

Divvy Van Navigator Map Optimization Scenario

Group 1: Andy Hansana, Kevin Elliott, Vincent Weaver, Ryan Crowley

The Divvy Van Navigator will pull from various Divvy APIs, which provide values including: station name, docks in service, status of bikes/scooters, and location. Other variables should be implemented as needed in order to produce more efficient routing results.

The purpose of this application is to constantly optimize the best convenient route for a Divvy services driver making maintenance stops at various Divvy locations. They should know how many bikes are broken, and how many need to be repaired. The same goes for the Divvy scooters as well. Google Maps, or a similarly reliable location service should be utilized as it will be an effective tool for guiding the driver along the best route possible.

There are numerous factors during the day that need to be taken into consideration in order to produce a convenient route. Primarily, the app should recalculate and optimize the route during high-traffic hours (especially in cities like Chicago). Secondly, the software should pull relevant construction data and alter the route in order to minimize idle time in certain high-concentrated areas. Finally, the same should be done for hazardous weather conditions, which may also cause some of the previous two obstacles mentioned.

Scenario: "Map Optimization"

In this scenario, it will be primarily focused on map implementation and optimization as well as improving the login screen for the DIVVY driver for their unique credentials. The startup screen will prompt the user to input a username, password, and van ID number that will be able to identify which driver is logging into the application. Currently it doesn't take in those data values and identify that driver. Also to implement will be the map that will take care of how the data of what divvy stations nearby are displayed and

The current location of where the user is located. In the near future a pathway will be optimized for displaying the quickest route to a location using the dijkstra's algorithm. Once the map is implemented whether it is (Apple, Bing, Yahoo) the application will be able to move at a quicker rate of development as the map display has a huge part of the application and its role. Afterwards, certain notifications will be implemented.

As for the controller class, the plan is to implement action events to the buttons created by the fxml file and allow the user to build their own personal transit job data using the DIVVY API along with the map API. The challenge encountered involves pulling data from the data structures and both accurately and efficiently parsing and

updating the data for the user to use. The controller class must handle all user interactions with the program and be thoroughly tested using multiple test cases to ensure code is running properly. The plan is to test the users data being updated properly, the boundaries that each action event should have to ensure program stability, and that all actions are accurately being updated into the GUI. This goes hand-in-hand with making the controller class functions, and tracking the data structures from the API and differentiating it with the user data.

Instead of the user being directly placed into the application on startup, there will now be a login screen presented prior to launching the app. This login screen will have 3 inputs to provide such as drivers name, drivers ID, truck license plate. Besides a prompt to enter information there will also be a tab where users can get help and or a tutorial on how to do various functions on the application. While this information will be available to the user at any point in the app, it will be more prominent on the login screen. There will also be another point of input on the log in where users will have to select the region they are resupplying in, such as little italy, north loop, south loop, etc. This selection of the region will be used for initial searching of path options, to give the user a starting point of tasks.

