleting review by admin only
    - Bookmarks
      * Viewing previously bookmarked items
      * Add new bookmarks
      * Remove bookmark items
    - Forgot password apis
      * Change password apis
* *Checkpoint Three:*
  + Will demonstrate the frontend of the app with integrated apis
  + Site will be very user-friendly. Some additional features include:
    - There will be a rating chart of overall ratings for the professor.
  + About us page or section for details of the app
  + Common components include:
    - Header
    - Footer
    - View all teachers
    - View specific teacher
    - Ratings component
    - Add review component
    - Delete review component
    - Responding to review component
    - Admin dashboard component
    - User dashboard component
    - Add, remove, and view Bookmark component
    - Upvote and downvote component
  + Profile page for users
  + Search results page
    - Search by professors
    - Search by courses
    - Search by school
* *Checkpoint Four:*
  + Remaining pages for the app with integrated apis
  + Deployment on web server
  + Forgot password demonstration
  + Overall project improvement

# **Future Enhancements**

* To provide a better user experience, a chatbot feature could be implemented for its user-friendly interface.
* Rather than relying on manual data entry by an admin, a more efficient way to populate the system with information on professors, schools, and courses could be through web scraping. By using automated tools to extract relevant data from publicly available online sources, the system could quickly and accurately gather information without requiring human intervention. This approach could save time and resources, while also ensuring that the data is up-to-date and comprehensive.
* Another potential enhancement to consider is the addition of a sharing feature that would allow users to easily share their reviews with friends on other social media platforms such as WhatsApp and Facebook.