

TRƯỜNG ĐẠI HỌC KHOA HỌC
KHOA CÔNG NGHỆ THÔNG TIN



Đề tài: *Sử dụng PIR sensor để phát hiện chuyển động, gửi tin nhắn qua Telegram hoặc email khi có sự xâm nhập.*

HỌC PHẦN : Phát triển ứng dụng IoT

Giảng viên hướng dẫn : Võ Việt Dũng

Học viên thực hiện : Huỳnh Văn Nhân

Chuyên ngành : Công nghệ thông tin

Khóa học : K45

HUẾ, THÁNG 4 NĂM 2025

Mục Lục

1. Mở đầu	1
2. Nội dung của đề tài	2
2.1 Tổng quan về cảm biến PIR	2
a. Cấu tạo và nguyên lý hoạt động của cảm biến PIR:	2
b. Ứng dụng thực tế của cảm biến PIR trong an ninh, tự động hóa:	2
2.2 Các nội dung thực hiện	3
c. Lập trình chức năng xử lý và cảnh báo	3
2.3 Thiết kế mạch IoT	4
a. Sơ đồ logic của hệ thống	4
b. Giải thích chức năng linh kiện	5
c. Nguyên lý hoạt động	5
d. Mạch vật lý:	5
2.4 Cài đặt hệ thống	6
a. Công cụ sử dụng:	6
b. Quy trình:	6
2.5 Kết quả thực hiện	8
a. Ảnh chụp mạch trên Wokwi:	8
b. Giao diện phản hồi trên Telegram hoặc email:	9
c. Giải thích	9
3. Kết luận và hướng phát triển	11
3.1 Các công việc đã thực hiện:	11
3.2 công việc chưa thực hiện:	11
3.3 Hướng phát triển của đề tài:	11
Tài Liệu Tham Khảo:	12
Phụ Lục:	12

1. Mở đầu

Trong thời đại công nghệ ngày càng phát triển, việc ứng dụng các thiết bị điện tử vào hệ thống an ninh và giám sát ngày càng trở nên phổ biến và cần thiết. Trong số đó, cảm biến hồng ngoại thụ động (PIR sensor) là một thiết bị đơn giản nhưng hiệu quả, có khả năng phát hiện chuyển động của con người trong khu vực giám sát.

Việc kết hợp cảm biến PIR với các nền tảng giao tiếp hiện đại như Telegram hoặc email cho phép người dùng nhận được cảnh báo tức thời khi có sự xâm nhập, từ đó nâng cao tính chủ động và hiệu quả trong việc bảo vệ tài sản và không gian sống.

Tiểu luận này tập trung nghiên cứu và xây dựng một hệ thống phát hiện chuyển động sử dụng cảm biến PIR, đồng thời tích hợp chức năng gửi cảnh báo đến người dùng thông qua ứng dụng Telegram hoặc email. Hệ thống được triển khai trên nền tảng vi điều khiển ESP32 và mô phỏng hoạt động thông qua môi trường giả lập Wokwi, nhằm đánh giá tính khả thi và hiệu suất hoạt động trong điều kiện thực tế.

Đây là một bước tiếp cận thiết thực và tiết kiệm trong việc xây dựng các giải pháp an ninh thông minh, phù hợp với nhu cầu của cá nhân, hộ gia đình cũng như các mô hình giám sát quy mô nhỏ.

Đề tài “Phát hiện chuyển động sử dụng cảm biến PIR và gửi cảnh báo qua Telegram hoặc Email” là một giải pháp thực tiễn, tiết kiệm, dễ triển khai, giúp người dùng nhanh chóng nhận biết được các hoạt động bất thường và có phản ứng kịp thời.

2. Nội dung của đề tài

2.1 Tổng quan về cảm biến PIR

a. Cấu tạo và nguyên lý hoạt động của cảm biến PIR:

- Cấu tạo của cảm biến PIR khá đơn giản, bao gồm một ống kính fresnel, một bộ lọc hồng ngoại và một đĩa phẳng gồm nhiều phần tử nhạy cảm với bức xạ hồng ngoại. Khi một vật thể phát nhiệt di chuyển qua phạm vi phát hiện của cảm biến, sự thay đổi về bức xạ hồng ngoại sẽ được ghi nhận và tín hiệu được tạo ra.

- Nguyên lý cốt lõi của cảm biến PIR dựa trên khả năng phát hiện sự thay đổi bức xạ hồng ngoại từ các vật thể phát nhiệt trong môi trường xung quanh. Cơ chế hoạt động chi tiết của cảm biến này khá thú vị và đáng để tìm hiểu:

+ Đầu tiên, ống kính fresnel trên cảm biến PIR sẽ hội tụ bức xạ hồng ngoại từ môi trường lên đĩa phẳng gồm nhiều phần tử nhạy cảm.

+ Khi không có vật thể chuyển động, các phần tử này nhận được lượng bức xạ hồng ngoại đồng đều và ổn định từ môi trường xung quanh.

+ Tuy nhiên, khi một vật thể phát nhiệt như con người hay động vật di chuyển qua phạm vi phát hiện của cảm biến, lượng bức xạ hồng ngoại nhận được bởi các phần tử sẽ thay đổi đột ngột. Sự thay đổi này được ghi nhận và tạo ra tín hiệu điện tử tương ứng.

+ Các mạch điện tử tiên tiến bên trong cảm biến PIR sẽ xử lý và khuếch đại tín hiệu này, loại bỏ các nhiễu không mong muốn. Khi tín hiệu đạt đến ngưỡng nhất định, cảm biến sẽ kích hoạt và gửi tín hiệu đầu ra đến thiết bị hoặc hệ thống liên kết.

b. Ứng dụng thực tế của cảm biến PIR trong an ninh, tự động hóa:

- Hệ thống an ninh: PIR là thành phần chính trong nhiều hệ thống báo động chống trộm và giám sát an ninh. Khi phát hiện chuyển động của con người hoặc động vật trong khu vực được giám sát, cảm biến sẽ kích hoạt tín hiệu báo động, cảnh báo cho chủ nhà hoặc lực lượng an ninh.

- Chiếu sáng thông minh: Cảm biến PIR được tích hợp trong các đèn chiếu sáng thông minh, giúp tự động bật/tắt đèn khi có người đi qua. Điều này không chỉ mang lại sự tiện lợi, mà còn giúp tiết kiệm năng lượng đáng kể.
- Hệ thống làm mát/sưởi ấm tự động: Trong các tòa nhà, cảm biến PIR có thể được sử dụng để phát hiện sự hiện diện của con người và tự động điều chỉnh hệ thống làm mát hoặc sưởi ấm, giúp tiết kiệm năng lượng khi không có người trong phòng.
- Thiết bị điều khiển tự động: Cảm biến PIR được tích hợp vào các thiết bị điều khiển tự động như vòi nước, cửa tự động, máy chiếu... Khi phát hiện chuyển động, các thiết bị này sẽ tự động kích hoạt, mang lại sự tiện lợi và hiệu quả cao.

2.2 Các nội dung thực hiện

a. Tìm hiểu kiến thức cơ bản và công nghệ sử dụng

- Cảm biến PIR (Passive Infrared Sensor): tìm hiểu nguyên lý hoạt động cảm biến phát hiện chuyển động thông qua việc đo sự thay đổi hồng ngoại phát ra từ cơ thể người
- ESP32 nghiên cứu cấu trúc và khả năng của vi điều khiển ESP32 đặc biệt là khả năng kết nối Wi-Fi và xử lý dữ liệu cảm biến
- Giao tiếp qua Internet: Telegram Bot tìm hiểu cách tạo bot trên Telegram, cách sử dụng API để gửi tin nhắn SMTP email nghiên cứu cách cấu hình gửi email thông qua giao thức SMTP sử dụng ESP32 chẳng hạn dùng Gmail hoặc dịch vụ SMTP miễn phí

b. Thiết kế mạch nguyên lý

- Thiết lập sơ đồ kết nối đơn giản giữa cảm biến PIR và ESP32
- Chân tín hiệu của PIR nối với chân digital ví dụ GPIO 14 của ESP32
- Thiết kế trên nền tảng Wokwi để dễ dàng mô phỏng và kiểm tra

c. Lập trình chức năng xử lý và cảnh báo

- Viết mã điều khiển với các chức năng chính:
 - + Kết nối ESP32 với mạng Wi-Fi cục bộ
 - + Đọc tín hiệu digital từ cảm biến PIR

+ Khi phát hiện chuyển động gọi hàm gửi thông báo qua Telegram Bot API hoặc gửi email thông qua SMTP server

- Sử dụng các thư viện như

+ WiFi.h HTTPClient.h UniversalTelegramBot.h cho Telegram

+ ESP_Mail_Client.h nếu sử dụng phương án gửi email

d. Tạo và cấu hình dịch vụ cảnh báo

- Telegram:

+ Tạo bot mới thông qua BotFather trên Telegram

+ Lấy Token và Chat ID để sử dụng trong chương trình ESP32

+ Cấu hình nội dung tin nhắn thông báo thời gian và mô tả sự kiện chuyển động

- Email tùy chọn:

+ Cấu hình địa chỉ người gửi người nhận và máy chủ SMTP

+ Đảm bảo sử dụng đúng thông tin xác thực và bật quyền truy cập SMTP nếu dùng Gmail

e. Giả lập hệ thống trên Wokwi

Sử dụng Wokwi.com để mô phỏng toàn bộ mạch và chương trình

Những thứ cần kiểm tra:

- Hoạt động của cảm biến PIR trong môi trường giả lập

- Tình trạng kết nối Wi-Fi

- Khả năng gửi và nhận thông báo thực tế qua Telegram hoặc email

- Ghi nhận hình ảnh giao diện và log hiển thị khi chương trình chạy thành công

f. Kiểm thử và đánh giá

- Chạy nhiều lần để kiểm tra khả năng nhận diện chuyển động chính xác

- Đánh giá độ trễ trong việc gửi thông báo từ khi phát hiện đến khi người dùng nhận được

- Đảm bảo tính ổn định khi hệ thống hoạt động trong thời gian dài trên giả lập

2.3 Thiết kế mạch IoT

a. Sơ đồ logic của hệ thống

[PIR Sensor] → [ESP32] → [Kết nối Wi-Fi] → [Telegram / Email]

b. Giải thích chức năng linh kiện

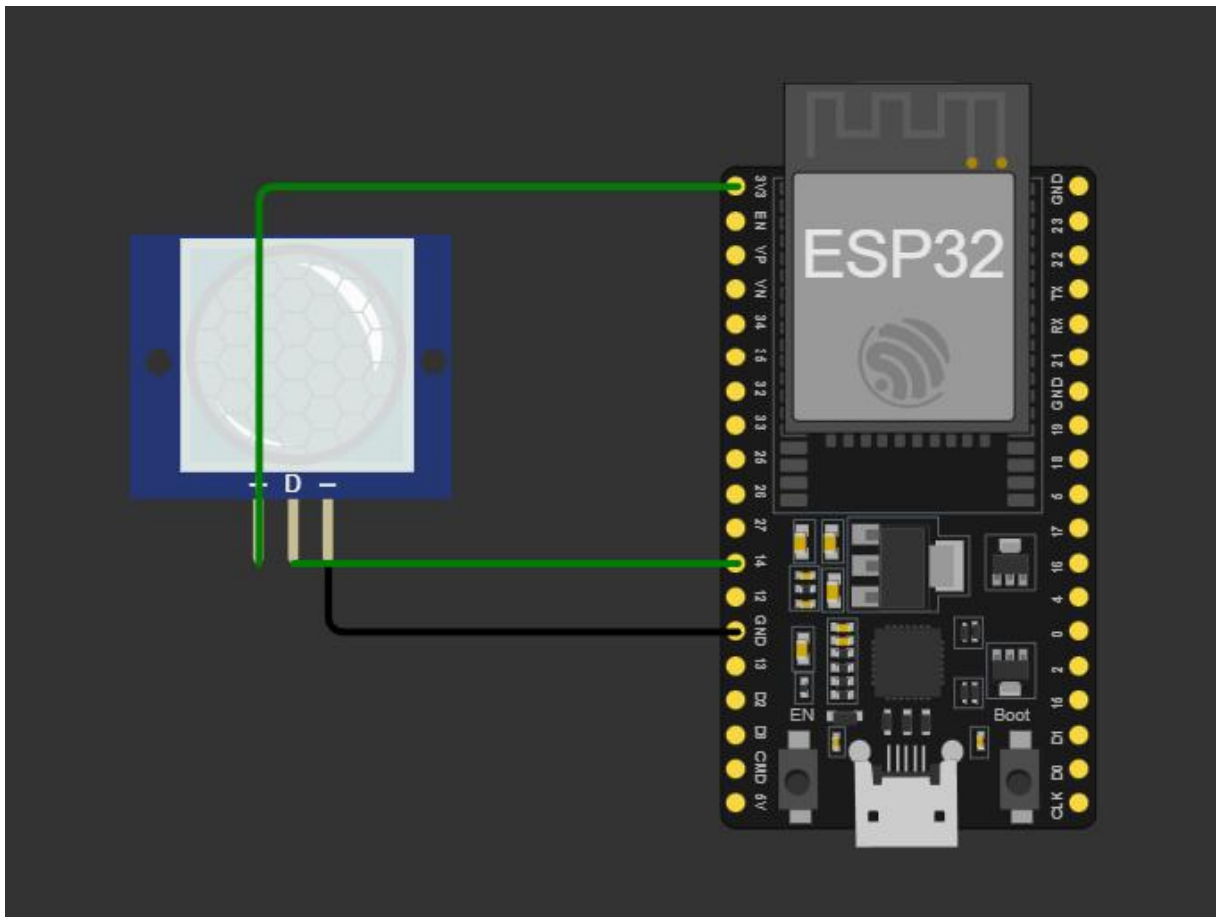
- **Cảm biến PIR (HC-SR501):** Phát hiện chuyển động của con người thông qua hồng ngoại.
- **ESP32:** Vi điều khiển có tích hợp Wi-Fi, nhận tín hiệu từ PIR và gửi thông báo.
- **Nguồn điện:** Cung cấp năng lượng cho mạch (trên Wokwi giả lập nên dùng nguồn mặc định).
- **Kết nối Wi-Fi:** Giúp hệ thống truyền thông báo ra ngoài internet.

c. Nguyên lý hoạt động

- Khi cảm biến PIR phát hiện có chuyển động, nó gửi tín hiệu HIGH đến chân digital của ESP32.
- ESP32 xử lý tín hiệu và nếu có chuyển động:
 - + Kết nối đến mạng Wi-Fi đã cấu hình.
 - + Gửi thông báo đến người dùng qua Telegram Bot hoặc qua SMTP (email).
- Nếu không có chuyển động, hệ thống chờ và tiếp tục giám sát.

d. Mạch vật lý:

Sử dụng giả lập wokwi:



2.4 Cài đặt hệ thống

a. Công cụ sử dụng:

- **Wokwi:** Mô phỏng phần cứng online.
- **PlatformIO:** Viết và biên dịch mã nguồn.
- **BotFather & Telegram Bot API:** Tạo bot và nhận token.
- **SMTP Server** (nếu chọn gửi email): Nhận thông báo qua tài khoản Gmail hoặc Mailtrap.

b. Quy trình:

- Viết code đọc tín hiệu PIR:


```

void setup() {
  Serial.begin(115200);
  pinMode(PIR_PIN, INPUT);

  // Kết nối Wi-Fi
  WiFi.begin(ssid, password);
  Serial.print("🔄 Đang kết nối Wi-Fi");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(1000);
  }
  Serial.println("\n✅ Đã kết nối Wi-Fi!");
}

void loop() {
  int motion = digitalRead(PIR_PIN);
  if (motion == HIGH) {
    Serial.println("🔥 Phát hiện chuyển động! Gửi cảnh báo...");
    sendTelegramMessage("🔥 Cảnh báo: Phát hiện chuyển động!");
    sendEmailAlert();
    delay(10000);
  }
  delay(500);
}

```

- Cài đặt thư viện Wi-Fi và Telegram Bot (hoặc thư viện gửi email).

```

#include <WiFi.h>
#include <HTTPClient.h>
#include <Base64.h>

//Thông tin Wi-Fi
const char* ssid = "Wokwi-GUEST";
const char* password = "";

//Thông tin Telegram Bot
String BOT_TOKEN = "8027625067:AAHdREtzh1lGcXPLv2VNkIr8-Lo5ffGE3pI"; //Bot Token
String CHAT_ID = "-1002439452600"; //Chat ID

//Thông tin Gmail (SMTP Server)
const char* smtpServer = "smtp.gmail.com";
const int smtpPort = 443;
const char* emailSender = "nhanhuynh1203@gmail.com"; // Email gửi
const char* emailPassword = "Banh10112003."; // Mật khẩu ứng dụng
const char* emailReceiver = "banhbodoi1203@gmail.com"; // Email nhận

```

- Lập trình chức năng gửi cảnh báo.

```
void sendTelegramMessage(String message) {
    String url = "https://api.telegram.org/bot" + BOT_TOKEN + "/sendMessage?chat_id=" + CHAT_ID + "&text=" + message;
    HTTPClient http;
    http.begin(url);
    int httpResponseCode = http.GET();
    http.end();
}

void sendEmailAlert() {
    WiFiClientSecure client;
    client.setInsecure();

    if (!client.connect(smtpServer, smtpPort)) {
        Serial.println("✗ Kết nối SMTP thất bại!");
        return;
    }

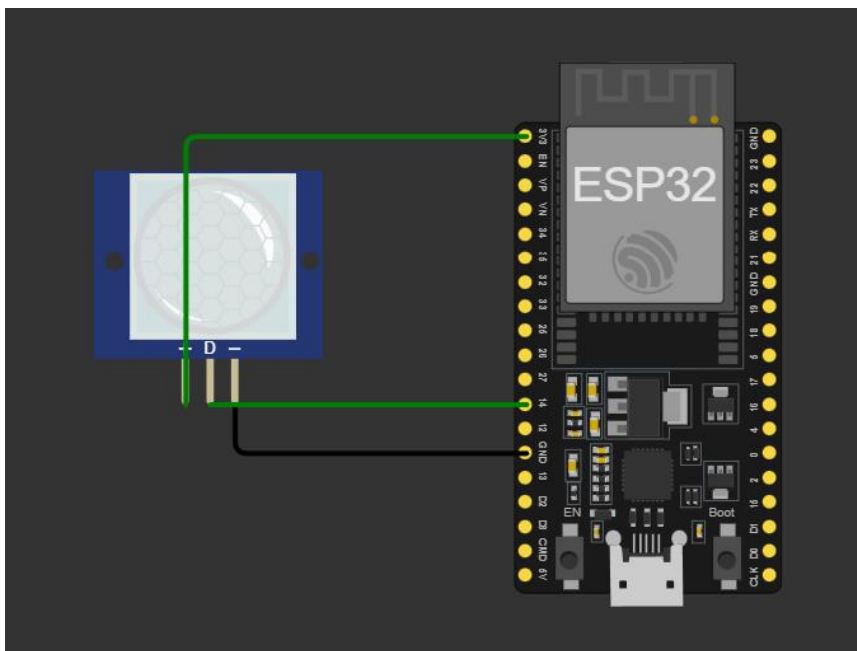
    // Gửi lệnh SMTP
    client.println("EHLO smtp.gmail.com");
    client.println("AUTH LOGIN");
    client.println(base64::encode(emailSender));
    client.println(base64::encode(emailPassword));
    client.println("MAIL FROM:<" + String(emailSender) + ">");
    client.println("RCPT TO:<" + String(emailReceiver) + ">");
    client.println("DATA");
    client.println("Subject: 🚨 Cảnh báo chuyển động!");
    client.println("Phát hiện có người di chuyển!");
    client.println(".");
    client.println("QUIT");

    Serial.println("✅ Email đã gửi!");
}
```

- Giả lập trên Wokwi và kiểm tra phản hồi từ Telegram hoặc email.

2.5 Kết quả thực hiện

a. Ảnh chụp mạch trên Wokwi:



b. Giao diện phản hồi trên Telegram hoặc email:



c. Giải thích

- Cảm biến PIR phát hiện chuyển động(Sử dụng giả lập wokwi nên sẽ nhấn “Simulate Motion”): Cảm biến PIR hoạt động dựa trên nguyên lý phát hiện sự thay đổi bức xạ hồng ngoại phát ra từ cơ thể người hoặc động vật so với nhiệt độ môi trường xung quanh. Khi có người di chuyển ngang qua vùng cảm nhận của cảm biến PIR, mức tín hiệu điện tại chân OUT của cảm biến sẽ thay đổi từ mức thấp sang mức cao thường là từ 0V lên 3.3V hoặc 5V tùy cảm biến

- ESP32 nhận tín hiệu từ cảm biến PIR

+ Chân OUT của cảm biến được nối với một chân digital input trên ESP32 ví dụ GPIO 14

+ ESP32 liên tục kiểm tra trạng thái mức logic của chân này

+ Khi phát hiện mức cao nghĩa là có chuyển động ESP32 sẽ kích hoạt một hành động xử lý tương ứng

- Gửi cảnh báo đến người dùng qua Internet

Sau khi phát hiện chuyển động ESP32 tiến hành gửi cảnh báo đến người dùng thông

qua kết nối Wi-Fi đã được thiết lập sẵn

Tùy vào cấu hình hệ thống có thể chọn một trong hai phương án cảnh báo:

+ Phương án 1 Gửi tin nhắn Telegram:

ESP32 sử dụng HTTP request để gửi dữ liệu đến API của Telegram Bot

Thông điệp cảnh báo ví dụ như “Cảnh báo có chuyển động phát hiện”

ESP32 sử dụng Token và Chat ID để xác định đúng bot và người nhận

+ Phương án 2 Gửi email cảnh báo

Lý do Wokwi không gửi được email:

Wokwi không hỗ trợ kết nối mạng thực tế: Wokwi chỉ giả lập việc kết nối Wi-Fi, nghĩa là khi bạn gọi `WiFi.begin()` và kiểm tra `WiFi.status()`, nó sẽ trả về trạng thái "connected" như thật. Tuy nhiên, các request HTTP hoặc SMTP gửi ra Internet thật sự không thể thực hiện được vì Wokwi không có quyền truy cập vào mạng Internet thực

SMTP yêu cầu kết nối mạng thật: Để gửi email qua SMTP (ví dụ như Gmail), ESP32 cần mở một kết nối TCP tới server SMTP như `smtp.gmail.com`. Điều này đòi hỏi truy cập Internet thật sự, thứ mà Wokwi không cung cấp

Một số thư viện như `ESP_Mail_Client` cần môi trường thực tế: Các thư viện gửi email cần truy cập DNS, mở cổng TCP bảo mật (TLS/SSL) và xử lý phản hồi từ máy chủ. Những bước này không được mô phỏng trong Wokwi

- Hệ thống quay lại trạng thái chờ

Sau khi gửi xong cảnh báo ESP32 quay lại chế độ lắng nghe tín hiệu từ cảm biến PIR
Hệ thống tiếp tục giám sát để phát hiện các chuyển động tiếp theo

3. Kết luận và hướng phát triển

3.1 Các công việc đã thực hiện:

- Hệ thống đã được thiết kế và mô phỏng hoàn chỉnh dựa trên cảm biến PIR và vi điều khiển ESP32 sử dụng nền tảng giả lập Wokwi
- Đã xây dựng thành công quy trình nhận diện chuyển động tự động và kích hoạt cảnh báo gửi qua hai phương thức phổ biến là Telegram Bot và email SMTP
- Cấu hình kết nối mạng Wi-Fi và xử lý tín hiệu cảm biến được lập trình đầy đủ trên VS Code
- Kiểm thử hệ thống qua nhiều tình huống giả lập giúp đảm bảo tính ổn định trong quá trình phát hiện và phản hồi
- Mô hình phù hợp để ứng dụng vào các giải pháp bảo vệ an ninh, giám sát không gian theo thời gian thực với chi phí thấp

3.2 công việc chưa thực hiện:

- thống hiện mới chỉ được kiểm thử trong môi trường giả lập chưa triển khai trên thiết bị thật vì lý do giới hạn phần cứng
- Chưa có tính năng ghi lại log hoặc lưu lịch sử phát hiện chuyển động để phục vụ truy xuất về sau
- Chưa tích hợp các chức năng mở rộng như chụp ảnh bằng ESP32-CAM hoặc điều khiển thiết bị từ xa thông qua Telegram Bot hoặc Web Server

3.3 Hướng phát triển của đề tài:

- Phát triển thêm chức năng kết nối với camera ESP32-CAM để gửi kèm hình ảnh thực tế tại thời điểm có chuyển động giúp tăng độ tin cậy và trực quan cho cảnh báo
- Tích hợp chức năng lưu trữ dữ liệu đám mây hoặc gửi log về server giúp người dùng quản lý và theo dõi lịch sử cảnh báo từ xa
- Nâng cao khả năng xử lý thông minh bằng việc ứng dụng công nghệ AI để nhận diện người từ chuyển động từ đó phân loại cảnh báo hợp lý tránh gửi cảnh báo giả
- Tích hợp giao diện người dùng đơn giản trên nền web hoặc mobile để người dùng theo dõi và điều khiển trực tiếp các chế độ hoạt động của hệ thống

- Tối ưu hóa thời lượng pin nếu triển khai trên thiết bị thực tế với nguồn di động bằng cách sử dụng chế độ sleep và đánh thức bằng ngắt ngoài từ cảm biến PIR

Tài Liệu Tham Khảo:

[1] Thông tin về PIR - <https://fptshop.com.vn/tin-tuc/dien-may/pir-la-gi-180717>

[2] Cách tạo bot telegram - https://www.iotzone.vn/esp32/cach-dung-esp32-telegram-dieu-khien-den-led-voi-arduino-ide/#Tao_Bot_Telegram

[3] Thông tin về STMP - <https://aws.amazon.com/vi/what-is/smt>

[4] Cách sử dụng API Telegram - <https://core.telegram.org/api>

[5] Công cụ hỗ trợ - <https://chatgpt.com>

Phụ Lục:

Phụ Lục A. Source Code: https://github.com/vvdung-husc/2024-2025.2.TIN4024.006/tree/main/TEAM_08/Huynh_Van_Nhan/PIR

