

May1609

A.R.T.I.M.U.S.

Phase II

**Augmented Reality Tractor Information
Management Utility System**

Contents

| | |
|-------------------------------------------|----|
| References..... | 2 |
| Figures..... | 2 |
| Tables..... | 2 |
| Project Phase I..... | 3 |
| Previous Key Functional Requirements..... | 3 |
| Previous System Requirements..... | 4 |
| Previous Issues and Challenges..... | 4 |
| Problem Statement..... | 5 |
| What Value Does the Project Offers..... | 5 |
| Is this Project Feasible..... | 5 |
| Deliverables..... | 6 |
| Technical Approach and Design..... | 6 |
| System Block Diagram..... | 7 |
| Application Block Diagram..... | 8 |
| User Interface..... | 9 |
| Design Standards..... | 14 |
| Use Cases..... | 15 |
| Test Plan..... | 18 |
| Assessment of Proposed Solution..... | 20 |
| Validation and Acceptance Test..... | 21 |
| Resources..... | 22 |
| Risks..... | 24 |
| Market Literature/Survey..... | 24 |
| Intellectual Property..... | 24 |
| Conclusion..... | 25 |
| Team Information..... | 27 |
| References..... | 27 |

References

Figures

| | |
|---------------------------------------|----|
| System Block Diagram..... | 6 |
| Application Block diagram..... | 8 |
| UI..... | 8 |
| Overall Project Tree Diagram..... | 24 |
| Overall Project Completion Chart..... | 24 |

Tables

| | |
|--------------------------------------|----|
| Technical Approach and Design..... | 6 |
| Assessment of Proposed Solution..... | 20 |
| Resource Requirements..... | 23 |
| Project Schedule..... | 23 |
| Risks..... | 24 |

Project Phase I

This application is aimed at providing the owners of John Deere equipment with an easy resource for checking the status of their vehicles. As this project is a continuation of the previous year's work, it is important to recognize the work that was previously accomplished. This will also help clarify the new requirements and work that will be new to the project.

Previous Accomplishments:

- Connection with the MTG
 - Uses TPLink to connect with MTG to create wireless
 - Gather machine information and display it to the user on the device
- Connect to a vehicle using the connection ID
- UI is functional
- Stores account information on the phone
- AR works on small objects

Changes/Additions:

1. Connection with the MTG
 - a. New MTG will have on board Wi-Fi chip
 - b. Wont need TPLink
2. UI is themed like other John Deere apps
3. Stores data on the cloud rather than the mobile device itself
4. The mobile device becomes the server and the MTG becomes the client
5. Makes the design more secure
6. Send sms to wake MTG if it's not listening
7. AR works on the tractors in the field
8. Displays information to the user
9. Widgets for various areas of the machine
 - a. No need for Developer mode
10. All important information will be logged and available

Previous Key Functional Requirements

- Connects directly to MTG device on board the tractor via wireless and downloads maintenance data securely to the mobile device
- Maintenance Information and Alerts—provides information on the maintenance state of the tractor and will display alerts when maintenance parameters pass predefined thresholds.
- Augmented Reality View—provides a point and shoot method to identify a tractor and display maintenance information on this tractor.
- Farmer Mode—provides all the features listed above.
- Dealer Mode—provides all the features of the farmer mode and in addition this mode will allow a dealer to quickly sort, classify, tag, and search for vehicles he has registered on his app.
- Developer Mode—provides all the features of the dealer mode and in addition the app will be able to download logs from the MTG and may provide some information about these logs.

Previous System Requirements

- Runs on Apple's iOS mobile operating system starting from version 7 up to 8
- Must have working camera
- WiFi connectivity must be available on the device and enabled

Previous Issues and Challenges

- Image Recognition on Mobile Devices
- Learning curve for developing with iOS, mobile development, and Augmented Reality
- Security limitations of the iOS system which prevent easy handling of WiFi and other built in libraries provided by the Apple system.

Proposed Non-Functional Requirements

- Store data on the John Deere cloud rather than on the mobile device itself. This allows the account information to be recognized and retrieved from any device rather than just the device it was created on. This will also not have the app use a lot of memory on the device.
- Have the mobile device be the server and the MTG be the client. This allows the MTG to not always be powered on and can either listen or wake up when the server wants to request information from the MTG.
- Implement some standard security practices if the MTG software can support it. This is something that needs to be looked into, as it can be very beneficial.

Proposed Key Functional Requirements

- Mobile device connects to the MTG of the tractor via bluetooth to setup wireless and downloads machine data securely to the mobile device and stores in on the cloud.
- Augmented Reality View: allows the user to take of picture of their vehicle which will then show machine information and overlay buttons on the vehicle. These buttons will then provide more information of that status of the vehicle.
- Maintenance Information and Alerts: provides information on the maintenance state of the tractor and will display alerts when machine vitals pass predefined thresholds.
- The application will work in the following 2 modes:
 - Farmer Mode—provides all the features listed above.
 - Dealer Mode—provides all the features of the farmer mode and in addition this mode will allow a dealer to quickly sort, classify, tag, and search for vehicles he has registered on his app.

Project Statement

The current situation is that the John Deere API allows the individual to have access to some maintenance machine information of their tractors. This is done by connecting to John Deere's servers in order to view the information. The problem that arises from this implementation is that the farmer may not be able to connect to a network while working in the field. To bridge this gap our group proposes an iPhone app that allows the farmer to directly connect to their machine through wireless. The application would then allow the user to gain key knowledge of their machine by providing them with alerts, service information, and great visuals. This application will sustain a similar theme to other John Deere apps such that it makes the experience for the user more familiar and easier to use. Giving the farmer vehicle information on the mobile device with the machine health information overlaid right on top of their vehicle, makes their job quicker and simpler so they can get back to work as fast as possible. The application will have two different types of users: farmers and dealers.

What Value Does the Project Offers

Our project is designed for the owners of John Deere tractors in order to give them an easy resource to gain information and know the status of their vehicle. In a sense our project is designed to be a one stop shop for the farmer so that they have access to all of their vehicle's health information in one place. Through the use of image recognition and augmented reality software our application provides the user with a quick, interactive way of displaying the vehicle's information by just having the user simply take a picture of their vehicle. The design of this application will provide developers with information of the tractor through the use of log files. These log files will be logged to the MyJohnDeere API and be available to developers. 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.

Is this Project Feasible

Given our resources and schedule this project is definitely feasible. Last year's project team completed a lot of the major functionality and provided us with a great base. 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 focussed mainly on communication, user interface, and a robust augmented reality component. Mobile applications are extremely powerful and common, which means the support for developers is great. More specifically the resources for Apple iOS development are large and will become extremely beneficial.

Deliverables

By April 30, 2016, an iPhone app will be finalized, running on iOS 7 and above, which uses real-time image recognition to provide an augmented reality display of information about John Deere tractors. Progress will be shown, via live demos, with John Deere representatives monthly.

The robust, augmented reality feature should be built and tested by December (end of the Fall semester). By February, the app should also feature an improved, more reliable connection to each tractor's on-board MTG. The ability to store information and send reports to MyJohnDeere.com, as well as make decisions regarding machine maintenance issues should be implemented and delivered by the April 30th deadline.

Technical Approach and Design

| Component | Description |
|----------------------|-------------------------------------------------------------------------|
| System 1 | iOS devices with iOS 7+ |
| System 2 | Yukon MTG Low Energy Bluetooth Capability (client provided) |
| Xcode | Version 7+ Apple's iOS development Platform |
| MyJohnDeere | Cloud services provided by client API |
| Wikitude | Augmented Reality and image recognition software (free trial) |
| Vuforia | Augmented Reality and image recognition software (free) |
| Bluetooth Connection | Establish connection securely with Yukon through bluetooth |
| Bonjour | Establish wi-fi connectivity |
| UI | User interface specific to fit client and user needs |
| AR | Augmented reality to provided user with easy point and shoot capability |

Table 1

System Block Diagram

Currently data is being stored on the mobile device itself and we are looking to store all of the data on John Deere’s cloud component instead. Also the current MTG version does not have an on board WiFi chip however the newest one will (expected in about 3 months).

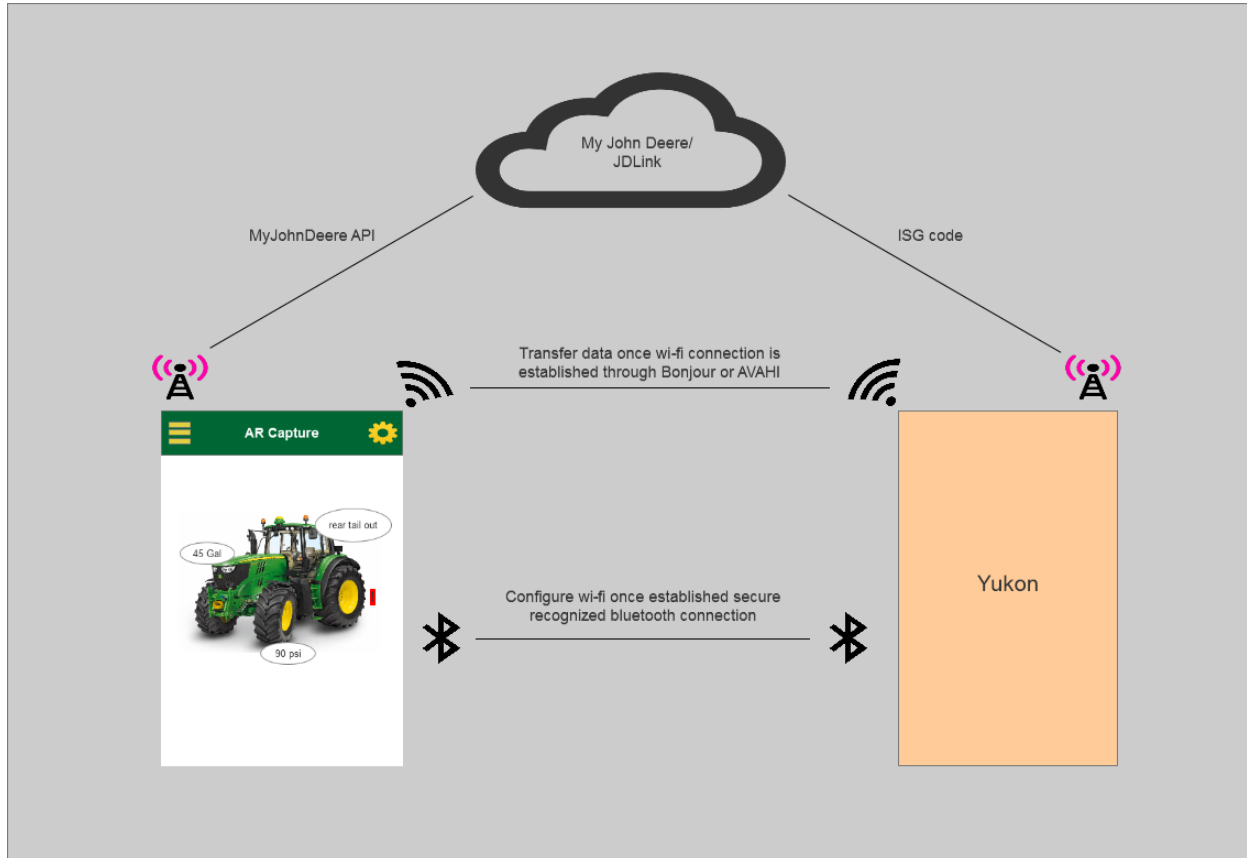


Figure 1

Application Block Diagram

Our application makes use of the Apple iOS navigation stack, which provides the user with a familiar and natural user interface. Our design standards are meant to comply with John Deere and Apple.

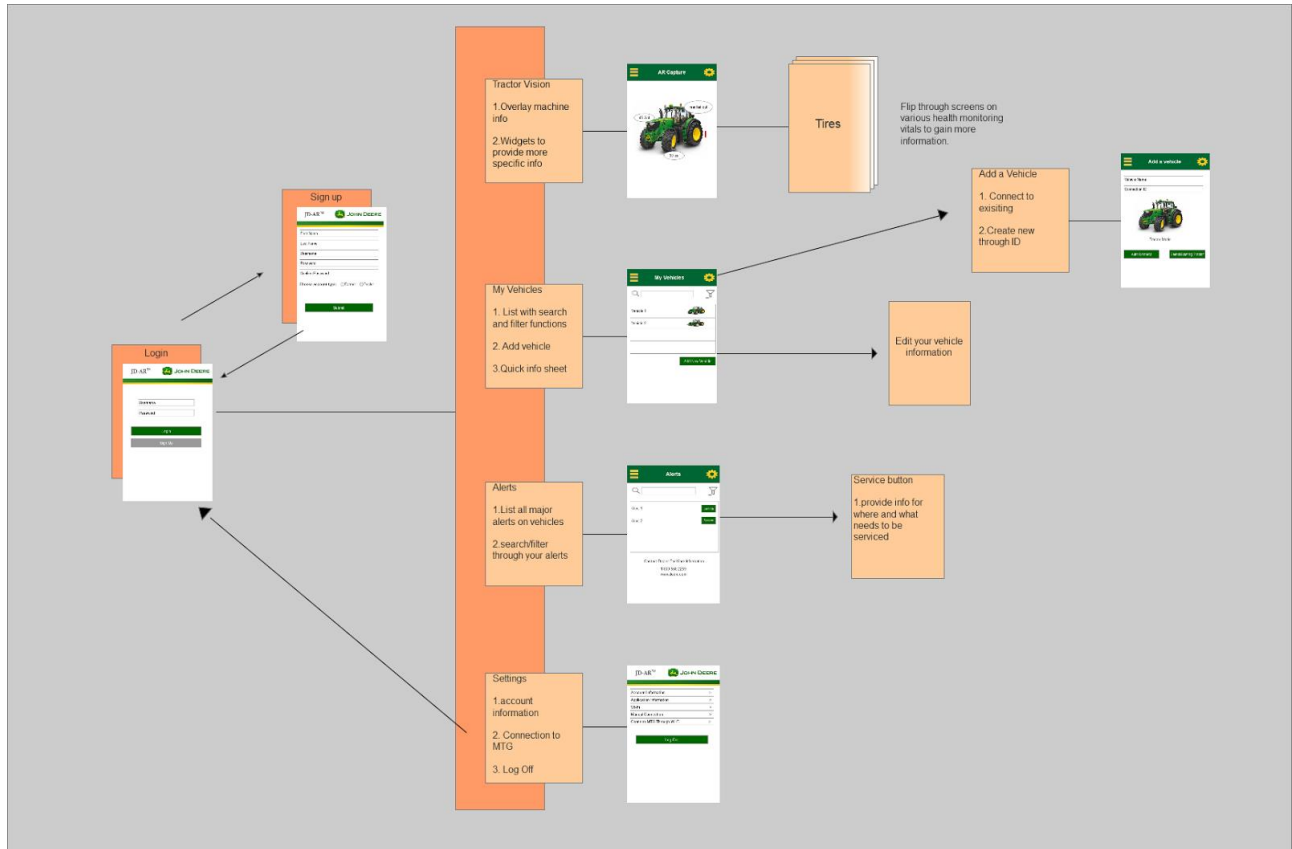
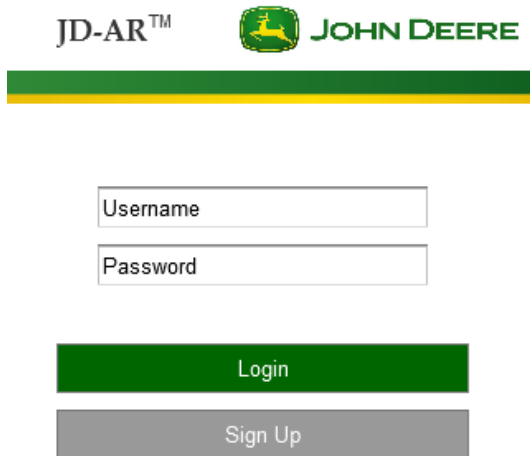


Figure 2

User Interface

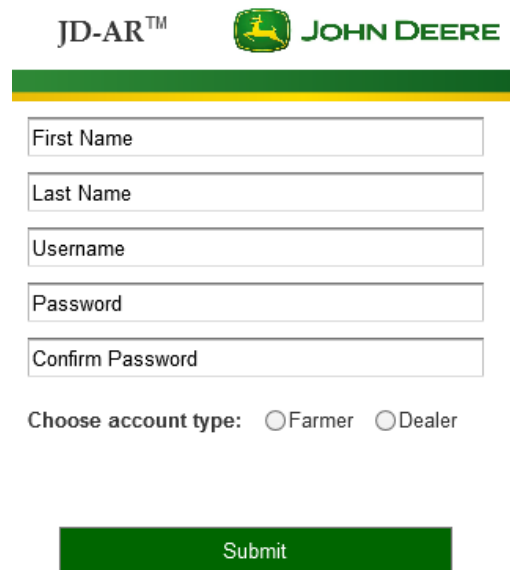
Login and Signup

After the application loads the user will be first presented with the login screen. The user will have to create an account before entering the application.



The login form features the JD-AR™ and JOHN DEERE logos at the top. Below the logos is a green and yellow horizontal bar. The form contains two input fields: 'Username' and 'Password'. Below these fields are two buttons: a green 'Login' button and a grey 'Sign Up' button.

Figure 3



The signup form features the JD-AR™ and JOHN DEERE logos at the top. Below the logos is a green and yellow horizontal bar. The form contains five input fields: 'First Name', 'Last Name', 'Username', 'Password', and 'Confirm Password'. Below these fields is a radio button selection for 'Choose account type: Farmer Dealer'. At the bottom is a green 'Submit' button.

Figure 4

The user will input the information shown above with some password constraints. The password constraints are 6-10 characters with one numerical, and capital letter. Upon creating the account the user will receive a notification showing that their account has been successfully created and then they can login.

My Vehicles

The user can then navigate to the my vehicles page once they have successfully logged in. This is the default home screen. Here the user can see their vehicles in a list view and search throughout the list based on some standard filters or by name. Those filters will include: date and type. Within the my vehicles page the user can also create a new entry if they choose.

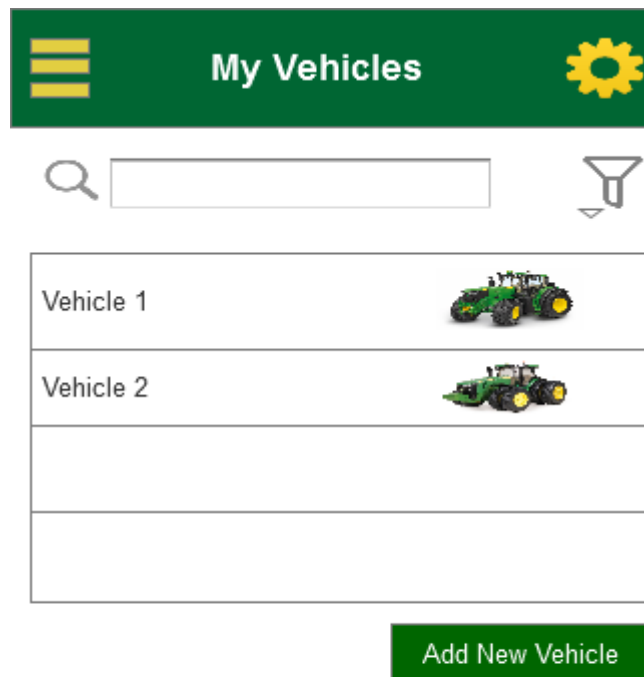


Figure 5

Add Vehicle

This page provides the user with a few ways to connect to a vehicle and add it to their list. The first way is to use the auto connect button which will use the connection ID if inputted to connect to the tractor. The other way to connect is to connect to network provided by the TPLink device.

Once connected you'll need to select the appropriate IP address. Click on the Settings tab and then on the 'Manual Connection' option. In the fields provided enter the appropriate IP addresses. If using the simple server you will enter the IP of the computer running the server and the port will be 4444. If using the MTG the IP will be 172.16.0.2 and the port will be 45544.

Finally you can return to the My Vehicles page and choose 'Connect' for your vehicle. You will be asked if you are connected to the appropriate network, choose the option that indicates you are to continue.

Add a vehicle



Tractor Model

Figure 6

Settings

The settings page can be accessed by selecting the settings icon in the upper right corner of the screen. From here the user has the option to see account and application information along with connection information.

The connection information is designed to provide the user with easy instruction on how to connect to their vehicle through the MTG.

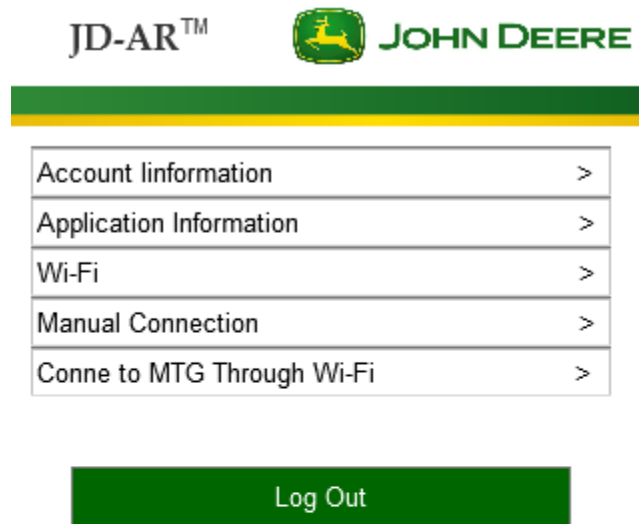
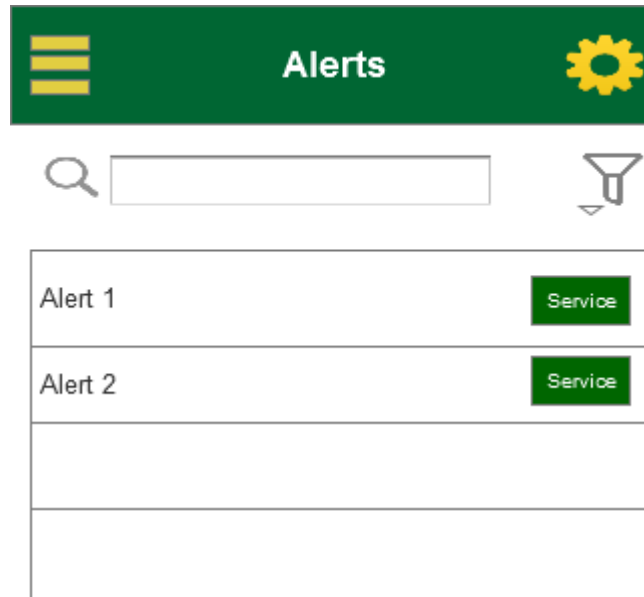


Figure 7

Alerts

The alerts page provides the user with a list view of all major machine health alerts of all their vehicles. The user can search through these alerts by filtering them based on category, importance, and date. From there the user can also click on the service button to brought to John Deere’s website for maintenance scheduling.



Contact Dealer For More Information

1-800-568-2259

www.deere.com

Figure 8

Augmented Reality

The augmented reality capture screen allows the user to take a picture of their vehicle and in turn see machine information being overlaid on top of their vehicle. From there the augmentation will allow the user to navigate through widgets for more information about the status of their vehicle.



Figure 9

Design Standards

Apple provides guidelines for all iOS developers to follow. Much of it is a focus on clean, consistent, and easy user interfaces. Follow these links to read the iOS [Human Interface Guidelines](#), or [Design Principles](#). This project will also consistently follow the John Deere design, shown in some example screenshots above, and fully disclosed in a mobile app styling guide received from a John Deere representative.

Use Cases

Use Case 1: Login

Primary actor: application user

Preconditions: the application has been launched successfully

Basic Flow:

The user launches the application, sees the splash screen, and then continues to the login screen. The user reaches the login screen and enters their credentials. The user is validated and saved through MyJohnDeere and is navigated to their home screen.

Alternative flow 1: Signup

The user cannot login because they haven't created an account. The user chooses the signup button and then is brought to the signup screen. The user has to enter first, last, email, and password. Password requirements are 6-10 characters with one number and a capital letter. If the password requirements are not met the signup process doesn't complete and an alert pops up stating the password requirements aren't met.

Alternative flow 2: Login fails

The user launches the application, sees the splash screen, and then continues to the login screen. They enter their credentials and an alert pops up stating that their username or password is incorrect.

Alternative flow 3: Still logged in

The user launches the application, sees the splash screen, and then instead of being brought to the login screen the user is brought to the last screen they were currently on in their session. The session timeout for this will be 1 hour.

Use Case 2: Vehicles (Home Screen)

Primary actor: application user

Preconditions: the application has been launched successfully and the user logged in.

Basic Flow:

The user has completed the preconditions and will be brought immediately to their vehicles screen. This screen shows all of the users vehicles in the form of a list view.

Alternative flow 1: Add vehicle

The user has completed the login process and selects the plus icon from their their vehicles (home) screen to add a new vehicle. The user is then brought to the add vehicle screen.

Alternative flow 2: Connect to current vehicle

The user has completed the login process and selects the connect button from one of their previously added vehicles in the list. The user will be prompted with a success or failure alert upon choosing connect.

Use Case 3: Add Vehicle

Primary actor: application user

Preconditions: the application has been launched successfully and the user logged in. The user selects the plus icon from their home screen to add a new vehicle.

Basic Flow:

The user has completed the preconditions and will be brought to the add vehicle screen. The user can input a vehicle's name or the connection ID from which they are already accustomed to knowing. From there they select the tractor by clicking on the choose tractor button which will show all of the possible vehicles.

Alternative flow 1: User selects connect button

The Application will attempt to connect to connect the tractor via TCP connection over wireless. If successful an alert will be shown and connect will be disabled, if not successful an alert will be shown and button will still be enabled.

Use Case 4: Augmented Reality

Primary actor: application user

Preconditions: the application has been launched successfully and the user logged in. The user selects the tractor vision icon from their home screen menu.

Basic Flow:

The user has completed the preconditions and will be brought to the tractor vision screen. The user is provided with a camera view of their surroundings ready to capture an image. The user snaps a picture and is presented with their augmented view of their vehicle.

Alternative flow 1: User selects a widget

Upon snapping the picture of the vehicle the user selects a widget of some machine information displayed. The user is brought to a new screen with tabs where they can see more information on the status of their vehicle. They can continue swiping horizontally to go through all of the widgets for that vehicle.

Use Case 5: Alerts

Primary actor: application user

Preconditions: the application has been launched successfully and the user logged in. The user selects the settings icon from their home screen.

Basic Flow:

The user has completed the preconditions and will be brought to the alerts screen. All alert information on all of the users vehicles will be displayed. The user can search through each alert to gain more information on how to service the issue for their vehicle.

Alternative flow 1: User selects service

The user will be brought to a new screen specific to the alert they selected. Based on the type of the alert the user will be provided with options and guidance for how to contact John Deere about the issue or service the vehicle themselves.

Use Case 6: Settings

Primary actor: application user

Preconditions: the application has been launched successfully and the user logged in. The user selects the settings icon from their home screen.

Basic Flow:

The user has completed the preconditions and will be brought to the settings screen. The user will be provided with a list of settings options and a logout option.

Alternative flow 1: User selects account information

The user will be brought to a new screen displaying their account information that is saved in MyJohnDeere.

Alternative flow 2: User selects application information

The user will be brought to a new screen with application information: version, date modified, creator.

Alternative flow 3: User selects logout

The user's session will expire and they will be brought back to the login screen.

Use Case 7: Bluetooth Connection to MTG

Primary actor: application user

Preconditions: the application has been launched successfully and the user logged in. The user selects the connect to tractor via bluetooth from the home screen.

Basic Flow:

The user has completed the preconditions and will be brought to the bluetooth screen. The user will have directions displayed on how to pair with their tractor. A successful pair will happen if the device name shows up correctly as the user expected.

Alternative flow 1: User pairs unsuccessfully

The user will see an alert that pops up showing that there was no device paired. From here the user can try pairing again or navigate to somewhere else in the application.

Alternative flow 2: User pairs with incorrect device

The user will pair with a device but then see an incorrect device name. The user will click on the button to unpair the device and retry to pair with their correct device.

Test Plan

Unit Testing

Access Control

In Xcode 7 you can develop with the new swift 2 language instead of objective C. Swift 2 brings some new great ways to unit test. The first is in access control. This is done by using the @testable attribute the unit test target has access to every internal entity of another module in the project. The @testable attribute allows some basic unit testing practices like the assertEquals function.

Code Coverage

There is also another great feature for testing that comes with the new Xcode 7 version- code coverage. This process is done by some simple configuration in the Xcode scheme settings and it allows the developer to run all of their unit tests throughout the workspace at once and view the results. From there the developer can click on the individual unit test case and go straight to the source code to see how many times the test was called and the number of successes or failures occurred. This in combination with access control make unit testing pretty simple in Xcode.

UI Testing

There has never been a great, built in way to test the UI in the past versions of Xcode, thankfully that has changed. The process is an expansion on Apples XCTest framework through adding three classes: XCUIApplication, XCUIElement, and XCUIElementQuery. UI testing is only available for iOS 9+. Testing the UI in addition to the unit test will cover a lot of ground for testing our application.

Testing Process

Our testing process will be carried out in three major ways: testing the database entries, unit testing, and communication testing with the MTG.

Database Testing

1. When a user account is created check the database
2. When a user adds a new vehicle check the database
3. When a user is logged out make sure the session variable is false
4. When an alert happens on a vehicle make sure the alert is stored

Unit Testing

1. Check major components of UI to make sure updating is correct
2. Check logic within the models to make sure variables are correct
3. Check that the view controllers are navigating properly

MTG Testing

1. Make sure bluetooth connection has correct name when paired
 - a. Includes correct VIN and model number
2. Test the wi-fi setup connect
3. Make sure MTG is relaying all of the data points to the mobile device

Assessment of Proposed Solution

Based on our requirements we believe our project has great potential for a good, finished product. Currently we already have major application functionality working thanks to the previous project team and we know how to move forward. Given our resources and knowledge the additional requirements we are asked to do can be done in the allotted time. Some concerns however arise with the timeline on when the new Yukon MTG will be completed with the low energy Bluetooth. We will be able to develop and test this without the actual MTG however problems may arise when put in place with the actual device. In addition the augmented reality portion of this application might not be what our client and our group hopes it to be. This is due to the augmented reality software being a new technology and not quite meant for our situation. Here is a table that better illustrates all of the strengths and weaknesses of our project solution:

| Strengths | Weaknesses |
|---------------------------------------------|--------------------------------------------------------------------------------|
| Solid code base to start from | Augmented reality software could potentially not be what we want |
| Clear understanding of project requirements | Timeline with new MTG and our testing might not go hand in hand |
| Great resources | New to iOS development |
| No budgeting issues | Testing augmented reality could potentially be difficult on objects that large |
| Timeline seems feasible | |

Table 2

Validation and Acceptance Test

Reliability

The reliability aspect of our validation and acceptance will be a combination of our testing plan in pair with other reliability testing. The first reliability testing will take place when the phone has access to cell service to see if the data is transferred to MyJohnDeere properly and how long that process takes. Other testing will be done with the MTG communication. We will test to make sure that the MTG will always be able to woken up via bluetooth and that the wireless is set up accordingly. We will also test to see exactly how far away the user can be from the vehicle to remain connected to the MTG.

Scalability

There isn't much we can do for a scalability test for our project. The only we see fit for this is to have the user perform a high number of tasks such as adding a lot of vehicles to see how the system reacts. This scenario relates more towards the application operating in dealer mode having a large number of vehicles on their account.

Usability

Our usability testing will be done through validating that our use cases our functioning properly. This means that based on the action done by the user that they're being presented with the proper screen, information, and dialogue. In addition we will check that the customer is okay with the UI as far as functionality, look, and feel.

Security

Our procedure for secure will be to demonstrate this to the customer directly. This will be done to show that the user can only connect to their added vehicles and they won't accidentally or purposely connect to another's vehicle.

Maintainability and Configurability

We will leave this up to the customer on how they will provide security patches or updates for the application. The configurability aspect will be validated by demonstration on how the customer can modify the application to better suit their needs in the future.

Resources

Team Organization

Collin Goedken, Team Leader
Brent Barth, Communicator
Ben Myshkowec, Communicator
Josh Harper, Key Concepts
Matt Koziy, Key Concepts
Jordan Harper, Webmaster
Joel DeLong, Webmaster

Project Organization

unofficial:

1. Clean up UI and make the theme similar to other john deere apps
 - a. Use John Deere skeleton templates if possible
 - b. jordan, josh
2. AR
 - a. Connect to the tractor first then take a picture and overlay machine info on the screen
 - b. More Robust
 - c. brent, matt, josh, ben, joel
3. Store data on the cloud not the phone
 - a. brent, ben
4. Have the phone be the server and the mtg be the client
 - a. collin
5. Website
 - a. matt, jordan
6. Security
 - a. joel, matt, collin
7. Testing
 - a. joel, matt, collin
8. Linking alerts to specific pages on John Deere's website
 - a. Gives quick feedback to the user
9. Other client requests

Resource Requirements

| Resource | Acquired From | Cost |
|------------------------------------|--------------------|------|
| iPhone | Personal or Client | \$0 |
| Mac with Xcode | Personal or Campus | \$0 |
| Apple Developer Account (possibly) | Client | \$99 |
| MTG | Client | \$0 |
| Wikitude Free Trial | Personal | \$0 |

Table 3

Project Schedule

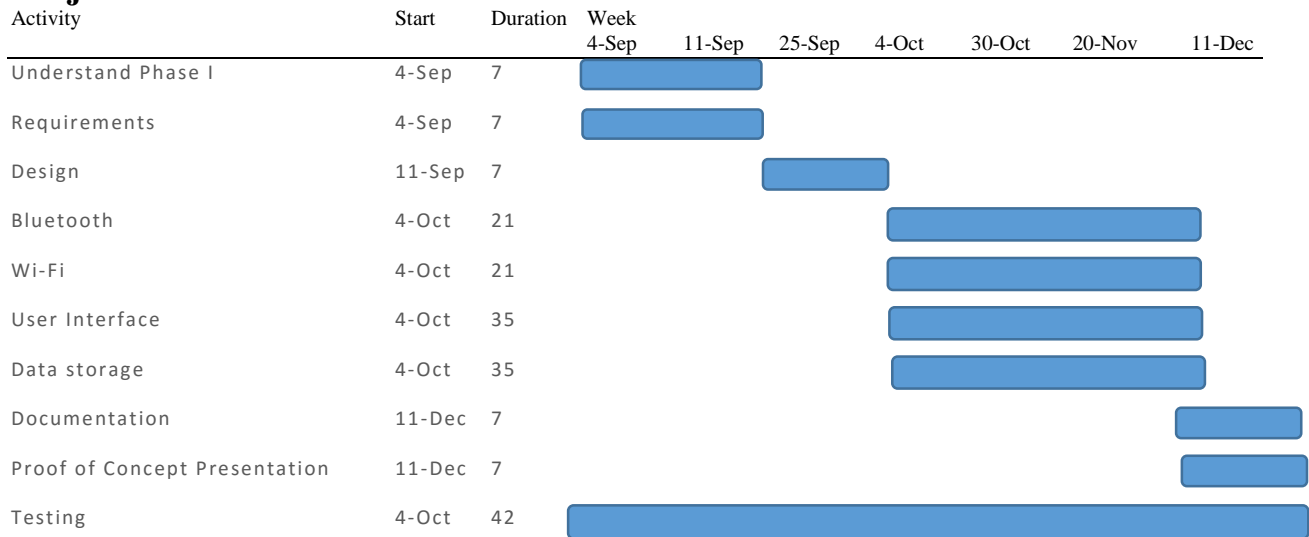


Figure 10

Overall Project Tree Diagram

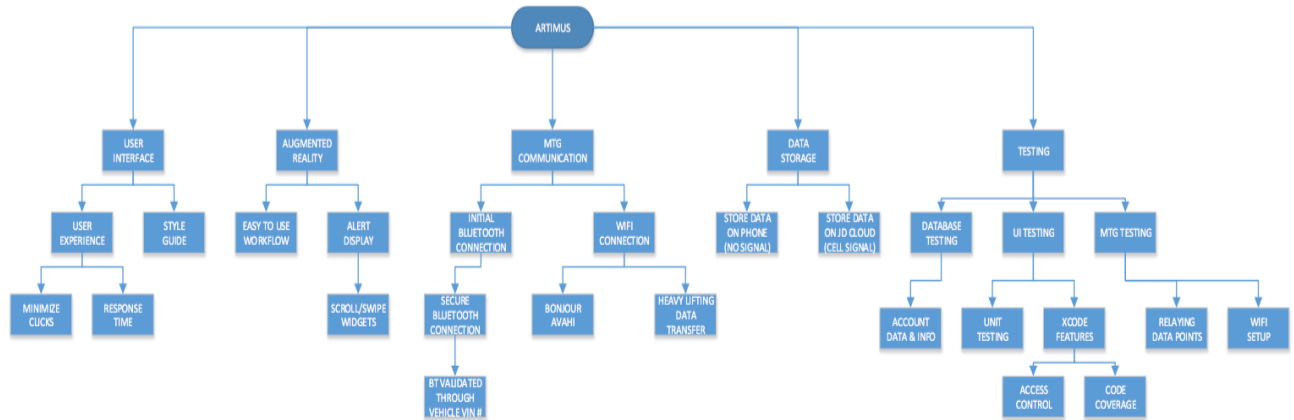


Figure 10

If image is too small go here <http://may1609.sd.ece.iastate.edu/uploads/treeDiagram.jpg>

Overall Project Completion Chart

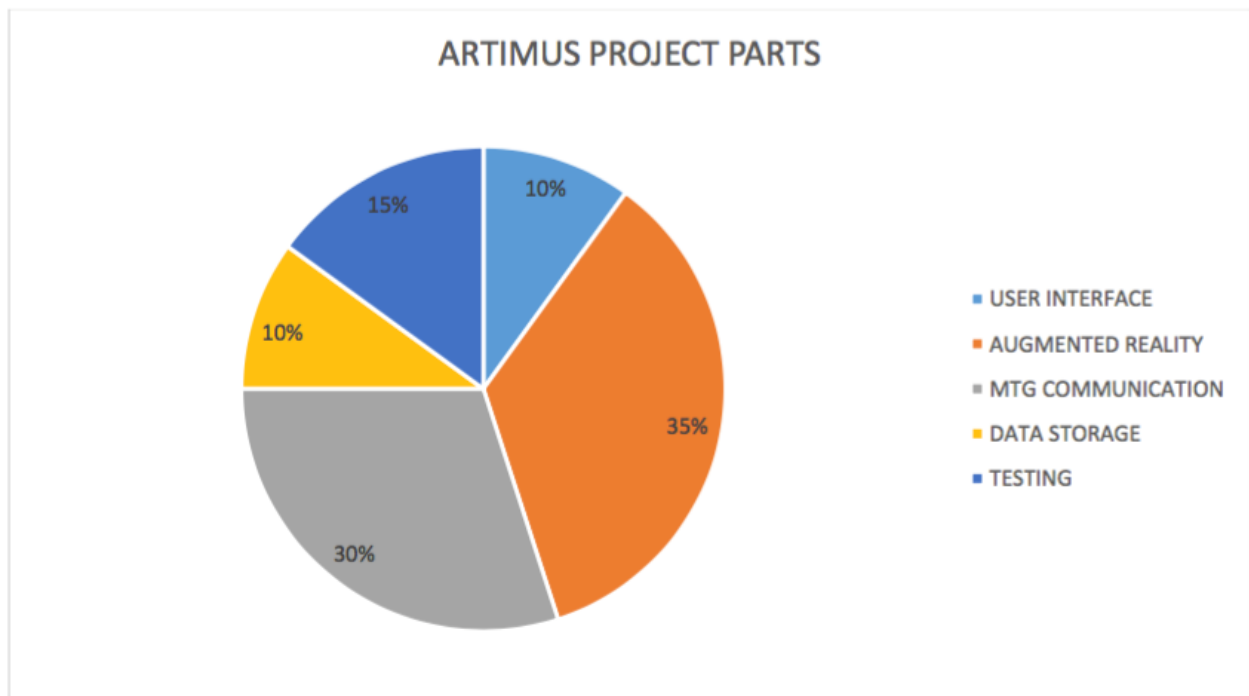


Figure 11

Risks

| Risk | Description of Risk | Action to Resolve Risk |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Data Leakage | The chance of data getting leaked over the wireless networks such as Cellular and WIFI. | Using the correct secure protocols (HTTPS) and using tested standard libraries that are available for the iOS platform should allow for us to avoid these issues. |
| Permission Usage | This application should not be asking for any permissions that are not necessary based on the requirements of the project. | Use the base requirements of the project as a standard for the only permissions needed to create the application. |
| Bluetooth Connectivity | Bluetooth could also pose a threat for data leakage. | Create secure calls that the iOS platform allows to build and test the bluetooth requirements. |
| Resources | The amount of resources to our team is not an adequate amount. We are lacking in usable software computers as well as testing equipment. | Use our resources effectively and make sure to share devices as much as possible. Take advantage of the devices provided for the project. |
| Division of Tasks | There are many tasks for this project and having a large team presents challenges allocating these tasks out to the individuals. | Creating lists of the tasks and allocating these tasks to the individuals comfortable with them. Discussion and analysis of each task will help determine which individuals are a fit for a given task. |
| Meetings & Locations | Locations for meetings and demos could be difficult with many events happening on the Iowa State campus. | Carefully planning ahead for future meetings. Keeping in contact with our advisor and client to address potential issues. If potential issues happen for campus meetings, being able to go off campus will be an option. |
| Communication | Communication between team members, advisor, and client could become an issue. | Use group messaging and create mailing lists to help distribute the communication between team members. |

Table 4

Market Literature/Survey

ISG has provided us with information that the majority of tractor owners who will be using this application, will be using iOS devices. This allows us to center our development and design around the iOS platform.

Having benchmarked augmented reality software that is compatible with iOS devices, our team feels that Wikitude provides the best tools to create a robust augmented reality application. Wikitude software contains focus for indoor/outdoor, moving objects, changing light conditions, color contrast recognition, and step by step documentation with XCode. As this software is compatible with iPhone 4 and above, majority of the user base will be able to use it.

As the previous version of this project used Vuforia for augmented reality software, we decided to also benchmark Vuforia. The pros are that Vuforia is currently implemented, functions with XCode, and its virtual buttons are implemented and working. The cons include not recognizing large objects, difficult to develop with iOS, and lack of tutorials and documentation. The fact that Vuforia lacks in being able to recognize and distinguish large objects, has been enough reason to look for better solutions.

Intellectual Property

Our team has read over the NDA and the IP agreements for this project. We obliged to these documents and honor that our project is property of John Deere. We look forward to working with the John Deere team and are setting our goals and expectations high.

Conclusion

This project is going rely heavily on the communication with the MTG, thus this is our main focus to get the application running efficiently. Next our focus is on the augmented reality workflow and design. Our goal is to make the app as easy for the user as possible, however given that augmented reality software today isn't still super robust we may have to be creative in how we incorporate it into our project. Through our documentation we believe we have a great base plan to continue and modify development for this application. All of the resources for this project are accounted for, and at the moment we do not think there will be any additional resources needed to start development. Based on our extensive research and meetings we believe to have some great ideas for the augmented reality and improved MTG connection portions for this project.

Team Information

| Name | Role | Email |
|-----------------------|---------------|----------------------------------------------------------------------------|
| Brent Barth | Communication | babarth@iastate.edu |
| Joel DeLong | Webmaster | jjdelong@iastate.edu |
| Collin Goedken | Team Leader | goedken@iastate.edu |
| Jordan Harper | Webmaster | jharper@iastate.edu |
| Josh Harper | Key Concepts | jlharper@iastate.edu |
| Matt Koziy | Key Concepts | koziy@iastate.edu |
| Ben Myshkowec | Communicator | bam@iastate.edu |
| Manimaran Govindarasu | Advisor | gmani@iastate.edu |
| Benazir Fateh | Client | fatehbenazir@johndeere.com |

References

"IOS Human Interface Guidelines: Designing for IOS." *IOS Human Interface Guidelines: Designing for IOS*. N.p., n.d. Web. 14 Oct. 2015.

"May1505 - My John Deere App - Project Documents." *May1505 - My John Deere App - Project Documents*. N.p., n.d. Web. 14 Oct. 2015.

"Try It On: Vuforia 5." *Vuforia Developer Portal*. N.p., n.d. Web. 14 Oct. 2015.

"Wikitude - The World's Leading Augmented Reality SDK." *Wikitude*. N.p., n.d. Web. 14 Oct. 2015.