

# **Plant Growth Station: Aiding the Average Person in Becoming a Better Gardener.**

José Amilcar Mata Calidonio  
Department of Physics and Engineering, Eastern Nazarene College  
Quincy, MA 02170 USA  
jose.a.mata-calidonio@lions.enc.edu

## **Abstract**

The Plant Growth Station (PGS) is a product designed to aid customers in the process of growing their own plants. Using a web application, a Raspberry Pi, and two sensors, the product gives users access to sensor data such as the air temperature, humidity level, and light level that the plant is exposed to. This sensor data is displayed in graphs that update every five seconds to achieve near real-time monitoring of the plant's environmental conditions. The application provides multiple features for users such as user registration and login, password reset function, username and profile picture creation and personalization, post creation and sharing, etc. Email services are also included in the application, such as personalized email alerts when sensor values go above or below user-defined thresholds, and daily email reports with the average sensor values of the past day. Unlike self-watering pots or similar automated pots and comparable products, the main purpose of the PGS is to give the user useful information about the plant's environment to help ensure that the plant is evolving under optimal conditions. Therefore, the PGS allows the user to have the complete gardening experience for themselves, with all the decisions involved in it, while providing useful information that enables the user to better take care of the plants.

## **Introduction**

Many people do not have significant gardening experience. It is not common for the average person to know enough about specific types of plants to successfully know how to grow them and take care of them, so the plants grow as healthy as possible. This is because different plants have different environmental conditions for their optimal growth, like the temperature of the environment, the air humidity, and the amount of light they get every day. Moreover, even if a person does research and investigates the optimal conditions for their plants' development, it is difficult to ensure that the plant stays in these conditions without the necessary tools.

The Plant Growth Station is a product designed to help the average person trying to grow plants. The main purpose of the PGS is to aid customers in the process of growing their own plants by providing a way to monitor the plant's conditions online. The PGS consists of a physical container that serves as a pot for a plant with hardware that monitors the plant when desired. Additionally, the PGS has a web application and website through which the user can monitor the plants' environment online and personalize email services to ensure the plant's conditions are within established parameters. With the PGS, users are able to view the temperature, air humidity, and light level values online close to real-time. Furthermore, the PGS provides two email services: the customizable email alert service

and the daily email report. The PGS website also includes other features like user registration and post sharing as social features that allow users to interact. These features are aimed to build a community of people that grow all kinds of plants and want to share their gardening experience with others who are experiencing it too.

## **Methods and Development**

The hardware used to develop the Plant Growth Station includes a plexiglass container, a Raspberry Pi 4, two digital sensors for the Raspberry Pi, cables used to connect the sensors, a fan to cool the Raspberry Pi, and a power chord for the Raspberry Pi.

The plexiglass container has a length of 16.2 inches, a width of 8.4 inches, and a height of 10.3 inches. This container is meant to be used as the pot for the plant, where the soil and plant will be placed.

The Raspberry Pi 4 is a single-board computer with the approximate size of a credit card. It is a very powerful piece of hardware that can be used to browse the internet, run code in high-level languages, process data, run applications, and more. It counts with 40 GPIO pins, some of which can be used to transfer data supported by different protocols [1]. The Raspberry Pi functions as the CPU of the Plant Growth Station, powered by its own power cord connected to a standard 120V, 60Hz outlet. It is also connected to an ethernet cord that provides an internet connection.

The DHT22 Temperature and Humidity sensor is used to measure air temperature and humidity. The DHT22 integrated circuit (IC) used serves as a digital sensor that transfers data serially with 16-bit accuracy. The IC can measure temperatures from -40 °C to 80 °C with an accuracy of  $\pm 0.5$  °C and humidity levels from zero to 100 percent with an accuracy of  $\pm 1$  percent [2].

The BH1750 Light Sensor is used to measure the light level near the plant in lux units. The IC sends 16-bit digital values using the I2C protocol. It can measure light levels from zero to 65535 lux of light [3]. Both the DHT22 and the BH1750 sensors are powered and read by the Raspberry Pi's GPIO pins by using insulated wires to connect them.

In the Raspberry Pi, a Python script was created which implemented existing libraries to read and send the data from both sensors. The data is read every three seconds and sent directly to the server where the web application runs as a POST request. The Python script also includes a hardcoded identifying number. This identifier is meant to be unique for every PGS unit and is associated with one unique PGS account. It was implemented as a step towards an easier implementation of more PGS units as more users use the product.

Several tools were used in the development of the PGS web application. The web application was developed and still currently runs in a server of Eastern Nazarene College. The server runs the Ubuntu 18.04 operating system from Linux. Most of the code utilized was obtained from Corey Schafer's GitHub repository "code\_snippets" using the MIT License [4]. This code was edited using Visual Studio, a text editor with a package that enables editing files in a remote server. GitHub, an online cloud service used to store and download repositories, was also used to save all the code of the web application.

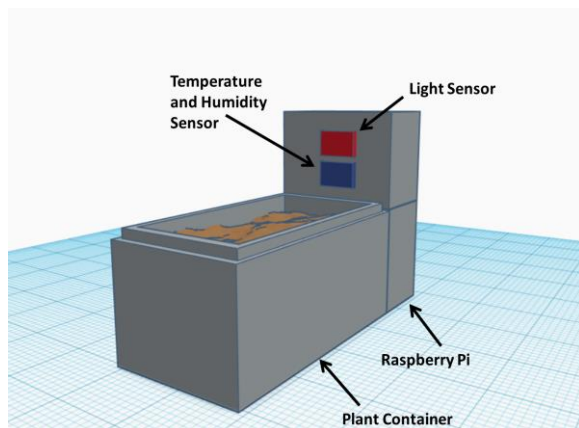
To build the web application, Flask was used. Flask is a micro web frame used to develop web applications. The PGS web application makes use of several Flask extensions that ease different processes like registration, login, data forms, image processing, among others. An SQLite database, a relational database, was created and used by the web application to store user information like email, password, profile

picture, posts, etc. An Influx time-series database, a database made explicitly for time-sensitive data, was also created and used to store the sensor values received from the Raspberry Pi. Lastly, Grafana, a visualization platform, was used to display the sensor data in the Influx database. Grafana displays the data in graphs and enables further customization by users if desired.

The costs of the resources used to develop and build the Plant Growth Station totaled to around \$114. The costs of certain elements may vary, like the Raspberry Pi set desired, the container, etc.

## Device and System Design

The following figures show the device and the hardware used to build it. Figure 1 shows a conceptual 3D model (made with Tinkercad) of the PGS and its components.



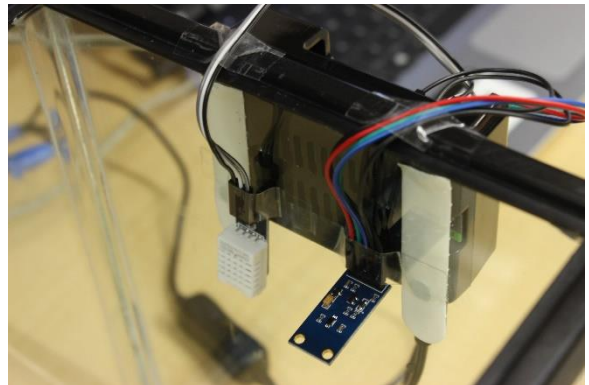
**Figure 1.** 3D Model of Conceptual System Structure.

Figure 2 shows the completed PGS Device. The black device attached to the container is the Raspberry Pi in its case. The only cables connected to it are its power cable and an ethernet cord.

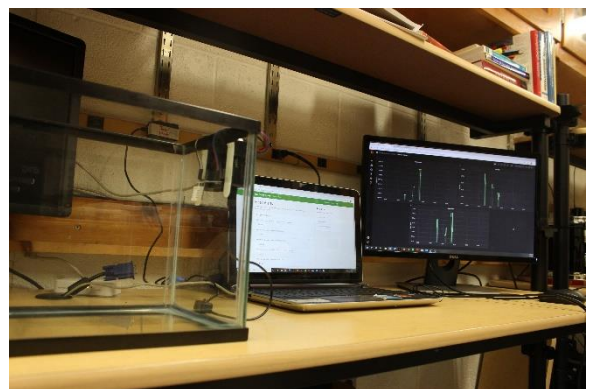


**Figure 2.** Completed PGS device.

Figure 3 depicts the sensors used. The white IC on the left is the DHT22 sensor, and the blue IC on the right is the BH1750 sensor. Figure 4 depicts a complete setup of the Plant Growth Station. The screen on the left shows the PGS website, and the screen on the right shows the graphs displaying the sensor data.



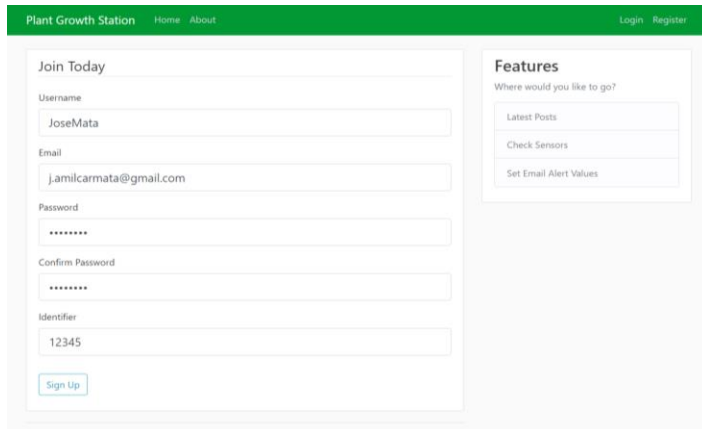
**Figure 3.** DHT22 and BH1750 sensors.



**Figure 4.** Complete setup of the PGS.

## Features

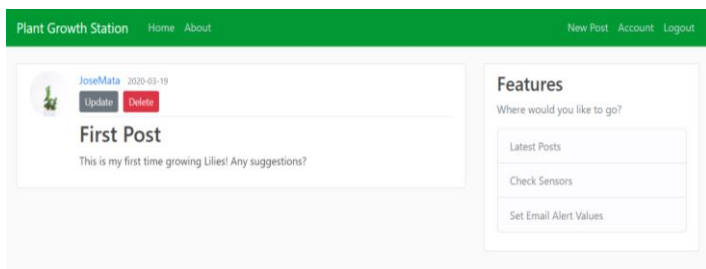
The PGS includes several features for its users. On the website, the first thing a user does is create an account. An email, username, password, and the unique identifier number specified in the Raspberry Pi are required to create an account. As a security feature, all user passwords are encrypted. Figure 5 below shows the registration screen of the website.



The registration page features a green header with 'Plant Growth Station', 'Home', and 'About' links, and 'Login' and 'Register' buttons. The main content area is divided into two sections. On the left, under 'Join Today', there are input fields for Username (filled with 'JoseMata'), Email (filled with 'j.amilcarmata@gmail.com'), Password (masked with dots), Confirm Password (masked with dots), and Identifier (filled with '12345'). A 'Sign Up' button is at the bottom. On the right, under 'Features', there is a sub-header 'Where would you like to go?' followed by three buttons: 'Latest Posts', 'Check Sensors', and 'Set Email Alert Values'.

**Figure 5.** Registration Page.

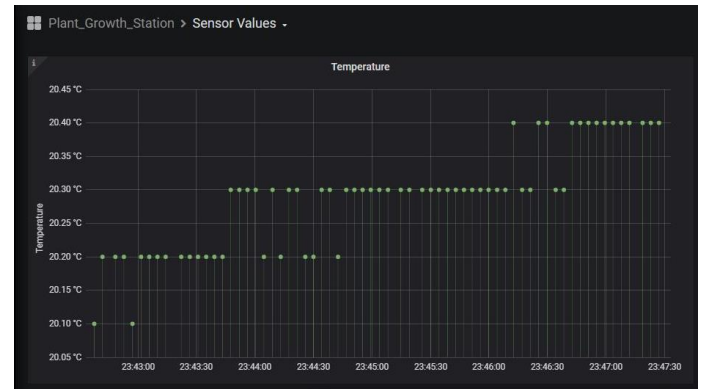
Once the account is created, a user can log in. In their account, a user can change their username, profile picture, and email address. The website includes a password-reset feature that sends a password-reset email to the user's email inbox. A user can also make and edit posts for others to see. Figure 6 shows the screen for editing a post created by a user.



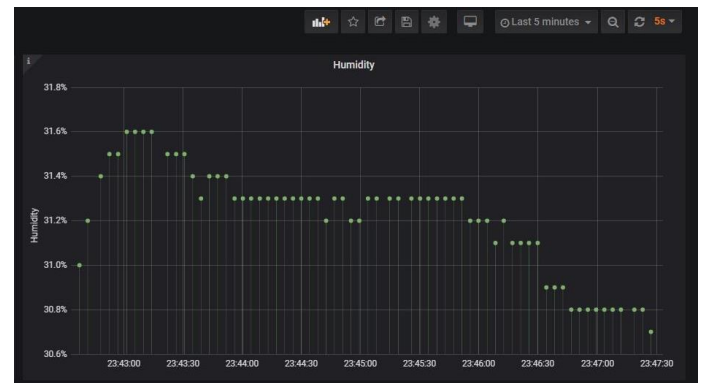
The post edition page has a green header with 'Plant Growth Station', 'Home', and 'About' links, and 'New Post', 'Account', and 'Logout' buttons. The main content area shows a user profile for 'JoseMata' with a profile picture and a post titled 'First Post' with the text 'This is my first time growing Lilies! Any suggestions?'. There are 'Update' and 'Delete' buttons for the post. On the right, under 'Features', there is a sub-header 'Where would you like to go?' followed by three buttons: 'Latest Posts', 'Check Sensors', and 'Set Email Alert Values'.

**Figure 6.** Post Edition Page

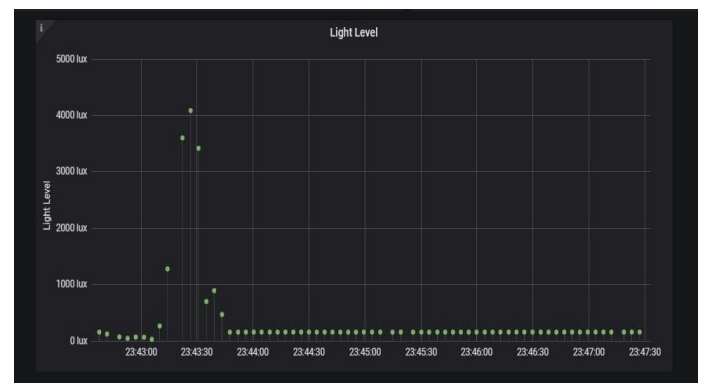
The graphs of the temperature, humidity, and light level can also be viewed at near real-time speeds. Figures 7, 8, and 9 show the sample graphs, respectively. These can be personalized if desired. The graphs can be set to update up to every five seconds.



**Figure 7.** Temperature Graph.



**Figure 8.** Humidity Graph.



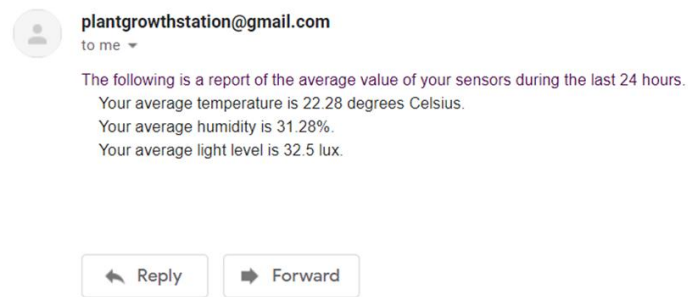
**Figure 9.** Light Level Graph.

There are two email service features of the PGS: the email alert feature and the daily report feature. The email alert feature lets users decide high and low values for temperature, humidity, and light levels. These values are typed in by the user on the PGS website by clicking “Set Email Alert Value” on the “Features” sidebar. If the PGS measures a value higher than the user-defined high values or lower than the user-defined low values, the user will receive an email warning about the specific measurement that triggered it. The email includes both user-defined threshold values for that measurement and the value that triggered the email. Once an email is sent for a specific measurement, another email for the same measurement won’t be sent for at least another half hour. Figure 10 below shows a sample email alert for a light level measurement that exceeds the user-defined maximum light level.



**Figure 10.** Sample Email Alert.

Finally, the daily report feature is used to send the user a report every day with the average temperature, humidity, and light level values. This report is sent out at 8:00 PM EST every day and includes the average values of each measurement for the previous 24 hours. This feature is meant to give users a general idea of the conditions that their plant has been under during the previous 24 hours. Figure 11 below shows a sample daily report email.



**Figure 11.** Sample Daily Report Email.

To turn on the Plant Growth Station, all that is necessary is to flip the Raspberry Pi’s power switch on. After around 15 seconds, the updated values can be seen on the graphs and are automatically updated every five seconds.

## Conclusions

In conclusion, the Plant Growth Station is a product that allows users to monitor the temperature, humidity, and light-level at all times online near real-time. Its email features help the user ensure that the plants are under the user’s desired conditions and help the user have an idea of what the last 24 hours were like in the plant’s environment. The social components of the website enable users to share posts and interact with other users. This feature could be an effective way of seeing how others use the PGS, getting advice, and sharing progress with other users. The PGS as a product aids users by maintaining them informed but does not automate the care for the plants for users who want this part of the experience themselves.

## Future Work

There is room for improvement of the PGS by adding additional features. Currently, the application is deployed on an ENC server. If the application was deployed in a server

from a service like Google Cloud or AWS, it could be accessed using Wi-Fi instead of an ethernet cord. Another feature that could be implemented is a database containing information of many types of plants, so the user can find information specific to the plant they are trying to grow and their ideal conditions on the PGS website. This database would also give users the option to automate the threshold values for the email alert system, depending on the type of plant grown.

## Acknowledgments

Firstly, God. I thank Eastern Nazarene College for funding this project and providing the facilities and resources necessary for its development and implementation. I also thank professor Pierre-Richard Cornely for his instruction of the systems engineering procedures as well as his guidance and advice throughout the project's development.

## References

1. Raspberry Pi 4 Tech Specs. Retrieved from: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/>
2. DHT22 – Temperature and Humidity Sensor. (2018, April 19). Retrieved from: <https://components101.com/sensors/dht22-pinout-specs-datasheet>
3. HiLetgo GY-30 BH1750FVI Digital Light Intensity Sensor Module For AVR Arduino. (2016, March 30). Retrieved from: [https://www.amazon.com/HiLetgo-BH1750FVI-Digital-Intensity-Arduino/dp/B01DLG4NZC/ref=sr\\_1\\_1\\_sspa?keywords=bh1750&qid=1583987100&sr=8-1-spons&psc=1&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUEzTEZOM0JGWEI5RFIQJ](https://www.amazon.com/HiLetgo-BH1750FVI-Digital-Intensity-Arduino/dp/B01DLG4NZC/ref=sr_1_1_sspa?keywords=bh1750&qid=1583987100&sr=8-1-spons&psc=1&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUEzTEZOM0JGWEI5RFIQJmVuY3J5cHRlZElkPUExMDM1MjEyREkzQkQ3R09LTVJXJmVuY3J5cHRlZEFkSWQ9QTA2NjQ3NjBLQkpYSIIxWjhHRTMmd2lkZ2V0TmFtZT1zcF9hdGYmYWN0aW9uPWNsaWNrUmVkaXJlY3QmZG9Ob3Rmb2dDbGljaz10cnVl)

mVuY3J5cHRlZElkPUExMDM1MjEyREkzQkQ3R09LTVJXJmVuY3J5cHRlZEFkSWQ9QTA2NjQ3NjBLQkpYSIIxWjhHRTMmd2lkZ2V0TmFtZT1zcF9hdGYmYWN0aW9uPWNsaWNrUmVkaXJlY3QmZG9Ob3Rmb2dDbGljaz10cnVl

4. Schafer, Corey M. Code\_Snippets. Retrieved from: [https://github.com/CoreyMSchafer/code\\_snippets](https://github.com/CoreyMSchafer/code_snippets)