

## Interface (with interface) Solution:

This solution provides answers to the requirements in Part 3 and Part 4

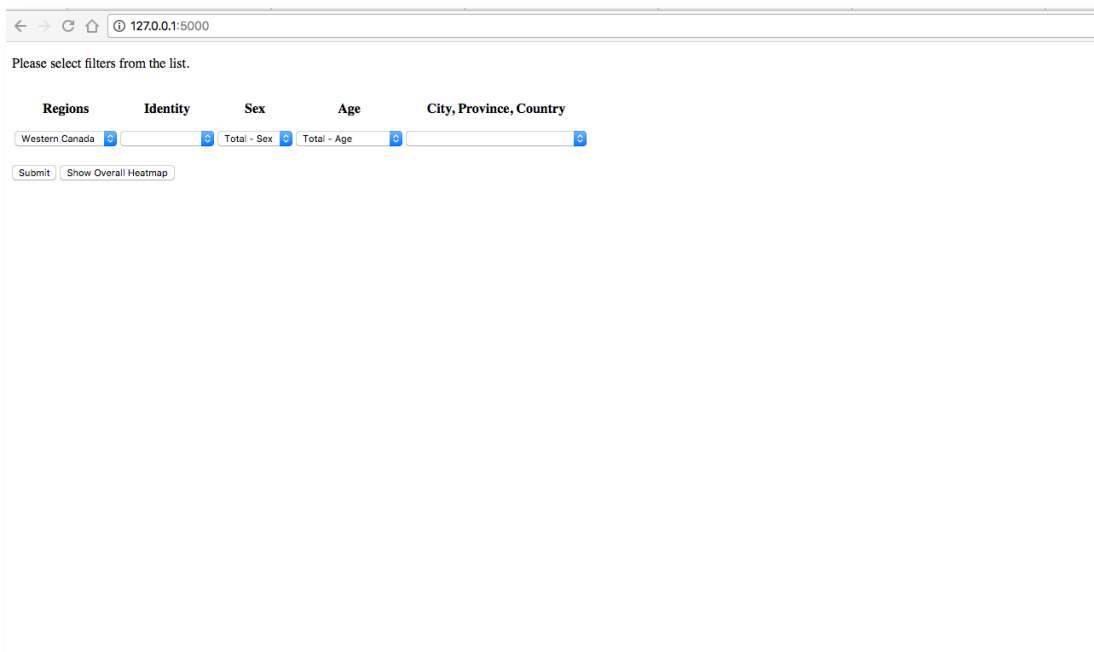
The solution is based on the base solution. Also, it uses flask, matplotlib, folium, io, os to generate front-end experience and drawing plots and heat maps. The “server.py” provides a local server at “127.0.0.1:5000” connection from a browser for users to interact with on a very simple webpage. Results of the bar chart will then be demonstrated at the lower part of the page as a connected graph. Software such as Excel and Tableau can also provide interface experience to the user depending on preferences.

## Assumptions:

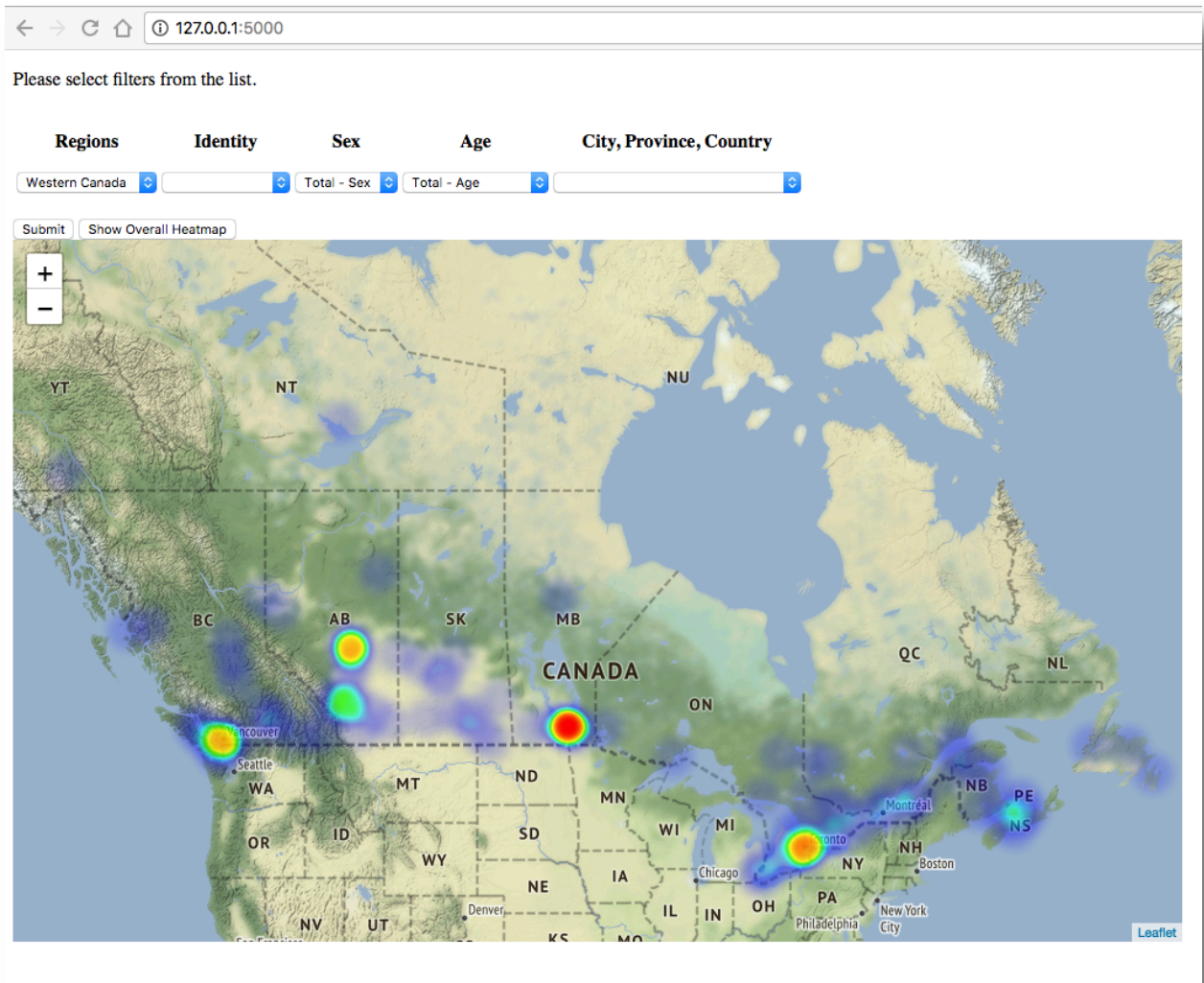
- this has inherited assumptions from base solutions
- assuming webpage is the interface mentioned in the requirements
- if filters of regions and cities both existed, then cities will override regions
- cities filter is the listed geographical dimension in the provided dataset, disregard the geographical level (can be modified to geographical level 2 and above if needed)
- heatmap geographical data is based on a free open resource

## How to run:

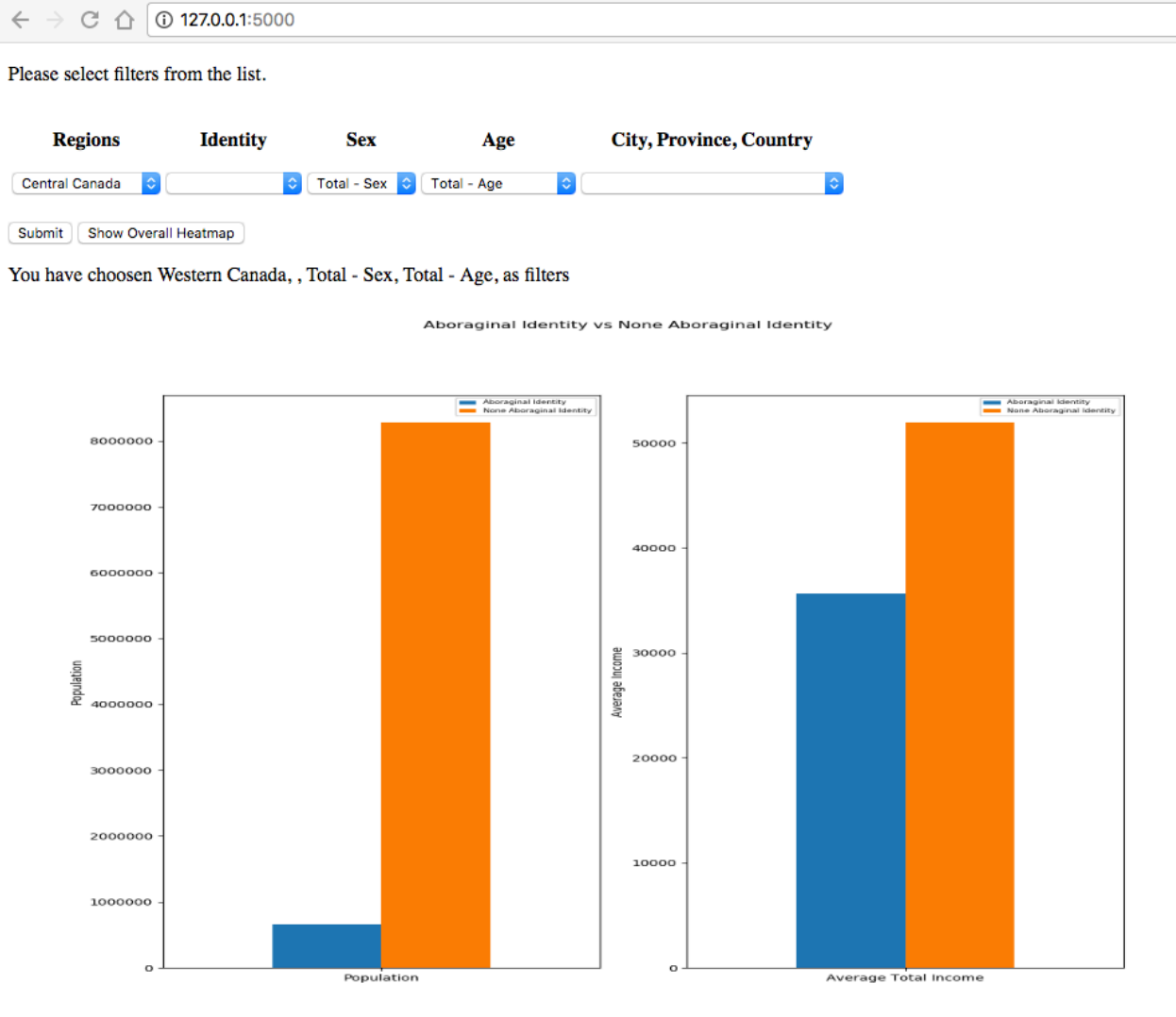
To use, run the “server.py”, open a web-browser and put “127.0.0.1:5000” in the address. Depending on the browser, the experience may be different. It has been tested on Chrome browser on a MacOS machine. The following page will show up initially:



By clicking on the “Show Overall Heatmap” button, a heat map will be displayed at the bottom of the screen. The heatmap is generated based on the aboriginal population by Canadian geography (provided by the free source)



Select filters at the top and then click on submit, two bar charts will be generated representing the population and average total income



Please view code to see detailed explanation of functions