Assignment 3
Web Service
Due Date: October 23, 2015 11.59PM

In this assignment, you will develop a set of REST APIs capable of reading a JSON file deployed on a server and returning information by using a collection of simple HTTP requests. You will also build a single HTML web page in which the requests to the APIs will be invoked and the retrieved information will be displayed.

Overview

Companies like Twitter have strong RESTful APIs that allows users to explore the wealth of data they manage and even provide ways to download your entire archive of tweets. Many users rely on Twitter's favorites as a mechanism to bookmark tweets that link to interesting content elsewhere on the web. However, in order to avoid a potential downfall in the future, where those favorites could one day become unavailable, you've decided to retrieve a list with your favorite tweets as a JSON object. Here is a sample JSON file.

Specification

Your RESTful APIs should be able to respond, with the appropriate data, to the following requests:

- Get all tweets available in the archive.
- Get all known Twitter users included in the archive.
- Get a list of all external links included in tweets from the archive.
- Get the details about a given tweet (given the tweet's id).
- Get detailed profile information about a given Twitter user (given the user's screen name).
- At least one request to showcase your ability to extract interesting information from the archive.

Requirements

1. The system will consist of two main components: the front end, which should be implemented as ONE web page called index.html and the back end, which should be implemented as a Node.js server implementing the REST APIs.
2. The web page should display any usage instructions, as well as links or buttons that trigger the proper HTTP Requests on your server and then display the retrieved data.
3. You are allowed to use the basic jQuery library (no PlugIn) for the web page. Use Ajax to only update some part of you page, as opposed to refresh the full page.

4. All the operations need to be performed by the server and can only be requested via your own REST API. This means that aside from loading the initial index.html page, there should be no other pages served by the server.
5. The Node.js server does NOT require a database. For the purpose of this assignment it is ok to lose all data when the server is stopped.
6. You are not allowed to use external node modules (e.g. Express). Concretely, no *npm install* should be required prior to running your Node.js server.
7. Your REST API will read data from the JSON file provided, which contains a number of tweets. You are not going to pull data from the actual Twitter API.
8. The frontend must be accessible using the root URL. Your Node.js server should use port 3000 on local machine, we will only test http://127.0.0.1:3000/

Deliverables

Your assignment should be submitted as a zip file on MarkUs:

- `a3.zip`:
  - The index.html file along with any CSS files.
  - JavaScript file(s) included in the index.html page and for your NodeJS server.
  - The jQuery libraries required by your project.
  - A instructional README file with a list of HTTP Requests expected to be used to test your server.
- Do NOT include the Node.js framework or any module (no module is allowed, anyway!).
- The sample JSON file provided is just for testing purposes. When evaluating the assignment we will use a different version.
- Finally, all deliverables should be neatly formatted, readable, and be properly documented (aka. have good code comments).

Marking Scheme

Proper design of your REST apis.

Updating DOM without refreshing the entire page

Does it WORK ?

Clear instructions on how to use it in README file

Well-documented code?