



เว็บแอปพลิเคชันหารูมเมทด้วย K-NN

Web Application Roommate Finder with K-NN

นายจิตตพงษ์ จงใจ  
นายธีระภัทร ชมเชย

ปริญญา妮พน์เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิគฤตศาสตรบัณฑิต  
สาขาวิชาบริการคอมพิวเตอร์ คณะวิศวกรรมศาสตร์  
มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา เชียงใหม่  
ปีการศึกษา 2567

## เว็บแอปพลิเคชันหารูปเมทด้วย K-NN

นายจิตตพงษ์ จงใจ  
นายธีระภัทร ชมเชย

ปริญญา妮พนร์เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิគរรรมศาสตรบัณฑิต<sup>๑</sup>  
สาขาวิชาบริการคอมพิวเตอร์ คณะวิគរรรมศาสตร์  
มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา เชียงใหม่  
ปีการศึกษา 2567

WEB APPLICATION ROOMMATE FINDER WITH K-NN

MR. JITTAPONG JONGJAI

MR. TEERAPAT CHOMCHOEI

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE BACHELOR DEGREE OF ENGINEERING  
DEPARTMENT OF COMPUTER ENGINEERING FACULTY OF ENGINEERING  
RAJAMANGLA UNIVERSITY OF TECHNOLOGY LANNA  
ACADEMIC YEAR 2024

ปริญญาบัตรเรื่อง เว็บแอปพลิเคชันหารูปเมทด้วย K-NN  
ชื่อนักศึกษา นายจิตต์พงษ์ ใจ รหัส 65543206007-8  
นายธีระภัทร ชมเชย รหัส 65543206019-3  
หลักสูตร วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา สาขาวิชาวิศวกรรมคอมพิวเตอร์  
คณะ วิศวกรรมศาสตร์  
อาจารย์ที่ปรึกษา อาจารย์ธนิต เกตุแก้ว  
ปีการศึกษา 2567

คณะกรรมการคณบดีคณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา อนุมัติให้ปริญญานิพนธ์นี้เป็น ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

รองคณบดีคณะวิศวกรรมศาสตร์

(ผู้ช่วยศาสตราจารย์กฤชดา ยิ่งขยัน)

คณะกรรมการสอบโครงการงานวิศวกรรมคอมพิวเตอร์

ประธานกรรมการ

(อาจารย์ธนิต เกตุแก้ว)

กรรมการ

(ผู้ช่วยศาสตราจารย์อนันท์ ทับเกิด)

กรรมการ

(ผู้ช่วยศาสตราจารย์ขวัญชัย เอื้อวิริyanugul)

ปริญญา呢พนรเรื่อง	เว็บแอปพลิเคชันหารูมเมทด้วย K-NN
ชื่อนักศึกษา	นายจิตตพงษ์ ใจ
อาจารย์ที่ปรึกษา	นายธีระภัทร ชมเชย
หลักสูตร	อาจารย์ธนิต เกตุแก้ว
สาขาวิชา	วิศวกรรมศาสตรบัณฑิต
ปีการศึกษา	วิศวกรรมคอมพิวเตอร์
	2567

### บทคัดย่อ

โครงการเว็บแอปพลิเคชันหารูมเมทด้วย K-NN เกิดขึ้นจากปัญหาที่นักศึกษาไม่สามารถหารูมเมท (เพื่อร่วมหอพัก) ที่เข้ากับตัวเองได้ดี เนื่องจากในการหารูมเมทจำเป็นจะต้องใช้ข้อมูลในหลาย ๆ ด้าน มาประกอบการตัดสินใจ ปัญหาที่เกิดขึ้นส่วนใหญ่เกิดจากการแลกเปลี่ยนข้อมูลที่ไม่ชัดเจน ไม่รอบคอบจึงอาจเกิดปัญหาการเข้าใจผิดและการเข้ากันไม่ได้ในภายหลัง

ผู้จัดทำโครงการจึงจัดทำระบบเว็บแอปพลิเคชันหารูมเมทด้วยหลักการทำงานอัลกอริทึม K-NN เพื่อ ช่วยลดความผิดพลาดในการแลกเปลี่ยนข้อมูล สามารถสร้างໂປຣໄຟລ໌ ระบุลักษณะนิสัยส่วนบุคคล ระบุความชอบเพื่อนำไปวิเคราะห์ มีระบบการคัดกรองตามความต้องการ แนะนำรูมเมทที่เหมาะสม จัดอันดับผลลัพธ์เป็นตัวเลือกตามลำดับจากมากไปน้อย ให้ผู้ใช้งานสามารถตัดสินใจได้ง่ายขึ้น แก้ไข ปัญหาการเลือกรูมเมทที่อาจเข้ากันไม่ได้และสามารถได้รับข้อมูลอย่างครบถ้วนและรวดเร็ว

ดังนั้นเว็บไซต์หารูมเมทจึงถูกพัฒนาและออกแบบโดยเฉพาะสำหรับการทำให้คนร่วมห้อง เพื่อ เป็นเครื่องมือในการแก้ไขปัญหาและอำนวยความสะดวกในการค้นหารูมเมทที่ตรงตามความต้องการ ทำการเก็บข้อมูลขั้นต้นที่เกี่ยวข้อง เพื่อนำมาคำนวณในระบบประเมินความเข้ากันได้ เป็นเทคนิคและ อัลกอริทึมการเรียนรู้ของเครื่องในการหาเพื่อนบ้านที่ใกล้ที่สุด นำผลลัพธ์ที่ได้ออกมาเป็นเปอร์เซ็นต์ ความเข้ากันได้ของผู้ใช้แต่ละบุคคล โดยจะแนะนำความเข้ากันได้มากสุดอันดับต้น ๆ ให้แก่ผู้ใช้ให้ สามารถเลือกดูได้ตามความสนใจ เพิ่มโอกาสการหารูมเมทที่เข้ากันได้ดียิ่งขึ้นและรวดเร็วยิ่งขึ้น

**คำสำคัญ:** รูมเมท, ข้อมูล, ความเข้ากันได้, ความผิดพลาด, ลักษณะนิสัยส่วนบุคคล, คัดกรอง, จัด อันดับ

<b>Project Title</b>	Web Application Roommate Finder With K-NN
<b>Students</b>	Mr. Jittapong Jongjai Mr. Teerapat Chomchoei
<b>Project Advisor</b>	Mr. Thanit Keatkaew
<b>Curriculum</b>	Bachelor of Engineering
<b>Major Field</b>	Computer Engineering
<b>Academic Year</b>	2024

## ABSTRACT

The “Web Application Roommate Finder With K-NN” project was created because students could not find a roommate who was compatible with them. This is because finding a roommate requires information from various aspects to decide. Most of the problems are caused by the exchange of unclear and careless information, which may lead to misunderstandings and incompatibility later.

Therefore, the K-NN roommate-finding web application system was created to help reduce errors in exchanging information. It can create profiles, specify personal characteristics, specify preferences for analysis, has a filtering system based on needs, recommends suitable roommates, and ranks the results as options from most to least, allowing users to make decisions more easily. It solves the problem of choosing a roommate who may not be compatible and can receive complete and fast information.

Therefore, the roommate website is developed and designed specifically for finding roommates. It is a tool to solve problems and facilitate the search for roommates that meet the requirements. It collects the relevant preliminary data to calculate in the compatibility assessment system. It is a technique and machine learning algorithm to find the nearest neighbors. The results are the percentage of compatibility of each user. It will recommend the top compatibility for users to choose from according to their interests, increasing the chance of finding a compatible roommate better and faster.

**Keywords:** Roommates, Information, Compatibility, Faults, Personal Characteristics, Filtering, Ranking

## กิตติกรรมประกาศ

ปริญญาอุดมศึกษานี้สำเร็จลุล่วงได้ด้วยความช่วยเหลือและการสนับสนุนจากหลายฝ่าย ผู้จัดทำ  
ขอกราบขอบพระคุณอาจารย์ธนิต เกตุแก้ว อารามย์ที่ปรึกษาปริญญาอุดมศึกษา ที่ได้ให้คำแนะนำ  
อันทรงคุณค่า และข้อคิดเห็นที่เป็นประโยชน์อย่างยิ่งต่อการดำเนินโครงการโดยตลอด

ขอขอบพระคุณ มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา สำหรับทุนสนับสนุนการทำโครงการ  
บางส่วน ซึ่งมีบทบาทสำคัญในการช่วยให้การดำเนินงานครั้งนี้เป็นไปได้อย่างราบรื่น

นอกจากนี้ ขอขอบคุณสมาชิกโครงการ เพื่อน ๆ และอาจารย์ทุกท่าน ที่มีส่วนร่วมในการสืบค้น  
ข้อมูล และเปลี่ยนความรู้ความคิด และให้กำลังใจอย่างต่อเนื่องตลอดระยะเวลาการทำโครงการ

สุดท้ายนี้ ผู้วิจัยขอกราบขอบพระคุณ บิดามารดา ผู้ให้การสนับสนุนทั้งในด้านการเงินและกำลังใจ  
ด้วยความรักและความเสียสละอันยิ่งใหญ่ จนทำให้ผู้วิจัยสามารถสำเร็จการศึกษาในครั้งนี้ได้สำเร็จ

คณะผู้จัดทำ

นายจิตตพงษ์ จงใจ

นายธีระภัทร ชมเชย

## สารบัญ

	หน้า
บทคัดย่อภาษาไทย	๑
บทคัดย่อภาษาอังกฤษ	๒
กิตติกรรมประกาศ	๓
สารบัญรูป	๔
สารบัญตาราง	๕
รายการสัญลักษณ์	๖
ประมวลคำศัพท์และคำย่อ	๗
<b>บทที่ 1 บทนำ</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขต	2
1.4 ขั้นตอนการดำเนินงาน	4
1.5 ประโยชน์ที่คาดว่าจะได้รับ	7
<b>บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง</b>	<b>8</b>
2.1 ทฤษฎีและหลักการ	8
2.2 งานวิจัยที่เกี่ยวข้อง	23
<b>บทที่ 3 วิธีการออกแบบ</b>	<b>28</b>
3.1 ขั้นตอนการออกแบบ	28
3.2 แผนภาพรวมของระบบ	31
3.3 ขั้นตอนการออกแบบฐานข้อมูล	37
3.4 ขั้นตอนการออกแบบโครงสร้างเว็บไซต์	55
3.5 ขั้นตอนการออกแบบอัลกอริทึม K-NN	62

## สารบัญ (ต่อ)

	หน้า
<b>บทที่ 4 ผลการทดลอง</b>	<b>70</b>
4.1 บทนำ	70
4.2 ผลการทำงานตามขอบเขต	72
4.3 การคำนวณทางอัลกอริทึม K-NN	83
4.4 การประเมินแบบสอบถาม	96
4.5 สรุปผลการประเมินแบบแผนภูมิ	97
4.6 สรุปผลการประเมินแบบตาราง	102
<b>บทที่ 5 สรุปและข้อเสนอแนะ</b>	<b>105</b>
5.1 สรุปตามขอบเขต	105
5.2 ปัญหาและอุปสรรค	107
5.3 ข้อเสนอแนะ	108
5.4 สรุปผลการทดลอง	108
<b>บรรณานุกรม</b>	<b>109</b>
<b>ภาคผนวก</b>	<b>112</b>
ภาคผนวก ก โค้ดโปรแกรม	112
ภาคผนวก ข คู่มือการใช้งาน	114
ภาคผนวก ค แบบประเมินความพึงพอใจ	134
ภาคผนวก ง ประวัติผู้เขียนโครงการ	138
<b>ประวัติผู้เขียนโครงการ</b>	<b>139</b>

## สารบัญรูป

	หน้า
รูปที่ 2.1 Web Application	8
รูปที่ 2.2 React	9
รูปที่ 2.3 Material UI	11
รูปที่ 2.4 NodeJS	12
รูปที่ 2.5 MySQL	13
รูปที่ 2.6 Express API	15
รูปที่ 2.7 K-Nearest Neighbors (K-NN)	16
รูปที่ 2.8 Euclidean Distance	17
รูปที่ 2.9 สูตรการคำนวณของ 2D Euclidean Distance	17
รูปที่ 2.10 ตัวอย่างการคำนวณ 2D Euclidean Distance	18
รูปที่ 2.11 ผลลัพธ์การคำนวณ 2D Euclidean Distance	18
รูปที่ 2.12 Vercel	19
รูปที่ 2.13 Railway	20
รูปที่ 2.14 Cloudinary	21
รูปที่ 2.15 Nodemailer	22
รูปที่ 3.1 System Diagram	31
รูปที่ 3.2 Use Case การเข้าสู่ระบบ	32
รูปที่ 3.3 Use Case การสมัครสมาชิก	33
รูปที่ 3.4 Use Case การใช้งานของผู้ใช้งาน	34
รูปที่ 3.5 Use Case การเปลี่ยนรหัสผ่าน	35
รูปที่ 3.6 Use Case การแก้ไขข้อมูลส่วนตัว	35
รูปที่ 3.7 Use Case การกรองข้อมูล	36
รูปที่ 3.8 Use Case การใช้งานของผู้ดูแลระบบ	36
รูปที่ 3.9 Context Diagram (DFD Level 0)	37
รูปที่ 3.10 Data Flow Diagram (DFD Level 1) Process 1-4	39
รูปที่ 3.11 Data Flow Diagram (DFD Level 1) Process 5-8	40
รูปที่ 3.12 Data Flow Diagram (DFD Level 2) Process 1	41

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.13 Data Flow Diagram (DFD Level 2) Process 2	42
รูปที่ 3.14 Data Flow Diagram (DFD Level 2) Process 3	43
รูปที่ 3.15 Data Flow Diagram (DFD Level 2) Process 4	43
รูปที่ 3.16 Data Flow Diagram (DFD Level 2) Process 5	44
รูปที่ 3.17 Data Flow Diagram (DFD Level 2) Process 6	45
รูปที่ 3.18 Data Flow Diagram (DFD Level 2) Process 7	45
รูปที่ 3.19 Data Flow Diagram (DFD Level 2) Process 8	46
รูปที่ 3.20 การออกแบบ ER Diagram ของเว็บแอปพลิเคชันหารูปเมทด้วย K-NN	47
รูปที่ 3.21 การออกแบบ Prototype หน้าแรก	55
รูปที่ 3.22 การออกแบบ Prototype หน้าเข้าสู่ระบบ	55
รูปที่ 3.23 การออกแบบ Prototype หน้าสมัครสมาชิก	56
รูปที่ 3.24 การออกแบบ Prototype หน้าค้นหารูปเมท	57
รูปที่ 3.25 การออกแบบ Prototype หน้าต่างตัวกรอง	57
รูปที่ 3.26 การออกแบบ Prototype หน้าต่างໂປຣີຟຣູມເມທ	58
รูปที่ 3.27 การออกแบบ Prototype หน้าถูกใจ	58
รูปที่ 3.28 การออกแบบ Prototype หน้าจับคู่สำเร็จ	59
รูปที่ 3.29 การออกแบบ Prototype หน้าต่างเมนู	59
รูปที่ 3.30 การออกแบบ Prototype หน้าแก้ไขໂປຣີ	60
รูปที่ 3.31 การออกแบบ Prototype หน้าแจ้งปัญหาการใช้งาน	60
รูปที่ 3.32 การออกแบบ Prototype หน้าແດບອົບຜູ້ດຸແລະບົບ	61
รูปที่ 4.1 หน้าแรก	72
รูปที่ 4.2 หน้าสมัครใช้งาน	72
รูปที่ 4.3 หน้าเข้าสู่ระบบ	73
รูปที่ 4.4 หน้าລັບບຸນຊື່ຜູ້ໃຊ້	73
รูปที่ 4.5 หน้าປຶກສິນ	74
รูปที่ 4.6 หน้าການຈົບປັດຂໍ້ມູນສ່ວນຕົວ	74
รูปที่ 4.7 หน้าເລືອກລັກຍົນນິສັຍສ່ວນຕົວ	75

## สารบัญรูป (ต่อ)

	หน้า
รูปที่ 4.8 หน้าแก้ไขโปรแกรม	75
รูปที่ 4.9 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้	76
รูปที่ 4.10 รายละเอียดโปรแกรมผู้ใช้	77
รูปที่ 4.11 การเน้นข้อมูลลักษณะนิสัยที่ตรงกันในหน้าโปรแกรม	77
รูปที่ 4.12 หน้าต่างระบบตัวกรองการค้นหา	78
รูปที่ 4.13 หน้าคนที่ถูกใจ	78
รูปที่ 4.14 หน้าคนที่จับคู่สำเร็จ	79
รูปที่ 4.15 ระบบแจ้งเตือนการจับคู่สำเร็จ	79
รูปที่ 4.16 การแจ้งปัญหาเกี่ยวกับผู้ใช้	80
รูปที่ 4.17 การแจ้งปัญหาเกี่ยวกับระบบการใช้งาน	80
รูปที่ 4.18 การแนะนำเกี่ยวกับระบบ	81
รูปที่ 4.19 หน้าให้คะแนนความพึงพอใจ	81
รูปที่ 4.20 หน้าแดชบอร์ดของผู้ดูแลระบบ	82
รูปที่ 4.21 การคำนวณทางอัลกอริทึม K-NN	83
รูปที่ 4.22 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบข้อที่ 1	97
รูปที่ 4.23 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบข้อที่ 2	97
รูปที่ 4.24 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบข้อที่ 3	98
รูปที่ 4.25 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบข้อที่ 4	98
รูปที่ 4.26 แผนภูมิการประเมินประสิทธิภาพในการทำงานข้อที่ 1	99
รูปที่ 4.27 แผนภูมิการประเมินประสิทธิภาพในการทำงานข้อที่ 2	99
รูปที่ 4.28 แผนภูมิการประเมินประสิทธิภาพในการทำงานข้อที่ 3	100
รูปที่ 4.29 แผนภูมิการประเมินด้านประโยชน์และการนำไปใช้ข้อที่ 1	100
รูปที่ 4.30 แผนภูมิการประเมินด้านประโยชน์และการนำไปใช้ข้อที่ 2	101
รูปที่ 4.31 แผนภูมิการประเมินด้านประโยชน์และการนำไปใช้ข้อที่ 3	101

## สารบัญตาราง

	หน้า
ตารางที่ 1.1 แผนการดำเนินงาน	5
ตารางที่ 1.2 ผลงานในแต่ละช่วงเวลา	7
ตารางที่ 3.1 ความต้องการเชิงธุรกิจ (Business Requirement)	28
ตารางที่ 3.2 ความต้องการของระบบ (System Requirement)	29
ตารางที่ 3.3 ตัวอย่างตารางข้อมูลของ users	49
ตารางที่ 3.4 ตัวอย่างตารางข้อมูลของ personality_information	50
ตารางที่ 3.5 ตัวอย่างตารางข้อมูลของ personality_traits	51
ตารางที่ 3.6 ตัวอย่างตารางข้อมูลของ likes	52
ตารางที่ 3.7 ตัวอย่างตารางข้อมูลของ matches	52
ตารางที่ 3.8 ตัวอย่างตารางข้อมูลของ reports	53
ตารางที่ 3.9 ตัวอย่างตารางข้อมูลของ suggestions	54
ตารางที่ 3.10 ตัวอย่างตารางข้อมูลของ app_reviews	54
ตารางที่ 3.11 การแปลงข้อมูลโดยใช้การเข้ารหัสแบบ Ordinal Encoding	67
ตารางที่ 4.1 ตัวอย่างข้อมูลลักษณะนิสัยผู้ใช้จากฐานข้อมูล	83
ตารางที่ 4.2 ตัวอย่างข้อมูลเฉพาะผู้ใช้ปัจจุบัน	83
ตารางที่ 4.3 ตัวอย่างข้อมูลในระบบจริง	86
ตารางที่ 4.4 เทียบข้อมูลเป็นตัวแปรที่ใช้งานจริง	87
ตารางที่ 4.5 การกำหนดค่าตัวแปรในการคำนวณ	88
ตารางที่ 4.6 การกำหนดค่าต่อวันน้ำหนักของปัจจัยแต่ละด้าน	89
ตารางที่ 4.7 คำนวณระยะห่างสูงสุดที่เป็นไปได้	91
ตารางที่ 4.8 คำนวณระยะห่างระหว่างผู้ใช้ James กับ JamesCompare	93
ตารางที่ 4.9 คำนวณระยะห่างระหว่างผู้ใช้ James กับ Alpha	94
ตารางที่ 4.10 การประเมินความพึงพอใจในการใช้งานระบบ	103
ตารางที่ 4.11 การประเมินประสิทธิภาพในการทำงาน	103
ตารางที่ 4.12 การประเมินด้านประโยชน์และการนำไปใช้	104

## รายการสัญลักษณ์

$\bar{x}$  = ค่าเฉลี่ย

$\sum x$  = ผลรวมของคะแนน

$N$  = จำนวนคนหรือจำนวนข้อมูลทั้งหมด

## ประมวลคำศัพท์และคำย่อ

UX/UI	=	User Experience / User Interface
K-NN	=	K-Nearest Neighbor
API	=	Application Program Interface
JSX	=	JavaScript XML
HTML	=	Hypertext Markup Language
MUI	=	Material User Interface
DOM	=	Document Object Model
OTP	=	One Time Password
MySQL	=	My Structured Query Language
DFD	=	Data Flow Diagram
ER Diagram	=	Entity Relationship Diagram
RTL	=	Right-to-left
ARM	=	Advanced RISC Machine
RISC	=	Reduced Instruction Set Computing
CPU	=	Central Processing Unit
I/O	=	Input / Output
NPM	=	Node Package Manager
HTTP	=	Hypertext Transfer Protocol
RDBMS	=	Relational Database Management System
REST	=	Representational State Transfer
SRS	=	Software Requirement Specification
PK	=	Primary Key
FK	=	Foreign key

## บทที่ 1

### บทนำ

#### 1.1 ความเป็นมาและความสำคัญ

ในปัจจุบัน การพักอาศัยในหอพักได้รับความนิยมอย่างแพร่หลายในกลุ่มนักศึกษา หลายคนมองหาโอกาสในการหารูมเมทด้วยเหตุผลต่าง ๆ เช่น การพบปะเพื่อนใหม่ การช่วยเหลือกัน การสร้างประสบการณ์การใช้ชีวิตร่วมกัน และที่สำคัญคือ การแบ่งเบาภาระค่าใช้จ่าย ซึ่งถือเป็นเรื่องสำคัญสำหรับนักศึกษา อย่างไรก็ตาม การหารูมเมทที่เข้ากันได้ดีนั้นไม่ใช่เรื่องง่าย เนื่องจากความแตกต่างด้านการใช้ชีวิต กิจวัตรประจำวัน ทัศนคติ หรือการแบ่งปันค่าใช้จ่าย ซึ่งอาจนำไปสู่ความขัดแย้ง และส่งผลกระทบต่อความสัมพันธ์และการใช้ชีวิตรประจำวัน ดังนั้น การเลือกและอยู่ร่วมกับรูมเมทจึงควรพิจารณาอย่างรอบคอบ เพื่อให้การใช้ชีวิตในหอพักเป็นไปอย่างราบรื่นและมีความสุข

การหารูมเมทเป็นทางเลือกหนึ่งที่ช่วยลดภาระค่าใช้จ่ายได้มาก แต่ก็มาพร้อมกับความท้าทาย เช่น ความไม่เข้าใจกันในการใช้ชีวิตประจำวัน การประ公示หารูมเมทผ่านกลุ่ม Facebook หรือแพลตฟอร์มที่ไม่ได้ออกแบบมาเฉพาะ การแลกเปลี่ยนข้อมูลระหว่างกันในการณ์ที่มีผู้สนใจจำนวนมาก ซึ่งจำเป็นต้องเปรียบเทียบข้อมูลต่าง ๆ ด้วยตนเอง กระบวนการนี้อาจยุ่งยาก ทำให้ข้อมูลที่สำคัญบางอย่างไม่ชัดเจนหรือตกหล่น และอาจนำไปสู่ความคิดเห็นที่ไม่ตรงกันเมื่อเริ่มพักอาศัยร่วมกัน

ปัญหาหลักที่พบเกี่ยวกับการค้นหารูมเมทสามารถแยกออกเป็น 2 ประเด็น ได้แก่

1.1.1 ปัญหาการหารูมที่ไม่เหมาะสมสมกัน

1.1.2 ปัญหาการหารูมเมทได้ยาก

โครงงานนี้จะเน้นไปที่การแนะนำรูมเมทที่เหมาะสมกับผู้ใช้งานโดยใช้พื้นฐานจากข้อมูลที่ผู้ใช้แต่ละคนป้อนเข้ามาด้วยหลักการทำงานด้าน AI และให้ผู้ใช้เป็นคนตัดสินใจในการเลือกด้วยตนเอง โดยปัญหาการหารูมเมทที่ไม่เหมาะสมกันเป็นปัญหาที่พบได้บ่อยในกลุ่มผู้ที่พักอาศัยร่วมกัน ความไม่เข้ากันนี้อาจเกิดจากหลายปัจจัย เช่น ลักษณะนิสัยที่แตกต่างกัน ความต้องการส่วนตัวที่ไม่สอดคล้องกัน หรือความคาดหวังที่ไม่ตรงกัน ทำให้การใช้ชีวิตร่วมกันเกิดความตึงเครียด และอาจส่งผลให้เกิดการย้ายออกหรือการขัดแย้งในภายหลัง

เพื่อแก้ปัญหานี้ โครงงานได้ใช้อัลกอริทึมที่ช่วยประเมินความเข้ากันได้ของแต่ละบุคคล โดยใช้ข้อมูลที่สำคัญ เช่น ลักษณะนิสัยส่วนบุคคล กิจวัตรประจำวัน และความสนใจต่าง ๆ เพื่อให้การจับคู่รูมเมทเป็นไปอย่างเหมาะสมและราบรื่น ลดความเสี่ยงในการเกิดปัญหาขัดแย้งในอนาคต และช่วยสร้างสภาพแวดล้อมการอยู่อาศัยที่ดีสำหรับทุกคนที่เกี่ยวข้อง

## 1.2 วัตถุประสงค์

- 1.2.1 เพื่อพัฒนาระบบจับคู่รูมเมทที่เหมาะสม
- 1.2.2 เพื่อลดความยุ่งยากในการค้นหารูมเมท เกิดความสะดวกรวดเร็วในการค้นหารูมเมท
- 1.2.3 เพื่อสร้างและพัฒนาระบบเฉพาะทางสำหรับการค้นหารูมเมท

## 1.3 ขอบเขต

- 1.3.1 ระบบสมัครสมาชิกและลงชื่อใช้งาน
  - 1.3.1.1 สามารถสมัครสมาชิกด้วย Email และรหัสผ่านได้
  - 1.3.1.2 สามารถลงชื่อเข้าใช้งานระบบด้วย Email และรหัสผ่านได้
  - 1.3.1.3 สามารถลบบัญชีได้
  - 1.3.1.4 สามารถเปลี่ยนรหัสผ่านได้
- 1.3.2 จัดเก็บข้อมูลผู้ใช้
  - 1.3.2.1 สามารถจัดเก็บข้อมูลส่วนตัวของผู้ใช้ได้
    - 1) เก็บข้อมูลที่สามารถระบุตัวตน เช่น ชื่อ อายุ เพศ และรูปภาพ
    - 2) เก็บข้อมูลเกี่ยวกับ การศึกษา หอพักและyanพานะ
  - 1.3.2.2 สามารถจัดเก็บข้อมูลลักษณะนิสัยผู้ใช้ได้
    - 1) ช่วงเวลาอน
    - 2) ความสะอาด
    - 3) การใช้เสียง
    - 4) การสูบบุหรี่
    - 5) การใช้งานเครื่องปรับอากาศ
    - 6) การเที่ยวกลางคืน
  - 1.3.2.3 สามารถแก้ไขรายละเอียดข้อมูลในໂປຣໄຟລ์ของผู้ใช้ได้
- 1.3.3 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้
  - 1.3.3.1 สามารถนำข้อมูลผู้ใช้มาเปรียบเทียบได้
  - 1.3.3.2 สามารถคำนวณรูปแบบร้อยละความใกล้เคียงของข้อมูลได้
  - 1.3.3.3 สามารถแสดงผลผู้ใช้ที่เข้ากันได้ตามลำดับจากมากไปน้อยได้
- 1.3.4 รายละเอียดໂປຣໄຟລ์ผู้ใช้
  - 1.3.4.1 แสดงรูปໂປຣໄຟລ์ผู้ใช้ได้
  - 1.3.4.2 แสดงรายละเอียดลักษณะนิสัยผู้ใช้ได้
  - 1.3.4.3 สามารถเน้นข้อมูลในส่วนที่ตรงกันให้เด่นชัดขึ้นได้

- 1.3.5 ระบบกรองการค้นหาข้อมูล
  - 1.3.5.1 สามารถกรองการค้นหาด้วยตำแหน่งของผู้ใช้ได้
  - 1.3.5.2 สามารถกรองการค้นหาด้วยลักษณะนิสัยตามต้องการได้
- 1.3.6 ระบบการจับคู่
  - 1.3.6.1 สามารถกดถูกใจผู้ใช้รายอื่นได้
  - 1.3.6.2 แสดงผู้ใช้ที่ได้กดถูกใจไว้ได้
  - 1.3.6.3 สามารถถอนการกดถูกใจได้
  - 1.3.6.4 แสดงรูปเมทัฟ์จับคู่สำเร็จเมื่อทั้งสองฝ่ายกดถูกใจทั้งคู่
- 1.3.7 ระบบแจ้งเตือน
  - 1.3.7.1 สามารถแจ้งเตือนให้ผู้ใช้เมื่อจับคู่สำเร็จ
- 1.3.8 ระบบแจ้งปัญหา
  - 1.3.8.1 สามารถแจ้งปัญหาผู้ใช้ที่ไม่พึงประสงค์ได้
  - 1.3.8.2 สามารถแจ้งปัญหาเกี่ยวกับระบบการใช้งานได้
  - 1.3.8.3 สามารถแนะนำแก่ผู้ใช้ระบบได้
- 1.3.9 ระบบให้คะแนนความพึงพอใจ
  - 1.3.9.1 สามารถให้คะแนนความพึงพอใจระบบการแนะนำรูปเมทในหน้าหลัก
  - 1.3.9.2 สามารถให้คะแนนความพึงพอใจต่อเว็บไซต์ได้
- 1.3.10 ระบบผู้ดูแลระบบ
  - 1.3.10.1 สามารถรับรายงานจากผู้ใช้ได้
  - 1.3.10.2 สามารถรับผู้ใช้ที่ไม่พึงประสงค์ได้
  - 1.3.10.3 สามารถปรับสถานะบัญชีผู้ใช้ได้ (เปิดหรือระงับบัญชี)
- 1.3.11 การประเมินผล
  - 1.3.11.1 เชิงปริมาณ
    - 1) คะแนนความพึงพอใจระบบการแนะนำรูปเมทเฉลี่ยมากกว่า 70%
    - 2) คะแนนความพึงพอใจการแสดงผลข้อมูลบุคคลเฉลี่ยมากกว่า 70%
    - 3) เว็บแอพพลิเคชันทำงานได้ตามขอบเขตมากกว่า 80%
  - 1.3.11.2 เชิงคุณภาพ
    - 1) ด้านความถูกต้องและเหมาะสมของเนื้อหาบนเว็บไซต์ โดยทดสอบใช้งานสร้างบัญชีตลอดจนถึงขั้นตอนการจับคู่อย่างน้อย 10 ครั้ง พบร่วมกับการทำงานถูกต้องไม่น้อยกว่า 80%

## 1.4 ขั้นตอนการดำเนินงาน

### 1.4.1 ขั้นตอนการดำเนินงานโครงงานวิศวกรรม

1.4.1.1 คิดหัวข้อโครงงานจากปัญหาที่พบในการหารูปเมทและเขียน Business Requirement เพื่อนำมากำหนดขอบเขตโครงงาน

1.4.1.2 ศึกษาค้นคว้ารวบรวมข้อมูลรายงานที่เกี่ยวข้องจากแหล่งข้อมูลทั้งภายใน (Thai) และภายนอกประเทศไทย (English) เพื่อนำมาอ้างอิงถึงหลักการและการวิเคราะห์ข้อมูลเพื่อนำไปปรับใช้

1.4.1.3 ศึกษาค้นคว้ารวบรวมข้อมูลเทคโนโลยีที่ใช้ทำโครงงาน

- 1) JavaScript
- 2) React Framework (Library)
- 3) Material UI Framework
- 4) ExpressAPI
- 5) NodeJS
- 6) K-NN (K-Nearest Neighbor)

1.4.1.4 วิเคราะห์และออกแบบระบบ

1) ศึกษาขั้นตอนและวิธีการค้นหารูปเมท จากกลุ่มตัวอย่างที่หารูปเมทสำเร็จ  
 2) รวบรวมข้อมูลที่จำเป็นต่อการวิเคราะห์ เช่น ลักษณะนิสัยส่วนตัว  
 3) วิเคราะห์ความต้องการของระบบ (Business Requirement) ทำความเข้าใจสาเหตุและเรื่องราวก่อนจะมาเป็นความต้องการของระบบ  
 4) ร่างขอบเขตการใช้งานและฟังก์ชันต่าง ๆ จากความต้องการของระบบเพื่อนำมาแก้ไขปัญหาการหารูปเมทที่อาจไม่เหมาะสมกันและเพิ่มความสะดวก รวดเร็วในการแลกเปลี่ยนข้อมูล

1.4.1.5 Front-End

- 1) ออกแบบรูปแบบการแสดงผลของหน้า UI
- 2) พัฒนาระบบ Front-End โดยใช้ React ร่วมกับ Material UI

1.4.1.6 Back-End

- 1) พัฒนาระบบ Back-End จัดการข้อมูลตามเงื่อนไข
- 2) สร้าง API โดยใช้ ExpressAPI สำหรับเรียกใช้ข้อมูลและให้บริการข้อมูลระบบ
- 3) เขียนโค้ดเชื่อมต่อฐานข้อมูล ทดสอบการ Query

#### 1.4.1.7 ทดลองและแก้ไข Web Application

1) ทดสอบความถูกต้องของเว็บไซต์ การทำงานระบบในแต่ละส่วนย่อย เพื่อให้แน่ใจว่าแต่ละส่วนทำงานได้ถูกต้อง

2) ทดสอบคุณภาพของเว็บไซต์ทดสอบประสิทธิภาพการใช้งานของระบบ ทดสอบ UI เหมาะสมต่อการใช้งานหรือไม่

3) ทดสอบเพื่อค้นหาข้อผิดพลาด

1.4.1.8 สรุปผลการใช้งานระบบและบันทึกลงในเอกสารรายงาน

1.4.1.9 จัดรูปเล่มเอกสารรายงาน

1.4.1.10 เตรียมความพร้อมสำหรับการนำเสนอโครงการ

1.4.1.11 ส่งเอกสารฉบับสมบูรณ์

#### 1.4.2 ตารางแผนการดำเนินงาน

ตารางที่ 1.1 แผนการดำเนินงาน

กิจกรรม	เดือน / ปี 67 - 68									
	ม.ย. 67	ก.ค. 67	ส.ค. 67	ก.ย. 67	ต.ค. 67	พ.ย. 67	ธ.ค. 67	ม.ค. 68	ก.พ. 68	มี.ค. 68
คิดหัวข้อโครงการเพื่อนำเสนออาจารย์ที่ปรึกษา	←	→								
ศึกษาค้นคว้ารวบรวมข้อมูลรายงานที่เกี่ยวข้อง			←	→						
ศึกษาค้นคว้ารวบรวมข้อมูลเทคโนโลยีที่ใช้ทำโครงการ			←	→						
วิเคราะห์และออกแบบระบบ										
ศึกษารูปแบบขั้นตอนและวิธีการค้นหารูปเมท				←	→					
รวบรวมข้อมูลที่จำเป็นต่อการวิเคราะห์ผล			←	→						
วิเคราะห์ความต้องการของระบบ BR			←	→						
ขอบเขตการใช้งานและฟังก์ชันต่าง ๆ			←	→						

### ตารางที่ 1.1 แผนการดำเนินงาน (ต่อ)

### ตารางที่ 1.2 ผลงานในแต่ละช่วงเวลา

ปีที่	เดือนที่	ผลงานที่คาดว่าจะสำเร็จ
1	1	- คิดหัวข้อโครงงานเพื่อนำเสนออาจารย์ที่ปรึกษา
	2 - 3	- ศึกษาค้นคว้ารวบรวมข้อมูลรายงานที่เกี่ยวข้อง - ศึกษาค้นคว้ารวบรวมข้อมูลเทคโนโลยีที่ใช้ทำโครงงาน
	4 - 5	วิเคราะห์และออกแบบระบบ - ศึกษารูปแบบขั้นตอนและวิธีการค้นหารูปเมท - รวบรวมข้อมูลที่จำเป็นต่อการวิเคราะห์ผล - วิเคราะห์ความต้องการของระบบ BR - ขอบเขตการใช้งานและฟังก์ชันต่าง ๆ
	6 - 8	Front-End - ออกแบบรูปแบบการแสดงผลของหน้า UI - เขียนโค้ดเพื่อแสดงผล Front-End Back-End - เขียนโค้ดระบบ Back-End จัดการข้อมูลตามเงื่อนไข - สร้าง API โดยใช้ ExpressAPI สำหรับเรียกใช้ข้อมูลและให้บริการข้อมูลระบบ
	9	ทดลองและแก้ไข Web Application - ทดสอบความถูกต้องของเว็บไซต์ - ทดสอบคุณภาพของเว็บไซต์ - ทดสอบเพื่อค้นหาข้อผิดพลาด
	10	- สรุปผลการใช้งานระบบและบันทึกลงในเอกสารรายงาน - เตรียมความพร้อมสำหรับการนำเสนอโครงงาน - ส่งเอกสารฉบับสมบูรณ์
	1 - 10	- จัดรูปเล่มเอกสารรายงาน

### 1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1.5.1 ช่วยให้นักศึกษาค้นหารูปเมทที่เข้ากันได้ดีขึ้น ลดความซ้ำและยังในชีวิตประจำวัน
- 1.5.2 ทำให้การค้นหารูปเมทเป็นไปอย่างรวดเร็วและมีประสิทธิภาพ
- 1.5.3 ช่วยลดภาระค่าใช้จ่ายในการอุดอาศัยในหอพัก
- 1.5.4 ได้สร้างแพลตฟอร์มที่ออกแบบมาเฉพาะสำหรับการหารูปเมท โดยใช้เทคโนโลยีอัลกอริทึมในการคำนวณความเข้ากันได้

## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

#### 2.1 ทฤษฎีและหลักการ

##### 2.1.1 Web Application

เป็นแอปพลิเคชันประเภทหนึ่งที่เขียนขึ้นเพื่อใช้งาน โดยสามารถแสดงผลผ่าน Browser หรือเว็บบราวเซอร์ได้โดยไม่ต้องติดตั้งแอปฯ นั่นหมายความว่าสามารถใช้งานจาก Browser ได้โดยตรง จึงทำให้ Web App นั้นใช้เพียงแค่อุปกรณ์ที่เชื่อมต่ออินเทอร์เน็ตและเปิด Browser ก็สามารถเริ่มใช้งานได้ทันทีไม่จำเป็นต้องลงโปรแกรมอะไรให้ยุ่งยาก ทำให้กินทรัพยากรเครื่องของเราค่อนข้างต่ำและสามารถเปิดใช้งานได้ไว โดยปัจจุบัน Web App นั้นก็ได้รับความนิยมเป็นอย่างมาก เลยก็ได้เช่นเดียวกัน ไม่ต้องติดตั้งโปรแกรมให้วุ่นวายเหมือนโปรแกรมอื่น ๆ อีกทั้งยังถูกปรับปรุงให้ Web App มีการทำงานได้รวดเร็วกว่าการเปิด Application แบบปกติ ส่งผลให้หน้าตา UI เป็นมิตรและเข้าใจง่ายด้วยกระบวนการ UX Design ซึ่งอาจต้องอาศัยความรู้เรื่องการออกแบบ UX UI ควบคู่ไปด้วย



รูปที่ 2.1 Web Application  
(ที่มา: [https://favpng.com/png\\_search/web-application](https://favpng.com/png_search/web-application))

###### 2.1.1.1 ความแตกต่างระหว่าง Web Application กับ Website

Web Application ทำหน้าที่คล้ายกับเว็บไซต์ แต่จะสามารถเป็นแอปพลิเคชันได้ด้วย คือเน้นให้ผู้คนเข้ามาใช้งานมากกว่าดู เช่นเว็บแอปสำหรับคิดเลข เว็บแอปสำหรับจับเวลาเว็บแอปสำหรับแปลงภาษา โดยส่วนมากแล้วจะมีความสะอาด รวดเร็ว และสนับสนุนมากกว่า เว็บไซต์ปกติ เนื่องจากเน้นใช้งานในเรื่องใดเรื่องหนึ่งเป็นหลัก

Website คือ หน้าเพจที่ถูกจัดทำขึ้นเพื่อนำเสนอข้อมูล สารสนเทศ และมีการเชื่อมโยงข้อมูลไปยังเว็บเพจย่อย ๆ ต่าง ๆ ตามแต่รูปแบบของเว็บไซต์นั้นได้กำหนดและตั้งค่าไว้โดยเว็บไซต์นี้เน้นให้ผู้คนเข้ามาดูเป็นหลัก

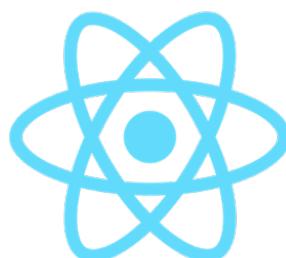
### 2.1.1.2 ข้อดีของ Web Application

- 1) เหมาะกับองค์กรขนาดเล็ก เพราะมีค่าใช้จ่ายน้อย
  - 2) เชื่อมต่อกับเว็บแอปหรือบริการออนไลน์ ๆ ได้ง่าย
  - 3) ไม่ต้องติดตั้ง ดูแล โปรแกรมให้ยุ่งยาก
  - 4) ง่ายต่อการใช้เรียกใช้งาน และ ไม่ซ้ำซ้อน
  - 5) ไม่ต้องมีบุคลากรด้านเทคนิคเป็นของตัวเอง เพราะผู้ให้บริการดูแลเซิร์ฟเวอร์และการบำรุงรักษาเองทั้งหมด
  - 6) สามารถขยายระบบให้รองรับผู้ใช้งานและข้อมูลที่เพิ่มขึ้นได้อย่างง่ายดาย
- ### 2.1.1.3 ข้อเสียของ Web Application
- 1) ไม่เหมาะสมสำหรับโปรแกรมที่ออกแบบมาเพื่อใช้งานกับข้อมูลส่วนบุคคล
  - 2) เสี่ยงต่อการถูกโจกรกรรมข้อมูล
  - 3) เว็บแอปพลิเคชันบางตัวอาจไม่รองรับอุปกรณ์หรือระบบปฏิบัติการบางรุ่น
  - 4) ต้องมีพื้นที่จัดเก็บไฟล์ที่ค่อนข้างใหญ่และการประมวลผลที่ช้า
  - 5) รูป ร่างหน้าตา และการใช้งานมีได้จำกัด

### 2.1.2 React

React คือไลบรารี JavaScript ที่ถูกมองว่าเป็นตัวช่วยให้สามารถสร้าง UI (User Interface หรือองค์ประกอบของเว็บที่เชื่อมต่อกับผู้ใช้งานโดยตรง) ได้แม่นยำและรวดเร็วมากยิ่งขึ้น และส่งผลให้การแสดงผลมีความเป็นระบบมากขึ้นไปพร้อม ๆ กัน สิ่งนี้ถูกนำมาใช้กันอย่างแพร่หลายในการเขียนโค้ดสำหรับเว็บไซต์และแอปพลิเคชันมีประโยชน์กับคนที่ทำงานในฐานะ Front-end

React ได้รับการออกแบบมาเพื่อแก้ปัญหาการทำงานของระบบให้เกิดการแสดงผลอย่างเป็นพลวัต คือสามารถอัปเดตการเปลี่ยนแปลงได้ทันทีเมื่อเกิดคำสั่งใหม่บนระบบ หรือก็คือ React ช่วยให้นักพัฒนาอัปเดตเนื้อหาขององค์ประกอบที่ต้องเปลี่ยนการแสดงผลในหน้าจอได้ทันที โดยที่ไม่ต้องทำการโหลดซ้ำใหม่ทั้งหน้าให้เสียเวลา



รูปที่ 2.2 React

(ที่มา: <https://www.iconfinder.com/icons/1174949>)

### 2.1.2.1 องค์ประกอบของ React

1) Component: โค้ดที่แบ่งออกเป็นบล็อกเล็ก ๆ หลายอัน คอมโพเนนต์ อันหนึ่งจะแทนที่ชิ้นส่วนของปุ่มหรือองค์ประกอบ UI ของหน้าเว็บ ซึ่งคอมโพเนนต์เหล่านี้สามารถ นำมาใช้ซ้ำได้ตลอดการสร้าง ระบบ คอมโพเนนต์จะทำให้ส่วนต่าง ๆ ของหน้าเว็บทำงานอย่างเป็น อิสระต่อกัน

2) JSX เป็นส่วนขยายสำหรับการเขียนโค้ดภาษา JavaScript ทำให้นักพัฒนา สามารถเขียนโค้ดที่เหมือน HTML ภายในคอมโพเนนต์ แต่เบื้องหลังจะถูกแปลงเป็นภาษา JavaScript ได้เอง

3) Props ใช้อธิบายวิธีการส่งผ่านข้อมูลจากคอมโพเนนต์หลักไปยัง คอมโพเนนต์ย่อย เพื่อการแสดงผลบางประการที่แตกต่างกันไปในคอมโพเนนต์หลักชิ้นเดียวกันโดยไม่ ต้องไปเขียนโค้ดชิ้นใหม่ทั้งหมด ทำให้สามารถใช้งานคอมโพเนนต์ได้อย่างหลากหลายมากขึ้น เช่น ปุ่มกดที่มีดีไซน์ UI เมื่อกlicken ก็เกิดจากคอมโพเนนต์อันเดียวกัน แต่ Text บนปุ่มในแต่ละหน้าเว็บอาจ ไม่เหมือนกัน

### 2.1.2.2 ข้อดีของ React

1) React ช่วยให้การจัดการสถานะของ Component มีประสิทธิภาพ

2) สามารถสร้างคอมโพเนนต์และนำมาใช้ใหม่ได้เสมอ ซึ่งนี้จะทำให้โค้ดมี ความเป็นระเบียบมากขึ้นแก้ไขได้ง่ายขึ้น และช่วยลดระยะเวลาในการประกอบ UI ของหน้าเว็บได้

3) React มี Virtual DOM ที่ช่วยลดการทำงานกับ DOM จริง ทำให้มี ประสิทธิภาพการแสดงผล UI ที่ดี

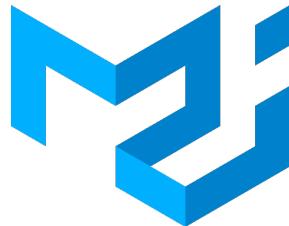
### 2.1.2.3 ข้อเสียของ React

1) React อาจมีความซับซ้อนสำหรับผู้เริ่มต้น โดยเฉพาะผู้ที่ไม่เคยใช้ Component-Based Architecture มา ก่อน

2) การอัปเดตไลบรารีที่เกิดขึ้นบ่อยจนเกินไป

3) การจัดการสถานะที่ซับซ้อนอาจทำให้โค้ดซับซ้อนขึ้น

### 2.1.3 Material UI



รูปที่ 2.3 Material UI  
(ที่มา: <https://iconduck.com/icons/27659>)

MUI (Material UI) คือ ไลบรารี UI สำหรับ React ที่สร้างโดย Google ซึ่งมีชุด Components และ Tools ที่ใช้ในการสร้างเว็บแอปพลิเคชันในแบบของ Material Design ที่เป็นมาตรฐานออกแบบ สำหรับผลิตภัณฑ์ด้าน UX/UI โดย Material UI มีความสามารถในการปรับแต่งใช้งานได้อย่างยืดหยุ่น และเป็นที่นิยมในการพัฒนาเว็บแอปพลิเคชันในวงกว้าง

#### 2.1.3.1 ความแตกต่างของ Material UI จาก Library อื่น ๆ

1) Material UI ใช้หลักแนวคิดของ Material Design ซึ่งเป็นมาตรฐานออกแบบจาก Google ที่ให้ความสำคัญกับความเรียบง่าย ความใส่ใจต่อรายละเอียด และประสิทธิภาพในการใช้งาน

2) ถูกออกแบบมาใช้งานกับ React และ การใช้งาน MUI กับ React ช่วยให้ state management เป็นไปอย่างมีประสิทธิภาพ

3) มีความยืดหยุ่นและสามารถปรับแต่งได้ง่ายตามความต้องการของผู้ใช้งาน ทำให้สามารถสร้างเว็บแอปพลิเคชันที่แตกต่างกันได้อย่างง่ายดาย

4) มาพร้อมกับชุดเครื่องมือที่มีความครอบคลุม ที่ช่วยให้การพัฒนาเว็บแอปพลิเคชันเป็นไปอย่างมีประสิทธิภาพและเรียบง่าย

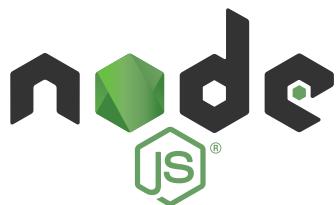
#### 2.1.3.2 ข้อดีของ Material UI

- 1) มี Components ที่หลากหลาย
- 2) ใช้งานง่ายช่วยให้นักพัฒนาเว็บแอปพลิเคชัน สามารถเริ่มต้นและสร้าง UI ที่สวยงามได้อย่างรวดเร็ว
- 3) สามารถปรับแต่งรูปลักษณ์และการใช้งานต่างๆ ได้ตามต้องการ
- 4) มีการอัปเดตและมีเวอร์ชันใหม่อย่างต่อเนื่อง
- 5) รองรับ RTL (ขวาไปซ้าย) Accessibility และ TypeScript
- 6) มีเครื่องมือ Testing และ State Management
- 7) มี Community ขนาดใหญ่ ที่ให้คำปรึกษาและชี้แนะสิ่งใหม่ ๆ

### 2.1.3.3 ข้อเสียของ Material UI

- 1) มีความซับซ้อน อาจทำให้เข้าใจและใช้งานได้ยาก สำหรับมือใหม่
- 2) มีขนาดไฟล์ใหญ่ ซึ่งอาจทำให้โปรเจกต์ของคุณมีขนาดใหญ่ขึ้นด้วย
- 3) มีการอัพเดทและมีเวอร์ชันใหม่ อาจมีความซับซ้อน และอาจทำให้มีปัญหาในการรวมกับโค้ดที่มีอยู่แล้ว
- 4) อาจไม่เหมาะสมกับโปรเจกต์ที่มีความซับซ้อน

### 2.1.4 Node JS



รูปที่ 2.4 Node JS

(ที่มา: <https://freebiesupply.com/logos/node-js-logo>)

NodeJS หรือ Open Source ที่ทำหน้าที่เป็น JavaScript Runtime (ตัวแปรภาษา JavaScript) ซึ่ง NodeJS จะนำเอา JavaScript ที่เคย Run อยู่ใน Browser (Client-Side) เท่านั้น ให้สามารถนำไป Run บน Server (Server-Side) ได้ ซึ่งถือได้ว่าเป็นจุดเปลี่ยนของ JavaScript กันเลยทีเดียว

NodeJS พัฒนาขึ้นมาจาก Chrome V8 JavaScript Engine ที่เป็น Open-Source พัฒนาขึ้นมาจากการนำ C++ สามารถนำไป run ได้ทุก Platform ทั้ง Windows, Linux และ MacOS รวมทั้งยังรองรับ CPU หลายสถาปัตยกรรม เช่น ARM, IA-32, x64

NodeJS ทำงานแบบ Single Process โดยมี Event-Loop เข้ามาช่วยในการทำงานแบบ Asynchronous คือ รูปแบบการทำงานของชุดคำสั่งที่เขียนขึ้นมา โดยทำงานแบบไม่เรียงขั้นตอน เนื่องจากชุดคำสั่งทำงานพร้อมกัน และเมื่อคำสั่งไหนเสร็จเรียบร้อยจะแสดงผลลัพธ์ก่อนแบบ non-blocking I/O สามารถส่ง Request ของ User 1 และ User 2 พร้อมกันได้เลย ทำให้ลดการใช้ Thread ได้ โดย NodeJS ไม่เน้นสำหรับการทำงานที่เป็น CPU Intensive เพราะทำให้ถูก Block การทำงานทั้งหมด

#### 2.1.4.1 ข้อดีของ NodeJS

- 1) การทำงานแบบ non-blocking I/O ที่จะทำให้เราสามารถส่ง Request เข้าไปได้พร้อม ๆ กันมากขึ้น
- 2) เป็น JavaScript Run-Time ที่มีขนาดเล็ก ใช้ CPU และ Memory น้อย
- 3) JavaScript เป็นภาษาที่มีคนใช้เยอะมาก ดังนั้น Library ต่าง ๆ จึงมีให้เลือกใช้งานมาก เช่นกัน
- 4) มีเครื่องมือที่ช่วยจัดการติดตั้ง Package ที่มีพร้อมกับ Node.js อย่าง NPM
- 5) สามารถพัฒนา Front-End และ Back-End ด้วยภาษาเดียวกันได้

#### 2.1.4.2 ข้อเสียของ NodeJS

- 1) การทำงานของ NodeJS จะมี Callback เยอะมาก หากเขียนซ้อนกันไปมาก ๆ จะทำให้เกิดปัญหา Callback Hell
- 2) เครื่องมือมีให้เลือกใช้เยอะมาก ซึ่งบางตัวอาจมีคุณภาพแย่และอาจมีช่องโหว่เกิดขึ้นได้
- 3) NodeJS จะมีปัญหากับงานที่ใช้ CPU ในแต่ละ Task สูงมาก ๆ เพราะจะทำให้เกิดคอขวด

#### 2.1.5 MySQL



รูปที่ 2.5 MySQL

(ที่มา: [https://wikitech.wikimedia.org/wiki/File:MySQL\\_logo.png](https://wikitech.wikimedia.org/wiki/File:MySQL_logo.png))

MySQL คือ ระบบจัดการฐานข้อมูล หรือ Database Management System (DBMS) แบบข้อมูลเชิงสัมพันธ์ หรือ Relational Database Management System (RDBMS) ซึ่งเป็นระบบฐานข้อมูลที่จัดเก็บรวบรวมข้อมูลในรูปแบบตาราง โดยมีการแบ่งข้อมูลออกเป็นแถว (Row) และในแต่ละแถวแบ่งออกเป็นคอลัมน์ (Column) เพื่อเข้มข้นระหว่างข้อมูลในตารางกับข้อมูลในคอลัมน์ที่กำหนด แทนการเก็บข้อมูลที่แยกออกจากกัน โดยไม่มีความเชื่อมโยงกัน ซึ่งประกอบด้วยข้อมูล

(Attribute) ที่มีความสัมพันธ์เชื่อมโยงกัน (Relation) โดยใช้ RDBMS Tools สำหรับการควบคุมและจัดเก็บฐานข้อมูลที่จำเป็น ทำให้นำไปประยุกต์ใช้งานได้ง่าย ช่วยเพิ่มประสิทธิภาพในการทำงานให้มีความยืดหยุ่นและรวดเร็วได้มากยิ่งขึ้น รวมถึงเชื่อมโยงข้อมูล ที่จัดแบ่งกลุ่มข้อมูลแต่ละประเภทได้ตามต้องการ จึงทำให้ MySQL เป็นโปรแกรมระบบจัดฐานข้อมูลที่ได้รับความนิยมสูง

MySQL มีหน้าที่จัดเก็บข้อมูลอย่างเป็นระบบ รองรับคำสั่งภาษา Structured Query Language หรือ SQL เพื่อจัดการกับฐานข้อมูลโดยเฉพาะ เป็นภาษามาตรฐานบนระบบฐานข้อมูลเชิงสัมพันธ์และเป็นระบบเปิด (Open System) ที่มีโครงสร้างของภาษาที่เข้าใจง่าย ไมซับซ้อน และนิยมใช้งานร่วมกับภาษาโปรแกรม PHP รวมถึงภาษาอื่น ๆ ที่สามารถทำงานร่วมกันกับฐานข้อมูล MySQL ได้หลากหลาย เช่น C, C++, Python, Java เป็นต้น

MySQL ยังได้รับการออกแบบและปรับให้มีความเหมาะสมสำหรับการพัฒนา Website และ Web Application ทำให้สามารถรองรับการทำงานได้ทุกแพลตฟอร์ม รวมถึงการอนุญาตให้ผู้ใช้งานหลายคนสามารถใช้งานพร้อมกันได้ (Multi-User) นอกจากนี้ยังสามารถจัดการและสร้างฐานข้อมูลจำนวนมากรวมถึงประมวลผลหลาย ๆ งานได้พร้อมกัน (Multi-Threaded) อย่างสมบูรณ์ จึงทำให้ MySQL เป็นตัวเลือกยอดนิยมสำหรับธุรกิจการพาณิชย์อิเล็กทรอนิกส์ หรือ Electronic Commerce (E-Commerce) และเหมาะสมสำหรับการนำไปใช้งานสร้างเว็บไซต์ทั่วไป เพราะมีความแม่นยำ ครอบครัน ช่วยให้เข้าถึงข้อมูลได้อย่างรวดเร็ว อีกทั้งยังมีความน่าเชื่อถือสูง และยังมีโปรแกรมเสริมช่วยจัดฐานข้อมูลที่ใช้งานง่าย เช่น MySQL Admin, phpMyAdmin เป็นต้น

#### 2.1.5.1 ข้อดีของ MySQL

- 1) มี Version Open Source ใช้งานได้ฟรี
- 2) การป้องกันความปลอดภัยของข้อมูลที่แข็งแกร่ง สามารถกำหนดสิทธิ์และการรักษาความปลอดภัยของข้อมูลที่มีความรัดกุมด้วยรหัสผ่านที่มีความปลอดภัยสูง
- 3) ช่วยเพิ่มพื้นที่เก็บข้อมูลและจัดสรรทรัพยากรได้ง่ายและเป็นระเบียบขึ้น
- 4) ปรับขนาดได้ตามต้องการ รวมถึงจัดเก็บข้อมูลในรูปแบบต่าง ๆ ได้ง่าย
- 5) มีความสามารถในการสร้างและแก้ไขฐานข้อมูลเพื่อใช้งานได้ง่ายขึ้น
- 6) เพิ่มประสิทธิภาพการทำงานของฐานข้อมูลได้มากยิ่งขึ้น

#### 2.1.5.2 ข้อเสียของ MySQL

- 1) ปัญหาเรื่องความเสถียร และมักจะเกิดการเสียหายได้ในบางกรณี
- 2) ประสิทธิภาพต่ำเมื่อโหลดสูง อาจไม่เหมาะสมกับระบบที่ต้องการประสิทธิภาพสูงมากสำหรับการบันทึก หรือการทำธุกรรมในจำนวนมากกว่าปกติ
- 3) ฟีเจอร์การรักษาความปลอดภัย อาจพบเจอปัญหาเรื่องระบบความปลอดภัย เมื่อเทียบกับระบบจัดการฐานข้อมูลอื่น

### 2.1.6 Express API



รูปที่ 2.6 Express API

(ที่มา: <https://www.manektech.com/express-js-development-service>)

Express เป็น Framework ของ Node.js ที่ได้รับความนิยมและใช้กันเป็นจำนวนมากมาก และเป็นวิธีที่สร้าง API สำหรับ Provide ให้ Web Application และ Mobile Application ที่มีประสิทธิภาพมากๆ อีกทั้ง Express ยังช่วยจัดการ จัดการเรื่อง HTTP Request/Response และ Handle Data อีกด้วย มีฟีเจอร์ต่างๆ ที่ช่วยให้เราทำเว็บได้สะดวกขึ้น เช่น การทำ Routing, Middleware การจัดการ Request และ Response เป็นต้น ทำให้เราสามารถพัฒนาเว็บโดยใช้ Node.js ได้สะดวกและรวดเร็วอีกด้วย

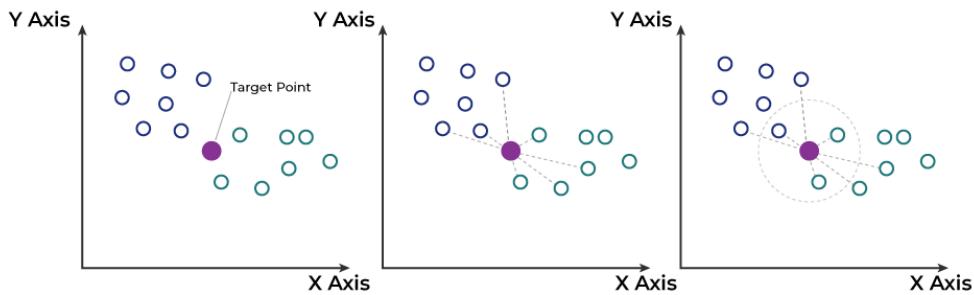
#### 2.1.6.1 ข้อดีของ Express API

- 1) Express มีรูปแบบการทำงานที่ไม่ซับซ้อน ทำให้เป็นทางเลือกที่สำคัญที่สุดสำหรับนักพัฒนาเว็บในทุกระดับ
- 2) ด้วยความที่เป็น Minimalistic Framework ตั้งนั้นผู้พัฒนาสามารถเพิ่มเติมหรือแก้ไขฟีเจอร์ที่ต้องการได้อย่างง่ายดาย
- 3) ช่วยให้สามารถจัดการ Request และ Response ได้อยู่อย่างเต็มที่ ทำให้สามารถสร้างการจัดการเส้นทางและการจัดการข้อมูลพลาได้อย่างมีประสิทธิภาพ
- 4) สามารถสร้าง RESTful API ได้อย่างง่ายดาย ทำให้การจัดการ Backend สำหรับทั้งเว็บและมือถือเป็นไปอย่างราบรื่น

#### 2.1.6.2 ข้อเสียของ Express API

- 1) ในบางครั้งความยืดหยุ่นของมันอาจนำไปสู่การทำหนดรูปแบบของโค้ดที่ไม่สม่ำเสมอ ซึ่งอาจทำให้มีปัญหาเรื่องการใช้งานร่วมกันในทีมพัฒนา
- 2) หากไม่มีการวางแผนและออกแบบโครงสร้างของแอปพลิเคชันที่ดี อาจทำให้โค้ดมีความยุ่งเหยิงและยากต่อการบำรุงรักษาในระยะยาว
- 3) เมื่อเทียบกับเฟรมเวิร์กอื่นๆ Express อาจขาดฟีเจอร์บางอย่างที่สร้างมาให้พร้อมใช้งาน เช่น การจัดการความปลอดภัย หรือการรองรับฐานข้อมูลบางประเภท

### 2.1.7 K-NN



รูปที่ 2.7 K-nearest neighbors (K-NN)

(ที่มา: <https://www.geeksforgeeks.org/k-nearest-neighbours>)

K-nearest neighbors (K-NN) เป็นอัลกอริทึมสำหรับ Supervised Learning ใช้ในการจัดกลุ่มของข้อมูล (Classification) หรือคำนวณค่าของตัวแปรต่อเนื่อง (Regression) โดยทำการหาข้อมูลที่ใกล้เคียงกับข้อมูลปัจจุบัน (Current Data) จากนั้นให้ค่าของตัวแปรเป้าหรือค่าที่ต้องการคำนวณจากข้อมูลที่ใกล้เคียงที่สุด (Nearest Neighbors)

วิธีการทำงานของ K-NN คือในขั้นตอนการทดสอบ โดยการหาข้อมูลที่ใกล้เคียงกับข้อมูลปัจจุบัน โดยหาข้อมูลที่ใกล้เคียงกันโดยใช้วิธีการคำนวณระยะห่าง Euclidean Distance หรือ Manhattan distance ระหว่างค่าของคุณสมบัติ (Feature) ในข้อมูล และจากนั้นเลือกข้อมูลที่ใกล้เคียงกันที่สุด  $k$  ตัว โดย  $k$  เป็นตัวแปรที่กำหนดไว้ จากนั้นจะให้ค่าของตัวแปรเป้าหรือค่าที่ต้องการคำนวณจากตัวเลือก  $k$  ตัวนี้ เช่น หากต้องการจัดกลุ่มข้อมูลให้เป็น 2 กลุ่ม จะต้องกำหนด  $k=1$  หรือ  $k=3$  ก่อน จากนั้นคำนวณระยะห่างระหว่างข้อมูลทุกตัวกับข้อมูลปัจจุบันและเลือก  $k$  ตัวที่ใกล้เคียงกับข้อมูลปัจจุบันมากที่สุด และนับจำนวนข้อมูลในแต่ละกลุ่มเพื่อกำหนดว่าข้อมูลปัจจุบันอยู่ในกลุ่มไหน

#### 2.1.7.1 ข้อดีของ K-NN

1) ไม่ต้องมีสมมติฐานเกี่ยวกับ Function ที่เลือกมาใช้ ทำให้เหมาะสมกับทุกประเภทของข้อมูล

2) เป็น Instance-Based Learning ทำให้ง่ายในการปรับตัว

3) รองรับปัญหาแบบ Multi-class Classification

#### 2.1.7.2 ข้อดีของ K-NN

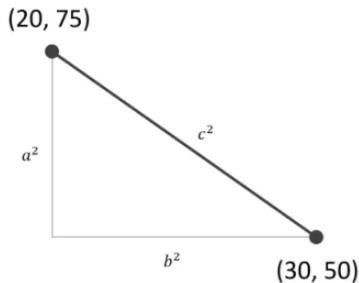
1) ต้องมี Storage สำหรับ Dataset ทั้งชุด การหา Nearest Neighbors บางครั้งมีความซับซ้อนในการคำนวณที่มาก

- 2) มีความไม่ต่อ Features ที่ไม่เกี่ยวข้อง เนื่องจากเป็นการคำนวณระยะทาง  
บางครั้ง Features ที่ไม่เกี่ยวข้องจะส่งผลต่อประสิทธิภาพของ Model ได้
- 3) มีความไม่ต่อ Scale สำหรับ Features ที่มี Scale ที่สูงกว่า (ค่ามากกว่า)  
จะส่งผลต่อการคำนวณระยะทางที่มากกว่า
- 4) ประสิทธิภาพลดลงหากเป็น High Dimensionality

#### 2.1.8 Euclidean Distance

ระยะทางแบบยุคลิด (Euclidean Distance) แสดงถึงเส้นทางที่สั้นที่สุดระหว่างจุดสอง  
จุดในปริภูมิยุคลิด ซึ่งเป็นระยะทางที่สามารถวัดได้โดยใช้มัตรทัด และสามารถขยายแนวคิดนี้ไปยัง  
มิติใด ๆ ก็ได้

แนวคิดนี้มีรากฐานมาจากทฤษฎีบทปีทาโกรัส (Pythagorean Theorem) ซึ่งระบุว่า  
ในสามเหลี่ยมมุมฉาก กำลังสองของความยาวด้านตรงข้ามมุมฉากจะเท่ากับผลรวมของกำลังสองของ  
ความยาวของอีกสองด้าน



รูปที่ 2.8 Euclidean Distance

(ที่มา: [https://medium.com/@srikong\\_n/cosine-similarity-f1f9a962ddc5](https://medium.com/@srikong_n/cosine-similarity-f1f9a962ddc5))

ในระบบสองมิติ ระยะทางแบบยุคลิดระหว่างจุด A( $x_1, y_1$ ) และ B( $x_2, y_2$ ) สามารถ  
คำนวณได้โดยใช้สูตรต่อไปนี้

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

รูปที่ 2.9 สูตรการคำนวณของ 2D Euclidean Distance

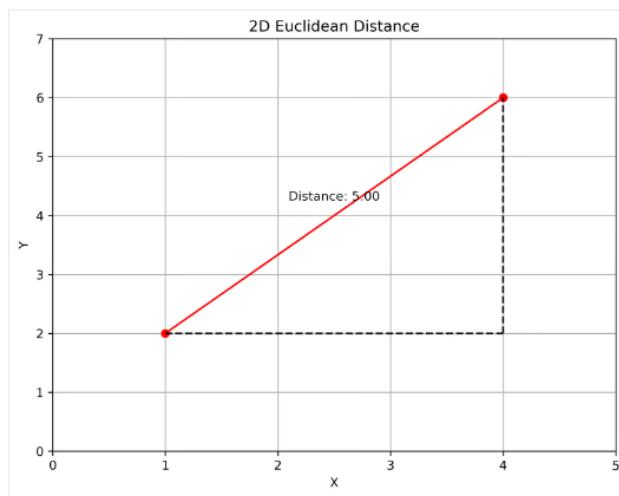
(ที่มา: <https://www.datacamp.com/tutorial/euclidean-distance>)

จากรูปที่ 2.9 สูตรการคำนวณระยะทางที่สั้นที่สุดระหว่างจุดสองจุดในระบบโดยอ้างอิงจากทฤษฎีบทพีทาゴรัส ซึ่งใช้ผลรวมของกำลังสองของระยะห่างในแนวแกน x และ y และลดรากที่สองเพื่อหาค่าระยะทางจริง

$$\begin{aligned}
 d &= \sqrt{(4-1)^2 + (6-2)^2} \\
 &= \sqrt{3^2 + 4^2} \\
 &= \sqrt{9+16} \\
 &= \sqrt{25} \\
 &= 5
 \end{aligned}$$

รูปที่ 2.10 ตัวอย่างการคำนวณ 2D Euclidean Distance  
(ที่มา: [https://medium.com/@srikong\\_n/cosine-similarity-f1f9a962ddc5](https://medium.com/@srikong_n/cosine-similarity-f1f9a962ddc5))

จากรูปที่ 2.10 ตัวอย่างการคำนวณ ให้จุด A(1,2) และจุด B(4,6) คำนวณ ระยะทางระหว่างสองจุดนี้โดยใช้สูตรการหาระยะทางของ Euclidean distance โดยคำนวณค่าผลต่างของพิกัดแต่ละแกน จากนั้นยกกำลังสองของแต่ละค่าและบวกค่าภายในเครื่องหมายรากที่สอง ขั้นตอนสุดท้ายให้ทำการลดรากที่สอง จะได้ผลลัพธ์ออกมานะ



รูปที่ 2.11 ผลลัพธ์การคำนวณ 2D Euclidean Distance  
(ที่มา: [https://medium.com/@srikong\\_n/cosine-similarity-f1f9a962ddc5](https://medium.com/@srikong_n/cosine-similarity-f1f9a962ddc5))

จากรูปที่ 2.11 การแสดงผลนี้แสดงระยะทางแบบยุคลิดระหว่างจุดสองจุดในระบบสองมิติ เส้นสีแดงแสดงถึงระยะทางโดยตรงระหว่างจุดทั้งสอง ขณะที่เส้นประสร้างเป็นรูปสามเหลี่ยมมุมฉาก เพื่อแสดงการทำงานของทฤษฎีบทพีทาゴรัส

### 2.1.9 Vercel



รูปที่ 2.12 Vercel  
(ที่มา: <https://en.wikipedia.org/wiki/Vercel>)

Vercel เป็น Cloud Platform สำหรับ Build และ Deploy ให้เมื่อมีการเชื่อมต่อกับ GitHub ก็สามารถที่จะ Automate Deployment ให้ได้เลย และถ้ามี Domain Name ก็สามารถเอามาเพิ่มได้เหมือนกัน ซึ่งขั้นตอนการเพิ่ม Domain นั้นก็ง่ายมาก และยังมี Analytics ซึ่งจะทำการ Monitor Application Performance, Logs และอื่น ๆ อีกมากมาย

Vercel สามารถช่วยให้การ Deploy รวดเร็วและง่ายดาย พร้อมการรองรับ การขยายขนาดอัตโนมัติ (Auto Scaling) และเทคโนโลยีเว็บยุคใหม่อย่าง Serverless Functions นอกจากนี้ยังมี เครือข่าย CDN ทั่วโลก และ HTTPS อัตโนมัติ ที่ช่วยให้สามารถส่งมอบเนื้อหาเว็บไปยังผู้ใช้ทั่วโลกได้อย่างปลอดภัยและมีประสิทธิภาพ มีอินเตอร์เฟชที่ใช้งานง่าย และรองรับการเชื่อมต่อกับเครื่องมืออุดนิยม เช่น GitHub, GitLab และ Bitbucket ทำให้การตั้งค่าและจัดการโปรเจกต์เป็นเรื่องง่าย มีแพ็กเกจฟรี ที่ให้ฟีเจอร์มากมาย หมายเหตุสำหรับโปรเจกต์ขนาดเล็กและการใช้งานส่วนตัว

Vercel ถูกพัฒนาบนเทคโนโลยีเชิร์ฟเวอร์เลสที่ทันสมัยที่สุด ทำให้สามารถรองรับปริมาณการใช้งานที่เพิ่งสูงขึ้นได้โดยไม่มีปัญหา

#### 2.1.9.1 ข้อดีของ Vercel

- 1) Deploy รวดเร็วและง่ายดาย
- 2) มีอินเตอร์เฟชที่ใช้งานง่าย
- 3) เชื่อมต่อกับระบบ Backend ได้สะดวก
- 4) ไม่ต้องตั้งค่าเซิร์ฟเวอร์ CDN เอง
- 5) รองรับการทำงานร่วมกันแบบเรียลไทม์

#### 2.1.9.2 ข้อเสียของ Vercel

- 1) มีข้อจำกัดของแพ็กเกจฟรี
- 2) รองรับเฉพาะบางเทคโนโลยี
- 3) ไม่มีฐานข้อมูลในตัว ต้องใช้บริการภายนอก
- 4) จำกัดระยะเวลาในการทำงานของฟังก์ชัน (Execution Time Limit) และขนาดของแพ็กเกจโค้ด

### 2.1.10 Railway



รูปที่ 2.13 Railway  
(ที่มา: <https://railway.com/design>)

Railway เป็นแพลตฟอร์มสำหรับ Deployment ที่ออกแบบมาเพื่อเพิ่มประสิทธิภาพ ให้กับการพัฒนาซอฟต์แวร์ ช่วยให้นักพัฒนาสามารถปรับใช้แอปพลิเคชันได้อย่างรวดเร็วและง่ายดาย สามารถจัดการบริการ ฐานข้อมูล และพื้นที่เก็บข้อมูลได้อย่างมีประสิทธิภาพ ไปจนถึงการผ่านรวม กับ CI/CD และเครื่องมือสังเกตการณ์ที่มีมาในตัว Railway ให้ความสำคัญกับ ความเสถียรและความ โปร่งใส ใช้เครื่องมือแจ้งเตือน ระบบภายใน และกระบวนการดำเนินงานที่มีประสิทธิภาพ

Railway ช่วยให้สามารถ Deploy Container Image ได้ ฯ ได้ง่าย ๆ เพียงระบุ แหล่งที่มา ไม่ว่าจะเป็น GitHub Repositories, Docker Images และอื่น ๆ มีแพลตฟอร์มรองรับ การปรับขนาดทั้งในแนวตั้งและแนวนอนตามความต้องการของแอปพลิเคชัน โดยสามารถเพิ่ม CPU และ RAM โดยอัตโนมัติ หรือเพิ่มจำนวนเครื่องเซิร์ฟเวอร์เพื่อรับรับงานที่ต้องใช้ทรัพยากรสูง

#### 2.1.10.1 ข้อดีของ Railway

- 1) Deploy รวดเร็วและง่ายดาย
- 2) สามารถเพิ่มหรือลดทรัพยากรได้ตามต้องการ
- 3) มีเครื่องมือสำหรับตรวจสอบการทำงานของระบบ
- 4) รองรับแอปพลิเคชันหลายประเภท

#### 2.1.10.2 ข้อเสียของ Railway

- 1) มีข้อจำกัดด้านระยะเวลาและทรัพยากร
- 2) ไม่รองรับบางฟีเจอร์ที่ซับซ้อน
- 3) ตัวเลือกการตั้งค่าขั้นสูงอาจมีจำกัด

### 2.1.11 Cloudinary



รูปที่ 2.14 Cloudinary

(ที่มา: [https://commons.wikimedia.org/wiki/File:Cloudinary\\_logo\\_blue\\_0720\\_2x.png](https://commons.wikimedia.org/wiki/File:Cloudinary_logo_blue_0720_2x.png))

Cloudinary เป็นโซลูชันแบบครบวงจรสำหรับการจัดการรูปภาพบนเว็บไซต์และแอปพลิเคชันมือถือ รองรับการอัปโหลดรูปภาพ พื้นที่จัดเก็บบนคลาวด์ การปรับแต่งรูปภาพ การเพิ่มประสิทธิภาพรูปภาพสำหรับเว็บ และการส่งมอบเนื้อหา Cloudinary มี API และฟังก์ชันการจัดการที่ยืดหยุ่น ซึ่งสามารถสนับสนุนรวมเข้ากับแอปพลิเคชันบนเว็บและมือถือ ทั้งที่มีอยู่เดิมและที่พัฒนาใหม่ได้อย่างราบรื่น

Cloudinary ช่วยให้ผู้ใช้สามารถอัปโหลดและจัดเก็บรูปภาพได้อย่างไม่จำกัดในลักษณะส่วนตัวและปลอดภัย พร้อมทั้งมีระบบสำรองข้อมูลอัตโนมัติและบันทึกประวัติการแก้ไข ผู้ใช้สามารถปรับแต่งรูปภาพได้หลากหลายรูปแบบ เช่น การใส่เอฟเฟกต์ การปรับขนาด การครอบตัด การตรวจสอบใบหน้า การใส่ลายน้ำ และอื่น ๆ โดยรูปภาพจะถูกส่งผ่านเครือข่าย CDN ระดับโลกของ Akamai เพื่อให้สามารถโหลดได้อย่างรวดเร็ว ตอบสนองได้ดี และเหมาะสมกับทุกอุปกรณ์ Cloudinary ทำหน้าที่เป็นโซลูชันการจัดการสื่อดิจิทัล (Digital Asset Management) โดยมีคลังสื่อที่ทำงานผ่าน RESTful API และ SDK ช่วยให้การจัดการรูปภาพเป็นไปโดยอัตโนมัติ

#### 2.1.11.2 ข้อดีของ Cloudinary

- 1) รองรับการอัปโหลดและเก็บรักษาภาพอย่างปลอดภัย
- 2) ปรับแต่งรูปภาพได้หลายรูปแบบ
- 3) ความสะดวกในการใช้งานและการนำไปใช้
- 4) รองรับรูปภาพหลากหลายรูปแบบไฟล์

#### 2.1.11.2 ข้อเสียของ Cloudinary

- 1) อาจมีค่าใช้จ่ายที่สูงสำหรับผู้ใช้ที่ต้องการพื้นที่จัดเก็บหรือการแปลงภาพในปริมาณมาก
- 2) แม้ว่าจะมี API ที่ใช้งานได้ง่าย แต่การปรับแต่งบางฟีเจอร์อาจต้องใช้เวลาหรือความเข้าใจลึกซึ้งในระบบ
- 3) การตั้งค่าและการบำรุงรักษาที่ซับซ้อน

### 2.1.12 Nodemailer



รูปที่ 2.15 Nodemailer  
(ที่มา: <https://nodemailer.com/about>)

Nodemailer ซึ่งเป็นแพ็คเกจยอดนิยมบน NPM ที่ให้ API ระดับสูงสำหรับการส่งอีเมล ใน JavaScript โดยทำงานบนพื้นฐานของโปรโตคอล SMTP (Simple Mail Transfer Protocol) แพ็คเกจนี้ถูกออกแบบมาเพื่อช่วยลดความซับซ้อนของ SMTP ทำให้นักพัฒนาสามารถส่งอีเมลได้โดยไม่ต้องกังวลเกี่ยวกับรายละเอียดของโปรโตคอลที่อยู่เบื้องหลัง

Nodemailer มี API ที่ใช้งานง่ายและเป็นมิตรกับนักพัฒนา ช่วยให้สามารถกำหนดค่า และส่งอีเมลผ่านเซิร์ฟเวอร์ SMTP ได้ด้วยโค้ดเพียงไม่กี่บรรทัด แพ็คเกจนี้รองรับฟีเจอร์ต่าง ๆ ของ อีเมลสมัยใหม่ เช่น เนื้อหาแบบ HTML ไฟล์แนบ และรูปภาพที่ฝังอยู่ในอีเมล นอกจากนี้ ยังรองรับ หลายวิธีในการส่งอีเมล หัน SMTP, Sendmail และ Amazon SES

#### 2.1.12.1 ข้อดีของ Nodemailer

- 1) มี API ที่เรียบง่าย ทำให้สามารถส่งอีเมลได้โดยใช้โค้ดเพียงไม่กี่บรรทัด
- 2) นอกจาก SMTP แล้ว ยังรองรับการส่งอีเมลผ่าน Sendmail, Amazon SES, และ Mailgun
- 3) สามารถใช้ร่วมกับ OAuth2, Plain Authentication, และวิธีการตรวจสอบสิทธิ์อื่น ๆ เพื่อความปลอดภัยในการส่งอีเมล
- 4) เป็นแพ็คเกจที่ได้รับความนิยม จึงมีเอกสารอธิบายการใช้งานที่ดี รวมถึง ชุมชนผู้ใช้งานที่สามารถช่วยเหลือและให้คำแนะนำได้
- 5) สามารถส่งอีเมลที่มีเนื้อหา HTML, แนบไฟล์, ฝังรูปภาพ และกำหนด พอร์แมตของอีเมลได้ง่าย

#### 2.1.12.2 ข้อเสียของ Nodemailer

- 1) ต้องใช้เซิร์ฟเวอร์ SMTP หรือบริการอีเมลภายนอก ซึ่งอาจมีข้อจำกัดด้านจำนวนอีเมลที่ส่งได้ต่อวัน
- 2) ไม่เหมาะสมสำหรับการส่งอีเมลจำนวนมาก
- 3) ต้องกำหนดค่าให้ถูกต้องเพื่อป้องกันปัญหาด้านความปลอดภัยและการส่งอีเมลไปยัง Spam

## 2.2 งานวิจัยที่เกี่ยวข้อง

จิตราภรณ์ ราษฎร์ทักษิวงศ์ ได้ทำการสร้างโมเดลจำแนกรูปแบบการเรียนรู้เพื่อจัดระดับมาตรฐานผลิตภัณฑ์ OTOP หัตถกรรม กลุ่มไม้ ด้วย K-Nearest Neighbor Algorithm และใช้พังก์ชันระยะห่างยูคลิดเดียน (Euclidean Distance) มาคำนวณระยะห่างของข้อมูลตัวอย่างที่สนใจ กับข้อมูลในฐานกรณี (Case Based) ทุกด้า โดยจัดเก็บข้อมูลผลิตภัณฑ์ OTOP ที่ผ่านการประเมินคัดสรรสุดยอดหนึ่งตำบล หนึ่งผลิตภัณฑ์ ซึ่งมี attribute ที่ใช้ในการพิจารณาทั้งหมด 22 attribute มาจำแนกประเภทด้วย K-Nearest Neighbor Algorithm โดยใช้โปรแกรม Weka เลือกวิธีทดสอบแบบ K-Fold Cross Validation ผลการศึกษาพบว่า การจำแนกประเภทการเรียนรู้เพื่อจัดระดับมาตรฐานผลิตภัณฑ์ OTOP หัตถกรรม กลุ่มไม้ ด้วย K-Nearest Neighbor Algorithm มีค่า Distance ที่ใช้ในการเปรียบเทียบความคล้ายกัน (Similarity) เพื่อนำมาคำนวณโดยพิจารณาจากข้อมูลที่มีระยะห่างยูคลิดเดียนน้อยที่สุดจำนวน 3 กรณี ซึ่งทำให้ได้ค่า Accuracy ที่ร้อยละ 88.34 ค่า Recall ร้อยละ 88.30 และค่า Precision ร้อยละ 83.4 [2]

รัวช รวมทรัพย์ ได้พัฒนาการจำแนกชนิดของพืชด้วยขั้นตอนวิธีเพื่อบ้านไกลสุด ร่วมกับการเลือกตัวแทนที่เหมาะสมด้วยขั้นตอนวิธีเชิงพันธุกรรมโดยใช้คุณลักษณะรูปทรงและพื้นผิวของใบพืช โดยในงานวิจัยนี้ได้ทำการทดลองกับใบพืช 30 สายพันธุ์ จำนวน 340 ใบ จากผลการทดลองโดย งานวิจัยนี้ใช้ขั้นตอนวิธีเชิงพันธุกรรมในการเลือกตัวแทนที่เหมาะสม และใช้ขั้นตอนวิธีเพื่อบ้านไกลสุด (K-NN) ในการจำแนกชนิดของพืชซึ่งให้ผลลัพธ์ความถูกต้องในการจำแนกชนิดของพืช (Accuracy) 86.27% ค่าความครบถ้วน (Recall) 83.06% และค่าความแม่นยำ (Precision) 84.06% เมื่อเปรียบเทียบกับการใช้ขั้นตอนวิธีเพื่อบ้านไกลเดียวในการจำแนกชนิดพืชเพียงอย่างเดียวซึ่งให้ผลลัพธ์ความถูกต้องในการจำแนกชนิดของพืช (Accuracy) 79.12% , ค่าความครบถ้วน (Recall) 79.50% และค่าความแม่นยำ (Precision) 73.17% ซึ่งจากผลการทดลองพบว่า เมื่อใช้ขั้นตอนเชิงพันธุกรรมมาช่วยในการเลือกตัวแทนที่เหมาะสมทำให้ประสิทธิภาพการจำแนกชนิดพืชด้วยขั้นตอนวิธีเพื่อบ้านไกลสุดมีความถูกต้องมากขึ้น มากกว่าการใช้ขั้นตอนเพื่อบ้านไกลสุดเพียงอย่างเดียว [5]

พรรณธิภา เพชรบุญมี ได้ทำงานวิจัยที่มีวัตถุประสงค์เพื่อเปรียบเทียบประสิทธิภาพอัลกอริทึม เหมือนข้อมูลเพื่อจำแนกประเภทข้อมูลความสามารถทางการเรียนรู้ตามแนวทางพหุปัญญา สำหรับนักศึกษา มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนาตาก ทำการเก็บรวบรวมข้อมูลแบบสอบถามที่ผู้วิจัยได้พัฒนาขึ้น ตั้งแต่ปีการศึกษา 2558 - 2559 จำนวน 1,407 คน ผลการเปรียบเทียบพบว่า อัลกอริทึม Decision Tree (J48) มีค่าความถูกต้องสูงสุด เท่ากับ 86.79% อัลกอริทึม K-Nearest Neighbor มีค่าความถูกต้องเท่ากับ 82.75% และ อัลกอริทึม Naive Bayes มีความถูกต้องเท่ากับ 79.88% ตามลำดับ [6]

ราดา จันตะคุณ ได้ทำงานวิจัยที่ได้พัฒนาตัวแบบสำหรับพยากรณ์ความเป็นไปได้ในการเลือกสมัครสาขาวิชาโดยใช้เทคนิคเหมือนข้อมูล โดยดำเนินการวิจัยตามกระบวนการตามมาตรฐานสำหรับการทำเหมือนข้อมูล CRISP-DM ด้วยโปรแกรม Rapid Miner Studio พบว่า ปัจจัยที่มีผลต่อการเลือกหลักสูตรการเรียนทั้งหมดมี 6 ตัวแปรหลัก ประกอบด้วย ได้แก่ 1. เกรดเฉลี่ยของผู้เรียนต่อการศึกษา เดิม 2. ระดับคะแนนรายวิชาวิทยาศาสตร์ 3. ระดับคะแนนวิชาคณิตศาสตร์ 4. ระดับคะแนนรายวิชา สังคม 5. ระดับคะแนนรายวิชาภาษาไทย 6. ระดับคะแนนรายวิชาภาษาอังกฤษ ผลการดำเนินการ วิจัยครั้งนี้โดยพัฒนาและเปรียบเทียบตัวแบบการจำแนกทั้ง 4 เทคนิคได้แก่ Decision Tree, Naive Bayes, k-NN, Rule Induction ซึ่งผลการประเมินประสิทธิภาพตัวแบบ คือ Decision Tree ซึ่งได้ค่าที่สูงที่สุดจากการแบ่งข้อมูลทดสอบออกเป็น 10 ชุด ค่าความถูกต้อง (Accuracy) ได้ 83.97% [4]

Sharukh RAHMAN S และ Manoj Kumar D S นำเสนอโปรเจกต์นวัตกรรมเกี่ยวกับ "ระบบการจับคู่ห้องและรูมเมทที่เหมาะสมโดยใช้อัลกอริทึม k-nearest neighbors พร้อมการกระจาย Cosine Similarity เพื่อทำให้การค้นหาห้องและรูมเมทที่เหมาะสมโดยพิจารณาจากความชอบของแต่ละบุคคล แบ่งโดยสรุปดังนี้ 1) สิ่งอำนวยความสะดวก (Amenities): เช่น อินเทอร์เน็ต, เครื่องปรับอากาศ, หรือเครื่องซักผ้า 2) เพศ (Gender) 3) ราคา (Price): ตั้งงบประมาณที่ต้องการ เพื่อให้ระบบแสดงตัวเลือกที่ตรงตามงบประมาณ 4) ทำเลที่ตั้ง (Location) โดยใช้วิธีการและเทคนิคต่าง ๆ ดังนี้ 1) การเก็บข้อมูลจากผู้ใช้ (User Data Collection): ในขั้นตอนการลงทะเบียน 2) การใช้คำถามหลายตัวเลือก (Multiple Choice Questions) 3) การคำนวณความคล้ายคลึง (Similarity Calculation) 4) อัลกอริทึมการจับคู่ (Matching Algorithms): ระบบจะใช้อัลกอริทึม Nearest Neighbors 5) การปรับปรุงผลลัพธ์ (Result Refinement): ผู้ใช้สามารถกรองและจัดเรียงผลลัพธ์ ตามความชอบ [21]

สุคนธิพย์ คำจันทร์, ประภาพร กุลลิ่มรัตน์ชัย - การประยุกต์ใช้ User Interface (UI) และ User Experience (UX) ในการออกแบบแพลตฟอร์ม (2022) : ปัจจุบันเทคโนโลยีมีการพัฒนาอย่างรวดเร็ว ทำให้องค์กรหรือบริษัทต้องมีการปรับตัวตามสถานการณ์ที่มีการเปลี่ยนแปลงของ Digital Transformation ซึ่งเป็นการเปลี่ยนแปลงทางด้านแนวคิดโดยมีการนำเทคโนโลยีมาปรับใช้กับองค์กร

หรือธุรกิจให้เกิดความน่าสนใจและเป็นจุดเด่นที่มีความแตกต่าง โดยเฉพาะกลุ่มธุรกิจทางด้านการให้บริการที่จะต้องคำนึงถึงผู้ใช้งานในการเลือกใช้แพลตฟอร์มด้านการให้บริการผ่านอุปกรณ์สื่อสารทุกประเภท ทำให้เกิดกลุ่มธุรกิจเหล่านี้ต้องให้ความสำคัญของการออกแบบส่วนต่อประสานผู้ใช้งาน User Interface (UI) และการออกแบบประสบการณ์ผู้ใช้ User Experience (UX) ที่มีผลต่ออารมณ์และความรู้สึกของการใช้งาน เช่น การเลือกสีในการออกแบบตัวอักษร การจัดวางตำแหน่งข้อมูล กราฟิก และปุ่มต่าง ๆ ที่ปรากฏในหน้าจอ เพื่อให้ผู้ใช้เกิดความพึงพอใจและเกิดแรงจูงใจในการใช้งานของแพลตฟอร์มนั้น ๆ ซึ่งต้องมีการนำความรู้เกี่ยวกับ UX มาออกแบบส่วนของผู้ใช้งานระบบ และ UI ในการทำ Wireframe ซึ่งเป็นการร่างองค์ประกอบต่าง ๆ ที่จะปรากฏในหน้าจอของแต่ละแพลตฟอร์ม และจึงนำมาสร้างเป็นต้นแบบ (Prototype) เพื่อนำไปออกแบบและทดสอบให้ได้มาซึ่งความสมบูรณ์ของหน้าจอ [7]

Rifkie Primartha, Adi Wijaya - การเปรียบเทียบวิธีการวัดระยะทางบนอัลกอริทึม K-Nearest Neighbor สำหรับการจำแนกประเภท : K-Nearest Neighbor เป็นอัลกอริทึมการจำแนกประเภทแบบไม่ใช้พารามิเตอร์ซึ่งไม่ใช้ข้อมูลการฝึกอบรมและสมมติฐานหรือแบบจำลองเริ่มต้นในกระบวนการคำนวณ คุณภาพของผลลัพธ์การจำแนกประเภท k-Nearest Neighbor นั้นขึ้นอยู่กับระยะห่างระหว่างวัตถุและค่า k ที่ระบุ ดังนั้นการเลือกวิธีการวัดระยะทางจึงกำหนดผลลัพธ์ของการจำแนกประเภท การศึกษาครั้งนี้เปรียบเทียบวิธีการวัดระยะทางหลายวิธี รวมถึงระยะทางแบบยุคลิด ระยะทางแบบแมนยัตตัน ระยะทางแบบเชบีเชฟ และระยะทางแบบโคงไชน์ เพื่อดูว่าวิธีการวัดระยะทางใดที่สามารถทำงานได้ดีที่สุดกับอัลกอริทึม k-Nearest Neighbor การเลือกค่า k ยังกำหนดผลลัพธ์ของอัลกอริทึมการจำแนกประเภท k-Nearest Neighbor ด้วย ดังนั้นการกำหนดค่า k จึงจำเป็นต้องคำนึงถึงความแม่นยำและค่าความคลาสสิกของข้อมูลที่ใช้ในการศึกษาครั้งนี้ เป็นชุดข้อมูลของมะเร็งปอดลูก ผลลัพธ์ความแม่นยำสูงสุดที่ได้จากการวัดระยะทางแบบโคงไชน์คือ เท่ากับ 92.559% โดยมีค่า k = 9 จากค่าความแม่นยำที่ได้เปรียบเทียบกัน วิธีการวัดระยะทางที่เหมาะสมที่สุดคือ การวัดระยะทางแบบโคงไชน์ โดยมีค่า k ที่ดีที่สุดที่ได้คือ k = 9 แม้ว่าวิธีการวัดระยะทางนี้จะมีเวลาในการประมวลผลสูงสุดซึ่งเท่ากับ 0.898 วินาทีก้าวตาม [19]

Ichwanul Muslim Karo Karo, Athallah Tsany Rakha Dzaky, Muhammad Alfhi Saputra - การวิเคราะห์เชิงเปรียบเทียบระหว่างอัลกอริทึม K-Nearest Neighbor และอัลกอริทึม K-Nearest Neighbor ที่ปรับเปลี่ยนแล้วสำหรับการจำแนกประเภทข้อมูลความเป็นอยู่ทางการเงิน (2021) : อัลกอริทึมการจำแนกประเภทที่ได้รับความนิยมมากที่สุดอย่างหนึ่งคือ K Nearest Neighbor (KNN) อย่างไรก็ตามยังมีอัลกอริทึมการจำแนกประเภท Modified K Nearest Neighbor (MKNN) ซึ่งเป็นการขยาย KNN ในบทความนี้ เราจะวิเคราะห์การเปรียบเทียบอัลกอริทึม KNN และ MKNN เพื่อจำแนกชุดข้อมูลความเป็นอยู่ที่ดีทางการเงิน การวิเคราะห์เชิงเปรียบเทียบนั้นอิงตาม

ความแม่นยำและเวลาดำเนินการของอัลกอริทึมทั้งสองก่อนกระบวนการจำแนกประเภท ได้มีการดำเนินการตรวจสอบแบบ k-Fold เพื่อค้นหาการสร้างแบบจำลองข้อมูลที่เหมาะสมที่สุด การวิเคราะห์เชิงเปรียบเทียบนั้นอิงตามความแม่นยำ ความแม่นยำ เวลาเรียกคืน และเวลาในการดำเนินการ ของอัลกอริทึมทั้งสอง เพื่อให้ได้ผลลัพธ์การจำแนกประเภทที่เหมาะสมที่สุด อัลกอริทึม KNN และ MKNN จะถูกเรียกใช้ด้วยการตรวจสอบแบบ k-Fold จากผลการทดสอบพบว่า ประสิทธิภาพของอัลกอริทึม KNN นั้นสูงกว่า MKNN ในทุกพารามิเตอร์การทดสอบ โดยมีช่องว่างเฉลี่ย 25 เปอร์เซ็นต์ นอกจากนี้ ยังพบว่าเวลาในการดำเนินการของอัลกอริทึม KNN นั้นเร็วกว่า MKNN อีกด้วย [16]

Mairéad A. Willis, Sean P. Lane - หลักฐานเบื้องต้นสำหรับโครงสร้างปัจจัย ความถูกต้องพร้อมกัน และความถูกต้องของโครงสร้างของมาตรฐานความสัมพันธ์เพื่อนร่วมห้องในกลุ่มตัวอย่างของวิทยาลัย (2022) : ความสัมพันธ์กับเพื่อนร่วมห้องเป็นปัจจัยพื้นฐานของสภาพแวดล้อมทางสังคมของผู้ใหญ่รุ่นใหม่ทั้งคน อย่างไรก็ตาม ยังไม่มีมาตรฐานความสัมพันธ์กับเพื่อนร่วมห้องที่ได้รับการยอมรับและใช้งานอย่างแพร่หลาย เพื่อตรวจสอบผลกระทบของความสัมพันธ์เหล่านี้ต่อการทำงานและสุขภาพของบุคคล ในรายงานนี้ เราได้นำเสนอหลักฐานเบื้องต้นเกี่ยวกับโครงสร้างปัจจัย ความถูกต้องพร้อมกัน และความถูกต้องเชิงสร้างสรรค์ของมาตรฐานความสัมพันธ์กับเพื่อนร่วมห้อง (RRS) ในฐานะมาตรฐานความสัมพันธ์กับเพื่อนร่วมห้อง โดยใช้กลุ่มตัวอย่างนักศึกษาวิทยาลัยสหรัฐอเมริกาที่เข้าร่วมการศึกษาหลายระยะ การวิเคราะห์ปัจจัยสำรวจในระลอกแรก และการวิเคราะห์ปัจจัยยืนยัน (CFAs) ด้วยกลุ่มตัวอย่างอิสระของผู้เข้าร่วมรายใหม่ในแต่ละระลอกต่อมาสองระลอก แสดงให้เห็นการโหลดปัจจัยที่มั่นคงและความเหมาะสมเพียงพอ นอกจากนี้ มาตรวัดยังแสดงให้เห็นถึงความเหมาะสมและความน่าเชื่อถือที่ดีในกรอบ CFA ระดับหลายระดับแบบตามiyaw RRS มีความสัมพันธ์เชิงบวกอย่างมีนัยสำคัญกับระยะเวลาความสัมพันธ์ ความนับถือตนเอง การหันเห ความเอื้อเฟื้อ และความมีระเบียบวินัย และมีความสัมพันธ์เชิงลบกับอาการของความวิตกกังวลและรูปแบบการยืดติดหลีกเลี่ยง ซึ่งบ่งชี้ถึงความถูกต้องพร้อมกันของมาตรฐานความสัมพันธ์ที่ยึดกับโครงสร้างเหล่านี้ สอดคล้องกับผลการวิจัยจากความสัมพันธ์ประเภทอื่นๆ คะแนน RRS ที่รายงานด้วยตนเองลดลงตามiyaw ทั้งข้ามภาคการศึกษาและระหว่างภาคการศึกษาของชีวิตนักวิชาการ ซึ่งบ่งชี้ถึงความถูกต้องเชิงสร้างสรรค์ของมาตรฐาน RRS มีประโยชน์สำหรับการประเมินคุณภาพความสัมพันธ์กับเพื่อนร่วมห้องในหมู่นักศึกษาวิทยาลัยสหรัฐอเมริกา และอาจจะมากกว่านั้น การวิจัยเพิ่มเติมควรตรวจสอบความสามารถใช้งานของมาตรฐานความสัมพันธ์ในประชากรอื่นๆ ที่หลากหลายมากขึ้น และปรับปรุงกระบวนการทางจิตวิทยาพื้นฐานที่สร้างขึ้น [17]

Aastha Bhatra, Mariyah Zubair Khan, Ashita Gupta, Amanat Bansal - การเพิ่มความสัมพันธ์ระหว่างเพื่อนร่วมห้องผ่านกลยุทธ์การแทรกแซงตามบุคลิกภาพ (2023) : ความสัมพันธ์

ระหว่างรูมเมทมีความสำคัญมากต่อความเป็นอยู่ที่ดี และความสัมพันธ์ระหว่างรูมเมทที่เข้ากันได้มีศักยภาพในการสร้างสภาพแวดล้อมที่บุคคลสามารถเติบโตได้ทั้งในด้านวิชาการ ชีวิตทางสังคม และชีวิตการทำงาน เอกสารที่ตามมาจะกล่าวถึงประเด็นที่เกี่ยวข้องกับผู้ถือผลประโยชน์ในพื้นที่อยู่อาศัยร่วมกัน โดยมุ่งหวังที่จะกำหนดกลยุทธ์การแทรกแซงที่รูมเมทสามารถใช้เพื่อเสริมสร้างความสัมพันธ์ระหว่างรูมเมทและหลักเลี้ยงความขัดแย้งระหว่างรูมเมท เอกสารนี้จะกล่าวถึงความสำคัญของความสัมพันธ์ระหว่างรูมเมท ปัจจัยที่มีผลต่อความสัมพันธ์ระหว่างรูมเมท สาเหตุของความขัดแย้งระหว่างรูมเมท และผลกระทบต่อชีวิตของบุคคล นอกจากนี้ ยังมีการหารือเกี่ยวกับลักษณะบุคลิกภาพที่ก่อให้เกิดความสัมพันธ์ระหว่างรูมเมทที่ดีด้วย เอกสารนี้มาจากการศึกษาวรรณกรรมที่มีอยู่อย่างเจาะลึก และมุ่งมั่นที่จะเพิ่มความเป็นอยู่ที่ดีของบุคคลโดยจัดการกับความกังวลของรูมเมท และปรับปรุงความสัมพันธ์ระหว่างรูมเมท [9]

## บทที่ 3

### วิธีการออกแบบ

#### 3.1 ขั้นตอนการออกแบบ

##### 3.1.1 Software Requirement Specification (SRS)

ตารางที่ 3.1 ความต้องการเชิงธุรกิจ (Business Requirement)

No.	Description
BR.1	เพื่อเป็นตัวช่วยแนะนำรูปเมทที่เหมาะสมกับผู้ใช้งาน
	BR.1.1 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้
BR.2	เพื่อช่วยให้เกิดความสะดวกรวดเร็ว ประหยัดเวลาในการค้นหารูปเมท
	BR.2.1 ระบบกรองการค้นหาข้อมูล
BR.3	เพื่อสร้างและพัฒนาระบบเฉพาะทางสำหรับการค้นหารูปเมท
	BR.3.1 ระบบสมัครสมาชิกและลงชื่อใช้งาน
	BR.3.2 จัดเก็บข้อมูลผู้ใช้
	BR.3.3 รายละเอียดโปรดไฟล์ผู้ใช้
	BR.3.4 ระบบการจับคู่
	BR.3.5 ระบบแจ้งเตือน
	BR.3.6 ระบบแจ้งปัญหา
	BR.3.7 ระบบให้คะแนนความพึงพอใจ
	BR.3.8 ระบบผู้ดูแลระบบ
	BR.3.9 การประเมินผล

จากตารางที่ 3.1 แสดงความต้องการเชิงธุรกิจ หรือ Business Requirement ระหว่างผู้ใช้งานเว็บแอปพลิเคชันหารูปเมทโดยใช้อัลกอริทึม K-NN โดยสามารถแบ่งเป็นหัวข้อตามวัตถุประสงค์ของโครงการได้ดังหัวข้อใหญ่ทั้ง 3 คือ

- 1) เพื่อเป็นตัวช่วยแนะนำรูปเมทที่เหมาะสมกับผู้ใช้งาน
- 2) เพื่อช่วยให้เกิดความสะดวกรวดเร็ว ประหยัดเวลาในการค้นหารูปเมท
- 3) เพื่อสร้างและพัฒนาระบบเฉพาะทางสำหรับการค้นหารูปเมท ซึ่งรายละเอียดสามารถ

ดูได้ในคอลัมน์ Description

### ตารางที่ 3.2 ความต้องการของระบบ (System Requirements)

Key Requirement	Status	ตรงกับ หัวข้อ BR
SRS.1 ระบบสมัครสมาชิกและลงชื่อใช้งาน		
SRS.1.1 สามารถสมัครสมาชิกด้วย Email และรหัสผ่านได้		BR.3
SRS.1.2 สามารถลงทะเบียนเข้าใช้งานระบบด้วย Email และรหัสผ่านได้		BR.3
SRS.1.3 สามารถลบบัญชีการได้		BR.3
SRS.1.4 สามารถเปลี่ยนรหัสผ่านได้		BR.3
SRS.2 จัดเก็บข้อมูลผู้ใช้		
SRS.2.1 สามารถจัดเก็บข้อมูลส่วนตัวของผู้ใช้ได้ <ul style="list-style-type: none"> <li>2.1.1 เก็บข้อมูลที่สามารถระบุตัวตน เช่น ชื่อ อายุ เพศและรูป</li> <li>2.1.2 เก็บข้อมูลเกี่ยวกับการศึกษา หอพักและยานพาหนะ</li> </ul>		BR.3
SRS.2.2 สามารถจัดเก็บข้อมูลักษณะนิสัยผู้ใช้ได้ <ul style="list-style-type: none"> <li>2.2.1 ช่วงเวลาอน</li> <li>2.2.2 ความสะอาด</li> <li>2.2.3 การใช้เสียง</li> <li>2.2.4 การสูบบุหรี่</li> <li>2.2.5 การใช้งานเครื่องปรับอากาศ</li> <li>2.2.6 การเที่ยวกลางคืน</li> </ul>		BR.3
SRS.2.3 สามารถแก้ไขรายละเอียดข้อมูลในโปรไฟล์ของผู้ใช้ได้		BR.3
SRS.3 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้		
SRS.3.1 สามารถนำข้อมูลผู้ใช้มาเปรียบเทียบได้		BR.1
SRS.3.2 สามารถคำนวณรูปแบบร้อยละความใกล้เคียงของข้อมูลได้		BR.1
SRS.3.3 สามารถแสดงผลผู้ใช้ที่เข้ากันได้ตามลำดับจากมากไปน้อยได้		BR.1
SRS.4 รายละเอียดโปรไฟล์ผู้ใช้		
SRS.4.1 แสดงรูปโปรไฟล์ผู้ใช้ได้		BR.3
SRS.4.2 แสดงรายละเอียดลักษณะนิสัยผู้ใช้ได้		BR.3
SRS.4.3 สามารถเน้นข้อมูลในส่วนที่ตรงกันให้เด่นชัดขึ้นได้		BR.3
SRS.5 ระบบกรองการค้นหาข้อมูล		
SRS.5.1 สามารถกรองการค้นหาด้วยตำแหน่งของผู้ใช้ได้		BR.2
SRS.5.2 สามารถกรองการค้นหาด้วยลักษณะนิสัยตามต้องการได้		BR.2

### ตารางที่ 3.2 ความต้องการของระบบ (System Requirements) (ต่อ)

Key Requirement	Status	ตรงกับหัวข้อ BR
SRS.6 ระบบการจับคู่		
SRS.6.1 สามารถกดถูกใจผู้ใช้รายอื่นได้		BR.3
SRS.6.2 แสดงผู้ใช้ที่ได้กดถูกใจไว้ได้		BR.3
SRS.6.3 สามารถถอนการกดถูกใจได้		BR.3
SRS.6.4 แสดงรูปเมทที่จับคู่สำเร็จเมื่อทั้งสองฝ่ายกดถูกใจทั้งคู่		BR.3
SRS.7 ระบบแจ้งเตือน		
SRS.7.1 สามารถแจ้งเตือนให้ผู้ใช้เมื่อจับคู่สำเร็จ		BR.3
SRS.8 ระบบรายงาน		
SRS.8.1 สามารถรายงานผู้ใช้ที่ไม่พึงประสงค์ได้		BR.3
SRS.8.2 สามารถรายงานปัญหาเกี่ยวกับระบบการใช้งานได้		BR.3
SRS.8.3 สามารถแนะนำเกี่ยวกับระบบได้		BR.3
SRS.9 ระบบให้คะแนนความพึงพอใจ		
SRS.9.1 สามารถให้คะแนนความพึงพอใจระบบแนะนำรูปเมทในหน้าหลัก		BR.3
SRS.9.2 สามารถให้คะแนนความพึงพอใจต่อเว็บไซต์ได้		BR.3
SRS.10 ระบบผู้ดูแลระบบ		
SRS.10.1 สามารถรับรายงานจากผู้ใช้ได้		BR.3
SRS.10.2 สามารถตรวจสอบผู้ใช้ที่ไม่พึงประสงค์ได้		BR.3
SRS.10.3 สามารถปรับสถานะบัญชีผู้ใช้ได้ (เปิดใช้งานหรือระงับบัญชี)		BR.3
SRS.11 การประเมินผล		
SRS.11.1 เชิงปริมาณ <ul style="list-style-type: none"> <li data-bbox="287 1646 1097 1724">11.1.1 คะแนนความพึงพอใจระบบการแนะนำรูปเมทในหน้าหลักเฉลี่ยมากกว่า 70%</li> <li data-bbox="287 1724 1097 1803">11.1.2 คะแนนความพึงพอใจการแสดงผลข้อมูลบุคคลเฉลี่ยมากกว่า 70%</li> <li data-bbox="287 1803 1097 1877">11.1.3 เว็บแอปพลิเคชันทำงานได้ตามขอบเขตมากกว่า 80%</li> </ul>		BR.3

ตารางที่ 3.2 ความต้องการของระบบ (System Requirements) (ต่อ)

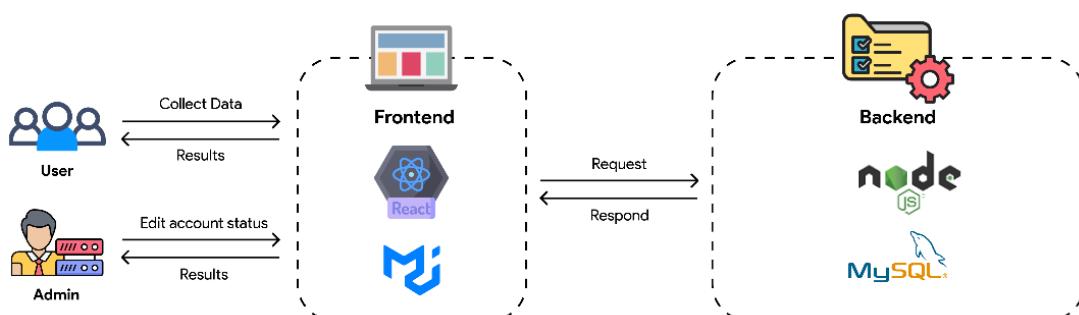
Key Requirement	Status	ตรงกับหัวข้อ BR
SRS.11.2 เชิงคุณภาพ 11.2.1 ด้านความถูกต้องและเหมาะสมของเนื้อหาบนเว็บไซต์ โดยทดสอบใช้งานสร้างบัญชีตลอดจนถึงขั้นตอนการจับคู่ขั้นต่ำ 10 ครั้ง พบร่วมระบบทำงานถูกต้องไม่น้อยกว่า 80%		BR.3

จากตารางที่ 3.2 ความต้องการของระบบ หรือ System Requirements แสดงความต้องการของระบบที่ทำการแต่งกายอย่างมาจากการวัดถูประสงค์ของโครงงาน และ Business Requirement โดย R หมายถึง ความต้องการของระบบประกอบด้วยส่วนใดบ้าง และ B หมายถึง อ้างอิงมาจาก ตารางที่ 3.1 ความต้องการเชิงธุรกิจ (Business Requirement) ข้อใด

### 3.2 แผนภาพรวมของระบบ

#### 3.2.1 System Architecture

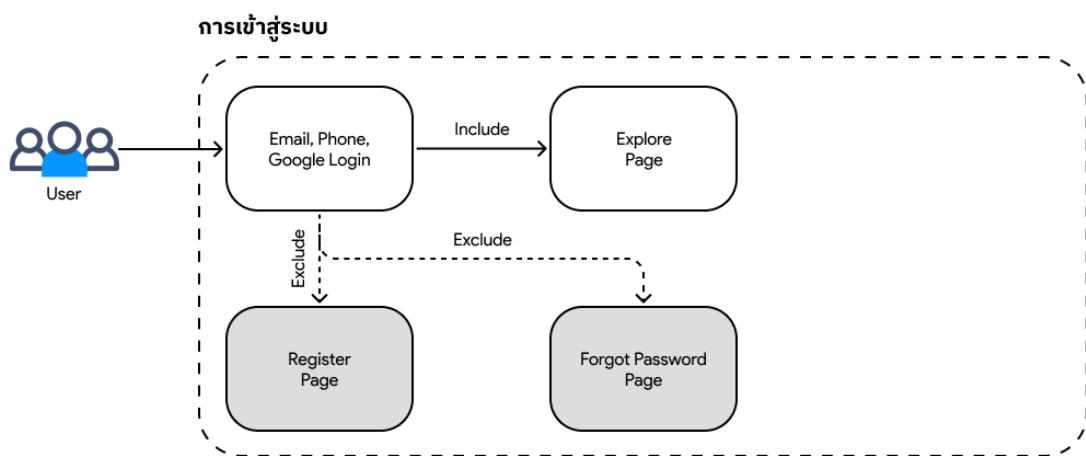
เว็บแอปพลิเคชันหารูមเมทโดยใช้อัลกอริทึม K-NN เป็นระบบที่จะแนะนำคนที่คาดว่ามีความเข้ากันอ้างอิงตามข้อมูลที่ผู้ใช้กรอกเข้ามาในระบบ โดยสามารถแสดงผลแนะนำคนที่คาดว่ามีความเข้ากันได้มากไปน้อยเรียงตามลำดับ และแสดงรายละเอียดข้อมูลลักษณะนิสัยส่วนบุคคล โดยละเอียด ระบบแนะนำสามารถลดความผิดพลาดในการแลกเปลี่ยนข้อมูลที่อาจไม่ครบถ้วนและไม่ซัดเจน อีกทั้งยังช่วยเพิ่มความสะดวกในการแลกเปลี่ยนข้อมูล เกิดความรวดเร็วในการใช้งาน



รูปที่ 3.1 System Diagram

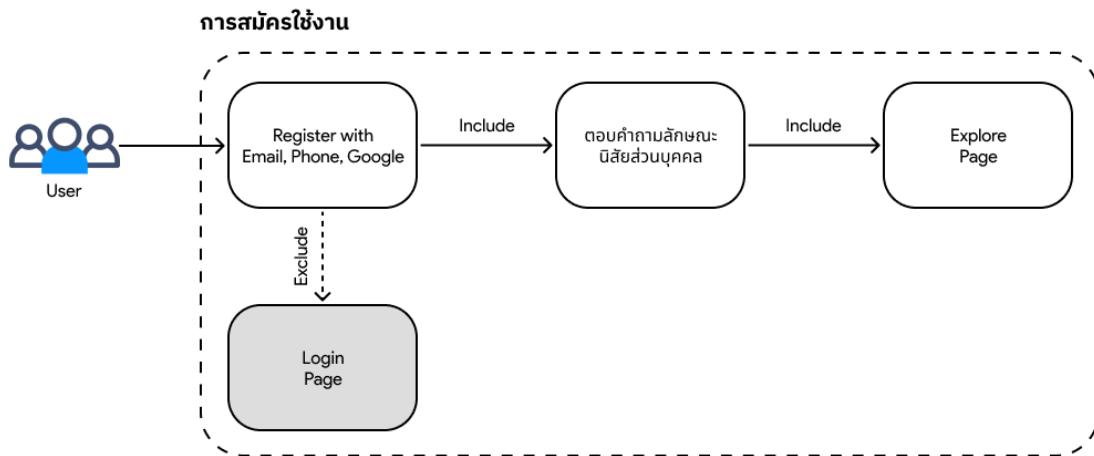
### 3.2.2 แผนภาพการทำงานของระบบ (Use Case)

เว็บแอปพลิเคชันหารูមเมทโดยใช้อัลกอริทึม K-NN เป็นระบบที่จะแนะนำคนที่คิดว่ามีความเข้ากันอ้างอิงตามข้อมูลที่ผู้ใช้กรอกเข้ามาในระบบ โดยสามารถแสดงผลแนะนำคนที่คิดว่ามีความเข้ากันได้จากมากไปน้อยเรียงตามลำดับ และแสดงรายละเอียดข้อมูลลักษณะนิสัยส่วนบุคคล โดยละเอียด ระบบแนะนำสามารถลดความผิดพลาดในการเลือกเปลี่ยนข้อมูลที่อาจไม่ครบถ้วนและไม่ชัดเจน อีกทั้งยังช่วยเพิ่มความสะดวกในการเลือกเปลี่ยนข้อมูล เกิดความรวดเร็วในการใช้งาน



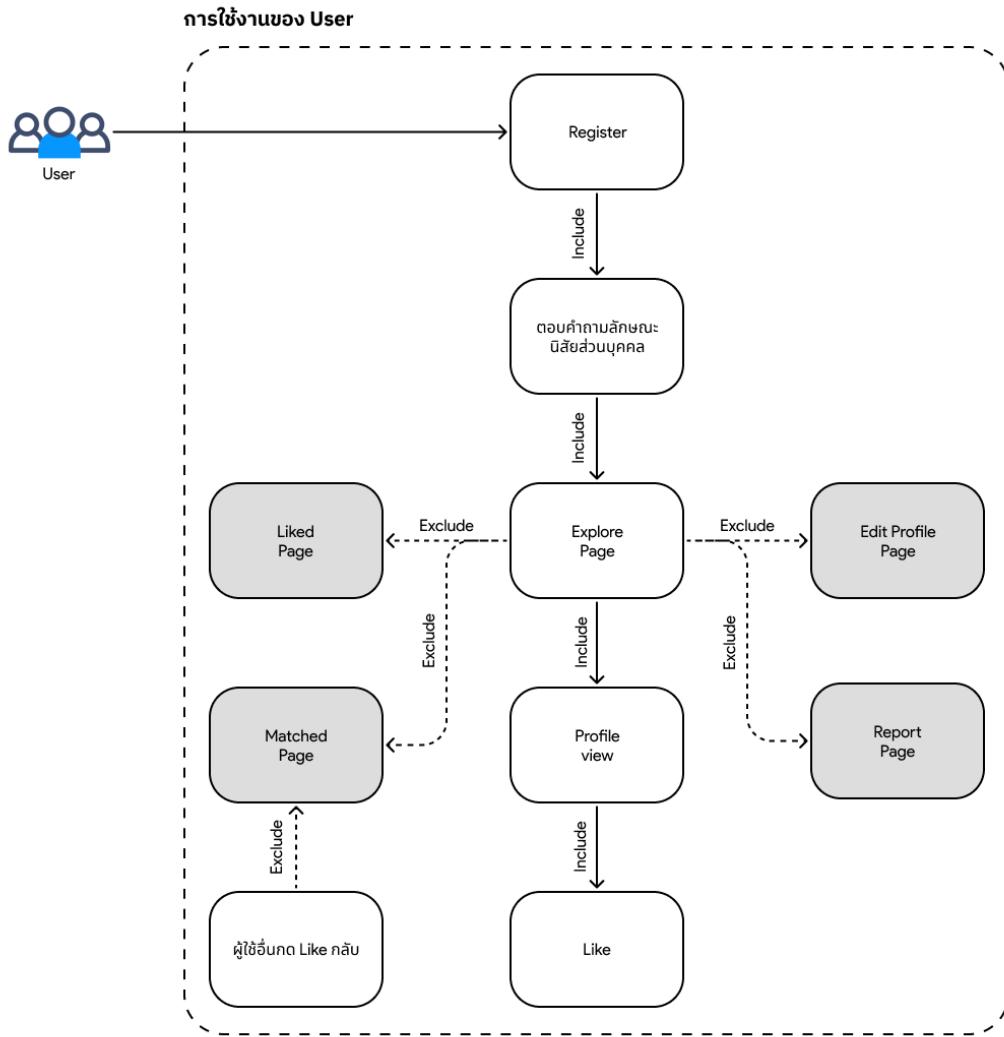
รูปที่ 3.2 Use Case การเข้าสู่ระบบ

จากรูปที่ 3.2 Use Case การเข้าสู่ระบบ แสดงขั้นตอนการเข้าสู่ระบบของผู้ใช้งานนี้ เมื่อผู้ใช้งาน ทำได้ทำการเปิดแอปพลิเคชัน ระบบจะนำผู้ใช้ไปยังหน้าจอการ Login หากผู้ใช้ยังไม่ได้ลงทะเบียน สามารถกดปุ่มสมัครสมาชิกเพื่อไปยังหน้า Register Page หรือหน้าลงทะเบียน หลังจากนั้นให้ผู้ใช้ทำการลงทะเบียนด้วยข้อมูลของตนเองให้เรียบร้อย จึงจะสามารถเข้าสู่ระบบของแอปพลิเคชันได้ หรือผู้ใช้สามารถที่จะลงทะเบียนเพื่อเข้าสู่ระบบด้วยบัญชี Google ได้เช่นกัน ก็จะสามารถเข้าสู่ระบบและใช้งานได้ทันที ถ้าหากผู้ใช้ลืมรหัสผ่าน สามารถกดปุ่ม ลืมรหัสผ่าน ระบบจะนำผู้ใช้ไปยังหน้า Forgot Password Page เพื่อให้ผู้ใช้ได้ทำการเปลี่ยนแปลงรหัสผ่านของตนเองเพื่อเข้าสู่ระบบต่อไป



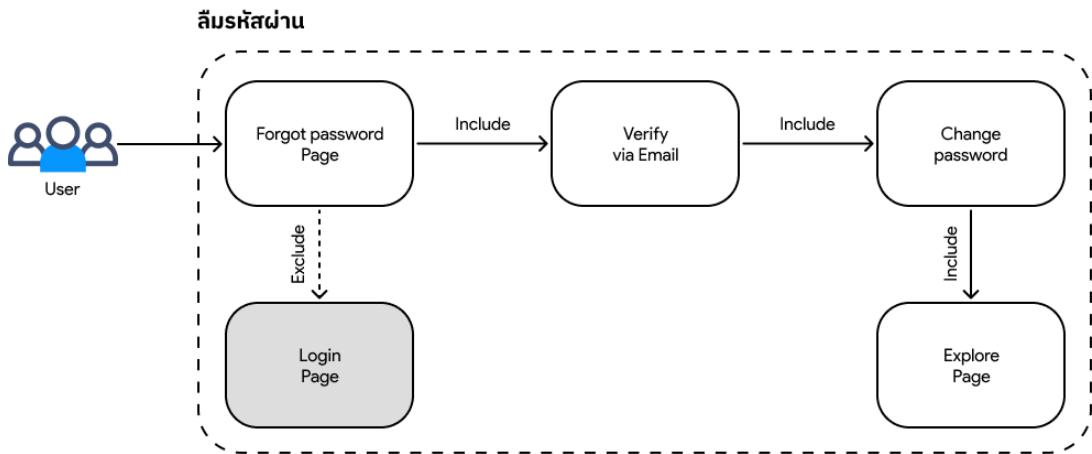
รูปที่ 3.3 Use Case การสมัครสมาชิก

จากรูปที่ 3.3 Use Case แผนภาพการสมัครสมาชิก หลังจากทำการสมัครสมาชิก เรียบร้อยแล้ว ระบบจะนำผู้ใช้ไปยังหน้าจอที่จะให้ผู้ใช้กรอกแบบสอบถาม โดยจะเป็นคำถามที่เป็นตัวเลือกให้ผู้ใช้ได้ทำการเลือกตอบ ที่เกี่ยวข้องกับลักษณะนิสัยและกิจวัตรประจำวันของผู้ใช้ โดยจำเป็นที่จะต้องกรอกแบบสอบถามให้ครบถ้วน เพื่อระบบจะนำข้อมูลไปเบริยบเทียบกับผู้ใช้รายอื่น ก่อนที่จะเข้าใช้งานระบบ



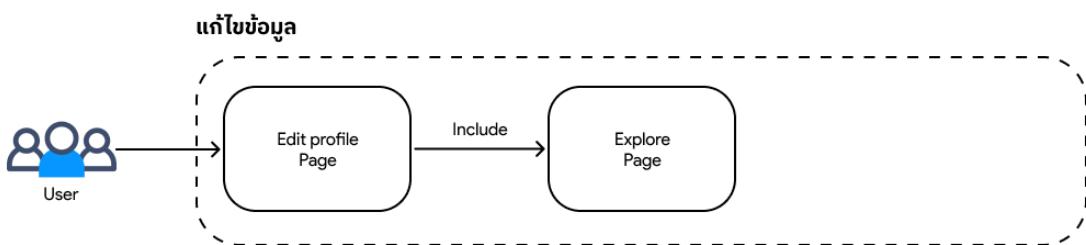
รูปที่ 3.4 Use Case การใช้งานของ User

จากรูปที่ 3.4 Use Case การใช้งานเว็บแอปพลิเคชันของผู้ใช้งาน โดยแสดงขั้นตอนการใช้โดยรวม ซึ่งเริ่มจากการที่ผู้ใช้งานได้ลงทะเบียนเข้าสู่ระบบ ระบบจะนำผู้ใช้งานไปยังหน้าจอที่จะให้ผู้ใช้กรอกแบบสอบถาม ระบบจะนำข้อมูลของผู้ใช้ไปเปรียบเทียบกับผู้ใช้รายอื่น แล้วนำผลลัพธ์ที่ได้มาแสดงในหน้า Explore Page หรือหน้าหลัก ที่จะแสดงผู้ใช้ที่มีลักษณะนิสัยใกล้เคียงกัน โดยเรียงลำดับความใกล้เคียงจากมากไปน้อย ผู้ใช้สามารถที่จะกดเข้าไปดูโปรไฟล์เพื่อดูรายละเอียดลักษณะนิสัยหรือใช้เชิญลิ้มเดียของผู้ใช้รายอื่นได้ หากสนใจ หรือถูกใจรูปเมทคนไหน ผู้ใช้สามารถที่จะกด Like และจะแสดงผู้ใช้ที่ได้ทำการกด Like ไปแล้วในหน้า Liked Page และเมื่อรูปเมทคนนั้นได้กด Like กลับ ทั้งคู่จะ Match กัน สามารถดูได้ในหน้า Matched Page เมื่อ Match สำเร็จ ในหน้า Profile View จะแสดงเบอร์โทรศัพท์มือถือเพื่อให้ผู้ใช้ติดต่อกันได้ง่ายขึ้น โดยที่ผู้ใช้สามารถที่จะแก้ไขข้อมูลที่จะแสดงในโปรไฟล์ได้ใน Edit Profile Page ได้ หรือหากพบเจอปัญหาเกี่ยวกับผู้ใช้ที่ไม่เหมาะสม หรือเจอปัญหาเกี่ยวกับระบบ สามารถแจ้งปัญหาเหล่านั้นได้ในหน้า Report Page



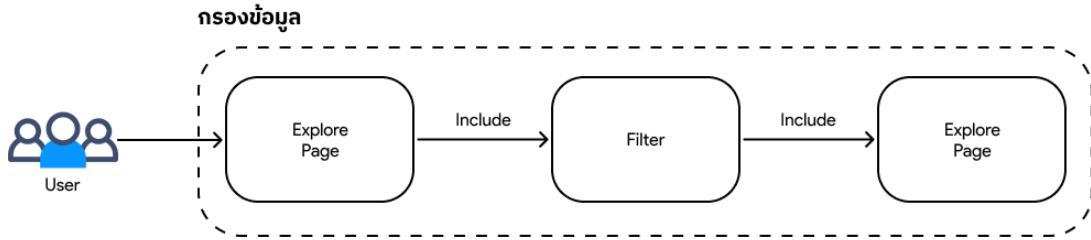
รูปที่ 3.5 Use Case การเปลี่ยนรหัสผ่าน

จากรูปที่ 3.5 Use Case การเปลี่ยนรหัสผ่าน แสดงระบบการเปลี่ยนรหัสผ่านเมื่อผู้ใช้งานทำการกดปุ่มลืมรหัสผ่าน ระบบจะนำผู้ใช้งานมายังหน้า Forgot Password Page หรือ หน้าลืมรหัสผ่าน โดยผู้ใช้งานจะได้รับข้อความพร้อมกับลิงค์ที่นำไปหน้าของการเปลี่ยนรหัสผ่าน ไปยัง Email ที่ได้ทำการลงทะเบียนไว้กับทางแอปพลิเคชัน หลังจากนั้นผู้ใช้งานสามารถกรอกข้อมูลที่ต้องการและกดปุ่มยืนยันเพื่อเปลี่ยนแปลงรหัสผ่านของตนเอง จากนั้นระบบจะนำผู้ใช้งานไปยังหน้าการเข้าสู่ระบบ หรือ Login โดยอัตโนมัติ



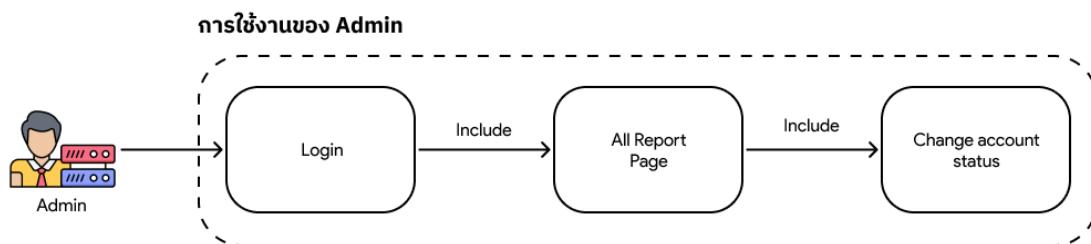
รูปที่ 3.6 Use Case การแก้ไขโปรไฟล์

จากรูปที่ 3.6 Use Case การแก้ไขโปรไฟล์ จะแสดงระบบการแก้ไขข้อมูลโปรไฟล์ของผู้ใช้ โดยที่หากผู้ใช้ต้องการจะเปลี่ยนแปลงข้อมูล ที่ผู้ใช้ได้ทำการเลือกในแบบสอบถาม สามารถที่จะแก้ไขได้ในหน้า Edit Profile Page โดยจะมีให้แก้ไขลักษณะนิสัย และข้อมูลส่วนตัว รวมไปถึงเพิ่มช่องทางการติดต่อ เช่น เบอร์โทรศัพท์ และโซเชียลมีเดีย เป็นต้น เมื่อแก้ไขเสร็จสิ้นระบบจะทำการอัพเดทข้อมูลไปยังหน้า Explore Page



รูปที่ 3.7 Use Case การกรองข้อมูล

จากรูปที่ 3.7 Use Case การกรองข้อมูล จะแสดงระบบการกรองข้อมูลที่จะแสดงรูปเมท ในหน้า Explore Page โดยสามารถแบ่งการกรองได้ 2 ประเภท ได้แก่ กรองข้อมูลตามลำดับความใกล้เคียงของผู้ใช้จำนวนมากไปน้อย และ กรองข้อมูลตามตำแหน่งที่อยู่ใกล้เคียงกับผู้ใช้



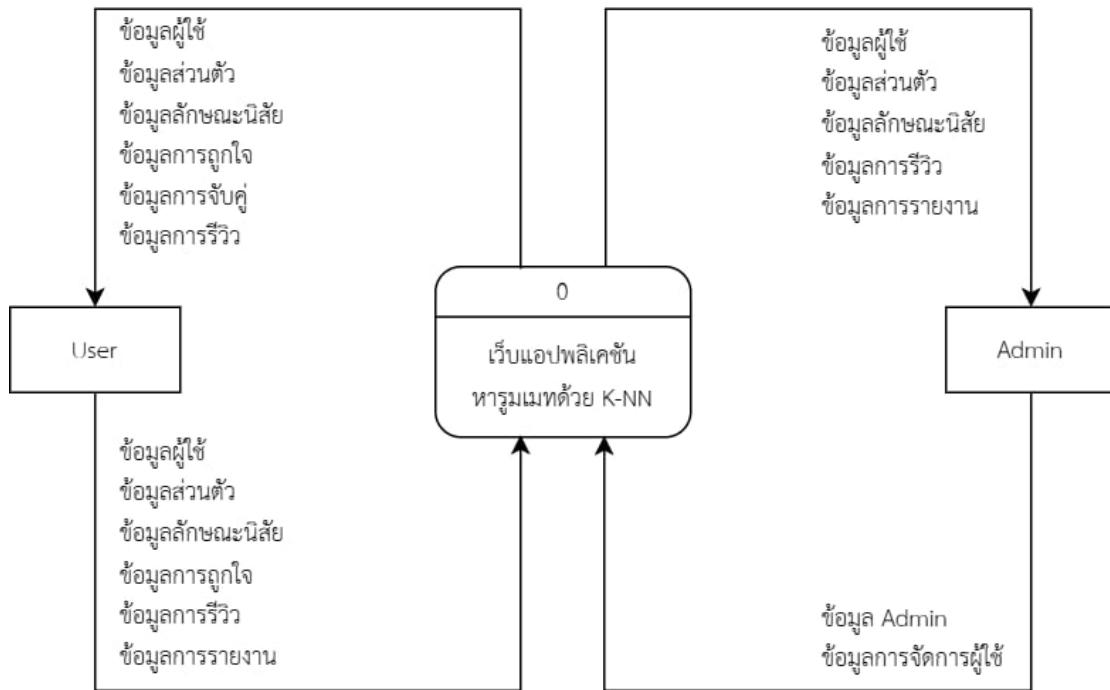
รูปที่ 3.8 Use Case การใช้งานของ Admin

จากรูปที่ 3.8 Use Case การใช้งานระบบของ Admin โดยจะแสดงระบบการใช้งานเมื่อ Admin ได้ Login เข้าสู่ระบบ โดยที่จะสามารถรับ Report หรือ ปัญหาที่ผู้ใช้ได้แจ้งมา ในหน้า All Report Page จะมีการ Report แบ่งออกเป็น 3 ประเภท ได้แก่

- 1) ปัญหาเกี่ยวกับผู้ใช้ที่ไม่เหมาะสม เช่น ผู้ใช้ที่มีรูปไฟล์ที่ลามกหรืออนาจาร และผู้ใช้ที่มีข้อมูลเชิงล่อแหลม เป็นต้น Admin สามารถตัดไฟล์นั้นเพื่อตรวจสอบตามที่ได้รับแจ้ง หากเป็นความจริง Admin สามารถทำการจัดการหรือระงับบัญชีนั้นได้ทันที
- 2) ปัญหาเกี่ยวกับระบบ เช่น เซิร์ฟเวอร์ทำงานล่าช้า เซิร์ฟเวอร์ทำงานผิดปกติ การแสดงข้อมูลผิดพลาด เป็นต้น
- 3) ข้อเสนอแนะเกี่ยวกับระบบการทำงานของเว็บแอปพลิเคชัน

### 3.3 ขั้นตอนการออกแบบฐานข้อมูล

#### 3.3.1 Context Diagram (DFD Level 0)



รูปที่ 3.9 Context Diagram (DFD Level 0)

จากรูปที่ 3.9 Process 0 : ของเว็บแอปพลิเคชันหารูมเมทด้วย K-NN มี Entity ที่เกี่ยวข้อง 2 ประเภท ได้แก่ User และ Admin โดย Process 0 จะมีการส่งข้อมูลผู้ใช้ ข้อมูลส่วนตัว ข้อมูลลักษณะนิสัย ข้อมูลการถูกใจ ข้อมูลการจับคู่ ข้อมูลการรีวิว ให้กับ User และรับข้อมูลผู้ใช้ ข้อมูลส่วนตัว ข้อมูลลักษณะนิสัย ข้อมูลการถูกใจ ข้อมูลการจับคู่ ข้อมูลการรีวิว ข้อมูลการรายงาน จาก User จากนั้นส่งข้อมูลผู้ใช้ ข้อมูลส่วนตัว ข้อมูลลักษณะนิสัย ข้อมูลการรีวิว ข้อมูลการรายงาน ให้กับ Admin และรับข้อมูล Admin ข้อมูลการรายงาน ข้อมูลการจัดการผู้ใช้ จาก Admin

#### 3.3.1.1 User หรือผู้ใช้งาน

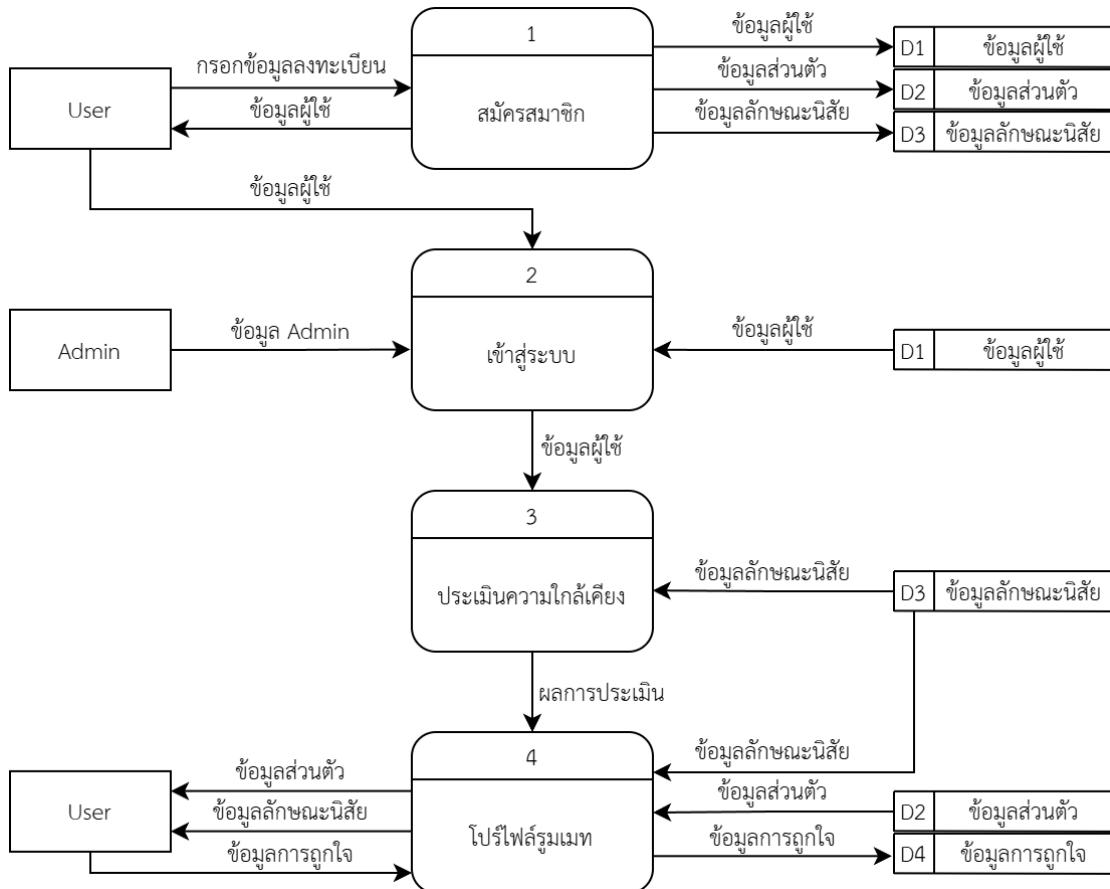
- 1) ลงทะเบียนสมัครสมาชิกของระบบ
- 2) กรอกข้อมูลส่วนตัวและข้อมูลลักษณะนิสัย
- 3) เข้าสู่ระบบบใช้งานเว็บแอปพลิเคชัน
- 4) ระบบทำการประเมินและแสดงความใกล้เคียงของผู้ใช้กับผู้ใช้อื่นที่อยู่ในระบบ
- 5) ค้นหารูมเมทที่สนใจและกดถูกใจ

- 6) รองรูมเมทที่สนใจกดถูกใจกลับจึงจะจับคู่สำเร็จ
- 7) เมื่อจับคู่สำเร็จระบบจะแสดงข้อมูลการติดต่อของรูมเมทในໂປຣໄຟລ໌
- 8) ให้คณะนั้นระบบการหารูมเมท
- 9) ส่งข้อมูลรายงานปัญหาการใช้งานให้ผู้ดูแลระบบ

### 3.3.1.2 Admin หรือผู้ดูแลระบบ

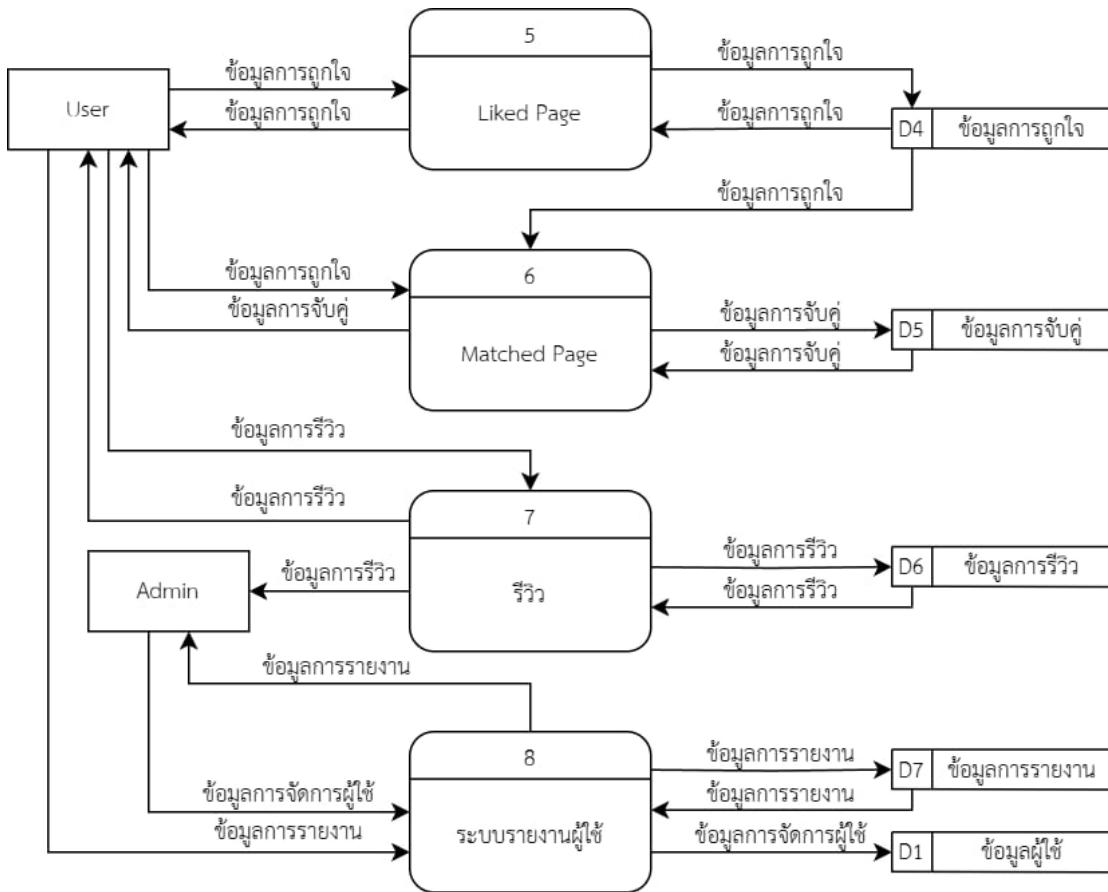
- 1) ลงทะเบียนเข้าสู่ระบบ
- 2) แสดงแดชบอร์ดของผู้ดูแลระบบ
- 3) ตรวจสอบรายงานและการรีวิวระบบจากผู้ใช้งาน
- 4) จัดการบัญชีผู้ใช้ตามการรายงานอย่างเหมาะสม

### 3.3.2 Data Flow Diagram (DFD Level 1)



รูปที่ 3.10 Data Flow Diagram (DFD Level 1) Process 1-4

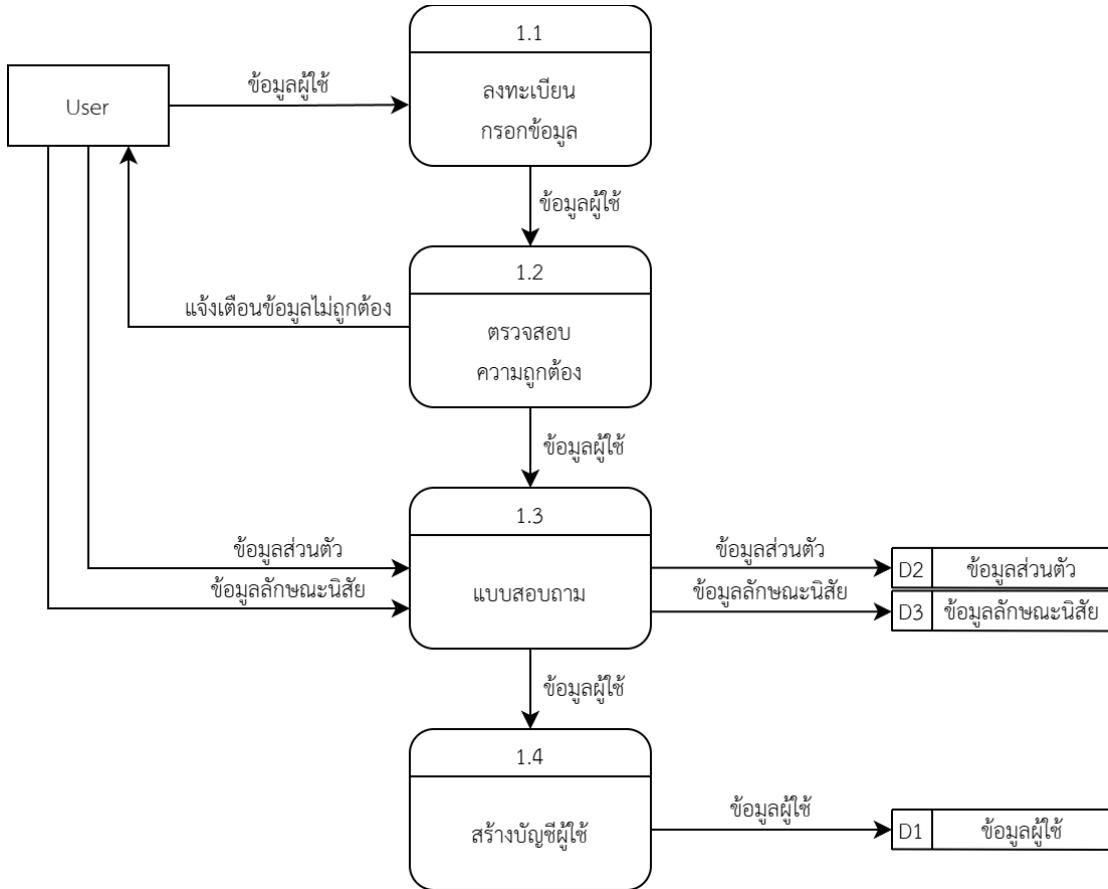
จากรูปที่ 3.10 Process 1: สมัครสมาชิก ระบบจะรับข้อมูลของ Entity User จากการกรอกข้อมูลลงทะเบียน ข้อมูลเหล่านี้จะแยกออกเป็น 3 ส่วน จะถูกส่งไปเก็บ D1: ข้อมูลผู้ใช้ D2: ข้อมูลส่วนตัว และ D3: ข้อมูลลักษณะนิสัย ข้อมูลผู้ใช้ที่ได้ทำการสมัครสมาชิกสำเร็จจะถูกส่งกลับไปยัง Entity User เพื่อนำไปเข้าสู่ระบบใน Process 2: เข้าสู่ระบบ ระบบจะรับข้อมูลของผู้ใช้จาก Entity User ข้อมูลนี้จะถูกนำไปเปรียบเทียบกับข้อมูลผู้ใช้ที่ถูกเก็บไว้ใน D1: ข้อมูลผู้ใช้ เพื่อตรวจสอบการเข้าสู่ระบบ หลังจากตรวจสอบผ่าน จะสามารถเข้าถึงระบบได้ และจะไปต่อไปยัง Process 3: ประเมินความใกล้เคียง ประเมินความใกล้เคียงของข้อมูลลักษณะนิสัยของผู้ใช้ทั้งหมดในระบบ ซึ่งจะดึงข้อมูลจาก D3: ข้อมูลลักษณะนิสัย ของผู้ใช้ มาเปรียบเทียบกับมาเป็นเปอร์เซ็นต์ความใกล้เคียง จำนวนมากไปน้อย เพื่อนำไปแสดงให้แก่ผู้ใช้ ใน Process 4: ໂປຣີຣົມເມທ ซึ่งจะดึงข้อมูลจาก D2: ข้อมูลส่วนตัว และ D3: ข้อมูลลักษณะนิสัย นำไปแสดงให้ผู้ใช้ ระบบจะแสดงໄປຣີຣົມເມທกอกมาเป็นลำดับความใกล้เคียง ให้ผู้ใช้ได้เลือกดูและกดถูกใจຮູມເມທທີ່ສັນໃຈ ระบบจะรับข้อมูลการถูกใจจากผู้ใช้ที่ได้ทำการกดถูกใจและเก็บข้อมูลไปยัง D4: ข้อมูลการกดถูกใจ



รูปที่ 3.11 Data Flow Diagram (DFD Level 1) Process 5-8

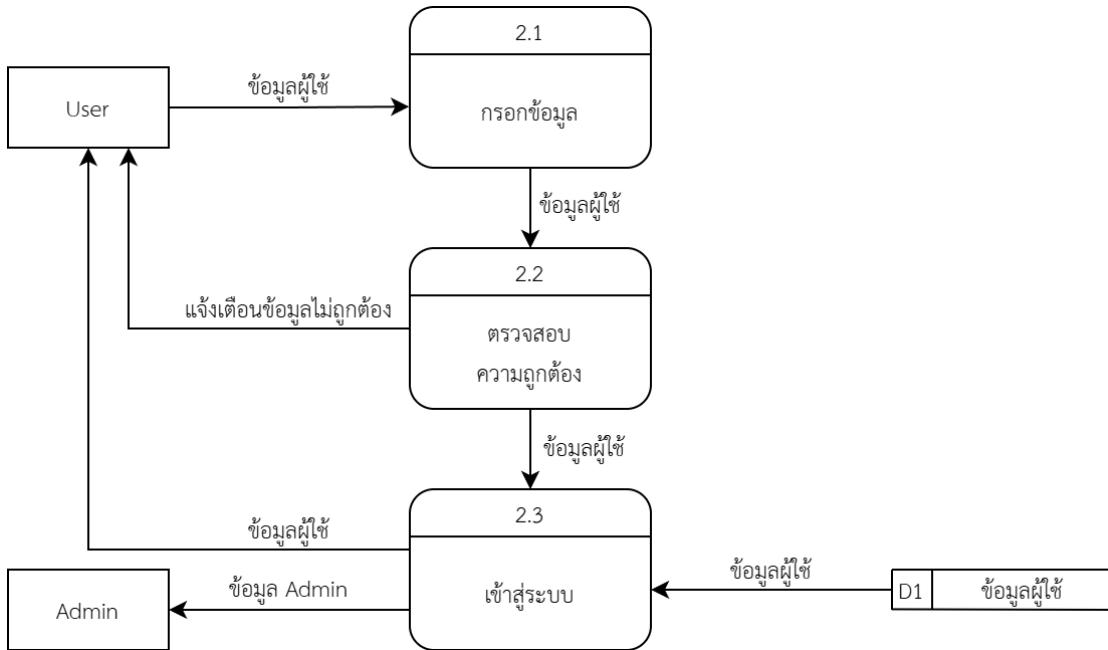
จากรูปที่ 3.11 Process 5: Liked Page ตรวจสอบการกดถูกใจ ซึ่งจะดึงข้อมูลจาก D4: ข้อมูลการถูกใจ ที่ถูกบันทึกไว้ ระบบจะแสดงผู้ใช้ที่ทำการกดถูกใจไว้ ในหน้า Liked Page ข้อมูลจะถูกบันทึกและเก็บไว้ใน D4: ข้อมูลการถูกใจ และส่งต่อข้อมูลไปยัง Process 6: Matched Page ตรวจสอบการกดถูกใจ หากผู้ใช้ทั้ง 2 ฝ่ายได้กดถูกใจซึ่งกันและกัน ระบบจะแสดงผลการจับคู่สำเร็จให้แก่ผู้ใช้ ในหน้า Matched Page โดยข้อมูลการจับคู่นี้จะถูกบันทึกและเก็บไว้ใน D5: ข้อมูลการถูกใจ ในส่วนต่อไป Process 7: รีวิว ระบบรับข้อมูลรีวิวจาก Entity User โดยข้อมูลการรีวินี้จะถูกบันทึกไปยัง D6: ข้อมูลการรีวิว และส่งข้อมูลการรีวิวกลับไปยัง Entity User และ Admin เพื่อการแสดงผล สุดท้าย Process 8: รายงาน รับข้อมูลรายงานจาก Entity User และส่งข้อมูลนั้นไปยัง D7: ข้อมูลการรายงาน และแสดงข้อมูลการรายงานไปยัง Entity Admin เพื่อให้ผู้ดูแลระบบตรวจสอบการรายงานของผู้ใช้ และจัดการบัญชีผู้ใช้อย่างเหมาะสม โดยเก็บข้อมูลการจัดการผู้ใช้ไปยัง D1: ข้อมูลผู้ใช้

### 3.3.3 Data Flow Diagram (DFD Level 2)



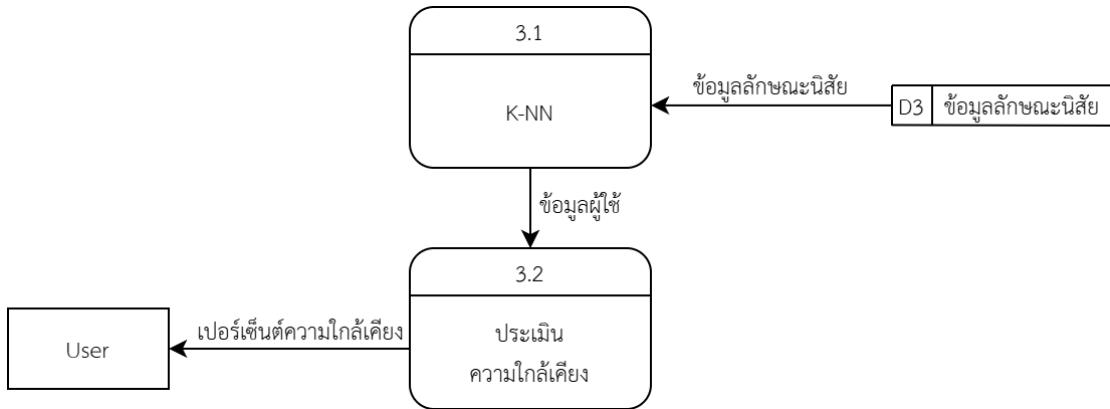
รูปที่ 3.12 Data Flow Diagram (DFD Level 2) Process 1

จากรูปที่ 3.12 เป็นกระบวนการทำงานใน Process 1: สมัครสมาชิก ใน Process 1.1 ระบบจะทำการรับข้อมูลของผู้ใช้ที่ทำการกรอก Username Password และ Email และทำการตรวจสอบความถูกต้องของบัญชี โดยระบบจะตรวจสอบความถูกต้องของข้อมูล เมื่อสมัครสมาชิกเสร็จสิ้น ส่วนต่อไป Process 1.3 ระบบจะมีแบบสอบถามที่ให้ผู้ใช้กรอกข้อมูลส่วนตัวและข้อมูลลักษณะนิสัย ข้อมูลเหล่านี้จะถูกส่งไปเก็บใน D2: ข้อมูลส่วนตัว และ D3: ข้อมูลลักษณะนิสัย และสุดท้าย Process 1.4 เมื่อกรอกข้อมูลต่างๆเสร็จสิ้น ระบบจะสร้างบัญชีผู้ใช้ใหม่ และบันทึกข้อมูลผู้ใช้ลงใน D1: ข้อมูลผู้ใช้ เมื่อบันทึกเสร็จสิ้นผู้ใช้จะสามารถเข้าสู่ระบบ และใช้งานระบบได้



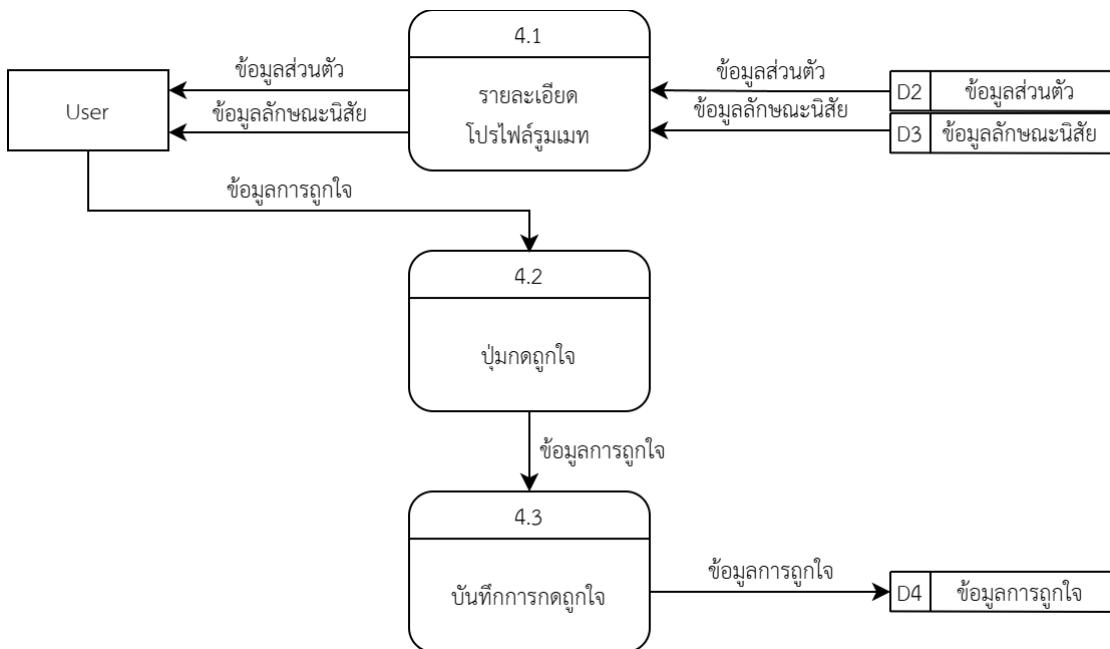
รูปที่ 3.13 Data Flow Diagram (DFD Level 2) Process 2

จากรูปที่ 3.13 เป็นกระบวนการทำงานใน Process 2: เข้าสู่ระบบ ของ User และ Admin โดยเริ่มจาก Process 2.1 กรอกข้อมูลสำหรับเข้าสู่ระบบ เช่น อีเมล รหัสผ่าน หรือบัญชี Google ข้อมูลจะถูกส่งไปยัง Process 2.2 ตรวจสอบความถูกต้องของข้อมูล หากข้อมูลไม่ถูกต้อง ระบบจะแสดงการแจ้งเตือนข้อมูลไม่ถูกต้อง แต่หากข้อมูลถูกต้อง ระบบจะส่งข้อมูลไปยัง Process 2.3 เข้าสู่ระบบ เพื่อให้ผู้ใช้และผู้ดูแลระบบ สามารถเข้าสู่ระบบได้ โดยข้อมูลนี้จะถูกนำไปเปรียบเทียบ กับข้อมูลผู้ใช้ที่ถูกเก็บไว้ใน D1: ข้อมูลผู้ใช้ เพื่อทำการตรวจสอบและยืนยันตัวตน



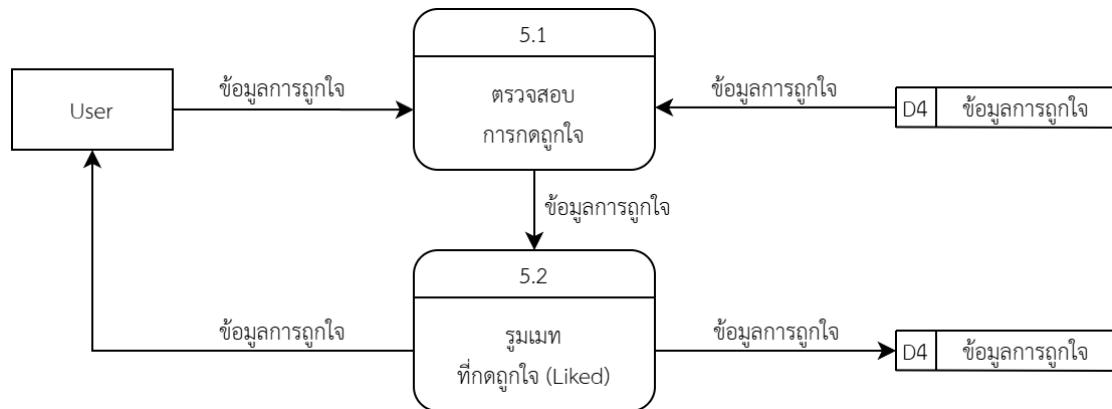
รูปที่ 3.14 Data Flow Diagram (DFD Level 2) Process 3

จากรูปที่ 3.14 เป็นกระบวนการทำงานใน Process 3: ประเมินความใกล้เคียงของข้อมูลลักษณะนิสัยของผู้ใช้ทั้งหมดในระบบ โดยเริ่มจาก Process 3.1 K-NN ซึ่งจะดึงข้อมูลจาก D3: ข้อมูลลักษณะนิสัย ของผู้ใช้ มาเปรียบเทียบใน Process 3.2 ประเมินความใกล้เคียงของผู้ใช้ในระบบ ออกแบบมาเป็นเปอร์เซ็นต์ความใกล้เคียงจากมากไปน้อย เพื่อนำไปแสดงให้แก่ผู้ใช้



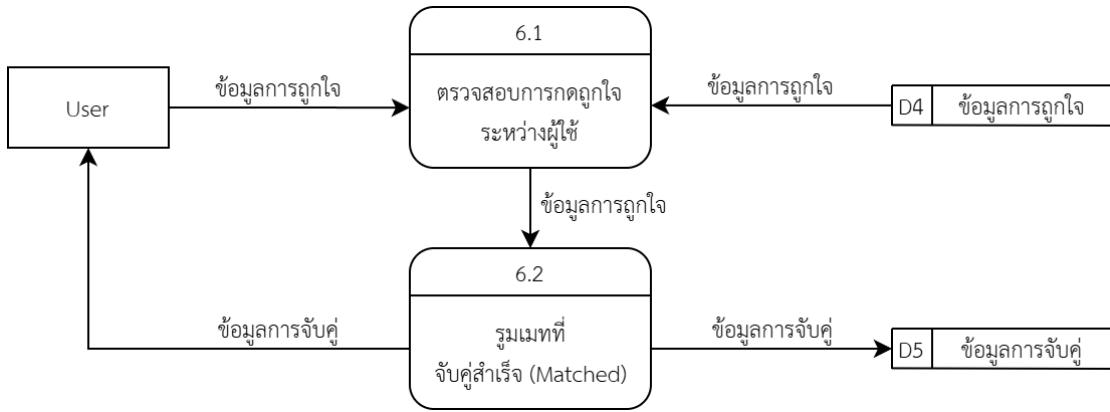
รูปที่ 3.15 Data Flow Diagram (DFD Level 2) Process 4

จากรูปที่ 3.15 เป็นกระบวนการทำงานใน Process 4: โพรไฟล์รูมเมท โดยเริ่มจาก Process 4.1 รายละเอียดโพรไฟล์รูมเมท ซึ่งจะดึงข้อมูลจาก D2: ข้อมูลส่วนตัว และ D3: ข้อมูลลักษณะนิสัย นำไปแสดงให้ผู้ใช้ ระบบจะแสดงโพรไฟล์รูมเมทออกมาเป็นลำดับความใกล้เคียง ให้ผู้ใช้ได้เลือกดูและกดถูกใจรูมเมทที่สนใจ ใน Process 4.2 ปุ่มกดถูกใจ จะรับข้อมูลการถูกใจจากผู้ใช้ ที่ได้ทำการกดถูกใจ ส่งไปยัง Process 4.3 บันทึกการกดถูกใจ เพื่อส่งข้อมูลการถูกใจเก็บไปยัง D4: ข้อมูลการกดถูกใจ



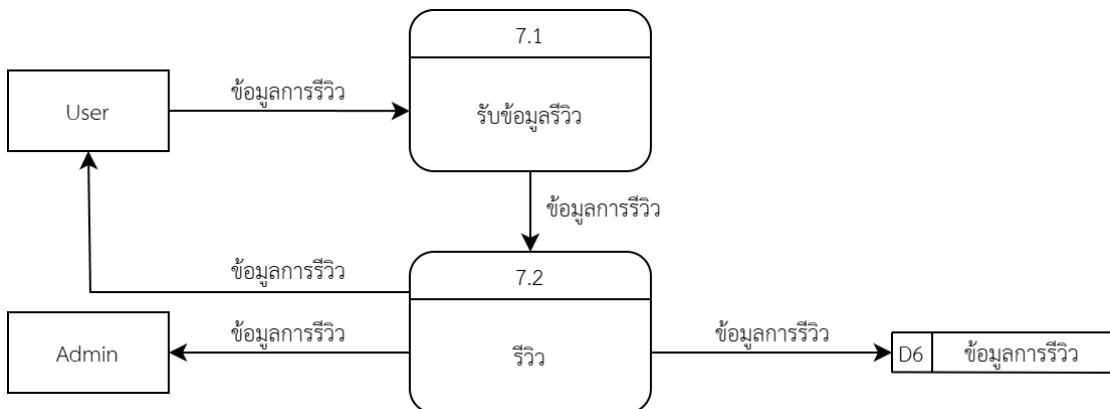
รูปที่ 3.16 Data Flow Diagram (DFD Level 2) Process 5

จากรูปที่ 3.16 เป็นกระบวนการทำงานใน Process 5: Liked Page โดยเริ่มจาก Process 5.1 ตรวจสอบการกดถูกใจ ซึ่งจะดึงข้อมูลจาก D4: ข้อมูลการถูกใจ ที่ถูกบันทึกไว้ หรือผู้ใช้กดยกเลิกถูกใจ จากนั้นส่งต่อไปยัง Process 5.2 รูมเมทที่กดถูกใจ ระบบจะแสดงผู้ใช้ที่ทำการกดถูกใจไว้ ในหน้า Liked Page ข้อมูลจะถูกบันทึกและเก็บไว้ใน D4: ข้อมูลการถูกใจ



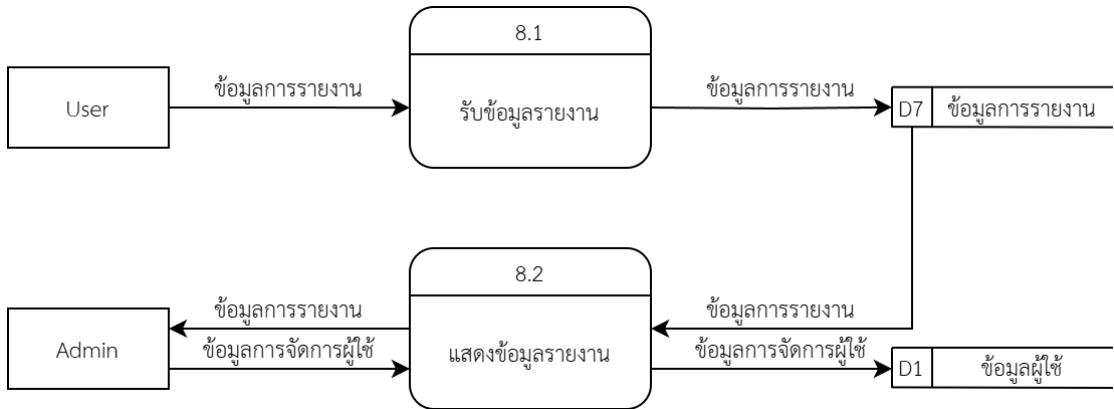
รูปที่ 3.17 Data Flow Diagram (DFD Level 2) Process 6

จากรูปที่ 3.17 เป็นกระบวนการทำงานใน Process 6: Matched Page โดยเริ่มจาก Process 6.1 ตรวจสอบการถูกใจ ซึ่งจะดึงข้อมูลจาก D4: ข้อมูลการถูกใจ ที่ถูกบันทึกไว้ หากผู้ใช้ทั้ง 2 ฝ่ายได้กดถูกใจซึ่งกันและกัน ข้อมูลนั้นส่งต่อไปยัง Process 6.2 รูปเมทที่จับคู่สำเร็จ ระบบจะแสดงผลการจับคู่สำเร็จให้แก่ผู้ใช้ ในหน้า Matched Page ข้อมูลการจับคู่นั้นจะถูกบันทึกและเก็บไว้ใน D5: ข้อมูลการถูกใจ



รูปที่ 3.18 Data Flow Diagram (DFD Level 2) Process 7

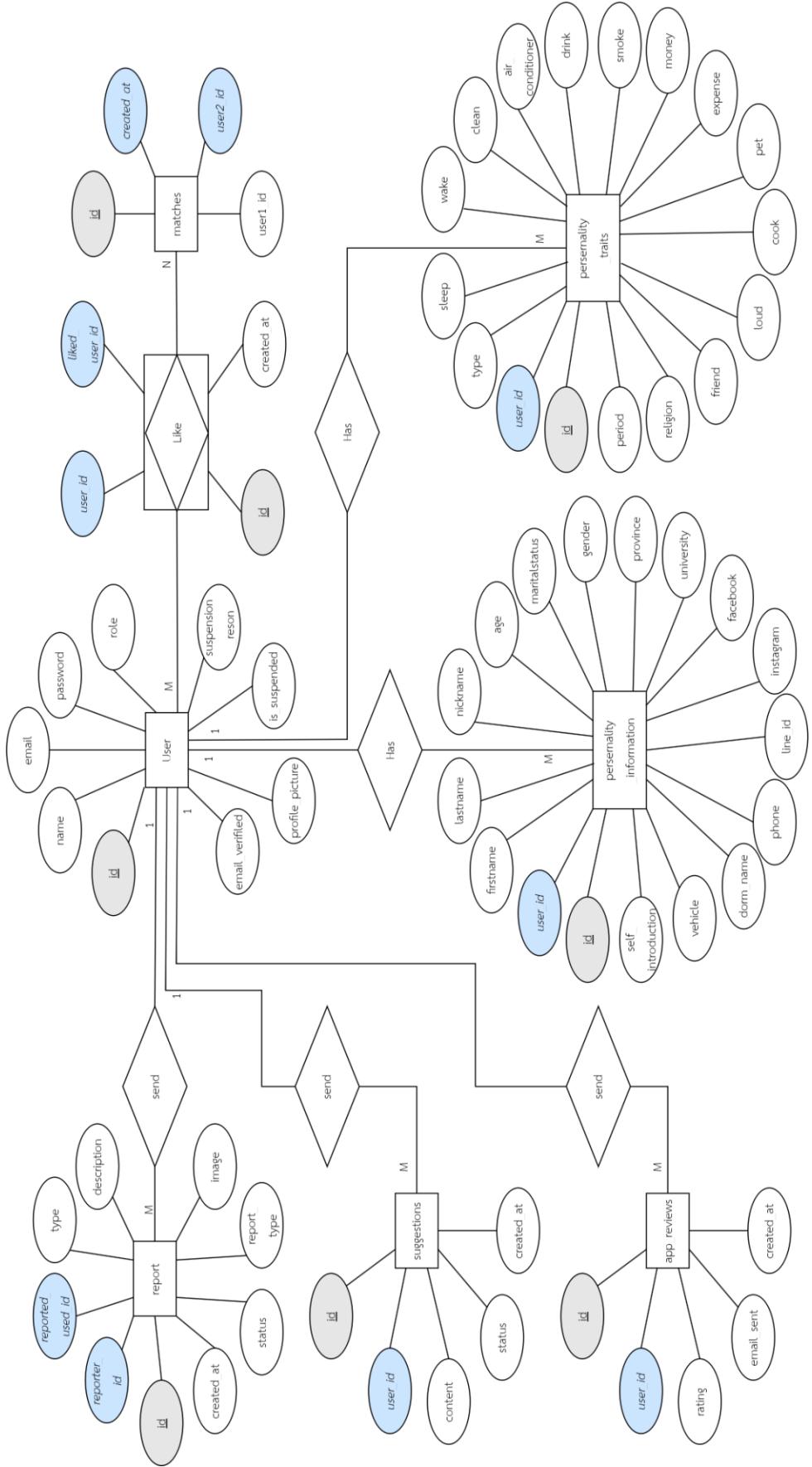
จากรูปที่ 3.18 เป็นกระบวนการทำงานใน Process 7: รีวิว โดยเริ่มจาก Process 7.1 รับ 'ข้อมูลรีวิว' จากผู้ใช้ และส่ง 'ข้อมูลรีวิวนี้' ไปยัง Process 7.2 รีวิว 'ข้อมูลการรีวิวนั้น' ที่ถูกบันทึกไปยัง D6: 'ข้อมูลการรีวิว' และส่ง 'ข้อมูลการรีวิว' กลับไปยัง Entity User และ Admin เพื่อการแสดงผล



รูปที่ 3.19 Data Flow Diagram (DFD Level 2) Process 8

จากรูปที่ 3.19 เป็นกระบวนการทำงานใน Process 8: รายงาน โดยเริ่มจาก Process 8.1 รับข้อมูลรายงานจาก Entity User และส่งข้อมูลนั้นไปยัง D7: ข้อมูลการรายงาน จากนั้น Process 8.2 แสดงข้อมูลการรายงาน ไปยัง Entity Admin เพื่อให้ผู้ดูแลระบบตรวจสอบการรายงานของผู้ใช้ และจัดการบัญชีผู้ใช้อย่างเหมาะสม โดยเก็บข้อมูลการจัดการผู้ใช้ ไปยัง D1: ข้อมูลผู้ใช้

### 3.3.4 การออกแบบ ER Diagram



รูปที่ 3.20 การออกแบบ ER Diagram ของเว็บแอปพลิเคชันหารูปเมห์ด้วย K-NN

จากรูปที่ 3.20 เป็นการออกแบบ ER Diagram หรือ Entity Relation Diagram ของเว็บแอปพลิเคชันหารูปเมด้วย K-NN ประกอบไปด้วย เออนทิตี้ (Entity) จำนวน 7 เออนทิตี้ โดยแต่ละเออนทิตี้จะมีความสัมพันธ์ดังต่อไปนี้

3.3.4.1 เออนทิตี้ users มีความสัมพันธ์กับ เออนทิตี้ personality\_information แบบ 1:M (One to Many) คือ 1 user สามารถมีข้อมูลส่วนบุคคลได้หลายรายการ

3.3.4.2 เออนทิตี้ users มีความสัมพันธ์กับ เออนทิตี้ personality\_traits แบบ 1:M (One to Many) คือ 1 user สามารถมีลักษณะนิสัยได้หลายอย่าง

3.3.4.3 เออนทิตี้ users มีความสัมพันธ์กับ เออนทิตี้ likes แบบ 1:M (One to Many) คือ 1 user สามารถกดถูกใจผู้อื่นได้หลายครั้ง

3.3.4.4 เออนทิตี้ users มีความสัมพันธ์กับ เออนทิตี้ matches แบบ 1:M (One to Many) คือ 1 user สามารถจับคู่กันได้เมื่อกดถูกใจซึ่งกันและกัน

3.3.4.5 เออนทิตี้ users มีความสัมพันธ์กับ เออนทิตี้ reports แบบ 1:M (One to Many) คือ 1 user สามารถรายงานผู้ใช้คนอื่นหรือแจ้งปัญหาเกี่ยวกับระบบได้หลายครั้ง

3.3.4.6 เออนทิตี้ users มีความสัมพันธ์กับ เออนทิตี้ suggestions แบบ 1:M (One to Many) คือ 1 user สามารถให้ข้อเสนอแนะหรือแจ้งปัญหาได้หลายครั้ง

3.3.4.7 เออนทิตี้ users มีความสัมพันธ์กับ เออนทิตี้ app\_reviews แบบ 1:M (One to Many) คือ 1 user สามารถให้ข้อเสนอแนะหรือแจ้งปัญหาได้หลายครั้ง

### 3.3.5 Data Dictionary (พจนานุกรมข้อมูล)

ชื่อตาราง	users
คำอธิบาย	ข้อมูลรหัสผู้ใช้
Foreign key (FK)	-

ตารางที่ 3.3 ตัวอย่างข้อมูลตารางของ users

Attribute	Date Type	Size (Byte)	Example	Description
<u>id</u>	Number	4	11	รหัสผู้ใช้
username	Text	50	“Toon”	ชื่อยูเซอร์ผู้ใช้
email	Text	100	“toon@gmail.com”	อีเมล
password	Text	60	“toon123456”	รหัสผ่าน
role	Text	5	“user”	บทบาท
profile_picture	Text	100	“/cloudinary/dhn6ltrqh.jpg”	รูปโปรไฟล์
is_suspended	Number	1	0	สถานะการระงับบัญชี
suspension_reason	Text	255	“บัญชีปลอม”	เหตุผลการระงับบัญชี
email_verified	Number	1	1	ยืนยันอีเมล

ชื่อตาราง	personality_information	
คำอธิบาย	ข้อมูลส่วนตัวผู้ใช้	
Foreign key (FK)	user_id	อ้างอิงถึงตาราง users(id)

ตารางที่ 3.4 ตัวอย่างข้อมูลตารางของ personality\_information

Attribute	Data Type	Size (Byte)	Example	Description
<u><b>id</b></u>	Number	4	8	ลำดับข้อมูล
<u><b>user_id</b></u>	Number	4	11	รหัสผู้ใช้
firstname	Text	100	“Jittapong”	ชื่อ
lastname	Text	100	“Jongjai”	นามสกุล
nickname	Text	100	“Toon”	ชื่อเล่น
age	Number	1	23	อายุ
maritalstatus	Text	20	“single”	สถานภาพ
gender	Text	10	“male”	เพศ
lgbt	Number	1	0	ความหลากหลายทางเพศ
province	Text	50	“เชียงใหม่”	จังหวัด
university	Text	255	“มหาวิทยาลัยเชียงใหม่”	สถานศึกษา
facebook	Text	50	“Jittapong”	ชื่อผู้ใช้ Facebook
instagram	Text	50	“Thiztoon”	ชื่อผู้ใช้ Instagram
line_id	Text	50	“szrybb”	ไอดี Line
phone	Text	15	“0617956385”	หมายเลขโทรศัพท์
dorm_name	Text	50	“Powadol”	ชื่อหอพัก
vehicle	Text	10	“motorbike”	ยานพาหนะ
self_introduction	Text	255	“เป็นคนชาญ”	แนะนำตัวเอง
monthly_dorm_fee	Number	4	4700.00	ค่าหอพักต่อเดือน

ชื่อตาราง	personality_traits	
คำอธิบาย	ข้อมูลลักษณะนิสัยผู้ใช้	
Foreign key (FK)	user_id	อ้างอิงถึงตาราง users(id)

ตารางที่ 3.5 ตัวอย่างข้อมูลตารางของ personality\_traits

Attribute	Data Type	Size (Byte)	Example	Description
<u><b>id</b></u>	Number	4	8	ลำดับข้อมูล
<u><b>user_id</b></u>	Number	4	11	รหัสผู้ใช้
type	Text	25	“type_introvert”	บุคลิกภาพ
sleep	Text	25	“sleep_after_midnight”	ช่วงเวลาอน
wake	Text	25	“wake_noon”	ช่วงเวลาตื่น
clean	Text	25	“clean_every_day”	การทำความสะอาด
air_conditioner	Text	25	“ac_only_hot”	การใช้งานเครื่องปรับอากาศ
drink	Text	25	“drink_never”	การดื่มแอลกอฮอล์
smoke	Text	25	“smoke_never”	การสูบบุหรี่
money	Text	25	“money_on_time”	การจ่ายค่าหอพักตรงเวลา
expense	Text	25	“money_ratio”	การแบ่งค่าใช้จ่ายหอพัก
pet	Text	25	“pet_dont_have”	การเลี้ยงสัตว์
cook	Text	25	“cook_no”	การทำอาหาร
loud	Text	25	“loud_medium”	การใช้เสียง
friend	Text	25	“friend_tell_first”	การพาเพื่อนมาห้อง
religion	Text	25	“religion_no_affect”	เรื่องเกี่ยวกับศาสนา
period	Text	25	“period_long”	ระยะเวลาการเช่า

ชื่อตาราง	likes	
คำอธิบาย	ข้อมูลการกดถูกใจ	
Foreign key (FK)	user_id	อ้างอิงถึงตาราง users(id)
	liked_user_id	อ้างอิงถึงตาราง users(id)

ตารางที่ 3.6 ตัวอย่างข้อมูลตารางของ likes

Attribute	Data Type	Size (Byte)	Example	Description
<b>id</b>	Number	4	8	ลำดับข้อมูล
<b>user_id</b>	Number	4	11	รหัสผู้ใช้
<b>liked_user_id</b>	Number	4	45	รหัสผู้ใช้ที่กดถูกใจ
<b>created_at</b>	Date/Time	8	‘2025-01-07 18:38:41’	เวลาที่กดถูกใจ

ชื่อตาราง	matches	
คำอธิบาย	ข้อมูลการจับคู่	
Foreign key (FK)	user1_id	อ้างอิงถึงตาราง users(id)
	user2_id	อ้างอิงถึงตาราง users(id)

ตารางที่ 3.7 ตัวอย่างข้อมูลตารางของ matches

Attribute	Data Type	Size (Byte)	Example	Description
<b>id</b>	Number	4	8	ลำดับข้อมูล
<b>user1_id</b>	Number	4	11	รหัสผู้ใช้ที่จับคู่คนที่ 1
<b>user2_id</b>	Number	4	45	รหัสผู้ใช้ที่จับคู่คนที่ 2
<b>created_at</b>	Date/Time	8	“2025-01-07 18:38:41”	เวลาที่จับคู่สำเร็จ

ชื่อตาราง	reports	
คำอธิบาย	ข้อมูลการรายงาน	
Foreign key (FK)	reporter_id reported_user_id	อ้างอิงถึงตาราง users(id) อ้างอิงถึงตาราง users(id)

ตารางที่ 3.8 ตัวอย่างข้อมูลตารางของ reports

Attribute	Data Type	Size (Byte)	Example	Description
<b>id</b>	Number	4	8	ลำดับข้อมูล
<b>reporter_id</b>	Number	4	12	รหัสผู้ใช้ที่รายงาน
<b>reported_user_id</b>	Number	4	33	รหัสผู้ใช้ที่ถูกรายงาน
type	Text	25	“harassment”	หัวข้อการรายงาน
description	Text	255	“รูปภาพ อนาจาร”	คำอธิบาย
image	Text	100	“/cloudinary/ dhn6ltrqh.jpg”	รูปภาพประกอบ
report_type	Text	10	“user”	ประเภทการรายงาน
status	Text	10	“pending”	สถานะของรายงาน
created_at	Date/Time	8	‘2025-01-07 18:38:41’	เวลาที่รายงาน

ชื่อตาราง	suggestions	
คำอธิบาย	ข้อมูลการแนะนำ	
Foreign key (FK)	user_id	อ้างอิงถึงตาราง users(id)

ตารางที่ 3.9 ตัวอย่างข้อมูลตารางของ suggestions

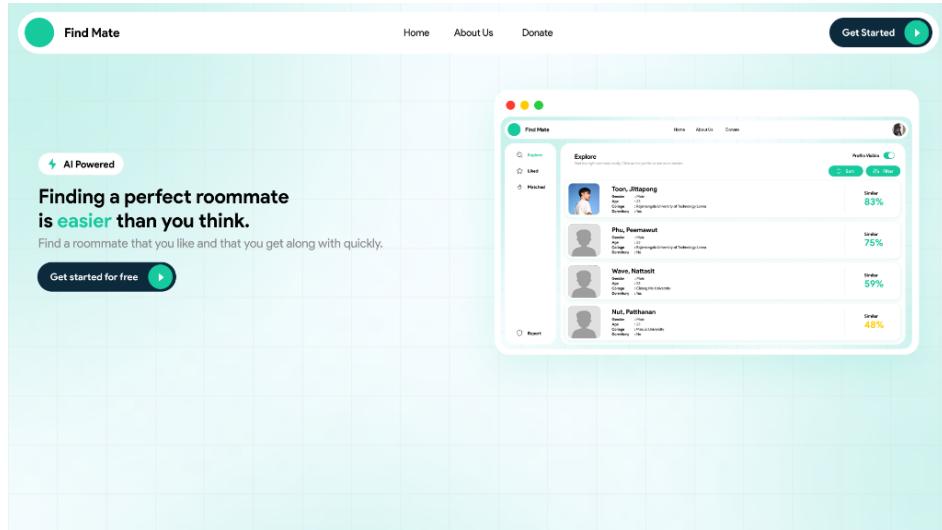
Attribute	Data Type	Size (byte)	Example	Description
<u>id</u>	Number	4	8	ลำดับข้อมูล
<u>user_id</u>	Number	4	11	รหัสผู้ใช้
content	Text	255	“เพิ่มหัวข้ออื่น ๆ”	คำแนะนำ
status	Text	10	“pending”	สถานะการจัดการ
created_at	Date/Time	8	‘2025-01-07 18:38:41’	เวลาที่กดถูกใจ

ชื่อตาราง	app_reviews	
คำอธิบาย	ข้อมูลการรีวิว	
Foreign key (FK)	user_id	อ้างอิงถึงตาราง users(id)

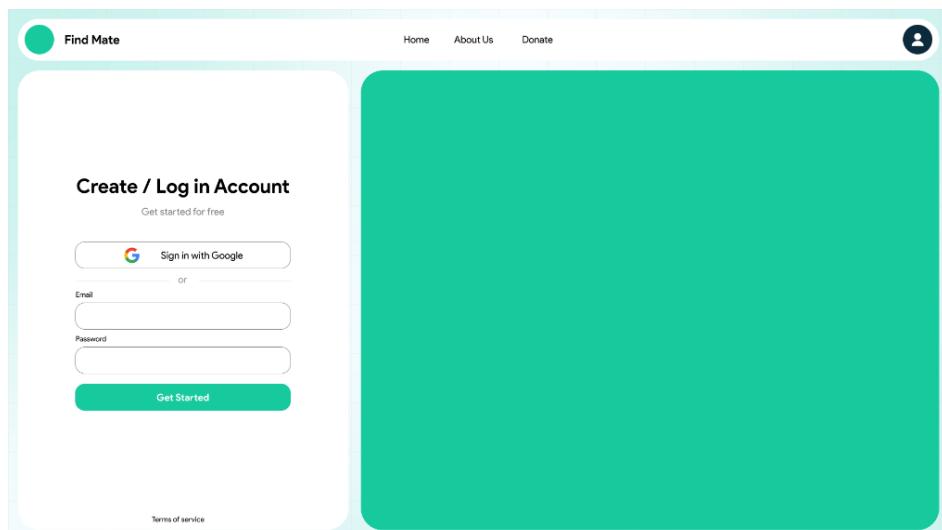
ตารางที่ 3.10 ตัวอย่างข้อมูลตารางของ app\_reviews

Attribute	Data Type	Size (byte)	Example	Description
<u>id</u>	Number	4	8	ลำดับข้อมูล
<u>user_id</u>	Number	4	11	รหัสผู้ใช้
rating	Number	1	5	คะแนนการรีวิว
feedback	Text	255	“อยากรู้更多 มากขึ้น”	ข้อเสนอแนะ
email_sent	Number	1	1	ส่งการรีวิวผ่านอีเมล สำเร็จ
created_at	Date/Time	8	‘2025-03-02 12:01:25’	วันและเวลาที่ทำรีวิว

### 3.4 ขั้นตอนการออกแบบโครงสร้างเว็บไซต์

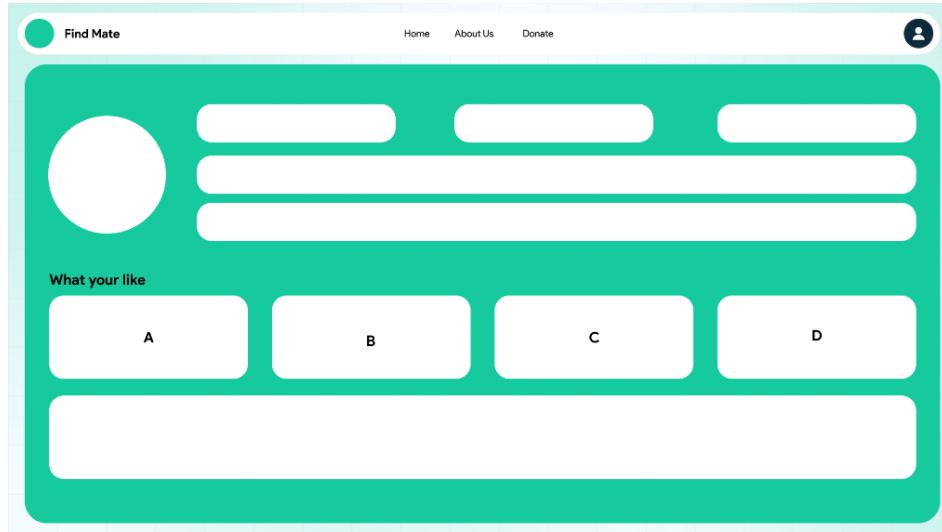


รูปที่ 3.21 การออกแบบ Prototype หน้าแรก



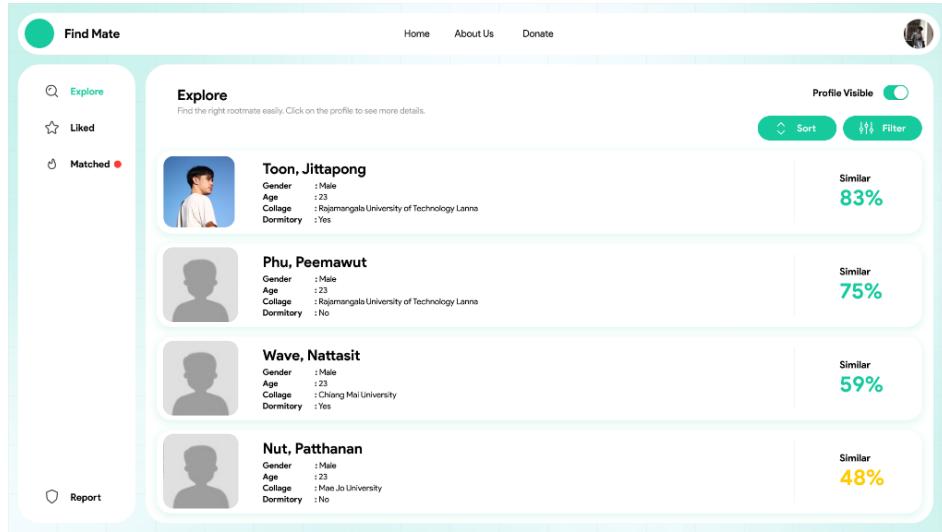
รูปที่ 3.22 การออกแบบ Prototype หน้าเข้าสู่ระบบ

จากรูปที่ 3.21 หน้าแรก จะเป็นหน้าเริ่มต้นของเว็บไซต์ ที่ให้ผู้ใช้ทำการกดปุ่มเข้าสู่ระบบ ตามรูปที่ 3.22 หน้าเข้าสู่ระบบ ที่จะประกอบไปด้วย ช่องกรอกอีเมล, ช่องกรอกรหัสผ่าน และปุ่มเข้าสู่ระบบ ผู้ใช้กรอกข้อมูลเข้าสู่ระบบตามที่ผู้ใช้ได้ลงทะเบียนไว้ แต่หากผู้ใช้ยังไม่ได้ลงทะเบียน สามารถไปยังหน้าสมัครสมาชิกได้



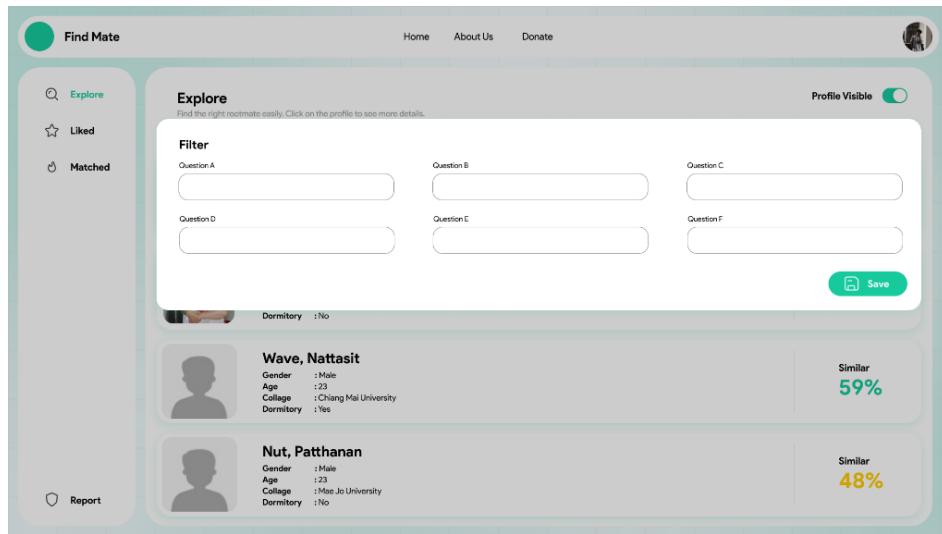
รูปที่ 3.23 การออกแบบ Prototype หน้าสมัครสมาชิก

จากรูปที่ 3.23 หน้าสมัครสมาชิก เป็นหน้าจอสำหรับผู้ใช้ที่ยังไม่เคยลงทะเบียนสมัครใช้งาน จำเป็นที่จะต้องสมัครใช้งานก่อนที่จะเข้าสู่ระบบ โดยจะประกอบไปด้วย ช่องกรอกชื่อเล่น ช่องกรอก อีเมล ช่องกรอกรหัสผ่าน ช่องกรอกยืนยันรหัสผ่าน บุ้มสมัครใช้งาน จากนั้นให้ผู้ใช้กรอกข้อมูลส่วนตัว ซึ่งจะให้ผู้ใช้กรอกข้อมูลประวัติส่วนตัว โดยประกอบไปด้วย ช่องกรอกชื่อจริง ช่องกรอกนามสกุล ช่องกรอกอายุ ช่องเลือกสถานะ ช่องเลือกเพศ บุ้มเลือกมีความหลากหลายทางเพศ และช่องเลือก จังหวัด จากนั้นเลือกลักษณะนิสัย จะเป็นแบบสอบถามที่จะมีคำถามเกี่ยวกับลักษณะนิสัยในการใช้ ชีวิตประจำวันของผู้ใช้ โดยจะให้ผู้ใช้ตอบเป็นตัวเลือกที่กำหนด ข้อมูลเหล่านี้จะนำไปเปรียบเทียบกับ ผู้ใช้ที่อยู่ในระบบ



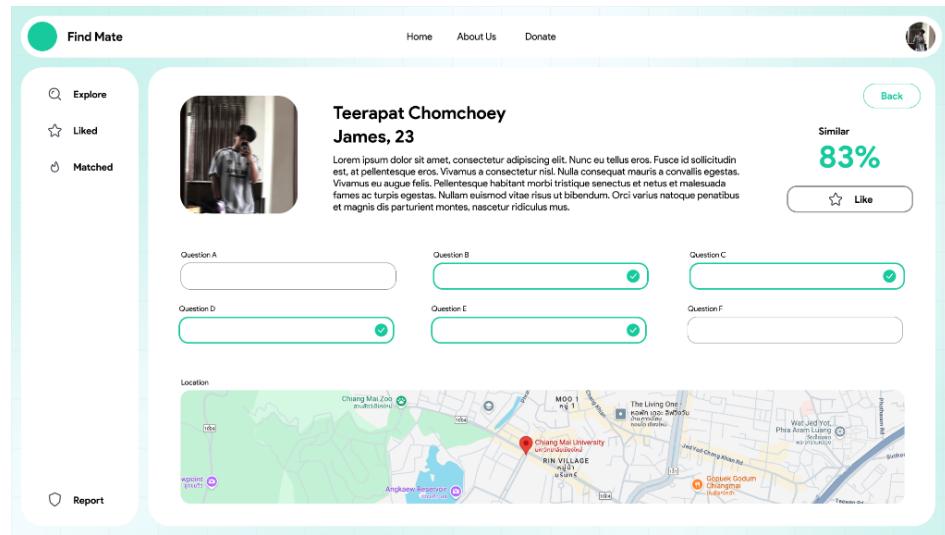
รูปที่ 3.24 การออกแบบ Prototype หน้าค้นหารูปเมท

จากรูปที่ 3.24 หน้าค้นหารูปเมท ซึ่งจะเป็นหน้าหลักของเว็บไซต์ ที่จะแสดงprofileรูปเมทเป็นลำดับตามความคล้ายคลึงกันของข้อมูลที่ระบบประเมินออกมาจาก ลักษณะนิสัยที่ผู้ใช้ได้เลือกไว้ โดยจะเรียงลำดับความคล้ายคลึงจากมากไปน้อย เพื่อให้ผู้ใช้ได้ติดสินใจเลือกรูปเมทได้ง่ายขึ้น จากความใกล้เคียงกันของลักษณะนิสัย และมีปุ่มตัวกรอง ที่ผู้ใช้สามารถเลือกรองการแสดงผล รูปเมทตามข้อมูลที่ผู้ใช้สนใจได้



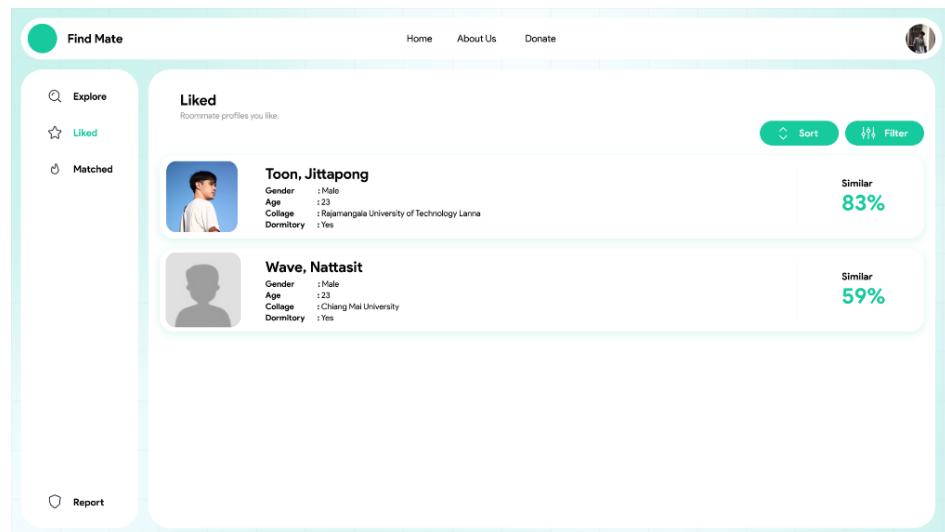
รูปที่ 3.25 การออกแบบ Prototype หน้าต่างตัวกรอง

จากรูปที่ 3.25 หน้าต่างตัวกรอง โดยจะมีปุ่ม ตัวกรอง ที่ผู้ใช้สามารถเลือกรองการแสดงผลของ รูปเมทตามข้อมูลที่ผู้ใช้สนใจได้ เช่น กรองตามจังหวัด มหาวิทยาลัย เพศ และลักษณะนิสัยต่าง ๆ



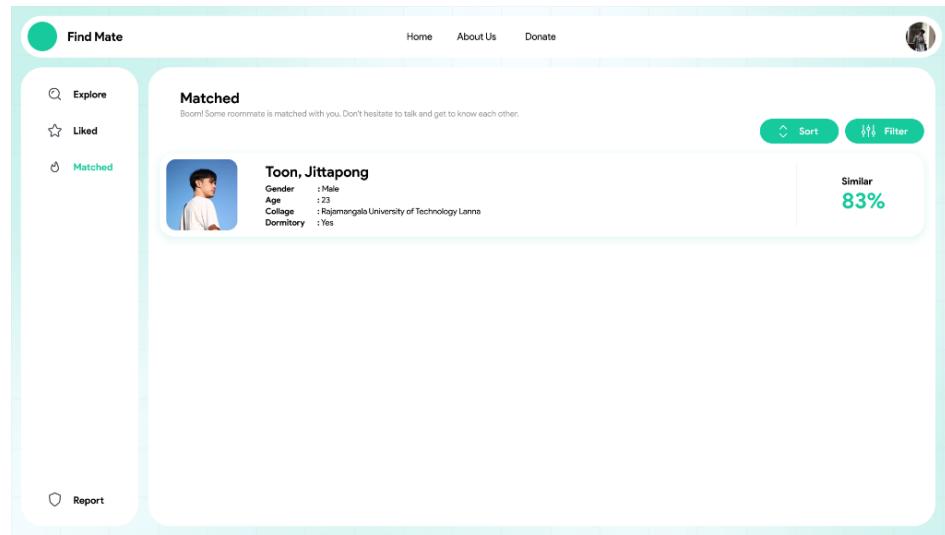
รูปที่ 3.26 การออกแบบ Prototype หน้าต่างໂປຣໄຟລ໌ຮູມເມທ

จากรูปที่ 3.26 หน้าต่างໂປຣໄຟລ໌ຮູມເມທ เมื่อผู้ใช้คลิกไปที่ໂປຣໄຟລ໌ของຮູມເມທທີ່ສັນໃຈ ระบบจะแสดงเป็นหน้าต่างທີ່ຈະแสดงຂໍ້ມູນສ່ວນຕົວ ຂໍ້ມູນການຕິດຕ່ອ ແລະ ຂໍ້ມູນລັກຂະນະນິສັຍ ຂອງຜູ້ໃຊ້ທີ່ໄດ້ກອກໄວ້ຕອນສົມຄຽນສາມາລືກ ໃນສ່ວນຂໍ້ມູນລັກຂະນະນິສັຍ ຈະມີການເນັ້ນໃນສ່ວນທີ່ຂໍ້ມູນຂອງຜູ້ໃຊ້ທີ່ 2 ຝ່າຍຕຽບກັນອຍ່າງໜັດເຈນ ພາກສນໄຈຮູມເມທຄົນນັ້ນກໍສາມາຄັດຄຸກໄຈໄດ້ ປຸ່ມຄຸກໃຈ ອີ່ວາກພບເຈົ້າຜູ້ໃຊ້ ທີ່ມີຮູບໂປຣໄຟລ໌ຫຼືອ໌ຂໍ້ມູນເສນ ໂປຣໄຟລ໌ປລອມ ສາມາຄັດປຸ່ມຮາຍຈາຍ ເພື່ອຮາຍຈານໄປຢັ້ງຜູ້ດູແລະບບ



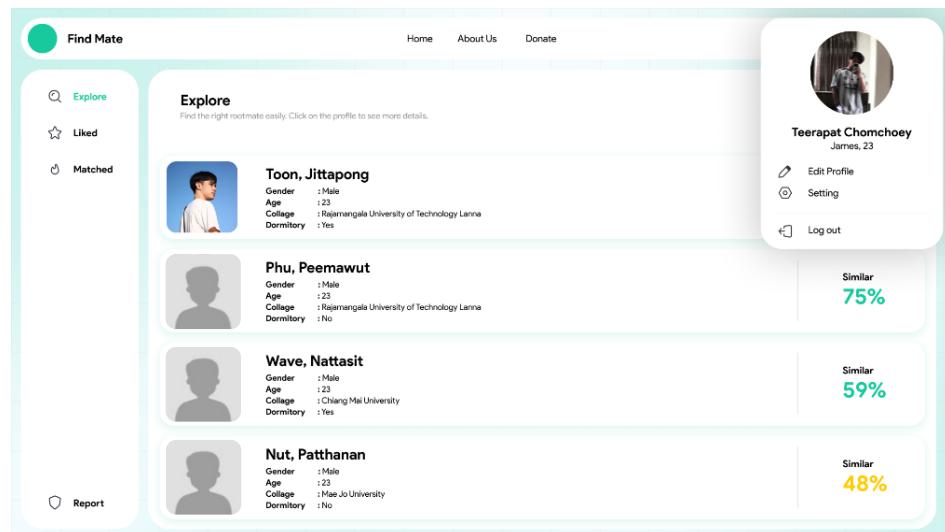
รูปที่ 3.27 การออกแบบ Prototype หน้าຄຸກໃຈ

จากรูปที่ 3.27 หน้า Liked Page ເປັນหน້າຈົດທີ່ຈະແສດງໂປຣໄຟລ໌ຮູມເມທທີ່ເຮົາໄດ້ເຄີຍຄັດຄຸກໃຈໄວ້ ມີໄວ້ເພື່ອໃຫ້ຜູ້ໃຊ້ດູຫຼືອ່ວຍເຫັນໂປຣໄຟລ໌ຮູມເມທທີ່ເຄີຍຄັດຄຸກໃຈໄວ້ໂດຍເພັະ



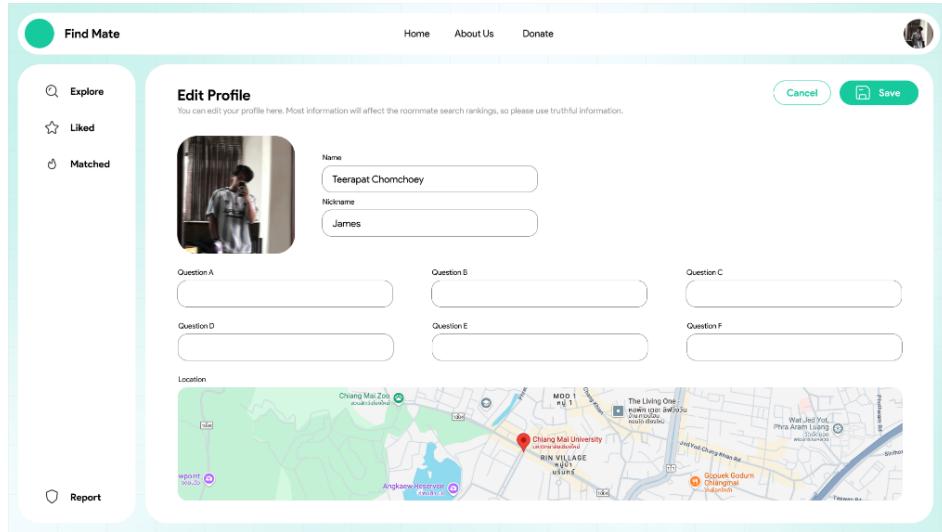
รูปที่ 3.28 การออกแบบ Prototype หน้า Matched Page

จากรูปที่ 3.28 หน้า Matched Page เป็นหน้าจอที่จะแสดงผู้ใช้ที่ได้ทำการกดถูกใจคุณกลับหรือก็คือผู้ใช้ทั้งสองฝ่ายกดถูกใจซึ่งกันและกัน เมื่อจับคู่สำเร็จโปรแกรมผู้ใช้ของคนที่จับคู่สำเร็จจะแสดงเบอร์โทรศัพท์มือถือ เพื่อที่จะสามารถติดต่อกันได้สะดวกยิ่งขึ้น



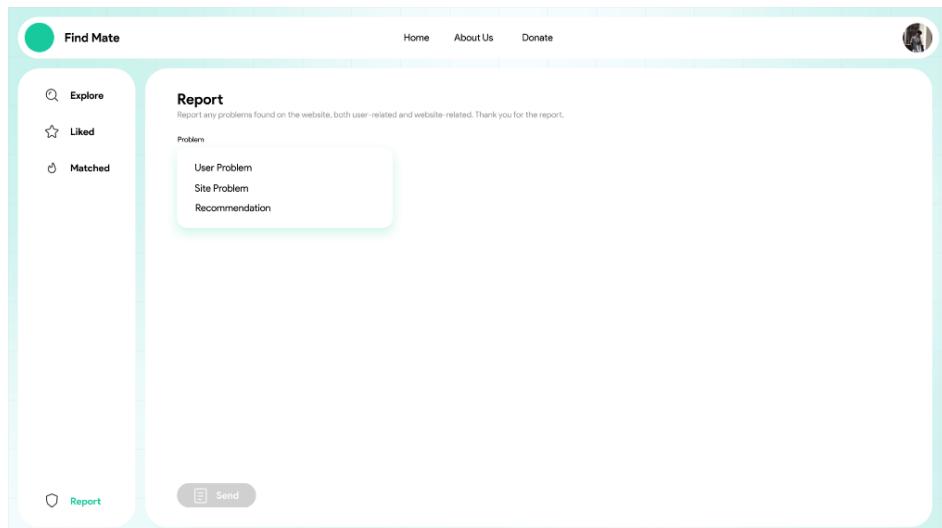
รูปที่ 3.29 การออกแบบ Prototype หน้าต่างเมนู

จากรูปที่ 3.29 หน้าต่างໂປຣໄຟລ് เมื่อผู้ใช้กดที่รูปໂປຣໄຟລ്ของตัวเอง จะแสดงปุ่มที่ประกอบไปด้วย ปุ่มแกໍາເຂົ້າໂປຣໄຟລ് ที่สามารถเข้าไปແກໍາເຂົ້າຂອ່ມລໂປຣໄຟລ໌ສ່ວນຕ້ວໄດ້ และปุ่ມອອກຈາກຮບບ ເນື້ອກດແລ້ວຜູ້ໃຊ້ຈະອອກຈາກຮບບທັນທີ



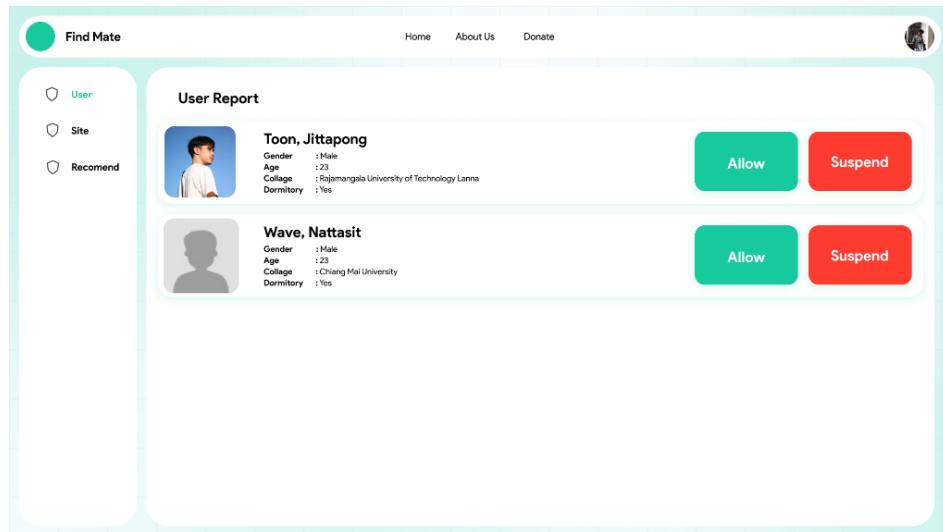
รูปที่ 3.30 การออกแบบ Prototype หน้าแก้ไขโปรไฟล์

จากรูปที่ 3.30 หน้าแก้ไขโปรไฟล์ โดยจะแสดงข้อมูลปัจจุบันของผู้ใช้ และให้ผู้ใช้เลือกแก้ไขข้อมูลที่ต้องการ โดยการแก้ไขจะแบ่งออกเป็น 2 ส่วน คือ ส่วนแรก แก้ไขประวัติส่วนตัว เช่น รูปโปรไฟล์ ชื่อจริง นามสกุล ชื่อเล่น อายุ และข้อมูลการติดต่อ เป็นต้น ส่วนที่สอง แก้ไขลักษณะนิสัย ผู้ใช้สามารถเลือกคำตอบใหม่ได้ทันที โดยเมื่อแก้ไขเรียบร้อยและกดปุ่มบันทึก ระบบจะอัพเดทโปรไฟล์ผู้ใช้ทันที



รูปที่ 3.31 การออกแบบ Prototype หน้าแจ้งปัญหาการใช้งาน

จากรูปที่ 3.31 หน้าแจ้งปัญหาการใช้งาน โดยจะแบ่งออกเป็น 3 ประเภทได้แก่ 1. แจ้งปัญหาเกี่ยวกับผู้ใช้ 2. แจ้งปัญหาเกี่ยวกับระบบ และ 3. แนะนำเกี่ยวกับระบบ หากผู้ใช้ต้องการแจ้งปัญหาเกี่ยวกับการใช้งานเว็บไซต์ สามารถทำได้โดยการกดปุ่มแจ้งปัญหาการใช้งาน ก็จะแสดงหน้าต่างที่จะประกอบไปด้วย การเลือกประเภทปัญหา พร้อมใส่คำอธิบายปัญหานั้น เพื่อแจ้งให้กับผู้ดูแลระบบ



รูปที่ 3.32 การออกแบบ Prototype หน้าแดชบอร์ดผู้ดูแลระบบ

จากรูปที่ 3.32 หน้าแดชบอร์ดผู้ดูแลระบบ จะประกอบไปด้วย จะแสดงการรายงานผู้ใช้จากผู้ใช้ทั้งหมดโดย โดยจะแสดง ชื่อผู้รายงาน ชื่อผู้ถูกรายงาน ประเภทการรายงาน สถานะของรายงาน วันที่ที่รายงาน และปุ่มการจัดการ ที่จะมีปุ่มดูรายละเอียดการรายงาน ปุ่มระงับการใช้งานของผู้ใช้ และปุ่มลบรายงาน

### 3.5 ขั้นตอนการออกแบบอัลกอริทึม K-NN

3.5.1 กำหนดลักษณะนิสัยที่ใช้ในการวิเคราะห์ 15 ข้อ (Feature ของ K-NN จำนวนทั้งหมด 15 Feature)

#### 3.5.1.1 บุคลิกภาพ

- 1) Introvert ชอบสังเกตและฟัง
- 2) Ambivert สมดุลระหว่างพูดและฟัง
- 3) Extrovert ชอบพูดและเข้าสังคม

#### 3.5.1.2 เวลาเข้าอน

- 1) ก่อนเที่ยงคืน
- 2) หลังเที่ยงคืน

#### 3.5.1.3 เวลาตื่น

- 1) ตื่นตอนเช้า
- 2) ตื่นตอนบ่าย
- 3) ตื่นตอนเย็น

#### 3.5.1.4 ความสะอาด

- 1) ทำความสะอาดทุกวัน
- 2) ทำความสะอาดวันเว้นวัน
- 3) ทำความสะอาดสัปดาห์ละครึ่ง
- 4) ไม่ค่อยชอบทำความสะอาด

#### 3.5.1.5 การใช้เครื่องปรับอากาศ

- 1) ไม่เปิดเลย
- 2) เปิดเฉพาะเวลากลางคืน
- 3) เปิดเฉพาะช่วงอากาศร้อน
- 4) เปิดทั้งวัน

#### 3.5.1.6 การดื่มแอลกอฮอล์

- 1) ไม่ดื่ม
- 2) ดื่มเฉพาะโอกาสพิเศษ
- 3) ดื่มช่วงสุดสัปดาห์
- 4) ดื่มเป็นประจำ

3.5.1.7 การสูบบุหรี่

- 1) ไม่สูบ
- 2) สูบเฉพาะเวลาสังสรรค์
- 3) สูบเป็นประจำ

3.5.1.8 การจ่ายเงินตรงเวลา

- 1) ตรงเวลาเสมอ
- 2) อาจคลาดเคลื่อนเล็กน้อย

3.5.1.9 การแบ่งค่าใช้จ่ายหอพัก

- 1) แบ่งเท่ากัน (ครึ่งต่อครึ่ง)
- 2) ตามสัดส่วนการใช้งาน (ใช้มากจ่ายมากกว่า)

3.5.1.10 การเลี้ยงสัตว์

- 1) ไม่เลี้ยง
- 2) เลี้ยง

3.5.1.11 การทำอาหารกลิ่นแรงในห้อง

- 1) ยอมรับได้
- 2) ได้ถ้าบอกล่วงหน้า
- 3) ไม่ยอมรับ

3.5.1.12 การใช้เสียง

- 1) ไม่ดัง
- 2) ปานกลาง
- 3) ดังมาก

3.5.1.13 การพาเพื่อนมาห้อง

- 1) โวเช
- 2) ได้แต่ต้องบอกล่วงหน้า
- 3) ไม่โวเช

3.5.1.14 ความเชื่อทางศาสนาที่แตกต่างกัน

- 1) สะเด็ก
- 2) ได้ถ้าไม่กระทบกัน
- 3) ไม่สะเด็ก

### 3.5.1.15 ความต้องการรูมเมทในระยะยาว

- 1) ต้องการ
- 2) ขึ้นอยู่กับสถานการณ์
- 3) ไม่จำเป็น

### 3.5.2 การกำหนดค่าถ่วงน้ำหนักของคุณลักษณะแต่ละด้าน

การกำหนดค่าถ่วงน้ำหนักของคุณลักษณะแต่ละด้านมีความจำเป็น เนื่องจากแต่ละคุณลักษณะมีระดับผลกระทบต่อความสัมพันธ์ในการอยู่ร่วมกันที่แตกต่างกัน จึงได้มีการวิเคราะห์และกำหนดอัตราคุณตามระดับความสำคัญดังต่อไปนี้

#### 3.5.2.1 บุคลิกภาพ

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1 เนื่องจากบุคลิกภาพเป็นคุณลักษณะพื้นฐานที่ส่งผลต่อการปรับตัวในการอยู่ร่วมกัน แต่สามารถปรับเปลี่ยนได้ตามสถานการณ์และมีผลกระทบในระดับปานกลางต่อความสัมพันธ์โดยรวม

#### 3.5.2.2 เวลาเข้าอน

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1 เนื่องจากอาจส่งผลกระทบต่อคุณภาพการพักผ่อนของอีกฝ่าย อย่างไรก็ตาม หากมีการควบคุมการใช้เสียงและแสงไฟอย่างเหมาะสม ผลกระทบดังกล่าวอาจลดน้อยลงหรือไม่ส่งผลกระทบต่อการอยู่ร่วมกัน

#### 3.5.2.3 เวลาตื่น

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1 เนื่องจากรูปแบบการตื่นนอนที่แตกต่างกันอาจส่งผลกระทบต่อกิจวัตรประจำวันของผู้อยู่ร่วม โดยเฉพาะในกรณีที่มีการใช้พื้นที่ร่วมกันในช่วงเช้าอย่างไรก็ตาม หากมีการสื่อสารและจัดระเบียบการใช้พื้นที่อย่างเหมาะสม ผลกระทบนี้สามารถบรรเทาได้

#### 3.5.2.4 ความสะอาด

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 2 เนื่องจากเป็นคุณลักษณะสำคัญในการอยู่ร่วมกับผู้อื่น การรักษาสุขอนามัยส่วนบุคคลและพื้นที่ส่วนรวมเป็นสิ่งจำเป็นอย่างยิ่ง เพื่อป้องกันความขัดแย้งหรือความรู้สึกไม่พึงพอใจระหว่างผู้อยู่ร่วม รวมถึงเป็นการส่งเสริมสภาพแวดล้อมที่ถูกสุขลักษณะสำหรับทุกฝ่าย

#### 3.5.2.5 การใช้เครื่องปรับอากาศ

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1.5 เนื่องจากการใช้เครื่องปรับอากาศมีผลโดยตรงต่อค่าสาธารณูปโภค ซึ่งการใช้งานที่ไม่เหมาะสมอาจส่งผลให้ค่าใช้จ่ายเพิ่มสูงขึ้นเกินความจำเป็น และอาจนำไปสู่ความไม่พึงพอใจระหว่างผู้อยู่ร่วม รวมถึงมีผลต่อความสบายในการพักอาศัยของทั้งสองฝ่าย

### 3.5.2.6 การดีมแอลกอฮอล์

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1.8 เนื่องจากการดีมเครื่องดีมแอลกอฮอล์อาจส่งผลต่อพฤติกรรมและการควบคุมตนเอง ซึ่งอาจก่อให้เกิดความไม่สงบหรือความไม่สบายใจแก่ผู้อยู่ร่วม นอกจากนี้ ยังอาจส่งผลต่อความปลอดภัยและบรรยายกาศโดยรวมของที่พักอาศัยโดยเฉพาะหากมีการดีมในปริมาณมากหรือเป็นประจำ

### 3.5.2.7 การสูบบุหรี่

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 2.5 เนื่องจากเป็นคุณลักษณะที่มีความสำคัญสูงสุด ด้วยเหตุที่กลืนของควันบุหรี่ส่งผลกระทบต่อทั้งสุขภาพและสภาพแวดล้อมการอยู่อาศัย กลืนดังกล่าวสามารถแพร่กระจายได้ในระยะไกลและมีแนวโน้มที่จะติดค้างบนเครื่องแต่งกาย เครื่องใช้ส่วนตัว และเครื่องนอน อันอาจนำไปสู่ความไม่พึงพอใจและความขัดแย้งระหว่างผู้อยู่ร่วมอย่างรุนแรง

### 3.5.2.8 การจ่ายเงินตรงเวลา

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 2 เนื่องจากความล่าช้าในการชำระเงินอาจส่งผลกระทบทางการเงินต่อผู้อยู่ร่วมอีกฝ่ายซึ่งอาจต้องรับภาระค่าใช้จ่ายล่วงหน้า สถานการณ์ดังกล่าวอาจนำไปสู่ความตึงเครียดในความสัมพันธ์ระยะยาว นอกจากนี้ การชำระเงินล่าช้ายังอาจก่อให้เกิดปัญหาแก่ผู้ให้บริการที่พักอาศัย ทั้งในด้านค่าปรับและความไม่เชื่อถือทางการเงิน

### 3.5.2.9 การแบ่งค่าใช้จ่ายหอพัก

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1.5 เนื่องจากการจัดสรรค่าใช้จ่ายที่ไม่เป็นธรรมหรือไม่มีความชัดเจนอาจนำไปสู่ความขัดแย้งทางการเงิน การที่ห้องสองฝ่ายมีความเข้าใจแตกต่างกันเกี่ยวกับวิธีการแบ่งค่าใช้จ่ายตั้งแต่เริ่มต้นจะช่วยลดความเสี่ยงในการเกิดข้อพิพาทและส่งเสริมบรรยายกาศการอยู่ร่วมกันอย่างราบรื่น

### 3.5.2.10 การเลี้ยงสัตว์

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1.3 เนื่องจากการมีสัตว์เลี้ยงในที่พักอาศัยร่วมกันอาจส่งผลกระทบในหลายด้าน ทั้งในเรื่องความสะอาด เสียงรบกวน กลิ่น และการดูแลรับผิดชอบ นอกจากนี้ ยังต้องคำนึงถึงนโยบายของที่พักอาศัยและความชอบส่วนบุคคลของผู้อยู่ร่วมรวมถึงประเด็นด้านกฎหมายแพ้หรือความกลัวสัตว์ที่อาจมี

### 3.5.2.11 การทำอาหารกลิ่นแรงในห้อง

กำหนดค่าถ่วงน้ำหนักที่อัตราคุณ 1 เนื่องจากกลิ่นอาหารสามารถแพร่กระจายและติดค้างในพื้นที่และเครื่องใช้ส่วนตัวได้อย่างไรก็ตาม คุณลักษณะนี้สามารถบริหารจัดการได้ด้วยการระบายอากาศที่เหมาะสมและการทำความสะอาดอย่างสม่ำเสมอ รวมถึงการสื่อสารและตกลงร่วมกันเกี่ยวกับช่วงเวลาที่เหมาะสมในการประกอบอาหาร

### 3.5.2.12 การใช้เสียง

กำหนดค่าถ่วงน้ำหนักที่อัตราคูณ 1.8 เนื่องจากเสียงระบบมีผลกระทบโดยตรงต่อความสงบและคุณภาพชีวิตของผู้อยู่ร่วม โดยเฉพาะในช่วงเวลาพักผ่อนหรือการทำงาน การศึกษา การควบคุมระดับเสียงและการเดินทางช่วงเวลาที่ต้องการความเงียบเป็นคุณลักษณะสำคัญในการรักษาความสัมพันธ์ที่ดีและส่งเสริมสภาพแวดล้อมที่เอื้อต่อการอยู่ร่วมกัน

### 3.5.2.13 การพาเพื่อนมาห้อง

กำหนดค่าถ่วงน้ำหนักที่อัตราคูณ 1.5 เนื่องจากการนำบุคคลภายนอกเข้ามาในพื้นที่ส่วนตัวรวมอาจส่งผลต่อความเป็นส่วนตัว ความปลอดภัย และความสะดวกสบายของผู้อยู่ร่วม การกำหนดข้อตกลงร่วมกันเกี่ยวกับความถี่ ระยะเวลา และช่วงเวลาที่เหมาะสมในการมีผู้มาเยี่ยม เป็นสิ่งสำคัญในการรักษาสมดุลระหว่างอิสรภาพส่วนบุคคลและการเดินทางพื้นที่ร่วม

### 3.5.2.14 ความเชื่อทางศาสนาที่แตกต่างกัน

กำหนดค่าถ่วงน้ำหนักที่อัตราคูณ 1.2 เนื่องจากความแตกต่างในความเชื่อทางศาสนาอาจส่งผลต่อวิถีปฏิบัติและกิจกรรมประจำวัน ซึ่งอาจมีผลกระทบต่อการอยู่ร่วมกัน ทั้งในแง่ของการประกอบพิธีกรรม ข้อปฏิบัติทางศาสนา หรือข้อห้ามต่าง ๆ การเปิดใจยอมรับความแตกต่าง และการสื่อสารเพื่อหาข้อตกลงร่วมกันเป็นสิ่งสำคัญในการส่งเสริมการอยู่ร่วมกันอย่างเคร่งกัน และกัน

### 3.5.2.15 ความต้องการรูมเมทในระยะยาว

กำหนดค่าถ่วงน้ำหนักที่อัตราคูณ 1 เนื่องจากความคาดหวังที่แตกต่างกัน เกี่ยวกับระยะเวลาในการอยู่ร่วมกันอาจส่งผลต่อการวางแผนและการตัดสินใจในอนาคต อย่างไรก็ตาม คุณลักษณะนี้สามารถจัดการได้ด้วยการสื่อสารที่ชัดเจนและการตกลงร่วมกันล่วงหน้าเกี่ยวกับระยะเวลาและเงื่อนไขในการอยู่ร่วมกัน รวมถึงข้อกำหนดในการแจ้งล่วงหน้าหากมีการเปลี่ยนแปลง

### 3.5.3 การกำหนดค่าตัวแปรในการคำนวณ

แปลงข้อมูลคุณลักษณะแต่ละด้านเป็นค่าตัวเลข เพื่อสามารถนำไปคำนวณในสมการได้ การแปลงข้อมูลจะอิงตามจำนวนตัวเลือก โดยใช้การเข้ารหัสแบบลำดับ (Ordinal Encoding) ซึ่งจะแปลงค่าเป็นตัวเลขตามตำแหน่งในลำดับความเข้มข้นหรือความสัมพันธ์ต่อกัน เช่น introvert = 0, ambivert = 1, extrovert = 2 และนำไปคูณกับค่าน้ำหนักของแต่ละคุณลักษณะ

ตารางที่ 3.11 การแปลงข้อมูลโดยใช้การเข้ารหัสแบบ Ordinal Encoding

คุณลักษณะ	ข้อมูล	Ordinal
บุคลิกภาพ	Introvert ชอบสังเกตและฟัง	0
	Ambivert สมดุลระหว่างพูดและฟัง	1
	Extrovert ชอบพูดและเข้าสังคม	2
เวลาเข้านอน	ก่อนเที่ยงคืน	0
	หลังเที่ยงคืน	1
เวลาตื่น	ตื่นตอนเช้า	0
	ตื่นตอนบ่าย	1
	ตื่นตอนเย็น	2
ความสะอาด	ทำความสะอาดทุกวัน	0
	ทำความสะอาดวันเว้นวัน	1
	ทำความสะอาดสัปดาห์ละครึ่ง	2
	ไม่ค่อยชอบทำความสะอาด	3
การใช้เครื่องปรับอากาศ	ไม่เปิดเลย	0
	เปิดเฉพาะเวลากลางคืน	1
	เปิดเฉพาะช่วงอากาศร้อน	2
	เปิดตลอด	3
การดี๊ด๊า และออกอโอล์	ไม่ดี๊ด๊า	0
	ดี๊ด๊าเฉพาะโอกาสพิเศษ	1
	ดี๊ด๊าช่วงฤดูสัปดาห์	2
	ดี๊ด๊าเป็นประจำ	3

ตารางที่ 3.11 การแปลงข้อมูลโดยใช้การเข้ารหัสแบบ Ordinal Encoding (ต่อ)

คุณลักษณะ	ข้อมูล	Ordinal
การสูบบุหรี่	ไม่สูบ	0
	สูบเฉพาะเวลาสังสรรค์	1
	สูบเป็นประจำ	2
การจ่ายเงิน	ตรงเวลาเสมอ	0
	อาจคลาดเคลื่อนเล็กน้อย	1
การแบ่งค่าใช้จ่าย หอพัก	แบ่งเท่ากัน (ครึ่งต่อครึ่ง)	0
	ตามสัดส่วนการใช้งาน (ใช้มากจ่ายมากกว่า)	1
การเลี้ยงสัตร์	ไม่เลี้ยง	0
	เลี้ยง	1
การทำอาหาร กลิ่นแรงในห้อง	ยอมรับได้	0
	ได้ถ้าบอกล่วงหน้า	1
	ไม่ยอมรับ	2
การใช้เสียง	ไม่ดัง	0
	ปานกลาง	1
	ดังมาก	2
การพาเพื่อน มาห้อง	โวโคว	0
	ได้แต่ต้องบอกล่วงหน้า	1
	ไม่โวโคว	2
ความเชื่อทาง ศาสนาที่แตกต่าง	šeดวก	0
	ได้ถ้าไม่กระทบกัน	1
	ไม่šeดวก	2
ความต้องการรูม เมทในระยะยาว	ต้องการ	0
	ขึ้นอยู่กับสถานการณ์	1
	ไม่จำเป็น	2

### 3.5.4 การคำนวณระยะห่างโดยใช้ Euclidean Distance

$$d(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (3.1)$$

โดยที่	$d$	คือ ผลลัพธ์ระยะทางจากจุด $p$ ไปยังจุด $q$
	$p, q$	คือ จุดข้อมูล
	$n$	คือ จำนวนมิติของข้อมูล
	$\Sigma$	คือ ผลรวมระยะห่างกำลังสอง

### 3.5.5 การคำนวณร้อยละความเข้ากันได้จากระยะห่าง

$$\text{Similarity} = 100 \times (1 - \frac{\text{Distance}}{\text{Max Dixtance}}) \quad (3.2)$$

โดยที่	$Similarity$	คือ ความคล้ายระหว่างบุคคลผลลัพธ์เป็นร้อยละ
	$Distance$	คือ ระยะห่างระหว่างบุคคล
	$Max Dixtance$	คือ ระยะห่างของข้อมูลทุกมิติสูงสุดที่เป็นได้

## บทที่ 4

### ผลการทดลอง

#### 4.1 บทนำ

หลังจากที่ได้ดำเนินการในขั้นตอนการออกแบบ และทำขึ้นงานเสร็จสิ้น คณะกรรมการผู้จัดทำได้ นำขึ้นมาทดลอง เพื่อให้ทราบถึงผลลัพธ์ของโครงงานที่ได้จัดทำขึ้น ดังต่อไปนี้

##### 4.1.1 ระบบสมัครสมาชิกและลงชื่อใช้งาน

- 4.1.1.1 สามารถสมัครสมาชิกด้วย Email และรหัสผ่านได้
- 4.1.1.2 สามารถลงชื่อเข้าใช้งานระบบด้วย Email และรหัสผ่านได้
- 4.1.1.3 สามารถลบบัญชีได้
- 4.1.1.4 สามารถเปลี่ยนรหัสผ่านได้

##### 4.1.2 จัดเก็บข้อมูลผู้ใช้

- 4.1.2.1 สามารถจัดเก็บข้อมูลส่วนตัวของผู้ใช้ได้
  - 1) เก็บข้อมูลที่สามารถระบุตัวตน เช่น ชื่อ อายุ เพศ และรูปภาพ
  - 2) เก็บข้อมูลเกี่ยวกับ การศึกษา หอพักและยานพาหนะ
- 4.1.2.2 สามารถจัดเก็บข้อมูลลักษณะนิสัยผู้ใช้ได้
  - 1) ช่วงเวลาอน
  - 2) ความสะอาด
  - 3) การใช้เสียง
  - 4) การสูบบุหรี่
  - 5) การใช้งานเครื่องปรับอากาศ
  - 6) การเที่ยวกลางคืน
- 4.1.2.3 สามารถแก้ไขรายละเอียดข้อมูลในໂປຣໄຟລ์ของผู้ใช้ได้

##### 4.1.3 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้

- 4.1.3.1 สามารถนำข้อมูลผู้ใช้มาเปรียบเทียบได้
- 4.1.3.2 สามารถคำนวณรูปแบบร้อยละความใกล้เคียงของข้อมูลได้
- 4.1.3.3 สามารถแสดงผลผู้ใช้ที่เข้ากันได้ตามลำดับจากมากไปน้อยได้

##### 4.1.4 รายละเอียดໂປຣໄຟລ์ผู้ใช้

- 4.1.4.1 แสดงรูปໂປຣໄຟລ์ผู้ใช้ได้
- 4.1.4.2 แสดงรายละเอียดลักษณะนิสัยผู้ใช้ได้
- 4.1.4.3 สามารถเน้นข้อมูลในส่วนที่ตรงกันให้เด่นชัดขึ้นได้

- 4.1.5 ระบบกรองการค้นหาข้อมูล
  - 4.1.5.1 สามารถกรองการค้นหาด้วยตำแหน่งของผู้ใช้ได้
  - 4.1.5.2 สามารถกรองการค้นหาด้วยลักษณะนิสัยตามต้องการได้
- 4.1.6 ระบบการจับคู่
  - 4.1.6.1 สามารถกดถูกใจผู้ใช้รายอื่นได้
  - 4.1.6.2 แสดงผู้ใช้ที่ได้กดถูกใจไว้ได้
  - 4.1.6.3 สามารถถอนการกดถูกใจได้
  - 4.1.6.4 แสดงรูปเมทัจ็บคู่สำเร็จเมื่อทั้งสองฝ่ายกดถูกใจทั้งคู่
- 4.1.7 ระบบแจ้งเตือน
  - 4.1.7.1 สามารถแจ้งเตือนให้ผู้ใช้เมื่อจับคู่สำเร็จ
- 4.1.8 ระบบแจ้งปัญหา
  - 4.1.8.1 สามารถแจ้งปัญหาผู้ใช้ที่ไม่พึงประสงค์ได้
  - 4.1.8.2 สามารถแจ้งปัญหาเกี่ยวกับระบบการใช้งานได้
  - 4.1.8.3 สามารถแนะนำเกี่ยวกับระบบได้
- 4.1.9 ระบบให้คะแนนความพึงพอใจ
  - 4.1.9.1 สามารถให้คะแนนความพึงพอใจระบบการแนะนำรูปเมทในหน้าหลัก
  - 4.1.9.2 สามารถให้คะแนนความพึงพอใจต่อเว็บไซต์ได้
- 4.1.10 ระบบผู้ดูแลระบบ
  - 4.1.10.1 สามารถรับรายงานจากผู้ใช้ได้
  - 4.1.10.2 สามารถรับผู้ใช้ที่ไม่พึงประสงค์ได้
  - 4.1.10.3 สามารถปรับสถานะบัญชีผู้ใช้ได้ (เปิดหรือระงับบัญชี)
- 4.1.11 การประเมินผล
  - 4.1.11.1 เชิงปริมาณ
    - 1) คะแนนความพึงพอใจระบบการแนะนำรูปเมทเฉลี่ยมากกว่า 70%
    - 2) คะแนนความพึงพอใจการแสดงผลข้อมูลบุคคลเฉลี่ยมากกว่า 70%
    - 3) เว็บแอพพลิเคชันทำงานได้ตามขอบเขตมากกว่า 80%
  - 4.1.11.2 เชิงคุณภาพ
    - 1) ด้านความถูกต้องและเหมาะสมของเนื้อหาบนเว็บไซต์ โดยทดสอบใช้งานสร้างบัญชีตลอดจนถึงขั้นตอนการจับคู่อย่างน้อย 10 ครั้ง พบร่วมระบบทำงานถูกต้องไม่น้อยกว่า 80%

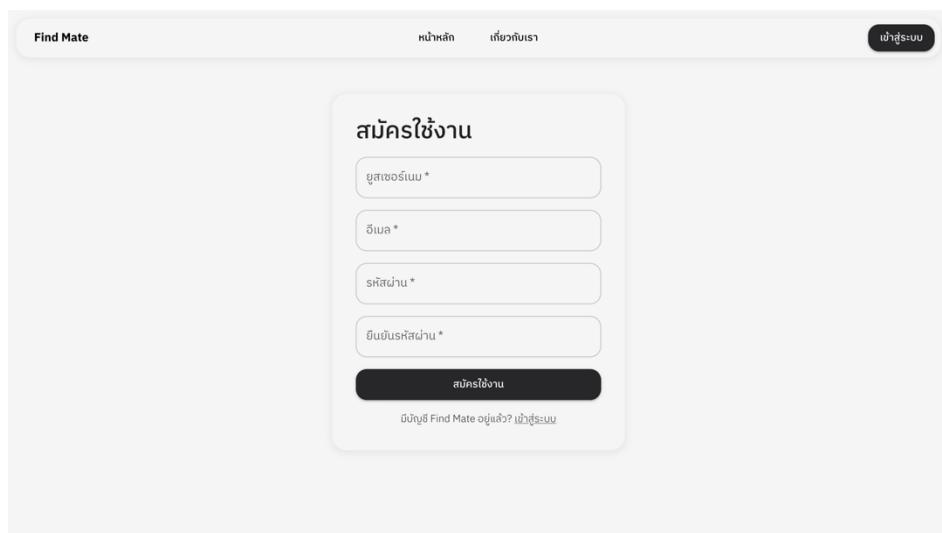
## 4.2 ผลการทำงานตามขอบเขต

### 4.2.1 หน้าแรกและระบบสมัครสมาชิกและลงชื่อเข้าใช้งาน



รูปที่ 4.1 หน้าแรก

จากรูปที่ 4.1 หน้าแรกจะเป็นหน้าเริ่มต้นของเว็บไซต์ โดยแนะนำเกี่ยวกับเว็บแอปพลิเคชันและแสดงผลสถิติการใช้งานและแนะนำพีเจอร์ของเว็บแอปพลิเคชัน



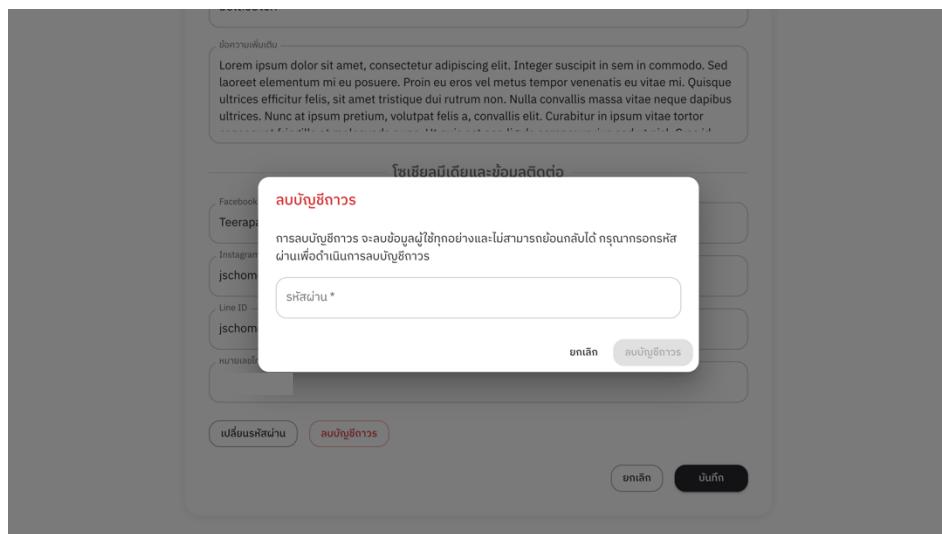
รูปที่ 4.2 หน้าสมัครใช้งาน

จากรูปที่ 4.2 เป็นหน้าสมัครใช้งาน ที่จะให้ผู้ใช้งานกรอกข้อมูล เมื่อกรอกข้อมูลครบตามที่กำหนด ให้กดปุ่มสมัครใช้งาน เพื่อเข้าสู่ระบบไปยังหน้าแรกของแอปพลิเคชัน



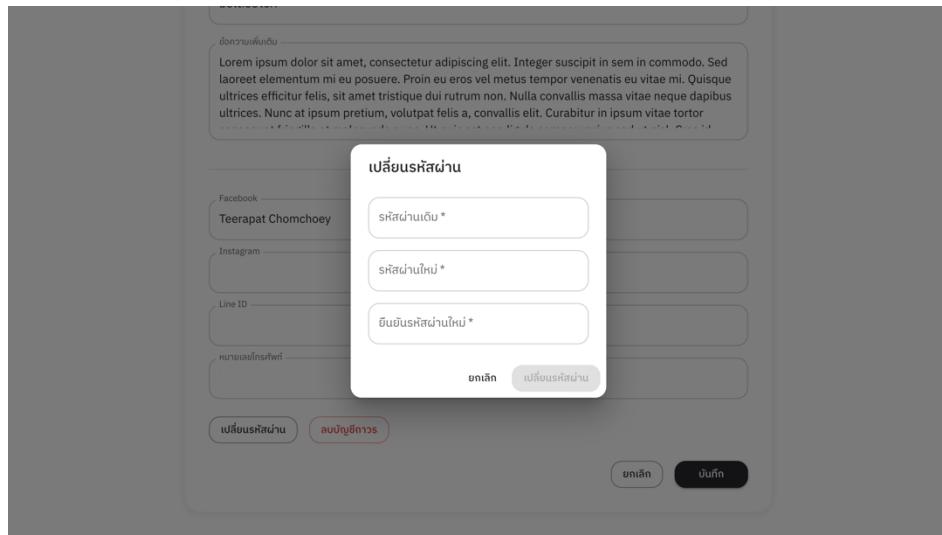
ຮູບທີ 4.3 ເຂົ້າສູ່ຮະບບ

ຈາກຮູບທີ 4.3 ເປັນໜ້າເຂົ້າສູ່ຮະບບ ຜູ້ໃຊ້ທີ່ຍັງໄມ້ໄດ້ສົມມັກສາມາຊີກ ໄທ້ທຳກາຣກດູ່ມື່ມ ສົມມັກເລຍ ເພື່ອສົມມັກສາມາຊີກ ສໍາຫັບຜູ້ໃຊ້ທີ່ເຄີຍສົມມັກສາມາຊີກໄວ້ແລ້ວຈະສາມາດຖື່ຈະກຣອກຂໍ້ອນມູລ ແລ້ວກັດປູ່ມເຂົ້າສູ່ຮະບບເພື່ອເຂົ້າສູ່ຮະບບໄປຢັ້ງໜ້າແຮກຂອງເອົປົລິເຄື່ນ



ຮູບທີ 4.4 ໜ້າລົບບັນຫຼຸດຕົວ

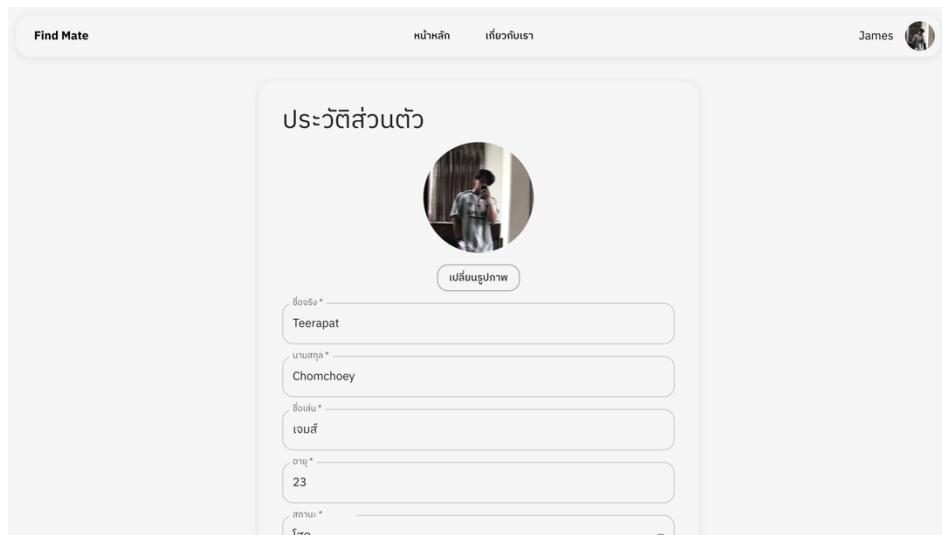
ຈາກຮູບທີ 4.4 ໜ້າລົບບັນຫຼຸດຕົວ ທີ່ຈະສາມາດລົບບັນຫຼຸດຂອງຜູ້ໃຊ້ນັ້ນອອກໄປຈາກຮະບບ ພາກຜູ້ໃຊ້ໄນ້ມີຄວາມຈຳເປັນທີ່ຈະໃຊ້ຮະບບ ຮູ່ໂນໄມ່ຕ້ອງການມີຕ້ວາຕນອຍໃນຮະບບຕ່ອໄປ



รูปที่ 4.5 หน้าเปลี่ยนรหัสผ่าน

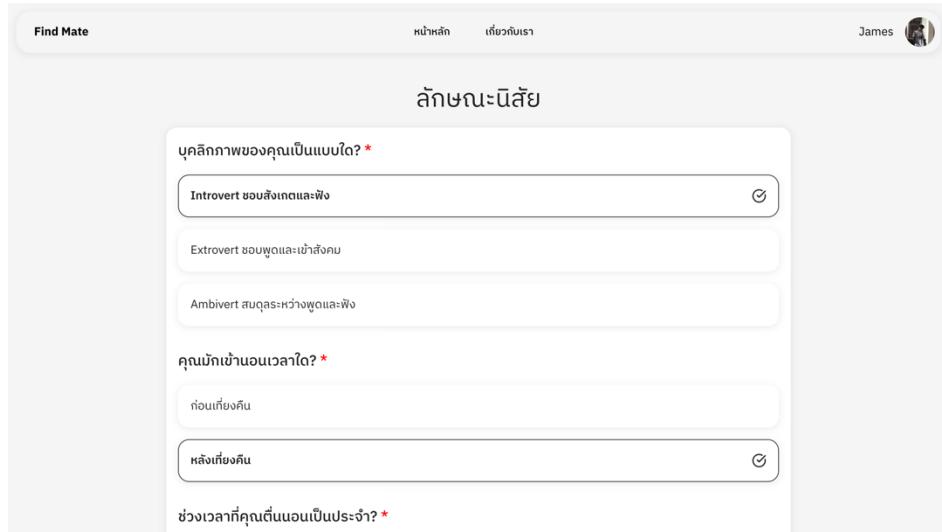
จากรูปที่ 4.5 หน้าเปลี่ยนรหัสผ่าน เมื่อผู้ใช้ต้องการที่จะเปลี่ยนรหัสผ่าน ระบบจะทำ การส่งลิงค์ไปยัง Email ของผู้ใช้ สำหรับนำไปยังหน้าจอสำหรับการเปลี่ยนรหัสผ่าน

#### 4.2.2 จัดเก็บข้อมูลผู้ใช้



รูปที่ 4.6 หน้ากรอกประวัติส่วนตัว

จากรูปที่ 4.6 หน้ากรอกประวัติส่วนตัว ที่จะให้ผู้ใช้กรอกข้อมูลส่วนตัวต่างๆ ที่จำเป็นต่อ การพิจารณาเลือกรูปแบบ ได้แก่ ข้อมูลส่วนตัวเบื้องต้น ข้อมูลสถานศึกษา ข้อมูลหอพัก ข้อมูลการ ติดต่อ โซเชียลมีเดีย และรูปภาพโปรไฟล์



รูปที่ 4.7 หน้าเลือกลักษณะนิสัยส่วนตัว

จากรูปที่ 4.7 หน้าเลือกลักษณะนิสัยส่วนตัว แสดงแบบสอบถามเกี่ยวกับลักษณะนิสัย และกิจวัตรประจำวันในการใช้ชีวิตในหอพัก ที่จะมีตัวเลือกให้ผู้ใช้ในการเลือกตอบคำถาม โดยจะสามารถเลือกตอบได้เพียง 1 ตัวเลือกเท่านั้น



รูปที่ 4.8 หน้าแก้ไขโปรไฟล์

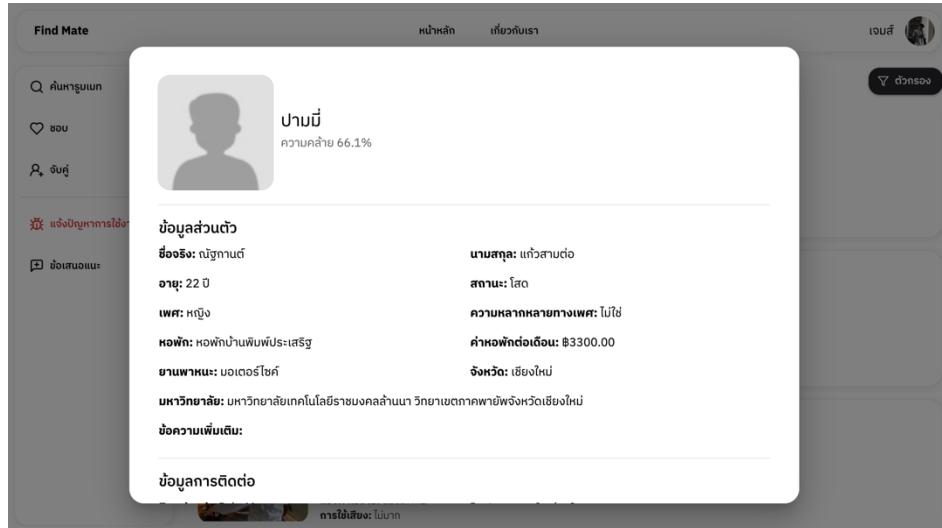
จากรูปที่ 4.8 หน้าแก้ไขโปรไฟล์ ที่จะให้ผู้ใช้ได้ทำการแก้ไขข้อมูลโปรไฟล์ของผู้ใช้ ซึ่ง เป็นข้อมูลที่อิงมาจากขั้นตอนหลังการสมัครสมาชิกครั้งแรก โดยที่จะแบ่งข้อมูลโปรไฟล์เป็น 2 ส่วน คือ แก้ไขประวัติส่วนตัว และแก้ไขลักษณะนิสัย

#### 4.2.3 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้

รูปที่ 4.9 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้

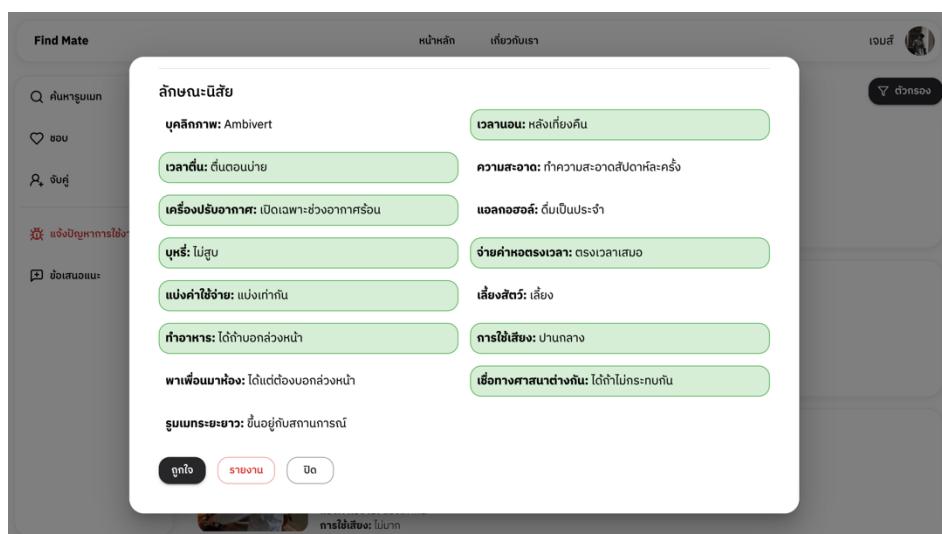
จากรูปที่ 4.9 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้ ที่จะนำข้อมูลลักษณะนิสัย ส่วนตัวของผู้ใช้มาเปรียบเทียบกัน โดยจะแสดงความเข้าได้ของข้อมูลออกมาเป็นเปอร์เซ็นต์ที่จะ เรียงลำดับความเข้ากันได้จากมากไปน้อย มาแสดงในหน้าหลัก

#### 4.2.4 รายละเอียดโปรไฟล์ผู้ใช้



รูปที่ 4.10 รายละเอียดโปรไฟล์ผู้ใช้ส่วนของข้อมูลส่วนตัว

จากรูปที่ 4.10 รายละเอียดโปรไฟล์ผู้ใช้ส่วนของข้อมูลส่วนตัวที่จะแสดงข้อมูลโปรไฟล์ผู้ใช้ที่ผู้ใช้ได้กรอกมาตอนสมัคร ได้แก่ ข้อมูลส่วนตัว ข้อมูลการติดต่อ และลักษณะนิสัย



รูปที่ 4.11 หน้าโปรไฟล์ การเน้นข้อมูลลักษณะนิสัยที่ตรงกัน

จากรูปที่ 4.11 หน้าโปรไฟล์ การเน้นข้อมูลลักษณะนิสัยที่ตรงกัน โดยที่จะเน้น หรือไฮไลท์ข้อความเป็นแบบสี ที่ลักษณะนิสัยที่ตรงกันให้เด่นขึ้น เพื่อที่จะให้ผู้ใช้ได้ดูได้อย่างชัดเจน

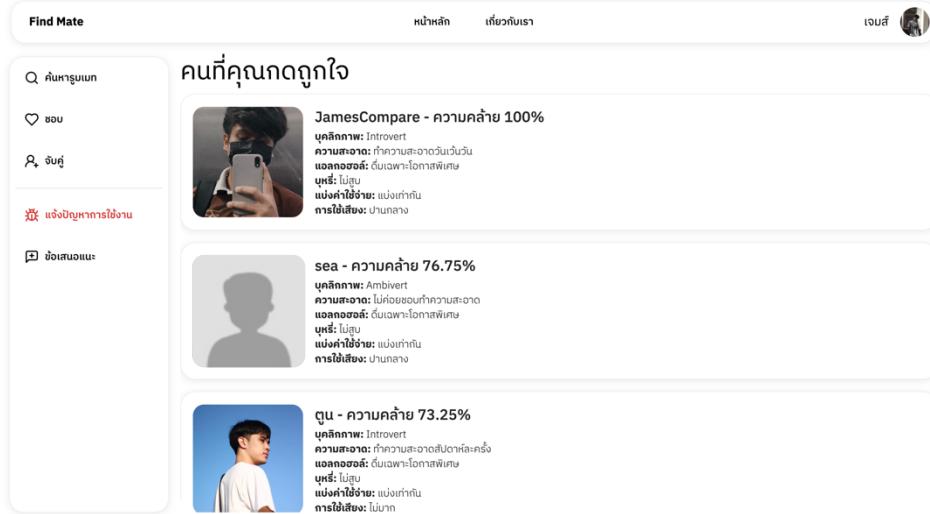
#### 4.2.5 ระบบกรองการค้นหาข้อมูล



รูปที่ 4.12 หน้าต่างระบบตัวกรองการค้นหา

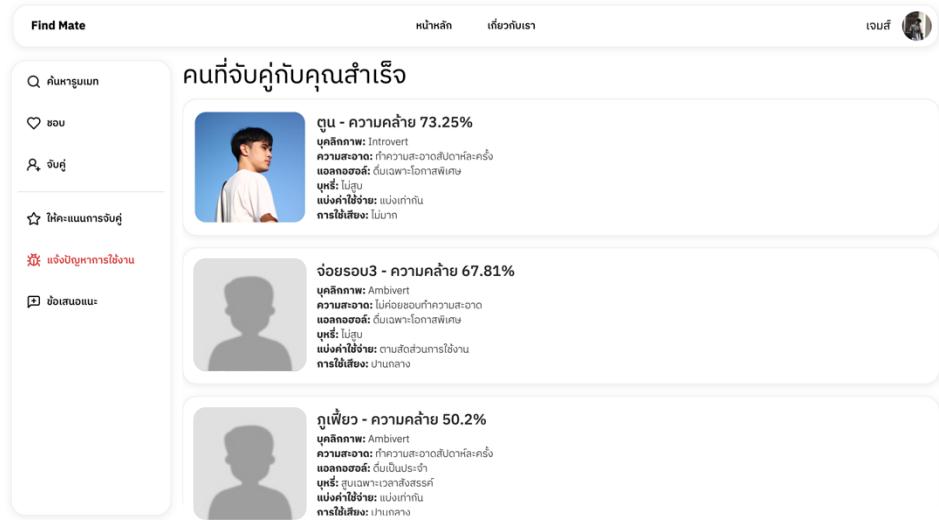
จากรูปที่ 4.12 หน้าต่างระบบตัวกรองการค้นหา ที่จะให้ผู้ใช้ได้ทำการค้นหารูปเมทที่สนใจได้ง่ายขึ้น โดยกดที่ปุ่ม ตัวกรอง จะแสดงหน้าต่างให้ผู้ใช้เลือกตัวเลือกการค้นหาได้ตามที่ต้องการ และมีปุ่มสำหรับล้างการกรองการค้นหา

#### 4.2.6 ระบบการจับคู่



รูปที่ 4.13 หน้าคนที่ถูกใจ

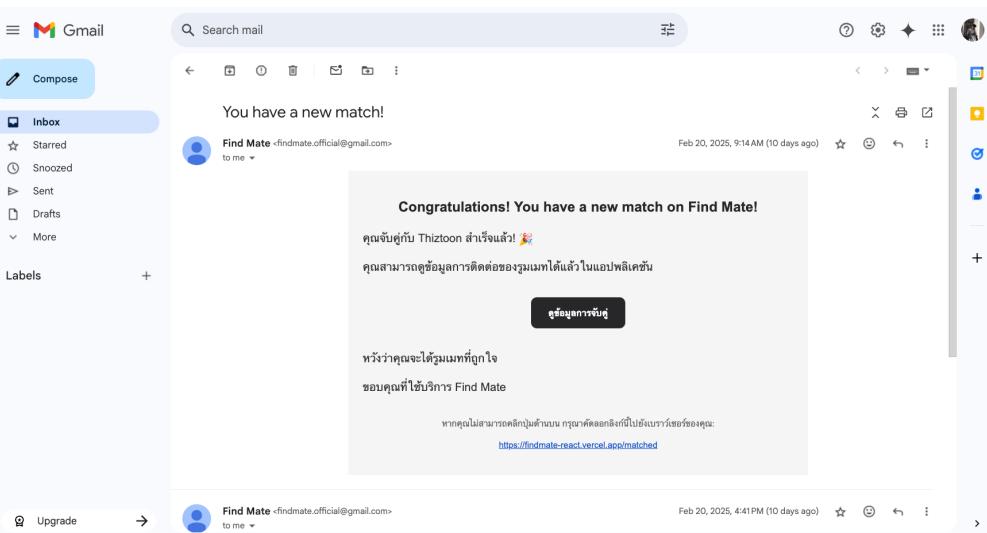
จากรูปที่ 4.13 หน้าคนที่ถูกใจ ก็จะแสดงผู้ใช้ที่คุณได้กดถูกใจในหน้าโปรไฟล์ของผู้ใช้นั้น ๆ เพื่อให้ผู้ใช้ได้เข้ามาดูผู้ใช้ที่เคยได้กดถูกใจไว้ได้ง่ายขึ้นกว่าการค้นหาในหน้าผู้ใช้ทั้งหมด หากผู้ใช้ต้องการยกเลิกถูกใจ ทำได้โดยการเข้าไปหน้าโปรไฟล์ที่ได้กดถูกใจไว้ และกดยกเลิกถูกใจได้



รูปที่ 4.14 หน้าคนที่จับคู่สำเร็จ

จากรูปที่ 4.14 หน้าคนที่จับคู่สำเร็จ จะแสดงผู้ใช้ที่ได้ทำการกดถูกใจคุณกลับ หรือกีดคิอ ผู้ใช้ทั้งสองฝ่ายกดถูกใจซึ่งกันและกัน เมื่อจับคู่สำเร็จໂປຣໄຟລ์ผู้ใช้ของคนที่จับคู่สำเร็จจะแสดงเบอร์ โทรศัพท์มือถือ เพื่อที่จะสามารถติดต่อกันได้สะดวกยิ่งขึ้น

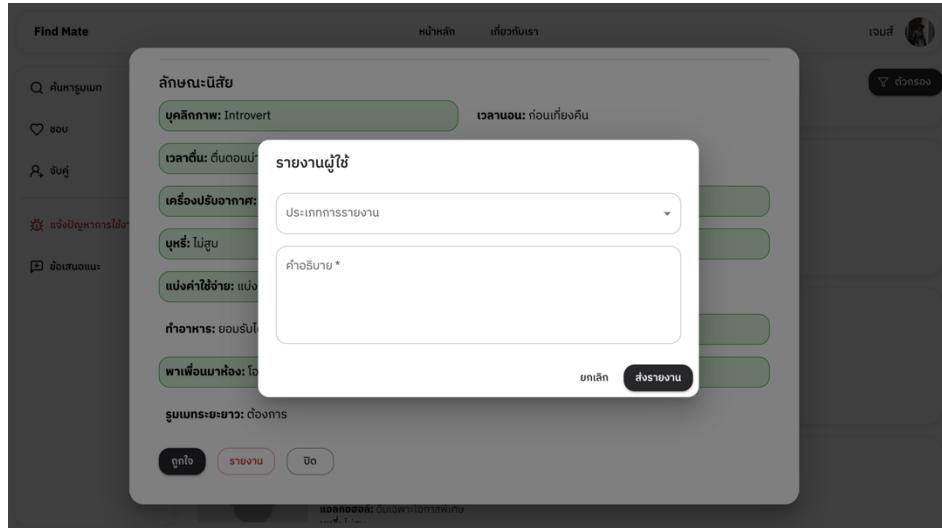
#### 4.2.7 ระบบการแจ้งเตือนการจับคู่สำเร็จ



รูปที่ 4.15 ระบบการแจ้งเตือนการจับคู่สำเร็จ

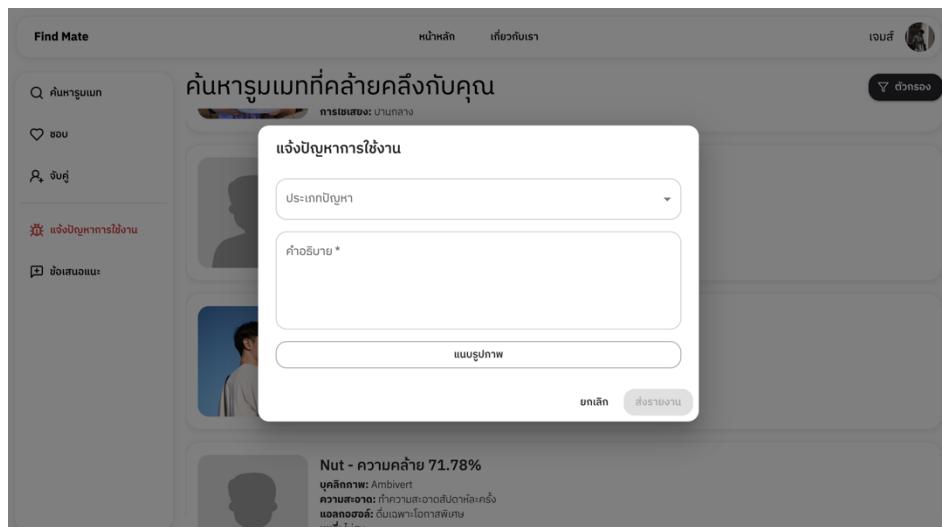
จากรูปที่ 4.15 ระบบการแจ้งเตือนการจับคู่สำเร็จ จะส่งแจ้งเตือนผ่าน Email ของผู้ใช้ ว่าได้ทำการจับคู่สำเร็จ โดยจะมีปุ่ม “ดูข้อมูลการจับคู่” จะนำผู้ใช้ไปยังหน้าจอจับคู่สำเร็จในระบบ

#### 4.2.8 ระบบแจ้งปัญหา



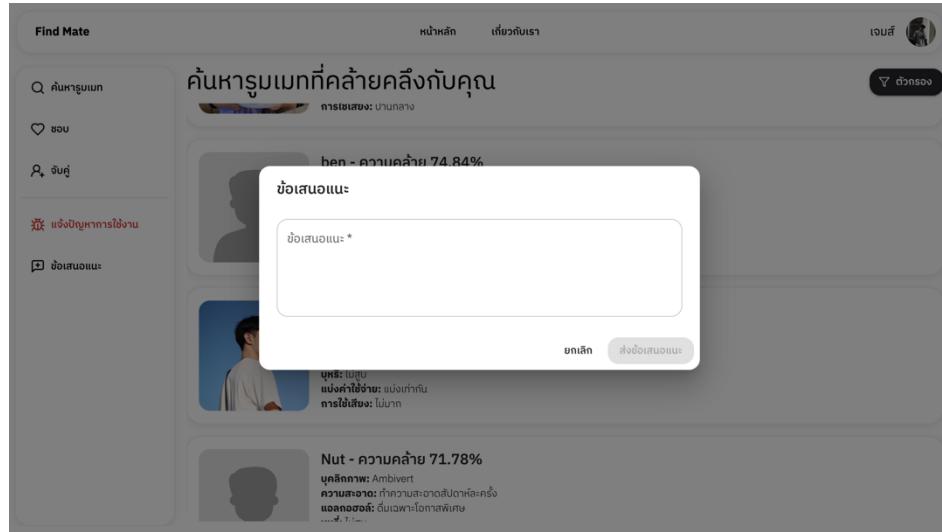
รูปที่ 4.16 การแจ้งปัญหาเกี่ยวกับผู้ใช้

จากรูปที่ 4.16 การแจ้งปัญหาเกี่ยวกับผู้ใช้ เมื่อผู้ใช้กดปุ่ม รายงาน ในหน้าໂປຣໄຟລ໌ອົງ ຜູ້ໃຫ້ນີ້ ຈະແສດງหน้าต่างรายงานຜູ້ໃຊ້ ໂດຍໃຫ້ຜູ້ໃຊ້ໄດ້ທຳການເລືອກປະເທດຮາຍງານ ເຊິ່ນ ຄຳອັບຍາຍຮາຍງານແລະແນບຮູ່ປາພຫລັກຮູ້ນ ໃຫ້ແກ່ຜູ້ດູແລະບົບເພີ້ມຕາມເໝາະສົມ



รูปที่ 4.17 การแจ้งปัญหาเกี่ยวกับระบบการใช้งาน

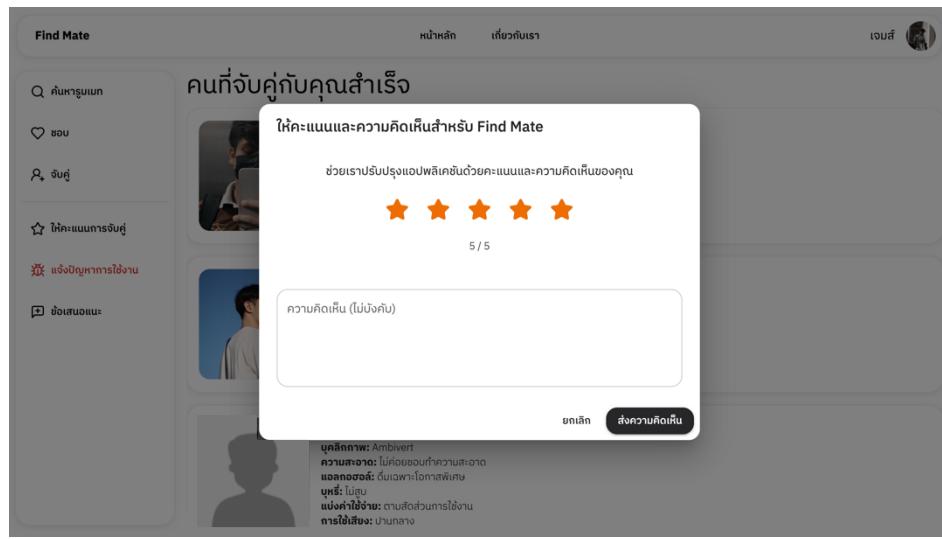
จากรูปที่ 4.17 การแจ้งปัญหาเกี่ยวกับระบบการใช้งาน เมื่อຜູ້ໃຊ້กดปุ่ມ ແຈ້ງປັນຫາການໃໝ່ ຈາກນີ້ ຈະແສດງหน้าຫຼັກຂອງຮະບບ ຮະບບຈະແສດງหน້າຕ່າງຮາຍງານຜູ້ໃຊ້ ໂດຍໃຫ້ຜູ້ໃຊ້ໄດ້ທຳການເລືອກປະເທດຮາຍງານ ແລ້ວເພີ້ມຕາມຄວາມຮູ່ປາພຫລັກຮູ້ນ ໃຫ້ແກ່ຜູ້ດູແລະບົບເພີ້ມຕາມເໝາະສົມ



รูปที่ 4.18 การให้ข้อเสนอแนะเกี่ยวกับระบบ

จากรูปที่ 4.18 การให้ข้อเสนอแนะเกี่ยวกับระบบ เมื่อผู้ใช้กดปุ่ม ข้อเสนอแนะ ในหน้าหลักของระบบ ระบบจะแสดงหน้าต่างการส่งข้อเสนอแนะ โดยให้ผู้ใช้ได้ทำการเขียนคำอธิบายข้อเสนอแนะ เพื่อส่งไปยังผู้ดูแลระบบ เพื่อตรวจสอบและแก้ไขตามความเหมาะสม

#### 4.2.9 ระบบให้คะแนนความพึงพอใจ



รูปที่ 4.19 ระบบให้คะแนนความพึงพอใจ

จากรูปที่ 4.19 ระบบให้คะแนนความพึงพอใจ โดยผู้ใช้งานสามารถให้คะแนนได้ในหน้าจัดคุณภาพ โดยมีระดับการให้คะแนน 1 ถึง 5 ดาว

#### 4.2.10 ระบบผู้ดูแล

The screenshot shows a web-based application titled 'Find Mate'. At the top, there are three tabs: 'หน้าหลัก' (Home), 'เพื่อนของเรา' (Our Friends), and 'Admin' with a user icon. Below the tabs, the title 'แดชบอร์ดผู้ดูแลระบบ' (System Moderator Dashboard) is displayed.

Three search filters are shown in boxes:

- 25** จำนวนผู้ใช้กั้งหมัด
- 8** จำนวนการจับคู่กั้งหมัด
- 13** จำนวนรายงานเก็บหมัด

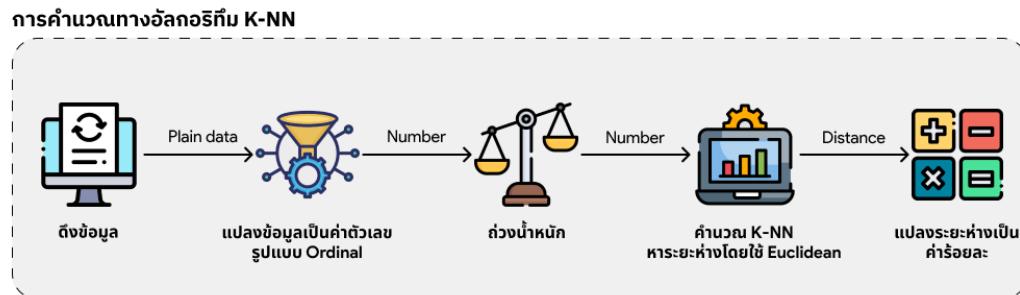
The main content area is a table titled 'รายงานผู้ใช้' (Report on Users). It includes columns: ผู้รายงาน (Reporter), ผู้ถูกรายงาน (Reported User), ประเภท (Type), สถานะ (Status), วันที่ (Date), and การจัดการ (Action). The table contains five rows of data:

ผู้รายงาน	ผู้ถูกรายงาน	ประเภท	สถานะ	วันที่	การจัดการ
James	Ozoe		pending	5/3/2568	<button>กราบขออภัย</button> <button>ร้องกําฟ้อง</button> <button>ลบรายงาน</button>
James	seatuan	scam	pending	5/3/2568	<button>กราบขออภัย</button> <button>ร้องกําฟ้อง</button> <button>ลบรายงาน</button>
James	leow	fake_profile	resolved	20/2/2568	<button>กราบขออภัย</button> <button>ร้องกําฟ้อง</button> <button>ลบรายงาน</button>
James	8003	other	pending	19/2/2568	<button>กราบขออภัย</button> <button>ร้องกําฟ้อง</button> <button>ลบรายงาน</button>

รูปที่ 4.20 หน้าแดชบอร์ดของผู้ดูแลระบบ

จากรูปที่ 4.20 หน้าแดชบอร์ดของผู้ดูแลระบบ จะเป็นหน้าที่แสดงจำนวนผู้ใช้งานในระบบทั้งหมด จำนวนการจับคู่ทั้งหมด และจำนวนการรายงานจากผู้ใช้ โดยผู้ดูแลระบบจะสามารถดูรายละเอียดการรายงานเกี่ยวกับการรายงานผู้ใช้ การรายงานระบบ ข้อเสนอแนะ ดูการให้คะแนน สามารถปรับสถานะบัญชีผู้ใช้ รวมไปถึงสามารถลบบัญชีผู้ใช้ได้

### 4.3 การคำนวณทางอัลกอริทึม K-NN



รูปที่ 4.21 การคำนวณทางอัลกอริทึม K-NN

ใช้หลักการ K-NN เพื่อการคำนวณความคล้ายคลึงระหว่างผู้ใช้งาน โดยมุ่งเน้นการใช้ระยะห่างแบบยุคลิด (Euclidean Distance) แทนการจำแนกประเภท (Classification) ได้ผลลัพธ์เป็นค่าระยะห่าง เพื่อนำไปคำนวณร้อยละความใกล้เคียงของแต่ละผู้ใช้งานอื่นเทียบกับผู้ใช้งานขณะนั้น

#### 4.3.1 กระบวนการคำนวณ

##### 4.3.1.1 ดึงข้อมูลผู้ใช้จากฐานข้อมูล

ซึ่งจะได้ข้อมูลในรูปแบบตารางข้อมูลแบบข้อความล้วน (Plain Text) ที่ยังไม่สามารถนำไปประมวลผลทางคณิตศาสตร์ได้โดยตรง

ตารางที่ 4.1 ตัวอย่างข้อมูลลักษณะนิสัยผู้ใช้จากฐานข้อมูล

<b>id</b>	<b>type</b>	<b>sleep</b>	<b>wake</b>	<b>clean</b>	<b>...</b>	<b>period</b>
1	introvert	after_midnight	noon	every_other_day	...	no_need
2	extrovert	after_midnight	morning	every_day	...	need
3	introvert	after_midnight	morning	every_day	...	no_need

##### 4.3.1.2 ทำการค้นหาเฉพาะผู้ใช้ปัจจุบัน

ตารางที่ 4.2 ตัวอย่างข้อมูลเฉพาะผู้ใช้ปัจจุบัน

<b>id</b>	<b>type</b>	<b>sleep</b>	<b>wake</b>	<b>clean</b>	<b>...</b>	<b>period</b>
2	extrovert	after_midnight	morning	every_day	...	need

#### 4.3.1.3 แปลงข้อมูลผู้ใช้อื่นเป็นค่าตัวเลข

แปลงข้อมูลคุณลักษณะแต่ละด้านเป็นค่าตัวเลข เพื่อสามารถนำไปคำนวณในสมการได้ โดยคุณด้วยค่าถ่วงน้ำหนักตามที่กำหนดไว้ และบันทึกลงในชุดข้อมูล (Dataset) เพื่อเตรียมสำหรับการคำนวณระยะห่างแบบยูคลิดีเดียนต่อไป

```
Dataset: [
    [
        0,     1,     1,     2,     3,     1.8,   0,     0,     0,     0,     1,
        3.6,   0,     2.4,   2,
    ],
    [
        2,     1,     1,     2,     3,     3.6,   0,     0,     0,     1.3,   1,
        3.6,   3,     2.4,   2,
    ],
]
```

#### 4.3.1.4 แปลงข้อมูลเป็นผู้ใช้ปัจจุบันเป็นค่าตัวเลข

แปลงข้อมูลคุณลักษณะแต่ละด้านของผู้ใช้ปัจจุบันเป็นค่าตัวเลข เพื่อสามารถนำไปคำนวณในสมการได้ โดยคุณด้วยค่าถ่วงน้ำหนักตามที่กำหนดไว้ เช่นเดียวกัน เพื่อใช้ในการคำนวณเปรียบเทียบกับข้อมูลผู้ใช้อื่น ๆ โดยใช้ระยะห่างแบบยูคลิดีเดียน

```
Current User [
    [
        0,     1,     1,     2,     3,     1.8,   0,     0,     0,     0,     1,
        3.6,   0,     2.4,   2,
    ],
]
```

#### 4.3.1.5 คำนวณระยะห่าง

$$d(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (4.1)$$

โดยที่	$d$	คือ ผลลัพธ์ระยะทางจากจุด $p$ ไปยังจุด $q$
	$p, q$	คือ จุดข้อมูล
	$n$	คือ จำนวนมิติของข้อมูล
	$\Sigma$	คือ ผลรวมระยะห่างกำลังสอง

ดำเนินการคำนวณระยะห่างระหว่างค่าตัวเลขของผู้ใช้ปัจจุบันกับค่าตัวเลขของผู้ใช้อื่น ๆ ทั้งหมด ผลลัพธ์ที่ได้คือค่าระยะห่างของข้อมูล โดยจำนวนของผลลัพธ์จะเท่ากับจำนวนผู้ใช้ที่นำมาคำนวณ

Distances: [  
0, 4.23438307194803  
]

#### 4.3.1.6 แปลงระยะห่างเป็นร้อยละความคล้ายคลึง

$$\text{Similarity} = 100 \times \left(1 - \frac{\text{Distance}}{\text{Max Dixtance}}\right) \quad (4.3)$$

โดยที่	$Similarity$	คือ ความคล้ายระหว่างบุคคลผลลัพธ์เป็นร้อยละ
	$Distance$	คือ ระยะห่างระหว่างบุคคลที่คำนวณได้จาก Euclidean Distance
	$Max Dixtance$	คือ ระยะห่างของข้อมูลสูงสุดที่เป็นไปได้ตามจำนวนคุณลักษณะ

จะได้รูปแบบดังนี้

Similarity: 100

Similarity: 65.93

#### 4.3.1 ตัวอย่างข้อมูลในระบบจริง

การทดสอบใช้ข้อมูลของผู้ใช้ชื่อ James เป็นผู้ใช้ปัจจุบัน เพื่อเปรียบเทียบกับผู้ใช้อื่น ส่วนรายได้แก่ JamesCompare และ Alpha โดยมีสมมติฐานดังนี้

ผู้ใช้ JamesCompare มีลักษณะนิสัยเหมือนกับ James ทุกประการ เพื่อแสดงให้เห็น กรณีที่ค่าการคำนวณให้ผลลัพธ์เป็นความคล้ายคลึงสูงสุด

ผู้ใช้ Alpha เป็นตัวแทนของผู้ใช้งานทั่วไปที่อาจพบได้ในสภาพแวดล้อมการใช้งานจริง

ตารางที่ 4.3 ตัวอย่างข้อมูลในระบบจริง

คุณลักษณะ	ข้อมูลบุคคล		
	James	Compare	Alpha
บุคลิกภาพ	Introvert	Introvert	Extrovert
เวลาเข้าอน	หลังเที่ยงคืน	หลังเที่ยงคืน	หลังเที่ยงคืน
เวลาตื่น	ตื่นตอนบ่าย	ตื่นตอนบ่าย	ตื่นตอนเช้า
ความสะอาด	ทำความสะอาดวันเว้นวัน	ทำความสะอาดวันเว้นวัน	ทำความสะอาดวันเว้นวัน
การใช้เครื่องปรับอากาศ	เปิดเฉพาะช่วงอากาศร้อน	เปิดเฉพาะช่วงอากาศร้อน	เปิดเฉพาะช่วงอากาศร้อน
การดื่มแอลกอฮอล์	ดื่มเฉพาะโอกาสพิเศษ	ดื่มเฉพาะโอกาสพิเศษ	ดื่มช่วงสุดสัปดาห์
การสูบบุหรี่	ไม่สูบ	ไม่สูบ	ไม่สูบ
การจ่ายเงินตรงเวลา	ตรงเวลาเสมอ	ตรงเวลาเสมอ	ตรงเวลาเสมอ
การแบ่งค่าใช้จ่ายหอพัก	แบ่งเท่ากัน	แบ่งเท่ากัน	แบ่งเท่ากัน
การเลี้ยงสัตว์	ไม่เลี้ยง	ไม่เลี้ยง	เลี้ยง
การทำอาหารกลิ่นแรง ในห้อง	ได้ถ้าบอกกล่าวหน้า	ได้ถ้าบอกกล่าวหน้า	ได้ถ้าบอกกล่าวหน้า
การใช้เสียง	ปานกลาง	ปานกลาง	ปานกลาง
การพาเพื่อนมาห้อง	โโอดี	โโอดี	ไม่โโอดี
ความเชื่อทางศาสนาที่แตกต่างกัน	ได้ถ้าไม่กระทบกัน	ได้ถ้าไม่กระทบกัน	ได้ถ้าไม่กระทบกัน
ความต้องการรูมเมทในระยะยาว	ไม่จำเป็น	ไม่จำเป็น	ไม่จำเป็น

### 4.3.2 เทียบข้อมูลเป็นตัวแปรที่ใช้งานจริง

ตารางที่ 4.4 เทียบข้อมูลเป็นตัวแปรที่ใช้งานจริง

คุณลักษณะ	ข้อมูลตัวแปรบุคคล		
	James	Compare	Alpha
type	introvert	introvert	extrovert
sleep	after_midnight	after_midnight	after_midnight
wake	noon	noon	morning
clean	every_other_day	every_other_day	every_other_day
air_conditioner	only_hot	only_hot	only_hot
drink	special	special	weekend
smoke	never	never	never
money	on_time	on_time	on_time
expense	half	half	half
pet	dont_have	dont_have	have
cook	tell_first	tell_first	tell_first
loud	medium	medium	medium
friend	ok	ok	no
religion	no_effect	no_effect	no_effect
period	no_need	no_need	no_need

#### 4.3.3 การกำหนดค่าตัวแปรในการคำนวณ

ระบบดำเนินการแปลงข้อมูลให้อยู่ในรูปแบบค่าตัวเลข โดยใช้วิธีการเข้ารหัสแบบ Ordinal Encoding จัดเรียงลำดับข้อมูลโดยมีความสัมพันธ์ต่อกัน จากตาราง 3.11 การแปลงข้อมูลโดยใช้การเข้ารหัสแบบ Ordinal Encoding หน้า 71

ตารางที่ 4.5 การกำหนดค่าตัวแปรในการคำนวณ

คุณลักษณะ	ข้อมูลแต่ละบุคคล		
	James	Compare	Alpha
type	0	0	2
sleep	1	1	1
wake	1	1	0
clean	1	1	1
air_conditioner	2	2	2
drink	1	1	2
smoke	0	0	0
money	0	0	0
expense	0	0	0
pet	0	0	1
cook	1	1	1
loud	2	2	2
friend	0	0	2
religion	2	2	2
period	2	2	2

#### 4.3.4 การกำหนดค่าถ่วงน้ำหนักของคุณลักษณะแต่ละด้าน

การกำหนดค่าถ่วงน้ำหนักมีความสำคัญอย่างยิ่ง เนื่องจากคุณลักษณะแต่ละด้านมีระดับของผลกระทบต่อความสัมพันธ์ในการอยู่ร่วมกันที่แตกต่างกัน ค่าถ่วงน้ำหนักกำหนดตามที่ระบุไว้ในหัวข้อ 3.5.2 การกำหนดค่าถ่วงน้ำหนักของคุณลักษณะแต่ละด้าน หน้า 68

ตารางที่ 4.6 การกำหนดค่าถ่วงน้ำหนักของคุณลักษณะแต่ละด้าน

คุณลักษณะ	ข้อมูลแต่ละบุคคล		
	James	Compare	Alpha
type	0	0	2
sleep	1	1	1
wake	1	1	0
clean	2	2	2
air_conditioner	3	3	3
drink	1.8	1.8	3.6
smoke	0	0	0
money	0	0	0
expense	0	0	0
pet	0	0	1.3
cook	1	1	1
loud	3.6	3.6	3.6
friend	0	0	3
religion	2.4	2.4	2.4
period	2	2	2

#### 4.3.5 คำนวณระยะห่างสูงสุดที่เป็นไปได้โดยใช้ Euclidean Distance

ระบบทราบว่าระยะห่างที่น้อยที่สุดคือศูนย์ ซึ่งเกิดขึ้นเมื่อข้อมูลของทุกคุณลักษณะมีความเหมือนกันทั้งหมด กล่าวคือ ข้อมูลมีความคล้ายคลึงกัน 100% หรือมีระยะห่าง 0%

สำหรับการคำนวณระยะห่างสูงสุดที่เป็นไปได้ ระบบพิจารณากรณีที่แยกที่สุดที่ทำให้ผลลัพธ์ความคล้ายคลึงเป็น 0% หรือมีระยะห่าง 100% ตัวอย่างของกรณีเหลวร้ายที่สุด มีดังนี้

$$\text{คำถาม 2 ตัวเลือก} \quad A = 0 \quad B = 1 \\ 0,1$$

$$\text{คำถาม 3 ตัวเลือก} \quad A = 0 \quad B = 2 \\ 0,1,2$$

$$\text{คำถาม 4 ตัวเลือก} \quad A = 0 \quad B = 3 \\ 0,1,2,3$$

คำนวณระยะห่างสูงสุดที่เป็นไปได้โดย คำนวณร่วมกับการทำหนดค่าต่างน้ำหนักของคุณลักษณะแต่ละด้าน

$$d(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (4.1)$$

โดยที่	$d$	คือ ผลลัพธ์ระยะทางจากจุด $p$ ไปยังจุด $q$
	$p, q$	คือ จุดข้อมูล
	$n$	คือ จำนวนมิติของข้อมูล
	$\Sigma$	คือ ผลรวมระยะห่างกำลังสอง

#### ตารางที่ 4.7 คำนวณระยะห่างสูงสุดที่เป็นไปได้

#### 4.3.6 คำนวณระยะห่างระหว่างผู้ใช้โดยใช้ Euclidean Distance

ดำเนินการคำนวณระยะห่างระหว่างผู้ใช้ James กับผู้ใช้ JamesCompare และ  
ระหว่างผู้ใช้ James กับผู้ใช้ Alpha เพื่อแสดงให้เห็นตัวอย่างของการคำนวณอย่างเป็นรูปธรรม

$$d(p, q) = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (4.2)$$

โดยที่	$d$	คือ ผลลัพธ์ระยะทางจากจุด $p$ ไปยังจุด $q$
	$p, q$	คือ จุดข้อมูล
	$n$	คือ จำนวนมิติของข้อมูล
	$\Sigma$	คือ ผลรวมระยะห่างกำลังสอง

ตารางที่ 4.8 คำนวณระยะห่างระหว่างผู้ใช้ James กับ JamesCompare

ตารางที่ 4.9 คำนวณระยะห่างระหว่างผู้ใช้ James กับ Alpha

#### 4.3.7 การคำนวณร้อยละความเข้ากันได้จากระยะห่าง

$$\text{Similarity} = 100 \times \left(1 - \frac{\text{Distance}}{\text{Max Distance}}\right) \quad (4.3)$$

โดยที่ **Similarity** คือ ความคล้ายระหว่างบุคคลผลลัพธ์เป็นร้อยละ  
**Distance** คือ ระยะห่างระหว่างบุคคลที่คำนวณได้จาก Euclidean Distance  
**Max Distance** คือ ระยะห่างของข้อมูลสูงสุดที่เป็นไปได้ตามจำนวนคุณลักษณะ

จากการคำนวณค่าระยะห่างสูงสุดที่เป็นไปได้ เท่ากับ 12.77 ดังนั้น

$$\text{Max Distance} = 12.77 \quad (4.4)$$

จากการคำนวณระยะห่างระหว่างผู้ใช้ James กับผู้ใช้ JamesCompare ได้ระยะห่างของข้อมูลเท่ากับ 0 นำมาแทนค่าในสมการข้างต้น

$$\text{Similarity} = 100 \times \left(1 - \frac{0}{12.77}\right) \quad (4.5)$$

$$\text{Similarity} = 100$$

ฉะนั้นร้อยละความเข้ากันได้ของผู้ใช้ James และผู้ใช้ JamesCompare เท่ากับ 100%

จากการคำนวณระยะห่างระหว่าง James กับผู้ใช้ Alpha ได้ระยะห่างของข้อมูลเท่ากับ 4.35 นำมาแทนค่าในสมการข้างต้น

$$\text{Similarity} = 100 \times \left(1 - \frac{4.35}{12.77}\right) \quad (4.6)$$

$$\text{Similarity} = 65.93$$

ฉะนั้นร้อยละความเข้ากันได้ของผู้ใช้ James และผู้ใช้ Alpha เท่ากับ 65.93%

ระบบจะเปรียบเทียบค่าของผู้ใช้ James กับผู้ใช้งานอื่นทั้งหมด จากนั้นแสดงผลผู้ใช้ทั้งหมดเรียงตามลำดับจากมากไปน้อย โดยไม่พิจารณาค่า K เนื่องจากเป็นการประยุกต์ใช้หลักการของ K-NN เนื่องในส่วนของการคำนวณระยะห่าง โดยไม่ได้นำไปใช้ในการจำแนกประเภท

#### 4.4 การประเมินแบบสอบถาม

เว็บแอปพลิเคชันหารูមเมทด้วย K-NN ได้ทำการทดลอง และแจกแบบฟอร์มสอบถามให้ผู้ทดลองใช้งานเพื่อเก็บข้อมูลจำนวน 10 คน โดยมีวิธีการประเมินผลดังนี้

##### 4.4.1 การประเมินความพึงพอใจในการใช้งานระบบ

###### 4.4.1.1 ระบบใช้งานง่าย สะดวก

###### 4.4.1.2 ด้านการออกแบบ Design และการจัดรูปแบบแอปพลิเคชัน

###### 4.4.1.3 การจัดรูปแบบของแอปพลิเคชันง่ายต่อการอ่านและการใช้งาน

###### 4.4.1.4 ความเหมาะสมในการใช้สัญลักษณ์ในการสื่อความหมาย

โดยประเมินจากให้ผู้ใช้งานระบบอย่างน้อยจำนวน 10 คน แล้วทำการกรอกแบบประเมินต้องพบร่วมกับผู้ประเมินพึงพอใจในการใช้งานระบบไม่ต่ำกว่าร้อยละ 70 ของจำนวนผู้ประเมิน

##### 4.4.2 การประเมินประสิทธิภาพในการทำงาน

4.4.2.1 ระบบสามารถจับคู่ผู้ใช้ที่กดถูกใจตรงกันและแสดงหมายเลขอรหัสที่ของผู้ใช้ได้อย่างถูกต้อง

###### 4.4.2.2 ระบบสามารถแสดงรายการการค้นหาได้เป็นลำดับอย่างถูกต้อง

4.4.2.3 แอปพลิเคชันสามารถทำงานได้อย่างราบรื่นและไม่เกิดปัญหาหรือความล่าช้าในขณะที่ใช้งาน

โดยประเมินจากให้ผู้ใช้งานระบบอย่างน้อยจำนวน 10 คน แล้วทำการกรอกแบบประเมินต้องพบร่วมกับผู้ประเมินพึงพอใจในการใช้งานระบบไม่ต่ำกว่าร้อยละ 70 ของจำนวนผู้ประเมิน

##### 4.4.3 การประเมินด้านประโยชน์และการนำไปใช้

###### 4.4.3.1 แอปพลิเคชันมีประโยชน์ต่อผู้ใช้งาน

###### 4.4.3.2 แอปพลิเคชันตอบโจทย์ความต้องการของผู้ใช้

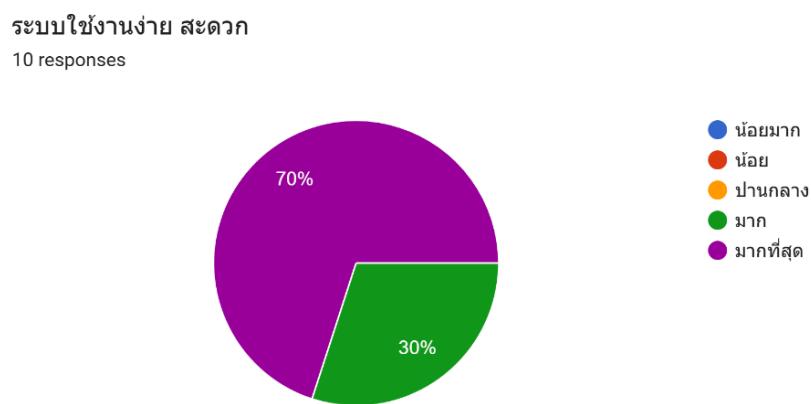
###### 4.4.3.3 แอปพลิเคชันมีผลต่อการนำไปใช้ในชีวิตประจำวัน

โดยประเมินจากให้ผู้ใช้งานระบบอย่างน้อยจำนวน 10 คน แล้วทำการกรอกแบบประเมินต้องพบร่วมกับผู้ประเมินพึงพอใจในการใช้งานระบบไม่ต่ำกว่าร้อยละ 70 ของจำนวนผู้ประเมิน

## 4.5 สรุปผลการประเมินแบบเป็นแผนภูมิ

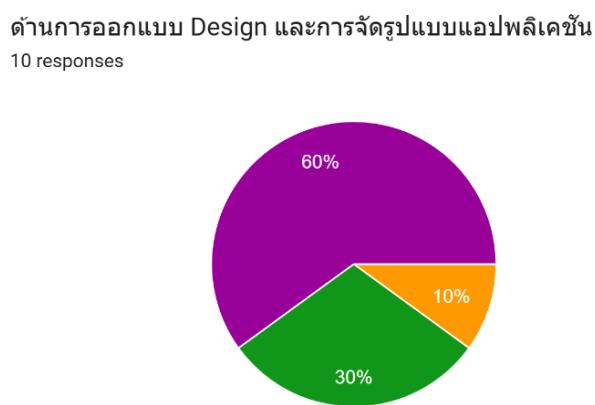
### 4.4.1 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบ

โดยประเมินจากให้ผู้ใช้งานระบบ จำนวน 10 คน และทำการกรอกแบบประเมินต้องพบว่าผู้ประเมินพึงพอใจในการใช้งานระบบไปแล้วกว่าร้อยละ 70 ของจำนวนผู้ประเมิน



รูปที่ 4.22 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบ ข้อที่ 1

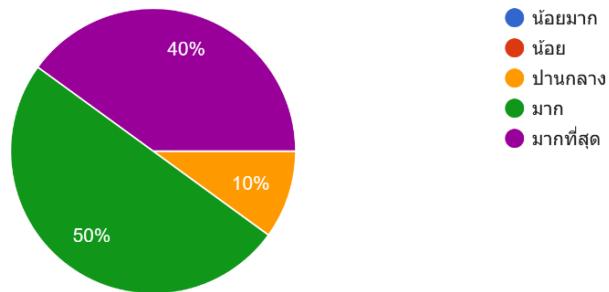
จากรูปที่ 4.22 พบร่วมกันว่าผู้ประเมินมีความพึงพอใจในระบบใช้งานง่าย สะดวก ระดับมากที่สุด 70% และระดับมาก 30% ของผู้ประเมินทั้งหมด



รูปที่ 4.23 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบ ข้อที่ 2

จากรูปที่ 4.23 พบร่วมกันว่าผู้ประเมินมีความพึงพอใจในระบบด้านการออกแบบ Design และการจัดรูปแบบแอปพลิเคชัน ระดับมากที่สุด 60% ระดับมาก 30% และระดับปานกลาง 10% ของผู้ประเมินทั้งหมด

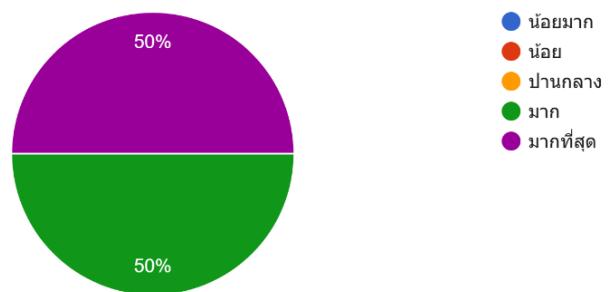
การจัดรูปแบบของแอปพลิเคชันง่ายต่อการอ่านและการใช้งาน  
10 responses



รูปที่ 4.24 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบ ข้อที่ 3

จากรูปที่ 4.24 พบร่วมกันว่าผู้ประเมินมีความพึงพอใจในการจัดรูปแบบของแอปพลิเคชันง่ายต่อการอ่านและการใช้งาน ระดับมากที่สุด 40% ระดับมาก 50% และระดับปานกลาง 10% ของผู้ประเมินทั้งหมด

ความเหมาะสมในการใช้สัญลักษณ์ในการสื่อความหมาย  
10 responses



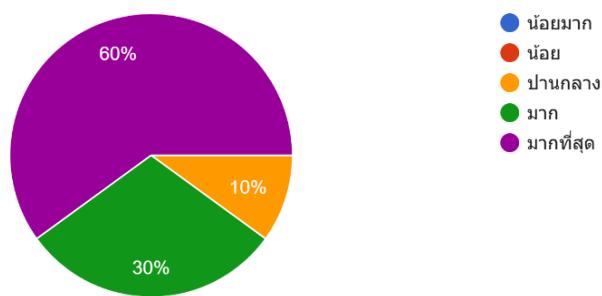
รูปที่ 4.25 แผนภูมิการประเมินความพึงพอใจในการใช้งานระบบ ข้อที่ 4

จากรูปที่ 4.25 พบร่วมกันว่าผู้ประเมินมีความพึงพอใจในระบบความเหมาะสมในการใช้สัญลักษณ์ในการสื่อความหมาย ระดับมากที่สุด 50% และระดับมาก 50% ของผู้ประเมินทั้งหมด

#### 4.4.2 การประเมินความพึงพอใจในการใช้งานระบบ

โดยประเมินจากให้ผู้ใช้งานระบบ จำนวน 10 คน แล้วทำการกรอกแบบประเมิน ต้องพบว่าผู้ประเมินพึงพอใจในประสิทธิภาพในการทำงานไม่ต่ำกว่าร้อยละ 70 ของจำนวนผู้ประเมิน

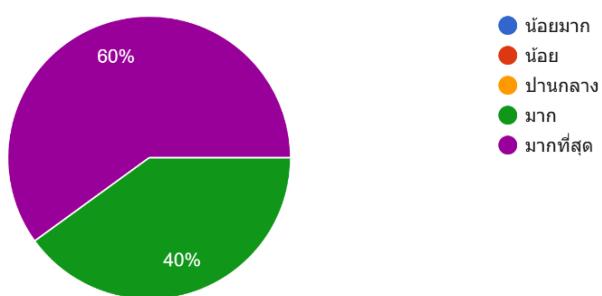
ระบบสามารถจับคู่ใช้ที่กดถูกใจตรงกันอย่างถูกต้อง  
10 responses



รูปที่ 4.26 แผนภูมิการประเมินประสิทธิภาพในการทำงาน ข้อที่ 1

จากรูปที่ 4.26 พบร่วมกันว่าผู้ประเมินประเมินประสิทธิภาพในการทำงาน ระบบสามารถจับคู่ใช้ที่กดถูกใจตรงกันอย่างถูกต้อง ระดับมากที่สุด 60% ระดับมาก 30% และระดับปานกลาง 10% ของผู้ประเมินทั้งหมด

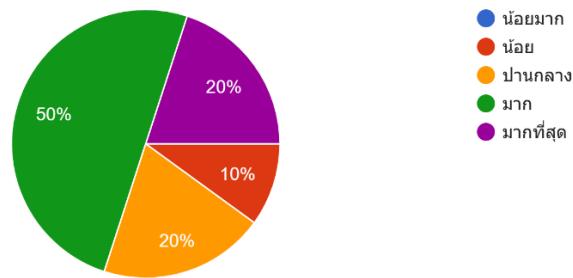
ระบบสามารถแสดงรายการการค้นหาได้เป็นลำดับอย่างถูกต้อง  
10 responses



รูปที่ 4.27 แผนภูมิการประเมินประสิทธิภาพในการทำงาน ข้อที่ 2

จากรูปที่ 4.27 พบร่วมกันว่าผู้ประเมินประเมินประสิทธิภาพในการทำงาน ระบบสามารถแสดงรายการการค้นหาได้เป็นลำดับอย่างถูกต้อง ระดับมากที่สุด 60% และระดับมาก 40% ของผู้ประเมินทั้งหมด

แอปพลิเคชันสามารถทำงานได้อย่างราบรื่นและไม่เกิดปัญหาหรือความล่าช้าในขณะที่ใช้งาน  
10 responses



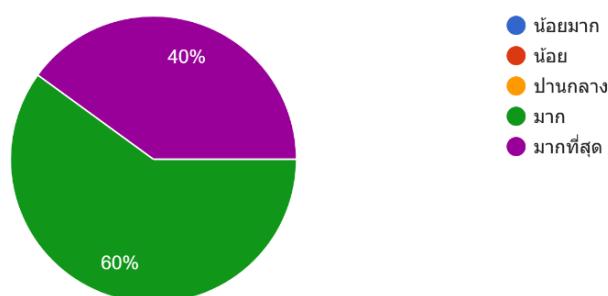
รูปที่ 4.28 แผนภูมิการประเมินประสิทธิภาพในการทำงาน ข้อที่ 3

จากรูปที่ 4.28 พบร่วมกันว่า ผู้ประเมินประเมินประสิทธิภาพในการทำงาน แอปพลิเคชันสามารถทำงานได้อย่างราบรื่นและไม่เกิดปัญหาหรือความล่าช้าในขณะที่ใช้งาน ระดับมากที่สุด 20% ระดับมาก 50% ระดับปานกลาง 25% และระดับน้อย 10% ของผู้ประเมินทั้งหมด

#### 4.4.3 การประเมินด้านประโยชน์และการนำไปใช้

โดยประเมินจากให้ผู้ใช้งานระบบ จำนวน 10 คน แล้วทำการกรอกแบบประเมิน ต้องพบร่วมกันว่า ผู้ประเมินพึงพอใจในด้านประโยชน์และการนำไปใช้ไม่ต่างกับร้อยละ 70 ของจำนวนผู้ประเมิน

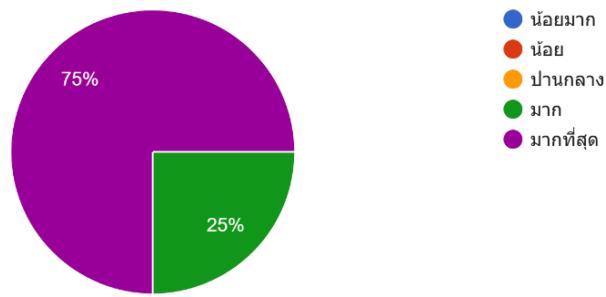
แอปพลิเคชันมีประโยชน์ต่อผู้ใช้งาน  
10 responses



รูปที่ 4.29 แผนภูมิการประเมินประสิทธิภาพในการทำงาน ข้อที่ 1

จากรูปที่ 4.29 พบร่วมกันว่า ผู้ประเมินด้านประโยชน์และการนำไปใช้ แอปพลิเคชันมีประโยชน์ต่อผู้ใช้งาน ระดับมากที่สุด 75% และระดับมาก 25% ของผู้ประเมินทั้งหมด

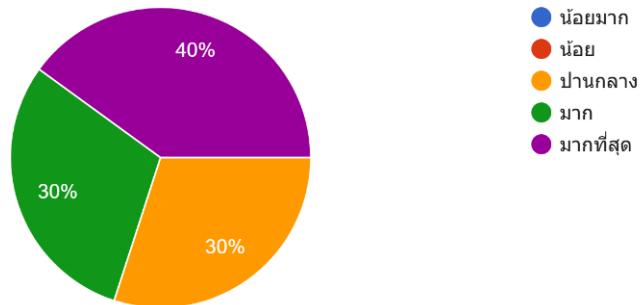
แอปพลิเคชันตอบโจทย์ความต้องการของผู้ใช้  
4 responses



รูปที่ 4.30 แผนภูมิการประเมินประสิทธิภาพในการทำงาน ข้อที่ 2

จากรูปที่ 4.30 พบร่วมกับความต้องการของผู้ใช้ ระดับมากที่สุด 75% และระดับมาก 25% ของผู้ประเมินทั้งหมด  
แอปพลิเคชันมีผลต่อการนำไปใช้ในชีวิตประจำวัน

10 responses



รูปที่ 4.31 แผนภูมิการประเมินประสิทธิภาพในการทำงาน ข้อที่ 3

จากรูปที่ 4.31 พบร่วมกับความต้องการของผู้ใช้ ระดับมากที่สุด 50% และระดับมาก 50% ของผู้ประเมินทั้งหมด  
นำไปใช้ในชีวิตประจำวัน ระดับมากที่สุด 50% และระดับมาก 50% ของผู้ประเมินทั้งหมด

## 4.6 สรุปผลการประเมินแบบตาราง

### 4.5.1 สถิติที่ใช้ในการวิเคราะห์ข้อมูล

#### 4.5.1.1 ค่าเฉลี่ย โดยใช้สมการ

$$\bar{x} = \frac{\sum x}{N} \quad (4.7)$$

โดยที่  $N$  คือ จำนวนคน

$\sum x$  คือ ผลรวมของคะแนน

$\bar{x}$  คือ ค่าเฉลี่ยของคะแนน

### 4.5.2 เกณฑ์การในคะแนน

โดยมีเกณฑ์การในคะแนนดังนี้

5 หมายถึง พึงพอใจมาก ค่าเฉลี่ย 4.51 - 5.00

4 หมายถึง พึงพอใจ ค่าเฉลี่ย 3.51 – 4.50

3 หมายถึง พึงพอใจปานกลาง ค่าเฉลี่ย 2.51 – 3.50

2 หมายถึง พึงพอใจน้อย ค่าเฉลี่ย 1.51 – 2.50

1 หมายถึง พึงพอใจควรปรับปรุง ค่าเฉลี่ย 1.00 – 1.50

การแปลความหมายของระดับความพึงพอใจ ใช้เกณฑ์การแปลความหมายจากร้อยละ ดังนี้

ต่ำกว่าร้อยละ 50 หมายถึง มีค่าความพึงพอใจน้อย

ร้อยละ 50 – 59 หมายถึง มีค่าความพึงพอใจค่อนข้างน้อย

ร้อยละ 60 – 69 หมายถึง มีค่าความพึงพอใจปานกลาง

ร้อยละ 70 – 79 หมายถึง มีค่าความพึงพอใจค่อนข้างมาก

ร้อยละ 80 – 89 หมายถึง มีค่าความพึงพอใจมาก

ร้อยละ 90 – 100 หมายถึง มีค่าความพึงพอใจมากที่สุด

ตารางที่ 4.10 การประเมินความพึงพอใจในการใช้งานระบบ

รายการ	ค่าเฉลี่ย	ร้อยละ	ระดับความพึงพอใจ
ระบบใช้งานง่าย สะดวก	4.7	94	มากที่สุด
ด้านการออกแบบ Design และการจัดรูปแบบแอปพลิเคชัน	4.5	90	มากที่สุด
การจัดรูปแบบของแอปพลิเคชันง่ายต่อการอ่านและการใช้งาน	4.3	86	มาก
ความเหมาะสมในการใช้สัญลักษณ์ในการสื่อความหมาย	4.5	90	มากที่สุด
รวม	4.5	90	มากที่สุด

จากตารางที่ 4.10 พบร่วมกันว่าผู้ประเมินมีความพึงพอใจในการใช้งานระบบอยู่ที่ 4.5 คิดเป็นร้อยละ 90 อยู่ในระดับพึงพอใจมากที่สุด

ตารางที่ 4.11 การประเมินประสิทธิภาพในการทำงาน

รายการ	ค่าเฉลี่ย	ร้อยละ	ระดับความพึงพอใจ
ระบบสามารถจับคู่ผู้ใช้ที่กดถูกใจตรงกันอย่างถูกต้อง	4.5	90	มากที่สุด
ระบบสามารถแสดงรายการการค้นหาได้เป็นลำดับอย่างถูกต้อง	4.6	92	มากที่สุด
แอปพลิเคชันสามารถทำงานได้อย่างราบรื่นและไม่เกิดปัญหาหรือความล่าช้าในขณะที่ใช้งาน	3.8	76	ค่อนข้างมาก
รวม	4.3	86	มากที่สุด

จากตารางที่ 4.11 พบร่วมกันว่าผู้ประเมินมีความพึงพอใจในประสิทธิภาพในการทำงานอยู่ที่ 4.3 คิดเป็นร้อยละ 86 อยู่ในระดับพึงพอใจมาก

**ตารางที่ 4.12 การประเมินด้านประโยชน์และการนำไปใช้**

รายการ	ค่าเฉลี่ย	ร้อยละ	ระดับความพึงพอใจ
แอปพลิเคชันมีประโยชน์ต่อผู้ใช้งาน	4.4	88	มาก
แอปพลิเคชันตอบโจทย์ความต้องการของผู้ใช้	4.3	86	มาก
แอปพลิเคชันมีผลต่อการนำไปใช้ในชีวิตประจำวัน	4.1	82	มาก
รวม	4.2	84	มาก

จากตารางที่ 4.12 พบร่วมกันว่าผู้ประเมินมีความพึงพอใจในด้านประโยชน์และการนำไปใช้อยู่ที่ 4.2 คิดเป็นร้อยละ 84 อยู่ในระดับพึงพอใจมาก

## บทที่ 5

### สรุปและข้อเสนอแนะ

ในส่วนนี้จะเป็นการสรุปผลของโครงการเว็บแอปพลิเคชันหารูปเมธด์ด้วย K-NN ที่ได้ปฏิบัติจริง จนนำไปสู่การทดลองของโครงการวิจัย และได้บันทึกผลการทดลองแล้วนำมาสรุปดังนี้

#### 5.1 สรุปตามขอบเขต

หลังจากที่คณฑ์จัดทำได้ดำเนินโครงการและทำการทดสอบการทำงานของเว็บแอปพลิเคชัน หารูปเมธด์ด้วย K-NN ได้ดำเนินการตามขอบเขตที่กำหนดไว้ได้ผลสรุปการทดลองดังนี้

5.1.1 ระบบสมัครสมาชิกและลงชื่อใช้งาน ผู้ใช้สามารถสมัครสมาชิกด้วย Email และรหัสผ่าน ได้ โดยจะต้องกรอก ชื่อผู้ใช้ อีเมล และ รหัสผ่าน ของผู้ใช้ที่ต้องการลงทะเบียน จากนั้นระบบจะส่ง รหัส OTP ไปยังอีเมลที่ผู้ใช้ได้ลงทะเบียนไว้ และนำรหัสนั้นมากรอกเพื่อยืนยันตัวตน ข้อมูลเหล่านี้จะ ถูกส่งไปยัง Server ทำการบันทึกลงในฐานข้อมูล MySQL ซึ่งอีเมลและรหัสผ่านนั้นจะสามารถใช้ใน การลงทะเบียนเข้าใช้งานระบบได้ในครั้งต่อไป โดยที่สามารถที่จะเปลี่ยนรหัสผ่านได้ในหน้าแก้ไขโปรไฟล์ใน ระบบและสามารถที่จะลบบัญชีออกจากได้หากไม่ต้องการที่จะต้องใช้งานระบบแล้ว

5.1.2 การจัดเก็บข้อมูลผู้ใช้ หลังจากที่ผู้ใช้ได้ทำการสมัครใช้งานเสร็จสิ้น ระบบจะให้ผู้ใช้กรอก ข้อมูลส่วนตัวของผู้ใช้ เช่น อายุ เพศ รูปภาพ การศึกษา หอพัก ยานพาหนะ และข้อมูลการติดต่อ เมื่อ กรอกข้อมูลครบถ้วนแล้ว จากนั้นจะให้ผู้ใช้กรอกข้อมูลลักษณะนิสัยของผู้ใช้ เช่น ช่วงเวลาอน ความ สะอาด การใช้เสียง การสูบบุหรี่ การใช้งานเครื่องปรับอากาศ และการเที่ยวกลางคืน เป็นต้น โดย ข้อมูลเหล่านี้จะการบันทึกลงในฐานข้อมูล MySQL เพื่อนำไปวิเคราะห์และแสดงผลต่อไป โดยที่ผู้ใช้ สามารถแก้ไขข้อมูลเหล่านี้ในโปรไฟล์ของผู้ใช้ได้

5.1.3 ระบบประเมินความเข้ากันได้ระหว่างผู้ใช้ จะนำข้อมูลลักษณะนิสัยของผู้ใช้มา เปรียบเทียบกัน โดยประยุกต์ใช้หลักการ K-NN เพื่อการคำนวณความคล้ายคลึงระหว่างผู้ใช้งาน โดยมุ่งเน้นการใช้ระยะห่างแบบยูคลิดีเดียน (Euclidean Distance) แทนการจำแนกประเภท (Classification) ได้ผลลัพธ์เป็นค่าระยะห่าง เพื่อนำไปคำนวณร้อยละความใกล้เคียงของแต่ละ ผู้ใช้งานอื่นเทียบกับผู้ใช้งานขณะนั้น และแสดงผลผู้ใช้ที่เข้ากันได้ตามลำดับจากมากไปน้อย แก่ผู้ใช้งานในระบบ

5.1.4 รายละเอียดโปรไฟล์ผู้ใช้ เป็นหน้าที่แสดงข้อมูลของผู้ใช้ โดยดึงข้อมูลจากฐานข้อมูลที่ ประกอบไปด้วย รูปโปรไฟล์ ข้อมูลส่วนตัว ข้อมูลการติดต่อและลักษณะนิสัยของผู้ใช้ ในส่วนของ ลักษณะนิสัยของผู้ใช้ ระบบจะแสดงข้อมูลในส่วนที่ตรงกันให้เด่นชัดขึ้นด้วยการไฮไลท์ด้วยสี เพื่อให้ ผู้ใช้สามารถสังเกตข้อมูลที่ตรงกันได้ง่ายขึ้น และข้อมูลการติดต่อจะแสดงต่อเมื่อได้จับคู่สำเร็จ

5.1.5 ระบบกรองการค้นหาข้อมูล สามารถกรองการค้นหาด้วยลักษณะนิสัยโดยที่ผู้ใช้สามารถเลือกที่จะเลือกลักษณะนิสัยที่ชอบได้ตามต้องการ ระบบจะแสดงผู้ใช้ที่มีข้อมูลที่ตรงกับที่ผู้ใช้เลือกแต่การกรอกการค้นหาด้วยตำแหน่งของผู้ใช้ยังไม่สามารถทำได้ ทำได้เพียงการเลือกตำแหน่งจังหวัดและสถานศึกษาเท่านั้น

5.1.6 ระบบการจับคู่ ผู้ใช้สามารถที่จะกดถูกใจผู้ใช้รายอื่นได้โดยกดปุ่มถูกใจในหน้าprofile ของผู้ใช้นั้น ๆ เมื่อกดถูกใจแล้วระบบจะทำการบันทึกการกดถูกใจของผู้ใช้ลงในฐานข้อมูล และนำผู้ใช้ที่ถูกกดถูกใจไปแสดงในหน้ากดถูกใจ เพื่อให้ผู้ใช้ได้ดูหรือพิจารณาอีกครั้ง โดยในหน้าprofile ของผู้ใช้ที่กดถูกใจไปแล้วจะแสดงปุ่มถอนถูกใจ ที่สามารถยกเลิกการกดถูกใจได้ เมื่อทั้งสองฝ่ายกดถูกใจซึ่งกันและกันทั้งคู่จะทำให้การจับคู่สำเร็จ ระบบจะแสดงผู้ใช้ที่ได้ทำการกดถูกใจคุณกลับในหน้าจับคู่ และจะแสดงข้อมูลการติดต่อของผู้ใช้นั้น เช่น โซเชียลมีเดีย และเบอร์โทรศัพท์มือถือ

5.1.7 ระบบแจ้งเตือนการจับคู่สำเร็จ จะส่งแจ้งเตือนผ่าน Email ของผู้ใช้ว่าได้ทำการจับคู่สำเร็จ โดยจะมีปุ่ม “ดูข้อมูลการจับคู่” จะนำผู้ใช้ไปยังหน้าจอจับคู่สำเร็จในระบบ

5.1.8 ระบบแจ้งปัญหา ในระบบจะมีการแจ้งปัญหาอยู่ 3 ส่วน ได้แก่

5.1.8.1 สามารถแจ้งปัญหาผู้ใช้ที่ไม่พึงประสงค์ได้ การแจ้งปัญหาเกี่ยวกับผู้ใช้ โดยให้ผู้ใช้ได้ทำการเลือกประเภทการรายงาน เขียนคำอธิบายรายงานและแนบรูปภาพหลักฐาน ให้แก่ผู้ดูแลระบบ เพื่อตรวจสอบและแก้ไขตามความเหมาะสม

5.1.8.2 สามารถแจ้งปัญหาเกี่ยวกับระบบการใช้งาน โดยให้ผู้ใช้ได้ทำการเลือกประเภทการรายงาน และเขียนคำอธิบายรายงานให้ระเอียดและชัดเจน และแนบรูปภาพหลักฐาน ให้แก่ผู้ดูแลระบบ เพื่อตรวจสอบและแก้ไขตามความเหมาะสม

5.1.8.3 สามารถให้ข้อเสนอแนะเกี่ยวกับระบบ โดยให้ผู้ใช้ได้ทำการเขียนคำอธิบายข้อเสนอแนะเพื่อส่งไปยังผู้ดูแลระบบ เพื่อตรวจสอบและแก้ไขตามความเหมาะสม

5.1.9 ระบบให้คะแนนความพึงพอใจ โดยผู้ใช้งานสามารถให้คะแนนได้ในหน้าจับคู่สำเร็จ โดยมีระดับการให้คะแนน 1 ถึง 5 ดาว

5.1.10 ระบบผู้ดูแลระบบ ผู้ดูแลระบบสามารถตรวจสอบจำนวนผู้ใช้งานในระบบทั้งหมด จำนวนการจับคู่ทั้งหมด และจำนวนการรายงานจากผู้ใช้ โดยผู้ดูแลระบบจะสามารถดูรายละเอียดการรายงานเกี่ยวกับการรายงานผู้ใช้ การรายงานระบบ ข้อเสนอแนะ ดูการให้คะแนน สามารถปรับสถานะบัญชีผู้ใช้ รวมไปถึงสามารถลบบัญชีผู้ใช้ได้

5.1.11 การประเมินผล

5.1.11.1 ในเชิงปริมาณ คะแนนความพึงพอใจระบบการแนะนำรูปเมทเนลี่มีมากกว่า 70% คะแนนความพึงพอใจการแสดงผลข้อมูลบุคคลเฉลี่ยมีมากกว่า 70% และเว็บแอพพลิเคชั่นทำงานได้ตามขอบเขตมากกว่า 80%

5.1.11.2 ในเชิงคุณภาพ ด้านความถูกต้องและเหมาะสมของเนื้อหาบนเว็บไซต์ โดยทดสอบใช้งานสร้างบัญชีตลอดจนถึงขั้นตอนการจับคู่อย่างน้อย 10 ครั้ง พบร่วมระบบทำงานถูกต้องไม่น้อยกว่า 80%

## 5.2 ปัญหาและอุปสรรค

5.2.1 การเรียนรู้ Stack ในการทำโครงงาน เนื่องจากผู้จัดทำมีความเข้าใจไม่มากพอ ทำให้ใช้เวลาในการเรียนรู้และพัฒนาสูง รวมถึง Source Code ของระบบที่อาจไม่มีประสิทธิภาพเท่าที่ควร แก้ปัญหาโดยเรียนรู้ Stack ที่ต้องการใช้งานผ่านคอร์สสอนออนไลน์ เช่น Udemy และ YouTube

5.2.2 การ Deploy บน Railway ไม่สามารถ Storage ข้อมูลไฟล์โดยตรงได้ ทำให้ไม่สามารถเก็บรูปภาพไปรีไฟล์ของผู้ใช้งานได้

แก้ปัญหาโดยการใช้ Cloudinary ในการเก็บไฟล์ โดยสามารถใช้ร่วมกับ ExpressAPI ได้ โดยมีพื้นที่การใช้งานฟรี 25GB ต่อเดือน มี CDN กระจายหลายพื้นที่ ทำให้โหลดภาพได้เร็ว

5.2.3 เนื่องจาก Library ml-knn ที่ใช้สำหรับใช้งาน K-NN บน JavaScript คืนค่ามาเพียงค่าที่ Predict แล้วเท่านั้น ไม่มีค่าระยะห่างของข้อมูล ทำให้ไม่สามารถคำนวณระยะห่างมาคำนวณได้

แก้ปัญหาโดยการคำนวณ Euclidean แยกด้วยตนเองตามสมการของ Euclidean

## 5.3 ข้อเสนอแนะ

5.3.1 สามารถปรับอัลกอริทึมหรือเลือกใช้อัลกอริทึมอื่นเพื่อทดสอบความแม่นยำของการแนะนำ ที่อาจจะสูงกว่า

5.3.2 การนำไปใช้งานจริงควรตรวจสอบช่องโหว่ของเว็บแอปพลิเคชัน เนื่องจากเป็นข้อมูลส่วนตัวผู้ใช้ที่ละเอียดอ่อน

5.3.3 โครงงานนี้อาจไม่ใช่รูปแบบการวางแผนโครงสร้างและการเขียนโค้ดที่ดีที่สุด (Best Practice) ซึ่งอาจส่งผลต่อประสิทธิภาพของระบบและเกิดค่าใช้จ่ายที่สูงเกินความจำเป็นเมื่อ Deploy ขึ้นเป็น Production บน Cloud หรืออาจไม่เหมาะสมกับการ Scaling หากมีผู้ใช้งานสูงมาก

5.3.4 ในอนาคตสามารถเพิ่มการปรับแต่งอัลกอริทึมแบบ Realtime ผ่านหน้าแดชบอร์ด เพื่อให้แดชบอร์ดสามารถปรับแต่งอัลกอริทึมได้โดยง่าย เพื่อหาจุดเดียวที่ดีและเร็วขึ้น

5.3.5 ความแตกต่างระหว่างบุคคล แม้ว่าค่าถ่วงน้ำหนักจะถูกกำหนดขึ้นจากการวิเคราะห์ผลกระทบโดยทั่วไป แต่ความสำคัญของแต่ละปัจจัยอาจแตกต่างกันไปตามบุคคล ระบบควรมีความยืดหยุ่นเพื่อรับความแตกต่างเหล่านี้ และควรหาช่วงค่าที่เหมาะสมเพื่อไม่ให้น้ำหนักส่งผลต่อระยะห่างด้านใดด้านหนึ่งมากเกินไป

5.3.6 ถึงแม้การคำนวณระยะห่างโดยใช้ Euclidean จะเห็นผลลัพธ์โดยรวมของผู้ใช้ อาจเพิ่มการคำนวณในรูปแบบอื่นร่วมด้วย เพื่อให้เห็นถึงค่าที่มองข้ามบางอย่าง เช่น ผู้ใช้นั้นอาจจะระยะห่างหนักในด้านใดด้านหนึ่ง

5.3.7 การแปลงค่าแบบ Ordinal Number จะมีผลกระทบเมื่อจำนวนตัวเลือกในแต่ละข้อไม่เท่ากัน อาจจะต้องวิเคราะห์การทำ Normalization และเปรียบเทียบว่าค่าใดเหมาะสมกว่ากัน

#### 5.4 สรุปผลการทดลอง

โครงการนี้มีวัตถุประสงค์เพื่อสร้างเว็บแอปพลิเคชันหารูปเมทด้วย K-NN เพื่อสร้างและพัฒนาระบบเฉพาะทางสำหรับการค้นหารูปเมทที่เหมาะสม ลดความยุ่งยากในการค้นหารูปเมท เพื่อให้เกิดความสะดวกรวดเร็วในการค้นหารูปเมท และง่ายต่อการตัดสินใจ โดยทั้งนี้ระบบพัฒนาโดยอาศัยองค์ความรู้ในการสร้างแอปพลิเคชัน การออกแบบ และการจัดการฐานข้อมูล การออกแบบ UX/UI โดยใช้ React ในการพัฒนา ใช้ NodeJS เซิร์ฟต์อกับฐานข้อมูลผ่าน API และใช้ MySQL ในการเก็บข้อมูลในฐานข้อมูล

ผลการทำงานของเว็บแอปพลิเคชันหารูปเมทด้วย K-NN เป็นที่น่าพึงพอใจ โดยระบบมีหน้าตาที่น่าใช้งาน สามารถแสดงรายละเอียดได้ถูกต้อง และยังสามารถช่วยให้ผู้ใช้ตัดสินใจเลือกหรือเข้าถึงรูปเมทที่เหมาะสมได้ ระบบสามารถบันทึกการทำงานลงฐานข้อมูล และตึงข้อมูลจากฐานข้อมูลมาใช้ได้ไม่ผิดพลาด แต่ยังมีส่วนที่ต้องปรับปรุงที่กล่าวไว้ในข้อเสนอแนะ

## บรรณานุกรม

- [1] กิตติมศักดิ์ ในจิต. (2566). K-Nearest Neighbors (KNN) คืออะไร. URL: <https://kittim-asak.com/k-nearest-neighbors-knn-machine-learning-python>. เมื่อ 25 มกราคม 2568
- [2] จิตราภรณ์ ราрапิทักษ์วงศ์. (2560). การใช้ K-Nearest Neighbor Algorithm เพื่อสร้างโมเดลการจำแนกรูปแบบการเรียนรู้สำหรับทำนายผลการจัดระดับมาตรฐานผลิตภัณฑ์ OTOP หัตถกรรม กลุ่มไม้. มหาวิทยาลัยราชภัฏเชียงใหม่. URL: <http://cmruir.cmru.ac.th/handle/123456789/427?locale=th>. สืบค้นเมื่อ 15 พฤษภาคม 2567
- [3] ไตรวิทย์ ชุปคำ. (2566). MySQL คืออะไร. URL: <https://blog.openlandscape.cloud/mysql>. สืบค้นเมื่อ 7 มกราคม 2568
- [4] รดา จันตะคุณ. (2559). การพยากรณ์ความเป็นไปได้ในการเลือกสมัครสาขาวิชาโดยใช้เทคนิคเหมืองข้อมูล. สาขาวิชาระบบสารสนเทศเพื่อการจัดการ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏมหาสารคาม. URL: [https://fulltext.rmu.ac.th/fulltext/2559/M1206\\_83/Jantakoon%20Thada.pdf](https://fulltext.rmu.ac.th/fulltext/2559/M1206_83/Jantakoon%20Thada.pdf). สืบค้นเมื่อ 15 พฤษภาคม 2567
- [5] รวช รวมทรัพย์, และสุรศักดิ์ มังสิงห์. (2559). การจำแนกชนิดของพืชด้วยวิธีเพื่อนบ้านไก่สุดร่วมกับการเลือกตัวแทนที่เหมาะสมด้วยขั้นตอนวิธีเชิงพันธุกรรมโดยใช้คุณลักษณะรูปทรงและพื้นผิวของใบพืช. สาขาวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยศิลปากร. URL: <https://ph01.tci-thaijo.org/index.php/VESTSU/article/view/75457>. สืบค้นเมื่อ 15 พฤษภาคม 2567
- [6] พรรณธิภา เพชรบุญมี. (2559). การวิเคราะห์ความสามารถทางการเรียนรู้ตามแนวทางพหุปัญญาโดยใช้เทคนิคต้นไม้ตัดสินใจ. คณะบริหารธุรกิจและศิลปศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนาตาก. URL: [https://kukr.lib.ku.ac.th/db/drought/search\\_detail/download\\_digital\\_file/197075/126569](https://kukr.lib.ku.ac.th/db/drought/search_detail/download_digital_file/197075/126569). สืบค้นเมื่อ 15 พฤษภาคม 2567
- [7] สุคนธ์พิพิธ คำจันทร์, และประภาพร กุลลีมรัตน์ชัย. (2565). การประยุกต์ใช้ User Interface (UI) และ User Experience (UX) ในการออกแบบแพลตฟอร์ม. คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยอีสเทิร์นเอเชีย. URL: <https://he01.tci-thaijo.org/index.php/EAUHJSci/article/view/255087>. สืบค้นเมื่อ 10 ธันวาคม 2567
- [8] 1stCraft Team. (2563). Web Application คืออะไร ต่างจากเว็บไซต์ทั่วไปอย่างไร. URL: <https://1stcraft.com/Website-application-vs-general-website/>. สืบค้นเมื่อ 10 ธันวาคม 2567

## บรรณานุกรม (ต่อ)

- [9] Aastha Bhatra, Mariyah Zubair Khan, Ashita Gupta, and Amanat Bansal. (2023). **Augmenting Roommate Relationships through personality-based intervention Strategies.** MA Psychology student at University of Delhi. URL: <https://advance.sagepub.com/users/720170/articles/705080>. สืบค้นเมื่อ 10 ธันวาคม 2567
- [10] Advanced IServices. (2564). **ข้อดีและข้อเสียของ Web Application คืออะไร.** URL: <https://www.advancedis.co.th/en/blog>. สืบค้นเมื่อ 15 พฤษภาคม 2567
- [11] Asim Ahmed. (2023). **What is Vercel and Why You Should Use It?.** URL: <https://www.getfishtank.com/insights/what-is-vercel>. เมื่อ 25 มกราคม 2568
- [12] Cloudinary. (2025). **What is Cloudinary?.** URL: <https://www.getapp.com/it-management-software/a/cloudinary/>. เมื่อ 25 มกราคม 2568
- [13] Datacamp. (2024). **Understanding Euclidean Distance.** URL: <https://www.datacamp.com/tutorial/euclidean-distance>. เมื่อ 25 มกราคม 2568
- [14] Devopenworld. (2568). **Material UI คืออะไร.** URL: <https://devopenworld.com>. สืบค้นเมื่อ 15 พฤษภาคม 2567
- [15] GeeksForGeeks. (2025). **How to choose the right distance metric in KNN?.** URL: <https://www.geeksforgeeks.org/how-to-choose-the-right-distance-metric-in-knn/>. เมื่อ 25 มกราคม 2568
- [16] Ichwanul Muslim Karo Karo, Athallah Tsany Rakha Dzaky, and Muhammad Alfhi Saputra. (2021). **Comparative Analysis of K-Nearest Neighbor and Modified K-Nearest Neighbor Algorithm for Financial Well-Being Data Classification.** School of Computing Telkom University. URL: <https://socjs.telkomuniversity.ac.id/ojs/index.php/indojc/article/view/593>. สืบค้นเมื่อ 10 ธันวาคม 2567
- [17] Mairéad A. Willis, and Sean P. Lane. (2022). **Preliminary evidence for the factor structure, concurrent validity, and construct validity of the Roommate Relationship Scale in a college sample.** Frontiers in Psychology. URL: <https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.960421/full>. สืบค้นเมื่อ 10 ธันวาคม 2567

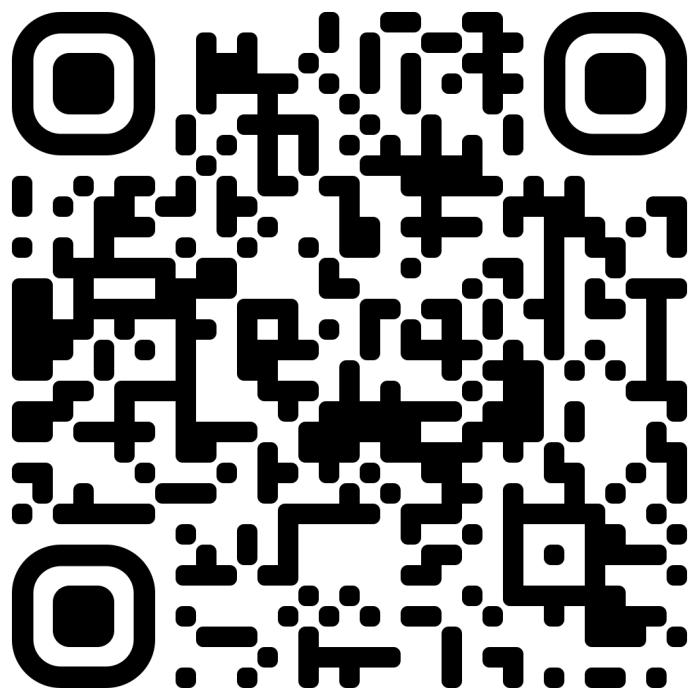
### บรรณานุกรม (ต่อ)

- [18] Petch.natawat. (2567). **NodeJS คืออะไร**. URL: <https://codinggun.com/javascript/node-js/>. สืบค้นเมื่อ 7 มกราคม 2568
- [19] Rifkie Primartha, and Adi Wijaya. (2019). **Comparison of Distance Measurement Methods on K-Nearest Neighbor Algorithm for Classification**. Sriwijaya International Conference on Information Technology and Its Applications. URL: <https://www.researchgate.net/publication/341393947>. สืบค้นเมื่อ 10 ธันวาคม 2567
- [20] Serge Angéloz. (2024). **What is Railway App?**. URL: [https://medium.com/@sangel\\_oz69/what-is-railway-app-5b1bd17aa8f0](https://medium.com/@sangel_oz69/what-is-railway-app-5b1bd17aa8f0). เมื่อ 25 มกราคม 2568
- [21] Sharukh RAHMAN, and Manoj Kumar D S. (2021). **Optimal Room and Roommate Matching System Using Nearest Neighbors Algorithm with Cosine Similarity Distribution**. Crescent Institute of Science and Technology Department of Computer Science and Engineering. URL: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3869826](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3869826). สืบค้นเมื่อ 15 พฤษภาคม 2567
- [22] TechUp Team. (2568). **React คืออะไร**. URL: <https://www.techupth.com/articles/react>. สืบค้นเมื่อ 7 มกราคม 2568
- [23] Traivith Chupkum. (2566). **ทำ API ด้วย Express**. URL: <https://medium.com/@traivith.chupkum>. สืบค้นเมื่อ 7 มกราคม 2568

## ភាគធនវក ៩

ແຫລ່ງខ້ុមតាមដើរគីឡូក្រកណ្ឌ

## 1. Source Code



URL: <https://github.com/jschomchoey/findmate-react.git>

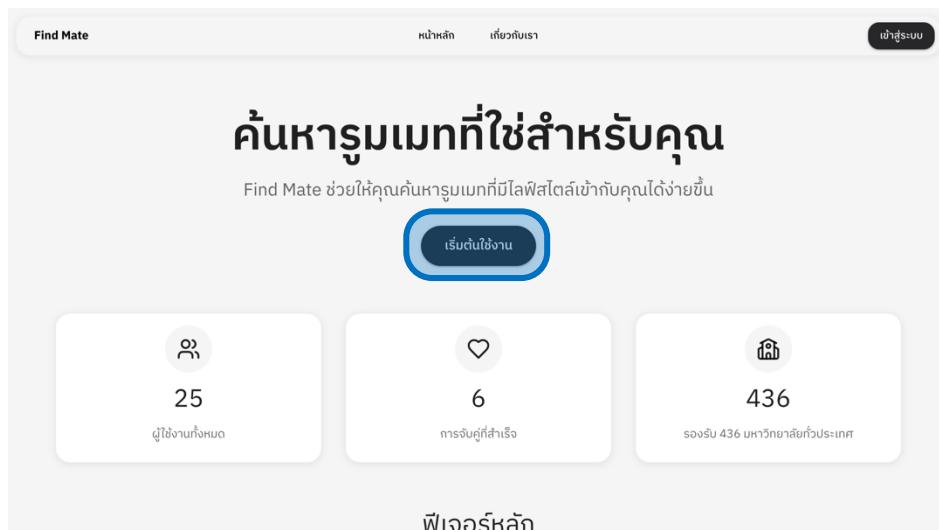
ภาคผนวก ข

คู่มือการใช้งาน

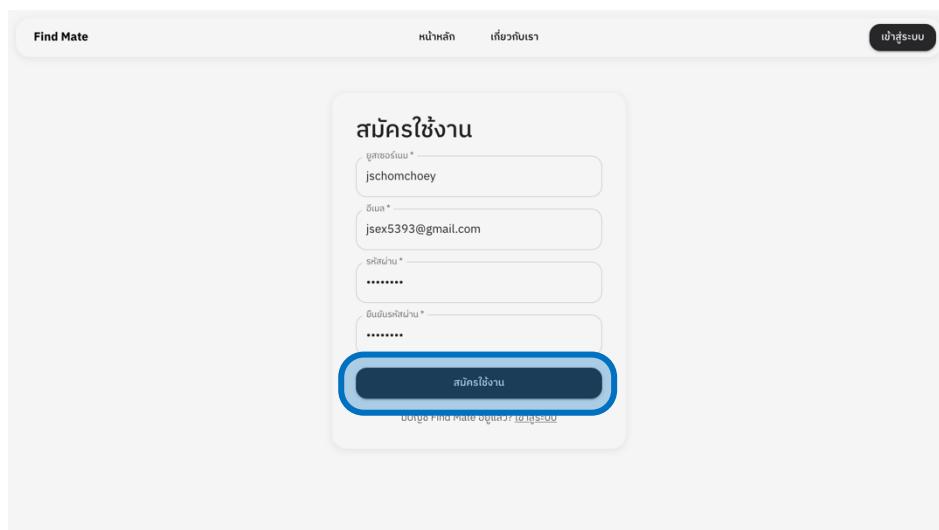
## คู่มือการใช้งาน

### 1. การสมัครสมาชิกและการเข้าสู่ระบบ

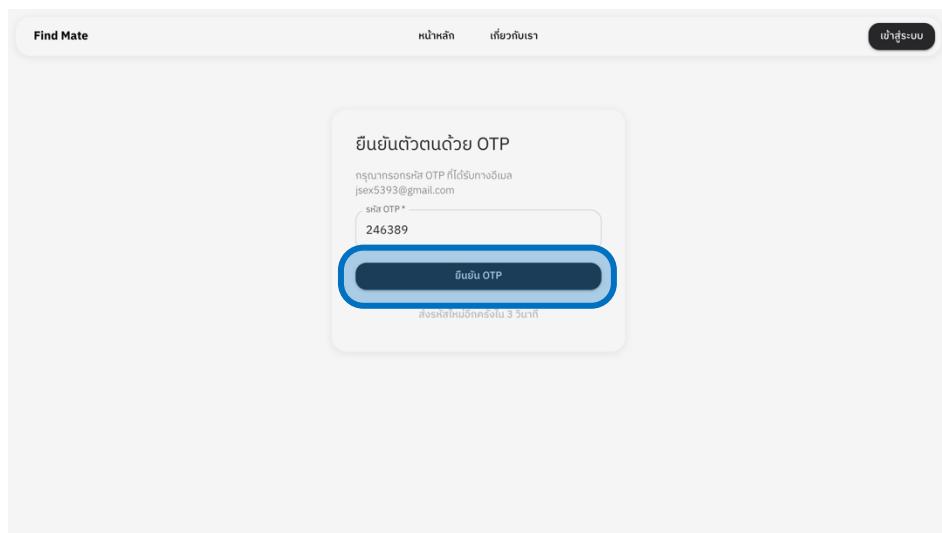
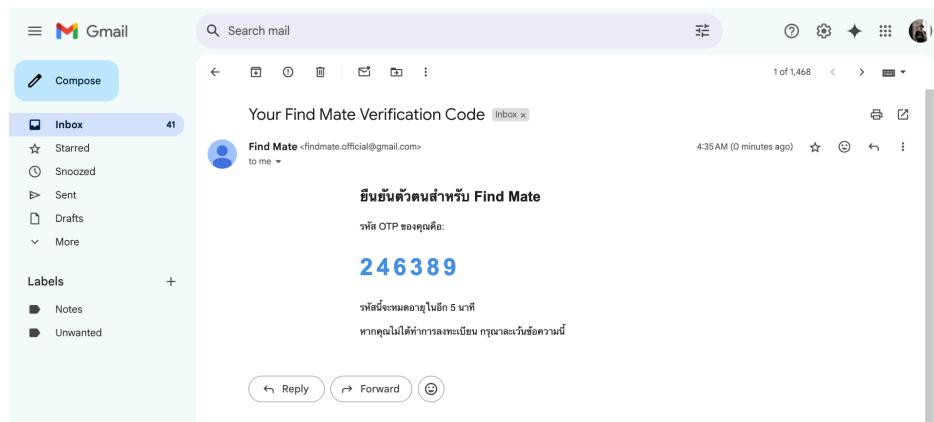
การทำงานของเว็บแอปพลิเคชัน Find Mate เมื่อเริ่มการทำงานจะเข้ามาที่หน้าจอเริ่มต้น การใช้งานจะพบกับหน้าที่จะแสดงข้อมูลแนะนำสำหรับผู้ใช้ใหม่ก่อนที่จะเริ่มเข้าสู่ระบบ



ผู้ใช้งานที่ยังไม่เคยสมัครสมาชิก หรือยังไม่มีบัญชีผู้ใช้ ต้องทำการสมัครสมาชิกโดยกดปุ่ม “เริ่มต้นใช้งาน” จะนำผู้ใช้ไปสู่หน้าจอสมัครใช้งาน



ผู้ใช้งานที่ยังไม่เคยสมัครสมาชิก หรือยังไม่มีบัญชีผู้ใช้ ต้องทำการสมัครสมาชิกโดยกดปุ่ม “เริ่มต้นใช้งาน” จะนำผู้ใช้ไปสู่หน้าจอสมัครใช้งาน



หลังจาก กดปุ่ม “สมัครใช้งาน” ระบบจะส่งรหัส OTP ไปยัง Email ที่สมัครใช้งาน จากนั้นให้ผู้ใช้กรอกรหัส OTP ที่ได้รับ และกดปุ่ม “ยืนยัน OTP” ก่อนเข้าสู่ระบบ

Find Mate

ເນົາເລັກ ເຖິງວັນເຮດ

jsex5393

ປະວັດສ່ວນຕົວ

ເປີ້ນຮູບພາບ

ເລືອດອັນດີ \*  
ຊື່ຮະກິກ

ນາມສັກ \*  
ໜິນເຊຍ

ເລືອດອັນ \*  
ເຈນສີ

ວັນ  
23

ສາມາດ \*  
ໄສດ

ນອດເຕີຣີໂຊດ

ນີ້ແຫ່ງວານເພີ້ນເປັນ ຂອບມາດວັດທະນີທີ່ສຳເນົາຂອງການ  
ຫາຮູນເມັກໃນສໍ້ອັນດີນັກຄົນ ພວດເປັນຄະແນນນາກ ສະດຸງເຕັນບ່າຍ

ໂທເຂົ້າລົມເດີຍແລະບ້ວນຸລືດົດຕ່ວ

ກຽນນາກໂກນໍບູລເດືອດເອນບ່ານເນັ້ນ 1 ມັງກາງ  
Facebook  
Teerapat Chomchoey

Instagram  
jschomchoey

Line ID  
jschomchoey

ນາມບ່ອນໂກສີພໍາ  
0616980000

ບັນທຶກ

ໃຫ້ຜູ້ໃຊ້ເລືອກຮູບໂປຣໄຟລ໌ເພື່ອເພີ່ມຮູບໂປຣໄຟລ໌ຂອງຕົນເອງ ອາກໄມ່ເພີ່ມຮູບຈະເປັນຮູບພື້ນຮູ້ານ  
ແລກຮອກຂໍ້ອນນູລສ່ວນຕົວໃຫ້ຮັບຄ້ວນຈາກນັ້ນກົດປຸ່ມ “ບັນທຶກ” ເພື່ອໄປໜ້າຕອນຄັດໄປ

ແມ່ວນດັບ

ຄຸນສະດວກໃຈກໍບຽນເພັກທີ່ມີຄວາມເຂົ້າການຄາສານແຕກຕ່າງຈາກຄຸນເຊີ້ວໄປ? \*

ເສດວກ

ໄດ້ລັ້າໄໝຕ່ອງຮັບເປັນ

ໄສສະດວກ

ຄຸນຕ່ອງການຮູນເພັກທີ່ວ່າງຈົນກັນຮະຍະຍາວເຊີ້ວໄປ? \*

ຕ່ອງການ

ບໍລິສັດ

ບັນດັບ

ขั้นตอนดังต่อไปนี้จะให้ผู้ใช้เลือกตอบคำถามลักษณะนิสัยส่วนตัว หรือกิจวัตรประจำวันในการอาศัยในหอพัก โดยคำถามแต่ละข้อจะเลือกตอบได้เพียงข้อละ 1 คำตอบเท่านั้น ข้อมูลส่วนนี้จะนำไปเปรียบเทียบกับผู้ใช้รายอื่น ๆ เพื่อแสดงความเข้ากันได้ระหว่างผู้ใช้ จากนั้นกดปุ่ม “บันทึก” เพื่อเข้าสู่หน้าหลักของแอปพลิเคชัน

Find Mate

ເນັ້ນດັບ

ເຖິງກົບເຮົາ

ເຂົ້າສູ່ຮະບບ

ອີເມວ \*

jssex5393@gmail.com

ລະຫັດຜົນ \*

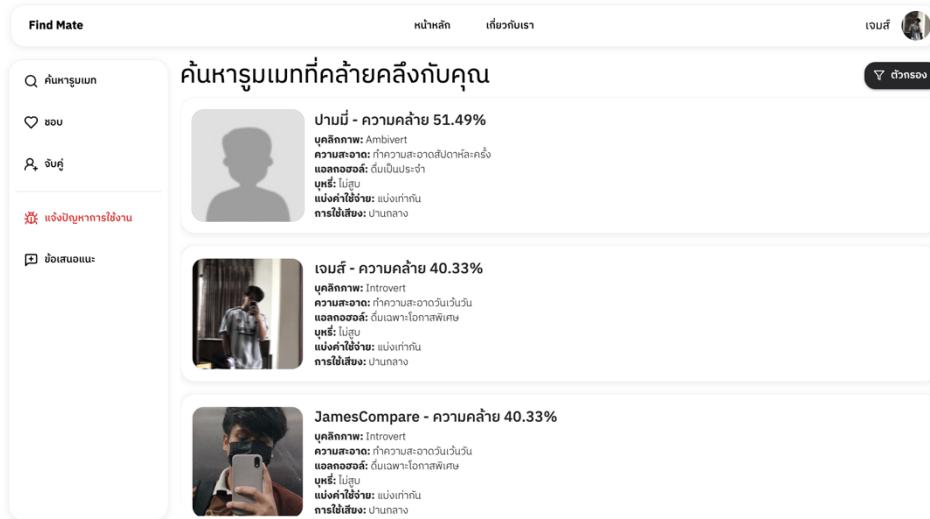
\*\*\*\*\*

ເຂົ້າສູ່ຮະບບ

หากผู้ใช้ที่เคยสมัครสมาชิก หรือมีบัญชีผู้ใช้อยู่แล้วสามารถข้ามขั้นตอนการสมัคร โดยให้กดที่ปุ่ม “ເຂົ້າສູ່ຮະບບ” เพื่อกรอกข้อมูลสำหรับเข้าสู่ระบบต่อไป

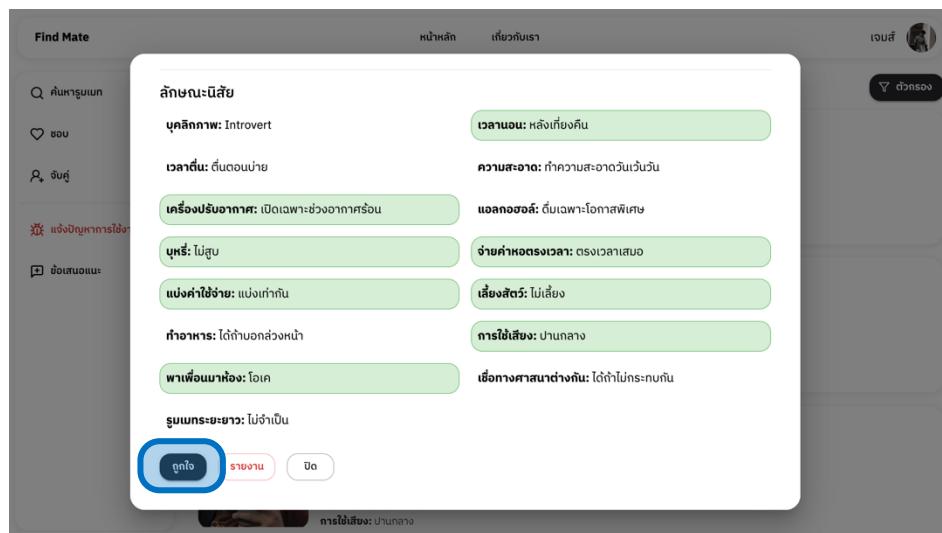
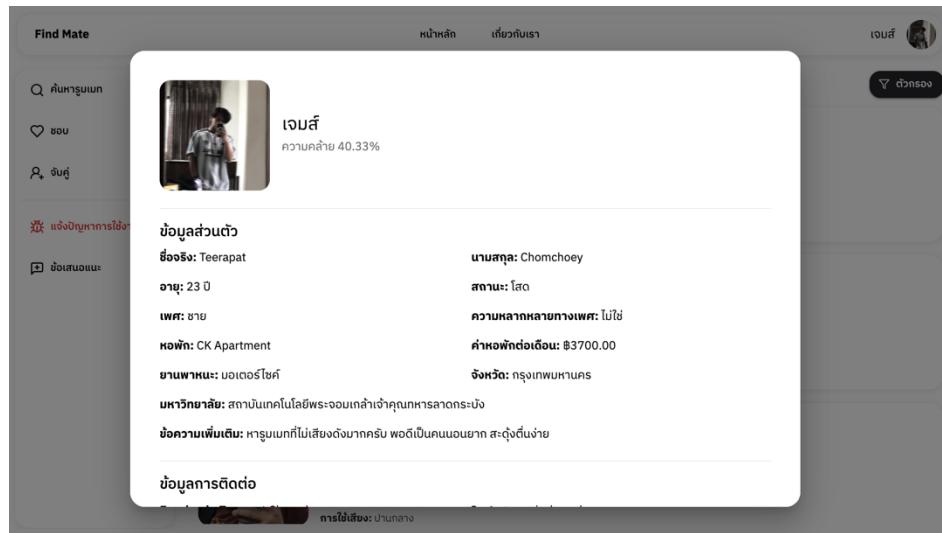
## 2. การค้นหารูปเมท

### 2.1 หน้าค้นหารูปเมทของเว็บแอปพลิเคชัน Find Mate



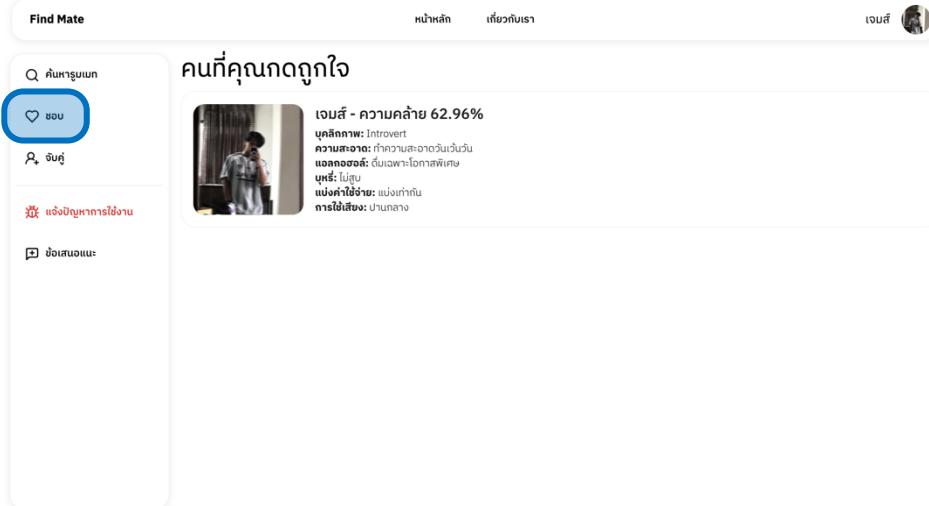
หลังจากเข้าสู่ระบบเรียบร้อย ก็จะมาสู่หน้าจอหลักของแอปพลิเคชัน Find Mate ที่จะแสดงลำดับรูปเมทที่มีความคล้ายคลึงกับคุณ โดยจะแสดงรูปเมทที่มีความคล้ายคลึงมากไปน้อย และแสดงข้อมูลลักษณะนิสัยบางส่วน เพื่อให้ง่าย และสะดวกต่อการตัดสินใจของผู้ใช้ในการเข้าเยี่ยมชมໂປร์เพล์ของรูปเมทนั้น ๆ

ผู้ใช้สามารถกรองการค้นหาได้โดยกดที่ปุ่ม “ตัวกรอง” จะแสดงหน้าต่างให้ผู้ใช้เลือกกรองข้อมูลตามจังหวัด สถานศึกษา หรือลักษณะนิสัยส่วนตัว โดยสามารถเลือกได้หลายตัวเลือกตามที่ผู้ใช้ต้องการ หรือสนใจ จากนั้นกดปุ่ม “นำไปใช่” ระบบจะแสดงรายการรูปเมท ตามตัวกรองการค้นหา หากไม่ถูกใจสามารถที่จะกดปุ่ม “ล้างตัวกรอง” เพื่อที่จะนำการกรองข้อมูลออก

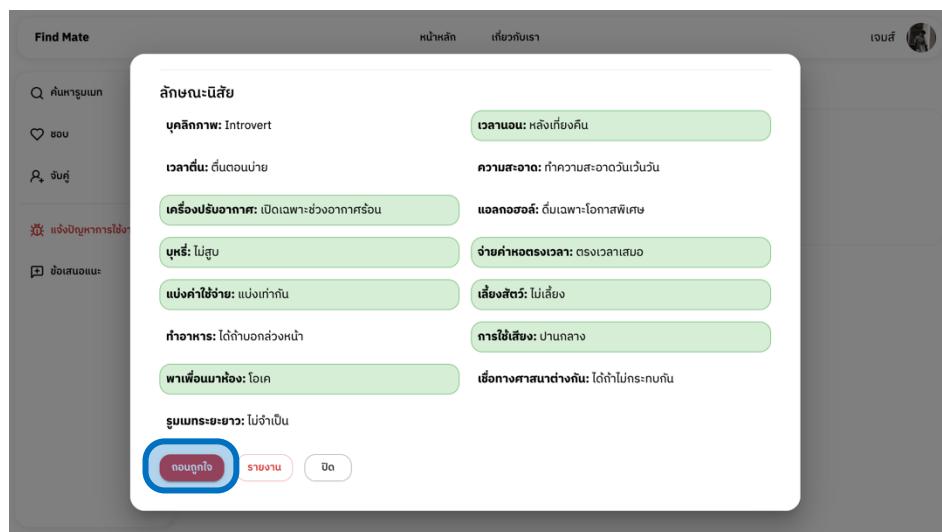


เมื่อคุณกดไปที่โปรไฟล์รูปเมทที่สนใจ ก็จะแสดงหน้าต่างโปรไฟล์รูปเมทคนนั้นออกมานั่นเอง ซึ่งจะแสดงข้อมูลส่วนตัวต่าง ๆ ที่นำข้อมูลจากการสมัครสมาชิกครั้งแรกมาแสดง ยกเว้น เบอร์โทรศัพท์มือถือ ที่จะแสดงเมื่อจับคู่สำเร็จ ยกทั้งยังแสดงลักษณะนิสัยที่ตรงกันกับคุณจากการเน้นข้อความเป็นแบบสีเขียว เพื่อให้ง่ายต่อการประกอบการตัดสินใจ หากสนใจ หรือถูกใจรูปเมทคนนั้นก็สามารถที่จะกดปุ่ม “ถูกใจ” ในโปรไฟล์ของรูปเมทคนนั้นได้

## 2.2 หน้าถูกใจ



หน้าคนกดถูกใจ จะแสดงรูปเมทที่คุณได้ทำการ กดถูกใจ ไว้ เพื่อให้ง่ายต่อการหารูปเมทที่คุณได้เคยถูกใจไว้ เพื่อตรวจเช็คโปรไฟล์รูปเมทนั้น ๆ

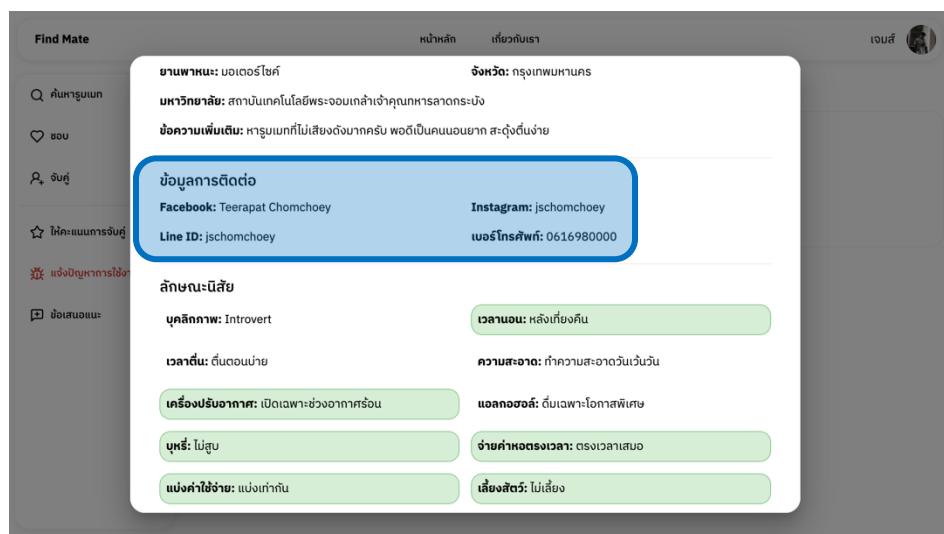


หากต้องการที่จะยกเลิกการกดถูกใจ ให้ผู้ใช้เข้าไปที่โปรไฟล์รูปเมทคนนั้น จากนั้นเลื่อนลงมาข้างล่างสุดจะมีปุ่ม “ถอนถูกใจ” เมื่อกดแล้วโปรไฟล์รูปเมทรายนั้นจะออกจากหน้าคนกดถูกใจ ทันที

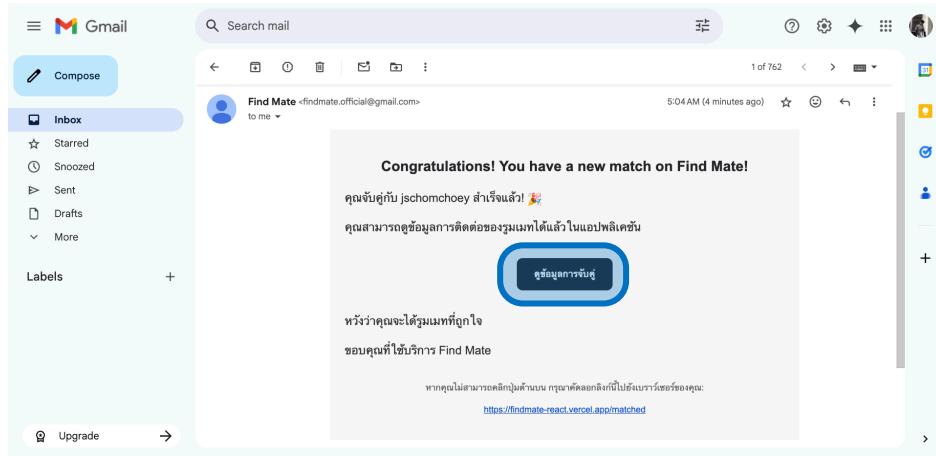
### 2.3 หน้าจับคู่



หน้าจับคู่ จะแสดงรูปเมทที่ได้จับคู่สำเร็จ เมื่อรูปเมทคนนั้นได้ทำการกด “ถูกใจ” คุณกลับหรือเมื่อทั้งสองฝ่ายได้ทำการกดถูกใจซึ่งกันและกันก็จะแสดงໂປຣີຣູມເມທ໌ເປັນຄຸກ່າບເຮົາ ในหน้าจับคู่



เมื่อทำการจับคู่สำเร็จระบบจะแสดงข้อมูลการติดต่อของรูปเมทรายนັ້ນ เพื่อได้ทำการติดตอกันได้ง่าย และสะดวกยิ่งขึ้น



ระบบจะแจ้งเตือนการจับคู่ โดยส่งผ่าน Email ของผู้ใช้ว่าได้ทำการจับคู่สำเร็จ โดยจะมีปุ่ม “ดูข้อมูลการจับคู่” จะนำผู้ใช้กลับไปยังหน้าจอจับคู่สำเร็จในระบบ

### 3. การแก้ไขโปรไฟล์



หากผู้ใช้ต้องการที่จะแก้ไขข้อมูลในโปรไฟล์ สามารถทำได้โดยการกดไปที่ “รูปโปรไฟล์” มุมบนขวาของหน้าจอ และกดปุ่ม “แก้ไขโปรไฟล์” เพื่อมายังหน้าจอแก้ไขโปรไฟล์ โดยในหน้านี้จะแบ่งการแก้ไขข้อมูลออกเป็น 2 ส่วน ได้แก่ แก้ไขประวัติส่วนตัว และ แก้ไขลักษณะนิสัย

บุคลากรเพื่อเป็น  
 Heraunthek (ไม่ใช่ชื่อเดิมของครูพ่อ พอดีเป็นคนเมืองมาก สะดวกต่อการสื่อสาร)

ขอเชิญสมัครเข้าและข้อมูลติดต่อ

Facebook  
Teerapat Chomchoey

Instagram  
jschomchoey

Line ID  
jschomchoey

หมายเลขโทรศัพท์  
0616980000

ไปโฉะคุณ

คุณจะต้องให้บัญชีที่มีความเชื่อถือทางศาสนาแยกต่างหากคุณหรือไม่? \*

สังฆ

ได้รับไม่ระบุตน

ไม่สังฆ

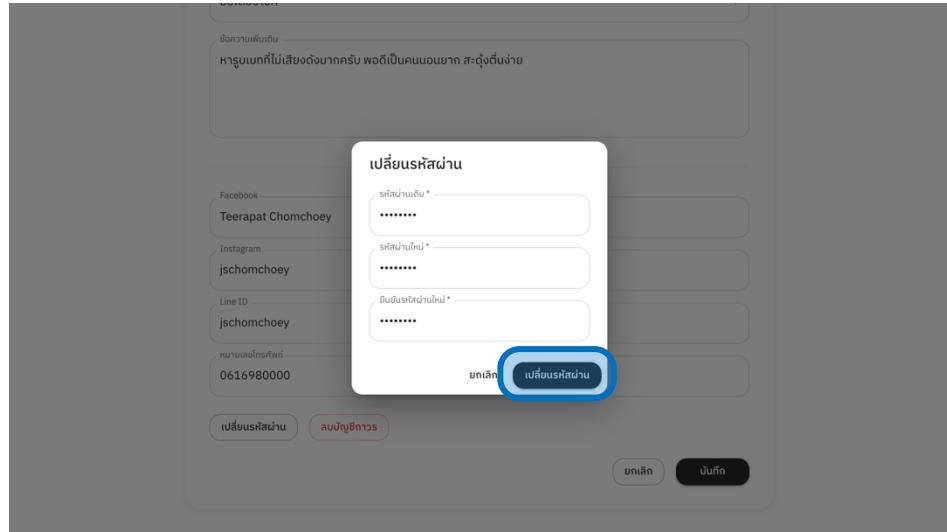
คุณต้องการรูปแบบที่อยู่ร่วมกันระหว่างชาว佛教หรือไม่? \*

ล่องการ

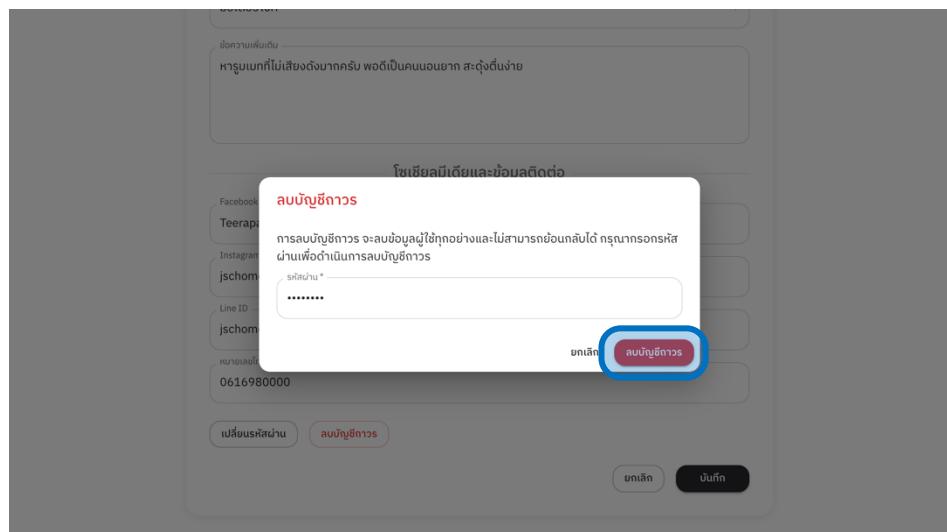
ขึ้นลงกับสถานที่

ไม่จำเป็น

เมื่อแก้ไขข้อมูลตามต้องการแล้ว ให้กดปุ่ม “บันทึก” เพื่อบันทึกการแก้ไข นอกจากการแก้ไขประวัติส่วนตัว และลักษณะนิสัยแล้ว สามารถที่จะเปลี่ยนรหัสผ่าน และลบบัญชีครัวได้



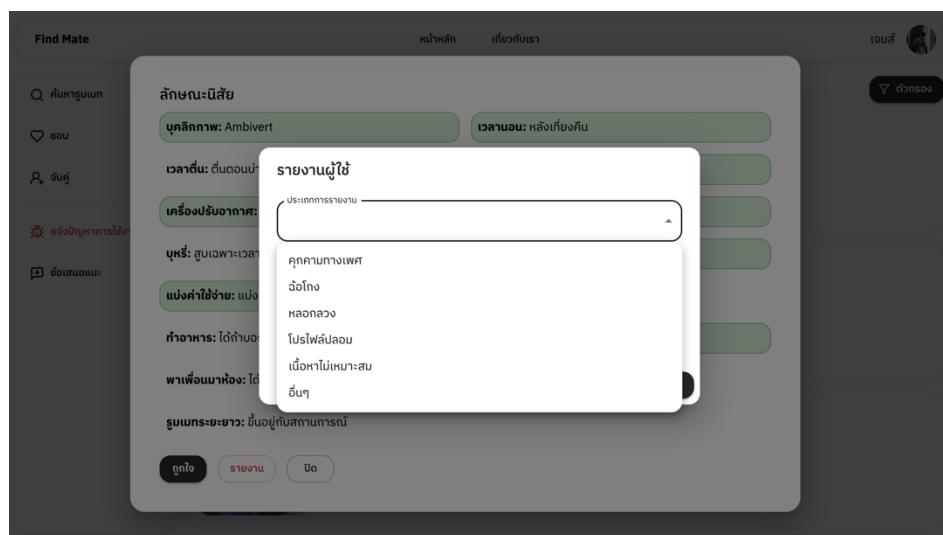
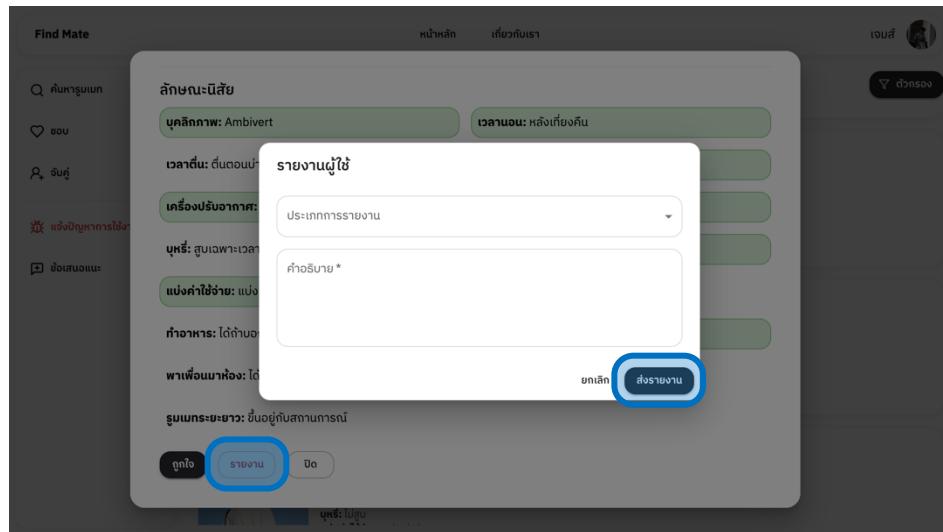
เมื่อกดปุ่ม “เปลี่ยนรหัสผ่าน” จะแสดงหน้าต่างการเปลี่ยนรหัสผ่าน โดยการกรอกรหัสผ่านปัจจุบัน และทำการกรอกรหัสผ่านใหม่ที่ต้องการ จากนั้นยืนยันรหัสผ่านใหม่อีกครั้ง เมื่อกรอกเสร็จสิ้นให้กดปุ่ม “เปลี่ยนรหัสผ่าน” เป็นอันเสร็จสิ้น



เมื่อผู้ใช้กดปุ่ม “ลับบัญชีถาวร” จะแสดงหน้าต่างการยืนยันการลบบัญชี ให้ผู้ใช้กรอกรหัสผ่านปัจจุบันเพื่อเป็นการยืนยันการลบบัญชี จากนั้นกดปุ่ม “ลับบัญชีถาวร” เป็นอันเสร็จสิ้น โดยจะไม่เก็บข้อมูลใด ๆ ของผู้ใช้ไว้

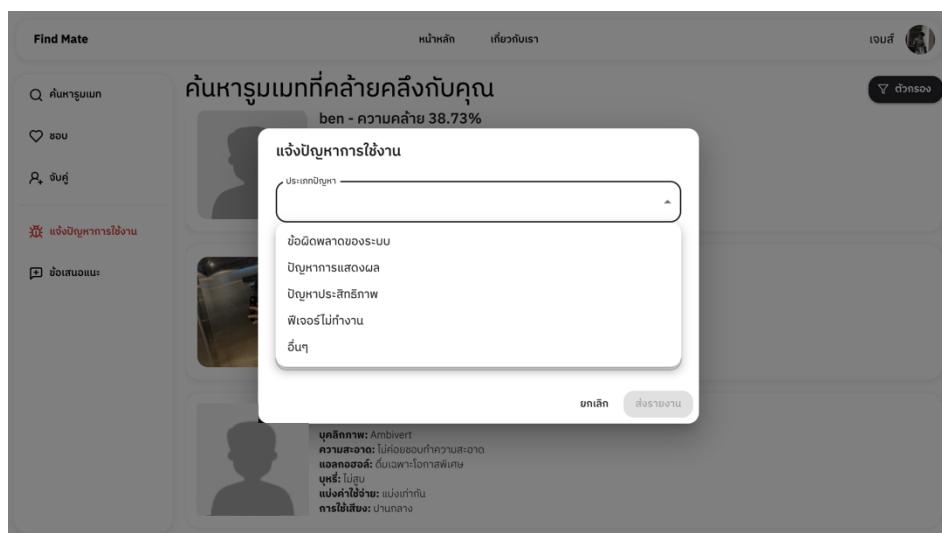
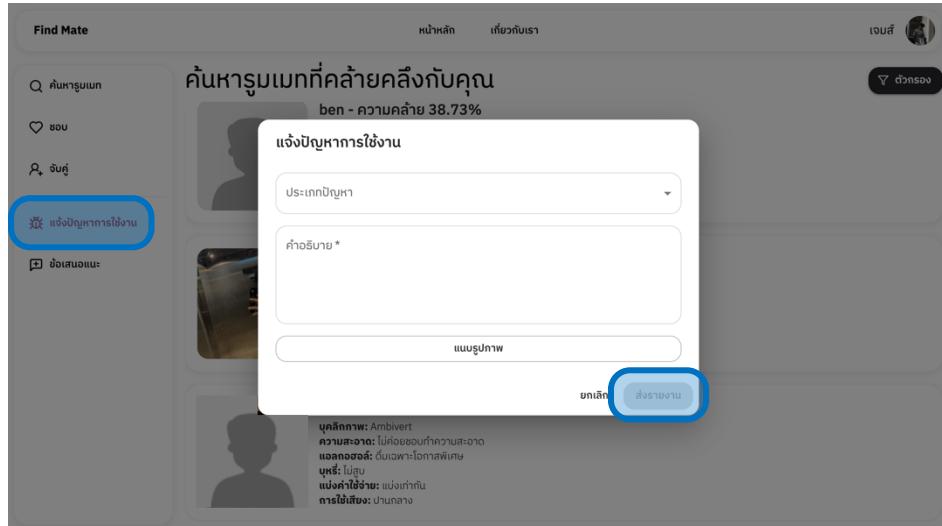
#### 4. การรายงานปัญหาให้ผู้ดูแลระบบและการให้คะแนนความพึงพอใจ

##### 4.1 การรายงานผู้ใช้



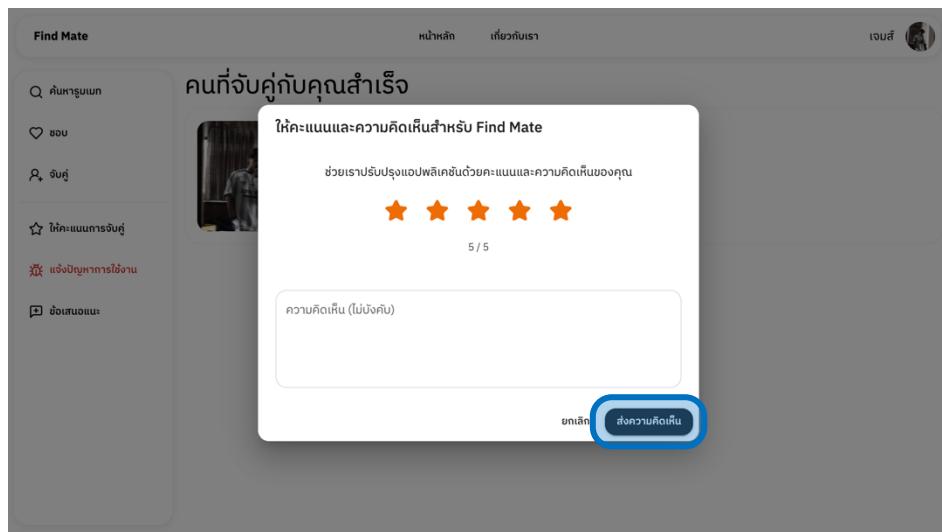
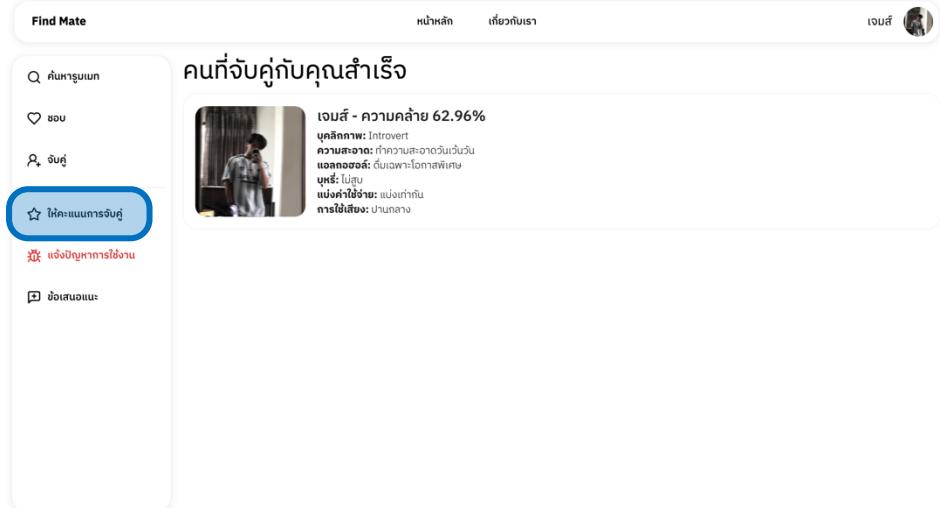
การแจ้งปัญหาเกี่ยวกับผู้ใช้ เมื่อผู้ใช้กดปุ่ม รายงาน ในหน้าต่างໂປຣໄຟລ໌ຂອງຮູມເມທນ້ນ ໆ ระบบจะแสดงหน้าต่างรายงานผู้ใช้ โดยให้ผู้ใช้ได้ทำการเลือกประเภทการรายงาน และเขียนคำอธิบายรายงานให้ระเอียดและชัดเจน ให้แก่ผู้ดูแลระบบ เพื่อตรวจสอบและแก้ไขตามความเหมาะสม

## 4.2 การรายงานปัญหาการใช้งาน



การแจ้งปัญหาการใช้งาน ให้ผู้ใช้กดปุ่ม แจ้งปัญหาการใช้งาน ในหน้าหลักของระบบ ระบบจะแสดงหน้าต่างรายงานแจ้งปัญหาการใช้งาน โดยให้ผู้ใช้ได้ทำการเลือกประเภทการรายงาน แล้วเขียนคำอธิบายรายงานให้ระเอียดและชัดเจน หากมีรูปภาพหลักฐาน หรือส่วนที่พบรูปปัญหา สามารถที่จะแนบรูปภาพ ให้แก่ผู้ดูแลระบบ เพื่อตรวจสอบและแก้ไขตามความเหมาะสม

### 4.3 การให้คะแนนความพึงพอใจ



ในหน้าจับคู่สำเร็จ จะมีเมนูสำหรับให้คะแนนความพึงพอใจระบบการจับคู่รวมไปถึงการใช้งานเว็บแอปพลิเคชันโดยรวม โดยมีระดับการให้คะแนน 1 ถึง 5 ดาว

### 4.3 การใช้งานฝั่งผู้ดูแลระบบ

Find Mate

เข้าสู่ระบบ

อีเมล \*

admin.admin@admin.com

รหัสผ่าน \*

เข้าสู่ระบบ

Find Mate

เข้าสู่ระบบ

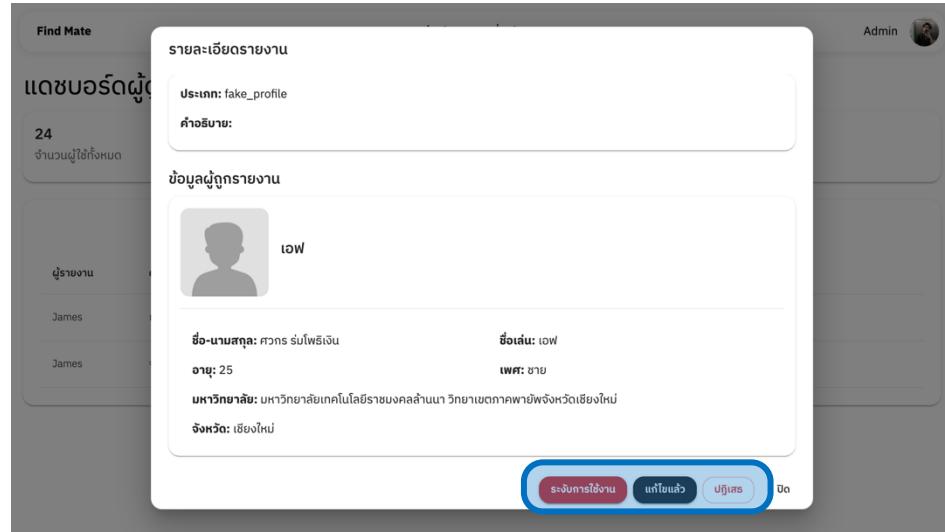
Admin

แดดชบอร์ดผู้ดูแลระบบ

จำนวนผู้ใช้	จำนวนการจับคู่ก่อหนี้	จำนวนรายงานเก็บหนี้
24 จำนวนผู้ใช้ก่อหนี้	7 จำนวนการจับคู่ก่อหนี้	11 จำนวนรายงานเก็บหนี้

ผู้รายงาน	ผู้ถูกรายงาน	ประเภท	สถานะ	วันที่	การจัดการ
James	เจฟ	fake_profile	resolved	20/2/2568	<span>ดูรายละเอียด</span> <span>รับแจ้งการจับคู่</span> <span>ลบรายงาน</span>
James	เจมส์	other	pending	19/2/2568	<span>ดูรายละเอียด</span> <span>รับแจ้งการจับคู่</span> <span>ลบรายงาน</span>

ทำการล็อกอินเข้าสู่ระบบด้วยบัญชีของผู้ดูแลระบบ โดยใช้งานหน้าเข้าสู่ระบบเดียวกัน ร่วมกับผู้ใช้งานทั่วไป ระบบจะนำทางมาอยู่หน้าแดดชบอร์ดผู้ดูแลระบบ โดยจะมีสถิติผู้ใช้งาน การรายงานทั้งหมด การจัดการผู้ใช้และการติดตามความพึงพอใจของผู้ใช้งาน



หมวดหมู่ 'รายงานผู้ใช้' จะแสดงผลรายงานผู้ใช้ไม่พึงประสงค์ที่ถูกรายงานเข้ามายังจากผู้ใช้สามารถดูรายละเอียดของผู้ใช้ที่ถูกรายงานว่ามีปัญหาตามที่มีรายงานเข้ามาหรือไม่ หากมีปัญหาจริงสามารถรับจัดการใช้งานบัญชีได้ หรือหากไม่มีปัญหาสามารถปฏิเสธการรายงานได้และสามารถลบรายงานทึ้งได้หากจัดการเรียบร้อยแล้ว

Find Mate		หน้าหลัก	ເພີ້ວກັນເຮົາ	Admin	
ແດບບອດຜູ້ດູແລະບັບ					
ຈຳນວນຜູ້ໃຊ້ທັງໝົດ	ຈຳນວນການຈັບຜູ້ໃຊ້ທັງໝົດ	ຈຳນວນຮ່າຍຈາກຜູ້ໃຊ້ທັງໝົດ			
24 ຈຳນວນຜູ້ໃຊ້ທັງໝົດ	7 ຈຳນວນການຈັບຜູ້ໃຊ້ທັງໝົດ	11 ຈຳນວນຮ່າຍຈາກຜູ້ໃຊ້ທັງໝົດ			
รายงานຜູ້ໃຊ້	รายงานຮັບບ	ມັນເສນອແນະ	ຈົດການຜູ້ໃຊ້ທັງໝົດ	ຮົວ	
ຜູ້ຮ່າຍ	ປະເທດ	ຄໍາອືບນາຍ	ສະພາ:	ວັນທີ	ການຈັດກາ
James	bug	ລັບບອຍຄົນ	pending	25/2/2568	<button>ດູຮາຍລະເບີດ</button>
ເວິໂງ	other	ບັນຍາການໃຫຍ່ອອົບຮັບ ອຸນຫຼາມເຕີບ	pending	19/2/2568	<button>ດູຮາຍລະເບີດ</button>
ກົ່ອງກາຣັດ	bug	iPhone 14 Pro ໃດໄດ້ປັດຕິ	resolved	19/2/2568	<button>ດູຮາຍລະເບີດ</button>
James	ui	ລັບບອຍຄົນ	pending	18/2/2568	<button>ດູຮາຍລະເບີດ</button>
James	feature	Test	pending	18/2/2568	<button>ດູຮາຍລະເບີດ</button>

หมวดหมู่ 'รายงานระบบ' จะเป็นส่วนที่ผู้ใช้งานแจ้งปัญหาเกี่ยวกับการใช้งานเว็บแอปพลิเคชัน สามารถนำข้อมูลการแจ้งไปปรับปรุงการทำงานของเว็บได้

ໝາວດໝູ່ຂໍອເສນອແນະ ເປັນຫ່ວງທາງເປີດໂວກາສັບພັ້ງຄວາມເທັນຂອງຜູ້ໃຊ້ຈາກເພື່ອນຳ  
ຄຳແນະນຳມາພັນນາເວັບແອປພລິເຄີ້ນໃຫ້ດີຍິ່ງໜີ້ນ

ໝາວດໝູ່ຈັດກາຮັບຜູ້ໃຊ້ທີ່ໜົດ ຜູ້ຊີ້ວິດແລະຮັບສາມາຮຽນຮັບການໃຊ້ຈາກເວັບໄຕ  
ໂດຍຕຽນໂດຍໄມ່ຕ້ອງຮອກການແຈ້ງຮາຍງານເຂົ້ານາ

Find Mate

หน้าหลัก เสิร์ฟกันเรา Admin

**ແດນບອຮດຝູດແລຣະບບ**

24 ຈ່ານວຍຸໃຊ້ກັ່ງໜບດ	7 ຈ່ານວນກາຮົບຄູ່ກັ່ງໜບດ	11 ຈ່ານວນຮາຍານກັ່ງໜບດ
-------------------------	----------------------------	--------------------------

ຮານຈານແນ້ໃຊ້ ຮາຍານຮະບບ ຂັ້ນເສນອແມ ອົດກາຮົບໃຊ້ກັ່ງໜບດ **ຮົວ**

ຮົວແອບພລິເຄຂັ້ນກັ່ງໜບດ

**5.0**  
ຄໍາແສ່ຍ່າຍາ 1 ຮົວ

ຮົບຜູ້ໃຊ້	ອີເມວ	ຄະແນນ	ຄວາມຄົດເຫັນ	ວັນທີ
ໄປຮະບຸ	ໄປຮະບຸ	5 ★★★★★	ໄປມີຄວາມຄົດເຫັນ	2/3/2568

ໜ້າວັດໜູ້ຮືວ ແສດງພລຄະແນນຄວາມພຶງພອໃຈຮະບບກາຮົບຈັບຄູ່ຮົມໄປຄືງກາຮົບໃຊ້ຈ່ານເວັບແອບພລິເຄຂັ້ນໂດຍຮົມທີ່ຜູ້ໃຊ້ໃຫ້ຄະແນນນາມ ໂດຍຄະແນນຈະເປັນຕົວບ່າງວ່າເວັບແອບພລິເຄຂັ້ນນີ້ດີໃນສາຍຕາຜູ້ໃຊ້ຈ່ານແຄ້ງໃໝ່

ภาคผนวก ค  
แบบประเมินความพึงพอใจ

### แบบประเมินผลแอปพลิเคชันหารูมเมทด้วย K-NN

คณะวิศวกรรมศาสตร์ หลักสูตร วศ.บ.คพ.(เที่ยบโอน) วิศวกรรมศาสตร์ มทร.ล้านนา เชียงใหม่

**คำชี้แจง** แบบทดสอบนี้จัดทำขึ้นเพื่อติดตามประเมินความพึงพอใจต่อการใช้งานแอปพลิเคชันหารูมเมทด้วย K-NN เพื่อเป็นข้อมูลในการปรับปรุงและพัฒนา โปรดให้ข้อมูลที่เป็นจริงตามความคิดเห็นของท่าน โดยทำเครื่องหมาย  หรือเขียนตอบในช่องที่กำหนดให้ขอขอบคุณที่ให้ความร่วมมือ

**ตอนที่ 1** ข้อมูลส่วนบุคคลของผู้ตอบแบบสอบถาม โปรดทำเครื่องหมาย  ลงในช่อง □

1. เพศ  ชาย  หญิง
2. อายุ  ต่ำกว่า 20 ปี  20 - 40 ปี  อื่น ๆ ..... (โปรดระบุ)
3. อาชีพ  นักเรียน/นักศึกษา  ผู้ประกอบการ  อื่น ๆ ..... (โปรดระบุ)

**ตอนที่ 2** ความพึงพอใจของผู้ใช้งานแอปพลิเคชันหารูมเมทด้วย K-NN โปรดเลือกตอบโดยใส่เครื่องหมาย  ลงในช่องว่างตามความคิดเห็นของท่าน ซึ่งมีเกณฑ์การพิจารณาตามค่าเลขดังต่อไปนี้

ระดับ 5 หมายถึง ความพึงพอใจอยู่ในระดับดีมาก

ระดับ 4 หมายถึง ความพึงพอใจอยู่ในระดับดี

ระดับ 3 หมายถึง ความพึงพอใจอยู่ในระดับปานกลาง

ระดับ 2 หมายถึง ความพึงพอใจอยู่ในระดับน้อย

ระดับ 1 หมายถึง ความพึงพอใจอยู่ในระดับควรปรับปรุง

หัวข้อการประเมิน	ระดับความพึงพอใจ				
	5	4	3	2	1
<b>1. ด้านการใช้งานระบบ</b>					
1.1 ระบบใช้งานง่าย สะดวก					
1.2 ด้านการออกแบบ Design และการจัดรูปแบบแอปพลิเคชัน					
1.3 การจัดรูปแบบของแอปพลิเคชันง่ายต่อการอ่านและการใช้งาน					
1.4 ความเหมาะสมในการใช้สัญลักษณ์ในการสื่อความหมาย					

หัวข้อการประเมิน	ระดับความพึงพอใจ				
	5	4	3	2	1
2. ประสิทธิภาพในการทำงาน					
2.1 ระบบสามารถจับคู่ผู้ใช้ที่กดถูกใจตรงกันอย่างถูกต้อง					
3. ประโยชน์และการนำไปใช้					
3.1 แอปพลิเคชันมีประโยชน์ต่อผู้ใช้งาน					
3.2 แอปพลิเคชันตอบโจทย์ความต้องการของผู้ใช้					
3.3 แอปพลิเคชันมีผลต่อการนำไปใช้ในชีวิตประจำวัน					

โปรดแสดงความคิดเห็นและข้อเสนอแนะเกี่ยวกับการพัฒนาโครงการ

.....

.....

.....

ขอขอบพระคุณเป็นอย่างสูงในการให้ข้อมูลที่เป็นจริงของท่าน

## แบบประเมินผลแอปพลิเคชันหารูមเมทด้วย K-NN ออนไลน์

**แบบประเมินความพึงพอใจต่อ เว็บ  
แอปพลิเคชันหารูមเมทด้วย K-NN**

tpcc.general@gmail.com [Switch account](#) 

Your email will be recorded when you submit this form

\* Indicates required question

การประเมินความพึงพอใจในการใช้งานระบบ

ระบบใช้งานง่าย สะดวก \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

ต้านการออกแบบ Design และการจัดรูปแบบแอปพลิเคชัน \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

การจัดรูปแบบของแอปพลิเคชันง่ายต่อการอ่านและการใช้งาน \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

ความเหมาะสมในการใช้สัญลักษณ์ในการสื่อความหมาย \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

!

[Back](#) [Next](#) [Clear form](#)

# แบบประเมินความพึงพอใจต่อ เว็บ แอปพลิเคชันหารูปเมทด้วย K-NN

tpcc.general@gmail.com [Switch account](#)

Your email will be recorded when you submit this form

\* Indicates required question

**การประเมินประสิทธิภาพในการทำงาน**

ระบบสามารถจับคู่ได้ที่กดถูกใจตรงกันอย่างถูกต้อง \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

ระบบสามารถแสดงรายการการค้นหาได้เป็นลำดับอย่างถูกต้อง \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

แอปพลิเคชันสามารถทำงานได้อย่างราบรื่นและไม่เกิดปัญหาหรือความล่าช้าในขณะที่ใช้งาน \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

[Back](#) [Next](#) [Clear form](#)

This content is neither created nor endorsed by Google. - [Terms of Service](#) - [Privacy Policy](#)  
 Does this form look suspicious? [Report](#)

Google Forms

# แบบประเมินความพึงพอใจต่อ เว็บ แอปพลิเคชันหารูปเมทด้วย K-NN

tpcc.general@gmail.com [Switch account](#)

Your email will be recorded when you submit this form

\* Indicates required question

**การประเมินด้านประโยชน์และการนำไปใช้**

แอปพลิเคชันมีประโยชน์ต่อผู้ใช้งาน \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

แอปพลิเคชันตอบโจทย์ความต้องการของผู้ใช้ \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

แอปพลิเคชันมีผลต่อการนำไปใช้ในชีวิตประจำวัน \*

น้อยมาก  
 น้อย  
 ปานกลาง  
 มาก  
 มากที่สุด

[Back](#) [Submit](#) [Clear form](#)

This content is neither created nor endorsed by Google. - [Terms of Service](#) - [Privacy Policy](#)  
 Does this form look suspicious? [Report](#)

**Google Forms**

ການພັນວັດ ຈ.  
ປະວັດຜູ້ເຂີຍນໂຄຮງການ

### ประวัติผู้เขียนโครงการ



ชื่อ - นามสกุล (ภาษาไทย)	นายธีระภัทร ชุมเชย
ชื่อ - นามสกุล (ภาษาอังกฤษ)	Mr. Teerapat Chomchoei
รหัสนักศึกษา	65543206019-3
สถานที่อยู่ที่ติดต่อได้	เลขที่ 71 หมู่ที่ 4 บ้านน้ำโทก ตำบลแม่ทะ อำเภอแม่ทะ จังหวัดลำปาง 52150
โทรศัพท์	061-698-9385
Email	tpcc.general@gmail.com

#### ประวัติการศึกษา

พ.ศ. 2562	ประกาศนียบัตรสำเร็จการศึกษาตามหลักสูตรวิชาชีพ (ปวช.) วิทยาลัยเทคนิคลำปาง
พ.ศ. 2564	ประกาศนียบัตรสำเร็จการศึกษาตามหลักสูตรวิชาชีพ (ปวส.) วิทยาลัยเทคนิคลำปาง
พ.ศ. 2568	คณะวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมไฟฟ้า หลักสูตรวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีราชมงคล ล้านนาเชียงใหม่

### ประวัติผู้เขียนโครงการ



ชื่อ - นามสกุล (ภาษาไทย)	นายจิตตพงษ์ จงใจ
ชื่อ - นามสกุล (ภาษาอังกฤษ)	Mr. Jittapong Jongjai
รหัสนักศึกษา	65543206007-8
สถานที่อยู่ที่ติดต่อได้	เลขที่ 652 หมู่ที่ 2 ตำบลศาลา อำเภอเกาะคา จังหวัดลำปาง 52130
โทรศัพท์	061-795-6385
Email	jittapongj@gmail.com

#### ประวัติการศึกษา

พ.ศ. 2562	ประกาศนียบัตรสำเร็จการศึกษาตามหลักสูตรวิชาชีพ (ปวช.) วิทยาลัยเทคนิคลำปาง
พ.ศ. 2564	ประกาศนียบัตรสำเร็จการศึกษาตามหลักสูตรวิชาชีพ (ปวส.) วิทยาลัยเทคนิคลำปาง
พ.ศ. 2568	คณะวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมไฟฟ้า หลักสูตรวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีราชมงคล ล้านนาเชียงใหม่