การเปรียบเทียบการพัฒนาโปรแกรมประยุกต์ IOT สำหรับโทรศัพท์มือถือ ระหว่าง IOT Cloud Platform Application และ End-to-End Application กรณีศึกษาเครื่องให้อาหารปลา

โสภณวิชญ์ เขียวคำจีน

สารนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม
วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์
มหาวิทยาลัยธุรกิจบัณฑิตย์
พ.ศ. 2562

Comparison of Mobile Application Development between IOT Cloud Platform Application and End-to-End Application Fish Feedder

Soponwit Khewkumjeen

A Thematic Paper Submitted in Partial Fulfillment of the Requirements

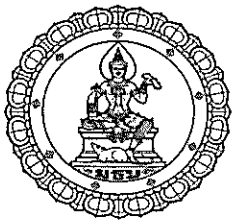
for the Degree of Master of Engineering

Department of Computer and Telecommunication Engineering

College of Innovative Technology And Engineering

Dhurakij Pundit University

2019



ใบรับรองสารนิพนธ์

วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์

มหาวิทยาลัยธุรกิจบัณฑิตย์

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

หัวข้อสารนิพนธ์ การเปรียบเทียบการพัฒนาโปรแกรมประยุกต์ IoT สำหรับโทรศัพท์มือถือ ระหว่าง IoT Cloud Platform และ End-to-end กรณีศึกษา เครื่องให้อาหารปลา

เสนอโดย นายโสภณวิชญ์ เขียวคำจีน

สาขาวิชา วิศวกรรมคอมพิวเตอร์และโทรคมนาคม

อาจารย์ที่ปรึกษาสารนิพนธ์ อาจารย์ ดร.ชัยพร เขมะภาตะพันธ์

ได้พิจารณาเห็นชอบโดยคณะกรรมการสอบสารนิพนธ์แล้ว

.....ประธานกรรมการ

(อาจารย์ ดร.ประศาสน์ จันทราทิพย์)

.....กรรมการและอาจารย์ที่ปรึกษาสารนิพนธ์

(อาจารย์ ดร.ชัยพร เขมะภาตะพันธ์)

.....กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.มัชฌิกา อ่องแตง)

วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์รับรองแล้ว

.....คณบดีวิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์

(ผู้ช่วยศาสตราจารย์ ดร.ณรงค์เดช กีรติพรานนท์)

วันที่ ...18... เดือน ...สิงหาคม... พ.ศ. ...2564...

หัวข้อสารนิพนธ์ การเปรียบเทียบการพัฒนาโปรแกรมประยุกต์IOTสำหรับโทรศัพท์มือถือระหว่าง IOT Cloud Platfrom Application และ End-to-End Application
ชื่อผู้เขียน โสภณวิชญ์ เขียวคำจีน
อาจารย์ที่ปรึกษา อาจารย์ ดร.ชัยพร เขมะภาตะพันธ์
สาขาวิชา วิศวกรรมคอมพิวเตอร์และโทรคมนาคม
ปีการศึกษา 2561

บทคัดย่อ

วัตถุประสงค์ของสารนิพนธ์ฉบับนี้เป็นการศึกษาออกแบบและเปรียบเทียบการพัฒนาโปรแกรมประยุกต์ IoT สำหรับโทรศัพท์มือถือระหว่าง IoT Cloud Platfrom Application และ End-to-End Application โดยกรณีศึกษาผ่านเครื่องให้อาหารปลา ปัจจุบันแอพพลิเคชั่นโปรแกรมประยุกต์ IoT สาหรับโทรศัพท์มือถือมีการพัฒนาและเกิดขึ้นจำนวนมากสามารถพัฒนาได้ทั้งรูปแบบ IOT Cloud Platfrom Application และ End-to-End Application จึงจำเป็นต้องศึกษาและทดสอบว่าแอพพลิเคชั่นโปรแกรมประยุกต์รูปแบบไหนมีคุณสมบัติและประสิทธิภาพเหมาะสมต่อการนำไปใช้งานได้

โดยงานวิจัยส่วนที่เป็น IoT Cloud Platfrom Application จะใช้แอพพลิเคชั่น Blynk เป็นตัวศึกษาออกแบบและทดสอบและในส่วน End-to-End Application จะใช้แอพพลิเคชั่น Inventor 2 เป็นตัวศึกษาออกแบบและทดสอบ โดยภาพรวมของงานวิจัยทั้งหมดจะเป็นการออกแบบและศึกษาและเปรียบเทียบประสิทธิภาพของโปรแกรมประยุกต์ IOT สำหรับโทรศัพท์มือถือระหว่าง IOTCloud Platfrom Application และ End-to-End Application โดยทดสอบประสิทธิภาพด้าน User Interface ประสิทธิภาพด้านเวลาและประสิทธิภาพด้านค่าใช้จ่าย โดยจะทดสอบผ่านเครื่องให้อาหารปลา

จากผลการทดสอบพบว่าการพัฒนาแอพพลิเคชั่นบน IoT Cloud Platfrom Application ให้ประสิทธิภาพการทำงานที่ช้ากว่าเล็กน้อย แต่มีขั้นตอนการพัฒนาและการนำไปใช้งานที่สะดวกกว่าเมื่อเทียบกับ End-to-End Application

Thematic Paper Title	Comparison of Mobile Application Development between IOT Cloud Platform Application and End-to-End Application Fish Feedder
Author	Soponwit Khewkumjeen
Thematic Advisor	Dr. Chaiyaporn Khemapatapan
Department	Computer and Telecommunication Engineering
Academic Year	2018

ABSTRACT

The objective of this thematic research is to design, implement and compare the development of mobile-based IoT application between IoT Cloud Platfrom Application and End-to-End Application in case of fish feeder. Currently, mobile-based IoT application can be developed by using both IoT Cloud Platfrom Application and End-to-End Application. Thus, the study of which one is suitable and provides the acceptable performance to develop a mobile-based IoT application .

Based on IoT Cloud Platfrom Application, Blynk is chosen for use in design and implement. Inventor2 is chosen for use in design and implement for End-to-End Application. The testing results include the performances, user interface design and cost of implementation.

From the studied results, it can be found that IoT Cloud Platfrom Application provides the worse performance than End-to-End Application. However, IoT Cloud Platfrom Application has more familiar in design, implementation and applying for use in comparison with End-to-End Application.

กิตติกรรมประกาศ

สารนิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้อย่างสมบูรณ์โดยได้รับความอนุเคราะห์และความกรุณาอย่างดียิ่งจาก อาจารย์ ดร. ชัยพร เขมะภาตะพันธ์ อาจารย์ที่ปรึกษาสารนิพนธ์ ที่ได้กรุณาสละเวลาอันมีค่าในการให้ความรู้ คำปรึกษา ตรวจสอบ แก้ไขข้อบกพร่อง รวมถึงชี้แนะและช่วยเหลือในสิ่งต่างๆ แก่ผู้จัดทำด้วยดีมาโดยตลอด จนทำให้สารนิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ผู้จัดทำมีความซาบซึ้งอย่างยิ่งต่อความเมตตากรุณาของท่านและขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้ด้วย

ขอกราบขอบพระคุณ อาจารย์ ดร.ประศาสน์ จันทราทิพย์ และ ผู้ช่วยศาสตราจารย์ ดร. มัชฌิกา อ่องแตง ที่คอยช่วยเหลือ และดูแลรวมทั้งการให้คำแนะนำต่างๆอันเป็นประโยชน์ต่อสารนิพนธ์ฉบับนี้

สุดท้ายนี้ ผู้จัดทำขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และสมาชิกในครอบครัวทุกท่าน สำหรับการดูแล อบรม และคอยเป็นกำลังใจอันสำคัญอย่างยิ่งสำหรับผู้จัดทำ และขอบคุณเพื่อนๆ ทุกคนที่สละเวลาอันมีค่าในการแนะนำ และให้ความช่วยเหลือเรื่องต่างๆ จนกระทั่งสารนิพนธ์ฉบับนี้เสร็จสมบรูณ์ หากมีข้อบกพร่องหรือข้อความผิดพลาดประการใดในสารนิพนธ์ฉบับนี้ ซึ่งเกิดจากการผิดพลาดของผู้จัดทำทั้งหมดจึงขออภัยมา ณ ที่นี้ด้วย

โสภณวิชญ์ เขียวคำจีน

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ฆ
บทคัดย่อภาษาอังกฤษ..... ง
กิตติกรรมประกาศ..... จ
สารบัญตาราง..... ฌ
สารบัญภาพ..... ญ
บทที่
1 บทนำ 1
1.1 ที่มาและความสำคัญของงานวิจัย 1
1.2 วัตถุประสงค์ของงานวิจัย 1
1.3 ขอบเขตของงานวิจัย 2
1.4 ประโยชน์ที่คาดว่าจะได้รับ 2
1.5 วิธีการทดสอบ 3
1.6 อุปกรณ์และเครื่องมือที่ใช้ 3
1.7 องค์ความรู้ใหม่ 4
1.8 โครงสร้างของรายงาน 4
2 แนวคิดทฤษฎีและงานวิจัยที่เกี่ยวข้อง 5
2.1 องค์ประกอบสำคัญในการเชื่อมต่อ Internet of Things เข้ากับ Cloud 5
2.2 ระบบ E2EE หรือ End-to-End Encryption ระบบเข้ารหัสตั้งแต่ต้นทางถึงปลายทาง 7
2.3 แอพพลิเคชั่น Blynk 8
2.4 แอพพลิเคชั่น Inventor 2 10
2.5 แอพลิเคชั่น NETPIE Freeboard 13
2.6 แอพลิเคชั่น Node-RED 15
2.7 Arduino IDE 16
2.8 Internet of Things (IoT) 17
2.9 NTP (Network Time Protocol) 18

สารบัญ (ต่อ)

บทที่		หน้า
	2.10 UDP (User Datagram Protocol)	20
	2.11 TCP/IP (Transmission Control Protocol/ Internet Protocol)	21
	2.12 MQTT Server Potocal	22
	2.13 งานวิจัยที่เกี่ยวข้อง	23
3	วิธีดำเนินการวิจัย	26
	3.1 แนวทางการวิจัยและพัฒนา	26
	3.2 แผนการดำเนินงาน	26
	3.3 ภาพรวมของระบบ	28
	3.4 การออกแบบแผนผังวงจรการทำงาน	29
	3.5 การออกแบบการทำงานของระบบเครื่องให้อาหารปลา	30
	3.6 การออกแบบระบบการให้อาหารปลาตามช่วงเวลา	31
	3.7 การออกแบบระบบให้อาหารปลาอัตโนมัติ	32
	3.8 การออกแบบระบบการแจ้งเตือน	33
	3.9 การออกแบบแอพพลิเคชั่น Blynk	34
	3.10 การออกแบบระบบรับส่งข้อมูลแอพพลิเคชั่น Blynk	34
	3.11 การออกแบบ User Interface แอพพลิเคชั่น Blynk	36
	3.12 การออกแบบในส่วนกำหนดค่าตัวแปรแอพพลิเคชั่น Blynk	37
	3.13 การออกแบบแอพพลิเคชั่น Inventor 2	38
	3.14 การออกแบบระบบรับส่งข้อมูล Application Invertor 2	39
	3.15 การออกแบบ User Interface แอพพลิเคชั่น Inventor 2	41
	3.16 การออกแบบใส่วนการเขียนกำหนดค่าตัวแปรแอพพลิเคชั่น Inventor 2	42
4	ผลการทดลอบ	45
	4.1 ผลการทดสอบการการเปรียบเทียบประสิทธิภาพการออกแบบ User Interface และ ฟีเจอร์และฟังชั่นเงื่อนไขต่างๆของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor 2	45

สารบัญ (ต่อ)

บทที่ **หน้า**

4.2 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายในการออกแบบแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2 47

4.3 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านเวลาในการใช้งานของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 47

4.4 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านการออกแบบการรับส่งข้อมูลของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น 48

4.5 ผลการทดสอบประสิทธิภาพของอุปกรณ์ Hardware ในการควบคุมการทำงาน 49

4.6 ผลการทดสอบประสิทธิภาพการทำงานโดยรวมของทั้งระบบ 49

5 สรุปผลการวิจัยและข้อเสนอแนะ 50

5.1 ผลการทดลองภาพรวมเปรียบเทียบประสิทธิภาพการทำงาน 50

5.2 การบรรลุวัตถุประสงค์ 50

5.3 องค์ความรู้สำคัญ 51

5.4 ข้อเสนอแนะแนวทางการพัฒนาต่อ 51

5.5 สรุป 52

บรรณานุกรม..... 53

ประวัติผู้เขียน..... 82

สารบัญตาราง

ตารางที่ **หน้า**

3.1 แผนการดำเนินงาน 27
3.2 ตารางแสดงค่าตัวแปรและฟังก์ชั่นการทำงานของตัวตัวแปรต่างๆของ Blynk 37
3.3 ตารางแสดงค่าตัวแปรและฟังชั่นการทำงานของของแอพพลิเคชั่น Inventor 2 43
4.1 ตารางการเปรียบเทียบประสิทธิภาพการออกแบบหน้า User Interface และฟีเจอร์และฟังชั่นเงื่อนไขต่างๆ ของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 46
4.2 ตารางการเปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายในการออกแบบและการนำไปใช้งานแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 47
4.3 ตารางกราฟเปรียบเทียบประสิทธิภาพด้านเวลาในการใช้งานของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 48
4.4 ตารางการการเปรียบเทียบประสิทธิภาพด้านการออกแบบการรับส่งข้อมูลของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 49

สารบัญภาพ

ภาพที่ **หน้า**

2.1 ภาพตัวอย่างการเชื่อมต่อ Internet of Things เข้ากับ Cloud 6

2.2 ภาพตัวอย่าง End-to-End Encryption 8

2.3 ภาพตัวอย่างการทำงานระบบของโปรแกรมแอพพลิเคชั่น Blynk 9

2.4 ภาพตัวอย่าง แอพพลิเคชั่นที่สร้างขึ้นด้วยโปรแกรม App Inventor 2 ส่วนออกแบบ ส่วนการเขียนโค้ด (App Inventor Blocks Editor) 12

2.5 ภาพตัวอย่าง แอพพลิเคชั่นที่สร้างขึ้นด้วยโปรแกรม App Inventor ส่วนการเขียนโค้ด 13

2.6 ภาพตัวอย่างการทำงาน แอพพลิเคชั่น NETPIE 14

2.7 ภาพตัวอย่างแอพพลิเคชั่น NETPIE บน Freeboard 15

2.8 ภาพตัวอย่างการเขียนโปรแกรมแอพลิเคชั่น Node-RED 15

2.9 ภาพตัวอย่างการทำงานแอพพลิเคชั่น Node-RED 16

2.10 ภาพตัวอย่างโปรแกรม Arduino IDE 17

2.11 ภาพตัวอย่างโปรแกรม Arduino IDE(ต่อ) 17

2.12 ภาพตัวอย่างการเชื่อมต่อของระบบ Internet of Things (IoT) 18

2.13 ภาพตัวอย่างแสดงลำดับชั้นของการเทียบเวลาใน NTP 19

2.14 ภาพตัวอย่างแสดงลำดับชั้นเปรียบเทียบขั้นตอนการรับส่งข้อมูล TCP กับ UDP 20

2.15 ภาพตัวอย่างแสดงลำดับชั้นขั้นตอนการรับส่งข้อมูล TCP/IP 21

2.16 ภาพตัวอย่างแสดงลำดับชั้นขั้นตอนการรับส่งข้อมูล MQTT 22

3.1 ภาพรวมของระบบ 28

3.2 แสดงแผนผังวงจร 29

3.3 ผังการทำงานของระบบเครื่องให้อาหารปลา 30

3.4 แสดงผังการทำงานการให้อาหารปลาตามช่วงเวลาที่กำหนด 31

3.5 แสดงผังการทำงานการให้อาหารปลาอัตโนมัติ 32

3.6 แสดงผังการทำงานการแจ้งเตือนความขุ่นของน้ำภายในตู้ปลา 33

3.7 แสดงแอพพลิเคชั่น Blynk 34

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

3.8 ระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Blynk 35
3.9 ระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Blynk (ต่อ) 35
3.10 แสดงข้อมูลการทำงานของ User Interface บนหน้าแอพพลิเคชั่น Blynk 36
3.11 แสดงแอพพลิเคชั่น Inventor 2 39
3.12 แสดงระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Inventor 2 40
3.13 แสดงระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Inventor 2 (ต่อ) 40
3.14 แสดงข้อมูลการทำงานของ User Interface บนแอพพลิเคชั่น Inventor 2 41
3.15 แสดงข้อมูลการทำงานบนหน้า Webserver 42
5.1 แสดงเครื่องให้อาหารปลา 52

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของงานวิจัย

ในปัจจุบันอินเตอร์เน็ตสรรพสิ่ง(IoT)เริ่มมีบทบาทความสำคัญต่อชีวิตประจำวันมนุษย์อย่างมากจึงทำให้เกิดการพัฒนาและออกแบบแอพพลิเคชั่นจำนวนมากที่ใช้ในการเชื่อมต่อสั่งการผ่านอุปกรณ์(IoT)เป็นที่นิยมและแพร่หลายอย่างรวดเร็วในปัจจุบันจึงจำเป็นต้องหาแอพพลิเคชั่นที่มีประสิทธิภาพในการใช้งานและการทำงานเพื่อให้บรรลุตามวัตถุประสงค์และความต้องการของผู้ใช้งาน

ผู้วิจัยจึงมีความคิดที่จะนำแอพพลิเคชั่นสำเร็จรูประหว่างแอพพลิเคชั่น Blynk กับแอพพลิเคชั่นInventor2นำมาเปรียบเทียบกันเพราะว่าในปัจจุบันนั้นผู้ที่อยากจะมีแอพพลิเคชั่นเป็นของตัวเองนั้นส่วนใหญ่แล้วจะเป็นผู้บริหารหรือพ่อค้าแม่ค้าหรือบุคลทั่วไปซึ่งบุคลเหล่านี้จะไม่มีทักษะการเขียนโปรแกรมมากนักจึงทำให้ผู้วิจัยเลือกแอพพลิเคชั่นสำเร็จรูป2แอพพลิเคชั่นนี้เป็นแอพพลิเคชั่นที่ใช้งานง่ายและออกแบบการทำงานสำหรับผู้ไม่มีพื้นฐานด้านการเขียนโปรแกรมมากผู้วิจัยจึงนำมาเปรียบเทียบออกแบบและทดสอบประสิทธิภาพการในทำงานและการใช้งานด้านต่างๆเพื่อให้มีประสิทธิภาพและตรงตามความต้องการของผู้ออกแบบและใช้งานได้สะดวกและง่ายโดยจะทดสอบผ่านการบริหารจัดการผ่านโทรศัพท์เคลื่อนที่และจะทำการทดสอบผ่านเครื่องให้อาหารปลา

1.2 วัตถุประสงค์ของงานวิจัย

1. เพื่อการพัฒนาออกแบบเปรียบเทียบประสิทธิภาพในส่วนการใช้งานของUser interface และ ฟีเจอร์ต่าง ๆ ของแอพพลิเคชั่นBlynk กับ แอพพลิเคชั่น Inventor2โดยบริหารจัดการผ่านโทรศัพท์เคลื่อนที่

2. เพื่อการพัฒนาออกแบบเปรียบเทียบประสิทธิภาพในส่วนการรับส่งข้อมูลบนเครือข่ายอินเตอร์เน็ตของแอพพลิเคชั่นBlynk กับ แอพพลิเคชั่น Inventor 2

3. เพื่อการพัฒนาเปรียบเทียบประสิทธิภาพด้านเวลาของการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2

4. เพื่อการพัฒนาเปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายของการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2

5. เพื่อการพัฒนาออกแบบในส่วนแอพพลิเคชั่นในการทดสอบการใช้งานผ่านเครื่องให้อาหารปลา

1.3 ขอบเขตของงานวิจัย

1. เปรียบเทียบประสิทธิภาพในส่วนการใช้งานของUser Interface และฟีเจอร์ต่างๆ ของแอพพลิเคชั่นBlynk กับ แอพพลิเคชั่น Inventor 2 โดยบริหารจัดการผ่านโทรศัพท์เคลื่อนที่

2. เปรียบเทียบประสิทธิภาพในส่วนการรับส่งข้อมูลบนเครือข่ายอินเตอร์เน็ตของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor 2

3. เปรียบเทียบประสิทธิภาพเวลาของการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2

4. เปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายของการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2

5. เปรียบเทียบและทดสอบระบบการทำงานโดยรวมของระบบ

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. รู้ถึงข้อเปรียบเทียบประสิทธิภาพในส่วนการใช้งานของ Userinterface และฟีเจอร์ต่างๆ ของแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2โดยบริหารจัดการผ่านโทรศัพท์เคลื่อนที่

2. รู้ถึงวิธีการประสิทธิภาพในส่วนการรับส่งข้อมูลบนเครือข่ายอินเตอร์เน็ตของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor 2

3. รู้ถึงประสิทธิภาพเวลาของการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2

4. รู้ถึงวิธีการประสิทธิภาพด้านค่าใช้จ่ายของการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น

1.5 วิธีการทดสอบ

1. ทดสอบประสิทธิภาพในประสิทธิภาพในส่วนการใช้งานของ User interface ของแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2 โดยบริหารจัดการผ่านโทรศัพท์เคลื่อนที่
2. ทดสอบประสิทธิภาพในส่วนการรับส่งข้อมูลบนเครือข่ายอินเตอร์เน็ตของแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2
3. ทดสอบประสิทธิภาพเวลาของการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2
4. ทดสอบการใช้งานแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor 2 ผ่านเครื่องให้อาหารปลา

1.6 อุปกรณ์และเครื่องมือที่ใช้

1.6.1 ในส่วนของซอฟต์แวร์(Software)

1. Arduino IDE 1.8.7
2. Application Blynk
3. Application Inventor 2
4. Microsoft Windows 7

1.6.2 ในส่วนของฮาร์ดแวร์(Hardware)

1. Motor Servo
2. Arduino Node MCU WeMosD1 ESP8266
3. Adapter 12V
4. Real Time Clock (RTC)
5. LCD16x2a
6. Analog turbidity sensor
7. Switch On-Off
8. Switch Set-Up-Down
9. LM2596 module
10. Food pan
11. Connector
12. Control box

1.7 องค์ความรู้ใหม่

การนำเสนอแนวทางในการเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานแอพพลิเคชั่นสำเร็จรูปกับอุปกรณ์(IoT)สามารถนำไปพัฒนาและต่อยอดประยุกต์ใช้เทคโนโลยีของอินเตอร์เน็ตสรรพสิ่ง (Internet of Things) ทั้งรองรับกับยุค THAILAND 4.0 เป็นยุคแห่งนวัตกรรมแห่งการเรียนรู้และเข้าใจในเทคโนโลยีสมัยใหม่ที่กำลังเติบโตอย่างรวดเร็ว โดยการทำงานบนระบบเครือข่ายเป็นหลัก อีกทั้งหลักการทำงานของแอพพลิเคชั่น Blynk กับแอพพลิเคชั่นInventor2 ในแบบต่างๆ ที่ใช้ในการให้บริการต่างๆ และให้บริการในปัจจุบัน

1.8 โครงสร้างของรายงาน

บทที่ 2 กล่าวถึงทฤษฎีที่เกี่ยวข้องที่จำเป็น ต่องานวิจัย เป็นการนำเสนอแอพพลิเคชั่นสำเร็จรูปแอพพลิเคชั่น Blynk กับแอพพลิเคชั่นInventor2 และเทคโนโลยี ของ อินเตอร์เน็ตสรรพสิ่ง (Internet of Things) ซึ่งเป็นองค์ประกอบในแต่ละส่วนที่จำเป็น และการนำมาประยุกต์ใช้งานด้านต่างๆต่อไป

บทที่ 3 รายละเอียดของแนวทางการดำเนินการ, การออกแบบแอพพลิเคชั่นและองค์ประกอบต่างๆ ที่ใช้ภายในระบบ รวมถึงขั้นตอนและกระบวนการของการทำงานภายในระบบ โดยมีการสร้างเงื่อนไขของในการทำงานเพื่อควบคุมอุปกรณ์ที่เกิดขึ้นภายในระบบ

บทที่ 4 ผลลัพธ์การทำงานจากการทดลองระบบในแต่ละส่วนของระบบและความสามารถในการทำงานของระบบที่เกิดขึ้นจากการทดลองใช้งานจริง

บทที่ 5 สรุปงานวิจัยและข้อเสนอแนะ

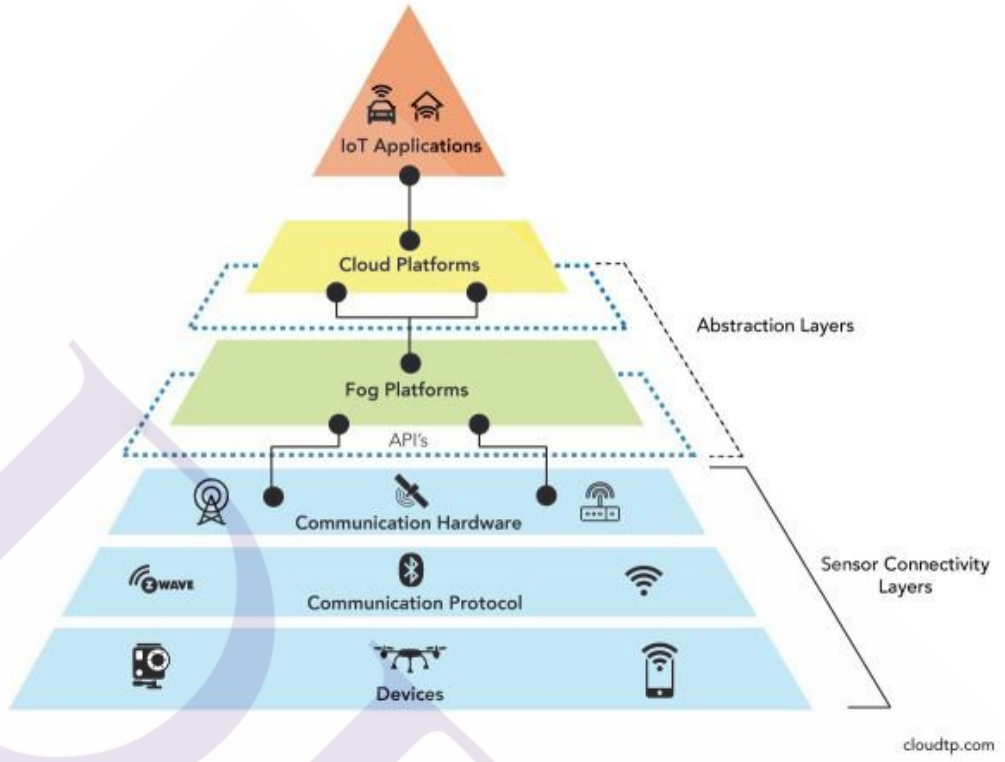
บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ผู้วิจัยมีแนวคิดการออกแบบและทดสอบการเปรียบเทียบประสิทธิภาพของแอพพลิเคชั่นสำเร็จรูประหว่างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 เพื่อที่จะให้ทราบถึงข้อแตกต่างและประสิทธิภาพการทำงานและการใช้งานในด้านด่างๆได้อย่างเหมาะสมตามความต้องการของผู้ใช้งานโดยผู้วิจัยจะทดสอบผ่านเครื่องให้อาหารปลา นอกจากนี้ต้องมีความรู้เกี่ยวกับการเขียนโปรแกรมภาษาซีและภาษาโรโก้ เพื่อควบคุมการสั่งการทำงานและระบบการรับส่งข้อมูลระหว่างตัวแอพพลิเคชั่นกับระบบServerและประกอบกับมีความรู้พื้นฐานด้านอิเล็กทรอนิกส์ในการต่อวงจร

งานวิจัยนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องของการพัฒนาและออกแบบแอพพลิเคชั่นสำเร็จรูประหว่างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 ซึ่งประกอบด้วยทฤษฎีต่างๆ คือ แอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 และการนำมาประยุกต์ใช้ของระบบเทคโนโลยีของอินเตอร์เน็ตสรรพสิ่ง (Internet of Things) โดยตัวแอพพลิเคชั่นและนำมาทดสอบใช้งานกับเครื่องให้อาหารปลา

2.1 องค์ประกอบสำคัญในการเชื่อมต่อ Internet of Things เข้ากับ Cloud

การมาของ Internet of Things (IoT) นี้เป็นหนึ่งในสิ่งที่ทำให้วงการ IT มีการเปลี่ยนแปลงมากที่สุดอย่างหนึ่งในปัจจุบัน ไม่ว่าจะเป็นระบบเครือข่าย, Sensor, อุปกรณ์ประมวลผล, ซอฟต์แวร์จัดเก็บข้อมูล, ซอฟต์แวร์วิเคราะห์ข้อมูล หรือแต่ระบบ Cloud เองก็ตาม และนอกจากการเปลี่ยนแปลงแล้ว Layerใหม่ที่เกิดขึ้นมาในระบบ Ecosystem นี้อย่าง Fog Platform ก็เป็นอีกหนึ่งสิ่งที่น่าสนใจด้าน Internet of Things



ภาพที่ 2.1 ภาพตัวอย่างการเชื่อมต่อ Internet of Things เข้ากับ Cloud

Fog Platform: PaaS สำหรับ Internet of Things

ในการพัฒนา Internet of Things Application ใดๆ นั้น ความซับซ้อนที่จะเพิ่มขึ้นมาจากแต่ก่อนในอดีตก็คือชั้นของการรับข้อมูลจาก Sensor ต่างๆ ที่มีความหลากหลายมาประมวลผล ซึ่งการพัฒนาระบบให้สามารถรองรับ Behavior ต่างๆ ของ Sensor ที่มีความแตกต่างกันทั้งในเชิงของผู้ผลิต, Protocol, ข้อมูลที่รับส่ง และวิธีการเข้ารหัสนี้ก็ถือเป็นงานที่ยากลำบากเป็นอย่างมากอย่างแน่นอน ซึ่งในปัจจุบันนี้ที่เรายังไม่เห็นความซับซ้อนตรงนี้มากนัก เพราะ IoT ตอนนี้ยังอยู่ในระยะเริ่มต้นที่ Sensor ส่วนใหญ่ในแต่ละ Application ยังเป็น Sensor กลุ่มเดียวกันอยู่ ซึ่งในอนาคตภาพเหล่านี้ก็จะค่อยๆ หายไป

นอกจากนี้ การนำ IoT ไปประยุกต์ใช้ในแต่ละกลุ่มอุตสาหกรรมนั้นก็มีความแตกต่างกันในเชิงกฎหมายและข้อบังคับหรือ Compliance ที่องค์กรต้องเผชิญ ระดับความปลอดภัยที่ต้องทำการควบคุม หรือความสำคัญของข้อมูลที่ต้องจัดเก็บและวิเคราะห์นั้นก็จะแตกต่างกันออกไป ทำให้การพัฒนา IoT Application แต่ละครั้งนั้นไม่ใช่เรื่องง่ายและนำมา Reuse ได้ค่อนข้างยาก

Fog Platform คือระบบที่เกิดขึ้นมาเพื่อแก้ไขสองปัญหาหลักๆ เหล่านี้ โดย Fog Platform นี้จะอยู่กึ่งกลางระหว่างชั้นของ Cloud Platform ที่ทำหน้าที่จัดเก็บข้อมูล, ประมวลผล และแสดงผลข้อมูลให้กับผู้ใช้งาน กับชั้นของ Sensor ที่ส่งข้อมูลต่างๆ ผ่านมายังระบบเครือข่ายนั่นเอง

เรียกได้ว่า Fog Platform นี้คือ Platform-as-a-Service (Paas) สำหรับการทำ Internet of Things โดยเฉพาะ เพื่อทำให้การเชื่อมต่อและการควบคุม IoT Sensor สามารถทำได้ง่ายขึ้น รวมถึงการประมวลผลข้อมูลขั้นต้นเองก็จะทำได้อย่างรวดเร็วและง่ายดาย ตรงตามความต้องการของข้อกำหนดต่างๆ ในแต่ละอุตสาหกรรม ทำให้เหล่านักพัฒนาสามารถ Focus ไปกับการพัฒนา Application ที่นำข้อมูลจาก Sensor และสั่งการกลับไปยัง Sensor ต่างๆ ได้อย่างมีคุณภาพมากยิ่งขึ้นนั่นเอง

Fog Platform จะพบปัญหาแบบเดียวกับ Cloud Platform

ในปัจจุบันนี้เราเริ่มเจอกับการต้องใช้งาน Cloud Platform จากหลายๆ ผู้ให้บริการเพื่อนำมาสร้างเป็น Application เดียว และทำให้การบริหารจัดการระบบ Cloud หลายๆ ระบบนี้เป็นอีกปัญหาหนึ่ง ซึ่ง Fog Platform เองก็จะต้องพบกับปัญหาลักษณะนี้เช่นกัน และผู้ใช้งานก็ต้องเตรียมรับมือกับการบริหารจัดการหลายๆ Fog Platform นี้พร้อมๆ กันให้ได้ และ Internet of Things จะทำให้ Cloud เติบโตอย่างรวดเร็วยิ่งกว่าเดิมด้วยปริมาณของข้อมูลที่มหาศาล และการมาของ Fog Platform นี้ก็จะเป็นตัวเร่งให้ Cloud เติบโตยิ่งขึ้นไปอีกสำหรับการจัดการ IoT Device จำนวนมหาศาลที่กำลังจะเกิดขึ้นในตลาดในอนาคตนี้อย่างไรก็ดี Fog Platform นี้ก็จะกลายเป็นอีกเทคโนโลยีที่ทุกๆ องค์กรไม่สามารถหลีกเลี่ยงได้เช่นเดียวกับ Cloud เพราะการที่จะนำเทคโนโลยี IoT มาใช้งานให้ได้อย่างรวดเร็วและปลอดภัยนั้น การใช้ Fog Platform จึงถือเป็นหนทางที่จะทำให้ขีดความสามารถในการแข่งขันสูงยิ่งขึ้น ในขณะที่การพัฒนา Software แบบเดิมๆ

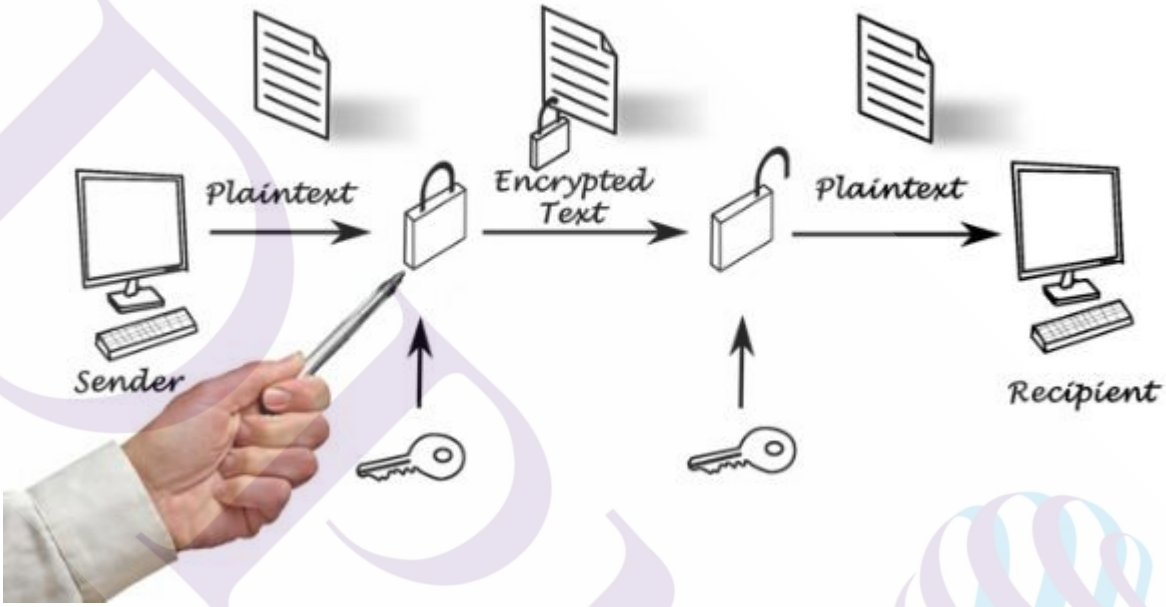
2.2 ระบบ E2EE หรือ End-to-End Encryption ระบบเข้ารหัสตั้งแต่ต้นทางถึงปลายทาง

End-to-End Encryption หรือ E2EE คือการเข้ารหัสผ่านระหว่างคู่สนทนาสองคนตั้งแต่ต้นทางถึงปลายทาง ที่มีผู้ให้บริการหลายรายใช้โฆษณากันว่าบริการส่งข้อมูลของตนมีความปลอดภัย จากการโดนดักฟัง หรือการสอดส่องข้อมูล

ระบบ E2EE หรือ End-to-End Encryption มีวิธีการทำงานอย่างไร

ยกตัวอย่าง เช่น ให้มีคู่สนทนาเป็นนาย A และนาย B เมื่อนาย A ส่งข้อความหานาย B โดยเปิดระบบ E2EE ข้อความที่ออกไปหานาย B จะถูกแปลงเป็นรหัสระหว่างทาง โดยมีถนนคือ อินเตอร์เน็ต ระหว่างที่ข้อความถูกส่งนั้น หากมีผู้ไม่ประสงค์ดีต้องการดูข้อความก็จะแสดงเป็นแค่รหัสเท่านั้น ไม่สามารถเห็นข้อความที่นาย A ส่งถึงนาย B ได้แบบตรงๆ

สรุป คือ การเข้ารหัสตั้งแต่ต้นทางถึงปลายทาง ช่วยให้เรามั่นใจได้ว่าข้อความของเราจะถูกเปลี่ยนเป็นความลับ โดยผู้ส่งดั้งเดิม และสามารถถอดรหัสได้โดยผู้รับปลายทางเท่านั้น รูปแบบอื่นๆในการเข้ารหัสอาจขึ้นอยู่กับการเข้ารหัสที่ดำเนินการโดยบุคคลภายนอก โดยทั่วไปแล้วการเข้ารหัสแบบต้นทางถึงปลายทางจัดว่าเป็นวิธีที่ปลอดภัยกว่าแบบอื่น เนื่องจากสามารถลดจำนวนของผู้ที่สามารถเข้าแทรกแซงหรือแฮ็กการเข้ารหัสได้



ภาพที่ 2.2 ภาพตัวอย่าง End-to-End Encryption

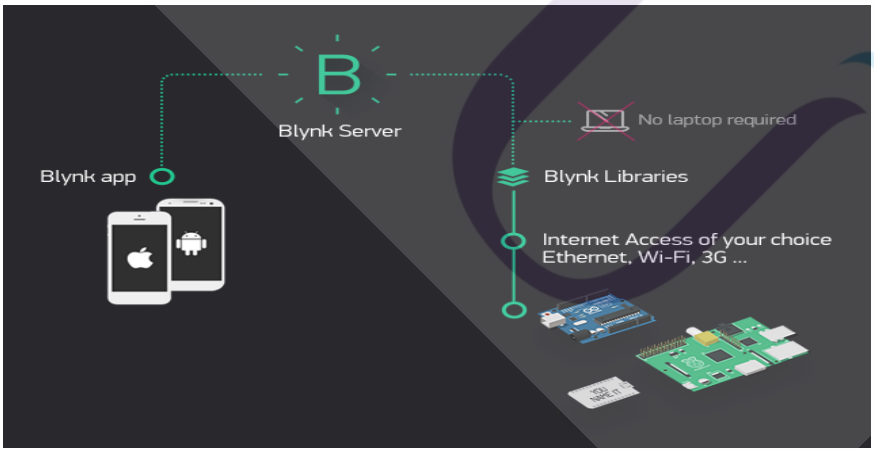
2.3 แอพพลิเคชั่น Blynk

Blynk คือ Application สำเร็จรูปสำหรับงาน IOT มีความน่าสนใจคือการเขียนโปรแกรมที่ง่าย ไม่ต้องเขียน App เองสามารถใช้งานได้อย่าง Real time สามารถเชื่อมต่อ Device ต่างๆเข้ากับ Internet ได้อย่างง่ายดาย ไม่ว่าจะเป็น Arduino, Esp8266, Esp32, Nodemcu,Rasberry pi นำมาแสดงบน Application ได้อย่างง่ายดาย แล้วที่สำคัญ Application Blynk ยัง รองรับในระบบ IOS และ Androidและในยุคสมัยก่อน การเขียนโปรแกรมเชื่อมต่อกันระหว่าง อุปกรณ์ 2 ชิ้นเข้าด้วยกันมักจะใช้งานในลักษณะของ Server >>> Client ทำให้เกิดข้อจำกัดต่างๆมากมาย ยกตัวอย่างเราต้องการเปิดปิดไฟ ผ่านหน้าเว็บ เราก็จะให้ Arduino เป็น Server และ เครื่องคอมพิวเตอร์ (Client) เป็นเครื่องลูก ข้อจำกัดที่เกิดขึ้นคือทรัพยากร เช่น CPU RAM ROM ของเราอาจจะไม่พอ มักจะเจอปัญหาเออเร่อบ่อยหรือค้างไปดื้อๆ ก็มี ทำให้การเขียนโปรแกรมเป็นไปได้ยากต้อง

ประหยัดทรัพยากรให้ได้มากที่สุดเพื่อจะให้สามารถทำงานได้ และการเช็ค Network เป็นไปได้ยาก ส่วนใหญ่มักจะใช้ในวง Lan หรือถ้าต้องการควบคุมผ่าน Wan จะต้อง Forward Set ระบบ Network ที่ยุ่งยากต่อมาเป็นยุคของ Cloud เกิดขึ้น บวกกับมี Chip WiFi ราคาถูก Esp8266 ถูกผลิตขึ้นมา แต่ด้วยข้อจำกัดทางด้านทรัพยากร จึงมือวิธีการคิดว่า ถ้านำข้อมูลไปใส่ลงใน Server เลยแล้วให้ Device ของเราเรียกเข้าไปแก้ไข หรืออ่านข้อมูลโดยตรง ทำให้ความฉลาดของตัวอุปกรณ์ของเราไม่มีวันสิ้นสุดหมดข้อจำกัดหลายอย่าง Device กลายเป็นแค่ตัวรับ Data และส่ง Data มาแสดงเท่านั้น ทำให้ Chip Esp8266 จึงได้รับความนิยมในปัจจุบันและ[13]ในส่วนแอพพลิเคชั่น Blynk เป็น platform ที่ช่วยให้ผู้ใช้งานสามารถเชื่อมต่อบอร์ดชนิดต่าง ๆ ที่เชื่อมต่ออินเตอร์เน็ตได้แล้วสามารถควบคุมการทำงานได้ง่ายขึ้น บริการของ Blynk มี 3 ส่วนหลักๆคือ

1. Mobile App สำหรับควบคุมหรือดูข้อมูล
2. Cloud Service สำหรับเก็บข้อมูลจากอุปกรณ์
3. Library สำหรับเขียนโค้ดเชื่อมต่อกับ Cloud Service

ในส่วน Blynk server ถือว่าเป็น IoT cloud จะคอยรับการติดต่อจาก IoT Node และBLYNK SERVER ยังถูกพัฒนามาจากภาษา JAVA มันสามารถทำงานภายใต้ OS อะไรก็ได้ไม่ว่า จะ เป็น WINDOWS / MAC / LINUX และBLYNK SERVER เป็น OPEN-SORCE แบบ GNU GENERAL PUBLIC LICENSE ที่สามารถ นำไปใช้งานได้จริงและสามารถนำ ไปต่อยอดเชิงธุรกิจได้และเผยแพร่ได้กว้างขวางอีกด้วย



ภาพที่ 2.3 ภาพตัวอย่างการทำงานระบบของโปรแกรมแอพพลิเคชั่น Blynk

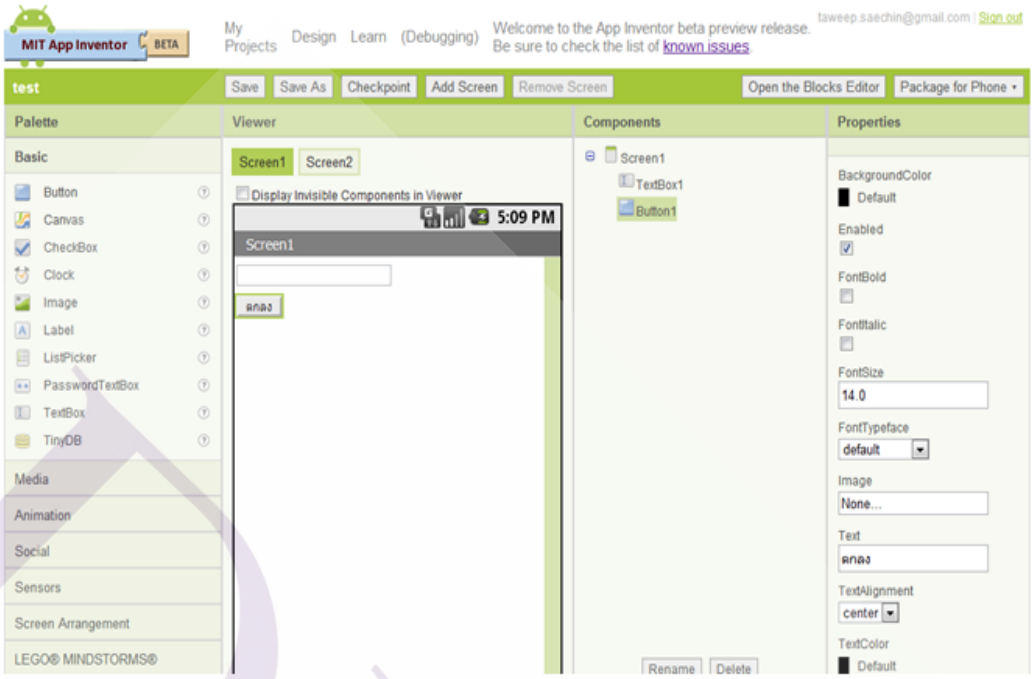
2.4 แอพพลิเคชั่น Inventor 2

โปรแกรม App Inventor พัฒนาขึ้นโดย MIT โดยโปรแกรม App Inventor เป็นเครื่องมือที่ใช้ในการเขียนโปรแกรมเพื่อพัฒนาเป็นแอพพลิเคชั่นสำหรับใช้งานบนระบบปฏิบัติการแอนดรอยด์ (Android) ซึ่งเป็นระบบปฏิบัติการที่ใช้ในอุปกรณ์สตาร์ทโฟนและแท็บเล็ต โปรแกรม App Inventor พัฒนาขึ้นโดย Professor Hal Abelson และคณะซึ่งเคยเป็นผู้พัฒนาภาษาโลโก้มาก่อน เขาพัฒนาโปรแกรม App Inventor โดยได้รับการสนับสนุนจากบริษัทกูเกิล (Google Inc.) ซึ่งมีแนวคิดในการพัฒนาอยู่ที่ว่าคนที่อยากสร้างแอพพลิเคชั่นเพื่อใช้งานบนระบบปฏิบัติการแอนดรอยด์แต่ไม่มีความรู้ ไม่มีทักษะในการเขียนโปรแกรมมาก่อนก็สามารถสร้างแอพพลิเคชั่นขึ้นได้ง่ายๆ โดยโปรแกรม App Inventor พัฒนาขึ้นบนพื้นฐานทฤษฎีการเรียนรู้ตามแนวทางคอนสตรัคชั่นนิสซึ่ม ซึ่งเน้นให้ใช้การเขียนโปรแกรมเป็นส่วนช่วยส่งเสริมให้เกิดประสิทธิภาพทางความคิดผ่านการเรียนรู้ด้วยการปฏิบัติงานและได้ถูกนำไปใช้สอนนักเรียนในระดับมัธยมศึกษาและในมหาวิทยาลัยหลายแห่งในสหรัฐอเมริกา (Wikipedia, 2012) ด้วยข้อดีของโปรแกรม App Inventor ที่ทำให้ผู้ใช้สามารถพัฒนาแอพพลิเคชั่นได้ง่ายและสนุกเหมือนการต่อจิ๊กซอว์ (Jigsaw puzzle) หรือการต่อตัวต่อเลโก้ (Lego bricks) App Inventor จึงได้รับความสนใจเป็นอย่างมากสำหรับนักการศึกษาและนักพัฒนาแอพพลิเคชั่นที่ยังไม่มีประสบการณ์ในการพัฒนาแอพพลิเคชั่นเพื่อใช้งานบนระบบปฏิบัติการแอนดรอยด์ สอดคล้องกับงานวิจัยของ Pavel Smutty (2011, p.358) ได้กล่าวถึงข้อดีของโปรแกรม App Inventor สำหรับใช้ในการเรียนการสอนว่าการเขียนโปรแกรมด้วย App Inventor ที่มีลักษณะเป็นการต่อบล็อกนั้นง่ายต่อการจำรูปแบบของคำสั่งต่างจากการเขียนโปรแกรมด้วยคำสั่งที่เป็นตัวอักษรซึ่งนักเรียนส่วนใหญ่มักจะพิมพ์คำสั่งผิดทำให้เกิดข้อผิดพลาดในการเขียนโปรแกรมจำนวนมาก และนักเรียนยังสับสนกับข้อความแสดงข้อผิดพลาด (error messages) ที่แสดงออกมา ด้วยกลุ่มของคอมโพเนนท์และ ฟังก์ชันที่โปรแกรม App Inventor จัดเตรียมไว้ให้ ช่วยให้เมื่อเริ่มเขียนโปรแกรมผู้พัฒนาเพียงแค่หาบล็อกที่ต้องการแล้วคลิกลากบล็อกนั้นไปวางไว้ในโปรแกรม ซึ่งไม่จำเป็นต้องจำว่าจะเริ่มต้นเขียนด้วยคำสั่งอย่างไร โครงสร้างของคำสั่งเป็นแบบไหน ด้วยความเป็นบล็อกที่นำมาต่อกันเหมือนจิ๊กซอว์ช่วยลดข้อผิดพลาดในการเขียนโปรแกรมเพราะหากเป็นบล็อกที่เลือกไม่ถูกต้องก็จะไม่สามารถนำมาต่อเข้าด้วยกันได้ และด้วยคอมโพเนนท์ที่มีความสามารถสูง (High-level components) เช่น คอมโพเนนท์ที่เกี่ยวกับระบบ GPS (Global Positioning System) ระบบควบคุมหุ่นยนต์เลโก้ (Lego Mindstorms NXT Robot) ระบบอ่านข้อความเป็นเสียง (text-to-speech) และระบบรู้จำเสียง (speech recognition) เป็นต้น ช่วยลดเวลาในการพัฒนาแอพพลิเคชั่นเพราะผู้พัฒนาสามารถนำคอมโพเนนท์นั้นมาใช้งานได้เลยทันทีโดยไม่ต้องเสียเวลาเขียนโปรแกรมนานและแอพพลิเคชั่น

Inventor 2 ยังช่วยให้สามารถพัฒนาแอพพลิเคชั่นสำหรับโทรศัพท์ระบบปฏิบัติการ Android ซึ่งทำผ่านการใช้เว็บเบราเซอร์และทดสอบบนโทรศัพท์ที่เชื่อมต่ออยู่กับคอมพิวเตอร์หรือทดสอบบนโทรศัพท์จำลองบนเครื่องคอมพิวเตอร์ โปรเจคที่สร้างทั้งหมดจะถูกจัดเก็บไว้บนเซิร์ฟเวอร์ App Inventor ซึ่งช่วยให้สามารถพัฒนางานต่อที่เครื่องคอมพิวเตอร์เครื่องใดก็ได้ เพียงแค่ได้มีการเชื่อมต่อกับระบบอินเทอร์เน็ตไว้เท่านั้น กาสร้างแอพพลิเคชั่นจะแบ่งการทำงานออกเป็นสองส่วน คือ ส่วนออกแบบ (App Inventor Designer) ที่จะให้เราเลือกคอมโพเนนท์ที่ต้องการสำหรับที่จะให้สร้างแอพพลิเคชั่น ส่วนที่สองเป็นส่วนการเขียนโค้ด (App Inventor Blocks Editor) ที่ให้เราเขียนโค้ดด้วยการต่อบล๊อกต่างๆ เข้าด้วยกันเป็นคำสั่ง ซึ่งจะเป็นการกำหนดพฤติกรรมหรือเหตุการณ์ที่เกิดขึ้นกับคอมโพเนนท์ การเขียนโปรแกรมจะเสมือนการต่อชิ้นส่วนตัวต่อจิ๊กซอว์เข้าด้วยกัน ในแต่ละขั้นตอนการสร้างจะสามารถทำการทดสอบได้ทุกขณะ และเมื่อสร้างเสร็จสมบูรณ์แล้วจะสามารถแพ็คเก็ตแอพพลิเคชั่นเพื่อนำไปใช้งานบนโทรศัพท์ระบบปฏิบัติการ Android เครื่องใดก็ได้ หรือหากไม่มีโทรศัพท์ระบบปฏิบัติการ Android ก็สามารถที่จะทดสอบได้บนโทรศัพท์จำลองที่ทำงานอยู่บนคอมพิวเตอร์ซึ่งจะมีลักษณะการทำงานเหมือนโทรศัพท์จริงทุกประการ สภาพแวดล้อมในการพัฒนาด้วยโปรแกรม App Inventor นั้น สนับสนุนระบบปฏิบัติการที่หลากหลาย ไม่ว่าจะเป็นระบบปฏิบัติการ Mac OS X, GNU / Linux และระบบปฏิบัติการ Windows และแอพพลิเคชั่นที่สร้างขึ้นนั้นสามารถติดตั้งและทำงานได้บนโทรศัพท์ระบบปฏิบัติการ Android หลากหลายรุ่น

ส่วนออกแบบ (App Inventor Designer)

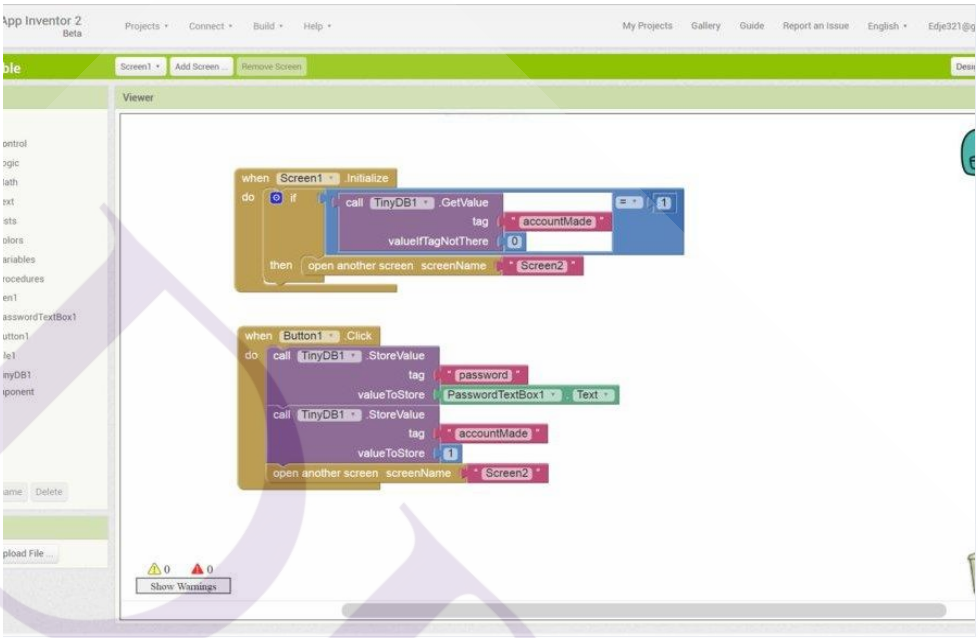
ในขั้นตอนแรกของการสร้างแอพพลิเคชั่นด้วย App Inventor เริ่มจากการเลือกคอมโพเนนท์ที่ต้องการและจัดวางลงในส่วนของการออกแบบโดยจะทำผ่านส่วนของการออกแบบ (App Inventor Designer) ดังที่แสดงในภาพที่ แสดงให้เห็นถึงแอพพลิเคชั่นที่สร้างขึ้นในหน้าต่างเว็บเบราเซอร์ โดยด้านซ้ายจะเป็นส่วนของคอมโพเนนท์ที่ App Inventor เตรียมไว้ให้จัดเรียงเป็นหมวดหมู่ เช่น ปุ่ม (button) ข้อความ (label) กล่องข้อความ (text box) เป็นต้น ผู้ใช้ทำการเพิ่มคอมโพเนนท์ที่เลือกด้วยการคลิกลากลงไปวางไว้ในโปรเจ็ค



ภาพที่ 2.4 ภาพตัวอย่าง แอพพลิเคชั่นที่สร้างขึ้นด้วยโปรแกรม App Inventor 2 ส่วนออกแบบส่วนการเขียนโค้ด (App Inventor Blocks Editor)

หลังจากที่ทำการเลือกจัดวางคอมโพเนนท์ที่จะใช้สำหรับโปรเจ็คครบแล้ว ผู้ใช้จะสามารถเขียนโค้ดคำสั่งสำหรับแอพพลิเคชั่นได้ในส่วนการเขียนโค้ด (App Inventor Blocks Editor) สำหรับพื้นที่การทำงานในส่วนหน้าจอการเขียนโค้ดแสดงดัง ซึ่งจะประกอบไปด้วยคำสั่งที่อยู่ในรูปของบล็อกรวบรวมไว้บริเวณด้านซ้ายของหน้าจอ ผู้ใช้สามารถเลือกคำสั่งที่ต้องการโดยการคลิกลากบล็อกคำสั่งมาวางไว้ในโปรเจ็คคือบริเวณที่เป็นพื้นที่วางตรงกลางหน้าจอ ตัวอย่างของบล็อกคำสั่ง ซึ่งจะเป็นคำสั่งพื้นฐานที่ผู้ใช้จะนำมาใช้ในการสร้างแอพพลิเคชั่นขึ้นมา บล็อกเหล่านี้จะถูกแยกและจัดแบ่งออกเป็นกลุ่มๆ ตามลักษณะของคำสั่ง ตัวอย่างเช่น บล็อกข้อความที่ใช้ในการทำงานที่เกี่ยวข้องกับข้อความที่เป็นสายอักขระ บล็อกทางคณิตศาสตร์ที่ใช้ในการทำงานที่เกี่ยวข้องกับฟังก์ชันทางคณิตศาสตร์ ตัวเลข หรือเครื่องหมายทางคณิตศาสตร์ เป็นต้น App Inventor ยังสามารถสร้างกระบวนการทำงาน (procedure) และตัวแปร (variable) ได้โดยการเลือกใช้บล็อกในส่วนที่เกี่ยวกับการสร้างกระบวนการทำงานและเหตุการณ์ (event handler) ที่เกิดกับคอมโพเนนท์ โดยบล็อกที่เกี่ยวข้องกับคอมโพเนนท์จะถูกจัดเตรียมไว้ให้ตามคอมโพเนนท์ที่ผู้ใช้เลือกนำมาวางไว้ในโปรเจ็คและจัดเก็บรวมกันไว้ในแท็บ My Blocks แยกไว้ต่างหาก บล็อกที่เกี่ยวข้องกับคอมโพเนนท์เหล่านี้จะแบ่งออกได้เป็น 4 แบบตามประเภทของคำสั่ง คือ ประเภทการเรียกค่าคุณสมบัติ

จากคอมโพเนนท์ (property getter) ประเภทการกำหนดค่าคุณสมบัติให้กับคอมโพเนนท์ (property setter)ประเภทเหตุการณ์ (event handler) และประเภทการเรียกใช้กระบวนการทำงาน (method call)



ภาพที่ 2.5 ภาพตัวอย่าง แอพพลิเคชั่นที่สร้างขึ้นด้วยโปรแกรม App Inventorส่วนการเขียนโค้ด

2.5 แอพลิเคชั่น NETPIE Freeboard

NETPIE ย่อมาจาก Network Platform for Internet of Everything จากชื่อก็คือมันจะทำให้อุปกรณ์ต่างๆ บนโลกเราสามารถติดต่อกันง่ายขึ้น โดยไม่ต้องมานั่งกังวลอะไรทั้งสิ้นเพียงแค่ไว้ใจ NETPIE เพราะ NETPIE จะเป็นตัวดูแลการเชื่อมต่อทั้งหมดให้ ขอแค่เพียงอุปกรณ์นั้นอยู่บนอินเตอร์เน็ต เริ่มแรกเลยในการเริ่มใช้งาน netpie โดยเราเริ่มต้นจากการเข้าไปยังที่เว็บ NETPIE เลือก Get Started Now จะเป็นการอธิบายเกี่ยวกับความรู้พื้นฐานของ NETPIE ให้พอเข้าใจว่าเราควรจะเริ่มต้นตรงไหน และหลักการที่ netpie จะให้อุปกรณ์เชื่อมต่อกันทำอย่างไร หลังจากนั้นเข้ามายัง Github/Microgear-ESP8266 จะมีตัวอย่างการใช้งาน Microgear ที่เป็น library ให้เราสามารถเชื่อมต่อกับ NETPIE

ภาพที่ 2.6 ภาพตัวอย่างการทำงาน แอพพลิเคชั่น NETPIE

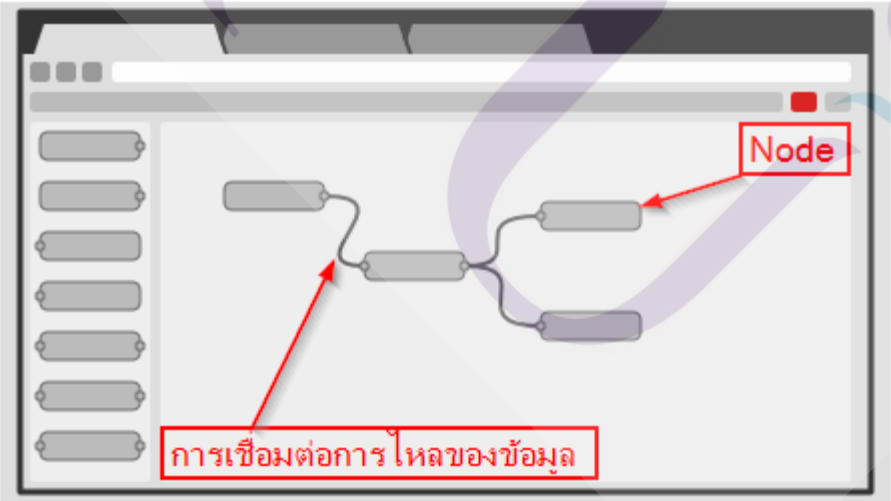
Freeboard คือ

Freeboard นั้นเป็น Web Application ที่ทำให้เราสามารถสร้าง Dashboard (แดชบอร์ด) สำหรับระบบ IoT ของเราขึ้นมาได้ครับ ถ้าจะให้พูดง่ายๆ ก็คือคล้ายๆกับการสร้างกระดาน IoT ส่วนตัวที่เราสามารถวางปุ่มกด,สวิชต์ ไว้ใช้สำหรับควบคุมอุปกรณ์ หรือวางหน้าปัดไว้สำหรับแสดงผลข้อมูลต่างๆที่ได้จากอุปกรณ์ในระบบ IoT ของเรานอกจากนี้ยังสามารถนำข้อมูลที่ได้มาไปแสดงผลเป็นกราฟและสามารถปรับแต่งได้ตามใจชอบได้ง่ายๆ เพียงแค่คลิก-ลาก-วาง และมีการป้อนข้อมูลหรือคำสั่งอีกนิดหน่อยก็สามารถทำงานได้แล้วครับคือง่ายมากๆ ไม่ต้องเสียเวลานั่งงมเขียนเป็น HTML Webpageให้วุ่นวายที่สำคัญคือข้อมูลนั้นมีการอัพเดทแบบ real-time มีความเสถียรและเชื่อถือได้และเป็น Open-Source ซึ่งทำให้นักพัฒนาสามารถต่อยอดให้ดียิ่งขึ้นได้อีกด้วย

ภาพที่ 2.7 ภาพตัวอย่างแอพพลิเคชั่น NETPIE บน Freeboard

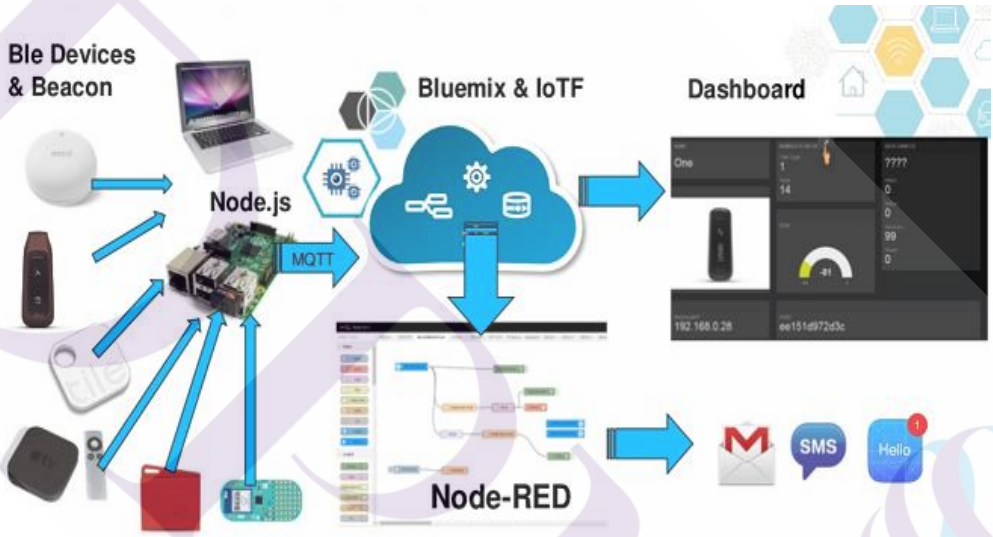
2.6 แอพลิเคชั่น Node-RED

Node-RED เป็นเครื่องมือสำหรับนักพัฒนาโปรแกรมในการเชื่อมต่ออุปกรณ์ฮาร์ดแวร์เข้ากับAPIs (Application Programming Interface) ซึ่งเป็นการพัฒนาโปรแกรมแบบ Flow-Based Programming ที่มีหน้า UI สำหรับนักพัฒนาให้ใช้งานผ่าน Web Browser ทำให้การเชื่อมต่อเส้นทางการไหลของข้อมูลนั้นเป็นเรื่องง่าย



ภาพที่ 2.8 ภาพตัวอย่างการเขียนโปรแกรมแอพลิเคชั่น Node-RED

เนื่องจาก Node-RED เป็น Flow-Based Programming นั้นทำให้เราแทบจะไม่ต้องเขียน Code ในการพัฒนาโปรแกรมเลย แค่เพียงเลือก Node มาวางแล้วเชื่อมต่อก็สามารถควบคุม I/O ได้ โดย Node-RED จะมี Node ให้เลือกใช้งานอย่างหลากหลายสามารถสร้างฟังก์ชัน JavaScript ได้ โดยใช้ Text Editor ที่มีอยู่ใน Node-RED และยังสามารถบันทึก Function, Templates, Flows เพื่อไปใช้งานกับงานอื่นต่อไป Node-RED นั้นทำงานบน Node.js ทำให้เหมาะสำหรับการใช้งานกับ Raspberry Pi เนื่องจากใช้ทรัพยากรน้อย ขนาดไฟล์ไม่ใหญ่และ Node.js ยังทำหน้าที่เป็นตัวกลางให้ Raspberry Pi สามารถติดต่อกับ Web Browser และอุปกรณ์อื่นๆ ได้

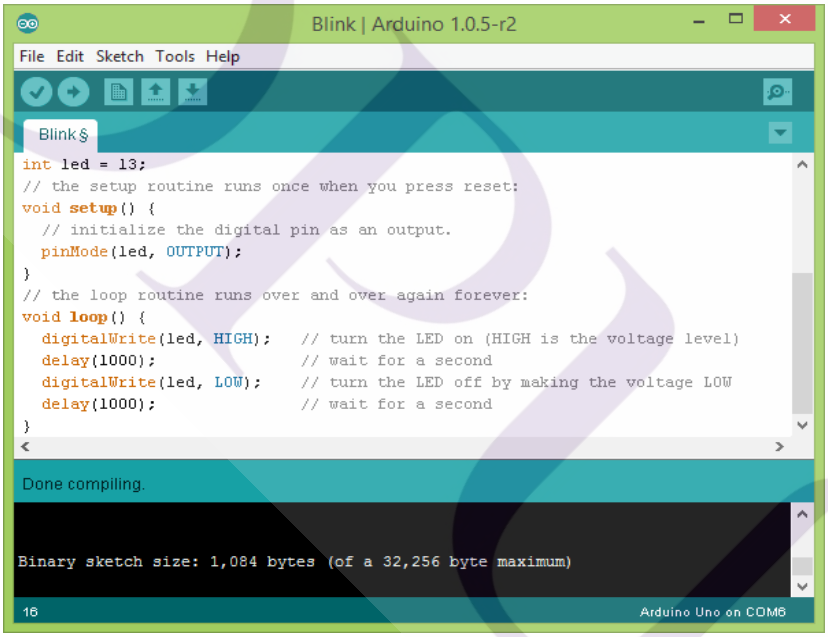


ภาพที่ 2.9 ภาพตัวอย่างการทำงานแอพพลิเคชั่น Node-RED

2.7 Arduino IDE

Arduino IDE (Arduino Integrated Development Environment) คือ โปรแกรมที่ใช้สำหรับสร้างชุดคำสั่ง(Coding) คอมไฟล์(Compiler) และอัพโหลด(Burn) โปรแกรมลงหน่วยประมวลผล Arduino โปรแกรม Arduino IDE เป็นโปรแกรมโอเพนซอร์ส (open source) ที่สามารถดาวน์โหลดได้ฟรีที่ www.arduino.cc โดยใช้ภาษาC ในการเขียนโปรแกรมสั่งการต่างๆ ระหว่างโหนดเซนเซอร์กับอุปกรณ์ควบคุม

ภาพที่ 2.10 ภาพตัวอย่างโปรแกรม Arduino IDE



ภาพที่ 2.11 ภาพตัวอย่างโปรแกรม Arduino IDE(ต่อ)

2.8 Internet of Things (IoT)

อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things : IOT) คือ เครือข่ายของวัตถุ อุปกรณ์พาหนะ สิ่งปลูกสร้าง และสิ่งของอื่นๆ ที่มีวงจรอิเล็กทรอนิกส์ ซอฟต์แวร์ เซ็นเซอร์ และการเชื่อมต่อกับเครือข่าย ฝังตัวอยู่ และทำให้วัตถุเหล่านั้นสามารถเก็บบันทึกและแลกเปลี่ยนข้อมูลได้ ซึ่งในปัจจุบันนั้น อินเทอร์เน็ต เรียกได้ว่ามีความจำเป็นไม่แพ้ปัจจัย4 เพราะทุกอย่างของการทำงาน

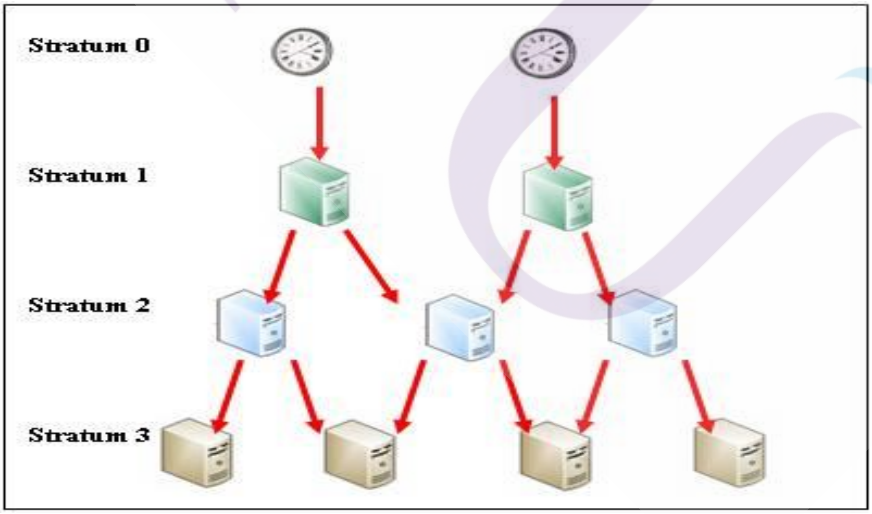
นั้นทำงานบนระบบเครือข่ายอินเทอร์เน็ตแทบทั้งสิ้น เช่น สามารถทำให้วัตถุสามารถรับรู้สภาพแวดล้อมและถูกควบคุมได้จากระยะไกลผ่านโครงสร้างพื้นฐานเครือข่ายที่มี อีกทั้งสามารถผสานโลกกายภาพกับระบบคอมพิวเตอร์ได้แนบแน่นมากขึ้น ทำให้มีประสิทธิภาพ ความแม่นยำและประโยชน์ทางเศรษฐกิจที่เพิ่มมากขึ้นตาม เมื่อระบบ IoT ถูกเสริมด้วยเซ็นเซอร์และแอคชูเอเตอร์ซึ่งสามารถเปลี่ยนลักษณะทางกลได้ตามการกระตุ้น กลายเป็นระบบที่ถูกจัดประเภทโดยทั่วไปว่าระบบไซเบอร์-กายภาพ (cyber-physical system) ซึ่งรวมถึงเทคโนโลยีอย่าง กริดไฟฟ้าอัจริยะ (สมาร์ตกริด) บ้านอัจฉริยะ (สมาร์ตโฮม) ระบบขนส่งอัจฉริยะ (อินเทลลิเจนต์ทรานสปอร์ต) และเมืองอัจฉริยะ (สมาร์ตซิตี้) วัตถุแต่ละชิ้นสามารถถูกระบุได้โดยไม่ซ้ำกันผ่านระบบคอมพิวเตอร์ฝังตัว และสามารถทำงานร่วมกันได้บนโครงสร้างพื้นฐานอินเทอร์เน็ตที่มีอยู่แล้วในปัจจุบัน และมีแนวโน้มที่จะเติบโตรวดเร็วอย่างต่อเนื่อง

ภาพที่ 2.12 ภาพตัวอย่างการเชื่อมต่อของระบบ Internet of Things (IoT)

2.9 NTP (Network Time Protocol)

NTP (เป็นโปรโตคอลสำหรับใช้เทียบเวลา (Synchronize) ระหว่างอุปกรณ์ที่ให้บริการเทียบเวลา (Time Server) กับคอมพิวเตอร์หรืออุปกรณ์ที่ต้องการเทียบเวลา (Time Client) ผ่านทางเครือข่ายการสื่อสารต่างๆ โดยโปรโตคอลจำทำงานที่พอร์ต 123 และใช้โปรโตคอล UDP ในการให้บริการ ลักษณะการให้บริการเทียบเวลาของโปรโตคอล NTP จะแบ่งออกเป็นลำดับชั้น เรียกว่า Clock Strata โดยในแต่ละลำดับชั้นจะเรียกว่า Stratum โดยจะเริ่มต้นอยู่ที่ Stratum 0 ไปจนถึงลำดับชั้นที่ยอมรับว่ายังมีความเที่ยงตรง คือ Stratum 4 หากมากกว่านี้จะไม่ได้รับการยอมรับตาม

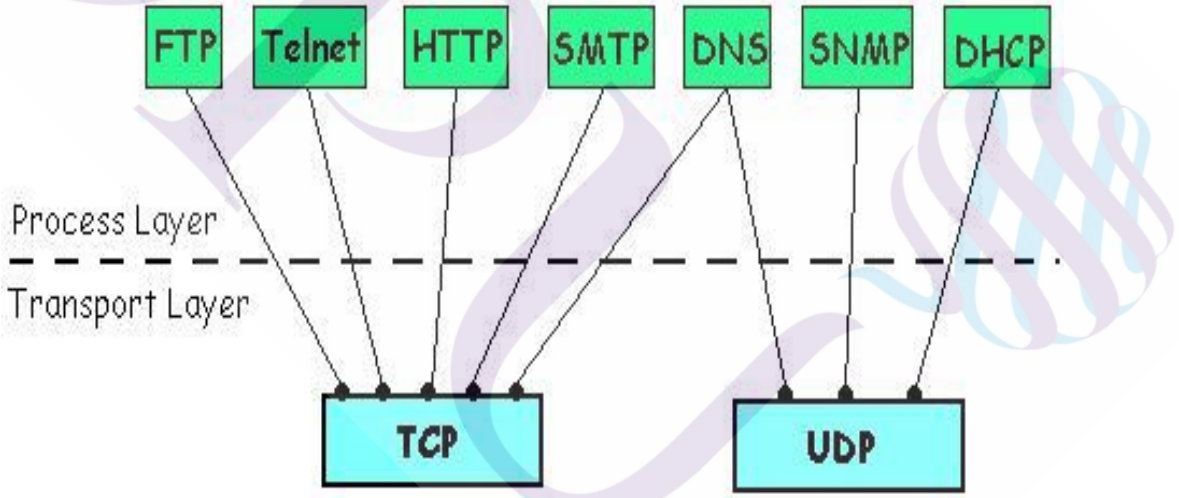
มาตรฐานที่กำหนดขึ้นมาจากหน่วยงาน ANSI (American National Standards Institute) สำหรับอุปกรณ์คอมพิวเตอร์ที่เทียบเวลากับ Stratum 0 เรียกว่า Stratum 1 ถ้ามีอุปกรณ์คอมพิวเตอร์อื่นๆ ขอเทียบเวลากับ Stratum 1 จะเรียกว่า Stratum 2 ตามลำดับ จนถึง Stratum 4 นั่นหมายถึงลำดับของ Stratum ที่มากขึ้นจะมีค่าเวลาที่มีความห่างกับเวลามาตรฐานสากล Stratum 0 มากขึ้นด้วย Stratum 0 เป็นลำดับชั้นแรกในการเทียบเวลา ซึ่งใช้อุปกรณ์ที่ทำหน้าที่ Synchronize เวลามาตรฐานสากลโดยไม่มีค่าหน่วงเวลาใดๆ โดยใช้เทคโนโลยีต่างๆ ได้แก่ Atomic Clock, คลื่นยาว (Long wave radio), การส่งสัญญา GPS, เทคโนโลยี CDMA (เทคโนโลยีแบบที่ค่ายมือถือ) หรืออุปกรณ์เกี่ยวกับเวลาอื่นๆ เช่น WWV, DCF77 ฯลฯ อุปกรณ์ที่เป็น Stratum 0 จะไม่ได้ต่อในระบบ Network แต่จะเชื่อมโดยตรงกับเครื่องที่ทำหน้าที่เป็น Stratum 1 ดังนั้นเซิร์ฟเวอร์ ที่ต่อโดยตรงกับ อุปกรณ์พวก Stratum-0 จะเรียกว่าเป็น Stratum-1 server ซึ่ง Stratum-1 server ถือว่าเป็น Time server ระดับต้น (Primary Time Server) ที่อยู่ในระบบ Network ที่ผู้ใช้บริการ Network Time Protocol (NTP) สามารถมาเชื่อมผ่าน Network มาอ้างอิงเวลาได้ Stratum 1 เป็นลำดับที่ใช้คอมพิวเตอร์เซิร์ฟเวอร์เชื่อมต่อเข้ากับ Stratum 0 เพื่อขอเทียบเวลา โดยใช้โปรโตคอล NTP ในประเทศไทยมีหน่วยงานที่ทำหน้าที่ระดับ Stratum 1 ได้แก่ สถาบันมาตรวิทยาแห่งชาติ และ กรมอุทกศาสตร์กองทัพเรือ Stratum 2 เป็นลำดับที่ขอเทียบเวลาจากเซิร์ฟเวอร์ในระดับ Stratum 1 โดยใช้เครื่องเซิร์ฟเวอร์เชื่อมผ่านระบบเครือข่ายอินเทอร์เน็ต สามารถร้องขอบริการเทียบเวลาได้มากกว่าหนึ่งแหล่ง Stratum เพื่อรองรับการทำงานกรณีที่ Stratum 1 เครื่องใดเครื่องหนึ่งไม่สามารถให้บริการได้



ภาพที่ 2.13 ภาพตัวอย่างแสดงลำดับชั้นของการเทียบเวลาใน NTP

2.10 UDP (User Datagram Protocol)

UDP เป็นการส่งข้อมูลที่ไม่มีการยืนยันการรับส่งข้อมูล คือผู้ส่งไม่สามารถรู้ได้ว่าข้อมูลได้ถึงผู้รับแล้วหรือไม่ เราจะต้องเขียนการตรวจสอบข้อมูลเอาเอง เช่น ถ้าฝั่งผู้ส่งข้อมูลไปและฝั่งที่รับข้อมูลได้รับข้อมูลแล้ว ก็ให้ตอบกลับมาว่าข้อมูลไปถึงแล้ว แต่ถ้าไม่มีการตอบกลับภายในระยะเวลาที่กำหนด ก็ให้แจ้งว่าผู้รับว่าไม่ได้รับข้อความ UDP อยู่ใน Transport layer(ทรานสโพทเลเยอร์) ทำหน้าที่จัดการและควบคุมการรับส่งข้อมูล แต่ไม่มีกลไกความคุมการรับส่งข้อมูลให้มีเสถียรภาพและเชื่อถือได้ UDP มีข้อได้เปรียบในการส่งข้อมูลได้ทั้งแบบ unicast (ยูนิคาสต์), multicast (มัลติคาสต์) และ broadcast (บรอดคาสต์) อีกทั้งยังทำการติดต่อสื่อสารได้เร็วกว่า TCP เนื่องจาก TCP ต้องเสีย overhead (โอเวอร์เฮด) ให้กับขั้นตอนการสื่อสารที่ทำให้ TCP มีความน่าเชื่อถือในการรับส่งข้อมูล จุดเด่นของ UDP ซึ่งมีจุดเด่นที่ความเร็วและไม่มีการทำงานเกี่ยวการส่งข้อมูลที่ซ้ำซ้อนจึงเหมาะกับการส่งข้อมูลแบบ realtime (เรียลไทม์) ซึ่งข้อมูลที่สูญหายบางส่วนหรือข้อมูลที่เกิด delay (ดิเลย์) จะถูกละความสนใจไป จึงทำให้การรับส่งข้อมูลเร็วกว่าแบบ TCP



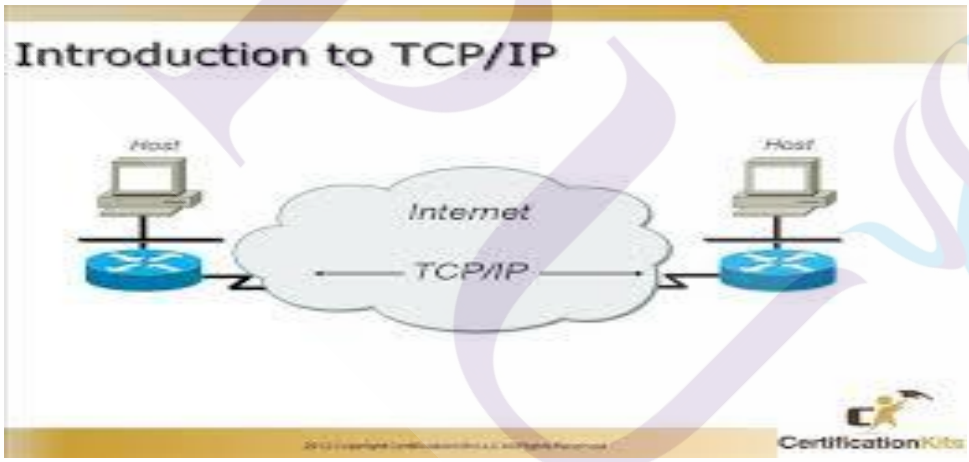
ภาพที่ 2.14 ภาพตัวอย่างแสดงลำดับชั้นเปรียบเทียบขั้นตอนการรับส่งข้อมูล TCP กับ UDP

2.11 TCP/IP (Transmission Control Protocol/ Internet Protocol)

TCP/IP คือ การที่เครื่องคอมพิวเตอร์ที่ถูกเชื่อมโยงกันไว้ในระบบจะสามารถติดต่อสื่อสารกันได้นั้น จำเป็นจะต้องมีภาษาสื่อสารที่เรียกว่าโปรโตคอล (Protocol) ซึ่งในระบบ Internet จะใช้ภาษาสื่อสารมาตรฐานที่ชื่อว่า TCP/IP เป็นภาษาหลัก ดังนั้นหากเครื่องคอมพิวเตอร์ไม่ว่าจะเป็นเครื่องระดับไมโครคอมพิวเตอร์ มินิคอมพิวเตอร์ หรือเมนเฟรมคอมพิวเตอร์ ก็สามารถเชื่อมโยงเข้าสู่อินเทอร์เน็ตได้ TCP/IP คือชุดของโปรโตคอลที่ถูกใช้ในการสื่อสารผ่านเครือข่ายอินเทอร์เน็ตโดยมีวัตถุประสงค์เพื่อให้สามารถใช้สื่อสารจากต้นทางข้ามเครือข่ายไปยังปลายทางได้และสามารถหาเส้นทางที่จะส่งข้อมูลไปได้เองโดยอัตโนมัติ TCP และ IP มีหน้าที่ต่างกัน คือ

1. TCP จะทำหน้าที่ในการแยกข้อมูลเป็นส่วน ๆ หรือที่เรียกว่า Package ส่งออกไป ส่วน TCP ปลายทาง ก็จะทำการรวบรวมข้อมูลแต่ละส่วนเข้าด้วยกัน เพื่อนำไปประมวลผลต่อไป โดยระหว่างการรับส่งข้อมูลนั้นก็จะมีการตรวจสอบความถูกต้องของข้อมูลด้วย ถ้าเกิดผิดพลาด TCP ปลายทางก็จะขอไปยัง TCPต้นทางให้ส่งข้อมูลมาใหม่

2. IP จะทำหน้าที่ในการจัดส่งข้อมูลจากเครื่องต้นทางไปยังเครื่องปลายทางโดยอาศัย IP Address

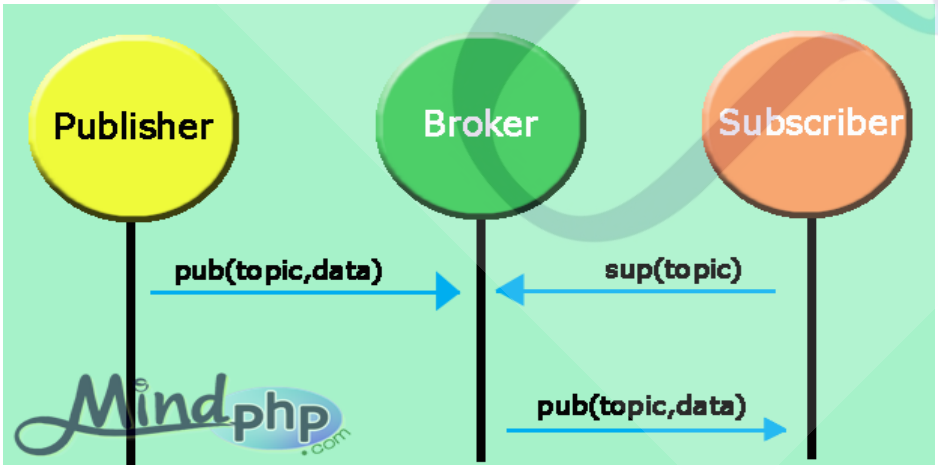


ภาพที่ 2.15 ภาพตัวอย่างแสดงลำดับชั้นขั้นตอนการรับส่งข้อมูล TCP/IP

2.12 MQTT Server Potocal

MQTT คือ Message Queuing Telemetry Transport (MQTT) เป็น Protocol ที่ออกแบบมาเพื่อการเชื่อมต่อแบบ M2M (machine-to-machine)คืออุปกรณ์กับอุปกรณ์ สนับสนุนเทคโนโลยี IoT (Internet of Things) คือเทคโนโลยีที่อินเทอร์เน็ตเชื่อมต่อกับอุปกรณ์ต่าง ๆ เช่น โทรศัพท์มือถือ รถยนต์ โทรทัศน์ ตู้เย็น เข้ากับอินเทอร์เน็ตทำให้สามารถเชื่อมโยงสื่อสารกับอุปกรณ์ต่าง ๆ ได้ โดยผ่านเครือข่ายอินเทอร์เน็ต ซึ่งจะทำให้มนุษย์สามารถ ควบคุมอุปกรณ์ต่าง ๆ จากที่อื่นได้ เช่นการสั่งปิดเปิดไฟในบ้านจากที่อื่นๆ เนื่องจากโปรโตคอลตัวนี้มีน้ำหนักเบาออกแบบมาเพื่อใช้งานกับอุปกรณ์อิเล็กทรอนิกส์ขนาดเล็ก การรับส่งข้อมูลในเครื่อข่ายที่มีขนาดเล็ก แบนร์วิธต่ำ ใช้หลักการแบบ publisher / subscriber คล้ายกับหลักการที่ใช้ใน Web Service ที่ต้องใช้ Web Server เป็นตัวกลางระหว่างคอมพิวเตอร์ของผู้ใช้ แต่ MQTT จะใช้ตัวกลางที่เรียกว่า Broker เพื่อทำหน้าที่ จัดการคิว รับ-ส่ง ข้อมูลระหว่างอุปกรณ์ และทั้งในส่วนที่เป็น Publisher และ Subscriber ดังภาพ

Broker ทำหน้าที่เป็นตัวกลางในการรับส่งข้อมูลระหว่าง Client โดยมีวิธีการสร้างเส้นทาง (Routing) ด้วยหัวข้อ (Topic) โดยที่ Client ต้องทำการ Subscribe ใน Topic ที่ตัวเองต้องการ จากนั้น Broker ก็จะส่งข้อมูลทั้งหมดที่ถูก Publish ใน Topic นั้นๆให้ โดยที่ Client สื่อสารกันโดยที่ไม่รู้จักกัน ซึ่งถือเป็นข้อดีเมื่อต้องการขยายเครือข่ายก็สามารถดำเนินการได้ง่ายและหน้าที่ที่สำคัญมากอีกอย่างของ broker คือรักษาความปลอดภัยของเครือข่าย เช่น การทำ Authorization และ Authentication ของ Client



ภาพที่ 2.16 ภาพตัวอย่างแสดงลำดับขั้นขั้นตอนการรับส่งข้อมูล MQTT

2.13 งานวิจัยที่เกี่ยวข้อง

หัวข้อนี้เป็นการรวบรวมบทความหรืองานวิจัยที่เกี่ยวข้องกับสารนิพนธ์การเปรียบเทียบการพัฒนาโปรแกรมประยุกต์IOTสำหรับโทรศัพท์มือถือ ระหว่าง IOT Cloud Platfrom Application และ End-to-End Application โดยมีวรรณกรรมและงานวิจัยที่เกี่ยวข้องดั่งต่อไปนี้

สุชาดา พลาชัยภิรมย์ ได้จัดทำวิจัยเรื่องแนวโน้มการใช้แอพพลิเคชั่น มหาวิทยาลัยกรุงเทพ โดยแนวโน้มการใช้งาน Mobile Device อย่างสมาร์ทโฟนเพิ่มขึ้นอย่างก้าวกระโดดในช่วงไม่กี่ปีที่ผ่านมา ซึ่งเป็นผลมาจากการพัฒนา Mobile Applications และเทคโนโลยี ของตัวเครื่องโทรศัพท์จากค่าย ผู้ผลิตโทรศัพท์ โดยเฉพาะการพัฒนาต่อยอดแอพพลิเคชั่นบนอุปกรณ์เคลื่อนที่ของบริษัทต่างๆ ที่ แข่งขันกันเพื่อชิงความเป็นหนึ่งในตลาดด้าน Mobile Application ซึ่งการพัฒนาแอพพลิเคชั่นแบ่งเป็นการ พัฒนาแอพพลิเคชั่นระบบ (Operation System) และแอพพลิเคชั่นซอฟต์แวร์ที่ ตอบสนองการใช้งานบนอุปกรณ์ และด้วยแอพพลิเคชั่นที่เพิ่มขึ้นและมีประสิทธิภาพมากขึ้นทำให้ผู้ใช้ อุปกรณ์เคลื่อนที่มีแนวโน้มใช้โปรแกรมต่างๆ เพื่อตอบสนองกิจกรรมในชีวิตประจำวันได้แก่ ท า ธุรกรรมทางการเงิน เชื่อมต่อและสืบค้นข้อมูลบนเครือข่ายอินเตอร์เน็ต ชมภาพยนตร์ ฟังเพลง หรือ แม้แต่การเล่นเกมส์ ซึ่งมีทั้งออนไลน์ และออฟไลน์ด้วยอัตราการขยายตัวด้านการใช้งานอุปกรณ์ เคลื่อนที่ ทำให้บริษัทชั้นนำด้านโทรศัพท์มือถือหลายแห่งหันมาให้ความสำคัญกับการพัฒนาโปรแกรมบนโทรศัพท์มือถือ โดยเชื่อว่าจะมีอัตราการดาวน์โหลดเพื่อใช้งานที่เติบโตอย่างเห็นได้ชัดเจน

พิชิต บุญครอง* และ พิศณุ บุรมศรี ได้จัดทำวิจัยเรื่องการพัฒนาแอพพลิเคชั่นเพื่อการพยากรณ์อนุกรมเวลาโดยใช้MIT App Inventor MIT App Inventor-Based Application for Time Series Forecasting สาขาวิชาสารสนเทศการลงทุน วิทยาลัยเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยรังสิต งานวิจัยนี้นำเสนอหลักเบื้องต้นในการพัฒนาแอพพลิเคชั่นสำหรับการพยากรณ์อนุกรมเวลาโดยใช้ MIT App Inventor แอพพลิเคชั่นจะประมวลผลข้อมูลอนุกรมเวลาวิยุตเพื่อการพยากรณ์เชิงปริมาณพื้นฐาน 8 วิธี ได้แก่ วิธีการนาอีฟลำดับแรกวิธีการ นาอีฟแบบเพิ่มผลต่างวิธีนาอีฟแบบอัตราการเปลี่ยนแปลง วิธีนาอีฟสำหรับข้อมูลฤดูกาล วิธีนาอีฟสำหรับข้อมูลแนวโน้มและฤดูกาล วิธีค่าเฉลี่ยธรรมดา วิธีค่าเฉลี่ยเคลื่อนที่ธรรมดาและวิธีค่าเฉลี่ยเคลื่อนที่เอ็กซ์โปเนนเชียล ซึ่งข้อมูลแต่ละชุดจะเหมาะสมกับ วิธีพยากรณ์ต่างกัน แอพพลิเคชั่นที่สร้างขึ้นจะทำการเปรียบเทียบความแม่นของวิธีพยากรณ์และเลือกวิธีพยากรณ์ที่มีรากที่สอง ของค่าเฉลี่ยกำลังสองของความคลาดเคลื่อน (RMSE) ที่น้อยที่สุดเพื่อพยากรณ์ข้อมูลชุดนั้น โครงการวิจัยนี้บูรณาการระหว่างระเบียบวิธีพยากรณ์และการเขียนแอพพลิเคชั่นผ่าน Google App Inventor Servers, MIT App Inventor และ สมาร์ทโฟน ระบบแอนดรอยด์ การพยากรณ์เชิงปริมาณของอนุกรมเวลาควรใช้ตัว

แบบที่มีความซับซ้อนน้อยที่สุดและให้ความคลาดเคลื่อนน้อย ที่สุดเป็นที่คาดหวังว่าแอพพลิเคชั่นที่พัฒนาขึ้นจะอำนวยความสะดวกและเป็นประโยชน์ต่อการตัดสินใจที่เกี่ยวข้องกับอนุกรมเวลาในชีวิตประจำวันได้

จามจุรี กุลยอด1 , ศิลป์ณรงค์ ฉวีพัฒน์2 ได้จัดทำวิจัย เรื่องต้นแบบระบบควบคุมการเปิด-ปิดไฟผ่านแอพพลิเคชั่น บนระบบปฏิบัติการแอนดรอยด์ Wireless Light Switch Prototype with an Android Application สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกำแพงเพชร ปัจจุบันเทคโนโลยีต่างๆ ได้มีการพัฒนาอย่างต่อเนื่อง ทำให้ผู้ใช้ได้รับความสะดวกสบาย เนื่องจากการใช้ ชีวิตประจำวันของมนุษย์เป็นไปด้วยความเร่งด่วนต้องการความสะดวกประกอบกับเทคโนโลยีทางด้านคอมพิวเตอร์ และอิเล็กทรอนิกส์โดยเฉพาะระบบเครือข่ายไร้สายซึ่งกำลังมีการขยายตัวและใช้งานกันอย่างแพร่หลาย การเปิด-ปิด ไฟภายในบ้านของเรา ยังจำเป็นที่จะต้องเดินไปเปิดปิดไฟตามจุดต่างๆ ที่ต้องการต้องการใช้งาน ทำให้เกิดความ ล่าช้าในเวลาเร่งด่วน รวมไปถึงอาจจะทำให้ลืมปิดไฟตามจุดต่างๆ ที่เราได้เปิดไว้ จนทำให้เกิดความกังวลเมื่อเราออก จากบ้านไปแล้ว และยังทำให้เกิดการสิ้นเปลืองค่าไฟฟ้า และอาจจะก่อให้เกิดอัคคีภัยได้ งานวิจัยนี้ได้พัฒนาต้นแบบ ควบคุมการปิดเปิดไฟโดยใช้อุปกรณ์โนด ESP8266/Node MCU โดยต่อผ่านชุดควบคุมสวิทช์ไฟฟ้า 220 โวลต์ ผลจากการทดสอบ สามารถที่จะควบคุมการเปิด-ปิดไฟ ผ่านแอพพลิเคชั่นบนอุปกรณ์ที่รองรับระบบปฏิบัติการ แอนดรอยด์นอกจากนี้ยังสามารถตรวจสอบสถานการณ์ทำงานของอุปกรณ์ผ่านแอพพลิเคชั่นดังกล่าวได้ว่าความสำคัญ: ESP8266 / Node MCU / ระบบปฏิบัติการ แอนดรอยด์/ ส่วนควบคุมเปิด-ปิดไฟ

สุจินต์ ณ นคร ได้จัดทำวิจัย เรื่องระบบควบคุมผักสลัดอัจฉริยะ(Smart Salad Planting) สาขาวิชาวิศวกรรมวิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์ ในงานวิจัยนี้ ผู้วิจัยได้ทำการประยุกต์ใช้ IoT เพื่อมาช่วยจัดการระบบควบคุมปลูกผักสลัดอัจฉริยะ โดยอาศัยหลักทฤษฎีของ ระบบไฮโดรโปนิกส์ ที่สามารถควบคุมการเพาะปลูกโดยที่สามารถควบคุมได้จากระยะไกลผ่านโทรศัพท์เคลื่อนที่ เพื่อการใช้งานที่สะดวกสบาย พร้อมทั้งมีระบบจัดเก็บข้อมูล เพื่อติดตามผลได้ในระดับต่อไป โดยการนำอุปกรณ์เซ็นเซอร์ต่าง ๆ มาติดตั้งพร้อมทั้งการทำงานในระบบย่อยส่วนอื่น ๆ ในการจัดการระบบ ระบบควบคุมการปลูกผักสลัดอัจฉริยะ สามารถใช้เพาะปลูกผักสลัดและการเพาะปลูกพืชในระบบไฮโดรโปนิกส์ต่างๆ ยังนำไปประยุกต์ใช้กับพืชผักชนิดอื่น ๆ ได้หลากหลายในระดับต่อไปได้ ใช้แก้ปัญหา การขาดแคลนน้ำและแหล่งเพาะปลูกที่มีอย่างจำกัด สภาพแวดล้อมที่ต้องการความควบคุมอย่างเหมาะสมกับการเพาะปลูกของพืชแต่ละชนิด

ฐิติพงษ์ รักษาริกรณ์ ได้จัดทำวิจัยเรื่อง แนวทางการประยุกต์ใช้ Internet of Things (IoT) กับ Smart Agriculture 4.0 สาขาวิศวกรรมข้อมูลขนาดใหญ่ วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์ ปัจจุบันได้เริ่มมีการนำเกษตรกรรมบทบาทมากขึ้นในด้านการพัฒนาประเทศอย่างยั่งยืน ซึ่งสอดคล้องกับนโยบายของรัฐบาล Thailand 4.0 เน้นเศรษฐกิจขับเคลื่อนด้วยนวัตกรรม Value-Based Economy โดยทำให้เห็นความสำคัญการคิดค้นนวัตกรรมใหม่ๆ เพื่อช่วยขับเคลื่อนเศรษฐกิจใน Thailand 4.0 ซึ่งเรื่องที่เป็นจุดเน้นมากที่สุด คือ กลุ่มอาหาร เกษตร และเทคโนโลยีชีวภาพ (สำนักวิชาการ, สำนักงานเลขาธิการสภาผู้แทนราษฎร. 2559) ถ้าเรามองอัตราการเติบโตของจีดีพี 15 ปีย้อนหลัง ประมาณปี 2545-2548 ช่วง 3 ปีนี้จีดีพีโตประมาณ 6-7% พอมาในช่วง 2549-2550 ขยับลงมาเหลือ 4-5% ปี 2550-2551 เหลือประมาณ 0.1 ตอนนั้นเริ่มมีวิกฤติเศรษฐกิจที่อเมริกา มาถึงช่วง 3 ปีสุดท้ายที่ผ่านมา ประมาณ 0-2% ปีสุดท้ายประมาณ 2.8% ถ้าดูอย่างนี้ เปรียบเทียบจีดีพีเฉลี่ย 15 ปี ประมาณ 4% กว่าๆ ถ้า 10 ปีเฉลี่ย 3%กว่าๆ ถ้า 5 ปีเหลือแค่ 2%กว่าๆ ไม่ใช่เพราะการเมืองอย่างเดียว แต่เป็นเพราะว่าโครงสร้างเศรษฐกิจของเรามีปัญหา ความไม่เท่าเทียมกันขยายห่างขึ้นเรื่อยๆ ภาคที่แข็งแรงคือภาคอุตสาหกรรม ขณะที่ภาคเกษตรเตี้ยลงเรื่อยๆ สิ่งแวดล้อมเสียหาย ปีก่อนน้ำท่วม ปีนี้แล้ง ถ้าเราปล่อยให้เป็นอย่างนี้ต่อไปอนาคตเมืองไทยมีปัญหาแน่นอน (สมคิด จาตุศรีพิทักษ์ 2559)เกษตรกรไทยยุค THAILAND 4.0 จุดเริ่มต้นหรือที่มาส่วนหนึ่งของ Smart Farmer คือ การไม่ทำร้ายธรรมชาติ ใช้ทรัพยากรเท่าที่จำเป็น ทำแล้วต้องสบายขึ้นเรื่อย ๆ ไม่ใช่ยิ่งทำยิ่งเหนื่อย เช่น การมีพื้นที่เล็ก ๆ แต่สามารถออกแบบให้ปลูกแบบผสมผสานและเกื้อกูลกันได้ ต้องใช้เทคโนโลยีเป็น ซึ่งก็ถูกต้อง เพราะคนที่จะเป็น Smart Farmer ต้องเชื่อมโลกได้เอง Smart Farmer ต้องเข้าใจตั้งแต่กระบวนการผลิต การบริหารจัดการ เข้าใจธรรมชาติ และเข้าใจเทคโนโลยี (สุมิท แช่มประสิทธิ์, 2559)จึงเกิดแนวคิดการประยุกต์ใช้ Internet of Things ช่วยในการจัดการปลูกพืชในครัวเรือน ในพื้นที่ที่มีจำกัดให้ได้ผลประโยชน์มากที่สุด ได้เชื่อมโยงไปถึง Smart Farmer กล่าวคือตัวเกษตรกรต้องเป็นบุคคลที่มีความรู้ในด้านเกษตรกรรมและเทคโนโลยี สามารถนำมาประยุกต์ใช้ร่วมกันได้อย่างสมบูรณ์ สามารถแก้ไขปัญหาได้ มีความคิด รู้จักการวางแผนงาน และเป็นคนที่รู้จักใช้เทคโนโลยีเพื่อลดปัญหาเรื่องของแรงงาน

ดร.มหาศักดิ์ เกตุฉ่ำ ได้จัดทำวิจัยเรื่อง Internet of Things (IoT) ภาควิชาการจัดการเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ งานวิจัยนี้เป็นการกล่าวถึงต้นประวัติที่มาและความสำคัญของInternet of Things (IoT) และการนำ Internet of Things (IoT)ไปประยุกต์ใช้งานในด้านต่างๆของชีวิตประจำวันมนุษย์ต่อไป

บทที่ 3
วิธีดำเนินการวิจัย

3.1 แนวทางการวิจัยและพัฒนา

งานวิจัยนี้ได้ทำการวิจัยพัฒนาออกแบบเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานระหว่างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่นInventor 2 โดยการประยุกต์ใช้งานจากเทคโนโลยีของอินเตอร์เน็ตสรรพสิ่ง Internet of Things (IoT) โดยทดสอบผ่านเครื่องอาหารปลา ทั้งนี้ได้วางแผนดำเนินการตามขั้นตอน โดยเริ่มจาก ศึกษาค้นคว้าทฤษฎีของการออกแบบสร้างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่นInventor 2 รวมถึงเทคนิคและรับส่งข้อมูลของแอพพลิเคชั่นให้มีประสิทธิภาพและ ศึกษาการทำงานของอุปกรณ์(IoT) ที่จำเป็นต่อการให้อาหารปลาในแต่ละส่วนทั้งนโยบายเงื่อนไขข้อกำหนดและสิทธิ์ในการใช้งานภาษาที่ต้องใช้เพื่อเขียนโปรแกรมการทำงานของแอพพลิเคชั่นเพื่อสำหรับสั่งการอุปกรณ์และในส่วนแอพพลิเคชั่นในการนำมาประยุกต์ใช้งานเพื่อนำมาพัฒนาอุปกรณ์และ ออกแบบหน้าตาในส่วนของ User Interface เพื่อให้ง่ายต่อการใช้งานของผู้ใช้งานและทำการทดสอบเปรียบการทำงานระหว่างสองแอพพลิเคชั่นโดยการรวมของทุกส่วนประกอบและหาจุดบกพร่องแล้วนำมาปรับปรุงแก้ไขและสุดท้ายสรุปผลงานวิจัยโดยได้มีการวางแผนการดั่งนี้

3.2 แผนการดำเนินงาน

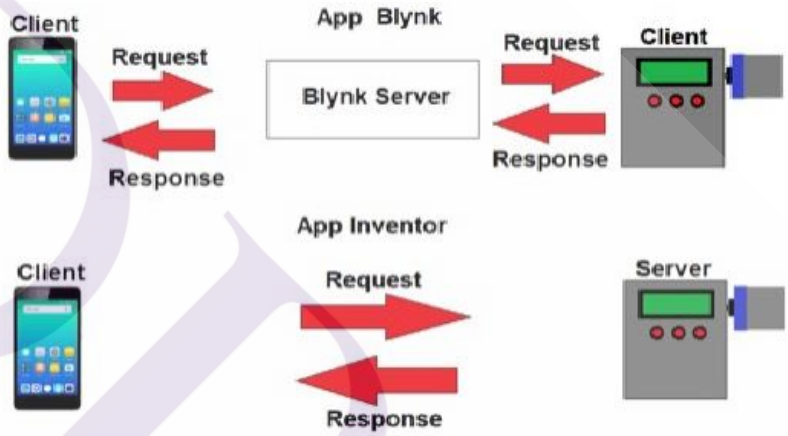
ผู้วิจัยได้วางแผนการและระยะเวลาดำเนินงานโดยแบ่งขั้นตอนดำเนินงานเป็น 8 ขั้นตอน และใช้ระยะเวลาในการดำเนินงานทั้งสิ้นเป็นเวลา 12 เดือน โดยเริ่มต้นตั้งแต่เดือนกันยายน พ.ศ.2560 จนถึง เดือนพฤษภาคม พ.ศ.2562 โดยรายละเอียดของแผนการดำเนินงานดูได้จากตารางที่ 3.1

ตารางที่ 3.1 แผนการดำเนินงาน

แผนการดำเนินงานดำเนินงาน ตามตารางที่ 3.1	ปี 2561-2562								
	ก.ย. 61	ต.ค. 61	พ.ย. 61	ธ.ค. 61	ม.ค. 62	ก.พ. 62	มี.ค. 62	เม.ย. 62	พ.ค. 62
1.ศึกษาค้นคว้าทฤษฎีและงานวิจัยในแต่ละขั้นตอนวิธี	→	→	→						
2.วิเคราะห์และออกแบบภาพรวมของระบบ			→	→	→				
3.ศึกษาพัฒนาอุปกรณ์และโปรแกรมประยุกต์ IoT			→	→	→				
4.พัฒนาอุปกรณ์ IoT สำหรับใช้งานระบบแอพพลิเคชั่น			→	→	→	→			
5.ทดสอบออกแบบเปรียบเทียบประสิทธิภาพแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2					→	→	→		
6.ปรับปรุงแก้ไขจุดบกพร่องในการออกแบบทดสอบประสิทธิภาพแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2					→	→	→		
7.ศึกษาทดสอบแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2 กับ เครื่องให้อาหารปลา							→	→	
8.สรุปผลงานวิจัยและนำเสนอ									→

3.3 ภาพรวมของระบบ

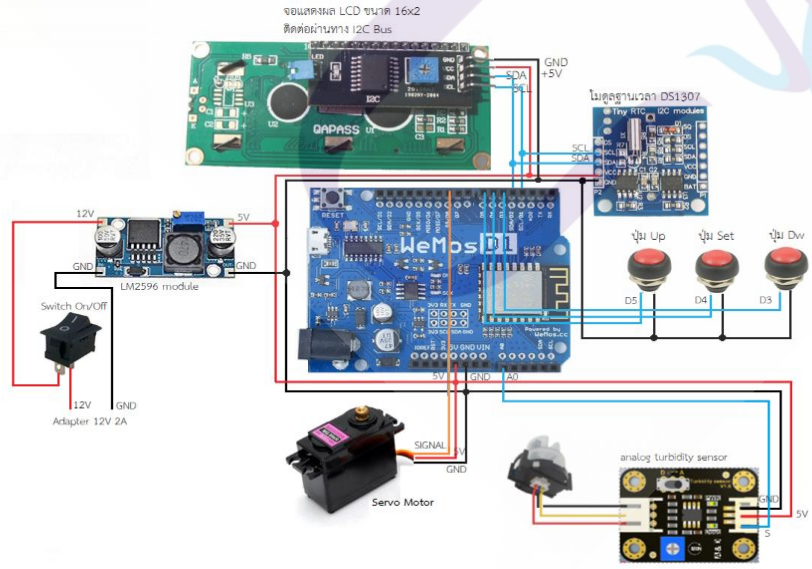
งานวิจัยนี้ได้ทำการวิจัยพัฒนาออกแบบเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานระหว่างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่นInventor 2 โดยการประยุกต์ใช้งานจากเทคโนโลยีของอินเตอร์เน็ตสรรพสิ่ง Internet of Things (IoT) โดยทดสอบผ่านเครื่องอาหารปลา โดยแบ่งออกเป็น 4 ส่วนการทำงานย่อย ได้แก่ ส่วนออกแบบแอพพลิเคชั่น , ส่วนการรับส่งข้อมูล, ส่วนการบริหารจัดการผ่านโทรศัพท์เคลื่นที่และ ส่วนทดสอบระบบเครื่องให้อาหารปลาสำหรับผู้ใช้งาน ตามภาพที่ 3.1



ภาพที่ 3.1 ภาพรวมของระบบ

3.4 การออกแบบแผนผังวงจรการทำงาน

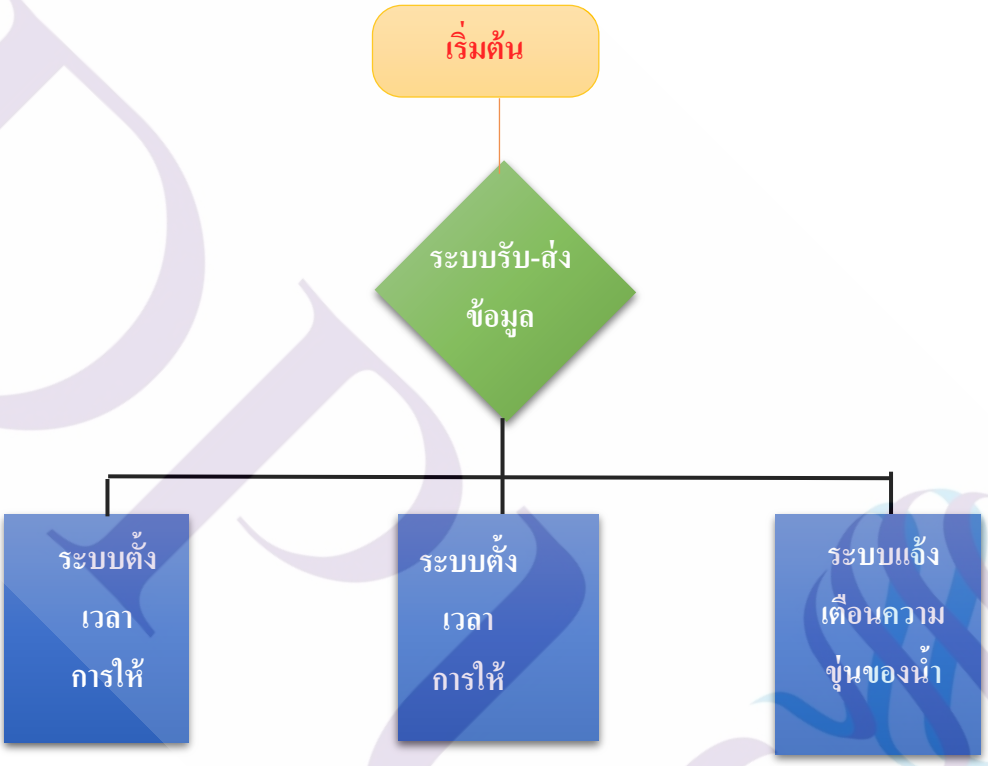
ในส่วนการออแบบวงจรการทำงานนั้นในส่วนของจอ LCD จะเป็นการเชื่อมต่อสื่อสารแบบพาราเรียลซึ่งจะใช้ขาในการเชื่อมต่อIoถึง8ขาซึ่งจะยุ่งยากและเพิ่มภาระให้กับบอร์ด Wemos D1 ผู้วิจัยจึงออกแบบวงจรLCD ให้เชื่อมต่อแบบ I2cbusแทนแบบพาราเรียล ซึ่งจะลดจำนวนขาที่ใช้ในการเชื่อมต่อจากเดิมต้องใช้ถึง8ขาในการเชื่อมแต่ถ้าใช้การเชื่อมต่อแบบI2cbusนั้นจะเหลือIoแค่4ขาในการเชื่อมต่อเท่านั้นซึ่งจะได้แก่ขาไฟเลี้ยง 5V. ขา Groud และอีก 2 ขาเป็น ขา SDA กับ SCL ซึ่งขา SDA จะเป็นขาในส่วนการรับส่งข้อมูลและขา SCL จะเป็นขาในการให้หน่วงเวลาในการรับส่งข้อมูล และในส่วนตัวบอร์ด RTC จะเป็นบอร์ดที่ดึงเวลาจากเว็บผู้ให้บริการเวลาสากลโดยผู้วิจัยจะให้บอร์ดตัวนี้ดึงเวลามาจากเว้บโดยจะทำการเขียนฟังช์ชั่นในตัวบอร์ด RTC บวกไป7ชั่วโมงเพื่อให้ปรับเป็นเวลาในประเทศไทยจากนั้นก็จะดำเนินการต่อพ่วงกับ บอร์ด จอ LCD ต่อในการต่อพ่วงกันนั้นต้องรู้ ค่า Adess ของบอร์ดที่จะเชื่อมต่อกันจึงจะพ่วงต่อกันได้จากนั้นในส่วนของปุ่มก็จะมี3ปุ่มได้แก่ปุ่ม Up เพิ่มค่าปุ่ม Down ลดค่า และปุ่มเซ็ตในการเซ็ตค่าโดยจะต่อกับให้กับบอร์ด Wemos D1ในขาที่ 12,13และ 14 และในส่วนของมอเตอร์ Servo ก็จะต่อเข้าในขาที่15ให้กับบอร์ด Wemos D1 และในส่วนของเซ็นเซอร์วัดความขุ่นของน้ำนั้นจะเชื่อมต่อเข้าขาวัดค่าแรงดันของบอร์ด Wemos D1ซึ่งมีอยู่1ขาพอดีและตัวแปลงไฟผู้วิจัยใช้ตัวแปลงไฟแยกจากจากให้กับบอร์ด Wemos D1 แปลงจาก12V.เป็น5V.3A ซึ่งจะช่วยลดภาระการทำงานของให้กับบอร์ด Wemos D1 ได้ดังภาพที่ 4.1



ภาพที่ 3.2 แสดงแผนผังวงจร

3.5 การออกแบบการทำงานของระบบเครื่องให้อาหารปลา

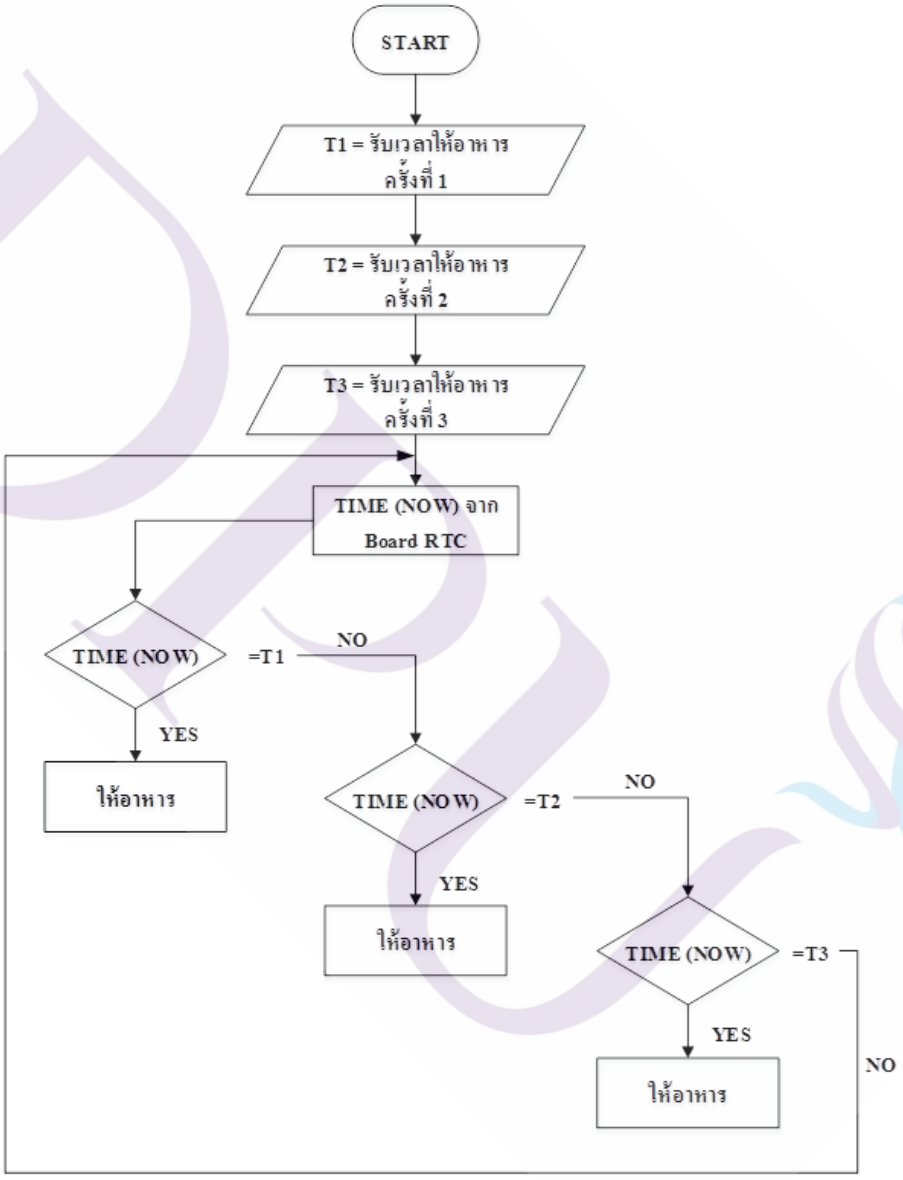
การทำงานของระบบเริ่มจากการรับค่าจากเซ็นเซอร์แล้วจะส่งค่าที่ได้รับมาไปเก็บไว้ที่ระบบแอพพลิเคชั่น Blynk และ แอพพลิเคชั่นInventor 2 โดยส่งผ่านไปยัง Gateway แล้วจะนำค่าที่ได้มาคำนวณต่อเพื่อเทียบกับค่าที่ตั้งไว้หรือค่าที่รับมาจากส่วน User Interface เมื่อได้ผลลัพธ์แล้วจึงสั่งงานไปยังอุปกรณ์เพื่อทำการสั่งการอุปกรณ์ โดยระบบมีการแบ่งการทำงานออกเป็นระบบย่อยๆ อีกหลายอย่างดังภาพที่ 4.2



ภาพที่ 3.3 ผังการทำงานของระบบเครื่องให้อาหารปลา

3.6 การออกแบบระบบการให้อาหารปลาตามช่วงเวลา

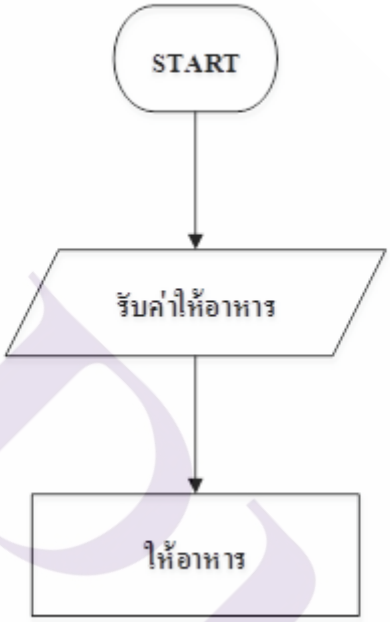
ในการออกแบบการให้อาหารปลา3ช่วงเวลานั้นผู้วิจัยได้ทำการเขียนชุดคำสั่งบนแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor 2 ให้สั่งการตั้งค่าเวลาในการให้อาหารปลาได้ 3 ช่วงเวลาโดยจะกำหนดเวลาและปริมาณอาหารได้ในแต่ละช่วงเวลาที่กำหนดไว้ดังภาพที่ 4.3



ภาพที่ 3.4 แสดงผังการทำงานการให้อาหารปลาตามช่วงเวลาที่กำหนด

3.7 การออกแบบระบบให้อาหารปลาอัตโนมัติ

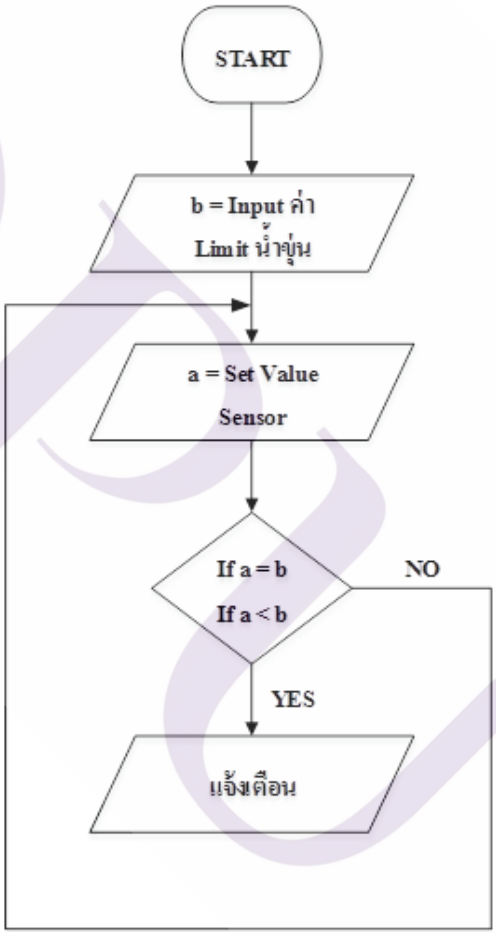
ผู้วิจัยได้ออกแบบระบบการสั่งให้อาหารปลาอัตโนมัติโดยกดปุ่มผ่านบนแอพพลิเคชั่น Blynk และ แอพพลิเคชั่นInventor 2 ได้ โดยสามารถให้อาหารปลาอัตโนมัติได้ทันที โดยผู้วิจัยได้เขียนชุดคำสั่งการทำงานให้Motor Servo หมุน1ครั้งกรณีที่ผู้ใช้กดปุ่มให้อาหารปลาอัตโนมัติโดยทันทีดังภาพที่ 4.4



ภาพที่ 3.5 แสดงผังการทำงานการให้อาหารปลาอัตโนมัติ

3.8 การออกแบบระบบการแจ้งเตือน

ในส่วนการออกแบบระบบการแจ้งเตือนผู้วิจัยได้ออกแบบการทำงานโดยการรับค่าจาก Sensor Node โดย ค่าที่ได้รับจะเป็นสัญญาณไฟฟ้า HIGH , LOW โดยจะมีSensor Node เป็นตัวจับค่าความขุ่นของน้ำและจะแจ้งเตือนความขุ่นของน้ำภายในตู้ปลาที่มีค่าความขุ่นของน้ำภายในตู้เกินค่าที่ผู้ใช้ได้กำหนดและตั้งค่าความขุ่นของน้ำในตู้ปลาไว้ผ่านแอพพลิเคชั่น Blynk และแอพพลิเคชั่น Inventor2 ดังภาพที่ 4.5



ภาพที่ 3.6 แสดงผังการทำงานการแจ้งเตือนความขุ่นของน้ำภายในตู้ปลา

3.9 การออกแบบแอพพลิเคชั่น Blynk

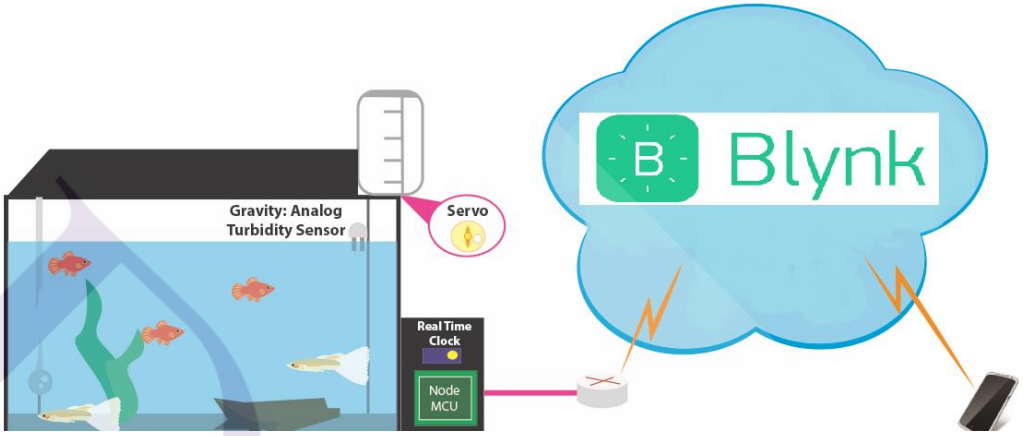
การออกแบบในส่วน แอพพลิเคชั่น Blynk จะทำหน้าเป็นที่รับ-ส่งข้อมูลต่างๆและแสดงผลขึ้นทาง แอพพลิเคชั่น โดยส่วนนี้จะให้บริการข้อมูลจากเซนเซอร์ผ่านกระดานข้อมูล (Dashboard) ในรูปแบบต่างๆ เช่น เวลาการให้อาหาร3ช่วงเวลาหรือการให้อาหารปลาอัตโนมัติ และค่าตัวเลขแสดงสถานะความขุ่นของน้ำในตู้ปลา เป็นต้น การออกแบบนั้นดูสวยงามและเข้าใจสถานะต่างๆ ได้ง่าย และสามารถเพิ่มเติมอุปกรณ์ต่างๆ บนกระดานข้อมูลได้ด้วยตนเองได้ และแอพพลิเคชั่น Blynk ยังมี Server และLibrary สำหรับ อุปกรณ์ Iot เป็นแบบสำเร็จรูปจึงทำให้สะดวกต่อการเขียนโปรแกรมและลดขั้นตอนการเขียนโปรแกรมได้อีกด้วยแต่ แอพพลิเคชั่น Blynk นั้นจะมีในส่วนค่าใช้จ่ายเพิ่มเข้ามาในการซื้อENERGY BALANCE เพื่อที่ผู้ใช้จะสร้างไอค่อนต่างๆ แอพพลิเคชั่น Blynk โดยการซื้อ ENERGY BALANCE นั้นจะซื้อผ่านบัตรเดบิตหรือบัตรเครดิตเท่านั้น

ภาพที่ 3.7 แสดงแอพพลิเคชั่น Blynk

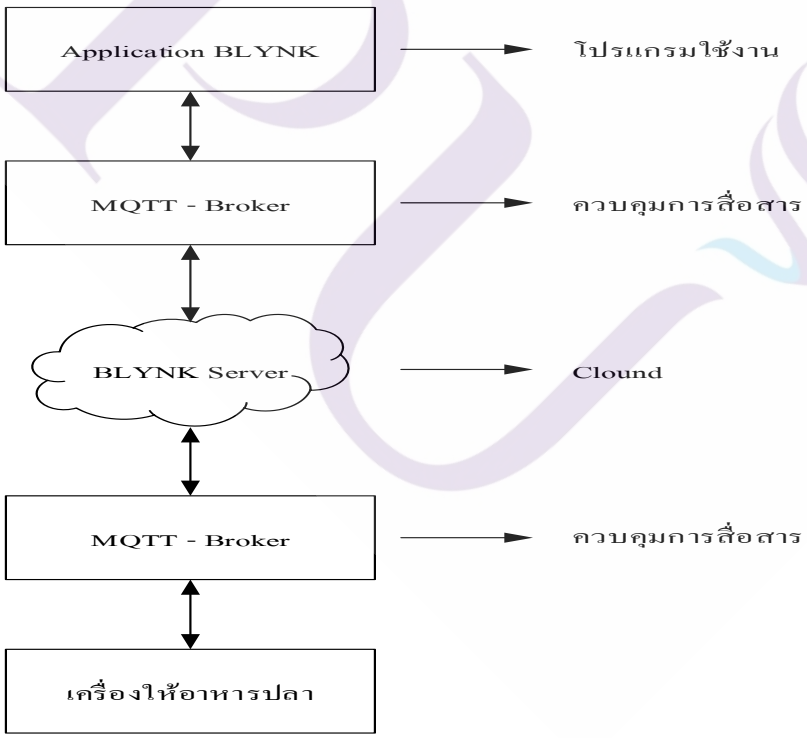
3.10 การออกแบบระบบรับส่งข้อมูลแอพพลิเคชั่น Blynk

ในการออกแบบผู้วิจัยได้ออกแบบให้ Blynk Server เป็นตัวกลางในการรับส่งข้อมูล โดยทำหน้าที่รับข้อมูลจากตัว Sensor Node แล้วส่งต่อไปยังที่ Blynk Server ต่อไป โดยระบบได้ถูกออกแบบมาให้ ตัว Sensor Node ทำการส่งข้อมูลมาเป็นระยะๆ ข้อมูลจะถูกส่งไปเก็บเอาไว้ที่ตัว Gateway แล้วจะถูกส่งต่อไปยัง Blynk Server ในส่วนฝั่งรับค่า ผู้วิจัยออกแบบให้รอรับค่าอยู่ตลอดเวลา เมื่อมีค่าร้องขอ ที่ถูกส่งเข้ามาใหม่ ระบบจะส่งไปยังตัว Sensor Node โดยทันทีแล้วจะแจ้งสถานการณ์ปรับแต่งค่ากลับไปยัง ระบบ Blynk Server อีกครั้งหนึ่ง เพื่อเป็นการยืนยันการรับการค่าสำเร็จโดยที่แอพพลิเคชั่น Blynk จะทำงานในการรับส่งข้อมูลผ่าน Potocal MQTT

เป็นตัวการในการสื่อสารของข้อมูลโดย Potocal MQTT จะมีรหัสTokenในการเข้าถึงข้อมูลเพื่อให้ความปลอดภัยในการเข้าถึงข้อมูลมากยิ่งขึ้น



ภาพที่ 3.8 ระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Blynk



ภาพที่ 3.9 ระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Blynk (ต่อ)

3.11 การออกแบบ User Interface แอพพลิเคชั่น Blynk

ในส่วนการออกแบบ User Interface ผู้วิจัยได้ออกแบบไว้เพื่อวัตถุประสงค์ให้ผู้ใช้งานได้ใช้งานโดยง่ายและสะดวกต่อการใช้งาน ผู้วิจัยจึงได้ออกแบบในรูปแบบแอพพลิเคชั่น Blynk ที่สามารถเข้าถึงได้ง่ายจากโทรศัพท์เคลื่อนที่และสามารถติดตามและสั่งการได้โดยง่ายโดยมีความสามารถสั่งการได้ดังนี้คือ

- ตั้งค่าเวลาในการให้อาหารปลาทองในตามช่วงเวลา
- ตั้งค่าแสดงจำนวนปริมาณการให้อาหารปลา
- ตั้งค่าวัดระดับความขุ่นของน้ำในตู้ปลา
- ตั้งค่าในการให้อาหารปลาทองได้อัตโนมัติ

ภาพที่ 3.10 แสดงข้อมูลการทำงานของ User Interface บนหน้าแอพพลิเคชั่น Blynk

3.12 การออกแบบในส่วนกำหนดค่าตัวแปรแอพพลิเคชั่น Blynk

ในการออกแบบส่วนของการเขียนโปรแกรมแอพพลิเคชั่น Blynk นั้นหลักจากที่ผู้วิจัยได้ทำการออกแบบในส่วนหน้า User Interface แล้วในส่วนการสั่งการใช้งานของแอพพลิเคชั่น ผู้วิจัยได้ทำการเขียนโปรแกรมขึ้นมาในฝั่งการรับส่งข้อมูลของตัวบอร์ด WemosD1 โดยการเขียนโปรแกรมของแอพพลิเคชั่น Blynk นั้น จะทำการเขียนโปรแกรมฝั่งเดียวคือฝั่งบอร์ด WemosD1 ให้รับส่งค่าและรันค่าต่างๆไปเก็บไว้บน Cloud ของ Blynk Server โดยการที่จะ Run code โปรแกรมของ Blynk นั้น จะต้องกำหนดค่าตัวแปรต่างๆในแอพพลิเคชั่น Blynk ตามที่ผู้วิจัยกำหนดและในการสร้างตัวแปรต่างๆนั้นแอพพลิเคชั่น Blynk จะมีค่าใช้จ่ายในการสร้างตัวแปรแต่ละตัว โดยผู้ออกแบบจะต้องซื้อ ENNRGY BALANCE.ในการที่จะซื้อปุ่มหรือฟีเจอร์ในการสร้างตัวแปรต่างๆ ตามตารางที่ 3.2

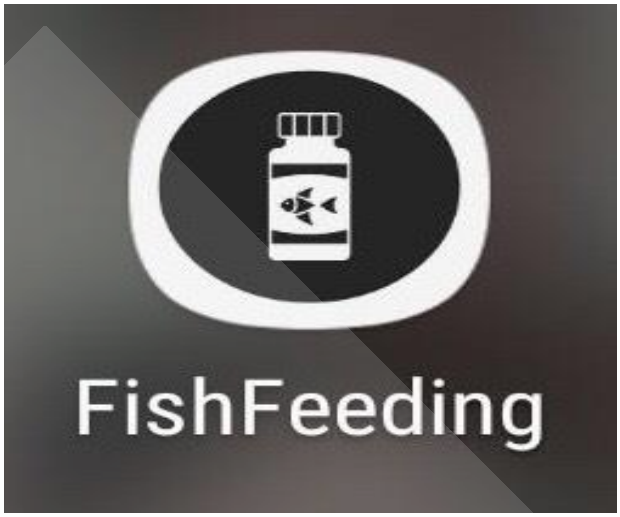
ตารางที่ 3.2 ตารางแสดงค่าตัวแปรและฟังก์ชั่นการทำงานของตัวตัวแปรต่างๆของ Blynk

ค่าตัวแปร	ฟังชั่นการทำงาน
V0	ช่วงเวลาการให้อาหารปลาช่วงที่1
V1	ช่วงเวลาการให้อาหารปลาช่วงที่2
V2	ช่วงเวลาการให้อาหารปลาที่3
V3	ปริมาณการให้อาหารปลาช่วงเวลาที่1
V4	ปริมาณการให้อาหารปลาช่วงเวลาที่2
V5	ปริมาณการให้อาหารปลาช่วงเวลาที่3
V6	การให้อาหารปลาอัตโนมัติ1ครั้ง
V7	กำหนดตั่งค่าช่วงเวลาที่จะให้อาหารปลา
V8	กำหนดตั่งค่าตั่งค่าชั่วโมงที่จะให้อาหารปลา
V9	กำหนดตั่งค่าตั่งค่านาทีที่จะให้อาหารปลา
V10	กำหนดตั่งค่าปริมาณอาหารปลาในแต่ละช่วงเวลา
V11	ปุ่มบันทึกการตั้งค่า
V12	แสดงวัน/เดือน/ปี/เวลา/ปัจจุบัน
V13	อ่านค่าความขุ่นของน้ำในตู้ปลา
V14	กำหนดตั้งค่าความขุ่นของน้ำในตู้ปลาที่กำหนดไว้
V15	อ่านค่าความขุ่นของน้ำในตู้ปลาที่กำหนดไว้

หลังจากผู้วิจัยได้กำหนดค่าตัวแปรต่างๆในแอพพลิเคชั่น Blynk แล้วผู้วิจัยก็จะดำเนินการเขียนโปรแกรมในการสั่งการบนแอพพลิเคชั่น Blynk กับตัวบอร์ด WemosD1 โดยการเขียนโปรแกรมนั้นจะมีขั้นตอนการทำงานได้ดั่งต่อไปนี้ ผู้วิจัยได้สร้างโปรเจ็คในแอพพลิเคชั่น Blynk แล้ว ผู้วิจัยจะนำ Token ที่ Blynk Server ได้ส่งมาใน E-mail แล้ว ผู้วิจัยจะนำไปใส่ในส่วนของโปรแกรมเพื่อที่จะให้ตัวบอร์ด WemosD1 ทำการรันโปรแกรมที่จะเชื่อมต่อกับแอพพลิเคชั่น Blynk หลังจากนั้นผู้วิจัยจะเรียกไฟล์ค่าฟังชั่นการทำงานส่วนต่างๆดั่งรูปที่ผู้วิจัยได้ใส่ไว้และในการเชื่อมต่อ WiFi นั้นและจะใช้ตัว ESP8266 ในการเชื่อมต่อ Wifi และจะใช้ตัว EEPROM ทำงานส่วนในส่วนของการอ่านค่าและเก็บข้อมูลการเข้ารหัส Wifi และการเก็บค่าเวลาต่างๆที่ผู้วิจัยได้ตั้งค่าเวลาไว้และในส่วนของตัว RTC นั้นจะเป็นตัวเทียบเวลาและตัวคอยเช็คเวลาของข้อมูลในฟังชั่นเวลาในการเชื่อมต่อเวลาเข้ากับตัวตัวบอร์ด WemosD1 และแอพพลิเคชั่น Blynk และในส่วนการรับส่งข้อมูลผู้วิจัยได้ออกแบบในการรับส่งข้อมูลโดยใช้ Potocal UDP กับ NTP ในการรับส่งข้อมูลการสั่งการต่างๆของแอพพลิเคชั่น Blynk ต่อไป

3.13 การออกแบบแอพพลิเคชั่น Inventor 2

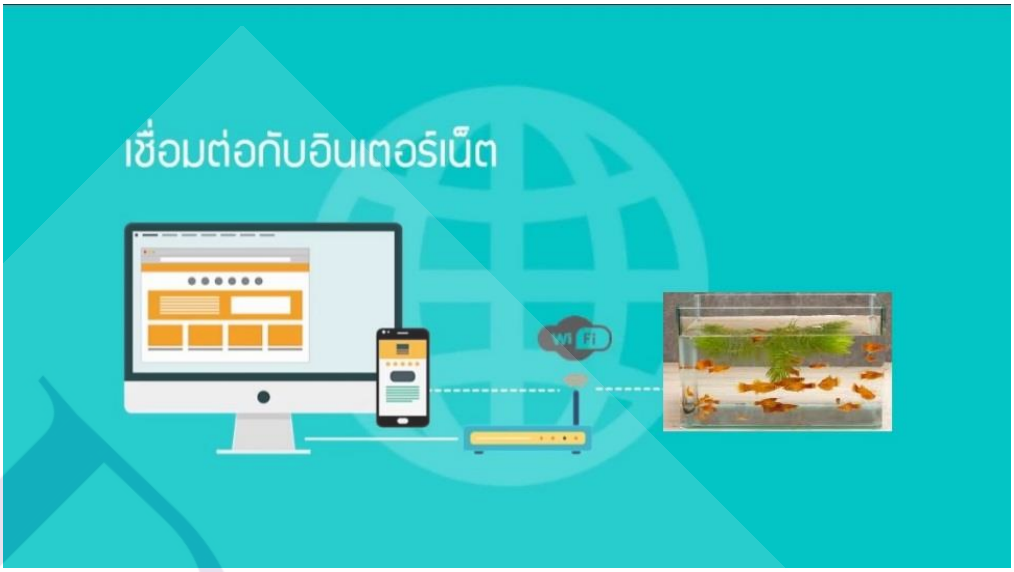
ในส่วนการออกแบบแอพพลิเคชั่น Inventor 2 ผู้วิจัยเลือกใช้แอพพลิเคชั่น Inventor 2 เพราะว่าเป็นแอพพลิเคชั่นที่สามารถออกแบบได้อย่างอิสระและไม่มีค่าใช้จ่ายใดๆและการเขียนออกแบบ แอพพลิเคชั่น Inventor 2 นั้น โปรแกรมจะถูกออกแบบให้เขียนโปรแกรมผ่านทางหน้าเว็บไซร์โดยจะไม่มีตัวติดตั้งไฟล์โปรแกรมเหมือนโปรแกรมทั่วๆไปในการเข้าไปเขียนโปรแกรมแอพพลิเคชั่น Inventor 2 นั้นจะต้องเข้าผ่านทาง Google Chrome และเข้าไปยัง www.google.com แล้วพิมพ์ค้นหาคำว่า แอพพลิเคชั่น Inventor 2 จากนั้นเข้าไปในเว็บแอพพลิเคชั่น Inventor 2 ในการจะเขียนนั้นต้องมี Gmail ในการ Login เข้าไปเขียน Application Inventor 2 หลังจาก Login แล้วในการเขียนแอพพลิเคชั่น Inventor 2 นั้น สิ่งสำคัญอีกอย่างคือ Internet เพราะจะต้องใช้ในการเชื่อมต่อในการออกแบบแอพพลิเคชั่นและในการ Login เข้าแอพพลิเคชั่น Inventor 2 ทุกครั้งแแอพพลิเคชั่น Inventor 2 ที่ถูกเขียนขึ้นหรืออัพเดทนั้น ก็จะถูกบันทึกเก็บไว้บน Cloud ของแอพพลิเคชั่น Inventor 2 ด้วย ในส่วนของหน้าของการออกแบบนั้นแบ่งได้เป็น2ส่วนคือ ส่วนที่1 คือหน้า Designer เป็นหน้าที่ไว้สำหรับสามรถออกแบบไอค่อนต่างๆบน UserInterface และส่วนที่2 คือหน้า Block เป็นหน้าที่ไว้สำหรับเขียน Codeโปรแกรมในการสั่งการแอพพลิเคชั่นในการให้อาหารปลาหรือการแจ้งเตือนตั้งค่าความขุ่นของน้ำภายในตู้ปลา



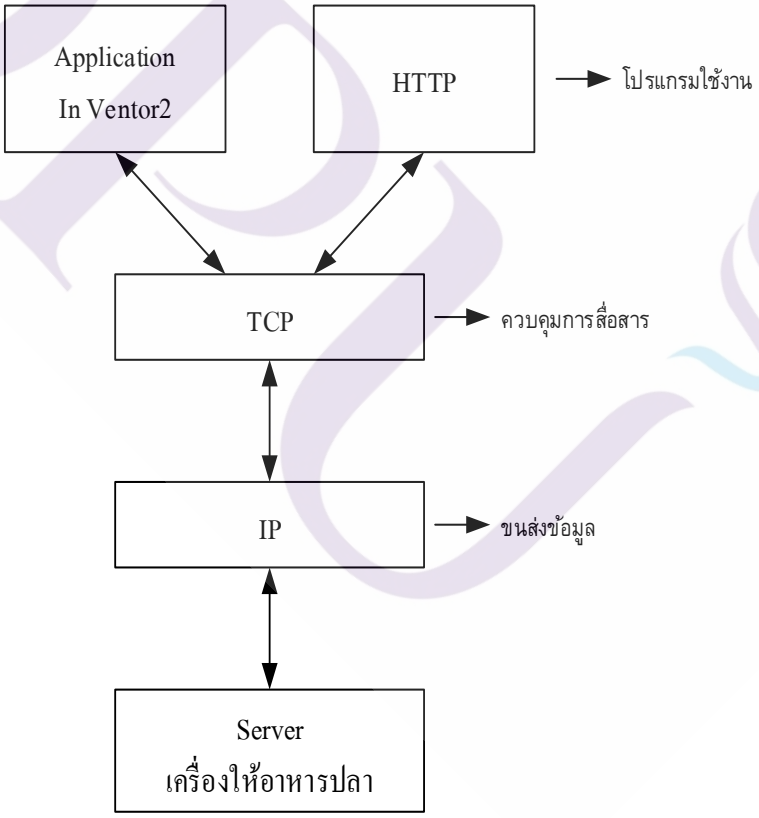
ภาพที่ 3.11 แสดงแอพพลิเคชั่น Inventor 2

3.14 การออกแบบระบบรับส่งข้อมูล Application Inventor 2

ระบบการรับส่งข้อมูลนั้นแอพพลิเคชั่น Inventor 2 นั้นจะต่างกับแอพพลิเคชั่น Blynk ตรงที่จะไม่มี Server แบบสำเร็จรูปและ Cloud ของตัวเอง จึงจำเป็นต้องสร้าง Server ขึ้นมาเอง และเขียนแอพพลิเคชั่นให้เชื่อมต่อและคำสั่งต่างๆขึ้นมาเอง จึงอาจจะมองว่ายุ่งยากกว่าแอพพลิเคชั่น Blynk ที่มี Server สำเร็จรูปและระบบทั้ง 2 ฝั่งเป็น Client จึงทำแอพพลิเคชั่น Blynk มองว่าใช้งานและสร้างขึ้นง่ายกว่า จึงทำให้แอพพลิเคชั่นInventor 2 และ แอพพลิเคชั่น Inventor ยังมีข้อจำกัดการใช้งานอีกด้วยในการใช้งานในที่ต่างๆ เพราะต้อง ให้ระบบ Network ที่เชื่อมต่อสามารถมองเห็น Network จากภายนอกด้วยกันได้ ถ้าจะไปใช้ตามหาลัยหรือสถานที่สำคัญที่เป็นระบบ Network ที่ค่อนข้าง Private หรือมี Security สูงก็จะไม่สามารถใช้งานได้เพราะจำเป็นต้องเข้าไป FixIp ของ vRouter เพื่อที่จะให้ Network มองเห็นภายนอกได้ แต่ข้อดีของ Server Application Inventor 2 ที่เขียนขึ้นมาเอง จะมีข้อดีกว่าแอพพลิเคชั่น Blynk ตรงที่เวลาระบบขัดข้องสารถตรวจสอบแก้ไขได้ด้วยตนเองได้เลยเพราะเป็นระบบ Server และ แอพพลิเคชั่นที่ผู้วิจัยออกแบบเขียนขึ้นมาเองทุกส่วนจึงง่ายต่อการดูแลตรวจสอบและพัฒนาต่อยอดได้อีก



ภาพที่ 3.12 แสดงระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Inventor 2

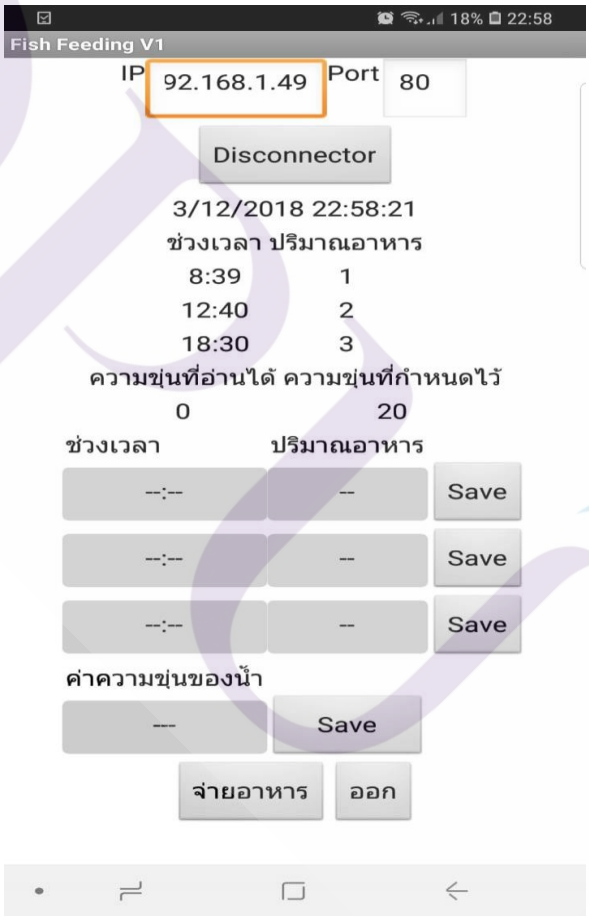


ภาพที่ 3.13 แสดงระบบรับ-ส่งข้อมูลของแอพพลิเคชั่น Inventor 2 (ต่อ)

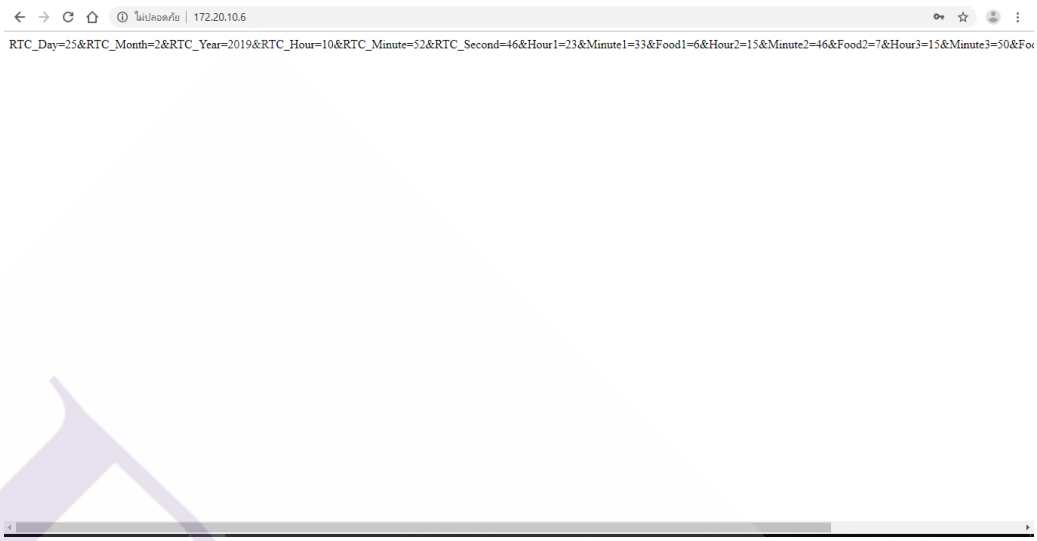
3.15 การออกแบบ User Interface แอพพลิเคชั่น Inventor 2

การออกแบบส่วน User Interface ได้ออกแบบไว้เพื่อวัตถุประสงค์ให้ใช้งานได้ง่ายและสะดวกจึงได้ออกแบบในรูปแบบ Application ที่สามารถเข้าถึงได้ง่ายจากทั้งโทรศัพท์เคลื่อนที่และคอมพิวเตอร์ผ่านหน้า Web browser โดยเน้นใช้งานที่โทรศัพท์เคลื่อนที่เป็นสำคัญสามารถติดตามและสั่งการได้โดยง่าย โดยมีความสามารถสั่งการได้ดังนี้คือ

- ตั้งค่าวลาในการให้อาหารปลาทองในตามช่วงเวลา
- ตั้งค่าแสดงจำนวนปริมาณการให้อาหารปลา
- ตั้งค่าวัดระดับความขุ่นของน้ำในตู้ปลา
- ตั้งค่าวลาในการให้อาหารปลาทองได้อัตโนมัติ



ภาพที่ 3.14 แสดงข้อมูลการทำงานของ User Interface บนแอพพลิเคชั่น Inventor 2



ภาพที่ 3.15 แสดงข้อมูลการทำงานบนหน้า Webserver

3.16 การออกแบบในส่วนการเขียนกำหนดค่าตัวแปรแอพพลิเคชั่น Inventor 2

ในส่วนการเขียนโปนแกรมแอพพลิเคชั่น Inventor 2 นั้น ผู้วิจัยได้ออกแบบหน้า User interface ของแอพพลิเคชั่น Inventor 2 ใกล้เคียงกับแอพพลิเคชั่น Blynk แต่ในส่วนของการเขียนโปรแกรมนั้นแอพพลิเคชั่น Inventor 2 ต้องเขียนโปรแกรมทั้ง 2 ฝั่งทั้งฝั่งแอพพลิเคชั่นและฝั่งบรอด์ WemosD1 เพราะว่าตัวกล่องเครื่องให้อาหารปลาจะเป็น Serverด้วยตนเอง ดั่งนั้นการรับส่งข้อมูลจะไม่อยู่บนCloud ทำให้การเขียนโปรแกรมจึงจะค่อนข้างเยอะและยุ่งยากซับซ้อนกว่าการเขียนโปรแกรมแอพพลิเคชั่น Blynk ตรงที่จะต้องเขียน Server และการรับส่งข้อมูลและฟังชั่นต่างๆด้วยตนเองทั้งหมด และในส่วนของการเขียนโปรแกรมฝั่งแอพพลิเคชั่น Inventor 2 จะเป็นภาษาโลโก้ซึ่งในส่วนของการเขียนโปรแกรมนั้นจะไม่ซับซ้อนมากนักซึ่งจะเป็นการลากฟังชั่นการทำงานของส่วนต่างๆและค่าตัวแปรต่างๆที่ผู้วิจัยได้ออกแบบไว้มาวางต่อกันดังตารางที่ 3.3

ตารางที่ 3.3 ตารางแสดงค่าตัวแปรและฟังชั่นการทำงานของของแอพพลิเคชั่น Inventor 2

ค่าตัวแปร	ฟังก์ชั่นการทำงาน
Screen	หน้าจอแอพพลิเคชั่น
Horizonal Arrangment1	ส่วนการทำงาน IP,Port
Label1	หัวข้อ IP
Label2	หัวข้อ Port
IPtexBox	ใส่ IP
PorttexBox	ใส่ Port
Connect	ปุ่ม Connect
LabelTimeDate	แสดงวัน/เดือน/ปี
Table Arrangment 4	ส่วนการแสดงทำงานกำหนดเวลาและปริมาณการให้อาหารปลาในแต่ละช่วงเวลา
Label_Section 1	กำหนดค่าเวลาการให้อาหารปลาช่วงเวลาที่1
Label_Section 2	กำหนดค่าเวลาการให้อาหารปลาช่วงเวลาที่2
Label_Section 3	กำหนดค่าเวลาการให้อาหารปลาช่วงเวลาที่3
Label_Food 1	กำหนดค่าปริมาณการให้อาหารปลาช่วงเวลาที่1
Label_Food 2	กำหนดค่าปริมาณการให้อาหารปลาช่วงเวลาที่2
Label_Food 3	กำหนดค่าปริมาณการให้อาหารปลาช่วงเวลาที่3
Label 17	หัวข้อ ช่วงเวลา
Label 18	หัวข้อ ปริมาณอาหาร
Label19	หัวข้อความขุ่นที่อ่านได้
Label20	หัวข้อข้อความขุ่นที่กำหนดไว้
Label_Water	ตั้งค่าความขุ่นที่อ่านได้
Label_SetWater	ตั่งค่าความขุ่นที่กำหนดไว้
Tabel Arrangment2	ส่วนตั่งค่าทำงานกำหนดเวลาและปริมาณการให้อาหารปลาในแต่ละช่วงเวลา
Button Save 1	บันทึกข้อมูลการให้อาหารปลาช่วงเวลาที่1
Button Save 2	บันทึกข้อมูลการให้อาหารปลาช่วงเวลาที่2

Button Save 3	บันทึกข้อมูลการให้อาหารปลาช่วงเวลาที่3
Button Save 4	บันทึกข้อมูลการค่าความขุ่นของน้ำในตู้ปลาที่กำหนดไว้
TimePickerFood 1	เวลาการให้อาหารปลาช่วงเวลาที่1
TimePickerFood 2	เวลาการให้อาหารปลาช่วงเวลาที่2
TimePickerFood 3	เวลาการให้อาหารปลาช่วงเวลาที่3
Label21	หัวข้อช่วงเวลาการให้อาหารปลา
ListPickerFood 1	ปริมาณการให้อาหารปลาช่วงเวลาที่1
ListPickerFood 2	ปริมาณการให้อาหารปลาช่วงเวลาที่2
ListPickerFood 3	ปริมาณการให้อาหารปลาช่วงเวลาที่3
ListPickerWater	ค่าความขุ่นของน้ำในตู้ปลาที่กำหนดไว้
ButtonPay	ปุ่มการให้อาหารปลาอัตโนมัติ1ครั้ง
ExitButton	ปุ่มออก
Notifier 1	แจ้งเตือนน้ำในตู้ปลาขุ่นเกินกว่ากำหนด

บทที่ 4
ผลการทดลอง

งานวิจัยนี้ผู้วิจัยได้นำแอพพลิเคชั่นสำเร็จรูประหว่างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 มาเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานในด้านต่างๆโดยจะทดสอบผ่านการทดสอบผ่านทางหน้า User interfacec และฟีเจอร์ต่างๆและฟังชั่นและเงื่อนไขของแอพพลิเคชั่นโดยจะจะทำการทดสอบการใช้งานกับเครื่องให้อาหารปลาโดยผลการทดสอบจะอยู่ในตารางที่ผู้วิจัยทำขึ้นมาเพื่อให้เห็นการเปรียบเทียบข้อแตกต่างในด้านต่างๆแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 ว่าแตกต่างกันอย่างไรบ้างในด้านประสิทธิภาพด้านต่างๆ เช่น ด้านเวลา ด้านค่าใช้จ่ายและสามรถนำไปใช้งานจริงได้ถูกต้องแม่นยำตามวัตถุประสงค์ของผู้ใช้งาน

4.1 ผลการทดสอบการการเปรียบเทียบประสิทธิภาพการออกแบบ User Interface และ ฟีเจอร์และ ฟังชั่นเงื่อนไขต่างๆของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor 2

ผลการทดสอบจะเห็นได้ว่าแอพพลิเคชั่น Blynk นั้นจะสามารถออกแบบในส่วนฟีเจอร์และในส่วนหน้า User Interface นั้นจะออกแบบได้ไม่เยอะเหมือนแอพพลิเคชั่น Inventor2 แต่ในส่วนระบบการทำงานด้านการเชื่อมต่ออินเตอร์เน็ตและการใช้งานนั้นแอพพลิเคชั่น Blynk จะสามารถใช้งานได้หลายระบบจะแตกต่างจาก แอพพลิเคชั่น Inventor2 ตรงที่สามรถใช้งานได้แค่ระบบ Android แต่ในส่วนฟังชั่นต่างๆการทดสอบทั้งสองแอพพลิเคชั่นจะไม่ต่างอะไรกันมากจะแตกต่างตรงที่ ว่าแอพพลิเคชั่น Inventor2 ค่อนข้างจะฟังชั่นที่ผู้ใช้สามรถแก้ไขออกแบบได้ด้วยตัวเองแต่แอพพลิเคชั่น Blynk จะเป็นแอพพลิเคชั่นที่ค่อนข้างจะสำเร็จรูปจะไม่สามรถใส่ฟังชั่นหรือออกแบบอะไรตามความชอบได้มาก ดังตารางที่ผู้วิจัยได้เปรียบเทียบดังตารางที่ 4.1

ตารางที่ 4.1 ตารางการเปรียบเทียบประสิทธิภาพการออกแบบหน้า User Interface และฟีเจอร์และฟังชั่นเงื่อนไขต่างๆ ของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2

การเปรียบเทียบประสิทธิภาพการออกแบบหน้าฟีเจอร์และฟังชั่นเงื่อนไขต่างๆ	**แอพพลิเคชั่น Blynk**	**แอพพลิเคชั่น Inventor2**
สามรถออกแบบในส่วนUser Interface	✓	✓
สามารถออกแบบในส่วนอักษร,สี,ภาพ,เสียง	✕	✓
สามารถลิ้งค์ข้อมูลเว็บไซต์	✓	✓
สามารถแสดงข้อมูลบนเว็บบราวเซอร์	✕	✓
สามารถแจ้งเตือนผ่านแอพพลิเคชั่น	✓	✓
สามรถเก็บข้อมูลไว้บนCloud	✓	✓
สามรถดึงไฟล์ข้อมูลเข้าไปบนแอพพลิเคชั่น	✕	✓
สามรถแชร์แอพพลิเคชั่นผ่าน QR Code	✓	✓
สามรถจำลองสั่งการผ่านคอมพิวเตอร์	✕	✓
สามรถLogin ผ่าน E-Mail	✓	✓
สามรถLogin Facebook	✓	✕
สามรถเลือกภาษาบนแอพพลิเคชั่น	✓	✓
สามรถใช้งานผ่านระบบAndroid V.2.3 ขึ้นไป	✓	✓
สามรถใช้งานผ่านระบบ IOS V.6 ขึ้นไป	✓	✕

4.2 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายในการออกแบบแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2

ผลการทดสอบจะเห็นได้ว่าแอพพลิเคชั่น Blynk จะมีค่าใช้จ่ายในส่วนการออกแบบการสร้างแอพพลิเคชั่นโดยผู้ใช้ต้องซื้อ ENEGRY BALANCE เพื่อที่จะสร้างไอค่อนและในกรณีผู้อื่นจะมาดาวน์โหลดแอพพลิเคชั่นเพื่อจะนำไปใช้ต่อนั้นก็ต้องมีค่าใช้จ่ายในการซื้อ ENEGRY BALANCE เหมือนกันถึงจะดาวน์โหลดแอพพลิเคชั่นไปใช้งานต่อไปได้ จะแตกต่าง กับแอพพลิเคชั่น Inventor2 จะไม่มีค่าใช้จ่ายในการใช้งานใดๆทั้งสิ้น แต่แอพพลิเคชั่น Inventor2 จะมีข้อจำกัดตรงที่ใช้งานได้แค่ระบบAndroid อย่างเดียวดังตารางที่ 4.2

ตารางที่ 4.2 ตารางการเปรียบเทียบประสิทธิภาพด้านค่าใช้จ่ายในการออกแบบและการนำไปใช้งานแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2

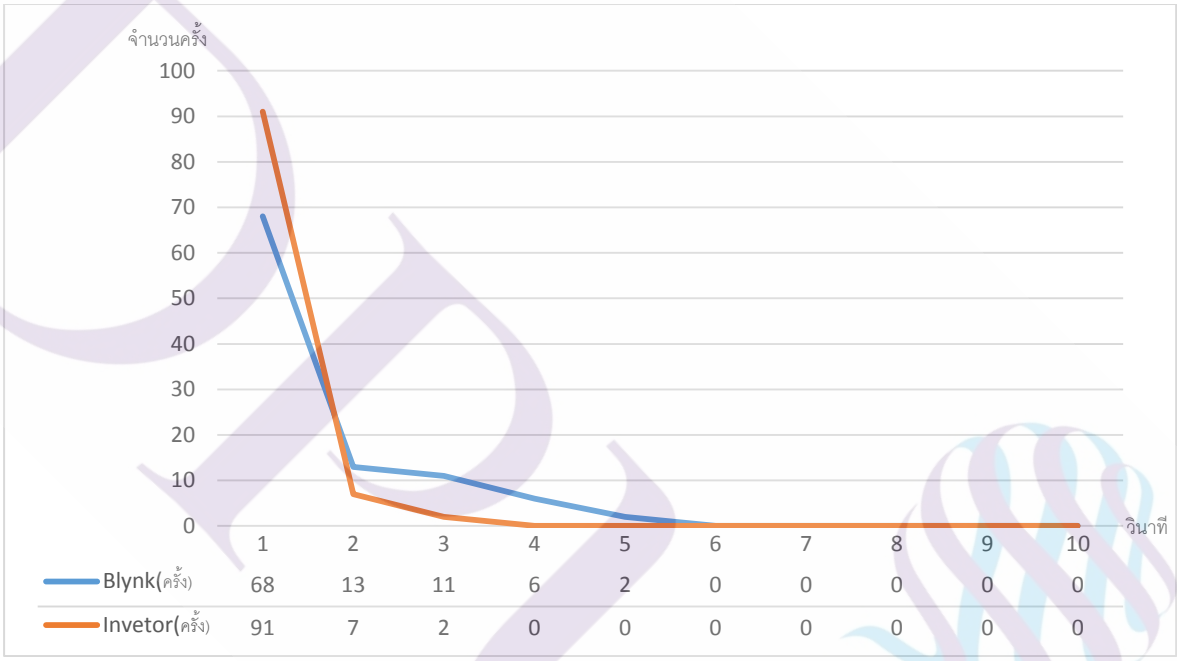
เปรียบเทียบประสิทธิภาพด้านงบประมาณค่าใช้จ่ายในการออกแบบและการนำไปใช้งาน	**แอพพลิเคชั่น Blynk**	**แอพพลิเคชั่น Inventor2**
ค่าใช้จ่ายในการออกแบบในส่วน User Interface	✓	×
ค่าใช้จ่ายในการดาวน์โหลดในส่วนการนำไปใช้งาน	✓	×

4.3 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านเวลาในการใช้งานของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2

ผลการทดสอบการประสิทธิภาพด้านเวลาในการใช้งานของแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2 นั้นจะมีปัจจัยในการสั่งการใช้งานขึ้นอยู่กับอินเตอร์เน็ตที่ใช้ในการเชื่อมต่อในการรับส่งข้อมูล กรณีถ้าอินเตอร์เน็ตที่เชื่อมต่อนั้นมีความเสถียรปกติแอพพลิเคชั่นนั้นจะสามารถสั่งการแอพพลิเคชั่นได้ทันทีแต่ถ้าการเชื่อมต่ออินเตอร์เน็ตเกิดขัดข้องไม่ว่าจะเป็นไฟฟ้าดับหรือระบบอินเตอร์เน็ตขัดข้อง ก็อาจจะจะไม่สามรถสั่งการแอพพลิเคชั่นได้ทันทีอาจจะเกิดการหน่วงด้านเวลาบ้างโดยจากผลการทดลองผู้วิจัยสั่งการทำการเชื่อมต่ออินเตอร์เน็ตทั้งสองแอพพลิเคชั่นแบบเดียวโดยจะจับเวลาในการสั่งการของแต่แอพพลิเคชั่นทั้งสอง แอพพลิเคชั่นโดยจะจับเวลาในการสั่งการ100ครั้งจะเห็นได้ว่าแอพพลิเคชั่น Blynk นั้นจะเกิดการหน่วงในการ

รับส่งข้อมูลมากกว่าแอพพลิเคชั่น Inventor2 เพราะว่าแอพพลิเคชั่น Blynk นั้น Server อยู่บน Cloud ทำให้เวลาในการเดินทางข้อมูลอาจจะช้ากว่าแอพพลิเคชั่น Inventor2 โดยผู้วิจัยได้ทำการทดสอบในการจับหน่วงเวลาโดยการสั่งการผ่านเครื่องให้อาหารปลาเป็นจำนวน 100 ครั้ง ของเวลาทุกวินาที ดังตารางที่ 4.3

ตารางที่ 4.3 ตารางกราฟเปรียบเทียบประสิทธิภาพด้านเวลาในการใช้งานของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2



4.4 ผลการทดสอบการเปรียบเทียบประสิทธิภาพด้านการออกแบบการรับส่งข้อมูลของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2

ผลการทดสอบการประสิทธิภาพด้านเวลาในการใช้งานของแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2 นั้นจะมีปัจจัยในการสั่งการใช้งานขึ้นอยู่กับอินเตอร์เน็ตที่ใช้ในการเชื่อมต่อในการรับส่งข้อมูล กรณีถ้าอินเตอร์เน็ตที่เชื่อมต่อนั้นมีความเสถียรปกติแอพพลิเคชั่นนั้นจะสามารถสั่งการแอพพลิเคชั่นได้ทันทีแต่ถ้าการเชื่อมต่ออินเตอร์เน็ตเกิดขัดข้องไม่ว่าจะเป็นไฟฟ้าดับหรือระบบอินเตอร์เน็ตขัดข้อง ก็อาจจะจะไม่สามรถสั่งการแอพพลิเคชั่นได้ทันทีอาจจะเกิดการหน่วงด้านเวลาบ้างโดยจากผลการทดลองผู้วิจัยสั่งการทำการเชื่อมต่ออินเตอร์เน็ตทั้งสองแอพพลิเคชั่นแบบเดียวโดยจะจับเวลาในการสั่งการของแต่แอพพลิเคชั่นทั้งสองแอพพลิเคชั่น

โดยจะจับเวลาในการสั่งการ100ครั้งจะเห็นได้ว่าแอพพลิเคชั่น Blynk นั้นจะเกิดการหน่วงในการรับส่งข้อมูลมากกว่าแอพพลิเคชั่น Inventor2 เพราะว่าแอพพลิเคชั่น Blynk นั้นServer อยู่บนCloud ต่างประเทศ จึงทำให้เวลาในการเดินทางข้อมูลอาจจะช้ากว่าแอพพลิเคชั่น Inventor2 โดยผู้วิจัยได้ทำการทดสอบในการจับหน่วงเวลาโดยการสั่งการผ่านเครื่องให้อาหารปลาเป็นจำนาน 100 ครั้ง ของเวลาทุกวินาทีดังตารางที่ 4.4

ตารางที่ 4.4 ตารางการการเปรียบเทียบประสิทธิภาพด้านการออกแบบการรับส่งข้อมูลของแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2

เปรียบเทียบประสิทธิภาพด้านการออกแบบการรับส่งข้อมูล	**แอพพลิเคชั่น Blynk**	**แอพพลิเคชั่น Inventor2**
การใช้งานผ่านเครือข่าย Network Private IP	✓	✓
การใช้งานผ่านเครือข่าย Network Public IP	✓	✓

4.5 ผลการทดสอบประสิทธิภาพของอุปกรณ์ Hardware ในการควบคุมการทำงาน

ผลการทดสอบประสิทธิภาพในส่วนอุปกรณ์ Hardware ในการควบคุมการทำงานสามารถทำงานได้ดีโดยสามารถควบคุมสั่งการผ่านแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2ได้อย่างเหมาะสมเป็นไปตามเป้าหมายที่ตั้งไว้

4.6 ผลการทดสอบประสิทธิภาพการทำงานโดยรวมของทั้งระบบ

ผลการทดสอบประสิทธิภาพการทำงานโดยรวมของทั้งระบบ เมื่อระบบทั้งหมดถูกใช้งานจริง สามารถทำงานได้อย่างมีประสิทธิภาพแต่ระบบก็อาจมีปัญหาในกรณีที่ Network หรือ Wi-Fi ที่ทำการเชื่อมต่อกับ Internet ไม่ได้ แต่ยังสามารถกลับมาใช้งานได้ในภายหลัง ปัญหาเกิด Internet ทำให้ข้อมูลที่ต้องการจะส่งตามช่วงเวลา ไม่สามารถส่งข้อมูลได้ตามต้องการ แต่ยังสามารถกลับทำงานได้ตามข้อมูลที่สั่งการตั้งค่าไว้ กรณีที่ Wi-Fi internet กลับมาเป็นสภาวะปกติแล้ว ผู้ใช้งานสามารถใช้งานตามคำสั่งที่ตั้งค่าไว้ได้อีกครั้ง

บทที่ 5
สรุปผลการวิจัยและข้อเสนอแนะ

5.1 ผลการทดลองภาพรวมเปรียบเทียบประสิทธิภาพการทำงาน

ในงานวิจัยชิ้นนี้ผู้วิจัยได้ออกแบบและเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานระหว่างแอพพลิเคชั่นสำเร็จรูประหว่างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2 โดยประยุกต์ใช้เทคโนโลยี (IoT) เข้ามาช่วยเพื่อศึกษาและเพิ่มประสิทธิภาพในการใช้งานแอพพลิเคชั่นให้เป็นไปตามวัตถุประสงค์และขอบเขตของผู้วิจัยที่ได้ตั้งเป้าหมายไว้ให้บรรลุวัตถุประสงค์

5.2 การบรรลุวัตถุประสงค์

1. แอพพลิเคชั่น Blynk มีไฟล์ Library สำหรับอุปกรณ์ IoT ทำให้ช่วยลดขั้นตอนการเขียนโปรแกรม สามารถนำคำสั่งไปใช้งานในการรับส่งข้อมูลได้เลย แต่ในส่วนของแอพพลิเคชั่น Inventor2จะไม่มี Library ให้สำหรับอุปกรณ์ IOT จึงต้องเขียนโปรแกรมServerในการรับส่งข้อมูลระหว่างอุปกรณ์กับแอพพลิเคชั่นขึ้นมาเอง

2. แอพพลิเคชั่นมี Component ที่สามารถใช้งานได้สะดวก เพียงกำหนดค่า Parameter ให้ถูกต้องก็สามารถเชื่อมต่อกับอุปกรณ์ IOT ได้เลย ส่วนแอพพลิเคชั่น Inventor2 ในการสั่งงาน Component จะต้องเขียนโค้ดโปรแกรมในการสั่งการขึ้นมาเอง

3. ระบบที่ใช้แอพพลิเคชั่น Blynk อุปกรณ์ทั้ง 2 ฝั่ง จะเป็น Client ทำให้ง่ายต่อการใช้งาน สามารถนำเครื่องไปใช้งานที่ใดก็ได้ ส่วนแอพพลิเคชั่น Inventor2 ตัวเครื่องให้อาหารปลาจะทำหน้าที่เป็น Server ซึ่งการนำไปใช้งานจะต้องเช็คค่า Network IP ให้ตัวกล่องสามารถมองเห็น Network จากภายนอกได้ ซึ่งมีข้อจำกัดในการนำไปใช้งาน

4. แอพพลิเคชั่น Blynk สามารถใช้งานระบบโทรศัพท์เคลื่อนที่ได้ทั้ง2 ระบบ Android และIOS แต่ แอพพลิเคชั่นInventor 2 สามารถใช้ได้แค่ ระบบเดียวคือ ระบบ Android แต่สามารถจำลองผ่านคอมพิวเตอร์ได้

5. แอพพลิเคชั่น Blynk จะมีข้อเสียคือ เป็น Server ที่อยู่บน Cloud แบบสำเร็จรูปซึ่งเวลา Cloud หรือServer มีปัญหาขัดข้องจะไม่สามารถตรวจสอบหรือแก้ไขอะไรได้เลยจะต่างกับแอพพลิเคชั่น Inventor2 ที่มีServer เป็นของตัวเองซึ่งสามารถตรวจสอบแก้ไขได้ในกรณีเวลาที่ Server หรือระบบมีปัญหา

6. แอพพลิเคชั่นมี Blynk จะมีค่าใช้จ่ายในการซื้อ ENERGY BALANCE เพื่อสร้างไอค่อนต่างๆแต่ในส่วนแอพพลิเคชั่น Inventor2จะ ไม่มีค่าใช้จ่ายใดๆทั้งสิ้นในการสร้างไอค่อนแอพพลิเคชั่น

5.3 องค์ความรู้สำคัญ

ในงานวิจัยนี้ได้นำเสนอองค์ความรู้เกี่ยวกับการออกแบบและเปรียบเทียบประสิทธิภาพการทำงานและการใช้งานระหว่างแอพพลิเคชั่น Blynk กับ แอพพลิเคชั่น Inventor2โดยประยุกต์ใช้เทคโนโลยี (IoT) เข้ามาช่วยเพื่อศึกษาประสิทธิภาพในการใช้งานของแอพพลิเคชั่นให้เป็นไปตามวัตถุประสงค์และขอบเขตของผู้วิจัยที่ได้ตั้งเป้าหมายไว้ให้บรรลุวัตถุประสงค์

5.4 ข้อเสนอแนะแนวทางการพัฒนาต่อ

1. สามารถนำผลการทดลองนี้ไปเปรียบเทียบประสิทธิภาพในการทำงานและการใช้งานระหว่างแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2ไปเปรียบเทียบประสิทธิภาพกับแอพพลิเคชั่นอื่นๆได้

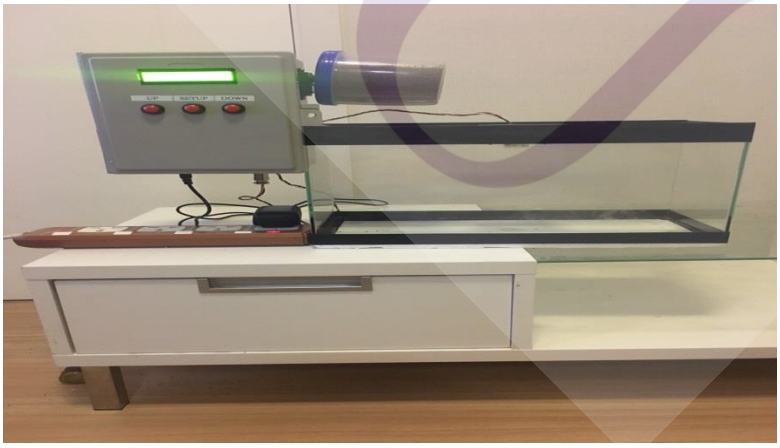
2. สามารถนำความรู้เกี่ยวกับ แอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2ไปสร้างและพัฒนาต่อยอดโปรเจ็คอื่นๆได้ตามวัตถุประสงค์ที่ต้องการ

3. สามารถสามรถนำความรู้เกี่ยวกับแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2 นำไปพัฒนาต่อยอดเชิงธุรกิจต่อไปได้

4. สามารถนำความรู้ที่ได้รับเกี่ยวกับอุปกรณ์ (IoT) นำไปพัฒนาต่อยอดเพื่อให้เกิดประสิทธิภาพต่อไปได้

5.5 สรุป

ในงานวิจัยนี้ได้เปรียบเทียบประสิทธิภาพกาทำงานและการใช้งานงานระหว่างแอพพลิเคชั่น Blynk กับแอพพลิเคชั่น Inventor2 นั้น จะมีข้อแตกต่างกันทั้งในส่วนการออกแบบแอพพลิเคชั่นและในส่วนการรับส่งข้อมูลการจากที่ผู้วิจัยได้ทำการเปรียบเทียบประสิทธิภาพในการทำงานและการใช้งานของทั้งสองแอพพลิเคชั่นนั้นจะสังเกตได้ว่าแอพพลิเคชั่น Blynk จะใช้งานและเชื่อมต่ออินเตอร์เน็ตสะดวกและง่ายกว่าแอพพลิเคชั่น Inventor2 เพราะมี Server สำเร็จรูปอยู่บน Cloud และในส่วนการเขียนโปรแกรมนั้นแอพพลิเคชั่น Blynk ยังช่วยลดขั้นตอนการเขียนโปรแกรมได้อีกด้วยเพราะมีไฟล์ Liberty ที่สำเร็จรูปจึงทำให้ลดภาระการทำงานในส่วนของ Server และลดภาระทรัพยากรของคอมพิวเตอร์ไปอีกด้วยซึ่งจะแตกต่างกับแอพพลิเคชั่น Inventor2ที่ต้องเขียนServerในการรับส่งข้อมูลขึ้นมาเองจึงมีข้อจำกัดในการเชื่อมต่ออินเตอร์เน็ตที่จะค่อนข้างยุ่งยากซับซ้อนกว่าแอพพลิเคชั่น Blynk แต่ในส่วนของการออกแบบหน้าตาและฟีเจอร์และฟังชั่นต่างๆนั้นแอพพลิเคชั่น Inventor2 จะสามรถออกแบบได้อิสระกว่าแอพพลิเคชั่น Blynk ทั้งในส่วนหน้าตาของแอพพลิเคชั่นและในส่วนการสั่งการต่างๆและในส่วนด้านค่าใช้จ่ายในการออกแบบนั้นแอพพลิเคชั่น Blynk จะมีค่าใช้จ่ายในการออกแบบแต่แอพพลิเคชั่น Inventor2 จะไม่มีค่าใช้จ่ายแต่จะสามารถใช้งานได้ในระบบAndroidอย่างเดียวแต่สามรถจำลองผ่านคอมพิวเตอร์ได้และในส่วนแอพพลิเคชั่น Blynk นั้น จะสามารถใช้งานได้ทั้งในระบบ IOS และในระบบ Android และเป็นที่นิยมแพร่หลายมากกว่าแอพพลิเคชั่น Inventor2 ดั้งนั้นในงานวิจัยนี้ผู้วิจัยหวังว่าการในการเปรียบเทียบประสิทธิภาพการทั้งสองแอพพลิเคชั่นนั้นจะเกิดประโยชน์สูงสุดต่อผู้พัฒนาและผู้ใช้งานต่อไป



ภาพที่ 5.1 แสดงเครื่องให้อาหารปลา

บรรณานุกรม

บรรณานุกรม

ภาษาไทย

จามจุรี กุลยอดและศิลป์ณรงค์ ฉวีพัฒน์. ได้วิจัย เรื่อง *ต้นแบบระบบควบคุมการเปิด-ปิดไฟผ่านแอพพลิเคชั่น บนระบบปฏิบัติการแอนดรอยด์ Wireless Light Switch Prototype with an Android Application.* สาขาวิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏกำแพงเพชร.

ฐิติพงษ์ รักษาริกรณ์. ได้วิจัยเรื่อง *แนวทางการประยุกต์ใช้ Internet of Things (IoT) กับ Smart Agriculture 4.0.* (วิทยานิพนธ์มหาบัณฑิต). สาขาวิศวกรรมข้อมูลขนาดใหญ่ วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์.

พิชิต บุญครอง และพิศณุ บุรมศรี. ได้วิจัยเรื่อง *การพัฒนาแอพพลิเคชั่นเพื่อการพยากรณ์อนุกรมเวลาโดยใช้ MIT App Inventor MIT App Inventor-Based Application for Time Series Forecasting.* สาขาวิชาสารสนเทศการลงทุน วิทยาลัยเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยรังสิต.

มหศักดิ์ เกตุฉ่ำ. ได้วิจัยเรื่อง *Internet of Things (IoT) ภาควิชาการจัดการเทคโนโลยีสารสนเทศ.* คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ.

สุจินต์ ณ นคร. ได้วิจัย เรื่อง *ระบบควบคุมผักสลัดอัจฉริยะ(Smart Salad Planting).* (วิทยานิพนธ์-มหาบัณฑิต). สาขาวิชาวิศวกรรมคอมพิวเตอร์และโทรคมนาคม วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์.

สุชาดา พลาชัยภิรมย์ศิล. ได้วิจัยเรื่อง *แนวโน้มการใช้แอพพลิเคชั่น* . (วิทยานิพนธ์มหาบัณฑิต).

สารสนเทศจากสื่ออิเล็กทรอนิกส์

การศึกษาค้นคว้าแอพพลิเคชั่น Blynk. สืบค้นเมื่อ 22 กรกฎาคม 2561 ,จาก https://bit.ly/2FXASEV

การศึกษาค้นคว้า แอพพลิเคชั่น Inventor 2.สืบค้นเมื่อ 22 กรกฎาคม 2561 https://bit.ly/2rpX0PS

การศึกษาค้นคว้าแอพพลิเคชั่น Inventor 2ส่วนออกแบบ,ส่วนเขียนโค้ด.สืบค้นเมื่อ 22 กรกฎาคม 2561 จาก https://bit.ly/2CK1w3L

การศึกษาค้นคว้าแอพลิเคชั่น NETPIE Freeboard. สืบค้นเมื่อ 3 สิงหาคม 2561 จาก https://bit.ly/2tADvXJ

การศึกษาค้นคว้า แอพลิเคชั่น Node-RED. สืบค้นเมื่อ 3 สิงหาคม 2561 ,https://bit.ly/2TkaFsW

การศึกษาค้นคว้า Arduino. สืบค้นเมื่อ 22 กรกฎาคม 2561 , จาก IDEhttps://bit.ly/2BTtY2C

การศึกษาค้นคว้า Internet of Things(Iot) . สืบค้นเมื่อ 14 ตุลาคม 2561 , จาก https://bit.ly/2tp0kxz

การศึกษาค้นคว้า NTP (Network Time Protocol) . สืบค้นเมื่อ 14 ตุลาคม 2561 2561, จาก https://bit.ly/2EhCbPD

การศึกษาค้นคว้า UDP (User Datagram Protocol) . สืบค้นเมื่อ 14 ตุลาคม 2561 2561, จาก https://bit.ly/2VZ0itc

การศึกษาค้นคว้า TCP/IP (Transmission Control Protocol/ Internet Protocol) . สืบค้นเมื่อ 2 ธันวาคม 2561 , จาก https://bit.ly/2FxfQ1k

การศึกษาค้นคว้า MQTT Server Potocal. สืบค้นเมื่อ 2 ธันวาคม 2561 , จาก https://bit.ly/2tA6WZX

การศึกษาค้นคว้าองค์ประกอบสำคัญในการเชื่อมต่อ Internet of Things เข้ากับ Cloud สืบค้นเมื่อ 4 มีนาคม 2562 , จาก https://bit.ly/2FGOwMM

การศึกษาค้นคว้า End-to-End Encryption สืบค้นเมื่อ 4 มีนาคม 2562, จาก https://bit.ly/2WH1BfJ

ภาคผนวก

การเปรียบเทียบการพัฒนาโปรแกรมประยุกต์ IOT สำหรับโทรศัพท์มือถือระหว่าง IOT Cloud Platform Application และ End-to-End Application

กรณีศึกษา เครื่องให้อาหารปลา

<u>ในส่วนอุปกรณ์(Hardware)</u>

Arduino WeMos D1

ระบบ Sensor Node ในงานวิจัยนี้เลือกใช้ บอร์ด WeMos-D1 R2 เป็นบอร์ดที่นำเอา Arduino UNO R3 กับ ESP8266 WIFI มารวมกันในบอร์ดๆ เดียวทำให้ผู้ใช้งานไม่ต้องต่อสายไฟเอง คือ Arudino UNO with Buitin WiFi บอร์ดรุ่นนี้จะไม่มีสาย Sync. แถมมาให้เพราะผู้ใช้งานสามารถใช้งานสาย Micro USB Cable ที่ใช้กับมือถือAndroid ทั่วไปได้เลยครับ ส่วนในการอัพโหลดโปรแกรมลงบนบอร์ด เราสามารถใช้ Arudino IDE ได้เลย แต่จำเป็นต้องติดตั้ง USB Driver ของมันก่อน

ภาพแสดง Arduino WeMos D1

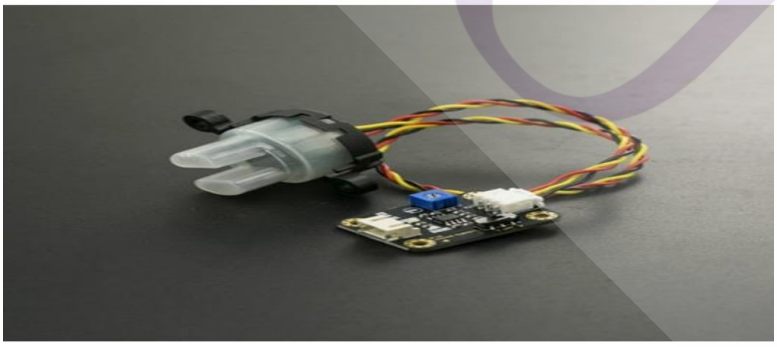
RTC Real Time Clock

(RTC) ก็คือ อุปกรณ์ที่ให้ค่าเวลาตามจริง ซึ่งทำงานของบอร์ดนี้จะจับสัญญาณนาฬิกาที่ได้มาจาก Crystal นั่นเอง บางรุ่นก็จะมีถ่านสำรองมาให้ถ่านสำรองจะทำหน้าที่ในการบันทึกเวลาอย่างต่อเนื่องถึงแม้ว่าจะไม่มีไฟมาเลี้ยงมาที่ตัวบอร์ดส่วนเวลาก็ยังคงนับได้ต่อเนื่อง ทำให้ผู้ใช้ไม่ต้องเสียเวลามาตั้งเวลาใหม่หลังจากที่ไฟเลี้ยงหยุดจ่าย

ภาพแสดงRTC Real Time Clock Memory Module

Analog turbidity sensor

Sensor Node วัดระดับความขุ่นของน้ำ)Analog turbidity sensor) จะตรวจสอบจากการนำและหักเหของแสงของน้ำในตู้ปลาแล้วแจ้งค่าที่ได้เตือนไปยังเจ้าของให้ทำการเปลี่ยนน้ำในตู้ปลา โดยการทำงานแรงดัน 5 โวลต์ มีเอาต์พุตได้สองแบบ คือ อนาลอกแรงดันตั้งแต่ 0.4 ถึง 5 โวลต์ และดิจิตอล High / Low โดยปรับค่าที่ตัวต้านทานปรับค่าได้ ข้อควรระวังในการใช้งานคือหัวโพรบจะไม่กันน้ำ ไม่สามารถจมน้ำได้ การจุ่มวัดทำได้เฉพาะส่วนปลายโพรบที่เป็นแท่งพลาสติกโปร่งแสง



ภาพแสดง Analog turbidity sensor

Motor Servo

เป็นมอเตอร์ที่มีการควบคุมการเคลื่อนที่ไม่ว่าจะเป็นระยะ ความเร็ว มุมการหมุน โดยใช้การควบคุมแบบป้อนกลับ (Feedback control) เป็นอุปกรณ์ที่สามารถควบคุมเครื่องจักรกล หรือระบบการทำงานให้เป็นไปตามความต้องการ เช่น ควบคุมความเร็ว (Speed), ควบคุมแรงบิด (Torque), ควบคุมแรงตำแหน่ง (Position), ระยะทางในการเคลื่อนที่(หมุน) (Position Control) ของตัวมอเตอร์ได้ ซึ่งมอเตอร์ทั่วไปไม่สามารถควบคุมในลักษณะการทำงานเบื้องต้นได้ และให้ผลลัพธ์ตามความต้องการที่ต้องใช้ความแม่นยำสูง

ภาพที่แสดง Motor Servo

Character LCD Display

LCD ย่อมาจากคำว่า Liquid Crystal Display ซึ่งเป็นจอที่ทำมาจากผลึกคริสตอลเหลว หลักการคือด้านหลังจอจะมีไฟส่องสว่าง หรือที่เรียกว่า Backlight อยู่ เมื่อมีการปล่อยกระแสไฟฟ้าเข้าไปกระตุ้นที่ผลึก ก็จะทำให้ผลึกโปร่งแสง ทำให้แสงที่มาจากไฟ Backlight แสดงขึ้นมาบนหน้าจอ ส่วนอื่นที่โดนผลึกปิดกั้นไว้ จะมีสีที่แตกต่างกันตามสีของผลึกคริสตอล เช่น สีเขียว หรือสีฟ้า ทำให้เมื่อมองไปที่จอก็จะพบกับตัวหนังสือสีขาว แล้วพบกับพื้นหลังสีต่างๆกันไปและจอ LCD จะแบ่งเป็น 2 แบบใหญ่ๆตามลักษณะการแสดงผลดังนี้

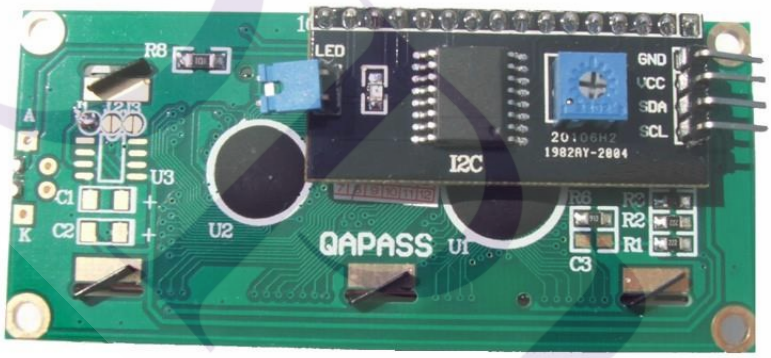
1. Character LCD เป็นจอที่แสดงผลเป็นตัวอักษรตามช่องแบบตายตัว เช่น จอ LCD ขนาด 16x2 หมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 16 ตัว และมีทั้งหมด 2 บรรทัดให้ใช้งาน ส่วน 20x4 จะหมายถึงใน 1 แถว มีตัวอักษรใส่ได้ 20 ตัว และมีทั้งหมด 2 บรรทัด

2. Graphic LCD เป็นจอที่สามารถกำหนดได้ว่าจะให้แต่ละจุดบนหน้าจอกั้นแสง หรือปล่อยแสงออกไป ทำให้จอนี้สามารถสร้างรูปขึ้นมาบนหน้าจอได้ การระบุขนาดจะระบุในลักษณะ

ของจำนวนจุด) Pixels) ในแต่ละแนว เช่น 128x64 หมายถึงจอที่มีจำนวนจุดตามแนวนอน 128 จุด และมีจุดตามแนวตั้ง 64 จุด

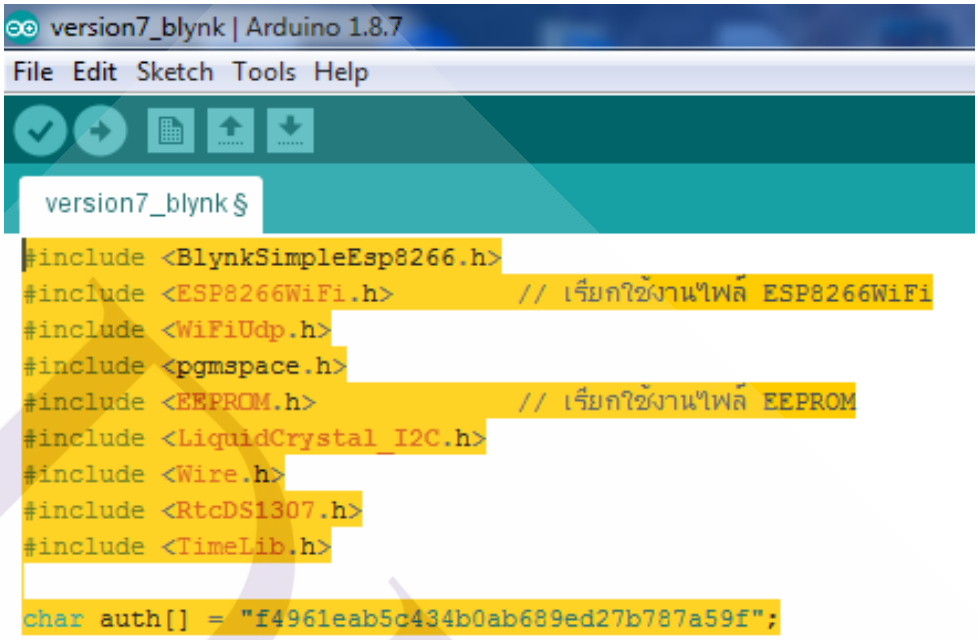
การเชื่อมต่อ Character LCD จะมีด้วยกัน 2 แบบ คือ

- การเชื่อมต่อแบบขนาน - เป็นการเชื่อมต่อจอ LCD เข้ากับบอร์ด Arduino โดยตรง โดยจะแบ่งเป็นการเชื่อมต่อแบบ 4 บิต และการเชื่อมต่อแบบ 8 บิต ใน Arduino จะนิยมเชื่อมต่อแบบ 4 บิต เนื่องจากใช้สายในการเชื่อมต่อน้อยกว่า
- การเชื่อมต่อแบบอนุกรม - เป็นการเชื่อต่อกับจอ LCD ผ่านโมดูลแปลงรูปแบบการเชื่อมต่อกับจอ LCD จากแบบขนาน มาเป็นการเชื่อมต่อแบบอื่นที่ใช้สายน้อยกว่า เช่น การใช้โมดูล I2C Serial Interface จะเป็นการนำโมดูลเชื่อมเข้ากับตัวจอ LCD แล้วใช้บอร์ด Arduino เชื่อมต่อกับบอร์ดโมดูลผ่านโปรโตคอล I2C ทำให้ใช่สายเพียง 4 เส้น ก็ทำให้หน้าจอแสดงผลข้อความต่างๆออกมาได้

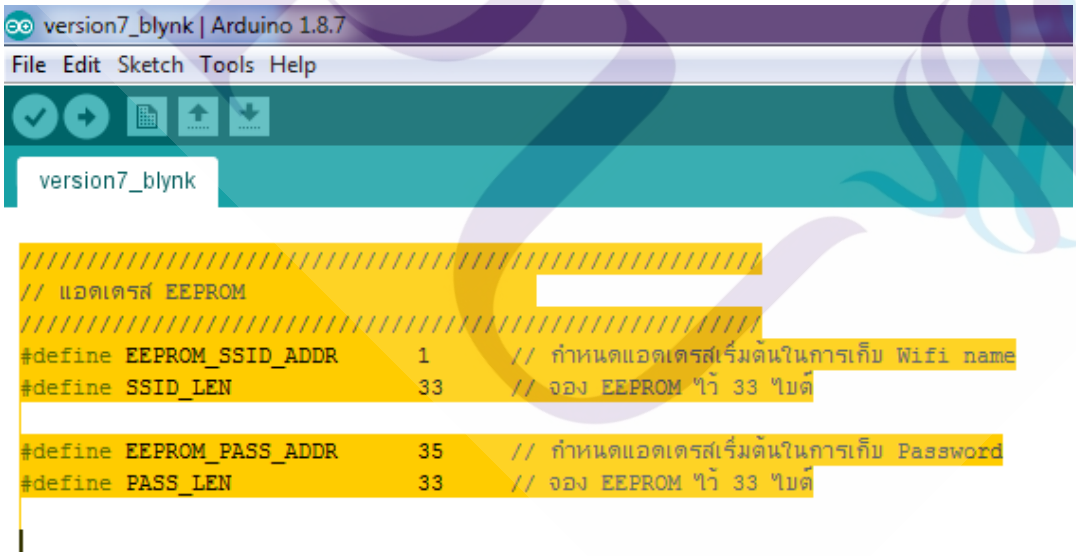


ภาพแสดง Character LCD Display

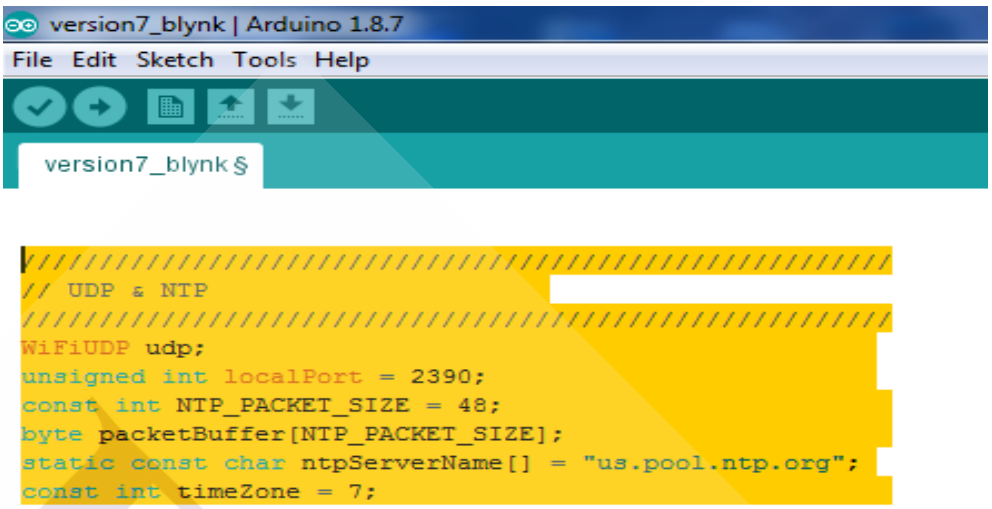
ในส่วนการเขียนโปรแกรมในส่วนแอพพลิเคชั่น Blynk



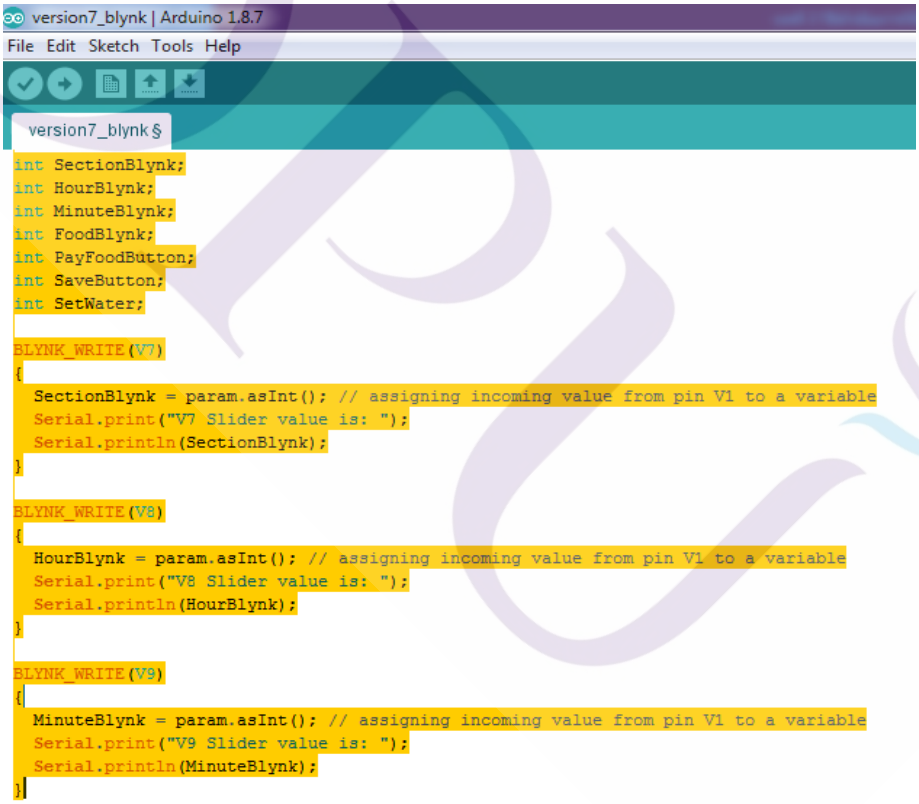
ภาพแสดงการเรียกใช้งานไฟล์คำสั่งฟังชั่นการทำงานต่างๆแอพพลิเคชั่น Blynk



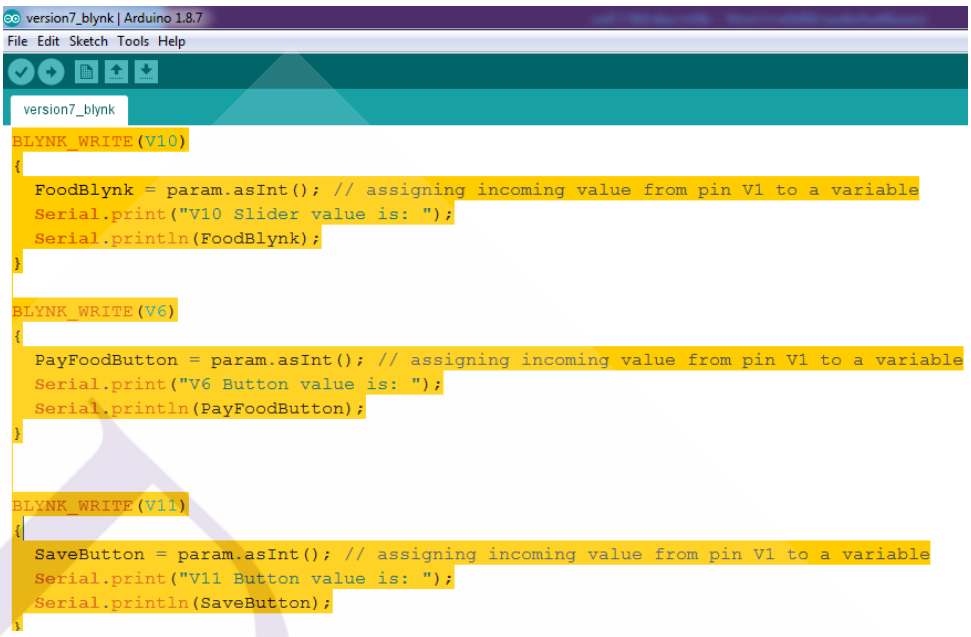
ภาพแสดงการกำหนดค่าการทำงานของEEPOMในการอ่านค่าเก็บข้อมูลชื่อและรหัสผ่านWiFiที่จะใช้เชื่อมต่อ



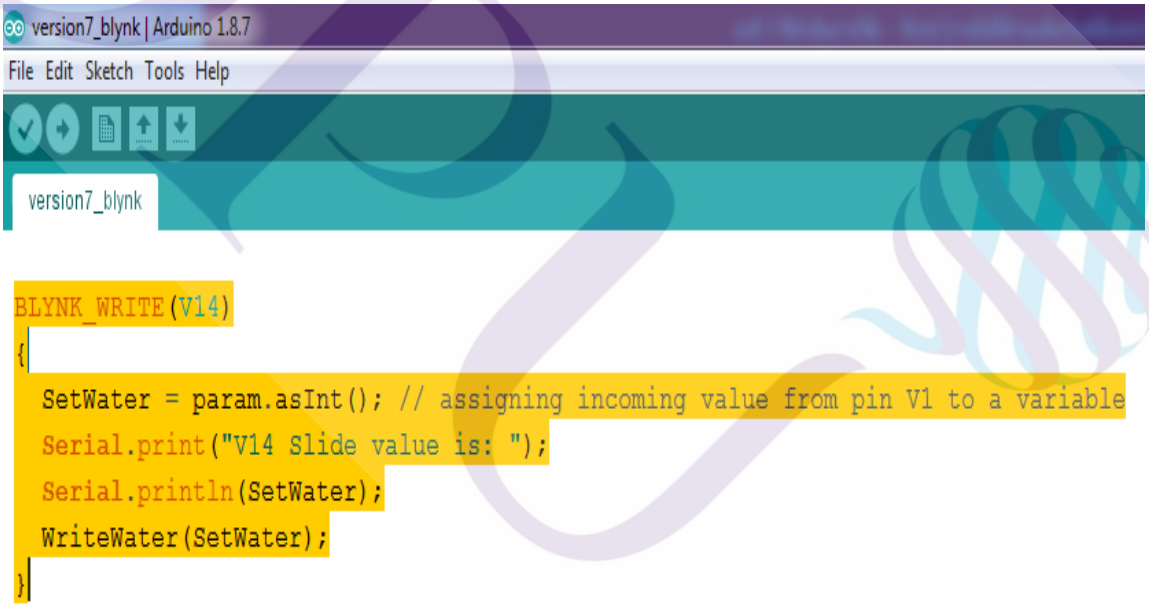
ภาพแสดงการกำหนดค่าเงื่อนไขการทำงานของ Potocal UDP กับ NTP



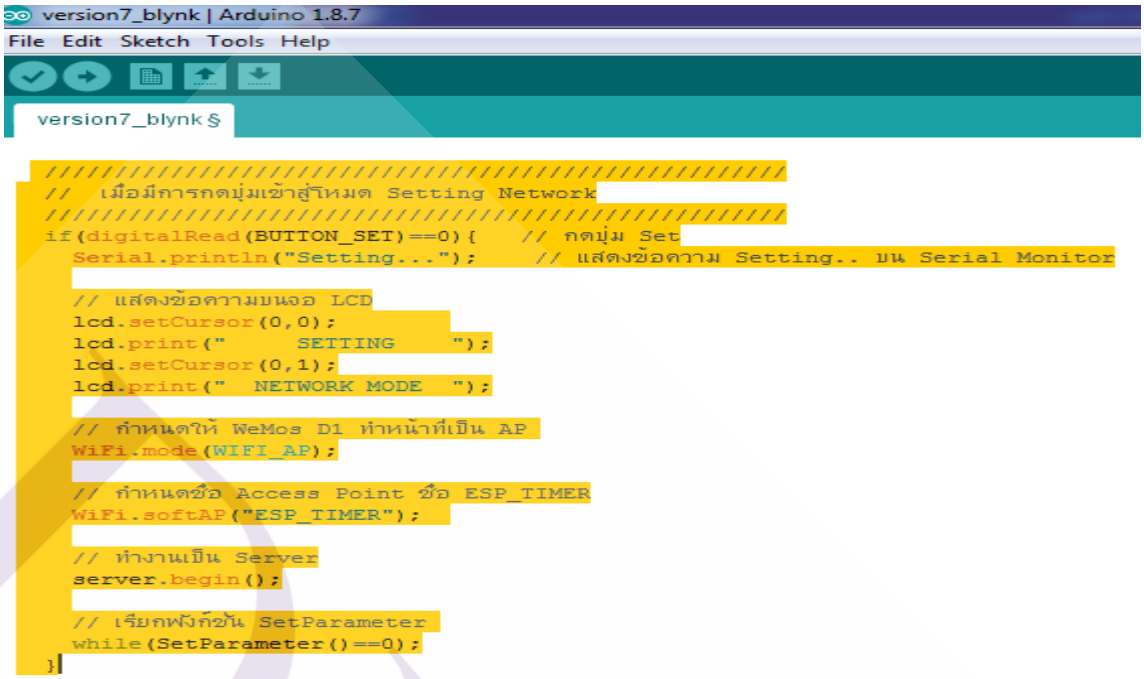
ภาพแสดงการกำหนดเงื่อนไขการทำงานของค่าตัวแปรต่างๆ ของแอพพลิเคชั่น Blynk



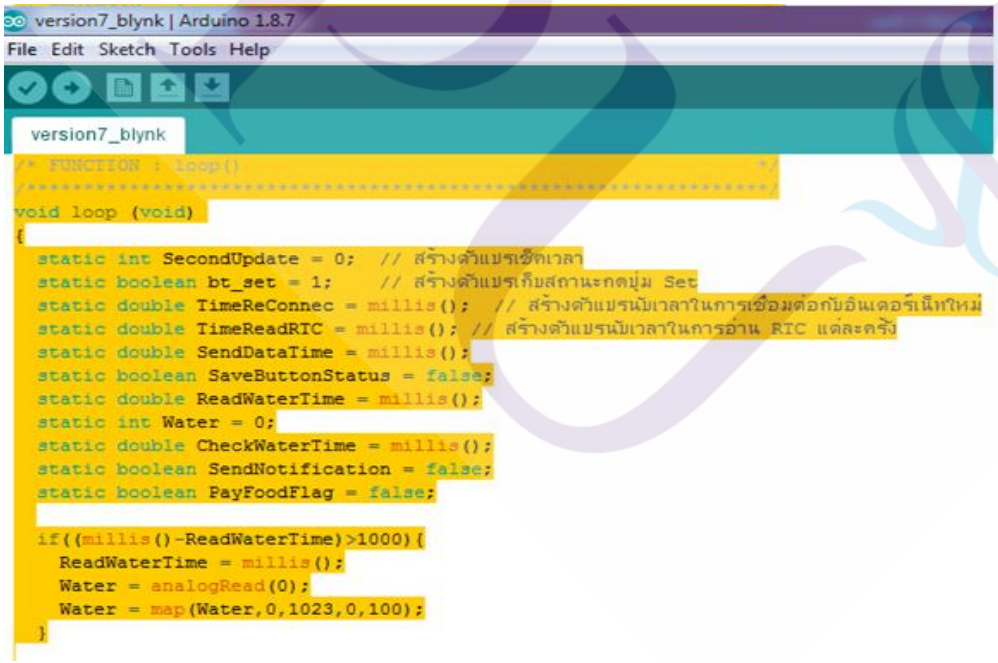
ภาพที่แสดงการกำหนดเงื่อนไขการทำงานของค่าตัวแปรต่างๆของแอพพลิเคชั่นBlynk (ต่อ)



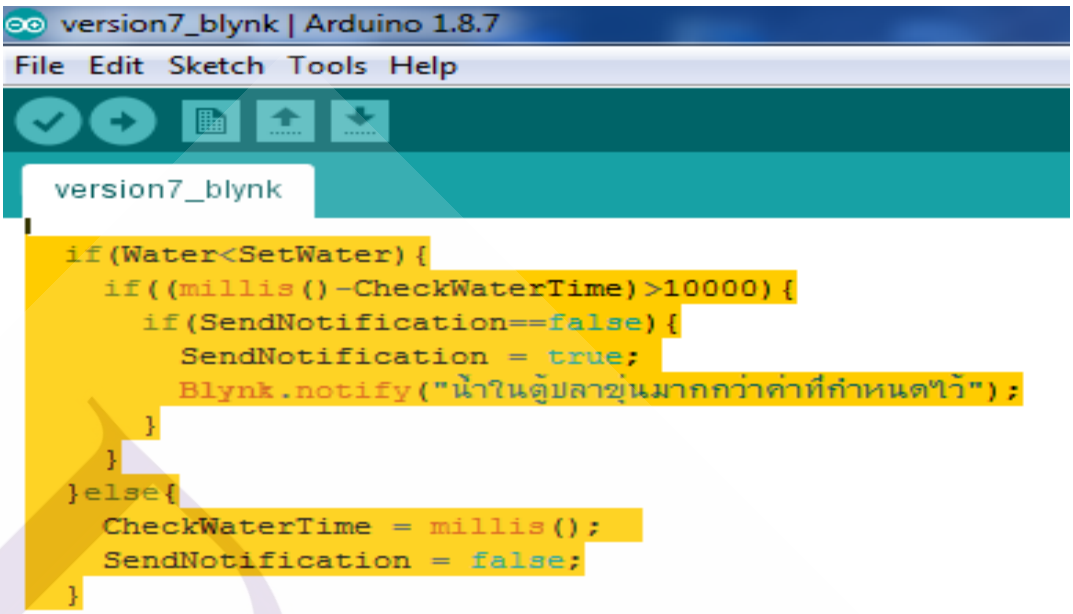
ภาพแสดงการกำหนดเงื่อนไขการทำงานของค่าตัวแปรต่างๆของแอพพลิเคชั่น Blynk (ต่อ)



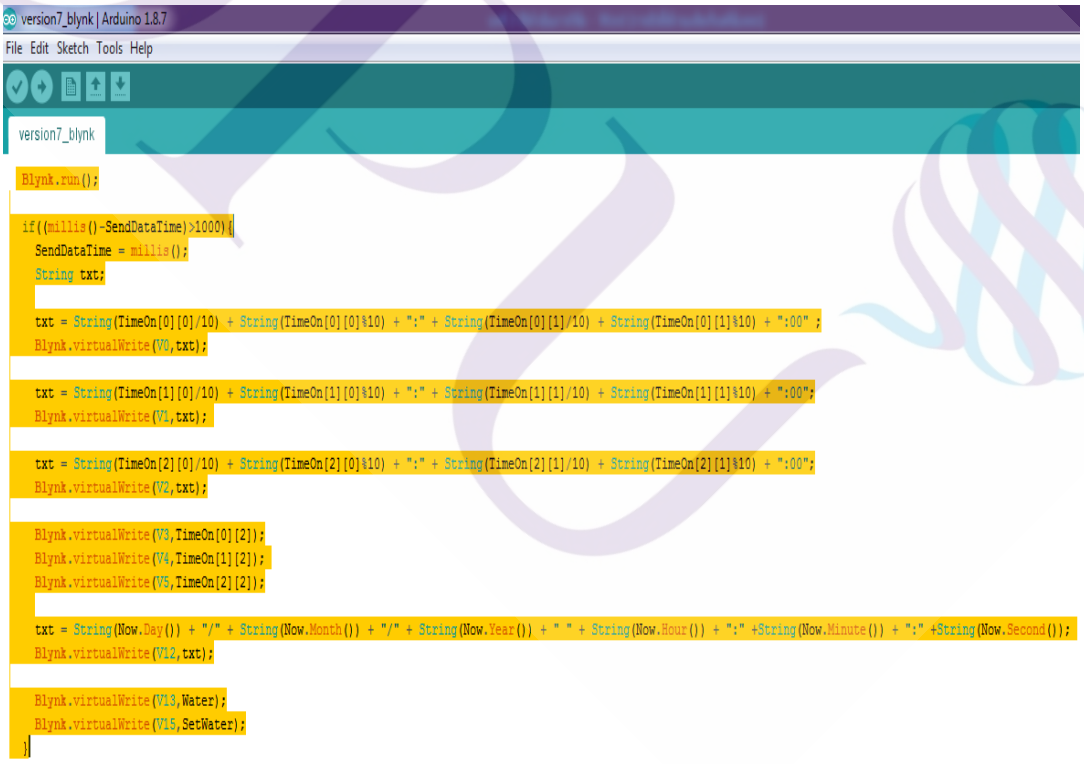
ภาพแสดงการกำหนดฟังชั่นเงื่อนไขการทำงานในการเข้าสู่โหมด SETTING NETWORK



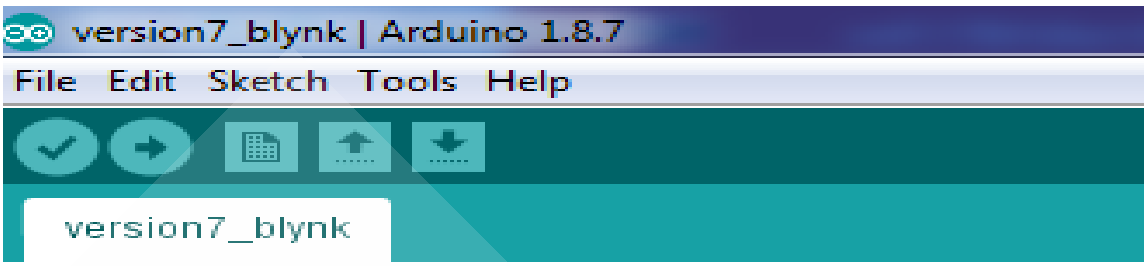
ภาพแสดงการกำหนดเงื่อนไขการทำงานฟังชั่นการวนลูปของระบบ



ภาพแสดงการกำหนดฟังชั่นเงื่อนไขการเช็คค่าเวลาความขุ่นของน้ำในตู้ปลา

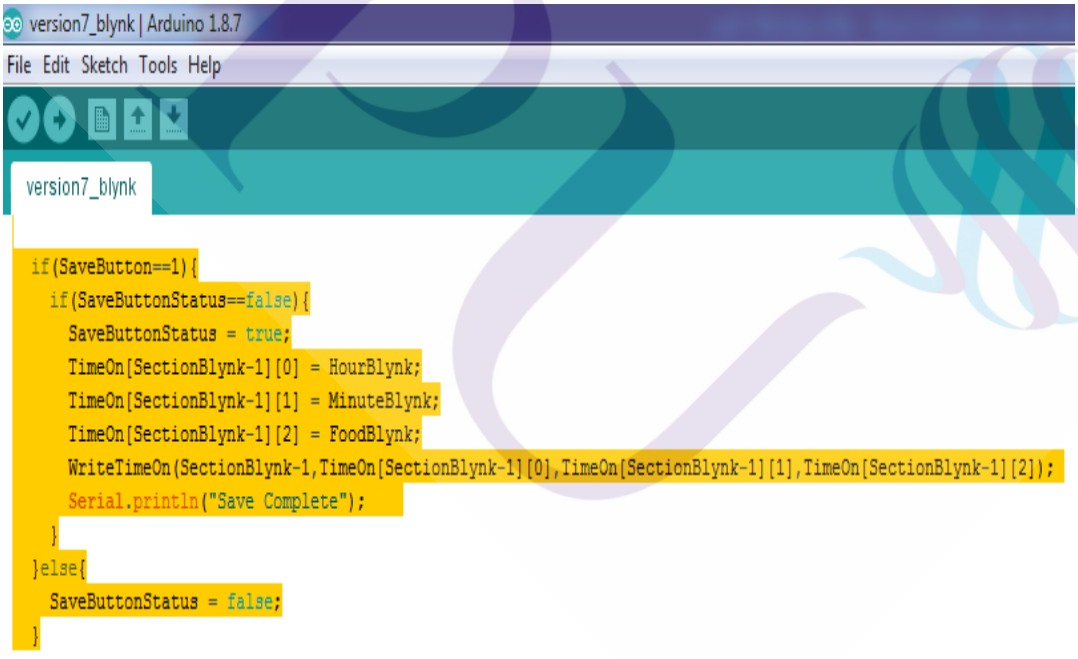


ภาพแสดงการรันค่าตัวแปรและฟังชั่นเงื่อนไขการทำงานส่งไปยังแอพพลิเคชั่นBlynk



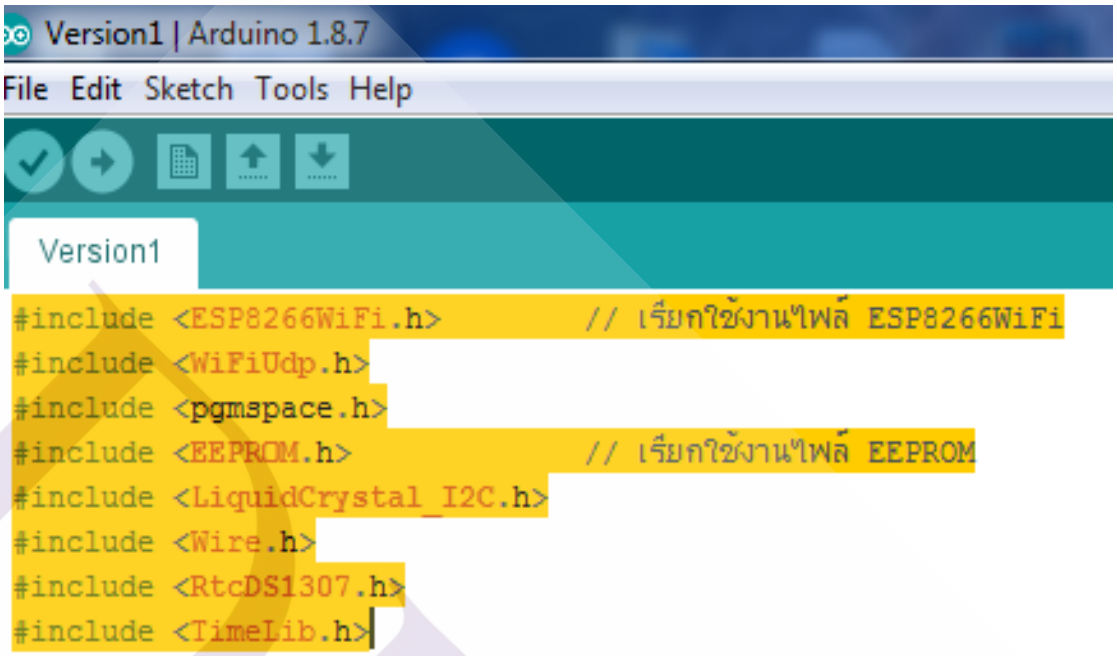

```
if(PayFoodButton==1){
  if(PayFoodFlag == false){
    PayFoodFlag = true;
    FeedFood();
    FeedStart();
  }
}else{
  PayFoodFlag = false;
}
```

ภาพแสดงการกำหนดฟังชั่นเงื่อนไขการทำงานในการให้อาหารปลา

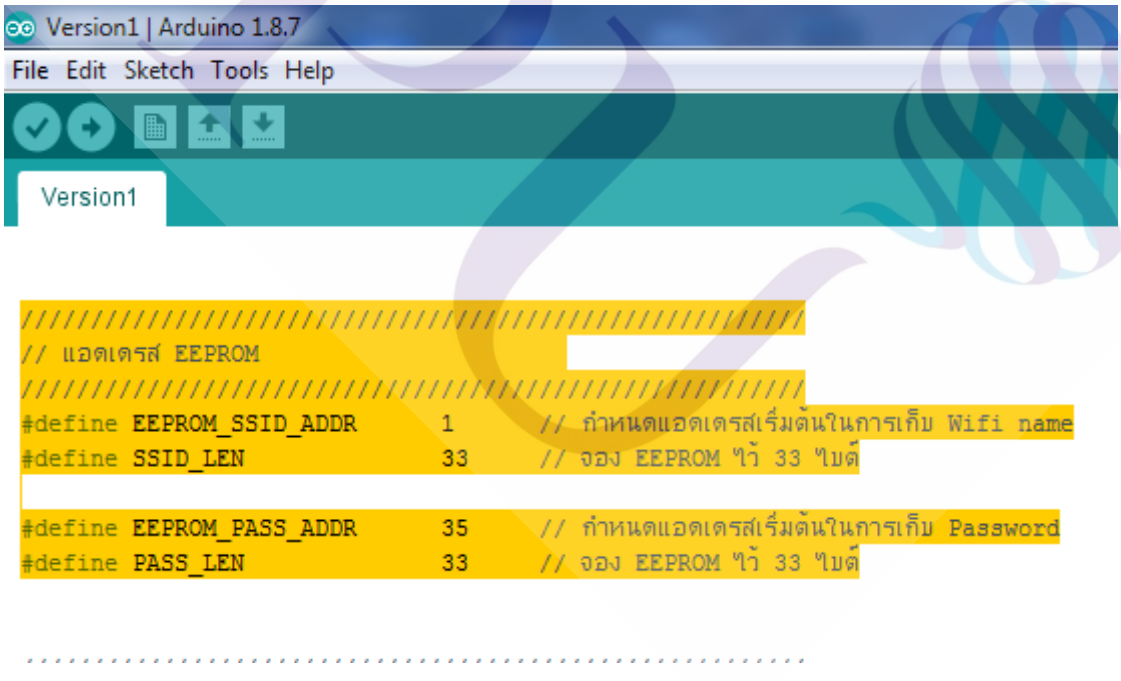


ภาพแสดงการกำหนดฟังชั่นเงื่อนไขการบันทึกข้อมูลการตั้งค่าเวลาแต่ละช่วงการเวลาให้อาหารปลา

ในส่วนการเขียนโปรแกรมในส่วนแอพพลิเคชั่น Inventor 2 (ส่วน Server)



ภาพแสดงการเรียกใช้งานไฟล์คำสั่งฟังชั่นการทำงานต่างๆแอพพลิเคชั่นInventor 2



ภาพแสดงการกำหนดค่าการทำงานของ EEPOM ในการอ่านค่าเก็บข้อมูลชื่อและรหัสผ่านWiFi ที่จะใช้เชื่อมต่อ

Version1 | Arduino 1.8.7

File Edit Sketch Tools Help

Version1

```
////////////////////////////////////////////////////////////
// UDP & NTP
////////////////////////////////////////////////////////////
WiFiUDP udp;
unsigned int localPort = 2390;
const int NTP_PACKET_SIZE = 48;
byte packetBuffer[NTP_PACKET_SIZE];
static const char ntpServerName[] = "us.pool.ntp.org";
const int timeZone = 7;
```

ภาพแสดงการกำหนดค่าเงื่อนไขการทำงานของ Protocal UDP กับ NTP

Version1 | Arduino 1.8.7

File Edit Sketch Tools Help

Version1

```
  ////////////////////////////////////////////////////
  //  เมื่อมีการกดปุ่มเข้าสู่โหมด Setting Network
  ////////////////////////////////////////////////////
  if(digitalRead(BUTTON_SET)==0){    // กดปุ่ม Set
    Serial.println("Setting...");     // แสดงข้อความ Setting.. บน Serial Monitor

    // แสดงข้อความบนจอ LCD
    lcd.setCursor(0,0);
    lcd.print("     SETTING     ");
    lcd.setCursor(0,1);
    lcd.print("  NETWORK MODE  ");

    // กำหนดให้ WeMos D1 ทำหน้าที่เป็น AP
    WiFi.mode(WIFI_AP);

    // กำหนดชื่อ Access Point ชื่อ ESP_TIMER
    WiFi.softAP("ESP_TIMER");

    // ทำงานเป็น Server
    server.begin();

    // เรียกฟังก์ชัน SetParameter
    while(SetParameter()==0);
  }
```

ภาพแสดงการกำหนดฟังชั่นเงื่อนไขการทำงานในการเข้าสู่โหมด SETTING NETWORK

Version1 | Arduino 1.8.7

File Edit Sketch Tools Help

Version1

```
/* FUNCTION : WebServer()                                                */
/***********************************************************************/
int WebServer (void)
{
  int len;
  boolean GetH1 = false;
  boolean GetM1 = false;
  boolean GetF1 = false;
  boolean GetH2 = false;
  boolean GetM2 = false;
  boolean GetF2 = false;
  boolean GetH3 = false;
  boolean GetM3 = false;
  boolean GetF3 = false;
  char Buffer[10];
```

ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงาน Webserver

Version1 | Arduino 1.8.7

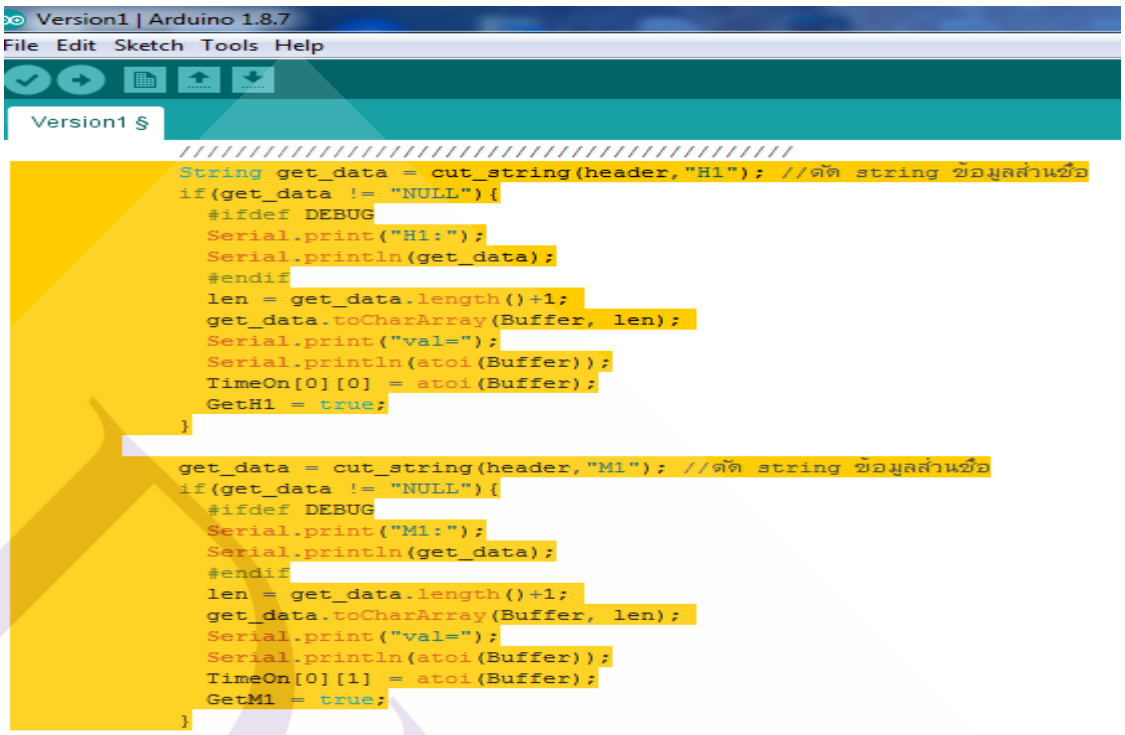
File Edit Sketch Tools Help

Version1

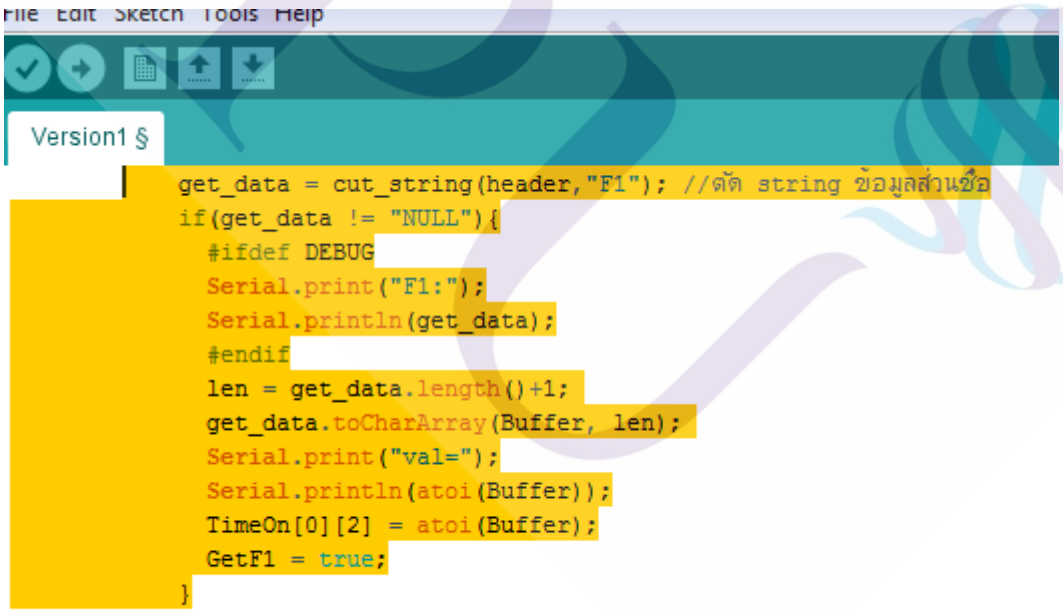
```
client = server.available();

if (client) {                             // If a new client connects,
  Serial.println("New Client.");          // print a message out in the serial port
  String currentLine = "";                // make a String to hold incoming data from the client
  while (client.connected()) {            // loop while the client's connected
    wdt_reset();
    if (client.available()) {             // if there's bytes to read from the client,
      char c = client.read();             // read a byte, then
      Serial.write(c);                    // print it out the serial monitor
      header += c;
      if (c == '\n') {                    // if the byte is a newline character
        // if the current line is blank, you got two newline characters in a row.
        // that's the end of the client HTTP request, so send a response:
        if (currentLine.length() == 0) {
          // HTTP headers always start with a response code (e.g. HTTP/1.1 200 OK)
          // and a content-type so the client knows what's coming, then a blank line:
          client.println("HTTP/1.1 200 OK");
          client.println("Content-type:text/html");
          client.println("Connection: close");
          client.println();
```

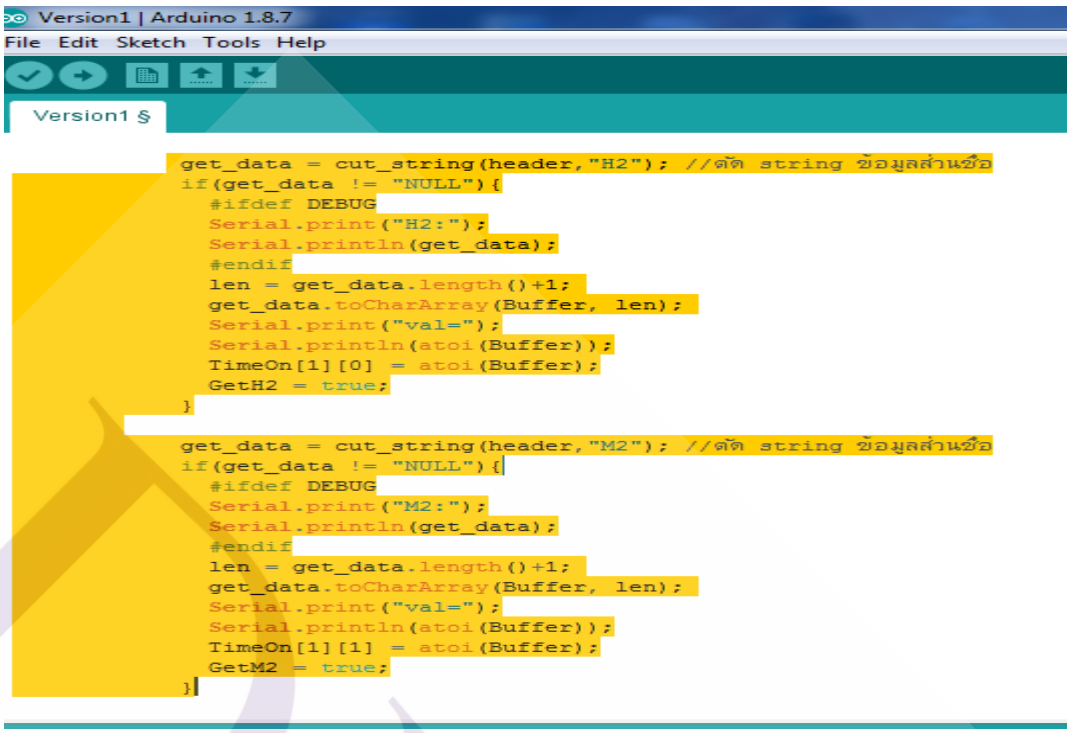
ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานWebserverแอพพลิเคชั่นInventor 2



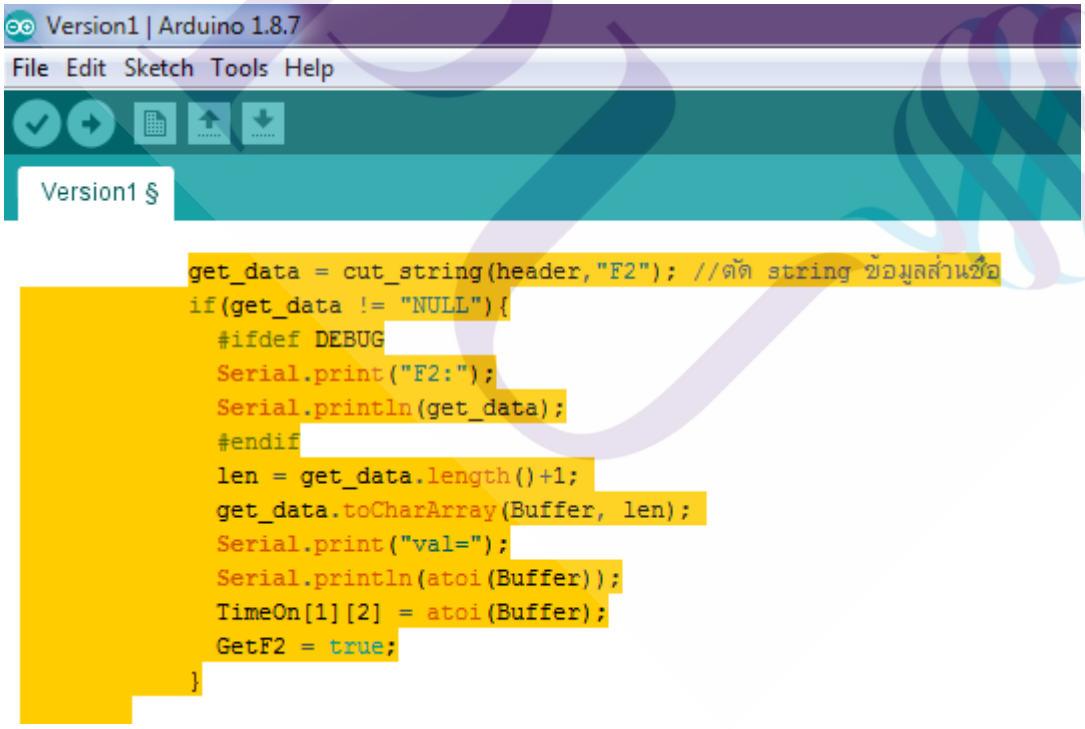
ภาพแสดงข้อมูลฟังชั่นการทำงานกำหนดค่าตัวแปรการให้อาหารปลาช่วงเวลาที่1



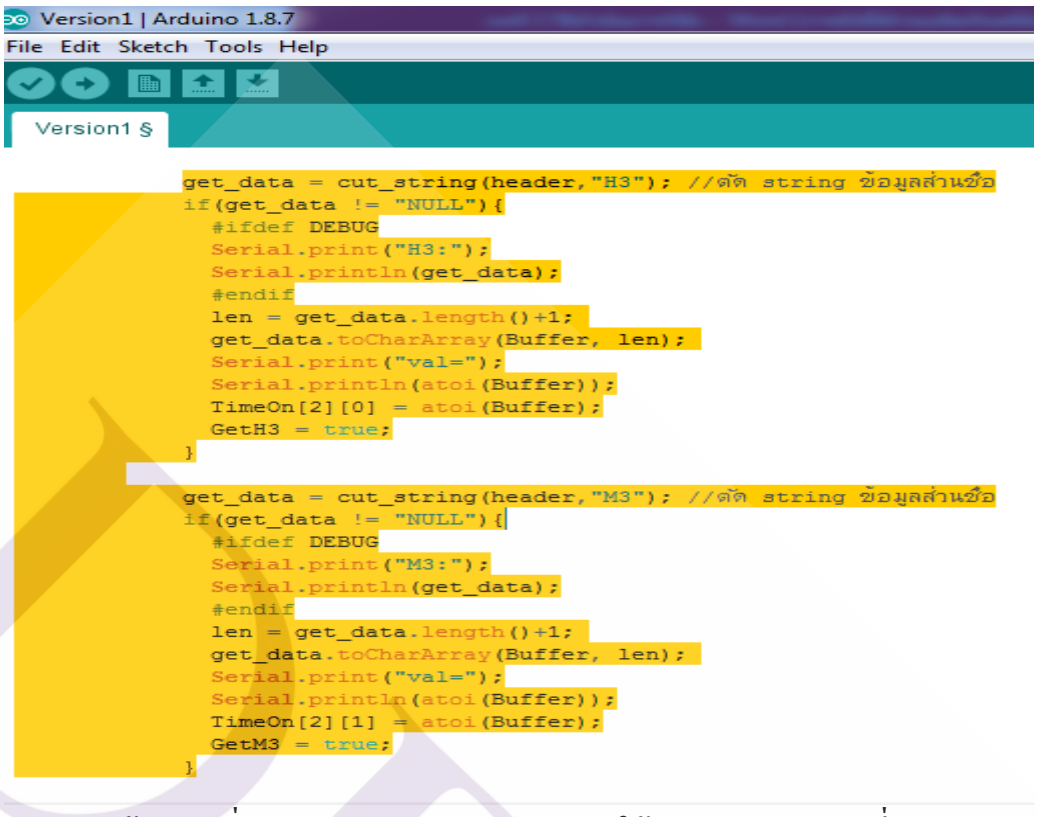
ภาพแสดงข้อมูลฟังชั่นการทำงานกำหนดค่าตัวแปรการให้อาหารปลาช่วงเวลาที่1 (ต่อ)



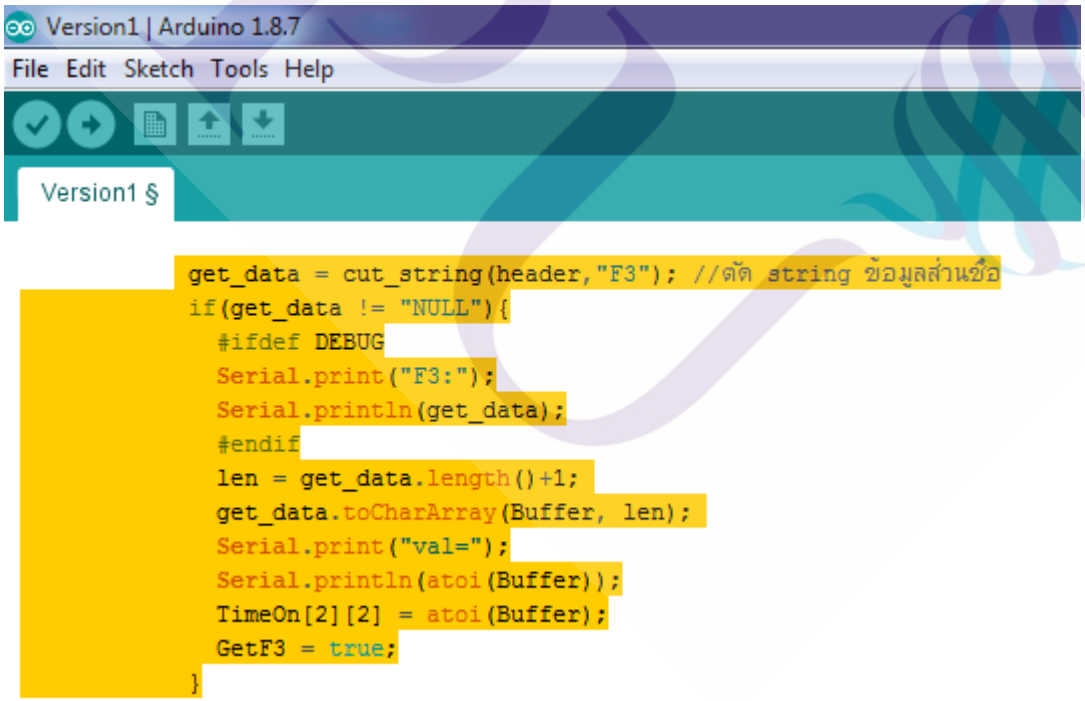
ภาพแสดงข้อมูลฟังชั่นการทำงานกำหนดค่าตัวแปรการให้อาหารปลาช่วงเวลาที่ 2



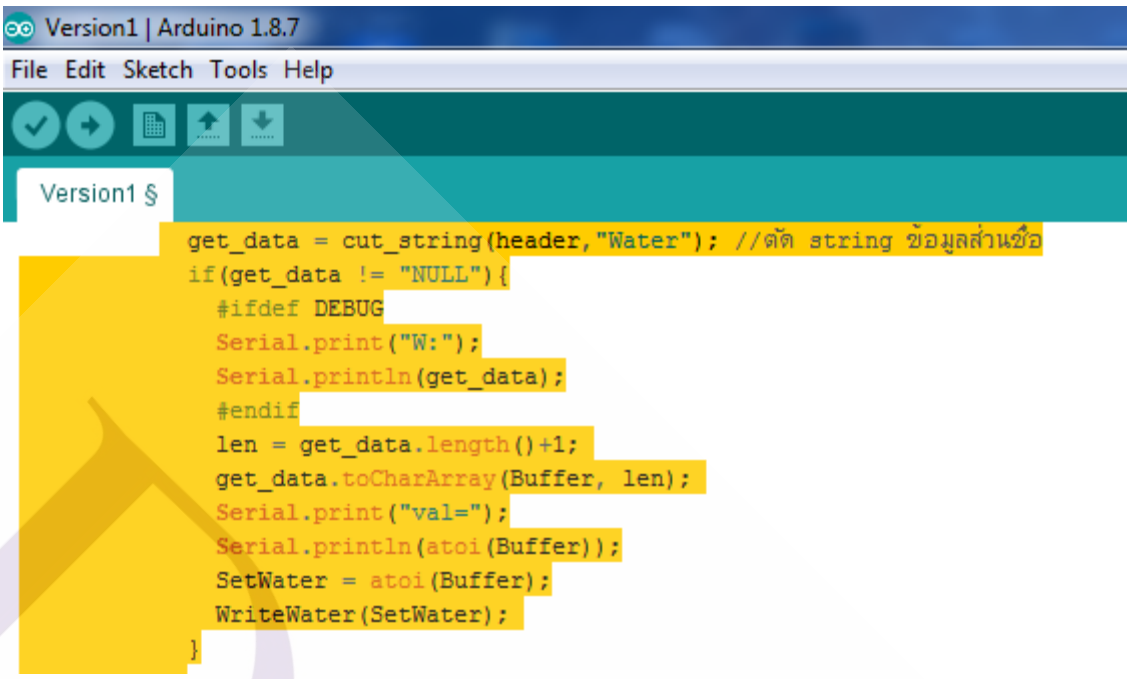
ภาพแสดงข้อมูลฟังชั่นการทำงานกำหนดค่าตัวแปรการให้อาหารปลาช่วงเวลาที่2 (ต่อ)



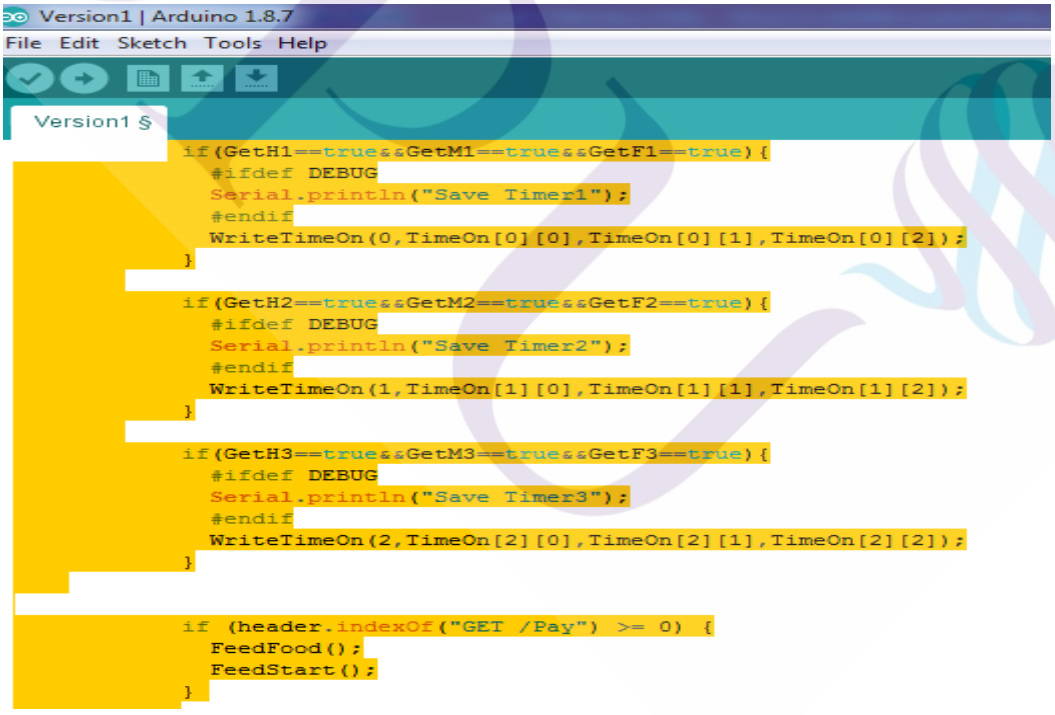
ภาพแสดงข้อมูลฟังชั่นการทำงานกำหนดค่าตัวแปรการให้อาหารปลาช่วงเวลาที่3



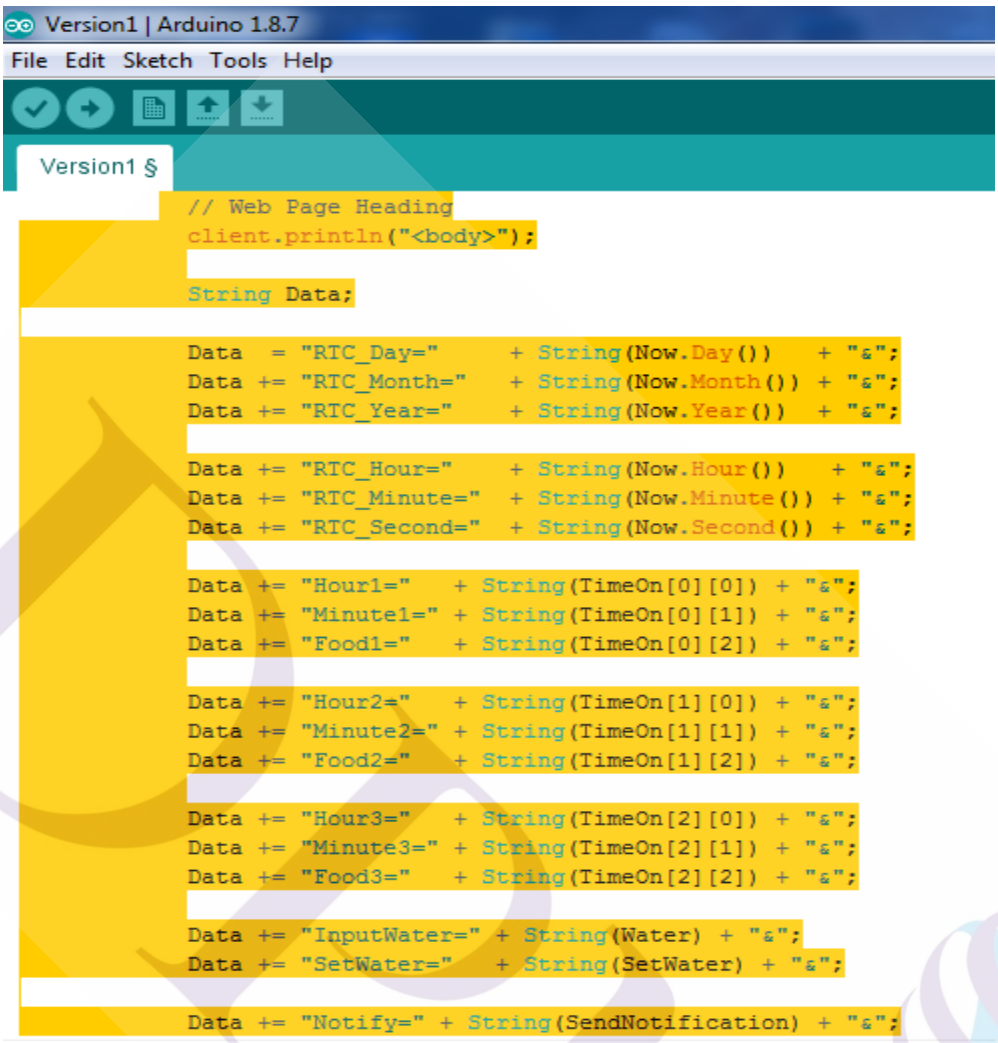
ภาพแสดงข้อมูลฟังชั่นการทำงานกำหนดค่าตัวแปรการให้อาหารปลาช่วงเวลาที่3 (ต่อ)



ภาพแสดงข้อมูลฟังชั่นการทำงานกำหนดค่าตัวแปรการวัดค่าความขุ่นของน้ำ

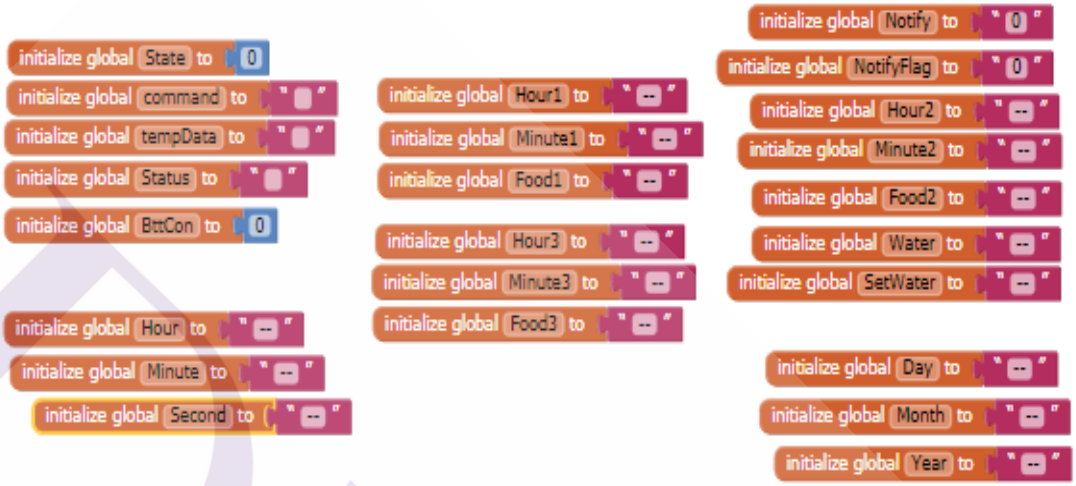


ภาพแสดงข้อมูลฟังชั่นการทำงานการกำหนดค่าตัวแปรต่างๆบนแอพพลิเคชั่น

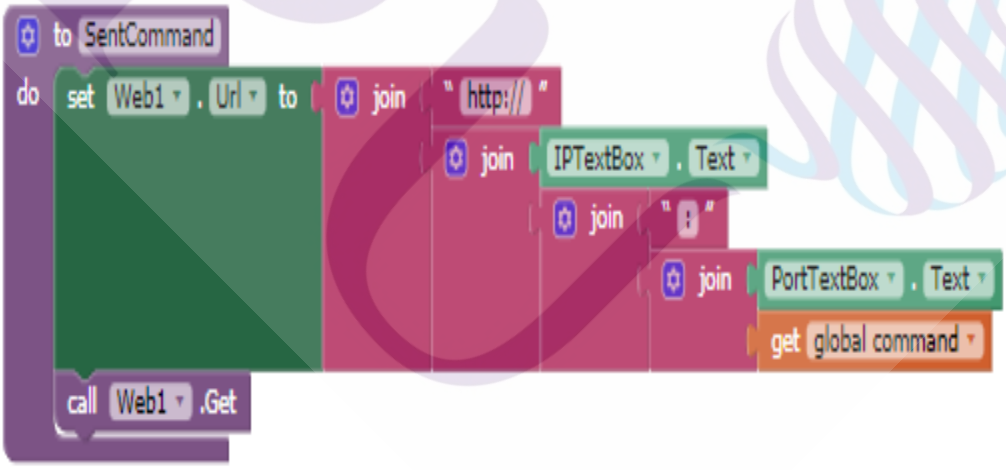


ภาพแสดงการกำหนดข้อมูลค่าตัวแปรโชว์บนWebserver (ต่อ)

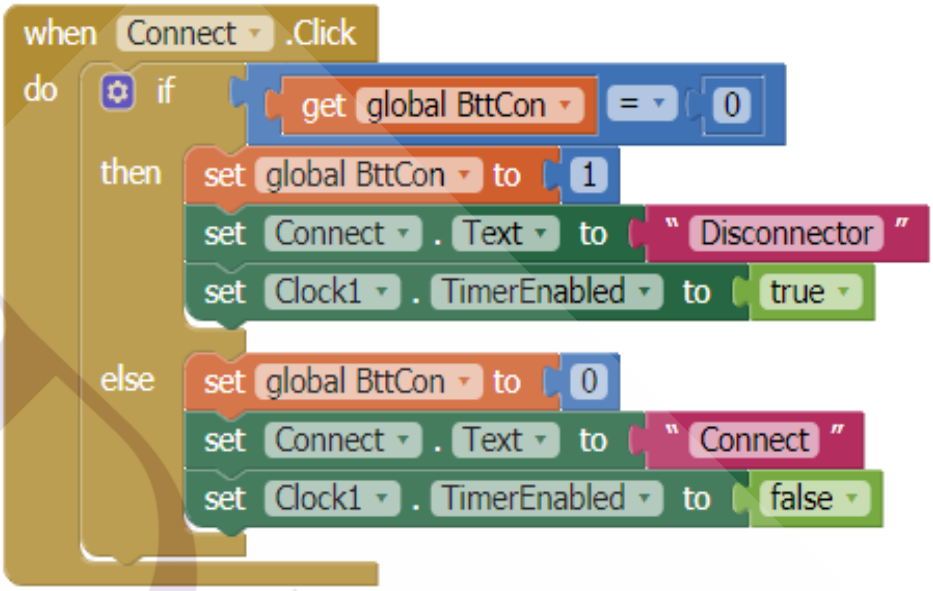
ในส่วนการเขียนโปรแกรมในส่วนแอพพลิเคชั่น Inventor 2 (ส่วน Application)



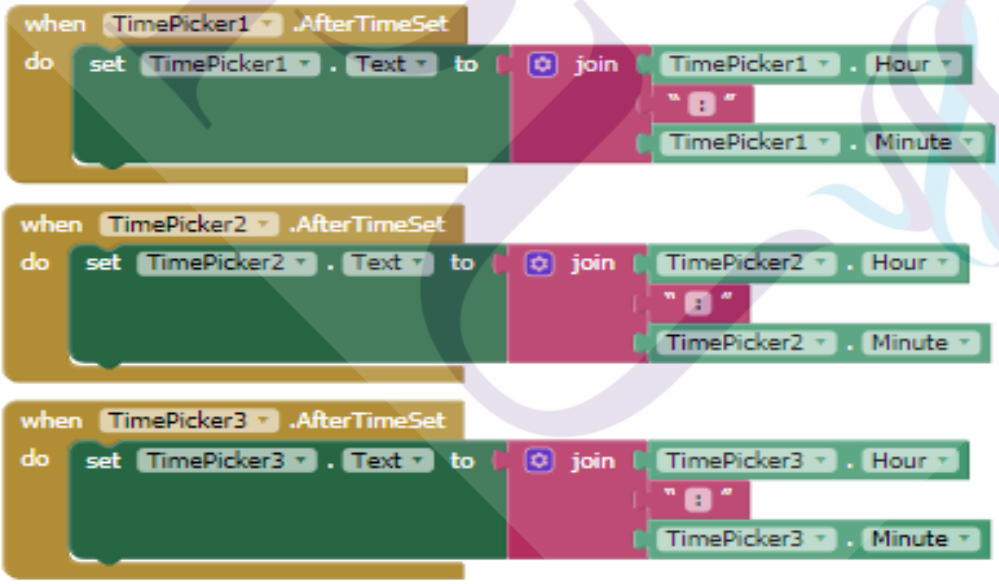
ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานประกาศค่าตัวแปร initialize global ต่างๆ



ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงาน SentCommand

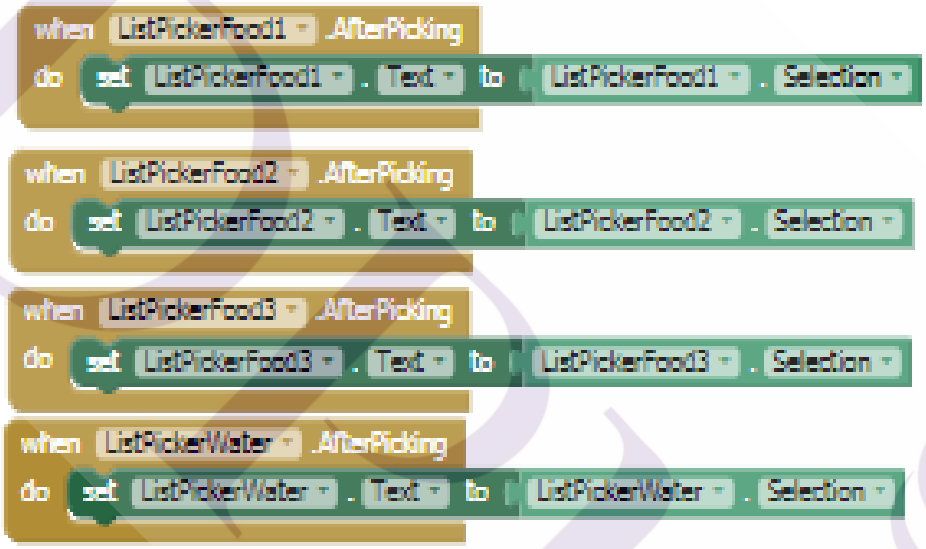


ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานของปุ่ม Connect



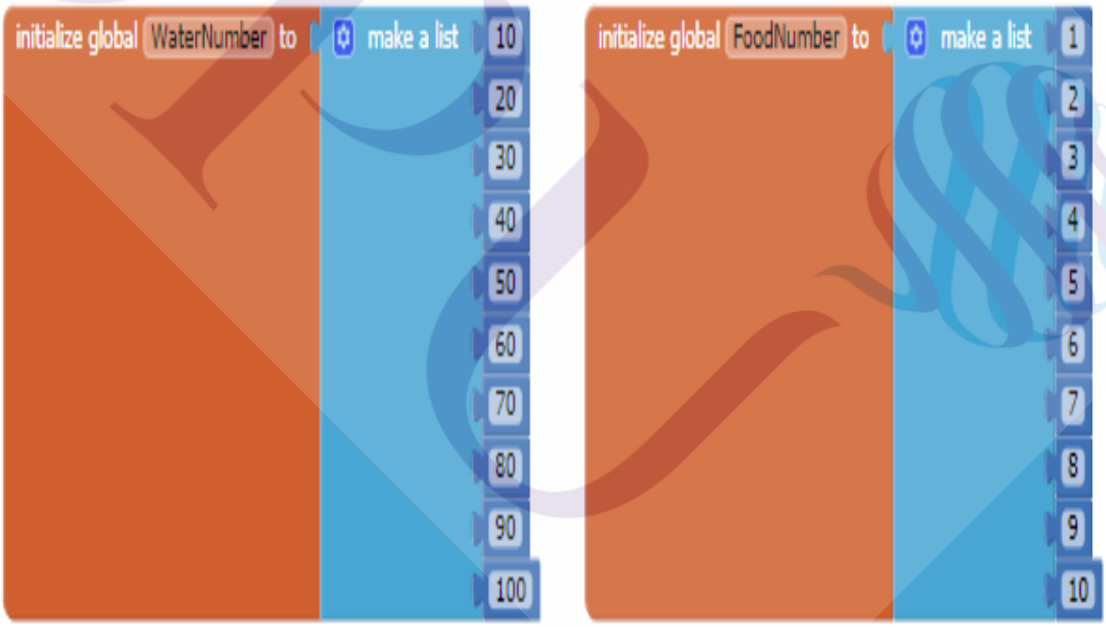
ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานตั้งค่าการให้อาหารปลาแต่ละช่วงเวลา

ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานการจ่ายอาหารปลาอัตโนมัติ

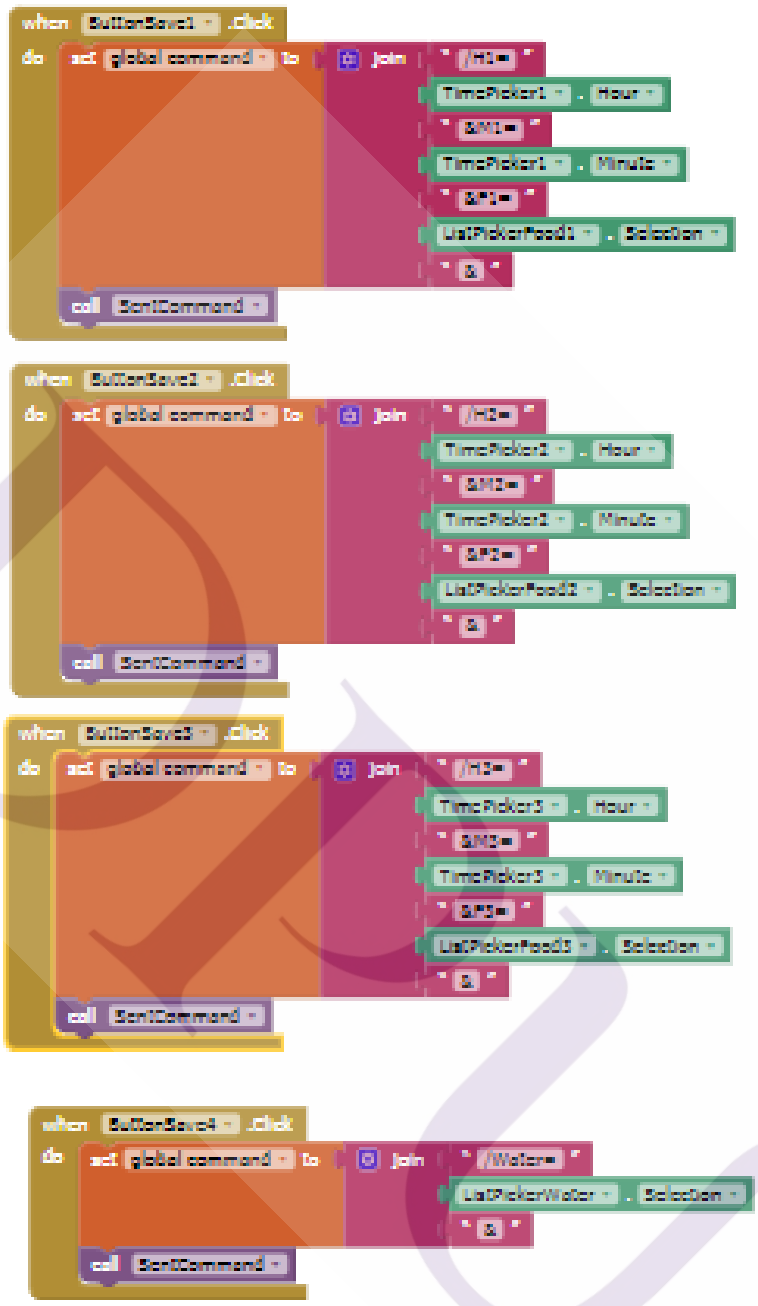


ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานการตั้งค่าปริมาณการให้อาหารปลาในแต่ละช่วงเวลา

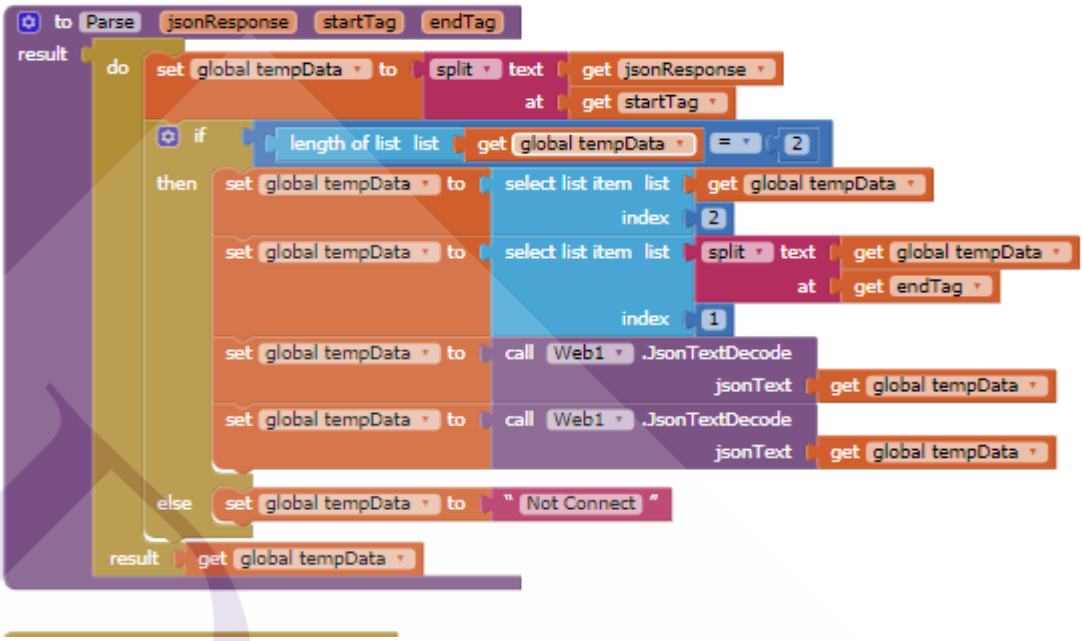
ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานกำหนดค่าตัวแปรปริมาณการให้อาหารปลาแต่ละช่วงเวลาและความขุ่นของน้ำภายในตู้ปลา



ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานกำหนดค่าตัวแปรปริมาณการให้อาหารปลาแต่ละช่วงเวลาและความขุ่นของน้ำภายในตู้ปลา (ต่อ)



ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานกำหนดค่าตัวแปรและบันทึกข้อมูลเวลาในแต่ละช่วงเวลาในการให้อาหารปลาและส่งค่าไปยังฟังชั่น Sent Command



ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นการทำงานการโชว์ข้อมูลต่างๆ บนแอพพลิเคชั่นไปยัง Webserver

ภาพแสดงข้อมูลการเขียนโปรแกรมฟังชั่นปุ่มการทำงานออกจากแอพพลิเคชั่น

ประวัติผู้เขียน

ชื่อ-นามสกุล	นาย โสภณวิชญ์ เขียวคำจีน
ประวัติการศึกษา	พ.ศ.2556 หลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมโทรคมนาคมและเครือข่ายคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์
สถานที่ทำงาน	บริษัท ทีโอทีจำกัด มหาชน ถนนแจ้งวัฒนะ เขตหลักสี่ แขวงทุ่งสองห้อง จังหวัดกรุงเทพมหานคร
ผลงานทางวิชาการ	สกอร์บอร์ดไร้สายเคลื่อนที่ (Movable Wireless Scoreboard) โครงงานศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมโทรคมนาคมและเครือข่ายคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยธุรกิจบัณฑิตย์ ปีการศึกษา 2556