

Abstract

This system we seek to design is an internet-of-things device designed to independently manage gardens through collection and analysis of plant growth data. Automatic gardening systems are commonplace, but this system truly automates by learning from past experiences, and applying that information to future practices. We will be using some kind of microcontroller (more will be revealed when we meet the client) to control and monitor watering, temperature, and humidity. This central monitor will be linked via network to microcontrollers in the surrounding garden, which administer the correct amount of water to each plant. The growth data can either be fed into the machine manually, or recorded by the networked microcontrollers and relayed to the central monitor. The end result should be a fully autonomous networked garden.

1 Problem Statement

Disclaimer: We have not talked to our client yet. Some of the following content will need to be revised.

The problem, on a large scale, is revolutionizing global food production to maximize power, water, and land efficiency while producing sustainable and healthy crops. Future generations will need to feed more people than ever with less arable land. This is where smart gardening (or farming) comes in. A system that not only automatically nurses a large array of plants, but that also learns over time how to grow them the most efficiently, would be a relevant achievement. This problem, when reduced to a smaller scale, amounts to automatically tending a home garden, and learning from past data how to grow plants as large and healthy as possible.

This project, titled The Green Smart Gardening System, aims to monitor, analyze, and implement the findings of environmental factors and their relationship to plant growth. The goal here is to have multiple sensors that monitor humidity, temperature, and soil moisture, and plant growth, and to feed the data from those sensors into a central computer that analyzes it and returns the results to the main controller. In this way, the system learns what watering techniques work best under different conditions, and can calculate the ratio of resource expenditure to plant growth to factor in to its analysis.

To implement this, we will need one or more humidity, temperature, and moisture sensors that can remotely transmit data. We need to connect them all to a centralized microcontroller. The connection will go through a computer that can analyze the data, and then pass to the microcontroller an analysis that informs its function. To be as self-sufficient as possible, the device should be solar powered or fueled by some other green source.

The binding principle behind this whole project is that it is fully automated and sustainable. It manages itself in terms of function and energy usage, and only needs to be interacted with to perform maintenance. To accomplish this, not only the microcontroller, but all of the sensors need to be solar powered. Also, the central monitor needs to be constantly receiving new information to

adjust its settings. In order to analyze the data, it needs to be beamed to the cloud, since the microcontroller will not be able to handle analysis computationally, and is restricted to local data anyway. However, the sensors shouldn't all connect to the cloud. To make things more efficient, the sensors should send their data to the microcontroller, which connects to the internet and sends the data to the cloud. In return it receives an updated, analyzed series of data packets that tell it what to do with the conditions specified. This data is then implemented by the microcontroller.

The microcontroller will need internet capability, and preferably some file transfer protocol. A program will need to be written to receive, assemble, send, and parse data, and subroutines will be written so that the information gathered will be implemented. As far as the analysis goes, some kind of low-level machine learning will need to be implemented so that the data is always managed in a way that makes the plant grow as well as possible.