

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



ĐỒ ÁN ĐA NGÀNH - ĐỊNH HƯỚNG AI

Dự án

Nhà thông minh tích hợp mô hình phát hiện con người

GVHD: Trần Thanh Bình
Nhóm 15 Lê Minh Hoàng - 1913424
Nguyễn Trần Phước Thành - 1912059
Hoa Phương Tùng - 2012385
Nguyễn Hoàn Thành - 1915146
Trương Công Hoàng - 1913459

TP. HỒ CHÍ MINH, THÁNG 12/2023

Mục lục

1	Giới thiệu	3
1.1	Xu thế phát triển của IOT, nhà thông minh và lý do chọn đề tài	3
1.2	Dự án của nhóm	4
1.2.1	Tính năng	4
1.2.2	Cảm biến được sử dụng	4
1.2.3	Các thiết bị được điều khiển	4
1.2.4	Công nghệ được sử dụng	5
2	Đặc tả hệ thống	5
2.1	Mô hình tổng thể và hoạt động	5
2.2	Các use case	6
3	Chi tiết hiện thực	7
3.1	Phần cứng ngoại biên	7
3.1.1	Yolo:bit	7
3.1.2	ESP32-CAM	7
3.1.2.a	Mô tả	7
3.1.2.b	Áp dụng và hệ thống	7
3.2	Server	8
3.2.1	Hosting	8
3.2.2	Cơ sở dữ liệu	8
3.2.3	API	9
3.2.4	Mô hình AI	9
3.2.4.a	Tổng quan về mô hình neural network	9
3.2.4.b	Tổng quan về mô hình nơ ron tích chập (CNN)	10
3.2.4.c	Giới thiệu về YOLOv5	11
3.2.4.d	Cấu trúc mô hình	11
3.2.4.e	Đặc điểm chính	12
3.2.4.f	Ưu điểm	12
3.2.4.g	Vai trò và hoạt động trong hệ thống	12
3.2.5	Các listener liên kết thành phần	12
3.3	UI	13
3.3.1	Web UI	13
3.3.1.a	Trang login	13
3.3.1.b	Trang Home	13
3.3.1.c	Trang Room	15
3.3.1.d	Trang Analytics	17
3.3.2	Mobile UI	19
4	Kết luận	21
5	Mã nguồn và demo	22

Danh sách hình vẽ

1	Mô hình khối tổng thể của hệ thốn	5
2	Sơ đồ use case	6
3	Sơ đồ cơ sở dữ liệu quan hệ	8
4	kernel size 3x3	10
5	Phép tính convolution	10
6	Tác dụng một số loại kernel	11
7	Các listener của hệ thống	12
8	Giao diện trang login	13
9	Giao diện Homepage	14
10	Giao diện Homepage responsive (1)	14
11	Giao diện Homepage responsive (2)	15
12	Giao diện trang Room	15
13	Giao diện trang Living room	16
14	Giao diện trang Room responsive	16
15	Giao diện trang Living room responsive	17
16	Giao diện trang Analytics chế độ xem nhiệt độ	17
17	Giao diện trang Analytics chế độ xem độ ẩm	18
18	Giao diện trang Analytics responsive	18
19	Giao diện Mobile trang Login	19
20	Giao diện trang Home Living room (1)	20
21	Giao diện trang Home Living room (2)	21

1 Giới thiệu

1.1 Xu thế phát triển của IOT, nhà thông minh và lý do chọn đề tài

IoT viết tắt cho cụm từ “Internet of Things” là một khái niệm công nghệ liên quan đến việc kết nối các thiết bị điện tử thông qua Internet, cho phép chúng có thể giao tiếp, thu thập và chia sẻ dữ liệu mà không cần sự tương tác trực tiếp của con người. Khái niệm IoT được ra đời từ năm 1999, tuy nhiên phải trải qua gần 20 năm, nhờ sự phát triển đầy đủ về phần cứng, các phương thức kết nối thì nó mới phát huy được hết tiềm năng, những ứng dụng của nó vào các hoạt động thực tiễn và kinh doanh. Cụm từ này được nhắc đến rất nhiều trên các diễn đàn công nghệ trong các năm trở lại đây, đặc biệt sau đại dịch Covid-19. Khi sự tương tác giữa con người và con người ngày càng hạn chế thì IoT mang đến cho chúng ta 1 giải pháp tuyệt vời để giao tiếp với các đồ vật, thiết bị và tận hưởng sự phục vụ mà chúng mang lại cho cuộc sống hiện đại. Mặc dù năm 2020 đang ở phía sau chúng ta, nhưng nó đã để lại những luồng gió mạnh mẽ sẽ tiếp tục phân nhánh cho đến năm 2023.

Một hệ thống IoT thường có ba thành phần:

- **Thiết bị thông minh:** Đây là một thiết bị bất kỳ ở trong nhà như đèn, quạt, tivi có khả năng điện toán, kết nối internet. Thiết bị này thu thập dữ liệu từ môi trường xung quanh, thao tác nhập liệu của người dùng hoặc mô thức sử dụng và truyền cũng như nhận dữ liệu qua Internet từ ứng dụng IoT của nó.
- **Ứng dụng IoT:** Là phần mềm quản lý có giao diện người dùng, thông qua ứng dụng IoT người dùng có thể dễ dàng tương tác với các thiết bị, đọc các dữ liệu thống kê mà thiết bị đã thu thập. Ứng dụng này có thể tích hợp AI để phân tích các dữ liệu và đưa ra quyết định sáng suốt nhằm hỗ trợ cho người dùng
- **Server:** Là nơi trung gian cho quá trình trao đổi dữ liệu giữa các thiết bị và người dùng, Server nhận giữ liệu từ thiết bị, chuyển đổi, lưu trữ và gửi cho người dùng và ngược lại.

Như đã đề cập ở trên, xu hướng của trong việc ứng dụng IoT hiện vẫn tiếp tục tăng trong năm 2023 và không có dấu hiệu dừng lại. Ta có thể bắt gặp các ứng dụng về IoT trong công nghiệp, sản xuất khi cảm biến, có thể giúp các nhà quản lý xác định sớm các vấn đề về máy móc. Các vấn đề được xác định trước khi các kỹ thuật viên được gửi đi để giải quyết các vấn đề. Cảm biến rất tiện dụng cho những người xử lý bảo trì phòng ngừa. IoT đã và đang tìm kiếm các ứng dụng trong các nhà máy thông minh. Việc sử dụng thiết bị đeo mang đến cho công nhân một bức tranh rõ ràng về điều kiện và sự an toàn của nhà máy.

Ngoài ra ngành công nghiệp chăm sóc sức khỏe đã nghiên cứu công nghệ IoT trong nhiều năm nay. Các chuyên gia nhận thấy lĩnh vực này dẫn đầu trong việc áp dụng và đổi mới IoT. Chăm sóc sức khỏe hiện sử dụng các cảm biến và thiết bị đeo được, theo dõi và công nghệ điều hướng trong nhà. Giống như các nhà máy, các cơ sở chăm sóc sức khỏe là những thiết bị khổng lồ và hiện đại. Khả năng cải thiện khả năng kiểm soát ánh sáng và nhiệt độ của IoT sẽ cải thiện những địa điểm này. Ánh sáng đóng một vai trò quan trọng trong quá trình hồi phục của bệnh nhân. IoT có thể tăng cường hệ thống chiếu sáng bằng cách liên kết chúng với màn hình sức khỏe và triển khai các cảm biến.

Các thành phố thông minh cũng dần hình thành khi ứng dụng công nghệ này, ví dụ như Mỹ đang sử dụng IoT để kết nối các tiện ích, đồng hồ đỗ xe và đèn giao thông.

Ngoài các ứng dụng có vẻ có tầm vĩ mô, IoT bây giờ gần như có thể bắt gặp ở mọi nơi, ở mọi gia đình, mọi khu dân cư với các đồng hồ thông minh đo nhịp thông thông báo tình hình sức

khỏe, ô tô thông minh có chức năng tự lái như Tesla, các căn hộ thông minh giúp gia chủ quản lý cả ngôi nhà với chỉ một chiếc smartphone,...

Về nhà thông minh, Việt Nam hiện đang có 3,2 triệu hộ gia đình thông minh và Statista đưa ra dự đoán tỷ lệ thâm nhập của các thiết bị smarthome tại Việt Nam sẽ đạt 22,6% vào năm 2027. Như vậy, sau 4 năm tới, cả nước sẽ có khoảng 6 triệu hộ gia đình thông minh.

Vì công nghệ IoT nói chung và smarthome đang phát triển nhanh như thế, nhằm bắt kịp xu thế của thời đại cũng như chuẩn bị trước cho sự nghiệp tương lai của các lập trình viên như bọn em, bọn em quyết định chọn đề tài “Nhà thông minh tích hợp” cho môn “Đồ án đa ngành” như một cơ hội tìm hiểu và học hỏi, thử sức trước với công nghệ IoT đang trên đà phát triển này.

1.2 Dự án của nhóm

1.2.1 Tính năng

Phát triển từ tài liệu hướng dẫn được cung cấp và kiến thức tìm hiểu thêm, đồng thời cũng bị giới hạn bởi phần cứng được cung cấp và yêu cầu dự án từ môn học, nhóm quyết định các chức năng của dự án sẽ bao gồm:

- Điều khiển cửa - với nhận diện mặt và cảm biến tiệm cận (siêu âm)
- Điều khiển các thiết bị khác:
 - Quạt (PWM)
 - Thiết bị bất kỳ (sử dụng relay)
 - Đèn RGB LED - sử dụng AI và cảm biến PIR để dò hiện diện của người
- Tưới cây
- Giám sát môi trường
 - Độ ẩm đất
 - Nhiệt độ/độ ẩm không khí

1.2.2 Cảm biến được sử dụng

Ứng với các yêu cầu trên, hệ thống sẽ sử dụng các cảm biến

- Cảm biến PIR (cung cấp bởi trường - đi cùng với Yolo:bit)
- Cảm biến nhiệt độ và độ ẩm không khí (cung cấp bởi trường - đi cùng với Yolo:bit)
- Cảm biến độ ẩm đất (cung cấp bởi trường - đi cùng với Yolo:bit)
- Cảm biến khoảng cách siêu âm (cung cấp bởi trường - đi cùng với Yolo:bit)
- Camera (ESP32-CAM tự cung cấp)

1.2.3 Các thiết bị được điều khiển

Các thiết bị được điều khiển bởi hệ thống, cấu thành nên nhà thông minh bao gồm:

- Relay (cung cấp bởi trường - đi cùng với Yolo:bit)
- Fan (cung cấp bởi trường - đi cùng với Yolo:bit)

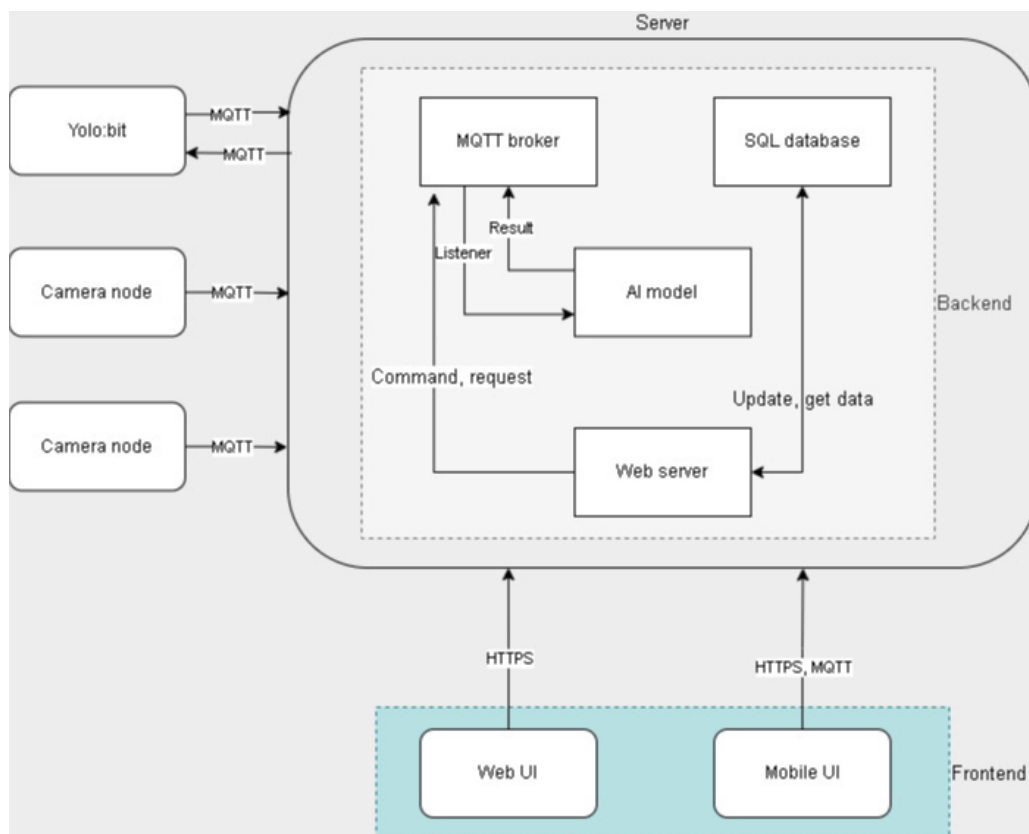
- Servo (cung cấp bởi trường - đi cùng với Yolo:bit)
- Buzzer (cung cấp bởi trường - đi cùng với Yolo:bit)
- RGB LED (cung cấp bởi trường - đi cùng với Yolo:bit)
- Pump (cung cấp bởi trường - đi cùng với Yolo:bit)

1.2.4 Công nghệ được sử dụng

- Web: Apache, PHP, Bootstrap 5, JS
- Mobile: React-native
- Database: MySQL
- Cloud service: Thuê máy, thiết lập DNS, mua tên miền sử dụng dịch vụ được cung cấp bởi AWS.

2 Đặc tả hệ thống

2.1 Mô hình tổng thể và hoạt động



Hình 1: Mô hình khối tổng thể của hệ thống

Thiết kế của hệ thống có thể được miêu tả một phần bằng mô hình client-server. Trong đó server vừa đóng vai trò lưu trữ (cơ sở dữ liệu), vừa đóng vai trò tính toán tập trung (thực hiện chạy các mô hình AI).

Một điểm đặc biệt cần chú ý ở đây: việc sử dụng giao thức MQTT cũng biến server thành một đầu mối chuyển tiếp thông tin - các thiết bị đầu cuối khi gửi/nhận thông tin phải đi qua trung gian là server.

Các thành phần của hệ thống của hệ thống có thể được phân loại vào 3 nhóm chính:

- (Nằm trong) Server: Cơ sở dữ liệu, mô hình AI, MQTT broker, web server (có REST API thực hiện thức năng), các listener đóng vai trò nối kết các thành phần lại.
- Phần cứng ngoại biên: Yolo:bit (cùng với các phần cứng đi kèm), ESP32-CAM.
- UI: Giao diện cho người dùng, bao gồm giao diện trên Web và trên di động.

2.2 Các use case



Hình 2: Sơ đồ use case

Chức năng cơ bản mà hệ thống cung cấp sẽ là việc xem và điều chỉnh trạng thái của các thiết bị. Đây cũng chính là use case mà nhóm tập trung làm nhiều nhất.

3 Chi tiết hiện thực

Trong phần này, các chi tiết của việc hiện thực một số thành phần đáng nói trong hệ thống

3.1 Phần cứng ngoại biên

3.1.1 Yolo:bit

Các nội dung trong phần này được lấy từ [file đặc tả lưu trên github](#).

Ta đặt ra các tín hiệu đầu vào cho Yolo:bit và vị trí kết nối của chúng trên mạch:

- PIR sensor - P0
- Temp and humidity (DHT20) - I2C1 port
- Soil humidity - P1 (ADC) port
- Ultra sonic - dual port P3/P6
- Camera - không được nối thẳng vào mạch, nhưng cung cấp thông tin điều khiển hoạt động của mạch

Tương tự như vậy cho các thiết bị điều khiển đầu ra:

- Relay - P2
- Fan - P14/P15
- Servo - header P4
- Buzzer - onboard
- RGB LED - P16/P12
- Pump - dual port P10/P13

3.1.2 ESP32-CAM

3.1.2.a Mô tả

Esp32-Cam là một module dựa trên SoC (System on Chip) ESP32 mạnh mẽ, được thiết kế để có thể gắn Module Camera để thực hiện các dự án với thu thập hình ảnh, stream video chất lượng trung bình, phù hợp cho một số dự án IoT về giám sát và theo dõi.

3.1.2.b Áp dụng và hệ thống

Module Esp32-Cam có thể kết nối wifi và stream trực tiếp hình ảnh ở localhost. Tuy vậy, nhóm gặp khó khăn khi restream lại luồng trực tiếp của camera từ server. Do đó, nhóm đã thực hiện một giải pháp khác là cập nhật hình ảnh từ camera mỗi giây và lưu lại. Hình ảnh này được gửi đi bằng giao thức MQTT [6]. Vì vậy, có thể xem như camera cũng là một loại "cảm biến" trong hệ thống. Sau đó, phía server sẽ xử lý hình ảnh này để hiển thị cho người dùng và điều khiển một số thiết bị.

3.2 Server

3.2.1 Hosting

Các phần mềm ứng dụng của dự án sẽ được chạy trên phần cứng được cung cấp bởi Amazon Web Service (AWS).

AWS là nhà cung cấp dịch vụ điện toán đám mây theo nhu cầu (on-demand cloud computing service), với phạm vi dịch vụ đa dạng - tính toán, lưu trữ, mạng, cơ sở dữ liệu, quản lý, học máy,... Nói một cách đơn giản, AWS cho thuê khả năng tính toán và mạng để người dùng hiện thực ứng dụng mà không cần phải lo lắng tới việc quản lý phần cứng.

Quy trình và cách thức thao tác với AWS được tham khảo trên trang tài liệu của AWS [3].

Hiện thực server của nhóm được cho chạy trên dịch vụ EC2 của AWS, với địa chỉ IPv4 tĩnh được cấp phát miễn phí. Với địa chỉ IP này, ta đã có thể kết nối với máy qua mạng.

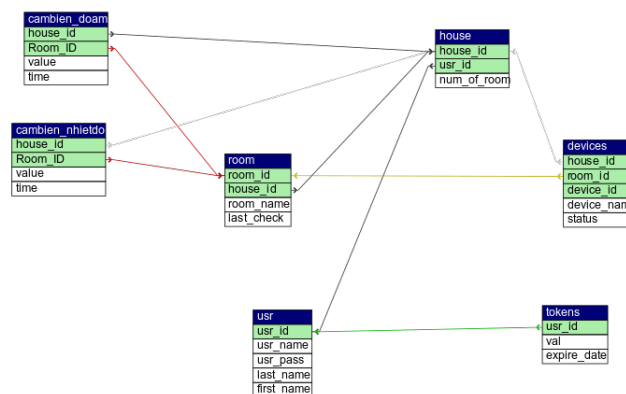
Nhưng không dừng lại ở đó, để tăng tính chuyên nghiệp, và đồng thời cũng là cơ hội học hỏi thêm, nhóm đã mua tên miền bk-hk231-dadn-smarthome.link. Dịch vụ mua tên miền cũng được AWS cung cấp. Sau đó, các DNS record liên quan được đưa lên mạng lưới DNS server của AWS. Tối đây, ta đã có thể kết nối tới server qua tên miền ở trên.

Sau khi cài đặt các thành phần của LAMP (Linux, Apache, MySQL, PHP), phpMyAdmin, [Eclipse Mosquitto](#) MQTT broker. Ta đã có một server tương đối hoàn chỉnh, với khả năng truy cập từ xa vào các thành phần chính (web server, MQTT broker, cơ sở dữ liệu).

Để có thể kết nối với web server qua HTTPS, nhóm sử dụng [Let's Encrypt](#) certificate authority để đăng ký certificate cho web server. Sau khi hoàn thành bước này, việc truy cập vào web UI sẽ không còn bị trình duyệt hiện đại cảnh báo không an toàn nữa.

3.2.2 Cơ sở dữ liệu

Cơ sở dữ liệu mà nhóm đã tạo ra nhằm phục vụ cho việc lưu trữ những dữ liệu cần thiết trong một hệ thống smart home có sơ đồ như sau:



Hình 3: Sơ đồ cơ sở dữ liệu quan hệ

Cụ thể:

1. Bảng **house** : chứa khóa đánh dấu nhà của người dùng.
2. Bảng **room** : chứa khóa đánh dấu từng căn phòng trong mỗi ngôi nhà của người dùng cùng tên phòng và thời gian người dùng kiểm tra phòng lần cuối.
3. Bảng **usr** : chứa thông tin người dùng.
4. Bảng **tokens** : chứa thông tin dùng để theo dõi phiên đăng nhập của người dùng
5. Bảng **devices** : các thiết bị trong từng phòng.
6. Bảng **cambien_nhietdo** và **cambien_doam** : giá trị cảm biến nhiệt độ và độ ẩm gửi về từ mỗi phòng.

3.2.3 API

Để thuận tiện cho việc lấy dữ liệu từ database cho cả ứng dụng web và ứng dụng mobile, phía server sẽ cung cấp những API sau:

- authentication : dùng trong việc xác thực, nhận username và password để kiểm tra và trả về token cho phiên đăng nhập đó nếu xác thực đúng.
- get_device_status : nhận khóa phòng và trả về trạng thái tất cả thiết bị trong phòng đó.
- get_room_image : nhận khóa phòng về trả về hình ảnh mới nhất của phòng dưới dạng byte string.
- logout : xóa token của phiên đăng nhập.
- get_data : nhận loại cảm biến, số phòng, số lượng dữ liệu cần lấy và trả về dữ liệu của cảm biến đó.
- get_rooms : lấy tất cả thông tin về phòng trong nhà tương ứng.
- send_request_to_device : gửi yêu cầu tắt/mở thiết bị.

Tất cả những API trên đều được xác thực dựa trên token nhằm đảm bảo yêu cầu về bảo mật cho hệ thống nhà của người dùng, tránh bị đánh cắp thông tin và dữ liệu.

3.2.4 Mô hình AI

3.2.4.a Tổng quan về mô hình neural network

Ở con người, các nơ-ron thần kinh được liên kết với nhau thông qua synapse, khi một nơ-ron muốn gửi thông tin cho nơ-ron khác, nó sẽ sinh ra điện thế hoạt động và lan truyền qua synapse, giải phóng chất dẫn truyền thần kinh tạo ra phản ứng ở nơ-ron kia [1]. Dựa vào đặc điểm này của não bộ con người, mô hình mạng nơ-ron thần kinh (neural network) đã được phát triển. Một neural network sẽ có 3 lớp gồm: input, hidden và output. Input layer nhận dữ liệu đầu vào như ảnh, video,... giống như não nhận thông tin từ mắt, tai,... Hidden layer sẽ phân tích dữ liệu ấy và Output layer sẽ cho ra kết quả, tương tự việc não phân tích và đưa ra đánh giá về thông tin mà tai, mắt,... nhận được. Các đơn vị tính toán ở mỗi layer là một node. Mỗi node này tương đương một nơ-ron trong não bộ. Chúng cũng được liên kết với nhau và có thể được kích hoạt bởi các hàm kích hoạt (activation function), tương tự việc synapse giải phóng chất dẫn truyền để gây ra phản ứng ở nơ-ron. Được phát triển dựa trên cách nơ-ron hoạt động nhưng neural network cũng không hoàn toàn giống với não bộ con người. Một đặc điểm khác là số lượng node không có

định, mỗi loại neural network sẽ sử dụng số lớp hidden khác nhau, từ đó có số node khác nhau, dẫn tới sự đa dạng trong quá trình phát triển mô hình.

3.2.4.b Tổng quan về mô hình nơ ron tích chập (CNN)

CNN là một mạng nơ-ron áp dụng phép tính tích chập (convolution) lên các pixel của frame để giảm kích thước dữ liệu, lấy ra các đặc điểm quan trọng của các pixel để phân tích.[2]

Về phép tính tích chập: Kernel (hay filter) là một ma trận vuông kích thước $k \times k$, trong đó k là số lẻ, k có thể bằng 1, 3, 5, 7, 9,...

Ví dụ kernel kích thước 3×3 (Hình 4):

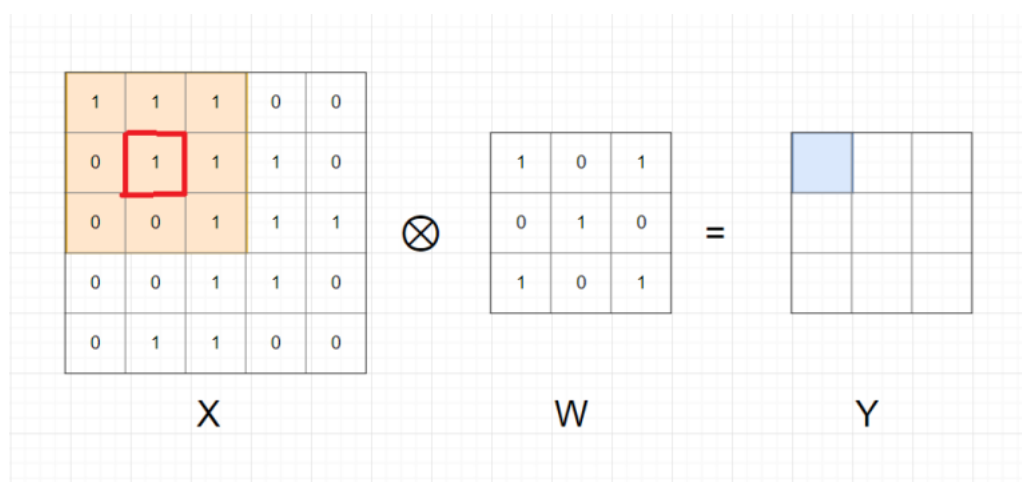
$$W = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

Hình 4: kernel size 3×3

Kí hiệu phép tính convolution: \otimes

$Y = X \otimes W$ nghĩa là:

Với mỗi phần tử x_{ij} trong ma trận X lấy ra một ma trận có kích thước bằng kích thước của kernel W có phần tử x_{ij} làm trung tâm (đây là vì sao kích thước của kernel thường lẻ) gọi là ma trận A . Sau đó tính tổng các phần tử của phép tính element-wise của ma trận A và ma trận W , rồi viết vào ma trận kết quả Y . Tiếp theo lấy một phần tử khác cách phần tử ban đầu một đoạn khoảng cách bằng stride (sẽ trình bày ở mục sau) làm phần tử trung tâm và tính như trên. Tiếp tục đến cuối ma trận đầu vào sẽ cho ra ma trận Y hoàn chỉnh.



Hình 5: Phép tính convolution

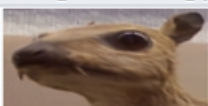
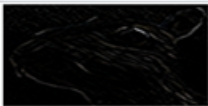




Ở hình 5, giá trị ở ô đầu tiên trong ma trận Y sẽ là $1*1 + 0*1 + 1*1 + 0*0 + 1*1 + 1*0 + 0*1 + 0*0 + 1*1 = 4$

→ Kích thước ma trận Y sẽ nhỏ hơn ma trận X ban đầu một lượng bằng stride ở cả 2 chiều, giảm lượng dữ liệu nhưng vẫn giữ được các thông tin quan trọng.

Ý nghĩa của phép tính convolution:

- Làm mờ
- Làm nét
- Xác định cạnh

Với mỗi kernel khác nhau thì sẽ phép tính convolution sẽ có ý nghĩa khác nhau (Hình 6).

Operation	Kernel w	Image result $g(x,y)$
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

Hình 6: Tác dụng một số loại kernel

3.2.4.c Giới thiệu về YOLOv5

YOLOv5 là một mô hình phát hiện đối tượng dựa trên deep learning, được phát triển bởi Alexey Bochkovskiy và đội ngũ Ultralytics. Mô hình này xây dựng trên ý tưởng của YOLO (You Only Look Once), một phương pháp phát hiện đối tượng nhanh chóng và chính xác.

3.2.4.d Cấu trúc mô hình

YOLOv5 sử dụng kiến trúc mạng nơ-ron tích chập (convolutional neural network - CNN) để xây dựng mô hình. Nó bao gồm một backbone network để trích xuất đặc trưng và một head network để dự đoán các hộp giới hạn (bounding boxes) và các lớp đối tượng.

3.2.4.e Đặc điểm chính

- Tính năng chính của YOLOv5 là khả năng phát hiện và phân loại đa đối tượng trong thời gian thực.
- Nó cho phép phát hiện người thông qua camera một cách hiệu quả và chính xác.
- Mô hình YOLOv5 được huấn luyện trên nhiều tập dữ liệu lớn để học các đặc trưng của các lớp đối tượng khác nhau, bao gồm cả người.
- Nó cung cấp khả năng phát hiện và theo dõi người trong các tình huống thời gian thực, như giám sát an ninh, phân loại hành vi và tương tác người-máy.

3.2.4.f Ưu điểm

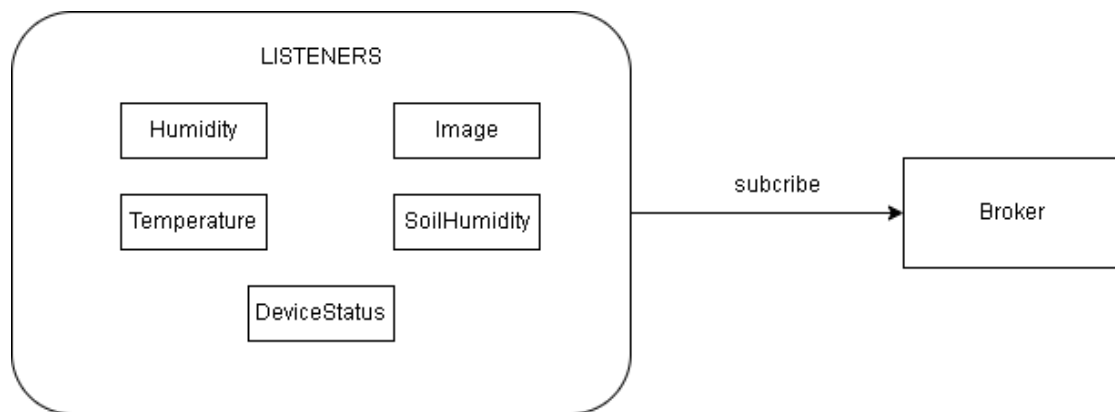
- YOLOv5 có tốc độ xử lý nhanh, cho phép phát hiện đối tượng và phân loại trong thời gian thực.
- Nó có khả năng xử lý ảnh và video trực tiếp từ camera.
- Mô hình này được cung cấp dưới dạng mã nguồn mở và dễ dàng tích hợp vào các ứng dụng và dự án khác.

3.2.4.g Vai trò và hoạt động trong hệ thống

Hệ thống AI trong dự án này được xây dựng dựa trên pretrained model của deepakcrk [4] Mô hình đóng vai trò phân tích hình ảnh để xác định xem có người ở hay không và quyết định gửi tín hiệu phù hợp cho thiết bị. Sau khi ảnh từ camera đã được lưu lại ở server (mục 3.1.2.b), mô hình sẽ tiến hành xác định vùng bounding box của người trong ảnh, nếu có thì sẽ gửi tín hiệu bật đèn nếu đèn đang tắt, nếu không có người xuất hiện trong vòng 5 giây thì sẽ gửi tín hiệu tắt đèn nếu đèn đang bật. Cuối cùng mô hình sẽ lưu lại hình ảnh đã xử lý để hiển thị cho người dùng khi cần thiết.

3.2.5 Các listener liên kết thành phần

Các listener trên server bao gồm:



Hình 7: Các listener của hệ thống

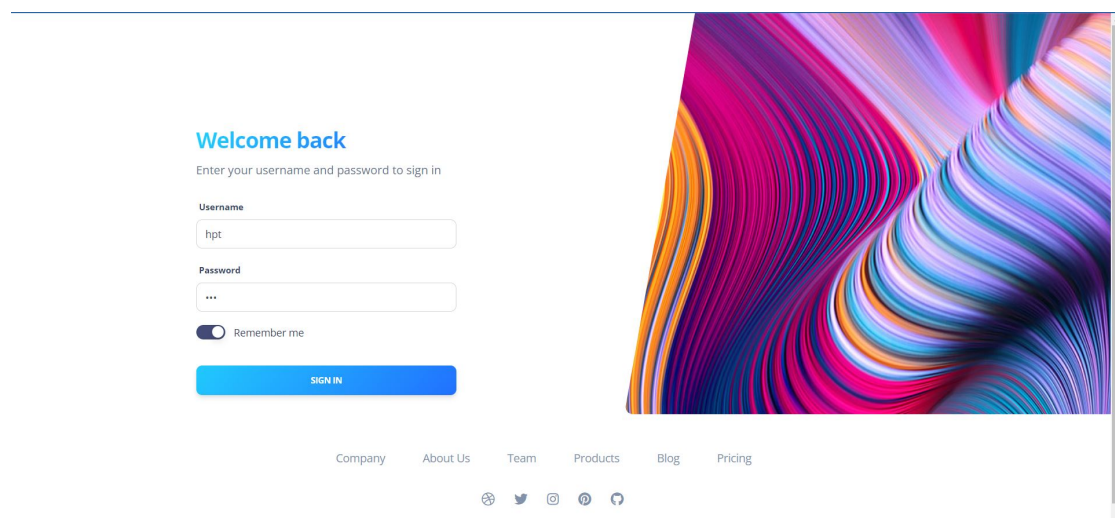
Trong đó:

- Humidity ,Tempature : nhận dữ liệu được gửi từ cảm biến nhiệt độ và độ ẩm và cập nhật vào database.
- SoilHumidity : nhận dữ liệu từ cảm biến độ ẩm đất.
- Image : như đã mô tả ở mục 3.1.2.b, hình ảnh từ camera được gửi về thông qua giao thức MQTT, do vậy cần listener này để nhận hình ảnh.
- DeviceStatus : cập nhật trạng thái các thiết bị liên tục để đồng bộ giữa các ứng dụng.

3.3 UI

3.3.1 Web UI

3.3.1.a Trang login



Hình 8: Giao diện trang login

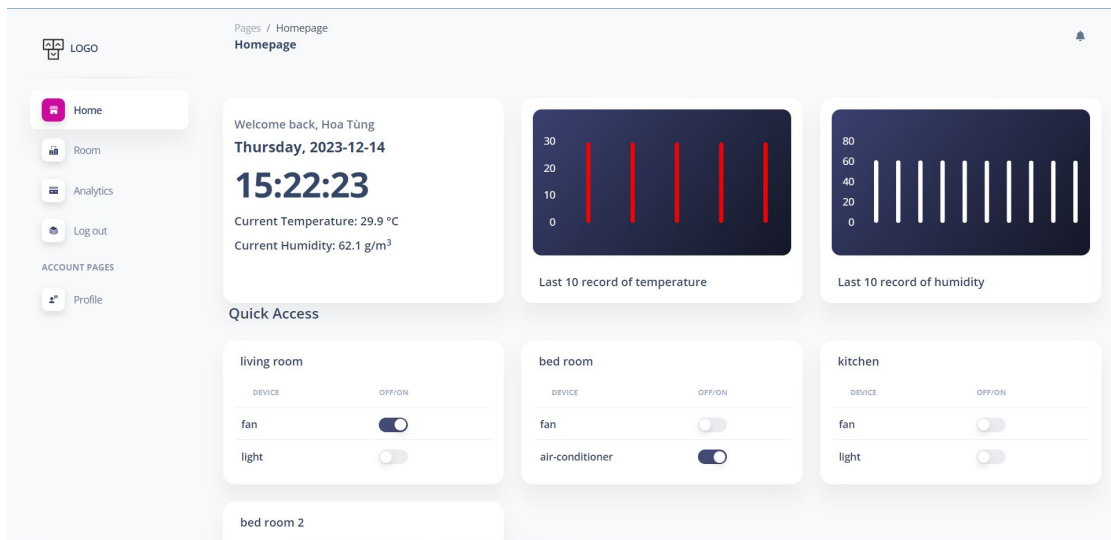
Khi user vừa mới truy cập vào trang web, user sẽ bắt gặp trang login, ở trang này user phải điền thông tin các thông tin như username và password để có thể được login vào server.

3.3.1.b Trang Home

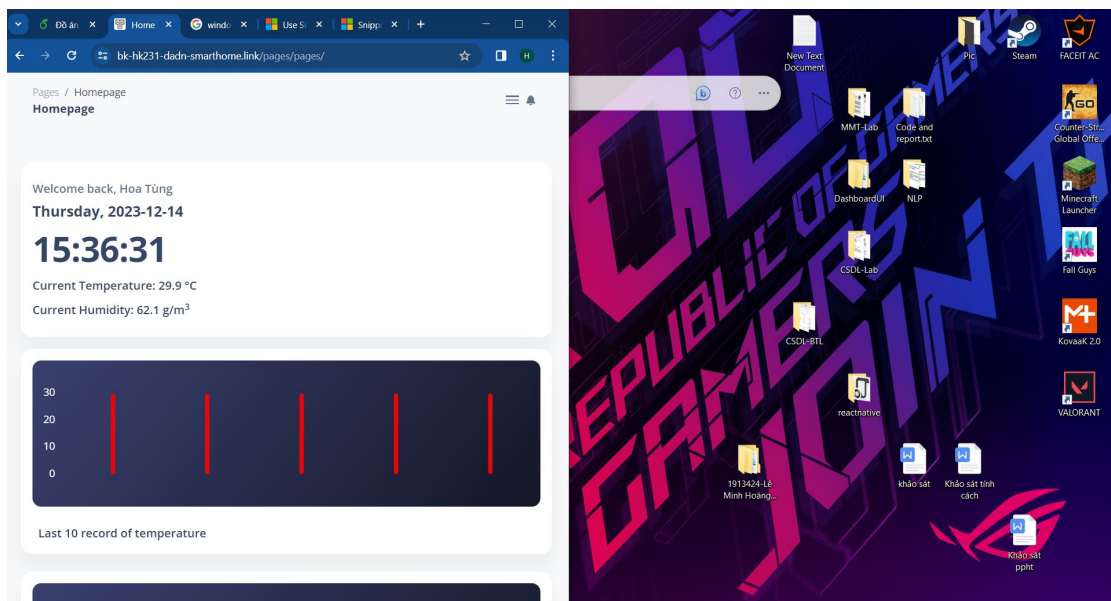
Login thành công, user sẽ được chuyển đến trang Homepage, tại trang này, các thông tin chính, tóm tắt sẽ được gửi đến user. Ngay tại hàng đầu tiên, user sẽ thấy các thông tin như giờ, ngày tháng, độ ẩm và nhiệt độ của 10 phút gần đây nhất.

Phần tiếp theo của trang Home là phần Quick Access, tại đây user sẽ được cung cấp các bảng điều khiển của các phòng thông qua đó user có thể điều khiển từng thiết bị của toàn bộ ngôi nhà chỉ với một cú click chuột. Phần Quick Access này sẽ giúp user tiết kiệm thời gian, thao tác dễ dàng và gia tăng trải nghiệm người dùng.

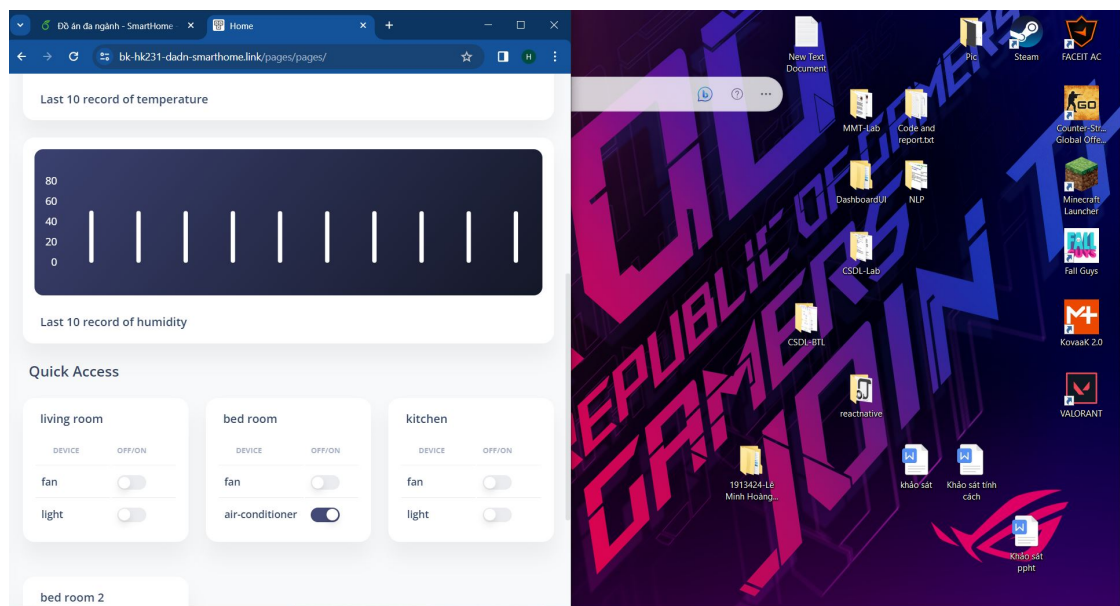
Ngoài ra, trang Home cũng được thiết kế responsive phù hợp cho mọi loại thiết bị



Hình 9: Giao diện Homepage



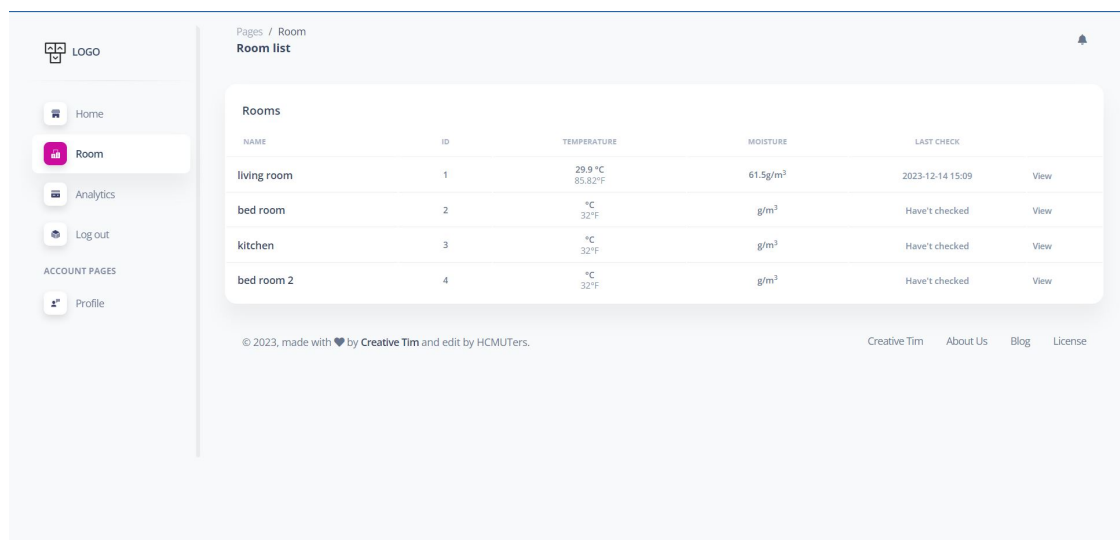
Hình 10: Giao diện Homepage responsive (1)



Hình 11: Giao diện Homepage responsive (2)

3.3.1.c Trang Room

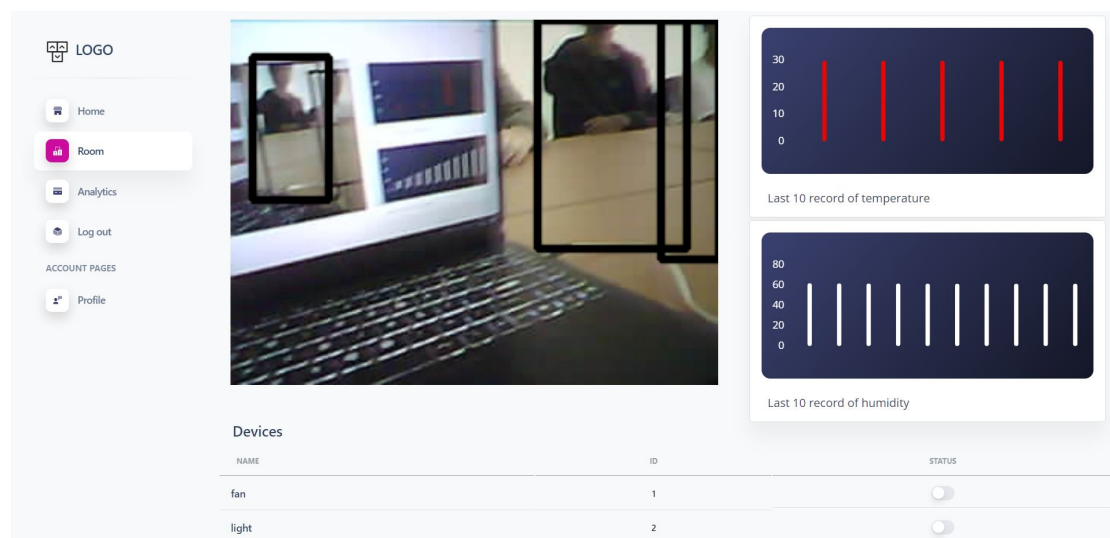
Khác với trang Home, chỉ chứa các thông tin chính của phòng Living room, thì ở trang Room các thông tin chính như ID phòng, nhiệt độ, độ ẩm ở tất cả các phòng đều được hiển thị cho người dùng.



Hình 12: Giao diện trang Room

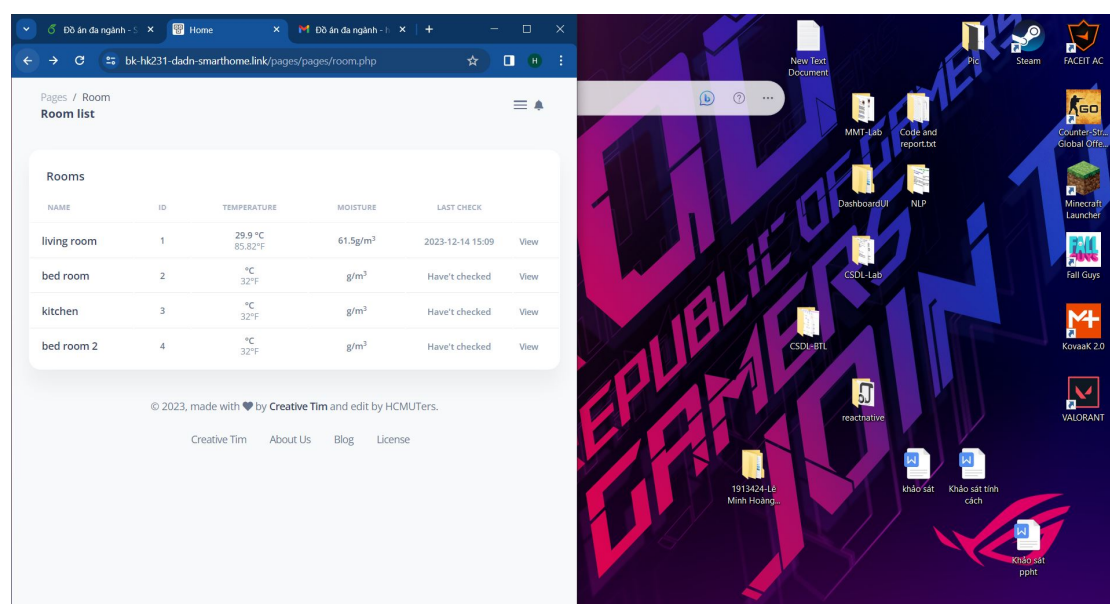
Khi user muốn xem chi tiết thông tin của từng phòng, user sẽ click vào nút View ở cột ngoài cùng tương ứng với phòng đó, hệ thống sẽ chuyển user đến trang có thông tin chi tiết của từng

phòng.

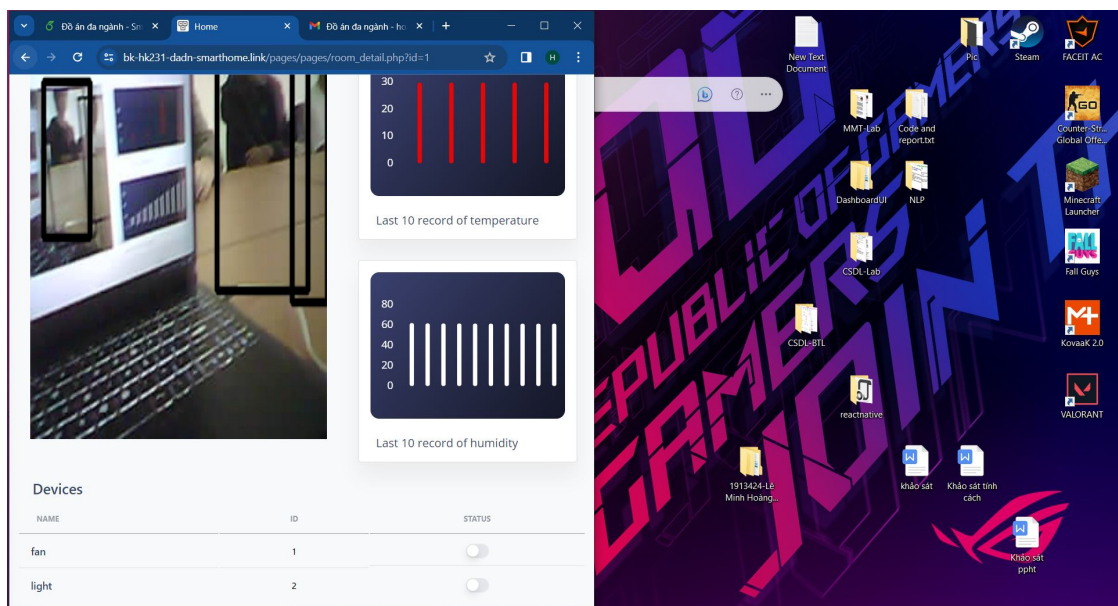


Hình 13: Giao diện trang Living room

Tại trang chi tiết của từng phòng, user có thể xem các thông tin về chi tiết về nhiệt độ cũng như độ ẩm đồng thời cũng có thể điều khiển được các thiết bị có trong phòng đó. Đồng thời sẽ hiển thị hình ảnh mới nhất của phòng (đã qua xử lý bằng mô hình AI).



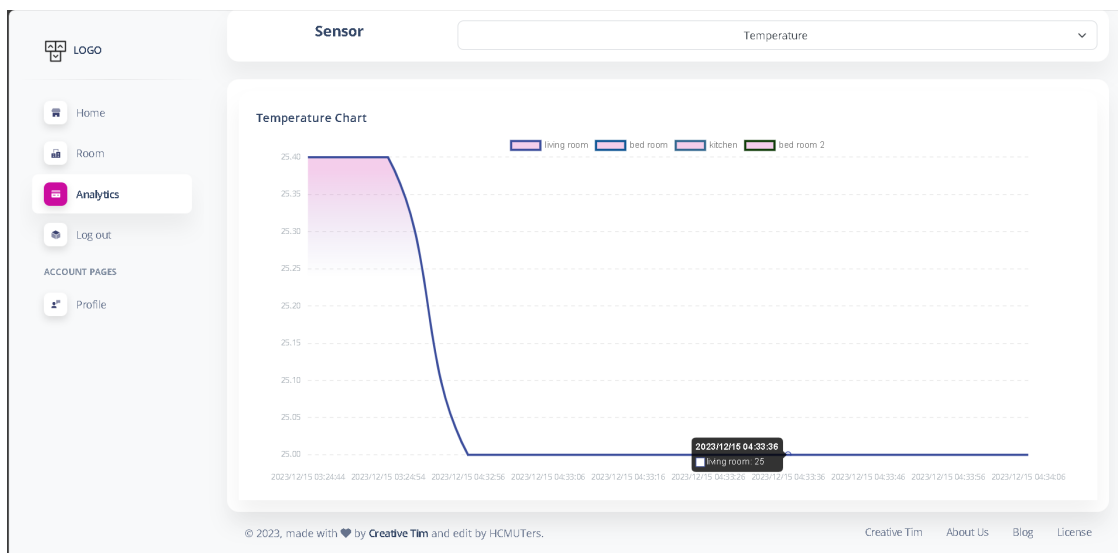
Hình 14: Giao diện trang Room responsive



Hình 15: Giao diện trang Living room responsive

3.3.1.d Trang Analytics

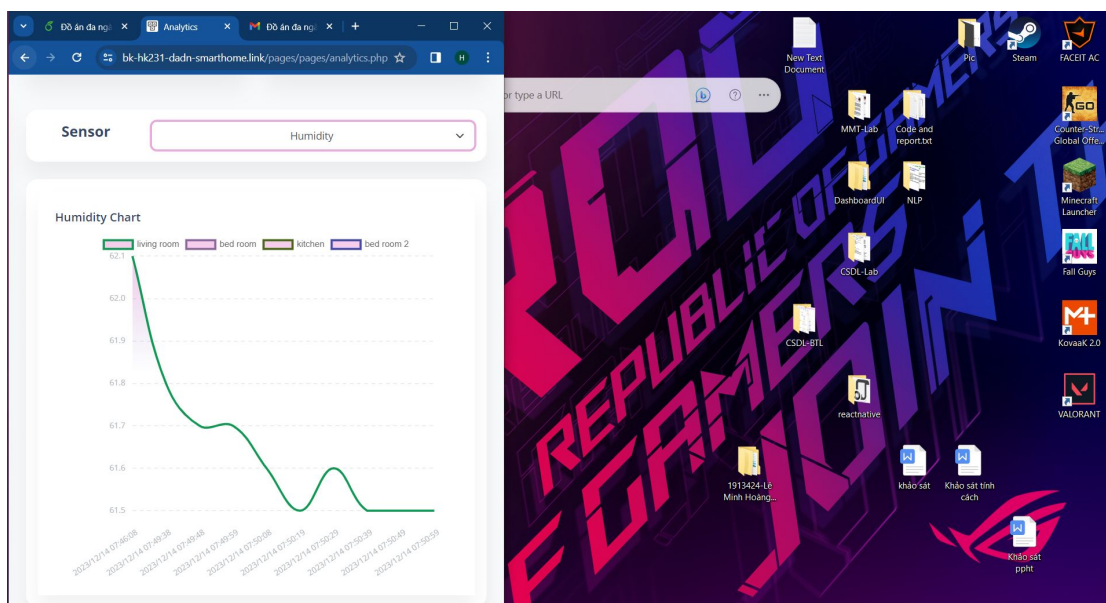
Cuối cùng là trang Analytics, trang này sẽ vẽ biểu đồ nhiệt độ/độ ẩm của toàn bộ các phòng để user có cái nhìn tổng quan về tình hình, ngôi nhà của mình. User có thể tùy chỉnh loại biểu đồ mình muốn xem thông qua dropdown Sensor



Hình 16: Giao diện trang Analytics chế độ xem nhiệt độ



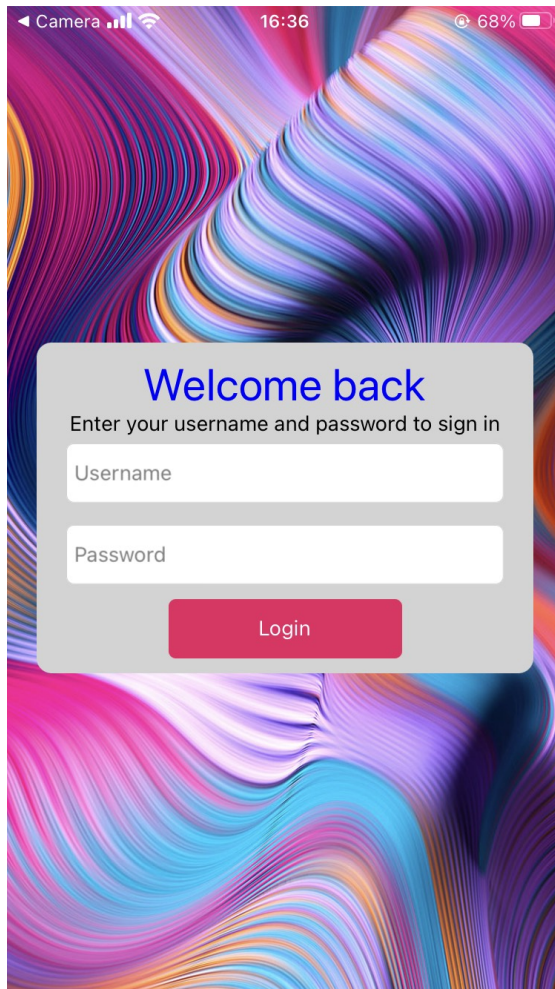
Hình 17: Giao diện trang Analytics chế độ xem độ ẩm



Hình 18: Giao diện trang Analytics responsive

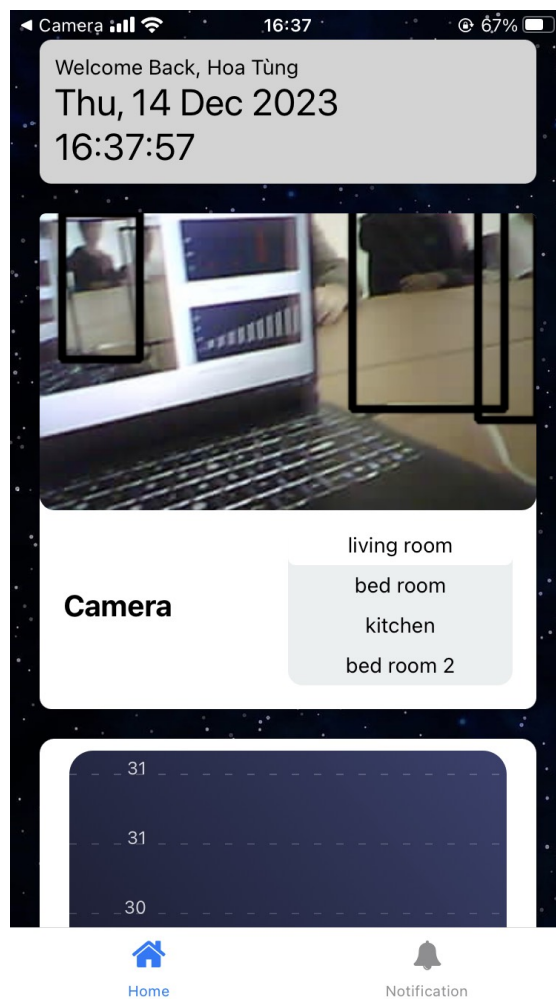
3.3.2 Mobile UI

Về phía Mobile app, cũng như phần web, khi user lần đầu tiên vào app thì user sẽ được chuyển đến trang đăng nhập.



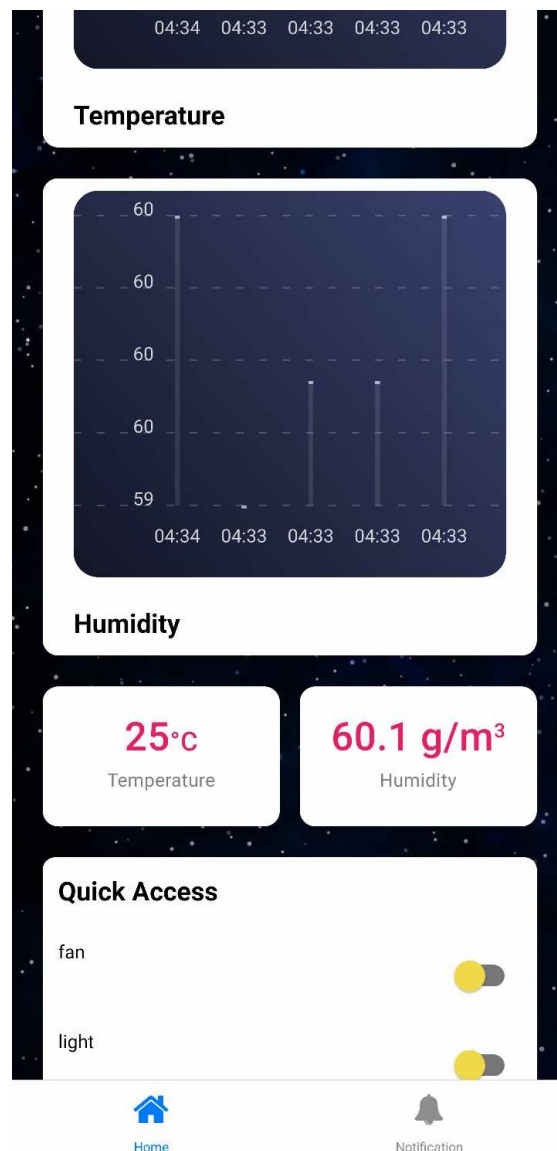
Hình 19: Giao diện Mobile trang Login

Sau khi đăng nhập, user sẽ được chuyển đến trang homepage và đây cũng sẽ là trang duy nhất của app này. Tuy chỉ có một trang duy nhất, nhưng trang này có đầy đủ các thông tin về nhiệt độ/độ ẩm của các phòng, đồng thời cũng đầy đủ các bảng điều khiển cho các phòng.



Hình 20: Giao diện trang Home Living room (1)

Có thể thấy ở hình 20, user có thể xem được giờ/ngày tháng và camera ở phòng đang được chọn, user có thể tùy chỉnh chọn phòng bằng các tap vào các lựa chọn phòng ở select menu bên cạnh chữ camera. Đồng thời việc chọn lựa các phòng ở mục camera cũng sẽ xem như chọn xem thông tin và điều khiển các thiết bị ở phòng đó. Nên khi chọn phòng ở mục Camera, các thông tin về nhiệt độ/độ ẩm và bảng điều khiển cũng sẽ thay đổi để phù hợp với phòng được chọn. Về thông tin nhiệt độ/độ ẩm và bảng điều khiển hiển thị như trong hình 21.



Hình 21: Giao diện trang Home Living room (2)

4 Kết luận

Kết quả của đồ án là thành quả tìm hiểu và làm việc của năm người trong nhóm. Quá trình này giúp nhóm hiểu nhiều hơn về cách một hệ thống IoT được xây dựng và hoạt động, cùng với vai trò của trí tuệ nhân tạo ngày nay trong lĩnh vực này. Tuy vẫn còn một vài hạn chế về tốc độ, độ trễ khi cập nhật trạng thái nhưng hệ thống cũng đã đáp ứng được nhu cầu giao tiếp giữa người dùng và thiết bị từ xa, giúp người dùng có thể cập nhật trạng thái về ngôi nhà của mình bất kỳ lúc nào. Từ đó hoàn thành được mục tiêu đặt ra lúc đầu.

5 Mã nguồn và demo

Mã nguồn và các tài nguyên liên quan tới dự án này có thể được tìm thấy trong Github của nhóm: [ntpt7921/MultidisciplinaryProj](https://github.com/ntpt7921/MultidisciplinaryProj)

Tài liệu

- [1] Sự dẫn truyền thần kinh, Kenneth Maiese, 2022 Oct.
- [2] Convolutional neural network, Nttuan8, 2019 Mar.
- [3] AWS Documentation, <https://docs.aws.amazon.com/>
- [4] YOLOv5-crowdhuman, deepakcrk, <https://github.com/deepakcrk/yolov5-crowdhuman>
- [5] Getting started programming ESP32-CAM with ESP32-CAM-MB micro USB programmer serial converter loader, Yellow Purple, <https://www.youtube.com/watch?v=z67mFL63e2M>
- [6] ESP32-cam-MQTT, luisomoreau, <https://github.com/luisomoreau/ESP32-cam-MQTT>