University of Arkansas – CSCE Department
Capstone II – Final Report – Fall 2021

Automated Greenhouse

Nathan Barlass, Lauren Jackson, Scott Jackson, and Zachary Tyler

Abstract

Plants and gardening have become a large part of many peoples lives and a popular hobby, and increased in popularity even more during the pandemic. With people starting to get back to their normal routines, this project will allow people to take the same care of their plants but use less of their time. This greenhouse will increase the chance of plants surviving with minimal care and encapsulates the idea of sustainable farming on a smaller scale. This could become critical with the climate crisis and allow for accessibility of food to more people.

This automated greenhouse is controlled directly through a Raspberry Pi and some sensors are used to detect moisture, temperature, and sunlight levels. The data from the sensors are displayed through a mobile app by a database and the user is able to interact with the greenhouse remotely.

1.0 Problem

The problem that we are addressing is that it can be difficult to keep plants healthy and alive. On the surface, the inability to maintain plants seems like a minor issue. Although, with the increasing threat of climate change, the capability for sustainable farming will be essential. This project is a small scale example of what could be used in a larger scope. It could also be used for individuals who lack the time and/or energy to take care of their house plants as well. Not having an automated greenhouse could result in several plants dying and this could be a huge problem if it were being used to produce food.

2.0 Objective

The objective of this project was to construct an automated greenhouse that can be monitored and controlled through a raspberry pi and mobile device. Information about the greenhouse such as the sunlight levels, temperature, or soil moisture levels are sent to a database that can interact with a mobile application that can output this information and water the plants from the mobile device.

3.0 Background

3.1 Key Concepts

There are three layers of technology required for this project. First, a Raspberry Pi was programmed with Python and used to: receive data from sensors such as a temperature sensor,

moisture sensor, and light sensor; send data to a database for remote access and viewing via a mobile smartphone; and receive controls data from a mobile smartphone which was used to actuate a water pump.

Second, a database was created to act as an interface for communication between the Raspberry Pi and mobile smartphone. Google Firebase is used to manage the data from the greenhouse and allows real-time communication between the Raspberry Pi and the Android application. JSON and XML were used to manage the data and define technical aspects.

Third, a mobile device is used for user interface and controls. The physical device is an android smartphone, the SDK is Android Studio, and the programming language used was Java.

3.2 Related Work

For the scope of this project, this greenhouse is fairly basic and implements the same ideas that many others have done such as detecting moisture and light levels. [1] We added a mobile application to improve user experience, which sets this automated greenhouse apart from others. With the app we are able to keep up with real time data remotely and start the water pump remotely. The other implementations usually just had a timer and would water at a scheduled time. That part is easy to implement but we added the watering function to the mobile app to have a better user experience and allow the user to water it when they see moisture levels are too low.

4.0 Design

4.1 Requirements and/or Use Cases and/or Design Goals

- A Raspberry Pi will be programmed to send and receive data
- A greenhouse will be built that can be controlled remotely
- The greenhouse will have a water pump connected to the Raspberry Pi
- A database will be created to manage and store the data
- Each plant shall have a moisture, temperature, and light sensor
- Each of these sensors will send data to the Raspberry Pi
- We will program a mobile app to retrieve data from the database
- We will use that data to be able to manage the greenhouse
- The user will be able to read the status of the sensors
- The user will be able to manually water with the app

Use Case 1: User sends data

1. User inputs command to actuate water pump
2. Mobile application sends command to database
3. Raspberry Pi pulls command data from database
4. Raspberry Pi sends command data to water pump
5. Water pump actuates

Use Case 2: User receives data

1. Sensor sends data to the Raspberry Pi

2. Raspberry Pi sends sensor data to the database
3. Mobile application pulls sensor data from database
4. Sensor data is displayed to the user

Use Case 3: User creates timer-based watering schedule

1. User inputs watering time frequency
2. Mobile device creates timer based on user input
3. Timer is activated and sends command to database
4. Raspberry Pi pulls command data from database
5. Raspberry Pi sends command data to water pump
6. Water pump actuates

Use Case 4: User selects sensor-based watering schedule

1. User selects minimum moisture level of plant
2. Mobile application receives moisture sensor data
3. Minimum moisture level is met and sends command to database
4. Raspberry Pi pulls command data from database
5. Raspberry Pi sends command data to water pump
6. Water pump actuates

4.2 Detailed Architecture(need screenshots and more specifics for final report)

There are 3 major components of this project: a mobile portion, a database portion, and an embedded systems portion. These three parts work together to achieve a greenhouse system which can be remotely and/or automatically watered and informs the user of plant states such as moisture, light, and temperature levels.

The embedded systems portion is used as a controls system for any relevant motors, pumps, or actuators and will be used to read and send sensor data. We used a Raspberry Pi programmed with Python to send/receive data to/from the mobile device via the database.

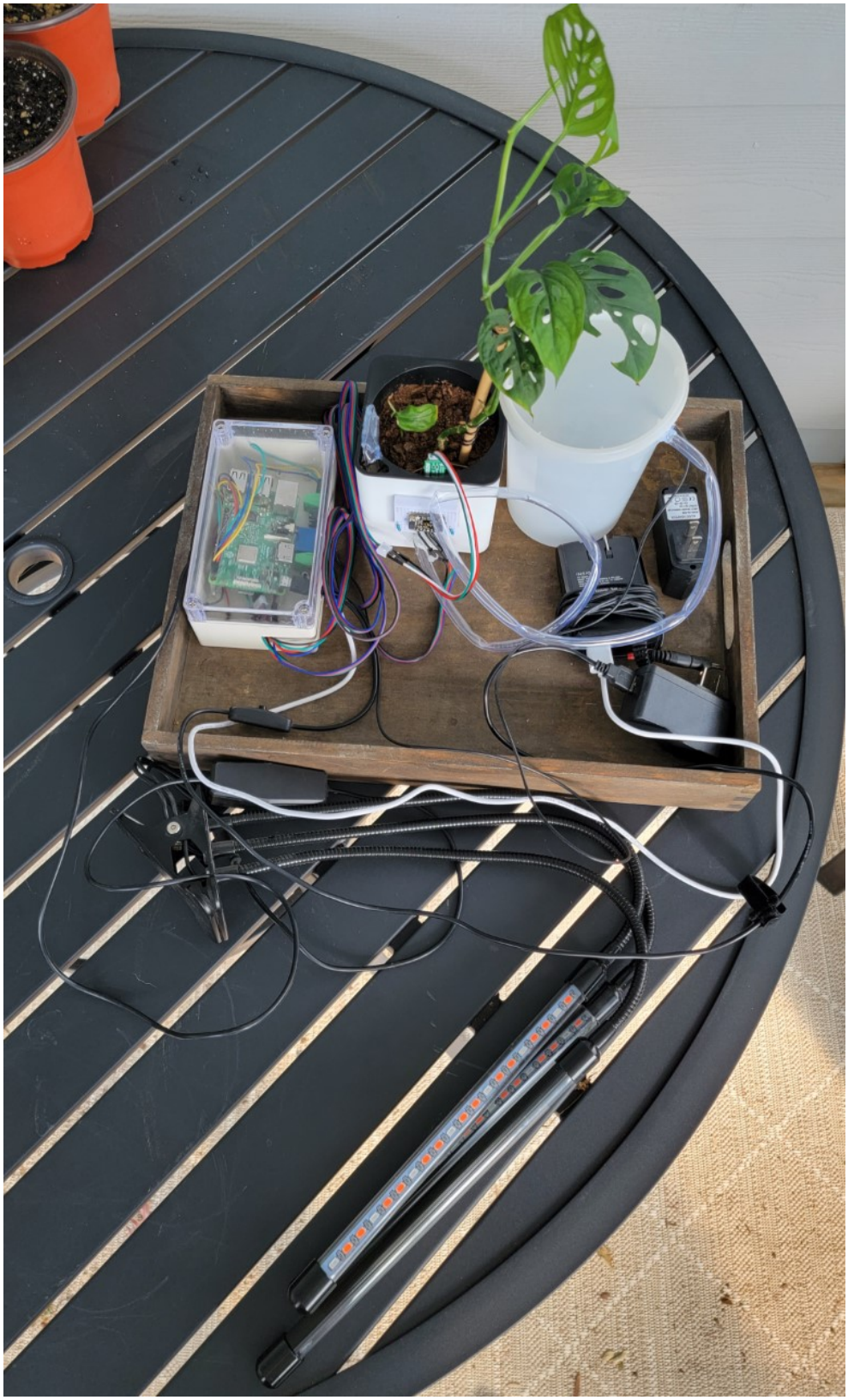


Figure 1: The Raspberry Pi connected to the sensors, light, and water pump.

Figure 2: Ouplant in the greenhouse with the grow light and water pump below.

For the mobile portion we used an android smartphone which acts as the user interface for viewing the plant states, setting the watering schedule, and sending control commands to the controller via the firebase real-time database. It was programmed in Android Studio with Java.

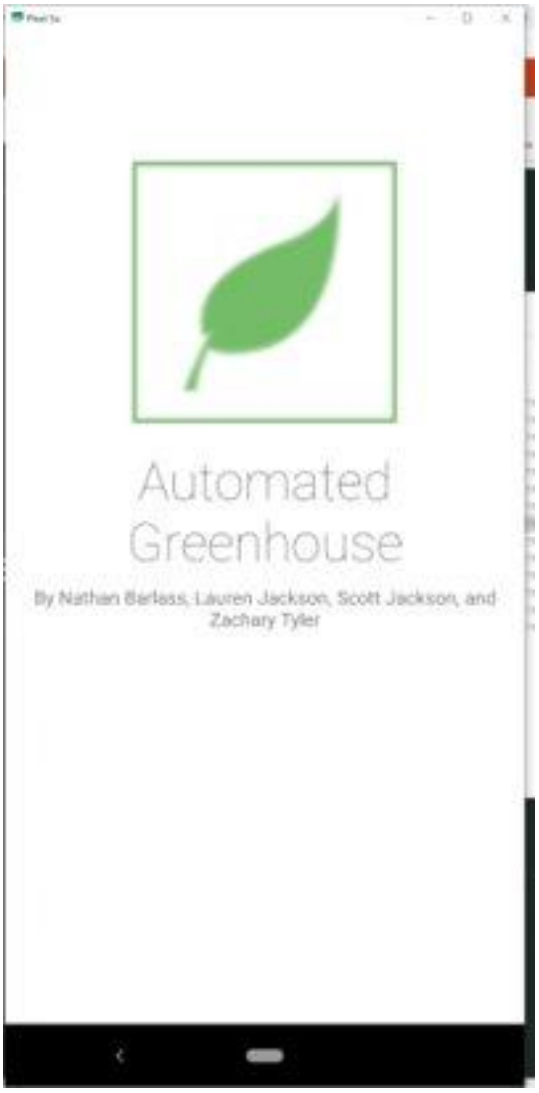


Figure 3: The home screen of our mobile application.

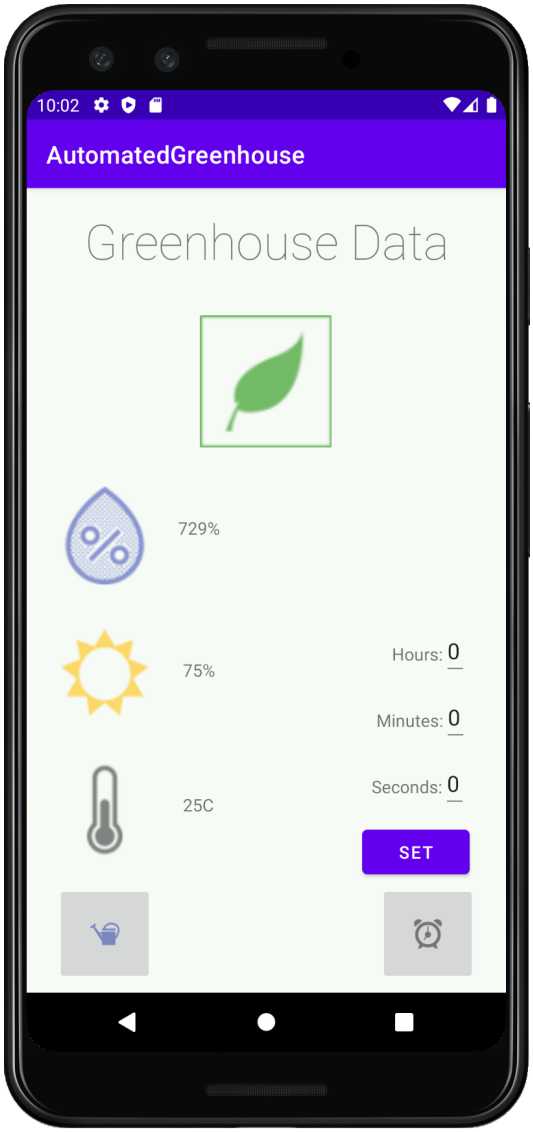


Figure 4: Our application displaying moisture, sunlight and temperature levels. It also has a button for setting a watering schedule, a button to manually water our plant, and not pictured is a button to turn on our light. The light level is based on a scale of 0 to 100% and the water is 0 to 100% as well. Zero representing no light and very dry respectively. The temperature is Celsius.

For the database portion, we utilized Realtime Google Firebase to act as a communication interface between the android smartphone and the raspberry pi.

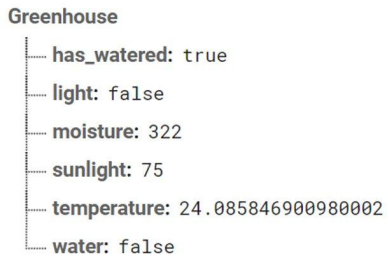


Figure 5: This shows the database using Google Firebase and shows our data collected from our sensors and Raspberry Pi.

Overall, this project was a success and we achieved our main goal of being able to access the data of our greenhouse through a mobile application and being able to water our plant remotely. This project helped us learn how to work well together as a team and also helped us realize what our capabilities are. We could definitely improve upon this project in multiple ways and orient it to a larger scope that could potentially aid with food sustainability. With us being able to do it on a smaller scale, I think it could easily be replicated to automated an entire garden. This could have a huge impact on farmers and other agricultural workers.

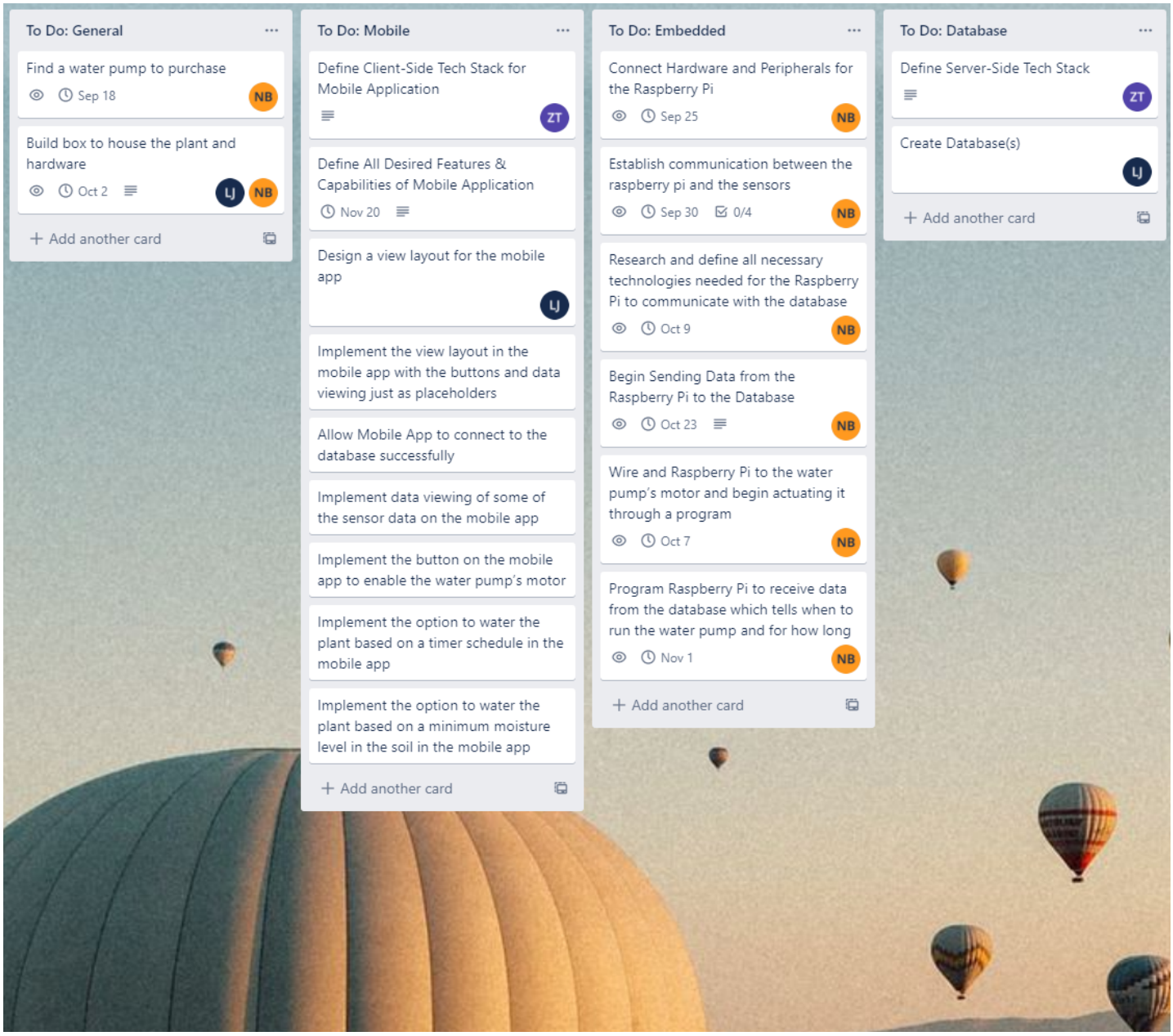
In the future we plan on adding a heat lamp and grow light to offer the same function when the sensors alert the application that the sunlight or temperature is too low. If we had time, we could have added pH sensors and pressure sensors to improve upon this project. The pH sensor would allow the user to know when they need to fertilize or put different soil in the pot based on mainly potassium, phosphate, and nitrogen levels. The pressure sensor would allow for knowing when the plant needed to be repotted based on how much the roots are growing. These would have also improved upon other implementations we have seen.

4.3 Risks

Risk	Risk Reduction
Code/hardware errors result in plant casualties	Revising each others code and testing the hardware before use
User error resulting in improper sensor readings	Including detailed instructions in the app for proper setup of the greenhouse

4.4 Tasks

For the tasks we used Trello so a screenshot of the created tasks is shown below:



4.5 Schedule

For the schedule we also used Trello which a screenshot is provided above.

4.6 Deliverables

- Design Document that contains details about the designs of the greenhouse such as the physical hardware, the raspberry pi and the software used on it, and details about the mobile application.

- Android studio project that contains the working java and xml files for the mobile app that controls and displays the greenhouse information
- The database schema for our Firebase database
- The Java code used for controlling the greenhouse and interacting with the database through the raspberry pi
- The Python code for programming the raspberry pi
- Final Report detailing the design and features of the automated greenhouse
- Zip file containing all reports and code

5.0 Key Personnel

Nathan Barlass- Barlass is a Senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has worked as a controls engineering intern and has/is completed/taking relevant courses such as Networks, Database Management, and Mobile Programming. He is responsible for programming the raspberry pi to read sensor inputs and send them to the database and to actuate the water pump based on the commands read from the database.

Lauren Jackson- Jackson is a Senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. She has/is completed/taking relevant courses such as Database Management, Networks, and Technical Writing. She is responsible for creating and managing the database and task list and writing the majority of the documentation.

Scott Jackson- Jackson is a Senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed relevant courses such as Networks, Database Management and currently in Mobile Programming, He is responsible for developing the mobile application that will interact with the database to run the raspberry pi.

Zachary Tyler- Tyler is a Senior Computer Science major in the Computer Science and Computer Engineering Department at the University of Arkansas. He has completed relevant courses such as Networks, Database Management, and currently in Mobile Programming. He was responsible for developing the mobile application portion of the project that interacts with the database.

6.0 Facilities and Equipment

Description of all facilities and/or equipment required and/or utilized for the complete project.

- The facilities needed to complete this project will be largely working remote from our places of residence. Other facilities that will be used include buildings on campus at the University of Arkansas for team meetings.

- The equipment required for the complete project include:

- Raspberry Pi
- Water pump
- Small greenhouse
- Waterproof enclosure (for Raspberry Pi)
- Sensors
- Plant, soil, water
- Smartphone

7.0 References

[1] T. Bhuvaneswari and J. T. H. Yao, "Automated greenhouse," IEEE International Symposium on Robotics and Manufacturing Automation (ROMA), IEEE, 2014