SmartPlant Hub

Brendon Hales, Daniel Bohl, Kristen Marks, Hunter Cheung

Group 17, Senior Design 2, Summer 2022

Dept. of Electrical and Computer Engineering, University of Central Florida, Orlando, Florida, 32816-2450

***Abstract* — The SmartPlant Hub is a small monitor for any plant which can show measurements by LED screen or by phone app. The SmartPlant Hub is able to monitor moisture levels, light levels, humidity levels, and temperature levels around the plant so the user does not have to. The SmartPlant Hub is connected to a user-friendly app that can be accessed anywhere. This project is made up of well thought out code as well as specifically chosen components, and a 3D printed shell. The SmartPlant Hub was created to make plant care easier on any person who takes their go at gardening.**

The SmartPlant Hub

I. Introduction

Owning and caring for plants, whether it be a small houseplant or a garden of vegetables or fruit, has always been a hobby for many people. Although some are not at all hard to take care of, many are; especially if you intend on ensuring that your plants are in the best environment possible. This will not only allow your plants to live longer, but for food bearing plants it will allow them to grow healthier as well. There's nothing worse in plant ownership than putting your all into taking care of a plant only to have it die because one of its environmental factors was lacking, and this is the problem aimed to solve.

When it comes to taking care of plants, nothing is more important to their livelihood than the environment they are living in. This is the reason many fruits and vegetables have to be grown in certain states or countries, as their climates are best suited for those plants. In wanting to make monitoring the most important factors of the environment easy for anyone, and the best way to do that would be to allow an owner access to all the information they might need in one place: in their pocket.

Our project aims to allow for easy monitoring of one plant with the eventual possibility of multiple at a time. These monitors include sensors for moisture, humidity, light, temperature and pH. The physical sensors themselves will be all put together into one main monitor PCB that is as low cost and low power as possible while maintaining accuracy.

These monitors will be available on a mobile app that is accessible on Android that utilizes wireless connection via WiFi or Bluetooth, depending upon where the user is located relative to their device. The companion app will also have access to a database of information regarding many different plants so that even the most uninformed plant owner can know the specifics of caring for whatever plant they may have. Although there is something that can be said about being self-informed and monitoring your plants yourself, our goal is simply to make this easier and more accessible for everyone.

II. Motivation

One of the most popular hobbies in today's age, especially during and after the Covid-19 pandemic, is gardening. A large issue, however, is most people consider themselves very poor plant parents. This lack of "green thumb" could be due to the lack of time dedicated to understanding the plant's requirements.

Some people decide to go to a plant store, pick out any random plant that they think looks cute, and then take it home and place it anywhere in their home. They think that all plants are cookie-cutter, that they all require the same basic needs.

Although this is true to an extent, plants all have individual needs and requirements that are needed to help the plant thoroughly thrive. These needs can be as basic as the amount of water needed, to elaborate as the amount of relative humidity in the air.

The main objective for this project was to attempt to design a product that helps even the ineptest plant owners be able to grow any plant they desire and have it thrive with ease. On top of this, it is important to provide information to anyone who wants to learn about common plants that would be grown, whether that is vegetables, herbs or just common plants you could find even outside of a garden. This will give a jump over similar products that just serve to allow someone to have a basic monitor of their plants by providing education as well through the application.

Not wanting to create a system that runs itself entirely, as wanting to allow the user to have some control and allow them to learn themselves. This means making the device a lot more portable and manufacturable for most people, as a smaller device ends up being easier to move as well as cheaper for the average consumer. This will allow even more people to have access to the knowledge being provided.

III. Goals

The team had multiple goals in mind when trying to come up with a solution for this problem that was identified.

A. *Cost effective*

- The project should be cost effective, down to a point that any individual would be very inclined to purchase the item.
- Another consideration would be to have the project be low enough cost if the team decides to try and scale the project out, meaning that with multiple units so there could be multiple units on one home monitoring different plants, as compared to having a single unit that had to be transported
- One of the bigger expenses of the project would be some type of enclosure as it did not want the monitor to be too exposed in case water was introduced into the environment
    - 3D printing was considered as it is relatively cheap as long as the system is already setup (strictly the cost of filament)
    - 3D printing could be utilized to create a custom enclosure, in any color that a customer might want, while still being conscious about the cost associated with it

B. *Small in size*

- The overall design of the project is needed to be small in size, as it is meant to be used in conjunction with a pre-existing plants
    - Too large of a size would not be good for perhaps a plant with a small pot, as the monitor could be too large for the plant and pot
- The rough desired dimensions are large enough to contain all necessary parts of the monitoring project, while also being small enough to move if necessary

C. *Improved plant health*

- The device should ultimately aid in the overall health of the plant. This device should in no way get in the way of plant growth but only help the user keep an extra eye on it to help keep up with the plant.
- This device should be able to detect what a plant needs based on what the sensors read
- The device will be able to record plant data and should be able to estimate what will need to happen next

D. *Ensuring success when growing plants*

- The device will act as an extra hand for the user when they are not around. This device will alert the user if something is low and needs to be done asap for the plant

- This device will allow for the user to be away from their garden or plant for a long period of time and able to care for the plant when the device alerts them of any problems

E. Help user control what the can not see

- This device should act like a second hand to the user as well as eyes where the user can not reach, feel or see.
- The added sensors will allow for the user to see the health level of their plant which they would not be able to see without our device.
    - pH
    - Nutrients levels
    - Light levels
    - Moisture levels within the soil
    - Temperature surrounding the plant
    - Humidity levels surrounding the plant
    - Overall plant health under and around the soil

IV. Related Works

In terms of related products, there are a number that are already out on the market. These all vary in execution immensely, from one utilizing the idea of an "all-in-one" plant monitoring and plant caretaker system, to another using an array of smaller probes to monitor various plants from a central hub station. Some companies also offer the ability to use an entire enclosed system from which a plant can be automatically grown. These systems take advantage of an overarching concept of hydroponics, where traditional soil is replaced with nutrient enriched water and supplements.

The issue with these devices is that they are often large and require maintenance, mainly being changing of pH solutions. Although there is a significant amount of overlap in the parts used between the hydroponics systems and our project, the methodology of being small and cheap are diametrically opposed to this design and therefore our project leans toward monitoring rather than growing. This is easily observable as in a hydroponics system, they utilize tanks for liquids as well as light systems meant to regulate the amount being received by the plant. The addition of those subsystems would require a significantly large rig/case in order to house the entire system.

The team aimed to create a product that is the culmination of all of these different executions. Having a user be able to monitor multiple plants, from a mobile application, without the need of purchasing a large, expensive self-contained system. The team aimed to reach a target audience of plant owners that want a quick overview of their plants' health, with easy to identify vital signs that can be easily monitored and fixed by the user in a clear, discernable way.

A. Northfifteen Home Plant Monitor

This device keeps tabs on your plant and allows the user to use an app while on the go. This device uses sensors to track various plant life such as water life, overall lighting, fertilizer, and temperature. This device uses bluetooth to connect to the app for the user. The main feature of this device is the water tracker for plant life that you are able to see through the app.

The SmartPlant Hub differs from this because the app is able to ping the user when any of the levels are near dangerously low and the user is able to interact with the app to know exactly what to do for the plant. It also makes suggestions for what to do.

B. Tulip Plant Monitoring System

This device is not only one monitor, but multiple monitor probes to form a system of probes for each plant in a garden. This also comes with a handheld spectrometer that the device sends the data to by bluetooth. The spectrometer can only be used when the user is in range of the plants it is trying to measure. The user can plug the spectrometer into their computer and download the data, which in turn charges the device. This device can measure moisture levels, air temperature, fertilizer, and light through wifi signals. The user can also use an associated app to create a virtual map of their garden environment.

The SmartPlant Hub differs because there is an app that the user does not have to be in the general area to see the status of their plant.

C. Click and Grow Smart Garden

This is an all in one planter device that you put your plant in. It is quite bulky as the plant sits on top of the potted bottom with a bar along the top. This device waters your plant for you as well as changes lighting levels and tracks the nutrients levels of the plant. This is pretty much self-growing technology because this planter does everything for you to take care of the plant. This planter requires the user to purchase the required seeds for this device which can get quite pricey and make it hard to branch out with different types. And due to the side and shape of this planter, the plant will only have a certain amount of room to grow and live.

The SmartPlant Hub is lightweight and small to fit inside the plant's pot or even just be stuck in a garden. The SmartPlant Hub is also waterproof which this planter is definitely not. The SmartPlant Hub also keeps the user in the loop and allows the user to take all the steps to care for the plant. There is also an app which this device does not have one connected to.

D. EasyBloom Plant Sensor

This device is shaped like a flower to hide from plain sight near the plant. This device can measure lighting, temperature, humidity and water. The device needs to stay on for at least 24 hours to get the most accurate results of the plant per day. This device has a web interface that lets the user categorize plants by soil, color and more. This device has three different modes which include monitoring mode, recommendation mode, and water mode. Their water mode tells the user about the water level and the device needs to be removed when watering the plant.

The SmartPlant Hub uses an app rather than an online interface which makes it easier for the user to test their plants from anywhere. The SmartPlant Hub also allows the user to leave the device in the plant and not have to frequently check on it. The SmartPlant Hub also allows the user to water the plant without removing the device.

V. ENGINEERING SPECIFICATIONS

The team aimed to demonstrate that our system can enter a sleep mode by ensuring that after a 30 second timer elapses following the last use of the system, the monitor will enter a “sleep mode”, also known as a standby mode. This mode will be activated to ensure that the system is focused on power saving.

On the attached LCD screen, the team aimed to have the data values that will appear on the display update every 5 seconds, just so the customer can ensure an accurate readout, eliminating the possibility of viewing an anomaly with the sensors. The team aimed to have all data transmitted from the monitoring system to the user, in under 3 seconds. This level of speed is on the slower side, but it allows for all possible forms of communication from the system to the user to be considered. The team planned to have communication time significantly quicker than this threshold value, however.

VI. SYSTEM COMPONENTS

This system is made up of many components which all work together to create a final product. The components are introduced below. During testing, a few issues were noted, and there are even parts that came in broken from the factory. These issues such as the inability to get particular sensors to communicate with the microcontroller will have to be dealt with, and resolved, before any feasible work is completed. These sensors are vital to the success of the project, and without them, the project will not be as well put together as planned. More time and research needs to go into these sensors, allowing the team to decide if there is an issue regarding the sensor itself, or if it is possibly as simple as a wiring, or a coding issue during testing.

E. Microcontroller

The microcontroller chosen to use for this monitoring device is the HUZZAH32. This microcontroller is lightweight and small which will fit perfectly in small plants as well as larger plants. This board had the best price to performance ratio and will allow for taking advantage of Arduino’s software and connections on a cheaper comparable board.

F. Humidity and Temperature Sensor

The humidity and temperature sensor chosen ended up not working due to extreme coding reasons, so it was decided to go with one our team member had from a previous project. This sensor was easily codable and compatible with Arduino. This sensor was accurate and fast which was important to make this monitor the best in the market.

G. Light Sensor

The light sensor chosen was a module for Arduino, however for some reason this sensor did not work. The team decided to rip off the photoresistor on the sensor and of course it worked so much better than anyone would have thought, the photoresistor is extremely accurate and fast which again is important for the monitor.

H. Moisture Sensor

The moisture sensor chosen is the STM32 which is a probe type moisture sensor. This sensor is inserted directly into the pot with the plant and is able to detect within seconds. This is very important to the monitor because moisture is arguably the most important part of a plant’s health.

I. Lithium-Ion Battery Pack

This battery pack is important to the monitor to keep it alive for a long time. Choosing something not strong enough that the user would need to change all the time would just get annoying and less helpful to buying this product. Using sleep mode often, this battery pack will last for as long as hopefully the plant's life or longer.

J. LCD Display

For this project, the HiLetgo display was chosen due to easier installation as well as small pocket sizing. Although the screen is small, a user will always be able to see everything through their phone as well. This screen shows all the measurements that our sensors detect and the LED lights near it show a red or green light that is easy to see as well.

K. PCB

The PCB was created to handle all power usage through the whole module, whether it be any single sensor or the microcontroller itself, the PCB will regulate all of it. The PCB is also able to convert USB input to code our project, to UART for the microchip on our board to understand. The PCB houses jumpers for our light and moisture sensor, as well as the battery pack. Without this PCB there would be no regulation of power and our microcontroller would be always working at full speed. The main microchip on this board is the WROOM which is also on the microcontroller. This was important for the team to have because then when waiting for all parts of this PCB to arrive, they were able to have all code ready and knew the pinouts would be right for the sensors.

VII. Sensor flowchart

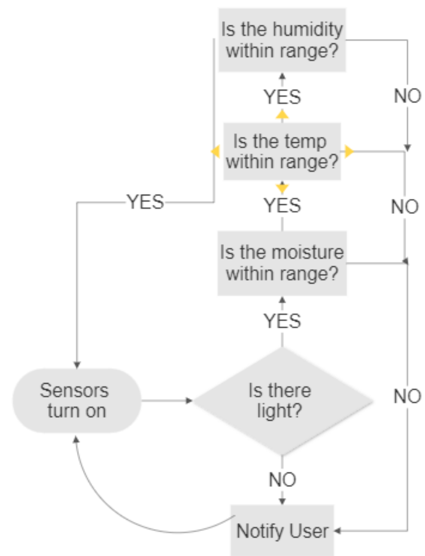

Fig. 1: Sensor Flowchart

The flowchart in figure 1 shows the basic flow of how the sensors are operating. No matter where you start in this system, a sensor is reading. When a sensor reads a value that is not wanted, for example moisture level is too high, then the system will notify the user by pinging them through the app or lighting up one of the LEDs. The system will then go again and when the answer is perfect moisture level, then the system will move on to the next.

The point of the flow being like this is due to keeping the user notified in almost every aspect of their plant life cycle. This monitor will not only monitor the plant but help and alert the user when something is off. The reason for continuous checking is because at any time, the temperature could have just been checked and maybe five or so minutes go by and now the temperature is a bit too high.

This flow chart does not take into account the times where the LCD or LEDs are used, only the sensor flow. This flowchart also does not take into account the intermittent sleep modes to save battery life as well as sensor life. This flowchart assumes that throughout each cycle, each sensor will be consulted as to if their piece is being met, when it is it moves on, when it is not it notifies the user to make a change and checks again.

VIII. Display flowchart

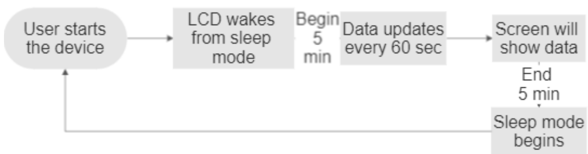

Fig. 2: LCD Flowchart

Figure 2 is the LCD display. Although relatively straightforward, certain requirements need to be considered such as placement, response time, and interactions between the button, microcontroller, and display. When initially approaching a workable design, the concept of a physical display to forgo the necessity of a smartphone was clear as it is inexpensive and extends usability. Upon a user pressing the physical button, the microcontroller will be woken up from a sleep state, the consensus is that the device will remain on for a period of 5 minutes and update the data on the lcd every 60 seconds.

Using interrupts and timers using the master signal, microcontroller, clock signal and then displaying the relevant data on the screen makes this achievable. There is already open-source code that is available online for the LCD selected and the use of such should make this implementation simple. Upon the end of the 5 minute interval, it is intended for the device to power off entirely. Should the user turn the device on using the app remotely, the LCD should not turn on as to save power. This choice was made after considering that the user is already on their phone to power the device, therefore the need to physically see the screen is redundant.

IX. Software flowchart

Figure 3 is the general software design flowchart for the device. This is a very simplified explanation of the big picture interactions between the larger systems of the device. What is not captured is the fact that the user will turn on the device either with the physical button or with the mobile app. In the case that the user turns on the device using our physical button, the process will still be the same in that the database will be updated, however they will not see the relevant notifications due to them not having the mobile app open.

It is possible to implement it in a way that once the sensor data is read, it will compare to the baseline on the microcontroller rather than the mobile app and alert the user on the LCD display as well. This however seems to be redundant as considering the user will be at home or on a trip when checking their plant, they would always have their phone on them therefore the mobile app is sufficient. The users will also be able to turn off the device using the mobile app, which is not captured but is an important software interaction with the device.

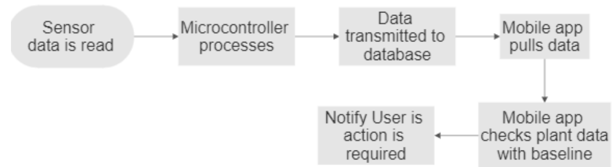

Fig. 3: General Software Flowchart

X. Code

The device took an extensive amount of research on documentation and sample code for the various sensors and processes implemented. In order to deliver usable data to the database from the microcontroller, it was imperative that the microprocessor act as a station from which the firebase can be communicated with via WiFi. There was also the challenge of employing the sleep/wake-up functionality from which the user can receive accurate readings on their plants data. By making use of the Arduino's time library the current time at which the device was woken up could be tracked and therefore the time which had passed would be compared to our set timer in order to initiate the sleep state. In doing so the device is able to save a considerable amount of power as without the sleep functionality and only constant running, it would last only twenty-two hours.

XI. App Flowchart

Figure 4 is the mobile app software flowchart which outlines the ways in which the mobile app interacts with our database to deliver the data our microcontroller gathers. In the general sensor flowchart (figure 1), at every point in which the sensors are meant to notify the user, it is actually just sending that sensor data to the database and when it is displayed to the user within the mobile app, that is when the user is notified of necessary action. One interaction that is not shown on this flowchart is that when the user starts the app, it sends the wake-up call to the device, from there the MongoDB tables are updated.

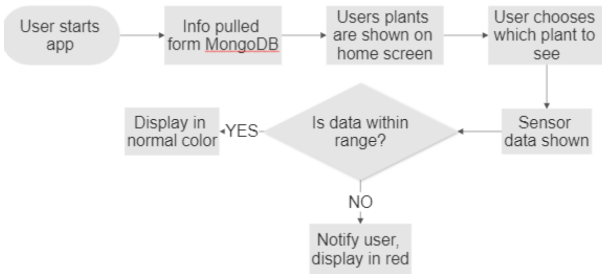

Fig. 4: App Flowchart

A more detailed explanation of the mobile app itself and the components included within it are detailed in the software interface section. The most important interaction of the mobile app is waking up the device upon logging in, without this action there would be an issue of incorrect data being displayed to the user which would negatively impact plant growth/health (unless they physically start the device with the button, enabling an update to occur).

XII. App

The SmartPlant Hub also has a mobile application that will allow users to monitor their plants remotely. The app works similar to the press of a button on the device itself that updates the LCD screen in terms of the data provided to the user. The mobile application was developed using Java and runs on the Android Operating System. Android is easier and more flexible for development than iOS for the SmartPlant Hub. The mobile app also provides access to a database of commonly owned house plants with information on each, including the optimal environment for each.

The mobile application gets information from the databases via Firebase which is updated live with sensor data from the user's device when prompted in-app. Users are able to create an account and log into the app to connect to their devices and view the data when they want to, even from far away.

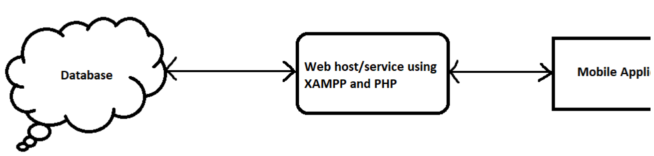

Fig. 4: Integration of Data Transfer

XIII. Conclusion

In conclusion, the team was able to come up with an overall project that will be able to take care of most, if not all, of the goals when creating the documentation for this. The final product is in a 3D printed shell which holds all components and is able to sit inside a pot with a plant.

There were many challenges the team had to face when design came around, such as ordering the PCB. The PCB was the very last component to come in which made design of the final product very hard. Of course soldering all small components onto the PCB was also a very stressful time.

The team was able to have many successful runs of diagnostics which are shown in the video documentation. The engineering specifications are also shown in this video.

Overall the team was extremely happy with how the final product has come out. The team was able to work very well together and hope the product works well for others and can continue to make people smile.

The engineers

Kristen Taylor Marks is a 26 year old graduating Electrical Engineer who has taken a job with Nelson Engineering Co to work on creating plans for power systems.

Daniel Bohl is a 24 year old graduating Computer Engineer aiming for a position in full stack software engineering and development.

Hunter Cheung is a 22 year old graduating Computer Engineer who is working a full-time position as a software engineer for 4C Solutions.

Brendon Hales is a 24 year old graduating Computer Engineer who has taken a job with Dignitas Technologies as a software engineer.

Acknowledgement

The authors wish to acknowledge the assistance of Dr. Lei Wei, Dr. Samuel Richie, and all the committee judges; University of Central Florida.