Capstone Project 1.5

Due:

March 14th at 11:59pm [don't let the apparent long time fool you, you have spring break and your exam in this period so even though this should (if you did a solid job on part 1) be not too rough of a project, don't wait]

Summary:

You will get the chance to continue your odyssey in investigating apis, reading documentation, writing a program and writing tests for the program by extending your program. Hopefully you will get a more complete feel for the value of automated tests.

Details:

In the first part of this project you scraped some data from stack overflow jobs for nearby jobs. But really, no one is interested in **all** of the nearby jobs. So I'm going to provide a few user stories to help you expand the project (and then you can look at the Nitty-Gritty section below for more:

User stories to be supported

- As a job seeker, I want to see only the local area jobs that are in my area of interest
- As a professor, I want to see where the local jobs in a particular area of interest are located.

So you need to let the users enter a keyword or two and filter the results using that keyword.

You need to put up a map (MA and RI are the only places that you need to show, everything else is more than 50 miles away) and somehow show where these jobs are, be it a heat map, map pins or whatever. Be sure your mapping indicates how many job listings are in this location in some way. Stackoverflow is really good at forcing their advertisers to include a city for the job so you should have a *fairly* easy time with this.

Nitty-gritty:

Extend your existing program.

- Java (target whatever version is on the university computers)
- or python (you can target any version of python 3.5+)
- write it to be cross platform on desktop machines
- use one of the jetbrains IDEs to do your work. We have them installed here at the university and

you can get the full professional suite of tools for free as students so long as you are only using them for student work
- Some hints to take a look at (you can of course use other approaches so long as they work):
- Java users: Google's Monigeo package looks easy to use.
  - for lat/lon try this tutorial
- Python users might use maplotlib should look at Basemap maybe look at this tutorial
  - for lat/lon have a look at geopy - the version in pypi works painlessly even if the version on github currently has a failing build.
- Write additional tests for your new project. As a hint for those of you new to automated testing, begin investigating the API you will use for this project, by writing tests for the API

Submission:

Zip your entire project directory and submit the whole thing on blackboard.