Collegium Sugit

College sucks, so learn better.

Purpose

The purpose of Collegium Sugit is to provide a Student-Educator model similar to the SI model which helps students learn in an interactive / hands on approach.

Profit Model

The profit model for this website is to charge an inconsequential fee (such as $10) for access to a course that directly rivals a college course. Instructors will be able to make courses for free and receive a share of the profit as incentive to share their knowledge.

React

The forms and the controls on the page are rendered with React through the library React Habitat, which allows the react components to be written directly into the html. This approach has downsides, and I would not be using it if I did not have to use a templating framework (as React can easily stand on its own).

MVC

The model-view-controller setup is implemented with the combination of Mongodb for the models, Express.js for the controllers, and EJS and React to serve the views. The file structure of the project demonstrates this separation of concern.

Mongo

The models that were stored in the Mongo database include the Account objects, the Asset objects, and the Course objects. You can create different types of accounts which see slightly different pages, and have different privilege levels. You can create Assets, which currently only include images and specify the location the asset is stored on an AWS S3 server. And you can create Courses, which hold some general information about the course.

Templating Language

The templating language used was EJS, the built in templating language that comes with Express.js. It is used to embed data such as the user, courses, and profiles into the different pages. In my opinion this is not the cleanest way to use React, and it was only added to satisfy the requirement of a templating language.

Above and Beyond

The first part of this project that is above and beyond is refactoring all of the Domomaker code instead of just taking the Domos out. All of the code is reorganized and multiple bad practices have been fixed, as well as prefixes and descriptive variable names added to increase the readability of the code. I would have no problem showing this code to a potential employer.

The second part of this project that goes above and beyond is the better build environment added by Webpack and Gulp. To build and test the website, only a single command is needed, npm run build. To watch for updates, only a single command is needed, npm run watch. Additionally, some of the technologies that I installed on my machine have helpful setup scripts added.

The third part of this project that goes above and beyond is the SASS styling that I wrote and integrated into Webpack, which includes several custom elements such as themed radio buttons and inputs, increasing the user experience.

Fourth, Amazon Web Services was added to the project to implement real image uploading. This involved creating a properly configured AWS account and bucket, and then linking the credentials in a safe way that would not expose them to Github.

Fifth, Google Cloud Platform was added to the project to implement automatic filtering of images. Because the app is public facing and linked to my Heroku account, I did not want to risk pornographic and/or illegal images being hosted on my server. So to avoid this, I set up a Google Cloud Platform Project which uses the Safe Search API to prevent adult images from being uploaded to my S3 bucket.

My Self Evaluation

There were definitely some struggles along the way, but in the end I am happy how this project turned out, and I definitely plan to continue working on it.

I believe this is A level work, and I am requesting that this project counts toward both my Project 3 grade and my Project 2 grade, and here’s why:

From the start, this was an ambitious project and required rewriting the base level code to allow building on the Domomaker code to avoid accumulating code debt. Additionally, adding every technology that was taught in class demonstrates that I understand the content of this class. To get an A in this class, all I needed was a 60% on the second project and a good grade on the final project. This would have been possible except that I simply did not have the time, and this was partly my fault (breaking my wrist being one of the major setbacks that I did not seek an extension for, and probably should have), but also due to the fact that the Project 2 due date was unexpectedly shifted an entire week earlier.

I think that with all these factors considered I have put in enough effort and demonstrated enough ability to get an A in this course, even if the timing was a little off. I hope that you dont take any of the complaints in this against you as that is not my intention, as regardless of my grade this has been far the most helpful class I have taken this semester, and my only regret is not having time to attend as many of the lectures as I really like your teaching style because it is super easy to pick up new concepts.