* Includes: Project Plan, Design, Implementation,Testing and Validation/Evaluation, and Conclusion

* Please utilize existing materials from Project Plan and Design Document.

Cover page

Table of Contents

0. Executive Summary

1. Requirements specification (15%)

1.1 Functional Requirements:

Requirement 1 - The leaf nodes must be able to accurately take measurements of the soil moisture levels.

Requirement 2 - The leaf nodes must be able to accurately record the soil temperature.

Requirement 3 - The leaf nodes must be able to transmit data throughout the network to the home node.

Requirement 4 - The home node must be able to upload data to the web server without a wired connection.

Requirement 5 - The home node must be able to issue commands to the leaf nodes.

Requirement 6 - The network must be able to automatically handle the death of a node.

Requirement 7 - Any node must be able to communicate with any other node.

Requirement 8 - The web server must be able to store the data long term and must not lose and data in the event of a crash.

Requirement 9- The web server must be able to serve data to clients using modern web browsers.

Requirement 10- The web server must allow users to only view their own data with login credentials.

Requirement 11- The gateway node must be able to communicate with every node.

Requirement 12- The sensor nodes must have a sleep cycle where they only wake up during certain times in the day in order to save power.

1.2 Ethical Considerations

If we are successful in making our leaf nodes very cheap, it is unlikely our users will go out and retrieve them after their batteries have died. If the leaf nodes are not biodegradable they will remain where they died for many centuries to come as litter. We must keep this in mind when determining the price and biodegradability of the nodes.

1.3 Non-Functional Requirements

Reliability - The web server must be able to handle transient overload without losing more than 5% of the data.

Availability - The web server must be running 98% of the time.

Reliability - The home nodes and leaf nodes must immediately run our programs on startup. They can't have any configuration required on startup.

Security - Our user's personal information must not be obtainable by non-authorized parties.

Security - Only administrators will be allowed to modify non-user data.

Reliability - Nodes must be able to handle transient overload without losing more than 5% of packets.

Reliability - We need to make sure that the web server won't lose any points of data under times of both low and high data loads. To determine whether this is true or not, we can ask all nodes to simultaneously generate a data packet and see whether the data received by the web server is complete.

Availability - We need to make sure that the web server runs at all times. We can run the web server for several months and examine the amount of downtime the server had over that time.

Reliability - The home nodes and leaf nodes run our programs on startup immediately. We can turn off and turn on the nodes to determine the reliability of the nodes to make sure that there is not any configuration required on startup.

Security - We need to make sure that only administrators are allowed to modify data. We can try to login with non-administrative accounts and see if we can access and modify only the data that account is allowed.

Data Transmission - It’s necessary for us to make sure that sensor nodes are able to transmit messages to the gateway nodes. To determine that we can transmit the data from the nodes and observe whether these data can be received by the PC successfully and completely.

1.4 Previous Work And Literature

To get an understanding of how we should implement the routing protocol for our mesh network, we looked into the strategies employed by a company that designs components for mesh networks called Zigbee. For their routing protocol, Zigbee uses Ad Hoc On-Demand Distance Vector (AODV)[9][10]. This protocol works by having each node store paths to other nodes in the network in a routing table. If a node needs to send a message to another node, it first checks if the route to that node is in its routing table. If it exists, then that route is used. If not, the algorithm uses a flooding technique to discover the route. This algorithm is one of the most efficient ways to communicate in a mesh network that we currently know of.

While analyzing AODV, we found that we do not require some of the functions this algorithm supports. For example, this algorithm allows any node in the network to communicate with any other node. In our solution however, we only need messages to go in one direction to the gateway node. Building this simplification into our protocol make it much easier to implement because instead of keeping a routing table with many routes, we only need to store the one route to the gateway. We decided to use similar strategies to AODV for route discovery and maintenance only built in complexity for what our solution requires. The simplifications we found make our protocol easier to implement and more efficient than a full implementation of AODV in our circumstance.

The devices in figure 2 are soil moisture sensors that are commonly used by farmers to find the moisture levels in their fields. These sensors calculate soil moisture by using the relationship that electrical resistance increases as the soil moisture decreases. We learned from this sensor that we can use the same properties it relies on to construct our own soil moisture sensor out of gypsum.

Figure 2. *Watermark soil moisture sensors*

At a price of around $50, the watermark sensor is affordable for many industries to use, but using it in our solution would be very expensive. Because water is an electrical conductor, the resistance between the wires is inversely proportional to the soil moisture (José O. Payero, et al.) [6]. We will use this property to create our own gypsum-based sensor instead of the Watermark sensor. We estimate that it will cost around about $1 to manufacture these sensors. This 98% reduction in price compared to buying Watermark sensors significantly lowers the cumulative price of the solution.

2. System Design & Development

2.1 Constraints Considerations

2.1.1 Standards

Our project contains systems with standards we must adhere to. Such standards are HTTPS, for communication with web server, SQL, when querying database, and IEEE best practice standards for writing c++. Most of these standards will be fairly easy to adhere to because the tools we are using automatically use them. However, the most difficult to follow will be the coding standards for c++. To make sure that this standard is followed, we will all become proficient in the standard and perform code reviews on each other’s code.

3. Implementation

3.1 Interface Specifications

Humidity sensors need to be able to detect and measure the humidity in a wide range, roughly between 20% and 90%, and that's sufficient to indicate the water requirement for the soil. Calibrations needs to be done on the sensors to determine the lower and the upper level of humidity the sensor can detect accurately.

The sensor nodes should be able to communicate with each other by radio signals. The sensor node should be able to interface with two other sensor nodes by using an Arduino microcontroller and an Atmega328P chip on a PCB which means if one of these two sensor nodes is broken or out of the range, the sensor node can still send the data to the other one. The gateway will send the data to the web server by 3G connection where the users can see the application and the data from any web browsers.

3.2 Hardware and software

Hardware:

- Arduinos - Microcontroller that is easy to program and connect with external devices.
- Radio Transmitter Receiver - specific hardware still undetermined must have at least 1km range.
- Hygroscopic Moisture sensor - Will be a gypsum based sensor because gypsum is cheap and hydroscopic.
- Adafruit Cellular Link - Connects the gateway to the internet through a cellular connection.

Software:

- NRF24 Arduino Library - Manages low level communication over radios
- Laravel - Simplifies web development
- MySQL - Stores our data safely with fast retrieval times
- PHP - Allows for custom web pages
- Java - Creates backend API
- Arduino IDE - Allows for easily programming our Arduinos.

4. Testing, Validation, and Evaluation

4.1 Functional Testing

Several tests needs to be conducted in our project, and theses includes the following:

1. Calibration of humidity sensor.
2. Test routing protocol by placing many sensor nodes in a field and recording how many nodes are able to accurately get their data to the gateway.
3. Test longevity of sensor nodes by doing a full scale test for a month and measuring the amount of energy left in the batteries by the end.
4. Determine the need of Polystyrene to protect the Gypsum and to stabilize the sensor.
5. Preliminary test for the accuracy of signal transmitted. (One-to-one)

4.2 Non-Functional Testing

Data tables which are collected and transmitted from nodes and CAD annotated drawings of the hydroscopic probe will be submitted. According to different types of data, like temperature under different time periods or humidities in different fields, these data are all going to to be finally created to drawings and reports. In order to make sure the documentation security and usability, the system requires pin numbers and keeps checking and reporting the results to the user.

5. Project and Risk Management

5.1 Task Approach

Project tasks were assigned for each month to track the progress for the project plan.

Semester 1

	September	**Oct**	**Nov**	**Dec**

Nodes	Nodes are functioning fully as they were by the end of last semester. We are fully up to speed with where the project left off	Has final radio with range. If new algorithm is needed, it is implemented. Data transmission from nodes has good reliability.	System if fully developed and is able to function as expected. Communication between components is reliable.	Testing and bug fixing
Homenode	Is capable of both sending data to the web server and receiving data from the nodes	New routing protocol is implemented and reliable for system.	System if fully developed and is able to function as expected. Communication between components is reliable.	Testing and bug fixing
Web Server	Is up and running and able to communicate with home node.	If additional development is required on the website, it is mostly complete.	Communication between components is reliable.	Testing and bug fixing

Semester 2

	January	February	March	April
Sensors	Exploring and implementing new types of data sensors.	All sensors are equipped with the final sensors and hardware that would be used in production.		Finalize the documentation, do bug fixes, make the code more extendable ect.
Web Server	Editing format of web server to display additional sensor data	Additional development to make UI more user friendly. Also make more API endpoints if needed	System if fully developed and is able to function as expected. Communication between components is reliable.	Finalize the documentation, do bug fixes, make the code more extendable ect...
Casing	Construct casing in CAD. May enlist the help of an ME.	Build an initial prototype of the casing.	Stress test the casing in the real environment.	Finalize the documentation, do bug fixes,

			Make more cases for more node prototypes	**make the code more extendable ect...**

5.2 Possible Risks And Risk Management

Reliability in designing the project components is vital. If the radio signal transmitted and received between the nodes interferes with unpredicted objects, that might cause issues in the data received.

When we start our large-scale test in March, the ground of the fields could be frozen making it so that we cannot record soil moisture data. This would delay the start of our tests and possibly make it so that our system is not fully tested by the end of the school year. Our mitigation strategy will be to improve our testing efficiency if this occurs and to hope that it does not.

5.3 Project Proposed Milestones and Evaluation Criteria

The mesh network will be able to ferry messages from any member of the network to the gateway. This will indicate that the routing protocol is complete. This milestone can be verified by performing a large-scale test and ensuring that the gateway is receiving data from each sensor node during each wake cycle.

The home node can receive data from leaf nodes and upload these data to the web server. This can be evaluated by watching the traffic that is received by the home node and ensuring that it is getting all messages that are intended for it.

The gypsum soil moisture sensor has been tested and successfully implemented on a sensor node. This milestone can be verified by by comparing the values that our soil moisture sensor produces to the values of a few commercially available products.

We will have tested our system on a scale of around 50 sensor nodes. We will be testing the system for the reliability with which packets traverse the network, the power usage of each node in the network, and the amount that sensor nodes become out of sync with each other. This milestone can be verified by inspecting our report which will contain the results from this test.

5.4 Project Tracking Procedures

We will have weekly meetings and periodic presentations using powerpoints to make sure that everybody in our team is on track and that we're not behind the schedule to finish the project.

6. Conclusions

Closing remarks for the project

- There is potential for this to be a powerful, effective, and affordable system if a little more work is done.
- It would be relatively easy to modify our work to allow the network to record other types of data, for different applications.
- We have set up a good framework for future development on this project.

Future work (potential directions)

- Automated deployment system
- Significant extensions to front-end
- Long term tests of system
- Power optimization strategies
- Custom Casing for leaf nodes

7. List of References

1. Cost estimation ordering resources, https://www.digikey.com, *Digikey Electronics*
2. *"Project Document"* and *"Project Plan"* from May19-45, published by Group 45 in May- 19.

8. Team Information

- Tyler Fritz - Gateway and Sensor nodes
- Ahmed Abuhjar - Gateway and Sensor nodes
- Dong Xing -Gateway and Web server
- Haoyue Ma-Gateway and Web server
- Yuanzhe Wang-Gateway and Web server

Ahmed Abuhjar	Haoyue Ma	Tyler Fritz	Dong Xing	Yuanzhe Wang
1. Design and implement sensor nodes 2. Testing and calibrating the sensors 3. Ensure the accuracy of data transmitted to the home node. 4. Design and implement PCB for the leaf nodes	1. Connect the circuit for data transmitting from the gateway to the web server. 2. Program and test the code needed for data transmitting to the web server.	1. Design and implement routing protocol for our mesh network. 2. Ensure that the code running on the sensor nodes uses minimal power. 3. Choosing hardware components for sensor nodes.	1. Test and improve the accuracy of data transmitted between gateway and to web server. 2. Create understandable & readable flow charts according to the data which the web server received for users to check 3. Set the security account number for the web server	1. Develop and debug the code for collecting data and transmitting to the web server. 2. Make sure the data transmitted to the web server is accurate.