

โครงการเลขที่ วศ.คพ. P809-2/2566

เรื่อง

แพลตฟอร์มการแลกเปลี่ยนความคิดสร้างสรรค์ดิจิทัล: การเสริมสร้างการเรียนรู้ผ่าน
Gallery Walk

โดย

นายญาณธิป ภู่อ่าง	รหัส 630612097
นางสาวณัฐวรรณ เรียบเรียง	รหัส 630612099
นายปิ่นฤทธิ์ร กันทรพย	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 : นายญาณธิป ภู่อ่าง รหัสนี้ 630612097
นางสาวณัฐวรรณ เรียบเรียง รหัสนี้ 630612099
นายปิ่นธิธร กันทรพิญ รหัสนี้ 630612105

ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : ผศ.โดม โพธิกานนท์
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2566

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเชียงใหม่ ได้อนุมัติให้โครงการนี้เป็นส่วน
หนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต (สาขาวิศวกรรมคอมพิวเตอร์)

..... หัวหน้าภาควิชาวิศวกรรมคอมพิวเตอร์
(ศ.ดร.สันติ พิทักษ์กัญญู)

คณะกรรมการสอบโครงการ

..... ประธานกรรมการ
(ผศ.โดม โพธิกานนท์)

..... กรรมการ
(อ.ดร.ชินวัตร อิศราดิษฐ์กุล)

..... กรรมการ
(อ.ดร.ธนาธิปย์ จันทร์คง)

หัวข้อโครงการ : แพลตฟอร์มการแลกเปลี่ยนความคิดสร้างสรรค์ดิจิทัล: การเสริมสร้างการเรียนรู้ผ่าน
Gallery Walk
: Digital Creativity Exchange Platform: Enhancing Learning through
Gallery Walk

โดย : นายญาณธิป ภู่อ่าง รหัส 630612097
นางสาวณัฐวรรณ เรียบเรียง รหัส 630612099
นายปณัฏฐ์ กันทรทรัพย์ รหัส 630612105

ภาควิชา : วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา : ผศ.โดม โพธิ์กานนท์
ปริญญา : วิศวกรรมศาสตรบัณฑิต
สาขา : วิศวกรรมคอมพิวเตอร์
ปีการศึกษา : 2566

บทคัดย่อ

Gallery Walk คือการเรียนรู้แบบเดิมชมผลงาน ซึ่งเป็นเทคนิคการเรียนรู้ที่กระตุ้นให้ผู้เรียนแลกเปลี่ยนข้อมูล ความรู้หรือประสบการณ์ร่วมกันโดยใช้ผลงานที่ผู้เรียนแต่ละกลุ่มร่วมกันสร้างสรรค์ขึ้นเป็นสื่อกลางในการถ่ายทอดและใช้กระตือรือร้นสนทนาเป็นสื่อกลางในการรับคำติชมและข้อเสนอแนะจากผู้อื่น เพื่อนำข้อมูลป้อนกลับนั้นมาพัฒนาผลงานของกลุ่มหรือเติมเต็มความรู้ความเข้าใจของตัวเอง

ในปัจจุบันการจัดกิจกรรมในรูปแบบ Gallery Walk นั้นมีความยุ่งยากในการจัดการ เนื่องจากผู้จัดการกิจกรรม (Event Manager) จะต้องจัดเตรียมสถานที่ สื่อการสื่อสาร และการจัดการเอกสารต่าง ๆ ที่เกี่ยวข้องกับกิจกรรม เป็นภาระต่อผู้จัดการกิจกรรม นอกจากนี้ยังไม่สามารถจัดกิจกรรมในรูปแบบ Hybrid ได้ ในกรณีที่ผู้เข้าร่วมงาน (Guest) ต้องการที่จะเยี่ยมชมผลงาน โครงการต่าง ๆ แบบออนไลน์ เพื่อให้ Virtual Money และแสดงความคิดเห็นต่าง ๆ และส่วนสุดท้ายไม่สามารถแสดงผลได้ทันทีว่าผลงานใดได้รับการประเมินสูงสุดหลังเสร็จสิ้นงานกิจกรรม

จากที่กล่าวมานั้น คณะผู้จัดทำจึงได้จัดทำโครงการเพื่อพัฒนาเว็บแอปพลิเคชันสำหรับการจัดกิจกรรม Gallery Walk ในรูปแบบ Hybrid ซึ่งสามารถให้ผู้จัดงานกิจกรรมสามารถจัดการกิจกรรมได้ง่ายขึ้น และสามารถให้ผู้เข้าร่วมงานสามารถเยี่ยมชมผลงาน โครงการต่าง ๆ แบบออนไลน์ และสามารถแสดงความคิดเห็นต่าง ๆ และส่วนสุดท้ายสามารถแสดงผลได้ทันทีว่าผลงานใดได้รับการประเมินสูงสุดหลังเสร็จสิ้นงานกิจกรรม

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.โดม โพธิกานนท์ ผู้ที่ได้มอบเวลาและความกรุณาเพื่อเป็นที่ปรึกษาในโครงการนี้ ได้ให้คำแนะนำที่มีคุณค่าและแนวทางที่ช่วยให้โครงการเกิดขึ้นอย่างสมบูรณ์และประสบความสำเร็จ

ไม่เพียงแต่นั้นเท่านั้น ขอขอบคุณอย่างสูงสำหรับคำปรึกษาจาก อ.ดร.ชินวัตร อิศราดิศัยกุล และ ผศ.ดร.ธนาทิพย์ จันทรงค์ ที่ได้มอบคำแนะนำและเสนอแนวทางที่มีประสิทธิภาพ ทำให้โครงการเกิดประสิทธิภาพมากขึ้น

นอกจากนี้ เราต้องขอบคุณเพื่อนร่วมทีมทุกคนที่ให้การสนับสนุนและกำลังใจ และขอบคุณครอบครัวที่เป็นกำลังใจสำคัญในการผ่านช่วงเวลาที่ทำหายไปด้วยกัน ขอขอบคุณทุกท่านที่เคยให้คำแนะนำและความช่วยเหลือ โดยที่ไม่ระบุนาม ทุกคำปรึกษาและการสนับสนุนที่มีค่ามากสำหรับเรา

หากหนังสือโครงการเล่มนี้มีข้อผิดพลาดประการใดผู้จัดทำขอน้อมรับด้วย ความยินดียิ่งและขออภัยมา ณ ที่นี้ด้วย

นายญาณธิป ภู่อ่าง
นางสาวณัฐวรรณ เรียบเรียง
นายปณัฏฐ์ร กัณฑ์พร
20 มีนาคม 2567

สารบัญ

บทคัดย่อ	ข
Abstract	ค
กิตติกรรมประกาศ	ง
สารบัญ	จ
สารบัญรูป	ช
สารบัญตาราง	ซ
1 บทนำ	1
1.1 ที่มาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบเขตของโครงการ	1
1.3.1 ขอบเขตด้านฮาร์ดแวร์	1
1.3.2 ขอบเขตด้านซอฟต์แวร์	1
1.4 ประโยชน์ที่ได้รับ	2
1.5 เทคโนโลยีและเครื่องมือที่ใช้	2
1.5.1 เทคโนโลยีด้านฮาร์ดแวร์	2
1.5.2 เทคโนโลยีด้านซอฟต์แวร์	2
1.6 แผนการดำเนินงาน	3
1.7 บทบาทและความรับผิดชอบ	3
2 ทฤษฎีที่เกี่ยวข้อง	4
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
3 โครงสร้างและขั้นตอนการทำงาน	13
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)	22
3.2.4	ผู้นำเสนอผลงาน (Presenter)	27
3.2.5	ผู้เข้าร่วมงาน (Guest)	29
4	การทดลองและผลลัพธ์	31
4.1	การประเมินความพึงพอใจของผู้ใช้	31
4.2	ผลการประเมินความพึงพอใจของผู้ใช้	31
4.3	สรุปผลการประเมินความพึงพอใจของผู้ใช้	34
5	บทสรุปและข้อเสนอแนะ	35
5.1	สรุปผล	35
5.2	ปัญหาที่พบและแนวทางการแก้ไข	35
5.3	ข้อเสนอแนะและแนวทางการพัฒนาต่อ	35
	บรรณานุกรม	36

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	หน้าแสดงกิจกรรมที่สร้าง (Event manager dashboard)	22
3.10	หน้าแสดงข้อมูลรายละเอียดของกิจกรรมที่สร้าง โดยแบ่ง Tab เป็น Event information, Project และ Result	23
3.11	หน้าแสดงรายการโครงการที่เข้าร่วมกิจกรรม	23
3.12	หน้าแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม โดยแสดงเป็น Ranking และตาราง Virtual Money	24
3.13	ตารางแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม	24
3.14	หน้าสร้างกิจกรรม First step (Event information)	25
3.15	หน้าสร้างกิจกรรม Second step (Images Dropzone)	25
3.16	หน้าสร้างกิจกรรม Final step (Check all information before submit)	26
3.17	หน้าเพิ่มโครงการที่เข้าร่วมกิจกรรม (Add project)	27
3.18	หน้าแสดงโครงการที่สร้าง (Presenter dashboard)	27
3.19	หน้าแสดงข้อมูลรายละเอียดของผลงานที่สร้างในกิจกรรม	28
3.20	หน้าเข้าสู่ระบบ (Login) สำหรับผู้เข้าร่วมกิจกรรม (Guest)	29
3.21	หน้าแสดงข้อมูลรายละเอียดของงานกิจกรรมที่จัดขึ้น และโครงการที่เข้าร่วมกิจกรรม	29
3.22	หน้าให้ Virtual Money และแสดงความคิดเห็นเกี่ยวกับโครงการที่เข้าร่วมกิจกรรม	30
3.23	หน้าแสดงความคิดเห็นเกี่ยวกับโครงการที่เข้าร่วมกิจกรรม	30
4.1	แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนข้อมูลทั่วไป	32
4.2	แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนการออกแบบและการใช้งาน (Design and Usability)	32
4.3	แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนฟังก์ชันการทำงาน (Functionality)	33
4.4	แบบสำรวจความพึงพอใจของผู้ใช้ในความพึงพอใจโดยรวม	33

สารบัญตาราง

3.1	ตารางข้อมูลผู้ใช้งาน (Users)	16
3.2	ตารางข้อมูลงานกิจกรรม (Events)	17
3.3	ตารางข้อมูลโครงการ (Projects)	17
3.4	ตารางข้อมูลเงินเสมือน (Virtual Money)	17
3.5	ตารางข้อมูลแขกผู้เข้าร่วมกิจกรรม (Guests)	18
3.6	ตารางข้อมูลภาพตัวอย่างโครงการ (Thumbnails)	18
3.7	ตารางข้อมูลเอกสารที่เกี่ยวข้องกับงานกิจกรรม (Documents)	18
3.8	ตารางข้อมูลความคิดเห็นของผู้เข้าร่วมกิจกรรม (Comments)	18
3.9	ตารางข้อมูลรูปภาพที่เกี่ยวข้องกับโครงการ (Project Images)	19

บทที่ 1

บทนำ

1.1 ที่มาของโครงการ

โครงการ Gallery Walk มีความหมายในการให้ผู้เรียนมีโอกาสดูแลเปลี่ยนความรู้และประสบการณ์ผ่านผลงานที่สร้างขึ้นเป็นสื่อกลาง และการใช้กระดานสนทนาเพื่อรับคำแนะนำและคำชมเชยจากผู้อื่น เพื่อให้สามารถพัฒนาผลงานหรือเพิ่มเติมความรู้แก่ตนเองได้

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

ดังนั้น เพื่อแก้ไขปัญหาดังกล่าว ทีมผู้จัดงานได้พัฒนาเว็บแอปพลิเคชันสำหรับการจัดกิจกรรม Gallery Walk ในรูปแบบ Hybrid เพื่อให้การจัดการกิจกรรมเป็นไปอย่างมีประสิทธิภาพ และเพื่อให้ผู้เข้าร่วมสามารถเยี่ยมชมผลงานและให้ความเห็นออนไลน์ได้ นอกจากนี้ยังสามารถแสดงผลการประเมินได้ทันทีหลังจากเสร็จสิ้นกิจกรรม

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อให้การจัดการกิจกรรมของผู้จัดกิจกรรมสะดวกและมีประสิทธิภาพมากขึ้น
2. เพื่อสร้างช่องทางสำหรับการดำเนินกิจกรรมทางออนไลน์ให้กับผู้เข้าร่วมกิจกรรม สามารถจัดกิจกรรมในรูปแบบ Hybrid ได้
3. การประเมินที่เน้นการมีส่วนร่วมและรับความคิดเห็น ซึ่งส่งเสริมการพัฒนาผลงานของผู้นำเสนอโครงการ

1.3 ขอบเขตของโครงการ

1.3.1 ขอบเขตด้านฮาร์ดแวร์

โครงการนี้ต้องการฮาร์ดแวร์ต่อไปนี้ จึงจะสามารถใช้งานได้มีประสิทธิภาพ

- คอมพิวเตอร์ส่วนบุคคลหรือโทรศัพท์มือถือที่สามารถใช้งานเว็บเบราว์เซอร์ได้

1.3.2 ขอบเขตด้านซอฟต์แวร์

โครงการนี้ต้องการซอฟต์แวร์ต่อไปนี้ จึงจะสามารถใช้งานได้มีประสิทธิภาพ

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

โดยการแสดงผลของเว็บไซต์จะแสดงผลได้เหมาะสม ตามอุปกรณ์ที่ใช้งานและตามบทบาทผู้ใช้งานดังต่อไปนี้

- ผู้จัดการกิจกรรม (Event Manager): คอมพิวเตอร์ส่วนบุคคล
- ผู้นำเสนอโครงการ (Presenter): คอมพิวเตอร์ส่วนบุคคล
- แขกผู้เข้าร่วมกิจกรรม (Guest): คอมพิวเตอร์ส่วนบุคคลหรือโทรศัพท์มือถือ

1.4 ประโยชน์ที่ได้รับ

ผู้ใช้งาน

- ผู้จัดการกิจกรรม (Event Manager)
 - สามารถจัดการกิจกรรมได้อย่างมีประสิทธิภาพ
 - สามารถจัดการกิจกรรมแบบ Hybrid ได้
 - สามารถแสดงผลการประเมินได้ทันทีหลังจากเสร็จสิ้นกิจกรรม
- ผู้นำเสนอโครงการ (Presenter)
 - สามารถนำเสนอผลงานของตนได้อย่างมีประสิทธิภาพ
 - สามารถรับความคิดเห็นและคำแนะนำจากผู้เข้าร่วมกิจกรรมได้อย่างมีประสิทธิภาพ
- ผู้เข้าร่วมกิจกรรม (Guest)
 - สามารถเยี่ยมชมผลงานและให้ความเห็นออนไลน์ได้อย่างมีประสิทธิภาพ

ผู้พัฒนา

- ได้รับความรู้และความเข้าใจในการพัฒนาเว็บแอปพลิเคชัน
- ได้ฝึกทักษะในการใช้เทคโนโลยีที่เกี่ยวข้องกับการพัฒนาเว็บแอปพลิเคชัน
- ได้ฝึกทักษะในการทำงานเป็นทีมและทักษะในการวิเคราะห์และแก้ไขปัญหาที่อาจเกิดขึ้นในการพัฒนา

1.5 เทคโนโลยีและเครื่องมือที่ใช้

1.5.1 เทคโนโลยีด้านฮาร์ดแวร์

1.5.2 เทคโนโลยีด้านซอฟต์แวร์

- ภาษาโปรแกรมมิ่ง: 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
คิดหัวข้อและนำเสนอหัวข้อโครงการ									
ศึกษาค้นคว้าเกี่ยวกับเทคโนโลยีที่เกี่ยวข้อง									
ออกแบบ UI/UX ของเว็บแอปพลิเคชัน									
พัฒนาเว็บแอปพลิเคชัน									
ทดสอบและปรับปรุงเว็บแอปพลิเคชัน									

1.7 บทบาทและความรับผิดชอบ

มีหน้าที่และความรับผิดชอบ ดังนี้

นายญาณธิป ภู่อ่าง รหัสนักศึกษา 630612097 รับผิดชอบในการศึกษาค้นคว้าเทคโนโลยีที่เกี่ยวข้อง, ออกแบบโครงสร้างของเว็บแอปพลิเคชันและพัฒนาเว็บแอปพลิเคชันส่วนหน้าบ้าน (Frontend)

นางสาวณัฐวรรณ เรียบเรียง รหัสนักศึกษา 630612099 รับผิดชอบในการศึกษาค้นคว้าเทคโนโลยีที่เกี่ยวข้อง, ออกแบบโครงสร้างของเว็บแอปพลิเคชันและพัฒนาเว็บแอปพลิเคชันส่วนหน้าบ้าน (Frontend)

นายปณัฏฐ์ กันทรพิทย์ รหัสนักศึกษา 630612105 รับผิดชอบในการศึกษาค้นคว้าเทคโนโลยีที่เกี่ยวข้อง, ออกแบบโครงสร้างของเว็บแอปพลิเคชันและพัฒนาเว็บแอปพลิเคชันส่วนหลังบ้าน (Backend)

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 ด้านโครงสร้างเว็บแอปพลิเคชัน

ในส่วนนี้จะอธิบายถึงโครงสร้างของเว็บแอปพลิเคชันที่ใช้ในการพัฒนา

2.1.1 MVC Architecture

MVC [1] เป็นตัวย่อของคำว่า Model View Controller ใช้เรียกรูปแบบการพัฒนาซอฟต์แวร์ที่มีโครงสร้างซึ่งแบ่งออกเป็น 3 ส่วนหลัก ตามตัวย่อของชื่อ รูปแบบการพัฒนาซอฟต์แวร์แบบ MVC ถูกนำไปใช้ในขั้นตอนการพัฒนาหลากหลายภาษา เพราะ MVC เป็นเพียงหลักการออกแบบโปรแกรม (Design Pattern) รูปแบบหนึ่งเท่านั้น ซึ่งเป็นที่นิยมมาก ในการนำมาพัฒนาแอปพลิเคชันซอฟต์แวร์แต่ละแพลตฟอร์มและประยุกต์ใช้ในอีกหลาย ๆ ด้าน

ส่วนของ Model (M)

model คือส่วนของการเก็บรวบรวมข้อมูล ไม่ว่าข้อมูลนั้น ๆ จะถูกจัดเก็บในรูปแบบใดก็ตาม ในฐานข้อมูลแบบเป็น Object Class หรือที่นิยมเรียกกันว่า VO (Value Object) หรือเก็บเป็นไฟล์ข้อมูลเลย เมื่อข้อมูลถูกโหลดเข้ามาจากที่ต่าง ๆ และเข้ามายังส่วนของโมเดล ตัวโมเดลจะทำการจัดการเตรียมข้อมูลให้เป็นรูปแบบที่เหมาะสม เพื่อรอการร้องขอข้อมูลจากส่วนของ Controller

ส่วนของ View (V)

view คือส่วนของการแสดงผล หรือส่วนที่จะปฏิสัมพันธ์กับผู้ใช้งาน (User Interface) หน้าที่ของ view ในการเขียนโปรแกรมแบบ MVC คือคอยรับคำสั่งจากส่วนของ Controller และ End User เริ่มแรกเลยตัว view อาจจะรับคำสั่งจาก Controller ให้แสดงผลหน้า Home และเมื่อผู้ใช้งานหน้าเว็บกดปุ่มสั่งซื้อ View จะส่งข้อมูลไปให้ Controller เพื่อประมวลผลและแสดงบางอย่างจาก Action นั้น

ส่วนของ Controller (C)

controller คือส่วนของการเริ่มทำงาน และรับคำสั่ง โดยที่คำสั่งนั้นจะเกิดขึ้นในส่วนการติดต่อกับผู้ใช้งานคือ view เมื่อผู้ใช้งานทำการ Interactive กับ UI view จะเกิดเหตุการณ์หรือข้อมูลบางอย่างขึ้น ตัว view จะส่งข้อมูลนั้น มายัง controller ตัว controller จะทำการประมวลผลโดยบางคำสั่งอาจจะต้องไปติดต่อกับ model ก่อน เพื่อทำการประมวลผลข้อมูลอย่างถูกต้องเรียบร้อยแล้วก็จะส่งไปยัง view เพื่อแสดงผลตามคำสั่งที่ end user ร้องขอมา Controller จะทำหน้าที่เป็นตัวกลางระหว่าง Model และ View ให้ทำงานร่วมกันอย่างมีประสิทธิภาพและตรงกับ ความต้องการของ End User มากที่สุด

2.1.2 RESTful API

RESTful API [2] เป็นอินเทอร์เฟซที่ระบบคอมพิวเตอร์สองระบบใช้เพื่อแลกเปลี่ยนข้อมูลผ่านอินเทอร์เน็ตได้อย่างปลอดภัย แอปพลิเคชันทางธุรกิจส่วนใหญ่ต้องสื่อสารกับแอปพลิเคชันภายในอื่นๆ และของบุคคล

ที่สามเพื่อทำงานต่างๆ ตัวอย่างเช่น หากต้องการสร้างสลิปเงินเดือน ระบบบัญชีภายในของคุณต้องแบ่งปันข้อมูลกับระบบธนาคารของลูกค้าเพื่อออกใบแจ้งหนี้และสื่อสารกับแอปพลิเคชันบันทึกเวลาปฏิบัติงานภายใน โดยอัตโนมัติ RESTful API ให้การสนับสนุนการแลกเปลี่ยนข้อมูลนี้เพราะเป็นระบบที่มีมาตรฐานการสื่อสารระหว่างซอฟต์แวร์ที่ปลอดภัย เสถียร และมีประสิทธิภาพ

API (Application Programming Interface)

ส่วนต่อประสานโปรแกรมประยุกต์ (Application Programming Interface หรือ API) กำหนดกฎที่คุณต้องปฏิบัติตามเพื่อสื่อสารกับระบบซอฟต์แวร์อื่น โดยนักพัฒนาเปิดเผยหรือสร้าง API เพื่อให้แอปพลิเคชันอื่นสามารถสื่อสารกับแอปพลิเคชันของตนได้ทางโปรแกรม ตัวอย่างเช่น แอปพลิเคชันบันทึกเวลาปฏิบัติงานแสดง API ที่ขอชื่อเต็มของพนักงานและช่วงวันที่ เมื่อได้รับข้อมูลนี้แล้ว ระบบจะประมวลผลบันทึกเวลาปฏิบัติงานของพนักงานเป็นการภายใน และส่งกลับจำนวนชั่วโมงที่ทำงานในช่วงวันที่ดังกล่าว ทั้งนี้คุณสามารถมองได้ว่า API เว็บเป็นเกตเวย์ระหว่างไคลเอนต์และทรัพยากรบนเว็บ

ไคลเอนต์ ไคลเอนต์คือผู้ใช้ที่ต้องการเข้าถึงข้อมูลจากเว็บ โดยไคลเอนต์อาจเป็นบุคคลหรือระบบซอฟต์แวร์ที่ใช้ API ก็ได้ ตัวอย่างเช่น นักพัฒนาสามารถเขียนโปรแกรมที่เข้าถึงข้อมูลสภาพอากาศจากระบบสภาพอากาศ หรือคุณสามารถเข้าถึงข้อมูลเดียวกันจากเบราว์เซอร์เมื่อคุณเยี่ยมชมเว็บไซต์รายงานสภาพอากาศได้โดยตรง

ทรัพยากร ทรัพยากรคือข้อมูลที่แอปพลิเคชันต่างๆ มอบให้แก่ไคลเอนต์ โดยทรัพยากรอาจเป็นรูปภาพ วิดีโอ ข้อความ ตัวเลข หรือข้อมูลประเภทใดก็ได้ ทั้งนี้เครื่องคอมพิวเตอร์ที่มอบทรัพยากรให้แก่ไคลเอนต์นั้นเรียกอีกอย่างว่าเซิร์ฟเวอร์ องค์กรต่างๆ ใช้ API เพื่อแบ่งปันทรัพยากรและให้บริการเว็บในขณะที่ยังคงดูแลรักษาความปลอดภัย การควบคุม และการรับรองความถูกต้องไปพร้อมกัน นอกจากนี้ API ยังช่วยให้ลูกค้าระบุได้ว่าไคลเอนต์ใดสามารถเข้าถึงทรัพยากรภายในที่เฉพาะเจาะจงได้

REST (Representational State Transfer)

REST เป็นสถาปัตยกรรมซอฟต์แวร์ที่กำหนดเงื่อนไขว่า API ควรทำงานอย่างไร โดยแต่แรกเริ่มนั้น มีการสร้าง REST ขึ้นเพื่อเป็นแนวทางในการจัดการการสื่อสารบนเครือข่ายที่ซับซ้อน เช่น อินเทอร์เน็ต คุณสามารถใช้สถาปัตยกรรม REST เพื่อรองรับการสื่อสารที่มีประสิทธิภาพสูงและเชื่อถือได้ในทุกระดับ คุณยังสามารถใช้และปรับเปลี่ยนสถาปัตยกรรมได้อย่างง่ายดาย โดยนำความสามารถในการมองเห็นและการเคลื่อนย้ายข้ามแพลตฟอร์มมาสู่ทุกระบบ API

นักพัฒนา API สามารถออกแบบ API ได้โดยใช้สถาปัตยกรรมต่างๆ 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 มีความสัมพันธ์ระหว่างข้อมูลต่าง ๆ ที่ชัดเจน ในระบบฐานข้อมูลจะประกอบด้วยแฟ้มข้อมูลหลายแฟ้มที่มีข้อมูลเกี่ยวข้องสัมพันธ์กันเข้าไว้ด้วยกันอย่างเป็นระบบและเปิดโอกาสให้ผู้ใช้สามารถใช้งาน และดูแล

รักษาป้องกันข้อมูลเหล่านี้ได้อย่างมีประสิทธิภาพ โดยมีซอฟต์แวร์ที่เปรียบเสมือนสื่อกลางระหว่าง ผู้ใช้และโปรแกรมต่าง ๆ ที่เกี่ยวข้องกับการใช้ฐานข้อมูล เรียกว่า ระบบจัดการฐานข้อมูล หรือ DBMS (data base management system) มีหน้าที่ช่วยให้ผู้ใช้เข้าถึงข้อมูลได้ง่ายสะดวกและมีประสิทธิภาพ การเข้าถึงข้อมูลของผู้ใช้อาจเป็นการสร้างฐานข้อมูล การแก้ไขฐานข้อมูล หรือการตั้งคำถามเพื่อให้ได้ข้อมูลมา โดยผู้ใช้ไม่จำเป็นต้องรับรู้เกี่ยวกับรายละเอียดภายในโครงสร้างของฐานข้อมูล

ประโยชน์ของฐานข้อมูล

1. ลดการเก็บข้อมูลที่ซ้ำซ้อน

ข้อมูลบางชุดที่อยู่ในรูปของแฟ้มข้อมูลอาจมีปรากฏอยู่หลาย ๆ แห่ง เพราะมีผู้ใช้ข้อมูลชุดนี้หลายคน เมื่อใช้ระบบฐานข้อมูลแล้วจะช่วยให้ความซ้ำซ้อนของข้อมูลลดน้อยลง

2. รักษาความถูกต้องของข้อมูล

เนื่องจากฐานข้อมูลมีเพียงฐานข้อมูลเดียว ใน กรณีที่มีข้อมูลชุดเดียวกันปรากฏอยู่หลายแห่งในฐานข้อมูล ข้อมูลเหล่านี้จะต้องตรงกัน ถ้ามีการแก้ไขข้อมูลนี้ทุก ๆ แห่งที่ข้อมูลปรากฏอยู่จะแก้ไขให้ถูกต้องตามกันหมดโดยอัตโนมัติด้วยระบบจัดการฐานข้อมูล

3. การป้องกันและรักษาความปลอดภัยให้กับข้อมูลทำได้อย่างสะดวก

การป้องกันและรักษาความปลอดภัยกับข้อมูลระบบฐานข้อมูลจะให้เฉพาะผู้ที่เกี่ยวข้องเท่านั้น ซึ่งก่อให้เกิดความปลอดภัย (security) ของข้อมูลด้วย

2.2 ด้านเทคโนโลยี

ในส่วนนี้จะอธิบายถึงเทคโนโลยีที่ใช้ในการพัฒนาเว็บแอปพลิเคชัน

2.2.1 HTML

HTML [4] ย่อมาจาก Hyper Text Markup Language คือภาษาคอมพิวเตอร์ที่ใช้ในการแสดงผลของเอกสารบน website หรือที่เราเรียกกันว่าเว็บเพจ ถูกพัฒนาและกำหนดมาตรฐานโดยองค์กร World Wide Web Consortium (W3C) และจากการพัฒนาทางด้าน Software ของ Microsoft ทำให้ภาษา HTML เป็นอีกภาษาหนึ่งที่ใช้เขียนโปรแกรมได้ หรือที่เรียกว่า HTML Application HTML เป็นภาษาประเภท Markup สำหรับการการสร้างเว็บเพจ โดยใช้ภาษา HTML สามารถทำได้โดยใช้โปรแกรม Text Editor ต่างๆ เช่น VS Code, Vim หรือจะอาศัยโปรแกรมที่เป็นเครื่องมือช่วยสร้างเว็บเพจ เช่น Dream Weaver ซึ่งอำนวยความสะดวกในการสร้างหน้า 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

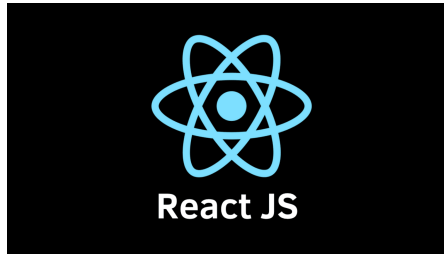
CSS [5] ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตลชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผล กำหนดให้รูปแบบของการแสดงผลเอกสาร ไม่ขึ้นอยู่กับเนื้อหาของเอกสาร เพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลลัพท์ของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้ง หรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน 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] คือภาษา 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 จะช่วยให้คุณสร้าง Software ที่ปรับใช้งานได้ง่ายและมีประสิทธิภาพมากยิ่งขึ้น

2.2.4 React JS

React JS [7] เป็นไลบรารี JavaScript ที่ใช้สร้าง User Interface (UI) ในเว็บแอปพลิเคชันแบบ Single Page Application (SPA) และเว็บแอปพลิเคชันที่มีการอัปเดตสดๆ 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 ได้ซึ่งช่วยให้การพัฒนาเร็วขึ้น
4. รองรับการทำงานแบบ Server-side Rendering (SSR) และ Client-side Rendering (CSR): ทำให้เหมาะสำหรับการพัฒนาเว็บแอปพลิเคชันที่มี SEO ดีและประสิทธิภาพการโหลดที่ดี

2.2.5 Mantine UI

Mantine UI [8] เป็นไลบรารีของ React Component ที่ถูกออกแบบมาเพื่อช่วยในการพัฒนา UI ในโปรเจกต์ของเรา มันมีชุดของ Components ที่หลากหลายและนำใช้งานอย่างง่าย เหมาะสำหรับการสร้าง UI ที่สวยงามและมีประสิทธิภาพ การใช้ Mantine UI ช่วยให้เราสามารถประสานงานกับผู้ออกแบบและนักพัฒนาได้อย่างมีประสิทธิภาพเพื่อสร้างผลลัพธ์ที่มีคุณภาพและประทับใจได้ในโปรเจกต์ของเรา การนำ Mantine UI เข้ามาใช้ยังช่วยลดเวลาในการพัฒนาโดยทั่วไปด้วยความสามารถในการทดสอบและปรับแต่งที่มีอยู่อย่างสมบูรณ์ โดยทั้งหมดนี้ช่วยให้เราสามารถให้ผลลัพธ์สุดท้ายในโปรเจกต์ของเราได้อย่างมีประสิทธิภาพและมีคุณภาพสูงสุดในเวลาที่มีจำกัด

2.2.6 PostgreSQL

PostgreSQL [9] เป็นฐานข้อมูลเชิงสัมพันธ์แบบโอเพ่นซอร์สระดับธุรกิจที่ทรงพลัง อนุญาตให้ใช้ข้อมูลและแบบสอบถาม SQL เชิงสัมพันธ์และ JSON ที่ไม่ใช่เชิงสัมพันธ์ PostgreSQL มีชุมชนที่แข็งแกร่งอยู่เบื้องหลัง PostgreSQL เป็นระบบจัดการฐานข้อมูลที่น่าเชื่อถือมาก พร้อมการสนับสนุน ความปลอดภัย และ



รูปที่ 2.4: PostgreSQL

ความแม่นยำในระดับดีเยี่ยม โทรศัพท์มือถือและเว็บแอปพลิเคชันจำนวนมากใช้ PostgreSQL เป็นฐานข้อมูลเริ่มต้น โซลูชันเชิงพื้นที่และการวิเคราะห์จำนวนมากใช้ประโยชน์จาก PostgreSQL เวอร์ชันล่าสุดคือ PostgreSQL 15 PostgreSQL รองรับประเภทข้อมูลที่ซับซ้อน ในความเป็นจริง ฐานข้อมูลถูกสร้างขึ้นโดยคำนึงถึงประเภทข้อมูลจำนวนมาก ประสิทธิภาพของฐานข้อมูลนั้นใกล้เคียงกับของคู่แข่งเช่น Oracle และ SQL Server AWS ให้บริการฐานข้อมูลที่ได้รับการบำรุงรักษาอย่างสมบูรณ์สำหรับ PostgreSQL ด้วยบริการฐานข้อมูลเชิงสัมพันธ์ของ Amazon PostgreSQL ยังใช้ในการสร้าง Amazon Aurora อีกด้วย

คุณสมบัติที่สำคัญของ PostgreSQL

หนึ่งในเหตุผลที่ PostgreSQL ได้รับความนิยมมากเนื่องจากชุดคุณลักษณะ ฐานข้อมูลช่วยในการ พัฒนาแอปพลิเคชัน โดยการรักษาความสมบูรณ์ของข้อมูล ช่วยให้ผู้ใช้ดูแลระบบสามารถสร้างสภาพแวดล้อมที่ทนต่อความผิดพลาดได้ นอกจากนี้ยังสามารถใช้ข้ามแพลตฟอร์มที่หลากหลายและใช้ประโยชน์จากภาษาโปรแกรมทั่วไประบบ เราจะได้เห็นรายชื่อที่แน่นอนในภายหลัง ฐานข้อมูลยังมีระบบล็อกขั้นสูงมาก นอกจากนี้ยังมีการควบคุมการทำงานพร้อมกันหลายเวอร์ชัน เซิร์ฟเวอร์ฐานข้อมูล PostgreSQL ยังมีฟังก์ชันการทำงานสำหรับการเขียนโปรแกรมฝั่งเซิร์ฟเวอร์สำหรับผู้ใหญ่อีกด้วย เป็นไปตามข้อกำหนด ANSI SQL และรองรับสถาปัตยกรรมเครือข่ายไคลเอ็นต์-เซิร์ฟเวอร์อย่างสมบูรณ์

PostgreSQL ยังมีความพร้อมใช้งานสูงและเซิร์ฟเวอร์สำรอง สอดคล้องกับ ANSI-SQL2008 และเชิงวัตถุ ความสามารถในการเชื่อมต่อกับคลังข้อมูลอื่นๆ เช่น NoSQL ซึ่งทำหน้าที่เป็นฮับแบบครบวงจรสำหรับระบบหลายภาษา สามารถทำได้ผ่านการสนับสนุน JSON ของฐานข้อมูล ข้อมูลของคลัสเตอร์ฐานข้อมูลเดียวจะได้รับการจัดการโดยอินสแตนซ์ PostgreSQL หนึ่งอินสแตนซ์เสมอ คลัสเตอร์ของฐานข้อมูลคือกลุ่มของเร็กคอร์ดที่เก็บไว้ในที่เดียวกันบนระบบไฟล์

2.2.7 Prisma

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

ประโยชน์ของ Prisma ORM ได้แก่:

1. ความสะดวกและความยืดหยุ่นในการใช้งาน: Prisma ORM ช่วยลดความซับซ้อนในการจัดการฐานข้อมูลและทำให้การเขียนโค้ดเป็นเรื่องที่ง่ายขึ้น ด้วยรูปแบบที่เป็นตัวถามและเรียกใช้ method ของ Prisma เพื่อสร้างและจัดการข้อมูล



รูปที่ 2.5: Prisma ORM

2. การสร้าง query ที่ปลอดภัย: Prisma ORM ช่วยป้องกันการโจมตีด้านความปลอดภัยเช่น SQL Injection โดยมีการ validate และ escape ข้อมูลอัตโนมัติ
3. การสร้างฐานข้อมูลแบบพลวัต: Prisma ORM ช่วยให้สามารถสร้างและแก้ไขโครงสร้างของฐานข้อมูลได้อย่างง่ายดาย ผ่านการใช้งาน migration เพื่อเปลี่ยนแปลง schema ของฐานข้อมูล
4. ประสิทธิภาพและความเร็ว: Prisma ORM มีการจัดการข้อมูลแบบอัตโนมัติที่เป็นไปอย่างมีประสิทธิภาพ ทำให้การเข้าถึงข้อมูลเป็นไปอย่างรวดเร็ว

Prisma ORM ได้ถูกพัฒนาโดย Prisma Labs และเป็นโปรเจกต์โอเพนซอร์ส ที่สามารถใช้งานได้ฟรีและเป็นส่วนหนึ่งของชุมชนนักพัฒนาซอฟต์แวร์ สามารถดาวน์โหลดและใช้งาน Prisma ORM ได้จากเว็บไซต์ของเครื่องมือนี้แบบฟรี นอกจากนี้ยังมีเอกสารและคู่มือการใช้งานที่มีอยู่ในชุมชนสำหรับการศึกษาและการใช้งานอื่น ๆ อีกด้วย

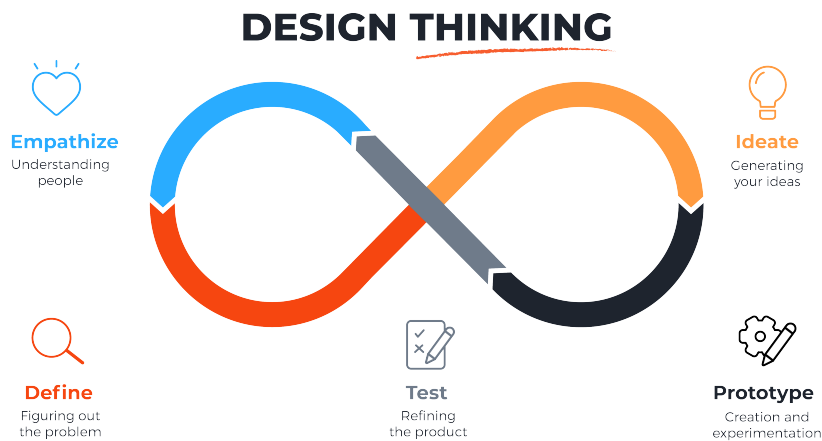
2.3 ด้าน User Interface

ในส่วนนี้จะอธิบายถึงการออกแบบ User Interface ของเว็บแอปพลิเคชัน

2.3.1 Design Thinking

กระบวนการออกแบบ design thinking นั้นมีหลากหลายรูปแบบ ทั้งรูปแบบ 3 ชั้น ไปจนถึง 7 ชั้น 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 การทำความเข้าใจกับปัญหาก่อน ตั้งแต่การเข้าใจผู้ใช้ กลุ่มเป้าหมาย หรือเข้าใจสิ่งที่ต้องการแก้ไขเพื่อหาหนทางที่เหมาะสม และดีที่สุดให้ได้ โดยเริ่มต้นจากการเข้าใจคำถาม สร้างสมมติฐาน กระตุ้นให้เกิดการใช้ความคิดที่นำไปสู่ความคิด สร้างสรรค์ และวิเคราะห์ปัญหาให้ถี่ถ้วน เพื่อหาแนวทางที่ชัดเจน นำไปสู่การแก้ไขปัญหาที่ตรงประเด็น และสร้างผลลัพธ์ที่ดีที่สุด



รูปที่ 2.6: กระบวนการออกแบบ Design Thinking

2. **Define** หรือ กำหนดปัญหาให้ชัดเจน คือ การเข้าใจความต้องการ ปัญหา และวิเคราะห์ข้อมูลเชิงลึก เพื่อคัดกรองหาปัญหาที่แท้จริง กำหนดหรือบ่งชี้ปัญหาอย่างชัดเจน เพื่อที่จะเป็นแนวทางในการปฏิบัติ และมีทิศทางในการแก้ไขปัญหาอย่างชัดเจน
3. **Ideate** หรือ ระดมความคิด คือ การนำเสนอแนวคิดต่างๆร่วมกัน ถึงวิธีการแก้ไขปัญหา อย่างไม่มีกรอบจำกัด การระดมความคิดควรมีมุมมองหลากหลาย และมีหลากหลายแนวทางให้ได้มากที่สุด เพื่อให้มีฐานข้อมูลในการนำไปวิเคราะห์และสรุปผล เพื่อนำไปแก้ไขปัญหา โดยไม่จำเป็นต้องเป็นแนวทางใดแนวทางหนึ่ง และการระดมความคิดยังช่วยมองเห็นปัญหาที่หลากหลายได้มากขึ้น
4. **Prototype** หรือ สร้างต้นแบบที่เลือก คือ การออกแบบผลิตภัณฑ์หรือนวัตกรรม เพื่อสร้างต้นแบบสำหรับการทดสอบ และนำไปใช้จริง ซึ่งคือ การลงมือปฏิบัติหรือทดลองตามแนวทางการแก้ไขปัญหาที่ได้กำหนดไว้
5. **Test** หรือ ทดสอบการแก้ไขปัญหา นำ **Prototype** ที่เราทำการทำขึ้นมาไปทดสอบกับผู้ใช้ว่าสามารถแก้ไขปัญหาของ ผู้ใช้ได้หรือไม่ และหลังจากนั้นถ้าหากการแก้ปัญหายังไม่สามารถช่วยแก้ไขได้ หรือแก้ไขได้ยังไม่ดีพอ ผู้จัดทำจะต้องกลับไปทำตั้งแต่ขั้นตอนแรกอีกครั้งจนกว่าจะสามารถออกแบบโปรแกรมที่แก้ไขปัญหา ของผู้ใช้ได้

2.4 ความรู้ตามหลักสูตรซึ่งถูกนำมาใช้หรือบูรณาการในโครงการ

- 261207 Basic CPE Lab นำความรู้ทางด้านการพัฒนาเว็บแอปพลิเคชัน เช่น HTML, CSS, Tailwind CSS, JavaScript, TypeScript, Next.js และ Node.js มาประยุกต์ใช้ในการพัฒนาเว็บแอปพลิเคชัน 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 รวมถึงการเชื่อมต่อกับฐานข้อมูล และการเขียนคำสั่ง SQL ในการดึงข้อมูลจากฐานข้อมูลและการจัดการข้อมูลในฐานข้อมูล
- 261200 Object-Oriented Programming ใช้เป็นพื้นฐานในการเขียนโปรแกรมในการพัฒนาเว็บแอปพลิเคชัน

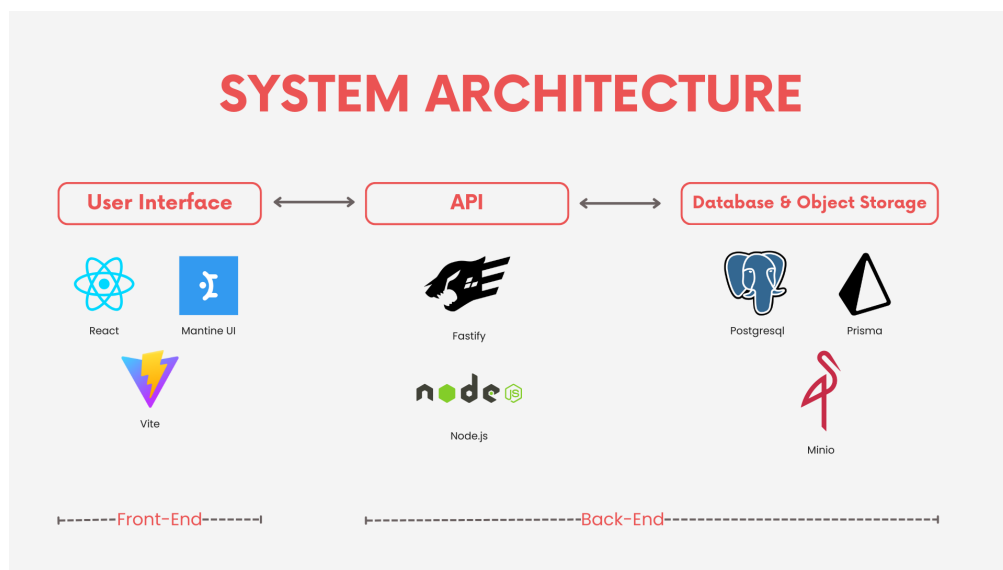
บทที่ 3

โครงสร้างและขั้นตอนการทำงาน

ในบทนี้จะกล่าวถึงหลักการ และการออกแบบระบบ

3.1 หลักการทำงานของระบบ

3.1.1 การทำงานของระบบ (System Architecture)



รูปที่ 3.1: System Architecture

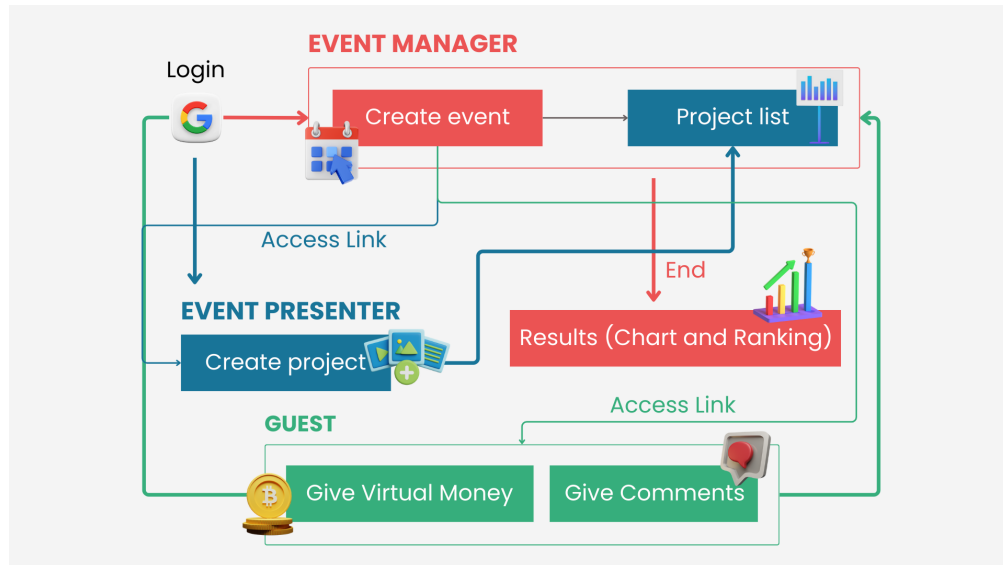
จากรูปที่ 3.1 จะเป็นภาพรวมของระบบที่เราได้ทำการออกแบบขึ้น โดยมีรายละเอียดดังนี้

- Frontend

ส่วนหน้าบ้าน (Frontend) เป็นส่วนการพัฒนาเพื่อแสดง User Interface (UI) โดยโครงการนี้ได้ใช้เทคโนโลยี React ร่วมกับ Mantine UI ในการออกแบบและสร้างส่วนติดต่อผู้ใช้ (UI) โดยมีหน้าที่แสดงผลต่อผู้ใช้ในรูปแบบที่เข้าใจง่าย รับข้อมูลป้อนจากผู้ใช้ผ่านอินเทอร์เฟซต่าง ๆ เช่น ปุ่ม ฟิลด์ข้อความ ฯลฯ และสื่อสารกับ API เพื่อส่งคำร้องขอและรับผลลัพธ์

- Backend

ส่วนหลังบ้าน (Backend) เป็นส่วนที่ทำหน้าที่รับข้อมูลจากผู้ใช้ จากนั้นทำการประมวลผลข้อมูล และส่งข้อมูลกลับไปยังผู้ใช้ 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 โดยป้อนข้อมูลต่างๆ เช่น ชื่องานกิจกรรม รายละเอียด วันเวลา สถานที่ เป็นต้น
- ผู้ใช้สร้าง Event เสร็จสิ้น จะมี QR Code หรือ Access Link สำหรับนำไปให้ผู้นำเสนอโครงการ (Presenter) สำหรับเพิ่มโครงการของตนและผู้เข้าร่วมกิจกรรม (Guest) สำหรับเข้าร่วมกิจกรรม
- ผู้ใช้สามารถดูผลลัพธ์ (Chart และ Ranking) ของโครงการต่าง ๆ ที่เข้าร่วมกิจกรรมในระหว่างจัดงานกิจกรรม

ผู้นำเสนอโครงการ (Presenter)

- ผู้ใช้เข้าสู่ระบบโดยใช้ Google Account
- ผู้ใช้สร้าง Project ใหม่ โดยป้อนข้อมูลต่าง ๆ เช่น ชื่อโครงการ รายละเอียด รูปภาพ ลิงก์ และไฟล์อื่น ๆ ที่เกี่ยวข้อง
- ผู้ใช้ดูผลลัพธ์ว่าโครงการของตนได้รับ Virtual Money จากผู้เข้าร่วมกิจกรรม (Guest) และความคิดเห็นเกี่ยวกับโครงการของตนหลังจากเสร็จสิ้นงานกิจกรรม

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

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
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)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
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	จำนวนเงินเสมือนที่จะให้แก่ผู้เข้าร่วมงาน (Guest)
unit_money	VARCHAR	หน่วยเงินเสมือนที่จะให้แก่ผู้เข้าร่วมงาน (Guest)
published	BOOLEAN	สถานะการเผยแพร่ของงานกิจกรรม
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
title	VARCHAR	ชื่อโครงการ
description	TEXT	รายละเอียดของโครงการ
user_id	INTEGER	รหัสผู้ใช้งานที่สร้างโครงการ
event_id	INTEGER	รหัสงานกิจกรรมที่โครงการเข้าร่วม
created_at	TIMESTAMP	วันที่สร้างข้อมูล

ตารางที่ 3.3: ตารางข้อมูลโครงการ (Projects)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
amount	INTEGER	จำนวนเงินเสมือนที่ได้รับ
project_id	INTEGER	รหัสโครงการที่ได้รับเงิน
event_id	INTEGER	รหัสงานกิจกรรมที่ได้รับเงิน
guest_id	INTEGER	รหัสผู้เข้าร่วมกิจกรรมที่ได้รับเงิน
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

ตารางที่ 3.4: ตารางข้อมูลเงินเสมือน (Virtual Money)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
first_name_th	VARCHAR	ชื่อจริงภาษาไทย
last_name_th	VARCHAR	นามสกุลภาษาไทย
first_name_en	VARCHAR	ชื่อจริงภาษาอังกฤษ
last_name_en	VARCHAR	นามสกุลภาษาอังกฤษ
email	TEXT	ที่อยู่อีเมล
profile_pic	TEXT	รูปภาพโปรไฟล์
virtual_money	INTEGER	จำนวนเงินเสมือนที่มีอยู่
created_at	TIMESTAMP	วันที่สร้างข้อมูล

ตารางที่ 3.5: ตารางข้อมูลแขกผู้เข้าร่วมกิจกรรม (Guests)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
event_id	INTEGER	รหัสงานกิจกรรมที่ได้รับเงิน
thumbnail	TEXT	รูปภาพตัวอย่างโครงการ
thumbnail_url	TEXT	ลิงก์รูปภาพตัวอย่างโครงการ
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

ตารางที่ 3.6: ตารางข้อมูลภาพตัวอย่างโครงการ (Thumbnails)

ชื่อคอลัมน์	ชนิดข้อมูล	คำอธิบาย
project_id	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
project_id	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
project_id	INTEGER	รหัสโครงการที่เกี่ยวข้องกับรูปภาพ
project_image	TEXT	รูปภาพโครงการ
project_image_url	TEXT	ลิงก์รูปภาพโครงการ
created_at	TIMESTAMP	วันที่สร้างข้อมูล
updated_at	TIMESTAMP	วันที่อัปเดตข้อมูล

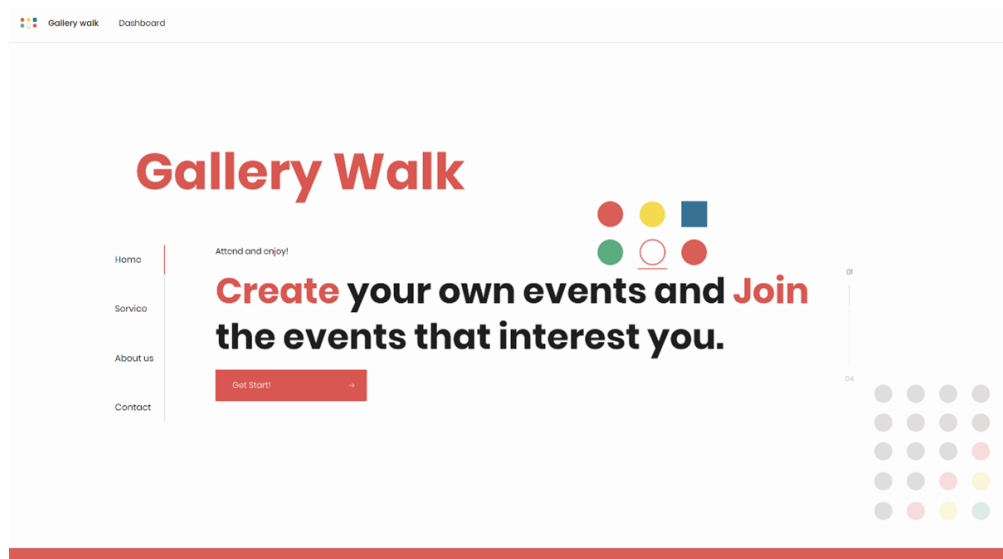
ตารางที่ 3.9: ตารางข้อมูลรูปภาพที่เกี่ยวข้องกับโครงการ (Project Images)

3.2 ส่วนเชื่อมต่อระหว่างผู้ใช้งานกับระบบ (User Interface)

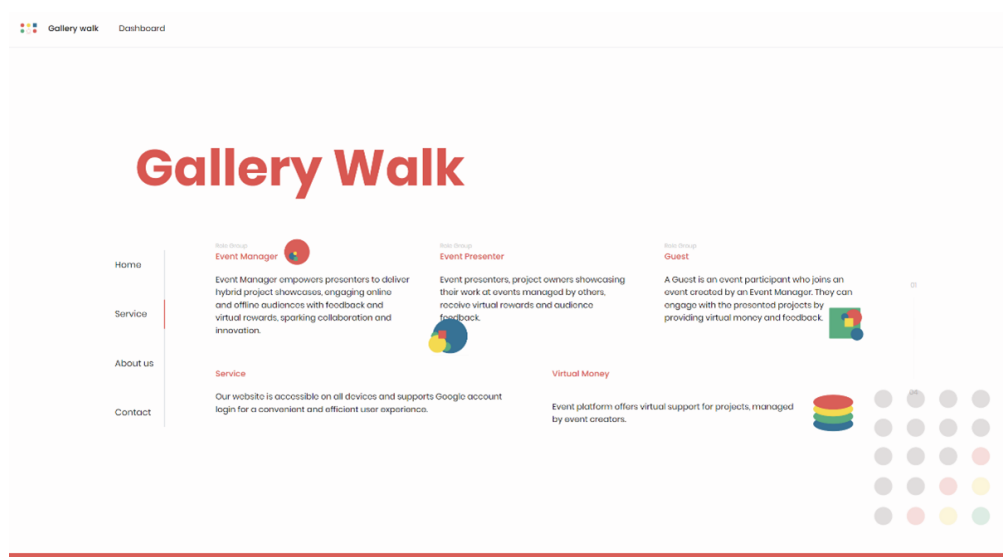
โดยแบ่งเป็น 3 ผู้ใช้งาน คือ ผู้สร้างงานกิจกรรม (Event manager), ผู้นำเสนอผลงาน (Presenter), ผู้เข้าร่วมงาน (Guest)

3.2.1 หน้าเริ่มต้นการใช้งาน (Homepage)

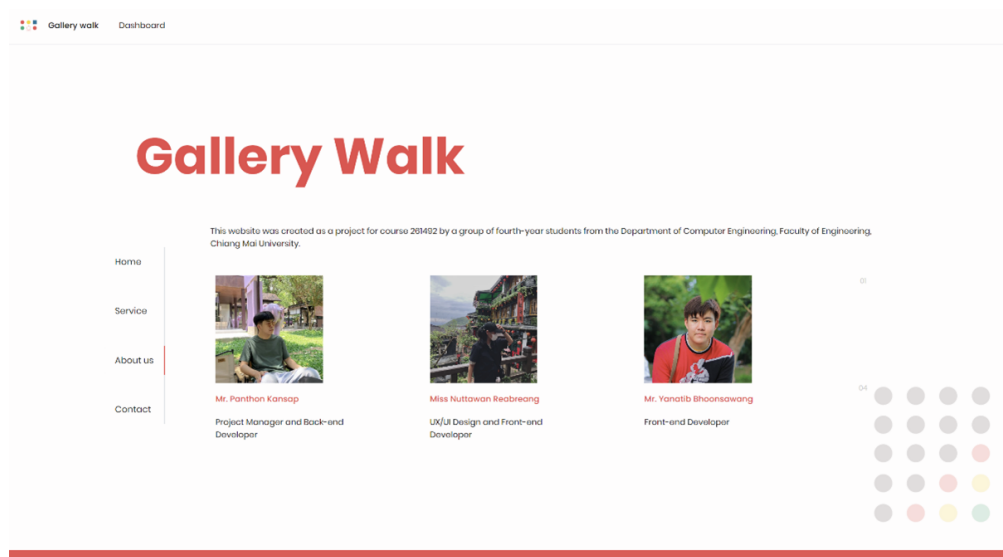
โดยแบ่ง Tab เป็น Home, Service, About us และ Contact สำหรับ Event manager และ Presenter



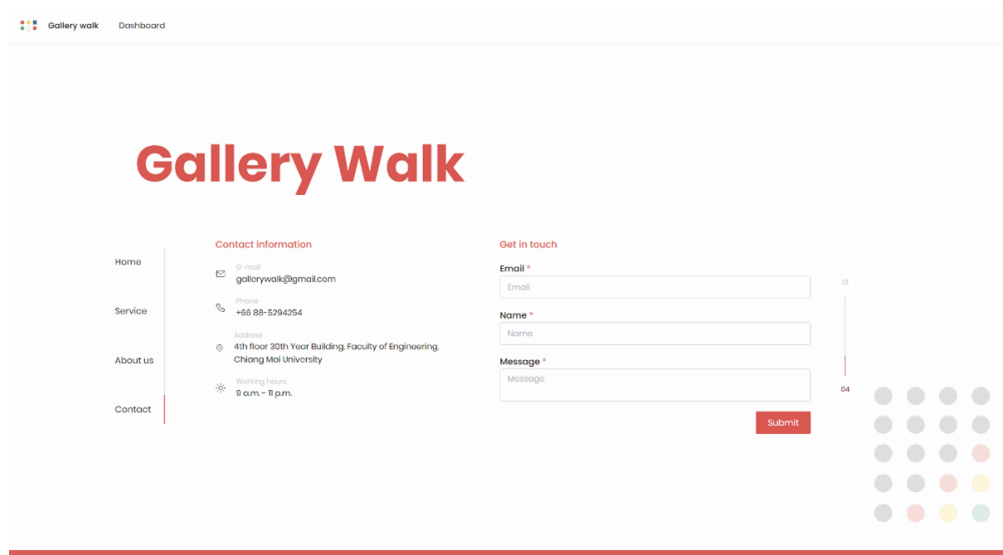
รูปที่ 3.4: หน้า Homepage



รูปที่ 3.5: หน้า Service



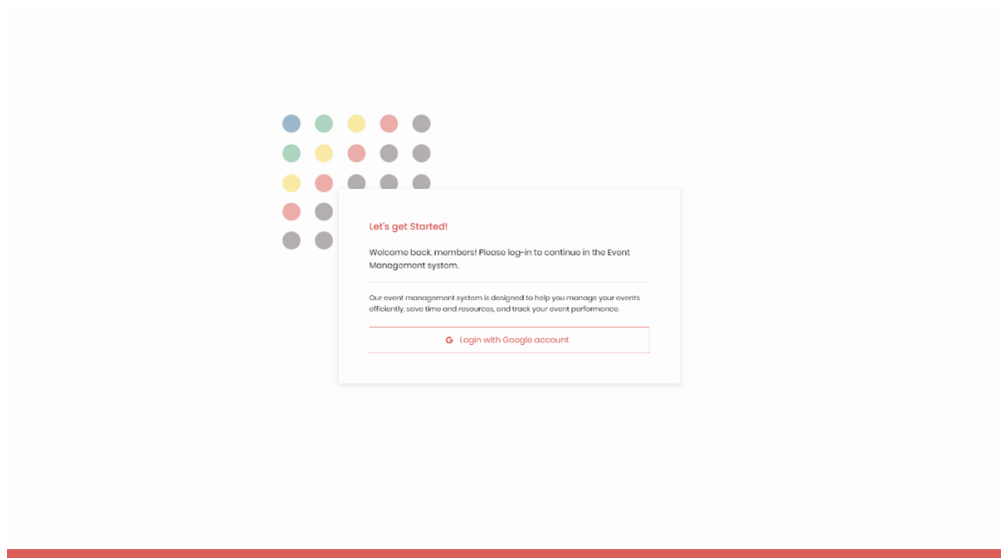
รูปที่ 3.6: หน้า About us



รูปที่ 3.7: หน้า Contact

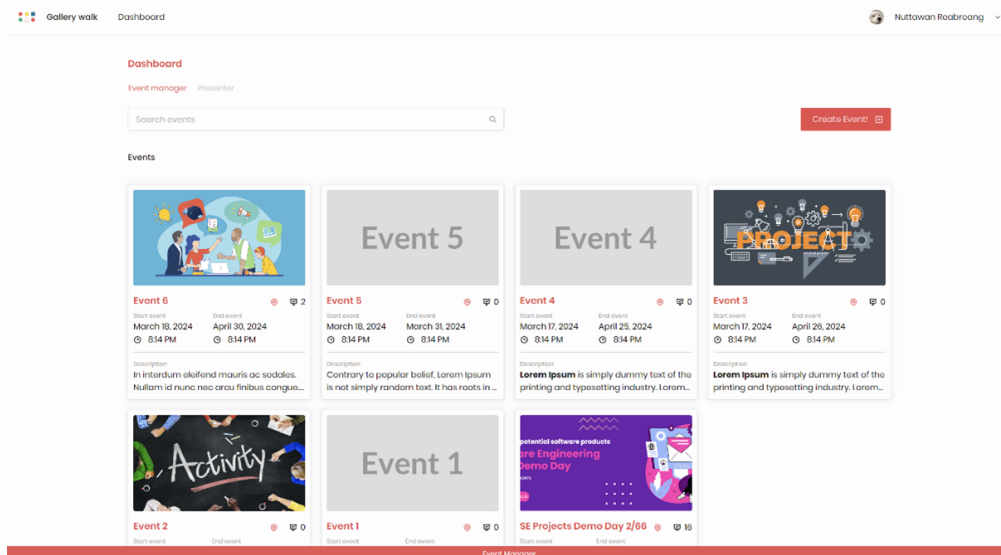
3.2.2 หน้าเข้าสู่ระบบ (Login)

Login ด้วย Google account สำหรับ Event manager และ Presenter

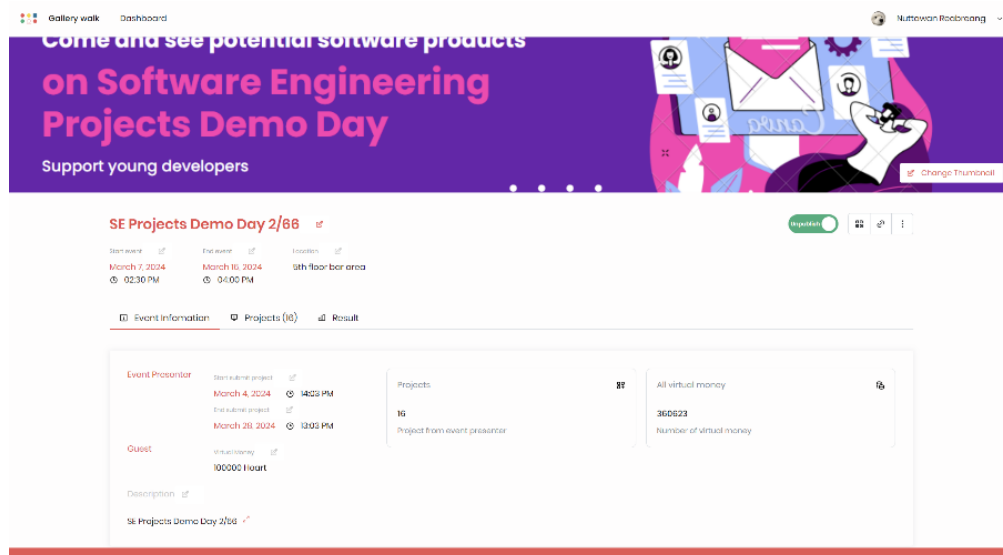


รูปที่ 3.8: หน้า Login สำหรับ Event manager และ Presenter

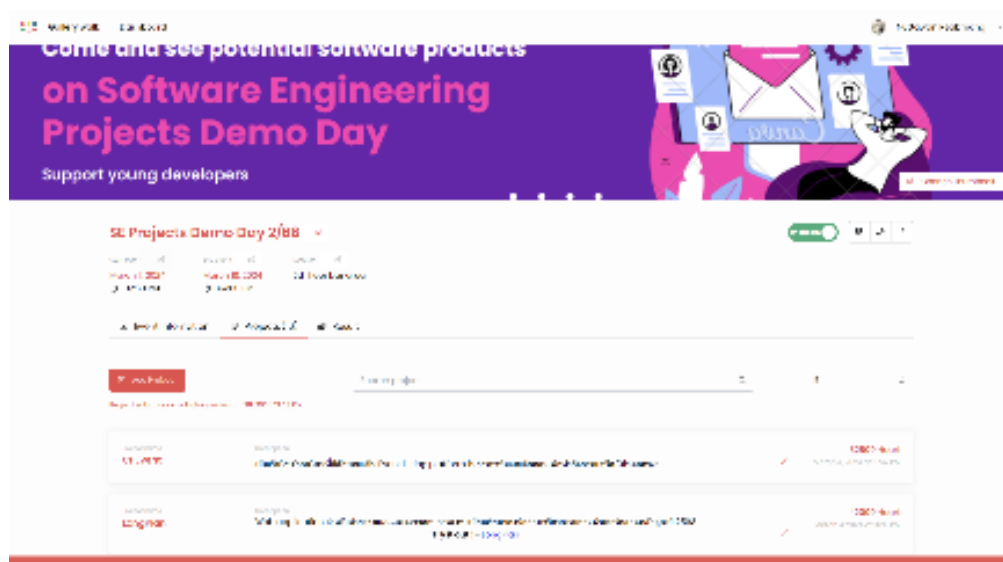
3.2.3 ผู้สร้างงานกิจกรรม (Event manager)



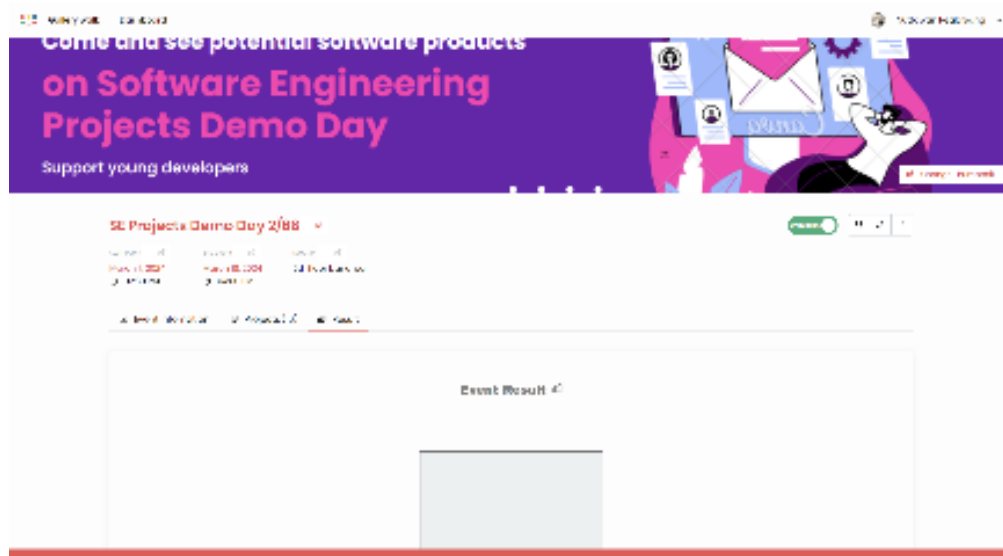
รูปที่ 3.9: หน้าแสดงกิจกรรมที่สร้าง (Event manager dashboard)



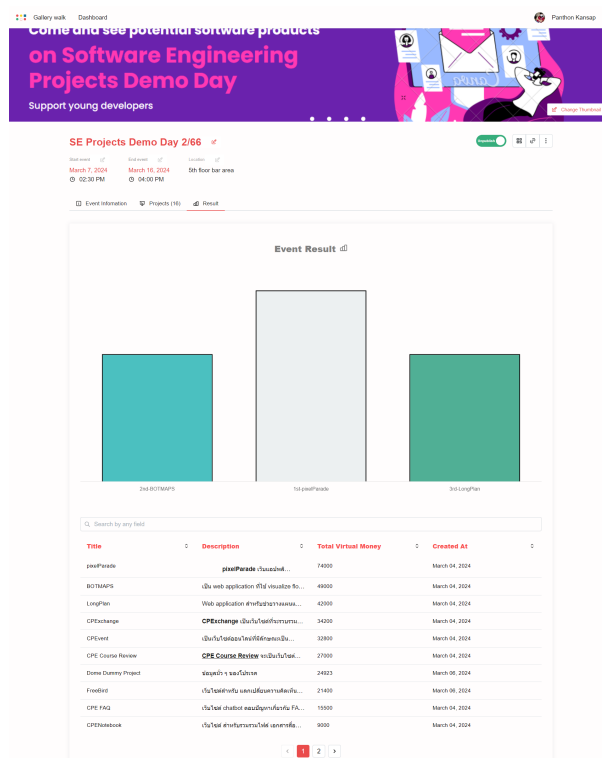
รูปที่ 3.10: หน้าแสดงข้อมูลรายละเอียดของกิจกรรมที่สร้าง โดยแบ่ง Tab เป็น Event information, Project และ Result



รูปที่ 3.11: หน้าแสดงรายการโครงการที่เข้าร่วมกิจกรรม



รูปที่ 3.12: หน้าแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม โดยแสดงเป็น Ranking และตาราง Virtual Money



รูปที่ 3.13: ตารางแสดงผลลัพธ์ของโครงการที่เข้าร่วมกิจกรรม

สร้างกิจกรรม (Create Event)

Gallery walk Dashboard Nuttaween Reibracing ▾

1 First step General information **2 Second step** Images/Dropzone **3 Final step** Check all information before submit

Event Manager Details

Event Name *

Start Date * for event	Start Time * for event	End Date * for event	End Time * for event
<input type="text" value="March 10, 2024"/>	<input type="time" value="--:--"/> ⓘ	<input type="text" value="End Date"/>	<input type="time" value="--:--"/> ⓘ

Description

B I U S T A E P
H H I L L
M X K
G¹ G²
[Link Icon] [Image Icon] [Video Icon]

Location

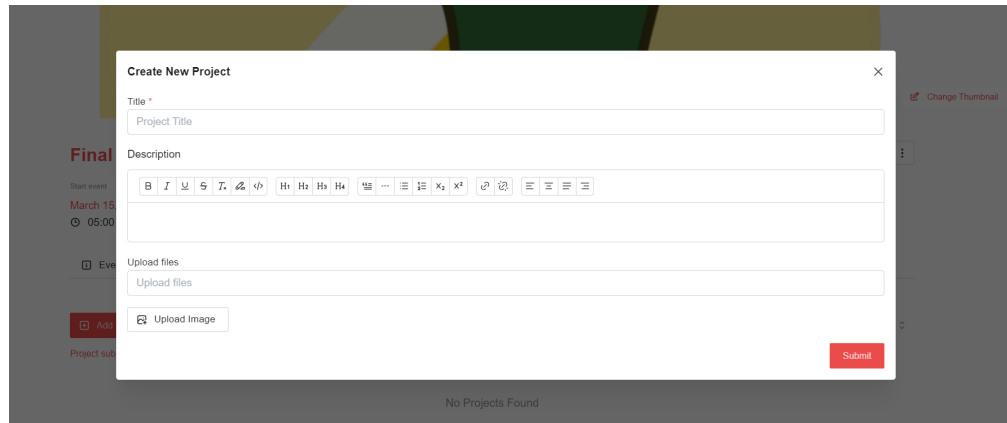
รูปที่ 3.14: หน้าสร้างกิจกรรม First step (Event information)

[illegible]

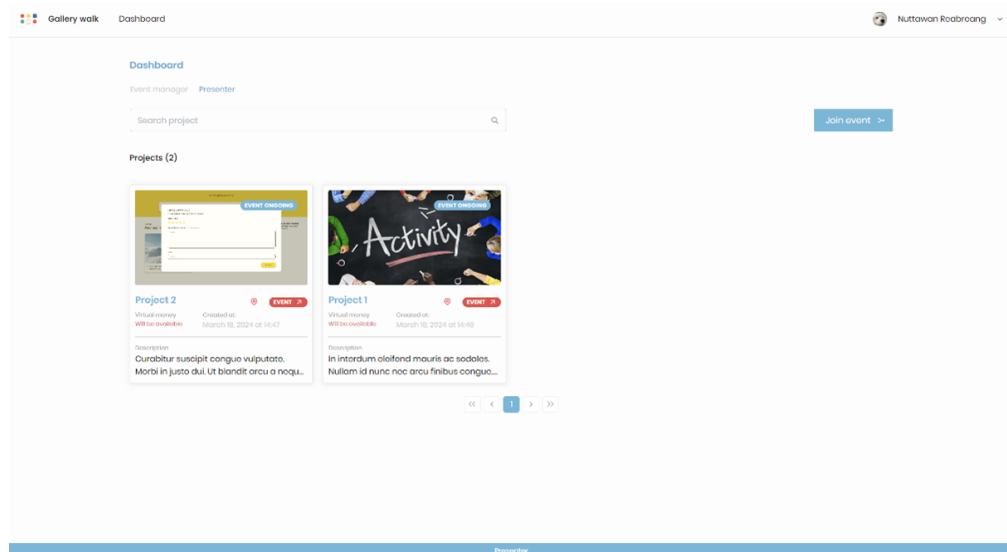
รูปที่ 3.15: หน้าสร้างกิจกรรม Second step (Images Dropzone)

3.2.4 ผู้นำเสนอผลงาน (Presenter)

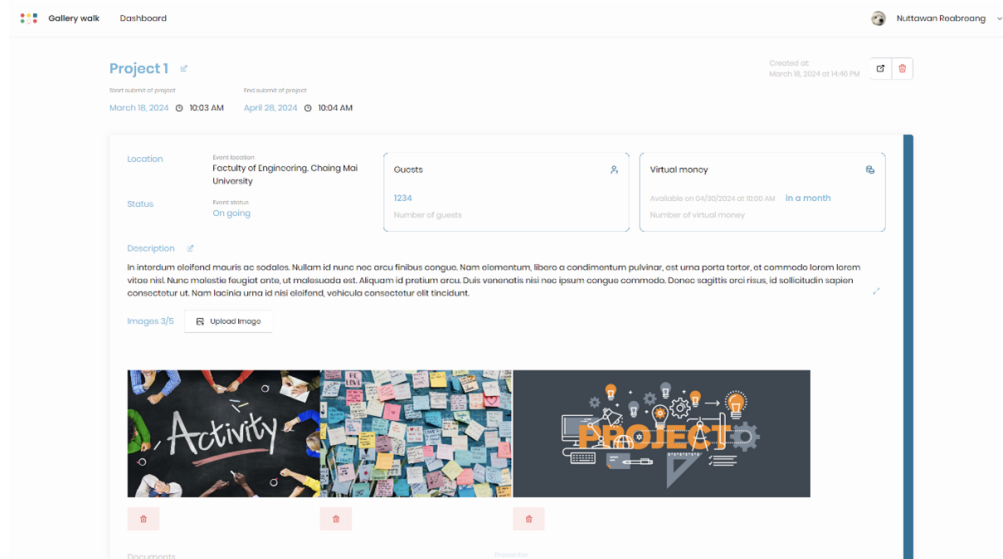
หลังจากที่เข้าสู่งานกิจกรรมจาก QR Code หรือ Access Link ที่ได้รับจาก Event manager จะเข้าสู่หน้าเข้าสู่ระบบ (Login) และหลังจากที่เข้าสู่ระบบจะเข้าสู่หน้าแสดงข้อมูลรายละเอียดของงานกิจกรรมที่จัดขึ้น และผู้นำเสนอโครงการสามารถเพิ่มโครงการเข้าร่วมงานกิจกรรมได้

A screenshot of a web application showing a 'Create New Project' modal form. The form has a title field labeled 'Title *' with a placeholder 'Project Title'. Below it is a 'Description' field with a rich text editor toolbar containing various icons for text formatting (bold, italic, underline, etc.) and alignment. There are two file upload sections: 'Upload files' with a text input and 'Upload Image' with a button icon. A red 'Submit' button is at the bottom right. The background shows a blurred view of the main dashboard with a 'No Projects Found' message at the bottom.

รูปที่ 3.17: หน้าเพิ่มโครงการที่เข้าร่วมกิจกรรม (Add project)



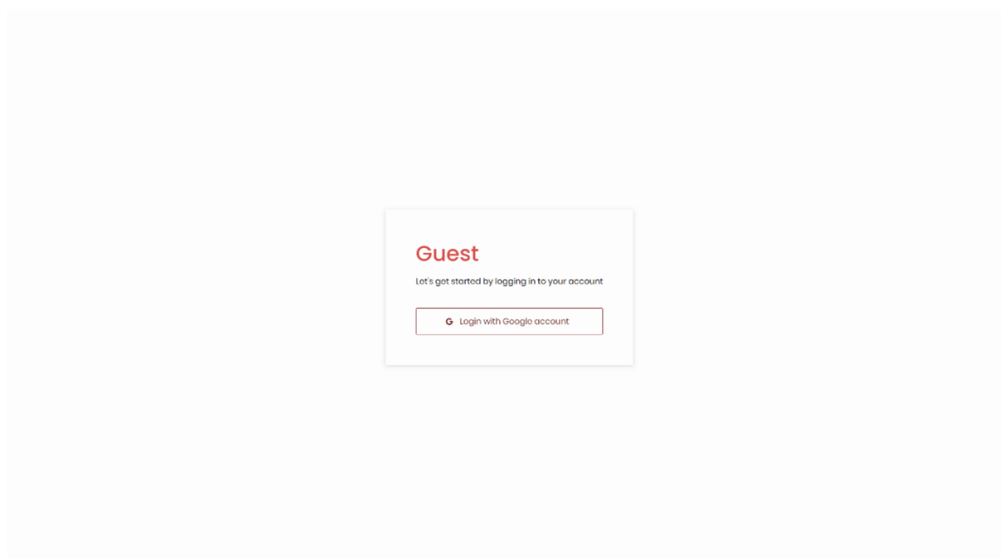
รูปที่ 3.18: หน้าแสดงโครงการที่สร้าง (Presenter dashboard)



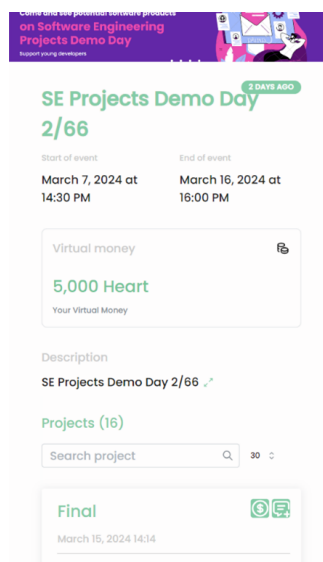
รูปที่ 3.19: หน้าแสดงข้อมูลรายละเอียดของผลงานที่สร้างในกิจกรรม

3.2.5 ผู้เข้าร่วมงาน (Guest)

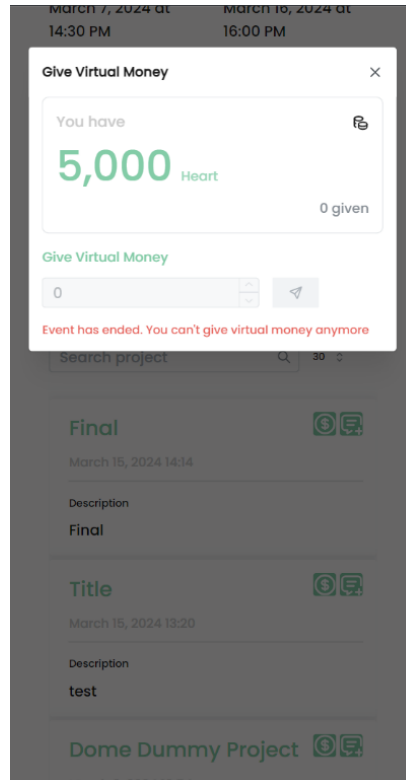
เมื่อเข้าสู่งานกิจกรรมจาก QR Code หรือ Access Link ที่ได้รับจาก Event manager จะเข้าสู่หน้าเข้าสู่ระบบ (Login) และหลังจากที่เข้าสู่ระบบจะเข้าสู่หน้าแสดงข้อมูลรายละเอียดของงานกิจกรรมที่จัดขึ้น และโครงการที่เข้าร่วมกิจกรรม



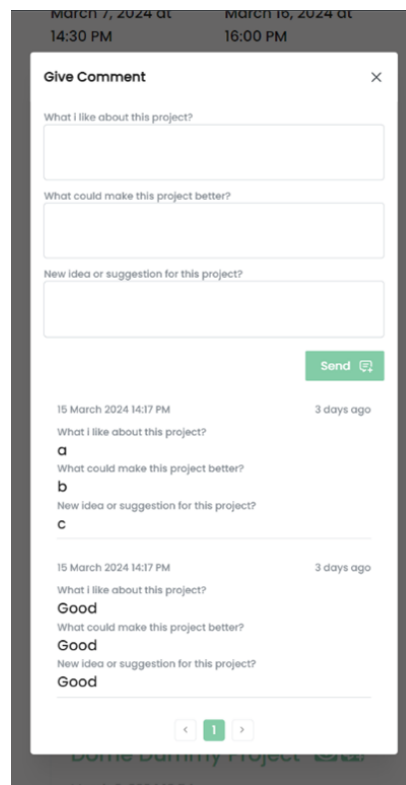
รูปที่ 3.20: หน้าเข้าสู่ระบบ (Login) สำหรับผู้เข้าร่วมกิจกรรม (Guest)



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 โดยการทดสอบกับผู้ใช้จริง ๆ โดยการให้ผู้ใช้ทดลองใช้งานจริง และให้คำแนะนำเพื่อปรับปรุงระบบให้ตรงตามความต้องการของผู้ใช้ ทั้ง 3 กลุ่มผู้ใช้งาน

โดยคณะได้นำเว็บแอปพลิเคชันนี้ไปทดลองใช้จริงในกระบวนวิชา 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

โดยการให้กลุ่มผู้ใช้อย่างน้อย 5 คนตอบแบบสำรวจผ่าน Google Form ประเมินความพอใจในด้านต่าง ๆ โดยแบ่งเป็น 2 ด้าน ได้แก่ การออกแบบและการใช้งาน (Design and Usability) และฟังก์ชันการทำงาน (Functionality) โดยแบ่งการ ประเมินออกเป็น 5 ระดับ

- 1. ไม่พอใจมาก (Very Dissatisfied) - หมายถึง ผู้ใช้งานไม่พอใจมากในข้อคำถามนั้น ๆ
- 2. พอใจน้อย (Somewhat Dissatisfied) - หมายถึง ผู้ใช้งานพอใจน้อยในข้อคำถามนั้น ๆ
- 3. พอใจปานกลาง (Neutral) - หมายถึง ผู้ใช้งานพอใจปานกลางในข้อคำถามนั้น ๆ
- 4. พอใจมาก (Somewhat Satisfied) - หมายถึง ผู้ใช้งานพอใจมากในข้อคำถามนั้น ๆ
- 5. พอใจมากที่สุด (Very Satisfied) - หมายถึง ผู้ใช้งานพอใจมากที่สุดข้อคำถามนั้น ๆ

panthon.f@gmail.com

สมัครบัญชี

ระบบจะป้อนรหัสในชื่อของคุณเมื่อส่งแบบฟอร์มนี้

* กรุณาเป็นค่าตามที่จำเป็น

หน้าสุดท้ายไป

เพศ *

☐ ชาย

☐ หญิง

☐ ไม่ต้องการระบุ

☐ อื่นๆ: _____

ท่านมีบทบาทใดในการใช้งาน Web Application นี้ *

☐ ผู้จัดการกิจกรรม (Event Manager)

☐ ผู้นำเสนอโครงการ (Presenter)

☐ ผู้เข้าร่วมกิจกรรม (Guest)

กลับ

ถัดไป

กำลังแนบไฟล์

ห้ามส่งรหัสผ่านให้ Google หรือ

ถ้าผู้ใช้ส่งรหัสผ่านไปยังระบบของ Google ระบบอาจขอเก็บ - ข้อมูลการใช้งานได้บ้าง - รายละเอียดเกี่ยวกับตัว

Google

ฟอรัม

รูปที่ 4.1: แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนข้อมูลทั่วไป

[illegible]

รูปที่ 4.2: แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนการออกแบบและใช้งาน (Design and Usability)

แบบสำรวจความพึงพอใจการใช้งาน Web Application: Gallery Walk

panthorn.f@gmail.com [คลิกที่นี่](#)

ระบบจะบันทึกข้อมูลของคุณเมื่อส่งแบบฟอร์มนี้

* กรุณาเป็นคำตอบที่จำเป็น

ฟังก์ชันการใช้งาน

โปรดเลือกตาม "ระดับความพึงพอใจ" ของงานเอกสารใช้งาน

ฟังก์ชันการทำงานครบถ้วนตามความต้องการหรือไม่ ? *

1 2 3 4 5

ไม่ครบถ้วนมาก ☐ ☐ ☐ ☐ ☐ ครบถ้วนมาก

ฟังก์ชันการทำงานใช้งานง่ายและเข้าใจง่ายหรือไม่ ? *

1 2 3 4 5

ใช้งานยาก ☐ ☐ ☐ ☐ ☐ ใช้งานง่ายมาก

มีข้อผิดพลาดหรือปัญหาในการใช้งานฟังก์ชันการทำงานใดบ้าง ? (ไม่ระบุ) *

คำขอของคุณ

ถึงแบบฟอร์ม

ดำเนินการโดย Google

เพื่อใช้ข้อมูลสำหรับการปรับปรุง Google Analytics - ข้อมูลจะถูกลบทิ้ง - ข้อมูลจะถูกลบทิ้ง - ข้อมูลจะถูกลบทิ้ง

Google

รูปที่ 4.3: แบบสำรวจความพึงพอใจของผู้ใช้ในส่วนฟังก์ชันการทำงาน (Functionality)

แบบสำรวจความพึงพอใจการใช้งาน Web Application: Gallery Walk

panthorn.f@gmail.com [คลิกที่นี่](#)

ระบบจะบันทึกข้อมูลของคุณเมื่อส่งแบบฟอร์มนี้

* กรุณาเป็นคำตอบที่จำเป็น

ความพึงพอใจโดยรวม

ระดับความพึงพอใจโดยรวมในการใช้งานเว็บไซต์หรือไม่ ? *

1 2 3 4 5

ไม่พึงพอใจมาก ☐ ☐ ☐ ☐ ☐ พึงพอใจมาก

มีข้อเสนอแนะเพิ่มเติมหรือไม่?

คำขอของคุณ

ถึงแบบฟอร์ม

ดำเนินการโดย Google

เพื่อใช้ข้อมูลสำหรับการปรับปรุง Google Analytics - ข้อมูลจะถูกลบทิ้ง - ข้อมูลจะถูกลบทิ้ง - ข้อมูลจะถูกลบทิ้ง

Google

รูปที่ 4.4: แบบสำรวจความพึงพอใจของผู้ใช้ในความพึงพอใจโดยรวม

4.3 สรุปผลการประเมินความพึงพอใจของผู้ใช้

จากการประเมินความพึงพอใจของผู้ใช้ พบว่าผู้ใช้งานทั้ง 3 กลุ่ม คือ ผู้สร้างงานกิจกรรม (Event Manager) ผู้นำเสนอโครงการ (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
- ผู้นำเสนอโครงการ (Presenter)
 - ต้องการให้มีการแจ้งเตือนเมื่อมีผู้มาให้ Virtual Money หรือแสดงความคิดเห็นในโครงการของตน
 - ต้องการให้ระบบสามารถเพิ่มผู้จัดการโครงการ (Project Manager) คนอื่นในโครงการของตน
- แขกผู้เข้าร่วมงาน (Guest)
 - ต้องการให้การแสดงผลโครงการต่าง ๆ ชัดเจนและสามารถกรองโครงการต่าง ๆ ได้ง่าย เช่น กรองจากหมวดหมู่ อักษรตัวแรก หรือจาก Virtual Money ที่ได้รับ และอื่น ๆ

บทที่ 5

บทสรุปและข้อเสนอแนะ

5.1 สรุปผล

การทำโครงการนี้สามารถพัฒนาเว็บแอปพลิเคชันที่สามารถใช้งานตามความต้องการของผู้ใช้งานได้ โดยผู้ใช้งานทั้ง 3 กลุ่ม คือ

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

5.2 ปัญหาที่พบและแนวทางการแก้ไข

ในการทำโครงการนี้ พบว่าเกิดปัญหาหลัก ๆ ดังนี้

- การทำงานร่วมกันของทีม ทีมมีคนทำงานร่วมกัน 3 คน แต่เนื่องจากการประชุมและการทำงานร่วมกันไม่สม่ำเสมอ ทำให้การทำงานมีความล่าช้าอยู่บ้าง
- การทดสอบระบบ การทดสอบระบบระหว่างการพัฒนาไม่ครอบคลุม ทำให้มีบางส่วนของระบบที่ไม่ทำงานอย่างที่ควรจะเป็น
- การแสดงผลข้อมูล การแสดงผลข้อมูลตามกลุ่มผู้ใช้งานไม่ครอบคลุม ทำให้ผู้ใช้งานบางกลุ่มไม่สามารถใช้งานระบบได้อย่างที่ควรจะเป็น

5.3 ข้อเสนอแนะและแนวทางการพัฒนาต่อ

ข้อเสนอแนะเพื่อพัฒนาโครงการนี้ต่อไป มีดังนี้

ข้อเสนอที่ได้รับจากแบบสำรวจความพึงพอใจของผู้ใช้งานและผู้พัฒนาระบบ ได้แก่

- พัฒนาส่วนแดชบอร์ดสำหรับผู้ใช้งานที่เป็นผู้สร้างงานกิจกรรม (Event Manager) ให้มีการแสดงผลข้อมูลที่ครอบคลุมมากขึ้นในแต่ละกิจกรรม ซึ่งอาจประกอบไปด้วย จำนวนผู้เข้าร่วมงาน จำนวนโครงการที่เข้าร่วม จำนวน Virtual Money ที่แต่ละโครงการได้รับ แขกผู้เข้าร่วมงานให้ Virtual Money และแสดงความคิดเห็นในโครงการต่าง ๆ
- พัฒนาการแจ้งเตือนให้ผู้ใช้ที่เป็นผู้นำเสนอโครงการ (Presenter) ให้มีการแจ้งเตือนเมื่อโครงการของตนได้รับ Virtual Money หรือมีความคิดเห็นใหม่ ๆ
- ในส่วนของการเพิ่มโครงการเข้ามาในงานกิจกรรม ให้มีการตรวจสอบและอนุมัติ (Approve) โครงการจากผู้สร้างงานกิจกรรมก่อนที่จะแสดงผลในงานกิจกรรม

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 เอชทีเอ็มแอล 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/ทำไมถึงต้องใช้-typescript-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>. (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/>