EK 125 Final Project: Population Distribution

Names:

Elisa Cordeiro Lopes

Shreyas Ahuja

Roshen Venkatesan

Folder Name: Elisa\_Roshan\_Shreyas

I, Elisa Cordeiro Lopes, state that the program in this project is my original work.

I, Shreyas Ahuja, state that the program in this project is my original work.

I, Roshan Venkatesan, state that the program in this project is my original work.

***“I have a feeling in a few years people are going to be doing what they always do when the economy tanks. They will be blaming immigrants and poor people”***

***-Mark Baum, The Big Short***

Population Distribution is among the most prominent challenges in our world today. People migrate from rural areas to urban areas in search of employment and a better standard of living. Since its inception, America has been a country with among the highest levels of social mobility. The American Dream was first achieved by Benjamin Franklin who moved to Philadelphia from Boston in search of better job opportunities. However, the population back then was not as dense as it is today. As population rises, we need to ensure the proper distribution of people amongst metropolitan areas and hubs. Otherwise, negative consequences such as uneven spread of resources, job opportunities, skewed economic development and income inequality will rise. We believe in guaranteeing a better standard of living for every citizen. We believe every civilian should be properly informed before migrating to a different location.

Our program achieves exactly that in a very concise manner. It considers various factors in such as Chronic Disease Rates, Unemployment Rates, and income inequality (Gini Coefficient) across a vast majority of cities. We then implement a mathematical algorithm that assigns a unique number to every city called the “migration coefficient”. We use the same algorithm to derive a single number known as the “optimum migration coefficient” which is used as a standard and benchmark with which we compare every city to. Ideally, we strive to achieve a world where every city is as close to the optimum coefficient as possible.

Our program prompts the user for the desired city he or she wishes to move to. We process the input and return feedback on whether the inputted city is a practical place for the user to migrate to based off our coefficient values. If the city is deemed impractical, we offer alternatives to the user for them to consider. This program is a small prototype for the fully-fledged application. We have only implemented 3 large datasets in our coefficient, but we need more large datasets such as the population density, traffic, median income, crime rates etc. Due to the time constraint on the project and impending deadline, we were able to fully process only 3 of the many necessary factors in our migration coefficient.

The links to the data sets:

<https://www.bls.gov/web/laus/laumstrk.htm> (Unemployment ranking for states)

<https://www.bls.gov/web/metro/laummtrk.htm> (Unemployment ranking for cities)

<http://www.towncharts.com/Top-500-Cities-in-the-US-for-The-Gini-Index.html> (Gini Coefficient)

<http://www.governing.com/gov-data/population-density-land-area-cities-map.html> (Pop. Density)

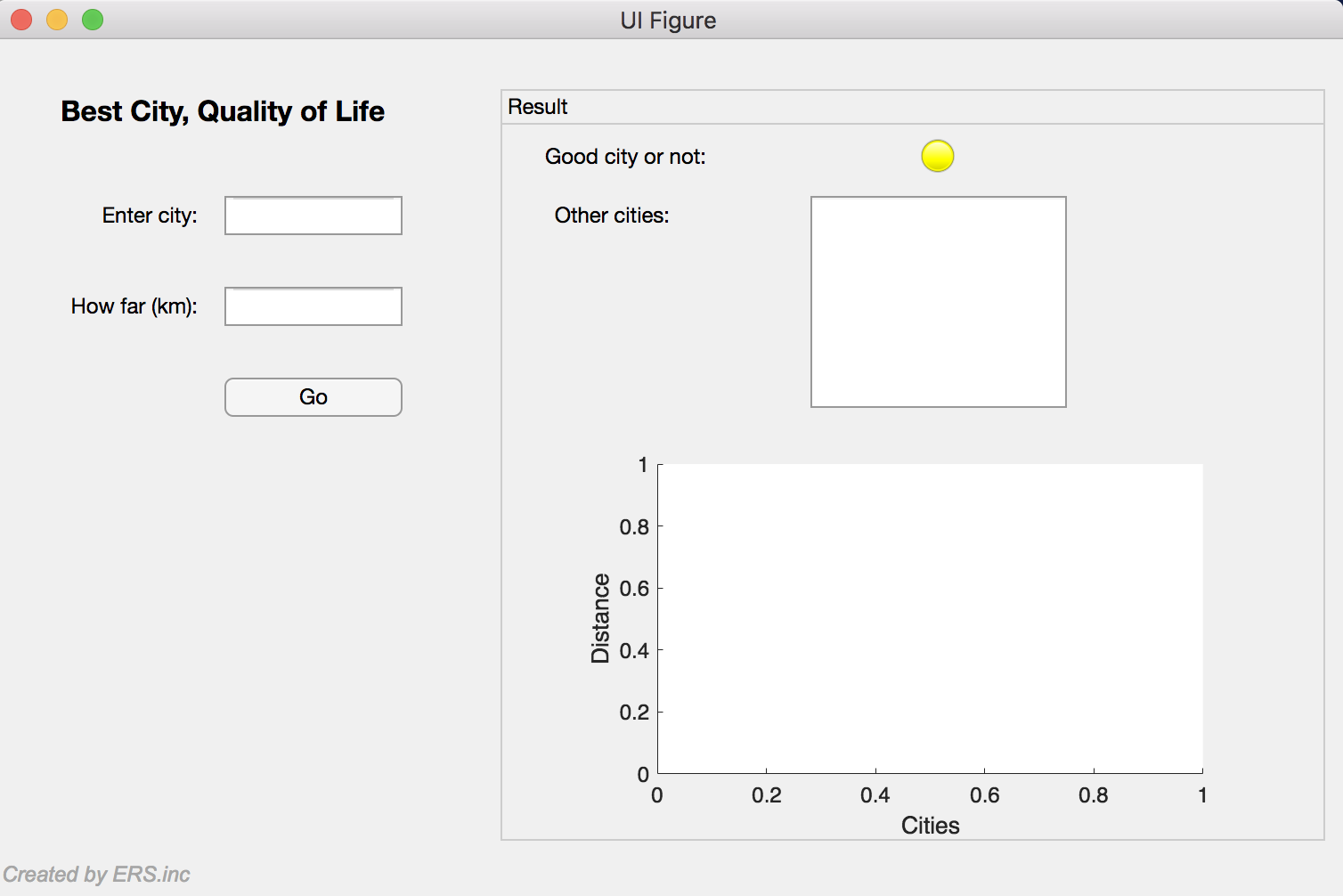
<https://www.artscipub.com/info/latlonofmajorcities.asp> (City locations)

<https://catalog.data.gov/dataset/u-s-chronic-disease-indicators-cdi> (Chronic Disease Indicator)

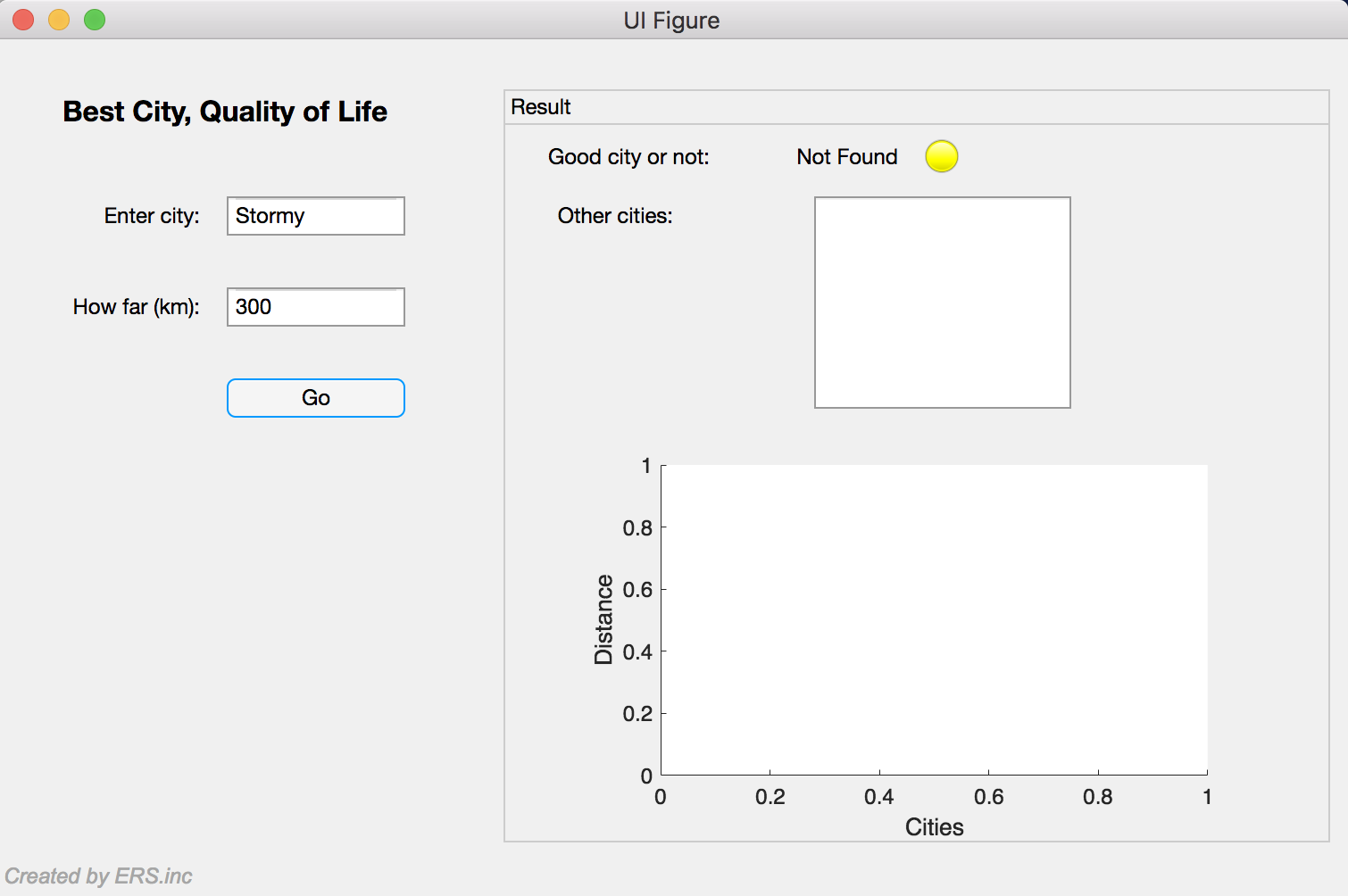
**Fulfilled Requirements**

* In the file ‘getChrDis.m’, we have a function which analyses the data set with more than 100,000 data sets and gives us a numerical value for disease indicator for each state.
* In the file, ‘graphFun.m’, we have the function which plots a graph in the app designer. It plots 3 good cities based on their distance from the city which is entered by the user.
* In the file ‘getData.m, from lines 79-84, we write the data about the cities close to the input cities in a file. This file stores all the good cities which are displayed on the app designer in the order of their relative closeness from the city entered by the user.

Open the application’ goodCities.mlapp’. The application may take some time to run as it is opened in a new matlab window.

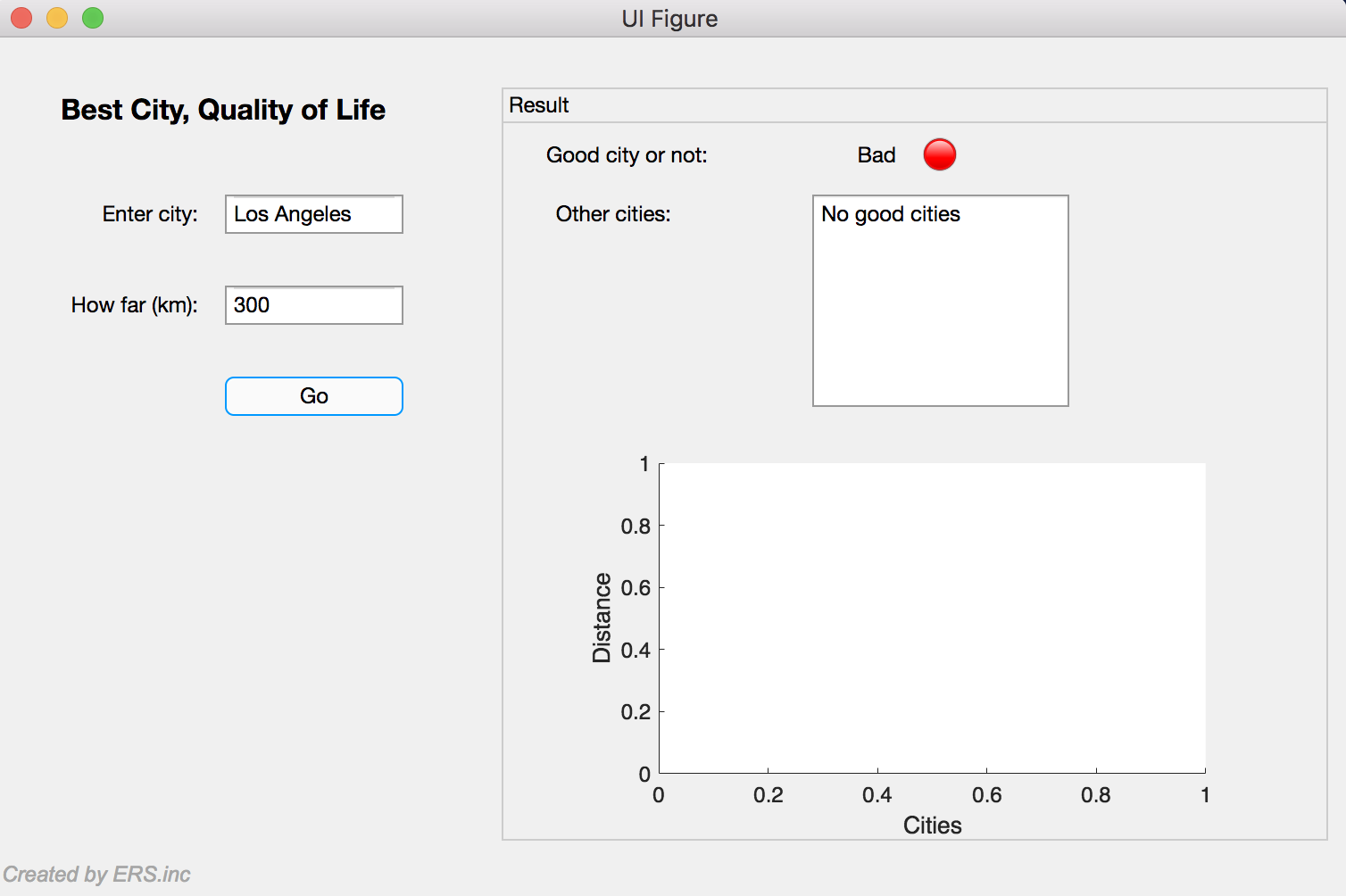


After opening the application, the user will be prompted for the city name they intend to move to and the maximum distance they are willing to travel if the city is not deemed optimal. Please type the city name carefully starting with a capital letter and the rest being small letters. For example, “Boston” is acceped and “boston” is not. If the city has two names, both the names should start in capital letters. For example, “Los Angeles” is accepted and “Los angeles” is not. Also, please input the distance in the other box and do not leave it empty. Once the user clicks go, the application will take approximately 10-15 seconds to process all the data and give the user the output. Below there are some examples:

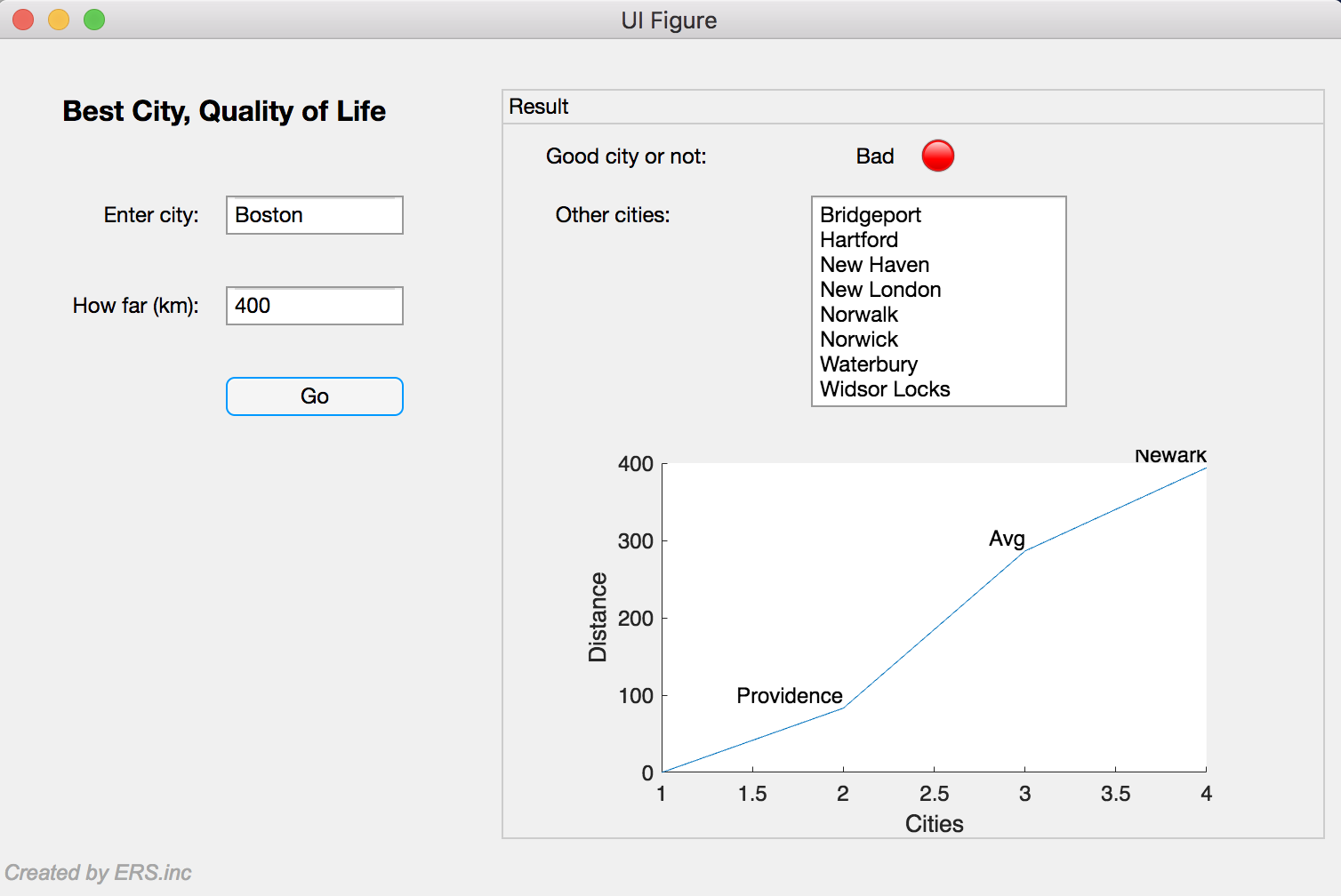


Please do not forget to input the distance.

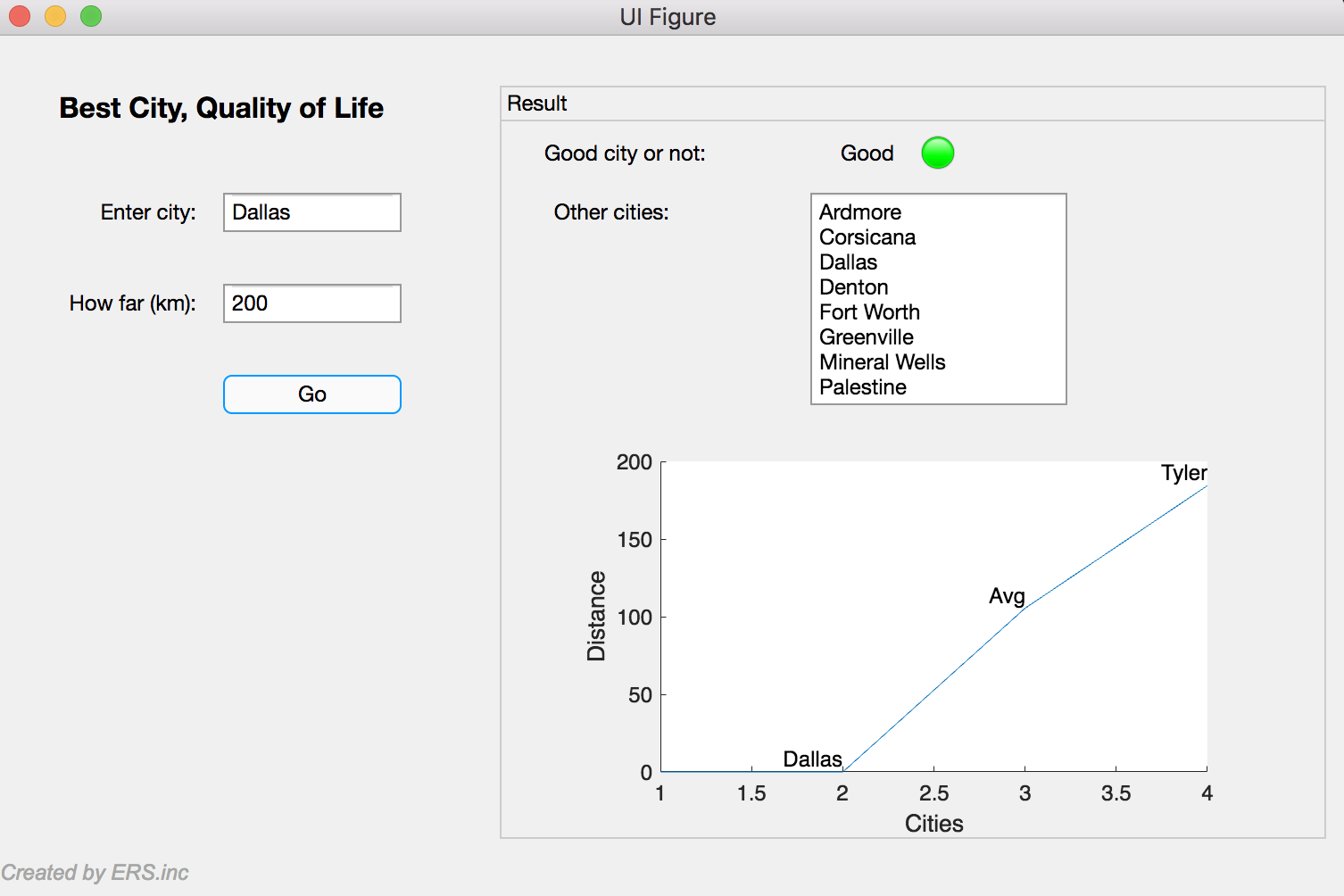
If the city is not found in the city dataset, the program will show 'Not Found' with a yellow lamp.



If the migration coefficient of the city is less than the optimum migration constant, the lamp will glow red. Then, it will look for other alternatives for the city entered by the user within the desired radius. If there are no cities with the migration coefficient greater than the optimum migration constant, it will display no good cities.



If there are alternative cities with migration coefficient greater than optimum migration constant, it will display the list of all the good cities and plots the graph based on the distance of the cities from the city entered. It will plot 3 points. The farthest city, the closest city, and the average distance of all the cities.



Finally, if the migration coefficient of the city input is good, the lamp will be green and the program will print 'Good'. The second plot value is the city input because, since it is a good city, it needs to be plotted.