AWS based Crop Care System

Chengyan Liu, Zhen Lin
Department of Computer Science
Binghamton University

ABSTRACT

Nowadays we have more crops growing than ever, but as crops need to be taken care of and in a world full of modern technology, we need a modern way to care about the crops. Here we have proposed an AWS-based crop care system that can monitor the status of crops with the use of raspberry pi as a node to fetch data wirelessly.

KEYWORDS

Raspberry pi, AWS, corp

1 INTRODUCTION

There are many factors that affect the growth and development of crops, including light, temperature, humidity, and soil nutrients, and it is quite laborious to track those factors. Therefore, we can establish a crop care system to reduce unnecessary boring manual labor and use this system to collect information.

2 HARDWARE DESIGN

We used a Raspberry Pi 4 (fig. 1) in order to lower the power consumption while maintaining a wide range of connectivity. By using Raspberry Pi 4 we will have WIFI to transmit data and I2C and a bunch of GPIO to connect different kinds of sensors. As for the soil sensor, we used Adafruit STEMMA Soil Sensor (fig.2) since it has I2C connectivity and it’s using capacitive measurement that it won’t oxidize and therefore last even longer. We used a AM2320 Digital Temperature and Humidity Sensor (fig.3) to measure environment temperature and humidity and it also has an I2C interface.

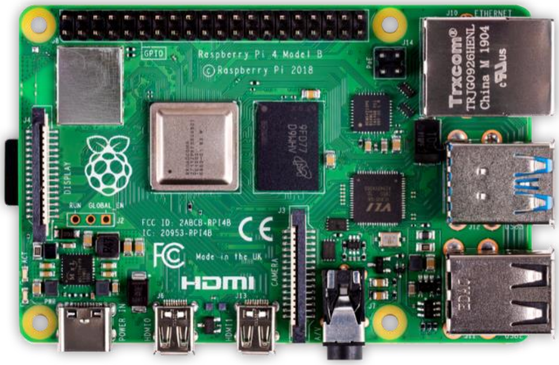

Fig.1 Raspberry Pi 4

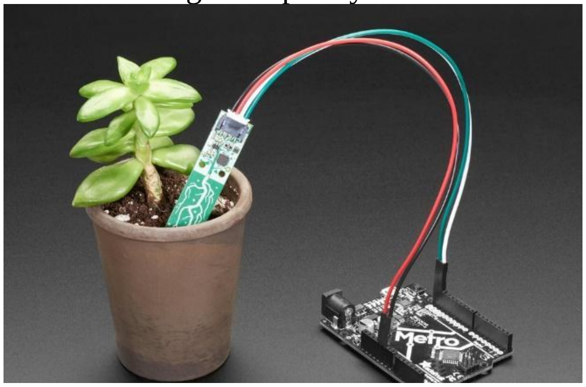

Fig.2 Adafruit STEMMA Soil Sensor

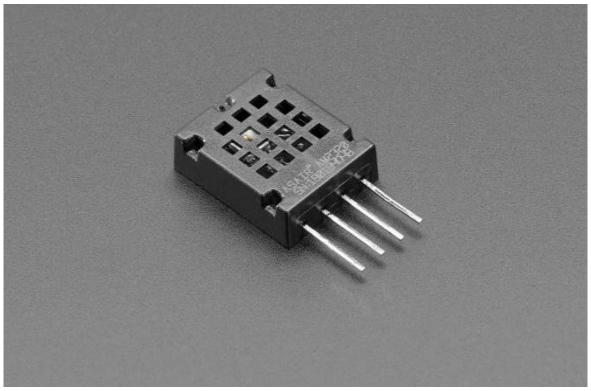

Fig.3 AM2320 Digital Temperature and Humidity Sensor

3 SOFTWARE DESIGN

For the server side, we implemented an API server using Spring Boot Framework. Spring Boot Framework is an open source Java-based framework used to create a micro service which allows the developers to develop and deploy services independently . And, Spring Boot is very suitable for building stand-alone and production ready spring applications. We can develop our API service first regardless of our production environment, then use Docker Container to deploy our service.

Also, we implemented a dashboard using React.js to display out data. React is a JavaScript library for building user interfaces. It is painless to use React to build interactive UIs, because we can design simple views for each state in our application, and React will efficiently update and render just the right component when our data changes. React is also component-based, which means we can build encapsulated components that manage their own state, then compose them to make complex UIs.

To store our data persistently, we used MySQL to store out data. Unlike other services like Spring Boot, MySQL is a stateful service, which means we need to store its data persistently. Therefore, deploying MySQL services using Docker Container may not be a good idea. So, we decided to build our MySQL service on Amazon RDS(Relational Database Service) Platform.

4 IMPLEMENTATION

For the sensor side we used python3 as program language and raspbian as the operating system. Both of them are widely used so they’re reliable for such usage. At the hardware side we will read raw data from the sensor periodically and transfer them into a data package as a JSON string and post them to the api on the cloud.

Once the API server receives the adding data points requests, the Spring Boot will help us interpret its parameters. After that, we can process the data, and finally store them onto the MySQL Database.

To display our data, the dashboard will send a request to get data from the API server. The API server will pull the data from the MySQL Database and then return them to the dashboard. Of course, we also develop some functions to group the data by time period, sort the data, and format the data, which makes our output data more valuable and readable.

4 CONCLUSION

In this project we have created a AWS-based crop care system. The system will collect data periodically and send them to the cloud. A dashboard will display the data and show the trend of the past 24 hours.

ACKNOWLEDGEMENT

We would like to thank Prof. Mo Sha and TA Junyang Shi for helping us through the CS588 class. They gave us a lot of useful suggestions on this project.