

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



BÁO CÁO
THỰC TẬP ĐỒ ÁN MÔN HỌC ĐA NGÀNH (CO3109)
ĐỀ TÀI
HỆ THỐNG BÁO ĐỘNG AN TOÀN CHO GIA ĐÌNH

Nhóm (Tổ) : 82
Học kỳ : 242 (Năm học 2024-2025)
Giảng viên hướng dẫn : Mai Đức Trung
Sinh viên thực hiện : Trương Minh Thông (L01) 2153005
Nguyễn Anh Duy (L02) 2233163
Nguyễn Thành Đạt (L02) 2012938
Nguyễn Ngọc Phú (L02) 2212588
Nguyễn Viết Ký (L03) 2151217

TP. Hồ Chí Minh, 31/05/2025

MỤC LỤC

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI	1
1.1. Hệ thống báo động an toàn cho gia đình	1
1.2. Yêu cầu của hệ thống	2
1.2.1. Yêu cầu chức năng	2
1.2.2. Yêu cầu phi chức năng	3
1.3. Phân công nhiệm vụ trong nhóm	3
CHƯƠNG 2. THIẾT KẾ HỆ THỐNG	4
2.1. Tổng quan hệ thống	4
2.2. Công nghệ sử dụng	6
CHƯƠNG 3. USE CASE CHI TIẾT VÀ SƠ ĐỒ HOẠT ĐỘNG, SƠ ĐỒ LỚP CỦA HỆ THỐNG	7
3.1. Sơ đồ Use Case của hệ thống	7
3.2. Đặc tả Use Case của hệ thống	7
3.3. Sơ đồ hoạt động của hệ thống	10
3.4. Sơ đồ lớp của hệ thống	10
CHƯƠNG 4. GIAO DIỆN NGƯỜI DÙNG VÀ DEMO SẢN PHẨM	11
4.1. Demo sản phẩm	11
4.2. Giao diện người dùng	11
4.3. Trang đăng nhập và xác thực	11
4.4. Trang chủ và điều hướng	11
4.5. Hiển thị cảnh báo	11
4.6. Báo cáo sự kiện	11
4.7. Quản lý cài đặt hệ thống	11
CHƯƠNG 5. KẾT LUẬN	12

Danh sách hình vẽ

1 Các thiết bị được chọn sử dụng trong hệ thống	4
---	---

Danh sách bảng

1 Phân công nhiệm vụ trong nhóm	3
2 Use Case 1: Thiết lập ngưỡng cảm biến	7
3 Use Case 2: Xem ngưỡng cảm biến	8
4 Use Case 3: Xóa ngưỡng cảm biến	8
5 Use Case 4: Theo dõi thông tin cảm biến	9
6 Use Case 5: Thống kê thông tin cảm biến	9

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

1.1. Hệ thống báo động an toàn cho gia đình

Hệ thống nhà thông minh (SmartHome) có lẽ là từ khóa quá quen thuộc với mọi người trong cuộc sống hiện đại ngày nay. Nhà thông minh (Smart Home) là một hệ thống tích hợp các thiết bị điện tử và công nghệ tiên tiến nhằm tạo ra một môi trường sống tiện nghi, an toàn và tiết kiệm năng lượng. Các thiết bị trong nhà thông minh có thể bao gồm đèn chiếu sáng, điều hòa nhiệt độ, hệ thống an ninh, và các thiết bị gia dụng khác, tất cả đều được kết nối và điều khiển thông qua mạng internet. Nhờ vào công nghệ Internet of Things (IoT), các thiết bị này có thể giao tiếp với nhau và với người dùng, cho phép điều khiển từ xa và tự động hóa các hoạt động hàng ngày. Hệ thống nhà thông minh không chỉ mang lại sự tiện lợi mà còn giúp tối ưu hóa việc sử dụng năng lượng, nâng cao mức độ an toàn và cải thiện chất lượng cuộc sống.

Trong đề án đa ngành này, nhóm chúng em sẽ thực hiện một hệ thống báo động an toàn cho gia đình (một mô hình SmartHome đơn giản) được thiết kế để giám sát môi trường trong nhà và cảnh báo khi phát hiện các tình huống nguy hiểm. Hệ thống sử dụng vi điều khiển Yolo:Bit cùng với các cảm biến để theo dõi nhiệt độ, độ ẩm, ánh sáng và chuyển động. Khi phát hiện nguy cơ, hệ thống sẽ kích hoạt cảnh báo bằng đèn LED RGB, hiển thị thông tin trên màn hình LCD và có thể gửi cảnh báo đến người dùng thông qua trang web trên thiết bị điện tử.

Mục tiêu của đề tài:

- Giám sát và phát hiện các yếu tố môi trường trong nhà như nhiệt độ, độ ẩm, ánh sáng, chuyển động.
- Đưa ra cảnh báo kịp thời khi có nguy cơ xảy ra cháy nổ, xâm nhập hoặc điều kiện môi trường bất thường.
- Tạo giao diện điều khiển trực quan trên màn hình LCD và nền tảng web để người dùng có thể dễ dàng giám sát và điều khiển hệ thống từ xa.
- Cung cấp khả năng mở rộng, nâng cấp để tích hợp thêm các cảm biến và thiết bị điều khiển khác.

Phạm vi đề tài:

- Ứng dụng trong các hộ gia đình, căn hộ nhỏ hoặc văn phòng có quy mô nhỏ.
- Hỗ trợ các thiết bị điều khiển cơ bản như đèn LED RGB, quạt mini, màn hình LCD.
- Kết nối qua giao tiếp không dây để hỗ trợ điều khiển từ xa thông qua remote hoặc nền tảng web.
- Không tích hợp chức năng giám sát hình ảnh qua camera hoặc hệ thống bảo mật chuyên sâu như khóa vân tay hoặc mã số.

Các yêu cầu chung của hệ thống:

- **Giám sát nhiệt độ và độ ẩm**

- Đọc dữ liệu từ cảm biến DHT20.
- Hiển thị thông tin lên màn hình LCD 16x2.
- Tự động kích hoạt quạt mini khi nhiệt độ vượt quá ngưỡng cài đặt.

- **Phát hiện chuyển động và khoảng cách**

- Sử dụng cảm biến hồng ngoại để phát hiện người hoặc vật thể di chuyển trong phạm vi giám sát.
- Khi phát hiện xâm nhập, hệ thống sẽ kích hoạt báo động bằng đèn LED RGB và hiển thị cảnh báo trên màn hình LCD.

- **Cảm biến ánh sáng**

- Đo mức độ ánh sáng trong phòng.
- Tự động bật/tắt đèn LED RGB khi trời tối/sáng.

- **Điều khiển từ xa**

- Sử dụng remote để bật/tắt quạt, đèn LED RGB.
- Cấu hình lại hệ thống bằng điều khiển từ xa.
- Hệ thống hỗ trợ một trang web dashboard để điều khiển thiết bị từ xa thông qua Internet.

- **Thông báo cảnh báo**

- Khi phát hiện nguy hiểm như nhiệt độ cao hoặc có người lạ xâm nhập, hệ thống sẽ kích hoạt cảnh báo nhấp nháy bằng đèn LED RGB.
- Hiển thị thông tin cảnh báo lên màn hình LCD.
- Báo cáo và ghi nhận các sự kiện xảy ra trong hệ thống.

1.2. Yêu cầu của hệ thống

1.2.1. Yêu cầu chức năng

- Hệ thống phải có khả năng thu thập dữ liệu từ các cảm biến và xử lý thông tin trong thời gian thực.
- Khi phát hiện tình huống nguy hiểm, hệ thống cần kích hoạt cảnh báo ngay lập tức.
- Giao diện điều khiển trên web phải có khả năng hiển thị dữ liệu và điều khiển thiết bị từ xa.
- Điều khiển quạt, đèn LED RGB từ xa thông qua trang web.
- Ghi nhận và lưu trữ lịch sử cảnh báo để người dùng có thể truy xuất khi cần thiết.

Quản lý lớp học

- Room info - Auto On/Off - Room sign up

- Dùng microphone điều khiển thiết bị - user profile management - API điều khiển thiết bị

Đặt lịch sử dụng phòng học

- Nhắc nhở lịch học - Quản lý lịch sử dụng (app dt)

Điều khiển thiết bị (voice/ 1 chạm)

- Theo dõi nhiệt độ, độ ẩm, ánh sáng (dashboard) - tự động/ tắt - báo khi có sự cố thiết bị - xem danh sách điểm danh

Điểm danh, quản lý học sinh (AI vision - số lượng ng trong phòng)

1.2.2. Yêu cầu phi chức năng

Đo được

- **Độ chính xác:** Cảm biến nhiệt độ có độ chính xác $\pm 2\%$ và cảm biến độ ẩm có độ chính xác $\pm 5\%$.
- **Thời gian phản hồi:** Hệ thống phải phản hồi trong vòng 1 giây sau khi nhận tín hiệu từ cảm biến hoặc remote. (Real time)
- **Độ bền:** Thiết bị phải hoạt động ổn định trong dải nhiệt độ từ 0°C đến 50°C .
- **Tiêu thụ điện năng:** Hệ thống sử dụng nguồn 5V, tối ưu để tiết kiệm năng lượng.
- **Dễ sử dụng:** Giao diện người dùng hiển thị rõ ràng, các nút điều khiển đơn giản, dễ thao tác, UI đẹp, dễ dùng
- **Khả năng mở rộng:** Hệ thống có thể dễ dàng tích hợp thêm các cảm biến và thiết bị điều khiển khác.

- Khả dụng: dễ sử dụng, học dùng trong 10 phút, dùng trên tất cả web browser - Bảo mật: mã hóa mật khẩu, data, nhập pass để xem thông tin - Có sẵn: hoạt động bất kỳ lúc nào 24/7, không được off quá 60p - Hiệu suất: xử lý thời gian thực, nhanh, chính xác - Mở rộng: có khả năng mở rộng - Tin cậy: dữ liệu thu được đảm bảo tin cậy - Bảo trì dễ: bảo trì theo tháng

1.3. Phân công nhiệm vụ trong nhóm

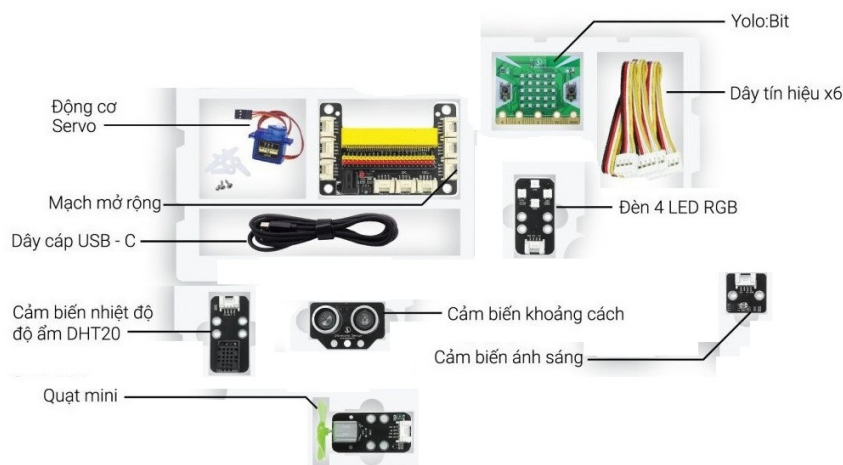
STT	Sinh viên thực hiện	Mã số sinh viên	Nhiệm vụ
1	Nguyễn Anh Duy	2233163	Front-End cho Web App
2	Nguyễn Thành Đạt	2012938	Code cho mạch YoloBit
3	Nguyễn Ngọc Phú	2212588	Code IoT Gateway
4	Trương Minh Thông	2153005	Viết báo cáo L ^A T _E X
5	Nguyễn Viết Ký	2151217	Back-End cho Web App

Bảng 1: Phân công nhiệm vụ trong nhóm

CHƯƠNG 2. THIẾT KẾ HỆ THỐNG

2.1. Tổng quan hệ thống

Thiết bị sử dụng - The **Micro:bit** acts as the core microcontroller, reading data from sensors and controlling actuators. - Sensors detect security-related events (e.g., motion, temperature, door status). - Actuators (e.g., alarms, lights, locks) respond to user commands. - The Micro:bit communicates with a computer via **serial connection** (USB or Bluetooth).



Hình 1: Các thiết bị được chọn sử dụng trong hệ thống

Thiết bị đầu vào Cảm biến nhiệt độ độ ẩm DHT20

- **Ứng dụng:** Đo nhiệt độ và độ ẩm trong môi trường. Hiển thị dữ liệu lên màn hình LCD. Kích hoạt quạt mini nếu nhiệt độ vượt quá ngưỡng cài đặt.
- **Đầu vào:** Nhiệt độ, độ ẩm - cảm biến được đặt trong môi trường giám sát.
- **Đầu ra:** Giá trị nhiệt độ, độ ẩm được gửi đến vi điều khiển và hiển thị trên màn hình LCD.

Cảm biến khoảng cách hồng ngoại

- **Ứng dụng:** Phát hiện vật thể trong phạm vi nhất định. Khi phát hiện người xâm nhập, kích hoạt báo động bằng đèn LED RGB hoặc hiển thị cảnh báo lên màn hình LCD.
- **Đầu vào:** Khoảng cách - cảm biến được lắp đặt tại vị trí giám sát.
- **Đầu ra:** Tín hiệu phát hiện chuyển động được gửi đến vi điều khiển để xử lý.

Cảm biến ánh sáng

- **Ứng dụng:** Đo mức độ ánh sáng môi trường. Khi trời tối, hệ thống có thể tự động bật đèn LED RGB và tắt khi trời sáng.

- **Đầu vào:** Cường độ ánh sáng - cảm biến được đặt tại môi trường ngoài trời hoặc trong phòng.
- **Đầu ra:** Tín hiệu cường độ ánh sáng được xử lý để điều khiển LED RGB.

Thiết bị đầu ra Động cơ Servo

- **Ứng dụng:** Điều khiển cửa tự động đóng/mở khi có tín hiệu từ remote hoặc cảm biến khoảng cách.
- **Đầu vào:** Tín hiệu điều khiển từ vi điều khiển.
- **Đầu ra:** Thay đổi góc mở cửa theo tín hiệu điều khiển.

Đèn 4 LED RGB

- **Ứng dụng:** Hiển thị trạng thái hệ thống (xanh: bình thường, đỏ: cảnh báo, nhấp nháy: báo động). Bật sáng khi trời tối và tắt khi trời sáng. Nhấp nháy khi có cảnh báo từ cảm biến khoảng cách hoặc nhiệt độ cao.
- **Đầu vào:** Tín hiệu điều khiển từ vi điều khiển.
- **Đầu ra:** Phát sáng theo màu sắc quy định.

Quạt mini

- **Ứng dụng:** Kích hoạt khi nhiệt độ vượt quá ngưỡng cài đặt (ví dụ: trên 30°C). Có thể bật/tắt từ remote.
- **Đầu vào:** Tín hiệu từ vi điều khiển hoặc người dùng.
- **Đầu ra:** Quạt hoạt động để làm mát môi trường.

Các thiết bị khác Yolo:Bit

- **Ứng dụng:** Vi điều khiển chính của hệ thống, xử lý dữ liệu từ cảm biến và điều khiển thiết bị đầu ra.
- **Đầu vào:** Dữ liệu từ các cảm biến.
- **Đầu ra:** Tín hiệu điều khiển thiết bị ngoại vi.

Mạch mở rộng

- **Ứng dụng:** Kết nối và cấp nguồn cho các thiết bị ngoại vi như cảm biến, động cơ, màn hình LCD, LED RGB.
- **Đầu vào:** Nguồn điện từ USB hoặc pin.
- **Đầu ra:** Phân phối nguồn và kết nối tín hiệu giữa các thiết bị.

Dây tín hiệu x6

- **Ứng dụng:** Truyền tín hiệu giữa Yolo:Bit và các thiết bị ngoại vi.
- **Đầu vào:** Kết nối giữa vi điều khiển và thiết bị ngoại vi.
- **Đầu ra:** Truyền dữ liệu và tín hiệu điều khiển.

Dây cáp USB-C

- **Ứng dụng:** Cấp nguồn và lập trình cho Yolo:Bit.
- **Đầu vào:** Nguồn điện từ adapter hoặc máy tính.
- **Đầu ra:** Cấp điện và truyền dữ liệu lập trình.

1. **Computer-side Python Script (Edge Processing & Cloud Communication)****
- Runs on a PC connected to the Micro:bit. - Collects sensor data from the Micro:bit and sends it to the **Adafruit IO server****. - Retrieves commands from the Adafruit IO server and forwards them to the Micro:bit. - Ensures real-time logging and processing of sensor data. 2. **Web Application (User Interface & Remote Control)**** - Allows users to **view live sensor data**** and **send commands**** to actuators. - Connects directly to **Adafruit IO**** for real-time data fetching and command sending. - Uses **Firebase**** for user authentication and actuator list management. 3. **Firebase (User Management & Device Configuration)**** - Stores user accounts, authentication details, and access permissions. - Maintains a list of registered actuators that each user can control. - Ensures secure access by verifying user identity before allowing device interaction.

Data Flow and Command Flow

Cập nhật dữ liệu mỗi 100ms

1. **Data Flow (Sensor to Web UI)**** - Micro:bit reads data from sensors. - Sends data to the **Python script**** on the computer via serial communication. - Python script **uploads the data to Adafruit IO****. - The web app fetches the data from Adafruit IO and displays it. 2. **Command Flow (User to Actuator)**** - User sends a command via the web application. - The command is **stored in Adafruit IO****. - The Python script running on the computer fetches the command. - It sends the command to the Micro:bit, which **activates the appropriate actuator****.

2.2. Công nghệ sử dụng

- React Native: Mobile App - ExpressJS: xây dựng API - Firebase - Google Cloud Service: API Text To Speech - YoloV8: AI Hình ảnh - Adafruit **Micro:bit**** - Handles sensors & actuators **Python (Computer-side Script)**** - Manages communication with Micro:bit & Adafruit **Adafruit IO**** - Cloud IoT service for data logging & command relay **Firebase**** - User authentication & actuator list storage **Web Technologies (HTML, CSS, JS)**** - User interface for remote monitoring & control Figma, Photo-shop thiết kế giao diện cho web app **Các hình ảnh diagram trong báo cáo vẽ bằng Lucidchart**** Báo cáo trên Overleaf Github lưu trữ mã nguồn host trang web này bằng github page

CHƯƠNG 3. USE CASE CHI TIẾT VÀ SƠ ĐỒ HOẠT ĐỘNG, SƠ ĐỒ LỚP CỦA HỆ THỐNG

3.1. Sơ đồ Use Case của hệ thống

Manager (admin)

- Điều chỉnh các thiết bị trong phòng học - User - Giáo viên - Học sinh

3.2. Đặc tả Use Case của hệ thống

Use Case	Quản lý hệ thống - Thiết lập ngưỡng cảm biến
Actors	Người quản lý hệ thống
Descriptions	Cho phép người quản lý hệ thống tạo ngưỡng giá trị của các cảm biến trong hệ thống báo động nhà thông minh.
Precondition	- Người quản lý hệ thống đã đăng nhập vào ứng dụng. - Hệ thống cảm biến đang hoạt động và có thể nhận dữ liệu.
Normal flow	1. Người quản lý hệ thống mở ứng dụng và chọn "Cấu hình cảm biến". 2. Hệ thống hiển thị danh sách các cảm biến có thể thiết lập ngưỡng. 3. Người quản lý chọn một cảm biến cụ thể (nhiệt độ, độ ẩm, ánh sáng, chuyển động). 4. Người quản lý nhập giá trị ngưỡng tối đa và tối thiểu cho cảm biến. 5. Hệ thống lưu giá trị ngưỡng vào cơ sở dữ liệu. 6. Hệ thống xác nhận rằng thiết lập đã được cập nhật thành công.
Alternative flow	Alternative 1 ở Bước 4: - Nếu giá trị nhập vào vượt quá giới hạn hợp lệ (ví dụ: nhiệt độ ngưỡng vượt quá 100°C), hệ thống sẽ báo lỗi và yêu cầu nhập lại.
Exceptions	Exception ở Bước 2: - Nếu hệ thống không thể kết nối với cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi và không thể thực hiện thao tác. Exception ở Bước 6: - Nếu xảy ra lỗi khi lưu giá trị ngưỡng, hệ thống sẽ yêu cầu thử lại hoặc kiểm tra kết nối.

Bảng 2: Use Case 1: Thiết lập ngưỡng cảm biến

Use Case	Quản lý hệ thống - Xem ngưỡng cảm biến
Actors	Người quản lý hệ thống
Descriptions	Cho phép quản lý hệ thống xem ngưỡng giá trị của các cảm biến trong hệ thống báo động nhà thông minh.
Precondition	- Người quản lý hệ thống đã đăng nhập vào ứng dụng. - Hệ thống cảm biến đang hoạt động và có thể xuất dữ liệu.

Normal flow	<ol style="list-style-type: none"> 1. Người quản lý hệ thống mở ứng dụng và chọn "Cấu hình cảm biến". 2. Hệ thống hiển thị danh sách các cảm biến và giá trị ngưỡng hiện tại. 3. Người quản lý kiểm tra thông tin và có thể lựa chọn chỉnh sửa hoặc xóa. 4. Người quản lý thoát màn hình xem ngưỡng hoặc quay lại màn hình chính.
Alternative flow	Alternative 1 ở Bước 3: - Nếu không có dữ liệu ngưỡng được thiết lập trước đó, hệ thống sẽ hiển thị thông báo "Chưa có ngưỡng nào được thiết lập".
Exceptions	Exception ở Bước 2: - Nếu hệ thống không thể kết nối với cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi và không thể thực hiện thao tác.

Bảng 3: Use Case 2: Xem ngưỡng cảm biến

Use Case	Quản lý hệ thống - Xóa ngưỡng cảm biến
Actors	Người quản lý hệ thống
Descriptions	Cho phép quản lý hệ thống tạo, xem và xóa ngưỡng giá trị của các cảm biến trong hệ thống báo động nhà thông minh.
Precondition	- Người quản lý hệ thống đã đăng nhập vào ứng dụng. - Hệ thống cảm biến đang hoạt động và có thể nhận dữ liệu.
Normal flow	<ol style="list-style-type: none"> 1. Người quản lý hệ thống mở ứng dụng và chọn "Cấu hình cảm biến". 2. Hệ thống hiển thị danh sách các cảm biến có thể thiết lập. 3. Người quản lý chọn một cảm biến cụ thể để xóa ngưỡng. 4. Hệ thống yêu cầu xác nhận trước khi xóa. 5. Người quản lý xác nhận xóa ngưỡng. 6. Hệ thống xóa dữ liệu ngưỡng khỏi cơ sở dữ liệu. 7. Hệ thống hiển thị thông báo xác nhận xóa thành công.
Alternative flow	Alternative 1 ở Bước 5: - Nếu quản lý không xác nhận xóa, hệ thống sẽ hủy thao tác và quay lại màn hình trước đó.
Exceptions	Exception ở Bước 2: - Nếu hệ thống không thể kết nối với cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi và không thể thực hiện thao tác.

Bảng 4: Use Case 3: Xóa ngưỡng cảm biến

Use Case	Theo dõi thông tin cảm biến
Actors	Người quản lý hệ thống
Descriptions	Hệ thống cho phép quản lý hệ thống theo dõi, thống kê và xuất báo cáo dữ liệu cảm biến theo từng tháng hoặc quý.
Precondition	- Quản lý hệ thống đã đăng nhập vào ứng dụng. - Dữ liệu cảm biến đã được ghi nhận trong hệ thống.

Normal flow	<ol style="list-style-type: none">1. Quản lý hệ thống mở ứng dụng và chọn "Thông kê cảm biến".2. Hệ thống hiển thị các tùy chọn thời gian: tháng/quý/năm.3. Quản lý chọn khoảng thời gian cần xem (ví dụ: tháng 1 năm 2025, quý 1 năm 2025).4. Hệ thống truy xuất dữ liệu cảm biến tương ứng trong khoảng thời gian đó.5. Hệ thống hiển thị biểu đồ, số liệu thống kê về mức độ thay đổi của các cảm biến (nhiệt độ, độ ẩm, ánh sáng, chuyển động).6. Quản lý có thể chọn xem chi tiết từng chỉ số cảm biến.7. Quản lý thoát màn hình thống kê hoặc tiếp tục thao tác khác.
Alternative flow	<p>Alternative 1 ở Bước 3:</p> <p>- Nếu không có dữ liệu trong khoảng thời gian đã chọn, hệ thống hiển thị thông báo "Không có dữ liệu cảm biến nào được ghi nhận trong khoảng thời gian này".</p> <p>Alternative 2 ở Bước 5:</p> <p>- Quản lý có thể lọc dữ liệu theo loại cảm biến cụ thể (chỉ hiển thị nhiệt độ hoặc chỉ hiển thị độ ẩm).</p>
Exceptions	<p>Exception ở Bước 4:</p> <p>- Nếu hệ thống không thể truy xuất dữ liệu từ cơ sở dữ liệu, hệ thống hiển thị thông báo lỗi và yêu cầu thử lại.</p>

Bảng 5: Use Case 4: Theo dõi thông tin cảm biến

Use Case	Thông kê thông tin cảm biến
Actors	Người quản lý hệ thống
Descriptions	Hệ thống cho phép quản lý hệ thống theo dõi, thống kê và xuất báo cáo dữ liệu cảm biến theo từng tháng hoặc quý.
Precondition	<ul style="list-style-type: none">- Quản lý hệ thống đã đăng nhập vào ứng dụng.- Dữ liệu cảm biến đã được ghi nhận trong hệ thống.
Normal flow	<ol style="list-style-type: none">1. Quản lý hệ thống chọn "Xuất báo cáo" trong màn hình thống kê.2. Hệ thống cho phép chọn định dạng báo cáo (PDF, CSV, Excel).3. Hệ thống tổng hợp dữ liệu và tạo file báo cáo.4. Quản lý tải xuống báo cáo hoặc gửi báo cáo qua email.5. Hệ thống hiển thị thông báo hoàn tất quá trình xuất báo cáo.6. Quản lý thoát màn hình hoặc tiếp tục thao tác khác.
Alternative flow	<p>Alternative 1 ở Bước 3:</p> <p>- Quản lý có thể chọn khoảng thời gian tùy chỉnh (ví dụ: từ ngày 10/01/2025 đến 15/03/2025).</p>
Exceptions	<p>Exception ở Bước 3:</p> <p>- Nếu quá trình tạo báo cáo thất bại, hệ thống sẽ hiển thị thông báo lỗi và cho phép thử lại sau.</p>

Bảng 6: Use Case 5: Thống kê thông tin cảm biến

3.3. Sơ đồ hoạt động của hệ thống

User | System | Other server

Đăng ký/Xóa lịch sd phòng

Chính tay thiết bị

Thiết lập thời gian tắt cho thiết bị

Tự động bật/tắt thiết bị

Thêm xóa sửa thiết bị

Sửa hồ sơ người dùng

Điều khiển thiết bị bằng giọng nói (AI)

Kiểm tra số lượng người trong phòng (AI)

Validate input, notify user

Xem thông tin phòng

Xem history dùng phòng

Gửi phản hồi

Thêm sửa xóa account

3.4. Sơ đồ lớp của hệ thống

- View - Controller - External API - Model - Database

CHƯƠNG 4. GIAO DIỆN NGƯỜI DÙNG VÀ DEMO SẢN PHẨM

4.1. Demo sản phẩm

Link video drive

Báo cáo LaTeX và Mã nguồn Web App: https://github.com/yymt242/DADN242_Group82

Trang web điều khiển được host tại: https://yymt242.github.io/DADN242_Group82/

Email và Mật khẩu đăng nhập admin:

Yolobit Dùng trình duyệt Google Chrome > Lập trình cho Yolobit tại trang web: <https://app.ohstem.vn/>

Cập nhật firmware?

Nạp chương trình?

Reset: Nhấn nút A 3s > nhấn nút Reset 1s (vẫn giữ nút A) > Thả nút A

Chạy chương trình IoT Gateway (Python trên Windows)

Trên Terminal, gõ:

```
1 py -m pip install --upgrade pip
2 pip install requests pyserial
3 py .\gateway.py
```

Để tắt, nhấn Ctrl + C

4.2. Giao diện người dùng

4.3. Trang đăng nhập và xác thực

4.4. Trang chủ và điều hướng

4.5. Hiển thị cảnh báo

4.6. Báo cáo sự kiện

4.7. Quản lý cài đặt hệ thống

Đăng nhập đang ký

Quản lý thiết bị (thêm sửa xóa)

Xem dữ liệu

Acc cá nhân



CHƯƠNG 5. KẾT LUẬN

Tự đánh giá

Học lập trình web cơ bản (html, css, JavaScript), và 1 vài công nghệ/framework cho backend/frontend

Học python, javascript > OOP > design pattern

Hướng phát triển tương lai

TÀI LIỆU THAM KHẢO

- [Dal08] Dalgaard, P. *Introductory Statistics with R*. Springer, 2008.
- [KZ14] Kenett, R. S., and Zacks, S. *Modern Industrial Statistics: With Applications in R, MINITAB, and JMP*. 2nd ed., John Wiley and Sons, 2014.
- [Ker15] Kerns, G. J. *Introduction to Probability and Statistics Using R*. 2nd ed., CRC Press, 2015.
- [Ein] Albert Einstein. *Wikipedia*. Retrieved from http://en.wikipedia.org/wiki/Albert_Einstein.