May15-05 Project Plan

Augmented Reality Tractor Maintenance Application

John Deere Intelligent Solutions Group

Jesse Walther, Team Lead
Han Sang Youn, Testing Lead
Haoyu Liu, Webmaster
Brian Moran, Key Concept Holder
Tanner Hildebrand, Communications Lead

Manimaran Govindarasu, Advisor

Contents

References

Figures

Tables

Problem Statement

What Problem Does the Project Solve

Currently the John Deere API can serve some maintenance information on their tractors from their servers that have been downloaded from the tractors connection in the field. However, a farmer in the field may not have connection to cellular or wireless service; this application is intended to fill that gap by connecting directly the tractor via wireless to a device on board the tractor.

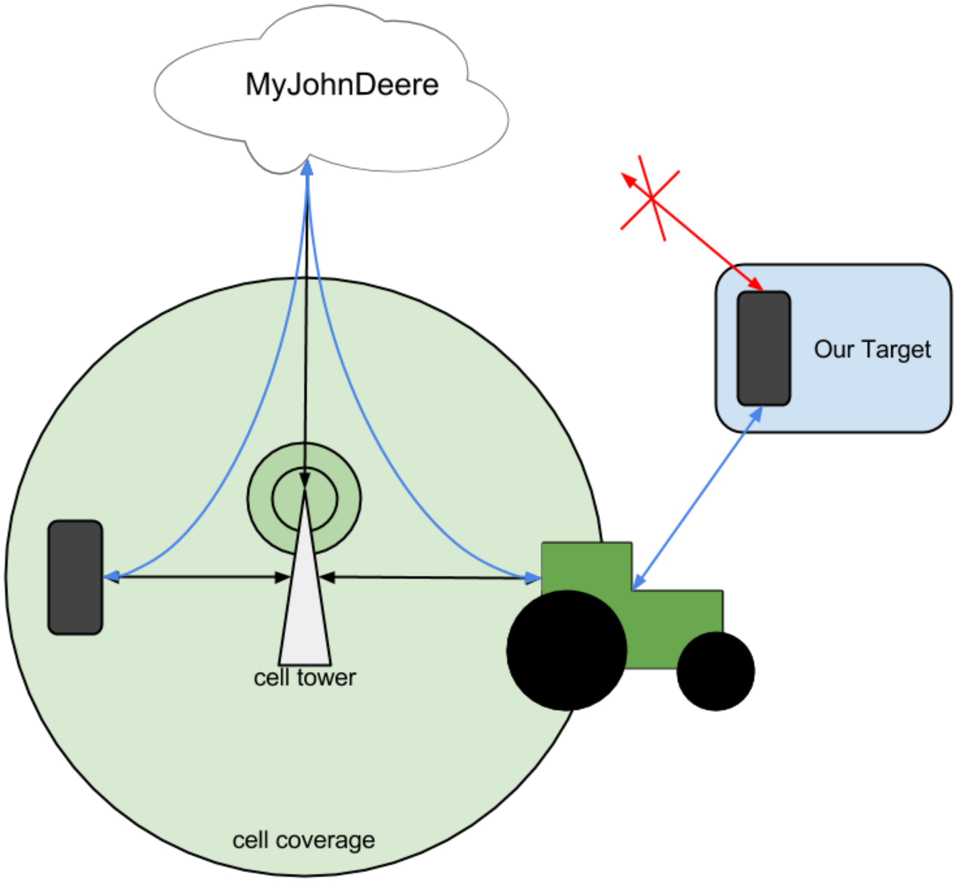

Figure 1: Problem Visualization

We propose a smartphone app for a farmer to review the status of his machines via a touch screen interface. Such an app would allow for farmers to interact with their vehicles to know the fuel levels and other important information about their vehicle. If this app is brought to completion it could be expanded to go cross platform and have other utilities as well.

Our primary customer is John Deere and their clients and employees. It would be targeted to three main users: farmers, dealers, and developers. Given the app works on one platform of smartphone, we would focus on making the app mutli-platform to reach the maximum amount of people.

What Value Does the Project Offers

This application is aimed at providing the owners of John Deere tractors with an easy to access resource for checking the status of their vehicles. It will allow the owners of the tractors to quickly look up maintenance information, location, vehicle identification, and other miscellaneous data on the tractor. Using image recognition software and Augmented Reality libraries the process should provide a simple point-and-shoot style of look up to make operation easy for tractor owners.

Developers will find use in the applications 'Developer Mode' which will allow a developer to connect directly with the tractor in order to communicate important status information. The developer would then be able to perform administrative and debugging tasks such as download log files.

Vehicle dealers are intended to benefit from having a convenient method to look up, sort, get alerts, and quickly display information about vehicles they have registered. This application will allow them to manage their vehicles remotely and stay up to date on their condition at all times.

Is this Project Feasible

This project is entirely feasible to complete given the resources and timeline available. Much of the required data is available through John Deere's developer API and the constant connection with their tractors. Our team will not have to make any changes in hardware and will be focused primarily on communication with the API data provider and tractor internals. Modern mobile applications are powerful and many resources are available to support developers; the development community for our primary development platform (Apple iOS) is large and experienced and will provide a solid foundation to build this application.

Deliverables

First Semester

Design Plan

Monthly demos
Prototype iOS Application and Demonstration

Second Semester

Monthly demos
Final Working iOS Application

Specifications

This application will work on the Apple iOS platform, minimum version 07, and operate in these modes and conditions.

Farmer Mode

The farmer mode should provide the following features to the user:

Augmented Reality View for Tractor Maintenance: this feature will attempt to automatically detect the owner's vehicle and download and display the most up to date mechanical and maintenance information for that vehicle in an augmented reality environment. Information may be displayed as widgets in full augmentation or, if the user chooses, in a simpler mode to conserve battery.

Maintenance Information: display information relevant to the owner in a low battery consumption/simple view mode. This feature will include two distinct pages: a 'Quick View' page for quickly seeing data on the vehicle, and 'Alerts' which provide notice of any mechanical/maintenance parameters which have crossed a given threshold.

User Account: This feature should allow the owner to securely log in to an account registered on the local device.

Dealer Mode

Dealer mode should provide all of the features from the Farmer mode plus the following:

The Dealer Mode will provide additional maintenance information and alerts detailed in the parameters table.

Developer Mode

Developer Mode will include all features from Farmer and Dealer mode plus the following:

The app will be able to download logs from the MTG and may provide some information about these logs.

Security

This application must make use of best practices in order to avoid transmitting sensitive data over unsecured networks and take steps to protect any data that does pass over a network through any medium. The applications operation must not expose or reveal any sensitive information to anyone but an authenticated user.

Connection

Data for each mode will be accessed through a connection to the MTG on board the tractor via WiFi. The MTG will provide an API for requesting mechanical and maintenance information through a wireless TCP connection.

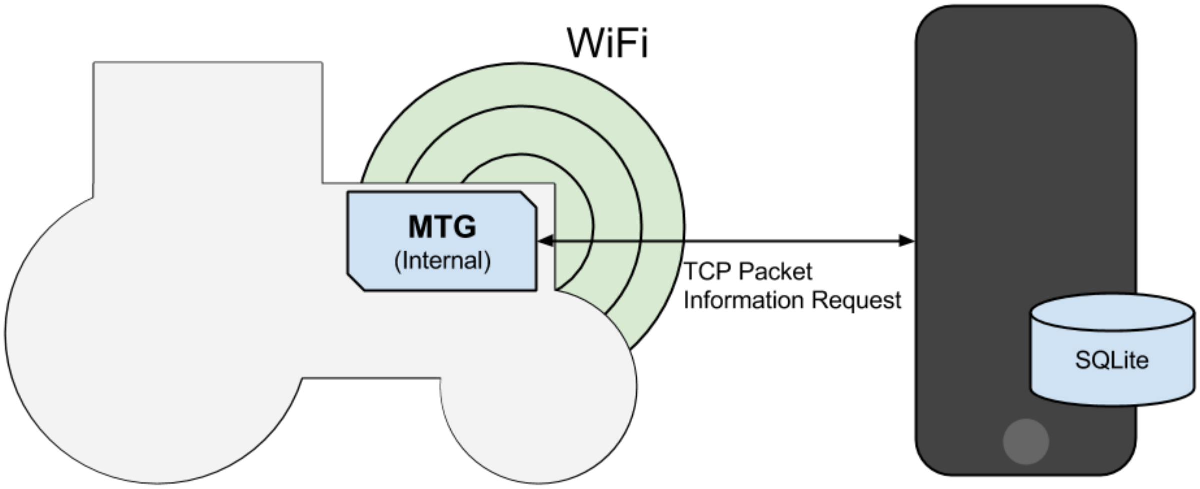

Figure 2: Connection Visualization

Concept Sketch/Mockup

Screen Flow

The overall navigation and traversing of the application should should maintain a clear and intuitive path for the user to progress quickly to the information which they are most interested in. In this way there are two primary paths to take:

Tab Button Navigation

The tabbed menu navigation style puts the highest level of navigation in a small array of tab buttons which are always available to quickly switch to important views.

Drawer Navigation

The second option is using a ‘Drawer Menu’ style which puts the highest level of navigation in a sliding menu which is hidden from view unless called upon by tapping a button or gesture swiping.

Solution

Particularly with iOS there has developed a stigma towards the drawer menu and the ‘Hamburger Button’ associated with it which has encouraged mainstream development practices towards more direct user interfaces such as the tabbed menu. As such we will use primarily the tab menu approach with the less used features and settings set in a separate tab as a sort of compromise between the two styles.

(Figure continued on next page)

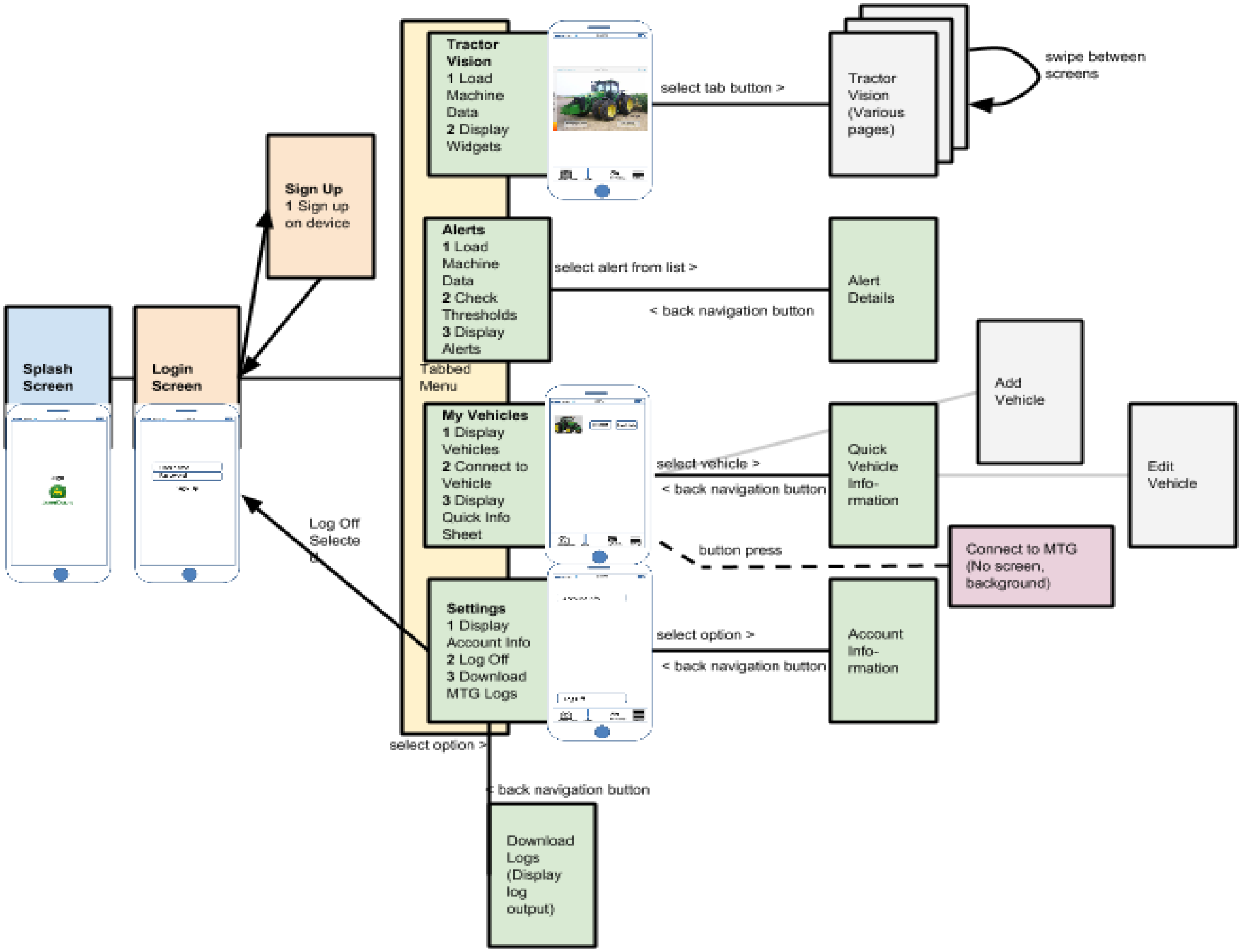

Figure 3: Screen flow for Application Transitions

Screen Concepts

We strive to maintain a comfortable and familiar user interface for iPhone users.

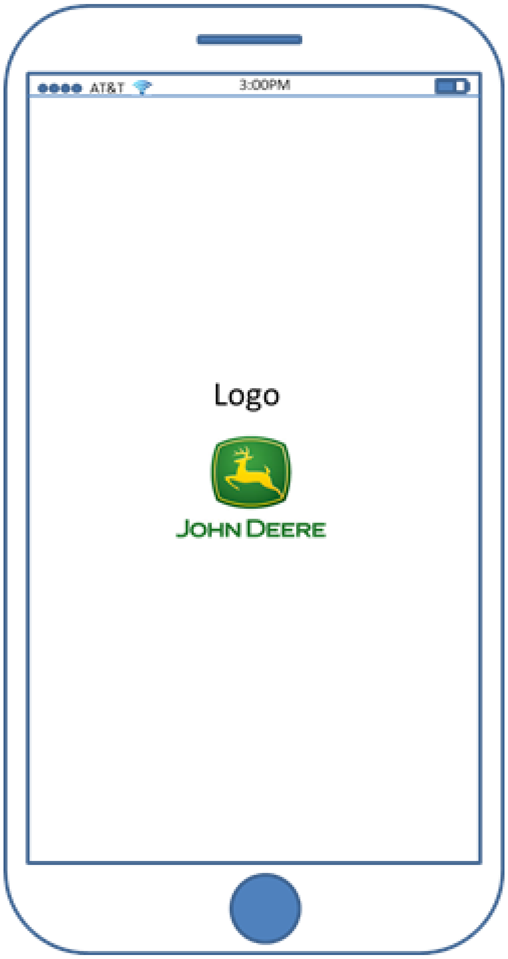

Figure 4: Splash Screen

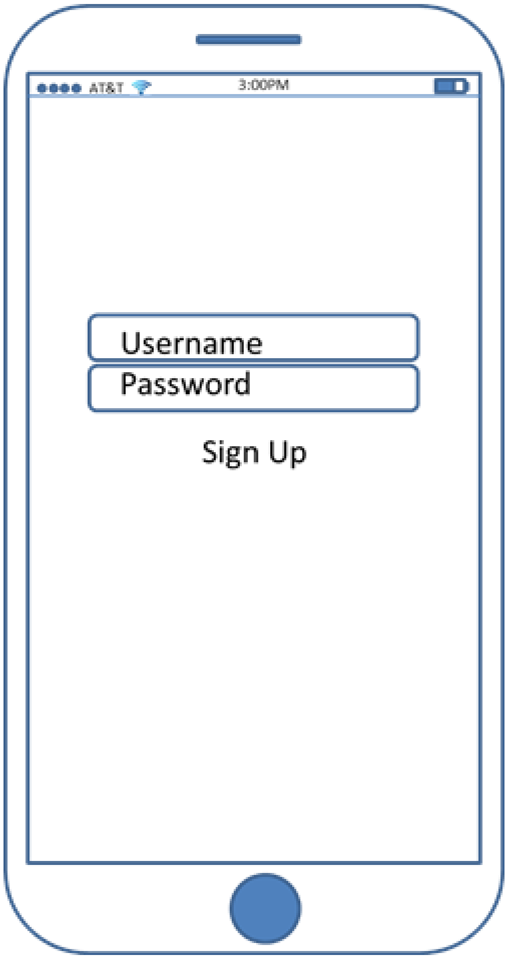

Figure 5: Login Screen

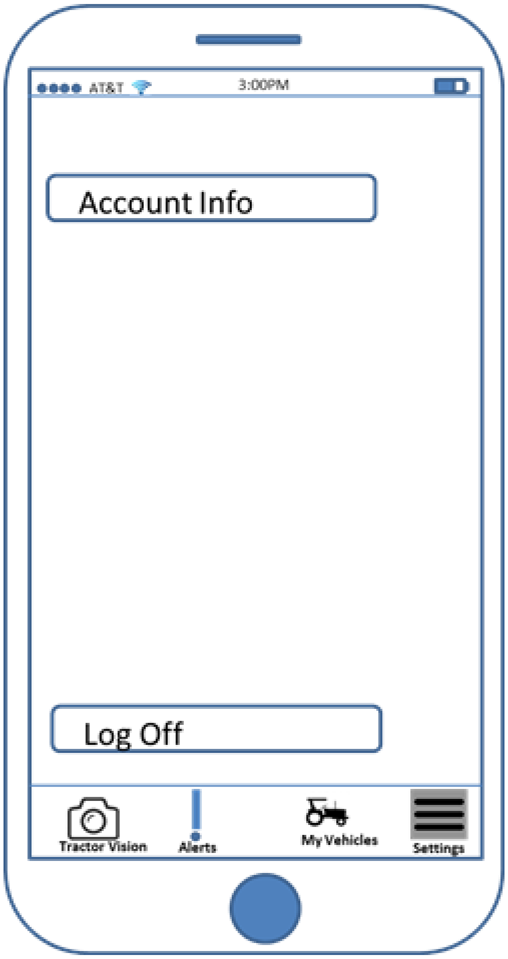

Figure 6: Settings Tab

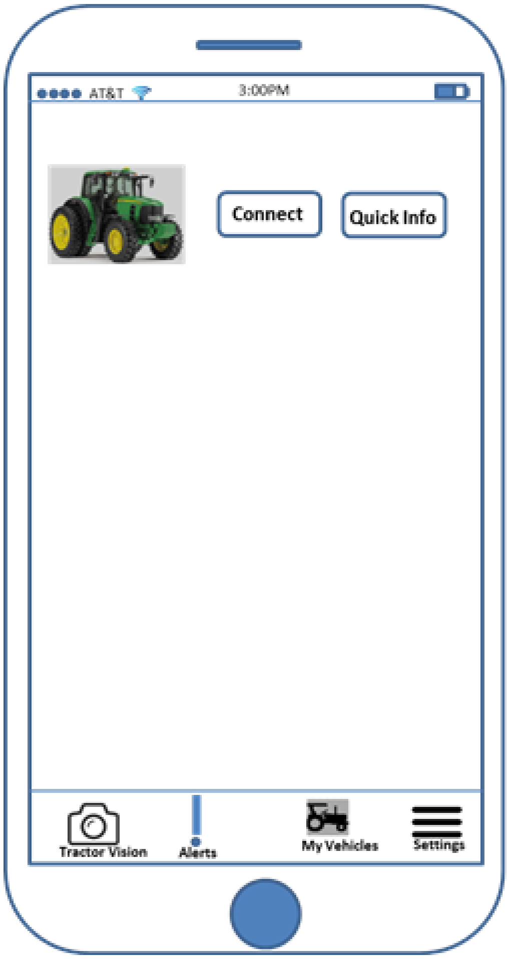

Figure 7: My Vehicles Tab

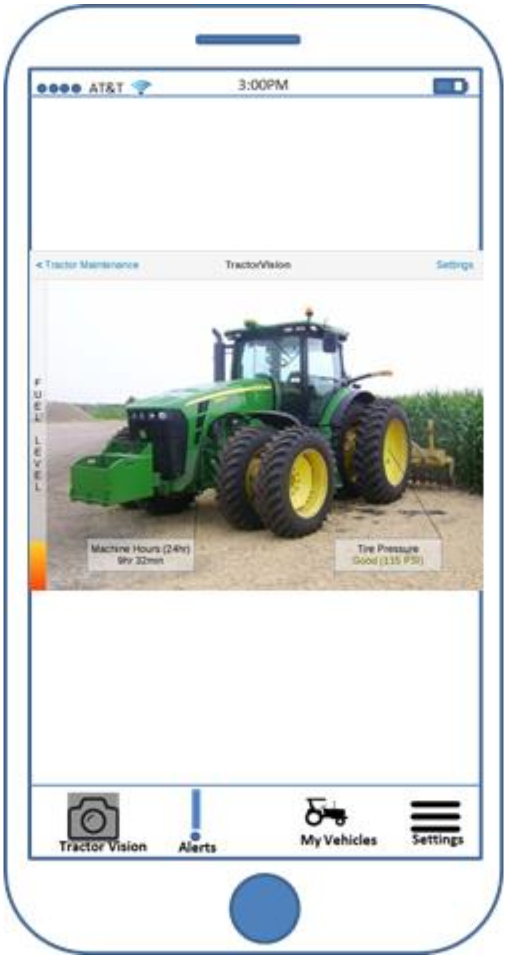

Figure 8: Tractor Vision Tab

User Interface Description

Software

The user interface will make use of Apple iOS design guidelines in order to provide a consistent and familiar user experience and provide functionality to the application.

For more information on designing for iOS see:

https://developer.apple.com/library/ios/documentation/userexperience/conceptual/mobilehig/

Additionally the applications interface will adhere to John Deere's trademark guidance. For more information on this see:

http://www.deere.com/en_US/docs/Corporate/citizenship/john_deere_tm_guides_sponsorships.pdf

Functional Requirements

The application must be able to meet the following requirements:

- Work for the most recent iOS versions (7 and 8)
- Provide basic local log in
- Add tractors to a user account
- Display the vehicles and available maintenance information associated with individual tractors
- Provide alerts for any tractor maintenance information which has passed a provided threshold
- Provide an Augmented Reality environment in which the owner's tractor is automatically recognized and maintenance information on the vehicle is displayed
- Operate in 3 modes: Owner, Developer, Dealer

Non-functional Requirements

In order for this application to be practical the following conditions must be met:

- The Tractor Vision should identify and display information for a tractor in under 5 seconds from target acquisition
- The application should not leak any sensitive data through networks or application interfaces

Additional Requirements

In order to make this application usable and maintainable it is important to us to make the code therein as portable, readable, and modular as possible for future developers to build upon the base it will provide. As we will not be the developers maintaining the application our implementation of this application must be constructed with future teams in mind.

Additional requirements may be considered iteratively while developing.

Testing Procedures

Each modular interface servicing our application will have a unit test for each method. Each test must pass without a failure.

Additionally ISG will provide an MTG/API to test connection and interface interaction with once application and MTG API capabilities are complete. An iPhone 4 has been provided to test the application on in field environments.

A demo is provided monthly for ISG to validate that the requirements are being met and that the progress is acceptable to their standard and vision of the application.

Resources

Work Breakdown Structure

Because this is a single application with five persons developing for it attention to detail and clear designation of responsibility is one of our primary focuses. Weekly coordinating meetings and open communication will continue to keep the entire team updated on the status of the project and key deadlines.

Manimaran Govindarasu, Advisor

Team Organization

Jesse Walther, Team Lead
Han Sang Youn, Testing Lead
Haoyu Liu, Webmaster
Brian Moran, Key Concept Holder
Tanner Hildebrand, Communications Lead

Project Organization

ISG has suggested that we work this project in an iterative manner, that is constantly update requirements and timelines in short term leaps. As such tasks and requirements will be divided and assigned as we see the need arise.

Resource Requirements

Resource	Acquisition	Estimated Cost
Phone 4 for testing	ISG	~$100
Mac OS w/ Xcode	Personal or Campus	$0
Apple Developer Account	ISG	$99

Table 1: Resource Requirements

Project Schedule

Gantt Schedule figure:

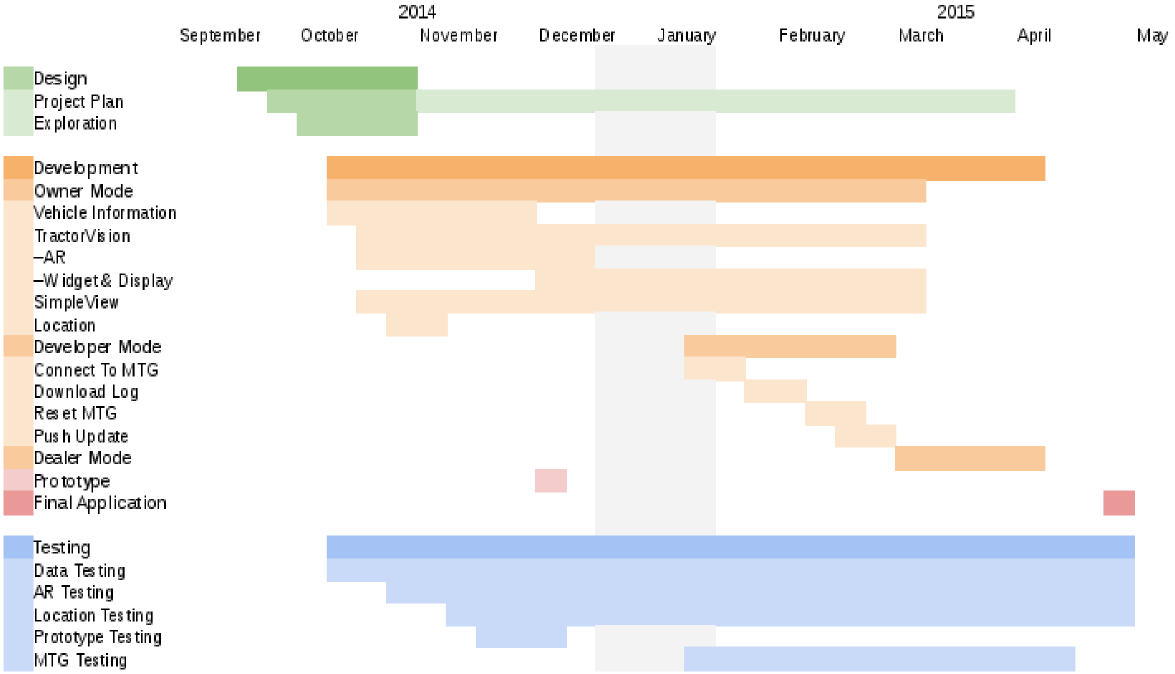

Figure 9: Tentative Project Schedule

Risks

Security

The most prominent security risk is accidental data leak over wireless networks such as WiFi and cellular. Using appropriate secure protocols (HTTPS) and well tested standard libraries for the iOS platform we expect to avoid any such data leakage effectively.

Internally the application must not require any more permissions than necessary for full use of the application. Similarly, the application should not make available any information which it has stored to any outside application unless otherwise specified by the requirements.

Market Literature/Survey

Information provided by ISG indicates that the majority of the tractor owners likely to use this application tend to be iPhone users. As such our development will be focused around iOS platform development.

Conclusion

This project has been broken down and steps chosen to facilitate our schedule and ensure that the project is complete on time and to standard. We are confident in the ability to complete the project fully and as meet as many requirements as allowed by the iOS structure as possible.

Given the limitations emplaced by the iOS operating system some concessions may need to be made and will be addressed on a case by case basis.