Final Exam Project (CMSC335 (Spring 2022) (https://www.cs.umd.edu/class/spring2022/cmsc335/))

Overview

The goal of this final exam class project is to practice concepts covered in class by implementing a simple Node.js application. What the application does is up to you. For this work you can work with classmates or by yourself. **There is no late deadline for this work. The deadline for your submission is Monday, May 16, 3:30 pm.**

Requirements

1. Design and implement a Node.js application.
2. Your application must rely on Node.js and Express.
3. Your application must rely on MongoDB.
4. Your application must have at least one form where users provide some data.
5. Your application must use some CSS.
6. Your application must rely on at least one API. Some API resources (you are not restricted to only these resources):
    a. https://rapidapi.com/ (https://rapidapi.com/)
    b. https://rapidapi.com/blog/rapidapi-featured-news-apis/ (https://rapidapi.com/blog/rapidapi-featured-news-apis/)
    c. https://rapidapi.com/blog/how-to-use-an-api/amp/ (https://rapidapi.com/blog/how-to-use-an-api/amp/)
    d. https://developer.vonage.com/blog/2021/03/15/the-ultimate-list-of-fun-apis-for-your-next-coding-project (https://developer.vonage.com/blog/2021/03/15/the-ultimate-list-of-fun-apis-for-your-next-coding-project)

Grading

1. (40%) Application relies on Node.js and Express.
2. (10%) Application relies on MongoDB.
3. (10%) Application relies on at least one form.
4. (5%) Application makes use of some CSS.
5. (30%) Application makes use of some API.
6. (5%) Application is deployed online (e.g., using https://www.heroku.com/ (https://www.heroku.com/)).

Additional Information

1. Your application can be as simple as you want. You will get full credit as long as you satisfy the above requirements.
2. You can work by yourself or in teams of at most three people (no more).
3. You should use this work as an opportunity to create an app that you can add to your resume and can show during a career fair :). Also, you might be consider for recommendations or TA positions if excellent is completed.
4. Feel free to help classmates (even if they are not part of your group).
5. **Applications that are extensions of class projects will not receive any credit.**
6. You must have a text file called README.txt in the main directory of your application. This file will have the information specified below.
7. Please, check the clarifications Piazza folder associated with this work often.

README.txt file

You must have a text file called **README.txt** in the main directory of your application with the following information:

1. **Team Members** - Add the name and directory id (in parentheses) of each member of your team. If you are working by yourself, add your name and directory id too.
2. **App Description** - One or two line description.
3. **API links** - Links to API(s) you are using.
4. **YouTube Demo Video** - Link to a YouTube video that provides a demo of your application.

Submission

1. Remove the **node_modules** folder from your application. Make sure we can recreate this folder when executing **npm install**.
2. Make sure the README.txt file described above is part of the project. It should appear in the main directory of your application.
3. Upload a zip file with your work using the submit server entry "Final Exam."
4. **Make sure that after uploading your work, you download the work and verify you are not missing any files. If you are missing any files, you will be penalized -12 pts.**
5. **We will only grade the last submission you provide. If you provide both an on-time and a late submission, only the late submission will be graded and a 12 pts penalty will be applied.**

Web Accessibility (https://www.umd.edu/web-accessibility/)