ST-08

# ระบบอินเทอร์เน็ตของสรรพสิ่งสำหรับควบคุมดูแลสวนผ่านแอพพลิเคชั่น

# IoT system for smart garden application

กรกนก แก่นสุวรรณ์[1] กุลณัฐ หนชัย[2] ณัฐนันท์ สิงห์สังฆ์[3] ธนสรณ์ อินทร์สุวรรณ์[4] และณัฐกมล นาโควงค์[5]
Kornkanok Kaensuwan[1], Kullanat Honchai[2], Nuthanun Singsang[3], Thanasorn Insuwan[4],
and nuttakamol-nakowong[5]
[1,2,3,4,5]ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยศรีปทุม
อีเมล: zmacger@gmail.com

**บทคัดย่อ**

การดูแลสวนโดยส่วนใหญ่จำเป็นที่จะต้องมีคนดูแลตลอดเวลา ทำให้เวลาส่วนใหญ่จะต้องหมดไปกับการดูแลสวน ภายหลังมีอุปกรณ์ตั้งเวลาเพื่อการรดน้ำ ซึ่งอาจสามารถลดเวลาการดูแลสวนน้อยลง แต่อาจไม่ทราบข้อมูลอื่นเช่นปริมาณน้ำที่ใช้และเวลาที่ตั้งเหมาะสมในการรดน้ำจริงหรือไม่ ทางผู้พัฒนาจึงมีการพัฒนาระบบIOT ซึ่งเกี่ยวกับการควบคุมดูแลสวนผ่านแอพพลิเคชั่น(Smart Garden) โดยที่ภายในแอพพลิเคชั่นนั้นจะมีในส่วนการเปิดปิด ตั้งเวลา ตั้งค่าอุณหภูมิความชื้นและดูสถิติโดยใช้รูปแบบ Data Visualization เป็นต้น ทั้งนี้ผู้พัฒนาใช้รูปแบบการวิจัยเชิงปฏิบัติ เพื่อการศึกษา โดยกลุ่มตัวอย่างเป็นคนสวน ภายในมหาวิทยาลัยศรีปทุม

ผลวิจัยจากการทดลองภาคสนาม พบว่า การทำงานในทุกๆ ด้าน สามารถใช้ได้จริงโดยควบคุมแอพพลิเคชั่นผ่านมือถือ แต่มีเพียงผู้พัฒนานั้นที่ลองใช้แล้ว แต่ผู้ใช้จริงๆ ยังไม่ได้เปิดให้ใช้นอกจากนี้มีเรื่องของเซิฟเวอร์กับ Thingboard (IoT Platform)เนื่องจาก Thingboard ใช้ memory usage ค่อนข้างมากเมื่อเปิดใช้งานจึงทำให้เวลาลง Thingboard และเปิดใช้งานจึงทำให้เซิฟเวอร์ไม่ทำงาน ข้อเสนอแนะเพิ่มเติมคือ การที่ Thingboard สามารถทำงานร่วมกับเซิฟเวอร์ได้ โดยที่เซิฟเวอร์อาจจะต้องมีการเพิ่มสเปค ทั้งนี้จะใช้ open landscape ที่เป็นบริการโฮสต์เซิฟเวอร์ ในการพัฒนาระบบ

**คำหลัก:** ควบคุมดูแลสวนผ่านแอพพลิเคชั่น, เทคโนโลยี Internet of things, Data Visualization

**Abstract**

Caring for the garden is mainly necessary to be cared for at all times. As a result, most of the time is spent caring for the garden. Later, there is a timer for watering, which may reduce the time of care for the garden, but may not know other information such as how much water is used and whether the time is set to be watered. Therefore, the developer has developed the IoT system. This is about supervising the garden through the application (Smart Garden). Within the application, there is an on and off section. time Set temperature, humidity, and view statistics using data visualization formats, for example. The developers used a practical research model for education, with the sample being gardeners. Inside Sripatum University Research results from field trials It was found that every aspect of an operation can be used by ontrolling the mobile application, but only that developer has tried it, but the actual user has not yet enabled it, in

addition, there is a matter of server and thing board (IoT platform) since Thingboard uses quite a lot of memory usage when it is activated, thereby making the time to land on thingboard and activating it, thereby causing the server not to work. An additional suggestion is that Thingboard can work with the server. The server may need to be added specifications, which uses open landscape, which is a host server service, to develop the system.

**Keywords**: control system for garden on application, Internet of things, Data Visualization

**บทนำ**

การดูแลสวนนั้น หากเป็นสวนขนาดเล็ก การดูแลรักษานั้นจะค่อนข้างใช้เวลาน้อย ด้วยจำนวนต้นไม้หรือพื้นที่สวนที่ไม่ได้กว้างมาก แต่หากเป็นสวนที่มีขนาดใหญ่ การดูแลสวนจะลำบากยิ่งขึ้น เนื่องด้วยสวนมีพื้นที่ที่ค่อนข้างกว้างและมีหลายแหล่ง จึงทำให้เสียเวลาในการดูแล แม้ว่าต่อมาอาจจะมีการติดตั้งอุปกรณ์ Timer เพื่อช่วยตั้งเวลาในการรดน้ำ แต่การจะปิดนั้นต้องไปปิดที่ตัววาล์วหรือตู้ไฟที่ทำหน้าที่ควบคุม และตัวอุปกรณ์ไม่สามารถรับรู้สภาพดินได้ว่าควรให้น้ำ ณ เวลานั้นหรือไม่

ดังนั้นจึงทำให้ผู้พัฒนานั้นทำเกี่ยวกับระบบอินเทอร์เน็ตของสรรพสิ่งดูแลสวนด้านแอพพลิเคชั่น จากเดิมคือมีสวิสต์เปิดปิดหรือ Timer เพื่อตั้งเวลาด้วยตัวเอง แต่จะพัฒนาด้วยการที่สามารถมาทำสิ่งเหล่านี้ในแอพพลิเคชั่นได้ไม่ว่าจะเป็นการเปิดปิด การตั้งเวลา การยืนยันสิทธิ์การใช้งาน การตั้งค่าอุณหภูมิความชื้นที่เหมาะสมเพื่อให้ระบบรดน้ำทำงานตามที่ต้องการได้ เป็นต้น เพื่อให้การดูแสสวนขนาดใหญ่นั้น สามารถดูแลได้ง่ายขึ้นโดยที่ไม่จำเป็นต้องไปดูจุดเปิดปิดตามวาล์วหรือตู้ไฟ แต่สามารถดูและเช็คอุปกรณ์หรือค่าเปิดปิดต่างๆ ผ่านแอพพลิเคชั่นได้ นอกจากนี้ยังมีการทำตัว Web Application เพื่อที่สามารถดูข้อมูลในการใช้ว่า มีการใช้ไฟฟ้าไปแล้วเท่าไหร่ และดูสถิติการใช้งาน จำนวนกระแสไฟที่ใช้ ซึ่งจะนำไปคำนวณเกี่ยวกับค่าน้ำค่าไฟด้ล่วงหน้าว่า เดือนนี้ใช้ไปเท่าไร แล้วจะวางแผนการใช้น้ำและไฟต่ออย่างไร

**วัตถุประสงค์**

1.เพื่อใช้ในการบริหารและควบคุมการจัดการในสวนหย่อมขนาดใหญ่ด้วย Internet of Things (IOT)

2.เพื่อให้ผู้ใช้สามารถตรวจสอบปริมาณน้ำกับพลังงานไฟฟ้าที่ใช้ในสวนหย่อม

**วิธีดำเนินการวิจัย**

- การออกแบบอุปกรณ์และระบบ
- ตรวจสอบสถานที่กับระบบเก่าที่ใช้งานอยู่
- การเก็บข้อมูลในส่วนของปัญหาการใช้งานของระบบเก่าจากคนสวนใน มหาวิทยาลัยศรีปทุมพร้อมถึงสิ่งที่ต้องการ

**อุปกรณ์การวิจัย**

โดยแบ่งได้เป็น 2 ส่วนได้แก่ software และ hardware ดังนี้

Software

- Visual Studio Code
- Figma
- Arduino

- phpMyAdmin
- Node red
- apache2
- putty
- Bootstrap

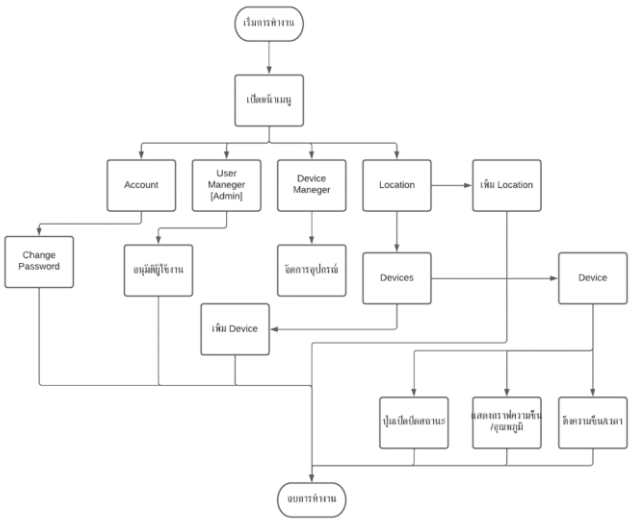
Hardware

- Sonoff 4CH PRO R2
- Breaker
- Magnetic conductor
- Overload
- หม้อแปลงไฟ 24V
- สายไฟ
- Soil Moisture Capacitive
- Kidbright

ทางผู้พัฒนาได้สอบถามเกี่ยวกับรายละเอียดและปัญหาที่ผู้ดูแลสวนได้เจอกับปัญหาการดูแลสวนภายในมหาวิทยาลัยศรีปทุมด้วยตัวเองและระบบเทคโนโลยีการดูแลสวนแบบเดิม ที่ก่อนหน้านี้เคยมีการติดตังมาแล้วแต่ก็ยังมีปัญหาอยู่ ผู้พัฒนาจึงสอบถามต่างๆ โดยแบ่งเป็น 2 ส่วนใหญ่ คือ 1) รายละเอียดของอุปกรณ์เดิมที่มีอยู่ในส่วนองค์ประกอบจำนวน หลักการทำงาน และสถานที่ติดตั้ง และ 2) ปัญหาที่พบระหว่างในการดูแลสวนและถ้าสามารถเพิ่มเติมสิ่งที่ทำให้ผู้ดูแลสวนสะดวกในการดูแลขึ้นจะต้องการพัฒนาปรับปรุง จากปัญหาและรายเอียดที่ได้รวบรวมไว้จึงได้คิดที่จะพัฒนา Application ที่ไว้ควบคุมและตรวจเช็คระยะเอียดทุกอย่างไว้ จากที่ระบบเก่านั้นจำเป็นต้องเข้าควบคุมที่ตู้อุปกรณ์โดยตรง และด้วยตัวอุปกรณ์ถูกติดตั้งไว้ทั่วมหาวิทยาลัย จึงลำบากมากที่จะควบคุมอุปกรณ์ทุกตัวนั้น

## การออกแบบ

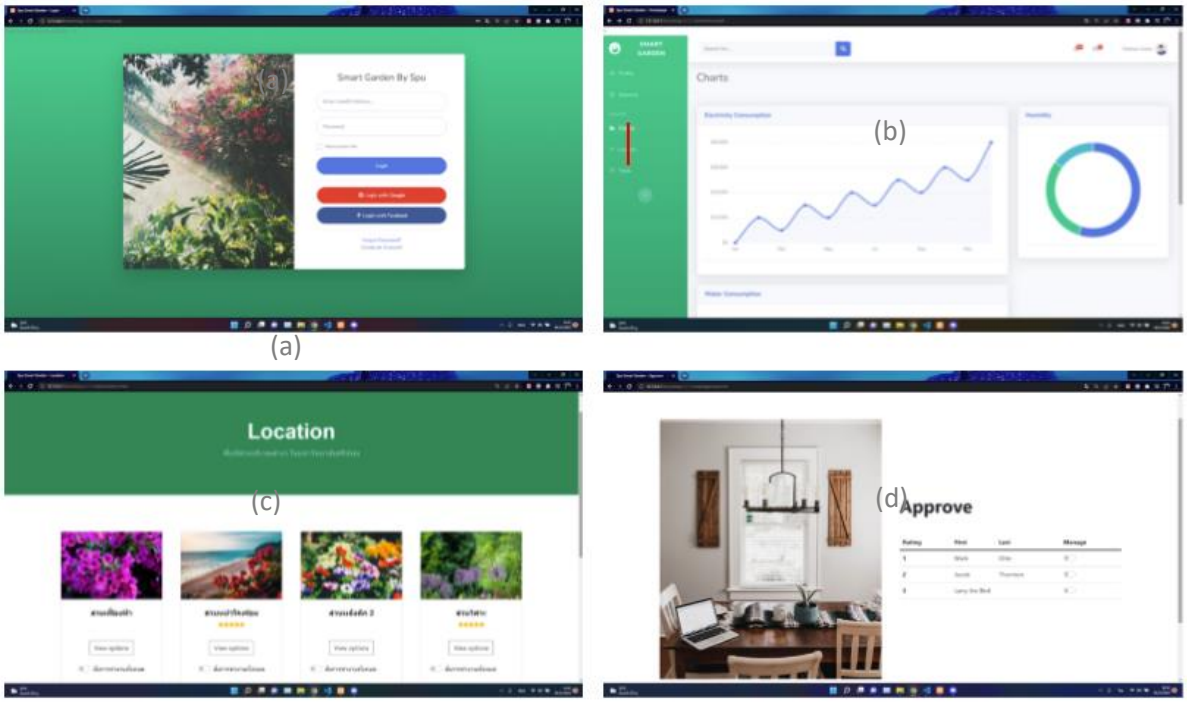
### Flowchart




**ภาพที่ 1** Flowchart application

ภาพที่ 1 คือภาพรวมการทำงานของโปรแกรมจะมีตั้งแต่เริ่มโปรแกรม ล็อกอินเข้ามาก็จะเห็นในส่วนของหน้าต่างเมนู ตามที่ออกแบบ และเมื่อกดเข้าไปก็จะพาเข้าไปยังฟังงก์ชั่นนั้นๆ เช่นหากเข้า Location หน้าโปรแกรมก็จะแสดงตัวอุปกรณ์นั้นมีการติดตั้งอยู่ที่ใดบ้าง และเลือกอุปกรณ์เพื่อเข้าไปใช้งาน

**web application**



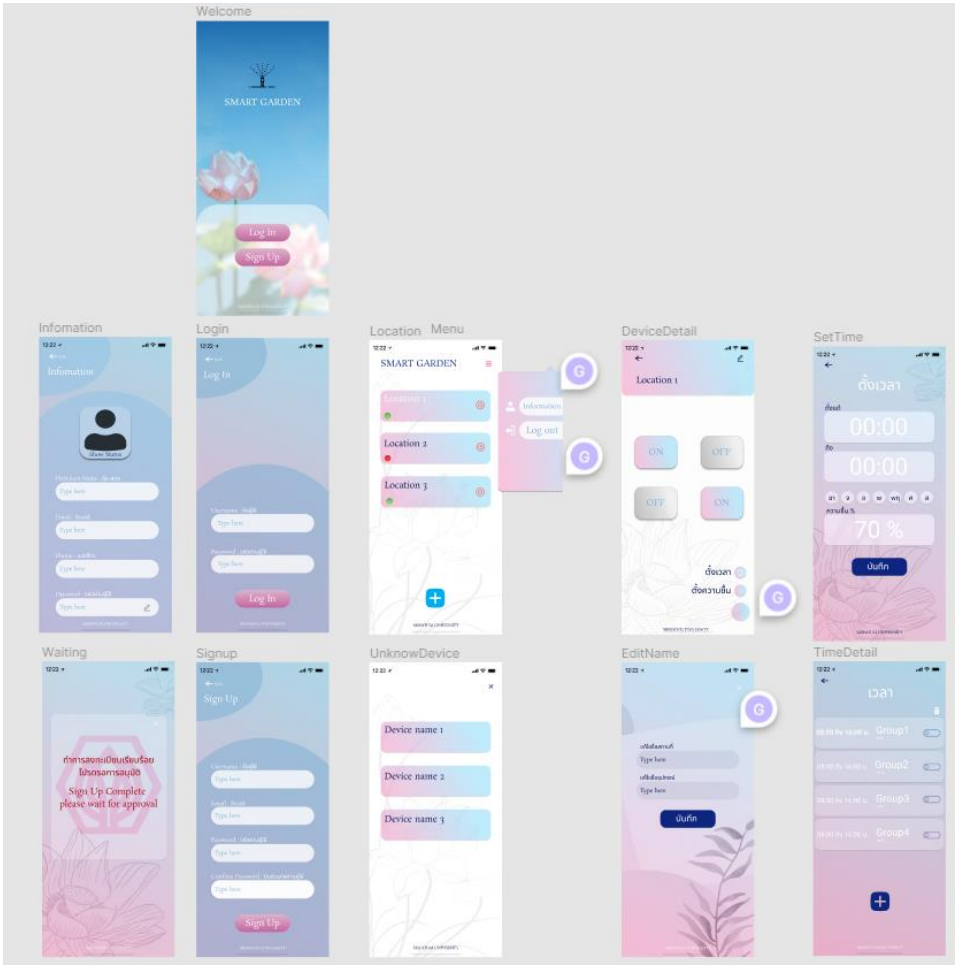

**ภาพที่ 2** web application 1

web application ออกแบบมาเพื่อใช้สำหรับ ผู้บริหารหรือหัวหน้างานที่ได้รับอนุญาต ซึ่งล็อคอินเข้ามาจะมีฟังก์ชั่นที่กำหนดไว้ให้เลือกหลายอย่างไม่ว่าจะเป็น หน้าการนำเสนอภาพข้อมูล (data visualization) ของค่าต่างๆเช่นอุณหภูมิ ความชื้น, ค่าน้ำค่าไฟที่ถูกใช้ไปหรือเวลาของผู้ใช้งานที่ใช้งานอุปกรณ์ว่าแต่ละคนได้ใช้งานไปเท่าไหร่, หน้าสั่งใช้งานอุปกรณ์ และ หน้าให้สิทธิในการเข้าใช้งานของสมาชิกต่างๆ

**ภาพที่** 3 web application 2

จากภาพที่ 3 เป็นตัวอย่างการแสดงกราฟข้อมูล โดยจะเรียงลำดับซ้ายไปขวา มีการใช้กระแสไฟฟ้าและคำนวณเป็นค่าไฟมานำเสนอในลักษณะกราฟ มีค่าใช้จ่ายสำหรับน้ำประปาที่ใช้และมีร้อยละของความชื้นในดิน

UX/UI Design

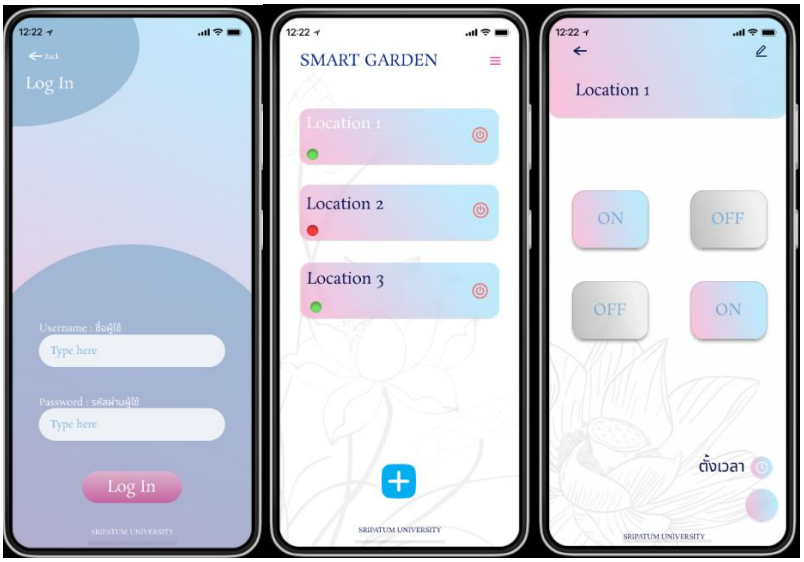



**ภาพที่ 4** UX/UI Design Mobile Application

การออกแบบหน้า Mobile Application ด้วยโปรแกรม โดยออกแบบให้ตรงตาม Requirement หน้า Application จะประกอบด้วยหน้าการเข้ารหัส หน้าการสมัครสมาชิก หน้าโปรไฟล์ หน้าแอดlocation หน้าเลือกอุปกรณ์ และหน้าใช้งานอุปกรณ์

**ภาพที่ 5** Mobile Application 1

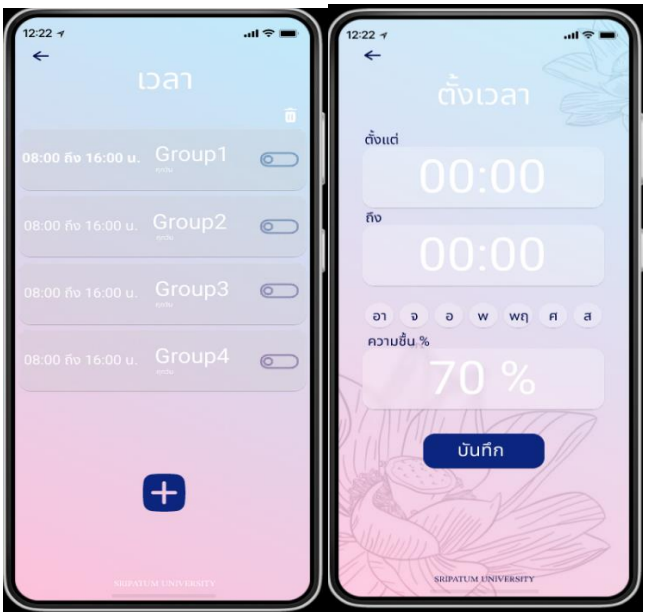
Mobile Application ส่วนนี้จะแตกต่างจากในส่วนเว็บไซต์ตรงที่ถูกออกแบบมาให้สำหรับ พนักงานหรือผู้ดูแลสวนทั่วไปใช้งานเพื่อควบคุมอุปกรณ์ต่างๆ สำหรับแอพพลิเคชันในมือถือนั้นจะจำกัดฟังก์ชั่นบางอย่าง ตัวอย่างเช่น การนำเสนอภาพข้อมูล ที่มีแค่ใน web เท่านั้นส่วนหน้าฟังก์ชั่นหลักๆที่มี คือ




(a) (b) (c)

**ภาพที่ 6** Mobile Application 2

จากภาพที่ 6(a)เป็นส่วนของ Log In จะให้ผู้ที่มี Account ที่ยืนยันสิทธิ์แล้วสามารถเข้าใช้งานได้ โดยผู้ที่สามารถเข้าถึงการยืนยันสิทธิ์จะต้องเป็น Admin หรือผู้ที่ได้รับอนุญาตเท่านั้น(b)เป็นหน้าที่แสดงจุดต่างๆที่ติดตั้งอุปกรณ์เอาไว้ และเพิ่มอุปกรณ์ต่างๆที่ยังไม่มีจุดติดตั้งและ(c)จะแสดงรายละเอียดของอุปกรณ์ มีปุ่มคำสั่ง ปิด/เปิด และตั้งเวลากับความชื้น

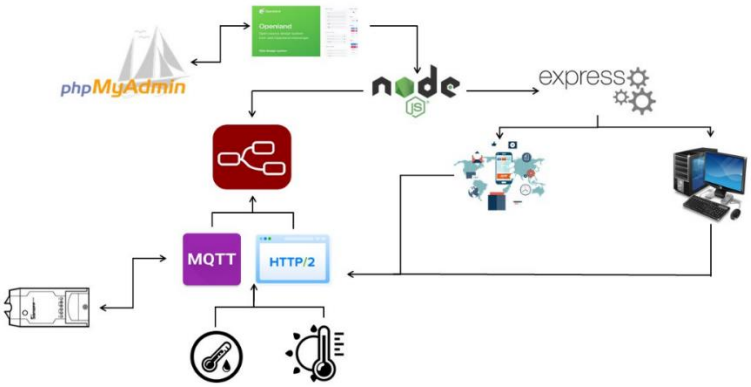



(a) (b)

**ภาพที่ 7** Mobile Application 3

(a)จะเป็นหน้าแสดงเวลา และ(b)เป็นการตั้งเวลากับความชื้นในการเปิด/ปิด โดยความชื้นจะตั้งเป็นเปอร์เซ็นต์ 1-100%

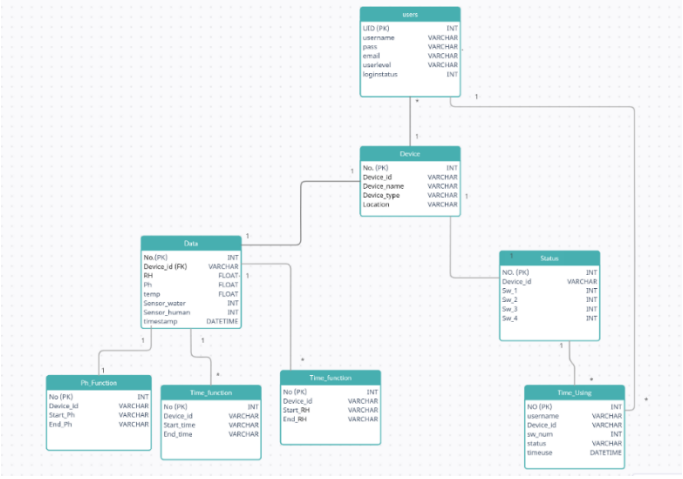
**Server**




**ภาพที่ 8** Server และการเชื่อมต่อ

จากภาพที่ 8ในส่วนของ SERVER ได้ใช้ open landscape ที่เป็น host server เพราะมี service มากมายทำให้ง่ายต่อการใช้งาน และใช้ Nodejs ที่เป็น JavaScript ในการสร้าง web server ขึ้นมา Node-Red และ express ใช้ในการสร้างเส้นทางในติดต่อ client ต่างๆ กับ server ซึ่งตัวอุปกรณ์และแอพจะติดต่อกันด้วย MQTT และ HTTP protocol เป็นหลัก

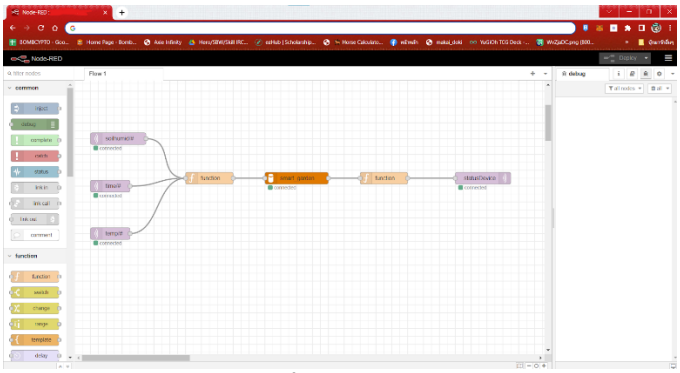
ER Diagram



**ภาพที่ 9** ER-diagram

จากภาพที่ 9 จะเป็นการออกแบบการเก็บข้อมูลในรูปแบบของ ER Diagram มีการกำหนด Datatype ที่ต้องใช้โดยจะมีตารางของหน้า ข้อมูลสมาชิกไวเก็บรายชื่อสมาชิกต่างๆ , ข้อมูลอุปกรณ์ทั้งหมดที่แอดเข้าระบบและตางรางนี้จะเป็นตารางแม่ในหลายๆฟังก์ชั่น ไม่ว่าจะเป็นตารางเก็บค่า input ต่างๆ , ตารางเก็บเวลาหรือความชื้นที่เหมาะสมไว้เปิด-ปิดตามกำหนด ต้องใช้ข้อมูลอุปกรณ์อ้างอิงทั้งสิ้น

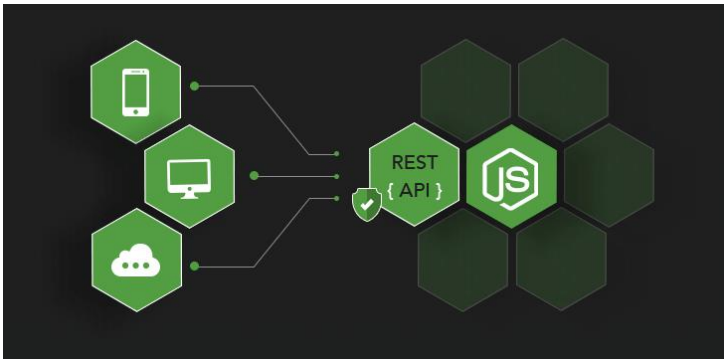
Node-red



**ภาพที่ 10** node-red

จากภาพที่ 10 มีการใช้ Node-Red เข้ามาช่วยในเรื่องของการนำข้อมูลจากอุปกรณ์ที่ส่งมา นำมาตรวจสอบค่าที่มีอยู่ในฐานข้อมูล เพื่อใช้ระบบการเปิด-ปิด ด้วยเวลา , ความชื้นและอุณภูมิ สะดวกยิ่งขึ้น

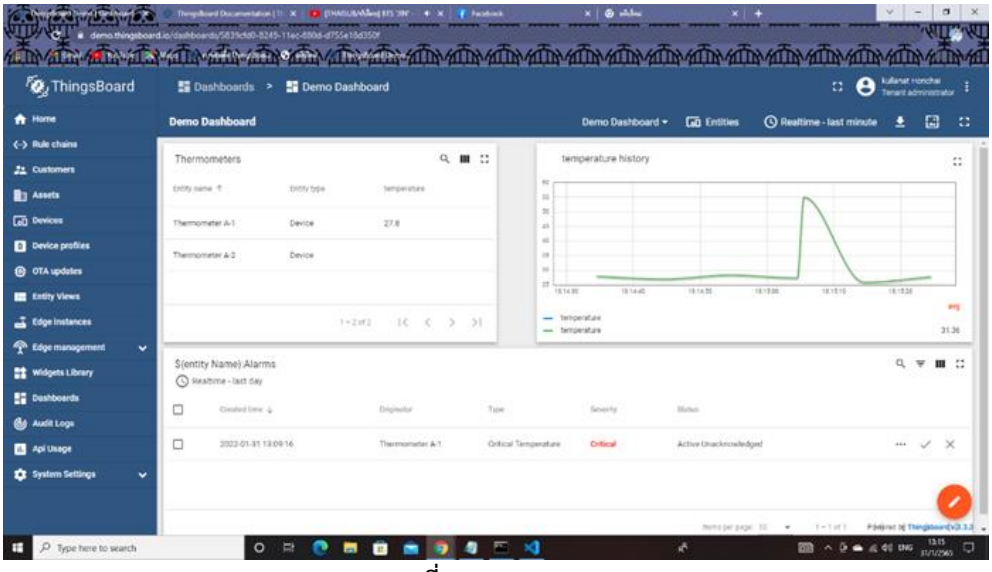
API




**ภาพที่ 11** API

จากรูป 11 ใช้ Express JS ด้วยการจัดการ routing ที่ง่าย ทำงานได้รวดเร็วและมีประสิทธิภาพ ส่วนนี้ มันถูกใช้สร้าง RestAPI ใช้ในการนำข้อมูลไปแสดงในหน้า Application โดยส่วนนี้จะไปเชื่อมกับส่วนของ Database เพราะ Database จะเป็นที่เก็บข้อมูลต่างๆรวมไว้ด้วยกัน และ API จะเป็นตัวกลางที่นำพาข้อมูลไปส่งให้กับส่วน ที่ต้องใช้

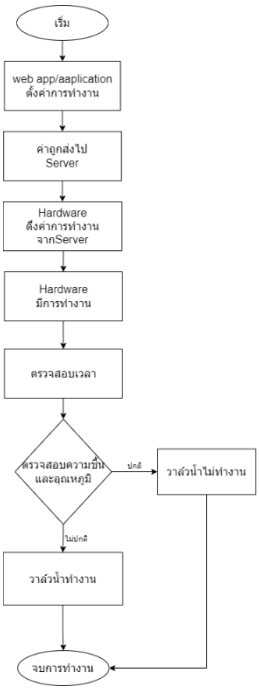
**Thingsboard**




**ภาพที่ 12** thingsboard

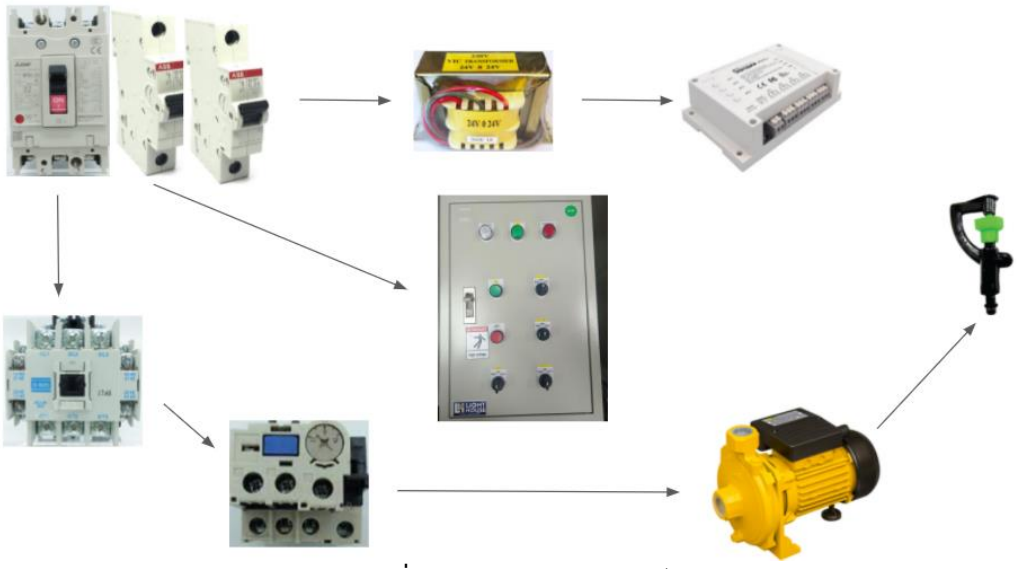
จากภาพที่ 12ใช้ Thingsboard เป็นอีกช่องทางในการนำเสนอภาพข้อมูล (data visualization) ต่างๆของอุปกรณ์แต่ละจุด ซึ่งตัว thingsboard ได้มีการติดตั้งใน server และดึงข้อมูลจากฐานข้อมูลมาโชว์ได้ และจะมีข้อมูลที่ส่งผ่านมาจากอุปกรณ์โดยตรงด้วยการสื่อสารแบบ MQTT protocol ในตัวข้อมูลที่แสดงออกมานั้นจะมีค่าต่างๆเช่นอุณหภูมิ ความชื้น , ค่าน้ำค่าไฟที่ถูกใช้ไปหรือเวลาของผู้ใช้งานที่ใช้งานอุปกรณ์ว่าแต่ละคนได้ใช้งานไปเท่าไหร่ เป็นต้น

**Hardware**




**ภาพที่ 13** device operation

เป็นการแสดงการทำงานของ Hardwareและระบบโดยที่ Application จะมีการตั้งค่าการทำงานโดยการตั้งค่านั้นจะตั้งค่าของความชื้นและอุณหภูมิ หลังจากนั้นค่าจะถูกส่งไปที่ Server แล้ว Hardware ก็จะดึงค่าใน Server มาใช้ ทีนี้อุปกรณ์ก็จะทำงานโดยถ้าค่าความชื้นหรืออุณหภูมิค่าใดค่าหนึ่งถึงจุดที่กำหนด ตัว Sonoff จะตรวจสอบค่าและทำงานทันทีตามค่าที่กำหนดไว้

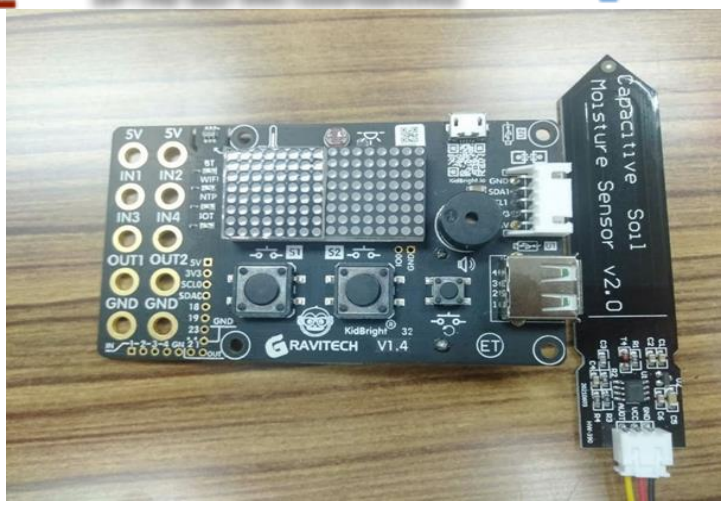


**ภาพที่ 14** Diagram ของอุปกรณ์

จากภาพที่ 14 เบรกเกอร์จะมีอยู่ 3 ลูกด้วยกันโดยลูกแรกจะเป็น main ซึ่งทำหน้าที่ควบคุมไฟทั้งหมดและส่งไฟไป magnetig contactor ซึ่งมันทำหน้าที่เปิดปิดเหมือนเป็นสวิสต์ไฟตัวหนึ่งที่สามารถรับกำลังไฟได้สูงและมี Overload ทำงานร่วมกันอยู่ตรงนั้นด้วย เบรกเกอร์ลูกที่ 2 จะทำหน้าที่ส่งไฟไปหน้าตู้ ซึ่งหน้าตู้นั้นจะมีไฟบอกสถานะการเปิดปิดปั๊มน้ำ ปุ่ม ON/OFF และปุ่มSelector และเบรกเกอร์ลูกที่ 3 จะเปิดหม้อแปลงไฟฟ้าเพื่อส่งไฟไปยัง Sonoff เพื่อให้อุปกรณ์สามารถทำงานได้

**ภาพที่ 15** hardware(a)

จากภาพที่ 15 แสดงถึงการต่ออุปกรณ์ภายในตู้ รวมไปถึงสายต่างๆ ที่จะส่งไฟไปยังหน้าตู้ได้ ซึ่งการต่อภายในตู้นี้ จะเป็นการต่อให้ปั๊มน้ำหอยโข่งสำหรับ 1 เฟสเท่านั้น




**ภาพที่ 16** hardware(b)

อุปกรณ์ที่ใช้คือ Sonoff4ch PRO R2 เข้ามาเพื่อควบคุมการเปิดปิดวาล์วน้ำ การควบคุมนั้น สามารถควบคุมได้ทั้งผ่านแอพพลิเคชั่น เว็บ และหน้าตู้ไฟ โดยหน้าตู้ไฟก็จะมีไฟแสดงสถานะ และปุ่มเปิดปิด ส่วนภายในตู้นั้นจะมีเบรกเกอร์ไว้ควบคุมไฟ Magnetic Contactor ไว้ทำหน้าที่ตัดต่อวงจรของไฟฟ้าในระบบเพื่อเปิดปิดหน้าสัมผัส ซึ่งจะทำงานควบคู่กับ Overload เพื่อช่วยตัดวงจรเพื่อจ่ายกระแสไปให้โหลด ต่อมาคือในส่วนของหม้อแปลงที่ทำหน้าที่แปลงพลังงานไฟฟ้าแรงสูงไปหาต่ำ โดยที่มันจะแปลงไฟแล้วเข้าสู้ Sonoff เพื่อให้มันทำงาน

ต่อมาคือในส่วนของการวัดอุณหภูมิและวัดความชื้นในดิน ซึ่งอุปกรณ์อุณหภูมิใช้เป็นตัว Kidbright และอุปกรณ์ความชื้นในดินใช้ Capacitive Soil Moisture Sensor ในการวัดค่าความชื้น วิธีการใช้นั้นคือทำการตั้งค่าให้มัน เมื่อค่าถึงตามที่ตั้งไว้(ดินแห้งเกินไป) จะให้ค่าส่งไปยัง Database และให้ตัว Relay ใน Sonoff ทำการเปิดน้ำ

**ผลการวิจัย**

การทำงานทั้งในด้านระบบและอุปกรณ์ดำเนินไปได้ด้วยดี อย่างระบบที่มีการเปิดปิด, การเพิ่มอุปกรณ์เข้าระบบ, การแยกพื้นที่(location) ให้กับอุปกรณ์ต่างๆ ก็สามารถใช้งานได้เป็นปกติ และอุปกรณ์ก็สนองกับสิ่งที่แอพพลิเคชั่นสั่งออกมา ส่วนขั้นตอนที่กำลังจะทำทำต่อจากนี้คือเรื่องของการให้อุปกรณ์รดน้ำทำงานตามการตั้งค่าเวลาและความชื้นที่กำหนดไว้ และการนำค่าที่ได้จริงมานำเสนอในลักษณะกราฟ

| Data | | | | |
|---|---|---|---|---|
| NO | Deice_id | RH | Temp | logtime |
| 1 | 485104 | 69% | NULL | 2022-03-23 15:05:00 |
| 2 | 212618 | NULL | 25° | 2022-03-23 15:05:01 |
| 3 | 212618 | 67% | NULL | 2022-03-23, 15:05:15 |
| 4 | 485104 | NULL | 21° | 2022-03-23, 15:05:16 |
| 5 | 212618 | 66% | NULL | 2022-03-23, 15:05:30 |
| 6 | 485104 | NULL | 22° | 2022-03-23, 15:05:31 |

| consumption | | | |
|---|---|---|---|
| No | Location | Time | flow |
| 1 | หน้าตึก 2 | 13.00 - 13.10 | 2300 |
| 2 | ลานจอดรถ | 13.30 - 13.45 | 3600 |
| 3 | ริมรั้วหน้ามหาลัย | 14.00 - 14.15 | 3700 |
| 4 | ลานวิศวะ | 14.30 - 14.45 | 3550 |
| 5 | บนตึก 5 | 08.00 - 08.15 | 3400 |
| 6 | บนตึก 9 | 09.00 - 09.15 | 3400 |

**ภาพที่ 17** ข้อมูลอุณหภูมิความชื้น และ ข้อมูลการใช้เวลาที่เปิดกับการใช้น้ำ

นี่คือข้อมูลของค่าความชื้นในดินและอุณภูมิซึ่งจะส่งมาเก็บไว้ในฐานข้อมูลทุกๆ 15 นาทีซึ่งสามารถนำไปใช้เปิด-ปิดวาว์รดน้ำหรือตรวจสอบพืชว่าอยู่ในสภาพแวดล้อมที่เหมาะสมหรือไม่ และอีกภาพคือข้อมูลสถานที่ เวลาที่ตัวอุปกรณ์ใช้งานรวมถึง

ปริมาณน้ำที่ถูกใช้(ลิตร) ซึ่งข้อมูลเหล่านี้สามารถนำไปคำนวนน้ำที่ใช้หรือคำนวนเป็นยูนิตค่าไฟจากคุณสมบัติการจ่ายไฟของปั้มน้ำได้

สิ่งที่เป็นปัญหาสำหรับวิจัยนี้คือ การที่ Thingboard ยังไม่สามารถรวมเข้ากับเซิฟเวอร์ได้ เนื่องจาก ThingBoard มี memory usage ค่อนข้างมาก บางทีทำให้เปิดเซิฟเวอร์ได้ค่อนข้างยากและมีอาการเซิฟเวอร์ค้าง

**สรุปและอภิปรายผลการวิจัย**

ทั้งในตัวอุปกรณ์และแอพพลิเคชั่นต่างก็สามารถทำงานได้ตามที่ต้องการ แต่มีเพียงผู้พัฒนาเท่านั้นที่ทำการทดลอง แต่ไม่ได้ใช้กับผู้ใช้จริงๆ ปัญหาช่องทางการสื่อสาร(port)ถูกจำกัดจึงต้องใช้ wifi ส่วนบุคคลมาทดลองทำให้ไม่สามารถใช้งานต่อเนื่องเป็นเวลานานได้และเรื่องของ Thingboard ที่เชื่อมกับเซิฟเวอร์ยังเชื่อมต่อกันไม่ได้ จึงทำให้ยังต้องการแก้ไขในเรื่องที่กล่าวข้างต้นต่อไป

**ข้อเสนอแนะ**

1.เซิฟเวอร์อาจจะต้องมีการเพิ่มสเปคเพื่อให้ Thingboard สามารถเชื่อมต่อเข้ากับเซิฟเวอร์ได้ โดยเซิฟเวอร์ที่ได้ใช้นั้นแป็นของ open landscape ที่เป็นบริการโฮสต์เซิฟเวอร์ ในการพัฒนาระบบของ

2.เนื่องจากอินเทอร์เน็ตของมหาลัยนั้นไม่สามารถใช้กับโปรเจคนี้ได้เนื่องจากติดปัญหาเรื่องการปิดกั้น port และการ authentication จึงอยากได้สัญญาณอินเทอร์เน็ตเฉพาะของระบบ IOT ที่ควบคุมได้ทั้งมหาลัย ในการทดลองงาน

**เอกสารอ้างอิง**

Cybertice. (2021). *Sonoff 4CH Pro R2 สมาร์ทสวิตช์ ควบคุมสั่งเปิดปิด 4 ช่อง ผ่านมือถือ ได้ทุกที่ทั่วโลก.* สืบค้น 2 กุมภาพันธ์ 2022. จาก www.cybertice.com/product/2937/sonoff-4ch-pro-r2

Thanapon Tapala. (2021). *[Thingsboard] ควบคุมอุปกรณ์ผ่าน REST API*. สืบค้น 20 กุมภาพันธ์ 2022. จาก https://thanapon.info/thingsboard-rest-api/

ทวีป ตรีหะจินดารัตน์, ทศพร ปั้นจาด และปวรัชฎ์ คชรินทร์ . (2016). อินเตอร์เน็ตกับทุกสิ่งของสวนอัจฉริยะ. สืบค้น 20 มกราคม 2022.จาก http://ir.swu.ac.th/jspui/bitstream/123456789/5082/2/PRO2392.pdf?fbclid=IwAR3vjAjduz-nonXnHMoJBQvjjtAlzWBIWpmg3Ccpglk4FpA-91mf77a7Vd0

สุไฮลา ยูโซะ, วิชิต เรืองแป้น, นฤมล ทองมาก, จริยาภรณ์ มาสวัสดิ์, ปิยะรักษ์ ประดับเพชรรัตน์, สะอุดี มะประสิทธิ์, จุฑามาศ แก้วมณี, ชูชาน มะเซ็ง. (2017). *น้ำในดินและความชื้นในดิน.* สืบค้น10 กุมภาพันธ์ 2022.จาก 21 การประชุมวิชาการะดับชาติ ครั้งที่ 6 มหาวิทยาลัยฟาฏอนี.pdf (yru.ac.th)

ESPHOME. (2020). *Sonoff 4 CH Pro R2*. Retrieved 3 February 2022. From https://www.esphome-devices.com/devices/Sonoff-4CH-Pro-R2

Reactnative.dev .(2022). *Introduction React Native* . Retrieved 10 January 2022. Form https://reactnative.dev/docs/getting-started

_____.(2022). *Setting up the development environment*. Retrieved 10 January 2022. form https://reactnative.dev/docs/environment-setup