

การอธิบายบริการของอินเทอร์เน็ตของ สรรพสิ่งด้วยเอกซ์เอ็มแอล Service Description for Internet of Things using XML

ชัชวิน นามมัน (Chatchawin Namman)*

บทคัดย่อ

บทความนี้นำเสนอการอธิบายบริการของอินเทอร์เน็ตของสรรพสิ่งด้วยเอกสารเอกซ์เอ็มแอล เพื่อช่วยให้ผู้ใช้บริการสามารถเรียกใช้และเลือกเซอร์วิสมาทำงานร่วมกันได้ ผู้พัฒนาโปรแกรมสามารถสร้างส่วนติดต่อกับผู้ใช้ในการเรียกใช้เซอร์วิสของอินเทอร์เน็ตของสรรพสิ่งได้ในแบบอัตโนมัติ โดยได้นำเสนอตัวอย่างการทำงานผ่านอุปกรณ์ที่พัฒนาขึ้นด้วยบอร์ด Raspberry PI ที่มีเว็บเซอร์วิสแบบ RESTful ประกอบด้วยเซอร์วิสสำหรับเปิด/ปิดไฟฟ้า และเซอร์วิสสำหรับหรี่ไฟ และตัวอย่างการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์เพื่อทดสอบการสร้างส่วนติดต่อกับผู้ใช้และเรียกใช้เซอร์วิสจากอุปกรณ์ด้วยสมาร์ตโฟนผ่านการอ่านคิวอาร์โค้ดและการส่งงานด้วยเสียง ซึ่งแอปพลิเคชันที่พัฒนาขึ้นสามารถแสดงส่วนติดต่อกับผู้ใช้ได้ถูกต้อง และสามารถเรียกใช้งานได้ตามวัตถุประสงค์ นอกจากนี้ยังได้นำเสนอตัวอย่างการทดสอบการอธิบายและเรียกใช้เซอร์วิส และการประเมินความพึงพอใจที่มีต่อวิธีการอธิบายการเรียกใช้เซอร์วิส ซึ่งพบว่ามีความพึงพอใจในภาพอยู่ในระดับมากที่สุด

คำสำคัญ: อินเทอร์เน็ตของสรรพสิ่ง เว็บเซอร์วิส เอกซ์เอ็มแอล

Abstract

This paper presents a service description for the Internet of Things by using XML document. Which allows the requester to call the service more easily and can be chosen

them together. The developers can use XML document to build applications and user interfaces in dynamically. The test bed device was implemented by using Raspberry PI with a RESTful web service to creating service for dimmer and power the lamp. It also has developed an application for the android platform to experiment to create a user interface and run the service from a smartphone via QR code, and through the voice command. The test results showed that both applications can be generated the user interface correctly. And can be used as intended. It also presents the testing of describing and using services. The service description method was evaluated by the completion of a questionnaire. The highest level of satisfaction was found.

Keywords: Internet of Things, Web service, XML

1. บทนำ

อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) หรือ IoT [1] หมายถึงการที่สิ่งของสามารถสื่อสารและเชื่อมต่อกันผ่านโพรโทคอลของระบบเครือข่ายทั้งแบบมีสายและแบบไร้สาย ทำให้สามารถควบคุม ตรวจสอบการทำงาน บริหารจัดการ และสามารถระบุตัวตนของสิ่งของนั้นผ่านเครือข่ายท้องถิ่นหรือเครือข่ายอินเทอร์เน็ตได้ ทั้งนี้ฟังก์ชันในการทำงานของอินเทอร์เน็ตของสรรพสิ่ง อาจจะมีเพียงฟังก์ชันเดียวหรือมีหลากหลายฟังก์ชันก็ได้

มีการนำอินเทอร์เน็ตของสรรพสิ่งไปใช้ในหลายโดเมนตามความสนใจ ได้แก่เกษตรกรรม เกษตร สิ่งแวดล้อม และสังคม เช่น การจัดการห่วงโซ่อุปทาน การขนส่งและโลจิสติกส์ [2] ระบบอุตสาหกรรมในด้านต่างๆ [3] เทคโนโลยีทางการแพทย์และการดูแลสุขภาพ [4] การเกษตรและอาหาร [5] รวมถึงด้านอาคารและบ้านอัจฉริยะ [6] เป็นต้น

นอกจากนี้ยังมีการนำเอาแนวคิดของสถาปัตยกรรมเชิงบริการ (Service Oriented Architecture) หรือ SOA มาประยุกต์ใช้งานร่วมกับอินเทอร์เน็ตของสรรพสิ่ง [7] ซึ่ง SOA เป็นรูปแบบของการพัฒนาซอฟต์แวร์โดยมุ่งเน้นในการให้บริการโดยออกแบบซอฟต์แวร์ให้มีความยืดหยุ่นและข้อกำหนดน้อยที่สุดในการเรียกใช้ซอฟต์แวร์ ทำให้ซอฟต์แวร์สามารถถูกเรียกใช้จากแพลตฟอร์มและระบบปฏิบัติการใดๆ หรือ

* ภาควิชาคณิตศาสตร์ สถิติ และคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยอุบลราชธานี

* Department of Mathematics, Statistics, and Computer, Faculty of Science, Ubon Rachathani University.

ส่วนประกอบต่างๆ ของซอฟต์แวร์ โดย SOA มีส่วนประกอบ 3 ส่วนได้แก่ ผู้ให้บริการ (Service Provider) ซึ่งทำหน้าที่ในการให้บริการเซอร์วิสที่สามารถเรียกใช้งานได้โดยผู้ขอใช้บริการ (Service Requester) ซึ่งตัวแทนของผู้ให้บริการ (Service Broker) จะเป็นผู้รวบรวมรายชื่อ ที่อยู่และวิธีการในการใช้บริการเซอร์วิสจากผู้ให้บริการ ส่วนประกอบทั้ง 3 สามารถติดต่อกันผ่านฟังก์ชันพื้นฐานคือ การประกาศ (Publish) การค้นหา (Find) และการเรียกใช้ (Bind) โดยผู้ให้บริการจะทำการประกาศเซอร์วิสที่มีไปยังส่วนตัวแทนของผู้ให้บริการเพื่อทำการรวบรวม เมื่อผู้ขอใช้บริการต้องการใช้เซอร์วิสใด จะทำการค้นหาเซอร์วิสผ่านตัวแทนของผู้ให้บริการเมื่อพบเซอร์วิสที่ต้องการก็จะทำการเรียกใช้เซอร์วิสนั้นไปยังผู้ให้บริการ

เมื่อนำเอา SOA มาประยุกต์ใช้กับอินเทอร์เน็ตของสรรพสิ่ง โดยนิยามความหมายของเซอร์วิสว่าเป็นส่วนประกอบของฟังก์ชันการทำงานของอินเทอร์เน็ตของสรรพสิ่งที่ย่อยเป็นอิสระต่อกัน มีการประกาศและอธิบายการเรียกใช้ สามารถถูกเรียกใช้งานและนำกลับมาใช้ซ้ำเพื่อทำงานร่วมกับเซอร์วิสอื่นเพื่อสร้างเป็นฟังก์ชันการทำงานใหม่ๆ ได้ การรวมเซอร์วิสเหล่านี้ให้ทำงานร่วมกันจะนำมาซึ่งการรับรู้บริบทของสภาพแวดล้อม การปฏิสัมพันธ์โต้ตอบ และการทำงานร่วมกันระหว่างสิ่งของได้

อินเทอร์เน็ตของสรรพสิ่งถูกพัฒนาในหลายรูปแบบและมีหลาย โพรโทคอลที่สนับสนุนการทำงาน [8] โดยเว็บโพรโทคอลก็เป็นหนึ่งในนั้น ทำให้การพัฒนาอินเทอร์เน็ตของสรรพสิ่งในรูปแบบของการให้บริการเซอร์วิสที่ประยุกต์ใช้สถาปัตยกรรมเชิงบริการนั้นทำได้สะดวกยิ่งขึ้น อย่างไรก็ตามการให้บริการเซอร์วิสของอินเทอร์เน็ตของสรรพสิ่งผ่านเว็บโพรโทคอลนั้นหากผู้ขอใช้บริการไม่ทราบส่วนประกอบและวิธีการในการเรียกใช้บริการก็จะไม่สามารถขอใช้บริการได้ การกำหนดรูปแบบของการอธิบายเซอร์วิสจะช่วยให้ผู้ขอใช้บริการสามารถเรียกใช้เซอร์วิสจากอินเทอร์เน็ตของสรรพสิ่งได้ง่ายขึ้น สามารถการนำเซอร์วิสมาทำงานร่วมกัน และผู้พัฒนาโปรแกรมสามารถสร้างส่วนติดต่อกับผู้ใช้งานในการเรียกใช้เซอร์วิสจากอินเทอร์เน็ตของสรรพสิ่งได้ในแบบอัตโนมัติอีกด้วย

บทความนี้จะนำเสนอการอธิบายบริการของอินเทอร์เน็ตของสรรพสิ่งด้วยเอกซ์เอ็มแอล เพื่อช่วยให้ผู้ขอใช้

บริการและผู้พัฒนาโปรแกรมสามารถเรียกใช้และเลือกเซอร์วิสมาทำงานร่วมกันได้ง่ายขึ้น โดยในหัวข้อที่ 2 จะเป็นการอธิบายโครงสร้างของชื่อที่อยู่และการเรียกใช้เซอร์วิส หัวข้อที่ 3 จะกล่าวถึงโครงสร้างเอกสารเอกซ์เอ็มแอลเพื่อใช้สำหรับอธิบายเซอร์วิส หัวข้อที่ 4 เป็นตัวอย่างการพัฒนาอุปกรณ์ต้นแบบและแอปพลิเคชันสำหรับเรียกใช้เซอร์วิส และในหัวข้อที่ 5 เป็นการสรุป

2. โครงสร้างของชื่อที่อยู่และการเรียกใช้เซอร์วิส

สำหรับโครงสร้างของชื่อที่อยู่และการเรียกใช้เซอร์วิสนั้น ในบทความนี้ได้ใช้เว็บเซอร์วิสสำหรับพัฒนาเซอร์วิสของอุปกรณ์อินเทอร์เน็ตของสรรพสิ่ง ซึ่งเว็บเซอร์วิส [9] ถูกพัฒนาขึ้นเพื่อสนับสนุนการทำงานของโปรแกรมร่วมกันผ่านเว็บโพรโทคอล ที่มีพื้นฐานมาจากโครงสร้างของสถาปัตยกรรม SOA ซึ่งสามารถพัฒนาในแบบ SOAP [10] หรือ RESTful [11] ก็ได้ ทั้งนี้ในแบบ SOAP จะมีส่วนประกอบหลัก 3 ส่วน ได้แก่ SOAP ซึ่งเป็นโพรโทคอลสำหรับการแลกเปลี่ยนข้อมูลในสภาพแวดล้อมแบบกระจายศูนย์ WSDL ที่ใช้เป็นมาตรฐานในการอธิบายองค์ประกอบของเว็บเซอร์วิส และ UDDI ส่วนเว็บเซอร์วิสแบบ RESTful นั้นจะอาศัยหลักการพื้นฐานของเว็บโพรโทคอลที่มองว่าข้อมูลต่างๆ เป็นทรัพยากรที่สามารถเรียกใช้ได้ด้วยการส่งคำร้องขอ 4 แบบคือ GET, PUT, DELETE และ POST เพื่อให้ผู้บริการประมวลผลแล้วส่งข้อมูลกลับมาให้ผู้รับบริการเป็นข้อมูลกลับคืนมาในหลายรูปแบบเช่น JSON และ XML เป็นต้น เนื่องจากแบบ RESTful มีขั้นตอนในการประมวลผลและการเรียกใช้งานที่ไม่ซับซ้อน ทำให้เขียนโปรแกรมได้ง่ายและใช้ทรัพยากรทางด้านฮาร์ดแวร์ของระบบต่ำ [12] จึงเหมาะกับการพัฒนาเซอร์วิสสำหรับอินเทอร์เน็ตของสรรพสิ่งซึ่งมีขีดจำกัดในด้านฮาร์ดแวร์

การเรียกใช้บริการเว็บเซอร์วิสแบบ RESTful ผ่านเว็บโพรโทคอลของอุปกรณ์ อินเทอร์เน็ตของสรรพสิ่งนั้นจะเรียกใช้ผ่านโครงสร้างชื่อที่อยู่ (Uniform Resource Locator) หรือ URL ซึ่งเป็นการระบุถึงเซอร์วิสที่ต้องการขอใช้บริการ ส่วนประกอบของ URL นั้นมี 4 ส่วนได้แก่ Protocol, Authority, Path และ Parameter

ส่วนแรก Protocol จะเป็นการบอกถึงโพรโทคอลที่ใช้ในการร้องขอเซอร์วิสซึ่งในงานวิจัยนี้จะใช้เว็บโพรโทคอลที่

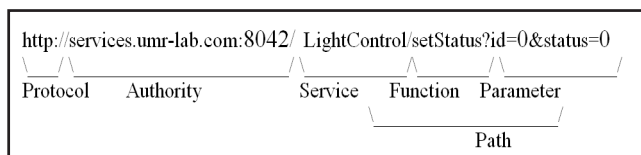
ทำงานในชั้นสื่อสารโปรแกรมประยุกต์ของ TCP/IP model คือ http ตามด้วยเครื่องหมาย “://”

ส่วนที่ 2 คือ Authority หรือ Base เป็นการอ้างถึงตำแหน่งที่อยู่ของอุปกรณ์ IoT ที่ให้บริการเซอร์วิสในระบบเครือข่ายหรือบนอินเทอร์เน็ตที่กำลังจะถูกร้องขอ นั้นๆ โดยอาจจะเป็นชื่อ Hostname หรือ หมายเลข IP Address เพียงอย่างเดียว หรืออาจจะมีการระบุหมายเลขพอร์ตที่ให้บริการร่วมด้วยก็ได้ ซึ่งในกรณีที่มีการระบุหมายเลขพอร์ตที่ให้บริการจะใช้เครื่องหมาย “:” เป็นตัวแบ่งระหว่าง Hostname และ Port

ส่วนประกอบลำดับที่ 3 ได้แก่ Path ซึ่งเป็นตำแหน่งที่ชี้ไปยังที่ตำแหน่งของเซอร์วิสที่ให้บริการและฟังก์ชันของเซอร์วิส นั้น โดยเป็นการอ้างถึงตามไวยากรณ์ที่ใช้เครื่องหมาย “/” ในการอ้างตำแหน่งตามลำดับซึ่งตำแหน่งก่อนสุดท้ายจะเป็นชื่อเซอร์วิสและตำแหน่งสุดท้ายจะเป็นชื่อฟังก์ชันของเซอร์วิส นั้นเอง

ส่วนสุดท้าย Parameter ซึ่งหมายถึงค่าพารามิเตอร์ที่อาจจะมีหรือไม่มีก็ได้ หากมีพารามิเตอร์นั้นๆ ก็จะถูกส่งไปยังฟังก์ชันของเซอร์วิส นั้นๆ เพื่อบอกวัตถุประสงค์ หรือเพื่อเรียกข้อมูลบางอย่างออกมาแสดง ในบางครั้งจะเรียกว่า Query string ก็ได้ โดยการอ้างค่าพารามิเตอร์จะใช้ “?” ตามด้วยชื่อพารามิเตอร์ และใช้เครื่องหมาย “=” เพื่อกำหนดค่าของพารามิเตอร์นั้นๆ ในกรณีที่มีหลายพารามิเตอร์จะใช้เครื่องหมาย “&” ในการแบ่งค่าพารามิเตอร์

ในภาพที่ 1 เป็น URL สำหรับเรียกใช้บริการจากเซอร์วิส LightControl ผ่านเว็บโพรโทคอลจาก service.umn.com ที่พอร์ตหมายเลข 8042 เพื่อเรียกใช้ฟังก์ชันเพื่อกำหนดสถานะของหลอดไฟ (setStatus) และมีการส่งค่าพารามิเตอร์ไปสองค่าได้แก่ค่า id = 0 และ status = 0 เป็นต้น ทั้งนี้ในการขอใช้บริการจากเซอร์วิสในบางฟังก์ชันอาจไม่จำเป็นต้องส่งค่าพารามิเตอร์ก็ได้



ภาพที่ 1 โครงสร้างของ URL สำหรับเรียกใช้เซอร์วิส

3. โครงสร้างเอกสารเอกซ์เอ็มแอลเพื่อใช้สำหรับอธิบายเซอร์วิส

จากตัวอย่างในภาพที่ 1 จะเห็นว่าหากผู้ขอใช้บริการไม่ทราบส่วนประกอบและวิธีในการเรียกใช้บริการก็จะไม่สามารถขอใช้บริการได้ การกำหนดรูปแบบของการอธิบายเซอร์วิสจะช่วยให้ผู้ขอใช้บริการสามารถเรียกใช้เซอร์วิสจากอินเทอร์เน็ตของสรรพสิ่งได้ง่ายขึ้น และยังสามารถสร้างส่วนติดต่อกับผู้ใช้งานในการเรียกใช้เซอร์วิสจากอินเทอร์เน็ตของสรรพสิ่งได้ในแบบอัตโนมัติอีกด้วย

เอกซ์เอ็มแอล[13] ถือเป็นภาษาที่ใช้ในการนิยามข้อมูลที่เป็นมาตรฐานของการแลกเปลี่ยนข้อมูลและได้รับความนิยมนอย่างสูง โดยเอกซ์เอ็มแอลเป็นภาษาประเภทมาร์กอัป (Markup) ที่อนุญาตให้ผู้ใช้งานสามารถสร้างหรือนิยามแท็ก (Tag) ขึ้นมาใช้เองได้โดยไม่ต้องรอการประกาศใช้อย่างเป็นทางการ ทำให้สามารถอธิบายความหมายของข้อมูลได้หลากหลายและมีความยืดหยุ่นในการนำไปใช้งาน ในบทความนี้จึงได้นำเอกซ์เอ็มแอล มาช่วยในการอธิบายเซอร์วิส โดยมีโครงสร้างของเอกสารดังภาพที่ 2

```
<!ELEMENT service (doc, method+)>
<!ATTLIST service
  base CDATA #REQUIRED
  name CDATA #REQUIRED
  path CDATA #REQUIRED>
<!ELEMENT doc (#PCDATA)>
<!ELEMENT method (doc, request, response)>
<!ATTLIST method
  mtype CDATA #REQUIRED
  name CDATA #REQUIRED
  path CDATA #REQUIRED>
<!ELEMENT request (param+)>
<!ELEMENT param (doc, range?, option*)>
<!ATTLIST param
  name CDATA #REQUIRED
  required CDATA #IMPLIED
  type CDATA #REQUIRED>
<!ELEMENT response (param)>
<!ELEMENT option (doc)>
<!ATTLIST option
  value CDATA #REQUIRED>
<!ELEMENT range (doc)>
<!ATTLIST range
  max CDATA #REQUIRED
  min CDATA #REQUIRED>
```

ภาพที่ 2 โครงสร้างเอกสารเอกซ์เอ็มแอลสำหรับอธิบายเซอร์วิส

จากภาพที่ 2 แสดงโครงสร้างของเอกสารเอกซ์เอ็มแอลเพื่ออธิบายเซอร์วิส ซึ่งประกอบด้วย อิลิเมนต์ (Element) หลักคือ Service ที่มี Attribute เพื่อใช้ในการระบุตำแหน่งที่อยู่และบอกถึงชื่อเซอร์วิส มีอิลิเมนต์รองได้แก่ Method ที่ใช้เพื่ออธิบายฟังก์ชันในการทำงาน ซึ่งประกอบด้วยอิลิเมนต์ย่อยภายในที่สำคัญคือ Request สำหรับอธิบายค่าที่ต้องการเพื่อเรียกใช้ฟังก์ชัน และ Response สำหรับอธิบายผลลัพธ์ที่ได้จากการเรียกใช้ฟังก์ชัน โดยทั้งสองอิลิเมนต์นี้จะใช้อิลิเมนต์ Param ในการบอกถึงค่าพารามิเตอร์ต่างๆ และในทุกอิลิเมนต์จะมีแท็ก Doc เพื่อใช้ในการอธิบายถึงข้อมูลของข้อมูลเหล่านี้ซึ่งจะเป็นประโยชน์ต่อการเรียกใช้เซอร์วิสหรือการนำไปสร้างส่วนติดต่อกับผู้ใช้ในแบบไดนามิกตามเซอร์วิสที่เรียกใช้

4. ตัวอย่างการพัฒนาอุปกรณ์ต้นแบบและแอปพลิเคชันสำหรับเรียกใช้เซอร์วิส

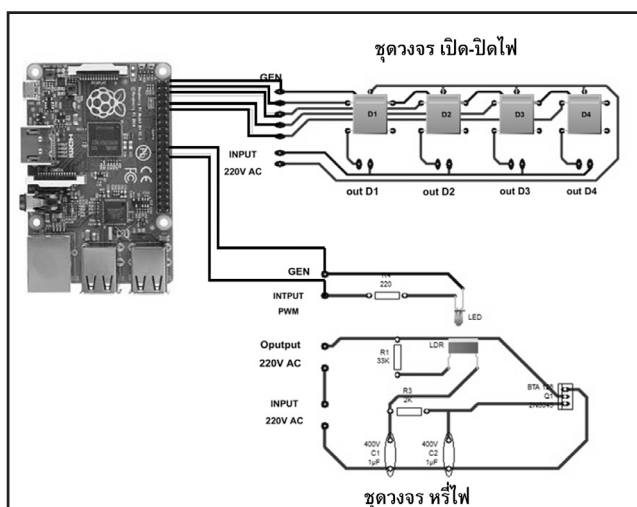
4.1 การพัฒนาอุปกรณ์ต้นแบบ

หลังจากที่ได้ทราบถึงโครงสร้างเอกสารเอกซ์เอ็มแอลเพื่อใช้สำหรับอธิบายเซอร์วิสแล้ว ในหัวข้อนี้จะนำเสนอการออกแบบอุปกรณ์ที่สามารถอธิบายเซอร์วิส และให้บริการเซอร์วิสเพื่อทดสอบการทำงาน โดยใช้บอร์ด Raspberry PI [14] ซึ่งประกอบด้วยส่วนฮาร์ดแวร์ที่เป็นการเชื่อมต่อวงจรทางไฟฟ้ากับบอร์ด ส่วนของซอฟต์แวร์ที่เป็นการควบคุมและสั่งงานให้อุปกรณ์ทำงานตามที่ต้องการ และส่วนของเซอร์วิสที่เป็นการนำเอาฟังก์ชันการทำงานเหล่านี้มาให้บริการในรูปแบบ RESTful ผ่านเว็บโปรโตคอล เซอร์วิสของอุปกรณ์ที่ได้พัฒนาขึ้นนั้นประกอบไปด้วยเซอร์วิสสำหรับเปิด/ปิดไฟฟ้า

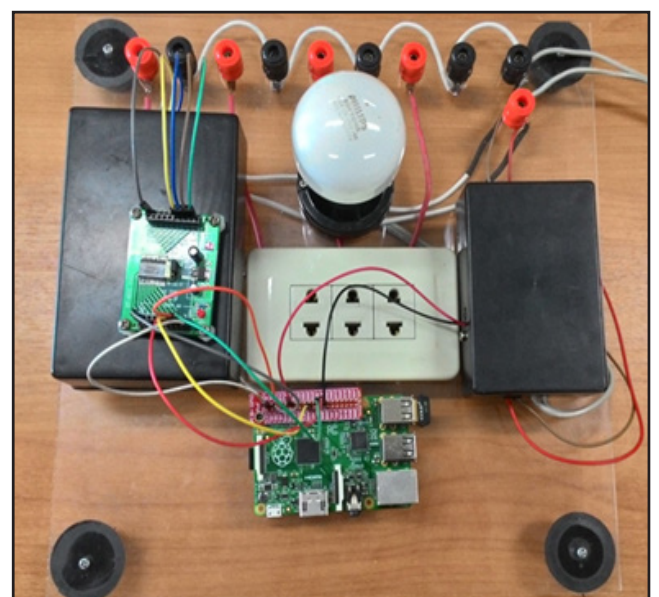
และเซอร์วิสสำหรับหรี่ไฟ ภาพวงจรและการเชื่อมต่อของอุปกรณ์แสดงได้ดังภาพที่ 3

โดยวงจรเปิด-ปิดไฟนี้จะทำงานต่อเมื่อมีสัญญาณ อินพุตจากบอร์ด Raspberry PI เข้ามาสั่งงานในชุดวงจรเปิด/ปิดไฟจะเป็นชุดวงจรควบคุมแรงดันในลักษณะเป็นสวิตช์เปิด/ปิดแรงดันไฟ 220 AC โดยใช้รีเลย์ในวงจรตัดไฟ มีการแบ่งจ่ายแรงดันออกเป็นสองส่วน คือ แรงดันไฟ 5v DC สำหรับเลี้ยงวงจร และ 220v AC ที่ใช้สำหรับควบคุมแรงดันไฟฟ้า 220v AC ออกไปทางเอาต์พุต ในส่วนของชุดวงจรหรี่ไฟนั้นจะทำงานต่อเมื่อมีสัญญาณ PWM (Pulse width modulation) จากบอร์ด Raspberry PI เข้ามาเพื่อสั่งงานให้หลอด LED ปรับเพิ่มลดแสง จากนั้น LDR จะทำการตรวจจับแสงจาก LED เพื่อปรับค่าความต้านทานให้กับวงจร ในการปรับค่าความต้านทานให้กับวงจรนี้ จะขึ้นอยู่กับความเข้มของแสง ถ้าแสงมีความสว่างมาก LDR ก็จะปรับค่าความต้านทานเพิ่มขึ้น ถ้าแสงมีความสว่างน้อย LDR ก็จะปรับค่าความต้านทานลดลงตามความเข้มของแสง โดยส่วนของซอฟต์แวร์นั้นถูกพัฒนาด้วยภาษา C# และ PHP ในการเขียนชุดคำสั่งเพื่อควบคุมและให้บริการเซอร์วิสที่เป็นการควบคุมและสั่งงานให้อุปกรณ์ทำงานตามที่ต้องการ ซึ่งประกอบไปด้วยเซอร์วิสสำหรับเปิด/ปิดไฟฟ้า และเซอร์วิสสำหรับหรี่ไฟ

จากนั้นได้ทำการพัฒนาอุปกรณ์ให้สามารถอธิบายเซอร์วิส และให้บริการเซอร์วิสเพื่อทดสอบการทำงานดังภาพที่ 4 ต้นแบบของอุปกรณ์ที่พัฒนาขึ้น สามารถอธิบาย



ภาพที่ 3 ภาพวงจรและการเชื่อมต่อกับ Raspberry PI



ภาพที่ 4 ต้นแบบของอุปกรณ์ที่พัฒนาขึ้น

การเรียกใช้เซอร์วิสผ่านเว็บโพรโทคอล ซึ่งสามารถป้อน URL ในเว็บเบราว์เซอร์เพื่อทดสอบได้ โดยจะแสดงเอกสารเอกซ์เอ็มแอลที่บอกถึงการขอใช้บริการจากเซอร์วิสและค่าพารามิเตอร์ต่างๆ โดยในกรณีที่ผู้ขอใช้บริการไม่ทราบส่วนประกอบและวิธีในการเรียกใช้เซอร์วิสหรือกรณีที่มีการเรียกใช้บริการโดยอ้างอิง URL ไม่ครบถ้วนอุปกรณ์จะแสดงเอกสารเอกซ์เอ็มแอลเพื่ออธิบายเซอร์วิสเช่นกัน

```
<?xml version="1.0" encoding="UTF-8"?>
<service name="LightControl"
base="192.168.1.88" path="LightControl" >
<doc>Light Control Service</doc>
<method name="setStatus" path="setStatus" mtype="POST">
<doc>กำหนดสถานะของหลอดไฟ</doc>
<request>
<param name="id" type="xsd:string" required="true">
<doc>กำหนดหมายเลขหลอดไฟ</doc>
<option value="0"><doc>หลอดไฟหมายเลข 0</doc></option>
<option value="1"><doc>หลอดไฟหมายเลข 1</doc></option>
<option value="2"><doc>หลอดไฟหมายเลข 2</doc></option>
</param>
<param name="status" type="xsd:string" required="true">
<doc>กำหนดสถานะ</doc>
<option value="0"><doc>ปิด</doc></option>
<option value="1"><doc>เปิด</doc></option>
</param>
</request>
<response><param name="result" type="xsd:string">
<doc>ผลการดำเนินการ</doc></param>
</response>
</method >
</service>
```

ภาพที่ 5 ข้อมูลอธิบายการเรียกใช้เซอร์วิส

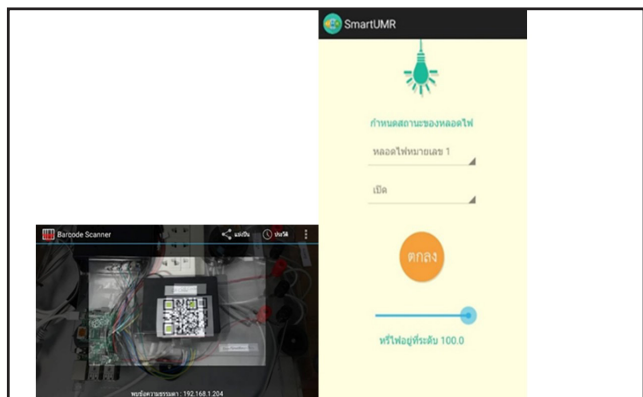
ข้อมูลในเอกสารเอกซ์เอ็มแอลในภาพที่ 5 สามารถบอกได้ว่าในการเรียกใช้เซอร์วิสเปิด/ปิดไฟฟ้าที่ใช้ URL คือ <http://192.168.1.88/LightControl/setStatus?id=0 &status=0> เป็นการสั่งปิดหลอดไฟหมายเลข 1

หากต้องการเปิดหลอดไฟหมายเลข 1 จะต้องใช้ URL เป็น <http://192.168.1.88/LightControl/setStatus?id=0 &status=1> เป็นต้น

4.2 การพัฒนาแอปพลิเคชันสำหรับเรียกใช้เซอร์วิส

ในการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์สำหรับเรียกใช้เซอร์วิสจากอุปกรณ์ด้วยสมาร์ทโฟนผ่านการอ่านคิวอาร์โค้ด เพื่อให้สะดวกในการเรียกใช้เซอร์วิสจากอุปกรณ์แต่ละตัว แอปพลิเคชันที่พัฒนาขึ้นจะสร้างเมนูในการเรียกใช้เซอร์วิสในแต่ละอุปกรณ์ผ่าน URL ของอุปกรณ์

นั้นๆ ซึ่งมีขั้นตอนทำงานดังนี้ (1) โปรแกรมทำการสแกนคิวอาร์โค้ด (2) โปรแกรมทำการเปิด URL ที่ได้เพื่อดึงค่าข้อมูลของเซอร์วิสจากเอกสาร XML (3) ทำการประมวลผลข้อมูลที่ได้นำมาสร้างเป็นเมนูให้ผู้ใช้ได้เลือกใช้เซอร์วิส (4) เมื่อผู้ใช้เลือกใช้เซอร์วิสจะสร้าง URL สำหรับเรียกใช้เซอร์วิสและส่งไปยังอุปกรณ์ (5) อุปกรณ์ให้บริการเซอร์วิสและส่งผลกลับคืนให้โปรแกรม (6) โปรแกรมทำการประมวลผลข้อมูลที่ได้นำมาแสดงผลต่อผู้ใช้ แอปพลิเคชันดังกล่าวถูกพัฒนาด้วยภาษาจาวา และใช้ไลบรารี Zxing [15] สำหรับการอ่านคิวอาร์โค้ด ซึ่งมีการสร้างเมนูในการเรียกใช้เซอร์วิสในแต่ละอุปกรณ์ผ่าน URL ของอุปกรณ์นั้นๆ ในแบบอัตโนมัติรับคำสั่งจากผู้ใช้เพื่อขอใช้บริการจากอุปกรณ์

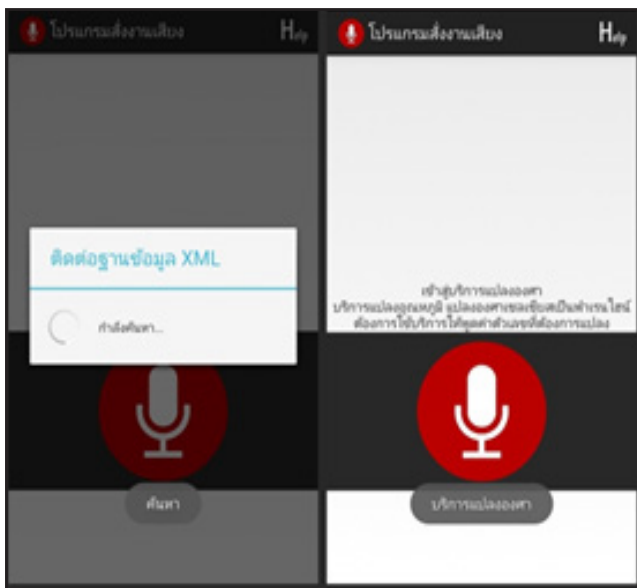


ภาพที่ 6 ตัวอย่างหน้าจอแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์

ในภาพที่ 6 แสดงหน้าจอแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์ที่เรียกใช้เซอร์วิสจากอุปกรณ์ด้วยสมาร์ทโฟนผ่านการอ่านคิวอาร์โค้ดที่ช่วยให้ง่ายในการระบุ URL ของอุปกรณ์อินเทอร์เน็ตของสรรพสิ่งแต่ละตัว แอปพลิเคชันถูกพัฒนาด้วยภาษาจาวา ซึ่งเป็นการสร้างเมนูในการเรียกใช้เซอร์วิสในแต่ละอุปกรณ์ผ่าน URL ของอุปกรณ์นั้นๆ ในแบบอัตโนมัติ โดยแอปพลิเคชันที่พัฒนาขึ้นสามารถแสดงส่วนติดต่อกับผู้ใช้ได้อย่างถูกต้องและเมื่อมีการเรียกใช้เซอร์วิสผ่านเมนูที่ถูกสร้างขึ้นก็สามารถขอใช้บริการเซอร์วิสจากอุปกรณ์ที่พัฒนาขึ้นได้อย่างถูกต้องเช่นเดียวกัน

ส่วนแอปพลิเคชันเพื่อเรียกใช้เซอร์วิสผ่านการสั่งงานด้วยเสียงเป็นการสร้างเมนูสำหรับเรียกใช้เซอร์วิสผ่านเซิร์ฟเวอร์ที่เก็บข้อมูลของเซอร์วิสต่างๆ ไว้ โดยมีหลักการทำงานดังนี้ (1) โปรแกรมทำการเปิด URL เพื่อดึงค่าข้อมูลของเซอร์วิสจากเซิร์ฟเวอร์ (2) โปรแกรมทำการประมวลผลข้อมูลที่ได้นำมาสร้างเป็นเมนูให้ผู้ใช้ได้เลือกใช้เซอร์วิส (3) ทำการประมวลผลข้อมูลที่ได้นำมาสร้างเป็นเมนูให้ผู้ใช้ได้เลือกใช้เซอร์วิส (4) เมื่อผู้ใช้เลือกใช้เซอร์วิสจะสร้าง URL สำหรับเรียกใช้เซอร์วิสและส่งไปยังอุปกรณ์ (5) อุปกรณ์ให้บริการเซอร์วิสและส่งผลกลับคืนให้โปรแกรม (6) โปรแกรมทำการประมวลผลข้อมูลที่ได้นำมาแสดงผลต่อผู้ใช้

มารอรับการเรียกใช้บริการจากผู้ (3) เมื่อผู้ใช้งานจะทำการแปลงคำสั่งให้เป็นตัวอักษรเพื่อเปรียบเทียบกับเซอร์วิสที่มีแล้ว สร้าง URL สำหรับเรียกใช้เซอร์วิสและส่งไปยังอุปกรณ์ (4) อุปกรณ์ให้บริการเซอร์วิสและส่งผลกลับคืนให้โปรแกรม (5) โปรแกรมทำการประมวลผลข้อมูลที่ได้อีกกลับมาแปลงให้เป็นเสียงเพื่อแจ้งผู้ใช้ต่อไป โดยตัวอย่างหน้าจอแอปพลิเคชันสำหรับเรียกใช้เซอร์วิสจากอุปกรณ์ด้วยสมาร์ตโฟนผ่านการสั่งงานด้วยเสียงบนระบบปฏิบัติการแอนดรอยด์ แสดงได้ในภาพที่ 7 โดยเป็นการดึงข้อมูลเอกสารเอกซ์เอ็มแอลที่ใช้ในการอธิบายเซอร์วิสไปจากเซิร์ฟเวอร์มาแสดงผล



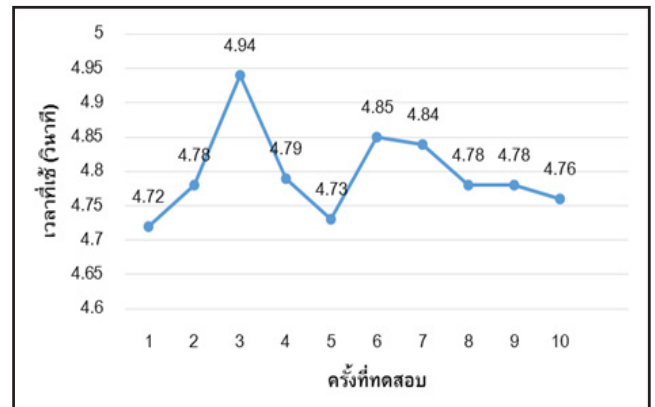
ภาพที่ 7 ตัวอย่างหน้าจอแอปพลิเคชันสำหรับเรียกใช้เซอร์วิสผ่านการสั่งงานด้วยเสียง

แอปพลิเคชันพัฒนาด้วยภาษาจาวาและไลบรารีเอสไอโอเอกซ์ [16] สำหรับแปลงเสียงให้เป็นข้อความและแปลงข้อความให้เป็นเสียง ซึ่งมีการสร้างเมนูในการเรียกใช้เซอร์วิสในแต่ละอุปกรณ์ผ่าน URL ของอุปกรณ์นั้นๆ ในแบบอัตโนมัติผ่านการอ่านออกเสียงและรอรับคำสั่งเสียงจากผู้ใช้งานเพื่อขอใช้บริการจากจากอุปกรณ์

4.3 ตัวอย่างการทดสอบการอธิบายและเรียกใช้เซอร์วิส

ในการทดสอบการทำงานนั้น ในบทความนี้ได้ทำการทดสอบการอธิบายการเรียกใช้เซอร์วิสต่างๆ ผ่านเว็บเบราว์เซอร์ด้วยการป้อนหมายเลขไอพีแอดเดรสของอุปกรณ์เพื่อทำการทดสอบการทำงานของอุปกรณ์ว่าสามารถอธิบายเซอร์วิสและให้บริการเซอร์วิสได้หรือไม่โดยเริ่มจาก

การทดสอบการอธิบายการเรียกใช้เซอร์วิสต่างๆ ผ่านเว็บเบราว์เซอร์ด้วยการป้อน URL สำหรับการขอใช้บริการจากเซอร์วิสและค่าพารามิเตอร์ต่างๆ ซึ่งพบว่าอุปกรณ์ต้นแบบสามารถอธิบายเซอร์วิสและให้บริการเซอร์วิสได้ตามวัตถุประสงค์ นอกจากนี้ยังได้ทำการทดสอบโดยใช้แอปพลิเคชันที่ได้พัฒนาขึ้นเพื่อเรียกใช้เซอร์วิสจากอุปกรณ์ด้วยสมาร์ตโฟนผ่านการอ่านคิวอาร์โค้ดที่ติดอยู่ที่ตัวอุปกรณ์เพื่อแสดงส่วนติดต่อกับผู้ใช้จากอุปกรณ์ต้นแบบเพื่อทดสอบประสิทธิภาพในด้านการเวลาในการแสดงผลและความถูกต้องในการสร้างส่วนติดต่อกับผู้ใช้



ภาพที่ 8 กราฟแสดงเวลาในการแสดงผลในการสร้างส่วนติดต่อกับผู้ใช้งาน

ในการทดสอบเพื่อหาผลประสิทธิภาพในการสร้างส่วนติดต่อกับผู้ใช้งานในการเรียกใช้เซอร์วิสในแบบอัตโนมัติด้วยโปรแกรมที่พัฒนาขึ้น เวลาที่ใช้ในการสร้างส่วนติดต่อกับผู้ใช้แสดงได้ดังภาพที่ 8 ซึ่งจากการทดสอบพบว่าโปรแกรมสามารถสร้างส่วนติดต่อกับผู้ใช้งานได้ตรงตามข้อมูลเซอร์วิสจากเอกสารเอกซ์เอ็มแอลและใช้เวลาในการสร้างส่วนติดต่อกับผู้ใช้อยู่ระหว่าง 4.72 - 4.94 วินาที ซึ่งไม่ต่างกันมากนัก

4.4 การประเมินความพึงพอใจที่มีต่อหน้าวิธีอธิบายการเรียกใช้เซอร์วิส

นอกจากนี้ในบทความนี้ยังได้ทำการประเมินความพึงพอใจที่มีต่อวิธีอธิบายการเรียกใช้เซอร์วิสด้วยแบบสอบถามความพึงพอใจโดยผู้เชี่ยวชาญด้านคอมพิวเตอร์ จำนวน 5 ท่าน ซึ่งใช้วิธีเลือกในแบบเจาะจง แล้ววิเคราะห์ข้อมูลใช้สถิติพื้นฐาน คือค่าเฉลี่ยและส่วนเบี่ยงเบนมาตรฐาน [17] เพื่อนำค่าเฉลี่ยมาแปลผลดังนี้

ค่าเฉลี่ยเท่ากับ 4.50 - 5.00 หมายความว่า ระดับมากที่สุด
ค่าเฉลี่ยเท่ากับ 3.50 - 4.49 หมายความว่า ระดับมาก

ค่าเฉลี่ยเท่ากับ 2.50 - 3.49 หมายความว่า ระดับปานกลาง
ค่าเฉลี่ยเท่ากับ 1.50 - 2.49 หมายความว่า ระดับน้อย
ค่าเฉลี่ยเท่ากับ 1.00 - 1.49 หมายความว่า ระดับน้อยที่สุด

ตารางที่ 1 ผลการประเมินความพึงพอใจที่มีต่อหน้าวิธีอธิบายการเรียกใช้เซอร์วิสโดยผู้เชี่ยวชาญ

ข้อคำถามของแบบประเมินความพึงพอใจ	$\bar{x}$	S.D	การแปลผล
1. ความถูกต้องของโครงสร้างเอกสาร XML	5	0	มากที่สุด
2. ความถูกต้องของผลลัพธ์ที่ได้จากการประมวลผลในโปรแกรม	4.8	0.45	มากที่สุด
3. ความง่ายต่อการใช้งานของระบบ	4.2	0.45	มาก
4. คำศัพท์ที่ใช้ผู้ใช้มีความคุ้นเคยและสามารถปฏิบัติตามได้โดยง่าย	4.4	0.54	มาก
5. ความน่าเชื่อถือได้ของระบบ	5	0	มากที่สุด
ค่าเฉลี่ย	4.68	0.48	มากที่สุด

ผลการการประเมินความพึงพอใจที่มีต่อวิธีอธิบายการเรียกใช้เซอร์วิสด้วยแบบสอบถามความพึงพอใจจากผู้เชี่ยวชาญด้านคอมพิวเตอร์จำนวน 5 ท่าน แสดงในตารางที่ 1 พบว่าผู้เชี่ยวชาญมีความพึงพอใจในความถูกต้องของโครงสร้างเอกสาร XML และความน่าเชื่อถือได้ของระบบอยู่ในระดับมากที่สุด โดยมีค่าความพึงพอใจเฉลี่ยเท่ากับ 5 ความถูกต้องของผลลัพธ์ที่ได้จากการประมวลผลในโปรแกรมคิดเป็นค่าเฉลี่ย 4.8 คำศัพท์ที่ใช้ผู้ใช้มีความคุ้นเคยและสามารถปฏิบัติตามได้โดยง่ายคิดเป็นค่าเฉลี่ย 4.4 ความง่ายต่อการใช้งานของระบบคิดเป็นค่าเฉลี่ย 4.2 และมีความพึงพอใจในภาพรวมคิดเป็นค่าเฉลี่ย 4.68 ซึ่งอยู่ในระดับมากที่สุด

5. สรุป

บทความนี้ได้นำเสนอวิธีอธิบายการเรียกใช้เซอร์วิสของอินเทอร์เน็ตของสรรพสิ่งที่ให้บริการเว็บเซอร์วิสแบบ RESTful ด้วยเอกสารเอกซ์เอ็มแอลเพื่อให้ผู้ใช้บริการได้ทราบส่วนประกอบและวิธีในการเรียกใช้บริการ โดยได้นำเสนอโครงสร้างเอกสารเอกซ์เอ็มแอลเพื่อการอธิบายการเรียกใช้บริการจากอินเทอร์เน็ตของสรรพสิ่งที่ใช้เว็บเซอร์วิสแบบ RESTful และได้แสดงตัวอย่างการพัฒนาอุปกรณ์

ต้นแบบที่สามารถอธิบายการเรียกใช้เซอร์วิสและให้บริการเซอร์วิสได้ ซึ่งประกอบไปด้วยเซอร์วิสสำหรับเปิด/ปิดไฟและเซอร์วิสสำหรับหรีไฟ ในการทดสอบการอธิบายการเรียกใช้เซอร์วิสได้ทดสอบด้วยเว็บเบราว์เซอร์เพื่อตรวจสอบการอธิบายเซอร์วิส และได้นำเสนอการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการแอนดรอยด์เพื่อเรียกใช้เซอร์วิสด้วยสมาร์ตโฟนผ่านการอ่านคิวอาร์โค้ดเพื่ออ้างอิงหมายเลขไอพีแอดเดรสของอุปกรณ์ที่มีข้อมูลวิธีอธิบายการเรียกใช้เซอร์วิสจากเอกสารเอกซ์เอ็มแอล ซึ่งได้พัฒนาขึ้นมาแสดงเป็นส่วนติดต่อกับผู้ใช้ในแบบอัตโนมัติ และแอปพลิเคชันทำงานด้วยเสียง

จากการนำทั้งสองแอปพลิเคชันมาทดสอบกับอุปกรณ์ต้นแบบพบว่าแอปพลิเคชันสามารถนำวิธีอธิบายการเรียกใช้เซอร์วิสจากเอกสารเอกซ์เอ็มแอล จากอุปกรณ์ต้นแบบมาแสดงเป็นส่วนติดต่อกับผู้ใช้ได้ถูกต้องและสามารถเรียกใช้งานได้อย่างถูกต้องเช่นเดียวกัน นอกจากนี้ยังได้ทำการทดสอบประสิทธิภาพของในด้านเวลาและความถูกต้องจากการสร้างส่วนติดต่อกับผู้ใช้ซึ่งจะเห็นได้ว่าโปรแกรมสามารถสร้างส่วนติดต่อกับผู้ใช้ได้ตรงตามข้อมูลเซอร์วิสจากเอกสารเอกซ์เอ็มแอลและใช้เวลาที่ใกล้เคียงกัน รวมถึงการประเมินความพึงพอใจที่มีต่อวิธีอธิบายการเรียกใช้เซอร์วิสด้วยแบบสอบถามความพึงพอใจจากผู้เชี่ยวชาญด้านคอมพิวเตอร์จำนวน 5 ท่าน ซึ่งพบว่ามีค่าความพึงพอใจในภาพอยู่ในระดับมากที่สุด ทำให้สามารถนำไปใช้เป็นแนวทางในการอธิบายเซอร์วิสของอินเทอร์เน็ตของสรรพสิ่งได้ อย่างไรก็ตามในอนาคตอาจมีการพัฒนาอุปกรณ์ต้นแบบที่สามารถให้บริการเซอร์วิสได้หลากหลายขึ้น และสามารถระบุตำแหน่งที่อยู่ของอุปกรณ์ได้โดยอัตโนมัติ ซึ่งจะทำให้ผู้ใช้บริการได้รับความสะดวกมากยิ่งขึ้น

6. เอกสารอ้างอิง

- [1] J. Gubbi, R. Buyya., S. Marusic, and M. Palaniswami. "Internet of Things (IoT): A vision, architectural elements, and future directions." *Future Generation Computer Systems*. Vol. 29, No. 7, pp. 1645-1660, 2013.
- [2] A. P. T. Hsu, W. T. Lee, A. J. C. Trappey, C. V. Trappey and A. C. Chang. "Using System Dynamics Analysis for Performance Evaluation of IoT Enabled One-Stop

- Logistic Services.” *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, Kowloon, pp. 1291-1296, 2015.
- [3] H. P. Breivold and K. Sandström. “Internet of Things for Industrial Automation -- Challenges and Technical Solutions.” *Proceedings of IEEE International Conference on Data Science and Data Intensive Systems*, Sydney, pp. 532-539, 2015.
- [4] G. Sun, F. Yu, X. Lei, Y. Wang, and H. Hu. “Research on Mobile Intelligent Medical Information System Based on the Internet of Things Technology.” *Proceedings of Frontiers 8th International Conference on Information Technology in Medicine and Education (ITME)*, Fuzhou, pp. 260-266, 2016.
- [5] R. Dolci. “IoT Solutions for Precision Farming and Food Manufacturing: Artificial Intelligence Applications in Digital Food.” *Proceedings of IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, Turin, pp. 384-385, 2017.
- [6] S. S. I. Samuel. “A review of connectivity challenges in IoT-smart home.” *Proceedings of MEC International Conference on Big Data and Smart City (ICBDSC)*, pp. 1-4, 2016.
- [7] K. Rafiullah, K. Sarmad, Z. Rifaqat, and K. Shahid. “Future Internet: The Internet of Things architecture, possible applications and key challenges.” *Proceedings of Frontiers of Information Technology (FIT)*, pp. 257-260, 2012.
- [8] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash. “Internet of Things: A Survey on Enabling Technologies, Protocols and Applications.” *IEEE Communications Surveys & Tutorials*. Vol. 17, No. 4, pp. 2347-2376, 2015.
- [9] World Wide Web Consortium. *Web services*. Available Online at <https://www.w3.org/2002/ws/>, accessed on 27 June 2017.
- [10] F. Curbera, M. Duftler, R. Khalaf, W. Nagy, N. Mukhi, and S. Weerawarana. “Unraveling the Web services web: an introduction to SOAP, WSDL, and UDDI.” *IEEE Internet Computing*, Vol. 6, No. 2, pp. 86-93, 2002.
- [11] A. Subbu. *RESTful Web Services Cookbook*. Published by Yahoo Press, Inc., ISBN 0-59-680168-8, 2010.
- [12] S. Malik and D. H. Kim. “A comparison of RESTful vs. SOAP web services in actuator networks.” *Proceedings of Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, Milan, pp. 753-755, 2017.
- [13] World Wide Web Consortium. *Extensible Markup Language (XML) 1.0 (Fifth Edition)*, Available Online at <https://www.w3.org/TR/xml/>, accessed on 27 June 2017.
- [14] M. Simon. *Raspberry Pi Cookbook*. Published by O’Reilly Media, Inc., ISBN 1-44-936522-1, 2014.
- [15] ZXing and ZXing. *Zebra crossing*, Available Online at <https://github.com/zxing/zxing/>, accessed on 27 June 2017.
- [16] SVOX. *SVOX Mobile Voices*, Available Online at <https://svoxmobilevoices.wordpress.com/developers/>, accessed on 27 June 2017.
- [17] M. Tiantong. *Statistics and research methodology for Information Technology*. Published by King Monkut’s University of Technology North Bangkok, 2005.