

## บทที่ 3

### ขั้นตอนการดำเนินงาน

สำหรับบทนี้จะกล่าวถึงขั้นตอนการดำเนินงานของโครงการปริญญาโทซึ่งจะทำให้ทราบถึงการวิเคราะห์และออกแบบระบบโดยละเอียดว่ามีแนวทางในการทำงานหรือมีขั้นตอนในการทำงานของระบบอย่างไรบ้าง โดยขั้นตอนในการดำเนินงานมีรายละเอียดดังนี้

#### 3.1 กรอบการดำเนินงาน

#### 3.2 แผนภาพการไหลเวียนของข้อมูล (Data Flow Diagram)

3.2.1 Context Diagram

3.2.2 Data Flow Diagram Level 1

3.2.3 Data Flow Diagram Level 2

#### 3.3 พจนานุกรมข้อมูล (Data Dictionary)

3.3.1 External Entity Description

3.3.2 Data Flow and Data Structure Description

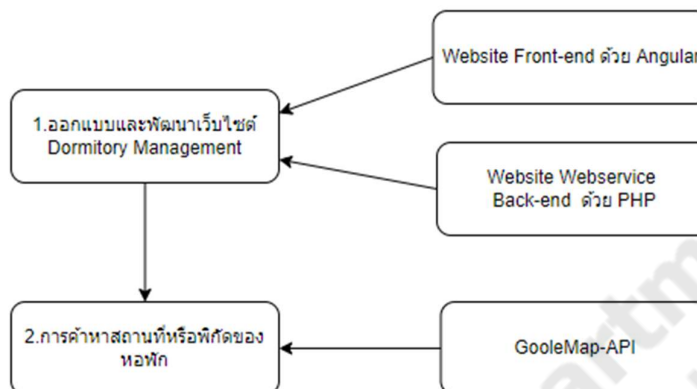
3.3.3 Data Store Description and Data Structure

3.3.4 Process Description

#### 3.4 แผนภาพแสดงความสัมพันธ์ของข้อมูล (Er-Diagram)

#### 3.5 การออกแบบฐานข้อมูล (Database Design)

### 3.1 กรอบการดำเนินงาน



ภาพประกอบที่ 3.1 กรอบการดำเนินงาน

#### คำอธิบาย

#### 1. ออกแบบและพัฒนาเว็บไซต์ Dormitory Management โดยมีเครื่องมือที่ใช้ดังนี้

1. Visual Studio Code หรือ VSCode เป็นโปรแกรม Code Editor ที่ใช้ในการแก้ไขและปรับแต่งโค้ด จากค่ายไมโครซอฟท์ มีการพัฒนาออกมาในรูปแบบของ จึงสามารถนำมาใช้งานได้แบบฟรีๆ ที่ต้องการความเป็นมืออาชีพ ซึ่ง Visual Studio Code นั้น เหมาะสำหรับนักพัฒนาโปรแกรมที่ต้องการใช้งานข้ามแพลตฟอร์ม รองรับการใช้งานทั้งบน Windows, macOS และ Linux สนับสนุนทั้งภาษา JavaScript, TypeScript และ Node.js สามารถเชื่อมต่อกับ Git ได้ นำมาใช้งานได้ง่ายไม่ซับซ้อน มีเครื่องมือส่วนขยายต่าง ๆ ให้เลือกใช้อย่างมาก

2. AngularJS เป็น JavaScript Framework ที่พัฒนาโดย Google เพื่อลดการใช้งานของข้อมูลในเครื่องแม่ข่าย (Server) ใช้ในการพัฒนาเว็บแอปพลิเคชันแบบ single-page application และยังสามารถรองรับการใช้งานกับภาษาทางด้านโปรแกรมอื่นๆได้หลากหลาย โดยการสร้าง Single-Page Application โดย AngularJS มีส่วนประกอบอยู่ 3 ส่วน ได้แก่ HTML CSS และ Javascript

#### 2. การเชื่อมต่อกับ Web service API เพื่อใช้การติดต่อกับฐานข้อมูลจะต้องมีการใช้เครื่องมือดังนี้

1. PHP หรือที่ย่อมาจาก “PHP Hypertext Preprocessor” คือภาษาคอมพิวเตอร์ประเภท scripting language ที่ใช้ในการเขียนเพื่อติดต่อกับฐานข้อมูลเพื่อทำการ query ฐานข้อมูลPHP จะถูกนำขึ้น web hosting เพื่อการรับส่งข้อมูลผ่านแอปพลิเคชัน

2.phpMyAdmin เป็นส่วนต่อประสานที่สร้างโดยภาษา PHP ซึ่งใช้จัดการฐานข้อมูล MySQL ผ่านเว็บเบราว์เซอร์ โดยสามารถที่จะทำการสร้างฐานข้อมูลใหม่ หรือทำการสร้าง TABLE ใหม่ๆ และยัง มี function ที่ใช้สำหรับการทดสอบการ query ข้อมูลด้วยภาษา SQL

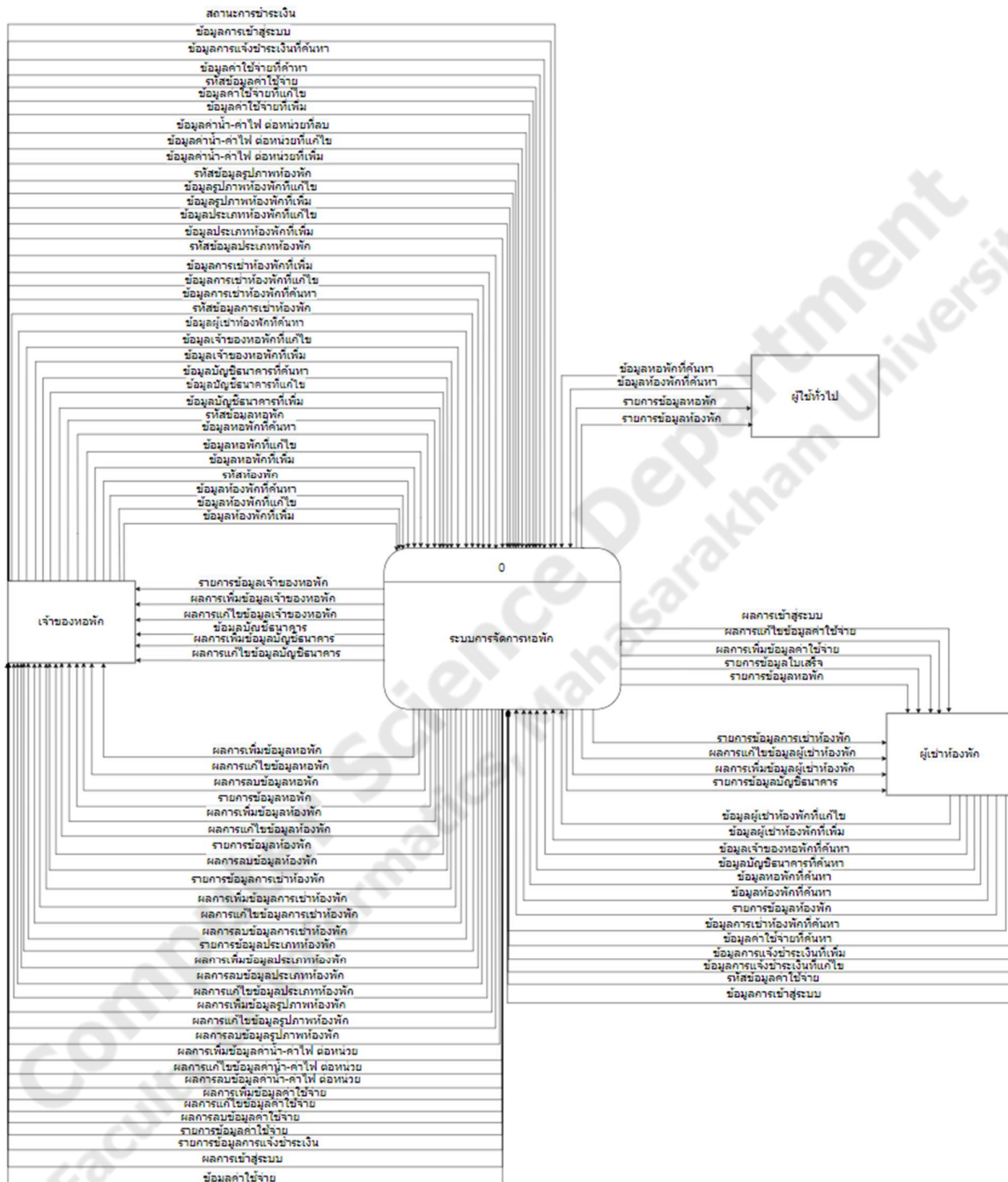
3. การเชื่อมต่อกับ API ภายนอก มีดังนี้

google Maps API เป็นชุด API ของ Google สำหรับพัฒนา web application และ mobile application (Android, iOS)ไว้สำหรับเรียกใช้แผนที่และชุด service ต่าง ๆ ของ Google เพื่อพัฒนา Application ได้เหมือนกับที่ Google โดยแผนที่ยัง features ต่าง ๆ มากมายให้เรียกใช้

### 3.2 แผนภาพการไหลเวียนของข้อมูล

แผนภาพกระแสข้อมูล(DFD) เป็นเครื่องมือที่ใช้กันอย่างแพร่หลายในการเขียนแบบระบบใหม่ ในการเขียนแผนภาพจำลองการทำงานของกระบวนการ (Process) ต่างๆในระบบ

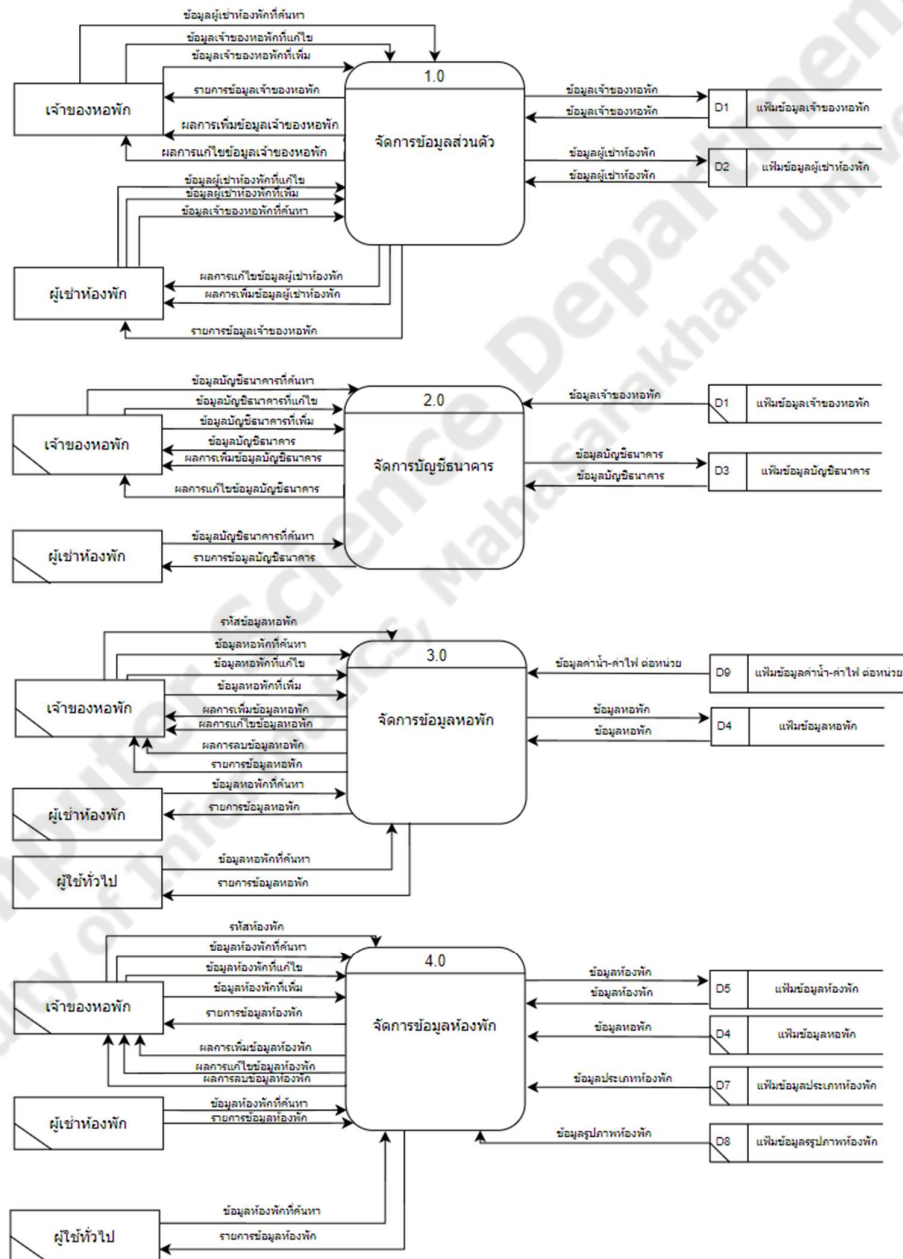
3.2.1 Context Diagram คือ แผนภาพกระแสข้อมูลระดับบนสุดที่แสดง ภาพรวมการทำงาน ของระบบที่มีความสัมพันธ์กับสภาพแวดล้อมภายนอกระบบ



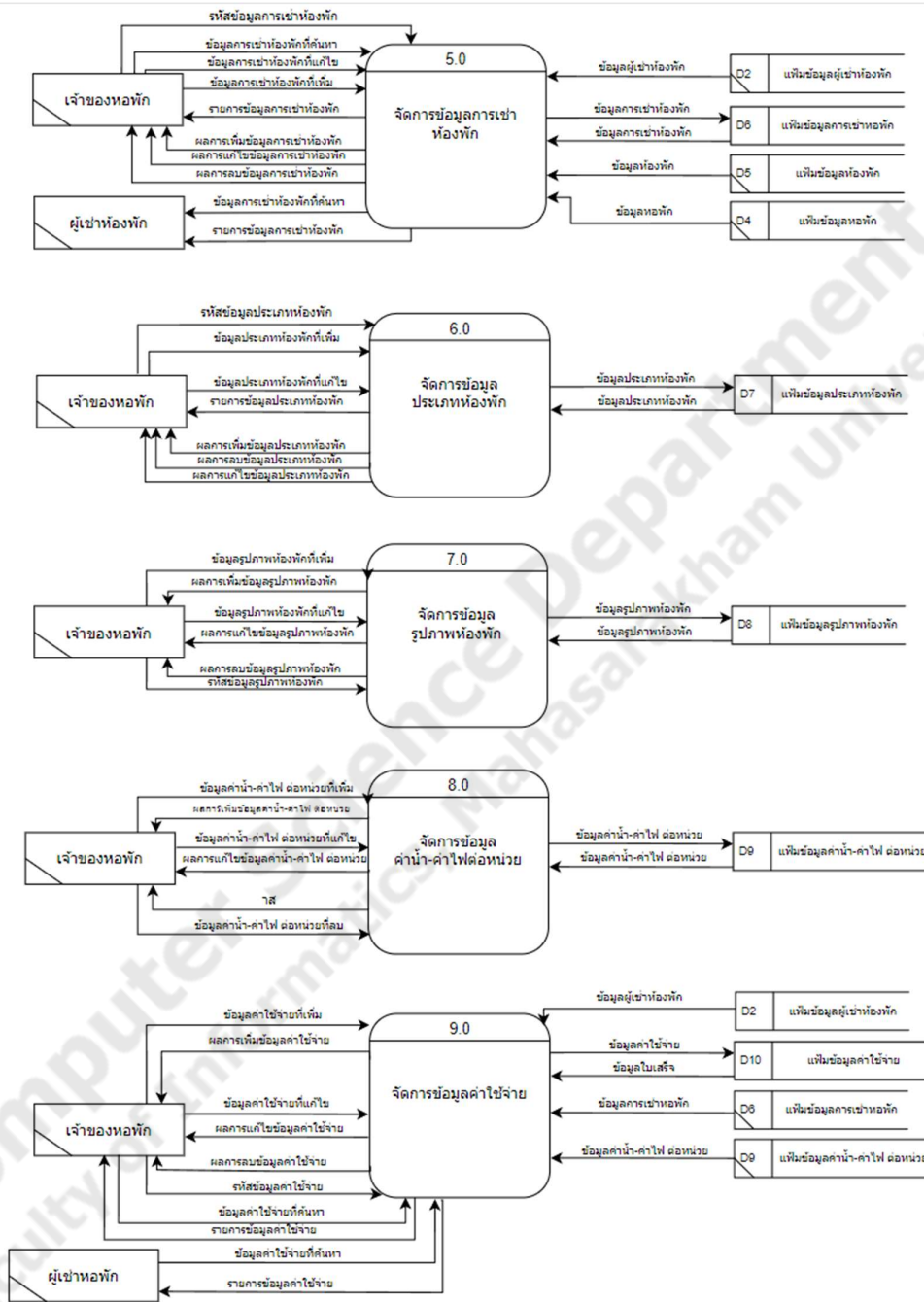
ภาพประกอบที่ 3.2 Context Diagram

3.2.2 Data Flow Diagram Level 1

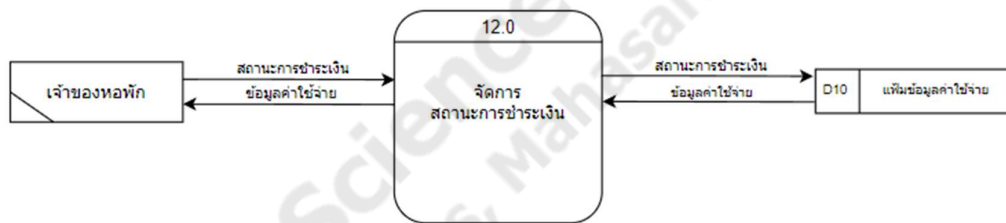
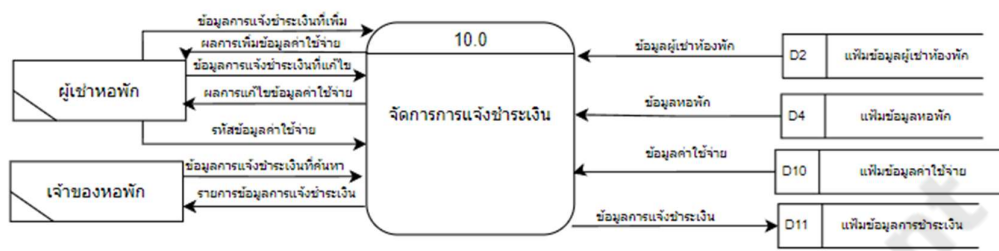
แผนภาพแสดงการไหลของข้อมูลจากแหล่งข้อมูลภายนอกเข้ามาในระบบ เพื่อแสดงวิธีการไหลของข้อมูลจากกระบวนการหนึ่งไปอีกกระบวนการหนึ่ง ซึ่งมีสัญลักษณ์ 4 แบบคือ สี่เหลี่ยม สี่เหลี่ยมมุมมน ลูกศร และสี่เหลี่ยมปลายเปิดข้างหนึ่ง ดังนี้



ภาพประกอบที่ 3.3 Data Flow Diagram Level 1



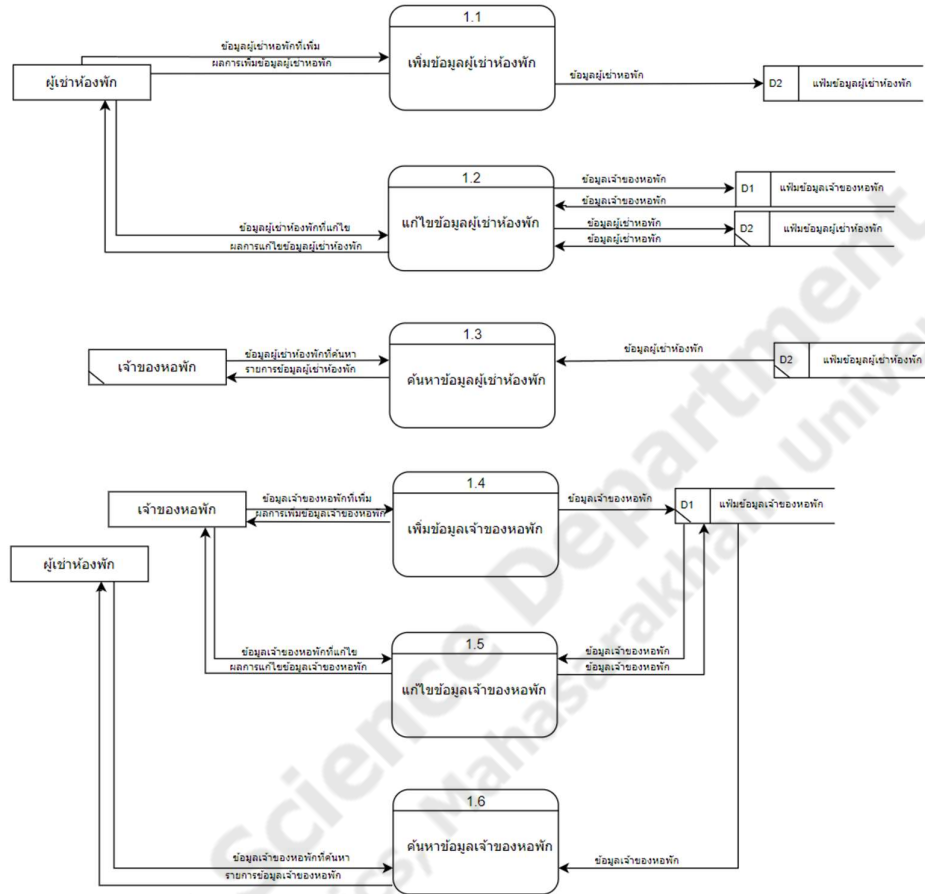
ภาพประกอบที่ 3.2 Data Flow Diagram Level 1 (ต่อ)



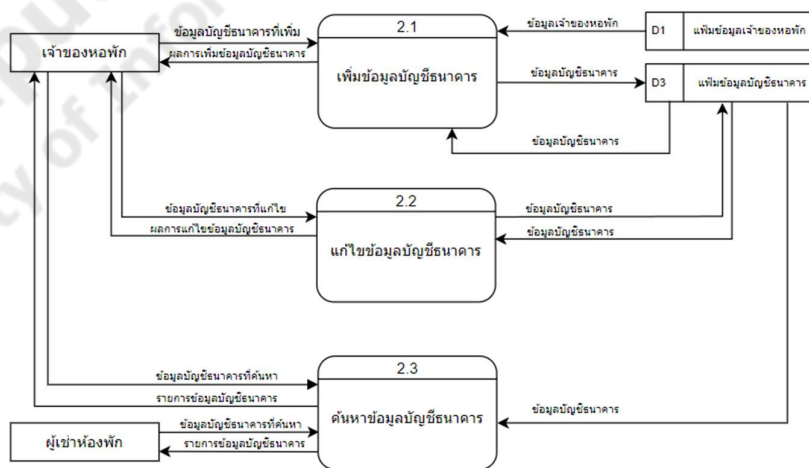
ภาพประกอบที่ 3.2 Data Flow Diagram Level 1 (ต่อ)

### 3.2.3 Data Flow Diagram Level 2

คือการแตกการประมวลผลย่อย โดยใช้แผนภาพกระแสข้อมูล DFD Level 1 แบ่งการประมวลผลภายในออกเป็นขั้นเป็นตอนต่างๆ

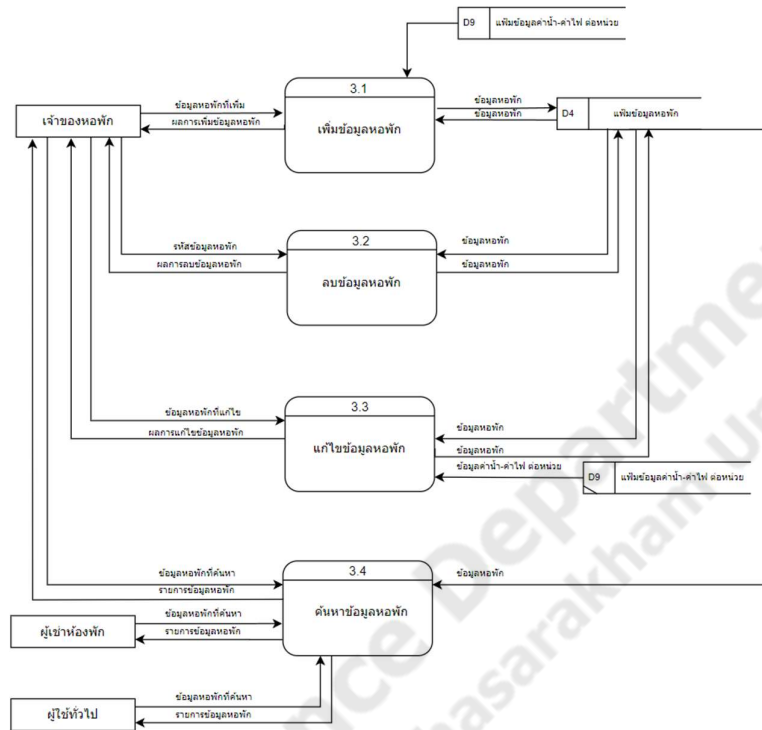


ภาพประกอบที่ 3.4 Data Flow Diagram Level 2 Process 1

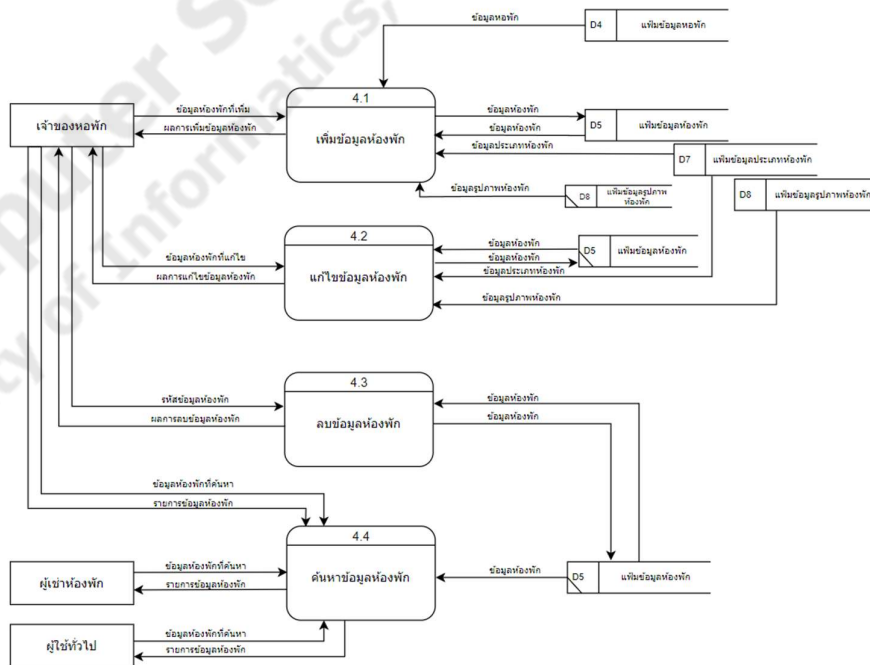


ภาพประกอบที่ 3.5 Data Flow Diagram Level 2 Process 2

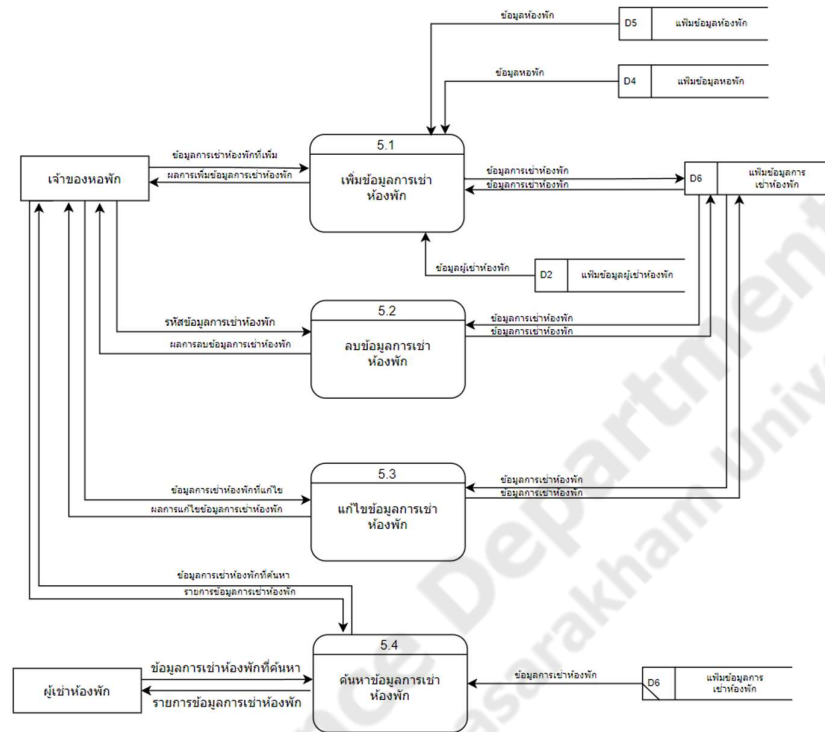




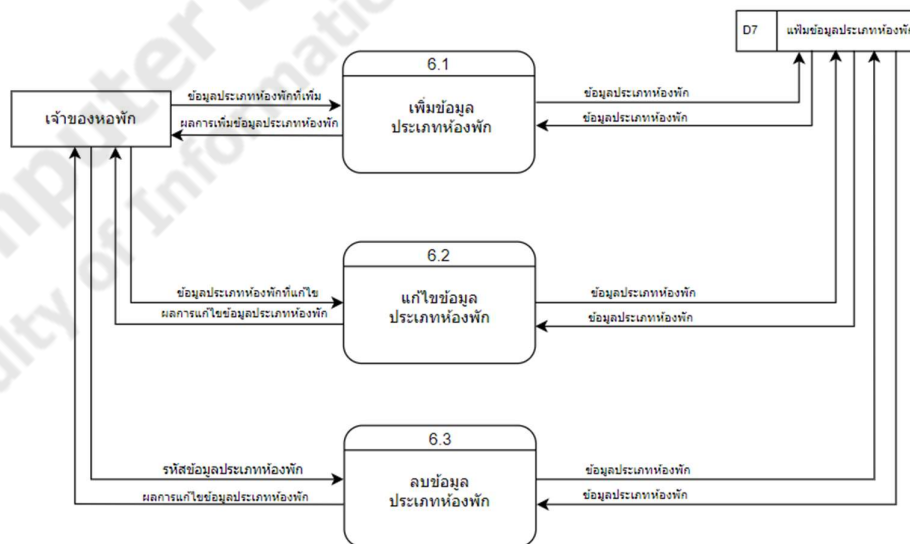
ภาพประกอบที่ 3.6 Data Flow Diagram Level 2 Process 3



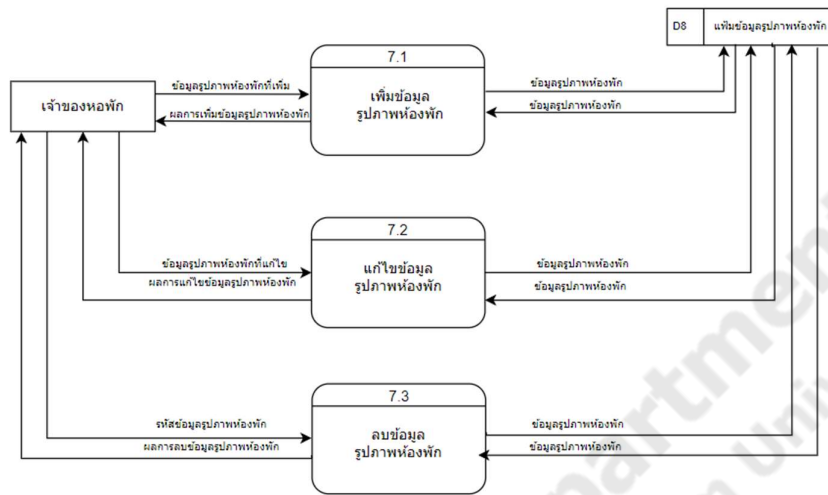
ภาพประกอบที่ 3.7 Data Flow Diagram Level 2 Process 4



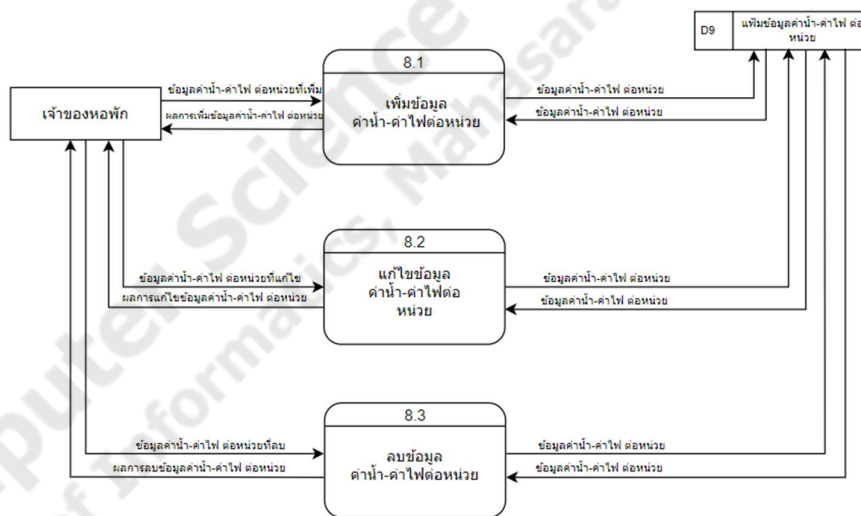
ภาพประกอบที่ 3.8 Data Flow Diagram Level 2 Process 5



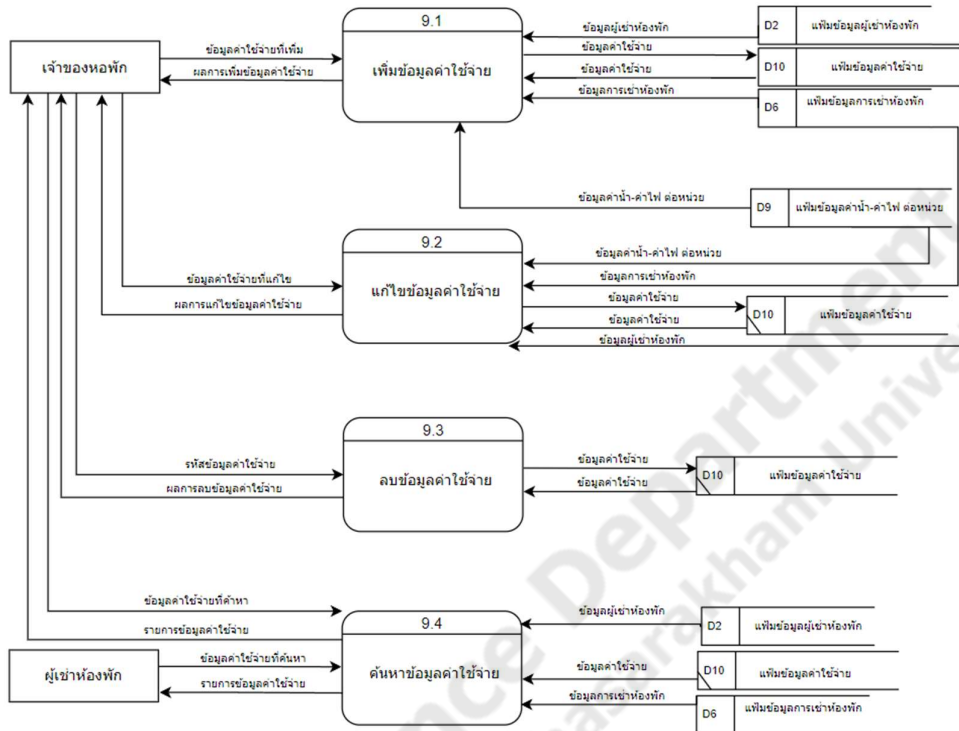
ภาพประกอบที่ 3.9 Data Flow Diagram Level 2 Process 6



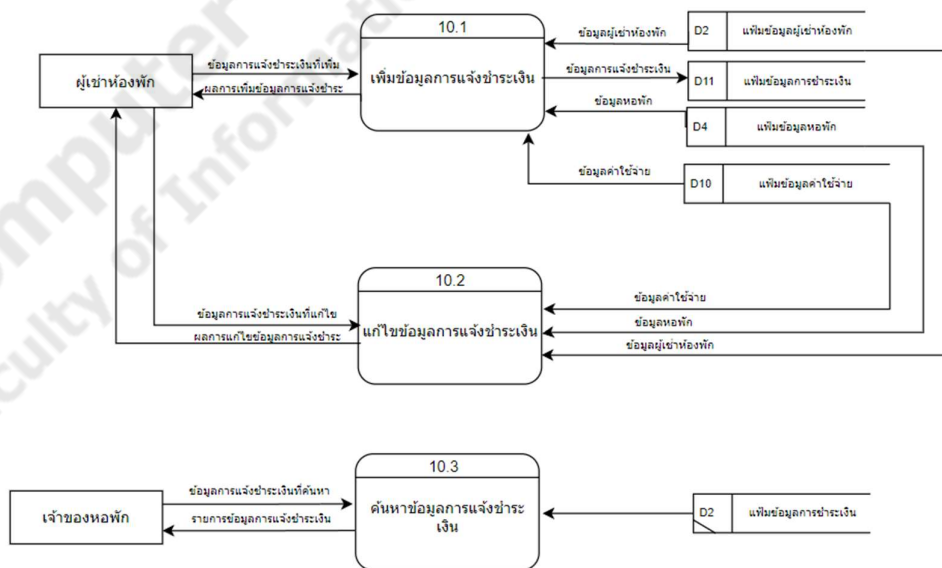
ภาพประกอบที่ 3.10 Data Flow Diagram Level 2 Process 7



ภาพประกอบที่ 3.11 Data Flow Diagram Level 2 Process 8



ภาพประกอบที่ 3.12 Data Flow Diagram Level 2 Process 9



ภาพประกอบที่ 3.13 Data Flow Diagram Level 2 Process 10

## สรุปการไหลเวียนของข้อมูล

แผนการไหลเวียนของข้อมูลของผู้จัดทำ จะมีทั้งหมด 3 level ด้วยกัน ประกอบไปด้วย

1. Context Diagram จะเป็นแผนภาพอธิบายการทำงานของระบบทั้งหมด โดยในแต่ละเส้นเข้าและออก จะแตกย่อย เป็น Data Flow Diagram level 1 ต่อไป

2. Data Flow Diagram level 1 แผนภาพแสดงการไหลเวียนของข้อมูลจากแหล่งภายนอกเข้ามาในระบบ ซึ่งมีทั้งหมด 17 process แต่ละ process จะบอกถึงกระบวนการของระบบแต่ละ process ว่าสามารถจัดการข้อมูลอะไรได้บ้าง โดยถ้า process ไหนสามารถทำได้มากกว่านั้น ก็จะแตกย่อยไปสู่ Data Flow Diagram level 2 ต่อไป

3. Data Flow Diagram level 2 การแตกการประมวลผลย่อย โดยใช้แผนภาพกระแสข้อมูลจาก DFD level 1 ซึ่งตัวที่แตกย่อยจะแสดง process ที่มีกระบวนการที่มากกว่านั้นหรือซับซ้อนกว่านั้น เช่น เพิ่ม ลบ แก้ไข และค้นหา ของระบบเพื่อให้เห็นว่า process นั้นมีกระบวนการทำงานได้มาก น้อยแค่ไหน และเพื่อให้คนพัฒนาระบบเข้าใจกระบวนการของระบบได้ง่ายต่อการพัฒนาต่อไป

### 3.3 พจนานุกรมข้อมูล(Data Dictionary)

#### 3.3.1 External Entity Description

พจนานุกรมข้อมูลช่วยอธิบายการทำงานของแผนภาพการไหลของข้อมูล (Data Flow Dictionaries)

ตารางที่ 3.1 External Entity Description

ผู้ใช้งาน	ความหมาย	Input Data	Output Data
เจ้าของหอพัก	เจ้าของหอพัก	<ul style="list-style-type: none"> <li>- รหัสข้อมูลหอพัก</li> <li>- ข้อมูลบัญชีธนาคารที่เพิ่ม</li> <li>- ข้อมูลบัญชีธนาคารที่แก้ไข</li> <li>- ข้อมูลบัญชีธนาคารที่ค้นหา</li> </ul>	<ul style="list-style-type: none"> <li>- รายการข้อมูลเจ้าของหอพัก</li> <li>- ผลการเพิ่มข้อมูลเจ้าของหอพัก</li> <li>- ผลการแก้ไขข้อมูลเจ้าของหอพัก</li> <li>- ข้อมูลบัญชีธนาคาร</li> </ul>

ตารางที่ 3.1 External Entity Description (ต่อ)

ผู้ใช้งาน	ความหมาย	Input Data	Output Data
		<ul style="list-style-type: none"> <li>- ข้อมูลเจ้าของห้องพักที่เพิ่ม</li> <li>- ข้อมูลเจ้าของห้องพักที่แก้ไข</li> <li>- ข้อมูลเจ้าของห้องพักที่ค้นหา</li> <li>- รหัสข้อมูลการเช่าห้องพัก</li> <li>- ข้อมูลการเช่าห้องพักที่เพิ่ม</li> <li>- ข้อมูลการเช่าห้องพักที่แก้ไข</li> <li>- ข้อมูลการเช่าห้องพักที่ค้นหา</li> <li>- รหัสข้อมูลประเภทห้องพัก</li> <li>- ข้อมูลประเภทห้องพักที่เพิ่ม</li> <li>- ข้อมูลประเภทห้องพักที่แก้ไข</li> <li>- ข้อมูลรูปภาพห้องพักที่เพิ่ม</li> <li>- ข้อมูลรูปภาพห้องพักที่แก้ไข</li> <li>- รหัสข้อมูลรูปภาพห้องพัก</li> <li>- ข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วยที่เพิ่ม</li> </ul>	<ul style="list-style-type: none"> <li>- ผลการเพิ่มข้อมูลบัญชีธนาคาร</li> <li>- ผลการแก้ไขข้อมูลบัญชีธนาคาร</li> <li>- ผลการเพิ่มข้อมูลห้องพัก</li> <li>- ผลการแก้ไขข้อมูลห้องพัก</li> <li>- ผลการลบข้อมูลห้องพัก</li> <li>- รายการข้อมูลห้องพัก</li> <li>- ผลการเพิ่มข้อมูลห้องพัก</li> <li>- ผลการแก้ไขข้อมูลห้องพัก</li> <li>- รายการข้อมูลห้องพัก</li> <li>- ผลการลบข้อมูลห้องพัก</li> <li>- รายการข้อมูลการเช่าห้องพัก</li> <li>- ผลการเพิ่มข้อมูลการเช่าห้องพัก</li> <li>- ผลการแก้ไขข้อมูลการเช่าห้องพัก</li> <li>- ผลการลบข้อมูลการเช่าห้องพัก</li> <li>- รายการข้อมูลประเภทห้องพัก</li> </ul>

ตารางที่ 3.1 External Entity Description (ต่อ)

ผู้ใช้งาน	ความหมาย	Input Data	Output Data
		<ul style="list-style-type: none"> <li>- ข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วยที่แก้ไข</li> <li>- ข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วยที่ลบ</li> <li>- ข้อมูลค่าใช้จ่ายที่เพิ่ม</li> <li>- ข้อมูลค่าใช้จ่ายที่แก้ไข</li> <li>- ข้อมูลค่าใช้จ่ายที่ค้นหา</li> <li>- รหัสข้อมูลค่าใช้จ่าย</li> <li>- ข้อมูลการแจ้งชำระเงินที่ค้นหา</li> <li>- ข้อมูลการเข้าสู่ระบบ</li> <li>- สถานะการชำระเงิน</li> </ul>	<ul style="list-style-type: none"> <li>- ผลการเพิ่มข้อมูลประเภทห้องพัก</li> <li>- ผลการแก้ไขข้อมูลประเภทห้องพัก</li> <li>- ผลการลบข้อมูลประเภทห้องพัก</li> <li>- ผลการเพิ่มข้อมูลรูปภาพห้องพัก</li> <li>- ผลการแก้ไขข้อมูลรูปภาพห้องพัก</li> <li>- ผลการลบข้อมูลรูปภาพห้องพัก</li> <li>- ผลการเพิ่มข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วย</li> <li>- ผลการแก้ไขข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วย</li> <li>- ผลการลบข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วย</li> <li>- ผลการเพิ่มข้อมูลค่าใช้จ่าย</li> <li>- ผลการแก้ไขข้อมูลค่าใช้จ่าย</li> <li>- ผลการลบข้อมูลค่าใช้จ่าย</li> <li>- รายการข้อมูลค่าใช้จ่าย</li> <li>- รายการข้อมูลการแจ้งชำระเงิน</li> <li>- ผลการเข้าสู่ระบบ</li> </ul>

ตารางที่ 3.1 External Entity Description (ต่อ)

ผู้ใช้งาน	ความหมาย	Input Data	Output Data
ผู้เช่าห้องพัก	ผู้เช่าห้องพัก	<ul style="list-style-type: none"> <li>- ข้อมูลผู้เช่าห้องพักที่เพิ่ม</li> <li>- ข้อมูลผู้เช่าห้องพักที่แก้ไข</li> <li>- ข้อมูลเจ้าของห้องพักที่ค้นหา</li> <li>- ข้อมูลบัญชีธนาคารที่ค้นหา</li> <li>- ข้อมูลห้องพักที่ค้นหา</li> <li>- ข้อมูลห้องพักที่ค้นหา</li> <li>- รายการข้อมูลห้องพัก</li> <li>- ข้อมูลการเช่าห้องพักที่ค้นหา</li> <li>- ข้อมูลค่าใช้จ่ายที่ค้นหา</li> <li>- ข้อมูลการแจ้งชำระเงินที่เพิ่ม</li> <li>- ข้อมูลการแจ้งชำระเงินที่แก้ไข</li> <li>- รหัสข้อมูลค่าใช้จ่าย</li> <li>- ข้อมูลการเข้าสู่ระบบ</li> </ul>	<ul style="list-style-type: none"> <li>- ผลการเข้าสู่ระบบ</li> <li>- ผลการเพิ่มข้อมูลค่าใช้จ่าย</li> <li>- ผลการแก้ไขข้อมูลค่าใช้จ่าย</li> <li>- รายการข้อมูลใบเสร็จ</li> <li>- รายการข้อมูลห้องพัก</li> <li>- รายการข้อมูลการ</li> <li>- ผลการเพิ่มข้อมูลผู้เช่าห้องพักเช่าห้องพัก</li> <li>- ผลการแก้ไขข้อมูลผู้เช่าห้องพัก</li> <li>- รายการข้อมูลบัญชีธนาคาร</li> </ul>
ผู้ใช้ทั่วไป	ผู้ใช้ทั่วไป	<ul style="list-style-type: none"> <li>- ข้อมูลห้องพักที่ค้นหา</li> <li>- ข้อมูลห้องพักที่ค้นหา</li> </ul>	<ul style="list-style-type: none"> <li>- รายการข้อมูลห้องพัก</li> <li>- รายการข้อมูลห้องพัก</li> </ul>



### 3.3.2 Data Flow and Data Structure Description

เป็นการแสดงขั้นตอนการทำงานทิศทางการไหลของข้อมูลว่ามีทิศทางเป็นอย่างไร และมีข้อมูลใดบ้างที่ถูกส่งออกและส่งเข้า

ตารางที่ 3.2 Data Flow and Data Structure Description

Name	Description	Source	Destination	Data Structure
ข้อมูลผู้เช่า ห้องพักราคาค้นหา	การส่งข้อมูล keyword ผู้เช่า ห้องพักราคาค้นหา ค้นหา	Process 1.3 ค้นหาข้อมูลผู้เช่า ห้องพักราคาค้นหา	เจ้าของห้องพัก	[User_id   email   name   lastname   tel_no]
ข้อมูลเจ้าของ ห้องพักที่แก้ไข	การส่งข้อมูล เจ้าของห้องพักที่ แก้ไขแล้ว	เจ้าของห้องพัก	Process 1.5 แก้ไขข้อมูล เจ้าของห้องพัก	รหัสเจ้าของห้องพัก + email + password + name + lastname + tel_no_ + image+ status
ข้อมูลเจ้าของ ห้องพักที่เพิ่ม	การเพิ่มข้อมูล เจ้าของห้องพัก	เจ้าของห้องพัก	Process 1.4 เพิ่ม ข้อมูลห้องพัก	รหัสเจ้าของห้องพัก + email + password + name + lastname + tel_no_ + (image) + status
ผลการแก้ไข ข้อมูลเจ้าของ ห้องพัก	การส่งข้อมูลผล การ แก้ไขข้อมูล เจ้าของห้องพัก	Process 1.5 แก้ไขข้อมูลห้องพัก	เจ้าของห้องพัก	ผลการแก้ไขข้อมูล เจ้าของห้องพัก

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ผลการเพิ่มข้อมูล เจ้าของหอพัก	การส่งข้อมูลผล การ เพิ่มข้อมูล เจ้าของหอพัก	Process 1.1 เพิ่มข้อมูลหอพัก	เจ้าของหอพัก	ผลการเพิ่มข้อมูล เจ้าของหอพัก
รายการข้อมูลผู้ เช่าหอพัก	การส่งข้อมูล รายการที่ได้จาก การค้นหาข้อมูลผู้ เช่าหอพัก	Process 1.3 ค้นหาข้อมูลผู้เช่า หอพัก	เจ้าของหอพัก	{User_id + email + name + lastname + tel_no_ + image + status}
ข้อมูลผู้เช่า หอพักที่แก้ไข	การส่งข้อมูลผู้เช่า หอพักที่แก้ไข แล้ว	ผู้เช่าหอพัก	Process 1.2 จัดการข้อมูลผู้เช่า หอพัก	User_id + email + name + lastname +
ข้อมูลผู้เช่า หอพักที่เพิ่ม	การส่งข้อมูลผู้เช่า หอพักเพิ่ม	ผู้เช่าหอพัก	Process 1.1 เพิ่ม ข้อมูลผู้เช่า หอพัก	tel_no_ + image + status
ข้อมูลเจ้าของ หอพักที่ค้นหา	ส่งข้อมูล keyword ที่จะใช้ ค้นหาข้อมูล เจ้าของหอพัก	ผู้เช่าหอพัก	Process 1.6 ค้นหาข้อมูล เจ้าของหอพัก	[email   name   lastname   tel_no]
รายการข้อมูล เจ้าของหอพัก	ส่งข้อมูลที่ได้จาก การค้นหาข้อมูล เจ้าของหอพัก	Process 1.6 ค้นหาข้อมูล เจ้าของหอพัก	ผู้เช่าหอพัก	{รหัสเจ้าของ หอพัก + email + name + lastname + tel_no_ + image + status}
ข้อมูลผู้เช่า หอพัก	ข้อมูลผู้เช่า หอพัก	Process 1.1 เพิ่มข้อมูลผู้เช่า หอพัก	D2 เพิ่มข้อมูลผู้ เช่าหอพัก	User_id +. email + password + name +

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		Process 1.2 แก้ไขข้อมูลผู้เช่า ห้องพัก	D2 เพิ่มข้อมูลผู้ เช่าห้องพัก	lastname + tel_no_ + image + status
		D2 เพิ่มข้อมูลผู้ เช่าห้องพัก	Process 1.2 แก้ไขข้อมูลผู้เช่า ห้องพัก	
		D2 เพิ่มข้อมูลผู้ เช่าห้องพัก	Process 1.3 ค้นหาข้อมูลผู้เช่า ห้องพัก	
ข้อมูลเจ้าของ ห้องพัก	ข้อมูลเจ้าของ ห้องพัก	Process 1.4 เพิ่มข้อมูลเจ้าของ ห้องพัก	D1 เพิ่มข้อมูล เจ้าของห้องพัก	รหัสเจ้าของห้องพัก + email + password + name +
		D1 เพิ่มข้อมูล เจ้าของห้องพัก	Process 1.5 แก้ไขข้อมูล เจ้าของห้องพัก	lastname + tel_no_ + image + status
		Process 1.5 แก้ไขข้อมูล เจ้าของห้องพัก	D1 เพิ่มข้อมูล เจ้าของห้องพัก	
		D1 เพิ่มข้อมูล เจ้าของห้องพัก	Process 1.6 ค้นหาข้อมูล เจ้าของห้องพัก	
		D1 เพิ่มข้อมูล เจ้าของห้องพัก	Process 2.1 เพิ่ม ข้อมูลบัญชี ธนาคาร	
ข้อมูลบัญชี ธนาคารที่เพิ่ม	การส่งข้อมูลบัญชี ธนาคารที่เพิ่ม	เจ้าของห้องพัก	Process 2.1 เพิ่ม ข้อมูลบัญชี ธนาคาร	รหัสบัญชี + เลข บัญชี + ชื่อบัญชี + ธนาคาร

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ผลการเพิ่มข้อมูล บัญชีธนาคาร	รายงานผลการ เพิ่มข้อมูลบัญชี ธนาคาร	Process 2.1 เพิ่มข้อมูลบัญชี ธนาคาร	เจ้าของหอพัก	ผลการเพิ่มข้อมูล บัญชีธนาคาร
ผลการแก้ไขบัญชี ธนาคาร	รายงานผลการ แก้ไขข้อมูลบัญชี ธนาคาร	Process 2.2 แก้ไขข้อมูลบัญชี ธนาคาร	เจ้าของหอพัก	ผลการแก้ไขบัญชี ธนาคาร
ข้อมูลบัญชี ธนาคารที่แก้ไข	ส่งข้อมูลบัญชี ธนาคารที่แก้ไข	เจ้าของหอพัก	Process 2.2 แก้ไขข้อมูลบัญชี ธนาคาร	รหัสบัญชี + เลข บัญชี + ชื่อบัญชี + ธนาคาร
ข้อมูลบัญชี ธนาคารที่ค้นหา	ส่งข้อมูล keyword ของ บัญชีธนาคารที่ ค้นหา	เจ้าของหอพัก	Process 2.3 ค้นหาข้อมูลบัญชี ธนาคาร	[ เลขบัญชี   ชื่อบัญชี   ธนาคาร]
		ผู้เช่าห้องพัก	Process 2.3 ค้นหาข้อมูลบัญชี ธนาคาร	
ข้อมูลบัญชี ธนาคาร	ข้อมูลบัญชี ธนาคาร	Process 2.1 เพิ่มข้อมูลบัญชี ธนาคาร	D3 เพิ่มข้อมูล เจ้าของหอพัก	รหัสบัญชี + เลข บัญชี + ชื่อบัญชี + ธนาคาร
		D3 เพิ่มข้อมูล เจ้าของหอพัก	Process 2.1 เพิ่ม ข้อมูลบัญชี ธนาคาร	
		Process 2.2 แก้ไขข้อมูลบัญชี ธนาคาร	D3 เพิ่มข้อมูล เจ้าของหอพัก	
		D3 เพิ่มข้อมูล เจ้าของหอพัก	Process 2.2 แก้ไขข้อมูลบัญชี ธนาคาร	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		D3 เพิ่มข้อมูล เจ้าของหอพัก	Process 2.3 ค้นหาข้อมูลบัญชี ธนาคาร	
ข้อมูลหอพักที่ เพิ่ม	ส่งข้อมูลหอพักที่ ต้องการเพิ่ม	เจ้าของหอพัก	Process 3.1 เพิ่ม ข้อมูลหอพัก	รหัสหอพัก + ชื่อ หอพัก + รายละเอียด ห้องพัก + ที่อยู่ หอพัก + (facebook) + (Line) +เบอร์ โทร+ รูปภาพ + ฟอร์มสัญญาเช่า
ผลการเพิ่มข้อมูล หอพัก	ส่งข้อมูลผลการ เพิ่มข้อมูลหอพัก	Process 3.1 เพิ่มข้อมูลหอพัก	เจ้าของหอพัก	ผลการเพิ่มข้อมูล หอพัก
ผลการลบข้อมูล หอพัก	ส่งข้อมูลผลการ ลบข้อมูลหอพัก	Process 3.2 ลบ ข้อมูลหอพัก	เจ้าของหอพัก	
ผลการแก้ไข ข้อมูลหอพัก	ส่งข้อมูลผลการ แก้ไขข้อมูลหอพัก	Process 3.3 แก้ไขข้อมูลหอพัก	เจ้าของหอพัก	
รายการข้อมูล หอพัก	ส่งรายการข้อมูล หอพัก	Process 3.4 ค้นหาข้อมูล หอพัก	เจ้าของหอพัก	{รหัสหอพัก + ชื่อ หอพัก + รายละเอียด ห้องพัก + ที่อยู่ หอพัก + facebook + Line + เบอร์โทร + รูปภาพ + ฟอร์มสัญญาเช่า}
		Process 3.4 ค้นหาข้อมูล หอพัก	ผู้เช่าห้องพัก	
		Process 3.4 ค้นหาข้อมูล หอพัก	ผู้ใช้ทั่วไป	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
รหัสข้อมูลหอพัก	ส่งรหัสข้อมูลหอพัก	เจ้าของหอพัก	Process 3.2 ลบข้อมูลหอพัก	รหัสข้อมูลหอพัก
ข้อมูลหอพักที่แก้ไข	ส่งข้อมูลหอพักที่แก้ไข	เจ้าของหอพัก	Process 3.3 แก้ไขข้อมูลหอพัก	รหัสหอพัก + ชื่อหอพัก + รายละเอียดห้องพัก + ที่อยู่หอพัก + facebook + Line + เบอร์โทร + รูปภาพ + ฟอรัมสัญญาเช่า
ข้อมูลหอพักที่ค้นหา	ส่งข้อมูล keyword ของหอพักที่ค้นหา	เจ้าของหอพัก	Process 3.4 ค้นหาข้อมูลหอพัก	{ชื่อหอพัก + รายละเอียดห้องพัก + ที่อยู่หอพัก + facebook + Line + เบอร์โทร + รูปภาพ}
		ผู้เช่าห้องพัก	Process 3.4 ค้นหาข้อมูลหอพัก	
		ผู้ใช้ทั่วไป	Process 3.4 ค้นหาข้อมูลหอพัก	
ข้อมูลหอพัก	ข้อมูลหอพัก	Process 3.1 เพิ่มข้อมูลหอพัก	D4 เพิ่มข้อมูลหอพัก	รหัสหอพัก + ชื่อหอพัก + รายละเอียดห้องพัก + ที่อยู่หอพัก + facebook + Line + เบอร์โทร
		D4 เพิ่มข้อมูลหอพัก	Process 3.1 เพิ่มข้อมูลหอพัก	
		Process 3.2 ลบข้อมูลหอพัก	D4 เพิ่มข้อมูลหอพัก	
		D4 เพิ่มข้อมูลหอพัก	Process 3.2 ลบข้อมูลหอพัก	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		Process 3.3 แก้ไขข้อมูลห้องพัก	D4 เพิ่มข้อมูล ห้องพัก	+ รูปภาพ + ฟอร์มสัญญาเช่า
		D4 เพิ่มข้อมูล ห้องพัก	Process 3.3 แก้ไขข้อมูลห้องพัก	
		D4 เพิ่มข้อมูล ห้องพัก	Process 3.4 ค้นหาข้อมูล ห้องพัก	
ข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	ข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 3.1 เพิ่มข้อมูลห้องพัก	รหัสค่าน้ำไฟต่อหน่วย + ค่าน้ำต่อหน่วย + ค่าไฟต่อหน่วย
		D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 3.3 แก้ไขข้อมูลห้องพัก	
ข้อมูลห้องพักที่เพิ่ม	ส่งข้อมูลห้องพักที่ต้องการเพิ่ม	เจ้าของห้องพัก	Process 4.1 เพิ่มข้อมูลห้องพัก	รหัสห้องพัก + หมายเลขห้องพัก
ข้อมูลห้องพักที่แก้ไข	ส่งข้อมูลห้องพักที่แก้ไข	เจ้าของห้องพัก	Process 4.2 แก้ไขข้อมูลห้องพัก	+ รายละเอียดห้องพัก + ราคาห้องพัก + ราคาประกัน + สถานะ
รหัสข้อมูลห้องพัก	ส่งรหัสข้อมูลห้องพัก	เจ้าของห้องพัก	Process 4.3 ลบข้อมูลห้องพัก	รหัสห้องพัก
ข้อมูลห้องพักที่ค้นหา	ส่ง keyword ของห้องพักที่ต้องการค้นหา	เจ้าของห้องพัก	Process 4.4 ค้นหาข้อมูลห้องพัก	[ หมายเลขห้องพัก   ราคาห้องพัก   สถานะ ]
		ผู้เช่าห้องพัก	Process 4.4 ค้นหาข้อมูลห้องพัก	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		ผู้ใช้ทั่วไป	Process 4.4 ค้นหาข้อมูล ห้องพัก	
ผลการเพิ่มข้อมูล ห้องพัก	รายงานผลการ เพิ่มข้อมูลห้องพัก	Process 4.1 เพิ่มข้อมูลห้องพัก	เจ้าของห้องพัก	ผลการเพิ่มข้อมูล ห้องพัก
ผลการแก้ไข ข้อมูลห้องพัก	รายงานผลการ แก้ไขข้อมูล ห้องพัก	Process 4.2 แก้ไขข้อมูล ห้องพัก	เจ้าของห้องพัก	ผลการแก้ไขข้อมูล ห้องพัก
ผลการลบข้อมูล ห้องพัก	รายงานผลการ ลบข้อมูลห้องพัก	Process 4.3 ลบ ข้อมูลห้องพัก	เจ้าของห้องพัก	ผลการลบข้อมูล ห้องพัก
รายการข้อมูล ห้องพัก	ส่งข้อมูลรายการ ห้องพัก	Process 4.4 ค้นหาข้อมูล ห้องพัก	เจ้าของห้องพัก	{รหัสห้องพัก + หมายเลขห้องพัก + รายละเอียด ห้องพัก + ราคา ห้องพัก + ราคา ประกัน + สถานะ}
		Process 4.4 ค้นหาข้อมูล ห้องพัก	ผู้เช่าห้องพัก	
		Process 4.4 ค้นหาข้อมูล ห้องพัก	ผู้ใช้ทั่วไป	
ข้อมูลห้องพัก	ข้อมูลห้องพัก	Process 4.1 เพิ่มข้อมูลห้องพัก	D5 เพิ่มข้อมูล ห้องพัก	รหัสห้องพัก + หมายเลขห้องพัก + รายละเอียด ห้องพัก + ราคา ห้องพัก + ราคา ประกัน + สถานะ
		D5 เพิ่มข้อมูล ห้องพัก	Process 4.1 เพิ่ม ข้อมูลห้องพัก	
		Process 4.2 แก้ไขข้อมูล ห้องพัก	D5 เพิ่มข้อมูล ห้องพัก	



ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		D5 เพิ่มข้อมูล ห้องพัก	Process 4.2 แก้ไขข้อมูล ห้องพัก	
		Process 4.3 ลบ ข้อมูลห้องพัก	D5 เพิ่มข้อมูล ห้องพัก	
		D5 เพิ่มข้อมูล ห้องพัก	Process 4.3 ลบ ข้อมูลห้องพัก	
ข้อมูลห้องพัก	ดึงข้อมูลห้องพัก	D4 เพิ่มข้อมูล ห้องพัก	Process 4.1 เพิ่ม ข้อมูลห้องพัก	รหัสข้อมูลห้องพัก
ข้อมูลประเภท ห้องพัก	ดึงข้อมูลประเภท ห้องพัก	D7 เพิ่มข้อมูล ประเภทห้องพัก	Process 4.1 เพิ่ม ข้อมูลห้องพัก	รหัสข้อมูล ประเภทห้องพัก + ประเภท
		D7 เพิ่มข้อมูล ประเภทห้องพัก	Process 4.2 แก้ไขข้อมูล ห้องพัก	
ข้อมูลรูปภาพ ห้องพัก	ดึงข้อมูลรูปภาพ ห้องพัก	D8 เพิ่มข้อมูล รูปภาพห้องพัก	Process 4.1 เพิ่ม ข้อมูลห้องพัก	รหัสข้อมูลรูปภาพ ห้องพัก + รูปภาพ
		D8 เพิ่มข้อมูล รูปภาพห้องพัก	Process 4.2 แก้ไขข้อมูล ห้องพัก	
ข้อมูลการเช่า ห้องพักที่เพิ่ม	ส่งข้อมูลการเช่า ห้องพักที่ต้องการ เพิ่ม	เจ้าของห้องพัก	Process 5.1 เพิ่ม ข้อมูลการเช่า ห้องพัก	รหัสข้อมูลการเช่า ห้องพัก + สถานะการเช่า + (วันเช่า) + (วัน ออก) + สัญญา เช่า + วันที่ทำ สัญญา + วันที่ หมดสัญญา

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
รหัสข้อมูลการ เช่าห้องพัก	ส่งรหัสข้อมูลการ เช่าห้องพัก	เจ้าของห้องพัก	Process 5.2 ลบ ข้อมูลการเช่า ห้องพัก	รหัสข้อมูลการเช่า ห้องพัก
ข้อมูลการเช่า ห้องพักที่แก้ไข	ส่งข้อมูลการเช่า ห้องพักที่ต้องการ แก้ไข	เจ้าของห้องพัก	Process 5.3 แก้ไขข้อมูลการ เช่าห้องพัก	รหัสข้อมูลการเช่า ห้องพัก + สถานะการเช่า + วันเช่า + วันออก + สัญญาเช่า + วันที่ทำสัญญา + วันที่หมดสัญญา
ข้อมูลการเช่า ห้องพักที่ค้นหา	ส่ง keyword ของข้อมูลการเช่า ห้องพักที่ต้องการ ค้นหา	เจ้าของห้องพัก	Process 5.4 ค้นหาข้อมูลการ เช่าห้องพัก	[รหัสข้อมูลการ เช่าห้องพัก   สถานะการเช่า
		ผู้เช่าห้องพัก	Process 5.4 ค้นหาข้อมูลการ เช่าห้องพัก	ชื่อผู้เช่า   นามสกุลผู้เช่า]
ผลการเพิ่มข้อมูล การเช่าห้อง พัก	รายงานผลการ เพิ่มข้อมูลการเช่า ห้อง พัก	Process 5.1 เพิ่มข้อมูลการเช่า ห้องพัก	เจ้าของห้องพัก	ผลการเพิ่มข้อมูล การเช่าห้อง พัก
ผลการลบข้อมูล การเช่าห้องพัก	รายงานผลการ ลบข้อมูลการเช่า ห้องพัก	Process 5.2 ลบ ข้อมูลการเช่า ห้องพัก	เจ้าของห้องพัก	ผลการลบข้อมูล การเช่าห้องพัก
ผลการแก้ไข ข้อมูลการเช่า ห้องพัก	รายงานผลการ แก้ไขข้อมูลการ เช่าห้องพัก	Process 5.3 แก้ไขข้อมูลการ เช่าห้องพัก	เจ้าของห้องพัก	ผลการแก้ไขข้อมูล การเช่าห้องพัก

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
รายการข้อมูล การเข้าห้องพัก	ส่งรายการข้อมูล การเข้าห้องพัก	Process 5.4 ค้นหาข้อมูลการ เข้าห้องพัก	เจ้าของห้องพัก	{รหัสข้อมูลการ เข้าห้องพัก + สถานะการเข้า + วันเข้า + วันออก + สัญญาเช่า + วันที่ทำสัญญา + วันที่หมดสัญญา}
		Process 5.4 ค้นหาข้อมูลการ เข้าห้องพัก	ผู้เช่าห้องพัก	
ข้อมูลการเช่า ห้องพัก	ข้อมูลการเช่า ห้องพัก	Process 5.1 เพิ่มข้อมูลการเช่า ห้องพัก	D6 เพิ่มข้อมูล การเช่าห้องพัก	รหัสข้อมูลการเช่า ห้องพัก + สถานะการเช่า + วันเข้า + วันออก + สัญญาเช่า + วันที่ทำสัญญา + วันที่หมดสัญญา
		D6 เพิ่มข้อมูล การเช่าห้องพัก	Process 5.1 เพิ่ม ข้อมูลการเช่า ห้องพัก	
		Process 5.2 ลบ ข้อมูลการเช่า ห้องพัก	D6 เพิ่มข้อมูล การเช่าห้องพัก	
		D6 เพิ่มข้อมูล การเช่าห้องพัก	Process 5.2 ลบ ข้อมูลการเช่า ห้องพัก	
		Process 5.3 แก้ไขข้อมูลการ เช่าห้องพัก	D6 เพิ่มข้อมูล การเช่าห้องพัก	
		D6 เพิ่มข้อมูล การเช่าห้องพัก	Process 5.3 แก้ไขข้อมูลการ เช่าห้องพัก	
		D6 เพิ่มข้อมูล การเช่าห้องพัก	Process 5.4 ค้นหาข้อมูลการ เช่าห้องพัก	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ข้อมูลผู้เช่า ห้องพัก	ดึงข้อมูลผู้เช่า ห้องพัก	D2 เพิ่มข้อมูลผู้ เช่าห้องพัก	Process 5.1 เพิ่ม ข้อมูลการเช่า ห้องพัก	User_id + image + email + name + lastname + tel_no + password
ข้อมูลห้องพัก	ดึงข้อมูลห้องพัก	D5 เพิ่มข้อมูล ห้องพัก	Process 5.1 เพิ่ม ข้อมูลการเช่า ห้องพัก	รหัสข้อมูลห้องพัก + รายละเอียด ห้องพัก + หมายเลข ห้องพัก + สถานะ + ราคาประกัน + ราคาห้องพัก
ข้อมูลห้องพัก	ดึงข้อมูลห้องพัก	D4 เพิ่มข้อมูล ห้องพัก	Process 5.1 เพิ่ม ข้อมูลการเช่า ห้องพัก	รหัสข้อมูลห้องพัก + ที่อยู่ห้องพัก + รายละเอียดห้องพัก + Line + facebook + เบอร์โทร + ชื่อ ห้องพัก + รูปภาพ + พอร์มสัญญา เช่า
ข้อมูลประเภท ห้องพักที่เพิ่ม	ส่งข้อมูลประเภท ห้องพักที่ต้องการ เพิ่ม	เจ้าของห้องพัก	Process 6.1 เพิ่ม ข้อมูลประเภท ห้องพัก	รหัสประเภท+ ประเภท
ข้อมูลประเภท ห้องพักที่แก้ไข	ส่งข้อมูลประเภท ห้องพักที่ต้องการ แก้ไข	เจ้าของห้องพัก	Process 6.2 แก้ไขข้อมูล ประเภทห้องพัก	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
รหัสข้อมูล ประเภทห้องพัก	ส่งรหัสข้อมูล ประเภทห้องพักที่ ต้องการลบ	เจ้าของห้องพัก	Process 6.3 ลบ ข้อมูลประเภท ห้องพัก	รหัสประเภท ห้องพัก
ผลการเพิ่มข้อมูล ประเภทห้องพัก	รายงานผลการ เพิ่มข้อมูล ประเภทห้องพัก	Process 6.1 เพิ่มข้อมูล ประเภทห้องพัก	เจ้าของห้องพัก	ผลการเพิ่มข้อมูล ประเภทห้องพัก
ผลการแก้ไข ข้อมูลประเภท ห้องพัก	รายงานผลการ แก้ไขข้อมูล ประเภทห้องพัก	Process 6.2 แก้ไขข้อมูล ประเภทห้องพัก	เจ้าของห้องพัก	ผลการแก้ไขข้อมูล ประเภทห้องพัก
ผลการลบข้อมูล ประเภทห้องพัก	รายงานผลการ ลบข้อมูลประเภท ห้องพัก	Process 6.3 ลบ ข้อมูลประเภท ห้องพัก	เจ้าของห้องพัก	ผลการลบข้อมูล ประเภทห้องพัก
ข้อมูลประเภท ห้องพัก	ข้อมูลประเภท ห้องพัก	Process 6.1 เพิ่มข้อมูล ประเภทห้องพัก	D7 เพิ่มข้อมูล ประเภทห้องพัก	รหัสประเภท ห้องพัก + ประเภทห้องพัก
		D7 เพิ่มข้อมูล ประเภทห้องพัก	Process 6.1 เพิ่ม ข้อมูลประเภท ห้องพัก	
		Process 6.2 แก้ไขข้อมูล ประเภทห้องพัก	D7 เพิ่มข้อมูล ประเภทห้องพัก	
		D7 เพิ่มข้อมูล ประเภทห้องพัก	Process 6.2 แก้ไขข้อมูล ประเภทห้องพัก	
		Process 6.3 ลบ ข้อมูลประเภท ห้องพัก	D7 เพิ่มข้อมูล ประเภทห้องพัก	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		D7 เพิ่มข้อมูลประเภทห้องพัก	Process 6.3 ลบข้อมูลประเภทห้องพัก	
ข้อมูลรูปภาพห้องพักที่เพิ่ม	ส่งข้อมูลรูปภาพห้องพักที่ต้องการเพิ่ม	เจ้าของหอพัก	Process 7.1 เพิ่มข้อมูลรูปภาพห้องพัก	รหัสรูปภาพ + รูปภาพ
ข้อมูลรูปภาพห้องพักที่แก้ไข	ส่งข้อมูลรูปภาพห้องพักที่ต้องการแก้ไข	เจ้าของหอพัก	Process 7.2 แก้ไขข้อมูลรูปภาพห้องพัก	
รหัสข้อมูลรูปภาพห้องพัก	ส่งรหัสข้อมูลรูปภาพห้องพักที่ต้องการลบ	เจ้าของหอพัก	Process 7.3 ลบข้อมูลรูปภาพห้องพัก	รหัสรูปภาพ
ผลการเพิ่มข้อมูลรูปภาพห้องพัก	รายงานผลการเพิ่มข้อมูลรูปภาพห้องพัก	Process 7.1 เพิ่มข้อมูลรูปภาพห้องพัก	เจ้าของหอพัก	ผลการเพิ่มข้อมูลรูปภาพห้องพัก
ผลการแก้ไขข้อมูลรูปภาพห้องพัก	รายงานผลการแก้ไขข้อมูลรูปภาพห้องพัก	Process 7.2 แก้ไขข้อมูลรูปภาพห้องพัก	เจ้าของหอพัก	ผลการแก้ไขข้อมูลรูปภาพห้องพัก
ผลการลบข้อมูลรูปภาพห้องพัก	รายงานผลการลบข้อมูลรูปภาพห้องพัก	Process 7.3 ลบข้อมูลรูปภาพห้องพัก	เจ้าของหอพัก	ผลการลบข้อมูลรูปภาพห้องพัก
ข้อมูลรูปภาพห้องพัก	ข้อมูลรูปภาพห้องพัก	Process 8.1 เพิ่มข้อมูลรูปภาพห้องพัก	D8 เพิ่มข้อมูลรูปภาพห้องพัก	รหัสรูปภาพ + รูปภาพ
		D8 เพิ่มข้อมูลรูปภาพห้องพัก	Process 8.1 เพิ่มข้อมูลรูปภาพห้องพัก	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		Process 8.2 แก้ไขข้อมูล รูปภาพห้องพัก	D8 เพิ่มข้อมูล รูปภาพห้องพัก	
		D8 เพิ่มข้อมูล รูปภาพห้องพัก	Process 8.2 แก้ไขข้อมูล รูปภาพห้องพัก	
		Process 8.3 ลบ ข้อมูลรูปภาพ ห้องพัก	D8 เพิ่มข้อมูล รูปภาพห้องพัก	
		D8 เพิ่มข้อมูล รูปภาพห้องพัก	Process 8.3 ลบ ข้อมูลรูปภาพ ห้องพัก	
ข้อมูลค่าน้ำ-ค่า ไฟต่อหน่วยที่เพิ่ม	ส่งข้อมูลค่าน้ำ-ค่า ไฟต่อหน่วยที่ ต้องการเพิ่ม	เจ้าของหอพัก	Process 8.1 เพิ่ม ข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วย	รหัสค่าน้ำไฟต่อ หน่วย + ค่าน้ำต่อ หน่วย + ค่าไฟต่อ หน่วย
ข้อมูลค่าน้ำ-ค่า ไฟต่อหน่วยแก้ไข	ส่งข้อมูลค่าน้ำ-ค่า ไฟต่อหน่วยที่ ต้องการแก้ไข	เจ้าของหอพัก	Process 8.2 แก้ไขข้อมูลค่าน้ำ- ค่าไฟต่อหน่วย	หน่วย
รหัสข้อมูลค่าน้ำ- ค่าไฟต่อหน่วย	ส่งรหัสข้อมูลค่า น้ำ-ค่าไฟต่อ หน่วย	เจ้าของหอพัก	Process 8.3 ลบ ข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วย	รหัสค่าน้ำไฟต่อ หน่วย
ผลการเพิ่มข้อ มูลค่าน้ำ-ค่าไฟ ต่อหน่วย	รายงานผลการ เพิ่มข้อมูลค่าน้ำ- ค่าไฟต่อหน่วย	Process 8.1 เพิ่มข้อมูลค่าน้ำ- ค่าไฟต่อหน่วย	เจ้าของหอพัก	ผลการเพิ่มข้อ มูลค่าน้ำ-ค่าไฟต่อ หน่วย

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ผลการแก้ไขข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	รายงานผลการแก้ไขข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 8.2 แก้ไขข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	เจ้าของหอพัก	ผลการแก้ไขข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย
ผลการลบข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	รายงานผลการลบข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 8.3 ลบข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	เจ้าของหอพัก	ผลการลบข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย
ข้อมูลข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	ข้อมูลข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 8.1 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	รหัสค่าน้ำไฟต่อหน่วย + ค่าน้ำต่อหน่วย + ค่าไฟต่อหน่วย
		D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 8.1 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	
		Process 8.2 แก้ไขข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	
		D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 8.2 แก้ไขข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	
		Process 8.3 ลบข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	
		D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 8.3 ลบข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	
		D9 เพิ่มข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	Process 8.3 ลบข้อมูลค่าน้ำ-ค่าไฟต่อหน่วย	



ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ข้อมูลค่าใช้จ่ายที่เพิ่ม	ข้อมูลค่าใช้จ่ายที่ต้องการเพิ่ม	เจ้าของหอพัก	Process 9.1 เพิ่มข้อมูลค่าใช้จ่าย	รหัสค่าใช้จ่าย + สถานะ + หน่วย
ข้อมูลค่าใช้จ่ายที่แก้ไข	ข้อมูลค่าใช้จ่ายที่ต้องการแก้ไข	เจ้าของหอพัก	Process 9.2 แก้ไขข้อมูลค่าใช้จ่าย	น้ำ + หน่วยไฟ + เลขมิเตอร์น้ำ ล่าสุด + เลขมิเตอร์ไฟล่าสุด + ค่าห้อง + ค่าน้ำ + ค่าไฟ + ราคารวม + วันที่ออก ค่าใช้จ่าย + ค่าปรับ+ วันครบกำหนด
รหัสข้อมูลค่าใช้จ่าย	รหัสข้อมูลค่าใช้จ่ายที่ต้องการลบ	เจ้าของหอพัก	Process 9.3 ลบข้อมูลค่าใช้จ่าย	รหัสค่าใช้จ่าย
ข้อมูลค่าใช้จ่ายที่ค้นหา	Keyword ข้อมูลค่าใช้จ่ายที่ต้องการค้นหา	เจ้าของหอพัก	Process 9.4 ค้นหาข้อมูลค่าใช้จ่าย	[รหัสค่าใช้จ่าย   สถานะ ชื่อผู้เช่า  เลขห้องพัก]
		ผู้เช่าห้องพัก	Process 9.4 ค้นหาข้อมูลค่าใช้จ่าย	
ผลการเพิ่มข้อมูลค่าใช้จ่าย	รายงานผลการเพิ่มข้อมูลค่าใช้จ่าย	Process 9.1 เพิ่มข้อมูลค่าใช้จ่าย	เจ้าของหอพัก	ผลการเพิ่มข้อมูลค่าใช้จ่าย
ผลการแก้ไขข้อมูลค่าใช้จ่าย	รายงานผลการแก้ไขข้อมูลค่าใช้จ่าย	Process 9.2 แก้ไขข้อมูลค่าใช้จ่าย	เจ้าของหอพัก	ผลการแก้ไขข้อมูลค่าใช้จ่าย

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
ผลการลบบข้อมูล ค่าใช้จ่าย	รายงานผลการ ลบบข้อมูล ค่าใช้จ่าย	Process 9.3 ลบ ข้อมูลค่าใช้จ่าย	เจ้าของหอพัก	ผลการลบบข้อมูล ค่าใช้จ่าย
รายการข้อมูล ค่าใช้จ่าย	รายการข้อมูล ค่าใช้จ่าย	Process 9.4 ค้นหาข้อมูล ค่าใช้จ่าย	เจ้าของหอพัก	{รหัสค่าใช้จ่าย + สถานะ + หน่วย น้ำ + หน่วยไฟ + เลขมิเตอร์น้ำ ล่าสุด + เลข มิเตอร์ไฟล่าสุด + ค่าห้อง + ค่าน้ำ + ค่าไฟ + ราคารวม + วันที่ออก ค่าใช้จ่าย + ค่าปรับ+ วันครบ กำหนด}
		Process 9.4 ค้นหาข้อมูล ค่าใช้จ่าย	ผู้เช่าห้องพัก	
ข้อมูลค่าใช้จ่าย	ข้อมูลค่าใช้จ่าย	Process 9.1 เพิ่มข้อมูล ค่าใช้จ่าย	D10 เพิ่มข้อมูล ค่าใช้จ่าย	รหัสค่าใช้จ่าย + สถานะ + หน่วย น้ำ + หน่วยไฟ + เลขมิเตอร์น้ำ ล่าสุด + เลข มิเตอร์ไฟล่าสุด + ค่าห้อง + ค่าน้ำ + ค่าไฟ + ราคารวม + วันที่ออก ค่าใช้จ่าย + ค่าปรับ+ วันครบ กำหนด
		D10 เพิ่มข้อมูล ค่าใช้จ่าย	Process 9.1 เพิ่ม ข้อมูลค่าใช้จ่าย	
		Process 9.2 แก้ไขข้อมูล ค่าใช้จ่าย	D10 เพิ่มข้อมูล ค่าใช้จ่าย	
		D10 เพิ่มข้อมูล ค่าใช้จ่าย	Process 9.2 แก้ไขข้อมูล ค่าใช้จ่าย	
		Process 9.3 ลบ ข้อมูลค่าใช้จ่าย	D10 เพิ่มข้อมูล ค่าใช้จ่าย	

ตารางที่ 3.2 Data Flow and Data Structure Description (ต่อ)

Name	Description	Source	Destination	Data Structure
		D10 เพิ่มข้อมูล ค่าใช้จ่าย	Process 9.3 ลบ ข้อมูลค่าใช้จ่าย	
		D10 เพิ่มข้อมูล ค่าใช้จ่าย	Process 9.4 ค้นหาข้อมูล ค่าใช้จ่าย	
ข้อมูลค่าน้ำ-ค่า ไฟต่อหน่วย	ข้อมูลค่าน้ำ-ค่าไฟ ต่อหน่วย	D9 เพิ่มข้อมูลค่า น้ำ-ค่าไฟ ต่อ หน่วย	Process 9.1 เพิ่ม ข้อมูลค่าใช้จ่าย	รหัสค่าน้ำไฟต่อ หน่วย + ค่าน้ำต่อ หน่วย + ค่าไฟต่อ หน่วย
		D9 เพิ่มข้อมูลค่า น้ำ-ค่าไฟ ต่อ หน่วย	Process 9.2 แก้ไขข้อมูล ค่าใช้จ่าย	หน่วย
ข้อมูลการเช่า ห้องพัก	ข้อมูลการเช่า ห้องพัก	D6 เพิ่มข้อมูล การเช่าห้องพัก	Process 9.1 เพิ่ม ข้อมูลค่าใช้จ่าย	รหัสข้อมูลการเช่า ห้องพัก

### 3.3.3 Data Store Description and Data Structure

Data Store คือการนำ เพิ่มข้อมูลมาอธิบายรายละเอียดและแสดงโครงสร้างของแฟ้ม

ตารางที่ 3.3 Data Store Description

ID	Data Store	Description	Data Structure
D1	เพิ่มข้อมูลเจ้าของห้องพัก	เก็บข้อมูลเจ้าของห้องพัก	รหัสเจ้าของห้องพัก + Email + password + tel_no + name + lastname + image
D2	เพิ่มข้อมูลผู้เช่าห้องพัก	เก็บข้อมูลผู้เช่าห้องพัก	User_id + email + password + name + lastname + tel_no + image

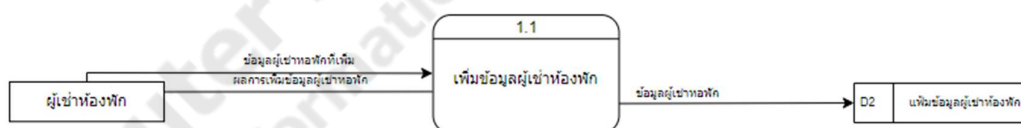
ตารางที่ 3.3 Data Store Description (ต่อ)

ID	Data Store	Description	Data Structure
D3	เพิ่มข้อมูลบัญชีธนาคาร	เก็บข้อมูลบัญชีธนาคาร	รหัสบัญชี + รหัสเจ้าของ หอพัก+เลขบัญชี + ชื่อบัญชี + ธนาคาร
D4	เพิ่มข้อมูลหอพัก	เก็บข้อมูลหอพัก	รหัสหอพัก + รหัสค่าน้ำค่าไฟ ต่อหน่วย+ รายละเอียดหอพัก + ที่อยู่หอพัก + ชื่อหอพัก + Line + facebook + เบอร์ โทร + รูปภาพ + โฟर्म สัญญาเช่า+รหัสเจ้าของหอพัก
D5	เพิ่มข้อมูลห้องพัก	เก็บข้อมูลห้องพัก	รหัสหอพัก + รหัสห้องพัก + รหัสประเภทห้องพัก + รหัส รูปภาพห้องพัก+หมายเลข ห้องพัก + รายละเอียดห้องพัก + สถานะ + ราคาห้องพัก + ราคาประกัน
D6	เพิ่มข้อมูลการเช่าหอพัก	เก็บข้อมูลการเช่าหอพัก	รหัสข้อมูลการเช่าห้องพัก +User_id + รหัสห้องพัก + รหัสหอพัก+สถานะการเช่า + วันเข้า + วันออก + สัญญา เช่า + วันที่ทำสัญญา + วันที่ หมดสัญญา
D7	เพิ่มข้อมูลประเภทห้องพัก	เก็บข้อมูลประเภทห้องพัก	รหัสประเภท + ประเภท
D8	เพิ่มข้อมูลรูปภาพห้องพัก	เก็บข้อมูลรูปภาพ ห้องพัก	รหัสรูปภาพ + รูปภาพ
D9	เพิ่มข้อมูลค่าน้ำ-ค่าไฟ ต่อ หน่วย	เก็บข้อมูลค่าน้ำ-ค่าไฟ ต่อ หน่วย	รหัสค่าน้ำไฟต่อหน่วย + ค่า น้ำต่อหน่วย + ค่าไฟต่อหน่วย

ตารางที่ 3.3 Data Store Description (ต่อ)

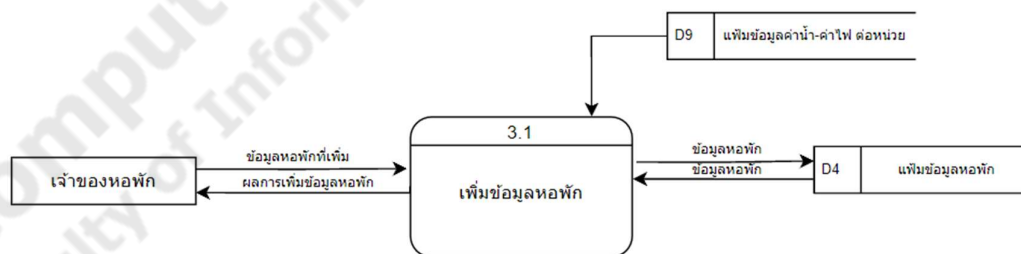
ID	Data Store	Description	Data Structure
D10	เพิ่มข้อมูลค่าใช้จ่าย	เก็บข้อมูลค่าใช้จ่าย	รหัสค่าใช้จ่าย+รหัสข้อมูลการเข้าห้องพัก + สถานะ + หน่วยน้ำที่ใช้ + หน่วยไฟที่ใช้ + เลขมิเตอร์น้ำล่าสุด + เลขมิเตอร์ไฟล่าสุด + ค่าห้อง + ค่าน้ำ + ค่าไฟ+ค่าปรับ + วันที่ออกค่าใช้จ่าย + วันครบกำหนด + ราคารวม
D11	เพิ่มข้อมูลการชำระเงิน	เก็บข้อมูลการชำระเงิน	รหัสข้อมูลการชำระเงิน+รหัสข้อมูลค่าใช้จ่าย+สลิปโอนเงิน+เวลาที่ชำระ+ชื่อธนาคารที่โอน

## 3.3.4 Process Description



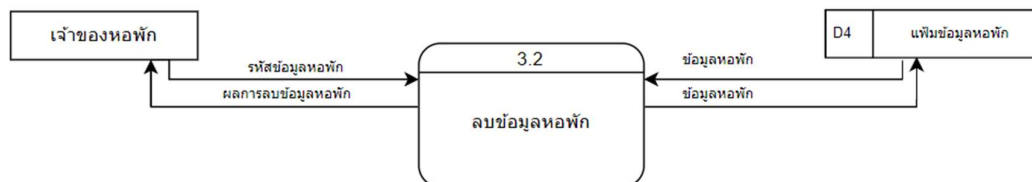
ID	1.1
Name	เพิ่มข้อมูลผู้เช่าห้องพัก
DESCRIPTION	ผู้เช่าห้องพักเพิ่มข้อมูลตนเองลงสู่ระบบฐานข้อมูล (สมัครสมาชิก)
INPUT DATA FLOWS	ข้อมูลผู้เช่าห้องพักที่ต้องการเพิ่ม
OUTPUT DATA FLOWS	ผลการเพิ่มข้อมูลผู้เช่าห้องพัก

PROCESS DESCRIPTION	<p>1. ผู้ใช้กดสมัครสมาชิก</p> <p>2. ผู้ใช้กรอกข้อมูลลงแบบฟอร์ม ดังนี้ อีเมลล์ รหัสผ่าน ชื่อ นามสกุล เบอร์โทร รูปภาพ (มีหรือไม่ก็ได้)</p> <p>3. ระบบทำการตรวจสอบว่ากรอกข้อมูลครบตามที่ระบุในข้อ 2 หรือไม่ ถ้าไม่ครบ ระบบแสดงข้อความ “กรุณากรอกข้อมูลให้ครบ” ถ้าครบไปยังข้อ 4</p> <p>4. ระบบทำการตรวจสอบว่า อีเมลล์ ที่กรอกมีซ้ำในระบบหรือไม่ หากซ้ำให้แสดงข้อความ “อีเมลล์นี้มีผู้ใช้แล้ว” หากไม่ซ้ำ ให้ไปข้อ 5</p> <p>5. ระบบทำการบันทึกข้อมูลลงฐานข้อมูล</p> <p>6. ระบบแสดงข้อความ “สมัครสมาชิกเสร็จสิ้น”</p> <p>7. จบการทำงาน</p>
---------------------	---



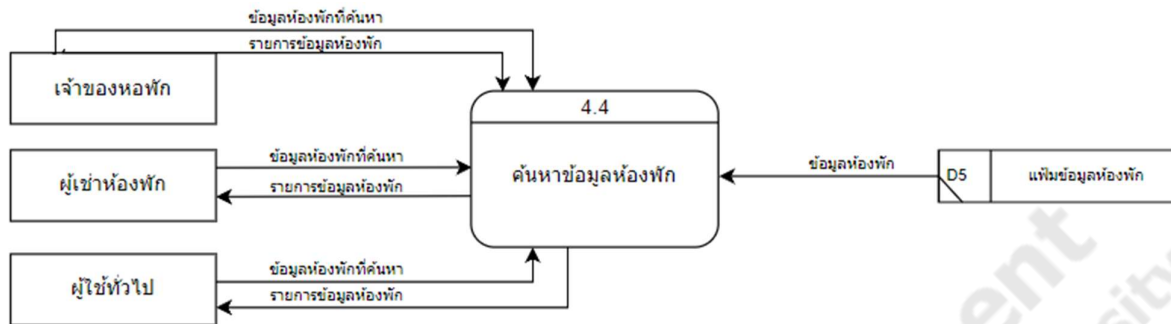
ID	3.1
Name	เพิ่มข้อมูลห้องพัก
DESCRIPTION	เจ้าของห้องพัก เพิ่มข้อมูลห้องพัก

INPUT DATA FLOWS	ข้อมูลหอพักที่ต้องการเพิ่ม
OUTPUT DATA FLOWS	ผลการเพิ่มข้อมูลหอพัก
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1. ผู้ใช้กดปุ่ม เพิ่มหอพัก</li> <li>2. ผู้ใช้กรอกข้อมูลลงใน field ดังนี้ ชื่อหอพัก รายละเอียดหอพัก ที่อยู่หอพัก Line Facebook เบอร์โทร</li> <li>3. ผู้ใช้ทำการอัปโหลดไฟล์ ดังนี้ รูปภาพหอพัก เอกสารฟอร์มสัญญาเช่า</li> <li>4. ผู้ใช้กด ยืนยันการเพิ่ม</li> <li>5. ระบบทำการตรวจสอบข้อมูลว่ากรอกข้อมูลครบ หรือ ไฟล์ที่อัปโหลดเกินขนาดหรือไม่ หากกรอกข้อมูลไม่ครบ ระบบแสดงข้อความ “กรุณากรอกข้อมูลให้ครบ” หากไฟล์ที่อัปโหลดมีขนาดใหญ่เกินไป ระบบแสดงข้อความ “ไฟล์รูปภาพมีขนาดใหญ่เกินไป” หรือ “ไฟล์สัญญาเช่ามีขนาดใหญ่เกินไป” หากข้อมูลถูกต้อง ไปยังข้อ 6</li> <li>6. ระบบทำการบันทึกข้อมูลลงฐานข้อมูล</li> <li>7. ระบบแสดงข้อความ “เพิ่มหอพักเสร็จสิ้น”</li> <li>8. จบการทำงาน</li> </ol>

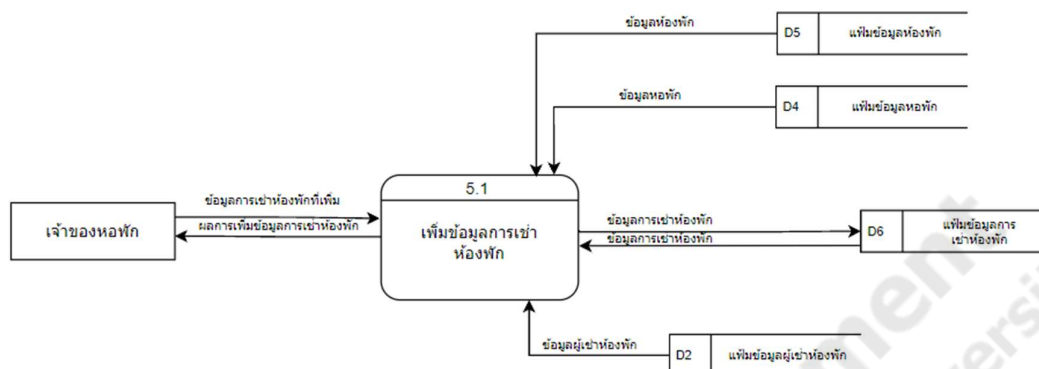


ID	3.2
Name	ลบข้อมูลห้องพัก
DESCRIPTION	เจ้าของห้องพัก ต้องการลบข้อมูลห้องพัก
INPUT DATA FLOWS	รหัสห้องพัก
OUTPUT DATA FLOWS	ผลการลบข้อมูลห้องพัก
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1.ผู้ใช้เลือกห้องพักที่ต้องการลบ โดยระบบจะส่งข้อมูลเป็น รหัสห้องพัก</li> <li>2.ระบบค้นหาข้อมูลห้องพักในฐานข้อมูลจากรหัสห้องพัก</li> <li>3.ระบบแสดงกล่องข้อความ “ยืนยันการลบ”</li> <li>4.ผู้ใช้ทำการยืนยันการลบ</li> <li>หากยืนยันการลบ ไปยังข้อ 5</li> <li>หากยกเลิกการลบ ระบบกลับไปยังหน้าแสดงห้องพัก และไปยังข้อ 7</li> <li>5.ระบบทำการลบข้อมูลห้องพัก</li> <li>6.ระบบแสดงข้อความ “ลบห้องพักเรียบร้อยแล้ว”</li> <li>7.จบการทำงาน</li> </ol>





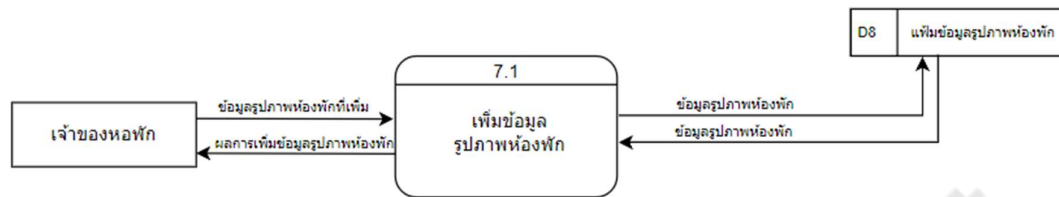
ID	4.4
Name	ค้นหาข้อมูลห้องพัก
DESCRIPTION	เจ้าของห้องพัก ผู้เช่าห้องพัก ผู้ใช้ทั่วไป ต้องการค้นหาข้อมูลห้องพัก
INPUT DATA FLOWS	ข้อมูลห้องพักที่ต้องการค้นหา
OUTPUT DATA FLOWS	รายการห้องพักที่พบ
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1. ผู้ใช้เลือกห้องพักที่ต้องการ</li> <li>2. ผู้ใช้กดปุ่ม รายการห้องพัก</li> <li>3. ผู้ใช้ค้นหาห้องพัก โดยเลือกใช้ keyword ดังนี้ เลขห้องพัก ประเภทห้องพัก สถานะห้องพัก</li> <li>4. ระบบค้นหาข้อมูลในฐานข้อมูล หากพบข้อมูล ระบบแสดงรายการห้องพักที่พบ หากไม่พบข้อมูล ระบบแสดงข้อความ “ไม่พบข้อมูล”</li> <li>5. จบการทำงาน</li> </ol>



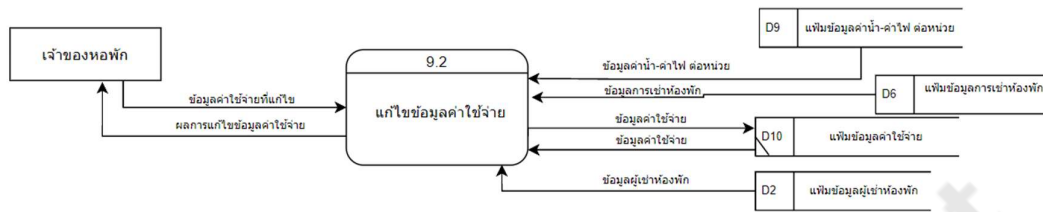
ID	5.1
Name	เพิ่มข้อมูลการเช่าห้องพัก
DESCRIPTION	เจ้าของห้องพักต้องการเพิ่มข้อมูลการเช่าห้องพัก
INPUT DATA FLOWS	ข้อมูลการเช่าห้องพักที่ต้องการเพิ่ม
OUTPUT DATA FLOWS	ผลการเพิ่มข้อมูลการเช่าห้องพัก
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1. ผู้ใช้เลือกห้องพักที่ต้องการเพิ่มข้อมูลผู้เช่า</li> <li>2. ผู้ใช้เลือกวิธีการเพิ่มข้อมูลผู้เช่าดังนี้ <ol style="list-style-type: none"> <li>2.1. เพิ่มจากคำขอที่ผู้เช่าส่งมา</li> <li>2.2. เพิ่มด้วยตนเอง</li> </ol> </li> <li>3. ผู้ใช้กรอก อีเมลผู้เช่า และไปข้อ 3</li> <li>4. ผู้ใช้กดเพิ่มข้อมูลจากรายการคำขอ และไปข้อ 4</li> <li>5. ระบบค้นหาข้อมูลผู้เช่าในฐานข้อมูล</li> <li>6. หากพบข้อมูล ระบบแสดงข้อมูลผู้เช่าที่พบ และไปข้อ 4</li> <li>7. หากไม่พบข้อมูล ระบบแสดงข้อความ “ไม่พบข้อมูล”</li> <li>8. ผู้ใช้ตรวจสอบและยืนยันข้อมูล</li> <li>9. ยืนยันข้อมูล ไปยังข้อ 5</li> <li>10. ยกเลิก ระบบกลับไปยังหน้ารายการห้องพัก และไปข้อ 7</li> <li>11. ระบบบันทึกข้อมูลลงฐานข้อมูล</li> <li>12. ระบบแสดงข้อความ “เพิ่มข้อมูลสำเร็จ”</li> <li>13. จบการทำงาน</li> </ol>



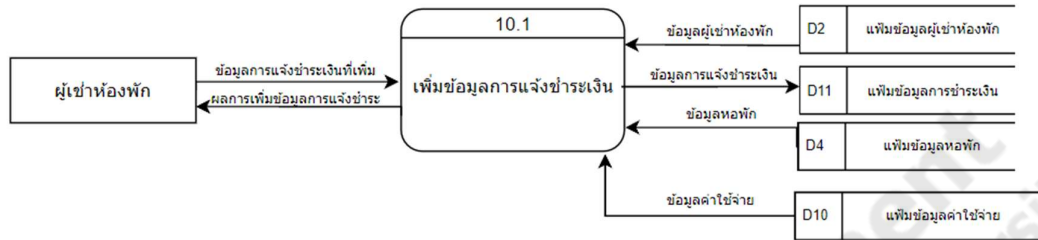
ID	5.3
Name	แก้ไขข้อมูลการเข้าห้องพัก
DESCRIPTION	เจ้าของห้องพักต้องการแก้ไขข้อมูลการเข้าห้องพัก
INPUT DATA FLOWS	ข้อมูลการเข้าห้องพักที่ต้องการแก้ไข
OUTPUT DATA FLOWS	ผลการแก้ไขข้อมูลการเข้าห้องพัก
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1. ผู้ใช้เลือกห้องพักที่ต้องการแก้ไขข้อมูลผู้เช่า</li> <li>2. ระบบแสดงข้อมูลผู้เช่าจากห้องพักที่เลือก</li> <li>3. ผู้ใช้ทำการแก้ไขข้อมูล โดยข้อมูลที่แก้ไขได้มีดังนี้ สถานะการเช่า วันเข้า วันออก สัญญาเช่า วันที่ทำสัญญา วันที่หมดสัญญา</li> <li>4. ผู้ใช้กดยืนยันการแก้ไข</li> <li>5. ระบบทำการตรวจสอบว่ากรอกข้อมูลครบทุก field หรือไม่ หากครบ ไปยังข้อ 6 หากไม่ครบ ระบบแสดงข้อความ “กรุณากรอกข้อมูลให้ครบ”</li> <li>6. ระบบทำการบันทึกข้อมูลลงฐานข้อมูล</li> <li>7. ระบบแสดงข้อความ “แก้ไขข้อมูลเสร็จสิ้น”</li> <li>8. จบการทำงาน</li> </ol>



ID	7.1
Name	เพิ่มข้อมูลรูปภาพห้องพัก
DESCRIPTION	เจ้าของห้องพักต้องการเพิ่มรูปภาพห้องพัก
INPUT DATA FLOWS	รูปภาพห้องพัก
OUTPUT DATA FLOWS	ผลการเพิ่มรูปภาพห้องพัก
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1. ผู้ใช้เลือกห้องพักที่ต้องการเพิ่มรูปภาพ</li> <li>2. ผู้ใช้กดปุ่ม เลือกรูปภาพ เพื่ออัปโหลดรูปภาพ</li> <li>3. ผู้ใช้กด อัปโหลดรูปภาพ</li> <li>4. ระบบตรวจสอบขนาดข้อมูลว่าเกินที่กำหนดหรือไม่ หากเกิน ระบบแสดงข้อความ “ขนาดไฟล์ใหญ่เกินไป” หากไม่เกิน ไปยังข้อ 5</li> <li>5. ระบบแสดงข้อความ “อัปโหลดสำเร็จ”</li> <li>6. จบการทำงาน</li> </ol>

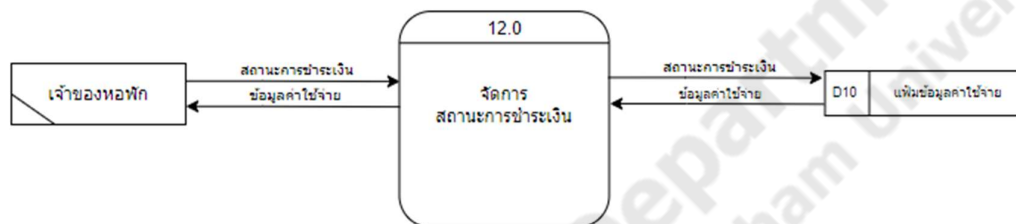


ID	9.2
Name	แก้ไขข้อมูลค่าใช้จ่าย
DESCRIPTION	เจ้าของห้องพักต้องการแก้ไขข้อมูลค่าใช้จ่าย
INPUT DATA FLOWS	ข้อมูลค่าใช้จ่ายที่ต้องการแก้ไข
OUTPUT DATA FLOWS	ผลการแก้ไขข้อมูลค่าใช้จ่าย
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1. ผู้ใช้เลือกรายการค่าใช้จ่าย ดังนี้ <ol style="list-style-type: none"> <li>1.1. เลือกจากรายการค่าใช้จ่ายทั้งหมด ระบบแสดงค่าใช้จ่ายทั้งหมด</li> <li>1.2. เลือกจากรายการห้องพัก ระบบแสดงรายการค่าใช้จ่ายของห้องพักที่เลือก</li> </ol> </li> <li>2. ระบบแสดงข้อมูลค่าใช้จ่ายของรายการที่เลือก</li> <li>3. ผู้ใช้ทำการแก้ไขข้อมูลค่าใช้จ่ายที่ต้องการแก้ไข</li> <li>4. ผู้ใช้กดยืนยันการแก้ไข</li> <li>5. ระบบทำการตรวจสอบข้อมูลว่าครบหรือไม่ หากไม่ครบ ระบบแสดงข้อความ “กรุณากรอกข้อมูลให้ครบ” หากครบ ไปยังข้อ 6</li> <li>6. ระบบบันทึกข้อมูลลงฐานข้อมูล</li> <li>7. ระบบแสดงข้อความ “แก้ไขข้อมูลเสร็จสิ้น”</li> <li>8. จบการทำงาน</li> </ol>



ID	10.1
Name	เพิ่มข้อมูลการแจ้งชำระหนี้
DESCRIPTION	ผู้เช่าห้องพักต้องการแจ้งการชำระหนี้ค่าใช้จ่าย
INPUT DATA FLOWS	ข้อมูลการชำระหนี้
OUTPUT DATA FLOWS	ผลการเพิ่มข้อมูลการชำระหนี้
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1. ผู้ใช้ทำการเลือกรายการค่าใช้จ่ายที่ต้องการชำระ</li> <li>2. ระบบแสดงข้อมูลค่าใช้จ่ายที่เลือก</li> <li>3. ผู้ใช้ทำการกรอกข้อมูล ดังนี้ ชื่อธนาคารที่โอน เวลาที่ชำระหนี้</li> <li>4. ผู้ใช้ทำการอัปโหลดรูปภาพสลิปโอนเงิน</li> <li>5. ผู้ใช้กด ยืนยัน</li> <li>6. ระบบทำการตรวจสอบว่ากรอกข้อมูลครบหรือไม่ หรือขนาดไฟล์ใหญ่เกินไปหรือไม่ หากข้อมูลไม่ครบ ระบบแสดงข้อความ “กรุณากรอกข้อมูลให้ครบ” หากขนาดไฟล์รูปภาพใหญ่เกินไป ระบบแสดงข้อความ “ขนาดไฟล์รูปภาพใหญ่เกินไป”</li> <li>หากข้อมูลครบถูกต้อง ไปข้อ 7</li> <li>7. ระบบบันทึกข้อมูลลงฐานข้อมูล</li> <li>8. สถานะเปลี่ยนเป็น รอตรวจสอบ</li> </ol>

	<p>9.ระบบแสดงข้อความ “เพิ่มข้อมูลเสร็จสิ้น”</p> <p>10.จบการทำงาน</p>
--	--



ID	12.0
Name	จัดการสถานะการชำระเงิน
DESCRIPTION	เจ้าของหอพักต้องการแก้ไขสถานะการชำระเงิน
INPUT DATA FLOWS	สถานะการชำระเงิน
OUTPUT DATA FLOWS	ผลการแก้ไขสถานะการชำระเงิน
PROCESS DESCRIPTION	<ol style="list-style-type: none"> <li>1.ผู้ใช้เลือกรายการ ค่าใช้จ่าย</li> <li>2.ผู้ใช้ทำการเปลี่ยนสถานะ ดังนี้ จ่ายแล้ว</li> <li>รอการตรวจสอบ</li> <li>ข้อมูลผิดพลาด</li> <li>3.ผู้ใช้กด ยืนยัน</li> <li>4.ระบบบันทึกข้อมูลลงฐานข้อมูล</li> <li>5.ระบบแสดงข้อความ “เปลี่ยนสถานะเสร็จสิ้น”</li> <li>6.จบการทำงาน</li> </ol>





ตารางที่ 3.4 เพิ่มข้อมูลเจ้าของหอพัก (Owner)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
owner_id	Int (Autorun)	รหัสเจ้าของหอพัก	0	Primary Key
email	varchar	อีเมล	owner@gmail.com	Not null
password	varchar	รหัสผ่าน	Owner1234	Not null
name	varchar	ชื่อ	Poramin	Not null
lastname	varchar	นามสกุล	Satipchan	Not null
tel_no	varchar	เบอร์โทร	0611111111	unique
image	varchar	รูปภาพ	images1.jpg	Null

ตารางที่ 3.5 เพิ่มข้อมูลผู้เช่าหอพัก (Tenant)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
user_id	Int (Autorun)	รหัสเจ้าของหอพัก	0	Primary Key
email	varchar	อีเมล	tenant@gmail.com	Not null
password	varchar	รหัสผ่าน	tenant1234	Not null
name	varchar	ชื่อ	Thanapat	Not null
lastname	varchar	นามสกุล	Nuanpad	Not null
tel_no	varchar	เบอร์โทร	0622222222	unique
image	varchar	รูปภาพ	images2.jpg	Null

ตารางที่ 3.6 เพิ่มข้อมูลบัญชีธนาคาร (Bank\_Account)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
acc_id	Int (Autorun)	รหัสบัญชีธนาคาร	0	Primary Key
acc_no	varchar	เลขบัญชีธนาคาร	4235647890	unique
name	varchar	ชื่อบัญชี	Thanapat	Not null
bank	varchar	ธนาคาร	กสิกร	Not null
owner	Int	ไอดีเจ้าของบัญชี	0	foreign Key owner(owner_id)

ตารางที่ 3.7 เพิ่มข้อมูลหอพัก (Dormitory)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
dm_id	int(Autorun)	รหัสหอพัก	0	Primary Key
name	varchar	ชื่อหอพัก	The Garden	Not null
detail	varchar	รายละเอียดหอพัก	ห้องพักติดถนน ราคาถูก พร้อม แอร์ ทีวี ตู้เย็น ครบ ไม่ ต้องเสียเพิ่ม....	Null
image	varchar	รูปภาพหอพัก	dorm_img1.jpg	Null
contract	varchar	เอกสารสัญญาเช่าหอพัก	contract1.pdf	Not null
tel_no	varchar	เบอร์โทร	0623123434	Not null
line	varchar	line	garden_line	Null

ตารางที่ 3.7 เพิ่มข้อมูลหอพัก (Dormitory) (ต่อ)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
facebook	varchar	facebook	The garden	Null
address	varchar	address	หมู่ที่ 7 245 ตำบล ขามเรียง อำเภอกันทรวิชัย มหาสารคาม 44150	Not null
owner	int	รหัสเจ้าของหอพัก	0	foreign Key owner(owner_id)

ตารางที่ 3.8 เพิ่มข้อมูลค่าน้ำค่าไฟต่อหน่วย (Unit)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
unit_id	int(Autorun)	รหัสค่าน้ำค่าไฟ		Primary Key
electric_cost	int	ค่าไฟ	8	Not null
water_cost	int	ค่าน้ำ	3	Not null
dorm	int	รหัสหอพัก	0	foreign Key Dormitory (dm_id)

ตารางที่ 3.9 เพิ่มข้อมูลห้องพัก (room)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
room_id	Int (Autorun)	รหัสห้องพัก	0	Primary Key
room_no	varchar	หมายเลขห้องพัก	101	Not null

ตารางที่ 3.9 เพิ่มข้อมูลห้องพัก (room) (ต่อ)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
status	int	สถานะห้องพัก	0	Not null and 0 =ว่าง , 1 =ไม่ว่าง
room_price	int	ราคาห้องพักต่อเดือน	3500	Not null
guarantee_price	int	ราคาประกัน	6000	Not null
detail	varchar	รายละเอียดห้องพัก	มี ตู้เย็น แอร์ ห้องน้ำ	null
image	int	รูปภาพห้องพัก	1	foreign Key Room_image (img_id)
type	int	ประเภทห้องพัก	1	foreign Key Roomtype (type_id)
dorm	int	หอพัก	0	foreign Key Dormitory (dm_id)

ตารางที่ 3.10 เพิ่มข้อมูลประเภทห้องพัก (Roomtype)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
type_id	Int (Autorun)	รหัสประเภทห้องพัก	0	Primary Key
type	varchar	ประเภทห้องพัก	ห้อง VIP	Not null

ตารางที่ 3.11 เพิ่มรูปภาพห้องพัก (Room\_image)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
img_id	Int (Autorun)	รหัสรูปภาพ ห้องพัก	0	Primary Key
image	varchar	รูปภาพห้องพัก	roomimg1.jpg	Not null

ตารางที่ 3.12 เพิ่มข้อมูลการเช่าห้องพัก (Rental\_detail)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
rental_id	Int (Autorun)	รหัสข้อมูลการ เช่า	0	Primary Key
status	int	สถานะผู้เช่า	0	Not null and 0 = ออกไปแล้ว , 1 = เช่าอยู่
entry_time	date	วันที่เข้าพัก	02-05-2021	null
out_time	date	วันที่ออก	02-05-2022	null
contract	varchar	สัญญาเช่า	contract.pdf	null
contract_start	date	วันเริ่มสัญญา	02-05-2021	null
contract_end	date	วันหมดอายุ สัญญา	02-05-2022	null
tenant	int	รหัสผู้เช่า	1	foreign Key Tenant (user_id)

ตารางที่ 3.12 เพิ่มข้อมูลการเช่าห้องพัก (Rental\_detail) (ต่อ)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
room	int	รหัสห้องพัก	1	foreign Key Room (room_id)
dorm	int	รหัสหอพัก	1	foreign Key Dormitory (dm_id)

ตารางที่ 3.13 เพิ่มข้อมูลการชำระเงิน (Bill)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
bill_id	Int (Autorun)	รหัสค่าใช้จ่าย	0	Primary Key
status	Int	สถานะการจ่าย	1	Not null and 0 = ยังไม่จ่าย, 1 = จ่ายแล้ว
w_unit_use	int	หน่วยน้ำที่ใช้	200	Not null
e_unit_use	int	หน่วยไฟที่ใช้	200	Not null
w_meter	int	เลขมิเตอร์น้ำ	12000	Not null
e_meter	int	เลขมิเตอร์ไฟ	12000	Not null
room_price	int	ราคาห้อง	3500	Not null
w_price	int	ค่าน้ำ	200	Not null
e_price	int	ค่าไฟ	1000	Not null

ตารางที่ 3.13 เพิ่มข้อมูลการชำระเงิน (Bill) (ต่อ)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
fine_price	int	ค่าปรับ	100	Not null
sum_price	int	ยอดรวม	4800	Not null
date_due	date	กำหนดวันจ่าย	2-2-2565	Not null
date_bill	date	วันที่ออกบิล	28-1-2565	Not null
rental_detail	int	รหัสข้อมูลการเช่าห้องพัก	0	foreign Key Dormitory (rental_id)

ตารางที่ 3.14 เพิ่มข้อมูลการชำระเงิน (Pay\_Bill)

ชื่อ	ชนิดข้อมูล	คำอธิบาย	ตัวอย่าง	หมายเหตุ
pb_id	Int (Autorun)	รหัสข้อมูลการชำระเงิน	0	Primary Key
slip_img	Int	รูปภาพสลิปโอนเงิน	slip.png	Not null
date_pay	int	วันที่จ่าย	28-1-2565	Not null
bankname	int	ชื่อธนาคารที่จ่าย	กสิกรไทย	Not null
bill_id	int	รหัสข้อมูลค่าใช้จ่าย	1	Not null

## 3.6 การพัฒนาระบบ

### 3.6.1 การเข้าสู่ระบบ

```

39 Login(){
40   console.log('Login');
41   console.log(this.loginMode);
42   let json = {email : this.email, password: this.password, loginMode: this.loginMode};
43   let request = this.http.post('http://127.0.0.1:81/webAPI/login', JSON.stringify(json))
44   .subscribe(response =>{
45     console.log(response);
46     this.data = response;
47     console.log(this.data);
48
49     if(this.loginMode == 'owner'){
50       document.cookie = "id = "+this.data[0].owner_id;
51       document.cookie = "permission = Owner";
52     }else{
53       document.cookie = "id = "+this.data[0].user_id;
54       document.cookie = "permission = Tenant";
55     }
56     document.cookie = "email = "+this.data[0].email;
57     document.cookie = "name = "+this.data[0].name;
58     document.cookie = "lname = "+this.data[0].lastname;
59     this.alert = false;
60     var x = document.cookie;
61     console.log("cookie : "+x);
62     window.location.reload();
63
64   }, error =>{
65     console.log(error);
66     this.alert = true;
67   });
68 }

```

ภาพประกอบที่ 3.15 การเข้าสู่ระบบ

บรรทัดที่ 42-43 : ส่งข้อมูล email , password และโหมดผู้ใช้งานไปยัง Webservice Login โดยส่งข้อมูลเป็น Json

บรรทัดที่ 49-58 : หากได้ Response กลับมาเป็น True จะทำการเช็คโหมดผู้ใช้งาน ว่าเป็นเจ้าของหอพัก(Owner) หรือผู้เช่า(Tenant) และทำการเก็บ cookie id และ permission ตามประเภทผู้ใช้งาน และทำการเก็บ cookie email, name, lastname ของผู้ใช้งาน

บรรทัดที่ 64-66 : หากได้ response กลับมาเป็น error จะแสดงข้อความแจ้งเตือน



```

8  ✓ $app->post('/login', function (Request $request, Response $response, $args) {
9      $body = $request->getBody();
10     $bodyArr = json_decode($body, true);
11
12     $con = $GLOBALS['dbconn'];
13     ✓ if($bodyArr['loginMode'] == 'owner'){
14         $stmt = $con->prepare("select * from owner where email=? ");
15     ✓ }else{
16         $stmt = $con->prepare("select * from tenant where email=? ");
17     }
18
19     $stmt->bind_param("s",$bodyArr['email']);
20     $stmt->execute();
21     $result = $stmt->get_result();
22
23     ✓ if($result->num_rows > 0) {
24         $row = $result->fetch_assoc();
25         $dbPass = $row["password"];
26     ✓ } else {
27         return "";
28     }
29
30     $data = array();
31
32     //password_verify($password, $dbPass)
33     ✓ if($bodyArr['password'] == $dbPass) {
34         array_push($data, $row);
35         $json = json_encode($data);
36         $response->getBody()->write($json);
37         return $response->withHeader('Content-Type', 'application/json');
38     ✓ }else{
39         http_response_code(400);
40         die();
41     }

```

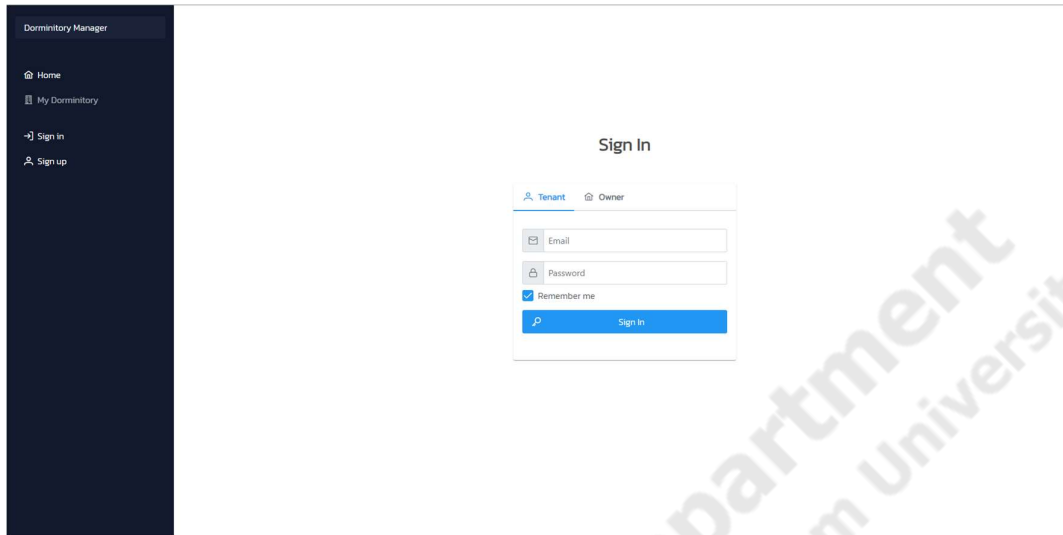
ภาพประกอบที่ 3.16 API การเข้าสู่ระบบ

บรรทัดที่ 9-10 : รับข้อมูล Json จากฝั่งเว็บไซต์มา และทำการ json decode เก็บไว้ใน \$bodyArr

บรรทัดที่ 13-21 : เช็คว่าผู้ใช้ว่าเป็น owner หรือ tenant จากนั้นทำการ query sql โดย select ข้อมูลของผู้ใช้มา

บรรทัดที่ 23-25 : เช็คว่ามีข้อมูลตอบกลับมาหรือไม่ ถ้ามี ให้ดึง password มาเก็บไว้ใน dbPass

บรรทัดที่ 33-41 : ตรวจสอบว่า Password ที่ผู้ใช้กรอกเข้ามากับ Password ที่ดึงมาจาก Database ตรงกันหรือไม่ ถ้าถูกต้อง ให้ push ข้อมูลทั้งหมดของ User และทำการ json\_encode และส่งกลับไปยังเว็บไซต์ แต่ถ้าหาก Password ไม่ตรงกัน จะส่ง error 400 กลับไป



ภาพประกอบที่ 3.17 UI หน้า Login

### 3.6.2 การสมัครสมาชิก

```

52 ✓ CheckEmail(){
53   console.log('checkEmail');
54   let json = {email : this.email};
55   let request = this.http.post('http://127.0.0.1:81/webAPI/check-email', JSON.stringify(json))
56 ✓   .subscribe(response =>{
57     console.log(response);
58
59 ✓   }, error =>{
60     console.log(error);
61     this.emailbox_class = "ng-invalid ng-dirty";
62     this.email_help = true;
63   });
64 }

```

ภาพประกอบที่ 3.18 ตรวจสอบการซ้ำของ Email

บรรทัดที่ 54-55 : ส่ง email ไปยัง webAPI-checkemail เป็น json

บรรทัดที่ 56 : หากได้ response กลับมาเป็น true คือไม่พบ email ซ้ำ จะไม่เกิดอะไรขึ้น

บรรทัดที่ 59-62 : หากได้ response เป็น error จะแสดง กล่องข้อความแจ้งเตือนว่ามี email ซ้ำ

```

44  ✓ $app->post('/check-email', function (Request $request, Response $response, $args) {
45      $body = $request->getBody();
46      $bodyArr = json_decode($body, true);
47
48      $con = $GLOBALS['dbconn'];
49      $stmt = $con->prepare("select * from owner, tenant where owner.email = ? or tenant.email = ?");
50      $stmt->bind_param("ss", $bodyArr['email'], $bodyArr['email']);
51      $stmt->execute();
52      $result = $stmt->get_result();
53
54  ✓   if($result->num_rows > 0){
55       http_response_code(400);
56       die();
57   }
58  ✓   }else{
59       echo "true";
60   }
61   return $response;
62  });

```

ภาพประกอบที่ 3.19 API การตรวจสอบ Email

บรรทัดที่ 45-46 : รับข้อมูลจากเว็บไซต์ และทำการ json\_decode

บรรทัดที่ 49-52 : query sql โดย select ข้อมูลทั้งหมดจาก owner และ tenant โดยที่เลือก select เฉพาะที่มี email เหมือนกับ email ที่ผู้ใช้กรอกเข้ามา

บรรทัดที่ 54-60 : ตรวจสอบว่า ข้อมูลที่ได้จาก Database มีมากกว่า 0 หรือไม่ (พบ email ใน Database) ถ้ามี ให้ส่ง error 400 กลับไป ถ้าไม่มี email จะส่ง True กลับไป

```

103  SignUp(){
104      console.log(this.signup_Mode);
105      let error = this.CheckError();
106      if(error > 0){
107          this.alert = true;
108      }else{
109          this.alert = false;
110
111          let json = {email : this.email, password: this.password, name: this.name, lastname:this.lastname,
112                    tel_no: this.tel_no, signupMode: this.signup_Mode};
113          let request = this.http.post('http://127.0.0.1:81/webAPI/sign-up', JSON.stringify(json))
114              .subscribe(response =>{
115                  console.log(response);
116                  console.log("Success");
117                  //this.Clearform();
118
119                  this.signUpCard = false;
120                  this.successCard = true;
121
122          }, error =>{
123              console.log(error);
124          });
125  }

```

ภาพประกอบที่ 3.20 สมัครสมาชิก

บรรทัดที่ 105-109 : ตรวจสอบว่ามี error หรือไม่ ถ้ามีให้แสดงข้อความแจ้งเตือน

บรรทัดที่ 111-124 : ส่งข้อมูลที่ผู้ใช้กรอกในลงฟอร์มสมัครสมาชิก ไปยัง API sign-up ในรูปแบบ JSON และตรวจสอบ response ที่ส่งกลับมา หากเป็น true จะแสดงข้อความว่าเสร็จสิ้น หากเป็น false จะแสดง error

```

64 ✓ $app->post('/sign-up', function (Request $request, Response $response, $args) {
65     $body = $request->getBody();
66     $bodyArr = json_decode($body, true);
67
68     $con = $GLOBALS['dbconn'];
69
70 ✓     if($bodyArr['signupMode'] == 'owner'){
71
72         $stmt = $con->prepare("insert into owner (email, password, name, lastname, tel_no) values (?,?,,?,?)");
73 ✓     }else{
74         $stmt = $con->prepare("insert into tenant (email, password, name, lastname, tel_no) values (?,?,,?,?)");
75     }
76     $stmt->bind_param("sssss",$bodyArr['email'], $bodyArr['password'], $bodyArr['name'], $bodyArr['lastname'], $bodyArr['tel_no']);
77     $stmt->execute();
78     $result = $stmt->get_result();
79
80 ✓     if($result->row_nums > 0){
81         echo 'success';
82     }
83     return $response;
84 });

```

ภาพประกอบที่ 3.21 API สมัครสมาชิก

บรรทัดที่ 65-66 : รับข้อมูลจากเว็บไซต์ในรูปแบบ JSON และทำการ json decode  
 บรรทัดที่ 70-78 : ตรวจสอบ โหมดการสมัครว่าสมัครมาในแบบ owner หรือ tenant และทำการ insert ข้อมูลลง Database ตามประเภทของผู้ใช้

The image shows a web application interface. On the left is a dark blue sidebar menu with the following items: Home, My Dormitory, Sign in, and Sign up. On the right is a white 'Sign Up' form with the following fields: Email, Password, Confirm Password, Name, Lastname, and Phone Number. Below these fields are two radio buttons for 'Tenant' and 'Owner', and a blue 'Sign Up' button at the bottom.

ภาพประกอบที่ 3.22 UI หน้าสมัครสมาชิก

### 3.6.3 หน้าแสดงรายการหอพัก

```
18   ngOnInit(): void {
19     this.LoadDormitoryData();
20   }
21
22   LoadDormitoryData(){
23     console.log('Load DormitoryData');
24     let json = {name: this.search};
25     let request = this.http.post('http://127.0.0.1:81/webAPI/search-dor', JSON.stringify(json))
26     .subscribe(response =>{
27       this.DormitoryData = response;
28       this.SearchData();
29     }, error =>{
30       //console.log(error);
31     });
32   }
33 }
```

ภาพประกอบที่ 3.23 Source Code การดึงข้อมูลหอพัก

บรรทัดที่ 19 : เมื่อเปิดหน้าเว็บขึ้นมาเรียกใช้ LoadDormitoryData()

บรรทัดที่ 22-30 : เรียกใช้ API เพื่อดึงข้อมูลหอพักมาและนำ response ที่ได้คือข้อมูลหอพักทั้งหมดเก็บไว้ใน DormitoryData

```

8  $app->post('/search-dor', function (Request $request, Response $response, $args) {
9      $body = $request->getBody();
10     $bodyArr = json_decode($body, true);
11     $con = $GLOBALS['dbconn'];
12     $stmt = $con->prepare("SELECT a.*, COUNT(b.room_id) empty
13                             FROM dormitory a
14                             LEFT JOIN room b ON a.dm_id = b.dorm
15                             WHERE b.status IS NULL or b.status = 0
16                             GROUP BY a.dm_id");
17
18     $stmt->execute();
19     $result = $stmt->get_result();
20
21     $data = array();
22     while ($row = $result->fetch_assoc()) {
23         array_push($data, $row);
24     }
25
26     $json = json_encode($data);
27     $response->getBody()->write($json);
28     return $response->withHeader('Content-Type', 'application/json');
29 });

```

ภาพประกอบที่ 3.24 API ดึงข้อมูลหอพัก

บรรทัดที่ 12-19 : query sql โดย Select ข้อมูลหอพักจาก table dormitory

และนับจำนวนห้องพักที่ว่างจาก table room โดยอาศัยการทำ LEFT JOIN

บรรทัดที่ 22-24 : push ข้อมูลที่ได้ลงไป ใน \$data

บรรทัดที่ 26-28 : ส่งข้อมูลที่ได้ (\$data) กลับไปในรูปแบบ JSON

```

35     SearchData(){
36         this.ShowData = [];
37         this.DormitoryData.forEach((items: any) => {
38             if(items.name.toUpperCase().includes(this.search.toUpperCase())){
39                 this.ShowData.push(items);
40             }
41         });
42     }

```

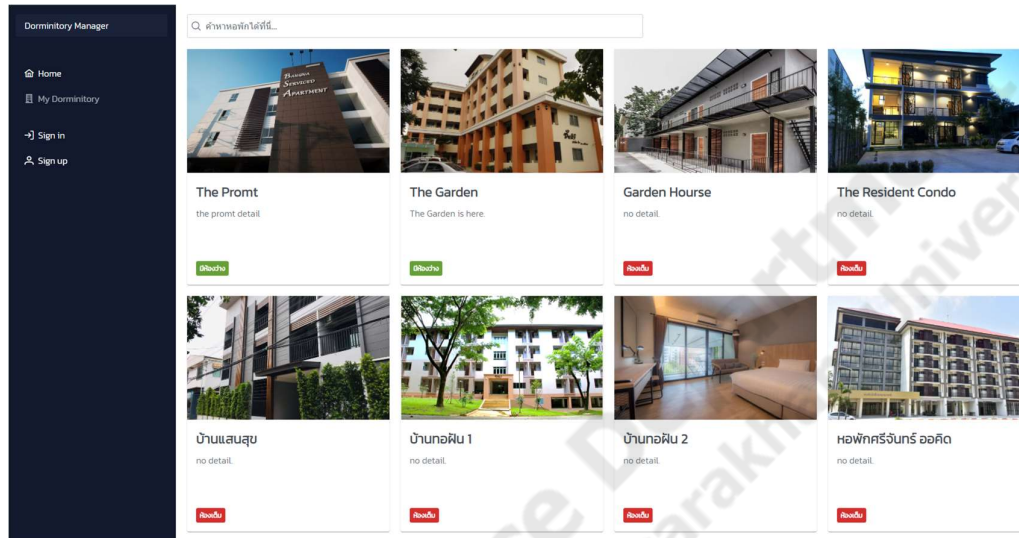
ภาพประกอบที่ 3.25 ค้นหาข้อมูลหอพัก

บรรทัดที่ 36 : ประกาศตัวแปร ShowData เป็น Array

บรรทัดที่ 37-40 : ทำการวนลูปตัวแปร DormitoryData ที่เก็บข้อมูลหอพักทั้งหมด

และนำชื่อของหอพักทั้งหมดมาแปลงเป็นตัวอักษรใหญ่ และนำมาเทียบกับ search

คือชื่อหอพักที่ผู้ใช้กรอกลงบนช่องค้นหาว่าตรงกันหรือไม่ หากตรงกัน จะ push ข้อมูลลงใน ShowData และนำไปแสดงบนเว็บไซต์



ภาพประกอบที่ 3.26 UI หน้าแรก

### 3.6.4 หอพักของฉัน

```

31 $app->post('/get-mydorm-list', function (Request $request, Response $response, $args) {
32     $body = $request->getBody();
33     $bodyArr = json_decode($body, true);
34     $con = $GLOBALS['dbconn'];
35     if ($bodyArr['permis'] == 'Owner') {
36         $stmt = $con->prepare("select * from dormitory where owner = ?");
37     } elseif ($bodyArr['permis'] == 'Tenant') {
38         $stmt = $con->prepare("SELECT dormitory.* FROM dormitory, rental_info WHERE rental_info.tenant = ? AND dormitory.dm_id = rental_info.dorm");
39     }
40
41     $stmt->bind_param("i", $bodyArr['id']);
42     $stmt->execute();
43     $result = $stmt->get_result();
44
45     $data = array();
46     while ($row = $result->fetch_assoc()) {
47         array_push($data, $row);
48     }
49
50     $json = json_encode($data);
51     $response->getBody()->write($json);
52     return $response->withHeader('Content-Type', 'application/json');
53 }
54

```

ภาพประกอบที่ 3.27 API ดึงข้อมูลหอพักของฉัน

บรรทัดที่ 35-36 : ตรวจสอบว่าผู้ใช้เป็น owner หรือ tenant หากเป็น owner จะใช้คำสั่ง sql ที่ select ข้อมูลหอพักโดยตรวจสอบจาก owner id

บรรทัดที่ 37-38 : หากผู้ใช้เป็น Tenant จะใช้คำสั่ง sql ที่ select ข้อมูลห้องพักจาก dormitory ซึ่งข้อมูลของ tenant จะอยู่ใน rental\_info จึงต้องทำการ join ข้อมูลกับ table rental\_info ด้วย

### 3.6.5 เพิ่มห้องพัก (เจ้าของห้องพัก)

```

44     handleUpload(event:any, mode:any) {
45         const file = event.target.files[0];
46         const reader = new FileReader();
47         reader.readAsDataURL(file);
48         reader.onload = () => {
49             if(mode == 'img'){
50                 this.image = this.name+this.id+file.name;
51                 this.img_base64 = reader.result;
52                 console.log(mode);
53             }else{
54                 this.contract = this.name+this.id+file.name;
55                 this.contract_base64 = reader.result;
56                 console.log(mode);
57             }
58         };
59     }

```

ภาพประกอบที่ 3.28 อ่านไฟล์รูปภาพและไฟล์สัญญาเช่า

บรรทัดที่ 45 : ดึงข้อมูลไฟล์ที่ผู้ใช้เลือกจากหน้าเว็บไซต์

บรรทัดที่ 46-47 : สร้าง FileReader ขึ้นมาเพื่ออ่านข้อมูลไฟล์

บรรทัดที่ 48-58 : เมื่ออ่านไฟล์ได้ จะตรวจสอบ mode

ว่าที่อัปโหลดมาเป็นไฟล์รูปภาพหรือไฟล์สัญญาเช่า และทำการดึงชื่อไฟล์เก็บไว้ในตัวแปร image และ contract และทำการเก็บข้อมูลไฟล์ลงใน img\_base64 และ contract\_base64

```

76     Upload() {
77         console.log('Upload');
78         let json = {image: this.image, img_base64: this.img_base64, contract: this.contract, contract_base64: this.contract_base64};
79         let request = this.http.post('http://127.0.0.1:81/webAPI/img-upload', JSON.stringify(json))
80         .subscribe(response =>{
81             console.log(response);
82         }, error =>{
83             //console.log(error);
84         });
85     });
86 }

```

ภาพประกอบที่ 3.29 อัปโหลดไฟล์

บรรทัดที่ 78-79 : ส่งชื่อไฟล์ และ base64 ของไฟล์ไปยัง API upload



```

8 function base64_to_file($base64_string, $output_file){
9     $ifp = fopen($output_file, 'wb');
10    $data = explode(',',$base64_string);
11    if(count($data) == 2){
12        fwrite($ifp, base64_decode($data[1]));
13    }else{
14        fwrite($ifp, base64_decode($data[0]));
15    }
16    fclose($ifp);
17    return $output_file;
18 }

```

ภาพประกอบที่ 3.30 Function แปลง base64 to file

บรรทัดที่ 8 : รับ parameter มา 2 ตัวคือ base64 และ path ที่จะบันทึกไฟล์  
 บรรทัดที่ 10 : split ข้อมูล base64 ออกจากส่วน header  
 บรรทัดที่ 12-17 : ตรวจสอบข้อมูลว่า ข้อมูลที่ส่งมา split แล้วได้กี่ index และทำการ แปลง base64 เป็น file และ save ไฟล์ ด้วยคำสั่ง fwrite

```

20 ~ $app->post('/img-upload', function (Request $request, Response $response, $args) {
21     $json = $request->getBody();
22     $jsonArr = json_decode($json, true);
23
24     if(array_key_exists('image', $jsonArr)){
25         try{
26             base64_to_file($jsonArr['img_base64'], 'D:\Work\angular\ng-test\src\assets\showcase\images\dorm_img\' . $jsonArr['image']);
27             base64_to_file($jsonArr['contract_base64'], 'D:\Work\angular\ng-test\src\assets\contract-form\' . $jsonArr['contract']);
28
29             $response->getBody()->write(json_encode('Success'));
30             return $response->withHeader('Content-Type', 'application/json')
31                 ->withStatus(200);
32         }catch(Exception $error){
33             $response->getBody()->write(json_encode($error));
34             return $response->withHeader('Content-Type', 'application/json')
35                 ->withStatus(400);
36         }
37     }else{
38         $response->getBody()->write(json_encode('No file name.));
39         return $response->withHeader('Content-Type', 'application/json')
40             ->withStatus(400);
41     }
42 });
43

```

ภาพประกอบที่ 3.31 API Upload File

บรรทัดที่ 24 : ตรวจสอบว่าพบข้อมูลที่ส่งมาหรือไม่  
 บรรทัดที่ 26-27 : เรียก function base64\_to\_file โดยส่ง parameter base64 และ ชื่อไฟล์ ไปด้วย  
 บรรทัดที่ 29-31 : ส่ง response กลับไปเป็น Success และ status 200  
 บรรทัดที่ 33-36 : หากเกิด error ส่ง response error กลับไป  
 บรรทัดที่ 39-41 : หากไม่พบชื่อไฟล์ จะส่ง response กลับไปว่า No file name.

```

61 AddDorm(){
62   console.log(this.name,this.detail, this.image, this.contract, this.tel_no, this.line, this.facebook, this.type.name, this.address, this.permiss);
63   this.Upload();
64   console.log("Add DormitoryData");
65   let json = {name: this.name, detail: this.detail, image: this.image, contract: this.contract, tel_no: this.tel_no, line: this.line,
66             facebook: this.facebook, type:this.type.name, address:this.address, permiss: this.permiss, owner: this.id};
67   let request = this.http.post('http://127.0.0.1:81/webAPI/add-mydorm', JSON.stringify(json))
68   .subscribe(response =>{
69     console.log(response);
70   }, error =>{
71     //console.log(error);
72   });
73 }
74 }

```

### ภาพประกอบที่ 3.32 เพิ่มห้องพัก

บรรทัดที่ 63 : เรียก function Upload

บรรทัดที่ 65-66 : ส่งข้อมูลที่ผู้ใช้กรอกไปยัง API เพิ่มห้องพัก ในรูปแบบ JSON

```

56 $app->post('/add-mydorm', function (Request $request, Response $response, $args) {
57   $body = $request->getBody();
58   $bodyArr = json_decode($body, true);
59   $con = $GLOBALS['dbconn'];
60   if ($bodyArr['permiss'] == 'Owner') {
61     $stmt = $con->prepare("INSERT INTO dormitory (name, detail, image, contract, tel_no, line, facebook, address, type, owner) values (?, ?, ?, ?, ?, ?, ?, ?, ?);");
62   }
63   // elseif ($bodyArr['permiss'] == 'Tenant') {
64   //   $stmt = $con->prepare("SELECT dormitory.* FROM dormitory, rental_info WHERE rental_info.tenant = ? AND dormitory.dm_id = rental_info.dorm");
65   // }
66   // }
67
68   $stmt->bind_param("sssssssssl", $bodyArr['name'], $bodyArr['detail'], $bodyArr['image'], $bodyArr['contract'], $bodyArr['tel_no'], $bodyArr['line'],
69                   $bodyArr['facebook'], $bodyArr['address'], $bodyArr['type'], $bodyArr['owner']);
70   $stmt->execute();
71
72   $result = $stmt->get_result();
73
74   $json = json_encode($result, "true");
75   $response->getBody()->write($json);
76   return $response->withHeader('content-type', 'application/json');
77 });

```

### ภาพประกอบที่ 3.33 API เพิ่มห้องพัก

บรรทัดที่ 61-70 : ตรวจสอบว่า ผู้ใช้เป็น Owner และ ทำการ query sql ที่ Insert ข้อมูลลงใน table dormitory

บรรทัดที่ 72-76 : รับข้อมูลที่ได้กลับมาจากการ query และ ส่งข้อมูลกลับไปในรูปแบบ JSON

เพิ่มสถานที่

ชื่อสถานที่  ประเภทสถานที่

รายละเอียด

ที่อยู่

รูปภาพ

เลขการพอร์ทัลบนหน้า

แอดเรสติดต่อ

Line

Facebook

ภาพประกอบที่ 3.34 UI หน้าเพิ่มหอพัก

The Prompt

Map

Map showing location of The Prompt and surrounding areas.

ภาพประกอบที่ 3.35 UI หน้ารายละเอียดหอพัก

```

36 LoadDormitoryData(){
37   console.log('Load DormitoryData dorm');
38   let json = {id: this.dorm, permis: this.permis};
39   let request = this.http.post('http://127.0.0.1:81/webAPI/get-dorm-info', JSON.stringify(json))
40   .subscribe(response =>{
41     this.DormitoryData = response;
42     console.log(this.DormitoryData);
43   }, error =>{
44     console.log(error);
45   });
46 }

```

ภาพประกอบที่ 3.36 ดึงข้อมูลหอพักและรายละเอียด

บรรทัดที่ 38-44 : ส่งข้อมูล ไอดีหอพัก ไปยัง API get-dorm-info เพื่อดึงข้อมูลหอพัก และนำ response ที่ได้กลับมาเก็บไว้ในตัวแปร DormitoryData

```

79 $app->post('/get-dorm-info', function (Request $request, Response $response, $args) {
80   $body = $request->getBody();
81   $bodyArr = json_decode($body, true);
82   $con = $GLOBALS['dbconn'];
83   $stmt = $con->prepare("
84     SELECT dormitory.* , COUNT(room.status) AS empty
85     FROM room
86     INNER JOIN dormitory ON room.dorm = dormitory.dm_id
87     WHERE room.status = 0 AND dormitory.dm_id = ?
88   ");
89   $stmt->bind_param("i", $bodyArr['id']);
90   $stmt->execute();
91   $result = $stmt->get_result();
92
93
94   $data = array();
95   while ($row = $result->fetch_assoc()) {
96     array_push($data, $row);
97   }
98
99   $json = json_encode($data);
100  $response->getBody()->write($json);
101  return $response->withHeader('Content-Type', 'application/json');
102 });

```

ภาพประกอบที่ 3.37 API get ข้อมูลและรายละเอียดหอพัก

บรรทัดที่ 80-81 : รับข้อมูล Json จากเว็บไซต์และทำการ Json\_decode

บรรทัดที่ 83-91 : query sql โดย select ข้อมูลหอพักจาก table dormitory

และนับจำนวนห้องที่ว่างจาก table room โดยอาศัยการ join table และ select เฉพาะ id หอพักที่เลือก