

**TRƯỜNG ĐẠI HỌC VĂN LANG**

**KHOA CÔNG NGHỆ THÔNG TIN**



******

**ĐỒ ÁN CUỐI KỲ**

**MÔN HỌC NHẬP MÔN HỆ THỐNG NHÚNG**

**INTERNET OF THINGS ( IOT )**

***Chủ đề:***

**Hệ Thống Theo Dõi an ninh**

**(((**

**SVTH1:** Huỳnh Tấn Ngọc Sơn - 2374802010434

**SVTH2:** Trần Phan Trọng Nguyễn - 2374802010343

**LỚP:**  243\_71NWSE40103\_01

**GVHD: Nguyễn Trương Khang**

**TP. Hồ Chí Minh – 26/7/2025**

**I.LỜI MỞ ĐẦU**

**1.GIỚI THIỆU TỔNG QUAN VỀ ĐỀ TÀI**

**- Trong thời đại công nghệ hiện nay, nhu cầu đảm bảo an ninh cho nhà ở, văn phòng, phòng riêng... ngày càng tăng. Các hệ thống báo động truyền thống thường đắt tiền và khó triển khai. Vì vậy, đề tài này hướng đến việc xây dựng một hệ thống báo động thông minh sử dụng vi điều khiển ESP32 và cảm biến chuyển động PIR, có khả năng gửi cảnh báo qua PushSafer khi phát hiện có người di chuyển trong khu vực giám sát. Hệ thống sử dụng các linh kiện giá rẻ, dễ tiếp cận, nhưng vẫn đảm bảo khả năng hoạt động hiệu quả và có thể mở rộng thêm các tính năng IoT khác trong tương lai.**

**2.LÝ DO CHỌN ĐỀ TÀI**

**- Hệ thống có khả năng ứng dụng cao trong cuộc sống: giám sát nhà ở, phòng riêng, kho hàng…**

**- Sử dụng các linh kiện phổ biến như ESP32, cảm biến PIR, LED và buzzer giúp sinh viên dễ tiếp cận.**

**- PushSafer là nền tảng miễn phí, hoạt động ổn định, phù hợp cho sinh viên triển khai.**

**- Giúp sinh viên áp dụng các kiến thức về hệ thống nhúng, vi điều khiển, lập trình MicroPython, và kết nối Internet (IoT).**

**- Là bước khởi đầu tốt cho các hệ thống giám sát nâng cao trong các khóa học hoặc dự án thực tế sau này.**

**3.MỤC TIÊU VÀ PHẠM VI THỰC HIỆN**

**Mục tiêu:**

**- Xây dựng hệ thống phát hiện chuyển động sử dụng cảm biến PIR và ESP32.**

**- Cảnh báo bằng cả đèn LED và âm thanh từ buzzer khi phát hiện chuyển động.**

**- Gửi thông báo khẩn cấp đến điện thoại người dùng qua PushSafer API.**

**- Viết chương trình bằng MicroPython, thực thi trên Thonny IDE.**

**Phạm vi thực hiện:**

**- Giám sát 1 khu vực cố định như phòng ngủ, phòng làm việc, nhà kho,...**

**- Hệ thống hoạt động cục bộ có kết nối WiFi (không cần điện toán đám mây phức tạp).**

**- Không sử dụng server riêng hay app phức tạp như Blynk, Firebase,...**

**II.CƠ SỞ LÝ THUYẾT**

**1.TỔNG QUAN VỀ HỆ THỐNG NHÚNG VÀ IOT**

**- Hệ thống nhúng là một hệ thống máy tính chuyên dụng được tích hợp bên trong các thiết bị điện tử. Nó có khả năng xử lý dữ liệu từ cảm biến, điều khiển đầu ra (như LED, còi...) và hoạt động theo chương trình được lập trình sẵn.**

**- Internet of Things (IoT) là khái niệm mô tả các thiết bị vật lý có khả năng kết nối Internet để thu thập, trao đổi và chia sẻ dữ liệu. Trong hệ thống này, thiết bị có thể gửi dữ liệu đến người dùng từ xa (như qua PushSafer), giúp giám sát hiệu quả.**

**- Mối quan hệ: Hệ thống nhúng là nền tảng phần cứng của IoT. ESP32 là một ví dụ điển hình khi vừa là vi điều khiển, vừa hỗ trợ kết nối WiFi để trở thành một thiết bị IoT thông minh.**

**2.CÁC CÔNG NGHỆ PHẦN MỀM , PHẦN CỨNG ĐƯỢC SỬ DỤNG.**

**Phần cứng**:

- **ESP32-S3**: vi điều khiển có Wi-Fi tích hợp, lập trình bằng MicroPython.

- **Cảm biến PIR (HC-SR501)**: phát hiện chuyển động bằng hồng ngoại.

- **LED**: hiển thị cảnh báo trực quan khi có chuyển động.

- **Buzzer**: phát ra âm thanh cảnh báo khi có xâm nhập.

- **Breadboard, điện trở, dây nối**: hỗ trợ lắp mạch thử nghiệm.

**Phần mềm**:

- **Thonny IDE**: môi trường lập trình MicroPython dễ dùng cho ESP32.

- **MicroPython firmware**: hệ điều hành nhúng nhẹ cho ESP32.

- **PushSafer API**: dịch vụ gửi thông báo về điện thoại miễn phí thông qua HTTP request, thay thế cho Blynk, Telegram...

**3. Các khái niệm liên quan**

**- ESP32: là dòng vi điều khiển tích hợp Wi-Fi và Bluetooth, phù hợp cho ứng dụng IoT nhúng. Nó có thể lập trình bằng nhiều ngôn ngữ như Arduino, C++, MicroPython,...**

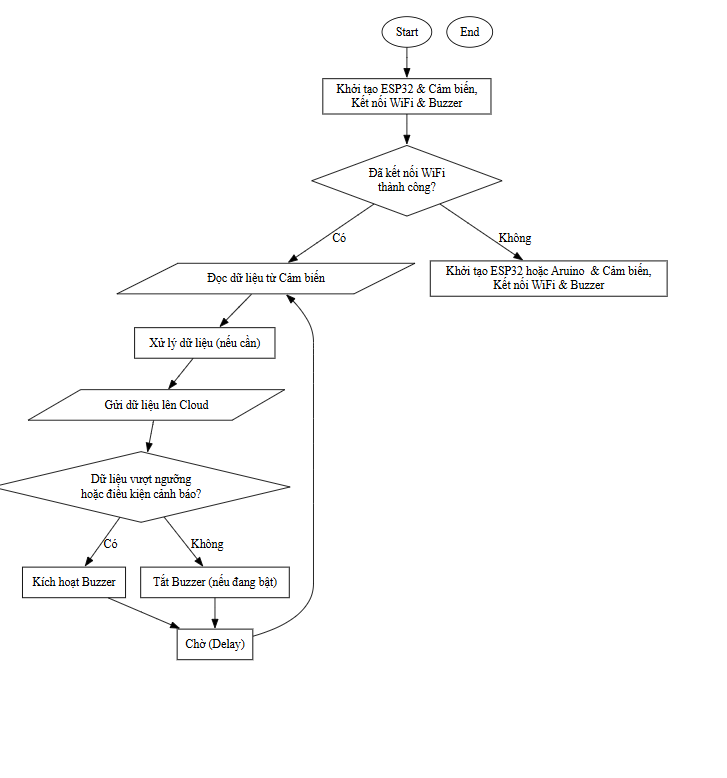
**- PIR Sensor: viết tắt của Passive Infrared Sensor, cảm biến phát hiện hồng ngoại từ cơ thể người.**

**- PushSafer: là dịch vụ miễn phí hỗ trợ gửi thông báo đến điện thoại thông qua Internet, hoạt động như một webhook. Không cần app riêng.**

**- MicroPython: là phiên bản thu gọn của Python, dùng trong thiết bị nhúng, rất phù hợp để lập trình ESP32.**

**IV.c**

**1.Sơ đồ khối hệ thống**



**2.Mô hình hoạt động**

**- Hệ thống phát hiện chuyển động hoạt động dựa trên cảm biến PIR. Khi có người đi qua vùng quét của cảm biến, tín hiệu được truyền đến vi điều khiển ESP32. Vi điều khiển sẽ điều khiển bật đèn LED, bật còi báo động (buzzer) và gửi thông báo đến điện thoại người dùng qua dịch vụ PushSafer. Mỗi lần phát hiện chuyển động, nếu đủ 10 giây kể từ lần trước, hệ thống sẽ gửi thông báo một lần để tránh spam.**

**3.Phân tích chức năng từng thành phần**

**- Cảm biến PIR (HC-SR501): phát hiện chuyển động dựa trên tia hồng ngoại từ cơ thể người.**

**- ESP32: xử lý tín hiệu đầu vào từ cảm biến, điều khiển LED, buzzer, và gửi HTTP request đến PushSafer.**

**- Đèn LED: hiển thị cảnh báo ánh sáng khi có chuyển động.**

**- Buzzer: phát âm thanh cảnh báo.**

**- PushSafer: dịch vụ đẩy thông báo đến điện thoại qua Internet mà không cần viết app riêng.**

**4.Thiết kế phần cứng**

**- PIR kết nối GPIO5 (tín hiệu), 3V3 (nguồn), GND (mass).**

**- LED nối qua điện trở đến GPIO6, đầu còn lại nối GND.**

**- Buzzer kết nối GPIO16 và GND.**

**- ESP32 được cấp nguồn qua cổng USB từ máy tính hoặc adapter 5V**

**5.Thiết kế phần mềm**

**- Sử dụng Thonny IDE để lập trình MicroPython cho ESP32.**

**- Chương trình chia thành các phần chính:**

**+ Kết nối WiFi.**

**+ Đọc tín hiệu cảm biến PIR.**

**+ Bật tắt LED và buzzer.**

**+ Gửi HTTP POST đến PushSafer API.**

**V. TRIỂN KHAI THỰC TẾ**

**1.Qúa trình lắp ráp**

**- Gắn các linh kiện trên breadboard.**

**- Nối đúng các chân tín hiệu với ESP32 theo sơ đồ phần cứng.**

**- Cắm ESP32 vào máy tính để lập trình và cấp nguồn.**

**2.Lập trình phần mềm**

**- Sử dụng Thonny IDE để flash firmware MicroPython.**

**- Viết chương trình với thư viện machine, network, urequests, time.**

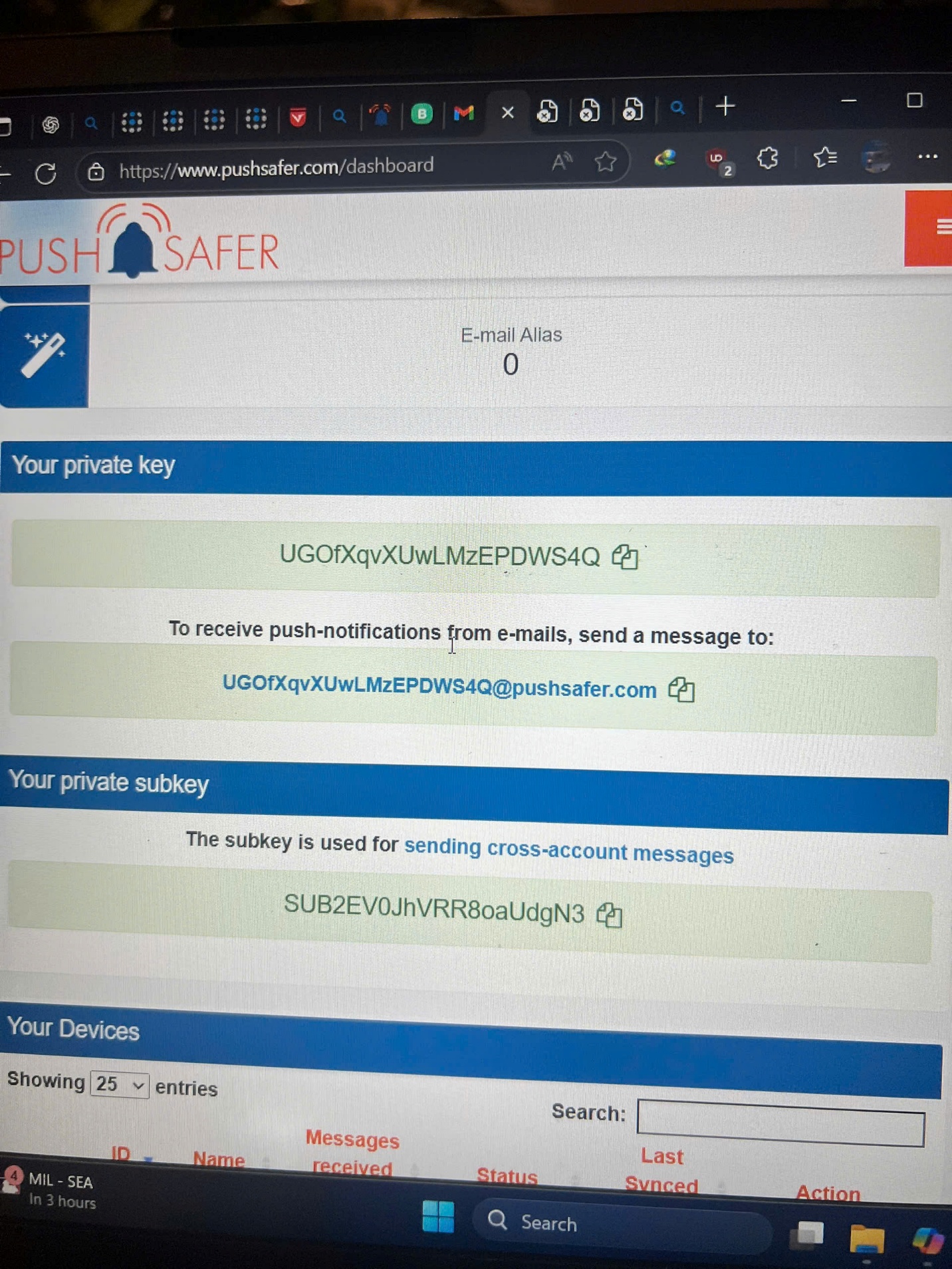
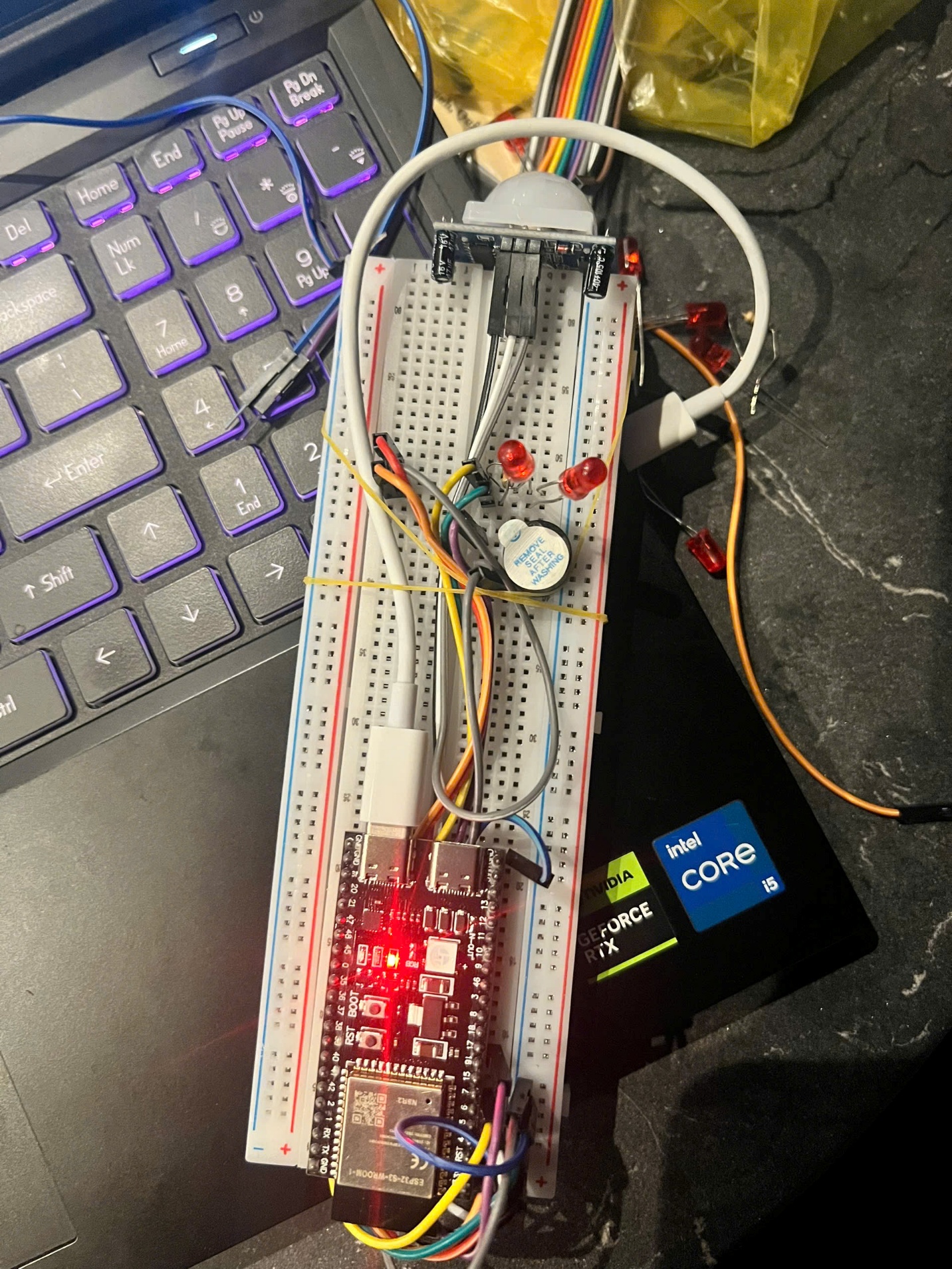
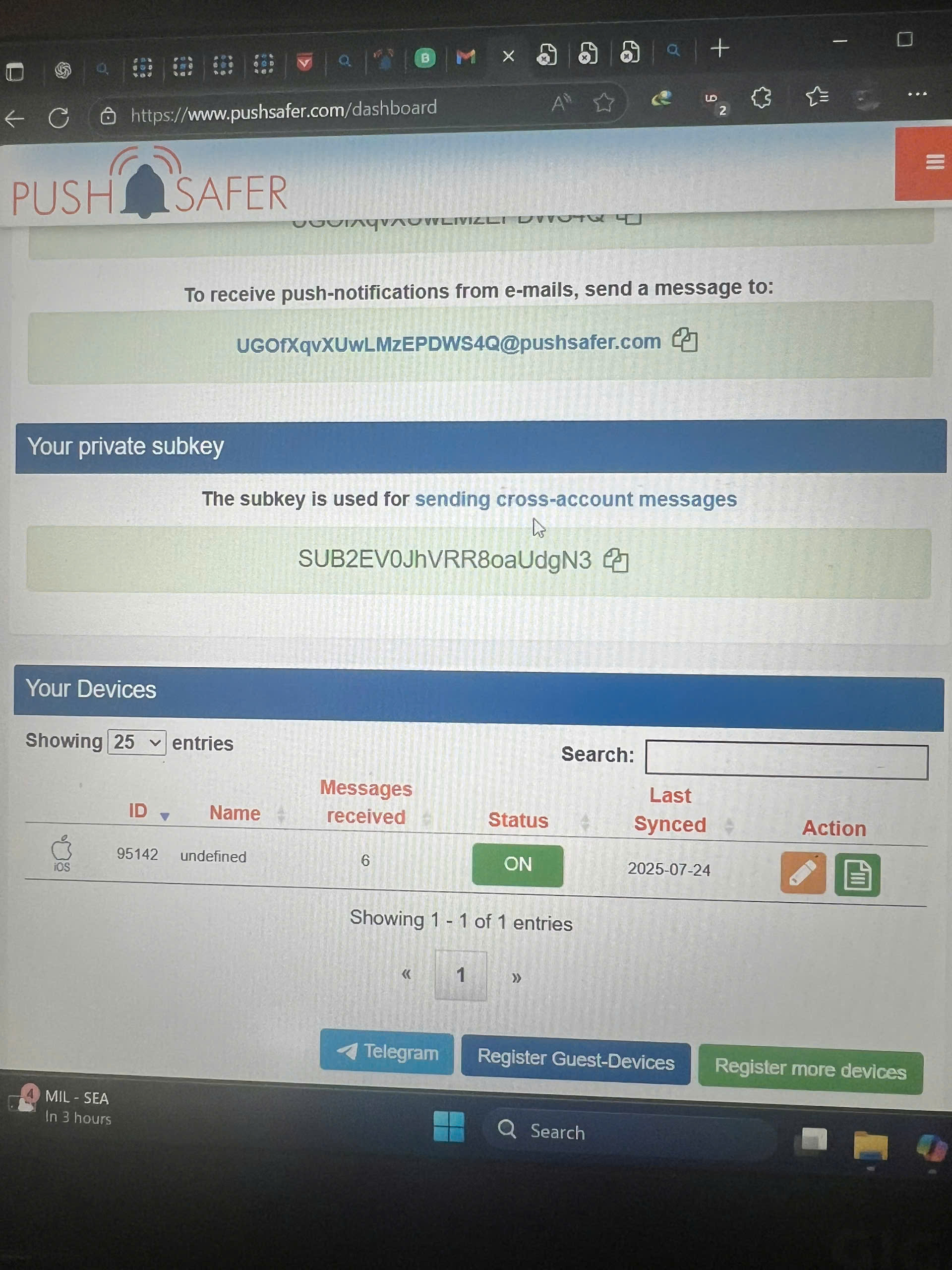
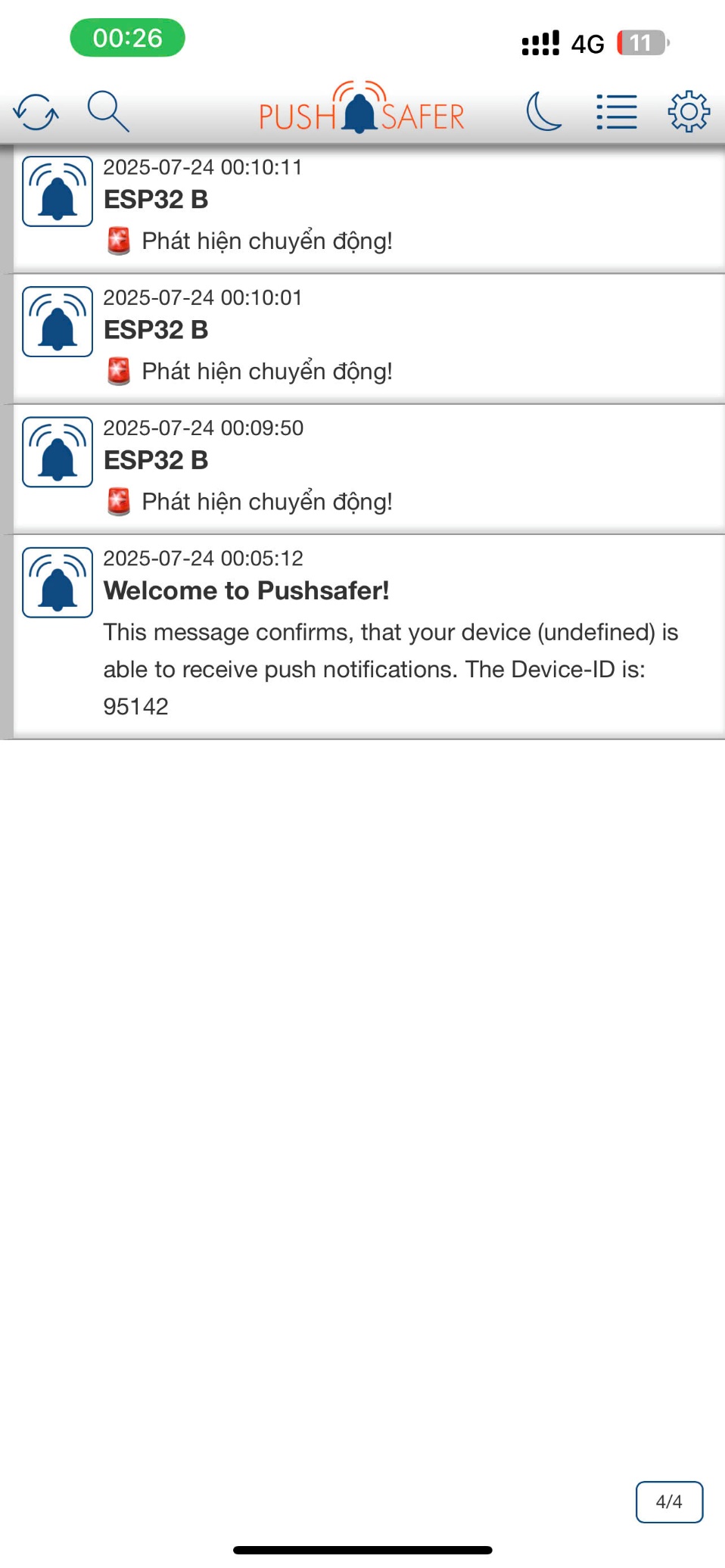
**- Chạy thử và theo dõi output qua terminal Thonny.**

**3.Kết nối IoT**

**- ESP32 kết nối WiFi theo SSID/password lập trình sẵn.**

**- Gửi thông báo về điện thoại qua PushSafer (dùng HTTP POST với private key).**

**4.Hình ảnh minh họa**



**VI. KẾT QUẢ VÀ ĐÁNH GIÁ**

**1.Kết quả thu được**

**- Phát hiện chuyển động chính xác.**

**- LED sáng và buzzer kêu đúng khi có người đi qua vùng quét.**

**- Thông báo gửi về điện thoại thành công với độ trễ thấp.**

**2.Đánh giá hiệu quả**

**- Hệ thống đơn giản, giá thành thấp, dễ thi công.**

**- Đáp ứng mục tiêu cảnh báo nhanh và từ xa.**

**- PushSafer ổn định và dễ dùng hơn so với nhiều app phức tạp như Blynk.**

**3.So sánh với mục tiêu**

**- Hệ thống đạt đầy đủ yêu cầu: phát hiện – cảnh báo – thông báo từ xa.**

**- Không cần viết app riêng, dùng API có sẵn để gửi tin.**

**VII. KHÓ KHĂN VÀ HƯỚNG PHÁT TRIỂN**

**1.Khó khăn**

**- Cảm biến PIR bị ảnh hưởng bởi ánh sáng mạnh hoặc gió.**

**- ESP32 đôi lúc kết nối WiFi chậm hoặc không ổn định.**

**- Cần tạo tài khoản PushSafer và lấy mã API riêng.**

**2.Hướng phát triển**

**- Kết hợp camera để chụp ảnh gửi kèm khi phát hiện chuyển động.**

**- Gửi thông báo đồng thời đến cả PushSafer, Telegram hoặc Google Sheets.**

**- Lắp cố định vào mô hình thực tế như nhà, lớp học, kho đồ...**

**VII. KẾT LUẬN**

**- Đề tài đã xây dựng thành công hệ thống báo động thông minh dùng ESP32 và cảm biến PIR. Hệ thống có khả năng cảnh báo bằng cả âm thanh, ánh sáng và gửi thông báo đến điện thoại người dùng thông qua PushSafer. Đây là một giải pháp đơn giản, hiệu quả, chi phí thấp, phù hợp với các ứng dụng giám sát cơ bản trong thực tế.**

**VIII.TÀI LIỆU THAM KHẢO**

- PushSafer. (n.d.). *Dịch vụ gửi thông báo đẩy đến điện thoại*. Truy cập tại: <https://www.pushsafer.com>

- Espressif Systems. (2024). *Tài liệu kỹ thuật vi điều khiển ESP32*. Truy cập tại: https://www.espressif.com/en/products/socs/esp32

- Datasheet cảm biến HC-SR501 PIR. (n.d.).

Truy cập tại: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/IR%20Motion%20Sensor.pdf