COMP2068 – JavaScript Frameworks
Assignment 1 – GROUP (or Individual)
Express Portfolio Site

Part 1 Due Friday Oct 30, 2020 @ 5 pm – Create private Repo, For Project GROUPS.

Part 2 Due Friday, Nov 13, 2020 @ 5 pm – Instructions below

Value 20%

Express Portfolio Site **Maximum Mark: 40**

Overview: Create your *Company Portfolio Website* using **ExpressJS** and implementing the **PUG templating engine**.

- **Link to a Video (MP4 or YouTube/Other) demonstrating all the items listed in the marking guide below. You MUST demonstrate to me each CRITERIA has been met, in your video. See Video Presentation Format, for how to present your video.**
- **There are free software tools for screen capturing your demonstration. I use OBS and edit with DaVinci Resolve 16. Both are Free**

Instructions:

This Express site must include the pages from your Company Portfolio 5 pages – your **Home page**, an **About** page, a **Projects page**, a **Services page**, and a **Contact** page.

1. Your Portfolio site will use **ExpressJS** and **NodeJS** and your web pages have been split to use different **View templates** and partials by implementing the **PUG templating engine** and Express Routes **(4 Marks: GUI, 4 Marks: Functionality):**
    a. You need minimum 1 view template (4 Marks: Functionality, 4 Marks: GUI).
    b. An Express Route must exist for each page of your site. **Note**: You will need to use the **app.get(path, callback(req, res, next))** method structure with a **res.render(view, locals)** method call to render each view (4 Marks: Functionality).
2. Your site will use the new structure created by the **Express Generator**. Your site files will be migrated to work within the **public**, **routes** and **views** folders **(20 Marks: Site Structure):**
    a. Generate your site structure with the Express Generator. **Note**: You must use the **-e** option to ensure that you implement the **PUG templating engine** for Express (3 Marks: Site Structure).
    b. Your **JavaScript**, **CSS and Multimedia Asset Files** should be moved to separate folders within the **public** folder. Using the Twitter Bootstrap CSS framework or a 3rd party theme (must reference the source) is strongly recommended. **Note:** the **public** folder is part of the path and does not have to be referenced (3 Marks: Site Structure).

c. You will define routes for all of your site pages in the **index.js** file in your **routes** folder (1 Mark: Site Structure).
d. Your **view**s folder will contain your **PUG page templates**. You will create a separate folder named **partials** for all of your partial **PUG files**. You will need at least 2 partials: **header.PUG** and **footer.PUG**. (6 Marks: Site Structure).
e. All Your Code (HTML, CSS, JavaScript, jQuery, etc.) is error free (1 Mark: Site Structure).
f. Your site uses appropriate **JavaScript** libraries that function through the use of links to a various **CDNs** (Content Delivery Networks) (1 Mark: Site Structure).

3. Include **Internal Documentation** for your site **(4 Marks: Internal Documentation):**
   a. Ensure you include a **comment header** for your **HTML**, **CSS and JavaScript files** that indicate: the **File name**, **Author's name**, **web site name,** and **file description** (2 Marks: Internal Documentation).
   b. Ensure you include a **section header** for all of your **HTML structure, CSS style sections,** and any **JavaScript functions** (1 Marks: Internal Documentation)
   c. Ensure all your code uses **contextual variable names** that help make the files human-readable (1 Marks: Internal Documentation).
   d. Ensure you include **inline comments** that describe your GUI Design and Functionality. **Note:** Please avoid "over-commenting" (1 Marks: Internal Documentation)
4. Share your files on **GitHub** to demonstrate Version Control Best Practices **(4 Marks: Version Control).**
   a. Your repository must include **your code** and be well structured (2 Marks: Version Control).
   b. Your repository must include **commits** that demonstrate the project being updated at different stages of development – each time a major change is implemented. ***Each member of a group MUST make commits*** (2 Marks: Version Control).

Optional Site Features (i.e. Potential Bonus Marks).

A. Use the **AngularJS** front-end framework to improve interactivity and to implement the MVC design pattern (4 Bonus Marks).
B. Create a working **contact form** on the Contact Me page using **node.js** and appropriate node packages (4 Bonus Marks).
C. Create an external document (MS Word or PDF) that includes page wireframes, an explanation of the technologies used, and a Style Guide that details visual elements such as fonts and colours (4 Bonus Marks).
D. Attach a full domain name to your site (e.g. http://www.TomM.ca) or host it via the cloud (Azure, Heroku, DigitalOcean). You get 1 free domain name for 1 year with your GitHub Student Pack account. (2 Bonus Marks).

SUBMITTING YOUR WORK

Your submission should include:

1. A zip archive of your website's Project files and a link to GitHub (preferable).
2. A link to your live portfolio site hosted with a Cloud provider (bonus)

Evaluation Method – I will mark your A1B from your Video from this Table below.

Video Presentation Format:

a. **Introduction –** Start by showing me your entire site, without adding new records to the table. I just want to see a nice overview of the entire site so that I can relate it back to your code.
b. **Rubric** – Demonstrate the items in the "Criteria" column of the rubric
c. **Length of Video** – You video should be no more than 5 minutes long but no less than 3 minutes.

NOTE: *This video is not meant as a tutorial explaining how to code. It is meant to demonstrate that you have coded the functionality already and are showing me what you have done!*

3.

Feature	Description	Marks
GUI / Interface Design	Display elements meet requirements. Appropriate spacing, graphics, color, and typography used.	4
Functionality	Site deliverables are me and site functions are met. No errors, including submission of user inputs.	4
Site Structure	Well organized site files. Separate HTML and CSS. Appropriate links to external documents and code. Code is error free. JavaScript libraries use a CDN. 4 marks	20
Internal Documentation	File header present, including site & student name & description. Functions and classes include headers describing functionality & scope. Inline comments and descriptive variable names included.	4
Version Control	GitHub commit history demonstrating regular updates. 2 marks for 4 commits by	4
Total		**40**

This assignment is weighted **20%** of your total mark for this course.
Late submissions: **After Monday 16 Nov 2020, 5 pm, you will receive a grade of zero (0).**
External code (e.g. from the internet or other sources) can be used for student submissions within the following parameters:

1. The code source (i.e. where you got the code and who wrote it) must be cited in your internal documentation.
2. It encompasses a maximum of 10% of your code (any more will be considered cheating).
3. You must understand any code you use and include documentation (comments) around the code that explains its function.
4. You must get written approval from me via email.