

Smart Room Security System Using IoT and Web-Based Monitoring

Mr. Theerut Mueangsa
B.Eng. Computer Engineering
Khon Kaen, Thailand

Mr. Akkadech Jaengpromma
B.Eng. Computer Engineering
Khon Kaen, Thailand

Mr. Kaisak Kabklon
B.Eng. Computer Engineering
Khon Kaen, Thailand

บทความนี้นำเสนอการพัฒนาระบบรักษาความปลอดภัยห้องอัจฉริยะโดยใช้เทคโนโลยี *Internet of Things (IoT)* เพื่อเฝ้าระวังเหตุการณ์ผิดปกติภายในห้อง ระบบประกอบด้วยเซ็นเซอร์ตรวจจับควัน MQ-2 เซ็นเซอร์วัดอุณหภูมิและความชื้น DHT11 และเซ็นเซอร์อัลตร้าโซนิก HC-SR04 สำหรับตรวจสอบสถานะการเปิด–ปิดประตู ข้อมูลจากเซ็นเซอร์ส่งผ่านเครือข่าย WiFi ไปยัง *Web Server* ซึ่งพัฒนาด้วยภาษา PHP และจัดเก็บลงฐานข้อมูล *MariaDB* ระบบสามารถแสดงผลแบบ Real-time ผ่าน *Web Dashboard* พร้อมบันทึกข้อมูลย้อนหลังเพื่อการวิเคราะห์ ผลการทดลองพบว่าระบบสามารถตรวจจับควันและเปลี่ยนแปลงสถานะประตูได้อย่างแม่นยำและตอบสนองได้รวดเร็ว หมายเหตุที่สำคัญคือใช้ในบ้านอัจฉริยะหรือสำนักงานขนาดเล็ก

IoT, Smart Security, Web Monitoring, MariaDB, PHP

I. INTRODUCTION (HEADING I)

ในปัจจุบันเทคโนโลยี Internet of Things (IoT) ได้เข้ามามีบทบาทสำคัญในการพัฒนาระบบอัตโนมัติและระบบรักษาความปลอดภัย ระบบแบบดั้งเดิมมักนั้นเพียงการบันทึกการณ์ แต่ยังขาดการตรวจสอบสภาพแวดล้อม เช่น ควันไฟหรืออุณหภูมิที่สูงผิดปกติ

โครงการนี้จึงพัฒนาระบบ Smart Room Security ที่สามารถตรวจจับควัน ตรวจจับอุณหภูมิและความชื้น ควบคุมการเปิด–ปิดประตู และแสดงผลผ่าน Web Application แบบเรียลไทม์ พร้อมจัดเก็บข้อมูลลงฐานข้อมูลเพื่อวิเคราะห์ย้อนหลัง

II. SYSTEM ARCHITECTURE

ระบบถูกออกแบบตามสถาปัตยกรรม IoT แบ่งออกเป็น 3 ชั้นหลัก ได้แก่

1. Sensor Layer

ประกอบด้วย MQ-2, DHT11 และ HC-SR04 ทำหน้าที่ตรวจจับข้อมูลสภาพแวดล้อมและสถานะประตู

2. Application Layer

ไมโครคอนโทรลเลอร์ (เช่น ESP8266/ESP32) รับคำจากเซ็นเซอร์และส่งข้อมูลผ่าน WiFi ไปยัง *Web Server* โดยใช้ HTTP Request

3. Database Layer

Web Server พัฒนาด้วยภาษา PHP รับข้อมูลและบันทึกลงฐานข้อมูล *MariaDB* พร้อมทั้งจัดเก็บข้อมูลมาแสดงผลบน *Dashboard*

ลำดับการทำงานของระบบ:

- เซ็นเซอร์
- ไมโครคอนโทรลเลอร์
- Web Server (PHP)
- MariaDB*
- Web Dashboard

III. HARDWARE COMPONENTS

1. MQ-2 Gas Sensor

ใช้ตรวจจับควันและก๊าซไวไฟ เมื่อค่าควันเกินเกณฑ์ที่กำหนด ระบบจะส่งไฟ Buzzer ทำงานและแสดงสถานะ Fire Alert

2. DHT11 Sensor

ใช้ตัดอุณหภูมิและความชื้นภายในห้อง เพื่อตรวจสอบสภาพแวดล้อมและช่วยยืนยันเหตุการณ์ไฟไหม้

3. HC-SR04 Ultrasonic Sensor

ใช้ตรวจจับห้องของประตูเพื่อรับสถานะเปิด–ปิด

4. Servo Motor 180°

ใช้ควบคุมกลไกการเปิด–ปิดประตู

5. IRF520N MOSFET Module

ใช้บอร์ดที่ต้องการกระแสสูง เช่น mosfet

6. LED และ Buzzer

LED ใช้แสดงสถานะ WiFi และสถานะประตู Buzzer ใช้แจ้งเตือนเมื่อเกิดเหตุการณ์ไฟไหม้

IV. WEB APPLICATION AND DATABASE

ระบบเว็บพัฒนาด้วยภาษา PHP ทำงานร่วมกับ HTML, CSS และ JavaScript เพื่อสร้าง *Dashboard* และแสดงผลแบบ Real-time

หน้าที่ของ PHP ได้แก่:

- รับข้อมูลจากอุปกรณ์ IoT
- ประมวลผลเงื่อนไข เช่น Fire Status
- บันทึกข้อมูลลงฐานข้อมูล *MariaDB*
- ดึงข้อมูลล่าสุดมาแสดงผล

โครงสร้างฐานข้อมูล (Table: sensor_data)

- id – Primary Key
- smoke – ค่าควัน (PPM)
- temperature – อุณหภูมิ (°C)
- humidity – ความชื้น (%)
- door_status – สถานะประตู
- created_at – เวลาบันทึกข้อมูล

ข้อมูลจะถูกส่งจากอุปกรณ์ทุก ๆ 5 วินาที และบันทึกลงฐานข้อมูลเพื่อแสดงผลย้อนหลังผ่านกราฟอุณหภูมิและความชื้น

V. EXPERIMENTAL RESULTS

ผลการทดสอบระบบภายใต้สภาพแวดล้อมจริงพบว่า

- ระบบสามารถตรวจจับควันและส่ง Buzzer ทำงานภายใน 2–3 วินาที
- การเปิดประตูสามารถตรวจจับได้อย่างแม่นยำจากการเปลี่ยนแปลงระยะของ Ultrasonic Sensor
- ข้อมูลถูกบันทึกลง *MariaDB* อย่างต่อเนื่อง
- Dashboard สามารถแสดงผลข้อมูลแบบ Real-time ได้ถูกต้อง

ระบบมีความเสถียรและเหมาะสมสำหรับการใช้งานจริงในระดับบ้านอัจฉริยะ

VI. CONCLUSION

ระบบ Smart Room Security ที่พัฒนาขึ้นสามารถทำงานได้อย่างมีประสิทธิภาพโดยบูรณาการเทคโนโลยี IoT เข้ากับ Web Application และฐานข้อมูล ทำให้สามารถเฝ้าระวังเหตุการณ์ผิดปกติแบบต่อเนื่องและจัดเก็บข้อมูลเพื่อการวิเคราะห์ในอนาคต แนวทางพัฒนาต่อไป "ได้แก่

- เชื่อมต่อ Cloud Platform
- เพิ่ม Mobile Application
- เพิ่มระบบแจ้งเตือนผ่าน LINE หรือ Email
- วิเคราะห์ข้อมูลด้วย AI

REFERENCES

- [1] A. Bahga and V. Madisetti, *Internet of Things: A Hands-on Approach*, 2015.
- [2] Hanwei Electronics, "MQ-2 Gas Sensor Datasheet," 2018.
- [3] MariaDB Foundation, "MariaDB Documentation," 2023.
- [4] PHP Group, "PHP Manual," 2023.

1 Smart Room Security System

TH กanyaทัย

🚨 รายละเอียดโครงงาน

Smart Room Security System เป็นระบบรักษาความปลอดภัยห้องอัจฉริยะที่พัฒนาโดยใช้เทคโนโลยี Internet of Things (IoT) ร่วมกับ Web Application และฐานข้อมูล MariaDB เพื่อเฝ้าระวังเหตุการณ์สำคัญแบบ Real-time

ระบบสามารถตรวจจับควัน ตรวจวัดอุณหภูมิและความชื้น ตรวจสอบสถานะประตู และแสดงผลผ่าน Web Dashboard พร้อมบันทึกข้อมูลข้อความหลัง

🔥 ความสามารถของระบบ (Features)

- ตรวจจับควันด้วย MQ-2
- วัดอุณหภูมิและความชื้นด้วย DHT11
- ตรวจสอบสถานะประตูด้วย Ultrasonic Sensor (HC-SR04)
- ควบคุมประตูด้วย Servo Motor 180°
- แจ้งเตือนด้วย Buzzer
- LED แสดงสถานะ WiFi และประตู
- แสดงผลผ่าน Web Dashboard
- บันทึกข้อมูลฐานข้อมูล MariaDB
- ปุ่มเบรกประตูฉุกเฉิน

🛡️ โครงสร้างระบบ (System Architecture)

Sensors → ESP32 → Web Server (PHP) → MariaDB → Web Dashboard

🛠️ อุปกรณ์ที่ใช้ (Hardware)

- MQ-2 Gas Sensor
- DHT11 Temperature & Humidity Sensor
- HC-SR04 Ultrasonic Sensor
- Servo Motor 180°
- IRF520N MOSFET Module
- ESP32
- LED Indicators
- Buzzer
- Emergency Button

📊 เทคโนโลยีที่ใช้ (Software Stack)

- PHP
- MariaDB
- HTML / CSS / JavaScript
- XAMPP (Apache Server)
- Arduino IDE

📁 โครงสร้างฐานข้อมูล (Database Structure) ***

Table 1: `sensor_logs`

Field	Type	Description
id	INT (PK)	รหัส Auto Increment
temperature	FLOAT(5, 2)	อุณหภูมิ
humidity	FLOAT(5, 2)	ความชื้น
smoke_level	INT	ระดับควัน
created_at	TIMESTAMP	เวลาที่บันทึกข้อมูล

Table 2: `device_status`

Field	Type	Description
id	INT (PK)	รหัสอุปกรณ์
device_name	VARCHAR(50)	ชื่ออุปกรณ์ (door, buzzer, fan)
status	TINYINT(1)	0 = Off/Closed, 1 = On/Open

🚀 วิธีติดตั้งระบบ

1 ติดตั้ง Web Server

1. ติดตั้ง XAMPP
2. ติดตั้ง Apache และ MariaDB
3. สร้างฐานข้อมูลชื่อ `iot_security`
4. สร้างตาราง `sensor_logs` และ `device_status`
5. นำไฟล์รับไปไว้ในโฟเดอร์ `htdocs/`
6. ตั้งค่า WiFi และ URL ของเซิร์ฟเวอร์ในโค้ด ESP
7. อัปโหลดไฟล์ (Firmware) ลงบอร์ด

2 นำไฟล์รับไปไว้ใน

htdocs/

เมื่อติดตั้ง Browser:

http://localhost/iot-miniproject

3 ขั้นตอนในการติดตั้ง ESP

- ติดตั้ง Arduino IDE
- ใส่ WiFi SSID และ Password
- กำหนด URL ของ Web Server
- Upload ลงบอร์ด

🌐 หลักการทำงาน

1. ESP อ่านค่าจากเซ็นเซอร์
2. ส่งข้อมูลผ่าน WiFi ไปยัง PHP
3. PHP บันทึกข้อมูลลง MariaDB
4. Dashboard แสดงผลแบบ Real-time
5. PHP ส่งสถานะของ Outputs ไปยัง ESP

📈 ผลการทดสอบ

- ตรวจจับควันได้ภายใน 2–3 วินาที
- ตรวจจับการเปิดประตูได้แม่นยำ
- บันทึกข้อมูลต่อเนื่องทุก 5 วินาที

GB English Version

🚧 Project Overview

Smart Room Security System is an IoT-based smart room monitoring system integrated with a Web Application and MariaDB database for real-time monitoring and data logging.

The system detects smoke, measures temperature and humidity, monitors door status, and displays data through a web dashboard.

🔥 Features

- Smoke detection using MQ-2
- Temperature & humidity monitoring using DHT11
- Door status detection using Ultrasonic Sensor (HC-SR04)
- Door control via 180° Servo Motor
- Buzzer alert system
- WiFi and door status LED indicators
- Web-based real-time dashboard
- Data logging with MariaDB
- Emergency door release button

🏢 System Architecture

Sensors → ESP8266/ESP32 → Web Server (PHP) →
MariaDB → Web Dashboard

🛠️ Hardware Components

- MQ-2 Gas Sensor
- DHT11 Sensor
- HC-SR04 Ultrasonic Sensor
- 180° Servo Motor
- IRF520N MOSFET Module
- ESP8266 / ESP32
- LEDs
- Buzzer
- Emergency Button

💻 Software Stack

- PHP
- MariaDB
- HTML / CSS / JavaScript
- Apache (XAMPP)
- Arduino IDE

🗁 Database Structure

Table 1: `sensor_logs`

Field	Type	Description
id	INT (PK)	Auto Increment ID
temperature	FLOAT(5, 2)	Temperature value
humidity	FLOAT(5, 2)	Relative humidity value
smoke_level	INT	Smoke level
created_at	TIMESTAMP	Record timestamp

Table 2: `device_status`

Field	Type	Description
id	INT (PK)	Device ID
device_name	VARCHAR(50)	Device name (door, buzzer, fan)
status	TINYINT(1)	0 = Off/Close, 1 = On/Open

🚀 Installation

1. Install XAMPP
2. Start Apache & MariaDB
3. Create database 'iot_security'
4. Create table 'sensor_logs' and 'device_status'
5. Upload web files to 'htdocs/'
6. Configure WiFi and server URL in ESP code
7. Upload firmware

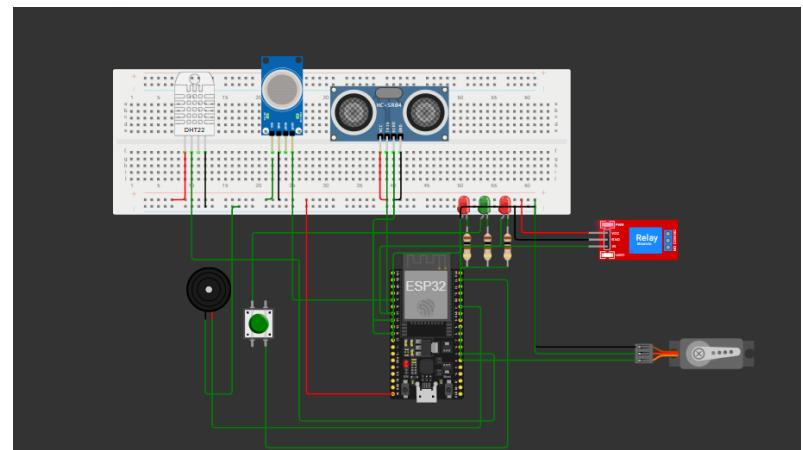


🔍 How It Works

1. ESP reads sensor data
2. Sends data via WiFi to PHP server
3. PHP stores data in MariaDB
4. Dashboard displays real-time information
5. PHP send outputs status to ESP

📈 Testing Results

- Smoke detected within 2–3 seconds
- Accurate door detection
- Continuous data logging every 5 seconds



Environment Monitoring

