CT5061 Internet of Things

Table of Contents

[Table of Figures 2](#_Toc211502024)

[Introduction 3](#_Toc211502025)

[Design and Specification 4](#_Toc211502026)

[System Overview 4](#_Toc211502027)

[System Requirements 4](#_Toc211502028)

[Ethical Considerations 6](#_Toc211502029)

[Data Privacy and Security 6](#_Toc211502030)

[Environmental Impact 6](#_Toc211502031)

[Accessibility and Equity 6](#_Toc211502032)

[Reliability and Responsibility 6](#_Toc211502033)

[Labor impacts 7](#_Toc211502034)

[Hardware and Software Used 7](#_Toc211502035)

[Hardware 7](#_Toc211502036)

[Software 8](#_Toc211502037)

[Hardware Design 9](#_Toc211502038)

[System Layout 9](#_Toc211502039)

[Software Design 10](#_Toc211502040)

[Sensing Modules 10](#_Toc211502041)

[Data Processing Modules 10](#_Toc211502042)

[Communication Modules 10](#_Toc211502043)

[System Development 11](#_Toc211502044)

[Software Development Process 11](#_Toc211502045)

[Hardware Development Process 11](#_Toc211502046)

[System Integration 11](#_Toc211502047)

[System Testing 12](#_Toc211502048)

[Software Unit Testing 12](#_Toc211502049)

[Software Application Testing 12](#_Toc211502050)

[Hardware Unit Testing 12](#_Toc211502051)

[System Testing 12](#_Toc211502052)

[Appendix 13](#_Toc211502053)

[References 13](#_Toc211502054)

## Table of Figures

**No table of figures entries found.**

# Introduction

Managing the health of household plants can quickly become tedious and time expensive, this difficulty is amplified when having to manage other household chores and other responsibilities. More industrial uses of crop growth are highly repetitive and tedious, meaning that they can likely be partially or fully automated. These problems create a demand for a smart and connected agricultural management system.

# Design and Specification

## System Overview

The agricultural management system must manage key environmental factors surrounding the crop’s health. These factors include:

* Soil Moisture and pH
* Soil nutrient content
* Temperature
* Light intensity

The aim of this system is to provide the crop with the most optimal environment for growth and sustainment. Given this aim the system must monitor the state of the factors listed above, it must then process, predict and take action to correct deviations from optimal conditions

The system must also monitor its own health and its ability to support the life of the crop. If a system fault is detected then the system must attempt to resolve the issue, if it fails to resolve the issue then it must alert the user(s) of the fault and provide instructions on how to fix the issue.

## System Requirements

The system can be broken down into a set of modules. Each module contains a sensor(s) that feeds data into a processing module which then provides actions to an actioning device or an alert module.

A processing module in this document refers to a piece of software with inputs and outputs that runs on a microcontroller.

An actioning device performs corrective actions, e.g., a fan activating when temperature rises.

An alert module is a generic term for a piece of software and or hardware that is designed to alert the user(s) of any critical information.

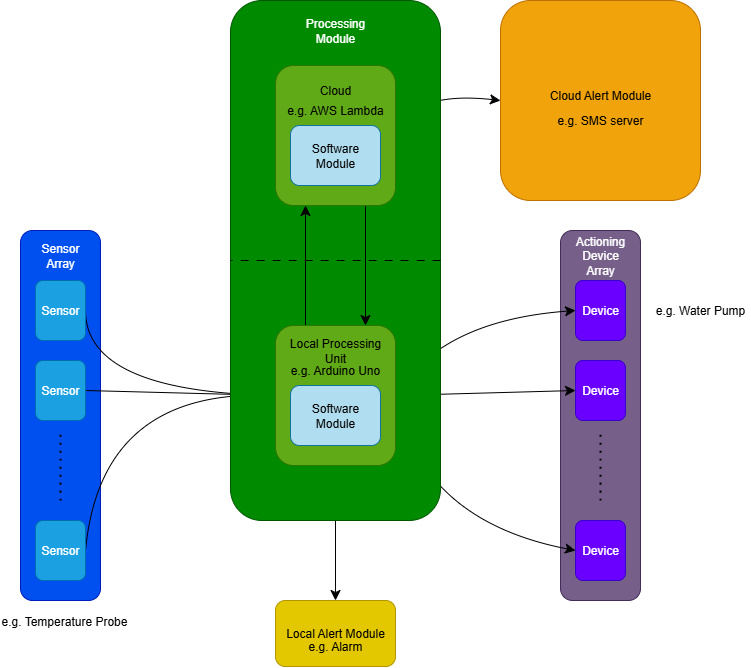


Figure 1 Diagram depicting the components of a module and the flow of data through components

Using the above definition of a module, the following modules can be determined as necessary, in order of priority.

1. Soil Content Management Module
2. Light Intensity Management Module
3. System Monitoring Module
4. Temperature and Humidity Management Module

Household applications of the system would typically involve the system being small scale, energy efficient and cost effective. Industrial applications likely require the system to be robust, modular and highly automated. This poses a problem as the system cannot meet all the requirements at the same time, thus the system must be designed with part interchangeability in mind, this would allow the maintainer of the system to easily modify the system to meet any users requirements.

The most effective solution to making the system interchangeable would be to physically separate most components of the system and program them to use common communication protocols allowing for them to be easily interchangeable.

## Ethical Considerations

### Data Privacy and Security

Sensor data may be used by malicious actors to predict user behaviour patterns or business processes.

In the case of behaviour patterns temperature data can be used to predict when users turn off their heating and thus when they will likely be out of the house.

In the case of business processes the sensor data can be used to predict whether the business has had successful crop growth and thus could be used as part of insider trading.

Given the above threats it is important to ensure that appropriate security measures are in place, in particular securing the communications between sensors and processing modules, and between processing modules and actioning devices.

### Environmental Impact

Manufacturing the system can harm the environment through CO2 emissions, reducing this impact would be ideal and thus components of the system should be sourced from sustainable providers.

### Accessibility and Equity

To ensure all ranges of users can access and use the system the system should be designed to minimise cost and complexity, reliability should also be considered to reduce the cost of maintenance.

A range of components should be supported to allow users in different scenarios to be able to use the system. For example, growing crops in a desert requires the system to be resistant to sand damage.

### Reliability and Responsibility

Automated systems introduce the risk of mistakes being made that go unaccounted for. It’s important to design the system with reliability in mind to make sure the system continues to operate correctly, it should also be made clear to users that someone should be assigned the risk owner of the system so if something were to go wrong with the system and crops were damaged because of the fault then the risk owner would be held accountable.

### Labor impacts

A large-scale automation system could reduce the need for human labour; this could lead to job loss which would harm those who lost their jobs. It should be made clear to user(s) of the system that not only a risk owner should be assigned but also system maintainers should be allocated thus reducing the impact of large-scale job loss due to automation.

## Hardware and Software Used

### Hardware

#### Central System Management Server

As the primary interface between the microcontroller system and other systems and the cloud I decided to go with the [Raspberry Pi Zero 2 W](https://thepihut.com/products/raspberry-pi-zero-2) due to its high performance, high efficiency and built in support for most wireless technologies.

This device will communicate with the Microcontrollers in the overall system telling them any configurations and sending and receiving updates. It will also communicate with the cloud service allowing users to see the status of the overall system and make any configuration changes they seem fit.

#### Microcontroller

For this project I decided to use the [Seeed XIAO ESP32S3 Plus](https://thepihut.com/products/seeed-xiao-esp32s3-plus?srsltid=AfmBOoo5-hsizF71-7rRghk5Ye0YzJHWyAl-y0o-qXBYDaSd98_w78jp). This device was chosen due to its low cost, high efficiency and built in ADC, WiFi and Bluetooth technologies.

#### Soil Moisture Management Module

The hardware for monitoring and managing the soil moisture content consists of a Capacitive [Soil Moisture Sensor](https://thepihut.com/products/capacitive-soil-moisture-sensor) and a [Submersible 3V DC Water Pump](https://thepihut.com/products/submersible-3v-dc-water-pump-vertical-type). Alongside a water container and a system of flexible pvc tubes this will allow for full end to end management of soil moisture.

#### Temperature and Humidity Management Module

The hardware for monitoring and managing the temperature and humidity around the crop consists of a [PTC Heating Element](https://thepihut.com/products/ptc-heating-element-5v-40-c), an [aluminium heat sink](https://thepihut.com/products/al-heat-sink-with-adhesive-tape-30-30-10mm), a [Miniature 5V Cooling Fan](https://thepihut.com/products/miniature-5v-cooling-fan-with-molex-picoblade-connector) for spreading the heat, a [Ultrasonic Mist Maker Atomizer](https://www.amazon.co.uk/dp/B0FFN23RLC?ref=ppx_yo2ov_dt_b_fed_asin_title) and a [Adafruit Sensirion SHT45 Precision Temperature & Humidity Sensor](https://thepihut.com/products/adafruit-sensirion-sht45-precision-temperature-humidity-sensor-stemma-qt-qwiic?variant=54007545102721).

#### Light Intensity Management Module

The light intensity management module will be a circuit consisting of a LDR, white LED(s), NPN transistor, and a [Adafruit DS3502 I2C Digital 10K Potentiometer](https://thepihut.com/products/adafruit-ds3502-i2c-digital-10k-potentiometer-breakout-ada4286). The digitally controlled potentiometer allows for the microcontroller to control the automatic feedback loop of the circuit. A circuit that operates independently of the microcontroller was chosen due to the circuits simplicity and to improve the security of the system.

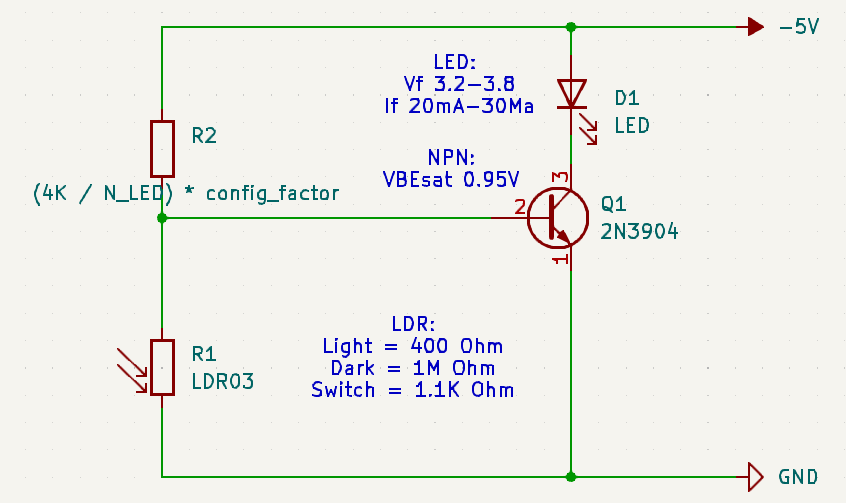




Figure 2 Light Intensity Management Module circuit diagram

#### Battery Module

The microcontroller system would ideally support power over both live power and battery. This is to allow the system to continue operating during power cuts. The components for this consist of a [1200mAh 3.7V LiPo Battery](https://thepihut.com/products/1200mah-3-7v-lipo-battery) and a [PowerBoost 1000 Charger](https://thepihut.com/products/adafruit-powerboost-1000-charger-rechargeable-5v-lipo-usb-boost-1a). The specific batter management IC was chosen due to its ability to simultaneously manage direct power to the system through the USB port and battery power/charging.

### Software

#### Software Development IDE

To develop the software for this project, Visual Studio Code (VS Code) was chosen as the Integrated Development Environment (IDE). This was chosen because of its community extensions support that make development easier.

#### Code Deployment

To deploy the code to the ESP32 and Raspberry pi zero 2W, the Arduino CLI tool was chosen as the tool of choice. This was chosen because of its ability to integrate with the Arduino Community Edition Plugin for VS code.

#### Circuit Simulation

Most testing for this project will be done manually, however some circuit simulation tests are done, for example the light intensity management module, using KiCAD’s SPICE tool. This tool was chosen due to my familiarity with it and it’s ability to analyse Voltage stages in any part of the circuit.

## Hardware Design

### System Layout

## Software Design

### Sensing Modules

### Data Processing Modules

### Communication Modules

# System Development

## Software Development Process

## Hardware Development Process

## System Integration

# System Testing

## Software Unit Testing

## Software Application Testing

## Hardware Unit Testing

## System Testing

# Appendix

# References

**There are no sources in the current document.**