



เว็บแอปพลิเคชันแจ้งของหายในมหาวิทยาลัยธรรมศาสตร์

โดย

นายณัชรนกฤตย์ ชาตะมีนา¹
นายเสธียรพงษ์ เกตุวิวัฒน์²

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยธรรมศาสตร์
ปีการศึกษา 2567
ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

เว็บแอปพลิเคชันแจ้งของหายในมหาวิทยาลัยธรรมศาสตร์

โดย

นายณัชรนกฤตย์ ชาตะมีนา¹
นายเสธียรพงษ์ เกตุวิวัฒน์²

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัย
ธรรมศาสตร์
ปีการศึกษา 2567
ลิขสิทธิ์ของมหาวิทยาลัยธรรมศาสตร์

TU Item Finder

BY

**Mr. Nuttanakit Chatameena
Mr. Sathienphong Ketviwat**

**A FINAL-YEAR PROJECT REPORT SUBMITTED IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF BACHELOR OF SCIENCE
COMPUTER SCIENCE
FACULTY OF SCIENCE AND TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2024
COPYRIGHT OF THAMMASAT UNIVERSITY**

มหาวิทยาลัยธรรมศาสตร์
คณะวิทยาศาสตร์และเทคโนโลยี

รายงานโครงการพิเศษ

ของ

นายณัชรนกฤตย์ ชาตะมีนา
นายเสธียรพงษ์ เกตุวัฒน์

เรื่อง

เว็บแอปพลิเคชันแจ้งของหายในมหาวิทยาลัยธรรมศาสตร์

ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาการคอมพิวเตอร์
เมื่อ วันที่ 7 พฤษภาคม พ.ศ. 2567

อาจารย์ที่ปรึกษา

(ผศ.ดร. ทรงศักดิ์ รองวิริยะพาณิช)

กรรมการสอบโครงการพิเศษ

(ผศ.ดร. เด่นดวง ประดับสุวรรณ)

กรรมการสอบโครงการพิเศษ

(ผศ.ดร. อรุณรा สิทธิศักดิ์)

มหาวิทยาลัยธรรมศาสตร์
คณะวิทยาศาสตร์และเทคโนโลยี

รายงานโครงการพิเศษ

ของ

นายณัชรนกฤตย์ ชาตatemina
นายเสฎฐิยรพงษ์ เกตุวิวัฒน์

เรื่อง

เว็บแอปพลิเคชันแจ้งของหายในมหาวิทยาลัยธรรมศาสตร์

ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตาม
หลักสูตร
หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาการคอมพิวเตอร์
เมื่อ วันที่ 7 พฤษภาคม พ.ศ. 2567

อาจารย์ที่ปรึกษา

(ผศ.ดร. ทรงศักดิ์ รองวิริยะพาณิช)

กรรมการสอบโครงการพิเศษ

(ผศ.ดร. เด่นดวง ประดับสุวรรณ)

กรรมการสอบโครงการพิเศษ

(ผศ.ดร. อรจิรา สิทธิศักดิ์)

หัวข้อโครงการพิเศษ

ชื่อผู้เขียน
ชื่อผู้เขียน
ชื่อปริญญา

สาขาวิชา/คณะ/มหาวิทยาลัย

อาจารย์ที่ปรึกษาโครงการพิเศษ
ปีการศึกษา

เว็บแอปพลิเคชันแจ้งของหายภายในมหาวิทยาลัยธรรมศาสตร์ รังสิต
นายณัชอนกฤตย์ ชาตะมีนา
นายเสธียรพงษ์ เกตุวิวัฒน์
วิทยาศาสตรบัณฑิต สาขาวิชา
วิทยาการคอมพิวเตอร์
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี
มหาวิทยาลัยธรรมศาสตร์
ผศ.ดร. ทรงศักดิ์ รองวิริยะพาณิช
2567

บทคัดย่อ

ในปัจจุบันการจัดการของหายหรือพบของหายในมหาวิทยาลัย ธรรมศาสตร์ ยังพบปัญหาต่างๆ เช่น การพบของสูญหายแต่ไม่ทราบสถานที่รับเก็บของสูญหาย ในปัจจุบันไม่มีแพลตฟอร์มที่จัดการของหายครอบคลุมพื้นที่ทั้งหมดของมหาวิทยาลัยธรรมศาสตร์ได้ โครงการ "TU Item Finder" เป็นเว็บแอปพลิเคชันที่ถูกพัฒนาขึ้นเพื่อแก้ปัญหานี้ โดยเน้นให้ผู้ใช้สามารถรายงานสิ่งของที่พบเจอและติดต่อกับผู้ที่พบของสูญหายได้โดยง่าย เพื่อช่วยให้กระบวนการจัดการของหายมีประสิทธิภาพยิ่งขึ้น

โครงการนี้มุ่งเน้นการพัฒนาระบบของหายในมหาวิทยาลัยธรรมศาสตร์ โดยผู้ใช้จะต้องทำการยืนยันตัวตนผ่าน TU APIs Developers ก่อนที่จะเข้าใช้งาน เพื่อเพิ่มความเชื่อถือและความปลอดภัยในการใช้งาน นอกจากนี้เว็บแอปพลิเคชันยังให้ความสะดวกสบายในการโพสต์ของที่หายสามารถปักหมุดสถานที่พบได้ผ่าน LeafletMap API และเป็นตัวกลางในการติดต่อผู้ใช้งานที่พบของสูญหาย เพื่อช่วยลดปัญหาความสับสนและเพิ่มประสิทธิภาพในการค้นหาสิ่งของที่สูญหายอย่างมีประสิทธิภาพในชุมชนมหาวิทยาลัยธรรมศาสตร์ รังสิต นอกจากนี้โครงการนี้ยังเป็นเว็บแอปพลิเคชันที่ให้ผู้ใช้สามารถติดต่อแจ้งพบของสูญหายและติดต่อรับของที่สูญหายได้อย่างสะดวกสบายทำให้กระบวนการค้นหาของหายเป็นไปอย่างรวดเร็ว และมีประสิทธิภาพมากยิ่งขึ้น

คำสำคัญ: ระบบการแจ้งพบของหาย, ระบบค้นหาของหาย, ช่วยแก้ปัญหา
ในการหายของหาย

Thesis Title	TU Item Finder
Author	Mr. Nuttanakit Chatameena
Author	Mr. Sathienphong Ketviwat
Degree	Bachelor of Science
Major	Computer Science
Field/Faculty/University	Faculty of Science and Technology Thammasat University
Project Advisor	Asst.Prof.Dr. Songsakdi Rongviriyapanish
Academic Years	2024

ABSTRACT

Currently, the management of lost and found items in Thammasat University still faces various issues such as finding lost items without knowing where the lost items are kept. Currently, there is no platform that can manage lost items across the entire area of Thammasat University efficiently. The "TU Item Finder" project is a web application developed to solve this problem, focusing on allowing users to easily report found items and contact those who have lost items, thereby improving the efficiency of the lost and found management process.

This project focuses on developing a lost and found system within Thammasat University, where users need to verify their identity through TU APIs Developers before using the service to enhance trust and security. Additionally, the web application provides convenience by allowing users to pinpoint the location where items are found through the LeafletMap API and contact users who have lost items with just a few clicks, reducing confusion and increasing efficiency in searching for lost items effectively within the university community. Furthermore, this project serves as a web application that allows users to conveniently report lost items and receive found items, speeding up the search process and making it more efficient.

Keywords: Lost and found Item report system,Lost and found search system, Solve problems in finding lost items.

กิตติกรรมประกาศ

โครงการพิเศษฉบับนี้ดำเนินการลุล่วงได้ด้วยความกรุณาจากอาจารย์ พศ.ดร.ทรงศักดิ์ รองวิริยะพานิช อาจารย์ที่ปรึกษาโครงการ ที่ได้ให้คำปรึกษา และแก้ไขข้อบกพร่องต่างๆมาโดยตลอดจนโครงการฉบับนี้เสร็จสมบูรณ์ ผู้พัฒนาจึงขอ ขอบพระคุณ เป็นอย่างสูง

สุดท้ายนี้ขอขอบคุณเพื่อนและอาจารย์ที่ปรึกษาที่ให้คำแนะนำให้คำปรึกษา และกำลังใจที่ดีเสมอมา

คณะผู้จัดทำหวังว่าโครงการพิเศษฉบับนี้จะเป็นประโยชน์ต่อผู้ที่ต้องการจะนำไปใช้ต่อยอดหรือนำไปประยุกต์ใช้งานหากมีข้อผิดพลาด ประการใด ผู้จัดทำต้องขออภัยมา ณ ที่นี่ด้วย

นาย ณัชธนกฤตย์ ชาตะมีนา
นาย เสนียรพงษ์ เกตวิวัฒน์

	สารบัญ หน้า
บทคัดย่อ	1
ABSTRACT	2
กิตติกรรมประกาศ	3
บทที่ 1	12
บทนำ	12
1.1 ความเป็นมาและความสำคัญของโครงการ	12
1.2 วัตถุประสงค์	12
1.3 ขอบเขตของโครงการ	13
1.4 ประโยชน์ของโครงการ	13
บทที่ 2 วรรณกรรมและงานวิจัยที่เกี่ยวข้อง	14
2.1 แนวคิดทฤษฎีที่เกี่ยวข้อง	14
2.1.1 Leaflet Map API	14
2.1.1.1 การติดตั้ง Leaflet map	14
2.1.1.2 ฟังก์ชันการทำงานหลักของ Leaflet Map	15
2.1.1.3 การเรียกใช้งาน Leaflet Map	16
2.1.2 iFrame (Google Maps)	17
2.1.2.1 ขั้นตอนการใช้งาน iFrame (Google Maps)	17
2.2 เทคโนโลยีที่นำมาใช้ในการพัฒนาระบบ	20
2.2.1 Figma	20
2.2.2 Visual Studio Code	21
2.2.3 Tailwind CSS	21
2.2.4 Next.js	22
2.2.5 Typescript	22
2.2.6 React	23
2.2.7 Prisma	23

2.2.8 PostgreSQL	24
2.2.9 Leaflet	24
2.2.10 Firebase	25
2.3 แอปพลิเคชันที่เกี่ยวข้อง	25
2.3.1 Vercel	25
2.3.2 TU Library Lost & found	26
บทที่ 3 วิธีการวิจัย	27
3.1 ภาพรวมของโครงงาน	27
3.2 การวิเคราะห์ขอบเขตและความต้องการของระบบ	28
แผนภาพยุสเคสและรายละเอียดยุสเคส	28
3.2.1 ตารางแสดงความสัมพันธ์ระหว่าง Actor และ Use case	28
3.2.2 คำอธิบาย use case ของระบบ	30
3.2.3 Activity Diagram	39
3.2.4 การออกแบบฐานข้อมูล	47
3.2.5 พจนานุกรมข้อมูล	48
3.2.5.1 ตาราง User	48
3.2.5.2 ตารางผู้ดูแล	48
3.2.5.3 ตารางโพสต์	49
3.3 ประเด็นที่น่าสนใจและสิ่งที่ท้าทาย	50
3.3 ผลลัพธ์ที่คาดหวัง	50
3.4 ระบบใช้งานจริง	51
3.5.1 หน้าหลักของผู้ใช้	51
3.5.2 หน้าแรกของแอปพลิเคชันที่ผู้ใช้สามารถค้นหาของหายที่ถูกพบได้โดยไม่ต้อง Login เข้าใช้งาน	52
3.5.3 หน้าของการเรียกดูข้อมูลที่ต้องการ (โพสต์ที่เป็นสถานะอยู่ในคลัง และยังไม่ login)	53
3.5.4 หน้าแสดงแผนที่ปักหมุดที่พบร่องสูญหาย	53

3.5.5 หน้าแจ้งเตือนผู้ใช้เรื่อง Login	54
3.5.7 หน้า Login	55
3.5.8 หน้าหลักหลังจาก Login	56
3.5.9 หน้าดูรายละเอียดโพสต์	57
3.5.10 หน้าดูรายละเอียดของรายการแจ้งพบรอยalty(สถานะไม่มีอยู่ในคลัง)	58
3.5.11 หน้าดูรายละเอียดของรายการแจ้งพบรอยalty(สถานะถูกรับไปแล้ว)	58
3.5.12 หน้าแจ้งพบรอยalty	59
3.5.13 หน้าโพสต์ของฉัน	62
3.5.14 หน้าแก้ไขโพสต์ (โพสต์ของฉัน)	62
3.5.15 หน้าหลักของผู้ดูแลระบบ	63
3.5.16 หน้าตรวจสอบโพสต์ (สำหรับผู้ดูแล)	63
3.5.17 หน้าແນບຮູບໜັກສູນ(สำหรับผู้ดูแล)	64
3.5.18 หน้าดูรายละเอียดของโพสต์แจ้งพบรอยalty (สถานะ ถูกรับไปแล้ว , เป็นผู้ดูแล)	65
3.5.19 หน้าจัดการโพสต์ (สำหรับผู้ดูแล)	66
3.5.20 หน้าจัดการโพสต์ของฉัน (สำหรับผู้ดูแล)	66
3.5.20 หน้าดูตรวจสอบผู้ใช้ (สำหรับผู้ดูแล)	67
3.5.21 หน้าดูแดชบอร์ด (สำหรับผู้ดูแล)	68
บทที่ 4 ทรัพยากรและแผนการดำเนินงาน	69
4.1 การจัดเตรียมฮาร์ดแวร์และซอฟต์แวร์	69
4.1.1 ฮาร์ดแวร์ที่ใช้ในการพัฒนา	69
4.1.2 ซอฟแวร์ที่ใช้ในการพัฒนา	69
4.1.3 ภาษาที่ใช้ในการพัฒนา (Programing Language)	69
4.1.4 โปรแกรมที่ใช้ในการพัฒนา	69
4.2 แผนการดำเนินงาน	70
4.3 วิธีทดสอบระบบ	72

สรุป	81
5.1 สรุปผลการดำเนินงาน	81
5.2 สิ่งที่ได้เรียนรู้จากการทำโครงการ	82
5.2.1 การพัฒนาระบบเว็บแอปพลิเคชัน	82
5.2.2 การใช้ API และการยืนยันตัวตน	82
5.2.3 การจัดการ Deployment บนแพลตฟอร์ม Vercel	83
5.3 แนวคิดการพัฒนา	83
รายการอ้างอิง	84

สารบัญตาราง

	หน้า
ตารางที่ 3.1 ตารางแสดงความสัมพันธ์ระหว่าง Actor และ Use case	28
ตารางที่ 3.2 ตารางอธิบายuse caseค้นหารายชื่อของหายที่ถูกพบ	30
ตารางที่ 3.3 ตารางคำอธิบายuse caseโพสต์แจ้งพบของหาย	31
ตารางที่ 3.4 ตารางคำuse caseแก้ไขโพสต์แจ้งพบของหาย	32
ตารางที่ 3.5 ตารางคำuse caseแก้ไขโพสต์แจ้งพบของหาย	33
ตารางที่ 3.6 คำอธิบายuse caseตรวจสอบผู้ใช้ในระบบ	34
ตารางที่ 3.7 คำอธิบายuse caseแก้ไขสถานะโพสต์แจ้งพบของหาย	35
ตารางที่ 3.8 คำอธิบายuse caseดูແຜ່ບອ້ດ	36
ตารางที่ 3.9 คำอธิบายuse caseลบโพสต์	37
ตารางที่ 3.10 คำอธิบายuse caseจัดการโพสต์	38
ตารางที่ 3.11 รายละเอียดข้อมูลตาราง user	48
ตารางที่ 3.12 รายละเอียดข้อมูลตาราง ผู้ดูแล	48
ตารางที่ 3.13 รายละเอียดข้อมูลโพสต์	49
ตารางที่ 4.1 ຍາຣດແວຣ໌ທີ່ໃຊ້ໃນການພົມນາ	69
ตารางที่ 4.2 ກາຮດຕະນາການທີ່ຜ່ານມາຈານຄຶງປັຈບັນ (ກາຄກາຮສຶກຫາທີ່ 02/2566)	
ตารางที่ 4.3 ແຜນກາຮດຕະນາການໃນອານັດຕ (ກາຄກາຮສຶກຫາທີ່ 1/2567)	71
ตารางที่ 4.4 ຕາຮາງທດສອບກາຮໃຊ້ງານຄົນຫາຂອງຫາຍ	72
ตารางที่ 4.5 ຕາຮາງກາຮເຂົ້າໃຊ້ງານເຂົ້າສູ່ຮະບບ	73
ตารางที่ 4.6 ຕາຮາງທດສອບກາຮໃຊ້ງານໂພສຕໍຈັງພບຂອງຫາຍ	74
ตารางที่ 4.7 ຕາຮາງທດສອບກາຮໃຊ້ງານແກ້ໄຂໂພສຕໍຈັງພບຂອງຫາຍ	75
ตารางที่ 4.8 ຕາຮາງທດສອບກາຮໃຊ້ງານຕວລະສອບຜູ້ໃຊ້ຮະບບ	76
ตารางที่ 4.9 ຕາຮາງທດສອບກາຮໃຊ້ງານແກ້ໄຂສະຖານະ	77
ตารางที่ 4.10 ຕາຮາງທດສອບກາຮໃຊ້ງານດູແຜ່ບອ້ດ	78
ตารางที่ 4.11 ຕາຮາງທດສອບກາຮໃຊ້ງານລົບໂພສຕໍ	79
ตารางที่ 4.12 ຕາຮາງທດສອບກາຮໃຊ້ງານຈັດກາຮໂພສຕໍ	80

สารบัญภาพ

	หน้า
ภาพที่ 2.1 ภาพการติดตั้งไลบรารี Leaflet บน terminal	14
ภาพที่ 2.2 ภาพการติดตั้งไลบรารี Leaflet บน terminal อีกครึ่งหนึ่ง	14
ภาพที่ 2.3 ตัวอย่างการเรียกใช้งาน Leaflet Map	16
ภาพที่ 2.4 ภาพขั้นตอนการใช้งาน iFrame ที่ 1	17
ภาพที่ 2.5 ภาพขั้นตอนการใช้งาน iFrame ที่ 2	18
ภาพที่ 2.6 ภาพขั้นตอนการใช้งาน iFrame ที่ 3	19
ภาพที่ 2.7 ภาพตัวอย่างการใช้งานโค้ด iFrame(Google Map)	20
ภาพที่ 2.8 โลโก้Figma	20
ภาพที่ 2.9 โลโก้Visual Studio	21
ภาพที่ 2.10 Tailwind CSS	21
ภาพที่ 2.11 โลโก้Next.js	22
ภาพที่ 2.12 โลโก้TypeScript	22
ภาพที่ 2.13 โลโก้React	23
ภาพที่ 2.14 โลโก้Prisma	23
ภาพที่ 2.15 โลโก้PostgreSQL	24
ภาพที่ 2.16 โลโก้ Leaflet Map	24
ภาพที่ 2.17 โลโก้Firebase	25
ภาพที่ 2.18 โลโก้Vercel	25
ภาพที่ 2.19 โลโก้ TU Library LOST and FOUND	26
ภาพที่ 3.1 สถาปัตยกรรมของระบบTU ItemFinder	27
ภาพที่ 3.2 ยูสเคสของระบบ (Use case diagram)	28
ภาพที่ 3.3 Activity diagram ค้นหารายชื่อของหายที่ถูกพบ	39

ภาพที่ 3.4 Activity diagram โพสต์ของหายที่ถูกพบ	40
ภาพที่ 3.5 Activity diagram แก้ไขโพสต์แจ้งพบของหาย	41
ภาพที่ 3.6 ภาพ Activity diagram ตรวจสอบผู้ใช้งานระบบ	42
ภาพที่ 3.7 ภาพ Activity diagram แก้ไขสถานะโพสต์แจ้งพบของหาย	43
ภาพที่ 3.8 ภาพ Activity diagram ดูเดชบอร์ด	44
ภาพที่ 3.9 ภาพ Activity diagram ตรวจสอบผู้ใช้งาน	45
ภาพที่ 3.10 ภาพ Activity diagram จัดการโพสต์	46
ภาพที่ 3.11 แผนภาพ ER Diagram	47
ภาพที่ 3.12 ตัวอย่างหน้าแรกของแอปพลิเคชัน	51
ภาพที่ 3.13 ตัวอย่างหน้าแรกของแอปพลิเคชันที่ผู้ใช้สามารถค้นหาของหายที่ถูกพบได้โดยไม่ต้อง Login เข้าใช้งาน	52
ภาพที่ 3.14 ตัวอย่างหน้าของการเรียกดูข้อมูลที่ต้องการ (โพสต์ที่เป็นสถานะถูกรับไปแล้ว และยังไม่ login)	53
ภาพที่ 3.15 ตัวอย่างหน้าแสดงแผนที่ปักหมุดที่พบของสูญหาย	53
ภาพที่ 3.16 ตัวอย่างหน้าแจ้งผู้ใช้เรื่อง login	54
ภาพที่ 3.17 ตัวอย่างหน้าแสดงแผนที่เมื่อผู้ใช้ทำการดูหมุดสถานที่เก็บของหาย	54
ภาพที่ 3.18 ตัวอย่างหน้า Login ของแอปพลิเคชัน	55
ภาพที่ 3.19 ตัวอย่างหน้า Login ของแอปพลิเคชันหน้าหลักหลังจาก login	56
ภาพที่ 3.20 ตัวอย่างหน้าดูรายละเอียดโพสต์ของรายการแจ้งพบของหาย(สถานะอยู่ในคลัง)	57
ภาพที่ 3.21 ตัวอย่างหน้าดูรายละเอียดของรายการแจ้งพบของหาย(สถานะไม่อยู่ในคลัง)	58
ภาพที่ 3.22 ตัวอย่างหน้าดูรายละเอียดของรายการแจ้งพบของหาย(สถานะถูกรับไปแล้ว)	58
ภาพที่ 3.23 ตัวอย่างหน้าการแจ้งพบของหาย	59

ภาพที่ 3.24 ตัวอย่างหน้าการแจ้งพบของหายแบบกรอกรายละเอียดครบหลือแนบรูป	59
ภาพที่ 3.25 หลังแนบรูป จะขึ้นหน้าให้เบลอรูปภาพ	60
ภาพที่ 3.26 หน้าตัวอย่างหลังจากการเบลอ	60
ภาพที่ 3.27 หน้าตัวอย่างหลังจากกดปุ่มโพสต์ไป	61
ภาพที่ 3.28 หน้าตัวอย่างหลังจากการเบลอ	61
ภาพที่ 3.29 ตัวอย่างหน้าดูโพสต์ของฉัน	62
ภาพที่ 3.30 ตัวอย่างหน้าแก้ไขโพสต์ของฉัน	62
ภาพที่ 3.31 ตัวอย่างหน้าแรกของผู้ดูแลระบบ	63
ภาพที่ 3.32 ตัวอย่างหน้าตรวจสอบโพสต์ (สำหรับผู้ดูแล)	63
ภาพที่ 3.33 ตัวอย่างหน้าหน้าแนบรูปหลักฐาน (สำหรับผู้ดูแล)	64
ภาพที่ 3.34 ดูรายละเอียดของโพสต์แจ้งพบของหาย (สถานะ ถูกรับไปแล้ว , เป็นผู้ดูแล)	65
ภาพที่ 3.35 หลังจากการดูรูปหลักฐาน	65
ภาพที่ 3.36 หน้าจัดการโพสต์ (สำหรับผู้ดูแล)	66
ภาพที่ 3.37 หน้าดูโพสต์ของฉัน (สำหรับผู้ดูแล)	66
ภาพที่ 3.38 หน้าดูตรวจสอบผู้ใช้ (สำหรับผู้ดูแล)	67
ภาพที่ 3.39 หน้าดูตรวจสอบผู้ใช้หลังกดปุ่มจำนวนโพสต์ (สำหรับผู้ดูแล)	67
ภาพที่ 3.40 หน้าดูตรวจสอบผู้ใช้ (สำหรับผู้ดูแล)	68

บทที่ 1 บทนำ

1.1 ความเป็นมาและความสำคัญของโครงการ

จำนวนคนเข้าออกในมหาวิทยาลัยในแต่ละวันนั้น มีจำนวนมาก ไม่ว่า จะเข้ามาเพื่อเรียนรายวิชาต่างๆ หรือ เข้ามาทำธุระส่วนตัวหรือซึมกิจกรรม ต่างๆ โดยบางครั้งอาจจะเป็นการเข้ามาในพื้นที่มหาวิทยาลัยที่ใช้เวลาไม่นาน มีวัตถุประสงค์ที่เร่งด่วนจนทำให้อาจจะลืมสิ่งของได้ และ มีปัญหาไม่ทราบว่าจะติดต่อสอบถามความเพื่อหายของที่ลืมได้อย่างไร

ด้วยเหตุนี้ที่มุ่งผู้พัฒนาโครงการพิเศษจึงพัฒนาเว็บแอปพลิเคชัน หายของหาย เพื่อที่จะแก้ปัญหา ดังนี้

- 1) แก้ปัญหาคนที่ทำของหายไม่รู้ว่าจะต้องหาได้ที่ไหน
- 2) เพื่อช่วยในการติดตามหายของหายได้ง่ายขึ้น
- 3) เพื่อให้ผู้ใช้สามารถติดต่อแจ้งของสูญหายและรับของที่สูญหายคืน

1.2 วัตถุประสงค์

- 1) เพื่อศึกษาและออกแบบการพัฒนาระบบแพลตฟอร์มเว็บแอปพลิเคชันแจ้งของหายภายในมหาวิทยาลัยธรรมศาสตร์
- 2) เพื่อให้เกิดความสะดวกในการค้นหาของที่สูญหายภายในมหาวิทยาลัยธรรมศาสตร์
- 3) เพื่อเพิ่มช่องทางให้ผู้ใช้ติดต่อแจ้งของสูญหายแจ้งพบของและรับของที่สูญหายคืนได้

1.3 ขอบเขตของโครงการ

1) เว็บแอปพลิเคชันนี้จะครอบคลุมเฉพาะภายในมหาวิทยาลัย
ธรรมศาสตร์ศูนย์รังสิต และ ครอบคลุม use cases ดังนี้

- 1) แจ้งการพบของหาย
- 2) ค้นหารายการของหายที่มีผู้แจ้งพบ
- 3) แก้ไขโพสต์แจ้งของหายที่ถูกลบ
- 4) ตรวจสอบโพสต์ของผู้ใช้

2) ผู้ที่พบร่องจะต้องเป็นบุคลากรของมหาวิทยาลัยธรรมศาสตร์ที่สามารถยืนยันตัวบุคคลได้ผ่านระบบ TU APIs Developers ของมหาวิทยาลัยธรรมศาสตร์

1.4 ประโยชน์ของโครงการ

หลังการดำเนินโครงการตามวัตถุประสงค์ ภายใต้ขอบเขตโครงการที่ได้กำหนดไว้แล้วผู้จัดทำคาดหวังคาดหวังประโยชน์ของโครงการ ดังนี้

- 1) เป็นเว็บแอปพลิเคชันช่วยอำนวยความสะดวกในการตามหาสิ่งของที่หายและแจ้งสิ่งของที่หาย
- 2) เป็นแพลตฟอร์มตัวกลางในการค้นหาสิ่งของที่สูญหาย และช่วยให้ผู้ใช้ที่ทำของสูญหายได้รับสิ่งของคืน

บทที่ 2 วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

2.1 แนวคิดทฤษฎีที่เกี่ยวข้อง

2.1.1 Leaflet Map API

เป็นไลบรารีที่ใช้ในการสร้างแผนที่แบบอินเทอร์แอคทีฟบนเว็บไซต์ ซึ่งทำงานได้ดีบนเบราว์เซอร์ส่วนใหญ่และรองรับการใช้งานบนอุปกรณ์มือถือด้วย

2.1.1.1 การติดตั้ง Leaflet map

```
npm install leaflet
```

ภาพที่ 2.1 ภาพการติดตั้งไลบรารี Leaflet บน terminal

```
yarn add leaflet
```

ภาพที่ 2.2 ภาพการติดตั้งไลบรารี Leaflet บน terminal อีกวิธีหนึ่ง

2.1.1.2 ฟังก์ชันการทำงานหลักของ Leaflet Map

ฟังก์ชันการทำงานหลักของ Leaflet เพื่อสร้างและจัดการแผนที่แบบ Interactive ในเว็บแอปพลิเคชัน โดยมีฟังก์ชันการทำงานหลักของคอมโพเนนต์ประกอบด้วย 3 ฟังก์ชันหลักดังนี้:

1) การสร้างแผนที่

เมื่อโหลดคอมโพเนนต์ LeafletMap ครั้งแรก จะมีการตรวจสอบว่าแผนที่ได้ถูกสร้างขึ้นมาก่อนหรือไม่ หากยังไม่มีการสร้างแผนที่ระบบจะสร้างแผนที่โดยใช้ไลบรารี Leaflet ที่ระบุพิกัดเริ่มต้นและระดับการซูมจากข้อมูลที่ส่งผ่าน props `posix` และ `zoom` โดยโค้ดนี้ใช้ฟังก์ชัน `L.map` เพื่อสร้าง instance ของแผนที่ใหม่ และใช้ `L.tileLayer` สำหรับการดึงข้อมูลแผนที่พื้นฐานจาก OpenStreetMap (OSM) เพื่อแสดงเป็นพื้นหลังของแผนที่

2) การเพิ่มマーคเกอร์หรือหมุดในแผนที่แบบกำหนดเอง

ผู้ใช้สามารถกำหนดマーคเกอร์ (Marker) ซึ่งจะทำให้ผู้ใช้สามารถระบุจุดใดๆ บนแผนที่ เมื่อผู้ใช้คลิกบนตำแหน่งนั้นๆ ของแผนที่ โดยฟังก์ชันการเพิ่มマーคเกอร์ทำงานผ่านฟังก์ชัน “`onMapClick`” ที่รับพิกัดของตำแหน่งที่คลิก และเพิ่มマーคเกอร์หรือปรับตำแหน่งของマーคเกอร์เดิมไปยังตำแหน่งใหม่ หากมีマーคเกอร์อยู่แล้ว

3) ปุ่มระบุตำแหน่งผู้ใช้

โค้ดนี้มีปุ่ม "Mark My Position" ซึ่งเมื่อต้องการทราบตำแหน่งปัจจุบันของผู้ใช้ ผู้ใช้สามารถกดปุ่มนี้เพื่อให้แผนที่ทำการเลื่อนไปยังตำแหน่งของผู้ใช้ ปุ่มนี้เรียกว่าฟังก์ชัน “`handleMarkMyPosition`” ที่ดึงพิกัดตำแหน่งปัจจุบันจาก GPS ของอุปกรณ์ผ่าน API “`navigator.geolocation.getCurrentPosition`” ซึ่งหากอุปกรณ์สามารถระบุตำแหน่งได้ ฟังก์ชันนี้จะอัปเดตพิกัดและแผนที่ให้ไปยังตำแหน่งปัจจุบันของผู้ใช้พร้อมเลื่อนマーคเกอร์ไปที่ตำแหน่งนั้นๆ

2.1.1.3 การเรียกใช้งาน Leaflet Map

การเรียกใช้งานคอมโพเนนต์ LeafletMap มีวัตถุประสงค์ เพื่อแสดงแผนที่ที่สามารถโต้ตอบได้ (Interactive Map) ซึ่งผู้ใช้สามารถเลื่อนตำแหน่งแผนที่ เพิ่มmarker (Marker) ในตำแหน่งที่คลิก หรือระบุตำแหน่งปัจจุบันของผู้ใช้ได้ คอมโพเนนต์นี้ได้รับการออกแบบให้มีความยืดหยุ่นในการปรับแต่ง เช่น พิกัดเริ่มต้นของแผนที่ ระดับการขยาย และการตอบสนองต่อการคลิกบนแผนที่หรือการระบุตำแหน่งผู้ใช้

โดยมีตัวอย่างการเรียกใช้งานดังนี้

```
<Map
  posix={[lat, long]} // พิกัดเริ่มต้นของแผนที่ (latitude, longitude)
  zoom={13} // ระดับการขยายเริ่มต้นของแผนที่
  onMapClick={handleMapClick} // ฟังก์ชันที่เรียกเมื่อมีการคลิกบนแผนที่
  onLocationUpdate={handleMapClick} // ฟังก์ชันที่เรียกเมื่อระบุตำแหน่งผู้ใช้
  style={{ height: "200px", width: "100%" }} // กำหนดความสูงและความกว้างของแผนที่
/>
```

ภาพที่ 2.3 ตัวอย่างการเรียกใช้งาน Leaflet Map

2.1.2 iFrame (Google Maps)

iFrame ของ Google Maps สามารถใช้งานได้ฟรี สำหรับการฝังแผนที่ทั่วไป เช่น การแสดงตำแหน่งหรือสถานที่ โดยไม่ต้องใช้ API Key หรือเสียค่าบริการใดๆ

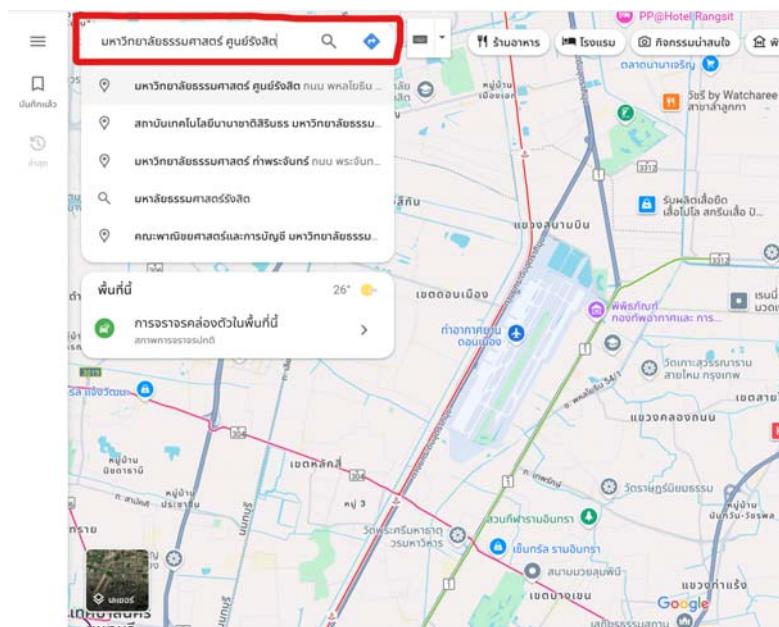
2.1.2.1 ขั้นตอนการใช้งาน iFrame (Google Maps)

(1) ค้นหาสถานที่บน Google Maps

1) เปิด Google Maps

2) ใช้แบบค้นหาเพื่อป้อนชื่อสถานที่ที่ต้องการ เช่น "มหาวิทยาลัยธรรมศาสตร์ ศูนย์รังสิต"

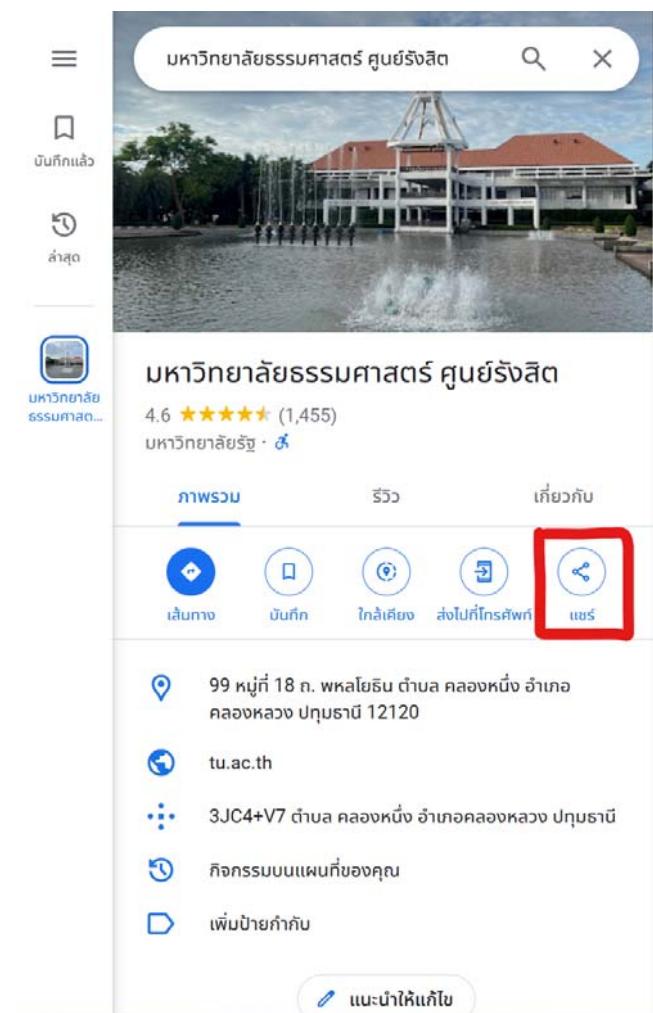
3) กดปุ่มค้นหาเพื่อแสดงตำแหน่งบนแผนที่



ภาพที่ 2.4 ภาพขั้นตอนการใช้งาน iFrame ที่ 1

(2) คลิกปุ่ม "แชร์" เพื่อเข้าถึงตัวเลือกการฝังแผนที่

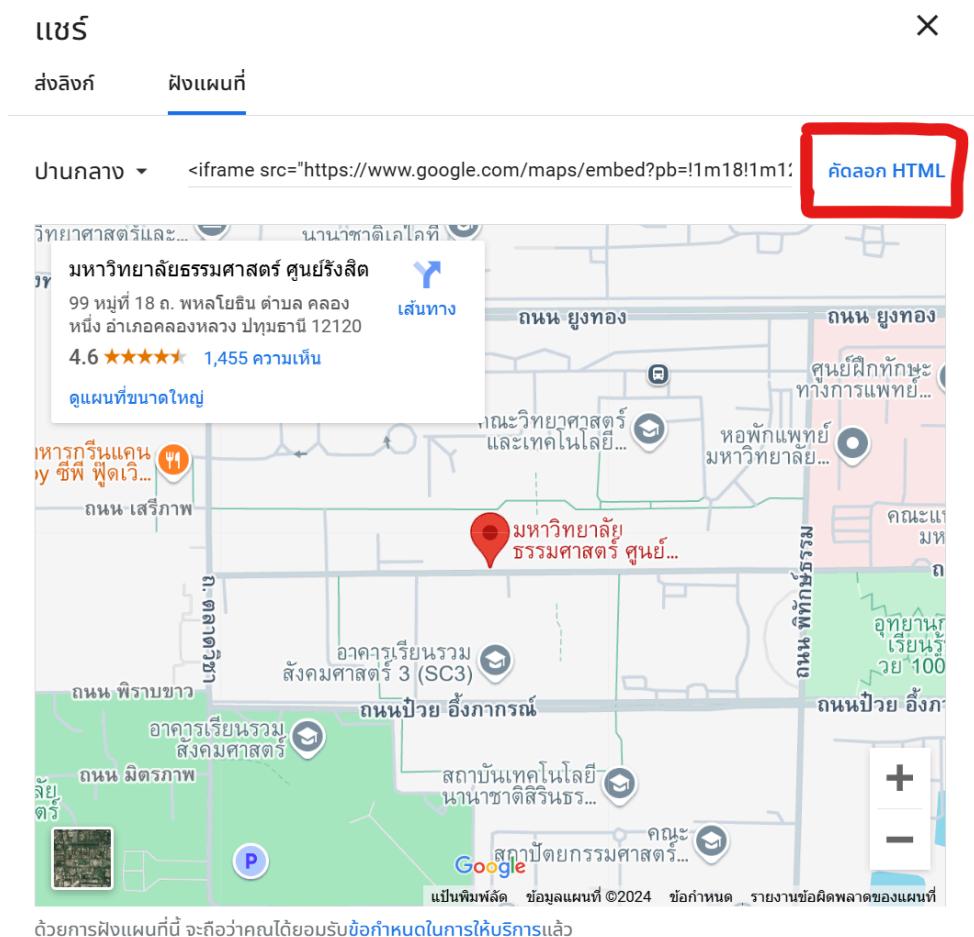
- 1) หลังจากแสดงตำแหน่งของสถานที่ที่ต้องการบนแผนที่ ให้คลิกที่ปุ่ม "แชร์" ซึ่งอยู่ด้านล่างชื่อสถานที่
- 2) หน้าต่างใหม่จะปรากฏขึ้นเพื่อแสดงตัวเลือกในการแชร์



ภาพที่ 2.5 ภาพขั้นตอนการใช้งาน iFrame ที่ 2

(3) คัดลอกโค้ด iFrame สำหรับการฝังแผนที่

- 1) ในหน้าต่างแชร์ ให้เลือกแท็บ "ฝังแผนที่"
- 2) จะปรากฏโค้ด HTML ที่สามารถนำไปใช้งานได้
- 3) คลิกปุ่ม "คัดลอก HTML" เพื่อคัดลอกโค้ด สำหรับนำไปวางในหน้าเว็บไซต์



ภาพที่ 2.6 ภาพขั้นตอนการใช้งาน iFrame ที่ 3

ตัวอย่างการใช้งานโค้ด iFrame ของ Google Map

```
<iframe
  src="https://www.google.com/maps/embed?pb=!1m18!
  style={{ border: 0 }}
  allowFullScreen={true}
  loading="lazy"
  referrerPolicy="no-referrer-when-downgrade"
  className="h-full w-full"
></iframe>
```

ภาพที่ 2.7 ภาพตัวอย่างการใช้งานโค้ด iFrame(Google Map)

2.2 เทคโนโลยีที่นำมาใช้ในการพัฒนาระบบ

2.2.1 Figma

เป็นแพลตฟอร์มการออกแบบ User interface, ดีไซน์กราฟิก, และโปรดิไกป์อื่น ๆ สำหรับเว็บไซต์

Figma สามารถออกแบบหน้า User Interface รวมถึงสี รูปแบบตัวอักษร และสีสันต่าง ๆ ที่เป็นส่วนหนึ่งของแอปพลิเคชันและสร้าง Prototype ของแอปพลิเคชัน ที่ช่วยให้ผู้ใช้สามารถทดสอบและตรวจสอบประสิทธิภาพของแอปพลิเคชันได้



ภาพที่ 2.8 โลโก้ Figma

2.2.2 Visual Studio Code

Visual Studio Code (VS Code) เป็น(Open Source) ออกแบบมาเพื่อให้นักพัฒนาสามารถเขียนโค้ดได้ง่ายและมีประสิทธิภาพ โดยรองรับหลายภาษาโปรแกรม เช่น JavaScript, TypeScript, Python, Java, HTML, CSS, และอื่น ๆ โดย

Visual Studio Code เป็นIDEที่ดีที่สุดในการพัฒนาเว็บไซต์ในปัจจุบัน มีความสามารถในการขยายเสียงและรองรับภาษาโปรแกรมได้หลายประเภท



ภาพที่ 2.9 โลโก้Visual Studio

2.2.3 Tailwind CSS

เป็นเฟรมเวิร์กส్టైล์แบบ utility-first ที่ใช้สร้างเว็บไซต์และแอปพลิเคชัน. มีความเฉพาะเจาะจงในการให้คุณสร้างส్టైล์ของเว็บไซต์ของคุณได้อย่างรวดเร็วและสะดวก

Tailwind CSS มีแนวคิดต่างจาก Framework อื่น ๆ โดยมุ่งเน้นให้เพิ่มคลาสใน HTML เพื่อสร้างส్టైล์ มีความยืดหยุ่นสูงและสามารถปรับแต่งได้อย่างอิสระ โดยเราจะใช้ Tailwind CSS เป็นส่วนเสริมไว้ใช้ใน Next.js



ภาพที่ 2.10 Tailwind CSS

2.2.4 Next.js

Next.js คือ React Framework สำหรับการทำ Full Stack แปลว่าสามารถทำได้ทั้ง Frontend และ Backend ได้ใน Framework ตัวเดียว ทำให้การใช้งานง่าย สะดวกและรวดเร็วขึ้น โดยฟีเจอร์ที่ทำให้ Next.js

Next.js จะมีระบบ SSR (Server-side rendering) ที่ช่วยลดเวลาโหลดหน้าเว็บไซต์ของเรา และเพิ่มประสิทธิภาพในการทำ SEO (Search Engine Optimization) โดยทำให้เว็บไซต์สามารถรับข้อมูลจากเซิร์ฟเวอร์ก่อนที่จะถูกส่งให้กับผู้ใช้



ภาพที่ 2.11 โลโก้Next.js

2.2.5 Typescript

เป็นภาษาโปรแกรมที่ถูกพัฒนาขึ้นมาบนพื้นฐานของ JavaScript และเสริมเติมความสามารถในการพัฒนาโปรแกรมด้วยการเพิ่มระบบประสิทธิภาพทางชนิด (Type System) ที่ช่วยในการจัดการข้อผิดพลาดที่อาจเกิดขึ้นในระหว่างรันโปรแกรม

TypeScript มีระบบ Type system ที่ช่วยในการตรวจสอบข้อผิดพลาดในขั้นตอนของการเขียนโค้ดซึ่งทำให้นักพัฒนาสามารถกำหนดประเภทข้อมูลให้กับตัวแปร, พารามิเตอร์, และการส่งค่าผ่านฟังก์ชัน เพื่อลดข้อผิดพลาดที่เกิดจากการใช้งานข้อมูลอย่างไม่ถูกต้องซึ่งเราจะใช้เป็นตัวเสริมใน Next.js

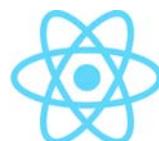


ภาพที่ 2.12 โลโก้TypeScript

2.2.6 React

React เป็นไลบรารี JavaScript ที่ถูกพัฒนาโดย Facebook และให้บริการสำหรับการสร้าง User Interface (UI) ของเว็บแอปพลิเคชัน. React นับเป็นไลบรารีที่ได้รับความนิยมมากในการพัฒนาเว็บเนื่องจากความสะดวกในการจัดการส่วนต่อประสานผู้ใช้และการจัดการสถานะของแอปพลิเคชัน

การใช้ React ใช้โครงสร้างแบบ Component-Based ทำให้นักพัฒนาสามารถสร้างและรวมส่วนประกอบของเว็บแอปพลิเคชันได้โดยมีความสะดวกและมีความยืดหยุ่นมาก แต่ละส่วนประกอบมีสถาปัตยกรรมของตนเองทำให้ง่ายต่อการบำรุงรักษาและทดสอบ



ภาพที่ 2.13 โลโก้React

2.2.7 Prisma

Prisma ที่เป็น ORM (Object-Relational Mapping) library สำหรับภาษาโปรแกรม JavaScript และ TypeScript โดยนำเทคโนโลยีนี้มาใช้เป็นเครื่องมืออำนวยความสะดวกในการสร้าง schema, query, และ migration กับ database

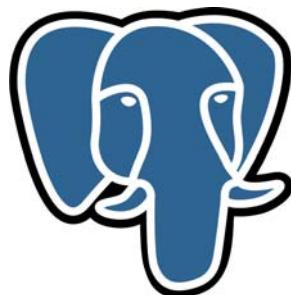
Prisma ใช้Next js ในการพัฒนา โดยสองตัวนี้ช่วยเพิ่มประสิทธิภาพในการจัดการกับข้อมูลให้ดียิ่งขึ้น และมีความสะดวกสบายมากขึ้น



ภาพที่ 2.14 โลโก้Prisma

2.2.8 PostgreSQL

PostgreSQL เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์แบบโอเพนซอร์ส (RDBMS) ที่มีความสามารถสูงและมีฟีเจอร์หลากหลาย ได้รับการพัฒนาและปรับปรุงอย่างต่อเนื่องเพื่อรองรับการใช้งานฐานข้อมูลขนาดใหญ่ และความต้องการที่ซับซ้อนในระดับองค์กร ด้วยเหตุนี้ PostgreSQL จึงได้รับความนิยมในหลาย ๆ สาขา เช่น การเงิน การศึกษา อีคอมเมิร์ซ และการวิเคราะห์ข้อมูล



ภาพที่ 2.15 โลโก้ PostgreSQL

2.2.9 Leaflet

Leaflet เป็นไลบรารีโอเพนซอร์สที่ช่วยในการสร้างและใช้งานแผนที่ในเว็บแอปพลิเคชัน โดยใช้ JavaScript เป็นหลัก ไลบรารี Leaflet มีขนาดเบา ทำงานรวดเร็ว และสามารถใช้งานได้ง่ายสำหรับการแสดงผลแบบโต้ตอบ เช่น การซูม ขยาย การวางแผนマーกเกอร์บนแผนที่ (marker)



ภาพที่ 2.16 โลโก้ Leaflet Map

2.2.10 Firebase

เป็นแพลตฟอร์มพัฒนาแอปพลิเคชันที่ Google พัฒนาและให้บริการ โดยมีเครื่องมือและบริการที่ช่วยให้การพัฒนาแอปพลิเคชันทั้งบนมือถือและเว็บทำได้ง่ายขึ้นและมีประสิทธิภาพมากขึ้น สามารถใช้เก็บรูปภาพ และนำรูปภาพมาใช้ต่อได้ผ่าน ลิงค์ url



ภาพที่ 2.17 โลโก้ Firebase

2.3 แอปพลิเคชันที่เกี่ยวข้อง

2.3.1 Vercel

Vercel คือแพลตฟอร์มสำหรับการพัฒนาและปรับใช้ (deployment) เว็บแอปพลิเคชันที่เน้นการทำงานแบบ **Serverless** และ **Frontend-first** ซึ่งช่วยให้นักพัฒนาเว็บสามารถสร้างเว็บแอปพลิเคชันที่มีประสิทธิภาพสูงได้ง่ายและรวดเร็ว โดยรองรับเทคโนโลยีสมัยใหม่ เช่น **Next.js**, **React**, **Vue.js**, และ **Svelte**



ภาพที่ 2.18 โลโก้ Vercel

2.3.2 TU Library Lost & found

TU Library Lost&found เป็นเว็บแอปพลิเคชันสำหรับแจ้งและหาของหายภายในมหาวิทยาลัยธรรมศาสตร์ ที่ถูกพัฒนาโดย ทีมผู้พัฒนา TU library โดยมีจุดประสงค์ให้นักศึกษาที่ใช้บริการห้องสมุด ให้ได้รับของคืน

ข้อดี

- 1) ผู้ใช้สามารถแจ้งและหาของหายได้ง่ายๆ ผ่านเว็บแอปพลิเคชัน โดยไม่ต้องไปสถานีตำรวจ
- 2) ช่วยเพิ่มโอกาสในการหาผู้ที่ทำของหายให้ได้รับของคืนภายใน มหาวิทยาลัยธรรมศาสตร์

ข้อเสีย

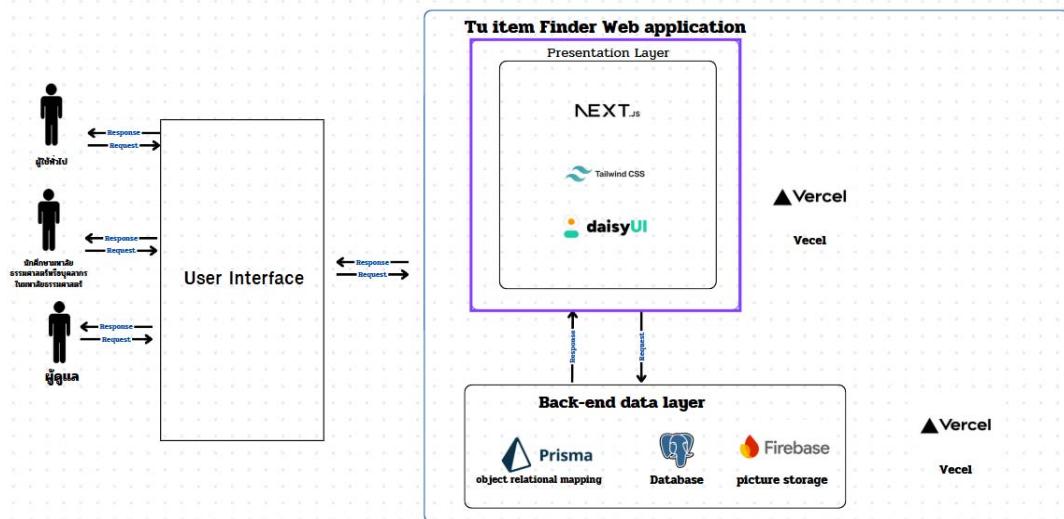
- 1) ไม่สามารถถูรูปหลักฐานคนที่รับของได้
- 2) ไม่สามารถถูรูปภาพของที่สูญหายได้
- 3) ไม่มีประวัติการรับของที่สูญหายให้ผู้ใช้สามารถถูได้



ภาพที่ 2.19 โลโก้ TU Library LOST and FOUND

บทที่ 3 วิธีการวิจัย

3.1 ภาพรวมของโครงการ



ภาพที่ 3.1 สถาปัตยกรรมของระบบ TU ItemFinder

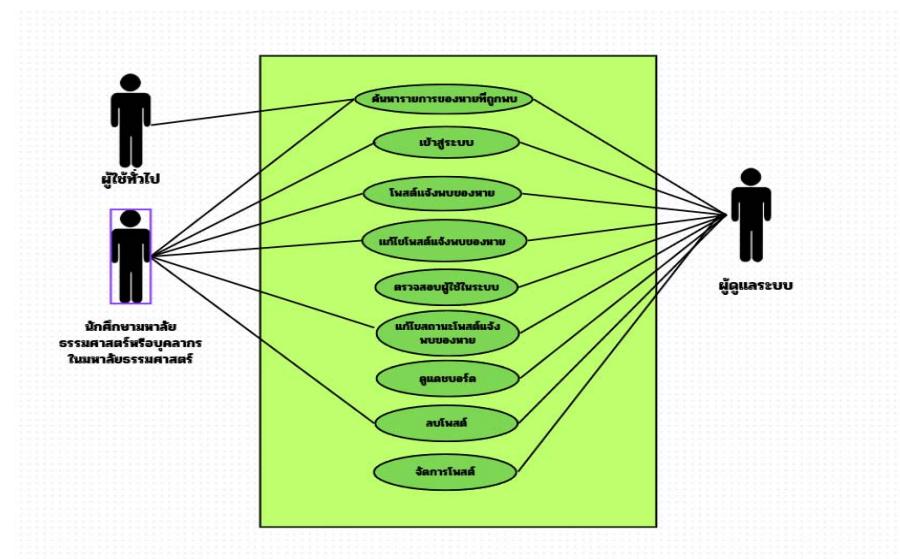
คำอธิบายสถาปัตยกรรมของระบบ

สถาปัตยกรรมของระบบในการพัฒนาแอปพลิเคชันสำหรับเจ้าของสูญหายสามารถแบ่งได้ออกเป็น 3 ส่วน คือ

- 1) **Frontend** ในส่วนของหน้าแอปพลิเคชันพัฒนาโดยใช้ Visual studio ซึ่งเป็น IDE ในการพัฒนาเว็บแอปพลิเคชัน เป็นหน้าต่างที่ User สามารถ interact กับตัวแอพได้
- 2) **Backend** เชื่อมต่อข้อมูลระหว่างเว็บแอปพลิเคชันกับ Database โดยจะใช้ Postgresql ร่วมกับ TU API เพื่อดึงข้อมูลนักศึกษา โดยทำการการดึงข้อมูลจาก TU API และเก็บข้อมูลที่ได้ลงในฐานข้อมูล Postgresql เพื่อนำมาใช้ในการแสดงและประมวลผลในเว็บแอปพลิเคชัน
- 3) **Database** จัดเก็บข้อมูลต่างๆ โดยจะใช้ Postgresql เป็นฐานข้อมูลในการจัดเก็บข้อมูลจากการดึงข้อมูลจาก TU API และมี Firebase ที่จะเอาไว้เก็บรูปภาพสิ่งของที่สูญหายและรูปภาพหลักฐานในการรับของที่สูญหาย

3.2 การวิเคราะห์ขอบเขตและความต้องการของระบบ

แผนภาพยุสเคสและรายละเอียดยุสเคส



ภาพที่ 3.2 ยุสเคสของระบบ (Use case diagram)

3.2.1 ตารางแสดงความสัมพันธ์ระหว่าง Actor และ Use case

Use case ID	Use case name	Description
UC-1	ค้นหารายการของ ้ายที่ถูกพบ	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากร, ผู้ดูแลระบบและผู้ใช้งานทั่วไปที่ต้องการจะหาของหายสามารถที่จะกดค้นหาของหายที่ถูกพบได้โดยการค้นหานั้นจะแบ่งเป็นหมวดอย่างชัดเจนยกตัวอย่างเช่น หมวดกระเพาเงินโทรศัพท์ เป็นต้น
UC-2	เข้าสู่ระบบ	นักศึกษาหรือบุคลากรมาลัยธรรมศาสตร์ รังสิต ที่มีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์ สามารถเข้าสู่ระบบเพื่อแจ้งโพสต์ของหายในเว็บแอปพลิเคชัน TU Item Finder

UC-3	โพสต์แจ้งpubของ hay	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรและผู้ดูแล pub ของที่ ตกไม่มีเจ้าของในบริเวณนั้น สามารถที่จะแจ้งpubของhayได้โดยจะมีปุ่มให้กดแจ้งของที่ถูกpub
UC-4	แก้ไขโพสต์แจ้งpub ของhay	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรและผู้ดูแล สามารถที่ จะแก้ไขโพสต์แจ้งpubของhayได้โดยจะต้องเป็นเจ้าของโพสต์ เท่านั้นที่จะสามารถแก้ไขโพสต์ได้ แต่ผู้ดูแลสามารถแก้ไขทุกโพสต์ได้
UC-5	ตรวจสอบผู้ใช้ในระบบ	ผู้ใช้ที่เป็นผู้ดูแลระบบสามารถที่จะตรวจสอบโพสต์ ของผู้ใช้งานที่เป็นนักศึกษาได้
UC-6	แก้ไขสถานะโพสต์ แจ้งpubของhay	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรและผู้ดูแลระบบ สามารถที่จะสามารถแก้ไขโพสต์แจ้งpubของhayได้ เมื่อมีการเปลี่ยนแปลงของสถานะ ซึ่งจะมีทั้งหมด 3 สถานะด้วยกันคือ 1)สถานะไม่อยู่ในคลัง 2)สถานะอยู่ ในคลัง 3)สถานะรับไปแล้ว แต่ผู้ใช้งานปกติ จะมี 2 สถานะ 1)สถานะไม่อยู่ในคลัง 2)สถานะรับไปแล้ว
UC-7	ดูแดชบอร์ด	ผู้ดูแลสามารถที่จะดูภาพรวมของระบบผ่านหน้าแดชบอร์ดได้
UC-8	ลบโพสต์	ผู้ดูแลระบบหรือนักศึกษาที่เป็นเจ้าของโพสต์ สามารถที่จะลบโพสต์ได้
UC-9	จัดการโพสต์	ผู้ดูแลสามารถที่จะจัดการโพสต์ทุกโพสต์ได้

ตารางที่ 3.1 ตารางแสดงความสัมพันธ์ระหว่าง Actor และ Use case

3.2.2 คำอธิบาย use case ของระบบ

รหัสUse-Case	UC-1
ชื่อUse-Case	ค้นหารายการของหายที่ถูกพบ
ผู้ใช้งาน	ผู้ใช้ทั่วไปและนักศึกษาหรือบุคลากรและผู้ดูแลระบบ
คำอธิบาย	การค้นหารายการของหายที่ถูกพบ ได้โดยการค้นหานั้นจะแบ่งเป็นหมวดอย่างชัดเจน ตัวอย่าง เช่น หมวดกระเบื้องตั้ง โทรศพ เป็นต้น
เงื่อนไขก่อนหน้า	ผู้ใช้ไม่จำเป็นต้องเข้าสู่ระบบ ก็สามารถค้นหารายการของหายได้
เงื่อนไขภายหลัง	ผู้ใช้สามารถค้นหาแล้วเจอสิ่งของที่เลือกค้นหาได้จริง
กระแสหลัก	<p>1. ผู้ใช้ทำการกดเลือกหมวดของหายที่ถูกพบ</p> <p>1.1 พิมพ์คำที่ต้องการค้นหา (ถ้ามี)</p> <p>1.2 เลือกหมวดหมู่ของหายที่ถูกพบ</p> <p>1.3 เลือกสถานที่</p> <p>1.4 เลือกสถานะ</p> <p>2. กดค้นหาของหายที่ถูกพบ</p> <p>3. ระบบแสดง list ของหายที่ถูกพบตามหมวดที่ผู้ใช้เลือกไว้</p>
กระแสรอง	ในขั้นตอนที่ 1.1 ไม่จำเป็นพิมพ์ค้นหาเพิ่ม สามารถเลือก ค้นหาที่เป็นแบบตัวเลือกได้เลย

ตารางที่ 3.2 ตารางอธิบาย use case ค้นหารายชื่อของหายที่ถูกพบ

รหัสUse-Case	UC-2
ชื่อUse-Case	เข้าสู่ระบบ
ผู้ใช้งาน	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรและผู้ดูแลภายในมหาลัยธรรมศาสตร์ รังสิต
คำอธิบาย	การเข้าสู่ระบบผู้ใช้จำเป็นต้องมีบัญชีหรือข้อมูลภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
เงื่อนไขก่อนหน้า	ผู้ใช้ต้องมีบัญชียืนยันตัวบุคคลได้ผ่านระบบ TU APIs Developers ของมหาวิทยาลัยธรรมศาสตร์
เงื่อนไขภายหลัง	ผู้ใช้สามารถเข้าสู่ระบบได้
กระแสหลัก	<ol style="list-style-type: none"> 1. ผู้ใช้กดปุ่ม “เข้าสู่ระบบ” 2. ผู้ใช้กรอก Username(รหัสนักศึกษา) และ Password(รหัสประจำตัวประชาชน) 3. ผู้ใช้เข้าสู่ระบบ
กระแสรอง	-

ตารางที่ 3.3 ตารางคำอธิบายuse caseโพสต์แจ้งพบของหาย

รหัสUse-Case	UC-3
ชื่อUse-Case	โพสต์แจ้งpubของหาย
ผู้ใช้งาน	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรและผู้ดูแลมหาวิทยาลัยธรรมศาสตร์
คำอธิบาย	แจ้งของหายที่ถูกpub สามารถที่จะโพสต์ของหายที่ถูกpubได้โดยจะมีปุ่มให้กดแจ้งของหายที่ถูกpub
เงื่อนไขก่อนหน้า	ผู้ใช้จำเป็นต้องมีบัญชียืนยันตัวบุคคลได้ผ่านระบบ TU APIs ของมหาวิทยาลัยธรรมศาสตร์ ถึงจะสามารถแจ้งของหายที่ถูกpubได้
เงื่อนไขภายหลัง	ผู้ใช้ที่เป็นบุคลากรมหาวิทยาลัยธรรมศาสตร์สามารถโพสต์ได้จริง
กระแสหลัก	<ol style="list-style-type: none"> 1. กดปุ่มแจ้งของหายที่ถูกpub 2. กรอกรายละเอียดต่างๆเกี่ยวกับสิ่งของที่pub <ol style="list-style-type: none"> 2.1 กรอกชื่อสิ่งของที่พบ 2.2 เลือกหมวดหมู่ 2.3. อธิบายรายละเอียดของสิ่งของที่ถูกpub 2.4 เลือกสถานที่ที่พบสิ่งของที่หาย 2.5 แนบรูปได้ (ถ้ามี) <ol style="list-style-type: none"> 2.5.1 สามารถเลือกที่จะเบลอรูปภาพได้ 2.5.2 กดปุ่ม “ตกลง” 2.6 ปักหมุดบนแผนที่ได้ (ถ้ามี) 3. ผู้ใช้กดปุ่ม “โพสต์”
กระแสรอง	<p>ในขั้นตอนที่ 2.4 ถ้าไม่เจอสถานที่ ที่มีให้เลือกสามารถเลือกอื่นๆ และทำการปักหมุดพิกัดสถานที่ที่พบ</p> <p>ในขั้นตอนที่ 2.5 สามารถแนบรูปได้แต่จะไม่สามารถแนบรูปที่บ่งบอกที่ข้อมูลส่วนตัวของคนนั้นได้</p>

ตารางที่ 3.4 ตารางคำอธิบายuse case โพสต์แจ้งpubของหาย

รหัสUse-Case	UC-4
ชื่อUse-Case	แก้ไขโพสต์แจ้งพบของหาย
ผู้ใช้งาน	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรและผู้ดูแลที่เป็นเจ้าของโพสต์
คำอธิบาย	สามารถที่จะกดแก้ไขของหายที่ถูกแจ้งไว้ได้(เจ้าของโพสต์)
เงื่อนไขก่อนหน้า	ผู้ใช้จำเป็นต้องมีบัญชียืนยันตัวบุคคลได้ผ่านระบบ TU APIs Developers ของมหาวิทยาลัยธรรมศาสตร์ ถึงจะสามารถแก้ไขแจ้งของหายที่ถูกพบได้
เงื่อนไขภายหลัง	ผู้ใช้ที่เป็นเจ้าของโพสต์สามารถที่จะแก้โพสต์ของตนเองได้จริง
กระแสหลัก	<ol style="list-style-type: none"> 1. ผู้ใช้กดปุ่ม "แก้ไข" โพสต์ที่ต้องการ 2. ผู้ใช้กรอกรายละเอียดโพสต์ที่ต้องการแก้ไข 3. ผู้ใช้กดยืนยันการแก้ไขโพสต์ 4. ระบบทำการอัพเดทข้อมูลโพสต์ใหม่
กระแสรอง	-

ตารางที่ 3.5 ตารางคำนวณ use case แก้ไขโพสต์แจ้งพบของหาย

รหัส Use-Case	UC-5
ชื่อ Use-Case	ตรวจสอบผู้ใช้ในระบบ
ผู้ใช้งาน	ผู้ดูแลระบบ
คำอธิบาย	ผู้ดูแลสามารถดูข้อมูลและโพสต์ของ User ในระบบได้ว่าแต่ละ User มีโพสต์อะไรบ้าง
เงื่อนไขก่อนหน้า	ผู้ใช้ต้องเป็นผู้ดูแลระบบ
เงื่อนไขภายหลัง	ระบบแสดงผู้ใช้ในระบบ
กระแสหลัก	1. ผู้ดูแลระบบกดปุ่ม “ตรวจสอบผู้ใช้” 2. ระบบแสดงผู้ใช้ระบบ
กระแสรอง	1. ผู้ดูแลกดปุ่ม “[x Post]” ที่ User ที่ต้องการ 2. ระบบแสดงโพสต์ของ User ที่เป็นเจ้าของโพสต์

ตารางที่ 3.6 คำอธิบาย use case ตรวจสอบผู้ใช้ในระบบ

รหัสUse-Case	UC-6
ชื่อUse-Case	แก้ไขสถานะโพสต์แจ้งพบของหาย
ผู้ใช้งาน	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากร,ผู้ดูแลระบบ
คำอธิบาย	นักศึกษาหรือบุคลากรในมหาวิทยาลัย และผู้ดูแลระบบสามารถที่จะแก้ไขสถานะโพสต์แจ้งพบของหายได้ เมื่อมีการเปลี่ยนแปลงของสถานะ ซึ่งจะมีทั้งหมด 3 สถานะด้วยกันคือ 1.สถานะไม่อยู่ในคลัง 2. สถานะอยู่ในคลัง 3. สถานะรับไปแล้ว แต่ผู้ใช้งานปกติ จะมี 2 สถานะ 1.สถานะไม่อยู่ในคลัง 2. สถานะรับไปแล้ว
เงื่อนไขก่อนหน้า	ผู้ใช้ต้องเป็นผู้ดูแลระบบ
เงื่อนไขภายหลัง	ผู้ดูแลระบบสามารถเปลี่ยนแปลงสถานะของโพสต์ได้
กระแสแหลักษ์	1.ผู้ดูแลระบบกดปุ่ม “ตรวจสอบ” ในโพสต์ 2.ในหน้าของโพสต์จะมี 3 สถานะให้เลือกคือ สถานะไม่อยู่ในคลัง 2.สถานะอยู่ในคลัง 3. สถานะรับไปแล้ว 2.1 เลือกสถานะไม่อยู่ในคลัง 2.2 เลือกสถานะอยู่ในคลัง 2.3 เลือกสถานะรับไปแล้ว
กระแสร่อง	-

ตารางที่ 3.7 คำอธิบายuse caseแก้ไขสถานะโพสต์แจ้งพบของหาย

รหัสUse-Case	UC-7
ชื่อUse-Case	ดูแดชบอร์ด
ผู้ใช้งาน	ผู้ดูแลระบบ
คำอธิบาย	ผู้ดูแลสามารถที่จะดูภาพรวมของระบบ ผ่านหน้าแดชบอร์ดได้
เงื่อนไขก่อนหน้า	ผู้ใช้ต้องเป็นผู้ดูแลระบบ
เงื่อนไขภายหลัง	ผู้ดูแลระบบสามารถดูภาพรวมของ ระบบได้
กระแสหลัก	1.ผู้ดูแลระบบกดปุ่ม “แดชบอร์ด” 2.ระบบแสดงภาพรวมข้อมูลทั้งหมดใน ระบบ
กระแสรอง	-

ตารางที่ 3.8 คำอธิบายuse caseดูแดชบอร์ด

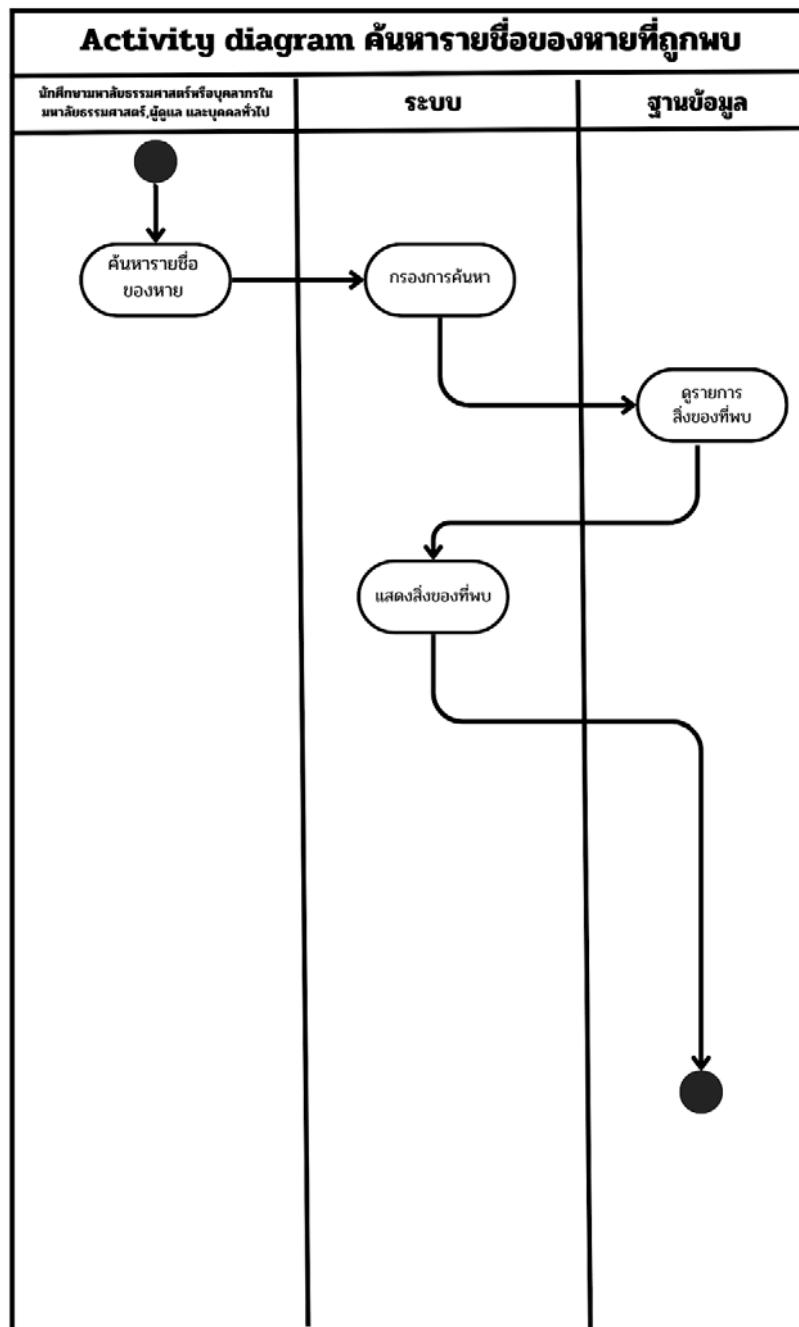
รหัสUse-Case	UC-8
ชื่อUse-Case	ลบโพสต์
ผู้ใช้งาน	นักศึกษาหรือบุคลากรภายในมหาลัย ธรรมศาสตร์ ที่เป็นเจ้าของโพสต์ หรือผู้ ดูแลระบบ สามารถที่จะลบโพสต์ของตัว เองได้
คำอธิบาย	นักศึกษาหรือบุคลากรภายในมหาลัย ธรรมศาสตร์ ที่เป็นเจ้าของโพสต์ หรือผู้ ดูแลระบบ สามารถที่จะลบโพสต์ของตัว เองได้
เงื่อนไขก่อนหน้า	นักศึกษาหรือบุคลากรภายในมหาลัย ธรรมศาสตร์ ต้องเป็นเจ้าของโพสต์นั้น จะสามารถลบโพสต์ได้ หรือเป็นผู้ดูแล ระบบ
เงื่อนไขภายหลัง	นักศึกษาหรือบุคลากรภายในมหาลัย ธรรมศาสตร์ ที่เป็นเจ้าของโพสต์ หรือผู้ ดูแลระบบ สามารถที่จะลบโพสต์ได้
กระแสหลัก	1.นักศึกษาหรือบุคลากรภายในมหาลัย ธรรมศาสตร์ ที่เป็นเจ้าของโพสต์ กดปุ่ม ”โพสต์ของฉัน” 2.ระบบจะแสดงโพสต์ทั้งหมดที่User นั้น เป็นเจ้าของโพสต์ 3.นักศึกษาหรือบุคลากรภายในมหาลัย ธรรมศาสตร์ ที่เป็นเจ้าของโพสต์ ทำการ กดปุ่ม “ลบโพสต์” โพสต์ที่ต้องการ
กระแสรอง	-

ตารางที่ 3.9 คำอธิบายuse caseลบโพสต์

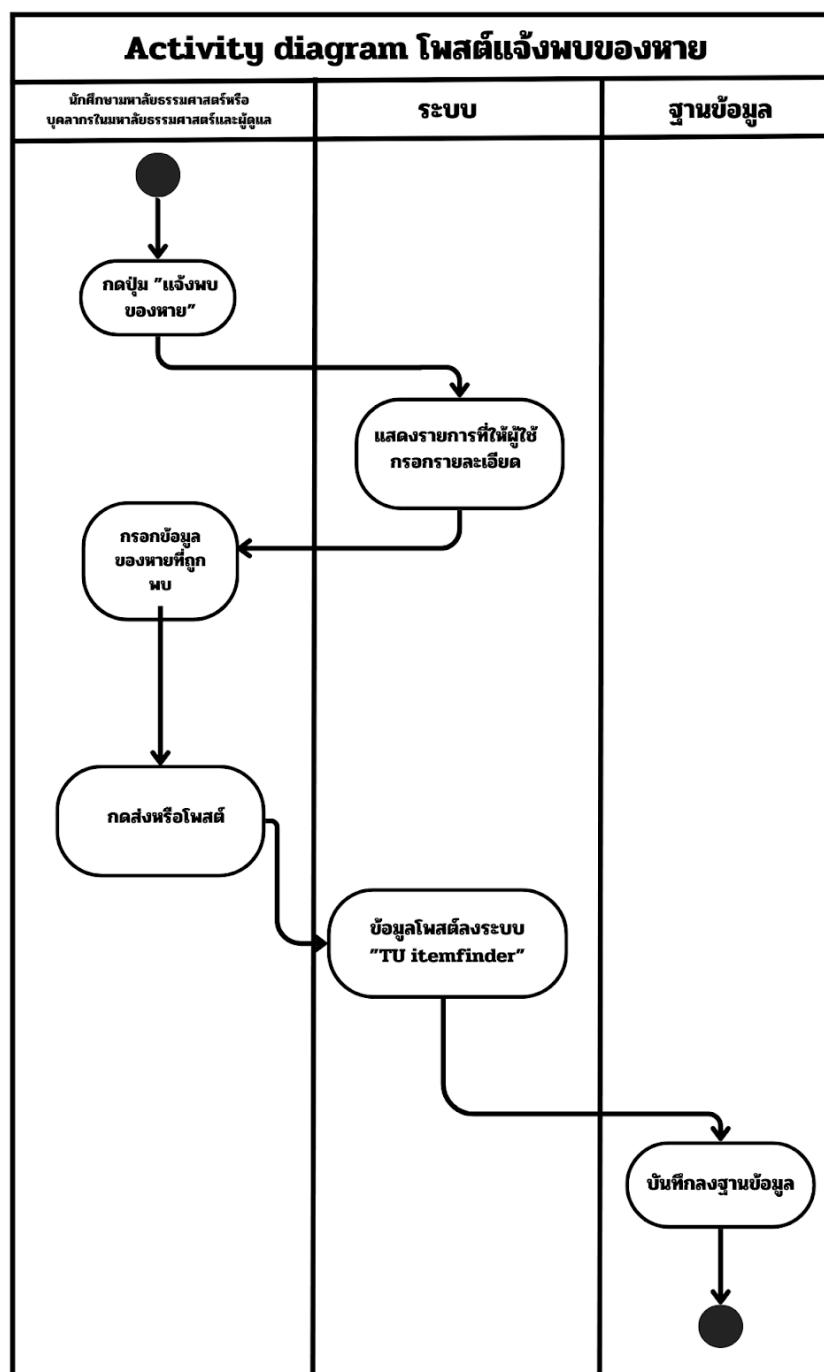
รหัสUse-Case	UC-9
ชื่อUse-Case	จัดการโพสต์
ผู้ใช้งาน	ผู้ดูและระบบ
คำอธิบาย	ผู้ดูและระบบสามารถที่จะจัดการโพสต์ ของผู้ใช้งานทุกคนได้ โดยจะสามารถแก้ไข ลบโพสต์ และดูสถานะหรือดูลายละเอียดโพสต์ทั้งหมดได้
เงื่อนไขก่อนหน้า	ต้องเป็นบัญชีที่เป็นผู้ดูแลถึงจะสามารถเข้าใช้ได้
เงื่อนไขภายหลัง	สามารถจัดการรถแก้ไข ลบโพสต์ และดูสถานะหรือดูลายละเอียดโพสต์ทั้งหมดได้
กระแสหลักษณะ	<ul style="list-style-type: none"> 1. กดปุ่มเข้าใช้งานไปยังหน้าจัดการโพสต์ จาก navbar 2. ระบบจะแสดงโพสต์ทั้งหมด 3. เลือกจัดการตามวัตถุประสงค์ของผู้ดูแล (แก้ไข, ลบโพสต์, ปุ่มสถานะ)
กระแสร่อง	-

ตารางที่ 3.10 คำอธิบายuse caseจัดการโพสต์

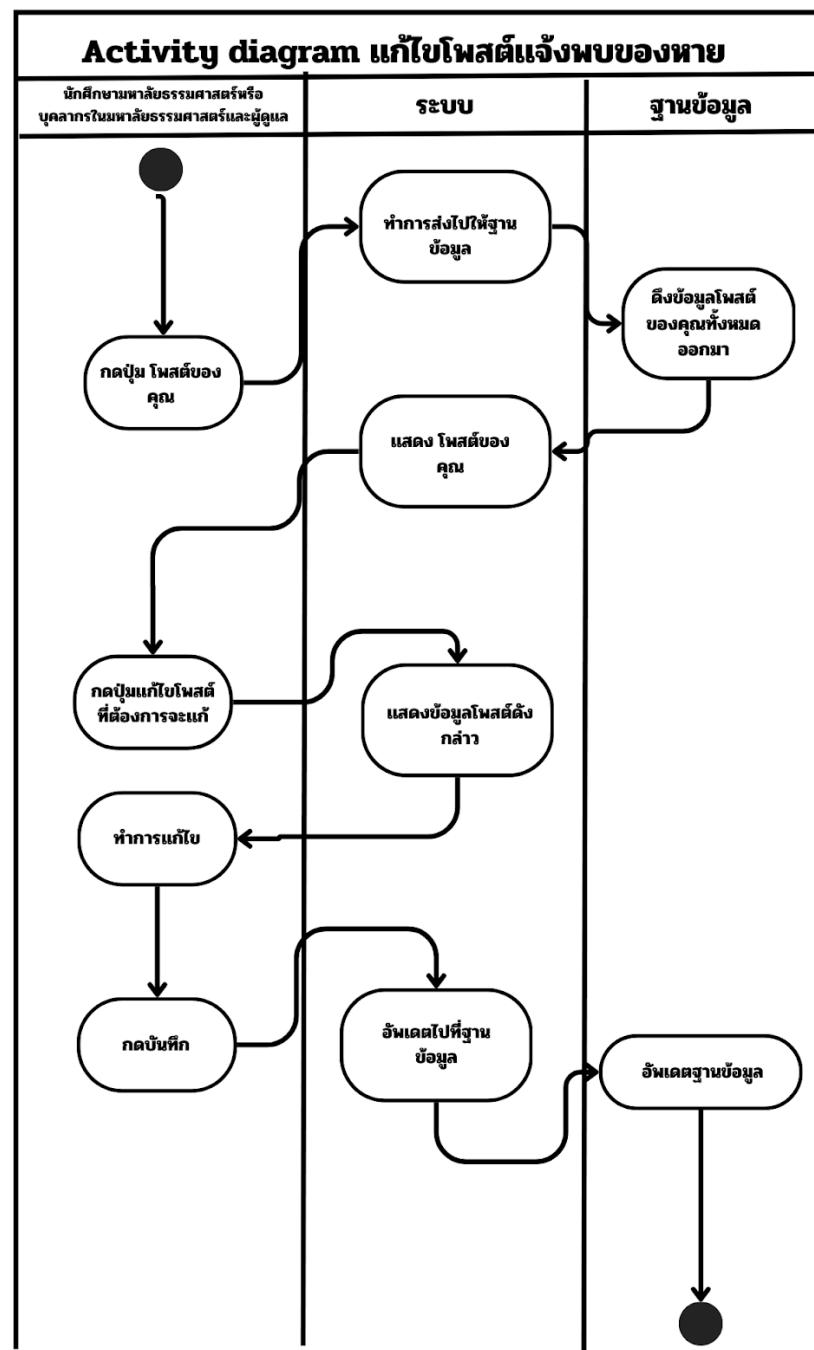
3.2.3 Activity Diagram



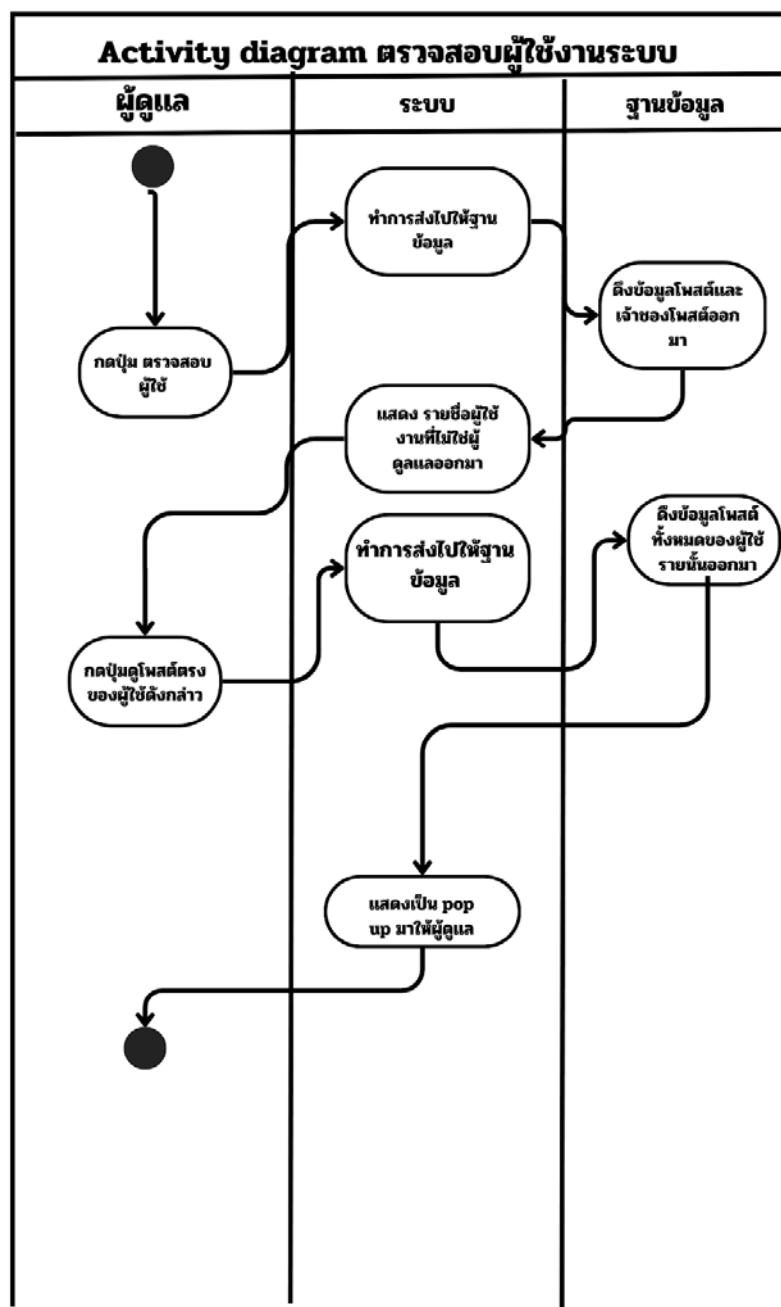
ภาพที่ 3.3 Activity diagram ค้นหารายชื่อของหายที่ถูกพบ



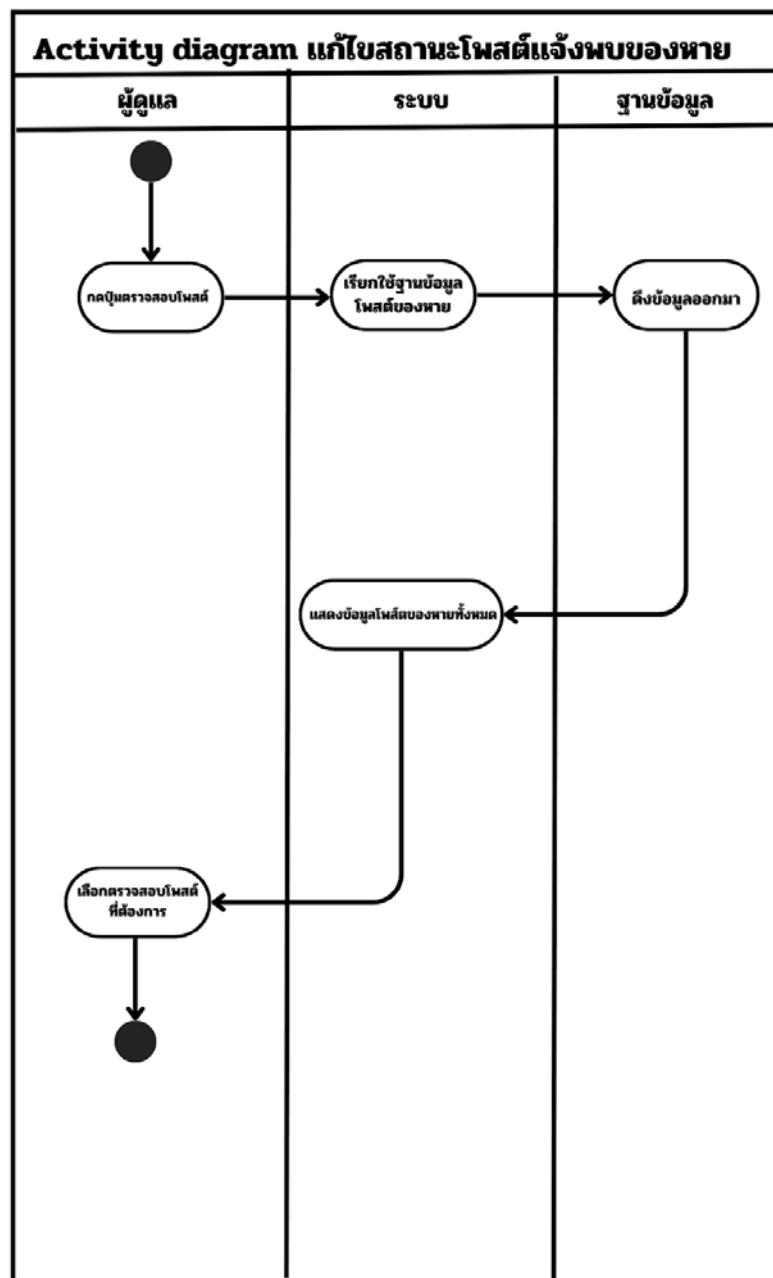
ภาพที่ 3.4 Activity diagram โพสต์ของหายที่ถูกพบ



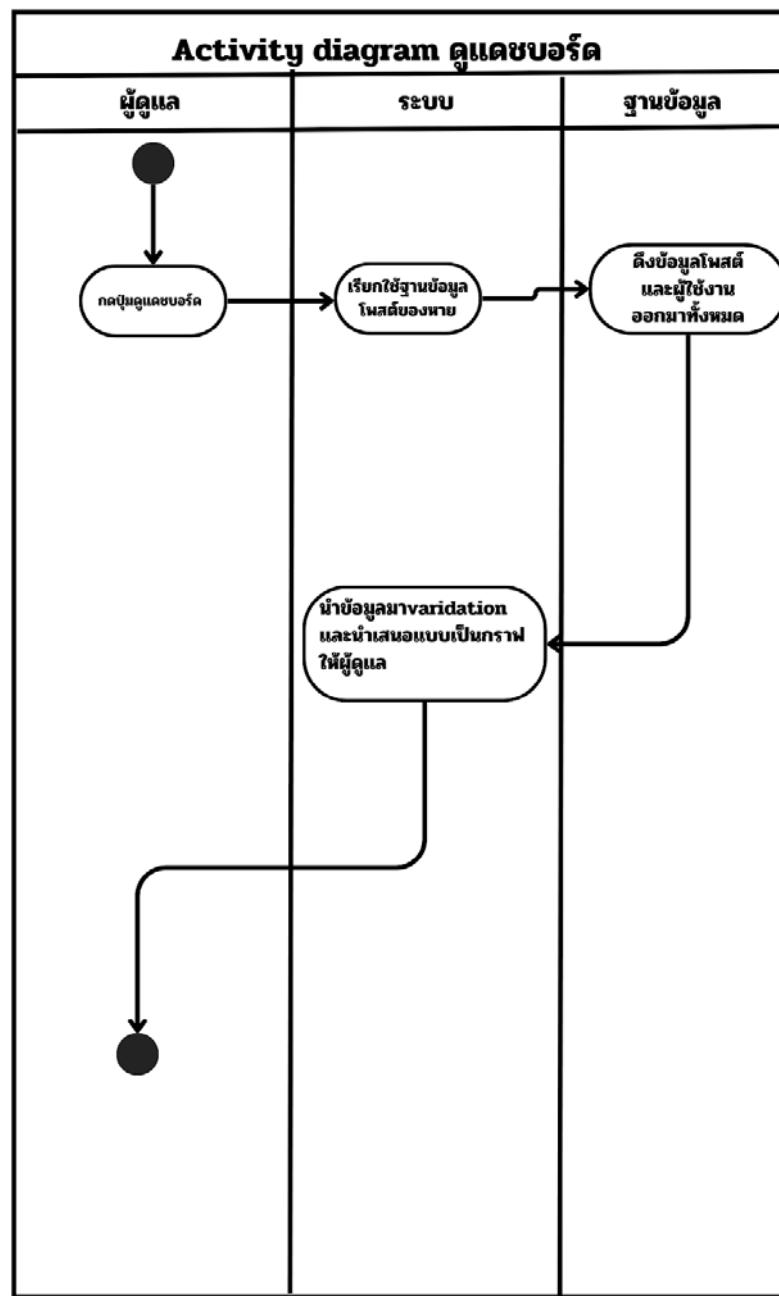
ภาพที่ 3.5 Activity diagram แก้ไขโพสต์แจ้งพบของหาย



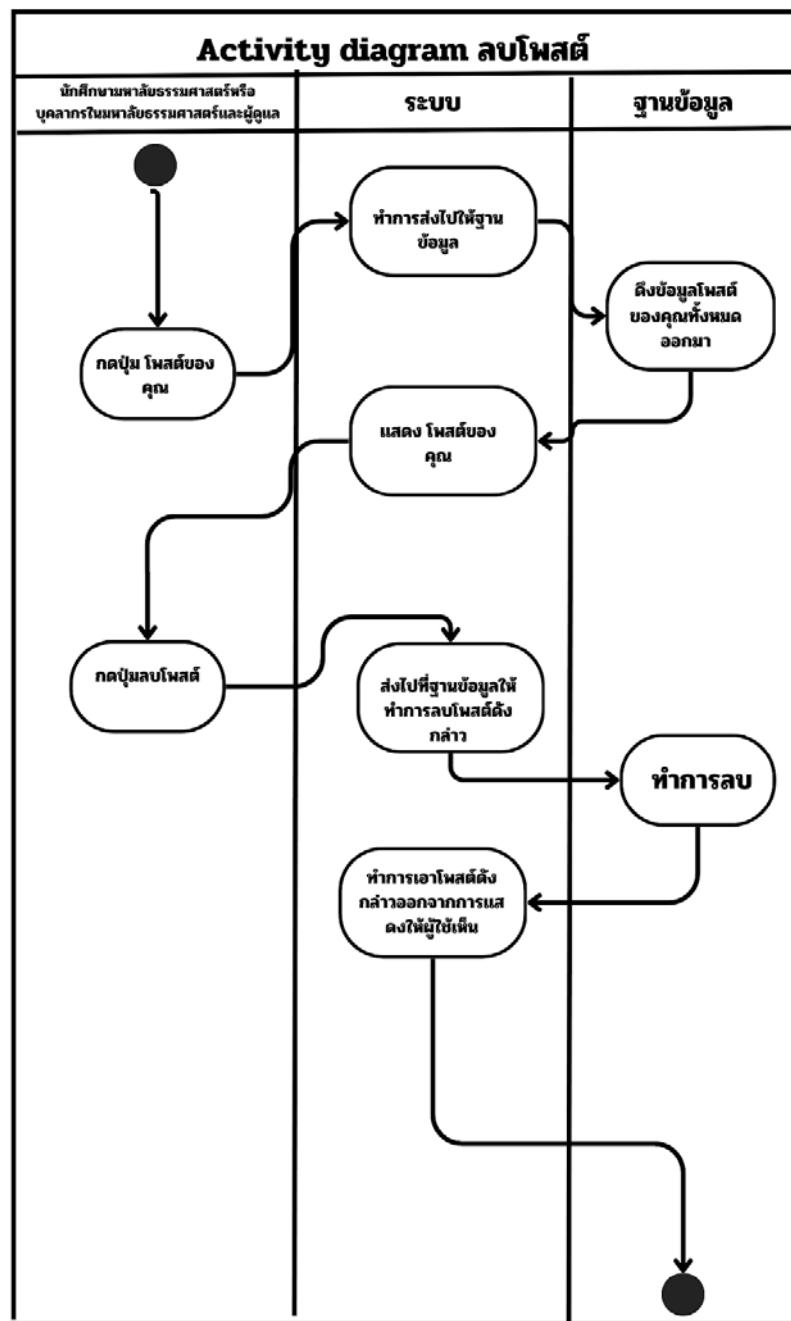
ภาพที่ 3.6 ภาพ Activity diagram ตรวจสอบผู้ใช้งานระบบ



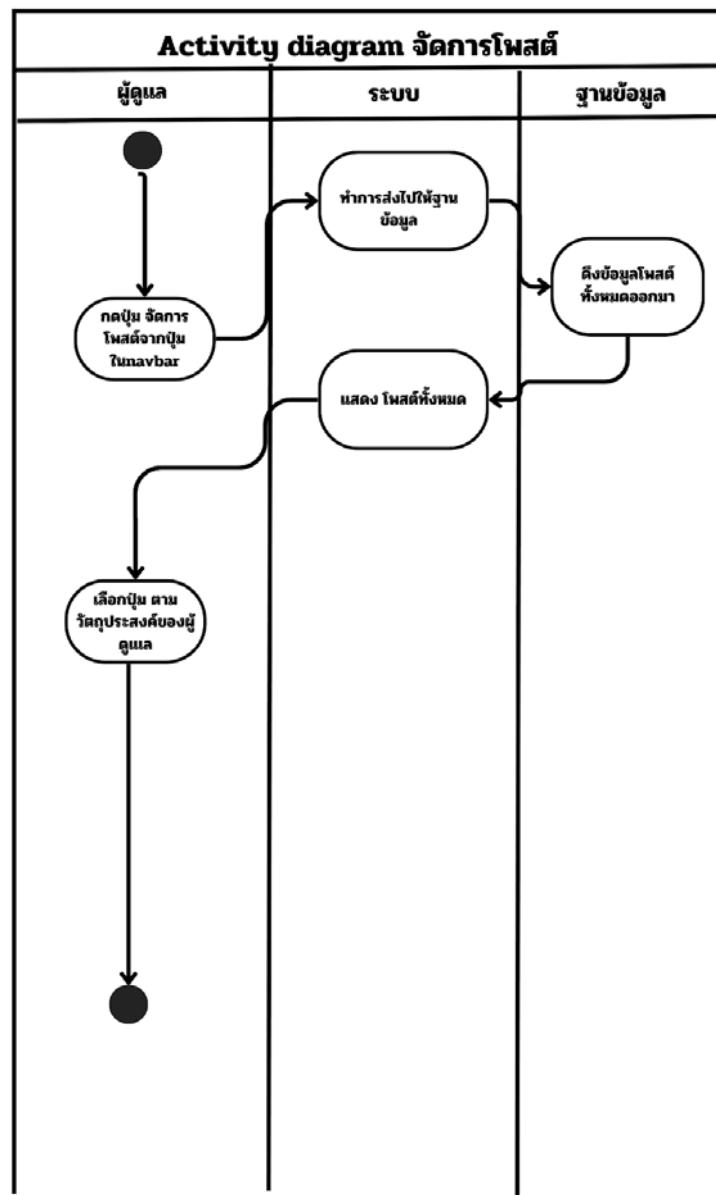
ภาพที่ 3.7 ภาพ Activity diagram แก้ไขสถานะโพสต์แจ้งพบของหาย



រាយទី 3.8 រាយ Activity diagram គុណដេខប័ណ្ណ

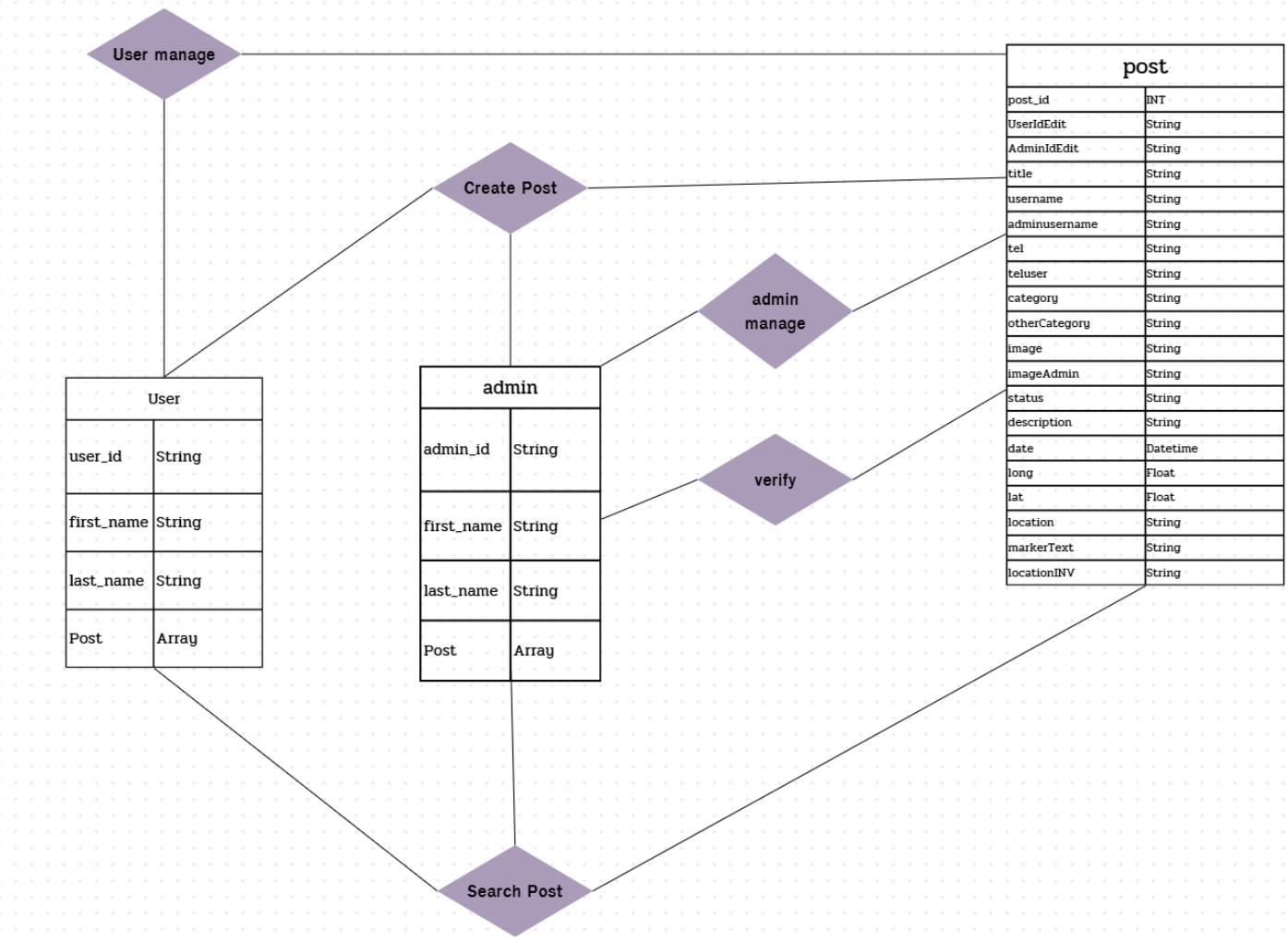


ภาพที่ 3.9 ภาพ Activity diagram ตรวจสอบผู้ใช้งาน



ภาพที่ 3.10 ภาพ Activity diagram จัดการโพสต์

3.2.4 การออกแบบฐานข้อมูล



ภาพที่ 3.11 แผนภาพ ER Diagram

3.2.5 พจนานุกรมข้อมูล

3.2.5.1 ตาราง User

Column Name	Data Type	Description
user_id	String	Primary key
first_name	String	User's first name
last_name	String	User's last name
Post	Array	User Post

ตารางที่ 3.11 รายละเอียดข้อมูลตาราง user

3.2.5.2 ตารางผู้ดูแล

Column Name	Data Type	Description
admin_id	String	Primary key
first_name	String	Admin first name
last_name	String	Admin last name
Post	Array	Admin Post

ตารางที่ 3.12 รายละเอียดข้อมูลตาราง ผู้ดูแล

3.2.5.3 ตารางโพสต์

Column Name	Data Type	Description
post_id	INT	Primary key, unique identifier for each post
UserIdEdit	String	Foreign key referencing the user_id column in the users table
AdminIdEdit	String	Foreign key referencing the admin_id column in the admin table
title	String	Post title
username	String	username of user
adminusername	String	username of admin
tel	String	telephone number
teluser	String	telephone number of user
category	String	Post category
otherCategory	String	otherCategory
image	String	Path to the post's image file
imageAdmin	String	image of Admin
status	String	Post status, such as "published" or "draft"
description	String	Post description
date	Datetime	Day of the month the post was created
long	Float	longitude
lat	Float	latitude
location	String	location
markerText	String	marker description
locationINV	String	Inventory of Landmark location

ตารางที่ 3.13 รายละเอียดข้อมูลโพสต์

3.3 ประเด็นที่น่าสนใจและสิ่งที่ห้ามทำ

ปัญหาที่มีความห้ามทำในโครงการนี้ประกอบด้วย 5 ข้อ

- 1) การศึกษาการเขียนโค้ดโดยใช้ Leaflet Map API เนื่องจากทีมพัฒนาไม่เคยใช้ Leaflet Map API มา ก่อน
- 2) การออกแบบหน้า User Experience และ User Interface ให้ใช้งานง่าย และมีความสวยงาม
- 3) การเขียน Logic การเก็บรูปภาพลง Firebase Storage และการแปลงจากรูปภาพให้เป็น url เพื่อนำไปเก็บไว้ในฐานข้อมูล Tu item Finder
- 4) การเชื่อมต่อ API เพื่อติดตอกันระหว่างหน้าบ้านและหลังบ้าน
- 5) การศึกษาและวิธีการ Deploy ลง Vercel เพื่อทดสอบการทำงาน

ประเด็นที่มีความน่าสนใจในโครงการมี 2 ประเด็นคือ

- 1) การเลือกใช้เครื่องมือในการพัฒนาระบบที่เหมาะสมกับเว็บแอปพลิเคชัน ตัวอย่าง เช่น การเลือกใช้ Figma ในการออกแบบ User Experience และ User Interface เพราะ Figma มี Design Features ที่โดดเด่นด้วยการสร้าง Shape ได้หลากหลายด้วย Pen tool ที่คุ้นเคย กันดีอยู่แล้ว เพิ่มความสะดวกสบายด้วย Arc tool ช่วยในการสร้างโครงสร้างต่างๆ ได้ง่ายมากขึ้น และมี Auto Layout สามารถ resize สิ่งต่างๆ ได้อัตโนมัติเมื่อความยาว text เปลี่ยนแปลง และง่ายต่อการเคลื่อนย้าย Component ต่างๆ
- 2) การทำ censored หรือการเบลอรูปภาพในการซ่อนรูปภาพบางรูปที่อาจจะส่งผลถึงตัวบุคคลได้

3.3 ผลลัพธ์ที่คาดหวัง

ผลลัพธ์ที่คาดหวังของโครงการมีดังนี้

- 1) TU Item finder สามารถทำได้ครบตาม use case
- 2) สามารถทำ User Interface ได้สวยงามและตรงตามที่ผู้พัฒนาต้องการ
- 3) ลดระยะเวลาในการค้นหาสิ่งของที่สูญหายและเป็นอีกหนึ่งช่องทางการหาของสูญหายในมหาวิทยาลัยธรรมศาสตร์ ศูนย์รังสิต

3.4 ระบบใช้งานจริง

3.5.1 หน้าหลักของผู้ใช้

เมื่อเข้ามาอยู่เว็บไซต์จะปรากฏหน้าแรก คุณสามารถใช้งานได้ดูข้อมูลของขายที่ถูกพับได้ทุกโพสต์ โดยที่ไม่ต้อง Login แต่จะจำกัดสิทธิการเข้าถึง การแจ้งพบของสูญหาย ที่จะต้องให้ผู้ใช้ทำการ Login

The screenshot shows the TultemFinder mobile application interface. At the top, there is a navigation bar with the logo 'TultemFinder' and icons for adding a post, logging in, and searching. Below the navigation bar is a search bar with placeholder text 'ค้นหา...'. Underneath the search bar are four product cards, each with a small image, the item name, a brief description, and a date. The first card is for an 'ipad' (สีชมพูปี๊บ) from 'ล็อกเกอร์' (Logger) with a yellow background. The second card is for a 'กระเป๋าสะพายข้าง' (Shoulder Bag) from 'ห้องน้ำดีป้า' (Dipa Bath) with a white background. The third card is for a 'กระเป๋าลังค์' (Lungkot Bag) from 'ล็อกเกอร์' (Logger) with a yellow background. The fourth card is for a 'กระเป๋าคอม' (Computer Bag) from 'ศูนย์อาหารกรีนแคนทัน' (Green canteen) with a grey background. Each card has a green 'อภินิคลัง' (Abhiniklang) button at the bottom.

ก่อนถึงจะสามารถใช้ฟีเจอร์ต่างๆ ได้อย่างเต็มที่

ภาพที่ 3.12 ตัวอย่างหน้าแรกของแอปพลิเคชัน

3.5.2 หน้าแรกของแอปพลิเคชันที่ผู้ใช้สามารถค้นหาของหายที่ถูกพบได้โดยไม่ต้อง Login เข้าใช้งาน

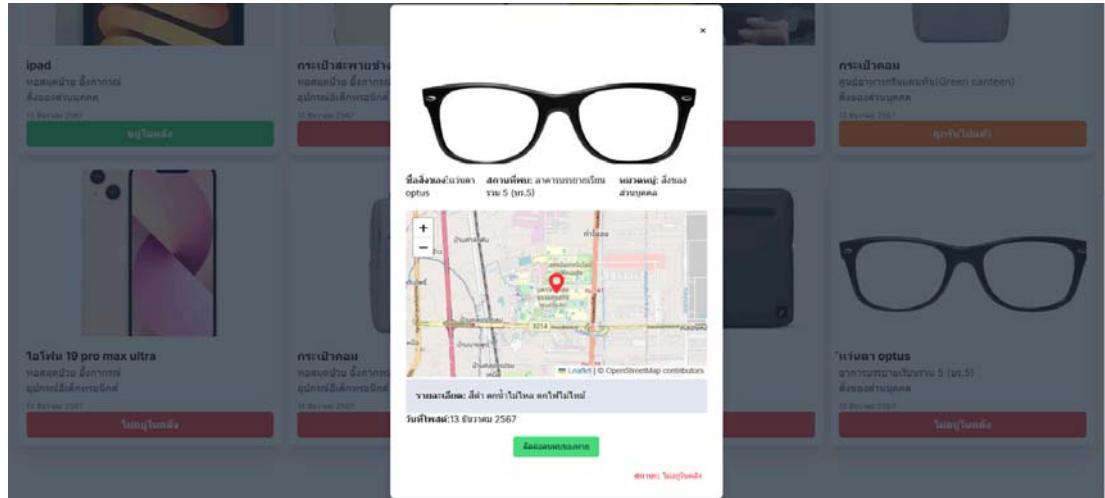
The screenshot shows the TultemFinder mobile application interface. At the top, there is a navigation bar with the TultemFinder logo, a search bar, and buttons for adding items, logging in, and a QR code scanner. Below the navigation bar, there is a search input field and several filters: 'หมวดหมู่' (Category) with a dropdown menu open, 'สถานที่' (Location), and 'สถานะ' (Status). A green 'ค้นหา' (Search) button is located to the right of these filters.

The main content area displays four search results cards:

- ipad**
ห้องบุดดี้บีบี รังษีภารกณ์
สั่งของส่วนบุคคล
13 ธันวาคม 2567
อยู่ในคลัง
- กระเป๋าสะพายข้าง**
ห้องบุดดี้บีบี รังษีภารกณ์
อุปกรณ์มือถือทรูมิลล์
สั่งของส่วนบุคคล
13 ธันวาคม 2567
ไม่อยู่ในคลัง
- กระเป๋าดังค์**
อาคารนรรยาเรียนรวม 5 (มร.5)
สั่งของส่วนบุคคล
13 ธันวาคม 2567
ไม่อยู่ในคลัง
- กระเป๋าคอม**
ศูนย์อาหารกรีนแคนท์เท็น(Green canteen)
สั่งของส่วนบุคคล
13 ธันวาคม 2567
ถูกรับไปแล้ว

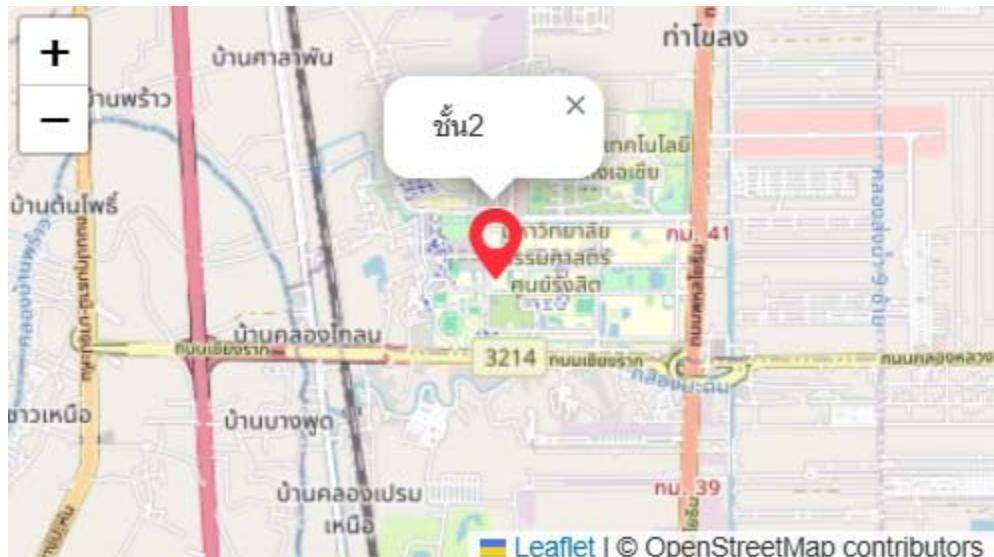
ภาพที่ 3.13 ตัวอย่างหน้าแรกของแอปพลิเคชันที่ผู้ใช้สามารถค้นหาของหายที่ถูกพบได้โดยไม่ต้อง Login เข้าใช้งาน

3.5.3 หน้าของการเรียกดูข้อมูลที่ต้องการ (โพสต์ที่เป็นสถานะอยู่ในคลัง และยังไม่login)



ภาพที่ 3.14 ตัวอย่างหน้าของการเรียกดูข้อมูลที่ต้องการ (โพสต์ที่เป็นสถานะถูกรับไปแล้ว และยังไม่login)

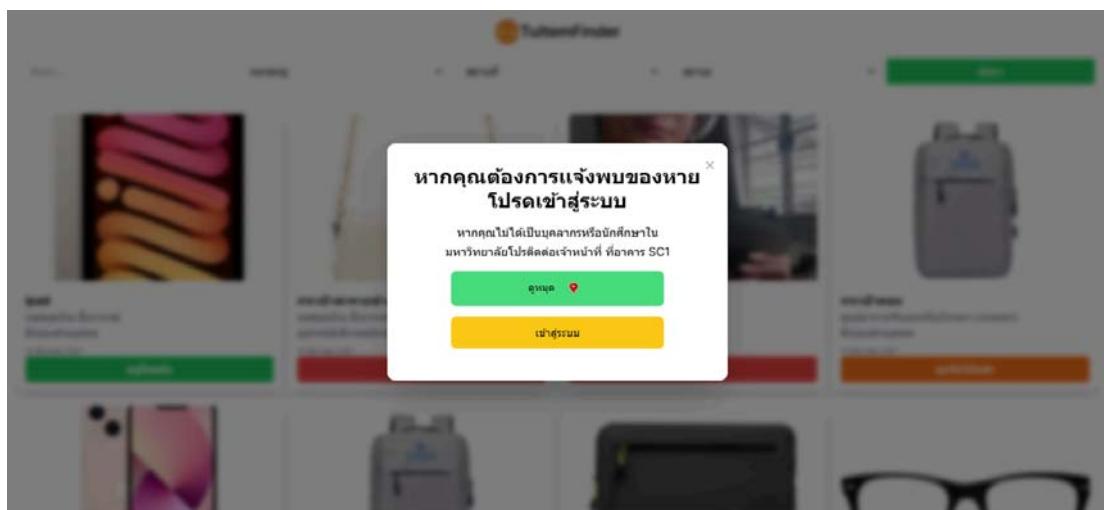
3.5.4 หน้าแสดงแผนที่ปักหมุดที่พบรของสูญหาย



ภาพที่ 3.15 ตัวอย่างหน้าแสดงแผนที่ปักหมุดที่พบรของสูญหาย

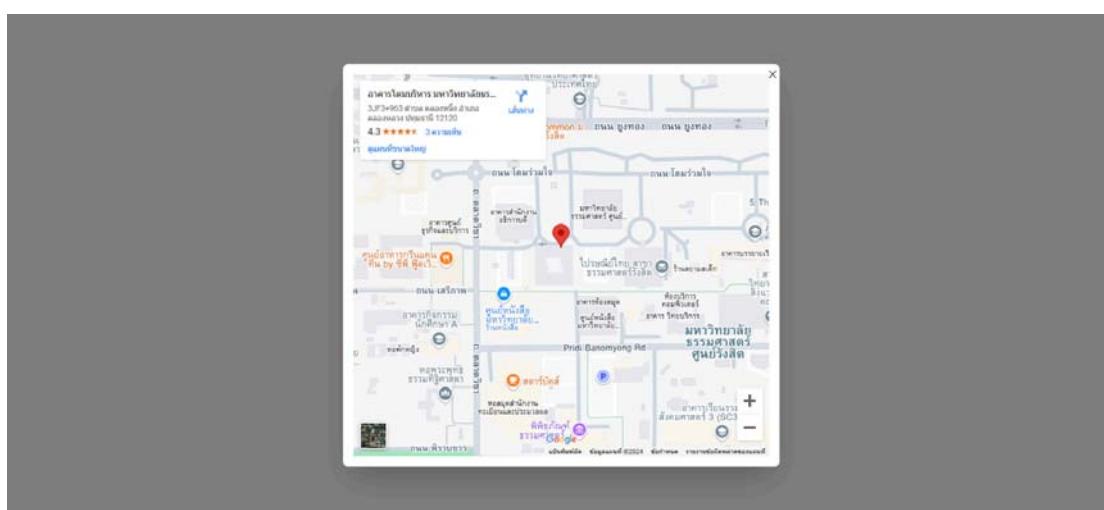
3.5.5 หน้าแจ้งเตือนผู้ใช้เรื่อง Login

หน้าดังกล่าวจะแสดงขึ้นก็ต่อเมื่อผู้ใช้ทำการกดปุ่ม แจ้งพบของสูญหาย โดยรอบหัวดังกล่าว จะแสดง คำอธิบายให้ผู้ใช้ทำการ login เข้าสู่ระบบ หากไม่ใช่บุคลากรหรือนักศึกษา ของมหาวิทยาลัยธรรมศาสตร์ ให้ทำการติดต่อ เจ้าหน้าที่โดยตรง ที่ อาคาร SC1 ซึ่งผู้ใช้สามารถ เลือกดูหมุดที่ตั้งของ อาคาร SC1 ได้ ในกรณีที่ผู้ใช้ไม่ทราบสถานที่ดังกล่าวว่าอยู่ที่ใด



ภาพที่ 3.16 ตัวอย่างหน้าแจ้งผู้ใช้เรื่อง login

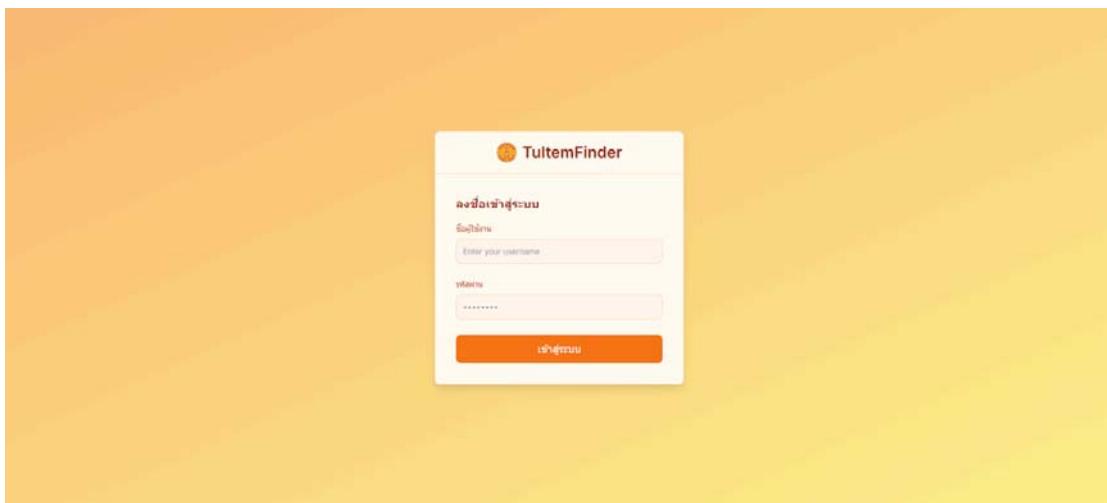
3.5.6 หน้าแสดงแผนที่เมื่อผู้ใช้ทำการดูหมุดสถานที่เก็บของหาย



ภาพที่ 3.17 ตัวอย่างหน้าแสดงแผนที่เมื่อผู้ใช้ทำการดูหมุดสถานที่เก็บของหาย

3.5.7 หน้า Login

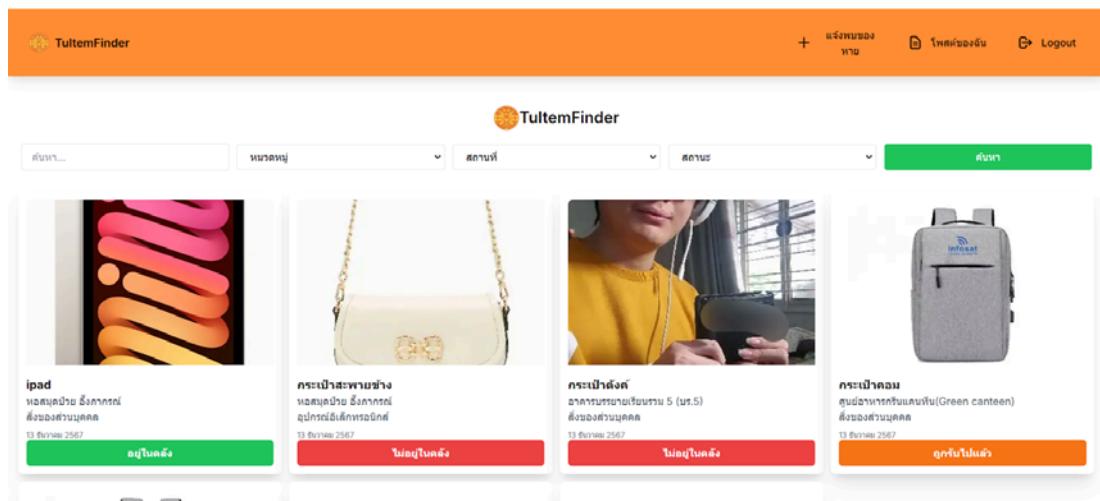
หน้านี้จะให้ผู้ใช้ทำการใส่เลขรหัสบุคลากรของมหาวิทยาลัยธรรมศาสตร์ และ password เป็นเลขรหัสบัตรประชาชน หลังจากที่ลงชื่อเข้าใช้ไปแล้ว ระบบจะทำการเทียบกับ ระบบ TU APIs Developers ของมหาวิทยาลัย ธรรมศาสตร์ ว่าตรงกันหรือไม่



ภาพที่ 3.18 ตัวอย่างหน้าLoginของแอปพลิเคชัน

3.5.8 หน้าหลักหลังจาก Login

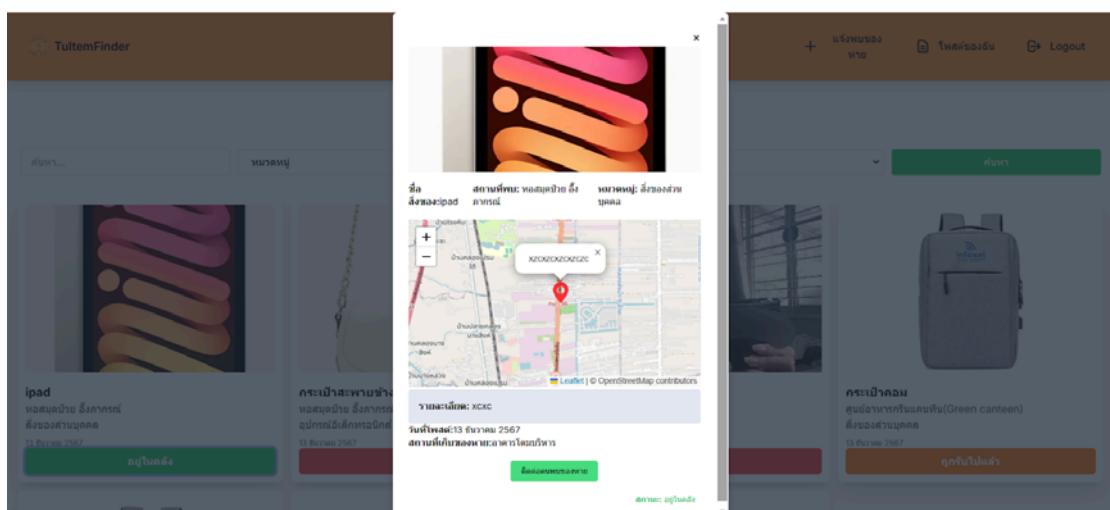
หลังจากที่ผู้ใช้ทำการ เข้าสู่ระบบ มาแล้ว จะแสดงหน้าหลักของมา ซึ่งหน้านี้ ในแต่ละเมนู ผู้ใช้จะสามารถ แจ้งพบรของหายได้ และสามารถดูโพสต์ที่ตัวเอง เคยโพสต์ไว้ได้



ภาพที่ 3.19 ตัวอย่างหน้าLoginของแอปพลิเคชันหน้าหลักหลังจาก login

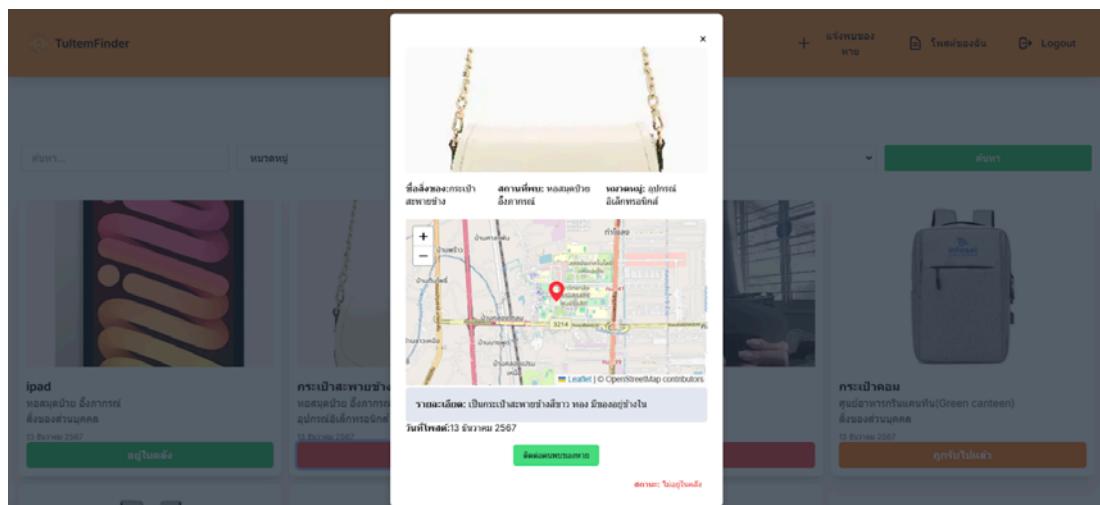
3.5.9 หน้าดูรายละเอียดโพสต์

เมื่อผู้ใช้ทำการกดดูโพสต์ ที่ต้องการจะดูแล้ว หน้าจะแสดง popup ขึ้นมา โดยหน้าpopup จะแสดงรายละเอียดของโพสต์ที่คุณต้องการจะดูอีกมา โดยจะมีบอกชื่อ หมวดหมู่ สถานที่พับ และรายละเอียดของ ของขึ้นนั้น มีปุ่ม แสดงหมวดที่จะบอกถึงจุดที่พับเจอ และมีบอกสถานะของ ของนั้น ว่าอยู่ สถานะไหน และถูกเก็บไว้ที่ใด(อยู่ในคลัง,ไม่อยู่ในคลัง,ถูกรับไปแล้ว)



ภาพที่ 3.20 ตัวอย่างหน้าดูรายละเอียดโพสต์ของรายการแจ้งพบของ hairy (สถานะอยู่ในคลัง)

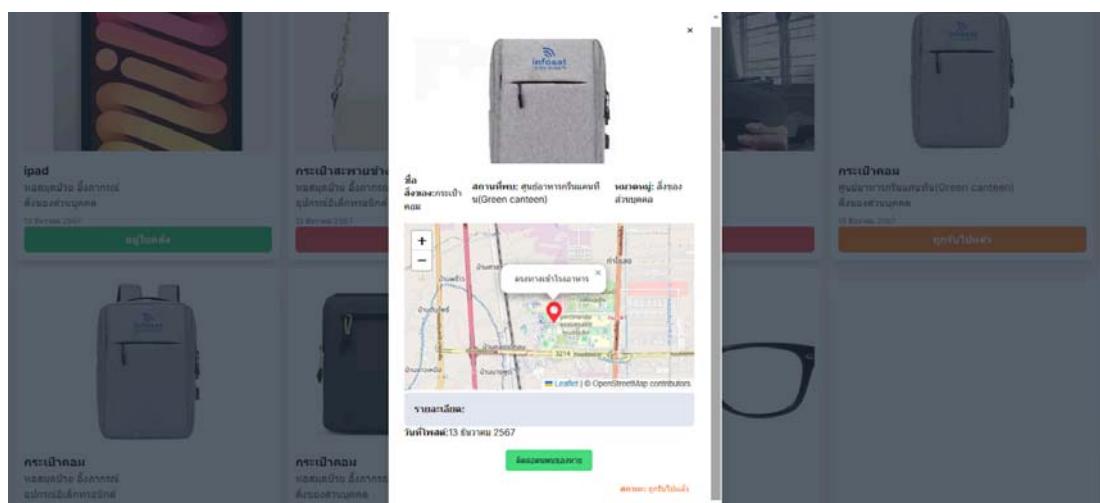
3.5.10 หน้าแดรย์ල์เอียดของรายการแจ้งพบรของ hairy(สถานะไม่อยู่ในคลัง)



ภาพที่ 3.21 ตัวอย่างหน้าแดรย์ล์เอียดของรายการแจ้งพบรของ hairy(สถานะไม่อยู่ในคลัง)

3.5.11 หน้าแดรย์ล์เอียดของรายการแจ้งพบรของ hairy(สถานะถูกรับไปแล้ว)

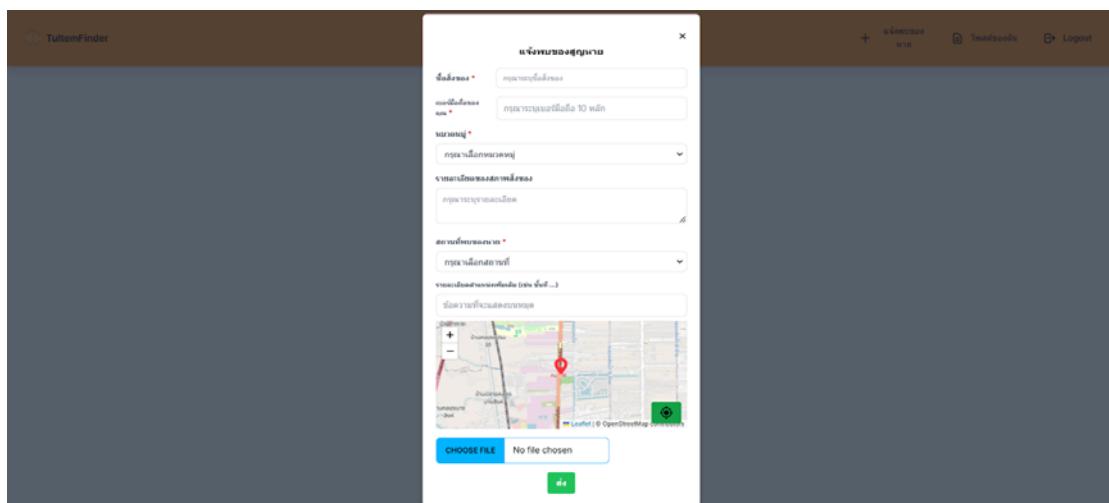
หน้านี้จะต่างจากหน้า 2 สถานะ ที่ได้อธิบายไปข้างต้นโดยสถานะนี้จะมีปุ่มเพิ่มเข้ามา 1 ปุ่ม คือ ปุ่ม ดูรูปหลักฐานกับรับของ เพราะสถานะนี้เป็นสถานะ ที่คนที่ทำของหายมารับเรียบร้อยแล้ว และจะต้องถ่ายรูปเพื่อเป็นหลักฐานและเก็บไว้ในระบบ



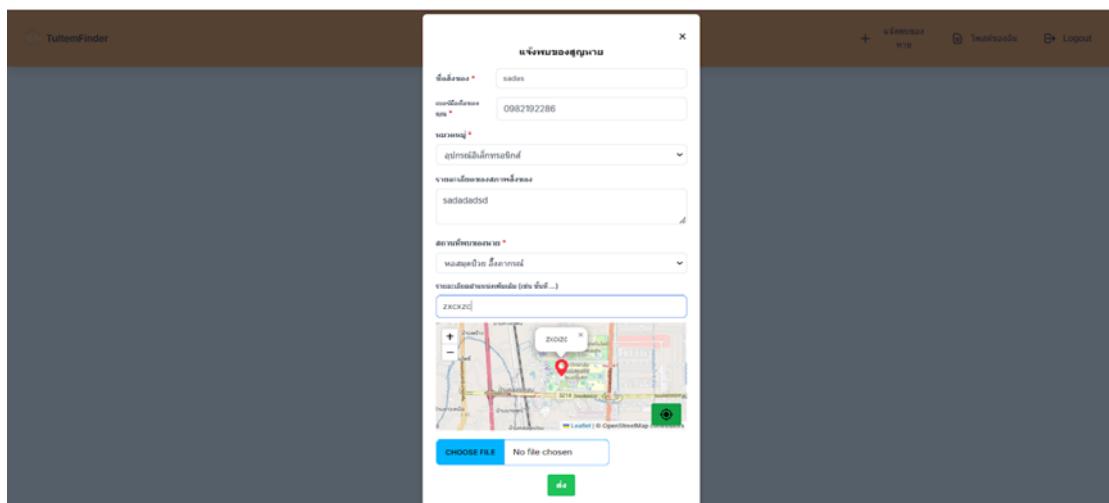
ภาพที่ 3.22 ตัวอย่างหน้าแดรย์ล์เอียดของรายการแจ้งพบรของ hairy(สถานะถูกรับไปแล้ว)

3.5.12 หน้าแจ้งpubของหาย

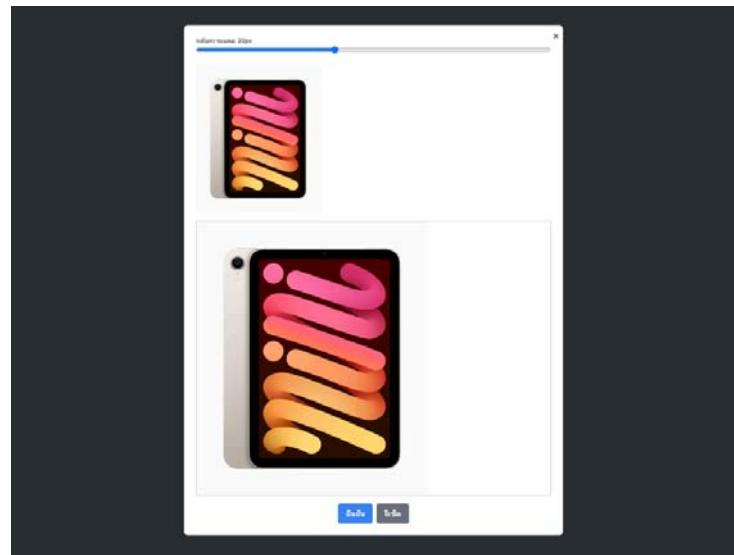
เมื่อผู้ใช้จะกดปุ่ม “แจ้งpubของหาย” จะปรากฏหน้าสร้างโพสต์ ที่จะต้องกรอกรายละเอียดของที่พบรูป ดังนี้ 1.ชื่อสิ่งของ 2.หมวดหมู่ 3.รายละเอียดของสิ่งของที่พบรูป 4.สถานที่พบรูปของ 5.รูปภาพ 6.ตำแหน่งที่พบรูปของสูบหาย หลังแนบรูป จะขึ้นหน้าให้เบลอรูปภาพ



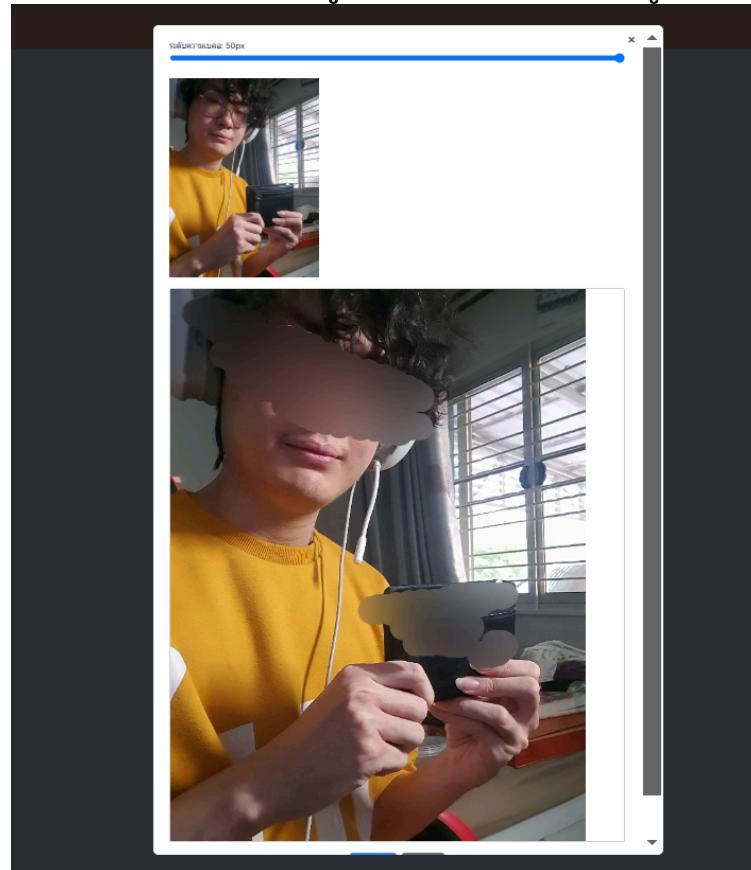
ภาพที่ 3.23 ตัวอย่างหน้าการแจ้งpubของหาย



ภาพที่ 3.24 ตัวอย่างหน้าการแจ้งpubของหายแบบกรอกรายละเอียดครบ
หลือแนบรูป



ภาพที่ 3.25 หลังแนบรูป จะขึ้นหน้าให้เบลอรูปภาพ



ภาพที่ 3.26 หน้าตัวอย่างหลังจากการเบลอ



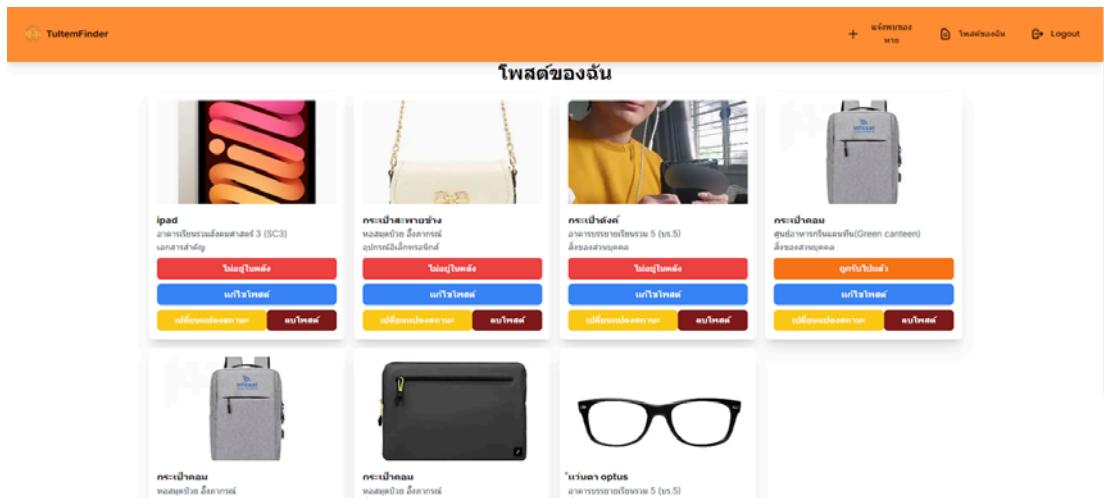
ภาพที่ 3.27 หน้าตัวอย่างหลังจากการกดปุ่มโพสต์ไป



ภาพที่ 3.28 หน้าตัวอย่างหลังจากการเบลอ

3.5.13 หน้าโพสต์ของฉัน

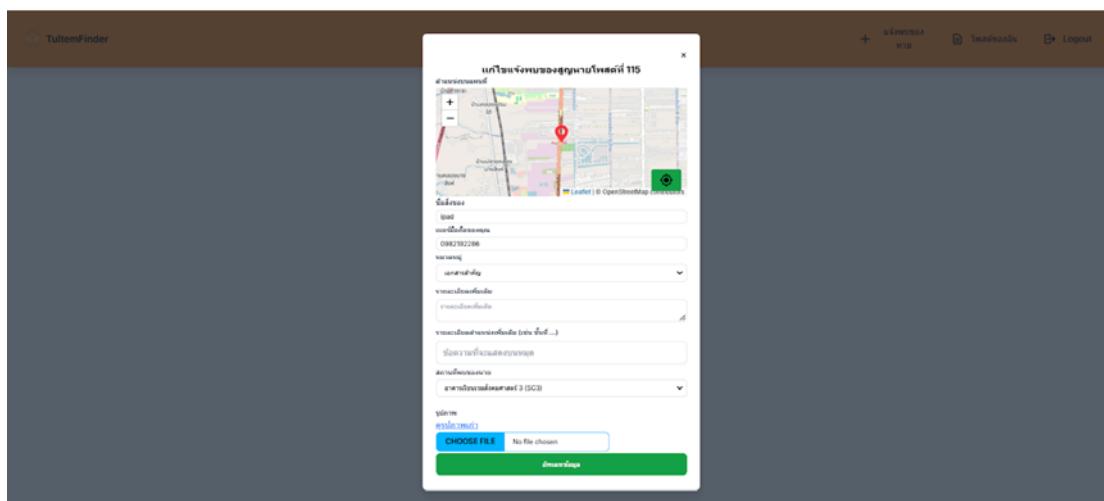
เมื่อผู้ใช้ทำการกดที่ปุ่ม ”โพสต์ของฉัน” จะแสดงหน้าที่มีโพสต์ของผู้ใช้ที่เคยโพสต์ไว้ทั้งหมด ผู้ใช้จะสามารถที่จะกดปุ่มแก้ไขโพสต์ที่เคยโพสต์ไว้ได้ และสามารถเปลี่ยนแปลงสถานะได้



ภาพที่ 3.29 ตัวอย่างหน้าดูโพสต์ของฉัน

3.5.14 หน้าแก้ไขโพสต์ (โพสต์ของฉัน)

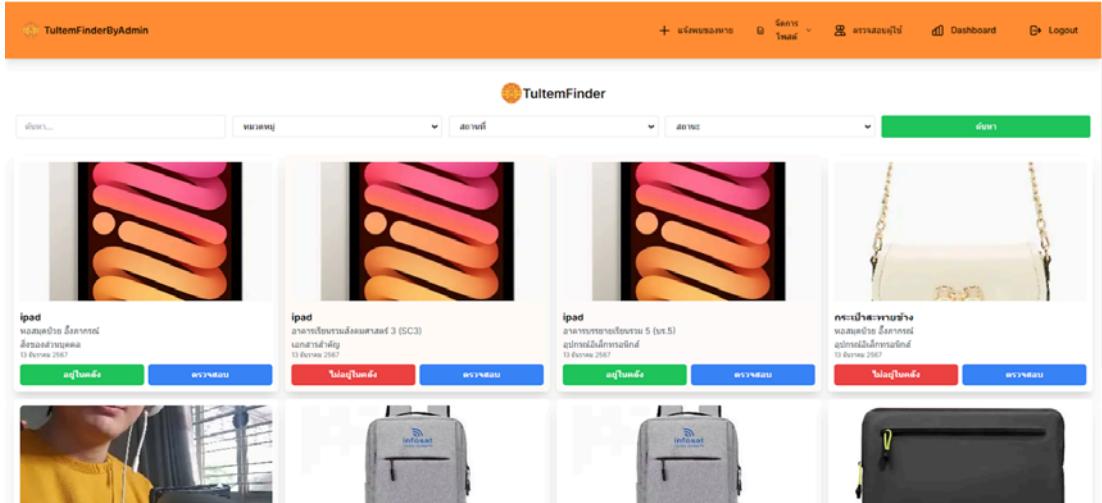
เมื่อผู้ใช้ทำการกดปุ่ม ”แก้ไข” มันจะแสดงหน้าสำหรับแก้ไขข้อมูลการแจ้งพบของหาย โดยผู้ใช้สามารถแก้ไขรายละเอียดต่างๆ ของโพสต์ที่เคยสร้างไว้แล้วถ้าผู้ใช้ทำการแก้ไขเสร็จแล้วทำการกด “อัพเดทข้อมูล” ระบบจะทำการบันทึกข้อมูลที่แก้ไขไปแล้วล่าสุด และทำการอัพเดทโพสต์ให้เป็นปัจจุบัน



ภาพที่ 3.30 ตัวอย่างหน้าแก้ไขโพสต์ของฉัน

3.5.15 หน้าหลักของผู้ดูแลระบบ

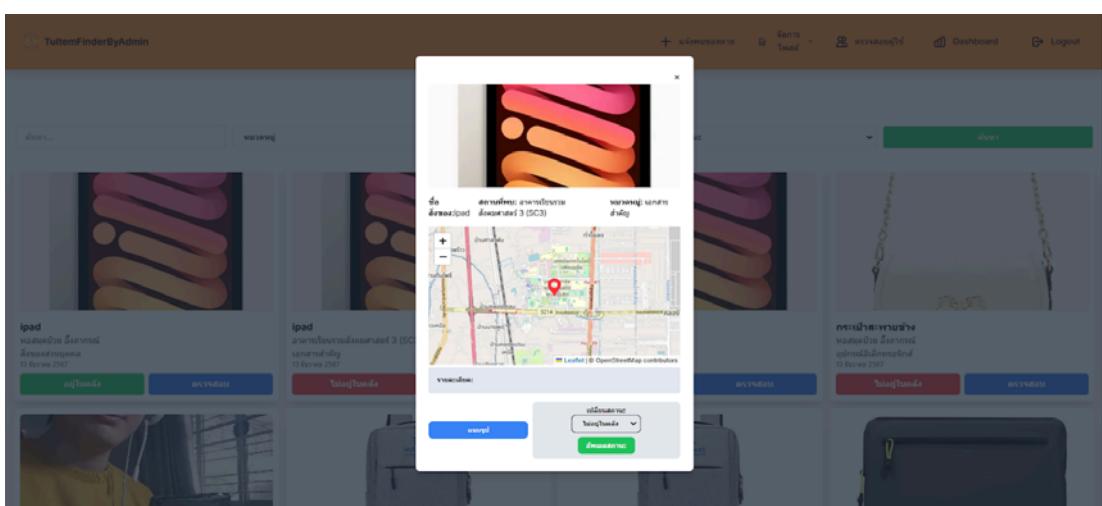
หน้านี้จะแสดง หน้าหลักสำหรับผู้ดูแลซึ่งสามารถดูของได้ได้ตามปกติเหมือนกับผู้ใช้งานปกติและแจ้งพบของหายได้เหมือนกัน แต่จะต่างกันตรงที่จะไม่มีปุ่มโพสต์ของฉัน และผู้ดูแลระบบจะสามารถตรวจสอบโพสต์ อัพเดตสถานะได้



ภาพที่ 3.31 ตัวอย่างหน้าแรกของผู้ดูแลระบบ

3.5.16 หน้าตรวจสอบโพสต์ (สำหรับผู้ดูแล)

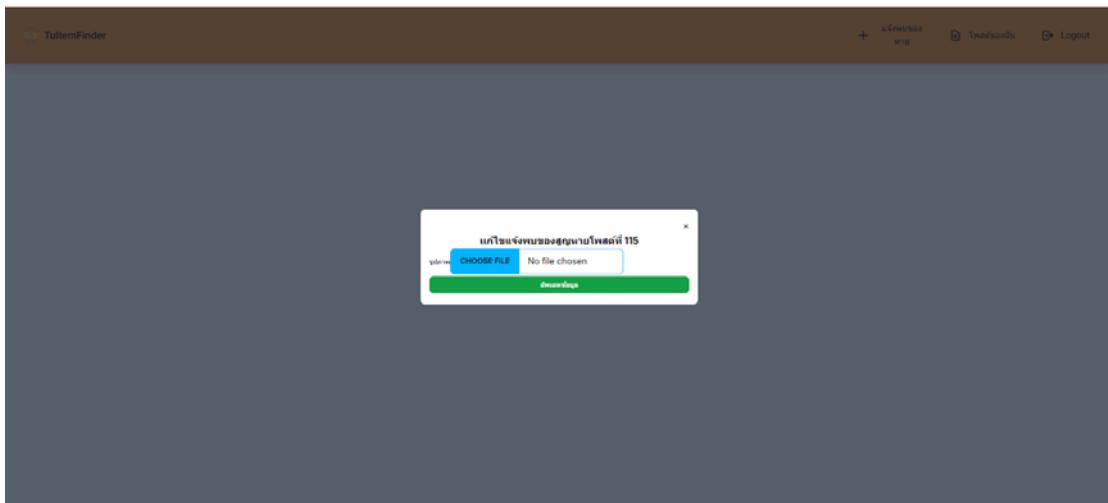
โดยหน้านี้ ปกติแล้วถ้ามีผู้ใช้งานโพสต์มาสถานะตั้งต้นจะเป็นไม่ได้อยู่ในคลัง (default)ทันที และเมื่อผู้ใช้ได้ทำการนำของมาให้เจ้าหน้าที่เก็บ เจ้าหน้าที่จะทำการตรวจสอบและตรวจสอบโพสต์ ดังกล่าว เพื่อปรับสถานะของโพสต์



ภาพที่ 3.32 ตัวอย่างหน้าตรวจสอบโพสต์ (สำหรับผู้ดูแล)

3.5.17 หน้าแบบรูปหลักฐาน(สำหรับผู้ดูแล)

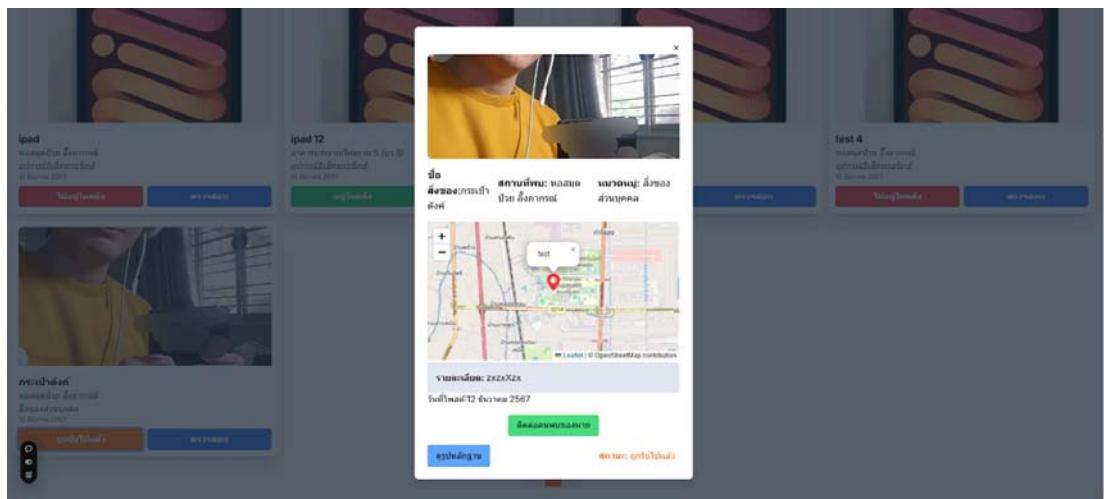
เมื่อมีเจ้าของของหายมารับของหายของตัวเอง ผู้ดูแลจะเป็นจะต้องถ่ายรูปเพื่อทำเป็นหลักฐานไว้เพื่อกรณีที่คนที่มา_rับของหายไม่ใช่เจ้าของจริงๆถ่ายบัตรประชาชน หรือ ถ่ายรูปกับคนที่มา_rับ สุดท้ายคนที่เห็นรูปหลักฐานได้จะมีเพียงผู้ดูแลเท่านั้นที่สามารถเห็นได้ หลังจากแนบรูปไปแล้ว ให้ผู้ดูแลเปลี่ยนสถานะเป็น ถูกรับไปแล้ว



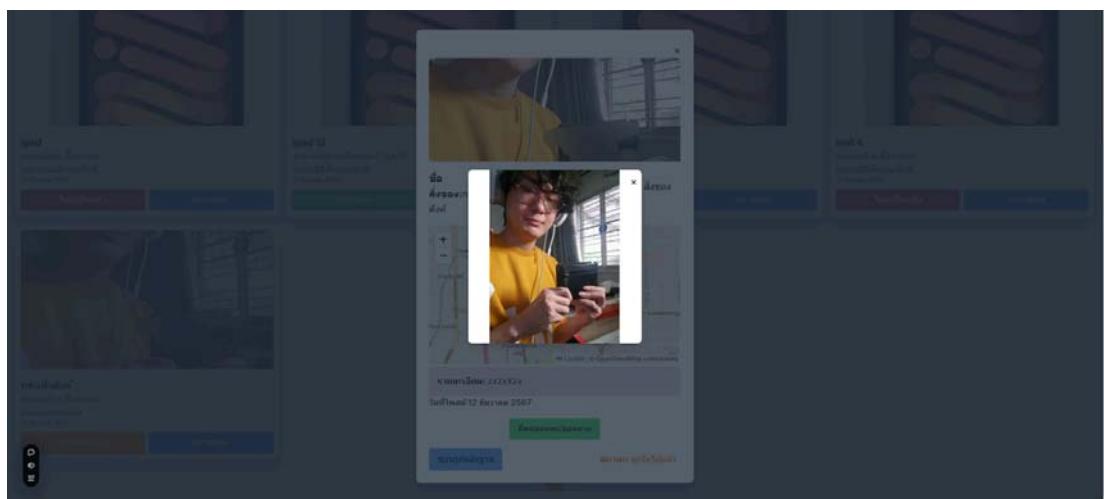
ภาพที่ 3.33 ตัวอย่างหน้าหน้าแบบรูปหลักฐาน (สำหรับผู้ดูแล)

3.5.18 หน้าคูรายละเอียดของโพสต์แจ้งพบของหาย (สถานะ ถูกรับไปแล้ว , เป็นผู้ดูแล)

โดยหน้าคูรายละเอียดของโพสต์แจ้งพบของหาย สถานะถูกรับไปแล้ว จะไม่เหมือนหน้าคูรายละเอียดของโพสต์ของผู้ใช้ปกติ โดยมุ่งมองของผู้ดูแลจะมีปุ่มแสดงอุปกรณ์เพิ่มนั่นคือ ดูรูปหลักฐานหลังกดปุ่มไปแล้ว จะแสดงป้อมป้อปอัพรูปหลักฐานขึ้นมา



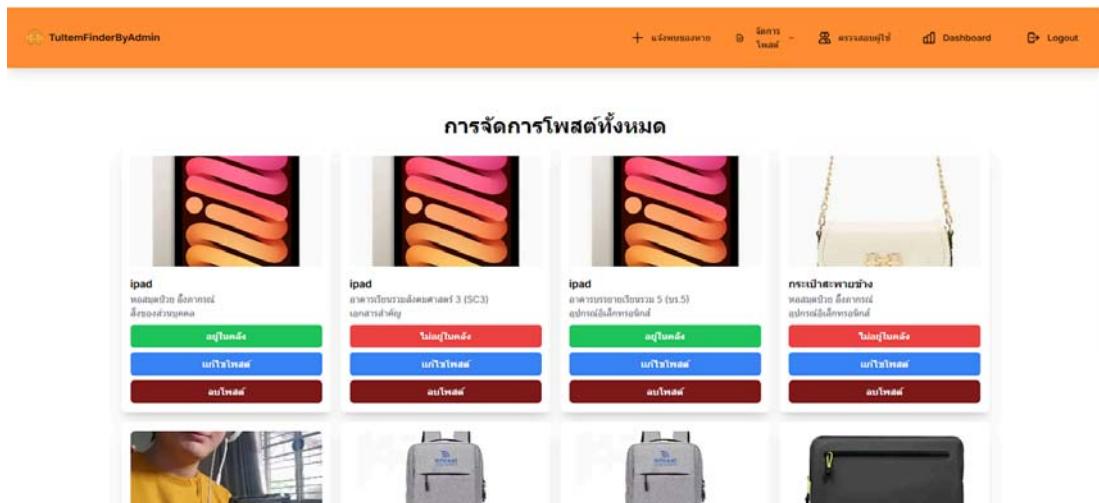
ภาพที่ 3.34 ดูรายละเอียดของโพสต์แจ้งพบของหาย (สถานะ ถูกรับไปแล้ว , เป็นผู้ดูแล)



ภาพที่ 3.35 หลังจากกดดูรูปหลักฐาน

3.5.19 หน้าจัดการโพสต์ (สำหรับผู้ดูแล)

โดยหน้าจัดการโพสต์ของผู้ดูแล จะสามารถจัดการได้ทุกโพสต์ไม่ว่าจะเป็นผู้ใช้หรือตัวเองก็ตาม



ภาพที่ 3.36 หน้าจัดการโพสต์ (สำหรับผู้ดูแล)

3.5.20 หน้าจัดการโพสต์ของฉัน (สำหรับผู้ดูแล)

โดยหน้าดูโพสต์ของฉันของผู้ดูแล จะสามารถจัดการโพสต์ที่ผู้ดูแล เป็นคนโพสต์เองเท่านั้น โดยจะมีปุ่ม แก้ไขโพสต์ , เปลี่ยนสถานะ , ลบโพสต์



ภาพที่ 3.37 หน้าดูโพสต์ของฉัน (สำหรับผู้ดูแล)

3.5.20 หน้าดูตรวจสอบผู้ใช้ (สำหรับผู้ดูแล)

หน้า “ตรวจสอบผู้ใช้” สามารถที่จะดูรายชื่อผู้ใช้ที่มีในระบบและสามารถดูจำนวน และ โพสต์ ของผู้ใช้แต่ละคนได้ ว่าผู้ใช้แต่ละคนมีโพสต์อะไรบ้าง

รหัสผู้ใช้	ชื่อผู้ใช้	สถานะ	ดำเนินการ
6409650121	น้องน้ำ	ไม่พบ	<button>ดูผลลัพธ์</button>
6409650782	พี่สาวน้ำ	ไม่พบ	<button>ดูผลลัพธ์</button>
6409650360	น้องน้ำ	ไม่พบ	<button>ดูผลลัพธ์</button>
6409650204	น้องน้ำ	ไม่พบ	<button>ดูผลลัพธ์</button>

ภาพที่ 3.38 หน้าดูตรวจสอบผู้ใช้ (สำหรับผู้ดูแล)

รายละเอียดโพสต์

ipad โพสต์เมื่อ 10 นาทีที่แล้ว

รายละเอียด

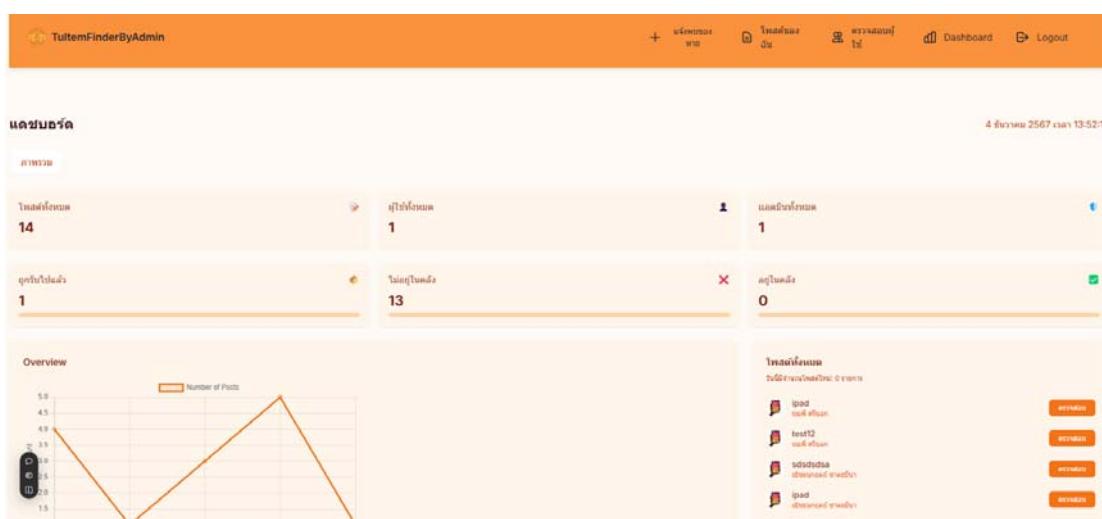
13 ผู้ชม 2567

ดำเนินการ >

ภาพที่ 3.39 หน้าดูตรวจสอบผู้ใช้หลังกดปุ่มจำนวนโพสต์ (สำหรับผู้ดูแล)

3.5.21 หน้าดูแดชบอร์ด (สำหรับผู้ดูแล)

หน้านี้เป็นหน้าแดชบอร์ดสำหรับผู้ดูแลระบบ (Admin) ที่แสดงข้อมูลสถิติและสามารถที่จะจัดการโพสต์ต่างๆได้



ภาพที่ 3.40 หน้าดูตรวจสอบผู้ใช้ (สำหรับผู้ดูแล)

บทที่ 4 ทรัพยากรและแผนการดำเนินงาน

4.1 การจัดเตรียมฮาร์ดแวร์และซอฟต์แวร์

4.1.1 ฮาร์ดแวร์ที่ใช้ในการพัฒนา

Computer 1	Computer 2	Smart phone
Device name: Lenovo Legion 5 15ACH6H CPU : AMD Ryzen 7 5800H Memory : 16 GB Graphic : NVIDIA Geforce GTX 3060 Storage : 512GB System type : 64 bit operating system	Device name: SKU CPU : core i5 1350 Memory : 32 GB Graphic : NVIDIA Geforce RTX 3060TI Storage : 512 GB System type : x64 basePC	รุ่น : Samsung Galaxy S9+ Memory : 128 GB OS : Android Processor : [Exynos 9810 Octa-core (4x2.7 GHz Mongoose M3 & 4x1.8 GHz Cortex-A55) - Global version

ตารางที่ 4.1 ฮาร์ดแวร์ที่ใช้ในการพัฒนา

4.1.2 ซอฟแวร์ที่ใช้ในการพัฒนา

- 1) PostgreSQL
- 2) Next.js version 14.2.13

4.1.3 ภาษาที่ใช้ในการพัฒนา (Programing Language)

- 1) Javascript FrameWork (typescript,React)
- 2) Tailwind Css
- 3) HTML

4.1.4 โปรแกรมที่ใช้ในการพัฒนา

- 1) Visual Studio
- 2) Vercel

4.2 แผนการดำเนินงาน

ขั้นตอน	มค. 67				กพ. 67				มีค. 67				เมย. 67				พค. 67			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. กำหนดหัวข้อโครงงานและพบปρีกษาอาจารย์ที่ปรึกษา																				
2. ค้นคว้าข้อมูลที่เกี่ยวข้องและบทหวานวรรณกรรมของโครงงาน																				
3. ศึกษาเทคโนโลยีที่เกี่ยวข้องกับโครงงาน																				
4. ศึกษาและออกแบบวิธีการทำงานของระบบ																				
5. กำหนดทรัพยากรและแผนการดำเนินงาน																				
6. จัดทำเอกสารโครงงานพิเศษ 1																				
7. นำเสนอโครงงานพิเศษ 1																				

ตารางที่ 4.2 การดำเนินงานที่ผ่านมาจนถึงปัจจุบัน (ภาคการศึกษาที่ 2/2566)

ขั้นตอน	สค. 67				กย. 67				ตค. 67				พย. 67				ธค. 67			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. ปรับปรุงและแก้ไขระบบจากโครงงานพิเศษ 1																				
2. ลงพื้นที่หาข้อมูลและสำรวจข้อมูล																				
3. ศึกษาการพัฒนาระบบที่เป็นไปได้																				
4. ทบทวนวรรณกรรม/องค์ความรู้ที่ใช้																				
5. ออกแบบระบบ																				
6. ทดสอบระบบ																				
7. แก้ไขข้อบกพร่องที่ได้จากการทดสอบ																				
8. จัดทำเอกสารโครงงานพิเศษ 2																				
9. นำเสนอข้อเสนอโครงงานโครงงานพิเศษ 2 แก่คณะกรรมการ และอาจารย์ที่ปรึกษาโครงการ																				

ตารางที่ 4.3 แผนการดำเนินงานในอนาคต (ภาคการศึกษาที่ 1/2567)

4.3 วิธีทดสอบระบบ

จากการที่ได้พัฒนาเว็บไซต์ฯและเจ้งพบของหาย ได้นำเว็บดังกล่าว มาทดสอบระบบตามฟังก์ชันที่มีในระบบ มีดังนี้

test case 1	ค้นหาของหาย
Relationship	UC-1
description	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากร,ผู้ดูแลระบบและผู้ใช้งานทั่วไปที่ต้องการจะหาของหาย
pre-condition	-
post-condition	สามารถค้นหาของหายผ่าน Filter search ได้ หรือไม่กรอก Filter search ก็สามารถเลื่อนหาได้ และเห็นโพสต์จริง
result	ผ่านการทดสอบ

Case	Actor	test data				ผลการแสดง	Result		
		filter search							
		ค้นหา	หมวดหมู่	สถานที่	สถานะ				
1	ค้นหาแบบไม่กรอก filter	-	-	-	-	เห็นโพสต์ทั้งหมด	ผ่านการทดสอบ		
2	ค้นหาผ่านการพิมพ์ ที่กล่องค้นหาใน filter search	ipad	-	-	-	เฉพาะโพสต์ที่ เป็น ipad	ผ่านการทดสอบ		
3	ค้นหาผ่านการเลือก หมวดหมู่	-	อุปกรณ์ อิเล็กทรอนิกส์	-	-	เฉพาะโพสต์ที่ เป็นอุปกรณ์ อิเล็กทรอนิกส์	ผ่านการทดสอบ		
4	ค้นหาผ่านการเลือก สถานที่	-	-	sc3	-	เฉพาะโพสต์ที่ เป็น sc3	ผ่านการทดสอบ		
5	ค้นหาผ่านการเลือก สถานะ	-	-	-	ไม่อยู่ในคลัง	เฉพาะโพสต์ที่ไม่ อยู่ในคลัง	ผ่านการทดสอบ		
6	ค้นหาผ่านการเลือก ทั้งหมด	ipad	อุปกรณ์ อิเล็กทรอนิกส์	sc3	ไม่อยู่ในคลัง	เฉพาะโพสต์ที่ ตรงตามที่กรอก ทั้งหมด	ผ่านการทดสอบ		

ตารางที่ 4.4 ตารางทดสอบการใช้งานค้นหาของหาย

test case 2	เข้าสู่ระบบ
Relationship	UC-2
description	นักศึกษาหรือบุคลากรมหาลัยธรรมศาสตร์ รังสิต ที่มีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
post-condition	ต้องเข้าสู่ระบบได้จริง
result	ผ่านการทดสอบ

case	actor	test data	ผลการแสดง	result
1	ผู้ใช้ที่เป็นนักศึกษา หรือบุคลากรใน มหาวิทยาลัย เข้าสู่ ระบบ	กรอกรหัสนัก ศึกษาและเลข บัตรประชาชน	เข้าสู่หน้าหลักหลังเข้าสู่ระบบ	ผ่านการทดสอบ
2	ผู้ดูแลเข้าสู่ระบบ	กรอกเลขผู้ ดูแลและรหัส ผ่าน	เข้าสู่หน้าหลักผู้ดูแลหลังเข้าสู่ระบบ	ผ่านการทดสอบ
3	ผู้ใช้นักศึกษาหรือ บุคลากรใน มหาวิทยาลัย ข้าสู่ ระบบ	กรอกเลขรหัส นักศึกษาผิด	ต้องขึ้นว่าusernameหรือpassword ผิด	ผ่านการทดสอบ
4	ผู้ใช้นักศึกษาหรือ บุคลากรใน มหาวิทยาลัย ข้าสู่ ระบบ	กรอกเลขรหัส ผ่านเลขบัตร ประชาชนผิด	ต้องขึ้นว่าusernameหรือpassword ผิด	ผ่านการทดสอบ
5	ผู้ดูแลเข้าสู่ระบบ	กรอกเลขผู้ดูแล ผิด	ต้องขึ้นว่าusernameหรือpassword ผิด	ผ่านการทดสอบ
6	ผู้ดูแลเข้าสู่ระบบ	กรอกรหัสผ่าน ผิด	ต้องขึ้นว่าusernameหรือpassword ผิด	ผ่านการทดสอบ

ตารางที่ 4.5 ตารางการเข้าใช้งานเข้าสู่ระบบ

test case 3	โพสต์แจ้งพบของหาย
Relationship	UC-3
description	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากร,ผู้ดูแลระบบ ต้องการโพสต์แจ้งพบของหาย
pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
post-condition	ต้องโพสต์แจ้งพบของหายได้สำเร็จ
result	ผ่านการทดสอบ

case	actor	test data	ผลการแสดง	result
1 .	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรในมหาวิทยาลัยแจ้งพบของหาย	กรอกรายละเอียดของโพสต์แจ้งของหายดังกล่าว	โพสต์ต้องปรากฏที่หน้าหลัก และหน้าโพสต์ของฉัน	ผ่านการทดสอบ
2 .	ผู้ดูแลแจ้งพบของหาย	กรอกรายละเอียดของโพสต์แจ้งของหายดังกล่าว	โพสต์ต้องปรากฏที่หน้าหลักและหน้าโพสต์ของฉัน	ผ่านการทดสอบ
3.	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรในมหาวิทยาลัยแจ้งพบของหาย	ไม่กรอกรายละเอียดสักอย่างของโพสต์แจ้งหายดังกล่าว	จะต้องแจ้งว่าต้องกรอกทุกอันด้วย	ผ่านการทดสอบ
4.	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรในมหาวิทยาลัยแจ้งพบของหาย	ไม่กรอกแค่ช่องรายละเอียดของโพสต์แจ้งหายดังกล่าว	โพสต์ต้องปรากฏที่หน้าหลักและหน้าโพสต์ของฉัน	ผ่านการทดสอบ
5.	ผู้ดูแลแจ้งพบของหาย	ไม่กรอกรายละเอียดสักอย่างของโพสต์แจ้งหายดังกล่าว	จะต้องแจ้งว่าต้องกรอกทุกอันด้วย	ผ่านการทดสอบ
6.	ผู้ดูแลแจ้งพบของหาย	ไม่กรอกแค่ช่องรายละเอียดของโพสต์แจ้งหายดังกล่าว	โพสต์ต้องปรากฏที่หน้าหลักและหน้าโพสต์ของฉัน	ผ่านการทดสอบ

ตารางที่ 4.6 ตารางทดสอบการใช้งานโพสต์แจ้งพบของหาย

Test case 4	แก้ไขโพสต์แจ้งพบของหาย
Relationship	UC-4
Description	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรและผู้ดูแล สามารถที่จะแก้ไขโพสต์แจ้งพบของหายได้
Pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
Post-condition	แก้ไขข้อมูลในโพสต์ได้จริง
Result	ผ่านการทดสอบ

Case	Actor	Test data	ผลการแสดง	Result
1.	ผู้ใช้ที่เป็นนักศึกษา หรือบุคลากรในมหาวิทยาลัยแก้ไข โพสต์แจ้งพบของหาย	แก้ไขรายละเอียดของโพสต์หายดังกล่าว	โพสต์ที่แก้ไขจะต้องอัพเดตและปรากฏที่หน้าหลัก และหน้า โพสต์ ของฉัน	ผ่านการทดสอบ
2 .	ผู้ดูแลแก้ไขโพสต์แจ้งพบของหาย	แก้ไขรายละเอียดของโพสต์หายดังกล่าว	โพสต์ที่แก้ไขจะต้องอัพเดตและปรากฏที่หน้าหลัก และหน้า โพสต์ ของฉัน	ผ่านการทดสอบ
3.	ผู้ใช้ที่เป็นนักศึกษา หรือบุคลากรในมหาวิทยาลัยแก้ไข โพสต์แจ้งพบของหาย	ไม่แก้ไขอะไร เลยแล้วกด อัปเดต	โพสต์ที่อัพเดตไป จะหน้าตาไม่เปลี่ยนแปลง	ผ่านการทดสอบ
4.	ผู้ดูแลแก้ไขโพสต์แจ้งพบของหาย	ไม่แก้ไขอะไร เลยแล้วกด อัปเดต	โพสต์ที่อัพเดตไป จะหน้าตาไม่เปลี่ยนแปลง	ผ่านการทดสอบ

ตารางที่ 4.7 ตารางทดสอบการใช้งานแก้ไขโพสต์แจ้งพบของหาย

Test case 5	ตรวจสอบผู้ใช้ในระบบ
Relationship	UC-5
Description	ผู้ใช้ที่เป็นผู้ดูแลระบบสามารถที่จะตรวจสอบโพสต์ของผู้ใช้งานที่เป็นนักศึกษาได้
Pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
Post-condition	สามารถตรวจสอบโพสต์ของผู้ใช้งานที่เป็นนักศึกษาได้
Result	ผ่านการทดสอบ

Case	Actor	Test data	ผลการแสดง	Result
1	ผู้ดูแลระบบแก้ไขสถานะโพสต์ที่ 1	กดดูหนังในโพสต์ของผู้ใช้รายนี้	จะต้องแสดงโพสต์เฉพาะโพสต์ของผู้ใช้รายนี้ออกมาก่อน	ผ่านการทดสอบ
2	ผู้ดูแลระบบแก้ไขสถานะโพสต์ที่ 2	กดดูหนังในโพสต์ของผู้ใช้รายนี้	จะต้องแสดงโพสต์เฉพาะโพสต์ของผู้ใช้รายนี้ออกมาก่อน	ผ่านการทดสอบ

ตารางที่ 4.8 ตารางทดสอบการใช้งานตรวจสอบผู้ใช้ระบบ

Test case 6	แก้ไขสถานะโพสต์แจ้งพบของหาย
Relationship	UC-6
Description	นักศึกษาหรือบุคลากรในมหาวิทยาลัย และผู้ดูแลระบบสามารถที่จะแก้ไขสถานะโพสต์แจ้งพบของหายได้
Pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
Post-condition	สามารถที่จะแก้ไขสถานะโพสต์แจ้งพบของหายได้
Result	ผ่านการทดสอบ

Case	Actor	Test data	ผลการแสดง	Result
1.	ผู้ดูแลระบบตรวจสอบผู้ใช้คนแรก	เปลี่ยนจากสถานะไม่มีอยู่ในคลังเป็นอยู่ในคลัง	สถานะโพสต์ถูกเปลี่ยนเป็นอยู่ในคลัง	ผ่านการทดสอบ
2.	ผู้ดูแลระบบตรวจสอบผู้ใช้คนที่สอง	เปลี่ยนจากสถานะอยู่ในคลังเป็นถูกรับไปแล้ว	สถานะโพสต์ถูกเปลี่ยนเป็นถูกรับไปแล้ว	ผ่านการทดสอบ
4.	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรในมหาวิทยาลัยแก้ไขสถานะของโพสต์ตัวเอง	เปลี่ยนจากสถานะไม่มีอยู่ในคลังเป็นถูกรับไปแล้ว	สถานะโพสต์ถูกเปลี่ยนเป็นถูกรับไปแล้ว	ผ่านการทดสอบ
5.	ผู้ใช้ที่เป็นนักศึกษาหรือบุคลากรในมหาวิทยาลัยแก้ไขสถานะของโพสต์ตัวเอง	เปลี่ยนจากสถานะถูกรับไปแล้วเป็นไม่มีอยู่ในคลัง	สถานะโพสต์ถูกเปลี่ยนเป็นไม่มีอยู่ในคลัง	ผ่านการทดสอบ

ตารางที่ 4.9 ตารางทดสอบการใช้งานแก้ไขสถานะ

test case 7	ดูเดชบอร์ด
Relationship	UC-7
description	ผู้ดูแลสามารถที่จะดูภาพรวมของระบบผ่านหน้าแดชบอร์ดได้
pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์
post-condition	สามารถที่จะดูภาพรวมของทุกอย่างได้
result	ผ่านการทดสอบ

case	actor	test data	ผลการแสดง	result
1 .	ผู้ดูแลระบบต้องการดูเดชบอร์ด	กดปุ่ม”เดชบอร์ด”	แสดงหน้าเดชบอร์ด	ผ่านการทดสอบ

ตารางที่ 4.10 ตารางทดสอบการใช้งานดูเดชบอร์ด

test case 8	ลบโพสต์
Relationship	UC-8
description	ผู้ดูแลระบบหรือนักศึกษาที่เป็นเจ้าของโพสต์ สามารถที่จะลบโพสต์ได้
pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์ รังสิต
post-condition	สามารถที่จะลบโพสต์ได้
result	ผ่านการทดสอบ

Case	Actor	Test data	ผลการแสดง	Result
1 .	ผู้ใช้ที่เป็นนักศึกษา ต้องการลบโพสต์ ของตัวเอง	กดปุ่มลบโพสต์ ตั้งกล่าว	โพสต์ดังกล่าวจะ ต้องไม่ปรากฏ ออกรมา	ผ่านการทดสอบ
2.	ผู้ดูแลระบบ ต้องการลบโพสต์ ของตัวเอง	กดปุ่มลบโพสต์ ตั้งกล่าว	โพสต์ดังกล่าวจะ ต้องไม่ปรากฏ ออกรมา	ผ่านการทดสอบ
3.	ผู้ดูแลระบบ ต้องการลบโพสต์ ของผู้ใช้คนอื่น	กดปุ่มลบโพสต์ ตั้งกล่าว	โพสต์ดังกล่าวจะ ต้องไม่ปรากฏ ออกรมา	ผ่านการทดสอบ

ตารางที่ 4.11 ตารางทดสอบการใช้งานลบโพสต์

test case 9	จัดการโพสต์
Relationship	UC-9
description	ผู้ดูแลระบบสามารถที่จะจัดการโพสต์ของผู้ใช้งานทุกคนได้ โดยจะสามารถแก้ไข ลบโพสต์ และดูสถานะหรือคุறำรายละเอียดโพสต์ทั้งหมดได้
pre-condition	ต้องมีข้อมูลอยู่ภายใน TU APIs ของมหาวิทยาลัยธรรมศาสตร์ รังสิต
post-condition	สามารถที่จะดูรายละเอียดของโพสต์ได้,แก้ไขและลบโพสต์ได้
result	ผ่านการทดสอบ

Case	Actor	Test data	ผลการแสดง	Result
1 .	ผู้ดูแลที่เป็นนักศึกษาต้องการลบโพสต์หนึ่ง	กดปุ่มลบโพสต์ดังกล่าว	โพสต์ดังกล่าวจะต้องไม่ปรากฏอีก	ผ่านการทดสอบ
2.	ผู้ดูแลระบบต้องการแก้ไขโพสต์หนึ่ง	กดปุ่มแก้ไขโพสต์ดังกล่าว	โพสต์ดังกล่าวจะต้องถูกอัพเดต	ผ่านการทดสอบ
3.	ผู้ดูแลระบบต้องการดูรายละเอียดของโพสต์หนึ่ง	กดปุ่มสถานะเพื่อดูรายละเอียด	โพสต์ดังกล่าวจะแสดงรายละเอียดอีกครั้ง	ผ่านการทดสอบ

ตารางที่ 4.12 ตารางทดสอบการใช้งานจัดการโพสต์

บทที่ 5 สรุป

5.1 สรุปผลการดำเนินงาน

โครงการ “เว็บแอปพลิเคชันแจ้งของหายภัยในมหาวิทยาลัยธรรมศาสตร์” เป็นโครงการที่มุ่งเน้นแก้ปัญหาการค้นหาสิ่งของสูญหายซึ่งเป็นปัญหาที่เกิดขึ้นบ่อยในพื้นที่มหาวิทยาลัย โดยเฉพาะในมหาวิทยาลัยธรรมศาสตร์ ศูนย์รังสิต ซึ่งมีจำนวนบุคลากร นักศึกษา และบุคคลภายนอกเข้าออกเป็นจำนวนมากในแต่ละวัน ไม่ว่าจะเป็นการเข้ามาเรียน ทำงาน หรือร่วมกิจกรรมต่าง ๆ ซึ่งอาจเกิดกรณีที่ผู้ใช้บริการพื้นที่ลืมสิ่งของไว้โดยไม่ได้ตั้งใจ และมักประสบปัญหาไม่ทราบว่าจะต้องดำเนินการอย่างไรในการค้นหาสิ่งของเหล่านั้น ทีมผู้จัดทำจึงพัฒนาโครงการนี้เพื่อสร้างระบบที่ช่วยอำนวยความสะดวกให้แก่ผู้ที่ทำสิ่งของสูญหายหรือผู้ที่พบสิ่งของและต้องการคืนให้เจ้าของ

ผลการดำเนินงานของโครงการนี้ประสบความสำเร็จในการพัฒนาแพลตฟอร์มเว็บแอปพลิเคชันที่ทำได้ตามวัตถุประสงค์ที่ทีมผู้พัฒนาได้กำหนดไว้รวมถึงครอบคลุมการใช้งานในกรณีต่าง ๆ ที่เกี่ยวข้องกับของหาย เช่น การแจ้งพบสิ่งของ การค้นหารายการสิ่งของที่มีผู้แจ้งพบ การแก้ไขโพสต์แจ้งของหาย และการตรวจสอบสถานะของโพสต์ต่าง ๆ ระบบได้รับการออกแบบมาให้ใช้งานง่าย มีส่วนติดต่อผู้ใช้ที่ชัดเจน และสามารถรองรับการใช้งานของบุคลากรมหาวิทยาลัยธรรมศาสตร์ โดยใช้บริการ TU APIs Developers ในการยืนยันตัวตนของผู้ใช้

ในโครงการนี้ได้มีการนำ Leaflet ซึ่งเป็นไลบรารีสำหรับแผนที่แบบโอเพนซอร์ส มาใช้เพื่อเพิ่มประสิทธิภาพของเว็บแอปพลิเคชันในการแจ้งตำแหน่งของสิ่งของที่สูญหายและพบเจอ ผู้ใช้งานสามารถปักหมุดตำแหน่งลงบนแผนที่เพื่อบุสตันที่ที่เกี่ยวข้องกับการแจ้งข้อมูลสิ่งของ โดยการพัฒนานี้ช่วยให้การค้นหาและการส่งคืนสิ่งของเป็นไปอย่างแม่นยำและสะดวกยิ่งขึ้น

ทีมผู้พัฒนาได้ทำการทดลอง Deploy ระบบเว็บแอปพลิเคชันลงบนแพลตฟอร์ม Vercel เพื่อให้ระบบสามารถใช้งานได้ในรูปแบบออนไลน์และรองรับการเข้าถึงจากผู้ใช้ปลายทางได้อย่างมีประสิทธิภาพ

จากการดำเนินงานดังกล่าว ระบบเว็บแอปพลิเคชันนี้สามารถช่วยลดความยุ่งยากในการค้นหาสิ่งของที่สูญหาย และเพิ่มโอกาสให้เจ้าของได้รับสิ่งของคืนในระยะเวลาอันรวดเร็ว อีกทั้งยังเป็นตัวกลางที่สร้างการสื่อสารระหว่างผู้ที่พบสิ่งของและเจ้าของได้อย่างมีประสิทธิภาพ

5.2 สิ่งที่ได้เรียนรู้จากการทำงาน

5.2.1 การพัฒนาระบบเว็บแอปพลิเคชัน

- 1) ได้เรียนรู้เกี่ยวกับกระบวนการพัฒนาเว็บแอปพลิเคชันตั้งแต่การวางแผน, ออกแบบระบบ และการเขียนโค้ด โดยเฉพาะการใช้เฟรมเวิร์กและไลบรารีต่าง ๆ เช่น React และ Leaflet.js ซึ่งช่วยให้การพัฒนาแอปพลิเคชันมีความรวดเร็วและตอบโจทย์การใช้งานที่หลากหลาย
- 2) ได้เรียนรู้การจัดการกับข้อมูล เช่น การออกแบบฐานข้อมูล การเชื่อมต่อระหว่าง Backend และ Frontend ตลอดจนการจัดการข้อมูลตำแหน่ง (Geolocation) เพื่อให้ระบบสามารถทำงานได้อย่างราบรื่น

5.2.2 การใช้ API และการยืนยันตัวตน

1. ได้ศึกษาการใช้ **TU APIs Developers** เพื่อยืนยันตัวตนของผู้ใช้งาน ซึ่งช่วยเพิ่มความปลอดภัยและความน่าเชื่อถือในระบบ โดยได้เรียนรู้การส่งคำขอ (request) และจัดการกับการตอบสนอง (response) จาก API
2. เข้าใจความสำคัญของการปกป้องข้อมูลผู้ใช้ และแนวทางการจัดการ Environment Variables เพื่อป้องกันข้อมูลสำคัญจาก การเข้าถึงโดยไม่ได้รับอนุญาต

5.2.3 การจัดการ Deployment บนแพลตฟอร์ม Vercel

1. ได้เรียนรู้กระบวนการ Deploy ระบบขึ้นโฮสต์ออนไลน์โดยใช้ Vercel ตั้งแต่การตั้งค่าโปรเจกต์ การจัดการ Branch ใน GitHub เพื่อการ Build อัตโนมัติ และการแก้ไขปัญหาที่เกิดขึ้น หลังการ Deploy
2. ได้เข้าใจแนวทางการปรับปรุงประสิทธิภาพของระบบสำหรับ การใช้งานจริง เช่น การตรวจสอบความเร็วการโหลดหน้าเว็บ และการจัดการข้อผิดพลาดที่อาจเกิดขึ้น

5.3 แนวคิดการพัฒนา

1. ทำระบบ “แจ้งของหาย” ที่สามารถให้ผู้ใช้ที่ทำ สิ่งของหาย แจ้งโพสต์ของหาย ไว้ได้
2. ทำระบบ Matching ระหว่าง “แจ้งพบของหาย” และ “แจ้ง ของหาย” ที่จะสามารถทำให้ผู้ใช้ค้นหาของหายที่ลักษณะ เหมือนกับสิ่งของที่ผู้ใช้ได้ทำการแจ้งไว้

รายการอ้างอิง

1. Patchara Boonmathanaruk. (2565). Figma คืออะไร? ทำไมถึงเป็น Tool มาแรงที่สุดในการ Design! . สืบค้น 15 เมษายน 2567, จาก <https://blog.skooldio.com/figma-ui-design-tool/>
2. Mindphp. (2560). รู้จัก Visual Studio code . สืบค้น 15 เมษายน 2567 , จาก <https://www.mindphp.com/%E0%B8%9A%E0%B8%97%E0%B8%84%E0%B8%A7%E0%B8%B2%E0%B8%A1/microsoft/4829-visual-studio-code.html>
3. Charintorn (Nut) Ruanglaikram. (2566). Tailwind CSS เพื่อเวิร์กที่ช่วยให้ Dev ทำงานง่ายขึ้น. สืบค้น 15 เมษายน 2567 , จาก <https://morphos.is/th/blog/tailwind-css-a-framework-that-makes-dev-work-easier>
4. Sirasit Boonklan . (2567).เขียน API ด้วย Next.js จะบันเร็วจนไฟลุก. สืบค้น 15 เมษายน 2567 , จาก <https://www.borntodev.com/2024/02/28/%E0%B9%80%E0%B8%82%E0%B8%B5%E0%B8%A2%E0%B8%99-api-%E0%B8%94%E0%B9%89%E0%B8%A7%E0%B8%A2-next-js/>
5. Kanthima Muangsen. (2564). ทำไมถึงต้องใช้ Typescript หั้งที่ก็มี JavaScript ออยู่แล้ว . สืบค้น 15 เมษายน 2567 จาก <https://www.borntodev.com/2021/05/13/%E0%B8%97%E0%B8%B3%E0%B9%84%E0%B8%A1%E0%B8%96%E0%B8%B6%E0%B8%87%E0%B8%95%E0%B9%89%E0%B8%AD%E0%B8%87%E0%B9%83%E0%B8%8A%E0%B9%89-typescript/>
6. BorntoDev Co., Ltd. (2564).React คืออะไร ? . สืบค้น 17 เมษายน 2567 , จาก <https://www.borntodev.com/2020/07/15/react-101/>

7. Kimiiz Pongnares . (2561). [Prisma-101] สร้าง GraphQL Sever แบบขี้เกียจด้วย Prisma.
สืบค้น 17 เมษายน 2567 , จาก
<https://medium.com/@kimiiz/prisma-101-สร้าง-graphql-sever-แบบขี้เกียจด้วย-prisma-6fea3cd98932#:~:text=Prisma%20คือ%20GraphQL%20database%20proxy.> ด้วย%20Prisma%20กันดีกว่า
 8. AppMaster . (2565).PostgreSQL คือ อะไร ? โปรแกรมจัดการฐานข้อมูล Open Source ยอดนิยม !. สืบค้น 17 เมษายน 2567 , จาก <https://appmaster.io/th/blog/postgresql-khuue-aair>
 9. Amazon . (2562).Docker คืออะไร?. สืบค้น 25 เมษายน 2567 , จาก <https://aws.amazon.com/th/docker/>
 10. Thanatip Suwanjandee. (2561).Google Cloud คืออะไร (ฉบับมือใหม่).สืบค้น 25 เมษายน 2567 , จาก
<https://cloud-ace.co.th/blogs/o6t7k4-google-cloud>
 11. Leafletjs . (2565.).ไลบรารี JavaScript โอเพนซอร์ส. สืบค้น 2 ตุลาคม 2567 , จาก
https://leafletjs-com.translate.goog/?_x_tr_sl=en&_x_tr_tl=th&_x_tr_hl=th&_x_tr_pto=tc
 12. BornToDev . (2561). การแสดงผลเว็บไซต์ด้วยGoogleMap.
สืบค้น 2 ตุลาคม 2567, จาก
<https://www.bornstodev.com/2023/11/14/google-map>
 13. จัสติน เกจ . (2566). vercel ทำอะไร?. สืบค้น 2 ตุลาคม 2567,
จาก
https://vercel-com.translate.goog/blog/what-is-vercel?_x_tr_s_l=en&_x_tr_tl=th&_x_tr_hl=th&_x_tr_pto=tc&_x_tr_hist=true

