

ความเป็นจริงเสมือนสำหรับระบบบริหารจัดการพลังงานภายในบ้าน

Virtual Reality for Home Energy Management System



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ความเป็นจริงเสมือนสำหรับระบบบริหารจัดการพลังงานภายในบ้าน

Virtual Reality for Home Energy Management System



600265337

TB00099

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2560

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ความเป็นจริงเสมือนสำหรับระบบบริหารจัดการพลังงานในบ้าน

Virtual Reality for Home Energy Management System

ผู้จัดทำ

นายกฤตภาส นิรมิตรมหาปัญญา รหัสนักศึกษา 57010029



(ผศ. ดร. วิศิษฐ์ หิรัญกิตติ)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความเป็นจริงเสมือนสำหรับระบบบริหารจัดการพลังงานภายในบ้าน

นายกฤตภาส นิรมิตรมหาปัญญา 57010029

ผศ.ดร.วิศิษฎ์ หิรัญกิตติ

อาจารย์ที่ปรึกษา

ปีการศึกษา 2560

บทคัดย่อ

โครงการนี้ได้นำเทคโนโลยีความเป็นจริงเสมือนมาพัฒนาในรูปแบบเว็บแอปพลิเคชันที่ใช้งานบนสมาร์ตโฟนร่วมกับแว่น VR รุ่น Daydream ซึ่งเว็บแอปพลิเคชันนี้พัฒนาด้วยเทคโนโลยี WebVR โดยนี้มีวัตถุประสงค์เพื่อสร้างส่วนติดต่อกับผู้ใช้แบบ 3 มิติผ่านประสบการณ์แบบ VR เว็บแอปพลิเคชันเรียกว่า “VRHome” สื่อสารกับซอฟต์แวร์ที่ใช้ควบคุมและตรวจสอบปริมาณการใช้พลังงานภายในบ้านที่เรียกว่า “ระบบบริหารจัดการพลังงานภายในบ้าน” (HEMS: Home Energy Management System) การพัฒนาในครั้งนี้เราใช้ A-Frame framework ซึ่งเป็น framework ในการสร้างความเป็นจริงเสมือนบนแพลตฟอร์มเว็บ ร่วมกับ React และเพื่อที่ WebVR จะสามารถทำงานได้ ต้องอาศัยเว็บเบราว์เซอร์ที่รองรับเทคโนโลยี WebVR โดยในโครงการนี้เลือกใช้ Google Chrome

HEMS เป็นระบบที่มีการทำงานเป็นตัวกลางติดต่อกับอุปกรณ์ไฟฟ้าต่าง ๆ ภายในบ้านซึ่งจะมีรูปแบบโปรโตคอลการสื่อสารที่แตกต่างกันออกไป ตัวอย่างเช่น หลอดไฟที่สามารถเชื่อมต่อผ่าน ZigBee ได้ เป็นต้น โดยภายในระบบ HEMS จะมีการเก็บข้อมูลการใช้พลังงานไฟฟ้าของอุปกรณ์แต่ละตัวและยังสามารถตั้งเวลาและเงื่อนไขในการทำงานต่าง ๆ ได้ด้วยกฎ แต่ในโครงการนี้เป็นเพียงการสร้างอินเทอร์เฟซ 3 มิติ แบบ VR ที่ให้ผู้ใช้เรียกสั่งงาน HEMS ซึ่งเว็บแอปพลิเคชันที่พัฒนาขึ้นมาใช้เรียกใช้ RESTful API ของ HEMS ในการสั่งงานอุปกรณ์ไฟฟ้าอีกทอดหนึ่ง

ในโครงการนี้ยังมีการสร้างสภาพแวดล้อมในบ้านแบบ VR มี Control Panel แบบ 3 มิติสำหรับกดปุ่มสั่งงานอุปกรณ์ โดยใช้สมาร์ตโฟนในการสั่งงาน โดยอุปกรณ์ที่สั่งงานได้มีทั้งหมด 3 ชนิด คือ โทรท์สัน หลอดไฟ และเครื่องปรับอากาศ ผู้ใช้สามารถสั่งงานได้จำพวกคำสั่งพื้นฐาน และยังสามารถแสดงปริมาณการใช้พลังงานในแต่ละอุปกรณ์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Virtual Reality for Home Energy Management Systems

Mr. Krittaphat Niramitmahapanya 57010029

Asst.Prof.Dr. Visit Hirankitti Advisor

Academic Year 2017

ABSTRACT

In this project, we develop a client-side web application which connects to a home energy management system or HEMS. HEMS allows its user to monitor energy usage and to manually control home appliances. We develop a VR (virtual reality) JavaScript application, namely “VRHome”, running on a WebVR-enabled web browser, i.e. Chrome, to perform as a 3D user interface in order to visualize data from home appliances as well as to control them in 3D virtual world.

The VRHome is implemented using React and A-Frame framework whereas its 3D home model is created using Blender. The VRHome takes the use’s request via its 3D interface and translates that into a request to a RESTful web service provided by HEMS in order to retrieve data from a home appliance or to control it. In this project, we use a Daydream VR headset installed with a smartphone together with Chrome to run VRHome. There are three types of home appliances which VRHome can operate upon, i.e. a TV, air conditioners and light bulbs.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีด้วยความช่วยเหลือจากหลายฝ่าย ปริญญา
นิพนธ์ฉบับนี้จะสำเร็จไม่ได้หากปราศจากความช่วยเหลือจากอาจารย์ที่ปรึกษา ผศ.ดร. วิศิษฐ์ หิรัญ
กิตติ ที่ให้โอกาสในการ ทำโครงการนี้โดยได้รับคำแนะนำดีๆ และประสบการณ์ แนวคิด วิธีการ
ทำงาน จากอาจารย์เสมอมาจนสำเร็จลุล่วง ขอขอบคุณอาจารย์และบุคลากรต่าง ๆ ในสาขาวิชา
วิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำและสั่งสอนความรู้ต่าง ๆ มาโดยตลอด ขอขอบคุณรุ่นพี่
และเพื่อนๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำและแบ่งปันความรู้ในหลายๆ ด้าน
และให้กำลังใจผู้ไปด้วยกันเสมอมา ที่ขาดไม่ได้เลยคือ ขอขอบคุณ บิดา มารดา และครอบครัว ของ
ผู้จัดทำที่ให้การสนับสนุนอย่างเต็มที่ทั้งกำลังใจและกำลังทรัพย์จน โครงการนี้สำเร็จลุล่วง



กฤตภาส นิรมิตรมหาปัญญา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
สารบัญ.....	IV
สารบัญตาราง	IV
สารบัญรูป.....	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและปัญหา	1
1.2 ขอบเขตของโครงการ	2
1.3 วัตถุประสงค์ของโครงการ.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.5 ขั้นตอนการดำเนินงาน	3
บทที่ 2 ผลงานที่เกี่ยวข้อง	4
2.1 VR Home on STEAM	4
2.2 HoLolens for Control Smart Home	5
2.3 Hololens for Home Energy Management System	5
บทที่ 3 แนวคิดและทฤษฎีที่เกี่ยวข้อง	7
3.1 Home Energy Management Systems (HEMS).....	7
3.2 ความเป็นจริงเสมือน (Virtual Reality).....	9
ข้อจำกัดของความเป็นจริงเสมือน	10
3.3 WebGL and WebVR API.....	11
3.4 A-Frame framework	12
3.5 React framework.....	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 การออกแบบระบบและฟังก์ชัน	17
4.1 ความต้องการของระบบ	17
4.2 Use case diagram	18
4.3 การออกแบบส่วนต่อประสานกราฟฟิกกับผู้ใช้	19
4.4 อุปกรณ์และฟังก์ชันที่สามารถควบคุมได้ใน VRHome.....	20
1. โทรทัศน์	20
2. หลอดไฟ LED	20
3. เครื่องปรับอากาศ	20
4.5 ภาพรวมของระบบ	21
บทที่ 5 การพัฒนาเว็บแอปพลิเคชัน VRHome ที่สามารถใช้งานบนเว็บเบราว์เซอร์	23
5.1 การสร้างโปรเจกต์ด้วย create-react-app	23
5.2 การติดตั้ง A-frame ลงใน React application	24
5.3 การใช้งาน A-frame กับ React เพื่อเรียกใช้ API	26
5.3.1 การเพิ่มส่วนควบคุมลงใน A-frame	26
5.3.2 การเพิ่ม Events ให้กับ Avatar	28
5.3.3 การอินเตอร์เฟสกับ RESTful API	30
5.4 การใส่โมเดลให้กับ Components	32
5.2 VRHome และการใช้งานแต่ละอุปกรณ์	33
บทที่ 6 ผลการทดลอง.....	35
6.1 ทิวทัศน์หรือโทรทัศน์	35
6.2 หลอดไฟ	36
6.3 เครื่องปรับอากาศ	40
6.4 Teleport.....	42

7.1 บทสรุปของโครงการ.....	45
7.2 ปัญหาและอุปสรรค.....	46
7.3 แนวทางการพัฒนาต่อ.....	46
บรรณานุกรม.....	47



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง	หน้า
ตารางแสดงส่วนแบ่งทางการตลาดของ VR headsets ในปี 2016.....	10
ตารางแสดงฟังก์ชันการทำงานของโทรทัศน์	20
ตารางแสดงฟังก์ชันการทำงานของหลอดไฟ LED	20
ตารางแสดงฟังก์ชันการทำงานของเครื่องปรับอากาศ.....	21
ตาราง API ของ HEMS.....	22



สารบัญรูป

ภาพ	หน้า
ภาพที่ 2.1.1 ตัวอย่างการแสดงผลภายใน VR Home.....	4
ภาพที่ 2.2.1 ตัวอย่างการแสดงผลภายใน VR Home.....	5
ภาพที่ 2.3.1 การใช้ Hololens ติดต่อกับ HEMS	6
ภาพที่ 2.3.2 VuMark	6
ภาพที่ 3.1.1 ภาพแสดงโครงสร้างของระบบ HEMS	8
ภาพที่ 3.1.2 ภาพแสดงการทำงานของ HEMS.....	9
ภาพที่ 3.3.1 โลโก้ WebGL.....	11
ภาพที่ 3.3.2 โลโก้ WebVR.....	12
ภาพที่ 3.4.2 ภาพผลลัพธ์.....	13
ภาพที่ 3.5.1 Concept หลักของ React.....	15
ภาพที่ 3.5.2 การทำงานของ lifecycle ใน React.....	16
ภาพที่ 4.2.1 Use case diagram ของการใช้งาน VRHome.....	18
ภาพที่ 4.2.2 Use case diagram ของการใช้งาน HEMS.....	18
ภาพที่ 4.5.1 แผนภาพการทำงานระหว่าง VRHome กับ HEMS.....	21
ภาพที่ 5.1.1 ภาพหน้าเว็บจากการสร้างโปรเจกต์ด้วย create-react-app.....	24
ภาพที่ 5.1.2 ภาพโค้ดที่แสดงหน้าแรกจากไฟล์ src/App.js.....	24
ภาพที่ 5.2.1 ภาพการเขียนโค้ดการใช้ A-frame ด้วย React.....	25
ภาพที่ 5.3.1.1.1 โค้ดการเรียกใช้ cursor.....	26
ภาพที่ 5.3.1.2.1 โค้ดตัวอย่างการเรียกใช้ Google Daydream.....	27
ภาพที่ 5.3.1.2.2 ตัวอย่างการใช้ Google Daydream เป็นตัวชี้.....	28
ภาพที่ 5.3.2.1 ภาพตัวอย่างการเพิ่ม events ให้กับ avatar.....	28
ภาพที่ 5.3.2.2 ภาพก่อนถูกคลิก	29
ภาพที่ 5.3.2.3 ภาพหลังถูกคลิก.....	29
ภาพที่ 5.3.3.1 แผนภาพการแสดงผลการแลกเปลี่ยนข้อมูลด้วย RESTful API.....	30
ภาพที่ 5.3.3.2 โค้ดที่ทำงานเมื่อเปิดเว็บ	31
ภาพที่ 5.3.3.3 ภาพโค้ดการส่ง POST ไปที่ HEMS.....	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญตราเห็นไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 5.4.1 การใส่โมเดล.....	33
ภาพที่ 5.5.1 ภาพห้องจำลองใน VRHome	34
ภาพที่ 6.1.1 ภาพทีวีเปิด.....	35
ภาพที่ 6.1.2 ภาพทีวีปิด.....	36
ภาพที่ 6.2.1 หลอดไฟเปิด.....	36
ภาพที่ 6.2.2 แผงควบคุมของหลอดไฟ	37
ภาพที่ 6.2.3 ภาพแสดงการเปลี่ยนสีของหลอดไฟ	38
ภาพที่ 6.2.4 ภาพแสดงสภาพห้องตอนปิดไฟ	39
ภาพที่ 6.2.5 ภาพแสดงโมเดลหลอดไฟตอนปิดไฟ	40
ภาพที่ 6.3.1 โมเดลเครื่องปรับอากาศ	41
ภาพที่ 6.3.2 ภาพเมื่อแผงควบคุมเครื่องปรับอากาศพร้อมทำงาน	42
ภาพที่ 6.3.3 ภาพโมเดลของเครื่องปรับอากาศตอนปิดการทำงาน	42
ภาพที่ 6.4.1 ภาพก่อน Teleport.....	43
ภาพที่ 6.4.2 ภาพหลัง Teleport	43
ภาพที่ 6.4.3 ภาพไต่การกำหนดตำแหน่งมุมมอง.....	44
ภาพที่ 6.4.4 ภาพไต่ event ของลูกทรงกลม.....	44



บทที่ 1

บทนำ

1.1 ความเป็นมาและปัญหา

บ้านอัจฉริยะเป็นชื่อที่ใช้เรียกบ้านที่มีระบบบริหารจัดการสิ่งต่างๆ ที่ช่วยให้ผู้อยู่อาศัยนั้นมีชีวิตที่ดีขึ้น ช่วยให้ผู้อยู่ใช้งานมีชีวิตที่สะดวกสบายมากขึ้น ยกตัวอย่างเช่นสั่งการเปิดเครื่องปรับอากาศก่อนที่จะถึงบ้าน การปิดไฟอัตโนมัติในเวลากลางคืน เป็นต้น แต่ในช่วงหลายปีที่ผ่านมาเทคโนโลยีได้มีการพัฒนาอย่างก้าวกระโดด อินเทอร์เน็ตมีความเร็วที่สูงขึ้น คอมพิวเตอร์มีความเร็วที่แรงขึ้น เราสามารถผลิตกระแสไฟฟ้าจากแสงอาทิตย์ได้มากขึ้น และยังมีอุปกรณ์ในการสั่งงานเป็นที่ AI ยกตัวอย่างเช่น Google home ที่ผลิตโดยบริษัท Google และ Alexa ที่ผลิตโดย Amazon สองบริษัทยักษ์ใหญ่ทางด้านเทคโนโลยีบวกกับราคาค่าต้นทุนของอุปกรณ์ต่างๆ ทั้งแผงโซลาร์เซลล์ อุปกรณ์ IoT ต่างๆ นั้นมีราคาที่ถูกลง ทำให้บ้านอัจฉริยะนั้นเริ่มได้รับความนิยมมากขึ้นมีการนำมาใช้อย่างแพร่หลาย

ซึ่งเมื่อพูดถึงบ้านอัจฉริยะที่สามารถควบคุมสิ่งต่างๆ ได้แบบอัตโนมัติแล้วนั้น ในระบบก็ยังมีบางส่วนของระบบที่สามารถควบคุมด้วยตนเอง ทำให้สิ่งที่จำเป็นคือส่วนต่อประสานระหว่างระบบบ้านอัจฉริยะและผู้อยู่ใช้งาน โดยในระบบบ้านอัจฉริยะส่วนใหญ่จะพัฒนาอยู่ในรูปแบบของ สมาร์ทโฟน แอปพลิเคชัน เว็บแอปพลิเคชัน เนื่องจากในปัจจุบันผู้อยู่ใช้งานส่วนใหญ่จะมีอุปกรณ์ที่สามารถเข้าถึงแพลตฟอร์มอย่างเว็บและสมาร์ทโฟน แอปพลิเคชันอยู่แล้ว ทำให้ผู้ใช้ไม่จำเป็นต้องมีต้นทุนในการซื้ออุปกรณ์เพิ่มเติม โดยในซอฟต์แวร์ที่ติดตั้งในระบบบ้านอัจฉริยะก็จะมีจุดประสงค์ที่แตกต่างกันออกไปในแต่ละซอฟต์แวร์ โดยในโครงการได้ใช้ HEMS เป็นซอฟต์แวร์ในการจัดการอุปกรณ์ต่างๆ ภายในบ้าน และเป็นตัวกลางในการสื่อสารแลกเปลี่ยนข้อมูล

ระบบบริหารจัดการพลังงานภายในบ้าน (Home Energy Management System (HEMS)) เป็นระบบที่มีหน้าที่ตัวกลางในการเชื่อมต่อกับอุปกรณ์ต่าง ๆ ภายในบ้านกับหน้าเว็บแอปพลิเคชันผ่าน API ของระบบ ซึ่งเราจะสามารถสั่งงานเริ่มหรือหยุดการทำงานของอุปกรณ์รวมถึงตั้งค่าต่าง ๆ ของอุปกรณ์ผ่านระบบนี้ได้ ผ่านทางเว็บแอปพลิเคชันของ HEMS และทาง API services ที่ HEMS นั้นที่เปิดให้เรียกใช้เพื่อทำให้เราสามารถมีปฏิสัมพันธ์กับระบบได้ และปัญหาที่มักจะเกิดขึ้นในบ้านที่มีความซับซ้อนคือการระบุตำแหน่งของอุปกรณ์เครื่องใช้ไฟฟ้าต่าง ๆ หรือก็คือเราจะสามารถเห็นเพียงชื่อของอุปกรณ์และตำแหน่งของอุปกรณ์ในหน้าจอก็มีจำนวนมากๆ จะทำให้เราสับสน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ว่าอุปกรณ์นั้นอยู่ตำแหน่งไหนของบ้าน เราจึงได้ทำการศึกษาการทำ 3D User Interface เพื่อให้เราสามารถมองเห็น โครงสร้างของสถานที่ หรือบ้าน และตำแหน่งของอุปกรณ์ต่าง ๆ ได้ดีมากขึ้น

และเมื่อเราได้ศึกษาการเลือกใช้ 3D User Interface ก็พบว่าในปัจจุบันจะมีเทคโนโลยีสองตัวที่เป็นที่นิยมในขณะนี้คือ Virtual reality และ Augmented Reality ซึ่งมีข้อแตกต่างกันคือ Virtual reality นั้นจะทำให้ผู้ใช้นั้นเห็นภาพที่สร้างขึ้นโดยสมบูรณ์โดยการทำให้ผู้ใช้รู้สึกว่าคุณใช้ในเข้าไปอยู่ในสภาพแวดล้อมนั้นๆ จริงๆ และ Augmented Reality นั้นจะเป็นการสร้างวัตถุ 3 มิติลอยขึ้นมาในอากาศโดยเราต้องมองผ่านเลนส์ หรือหน้าจอถึงจะเห็นวัตถุสามมิติ นั้น ซึ่งทั้ง AR และ VR นั้นจะมีข้อจำกัดการใช้งานที่ต่างกันคือ AR จะไม่สามารถใช้งานเมื่อผู้ใช้ไม่ได้อยู่สถานที่จริงได้ แต่ VR นั้นทำได้ และเนื่องจากการทำงานส่วนใหญ่จะเป็นการควบคุมระยะไกลทำให้เราเลือกที่จะใช้เทคโนโลยีที่เรียกว่า Virtual Reality ในการศึกษาการนำมาใช้เป็นส่วนต่อประสานระหว่างผู้ใช้กับระบบ HEMS

โดยส่วนของ Virtual Reality ปัจจุบันนั้นมีคุณภาพที่หลากหลาย ซึ่งตัวที่มีคุณภาพสูงนั้นจะมีราคาที่สูงมาก อย่างไรก็ตามในการ VR นั้นเป็นเพียงแค่การนำส่วนประกอบต่างๆ เข้าด้วยกันและสร้างความรู้สึกให้กับผู้ใช้ ทำให้มีกลุ่มนักพัฒนาเริ่มพัฒนาโค้ดที่สามารถให้ประสบการณ์นั้นได้ผ่านเว็บเบราว์เซอร์ ทำให้ VR เป็นสิ่งที่ใกล้ตัวผู้ใช้ในระดับต่างๆ มากขึ้นเพราะเราสามารถเรียกใช้งานได้ผ่านเว็บเบราว์เซอร์โดยทั่วไปไม่จำเป็นต้องซื้ออุปกรณ์เสริมที่มีราคาแพงอีกต่อไป

1.2 ขอบเขตของโครงการ

- 1) สร้าง Virtual reality ที่ทำงานด้วยภาษา JavaScript ทำให้สามารถทำงานได้บนเว็บเบราว์เซอร์
- 2) สามารถเปิด/ปิดอุปกรณ์ผ่านระบบ HEMS ด้วย Virtual reality
- 3) สามารถเปลี่ยนแปลงสถานะตามแต่ละอุปกรณ์ที่ออกแบบ

1.3 วัตถุประสงค์ของโครงการ

- 1) เพื่อศึกษาการนำเทคโนโลยีความเป็นจริงเสมือนมาใช้กับระบบจัดการพลังงานภายในบ้าน เพื่อการควบคุมสั่งการอุปกรณ์บางชนิด
- 2) เพื่อศึกษาการออกแบบการควบคุมต่าง ๆ ในโลก VR
- 3) เพื่อศึกษาและทดลองการทำงานของ VR ที่สามารถทำงานผ่านเว็บเบราว์เซอร์บนสมาร์ตโฟนในการใช้งานจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ประโยชน์ที่คาดว่าจะได้รับ

สามารถใช้การควบคุมอุปกรณ์ต่าง ๆ ภายในบ้านโดยระบบจัดการพลังงานภายในบ้านด้วย VR ผ่านเว็บเบราว์เซอร์บนสมาร์ทโฟน และช่วยให้ผู้ใช้มีปฏิสัมพันธ์กับอุปกรณ์ต่างๆ ได้ง่ายขึ้น

1.5 ขั้นตอนการดำเนินงาน

- 1) ศึกษาและหาข้อมูลเกี่ยวกับเครื่องมือการสร้างเว็บแอปพลิเคชัน
- 2) ศึกษาและหาข้อมูลในการสร้าง Virtual reality
- 3) ศึกษาและทดลองใช้เครื่องมือต่างๆ เข้าด้วยกันทดสอบความเป็นไปได้
- 4) ทดลองสร้างการควบคุมที่มีการติดต่อแลกเปลี่ยนข้อมูลกับระบบ HEMS ด้วย RESTful
- 5) ออกแบบระบบโดยรวม
- 6) ดำเนินการสร้างโมเดลเพื่อนำมาใช้งาน โลกความเป็นจริงเสมือน
- 7) สร้างตัวโครงการงาน
- 8) ปรับปรุงโครงการตามความเหมาะสม
- 9) สรุปผลการทำโครงการและจัดทำเอกสาร

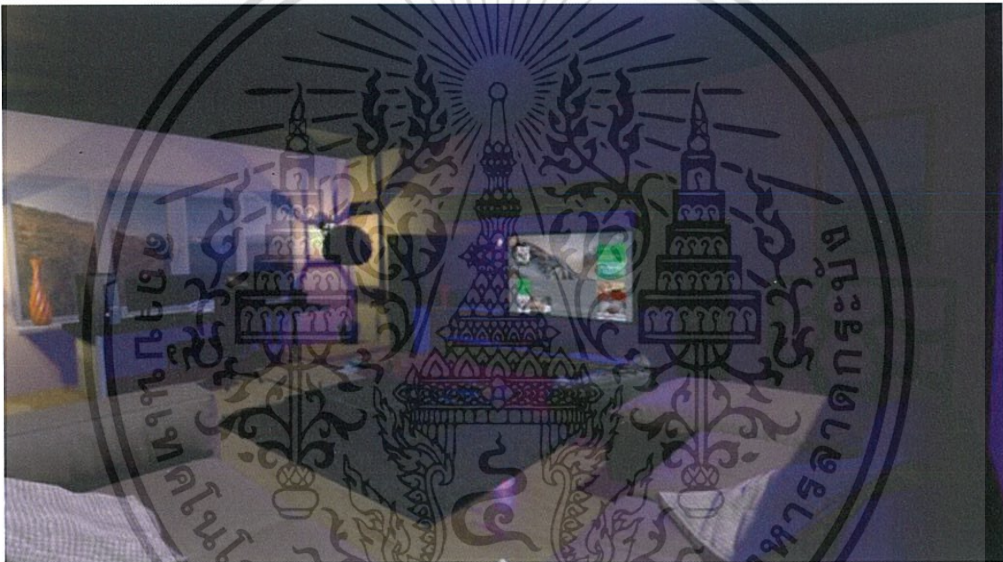
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ผลงานที่เกี่ยวข้อง

2.1 VR Home on STEAM

VR Home เป็นซอฟต์แวร์ที่ใช้สร้างบ้านในโลกความจริงเสมือน เพื่อให้ผู้ใช้งานเข้าไปอยู่อาศัย และมีปฏิสัมพันธ์กับอุปกรณ์ต่างๆ ในโลกที่สร้างขึ้นนั้น โดยตัวซอฟต์แวร์จะรองรับอุปกรณ์เฉพาะของ HTC Vive ซึ่งต้องใช้ PC ที่ค่อนข้างมีประสิทธิภาพในการประมวลผล ถึงจะได้คุณภาพที่ดี ตามตัวอย่างภาพที่ 2.1.1



ภาพที่ 2.1.1 ตัวอย่างการแสดงผลภายใน VR Home

ดังนั้นเราจึงมีความคิดเห็นว่า ในเมื่อซอฟต์แวร์ตัวนี้สามารถทำให้ผู้ใช้งานรู้สึกเสมือนอยู่บ้านจริงๆ ทำไม่ถึงไม่ลองนำมาใช้กับบ้านจริงๆ ไปเลย ในการควบคุมอุปกรณ์ต่างๆ ภายในบ้านแม้เราจะไม่ได้อยู่ในบ้านก็ตาม ทั้งยังมีแนวคิดที่ว่าทำเพื่อให้คนพิการได้รู้สึกว่าได้เดินไปรอบๆ บ้าน โดยการใช้ชีวิตอยู่ในโลกเสมือนจริงๆ ใช้ชีวิตอยู่ข้างใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 HoLolens for Control Smart Home

เป็นตัวอย่างการใช้ Hololens ในการควบคุมการเปิดไฟภายในบ้านที่ค่อนข้างชัดเจน โดยอุปกรณ์ที่ใช้คือ Microsoft HoLoLens เพื่อสั่งงานหลอดไฟ โดยงานตัวอย่างนี้จะสามารถทำได้เพียงควบคุมหลอดไฟเท่านั้น



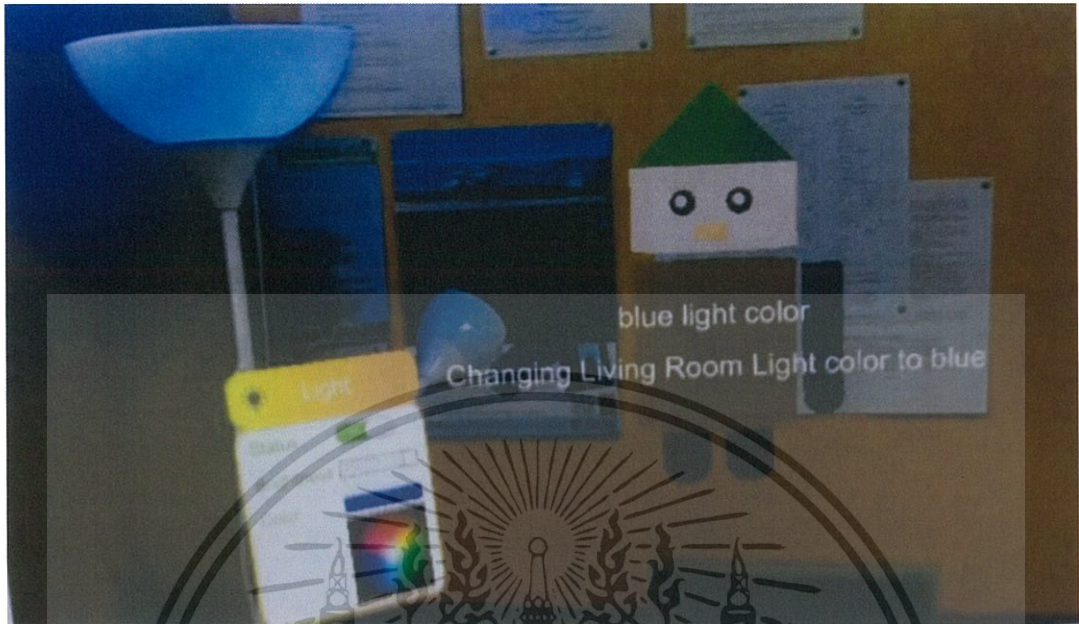
ภาพที่ 2.2.1 ตัวอย่างการแสดงผลภายใน VR Home

ซึ่งในตัวอย่างเป็นเพียงความพยายามในงานใช้งานเพื่อศึกษาการใช้งานในการใช้ HoLolens ในควบคุมอุปกรณ์เพียงอย่างเดียว และปัญหาในการใช้งานคือตัวอุปกรณ์ HoLolens นั้นมีราคาแพง และเมื่อเราอยู่ในบ้านเราสามารถเข้าไปสั่งงานแบบปกติได้ ไม่จำเป็นต้องมีอุปกรณ์ วิธีนี้จึงไม่เป็นที่นิยมมากนัก

2.3 Hololens for Home Energy Management System

เป็นโครงการที่นำ Hololens ใช้สำหรับควบคุมอุปกรณ์ไฟฟ้าภายในบ้านผ่านระบบ HEMS สามารถควบคุมอุปกรณ์ได้หลากหลายแต่ยังต้องอาศัย Marker ที่เรียกว่า VuMark ไว้อ้างอิงว่า เอกสารนี้เป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์นั้นคืออะไรและอยู่ตำแหน่งที่ Marker ปรากฏอยู่ ซึ่งสามารถใช้ควบคุมอุปกรณ์ต่างๆ ในระบบได้ ดังภาพ



ภาพที่ 2.3.1 การใช้ Hololens ติดต่อกับ HEMS



ภาพที่ 2.3.2 VuMark

ซึ่งในโครงการนี้ได้ใช้ HEMS ตัวเดียวกับกับโครงการเล่มนี้ ซึ่งโครงการนี้ใช้ Augmented reality เป็นส่วนประสานกับผู้ใช้ ซึ่งใช้ได้เพียงระยะใกล้ๆ ซึ่งในบางครั้งเราก็สามารถเดินเข้าไปสั่งงานอุปกรณ์ได้โดยตรงไม่จำเป็นต้องใช้ Hololens แต่โครงการเล่มนี้พัฒนาด้วย Virtual reality ทำให้เราสามารถใช้งานได้จากระยะไกลแม้ว่าเราไม่ได้อยู่ในบ้านก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

แนวคิดและทฤษฎีที่เกี่ยวข้อง

3.1 Home Energy Management Systems (HEMS)

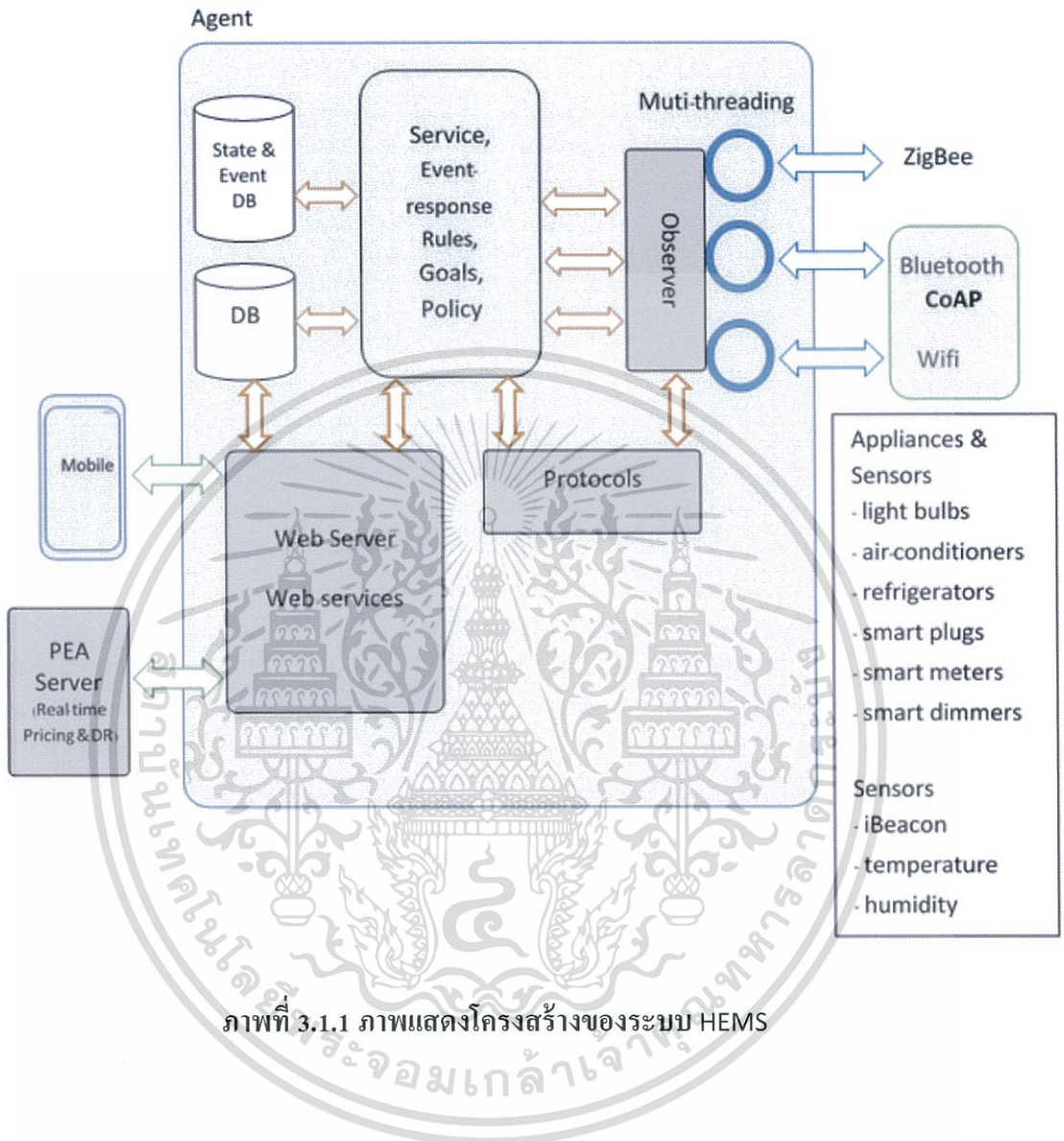
ระบบบริหารจัดการพลังงานในบ้าน (Home Energy Management System: HEMS) เป็นระบบที่เชื่อมโยงอุปกรณ์เครื่องใช้ไฟฟ้าภายในบ้านเข้าด้วยกัน โดยสามารถแสดงสถานการณ์ใช้ไฟฟ้า (เช่น ปริมาณการใช้ไฟฟ้าของอุปกรณ์ไฟฟ้าหรือบริเวณต่างๆ ในบ้านในช่วงเวลานั้น เป็นต้น) นอกจากนี้ HEMS ที่มีขีดความสามารถในระดับที่สูงขึ้นจะสามารถนำข้อมูลอื่นที่เกี่ยวข้อง เช่น สภาพอากาศ ราคาไฟฟ้าในช่วงนั้นๆ เป็นต้น มาประมวลผล พร้อมทั้งเสนอแนะต่อผู้ใช้ไฟฟ้าถึงวิธีการใช้พลังงาน หรือควบคุมการใช้พลังงานอย่างอัตโนมัติให้เกิดประสิทธิภาพสูงสุด

ในอนาคต หากมีการติดตั้งอุปกรณ์ด้านพลังงานใหม่ๆ เพิ่มเติมในบ้านเรือน เช่น ระบบผลิตไฟฟ้าพลังงานแสงอาทิตย์แบบติดตั้งบนหลังคา แบตเตอรี่ หรืออุปกรณ์ไฟฟ้าที่มีความสามารถในการดำเนินการตอบสนองด้านโหลด เป็นต้น HEMS จะสามารถเชื่อมโยงอุปกรณ์ต่างๆ เหล่านั้นเข้าด้วยกัน เพื่อให้การบริหารการทำงานของอุปกรณ์ต่างๆ เป็นไปอย่างมีประสิทธิภาพสูงสุด ตัวอย่างเช่น ในช่วงเวลาเที่ยงวันเมื่อระบบผลิตไฟฟ้าพลังงานแสงอาทิตย์แบบติดตั้งบนหลังคาผลิตไฟฟ้าได้มากกว่าการใช้ไฟฟ้าภายในบ้าน ณ ขณะนั้น HEMS สามารถสั่งให้แบตเตอรี่ชาร์จไฟเพื่อเก็บพลังงานไฟฟ้าส่วนเกินไว้ ในอนาคตหากมีการนำกลไกค่าไฟฟ้าที่เปลี่ยนแปลงตามเวลามาใช้ งาน HEMS เมื่อทราบว่าช่วงเวลานั้นมีราคาไฟฟ้าสูงสามารถสั่งให้แบตเตอรี่จ่ายไฟฟ้าง่ายกลับเข้าสู่ระบบไฟฟ้าในบ้าน เพื่อลดการพึ่งพาไฟฟ้าจากระบบโครงข่ายหลักซึ่งมีราคาสูง เป็นการช่วยให้เกิดการประหยัดค่าไฟฟ้าได้อย่างมีประสิทธิภาพ

นอกจากนั้น HEMS ยังสามารถทำให้ผู้ใช้ไฟฟ้าสามารถควบคุมเครื่องใช้ไฟฟ้าได้โดยตรงจากนอกบ้าน รวมถึงสามารถกำหนดการเปิด-ปิดเครื่องใช้ไฟฟ้าล่วงหน้าผ่านทางโทรศัพท์เคลื่อนที่ก็ได้ หากไม่มี HEMS ผู้ใช้ไฟฟ้าจำเป็นต้องควบคุมอุปกรณ์ไฟฟ้าต่างๆ แยกกัน เช่น การเปิดปิดไฟที่สวิตช์แต่ละจุดในบ้านโดยตรง เป็นต้น เมื่อมีการติดตั้ง HEMS ในบ้าน ผู้ใช้ไฟฟ้าสามารถควบคุมอุปกรณ์ไฟฟ้าต่างๆ ผ่านทางอุปกรณ์เคลื่อนที่ เช่น สมาร์ทโฟน หรือแท็บเล็ต เป็นต้น นอกจากนี้ อุปกรณ์อัตโนมัติบางส่วนสามารถติดตั้งเพิ่มเติมเข้าไปเพื่อให้ทำงานร่วมกับ HEMS ได้ เช่น เซ็นเซอร์จับการเคลื่อนไหว เพื่อปิดไฟส่องสว่างในบริเวณบ้านที่ไม่มีคนอยู่ เป็นต้น การที่ผู้ใช้ไฟฟ้าทราบข้อมูลการใช้ไฟฟ้าในส่วนต่างๆ โดยละเอียด รวมถึงสามารถควบคุมอุปกรณ์ต่างๆ ได้อย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

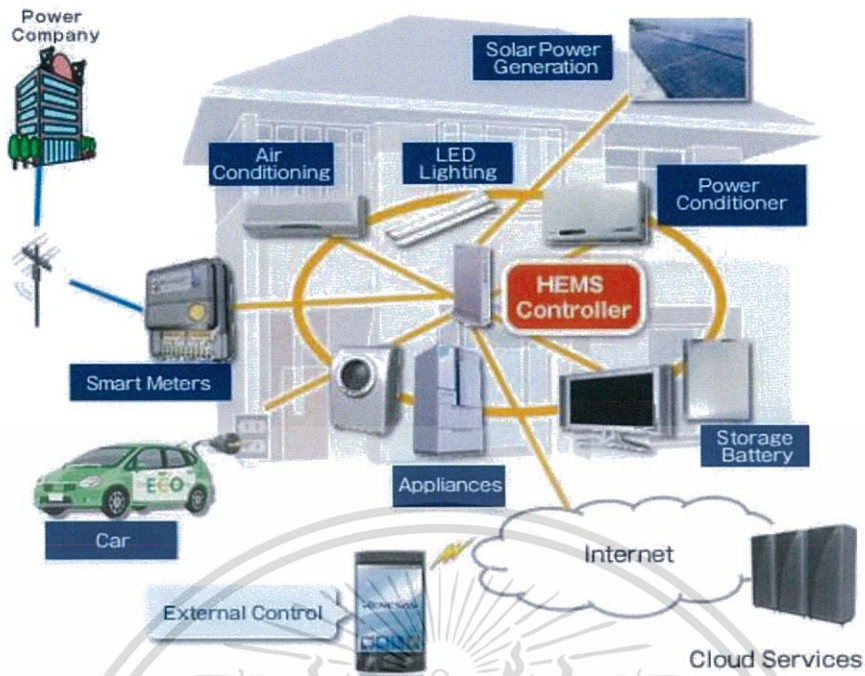
สะดวกมากขึ้นเป็นปัจจัยสำคัญที่ทำให้ผู้ใช้ไฟฟ้าปรับเปลี่ยนพฤติกรรมตนเองลงโดยลดการไฟฟ้าในส่วนที่ไม่จำเป็นลง อันจะส่งผลให้การใช้ไฟฟ้าในบ้านเรือนลดลง รวมถึงลดค่าไฟฟ้าลงได้



ภาพที่ 3.1.1 ภาพแสดงโครงสร้างของระบบ HEMS

สำหรับบ้านเรือนโดยทั่วไปแล้วจะมีการใช้งานอุปกรณ์ไฟฟ้าที่คล้ายคลึงกัน เช่น โทรทัศน์ เครื่องปรับอากาศ อุปกรณ์ไฟฟ้าในห้องครัว เป็นต้น ดังนั้นโดยทั่วไปแล้ว HEMS จะประกอบไปด้วย สมาร์ทมิเตอร์ อุปกรณ์กักเก็บพลังงาน รวมถึงระบบควบคุมและสั่งการ HEMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.1.2 ภาพแสดงการทำงานของ HEMS

3.2 ความเป็นจริงเสมือน (Virtual Reality)

ความเป็นจริงเสมือน (Virtual Reality) หรือที่เรียกสั้นๆ ว่า "วีอาร์" (VR) เป็นกลุ่มเทคโนโลยีเชิงโต้ตอบที่ผลักดันให้ผู้ใช้เกิดความรู้สึกของการเข้าร่วมอยู่ภายในสิ่งแวดล้อมที่สร้างขึ้นมาโดยคอมพิวเตอร์ พัฒนาการของความเป็นจริงเสมือนได้รับอิทธิพลมาจากแนวความคิดง่าย ๆ แต่มีอำนาจมากเกี่ยวกับการที่จะเสนอสารสนเทศอย่างไรให้ดีที่สุด ถ้าผู้ออกแบบสามารถให้ประสาทสัมผัสของผู้ใช้มีความค่อยเป็นค่อยไปในปฏิสัมพันธ์กับโลกทางกายภาพซึ่งเป็นสิ่งที่อยู่ล้อมรอบตัวเราแล้ว มนุษย์ก็จะสามารถรับและเข้าใจสารสนเทศได้ง่ายขึ้น ถ้าสารสนเทศนั้นกระตุ้นการรับรู้สัมผัสของผู้รับ ได้เหมือนกับโลกทางกายภาพ

เทคโนโลยีความเป็นจริงเสมือนสามารถเลียนการรับรู้สัมผัสของโลกทางกายภาพได้โดยสร้างการรับรู้หลายทางในสิ่งแวดล้อมสามมิติขึ้นมา ความเป็นจริงเสมือนได้สร้างเนื้อหาสาระของสิ่งที่แสดงให้เห็น โดยการรับรู้ซึ่งเป็นผลลัพธ์จากคอมพิวเตอร์ และตอบสนองต่อการเคลื่อนไหวทางกายภาพของผู้ใช้ที่สืบหาด้วยเครื่องรับรู้ของคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อจำกัดของความเป็นจริงเสมือน

การใช้งานในการทำงานที่เสมือนจริงนั้นจำเป็นต้องมีอุปกรณ์ (VR Headsets) โดยตามข้อมูลในปี 2017 จำแนกได้ดังนี้

1. มีการรับรู้การเคลื่อนที่และการหมุน (Six degrees of freedom (6DoF)) กับมีการรับรู้เพียงการหมุนอย่างเดียว (three degrees of freedom (3DoF))
2. มีรีโมตคอนโทรลหรือไม่ และรีโมตคอนโทรลนั้นเป็นแบบ 3DoF หรือ 6DoF
3. ทำงานด้วย PC หรือ Smartphone

โดยปกติทุกอุปกรณ์จะสามารถรับรู้การหมุนหรือเป็น 3DoF ส่วนอุปกรณ์ที่สามารถรับรู้ได้ทั้งการหมุนและการเคลื่อนที่หรือเป็น 6DoF จะทำให้ผู้ใช้สามารถเดินเข้าไปหาวัตถุในโลกเสมือนจริงได้ เพราะอุปกรณ์สามารถรับรู้การเคลื่อนที่ได้และราคาก็สูงกว่าอุปกรณ์ที่เป็นแบบ 3DoF

VR Platform	2016 Shipment Volume Market Share
Google Cardboard	69%
Samsung Gear	17%
Sony PS VR	3%
Steam / HTC	1%
Oculus Rift	<1%
Google Daydream	<0.1%
Others	9%

ตารางแสดงส่วนแบ่งทางการตลาดของ VR headsets ในปี 2016

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางสรุปได้ว่า ผู้ใช้ส่วนใหญ่จะเลือกที่จะบริโภค VR headsets ในระดับต่ำกว่าก่อนเนื่องจาก VR headsets คุณภาพสูงจะมีราคาแพงแล้วส่วนใหญ่ยังจำเป็นต้องมี PC เพื่อผลักดันประสิทธิภาพทำให้ต้นทุนในการเข้าถึง VR นั้นเข้าถึงได้ยากทำให้ VR บนสมาร์ตโฟนเป็นสิ่งที่เข้าถึงผู้ใช้ได้ง่ายกว่า VR headsets คุณภาพสูง ส่งผลให้การใช้งาน VR บนสมาร์ตโฟนเป็นตัวเลือกที่ดีในการมอบประสบการณ์ VR แก่ผู้ใช้ทุกระดับ

ไม่เพียงแต่เราต้องมี VR headsets แล้วเรายังจำเป็นต้องมีซอฟต์แวร์ในการสร้าง VR ขึ้นมา ยกตัวอย่างซอฟต์แวร์ที่ได้รับความนิยมเช่น Unity ที่สามารถสร้าง VR ที่ทำงานด้วยคอมพิวเตอร์ หรือแอปพลิเคชันบนมือถือ และยังมีซอฟต์แวร์ที่เขียนด้วยภาษา JavaScript ทำสามารถทำให้ใช้งานได้ผ่านเว็บเบราว์เซอร์ยกตัวอย่างเช่น WebGL และ WebVR API ซึ่งสิ่งที่ทำให้ VR ที่ทำงานบนเว็บเบราว์เซอร์สนใจ คือ เราไม่จำเป็นต้องติดตั้งตัวซอฟต์แวร์ลงบนเครื่อง เพียงแค่เราเปิดใช้งานเว็บเบราว์เซอร์สำหรับสมาร์ตโฟน กล่าวคือเราไม่จำเป็นต้องใช้สมาร์ตโฟนที่มีประสิทธิภาพสูงก็สามารถใช้งาน VR บนเบราว์เซอร์ได้ แม้คุณภาพของ VR ที่สามารถใช้งานบนเว็บเบราว์เซอร์ยังไม่สามารถเทียบได้เท่ากับการสร้างด้วย Unity แต่คุณภาพก็ยังถือได้ว่าน่าพึงพอใจ

3.3 WebGL and WebVR API



ภาพที่ 3.3.1 โลโก้ WebGL



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 3.3.2 โลโก้ WebVR

WebGL และ WebVR API เป็น JavaScript API ที่ช่วยเราในการสร้าง VR ที่ทำงานบนเว็บเบราว์เซอร์ โดยทำหน้าที่ต่างกันดังนี้ WebGL (Web Graphic Library) เป็น API ที่ใช้งานในการสร้างและแสดงภาพวัตถุสองมิติและสามมิติให้แสดงบนเว็บเบราว์เซอร์ที่รองรับการทำงาน WebGL ซึ่งเบราว์เซอร์เกือบทั้งหมดจะรองรับการทำงานของ WebGL API ยกเว้นบางรุ่นที่ออกแบบมาให้มีขนาดที่น้อยทำงานได้อย่างจำกัดและสุดท้ายคือ WebVR เป็น API ทำให้เราสามารถติดต่อกับ VR headsets ผ่านเว็บเบราว์เซอร์ได้ ยกตัวอย่างเช่นรับข้อมูลจากเซนเซอร์ของอุปกรณ์ว่าเราหมุนไปทางไหน หรือรับข้อมูลการทำงานของคอนโทรลเลอร์ว่าผู้ใช้กดปุ่มใดบ้างเพื่อควบคุม และตำแหน่งของคอนโทรลเลอร์ที่ตำแหน่งใด

ในการสร้าง VR ให้สามารถทำงานได้บนเว็บเบราว์เซอร์นั้นเราสามารถใช่เพียง WebGL และ WebVR API เราก็สามารถสร้าง VR บนเว็บเบราว์เซอร์ได้ แต่ถ้าเราทำงานที่มีขนาดงานที่มีความซับซ้อนมากขึ้นก็就会有ความซับซ้อนในการพัฒนาที่มากขึ้นด้วยทำให้เรานิยมใช้ VR framework เพื่อช่วยลดขั้นตอนในการทำ VR ให้สามารถทำงานได้บนเว็บเบราว์เซอร์ ทำให้มีกลุ่มนักพัฒนาซอฟต์แวร์พัฒนา VR framework ขึ้นมา ยกตัวอย่างเช่น ReactVR และ A-Frame framework เป็นต้น

3.4 A-Frame framework

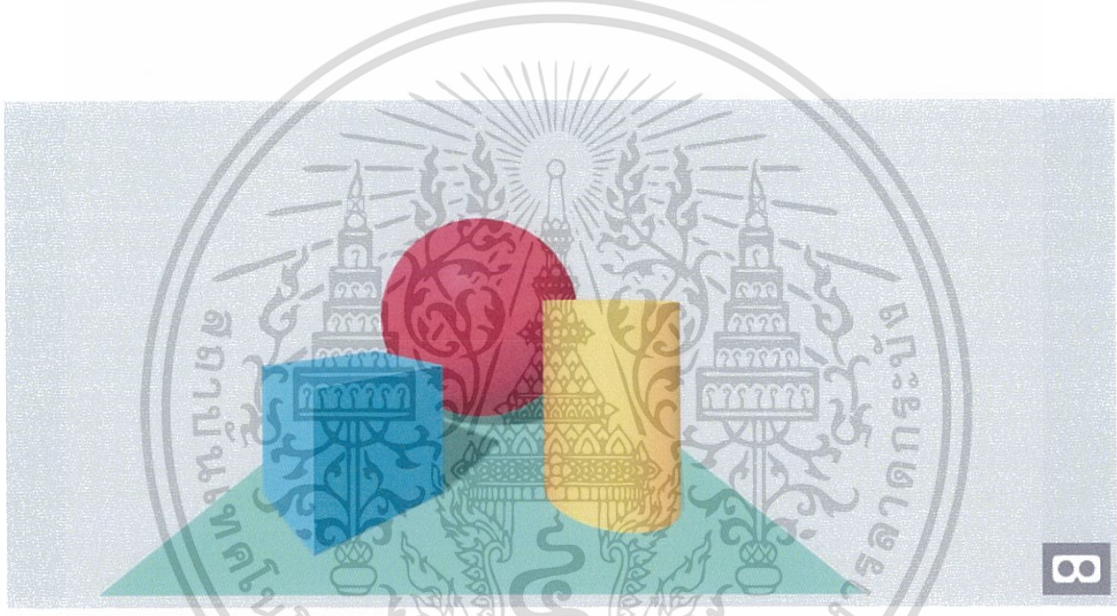
A-Frame คือ Web framework ที่ใช้สำหรับการสร้าง VR โดยตัว A-Frame นั้นเป็นโปรเจกต์ที่เริ่มโดย Mozilla และได้เปลี่ยนแนวทางการพัฒนาเป็นแบบ Open source ที่ให้นักพัฒนาทั่วโลกสามารถมีส่วนร่วมในการพัฒนาโค้ดของ A-Frame ได้ ทำให้ปัจจุบัน A-Frame เป็นหนึ่งใน Web framework ที่ประสบความสำเร็จสูงสุดในการสร้าง VR

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Hello, WebVR! - A-Frame</title>
5     <meta name="description" content="Hello, WebVR! - A-Frame">
6     <script src="https://aframe.io/releases/0.8.0/aframe.min.js"></script>
7   </head>
8   <body>
9     <a-scene>
10      <a-box position="-1 0.5 -3" rotation="0 45 0" color="#4CC3D9" shadow></a-box>
11      <a-sphere position="0 1.25 -5" radius="1.25" color="#EF2D5E" shadow></a-sphere>
12      <a-cylinder position="1 0.75 -3" radius="0.5" height="1.5" color="#FFC65D" shadow></a-cylinder>
13      <a-plane position="0 0 -4" rotation="-90 0 0" width="4" height="4" color="#7BC8A4" shadow></a-plane>
14      <a-sky color="#ECECEC"></a-sky>
15    </a-scene>
16  </body>
17 </html>

```

ภาพที่ 3.4.1 ตัวอย่างโค้ดใน A-Frame



ภาพที่ 3.4.2 ภาพผลลัพธ์

โดยการเรียกใช้ A-Frame สามารถเรียกใช้ได้เหมือน HTML ทั้งหมดตามตัวอย่างภาพที่ 2.3.1 ซึ่งเบื้องหลังการทำงานอาศัยการเรียกใช้ WebGL และ WebVR API และ JavaScript library อื่นๆ ที่ทำงานอยู่บนแพลตฟอร์มของเว็บเพื่อสร้าง VR และ A-Frame จะพัฒนาโดยใช้โครงสร้างของ Entity-Component-System (ECS) ซึ่งเป็นการทำให้เราสามารถพัฒนาได้ง่ายขึ้น ซึ่งมีแนวคิดคือ

1. Entities เป็นส่วนที่ใช้ห่อหุ้มมีลักษณะการใช้เหมือน div ที่วางเปล่าใน HTML ที่ใช้ในการแยกส่วนของ Object ต่างๆ ออกจากกัน
2. Components เป็นส่วนของการเก็บคุณลักษณะ และพฤติกรรมหรือฟังก์ชันการทำงานของ Component นั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Systems เป็นส่วนที่เก็บข้อมูลทั้งหมดของ Entities ต่างๆ ไว้ด้วยกันและสามารถทำงานร่วมกันได้

ตัวอย่างในการใช้ ECS เข้าด้วยกันในหลายๆ Component ให้กลายเป็นหนึ่ง Entity

Box = Position + Geometry + Material

Light Bulb = Position + Light + Geometry + Material + Shadow

Sign = Position + Geometry + Material + Text

VR Controller = Position + Rotation + Input + Model + Grab + Gestures

Ball = Position + Velocity + Physics + Geometry + Material

Player = Position + Camera + Input + Avatar + Identity

และยังสามารถปฏิสัมพันธ์กับวัตถุเหล่านี้ได้ ยกตัวอย่างเช่นการคลิกที่วัตถุแล้วเปลี่ยนสี หรือเคลื่อนที่ โดยเมื่อคลิกแล้วสามารถทำให้วัตถุหมุนได้ ด้วย Events และ Event listeners ในภาษา JavaScript

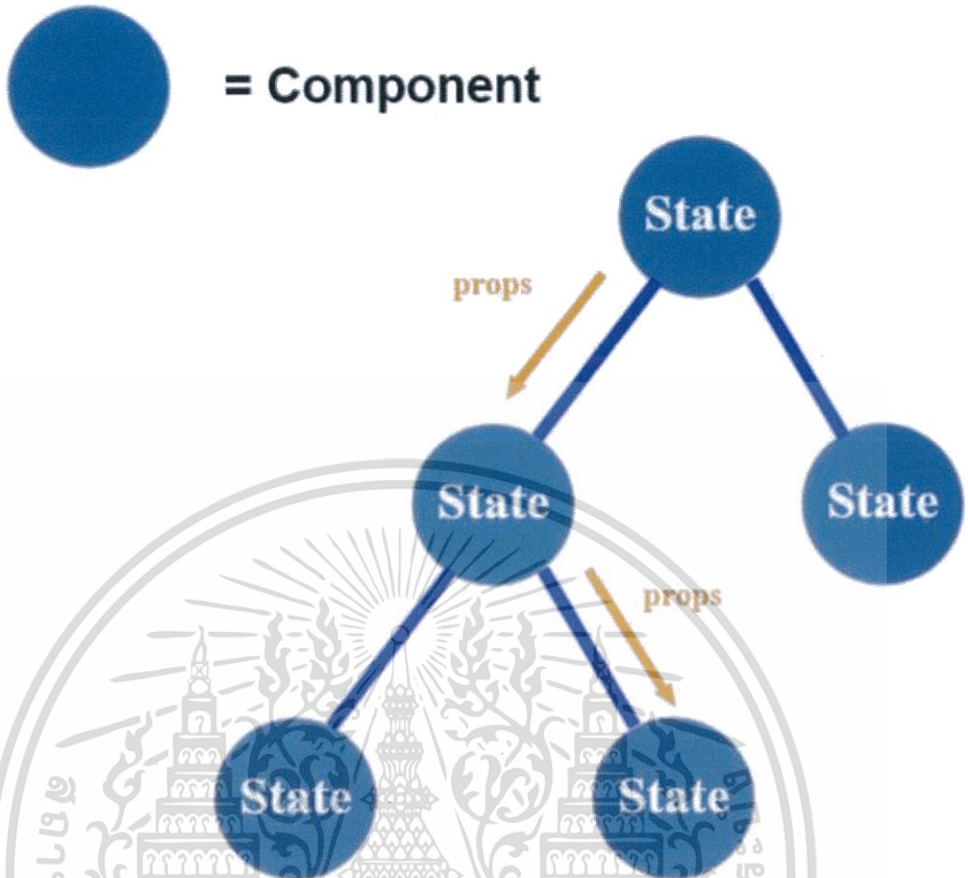
3.5 React framework

React framework คือ JavaScript library ที่ใช้เพื่อสร้าง User interface ช่วยลดความซับซ้อนในการเขียน User interface และช่วยให้นักพัฒนาจัดการข้อมูลต่าง ๆ ภายในแอปพลิเคชันในที่เป็นรูปแบบของ Component ที่จะมีการแบ่งแยกส่วนต่างๆ อย่างชัดเจน ทำให้นักพัฒนาสามารถพัฒนาได้ง่ายขึ้น นอกจากนักพัฒนาจะสามารถพัฒนา User interface ได้ง่ายขึ้น เรายังสามารถใช้ JavaScript library อื่นๆ เข้ามาทำงานร่วมกันในการดึงข้อมูลภายนอก หรือการจัดการข้อมูลภายใน

คอนเซปต์ที่เราต้องรู้เพื่อเขียน React หลัก ๆ มีแค่ 3 Concept เท่านั้น

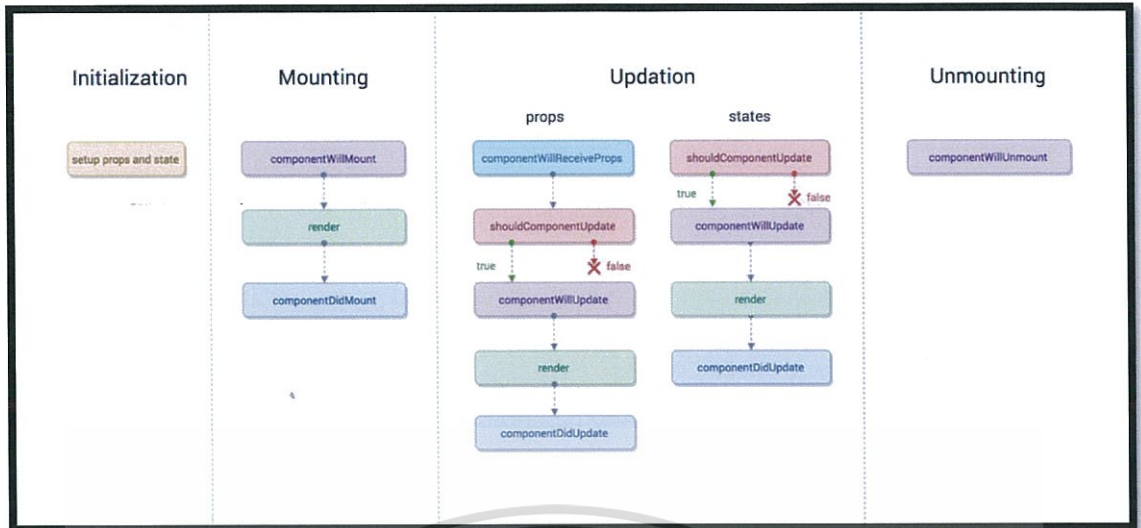
1. **Component** – ส่วนต่าง ๆ ในเว็บเราจะมองเป็น Component
2. **State** – ข้อมูลที่อยู่ใน Component แต่ละชิ้น เราเรียกว่า State
3. **Props** – ข้อมูลที่ถูกส่งต่อจาก Component ชั้นบนลงไปยังชั้นล่าง เราเรียกว่า Props (Properties)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.5.1 Concept หลักของ React

นอกจากนี้ในการทำงานของ React จะมีกลุ่มของฟังก์ชันที่เรียกว่า lifecycle ที่จะช่วยในการจัดการแอปพลิเคชันให้เป็นระบบมากขึ้นตามช่วงเวลาของ Component นั้นๆ โดยข้อดีของ lifecycle คือทำให้เรารู้ว่าฟังก์ชันตัวนี้ทำงานตอนไหน ทำให้เราสามารถแก้ปัญหาหรือพัฒนาเว็บได้มีประสิทธิภาพมากขึ้น โดยการทำงานของ lifecycle ทำงานตาม ภาพที่ 3.5.1



ภาพที่ 3.5.2 การทำงานของ lifecycle ใน React

และเนื่องจาก React นั้นใช้ภาษา JavaScript จึงสามารถใช้ประโยชน์จากการใช้หลักการของ Reusable component ซึ่งในโครงการนี้จะสังเกตได้ว่าเรามีหลอดไฟ กับเครื่องปรับอากาศอย่างละสองตัวซึ่งเราสร้างเพียงชนิดละ component เท่านั้นแต่เรียกใช้ถึงสองอุปกรณ์

บทที่ 4

การออกแบบระบบและฟังก์ชัน

4.1 ความต้องการของระบบ

เว็บแอปพลิเคชันฝั่งไคลเอนต์ที่พัฒนาขึ้นขอเรียกชื่อว่า “VRHome” ซึ่งมีความสามารถและคุณลักษณะดังนี้

Functional requirement

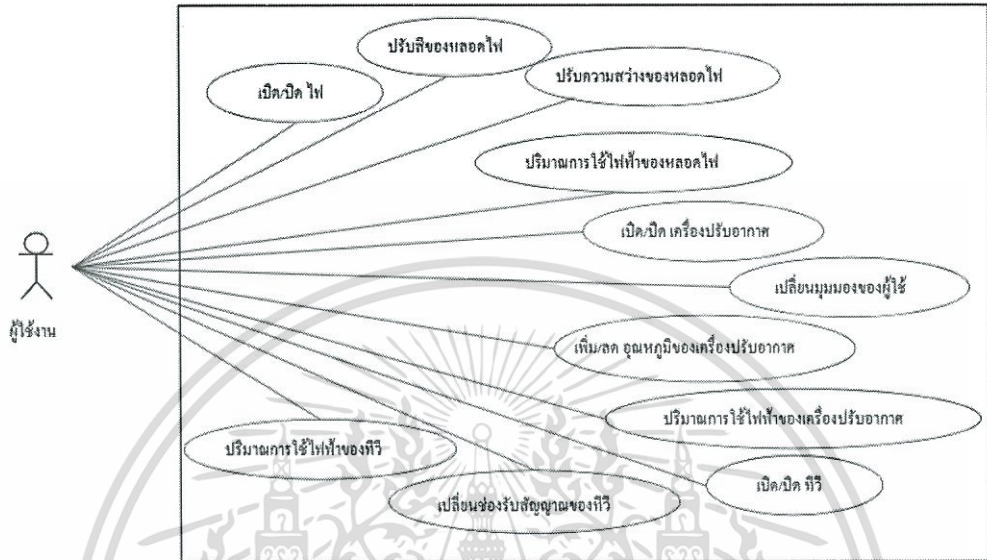
- ผู้ใช้สามารถสั่งเปิดปิดไฟ เปลี่ยนสีและปรับความสว่างของหลอดไฟได้โดยใช้ VRHome
- ผู้ใช้สามารถสั่งเปิดปิดทีวี เปลี่ยนช่องรับสัญญาณและปรับระดับความดังของทีวีได้โดยใช้ VRHome
- ผู้ใช้สามารถสั่งเปิดปิดเครื่องปรับอากาศ ปรับอุณหภูมิของเครื่องปรับอากาศได้โดยใช้ VRHome
- ผู้ใช้สามารถเรียกดูประวัติการใช้พลังงานของหลอดไฟ ทีวี และเครื่องปรับอากาศได้
- ผู้ใช้สามารถปรับเปลี่ยนมุมมองไปรอบๆ โลกเสมือนจริงได้ตามที่กำหนด

Non-functional requirement

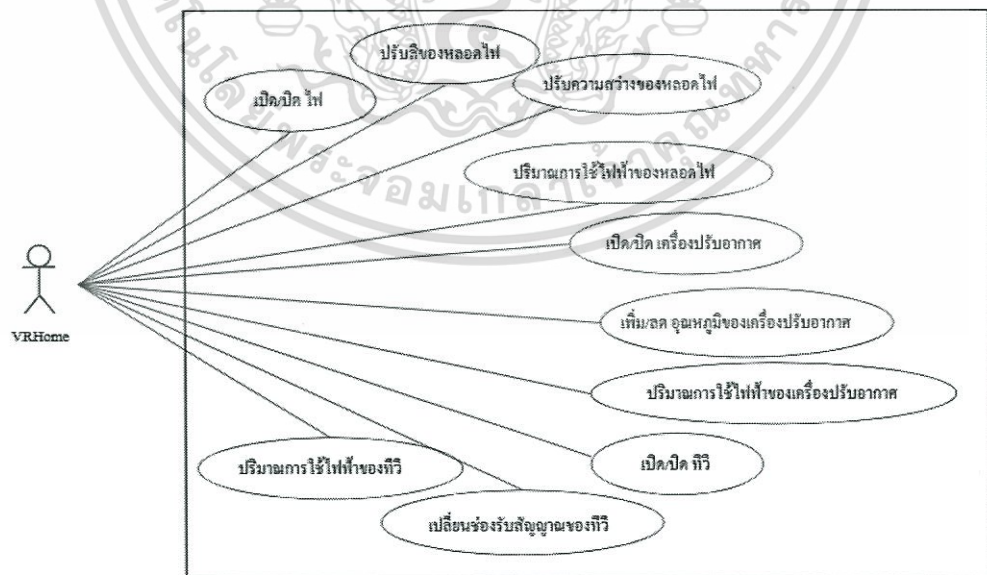
- VRHome สามารถทำงานได้ด้วย Google Chrome
- สถานะของอุปกรณ์ภายในโลกความจริงเสมือนมีการอัปเดตให้ตรงกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 Use case diagram



ภาพที่ 4.2.1 Use case diagram ของการใช้งาน VRHome



HEMS services

ภาพที่ 4.2.2 Use case diagram ของการใช้งาน HEMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การออกแบบส่วนต่อประสานกราฟฟิกกับผู้ใช้

ในส่วนของการออกแบบมุมมองในโลกสามมิตินั้นจะมีความแตกต่างกับมุมมองสามมิติที่เราคุ้นเคยทำให้ต้องใช้เวลาในการเรียนรู้และเข้าใจถึงประสบการณ์การใช้งาน VR เราจึงได้ทดลองหลากหลายวิธีที่ A-frame พอจะทำได้ ซึ่งในกลุ่มนักพัฒนาที่ใช้ A-frame นั้นจะมีการแลกเปลี่ยนความคิดการสร้าง VR อยู่ตลอดเวลาทำให้บางผลงานนั้นสามารถนำมาปรับใช้กับตัวโครงงานชิ้นนี้ได้ รวมทั้งยังใช้เวลาหาประสบการณ์จากแอปพลิเคชันที่มีอยู่ใน App Store โดยใช้ Google Daydream เป็นเครื่องมือ

แต่สิ่งแรกที่เราพบ คือ Google Daydream นั้นไม่สามารถรับรู้การเคลื่อนที่ของเราได้ เพราะเป็น Headset แบบ 3DoF ดังนั้นเราจึงใช้วิธีสร้างจุดขึ้นมานบนพื้นเพื่อรับคำสั่งการย้ายตำแหน่งมุมมองของผู้ใช้ ทำให้เราสามารถเห็นจากมุมมองที่ต่างกันได้ แต่ประสิทธิภาพของ A-Frame หรือ Framework อื่นๆ ที่ใช้เพื่อสร้าง VR ที่สามารถทำงานบนเว็บเบราว์เซอร์นั้นยังคงติดปัญหาเรื่องคุณภาพของภาพเนื่องจากตัวขับเคลื่อนในด้านการประมวลผลของเว็บเบราว์เซอร์ไม่ได้ออกแบบมาเพื่อการใช้งาน VR ซึ่งเมื่อเทียบกับแอปพลิเคชันอื่นๆ ยกตัวอย่างเช่น Google Daydream เองนั้นจะเห็นได้ชัดว่าจะมีคุณภาพที่สูงกว่ามาก เนื่องจากถูกออกแบบมาให้ใช้การประมวลผลประสิทธิภาพสูงอยู่แล้ว ส่วนของการปฏิสัมพันธ์จะมีลักษณะการคลิกอิเว้นท์คล้ายๆ กับหน้าเว็บปกติจึงไม่มีปัญหาแต่อย่างใด

4.4 อุปกรณ์และฟังก์ชันที่สามารถควบคุมได้ใน VRHome

ในโปรเจกต์นี้ใช้อุปกรณ์เพียงสามชนิดเพื่อมาทดลองการใช้งาน ได้แก่

- 1) โทรทัศน์
- 2) หลอดไฟ LED ปรับสีได้
- 3) เครื่องปรับอากาศ

โดยอุปกรณ์แต่ละชนิดประกอบฟังก์ชันการทำงานดังนี้

1. โทรทัศน์

Function	Value
เปิด/ปิด โทรทัศน์	True หรือ False
เปลี่ยน Channel	1 ถึง 6
เพิ่ม/ลด Volume	0 ถึง 100

ตารางแสดงฟังก์ชันการทำงานของโทรทัศน์

2. หลอดไฟ LED

Function	Value
เปิด/ปิด หลอดไฟ	True หรือ False
เปลี่ยนสี	ขาว เหลือง ฟ้า
เพิ่ม/ลด ความสว่าง	0 ถึง 100

ตารางแสดงฟังก์ชันการทำงานของหลอดไฟ LED

3. เครื่องปรับอากาศ

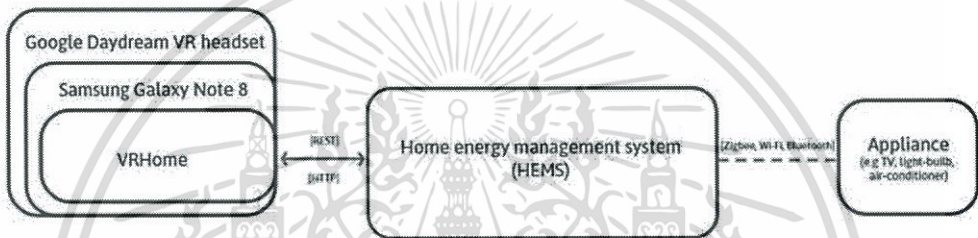
Function	Value
เปิด/ปิด เครื่องปรับอากาศ	True หรือ False
เพิ่มลดอุณหภูมิ	23-30 °C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางแสดงฟังก์ชันการทำงานของเครื่องปรับอากาศ

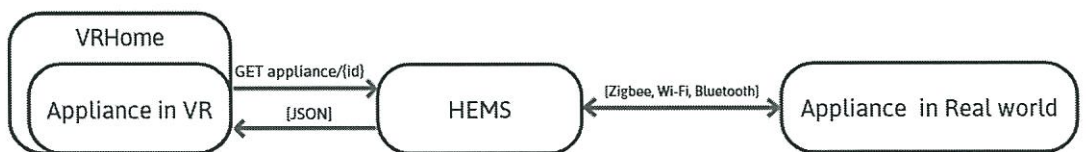
4.5 ภาพรวมของระบบ

การทำงานจะเริ่มจากเปิดเว็บเบราว์เซอร์บนสมาร์ตโฟนแล้วเข้า URL ของ “VRHome” แล้วเว็บเบราว์เซอร์จะแสดง VR ซึ่งแสดงภาพห้องจำลอง และในโลก VR จะมีอุปกรณ์อยู่สามชนิดคือ หลอดไฟ เครื่องปรับอากาศ และทีวี และเพื่อการใช้งานที่ดีขึ้นเราจึงต้องมี Daydream headset ทำให้เราสามารถสร้างปฏิสัมพันธ์กับวัตถุในโลก VR ซึ่งวัตถุเหล่านี้ทำงานโดยการเชื่อมต่อสถานะและข้อมูลของอุปกรณ์แต่ละตัวด้วยการแลกเปลี่ยนข้อมูลกับ HEMS ดังตัวอย่างในภาพที่ 4.1.1



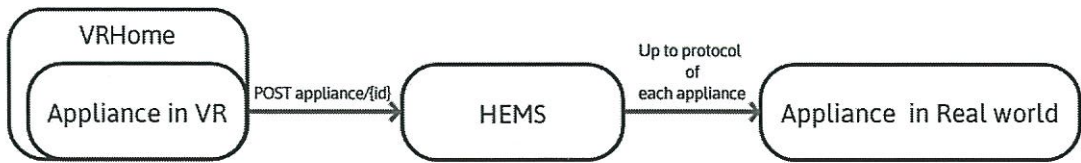
ภาพที่ 4.5.1 แผนภาพการทำงานระหว่าง VRHome กับ HEMS

กล่าวคือเมื่อเราใช้เว็บเบราว์เซอร์เปิดหน้า VRHome ขึ้นมาแล้วหน้าเว็บจะทำการดึงข้อมูลปัจจุบันของอุปกรณ์ด้วย method GET ไปยัง HEMS แล้ว HEMS จะส่งสถานะที่อัปเดตที่สุดกลับมาเพื่อเชื่อมสถานะของอุปกรณ์จริงๆ เข้ากับโลก VR และเมื่อเราทำการสั่งเปิด/ปิด VRHome ก็จะมีการส่ง request ด้วยวิธี POST ไปยัง HEMS เพื่อให้ HEMS สั่งงานอุปกรณ์ภายในครัวเรือนตามภาพที่ 4.1.2 และภาพที่ 4.1.3 ตามลำดับ



ภาพที่ 4.5.2 แผนภาพการดึงข้อมูลจาก HEMS เพื่อเชื่อมต่อสถานะของอุปกรณ์ใน VRHome

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.5.3 แผนภาพการส่งคำสั่งจากอุปกรณ์ใน VRHome ไปยัง HEMS

URL	HTTP Verb	Description
api/preferences/	GET	Return the current preferences
api/preferences/	POST	Create new preferences
api/preferences/{id}	PUT	Update preference that has id
api/preferences/{id}	DELETE	Delete preference that has id
api/rules/	GET	Return all rules
api/rules/	POST	Create a new rule
api/rules/{id}	PUT	Update rules that has id
api/rules/{id}	DELETE	Delete rules that has id
api/appliances/	GET	Get a list of appliances
api/appliances/	POST	Create a new appliance
api/appliances/{id}	PUT	Update appliance that has id
api/appliances/{id}	DELETE	Delete appliance that has id
api/appliances/{id}/info	GET	Get appliance more detailed info that has id
api/appliances/{id}/command	POST	Command appliance that has id

ตาราง API ของ HEMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การพัฒนาเว็บแอปพลิเคชัน VRHome ที่สามารถใช้งานบนเว็บเบราว์เซอร์

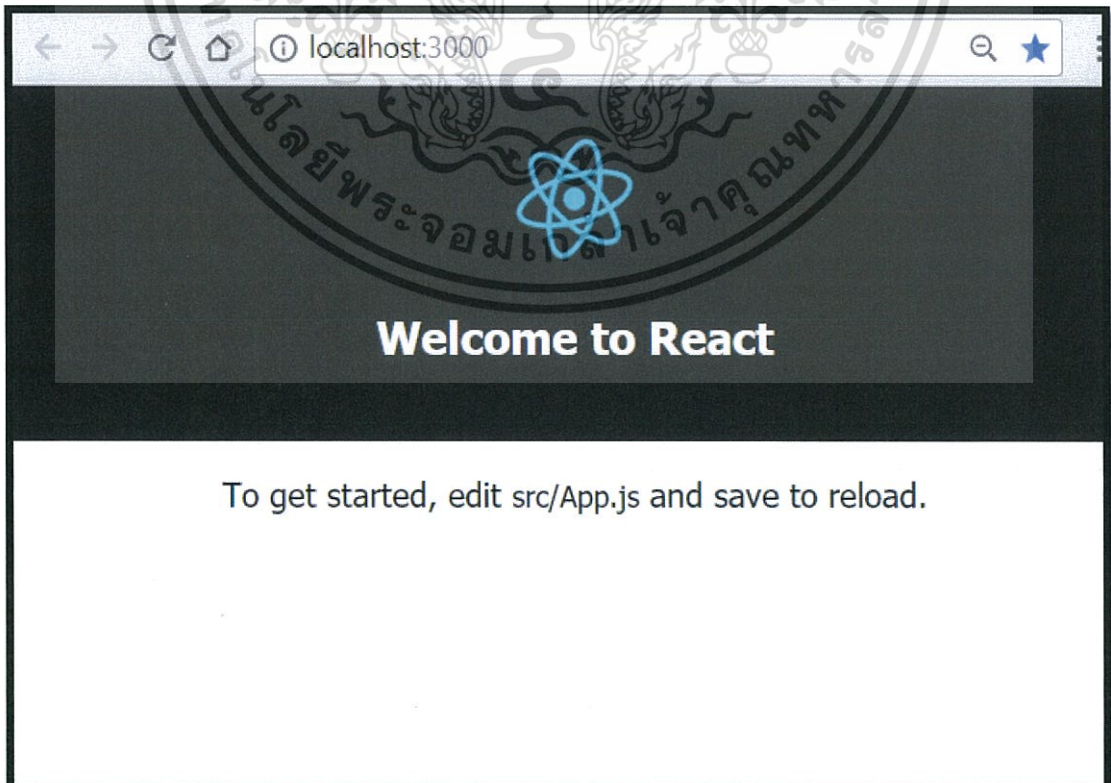
ในโครงการนี้ใช้ React Framework เพื่อช่วยทำให้ขั้นตอนในการพัฒนาแอปพลิเคชัน VR ง่ายขึ้น โดยในการสร้างโปรเจกต์ที่ทำงาน โดยการส่งหน้าเว็บให้ผู้ใช้งานนั้นเราได้เลือกใช้ create-react-app ในการสร้างโปรเจกต์ซึ่งเป็นซอฟต์แวร์ที่ใช้สำหรับสร้าง React ที่ถูกพัฒนาโดย Facebook

5.1 การสร้างโปรเจกต์ด้วย create-react-app

create-react-app เป็นเครื่องมือไว้สำหรับสร้าง React Application ที่ได้มีการตั้งค่าให้สามารถพัฒนาเว็บแอปพลิเคชันด้วย React และช่วยลดขั้นตอนในการตั้งค่าให้กับโปรเจกต์ โดยการเรียกใช้งานเราจำเป็นต้องติดตั้ง NodeJS แล้วเข้าสู่ Command prompt แล้วพิมพ์คำสั่ง

```
npm create-react-app myproject  
cd myproject  
npm start
```

แล้วจะปรากฏหน้าเว็บดังภาพที่ 5.1.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 5.1.1 ภาพหน้าเว็บจากการสร้างโปรเจกต์ด้วย create-react-app

```

1  import React, { Component } from 'react'; 7.7K (gzipped: 3.3K)
2  import logo from './logo.svg';
3  import './App.css';
4
5  class App extends Component {
6    render() {
7      return (
8        <div className="App">
9          <header className="App-header">
10         <img src={logo} className="App-logo" alt="logo" />
11         <h1 className="App-title">Welcome to React</h1>
12       </header>
13       <p className="App-intro">
14         To get started, edit <code>src/App.js</code> and save to reload.
15       </p>
16     </div>
17   );
18 }
19
20
21 export default App;

```

ภาพที่ 5.1.2 ภาพโค้ดที่แสดงหน้าแรกจากไฟล์ src/App.js

เท่านี้เราก็ได้ React Application มาที่สามารถเข้าได้ผ่าน <http://localhost:3000> และสามารถพัฒนาส่วนต่างๆ ลงไปตามความต้องการ ต่อไปเราจำเป็นต้องติดตั้ง package ของ A-frame เพื่อให้เราสามารถเรียกใช้ส่วนต่างๆ ของ A-Frame ได้

5.2 การติดตั้ง A-frame ลงใน React application

หลังจากได้ React มาแล้วเราจำเป็นต้องติดตั้ง aframe และ aframe-react เพื่อให้ A-frame สามารถใช้งานร่วมกับ React โดยติดตั้ง package ที่เหล่านี้ได้โดยได้โดยเข้าไปที่ Command prompt แล้วพิมพ์คำสั่ง

```
npm install --save aframe aframe-react
```

โดยแพ็คเกจ aframe คือตัว library ที่รวมฟังก์ชันการใช้งานองค์ประกอบต่างๆ ของ VR บนเว็บเบราว์เซอร์ได้ และ aframe-react เป็นเพียง API ที่ทำหน้าที่ครอบ aframe ไว้เพื่อทำให้เรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถใช้ aframe ได้ในลักษณะ component ที่ทำให้เราสามารถสร้าง reusable component ได้ เพื่อลดความซับซ้อนของการพัฒนา

โดยเมื่อติดตั้งเสร็จจึงได้ตามโค้ดในภาพที่ 3.4.1 มาดัดแปลงเพื่อทำให้เป็น component ทำให้เราสามารถใช้งานได้ใน React application นี้ได้ตามภาพที่ 4.3.1 โดยที่ต่างกันอย่างเห็นได้ชัดคือทุก Tag ของ React จะขึ้นต้นด้วยตัวพิมพ์ใหญ่ แม้ aframe-react จะใช้คำคล้ายๆ กันแต่จะขึ้นต้นด้วยตัวใหญ่ทั้งหมด เป็นส่วนที่ต้องระวังให้ดีเพราะว่าใน ReactDOM เราจะสามารถใช้ได้ทั้งพิมพ์ใหญ่หรือพิมพ์เล็กแต่ที่ต่างกันคือความเป็น component ที่สามารถโปรแกรมได้ใน React



```

15 App.js
1 import React, { Component } from "react"; 7.7K (gzipped: 3.3K)
2 import { Entity, Scene } from "aframe-react"; 3.6K (gzipped: 1.5K)
3 import "aframe"; 1M (gzipped: 291.1K)
4 class App extends Component {
5   render() {
6     return (
7       <Scene stat>
8         <Entity
9           primitive="a-box"
10          position="-1 0.5 -3"
11          rotation="0 45 0"
12          color="#4CC3D9"
13          shadow
14        />
15        <Entity
16          primitive="a-sphere"
17          position="0 1.25 -5"
18          radius="1.25"
19          color="#EF2D5E"
20          shadow
21        />
22        <Entity
23          primitive="a-cylinder"
24          position="1 0.75 -3"
25          radius="0.5"
26          height="1.5"
27          color="#FFC65D"
28          shadow
29        />
30        <Entity
31          primitive="a-plane"
32          position="0 0 -4"
33          rotation="-90 0 0"
34          width="4"
35          height="4"
36          color="#7BC8A4"
37          shadow
38        />
39        <Entity primitive="a-sky" color="#ECECEC" />
40      </Scene>
41    );
42  }
43 }
44 export default App;

```

ภาพที่ 5.2.1 ภาพการเขียนโค้ดการใช้ A-frame ด้วย React

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งผลลัพธ์ของก็จะได้เป็นเหมือนกับภาพที่ 3.4.2

5.3 การใช้งาน A-frame กับ React เพื่อเรียกใช้ API

โดยปกติเมื่อเริ่มใช้งานตัว Avatar ที่เป็นตัวแทนของอุปกรณ์ต่างๆ ภายในบ้านจะต้องมีข้อมูลของสถานะปัจจุบันของอุปกรณ์ที่เป็นตัวจริง โดยข้อมูลเหล่านี้จะทำการเก็บและอัปเดตอยู่ตลอดเวลาที่ฐานข้อมูลของ HEMS ซึ่งเราจำเป็นต้องส่ง request ด้วยวิธี GET เพื่อดึงข้อมูลปัจจุบันของอุปกรณ์นั้นมาเพื่ออัปเดตเข้ากับอุปกรณ์ใน VR ตามภาพที่ 4.1.2

ในทางตรงกันข้าม ถ้าเราจะส่งงาน Avatar ให้เริ่มทำงานตามคำสั่ง เราจำเป็นต้องส่ง request ด้วยวิธี POST เพื่อส่งคำสั่งไปยัง HEMS แล้ว HEMS จะส่งต่อคำสั่งไปยังอุปกรณ์เพื่อส่งงานอุปกรณ์นั้นตามภาพที่ 4.1.3

ดังนั้น Avatar จำเป็นต้องสามารถรับคำสั่งจากผู้ใช้งานได้ ยกตัวอย่างเช่นการคลิก และเมื่อรับคำสั่งมาแล้วต้องสามารถส่งคำสั่งนั้นไปยัง HEMS ได้อีกด้วย โดยในขั้นตอนทั้งหมดเราจะเริ่มจากขั้นตอนต่างๆ ดังนี้

5.3.1 การเพิ่มส่วนควบคุมลงใน A-frame

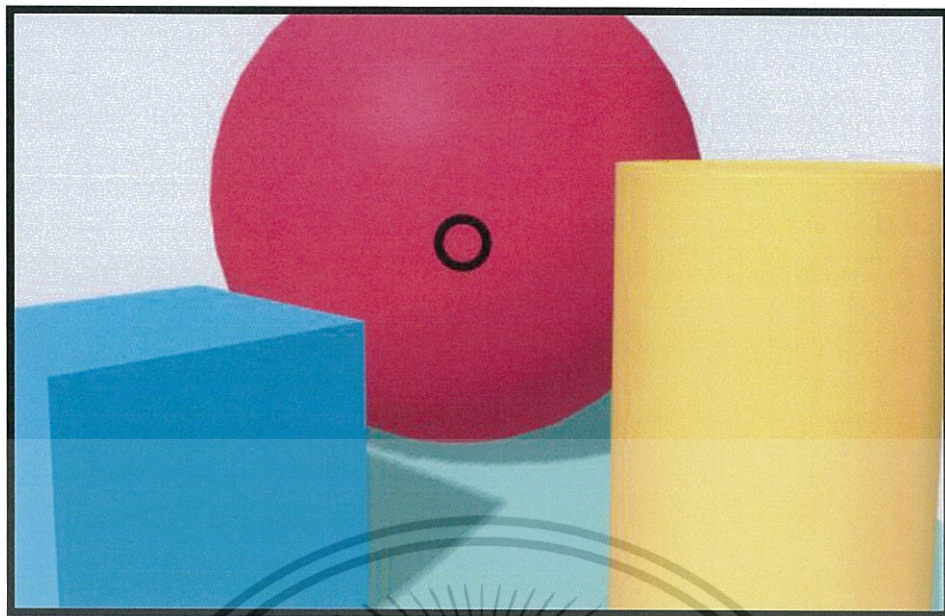
การเพิ่มส่วนควบคุมจะมีอยู่ด้วยกันสองวิธี

5.3.1.1 การเพิ่ม cursor ซึ่งเป็นส่วนที่ช่วยทำให้เราสามารถมีปฏิสัมพันธ์กับวัตถุใน VRHome ได้

```
<Entity camera position="0 1.6 0" look-controls>
  <Entity
    cursor="fuse: true; fuseTimeout: 500"
    position="0 0 -1"
    geometry="primitive: ring; radiusInner: 0.02; radiusOuter: 0.03"
    material="color: black; shader: flat"
  />
</Entity>
```

ภาพที่ 5.3.1.1 โค้ดการเรียกใช้ cursor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



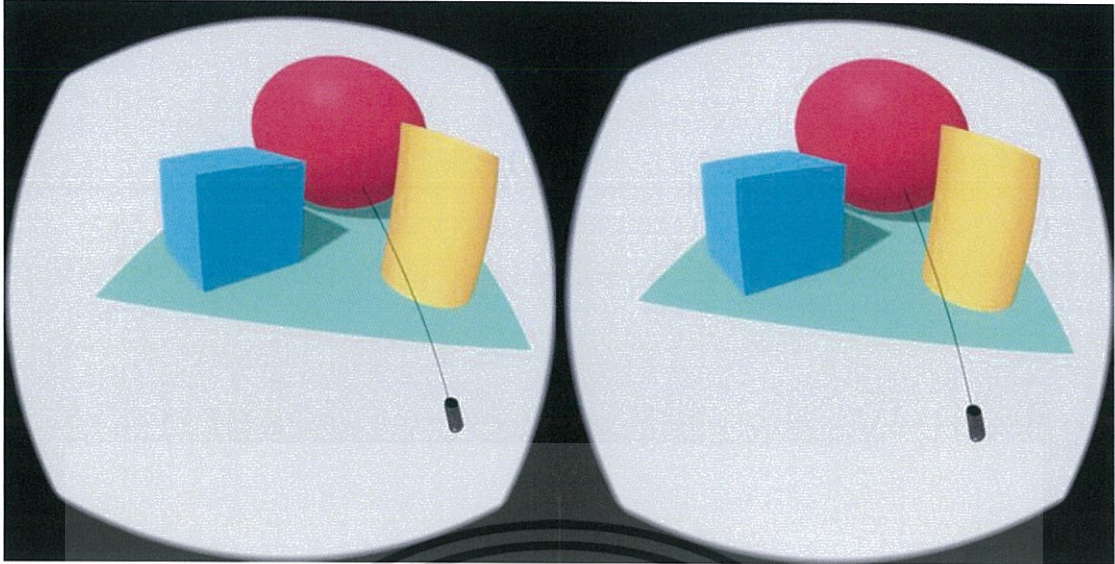
ภาพที่ 5.3.1.1.2 ภาพตัวอย่างการใช้ cursor (วงกลมสีดำ)

5.3.1.2 การใช้ VR controller หรือ VR headsets ต่างๆ เชื่อมต่อเว็บเบราว์เซอร์โดยใช้ A-frame โดย VR headsets ที่รองรับ “HTC Vive” “Oculus Rift” “Google Daydream” “Samsung GearVR” และ “Windows Mixed Reality”

```
<Entity
  daydream-controls
  raycaster="showLine: true; far: 100"
  line="color: black; opacity: 1"
/>
```

ภาพที่ 5.3.1.2.1 โค้ดตัวอย่างการเรียกใช้ Google Daydream

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 5.3.1.2.2 ตัวอย่างการใช้ Google Daydream เป็นตัวชี้

5.3.2 การเพิ่ม Events ให้กับ Avatar

เราจำเป็นต้องเพิ่ม Event listener เพื่อให้วัตถุในโลกความจริงเสมือนนั้นสามารถรับคำสั่งจากผู้ใช้งานได้ โดยการเพิ่ม events เราสามารถทำได้เหมือนกับการเพิ่ม events ใน React component อื่นๆ ดังภาพที่ 5.3.2.1

```

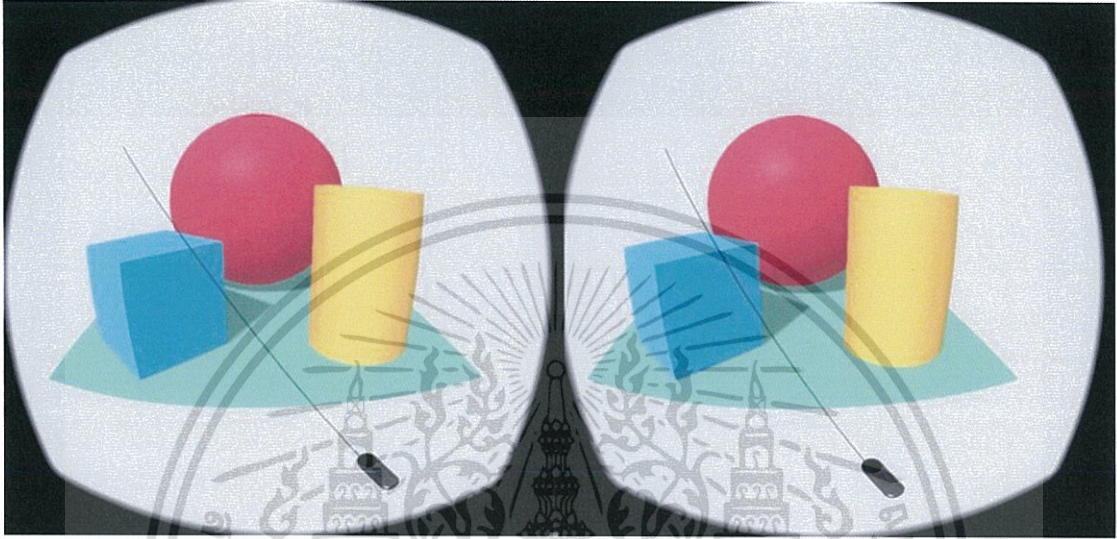
4  class App extends Component
5
6  state = {
7    color: "#4CC3D9"
8  };
9
10 render() {
11   return (
12     <Scene>
13       <Entity
14         primitive="a-box"
15         position="-1 0.5 -3"
16         rotation="0 45 0"
17         color={this.state.color}
18         shadow
19         events={{
20           click: () =>
21             this.setState({
22               color: [this.state.color == "#4CC3D9" ? "green" : "#4CC3D9"]
23             })
24         }}
25       />

```

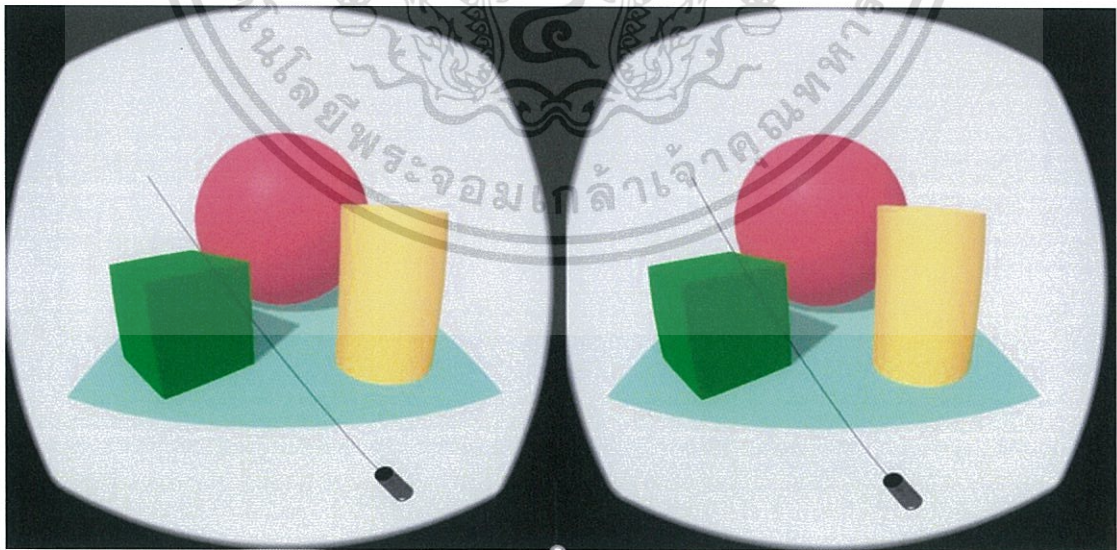
ภาพที่ 5.3.2.1 ภาพตัวอย่างการเพิ่ม events ให้กับ avatar

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในภาพจะเป็นการรับ events การคลิกเพื่อเปลี่ยนสีกลับไปมาระหว่างสีเขียวกับสีฟ้าดัง
ภาพที่ 5.3.2.2 และภาพที่ 5.3.2.3



ภาพที่ 5.3.2.2 ภาพก่อนถูกคลิก

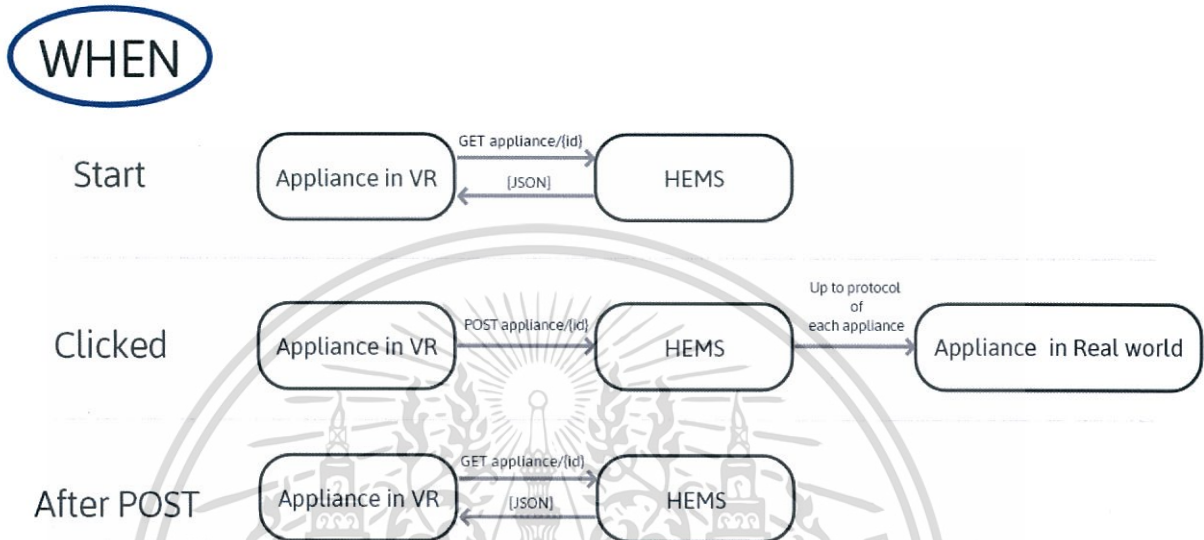


ภาพที่ 5.3.2.3 ภาพหลังถูกคลิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3.3 การอินเตอร์เฟสกับ RESTful API

โครงการนี้ใช้ RESTful API ในการติดต่อและแลกเปลี่ยนข้อมูลระหว่างแอปพลิเคชัน โดยจะเป็นการเรียกใช้ข้อมูลของแต่ละอุปกรณ์ที่ต้องเชื่อมต่อกับอุปกรณ์ภายในระบบ ซึ่งจะถูกเรียกใช้งานจาก Event ที่เกิดจากผู้ใช้



ภาพที่ 5.3.3.1 แผนภาพการแสดงผลการแลกเปลี่ยนข้อมูลด้วย RESTful API

โดยในที่นี้เราจะยกตัวอย่างการทำงานของหลอดไฟ โดยการทำงานจะเริ่มจากการเปิดหน้าเว็บขึ้นมาแล้วทำการ GET ข้อมูลสถานะของหลอดไฟจาก HEMS เพื่อให้ตัวอุปกรณ์กับตัว Avatar ของอุปกรณ์มีสถานะการเปิดไฟที่ตรงกัน และเมื่อผู้ใช้งานคลิกไปที่ Avatar จากฝั่ง VRHome ก็จะทำการ POST เพื่อส่งคำสั่งไปยัง HEMS เพื่อให้ HEMS ทำการสั่งเปิดอุปกรณ์ที่เชื่อมต่ออยู่ จากนั้นจะทำการ GET อีกครั้งหลังจากที่ส่ง POST เสร็จแล้วเพื่อยืนยันสถานะของอุปกรณ์ก่อนที่จะมาเปลี่ยนสถานะของ Avatar ภายใน VRHome

โดยเริ่มจากประกาศ state ขึ้นมารับค่า 3 ค่าโดย

- id คือ state ที่เก็บค่าหมายเลขของอุปกรณ์ที่อยู่ใน HEMS
- lamp คือ state ที่บ่งบอกว่าหลอดไฟเปิดหรือปิด
- color คือ state ที่เก็บค่าของสีของวัตถุ เมื่อไฟเปิดหรือปิด Avatar จะสีต่างกัน

เมื่อเริ่มเปิดเว็บขึ้นมา React จะทำการ GET ข้อมูลของหลอดไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

getstateLamp() {
  let stateLamp;
  fetch(
    "http://[REDACTED]/api/appliances/" +
    this.state.id +
    "/info/?format=json&username=test&api_key=[REDACTED]"
  )
    .then((response) => response.json())
    .then((data) => {
      stateLamp = data[0].value == 1 ? true : false;
    })
    .then(() =>
      this.setState({
        lamp: stateLamp == 1 ? true : false,
        color: stateLamp ? "#4CC3D9" : "green"
      })
    );
}
componentDidMount() {
  this.getstateLamp.bind(this);
}

```

ภาพที่ 5.3.3.2 โค้ดที่ทำงานเมื่อเปิดเว็บ

เพราะ React component มีการทำงานที่เรียกว่า lifecycle ซึ่งเราสามารถใช้ประโยชน์จาก lifecycle เพื่อกำหนดช่วงเวลาในการทำงานของฟังก์ชันได้ และ componentDidMount จะเป็นฟังก์ชันที่ทำงานหลังจาก ReactDOM ได้ Render หน้าเว็บเสร็จ ทำให้เมื่อเปิดเว็บมาตัวเว็บก็จะทำการตรวจสอบข้อมูลของอุปกรณ์ว่าสถานะตรงกันหรือไม่ เป็นอันดับแรก

ต่อมาเมื่อเรากดไปที่ Avatar ก็จะมีการเรียกฟังก์ชันที่ใช้เรียกฟังก์ชันส่ง POST ให้ HEMS เพื่อให้ HEMS ส่งงานไปยังอุปกรณ์ภายในบ้าน หลังจากนั้นจะเห็นว่าจะมีการเรียกฟังก์ชัน getstateLamp เพื่อยืนยันว่าคำสั่งที่เพิ่งส่งไปก่อนหน้านี้มันได้ถูกทำงานหรือยัง ตามภาพที่ 3.4.3.3

```

toggleLamp() {
  fetch(
    "http://[redacted]/api/appliances/" +
    this.state.id +
    "/command/?format=json&username=t[redacted]",
    {
      method: "POST",
      headers: {
        Authorization: [redacted]
        "Content-Type": "application/json"
      },
      body: JSON.stringify({
        cmd_name: "set_On_Off",
        parameters: {
          setTo: this.state.lamp ? 0 : 1
        }
      })
    }
  ).then(this.getstateLamp.bind(this));
}

```

ภาพที่ 5.3.3.3 ภาพโค้ดการส่ง POST ไปที่ HEMS

5.4 การใส่โมเดลให้กับ Components

นอกจากเราจะกำหนดคุณสมบัติและฟังก์ชันต่างๆ ของแต่ละ Component ใน React แล้วเรายังจำเป็นต้องใส่โมเดลสามมิติ เพื่อเป็นส่วนที่ช่วยแสดงผลให้เห็นว่าแต่ละ Component นั้นแสดงถึงสิ่งใด และเพื่อจำลองสภาพแวดล้อมให้ใกล้เคียงกับสภาพแวดล้อมในโลกแห่งความเป็นจริงให้มากที่สุด

โดยใน A-frame เราใช้โมเดลที่มีนามสกุลไฟล์ gltf ซึ่งเป็นมาตรฐานของไฟล์โมเดลสามมิติในปัจจุบันที่นิยมใช้ผ่านเว็บเบราว์เซอร์เพราะในโครงสร้างไฟล์นั้นมีลักษณะคล้ายกับ JSON ซึ่งถูกออกแบบมาเพื่อแสดงผลในรูปแบบสามมิติโดยเฉพาะ ทำให้มีความเร็วในการแสดงผลที่สูงกว่าไฟล์ชนิดอื่นๆ เพราะตัวไฟล์มีขนาดเล็ก และสามารถประมวลผลด้วย WebGL ที่ติดตั้งอยู่บนเว็บเบราว์เซอร์ได้อย่างรวดเร็ว

โดยวิธีการใส่โมเดลเราจำเป็นต้องมีโมเดลที่มีนามสกุลไฟล์ gltf ก่อนแล้วจึงนำมาใช้กับ Component และใส่ใน props ที่ชื่อ gltf-model ตามภาพ

```

<Entity
  gltf-model={this.state.lamp ? "#LIGHT_ON" : "#LIGHT_OFF"}
  position={{ x: -3.642, y: 0, z: 5 }}
  scale={{ x: 2, y: 2, z: 2 }}
  events={{
    click: this.lampShow.bind(this),
    mouseenter: context.mouseIN,
    mouseleave: context.mouseOUT
  }}
/>

```

ภาพที่ 5.4.1 การใส่โมเดล

ซึ่งในแต่ละโมเดลของอุปกรณ์ในการออกแบบของโครงการนี้ ได้ออกแบบให้มีสองรูปแบบ คือ โมเดลแสดงสถานะเปิดและโมเดลแสดงสถานะปิด ซึ่งในตัวอย่างเป็นตัวอย่างโมเดลของหลอดไฟ ซึ่งจะทำตามตัวแปร state.lamp ถ้า state.lamp มีค่าเป็น true หรือหลอดไฟเปิดอยู่ตัวโมเดลจะแสดงรูปของโมเดลหลอดไฟที่เปิด แต่ถ้า state.lamp มีค่าเป็น false หรือหลอดไฟปิดอยู่ตัวโมเดลจะแสดงรูปของโมเดลหลอดไฟที่ปิด

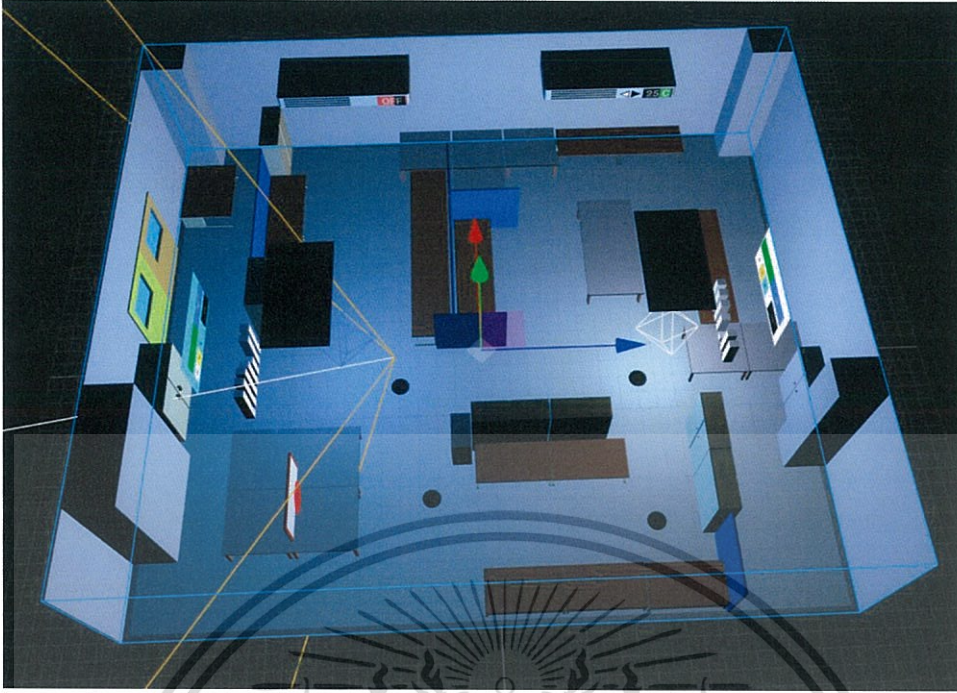
ซึ่งนอกจากเราจะใส่รูปโมเดลที่เป็นไฟล์ที่มีนามสกุล gltf แล้วนั้นเรายังจำเป็นต้องใส่ตำแหน่งของโมเดล และขนาดของโมเดล รวมถึง events ของโมเดลเมื่อรับการคลิกจะเปิดหรือปิดเป็นการทำงานแบบ toggle และเมื่อเลื่อน cursor มากระทบที่โมเดลจะทำให้ cursor นั้นเปลี่ยนสีเพื่อแสดงให้ผู้ใช้งานเห็นว่า cursor นั้นพร้อมที่จะมีปฏิสัมพันธ์กับ โมเดลหรือวัตถุอื่นๆ แล้ว

5.2 VRHome และการใช้งานแต่ละอุปกรณ์

ในหัวข้อนี้จะเป็นการใช้ความรู้จากหัวข้อที่ 5.1-5.4 เพื่อมาผสมผสานกันออกมาเป็นห้องจำลองที่สามารถติดต่อสื่อสารกับระบบ HEMS มาประกอบเข้าด้วยกัน เริ่มจากการสร้างห้องจำลองขนาดใหญ่ออกมาเพื่อสร้างสภาพแวดล้อมของห้องแลปที่ภาควิชาคอมพิวเตอร์ และกำหนดตำแหน่งของห้องและไว้ที่จุด $(X,Y,Z) = (0,0,0)$ ซึ่งเป็นจุดเดียวกับจุดอ้างอิงของ WebGL ตามภาพที่

5.5.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 5.5.1 ภาพห้องจำลองใน VRHome

จากภาพก็จะเห็นว่าตำแหน่งของห้องจำลองจะอยู่ตรงกลางพอดี และเนื่องจากเราต้องการความสมจริงให้กับสภาพแวดล้อม โดยรอบเราจึงใช้ Light component ของ A-frame เพื่อให้แสงกับห้องแลป และยังสร้างเงาที่เกิดจากวัตถุ โดย Light component นี้เราได้ใช้คู่กับโมเดลของหลอดไฟ เพื่อที่จะให้ความรู้สึกว่แสงไฟนั้นมาจากหลอดไฟจริงๆ ช่วยเพิ่มความสมจริง และนอกจากหลอดไฟเรายังมีทีวี และเครื่องปรับอากาศสองตัวที่มีแถบแสดงสถานะการเปิดปิด และสถานอื่นๆ ขึ้นอยู่กับความเหมาะสมของสถานที่และอุปกรณ์

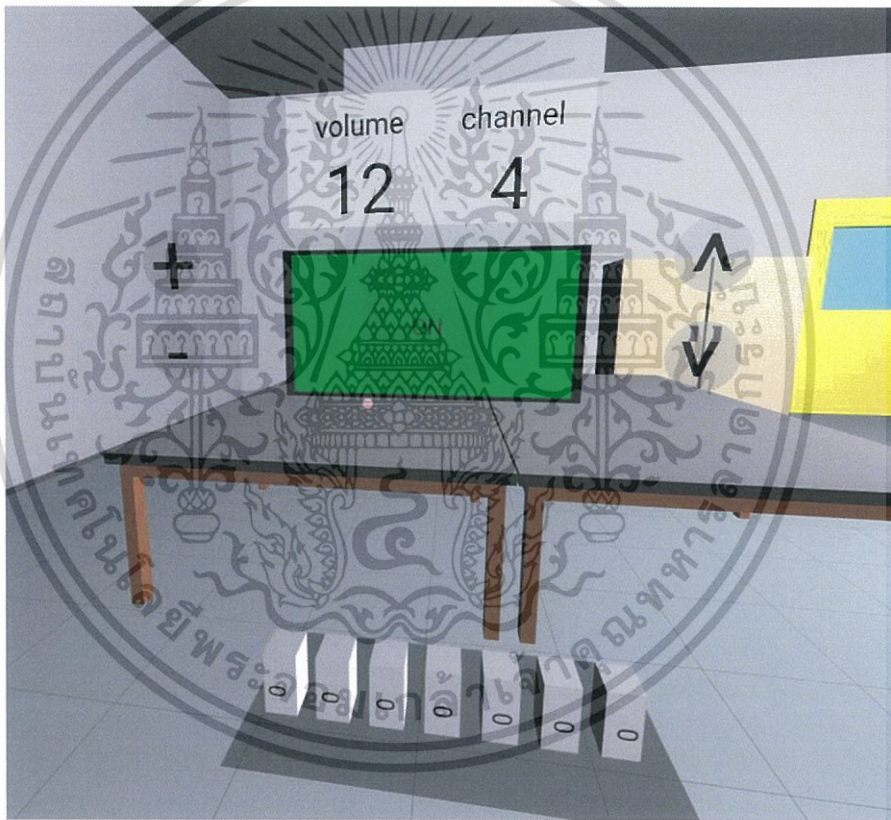
บทที่ 6

ผลการทดลอง

จากการหาประสบการณ์จากใช้ VR แอปพลิเคชันต่างๆ และนำมาปรับใช้ในออกแบบแอปพลิเคชัน ซึ่งจะอธิบายการทำงานเริ่มจาก

6.1 ทิว หรือโทรทัศน์

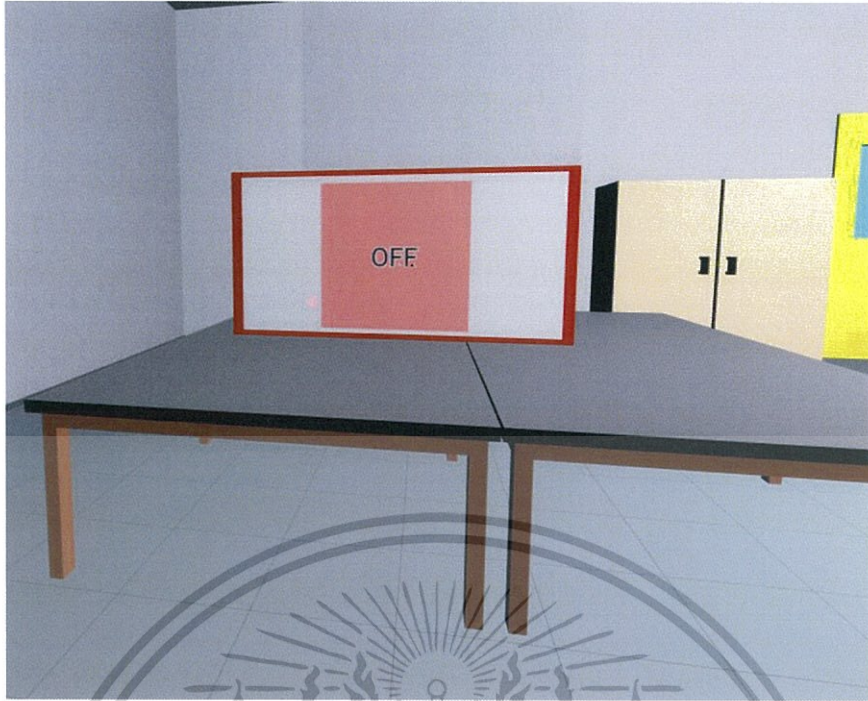
ทีวีจะมีส่วนของการเพิ่มลดเสียงและช่องรับสัญญาณ และแท็บหน้าจอแสดงการเปิดปิด โดยโมเดลของทีวีที่เปิดจะเป็นดังภาพ



ภาพที่ 6.1.1 ภาพทีวีเปิด

โดยในส่วนของการเพิ่มลดเสียงจะอยู่ด้านซ้าย และส่วนของการเปลี่ยนช่องรับสัญญาณจะอยู่ด้านขวา และปุ่มตรงกลางมีไว้เพื่อสั่งเปิดหรือปิด และแผงแสดงสถานะด้านบนไว้แสดงช่องสัญญาณปัจจุบันและระดับเสียงปัจจุบัน และแท็บกราฟด้านล่างแสดงถึงการใช้พลังงานใน 7 วันที่ผ่าน โดยวันปัจจุบันจะเริ่มจากด้านขวาสุด ถัดมาทางซ้ายจะเป็นเมื่อวานและต่อมาเรื่อยๆ ซึ่งเมื่อปิดทีวีแล้วจะแสดงผลดังภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

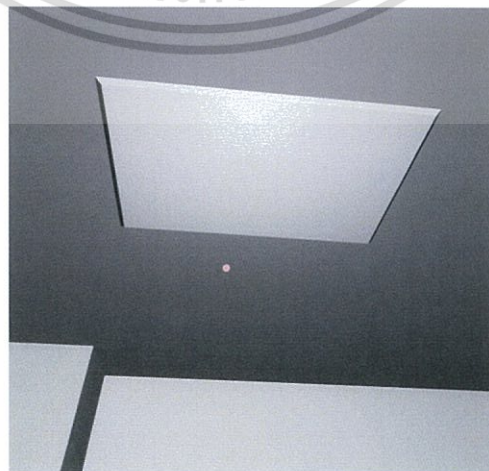


ภาพที่ 6.1.2 ภาพทีวีปิด

โดยจะเห็นได้ว่าแท็บแสดงสถานะและหน้าสำหรับควบคุมต่างๆ จะถูกซ่อนลงไป เพราะว่าในเมื่อทีวีปิดแล้วจะไม่สามารถสั่งงานสิ่งใดกับทีวีได้ ซึ่งลักษณะการทำงานจะคล้ายกับทีวีในโลกของความจริงซึ่งเมื่อถ้าเปิดแล้วเราจะไม่สามารถเพิ่มลดเสียงได้จะรอเพียงคำสั่งเปิดเท่านั้น

6.2 หลอดไฟ

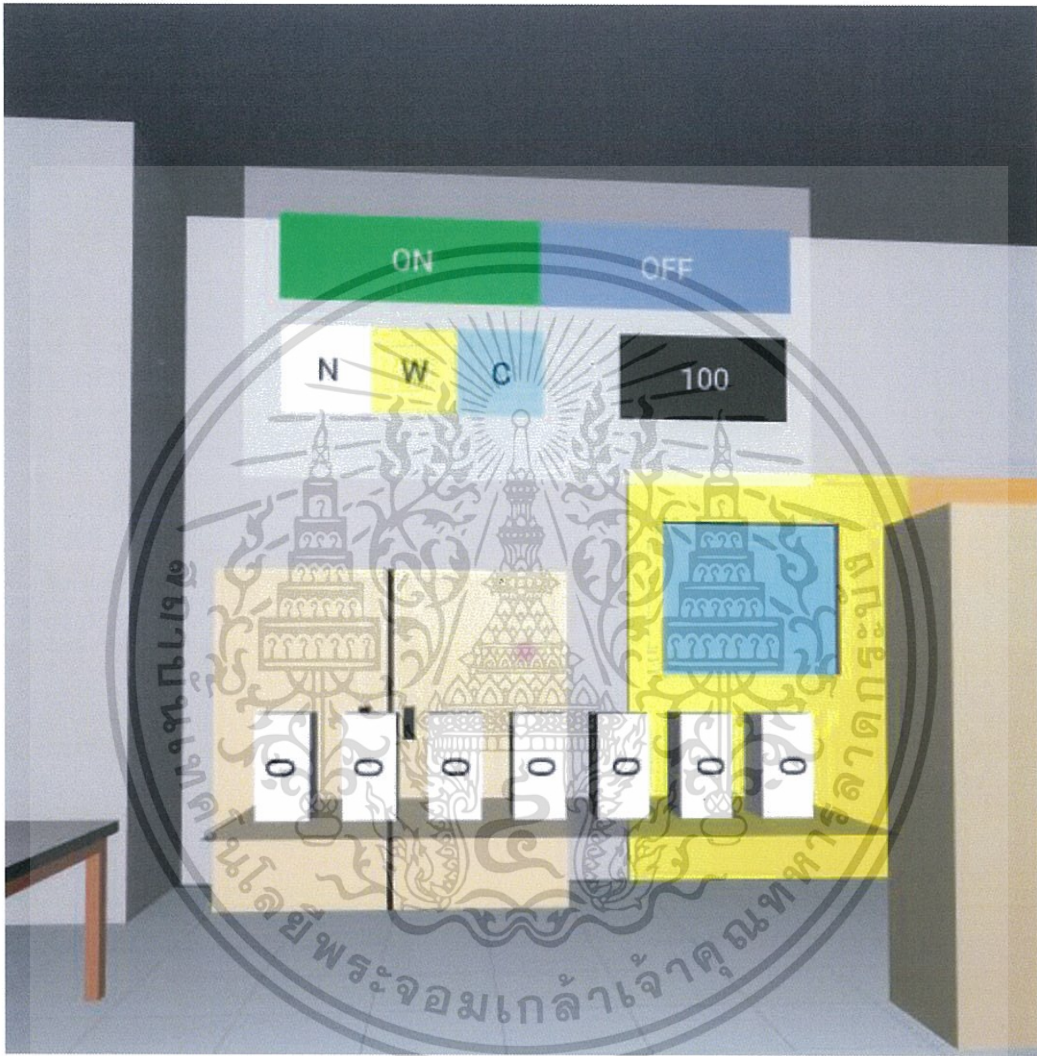
และในส่วนของหลอดไฟในโครงการนี้จะใช้หลอดไฟเพียงสองหลอดซึ่งหลอดไฟที่เราใช้งานนั้นเป็นหลอดไฟ LED ที่เราสามารถปรับค่าสี และความสว่างได้ โดยโมเดลของหลอดไฟจะเป็นตามภาพ



ภาพที่ 6.2.1 หลอดไฟเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

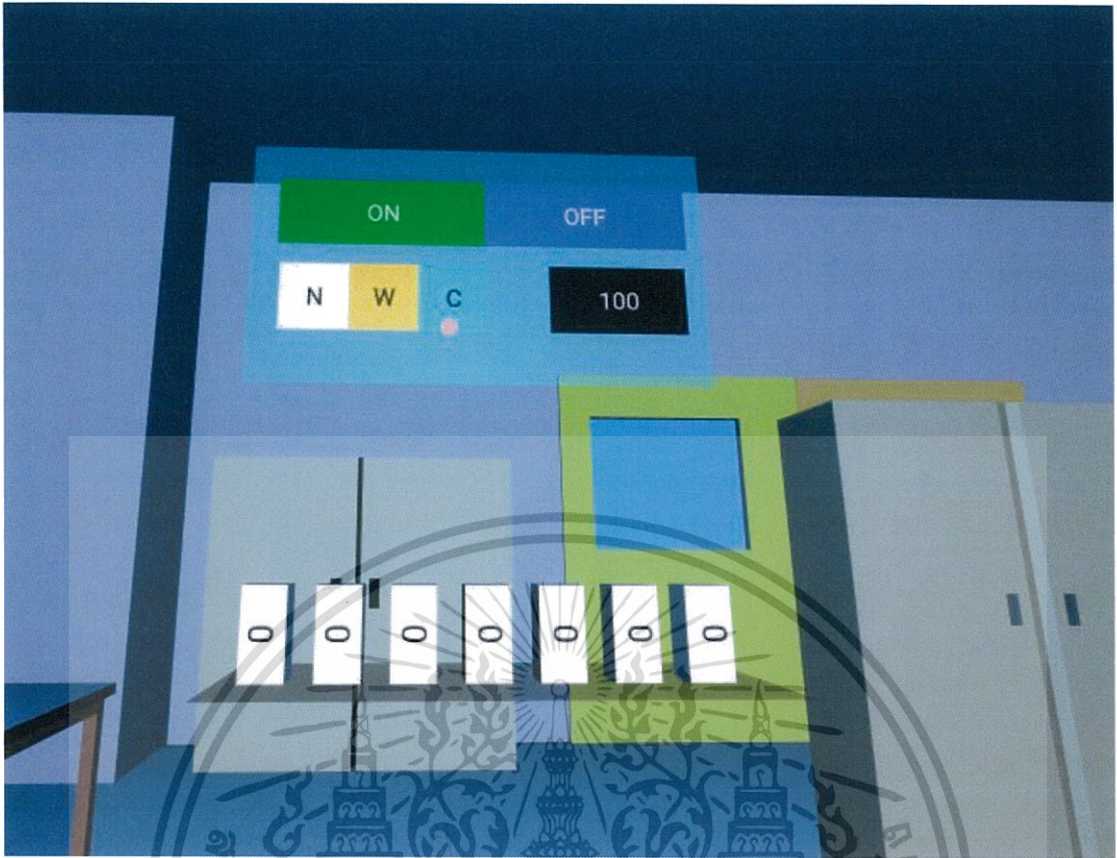
ซึ่งจากภาพจะเห็นว่า โมเดลของหลอดไฟที่เปิดนั้นจะมีสีขาว แสดงว่าหลอดไฟนั้นเปิดทำงานอยู่ ซึ่งตำแหน่งของหลอดไฟจะอยู่ด้านบนเพดานของผู้ใช้ ในลักษณะปกติจะไม่มีการแสดงหน้าต่างควบคุมได้ เพราะจะทำให้บดบังวิสัยทัศน์ได้ แต่หากเราต้องการที่จะสั่งงานเราจำเป็นต้องกดคลิกไปที่โมเดลหนึ่งครั้งเพื่อแสดงหน้าต่างการทำงานดังภาพ



ภาพที่ 6.2.2 แผงควบคุมของหลอดไฟ

จากภาพเราจะเห็นแผงที่มีชื่อว่า ON หรือ OFF เป็นแผงที่ใช้รับคำสั่งเปิดและปิดตามลำดับ และเรายังสามารถเปลี่ยนสีของหลอดไฟได้จากแผงควบคุมที่ชื่อ N W และ C ซึ่งแต่ละแผงควบคุมจะถูกกำหนดสีของแสงที่ต่างกัน โดย N จะเป็นแสงปรกติ W จะเป็นแสงสีเหลืองอบอุ่น และ C เป็นแสงสีบานเย็น ซึ่งเมื่อเรามีปฏิสัมพันธ์กับแผงที่ควบคุมการเปลี่ยนสีอื่นไหนแล้ว แสงของหลอดไฟนั้นจะมีการเปลี่ยนแปลงไปตามคำสั่งนั้นๆ ยกตัวอย่างเช่นการเปลี่ยนเป็นฟ้าหรือ C ดังภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



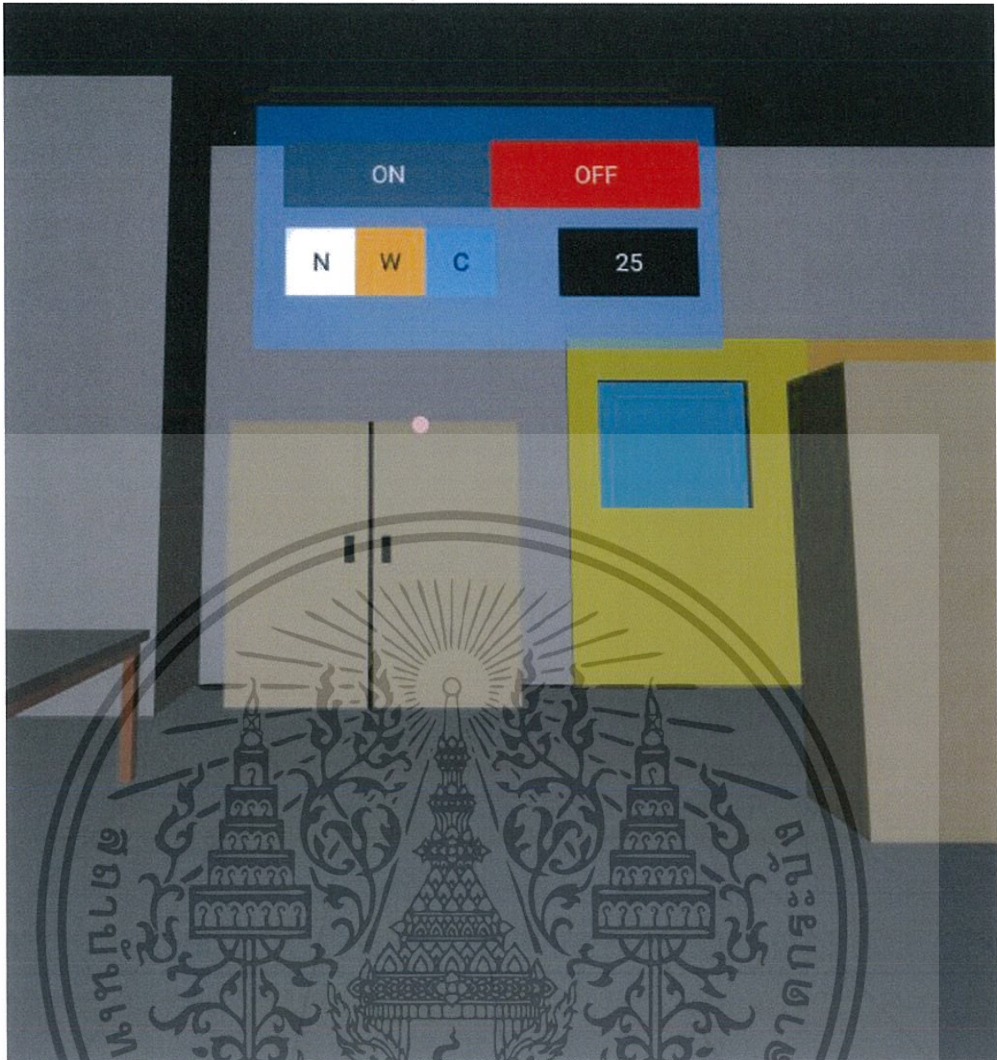
ภาพที่ 6.2.3 ภาพแสดงการเปลี่ยนสีของหลอดไฟ

ซึ่งจะเห็นได้ว่าสภาพแวดล้อมรอบๆ นั้นจะถูกอาบไปด้วยสีฟ้าตามสีของแสงไฟที่เราเปลี่ยน นอกจากนี้เรายังสามารถปรับความเข้มของแสงได้อีกด้วยซึ่งในตัวอย่างในภาพที่ 5.5.6 นั้นได้ใช้ความเข้มของแสงมากที่สุดที่ 100 เปอร์เซ็นต์ โดยเมื่อเรากดคลิกเพื่อปรับความเข้มของแสงจะมีด้วยกัน 4 กรณี คือ

- 1) 100 เปอร์เซ็นต์
- 2) 75 เปอร์เซ็นต์
- 3) 50 เปอร์เซ็นต์
- 4) 25 เปอร์เซ็นต์

โดยเมื่อเราปิดไฟเราจะแสดงความเข้มแสงเพียง 25 เปอร์เซ็นต์ดังภาพ

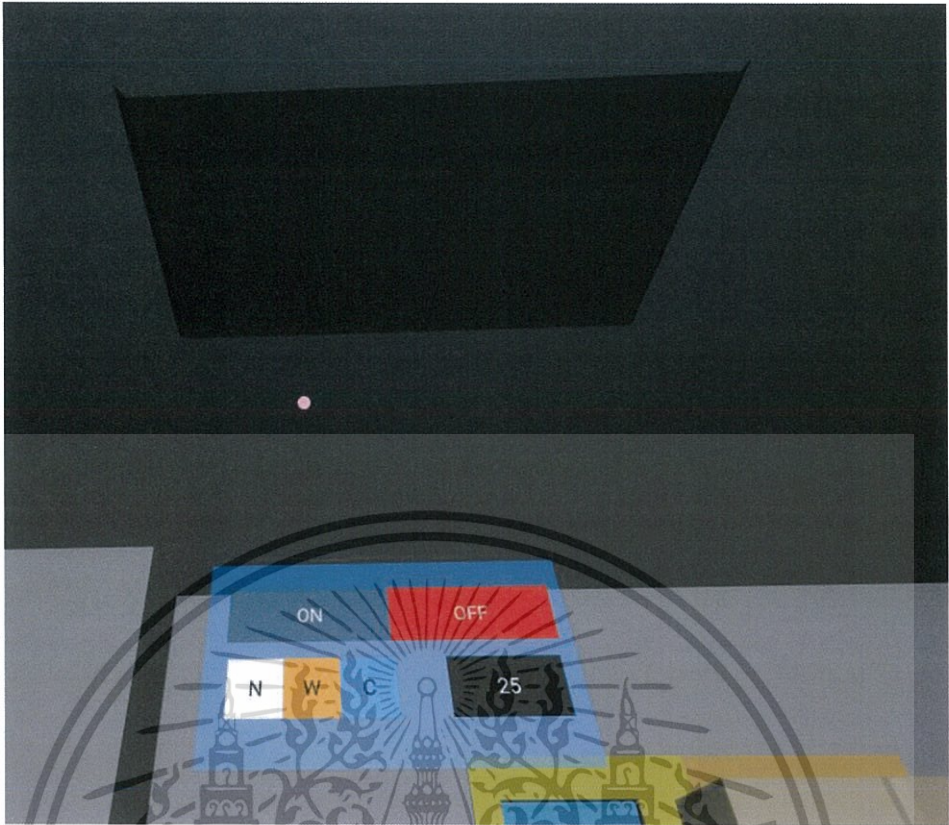
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 6.2.4 ภาพแสดงสภาพห้องตอนปิดไฟ

เพื่อให้เราสามารถมองเห็นวัตถุต่างๆ เราไม่สามารถลดความเข้มของแสงลงเหลือ 0 ได้เพราะจะทำให้เราไม่สามารถมองเห็นวัตถุใดๆ ได้ และเมื่อเราปิดไฟภาพโมเดลจะเปลี่ยนเป็นรูปของโมเดลไฟที่ปิดแล้ว ดังภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

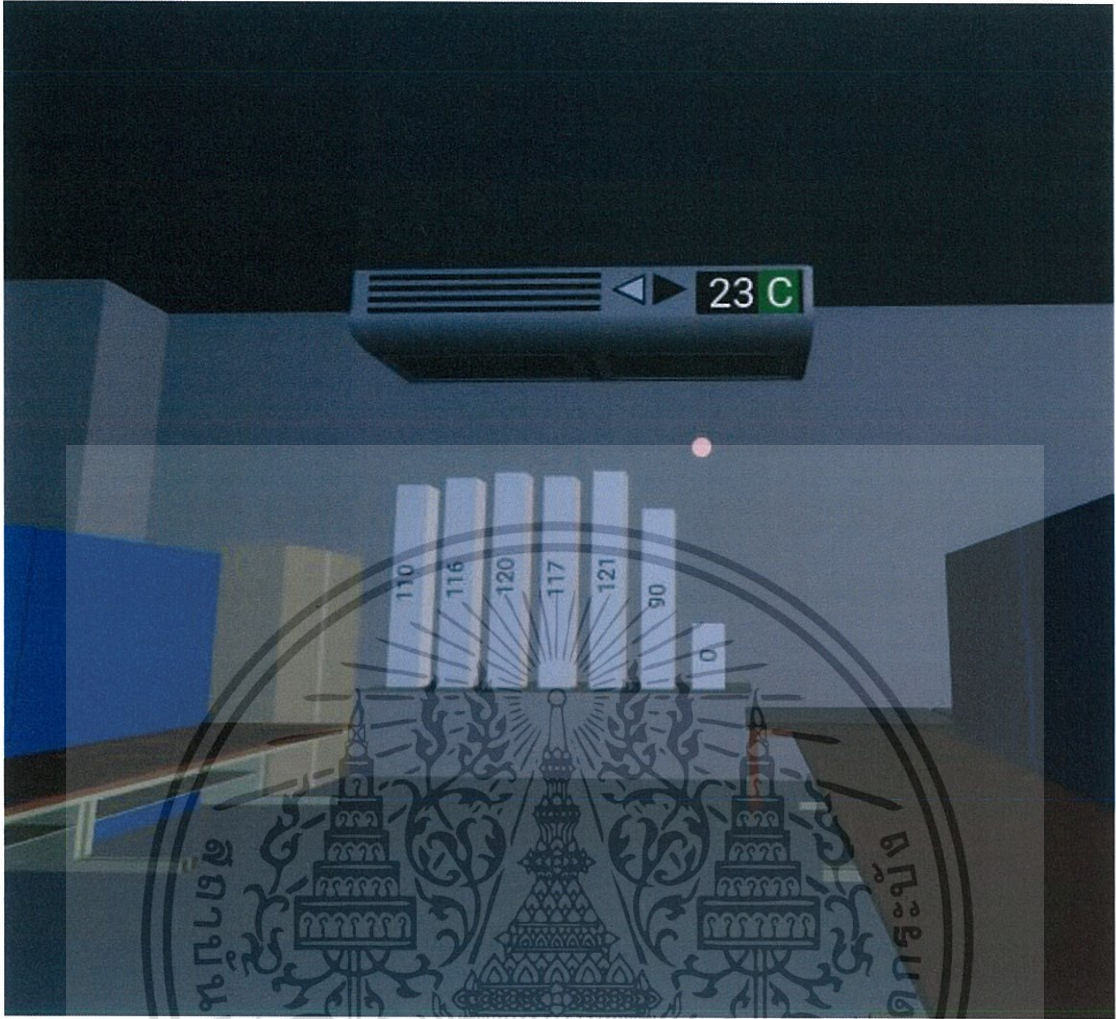


ภาพที่ 6.2.5 ภาพแสดงโมเดลหลอดไฟตอนปิดไฟ

และเมื่อเราปิดไฟไม่ว่าเราจะเปลี่ยนสีของไฟเป็นสี W หรือสี C เมื่อเราปิดไฟแล้วไฟจะแสดงเป็นสี N เพราะจะทำให้รู้สึกว่ไฟนั้นถูกปิดจริงๆ ยกตัวอย่างเช่นในกรณีที่เราเปลี่ยนไฟเป็นสีฟ้าแล้วเราแกล้งความเข้มแสงลงเหลือ 25 แต่ยังใช้แสงไฟสีฟ้าอยู่จะทำให้มันขัดกับความรู้สึก เพราะแสงธรรมชาติควรจะเป็นสีขาว

6.3 เครื่องปรับอากาศ

ต่อมาเป็นโมเดลของเครื่องปรับอากาศ โดยในส่วนของเครื่องปรับอากาศนั้นจะสามารถสั่งงานได้เพียงการเปิดและปิด และการเพิ่มลดอุณหภูมิเท่านั้น โดยลักษณะโมเดลของเครื่องปรับอากาศจะเป็นดังภาพ

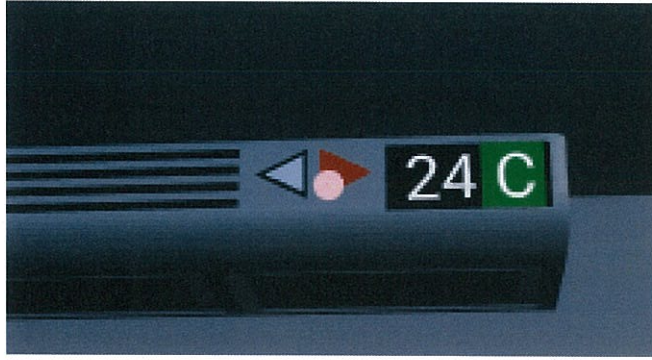


ภาพที่ 6.3.1 โมเดลเครื่องปรับอากาศ

โดยจะเห็นว่านอกจากจะมีส่วนควบคุมแล้วยังมีส่วนของการแสดงปริมาณการใช้ไฟฟ้าภายใน 7 วัน คล้ายกับของหลอดไฟและทีวี โดยตัวโมเดลเครื่องปรับอากาศจะมีแผงแสดงอุณหภูมิขนาดใหญ่ดังในภาพที่ 5.5.8 จะแสดงเลข 23 และตามด้วยหน่วยคือ องศาเซลเซียส

และเนื่องจากแผงควบคุมอุณหภูมิของเครื่องปรับอากาศตัวนี้เราออกแบบมาให้มีขนาดเล็กและไม่ใหญ่จนเกินไปเพื่อความสวยงาม แต่เมื่อเราใช้งานผ่านแว่น VR ของ Google Daydream เรากลับพบว่าความละเอียดและความแม่นยำของการแสดงผลนั้นผิดเพี้ยนไปพอสมควรเนื่องจาก A-frame ยังไม่สามารถแสดงผลในความละเอียดสูงๆ ได้ เราจึงแก้ปัญหาโดยการเมื่อมี cursor มากระทบกับแผงควบคุมเราจะมีสีไว้บ่งบอกว่าปุ่มนี้พร้อมทำงานแล้ว ตามภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 6.3.2 ภาพเมื่อแผงควบคุมเครื่องปรับอากาศพร้อมทำงาน

จากภาพเราจะเห็นได้ว่าก่อนที่เราจะนำ cursor มาวางปุ่มยังเป็นสีดำ แต่เมื่อวางแล้วปุ่มกลายเป็นสีแดงช่วยให้ผู้ใช้รู้ว่าผู้ใช้เองกำลังชี้ไปที่แผงควบคุมส่วนไหน และเมื่อเราสั่งปิดเครื่องปรับอากาศจะมีแผงแสดงว่า OFF ดังภาพ



ภาพที่ 6.3.3 ภาพโมเดลของเครื่องปรับอากาศตอนปิดการทำงาน

6.4 Teleport

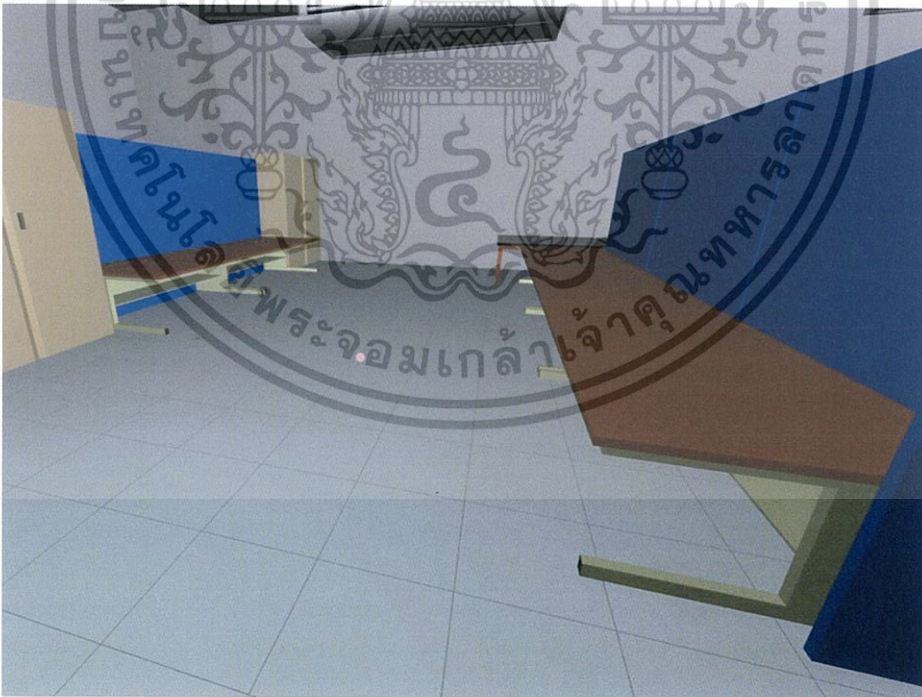
และเนื่องจากอุปกรณ์ที่เราใช้นั้นไม่สามารถติดตามการเคลื่อนที่ของผู้ใช้ได้ทำให้เราไม่สามารถเคลื่อนที่ไปได้รอบห้อง เราจึงได้สร้าง Component ที่มีลักษณะเป็นทรงกลมไว้เพื่อรับคำสั่งการเปลี่ยนมุมมองคล้ายๆ กับการ Teleport ตามภาพที่ 5.5.1 จะเห็นเป็นจุดดำๆ อยู่บนพื้นห้อง โดยการใช้งานโดยการนำ cursor ไปคลิกบนทรงกลมสีดำดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 6.4.1 ภาพก่อน Teleport

โดยเมื่อเรามีปฏิสัมพันธ์กับทรงกลมสีดำนี้แล้วจะทำให้เราเปลี่ยนมุมมองของผู้ใช้ดังภาพ



ภาพที่ 6.4.2 ภาพหลัง Teleport

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพจะเห็นได้ว่ามุมมองของผู้ใช้นั้น ได้ถูกเปลี่ยนเรียบร้อยแล้ว โดยในการสร้างจุด Teleport นั้นเราเปลี่ยนตำแหน่งของ Camera component ไปเป็นตำแหน่งที่เราต้องการดั่งภาพ

```
<Entity camera look-controls position={this.state.pos}>
```

ภาพที่ 6.4.3 ภาพโค้ดการกำหนดตำแหน่งมุมมอง

โดยวิธีการคือเราสร้าง Component ที่เป็นลูกทรงกลมสีดำไว้ที่ตำแหน่งต่างๆ และเมื่อผู้ใช้มีปฏิสัมพันธ์กับ Component นั้นๆ จะทำการเรียกฟังก์ชัน this.setState เพื่อเปลี่ยนค่า pos ไปเป็นค่าที่เราต้องการ ดั่งภาพตัวอย่าง โค้ด

```
<Entity
  events={{
    mouseenter: this.trigColorIn.bind(this),
    mouseleave: this.trigColorOut.bind(this),
    click: this.movePointB.bind(this)
  }}
>
```

ภาพที่ 6.4.4 ภาพโค้ด event ของลูกทรงกลม

ในภาพจะเห็นว่ามีการรับ events การ click โดยเมื่อถูก click จะไปทำการเรียกฟังก์ชันที่มีหน้าที่ในการเปลี่ยนตำแหน่งของมุมมองไปยังตำแหน่งอื่น

บทที่ 7

บทสรุปและข้อเสนอแนะ

7.1 บทสรุปของโครงการ

การทำงานของ VRHome ณ ปัจจุบันสามารถทำงานได้ปกติสามารถสั่งงานอุปกรณ์ทั้งสามประเภทได้ปกติ สามารถควบคุมสั่งงาน โทรทัศน์ หลอดไฟ และเครื่องปรับอากาศได้ สามารถดึงข้อมูลเพื่อมาแสดงผลใน VRHome และข้อจำกัดของ A-frame ที่ไม่สามารถทำให้เล่นในคุณภาพที่สูงมากเมื่อเทียบกับ VR ที่ทำงานด้วย PC หรือแบบเป็น Native application แต่อย่างไรก็ตาม A-frame สามารถทำให้คุณภาพใกล้เคียงกับ VR คุณภาพสูง ๆ ได้ แต่เนื่องจากเวลาในการทำโครงการที่จำกัด ทำให้ไม่สามารถทำให้ผลงาน โมเดล ต่างๆ ออกมาคุณภาพดีที่สุดได้ เพียงแต่ทำออกมาแค่พอทำให้รู้สึกว่ามันยังเป็น ตัวแทนของอุปกรณ์นั้นจริงๆ

จากโครงการนี้จะสังเกตได้ว่า อุปกรณ์บางชนิดตัวอย่างเช่น โทรทัศน์นั้นไม่เหมาะนำมาแสดงบน VR เพราะคุณลักษณะของอุปกรณ์เป็นอุปกรณ์แสดงผลอยู่แล้ว ทำให้รู้สึกว่ามันยังไม่ประโยชน์มากที่จะสั่งการทางไกล และปัญหาจากการใช้ Google Daydream หากใช้เวลานานเกินไปอาจจะทำให้มันสับสนได้ นอกจากนี้ยังทำให้เครื่องสมาร์ตโฟนร้อนขึ้นอีกด้วยทำให้เราสามารถใช้ได้เพียงเวลาที่จำกัดในการใช้งานแต่ละครั้ง

ทั้งนี้โครงการนี้เป็นเสมือนการทดลองการทำเทคโนโลยี VR มาใช้ควบคุมสิ่งของที่ใช้ใน ชีวิตประจำวัน อย่างอุปกรณ์ภายในบ้านรวมทั้งทดลองการแสดงผลของการใช้ไฟฟ้า มีความเป็นไปได้ว่าอีกไม่นานคุณภาพของ VR ที่ทำงานบนสมาร์ตโฟนจะมีคุณภาพที่สูงขึ้น อาจจะทำให้มุมมองการใช้ VR ของใครๆ หลายคนเปลี่ยนไปก็เป็นได้

7.2 ปัญหาและอุปสรรค

- 1) ในอุปกรณ์ที่ใช้ในการทดลองครั้งนี้ไม่สามารถรับการเคลื่อนที่ได้ ทำให้ลดความรู้สึกค้อย เป็นค้อยไป ซึ่งเป็นลักษณะของมนุษย์ในการเรียนรู้ในโลกแห่งความเป็นจริง
- 2) ในการสร้างความสมจริง สวยงามนั้น ต้องอาศัยแรงบันดาลใจรวมทั้งประสบการณ์ที่ยาวนานพอสมควรจึงจะออกแบบสิ่งที่ทำให้ผู้ใช้งานรู้สึกประทับใจ ทั้งนี้ผู้ที่เริ่มต้นทำเพียงคนเดียวกับด้วยเวลาที่จำกัด เลยทำออกมาได้แค่พอเห็นว่าทำงานได้เพียงเท่านั้น
- 3) ไม่สามารถรับข้อมูลจาก Web socket จาก HEMS ได้ ทำให้ต้องแก้ปัญหาโดยทำการ GET สถานะของอุปกรณ์อยู่เป็นพักๆ เพื่ออัปเดตสถานะของอุปกรณ์ในกรณีที่มีผู้ใช้งานหลายคน
- 4) มีเวลาใช้กับความเป็น HEMS น้อยทำให้ไม่เข้าใจการใช้ชีวิตประจำวันที่ต้องคอยดูการใช้พลังงาน หรือสามารถสั่งเริ่มการทำงานจากระยะไกลก่อนกลับบ้านได้
- 5) ทุกสิ่งในการทำโปรเจกนี้เป็นเรื่องใหม่สำหรับผู้จัดทำ ทำให้ต้องอาศัยระยะเวลาการเรียนรู้ที่มาก ทั้งยังต้องศึกษาเครื่องมือบางตัวที่อาจจะได้ใช้ แต่บางตัวก็ไม่ได้นำมาใช้ เพราะเวลาในการพัฒนาไม่เพียงพอ

7.3 แนวทางการพัฒนาต่อ

- 1) สามารถพัฒนาระบบสั่งงานด้วยเสียง ทำให้สามารถสั่งงานอุปกรณ์ในบ้านผ่านเสียงเราได้
- 2) พัฒนาโมเดลให้มีความสมจริงมากขึ้น ช่วยให้ผู้ใช้งานได้รับประสบการณ์ที่สมจริงมากกว่า รู้สึกว่ามันเป็นการ์ตูน
- 3) ลองพัฒนาด้วย Augmented Reality (AR) เพราะ AR นั้นไม่ต้องการอุปกรณ์ที่มากมายทำให้ง่ายต่อการใช้งานในทุกที่

บรรณานุกรม

Mozilla, **A-Frame project** Available at: <https://aframe.io>

Microsoft Developers, **Mixed Reality** Available at: https://developer.microsoft.com/en-us/windows/mixed-reality/mixed_reality

Google, **ARCore and WebARonARCore** Available at:
<https://developers.google.com/ar/discover/>

Facebook, **ReactJS** Available at: <https://reactjs.org/>

Developer blog, **Hackernoon** Available at <https://hackernoon.com/tagged/software-development>

Strategy Analytics, **Google Dominates VR Headset Shipment** available at
<https://www.strategyanalytics.com/strategy-analytics/news/strategy-analytics-press-releases/strategy-analytics-press-release/2017/02/21/strategy-analytics-google-dominates-vr-headset-shipments-samsung-takes-top-spot-for-vr-revenues#.Wtm3OohuaMo>

Peeranut Chindanonda, Thisada Sutthitham, Thitiwat Watanajaturaporn. *Hololens for Home Energy Management System* (Bachelor of Engineering in Software Engineering International College). King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้