COMP3700: WEB APPLICATION DEVELOPMENT
SPRING 2025/ COURSE PROJECT
PHASE THREE: BOOTSTRAP, JAVASCRIPT AND DYNAMIC HTML – 40 POINTS
DUE DATE - SUNDAY 27TH APRIL 2022 @ 23:59

In this phase, your team is expected to enhance the website developed in earlier parts of the course. You will incorporate the **Bootstrap framework** to create a responsive, consistent, and visually appealing layout, and write **custom JavaScript** to add interactive and dynamic functionality.

1. **Bootstrap Integration:** You are required to remove all CSS rules and use Bootstrap exclusively for styling and layout. The final design should follow modern web development standards and offer a consistent look across different devices.
   a) **Grid System for Layout:** Structure all pages using a responsive layout based on Bootstrap's grid system
   b) **Replace All CSS Rules with Bootstrap Classes**: Remove all inline or external CSS styles and apply Bootstrap classes for all styling, layout, and spacing.
   c) **Form Styling:** Apply consistent Bootstrap styling to all form elements to enhance usability and presentation.
   d) **Form Validation:** Implement Bootstrap-based validation for all forms to give users clear feedback when submitting data.
   e) **Interactive Components:** Integrate at least two Bootstrap components that involve dynamic interaction to enrich the user experience.
   f) **Navigation Bar:** Create a responsive navigation bar that adapts to different screen sizes and improves usability.
   g) **Typography and Utilities:** Use Bootstrap classes to style text content and manage spacing and alignment consistently across the site.
2. **Dynamic Contents using JavaScript Arrays and functions:** Update the two Web pages that are used to display two XHTML tables in Part 2.
   a) Using the information in the XTML tables in part 2 of the project, define TWO object types with multiple properties using construction methods to be used to maintain the need information in your project.
   b) You need to maintain a collection of the information using arrays appropriate initialization using actual information.
   c) Use two empty table element (including header rows and needed format) to dynamically display information from the two arrays in point 1.
   d) Use TWO JavaScript functions dynamically display information from one of the two arrays to the corresponding XHTML element (dynamically adding rows to the tables).
   e) Use forms with appropriate buttons and elements to enable users to
      i Add an element to the arrays and updated the corresponding table using the two functions.
      ii Search for information from the two arrays.
3. Include a new page (Questionnaire page), which works as a Questionnaire form. The aim of this page is to collect feedback from your customers and validate user input. The added page should satisfy the following minimum requirements:
   a) The form should have a variety of controls (in addition to the submit button add at least 4 other controls; from at least 3 different types); textbox, radio-buttons, checkbox, text-area ….etc.
   b) The page should include validation for at least THREE controls, and the validation rules must be different and using advanced patterns. Do not use default password validation. Add full comments to your JavaScript code explaining how does it work.
   c) The form should be well-designed, nice-looking, have look-n-feel concepts.

4. Include a second new page (calculate Page), that use Java Scripting to perform some calculations; for example, a page that works as bill-calculator when the user purchases items from the website. The page should have the following minimum requirements:
    a. There must be a rationale of the page's content related to the website's objectives.
        i.e. concept of calculation is related to the topic of your website using clear formula.
    b. Use multiple mathematical operators; join them with some conditions (i.e., using relational and logical operators) depending on the user's interactions. For example; if you are designing a bill, you may decide to give 10% discount against some items within the bill for the elderly customers (those who are above 60-year-old). i.e. use case statement of if-else statement.
    c. Add full comments to your JavaScript code explaining ow does it work.
    d. The form should be well-designed, nice-looking, have look-n-feel concepts.

5. Include a third new page named "funpage.html", which allows the users to interact with your page and have fun with it when accessing your website (e.g., play a game such re-assembling a cut-photo; or do simple drawings; or gathering words to make English sentences from given a set of words); be creative. The page should have the following minimum requirements:
    a) Use of dynamic html concepts.
    b) Add full comments to your JavaScript code explaining how does it work.
    Do NOT copy script from other resources.
    c) The page should be well-designed, nice-looking, have look-n-feel concepts. The game should be related to the topic of the website.

6. Update your **index.html** page to include the followings:
    a) A text-banner at the bottom of the page, moving from left to right (or right to left) in the page; displaying a welcome message; such as "Welcome to the <company_name> website! Today is <current_date>, and the time is <current_time>"
    b) Add a photo-gallery, which displays a set of photos (at least 3) in rotational manner where each photo remains at least 3 seconds before it is exchanged automatically.
    c) Edit your home page to include a direct link to the three new pages in questions 1, 2, 3 described above (i.e.; Questionnaire page, calculate page, and fun page).

7. **Report and Submission:** Submit the following to Moodle before the due date:
    - A compressed folder that contains all pages and their related files/ pictures of your website.
    - Website must be well-designed, nice-looking, have look-n-feel concepts,
    - Add full comments to all JavaScript codes explaining what it does and how it works.
        ⇨ **Do NOT copy script from other resources.**
    - Report: Do not include the reports for the previous project parts.
        - Cover page includes team members and project title.
        - Includes screenshot of website pages that includes JavaScript along with a screenshot of fully commented JavaScript code.
        - Add a contribution table to specify the components that were performed by each team member with the percentage for each.
        - **You need to upload your website in the webhosting site.**

8. **Team Collaboration, GitHub Submission, and Report Update**
    1. **Teamwork:** All team members must contribute equally to the project. Clear collaboration should be demonstrated.
    2. **Task Allocation Table:** Include in your report a table describing the responsibilities assigned to each team member.
    3. **GitHub Repository with Versioning:** The project must be hosted on GitHub with evidence of versioning and collaborative development (commits by all members).
    4. **Updated Report Submission:** Submit a revised report including: Description of Bootstrap integration, Description of JavaScript features, Screenshots of new enhancements, and GitHub repository link

Grading Scheme:

#	Criteria	Description	Points
1	Grid System Layout	Use Bootstrap’s responsive grid system for page structure across devices	2
2	Replace Custom CSS	Remove all custom CSS and apply Bootstrap utility and layout classes	2
3	Form Styling	Use Bootstrap classes to style form inputs, labels, and buttons	1
4	Bootstrap Form Validation	Implement Bootstrap validation styles for form feedback	1
5	Interactive Bootstrap Components	Use at least two Bootstrap components like modals or carousels	2
6	Responsive Navigation Bar	Add a Bootstrap-based responsive navbar with links to all pages	1
7	Typography and Utilities	Style text, spacing, and layout using Bootstrap utility classes	1
8	Object Constructors	Define two object types using JavaScript constructor functions	2
9	Array Initialization	Create and initialize two arrays with sample object data	2
10	Dynamic Table Generation	Dynamically populate tables from arrays using JavaScript	2
11	Add New Data	Provide forms to add new data entries and update tables in real-time	2
12	Search Functionality	Implement a search/filter feature to update table views	2
13	Form Controls & Diversity	Create a feedback form with 5+ controls using at least 3 input types	2
14	JavaScript Form Validation	Apply and explain custom JS validation for at least 3 fields	2
15	Form Design	Design the form for usability and style it using Bootstrap	2
16	Use Case for Calculator	Create a calculator with a meaningful, realistic use case	2
17	JavaScript Logic & Conditions	Implement logic with mathematical operations and conditions	2
18	Code Comments & Layout	Fully comment the JS logic and style the page using Bootstrap	2
19	Scrolling Banner	Display a live welcome message with current date and time	2
20	Image Slideshow	Implement an auto-rotating image gallery (3+ images, 3s delay)	2
21	Navigation Links	Add links to Feedback and Calculator pages in the homepage navbar	1
22	Project Report	Submit a clear report with screenshots, descriptions, and code samples	1
23	GitHub Repository	Upload full project with clear version control and meaningful commit history	2
24	Team Contribution Table	Include a detailed task allocation table with % contributions	2
	Total		**40 Points**