

การประยุกต์ใช้เทคโนโลยี Internet of Things ในการควบคุม

ระบบส่องสว่างสำหรับบ้านอัจฉริยะ

An Application of Internet of Things Technology to Control the Illumination Systems in Smart Home

เจษฎา ขจรฤทธิ์¹, ปิยนุช ชัยพรแก้ว², หนึ่งฤทัย เอ็งฉ้วน³

วิทยาลัยนวัตกรรมการด้านเทคโนโลยีและวิศวกรรมศาสตร์, มหาวิทยาลัยธุรกิจบัณฑิตย์

110/1-4 ถนนประชาชื่น หลักสี่ กรุงเทพฯ 10210

jesada.kaj@dpu.ac.th¹, piyanuch.chw@dpu.ac.th², nuengruethai_a@fiber1.co.th³

ABSTRACT – Presently, the Internet of things technology has become an important role in modern human life. The technology allows intelligent devices to connect to the Internet and thus enhance the ability to access and control of those devices. Smart home is one of innovative product of this technology. This paper presents a prototype system to control the illumination systems from smart phones by using Internet of things technology. The implemented system consists of android application, NETPIE service, and microcontroller unit. Users are able to control illumination devices in the house through smart phones from anywhere that the Internet is available. The android application has the features of touch screen control and voice control. This implemented prototype shows the potential for further development of smart home devices for Thailand 4.0 policy.

KEY WORDS – Internet of Things, Smart Home, NETPIE Service, Android Application, Arduino Microcontroller

บทคัดย่อ – ในปัจจุบันเทคโนโลยี Internet of Things เข้ามามีบทบาทในชีวิตประจำวันของมนุษย์มากขึ้น เทคโนโลยีดังกล่าวเชื่อมโยงอุปกรณ์อัจฉริยะต่างๆ ผ่านระบบอินเทอร์เน็ต ทำให้การควบคุมอุปกรณ์มีความเป็นอิสระมากขึ้น บ้านอัจฉริยะเป็นผลผลิตหนึ่งของการประยุกต์ใช้เทคโนโลยีนี้ ในงานวิจัยชิ้นนี้ผู้เขียนได้พัฒนาระบบต้นแบบการควบคุมระบบส่องสว่างในครัวเรือนจากสมาร์ทโฟน ระบบดังกล่าวประกอบด้วยสามส่วนได้แก่ แอปพลิเคชัน Android, บริการ NETPIE และ หน่วยควบคุมไมโครคอนโทรลเลอร์ ผู้ใช้สามารถควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านสมาร์ทโฟนได้จากทุกที่ที่สามารถเข้าถึงระบบอินเทอร์เน็ต การควบคุมสามารถทำได้ทั้งระบบทัชสกรีนและการสั่งงานด้วยเสียง ผลงานวิจัยชิ้นนี้เป็นต้นแบบเพื่อนำไปสู่การพัฒนาผลิตภัณฑ์สำหรับบ้านอัจฉริยะและเพื่อตอบโจทย์ความต้องการในยุคไทยแลนด์ 4.0

คำสำคัญ – Internet of Things, บ้านอัจฉริยะ, บริการ NETPIE, แอปพลิเคชัน Android, ไมโครคอนโทรลเลอร์ Arduino

1. บทนำ

ในปัจจุบันเทคโนโลยี Internet of Things (IoT) [1] ได้เข้ามามีบทบาทในชีวิตประจำวันของผู้คนมากขึ้น การเชื่อมโยงสิ่งต่างๆ เข้าสู่อินเทอร์เน็ตทำให้การควบคุมอุปกรณ์ต่างๆ เป็นไปอย่างมีประสิทธิภาพมากขึ้น เช่น การควบคุมเครื่องใช้ไฟฟ้าต่างๆ ภายในครัวเรือนหรือที่ทำงานผ่านเครือข่ายอินเทอร์เน็ต เป็นที่น่าสังเกตว่าลักษณะการเข้ามาของเทคโนโลยี IoT สำหรับผู้บริโภคหรือผู้ประกอบการทั่วไปมักจะอยู่ในรูปแบบของ บ้านอัจฉริยะ (Smart Home) หรือ ฟาร์มอัจฉริยะ (Smart Farming) เป็นต้น [2]

ในเอกสารฉบับนี้ ผู้เขียนได้นำเสนอผลการพัฒนาผลิตภัณฑ์ IoT สำหรับบ้านอัจฉริยะเพื่อตอบโจทย์ความต้องการในยุคดิจิทัล ผู้เขียนได้พัฒนาชุดซอฟต์แวร์และฮาร์ดแวร์สำหรับการควบคุมระบบส่องสว่างภายในบ้านจากสมาร์ตโฟนผ่านระบบอินเทอร์เน็ต โดยเนื้อหาของเอกสารมีดังต่อไปนี้ ข้อที่ 2 กล่าวถึงเทคโนโลยีที่เกี่ยวข้อง ข้อที่ 3 กล่าวถึงสถาปัตยกรรมของระบบ ข้อที่ 4 แสดงผลการทำงานของระบบที่พัฒนาขึ้น และ ข้อที่ 5 เป็นบทสรุป

2. เทคโนโลยีที่เกี่ยวข้อง

เนื้อหาในส่วนนี้กล่าวถึงเทคโนโลยีหรือนิยามเบื้องต้นที่เกี่ยวข้องกับงานวิจัยชิ้นนี้ ดังต่อไปนี้

2.1 Internet of Things

Internet of Things คือเทคโนโลยีที่เชื่อมต่อสิ่งของหรืออุปกรณ์ต่างๆ เข้าสู่ระบบอินเทอร์เน็ต [3] สิ่งของต่างๆ เช่น รถยนต์ ตู้เย็น โทรทัศน์ สามารถเชื่อมโยงและสื่อสารกันได้ ด้วยเทคโนโลยีนี้ทำให้ผู้ใช้ทั่วไปสามารถเข้าถึงและควบคุมสิ่งของต่างๆ จากในบ้าน ที่ทำงาน หรือที่ใดๆ ซึ่งเป็นการยกระดับความสะดวกสบายในชีวิตประจำวัน ยกตัวอย่างเช่น เจ้าของบ้านสามารถเปิดเครื่องปรับอากาศภายในบ้านก่อนกลับบ้าน เป็นต้น

เทคโนโลยีทางด้าน IoT ที่นำมาประยุกต์ใช้ในปัจจุบันนี้ แบ่งเป็น 3 ประเภท [4] ประเภทแรกคือเทคโนโลยีที่ทำให้สรรพสิ่งสามารถรับรู้ข้อมูล เมื่อมีการปฏิสัมพันธ์หรือเข้าใกล้ เช่น เซ็นเซอร์ (sensor) ซึ่งทำหน้าที่ตรวจสอบและตอบสนองต่ออินพุต (input) ต่างๆ โดยอินพุตมีหลายประเภท ได้แก่ แสง

การเคลื่อนไหว ความชื้น ความดัน เป็นต้น หลังจากเซ็นเซอร์ได้รับอินพุตจะมีการสร้างเอาต์พุต (output) เป็นสัญญาณโดยสัญญาณที่สร้างขึ้นจะถูกแปลงไปเป็นผลลัพธ์ที่มนุษย์สามารถเข้าใจได้

ประเภทที่สองคือเทคโนโลยีที่ทำให้สรรพสิ่งสามารถติดต่อสื่อสารกันได้ เช่น ระบบฝังตัว (Embedded System) ที่ฝังอยู่ในเครื่องมืออิเล็กทรอนิกส์หรือเครื่องใช้ไฟฟ้า เพื่อให้อุปกรณ์เหล่านั้นสามารถทำตามคำสั่งได้ โดยคำสั่งต่างๆ จะถูกเขียนในชิป (Chip) หรือไมโครโพรเซสเซอร์ (Microprocessor) ที่ถูกฝังไว้ใน embedded System นอกจากนี้ เทคโนโลยีที่ทำให้สรรพสิ่งสามารถติดต่อสื่อสารกันได้ ยังรวมถึงการสื่อสารแบบไร้สายที่ใช้พลังงานต่ำ เช่น Zigbee, Bluetooth, Low Power Wide Area Network (LPWAN) เป็นต้น ซึ่งเป็นเทคโนโลยีที่ทำให้เครือข่ายเชื่อมต่อในระยะทางไกลได้โดยใช้พลังงานต่ำ ซึ่งเหมาะกับการสื่อสารระหว่าง IoT

ประเภทที่สามคือเทคโนโลยีที่ช่วยให้สรรพสิ่งประมวลผลข้อมูล เช่น การประมวลผลแบบคลาวด์ (Cloud Computing) เป็นบริการที่ให้บริการประมวลผล หน่วยจัดเก็บข้อมูล และโปรแกรมออนไลน์ต่างๆ จากผู้ให้บริการ อีกเทคโนโลยีหนึ่งที่เป็นที่นิยม คือ การวิเคราะห์ข้อมูลขนาดใหญ่ (Big Data Analytics) เป็นกระบวนการวิเคราะห์ข้อมูลขนาดใหญ่เพื่อค้นหารูปแบบความเชื่อมโยงของข้อมูลเหล่านั้น ผลจากการวิเคราะห์สามารถนำไปสู่การวางแผนการดำเนินงานธุรกิจได้อย่างมีประสิทธิภาพ

จากการคาดการณ์ในอีก 5 ปีข้างหน้า อุปกรณ์ที่เกี่ยวข้องกับชีวิตประจำวันจะสามารถเชื่อมต่อกับระบบอินเทอร์เน็ตเพิ่มมากขึ้นกว่า 300% (หรือประมาณกว่าแสนล้านชิ้น) อุปกรณ์ sensor ต่างๆ ที่ติดตั้งในเครื่องใช้ที่เกี่ยวข้องกับชีวิตประจำวันมีราคาลดลงไม่น้อยกว่า 80 - 90% จากราคาในปัจจุบัน และยังรวมไปถึงส่วนของระบบเครื่องกลไฟฟ้าจุลภาค (Micro Electromechanical Sensors - MEMS) อีกด้วย [5][6]

2.2 บ้านอัจฉริยะ

ในปี 2003 Housing Learning & Improvement Network (<http://www.housinglin.org.uk/>) ได้ตีพิมพ์คำจำกัดความของคำว่า บ้านอัจฉริยะ (Smart Home) ว่าเป็นเครือข่ายการสื่อสารของที่อยู่อาศัยที่ถูกรวมเข้าด้วยกัน เพื่อเชื่อมต่อเครื่องใช้ไฟฟ้า การบริการ การตรวจตรา รวมทั้งสามารถเข้าถึงและควบคุมอุปกรณ์

ต่างๆ ได้ (โดยที่การควบคุมหมายถึงการควบคุมทั้งที่เกิดจากภายในหรือภายนอกที่อยู่อาศัย) [7] Smart Home มีองค์ประกอบ 3 ประการ คือ

- *Smart Home Networks* คือระบบเครือข่ายพื้นฐานของ smart home อาจเป็นการเชื่อมต่อแบบมีสาย เช่น power line system หรือ bus line และการเชื่อมต่อแบบไร้สาย เช่น คลื่นความถี่วิทยุ (Radio Frequency) และ สัญญาณอินฟราเรด (Infrared)
- *Intelligent Control Systems* คือระบบควบคุมอัจฉริยะเพื่อเชื่อมต่อระหว่างเทคโนโลยีที่แตกต่างกันของอุปกรณ์อิเล็กทรอนิกส์ภายในบ้าน และเป็นเสมือน gateway เพื่อเชื่อมต่อกับบริการที่อยู่ภายนอกบ้าน
- *Home Automation Devices* คืออุปกรณ์อิเล็กทรอนิกส์ภายในบ้านที่สามารถเชื่อมโยงกันได้ ตัวอย่างเช่น ตู้เย็นอัจฉริยะที่สามารถบอกวันหมดอายุของอาหาร โซฟาอัจฉริยะที่สามารถปรับความอ่อนนุ่มตามความต้องการของผู้ใช้ ห้องน้ำอัจฉริยะที่สามารถควบคุมอุณหภูมิ เสียง และแสง ประต้อัจฉริยะคือประตูที่สามารถเปิดปิดให้เฉพาะบุคคลที่ได้จดจำใบหน้าไว้ หุ่นยนต์อัจฉริยะที่สามารถทำงานตามคำสั่งได้ เป็นต้น

สรุปโดยรวมคือ smart home จะต้องมีโครงสร้าง 3 ส่วนโดยเริ่มจากอุปกรณ์ที่เป็น smart devices ที่สามารถเชื่อมโยงเป็นเครือข่ายได้ซึ่งเรียกว่า smart home network ซึ่งจะถูกรวบรวมโดยส่วนควบคุมหลักที่เกิดจากการเขียนโปรแกรมให้อุปกรณ์เหล่านั้นสามารถทำงานตามความต้องการได้ ระบบดังกล่าวเรียกว่า intelligent control system

2.3 บริการ NETPIE

ข้อมูลจาก Network Platform for Internet of Everything [8] เป็น Cloud Platform ที่ถูกออกแบบและพัฒนาขึ้นเพื่ออำนวยความสะดวกสำหรับนักพัฒนาในการสื่อสารระหว่างอุปกรณ์ในเครือข่าย IoT NETPIE ช่วยให้อุปกรณ์สามารถสื่อสารกันได้ โดยผู้พัฒนาไม่ต้องสนใจว่าอุปกรณ์นั้นจะอยู่ที่ใด ทั้งในด้าน physical และ logical นักพัฒนาเพียงแค่นำไลบรารี NETPIE ไปติดตั้งในอุปกรณ์ NETPIE จะรับหน้าที่ดูแลการเชื่อมต่อให้ทั้งหมดไม่ว่าอุปกรณ์นั้นจะอยู่ในเครือข่ายชนิดใด ลักษณะใด หรือแม้กระทั่งเคลื่อนย้ายไปอยู่ที่ใด [8]

NETPIE ถูกออกแบบให้มี Authorization/Access Control ในระดับความละเอียดสูง กล่าวคือผู้พัฒนาสามารถออกแบบรายละเอียดได้เองทั้งหมด เช่น สิ่งใดมีสิทธิ์คุยกับสิ่งใด สิ่งใดมีสิทธิ์หรือไม่มีสิทธิ์ในการอ่านหรือเขียนข้อมูล และสิทธิ์เหล่านี้จะมีอายุเท่าใดหรือถูกเพิกถอนภายใต้เงื่อนไขใด เป็นต้น [8]

NETPIE มีสถาปัตยกรรมเป็น Cloud อย่างแท้จริงในทุกๆ ระดับของระบบ ทำให้เกิดความยืดหยุ่นและความคล่องตัวสูงในการเพิ่มขยาย นอกจากนี้โมดูลต่างๆ ยังถูกออกแบบให้ทำงานแยกจากกัน (Loose Coupling) และสื่อสารกันด้วยวิธี asynchronous messaging ช่วยให้แพลตฟอร์มมีความน่าเชื่อถือสูง สามารถนำไปพัฒนาต่อเติมได้ง่ายผู้พัฒนาไม่จำเป็นต้องกังวลกับการขยายตัวเพื่อรับโหลดที่เพิ่มขึ้นในระบบอีกต่อไป [9]

นอกจากนี้ NETPIE สามารถดูแลและซ่อมแซมตัวเองได้โดยอัตโนมัติ โดยไม่ต้องพึ่งผู้ดูแลระบบ [10] การบริหารจัดการระบบเป็นแบบ plug-and-play ไม่ต้องปรับแต่งในฝั่งอุปกรณ์ NETPIE มีไลบรารี client ที่เรียกว่า microgear ทำหน้าที่สร้าง ดูแล และรักษาความปลอดภัยของทางการสื่อสารระหว่างอุปกรณ์กับ NETPIE ในปัจจุบันมี microgear สำหรับ OS และ embedded board เกือบทุกชนิด เช่น Windows, Android, Apple, Ubuntu, Raspberry Pi และ Arduino เป็นต้น

อย่างไรก็ตาม การทำงานของ NETPIE ยังมีข้อจำกัดบางประการ เช่น มีการกำหนดอัตราการเขียน Feed ต่อ API Key ไว้ที่ 4 ครั้งในเวลาประมาณ 60 วินาที รวมถึง โพรโทคอลที่ใช้ยังไม่ครอบคลุมทุกชนิด

NETPIE รองรับโพรโทคอล MQTT ผ่านไลบรารีของ NETPIE ที่เรียกกันว่า microgear ในแอปพลิเคชันต่างๆ ผู้ใช้สามารถสร้าง application key ได้ไม่จำกัด โดย key ดังกล่าวแบ่งออกเป็น 2 ชนิดได้แก่ device key และ session key ซึ่งมีข้อแนะนำในการใช้งานดังนี้

Device key เหมาะสำหรับใช้กับอุปกรณ์ที่จะไม่นำมาแก้ไขค่า key อีก เช่นในกรณีงานวิจัยนี้คือชุดไมโครคอนโทรลเลอร์ ESP8266 ผู้วิจัยแนะนำว่าควรใช้ device key 1 ค่า ต่ออุปกรณ์ 1 ตัว เพราะจะไม่สามารถนำ key เดียวกันมาเชื่อมต่อกับ NETPIE พร้อมกันได้

Session key เหมาะสำหรับใช้กับอุปกรณ์ที่มีการใช้งานไม่ถาวรเช่นแอปพลิเคชัน android หรือ เบราวเซอร์ เมื่อยกเลิกการเชื่อมต่อ ตัวตนของอุปกรณ์จะถูกลบทิ้ง และ NETPIE จะสร้าง

session key ตัวใหม่จาก key ตัวเดิมเมื่อมีการเชื่อมต่อกลับเข้ามาอีกครั้ง

2.4 MQTT Protocol

Message Queuing Telemetry Transport (MQTT) [11] เป็นโพรโทคอล ที่ออกแบบมาเพื่อการเชื่อมต่อแบบ M2M (Machine to Machine) หรือระหว่างอุปกรณ์ที่สนับสนุนเทคโนโลยี IoT โพรโทคอลตัวนี้มีโหนดการทำงานที่เบา (Light Weight) เหมาะสำหรับอุปกรณ์ปลายทางที่มีขนาดเล็กและพลังงานจำกัด หรือในการสื่อสารระยะไกลที่ต้องการใช้งานแบนด์วิดธ์อย่างมีประสิทธิภาพ

MQTT มีหลักการทำงานแบบ publisher/subscriber คล้ายกับที่ใช้ใน web Services ที่ต้องใช้ web Servers เป็นตัวกลางระหว่างคอมพิวเตอร์ของผู้ใช้ แต่ต่างกันตรงที่ MQTT จะใช้ตัวกลางที่เรียกว่า brokers ทำหน้าที่จัดการคิวรับ-ส่งข้อมูลระหว่างอุปกรณ์ทั้งในส่วนที่เป็น publishers และ subscribers

โมเดลการสื่อสารของโพรโทคอล ประกอบด้วย 2 ส่วน คือ client และ broker [10] ในส่วนของ broker นั้นเป็นตัวกลางในการรับส่งข้อความระหว่าง client วิธีการกำหนดเส้นทาง (Routing) จะดำเนินการตามหัวข้อ (Topic) โดย client จะ subscribe ในหัวข้อที่ตนเองต้องการ จากนั้น broker จะส่งข้อความที่ถูก publish และสอดคล้องกับหัวข้อนั้นๆ กลับไปยัง client ทำให้ client สามารถส่งข้อความหากันได้โดยไม่ต้องรู้จักกัน การทำงานในลักษณะดังกล่าว ทำให้การขยายเครือข่ายเป็นไปได้ง่าย หน้าที่ที่สำคัญอีกอย่างหนึ่งของ broker คือ การรักษาความปลอดภัยของ client ทั้งในแง่ของการพิสูจน์ตัวตน (Authentication) และ การขออนุญาตสิทธิ์ (Authorization) โดยในการพิสูจน์ตัวตนจะใช้ username และ password ของ client ในการเข้าถึง broker ส่วนการขออนุญาตสิทธิ์การเข้าถึง broker จะอ้างอิงข้อมูลประจำตัว (Credentials) ของอุปกรณ์

ในส่วนของ client เป็นอุปกรณ์ที่ทำหน้าที่เป็น publisher หรือ subscriber หรือ ทั้ง publisher และ subscriber ในเวลาเดียวกัน โดยอุปกรณ์ดังกล่าวจะต้องสามารถรันไลบรารี MQTT client ได้ ซึ่งไลบรารีประเภทนี้จัดเป็นไลบรารีขนาดเล็ก เนื่องจากการทำงานหลักจะเป็นหน้าที่ของ broker ทำให้ไลบรารีประเภทนี้เหมาะกับอุปกรณ์ที่มีทรัพยากรจำกัด และสามารถติดตั้งได้ง่าย โดย client จะติดต่อกับ broker ผ่านการเชื่อมต่อแบบ TCP ดังนั้น client จะต้องเปิดการเชื่อมต่อไว้

ตลอด ในกรณีที่ขาดการเชื่อมต่อ broker จะเก็บข้อความไว้จนกว่ามีการเชื่อมต่ออีกครั้ง

จะเห็นได้ว่าการทำงานแบบ MQTT มีประสิทธิภาพมากกว่า HTTP (request / response) เนื่องจากการทำงานแบบ MQTT broker จะส่งข้อความไปยัง client ตามเหตุการณ์เท่านั้น แต่การทำงานแบบ HTTP client จะต้องคอยสุ่มถามข้อมูลเป็นระยะ ทำให้เกิดความสิ้นเปลืองของทรัพยากรโดยไม่จำเป็น

3. สถาปัตยกรรมและองค์ประกอบของระบบ

ในการพัฒนาผลิตภัณฑ์เกี่ยวกับบ้านอัจฉริยะ นั้นต้องใช้ องค์ประกอบหลายส่วนทำงานร่วมกัน ไม่ว่าจะเป็นในส่วนของ ฮาร์ดแวร์ ซอฟต์แวร์ หรือเครือข่ายอินเทอร์เน็ต สำหรับ ภาพรวมของสถาปัตยกรรมระบบของงานวิจัยชิ้นนี้แสดงในรูปที่ 1 โดยมีองค์ประกอบหลักดังต่อไปนี้

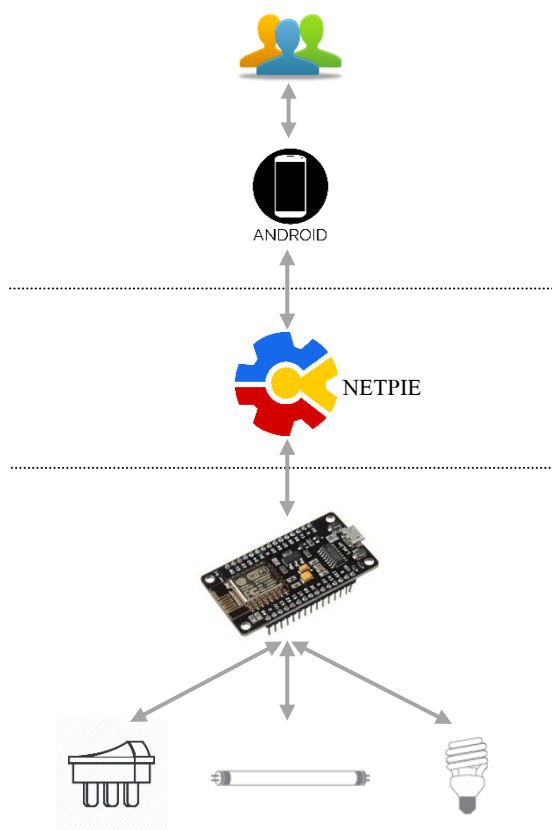
ส่วนแรกคือแอปพลิเคชันบนระบบ android ส่วนนี้ทำหน้าที่รับคำสั่งจากผู้ใช้เพื่อใช้ควบคุมอุปกรณ์ IoT แอปพลิเคชันจะทำงานผ่านไลบรารีเฉพาะที่เรียกว่า android microgear ซึ่งทำหน้าที่เป็นตัวกลางในการเชื่อมต่อแอปพลิเคชันเข้ากับบริการของ NETPIE

ส่วนที่สองคือบริการ NETPIE ส่วนนี้ทำหน้าที่เป็นตัวกลางในการเชื่อมต่อและแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชัน android กับ IoTs microcontroller เหตุผลที่งานวิจัยนี้เลือกใช้ บริการ NETPIE เนื่องจากสามารถใช้งานร่วมกับไลบรารีของ อุปกรณ์ประเภท android ได้เป็นอย่างดี

ส่วนสุดท้ายคือ IoT microcontrollers ส่วนนี้ทำหน้าที่รับคำสั่งจากแอปพลิเคชัน android ผ่านบริการ NETPIE แล้วนำคำสั่งไปควบคุมอุปกรณ์ประเภท relay Switch การเขียนชุดคำสั่งให้กับไมโครคอนโทรลเลอร์ทำได้โดยใช้โปรแกรม arduino IDE และเรียกใช้ไลบรารีเฉพาะที่เรียกว่า arduino microgear ในการสื่อสารกับบริการ NETPIE

3.1 การใช้บริการ NETPIE

ตามที่ได้อธิบายไว้ในส่วนที่สองว่า NETPIE เป็นบริการ IoT ในลักษณะคลาวด์แพลตฟอร์ม (Cloud Platform) ดังนั้นการใช้งาน NETPIE นักพัฒนาจะต้องทำการสมัครสมาชิกเพื่อขอใช้บริการก่อน [8]



รูปที่ 1: ภาพรวมของสถาปัตยกรรมระบบ

รูปที่ 2 แสดงตัวอย่างผลการสร้าง application และการสร้าง key ขึ้นมาใช้งาน โดยมีทั้ง device key สำหรับใช้กับ IoT microcontroller และ session key สำหรับแอปพลิเคชัน android ในแต่ละ key จะมี ค่า key และ secret ให้กับอุปกรณ์แต่ละตัว ก่อนจะเริ่มใช้งาน

3.2 การออกแบบแอปพลิเคชัน Android

แอปพลิเคชัน Android นี้จะเชื่อมต่อกับบริการ NETPIE อัตโนมัติเมื่อเปิดใช้งาน ซึ่งตัว microgear ที่ทำงานบนโปรโตคอล MQTT จะติดต่อกับ NETPIE broker ซึ่งจะใช้ OAuth เข้ามาช่วยในการสร้างข้อมูลสำคัญที่ใช้เชื่อมต่อกับ MQTT broker ดังนั้น MQTT username, MQTT password และ MQTT clientId หน้าที่ของ OAuth (OAuth 1.0a) คือการเชื่อมต่อกับ NETPIE ให้สามารถใช้ไลบรารี MQTT ได้

การควบคุมหรืออินเทอร์เฟซบนแอปพลิเคชันแบ่งออกเป็น 2 ส่วน (รูปที่ 3) ส่วนแรกคือ Smart Light มีลักษณะการทำงาน

แบบสวิตช์ เปิด/ปิด มีไว้สำหรับควบคุมการเปิด/ปิดสวิตช์รีเลย์แต่ละตัวเพื่อควบคุมแหล่งจ่ายไฟของอุปกรณ์ส่องสว่าง

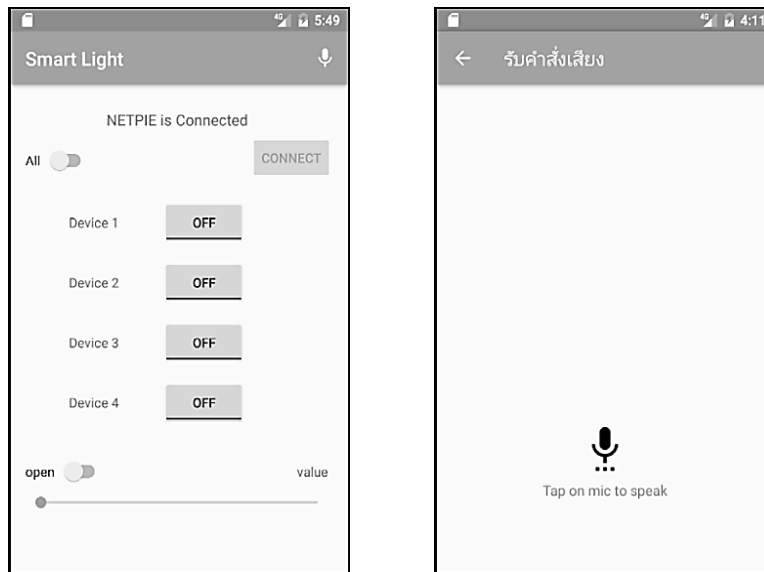
ส่วนที่สองเป็นการควบคุมด้วยเสียง [10] ผู้ใช้สามารถเข้าถึงหน้าจอนี้ได้เมื่อมีสัมผัสที่เครื่องหมาย Voice Control จาก action Bar ของหน้า Smart Light ระบบจะรับคำสั่งเสียงจากผู้ใช้แล้วไปควบคุม IoT ไมโครคอนโทรลเลอร์สำหรับการเปิด/ปิดรีเลย์สวิตช์แต่ละตัวได้

รูปที่ 4 แสดงตัวอย่างชุดคำสั่งบางส่วนที่สำคัญในแอปพลิเคชันมี ส่วนที่ 1 แสดงการกำหนดค่าตัวแปรเริ่มต้น ได้แก่ application ID, application key (session key) และ application secret ส่วนที่ 2 คือ *MicrogearCallBack* แอปพลิเคชันที่รันบน microgear จะมีการทำงานในแบบ event driven กล่าวคือเป็นการทำงานตอบสนองต่อ event ต่างๆ ด้วยการเขียน callback function ขึ้นมารองรับและจะทำงานก็ต่อเมื่อมีการเชื่อมต่ออินเทอร์เน็ต

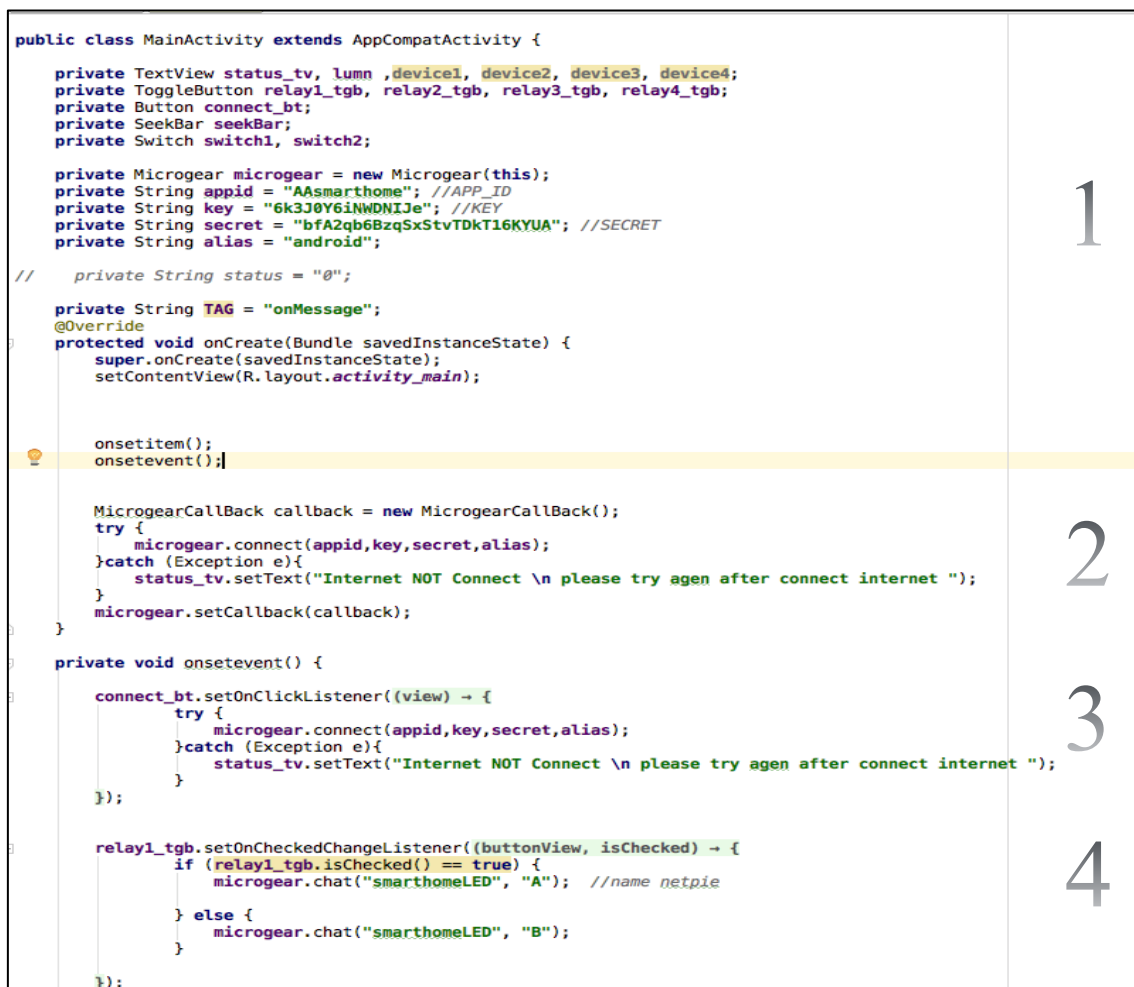
ส่วนที่ 3 คือการเชื่อมต่อ event connected จะเกิดขึ้นเมื่อ microgear เชื่อมต่อกับ NETPIE สำเร็จ และ ส่วนที่ 4 คือการส่งข้อความ เมื่อมีการเชื่อมต่อ NETPIE สำเร็จแล้ว การ chat ของ microgear สามารถส่งข้อความแบบเจาะจงถึงผู้รับด้วยการระบุ name ที่ microgear เคยตั้งค่า alias ไว้

APPLICATION (APPIO)	
ExampleSmartHome	
2 % APPLICATION KEY	
[CREATE]	
Device Key	Session Key
APPLICATION KEY	
mobile1 [online]	
node1 [online]	
mobile1	
Key :	y5RUdFciznTqEoy
Secret :	pMuji5VRw1lvH4hNvetr6usPZ
REST API auth :	y5RUdFciznTqEoy:pMuji5VRw1lvH4hNvetr6usPZ
[RENAME] [CANCEL]	
node1	
Key :	1VkeIDHeVE10ikQ
Secret :	22NKYSTbLj5kH3IWHeHV5ZLjj
REST API auth :	1VkeIDHeVE10ikQ:22NKYSTbLj5kH3IWHeHV5ZLjj
[RENAME] [CANCEL]	

รูปที่ 2: ตัวอย่างการสร้าง Application Key ใน NETPIE



รูปที่ 3: อินเทอร์เฟซบนแอปพลิเคชัน Android



รูปที่ 4: ตัวอย่างชุดคำสั่งบางส่วนในแอปพลิเคชัน Android

รายละเอียดในส่วนการทำงานด้วยเสียงแสดงไว้ในรูปที่ 5ก และ 5ข ระบบ Voice Recognition รับข้อความเป็นภาษาไทย ถึงแม้ว่าเครื่องนั้นๆ จะใช้เป็นภาษาอังกฤษอยู่ก็ตาม สามารถกำหนดค่าเข้าไปใน intent ได้ เพื่อเลือกภาษาที่ต้องการเมื่อมีการแปลงจากเสียงเป็นข้อความผลลัพธ์ที่ได้นั้น

เป็น string array list อย่างไรก็ตามผลลัพธ์ไม่ได้แม่นยำ 100% เสมอไป ดังนั้นอันไหนที่น่าจะใกล้เคียงก็จะถูกส่งกลับมาด้วย ซึ่งเรียงลำดับตามความน่าจะเป็น array index ตัวที่ 0 ก็จะใกล้เคียงที่สุด [12]

```
private void promptSpeechInput() {
    Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
        RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, (new Locale("th-TH")));
    intent.putExtra(RecognizerIntent.EXTRA_PROMPT,
        "Say something...");

    try {
        startActivityForResult(intent, REQ_CODE_SPEECH_INPUT);
    } catch (ActivityNotFoundException a) {
        Toast.makeText(getApplicationContext(), "Sorry! Your device doesn't support speech input",
            Toast.LENGTH_LONG).show();
    }
}
```

รูปที่ 5ก: ตัวอย่างชุดคำสั่งสำหรับการสั่งการด้วยเสียง

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    switch (requestCode) {
        case REQ_CODE_SPEECH_INPUT: {
            if (resultCode == RESULT_OK && null != data) {
                ArrayList<String> result = data
                    .getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                txtSpeechInput.setText(result.get(0));

                microgear.chat("smarthomeLED", result.get(0));
            }
        }
    }
}

protected void onDestroy() {
    super.onDestroy();

    microgear.disconnect();
}

protected void onResume() {
    super.onResume();
    microgear.bindServiceResume();
}
```

รูปที่ 5ข: ตัวอย่างชุดคำสั่งสำหรับการสั่งการด้วยเสียง (ต่อ)

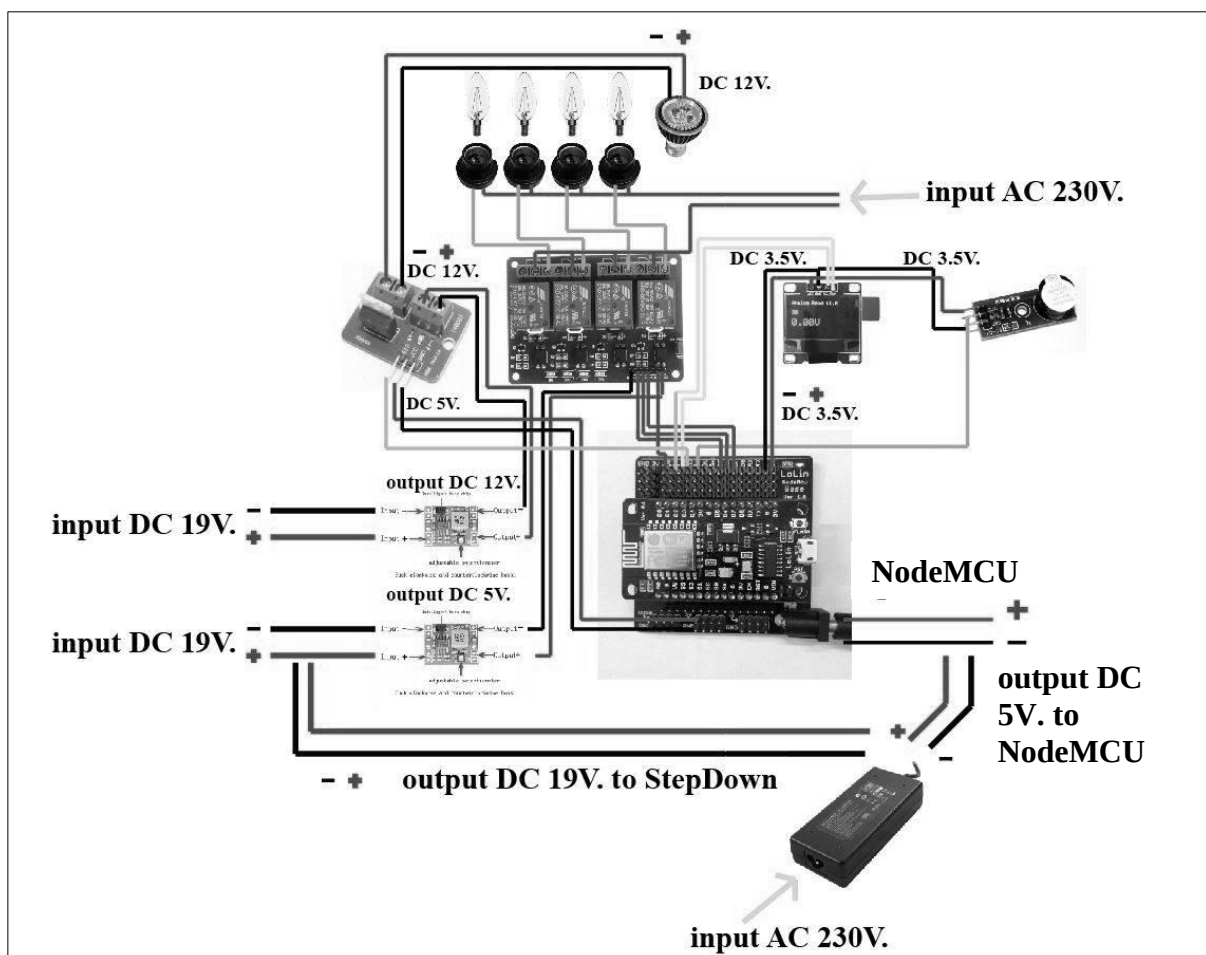
3.3 IoT Microcontroller Unit

ชุดควบคุม (IoT Microcontroller Unit) แสดงในรูปที่ 6 ในชุดควบคุมนี้ประกอบด้วยอุปกรณ์ดังต่อไปนี้

- AC/DC adapter ใช้แปลงไฟ AC 220 V. เป็น DC 19 V.
- Step Down adapter ใช้ลดแรงดันไฟฟ้าจาก DC 19 V. เป็น DC 5 V.
- NodeMCU เป็น IoT microcontroller หลักในการเชื่อมต่อกับ NETPIE และควบคุมอุปกรณ์ต่างๆ
- OLED display ใช้แสดงผลการทำงานของระบบ
- Sound buzzer ใช้สร้างสัญญาณเสียงเตือน
- 4-Channel relay switch สำหรับการเปิด/ปิดกระแสไฟฟ้า
- MOSFET relay switch สำหรับการเปิด/ปิดกระแสไฟฟ้าตามสัญญาณขับแบบ PWM

จากรูปที่ 6 ระบบทั้งหมดใช้ไฟเลี้ยงจาก AC/DC adapter โดยผ่าน step down adapter ที่แปลงจาก 19 V. เป็น 3.5 V. และ 5 V. ตามอุปกรณ์ที่ใช้ สัญญาณจาก nodeMCU เข้าควบคุม สวิตช์รีเลย์ 4 ตัว (On/Off Light) และสวิตช์ MOSFET รีเลย์ 1 ตัว (Dimmer Light) นอกจากนี้ยังมีสัญญาณจาก nodeMCU เพื่อการแสดงผลผ่าน OLED display และ sound buzzer แยกต่างหาก

สำหรับในชุดคำสั่งภาษา C ที่เป็นส่วนของการกำหนดตัวแปรเพื่อการเชื่อมต่อแสดงตัวอย่างดังรูปที่ 7 บรรทัดที่ 2 และ 3 เป็นการเรียกไลบรารีที่เกี่ยวข้อง บรรทัดที่ 5 และ 6 เป็นการกำหนดรหัสการเชื่อมต่อสัญญาณ WiFi บรรทัดที่ 8 ถึง 11 เป็นการกำหนดตัวแปรเพื่อการติดต่อกับ MQTT broker ของ NETPIE [13][14]



รูปที่ 6: IoT Microcontroller Unit

```

1. /* Include Packages */
2. #include <ESP8266WiFi.h>
3. #include <MicroGear.h>
4. /* WiFi Connection */
5. const char* ssid "JadeWiFi";
6. const char* password "123456";
7. /* MQTT Broker Connection*/
8. #define APPID "JadeSmartHome1";
9. #define GEARKEY "zZ7GFfxJuWJrHBI";
10. #define GEARSECRET "XJp1lzLDMvgdkfawL";
11. #define SCOPE "SmartHomeLED";

```

รูปที่ 7: ตัวอย่างการกำหนดตัวแปรเพื่อการเชื่อมต่อ

4. ผลลัพธ์ที่ได้จากการพัฒนา

รูปที่ 8 แสดงชุดต้นแบบของระบบที่พัฒนาขึ้นมา ผู้เขียนได้ทดสอบการใช้งานฟีเจอร์ต่างๆ โดยสามารถสรุปได้ตามตารางที่ 1

ตารางที่ 1. สรุปการทำงานของระบบ

ลำดับ	หน้าที่การทำงาน	ผลการทดสอบ
1	การเชื่อมต่อ NETPIE	เชื่อมต่ออัตโนมัติได้ทุกครั้งที่เปิดระบบ
2	การเปิด/ปิดไฟ	ทำงานได้ดี ไม่มีการ delay ของสัญญาณ (ไม่เกิน 5 วินาที)
3	การหรีไฟ	ทำงานได้ แต่ตอบสนองช้า (นานกว่า 5 วินาที) และ บางครั้งความสว่าง สอดคล้องกับสเกลที่กำหนดไว้
4	การสั่งงานด้วยเสียง	ทำงานได้ แต่มีเสถียรภาพน้อย ผู้ใช้อาจต้องพูดซ้ำหลายครั้ง

ข้อสังเกตแรกของการทดสอบ ระบบสามารถสั่งงานด้วยเสียงในการควบคุมการเปิด/ปิดหลอดไฟได้ แต่ยังไม่มีความแม่นยำ เช่น ผู้ใช้อาจต้องพูดมากกว่าหนึ่งครั้ง มีความเป็นไปได้

สูงว่าสาเหตุมาจากการทำงานของ Android กับภาษาไทยที่ยังมีขีดจำกัด

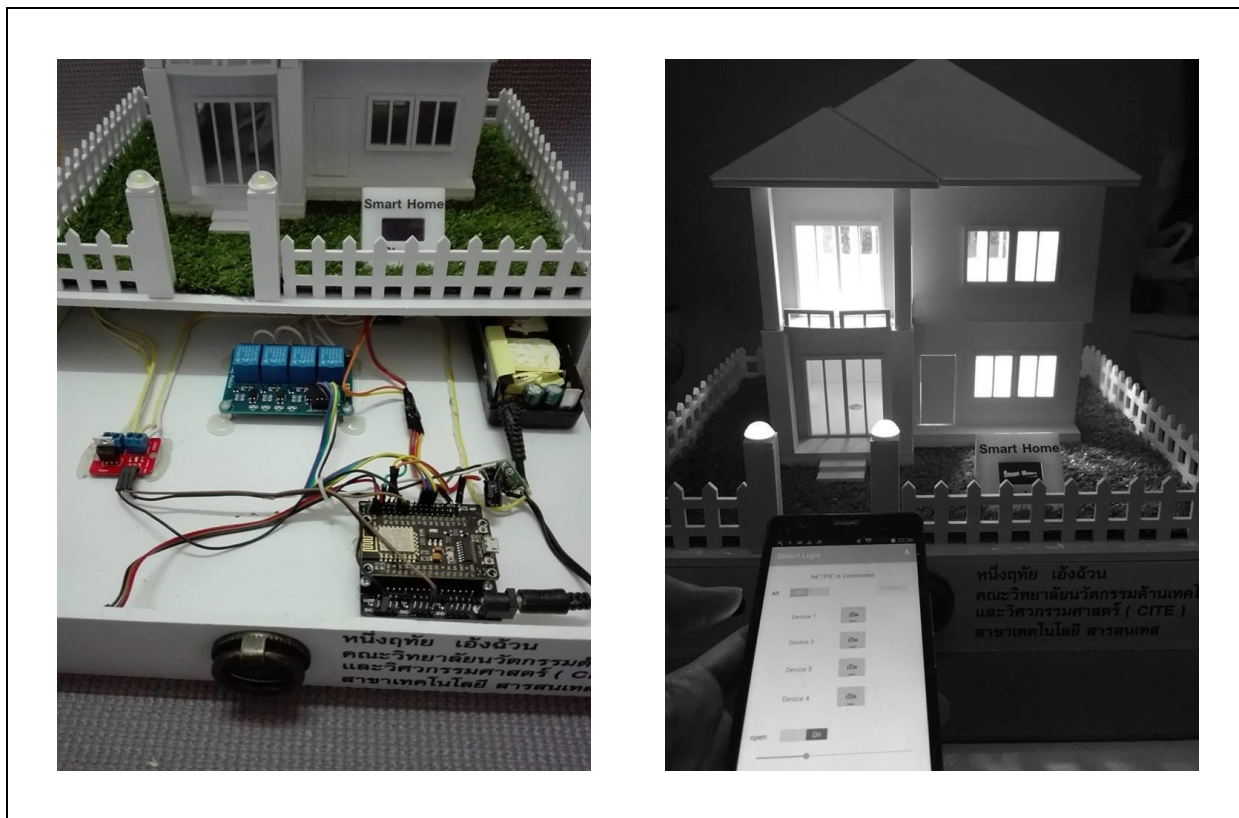
ข้อสังเกตที่สองของการทดสอบ ระบบการควบคุมการหรีไฟยังมีเสถียรภาพต่ำ มีผลให้ในขณะส่งค่าไปยังบริการ NETPIE แอปพลิเคชันเกิดการ disconnect ขึ้น สาเหตุเนื่องมาจากการอ่านค่าที่เร็วเกินไป ถึงแม้ว่าจะพบปัญหาในการทำงานอยู่บ้าง แต่ระบบก็สามารถตอบโจทยความต้องการของผู้ใช้ในภาพรวมได้ตามที่ออกแบบไว้

5. บทสรุป

เทคโนโลยี Internet of Things ได้เข้ามามีบทบาทในหลายสิ่งของสังคมยุคดิจิทัล ระบบบ้านอัจฉริยะ (Smart Home) เป็นอีกหนึ่งสิ่งที่ได้รับอิทธิพลจากเทคโนโลยี Internet of Things ในงานวิจัยชิ้นนี้ผู้วิจัยได้ศึกษา ออกแบบ และสร้างชุดต้นแบบสำหรับการควบคุมระบบไฟส่องสว่างในครัวเรือนผ่านเทคโนโลยีดังกล่าว ระบบทั้งหมดประกอบด้วยสามส่วน ได้แก่ (1) แอปพลิเคชัน android (2) บริการ NETPIE และ (3) ชุดกล่องควบคุมหลัก ผลที่ได้รับคือผู้ใช้สามารถควบคุมระบบไฟส่องสว่างจาก smartphone ในที่ใดก็ได้ที่มีการเชื่อมต่อกับอินเทอร์เน็ต อย่างไรก็ตาม ระบบการหรีไฟและการสั่งงานด้วยเสียงยังต้องมีการปรับปรุง งานที่ผู้วิจัยจะดำเนินการต่อไปคือการสร้างตัวผลิตภัณฑ์สำหรับความต้องการของตลาดจริง

เอกสารอ้างอิง

- [1] Internet of Thing (IoT) คืออะไร เข้าถึงได้จาก <http://www.mcuthailand.com/articles/iot/IOT.html> สืบค้นเมื่อ กันยายน 2560.
- [2] อรรถพล กันหเวก, ความหมายของ Smart Home เข้าถึงได้จาก <https://www.arm.co.th/Knowledge.Asp?id=2> สืบค้นเมื่อ กันยายน 2560.
- [3] Internet of Things. Available from: URL: https://en.wikipedia.org/wiki/Internet_of_things, Accessed September 30, 2017.



รูปที่ 8: ชุดต้นแบบผลิตภัณฑ์ที่ได้จากการพัฒนา

[4] ประภาพร กุลลิมรัตน์ชัย. (2559). Internet of Things:

แนวโน้มเทคโนโลยีปัจจุบันกับการใช้งานในอนาคต.

วารสารวิชาการมหาวิทยาลัยอีสเทิร์นเอเซีย ฉบับวิทยาศาสตร์และเทคโนโลยี, 10(1), หน้า 29-36.

[5] Strategic Man, เทคโนโลยีที่ 3: Internet of Things เข้าถึงได้จาก <http://www.strategic-man.com/articles/detail/19#>.

WO9eMWmGOUk สืบค้นเมื่อ กันยายน 2560.

[6] อลงกรณ์ พิมพ์พิณ, MEMS: Micro Electro Mechanical Systems เข้าถึงได้จาก <http://www.vcharkarn.com/varticle/41941> สืบค้นเมื่อ กันยายน 2560.

[7] Nicola King. SMART HOME – A DEFINITION. Available from: URL: https://www.housinglin.org.uk/_assets/Resources/Housing/Housing_advice/Smart_Home_-_A_definition_September_2003.pdf. Accessed September 30, 2017.

[8] NETPIE. Available from: URL: <https://netpie.io>. Accessed September 30, 2017.

[9] NETPIE: Internet of Things เข้าถึงได้จาก <https://www.nectec.or.th/innovation/innovation-software/netpie.html> สืบค้นเมื่อ กันยายน 2560.

[10] NETPIE. Available from: URL: <https://netpie.io/tutorial>. Accessed May 19, 2017.

[11] MQTT. Available from: MQTT.org. Accessed May 19, 2017.

[12] [Android Code] เปลี่ยนเสียงพูดให้กลายเป็นข้อความด้วย Voice Recognition เข้าถึงได้จาก <http://www.akexorcist.com/08/2013android-code-string-speech-recognizer.html> สืบค้นเมื่อ กันยายน 2560.

[13] การสร้าง MicroGear Library สำหรับ NETPIE (และ Android MicroGear) เข้าถึงได้จาก <http://archive.cmmaker>

club.com/2016/08/iot/netpie/how-to-create-microgear-library-android-studio/ สืบค้นเมื่อ กันยายน 2560.

[14] Smart Plug, Available from: URL: https://netpie.io/projects/smart_plug. Accessed September 30, 2017.