PROG8681 Adv. JavaScript Programming  
Group Project 1

# Interactive Multi-Page Web Application

Create a custom website using the techniques of JavaScript and jQuery that you learned in this course.

## Details

Students will work in groups of 2 - 4 to create a multi-page (3+) web application choosing a theme from the supplied list of choices.

Choose a **main topic** from either from the **Games, Creativity & Interactive** list **OR** the **Productivity, Utility & Tools** list. Once you have your main topic, choose a topic from the **Addons** list to incorporate into your main topic.

\*\*NOTE: Only **two (2)** groups may choose the same main topic. Topic choices are first come, first served so it is recommended to include **two (2)** potential selections of **main** and **addon** topics. Topic and addon choice must be sent to the professor for approval by email before the next class in Week 7 or the date outlined in the Group Project 1 module instructions.

The application must demonstrate proficiency in the JavaScript concepts covered in the course up to Chapter 16. Consider the following requirements:

## Project Requirements

1. **Multi-Page Structure:** The application must have at least three (3+) interconnected HTML pages. Navigation between pages should be smooth and logical.
2. **Dynamic Content:** JavaScript should be used to dynamically update content on the pages. This can include:
   1. Manipulating the DOM (adding, removing, or changing elements).
   2. Updating text, images, and other content based on user interaction or data.
   3. Generating HTML elements programmatically.
3. **User Interaction:** The application must include interactive elements. Examples:
   1. Form input and validation (Chapter 10).
   2. Event handling (clicks, mouseovers, form submissions - Chapter 4, 8).
   3. Use of timers (Chapter 7).
   4. Basic animations or effects (Chapter 9).
4. **Data Handling:**
   1. Use of arrays and objects to store and manipulate data (Chapters 3, 4, 15, 16).
   2. Local Storage or Session Storage to persist data between sessions (Chapter 14). Consider a "save" feature.
5. **Code Quality:**
   1. Well-structured and commented code.
   2. Use of appropriate variable and function names.
   3. Error handling (Chapters 5, 13).
   4. Use of strict mode.
6. **jQuery Usage:** The project must incorporate jQuery for DOM manipulation, event handling, and/or effects (Chapters 8, 9, 10, 11).
7. **Create a JavaScript Library**: In addition to the main application code, create at least one separate JavaScript library file (.js). This library file should contain functions or objects that provide reusable functionality for your application. The library should be well-organized, include comments explaining how to use it, and handle potential errors (Chapter 16).

## Also to Consider

* JavaScript should be found in all pages of the website/app.
* Styles should be consistent across all pages.
  + Keep in mind font and colours.
  + Don't forget to consider readability!
* No inline styles or coding; must be in the external JS/CSS files.
* JavaScript, images, and CSS files must be in their corresponding folders (i.e., CSS files in a /css folder; JavaScript in a /js folder)
* Comments should include your name on pages that you helped create.
* Feel free to add enhancements as you'd like, have fun with it!
* Marks will be awarded for uniqueness and creativity.
* Marks will be deducted for JavaScript errors that are found in the Chrome Console and poor syntax.
* Incorporate a variety of coding techniques/concepts from the textbook to maximize your grades.
* Implementation of the techniques/concepts (does it make sense and provide functionality where used).

Remember that these projects can be adapted and expanded based on the skills and interests of your group members. It's important to plan and communicate effectively as a team, divide tasks, and collaborate to complete the project successfully. Also, reach out to your professor if you have any questions or concerns.

Once you have completed the work, please zip up all your project files and folders then one (1) group member submits them to the assignment folder. The file name should have this format:   
  
**GroupID\_GroupFinalProject.zip**.   
  
Where GroupID is the group ID assigned to your group.

## Grading Criteria

### The project will be graded on the following criteria:

#### Design Requirements:

* Single Homepage (index.html) per project.
* Styling and colour contrast. For colour contrast (Accessibility):
  + Ensure your foreground (text) colors contrast against your background colours.
  + Use the WebAIM color checker to verify your colors meet accessibility standards: <http://webaim.org/resources/contrastchecker/>
* Overall Appeal.
* The website displays the information appropriately.
* The site conveys a common message and common styling.
* The site is eye-catching, stylish, unique, and modern.

#### JavaScript Requirements:

* Utilize concepts found in Chapters 1 – 16. Grading will be based on how well these concepts have been implemented.
* How well is the code applied to the project (does the code make sense in the context of the project; does it have relevance to the project outcome).
* Your student name and number must be commented on each page your design/work on.
* Make sure to apply a variety of coding patterns to maximize your grade!!

## Rubrics:

For more information on the grading criteria, please review the project rubric. Rubrics can be found in the course shell from the top menu: **Course Tools -> Rubrics**

Click the drop down next to the rubric for the project and select *Preview*.

Let me know if you have any questions or concerns. Good luck and happy coding!