

โครงการเลขที่ วศ.คพ. P809-2/2566

เรื่อง

แพลตฟอร์มการแลกเปลี่ยนความคิดสร้างสรรค์ดิจิทัล: การเสริมสร้างการเรียนรู้ผ่าน  
**Gallery Walk**

โดย

นายณานารีป ภู่สว่าง รหัส 630612097

นางสาวณัฐวรรณ เรียบเรียง รหัส 630612099

นายปัณฑร กันทรัพย์ รหัส 630612105

โครงการนี้

เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่  
ปีการศึกษา 2566

**PROJECT No. CPE P809-2/2566**

**Digital Creativity Exchange Platform: Enhancing Learning through  
Gallery Walk**

**Yanatip Bhoosawang 630612097**

**Nuttawan Reabreang 630612099**

**Panthon Kansap 630612105**

**A Project Submitted in Partial Fulfillment of Requirements  
for the Degree of Bachelor of Engineering  
Department of Computer Engineering  
Faculty of Engineering  
Chiang Mai University  
2023**

หัวข้อโครงการ : แพลตฟอร์มการแลกเปลี่ยนความคิดสร้างสรรค์ดิจิทัล: การเสริมสร้างการเรียนรู้ผ่าน  
Gallery Walk

: Digital Creativity Exchange Platform: Enhancing Learning through  
Gallery Walk

โดย : นายณานาธิป ภู่สว่าง รหัส 630612097  
นางสาวณัฐวรรณ เรียมเรียง รหัส 630612099  
นายปัณฑร กันทรพย รหัส 630612105

ภาควิชา : วิศวกรรมคอมพิวเตอร์

อาจารย์ที่ปรึกษา : ผศ. โอดิม โพธิ์กานท์

ปริญญา : วิศวกรรมศาสตรบัณฑิต

สาขา : วิศวกรรมคอมพิวเตอร์

ปีการศึกษา : 2566

---

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ได้อนุมัติให้โครงการนี้เป็นส่วน-  
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมคอมพิวเตอร์)

..... หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์

(ศ.ดร. สันติ พิทักษ์กิจนุกร)

คณะกรรมการสอบโครงการ

..... ประธานกรรมการ

(ผศ. โอดิม โพธิ์กานท์)

..... กรรมการ

(อ.ดร. ชินวัตร อิศราดิสัยกุล)

..... กรรมการ

(อ.ดร. ธนาทิพย์ จันทร์คง)

หัวข้อโครงการ	: แพลตฟอร์มการแลกเปลี่ยนความคิดสร้างสรรค์ดิจิทัล: การเสริมสร้างการเรียนรู้ผ่าน Gallery Walk
	: Digital Creativity Exchange Platform: Enhancing Learning through Gallery Walk
โดย	: นายณานารีป ภู่สว่าง รหัส 630612097 นางสาวณัฐวรรณ เรียบเรียง รหัส 630612099 นายปัณฑร กันทรพย รหัส 630612105
ภาควิชา	: วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	: ผศ. โอดี้ โพธิ์กานท์
ปริญญา	: วิศวกรรมศาสตรบัณฑิต
สาขา	: วิศวกรรมคอมพิวเตอร์
ปีการศึกษา	: 2566

---

### บทคัดย่อ

Gallery Walk คือ การเรียนรู้แบบเดิมชนิดงาน ซึ่งเป็นเทคนิคการเรียนรู้ที่กระตุ้นให้ผู้เรียนแลกเปลี่ยนข้อมูล ความรู้ หรือประสบการณ์ร่วมกันโดยใช้ผลงานที่ผู้เรียนแต่ละกลุ่มร่วมกันสร้างสรรค์ขึ้นเพื่อเป็นสื่อกลางในการถ่ายทอดและใช้กระดานสนทนากำนัลในการรับคำติชมและข้อเสนอแนะจากผู้เข้าร่วม เพื่อนำข้อมูลที่ได้รับกลับมาพัฒนาผลงานของกลุ่มหรือเติมเต็มความรู้ความเข้าใจของตัวผู้เรียนเอง

ในปัจจุบันการจัดกิจกรรมในรูปแบบ Gallery Walk นั้นมีความยุ่งยากในการจัดการ เนื่องจากผู้จัดงานกิจกรรม (Event Manager) จะต้องจัดเตรียมสถานที่ สื่อการนำเสนอ และการจัดการเอกสารต่าง ๆ ที่เกี่ยวข้องกับกิจกรรม เป็นภาระต่อผู้จัดงานกิจกรรม นอกจากนี้ยังไม่สามารถจัดกิจกรรมในรูปแบบ Hybrid ได้ในกรณีที่ผู้เข้าร่วมงาน (Guest) ต้องการที่จะเยี่ยมชมผลงาน โครงการต่าง ๆ แบบออนไลน์ เพื่อให้ Virtual Money และแสดงความคิดเห็นต่าง ๆ และส่วนสุดท้ายการแสดงผลหลังสิ้นสุดกิจกรรมไม่สามารถแสดงได้ทันที ทำให้การประกาศผลของผลงานที่ได้รับการประเมินสูงสุดทำได้ล่าช้า

จากเหตุผลข้างต้นที่กล่าวมานี้ คณะผู้จัดทำจึงได้จัดทำโครงการเพื่อพัฒนาเว็บแอปพลิเคชันสำหรับการจัดกิจกรรม Gallery Walk ที่สามารถจัดกิจกรรมทั้งในรูปแบบ online และ hybrid ซึ่งสามารถให้ผู้จัดงานกิจกรรมสามารถจัดการกิจกรรมได้ง่ายขึ้น และสามารถให้ผู้เข้าร่วมงานสามารถเยี่ยมชมผลงาน โครงการต่าง ๆ ทั้ง online และ hybrid ซึ่งยังสามารถแสดงความคิดเห็นต่าง ๆ และส่วนสุดท้ายสามารถแสดงผลได้ทันที ว่าผลงานได้รับการประเมินสูงสุดหลังเสร็จสิ้นงานกิจกรรม

Project Title : Digital Creativity Exchange Platform: Enhancing Learning through  
Gallery Walk

Name : Yanatip Bhoosawang 630612097  
Nuttawan Reabreang 630612099  
Panthon Kansap 630612105

Department : Computer Engineering

Project Advisor : Asst. Prof. Dome Potikanond

Degree : Bachelor of Engineering

Program : Computer Engineering

Academic Year : 2023

---

## **ABSTRACT**

Gallery Walk is the traditional way of learning and viewing works. This is a learning technique that encourages students to exchange information. Shared knowledge or experience by using the work that each group of learners jointly created as a medium for transferring and using a discussion board as a medium for receiving feedback and suggestions from others. To use that feedback to develop the group's work or complete the learners' own knowledge and understanding.

At present, organizing an activity in the form of a gallery walk is difficult to manage. This is because the event organizer (the event manager) must prepare the venue, communication media and managing various documents related to activities. It is a burden on event organizers. In addition, activities cannot be organized in a hybrid format in cases where attendees (guests) would like to visit the various projects online in order to provide virtual money and express various opinions. And the last part is that it cannot immediately show which work received the highest evaluation after the event has been completed.

From the above-mentioned The organizing team therefore created a project to develop a web application for organizing Gallery Walk activities in a hybrid format, which can allow event organizers to manage their activities more easily. And it can allow attendees to visit the various projects online and express various opinions, and the last part can immediately show which work received the highest evaluation after the event has finished.

## กิตติกรรมประกาศ

โครงการนี้ได้เสร็จสมบูรณ์ด้วยความกรุณาอย่างยิ่งจาก ผศ.ดร. โพธิ์กานนท์ ผู้ที่ได้มอบเวลาและความกรุณาเพื่อเป็นที่ปรึกษาในโครงการนี้ ได้ให้คำแนะนำที่มีคุณค่าและแนวทางที่ช่วยให้โครงการเกิดขึ้นอย่างสมบูรณ์ และประสบความสำเร็จ

ไม่เพียงแต่นั้นเท่านั้น ขอขอบคุณอย่างสูงสำหรับคำปรึกษาจาก อ.ดร.ชินวัตร อิศราดิสัยกุล และ ผศ.ดร.ธนา ทิพย์ จันทร์คง ที่ได้มอบคำแนะนำและเสนอแนวทางที่มีประสิทธิภาพ ทำให้โครงการเกิดประสิทธิภาพมากขึ้น

นอกจากนี้ เรายังขอขอบคุณเพื่อนร่วมทีมทุกคนที่ให้การสนับสนุนและกำลังใจ และขอบคุณครอบครัวที่เป็นกำลังใจสำคัญในการผ่านช่วงเวลาที่ท้าทายไปด้วยกัน ขอบคุณทุกท่านที่เคยให้คำแนะนำและความช่วยเหลือ โดยที่ไม่ระบุนาม ทุกคำปรึกษาและการสนับสนุนที่มีค่ามากสำหรับเรา

หากหนังสือโครงการเล่มนี้มีข้อผิดพลาดประการใด ผู้จัดทำขออภัยรับด้วย ความยินดียิ่งและขออภัยมา ณ ที่นี้ด้วย

นายญาณาริป ภู่สว่าง  
นางสาวณัฐวรรณ เรียบเรียง  
นายปันท์ธร กันทร์พย์  
20 มีนาคม 2567

# สารบัญ

บทคัดย่อ . . . . .	๑
Abstract . . . . .	๑
กิตติกรรมประกาศ . . . . .	๑
สารบัญ . . . . .	๑
สารบัญรูป . . . . .	๑
สารบัญตาราง . . . . .	๑
<b>1 บทนำ</b>	<b>1</b>
1.1 ที่มาของโครงการ . . . . .	1
1.2 วัตถุประสงค์ของโครงการ . . . . .	1
1.3 ขอบเขตของโครงการ . . . . .	1
1.3.1 ขอบเขตด้านซอฟต์แวร์ . . . . .	1
1.4 ประโยชน์ที่ได้รับ . . . . .	2
1.5 เทคโนโลยีและเครื่องมือที่ใช้ . . . . .	3
1.5.1 เทคโนโลยีด้านอาร์ดแวร์ . . . . .	3
1.5.2 เทคโนโลยีด้านซอฟต์แวร์ . . . . .	3
1.6 แผนการดำเนินงาน . . . . .	3
1.7 บทบาทและความรับผิดชอบ . . . . .	3
<b>2 หัวข้อที่เกี่ยวข้อง</b>	<b>4</b>
2.1 ด้านโครงสร้างเว็บแอปพลิเคชัน . . . . .	4
2.1.1 MVC Architecture . . . . .	4
2.1.2 RESTful API . . . . .	4
2.1.3 ระบบฐานข้อมูล (Database System) . . . . .	5
2.2 ด้านเทคโนโลยี . . . . .	6
2.2.1 HTML . . . . .	6
2.2.2 CSS . . . . .	7
2.2.3 TypeScript . . . . .	7
2.2.4 React JS . . . . .	7
2.2.5 Mantine UI . . . . .	8
2.2.6 PostgreSQL . . . . .	8
2.2.7 Prisma . . . . .	9
2.3 ด้าน User Interface . . . . .	10
2.3.1 Design Thinking . . . . .	10
2.4 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ . . . . .	11
<b>3 โครงสร้างและขั้นตอนการทำงาน</b>	<b>13</b>
3.1 หลักการทำงานของระบบ . . . . .	13
3.1.1 การทำงานของระบบ (System Architecture) . . . . .	13
3.1.2 เส้นทางของผู้ใช้ (User Flow) . . . . .	14
3.1.3 โครงสร้างฐานข้อมูล (Database Schema) . . . . .	16
3.2 ส่วนเชื่อมต่อระหว่างผู้ใช้งานกับระบบ (User Interface) . . . . .	20
3.2.1 หน้าเริ่มต้นการใช้งาน (Homepage) . . . . .	20
3.2.2 หน้าเข้าสู่ระบบ (Login) . . . . .	22
3.2.3 ผู้สร้างงานกิจกรรม (Event manager) . . . . .	23

3.2.4 ผู้นำเสนองาน (Presenter) . . . . .	28
3.2.5 ผู้เข้าร่วมงาน (Guest) . . . . .	30
<b>4 การทดลองและผลลัพธ์</b>	<b>33</b>
4.1 การประเมินความพึงพอใจของผู้ใช้ . . . . .	33
4.2 ผลการประเมินความพึงพอใจของผู้ใช้ . . . . .	33
4.3 สรุปผลการประเมินความพึงพอใจของผู้ใช้ . . . . .	38
<b>5 บทสรุปและข้อเสนอแนะ</b>	<b>39</b>
5.1 สรุปผล . . . . .	39
5.2 ปัญหาที่พบและแนวทางการแก้ไข . . . . .	39
5.3 ข้อเสนอแนะและแนวทางการพัฒนาต่อ . . . . .	39
<b>บรรณานุกรม</b>	<b>42</b>
<b>ก คู่มือการติดตั้ง (สำหรับการพัฒนา)</b>	<b>44</b>
ก.1 ฝั่งหน้าบ้าน (Front-End) . . . . .	44
ก.2 ฝั่งหลังบ้าน (Back-End) . . . . .	44
<b>ข คู่มือการใช้งาน</b>	<b>46</b>
<b>ประวัติผู้เขียน</b>	<b>47</b>

## สารบัญรูป

2.1 HTML . . . . .	7
2.2 React JS . . . . .	8
2.3 Mantine UI . . . . .	8
2.4 PostgreSQL . . . . .	9
2.5 Prisma ORM . . . . .	10
2.6 กระบวนการออกแบบ Design Thinking . . . . .	11
3.1 System Architecture . . . . .	13
3.2 User Flow . . . . .	14
3.3 Database Schema . . . . .	16
3.4 หน้า Homepage . . . . .	20
3.5 หน้า Service . . . . .	20
3.6 หน้า About us . . . . .	21
3.7 หน้า Contact . . . . .	21
3.8 หน้า Login สำหรับ Event manager และ Presenter . . . . .	22
3.9 หน้าแสดงข้อมูลรายละเอียดของกิจกรรมที่สร้าง (Event manager dashboard) . . . . .	23
3.10 หน้าแสดงข้อมูลรายละเอียดของกิจกรรมที่สร้าง โดยแบ่ง Tab เป็น Event information, Project และ Result . . . . .	24
3.11 หน้าแสดงรายการโครงการที่เข้าร่วมกิจกรรม . . . . .	24
3.12 หน้าแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม โดยแสดงเป็น Ranking และตาราง Virtual Money . . . . .	25
3.13 ตารางแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม . . . . .	25
3.14 หน้าสร้างกิจกรรม First step (Event information) . . . . .	26
3.15 หน้าสร้างกิจกรรม Second step (Images Dropzone) . . . . .	26
3.16 หน้าสร้างกิจกรรม Final step (Check all information before submit) . . . . .	27
3.17 หน้าเพิ่มโครงการที่เข้าร่วมกิจกรรม (Add project) . . . . .	28
3.18 หน้าแสดงโครงการที่สร้าง (Presenter dashboard) . . . . .	28
3.19 หน้าแสดงข้อมูลรายละเอียดของผลงานที่สร้างในกิจกรรม . . . . .	29
3.20 หน้าเข้าสู่ระบบ (Login) สำหรับผู้เข้าร่วมกิจกรรม (Guest) . . . . .	31
3.21 หน้าแสดงข้อมูลรายละเอียดของงานกิจกรรมที่จัดขึ้น และโครงการที่เข้าร่วมกิจกรรม . . . . .	31
3.22 หน้าให้ Virtual Money และแสดงความคิดเห็นเกี่ยวกับโครงการที่เข้าร่วมกิจกรรม . . . . .	32
3.23 หน้าแสดงความคิดเห็นเกี่ยวกับโครงการที่เข้าร่วมกิจกรรม . . . . .	32
4.1 แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนข้อมูลทั่วไป . . . . .	34
4.2 แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนการออกแบบและการใช้งาน (Design and Usability) . . . . .	35
4.3 แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนฟังก์ชันการทำงาน (Functionality) . . . . .	36
4.4 แบบสำรวจความพึงพอใจของผู้ใช้ในความพึงพอใจรวม . . . . .	37

## สารบัญตาราง

3.1 ตารางข้อมูลผู้ใช้งาน (Users) . . . . .	17
3.2 ตารางข้อมูลงานกิจกรรม (Events) . . . . .	17
3.3 ตารางข้อมูลโครงการ (Projects) . . . . .	18
3.4 ตารางข้อมูลเงินเสมือน (Virtual Money) . . . . .	18
3.5 ตารางข้อมูลแขกผู้เข้าร่วมกิจกรรม (Guests) . . . . .	18
3.6 ตารางข้อมูลภาพตัวอย่างโครงการ (Thumbnails) . . . . .	18
3.7 ตารางข้อมูลเอกสารที่เกี่ยวข้องกับงานกิจกรรม (Documents) . . . . .	19
3.8 ตารางข้อมูลความคิดเห็นของผู้เข้าร่วมกิจกรรม (Comments) . . . . .	19
3.9 ตารางข้อมูลรูปภาพที่เกี่ยวข้องกับโครงการ (Project Images) . . . . .	19

## บทที่ 1

### บทนำ

#### 1.1 ที่มาของโครงการ

โครงการ Gallery Walk มีความหมายในการให้ผู้เรียนมีโอกาสแลกเปลี่ยนความรู้และประสบการณ์ผ่านผลงานที่สร้างขึ้นเป็นสื่อกลาง และการใช้กระดานสนทนาร่วมรับคำแนะนำและคำชี้แจงจากผู้อื่น เพื่อให้สามารถพัฒนาผลงาน หรือเพิ่มเติมความรู้แก่ต้นของได้

การจัดกิจกรรมแบบ Gallery Walk ในปัจุบันมีความยุ่งยากในการจัดการ เนื่องจากต้องมีการจัดเตรียมสถานที่ สื่อการสื่อสาร และการจัดการเอกสารต่าง ๆ ซึ่งเป็นภาระต่อผู้จัดงาน นอกจากนี้ยังไม่สามารถจัดกิจกรรมแบบ Hybrid ได้ โดยเฉพาะในกรณีที่ผู้เข้าร่วมต้องการที่จะเยี่ยมชมผลงานแบบออนไลน์และให้คำแนะนำ

ดังนั้น เพื่อแก้ไขปัญหาดังกล่าว ทีมผู้จัดงานได้พัฒนาเว็บแอปพลิเคชันสำหรับการจัดกิจกรรม Gallery Walk ทั้งในรูปแบบ online และ hybrid เพื่อให้การจัดการกิจกรรมเป็นไปอย่างมีประสิทธิภาพ และเพื่อให้ผู้เข้าร่วมสามารถเยี่ยมชมผลงานและให้ความเห็นออนไลน์ได้ นอกจากนี้ยังสามารถแสดงผลการประเมินได้ทันทีหลังจากเสร็จสิ้นกิจกรรม

#### 1.2 วัตถุประสงค์ของโครงการ

- เพื่อพัฒนาเครื่องมือช่วยดำเนินการจัดกิจกรรมแบบ Gallery Walk ที่รองรับการจัดกิจกรรมทั้งในรูปแบบออนไลน์ และการจัดกิจกรรมยังสถานที่จริง
- เพื่อพัฒนาเครื่องมือให้ผู้ร่วมกิจกรรมสามารถให้ความคิดเห็น และแสดงการสนับสนุนผลงานที่ต้องการได้สะดวกและรวดเร็ว

#### 1.3 ขอบเขตของโครงการ

##### 1.3.1 ขอบเขตด้านซอฟต์แวร์

โครงการนี้ต้องการซอฟต์แวร์ต่อไปนี้ จึงจะสามารถใช้งานได้อย่างมีประสิทธิภาพ

- สามารถใช้งานเว็บไซต์บนระบบปฏิบัติการทั่วไปได้ เช่น Windows, macOS, Linux, Android, iOS และอื่น ๆ

โดยการแสดงผลของเว็บไซต์จะแสดงผลได้เหมาะสม ตามบทบาทผู้ใช้งานและมีฟังก์ชันการทำงานดังต่อไปนี้

- ผู้จัดงานกิจกรรม (Event Manager): คอมพิวเตอร์ส่วนบุคคล

- \* สร้างงานกิจกรรมได้
- \* จัดการข้อมูลกิจกรรมได้
- \* สร้างช่องทางในการเข้าใช้งาน (URL, QR) สำหรับให้ผู้ใช้กลุ่มอื่น ๆ ได้

- \* ตรวจสอบข้อมูลโครงการ/ผลงานที่เข้าร่วมกิจกรรมได้
- \* ดูสรุปข้อมูลผลการจัดกิจกรรมด้านผลการแข่งขัน สถิติเกี่ยวกับผู้เข้าร่วมกิจกรรม สถิติ ด้านการมีส่วนร่วมของผู้ร่วมกิจกรรม
- ผู้นำเสนองาน (Presenter): คอมพิวเตอร์ส่วนบุคคล
  - \* เข้าร่วมงานกิจกรรมผ่านทาง QR Code หรือ URL Link
  - \* เพิ่มโครงการเข้าไปยังงานกิจกรรมได้
  - \* จัดการข้อมูลโครงการได้
  - \* ดู Virtual Money และความคิดเห็นต่อโครงการได้
- แขกผู้เข้าร่วมกิจกรรม (Guest): คอมพิวเตอร์ส่วนบุคคล หรือ โทรศัพท์มือถือ
  - \* เข้าร่วมงานกิจกรรมผ่านทาง QR Code หรือ URL Link
  - \* ดูข้อมูลงานกิจกรรมได้
  - \* ดูข้อมูลโครงการต่าง ๆ ในงานกิจกรรมได้
  - \* ให้ Virtual Money และแสดงความคิดเห็นต่อโครงการต่าง ๆ ได้

## 1.4 ประโยชน์ที่ได้รับ

### ผู้ใช้งาน

- ผู้จัดงานกิจกรรม (Event Manager)
  - สามารถจัดการกิจกรรมได้อย่างมีประสิทธิภาพ
  - สามารถจัดการกิจกรรมทั้งแบบ online และ hybrid ได้
  - สามารถแสดงผลการประเมินได้ทันทีหลังจากเสร็จสิ้นกิจกรรม
- ผู้นำเสนองาน (Presenter)
  - สามารถนำเสนอผลงานได้อย่างมีประสิทธิภาพ
  - สามารถรับความคิดเห็นและคำแนะนำจากผู้เข้าร่วมกิจกรรมได้อย่างมีประสิทธิภาพ
- ผู้เข้าร่วมกิจกรรม (Guest)
  - สามารถเยี่ยมชมผลงานและให้ความเห็นได้อย่างมีประสิทธิภาพ

### ผู้พัฒนา

- ได้รับความรู้และความเข้าใจในการพัฒนาเว็บแอปพลิเคชัน
- ได้ฝึกทักษะในการใช้เทคโนโลยีที่เกี่ยวข้องกับการพัฒนาเว็บแอปพลิเคชัน
- ได้ฝึกทักษะในการทำงานเป็นทีมและทักษะในการวิเคราะห์และแก้ไขปัญหาที่อาจเกิดขึ้นในการพัฒนา

## 1.5 เทคโนโลยีและเครื่องมือที่ใช้

### 1.5.1 เทคโนโลยีด้านฮาร์ดแวร์

### 1.5.2 เทคโนโลยีด้านซอฟต์แวร์

- ภาษาโปรแกรม: TypeScript, JavaScript, HTML, CSS
- ฐานข้อมูล: PostgreSQL, Prisma ORM, Minio (Object Storage)
- เครื่องมือและเทคโนโลยี: React, Mantine UI, Git, GitHub

## 1.6 แผนการดำเนินงาน

ขั้นตอนการดำเนินงาน	ก.ค. 2566	ส.ค. 2566	ก.ย. 2566	ต.ค. 2566	พ.ย. 2566	ธ.ค. 2566	ม.ค. 2567	ก.พ. 2567	มี.ค. 2567
คิดหัวข้อและนำเสนอหัวข้อโครงงาน									
ศึกษาค้นคว้าเกี่ยวกับเทคโนโลยีที่เกี่ยวข้อง									
ออกแบบ UI/UX ของเว็บแอปพลิเคชัน									
พัฒนาเว็บแอปพลิเคชัน									
ทดสอบและปรับปรุงเว็บแอปพลิเคชัน									

## 1.7 บทบาทและความรับผิดชอบ

มีหน้าที่และความรับผิดชอบ ดังนี้

นายญาณอิป ภู่ส่วน รหัสนักศึกษา 630612097 รับผิดชอบในการศึกษาค้นคว้าเทคโนโลยีที่เกี่ยวข้อง, ออกแบบโครงสร้างของเว็บแอปพลิเคชันและพัฒนาเว็บแอปพลิเคชันส่วนหน้าบ้าน (Frontend)

นางสาวณัฐวรรณ เรียมเรียง รหัสนักศึกษา 630612099 รับผิดชอบในการศึกษาค้นคว้าเทคโนโลยีที่เกี่ยวข้อง, ออกแบบโครงสร้างของเว็บแอปพลิเคชันและพัฒนาเว็บแอปพลิเคชันส่วนหน้าบ้าน (Frontend)

นายปัณฑร กันทรพย รหัสนักศึกษา 630612105 รับผิดชอบในการศึกษาค้นคว้าเทคโนโลยีที่เกี่ยวข้อง, ออกแบบโครงสร้างของเว็บแอปพลิเคชันและพัฒนาเว็บแอปพลิเคชันส่วนหลังบ้าน (Backend)

## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

#### 2.1 ด้านโครงสร้างเว็บแอปพลิเคชัน

ในส่วนนี้จะอธิบายถึงโครงสร้างของเว็บแอปพลิเคชันที่ใช้ในการพัฒนา

##### 2.1.1 MVC Architecture

MVC [1] เป็นตัวย่อของคำว่า Model View Controller ใช้เรียกรูปแบบการพัฒนาซอฟต์แวร์ที่มีโครงสร้างซึ่งแบ่งออกมาเป็น 3 ส่วนหลัก ตามตัวย่อของชื่อรูปแบบการพัฒนาซอฟต์แวร์แบบ MVC ถูกนำไปใช้ในขั้นตอนการพัฒนาหลากหลายภาษา เพราะ MVC เป็นเพียงหลักการออกแบบโปรแกรม (Design Pattern) รูปแบบหนึ่งเท่านั้น ซึ่งเป็นที่นิยมมาก ในการนำมาพัฒนาแอปพลิเคชันซอฟต์แวร์แต่ละแพลตฟอร์ม และประยุกต์ใช้ในอีกหลาย ๆ ด้าน

##### ส่วนของ Model (M)

Model คือส่วนของการเก็บรวบรวมข้อมูล ไม่ว่าข้อมูลนั้น ๆ จะถูกจัดเก็บในรูปแบบใดก็ตาม ในฐานข้อมูลแบบเป็น Object Class หรือที่นิยมเรียกว่า VO ( Value Object ) หรือเก็บเป็นไฟล์ข้อมูล เมื่อข้อมูลถูกโหลดเข้ามาจากที่ต่าง ๆ และเข้ามายังส่วนของ Model ตัว Model จะทำการจัดการตรวจสอบข้อมูลให้เป็นรูปแบบที่เหมาะสม เพื่อรอการร้องขอข้อมูลจากส่วนของ Controller

##### ส่วนของ View (V)

View คือส่วนของการแสดงผล หรือส่วนที่จะปฏิสัมพันธ์กับผู้ใช้งาน ( User Interface ) หน้าที่ของ View ในการเขียนโปรแกรมแบบ MVC คือ คอยรับคำสั่งจากส่วนของ Controller และ End User เริ่มแรกเลย ตัว View จะจะได้รับคำสั่งจาก Controller ให้แสดงผลหน้า Home และเมื่อผู้ใช้งานหน้าเว็บกดปุ่มสั่งซื้อ View จะส่งข้อมูลไปให้ Controller เพื่อประมวลผลและแสดงบางอย่างจาก Action นั้น

##### ส่วนของ Controller (C)

Controller คือ ส่วนของการเริ่มทำงานและรับคำสั่ง โดยที่คำสั่นนี้จะเกิดขึ้นในส่วนการติดต่อกับผู้ใช้งาน คือ View เมื่อผู้ใช้งานทำการ Interactive กับ UI view จะเกิดเหตุการณ์ หรือข้อมูลบางอย่างขึ้น ตัววิวจะส่งข้อมูลนั้น มา�ัง Controller ตัว Controller จะทำการประมวลผลโดยบางคำสั่งอาจจะต้องติดต่อกับ Model ก่อน เพื่อทำการประมวลผลข้อมูลอย่างถูกต้องเรียบร้อยแล้วก็จะส่งไปยัง View เพื่อแสดงผลตามคำสั่งที่ End User ร้องขอมา Controller จะทำหน้าที่เป็นตัวกลางระหว่าง Model และ View ให้ทำงานร่วมกันอย่างมีประสิทธิภาพและตรงกับความต้องการของ End User มากที่สุด

##### 2.1.2 RESTful API

RESTful API [2] เป็นอินเทอร์เฟซที่ระบบคอมพิวเตอร์สองระบบใช้เพื่อแลกเปลี่ยนข้อมูลผ่านอินเทอร์เน็ต ได้อย่างปลอดภัย และแอปพลิเคชันทางธุรกิจส่วนใหญ่ต้องสื่อสารกับแอปพลิเคชันภายนอกในอื่น ๆ และของบุคคลที่

สามเพื่อทำงานต่าง ๆ ตัวอย่างเช่น หากต้องการสร้างสิ่งใดสิ่งหนึ่งในระบบ ข้อมูลกับระบบธนาคารของลูกค้าเพื่อออกใบแจ้งหนี้และสื่อสารกับแอปพลิเคชันบันทึกเวลาปฏิบัติงานภายใน โดยอัตโนมัติ RESTful API ให้การสนับสนุนการแลกเปลี่ยนข้อมูลนี้ เพราะ เป็นระบบที่มีมาตรฐานการสื่อสารระหว่างซอฟต์แวร์ที่ปลอดภัย เชื่ิอถือ และมีประสิทธิภาพ

### API (Application Programming Interface)

ส่วนต่อประสานโปรแกรมประยุกต์ (Application Programming Interface หรือ API) กำหนดกฎที่คุณต้องปฏิบัติตามเพื่อสื่อสารกับระบบซอฟต์แวร์อื่น ๆ โดยนักพัฒนาเปิดเผย หรือสร้าง API เพื่อให้แอปพลิเคชันอื่นสามารถสื่อสารกับแอปพลิเคชันได้ทางโปรแกรม ตัวอย่างเช่น แอปพลิเคชันบันทึกเวลาปฏิบัติงานแสดง API ที่ขอข้อมูลของพนักงานและช่วงวันที่ เมื่อได้รับข้อมูลนี้แล้ว ระบบจะประมวลผลบันทึกเวลาปฏิบัติงานของพนักงานเป็นการภายใน และส่งกลับจำนวนชั่วโมงที่ทำงานในช่วงวันที่ดังกล่าว ทั้งนี้คุณสามารถมองได้ว่า API เป็น Gate Way ระหว่าง Client และทรัพยากรบนเว็บ

Client (ไคลเอนต์) คือ ผู้ใช้ที่ต้องการเข้าถึงข้อมูลจากเว็บ โดยไคลเอนต์อาจเป็นบุคคลหรือระบบซอฟต์แวร์ที่ใช้ API ได้ ตัวอย่างเช่น นักพัฒนาสามารถเขียนโปรแกรมที่เข้าถึงข้อมูลสภาพอากาศจากระบบสภาพอากาศ หรือคุณสามารถเข้าถึงข้อมูลเดียวกันจากเบราว์เซอร์เมื่อยืนชมเว็บไซต์รายงานสภาพอากาศได้โดยตรง

Resource (ทรัพยากร) คือ ข้อมูลที่แอปพลิเคชันต่าง ๆ มอบให้แก่ไคลเอนต์ โดยทรัพยากรอาจเป็นรูปภาพ วิดีโอ ข้อความ ตัวเลข หรือข้อมูลประเภทใดก็ได้ ทั้งนี้เครื่องคอมพิวเตอร์ที่มอบทรัพยากรให้แก่ไคลเอนต์นั้นเรียกว่า "เซิร์ฟเวอร์" องค์กรต่าง ๆ ใช้ API เพื่อแบ่งปันทรัพยากรและให้บริการเว็บในขณะที่ยังคงดูแลรักษาความปลอดภัย การควบคุม และการรับรองความถูกต้องไปพร้อมกัน นอกจากนี้ API ยังช่วยให้ลูกค้าระบุได้ว่าไคลเอนต์ได้สามารถเข้าถึงทรัพยากรภายในที่เฉพาะเจาะจงได้

### REST (Representational State Transfer)

REST เป็นสถาปัตยกรรมซอฟต์แวร์ที่กำหนดเงื่อนไขว่า API ควรทำงานอย่างไร โดยแต่แรกเริ่มนั้น มีการสร้าง REST ขึ้นเพื่อเป็นแนวทางในการจัดการการสื่อสารบนเครือข่ายที่ซับซ้อน เช่น อินเทอร์เน็ต สามารถใช้สถาปัตยกรรม REST เพื่อรับการสื่อสารที่มีประสิทธิภาพสูงและเชื่อถือได้ในทุกระดับ ยังสามารถใช้และปรับเปลี่ยนสถาปัตยกรรมได้อย่างง่ายดาย โดยนำความสามารถในการมองเห็นและการเคลื่อนย้ายข้อมูลเพลตฟอร์มมาสู่ระบบ API

นักพัฒนา API สามารถออกแบบ API ได้โดยใช้สถาปัตยกรรมต่าง ๆ โดย API ที่เป็นไปตามรูปแบบสถาปัตยกรรม REST เรียกว่า REST API บริการเว็บที่ใช้สถาปัตยกรรม REST เรียกว่าบริการเว็บ RESTful คำว่า RESTful API โดยทั่วไปหมายถึง API เว็บแบบ RESTful อย่างไรก็ตาม คุณสามารถใช้คำว่า REST API และ RESTful API แทนกันได้

#### 2.1.3 ระบบฐานข้อมูล (Database System)

ระบบฐานข้อมูล (Database System) [3] คือ ระบบที่รวบรวมข้อมูลต่าง ๆ ที่เกี่ยวข้องกันเข้าไว้ด้วยกันอย่างมีระบบ มีความสมมั่นคงระหว่างข้อมูลต่าง ๆ ที่ชัดเจน ในระบบฐานข้อมูลจะประกอบด้วยแฟ้มข้อมูลหลายแฟ้มที่มีข้อมูลเกี่ยวข้องสัมพันธ์กันเข้าไว้ด้วยกันอย่างเป็นระบบและเปิดโอกาสให้ผู้ใช้สามารถใช้งานและดูแล

รักษาป้องกันข้อมูลเหล่านี้ได้อย่างมีประสิทธิภาพ โดยมีซอฟต์แวร์ที่เปรียบเสมือนสื่อกลางระหว่าง ผู้ใช้และ โปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้งานข้อมูล เรียกว่า ระบบจัดการฐานข้อมูล หรือ DBMS (data base management system) มีหน้าที่ช่วยให้ผู้ใช้เข้าถึงข้อมูลได้ง่ายสะดวกและมีประสิทธิภาพ การเข้าถึงข้อมูลของผู้ใช้อาจเป็นการสร้างฐานข้อมูล การแก้ไขฐานข้อมูล หรือ การตั้งค่าตามเพื่อให้ได้ข้อมูลมา โดยผู้ใช้มี จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล

### ประโยชน์ของฐานข้อมูล

#### 1. ลดการเก็บข้อมูลที่ซ้ำซ้อน

ข้อมูลบางชุดที่อยู่ในรูปของแฟ้มข้อมูลอาจมีปรากฏอยู่หลาย ๆ แห่ง เพราะ มีผู้ใช้ข้อมูลชุดนี้หลายคน เมื่อใช้ระบบฐานข้อมูลแล้วจะช่วยให้ความซ้ำซ้อนของข้อมูลลดน้อยลง

#### 2. รักษาความถูกต้องของข้อมูล

เนื่องจากฐานข้อมูลมีเพียงฐานข้อมูลเดียว ในกรณีที่มีข้อมูลชุดเดียวกันปรากฏอยู่หลายแห่งในฐานข้อมูล ข้อมูลเหล่านี้จะต้องตรงกัน ถ้ามีการแก้ไขข้อมูลนี้ทุก ๆ แห่งที่ข้อมูลปรากฏอยู่จะแก้ไขให้ถูกต้อง ตามกันหมดโดยอัตโนมัติด้วยระบบจัดการฐานข้อมูล

#### 3. การป้องกันและรักษาความปลอดภัยให้กับข้อมูลทำได้อย่างสะดวก

การป้องกันและรักษาความปลอดภัยกับข้อมูลระบบฐานข้อมูลจะให้เฉพาะผู้ที่เกี่ยวข้องเท่านั้น ซึ่งก่อ ให้เกิดความปลอดภัย (Security) ของข้อมูลด้วย

## 2.2 ด้านเทคโนโลยี

ในส่วนนี้จะอธิบายถึงเทคโนโลยีที่ใช้ในการพัฒนาเว็บแอปพลิเคชัน

### 2.2.1 HTML

HTML [4] ย่อมาจาก Hyper Text Markup Language คือ ภาษาคอมพิวเตอร์ที่ใช้ในการแสดงผลของ เอกสารบน Website หรือที่เรียกว่า เว็บเพจ ถูกพัฒนาและกำหนดมาตรฐานโดยองค์กร World Wide Web Consortium (W3C) และจากการพัฒนาทางด้าน Software ของ Microsoft ทำให้ภาษา HTML เป็นอีกภาษาหนึ่งที่ใช้เขียนโปรแกรมได้ หรือที่เรียกว่า HTML Application HTML เป็นภาษาประเภท Markup สำหรับการสร้างเว็บเพจ โดยใช้ภาษา HTML สามารถทำโดยใช้โปรแกรม Text Editor ต่าง ๆ เช่น VS Code, Vim หรือ จะอาศัยโปรแกรมที่เป็นเครื่องมือช่วยสร้างเว็บเพจ เช่น Dream Weaver ซึ่ง aba นวยความสะดวกในการสร้างหน้า HTML ส่วนการเรียกใช้งานหรือทดสอบการทำงานของเอกสาร HTML จะใช้โปรแกรม Web Browser เช่น Google Chrome, Microsoft Edge, Mozilla Firefox, Safari และ Opera เป็นต้น



รูปที่ 2.1: HTML

### 2.2.2 CSS

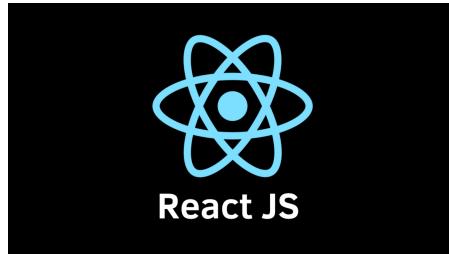
CSS [5] ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า ”สีต่อซึ่ด” คือ ภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ ”Style”) ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่มีข้อผูกพันเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลลัพธ์ของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสาร บ่อยครั้ง หรือ ต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วทั้งทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C

### 2.2.3 TypeScript

TypeScript [6] คือ ภาษา JavaScript ใน Version ที่ได้รับการ Upgrade สามารถทำงานบน Node.js Environment หรือ Web Browser ต่าง ๆ ที่มีการรองรับ ECMAScript 3 ขึ้นไป TypeScript เป็น Statically Compiled Language ที่ได้จัดเตรียมทั้ง Static Typing, Classes และ Interface ไว้ให้แล้ว ช่วยให้คุณสามารถเขียน Code ของ JavaScript ที่เรียบง่ายและ Clean ได้อย่างสะดวกขึ้น ดังนั้น การใช้ TypeScript จะช่วยให้คุณสามารถสร้าง Software ที่ปรับใช้งานได้ง่ายและมีประสิทธิภาพมากยิ่งขึ้น

### 2.2.4 React JS

React JS [7] เป็นไลบรารี JavaScript ที่ใช้สร้าง User Interface (UI) ในเว็บแอปพลิเคชันแบบ Single Page Application (SPA) และเว็บแอปพลิเคชันที่มีการอัปเดตสด ๆ โดย React ช่วยให้ผู้พัฒนาสามารถสร้าง UI ที่มีประสิทธิภาพและเรียบง่ายได้ง่ายขึ้น ด้วยการใช้ Component-based architecture ที่ช่วยให้โค้ดสามารถรับมือกับขนาดของโปรเจกต์ได้ง่ายขึ้น และการใช้ Virtual DOM ที่ช่วยเพิ่มประสิทธิภาพในการเปลี่ยนแปลง UI โดยไม่ต้องทำการอัปเดตทั้งหมดของ DOM



รูปที่ 2.2: React JS



รูปที่ 2.3: Mantine UI

ประโยชน์ของ React JS ได้แก่:

1. ประสิทธิภาพสูง: โดยใช้ Virtual DOM ทำให้การเปลี่ยนแปลง UI มีประสิทธิภาพและเร็วขึ้น
2. Component-based: ช่วยให้การจัดการ UI เป็นเรื่องง่ายและมีระเบียบมากขึ้น
3. Reusable Components: สามารถ Reuse โค้ดของ Component ได้ช่วยในการพัฒนาเร็วขึ้น
4. รองรับการทำงานแบบ Server-side Rendering (SSR) และ Client-side Rendering (CSR): ทำให้เหมาะสมสำหรับการพัฒนาเว็บแอปพลิเคชันที่มี SEO ที่ดีและประสิทธิภาพการโหลดที่ดี

### 2.2.5 Mantine UI

Mantine UI [8] เป็นไลบรารีของ React Component ที่ถูกออกแบบมาเพื่อช่วยในการพัฒนา UI ในโปรเจกต์ของเรา มันมีชุดของ Components ที่หลากหลายและน่าใช้งานอย่างง่าย เหมาะสมสำหรับการสร้าง UI ที่สวยงามและมีประสิทธิภาพ การใช้ Mantine UI ช่วยให้สามารถประสานงานกับผู้ออกแบบและนักพัฒนาได้อย่างมีประสิทธิภาพเพื่อสร้างผลลัพธ์ที่มีคุณภาพและประทับใจได้ในโปรเจกต์ของเรา การนำ Mantine UI เข้ามาใช้ยังช่วยลดเวลาในการพัฒนาโดยทั่วไปด้วยความสามารถในการทดสอบและปรับแต่งที่มีอยู่อย่างสมบูรณ์ โดยทั้งหมดนี้ช่วยให้เราสามารถให้ผลลัพธ์สุดยอดในโปรเจกต์ของเราได้อย่างมีประสิทธิภาพและมีคุณภาพสูงสุดในเวลาที่มีจำกัด

### 2.2.6 PostgreSQL

PostgreSQL [9] เป็นฐานข้อมูลเชิงสัมพันธ์แบบ Open source ระดับธุรกิจที่ทรงพลัง อนุญาตให้ใช้ข้อมูลและแบบสอบถาม SQL เชิงสัมพันธ์และ JSON ที่ไม่ใช้เชิงสัมพันธ์ PostgreSQL มีชุมชนที่แข็งแกร่งอยู่เบื้องหลัง PostgreSQL เป็นระบบจัดการฐานข้อมูลที่น่าเชื่อถือมาก พร้อมการสนับสนุน ความปลอดภัย และความแม่นยำในระดับดีเยี่ยม โทรศัพท์มือถือและเว็บแอปพลิเคชันจำนวนมากใช้ PostgreSQL เป็นฐานข้อมูลเริ่มต้น โซลูชันเชิงพื้นที่และการวิเคราะห์จำนวนมากใช้ประโยชน์จาก PostgreSQL เวอร์ชันล่าสุด คือ



รูปที่ 2.4: PostgreSQL

PostgreSQL 15 PostgreSQL รองรับประเภทข้อมูลที่ซับซ้อน ในความเป็นจริง ฐานข้อมูลถูกสร้างขึ้นโดย คำนึงถึงประเภทข้อมูลจำนวนมาก ประสิทธิภาพของฐานข้อมูลนั้นใกล้เคียงกับของคู่แข่ง เช่น Oracle และ SQL Server AWS ให้บริการฐานข้อมูลที่ได้รับการบำรุงรักษาอย่างสมบูรณ์สำหรับ PostgreSQL ด้วยบริการฐานข้อมูลเชิงสัมพันธ์ของ Amazon PostgreSQL ยังใช้ในการสร้าง Amazon Aurora อีกด้วย

### คุณสมบัติที่สำคัญของ PostgreSQL

หนึ่งในเหตุผลที่ PostgreSQL ได้รับความนิยมมากเนื่องจากชุดคุณลักษณะ ฐานข้อมูลช่วยในการ พัฒนา แอปพลิเคชัน โดยการรักษาความสมบูรณ์ของข้อมูล ช่วยให้ผู้ดูแลระบบสามารถสร้างสภาพแวดล้อมที่ทนต่อ ความผิดพลาดได้ นอกจากนี้ยังสามารถใช้ขั้นแพลตฟอร์มที่หลากหลายและใช้ประโยชน์จากภาษาโปรแกรม ทั่วไปทั้งหมด เราจะเห็นรายชื่อที่แนะนำในภายหลัง ฐานข้อมูลยังมีระบบล็อกขั้นสูงมาก นอกจากนี้ยังมีการ ควบคุมการทำงานพร้อมกันหลายเวอร์ชัน เชิร์ฟเวอร์ฐานข้อมูล PostgreSQL ยังมีฟังก์ชันการทำงานสำหรับ การเขียนโปรแกรมผ่านเชิร์ฟเวอร์สำหรับผู้ใหญ่อีกด้วย เป็นไปตามข้อกำหนด ANSI SQL และรองรับสถา- ปัตยกรรมเครือข่ายโคลอเน็ต-เชิร์ฟเวอร์อย่างสมบูรณ์

PostgreSQL ยังมีความพร้อมใช้งานสูงและเชิร์ฟเวอร์สำรอง ลดความล้าหลังกับ ANSI-SQL2008 และ เชิงวัตถุ ความสามารถในการเชื่อมต่อกับคลังข้อมูลอื่นๆ เช่น NoSQL ซึ่งทำหน้าที่เป็น Hub แบบครบวงจร สำหรับระบบภาษา Python สามารถทำได้ผ่านการสนับสนุน JSON ของฐานข้อมูล ข้อมูลของคลัสเตอร์ฐาน ข้อมูลเดียวจะได้รับการจัดการโดยอินสแตนซ์ PostgreSQL หนึ่งอินสแตนซ์เสมอ คลัสเตอร์ของฐานข้อมูล คือ กลุ่มของ Record ที่เก็บไว้ในที่เดียวกันบนระบบไฟล์

#### 2.2.7 Prisma

Prisma ORM [10] เป็นเครื่องมือช่วยในการจัดการฐานข้อมูลที่เป็น ORM (Object-Relational Mapping) ซึ่งถูกออกแบบมาเพื่อช่วยให้การเข้าถึงฐานข้อมูลเป็นไปอย่างมีประสิทธิภาพและง่ายดายขึ้นสำหรับผู้ พัฒนาซอฟต์แวร์ โดย Prisma ORM ช่วยในการสร้าง query ที่ปลอดภัยและมีประสิทธิภาพอีกด้วย

ประโยชน์ของ Prisma ORM ได้แก่:

1. ความสะดวกและความยืดหยุ่นในการใช้งาน: Prisma ORM ช่วยลดความซับซ้อนในการจัดการฐาน ข้อมูลและทำให้การเขียนโค้ดเป็นเรื่องที่ง่ายขึ้น ด้วยรูปแบบที่เป็นตัวถูกและเรียกใช้ Method ของ Prisma เพื่อสร้างและจัดการข้อมูล
2. การสร้าง Query ที่ปลอดภัย: Prisma ORM ช่วยป้องกันการโจมตีด้านความปลอดภัย เช่น SQL Injection โดยมีการ Validate และ Escape ข้อมูลอัตโนมัติ



รูปที่ 2.5: Prisma ORM

3. การสร้างฐานข้อมูลแบบพลวัต: Prisma ORM ช่วยให้สามารถสร้างและแก้ไขโครงสร้างของฐานข้อมูลได้อย่างง่ายดาย ผ่านการใช้งาน Migration เพื่อเปลี่ยนแปลง Schema ของฐานข้อมูล
4. ประสิทธิภาพและความเร็ว: Prisma ORM มีการจัดการข้อมูลแบบอัตโนมัติที่เป็นไปอย่างมีประสิทธิภาพ ทำให้การเข้าถึงข้อมูลเป็นไปอย่างรวดเร็ว

Prisma ORM ได้ถูกพัฒนาโดย Prisma Labs และเป็นโปรเจกต์ Open source ที่สามารถใช้งานได้ฟรีและเป็นส่วนหนึ่งของชุมชนนักพัฒนาซอฟต์แวร์ สามารถดาวน์โหลดและใช้งาน Prisma ORM ได้จากเว็บไซต์ของเครื่องมือนี้แบบฟรี นอกจากนี้ยังมีเอกสารและคู่มือการใช้งานที่มีอยู่ในชุมชนสำหรับการศึกษาและการใช้งานอื่น ๆ อีกด้วย

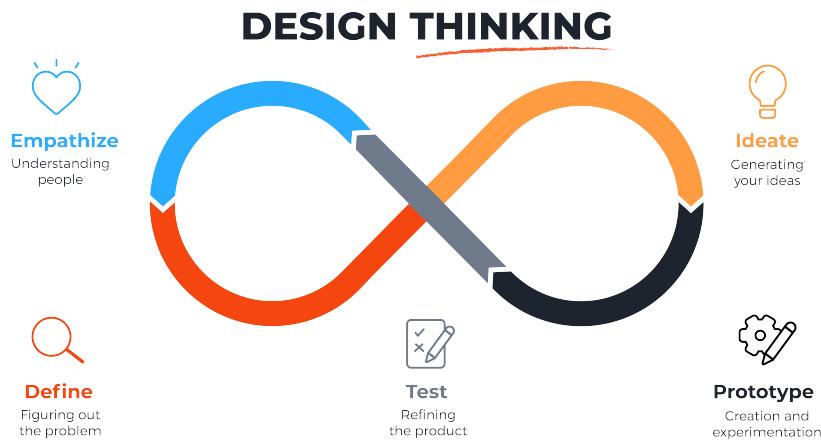
## 2.3 ด้าน User Interface

ในส่วนนี้จะอธิบายถึงการออกแบบ User Interface ของเว็บแอปพลิเคชัน

### 2.3.1 Design Thinking

กระบวนการออกแบบ Design thinking นั้นมีหลากหลายรูปแบบ ทั้งรูปแบบ 3 ขั้น ไปจนถึง 7 ขั้น ทุกรูปแบบมีความคล้ายคลึงมากที่สุด และใช้หลักการเดียวกันที่อ้างอิงจาก Herbert Simon ผู้ชนะรางวัลโนเบลในสาขา The Sciences of the Artificial ในปี 1969 โดยรูปแบบที่นิยมใช้กันมากที่สุด คือ รูปแบบของ Hasso-Plattner Institute of Design at Stanford มีทั้งหมด 5 กระบวนการด้วยกัน ดังนี้

1. Empathise หรือ การเข้าใจปัญหา คือ การทำความเข้าใจกับปัญหา ก่อน ตั้งแต่การเข้าใจผู้ใช้ กลุ่มเป้าหมาย หรือเข้าใจสิ่งที่ต้องการแก้ไขเพื่อทราบทางที่เหมาะสม และดีที่สุดให้ได้ โดยเริ่มต้นจากการเข้าใจคำถา สร้างสมมติฐาน กระตุ้นให้เกิดการใช้ความคิดที่นำไปสู่ความคิด สร้างสรรค์ และวิเคราะห์ปัญหาให้ถ่องแท้ เพื่อหาแนวทางที่ชัดเจน นำไปสู่การแก้ไขปัญหาที่ตรงประเด็น และสร้างผลลัพธ์ที่ดีที่สุด



รูปที่ 2.6: กระบวนการออกแบบ Design Thinking

2. Define หรือ กำหนดปัญหาให้ชัดเจน คือ การเข้าใจความต้องการ ปัญหา และวิเคราะห์ข้อมูลเชิงลึก เพื่อคัดกรองหาปัญหาที่แท้จริง กำหนดหรือปั๊มห้อย่างชัดเจน เพื่อที่จะเป็นแนวทางในการปฏิบัติ และมีทิศทางในการแก้ไขปัญหายอย่างชัดเจน
3. Ideate หรือ ระดมความคิด คือ การนำเสนอแนวคิดต่างๆร่วมกัน ถึงวิธีการแก้ไขปัญหา อย่างไม่มีกรอบจำกัด การระดมความคิดคร่าวๆมุ่งหลากหลาย และมีหลากหลายแนวทางให้ได้มากที่สุด เพื่อให้มีฐานข้อมูลในการนำไปวิเคราะห์และสรุปผล เพื่อนำไปแก้ไขปัญหา โดยไม่จำเป็นต้องเป็นแนวทางใดแนวทางหนึ่ง และการระดมความคิดยังช่วยมองให้เห็นปัญหาที่หลากหลายได้มากขึ้น
4. Prototype หรือ สร้างต้นแบบที่เลือก คือ การออกแบบผลิตภัณฑ์หรือนวัตกรรม เพื่อสร้างต้นแบบสำหรับการทดสอบ และนำไปใช้จริง ซึ่งคือ การลงมือปฏิบัติหรือทดลองตามแนวทางการแก้ไขปัญหาที่ได้กำหนดไว้
5. Test หรือ ทดสอบการแก้ไขปัญหา นำ Prototype ที่จัดทำขึ้นมาไปทดสอบกับผู้ใช้เพื่อว่าสามารถแก้ไขปัญหาของ ผู้ใช้ได้หรือไม่ และหลังจากนั้นถ้าหากการแก้ปัญหายังไม่สามารถช่วยแก้ไขได้ หรือแก้ไขได้ยังไม่ดีพอ ผู้จัดทำจะต้องกลับไปทำตั้งแต่ขั้นตอนแรกอีกรอบ จนกว่าจะสามารถต่อรองแบบโปรแกรมที่แก้ไขปัญหา ของผู้ใช้ได้

## 2.4 ความรู้ตามหลักสูตรชี้แจงถูกนำมาใช้หรือบูรณาการในโครงการ

- 261207 Basic CPE Lab นำความรู้ทางด้านการพัฒนาเว็บแอพพลิเคชัน เช่น HTML, CSS, Tailwind CSS, JavaScript, TypeScript, Next.js และ Node.js มาประยุกต์ใช้ในการพัฒนาเว็บแอพพลิเคชัน ทั้งด้านของ Front-End ซึ่งจะแสดงผลของเว็บไซต์ และ Back-End ที่จะจัดการการทำงานต่าง ๆ รวมถึงการเชื่อมต่อกับฐานข้อมูล
- 261361 Software Engineering การใช้กระบวนการทางวิศวกรรมในการดูแลการผลิต ตั้งแต่การเริ่มเก็บความต้องการ การตั้งเป้าหมายของระบบ การออกแบบ กระบวนการพัฒนา การตรวจสอบ การประเมินผลและทดสอบระบบ

- 261346 Database Systems การใช้งานฐานข้อมูล โดยใช้ PostgreSQL ในการจัดการฐานข้อมูล รวมถึงการเข้ามต่อ กับฐานข้อมูล และการเขียนคำสั่ง SQL ในการดึงข้อมูลจากฐานข้อมูลและการจัดการข้อมูลในฐานข้อมูล
- 261200 Object-Oriented Programming ใช้เป็นพื้นฐานในการเขียนโปรแกรมในการพัฒนาเว็บแอปพลิเคชัน

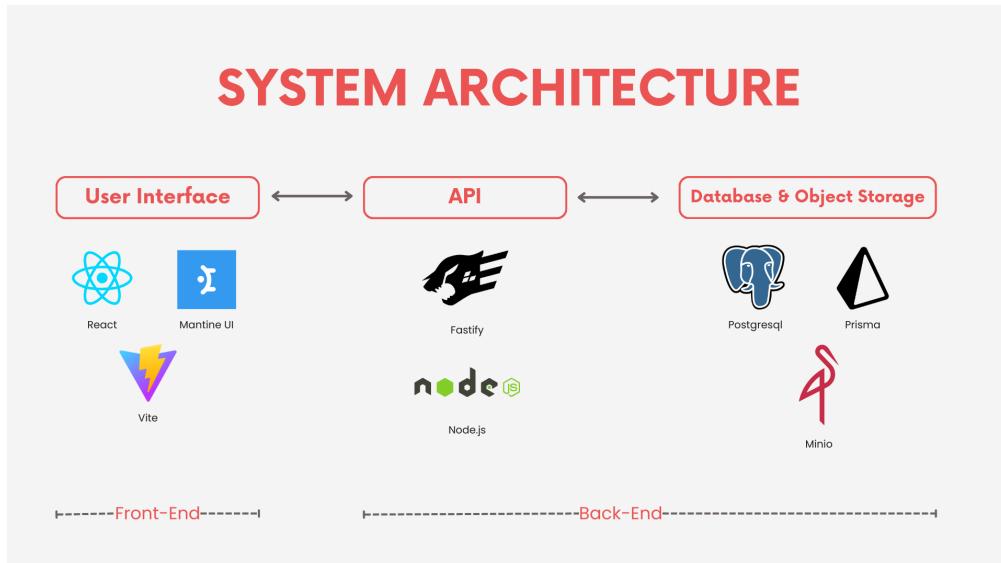
## บทที่ 3

### โครงสร้างและขั้นตอนการทำงาน

ในบทนี้จะกล่าวถึงหลักการ และการออกแบบระบบ

#### 3.1 หลักการทำงานของระบบ

##### 3.1.1 การทำงานของระบบ (System Architecture)



รูปที่ 3.1: System Architecture

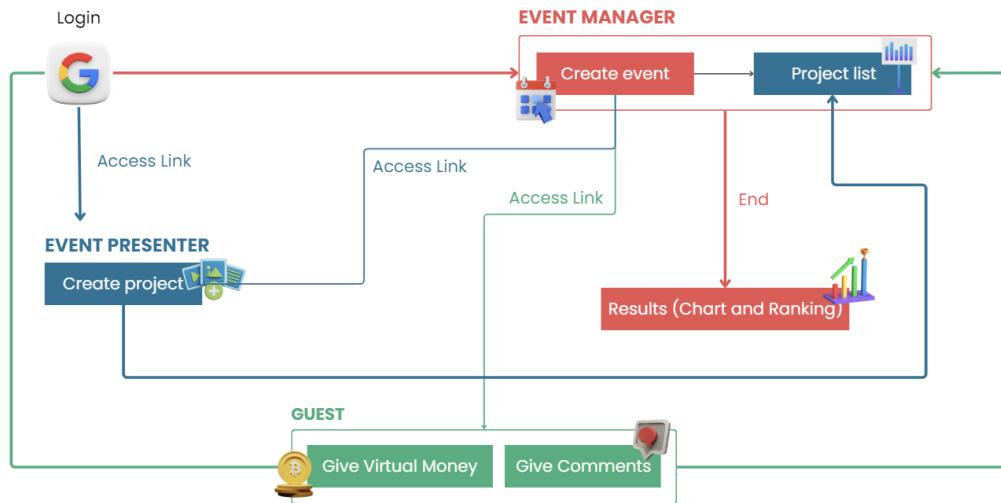
จากรูปที่ 3.1 จะเป็นภาพรวมของระบบที่เราได้ทำการออกแบบขึ้น โดยมีรายละเอียดดังนี้

##### • Frontend

ส่วนหน้าบ้าน (Frontend) เป็นส่วนการพัฒนาเพื่อแสดง User Interface (UI) โดยโครงการนี้ได้ใช้เทคโนโลยี React ร่วมกับ Mantine UI ในการออกแบบและสร้างส่วนติดต่อผู้ใช้ (UI) โดยมีหน้าที่แสดงผลลัพธ์ต่อผู้ใช้ในรูปแบบที่เข้าใจง่าย รับข้อมูลป้อนจากผู้ใช้ผ่านอินเทอร์เฟชต่าง ๆ เช่น ปุ่มฟิลด์ข้อความ ฯลฯ และสื่อสารกับ API เพื่อส่งคำร้องขอและรับผลลัพธ์

##### • Backend

ส่วนหลังบ้าน (Backend) เป็นส่วนที่ทำหน้าที่รับข้อมูลจากผู้ใช้ จากนั้นทำการประมวลผลข้อมูล และส่งข้อมูลลับไปยังผู้ใช้ โดยโครงการนี้ได้ใช้เทคโนโลยี Fastify เว็บเฟรมเวิร์ค Node.js ร่วมกับ Prisma ORM ในการจัดการข้อมูลในฐานข้อมูล รวมถึงทำการสื่อสารกับฐานข้อมูล PostgreSQL และ Minio (Object Storage) ในการจัดการข้อมูลที่เป็นไฟล์



รูปที่ 3.2: User Flow

### 3.1.2 เส้นทางของผู้ใช้ (User Flow)

จากรูปที่ 3.2 จะเป็นเส้นทางของผู้ใช้งานที่เข้าใช้งานระบบ โดยมีรายละเอียดดังนี้

#### ผู้สร้างงานกิจกรรม (Event Manager)

- ผู้ใช้เข้าสู่ระบบโดยใช้ Google Account
- ผู้ใช้สร้าง Event ใหม่ โดยป้อนข้อมูลต่าง ๆ เช่น ชื่องานกิจกรรม รายละเอียด วันเวลา สถานที่ เป็นต้น
- ผู้ใช้สร้าง Event เสร็จสิ้น จะมี QR Code หรือ Access Link สำหรับนำไปให้ผู้นำเสนองานกิจกรรม (Presenter) สำหรับเพิ่มโครงการและผู้เข้าร่วมกิจกรรม (Guest) สำหรับเข้าร่วมกิจกรรม
- ผู้ใช้สามารถดูผลลัพธ์ (Chart และ Ranking) ของโครงการต่าง ๆ ที่เข้าร่วมกิจกรรมในระหว่างจัดงานกิจกรรม

#### ผู้นำเสนองานกิจกรรม (Presenter)

- ผู้ใช้เข้าสู่ระบบโดยใช้ Google Account
- ผู้ใช้สร้าง Project ใหม่ โดยป้อนข้อมูลต่าง ๆ เช่น ชื่อโครงการ รายละเอียด รูปภาพ ลิงก์ และไฟล์ อื่น ๆ ที่เกี่ยวข้อง
- ผู้ใช้ดูผลลัพธ์ว่าโครงการได้รับ Virtual Money จากผู้เข้าร่วมกิจกรรม (Guest) และความคิดเห็นเกี่ยวกับโครงการหลังจากเสร็จสิ้นงานกิจกรรม

#### ผู้เข้าร่วมกิจกรรม (Guest)

- ผู้ใช้เข้าร่วม Event โดยใช้ QR Code หรือ Access Link ที่ได้รับจาก Event Manager
- ผู้ใช้เข้าสู่ระบบโดยใช้ Google Account

- ผู้ใช้ดูรายละเอียดของงานกิจกรรมที่จัดขึ้น รวมถึงโครงการที่เข้าร่วมกิจกรรม
- ผู้ใช้สามารถให้ Virtual Money และแสดงความคิดเห็นเกี่ยวกับ Project ต่าง ๆ ที่เข้าร่วมกิจกรรมได้

### 3.1.3 โครงสร้างฐานข้อมูล (Database Schema)



รูปที่ 3.3: Database Schema

จากรูปที่ 3.3 จะเป็นโครงสร้างของฐานข้อมูลที่ใช้ในระบบ โดยมีทั้งหมด 9 ตาราง ดังนี้

- **Users** ใช้เก็บข้อมูลผู้ใช้งาน โดยจะเก็บข้อมูลผู้ที่เป็นผู้จัดงานกิจกรรมและผู้นำเสนอโครงการ ซึ่งผู้ใช้งาน 1 คนสามารถเป็นได้ทั้งผู้จัดงานกิจกรรมและผู้นำเสนอโครงการ
- **Events** ใช้เก็บข้อมูลงานกิจกรรม โดยจะเก็บข้อมูลเช่น ชื่องานกิจกรรม รายละเอียด วันเวลา สถานที่ และข้อมูลอื่น ๆ ที่เกี่ยวข้อง ซึ่ง 1 ผู้ใช้งานสามารถสร้างงานกิจกรรมได้หลายงานกิจกรรม
- **Projects** ใช้เก็บข้อมูลโครงการ โดยจะเก็บข้อมูลเช่น ชื่อโครงการ รายละเอียด รูปภาพ ลิงก์ และไฟล์อื่น ๆ ที่เกี่ยวข้อง ซึ่ง 1 ผู้ใช้งานสามารถสร้างโครงการได้หลายโครงการ
- **Virtual Money** ใช้เก็บข้อมูลเงินเสมือนที่โครงการได้รับ ซึ่ง 1 โครงการสามารถมีเงินเสมือนได้หลายรายการจากผู้เข้าร่วมงาน
- **Guests** ใช้เก็บข้อมูลของผู้เข้าร่วมงาน ซึ่งจะเก็บข้อมูลทั่วไป และเก็บจำนวนเงินเสมือนที่มีอยู่ สำหรับให้โครงการต่าง ๆ
- **Thumbnails** ใช้เก็บข้อมูลรูปภาพ Thumbnail ของงานกิจกรรม โดย 1 งานกิจกรรมสามารถมี Thumbnail ได้เพียง 1 รูป

- **Documents** ใช้เก็บข้อมูลไฟล์เอกสารที่เกี่ยวข้องกับโครงการ ซึ่ง 1 โครงการสามารถมีไฟล์เอกสารได้หลายไฟล์
- **Comments** ใช้เก็บข้อมูลความคิดเห็นของแขกผู้เข้าร่วมงานที่มีต่อโครงการ ซึ่ง 1 โครงการสามารถมีความคิดเห็นได้หลายความคิดเห็น
- **Project Images** ใช้เก็บข้อมูลรูปภาพที่เกี่ยวข้องกับโครงการ ซึ่ง 1 โครงการสามารถมีรูปภาพได้หลายรูปภาพ

ชื่อ colummn	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสผู้ใช้งาน
first_name_th	VARCHAR	ชื่อภาษาไทย
last_name_th	VARCHAR	นามสกุลภาษาไทย
first_name_en	VARCHAR	ชื่อภาษาอังกฤษ
last_name_en	VARCHAR	นามสกุลภาษาอังกฤษ
email	VARCHAR	อีเมล
affiliation	TEXT	สังกัด
profile_pic	TEXT	รูปภาพโปรไฟล์
role	VARCHAR	บทบาทของผู้ใช้งาน
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

ตารางที่ 3.1: ตารางข้อมูลผู้ใช้งาน (Users)

ชื่อ colummn	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสงานกิจกรรม
event_name	VARCHAR	ชื่องานกิจกรรม
start_date	DATE	วันที่เริ่มงานกิจกรรม
end_date	DATE	วันที่สิ้นสุดงานกิจกรรม
description	TEXT	รายละเอียดงานกิจกรรม
submit_start	DATE	วันที่เริ่มรับส่งโครงการ
submit_end	DATE	วันที่สิ้นสุดรับส่งโครงการ
number_of_member	INTEGER	จำนวนสมาชิกของโครงการ
virtual_money	INTEGER	จำนวนเงินเสมือนที่จะให้แขกผู้เข้าร่วมงาน (Guest)
unit_money	VARCHAR	หน่วยเงินเสมือนที่จะให้แขกผู้เข้าร่วมงาน (Guest)
published	BOOLEAN	สถานะการเผยแพร่องานกิจกรรม
organization	TEXT	สังกัดของงานกิจกรรม
video_link	TEXT	ลิงก์วิดีโอที่เกี่ยวข้องกับงานกิจกรรม
user_id	INTEGER	รหัสผู้ใช้งานที่สร้างงานกิจกรรม
location	TEXT	สถานที่จัดงานกิจกรรม
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

ตารางที่ 3.2: ตารางข้อมูลงานกิจกรรม (Events)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสโครงการ
<b>title</b>	<b>VARCHAR</b>	ชื่อโครงการ
<b>description</b>	<b>TEXT</b>	รายละเอียดของโครงการ
<b>user_id</b>	<b>INTEGER</b>	รหัสผู้ใช้งานที่สร้างโครงการ
<b>event_id</b>	<b>INTEGER</b>	รหัสงานกิจกรรมที่โครงการเข้าร่วม
<b>created_at</b>	<b>TIMESTAMP</b>	วันที่สร้างข้อมูล

ตารางที่ 3.3: ตารางข้อมูลโครงการ (Projects)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสโครงการที่ได้รับเงิน
<b>amount</b>	<b>INTEGER</b>	จำนวนเงินเสมือนที่ได้รับ
<b>project_id</b>	<b>INTEGER</b>	รหัสโครงการที่ได้รับเงิน
<b>event_id</b>	<b>INTEGER</b>	รหัสงานกิจกรรมที่ได้รับเงิน
<b>guest_id</b>	<b>INTEGER</b>	รหัสผู้เข้าร่วมกิจกรรมที่ได้รับเงิน
<b>created_at</b>	<b>TIMESTAMP</b>	วันที่สร้างข้อมูล
<b>updated_at</b>	<b>TIMESTAMP</b>	วันที่อัปเดตข้อมูล

ตารางที่ 3.4: ตารางข้อมูลเงินเสมือน (Virtual Money)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสผู้เข้าร่วมกิจกรรม
<b>first_name_th</b>	<b>VARCHAR</b>	ชื่อจริงภาษาไทย
<b>last_name_th</b>	<b>VARCHAR</b>	นามสกุลภาษาไทย
<b>first_name_en</b>	<b>VARCHAR</b>	ชื่อจริงภาษาอังกฤษ
<b>last_name_en</b>	<b>VARCHAR</b>	นามสกุลภาษาอังกฤษ
<b>email</b>	<b>TEXT</b>	ที่อยู่อีเมล
<b>profile_pic</b>	<b>TEXT</b>	รูปภาพโปรไฟล์
<b>virtual_money</b>	<b>INTEGER</b>	จำนวนเงินเสมือนที่มีอยู่
<b>created_at</b>	<b>TIMESTAMP</b>	วันที่สร้างข้อมูล

ตารางที่ 3.5: ตารางข้อมูลแขกผู้เข้าร่วมกิจกรรม (Guests)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสภาพตัวอย่างโครงการ
<b>event_id</b>	<b>INTEGER</b>	รหัสงานกิจกรรมที่ได้รับเงิน
<b>thumbnail</b>	<b>TEXT</b>	รูปภาพตัวอย่างโครงการ
<b>thumbnail_url</b>	<b>TEXT</b>	ลิงก์รูปภาพตัวอย่างโครงการ
<b>created_at</b>	<b>TIMESTAMP</b>	วันที่สร้างข้อมูล
<b>updated_at</b>	<b>TIMESTAMP</b>	วันที่อัปเดตข้อมูล

ตารางที่ 3.6: ตารางข้อมูลภาพตัวอย่างโครงการ (Thumbnails)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสเอกสาร
<u>project_id</u>	INTEGER	รหัสโครงการที่เกี่ยวข้องกับเอกสาร
document_name	TEXT	ชื่อเอกสาร
document_url	TEXT	ลิงก์เอกสาร
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

ตารางที่ 3.7: ตารางข้อมูลเอกสารที่เกี่ยวข้องกับงานกิจกรรม (Documents)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสความคิดเห็น
<u>project_id</u>	INTEGER	รหัสโครงการที่เกี่ยวข้องกับความคิดเห็น
comment_better	TEXT	ความคิดเห็นที่ดีของโครงการ
comment_idea	TEXT	ความคิดเห็นที่เสนอไอเดียใหม่
comment_ilike	TEXT	ความคิดเห็นที่ชอบของโครงการ
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

ตารางที่ 3.8: ตารางข้อมูลความคิดเห็นของผู้เข้าร่วมกิจกรรม (Comments)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
<b>id</b>	<b>INTEGER</b>	รหัสรูปภาพที่เกี่ยวข้องกับโครงการ
<u>project_id</u>	INTEGER	รหัสโครงการที่เกี่ยวข้องกับรูปภาพ
project_image	TEXT	รูปภาพโครงการ
project_image_url	TEXT	ลิงก์รูปภาพโครงการ
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

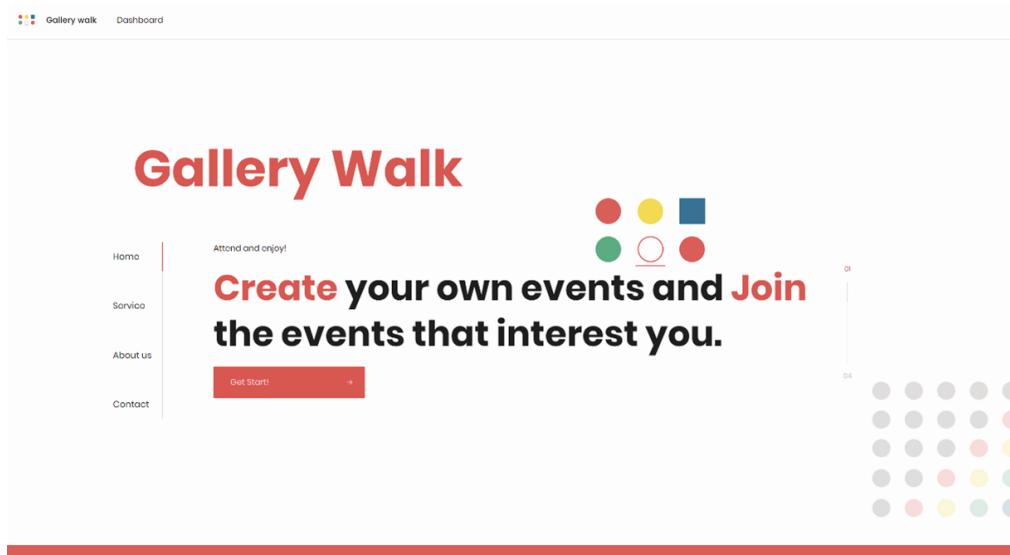
ตารางที่ 3.9: ตารางข้อมูลรูปภาพที่เกี่ยวข้องกับโครงการ (Project Images)

### 3.2 ส่วนเชื่อมต่อระหว่างผู้ใช้งานกับระบบ (User Interface)

โดยแบ่งเป็น 3 ส่วนผู้ใช้งาน คือ ผู้สร้างงานกิจกรรม (Event manager), ผู้นำเสนอผลงาน (Presenter), ผู้เข้าร่วมงาน (Guest)

#### 3.2.1 หน้าเริ่มต้นการใช้งาน (Homepage)

โดยแบ่ง Tab เป็น Home, Service, About us และ Contact สำหรับ Event manager และ Presenter



รูปที่ 3.4: หน้า Homepage



รูปที่ 3.5: หน้า Service

Gallery walk Dashboard

# Gallery Walk

This website was created as a project for course 201492 by a group of fourth-year students from the Department of Computer Engineering, Faculty of Engineering, Chiang Mai University.

- Home
- Service
- About us
- Contact



**Mr. Panthon Kansop**  
Project Manager and Back-end Developer



**Miss Nuttawan Reobreong**  
UX/UI Design and Front-end Developer



**Mr. Yanatib Bhoonsawang**  
Front-end Developer

01
02
03
04


รูปที่ 3.6: หน้า About us

Gallery walk Dashboard

# Gallery Walk

- Home
- Service
- About us
- Contact

**Contact information**

✉ Email: gallerywalk@gmail.com  
☎ Phone: +66 88-5294254  
📍 Address: 4th floor 30th Year Building, Faculty of Engineering, Chiang Mai University  
⌚ Working hours: 8 a.m. - 11 p.m.

**Get in touch**

Email \*   
Name \*   
Message \*

Submit

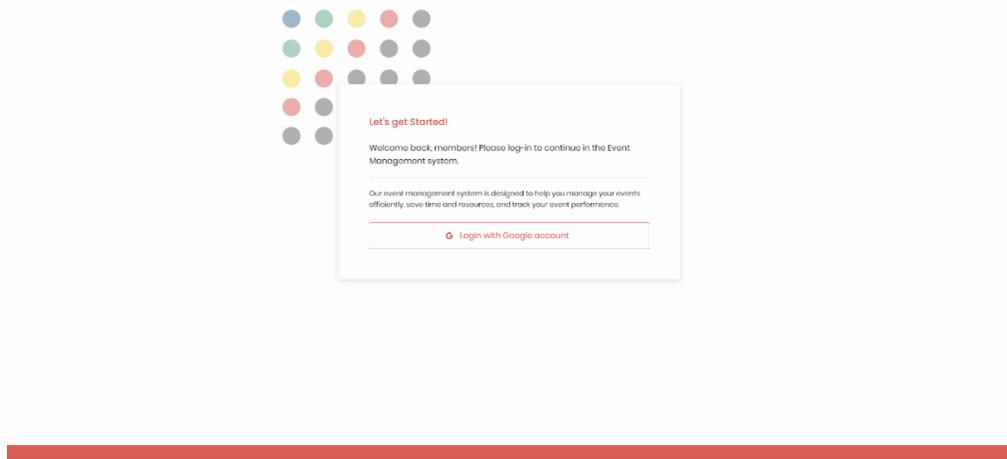
01
02
03
04


รูปที่ 3.7: หน้า Contact

จากรูป 3.4 3.5 3.6 และ 3.7 จะเป็นหน้าเริ่มต้นการใช้งานของระบบ โดยแบ่ง Tab เป็น Home, Service, About us และ Contact สำหรับ Event manager และ Presenter

### 3.2.2 หน้าเข้าสู่ระบบ (Login)

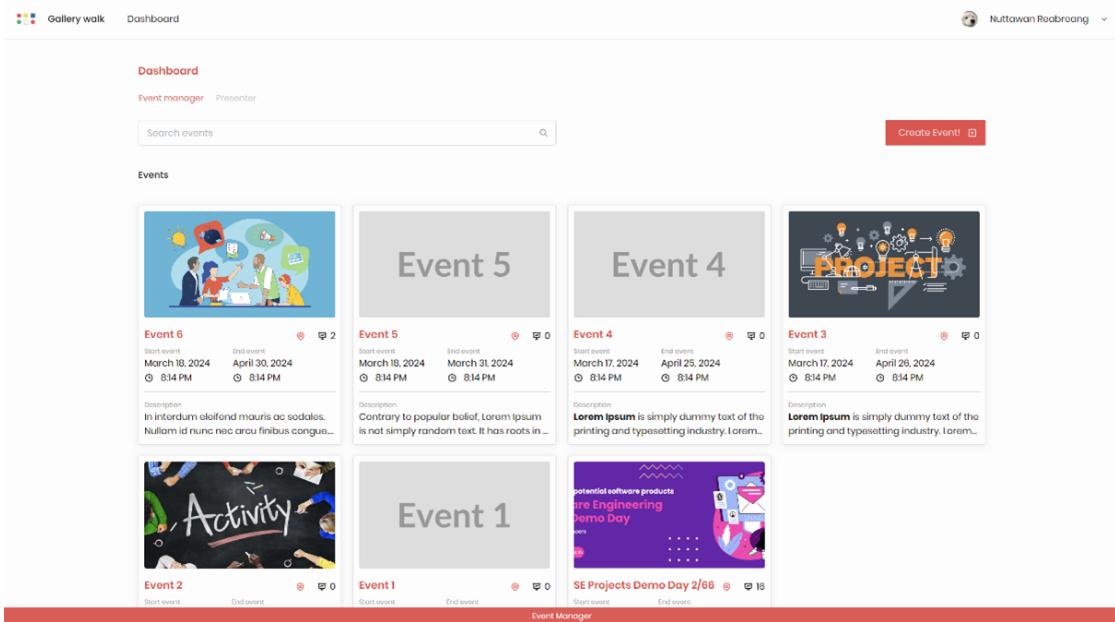
Login ด้วย Google account สำหรับ Event manager และ Presenter



รูปที่ 3.8: หน้า Login สำหรับ Event manager และ Presenter

จากรูป 3.8 จะเป็นหน้าเข้าสู่ระบบ (Login) โดยใช้ Google Account สำหรับ Event manager และ Presenter

### 3.2.3 ผู้สร้างงานกิจกรรม (Event manager)



The screenshot shows a dashboard titled "Dashboard" with tabs for "Event manager" and "Presenter". A search bar and a "Create Event!" button are at the top right. Below is a section titled "Events" displaying six event cards:

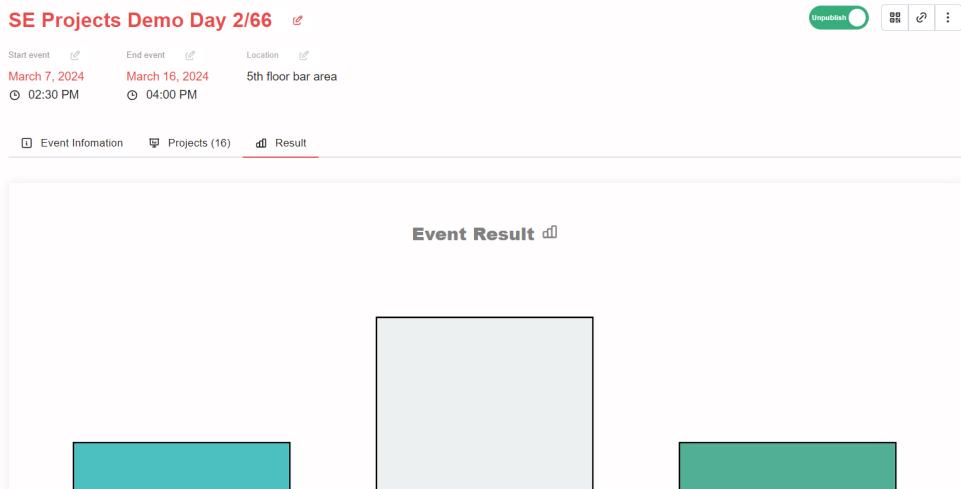
- Event 6**: An illustration of people at a table. Details: Start event March 18, 2024 at 8:14 PM, End event April 30, 2024 at 8:14 PM, 0 attendees, 2 sessions. Description: In intordum oleifend mauris ac sodales. Nullam id nunc nec arcu finibus congue...
- Event 5**: A grey card. Details: Start event March 18, 2024 at 8:14 PM, End event March 31, 2024 at 8:14 PM, 0 attendees, 0 sessions. Description: Contrary to popular belief, Lorem ipsum is not simply random text. It has roots in...
- Event 4**: A grey card. Details: Start event March 17, 2024 at 8:14 PM, End event April 25, 2024 at 8:14 PM, 0 attendees, 0 sessions. Description: Lorem ipsum is simply dummy text of the printing and typesetting industry. Lorem...
- Event 3**: An illustration of a project board with icons. Details: Start event March 17, 2024 at 8:14 PM, End event April 26, 2024 at 8:14 PM, 0 attendees, 0 sessions. Description: Lorem ipsum is simply dummy text of the printing and typesetting industry. Lorem...
- Event 2**: An illustration of people writing on a chalkboard with "Activity" written on it. Details: Start event End event, 0 attendees, 0 sessions.
- Event 1**: A grey card. Details: Start event End event, 0 attendees, 0 sessions. Description: potential software products Software Engineering Demo Day

รูปที่ 3.9: หน้าแสดงกิจกรรมที่สร้าง (Event manager dashboard)

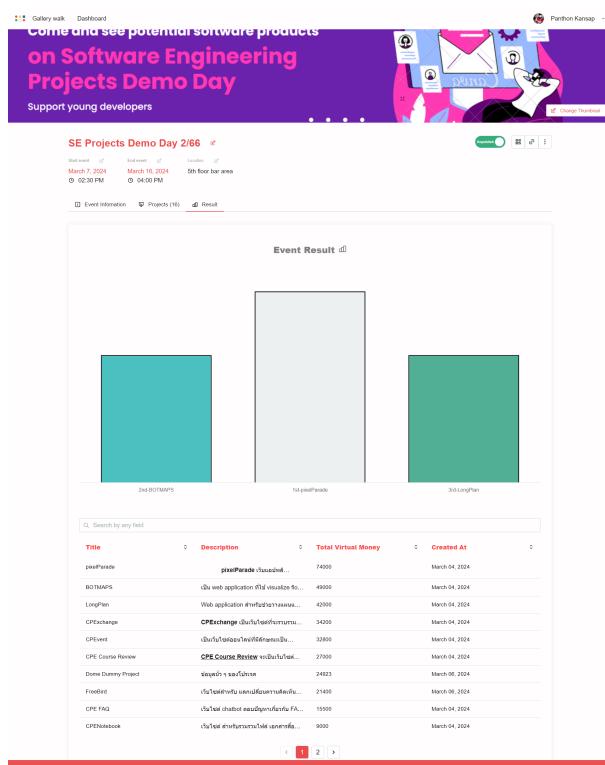
จากรูป 3.9 จะเป็นหน้าแสดงกิจกรรมที่สร้าง (Event manager dashboard) ในรูปแบบการ์ดที่จะแสดงรายละเอียดของงานกิจกรรม ได้แก่ ชื่องานกิจกรรม วันเวลาที่เริ่มงานกิจกรรม วันเวลาที่จบงานกิจกรรม คำอธิบายงานกิจกรรมและจำนวนโครงการในงานกิจกรรมนั้น ๆ โดยหน้านี้ยังแบ่ง Tab เป็น Event Manager และ Presenter ในส่วน Presenter จะแสดงผลโครงการที่เคยสร้างไว้ และจากหน้านี้ผู้สร้างงานกิจกรรมสามารถสร้างงานกิจกรรมใหม่ได้

รูปที่ 3.10: หน้าแสดงข้อมูลรายละเอียดของกิจกรรมที่สร้าง โดยแบ่ง Tab เป็น Event information, Project และ Result

รูปที่ 3.11: หน้าแสดงรายการโครงการที่เข้าร่วมกิจกรรม



รูปที่ 3.12: หน้าแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม โดยแสดงเป็น Ranking และตาราง Virtual Money



รูปที่ 3.13: ตารางแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม

จากรูป 3.10 3.11 3.12 และ 3.13 จะเป็นหน้าแสดงข้อมูลรายละเอียดของกิจกรรมที่สร้าง โดยแบ่ง Tab เป็น Event information ซึ่งจะแสดงข้อมูลรายละเอียดของงานกิจกรรม, Project ซึ่งจะแสดงรายการโครงการที่เข้าร่วมกิจกรรมและ Result ซึ่งจะแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม จำนวน Virtual Money ที่ได้รับและ Ranking ของโครงการ

## สร้างกิจกรรม (Create Event)

The screenshot shows the first step of an event creation process. At the top, there are three tabs: 'Gallery walk' (highlighted in blue), 'Dashboard', and 'Nutrawon Reabreang'. Below the tabs, a progress bar indicates three steps: 'First step General Information' (step 1, highlighted in red), 'Second step Images Dropzone' (step 2), and 'Final step Check all information before submit' (step 3). The main content area is titled 'Event Manager Details'. It includes fields for 'Event Name' (with placeholder 'Event Name'), 'Start Date' (set to 'March 10, 2024'), 'Start Time' (set to '10:00 AM'), 'End Date' (set to 'March 10, 2024'), 'End Time' (set to '11:00 AM'), 'Description' (with a rich text editor placeholder 'Description'), and 'Location' (with a text input placeholder 'Location'). Below this section, another 'Presenter Details' section is partially visible, showing similar date and time fields.

รูปที่ 3.14: หน้าสร้างกิจกรรม First step (Event information)

The screenshot shows the second step of the event creation process. The progress bar at the top shows 'First step General Information' (step 1, highlighted in red), 'Second step Images Dropzone' (step 2), and 'Final step Check all information before submit' (step 3). The main content area features a large dashed rectangular area labeled 'Drag image here or click to select file'. Below this area is a red button labeled 'Select files'. At the bottom right, there are 'Back' and 'Next step' buttons.

รูปที่ 3.15: หน้าสร้างกิจกรรม Second step (Images Dropzone)

 First step  
 General Information -  Second step  
 Images Dropzone -  Final step  
 Check all information before submit

March 2024

Mo	Tu	We	Th	Fr	Sa	Su
26	27	28	29	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Event Manager Details

---

Event Name: dd

Start event	End event	Location
March 21, 2024	March 28, 2024	
10:16 PM	11:16 PM	

Description

\* \*

Presenter Details

---

Start submission: March 21, 2024 @ 21:03 PM

End submission: March 23, 2024 @ 22:03 PM

Guest Details

---

Virtual money: 10000

Unit: Unit

← Back
Submit

รูปที่ 3.16: หน้าสร้างกิจกรรม Final step (Check all information before submit)

จากรูป 3.14 3.15 และ 3.16 จะเป็นหน้าสร้างกิจกรรม โดยแบ่งเป็น 3 ขั้นตอน คือ First step (Event information) ซึ่งจะให้กรอกข้อมูลรายละเอียดของงานกิจกรรม, Second step (Images Dropzone) ซึ่งจะให้เพิ่มรูปภาพตัวอย่างของโครงการ และ Final step (Check all information before submit) ซึ่งจะแสดงข้อมูลทั้งหมดก่อนที่จะสร้างงานกิจกรรม

### 3.2.4 ผู้นำเสนองาน (Presenter)

หลังจากที่เข้าสู่งานกิจกรรมจาก QR Code หรือ Access Link ที่ได้รับจาก Event manager จะเข้าสู่หน้า เข้าสู่ระบบ (Login) และหลังจากที่เข้าสู่ระบบจะเข้าสู่หน้าแสดงข้อมูลรายละเอียดของงานกิจกรรมที่จัดขึ้น และผู้นำเสนอโครงการสามารถเพิ่มโครงการเข้าร่วมงานกิจกรรมได้

The screenshot shows a modal window titled "Create New Project". It contains the following fields:

- Title \***: A text input field labeled "Project Title".
- Description**: A rich text editor with various formatting options like bold, italic, underline, etc.
- Upload files**: A file upload input field labeled "Upload files".
- Upload Image**: A file upload input field labeled "Upload Image".
- Submit**: A red button at the bottom right.

รูปที่ 3.17: หน้าเพิ่มโครงการที่เข้าร่วมกิจกรรม (Add project)

The screenshot shows the "Dashboard" page for the "Presenter" role. It features a search bar and a list of "Projects (2)".

- Project 2**: Status: EVENT ONGOING. Created on: March 18, 2024 at 14:47. Description: Curabitur suscipit congue vulputate. Morbi in justo dui. Ut blandit orci a nequ...
- Project 1**: Status: EVENT ONGOING. Created on: March 18, 2024 at 14:46. Description: In intordum eloifend mauris ac sodales. Nullam id nunc nec arcu finibus congue...

Pagination controls are at the bottom right.

รูปที่ 3.18: หน้าแสดงโครงการที่สร้าง (Presenter dashboard)

จากรูป 3.17 และ 3.18 จะเป็นหน้าเพิ่มโครงการที่เข้าร่วมกิจกรรม (Add project) และ หน้าแสดงโครงการที่สร้าง (Presenter dashboard)

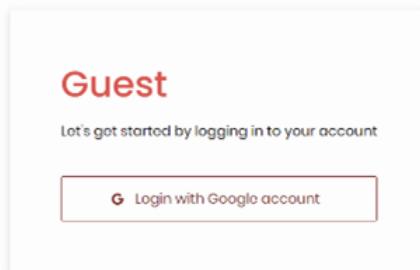
The screenshot shows a digital platform interface for 'Project1'. At the top, there are navigation links for 'Gallery walk' and 'Dashboard', and a user profile for 'Nuttawan Redbreang'. Below the header, the project title 'Project1' is displayed along with start and end dates: 'March 18, 2024 10:03 AM' and 'April 28, 2024 10:04 AM'. A 'Description' section contains placeholder text about an event at Chiang Mai University. The 'Guests' section shows '1234 Number of guests'. The 'Virtual money' section indicates 'Available on 04/20/2024 at 10:00 AM - In a month Number of virtual money'. Below these sections are three images: 'Activity' (people writing on a chalkboard), 'Post-it notes' (a wall covered in colorful sticky notes), and 'PROJECT' (a graphic with gears, lightbulbs, and a computer monitor). There are also 'Documents' and 'Presentations' tabs.

รูปที่ 3.19: หน้าแสดงข้อมูลรายละเอียดของผลงานที่สร้างในกิจกรรม

จากรูป 3.19 จะเป็นหน้าแสดงข้อมูลรายละเอียดของผลงานที่สร้างในกิจกรรม ได้แก่ ชื่อโครงการ, รายละเอียดโครงการ, รูปภาพโครงการ, เอกสารที่เกี่ยวข้องกับโครงการและจำนวน Virtual Money ที่ได้รับ และความคิดเห็นเกี่ยวกับโครงการ

### 3.2.5 ผู้เข้าร่วมงาน (Guest)

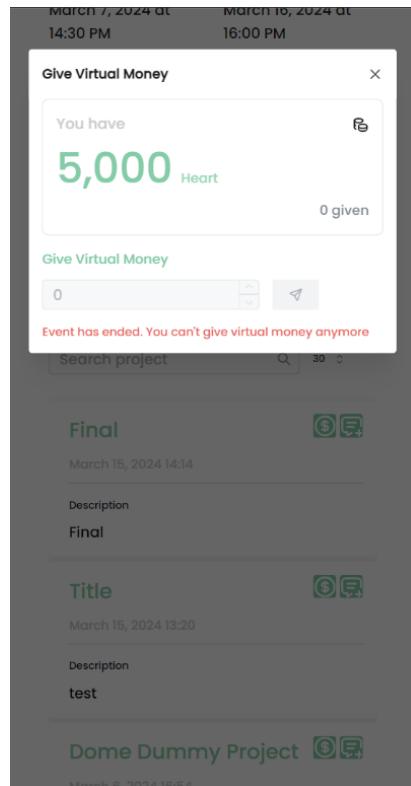
เมื่อเข้าสู่งานกิจกรรมจาก QR Code หรือ Access Link ที่ได้รับจาก Event manager จะเข้าสู่หน้าเข้าสู่ระบบ (Login) และหลังจากที่เข้าสู่ระบบจะเข้าสู่หน้าแสดงข้อมูลรายละเอียดของงานกิจกรรมที่จัดขึ้น และโครงการที่เข้าร่วมกิจกรรม



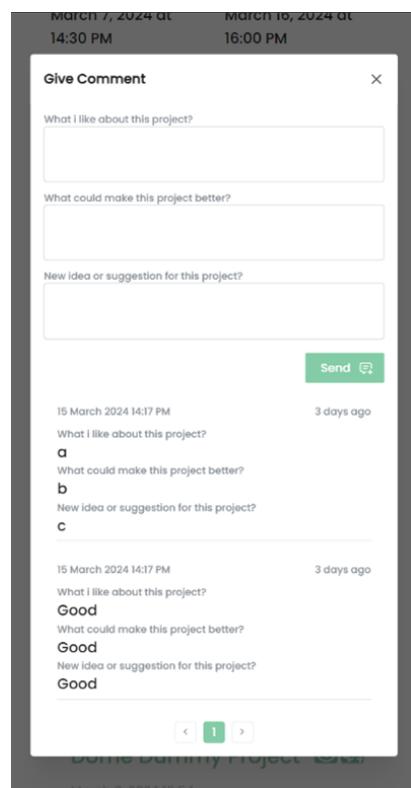
รูปที่ 3.20: หน้าเข้าสู่ระบบ (Login) สำหรับผู้เข้าร่วมกิจกรรม (Guest)

A screenshot of a mobile application interface for the 'SE Projects Demo Day'. At the top, there's a purple banner with text about supporting young developers. Below it, the title 'SE Projects Demo Day' and '2/66' are displayed. The main content area shows event details: 'Start of event' on March 7, 2024 at 14:30 PM, and 'End of event' on March 16, 2024 at 16:00 PM. A box labeled 'Virtual money' contains '5,000 Heart' and 'Your Virtual Money'. A 'Description' section includes a link to 'SE Projects Demo Day 2/66'. A 'Projects (16)' section shows a search bar and a list item 'Final' with a timestamp of 'March 15, 2024 14:14'.

รูปที่ 3.21: หน้าแสดงข้อมูลรายละเอียดของงานกิจกรรมที่จัดขึ้น และโครงการที่เข้าร่วมกิจกรรม



รูปที่ 3.22: หน้าให้ Virtual Money และแสดงความคิดเห็นเกี่ยวกับโครงการที่เข้าร่วมกิจกรรม



รูปที่ 3.23: หน้าแสดงความคิดเห็นเกี่ยวกับโครงการที่เข้าร่วมกิจกรรม

## บทที่ 4

### การทดลองและผลลัพธ์

#### 4.1 การประเมินความพึงพอใจของผู้ใช้

เนื่องจากการพัฒนาระบบนี้ พัฒนาเพื่อความต้องการจากผู้ใช้ที่เป็นผู้สร้างงานกิจกรรม (Event Manager) ผู้นำเสนอโครงการ (Presenter) และแขกผู้เข้าร่วมงาน (Guest) ซึ่งเป็นผู้ใช้ที่สำคัญของระบบ จึงได้ทำการประเมินว่าระบบซึ่งเป็นเว็บ-แอปพลิเคชันนั้นตรงตามความต้องการของผู้ใช้หรือไม่ โดยการทดสอบกับผู้ใช้งานจริง โดยการให้ผู้ใช้ทดลองใช้งานจริง และให้คำแนะนำเพื่อปรับปรุงระบบให้ตรงตามความต้องการของผู้ใช้ ทั้ง 3 กลุ่มผู้ใช้งาน

โดยคณิตน้ำหนึ่งเว็บแอปพลิเคชันนี้ ไปทดลองใช้จริงในกระบวนการวิชา Software Engineering ในวัน Project Demo Day 2/2566 โดยผู้ใช้ ดังนี้

- ผู้ใช้ที่เป็นผู้สร้างงานกิจกรรม (Event Manager) ได้ทดลองใช้งานระบบโดยการสร้างงานกิจกรรม และจัดการงานกิจกรรม (1 คน)
- ผู้ใช้ที่เป็นผู้นำเสนอโครงการ (Presenter) ได้ทดลองใช้งานระบบโดยการสร้างโครงการ และจัดการโครงการ (14 โครงการ)
- ผู้ใช้ที่เป็นแขกผู้เข้าร่วมงาน (Guest) ได้ทดลองใช้งานระบบโดยการเข้าร่วมงาน ให้ Virtual Money และแสดงความคิดเห็นในโครงการต่าง ๆ (5 คน)

#### 4.2 ผลการประเมินความพึงพอใจของผู้ใช้

โดยการให้กลุ่มผู้ใช้ตอบแบบสำรวจผ่าน Google Form ประเมินความพอใจในด้านต่าง ๆ โดยแบ่งเป็น 2 ด้าน ได้แก่ การออกแบบและการใช้งาน (Design and Usability) และฟังก์ชันการทำงาน (Functionality) โดยแบ่งการประเมินออกเป็น 5 ระดับ

- 1. ไม่พอใจมาก (Very Dissatisfied) - หมายถึง ผู้ใช้งานไม่พอใจมากในข้อคำถามนั้น ๆ
- 2. พอดีน้อย (Somewhat Dissatisfied) - หมายถึง ผู้ใช้งานพอใจน้อยในข้อคำถามนั้น ๆ
- 3. พอดีปานกลาง (Neutral) - หมายถึง ผู้ใช้งานพอใจปานกลางในข้อคำถามนั้น ๆ
- 4. พอดีมาก (Somewhat Satisfied) - หมายถึง ผู้ใช้งานพอใจมากในข้อคำถามนั้น ๆ
- 5. พอใจมากที่สุด (Very Satisfied) - หมายถึง ผู้ใช้งานพอใจมากที่สุดในข้อคำถามนั้น ๆ

# แบบสำรวจความพึงพอใจการใช้งาน Web Application: Gallery Walk

panthon.f@gmail.com สลับบัญชี



ระบบจะบันทึกอีเมลของคุณเมื่อส่งแบบฟอร์มนี้

\* ระบุว่าเป็นคำตามที่จำเป็น

## ข้อมูลทั่วไป

เพศ \*

- ชาย
- หญิง
- ไม่ต้องการระบุ
- อื่นๆ: \_\_\_\_\_

ท่านมีบทบาทใดในการใช้งาน Web Application นี้ \*

- ผู้สร้างงานกิจกรรม (Event Manager)
- ผู้นำเสนอโครงการ (Presenter)
- ผู้เข้าร่วมกิจกรรม (Guest)

กลับ

ถัดไป

ล้างแบบฟอร์ม

ห้ามส่งรหัสผ่านใน Google ฟอร์ม

เนื้อหาที่ได้ถูกสร้างขึ้นหรือรับรองโดย Google [รายงานการลงทะเบียน](#) - [ข้อกำหนดในการให้บริการ](#) - [นโยบายความเป็นส่วนตัว](#)

Google ฟอร์ม

รูปที่ 4.1: แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนข้อมูลทั่วไป

**แบบสำรวจความพึงพอใจการใช้งาน Web Application: Gallery Walk**

panthon.f@gmail.com ลับมัญชี 

ระบบจะบันทึกอีเมลของคุณเมื่อส่งแบบฟอร์มนี้

\* ระบุว่าเป็นค่าตามที่จำเป็น

**การออกแบบและการใช้งาน**

โปรดเลือกตาม "ระดับความพึงพอใจ" ของท่านต่อการใช้งาน ดังต่อไปนี้

การออกแบบหน้าจอใช้งานง่ายและเข้าใจง่ายหรือไม่? \*

1	2	3	4	5		
ใช้งานยากมาก	<input type="radio"/>	ใช้งานง่ายมาก				

โครงสร้างการจัดเรียงเนื้อหาเหมาะสมหรือไม่? \*

1	2	3	4	5		
ไม่เหมาะสมมาก	<input type="radio"/>	เหมาะสมมาก				

ความเร็วในการโหลดหน้าจอและการตอบสนองต่อการใช้งานเป็นอย่างไร? \*

1	2	3	4	5		
ช้ามาก	<input type="radio"/>	รวดเร็ว				

เงินแอปพลิเคชันมีเสถียรภาพหรือไม่? (มีการค้างหรือบุกทำงานบ่อย) \*

1	2	3	4	5		
ไม่เสถียรมาก	<input type="radio"/>	เสถียรมาก				

มีความสะดวกในการค้นหาข้อมูลหรือฟังก์ชันการทำงานที่ต้องการหรือไม่? \*

1	2	3	4	5		
ไม่สะดวกมาก	<input type="radio"/>	สะดวกมาก				

**กลับ**    **ต่อไป**    **ล้างแบบฟอร์ม**

ห้ามส่งรหัสผ่านใน Google ฟอร์ม

เงื่อนไขที่ได้ถูกสร้างขึ้นหรือรับรองโดย Google หมายความว่า - ข้อมูลนี้จะไม่สามารถใช้ในการให้บริการ - นโยบายความเป็นส่วนตัว

**Google ฟอร์ม**

รูปที่ 4.2: แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนการออกแบบและการใช้งาน (Design and Usability)

# แบบสำรวจความพึงพอใจการใช้งาน Web Application: Gallery Walk

panthon.f@gmail.com สลับบัญชี



ระบบจะบันทึกอีเมลของคุณเมื่อส่งแบบฟอร์มนี้

\* ระบุว่าเป็นค่าความที่จำเป็น

## พึงกันการทำงาน

โปรดเลือกตาม "ระดับความพึงพอใจ" ของท่านต่อการใช้งาน

พึงกันการทำงานครบถ้วนตามความต้องการหรือไม่ ? \*

1      2      3      4      5

ไม่ครบถ้วนมาก

ครบถ้วนมาก

พึงกันการทำงานใช้งานง่ายและเข้าใจง่ายหรือไม่ ? \*

1      2      3      4      5

ใช้งานยากมาก

ใช้งานง่ายมาก

มีข้อผิดพลาดหรือปัญหาในการใช้งานพึงกันการทำงานใดบ้าง ? (โปรดระบุ) \*

คำตอบของคุณ

กลับ

ตัดไป

ล้างแบบฟอร์ม

ห้ามส่งรหัสผ่านใน Google ฟอร์ม

เนื้อหาที่ได้ถูกสร้างขึ้นหรือรับรองโดย Google รายงานการลงทะเบียน - ข้อกำหนดในการให้บริการ - นโยบายความเป็นส่วนตัว

Google ฟอร์ม

รูปที่ 4.3: แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนพึงกันการทำงาน (Functionality)

## แบบสำรวจความพึงพอใจการใช้งาน Web Application: Gallery Walk

panthon.f@gmail.com สลับบัญชี



ระบบจะบันทึกอีเมลของคุณเมื่อส่งแบบฟอร์มนี้

\* ระบุว่าเป็นค่าตามที่จำเป็น

### ความพึงพอใจโดยรวม

ระดับความพึงพอใจโดยรวมในการใช้งานเว็บแอปพลิเคชัน \*

1      2      3      4      5

ไม่พึงพอใจมาก

พึงพอใจมาก

มีข้อเสนอแนะเพิ่มเติมหรือไม่?

ค่าตอบของคุณ

กลับ

ส่ง

ล้างแบบฟอร์ม

ห้ามส่งรหัสผ่านใน Google ฟอร์ม

เนื้อหาที่มีได้ถูกสร้างขึ้นหรือรับรองโดย Google [รายงานการประเมิน](#) - [ข้อกำหนดในการให้บริการ](#) - [นโยบายความเป็นส่วนตัว](#)

Google ฟอร์ม

รูปที่ 4.4: แบบสำรวจความพึงพอใจของผู้ใช้ในความพึงพอใจโดยรวม

### 4.3 สรุปผลการประเมินความพึงพอใจของผู้ใช้

จากการประเมินความพึงพอใจของผู้ใช้ พบร่วมกับผู้ใช้งานทั้ง 3 กลุ่ม คือ ผู้สร้างงานกิจกรรม (Event Manager) ผู้นำเสนองาน (Presenter) และแขกผู้เข้าร่วมงาน (Guest) มีความพึงพอใจในการใช้งานระบบที่พัฒนาขึ้น โดยคะแนนเฉลี่ยของการประเมินความพึงพอใจของผู้ใช้ทั้ง 3 กลุ่มโดยรวม คือ 4.2 จาก 5 โดยสามารถแยกตามด้านการประเมินความพึงพอใจของผู้ใช้ได้ดังนี้

- การออกแบบและการใช้งาน (Design and Usability) คะแนนเฉลี่ย 4.1 จาก 5
- ฟังก์ชันการทำงาน (Functionality) คะแนนเฉลี่ย 4.3 จาก 5

และมีข้อเสนอแนะในการปรับปรุงระบบเพื่อให้ตรงตามความต้องการของผู้ใช้ ดังนี้

- ผู้สร้างงานกิจกรรม (Event Manager)
  - ต้องการให้ระบบมีการแจ้งเตือนเมื่อมีโครงการใหม่เพิ่มเข้ามา
  - ต้องการให้มีการแสดงรายละเอียดโดยรวมของงานกิจกรรม (Dashboard) เช่น จำนวนโครงการที่เข้าร่วม จำนวนแขกที่เข้าร่วม การให้ Virtual Money ในแต่ละโครงการ และอื่น ๆ
- ผู้นำเสนองาน (Presenter)
  - ต้องการให้มีการแจ้งเตือนเมื่อมีผู้มาให้ Virtual Money หรือ แสดงความคิดเห็นในโครงการ
  - ต้องการให้ระบบสามารถเพิ่มผู้จัดการโครงการ (Project Manager) คนอื่นในโครงการ
- แขกผู้เข้าร่วมงาน (Guest)
  - ต้องการให้การแสดงผลโครงการต่าง ๆ ชัดเจนและสามารถกรองโครงการต่าง ๆ ได้ง่าย เช่น กรองจากหมวดหมู่ อักษรตัวแรก หรือ จาก Virtual Money ที่ได้รับ และอื่น ๆ

## บทที่ 5

### บทสรุปและข้อเสนอแนะ

#### 5.1 สรุปผล

การทำโครงการนี้สามารถพัฒนาเว็บ-แอปพลิเคชันที่สามารถใช้งานตามความต้องการของผู้ใช้งานได้ โดยผู้ใช้งานทั้ง 3 กลุ่ม คือ

- ผู้ใช้ที่เป็นผู้สร้างงานกิจกรรม (Event Manager) สามารถสร้างงานกิจกรรม และจัดการงานกิจกรรมได้
- ผู้ใช้ที่เป็นผู้นำเสนองานโครงการ (Presenter) สามารถสร้างโครงการ และจัดการโครงการได้
- ผู้ใช้ที่เป็นแขกผู้เข้าร่วมงาน (Guest) สามารถดูงานกิจกรรมและโครงการต่าง ๆ และให้ Virtual Money และแสดงความคิดเห็นในโครงการต่าง ๆ ได้

#### 5.2 ปัญหาที่พบและแนวทางการแก้ไข

ในการทำโครงการนี้ พบร่วมกับปัญหาหลัก ๆ ดังนี้

- การทำงานร่วมกันของทีม ทีมมีคืนทำงานร่วมกัน 3 คน แต่เนื่องจากการประชุมและการทำงานร่วมกันไม่สอดคล้อง ทำให้การทำงานมีความล่าช้าอยู่บ้าง
- การทดสอบระบบ การทดสอบระบบระหว่างการพัฒนาไม่ครอบคลุม ทำให้มีบางส่วนของระบบที่ไม่ทำงานอย่างที่ควรจะเป็น
- การแสดงผลข้อมูล การแสดงผลข้อมูลตามกลุ่มผู้ใช้งานไม่ครอบคลุม ทำให้ผู้ใช้งานบางกลุ่มไม่สามารถใช้งานระบบได้อย่างที่ควรจะเป็น

#### 5.3 ข้อเสนอแนะและแนวทางการพัฒนาต่อ

ข้อเสนอแนะเพื่อพัฒนาโครงการนี้ต่อไป มีดังนี้

ข้อเสนอที่ได้รับจากแบบสำรวจความพึงพอใจของผู้ใช้งานและผู้พัฒนาระบบ ได้แก่

- พัฒนาส่วน Dashboard สำหรับผู้ใช้งานที่เป็นผู้สร้างงานกิจกรรม (Event Manager) ให้มีการแสดงผลข้อมูลที่ครอบคลุมมากขึ้นในแต่ละกิจกรรม ซึ่งอาจประกอบไปด้วย จำนวนผู้เข้าร่วมงาน จำนวนโครงการที่เข้าร่วม จำนวน Virtual Money ที่แต่ละโครงการได้รับ แขกผู้เข้าร่วมงานให้ Virtual Money และแสดงความคิดเห็นในโครงการต่าง ๆ
- พัฒนาการแจ้งเตือนให้ผู้ใช้ที่เป็นผู้นำเสนองานโครงการ (Presenter) ให้มีการแจ้งเตือนเมื่อโครงการของตนได้รับ Virtual Money หรือมีความคิดเห็นใหม่ ๆ
- ในส่วนของการเพิ่มโครงการเข้ามาในงานกิจกรรม ให้มีการตรวจสอบและอนุมัติ (Approve) โดยการจากผู้สร้างงานกิจกรรมก่อนที่จะแสดงผลในงานกิจกรรม

### ข้อเสนอแนะที่ได้ผู้เป็นแรงบันดาลใจในการพัฒนา (Inspiration)

จากอาจารย์บุลวิชช์ ช่วยชูวงศ์ (พี่นุ) ผู้เชี่ยวชาญด้าน Design Thinking โดยคณะผู้จัดทำได้นำแพลทฟอร์มนี้ไปสาธิตการทำงานทั้ง 3 กลุ่มผู้ใช้งาน และได้รับคำแนะนำในการพัฒนาและแนวทางที่จะพัฒนาโปรแกรมของเราให้ดียิ่งขึ้น โดยได้เลือกเห็นว่าแพลทฟอร์มนี้จะช่วยตอบโจทย์กลุ่มผู้ใช้งานที่ต้องการจัดงานกิจกรรม และมี KPI (Key Performance Indicator) คือ จำนวนผู้เข้าร่วมชมกิจกรรมได้ เช่น กิจกรรมที่สสส. ต้องจัดกิจกรรมให้มีผู้เข้าชมไม่ต่ำกว่า 2,000 คน สสส. สามารถจัดงานกิจกรรมแบบ On-site เชิญชวนคนมาดูงาน และ จัดกิจกรรมแบบ Online ขนาดกันไป การจัด On-site ก็ไม่จำเป็นต้องรองรับคน 2,000 คนได้ อาจจะรับได้ 200 คน แล้วก็เชิญผู้ชมออนไลน์เข้าอีกได้พันกว่าคน ด้วยระบบนี้ ซึ่งจะช่วยอำนวยความสะดวกในเรื่องของการจัดงานเป็นอย่างมาก และเนื่องจากหลาย ๆ ท่านก็ได้เลือกเห็นถึงความเป็นไปได้ในการที่จะพัฒนาโครงการนี้ต่อ โดยมีสิ่งที่ได้แนะนำเพิ่มเติมดังนี้

- Investment return

โดยทางอาจารย์บุลวิชช์ ได้มองว่าเป็นคล้าย ๆ กับการลงทุน คือ สำหรับ Guest หรือนักลงทุนจำลอง ที่มาทำการลงทุน ก็จะต้องได้รับผลตอบแทนการลงทุน เพื่อกระตุ้นให้ Guest เลือกลงทุนในโปรเจค ที่มีศักยภาพในการพัฒนาได้จริง สร้างผลกำไรได้จริง ถ้าโครงการที่ Guest เลือกได้รับเงินระดมทุน เยอะในอันดับต้น ๆ ก็จะมี Reward กลับไปให้ เช่น เป็นจำนวนหุ้น (จำลอง) เงินปันผล เป็นต้น โดย ผู้จัดงานนี้ต้องทำการปรึกษากับผู้ใช้ระบบโดยละเอียดอีกครั้งว่าควรเป็นรีวอร์ด (Reward) ให้นักลงทุน ในรูปแบบใด

- Required Feedback

คำถาม Feedback ที่จะเจาะประเด็นต่าง ๆ โดยให้ผู้จัดงานกิจกรรมหรือผู้นำเสนองานสามารถ ตั้งคำถามเก็บข้อมูลจากผู้เข้าร่วมกิจกรรมได้ตรงกับเป้าหมายของกิจกรรมหรือโครงการนั้น ๆ มากยิ่ง ขึ้น

- Reveal The Guest

ผู้จัดงานกิจกรรม (Event manager) สามารถที่จะระบุได้ว่าอย่างจะให้กิจกรรมนั้น ๆ ชื่อนหรือเปิดเผยชื่อคนที่ให้ Virtual Money หรือชื่อคนที่มาแสดงความคิดเห็นได้ เนื่องจากบางโครงการที่เกี่ยวข้องกับการระดมทุนก็จำเป็นที่จะต้องเปิดเผยผู้ที่มาร่วมระดมทุน

- Comment Platform

เนื่องจากระบบของเรามีการแสดงความคิดเห็นเป็นจุดเด่น เราจึงต้องพัฒนาระบบนี้ให้รองรับการใช้งานเป็นแพลตฟอร์มสำหรับการสำรวจ เช่น ทดลองผลิตภัณฑ์ใหม่ ซึ่งมีตัวอย่างคร่าว ๆ ตามที่อาจารย์บุลวิชช์ได้แนะนำมา เช่น Sizzler ต้องการสร้างเมนูใหม่ แล้วอยากรู้ว่าลูกค้าจะยอมจ่ายในราคาเท่าไหร่

- โดยกำหนดให้ร้านเป็น Event Manager และสร้างเมนูอาหารไว้ในอีเวนท์
- ให้ลูกค้าเข้ามาเป็น Guest กำหนด Virtual money ให้ xxx Unit
- ให้ลูกค้าทดลองกินเมนูใหม่
- ให้ลูกค้าให้ Virtual Money ในราคายังที่ลูกค้าคิดว่าเหมาะสม

- มีริวอร์ด (Reward) กลับไปให้ลูกค้า (พีเจอร์ตอบแทนการลงทุนด้านบน) แต่กรณีนี้ ริวอร์ดจะไม่เกี่ยวกับเลือกโครงการ Top 3 แต่ให้ริวอร์ดกับลูกค้าทุกคนที่เข้าร่วมงานกิจกรรม

## บรรณานุกรม

- [1] CodeBee. (2016) Mvc คืออะไร ทำความเข้าใจรูปแบบการเขียนโปรแกรม. เข้าชมล่าสุดเมื่อ: 2 ตุลาคม 2566. [Online]. Available: <https://www.codebee.co.th/labs/mvc-คืออะไร-ทำความเข้าใจ/>
- [2] A. W. S. (AWS). (2023) Restful api คืออะไร. เข้าชมล่าสุดเมื่อ: 2 ตุลาคม 2566. [Online]. Available: <https://aws.amazon.com/th/what-is/restful-api/>
- [3] mindphp.com. (2022) Database คืออะไร ระบบฐานข้อมูล คือ การเก็บรวบรวมข้อมูลไว้ด้วยกัน อย่างมีระบบ. เข้าชมล่าสุดเมื่อ: 2 ตุลาคม 2566. [Online]. Available: <https://www.mindphp.com/คู่มือ/73-คืออะไร/2055-database-คืออะไร23.html>
- [4] ——. (2022) Html คืออะไร เอกซ์เพล็อก ภาษาคอมพิวเตอร์ที่ใช้ในการสร้างเว็บเพจ ใช้เขียนโปรแกรม ย่อมาจากอะไร. เข้าชมล่าสุดเมื่อ: 2 ตุลาคม 2566. [Online]. Available: <https://www.mindphp.com/คู่มือ/73-คืออะไร/2026-html-คืออะไร.html>
- [5] wynnsoft solution. (2016) Css คืออะไร ? มีประโยชน์อย่างไรบ้าง. เข้าชมล่าสุดเมื่อ: 3 ตุลาคม 2566. [Online]. Available: <https://www.wynnsoft-solution.net/th/article/view/80/>
- [6] S. Damyos. (2022) Typescript คืออะไร. เข้าชมล่าสุดเมื่อ: 3 ตุลาคม 2566. [Online]. Available: <https://il.mahidol.ac.th/th/i-learning-clinic/computer-articles/ทำไมถึงต้องใช้typescript-ทั้งที่/>
- [7] R. Documentation. (2023) React - a javascript library for building user interfaces. เข้าชมล่าสุดเมื่อ: 3 ตุลาคม 2566. [Online]. Available: <https://reactjs.org/>
- [8] V. Rtishchev. (2023) Mantine ui. เข้าชมล่าสุดเมื่อ: 3 กรกฎาคม 2566. [Online]. Available: <https://mantine.dev/>
- [9] [https://appmaster.io/th/blog/postgresql-khuue\\_aair](https://appmaster.io/th/blog/postgresql-khuue_aair). (2022) Postgresql คืออะไร. เข้าชมล่าสุดเมื่อ: 3 ตุลาคม 2566. [Online]. Available: <https://appmaster.io/th/blog/postgresql-khuue-aair>
- [10] Prisma. (2023) Prisma | next-generation orm for node.js and typescript. เข้าชมล่าสุดเมื่อ: 3 ตุลาคม 2566. [Online]. Available: <https://www.prisma.io/>

## ภาคผนวก

## ภาคผนวก ก

### คู่มือการติดตั้ง (สำหรับการพัฒนา)

การพัฒนาระบบทั้งหมดออกเป็น 2 ส่วน คือ ฝั่งหน้าบ้าน (Front-End) และฝั่งหลังบ้าน (Back-End) จึงต้องทำการติดตั้งแยกกัน ดังนี้

#### ก.1 ฝั่งหน้าบ้าน (Front-End)

- สิ่งที่เครื่องสำหรับติดตั้งต้องมีคือ Node.js และ npm โดยสามารถดาวน์โหลดได้จาก <https://nodejs.org/en/download/>
- Download source code หรือ Clone ได้จาก GitHub <https://github.com/Panthonf/frontend-gallery-walk.git>
- ทำการเปิด Terminal หรือ Command Prompt และเข้าไปที่โฟลเดอร์ที่เก็บ source code และทำการติดตั้งโดยใช้คำสั่ง npm install เพื่อทำการติดตั้ง package ที่จำเป็น
- เพิ่ม .env ไฟล์เพื่อกำหนดค่าต่าง ๆ ได้แก่
  - VITE\_BACKEND\_ENDPOINT คือ http://localhost:8080/api/login/google ที่ใช้ในการเชื่อมต่อกับฝั่งหลังบ้าน
  - VITE\_CHECK\_LOGIN คือ http://localhost:8080/api/isLoggedIn ที่ใช้ในการเชื่อมต่อกับฝั่งหลังบ้าน
  - VITE\_BASE\_ENDPOINTMENT คือ http://localhost:8080/api ที่ใช้ในการเชื่อมต่อกับฝั่งหลังบ้าน
  - VITE\_FRONTEND\_ENDPOINT คือ http://localhost:3000 ที่ใช้ในการเชื่อมต่อกับฝั่งหน้าบ้าน
- หลังจากติดตั้งเสร็จสิ้น สามารถทำการรันโปรแกรมได้โดยใช้คำสั่ง npm run dev และเข้าไปที่ <http://localhost:3000> ในเบราว์เซอร์ จะพบกับหน้าเว็บที่ใช้ในการใช้งาน

#### ก.2 ฝั่งหลังบ้าน (Back-End)

- สิ่งที่เครื่องสำหรับติดตั้งต้องมีคือ Node.js และ npm โดยสามารถดาวน์โหลดได้จาก <https://nodejs.org/en/download/>
- Download source code หรือ Clone ได้จาก GitHub <https://github.com/Panthonf/backend-gallery-walk.git>
- ทำการเปิด Terminal หรือ Command Prompt และเข้าไปที่โฟลเดอร์ที่เก็บ source code และทำการติดตั้งโดยใช้คำสั่ง npm install เพื่อทำการติดตั้ง package ที่จำเป็น
- เพิ่ม .env ไฟล์เพื่อกำหนดค่าต่าง ๆ ได้แก่

- PORT คือ 8080 ที่ใช้ในการเชื่อมต่อกับผู้คนบ้าน
  - CALLBACK\_URI คือ <http://localhost:8080/api/login/google/callback> ที่ใช้ในการเชื่อมต่อกับผู้คนบ้าน
  - DATABASE\_URL คือ Connection String ของ PostgreSQL
  - SECRET\_KEY คือ คีย์สำหรับการเข้ารหัส Token
  - NODE\_ENV คือ development สำหรับการพัฒนา และ production สำหรับการใช้งานจริง
  - FRONTEND\_URL คือ <http://localhost:3000>
  - MINIO\_ENDPOINT คือ Minio Server
  - MINIO\_URL คือ Minio Server
  - MINIO\_ACCESSKEY คือ Minio Access Key
  - MINIO\_SECRETKEY คือ Minio Secret Key
  - MINIO\_PORT คือ 443
  - MINIO\_USESSL คือ true
  - CALLBACK\_URI\_GUEST คือ <http://localhost:8080/api/guests/login/google/callback>
- ทำการสร้าง Database โดยใช้คำสั่ง `npx prisma migrate dev --name init` จะทำการสร้าง Database และ Table ตามที่กำหนดไว้ในไฟล์ `schema.prisma`
  - หลังจากติดตั้งเสร็จสิ้น สามารถทำการรันโปรแกรมได้โดยใช้คำสั่ง `npm run dev` และสามารถเรียกใช้งาน API ผ่าน Postman หรือเว็บเบราว์เซอร์ได้

ภาคผนวก ข  
คู่มือการใช้งาน

สามารถดูวิธีการใช้งานได้จากวิดีโอที่อัพโหลดไว้ที่ <https://www.youtube.com/watch?v=fKbrI9rsoqg>

## ประวัติผู้เขียน



**ชื่อ-นามสกุล:** นางสาวณัฐวรรณ เรียบเรียง **สถาบันการศึกษา:** มหาวิทยาลัยเชียงใหม่  
**สาขา:** วิศวกรรมคอมพิวเตอร์ **อีเมล:** nuttawan.reabreang@gmail.com **โทรศัพท์:** 0885294254



**ชื่อ-นามสกุล:** นายณานะอิป ภู่สว่าง **สถาบันการศึกษา:** มหาวิทยาลัยเชียงใหม่  
**สาขา:** วิศวกรรมคอมพิวเตอร์ **อีเมล:** zen2002yanatib@gmail.com **โทรศัพท์:** 0905293055



**ชื่อ-นามสกุล:** นายปันธ์ธร กันทรพย **สถาบันการศึกษา:** มหาวิทยาลัยเชียงใหม่  
**สาขา:** วิศวกรรมคอมพิวเตอร์ **อีเมล:** panthon.f@gmail.com **โทรศัพท์:** 0935314738