Advanced Green House Monitoring and Controlling System

Project Team

Nivethitha R (20MZ05)
Sivaranjani M D (20MZ07)
Shanmuhappriya M (20MZ33)

Table of Contents

TABLE OF CONTENT

1 INTRODUCTION

We live in a world where everything can be controlled and operated automatically, but there are still a few important sectors in our country where automation has not been adopted or not been put to a full-fledged use. One such field is agriculture. Agriculture has been one of the primary occupations of man since the early civilizations and even today manual interventions in farming are inevitable. Greenhouse is important part of Agriculture and Horticulture, as it can be used to grow plants and crops under controlled climatic conditions for optimal yield.

1.1 PURPOSE

This Software Requirements Specification (SRS) provides a detailed description of all the functions, product perspectives, features, specifications, external behaviors, design constraints, interface requirements, user characteristics, requirements (function and non-functional)and other factors necessary to provide a complete and comprehensive description of the proposed AGHMCS.

This document is primarily written for the stakeholders, such as the developers, programmers, integrators, testers and maintainers of the proposed system.

1.2 SCOPE

The scope is to design a automatic monitoring and controlling system which is a simple, secure, easy to install, microcontroller-based circuit to monitor and record the values of temperature, humidity, soil moisture and sunlight of the natural environment that are continuously modified and controlled in order optimize them to achieve maximum plant growth and yield. To access the greenhouse appliances from anywhere. Also to provide security to the greenhouse. It focuses on saving water, increasing efficiency and reducing the environmental impacts on plants production, improves security and accessibility. The user can see the atmospheric conditions of the greenhouse plants on mobile phones and control the greenhouse from faraway places. It is to increase the production of food stuff.

To save water, power etc. to increase the productivity and security of Greenhouse.

1.3 DEFINITIONS, ACRONYMNS, ABBREVIATIONS

Term	Description
AGHMCS	Advanced Green House Monitoring and Controlling System
DB	Data Base
FP	Finger Print
GH	Green House
GHO	Green House Owner
IEEE	Institute of Electrical and Electronics Engineering
LCD	Liquid crystal display
SMS	Short Message Service
SQL	Structured Query Language

1.4 REFERENCES

1.	IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications
2.	Heinzelman, WRH, Chandrakasan, AC, & Balakrishnan, HB. (2020). Energy-efficient communication protocol for wireless microsensor networks
3.	Ahonen, TA, Virrankoski, RV, & Elmusrati, ME. Greenhouse monitoring with wireless sensor network.
4.	Greenhouse guide. [Online]. Available: http://www.littlegreenhouse.com/guide.shtml
5.	M. Mancuso and F. Bustaffa, "A Wireless Sensors Network for Monitoring Environmental Variables in a Tomato Greenhouse," presented at 6th IEEE International Workshop on Factory Communication Systems in Torino, Italy
6.	H. Liu, Z. Meng and S. Cui, " A Wireless Sensor Network Prototype for Environmental Monitoring in Greenhouses," presented at Wireless Communications, Networking and Mobile Computing 2017 (WiCom 2017), International Conference on 21-25 Sept. 2017 Page(s): 2344 – 2347

2. Overall Descriptions

2.1. Product Perspective:

In this project, there are three main tasks to be accomplished. Firstly, programming the sensor nodes, then programming the gateway, where all the information is gathered from the sensor nodes, and lastly informing the manager of the GH with the information at the gateway, via both a user friendly web site and sending SMS.

2.1.1 Programming the Sensor Nodes:

It involves periodically measuring the temperature, light and humidity levels and sending these measurements to the gateway. However, this task should be completed in such a way that the sensor nodes use least possible energy. Thus, in this task, we will be implementing power conservative routing algorithms for sensor networks .

2.1.2 Programming the Gateway:

This task consists of three major operations. Firstly, if there is an error in the data coming from the sensor nodes, fix it. Then, consolidate the data. Lastly, save the data to an SQL database on the computer which the gateway is connected.

2.1.3 Develop a web application:

This application will provide a user friendly interface to the GHO, such that the manager will be able to see meaningful information about his/her GH almost instantly. The manager will also be able to see the past information and the comparisons of them with the current information.

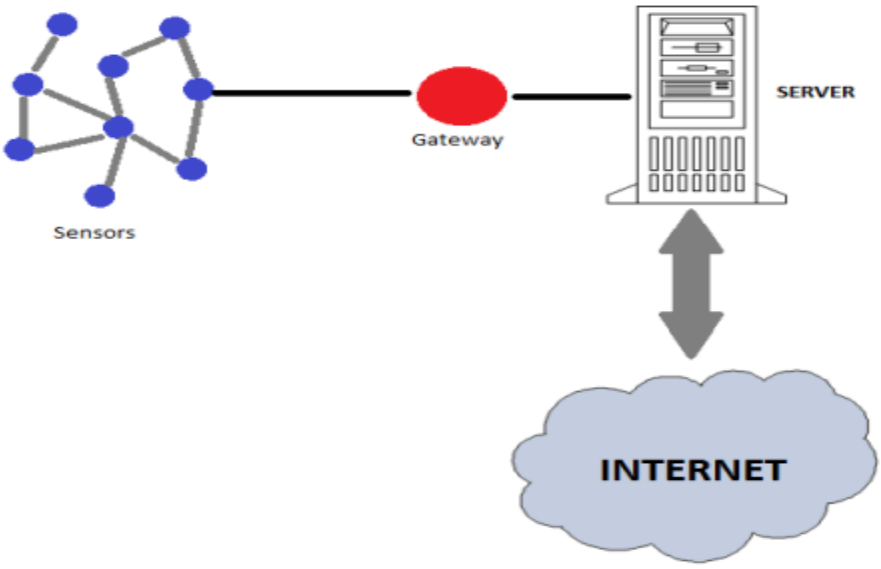


Figure 1 - Wireless Sensor Network

Star Topology:

A star topology is a topology for a Local Area Network (LAN) in which all nodes are individually connected to a central connection point, like a hub or a switch. The central hub is usually a fast, self contained computer and is responsible for routing all traffic to other nodes.

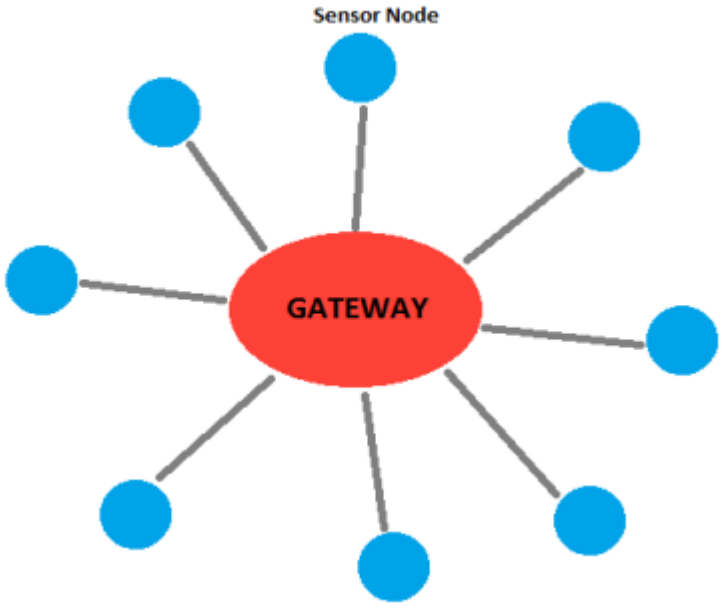
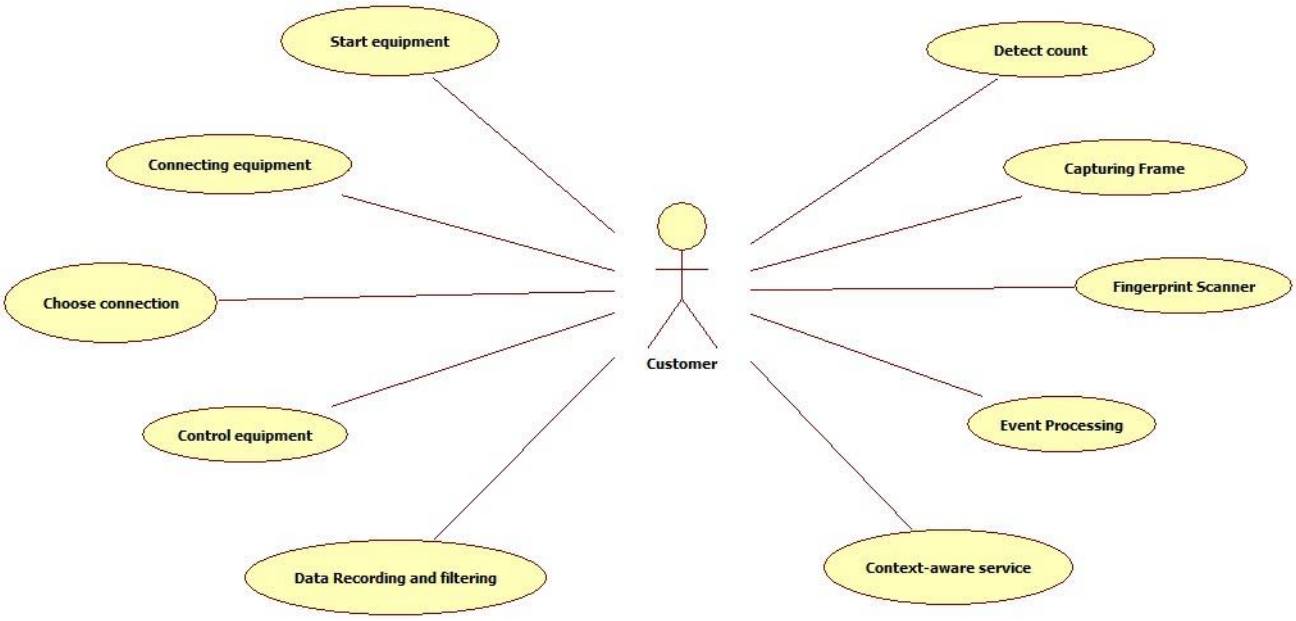


Figure 2 - Star Topology

2.2 PRODUCT FUNCTIONS



2.2.1 Start Equipment:

Title: Start Equipment

Description: User must have corresponding application to access the greenhouse monitoring system. Also all the equipment are well connected to the microprocessor to start the system.

Actor: GHO

Pre-conditions: User must have the corresponding application installed

Post conditions: Equipment started.

Basic flow:

1. The admin opens the GH monitoring and controlling application in his mobile

2. By opening the application, he can monitor and control the appliances of Greenhouse.

2.2.2 Connecting equipment

Title: Connecting equipment

Description: The Bluetooth module and the ESP8266 module in the system in powered ON.

Actor: Arduino

Pre-conditions: Pin connections should be done already.

Post conditions: Connection established successfully.

Basic flow:

1. The user should open the app.
2. Connection is established by clicking the connect option in the application menu

2.2.3 Choose connection

Title: Choose connection

Description: The system has the ability to have choice of using Bluetooth or Wi-Fi.

Actor: Greenhouse owner

Preconditions: The Bluetooth module and the ESP8266 module in the system in powered ON.

Post conditions: After successful connection the user (GH owner) can control the equipment.

Basic flow:

1. The Greenhouse owner can choose the connection mode.
2. If it is Bluetooth connection, the user can pair the devices

2.2.4 Control equipment

Title: Control equipment

Description: The user can control the appliances through the app. The Arduino (microcontroller) controls the equipment.

Actor: GH owner, Arduino

Pre-conditions: User must have the application pre-installed and all the connections have been made already

Post conditions: The changes are made in the GH and the notification is sent to the user (GH Owner).

Basic flow:

1. User must select the needed option in the application

2.Then he can control the greenhouse appliances

2.2.5 Data Recording and filtering

Title: Data Recording and filtering

Description: Sensors carry out data abstractions in real time.

Actor: GH owner, sensors

Pre-conditions: The sensors record the atmospheric values.

Post conditions: The values are converted from analog to digital signals.

Basic flow:

1) Data recording functionality is where the Sensors carry out data abstraction in real time.
2) The sensors record temperature, light, humidity and soil ph.
3) The values are converted from analog to digital values.

2.2.6 Context-aware service

Title: Context-aware service

Description: It specifies numerical values which is set as the threshold for certain data recorded such as temperature or light.

Actor: GH owner, sensors

Pre-conditions: The Threshold value is pre-defined.

Post conditions: The actuators control the GH

Basic flow: Specifies numerical values which is set as the threshold for certain data recorded such as temperature or light.

2.2.7 Event Processing

Title: Event Processing

Description: Event processing service functionality involves having IOT analytics where the GHO can view the trends of the environmental parameters.

Actor: GHO

Pre-conditions: The sensor should gather the environmental details.

Post conditions: Will receive e-mail notifications when the parameters surpass the set threshold.

Basic flow: It involves having IOT analytics where the GHO can view the trends of the environmental parameters.

2.2.8 FP Scanner

Title: FP Scanner

Description: Scans the employee's FP who enters the GH.

Actor: Employees

Pre-conditions: Employee FP should be stored in the database.

Post conditions: Door should be accessed as per constraints.

Basic flow:

1) The FP is scanned.
2) If the FP is matched with the existing one, then the servo motor is turned on.
3) The gate gets open for the employee.

Alternate Flow: If the FP is not matched for more than three times, then the person is not allowed in.

2.2.9 Capturing Frame

Title: Capturing Frame

Description: Capture frames is the functionality which involves camera for detecting the number of people entering the GH.

Actor: camera

Pre-conditions: The camera should be placed at the door.

Post conditions: The image is captured in camera.

Basic flow: The camera captures the image in RGB and convert the image into gray scale and forms the boundary box.

2.2.10 Detect people count

Title: Detect count

Description: Count details sent to the owner.

Actor: Greenhouse owner

Pre-conditions: The count limit should be pre-defined.

Post conditions: The door is accessed.

Basic flow:

1) It computes the current entry of the people inside the GH.
2) The information viewed by the GH owner.

2.4 CONSTRAINTS

2.4.1 To allow the people inside the greenhouse, the limit (human entry limit) is already set.

2.4.2 To automatically open the greenhouse door for the employee, a condition is set (i.e.,) The fingerprint should match with the existing fingerprint in the employee profile database.

2.4.3 The door should not be opened if the fingerprint does not match and the greenhouse owner should be notified with the alert message only if the scanner error count exceeds three times(error>=3)

2.4.4 The threshold value for each sensor is already pre- defined in the Arduino uno sketch(code). The sensor senses the environmental condition, simultaneously checks with the pre-defined threshold value to automate the appliances

2.5 ASSUMPTIONS AND DEPENDENCIES

2.5.1 It is assumed that the Arduino uno (microcontroller) is connected or powered up with the efficient battery and the code/sketch is fed into the Arduino board already.

2.5.2 It is assumed that the appliances to be controlled are connected with the micro controller.

2.5.3 It is assumed that the employees of the greenhouse are already registered with their fingerprints attached with their profile database.

2.5.4 It is assumed that the number of people inside greenhouse is restricted to only 500 people. That is, (people count=500)

Based on these assumptions and dependencies the system is drawn.

3.1 EXTERNAL INTERFACE REQUIREMENTS

3.1.2 Hardware Interfaces

3.1.2.1 Workstation

3.1.2.1.1 Customer workstation shall have keyboard input.

3.1.2.1.2 Customer workstation shall have roller ball mouse input.

3.1.2.1.3 Customer workstation shall have minimally a 19-inch monitor.

3.1.2.1.4 All interfaces on customer workstation shall be scaleable in information display size.

3.1.2.1.5 Customer workstation shall have appropriate hardware for network connection.

3.1.2.1 Sensors

3.1.2.1.1 A. Arduino Microcontroller Arduino has a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) which runs on the computer and enables it to write and upload the code to the board [11].

3.1.2.1.2 B. Liquid Crystal Display (LCD) LCD (liquid crystal display) is the technology used for displays in a notebook and, other smaller computers[12].

3.1.2.1.3 C. Temperature and Humidity Sensor The sensor used in this project is the DHT22 sensor which is a low-cost sensor used for the measurement of temperature (-40 to 80°C) and humidity (0-100%). Light Intensity Sensor A Light Intensity Resistor(LDR) is a component that has a (variable) resistance that changes with the light intensity that falls upon it and LDR is are also called photoresistor. This allows them to be used in light sensing circuits [13].

3.1.2.1.4 D. CO2 Sensor Gas Sensor (MQ2) is used for spotting gas leakage in an environment.

3.1.2.1.5 E. Fan This serves as the coolant for the Greenhouse monitoring and control system.

3.1.2.1.6 F. LED They serve as an indicator of the Greenhouse monitoring control system.

3.1.3 Software Interfaces

3.1.3.1 Customer workstation shall have the latest ARDUINO IDE installed.

3.1.3.2 Customer workstation shall be internet capable with at least one internet browser available.

3.1.3.3 Device shall be internet capable with an internet browser.

3.1.4 Communications Interfaces

3.1.4.1 The next step in implementing is interfacing of wifi module with the Arduino. This is done to access the sensed data from the sensors to the cloud. ESP8266 wifi module is low cost standalone wireless transceiver that can be used for end-point IoT developments.

3.1.4.2 ESP8266 WiFi module enables internet connectivity to embedded applications. It uses TCP/UDP communication protocol to connect with server/client. Microcontroller communicates with the module using a set of AT commands. Microcontroller communicates with ESP8266-01 WiFi module using UART having specified Baud rate.

3.1.4.3 ThingSpeak is a software that can monitor our data over the internet from anywhere, and we can also control our system over the Internet from anywhere, and we can also control our system over the Internet, using the Channels and webpages provided by ThingSpeak.

3.1.4.4 ThingSpeak ‘Collects’ the data from the sensors, ‘Analyze and Visualize’ the

data and ‘Acts’ by triggering a reaction. Working of this project is based on serial communication for fetching data from the sensors.

3.1.4.5 First Arduino sends a start signal to sensor and then it gives a response signal with containing data. Arduino collects and extracts the data and then sends it to ThingSpeak server. ThingSpeak displays the data in the form of graphs.

3.2 FEATURES

3.2.1 Interfacing features

3.2.1.1 Workstation input feature [1].

3.2.1.1.1 User shall interact with the workstation via mouse / keyboard .

3.2.1.1.2 System shall have large lettered keyboards.

3.2.1.2 Mobile Device input features

3.2.1.2.1 Customer mobile device shall use its pre-existing hardware interfaces such as USB connection, network connection, or other features that are applicable to the product.

3.2.1.3 Controlling features

3.2.1.3.1 User can control the greenhouse from remote areas with the help of values read from the Thingspeak to the application or web browser.

3.2.1.3.2 The user can on the fan or turn on water pump or switch on LED light whenever necessary.

3.2.1.4 Workstation output features

3.2.1.4.1 System shall have large letter text and screen symbols.

3.2.1.4.2 It senses the values from the sensor and it is given to the Thingspeak app for checking the threshold value

3.2.1.5 Mobile Device output features

3.2.1.5.1 System shall mimic the standard output functioning of particular style device.

3.2.1.6 Maintenance

3.2.1.6.1 System shall support standards for good plant growth.

3.2.1.6.2 Customer shall be able to perform maintenance through traditional interfacing.

3.2.1.6.3 Maintenance system shall be able to interface with command controls.

3.2.1.7 Security

3.2.1.7.1 Customer shall be able to function the security system through fingerprint interfacing.

3.2.1.7.2 Security system shall be able to interface with fingerprint matching.

3.3 PERFORMANCE REQUIREMENTS

3.3.1 The scanning of FP shall have a delay of no longer than two seconds. It takes less than two seconds to scan the FP , then no longer than three seconds to scan the FP that is stored in the DB.

3.3.2 Even though there are many number of employee's FP stored in the DB , it has very good storage capacity.

3.3.3 The system shall not require more than two seconds to detect the people count and shall not have the delay of more than two seconds to display it in LCD.

3.3.4 The system shall not take more than a second to send alert message to the GHO.

3.3.5 System confirmation of a requested action from the user in the AGHMCS environmentshall also occur in real time.

3.3.6 The system take the less than a second to start home automation or smart irrigation module.

3.4 DESIGN CONSTRAINTS

3.4.1 Customer Scalability

Only the employees, the owner and the regular customer have the access to enter into the greenhouse with the help of fingerprint identification.

3.5 SOFTWARE SYSTEM ATTRIBUTES

3.5.1 Reliability

- It is reliable in terms of data accuracy and time accuracy.
- Data accuracy – Checked to make sure sensors do not send wrong data.
- Time accuracy – achieved by programming the hardware and sensor nodes.

3.5.2 Availability

- The web interface will always be available.
- For network to be available, batteries of sensors should be replenished when 50% of the network is unavailable.

3.5.3 Security

- The only security constraint will be user information security.

- when the user does not remember his log in password, a request for a new password can be made.
- Since an attack to the greenhouse network is not expected we do not need to implement security constraints on the network.

3.5.4 Maintainability

- After some time, the batteries of sensors will run out. Batteries should be replaced yearly.

3.5.5 Portability

- Though there are portability issues with embedded programming, the code written for sensors will be mostly portable. There are no issues with the other parts of the project.

3.5.6 Performance

- The project mostly focuses on energy consumption and availability of the network as a whole. Therefore, the performance is not the most important constraint. Performance is mostly not considered for the sake of power saving.

3.5.7 Correctness, Understandability, Reusability.

3.6 OTHER REQUIREMENTS

All the requirements mentioned above are enough for AGHMCS.