User Interface Design

1. <https://www.youtube.com/watch?v=iAGeiWCkAI8>

(My Road Trip: A 15-112 Term Project)

The main concern I had with “My Road Trip” was the fact that the map depicting the visual directions for the route did not precisely follow the roads on the map. In my program, I used the Google Maps Javascript API and modified some HTML code I found online here (<http://aspforums.net/Threads/151361/Google-Maps-V3-How-to-draw-route-between-two-locations-points/>) to implement a map that pops up in the web browser that also shows a route that precisely follows the roads (at least for short distances).

1. Google Maps

Having a text box to enter in addresses, like Google Maps does, is perhaps the most straightforward way for a user to enter in the location he wants to visit. This was also the way I chose to implement adding an address. I also took the zoomable/scrollable map idea from Google Maps (I used Google Maps API as well for this aspect!).

1. <http://www.solver.com/traveling-salesman-problem-visual-basic-and-evolutionary-solver> (demo showing how different TSP algorithms work)

The above site provides a TSP simulation that shows the incremental nature of each algorithm (how each algorithm works). I implemented this feature in the Splash Screen (discussed below), as well as in the TSP Algorithm Demos section. In the TSP Algorithm Demos section, I decided to take this one step further – to show the demo overlaid on a map corresponding to actual locations entered by the user! This provides a more compelling, engaging user experience: the user can actually view the TSP algorithm at work on the very locations the user entered himself.

*Splash Screen*

For my Splash Screen, I wanted to include an eye-catching animation that draws the user in (the analog of a good introduction or “hook” in writing a paper).

Hence, I decided to include a simulation of the TSP greedy algorithm as well as a simulation of the 2-opt algorithm. The distance at each stage of the route also flashes on the screen, and is helpful in showing that the 2-opt algorithm, when applied to the route generated by the greedy algorithm, leads to a better solution.

*Moving from Page to Page*

I thought that an interesting way to move from one page to another in the program would be through buttons – each button corresponds to a page in the program. The motivation behind this design came from designs of most websites, which are often organized by links taking the user to specific parts of the website. In my program, buttons provide a similar functionality.

*Static Map*

Having a static map that zooms into the locations the user enters in the address box was a design decision inspired by the official Google Maps. Undo, Redo features and a “Go to Start?” button were also design choices taken from suggestions at the User-Study-Athon.

*Finding Nearby Gas Stations*

In finding nearby gas stations, I also thought it would be easy on the user if he could simply double click an address of a gas station to add it into the list of addresses he wants to visit. This is preferable to having to memorize the address, change back to the addresses page, and type it into the text box.