Matt Smith

John Garrison

Nicholas Sidwell

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

**1.** **Introduction**

**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 knowledge 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 requires the ability to track the location of an arbitrary number of vehicles.

**2.4.2** The user requires the ability to track 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** The user requires notifications informing them about the running state of the vehicle.

**2.4.5** The user requires notification of any anomalies with regard to route and/or schedule deviation.

**2.4.6** The user requires data to be stored for a year and a half.

**2.4.8** The user can edit information pertaining to each tracked vehicle.

**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 nauseam.

**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** *Costs and Pricing*

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

**4.** **Use Cases**

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

**4.2** The user fills in username and password fields with a username and password unique to their company.

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

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

**4.5** 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
* vehicle maintenance page to view/edit a selected vehicles maintenance schedule and information
* a vehicle status page based on the selected vehicle/vehicles.

**4.6** The user selects one or more employees from the scroll.

**4.7** The user clicks the snapshot page button.

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

* The current state of the vehicle (i.e.. on/off).
* The arrival and departure notifications.

* A flag indicating maintenance is needed, if needed.

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

**4.9** The user selects one or more employees from the scroll.

**4.10** The user clicks the vehicle maintenance page button.

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

* The employee assigned to the selected vehicle. Displayed as “None” if no employee is assigned to the selected vehicle.
* 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 to the right of each of the above items for the vehicle maintenance page.

**4.12** Clicking an edit button to the right of a field allows the user to modify the contents of the field and causes a save button to replace the edit button.

**4.13** Clicking the save button saves the contents of the selected field to the server.

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

**4.15** The user selects one or more employees from the scroll.

**4.16** The user clicks the vehicle status button.

**4.17** The user sees the following on the vehicle status page:

* A button to view the history of the vehicle.
* Current Speed.
* A list of speeds exceeding the limit and the duration spent at such speeds.
* Current Location.
* Expected time of departure.
* Time of departure from the company garage if the vehicle is in use that day.
* Expected time of arrival.
* Time of arrival to the company garage if the vehicle was that day out.
* Map of the expected route and current position of the selected vehicle.
* Distance traveled for the day.
* A list of anomalies for that 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.

**4.18** The user clicks the history button and is taken to the vehicle history page.

**4.19** The user sees a list of buttons corresponding to the days for the last year and a half the vehicle has been tracked. If tracking data is less than a year and a half, all days tracked are shown.

**4.20** The user clicks a day in the list.

**4.21** The user sees the following in new window:

* Time of departure from the company garage if the vehicle is in use that day.
* A list of speeds exceeding the limit and the duration spent at such speeds.
* Expected time of departure.
* Expected time of arrival.
* Time of arrival to the company garage if the vehicle was that day out.
* Distance traveled for the day.
* A list of anomalies for that day.
* Total time the vehicle was more than 5 miles per hour over the speed limit.
* A button to return to the vehicle status page.

**4.22** The user clicks the return to vehicle status page button.

**4.23** The user sees the vehicle status page.

**4.24** The user clicks the edit button to set the expected time of departure.

**4.25** The user enters the estimated time of departure in the time of departure field.

**4.26** The user clicks the save button to save their changes to the server.

**4.27** The user clicks the edit button to set the expected time of arrival.

**4.28** The user enters the estimated time of arrival in the time of arrival field.

**4.29** The user clicks the save button to save their changes to the server.

**4.30** 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.

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

**4.32** The user clicks the logout button.

**4.33** The user sees the login page.

**4.34** The user is no longer able access any features of the Sweet Fleet Tracking Suite unless they login again.

**5.** **Features**

**5.1** The Sweet Fleet Tracking Suite (SFTS) will track an arbitrary number of vehicles.

**5.2** The SFTS will maintain a record of top speeds achieved while the tracked vehicle is in operation.

**5.3** The SFTS will handle notifications sent from the tracking hardware of each vehicle signaling the departure of the tracked vehicle from the vehicles home.

**5.4** The SFTS will handle notifications sent from the tracking hardware of each vehicle signaling the arrival of the tracked vehicle to the vehicles home.

**5.5** The SFTS will handle notifications sent from the tracked vehicle when it is turned on.

**5.6** The SFTS will handle notifications sent from the tracked vehicle when it is turned off.

**5.7** The SFTS will display the intended route of the tracked vehicle.

**5.8** The SFTS will display the current position of the tracked vehicle.

**5.9** The SFTS will handle notifications sent from the tracked vehicle when an anomaly is encountered with regards to route deviation.

**5.10** The SFTS will handle notifications sent from the tracked vehicle when an anomaly is encountered with regards to departure time deviation.

**5.11** The SFTS will handle notifications sent from the tracked vehicle when an anomaly is encountered with regards to arrival time deviation.

**5.12** The SFTS will maintain vehicle data on the servers for a year and half.

**5.13** The SFTS will allow editing of all information pertaining to each tracked vehicle.

**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**

**7.4** *Anomalies*

Differences in what was expected.

**7.1** *Arbitrary Number*

A number greater than or equal to 0, but no larger than 100,000.

**7.2** *Handle*

Interpret hardware-encoded messages.

**7.2** *Home*

The default location where the tracked vehicles are kept.

**7.5** *Information*

All fields on the vehicle status page and maintenance page.

**7.3** *Notifications*

The hardware encoded message sent to the Sweet Fleet Tracking Suite.

**7.4** *SFTS*

Initialism standing for *Sweet Fleet Tracking Suite*.