Matt Smith

John Garrison

Nicholas Sidwell

Meridian's Vision Document For the Betterment Of Peace and Prosperity of the People On Nimbus III

1 - Intro:

1.1 Purpose:

This document is used to define the development of the Sweet Fleet Tracking Suite and how it will operate.

1.2 Product Overview:

The Sweet Fleet Tracking Suite will track the general state and position of company owned fleet vehicles.

2 - User Description:

2.1 User/Market Description:

The users would be companies or entities that require management of a large fleet of vehicles in order to streamline tracking and maintenance, or to facilitate correction of errant drivers.

2.2 User Profiles:

Although each user will have a unique username and password, the distinction between users is irrelevant. Supervisors will be assumed to have limited knowlege of operating computers.

2.3 User Environment:

Given that our software will be web-based, users will be able to access it from multiple sources, such as: desktops, laptops, tablets, or phones. Thus, the environment is user dependent and not necessarily easy to scope.

2.4 Key User Needs:

2.4.1 The user needs to be able to track the location of an arbitrary number of vehicles.

2.4.2 The user also needs to be able to keep tabs on top speeds achieved during any

given use of a vehicle.

2.4.3 The user requires notifications for the arrival and departure of any and

all vehicles.

2.4.4 Notifications informing about the running state of the vehicle are also required.

2.4.5 Software should make note of any anomalies with regard to route and/or

schedule deviation.

2.4.6 Data collected needs to be stored for a yet-to-be-clarified period of time.

2.5 Alternatives and Competition:

2.5.1 US Fleet Tracking.

2.5.2 Fleet Tracking.

2.5.3 Verizon Networkfleet.

2.5.4 Teletrac.

2.5.5 Fleetmatics.

2.5.6 Fleet Tracker.

2.5.7 Fleet Complete.

2.5.x . . . ad nauseum.

3 - Product Overview

3.1 Product Perspective:

The product will be integrated with already existing hardware, and interacts with users

through push notifications and data logs.

3.4 Assumptions and Deployment:

The client will need training for use of the software. Given that the product is a web service, and hardware manufacturer will install required software components on the hardware, deployment is assumed to be straightforward.

3.5 Cost and Pricing:

$100,000 will ensure the buyer receives all code, compiled software, training, and documentation.

4 - Use Cases

The user navigates to the Sweet Fleet Tracking Suite web page.

The user fills in username and password fields with a username and password unique to each supervisor.

The user clicks the login button or hits enter when the password text area is selected.

If login fails, a error message is displayed in a new window and the user is able to try again.

If after 5 attempts, the users account is locked and they have to wait for a day before attempting another login.

Upon successful login, the user sees the main interface containing a scroll view for selecting employees, a map with all currently active vehicles displayed at their current locations, buttons to

navigate to the following interfaces:

- snapshot page to view on/off, arrival/departure, and maintenance notifications for all vehicles that were active during that day for the supervisor

- vehicle maintenance page to view/edit a selected vehicles maintenance schedule and information

- a vehicle status page based on the selected vehicle/vehicles.

The user selects one or more employees from the scroll.

The user clicks the snapshot page button.

On the vehicle mainenance page, the user sees the following for each active vehicle:

- The current state of the vehicle ( ie. on/off ).

- The arrival and departure notifications.

- A flag indicating maintenance is needed, if needed.

The user clicks the home button to return the main interface.

The user selects one or more employees from the scroll.

The user clicks the vehicle maintenance page button.

The user sees the the following on the vehicle maintenance page:

- The date of the most recent maintenance of the selected vehicle which includes:

- The most recent tire change.

- The most recent tire rotation.

- The most recent oil change.

- The most recent state government defined vehicle inspection if required.

- The make, model, year, color, trim package, mileage, license plate number, and VIN of the selected vehicle.

- An edit button for to the right of each of the above items for the vehicle maintenance page.

Clicking an edit button to the right of a field allows the user to modify the contents of the field.

The user clicks the home button to return the main interface.

The user selects one or more employees from the scroll.

The user clicks the vehicle status button.

The user sees the following on the vehicle status page:

- Current Speed.

- A list of speeds exceeding the limit and the duration spent at such speeds.

- Current Location.

- Time of Departure from the company garage if the vehicle is in use that day.

- Time of Arrival to the company garage if the vehicle was that day out.

- Distance traveled for the day.

- A list of oddities from the day.

- Total time the vehicle was more than 5 miles per hour over the speed limit.

- The ability to "ping" the vehicle to ensure hardware functionality.

The user clicks the button to ping the selected employee's vehicle.

If the vehicle hardware is functioning correctly, the user will receive a notification stating that the vehicle hardware is functioning correctly.

If the vehicle hardware is not functioning correctly, the user will receive a notification stating that the vehicle hardware is not functioning correctly.

The user clicks the home button to return to the main interface.

The user clicks the logout button and is no longer able to access the Sweet Fleet Tracking Suite unless they login again.

5 - Features

\* Will be derived from use cases.

6 - Other Requirements

6.1 Standards

6.1.1 CamelCase.

6.1.2 Curly braces will always be on new lines.

6.1.3 Proper version management through Git or similar SVN.

6.1.4 Thorough documentation in code.

6.2 System Requirements

6.2.1 Persistent connection between cars and servers.

6.2.2 System is reliable, with uptime greater than 99%.

6.2.3 System is accessible on mobile, tablet, and desktop devices.

6.3 Licensing and Security.

6.3.1 Data needs to be encrypted.

6.3.1 Licensed GNU open source license.

7 - Glossary

N/A