CISC / CMPE 327, Fall 2024
Course Project Assignment #2: Front End Development and Testing
Due: Monday, October 21, 2024 (plus 24-hour automatic extension if needed)

In this assignment, you will start developing your group's project by building upon the repository you created in the first assignment. To ensure a fair evaluation, it is essential that each group member contributes equally. One way to show your contributions is through GitHub commits. By tracking the total number of commits, we can assess each member's involvement in the project.

What you need to do to complete this assignment.

1. First, make sure your project from the last assignment is up and running.
2. We highly recommend that you use the modular/MVC pattern while developing the project to ensure that your application is flexible enough for testing. Python-based web frameworks, such as Flask or Django, are preferred, although other frameworks (ReactJs, NextJs, Laravel etc.) can be used if you are more comfortable with them.
3. Implement at least three features (UI and logic) for which you have already created UI mockup in your last assignment. For this, you need to create necessary functions along with the UI within your chosen framework to implement the features. If you require backend connectivity, at this stage you can just mock the required backend services with data objects such as array, vector, map etc. The actual backend connectivity will be implemented in the next assignment (for enthusiasts, you may also create DB and write DB connectivity logic, but this part will not be marked).
4. Write test scripts (success and failure cases) and save the script in files. This is to unit test the functionality you just implemented. If you choose a Python-based solution, you can use unit testing library or framework, such as **unittest** or **pytest**.
5. Execute the test scripts and take the screenshots to add them to GitHub in a file named **screenshots** under the **Assignment-2** folder in your repository.
6. Please create a clear instruction file in markdown or PDF format that outlines how to run your project and execute the test scripts in **readme-first** file under **Assignment-2** folder. This is crucial, since the inability to successfully run your project will restrict us from evaluating your hard work.
7. While we track contributions through GitHub commits, we also require you to create a table to list the contributions of each member in **task-distribution** file under **Assignment-2** folder.

What will be checked?

Your solution to this assignment will be judged on the clarity and readability of the design, the code, and test script. Follow best programming practices, including naming

variables, classes, and methods. Clear commenting on the code will make it easier for anyone to follow your code.

How you should start and submit your work:

From your local GitHub repository, each member must checkout to a new branch and complete your work as you distributed among your group. Remember to make short commit with meaningful message and description (where applicable) after making some changes. We do not expect that you will make just one/two commits for the whole assignment. Before submitting, the team leader will merge all your work from the branches to the main branch and push it to remote repository. **[this process will be shown to you in a separate online session if needed**]

Marking Criteria: Marks will be assigned between zero and the number of marks shown to a resolution of 0.5 mark.

Working Source Code (Details of the source code) **3 marks**

Structure and format

- code is structured and formatted
- naming of classes and methods clearly reflect their role in the solution
- as little cloning or redundancy as possible

Maintainability

- simplest solution possible
- clearest solution possible

Test Scripts **2 marks**

Test Execution and Screenshots **3 marks**

Documentation **1 mark**

========

Total **9 marks**