WebApp Project Requirements Specifications

Assessment weightage (10%)

Problem Statement

Design and implement a web app using Python and Flask that allows a user in your school to accomplish one or more tasks. The task must be useful and appropriate to be deployed as a Web Application.

Submission 1: Use Case Proposal

To be submitted by end of Term 2 week 10.

Use Case Example 1:

When new students and staff join the school, they will not be familiar with the various rooms and facilities in the campus. They will also not be able to navigate from one place to another. The school will usually include a tour of the campus compound during student and staff orientation programme. The objective of the project is to build a web application that will allow a virtual tour of the campus to be conducted anytime.

Use Case Example 2:

Due to the large influx of new members in the NJ Film Society CCA, a large number of projects are currently being done by different groups of members. These projects are widely different in terms of aim, direction, deadlines, and members involved. Members may also enter or leave a project midway through production. This has led to the current EXCO members to face difficulty in managing and tracking each project, as they only use WhatsApp groups to communicate with the groups as well as many disparate Google drives and documents to do so. The objective of the project is to build a web application that will allow members of the CCA to

- easily view current or completed projects.
- add a new project.
- edit/update project status.

Submission 2: Final Web App Source Code

To be submitted by Term 4 Week 1

Project Requirements

1. Dynamic contents.
2. Contents must be stored, retrieved, updated and deleted in a sqlite database
3. Web portal must have a navigation system that allows a user to navigate to the different contents
4. Web Contents must meet the usability (UI/UX) design principles.

Marking Rubric

Front End

Note: You may use an internal/external CSS framework but marks for CSS will be awarded only for definitions that are created by you.

HTML/CSS	Page Organisation and Navigation	Additional features
✓ Correct use of html elements to create a structure for the contents. ✓ Use of internal/external style sheet. ✓ Correct use of selectors to style different parts of the html document ✓ Appropriate use of properties to create **visual hierarchy**. ✓ Appropriate use of form elements for user inputs with **affordance**.	✓ Contents are visible and placed in appropriate location in the web page. ✓ Navigation from page to page ✓ **Consistency**.	✓ Any additional features that provide enhanced user experience. Example ✓ Interactive/Animated UI experience. ✓ Web app **responsive** to client device ✓ Use of an external CSS framework.
[25%]	[5%]	[10%]

Back End

Data management	Function modularity	Error handling	Functionality	Additional Features
✓ Correct folders and placement of files ✓ Correct way of referencing file paths ✓ Correct use of SQL statements ✓ Correct design of database ✓ 3NF schema	✓ Appropriate use of view functions. ✓ Appropriate use of helper functions.	✓ Able to recover from runtime errors and allow user to continue.	✓ User must be able to successfully perform the tasks as provided by the Web App.	✓ Able to host the web app on a hosting platform, example PythonAnywhere. ✓ Use of external web APIs to retrieve live data, e.g. Live weather information, ChatGPT response ✓ Security features like authentication and authorisation
[20%]	[10%]	[10%]	[10%]	[10%]