ePortfolio Part 1

This assignment relates to the following Course Learning Requirements

- **CLR 5**: Develop Ajax-Enabled interactive web (Web 2.0) Applications

Objective of this Assignment

- Showcase your skills and demonstrate your understanding of the web programming technologies we have covered in the course. The project consists of three parts, each focusing on different technologies covered.

Pre-Assignment Instructions

- Read modules 1 through 10 and use the modules, practice exercises and graded assignments in the course to understand what has been covered.

Tasks

- Create a website for your portfolio that showcases the following criterias:
    - Create a webserver in either PHP or ASP.NET that serves webpages and documents.
    - That webserver needs to serve at least 1 document in the format of JSON or XML. JSON is recommended because it's more common and easier.
    - The webpage needs to acquire the JSON/XML document **asynchronously** using the library of your choice and process it locally. The javascript function `fetch` is a good starter choice.
    - Manipulate the web page using jQuery.
- Create a report that provides the following:
    - A description of the site.
    - The web technologies from above that you used and why.
    - Screenshots of your website.
    - What you want to accomplish with the site development (i.e. create a website that shows baseball statistics or create a site that keeps track of ocean clean-up operations), why you chose this, and how it is supposed to work.
    - Short descriptions of the struggles you encountered.

The goal of the report is to have a backup explanation in case your facilitator can't access or display your website for one reason or another. It also adds context to your work.

Submit your source code and your report. If you're using webpack or equivalent you can include your build IN ADDITION to your source code. Please avoid uploading your `node_modules`.

Example idea

You DO NOT have to do this project but you CAN. It is only mentionned here to explain the structure of the assignment.

This section describes a website that can be used to validate if a phone number is valid.

Create a webserver that has 2 endpoint.

- `GET /index.html`
  This endpoint serves a regular HTML webpage.
  That webpage contains:
  - A title "Phone validation service"
  - A text field with placeholder "phone number"
  - A button labeled "validate"
  - An explanation that the user should enter a phone number and press the button to validate it.
- `POST /validate` This endpoint takes in as `POST` body the JSON document below:

```
{
  "phone": string
}
```

The webserver then verifies that the given string matches the regex `\d{3}-\d{4}` and sends back the following JSON:

```
{
  "valid": boolean
}
```

When the validation button is pressed, an asynchronous request is made to the servers `/validation` endpoint using the javascript `fetch` function, including the appropriate JSON body. Upon reception of the response the JSON is parsed and the website uses the `alert` function to display wether the phone number was valid or not.

Other potential ideas

If you're stuck for ideas feel free to take inspiration or implement one of the following projects:

1. Create a website that keeps a movie list. Users can add movies that they have watched and give them a score 0-5.
2. Create a website that keeps track of kinds of fishes caught. You'd have to make a list of fish. Store it in a database. Make a webpage that displays the full list of all fishes (acquire the list of fish asynchronously). Then when clicking a fish will toggle between greyed out and not greyed out, saving the status in the database. Feel free to take fish inspiration from well known video games.
3. Create a website that gets a random pun from a pre-defined list.

If you have an idea but aren't sure if it's fits the criterions or if it's feasable within the time given. Reach out to your facilitator.

Example of async request

If you need a live example of an async request please see this github repository. Note that this is only the client side. You need to implement the server.

Submit the following files:

1. `CST8259_ePortfolio1_src_USERNAME.zip`
    - Your valid code, properly formatted.
2. `CST8259_ePortfolio1_USERNAME.pdf`
    - PDF format
    - Ensure that your assignment has the title at the top and the sections of your report have headings and subheadings to chunk your paper into sections for each of the topics you are writing about.
    - Recommended font is Arial 12 or Calibri 12. Text is recommended to be double spaced but any table may be single spaced.
    - You must cite all your sources of information using IEEE or APA formatting.
    - Create a references page that lists all of your sources that you have cited in text. Sources include software used. Also, personal communications from a professional in the field count as a reference source. For citation and referencing examples see this link.

Section	Criteria	Points
Technology Descriptions	- Descriptions are complete, accurate and clear.	2
Demonstration of use of technology	- High quality demonstration of technology that includes key relevant aspects of its use as covered in the course content. - Highlights key functionalities pertaining to data as covered in the course content.	5
Presentation	- Presents the portfolio as a neat and well-organized web page or web site.	2
Formatting	- All report information follows the formatting requirements listed in the instructions. - All use of references are in IEEE or APA format.	1