

## บทที่ 4

### ผลการศึกษา

การพัฒนาระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้านด้วยอุปกรณ์มือถือแบบ Wi-Fi ได้มีการดำเนินการตามขั้นตอน 7 ขั้นตอนตามที่ได้มีการวางแผนไว้ โดยได้มีการเริ่มตั้งแต่การเก็บข้อมูล จนไปถึงขั้นตอนการบำรุงรักษา โดยสามารถแบ่งได้ตามขั้นตอนไว้ ดังนี้

- 4.1 ผลลัพธ์จากการกำหนดความต้องการของระบบ
- 4.2 ผลลัพธ์จากการวิเคราะห์ความต้องการของระบบ
- 4.3 ผลลัพธ์การออกแบบสถาปัตยกรรมของระบบ
- 4.4 ผลลัพธ์การออกแบบการทำงานของระบบโดยละเอียด
- 4.5 Activity Diagram
- 4.6 การออกแบบหน้าจอติดต่อกับผู้ใช้ (UI Design)
- 4.7 Code
- 4.8 ผลลัพธ์การพัฒนาระบบ
- 4.9 ผลลัพธ์การบำรุงรักษาระบบ

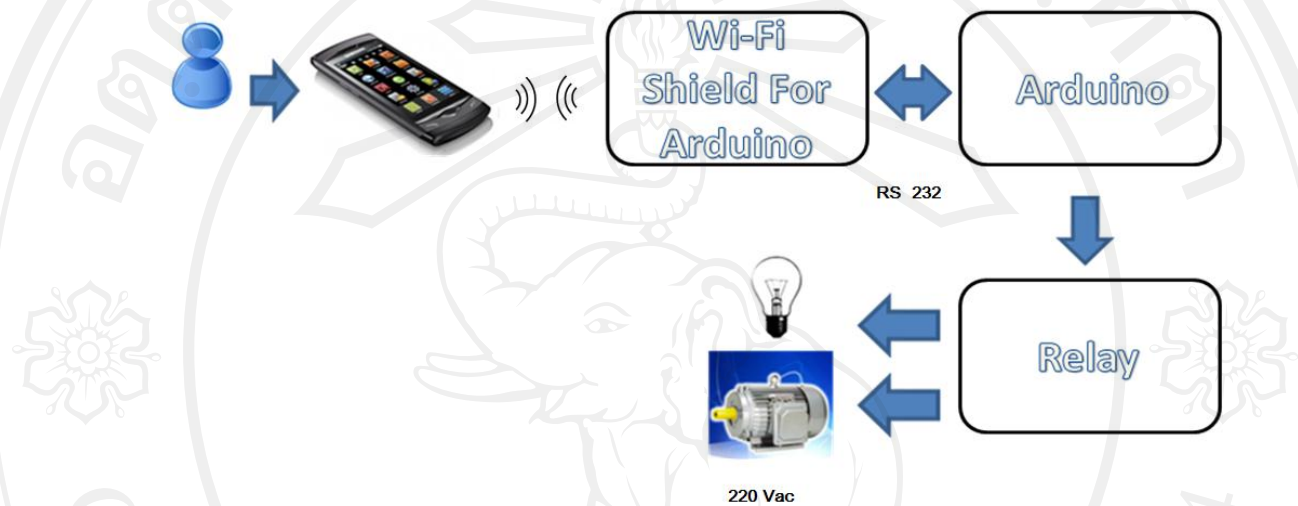
ผลลัพธ์ที่ได้จากการศึกษาจะเป็นระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้านด้วยอุปกรณ์มือถือแบบ Wi-Fi โดยเป็นไปตามแผนที่วางไว้ ทำให้ได้ระบบที่สามารถนำไปใช้งานได้จริงตามขอบเขตการศึกษา ตามวัตถุประสงค์ของการศึกษากันคว้าอิสระในครั้งนี้

#### 4.1 ผลลัพธ์จากการกำหนดความต้องการของระบบ

จากการสำรวจและการสอบถามข้อมูลจากผู้เกี่ยวข้องที่จำเป็นต่อระบบ ทำให้ผู้พัฒนาระบบได้ข้อมูลที่สำคัญในการที่จะวิเคราะห์และออกแบบระบบได้ต่อไป โดยมีผลลัพธ์ข้อมูลที่ได้มา ดังนี้

- ผู้ใช้ต้องสามารถทำการควบคุมอุปกรณ์ไฟฟ้าผ่านระบบได้
- สามารถใช้อุปกรณ์พกพามือถือในการเข้าใช้ระบบได้

- คนอื่นที่ไม่ใช่เจ้าของไม่สามารถเข้าใช้งานระบบได้
- มีระบบสำรองไฟฟ้ากรณีไฟฟ้าดับ



รูปที่ 20 การทำงานของระบบ

การทำงานของระบบผู้ใช้จะแบ่งออกเป็น เจ้าของบ้าน (User) กับ ผู้ดูแลระบบ (Admin) ต่างกันตรงที่ Admin เพิ่มผู้ใช้ที่สามารถเข้าใช้งานระบบได้ เจ้าของบ้าน สามารถเข้าใช้งานระบบได้ ไม่สามารถเพิ่มจำนวนผู้ใช้ได้ มือถือเป็นตัวที่ใช้ในการสั่งงานระบบจากเจ้าของบ้าน Wi-Fi Shield For Arduino เป็นตัวเชื่อมต่อสัญญาณการควบคุมจากมือถือไปยัง Arduino และจาก Arduino ไปยังมือถือ โดยที่การสั่งงานจากเจ้าของบ้าน สัญญาณการควบคุมจะถูกส่ง จากไมโครคอนโทรลเลอร์ ผ่าน Relay ไปเปิดปิดอุปกรณ์ไฟฟ้า

#### 4.2 ผลลัพธ์จากการวิเคราะห์ความต้องการของซอฟต์แวร์

หลังจากการเก็บข้อมูลที่สำคัญจากผู้ใช้ระบบแล้ว จากนั้นในขั้นตอนนี้จะแสดงผลลัพธ์จากการวิเคราะห์ความต้องการจากผู้ใช้ โดยจะแยกเป็น 2 ส่วนใหญ่ๆ คือ ส่วนของ Wifi Shield for Arduino และส่วนของแอปพลิเคชันบนมือถือ ดังนี้

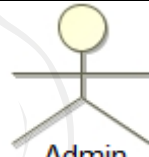
- ส่วนของ Wi-Fi Shield for Arduino
  - การทำงานของ Wi-Fi Shield
  - การทำงานของ Arduino
- ส่วนของแอปพลิเคชันบนมือถือ (Mobile Application)
  - การส่งสัญญาณไปยัง Wi-Fi Shield for Arduino

สำหรับระบบระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้านด้วยอุปกรณ์มือถือแบบ Wi-Fi นั้นจะนำความสะดวกให้กับเจ้าของบ้าน เนื่องจากในปัจจุบัน มีการใช้ Smart Phone มากยิ่งขึ้น จึงทำให้เล็งเห็นความสำคัญที่จะทำการพัฒนาซอฟต์แวร์เพื่อให้สามารถใช้ได้ในระบบ โดยสามารถเข้าได้ทางอุปกรณ์พกพาสื่อสาร เข้ามาใช้งานระบบ Wi-Fi โดยตรงโดยไม่ผ่านระบบอินเทอร์เน็ตซึ่งจะทำให้ระบบมีช่องทางในการติดต่อสื่อสารกับ อุปกรณ์ไฟฟ้าผ่านช่องทางไร้สาย ในรูปแบบของระบบ

Wi-Fi โดยสามารถที่จะเปิดประตูบ้านได้โดยไม่ต้องมีกุญแจ เปิดปิดไฟฟ้าภายในบ้านเพื่อความปลอดภัยและสะดวกรวดเร็วและไม่เสียค่าอินเทอร์เน็ต มีข้อที่ดีในเรื่องของความปลอดภัยโดยที่ไม่สามารถเข้าควบคุมอุปกรณ์ผ่านระบบอินเทอร์เน็ตได้ ถ้านำเอาไปใช้ในบ้านที่ต้องการระบบรักษาความปลอดภัยสูงก็จะมีระบบที่ระบบหนึ่ง คนที่ไม่มีตัว โปรแกรมในมือถือก็ไม่สามารถจะทำการเข้าใช้งานระบบได้จะมองเห็นแค่ส่วนของสัญญาณ Wi-Fi ที่ส่งออกมาเท่านั้นเอง อีกทั้งระบบมีการเขียนโปรแกรมในส่วนของ ไมโครคอนโทรลเลอร์ เป็นรหัสที่สามารถตั้งค่าได้มากมายจนคาดเดารหัสไม่ได้

ผู้ใช้งานระบบ (Actor) จากการวิเคราะห์ความต้องการของซอฟต์แวร์

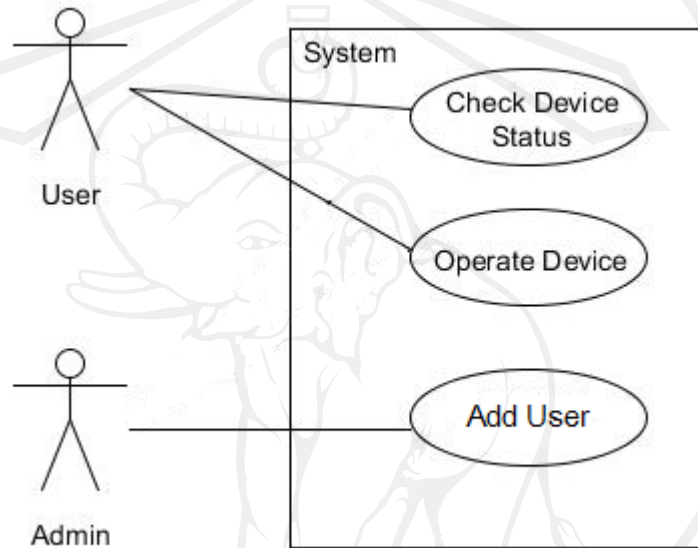
ตารางที่ 1 ผู้ใช้งานระบบ (Actor)

| ผู้ใช้งานระบบ         | สัญลักษณ์                                                                                   |
|-----------------------|---------------------------------------------------------------------------------------------|
| เจ้าของบ้าน (User)    | <br>User  |
| ผู้ดูแลทั่วไป (Admin) | <br>Admin |

การทำงานของระบบผู้ใช้งานจะแบ่งออกเป็น เจ้าของบ้าน (User) กับ ผู้ดูแลระบบ (Admin )  
 ต่างกันตรงที่ Admin เพิ่มผู้ใช้ที่สามารถเข้าใช้งานระบบได้ เจ้าของบ้าน ไปสามารถเข้าใช้งานระบบได้  
 ไม่สามารถเพิ่มจำนวนผู้ใช้ได้

แสดงการทำงานของระบบหรือยูสเคส (Use Case) จากการวิเคราะห์ความต้องการของซอฟต์แวร์

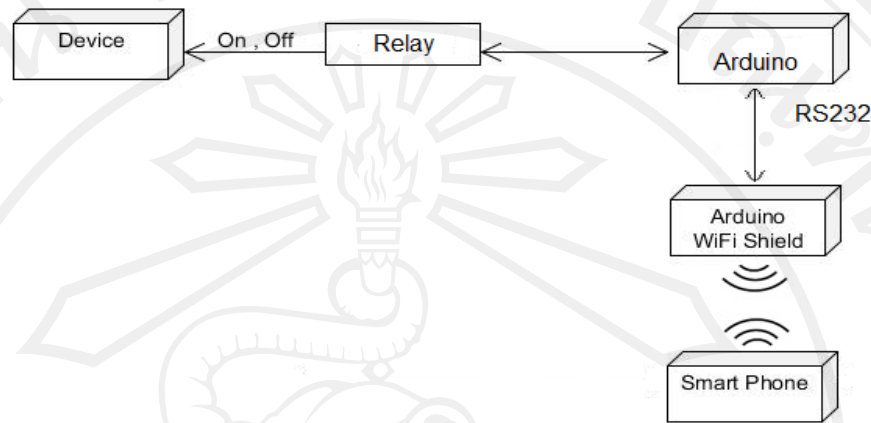
#### 4.2.1.1 ยูสเคสการทำงานโดยรวม (Overall)



รูปที่ 21 ยูสเคสการทำงานโดยรวม

User จะสามารถเช็คสถานะของระบบได้ และเข้าใช้งานระบบได้ ส่วน Admin จะสามารถเพิ่ม User ได้ก่อนใช้งานจะต้องมีการเข้าระบบ Admin ก่อนโดยใช้รหัส Admin ชื่อผู้ใช้ Admin เพื่อทำการเพิ่ม User จากนั้นระบบจะสามารถใช้งานได้จาก User เพื่อการเข้าใช้งานระบบตามที่ Admin อนุญาตให้ใช้งานโดยรับ Username และ Password จาก Admin

#### 4.3 ผลลัพธ์การออกแบบสถาปัตยกรรมของระบบ



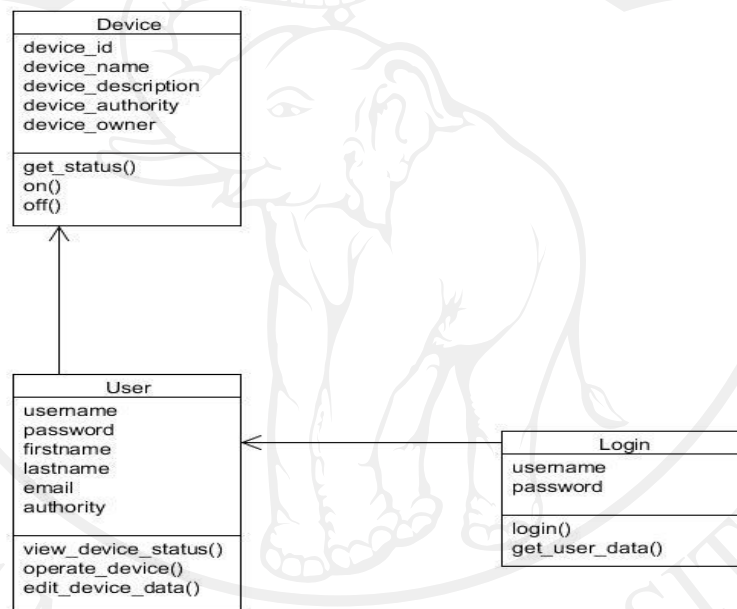
รูปที่ 22 การออกแบบสถาปัตยกรรมของระบบ

การพัฒนาระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้านด้วยอุปกรณ์มือถือแบบ Wi-Fi จะเป็นในรูปแบบ Mobile Application ซึ่งมีการติดต่อกันผ่านระบบ Wi-Fi Shield For Arduino และส่งไปควบคุมอุปกรณ์ไฟฟ้าผ่าน Relay ระบบจะทำงานเมื่อมือถือส่งสัญญาณ โดยการส่งตัวอักษร A B C ตามการกดที่หน้าจอในช่องที่ 1 จะส่งตัวอักษร A ผ่านสัญญาณไวไฟ ไปยัง Wi-Fi Shield for Arduino ก็จะรับตัวอักษร A ส่งต่อไปยัง Arduino ไมโครคอนโทรลเลอร์เขียนโปรแกรมรับ A แล้วสั่งให้ Port D22 เป็น Logic 1 หรือเป็น 5 volt Relay ก็จะส่งต่อวงจรอุปกรณ์ไฟฟ้าก็จะถูกเปิด ถ้ากด ช่องที่ 1 อีกครั้ง จะส่งตัวอักษร G ผ่านสัญญาณไวไฟ ไปยัง Wi-Fi Shield for Arduino ก็จะรับตัวอักษร G ส่งต่อไปยัง Arduino ไมโครคอนโทรลเลอร์เขียนโปรแกรมรับ G แล้วสั่งให้ Port D22 เป็น Logic 0 หรือเป็น 0 volt Relay ก็จะส่งต่อวงจรอุปกรณ์ไฟฟ้าก็จะถูกปิด ถ้ากด ช่องที่ 1 อีกครั้ง ก็จะเป็น A และเปิดอุปกรณ์ไฟฟ้า ถ้ากด ช่องที่ 1 อีกครั้ง ก็จะเป็น G และปิดอุปกรณ์ไฟฟ้า เข้าไป ซ้ำมา ถ้าเป็นปุ่มอื่นก็จะส่ง อักษรตัวอื่นและสั่งช่อง Port D23 – D30 เปิด ปิดอุปกรณ์ตามการกดปุ่มที่มีมือถือ

#### 4.4 ผลลัพธ์การออกแบบการทำงานของระบบโดยละเอียด

ผลลัพธ์จากขั้นตอนของการออกแบบการทำงานของระบบโดยละเอียด (Detail Design) ซึ่งได้ออกแบบสำหรับนักพัฒนาระบบ (Programmer) สามารถพัฒนาระบบได้ตามตรงความต้องการของผู้ใช้ โดยมีรายละเอียดดังนี้

##### 4.4.1 การออกแบบคลาสไดอะแกรม (Class Diagram)

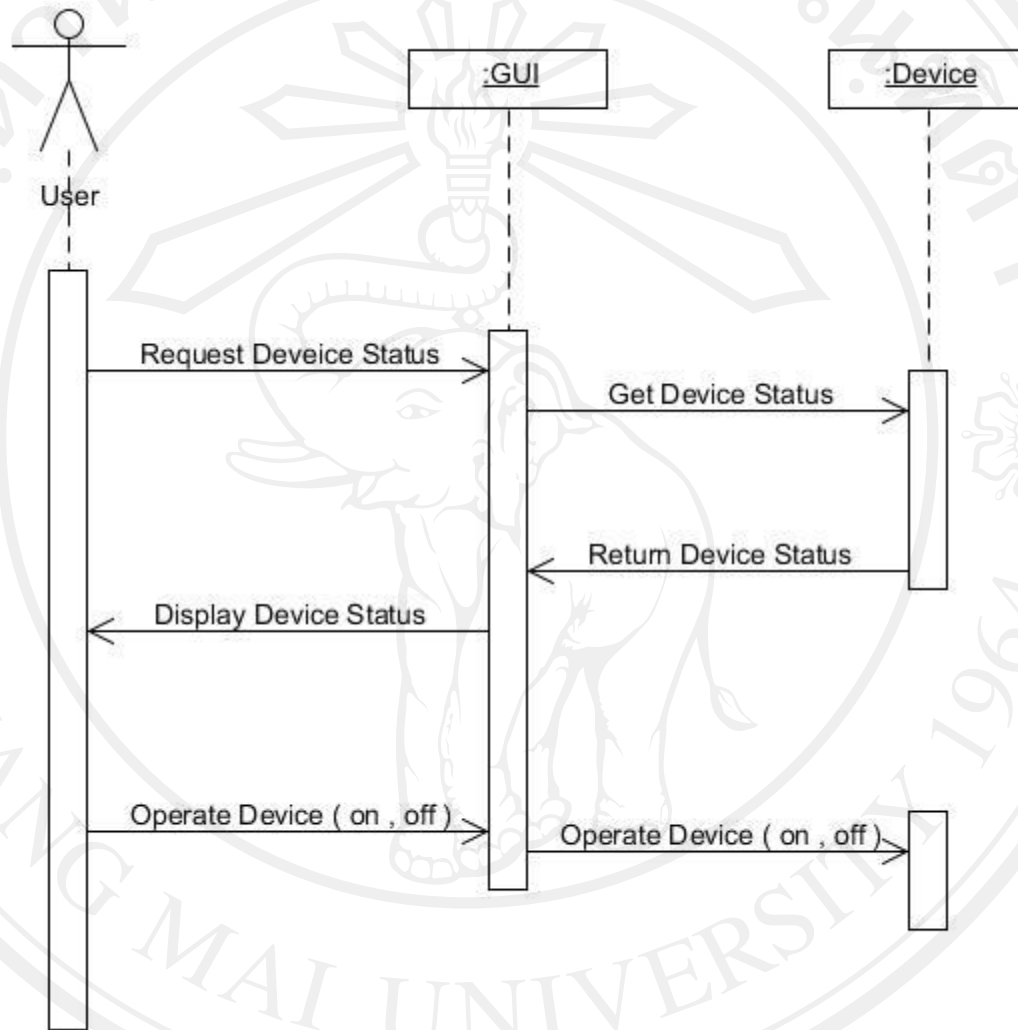


รูปที่ 23 การออกแบบคลาสไดอะแกรม

ระบบจะประกอบไปด้วยการ Login จะมีในส่วนของ Username Password ส่วนของ User จะมีส่วนของชื่อ Password เป็นหลักในการเข้าใช้งานระบบ ส่วนของอุปกรณ์ก็จะมีส่วนสำคัญ คือสถานะอุปกรณ์และการสั่งการเปิดปิดอุปกรณ์

## การออกแบบซีเควนซ์ไดอะแกรม (Sequence Diagram)

### 4.4.1.1 ซีเควนซ์ไดอะแกรมของ User

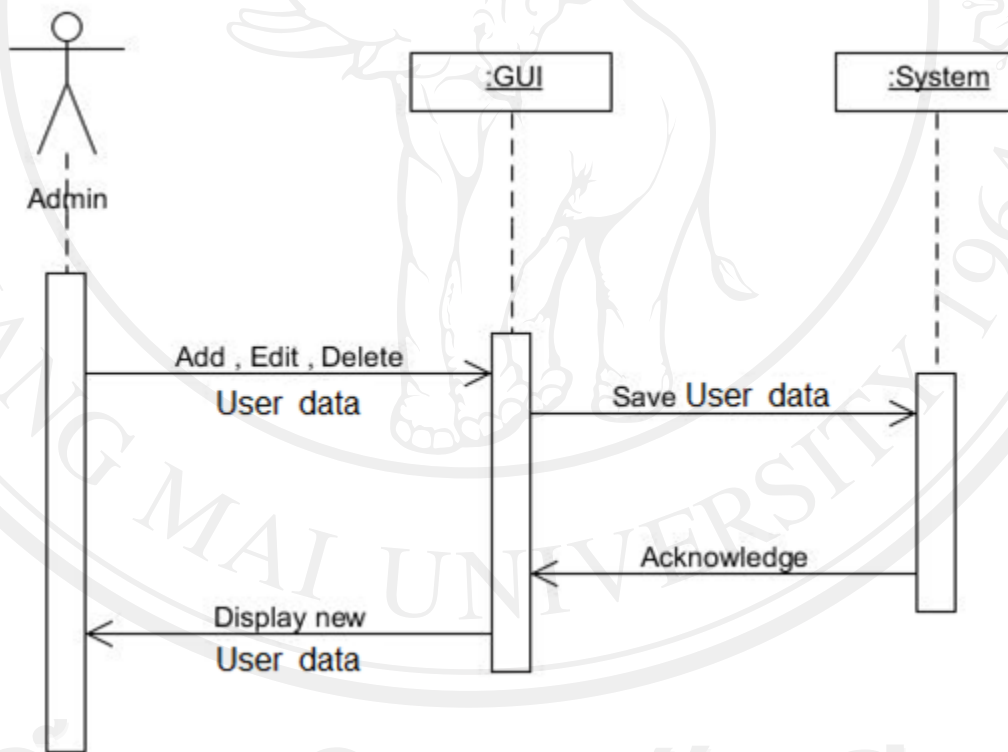


รูปที่ 24 ซีเควนซ์ไดอะแกรมของ User

การทำงานของซีเควนซ์ไดอะแกรมของ User มีส่วนหลัก ๆ จะเป็นส่วนของการสั่งการเปิดปิดอุปกรณ์ และการดูสถานะของอุปกรณ์ ผ่าน UI เมื่อ User กดหน้าจอ หน้าจอจะรับตำแหน่งการกดว่า User กดปุ่มไหน เมื่อ User กดปุ่มแรกซ้ายสุดบนสุดของหน้าจอจะเป็นการต้องการเปิดอุปกรณ์ช่องที่ 1 ซึ่ง UI จะส่งสัญญาณไปยังตัวอุปกรณ์ โดยการส่งตัวอักษร A B C ตามการกดที่หน้าจอในช่องที่ 1 จะส่งตัวอักษร A ผ่านสัญญาณไวไฟ ไปยัง Wi-Fi Shield for Arduino ตัวไวไฟ ก็จะรับตัวอักษร A ส่ง

ต่อไปยัง Arduino ไมโครคอนโทรลเลอร์เขียนโปรแกรมรับ A แล้วสั่งให้ Port D22 เป็น Logic 1 หรือเป็น 5 volt Relay ก็จะสั่งต่อวงจรอุปกรณ์ไฟฟ้าก็จะถูกเปิด ถ้ากด ช่องที่ 1 อีกครั้ง จะส่งตัวอักษร G ผ่านสัญญาณไวไฟ ไปยัง Wi-Fi Shield for Arduino ก็จะรับตัวอักษร G ส่งต่อไปยัง Arduino ไมโครคอนโทรลเลอร์เขียนโปรแกรมรับ G แล้วสั่งให้ Port D22 เป็น Logic 0 หรือเป็น 0 volt Relay ก็จะสั่งต่อวงจรอุปกรณ์ไฟฟ้าก็จะถูกปิด ถ้ากด ช่องที่ 1 อีกครั้ง ก็จะเป็น A และเปิดอุปกรณ์ไฟฟ้า ถ้ากด ช่องที่ 1 อีกครั้ง ก็จะเป็น G และปิดอุปกรณ์ไฟฟ้า ซ้ำไป ซ้ำมา ถ้าเป็นปุ่มอื่นก็จะส่ง อักษรตัวอื่นและตั้งช่อง Port D23 – D30 เปิด ปิดอุปกรณ์ตามการกดปุ่มที่มีมือถือ

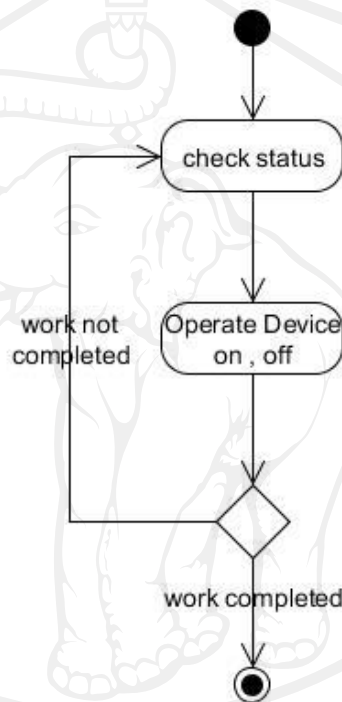
### ซีเควนซ์ไดอะแกรมการของ Admin



รูปที่ 25 ซีเควนซ์ไดอะแกรมการของ Admin

ส่วนการทำงานของซีเควนไคอะแกรมของ Admin มีส่วนหลัก ๆ จะเป็นส่วนของการเพิ่ม ลด User และดูว่ามี User อะไรบ้าง โดยที่ Admin จะทำการเพิ่ม User ให้กับเครื่องมือถือก่อนการใช้งาน มีการเก็บข้อมูล User ไว้ในเครื่องมือถือ User ก็จะสามารถใช้งานระบบได้โดยที่รับ Username และ Password ได้จาก Admin

#### 4.5 Activity Diagram



รูปที่ 26 Activity ของระบบ

ระบบจะมี Activity ของระบบคือการ ตรวจสอบสถานะการทำงานของอุปกรณ์ และ การสั่งการเปิดปิดอุปกรณ์ เมื่อทำงานเสร็จก็หมดขบวนการ การทำงาน เมื่อ User กดปุ่มแรกซ้ายสุดบนสุดของหน้าจอจะเป็นการต้องการเปิดอุปกรณ์ช่องที่ 1 ซึ่ง UI จะส่งสัญญาณไปยังตัวอุปกรณ์ โดยการส่งตัวอักษร A B C ตามการกดที่หน้าจอในช่องที่ 1 จะส่งตัวอักษร A ผ่านสัญญาณไวไฟ ไปยัง Wi-Fi Shield for Arduino ตัวไวไฟ ก็จะรับตัวอักษร A ส่งต่อไปยัง Arduino ไมโครคอนโทรเลอร์เขียนโปรแกรมรับ A แล้วสั่งให้ Port D22 เป็น Logic 1 หรือเป็น 5 volt Relay ก็จะสั่งต่อวงจรอุปกรณ์ไฟฟ้าก็จะถูกเปิด ถ้ากด ช่องที่ 1 อีกครั้ง จะส่งตัวอักษร G ผ่านสัญญาณไวไฟ ไปยัง Wi-Fi Shield for Arduino ก็จะรับตัวอักษร G ส่งต่อไปยัง Arduino ไมโครคอนโทรเลอร์เขียนโปรแกรมรับ G แล้วสั่งให้

Port D22 เป็น Logic 0 หรือเป็น 0 volt Relay ก็จะสั่งต่อวงจรอุปกรณ์ไฟฟ้าที่จะถูกปิด ถ้ากด ช่องที่ 1 อีกครั้ง ก็จะเป็น A และเปิดอุปกรณ์ไฟฟ้า ถ้ากด ช่องที่ 1 อีกครั้ง ก็จะเป็น G และปิดอุปกรณ์ไฟฟ้า เข้าไป ซ้ำมา ถ้าเป็นปุ่มอื่นก็จะส่ง อักษรตัวอื่นและสั่งช่อง Port D23 – D30 เปิด ปิดอุปกรณ์ตามการกดปุ่ม ที่มีมือถือ

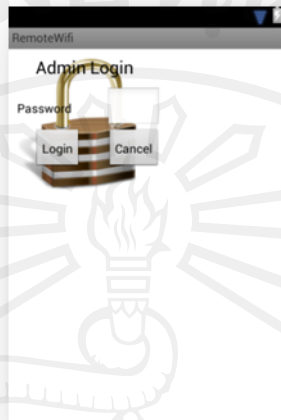
#### 4.6 การออกแบบหน้าจอติดต่อกับผู้ใช้ (UI Design)

เปิดใช้งาน Mobile Application จะเข้าสู่หน้าจอที่ 27 เป็นหน้าแรก



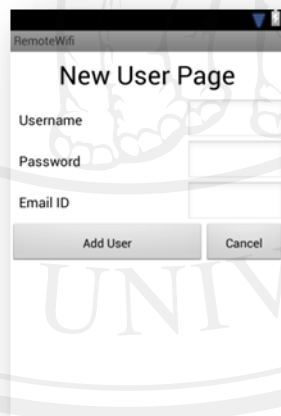
รูปที่ 27 หน้าจอแรกของ Mobile Application

รูปที่ 27 แสดงเมื่อเราเปิดใช้งาน โปรแกรมควบคุมระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้าน ด้วยอุปกรณ์มือถือแบบ Wi-Fi ก็จะเข้าสู่หน้าจอตามรูปที่ 27 หน่วงเวลา 5 วินาทีแล้วจะเข้าสู่หน้าจอ Admin Login ตามรูปที่ 28



รูปที่ 28 หน้าจอ Admin Login ของ Mobile Application

รูปที่ 28 แสดงหน้าจอ Admin Login ของ Mobile Application เพื่อให้ผู้ดูแลระบบสร้างรหัส และชื่อผู้ใช้งานว่ามีใครบ้าง เพื่ออนุญาตให้เข้าใช้งานระบบได้เพื่อรักษาความปลอดภัยโดยรหัส เข้าสร้างรหัสและชื่อผู้ใช้คือ ADMIN แล้วกดปุ่ม Login ก็จะเข้าสู่หน้าจอ New User Page



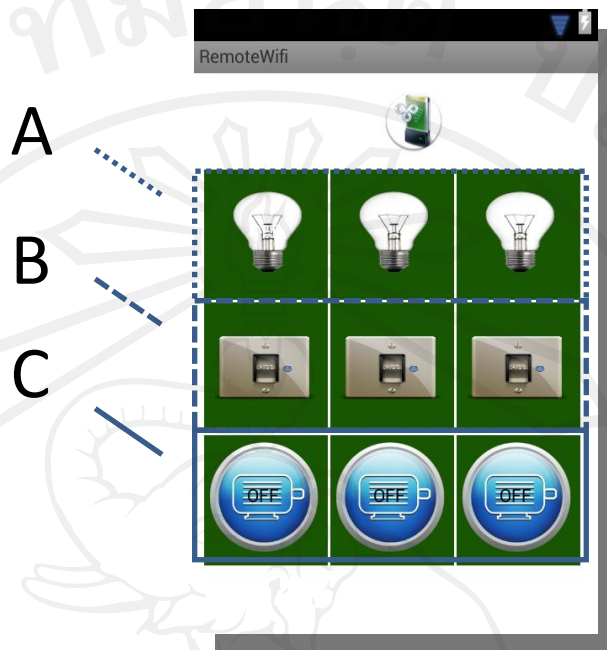
รูปที่ 29 หน้าจอ New User Page ของ Mobile Application

รูปที่ 29 แสดงหน้าจอ New User Page ของ Mobile Application เป็นหน้าจอสร้างรหัสและชื่อผู้ใช้งานว่ามีใครบ้าง เพื่ออนุญาตให้เข้าใช้งานระบบได้เพื่อความปลอดภัยในการเข้าใช้งานระบบเฉพาะผู้ที่ได้รับอนุญาตเท่านั้น



รูปที่ 30 หน้าจอ เมื่อ Login ผิดพลาดของ Mobile Application

รูปที่ 30 แสดงหน้าจอ เมื่อ Login ผิดพลาดของ Mobile Application เมื่อทำการป้อนรหัสผิดหรือว่ายังไม่ได้ป้อนรหัสก็จะแจ้งให้ผู้ใช้ทำการป้อนรหัสให้ถูกต้องเพื่อเข้าใช้งานระบบต่อไปถ้ารหัสไม่ถูกต้องก็จะไม่สามารถเข้าใช้งานระบบได้



รูปที่ 31 หน้าจอ เมื่อ Login สำเร็จจะเข้าสู่หน้าจอที่ใช้ควบคุม

รูปที่ 31 แสดงหน้าจอ เมื่อ Login สำเร็จจะเข้าสู่หน้าจอที่ใช้ควบคุม และสามารถทำการควบคุมอุปกรณ์ไฟฟ้าผ่านทาง Wi-Fi ได้โดยมีการเปิดปิดอุปกรณ์ไฟฟ้าได้ 9 ตามหน้าจอมีการเปิดปิดตามผู้ใช้งานมีการกดครั้งหนึ่งเปิด กดอีกครั้งหนึ่งปิด เป็น Toggle Switch เมื่อกดเปิดจะภาพที่หน้าจอจะเปลี่ยนเป็นรูปหลอดไฟสีเหลือง ถ้าเป็น สวิตช์ก็จะเป็นรูปสวิตช์และไฟที่สวิตช์ติดถ้าเป็น ปุ่มมอเตอร์ก็จะมีคำว่า ON เมื่อเปิด โดยที่หน้าจอนี้จะเป็นหน้าจอที่อยู่ในสถานะปิดทุกอุปกรณ์

กลุ่มของสวิตช์ A เป็นสวิตช์ ที่นำไปใช้กับอุปกรณ์ประเภทที่เป็นแสงสว่างหลอดไฟ เมื่อทำการเปิด ปิดจะมีการเปลี่ยนสีของตัว หน้าจอทำทราบดีว่าอยู่ในสถานะเปิดหรือปิด

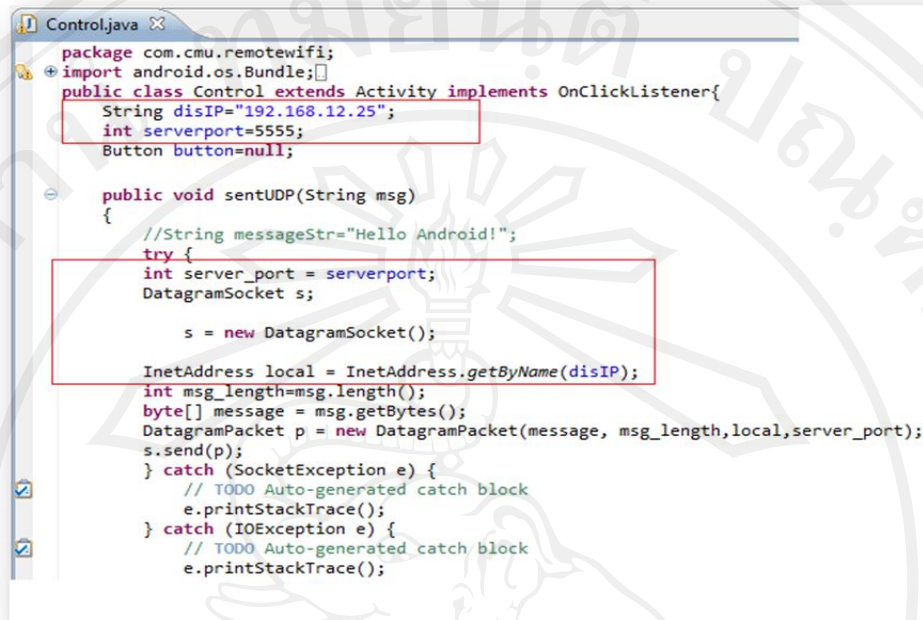
กลุ่มของสวิตช์ B เป็นสวิตช์ ที่นำไปใช้กับอุปกรณ์ประเภทที่เป็น การเปิดปิดม่าน ระบบความบันเทิงภายในบ้าน ระบบรักษาความปลอดภัย

กลุ่มของสวิทช์ C เป็นสวิทช์ ที่นำไปใช้กับอุปกรณ์ประเภทที่เป็นขดลวด หรือ มอเตอร์ เช่น การสั่งเปิดประตูหน้าบ้านโดยใช้มอเตอร์เปิดประตู การสั่งการให้เครื่องปรับอากาศ ทำงาน

โดยมีการออกแบบหน้าจอตามการทำงานของอุปกรณ์ไฟฟ้าในแบบต่าง ๆ เพื่อให้ผู้ใช้ได้รู้สึกถึงการสั่งการเปิดปิดอุปกรณ์ไฟฟ้าชนิดต่าง ๆ ในกลุ่มของการสั่งการแสงสว่างก็จะออกแบบให้มีการเปลี่ยนสีของหลอดไฟทำให้รู้สึกว่เมื่อกดสวิทช์ดังกล่าวแล้วหลอดไฟจะส่งแสงสว่างออกมา ในส่วนของสวิทช์ B ก็จะออกแบบอีกลักษณะเมื่อทำการเปิดเครื่องเสียงก็จะให้ความรู้สึกว่ต้องทำการเปิดค้างไว้ในรูปที่จะเป็นรูปของสวิทช์ เปิดแล้วมีแสงไฟกำกับ ในการเปิดระบบรักษาความปลอดภัยก็จะรู้สึกว่มีการเปิดจากหน้าจอ ส่วนสวิทช์ C ก็จะเป็นอีกรูปแบบคือการสั่งการมอเตอร์หมุน เราสั่งเปิดประตูบ้านก็จะแสดงที่หน้าจอว่ามอเตอร์ทำงานก็เป็นหน้าจอที่ให้ความรู้สึกตามอุปกรณ์ที่นำมาเชื่อมต่อกับระบบ

#### 4.7 Code

4.7.1 Code Control.java เป็นส่วน Code ที่ใช้ในการรับค่าการกดปุ่มบนมือถือแล้วทำการส่งค่าการกดปุ่มต่าง ๆ เช่นกดปุ่มด้านบนซ้ายมือ ก็จะทำการส่งตัวอักษร A หรือ B ตามสถานะของการสั่งเปิดและปิดอุปกรณ์ ผ่านทาง Port 5555 IP Address 192.168.12.25 ตามการเขียนโปรแกรมที่เขียนให้ส่งตัวอักษรแทนรหัสการกดปุ่มใด ๆ ที่หน้าจอในส่วนของ UI เพื่อส่งรหัสไปให้ Wi-Fi Shield for Arduino และส่งไปยัง Arduino เพื่อทำการเปิดปิด รีเลย์ ควบคุมอุปกรณ์ไฟฟ้าต่อไป



```

package com.cmu.remotewifi;
import android.os.Bundle;
public class Control extends Activity implements OnClickListener{
    String disIP="192.168.12.25";
    int serverport=5555;
    Button button=null;

    public void sentUDP(String msg)
    {
        //String messageStr="Hello Android!";
        try {
            int server_port = serverport;
            DatagramSocket s;

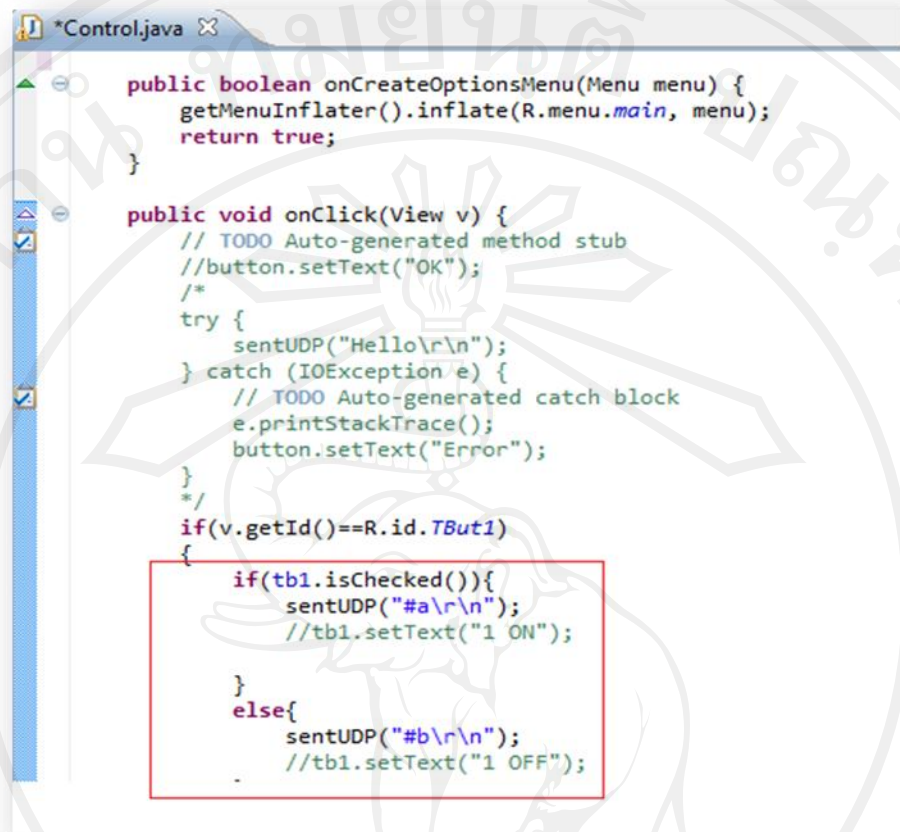
            s = new DatagramSocket();

            InetAddress local = InetAddress.getByName(disIP);
            int msg_length=msg.length();
            byte[] message = msg.getBytes();
            DatagramPacket p = new DatagramPacket(message, msg_length,local,server_port);
            s.send(p);
        } catch (SocketException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}

```

รูปที่ 32 โปรแกรมส่วนของการตั้งค่า IP Address และ Port

รูปที่ 32 แสดงโปรแกรมส่วนของการตั้งค่า IP Address และ Port ตำแหน่งที่ทำเครื่องหมายรูปเล็ก เป็นการตั้งค่าตัวแปร disIP ให้มีค่าเท่ากับ 192.168.12.25 และนำไปใช้เป็นการส่งค่าของ IP Address จริงในกรอบใหญ่ ตรงคำสั่ง `InetAddress local = InetAddress.getByName (disIP)` ในส่วนการตั้งค่าของ Port ก็เช่นกันจะทำการกำหนดค่าตัวแปรก่อนแล้วนำไปใช้จริงในโปรแกรม เป็น Server port = 5555 ในส่วนของโปรแกรมทั้งหมดจะอยู่ในส่วนของเอกสารอ้างอิงท้ายเล่ม



รูปที่ 33 โปรแกรมส่วนของการกดปุ่มแล้วส่งรหัส

รูปที่ 33 แสดงโปรแกรมส่วนของการกดปุ่มแล้วส่งรหัส ในการเขียนโปรแกรมจะมีการส่งรหัสออกไปอยู่ 2 ลักษณะคือขณะที่กดปุ่มที่ UI และเป็นสถานะ ON อีกแบบหนึ่งคือกดปุ่มที่ UI และเป็นสถานะ OFF โดยการส่งตัวอักษรที่ใช้แทนสถานะ ON เป็นตัว A ส่งตัวอักษรที่ใช้แทนสถานะ OFF เป็นตัว B ตามรูปที่ 33 ปุ่มกดอื่นก็จะส่งรหัสเป็น C,D,E,F,G,H ,I,J,K,L,,N,O,P,Q,R แทนสถานะการกดปุ่ม 9 ปุ่มที่เป็นสถานะ ON และ OFF โดยส่งผ่าน IP Address 192.168.12.25 Port 5555 ส่งออกไปแล้ว Wi-Fi Shield For Arduino ก็จะรับรหัสดังกล่าวแล้วส่งผ่าน RS232 ไปยัง Arduino และ Arduino ก็จะต้องเขียนโปรแกรมในการถอดรหัสการกดปุ่มแต่ละปุ่มให้ตรงกันก็จะสามารถที่จะควบคุมอุปกรณ์ไฟฟ้าตรงตามความต้องการผ่านมือถือได้อย่างถูกต้องตามปุ่มกดต่าง ๆ

4.7.2 Code Adminpage.java เป็นส่วน Code ที่ใช้ในการกำหนดรหัสของการเข้าใช้งานในส่วน Admin โดยมีรูปแบบการเขียนโปรแกรมตามรูปที่ 34



```

AdminPage.java
public void onClick(View v) {
    switch(v.getId()){
        case R.id.buttonLogin:
            mPassword = (EditText)findViewById(R.id.editPassword);

            String pass = mPassword.getText().toString();
            if(pass.equals("") || pass == null){
                Toast.makeText(getApplicationContext(), "Password Missing", Toast.LENGTH_SHORT).show();
            }else{
                if(pass.equals("admin")){
                    Toast.makeText(getApplicationContext(), "Retrieving Data", Toast.LENGTH_SHORT).show();
                    Intent i=new Intent(AdminPage.this,NewUserActivity.class);
                    startActivity(i);
                    finish();
                }else{
                    Toast.makeText(getApplicationContext(), "Admin Login Failed", Toast.LENGTH_SHORT).show();
                }
            }
    }
}

```

รูปที่ 34 โปรแกรมส่วนของการเข้าใช้งานในส่วน Admin

รูปที่ 34 แสดงโปรแกรมส่วนของการเข้าใช้งานในส่วน Admin มีการกำหนดรหัสตามการตั้งค่าในส่วนของการเพิ่ม Username และ Password ในหน้าจอ New User Page ถ้าป้อนรหัสถูกต้องก็จะเข้าสู่หน้าต่าง การใช้งานหน้า Activity แต่ถ้าผิดจะมีข้อความแจ้ง เป็น Password missing, Retrieving Data, Admin login Failed ตามรูปแบบของการป้อนรหัสผิดของแต่ละรูปแบบ

4.7.3 Code Arduino เป็นส่วน Code ที่ใช้ในการรับจาก Wi-Fi Shield For Arduino แล้วทำการถอดรหัสว่าการกดปุ่มจากมือถือเป็นปุ่มไหนก็ส่งให้ขา Arduino ดิจิตัล ตามการกดปุ่มที่มือถือ โดยมีการกำหนดขาที่จะทำงานสัมพันธ์กับปุ่มกดต่าง ๆ ของมือถือ



```

// initialize serial communication at 9600 bits per second:
Serial.begin(9600);
pinMode(chkLed,OUTPUT);
pinMode(load1,OUTPUT);pinMode(load2,OUTPUT);pinMode(load3,OUTPUT);
pinMode(load4,OUTPUT);pinMode(load5,OUTPUT);pinMode(load6,OUTPUT);
pinMode(load7,OUTPUT);pinMode(load8,OUTPUT);pinMode(load9,OUTPUT);

}

void loop()
{
  while(Serial.available() > 0)
  {
    while(!Serial.available());
    if(Serial.read()=='#'){
      while(!Serial.available());
      inChar = Serial.read();
    }
    chkLoad();
  }
}

void chkLoad()
{
  if(inChar=='a'){digitalWrite(load1,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='b'){digitalWrite(load1,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='c'){digitalWrite(load2,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='d'){digitalWrite(load2,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='e'){digitalWrite(load3,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='f'){digitalWrite(load3,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='g'){digitalWrite(load4,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='h'){digitalWrite(load4,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='i'){digitalWrite(load5,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='j'){digitalWrite(load5,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='k'){digitalWrite(load6,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='l'){digitalWrite(load6,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='m'){digitalWrite(load7,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='n'){digitalWrite(load7,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='o'){digitalWrite(load8,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='p'){digitalWrite(load8,LOW);Serial1.println(inChar);inChar=0;}
  if(inChar=='q'){digitalWrite(load9,HIGH);Serial1.println(inChar);inChar=0;}
  if(inChar=='r'){digitalWrite(load9,LOW);Serial1.println(inChar);inChar=0;}
}
  
```

จากรูปที่ 35 แสดงโปรแกรมส่วนของ Arduino 3 จุด ในจุด A เป็นส่วนของการตั้งค่าของการสื่อสารไว้ที่ 9600 B/S ตั้งค่าแต่ละขาที่จะใช้งานเป็น Load1, Load2, ..., Load9 เพื่อใช้เป็นขาที่รับสัญญาณการกดปุ่มที่มีมือถือ โดยการกดมือถือปุ่มที่ 1 ขา Load1 ก็จะติด ดับ ตามการกด ปุ่มที่ 1 และปุ่มอื่นก็จะเหมือนกันเรียงลำดับไปจนถึง Load9 ในจุด B เป็นส่วนของการตรวจสอบว่ามีการกดปุ่มที่มีมือถือ หรือ ไม่มีการกด ก็จะวนรอรับการ กดปุ่ม ถ้ามีการกดปุ่มที่มีมือถือก็จะมีตัวอักษรส่งมา และโปรแกรมก็จะเข้าไปตรวจสอบตัวอักษรที่มีมือถือส่งมา ในส่วนของ C ของโปรแกรมโดยมีการตรวจว่า ถ้าเป็นตัว A แสดงว่าต้องการเปิดอุปกรณ์ไฟฟ้าที่ ขา Load1 และถ้าเป็นตัว B แสดงว่าต้องการปิดอุปกรณ์ไฟฟ้าที่ ขา Load1 ในโปรแกรมส่วนที่เหลือก็จะตรวจสอบตัวอักษรที่ส่งมาจาก Wi-Fi Shield For Arduino ที่รับมาจากการกดปุ่มที่มีมือถือ ตัว C,D,E,F,G,H,I,J,K,L,N,O,P,Q,R และทำให้ขา Load2 ติด ถ้าเป็นตัว C Load2 ดับ ถ้าเป็นตัว D ขา Load3 ติด ถ้าเป็นตัว E Load3 ดับ ถ้าเป็นตัว F ขา Load4 ติด ถ้าเป็นตัว G Load4 ดับ ถ้าเป็นตัว H ขา Load5 ติด ถ้าเป็นตัว I Load5 ดับ ถ้าเป็นตัว J ขา Load6 ติด ถ้าเป็นตัว K Load6 ดับ ถ้าเป็นตัว L ขา Load7 ติด ถ้าเป็นตัว M Load7 ดับ ถ้าเป็นตัว N ขา Load8 ติด ถ้าเป็นตัว O Load8 ดับ ถ้าเป็นตัว P ขา Load9 ติด ถ้าเป็นตัว Q Load9 ดับ ถ้าเป็นตัว R เป็นการงานในส่วน

#### 4.8 ผลลัพธ์การพัฒนาระบบ

##### 4.8.1 ภาพรวมของอุปกรณ์ที่เป็นผลลัพธ์ของการพัฒนาระบบทั้งหมด



รูปที่ 36 เป็นส่วนต่าง ๆ ของการพัฒนาระบบ

รูปที่ 36 แสดงส่วนต่าง ๆ ของการพัฒนาระบบ จะประกอบด้วยส่วนของมือถือในรูปแบบของหน้าจอควบคุมตามรูป มีส่วนของ Wi-Fi Shield for Arduino ที่ใช้ในการสื่อสารระหว่างมือถือกับ Arduino และก็มี ET-EASY MEGA1280 ทั้งหมดจะใช้เป็นระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้านด้วยอุปกรณ์มือถือแบบ Wi-Fi การสั่งการจะทำการสั่งการจากมือถือส่งสัญญาณผ่าน Wi-Fi ตัว Wi-Fi Shield for Arduino เป็นตัวรับสัญญาณจากมือถือแล้วเปลี่ยนเป็นสัญญาณ RS232 ส่งต่อไปยัง ET-EASY MEGA1280 ตัว ET-EASY MEGA1280 ก็จะส่งสัญญาณไปยังบอร์ดรีเลย์เพื่อเปิดปิดอุปกรณ์ไฟฟ้าที่ต่อกับชุดควบคุม

#### 4.9 ผลลัพธ์การทดสอบระบบ

##### 4.9.1 Responsibility

ในขั้นตอนนี้ เป็นการกำหนดผู้ที่มีหน้าที่ในการทดสอบระบบ ดังนี้

ตารางที่ 2 การกำหนดผู้ที่มีหน้าที่ในการทดสอบ

| ชื่อ          | หน้าที่       |
|---------------|---------------|
| นายณพดล ภูเขา | Tester Leader |
| นายณพดล ภูเขา | Tester        |
| นายณพดล ภูเขา | Tester        |

##### 4.9.2 Test Records

(ดูภาคผนวกในหัวข้อ Testing Record)

#### 4.10 ผลลัพธ์การบำรุงรักษาระบบ

ในการทำงานบำรุงรักษาระบบควบคุมอุปกรณ์ไฟฟ้าภายในบ้านด้วยอุปกรณ์มือถือแบบไร้สายนี้ จะมีการทำงานในรูปแบบ Corrective Maintenance, Preventive Maintenance โดยได้แบ่งตามลักษณะงานดังนี้

1. Preventive Maintenance ประกอบไปด้วย

- a. เปิดการฝึกอบรมสัมมนาให้ความรู้ในการใช้งานโปรแกรมในระดับผู้ปฏิบัติงาน (user) และ ระดับผู้ดูแลระบบ (Admin) มีจุดประสงค์ให้ผู้ใช้งานมีความรู้ความเข้าใจข้อกำหนดต่างๆ ที่จำเป็นในการใช้งานผิดพลาดและป้องกันการป้อนข้อมูลที่ไม่ตรงตามวัตถุประสงค์ของโปรแกรม
- b. การทำเอกสารคู่มือการใช้งานโปรแกรมให้กับผู้ใช้
- c. การทำ Help ภายใน Application เพื่อให้ง่ายในการใช้งานโปรแกรม

2. Corrective Maintenance

- a. การรับแจ้งเหตุขัดข้องในการใช้งานผ่าน Email โดยเมื่อผู้ใช้งานพบเหตุขัดข้องจะต้องมีการคัดลอกหน้าจอและแสดงขั้นตอนในการพบความผิดพลาดและรายงานโดยการส่ง Email
- b. จัดตั้งทีมงาน support ในกรณีที่มีเหตุขัดข้องรุนแรงและเกิดเหตุจำเป็นที่ต้องเข้าไปทำงานในพื้นที่ของลูกค้า พร้อมจัดเตรียมเบอร์โทรศัพท์ติดต่อทั้งด้านลูกค้าและทางทีมงาน
- c. จัดเตรียมคู่มือแก้ไขปัญหาเบื้องต้นให้กับลูกค้าผ่านหน้า Web Site (FAQ) โดยการสรุปจากปัญหาที่พบบ่อยๆ