CSE 5335 Project 1 - Spring 2016

Overview

This is the first of 2 programming projects for CSE 5335. The purpose of this project is to establish the client-server components of a state-of-the-art web application. The project will be due on March 8th at 11:50PM.

Objectives

To successfully met the objectives of Project 1, your submission must include the following:

1. You will choose a server-side web application framework that will be able to (a) host a web page (b) on which you will include Javascript to dynamically modify the contents of the page (c) using data queried from a RESTful web service API. You may use any technology of your choice, but it must not be subject to any license fees. However, I would recommend your choosing either Ruby-on-Rails (http://rubyonrails.org/) or NodeJS (http://nodejs.org/), both Open Source frameworks that have large, active development communities and are being used at many of the most progressive web-based companies around the world. **(10 Points)**
2. You will choose a client-side (browser) Javascript framework capable of providing helpful classes and functions for manipulating the DOM and performing AJAX-style API calls to RESTful web services that will be available from your server-side framework. Again, you may use any Javascript framework of your choice that does not require any license fees. Similarly, I would recommend your choosing either jQuery (http://jquery.com/), AngularJS (https://angularjs.org/), or React JS (https://facebook.github.io/react/), each has large, active Open Source development communities, etc. **(10 Points)**
3. You will create a single web page that is displayed as the default page for your web application that will display a button. When an end-user clicks the button, Javascript on the page will call a RESTful web service API hosted by your web application that will return a collection of JSON data with at least 10 objects, then construct either an HTML table, list, or set of nested DIV and SPAN objects containing the JSON data you received. **(50 Points)** For extra credit, you can also include latitude/longitude values in your JSON data and display dots on a Google Map (https://developers.google.com/maps/) on the page **(10 Points)** and/or include some numeric values in your JSON data and display them in a chart (e.g., https://developers.google.com/chart/, http://d3js.org/, http://www.chartjs.org/) on the page **(10 Points)**.
4. You will deploy your website on a Heroku server (http://heroku.com) with the server name of "CSE5335-<your-netID>" and make our class GTA me collaborators so that we can review your site (emmons@uta.edu). NOTE: Do not make changes after the deadline or your project will be considered late. (**10 Points**)

5. You will include a README.md file (note that this file extension implies the following syntax available for formatting -- http://en.wikipedia.org/wiki/Markdown) as part of your server installation answering the following questions:
   a. What server framework did you choose and why?
   b. What client framework did you choose and why?
   c. What aspect of the implementation did you find easy, if any, and why?
   d. What aspect of the implementation did you find hard, if any, and why?
   e. What components OTHER than your client and server framework did you install, if any, and if so, what is their purpose for your solution?
   f. What Ubuntu commands are required to deploy and run your server?

   (20 Points)

Submitting Your Project

Your code should be saved in a private GitHub repository named “cse5335-project-1”. And you should add the class GTA and me (emmons-uta) as collaborators.

Additional Recommendations

As a general practice for modern software development, I strongly recommend following some kind of automated testing scheme to verify the correctness of your code (e.g., http://en.wikipedia.org/wiki/Test-driven_development). Ruby-on-Rails has a strong tradition of using various testing frameworks along with the development process (http://guides.rubyonrails.org/testing.html), and NodeJS likewise has similar tools (http://unitjs.com/).

In the Ruby world, you will want to familiarize yourself with RVM (https://rvm.io/) and bundler (http://bundler.io/) to manage access to 3rd party libraries called “gems”.

In the Javascript world, you will want to familiarize yourself with NPM (https://www.npmjs.com/) and bower (http://bower.io/) for managing addons and 3rd-party libraries (e.g., jQuery and Angular are 3rd-party libraries).

Note Concerning Collaboration

The active Open Source communities for web development mentioned above all heavily leverage the knowledge and experience of it members to get things done. I have no problem with students collaborating to help each other figure out techniques and strategies for implementing the project deliverables, but the size of this project is such that each student should be able to implement the steps individually. I will be performing random tests for uniqueness on the contents of your project upload files. Duplicate submissions will receive zero points.