PROG2001 – WEB DESIGN AND DEVELOPMENT

A-06 : JQUERY AND JSON BASED TEXT EDITOR

OVERVIEW

In this assignment, you will be investigating and using both jQuery (as the front-end to your AJAX calls) and JSON (as a data interchange format). The idea behind this assignment is to get you to create an *online text editor*.

This is a partner-based assignment and you can have partner if you wish. You do have to choose a partner from the same "Specialized La" section as you however. You can find the list of other students in your "Specialized Lab" section within eConestoga by looking in **Groups** under *Course Tools*....

OBJECTIVES

This assignment supports the following course objectives:

- Design and implement dynamic and interactive pages using jQuery and JSON

ACADEMIC INTEGRITY AND LATE PENALTIES

- Please refer to the SET Policies document regarding Academic Integrity Information
- Please refer to the SET Policies document regarding Late Policy

EVALUATION

- Please refer to the assignment weighting in the *Instructional Plan* for the course as well as the assignment's Rubric in the course shell.

PREPARATION

Review Module-11 lesson content as well the module's code samples – they will help you in this assignment.

1. You may use either ASP, PHP or ASP.NET as the server-side technology in the creation of this online editor.

2. The client page must allow you to choose a text file from a drop-down list (an HTML *select* element), have the online editor open the file, read its contents, and populate an editable area on the page
   - The list of allowable files are to be populated from a directory called `MyFiles` that exists on the server (you can assume that this directory exists as a sub-directory within your solution's web site – not necessarily in the root of the localhost web space)
   - Keep in mind that the file names are not limited (they can include spaces in the name)
   - Also keep in mind that you cannot assume that the file extension on the files will be `.txt` – but you can assume that all of the files in the directory will be ASCII (since they are text files)
   - if you are not using ASP.NET – name your first page startPage.html or `startPage.php` or `startPage.asp`. If you are using ASP.NET, then call it `startPage.aspx`

3. The user must be able to make simple text edits in the text edit area

4. Your text editor does NOT need to support any formatting requirements
   - It just captures text (like Notepad) – no need to allow bold, italics, etc.
   - You do NOT have to support cut/copy/paste programmatically within the editor

5. The user must be able to save the text to the same file, or select a new filename to save it to (i.e. different filename) on the server
   - Again the directory that the file is to be saved to is a directory called `MyFiles` on the server
   - You do NOT have to support multiple directories on the server.

6. You must use **jQuery** to make an AJAX call to populate the file list on the client.

7. You must use **jQuery** to make an AJAX call to populate the editable text area on the web page (i.e. with the file's contents).

8. The data interchange between the server and the client **must** be in JSON format.
   - This means that sending the file's contents from the server to the client must be in JSON
   - This also means that sending the (potentially) updated file contents from the client to the server must also be in JSON
   - This also means that the list of editable files (sent from the server to the client) must be in JSON

9. Make sure **to concentrate on the user experience** (i.e. the design and flow of the editor application)
   - This includes providing users with feedback as to what is happening within the application
   - Making the application pleasing to the eye and professional looking
   - Think like the user – what is the purpose of using your editor? Streamline their workflow – think about usability

1. Ensure that all of the necessary and required files are included in the **single** submission ZIP file
   a. Submit a directory structure that includes any and all web pages, CSS files, and any other supporting files that are necessary for your editor.
   b. If you have developed your solution in ASP.NET – then include **your entire cleaned VS solution**.
   c. You do not need to submit your test files … I have my own … ☺
2. Make sure you comment your source code (whether that be a web page or C# source file (in ASP.NET) appropriately, especially the header comments
3. Also remember that this solution will be tested using Microsoft Edge as well as Chrome
4. Please ZIP up these files and submit to the appropriate eConestoga Dropbox by the deadline
   a. Please give your ZIP submission the filename *lastName-firstInitial.zip* (e.g. if you are Sally Jones – then your ZIP should be named `jones-s.zip`
   b. If you are working with a partner, then include both your names in the ZIP filename (e.g. if Sally Jones is working with John Smith – then your ZIP should be named `jones-s-smith-j.zip`
      ==NOTE: If working with a partner, only one partner need submit the solution==

When submitting your solution to this assignment, hand-in a single ZIP'd file containing:
1. Your cleaned Visual Studio solution.
   a. <u>Do not include your own test files</u> – a set of pre-built test files will be used in the marking process.
2. Also remember that this solution will be tested using Internet Explorer v11 as well as Chrome
3. Please ZIP up these files and submit to the appropriate eConestoga Dropbox by the deadline
   a. Please give your ZIP submission the filename *lastName-firstInitial.zip* (e.g. if you are Sally Jones – then your ZIP should be named `jones-s.zip`
   b. If you are working with a partner, then include both your names in the ZIP filename (e.g. if Sally Jones is working with John Smith – then your ZIP should be named `jones-s-smith-j.zip`   Remember if you are working with a partner it **must be** someone from your own *Specialized Lab* section.

      ==NOTE: If working with a partner, only one partner need submit the solution==