WEB322 Assignment 4

Assessment Weight:

9% of your final course Grade

Objective:

Build upon the foundation established in Assignment 3 by providing new routes / views to support modern CSS techniques, html and publishing to a Cloud.

NOTE: You may work on Assignment 3 if you wish (not necessary though), to apply the requirements for this assignment.

Specification:

Part 1: Creating a "theme.css" file within a "public" folder (public folder is located in the same level as server.js file).

Step 1: Creating a "theme.css" file and linking to it from our HTML files.

- Within the root of your application, create a new folder called "public"
- Within the "public" folder, create a "css" folder (this is where we will place our CSS file(s)
- Finally, within the "css" folder, create a "theme.css" file.
- Now that we actually have a "theme.css" file (albeit an empty one), we must include it in all of our "views" (.html files):

  This will involve using the appropriate <link> element in the <head> of your files .
- With this complete, we must use the built in express "static" middleware within our server.js file to identify our newly created "public" folder as a source for static files.

Step 2: Updating "theme.css" to provide a unique look / feel for your application

Now that we have a "theme.css" file and its correctly linked in our html files, we can start to think about personalizing our web app by adding some CSS. There are plenty of resources online to help you pick colours and find (as well as generate) interesting styles to apply to selected elements. Some quality resources to get you started include:

- This "Colour Wheel", used to pick complimentary colours and get their "hex" values: https://www.canva.com/colors/color-wheel/
- A "Box Shadow" Generator, used to provide the complete CSS for adding a "box shadow" to an element: https://www.cssmatic.com/box-shadow
- Sample CSS for generating a "full page gradient" (**Note:** for this example to work for us, we must change the selector from "html" to "body, html" and instead of "#red" and "#blue", simply use "red" and "blue" – or whatever other colours you like): https://coderwall.com/p/ape0jg/full-page-gradient-background

This storefront.html page must have 2 applications – (1) use of Bootstrap and (2) the page has to be mobile responsive. You are allowed to include/embed css in the html page. The theme.css should have some common applications on this storefronmt.html.

If you did not complete assignment 3, then you may create/add other html pages of your choice to make the website a multipage web application. Remember the pages should have consistent look for user friendliness such as navigation, header, footer etc. – these elements should be consistently designed in all pages. The common styles need to be defined in theme.css.

Adding "Get" routes in server.js

- Adding body-parser: Add the express.urlencoded({ extended: true }) middleware (using app.use())
- Inside your server.js file, add routes for any html file(s) created such as the route "/storefront", which will simply send the newly created "storefront.html" etc.
- You must have a GET/ route defined for the home.html or to start the web application. All pages must have consistent navigation option to all other pages/routes.

Part 3: Pushing to Cyclic

Once you are satisfied with your application, deploy it to Cyclic:

- Ensure that you have checked in your latest code using **git** (from within Visual Studio Code)
- Open the integrated terminal in Visual Studio Code
- **NOTE:** If you have decided to create a new Cyclic application for this assignment, you can follow the "Cyclic Guide" on the course website.

Assignment Submission

- Add the following declaration at the top of your server.js file:

```
/*********************************************************************************
* WEB322 – Assignment 06
* I declare that this assignment is my own work in accordance with Seneca  Academic Policy.  No part of this
* assignment has been copied manually or electronically from any other source (including web sites) or
* distributed to other students.
*
* Name: _____ Student ID: _____ Date: _____
*
* Online (Cyclic) Link: _____
*
********************************************************************************/
```

- Compress (.zip) your assignment folder (optionally omitting node_modules, to make the upload faster) and submit the .zip file to My.Seneca under **Assignments** -> **Assignment 6**
- **Type your working Cyclic link in the comment box when you upload in blackboard. Failure to do so will result in zero mark for this assignment.**

Important Note:

- **NO LATE SUBMISSIONS** for assignments. Late assignment submissions will not be accepted and will receive a **grade of zero (0)**.
- After the end (11:30PM) of the due date, the assignment submission link on My.Seneca will no longer be available.
- Submitted assignments must run locally, ie: start up errors causing the assignment/app to fail on startup will result in a **grade of zero (0)** for the assignment.