



โครงการ

การเรียนการสอนเพื่อเสริมประสบการณ์

ชื่อโครงการ แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง

(ระบบส่วนหน้าบ้าน)

Construction process quality control assisted application

(Front-end)

ชื่อนิสิต นางสาวศิตาภัทร วงษาเนาวิ 5933607723

นางสาวศิรดา ทาบสุวรรณ 5933654123

ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์

สาขาวิชาวิทยาการคอมพิวเตอร์

ปีการศึกษา 2562

คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง

นางสาวศิตาภัทร วงษาเนาวิ

นางสาวศิรดา ทาบสุวรรณ

โครงการนี้เป็นส่วนหนึ่งของวิชาการศึกษิตตามหลักสูตรวิทยาศาสตร์บัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2562

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Construction process quality control assisted application

Keetapat Wongsanao

Sirada Thabsuwan

A Project submitted in partial fulfillment of the requirements

For the degree of bachelor of science program in computer science

Department of mathematics and computer science

Faculty of science

Chulalongkorn University

Academic Year 2019

Copyright of Chulalongkorn University

บทคัดย่อภาษาไทย

ปัจจุบันบริษัทอุตสาหกรรมก่อสร้างต่าง ๆ ใช้การสื่อสารผ่านแอปพลิเคชัน LINE เป็นหลัก และเนื่องจากการสื่อสารผ่านแอปพลิเคชัน LINE เป็นการสื่อสารที่เป็นกลุ่มขนาดใหญ่อาจเกิดปัญหาที่ตามมาได้ ซึ่งปัญหาส่วนใหญ่ที่มักเกิดได้แก่ การเกิดปัญหาในการอ่านข้อความแล้วมีสื่อสารผิดบุคคล หรือการสื่อสารแล้วผู้รับสารไม่ได้รับสาร ไม่มีบุคคลตอบรับสารหรืองานจากผู้ส่งสาร

ผู้จัดทำจึงได้เล็งเห็นถึงปัญหาดังกล่าว และพัฒนาแอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง ในระบบส่วนหน้าบ้าน เพื่อที่จะแก้ปัญหาการสื่อสารที่ผิดพลาดต่าง ๆ โดยผู้จัดทำมีการเก็บความต้องการของผู้ใช้งานจากบริษัทอุตสาหกรรมต่าง ๆ จำนวน 5 ครั้ง โดยแบ่งออกเป็น สำนักวิจัยและพัฒนาทางกรมทางหลวง 1 ครั้ง บริษัท Team built CO., LTD. 2 ครั้ง และ บริษัท ปรีญสิริ จำกัด (มหาชน) 2 ครั้ง และนำผลสรุปของความต้องการของผู้ใช้งานทั้งหมดมาวิเคราะห์ฟังก์ชันที่เป็นประโยชน์ต่อผู้ใช้งาน

เมื่อได้ผลสรุปความต้องการของผู้ใช้งานแล้ว ผู้จัดทำจะออกแบบส่วนต่อประสานผู้ใช้ด้วยโปรแกรม Adobe XD เพื่อให้ผู้ใช้งานเห็นภาพ Prototype ของแอปพลิเคชัน และนำ Prototype มาติดตามข้อเสนอแนะกับผู้ใช้งานจริงอีกครั้ง เมื่อได้ Prototype ที่ปรับแก้สมบูรณ์แล้ว จากนั้นจะนำ Prototype มาเป็นแบบในการเขียนแอปพลิเคชัน โดยแอปพลิเคชันเขียนในรูปของภาษาจาวาสคริปต์ด้วยโปรแกรม VSCode ผ่าน Framework ที่มีชื่อว่า React Native และ Expo ซึ่งเป็น Framework สำหรับภาษาจาวาสคริปต์ ที่สามารถทำให้แอปพลิเคชัน ใช้งานได้ทั้งระบบปฏิบัติการ IOS และ Android

ภายหลังจากการพัฒนาแอปพลิเคชัน ผู้จัดทำยังมีการติดตามอัปเดตความคืบหน้า และผลตอบรับกับผู้ใช้งานจริง โดยมีการนำแอปพลิเคชันไปให้ผู้ใช้งานทำการทดสอบจริง ซึ่งผู้ใช้งานจริงประกอบไปด้วยตำแหน่ง Project Manger และ Foreman จากบริษัท Team built CO., LTD. และ บริษัท ปรีญสิริ จำกัด (มหาชน) จำนวนทั้งสิ้น 13 คน ซึ่งผลจากการทดสอบ ได้ผลตอบรับที่น่าพึงพอใจ

บทคัดย่อภาษาอังกฤษ

Presently, the communication via the application such as LINE has been widely used—not only among individuals, but also in large scale organization. Although the benefits are noticeably seen, the communication via these application and internet still have some relevant limitations. For instance, miscommunication, misunderstanding the context, mis-send to the targeted person, and even ignorance/negligence of the received context.

Accordingly, we have realized the problems and intended to develop the “Construction process quality control assisted application (front end)” to solve various underlying communication issues. Requirement was gathered from users of three main industrial companies for 5 times including 1 visit at Department of Highways, 2 visits at Team built Co., Ltd., and 2 visits at Prinsiri Co., Ltd. The received information was then analyzed for appropriate features that are beneficial to users.

After summarizing the requirements, user interface was designed using Adobe XD in order to create the application’s prototype and receive users’ feedback for better improvement before developing the application. Upon developing, JavaScript with VScode through the framework called React Native and Expo is used due to high compatibility in both iOS and Android operating systems.

Upon developing, users’ feedback and needs are constantly updated and utilized to best improve the application. User testing was then executed with 13 real users—project manager and foreman positions—from Team built Co., Ltd and Prinsiri Co., Ltd., whom have claimed great satisfaction.

หัวข้อโครงการ แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง
(ระบบส่วนหน้าบ้าน)

โดย นางสาวศีตาภัทร วงษาเนา
นางสาวศิรดา ทาบสุวรรณ

สาขาวิชา วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษาโครงการหลัก รองศาสตราจารย์ ดร.นกุล คูหะโรจนานนท์

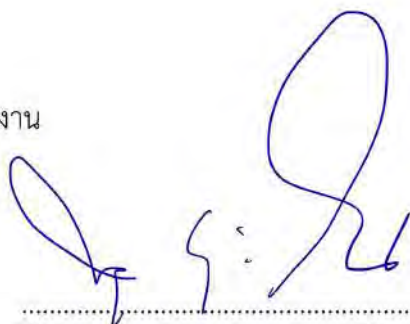
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้
นับโครงการฉบับนี้เป็นส่วนหนึ่ง ของการศึกษาตามหลักสูตรปริญญาบัณฑิต ในรายวิชา 2301499 โครงการ
วิทยาศาสตร์ (Senior Project)



..... หัวหน้าภาควิชาคณิตศาสตร์

(ศาสตราจารย์ ดร. กฤษณะ เนียมมณี) และวิทยาการคอมพิวเตอร์

คณะกรรมการสอบโครงการ



..... อาจารย์ที่ปรึกษาโครงการหลัก

(รองศาสตราจารย์ ดร.นกุล คูหะโรจนานนท์)



..... กรรมการ

(รองศาสตราจารย์ ดร.พีระพนธ์ โสพิศสถิตย์)



..... กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. สมใจ บุญศิริ)

คีตาภัทร วงษาเนา, ศิริดา ทาบสุวรรณ: แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง (ระบบส่วนหน้าบ้าน). Construction process quality control assisted application (Front-end) อ.ที่ปรึกษาโครงการหลัก: รองศาสตราจารย์ ดร. นกุล คูหะโรจนานนท์, 115 หน้า.

โครงการเรื่อง “แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง” จัดทำขึ้นเพื่อจะมาช่วยเหลือในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง เพื่อลดระยะเวลาในการสื่อสาร และลดความผิดพลาดจากการสื่อสารในกระบวนการก่อสร้าง โดยแอปพลิเคชันที่พัฒนาจะสามารถแก้ปัญหาในการสื่อสารของกระบวนการก่อสร้าง ให้การสื่อสารมีแบบแผนมากขึ้น โดยแอปพลิเคชันพัฒนาด้วยภาษาจาวาสคริปต์ด้วยเฟรมเวิร์ค (Framework) ที่มีชื่อว่า รีแอคเนทีฟ (React Native) และ เอ็กซ์โป (Expo) โดยสามารถใช้งานได้บนโทรศัพท์มือถือที่เป็นระบบปฏิบัติการ IOS และ Android โดยผู้ใช้งานจำเป็นต้องลงทะเบียนก่อนถึงจะสามารถเข้าใช้งานได้ มีฟังก์ชันหลัก ๆ ได้แก่ การจัดการโครงการ การจัดการโปรเจค การจัดการงาน และการแสดงความคิดเห็นในแต่ละปัญหาของงาน การแจ้งเตือนเพื่อให้สามารถติดตามการอัปเดตได้อย่างทันทั่วถึง อีกทั้งยังสามารถแนบรูปไปกับการสร้างงานแต่ละอย่างได้อีกด้วย ผลการสำรวจ และประเมินความคิดเห็นจากบริษัทปริยสุริ และบริษัท Team Built พบว่าระบบนี้สามารถทำให้การสื่อสารในโครงการก่อสร้างมีความรวดเร็วมากขึ้น สามารถติดตามความคืบหน้าของโครงการได้อย่างทันทั่วถึง และสามารถดูข้อมูลภายในโครงการได้อย่างทันทั่วถึงอีกด้วย โดยคะแนนประเมินของการทดสอบอยู่ที่ 4.03 คะแนน (จากคะแนนเต็ม 5 คะแนน)

ภาควิชา...คณิตศาสตร์และวิทยาการคอมพิวเตอร์.....ลายมือชื่อนิสิต.....

คีตาภัทร

ลายมือชื่อนิสิต.....ศิริดา ทาบสุวรรณ.....

สาขาวิชา.....วิทยาการคอมพิวเตอร์..... ลายมือชื่อ อ.ที่ปรึกษาโครงการหลัก.....

ปีการศึกษา.....2561.....

กิตติกรรมประกาศ

การจัดทำโครงการ แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้างสำเร็จ ล่วงไปด้วยดี ทั้งนี้ได้เพราะได้รับความอนุเคราะห์และความช่วยเหลือจากคณาจารย์และบุคลากรต่าง ๆ หลายน ท่าน

ขอขอบพระคุณ รองศาสตราจารย์ ดร.นกุล คูหะโรจนานนท์ อาจารย์ที่ปรึกษาโครงการ ที่ให้คำปรึกษา และข้อเสนอแนะทางวิชาการ ทั้งยังสละเวลาอันมีค่าชี้แนะแก้ไขตลอดทั้งโครงการ

ขอขอบพระคุณคณะกรรมการสอบทั้ง 2 ท่านได้แก่ รองศาสตราจารย์ ดร.พีระพันธ์ โสพัศสถิตย์ และ ผู้ช่วยศาสตราจารย์ ดร.สมใจ บุญศิริ ที่ให้คำแนะนำข้อเสนอแนะทำให้เกิดแนวทางในการพัฒนาให้โครงการนี้มีความสมบูรณ์มากยิ่งขึ้น

ขอบคุณที่ ๆ ที่บริษัท Solitech Innovantion Co., Ltd. สำหรับการให้คำแนะนำและคำปรึกษายาม เมื่อเกิดปัญหาในการติดต่อกับผู้ใช้งาน เพื่อให้งานได้พัฒนาได้อย่างสมบูรณ์ยิ่งขึ้น

ขอบคุณ ที่ ๆ ผู้จัดการโปรเจก และ หัวหน้าคนงาน ที่เป็นตัวแทนบริษัทก่อสร้างที่ได้ให้การสัมภาษณ์ และตอบแบบสอบถาม เพื่อที่จะได้นำมาพัฒนาแอปพลิเคชันให้สมบูรณ์ยิ่งขึ้น

ขอขอบคุณเพื่อนทุกคน พี่ศิษย์เก่าที่คอยช่วยเหลือ ให้คำปรึกษายามเมื่อเกิดปัญหาและยินดีรับฟัง ช่วยเหลือเสมอมา ทำให้โครงการนี้ประสบความสำเร็จ

ท้ายที่สุดนี้ขอขอบคุณความกรุณาอันดีจากทุกท่านที่ได้กล่าวนามไว้ข้างต้น รวมถึงบุคคลท่านอื่นที่ไม่ได้กล่าวนามไว้ ณ ที่นี้ด้วย สำหรับความช่วยเหลือและคำแนะนำต่าง ๆ ที่คอยผลักดันให้โครงการนี้ประสบผลสำเร็จไปได้ด้วยดี

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	จ
สารบัญ.....	ฉ
สารบัญตาราง.....	ช
สารบัญรูป.....	ฌ
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและเหตุผล.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตของโครงการ.....	2
1.4 ขั้นตอนการดำเนินงาน.....	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	5
1.6 โครงสร้างของรายงาน.....	5
บทที่ 2 งานวิจัยที่เกี่ยวข้อง.....	6
2.1 การวิเคราะห์การสื่อสารภายในองค์กรธุรกิจอุตสาหกรรม.....	6
2.2 ผลสัมฤทธิ์.....	7
2.3 ภาษาจาวาสคริปต์.....	7
บทที่ 3 การวิเคราะห์และออกแบบ.....	19
3.1 บทนำ.....	19
3.2 ข้อกำหนดความต้องการด้านซอฟต์แวร์.....	21
3.3 ความต้องการเชิงหน้าที่.....	21
3.4 ความต้องการเชิงคุณภาพ.....	43
3.5 การออกแบบส่วนต่อประสานผู้ใช้.....	44
บทที่ 4 การพัฒนาและทดสอบระบบ.....	50
4.1 การพัฒนาระบบ.....	50
4.2 การทดสอบระบบ.....	74

บทที่ 5 ข้อเสนอแนะ.....	87
5.1 สรุปผล.....	87
5.2 ปัญหาและอุปสรรค.....	87
5.3 วิธีการแก้ปัญหา.....	87
5.4 ข้อเสนอแนะในการพัฒนาระบบ.....	88
ภาคผนวก.....	89
ภาคผนวก ก แบบเสนอหัวข้อโครงการ รายวิชา 2301399 Project Proposal ปีการศึกษา 2562.....	90
ภาคผนวก ข คู่มือการใช้งาน.....	95
ภาคผนวก ค เอกสารอ้างอิง.....	102
ประวัติผู้เขียน.....	103

สารบัญตาราง

	หน้า
ตารางที่ 1.3.1 แสดงข้อมูลความต้องการของผู้ใช้งานจำแนกตามบริษัท.....	3
ตารางที่ 1.4.1 ขั้นตอนการดำเนินงาน.....	4
ตารางที่ 3.3.1 ตารางแสดงการ Register.....	23
ตารางที่ 3.3.2 ตารางแสดงการ Login.....	23
ตารางที่ 3.3.3 ตารางแสดง Add Floorplan.....	24
ตารางที่ 3.3.4 ตารางแสดงการ Send Email.....	25
ตารางที่ 3.3.5 ตารางแสดงการ Edit Foreman.....	26
ตารางที่ 3.3.6 ตารางแสดงการ Manage Site.....	26
ตารางที่ 3.3.7 ตารางแสดงการ Manage Project.....	27
ตารางที่ 3.3.8 ตารางแสดงการ Manage Task.....	28
ตารางที่ 3.3.9 ตารางแสดงการ Manage Subtask.....	29
ตารางที่ 3.3.10 ตารางแสดงการ Comment.....	29
ตารางที่ 3.3.11 ตารางแสดง Notification.....	30
ตารางที่ 3.3.12 ตารางแสดง Profile.....	30
ตารางที่ 3.3.13 ตารางแสดงการ Logout.....	31
ตารางที่ 3.3.14 ตารางแสดงความต้องการเชิงหน้าที่.....	32
ตารางที่ 3.3.15 ตารางเปรียบเทียบสิทธิ์ในการใช้งานแต่ละฟังก์ชันภายในระบบของแต่ละผู้ใช้งาน.....	42
ตารางที่ 3.4.1 ตารางแสดงความต้องการเชิงคุณภาพ.....	43
ตารางที่ 4.1.1 ตารางแสดงหน้าจอเปรียบเทียบกับความต้องการเชิงหน้าที่.....	51
ตารางที่ 4.2.1.1 ตารางข้อเสนอแนะ.....	74
ตารางที่ 4.2.2.1 ตารางกรณีทดสอบและผลการทดสอบโดยผู้พัฒนาแอปพลิเคชัน.....	77
ตารางที่ 4.2.3.1 ผลการทดสอบการทดลองใช้งานจากผู้ใช้งานจริง.....	84
ตารางที่ 4.2.3.2 สรุปผลการทดลองใช้งานจากผู้ใช้งานจริง.....	85
ตารางที่ 5.4.1 ตารางเปรียบเทียบแอปพลิเคชัน และโปรแกรมอื่น ๆ.....	88

สารบัญรูป

หน้า

รูปที่ 2.3.1.1 แสดงโค้ดสำหรับการประกาศตัวแปรภายในภาษาจาวาสคริปต์.....	7
รูปที่ 2.3.1.2 แสดงโค้ดสำหรับการใช้ Array ภายในภาษาจาวาสคริปต์.....	8
รูปที่ 2.3.1.3 แสดงโค้ดสำหรับการใช้ Shift ภายในภาษาจาวาสคริปต์.....	8
รูปที่ 2.3.1.4 แสดงโค้ดสำหรับการใช้ pop ภายในภาษาจาวาสคริปต์.....	8
รูปที่ 2.3.1.5 แสดงโค้ดสำหรับการใช้ push ภายในภาษาจาวาสคริปต์.....	9
รูปที่ 2.3.1.6 แสดงโค้ดสำหรับการใช้ concat ภายในภาษาจาวาสคริปต์.....	9
รูปที่ 2.3.1.7 แสดงโค้ดสำหรับการใช้ฟังก์ชันภายในภาษาจาวาสคริปต์.....	9
รูปที่ 2.3.1.8 แสดงโค้ดสำหรับการใช้ console ภายในภาษาจาวาสคริปต์.....	10
รูปที่ 2.3.1.9 แสดงโค้ดสำหรับการใช้ this ภายในภาษาจาวาสคริปต์.....	10
รูปที่ 2.3.1.10 แสดงโค้ดสำหรับการใช้ bind ภายในภาษาจาวาสคริปต์.....	10
รูปที่ 2.3.1.11 แสดงโค้ดสำหรับการใช้ bind ใน callback ภายในภาษาจาวาสคริปต์.....	11
รูปที่ 2.3.1.12 แสดงโค้ดสำหรับการใช้ apply ภายในภาษาจาวาสคริปต์.....	11
รูปที่ 2.3.1.13 แสดงโค้ดสำหรับการใช้ For of ภายในภาษาจาวาสคริปต์.....	12
รูปที่ 2.3.1.14 แสดงโค้ดสำหรับการใช้ Module ภายในภาษาจาวาสคริปต์.....	12
รูปที่ 2.3.1.15 แสดงโค้ดสำหรับการใช้ Promise ภายในภาษาจาวาสคริปต์.....	13
รูปที่ 2.5.1.1 แสดงหน้าจอแอปพลิเคชัน Plangrid.....	15
รูปที่ 2.5.1.2 แสดงหน้าจอแอปพลิเคชัน Plangrid.....	16
รูปที่ 2.5.1.3 แสดงหน้าจอแอปพลิเคชัน Plangrid.....	16
รูปที่ 2.5.2.1 แสดงหน้าจอแอปพลิเคชัน Sitewalk.....	17
รูปที่ 2.5.2.2 แสดงหน้าจอแอปพลิเคชัน Sitewalk.....	18
รูปที่ 2.5.2.3 แสดงหน้าจอแอปพลิเคชัน Plangrid.....	18
รูปที่ 3.1.1.1 รวบรวมความต้องการที่สำนักวิจัยและพัฒนาทาง กรมทางหลวง.....	20
รูปที่ 3.1.1.2 รวบรวมความต้องการที่ บริษัท Team built CO.,LTD.	20
รูปที่ 3.1.1.3 รวบรวมความต้องการที่ บริษัท ปรีญสิริ จำกัด (มหาชน)	20
รูปที่ 3.3.1 แผนภาพยูสเคสแสดงหน้าที่การทำงานของระบบการช่วยตรวจสอบคุณภาพ ภายในกระบวนการก่อสร้าง.....	22

สารบัญรูป

หน้า

รูปที่ 3.3.1.1 แผนภาพกิจกรรมในส่วนของการเข้าสู่ระบบ.....	31
รูปที่ 3.3.1.2 แผนภาพกิจกรรมในส่วนของการเพิ่มโครงการ.....	31
รูปที่ 3.3.1.3 แผนภาพกิจกรรมในส่วนของการกระทำภายในโปรเจค.....	32
รูปที่ 3.3.1.4 แผนภาพกิจกรรมในส่วนของการกระทำภายในงาน.....	33
รูปที่ 3.3.1.5 แผนภาพกิจกรรมในส่วนของการเพิ่มงานย่อย.....	34
รูปที่ 3.3.1.6 แผนภาพกิจกรรมในส่วนของการแสดงความคิดเห็น.....	34
รูปที่ 3.3.1.7 แผนภาพกิจกรรมในส่วนของการส่งจดหมายอิเล็กทรอนิกส์.....	35
รูปที่ 3.3.2.1 แผนภาพลำดับในส่วนของการเข้าสู่ระบบ.....	35
รูปที่ 3.3.2.2 แผนภาพลำดับในส่วนของการลงทะเบียนเข้าสู่ระบบ.....	36
รูปที่ 3.3.2.3 แผนภาพลำดับในส่วนของการกระทำภายในโปรเจค.....	37
รูปที่ 3.3.2.4 แผนภาพลำดับในส่วนของการกระทำภายในงาน.....	38
รูปที่ 3.3.2.5 แผนภาพลำดับในส่วนของการเพิ่มงานย่อย.....	38
รูปที่ 3.3.2.6 แผนภาพลำดับในส่วนของการแสดงความคิดเห็น.....	39
รูปที่ 3.3.2.7 แผนภาพลำดับในส่วนของการแก้ไขในส่วนของรายชื่อหัวหน้าคนงาน.....	39
รูปที่ 3.5.1 ส่วนต่อประสานผู้ใช้หน้าลงทะเบียน.....	44
รูปที่ 3.5.2 ส่วนต่อประสานผู้ใช้หน้าเข้าสู่ระบบ.....	44
รูปที่ 3.5.3 ส่วนต่อประสานผู้ใช้หน้าแปลน.....	45
รูปที่ 3.5.4 ส่วนต่อประสานผู้ใช้หน้าอีเมล.....	45
รูปที่ 3.5.5 ส่วนต่อประสานผู้ใช้หน้าแก้ไขรายชื่อหัวหน้าคนงาน.....	46
รูปที่ 3.5.6 ส่วนต่อประสานผู้ใช้หน้าโครงการทั้งหมด.....	46
รูปที่ 3.5.7 ส่วนต่อประสานผู้ใช้หน้าโครงการ.....	47
รูปที่ 3.5.8 ส่วนต่อประสานผู้ใช้หน้าโปรเจค.....	47
รูปที่ 3.5.9 ส่วนต่อประสานผู้ใช้หน้างาน.....	48
รูปที่ 3.5.10 ส่วนต่อประสานผู้ใช้หน้าปัญหา.....	48
รูปที่ 3.5.11 ส่วนต่อประสานผู้ใช้หน้าการแจ้งเตือน.....	49
รูปที่ 3.5.12 ส่วนต่อประสานผู้ใช้หน้าการตั้งค่า.....	49

สารบัญรูป

	หน้า
รูปที่ 3.5.13 ส่วนต่อประสานผู้ใช้หน้าการออกจากระบบ.....	49
รูปที่ 4.1.1 ภาพรวมของระบบ.....	50
รูปที่ 4.1.2 แผนภาพแสดงโครงสร้างหน้าจอของแอปพลิเคชัน.....	52
รูปที่ 4.1.1.1 ภาษาจาวาสคริปต์ หน้าการลงทะเบียน.....	53
รูปที่ 4.1.1.2 หน้าจอการลงทะเบียน.....	53
รูปที่ 4.1.1.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าการลงทะเบียน.....	54
รูปที่ 4.1.2.1 ภาษาจาวาสคริปต์ หน้าเข้าสู่ระบบ.....	55
รูปที่ 4.1.2.2 หน้าจอเข้าสู่ระบบ.....	55
รูปที่ 4.1.2.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าเข้าสู่ระบบ.....	56
รูปที่ 4.1.3.1 ภาษาจาวาสคริปต์ หน้าแปลน.....	57
รูปที่ 4.1.3.2 หน้าจอแปลน.....	57
รูปที่ 4.1.3.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าแปลน.....	57
รูปที่ 4.1.4.1 ภาษาจาวาสคริปต์ หน้าอีเมล.....	58
รูปที่ 4.1.4.2 หน้าจออีเมล.....	58
รูปที่ 4.1.5.1 ภาษาจาวาสคริปต์ หน้าแก้ไข Foreman.....	59
รูปที่ 4.1.5.2 หน้าจอแก้ไข Foreman.....	59
รูปที่ 4.1.5.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าแก้ไข Foreman.....	59
รูปที่ 4.1.6.1 ภาษาจาวาสคริปต์ หน้าโครงการทั้งหมด.....	60
รูปที่ 4.1.6.2 หน้าจอโครงการทั้งหมด.....	60
รูปที่ 4.1.6.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าโครงการทั้งหมด.....	60
รูปที่ 4.1.7.1 ตัวอย่างภาษาจาวาสคริปต์ หน้าโครงการ.....	62
รูปที่ 4.1.7.2 หน้าจอหน้าโครงการ.....	62
รูปที่ 4.1.7.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าโครงการ.....	63
รูปที่ 4.1.8.1 ภาษาจาวาสคริปต์ หน้าโปรเจค.....	64
รูปที่ 4.1.8.2 หน้าจอโปรเจค.....	64
รูปที่ 4.1.8.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าโปรเจค.....	65

สารบัญรูป

	หน้า
รูปที่ 4.1.9.1 ภาษาจาวาสคริปต์ หน้านำงาน.....	66
รูปที่ 4.1.9.2 หน้าที่องงาน.....	66
รูปที่ 4.1.9.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้านำงาน.....	67
รูปที่ 4.1.10.1 ภาษาจาวาสคริปต์ หน้าที่ปัญหา.....	68
รูปที่ 4.1.10.2 หน้าที่ปัญหา.....	68
รูปที่ 4.1.10.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าที่ปัญหา.....	69
รูปที่ 4.1.11.1 ภาษาจาวาสคริปต์ หน้าที่การแจ้งเตือน.....	70
รูปที่ 4.1.11.2 หน้าที่การแจ้งเตือน.....	70
รูปที่ 4.1.11.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าที่การแจ้งเตือน.....	70
รูปที่ 4.1.12.1 ภาษาจาวาสคริปต์ หน้าที่โปรไฟล์.....	71
รูปที่ 4.1.12.2 หน้าที่โปรไฟล์.....	71
รูปที่ 4.1.12.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าที่โปรไฟล์.....	72
รูปที่ 4.1.13.1 ภาษาจาวาสคริปต์ หน้าที่การตั้งค่า.....	73
รูปที่ 4.1.13.2 หน้าที่การตั้งค่า.....	73
รูปที่ 4.1.13.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าที่การตั้งค่า.....	74

บทที่ 1

บทนำ

ในบทนี้จะกล่าวถึง ความเป็นมาและเหตุผล วัตถุประสงค์ ขอบเขต ขั้นตอน และ ประโยชน์ที่คาดว่าจะได้รับ ของการพัฒนาแอปพลิเคชัน

1.1 ความเป็นมาและเหตุผล

ในปัจจุบันมีการสื่อสารกันผ่านอินเทอร์เน็ตกันมากขึ้นตั้งแต่การสื่อสารระหว่างบุคคล การสื่อสารภายในองค์กรและบริษัทอื่น ๆ รวมไปถึงบริษัทอุตสาหกรรมก่อสร้างต่าง ๆ ที่ใช้การสื่อสารในการช่วยตรวจสอบคุณภาพของกระบวนการในการก่อสร้าง ซึ่งภายในองค์กรนั้นมีการสื่อสาร ระหว่างองค์กรผ่านอินเทอร์เน็ตและใช้แอปพลิเคชันในการสื่อสารกัน เป็นแอปพลิเคชันที่ใช้กันอย่างแพร่หลาย อย่างเช่น แอปพลิเคชัน LINE [1] เนื่องจากการสื่อสารผ่านทางแอปพลิเคชัน LINE เป็นการสื่อสารที่เป็นกลุ่มใหญ่อาจจะเกิดปัญหาได้ และปัญหาส่วนใหญ่ที่มักเกิดได้แก่ การเกิดปัญหาในการอ่านข้อความแล้วมีการสื่อสารผิดบุคคล เนื่องจากการสื่อสารภายในกลุ่มใหญ่ ทำให้เวลาต้องการส่งสารเกิดความเข้าใจผิดระหว่าง ฝ่ายได้ [2] สื่อสารแล้วผู้รับสารไม่ได้รับสาร ไม่มีบุคคลตอบรับสารหรืองานจากผู้ส่งสาร เนื่องจากมีคนสื่อสาร หลายเรื่องในการสื่อสารภายในกลุ่ม LINE สื่อสารไม่ครบตามเนื้อหาสารที่ต้องการจะสื่อสาร จากที่ได้รับโจทย์จากภาคอุตสาหกรรมมานั้น ปัญหาที่เกิดจากการสื่อสารภายในองค์กรก่อสร้างที่พบเจอบ่อย ๆ ได้แก่ ไม่สามารถติดตามหรือรายงานผลได้อย่างมีแบบแผน บทสนทนาที่มีความสับสนเนื่องจากมีบุคคลภายในบทสนทนาเป็นจำนวนมาก จนทำให้ยากต่อการที่จะอ่านในทุก ๆ ประโยค อีกทั้งยังมีปัญหาของพนักงานก่อสร้างที่ต้องส่งรายงานเกี่ยวกับความคืบหน้า และจะต้องถ่ายภาพเพื่อรายงานประจำวัน ซึ่งปัญหาที่เกิดขึ้นคือ ถ่ายภาพบางจุดที่สำคัญไม่ครบ เช่น ลืมถ่ายภาพจุดที่พบปัญหาเพื่อที่จะแจ้งให้ผู้จัดการโปรเจกทราบ จึงอาจจะทำให้งานก่อสร้างเกิดเสร็จล่าช้า เนื่องจากถ้าหัวหน้าคนงานไม่สามารถรายงานความคืบหน้าให้กับผู้จัดการโปรเจกได้เรื่อย ๆ อาจจะทำให้คุณภาพในกระบวนการก่อสร้างไม่เป็นไปตามมาตรฐาน อาจเกิดปัญหาในกระบวนการก่อสร้างตามมา เช่น เกิดปัญหาที่การเทพื้นให้เรียบร้อยเพื่อที่จะปูกระเบื้อง แต่ถ้าเกิดปัญหาที่งานแรกอาจจะทำให้การปูกระเบื้องไม่สามารถทำได้ ทำให้กระบวนการก่อสร้างอาจจะไม่ได้คุณภาพ และทำให้ส่งผลเสียต่อภาพรวมขององค์กรอีกด้วย

ผู้จัดทำได้สังเกตเห็นปัญหาในการสื่อสารสำหรับภายในองค์กร จึงพัฒนาแอปพลิเคชันที่จะมาช่วยเหลือในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง เพื่อลดระยะเวลาในการสื่อสาร และลดการสื่อสารภายในองค์กรก่อสร้างเกิดข้อผิดพลาดในกระบวนการก่อสร้างให้น้อยลง

1.2 วัตถุประสงค์

1. เพื่อสร้างแอปพลิเคชันที่ช่วยลดระยะเวลาการสื่อสาร ให้เกิดความเข้าใจที่ตรงกันภายในโครงการก่อสร้าง ทำให้ลดระยะเวลาในกระบวนการก่อสร้าง
2. สามารถเพิ่มมัลติมีเดียและเอกสาร ในการรายงานความคืบหน้าของโครงการก่อสร้างแบบทันที่

1.3 ขอบเขตของโครงการ

1. บริษัทก่อสร้างที่ให้ข้อมูลความต้องการของผู้ใช้งาน
 - 1.1 บริษัท Team built CO.,LTD. โดยประธานกรรมการบริหาร
 - 1.2 สำนักวิจัยและพัฒนาทาง กรมทางหลวง โดยวิศวกรระดับสูง
 - 1.3 บริษัท ปริณสิริ จำกัด (มหาชน) โดยผู้จัดการโปรเจค
2. ข้อมูลความต้องการของผู้ใช้งาน

ตารางที่ 1.3.1 แสดงข้อมูลความต้องการของผู้ใช้งานจำแนกตามบริษัท

ความต้องการของผู้ใช้งาน	บริษัท Team built CO.,LTD.	สำนักวิจัยและพัฒนา ทาง กรมทางหลวง	บริษัท ปริณสุริ จำกัด (มหาชน)
1. แอปพลิเคชันที่สามารถสื่อสาร ได้คล้ายกับ แอปพลิเคชัน LINE	○		○
2. สามารถรายงานความคืบหน้าให้กับ ผู้บริหารระดับสูงได้ด้วยระบบแบบทันที	○	○	○
3. สามารถอัปโหลดมัลติมีเดียได้ เช่น รูปภาพ วีดีโอ แบบแปลนโครงสร้าง	○	○	○
4. มีสถานะรายงานความคืบหน้า ของแต่ละ งานย่อย	○	○	○
5. มีระบบการแจ้งเตือน	○		○
6. สามารถระบุบุคคลที่รับมอบหมายงานได้	○		○
7. สามารถส่งรายงานโดยใช้ จดหมาย อิเล็กทรอนิกส์	○	○	○

3. รายงานผลบนจอ iPhone 6 ขนาดหน้าจอสำหรับการแสดงผล 4.7 นิ้ว หรือเทียบเท่า iPhone8 ที่มี
ขนาดหน้าจอเท่ากัน และมีหน้าจอละเอียด 1334 x 750 พิกเซล ที่ 326 ppi

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาการใช้งานของ React Native, VScode และ Xcode
2. รวบรวมความต้องการของผู้ใช้งาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ต่อผู้พัฒนาโครงการ

1. ได้พัฒนาทักษะการเขียนภาษาจาวาสคริปต์และการใช้ React Native
2. ได้พัฒนาความรู้ในด้าน User Interface Design และ User Experience
3. ได้พัฒนาทักษะการคิด วิเคราะห์ และการทำงานเป็นทีม

ประโยชน์ต่อผู้ใช้งาน

1. ลดระยะเวลาการสื่อสารในกระบวนการก่อสร้าง
2. สามารถดูข้อมูลของโครงการผ่านทางระบบออนไลน์แบบประมวลแบบทันที
3. สามารถทราบถึงปัญหาในโครงการก่อสร้างจากการแจ้งเตือนทำให้ช่วยตรวจสอบคุณภาพของกระบวนการได้แบบทันที

1.6 โครงสร้างของรายงาน

บทที่ 2 จะกล่าวถึงงานวิจัยที่เกี่ยวข้องกับข้อผิดพลาดในการสื่อสารภายในองค์กรและพฤติกรรมการสื่อสารภายในองค์กร

บทที่ 3 จะกล่าวถึงการวิเคราะห์ความต้องการของผู้ใช้งาน ซึ่งจะประกอบไปด้วยการเก็บและรวบรวมความต้องการของผู้ใช้งาน จนถึงการนำมาวิเคราะห์ค้นหาฟังก์ชันภายในแอปพลิเคชัน เพื่อที่จะนำมาตอบสนองความต้องการของผู้ใช้งาน

บทที่ 4 จะกล่าวถึงผลการทดสอบการใช้งานของผู้ใช้งานที่ออกแบบในบทที่ 3

บทที่ 5 จะกล่าวถึงข้อสรุป และข้อเสนอแนะของโครงการ

บทที่ 2

งานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงงานวิจัยที่เกี่ยวข้องกับการสื่อสารภายในองค์กรธุรกิจอุตสาหกรรม ภาษาที่จะใช้ในการเขียนแอปพลิเคชัน โปรแกรมที่จะใช้เขียน และแสดงให้เห็นถึงแอปพลิเคชันที่มีระบบการทำงานที่น่าสนใจ

2.1 การวิเคราะห์การสื่อสารภายในองค์กรธุรกิจอุตสาหกรรม

จากงานวิจัย การศึกษาพฤติกรรมการสื่อสารภายในองค์กรธุรกิจอุตสาหกรรม [2] ที่มีการศึกษาพฤติกรรม และปัจจัยที่มีอิทธิพลต่อการสื่อสารภายในองค์กรธุรกิจ เนื่องจากการสื่อสารระหว่างบุคคลหรือกลุ่มบุคคล ก็มีการสื่อสารภายในที่มีรูปแบบเฉพาะ ซึ่งการสื่อสารนับว่าเป็นสายงานที่สำคัญ เพื่อให้เกิดประสิทธิผล และบรรลุวัตถุประสงค์และเป้าหมายที่กำหนด

ส่วนแรกในการศึกษาถึงพฤติกรรม และปัจจัยสำคัญที่มีอิทธิพลต่อการสื่อสารภายในองค์กร โดยจำแนกสถานภาพทั่วไปของบุคลากรด้านเพศ อายุ ระดับการศึกษา ระดับเงินเดือน ระดับตำแหน่งงาน และอายุงาน ซึ่งกลุ่มตัวอย่างที่ใช้ในการศึกษาคือ กลุ่มผู้บริหาร ผู้จัดการ หัวหน้างาน และพนักงาน โดยมีตัวแปรอิสระคือ สถานภาพทั่วไปของประชากร และตัวแปรตามได้แก่ พฤติกรรมการสื่อสารของบุคลากรในองค์กรธุรกิจอุตสาหกรรม

ส่วนที่สองเป็นการวิเคราะห์พฤติกรรมการสื่อสารของบุคลากรภายในองค์กร ผู้จัดทำจึงได้ออกแบบแบบทดสอบถามทั้งหมด 4 ตอน ซึ่งตอนที่ 1 เกี่ยวกับสถานภาพทั่วไปของผู้ตอบแบบสอบถาม เป็นแบบตรวจสอบรายการ (Check List) ตอนที่ 2 เกี่ยวกับพฤติกรรมการสื่อสารของบุคลากรในองค์กรธุรกิจอุตสาหกรรม เป็นแบบตรวจสอบรายการ (Check List) ตอนที่ 3 เกี่ยวกับปัจจัยที่มีอิทธิพลต่อการสื่อสารภายในองค์กรธุรกิจโดยเป็นแบบมาตราส่วนประมาณค่า (Rating Scale) และตอนที่ 4 เป็นแบบทดสอบเกี่ยวกับความคิดเห็นและข้อเสนอแนะอื่น ๆ ที่มีต่อการปรับปรุงและพัฒนาด้านการสื่อสารในองค์กร

จากงานวิจัยนี้จึงพบว่าพฤติกรรมการสื่อสารภายในองค์กรธุรกิจอุตสาหกรรม พบว่าส่วนใหญ่การสื่อสารที่ต้องถ่ายทอดความรู้ ความคิด ความเข้าใจ และความรู้สึก จะต้องสื่อสารความหมายให้เข้าใจมากที่สุด และเมื่อพบเจอปัญหาต่าง ๆ ควรรีบส่งสารไปยังคนที่ต้องการจะส่งสาร จะทำให้การสื่อสารมีประสิทธิภาพมากที่สุด

2.2 ผลสัมฤทธิ์

ดังนั้นผู้พัฒนาเมื่อได้ทราบถึงความต้องการของผู้ใช้งานว่าต้องการแอปพลิเคชันที่ต้องใช้ในการสื่อสารภายในองค์กร ผู้พัฒนาจึงศึกษาปัญหาที่มักเจอในการสื่อสารภายในองค์กรจากงานวิจัยอื่น ๆ เพื่อที่จะนำมาประกอบการวิเคราะห์เพื่อหาแนวทางในการแก้ปัญหาการสื่อสารภายในองค์กรให้แก่ผู้ใช้งานได้อย่างมีประสิทธิภาพยิ่งขึ้น

ในบทนี้ก็จะกล่าวถึงความรู้พื้นฐานที่เกี่ยวข้องกับการพัฒนาระบบในแอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง ซึ่งประกอบไปด้วยความรู้พื้นฐานเกี่ยวกับภาษาที่ใช้ในการพัฒนาและเครื่องมือที่เกี่ยวข้อง ได้แก่ ภาษาจาวาสคริปต์ และ การใช้งาน React - Native

2.3 ภาษาจาวาสคริปต์

ภาษาจาวาสคริปต์ เป็นภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมอินเทอร์เน็ต ที่กำลังได้รับความนิยมอย่างสูง ซึ่งเป็นภาษาเชิงวัตถุ สามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานคือ แปลความและดำเนินงานไปที่ละคำสั่ง หรือ Object Oriented Programming สามารถทำงานข้ามแพลตฟอร์มได้โดยทำงานร่วมกับภาษา HTML และ ภาษาจาวาสคริปต์ และ จาวาสคริปต์ มีการพัฒนาเวอร์ชันใหม่ ๆ ออกมา โดยเวอร์ชันล่าสุดของจาวาสคริปต์ คือ เวอร์ชัน 9 [10].

2.3.1 การใช้งานจาวาสคริปต์

ตัวแปร

จาวาสคริปต์มีระบบตัวแปรแบบ Dynamic Typing หรือชนิดตัวแปรจะเป็นอะไรก็ได้ เปลี่ยนได้เรื่อย ๆ ตามค่าที่ตัวมันเก็บเหมือนกับภาษา Python หรือ Ruby คือการไม่จำเป็นต้องประกาศ Type ของตัวแปร เช่น

```
var x;
var n = 123;
var f = 100.258;
var b = true;
var name = 'chula'
```

รูปที่ 2.3.1.1 แสดงโค้ดสำหรับการประกาศตัวแปรภายในภาษาจาวาสคริปต์

- Var เป็นตัวแปร แบบ Global scope ซึ่งเป็นตัวแปรที่สามารถเรียกใช้ได้ทุกที่
- Let เป็นตัวแปร แบบ Block scope ซึ่งเป็นตัวแปรที่สามารถเรียกใช้ได้เฉพาะฟังก์ชัน
- Const เป็นตัวแปร แบบ Block scope ซึ่งเป็นตัวแปรที่ไม่สามารถเปลี่ยนค่าได้ ยกเว้นเปลี่ยนค่าภายใน object

Array ชุดของข้อมูล

เป็นตัวแปรพิเศษซึ่งสามารถเก็บมากกว่าหนึ่งค่าในแต่ละครั้ง เช่น

```
var cars = ["Apple", "Orange", "Lemon"];
```

รูปที่ 2.3.1.2 แสดงโค้ดสำหรับการใช้ Array ภายในภาษาจาวาสคริปต์

ฟังก์ชันที่ใช้ใน Array

- Shift() คือฟังก์ชันเอาค่าด้านซ้ายสุดใน Array ออก เช่น

```
var list = ["Apple", "Orange", "Lemon"];
list.shift();
list;
//result ["Orange", "Lemon"];
```

รูปที่ 2.3.1.3 แสดงโค้ดสำหรับการใช้ Shift ภายในภาษาจาวาสคริปต์

- pop() คือ ฟังก์ชันที่จะเอาค่าด้านขวาสุดใน Array ออก เช่น

```
var list = ["Apple", "Orange", "Lemon"];
list.pop();
list;
//result ["Apple", "Orange",];
```

รูปที่ 2.3.1.4 แสดงโค้ดสำหรับการใช้ pop ภายในภาษาจาวาสคริปต์

- Push() คือ ฟังก์ชันที่เอาค่าใหม่ที่ใส่ไป ไว้ค่าท้ายสุดใน Array

```
var list = ["Apple","Orange","Lemon"];
list.push("Banana");
list;
//result ["Apple","Orange","Lemon","Banana"];
```

รูปที่ 2.3.1.5 แสดงโค้ดสำหรับการใช้ push ภายในภาษาจาวาสคริปต์

- Concat() คือ ฟังก์ชันที่ใช้เชื่อม 2 Array เข้าด้วยกัน เช่น

```
var list = ["Apple","Orange","Lemon"];
var newlist = ["A","B"];

var allList = list.concat(newlist);
allList;
//result ["Apple","Orange","Lemon","A","B"];
```

รูปที่ 2.3.1.6 แสดงโค้ดสำหรับการใช้ concat ภายในภาษาจาวาสคริปต์

Function

ฟังก์ชัน (Function) ชุดคำสั่งที่รวมการทำงานเอาไว้ สามารถเรียกชื่อมาใช้งานตามที่เรากำลังการใช้ได้ เช่น

```
function ชื่อฟังก์ชัน (ตัวแปรพารามิเตอร์ถ้ามี){
  statement ที่ต้องการให้ทำงาน เมื่อมีการเรียก function
  return <- ถ้าต้องการส่งค่าบางอย่างกลับไปนอกฟังก์ชัน เมื่อมีการทำงานมาถึงจุดนี้
}
```

รูปที่ 2.3.1.7 แสดงโค้ดสำหรับการใช้ฟังก์ชันภายในภาษาจาวาสคริปต์

Console.log

สำหรับภาษาจาวาสคริปต์แล้วการพิมพ์ console.log() คือคำสั่งสำหรับพิมพ์ค่าตัวแปร เช่น

```
console.log("Chula");
// result Chula
```

รูปที่ 2.3.1.8 แสดงโค้ดสำหรับการใช้ console ภายในภาษาจาวาสคริปต์

ตัวแปร this

เมื่อสร้างฟังก์ชันขึ้นมาแล้วทางจาวาสคริปต์ จะเติมพารามิเตอร์ในตำแหน่งแรกให้เราโดยอัตโนมัติ คือตัวแปร this แต่ไม่จำเป็นต้องพิมพ์ลงไปในฟังก์ชันเพราะทางจาวาสคริปต์ ได้สร้างไว้ให้อยู่แล้ว และในภาษาจาวาสคริปต์คำว่า this อ้างถึง Context หรือ บริบทในขณะนั้น

```
function add (this,x,y){
  return x + y ;
}
```

รูปที่ 2.3.1.9 แสดงโค้ดสำหรับการใช้ this ภายในภาษาจาวาสคริปต์

Bind

เมื่อมีการส่ง method ผ่านพารามิเตอร์ แต่ context ไม่ตามไปด้วย ทำให้ this ผิดความหมาย จึงต้องใช้คำสั่ง .bind ในการแนบ context ไปด้วย เช่น

```
caller(obj.printX.bind(obj));
```

รูปที่ 2.3.1.10 แสดงโค้ดสำหรับการใช้ bind ภายในภาษาจาวาสคริปต์

หรือ เมื่อต้องใช้กับ ฟังก์ชัน ที่เป็น Asynchronous ซึ่งทำให้เราสามารถเปลี่ยน context ของ callback ตัวนี้ได้ เช่น

```

var request = {
  call: function(){
    ajax( function(res)){
      this.response(res);
    }.bind(this) );
  },
  response: function(res){
    //..do something..
  }
}

```

รูปที่ 2.3.1.11 แสดงโค้ดสำหรับการใช้ bind ใน callback ภายในภาษาจาวาสคริปต์

เมื่อการใช้ bind เป็นเพียงการแนบ context ให้ติดไปกับฟังก์ชัน แต่ยังไม่สั่งให้ทำงาน ซึ่งอาจจะใช้คำสั่ง .call แทน แต่เมื่อมีการส่งพารามิเตอร์ไปอยู่ในรูปขอ array ควรใช้ .apply แทนการใช้ .call

```

var params = [12,20,30,40];

function dummy(x,y,z,w){
  // x is 10
  // y is 20
  // z is 30
  // w is 40
}

dummy.call(null,params[0],params[1],params[2],params[3]);

//หรือถ้าเปลี่ยนไปใช้ apply
dummy.apply(null,params);

```

รูปที่ 2.3.1.12 แสดงโค้ดสำหรับการใช้ apply ภายในภาษาจาวาสคริปต์

For of

เป็นการวนลูปเหมือนกับ Foreach เช่น

```
// แบบ for of
for (ech of data){
  | console.log(ech);
}
//result 10 20 30 40
```

รูปที่ 2.3.1.13 แสดงโค้ดสำหรับการใช้ For of ภายในภาษาจาวาสคริปต์

Module

โมดูลคือการยกโค้ดออกเป็นไฟล์ ๆ แล้วโหลดโค้ดเข้ามาประกอบ คล้าย ๆ กับ import ใน JAVA ซึ่งถ้าเราต้องการโหลดเพียงอย่างเดียวเข้ามาใช้ สามารถประกาศได้ด้วยการกำหนด default เช่น

```
//User.js
export default class User {
}
// main app
//สังเกตว่าครั้งนี้ไม่ต้องใส่ {} แล้ว
import User from './User.js'
```

รูปที่ 2.3.1.14 แสดงโค้ดสำหรับการใช้ Module ภายในภาษาจาวาสคริปต์

Promise

เป็นการเขียนโปรแกรมแบบ Asynchronous โดยจะรับ parameter เป็นฟังก์ชันที่ปกติจะใช้รูปแบบ (resolve,reject) => {...} ซึ่งจะสั่งรัน resolve เมื่อโค้ดทำงานเสร็จ หรือ reject เมื่อเกิด Error สำหรับ resolve จะใช้ .then แต่ถ้าเกิด reject จะใช้ .catchn เช่น

```

let p = new Promise( (resolve,reject)=>{
  if (success){
    resolve(1);
  }else{
    reject(2);
  }
});
p.then((val)=>{
  console.log();
}).catch((val)=>{
  console.log();
});

```

รูปที่ 2.3.1.15 แสดงโค้ดสำหรับการใช้ Promise ภายในภาษาจาวาสคริปต์

2.3.2 React Native

React Native เป็นการนำ ReactJS ซึ่งเป็น จาวาสคริปต์ และยังเป็นเฟรมเวิร์คของทาง Facebook ที่ใช้กับเว็บไซต์มาต่อยอดกับโทรศัพท์ จึงสามารถใช้ส่วนของข้อดีต่าง ๆ ของ ReactJS ได้ เช่น การใช้โค้ดร่วมกันในรูปแบบ Component โดยการทำงานของจาวาสคริปต์ที่เราเขียนภายใต้ React Native จะสื่อสารกับฝั่งของ Native ด้วย JSON แบบเฉพาะมีลักษณะเป็น Serialization โดยเราเรียกส่วนที่สื่อสารกันว่า Bridge [11].

2.3.3 Visual Studio Code

คือ Editor ของทาง Microsoft ด้วยความที่ทั้งเร็ว และมีลูกเล่นให้ค่อนข้างเยอะ เช่น Visual Studio Code มีการฝัง Git เข้ามาในตัว editor ทำให้เมื่อเราเอา cursor ไปวางไว้ที่โค้ด ก็จะได้เห็นว่าผู้ใช้งานใดเป็นคน commit โค้ดส่วนนั้นขึ้นไป และมี Auto Rename Tag ซึ่งจะเป็น Component Tag เปิด และ ปิด ให้ตรงกันอยู่เสมอ [12].

2.4 การเชื่อมต่อกับหลังบ้าน

เนื่องจากการทำหน้าบ้าน หรือ Frontend จึงต้องมีส่วนต่อประสานกับ หลังบ้านหรือ Backend ซึ่งหลังบ้านในแอปพลิเคชันนี้ได้ใช้ AWS Amplify สำหรับการเก็บข้อมูลต่าง ๆ เนื่องจากแพลตฟอร์มนี้ เป็นแพลตฟอร์มในการเก็บข้อมูลที่ปลอดภัย และยังสามารถผสมผสานการใช้งานได้กับ Machine Learning ได้ และ เมื่อจะมีการเรียก API ที่ต้องการนำข้อมูลต่าง ๆ จากหลังบ้านมาแสดงยังหน้าบ้านจะมีการพื้นฐานในการใช้โค้ดดังนี้

GET

เป็น API ที่จะแสดงค่าที่หน้าบ้านโดยการรับนำค่าจากหลังบ้านมาแสดง โดยมีวิธีใช้ดังนี้

```
API.get(apiName: any, path: any, init: any)
```

POST

เป็น API ที่จะเพิ่มค่าใหม่จากหน้าบ้านโดยการรับส่งไปยังหลังบ้าน โดยมีวิธีใช้ดังนี้

```
API.post(apiName: any, path: any, init: any)
```

PUT

เป็น API ที่จะปรับปรุงค่าเป็นค่าใหม่ที่ Input เข้าไปจากหน้าบ้าน ส่งไปยังหลังบ้าน โดยมีวิธีใช้ดังนี้

```
API.put(apiName: any, path: any, init: any)
```

DELETE

เป็น API ที่จะลบค่าดังที่เราได้ส่ง key จากหน้าบ้านส่งไปให้หลังบ้านได้ลบค่า โดยมีวิธีใช้ดังนี้

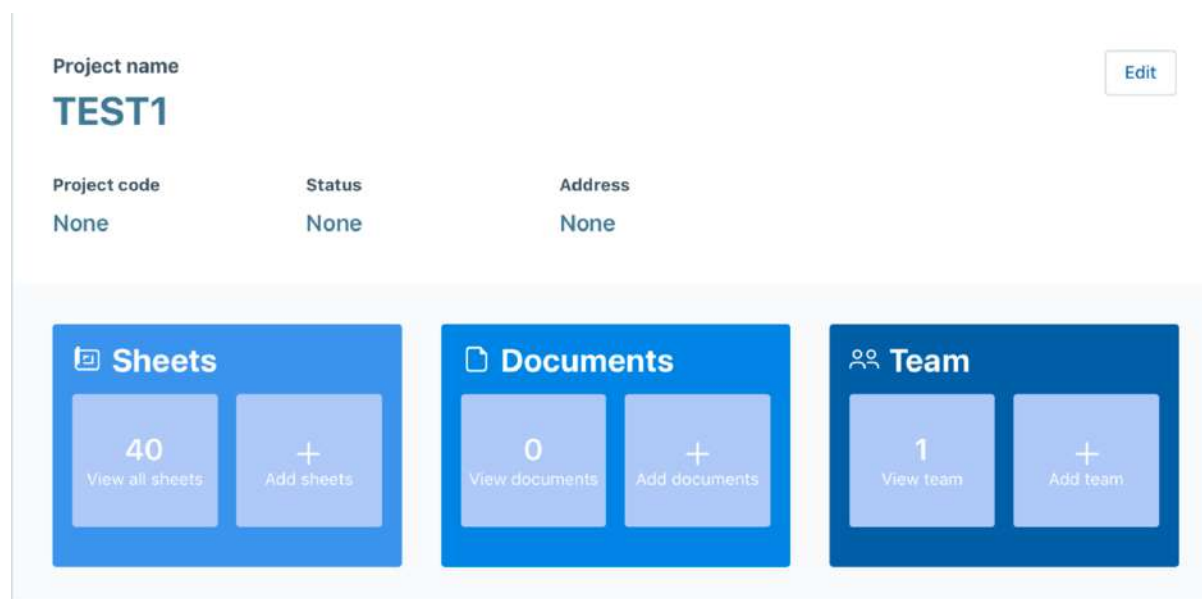
```
API.del(apiName: any, path: any, init: any)
```

2.5 แอปพลิเคชันที่มีระบบการทำงานที่น่าสนใจ

จากการศึกษาแอปพลิเคชันภายในโลกออนไลน์ในขณะนี้พบว่า มีแอปพลิเคชันที่มีระบบที่น่าสนใจอยู่ 2 แอปพลิเคชัน คือ

2.5.1 Plangrid

เป็นแอปพลิเคชันที่อยู่ในอุตสาหกรรมก่อสร้างก็มีการใช้กัน เป็นแอปพลิเคชันที่สามารถเพิ่มโปรเจกต์ลงในแอปพลิเคชัน [13]. และใช้สื่อสารกันได้ เช่น



รูปที่ 2.5.1.1 แสดงหน้าจอแอปพลิเคชัน Plangrid

ภายในแอปพลิเคชันสามารถเพิ่มงานได้โดยกรอกรายละเอียดงานต่างๆ ลงภายในกล่อง

รูปที่ 2.5.1.2 แสดงหน้าจอแอปพลิเคชัน Plangrid

อีกทั้งยังสามารถแสดง Progress ของแต่ละงานได้อีกด้วย



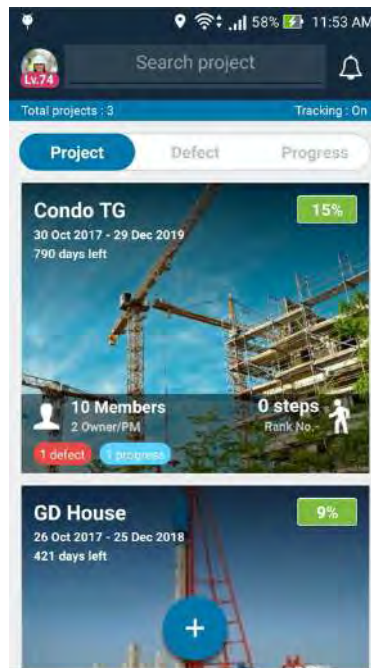
รูปที่ 2.5.1.3 แสดงหน้าจอแอปพลิเคชัน Plangrid

และมีฟังก์ชันต่าง ๆ ที่สามารถทำให้ผู้ใช้งานได้สามารถเลือกใช้งานได้อย่างหลากหลาย ซึ่งภายในแอปพลิเคชันนี้สามารถรองรับระบบ Live chat ได้อีกด้วย และเนื่องจากเป็นเวอร์ชันทดลองจึงสามารถใช้ฟรีได้ถึง 21 วัน หลังจากนั้นจะมีราคาที่แตกต่างกันออกไปให้ผู้ใช้งานได้เลือก

แต่แอปพลิเคชันนี้มีข้อจำกัดคือ ไม่สามารถมองเห็นงานย่อย ๆ แยกประเภทโปรเจคได้ ทำให้ไม่สามารถเพิ่มงานย่อยเพิ่มภายในโปรเจคที่มีงานย่อยอยู่แล้วได้

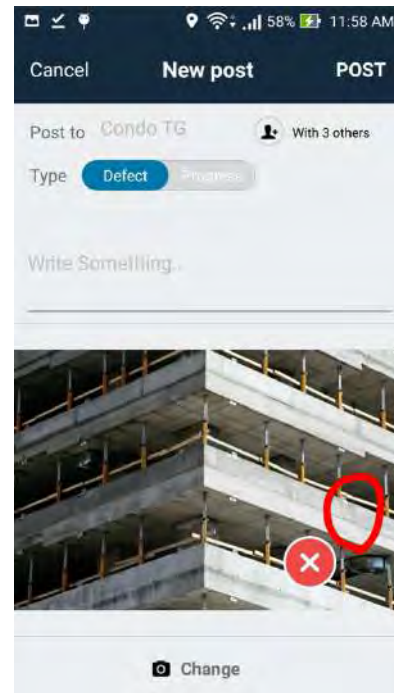
2.5.2 Sitewalk

เป็นแอปพลิเคชันที่อยู่ในอุตสาหกรรมก่อสร้างก็มีการใช้กันเช่นกันและเป็นแอปพลิเคชันที่สามารถเพิ่มโปรเจกต์ลงในแอปพลิเคชันและใช้สื่อสารข้อมูลกันภายในโปรเจกต์ได้ [14]. เช่น



รูปที่ 2.5.2.1 แสดงหน้าจอแอปพลิเคชัน Sitewalk

อีกทั้งยังสามารถแท็กชื่อผู้ใช้งานอื่น ๆ เพื่อให้เห็นโพสต์ที่ตนเองจะโพสต์ และมีติตแฮชแท็กเพื่อบ่งบอกประเภท เช่น Defect , Done เป็นต้น อีกทั้งยังสามารถเพิ่มรูปถ่าย ภายในโปรเจกต์ได้



รูปที่ 2.5.2.2 แสดงหน้าจอแอปพลิเคชัน Sitewalk รูปที่ 2.5.2.3 แสดงหน้าจอแอปพลิเคชัน Sitewalk

แต่แอปพลิเคชันนี้มีข้อจำกัดคือ ไม่สามารถเพิ่มงานย่อยภายในโปรเจกต์เดียวกันได้ เช่นเดียวกับแอปพลิเคชัน Plangrid คือจะเพิ่มได้แค่ 1 งานย่อยต่อ 1 โปรเจกต์ จึงทำให้ภายในแอปพลิเคชันที่อาจจะมีหลายโปรเจกต์ ทำให้ผู้ใช้งานอาจเกิดความสับสนได้

บทที่ 3

การวิเคราะห์และการออกแบบ

ในบทนี้จะกล่าวถึงวิธีการรวบรวมความต้องการของผู้ใช้งานนำมาวิเคราะห์เพื่อที่จะออกแบบฟังก์ชันการใช้งานของแอปพลิเคชัน

3.1 บทนำ

จากการศึกษาและรวบรวมความต้องการของผู้ใช้งาน จึงมีการจัดและรวบรวมความต้องการของผู้ใช้งานผ่านทางการสัมภาษณ์เป็นจำนวน 5 ครั้ง รวบรวมออกมาเป็น ข้อกำหนดความต้องการด้านซอฟต์แวร์เบื้องต้น ดังนี้

3.1.1 ขั้นตอนการรวบรวมความต้องการ

ซึ่งก่อนการรวบรวมความต้องการทางผู้จัดทำได้ทำการเตรียมตัวในการรวบรวมความต้องการคือการเขียนหัวข้อเรื่องในการที่จะสอบถามอย่างคร่าว ๆ เช่น ขั้นตอนในการทำงานเพื่อที่จะได้มา 1 งานเป็นอย่างไรบ้าง เป็นต้น เพื่อเป็นแนวทางในการสัมภาษณ์ผู้ใช้งาน

ในขั้นตอนการรวบรวมความต้องการเพื่อที่จะนำมาพัฒนาแอปพลิเคชัน ทางผู้พัฒนาได้มีการรวบรวมความต้องการโดยใช้วิธีการไปสัมภาษณ์ผู้ใช้งาน จาก 3 หน่วยงาน ตามระยะเวลาดังนี้

- 30 พฤษภาคม พ.ศ. 2562 สัมภาษณ์หน่วยงาน สำนักวิจัยและพัฒนาทาง กรมทางหลวง
- 17 ตุลาคม พ.ศ. 2562 สัมภาษณ์หน่วยงาน บริษัท Team built CO.,LTD.
- 17 พฤศจิกายน พ.ศ. 2562 สัมภาษณ์หน่วยงาน บริษัท ปริณูสิริ จำกัด (มหาชน)
- 19 ธันวาคม พ.ศ. 2562 สัมภาษณ์หน่วยงาน บริษัท Team built CO.,LTD.
- 22 ธันวาคม พ.ศ. 2562 สัมภาษณ์หน่วยงาน บริษัท ปริณูสิริ จำกัด (มหาชน)

ซึ่งหลังจากการรวบรวมความต้องการ ทางผู้พัฒนาได้การรวบรวมและวิเคราะห์ความต้องการออกมา ซึ่งจะได้ความต้องการทางด้านซอฟต์แวร์ออกมา ตามหัวข้อ 3.2 ข้อกำหนดความต้องการด้านซอฟต์แวร์



รูปที่ 3.1.1.1 รวบรวมความต้องการที่สำนักวิจัยและพัฒนาทาง กรมทางหลวง



รูปที่ 3.1.1.2 รวบรวมความต้องการที่ บริษัท Team built CO.,LTD.



รูปที่ 3.1.1.3 รวบรวมความต้องการที่ บริษัท ปริณสุริ จำกัด (มหาชน)

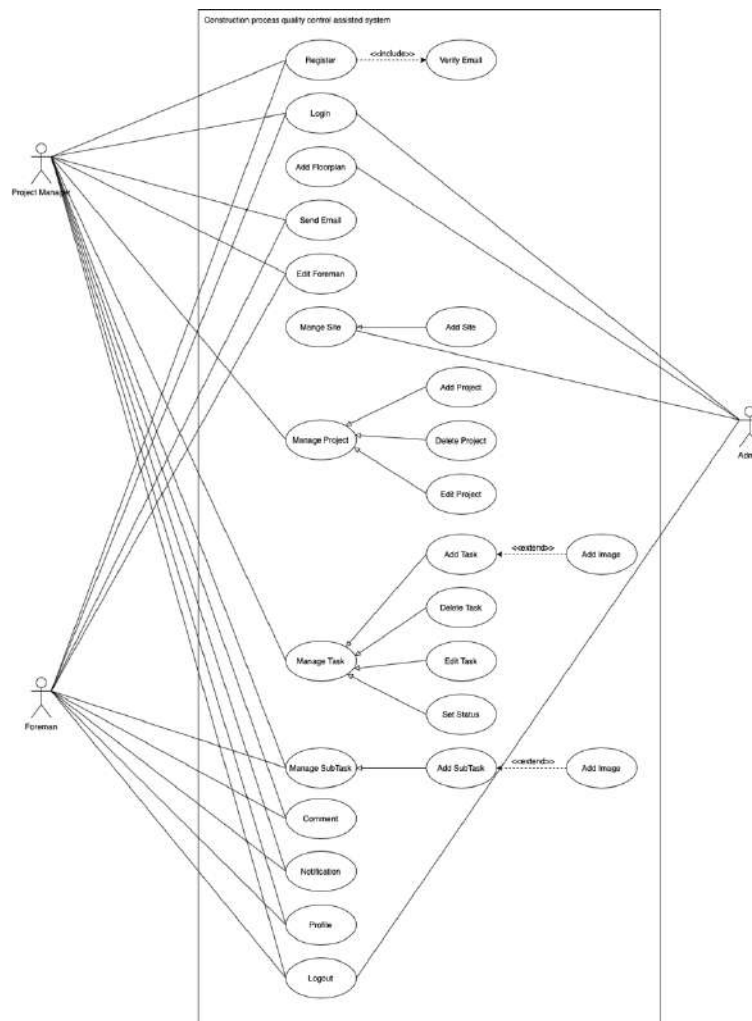
3.2 ข้อกำหนดความต้องการด้านซอฟต์แวร์

ระบบการสื่อสารภายในองค์กรการก่อสร้างนี้ เป็นเครื่องมือช่วยตรวจสอบคุณภาพในกระบวนการก่อสร้าง โดยจะเป็นระบบที่สามารถเพิ่มหัวข้อ รายละเอียด มัลติมีเดีย ในแต่ละงาน และสามารถเพิ่มความคิดเห็นในแต่ละปัญหา ซึ่งจะเป็นการสื่อสารที่เฉพาะภายในองค์กรก่อสร้าง พัฒนาอยู่ในรูปแบบของแอปพลิเคชัน โดยแอปพลิเคชันนี้สามารถช่วยให้บุคลากรภายในองค์กรก่อสร้างสื่อสารกันได้อย่างมีประสิทธิภาพ ซึ่งเป็นประโยชน์ต่อการพัฒนาประเทศด้านอุตสาหกรรม

โดยโครงสร้างในการทำงานภายในระบบจะสามารถแบ่งออกมาได้เป็น 4 ระดับ ดังนี้
 1.โครงการ 2.โปรเจค 3.งาน 4.งานย่อยหรือปัญหา ซึ่งแต่ละระดับจะมีลักษณะที่แตกต่างดังนี้ โครงการจะเป็นรายละเอียดถึงสถานที่ ซึ่งภายในโครงการมักจะมีหลายโปรเจค ภายในโปรเจคอาจจะมีงานหลายงาน ต่อมาคือ ในงานหลายงานอาจมี หลายงานย่อย โดยจะยกตัวอย่างเช่น ในโครงการ Foret Salaya จะมี 3 โปรเจค คือมี บ้านเดี่ยว คอนโดมิเนียม และทาวน์เฮ้าส์ ซึ่งในแต่ละโปรเจคอาจจะมีหลายงาน เช่น งานปูพื้นชั้น 1 งานผนัง งานระบบท่อน้ำ และในงานจะมีหลายงานย่อย เช่น ในงานปูพื้นชั้น 1 จะมีงานย่อย ดังนี้ ติเส้น เทปูน ฉาบปูน ปูกระเบื้อง เป็นต้น

3.3 ความต้องการเชิงหน้าที่

จากการรวบรวมความต้องการด้วยการสัมภาษณ์บุคคล การวิเคราะห์เอกสาร และการมีส่วนร่วมในการสร้างความต้องการเพื่อใช้ภายในแอปพลิเคชัน ซึ่งสามารถเขียนเป็นความต้องการเชิงหน้าที่ (Functional Requirements : FR) โดยแสดงเป็นแผนภาพยูเคส ดังรูปที่ 3.3.1



รูปที่ 3.3.1 แผนภาพยูสแคสแสดงหน้าที่การทำงานของระบบช่วยในการตรวจสอบคุณภาพสำหรับ
กระบวนการก่อสร้าง

จากแผนภาพยูสแคสแสดงหน้าที่การทำงานของระบบการสื่อสารที่ใช้ในการตรวจสอบคุณภาพ
กระบวนการก่อสร้าง แบ่งผู้ใช้งานออกเป็น 3 กลุ่มดังนี้

- 1) ผู้ดูแลระบบ หรือ Admin เป็นผู้ที่กำหนดสิทธิ์และหน้าที่ให้กับผู้ใช้งานอื่น ๆ
- 2) ผู้จัดการโปรเจค หรือ Project manager เป็นผู้จัดการโครงการก่อสร้าง สังกัดบริษัทตามขอบเขตโครงการ
คือ 2 บริษัท ได้แก่ บริษัท ปริณสุริ จำกัด (มหาชน) และ Team built CO.,LTD และอีก 1 สำนักงาน
ได้แก่ สำนักวิจัยและพัฒนาทาง กรมทางหลวง
- 3) หัวหน้าคนงาน หรือ Foreman ที่อยู่ภายในสังกัดตามบริษัท ปริณสุริ จำกัด (มหาชน) และ Team built
CO.,LTD และอนุวิศวกร ที่อยู่ภายในสังกัด สำนักวิจัยและพัฒนาทาง กรมทางหลวง

แผนภาพยูสเคสแสดงหน้าที่การทำงานของระบบตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้างมีระบบสำหรับการทำงานโดยมีรายละเอียดดังนี้

ตารางที่ 3.3.1 ตารางแสดงการ Register

Use Case Name	Register
Participating Actor	ผู้จัดการโปรเจค และ หัวหน้าคนงาน
Entry Condition	ลงทะเบียนเข้าสู่ระบบ
Flow of events	<ol style="list-style-type: none"> 1. ระบบแสดงหน้าต่างลงทะเบียนเข้าสู่ระบบ 2. ผู้ใช้กรอก รหัสผู้ใช้งาน , รหัสผ่าน ,อีเมล ,เบอร์โทร และตำแหน่ง 3. กรอกโค้ดที่ได้รับจากอีเมลเพื่อยืนยันอีเมล 4. ระบบจะแจ้งผลการลงทะเบียนว่าสำเร็จ
Exit condition	ลงทะเบียนสำเร็จ
Special requirement	-

ตารางที่ 3.3.2 ตารางแสดงการ Login

Use Case Name	Login
Participating Actor	ผู้จัดการโปรเจค , หัวหน้าคนงาน และ ผู้ดูแลระบบ

Entry Condition	กดปุ่มเข้าสู่ระบบ
Flow of events	<ol style="list-style-type: none"> 1. ระบบแสดงหน้าต่างสำหรับเข้าสู่ระบบ 2. ผู้ใช้กรอก รหัสผู้ใช้งาน , รหัสผ่าน 3. ระบบตรวจสอบความถูกต้องกับ ฐานข้อมูล <ol style="list-style-type: none"> 3.1) กรณีที่กรอก รหัสผู้ใช้งาน , รหัสผ่าน ถูกต้อง ระบบจะแสดง หน้าจอที่แสดงถึงรายละเอียดโครงการของผู้ใช้งาน 3.1) กรณีที่กรอก รหัสผู้ใช้งาน , รหัสผ่าน และ หน้า ที่ ผิดพลาด ระบบจะแจ้งผลการเข้าสู่ระบบไม่สำเร็จ
Exit condition	เข้าสู่ระบบสำเร็จ หรือ เข้าสู่ระบบไม่สำเร็จ
Special requirement	กรณีเข้าสู่ระบบไม่สำเร็จ ระบบจะแสดงข้อความแจ้งเตือนการผิดพลาดว่าเข้าสู่ระบบไม่สำเร็จ

ตารางที่ 3.3.3 ตารางแสดงการ Add Floorplan

Use Case Name	Add Floorplan
Participating Actor	ผู้ดูแลระบบ
Entry Condition	เพิ่มแบบแปลนของโครงการ
Flow of events	ผู้ดูแลระบบเพิ่มแบบแปลนให้ตามโครงการ

Exit condition	แบบแปลนเพิ่มเข้าสู่โครงการ
Special requirement	-

ตารางที่ 3.3.4 ตารางแสดงการ Send Email

Use Case Name	Send Email
Participating Actor	ผู้จัดการโปรเจค และ หัวหน้าคนงาน
Entry Condition	กดปุ่มฟังก์ชัน Email
Flow of events	<ol style="list-style-type: none"> 1. กรอกหัวข้อที่ต้องการส่ง 2. กรอกรายละเอียดที่ต้องการส่ง 3. กรอก Email ผู้ส่งและผู้รับ 4. กดปุ่มยืนยัน
Exit condition	ส่ง Email ไปยังผู้รับ
Special requirement	กรณีในการส่ง Email ผู้ใช้งานสามารถเลือกรายละเอียดโปรเจคที่ต้องการส่งไปพร้อมกับ Email ได้

ตารางที่ 3.3.5 ตารางแสดงการ Edit Foreman

Use Case Name	Edit Foreman
Participating Actor	ผู้จัดการโปรเจก และหัวหน้าคนงาน
Entry Condition	กดปุ่มฟังก์ชัน Edit Foreman
Flow of events	<ol style="list-style-type: none"> 1. ระบบแสดงรายชื่อหัวหน้าคนงาน 2. ระบบทำการแก้ไขรายชื่อหัวหน้าคนงาน <ol style="list-style-type: none"> 2.1) กรณีต้องการเพิ่มรายชื่อใหม่ เมื่อเลือกรายชื่อที่ต้องการเพิ่ม แล้ว ระบบจะทำการเพิ่มรายชื่อใหม่ลงไป 2.2) กรณีต้องการลบรายชื่อ เมื่อเลือกรายชื่อที่ต้องการลบออกจากรายชื่อ ระบบจะทำการลบรายชื่อนั้นออกไป
Exit condition	แก้ไขรายชื่อหัวหน้าคนงาน
Special requirement	-

ตารางที่ 3.3.6 ตารางแสดง Manage Site

Use Case Name	Manage Site
Participating Actor	ผู้ดูแลระบบ

Entry Condition	เพิ่มข้อมูลรายละเอียดโครงการในระบบ
Flow of events	<ol style="list-style-type: none"> 1. กรอกข้อมูลรายละเอียดโครงการในระบบ 2. เพิ่มการเข้าถึงให้ผู้ใช้งานที่อยู่ภายใต้ความรับผิดชอบในโครงการนี้
Exit condition	เพิ่มโครงการให้กับผู้ใช้งานที่อยู่ภายใต้ความรับผิดชอบโครงการ
Special requirement	-

ตารางที่ 3.3.7 ตารางแสดงการ Manage Project

Use Case Name	Manage Project
Participating Actor	ผู้จัดการโปรเจค
Entry Condition	คลิกโปรเจคที่ต้องการจัดการ
Flow of events	<ol style="list-style-type: none"> 1. ระบบแสดงโปรเจคทั้งหมดภายในโครงการ 2. ระบบทำการจัดการโปรเจค <ol style="list-style-type: none"> 2.1) กรณีต้องการเพิ่มโปรเจค ผู้ใช้งานกรอกข้อมูลโปรเจคใหม่ ระบบจะทำการเพิ่มโปรเจคใหม่ในโครงการ 2.2) กรณีต้องการลบโปรเจค ผู้ใช้งานเลือกโปรเจคที่ต้องการลบ ระบบจะทำการลบโปรเจคออกจากโครงการ 2.3) กรณีต้องการแก้ไขข้อมูลภายในโปรเจค ผู้ใช้งานเลือกโปรเจคที่ต้องการแก้ไขข้อมูล แล้วทำการแก้ไขข้อมูล ระบบจะทำการแก้ไขให้

Exit condition	จัดการโปรเจคที่เลือก
Special requirement	กรณีเพิ่มข้อมูล ลบ และแก้ไขโปรเจค หัวหน้าคนงานจะไม่สามารถทำรายการนี้ได้

ตารางที่ 3.3.8 ตารางแสดงการ ManageTask

Use Case Name	Manage Task
Participating Actor	ผู้จัดการโปรเจค
Entry Condition	คลิกลูกงานที่ต้องการจัดการ
Flow of events	<ol style="list-style-type: none"> 1. ระบบแสดงงานทั้งหมดภายในโปรเจค 2. ระบบทำการจัดการงาน <ol style="list-style-type: none"> 2.1) กรณีต้องการเพิ่มงาน ผู้ใช้งานกรอกข้อมูลงานใหม่ ระบบจะทำการเพิ่มงานใหม่ในโปรเจค 2.2) กรณีต้องการลบงาน ผู้ใช้งานเลือกงานที่ต้องการลบ ระบบจะทำการลบงานออกจากโปรเจค 2.3) กรณีต้องการแก้ไขข้อมูลภายในงาน ผู้ใช้งานเลือกงานที่ต้องการแก้ไขข้อมูล แล้วทำการแก้ไขข้อมูล ระบบจะทำการแก้ไขให้
Exit condition	จัดการงานที่เลือก
Special requirement	<ol style="list-style-type: none"> 1. .กรณีเพิ่มข้อมูลงานสามารถเพิ่มรูปถ่ายได้ 2. กรณีเพิ่มข้อมูล หัวหน้าคนงานจะไม่สามารถทำรายการนี้ได้

ตารางที่ 3.3.9 ตารางแสดงการ Manage Subtask

Use Case Name	Manage Subtask
Participating Actor	ผู้จัดการโปรเจค และ หัวหน้าคนงาน
Entry Condition	คลิกลานย่อยที่ต้องการจัดการ
Flow of events	<ol style="list-style-type: none"> 1. ระบบแสดงงานย่อยทั้งหมดภายในงาน 2. ระบบทำการจัดการงานย่อยโดยการที่ ผู้ใช้งานกรอกข้อมูลงานใหม่ ระบบจะทำการเพิ่มงานย่อยใหม่ในงาน
Exit condition	เพิ่มข้อมูลงานย่อยสำเร็จ
Special requirement	กรณีเพิ่มข้อมูลงานย่อยสามารถเพิ่มรูปถ่ายได้

ตารางที่ 3.3.10 ตารางแสดงการ Comment

Use Case Name	Comment
Participating Actor	ผู้จัดการโปรเจค และ หัวหน้าคนงาน
Entry Condition	คลิกปุ่มที่หน้าจอแสดงการเพิ่มความคิดเห็น
Flow of events	<ol style="list-style-type: none"> 1. เลือกงานย่อยที่ผู้ใช้ต้องการแสดงความคิดเห็น 2. กรอกแสดงความคิดเห็น 3. กดปุ่มยืนยัน
Exit condition	แสดงความคิดเห็นในงานย่อยที่เลือก

Special requirement	-
---------------------	---

ตารางที่ 3.3.11 ตารางแสดง Notification

Use Case Name	Notification
Participating Actor	ผู้จัดการโปรเจค และ หัวหน้าคนงาน
Entry Condition	คลิกปุ่มที่หน้าจอแสดงการแจ้งเตือน
Flow of events	<ol style="list-style-type: none"> 1. ระบบแสดงหน้าจอการแจ้งเตือน 2. คลิกการแจ้งเตือนในเรื่องที่ต้องการจะเปิด
Exit condition	ไปยังหน้าที่เชื่อมต่อการแจ้งเตือนในเรื่องที่ผู้ใช้งานต้องการเปิด
Special requirement	-

ตารางที่ 3.3.12 ตารางแสดง Profile

Use Case Name	Profile
Participating Actor	ผู้จัดการโปรเจค และ หัวหน้าคนงาน
Entry Condition	คลิกปุ่มที่หน้าจอแสดงหน้าโปรไฟล์

Flow of events	ระบบแสดงข้อมูลโปรไฟล์ของผู้ใช้งาน
Exit condition	แสดงข้อมูลโปรไฟล์ของผู้ใช้งาน
Special requirement	-

ตารางที่ 3.3.13 ตารางแสดงการ Logout

Use Case Name	Logout
Participating Actor	ผู้จัดการโปรเจค , หัวหน้าคนงาน และ ผู้ดูแลระบบ
Entry Condition	คลิกปุ่มที่หน้าจอโปรไฟล์
Flow of events	กดปุ่มลงชื่อออกจากระบบ
Exit condition	ผู้ใช้ออกจากระบบ
Special requirement	-

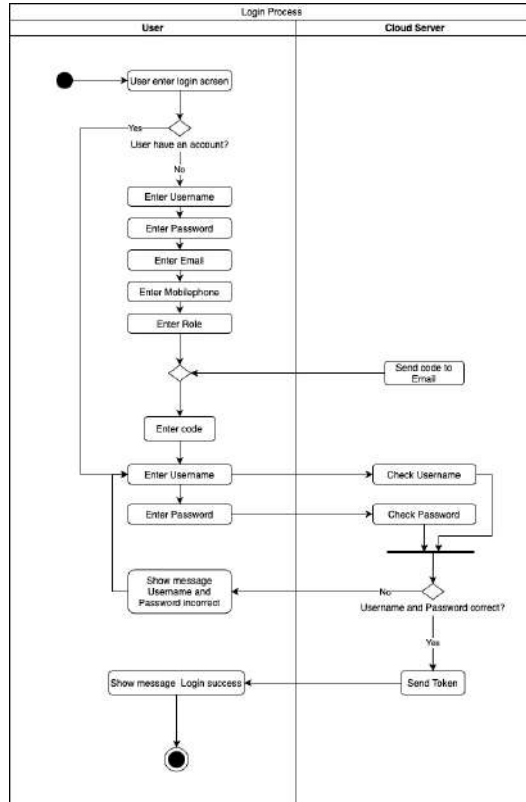
สามารถนำความต้องการเชิงหน้าที่ทั้งหมดสรุปเป็นตารางเพื่อเข้าใจได้ง่ายขึ้น ดังนี้

ตารางที่ 3.3.14 ความต้องการเชิงหน้าที่

Functional Requirement No.	Title and Description	
FR1	Title	Register
	Description	ผู้ใช้งานสามารถสมัครสมาชิกเพื่อใช้งานระบบได้
FR2	Title	Login
	Description	ผู้ใช้งานสามารถลงชื่อเข้าใช้งานระบบได้
FR3	Title	Add Floorplan
	Description	ผู้ใช้งานสามารถส่งแบบแปลนให้กับผู้ดูแลเพิ่มแบบแปลนในระบบได้
FR4	Title	Send Email
	Description	ผู้ใช้งานสามารถส่งจดหมายอิเล็กทรอนิกส์ได้
FR5	Title	Edit Foreman
	Description	ผู้ใช้งานแก้ไขรายชื่อหัวหน้าคนงานได้
FR6	Title	Manage Site
	Description	ผู้ดูแลสามารถเพิ่มโครงการให้กับผู้ใช้ที่รับผิดชอบโครงการนั้น ๆ ได้
FR7	Title	Manage Project
	Description	ผู้ใช้งานสามารถเพิ่ม ลบ แก้ไข โปรเจคได้
FR8	Title	Manage Task
	Description	ผู้ใช้งานสามารถเพิ่ม ลบ แก้ไข งานได้
FR9	Title	Manage Subtask
	Description	ผู้ใช้งานสามารถเพิ่มงานย่อยได้
FR10	Title	Comment
	Description	ผู้ใช้งานสามารถแสดงความคิดเห็นได้
FR11	Title	Notification
	Description	ผู้ใช้งานสามารถดูการแจ้งเตือนต่าง ๆ ได้
FR12	Title	Profile
	Description	ผู้ใช้งานสามารถดูข้อมูลของผู้ใช้งานได้
FR13	Title	Logout
	Description	ผู้ใช้งานออกจากระบบได้

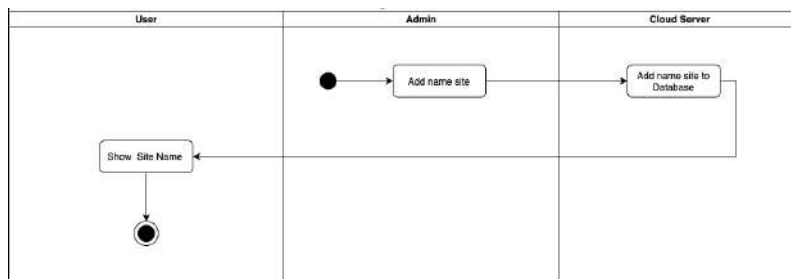
3.3.1 แผนภาพกิจกรรม

ในส่วนนี้จะแสดงถึงแผนภาพแต่ละกิจกรรม เพื่อให้เข้าใจในการทำงานได้ง่ายขึ้น ดังนี้ ในแผนภาพแรกนี้แสดงถึงส่วนของการเข้าสู่ระบบ โดยในเริ่มแรกเมื่อผู้ใช้งานต้องการล็อกอิน จะต้องมีบัญชีผู้ใช้ ถ้าหากว่าไม่มีบัญชีผู้ใช้จะต้องมีการสมัครสมาชิกก่อน โดยจะ จึงจะสามารถเข้าสู่ระบบได้สำเร็จ ดังรูป



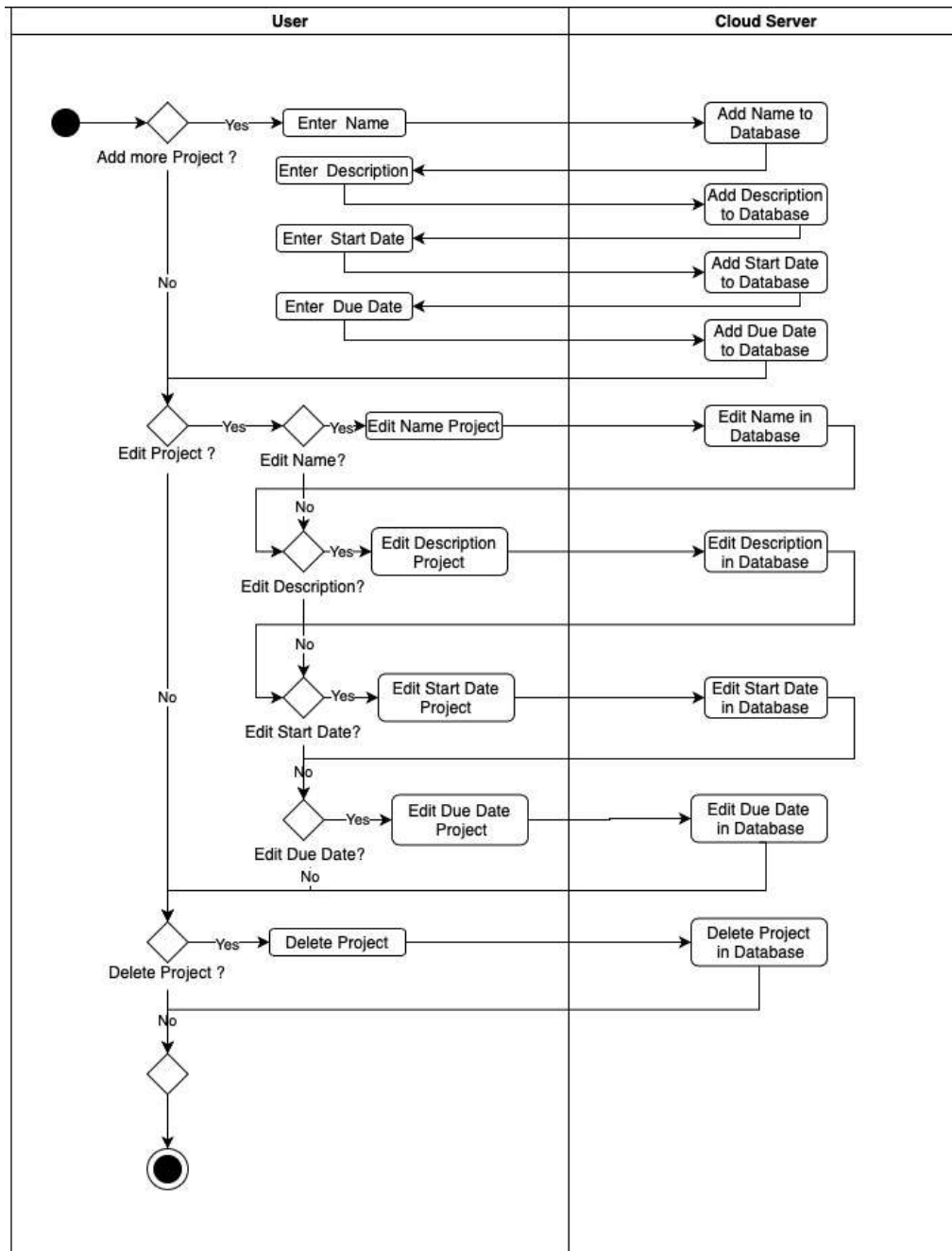
รูปที่ 3.3.1.1 แผนภาพกิจกรรมในส่วนของการเข้าสู่ระบบ

ในแผนภาพด้านล่างนี้จะแสดงถึงส่วนของการเพิ่มโครงการให้แก่ผู้ที่รับผิดชอบโครงการ ซึ่งจะเป็นการเพิ่มจากทางฝั่งผู้ดูแลระบบ ดังรูป



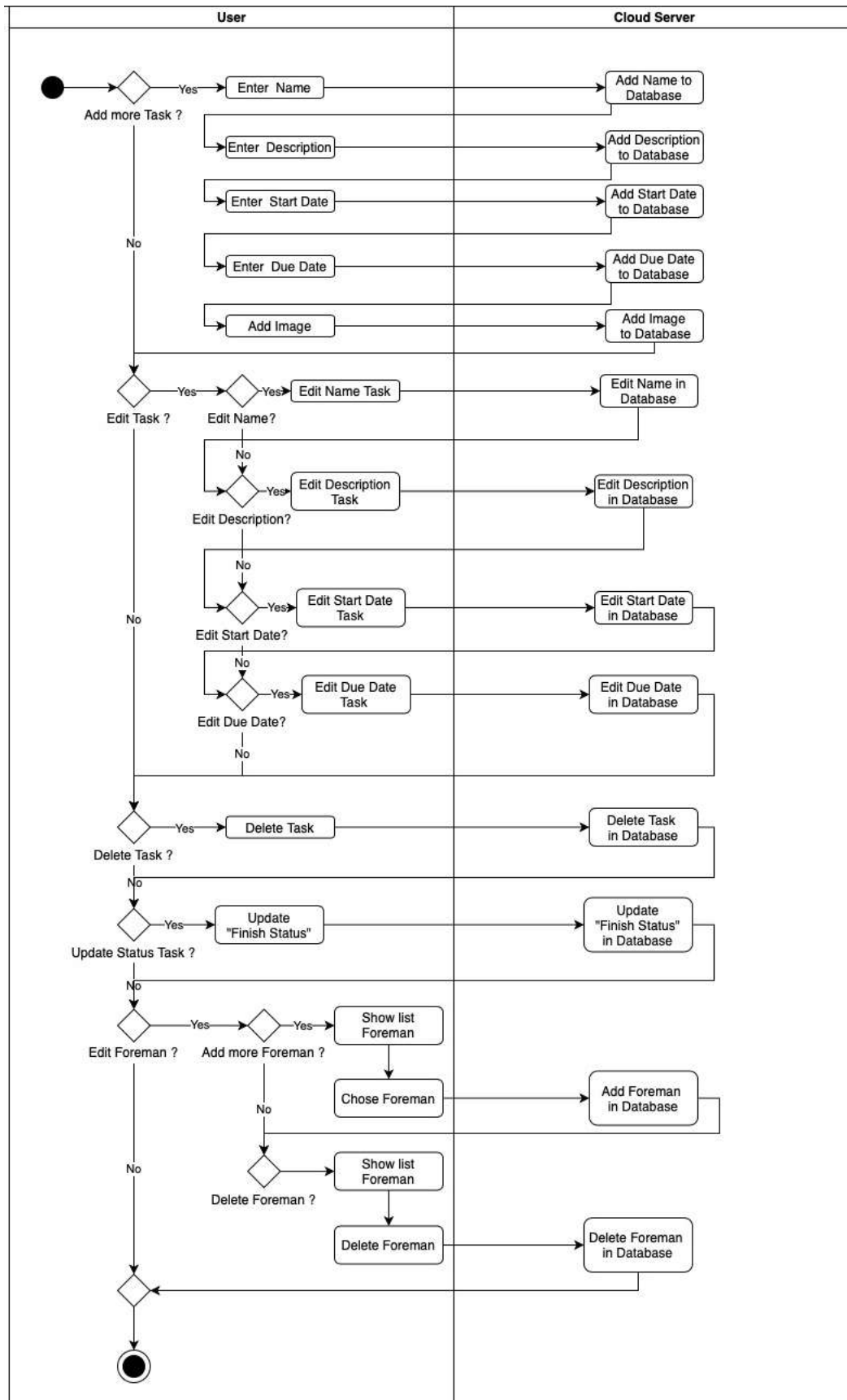
รูปที่ 3.3.1.2 แผนภาพกิจกรรมในส่วนของการเพิ่มโครงการ

แผนภาพด้านล่างนี้เป็นแผนภาพการกระทำภายในโปรเจกต์ โดยผู้ใช้งานจะสามารถเพิ่มโปรเจกต์ แก้ไขข้อมูลโปรเจกต์ ลบโปรเจกต์ ดังรูป



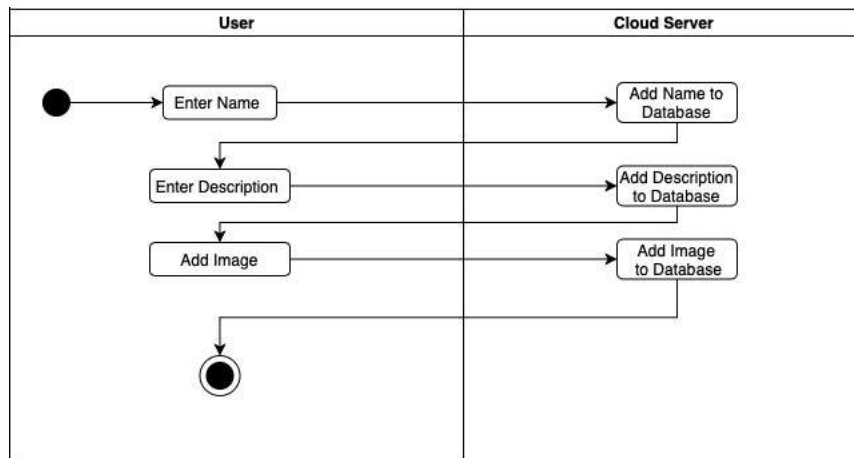
รูปที่ 3.3.1.3 แผนภาพกิจกรรมในส่วนของการกระทำภายในโปรเจกต์

แผนภาพด้านล่างนี้เป็นแผนภาพที่แสดงในส่วนของการกระทำภายในงาน ซึ่งผู้ใช้งานสามารถ เพิ่มงาน แก้ไขงาน ลบงาน อัปเดตสถานะงาน และ แก้ไขในส่วนรายชื่อหัวหน้างานที่รับผิดชอบในงานนั้น ๆ ได้ ดังรูป



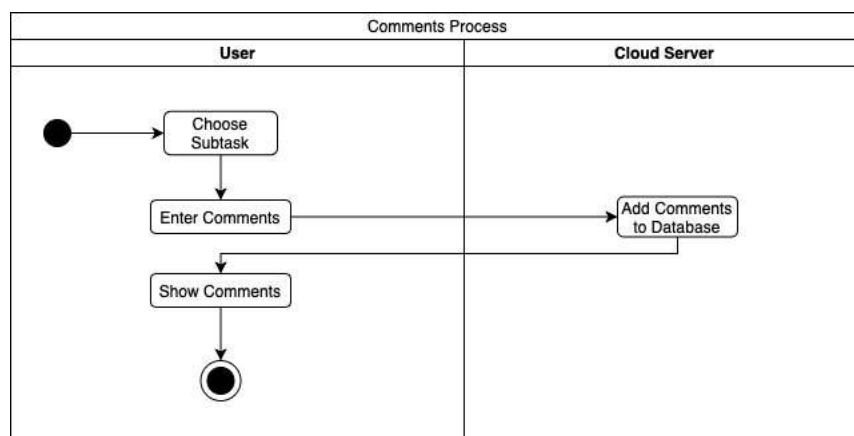
รูปที่ 3.3.1.4 แผนภาพกิจกรรมในส่วนของการกระทำภายในงาน

แผนภาพด้านล่างนี้เป็นแผนภาพในส่วนของการเพิ่มงานย่อย หรือ เพิ่มปัญหาที่พบ โดยผู้ใช้สามารถเพิ่มงานย่อยได้โดยการกรอก ชื่องานย่อย คำอธิบาย และเพิ่มภาพ ดังรูป



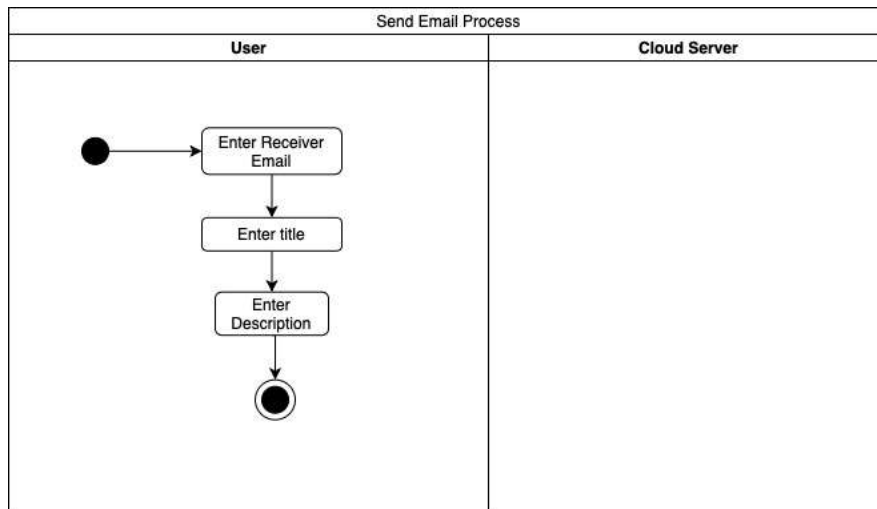
รูปที่ 3.3.1.5 แผนภาพกิจกรรมในส่วนของการเพิ่มงานย่อย

แผนภาพด้านล่างนี้เป็นแผนภาพในส่วนการแสดงความคิดเห็นโดยผู้ใช้งานสามารถเลือกงานที่ต้องการแสดงความคิดเห็น และแสดงความคิดเห็นลงบนงาน จะทำให้มีความคิดเห็นแสดงขึ้นมา ดังรูป



รูปที่ 3.3.1.6 แผนภาพกิจกรรมในส่วนของการแสดงความคิดเห็น

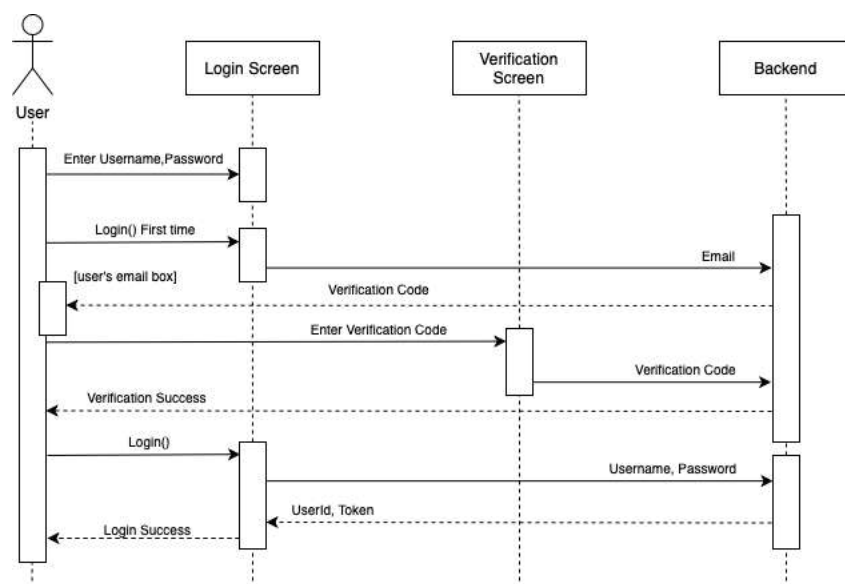
แผนภาพด้านล่างนี้เป็นแผนภาพแสดงในส่วนการส่งจดหมายอิเล็กทรอนิกส์ โดยผู้ใช้งานสามารถเพิ่มรายชื่ออีเมลที่ผู้ใช้งานต้องการส่งถึง โดยจะเพิ่มทั้งหัวข้อและคำอธิบายต่าง ๆ ดังรูป



รูปที่ 3.3.1.7 แผนภาพกิจกรรมในส่วนของการส่งจดหมายอิเล็กทรอนิกส์

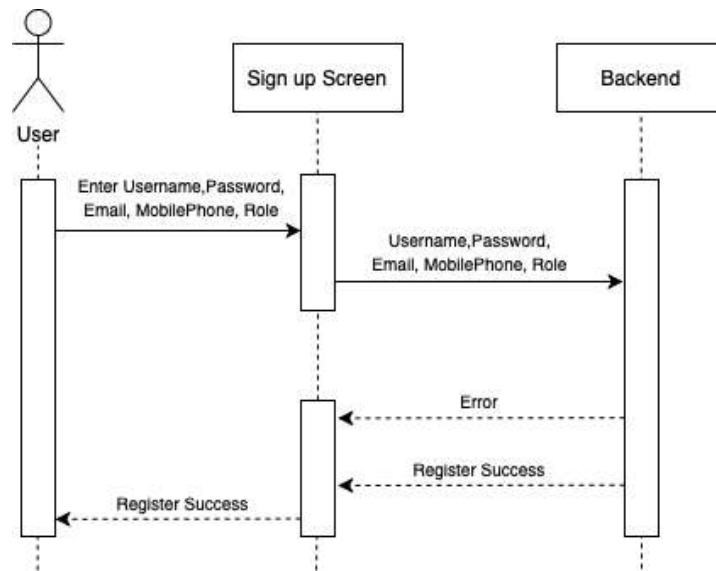
3.3.2 แผนภาพลำดับ

ในส่วนนี้จะแสดงแผนภาพลำดับ เพื่อแสดงกิจกรรมที่เกิดขึ้นจากคลาสหรือวัตถุในแผนผังการทำงาน ให้เข้าใจได้ง่ายขึ้น ดังนี้ แผนภาพแรกเป็นแผนภาพลำดับในส่วนของการเข้าสู่ระบบโดย เมื่อผู้ใช้งานลงชื่อเข้าสู่ระบบโดยเมื่อกรอกชื่อผู้ใช้งาน และรหัสผ่าน ทางหลังบ้านจะทำการส่งโทเคนไปสู่นำหน้าบ้านเพื่อทำการเข้าสู่ระบบให้กับผู้ใช้งานได้สำเร็จ ดังรูป



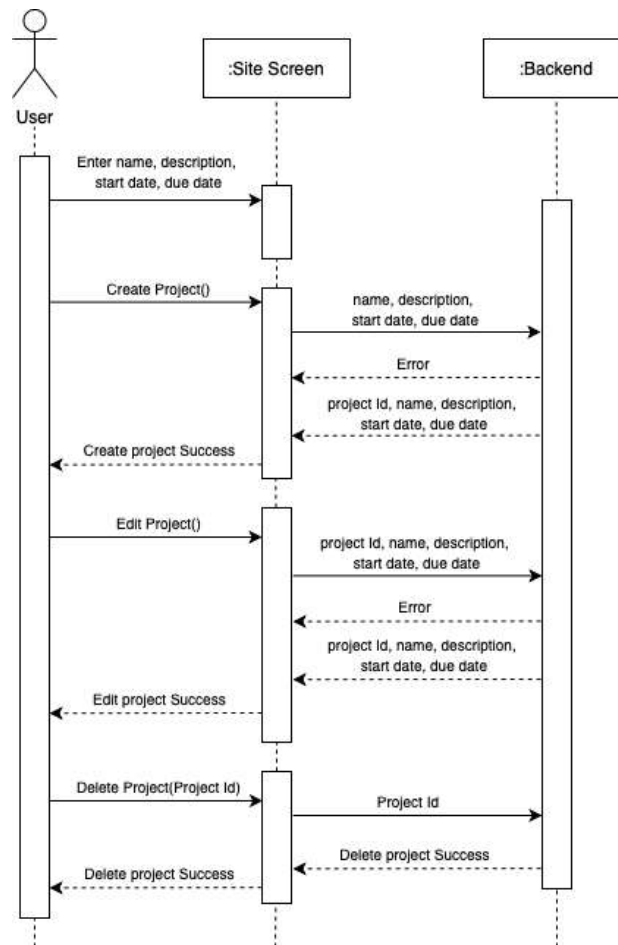
รูปที่ 3.3.2.1 แผนภาพลำดับในส่วนของการเข้าสู่ระบบ

แผนภาพด้านล่างนี้เป็นแผนภาพแสดงในส่วนของการลงทะเบียนเข้าสู่ระบบโดย ผู้ใช้งานต้องกรอกชื่อผู้ใช้งาน รหัสผ่าน อีเมล เบอร์โทรศัพท์ และตำแหน่งของผู้ใช้งานในหน้าจอส่วนของการลงทะเบียน เมื่อทำการกรอกครบเรียบร้อยแล้ว ข้อมูลจะส่งไปยังหลังบ้าน และหลังบ้านเห็นว่าลงทะเบียนสำเร็จ ก็จะเห็นได้ว่าในขั้นตอนนี้ก็สำเร็จ ดังรูป



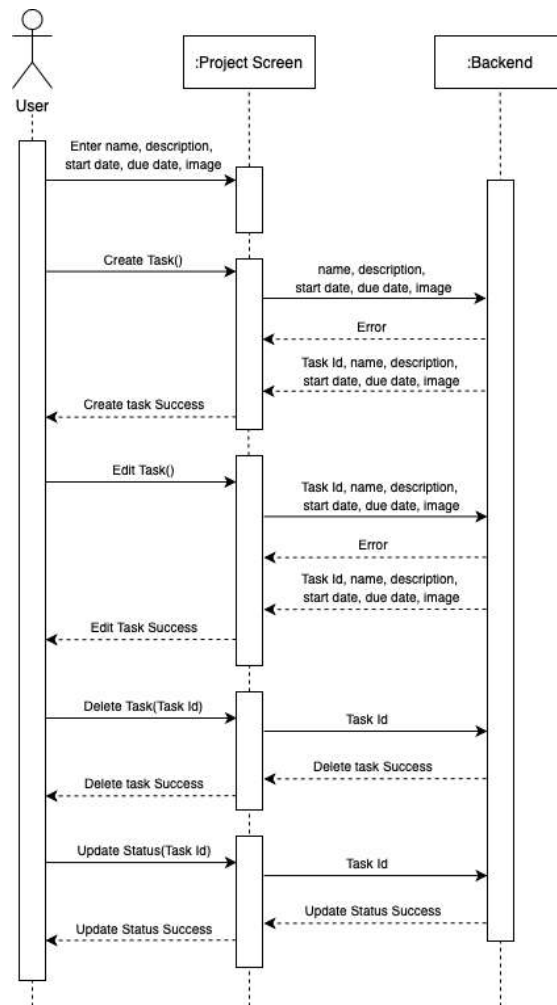
รูปที่ 3.3.2.2 แผนภาพลำดับในส่วนของการลงทะเบียนเข้าสู่ระบบ

แผนภาพด้านล่างนี้เป็นแผนภาพแสดงในส่วนของการกระทำภายในโปรเจค ซึ่งผู้ใช้งานสามารถเพิ่มโปรเจคในโครงการโดยการ กรอก ชื่อโปรเจค คำอธิบายโปรเจค วันที่เริ่มต้น และวันที่สิ้นสุดในการทำโปรเจคในส่วนของหน้าโครงการ อีกทั้งผู้ใช้งานสามารถแก้ไขข้อมูลภายในโปรเจคได้ อีกทั้งผู้ใช้งานยังสามารถลบโปรเจคได้อีกด้วย ดังรูป



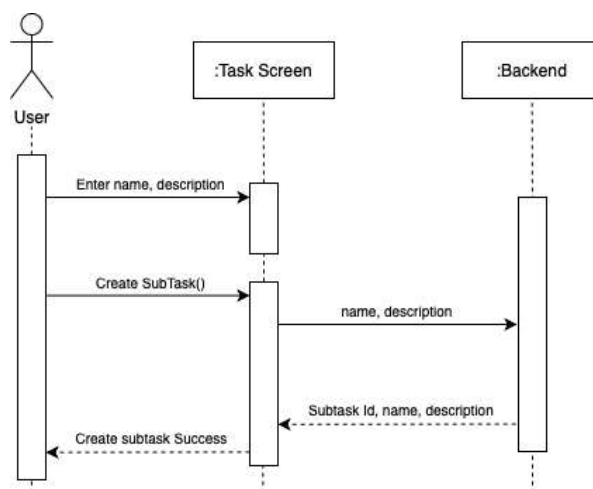
รูปที่ 3.3.2.3 แผนภาพลำดับในส่วนของการทำงานภายในโปรเจค

แผนภาพด้านล่างเป็นแผนภาพแสดงในส่วนของการกระทำภายในงาน โดยผู้ใช้งานสามารถเพิ่มงานภายในโปรเจค โดยการกรอก ชื่องาน คำอธิบายงาน และเพิ่มรูปภาพของงาน อีกทั้งผู้ใช้งานสามารถ แก้ไขข้อมูลภายในงาน ลบงานที่อยู่ภายในหน้าโปรเจค อีกทั้งยังสามารถอัปเดตสถานะของงาน โดยการเลือกงานที่ต้องการอัปเดตสถานะ และเมื่อทำการอัปเดตสถานะ ข้อมูลจะถูกส่งไปยังหลังบ้าน ดังรูป



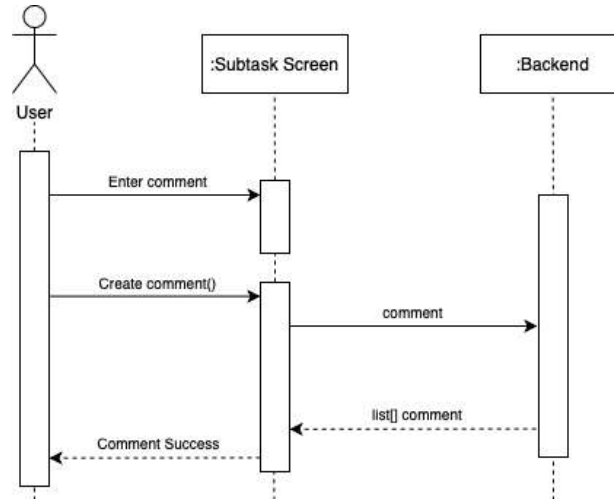
รูปที่ 3.3.2.4 แผนภาพลำดับในส่วนของการกระทำภายในงาน

แผนภาพด้านล่างนี้เป็นแผนภาพในส่วนของการเพิ่มงานย่อยในงานที่เลือก โดยผู้ใช้งานจะเพิ่มชื่องานย่อย และคำอธิบาย ภายในหน้างาน โดยเมื่อผู้ใช้งานกรอกข้อมูลครบจะส่งข้อมูลไปยังหลังบ้าน ดังรูป



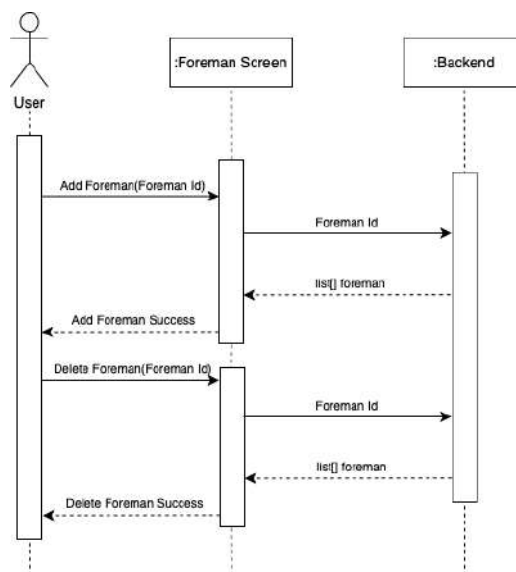
รูปที่ 3.3.2.5 แผนภาพลำดับในส่วนของการเพิ่มในงานย่อย

แผนภาพด้านล่างนี้เป็นแผนภาพในส่วนของการแสดงความคิดเห็นในหน้าแสดงงานย่อยทั้งหมด เมื่อผู้ใช้งานเพิ่มความคิดเห็นลงบนงานย่อยที่ต้องการแสดงความคิดเห็นในหน้าของงานย่อย ข้อมูลจะส่งไปยังหลังบ้าน และส่งกลับมายังหน้าบ้านเพื่อทำการแสดงความคิดเห็นที่ผู้ใช้งานได้แสดงความคิดเห็น ดังรูป



รูปที่ 3.3.2.6 แผนภาพลำดับในส่วนของการแสดงความคิดเห็น

แผนภาพด้านล่างเป็นแผนภาพในส่วนของการแก้ไขรายชื่อหัวหน้าคนงาน ผู้ใช้งานสามารถเพิ่มรายชื่อ โดยการเลือกที่รหัสของหัวหน้าคนงาน (Foreman Id) และจะส่งข้อมูลไปยังหลังบ้าน และทำการอัปเดตรายชื่อมาแสดงที่หน้าบ้าน อีกทั้งผู้ใช้งานสามารถ ลบรายชื่อหัวหน้าคนงาน โดยการเลือกรหัสหัวหน้าคนงาน ข้อมูลจะทำการส่งไปยังหลังบ้านเพื่อทำการอัปเดต และจะนำข้อมูลที่ได้ผ่านการอัปเดตแล้วมาแสดงยังหน้าบ้าน ในหน้าแก้ไขรายชื่อหัวหน้าคนงาน ดังรูป



รูปที่ 3.3.2.7 แผนภาพลำดับในส่วนของการแก้ไขในส่วนรายชื่อหัวหน้าคนงาน

จากแผนภาพทั้งหมดที่ได้แสดงมา จะเห็นว่าแต่ละกระบวนการ จะมีผู้ใช้งานที่สามารถเข้าถึงสิทธิ์ที่แตกต่างกัน โดยจะแบ่งออกเป็น 3 ส่วนที่เห็นชัดคือ ส่วนของ ผู้ดูแลระบบ ผู้จัดการโปรเจค และหัวหน้าคนงาน จะสรุปเป็นตารางสิทธิ์ของผู้ใช้งานดังนี้

ตารางที่ 3.3.15 ตารางเปรียบเทียบสิทธิ์ในการใช้งานแต่ละฟังก์ชันภายในระบบของแต่ละผู้ใช้งาน

ฟังก์ชัน	ผู้ใช้งาน		
	ผู้ดูแลระบบ	ผู้จัดการโปรเจค	หัวหน้าคนงาน
1. ระบบการแจ้งเตือน	○	○	○
2.สามารถเพิ่ม ลบ แก้ไข ข้อมูลภายในโครงการ	○	○	
3.สามารถเพิ่ม ลบ แก้ไข ข้อมูลภายในโปรเจค	○	○	
4.สามารถเพิ่ม ลบ แก้ไข ข้อมูลภายในงาน	○	○	
5.สามารถเพิ่ม ข้อมูลภายในงานย่อย	○	○	○
6.สามารถแสดงความคิดเห็นเพิ่มเติมในงานย่อย	○	○	○
7.สามารถแก้ไขรายชื่อหัวหน้าคนงาน	○	○	○

3.4 ความต้องการเชิงคุณภาพ

ในส่วนนี้จะแสดงถึง ความต้องการในส่วนเชิงคุณภาพ หรือ Non-Functional Requirement ดังนี้

ตารางที่ 3.4.1 ตารางแสดงความต้องการเชิงคุณภาพ

Non-Functional Requirement No.	Title and Description	
NFR1	Title	Timestamp Watermark
	Description	หลังผู้ใช้อัปโหลดภาพแล้ว ภาพจะมีการเขียนวันที่ทับลงไปโดยอัตโนมัติ
NFR2	Title	Resize Image
	Description	ระบบบีบภาพอัตโนมัติ เพื่อไม่ให้ภาพขนาดใหญ่จนเกินไป และช่วยให้การโหลดรูปเป็นไปอย่างรวดเร็ว
NFR3	Title	Monitoring Backend
	Description	สามารถตรวจสอบ Log / การทำงานทั้งหมดของส่วนหลังบ้านได้ตลอดเวลา
NFR4	Title	Storage Security
	Description	ข้อมูลและภาพจะต้องเป็นความลับและสามารถถูกเรียกได้โดยเจ้าของ และผู้เกี่ยวข้องเท่านั้น
NFR5	Title	Scalability
	Description	สามารถรองรับการประมวลผลที่มากขึ้นได้ในอนาคต เมื่อมีผู้ใช้และจำนวนงานมากขึ้น
NFR6	Title	Fast Loading
	Description	ระบบสามารถประมวลผลได้รวดเร็วและตอบสนองได้ทันท่วงทีต่อการใช้งาน

3.5 การออกแบบส่วนต่อประสานผู้ใช้

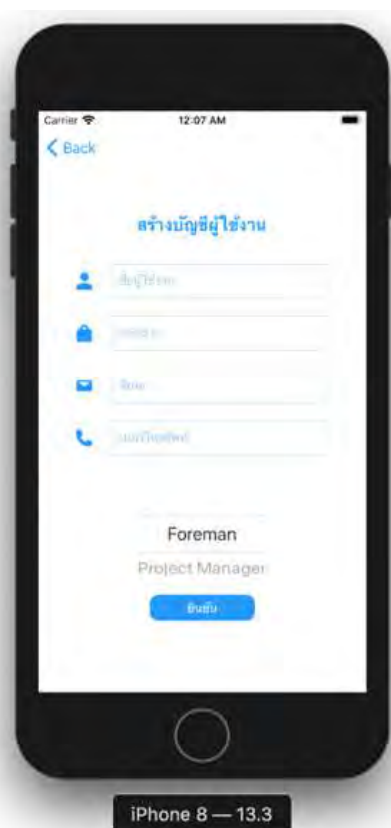
การออกแบบส่วนต่อประสานผู้ใช้จัดทำขึ้นเพื่อเป็นแนวทางในการพัฒนาระบบ โดยยึดหลักการออกแบบให้ผู้ใช้สามารถเข้าใจการใช้งานได้อย่างรวดเร็ว โดยการออกแบบของระบบช่วยตรวจสอบคุณภาพที่ใช้ในกระบวนการก่อสร้าง

3.5.1 หน้าลงทะเบียน

หน้าลงทะเบียนเป็นหน้าที่ประกอบไปด้วยกล่องข้อความที่ให้ผู้ใช้งานกรอก รหัสผู้ใช้งาน รหัสผ่าน อีเมล เบอร์โทรศัพท์ และมีให้เลือกหน้าที่ของผู้ใช้งานว่า หัวหน้าคนงาน หรือ ผู้จัดการโปรเจกต์

3.5.2 หน้าลงชื่อเข้าสู่ระบบ

หน้าแรกประกอบไปด้วยกล่องข้อความสำหรับกรอก รหัสผู้ใช้งาน รหัสผ่าน และ ปุ่มกดลงชื่อเข้าใช้ ซึ่งถ้าผู้ใช้งานยังไม่เคยเป็นสมาชิก สามารถกดข้อความสมัครสมาชิกเพื่อไปยังหน้าสมัครสมาชิก



รูปที่ 3.5.1 ส่วนต่อประสานผู้ใช้หน้าลงทะเบียน



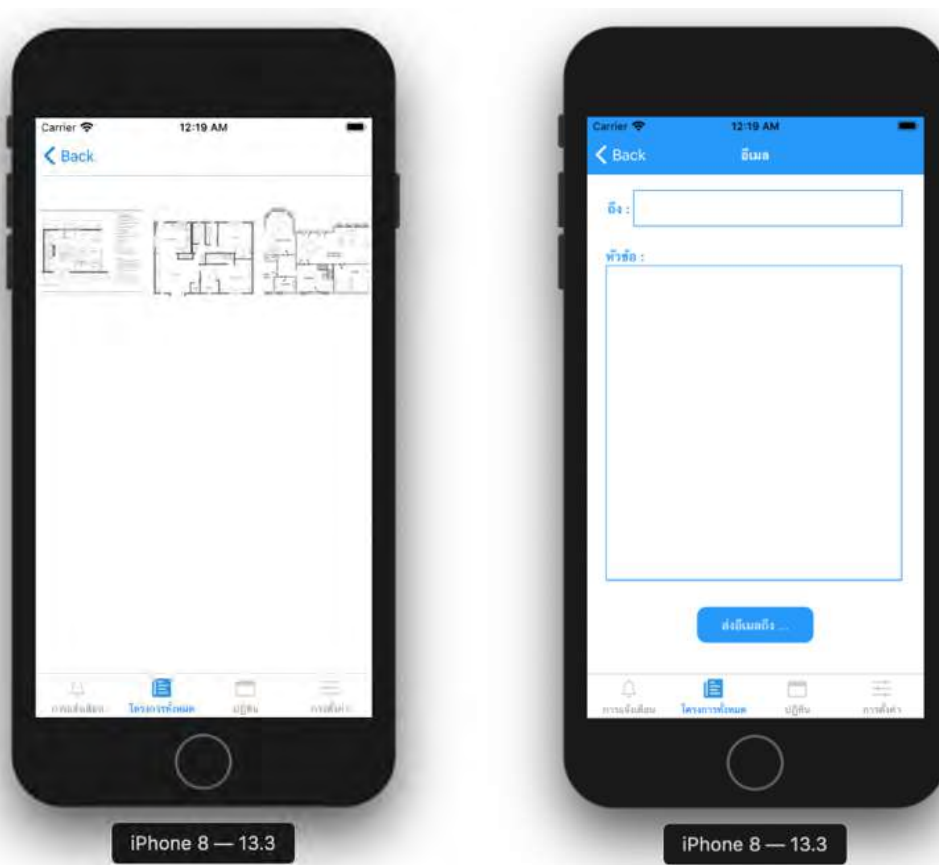
รูปที่ 3.5.2 ส่วนต่อประสานผู้ใช้หน้าเข้าสู่ระบบ

3.5.3 หน้าแปลน

หน้าแปลน จะประกอบไปด้วยรูป Floorplan ที่อยู่ในโปรเจกต์ทั้งหมด

3.5.4 หน้าอีเมล

หน้าอีเมลประกอบไปด้วยกล่องข้อความให้ผู้ใช้งานกรอกชื่ออีเมลที่ต้องการส่งถึง และรายละเอียดในการที่จะส่งข้อความ



รูปที่ 3.5.3 ส่วนต่อประสานผู้ใช้หน้าแปลน

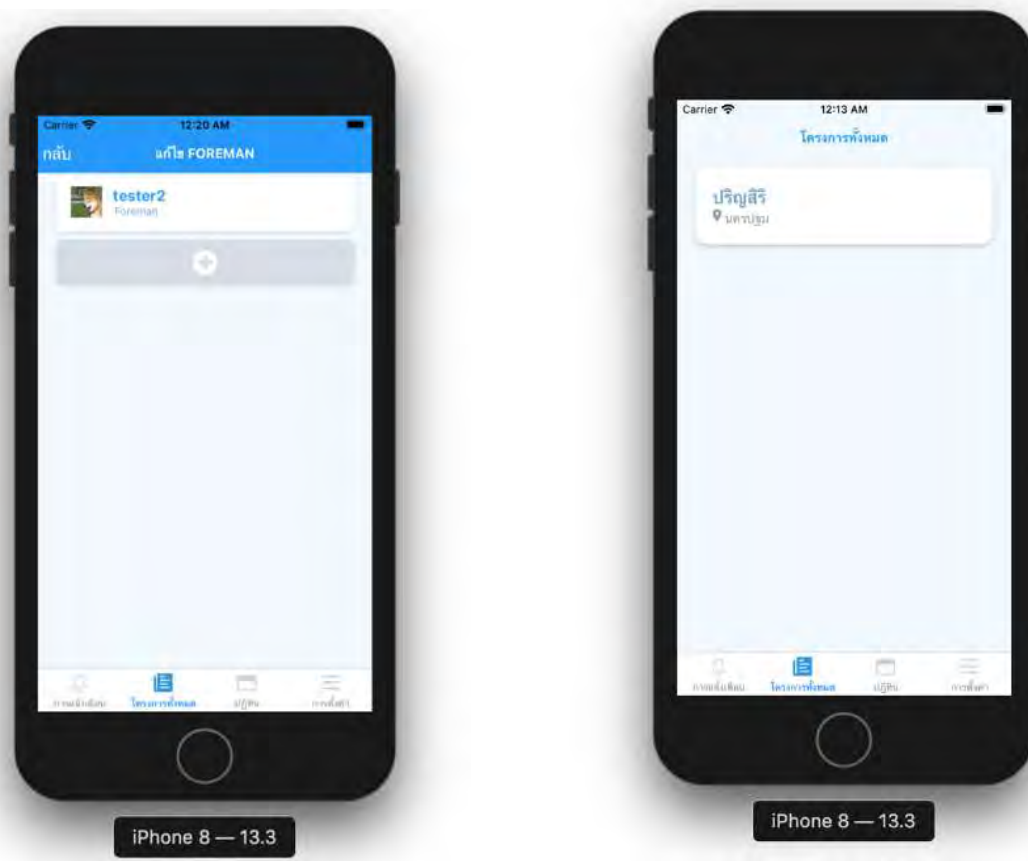
รูปที่ 3.5.4 ส่วนต่อประสานผู้ใช้หน้าอีเมล

3.5.5 หน้าแก้ไข Foreman หรือ แก้ไขรายชื่อหัวหน้าคนงาน

หน้าแก้ไขรายชื่อหัวหน้าคนงานประกอบไปด้วย ลิสต์รายชื่อหัวหน้าคนงานซึ่งผู้ใช้สามารถเพิ่ม ลบ รายชื่อของหัวหน้าคนงานใหม่ที่จะรับผิดชอบภายในโปรเจกต์นั้น ๆ ได้

3.5.6 หน้าโครงการทั้งหมด

หน้าแสดงรายละเอียดโครงการทั้งหมดที่ผู้ใช้งานรับผิดชอบ ซึ่งจะสามารถเชื่อมต่อไปยังโครงการต่างๆ เมื่อผู้ใช้งานได้กดเลือก



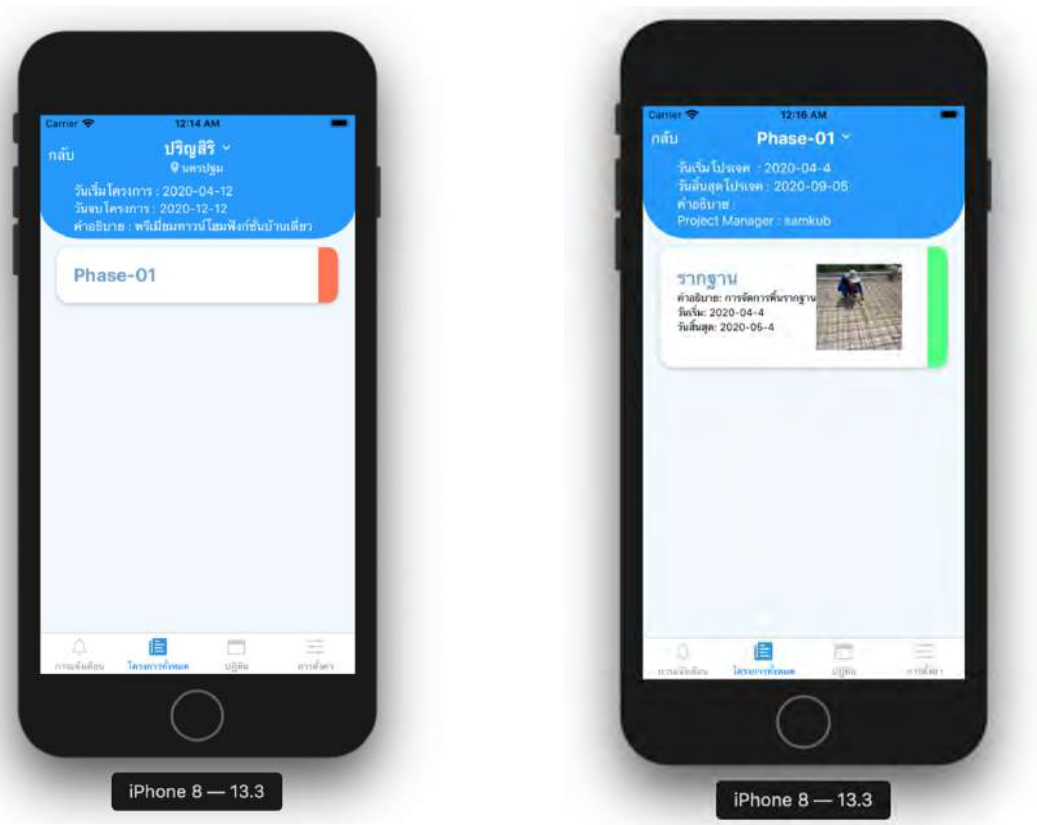
รูปที่ 3.5.5 ส่วนต่อประสานผู้ใช้หน้าแก้ไขรายชื่อหัวหน้าคนงาน รูปที่ 3.5.6 ส่วนต่อประสานผู้ใช้หน้าโครงการทั้งหมด

3.5.7 หน้าโครงการ

หน้าแสดงรายละเอียดโครงการต่าง ๆ มีทั้ง วันที่เริ่ม และสิ้นสุดโครงการ คำอธิบาย อีกทั้งยังมีรายชื่อโปรเจกต์ที่อยู่ภายใต้โครงการที่เลือกจากหน้าโครงการทั้งหมด อีกทั้งยังสามารถเพิ่มโปรเจกต์ใหม่ได้ และดูแบบแปลนสำหรับโครงการนี้ได้

3.5.8 หน้าโปรเจกต์

หน้าแสดงรายละเอียดโปรเจกต์ต่าง ๆ มีทั้ง วันที่เริ่ม และสิ้นสุดโครงการ คำอธิบาย อีกทั้งยังมีลิสต์ที่แสดงรายละเอียดงานที่อยู่ภายใต้โปรเจกต์ที่ได้เลือกมาทั้งหมด โดยหน้านี้สามารถเพิ่มงานใหม่ได้ อีกทั้งยังสามารถเลือกสถานะ “เสร็จ” สำหรับงานที่เลือกได้ เพื่อบ่งบอกว่างานชิ้นนี้ได้สำเร็จลุล่วงไปแล้ว



รูปที่ 3.5.7 ส่วนต่อประสานผู้ใช้หน้าโครงการ รูปที่ 3.5.8 ส่วนต่อประสานผู้ใช้หน้าโปรเจก

3.5.9 หน่วยงาน

หน้านี้ประกอบด้วย รายละเอียดงานที่อยู่ภายใต้งานที่ได้เลือกมา อีกทั้งยังสามารถเพิ่มหัวข้อย่อยใหม่ได้

3.5.10 หน้าปัญหา

หน้านี้ประกอบไปด้วยรายละเอียดงานอยู่ที่ได้เลือกมา อีกทั้งผู้ใช้งานสามารถแสดงความคิดเห็นกับผู้อื่น ๆ ที่อยู่ภายในงานย่อยนี้ได้



iPhone 8 — 13.3

รูปที่ 3.5.9 ส่วนต่อประสานผู้ใช้พนักงาน



iPhone 8 — 13.3

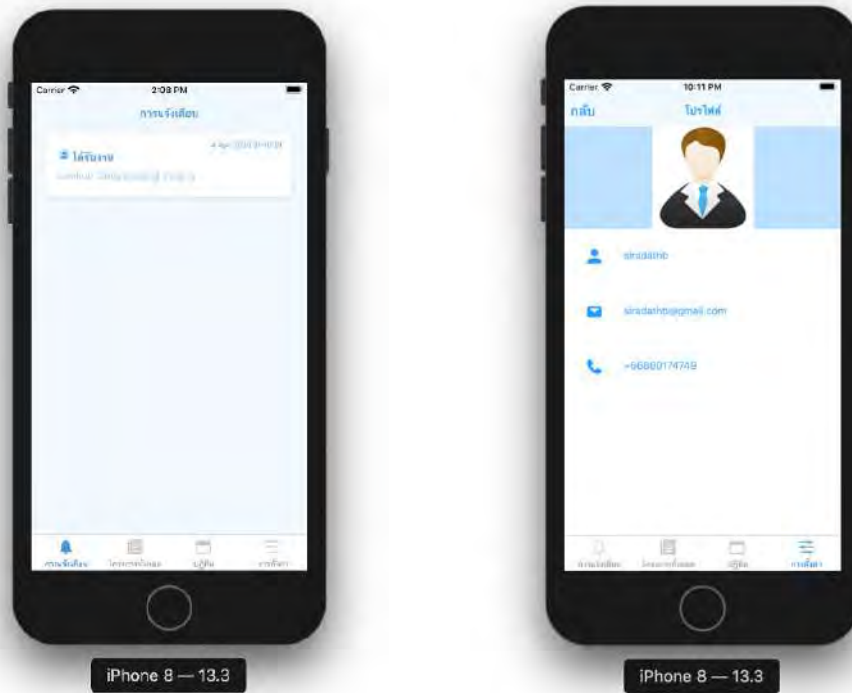
รูปที่ 3.5.10 ส่วนต่อประสานผู้ใช้หน้าปัญหา

3.5.11 หน้าการแจ้งเตือน

หน้าการแจ้งเตือนเป็นหน้าที่ประกอบไปด้วยลิสต์การแจ้งเตือนของผู้ใช้งานทั้งหมด

3.5.12 หน้าการตั้งค่า

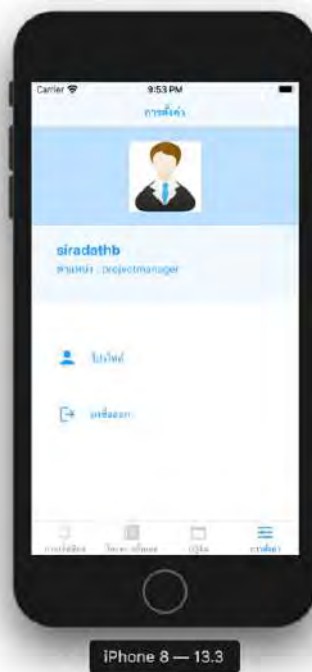
หน้าการตั้งค่าประกอบไปด้วยข้อมูลของผู้ใช้งาน เช่น ชื่อผู้ใช้งาน อีเมล และ เบอร์โทรศัพท์ ของผู้ใช้งาน



รูปที่ 3.5.11 ส่วนต่อประสานผู้ใช้หน้าการแจ้งเตือน รูปที่ 3.5.12 ส่วนต่อประสานผู้ใช้หน้าการตั้งค่า

3.5.13 หน้าออกจากระบบ

หน้าออกจากระบบ ผู้ใช้งานสามารถกด ลงชื่อออก เพื่อออกจากระบบ



รูปที่ 3.5.13 ส่วนต่อประสานผู้ใช้หน้าการออกจากระบบ

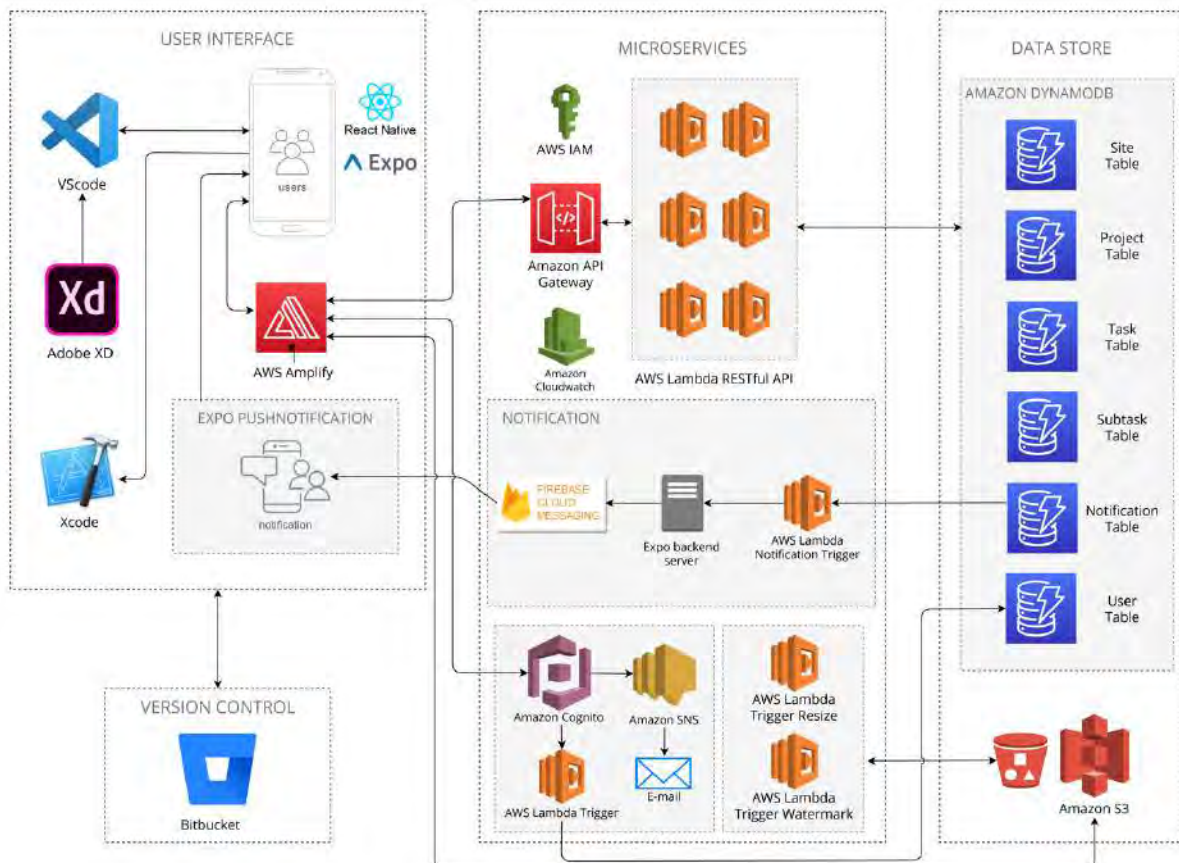
บทที่ 4

การพัฒนาและทดสอบระบบ

ในบทนี้จะกล่าวถึงการพัฒนาแอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง โดยจะกล่าวถึงเทคนิคการพัฒนาระบบและ การทดสอบการใช้งานของระบบเบื้องต้น ซึ่งมีรายละเอียดดังนี้

4.1 เทคนิคการพัฒนาระบบ

การพัฒนาแอปพลิเคชันช่วยในการตรวจสอบกระบวนการก่อสร้างแบ่งออกเป็นสองส่วน ได้แก่ ส่วนหน้าบ้านและส่วนหลังบ้าน ส่วนหน้าบ้านเป็นส่วนต่อประสานกับผู้ใช้ และส่วนหลังบ้านจะเป็นส่วนที่เชื่อมต่อกับฐานข้อมูล การทำงานของฟังก์ชันต่าง ๆ



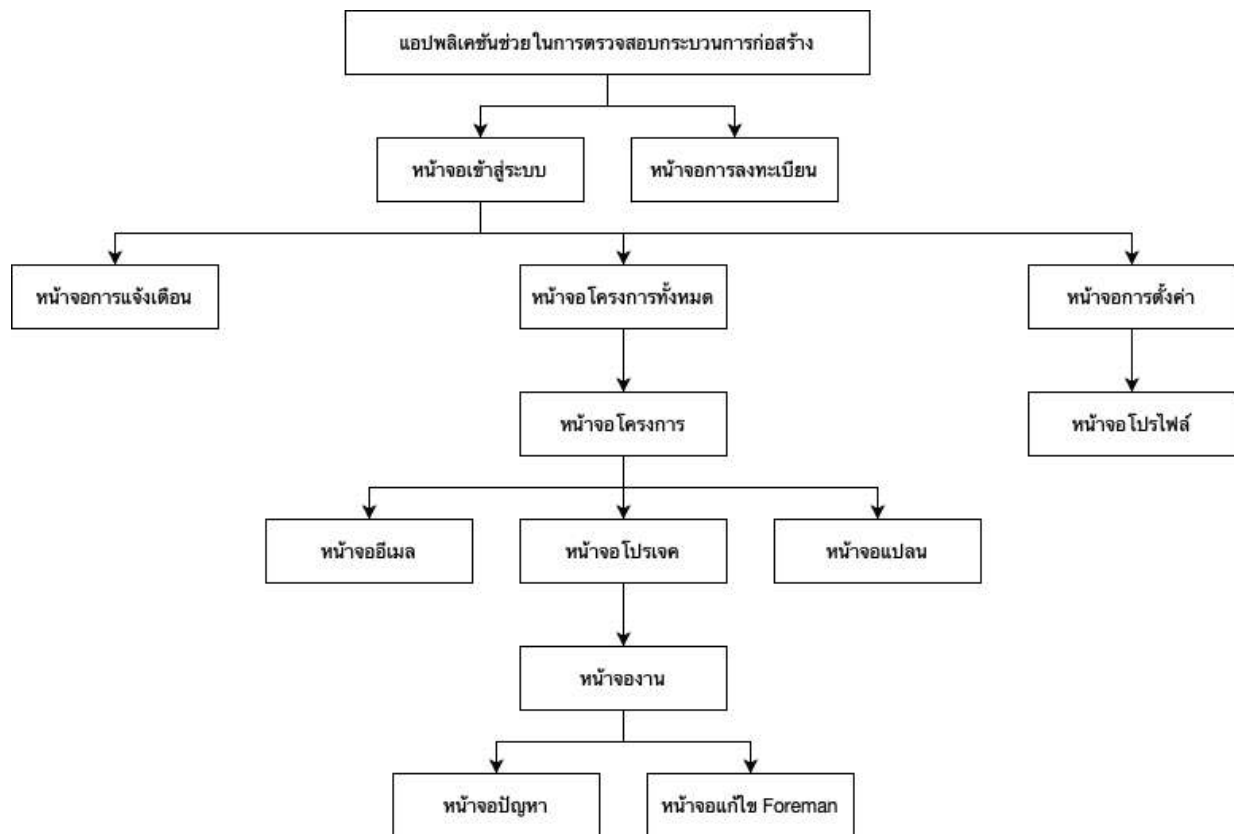
รูปที่ 4.1.1 ภาพรวมของระบบ

การพัฒนาแอปพลิเคชันช่วยในการตรวจสอบกระบวนการก่อสร้างส่วนหน้าบ้าน เป็นส่วนต่อประสานกับผู้ใช้งาน โดยตั้งแต่การรวบรวมความต้องการของผู้ใช้งานมา จะนำมาออกแบบต้นแบบแอปพลิเคชัน ด้วยโปรแกรม Adobe XD จากนั้นจะนำมาเขียนในรูปของภาษาจาวาสคริปต์ด้วยโปรแกรม VScode ผ่าน Framework ที่มีชื่อว่า React native และ Expo ซึ่งเป็น Framework สำหรับภาษาจาวาสคริปต์ ใช้เพื่อให้สามารถพัฒนาแอปพลิเคชันบนโทรศัพท์มือถือได้

แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้างมีทั้งหมด 13 หน้าจอ แสดงเปรียบเทียบกับความต้องการเชิงหน้าที่ได้ดังตารางที่ 4.1.1

ตารางที่ 4.1.1 ตารางแสดงหน้าจอเปรียบเทียบกับความต้องการเชิงหน้าที่

ลำดับ	หน้าจอ	ความต้องการเชิงหน้าที่
1	หน้าจอการลงทะเบียน	FR1 Register
2	หน้าจอเข้าสู่ระบบ	FR2 Login
3	หน้าจอแปลน	FR3 Add Floorplan
4	หน้าจออีเมล	FR4 Send Email
5	หน้าจอแก้ไข Foreman	FR5 Edit Foreman
6	หน้าจอโครงการทั้งหมด	FR6 Manage Site
7	หน้าจอโครงการ	FR7 Manage Project
8	หน้าจอโปรเจค	FR8 Manage Task
9	หน้าจางาน	FR9 Manage Subtask
10	หน้าจอปัญหา	FR10 Comment
11	หน้าจอการแจ้งเตือน	FR11 Notification
12	หน้าจอโปรไฟล์	FR12 Profile
13	หน้าจอการตั้งค่า	FR13 Logout



รูปที่ 4.1.2 แผนภาพแสดงโครงสร้างหน้าจอของแอปพลิเคชัน

4.1.1 การพัฒนาหน้าจอลงทะเบียน

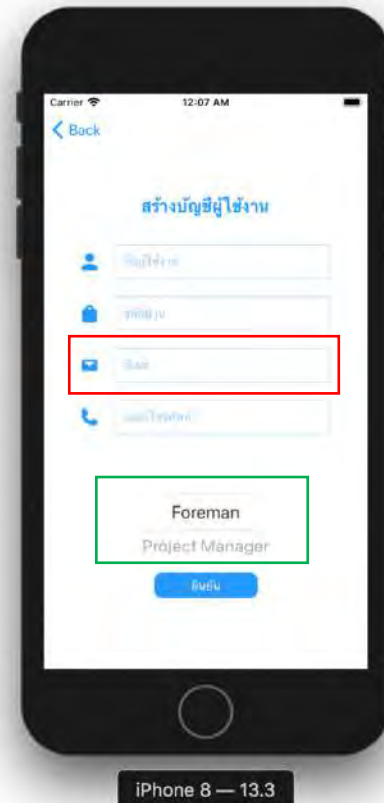
หน้าลงทะเบียน การทำส่วนต่อประสานกับผู้ใช้งานถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่ 4.1.1.1 และได้ผลลัพธ์ออกมาในรูป 4.1.1.2 ซึ่งในหน้าลงทะเบียน ผู้ใช้งานจะต้องกรอกข้อมูลที่จำเป็น ได้แก่ ชื่อผู้ใช้งาน รหัสผ่าน อีเมล เบอร์โทรศัพท์ และตำแหน่ง จากนั้นกดปุ่มยืนยัน

```

<View style={styles.container}>
  <View style={{ flex: 0.2, justifyContent: 'center',
    alignItems: 'center', marginBottom: 80 }}>
    <Text style={{ color: '#2699FB', fontWeight: 'bold',
      fontSize: 20 }}><สร้างบัญชีใช้งาน</Text>
  </View>
  <View style={styles.form}>
    <View style={{ flexDirection: 'row', alignItems:
      'center', justifyContent: 'center', marginBottom:
      20 }}>
      <Image style={{ width: 18, height: 20 }} source =
        {require('../assets/images/Editprofile_profile.png')} />
      <TextInput placeholder="ชื่อผู้ใช้งาน" placeholderTextColor=
        "#9FCDF5" style={{ height: 40, width: 250, borderColor:
        '#BCE0FD', borderWidth: 1, padding: 10, color:
        '#2699FB', margin: 10 }} onChangeText={text =>
        this.setState({ username: text })} autoCapitalize='none'
        value={this.state.username} />
    </View>
    <View style={{ flexDirection: 'row', alignItems: 'center',
      justifyContent: 'center', margin: 20 }}>
      <Image style={{ width: 18, height: 20 }} source =
        {require('../assets/images/Privacy.png')} />
      <TextInput placeholder="รหัสผ่าน" placeholderTextColor =
        "#9FCDF5" secureTextEntry={true} style={{ height: 40,
        width: 250, borderColor: '#BCE0FD', borderWidth: 1,
        padding: 10, color: '#2699FB', margin: 10 }}
        onChangeText={text => this.setState({ password: text })}
        autoCapitalize='none' value={this.state.password} />
    </View>
    <View style={{ flexDirection: 'row', alignItems: 'center',
      justifyContent: 'center', margin: 20 }}>
      <Image style={{ width: 18, height: 13 }} source =
        {require('../assets/images/Mail.png')} />
      <TextInput placeholder="อีเมล" placeholderTextColor =
        "#9FCDF5" style={{ height: 40, width: 250, borderColor:
        '#BCE0FD', borderWidth: 1, padding: 10, color:
        '#2699FB', margin: 10 }} onChangeText={text =>
        this.setState({ email: text })} autoCapitalize='none'
        value={this.state.email} />
    </View>
    <View style={{ flexDirection: 'row', alignItems: 'center',
      justifyContent: 'center', margin: 20 }}>
      <Image style={{ width: 18, height: 20 }} source =
        {require('../assets/images/Editprofile_Phone.png')} />
      <TextInput placeholder="เบอร์โทรศัพท์" placeholderTextColor
        ="#9FCDF5" style={{ height: 40, width: 250, borderColor:
        '#BCE0FD', borderWidth: 1, padding: 10, color:
        '#2699FB', margin: 10 }} onChangeText={text =>
        this.setState({ phone_number: text })} value =
        {this.state.phone_number} />
    </View>
    <View style={{ alignItems: 'center', justifyContent: 'center',
      margin: 70, flexDirection: "row", flex: 1 }}>
      <Picker mode="dropdown" style={{ flex: 0.4 }} selectedValue
        ={this.state.role} onValueChange={role =>
        this.setState({ role: role })} />
      <Picker.Item label="Foreman" value="foreman" />
      <Picker.Item label="Project Manager" value="projectmanager" />
    </View>
    <TouchableOpacity style={{ backgroundColor: '#2699FB',
      margin: 15, height: 30, width: 120, justifyContent:
      'center', alignItems: 'center', borderRadius: 10 }} onPress={()
      => { this.onRegister() }}>
      <Text style={{ color: '#fff' }}><ยืนยัน</Text>
    </TouchableOpacity>
  </View>
</View >

```

รูปที่ 4.1.1.1 ภาษาจาวาสคริปต์ หน้าการลงทะเบียน



รูปที่ 4.1.1.2 หน้าจอการลงทะเบียน

ภายหลังจากการกดปุ่มยืนยัน ระบบจะทำฟังก์ชันที่เชื่อมต่อกับหลังบ้านไว้ โดยฟังก์ชันในหน้านี้อคือการลงทะเบียน ดังที่แสดงในรูป 4.1.1.3 โดยฟังก์ชันจะทำหน้าที่เก็บข้อมูล ชื่อผู้ใช้งาน รหัสผ่าน เบอร์โทรศัพท์ อีเมล และตำแหน่ง ลงในฐานข้อมูล

```

async onRegister() {
  let { username, password, email, phone_number, role } = this.state
  if (username == '' | password == '') {
    username = "null"
    password = "null"
    Alert.alert("Username or password must not empty.")
    return
  } else {
    username = username.toLowerCase()
    email = email.toLowerCase()
  }
  phone_number = "+66" + phone_number.toString().substring(1)
  Auth.signUp({
    username,
    password,
    attributes: {
      email: email, // optional
      phone_number: phone_number,
      "custom:role": role,
    }
  })
  .then(data => {
    console.log(data)
    Alert.alert("Register Success")
    this.props.navigation.goBack()
  })
  .catch(err => {
    Alert.alert(err.message)
    console.log(err)
  });
}
}

```

รูปที่ 4.1.1.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าการลงทะเบียน

4.1.2 การพัฒนาหน้าเข้าสู่ระบบ

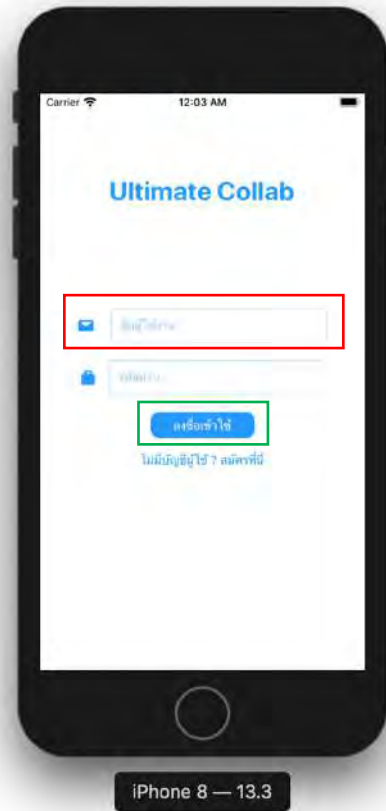
หน้าเข้าสู่ระบบ การทำส่วนต่อประสานกับผู้ใช้งานถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่ 4.1.2.1 และได้ผลลัพธ์ออกมาในรูป 4.1.2.2 ซึ่งในหน้าเข้าสู่ระบบ ผู้ใช้งานจะต้องกรอกชื่อผู้ใช้งาน และรหัสผ่านที่เคยลงทะเบียน จากนั้นกดปุ่มลงชื่อเข้าใช้

```

<View style={styles.container}>
  <View style={{ flex: 0.3, justifyContent: 'center',
    alignItems: 'center' }}>
    <Text style={{ color: '#2699FB', fontWeight: 'bold',
      fontSize: 30 }}>Ultimate Collab</Text>
  </View>
  <View style={styles.form}>
    <View style={{ flexDirection: 'row', alignItems:
      'center', justifyContent: 'center' }}>
      <Image style={{ width: 18, height: 13 }} source =
        {require('../assets/images/Mail.png')} />
      <TextInput placeholder="ชื่อใช้งาน" placeholderTextColor
        ="#9FCDF5" style={{ height: 40, width: 250, borderColor:
          '#BCE0FD', borderWidth: 1, paddingLeft: 10, color:
            '#2699FB', marginLeft: 20 }} onChangeText={text =>
              this.setState({ username: text })} autoCapitalize=
                'none' value={this.state.username}/>
    </View>
    <View style={{ flexDirection: 'row', alignItems: 'center',
      justifyContent: 'center' }}>
      <Image style={{ width: 18, height: 17 }} source =
        {require('../assets/images/Privacy.png')} />
      <TextInput placeholder="รหัสผ่าน" placeholderTextColor =
        "#9FCDF5" secureTextEntry={true} style={{ height: 40,
          width: 250, borderColor: '#BCE0FD', borderWidth: 1,
          paddingLeft: 10, marginVertical: 20, color: '#2699FB',
            marginLeft: 15 }} onChangeText={text =>
              this.setState({ password: text })} autoCapitalize='none'
                value={this.state.password}/>
    </View>
    <TouchableOpacity style={{ backgroundColor: '#2699FB',
      height: 30, width: 120, borderRadius: 10, justifyContent:
        'center', alignItems: 'center' }} onPress =
      {this.onLogin.bind(this)}>
      <Text style={{ color: 'white' }}>ลงชื่อเข้าใช้</Text>
    </TouchableOpacity>
    <TouchableOpacity style={{ marginTop: 15 }} onPress={() => {
      try { this.props.navigation.navigate("Register"); }
      catch (err) { console.log(err) } }}>
      <Text style={{ color: '#2699FB' }}>ไม่มีบัญชีผู้ใช้ ? สมัครที่นี่</Text>
    </TouchableOpacity>
  </View>
</View>

```

รูปที่ 4.1.2.1 ภาษาจาวาสคริปต์ หน้าเข้าสู่ระบบ



รูปที่ 4.1.2.2 หน้าจอเข้าสู่ระบบ

ภายหลังจากการกดปุ่มลงชื่อเข้าใช้ ระบบจะทำฟังก์ชันการเข้าสู่ระบบที่เชื่อมต่อกับหลังบ้านไว้ โดยการนำชื่อผู้ใช้และรหัสผ่านไปตรวจสอบกับฐานข้อมูล และฟังก์ชันจะคืนค่ามา โดยเราสามารถนำค่าไปตรวจสอบได้ว่าผู้ใช้งานได้เคยยืนยัน Verification Code แล้วหรือไม่ หากเป็นการลงชื่อเข้าใช้ครั้งแรกและไม่เคยยืนยัน Verification Code ระบบจะนำไปสู่หน้า Verification Code แต่หากเคยยืนยันแล้ว ระบบจะนำไปสู่หน้าจอการแจ้งเตือน ดังรูปที่ 4.1.2.3

```

async onLogin() {
  let { username, password } = this.state
  if (username == '' | password == '') {
    username = "null"
    password = "null"
    Alert.alert("Username or password must not empty.")
    return
  } else {
    username = username.toLowerCase()
  }
  Auth.signIn(username, password).then(user => {
    console.log('user successfully signed in!', user)
    this.state.user = this.user
    Alert.alert("Hello, " + user.username)
    this.props.navigation.navigate("Main", { navigationOptions: {headerShown: false,}})
    this.state.username = ""
    this.state.password = ""
  }).catch(err => {
    console.log({ err })
    Alert.alert(err.message)
    if (err.code == "UserNotConfirmedException") {
      this.props.navigation.navigate("Verification", {
        username: this.state.username
      })
    }
  })
}
}

```

รูปที่ 4.1.2.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าเข้าสู่ระบบ

4.1.3 หน้าแปลน

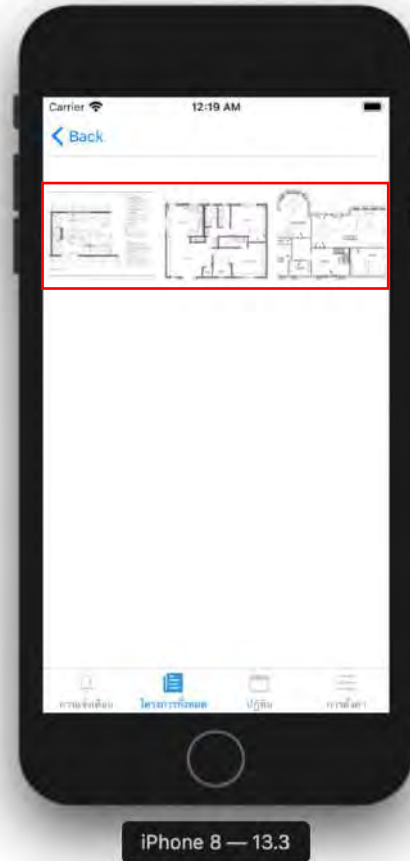
หน้าแปลน การทำส่วนต่อประสานผู้ใช้ถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังตัวอย่างรูปที่ 4.1.3.1 และได้ผลลัพธ์ในรูป 4.1.3.2 ซึ่งหน้าแปลนจะแสดงรูปของแปลนที่ถูกเก็บไว้ในฐานข้อมูล โดยสามารถกดที่รูปเพื่อขยายภาพ ให้ผู้ใช้งานสามารถเห็นได้ชัดเจนขึ้น

```

<View style={styles.MainContainer}>
  <ImageView images=
    {this.floorplansToURI(this.state.floorplans)}
    imageIndex={this.state.imageIndex} visible =
    {this.state.isImageView} onRequestClose={() =>
    { this.toggleImageView(0) }} />
  <FlatList data={this.state.floorplans} renderItem=
    ({ item, index }) => (
    <View style={styles.imageThumbnail}>
    <Item index={index} uri={item.src} toggleImageView
    ={this.toggleImageView}/>
    </View>)} numColumns={3} keyExtractor={({item, index})
    => index.toString()} />
</View>

```

รูปที่ 4.1.3.1 ภาษาจาวาสคริปต์ หน้าแปลน



รูปที่ 4.1.3.2 หน้าจอแปลน

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าแปลน คือฟังก์ชันที่แสดงรูปแปลนทั้งหมดที่เกี่ยวข้องกับโครงการนั้น ๆ ดังรูปที่ 4.1.3.3

```

return (
  <View style={styles.MainContainer}>
    <ImageView images={this.floorplansToURI(this.state.floorplans)} imageIndex={this.state.imageIndex}
    visible={this.state.isImageView}
    onRequestClose={() => { this.toggleImageView(0) }}
    />

    <FlatList data={this.state.floorplans} renderItem={({ item, index }) => (
      <View style={styles.imageThumbnail}>
        <Item index={index} uri = {item.src} toggleImageView={this.toggleImageView}/>
      </View>
    )}
    numColumns={3}
    keyExtractor={({item, index}) => index.toString()}
    />
  </View>
);
}
}

```

รูปที่ 4.1.3.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าแปลน

4.1.4 หน้าอีเมล

หน้าอีเมล การทำส่วนต่อประสานกับผู้ใช้ถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่ 4.1.4.1 และได้ผลลัพธ์ในรูป 4.1.4.2 ซึ่งหน้าอีเมลจะประกอบไปด้วยช่องใส่อีเมลของผู้ที่ต้องการส่งถึง และเนื้อหาของอีเมล จากนั้นกดปุ่มส่งอีเมล เพื่อทำการส่งอีเมล

```

<View style={{ flex: 1, flexDirection: 'column' }}>
  <View style={{ flex: 0.1, flexDirection: 'row',
    justifyContent: 'center', alignItems: 'center',
    marginTop: 10 }}>
    <Text style={{ fontWeight: 'bold', fontSize: 15,
      color: '#2699FB' }}>ถึง : </Text>
    <TextInput keyboardAppearance="default" onChangeText=
      {text => { this.state.to = text this.forceUpdate(); }}
      value={this.state.to} style={{ height: 40, width: 300,
        borderColor: '#2699FB', borderWidth: 1, color: '#2699FB'
      }} />
  </View>
  <View style={{ flex: 0.75, flexDirection: 'column',
    paddingLeft: 21 }}>
    <Text style={{ fontWeight: 'bold', fontSize: 15, color:
      '#2699FB', marginTop: 15 }}>หัวข้อ : </Text>
    <TextInput keyboardAppearance="default" onChangeText={text
      => { this.state.subject = text this.forceUpdate(); }}
      value={this.state.subject} style={{ height: 350, width:
        330, borderColor: '#2699FB', borderWidth: 1, color:
        '#2699FB' }} />
  </View>
  <View style={{ flex: 0.1, justifyContent: 'center',
    alignItems: 'center' }}>
    <TouchableOpacity style={{ backgroundColor: '#2699FB',
      height: 40, width: 130, borderRadius: 10, justifyContent:
        'center', alignItems: 'center' }} onPress=
      {this.handleEmail}>
      <Text style={{ color: '#fff' }}>ส่งอีเมลถึง ...</Text>
    </TouchableOpacity>
  </View>
</View>

```

รูปที่ 4.1.4.1 ภาษาจาวาสคริปต์ หน้าอีเมล



รูปที่ 4.1.4.2 หน้าจออีเมล

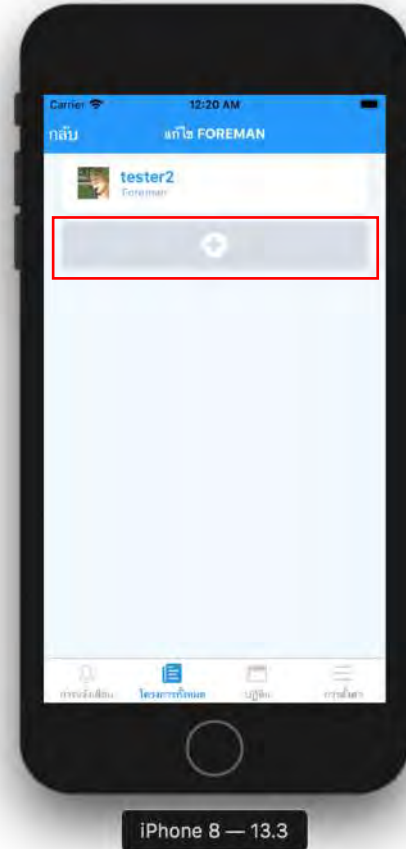
4.1.5 หน้าแก้ไข Foreman

หน้าแก้ไข Foreman การทำส่วนต่อประสานกับผู้ใช้ถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่ 4.1.5.1 และได้ผลลัพธ์ในรูป 4.1.5.2 ซึ่งหน้าแก้ไข Foreman จะแสดง Foreman ที่อยู่ในงานทั้งหมด ซึ่งในหน้านั้นผู้ใช้สามารถเพิ่มและลบ Foreman ที่เกี่ยวข้องกับโปรเจกต์ได้

```

<View style={{ flex: 1 }}>
  <View style={{ backgroundColor: 'white', flex: 1 }}>
    <SafeAreaView style={styles.container}>
      <ScrollView>
        <FlatList data={this.state.DATA} renderItem=
          <Item foreman={item} handleRefresh
            => {this.handleRefresh} taskData={this.state.taskData}
          /> keyExtractor={item => item} />
        <TouchableOpacity style={styles.itemfalse} onPress={() =>
          { this.setState({ isVisibleAdd:
            !this.state.isVisibleAdd }) }} >
          <Image style={{ width: 30, height: 30 }} source=
            {require('../assets/images/iconAdd.png')} />
        </TouchableOpacity>
      </ScrollView>
    </SafeAreaView>
    <Modal isVisible={this.state.isVisibleAdd}>
      <View style={{ flex: 0.90, backgroundColor: 'white',
        borderRadius: 20 }}>
        <SafeAreaView style={{ flex: 1 }}>
          <View style={{ flex: 0.35, padding: 30 }}>
            <MultiSelect hideTags items=
              {this.state.foremans_multiselect} uniqueKey="id"
              ref={component => { this.multiSelect = component; }}
              onSelectedItemsChange={this.onSelectedItemsChange}
              selectedItems={selectedItems} selectText="เลือก Foreman"
              searchInputPlaceholderText="ค้นหา Foreman" onChangeInput
              => {text => console.log(text)} itemTextColor="#000"
              displayKey="name" searchInputStyle={{ color: '#CCC' }}
              submitButtonColor="#48d22b" submitButtonText="OK" />
            </View>
          </SafeAreaView>
          <View style={{ flex: 0.55, margin: 5 }}>
            {this.multiSelect &&
              this.multiSelect.getSelectedItemsExt(selectedItems)}
          </View>
          <View style={{ flex: 0.1, flexDirection: 'row',
            justifyContent: 'space-around', marginVertical: 10 }}>
            <Button title="ยืนยัน" color="blue" onPress={() =>
              { this.handleSaveForeman() }} />
            <Button title="ยกเลิก" color="red" onPress={() =>
              { this.setState({ isVisibleAdd:
                !this.state.isVisibleAdd }) }} />
          </View>
        </View>
      </Modal>
    </View>
  </View>

```



รูปที่ 4.1.5.1 ภาษาจาวาสคริปต์ หน้าแก้ไข Foreman

รูปที่ 4.1.5.2 หน้าจอแก้ไข Foreman

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าแก้ไข Foreman คือฟังก์ชันที่แสดง Foreman ทั้งหมดที่เกี่ยวข้องกับงาน ฟังก์ชันเพิ่มและลบ Foreman ดังรูปที่ 4.1.5.3

```

componentDidMount() {
  this.animation.play()
  console.log(this.state)
  this.setState({ isLoading: true })
  API.get("taskAPI", "/tasks/foremans/" + this.state.taskData.id)
    .then(data => {
      this.setState({ DATA: data.foremans })
      this.setState({ isLoading: false })
    })
  this.setState({ isLoading: false })
}

handleSaveForeman = () => {
  let req = { body: { foremans: this.state.selectedItems } }
  API.put("taskAPI", "/tasks/addforemans/" + this.state.taskData.id, req).then(data => {this.handleRefresh()})
  this.setState({ isVisibleAdd: !this.state.isVisibleAdd })
}

onSelectedItemsChange = selectedItems => {
  this.setState({ selectedItems: selectedItems });
};

constructor(props) {
  super(props)
  this.state.taskData = this.props.navigation.state.params.taskData
  this.state.DATA = this.props.navigation.state.params.taskData.foremans
  if (this.state.taskData.foremans == null) { this.state.taskData.foremans = [] }
  API.get("siteAPI", "/sites/foremans/" + this.state.taskData.siteId).then(data => { let
  foremans_multiselect = data.foremans.filter(username => !this.state.taskData.foremans.includes(username))
  this.state.foremans_multiselect = foremans_multiselect.map((username, index) => {
    return { id: username, name: username }
  })
})
})
}

```

รูปที่ 4.1.5.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าแก้ไข Foreman

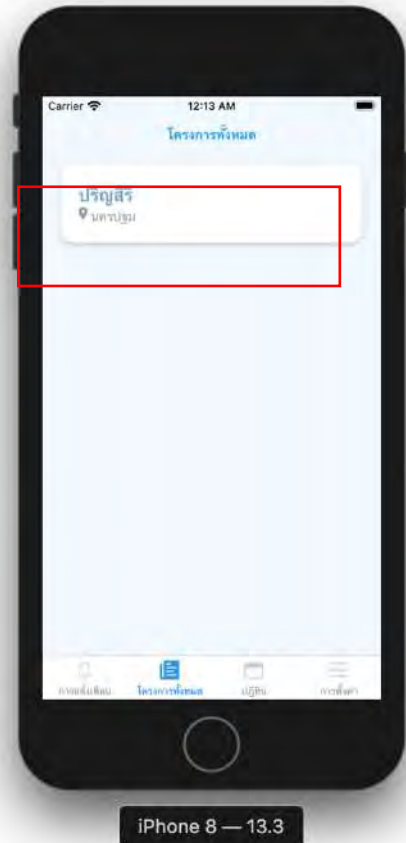
4.1.6 การพัฒนาหน้าโครงการทั้งหมด

หน้าโครงการทั้งหมด การทำส่วนต่อประสานกับผู้ใช้งานถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่ 4.1.6.1 และได้ผลลัพธ์ออกมาในรูป 4.1.6.2 ซึ่งในหน้าโครงการทั้งหมด จะแสดงโครงการทั้งหมดที่ผู้ใช้งานเกี่ยวข้อง ซึ่งข้อมูลของโครงการประกอบไปด้วยชื่อโครงการ และตำแหน่งที่ตั้งของโครงการ โดยผู้ใช้งานสามารถกดที่โครงการที่ต้องการ เพื่อจะนำไปสู่หน้าโครงการต่อไป

```
<TouchableOpacity onPress={() => {props.navigation.navigate("Site", { props: props, siteData: siteData })}}>
  <View style={styles.item}>
    <View style={{ flex: 0.92, backgroundColor: '#FFFFFF', padding: 20, borderRadius: 15 }}>
      <Text style={styles.title}>{title}</Text>
      <View style={{ flex: 0.3, flexDirection: 'row' }}>
        <Image style={{ width: 10, height: 14 }} source={require('../assets/images/location.png')} />
        <Text style={{ fontSize: 14, color: '#979797' }}>{location_string}</Text>
      </View>
    </View>
    <View style={{ flex: 0.08, backgroundColor: color, borderBottomRightRadius: 15, borderTopRightRadius: 15 }}>
    </View>
  </TouchableOpacity>

<View style={styles.backGround}>
  <View style={styles.backGround}>
    <SafeAreaView style={styles.container}>
      <ScrollView>
        <FlatList data={this.state.DATA} renderItem={({ item }) => <Item title={item.title} location_string = {item.location_string} props={this.props} color = {item.color} id={item.id} siteData={item} /> keyExtractor={item => item.id} />
      </ScrollView>
    </SafeAreaView>
  </View>
</View>
```

รูปที่ 4.1.6.1 ภาษาจาวาสคริปต์ หน้าโครงการทั้งหมด



รูปที่ 4.1.6.2 หน้าจอโครงการทั้งหมด

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าโครงการทั้งหมด คือฟังก์ชันที่จะแสดงข้อมูลโครงการทั้งหมดในฐานข้อมูล ดังรูปที่ 4.1.6.3

```
componentDidMount() {
  this.animation.play();
  Auth.currentUserInfo().then(data => { this.setState({ user: data })
  let username = this.state.user["username"]
  API.get("siteAPI", "/sites?username=" + username).then(data => {
    for (item in data) { this.state.DATA.push(data[item]) }
    this.setState({ isLoading: false })
  })
})
}
```

รูปที่ 4.1.6.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าโครงการทั้งหมด

4.1.7 หน้าโครงการ

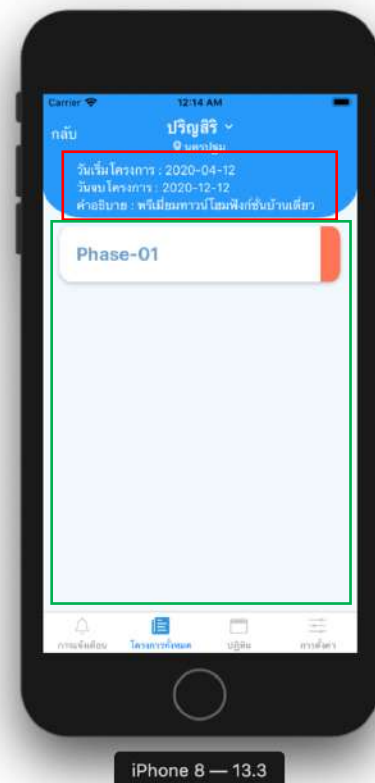
หน้าโครงการ การทำส่วนต่อประสานผู้ใช้งานถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังตัวอย่างรูปที่ 4.1.7.1 และได้ผลลัพธ์ในรูป 4.1.7.2 ซึ่งหน้าโครงการเป็นหน้าที่เชื่อมต่อการกดเลือกโครงการในหน้าโครงการทั้งหมด ในหน้าโครงการจะแสดงรายละเอียดของโครงการ วันที่เริ่ม วันที่สิ้นสุด และคำอธิบายโครงการ โดยแต่ละโครงการจะประกอบไปด้วยหลายโปรเจค โดยแต่ละโปรเจคผู้ใช้งานสามารถลบและแก้ไขได้ อีกทั้งในหน้านี้ผู้ที่มีตำแหน่งเป็น Project Manager เท่านั้นที่จะสามารถเพิ่มโปรเจคใหม่ได้

```

<View style={{ flex: 3, backgroundColor: '#F1F9FF' }}>
  <View style={{ height: 26, backgroundColor: '#2e99fb' }}></View>
  <Header props={this.props} siteData={this.state.siteData}
    setClickAddTask={() => this.toggleModal()} />
  <View style={styles.header}>
    <Text style={{ fontSize: 15, color: 'white', marginLeft:
      40 }}>วันที่โครงการ : {this.state.siteData.startDate}</Text>
    <Text style={{ fontSize: 15, color: 'white', marginLeft:
      40 }}>วันที่โครงการ : {this.state.siteData.dueDate}</Text>
    <Text style={{ fontSize: 15, color: 'white', marginLeft:
      40 }}>คำอธิบาย : {this.state.siteData.description}</Text>
  </View>
  <ScrollView>
    <Modal isVisible={this.state.isModalVisible}>
      <View style={{ flex: 0.37, backgroundColor: 'white',
        borderRadius: 20 }}>
        <View style={{ flexDirection: 'row', alignItems:
          'center', justifyContent: 'center', padding: 10 }}>
          <Text style={styles.title}>เพิ่มโปรเจกต์</Text>
        </View>
        <View style={{ flex: 0.8, justifyContent: 'space-between' }}>
          <View style={{ flexDirection: 'row', marginVertical:
            2, marginHorizontal: 20 }}>
            <Text style={{ fontSize: 17, color: '#9DBFD8' }}>
              ชื่อโปรเจกต์ : </Text>
            <TextInput style={{ height: 17, width: 155, fontSize:
              15, borderBottomWidth: 1, borderBottomColor: '#6FA8CB' }}
              keyboardType="default" onChangeText={text =>
                { this.state.form.title = text this.forceUpdate() }}
              value={this.state.form.title} />
          </View>
          <View style={{ flexDirection: 'row', marginHorizontal: 20 }}>
            <Text style={{ fontSize: 17, color: '#9DBFD8' }}>
              รายละเอียด : </Text>
            <TextInput style={{ height: 17, width: 177, fontSize: 15,
              borderBottomWidth: 1, borderBottomColor: '#6FA8CB' }}
              keyboardType="default" onChangeText={text =>
                { this.state.form.description = text this.forceUpdate() }}
              value={this.state.form.description} />
          </View>
          <View style={{ flexDirection: 'row', marginHorizontal: 20,
            alignItems: 'center', justifyContent: 'flex-start' }}>
            <Text style={{ fontSize: 17, color: '#9DBFD8' }}>
              วันที่เริ่มต้น : </Text>
            <DatePicker style={{ width: 100 }} date=
              {this.state.form.startDate} mode="date" placeholder=
              "select date" format="YYYY-MM-DD" confirmBtnText="Confirm"
              cancelBtnText="Cancel" showIcon={false} duration=(350)
              customStyles={{ dateIcon: { position: 'absolute', left: 0,
                top: 4, marginLeft: 0, height: 7, width: 3 }, dateInput:
                { marginLeft: 3, backgroundColor: 'white', borderColor:
                  'white', color: '#fff', fontSize: 14 }, fontSize: 15 }}
              onChange={date => { this.state.form.startDate = date
                this.forceUpdate() }} />
          </View>
          <View style={{ flexDirection: 'row', marginHorizontal: 20,
            alignItems: 'center', justifyContent: 'flex-start' }}>
            <Text style={{ fontSize: 17, color: '#9DBFD8' }}>
              วันที่สิ้นสุด : </Text>
            <DatePicker style={{ width: 100 }} date =
              {this.state.form.dueDate} mode="date" placeholder=
              "select date" format="YYYY-MM-DD" confirmBtnText="Confirm"
              cancelBtnText="Cancel" showIcon={false} duration=(350)
              customStyles={{ dateIcon: { position: 'absolute', left: 0,
                top: 4, marginLeft: 0, height: 7, width: 3 }, dateInput:
                { marginLeft: 3, backgroundColor: 'white', borderColor:
                  'white', color: '#fff', fontSize: 14 }, fontSize: 15 }}
              onChange={date => { this.state.form.dueDate = date
                this.forceUpdate() }} />
          </View>
          <View style={{ flex: 0.3, flexDirection: 'row', justifyContent:
            'space-between', marginVertical: 10 }}>
            <TouchableOpacity onPress={() => this.clearState()}>
              <Text style={{ fontSize: 18, color: 'red', marginEnd: 10 }}>
                ยกเลิก</Text>
            </TouchableOpacity>
            <TouchableOpacity onPress={() => {this.handleClickProject()}}>
              <Text style={{ fontSize: 18, color: 'blue', marginEnd: 10 }}>
                เพิ่ม</Text>
            </TouchableOpacity>
          </View>
        </View>
      </Modal>
      <View style={styles.background}>
        <SafeAreaView style={styles.container}>
          <FlatList data={this.state.DATA} renderItem={({ item }) =>
            <Item title={item.title} props={this.props} profilepic=
              {item.profilepic} color={item.color} projectData={item}
              handleOnPress={this.handleClickProject} projectId={item.projectId}
              users={this.state.users} finish={item.finish} />
            <KeyExtractor={item => item.id} />
          </SafeAreaView>
        </View>
      </ScrollView>
    </View>
  </View>

```

รูปที่ 4.1.7.1 ตัวอย่างภาษาจาวาสคริปต์ หน้าโครงการ



iPhone 8 — 13.3

รูปที่ 4.1.7.2 หน้าจอหน้าโครงการ

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าโครงการ คือฟังก์ชันที่จะแสดงข้อมูลโครงการ และโปรเจกต์ที่อยู่ภายใต้โครงการนี้ และฟังก์ชันเพิ่มโปรเจกต์และฟังก์ชันลบโปรเจกต์ ดังรูปที่ 4.1.7.3

```

const handleUpdateProject = () => {
  let req = {
    body: {
      startDate: projectData.startDate,
      dueDate: projectData.dueDate,
      description: projectData.description,
      title: projectData.title,
    }
  }
  API.put("projectAPI", "/projects/" + projectData.id, req).then(data => { handleRefresh() })
};

handleLoad() {
  this.state.isLoading = true
  this.state.DATA = []
  API.get("projectAPI", "/projects/bysiteid/" + this.state.siteData.id)
  .then(data => {
    for (item in data) { this.state.DATA.push(data[item]) }
    this.setState({ isLoading: false })
  })
}

handleCreateProject = () => {
  this.state.form.foremans = []
  let mention_json = EditorUtils.findMentions(this.state.message.text)
  for (var key in mention_json) {
    if (mention_json[key].username != " " && mention_json[key].username != "" && mention_json[key].username != null) { this.state.form.foremans.push(mention_json[key].username) }
  }
  let req = {
    body: {
      projectmanager: this.state.user["username"], foremans: this.state.form.foremans,
      startDate: this.state.form.startDate, dueDate: this.state.form.dueDate,
      description: this.state.form.description, siteId: this.state.siteData.id,
      images: this.state.form.images, title: this.state.form.title,
      blueprints: this.state.form.blueprints,
    }
  }
  API.post("projectAPI", "/projects", req).then(data => { this.handleRefresh() })
}
}

```

รูปที่ 4.1.7.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าโครงการ

4.1.8 หน้าโปรเจค

หน้าโปรเจค การทำส่วนต่อประสานกับผู้ใช้ถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังตัวอย่างรูปที่ 4.1.8.1 และได้ผลลัพธ์ในรูป 4.1.8.2 ซึ่งหน้าโปรเจคเป็นหน้าที่เชื่อมต่อมาจากการกดเลือกโปรเจคในหน้าโครงการ ในหน้าโปรเจคจะแสดงรายละเอียดของโปรเจค วันที่เริ่ม วันที่สิ้นสุด และคำอธิบายโปรเจค โดยแต่ละโปรเจคจะประกอบไปด้วยหลายงาน โดยแต่ละงานผู้ใช้สามารถแก้ไขและกำหนดสถานะของงานให้เสร็จได้ อีกทั้งในหน้านี้ผู้ที่มีตำแหน่งเป็น Project Manager เท่านั้นที่จะสามารถเพิ่มงานใหม่ได้

```

<view style={{ flex: 5, backgroundColor: '#F3F9FF' }}>
<view style={{ height: 25, backgroundColor: '#2699FB'
}}></view>
<Header props={this.props} projectIdData =
(this.state.projectData) setClickAddTask={() =>
this.toggleModal()} />
<TouchableOpacity style={{ position: 'absolute', left: 10,
top: 25 }} onPress={() => this.props.navigation.goBack()}>
<text style={{ color: 'white', fontSize: 17 }}>กลับ</text>
</TouchableOpacity>
<view style={styles.header}>
<text style={{ fontSize: 15, color: 'white', marginLeft:
40 }}>วันที่เริ่มงาน: {this.state.projectData.startDate}</text>
<text style={{ fontSize: 15, color: 'white', marginLeft:
40 }}>วันที่ถึงกำหนด : {this.state.projectData.dueDate}</text>
<text style={{ fontSize: 15, color: 'white', marginLeft:
40 }}>คำอธิบาย : {this.state.projectData.description}</text>
<text style={{ fontSize: 15, color: 'white', marginLeft: 40
}}>Project Manager : {this.state.projectData.projectManager}
</text>
</view>
<ScrollView>
<Modal isVisible={this.state.isModalVisible}>
<view style={{ flex: 0.65, backgroundColor: 'white',
borderRadius: 20 }}>
<view style={{ flexDirection: 'row', alignContent: 'center',
justifyContent: 'center', padding: 10 }}>
<text style={styles.title}>เพิ่มงาน</text>
</view>
<view style={{ flexDirection: 'row', alignContent: 'center',
justifyContent: 'center' }}>
{!this.state.form.image ? <TouchableOpacity onPress={()
=> ( this.pickImage() )}> <Image style={{ width: 70,
height: 70 }} source={require('../assets/images/addImage.png')}
/}> </TouchableOpacity> : <TouchableOpacity onPress={() =>
{ this.pickImage() }}> <Image source =
{ { url: this.state.form.image } } style={{ width: 100, height:
100 }} /> </TouchableOpacity>
</view>
<view style={{ flexDirection: 'row', marginVertical: 2,
marginHorizontal: 20 }}>
<text style={{ fontSize: 17, color: '#9DBFD8' }}>ชื่องาน : </text>
<TextInput style={{ height: 17, width: 150, fontSize: 15,
borderBottomWidth: 1, borderBottomColor: '#6A8CB' }}
keyboardAppearance='default' onChange={text =>
{ this.state.form.title = text this.forceUpdate() }} value=
{this.state.form.title} />
</view>
<view style={{ flexDirection: 'row', marginHorizontal: 20 }}>
<text style={{ fontSize: 17, color: '#9DBFD8' }}>เพิ่มรายละเอียด : </text>
<TextInput style={{ height: 17, width: 150, fontSize: 15,
borderBottomWidth: 1, borderBottomColor: '#6A8CB' }}
keyboardAppearance='default' onChange={text =>
{ this.state.form.description = text this.forceUpdate() }}
value={this.state.form.description} />
</view>
<view style={{ flexDirection: 'row', marginHorizontal: 20,
alignItems: 'center', justifyContent: 'flex-start' }}>
<text style={{ fontSize: 17, color: '#9DBFD8' }}>วันที่เริ่ม:</text>
<DatePicker style={{ width: 100 }} date=
{this.state.form.startDate} mode='date' placeholder=
"select date" format='YYYY-MM-DD' confirmBtnText='Confirm'
cancelBtnText='Cancel' showIcon={false} duration=(350)
customStyles={{ dateIcon: { position: 'absolute', left: 0,
top: 4, marginLeft: 0, height: 7, width: 3 }, dateInput:
{ marginLeft: 3, backgroundColor: 'white', borderColor:
'white', color: '#ffff', fontSize: 14 }, fontSize: 15 }}
onDateChange={(date) => { this.state.form.startDate = date
this.forceUpdate() }} />
</view>
<view style={{ flexDirection: 'row', marginHorizontal: 20,
alignItems: 'center', justifyContent: 'flex-start' }}>
<text style={{ fontSize: 17, color: '#9DBFD8' }}>วันที่ถึง :
</text>
<DatePicker style={{ width: 100 }} date =
{this.state.form.dueDate} mode='date' placeholder=
"select date" format='YYYY-MM-DD' confirmBtnText='Confirm'
cancelBtnText='Cancel' showIcon={false} duration=(350)
customStyles={{ dateIcon: { position: 'absolute', left: 0,
top: 4, marginLeft: 0, height: 7, width: 3 }, dateInput:
{ marginLeft: 3, backgroundColor: 'white', borderColor:
'white', color: '#ffff', fontSize: 14 }, fontSize: 15 }}
onDateChange={(date) => { this.state.form.dueDate = date
this.forceUpdate() }} />
</view>
<view style={{ flexDirection: 'row', justifyContent: 'center',
alignContent: 'center' }}>
<view>
<Editor list={this.state.users} initialValue =
{this.state.initialValue} clearInput={this.state.clearInput}
onChange={this.onChangeHandler} showEditor={this.state.showEditor}
toggleEditor={this.toggleEditor} showMentions =
{this.state.showMentions} onHideMentions={this.onHideMentions}
placeholder="say Foreman..."/>
</view>
</view>
<view style={{ flex: 0.4, flexDirection: 'row', justifyContent:
'space-between', padding: 10 }}>
<TouchableOpacity onPress={() => this.toggleModal()} >
<text style={{ fontSize: 15, color: 'red', marginEnd: 10 }}>
ยกเลิก</text>
</TouchableOpacity>
<TouchableOpacity onPress={() => { this.handleClick() }}>
<text style={{ fontSize: 15, color: 'blue', marginEnd: 10 }}>
เพิ่มงาน</text>
</TouchableOpacity>
</view>
</view>
</Modal>
<view style={styles.backgroundColor}>
<SafeAreaView style={styles.container}>
<FlatList data={this.state.DATA} renderItem={({ item } => <Item
title={item.title} props={this.props} taskData={item} handleRefresh
={this.handleClickRefresh} finish={item.finish} /> keyExtractor=
{item => item.id} />
</SafeAreaView>
</view>
</ScrollView>
</view>

```



รูปที่ 4.1.8.2 หน้าจอโปรเจกต์

รูปที่ 4.1.8.1 ภาษาจาวาสคริปต์ หน้าโปรเจกต์

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าโปรเจค คือฟังก์ชันที่จะแสดงข้อมูลโปรเจค และงานที่อยู่ภายใต้โปรเจคนี้ และฟังก์ชันเพิ่มงาน ฟังก์ชันแก้ไขงาน ฟังก์ชันลบงาน และฟังก์ชันกำหนดสถานะให้กับงาน ดังรูปที่ 4.1.8.3

```

handleLoad() {
  this.state.isLoading = true
  this.state.DATA = []
  API.get("taskAPI", "/tasks?username=" + this.state.user["username"] + "&id=" + this.state.projectData.id)
    .then(async data => {
      for (item in data) {
        data[item].image = await Storage.get(data[item].image)
        this.state.DATA.push(data[item])
      }
      this.setState({isLoading: false})
    })
}

async componentDidMount() {
  this.animation.play()
  Auth.currentUserInfo()
    .then(data => {
      this.setState({ user: data })
      let username = this.state.user["username"]
      API.get("taskAPI", "/tasks?username=" + username + "&id=" + this.state.projectData.id)
        .then(async data => {
          for (item in data) {
            data[item].image = await Storage.get(data[item].image)
            this.state.DATA.push(data[item])
          }
          API.get("siteAPI", "/sites/foremans/" + this.state.projectData.siteId).then(data => {
            let foremans_mention_tag = []
            let foremans = data.foremans
            for (index in foremans) {
              foremans_mention_tag.push({
                id: index + 1,
                username: foremans[index]
              })
            }
            this.state.users = foremans_mention_tag this.setState({ isLoading: false })
          })
        })
    })
}
}

```

รูปที่ 4.1.8.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าโปรเจค

4.1.9 หน้างาน

หน้างาน การทำส่วนต่อประสานผู้ใช้ถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังตัวอย่างรูปที่ 4.1.9.1 และได้ผลลัพธ์ในรูป 4.1.9.2 ซึ่งหน้างานเป็นหน้าที่เชื่อมต่อมาจากการกดเลือกงานในหน้าโปรเจค ในหน้างานจะแสดงชื่องาน วันที่เริ่ม วันที่สิ้นสุด และคำอธิบาย ซึ่งในหน้างานจะประกอบไปด้วย ปัญหาย่อย ๆ หลายปัญหา ผู้ใช้งานสามารถเพิ่มปัญหาพร้อมแนบรูปประกอบ และลบปัญหาได้


```

<View style={{ flex: 3, backgroundColor: '#F1F9FF' }}>
  <View style={{ height: 25, backgroundColor: '#2699FB'
  }}></View>
  <Header props={this.props} taskData={this.state.taskData}
  handleRefresh={this.handleRefresh} />
  <TouchableOpacity style={{ position: 'absolute', top: 25,
  left: 10 }} onPress={() => this.props.navigation.goBack()}>
    <Text style={{ color: 'white', fontSize: 17 }}>กลับ</Text>
  </TouchableOpacity>
  <ScrollView>
    <View style={{ flex: 2 }}>
      <View style={{ flex: 0.08 }}>
        <View style={styles.item}>
          <Text style={styles.title}>งาน :
          </Text>
          <Text style={styles.title}>วันที่เริ่ม :
          </Text>
          <Text style={styles.title}>วันที่สิ้นสุด :
          </Text>
          <Text style={styles.title}>วันที่ครบกำหนด :
          </Text>
        </View>
      </View>
      <View style={{ flex: 0.6 }}>
        <View style={styles.item}>
          <View style={{ flex: 1 }}>
            <View style={{ flex: 2.7 }}>
              <TextInput style={{ height: 30, backgroundColor:
              '#ebebfb', fontSize: 14, width: 300, borderRadius:
              7, padding: 5, marginBottom: 10 }} placeholder=
              "ระบุชื่อ" onChangeText={text => {
              this.state.form.title = text this.forceUpdate()}}/
            </View>
            <View style={{ flex: 8 }}>
              <TextInput style={{ height: 30, backgroundColor:
              '#ebebfb', fontSize: 14, width: 300, borderRadius:
              7, padding: 5 }} placeholder="คำอธิบาย" onChangeText={text
              => { this.state.form.description = text
              this.forceUpdate()}} />
            </View>
          </View>
          <View style={{ flexDirection: 'row', justifyContent:
          'flex-end'}}>
            <TouchableOpacity onPress={this.addSubtask}>
              <View style={{ width: 40, height: 30, backgroundColor:
              '#2699FB', alignItems: 'center', alignContent: 'center',
              justifyContent: 'center', borderRadius: 5 }}>
                <Text style={{ color: 'white' }}>บันทึก</Text>
              </View>
            </TouchableOpacity>
          </View>
        </View>
      </View>
      <View style={{ backgroundColor: '#F1F9FF', flex: 1 }}>
        <SafeAreaView>
          <FlatList data={this.state.DATA} renderItem={({ item }} =>
            <ItemSubTask name={item.name} handleRefresh={this.handleRefresh}
            props={this.props} description={item.description} subtaskData=
            {item} /> keyExtractor={item => item.id} />
          </SafeAreaView>
        </View>
      </View>
    </ScrollView>
  </View >

```

รูปที่ 4.1.9.1 ภาษาจาวาสคริปต์ หน้างาน



รูปที่ 4.1.9.2 หน้าจองาน

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้างาน คือฟังก์ชันที่จะแสดงข้อมูลงาน และปัญหาที่อยู่ภายใต้งานนี้ ฟังก์ชันเพิ่มปัญหาและฟังก์ชันลบปัญหา ดังรูปที่ 4.1.9.3

```

handleLoad() {
  this.state.isLoading = true
  this.state.DATA = []
  let username = this.state.user["username"]
  API.get("subtaskAPI", "/subtasks/" + this.state.taskData.id)
  .then(async data => { let new_data = []
    var bar = await data.map(async item => { let temp_img = []
      await item.images.map(async (image) => { var result = await Storage.get(image)
        .then(url => { temp_img.push(url) return new Promise((res, rej) => { res(result) }) })
      })
      item.images = temp_img
      new_data.push(item)
    })
    Promise.all(bar).then(() => {
      this.setState({ DATA: new_data })
      this.setState({ isLoading: false })
    })
  })
}

componentDidMount() {
  this.animation.play()
  Auth.currentUserInfo()
  .then(data => {
    this.setState({ user: data })
    API.get("subtaskAPI", "/subtasks/" + this.state.taskData.id)
    .then(async data => { let new_data = []
      var bar = await data.map(async item => { let temp_img = []
        await item.images.map(async (image) => { var result = await Storage.get(image)
          .then(url => { temp_img.push(url) return new Promise((res, rej) => { res(result) }) })
        })
        item.images = temp_img
        new_data.push(item)
      })
      Promise.all(bar).then(() => {
        this.setState({ DATA: new_data })
        this.setState({ isLoading: false })
      })
    })
  })
}
}

```

รูปที่ 4.1.9.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้างาน

4.1.10 หน้าปัญหา

หน้าปัญหา การทำส่วนต่อประสานผู้ใช้ถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังตัวอย่างรูปที่ 4.1.10.1 และได้ผลลัพธ์ในรูป 4.1.10.2 ซึ่งหน้าปัญหาเป็นหน้าที่เชื่อมต่อมาจากการกดเลือกปัญหาในหน้างาน หน้าปัญหาจะแสดงชื่อผู้แจ้งปัญหา ชื่อปัญหา และรายละเอียดของปัญหา ซึ่งในการแจ้งปัญหา สามารถแสดงความคิดเห็นเพื่อรายงานความคืบหน้าของปัญหาได้

```

<View style={styles.backGround}>
  <ImageView images={this.state.imagesSource} imageIndex=
    {0} visible={this.state.isImageView} onRequestClose={()
    => this.toggleImageView()} />
  <View style={styles.item}>
    <Text style={{ fontSize: 18, fontWeight: 'bold' }}>
      {this.state.subtaskData.submit_by}</Text>
    <Text style={{ fontSize: 18 }}> ปัญหา :
    <Text style={{ fontSize: 18 }}> {this.state.subtaskData.title}</Text>
    <Text style={{ fontSize: 15 }}>
      {this.state.subtaskData.description}</Text>
    <TouchableOpacity onPress={() => { this.toggleImageView()
    }}>
      <View style={{ flexDirection: 'row', justifyContent:
      'space-around', marginVertical: 7 }}>
        <Image style={{ width: 90, height: 90 }} source={{
        uri: this.state.subtaskData.images[0] }} />
        <Image style={{ width: 90, height: 90 }} source={{
        uri: this.state.subtaskData.images[1] }} />
        <Image style={{ width: 90, height: 90 }} source={{
        uri: this.state.subtaskData.images[2] }} />
      </View>
    </TouchableOpacity>
  </View>
  <ScrollView>
    <View style={{ backgroundColor: '#F1F9FF', flex: 1 }}>
      <SafeAreaView>
        <FlatList data={this.state.comments} renderItem={({
        item }) => <ItemComment item={item} />} keyExtractor=
        {item => item.id} />
      </SafeAreaView>
      <View style={styles.item}>
        <View style={{ flexDirection: 'row' }}>
          <TextInput style={{ flex: 1, backgroundColor:
          '#ebebeb', fontSize: 14, borderRadius: 7, padding: 5 }}
          placeholder="แสดงความคิดเห็น..." clearButtonMode="always"
          value = {this.state.form.comment} onChangeText={text =>
          this.setState({ form: { comment: text } })} />
          <TouchableOpacity onPress={() => this.handleComment()}>
            <View style={{ width: 40, height: 30, backgroundColor:
            '#2699FB', alignItems: 'center', alignContent:
            'center', justifyContent: 'center', borderRadius: 5,
            marginHorizontal: 2 }}>
              <Text style={{ color: 'white' }}>ยืนยัน</Text>
            </View>
          </TouchableOpacity>
        </View>
      </View>
    </ScrollView>
  </View>
</View>

```

รูปที่ 4.1.10.1 ภาษาจาวาสคริปต์ หน้าปัญหา



รูปที่ 4.1.10.2 หน้าปัญหา

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าปัญหา คือฟังก์ชันที่จะแสดงข้อมูลปัญหา ฟังก์ชันแสดงความคิดเห็นที่อยู่ภายในตัวงานนี้ และฟังก์ชันเพิ่มความคิดเห็น ดังรูปที่ 4.1.10.3

```

handleRefreshComment = () => {
  let req = { body: { subtask: this.state.subtaskData } }
  API.post("subtaskAPI", "/subtasks/comment", req).then(data => {
    this.setState({ comments: data.comments })
  })
}

componentDidMount() {
  Auth.currentUserInfo()
    .then(data => {
      this.state.subtaskData.images.forEach(item => { this.state.imagesSource.push({ uri: item }) })
      this.setState({ user: data })
      this._notificationSubscription = Notifications.addListener( this._handleNotification );
    })
}

handleComment = () => {
  let req = {
    body: { comment: this.state.form.comment, submit_by: this.state.user["username"], subtask:
this.state.subtaskData, images: []
  }
  }

  this.setState({ form: { comment: "" } })

  API.put("subtaskAPI", "/subtasks/comment", req).then(data => {
    this.setState({ comments: data.comments })
  })
}

```

รูปที่ 4.1.10.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าปัญหา

4.1.11 การพัฒนาหน้าแสดงการแจ้งเตือน

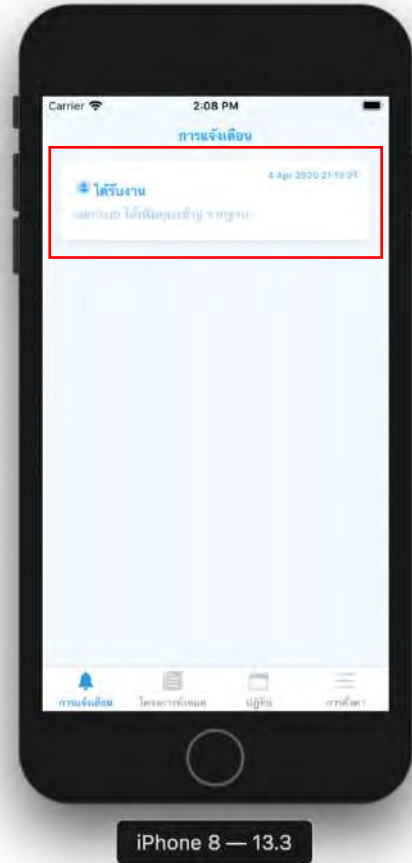
หน้าการแจ้งเตือน การทำส่วนต่อประสานกับผู้ใช้งานถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่ 4.1.11.1 และได้ผลลัพธ์ออกมาในรูป 4.1.11.2 ผู้ใช้งานสามารถเห็นการแจ้งเตือนทั้งหมดเมื่อมีการสร้างงาน หรือมอบหมายงาน โดยผู้ใช้งานสามารถกดที่การแจ้งเตือนนั้น ๆ เพื่อที่ระบบจะนำไปสู่หน้างานที่เกี่ยวข้องกับการแจ้งเตือนนั้น

```

<ScrollView>
  <TouchableOpacity style={styles.item} onPress={() => {
    handleTaskNavigation()}}>
    <View style={{ flex: 0.5, flexDirection: 'row-reverse' }}>
      <Text style={{ fontSize: 10, color: '#2699FB' }}>
        {timeShow}</Text>
      </View>
      <View style={{ flex: 0.5, flexDirection: 'row',
        marginBottom: 5 }}>
        <Image style={{ width: 15, height: 15, marginLeft: 7 }}
          source={require('../assets/images/Profile.png')}/>
        <Text style={styles.title}>
          {HEADER_MAPPING[notification.type]}</Text>
        </View>
        <View style={{ flex: 0.4, marginBottom: 10 }}>
          <Text style={{ color: '#9FCDF5' }}>
            {notification.message}</Text>
          </View>
        </TouchableOpacity>
      </ScrollView>

```

รูปที่ 4.1.11.1 ภาษาจาวาสคริปต์ หน้าการแจ้งเตือน



รูปที่ 4.1.11.2 หน้าจอการแจ้งเตือน

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าการแจ้งเตือน คือฟังก์ชันที่จะแสดงการแจ้งเตือนทั้งหมด ของผู้ใช้งานนั้น ๆ ดังรูปที่ 4.1.11.3

```

componentDidMount = async () => {
  const { status: existingStatus } = await Permissions.getAsync(Permissions.NOTIFICATIONS);
  this.setState({ user: await Auth.currentUserInfo() })

  let finalStatus = existingStatus;
  if (existingStatus !== "granted") {
    const { status } = await Permissions.askAsync( Permissions.NOTIFICATIONS );
    finalStatus = status;
  }

  Notifications.getExpoPushTokenAsync().then(token => {
    this._notificationSubscription = Notifications.addListener( this._handleNotification );
    let req = { body: { username: this.state.user.username, token: token } }
    API.put("userAPI", "/users/pushtoken", req).then(res => {
      API.get("notificationAPI", "/notifications/new?username=" + this.state.user.username)
        .then(data => {
          this.state.DATA = data
          this.setState({ isLoading: false })
        })
    })
  })
}

handleRefresh = () => {
  this.setState({ DATA: [], isLoading: true })
  API.get("notificationAPI", "/notifications/new?username=" + this.state.user.username)
    .then(data => {
      this.state.DATA = data
      this.setState({ isLoading: false })
    })
}

```

รูปที่ 4.1.11.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าการแจ้งเตือน

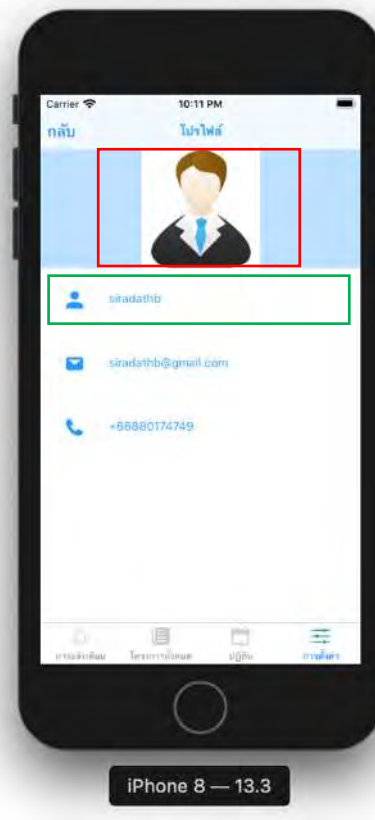
4.1.12 การพัฒนาหน้าโปรไฟล์

หน้าโปรไฟล์ การทำส่วนต่อประสานผู้ใช้งานถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่

4.1.12.1 และได้ผลลัพธ์ออกมาในรูป 4.1.12.2 ซึ่งในหน้าโปรไฟล์ ผู้ใช้งานจะสามารถเห็นข้อมูลของผู้ใช้งาน ได้แก่ ชื่อผู้ใช้งาน อีเมล และเบอร์โทรศัพท์

```
<View style={styles.container}>
  <View style={{ flex: 1, flexDirection: 'column' }}>
    <TouchableOpacity style={{ flex: 0.25, alignItems:
      'center', justifyContent: 'space-around',
      backgroundColor: '#BCE6FD' }} onPress={() =>
      { this.pickImage() }}>
      <Image style={this.state.user.image ? { width:
        '100%', height: '100%', aspectRatio: 1, } : {
        width: '20%', height: '20%', aspectRatio: 1,}}
        source={this.state.user.image ? { uri:
        this.state.user.image } :
        require('../assets/images/Add1.png')} />
    </TouchableOpacity>
    <View style={{ flex: 0.05 }} />
    <View style={{ flex: 0.3, alignItems: 'stretch',
      justifyContent: 'space-between', backgroundColor:
      '#fff' }}>
      <View style={{ flex: 0.10, flexDirection: 'row',
        alignItems: 'flex-start', marginLeft: 30 }}>
        <Image style={{ width: 20, height: 20 }} source =
        {require('../assets/images/Editprofile_profile.png')} />
        <Text style={{ flex: 1, alignItems: 'center',
          marginLeft: 30, color: '#2699FB' }}>
          {this.state.user['username']}</Text>
      </View>
      <View style={{ flex: 0.10, flexDirection: 'row',
        alignItems: 'flex-start', marginLeft: 30 }}>
        <Image style={{ width: 20, height: 20 }} source=
        {require('../assets/images/Editprofile_email.png')} />
        <Text style={{ flex: 1, alignItems: 'center',
          marginLeft: 30, color: '#2699FB' }}>
          {this.state.user.email}</Text>
      </View>
      <View style={{ flex: 0.10, flexDirection: 'row',
        alignItems: 'flex-start', marginLeft: 30 }}>
        <Image style={{ width: 20, height: 20 }} source=
        {require('../assets/images/Editprofile_Phone.png')} />
        <Text style={{ flex: 1, alignItems: 'center',
          marginLeft: 30, color: '#2699FB' }}>
          {this.state.user.phone_number}
        </Text>
      </View>
    </View>
    <View style={{ flex: 0.05 }} />
  </View>
</View>
```

รูปที่ 4.1.12.1 ภาษาจาวาสคริปต์ หน้าโปรไฟล์



รูปที่ 4.1.12.2 หน้าจอโปรไฟล์

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าโปรไฟล์ คือฟังก์ชันสำหรับแสดงข้อมูลของผู้ใช้งาน ดังรูปที่

4.1.12.3

```

async componentDidMount() {
  this.animation.play()
  Auth.currentUserInfo()
    .then(data => {
      API.get("userAPI", "/users/" + data.username)
        .then(query_user => {
          this.setState({ user: query_user })
          Storage.get(query_user.image)
            .then(url => {
              this.state.user.image = url
              this.setState({ isLoading: false })
            })
        })
    })
}

handleRefreshProfilePic = () => {
  this.setState({ isLoading: true })
  Auth.currentUserInfo().then(data => {
    API.get("userAPI", "/users/" + data.username)
      .then(query_user => {
        this.setState({ user: query_user })

        Storage.get(query_user.image)
          .then(url => {
            this.state.user.image = url
            this.setState({ isLoading: false })
          })
      })
  })
}
}

```

รูปที่ 4.1.12.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าโปรไฟล์

4.1.13 การพัฒนาหน้าการตั้งค่า

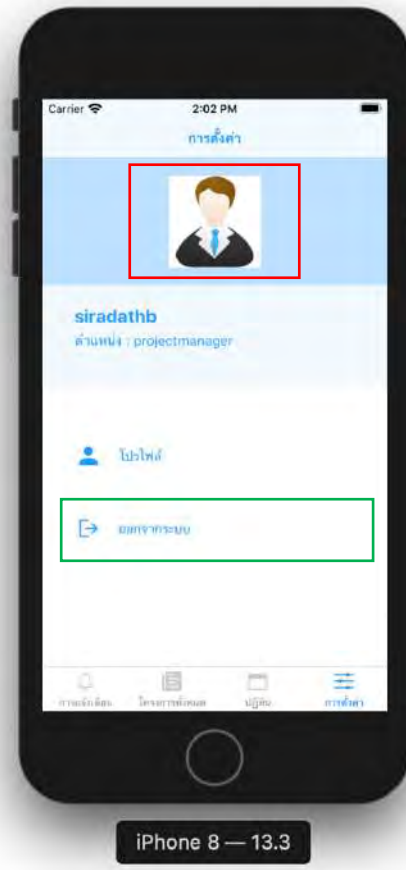
หน้าการตั้งค่า การทำส่วนต่อประสานผู้ใช้งานถูกพัฒนาด้วยภาษาจาวาสคริปต์ ดังรูปที่ 4.1.13.1 และได้ผลลัพธ์ออกมาในรูป 4.1.13.2 ซึ่งในหน้าการตั้งค่า ผู้ใช้งานจะเห็นรูปภาพโปรไฟล์ ชื่อผู้ใช้งาน ตำแหน่ง ปุ่มโปรไฟล์ และปุ่มลงชื่อออก

```

<View style={{ flex: 1, flexDirection: 'column' }}>
  <View style={{ flex: 0.25, alignItems: 'center',
    justifyContent: 'space-around', backgroundColor:
    '#BCE0FD' }}> {this.state.user.image ? <Image style=
    {{ width: 100, height: 100 }} source =
    {{ uri: this.state.user.image }} /> : <Image style=
    {{ width: 100, height: 100 }} source =
    {require('../assets/images/People.png')} /> } />
  </View>
  <View style={{ flex: 0.20, flexDirection: 'row',
    backgroundColor: '#F1F9FF', justifyContent:
    'center', alignItems: 'center' }}>
    <Text style={{ color: '#2699FB', fontSize: 20,
    fontWeight: 'bold', position: 'absolute', left:
    35, bottom: 65 }}>{this.state.user["username"]}
    </Text>
    <Text style={{ color: '#2699FB', fontSize: 15,
    position: 'absolute', left: 35, bottom: 40 }}>
    ตำแหน่ง : {this.state.user.role}</Text>
  </View>
  <View style={{ flex: 0.55, justifyContent: 'space-between' }}>
  <View style={{ flex: 0.02 }} />
  <View style={{ flex: 0.1 }}>
    <TouchableOpacity style={{ alignItems: 'center',
    padding: 10, flexDirection: 'row', marginLeft: 30 }}
    onPress={() => { try {
    this.props.navigation.navigate("EditProfile", {
    onGoBack: this.handleRefresh } ); } catch (err)
    { console.log({ err }) } }}>
    <Image style={{ width: 20, height: 20 }} source=
    {require('../assets/images/Account.png')} />
    <Text style={{ color: '#2699FB' }}>โปรไฟล์</Text>
  </TouchableOpacity>
  </View>
  <View style={{ flex: 0.1 }}>
    <TouchableOpacity style={{ alignItems: 'center',
    padding: 10, flexDirection: 'row', marginLeft: 30 }}
    onPress={() => { this.onLogout() }} >
    <Image style={{ width: 20, height: 20 }} source=
    {require('../assets/images/Logout.png')} />
    <Text style={{ color: '#2699FB' }}>ออกจากระบบ</Text>
  </TouchableOpacity>
  </View>
  <View style={{ flex: 0.3 }} />
  </View>
</View>

```

รูปที่ 4.1.13.1 ภาษาจาวาสคริปต์ หน้าการตั้งค่า



รูปที่ 4.1.13.2 หน้าจอการตั้งค่า

ฟังก์ชันที่เชื่อมต่อกับหลังบ้านในหน้าการตั้งค่ามีด้วยกัน 2 ฟังก์ชัน ได้แก่ ฟังก์ชันสำหรับแสดงข้อมูลผู้ใช้งาน และฟังก์ชันสำหรับออกจากระบบ ดังรูปที่ 4.1.13.3


```

async componentDidMount() {
  this.animation.play()
  Auth.currentUserInfo()
    .then(data => {
      API.get("userAPI", "/users/" + data.username)
        .then(query_user => {
          this.setState({ user: query_user })
          Storage.get(query_user.image)
            .then(url => {
              this.state.user.image = url
              this.setState({ isLoading: false })
            })
        })
    })
}

handleRefresh = () => {
  this.setState({ isLoading: true })
  Auth.currentUserInfo()
    .then(data => {
      API.get("userAPI", "/users/" + data.username)
        .then(query_user => {
          this.setState({ user: query_user })
          Storage.get(query_user.image)
            .then(url => {
              this.state.user.image = url
              this.setState({ isLoading: false })
            })
        })
    })
}
}

```

รูปที่ 4.1.13.3 ภาษาจาวาสคริปต์การเชื่อมต่อส่วนหลังบ้าน หน้าการตั้งค่า

4.2 การทดสอบระบบ

4.2.1 การทดสอบระบบโดยการรับข้อเสนอแนะ

ในช่วงแรกของการเก็บความต้องการของผู้ใช้งาน ทางผู้พัฒนาจำเป็นต้องมีการนำความต้องการของผู้ใช้งานมาวิเคราะห์และเขียนให้เป็นแบบแผน เพื่อนำเสนอ สอบถามและเก็บความต้องการของผู้ใช้งานในครั้งต่อไป มีประสิทธิภาพมากยิ่งขึ้น ซึ่งตารางที่ 4.2.1.1 ตารางข้อเสนอแนะ จะแสดงข้อเสนอแนะที่ได้จากผู้ใช้งานจริงของการไปเก็บความต้องการของผู้ใช้งานในแต่ละครั้ง

ตารางที่ 4.2.1.1 ตารางข้อเสนอแนะ

วันที่	บริษัท	ข้อเสนอแนะ
30/05/2562	สำนักวิจัยและพัฒนาทางกรรมทางหลวง	1. สำนักวิจัยและพัฒนาทางกรรมทางหลวง มีการดำเนินการตรวจสอบความคืบหน้าของการสร้างและแก้ไขสะพานในแต่ละครั้ง โดยการดำเนินการจะประกอบไปด้วย 2 ตำแหน่ง ได้แก่ วิศวกร และ หัวหน้าคนงาน

		<p>2. การดำเนินการของกรมทางหลวงคือการลงพื้นที่ และตรวจสอบความคืบหน้า ซึ่งกรมทางหลวงจะใช้การถ่ายรูปและส่งรายงานผล เพื่อติดตามความคืบหน้าของงาน โดยที่วิศวกรไม่จำเป็นต้องลงไปที่หน้างานเพื่อตรวจสอบ</p> <p>3. กรมทางหลวงสังเกตเห็นถึงความสำคัญของการแจ้งเตือน ว่ามีความคืบหน้าในการดำเนินการเป็นอย่างไรบ้าง ซึ่งสามารถลดระยะเวลาในการสื่อสาร และสามารถทำให้ติดตามผลได้อย่างทันที่</p>
17/10/2562	Team built CO.,LTD.	<p>1. โครงสร้างในบริษัท Team built CO., LTD. มีการแบ่งเป็น 2 ส่วนได้แก่ ส่วนของการจัดงานหน้างาน (Site) และส่วนการจัดการในบริษัท (Office) ซึ่งแอปพลิเคชันจะตอบโต้ภัยในส่วนของการจัดการหน้างาน</p> <p>2. ในการจัดงานหน้างานจะประกอบไปด้วยตำแหน่ง หลักๆ ได้แก่ Project Manager, Foreman, Admin</p> <p>3. แผนการทำงานหลัก ๆ ของบริษัท Team built CO., LTD. จำมีการเดินตรวจสอบหน้างานในช่วงเช้า โดยที่หน้าที่ของแต่ละตำแหน่ง มีดังนี้</p> <ul style="list-style-type: none"> - Project Manager มีหน้าที่ในการมอบหมายงานให้กับตำแหน่งต่าง ๆ - Foreman ตรวจสอบงาน ตามที่มีการมอบหมายงานมาจาก Project Manager - Admin ดูแลเบ็ดเตล็ด เช่น การทำเบิกของ บัญชี การสั่งซื้อของ <p>4. ปัญหาที่พบจากการใช้ระบบปัจจุบันด้วย Line คือมีการสื่อสารที่ผิดพลาดเนื่องจากมีข้อมูลที่เยอะ ในห้องสนทนาเดียวกัน ทำให้เกิดความวุ่นวาย อีกทั้งรูปภาพมีการเก็บในคลังรูปภาพ แต่ไม่มีการระบุวันที่ถ่ายหรือมีการอัปโหลด การสื่อสารไม่รู้ว่ามีสื่อสารถึงใคร ไม่มีการมอบหมายงานอย่างเป็นแบบแผน</p>
17/11/2562	ปริณสุริรี จำกัด (มหาชน)	<p>1. โครงสร้างในบริษัท ปริณสุริรี จำกัด จะมอบหมายให้ Project Manager มีตำแหน่งสูงสุด และแบ่งย่อยลงมาเป็น Foreman และฝ่ายการจัดการในบริษัท โดยที่ Project Manager สามารถติดต่อสื่อสารกับฝ่ายขายได้โดยตรงเองด้วย</p>

		<p>2. ในการจัดงานหน้างานจะประกอบไปด้วยตำแหน่ง หลักๆ ได้แก่ Project Manager, Foreman</p> <p>3. การทำงานในแต่ละตำแหน่งของบริษัท ปริณสุริ มีดังนี้</p> <ul style="list-style-type: none"> - Project Manager มีหน้าที่ในการมอบหมายงาน ให้กับตำแหน่งต่าง ๆ รวมถึงการจัดซื้อของ - Foreman ตรวจสอบหน้างานตามที่ได้รับ มอบหมายมาจาก Project Manager <p>4. ภายใน 1 โครงการมีหลาย Phase ประกอบด้วยกัน และใน แต่ละ Phase จะมีงานและงานย่อยประกอบกันลงไปอีก ยกตัวอย่างเช่น Phase ที่ 1 บ้านทาวนโฮม มีงานได้แก่ งาน เตรียมการ และม้งานย่อยอีกหลาย ๆ งาน ได้แก่ งานรั้ว งาน ไฟฟ้า งานประปาชั่วคราว บ้านคนงานชั่วคราว สำนักงาน ชั่วคราว และโกดังชั่วคราว</p> <p>5. ปัจจุบันใช้ Line ในการสื่อสาร พบว่าการสื่อสารผ่าน Line ทำให้เกิดการติดตามงานที่ยาก เนื่องจากมีข้อมูลที่เยอะ มีปัญหาใน ด้านการรายงานผลในแต่ละวัน การติดตามแผนงานว่างานที่ มอบหมายระบุให้แก่ Foreman คนไหนให้ชัดเจน การเก็บ รวบรวมข้อความ และรูปถ่าย</p>
19/12/2562	Team built CO.,LTD.	<p>มีการนำเสนอ Prototype แรกมีข้อเสนอแนะ ดังต่อไปนี้</p> <ol style="list-style-type: none"> 1. ปัจจุบันใช้โครงสร้างในรูปแบบของงานย่อย เช่น ใน 1 โครงการ มี 1 โพรเจค และใน 1 โพรเจค มี 1 งาน 2. การดูแปลนมีความจำเป็นอย่างมาก เนื่องจาก Foreman และ Project Manager จะใช้แปลนเปรียบเทียบเสมือนคู่มือในการ ตรวจสอบการก่อสร้าง 3. การอัปโหลดรูปภาพหรือการอัปเดตงานย่อย ตำแหน่งไหนก็สามารถสร้างได้ ในกรณีที่บริษัทขนาดเล็ก ทำให้มีการส่งงาน สลับไปมาได้ 4. การแจ้งเตือนมีความสำคัญ ปัจจุบันมีการแจ้งเตือนที่ไม่เกี่ยวข้องกับผู้ใช้งานโดยตรงเป็นจำนวนมาก
22/12/2562	ปริณสุริ จำกัด (มหาชน)	<p>มีการนำเสนอ Prototype มีข้อเสนอแนะ ดังต่อไปนี้</p> <ol style="list-style-type: none"> 1. ผลตอบรับจากการนำเสนอ Prototype ออกมาก่อนข้างดี ผู้ใช้งานเข้าใจถึงภาพรวมการทำงานของระบบได้อย่างถูกต้อง

		<p>การสร้าง โครงการ โปรเจค งานและงานย่อย พร้อมทั้งแสดงความคิดเห็น สามารถตอบโต้กับผู้ใช้งาน</p> <p>2. มีข้อเสนอแนะในส่วนของการออกแบบส่วนต่อประสานผู้ใช้ เพื่อให้ผู้ใช้จากหลากหลายบริษัทสามารถเข้าใจไปในทิศทางเดียวกันได้ดียิ่งขึ้น</p>
--	--	---

4.2.2 การทดสอบระบบโดยผู้พัฒนาแอปพลิเคชัน

การทดสอบระบบโดยผู้พัฒนาแอปพลิเคชันเป็นการทดสอบแบบ Functional Testing เป็นการทดสอบฟังก์ชันการทำงานของระบบ ว่าสามารถทำงานได้ตรงตามผลที่คาดหวัง โดยจะใช้การทดสอบแบบ Black Box Testing

ตารางที่ 4.2.2.1 ตารางกรณีทดสอบและผลการทดสอบโดยผู้พัฒนาแอปพลิเคชัน

เลขที่	ชื่อการทดสอบ	วิธีการทดสอบ	ข้อมูลขาเข้า	ผลที่คาดหวัง	ผลการทดสอบ
การทดสอบฟังก์ชันการทำงานหลัก					
1	การลงทะเบียน	กรอกข้อมูลในหน้าลงทะเบียนให้ครบถ้วน เลือกบทบาท และกดยืนยัน	ชื่อผู้ใช้: siradathb รหัสผ่าน: 12345678 อีเมล: siradathb@gmail.com เบอร์โทรศัพท์: 0880174749 ตำแหน่ง: Project Manager	มี Pop-up ขึ้น “การลงทะเบียนสำเร็จ” ลงทะเบียนสำเร็จ	มี Pop-up ขึ้น “การลงทะเบียนสำเร็จ” ลงทะเบียนสำเร็จ
2	เข้าสู่ระบบ	กรอกชื่อผู้ใช้งานและรหัสผ่าน จากนั้นกดปุ่มยืนยัน	ชื่อผู้ใช้: siradathb รหัสผ่าน: 12345678	มี Pop-up ขึ้น “Hello, siradathb” เข้าสู่หน้าการแจ้งเตือน	มี Pop-up ขึ้น “Hello, siradathb” เข้าสู่หน้าการแจ้งเตือน

3	ออกจาก ระบบ	กดปุ่มออกจากระบบ ในหน้าการตั้งค่า	-	ออกจาก ระบบสำเร็จ	ออกจากระบบ สำเร็จ
4	ยืนยันอีเมล	เข้าสู่ระบบครั้งแรก จะเชื่อมไปที่หน้า ยืนยันอีเมล นำ Verification code มาใส่ในช่อง แล้วกด ปุ่มยืนยัน	Verification Code: 455689	ยืนยันอีเมล สำเร็จ ย้อนกลับไป หน้าเข้าสู่ ระบบ	ยืนยันอีเมล สำเร็จ ย้อนกลับไป หน้าเข้าสู่ ระบบ
5	เพิ่มโปรเจค	กดชื่อโครงการเพื่อ แสดงแถบคำสั่ง ด้านบน กดเพิ่มโปร เจคและกรอกข้อมูล ให้ครบถ้วน จากนั้น กดยืนยัน	ชื่อโปรเจค: Phase-02 รายละเอียด: โปรเจค เฟสที่ 2 วันที่เริ่ม: 29-04-2563 วันที่จบ: 05-07-2563	แสดงโปรเจค Phase-02 ใน หน้าโครงการ	แสดงโปรเจค Phase-02 ใน หน้าโครงการ
6	ลบโปรเจค	ปิดซ้ายโปรเจคที่ ต้องการลบในหน้า โครงการ กดยืนยัน อีกครั้งเพื่อลบโปร เจค	ชื่อโปรเจค: Phase-02	โปรเจค Phase-02 หายไปจาก หน้าโครงการ	โปรเจค Phase-02 หายไปจาก หน้าโครงการ
7	แก้ไขโปร เจค	ปิดซ้ายที่โปรเจคที่ ต้องการแก้ไขในหน้า โครงการ แก้ไข ข้อมูลและกดยืนยัน	ชื่อโปรเจค: Phase-03 รายละเอียด: โปรเจค เฟสที่ 2 วันที่เริ่ม: 29-04-2563 วันที่จบ: 05-07-2563	ชื่อของโปร เจค เปลี่ยนเป็น Phase-03 ตามที่แก้ไข	ชื่อของโปรเจค เปลี่ยนเป็น Phase-03 ตามที่แก้ไข
8	เพิ่มงาน	กดชื่อโปรเจคเพื่อ แสดงแถบคำสั่ง ด้านบน กดเพิ่มงาน และกรอกข้อมูลให้	ชื่องาน: งานรากฐาน รายละเอียด: งาน รากฐาน วันที่เริ่ม: 29-04-2563 วันที่จบ: 05-05-2563	แสดงงาน งาน รากฐาน ใน หน้าของโปร เจค	แสดงงาน งาน รากฐาน ใน หน้าของโปร เจค

		ครบถ้วน จากนั้นกด ยืนยัน			
9	ลบงาน	ปิดซ้ายที่งานที่ ต้องการลบในหน้า โปรเจค กดยืนยันอีก ครั้งเพื่อลบงาน	ชื่องาน: งานรากฐาน	งาน งาน รากฐาน หายไปจาก หน้าของโปร เจค	งาน งาน รากฐาน หายไปจาก หน้าของโปร เจค
10	แก้ไขงาน	ปิดซ้ายที่งานที่ ต้องการแก้ไขในหน้า โครงการ แก้ไข ข้อมูลและกดยืนยัน	ชื่องาน: งานเทพื้น รายละเอียด: งานเทพื้น วันที่เริ่ม: 29-04-2563 วันที่จบ: 05-05-2563	ชื่อของงานที่ เปลี่ยนเป็น งานเทพื้น ตามที่แก้ไข	ชื่อของงานที่ เปลี่ยนเป็น งานเทพื้น ตามที่แก้ไข
11	กำหนด สถานะงาน	ปิดซ้ายที่งานที่ ต้องการกำหนด สถานะ กดปุ่มเสร็จ และยืนยันเพื่อ เปลี่ยนแปลงสถานะ ของงาน โดยสีแดง หมายถึงงานยังไม่ เสร็จ สีเขียวหมายถึง งานเสร็จสิ้นแล้ว	สถานะ: เสร็จ	สีของสถานะ ของงาน เปลี่ยนไป ตามที่กำหนด	สีของสถานะ ของงาน เปลี่ยนไป ตามที่กำหนด
12	เพิ่มปัญหา	กรอกชื่อและ รายละเอียดของ ปัญหาในหน้างาน และกดปุ่มยืนยัน	ชื่อปัญหา: ติดตามความ คืบหน้า รายละเอียด: ดำเนินการ ติดตามความคืบหน้า	แสดงปัญหา ติดตามความ คืบหน้า ใน หน้าของงาน	แสดงปัญหา ติดตามความ คืบหน้า ใน หน้าของงาน
13	ลบปัญหา	ปิดซ้ายที่ปัญหาที่ ต้องการลบ กด ยืนยันอีกครั้งเพื่อลบ งาน	ชื่อปัญหา: ติดตามความ คืบหน้า	ปัญหา ติดตามความ คืบหน้า	ปัญหา ติดตาม ความคืบหน้า หายไปจาก หน้างาน

				หายไปจาก หน้างาน	
14	เพิ่มความ คิดเห็น	กรอกความคิดเห็น ในช่องแสดงความ คิดเห็นในหน้าปัญหา กดปุ่มยืนยัน	ความคิดเห็น: มีความ คืบหน้า 70%	ความคิดเห็น มีความ คืบหน้า 70% เพิ่มในหน้า ของปัญหา	ความคิดเห็น มีความ คืบหน้า 70% เพิ่มในหน้า ของปัญหา
15	ดูแปลน	กดชื่อโครงการหรือ ชื่องานเพื่อแสดง คำสั่งด้านบน จากนั้นกดที่แปลน เลือกดูรูปที่ ต้องการ	-	สามารถดู รูปภาพแบบ ขยายได้ที่ละ รูปภาพ	สามารถดู รูปภาพแบบ ขยายได้ที่ละ รูปภาพ
16	ส่งอีเมล	กดชื่อโครงการเพื่อ แสดงแถบคำสั่ง ด้านบน จากนั้นกดที่ อีเมล กรอกอีเมลที่ ต้องการส่งถึงและ ข้อความ จากนั้นกด ยืนยัน	ส่งอีเมลถึง: tester2@gmail.com หัวข้อ: ขอดำเนินการ งบประมาณ	เปิดแอปพลิเคชัน อีเมลใน โทรศัพท์ พร้อมระบุผู้ที่ ส่งอีเมลถึง และหัวข้อ	เปิดแอปพลิเคชัน อีเมลใน โทรศัพท์ พร้อมระบุผู้ที่ ส่งอีเมลถึง และหัวข้อ
17	เพิ่ม Foreman	กดปุ่มบวกสีเทาใน หน้าแก้ไข Foreman เลือกชื่อ Foreman ที่ ต้องการเพิ่ม แล้วกด ยืนยัน	Foreman: Tester2	ชื่อของ Tester2 ที่ถูก เพิ่ม แสดงใน หน้าแก้ไข Foreman	ชื่อของ Tester2 ที่ถูก เพิ่ม แสดงใน หน้าแก้ไข Foreman
18	ลบ Foreman	ปิดซ้ายที่ Foreman ที่ต้องการลบ กด	Foreman: Tester2	ชื่อของ Tester2 ที่ถูก	ชื่อของ Tester2 ที่ถูก

		ยืนยันอีกครั้งเพื่อลบ Foreman		ลบ หายไป จากหน้าแก้ไข Foreman	ลบ หายไป จากหน้าแก้ไข Foreman
การทดสอบฟังก์ชันการทำงานที่เกิดจากความผิดพลาดของมนุษย์					
19	กรอกชื่อ ผู้ใช้งานหรือ รหัสผ่าน ไม่ ถูกต้อง	กรอกชื่อผู้ใช้งาน และรหัสผ่าน จากนั้นกดปุ่มยืนยัน	ชื่อผู้ใช้: siradathb รหัสผ่าน: abc	มี Pop-up การแจ้งเตือน ขึ้น “ชื่อ ผู้ใช้งานหรือ รหัสผ่านไม่ ถูกต้อง”	มี Pop-up การแจ้งเตือน ขึ้น “ชื่อ ผู้ใช้งานหรือ รหัสผ่านไม่ ถูกต้อง”
20	การ ลงทะเบียน กรอกข้อมูล ไม่ครบถ้วน	กรอกข้อมูลในหน้า ลงทะเบียน เลือก บทบาท และกด ยืนยัน	ชื่อผู้ใช้: รหัสผ่าน: 12345678 อีเมล: siradathb@gmail.com เบอร์โทรศัพท์: 0880174749 ตำแหน่ง: Project Manager	มี Pop-up การแจ้งเตือน ขึ้น “กรุณา กรอกข้อมูลให้ ครบถ้วน”	มี Pop-up การแจ้งเตือน ขึ้น “กรุณา กรอกข้อมูลให้ ครบถ้วน”
21	เพิ่มโปรเจค กรอกข้อมูล ไม่ครบถ้วน	กดชื่อโครงการเพื่อ แสดงแถบคำสั่ง ด้านบน กดเพิ่มโปร เจคและกรอกข้อมูล จากนั้นกดยืนยัน	ชื่อโปรเจค: รายละเอียด: โปรเจค เฟสที่ 2 วันที่เริ่ม: 29-04-2563 วันที่จบ: 05-07-2563	มี Pop-up การแจ้งเตือน ขึ้น “กรุณา กรอกข้อมูลให้ ครบถ้วน”	มี Pop-up การแจ้งเตือน ขึ้น “กรุณา กรอกข้อมูลให้ ครบถ้วน”
22	เพิ่มงาน กรอกข้อมูล ไม่ครบถ้วน	กดชื่อโปรเจคเพื่อ แสดงแถบคำสั่ง ด้านบน กดเพิ่มงาน และกรอกข้อมูล จากนั้นกดยืนยัน	ชื่องาน: รายละเอียด: งาน รากฐาน วันที่เริ่ม: 29-04-2563 วันที่จบ: 05-05-2563	มี Pop-up การแจ้งเตือน ขึ้น “กรุณา กรอกข้อมูลให้ ครบถ้วน”	มี Pop-up การแจ้งเตือน ขึ้น “กรุณา กรอกข้อมูลให้ ครบถ้วน”
23	กรอกรหัส Verification	กรอกรหัส Verification Code ที่เป็นตัวเลขจำนวน	Verification Code: 1234	มี Pop-up การแจ้งเตือน ขึ้น	มี Pop-up การแจ้งเตือน ขึ้น

	Code ไม่ถูกต้อง	6 หลัก ในหน้า Verification		“Verification Code ไม่ถูกต้อง”	“Verification Code ไม่ถูกต้อง”
24	Foreman เพิ่มโปรเจค	ผู้ใช้งานที่มีตำแหน่ง Foreman กดปุ่มเพิ่มโปรเจคในหน้าโครงการ	-	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถเพิ่มโปรเจคได้”	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถเพิ่มโปรเจคได้”
25	Foreman แก้ไขโปรเจค	ผู้ใช้งานที่มีตำแหน่ง Foreman กดปุ่มแก้ไขโปรเจคในหน้าโครงการ	-	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถแก้ไขโปรเจคได้”	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถแก้ไขโปรเจคได้”
26	Foreman ลบโปรเจค	ผู้ใช้งานที่มีตำแหน่ง Foreman กดปุ่มลบโปรเจคในหน้าโครงการ	-	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถลบโปรเจคได้”	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถลบโปรเจคได้”
27	Foreman เพิ่มงาน	ผู้ใช้งานที่มีตำแหน่ง Foreman กดปุ่มเพิ่มงานในหน้าโปรเจค	-	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถเพิ่มงานได้”	มี Pop-up การแจ้งเตือนขึ้น “Foreman ไม่สามารถเพิ่มงานได้”
28	Foreman ลบงาน	ผู้ใช้งานที่มีตำแหน่ง Foreman กดปุ่มลบงานในหน้าโปรเจค	-	มี Pop-up การแจ้งเตือนขึ้น “Foreman	มี Pop-up การแจ้งเตือนขึ้น “Foreman

				ไม่สามารถลบบงานได้”	ไม่สามารถลบบงานได้”
29	Foreman แก้ไขงาน	ผู้ใช้งานที่มีตำแหน่ง Foreman กดปุ่ม แก้ไขงานในหน้าโปร เจค	-	มี Pop-up การแจ้งเตือน ขึ้น “Foreman ไม่สามารถ แก้ไขงานได้”	มี Pop-up การแจ้งเตือน ขึ้น “Foreman ไม่สามารถ แก้ไขงานได้”
30	การกดลบ โปรเจค ผิดพลาด	ผู้ใช้งานกดปุ่มลบ โปรเจคโดยที่ไม่ได้ เกิดจากความตั้งใจ หรือเกิดจากความ ผิดพลาด	-	มี Pop-up การแจ้งเตือน ขึ้น “ต้องการ ยืนยันที่จะลบ โปรเจคนี้ หรือไม่” เพื่อ ยืนยันว่าจะ ดำเนินการลบ โปรเจคนี้	มี Pop-up การแจ้งเตือน ขึ้น “ต้องการ ยืนยันที่จะลบ โปรเจคนี้ หรือไม่” เพื่อ ยืนยันว่าจะ ดำเนินการลบ โปรเจคนี้
31	การกดลบ งาน ผิดพลาด	ผู้ใช้งานกดปุ่มลบ งานโดยที่ไม่ได้เกิด จากความตั้งใจ หรือ เกิดจากความ ผิดพลาด	-	มี Pop-up การแจ้งเตือน ขึ้น “ต้องการ ยืนยันที่จะลบ งานนี้หรือไม่” เพื่อยืนยันว่า จะดำเนินการ ลบงานนี้	มี Pop-up การแจ้งเตือน ขึ้น “ต้องการ ยืนยันที่จะลบ งานนี้หรือไม่” เพื่อยืนยันว่า จะดำเนินการ ลบงานนี้

4.2.3 การทดสอบระบบโดยผู้ใช้งานจริง

ผู้พัฒนาแอปพลิเคชันได้นำแอปพลิเคชันให้ผู้ทดลองใช้ทำการทดสอบ โดยผู้ทดสอบเป็น Foreman และ Project Manager จากบริษัทปริณสุริ และ Team Built รวมจำนวนผู้ใช้งานจริง ทั้งหมด 13 คน ผลการทดสอบการทดลองใช้ ได้ดังตาราง 4.2.3.1

ตารางที่ 4.2.3.1 ผลการทดสอบการทดลองใช้งานจากผู้ใช้งานจริง

รายละเอียด	ดีเยี่ยม (5)	ดี (4)	ปานกลาง (3)	พอใช้ (2)	ควรปรับปรุง (1)
1. การออกแบบส่วนต่อประสานผู้ใช้					
1.1 ตัวอักษรเป็นระเบียบ อ่านง่าย	2 (15%)	8 (62%)	3 (23%)		
1.2 คำและข้อความที่ใช้สื่อความหมาย	4 (31%)	6 (46%)	3 (23%)		
1.3 สี สัน และความสวยงาม	3 (23%)	7 (54%)	3 (23%)		
2. การทดสอบใช้งานจริง					
2.1 สามารถเพิ่ม ลบ โครงการ โปรเจค งาน และปัญหาได้ถูกต้อง	2 (15%)	7 (54%)	4 (31%)		
2.2 สามารถแนบรูปในการเพิ่มโปรเจค และงานได้	7 (54%)	4 (31%)	2 (15%)		
2.3 สามารถแสดงความคิดเห็นในปัญหาห้อยได้	4 (31%)	6 (46%)	3 (23%)		
2.4 ระบบการแจ้งเตือน สามารถแจ้งเตือนได้ถูกต้องและทันท่วงที	4 (31%)	6 (46%)	2 (15%)	1 (8%)	
2.5 สามารถดูแปลนทั้งหมดของโครงการได้	8 (62%)	3 (23%)	2 (15%)		
2.6 สามารถเพิ่ม ลบ Foreman ที่เกี่ยวข้องกับงานนั้น ๆ ได้	3 (23%)	7 (54%)	3 (23%)		
2.7 สามารถกำหนดสถานะของงาน และโปรเจคได้	4 (31%)	6 (46%)	3 (23%)		
2.8 สามารถส่งอีเมลได้	6 (46%)	5 (39%)	2 (15%)		
3. ประโยชน์ต่อผู้ใช้งาน					
3.1 ทำให้การสื่อสารภายในโครงการ มีความรวดเร็วมากขึ้น	2 (15%)	8 (62%)	3 (23%)		

3.2 ทำให้สามารถติดตามความคืบหน้า ของงานได้ทัน่วงที่	2 (15%)	9 (70%)	2 (15%)		
3.3 สามารถดูข้อมูลต่าง ๆ ในโครงการได้ทัน่วงที่	2 (15%)	7 (54%)	4 (31%)		

ตารางที่ 4.2.3.2 สรุปผลการทดลองใช้งานจากผู้ใช้งานจริง

รายละเอียด	คะแนนการทดสอบ
1. การออกแบบส่วนต่อประสานผู้ใช้	
1.2 ตัวอักษรเป็นระเบียบ อ่านง่าย	3.92
1.2 คำและข้อความที่ใช้สื่อความหมาย	4.08
1.3 สี สัน และความสวยงาม	4.00
2. การทดสอบใช้งานจริง	
2.1 สามารถเพิ่ม ลบ โครงการ โปรเจค งานและปัญหาได้ถูกต้อง	3.85
2.2 สามารถแนบรูปในการเพิ่มโปรเจค และงานได้	4.38
2.3 สามารถแสดงความคิดเห็นในปัญหาย่อยได้	4.08
2.4 ระบบการแจ้งเตือน สามารถแจ้งเตือนได้ถูกต้องและทัน่วงที่	4.08
2.5 สามารถดูแผนทั้งหมดของโครงการได้	4.46
2.6 สามารถเพิ่ม ลบ Foreman ที่เกี่ยวข้องกับงานนั้น ๆ ได้	4.00
2.7 สามารถกำหนดสถานะของงาน และโปรเจคได้	4.08
2.8 สามารถส่งอีเมลได้	4.31
3. ประโยชน์ต่อผู้ใช้งาน	
3.1 ทำให้การสื่อสารภายในโครงการ มีความรวดเร็วมากขึ้น	3.92
3.2 ทำให้สามารถติดตามความคืบหน้า ของงานได้ทัน่วงที่	4.00

3.3 สามารถดูข้อมูลต่าง ๆ ในโครงการได้ ทันทีทันที	3.85
--	------

จากการสรุปผลพบว่าคะแนนการออกแบบส่วนต่อประสานผู้ใช้งานมีคะแนนเฉลี่ย 4.00 คะแนน
คะแนนการทดสอบการใช้งานจริงมีคะแนนเฉลี่ยที่ 4.16 คะแนน และคะแนนความเป็นประโยชน์ต่อผู้ใช้งานมี
คะแนนเฉลี่ยที่ 3.92 สรุปคะแนนรวมทั้งหมดจะมีคะแนนอยู่ที่ 4.03 คะแนน

บทที่ 5

ข้อสรุปและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงการสรุปผล ปัญหาและอุปสรรค วิธีการแก้ปัญหา และข้อเสนอแนะในการพัฒนาระบบ

5.1 สรุปผล

ผู้พัฒนาได้จัดทำแอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง ซึ่งแอปพลิเคชันนี้จะช่วยในการสื่อสารภายในองค์กรและบริษัท รวมถึงอุตสาหกรรมก่อสร้างต่าง ๆ ให้สามารถลดข้อผิดพลาดในการสื่อสาร มีความเข้าใจที่ตรงกัน สามารถติดตามงานได้อย่างทันท่วงทีและมีแบบแผนที่แน่นอน อีกทั้งการถ่ายรูปแนบไปในการทำงานก็สามารถทำได้โดยง่าย ทำให้งานสามารถดำเนินการไปด้วยดีมากขึ้น สามารถลดระยะเวลาการสื่อสาร และทางผู้พัฒนา ก็ได้ทำแบบทดสอบเพื่อทดสอบความต้องการของผู้ใช้งาน ว่าการพัฒนาแอปพลิเคชันจะตรงตามจุดประสงค์ของผู้ใช้งานจริง สำหรับสิ่งที่ผู้พัฒนาได้เรียนรู้จากการพัฒนาแอปพลิเคชันนี้คือ สามารถนำความรู้ที่ได้จากการค้นคว้าต่าง ๆ และความรู้ที่ได้จากการออกแบบและพัฒนาระบบ ไปต่อยอดแอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้างต่อไป

5.2 ปัญหาและอุปสรรค

1. การรวบรวมความต้องการจากลูกค้า มีการสื่อสารที่ไม่เข้าใจกันในบางที่ ซึ่งอาจทำให้เกิดความล่าช้าและความผิดพลาดจากทั้งสองฝ่าย
2. การพัฒนาระบบใช้เวลานานกว่าที่คาดการณ์ไว้ ทำให้กำหนดการมีความล่าช้า
3. เนื่องด้วยจากสถานการณ์ COVID-19 ทำให้ไม่สามารถทดสอบการใช้งานจริงกับผู้ใช้งานจริงได้ ซึ่งการส่งแอปพลิเคชันไปให้ผู้ใช้งานจริง นำไปทดลองลองใช้งาน โดยที่ไม่มีการนำเสนอและสอนการใช้งานเบื้องต้น ทำให้ผู้ใช้งานจริง ไม่เข้าใจถึงการใช้งานที่ถูกต้องของแอปพลิเคชัน

5.3 วิธีการแก้ปัญหา

1. ควรมีการสรุปความต้องการที่แน่ชัด หลังจากเข้าไปเก็บความต้องการจากลูกค้าแล้ว และมีการยืนยันความต้องการของลูกค้าอีกรอบ เพื่อไม่ให้เกิดข้อผิดพลาด

2. วางแผนการทำให้ดีขึ้น และควรมีการศึกษาในด้านของการพัฒนาต่าง ๆ ให้มีความเชี่ยวชาญ เพื่อให้สามารถทำงานได้อย่างรวดเร็วยิ่งขึ้น
3. ใช้การทดสอบแบบออนไลน์กับผู้ใช้งาน โดยเป็นการทำแบบประเมินผ่าน Google Form และประเมินผลเบื้องต้นจากการทดลองใช้งานเอง จากผู้ใช้งานจริง

5.4 ข้อเสนอแนะในการพัฒนาระบบ

จากการนำแอปพลิเคชันไปทดสอบกับผู้ใช้งานจริงพบว่า แอปพลิเคชันมีส่วนที่จำเป็นจะต้องพัฒนาต่อไป เพื่อให้รองรับการใช้งานจากหลากหลายบริษัทมากขึ้น เนื่องด้วยแต่ละบริษัทมีระบบการจัดการ ภาษา และการเรียกใช้ที่แตกต่างกัน ซึ่งอาจทำให้แอปพลิเคชันบางส่วน ไม่ตอบสนองกับความต้องการของผู้ใช้งานจริงจากแต่ละบริษัท และมีบางส่วนที่อาจจะเกิดจากความผิดพลาดของระบบ

ตารางที่ 5.4.1 ตารางเปรียบเทียบแอปพลิเคชัน และโปรแกรมอื่น ๆ

ฟังก์ชันการใช้งาน	แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง	แอปพลิเคชัน Plangrid	แอปพลิเคชัน Sitewalk
1. สามารถเพิ่มโครงการ โปรเจค และงานได้	○	○	○
2. สามารถเพิ่มงานย่อยภายในงานได้	○		
3. มีการสื่อสารในรูปแบบของการ Chat	○	○	
4. สามารถระบุบุคคลที่รับมอบหมายงานได้	○		○
5. สามารถอัปโหลดมัลติมีเดียได้	○	○	○
6. มีสถานะรายงานความคืบหน้าของงาน	○	○	
7. มีระบบการแจ้งเตือน	○	○	○
8. สามารถส่งรายงานโดยใช้จดหมายอิเล็กทรอนิกส์	○	○	○

ภาคผนวก

ภาคผนวก ก

แบบเสนอหัวข้อโครงการ รายวิชา 2301399 Project Proposal ปีการศึกษา 2562

ชื่อโครงการ (ภาษาไทย)	แอปพลิเคชันช่วยในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง (ระบบส่วนหน้าบ้าน)
ชื่อโครงการ(ภาษาอังกฤษ)	Construction process quality control assisted application (Front-end)
อาจารย์ที่ปรึกษา	รองศาสตราจารย์ ดร.นกุล คูหะโรจนานนท์
ผู้ดำเนินการ	1. นางสาวศิตาภัทร วงษาเนา 5933607723 2. นางสาวศิรดา ทาบสุวรรณ 5933654123 สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

หลักการและเหตุผล

ในปัจจุบันมีการสื่อสารกันผ่านอินเทอร์เน็ตกันมากขึ้นตั้งแต่การสื่อสารระหว่างบุคคล การสื่อสารภายในองค์กรและบริษัทอื่น ๆ รวมไปถึงบริษัทอุตสาหกรรมก่อสร้างต่าง ๆ ซึ่งภายในองค์กรนั้นมีการสื่อสารระหว่างองค์กรผ่านอินเทอร์เน็ตและใช้แอปพลิเคชันในการสื่อสารกัน โดยเป็นการใช้แอปพลิเคชันที่ใช้กันอย่างแพร่หลาย เช่น แอปพลิเคชัน LINE [1] ซึ่งการสื่อสารผ่านทางแอปพลิเคชัน LINE จะเกิดปัญหาส่วนใหญ่ เช่น เกิดปัญหาในการอ่านข้อความแล้วมีการสื่อสารผิด หรือส่งถึงผิดบุคคลทำให้เกิดความเข้าใจผิดระหว่างฝ่ายได้ [2] สื่อสารแล้วผู้รับสารไม่ได้รับสาร ไม่มีบุคคลตอบรับสารหรืองานจากผู้ส่งสาร เนื่องจากมีคนสื่อสารจำนวนมากภายในการสื่อสารนั้น สื่อสารไม่ครบตามเนื้อหาที่ต้องการจะสื่อสาร จากการที่ได้รับโจทย์จากภาคอุตสาหกรรมมานั้น ปัญหาที่เกิดจากการสื่อสารภายในองค์กรก่อสร้างที่พบเจอบ่อย ๆ ได้แก่ ไม่สามารถติดตามหรือรายงานผลได้อย่างมีแบบแผน บทสนทนาที่มีความสับสนเนื่องจากมีบุคคลภายในบทสนทนาเป็นจำนวนมาก จนทำให้ยากต่อการที่จะอ่านในทุก ๆ ประโยค อีกทั้งยังมีปัญหาของพนักงานก่อสร้างที่ต้องส่งรายงานเกี่ยวกับความคืบหน้าที่จะต้องถ่ายภาพเพื่อรายงานประจำวัน ซึ่งปัญหาที่เกิดขึ้นคือ ถ่ายภาพแต่ละจุดไม่ครบ หรืออาจจะลืมถ่ายภาพบางจุดที่สำคัญ เช่น ลืมถ่ายภาพจุดที่พบปัญหาเพื่อที่จะแจ้งให้ผู้ที่จัดการโปรเจกทรอบ จึงอาจจะทำให้งานก่อสร้างเกิดเสร็จล่าช้า ทำให้ส่งผลเสียต่อภาพรวมขององค์กรอีกด้วย

ผู้จัดทำจึงได้พัฒนาแอปพลิเคชันที่จะมาช่วยเหลือในการตรวจสอบคุณภาพสำหรับกระบวนการก่อสร้าง เพื่อลดระยะเวลาในการสื่อสารในกระบวนการก่อสร้าง

วัตถุประสงค์

1. เพื่อสร้างแอปพลิเคชันที่ช่วยลดระยะเวลาในกระบวนการก่อสร้างย่อย ๆ
2. สามารถดูภาพรวมของโครงการผ่านทางระบบออนไลน์แบบประมวลแบบทันที

ขอบเขตของโครงการ

1. บริษัทก่อสร้างที่ให้ข้อมูลความต้องการของผู้ใช้งาน
 - 1.1 บริษัท Team built CO.,LTD. โดยประธานกรรมการบริหาร
 - 1.2 สำนักวิจัยและพัฒนาทาง กรมทางหลวง โดยวิศวกรระดับสูง
 - 1.3 บริษัท ปริณสุริ จำกัด (มหาชน) โดยผู้จัดการโครงการ
2. ข้อมูลความต้องการของผู้ใช้งาน

ตารางที่ 1 แสดงข้อมูลความต้องการของผู้ใช้งานจำแนกตามบริษัท

ความต้องการของผู้ใช้งาน	บริษัท Team built CO.,LTD.	สำนักวิจัยและพัฒนาทาง กรมทางหลวง	บริษัท ปริณสุริ จำกัด (มหาชน)
1. แอปพลิเคชันที่สามารถสื่อสารได้คล้ายกับแอปพลิเคชัน LINE	○		○
2. สามารถรายงานความคืบหน้าให้กับผู้บริหารระดับสูงได้ด้วยระบบแบบทันที	○	○	○
3. สามารถอัปโหลดมัลติมีเดียได้ เช่น รูปภาพ วีดีโอ แบบแปลนโครงสร้าง	○	○	○

4. มีสถานะรายงานความคืบหน้าของแต่ละงานย่อย	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5. มีระบบการแจ้งเตือน	<input type="radio"/>		<input type="radio"/>
6. สามารถระบุบุคคลที่รับมอบหมายงานได้	<input type="radio"/>		<input type="radio"/>
7. สามารถส่งรายงานโดยใช้จดหมายอิเล็กทรอนิกส์	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

3. iPhone 6 ขนาดหน้าจอสำหรับการแสดงผล 4.7 นิ้ว

วิธีการดำเนินงาน

ขั้นตอนการวิจัย

1. ศึกษาการใช้งานของ React Native, VScode และ Xcode
2. รวบรวมความต้องการของผู้ใช้งาน
3. ออกแบบรูปร่างต้นแบบของแอปพลิเคชันด้วย Adobe XD
4. พัฒนาแอปพลิเคชันด้วย React Native
5. นำข้อมูลจากระบบหลังบ้านมาแสดงผล
6. ทดสอบประสิทธิภาพในด้านการใช้งาน
7. สรุปผลการดำเนินงานและจัดทำเอกสารประกอบโครงการ

ระยะเวลาการดำเนินงาน

ขั้นตอนการดำเนินงาน	ปี พ.ศ. 2562				ปี พ.ศ. 2563		
	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.
1. ศึกษาการใช้งานของ React Native, VScode และ Xcode							
2. รวบรวมความต้องการผู้ใช้งาน							
3. ออกแบบรูปร่างต้นแบบของแอปพลิเคชันด้วย Adobe XD							
4. พัฒนาแอปพลิเคชันด้วย ReactNative							
5. นำข้อมูลจากระบบหลังบ้านมาแสดงผล							
6. ทดสอบประสิทธิภาพในด้านการใช้งาน							
7. สรุปผลการดำเนินงาน และจัดทำเอกสารประกอบโครงการ							

ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ต่อผู้พัฒนาโครงการ

1. ได้พัฒนาทักษะการเขียนภาษาจาวาสคริปต์และการใช้ React Native
2. ได้พัฒนาความรู้ในด้าน User Interface Design และ User Experience
3. ได้พัฒนาทักษะการคิด วิเคราะห์ และการทำงานเป็นทีม

ประโยชน์ต่อผู้ใช้งาน

1. ลดระยะเวลาการสื่อสารในกระบวนการก่อสร้าง
2. สามารถดูข้อมูลของโครงการผ่านทางระบบออนไลน์แบบประมวลแบบทันที

อุปกรณ์และเครื่องมือที่ใช้

1. ฮาร์ดแวร์
 - 1.1 เครื่องคอมพิวเตอร์ส่วนบุคคล หรือคอมพิวเตอร์พกพาที่มีคุณสมบัติดังนี้
Macbook Pro 13 นิ้ว Processor 3.1 GHz intel core i5
 - 1.2 iPhone 6 ขนาดหน้าจอสำหรับการแสดงผล 4.7 นิ้ว
 - 1.3 เมาส์ไร้สาย Magic mouse 2
 - 1.4 คีย์บอร์ดไร้สาย Magic Keyboard
2. ซอฟต์แวร์
 - 2.1 React Native
 - 2.2 VScode
 - 2.3 Xcode
 - 2.4 Adobe Illustrator
 - 2.5 Adobe XD
 - 2.6 Adobe Photoshop
3. อื่นๆ
 - 3.1 GitHub
 - 3.2 กระดาษ A4

งบประมาณ

1. ค่าคีย์บอร์ดไร้สาย Magic Keyboard จำนวน 2 ชิ้น	6,580 บาท
2. ค่าเมาส์ไร้สาย Magic Mouse	2,290 บาท
3. ค่าเมาส์ไร้สาย Logitech	1,299 บาท
รวม	10,139 บาท

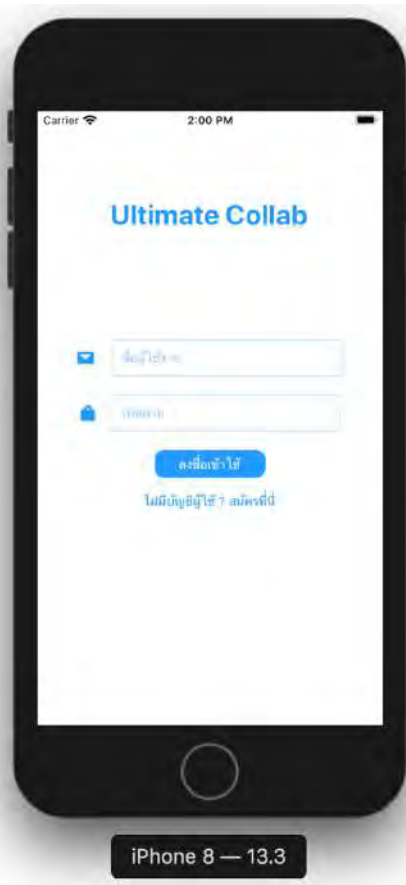
หมายเหตุ ค่าใช้จ่ายทั้งหมดถ้วนเฉลี่ยกันทุกรายการ

ภาคผนวก ข

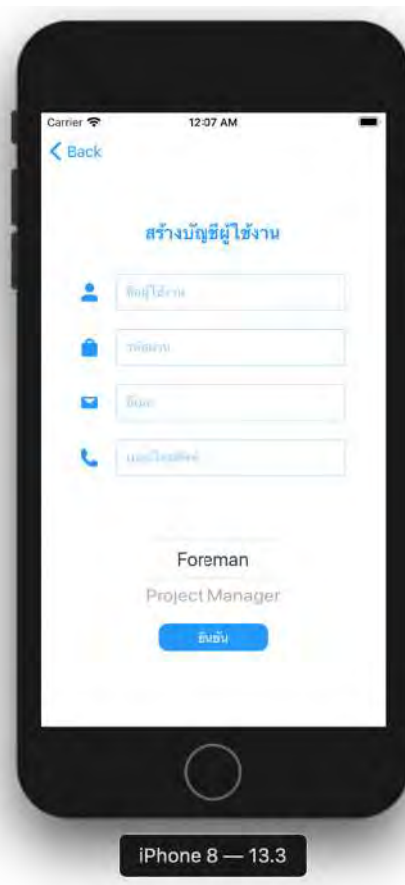
คู่มือการใช้งาน

คู่มือการใช้งานแอปพลิเคชันช่วยในการตรวจสอบกระบวนการก่อสร้าง

1. เปิดแอปพลิเคชัน Ultimate Collab ที่ติดตั้งอยู่ในโทรศัพท์มือถือ
2. หากเคยลงทะเบียนก่อนแล้วให้ทำการกรอก ชื่อผู้ใช้งานและรหัสผ่าน แต่หากไม่เคยลงทะเบียนมาก่อน ให้ทำการลงทะเบียนก่อน โดยกดที่ปุ่ม “ไม่มีบัญชีผู้ใช้ ? สมัครที่นี่” ดังรูปที่ ข-1 ระบบจะนำไปสู่หน้าลงทะเบียน



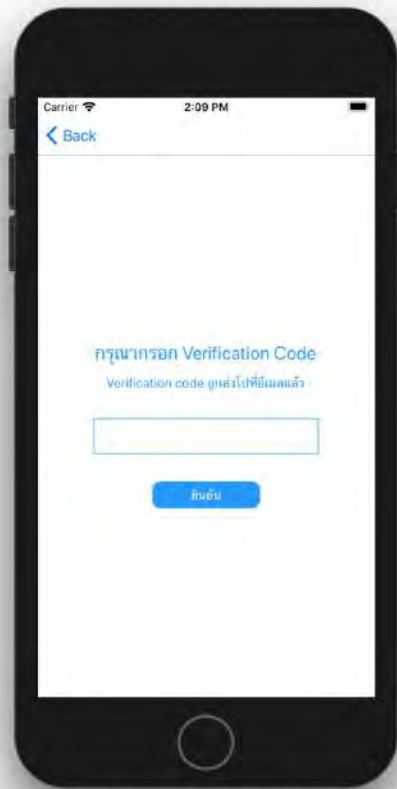
รูปที่ ข-1 รูปเมื่อเข้าสู่แอปพลิเคชัน



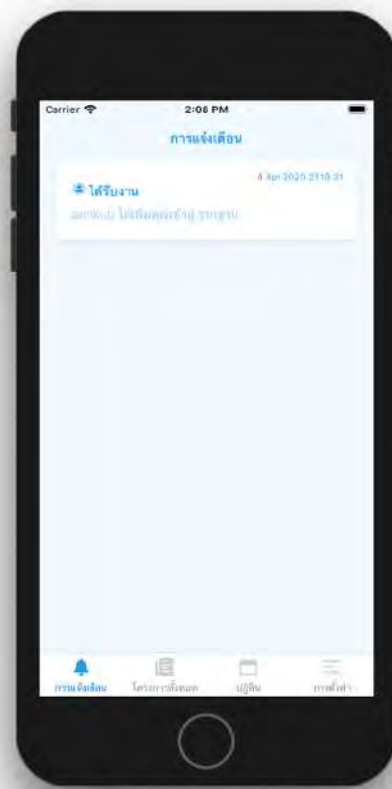
รูปที่ ข-2 รูปหน้าลงทะเบียน

3. ทำการกรอกข้อมูลในการสร้างบัญชีผู้ใช้งานให้ครบถ้วน เลือกตำแหน่งและกดยืนยัน ดังรูปที่ ข-2 เพื่อยืนยันการลงทะเบียน ระบบจะนำไปสู่หน้าเข้าสู่ระบบอีกครั้ง

4. หากเป็นการเข้าสู่ระบบครั้งแรกหลังจากการลงทะเบียน จำเป็นต้องนำ Verification Code ที่ส่งไปทางอีเมลที่ลงทะเบียนไว้ มากรอกในหน้านี้ ดังรูปที่ ข-3 เมื่อกรอกเสร็จจากนั้นกดยืนยัน เพื่อเข้าสู่ระบบ



iPhone 8 — 13.3

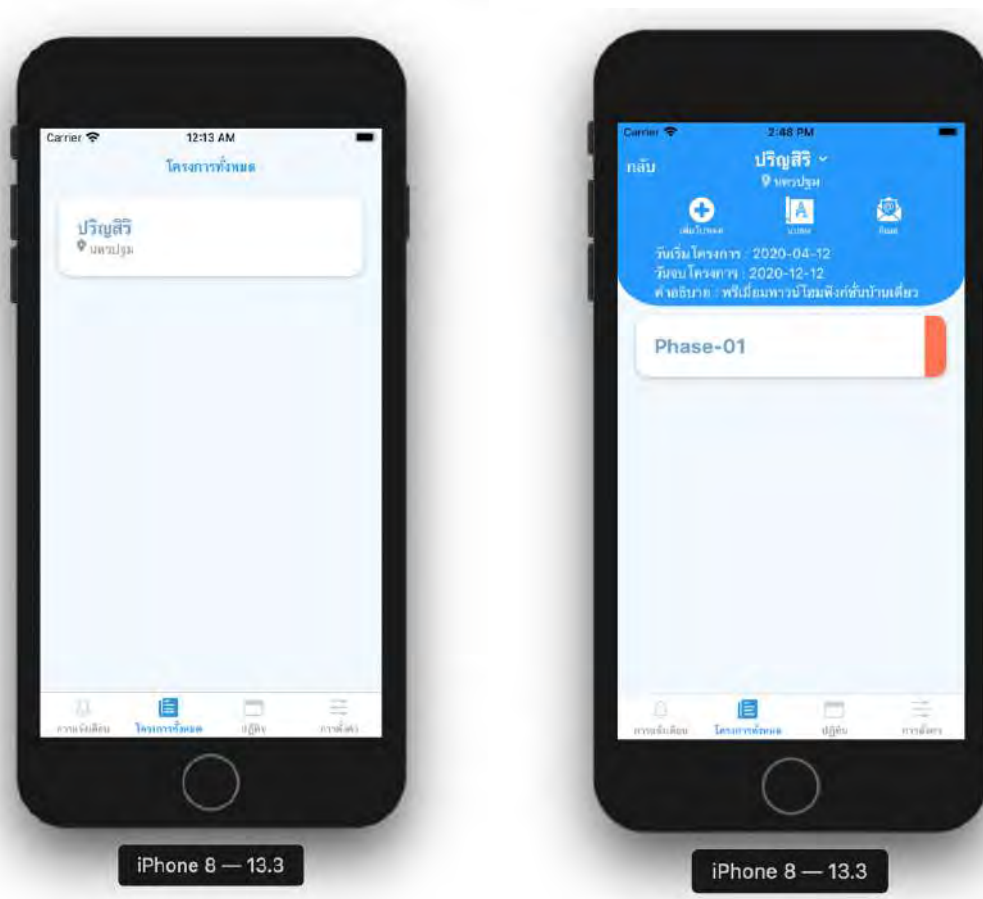


iPhone 8 — 13.3

รูปที่ ข-3 รูปหน้ายืนยัน Verification Code

รูปที่ ข-4 รูปหน้าการแจ้งเตือน

5. หลังจากเข้าสู่ระบบสำเร็จ ระบบจะพาไปที่หน้าการแจ้งเตือนเป็นหน้าแรก ดังรูปที่ ข-4 ในหน้านี้ ผู้ใช้งานจะได้ทราบถึงการแจ้งเตือนจากงานต่าง ๆ ได้แบบทันท่วงที และสามารถกดการแจ้งเตือนที่ต้องการ ระบบจะนำไปสู่หน้าจอให้ทันที
6. หน้าโครงการทั้งหมด จะแสดงโครงการทั้งหมดที่มีอยู่ ระบุชื่อของโครงการและตำแหน่งที่ตั้งของโครงการ เมื่อกดที่โครงการระบบจะนำไปสู่หน้าโปรเจค รูปที่ ข-5



รูปที่ ข-5 รูปหน้าโครงการทั้งหมด

รูปที่ ข-6 รูปหน้าโครงการ

7. หน้าโครงการจะบอกถึงรายละเอียดของโครงการ สถานที่ตั้ง วันที่เริ่มโครงการ วันจบโครงการ และคำอธิบาย โดยใน 1 โครงการจะประกอบไปด้วยโปรเจกย่อย ๆ หลายโปรเจก ซึ่งแต่ละโปรเจกสามารถปิดซ้าย เพื่อแก้ไขและลบโปรเจกได้ ซึ่งผู้ใช้งานที่มีตำแหน่งเป็น Project Manager เท่านั้นที่จะสามารถเพิ่มโปรเจกได้ รูปที่ ข-6
8. หน้าโปรเจกจะบอกถึงรายละเอียดของโปรเจก สถานที่ตั้ง วันที่เริ่ม วันสิ้นสุด และคำอธิบาย โดยใน 1 โปรเจกจะประกอบไปด้วยงานย่อย ๆ หลายงาน ซึ่งแต่ละงานสามารถปิดซ้าย เพื่อแก้ไข ลบงาน และกำหนดสถานะของงานได้ ซึ่งผู้ใช้งานที่มีตำแหน่งเป็น Project Manager เท่านั้นที่จะสามารถเพิ่มงานได้ รูปที่ ข-7



iPhone 8 — 13.3



iPhone 8 — 13.3

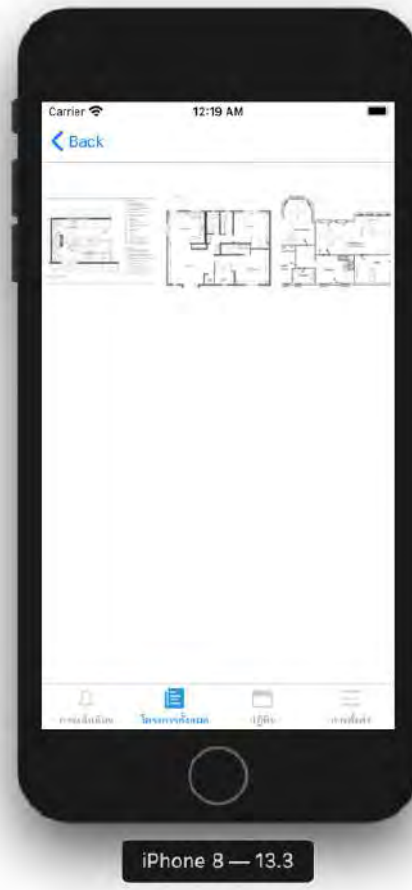
รูปที่ ข-7 รูปหน้าโปรเจค

รูปที่ ข-8 รูปหน้างาน

9. หน้างานจะบอกถึงรายละเอียดของงาน ชื่องาน วันเริ่มและวันสิ้นสุด และใน 1 งานจะประกอบไปด้วยปัญหาย่อย ๆ หลายปัญหา ซึ่งแต่ละปัญหาสามารถปิดซ้าย เพื่อลบได้ โดยผู้ใช้งานสามารถเพิ่มปัญหาที่พบเกี่ยวกับงานนั้น ๆ ลงในช่อง ระบุปัญหา พร้อมใส่คำอธิบายและแนบรูปเพิ่มเติมได้ทั้งหมด 3 รูป จากนั้นกดยืนยัน รูปที่ ข-8
10. หน้าปัญหาจะเป็นการดูว่าปัญหามีการดำเนินการอย่างไรบ้าง โดยผู้ใช้งานสามารถแสดงความคิดเห็นและความคืบหน้าเกี่ยวกับปัญหานั้น ๆ ได้ รูปที่ ข-9
11. หน้าแปลน รูปที่ ข-10 ผู้ใช้งานสามารถดูแปลนที่มีอยู่ในโครงการได้ โดยสามารถกดที่รูปภาพเพื่อขยาย ดูรูปที่ใหญ่ขึ้นได้



รูป ข-9 รูปหน้าปัญหา



รูป ข-10 รูปหน้าแปลน

12. หน้าอีเมล รูปที่ ข-11 ผู้ใช้งานสามารถส่งอีเมลถึงผู้ที่ต้องการส่งถึง รวมถึงใส่เนื้อหาลงในช่องที่ระบุ จากนั้นกดปุ่มส่งอีเมล เพื่อส่งอีเมล
13. หน้าแก้ไข Foreman รูปที่ ข-12 ผู้ใช้งานสามารถแก้ไข Foreman ที่เกี่ยวข้องในแต่ละงานได้ โดยสามารถเพิ่ม Foreman โดยการกดปุ่มบวกสีเทา หรือลบ Foreman โดยการปัดซ้ายและกดปุ่มลบสีแดง



iPhone 8 — 13.3

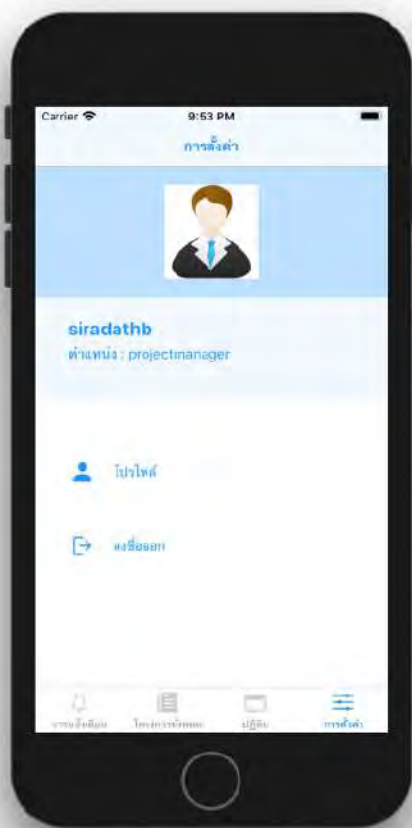


iPhone 8 — 13.3

รูป ข-11 รูปหน้าอีเมล

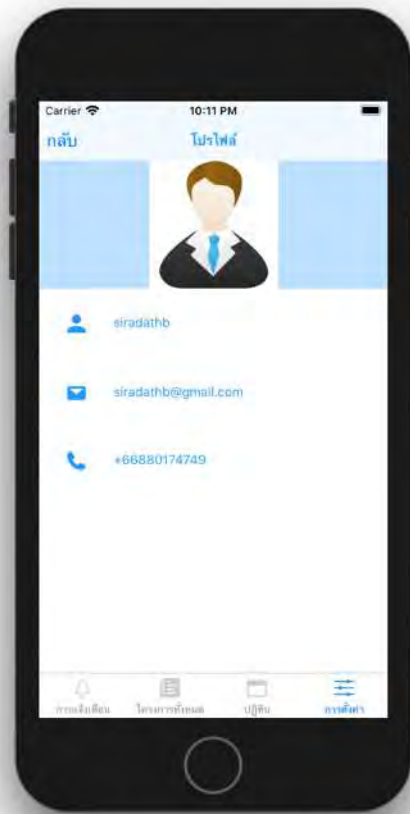
รูป ข-12 รูปหน้าแก้ไข Foreman

14. หน้าการตั้งค่า รูปที่ ข-13 ผู้ใช้งานจะเห็นรูปโปรไฟล์ของผู้ใช้งาน ชื่อผู้ใช้งานตำแหน่ง ปุ่มโปรไฟล์ และปุ่มออกจากระบบ ซึ่งเมื่อกดปุ่มโปรไฟล์จะนำไปสู่หน้าโปรไฟล์ และเมื่อกดปุ่มออกจากระบบก็จะทำการออกจากระบบและกลับไปสู่หน้าเข้าสู่ระบบ
15. หน้าโปรไฟล์ รูปที่ ข-14 จะแสดงข้อมูลของผู้ใช้งานได้แก่ รูปโปรไฟล์ ชื่อผู้ใช้งาน อีเมล และเบอร์โทรศัพท์



iPhone 8 — 13.3

รูป ข-13 รูปหน้าการตั้งค่า



iPhone 8 — 13.3

รูป ข-14 รูปหน้าโปรไฟล์

ภาคผนวก ค

เอกสารอ้างอิง

- [1] “อัปเดตพฤติกรรมและข้อมูลผู้ใช้งาน LINE ประเทศไทย”. [ระบบออนไลน์]. Available from: <https://www.twfdigital.com/blog/2018/09/thailandline-stats-and-behaviour/> [10 ตุลาคม 2562].
- [2] “การศึกษาพฤติกรรมการสื่อสารภายในองค์กรธุรกิจอุตสาหกรรม”. [ระบบออนไลน์]. Available from: <https://www.tci-thaijo.org/index.php/rpu/article/download/112383/> [11 ตุลาคม 2562].
- [3] “User Interviews: How, When, and Why to Conduct Them”. [ระบบออนไลน์]. Available from: <https://www.nngroup.com/articles/user-interviews/> [11 ตุลาคม 2562].
- [4] “PlanGrid”. [ระบบออนไลน์]. Available from: <https://www.plangrid.com> [1 พฤศจิกายน 2562].
- [5] “FIELDWIRE”. [ระบบออนไลน์]. Available from: <https://www.fieldwire.com> [1 พฤศจิกายน 2562].
- [6] “Mango Consultant”. [ระบบออนไลน์]. Available from: <https://www.mangoconsultant.com/th/products/optional/application> [1 พฤศจิกายน 2562].
- [7] Christian Mueller-Roterberg. (2018), Handbook of Design Thinking. USA
- [8] Karl Wieggers and Joy Beatty. (2013), Software Requirement. Washington, USA
- [9] Ian Sommerville. (2011), Software Engineering Ninth Edition. USA
- [10] “พื้นฐาน JavaScript” .[ระบบออนไลน์]. Available from: <https://medium.com/open-source-technology/สรุปพื้นฐาน-javascript-ec02f18cfe47> [3 พฤศจิกายน 2562].
- [11] “รู้จักพื้นฐาน React-native” .[ระบบออนไลน์]. Available from: <https://medium.com/jed-ng/react-native-ทำความรู้จัก-และเริ่มต้นสร้าง-project-91788ef6cac3> [3 พฤศจิกายน 2562].
- [12] “Visual studio Code กับการเขียนโค้ด” .[ระบบออนไลน์]. Available from: <https://medium.com/@rennerwin/10-visual-studio-code-extensions-ที่จะช่วยให้การเขียนโค้ดสะดวกยิ่งขึ้น-ac89dd55e453> [3 พฤศจิกายน 2562].
- [13] “Plangrid” .[ระบบออนไลน์]. Available from: <https://www.plangrid.com> [3 พฤศจิกายน 2562].
- [14] “Sitework” .[ระบบออนไลน์]. Available from: <https://play.google.com/store/apps/details?id=th.co.pps.sitewalk&hl=en>[3 พฤศจิกายน 2562].

ประวัติผู้เขียน



นางสาว คีตาภัทร วงษาเนาวิ

วันเกิด: 3 พฤษภาคม พ.ศ. 2541

อีเมล: keetapat.w@student.chula.ac.th

อีเมลสำรอง : keetapat.wongsanao@gmail.com

วุฒิการศึกษา: กำลังศึกษา หลักสูตรวิทยาศาสตรบัณฑิต คณะวิทยาศาสตร์
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย



นางสาวศิริดา ทาบสุวรรณ

วันเกิด: 15 พฤศจิกายน พ.ศ. 2540

อีเมล: sirada.thab@student.chula.ac.th

อีเมลสำรอง: siradathb@gmail.com

วุฒิการศึกษา: กำลังศึกษา หลักสูตรวิทยาศาสตรบัณฑิต คณะวิทยาศาสตร์
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย