

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



Thực tập đồ án đa ngành -
hướng hệ thống thông tin

Nhà thông minh tích hợp
nhận diện khuôn mặt

GVHD: Nguyễn Thị Ái Thảo
SV: Lữ Hoàng Anh - 2010113
Nguyễn Trọng Tín - 2012215
Hồ Trương Đức Tiến - 2012196

TP. HỒ CHÍ MINH, THÁNG 05/2023

Mục lục

1 Member list & Workload	2
2 Giới thiệu đề tài	3
3 Requirement Analysis	4
3.1 Yêu cầu chức năng	4
3.2 Yêu cầu phi chức năng	4
4 Use case	5
4.1 Use-case toàn hệ thống	5
4.2 Đặc tả Use-case	6
5 Class Diagram	8
6 Architecture	9
7 Công nghệ sử dụng	10
7.1 Ngôn ngữ Python	10
7.2 MongoDB	10
7.3 MQTT	11
7.4 Adafruit	12
7.5 Microbit	13
7.6 Deep face	13
7.7 Zapier	14
7.8 Gradio	14
8 Giao diện hiện thực	16
8.1 Giao diện mô phỏng nhận diện khuôn mặt	16
8.1.1 Giao diện chính	16
8.1.2 Nhận diện thành công	17
8.1.3 Nhận diện thất bại	20
8.2 Giao diện đăng ký	22
8.2.1 Đăng nhập	22
8.2.2 Giao diện chính	22
8.2.3 Đăng ký	23
8.3 Dashboard	25
8.3.1 Log Status Dashboard	25
8.3.2 Hiển thị trực quan dạng biểu đồ cột	25
8.3.3 Hiển thị trực quan dạng biểu đồ đường	26
8.3.4 Dashboard điều khiển các thiết bị trên Adafruit	26
9 Resources	27
9.1 Instruction for use	27
9.2 Link demo	27
9.3 Link github	27

1 Member list & Workload

No.	Fullname	Student ID	Workload	Percentage of work
1	Nguyễn Trọng Tín	2012215	- Implementing Server - Implementing Gateway - Implementing Database	100%
2	Lữ Hoàng Anh	2010113	- Implementing Dashboard - Database and Data Processing - Writing Report	100%
3	Hồ Trương Đức Tiến	2012196	- AI Research - Writing report	100%

2 Giới thiệu đề tài

Smart Home ngày càng trở nên phổ biến trong những năm gần đây, với nhiều chủ nhà đang tìm cách tự động hóa ngôi nhà của họ và tăng sự tiện lợi, an ninh và hiệu quả năng lượng. Một trong những công nghệ quan trọng có thể giúp ngôi nhà thông minh trở nên thông minh hơn là nhận dạng khuôn mặt.

Công nghệ nhận dạng khuôn mặt ngày càng trở nên phổ biến do tiềm năng cải thiện tính bảo mật và sự tiện lợi trong các ứng dụng khác nhau. Dự án này nhằm mục đích phát triển một ứng dụng nhận dạng khuôn mặt có thể được tích hợp vào hệ thống tự động hóa nhà thông minh để tăng cường an ninh cho ngôi nhà và mang lại trải nghiệm cá nhân hóa cho người dùng.

Hệ thống sẽ cho phép chủ nhà đăng ký danh sách thành viên trong gia đình cho hệ thống, mỗi khi có người trong gia đình về thì cửa sẽ mở, loa sẽ báo tên người vừa vào nhà. Hệ thống sẽ lưu thông tin về thời gian người đó về nhà, đồng thời hiển thị thông tin về người hiện đang ở nhà cho chủ nhà. Khi có người ra ngoài, hệ thống sẽ nhận diện và mở cổng, khóa cổng sau khi người đó ra ngoài, đồng thời thông báo cho chủ nhà biết là ai đã ra ngoài.

Ngoài ra, hệ thống sẽ có thể xử lý các trường hợp ngoại lệ như khách. Khi có xác nhận của chủ nhà, khách có thể mở cửa và hệ thống sẽ chụp ảnh để báo cáo cho chủ nhà.

Ứng dụng nhận dạng khuôn mặt có rất nhiều ứng dụng tiềm năng, từ tăng cường an ninh gia đình đến cung cấp trải nghiệm người dùng được cá nhân hóa. Nó cũng có thể được sử dụng trong các cài đặt khác như an ninh nơi làm việc, kiểm soát truy cập và nghiên cứu tiếp thị.

Nhìn chung, dự án ứng dụng nhận dạng khuôn mặt có tiềm năng cải thiện tính bảo mật, tiện lợi và cá nhân hóa trong các ứng dụng khác nhau và có thể là một tài sản quan trọng trong việc phát triển hệ thống tự động hóa nhà thông minh.

3 Requirement Analysis

3.1 Yêu cầu chức năng

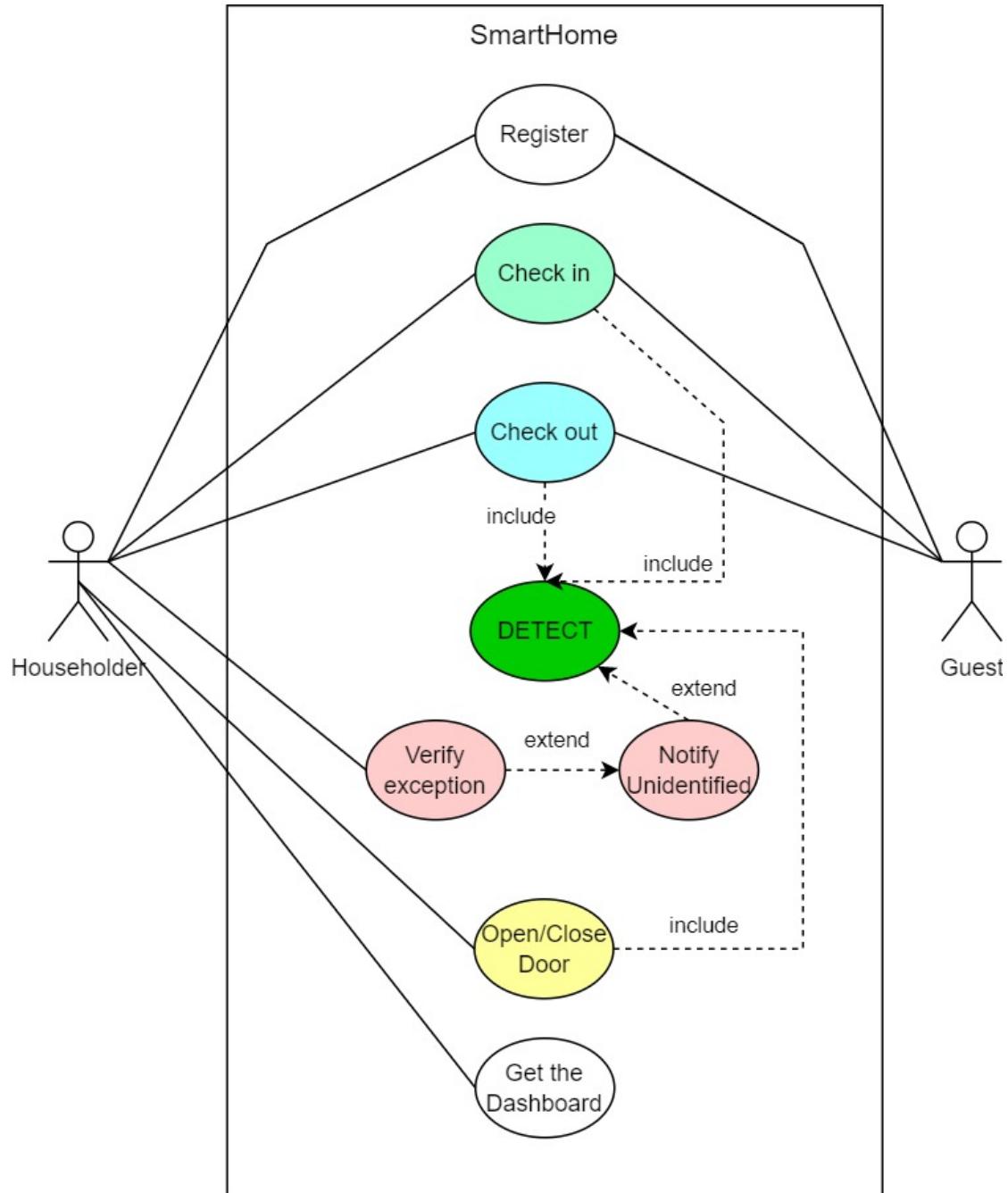
1. Hệ thống cho phép chủ nhà đăng ký các thành viên trong gia đình.
2. Hệ thống nhận diện được gương mặt của các thành viên đã đăng ký.
3. Hệ thống thông báo tên của người vừa vào nhà.
4. Hệ thống lưu lại thông tin của người ra/vào nhà.
5. Hệ thống có thể cấp quyền truy cập của khách với sự xác nhận của chủ nhà và chụp ảnh khách cho mục đích bảo mật.
6. Hệ thống có khả năng điều khiển hệ thống khóa cửa để đóng/mở cửa tự động.
7. Hệ thống có giao diện để hiển thị thông tin về những người đang ở trong nhà vào bất kỳ thời điểm nào.

3.2 Yêu cầu phi chức năng

1. Thuật toán nhận diện khuôn mặt phải chính xác và đáng tin cậy, phản hồi nhanh.
2. Camera có chất lượng tốt, góc nhìn rộng và chụp thiếu sáng tốt.
3. Hệ thống phải thông báo nhanh, với độ trễ tối đa 3s.
4. Hệ thống phải bảo mật được các thông tin người dùng.
5. Giao diện trực quan, dễ sử dụng, hiển thị rõ thông tin và ngắn gọn.
6. Hệ thống có khả năng mở rộng để xử lý các thành viên mà chủ nhà bổ sung trong tương lai.

4 Use case

4.1 Use-case toàn hệ thống



Hình 1: Use Case of whole system

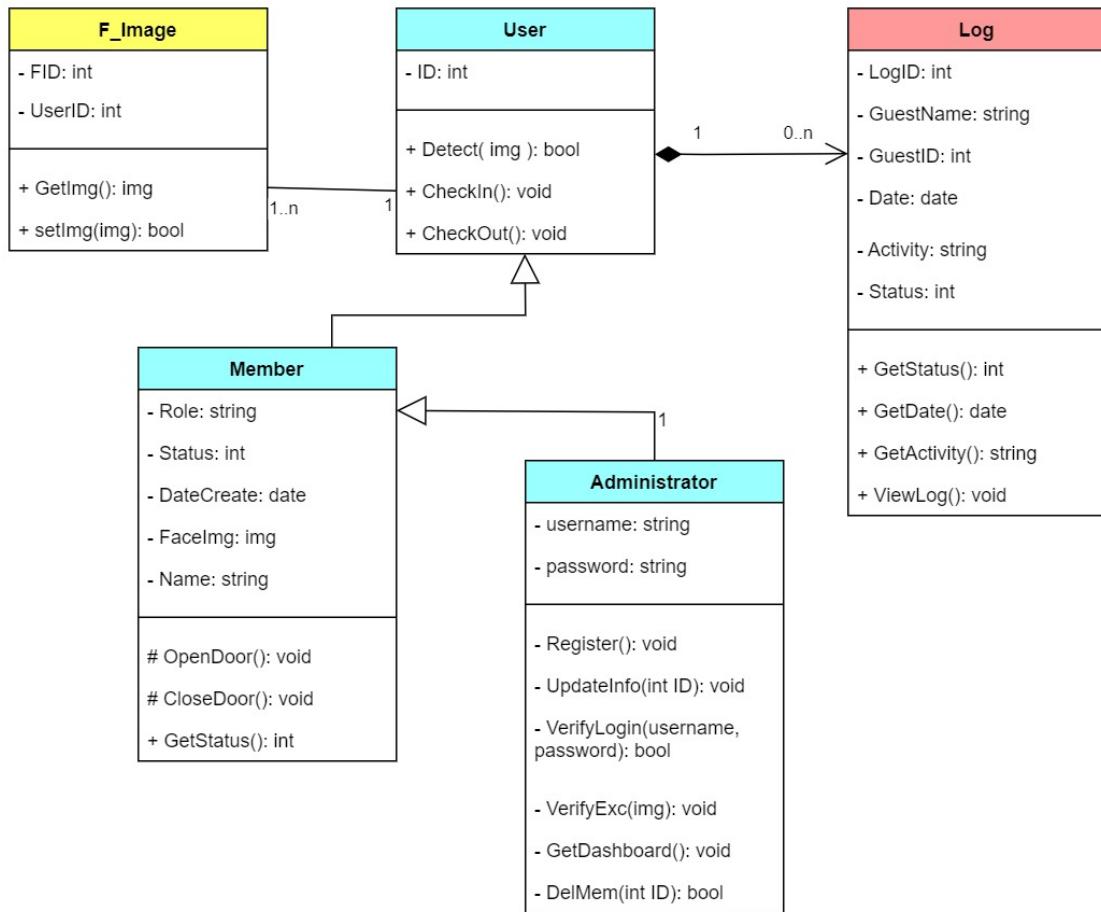
4.2 Đắc tả Use-case

Register	
Actors	Mọi người
Description	Chủ nhà đăng ký khuôn mặt cho khách Chủ nhà đăng ký dưới quyền Admin và đăng ký thêm thông tin khách mới
Precondition	Chủ nhà đang đăng nhập dưới quyền admin.
Trigger	1. Dùng giao diện đăng ký của khách 2. Chủ nhà dùng giao diện đăng ký của admin.
Normal Flow	1. Chủ nhà nhấn nút "Đăng ký". 2. Camera chụp lại hình ảnh của khách 3. Diền đầy đủ thông tin và gửi 4. Hệ thống gửi dữ liệu khuôn mặt đến Back-end và xử lý 5. Tại back-end xác minh và lưu toàn bộ thông tin vào database 6. Xác minh thành công và kết thúc hành động
Exception	Không
Alternative	Tại bước 2, chủ nhà gửi ảnh của khách đã chụp trước thay vì chụp trực tiếp 1. Chủ nhà chọn nút "Upload" 2. Chủ nhà chọn ảnh của khách 3. Chủ nhà bấm xác nhận. 4. Dữ liệu gửi xuống back-end để xử lý và lưu vào database. 5. Kết thúc hành động.

Check-in	
Actors	Mọi người.
Description	Tất cả mọi người phải nhận dạng khuôn mặt (check-in) trước khi muốn được vào nhà.
Precondition	1. Hệ thống đang làm việc, không báo trì.
Trigger	Dưa mặt vào gần camera để quét xác minh
Normal Flow	1. Án nút "Submit" 2. Camera quét khuôn mặt 3. Hệ thống gửi dữ liệu khuôn mặt đến Back-end 4. Tại back-end thực hiện xác minh khuôn mặt 5. Xác minh thành công. 6. Cửa mở. 7. Đèn phòng khách sẽ mở 8. Gửi thông báo đến Chủ nhà. 9. Sau 30 giây cửa sẽ tự động đóng.
Exception	Tại bước 4 nếu xác minh không thành công 1. Hệ thống báo cáo hình ảnh người lạ xâm nhập đến Chủ nhà 2. Chủ nhà không nhận ra người lạ, bật báo động, khóa cửa 3. Cửa sẽ tự động đóng.
Alternative	Tại bước 4 nếu xác minh không thành công 1. Hệ thống báo cáo hình ảnh người lạ xâm nhập đến Chủ nhà 2.1. Chủ nhà xác minh đó là người quen và mở cửa. 2.1.1 Chủ nhà có thể chuyển qua giao diện đăng ký thành viên cho người đó. 3. Cửa sẽ tự động đóng.

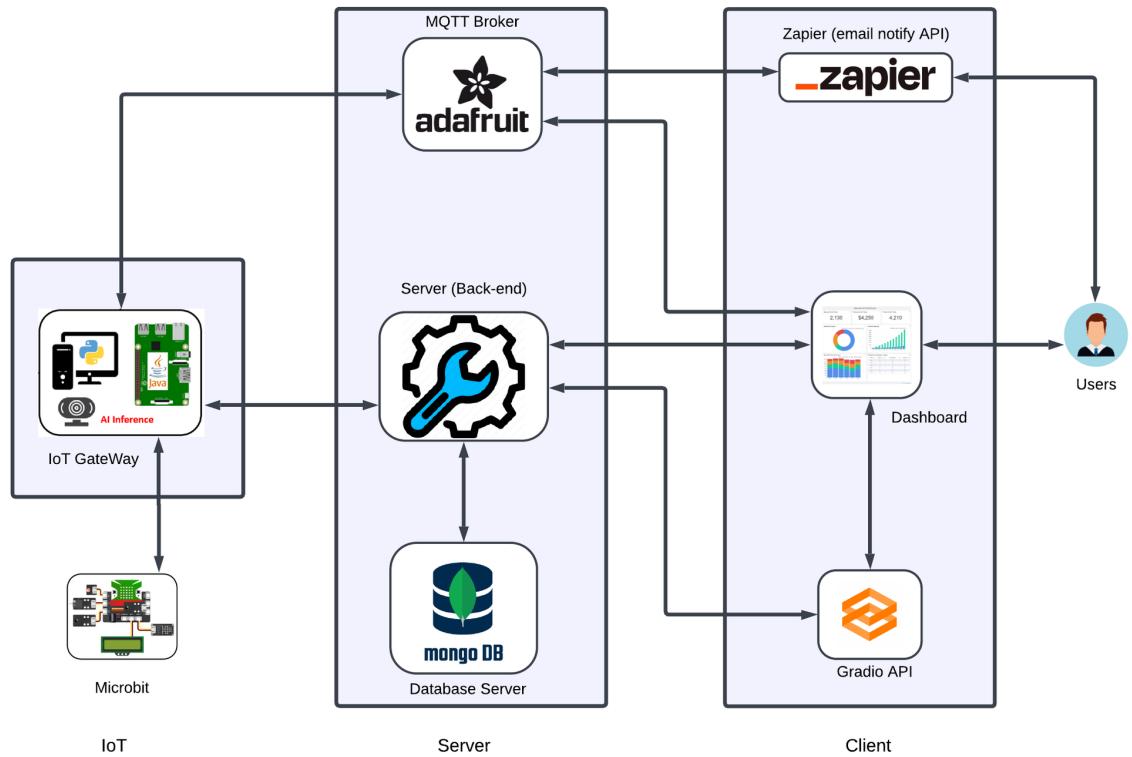
Check-out	
Actors	Guest or Administrator.
Description	Tất cả mọi người phải nhận dạng khuôn mặt (check-out) trước khi muốn được rời nhà.
Precondition	<ol style="list-style-type: none"> 1. Hệ thống đang làm việc, không bảo trì. 2. Người dùng đang ở trong nhà và quét khuôn mặt từ bên trong.
Trigger	Dưa mặt vào gần camera để quét xác minh
Normal Flow	<ol style="list-style-type: none"> 1. Camera quét khuôn mặt 2. Hệ thống gửi dữ liệu khuôn mặt đến Back-end 3. Tại back-end thực hiện xác minh khuôn mặt 4. Xác minh thành công. 5. Cửa mở. 6. Gửi thông báo đến Chủ nhà. 7. Sau 30 giây cửa sẽ tự động đóng.
Exception	None.
Alternative	<p>Tại bước 4 nếu xác minh không thành công</p> <ol style="list-style-type: none"> 1. Hệ thống báo cáo hình ảnh khách đến Chủ nhà 2. Chủ nhà xác minh đó là người quen và mở cửa. 3. Cửa sẽ tự động đóng.

5 Class Diagram



Hình 2: Class diagram of whole system

6 Architecture



Hình 3: Architecture of the system

Hệ thống được thiết kế theo mô hình Client - Server and Client - Broker, với flowchart được mô tả theo hình vẽ ở trên.

7 Công nghệ sử dụng

7.1 Ngôn ngữ Python



Python là một lựa chọn phổ biến cho các dự án IoT do tính đơn giản, dễ đọc, tính linh hoạt và hỗ trợ cộng đồng lớn.

Python có một bộ sưu tập lớn các thư viện và framework hữu ích cho các dự án IoT và AI. Các thư viện này giúp giao tiếp với phần cứng, xử lý dữ liệu và phát triển các ứng dụng web, trong số các tác vụ khác, trở nên dễ dàng hơn.

Python là ngôn ngữ đa nền tảng, có nghĩa là nó có thể chạy trên các hệ điều hành và nền tảng phần cứng khác nhau. Điều này giúp việc phát triển và triển khai các ứng dụng IoT trên các thiết bị và nền tảng khác nhau trở nên dễ dàng hơn.

7.2 MongoDB



1. Khái niệm

MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở, là CSDL thuộc NoSql và được hàng triệu người sử dụng. MongoDB là một database hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON thay vì dạng bảng như CSDL quan hệ nên truy vấn sẽ rất nhanh. MongoDB không dùng các bảng để lưu dữ liệu như các cơ sở dữ liệu quan hệ mà dùng khái niệm **collection**.

So với RDBMS thì trong MongoDB collection ứng với table, còn document sẽ ứng với row, MongoDB sẽ dùng các document thay cho row trong RDBMS. Các collection trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ không cần tuân theo một cấu trúc nhất định. Thông tin liên quan được lưu trữ cùng nhau để truy cập truy vấn nhanh thông qua ngôn ngữ truy vấn MongoDB.

2. Ưu điểm

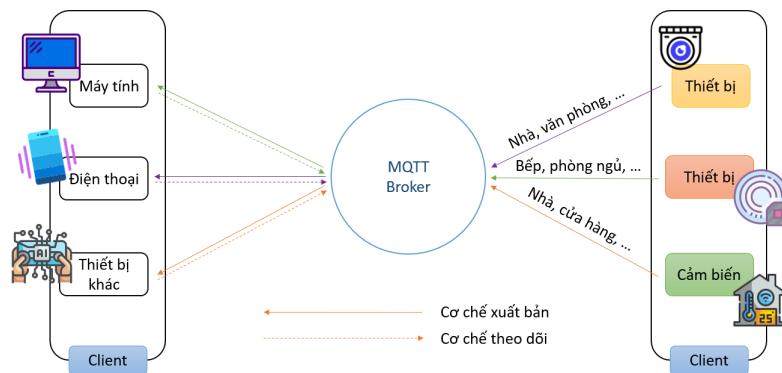
- Do MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ có các kích cỡ và các document khác nhau, linh hoạt trong việc lưu trữ dữ liệu.
- Dữ liệu trong MongoDB không có sự ràng buộc lẫn nhau, không có join như trong RDBMS nên khi insert, xóa hay update nó không cần phải mất thời gian kiểm tra xem có thỏa mãn các ràng buộc dữ liệu như trong RDBMS.
- MongoDB rất dễ mở rộng (Horizontal Scalability)
- Trường dữ liệu “`_id`” luôn được tự động đánh index (chỉ mục) để tốc độ truy vấn thông tin đạt hiệu suất cao nhất.
- Khi có một truy vấn dữ liệu, bản ghi được cached lên bộ nhớ Ram, để phục vụ lượt truy vấn sau diễn ra nhanh hơn mà không cần phải đọc từ ổ cứng.
- Hiệu năng cao: Tốc độ truy vấn (find, update, insert, delete) của MongoDB nhanh hơn hẳn so với các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS).

3. Lý do chọn

Lí do nhóm chọn MongoDB làm database để lưu trữ dữ liệu dựa trên các lý do sau:

- Dữ liệu mà hệ thống muốn lưu lại là các logs, các logs được lưu có thể có dạng khác nhau nên tính no-schema của NoSQL là phù hợp.
- MongoDB cung cấp MongoDB Atlas, cho phép lưu trữ miễn phí và dễ dàng kết nối với ứng dụng.
- Tốc độ truy vấn nhanh nên việc vẽ biểu đồ trên dashboard sẽ cho kết quả nhanh hơn.

7.3 MQTT



1. Định nghĩa

MQTT (Message Queuing Telemetry Transport) là giao thức truyền thông điệp (message) theo mô hình publish/subscribe (cung cấp / thuê bao), được sử dụng cho các thiết bị IoT

với băng thông thấp, độ tin cậy cao và khả năng được sử dụng trong mạng lưới không ổn định. Nó dựa trên một Broker (tạm dịch là “Máy chủ môi giới”) “nhẹ” (khá ít xử lý) và được thiết kế có tính mở (tức là không đặc trưng cho ứng dụng cụ thể nào), đơn giản và dễ cài đặt.

2. Tính năng

- Dạng truyền thông điệp theo mô hình Pub/Sub cung cấp việc truyền tin phân tán một chiều, tách biệt với phần ứng dụng.
- Việc truyền message là ngay lập tức, không quan tâm đến nội dung được truyền.
- Sử dụng TCP/IP là giao thức nền.
- Phần bao bọc dữ liệu truyền nhỏ và được giảm đến mức tối thiểu để giảm tải cho đường truyền

3. Ưu điểm

Với những tính năng, đặc điểm nổi bật trên, MQTT mang lại nhiều lợi ích nhất là trong hệ thống SCADA (Supervisory Control And Data Acquisition) khi truy cập dữ liệu IoT.

- Truyền thông tin hiệu quả
- Tăng khả năng mở rộng.
- Chi phí thấp.
- Rất an toàn, bảo mật.
- Tiết kiệm thời gian phát triển.

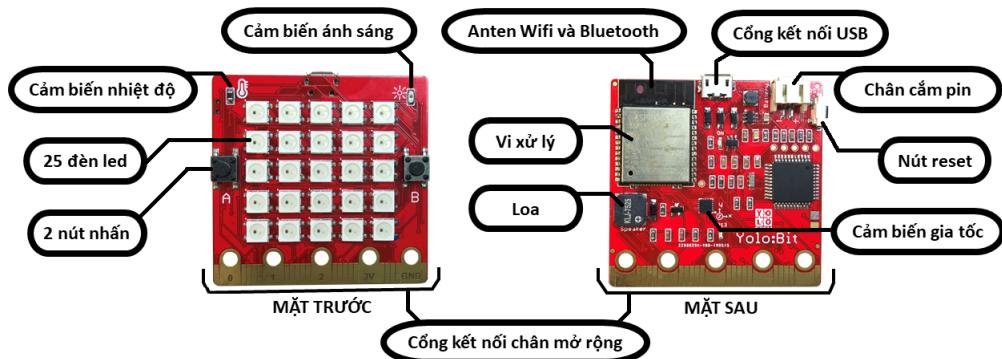
7.4 Adafruit



Để thiết bị có thể kết nối và chia sẻ dữ liệu trong thế giới kết nối vạn vật, chúng ta cần dùng đến điện toán đám mây – nơi sẽ lưu trữ dữ liệu của các thiết bị. Thông thường, chúng ta hay gọi đây là các máy chủ Server.

Có rất nhiều server được sử dụng trong Internet of things, hôm nay chúng tôi sẽ hướng dẫn các bạn làm quen với server Adafruit IO – một server khá phổ biến và được đánh giá là có giao diện khá thân thiện với người dùng.

7.5 Microbit

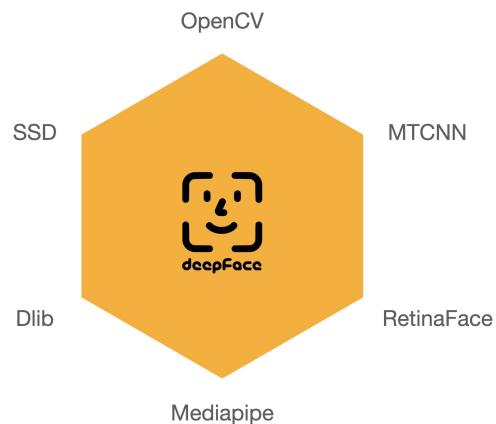


Micro:bit là một bo mạch vi điều khiển nhỏ, có thể lập trình được thiết kế để dạy và học lập trình máy tính, điện tử và điện toán vật lý. Nó được BBC ở Vương quốc Anh tạo ra vào năm 2015 như một phần của sáng kiến Make it Digital của họ và hiện được sử dụng bởi các nhà giáo dục và những người có sở thích trên khắp thế giới.

Micro:bit là một thiết bị bỏ túi có màn hình ma trận LED 5x5, hai nút có thể lập trình, gia tốc kế, la bàn và kết nối Bluetooth. Nó có thể được lập trình bằng nhiều ngôn ngữ lập trình, bao gồm Python, JavaScript và Microsoft MakeCode, đồng thời tương thích với nhiều tiện ích bổ sung và phụ kiện, chẳng hạn như cảm biến, động cơ và màn hình.

Dồ án này nhóm sẽ chọn loại mạch Yolo Bit để kết nối với các thiết bị và Gateway.

7.6 Deep face



DeepFace là một hệ thống nhận dạng khuôn mặt học sâu do một nhóm nghiên cứu tại Facebook tạo ra. Nó xác định khuôn mặt con người trong hình ảnh kỹ thuật số. Chương trình sử dụng một mạng thần kinh chín lớp với hơn 120 triệu trọng số kết nối và được đào tạo trên bốn triệu hình ảnh do người dùng Facebook tải lên.

DeepFace đạt độ chính xác $97,35\% \pm 0,25\%$ trên bộ dữ liệu Các khuôn mặt được dán nhãn trong tự nhiên (LFW) trong đó con người có 97,53%. [3] Điều này có nghĩa là DeepFace đôi khi thành công hơn con người.

7.7 Zapier



Zapier là một công cụ trực tuyến giúp người dùng giúp người dùng tự động hóa quy trình làm việc. Công cụ này cho phép dễ dàng kết nối Trainerize với những ứng dụng web như Gmail, Facebook, Dropbox, Slack,...

Trong đồ án này, nhóm sử dụng Zapier cho mục đích tự động hóa nhiệm vụ gửi email về cho chủ nhà mỗi khi nó được trigger.

7.8 Gradio



Gradio là một python package mã nguồn mở cho phép người dùng tạo các thành phần giao diện người dùng có thể tùy chỉnh. Các API và các hàm được viết bằng Python chỉ cần viết bằng một vài dòng code, rất dễ dàng sử dụng. Ngoài ra người dùng còn có thể thích hợp Gradio GUI trực tiếp vào Jupyter Notebook hoặc chia sẻ dưới dạng liên kết cho bất kỳ ai.

Trong đồ án này, nhóm sử dụng các API của Gradio để dễ dàng hiện thực và hiển thị các biểu đồ trực quan.

8 Giao diện hiện thực

Nhóm hiện thực một ứng dụng dành cho chủ nhà với 3 giao diện chính:

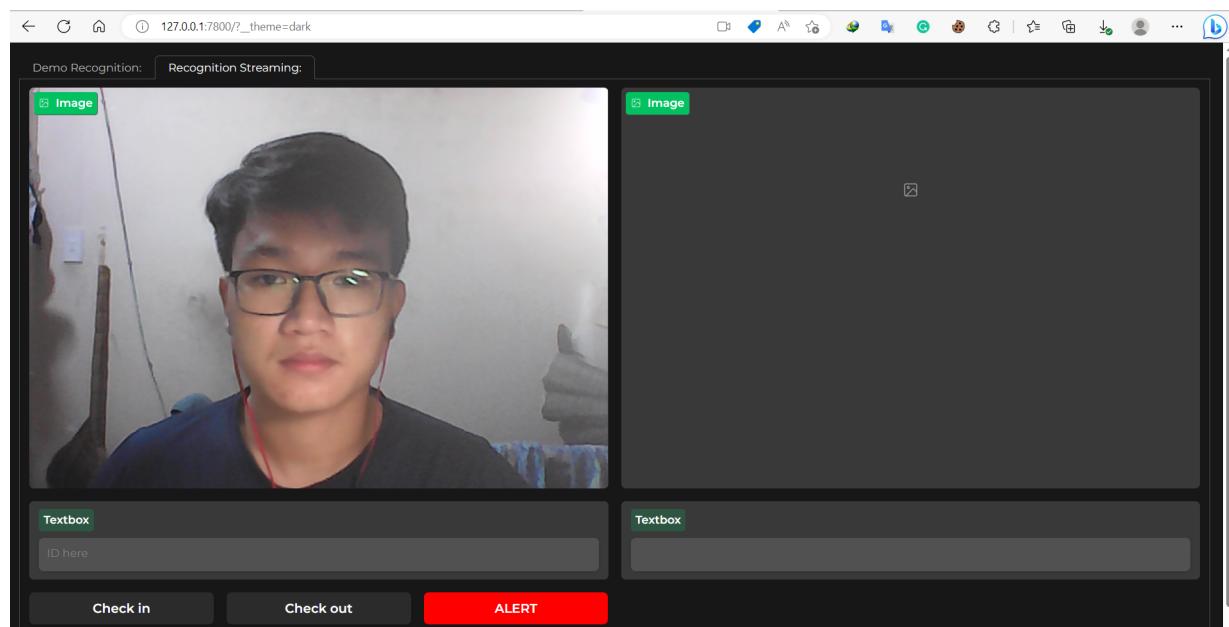
1. **Giao diện mô phỏng nhận diện khuôn mặt:** giao diện này mô phỏng việc nhận diện khuôn mặt của chủ nhà và những người được đăng ký để xác thực trước khi vào nhà hoặc ra khỏi nhà, cùng với đó là tự động mở cửa và bật đèn phòng khách.
2. **Giao diện đăng ký:** giao diện này dành cho chủ nhà đăng ký các thành viên được cho phép vào nhà mà không cần chủ nhà xác thực.
3. **Giao diện hiển thị trực quan:** đây là giao diện hiển thị các logs mà hệ thống ghi lại bất cứ khi nào có thành viên vào/ra nhà, và hiển thị các biểu đồ trực quan trong ngày.

8.1 Giao diện mô phỏng nhận diện khuôn mặt

Tính năng chính của giao diện mô phỏng nhận diện khuôn mặt là camera tại cửa sẽ luôn chạy, khi người muốn vào nhà thì sẽ nhập id của họ và bấm "Check in" hoặc "Check out".

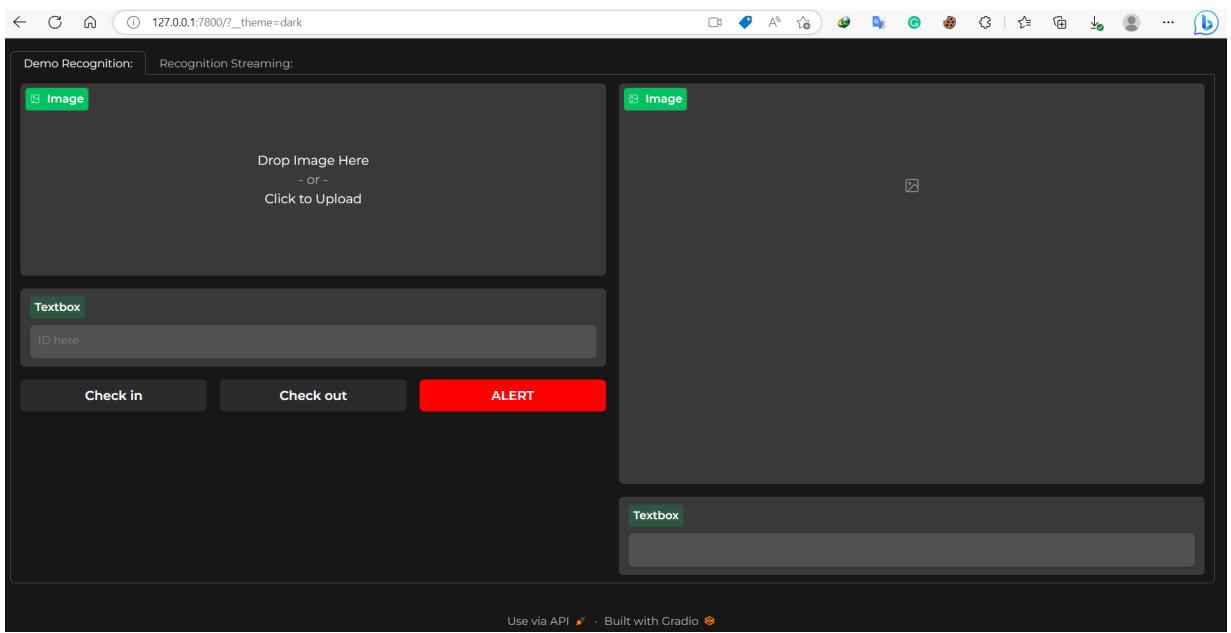
8.1.1 Giao diện chính

Gồm phần chính là nhận dạng khuôn mặt với streaming.



Hình 4: Giao diện dùng để checkin & checkout mặc định

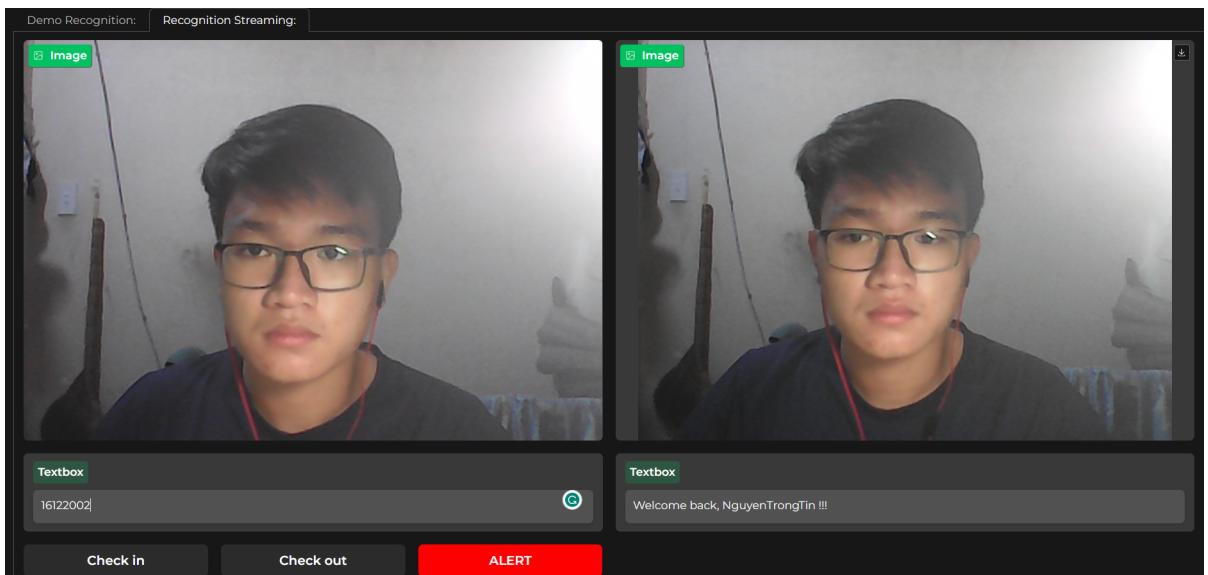
Và phần sử dụng ảnh upload để demo (chỉ dùng khi dùng để demo khả năng nhận dạng nhiều khuôn mặt khác nhau)



Hình 5: Giao diện dùng để checkin & checkout demo

8.1.2 Nhận diện thành công

Khi người dùng được nhận dạng đúng khuôn mặt và ID thì thông báo sẽ hiện lên lời chào kèm đúng tên thành viên.



Hình 6: Giao diện dùng thành công check in

Đồng thời ta có thể kiểm tra ở terminal những thông tin được xử lý:

Checkin thành công ID: 16122002

Hệ thống đã mở đèn trong nhà "Nhan du lieu tu bbc-led : 1"

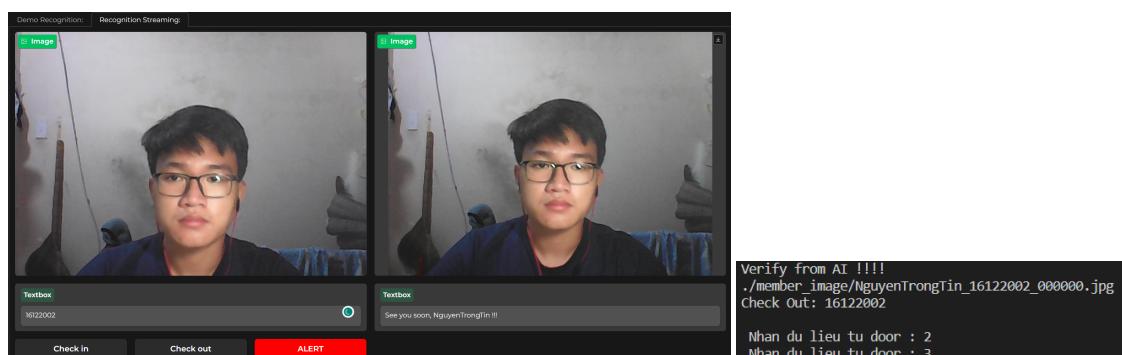
Cửa đã tự động mở với dữ liệu là 2 "Nhan du lieu tu door : 2"

Cửa tự động đóng sau 5 giây (tất nhiên ta có thể điều chỉnh thời gian đóng mở phù hợp) với dữ liệu là 3 "Nhan du lieu tu door : 3"

```
Check In: 16122002
Nhan du lieu tu bbc-led : 1
Nhan du lieu tu door : 2
Nhan du lieu tu door : 3
```

Hình 7

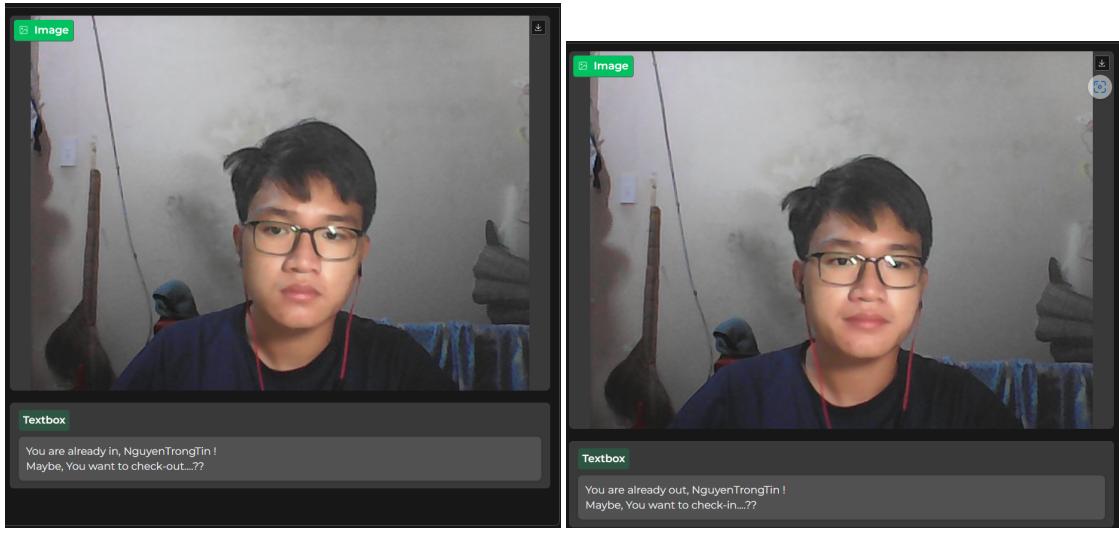
Tương tự khi ta checkout, nếu nhận diện được khuôn mặt và đúng ID thì hệ thống tự động mở cửa, tắt đèn tắt quạt (nếu hệ thống tự động nhận biết bạn là người cuối cùng ra khỏi nhà), và hiện lời chào tạm biệt bạn.



(a) Thành công check out

(b) Terminal

Đôi khi bạn lỡ bấm nhầm, vào nhà hay ra ngoài nhiều hơn 1 lần, hệ thống sẽ kiểm tra lịch sử và nhắc nhở bạn đã bấm nhầm.



(a) Bấm vào nhà nhiều hơn 1 lần

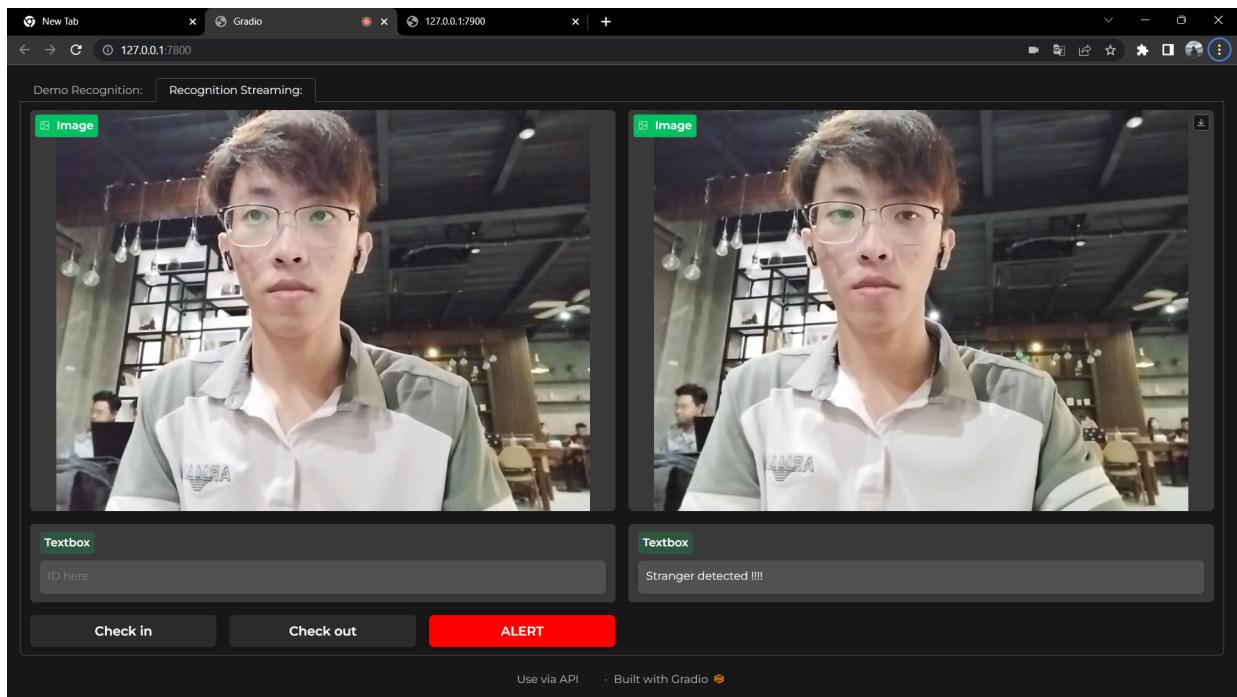
(b) Bấm đi ra nhiều hơn 1 lần

Nếu đôi khi bạn lỡ bấm nhầm mật khẩu ID thì sao? Hệ thống vẫn cho phép bạn nhập là vài lần, và thông báo bạn đã sai ID.



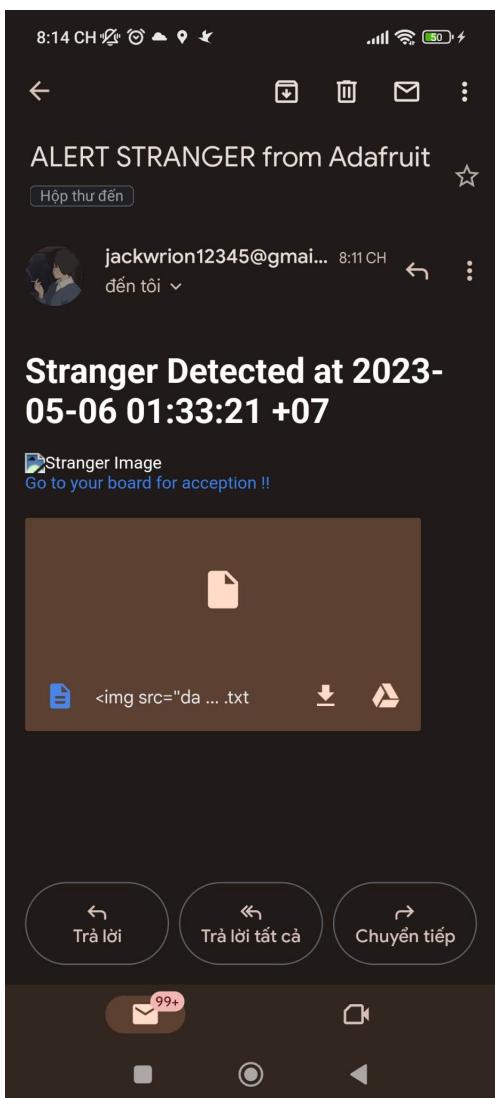
Hình 10: Sai mật khẩu ID

8.1.3 Nhận diện thất bại



Hình 11: Nhận diện thất bại

Khi nhận diện thất bại hệ thống sẽ gửi một email về cho chủ nhà để thông báo có người lạ muốn vào nhà.

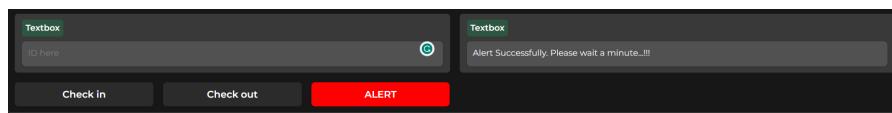


(a) Email gửi về chủ nhà



(b) Ảnh của người lạ

Những trường hợp đặc biệt khi bị kẹt trong nhà do sự cố, hay bạn quên mật khẩu, hãy nhấn vào nút ALERT màu đỏ để thông báo cho chủ nhà biết và xử lý.

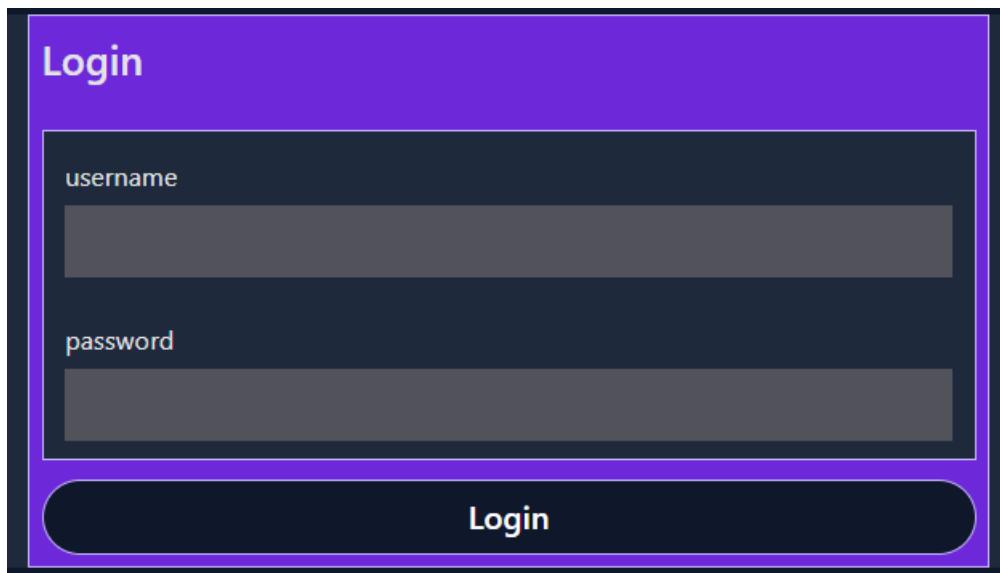


Hình 13: Thông báo hỗ trợ

Hệ thống sẽ tự động gửi khuôn mặt bạn và yêu cầu hỗ trợ qua mail đến chủ nhà.

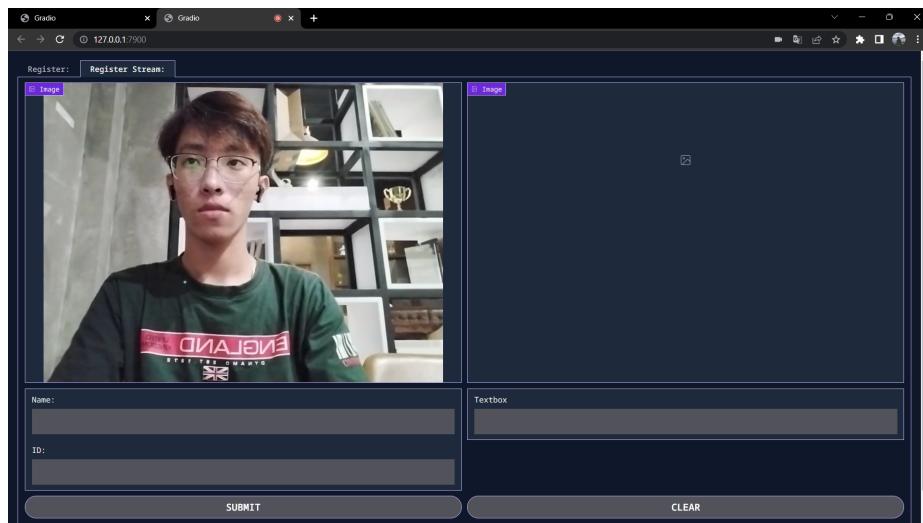
8.2 Giao diện đăng ký

8.2.1 Đăng nhập

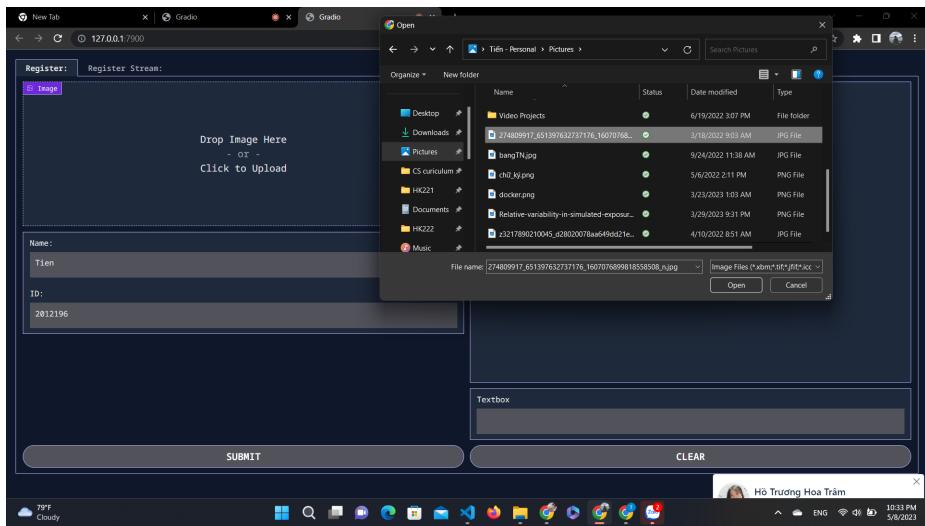


Chủ nhà sẽ đăng nhập tại đây để đăng ký thêm thành viên.

8.2.2 Giao diện chính



Người dùng sẽ nhập tên và ID mình muốn chọn vào và nhấn "Submit". Lưu ý người dùng nên chụp ở nơi có ánh sáng tốt và chụp khoảng 10 ảnh với tất cả các góc mặt của mình để tăng tính chính xác khi nhận diện.

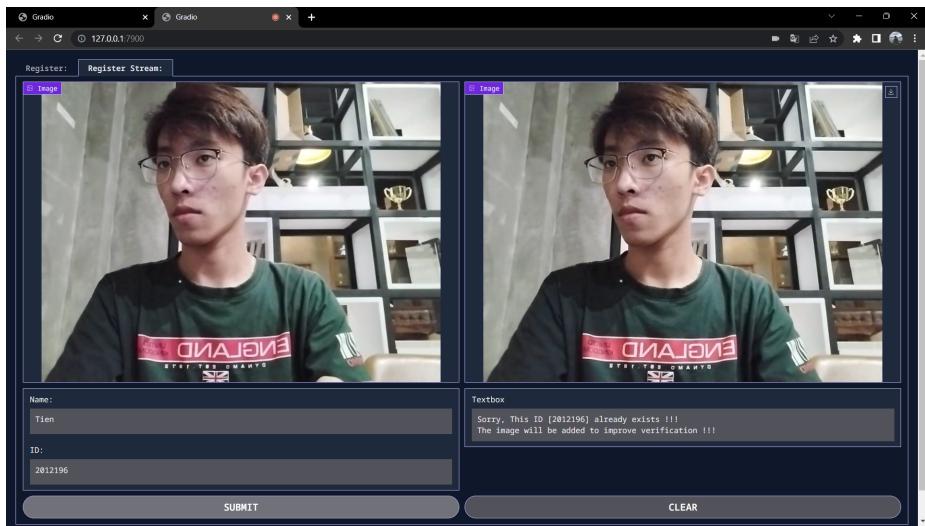


Hình 14: Đăng ký bằng hình ảnh có sẵn

8.2.3 Đăng ký



Hình 15: Ảnh chính diện



Hình 16: Góc mặt khác

Sau khi submit, ảnh của người dùng mới sẽ được lưu vào hệ thống để huấn luyện cho việc nhận diện sau này.

8.3 Dashboard

8.3.1 Log Status Dashboard

The screenshot shows a dashboard titled "Log Status Dashboard". At the top, there are three tabs: "Logs" (which is selected), "Bar Plot", and "Line Plot". Below the tabs, there is a "Show Log" button and a box labeled "Number of Logs" containing the value "102". The main area is a table with the following columns: ID, name, status, type, and date. The data in the table is as follows:

ID	name	status	type	date
111	ElonMusk	1	in	2023-04-20T21:37:18.716000
111	ElonMusk	0	out	2023-04-20T21:37:33.267000
111	ElonMusk	1	in	2023-04-20T21:38:43.408000
222	JeffBezos	1	in	2023-04-20T21:41:21.457000
222	JeffBezos	0	out	2023-04-20T21:42:08.867000
202	Thang	1	in	2023-04-24T19:00:07.258000
202	Thang	0	out	2023-04-24T19:00:15.478000
202	Thang	1	in	2023-04-24T19:00:24.960000
202	Thang	0	out	2023-04-24T19:01:57.417000
222	JeffBezos	1	in	2023-04-25T00:03:33.089000
222	JeffBezos	0	out	2023-04-25T00:03:49.905000
111	ElonMusk	1	in	2023-04-26T15:28:59.372000
222	JeffBezos	1	in	2023-04-26T15:29:26.855000

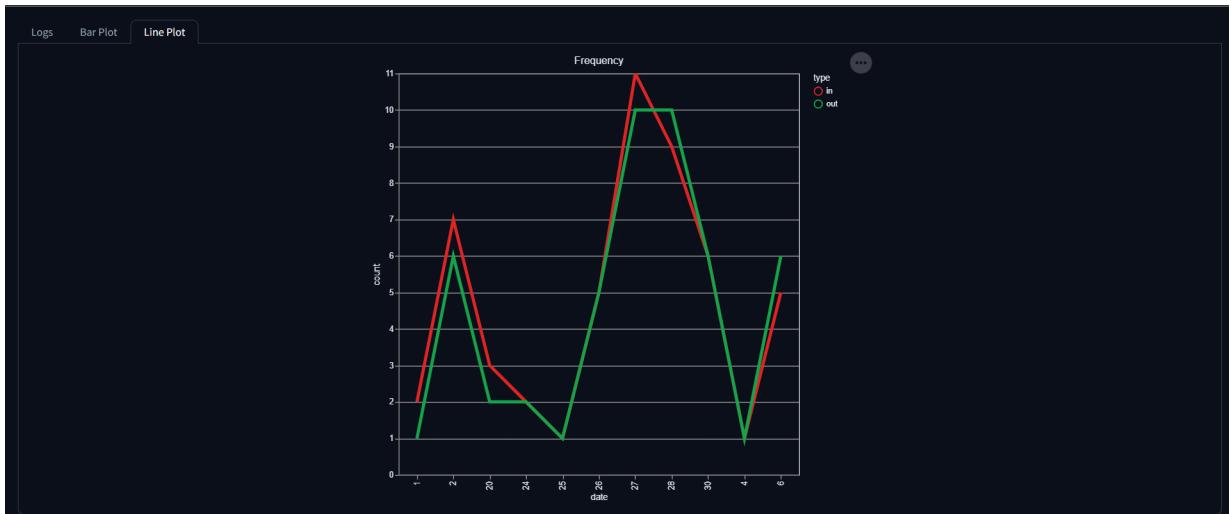
Hình 17: Log Status Dashboard

8.3.2 Hiển thị trực quan dạng biểu đồ cột



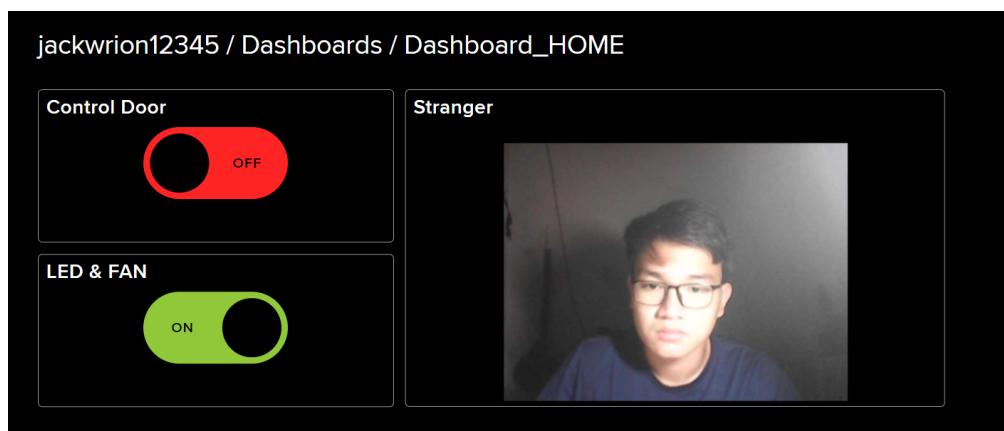
Hình 18: Bar Plot

8.3.3 Hiển thị trực quan dạng biểu đồ đường



Hình 19: Line Plot

8.3.4 Dashboard điều khiển các thiết bị trên Adafruit



Hình 20: Line Plot

9 Resources

9.1 Instruction for use

9.2 Link demo

https://drive.google.com/file/d/1vjjkJzuX587lw1M4jJ36YRLv8KZouKaM/view?fbclid=IwAR3-mtuMeBORj4c-JtysumFMPcncE_cJSh5PTa_5xvLBs90

9.3 Link github

<https://github.com/JackWrion/SmartHomeJWR.git>