

บทที่ 3

วิธีการดำเนินการศึกษาและพัฒนาโปรแกรม

3.1 ศึกษาความเป็นไปได้ของการพัฒนาโปรแกรม

การศึกษาความเป็นไปได้ คือ การพิจารณาถึงความเหมาะสมและการประเมินถึงผลประโยชน์เปรียบเทียบกับค่าใช้จ่ายที่ได้ใช้ไปในการพัฒนาระบบขององค์กรและมีจุดประสงค์ที่มีการกำหนดว่าปัญหาคืออะไร และตัดสินใจว่าการพัฒนาระบบสารสนเทศหรือการแก้ไขระบบสารสนเทศเดิมมีความเป็นไปได้หรือไม่ โดยเสียค่าใช้จ่ายและเวลาน้อยที่สุด รวมถึงได้ผลลัพธ์เป็นที่น่าพอใจ เป็นการเพิ่มความเชื่อมั่นให้กับผู้ใช้ และเจ้าของระบบก่อนที่จะมีการเปลี่ยนแปลงเกิดขึ้นและหากว่ามีสิ่งเหล่านี้อยู่แล้ว ก็จะมีความเป็นไปได้ที่จะดำเนินการพัฒนาระบบงาน หรือจัดหาระบบงานมาใช้ในองค์กร เพื่อให้เกิดผลตอบแทนที่จะได้รับและประสิทธิภาพสูงสุดในการดำเนินงาน

3.2 การศึกษาความต้องการของผู้ใช้งานโปรแกรม

การศึกษาถึงความต้องการของผู้ใช้งานโปรแกรมนั้นจะก่อให้เกิดการทำงานที่มีคุณภาพและประสบความสำเร็จตามความคาดหวังจากสิ่งที่ผู้ใช้ต้องการ จากการศึกษาความต้องการของผู้ใช้งานพบว่าผู้ใช้งานส่วนใหญ่ต้องการ โปรแกรมที่ดีมีคุณภาพ สะดวกในการใช้งาน และสามารถช่วยให้การรวบรวม บันทึกและจัดเก็บข้อมูลทางด้านการใช้พลังงานเป็นไปได้อย่างสะดวกและรวดเร็ว รวมทั้งมีความถูกต้อง และสามารถที่จะวิเคราะห์ข้อมูลทางด้านพลังงานที่เก็บรวบรวมได้ออกมาให้อยู่ในรูปของรายงาน กราฟ หรือข้อมูลทางด้านสถิติด้านการใช้พลังงานได้เมื่อทราบถึงความต้องการของผู้ใช้งานโปรแกรมแล้ว จะต้องทำการออกแบบโปรแกรมให้สอดคล้องกับความต้องการของผู้ใช้โปรแกรมหากกล่าวด้วย

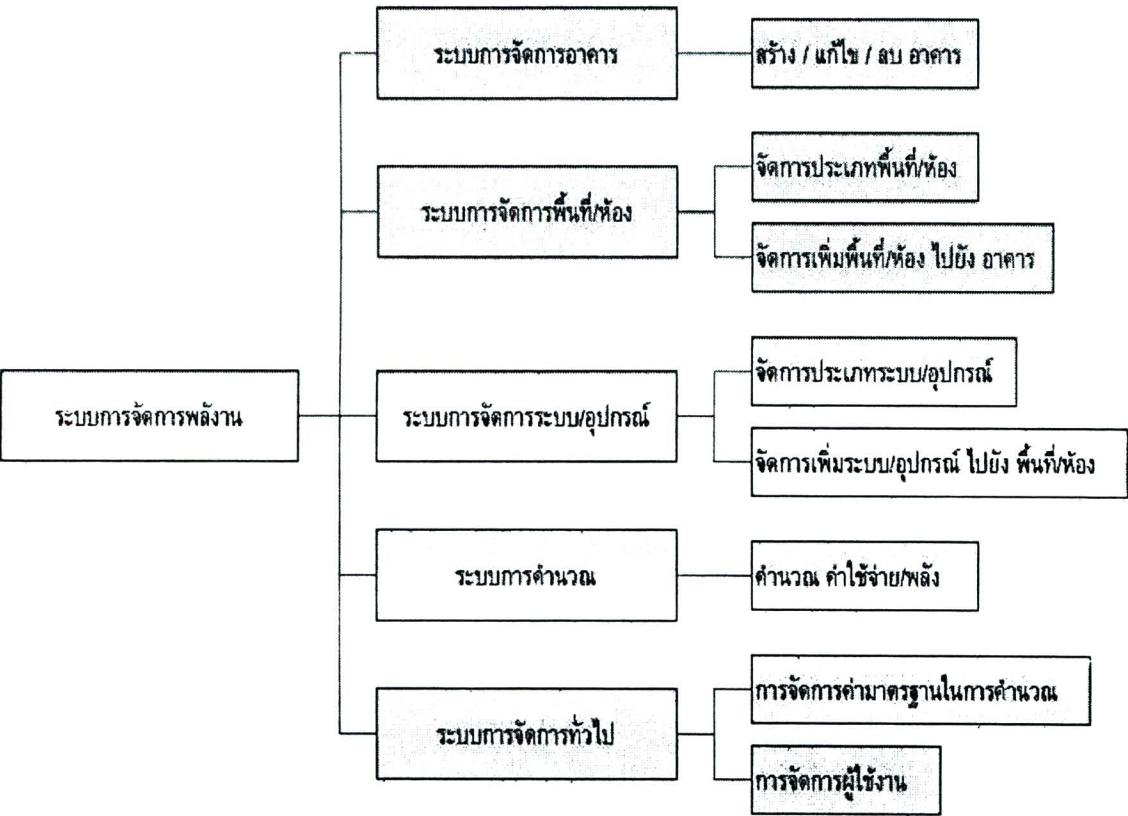
ในเบื้องต้นการศึกษาถึงความเป็นไปได้ของการพัฒนา Energy Simulation Program นั้นสามารถแยกออกเป็นประเด็นต่าง ๆ ได้ดังนี้

- ความพร้อมทางด้านการจัดหาข้อมูลตัวอย่างที่ต้องนำมาใช้ในการพัฒนาตัวโปรแกรม (จัดหาบริษัทที่จะขอข้อมูลทางด้านการใช้พลังงานไฟฟ้าต่าง ๆ มาใช้ในการพัฒนาโปรแกรม)

- ศึกษาลักษณะการเก็บและวิเคราะห์ข้อมูลด้านพลังงานในรูปแบบเดิมขององค์กร ตัวอย่างเพื่อทำความเข้าใจในระบบงานปัจจุบันและทำการระบุขอบเขตของโปรแกรม
- ศึกษาและกำหนดเครื่องมือในการพัฒนาโปรแกรมทั้งทางด้าน hardware และ software
- ศึกษาระยะเวลาที่ต้องใช้ในการพัฒนาโปรแกรม

3.3 ขอบเขตการทำงานของโปรแกรม

ในการพัฒนาโปรแกรม Energy Simulation ในรูปแบบของการทำ Energy Navigation and mapping นั้นจะมีการวางโครงสร้างของระบบภายในโปรแกรกดังรูปที่ 3.1



รูปที่ 3.1 โครงสร้างของระบบภายในโปรแกรม

โดยโปรแกรมจะแบ่งระบบย่อยออกเป็น 5 ระบบ ดังนี้

3.3.1 ระบบการจัดการอาคาร

เป็นระบบที่ออกแบบไว้เพื่อให้ผู้ใช้งานสามารถสร้าง / แก้ไข / ลบ อาคารได้

3.3.2 ระบบการจัดการพื้นที่/ห้อง

เป็นส่วนที่ผู้ใช้งานสามารถกำหนดหรือสร้างประเภทของห้อง เช่น สร้างประเภทของพื้นที่/ห้องเป็นห้องประชุม ห้องพัก เป็นต้น และยังสามารถสร้างพื้นที่/ห้องขึ้นมาใหม่ โดยพื้นที่/ห้องที่สร้างขึ้นมาจะอยู่ในอาคารตามที่ใช้กำหนดได้

3.3.3 ระบบการจัดการระบบ/อุปกรณ์

ในระบบนี้ผู้ใช้งานสามารถสร้างประเภทของระบบ/อุปกรณ์ เช่น สร้างประเภทของระบบ/อุปกรณ์เป็นเครื่องปรับอากาศ เครื่องคอมพิวเตอร์ หลอดไฟ เป็นต้น และยังสามารถสร้างระบบ/อุปกรณ์ขึ้นมาใหม่ โดยระบบ/อุปกรณ์ที่สร้างขึ้นมาจะอยู่ในพื้นที่/ห้องตามที่ใช้กำหนดได้

3.3.4 ระบบการคำนวณ

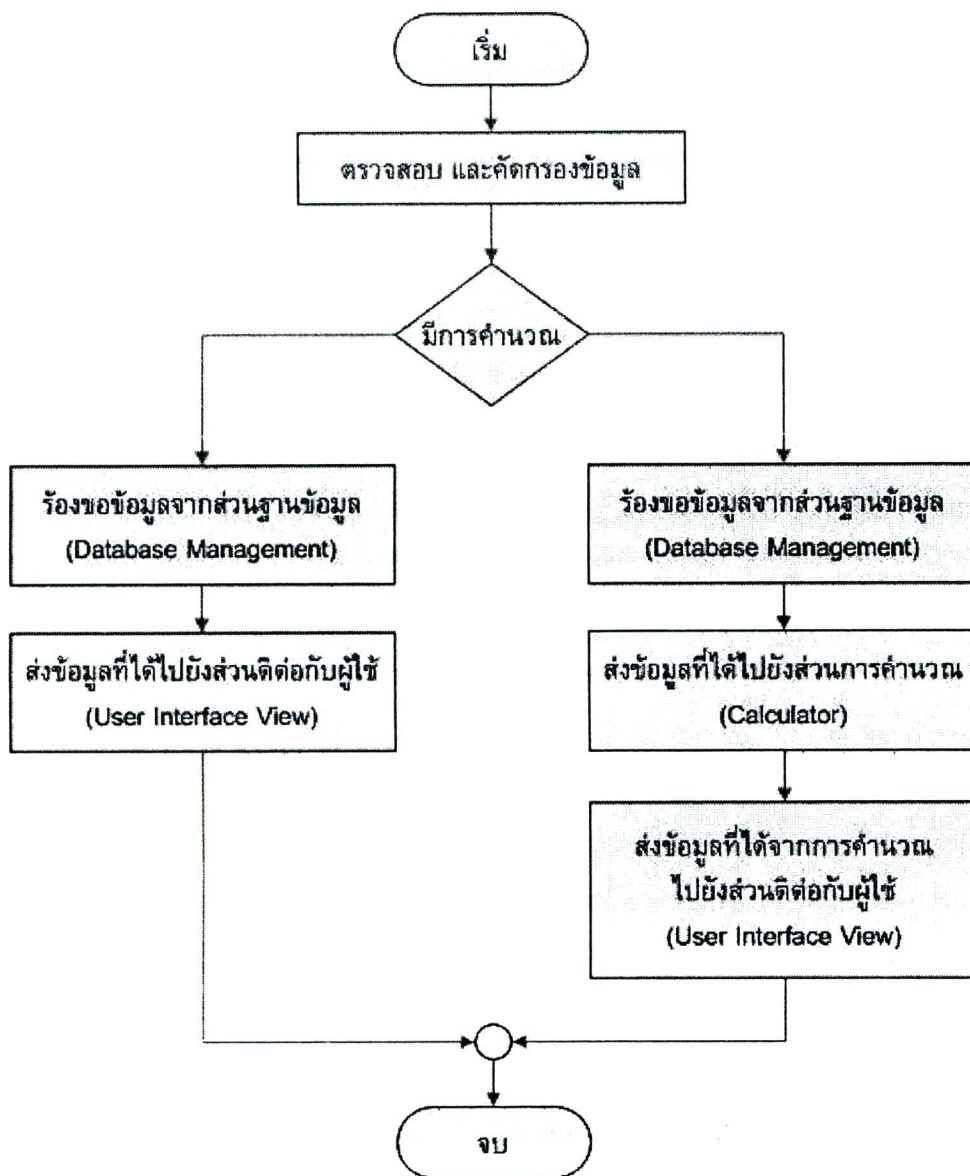
ในระบบนี้จะเป็นส่วนของภาคการคำนวณและเลียนแบบระบบโดยผู้ใช้งานสามารถทราบค่าปริมาณการใช้พลังงานรวมถึงค่าดัชนีต่างๆจากคำนวณค่าไฟฟ้า หรือดัชนีการใช้พลังงานต่อพื้นที่/ห้อง ชั้น อาคาร หรือต่อผลผลิตได้

3.3.5 ระบบการจัดการทั่วไป

ในระบบนี้เป็นระบบที่ผู้ใช้งานสามารถตั้งค่าพื้นฐานในการคำนวณ และสำหรับผู้ดูแลระบบจะสามารถจัดการกำหนดขีดจำกัดของผู้ใช้งานทั่วไปได้

3.4 ลำดับการทำงานของโปรแกรม

ในการออกแบบและพัฒนาโปรแกรมจะมีการออกแบบระบบการทำงานพื้นฐานของโปรแกรมให้ตรวจสอบข้อมูลเป็นลำดับแรก เพื่อวิเคราะห์ว่าข้อมูลดังกล่าวต้องการการคำนวณหรือไม่ ซึ่งถ้าไม่ต้องการคำนวณใดๆ โปรแกรมจะเข้าถึงฐานข้อมูล และแสดงผลให้กับผู้ใช้ โดยมีลำดับการทำงานดังรูปที่ 3.2



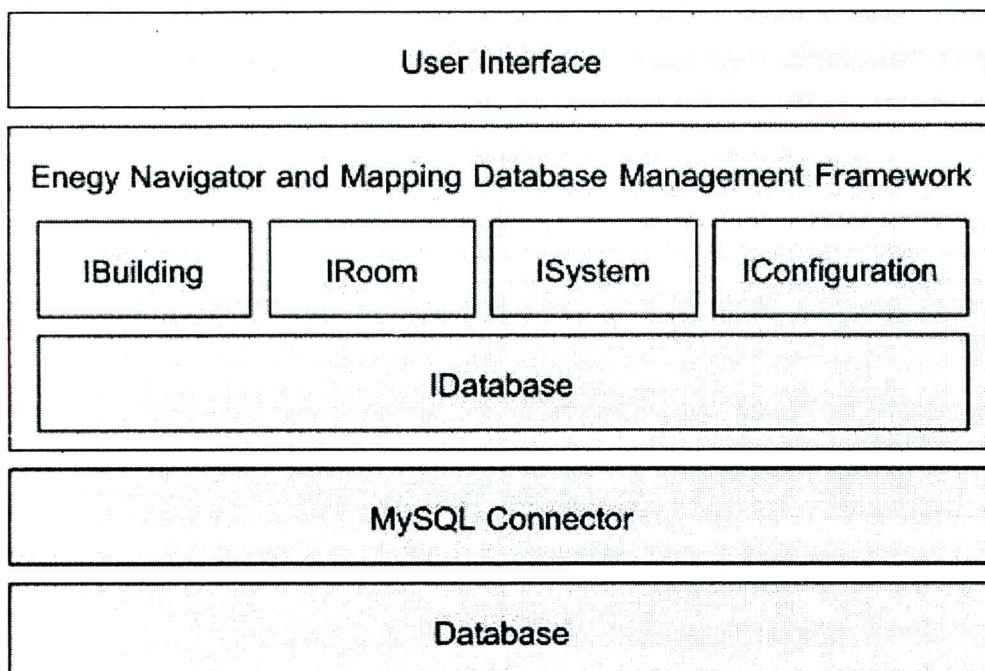
รูปที่ 3.2 ลำดับการทำงานของโปรแกรม

อย่างไรก็ตามเมื่อพิจารณาจากลำดับการทำงานของโปรแกรมจะเห็นว่าเมื่อมีการคำนวณใดๆเกิดขึ้น โปรแกรมจะทำการดึงข้อมูลที่จำเป็นจากฐานข้อมูล que ผู้ใช้ได้ป้อนไว้ในลำดับแรกเพื่อนำมาคำนวณ และแสดงผลให้กับผู้ใช้ต่อไป

3.5 การออกแบบโปรแกรม

สถาปัตยกรรมของโปรแกรมจะถูกแบ่งเป็น 4 ชั้นดังรูป 3.3 โดยมีรายละเอียดดังนี้

- 3.5.1 Database เป็นลำดับชั้นที่อยู่ล่างสุดเป็นส่วนของฐานข้อมูลซึ่งเก็บข้อมูลต่างๆ เช่น ข้อมูลของอาคาร ห้อง/พื้นที่ อุปกรณ์/ระบบ และการตั้งค่าต่างๆ เป็นต้น โดยในโปรแกรมจะใช้ MySQL
 - MySQL Connector เป็นลำดับชั้นสำหรับการเชื่อมต่อฐานข้อมูลหลักของโปรแกรม
 - Energy Navigation and Mapping Database Management Framework
- 3.5.2 Database เป็นลำดับชั้นย่อยที่อยู่ใน Energy Navigator and Mapping Database Management Framework ใช้ในการเชื่อมต่อกับฐานข้อมูล ซึ่งเป็นส่วนที่พัฒนาขึ้นมาเพื่อง่ายในการเข้าถึงข้อมูลในฐานข้อมูล เช่น การเพิ่มข้อมูล (Insert) การแก้ไขข้อมูล (Update) การลบข้อมูล (Delete) เป็นต้น ซึ่งคลาสในส่วนนี้จะมีฟังก์ชันสำหรับการเข้าถึงฐานข้อมูลพื้นฐานดังที่กล่าวมาข้างต้น เพื่อง่ายต่อการนำมาใช้ในลำดับชั้นที่สูงกว่า
- 3.5.3 IBuilding, IRoom, ISystem และ IConfiguration เป็นส่วนการเชื่อมกับฐานข้อมูล โดยใช้ IDatabase ในการเข้าถึงฐานข้อมูล ซึ่งแต่ละกลุ่มคลาสจะจัดการข้อมูล เช่น เพิ่มข้อมูล ลบข้อมูล แก้ไขข้อมูล เฉพาะเจาะจงแต่ละส่วน ดังนี้ IBuilding จะจัดการข้อมูลของอาคาร IRoom จะจัดการข้อมูลของห้อง/พื้นที่ และ ชนิดของห้อง/พื้นที่ ISystem จะจัดการข้อมูลของอุปกรณ์/ระบบ และชนิดของอุปกรณ์/ระบบ IConfiguration จะจัดการข้อมูลของการตั้งค่าทั่วไป รวมถึงระบบผู้ใช้งาน
- 3.5.4 User Interface เป็นส่วนของการแสดงผลในรูปแบบของ Graphics User Interface (GUI) โดยจะเชื่อมต่อกับส่วนของ Energy Navigator and Mapping Database Management Framework เพื่อร้องขอข้อมูลจากฐานข้อมูลเพื่อนำมาแสดงผลให้กับผู้ใช้งาน อีกทั้งยังตอบสนองต่อเหตุการณ์ต่างๆจากผู้ใช้งาน

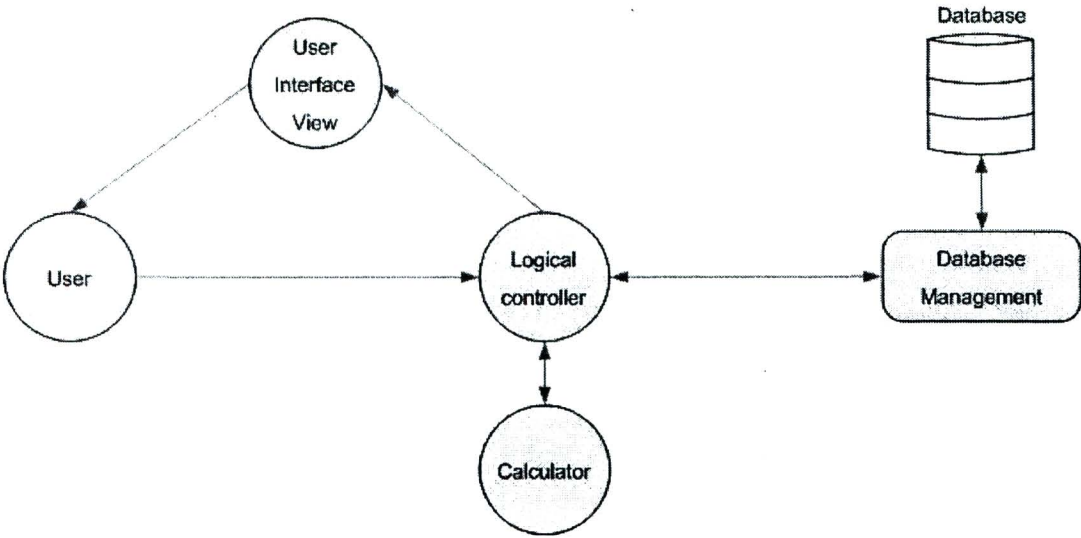


รูปที่ 3.3 สถาปัตยกรรมของโปรแกรม

3.6 โครงสร้างการทำงานของโปรแกรม

การทำงานของโปรแกรมจะแบ่งเป็นกลุ่มคลาส ซึ่งแต่ละกลุ่มจะทำงานร่วมกัน โดยจะแบ่งเป็น 4 กลุ่ม(รูปที่ 3.4) ดังนี้

- Logical Controller เป็นกลุ่มคลาสในการควบคุม และตัดสินใจการทำงานหลักของโปรแกรม กลุ่มคลาสนี้จะเชื่อมต่อกับส่วนอื่นๆทุกส่วน อีกทั้งยังตอบสนองการต่อกิจกรรมหรือเหตุการณ์จากผู้ใช้ด้วย
- User Interface View เป็นกลุ่มคลาสสำหรับการแสดงผลให้กับผู้ใช้งาน โดยในกลุ่มคลาสนี้จะประกอบด้วยหน้าต่างแสดงผลโปรแกรมในรูปแบบ Graphic User Interface (GUI) โดยข้อมูลที่นำมาแสดงผลจะต้องรับมาจากส่วนของ Logical Controller
- Database Management เป็นกลุ่มคลาสสำหรับการเชื่อมต่อฐานข้อมูล ซึ่งจะช่วยลดความยุ่งยากในการเข้าถึงฐานข้อมูลให้ง่ายขึ้นด้วยการแบ่งเป็นคลาสย่อยๆสำหรับเข้าถึงข้อมูลแต่ละชนิด ในกลุ่มคลาสนี้จะเชื่อมต่อไปยังส่วนของ Logical Controller โดยตรง
- Calculator เป็นกลุ่มคลาสสำหรับการคำนวณในโปรแกรม



รูปที่ 3.4 State diagram แสดงการทำงานทำงานของโปรแกรม

3.7 การออกแบบฐานข้อมูล

ในด้านของฐานข้อมูลจะถูกแบ่งออกเป็นฐานหลัก 4 ส่วนดังนี้
ข้อมูลส่วนที่หนึ่ง พื้นฐานภาพรวมของระบบซึ่งจะเป็นส่วนที่เก็บข้อมูลพื้นฐานภาพรวม
ใหญ่ทั้งหมดของ Global Table Energy Map ของอาคารหรือกลุ่มอาคารนั้นๆ โดยมีรายละเอียดดัง
ตารางที่ 3.1

ตาราง 3.1 Building เก็บข้อมูลของอาคาร

Field name	Data type	Length	Primary key	Foreign key	Detail
ID	Integer	11	Yes	-	Auto-increment
Building_type ID	Integer	11	-	-	-
Building_name	String	255	-	-	-
Floor_number	Integer	11	-	-	-
Detail	String	255	-	-	-

ข้อมูลส่วนที่สองเป็นข้อมูลส่วนรายละเอียดซึ่งจะมีรายละเอียดในอาคารย่อยแต่ละอาคารตาม Local Energy Map ซึ่งจะถูกกำหนดและแบ่งส่วนรายละเอียดปลีกย่อยที่ลึกลงไปในแต่ละชั้นของอาคารนั้นๆ โดยมีรายละเอียดของข้อมูลดังตารางที่ 3.2

ตาราง 3.2 Floor เก็บข้อมูลของห้อง/พื้นที่

Field name	Data type	Length	Primary key	Foreign key	Detail
ID	Integer	11	Yes	-	Auto-increment
ID_building	Integer	11	-	Yes	building.id
ID room	Integer	11	-	Yes	room_type.room_type_id
Room name	String	100	-	-	-
Floor	Integer	11	-	-	-
Quantity	Float	-	-	-	-
Unit	String	20	-	-	-
Area	Float	-	-	-	-

ข้อมูลส่วนที่สามเป็นข้อมูลส่วนรายละเอียดของอุปกรณ์ในห้องซึ่งจะมีรายละเอียดในอาคารย่อยแต่ละอาคาร ตาม Zone Energy Map ซึ่งจะถูกกำหนดและแบ่งส่วนรายละเอียดปลีกย่อยที่ลึกลงไปในแต่ละชั้นของอาคารนั้นๆ โดยมีรายละเอียดของข้อมูลดังตารางที่ 3.3

ตาราง 3.3 Room Type เก็บข้อมูลประเภทของห้อง/พื้นที่

Field name	Data type	Length	Primary key	Foreign key	Detail
Room Type ID	Integer	11	Yes	-	Auto-increment
Room Type name	String	255	-	-	-
Detail	String	255	-	-	-

ข้อมูลส่วนที่สี่เป็นข้อมูลส่วนรายละเอียดการใช้พลังงานของอุปกรณ์ที่อยู่ในห้องซึ่งจะมีรายละเอียดการใช้พลังงานไฟฟ้าในอุปกรณ์ต่างๆที่ติดตั้งในทุกห้อง ตาม Sub Zone Energy Map ซึ่งจะถูกกำหนดและแบ่งส่วนรายละเอียดปลีกย่อยที่ลึกลงไปในแต่ละชั้นของอาคารนั้นๆ โดยมีรายละเอียดของข้อมูลดังตารางที่ 3.4

ตาราง 3.4 Setting เก็บข้อมูลการตั้งของการใช้พลังงานของอุปกรณ์ในส่วนย่อย

Field name	Data type	Length	Primary key	Foreign key	Detail
Electricity charge	Float	-	-	-	

อย่างไรก็ตามในส่วนของรายละเอียดของอุปกรณ์ที่เป็นเครื่องจักรในแต่ละตัวนั้นเพื่อให้สามารถคำนวณและเลียนแบบระบบในการทำงานเพื่อให้สามารถได้ข้อมูลที่ถูกต้องและแม่นยำในทุกส่วนจึงต้องมีการเก็บรายละเอียดของปลีกย่อยของอุปกรณ์ดังตารางที่ 3.5

ตาราง 3.5 System เก็บข้อมูลการตั้งค่ารายละเอียดของอุปกรณ์ในส่วนย่อย

Field name	Data type	Length	Primary key	Foreign key	Detail
ID	Integer	11	Yes	-	Auto-increment
ID building	Integer	11	-	Yes	building.id
ID room	Integer	11	-	Yes	room.id
ID system	Integer	11	-	Yes	system_type. system_type_id
System_sn	String	30	-	-	Serial number
System_name	String	255	-	-	-
Suantity	Integer	11	-	-	-
Unit	String	50	-	-	-
Power	float	-	-	-	-
Year_install	Integer	11	-	-	-
Life time	Integer	11	-	-	-

ในการคำนวณค่าการใช้พลังงานของอุปกรณ์ให้มีความถูกต้องแม่นยำค่า Factor ตัวคูณของการทำงาน (Used factor) ค่าสัดส่วนการทำงานของเครื่อง (Work Ratio หรือ Duty Cycle) หรือตัวแปรในด้านของประสิทธิภาพของอุปกรณ์ เป็นต้น ดังตารางที่ 3.6

ตาราง 3.6 System_type เก็บข้อมูลการตั้งค่าต่างๆของอุปกรณ์

Field name	Data type	Length	Primary key	Foreign key	Detail
System_type_ID	Integer	11	Yes	-	Auto-increment
System_type_name	String	255	-	-	-
Detail	String	255	-	-	-
Factor	Float	-	-	-	-

ในด้านของการเข้าถึงฐานข้อมูลและการระบุผู้ใช้งานรวมถึงระบบด้านความปลอดภัยของข้อมูลซึ่งเป็นอีกปัจจัยที่สำคัญอย่างยิ่งต่อการทำแผนที่และระบบนำร่องด้านพลังงาน เนื่องจากในการใช้งานโปรแกรมต้องมีการป้อนข้อมูลที่เป็นข้อมูลทั้งหมดของอาคารหรือสถานประกอบการนั้น โดยเฉพาะอย่างยิ่งในโรงงานอุตสาหกรรมที่ข้อมูลต่างๆล้วนเป็นข้อมูลปกปิดดังนั้นการเข้าถึงข้อมูลของผู้ใช้โปรแกรมนี้จึงต้องมีระดับความปลอดภัยที่สูงพอโดยการเก็บข้อมูลในส่วนนี้แสดงในตารางที่ 3.7

ตาราง 3.7 User_info เก็บข้อมูลการตั้งค่าต่างๆของอุปกรณ์

Field name	Data type	Length	Primary key	Foreign key	Detail
user_id	Integer	11	Yes	-	Auto-increment
user_name	String	11	-	-	-
user_firstname	String	50	-	-	-
user_lastname	String	50	-	-	-
sex	Integer	11	-	-	-
citizen_id	String	11	-	-	-

ตาราง 3.7 User_info เก็บข้อมูลการตั้งค่าต่างๆของอุปกรณ์ (ต่อ)

Field name	Data type	Length	Primary key	Foreign key	Detail
address	String	255	-	-	-
telephone	String	10	-	-	-
Status	Integer	11	-	-	-