

เอกสารความต้องการของระบบซอฟต์แวร์ (SRS)

Project : เว็บไซต์แจ้งปัญหาในชุมชน

จัดทำโดย:

- นาย ศิวกร ประทุมสุวรรณ (6730300574)
- นาย ณัฐพล บ่อวารี (6630300301)

1. บทนำ (Introduction)

เอกสารฉบับนี้จัดทำขึ้นเพื่อระบุข้อกำหนดความต้องการของระบบซอฟต์แวร์

(Software Requirement Specification - SRS) สำหรับ Project “เว็บแจ้งปัญหาในชุมชน”

เพื่อใช้เป็นแนวทางในการพัฒนาและตรวจสอบการทำงานของระบบให้ตรงตามวัตถุประสงค์ที่ตั้งไว้

1.1 วัตถุประสงค์

เอกสารนี้จัดทำขึ้นเพื่ออธิบายความต้องการของระบบ “เว็บแจ้งปัญหาในชุมชน” ซึ่งเป็น Web Application สำหรับให้ประชาชนแจ้งปัญหา เช่น ไฟถนนเสีย ถนนเป็นหลุม น้ำรั่ว ขยะล้น

เป็นต้น โดยระบบรองรับการอัปโหลดรูปภาพและปักหมุดตำแหน่งบนแผนที่

1.2 ขอบเขตการทำงาน

ระบบจะเป็น Web Application พัฒนาโดยใช้ภาษา Python และใช้ฐานข้อมูล SQLite

มีฟังก์ชัน CRUD และระบบ Authentication ตามที่รายวิชากำหนด

2. ภาครวมระบบ

2.1 มุ่งมองผลิตภัณฑ์

ระบบเป็น Web Application ทำงานผ่าน Browser

ผู้ใช้ต้องสมัครสมาชิกและเข้าสู่ระบบก่อนแจ้งปัญหา

2.2 คุณลักษณะผู้ใช้

- ผู้ใช้ทั่วไป (User)
- ผู้ดูแลระบบ (Admin)

2.3 ข้อจำกัด

- ใช้ SQLite เป็นฐานข้อมูล
- ใช้ Python (Flask) พัฒนา
- Deploy บน Cloud

3. ความต้องการด้านฟังก์ชัน (Functional Requirements)

3.1 ระบบสมาชิก (Authentication)

รหัส	รายละเอียดความต้องการ
FR-1	ผู้ใช้สามารถสมัครสมาชิกได้
FR-2	ผู้ใช้สามารถเข้าสู่ระบบได้
FR-3	ผู้ใช้สามารถออกจากระบบได้
FR-4	ระบบต้องเข้ารหัสผ่านก่อนบันทึกลงฐานข้อมูล

3.2 การจัดการแจ้งปัญหา (CRUD)

รหัส	รายละเอียดความต้องการ
FR-5	ผู้ใช้สามารถสร้างรายการแจ้งปัญหาได้ (Create)
FR-6	ผู้ใช้สามารถดูรายการแจ้งปัญหาทั้งหมดได้ (Read)
FR-7	ผู้ใช้สามารถแก้ไขรายการของตนเองได้ (Update)
FR-8	ผู้ใช้สามารถลบรายการของตนเองได้ (Delete)

3.3 การอัปโหลดรูปภาพ

รหัส	รายละเอียดความต้องการ
FR-9	ผู้ใช้สามารถอัปโหลดรูปภาพประกอบการแจ้งปัญหาได้
FR-10	ระบบต้องจัดเก็บ path ของรูปภาพในฐานข้อมูล

3.4 การปักหมุดแผนที่

รหัส	รายละเอียดความต้องการ
FR-11	ผู้ใช้สามารถคลิกปักหมุดบนแผนที่ได้
FR-12	ระบบต้องบันทึกค่า Latitude และ Longitude

3.5 ระบบผู้ดูแล (Admin)

รหัส	รายละเอียดความต้องการ
FR-13	Admin สามารถดูรายการแจ้งปัญหาทั้งหมดได้
FR-14	Admin สามารถเปลี่ยนสถานะปัญหา
FR-15	Admin สามารถลบรายการได้ก็ได้

5. โครงสร้างฐานข้อมูล (Database Design)

ตาราง Users

- id (PK)
- username
- password (hashed)
- role (user/admin)

ตาราง Reports

- id (PK)
- title
- description
- image_path
- latitude
- longitude

- status
- created_at
- user_id (FK)

6. Use Case

User:

- สมัครสมาชิก
- เข้าสู่ระบบ
- แจ้งปัญหา
- แก้ไขปัญหา
- ลบปัญหา
- ดูรายการ

Admin:

- เข้าสู่ระบบ
- ดูทั้งหมด
- เปลี่ยนสถานะ
- ลบข้อมูล

7. ความต้องการด้านซอฟต์แวร์ (Software Components)

ส่วนประกอบ	เทคโนโลยีที่ใช้
Backend	Python (Flask)
Database	SQLite
Frontend	HTML, CSS, Bootstrap
Client-side Script	JavaScript
Map Service	Leaflet.js
Web Browser	Chrome, Edge, Safari
Cloud Platform	Render / Railway / PythonAnywhere