

Final Presentation

ありがとうみんな!!!

Team-8: CuProSA

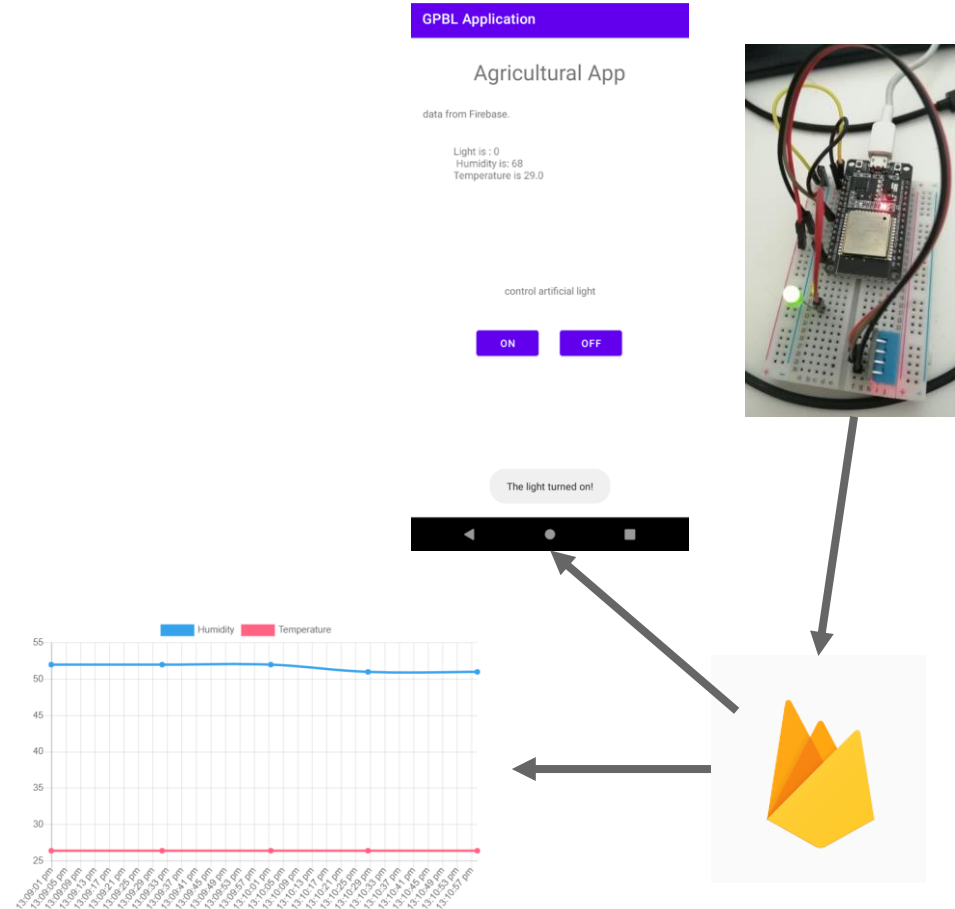
Tomoki Isoda	SIT, Japan
Zhong Ming Xiu	MCUT, Taiwan
Ng Tsu Yao	UCSI, Malaysia
Han-Aiman Rasyid	UTM-MJIIT, Malaysia

gPBL 2021 on Network Software
Mar. 9, 2021

Topic: IoT Agricultural Monitoring Application

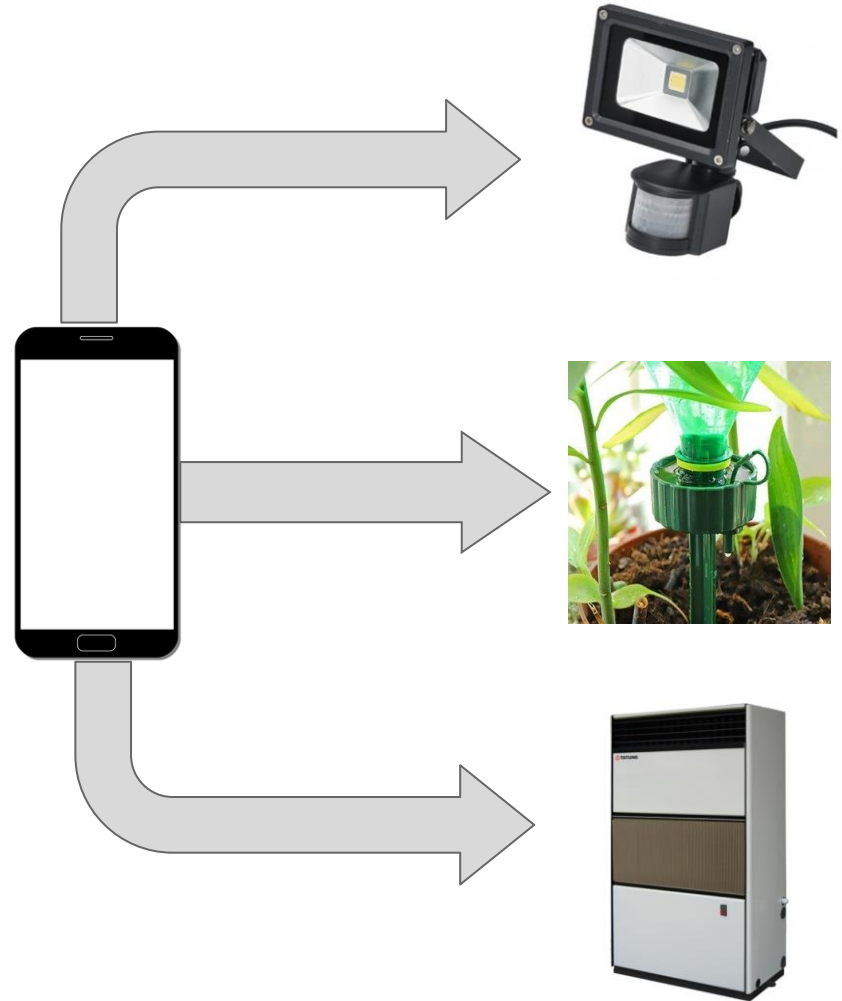
Concept:

- Monitor different readings of a plant crop (Humidity, lighting and temperature) from different location.
- Save readings in a database (Firebase) and can be accessed via website.
- Control devices from app or website.
- Send notification when something go wrong.



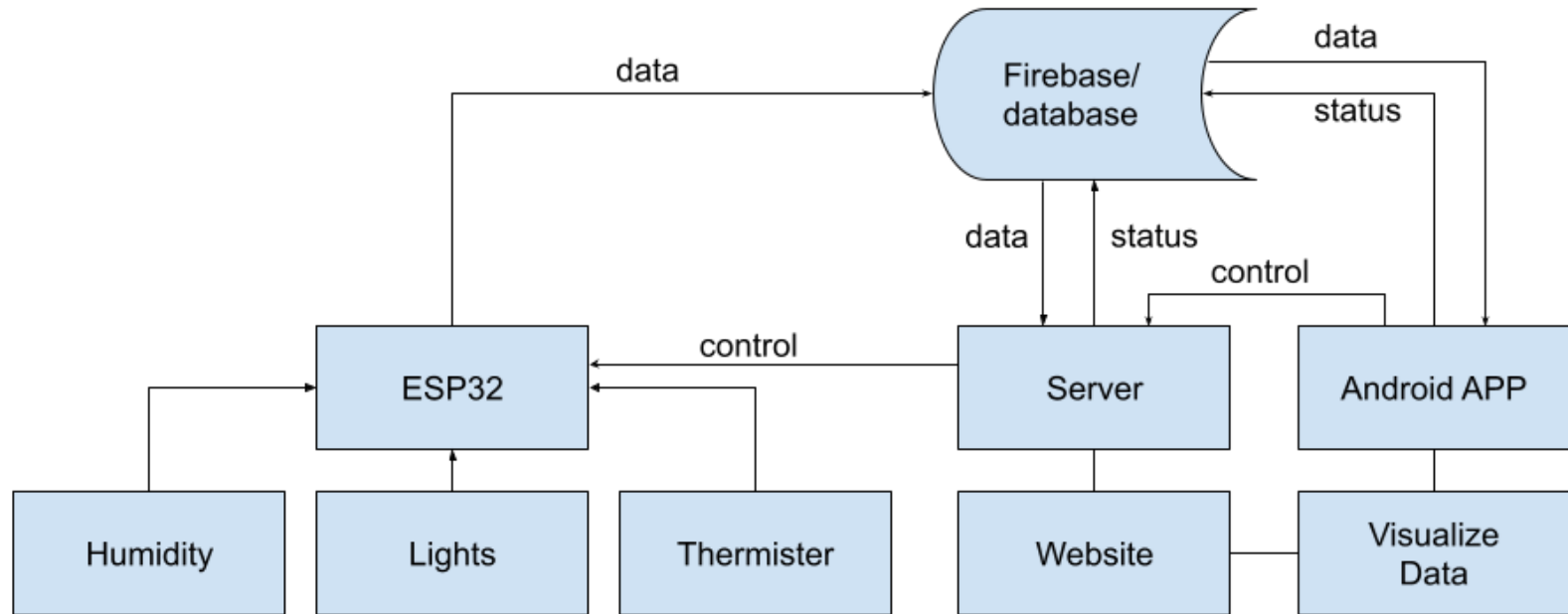
Motivation

- People can't stay besides the plants all the time.
- It's also easier to know the situation of the plants when there is a sensor of detecting humidity and temperature.
- People can control the device such as lights, waterer, and air conditioner by their phone far away or near the plants.
- As COVID-19 increase the opportunities of doing remote work, so they also have more opportunities to control various things remotely.



Structure

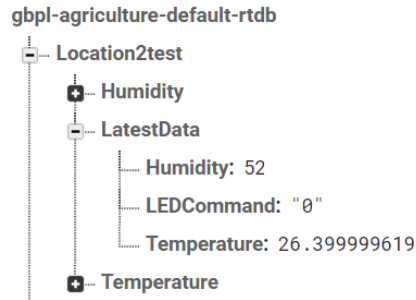
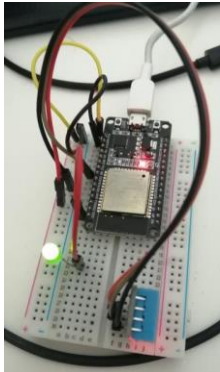
- Flowchart of our system



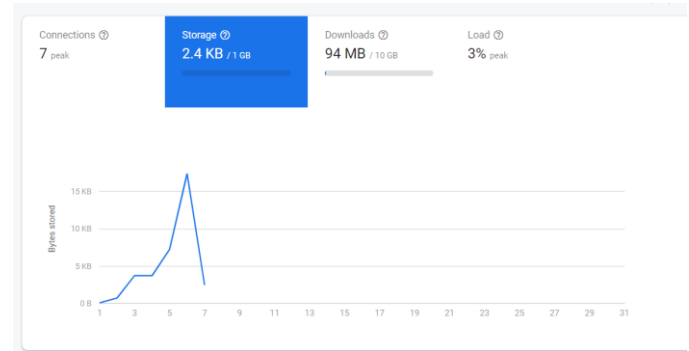
ESP32

Using an ESP32 connected to Firebase Database for data storage.

- Able to send data at a set time (default 1 minute) to firebase
- Data is stored historically on firebase from inception until now
- Occasional bug with storing to LatestData
- Able to receive data from firebase to do task (turning LED on/off) from android app (video demo later)



Latest Data for App



Firebase Data Usage

Website

- Receive and plot data.
- Fill same data while data no changing in 30 seconds.
- Data saved 100 points.
- Text color will change when data is abnormal.
- Customize gate value.

Humidity is: 51 abnormal

Gate value: 30 to 40

Temperature is: 26.10 abnormal

Gate value: 28 to 40

Agriculture WebSite

Humidity is: 49

Gate value: 40 to 70

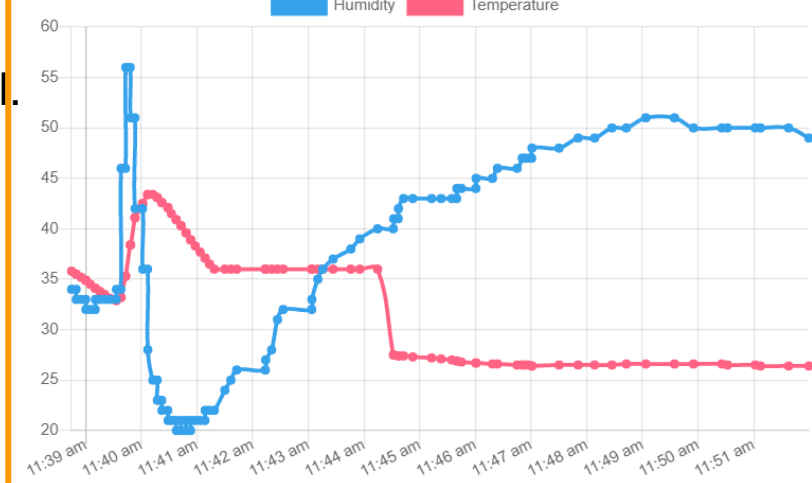
Temperature is: 26.40

Gate value: 20 to 30

Real-time

The Light is OFF

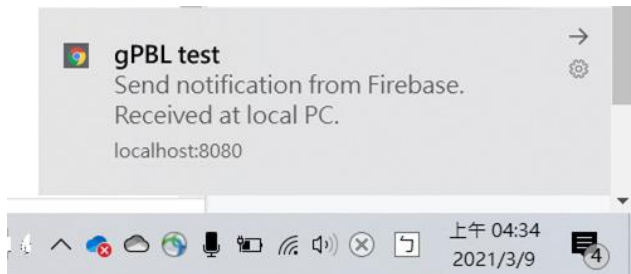
History(opened web for a while)



Data from sensor being heated by lighter

Firebase

- These features can be implemented if there is HTTPS. (succeed in local)
- People need to give the username that server can notify and distinguish everyone. (use cookies)
- It can send notification to PC via website when the website is closed. (need notification permission)
- It can send notification via App, though.



name: "Ji"
token: "eMFh5X5SGuSnH-KTN8v4"

localhost:8080 顯示

please give me a username:

確定 取消

```
messaging.getToken({vapidKey: vapidKey}).then((currentToken) => {  
  if (currentToken) {  
    var u = getCookieByName('user');  
    var name = u;  
    if(!u){ //check if user have registered a name.  
      name = prompt("please give me a username:");  
      document.cookie = 'user='+name;  
    }  
    //Send to Firebase  
    messageRef.child(name).set({  
      name: name,  
      token: currentToken});  
  } else {  
    // Show permission request UI  
    console.log('No registration token available. Request permission...');  
    // ...  
  }  
}).catch((err) => {  
  console.log('An error occurred while retrieving token. ', err);  
  // ...  
});
```

App(1)

GPBL Application

Agricultural App

data from Firebase.

Light is : 1
Humidity is 35
Temperature is 25

control artificial light

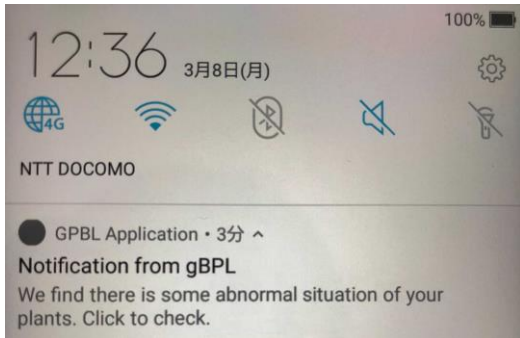
ON

OFF

Go to the website

MOVE

The light turned on!



Lower than 10% or Higher than 90%

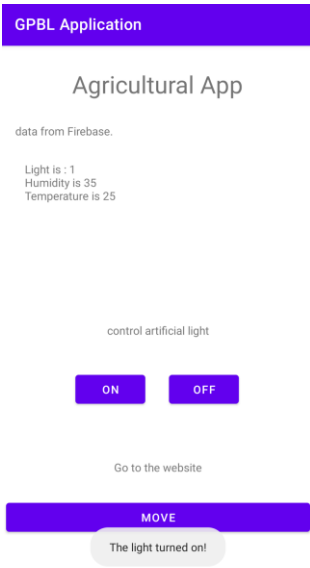
Alert: Abnormal humidity was observed

Lower than 20d or Higher than 50d

Alert: Abnormal temperature was observed

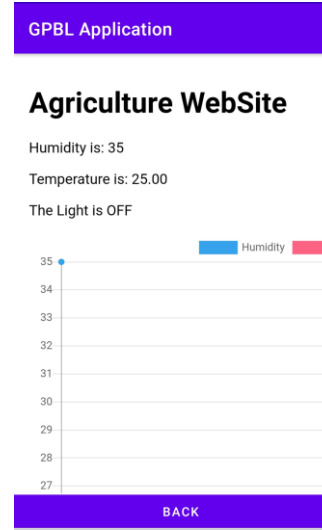
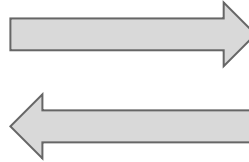
- Easy to see real-time data of sensors(Light status/Humidity/temperature) and change automatically.
- Change the light status by using the button(ON/OFF).
- When the light turned on/off, client receive the notification(Toast).
- Notified by the server about abnormal situation of the plants when the app is in the background.
- Notified by toast about abnormal situation of the plants when the app is in the foreground.

App(2)



usual screen

Push the button



web page screen

- See the web page from the app by pressing the button at the bottom of the screen.
- Return to the usual screen from the web page screen by pressing a button.
- See the detailed graph on the web page while opening and do the same things as the website from the app.

Demonstration(App & esp32 with Firebase)



- Communication between esp32 in Malaysia and android device in Japan is successful.

[illegible]

*Text color changing only occurs after data updated.

Conclusion

● Feature

- ESP32:
 - Send data to firebase, received control from App.
- Website:
 - Show the real-time data, make a signal about abnormal situation.
- App:
 - Show the real-time data and web page, control the device(light), received notification.
- Firebase:
 - Store data, notify users manually, transfer the control signal.

● Future

- Notify users automatically (can easily resolved by upgrading firebase plan).
- Not only the light, but also waterer and air-conditioner can be controlled.
- History search and analysis, which can be used by AI controlling.
- SQL database can handle history data more easily than JSON database.