Project Plan

TITAN
(**T**he **I**ntegrated,**T**echnologically **A**ssisted way to **N**avigate)

Hassan Hassan
Jon Montag
Alex Orellana
Greg Quintanilla
Griffin Rhein

Project Overview

TITAN is a fleet mapping solution aimed to take a set of addresses, process them into a route and itinerary for delivery drivers to go from address to address and complete each delivery.

Project Deliverables

The project aims to deliver a working web solution for both administrators and drivers. The administrators will be able to upload sets of address from comma separated files or input address one by one and have the solution calculate the the most efficient route between the whole set of addresses. Administrators will be able to edit and complete these routes and then send them to drivers. Drivers will be able to see and update their statuses on the route as they go about their deliveries. Drivers have functionality to mark successful and failed deliveries.

Project Organization

Web Application

Many of the the functionalities of the solution will be available through web application so many of the requirements are all parts of the main web application and as such are listed in a group of requirements.

3.7.1 Login Page

Alex O. Hassan H. 2/23 - 2/29

This will be the first page displayed when connected to the webpage. This page will be fitted with a username and password entry box. Drivers and administrators will

also be able to create accounts through a sign up link. Once the user has the appropriate account they will be able to log in and be taken to one of the two displays based on what authorizations the user has.

3.7.2 Map Display

Alex O. Hassan H. 2/23 - 2/29

Both drivers and administrators will have a map view on their pages. This will house the map of the location the deliveries consist of. It will also be the home of the waypoint markers for each of the addresses for the route as well as the path between each of the points. It has drag and zooming capabilities so viewers can change their perspective around the map.

3.7.3 Itinerary display

Alex O. Hassan H. 3/1 - 3/7

The itinerary display will allow drivers to view their current routes, their upcoming routes, and their previous routes. It will also allow drivers to update their current status. For example if a driver is on a certain route and they have arrived at the delivery location then the driver will be able to mark the route as completed.

3.7.4 Address Input

Alex O. Hassan H. 3/1 - 3/7

The address input bar will be able to grab addresses one by one and we will store those addresses into a database. When the user clicks done, the user will have the option to calculate routes. In the case where the user has an existing file with routes

the user will be able to upload those routes into our database from their machine and then the user will be able to calculate routes. In this case the file would have to be a csv file and our program will be able to extract the routes.

3.7.5 Route Display

Alex O. Hassan H. 3/8 - 3/14

For administrators once routes have been calculated the administrator will be able to see an entire map with highlighted routes and there will be indicators above the routes to indicate the route number. For drivers, the routes will be available to them in text form. This text form would take the format of turn by turn directions.

3.7.6 Driver update control display

Alex O. Hassan H. 3/8 - 3/14

Administrator will be able to update drivers accordingly. For example if a driver becomes unavailable then that driver should be able to be removed and another driver should be able to get assigned. New drivers will also be able to be added by the administrator.

3.7.7 Information Displays

Alex O. Hassan H. 3/8 - 3/14

Information displays such as distance travelled, time travelled, estimated time arrival would all be available to administrators and drivers.

3.7.8 Address Loader

Jon M. 2/23 - 2/29

_____This will take an uploaded CSV file and pre - process the addresses to be moved on to the address to GPS coordinate converter. It will also be used to pre - process single entry addresses. This step will also validate that all addresses are complete and can be used in the next phases. Invlaid addresses will be flagged and brought to the attention of the administrator.

3.7.9 Address to GPS Coordinate Converter

Jon M. 3/8 - 3/14

_____This function will take in the pre - processed data from the Address Loader and convert them into GPS coordinates to be passed along to the Pathfinding function.These coordinates will be housed in an array structure and will be placed into the map grid

3.7.10 Pathfinding

Gregory Q. 3/8 - 3/14

_____This function will take all the desired nodes in to be in the route and will traverse the map graph and create the shortest path it can that covers all of the addresses. This will use graphing algorithms to create the most efficient route it can amongst all of them and return a path from address to address with the distances and times between each point. The returned path will be passed along to compute an itinerary and will be displayed by the path display on the map display.

3.7.11 Itinerary Creator

Gregory Q. 3/22 - 3/28

This will take the created path and form an itinerary for the driver with an order they should go from location to location. This will be accompanied with an estimated arrival and departure time from each location as well as an estimated time for the full route. While the driver is on the route a real time will be tracked and compared to the estimated time to see if the driver is behind, ahead or on schedule.

3.7.12 Directions

Jon M. 3/22 - 3/28

Based on the routes created there will be a step by step process that will guide the driver each step of the way to each destination. This set of directions will be sent to the driver and displayed for them so they can navigate their way to their delivery points.

3.7.13 Driver and Administrator Login

Griffin R. 3/8 - 3/14

Users will be able to create accounts under either a driver or an administrator. When this account is created the information will be sent to be stored in a database. The password will be encrypted upon creation and storage.

3.7.14 Login Database

Griffin R. 3/1 - 3/7

A database will be stored with both administrator and driver login information. This will be updateable upon request. New users will be able to add themselves to the database via the login display through the sign up link. Users can be removed upon

special circumstance. Users will also be able to update their existing information from their account settings.

3.7.15 Password Security

Griffin R. 2/23 - 2/29

All user passwords will be encrypted and will go through a verification process to ensure that the user's accounts are secure and that even if the databases are accessed, no actual passwords will be stored only hashed keys.

3.7.16 Driver Update

Gregory Q. 3/29 - 4/4

As the driver moves along his route they will be updated the statuses of their deliveries. After each stop they will be able to mark the delivery as successful or unsuccessful and can report hazards or reasons for delays in the itinerary due to wrecks, traffic, or road closures.

3.7.17 Map Graph

Gregory Q 2/23 - 2/29

A graph will be implemented and will be used to store the back end processing of the addresses and the pathfinding. It will be a digraph as some streets can be one way and the distances between nodes will be stored. This will set up the basis for all of the pathfinding functionality.

Requirement Chart

Requirement	Duration (Weeks)	Dependencies	Developer
1. Login Page	1	14	Alex & Hassan
2. Map Display	1	10,11	Alex & Hassan
3. Itinerary Display	1	12	Alex & Hassan
4. Address Display	1	9	Alex & Hassan
5. Route Display	2	2	Alex & Hassan
6. Driver Update Display	1	12,11	Alex & Hassan
7. Information Displays	1	12	Alex & Hassan
8. load Addresses	1	None	Jon M.
9. Address -> GPS	2	9,18	Jon M.
10. Pathfinding	1	17	Gregory Q.
11. Itinerary Creator	2	10	Gregory Q.
12. Directions	2	11	Jon M.
13. Driver/Admin Login	2	15	Griffin R.
14. Login Database	2	16	Griffin R.
15. Password Encryption	2	None	Griffin R.
16. Driver Update	2	11,12	Gregory Q.

17. Map Graph	2	None	Gregory Q.

Project Calendar

Member	Wk 1 2/23-2/29	Wk 2 3/1 - 3/7	Wk 3 3 /8 - 3/14	Wk 4 3/15-3/21 Break	Wk 5 3/22-3/28	Wk 6 3/29 - 4/4	Wk 7 4/5-4/11	Wk 8 4/12-4/18	Wk 9 4/19-4/25	Wk 10 4/26-4/28
Greg	3.7.17	3.7.17	3.7.10	Off	3.7.10	3.7.11	3.7.11			
Alex	3.7.2	3.7.3	3.7.5 3.7.6	Off	3.7.5					
Hassan	3.7.1	3.7.4	3.7.7	Off						
Griffin	3.7.14	3.7.14	3.7.15	Off	3.7.15	3.7.13	3.7.13			
Jon	3.7.8	3.7.9	3.7.9	Off	3.7.12	3.7.12				

Configuration Management

A team github has been created and shared with each participating member. Members are told to acquaint themselves with its functionalities and are tasked with adding, committing and pushing to their respective branches. Coordinated merges will be scheduled based on the completion of components. Team members will have the autonomy to coordinate smaller merges for integration testing as it makes sense. Before builds all members will push their finished components so that. Versioning will start version 1.0 and only track Major.Minor versions.

Risk Analysis

The largest foundation for the project is processing all the information to set up the graph and create the paths. This is some of the highest risk pieces just because so much of the other functionalities depend on information created by these components. There is plenty of slack built into the project plan to account for the inevitable delays that come from pieces in the pipeline of the route and itinerary creation. The components of the web application do each have their dependencies of the functionality of the back end components. This can be done in two phases with the visuals of the front end being created regardless of the backend functionality and then can be hooked up as pieces of the the back end are completed.

Quality Assurance

To ensure the quality of our project, every code commit will be reviewed by a member. Also to ensure the robustness of our program, invalid addresses will be inputted to see if our code can identify that a route cannot be calculated. Proper benchmarks will be performed to access the speed and efficiency of our route calculations. Our addresses to coordinate conversion will also be tested to ensure that no matter the format of the address, the address should still be able to get recognized. Also, the password encryption part of our project will use a password encryption method

that is secure. The path finding part of our project should take into account that only streets can be part of the route which means that a path should not have diagonals that cross through non-street objects. The front end displays which include the map, the itinerary, the route display and so on will be properly formatted. For example in the user interface the map should be the center of attention and not be towards one corner of the screen and be super small. The same applies for the other displays. The login part of our project will properly identify drivers versus administrators and this will be tested.

The appropriate tests will be created to ensure the code is covered in handling all edge cases as well as invalid arguments. The proper functionality will be compared against small tests done by hand to ensure the proper algorithms are functioning properly. Team members are encouraged to write the appropriate tests for their components. If time allows, other members will review and test other pieces of code.