CSE 5335 Project Phase 2

Overview

This is the second of 3 phases of a programming project for CSE 5335.002. The purpose of this phase is a hands-on comparison of “SQL” and “NOSQL” data sources. You will leverage the base technologies used in Phase 1 on the server-side to create 2 simple scripts that will each perform 2 basic types of data queries on a SQL and NOSQL data source.

Objectives

To successfully met the objectives of Phase 2, your submission must include the following:

- You will again use your VirtualBox (https://www.virtualbox.org/) virtual machine with Ubuntu Server as the operating system. **(10 Points)**

- You will choose (a) one SQL and (b) one NOSQL data source to use in your project, install them. You may use any choice that may be installed on Ubuntu and has no license fees associated with it. There are many popular choices for both SQL and NOSQL available. If you have already chosen Ruby-on-Rails for your server-side web-application framework in Phase 1, you already have SQLite (https://sqlite.org/) available to use. Other good SQL alternatives are MySQL (http://www.mysql.com/) and Postgres (http://www.postgresql.org/). For NOSQL, there are many options which we will also be discussing in class. **(10 Points)**

  Good choices for NOSQL include:
  - Cassandra - http://cassandra.apache.org/
  - Couchbase - http://www.couchbase.com/
  - MongoDB - http://www.mongodb.com/
  - Redis - http://redis.io/

- You will choose a server-side web application framework that will be able to perform queries on both the SQL and NOSQL database. You may use any technology of your choice, but it must not be subject to any license fees. You may continue to use your choice from Phase 1 or make a different choice. **(10 Points)**

- You will select a web service data source that will return either JSON or XML data. Using your server-side web application framework, load identical data into your SQL and NOSQL database choices from your selected web service -- load at least 100 records into your databases. You will need to configure your SQL and NOSQL databases to receive the data that you retrieve from your chosen web service. For example, in the case of SQL you will need to create one or more database tables to store the data. **(10 Points for SQL, 10 Points for NOSQL)**

EXTRA CREDIT: You can get extra credit for adding more than 100 records into your databases: **10 Points for 1000 records**, **10 Points (additional) for 10,000 records**.

Example web services are as follows:

- http://www.omdbapi.com/
- http://www.data.gov/developers/apis
- http://www.census.gov/data/developers/data-sets.html
- http://aws.amazon.com/datasets

- You will create 2 scripts using your web application framework technology. You will not be running the server application, only using the code libraries to create scripts that are executed on the command line in the virtual machine. One script will use your choice of SQL database to perform 2 queries and another script will use your choice of NOSQL database to perform the same 2 logical queries. The queries are as follows:
  - Select a single record from the data source using its primary key and echo the data to the console. **(5 Points for SQL, 5 Points for NOSQL)**
  - Select a subset of records (at least 2) using an attribute of the data set that is NOT its primary key and echo the data to the console with one line per record from the result set. **(10 Points for SQL, 10 Points for NOSQL)**

- 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:
  - What SQL framework did you choose and why?
  - What NOSQL framework did you choose and why?
  - What server framework did you choose and why?
  - What aspect of the implementation did you find easy, if any, and why?
  - What aspect of the implementation did you find hard, if any, and why?
  - What components OTHER than your SQL, NOSQL, and server framework did you install, if any, and if so, what is their purpose for your solution?
  - What Ubuntu commands are required to deploy and run your server?

  (20 Points)

Submitting Your Project

Your implementation will consist of several files in a directory structure. You will need to create a ZIP (or GZ) file of this entire directory structure and upload it to Blackboard by 11:30PM CST on Wednesday, April 1. As listed in Objective #5, your README.md file should make it clear what steps are required to run your web application in a VirtualBox with Ubuntu.

Additional Recommendations

For your own security, I recommend that you use a version control system for capturing ongoing changes to your code, along with frequent external backups. My favorite is Git (http://git-scm.com/), but Subversion is also popular (https://subversion.apache.org/).

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.