

**IT 2567/ WEB15**

**เอกสารคู่มือการใช้งานระบบ**

**(User Manual)**

เว็บแอปพลิเคชันการบริหารและจัดการงานที่ได้รับมอบหมาย

Web application for Task Management

โดย

643020402-8 นาย ภพ ในจิตต์

643021105-9 นางสาว ชัญญา โทมี

อาจารย์ที่ปรึกษา: รองศาสตราจารย์ ดร.งามนิจ อาจอินทร์

รายงานนี้เป็นส่วนหนึ่งของการศึกษารายวิชา SC364772 โครงงานทางเทคโนโลยีสารสนเทศ 2

ภาคการศึกษาที่ 2 ปีการศึกษา 2567

หลักสูตรเทคโนโลยีสารสนเทศ สาขาวิชาวิทยาการคอมพิวเตอร์

วิทยาลัยการคอมพิวเตอร์ มหาวิทยาลัยขอนแก่น

(มีนาคม พ.ศ.2568)

**คำนำ**

เอกสารคู่มือการใช้งานแอปพลิเคชันการบริหารและจัดการงานที่ได้รับมอบหมายฉบับนี้เป็นส่วนหนึ่งของรายวิชา SC364771 โครงงานทางเทคโนโลยีสารสนเทศ 2 ชั้นปีที่ 4 โดยมีจุดประสงค์ เพื่อเพื่อใช้ในการประกอบการใช้งาน เพื่อให้ผู้ใช้ปฏิบัติตามขั้นตอนได้อย่างถูกวิธี และการศึกษาความรู้ที่ได้รับจากเรื่องการวิจัย และได้มีการนาความรู้ที่ได้เรียนมาประยุกต์ให้เกิดประโยชน์โดยในรายงานนี้มีเนื้อหาที่ประกอบไปด้วยความรู้เกี่ยวกับการเขียนวิจัย ตลอดจนการ ประยุกต์ใช้เนื้อหาจากที่เรียนมาด้วย

ผู้จัดทำโครงงานหวังเป็นอย่างยิ่งว่า โครงงานคอมพิวเตอร์ที่จัดทาจะเกิดประโยชน์แก่ผู้ที่สนใจ ให้ความรู้ และแนะแนวทางการศึกษา หวังว่ารายงานฉบับนี้จะให้ความรู้ และเป็นประโยชน์แก่ผู้อ่านทุกท่าน หากมีข้อเสนอแนะ หรือมีข้อผิดพลาดประการใด ทางผู้จัดทาขออภัยมา ณ ที่นี้ด้วย

ผู้จัดทำ

ภพ ในจิตต์

ชัญญา โทมี

**สารบัญ**

**หน้า**

คำนำ ก

สารบัญภาพ ข

สารบัญตาราง ค

บทที่ 1 บทนํา 1

1.1 สถาปัตยกรรมระบบ

1.2 เป้าหมายและขอบเขตของระบบ

1.3 โครงสร้างการใช้งาน (ถ้ามี)

บทที่ 2 การติดตั้งระบบ

2.1 (ตัวอย่าง) การติดตั้ง Web Server

2.2 (ตัวอย่าง) การติดตั้งฐานข้อมูล MySQL

2.3 (ตัวอย่าง) การติดตั้งเว็บแอปพลิเคชัน FinalProject บน Web Server

บทที่ 3 การใช้งานระบบ

3.1 (ตัวอย่าง) การใช้งานระบบสำหรับผู้ดูแลระบบ

3.2 (ตัวอย่าง) การใช้งานระบบสำหรับพนักงาน

3.3 (ตัวอย่าง) การใช้งานระบบสำหรับลูกค้า

บทที่ 4 บทสรุป

เอกสารอ้างอิง (ถ้ามี)

**สารบัญตาราง**

**สารบัญภาพ**

**บทที่ 1**

**บทนำ**

**1.1 สถาปัตยกรรมระบบ**

1.1.1 ส่วนหน้าของระบบ (Frontend)

(1) EJS (Embedded JavaScript Templates) เป็น template engine สำหรับเรนเดอร์ UI (User Interface) ฝั่งเซิร์ฟเวอร์

(2) CSS (Cascading Style Sheets) และ JavaScript สำหรับสไตล์และการโต้ตอบผู้ใช้

(3) Chart.js สำหรับการสร้างกราฟ

(4) Socket.IO เพื่อการสื่อสารแบบเรียลไทม์ระหว่าง client และ server

1.1.2 ส่วนหลังของระบบ (Backend)

(1) Node.js และ Express.js เป็น framework สำหรับพัฒนา RESTful API และแอปพลิเคชันฝั่งเซิร์ฟเวอร์

(2) express-session และ connect-mongo สำหรับ session management

(3) passport สำหรับระบบล็อกอินทั้งแบบธรรมดาและผ่านบัญชี Google

(4) connect-flash เพื่อแสดงข้อความแจ้งเตือนใน UI

(5) bcrypt สำหรับเข้ารหัสรหัสผ่านใช้

(6) multer สำหรับอัปโหลดไฟล์ เช่น รูปโปรไฟล์

1.1.3 ฐานข้อมูลที่ใช้ MongoDB และ Mongoose เป็นฐานข้อมูล NoSQL สำหรับจัดเก็บข้อมูลผู้ใช้และข้อมูลอื่น ๆ

**1.2 เป้าหมายและขอบเขต**

1.2.1 กลุ่มเป้าหมายของโครงงาน

อธิบายถึงกลุ่มเป้าหมายของโครงงาน

1.2.2 ขอบเขตของระบบ

อธิบายถึงขอบเขตของระบบ

**1.3 โครงสร้างการใช้งานหลัก**

อธิบายถึงโครงสร้างการใช้งานระบบ เช่น แผนผังเว็บไซต์ (Sitemap) ผังรายการ (Menu List) หรือ Conversational User Interface (CUI)

**บทที่ 2**

**การติดตั้งระบบ**

**2.1 การติดตั้ง Web Server**

2.1.2 ติดตั้ง Visual Studio Code (VS Code)

1. ดาวน์โหลดจาก https://code.visualstudio.com/ เลือกดาวน์โหลดตามระบบปฏิบัติการที่ใช้งาน

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเว็บไซต์ดาวน์โหลด Visual Studio Code

2. เลือกยอมรับเงื่อนไขและคลิกปุ่ม Next

รูปภาพประกอบด้วย ข้อความ, อิเล็กทรอนิกส์, ภาพหน้าจอ, ซอฟต์แวร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าข้อตกลงการใช้งาน Visual Studio Code

3.เลือกงานเพิ่มเติมที่ต้องการให้การตั้งค่าดำเนินการขณะติดตั้ง Visual Studio Code และคลิกปุ่ม Next

รูปภาพประกอบด้วย ข้อความ, อิเล็กทรอนิกส์, ภาพหน้าจอ, ซอฟต์แวร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเลือกการตั้งค่างานเพิ่มเติมขณะติดตั้ง Visual Studio Code

4. คลิกที่ Install เมื่อติดตั้งเสร็จสิ้นจะขึ้นให้คลิกที่ Finish

รูปภาพประกอบด้วย ข้อความ, อิเล็กทรอนิกส์, ภาพหน้าจอ, แสดง

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้ายืนยันติดตั้งโปรแกรม Visual Studio Code

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ระบบปฏิบัติการ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเสร็จสิ้นการติดตั้งโปรแกรม Visual Studio Code

2.1.2 ติดตั้งNode.js

1. ดาวน์โหลดจาก https://nodejs.org/en เลือกดาวน์โหลดตามระบบปฏิบัติการที่ใช้งาน

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเว็บไซต์ดาวน์โหลด Node.js

2. เปิดโปรแกรมที่ติดตั้งและคลิกที่ Next

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ระบบปฏิบัติการ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าติดตั้ง Node.js

3. เลือกยอมรับเงื่อนไขและคลิกปุ่ม Next และคลิกปุ่ม Next จนกว่าจะถึงหน้าติดตั้ง

รูปภาพประกอบด้วย ข้อความ, อิเล็กทรอนิกส์, ภาพหน้าจอ, ซอฟต์แวร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าข้อตกลงการใช้งาน Node.js

4. คลิกที่ Install เมื่อติดตั้งเสร็จสิ้นจะขึ้นให้คลิกที่ Finish

รูปภาพประกอบด้วย ข้อความ, อิเล็กทรอนิกส์, ภาพหน้าจอ, ซอฟต์แวร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้ายืนยันติดตั้งโปรแกรม Node.js

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ระบบปฏิบัติการ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเสร็จสิ้นการติดตั้งโปรแกรม Node.js

**2.2 การเชื่อมต่อฐานข้อมูล MongoDB**

1. ไปที่เว็บไซต์ mongodb.com/products/platform/atlas-database ทำการเข้าสู่ระบบ

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, หน้าเว็บ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเว็บไซต์ MongoDB

2. ทำการสร้างโปรเจกต์ใหม่โดยการคลิกที่ปุ่ม New Project

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, จำนวน, ตัวอักษร

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ โปรเจกต์ทั้งหมดใน MongoDB

3. กรอกชื่อโปรเจกต์ทำการคลิกที่ปุ่ม Next

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ตัวอักษร, ซอฟต์แวร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้ากรอกข้อมูลสร้างโปรเจกต์ MongoDB

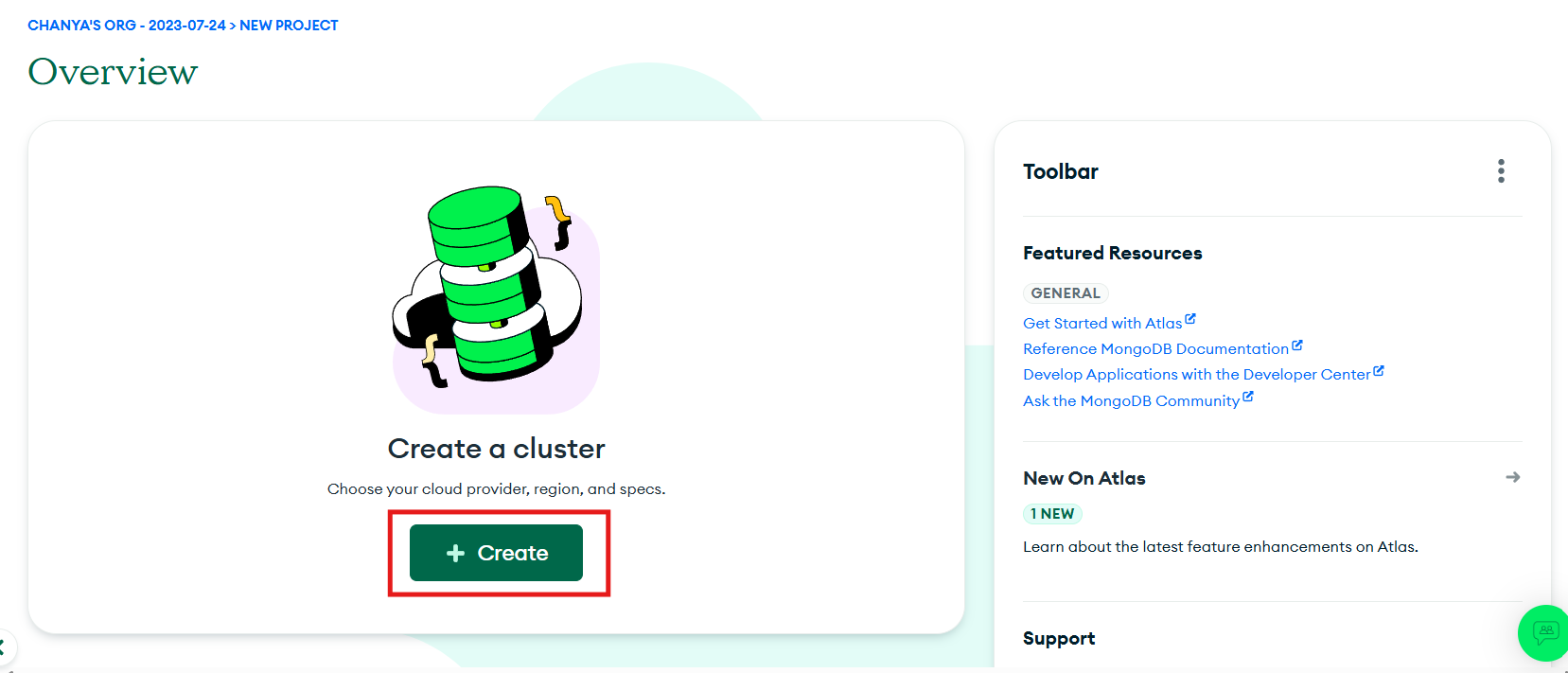
4. คลิกที่ปุ่ม Create Project

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ตัวอักษร, จำนวน

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าสร้างโปรเจกต์ MongoDB

5. คลิกที่ปุ่ม Create เพื่อสร้าง cluster ใหม่สำหรับเชื่อมต่อฐานข้อมูล

****

ภาพที่ หน้าสร้าง cluster ใหม่สำหรับเชื่อมต่อฐานข้อมูล MongoDB

6. เลือกแพลนการชำระเงินแบบฟรี กรอกชื่อ cluster และคลิกที่ปุ่ม Create deployment

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, หน้าเว็บ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้ากรอกข้อมูล cluster

7. สร้าง Username และ Password สำหรับเข้าใช้ Database หลังจากนั้น คลิก Create Database User

รูปภาพประกอบด้วย ข้อความ, อิเล็กทรอนิกส์, ภาพหน้าจอ, ตัวอักษร

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าสร้าง Username และ Password สำหรับเข้าใช้ Database

8. เชื่อมต่อ cluster คลิก Choose a connection method

รูปภาพประกอบด้วย ข้อความ, อิเล็กทรอนิกส์, ภาพหน้าจอ, ซอฟต์แวร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเชื่อมต่อ cluster

9. คลิกที่ connect และเลือก Drivers

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้า connect เพื่อเลือกตัวที่จะเชื่อมต่อ

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, จำนวน

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเลือกตัวที่จะเชื่อมต่อ

10. ทำการคัดลอกลิงก์สำหรับเชื่อมต่อฐานข้อมูล

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, จำนวน

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าข้อมูลเกี่ยวกับเชื่อมต่อฐานข้อมูล

11. ทำการเปิดโปรเจกต์ใน VS code เปิด Terminal และพิมพ์คำสั่ง npm install mongoose ใน Terminal

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์กราฟิก, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้า Terminal พิมพ์คำสั่งเพื่อติดตั้งฐานข้อมูล

12. สร้างไฟล์ app.js และ ลิงก์ที่จะเชื่อมฐานข้อมูลที่คัดลอกจากข้อที่ 10 วางลงในโค้ดดังภาพ และแทนที่ <username> และ <password> ด้วยข้อมูลจากข้อที่ 7

รูปภาพประกอบด้วย ภาพหน้าจอ, ซอฟต์แวร์มัลติมีเดีย, ซอฟต์แวร์, ข้อความ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าโค้ดสำหรับวางลิงก์ที่จะเชื่อมฐานข้อมูล

13. พิมพ์คำสั่ง npm install ตามด้วย nobe app.js ถ้าแสดงดังภาพ ถือว่าการเชื่อมต่อฐานข้อมูลสำเร็จ

รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ตัวอักษร

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้าเว็บไซต์พิมพ์คำสั่งเพื่อดูว่าเชื่อมกับฐานข้อมูล

**2.3 การติดตั้งเว็บแอปพลิเคชัน FinalProject บน Web Server โดยใช้ Render**

1. ไปที่เว็บไซต์ https://dashboard.render.com/login ทำการเข้าสู่ระบบ

**รูปภาพประกอบด้วย ภาพหน้าจอ, ข้อความ, ซอฟต์แวร์มัลติมีเดีย, ซอฟต์แวร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้าเว็บไซต์เข้าสู่ระบบ Render

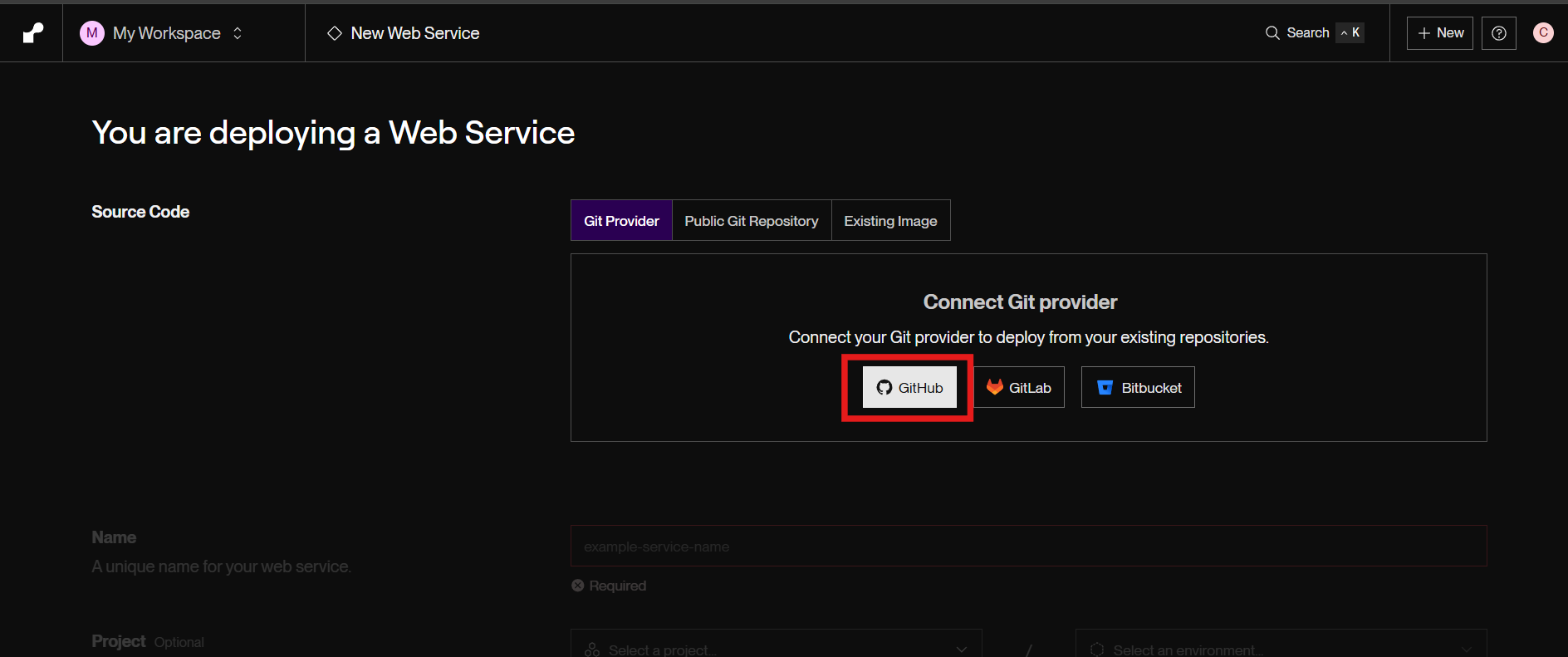
2. ทำการเพิ่มโปรเจกต์ใหม่โดย คลิกที่ปุ่ม Add new หลังจากนั้น คลิกเลือกที่ Web Service

**รูปภาพประกอบด้วย ภาพหน้าจอ, ข้อความ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้า Dashboard Render

3. เลือกเชื่อมต่อกับ GitHub

****

ภาพที่ หน้าเลือกเชื่อมต่อกับ GitHub

4. เลือกโปรเจกต์ที่ต้องการ deploy

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้าเลือกโปรเจกต์ที่ต้องการ deploy

5. ตั้งชื่อสำหรับ web service และเลือกแพลนชำระเงินเป็นแบบ Free และทำการเพิ่มตัวแปรสภาพแวดล้อมเช่น เพิ่มลิงก์เชื่อมต่อmongodb NAME\_OF\_VARIABLE คือชื่อของตัวแปรที่เราตั้งไว้ในโปรเจค เช่น MONGODB\_URI ส่วน Value คือ mongodb+srv://<dbuser>:<password>@cluster0.q9il

d.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0&tls=true

รูปภาพประกอบด้วย ภาพหน้าจอ, ข้อความ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

ภาพที่ หน้ากรอกข้อมูลเกี่ยวกับ web service

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้ากรอกข้อมูลเกี่ยวกับ web service(ต่อ)

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้ากรอกข้อมูลเกี่ยวกับ web service(ต่อ)

6.เมื่อ Deploy สำเร็จจะแสดงหน้าดังรูปภาพ โดยสามารถนำ URL ไปใช้ได้เลย

**รูปภาพประกอบด้วย ข้อความ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย, ซอฟต์แวร์กราฟิก

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้า Deploy สำเร็จ

**บทที่ 3**

**การใช้งานระบบ**

**3.1 (ตัวอย่าง) การใช้งานระบบสำหรับผู้ใช้**

อธิบายการใช้งานระบบสำหรับพนักงานในแต่ละฟังก์ชันการทำงาน

**รูปภาพประกอบด้วย ข้อความ, ซอฟต์แวร์, ระบบปฏิบัติการ, ไอคอนคอมพิวเตอร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

**รูปภาพประกอบด้วย ข้อความ, ซอฟต์แวร์, ระบบปฏิบัติการ, ไอคอนคอมพิวเตอร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

**รูปภาพประกอบด้วย ข้อความ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์, ภาพหน้าจอ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

**รูปภาพประกอบด้วย ข้อความ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์, จำนวน

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

รูปภาพประกอบด้วย ข้อความ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์, ระบบปฏิบัติการ

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง

**3.2 การใช้งานระบบสำหรับผู้ดูแลระบบ**

3.2.1 เข้าสู่ระบบเพื่อใช้งาน

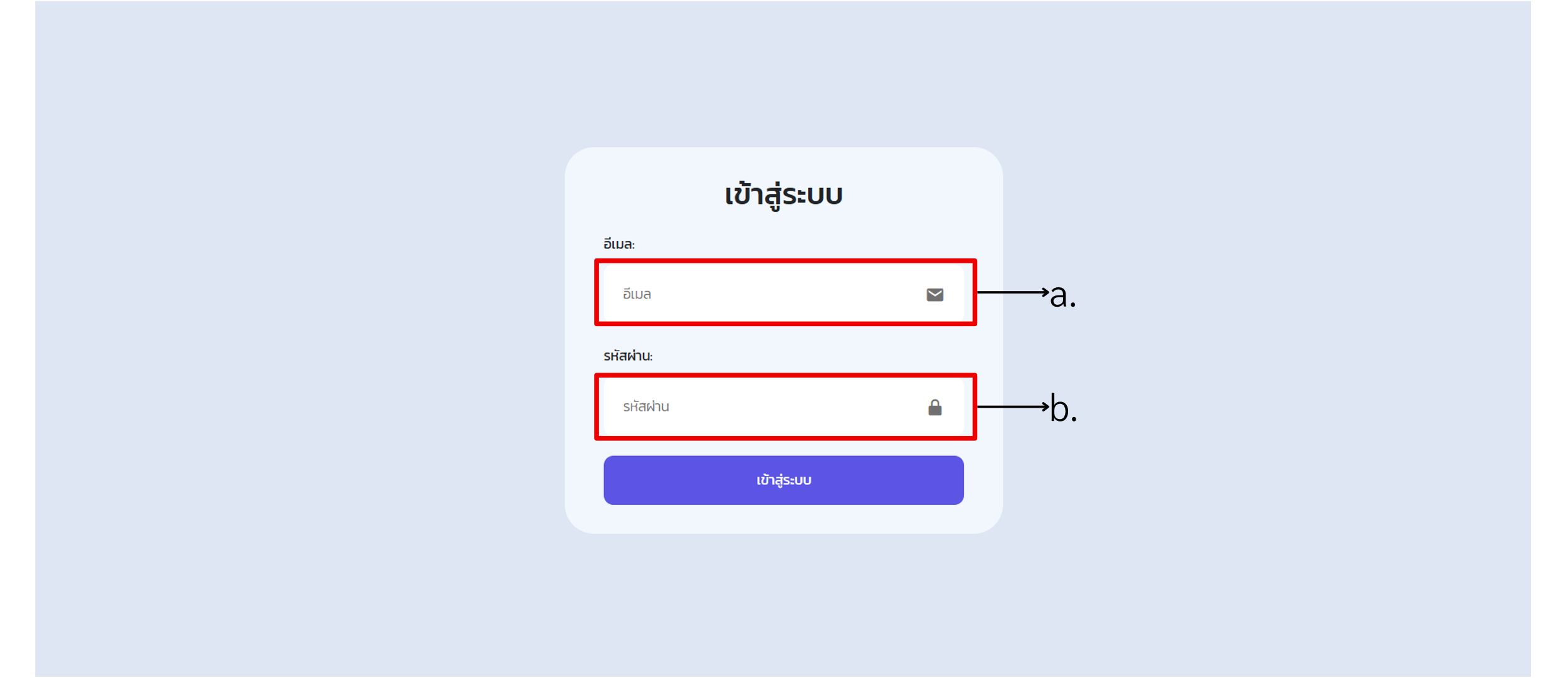
(1) หน้าแรกของผู้ดูแลระบบ คลิกที่ปุ่ม เข้าสู่ระบบ เพื่อไปที่หน้ากรอกข้อมูลการเข้าสู่ระบบ

(2) ผู้ดูแลระบบกรอกข้อมูล a.อีเมลผู้ดูแลระบบ b.รหัสผ่าน

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, การออกแบบกราฟิก

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้าแรกของผู้ดูแลระบบ



ภาพที่ หน้าแรกของผู้ดูแลระบบ

3.2.2 หน้าแดชบอร์ดของผู้ดูและระบบ

เมื่อผู้ดูแลระบบเข้าสู่ระบบจะเข้ามาหน้าแดชบอร์ดเป็นหน้าแรก

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้าแรกของผู้ดูแลระบบ

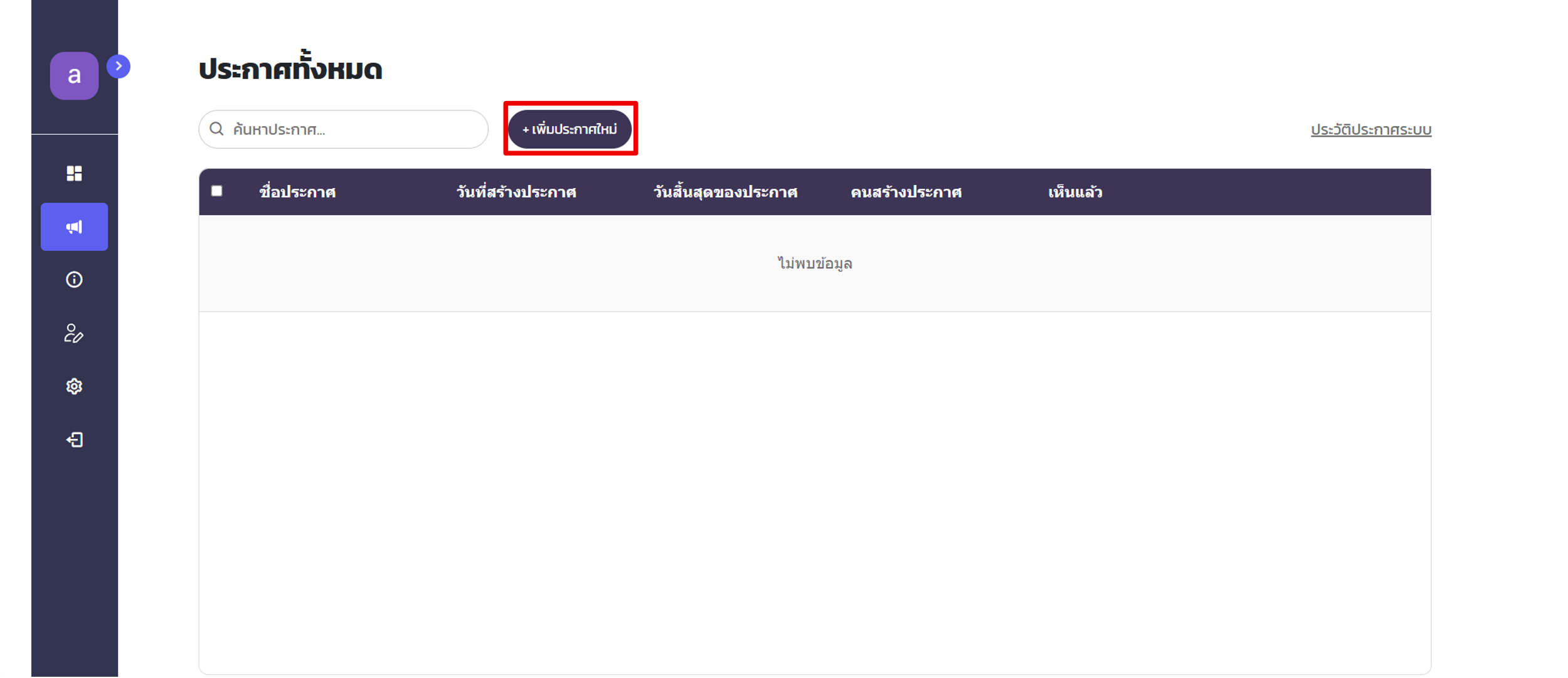
3.2.2 หน้าประกาศจากระบบ

(1) ผู้ดูแลระบบสร้างประกาศใหม่โดยทำการคลิกที่เพิ่มประกาศใหม่

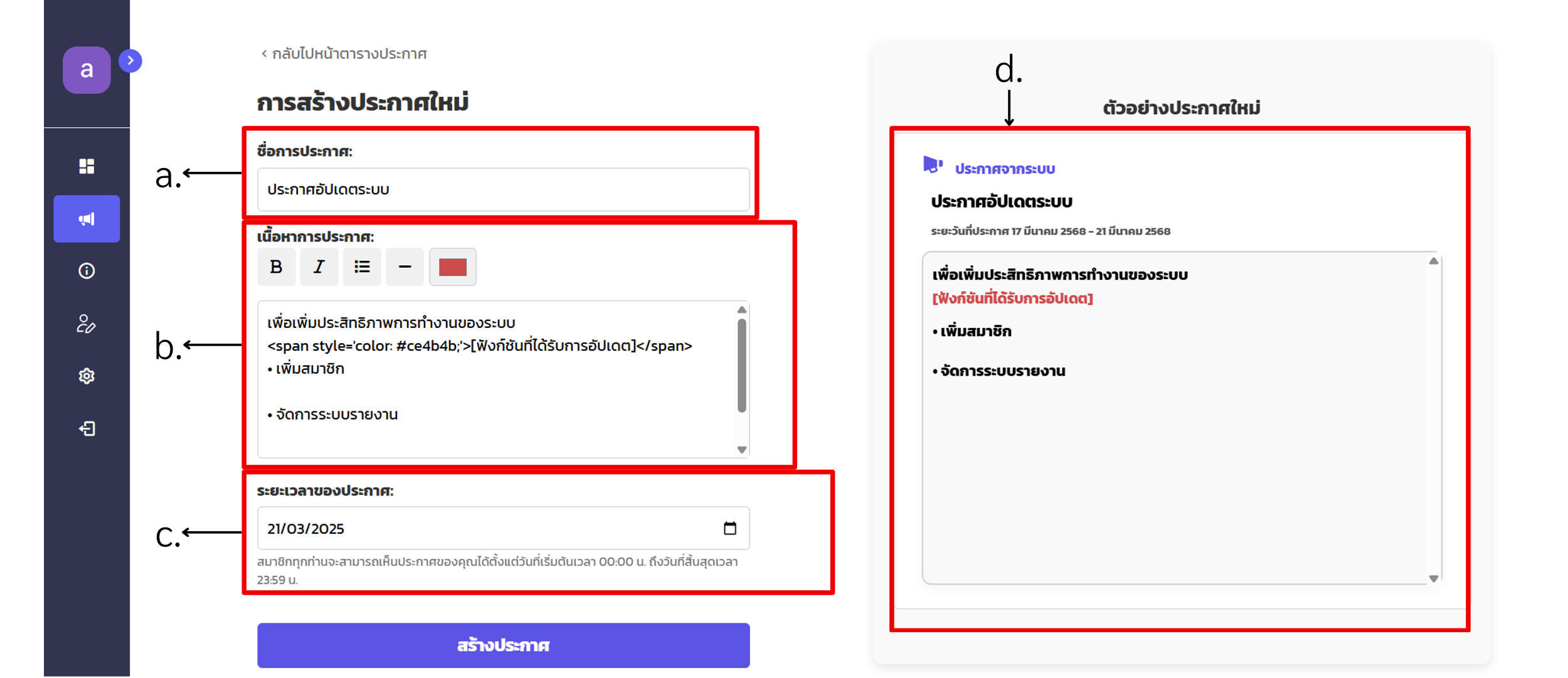
(2) ผู้ดูแลระบบกรอกข้อมูลเกี่ยวกับประกาศโดย a.ชื่อประกาศ b.รายละเอียดเนื้อหาโดยที่สามารถเพิ่มสี ตัวหนา เส้นแบ่งข้อความ และตัวเอียงเพื่อตกแต่งข้อความได้ c.เลือกเวลาสิ้นสุดประกาศ d.ตัวอย่างของประกาศที่จะนำไปแจ้งเตือนผู้ใช้ จากนั้นทำการคลิกที่ปุ่มสร้างประกาศเพื่อทำการสร้างประกาศ

(3) กรณีที่ประกาศหมดอายุหรือถูกลบผู้ดูแลระบบสามารถดูประวัติการประกาศได้โดยคลิกที่ ประวัติประกาศระบบ

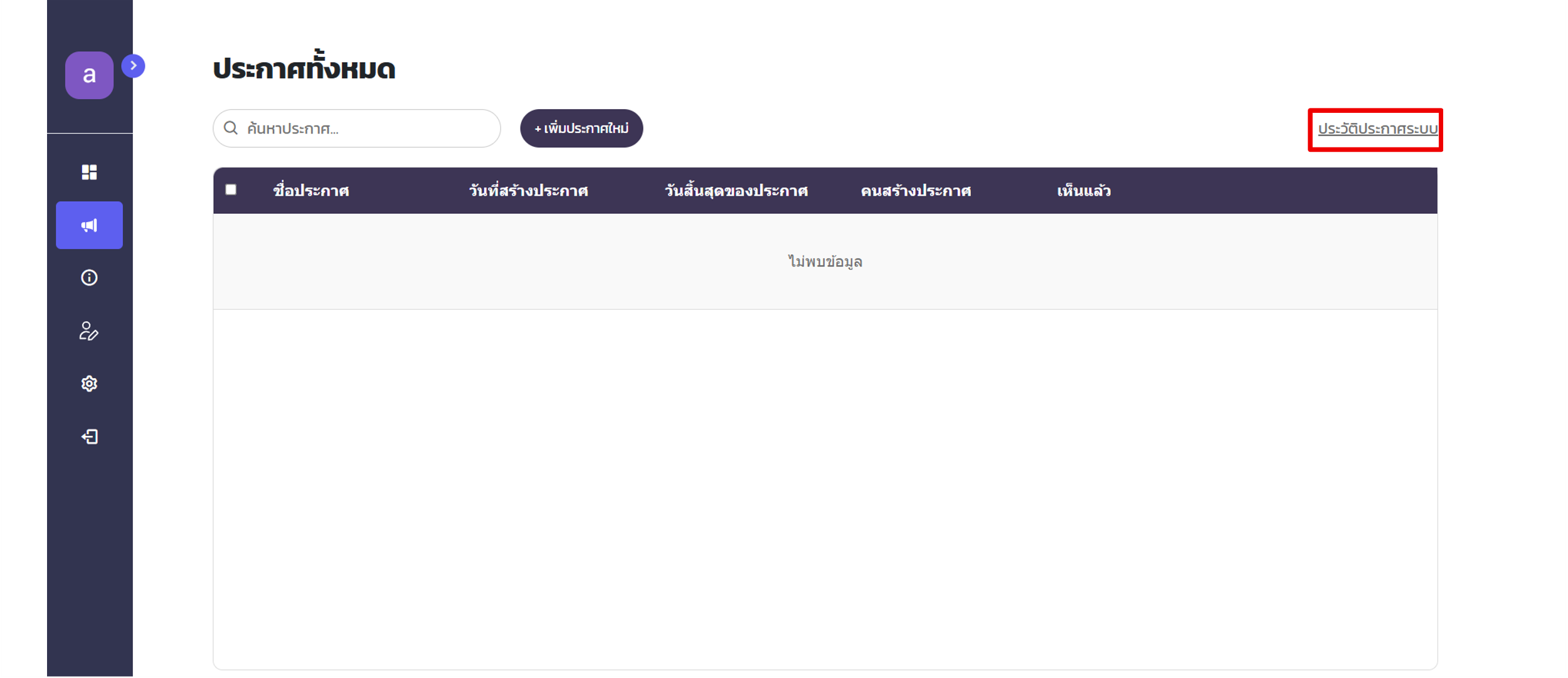
(4) ผู้ดูแลระบบดูประวัติการประกาศทั้งหมด



ภาพที่ หน้าประกาศทั้งหมด



ภาพที่ หน้าสร้างประกาศ



ภาพที่ หน้าประกาศทั้งหมดดูประวัติโดยคลิกที่ประวัติประกาศระบบ

**รูปภาพประกอบด้วย ภาพหน้าจอ, ข้อความ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

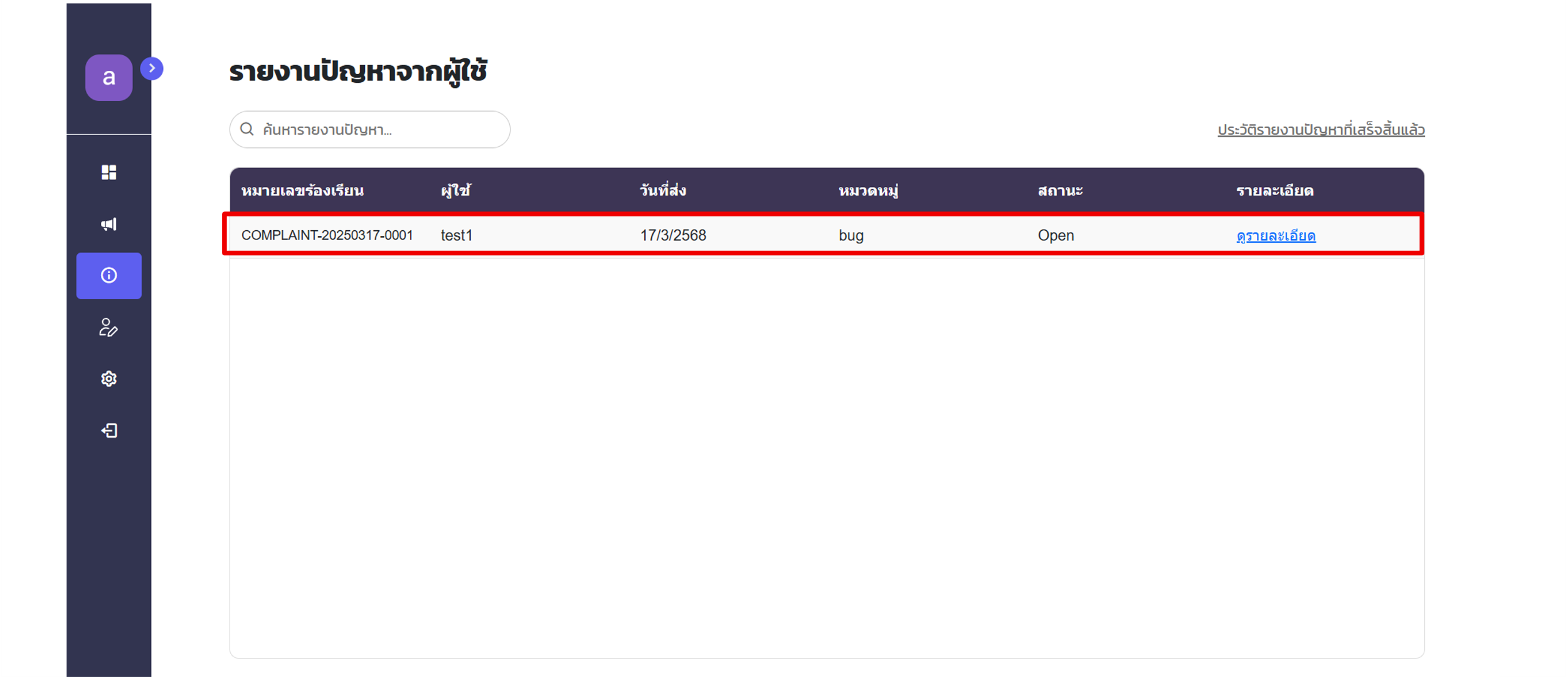
ภาพที่ หน้าประวัติการประกาศทั้งหมด

3.2.3 หน้ารายงานปัญหาจากผู้ใช้

(1) ผู้ดูแลระบบคลิกไปที่เมนูรายงานปัญหาผู้ใช้จะแสดงรายงานทั้งหมดที่ผู้ใช้ส่งมาให้ผู้ดูแลระบบทำการคลิกดูรายละเอียด เพื่อดูรายละเอียดของรายงานผู้ใช้

(2) ผู้ดูแลระบบเปลี่ยนสถานะระบบเพื่อส่งแจ้งเตือนไปให้ผู้ใช้ทราบว่าตอนนี้รายงานผู้ใช้มีความคืบหน้าอย่างไรบ้าง และเมื่อสถานะเปลี่ยนเป็นเสร็จสิ้นจะถือว่าจบการรายงานนี้แล้ว

(3) ผู้ดูแลระบบสามารถดูประวัติการรายงานได้โดยคลิกที่ ประวัติรายงานปัญหาที่สิ้นสุดแล้วและจะแสดงตารางที่โชว์รายงานที่สิ้นสุดไปแล้ว

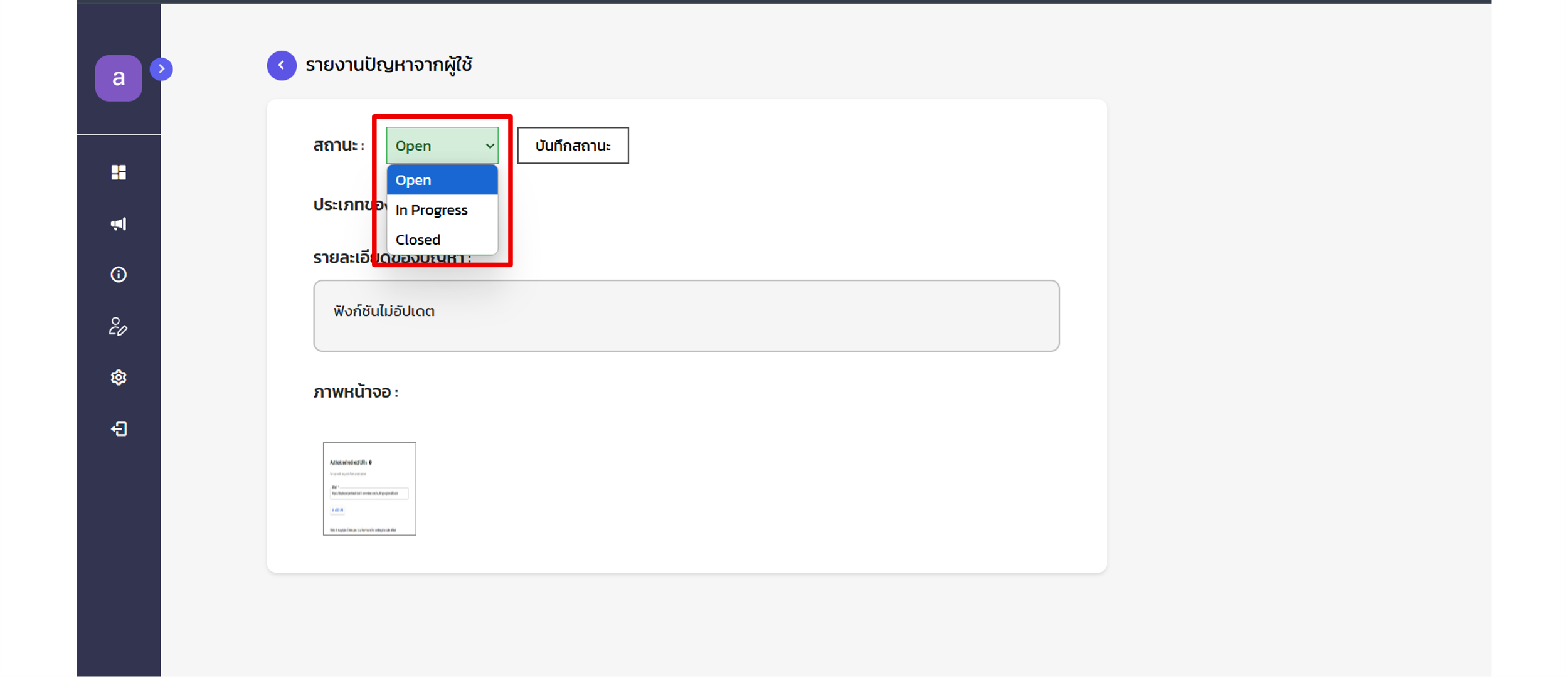


ภาพที่ หน้ารายงานปัญหาจากผู้ใช้ทั้งหมด

**รูปภาพประกอบด้วย ข้อความ, ภาพหน้าจอ, ซอฟต์แวร์, แสดง

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้ารายละเอียดของรายงานปัญหาจากผู้ใช้



ภาพที่ หน้ารายละเอียดของรายงานปัญหาการเปลี่ยนสถานะ

**รูปภาพประกอบด้วย ภาพหน้าจอ, ข้อความ, ซอฟต์แวร์, ไอคอนคอมพิวเตอร์

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้าประวัติรายงานปัญหาที่สิ้นสุดทั้งหมด

3.2.4 หน้ารายละเอียดผู้ใช้

ผู้ดูแลระบบดูรายละเอียดเกี่ยวกับผู้ใช้โดยคลิกผู้ใช้ที่ต้องการดู

**รูปภาพประกอบด้วย ภาพหน้าจอ, ข้อความ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้ารายละเอียดเกี่ยวกับผู้ใช้

3.2.5 หน้าตั้งค่าผู้ผู้ดูแลระบ

(1) ผู้ดูแลระบบสามารถเปลี่ยนโปรไฟล์และชื่อผู้ใช้ได้

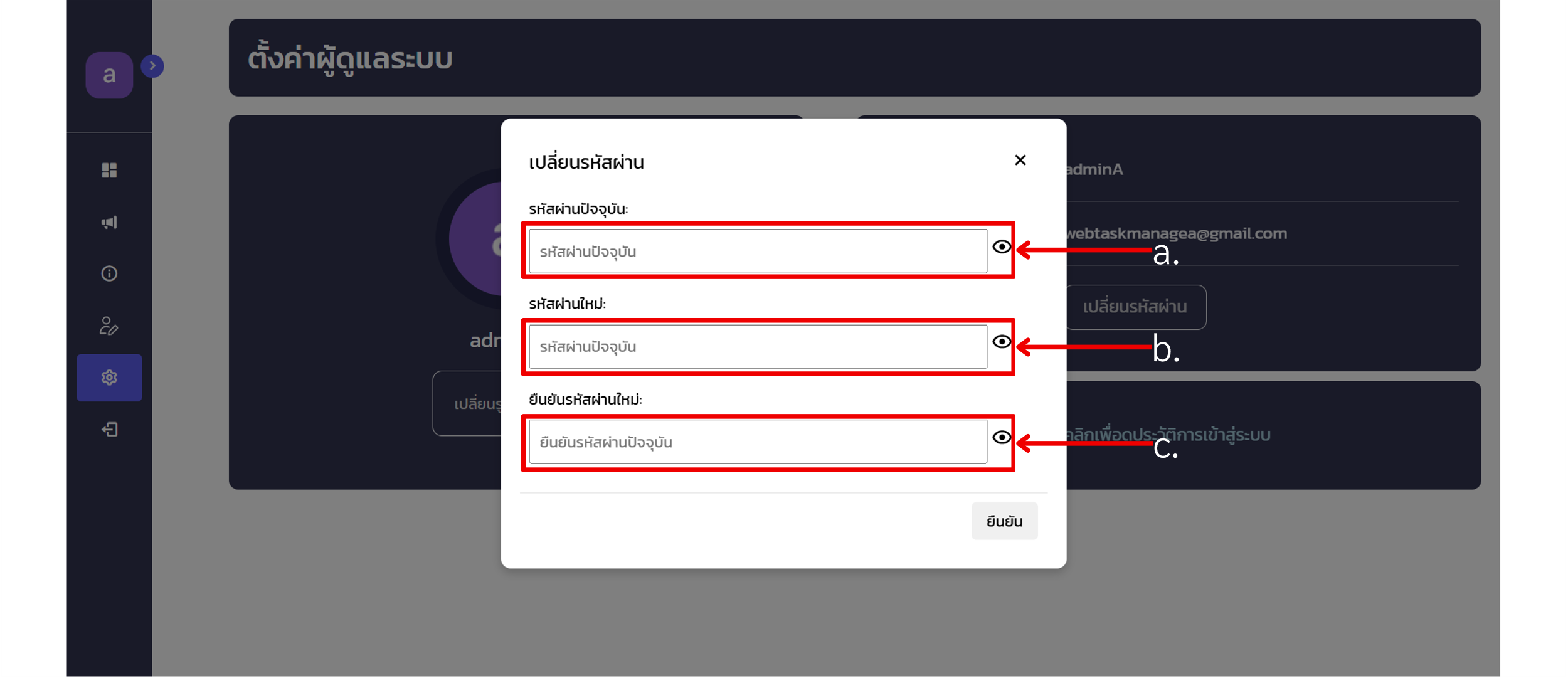
(2) กรณีที่ผู้ดูแลระบบต้องการเปลี่ยนรหัสผ่านให้คลิกที่ปุ่มเปลี่ยนรหัสผ่าน

(3) ผู้ดูแลระบบต้องกรอกรหัสผ่านโดย a.รหัสผ่านปัจจุบัน b.รหัสผ่านใหม่ที่ต้องการเปลี่ยน c.ยืนยันรหัสผ่านใหม่ที่ต้องการเปลี่ยน และคลิกที่ยืนยันเพื่อทำการเปลี่ยน

**รูปภาพประกอบด้วย ภาพหน้าจอ, ข้อความ, ซอฟต์แวร์, ซอฟต์แวร์มัลติมีเดีย

เนื้อหาที่สร้างโดย AI อาจไม่ถูกต้อง**

ภาพที่ หน้าตั้งค่าผู้ดูแลระบบ

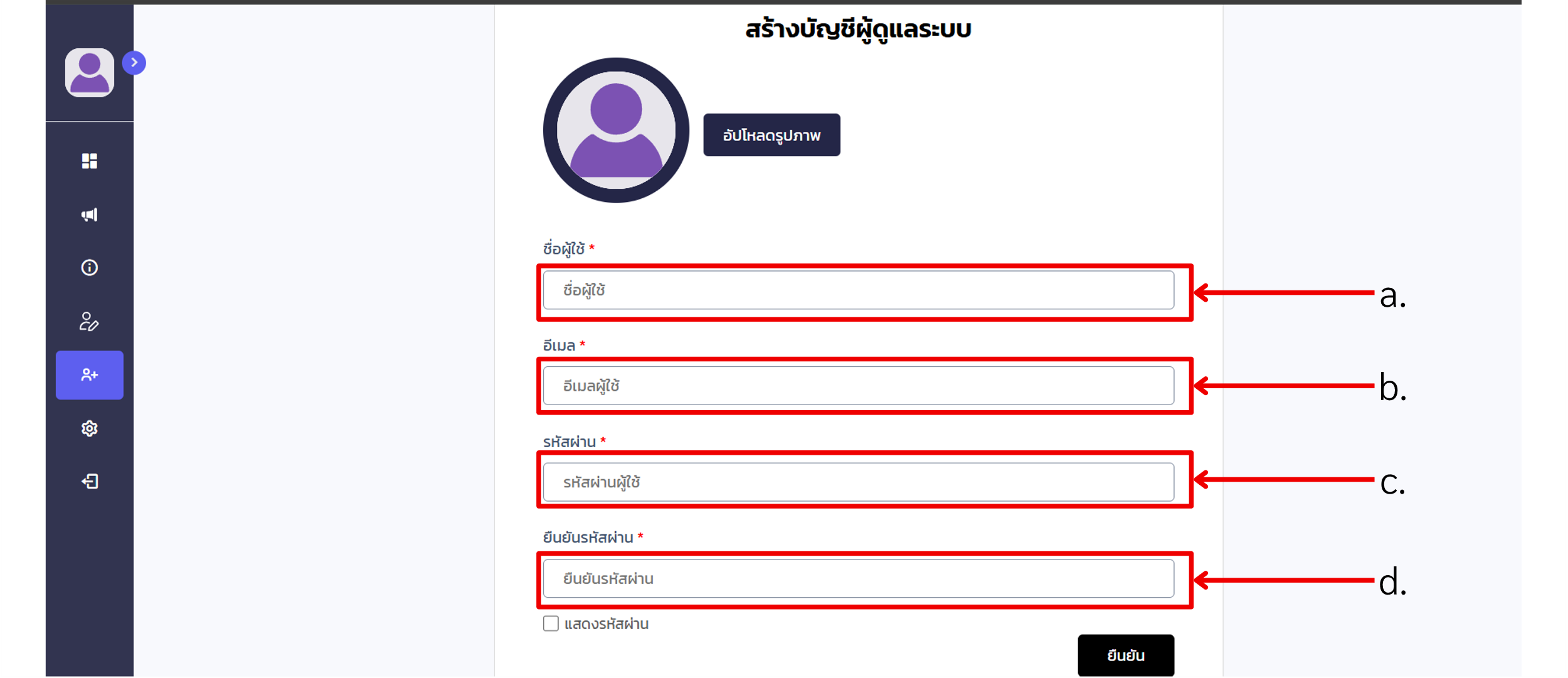


ภาพที่ หน้าเปลี่ยนรหัสผ่านตั้งค่าผู้ดูแลระบบ

3.2.5 หน้าสร้างผู้ดูแลระบบ

ผู้ดูแลระบบกรอกข้อมูลเพื่อสร้างผู้ดูแลระบบใหม่ a.ชื่อผู้ดูแลระบบ b.อีเมล c.รหัสผ่าน

d.ยืนยันรหัสผ่าน จากนั้นทำการคลิกปุ่ม ยืนยัน เพื่อสร้างผู้ใช้ใหม่



ภาพที่ หน้าสร้างผู้ดูแลระบบ

**บทที่ 4**

**บทสรุป**

เว็บแอปพลิเคชันการบริหารและจัดการงานที่ได้รับมอบหมาย Web application for Task Management ถูกพัฒนาขึ้นเพื่อช่วยให้ผู้ใช้สามารถจัดการงานที่ได้รับมอบหมาย และจัดการงานกลุ่มให้มีประสิทธิภาพเพิ่มขึ้นโดยมีฟังก์ชัน สร้างโปรเจกต์โดยกำหนดระยะเวลาในการทำโปรเจกต์ สามารถเพิ่มสมาชิกเข้าไปในโปรเจกต์และแบ่งงานให้สมาชิกได้ แดชบอร์ดสามารถดูความคืบหน้าของโปรเจกต์โดยจะมีการลำดับความสำคัญของงาน จำนวนงานที่กำลังทำ งานที่ยังไม่ทำ งานที่เสร็จสิ้น และบอกความคืบของงานในสมาชิกแต่ละคน มีปฏิทินที่แสดงกำนดเวลาต่างๆ มีระบบแชทที่สามารถพูดตอบโต้กันได้ของสมาชิกในโปรเจกต์ และแชทเดี่ยวที่สามารถคุยกับสมาชิกในโปรเจคได้

ในส่วนของผู้ดูแลระบบ จะมีหน้าที่ผู้ดูแลที่ทำหน้าที่ตรวจสอบระบบ ตรวจสอบสถานะสถิติการเข้าใช้งานผู้ใช้ ตรวจสอบรายงานผู้ใช้ และจัดการผู้ใช้ในระบบ ซึ่งระบบจะช่วยให้ผู้ใช้บริหารและจัดการงานได้มีประสิทธิภาพ

**แบบฟอร์มการตรวจเอกสารคู่มือการใช้งานระบบ**

**รายวิชา SC364772 โครงงานทางเทคโนโลยีสารสนเทศ 2**

**ภาคเรียนที่ 2 ปีการศึกษา 2567**

การตรวจสอบจากอาจารย์ที่ปรึกษาโครงงาน

(ลงชื่อ)

( ) วันที่ / /

ผู้จัดทำโครงงาน

(ลงชื่อ)

( นาย ภพ ในจิตต์ )

วันที่ / /

(ลงชื่อ)

( นางสาว ชัญญา โทมี )

วันที่ / /