

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



Nguyễn Sĩ Tùng

**ỨNG DỤNG NHẬN DIỆN GIỌNG NÓI VÀO
ĐIỀU KHIỂN THIẾT BỊ IOT**

**KHÓA LUẬN TỐT NGHIỆP HỆ CHẤT LƯỢNG CAO
Ngành: Khoa học máy tính CLC**

HÀ NỘI – 2021

**ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

Nguyễn Sĩ Tùng

**ỨNG DỤNG NHẬN ĐIỆN GIỌNG NÓI VÀO
ĐIỀU KHIỂN THIẾT BỊ IOT**

**KHÓA LUẬN TỐT NGHIỆP HỆ CHẤT LƯỢNG CAO
Ngành: Khoa học máy tính CLC**

Cán bộ hướng dẫn: TS. Ma Thị Châu

HÀ NỘI – 2021

**VIETNAM NATIONAL UNIVERSITY
UNIVERSITY OF ENGINEERING AND TECHNOLOGY**

Nguyễn Sĩ Tùng

**SPEECH PROCESSING APPLICATION IN
CONTROLLING IOT DEVICES**

Major: Computer Science CLC

Supervisor: Dr. Ma Thi Chau

HÀ NỘI – 2021

TÓM TẮT

Tóm tắt: Hiện nay, CNTT đang phát triển hết sức mạnh mẽ và được ứng dụng trong mọi mặt của cuộc sống con người. Đối với việc chăm sóc sức khỏe bệnh nhân cũng vậy, những người bệnh, yếu thể luôn cần có được sự quan tâm lớn từ chúng ta. Việc chăm sóc bệnh nhân dựa trên cách truyền thống; người giám hộ, người chăm sóc đảm bảo sức khỏe cho bệnh nhân; vẫn được áp dụng trong hầu hết các bệnh viện. Điều này có thể vẫn mang lại hiệu quả, tuy nhiên, đối với những khoảng thời gian nhạy cảm, đơn cử như thời kì đại dịch Covid-19 đang hoành hành, việc đảm bảo nhân lực hay đảm bảo sức khỏe cho cán bộ y bác sĩ để có thể chăm sóc tất cả bệnh nhân là điều hết sức khó khăn. Đặc biệt là việc chăm sóc đời sống thường ngày đối với những bệnh nhân bị suy giảm khả năng vận động hay những người già yếu. Hiện nay, những thiết bị thông minh đang phát triển rất mạnh và có tính ứng dụng cao vào đời sống con người. Chính vì vậy, việc áp dụng các công nghệ IOT và tương tác người máy HMI vào hỗ trợ người bệnh sẽ tạo được bước cải thiện lớn trong vấn đề này. Đề tài khóa luận này đưa ra một giải pháp, quy trình xây dựng, một hệ thống có thể giải quyết vấn đề nêu trên bằng cách áp dụng việc nhận diện giọng nói vào điều khiển các thiết bị IOT. Khóa luận sẽ tập trung trình bày quá trình giải quyết từ bài toán thực tế đến quá trình xây dựng hệ thống.

Từ khóa: HMI, tương tác người máy

ABSTRACT

Abstract: Currently, Information Technology is strongly, continuously developing and being applied to the whole different aspects of human life. Healthcare is one of the top sectors that demands most attentions. The traditional way; doctors, care givers take care of the patients; is still available in most hospitals. This is still taking some effort practically, however, in some specific periods, for example, the current time where Covid-19 being the tremendous disaster, it is most likely unable for the doctors to handle the huge number of the patients. Taking care of Covid-19 patients and regular patients separately is extremely hard and exhausting. As a consequence, the disableds and the elders do not have enough care in their normal life. Therefore, a combination of IOT devices and human-machine interaction will make a big step forward in patient's healthcare. This thesis topic will discuss and provide solutions, process to construct and design a system that can be able to solve the mentioned problems by applying speech recognition into controlling IOT devices. The thesis will focus on presenting the process of taking the real problem into designing a responsible system.

Key words: HMI – human-machine interaction

LỜI CẢM ƠN

Đầu tiên, xin cho phép em được gửi lời cảm ơn chân thành đến cô Ma Thị Châu đã dành cho em cơ hội được học tập và nghiên cứu trong một môi trường trẻ trung, năng động như phòng thí nghiệm tương tác người máy HMI. Em xin cảm ơn cô đã dành cho em những lời khuyên hữu ích, sự hướng dẫn, hỗ trợ tận tình cũng như động viên và tạo điều kiện cho em có thể nghiên cứu và hoàn thành khóa luận này.

Em cũng xin chân thành cảm ơn các thầy cô và trường Đại học Công Nghệ đã định hướng, giảng dạy và giúp đỡ em trong suốt quá trình học tập và nghiên cứu tại trường.

Em cũng xin được dành những lời cảm ơn đến bạn bè, anh chị và thầy cô tại phòng thí nghiệm tương tác người máy vì đã đồng hành, hỗ trợ em trong suốt quá trình tìm hiểu và xây dựng hệ thống, đưa ra những lời khuyên, góp ý chân thực để giúp cho hệ thống hoạt động tốt hơn và sớm hoàn thiện để đưa tới người dùng.

Em xin chân thành cảm ơn!

Hà Nội, ngày ... tháng ... năm 2020

Tác giả khóa luận

Nguyễn Sĩ Tùng

LỜI CAM KẾT

Tại đây, em xin cam đoan những kiến thức trình bày trong khóa luận “Ứng dụng điều khiển giọng nói và điều khiển thiết bị iot” là do em tự mình tìm hiểu và xây dựng dưới sự hướng dẫn của TS. Ma Thị Châu.

Tất cả các tham khảo từ các tài liệu đều được em nêu ra một cách rõ ràng, đầy đủ trong mục tài liệu tham khảo. Trong khóa luận không có việc sao chép tài liệu, công trình nghiên cứu của người khác mà không chỉ rõ về nguồn gốc của tài liệu.

Em xin chịu mọi trách nhiệm về công trình nghiên cứu của mình!

Hà Nội, ngày ... tháng ... năm 2020

Tác giả khóa luận

Nguyễn Sĩ Tùng

MỤC LỤC

CHƯƠNG 1: GIỚI THIỆU CHUNG	14
1.1. Đặt vấn đề	14
1.2. Bố cục khóa luận	16
CHƯƠNG 2: THU THẬP VÀ PHÂN TÍCH YÊU CẦU	17
2.1. Các yêu cầu chức năng của hệ thống	17
2.1.1. Yêu cầu của người quản lý (admin)	17
2.1.2. Yêu cầu của người giám hộ (bác sĩ, người nhà bệnh nhân)	17
2.1.3. Yêu cầu của người bệnh (người suy giảm khả năng vận động)	17
2.2. Các yêu cầu phi chức năng của hệ thống	18
CHƯƠNG 3: CÔNG NGHỆ SỬ DỤNG	18
3.1. FrontEnd	18
3.1.1. Reactjs	18
3.2. BackEnd	19
3.2.1. Spring boot	19
3.2.2. MariaDB	19
3.3. Deploy (triển khai)	20
3.4. Bộ điều khiển thiết bị thông minh	20
3.4.1. Giao thức MQTT	20
3.4.2. Mosquitto broker	21
3.4.3. LIRC	21
3.5. Bộ chuyển đổi lệnh giọng nói	23
3.5.1. Python SpeechRecognition	23
3.5.2. Google text-to-speech	23
CHƯƠNG 4: XÂY DỰNG VÀ TRIỂN KHAI HỆ THỐNG	24
4.1. Mô hình ca sử dụng (Use-Case Model)	24
4.1.1. Người bệnh	27
4.1.1.1. Điều khiển các thiết bị IOT thông qua giọng nói	27
4.1.2. Người giám hộ	29
4.1.2.1. Đăng nhập vào hệ thống	29
4.1.2.2. Thêm tệp cấu hình điều khiển hồng ngoại	30
4.1.2.3. Chỉnh sửa thông tin danh sách thiết bị được đăng ký	31
4.1.2.4. Giám sát thống kê thông tin người bệnh	32
4.1.3. Người quản lý (admin)	33
4.1.3.1. Quản lý thông tin điều khiển tương ứng với từng loại thiết bị	34

4.1.3.2.	Giám sát thống kê thông tin người bệnh	35
4.2.	Phân tích hệ thống	36
4.2.1.	Tổng quan kiến trúc hệ thống	37
4.2.1.1.	Module quản lý thiết bị IOT	38
4.2.1.2.	Module chuyển đổi lệnh giọng nói	38
4.2.1.3.	Module điều khiển thiết bị thông minh	39
4.2.2.	Phân tích kiến trúc – Key abstraction	40
4.3.	Thiết kế hệ thống phía máy chủ	41
4.3.1.	Xác định các thực thể và mối quan hệ giữa chúng	41
4.3.2.	Mô hình thực thể các mối quan hệ (ERD)	41
4.3.3.	Thiết kế cơ sở dữ liệu	42
4.4.	Xây dựng hệ thống phía máy chủ	43
4.4.1.	Các đầu APIs	44
4.4.2.	Xác thực người dùng với JWT	45
4.5.	Xây dựng hệ thống web phía người dùng admin	47
4.6.	Xây dựng hệ thống phía bộ chuyển đổi lệnh giọng nói	48
4.7.	Xây dựng hệ thống phía bộ điều khiển thiết bị thông minh	53
CHƯƠNG 5:	KẾT QUẢ	56
5.1.	Hình ảnh mô tả hệ thống về đăng ký thiết bị	56
5.1.1.	Giao diện đăng nhập hệ thống	56
5.1.2.	Giao diện thống kê thiết bị	57
5.1.3.	Giao diện thêm mới thiết bị	58
5.1.4.	Giao diện thông số dữ liệu	60
5.1.5.	Giao diện thêm thông tin tệp cấu hình điều khiển	62
5.1.6.	Giao diện thay đổi cấu hình các nút điều khiển thiết bị	63
CHƯƠNG 6:	KẾT LUẬN VÀ PHƯƠNG HƯỚNG PHÁT TRIỂN	64

DANH MỤC HÌNH ẢNH

Hình 1-1. Sơ đồ tổng thể của toàn bộ hệ thống lớn	15
Hình 4-1. Mô hình ca sử dụng – các tác nhân của hệ thống.....	24
Hình 4-2. Mô hình ca sử dụng của người bệnh	25
Hình 4-3. Mô hình ca sử dụng của người giám hộ bệnh nhân.....	26
Hình 4-4. Mô hình ca sử dụng của người quản lý.....	27
Hình 4-5. Kiến trúc hệ thống.....	37
Hình 4-6. Module quản lý thiết bị IOT	38
Hình 4-7. Module chuyển đổi lệnh giọng nói.....	38
Hình 4-8. Module điều khiển thiết bị thông minh	39
Hình 4-9. Mô hình thực thể mối quan hệ	42
Hình 4-10. Thiết kế cơ sở dữ liệu.....	43
Hình 4-11. Api xác thực tài khoản.....	44
Hình 4-12. Api quản lý thiết bị.....	44
Hình 4-13. Api quản lý thông tin nút điều khiển hồng ngoại.....	44
Hình 4-14. Api quản lý dữ liệu thu thập.....	45
Hình 4-15. Api thêm mới tệp cấu hình điều khiển hồng ngoại.....	45
Hình 4-16. Api quản lý mẫu model thiết bị được hỗ trợ	45
Hình 4-17. Ý tưởng của mô hình SPA.....	48
Hình 4-18. Lệnh “xin chào”	49
Hình 4-19. Lệnh “liệt kê thiết bị”	50
Hình 4-20. Lựa chọn thiết bị điều khiển	51
Hình 4-21. Danh sách các câu lệnh tương ứng với từng loại thiết bị	52
Hình 4-22. Lệnh giọng nói và mẫu payload gửi đến bộ điều khiển thiết bị thông minh.....	53
Hình 4-23. Mẫu dữ liệu điều khiển smart switch thông qua giao thức MQTT	54
Hình 4-24. Mẫu câu lệnh sử dụng trong điều khiển bằng sóng hồng ngoại	55
Hình 5-1. Giao diện đăng nhập hệ thống.....	56
Hình 5-2. Giao diện thống kê thiết bị.....	57
Hình 5-3. Giao diện thêm thiết bị sensor.....	58
Hình 5-4. Giao diện thêm thiết bị switch	59
Hình 5-5. Giao diện thêm thiết bị với các thiết bị điều khiển bằng hồng ngoại	59
Hình 5-6. Giao diện thông số dữ liệu với dữ liệu từ sensor.....	60
Hình 5-7. Giao diện thông số dữ liệu với các dữ liệu điều khiển	61
Hình 5-8. Giao diện thêm tệp cấu hình điều khiển thiết bị hồng ngoại 1.....	62
Hình 5-9. Giao diện thêm tệp cấu hình điều khiển thiết bị hồng ngoại 2.....	63
Hình 5-10. Giao diện thay đổi thông tin các nút điều khiển hồng ngoại	63

DANH MỤC BẢNG

Bảng 1. Danh sách Key abstraction của hệ thống

41

BẢNG KÝ HIỆU CHỮ VIẾT TẮT

STT	Chữ viết tắt	Ý nghĩa
1	CNTT	Công nghệ thông tin
2	IOT	Internet of things
3	SPA	Ứng dụng web chỉ có một trang
4	RPI4	Raspberry Pi 4
5	DOM	Document Object Model
6	MQTT	Message Queuing Telemetry Transport
7	TLS	Transport Layer Security
8	SSL	Secure Sockets Layer
9	LIRC	Linux Infrared remote control
10	LED	Light-emitting-Diode

MỞ ĐẦU

Hiện nay, trong thời đại công nghệ 4.0, mọi công việc, nhiệm vụ, sinh hoạt trong cuộc sống con người đang dần được số hóa. Rất nhiều nghiên cứu, ứng dụng đã được đưa vào sử dụng thực tế, trong đó, tương tác người máy là một chủ đề rất tiềm năng trong việc hỗ trợ con người trong các vấn đề cuộc sống thường ngày. Chăm sóc bệnh nhân luôn là vấn đề được cả xã hội quan tâm, giành được nhiều cống hiến và công sức bỏ ra từ những nhà nghiên cứu và bác sĩ. Đây không chỉ là việc điều trị bệnh cho con người mà còn mang vấn đề về nhân đạo. Ở Việt Nam, chúng ta luôn có những chủ trương, kế hoạch nhằm tri ân những người lính, chiến sĩ đã chiến đấu hết mình vì tổ quốc. Trong số hàng triệu người đó, rất nhiều người phải mang bên mình những thương tật, không thể sinh hoạt, làm việc bình thường do di chứng để lại từ các chấn thương chiến tranh. Tương tự như vậy, với các trường hợp là bệnh nhân, họ cần có nhiều sự trợ giúp hơn trong cuộc sống. Để có thể đáp ứng một phần nào trong những nhu cầu trên, phòng thí nghiệm tương tác người máy HMI đã triển khai một đề tài nhằm ứng dụng những công nghệ, tương tác người máy vào trong hỗ trợ các bệnh nhân bị suy giảm khả năng vận động.

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Đặt vấn đề

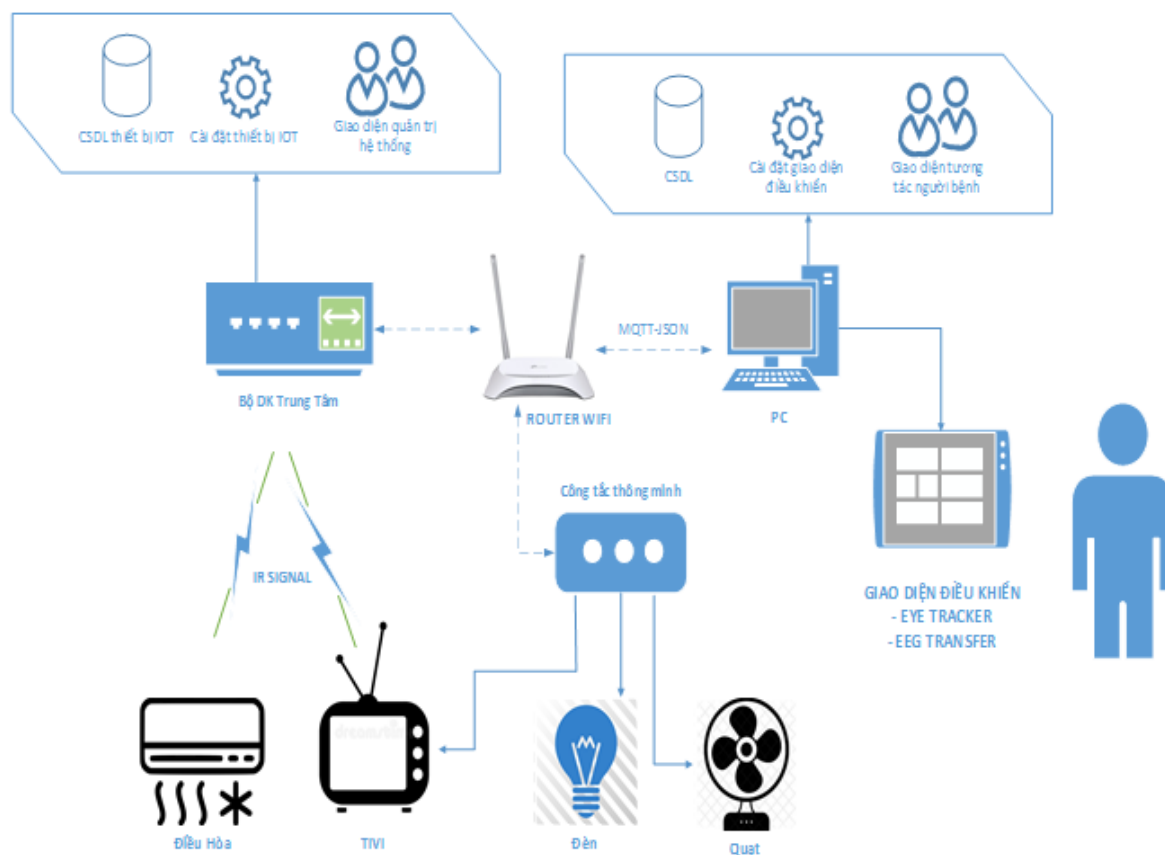
Sự phát triển của CNTT nói chung đang đạt được tốc độ đáng kinh ngạc, đặc biệt là trong thời kì cách mạng công nghệ 4.0. Kéo theo đó là những nhu cầu áp dụng những giải pháp tiện ích vào trong mọi mặt của cuộc sống cũng đang đạt được lượng quan tâm khổng lồ. Vấn đề chăm sóc sức khỏe người bệnh cũng là một yếu tố rất được chú ý trong thời đại hiện nay, khi con người ngày càng có ít đi thời gian chăm sóc bản thân và gia đình. Vì vậy một hệ thống có thể giúp người bệnh tự chủ được những sinh hoạt cá nhân và có thể giúp người giám hộ giám sát được tình trạng người bệnh là rất cần thiết.

Theo thống kê vào năm 2019¹[1], chỉ riêng ở Việt Nam đã có 6,2 triệu người khuyết tật, chiếm 7,06% dân số cả nước có độ tuổi từ 2 tuổi trở lên. Đây là một con số không hề nhỏ. Những người này cần được chăm sóc từ những người xung quanh do họ đã bị mất đi một phần nào đó khả năng sinh hoạt thường ngày. Tuy nhiên, hầu hết với tâm lý người bệnh, họ sẽ không muốn bản thân nhận được quá nhiều sự chăm sóc đến từ bên ngoài do điều đó sẽ mang cho họ cảm giác bất lực, khó chịu khi không thể tự chăm lo cho chính bản thân mình.

Trong số hàng triệu người khuyết tật kể trên, chắc chắn có tỉ lệ lớn những người bị suy giảm khả năng vận động, di chuyển. Những trường hợp này có thể bị gây nên do bẩm sinh, những chấn thương vật lý khi lao động, sinh hoạt, hay đến từ những thương tổn về hệ thần kinh trung ương (đột quỵ não, u não, áp xe não, viêm não, chứng bệnh Parkinson giai đoạn cuối). Đây là những người cần nhiều sự quan tâm, chăm sóc do họ không thể tự mình di chuyển và đáp ứng nhu cầu sinh hoạt của bản thân. Như vậy, đối với người bệnh, việc sử dụng được những thiết bị gia dụng điện tử trong nhà là rất khó khăn nếu như không có được sự giúp đỡ. Yêu cầu đặt ra là xây dựng được một hệ thống có thể hỗ trợ người bệnh điều khiển các thiết bị từ xa, không yêu cầu di chuyển khỏi vị trí hiện tại; Thêm nữa, người giám hộ hoặc bác sĩ điều trị có thể giám sát được môi trường xung quanh người bệnh, các biểu hiện của người bệnh thông qua các hành vi sử dụng thiết bị để có thể nắm bắt được tâm lý hay nhu cầu của người bệnh.

¹ “Viet Nams first large scale national survey people disabilities – unicef.org”,
<https://www.unicef.org/vietnam/press-releases/launch-key-findings-viet-nams-first-large-scale-national-survey-people-disabilities>

Dựa vào những yêu cầu thực tiễn nêu trên, Bằng cách sử dụng những công nghệ như nhà thông minh, những cảm biến, những thiết bị có khả năng điều khiển từ xa, thông minh hệ thống này sẽ được thiết kế và xây dựng với sơ đồ như sau:



Hình 1-1. Sơ đồ tổng thể của toàn bộ hệ thống lớn

Sơ đồ trên bao gồm: i. User interface – Eye tracker – EEG transfer: Đây là giao diện người dùng để điều khiển hệ thống (user interface), thiết bị theo dõi mắt (eye tracker) và thiết bị theo dõi tín hiệu điện não (eeg transfer), ii. PC: Nơi cài đặt các phần mềm phân tích, xử lý, tính toán và kết nối các dữ liệu từ giao diện người dùng, thiết bị theo dõi mắt và thiết bị theo dõi tín hiệu điện não cũng như tới các thiết bị thông minh trong nhà, iii. Router wifi: các thiết bị được kết nối với nhau qua đường mạng Lan hoặc wifi nên cần sử dụng 1 bộ định tuyến ở đây. Và có thể phục vụ cho việc giám sát, sử dụng thiết bị trực tuyến sau này, iv. Smart home controller: Đây là bộ điều khiển trung tâm của các thiết bị phần cứng. Nó chịu trách nhiệm quản lý thêm/bớt các thiết bị như công tắc và cụ thể ví dụ như đèn 1, đèn 2, quạt 1, quạt 2, Cũng như xử lý luồng dữ liệu nhận được từ PC để có

thể điều khiển tới chính xác thiết bị người sử dụng cần điều khiển. Thiết bị sử dụng kết nối Wifi hoặc Ethernet, v. Smart switch: Đây là công tắc thông minh, kết nối qua wifi tới bộ định tuyến để từ đó giao tiếp với toàn bộ hệ thống. Công tắc thông minh này chịu trách nhiệm điều khiển bật/tắt các thiết bị như đèn, quạt, tivi qua phương thức cấp hoặc không cấp nguồn điện cho thiết bị được điều khiển. Chi tiết hơn, hệ thống sẽ bao gồm các chức năng chính sau đây:

1. Thêm mới các tín hiệu điều khiển cho các thiết bị IOT
2. Đăng ký và quản lý các thiết bị
3. Thống kê thông tin từ các thiết bị ngoại vi và hành vi sử dụng thiết bị của người dùng
4. Điều khiển các thiết bị thông qua giọng nói
5. Điều khiển các thiết bị thông qua tương tác ánh nhìn
6. Đưa ra những biểu hiện bất thường dựa theo thông tin từ các thiết bị và hành vi sử dụng thiết bị

Theo đó, khóa luận này tập trung vào trình bày một phần của hệ thống đã được xây dựng bao gồm các chức năng: thêm mới các tín hiệu điều khiển cho các thiết bị IOT, đăng ký và quản lý các thiết bị, thống kê thông tin từ các thiết bị ngoại vi và hành vi sử dụng thiết bị của người dùng, điều khiển các thiết bị thông qua giọng nói. Những chức năng không được nêu tên sẽ được tiết tục xây dựng trong tương lai để hệ thống dần trở nên hoàn thiện hơn và có thể hỗ trợ được nhiều khía cạnh hơn cho người bệnh.

1.2. Bố cục khóa luận

Phạm vi của luận án sẽ trình bày chi tiết về ứng dụng nhận diện giọng nói vào điều khiển các thiết bị iot có cấu trúc như sau:

Chương 1: Giới thiệu chung

Chương 2: Thu thập và phân tích yêu cầu

Chương 3: Công nghệ sử dụng

Chương 4: Xây dựng và triển khai hệ thống

Chương 5: Kết quả

Chương 6: Kết luận và phương hướng phát triển

CHƯƠNG 2: THU THẬP VÀ PHÂN TÍCH YÊU CẦU

2.1. Các yêu cầu chức năng của hệ thống

2.1.1. Yêu cầu của người quản lý (admin)

Quản lý thông tin các phím điều khiển tương ứng với từng thiết bị: Mỗi thiết bị thuộc các nhóm khác nhau sẽ có số lượng phím hỗ trợ khác nhau, người quản lý có thể thay đổi thông tin này dựa theo các thiết bị điều khiển hiện có.

Chức năng thêm thông tin điều khiển hồng ngoại cho các thiết bị: Thêm các thông tin điều khiển hồng ngoại của các thiết bị được lắp đặt tại nơi ở của người bệnh.

Quản lý các thiết bị trong hệ thống: Thêm, sửa, xóa, tìm kiếm thông tin các thiết bị, thông tin cần thiết để đảm bảo thiết bị hoạt động và đưa ra thông tin ghi chú để người bệnh nhận biết được thiết bị mong muốn điều khiển.

Chức năng giám sát và thống kê dữ liệu từ người bệnh: Có thể theo dõi, giám sát biểu hiện của người dùng và môi trường xung quanh người dùng thông qua các bảng biểu thống kê dữ liệu sử dụng của người bệnh như: thông tin về các sensor thu tín hiệu về cơ thể hay môi trường, thông tin điều khiển các thiết bị của người bệnh để thấy được họ đang sử dụng với tần suất bao nhiêu và vào khung giờ nào.

2.1.2. Yêu cầu của người giám hộ (bác sĩ, người nhà bệnh nhân)

Chức năng thêm thông tin điều khiển hồng ngoại cho các thiết bị: Người giám hộ của bệnh nhân cũng có thể thêm thông tin điều khiển hồng ngoại hỗ trợ các loại thiết bị sử dụng sóng hồng ngoại để điều khiển.

Quản lý các thiết bị trong hệ thống: Chức năng tương tự như với yêu cầu của người quản lý, thêm, sửa, xóa các thiết bị vào hệ thống.

Chức năng giám sát và thống kê dữ liệu từ người bệnh: Người giám hộ cần biết được thông tin về người nhà, bệnh nhân của mình thông qua dữ liệu thu thập được từ hệ thống.

2.1.3. Yêu cầu của người bệnh (người suy giảm khả năng vận động)

Điều khiển các thiết bị thông qua giọng nói: Có thể điều khiển các thiết bị thông qua giọng nói và tên các thiết bị mà không cần đến những tương tác trực tiếp với thiết bị tại nơi ở của mình.

2.2. Các yêu cầu phi chức năng của hệ thống

Đối với yêu cầu phi chức năng, hệ thống được đặt trên các tiêu chuẩn sau: i. Hệ thống hoạt động ổn định, phản hồi nhanh, ii. Hệ thống tiếp tục hoạt động khi có lỗi, iii. Hệ thống tiêu tốn ít tài nguyên phần cứng.

CHƯƠNG 3: CÔNG NGHỆ SỬ DỤNG

Do người dùng cần có thể truy cập và sử dụng hệ thống một cách thuận tiện nhất mọi lúc nên hệ thống sẽ được viết trên nền tảng Web. Hơn nữa, những phần được trình bày trong khóa luận sẽ được đặt trên phần cứng RPI4, một phần cứng có sức mạnh không quá tốt so với những ứng dụng thông thường; vì vậy cần lựa chọn tốt công nghệ để đảm bảo hiệu năng cho phần cứng và phần mềm. Chương này sẽ giới thiệu các công nghệ và kỹ thuật mà em cảm thấy thật sự phù hợp và cần thiết để xây dựng hệ thống

3.1. FrontEnd

3.1.1. *Reactjs*

ReactJS²[2] là một thư viện Javascript mã nguồn mở được facebook sáng tạo và đồng phát triển, duy trì cùng với các nhà phát triển mã nguồn mở và các công ty cá nhân; được sử dụng để xây dựng các trang web, giao diện người dùng hoặc các thành phần nhỏ nằm trong chúng. Reactjs có thể nhìn nhận và sử dụng trong thực tế như một nền tảng, framework để xây dựng các SPA hay các ứng dụng di động.

Việc lựa chọn ReactJs làm công nghệ để xây dựng frontend dựa trên những lý do sau đây:

Thứ nhất, việc đặt ứng dụng trên một phần cứng như Raspberry Pi 4 làm hạn chế đi yếu tố performance. ReactJS được lựa chọn do ReactJS rất đơn giản, nhẹ, không tốn tài nguyên, hơn nữa, thư viện này còn rất dễ học, kể cả đối với những người mới làm việc, bắt đầu. Với một nguồn tư liệu lớn, bất kỳ người dùng nào cũng có thể cơ bản nắm bắt và phát triển ứng dụng của riêng mình chỉ sau một vài ngày bỏ ra tìm hiểu về ReactJS. Thứ hai, ReactJS cung cấp khả năng dễ dàng quản lý trạng thái của ứng dụng web với những thuộc tính props và state. Điều này giúp lập trình viên dễ dàng quản lý các luồng cũng như thay đổi trong việc hiển thị thông tin trên ứng dụng của mình. Thứ ba, DOM ảo (virtual DOM) cũng được ReactJs sử dụng, từ đó làm cho việc render ứng dụng nhanh

² "Why You Should Use React.js For Web Development - freeCodeCamp." 18 thg 2. 2021, <https://www.freecodecamp.org/news/why-use-react-for-web-development/>.

hơn, hiệu suất. Cuối cùng, ReactJS cũng hỗ trợ những thư viện UI (giao diện người dùng) nhằm giúp dễ dàng hơn trong việc thiết kế ra những trang web có tính thẩm mỹ, dễ dàng sử dụng và đúng quy cách.

3.2. BackEnd

3.2.1. Spring boot

Spring boot³[3] là một dự án được xây dựng và phát triển bởi JAV (ngôn ngữ Java) và nó nằm trong hệ sinh thái Spring (Spring framework). Spring boot là công cụ dựa trên nền tảng ngôn ngữ lập trình Java, giúp đơn giản hóa quá trình lập trình một ứng dụng Spring và tập trung nhiều hơn vào việc phát triển yêu cầu phần mềm hay business của hệ thống

Lí do Spring boot được lựa chọn sử dụng trong ứng dụng này bao gồm những lý do kể sau:

Thứ nhất, Spring boot là một framework nhẹ, phù hợp với lượng tài nguyên của RPI4. Thứ hai, Spring boot tích hợp sẵn web server như Tomcat hay Netty nên ứng dụng được tạo ra ngay trong quá trình khởi tạo bởi Spring boot có thể được chạy ở bất cứ đâu mà không cần quan tâm đến web server bên ngoài. Ứng dụng spring boot còn có thể được gọi là “just run” cũng vì lý do này. Thứ ba Spring boot hỗ trợ khả năng tự thiết lập (Autoconfiguration), cung cấp cho lập trình viên sự linh động trong việc quản lý các dependency trong ứng dụng. Các dependency có thể được Spring boot tự động thiết lập hoặc được thiết lập thủ công bởi lập trình viên bằng cách ghi đè lên các thiết lập tự động đó. Cuối cùng, Spring boot có khả năng tương tác tốt với các loại cơ sở dữ liệu hiện hành. Thông qua việc sử dụng Hibernate, người dùng có thể dễ dàng truy xuất cơ sở dữ liệu từ ứng dụng bằng các thiết lập tự động của framework.

3.2.2. MariaDB

MariaDB⁴[4] là một hệ quản trị cơ sở dữ liệu được phát triển bởi Michael “Monty” Widenius, nhà phát triển hàng đầu của MySQL và đang được đông đảo cộng đồng sử dụng. Mariadb có khả năng tương thích với nhiều hệ điều hành khác nhau bao gồm Windows, Linux, CentOS, Ubuntu.

³ “What is Spring Boot”,
<https://stackify.com/what-is-spring-boot/>

⁴ “Mariadb – mariadb.org”
<https://mariadb.org/>

Thông thường, trong một hệ thống IOT, cơ sở dữ liệu sẽ được lựa chọn là các loại cơ sở dữ liệu NoSQL như InfluxDB, MongoDB hay Couchbase. Tuy nhiên, đối với hệ thống này, cần phải xem xét khả năng tương thích của các loại cơ sở dữ liệu đó với hệ điều hành của RPI4 và các công cụ sử dụng trên RPI4. Đối với các mạch điện tử hay Arduino, luôn có các chân cắm để tích hợp thêm những mạch khác để tạo ra thiết bị hoàn chỉnh, vì vậy, hệ điều hành sử dụng trên các mạch này cũng cần phải hỗ trợ việc thao tác với những chân cắm; Với RPI4, không có nhiều các hệ điều hành có thể làm được những điều này, vì vậy mối ràng buộc giữa hệ điều hành của RPI4, các cơ sở dữ liệu được hỗ trợ trên hệ điều hành đó và được hỗ trợ bởi Spring boot là điều cần được xem xét kỹ càng. Mariadb là cơ sở dữ liệu có đầy đủ khả năng để đảm bảo được việc lưu trữ dữ liệu thời gian thực từ các thiết bị cảm biến và hành vi người dùng cũng như được hỗ trợ đầy đủ từ cả RPI4 và Spring boot cũng như hoàn toàn miễn phí

3.3. Deploy (triển khai)

Đối với các hệ thống, ứng dụng web thông thường, những nhà phát triển hầu hết sẽ có những công cụ tiện ích để triển khai hệ thống lên môi trường thật như các công cụ CI/CD tự động triển khai phần mềm như Jenkins, Gitlab hay các công cụ quản lý cài đặt tài nguyên cho ứng dụng như Docker, Kubernetes. Những công cụ này đều là những công cụ hàng đầu và tối quan trọng trong phát triển các ứng dụng lớn. Tuy nhiên, chúng không được lựa chọn sử dụng trong việc triển khai hệ thống lên phần cứng RPI4 do chúng có thể gây ra hao tổn thêm tài nguyên sử dụng, điều đó trực tiếp ảnh hưởng tới tốc độ xử lý của hệ thống và trải nghiệm người dùng. Đối với phần cứng không có quá nhiều tài nguyên, mọi sự sử dụng đều nên được cân nhắc kỹ càng và tiết kiệm để đảm bảo hiệu năng. Do vậy, tại hệ thống này, ứng dụng sẽ được triển khai theo dạng nguyên thủy trên chính server của ứng dụng đó.

3.4. Bộ điều khiển thiết bị thông minh

3.4.1. Giao thức MQTT

Giao thức MQTT⁵[5] là giao thức truyền thông điệp (message) dựa trên mô hình publish/subscribe (nhà cung cấp, thuê bao sử dụng), được sử dụng rộng rãi cho các thiết bị IOT với băng thông thấp, độ tin cậy cao và khả năng được sử dụng trong môi trường mạng lưới không ổn định. Kiến trúc tầng cao của giao thức này bao gồm Broker và

⁵ “MQTT và vai trò trong IOT - viblo”,
<https://viblo.asia/p/mqtt-la-gi-vai-tro-cua-mqtt-trong-iot-V3m5WL3bKO7>

Clients. Trong đó, Broker được coi như phần trung tâm, là điểm giao, phân phối của tất cả các kết nối đến từ client. Broker có nhiệm vụ nhận message từ publisher, đưa các message vào hàng đợi và chuyển tiếp các message này đến topic cụ thể. Clients được chia thành hai nhóm là các publishers và các subscribers. Client hầu hết là các thành phần nhỏ được đặt tại các thiết bị đầu cuối và có thiết kế có thể hoạt động một cách linh hoạt. Một client đảm nhiệm ít nhất một trong hai chức năng tương ứng với hai nhóm đó là publish message lên một topic cụ thể hoặc subscribe vào topic để nhận message từ topic này. Trong hệ thống giao thức MQTT được sử dụng như cổng chung chuyển các tín hiệu điều khiển thiết bị nhận được từ người dùng cũng như thực hiện điều khiển các thiết bị hỗ trợ điều khiển thông qua giao thức này.

3.4.2. Mosquitto broker

Mosquitto⁶[6] là một MQTT Broker mã nguồn mở cho phép truyền nhận dữ liệu thông qua giao thức MQTT. Mosquitto được phát triển bởi Eclipse Foundation và nằm trong dự án phát triển iot của eclipse. Mosquitto có ưu điểm là tốc độ truyền nhận dữ liệu cũng như xử lý dữ liệu nhanh, có độ ổn định cao và rất phù hợp với những ứng dụng nhúng. Mosquitto cũng hoàn toàn phù hợp với hệ thống do có thể được cài đặt trên RPI4 một cách dễ dàng và chiếm không quá nhiều tài nguyên hệ thống. Hơn nữa, Mosquitto cũng hỗ trợ giao thức TLS/SSL có thể sử dụng để xác thực client server cũng như mã hóa các message. Mosquitto là một sự lựa chọn hợp lý cho hệ thống hiện tại.

3.4.3. LIRC

LIRC⁷[7] là một gói mã nguồn mở dựa trên các hệ điều hành Linux, cung cấp cho người dùng khả năng đọc và phát tín hiệu hồng ngoại. Với LIRC được cài đặt trên hệ thống và IR receiver, người dùng có thể điều khiển hầu hết tất cả các thiết bị sử dụng điều khiển hồng ngoại như TV, điều hòa. Ngoài ra, LIRC còn cung cấp khả năng tạo các tệp cấu hình hỗ trợ cho việc nhận dạng tín hiệu điều khiển và đặt tên cho chúng. Điều này khiến cho LIRC thực sự hữu ích trong một ứng dụng IOT đòi hỏi đến việc điều khiển thiết bị. Hơn thế nữa, LIRC cũng có cộng đồng đóng góp các tệp mẫu này nên số lượng thiết bị

⁶ “Mosquitto là gì? – mosquitto.org”,
<https://mosquitto.org/>

⁷ “LIRC là gì? – lirc.org”
<https://www.lirc.org/>

có thể điều khiển và có sẵn các tệp cấu hình này cũng không hề nhỏ, có thể giảm bớt nhiều thời gian tạo ra chúng.

3.5. Bộ chuyển đổi lệnh giọng nói

Đối với bộ chuyển đổi lệnh giọng nói cũng sử dụng công nghệ là giao thức MQTT tương tự với bộ điều khiển trung tâm, tuy nhiên sẽ có thêm hai công nghệ sau:

3.5.1. *Python SpeechRecognition*

Python SpeechRecognition⁸[8] là một công cụ speech-to-text sử dụng rất hiệu quả và được sử dụng trên nền tảng ngôn ngữ python – một ngôn ngữ tối ưu cho việc xử lý dữ liệu. Đây là thư viện rất phù hợp cho mục đích tự tạo ra các ngôi nhà thông minh, nhận dạng giọng nói con người... Ngoài ra, thư viện này đóng gói nhiều công cụ nhận diện giọng nói bao gồm: Google speech recognition, google cloud speech api, IBM speech to text..., những công cụ này khiến cho thư viện trở nên linh hoạt hơn với các hình thức sử dụng.

3.5.2. *Google text-to-speech*

Google text-to-speech⁹ (gTTS) là một thư viện của ngôn ngữ python được phát triển bởi google. Công việc chính của công nghệ này chính là nhận dữ liệu đầu vào từ người dùng là dữ liệu thuộc dạng văn bản (text), sử dụng những phương thức xử lý ngôn ngữ tự nhiên để nhận dạng được ngôn ngữ đầu vào và đưa ra các đầu ra tương ứng. Trong hệ thống này, text-to-speech đóng vai trò quan trọng khi làm cầu nối để hệ thống có thể phản hồi lại kết quả cho người dùng về các lệnh đã được đưa ra.

⁸ “Trang chủ Speech Recognition – github.com”

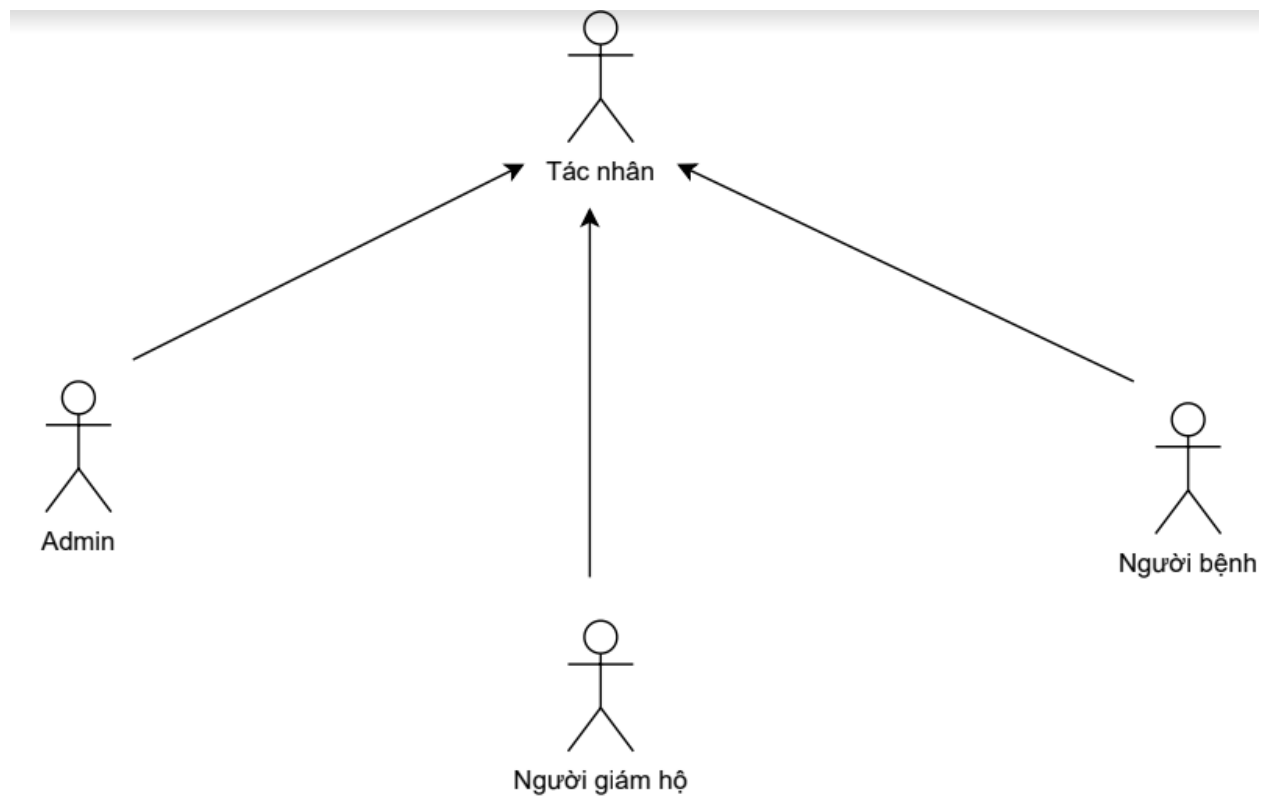
https://github.com/Uberi/speech_recognition#readme

⁹ “How to get started with google text-to-speech – towardsdatascience.com”,

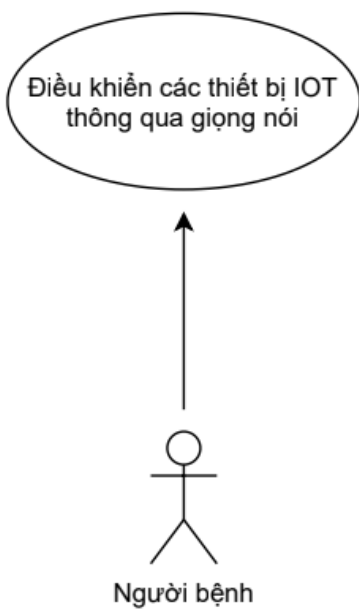
<https://towardsdatascience.com/how-to-get-started-with-google-text-to-speech-using-python-485e43d1d544>

CHƯƠNG 4: XÂY DỰNG VÀ TRIỂN KHAI HỆ THỐNG

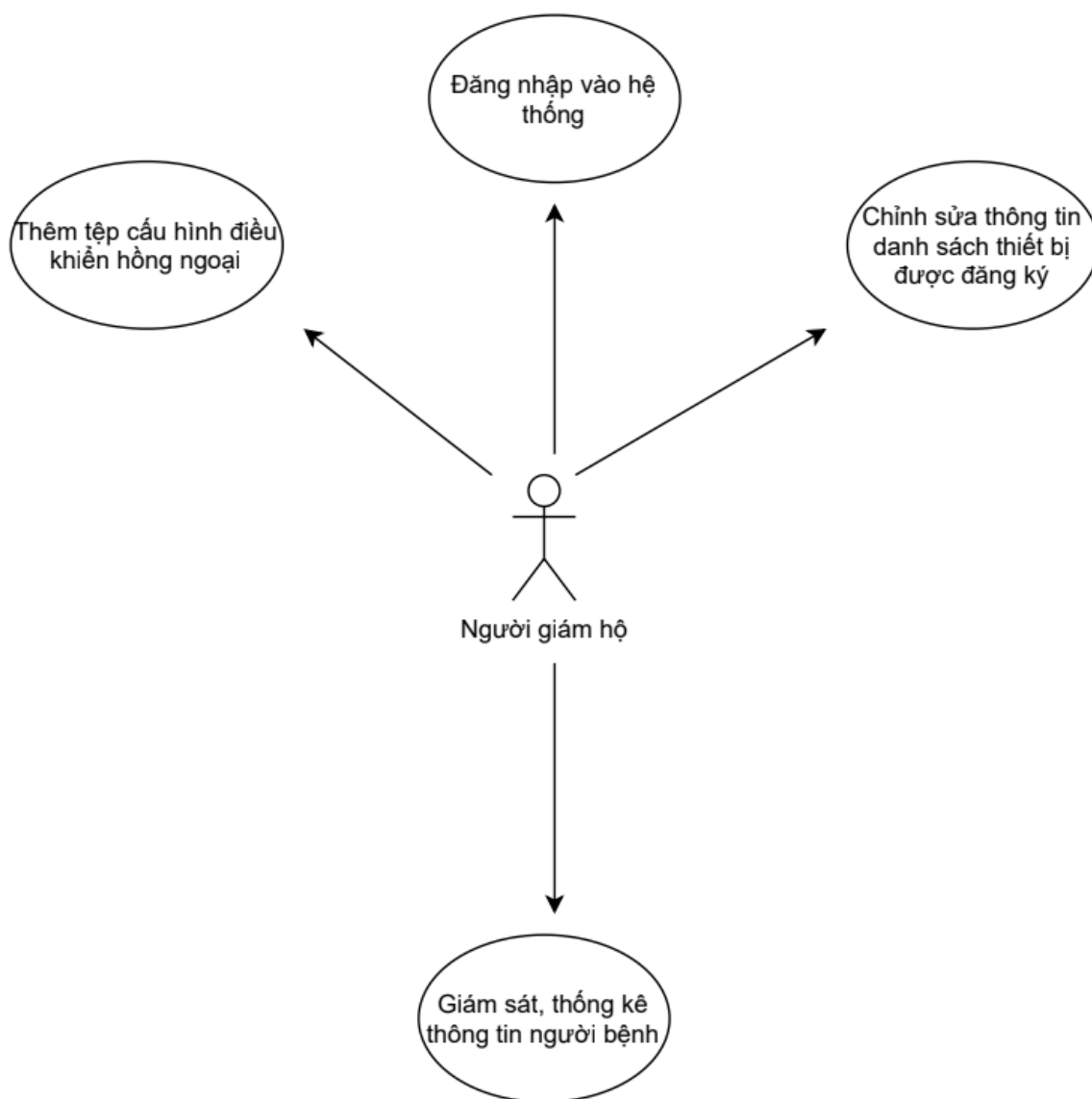
4.1. Mô hình ca sử dụng (Use-Case Model)



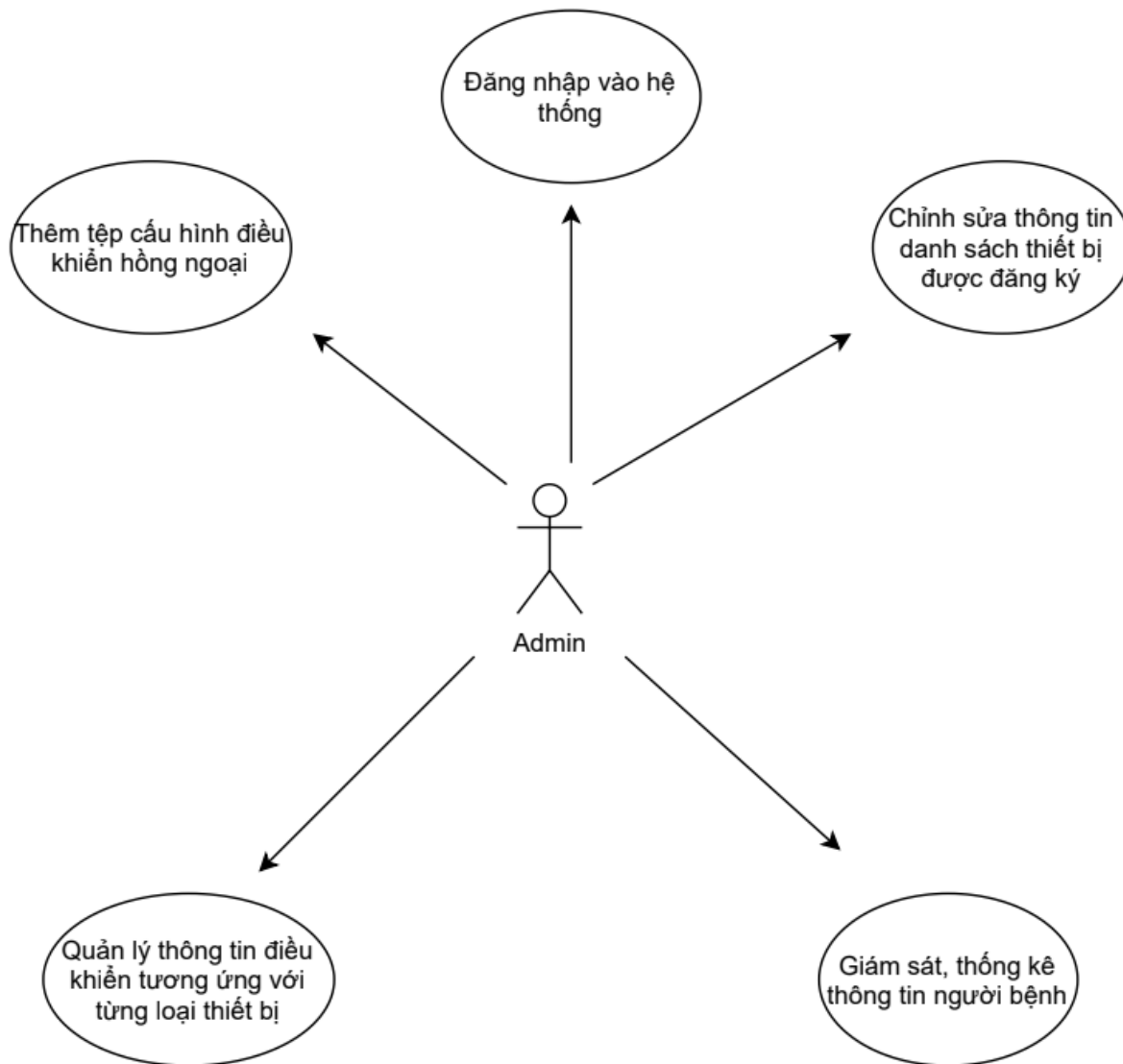
Hình 4-1. Mô hình ca sử dụng – các tác nhân của hệ thống



Hình 4-2. Mô hình ca sử dụng của người bệnh



Hình 4-3. Mô hình ca sử dụng của người giám hộ bệnh nhân



Hình 4-4. Mô hình ca sử dụng của người quản lý

4.1.1. Người bệnh

4.1.1.1. Điều khiển các thiết bị IOT thông qua giọng nói

Mô tả sơ lược

Ca sử dụng này mô tả cách người dùng có thể sử dụng giọng nói của mình để ra lệnh cho hệ thống điều khiển giọng nói thực hiện các lệnh điều khiển đối với những thiết bị đã được thêm vào hệ thống quản lý thiết bị IOT.

Luồng sự kiện

Mục đích: Ca sử dụng này bắt đầu khi người dùng tiến hành chuyển hệ thống từ trạng thái đợi sang trạng thái sẵn sàng bằng một câu lệnh đặt trước. Tiếp đó, người dùng có thể lựa chọn thiết bị muốn điều khiển và lần lượt đưa ra những mệnh lệnh cho hệ thống để điều khiển thiết bị.

Luồng chính:

1. Người dùng khởi động bộ điều khiển trung tâm bằng một câu lệnh đã được định nghĩa trước trong hệ thống, là một câu chào bằng tiếng việt như “chào”, “xin chào” hoặc “hello” bằng tiếng anh; lúc này hệ thống sẽ phản hồi với một lời chào với thời gian tương ứng trong ngày
2. Người dùng dùng tiếp tục lựa chọn thiết bị cần điều khiển bằng cách gọi tên các thiết bị đó tương ứng với tên đã được đăng ký trên hệ thống
3. Sau khi đã nhận được phản hồi thiết bị đã được chọn, người dùng có thể liên tục đưa ra các câu lệnh để điều khiển thiết bị đó.
4. Sau khi hoàn thành điều khiển thiết bị, nếu muốn chuyển sang thiết bị khác thì có thể tiếp tục lại bước 2 và 3.
5. Sau khi đã hết các mong muốn điều khiển, chuyển trạng thái của bộ điều khiển trung tâm về trạng thái đợi bằng một lời chào tạm biệt “bye”, “tạm biệt”, “kết thúc”

Luồng thay thế:

Không có.

Yêu cầu đặc biệt

Không.

Tiền điều kiện

Thiết bị phải được kết nối với internet.

Hậu điều kiện

Không.

Điểm mở rộng

Không.

4.1.2. Người giám hộ

4.1.2.1. Đăng nhập vào hệ thống

Mô tả sơ lược

Ca sử dụng này mô tả cách để người dùng có thể đăng nhập tài khoản vào hệ thống đăng ký thiết bị IOT.

Luồng sự kiện

Mục đích: Ca sử dụng này bắt đầu khi người dùng tiến hành đăng nhập tài khoản được cấp sẵn vào hệ thống.

Luồng chính:

1. Người sử dụng tiến vào trang đăng nhập của hệ thống tại: localhost:3000
2. Người dùng tiến hành nhập tên tài khoản và mật khẩu đã được cung cấp sẵn từ người quản lý.
3. Hệ thống sẽ xác thực người dùng bằng cách kiểm tra tài khoản và mật khẩu có đúng với tài khoản và mật khẩu được lưu trong hệ thống không và kiểm tra quyền của tài khoản được sử dụng những tài nguyên nào trên hệ thống quản lý thiết bị. Nếu xác thực thành công, chuyển vào màn hình hệ thống quản lý thiết bị. Nếu xác thực thất bại, thông báo đăng sai tên tài khoản hoặc mật khẩu, giữ nguyên trạng thái tại trang đăng nhập.

Luồng thay thế:

Không có.

Yêu cầu đặc biệt

Không.

Tiền điều kiện

Thiết bị truy cập vào trang quản lý thiết bị IOT cần được kết nối chung mạng với hệ thống quản lý và người dùng là người giám hộ hoặc bác sĩ của người bệnh.

Hậu điều kiện

Nếu ca sử dụng thành công, người dùng sẽ được đăng nhập vào hệ thống quản lý thiết bị ở màn hình chính. Nếu không, người dùng vẫn sẽ dừng lại ở trang đăng nhập vào hệ thống.

Điểm mở rộng

Không.

4.1.2.2. Thêm tệp cấu hình điều khiển hồng ngoại

Mô tả sơ lược

Ca sử dụng này mô tả cách người dùng tạo ra các tệp cấu hình các nút điều khiển thiết bị sử dụng điều khiển bằng sóng hồng ngoại tương ứng với mỗi loại thiết bị được hỗ trợ

Luồng sự kiện

Mục đích: Ca sử dụng mô tả cách thức người dùng thêm mới các tệp điều khiển thiết bị bằng sóng hồng ngoại vào hệ thống quản lý thiết bị

Luồng chính

Trên giao diện thêm tệp cấu hình điều khiển hồng ngoại, người dùng thêm tệp cấu hình theo luồng như sau:

1. Người sử dụng điền tên thiết bị cần lưu cấu hình điều khiển vào ô điền tên. Tên này đại diện cho mã thiết bị được nhà sản xuất điền trên điều khiển của những thiết bị đó hoặc một tên gọi dễ nhớ thể hiện thiết bị.
2. Sau khi đã đặt tên, người dùng lựa chọn loại thiết bị mà mình mong muốn thêm tệp điều khiển. Ví dụ: tivi, điều hòa... Các loại thiết bị này là những thiết bị được hỗ trợ trong hệ thống.
3. Sau khi đã chọn được loại thiết bị, một mảng các phím được hỗ trợ tương ứng với loại thiết bị đó sẽ được hiện thị. Người dùng chọn một nút điều khiển trong mảng. Tại thời điểm này, tất cả các nút sẽ bị vô hiệu hóa, không thể chọn để tránh ảnh hưởng tới việc thu tín hiệu
4. Tiếp đó, sử dụng điều khiển của thiết bị hướng đèn tín hiệu về đầu thu sóng hồng ngoại của hệ thống và tiến hành nhấn phím điều khiển tương ứng với nút đã chọn ở bước trên hai lần. Nếu thành công, hệ thống sẽ tái kích hoạt các nút thu tín hiệu khác, nếu không, người dùng chọn lại nút để tiếp tục thu lại tín hiệu.
5. Lần lượt lặp lại các bước ba và bốn cho đến khi thu hết tín hiệu các nút điều khiển.
6. Nhấn “lưu” để hoàn thành luồng sự kiện.

Luồng phụ

Không có

Yêu cầu đặc biệt

Không có

Tiền điều kiện

Người dùng đã đăng nhập vào hệ thống quản lý thiết bị IOT và khi bắt đầu thu tín hiệu, nên để xa các thiết bị có phát tín hiệu khác như wifi hoặc sóng hồng ngoại để tránh mất thu tín hiệu của hệ thống nhận nhầm hoặc thừa tín hiệu.

Hậu điều kiện

Một tệp cấu hình sẽ được sinh ra và chuyển tới thư mục gốc phục vụ cho thư viện LIRC có thể tìm và gửi tín hiệu khi cần thiết.

Điểm mở rộng

Không.

4.1.2.3. Chỉnh sửa thông tin danh sách thiết bị được đăng ký

Mô tả sơ lược

Ca sử dụng này mô tả cách người dùng thực hiện thay đổi thông tin về các thiết bị thông minh trong hệ thống

Luồng sự kiện

Mục đích: Ca sử dụng này bắt đầu khi người dùng tiến hành thêm, sửa, xóa thông tin đăng ký các thiết bị điều khiển trên hệ thống.

Luồng chính:

1. Người dùng sẽ bắt đầu tại trang “thiết bị” của hệ thống. Tại đây, người dùng có thể thấy một bảng thể hiện các thiết bị được đăng ký trong hệ thống. Bảng này bao gồm các thông tin chi tiết về những thiết bị đó.
2. Để xóa thiết bị, người dùng chọn nút xóa ở phía cuối bảng. Khi này, một thông báo hiện lên để xác nhận hành động, chọn “Có” để xác nhận hoặc “Không” để hủy bỏ hành động. Nếu xóa thành công, một thông báo sẽ hiện lên cùng kết quả tương ứng.

3. Để thêm mới thiết bị, người dùng chọn nút “thêm” trên trang thống kê thiết bị, tại đây, một bảng thêm thông tin thiết bị sẽ hiện ra.

4. Người dùng lần lượt điền các thông tin cần thiết có trong bảng. Các loại thiết bị sẽ có các thông tin khác nhau, tùy vào loại thiết bị người dùng muốn đăng ký sẽ điền thông tin cần thiết cho thiết bị đó. Đặc biệt đối với trường mô tả, đây là trường không bắt buộc, tuy nhiên nên được điền đủ thông tin để khi người bệnh muốn liệt kê thiết bị, họ có thể biết rõ được thiết bị này tương ứng với đồ dùng nào trong nhà của mình.

5. Sau khi đã điền đủ thông tin, nhấn chọn “lưu” để thực hiện lưu thiết bị vào hệ thống hoặc “hủy” để hủy bỏ việc lưu thông tin vào hệ thống. Nếu lưu thành công, một thông báo sẽ hiện lên với nội dung tương ứng và bảng thêm thiết bị sẽ được đóng lại.

Luồng phụ:

Không có

Yêu cầu đặc biệt

Không có

Tiền điều kiện

Người dùng cần đăng nhập vào hệ thống trước khi thực hiện luồng này.

Hậu điều kiện

Mỗi thay đổi đối với dữ liệu trong hệ thống sẽ làm bảng thống kê thiết bị thực hiện thay đổi tương ứng.

Điểm mở rộng

Không có

4.1.2.4. Giám sát thống kê thông tin người bệnh

Mô tả sơ lược

Ca sử dụng này mô tả các người dùng giám sát thông tin người bệnh qua những biểu đồ, thống kê được hệ thống đưa ra. Những thông tin này bao gồm thông tin về môi trường, người bệnh thông qua những cảm biến đi cùng người bệnh, tiếp theo là những thông tin về thói quen sử dụng thiết bị, khung giờ sử dụng thiết bị thông qua những lịch sử điều khiển thiết bị bằng hệ thống.

Luồng sự kiện

Mục đích: Ca sử dụng này bắt đầu khi người dùng lựa chọn mục “thông số dữ liệu” trên hệ thống.

Luồng chính:

1. Người dùng lựa chọn mục “thông số dữ liệu” trên hệ thống.
2. Một màn hình hiện ra với hai ô chọn. Người dùng chọn thiết bị muốn xem thống kê thông số ở ô lựa chọn thiết bị phía góc trái trên của màn hình.
3. Sau khi thiết bị được chọn, một dãy biểu đồ tương ứng với loại thiết bị được chọn sẽ hiện lên cho người dùng thấy được các thông số của thiết bị liên quan đến người bệnh. Đối với các thiết bị cảm biến, một bảng timeline sẽ được đưa ra thể hiện thông số của cảm biến theo thời gian; đối với các thiết bị điều khiển, các bảng dữ liệu thống kê như số lần sử dụng nút điều khiển, khoảng thời gian sử dụng nút điều khiển...
4. Sau đó, người dùng có thể chọn ngày tháng ở ô chọn ngày tháng phía góc trên phải của màn hình để thống kê cụ thể hơn thông tin trong khoảng thời gian mong muốn.

Luồng phụ:

Không có

Yêu cầu đặc biệt

Không có

Tiền điều kiện

Người dùng đã đăng nhập thành công vào hệ thống.

Hậu điều kiện

Không có

Điểm mở rộng

Không có

4.1.3. Người quản lý (admin)

Người quản lý có đầy đủ mọi chức năng của người dùng khác và có thêm chức năng sau đây:

4.1.3.1. Quản lý thông tin điều khiển tương ứng với từng loại thiết bị

Mô tả sơ lược

Ca sử dụng này mô tả cách người dùng thực hiện thay đổi các phím điều khiển tương ứng cho từng thiết bị được hỗ trợ bởi hệ thống.

Luồng sự kiện

Mục đích: Ca sử dụng này bắt đầu khi người dùng tiến hành thêm mới hoặc xóa bỏ các nút điều khiển vào hệ thống hoặc đặt lại dữ liệu mặc định cho hệ thống

Luồng chính:

1. Người dùng sẽ bắt đầu tại trang “thiết bị” của hệ thống. Tại đây, người dùng có thể thấy một bảng thể hiện các thiết bị được đăng ký trong hệ thống. Bảng này bao gồm các thông tin chi tiết về những thiết bị đó.
2. Để xóa thiết bị, người dùng chọn nút xóa ở phía cuối bảng. Khi này, một thông báo hiện lên để xác nhận hành động, chọn “Có” để xác nhận hoặc “Không” để hủy bỏ hành động. Nếu xóa thành công, một thông báo sẽ hiện lên cùng kết quả tương ứng.
3. Để thêm mới thiết bị, người dùng chọn nút “thêm” trên trang thống kê thiết bị, tại đây, một bảng thêm thông tin thiết bị sẽ hiện ra.
4. Người dùng lần lượt điền các thông tin cần thiết có trong bảng. Các loại thiết bị sẽ có các thông tin khác nhau, tùy vào loại thiết bị người dùng muốn đăng ký sẽ điền thông tin cần thiết cho thiết bị đó. Đặc biệt đối với trường mô tả, đây là trường không bắt buộc, tuy nhiên nên được điền đủ thông tin để khi người bệnh muốn liệt kê thiết bị, họ có thể biết rõ được thiết bị này tương ứng với đồ dùng nào trong nhà của mình.
5. Sau khi đã điền đủ thông tin, nhấn chọn “lưu” để thực hiện lưu thiết bị vào hệ thống hoặc “hủy” để hủy bỏ việc lưu thông tin vào hệ thống. Nếu lưu thành công, một thông báo sẽ hiện lên với nội dung tương ứng và bảng thêm thiết bị sẽ được đóng lại

Luồng phụ:

Không có

Yêu cầu đặc biệt

Không có

Tiền điều kiện

Người dùng cần đăng nhập vào hệ thống trước khi thực hiện luồng này.

Hậu điều kiện

Mỗi thay đổi đối với dữ liệu trong hệ thống sẽ làm bảng thống kê thiết bị thực hiện thay đổi tương ứng.

Điểm mở rộng

Không có

4.1.3.2. Giám sát thống kê thông tin người bệnh

Mô tả sơ lược

Ca sử dụng này mô tả các người dùng giám sát thông tin người bệnh qua những biểu đồ, thống kê được hệ thống đưa ra. Những thông tin này bao gồm thông tin về môi trường, người bệnh thông qua những cảm biến đi cùng người bệnh, tiếp theo là những thông tin về thói quen sử dụng thiết bị, khung giờ sử dụng thiết bị thông qua những lịch sử điều khiển thiết bị bằng hệ thống.

Luồng sự kiện

Mục đích: Ca sử dụng này bắt đầu khi người dùng lựa chọn “Nút điều khiển IR” trên hệ thống

Luồng chính:

1. Người dùng lựa chọn mục “Nút điều khiển IR” trên hệ thống. Tại đây, một màn hình xuất hiện với hai mục chính bao gồm một vùng thêm tệp thể hiện các nút điều khiển được hỗ trợ, mục thứ hai là nút đặt lại cài đặt cũ của hệ thống
2. Người dùng lựa chọn một trong hai mục trên để thực hiện thay đổi với hệ thống. Đối với mục thêm tệp vào hệ thống, người dùng sẽ phải cung cấp một tệp có các dữ liệu được sắp xếp theo quy định trước. Khi thêm tệp xong, các thay đổi sẽ được thêm vào hệ thống và một thông báo sẽ hiện lên thể hiện thay đổi đã thành công. Đối với mục đặt lại thông tin, người dùng chỉ cần nhấn nút đã được hiển thị trên màn hình, mọi thay đổi sẽ được thực hiện và thông báo sẽ được gửi đến sau khi thay đổi thành công.

Luồng phụ:

Không có

Yêu cầu đặc biệt

Không có

Tiền điều kiện

Người dùng đã đăng nhập thành công vào hệ thống.

Hậu điều kiện

Các phím khi người dùng thực hiện ca sử dụng “Thêm tệp cấu hình điều khiển hồng ngoại” sẽ thay đổi dựa theo những thay đổi vừa cập nhật

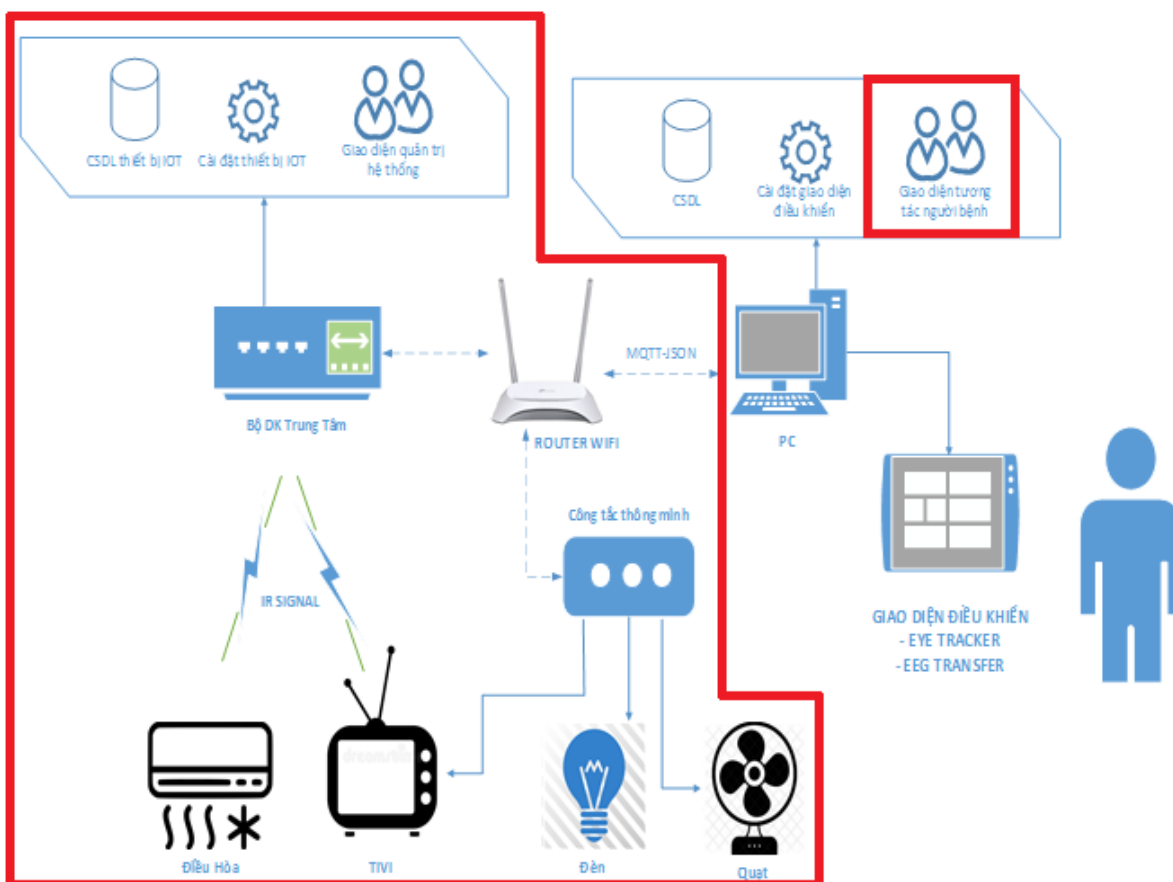
Điểm mở rộng

Không có

4.2. Phân tích hệ thống

Mục phân tích hệ thống sẽ bao gồm ba phần chính sau đây: i. tổng quan kiến trúc hệ thống, ii. danh sách key abstractions, iii. sơ đồ luồng dữ liệu trong hệ thống

4.2.1. Tổng quan kiến trúc hệ thống

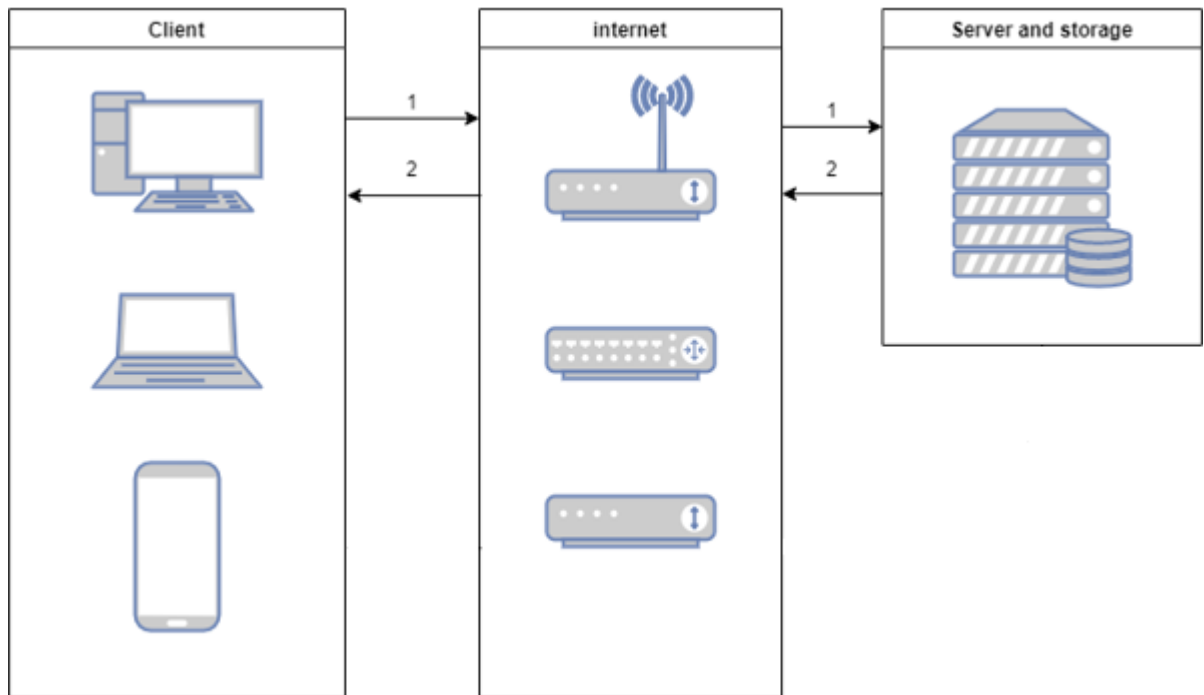


Hình 4-5. Kiến trúc hệ thống

Trong hệ thống lớn hiện tại của nhóm em, các phần được trình bày trong khóa luận sẽ bao gồm các phần trong khung khoanh đỏ ở hình trên và được chia làm 3 module chính. Module thứ nhất được xây dựng trên mô hình ứng dụng web được sử dụng rộng rãi là client-server bao gồm cơ sở dữ liệu, server và trang web. Đây là module cung cấp các chức năng quản lý thiết bị, quản lý các tệp điều khiển sóng hồng ngoại, thống kê thông tin người bệnh. Module này tương ứng với khung bao gồm: CSDL thiết bị IOT, cài đặt thiết bị IOT, giao diện quản trị hệ thống nằm phía trên trái của hình 4-5. Module thứ hai là module chuyển đổi lệnh giọng nói tương ứng với giao diện tương tác người bệnh nằm riêng lẻ bên phải của hình 4-5. Đây là module cung cấp chức năng chuyển đổi giữa lệnh giọng nói sang thành lệnh máy để module thứ ba có thể hiểu được và xử lý tương ứng. Module thứ ba là module điều khiển các thiết bị được đăng ký trong cơ sở dữ liệu của module thứ nhất. Đây là module cung cấp chức năng điều khiển các thiết bị tương ứng

sau khi nhận dữ liệu điều khiển từ module thứ hai. Module này tương ứng với các phần còn lại trong khung đỏ bên trái hình 4-5.

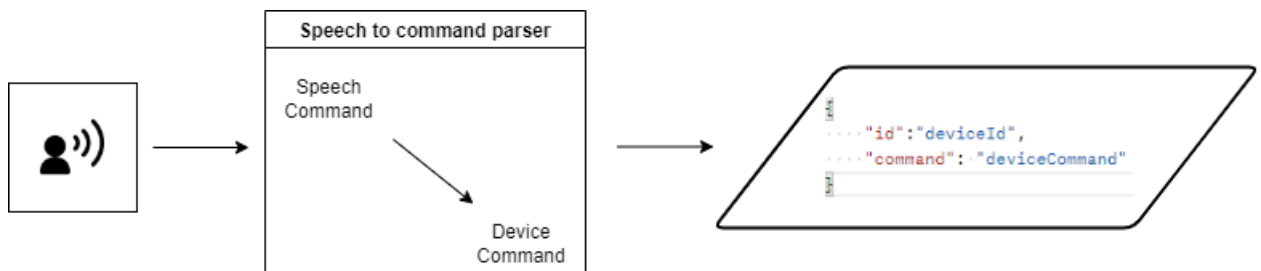
4.2.1.1. Module quản lý thiết bị IOT



Hình 4-6. Module quản lý thiết bị IOT

Với module này, luồng dữ liệu được đánh số và có ý nghĩa như sau: client là các máy tính, laptop, điện thoại thông minh hoặc các máy tính bảng được kết nối cùng mạng với module để có thể gửi request cho server thông qua giao diện người dùng (1). Sau khi server nhận được request, server sẽ thực hiện tra cứu, chuẩn bị dữ liệu cần thiết, tương ứng với request được gửi lên theo các chức năng đã được quy định từ trước, sau đó gửi lại response cho client để hiển thị kết quả của request (2)

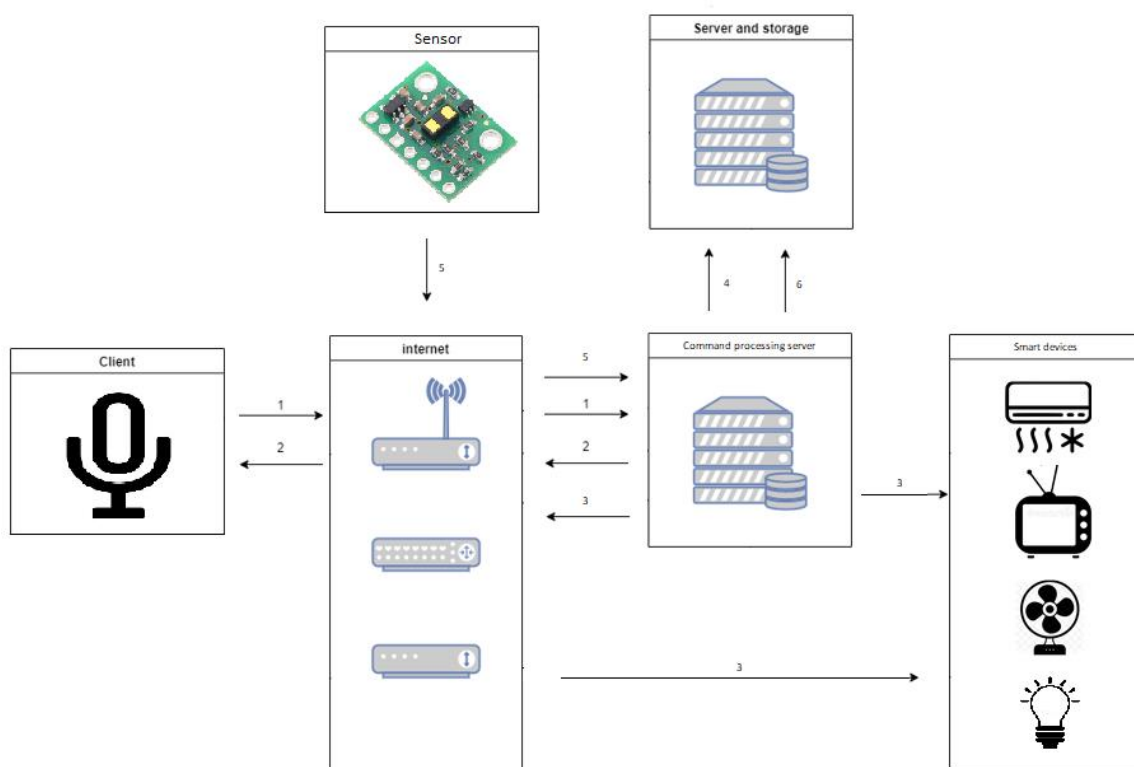
4.2.1.2. Module chuyển đổi lệnh giọng nói



Hình 4-7. Module chuyển đổi lệnh giọng nói

Với module chuyên đổi lệnh giọng nói, dữ liệu sẽ di chuyển theo luồng như sau: Module nhận lệnh điều khiển thiết bị thông qua việc người bệnh ra lệnh bằng giọng nói. Lệnh điều khiển bằng giọng nói được lưu lại cùng với thông tin thiết bị mong muốn điều khiển. Sau khi đã có được thông tin về câu lệnh điều khiển bằng giọng nói, module tiến hành tham chiếu câu lệnh đó với lệnh máy được định nghĩa sẵn trong hệ thống và đưa ra mẫu dữ liệu điều khiển. Dữ liệu này sau đó sẽ được gửi tới module điều khiển thiết bị thông minh thông bằng cách tận dụng lợi thế nhanh, nhẹ của giao thức MQTT để sử dụng cho việc điều khiển thiết bị.

4.2.1.3. Module điều khiển thiết bị thông minh



Hình 4-8. Module điều khiển thiết bị thông minh

Với module điều khiển thiết bị thông minh, luồng dữ liệu sẽ được chia làm hai luồng. Luồng thứ nhất là bắt đầu với thông tin điều khiển nhận được từ người dùng, luồng thứ hai bắt đầu với dữ liệu thu thập được từ các thiết bị cảm biến ngoại vi (sensors).

Với luồng dữ liệu điều khiển thiết bị, dữ liệu sẽ di chuyển theo luồng như sau: Module nhận được tín hiệu điều khiển từ phía module chuyển đổi lệnh giọng nói thông qua internet và chuyển tới server xử lý dữ liệu (1). Tín hiệu điều khiển sau khi đã được gửi tới server xử lý dữ liệu sẽ được kiểm tra đúng chuẩn và gửi thông báo cho người dùng (2). Tiếp đến, tín hiệu sẽ được xử lý từ dạng mẫu từ người dùng gửi lên thành tín hiệu điều khiển dạng nguyên thủy của từng thiết bị và gửi đến thiết bị tương ứng thông qua internet hoặc trực tiếp thông qua sóng hồng ngoại (3). Sau khi đã gửi tín hiệu điều khiển dạng nguyên thủy cho các thiết bị, server gửi request lưu thông tin điều khiển của người dùng lên server của module đầu tiên để tiến hành thu thập dữ liệu người dùng.

Với luồng dữ liệu thu thập từ thiết bị ngoại vi, dữ liệu sẽ di chuyển theo luồng sau đây: Dữ liệu được thu thập từ các sensor sẽ được gửi qua internet với topic tương ứng đã được đăng ký và chuyển tới server xử lý dữ liệu (5). Tại đây, server sẽ thực hiện lọc ra thiết bị tương ứng với topic được gửi lên, sau đó tạo ra dữ liệu có thể lưu vào hệ thống và tiến hành gửi request đến server của module quản lý thiết bị IOT để thực hiện lưu dữ liệu thu thập (6).

4.2.2. Phân tích kiến trúc – Key abstraction

STT	Key abstractions	Definition
1	Device	Đại diện cho các thiết bị thông minh cần được thêm vào hệ thống. Mỗi thiết bị sẽ có thông tin, tên, loại riêng tùy thuộc vào thiết bị sẽ được đăng ký
2	Device Command	Đại diện cho những câu lệnh điều khiển thiết bị được hỗ trợ bởi hệ thống. Mỗi câu lệnh sẽ có một mã điều khiển khác nhau và câu lệnh điều khiển khác nhau.
3	Device Data	Đại diện cho dữ liệu thu thập được từ người dùng. Mỗi dữ liệu sẽ phân biệt theo thời gian thu thập và loại dữ liệu
4	Supported Device	Đại diện cho các mã thiết bị điều khiển bằng sóng hồng ngoại được hệ thống hỗ trợ.

Bảng 1. Danh sách Key abstraction của hệ thống

Bảng key abstraction bên trên thể hiện, mỗi một abstraction được coi là một thực thể hay một bảng được lưu trong cơ sở dữ liệu. Mỗi quan hệ giữa những key abstraction sẽ được thể hiện trên phương diện mỗi quan hệ giữa các bảng trong cơ sở dữ liệu với nhau.

4.3. Thiết kế hệ thống phía máy chủ

Sau khi đã hoàn thành phân tích hệ thống, chúng ta sẽ chuyển sang thiết kế hệ thống. Tại đây, chúng ta sẽ chia thành ba phần chính như sau: i. Xác định thực thể và mối quan hệ giữa chúng, ii. Mô hình thực thể, các mối quan hệ, iii. Thiết kế cơ sở dữ liệu từ các bước trên.

4.3.1. Xác định các thực thể và mối quan hệ giữa chúng

Dựa theo các key abstraction đã được liệt kê phía trên, ta sẽ xác định ra các thực thể cần sử dụng trong hệ thống là:

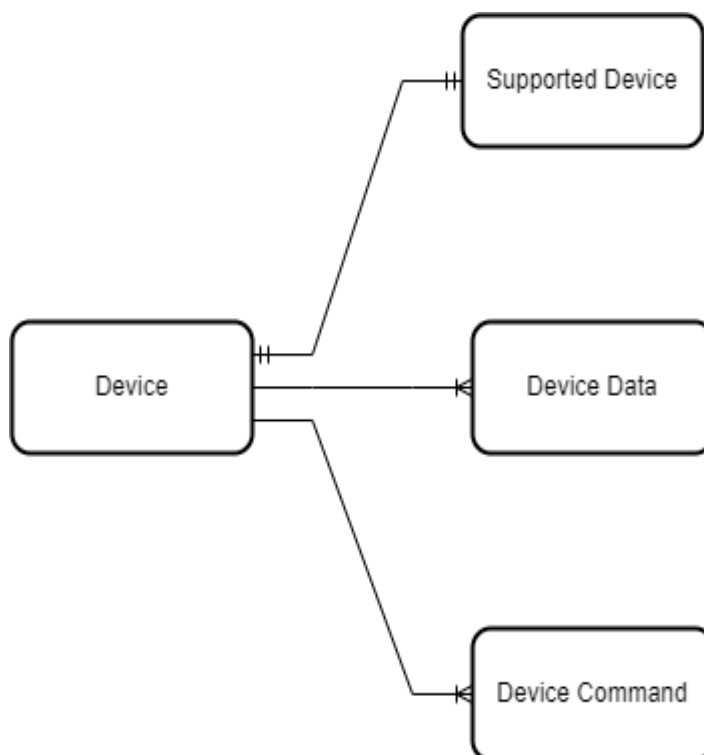
1. Device: id, name, model, product type, control topic, output topic, gangs, created date, description, status
2. Device Command: id, type, command, phrase command
3. Device Data: id, data type, data, device id, created date
4. Supported device: id, name

Mối quan hệ giữa các thực thể với nhau được miêu tả như sau:

1. Mỗi một thiết bị cụ thể sẽ có những tín hiệu điều khiển khác nhau và mỗi thiết bị sẽ có nhiều câu lệnh điều khiển, nên đây là mối quan hệ một nhiều (1, n)
2. Mỗi một thiết bị tạo ra dữ liệu khác nhau tùy thuộc vào thông tin mà thiết bị ấy sử dụng và mỗi thiết bị sẽ có nhiều lần điều khiển hoặc thu thập thông tin, cho nên đây là mối quan hệ (1, n)
3. Mỗi một thiết bị chỉ có duy nhất một mã thiết bị được nhà phát hành cung cấp, cho nên đây là mối quan hệ (1, 1)

4.3.2. Mô hình thực thể các mối quan hệ (ERD)

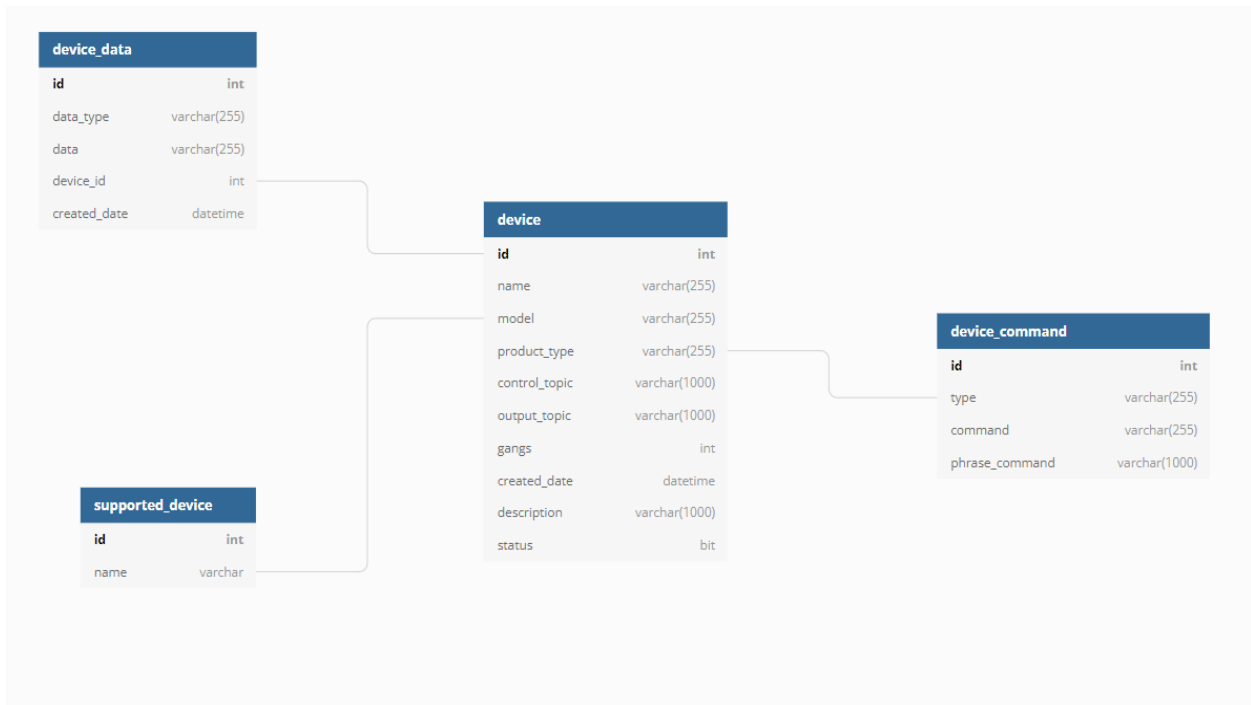
Dưới đây là mô hình thực thể mối quan hệ được sinh ra từ các thực thể và mối quan hệ ở phía trên.



Hình 4-9. Mô hình thực thể mối quan hệ

4.3.3. Thiết kế cơ sở dữ liệu

Dựa theo các key abstraction và các mô tả mối quan hệ thực thể, có thể thiết kế được mô hình cơ sở dữ liệu như sau:



Hình 4-10. Thiết kế cơ sở dữ liệu

4.4. Xây dựng hệ thống phía máy chủ

Nhằm đảm bảo tính đơn giản, dễ hiểu của hệ thống cho người mới nhưng vẫn có thể thích ứng với những thay đổi, có khả năng mở rộng tốt và vẫn có thể thực thi những tác vụ phức tạp, em sẽ sử dụng một kiểu kiến trúc đang được sử dụng rộng rãi và đáp ứng đầy đủ những yêu cầu được đặt ra của hệ thống. Ở đây, em sẽ sử dụng kiến trúc ba lớp¹⁰[10]. Với kiến trúc ba lớp này, hệ thống sẽ được chia thành ba lớp chính là: lớp controller, lớp service và lớp repository. Ngoài ba lớp chính này ra, em sẽ sử dụng thêm lớp entity để thể hiện các thực thể được lưu trong hệ thống và các dữ liệu cần thiết. Đối với ba lớp chính, mỗi lớp sẽ có định nghĩa và chức năng sau đây. Lớp repository là lớp có chức năng thao tác trực tiếp với dữ liệu trong cơ sở dữ liệu. Lớp này chứa những hàm chức năng như thêm mới dữ liệu, sửa dữ liệu đã có trong hệ thống hay thực hiện truy vấn để lấy ra dữ liệu cần thiết cho request. Lớp service là lớp xử lý logic. Đây là lớp đảm nhiệm vai trò xử lý nghiệp vụ của hệ thống. Lớp sẽ thực hiện lấy dữ liệu từ lớp repository và tiến vào quá trình xử lý nghiệp vụ. Tùy theo nghiệp vụ mà dữ liệu sẽ được xử lý theo logic riêng và tiến hành đưa response đến lớp tiếp theo. Cuối cùng, lớp controller, đây là lớp có các chức

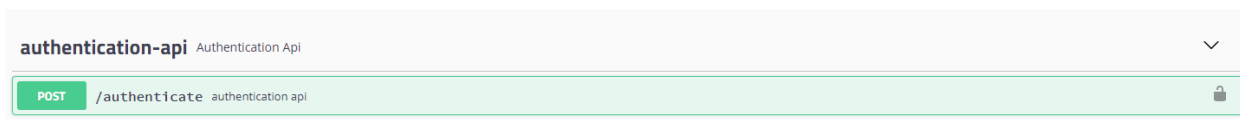
¹⁰ "Spring Restful và mô hình 3 lớp – kipalog.com",
<https://kipalog.com/posts/Spring-Boot--Spring-Restful-va-mo-hinh-3-lop>

năng xử lý, thao tác với request nhận về từ phía client và trả ra các response cần thiết và tương ứng đối với các request đã nhận.

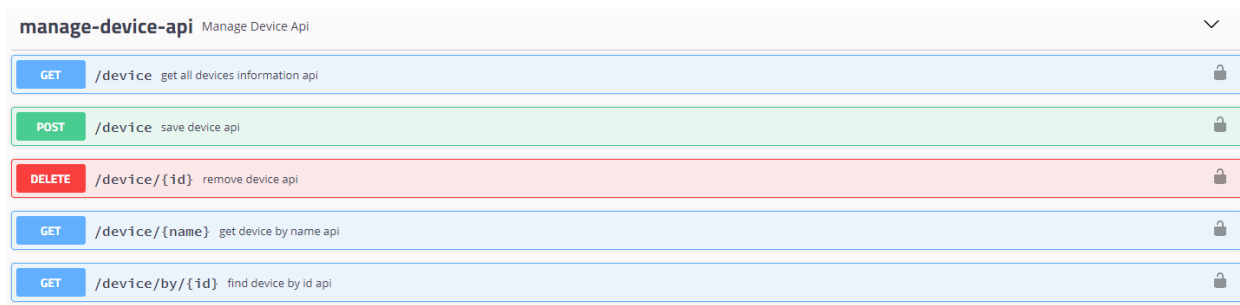
Mục xây dựng hệ thống phía máy chủ (backend) sẽ trình bày hai nội dung chính dựa trên những phần đã hoàn thành bao gồm: các đầu APIs, xác thực người dùng với JWT

4.4.1. Các đầu APIs

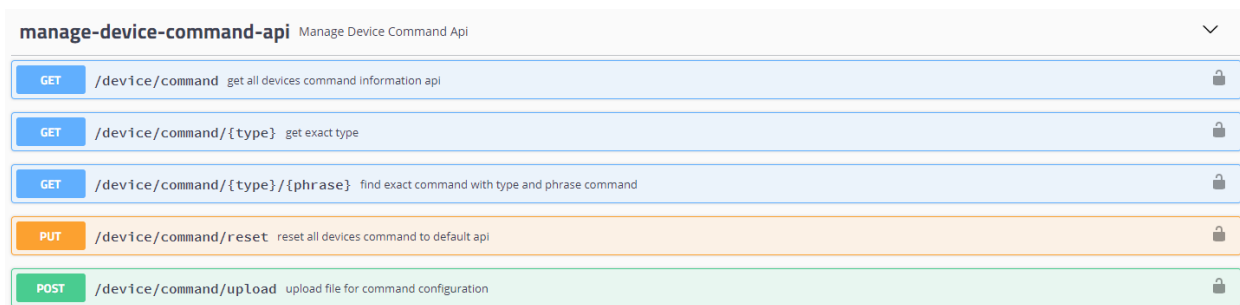
Các đầu api tính đến hiện tại của hệ thống sẽ được trình bày dưới đây. Em trực tiếp tham gia vào thiết kế và phát triển những api này.



Hình 4-11. Api xác thực tài khoản



Hình 4-12. Api quản lý thiết bị



Hình 4-13. Api quản lý thông tin nút điều khiển hồng ngoại

manage-device-data-api Manage Device Data Api			▼
GET	/device/data	get all devices data api	🔒
POST	/device/data	save device data	🔒
POST	/device/data/count	get device data count by device id	🔒
POST	/device/data/data	get device data by device id	🔒

Hình 4-14. Api quản lý dữ liệu thu thập

manage-lirc-config-api Manage Lirc Config Api			▼
DELETE	/lirc	init command line	🔒
POST	/lirc/{btnName}	start recording process	🔒
GET	/lirc/{deviceName}	init recording session	🔒

Hình 4-15. Api thêm mới tệp cấu hình điều khiển hồng ngoại

manage-supported-device-api Manage Supported Device Api			▼
GET	/device/supported/fetch	fetch supported devices information api	🔒
GET	/device/supported/get/all	get all supported devices information api	🔒

Hình 4-16. Api quản lý mẫu model thiết bị được hỗ trợ

4.4.2. Xác thực người dùng với JWT

Bảo mật – một vấn đề luôn được đặt trong mối quan tâm hàng đầu ở bất kỳ khía cạnh nào trong cuộc sống, đặc biệt với thời đại công nghệ phát triển, thời đại của công nghệ thông tin. Bảo mật còn là một trong những vấn đề quan trọng nhất đối với bất kỳ một nhà phát triển nào trong quá trình xây dựng ứng dụng. Mới đây, một lỗ hổng được cho là lớn nhất thập kỉ đã được tìm ra trong một thư viện của ngôn ngữ Java¹¹[11], được đánh giá là có thể gây ảnh hưởng rất lớn tới các hệ thống công nghệ thông tin. Tỷ lệ các cuộc tấn công này ở nhiều khu vực ở trên mức 40% như châu Âu với 42.2% hay châu Mỹ với 41.8%. Có thể thấy tổn thất sẽ rất nghiêm trọng nếu thông tin không được bảo mật đúng cách. Đối với hệ thống này, mặc dù những dữ liệu có thể không phải đạt đến mức độ nghiêm trọng tiêu tốn rất nhiều tiền và công sức, tuy nhiên, bảo mật vẫn rất cần thiết để đảm bảo thông tin cá nhân của người bệnh không bị lộ hoặc tránh bị kẻ xấu thay đổi thiết đặt làm cho người dùng khó khăn hoặc không thể sử dụng hệ thống. Trong phần này, em

¹¹ “Lỗ hổng log4j của java và mức độ ảnh hưởng – vnexpress.net”
<https://vnexpress.net/lo-hong-lon-nhat-thap-ky-co-the-anh-huong-nhieu-he-thong-o-viet-nam-4403132.html>

sẽ nêu ra cách xác thực người dùng bằng cách sử dụng JWT (JSON web token) mà em đã áp dụng trong hệ thống.

JWT là một chuẩn mở (RFC 7519) sử dụng như một cách đóng gói dữ liệu với chữ ký hoặc mã hóa tùy chọn để xây dựng những tập tin dữ liệu JSON¹²[12]. Các token được ký sử dụng cặp khóa bí mật/ công khai (private/public key) hoặc những khóa bí mật của một bên. Các token thường được ứng dụng trong đăng nhập một lần (SSO) và rất nhỏ gọn.

Đối với hệ thống này, JWT sẽ được sử dụng như một chiếc chìa khóa hoặc một tấm vé để cung cấp cho người dùng khả năng sử dụng tài nguyên của hệ thống. Sau khi người dùng tiến hành đăng nhập vào hệ thống, phía máy chủ sẽ kiểm tra tài khoản và mật khẩu được gửi lên. Nếu đã xác thực tài khoản hợp lệ, hệ thống sẽ sinh ra access token và trả về cho người dùng. Đối với người dùng hệ thống điều khiển thiết bị IOT, thời hạn của access token sẽ dài hơn rất nhiều so với thời hạn sử dụng của người dùng trên giao diện web. Lý do chính đó là vì người dùng điều khiển thiết bị IOT có ca làm việc dài hơn so với người dùng giao diện web, có thể dài đến vài tiếng đồng hồ tùy theo thời gian sử dụng của mỗi người.

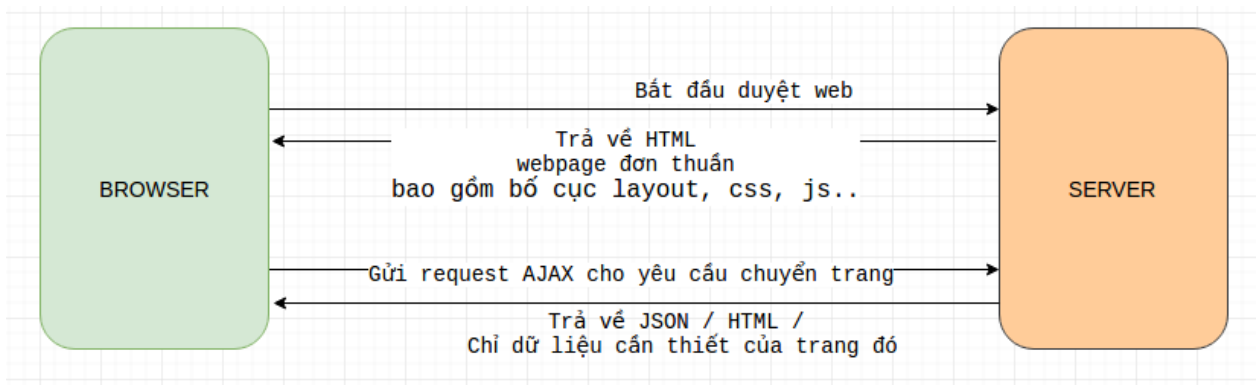
Đối với giao diện web, khi người dùng đăng nhập thành công sẽ nhận được một access token được gửi đến từ phía server. Hệ thống sẽ thực hiện lưu access token này vào localStorage để sử dụng cho các request về sau. Vì phía backend được bảo mật đối với từng đầu api nên phía client luôn luôn phải xác minh danh tính người dùng để có thể gọi được những api này. Để làm được việc đó, client phải thêm một trường “authorization” vào phần header của mỗi request để server có thể xác thực người dùng mỗi khi có request được gửi đến. Khi access token đã hết hạn, các request sẽ bị chặn lại bởi phía server và lúc này, người dùng sẽ phải thực hiện lại phần đăng nhập vào hệ thống. Trong một trường hợp khác, khi người dùng đóng trình duyệt hoặc tắt trang web, khi đó, trình duyệt sẽ không còn lưu lại access token ở trong localStorage nữa, khi này, người dùng cũng sẽ phải thực hiện đăng nhập lại vào hệ thống.

¹² “JWT là gì? – jwt.io”
<https://jwt.io/introduction>

4.5. Xây dựng hệ thống web phía người dùng admin

Việc xây dựng một hệ thống hoạt động tốt không chỉ nằm ở việc tối ưu các khối code, tăng thời gian uptime của hệ thống mà còn ở việc xây dựng kiến trúc nền tảng vững chắc, dễ bảo trì, dễ hiểu. ReactJS là thư viện rất đáng tin cậy và hoàn toàn phù hợp đối với hệ thống. Chính vì vậy, ReactJS đã được em lựa chọn làm công cụ xây dựng nên hệ thống phần frontend giao diện người dùng. So sánh với các framework có mặt trên thị trường hiện tại như VueJS, NextJS hay AngularJS, ReactJS có một hệ thống đồ sộ các component được thiết kế rất ổn định, thuộc rất nhiều các thể loại khác nhau và hơn nữa, những component này còn có thể sử dụng lại ở rất nhiều nơi. Hơn thế nữa, ReactJS còn mang lại trải nghiệm rất tốt khi đem tới tốc độ xử lý không kém gì so với những framework đã kể trên. Không những vậy, ReactJS còn cung cấp khả năng về quản lý state, đây là chức năng rất quan trọng đối với bất cứ ứng dụng web nào. Việc quản lý tốt state của một trang web đưa ra cho chúng ta cái nhìn tổng thể hơn về luồng dữ liệu đi qua trong hệ thống cũng như cung cấp khả năng gỡ lỗi, xử lý sự cố dễ dàng hơn.

SPA (Single-page App) là đang là xu hướng nổi bật giúp tăng cao trải nghiệm người dùng. Trong hệ thống, SPA được lựa chọn để xây dựng nên giao diện người dùng do cách phát triển này làm đơn giản hóa quá trình xây dựng nên trang web, hơn nữa, nó còn giảm tải đi những lượt request đến từ phía người dùng so với cách thức phát triển web thông thường. Đối với các trang web kiểu cũ, toàn bộ trang sẽ được tải lên mỗi khi người dùng truy cập vào thành phần bất kì trong trang đó. Ngược lại, đối với SPA, những trang này chỉ thay đổi những thành phần mà người dùng lựa chọn chứ hoàn toàn không ảnh hưởng tới các phần còn lại của hệ thống. SPA chỉ tập trung xử lý ở phần client, tách riêng frontend và backend, giảm tải cho backend khi giờ đây backend không còn cần phải xây dựng các mẫu html để gửi lên cho frontend. Frontend sẽ đảm nhiệm phần tiếp nhận thông tin request từ người dùng cùng những tính năng hay dữ liệu cần thiết, sau đó thực hiện gửi request đến cho backend. Backend khi này chỉ cần trả về dữ liệu dạng JSON cho client và dữ liệu này sẽ được render bởi client với nội dung hoàn chỉnh.



Hình 4-17. Ý tưởng của mô hình SPA

Để người dùng có trải nghiệm tốt nhất trên hệ thống thì cần có những tiêu chuẩn về giao diện của hệ thống. Một hệ thống tốt không phải là một hệ thống mang đến nhiều hàng loạt chức năng, hiển thị hàng loạt thông tin trên màn hình. Mà một hệ thống tốt cần phải đẹp mắt, dễ nhìn, sáng sủa và dễ sử dụng. Để làm được việc này với chỉ một mình ReactJS là không đủ, vì vậy em chọn thêm một thư viện hỗ trợ xây dựng giao diện UI để xử lý. Thư viện Ant Design là một thư viện mã nguồn mở với cấp độ doanh nghiệp, bao gồm rất nhiều những React component có chất lượng cao và có thể dễ dàng chỉnh sửa, tạo những thay đổi. Đặc biệt với việc hệ thống xây dựng trang web có xu hướng là một trang admin dùng để quản lý các thành phần và thống kê dữ liệu, ant design là thư viện xứng đáng được sử dụng.

4.6. Xây dựng hệ thống phía bộ chuyển đổi lệnh giọng nói

Để xây dựng bộ chuyển đổi lệnh giọng nói, em lựa chọn sử dụng ngôn ngữ python làm cơ sở để xây dựng do đây được coi là ngôn ngữ tốt nhất để xử lý và phân tích dữ liệu. Python đi kèm với hai package Speech Recognition và google text-to-speech là sự kết hợp hoàn hảo cho bộ chuyển đổi lệnh giọng nói này.

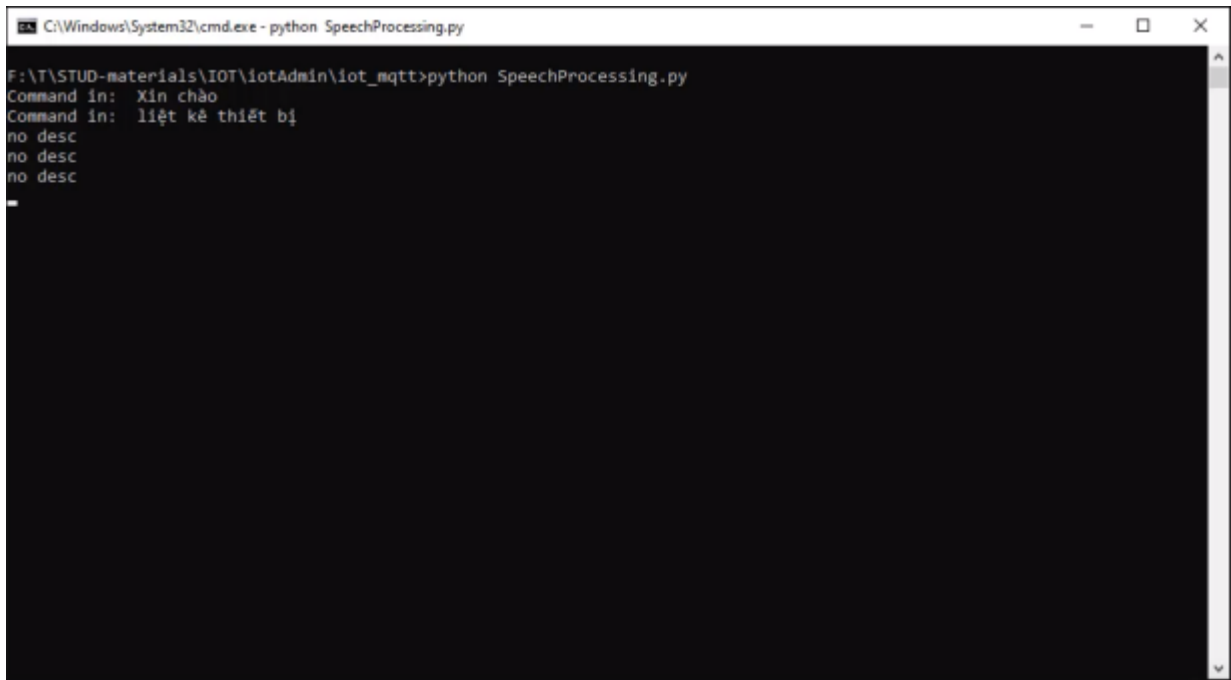
Bộ chuyển đổi lệnh giọng nói sẽ được xây dựng có thiên hướng giống với một voice bot và sau đây sẽ được gọi tắt là bot. Tại đây, dữ liệu đầu vào sẽ được sử dụng là những câu lệnh điều khiển bằng giọng nói được nhận dạng thông qua package Speech Recognition và sau khi đã nhận lệnh, bot sẽ trả lời người dùng với những dữ liệu tương ứng thông qua việc sử dụng package google text-to-speech. Một việc rất phiền đối với sử dụng bot chính là việc người sử dụng điều khiển bot một cách không mong muốn thông qua việc trò chuyện với những người xung quanh hay thậm chí những nội dung âm thanh mà người dùng sử dụng như tivi, đài, loa... Để tránh vấn đề này, em xây dựng bot với hai

trạng thái bao gồm: trạng thái chờ và trạng thái sẵn sàng. Bot chỉ thực hiện lệnh khi được chuyển sang trạng thái sẵn sàng bằng câu lệnh chào như “xin chào”, “chào”, “hello”.



Hình 4-18. Lệnh “xin chào”

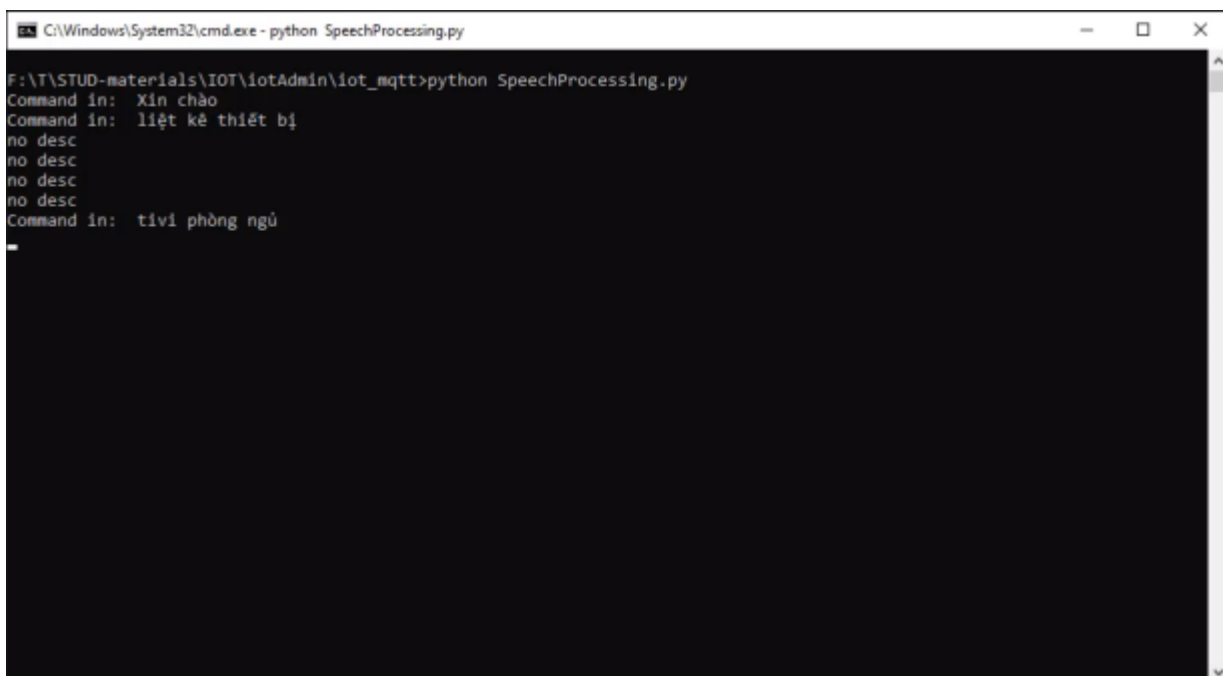
Sau khi chuyển sang sẵn sàng, người dùng sẽ được cung cấp một số chức năng sau đây: kiểm tra ngày giờ, kiểm tra thông tin các thiết bị hiện có trong hệ thống và điều khiển các thiết bị trong hệ thống. Đối với câu lệnh kiểm tra thông tin các thiết bị hiện có trong hệ thống, người dùng sẽ sử dụng câu lệnh “liệt kê thiết bị”.



```
C:\Windows\System32\cmd.exe - python SpeechProcessing.py
F:\T\STUD-materials\IOT\iotAdmin\iot_mqtt>python SpeechProcessing.py
Command in: Xin chào
Command in: liệt kê thiết bị
no desc
no desc
no desc
-
```

Hình 4-19. Lệnh “liệt kê thiết bị”

Sau khi nhận được câu lệnh, hệ thống sẽ sử dụng giao thức http để lấy thông tin các thiết bị hiện có trong hệ thống sau đó trả lời người dùng thông tin về tên thiết bị, mô tả về thiết bị. Trên hình 4-19 “no desc” được in ra thể hiện việc thiết bị được đăng ký nhưng không có mô tả thêm. Câu lệnh này được xây dựng nhằm mục đích cập nhật số lượng thiết bị trong hệ thống cho bot sau khi có thiết bị mới được đăng ký vào và mục đích thứ hai là cung cấp cho người bệnh tên và mô tả các thiết bị để người bệnh dễ dàng nhận diện đúng thiết bị cần điều khiển và sử dụng tên thiết bị cho việc điều khiển sau này. Sau khi đã cập nhật được danh sách thiết bị, lúc này, người dùng sẽ lần lượt gọi tên thiết bị cần điều khiển. Khi tên thiết bị được gọi lên, bot sẽ thực hiện tìm ra thông tin về thiết bị tương ứng nếu tên thiết bị trùng khớp với tên được đăng ký trong hệ thống.



```
C:\Windows\System32\cmd.exe - python SpeechProcessing.py
F:\T\STUD-materials\IOT\iotAdmin\iot_mqtt>python SpeechProcessing.py
Command in: Xin chào
Command in: liệt kê thiết bị
no desc
no desc
no desc
no desc
Command in: tivi phòng ngủ
```

Hình 4-20. Lựa chọn thiết bị điều khiển

Tại đây bot sẽ lấy ra được “id” của thiết bị và đưa vào payload để chờ sẵn cho việc gửi thông tin điều khiển sang bộ điều khiển trung tâm. Sau khi đã được xác thực về việc thiết bị đã được lựa chọn, người bệnh tiến hành đưa ra các câu lệnh điều khiển tương ứng với thiết bị. Các câu lệnh bằng giọng nói lúc này sẽ được tham chiếu đến danh sách câu lệnh điều khiển bằng giọng nói và đưa ra được lệnh máy để thêm vào payload. Tương quan giữa lệnh điều khiển bằng giọng nói và lệnh máy như hình dưới đây:

Loại thiết bị	Lệnh điều khiển	Lệnh điều khiển bằng tiếng việt
tivi	KEY_VOLUMEUP	tăng âm lượng
tivi	KEY_VOLUMEDOWN	giảm âm lượng
tivi	KEY_CHANNELUP	tiền kênh
tivi	KEY_CHANNELDOWN	hùi kênh
tivi	KEY_0	số 0:kênh 0
tivi	KEY_1	số 1:kênh 1
tivi	KEY_2	số 2:kênh 2
tivi	KEY_3	số 3:kênh 3
tivi	KEY_4	số 4:kênh 4
tivi	KEY_5	số 5:kênh 5
tivi	KEY_6	số 6:kênh 6
tivi	KEY_7	số 7:kênh 7
tivi	KEY_8	số 8:kênh 8
tivi	KEY_9	số 9:kênh 9
tivi	KEY_POWER	bật/tắt
switch	KEY_ON	bật
switch	KEY_OFF	tắt

Hình 4-21. Danh sách các câu lệnh tương ứng với từng loại thiết bị

Sau khi lệnh giọng nói đã được chuyển đổi và lấy ra được lệnh máy, bot sẽ tiến hành hoàn thiện payload và gửi cho bộ điều khiển thiết bị thông minh thông qua một mqtt topic là “iot/control”.

```
C:\Windows\System32\cmd.exe - python SpeechProcessing.py
F:\T\STUD-materials\IOT\iotAdmin\iot_mqtt>python SpeechProcessing.py
Command in: Xin chào
Command in: liệt kê thiết bị
no desc
no desc
no desc
no desc
Command in: tivi phòng ngủ
Command in: tăng âm lượng
{'id': 32, 'command': 'KEY_VOLUMEUP'}
Published: {'id': 32, 'command': 'KEY_VOLUMEUP'} on MQTT Topic: iot/control
```

Hình 4-22. Lệnh giọng nói và mẫu payload gửi đến bộ điều khiển thiết bị thông minh

Như hình trên, có thể thấy được thiết bị đã được lựa chọn để đưa ra được “id” của thiết bị được lưu trong hệ thống là “32” và từ câu lệnh điều khiển bằng giọng nói “tăng âm lượng” được chuyển đổi sang làm lệnh máy chung gian “KEY_VOLUMEUP” để hoàn thiện một payload và được gửi qua MQTT topic. Lúc này người bệnh có thể tiếp tục đưa ra các câu lệnh cho thiết bị hiện tại hoặc lựa chọn thiết bị khác thông qua việc gọi tên thiết bị đó. Nếu đã hoàn thành việc điều khiển thiết bị, người bệnh gọi câu lệnh “tạm biệt”, “bye” để tiến hành đưa bot về trạng thái chờ. Lúc này bot hoàn thành một phiên làm việc và tiếp tục chờ tới phiên tiếp theo.

4.7. Xây dựng hệ thống phía bộ điều khiển thiết bị thông minh

Để xây dựng được một hệ thống có thể điều khiển các thiết bị khác nhau, có cách điều khiển cũng như dữ liệu điều khiển khác nhau ta cần tổng quát được những cách điều khiển tín hiệu này và đưa chúng vào khuôn mẫu cũng như ghép với các câu lệnh điều khiển bằng giọng nói. Để làm được điều này, em đã lựa chọn ngôn ngữ Python làm cơ sở để xây dựng nên hệ thống điều khiển thiết bị bằng giọng nói. Python từ lâu đã là một ngôn ngữ được coi là một trong những ngôn ngữ tốt nhất để xử lý và phân tích dữ liệu. Trong hệ thống, để có thể tương tác phi vật lý được với những thiết bị trong nhà, cần phải phát triển được công cụ có thể điều khiển các thiết bị này từ xa. Để làm được điều đó, giao thức MQTT và thư viện LIRC là không thể thiếu đi. Hiện nay, các thiết bị trong nhà

hầu hết đều có thể được điều khiển thông qua sóng hồng ngoại (IR signal) hoặc thông qua mạng wifi với giao thức MQTT. Chính vì vậy, em lựa chọn hai công nghệ này làm nền tảng để xây dựng nên công cụ giúp người dùng điều khiển các thiết bị tại nơi ở của mình và chia các loại thiết bị được điều khiển làm hai loại chính là điều khiển thông qua MQTT và điều khiển hồng ngoại. Ngoài ra bộ điều khiển trung tâm cũng tích hợp việc nhận dữ liệu thu thập từ các sensor thông qua các kênh được định nghĩa sẵn.

Tại bộ điều khiển thiết bị thông minh, dữ liệu đầu vào sẽ là mẫu dữ liệu tương tự được gửi đến từ bộ chuyển đổi lệnh giọng nói. Bằng việc nhận dữ liệu theo mẫu và gửi qua MQTT topic đã giúp cho hệ thống có thể tích hợp nhiều loại giao diện người dùng hơn. Dữ liệu sau khi được đưa vào sẽ được tiến hành lấy ra thông tin thiết bị tương ứng với id và xác minh tính tương thích của câu lệnh gửi đến với câu lệnh được đăng ký cho loại thiết bị này. Nếu chính xác, dữ liệu sẽ được đưa vào xử lý với cách xử lý tương ứng với hai loại thiết bị điều khiển đã nêu trên.

Đối với những thiết bị điều khiển bằng giao thức MQTT, dữ liệu điều khiển đầu vào sẽ một lần nữa được tham chiếu tới dữ liệu điều khiển cụ thể của từng thiết bị. Với những thiết bị thuộc loại điều khiển qua MQTT có trong hệ thống hiện tại, hầu hết đều có chung một khuôn mẫu dành cho tín hiệu điều khiển, sự khác nhau chỉ nằm ở topic mà thiết bị đó sử dụng. Chính vì vậy, có thể tổng quát hóa các mẫu tín hiệu này thành một khuôn mẫu và chỉ cần thay đổi những tín hiệu cần thiết tương ứng với từng lệnh điều khiển của người dùng.

```
{  
  "id": "1037600",  
  "data": {"out1": 0, "out2": 100, "out3": 0}  
}
```

Hình 4-23. Mẫu dữ liệu điều khiển smart switch thông qua giao thức MQTT

Như hình trên, có thể thấy được mẫu dữ liệu điều khiển của một thiết bị smart switch. Chỉ cần thay thế dữ liệu của các out1, out2, out3, chúng ta có thể điều khiển ba thiết bị tương ứng. Chính vì vậy, từ thông tin “gangs” được lấy ra từ bảng device trong cơ sở dữ liệu, chúng ta có thể thực hiện chuyển đổi mẫu dữ liệu trên tùy theo thiết bị được đăng ký và gửi đi theo topic đã được đăng ký cùng thiết bị trên hệ thống.

Đối với những thiết bị có cách điều khiển thông qua sóng hồng ngoại. Hệ thống cũng đã cung cấp khả năng thu lại tín hiệu hồng ngoại tương ứng với từng nút điều khiển

trên backend và frontend. Sau khi đã thu được đủ tín hiệu hồng ngoại, một tệp cấu hình sẽ được lưu trong hệ thống và việc còn lại chính là sử dụng câu lệnh để điều khiển thông qua tệp vừa được sinh ra.

```
IRSEND SEND_ONCE tv_phongngu KEY_VOLUMEUP
```

Hình 4-24. Mẫu câu lệnh sử dụng trong điều khiển bằng sóng hồng ngoại

Chính vì vậy, ở đây, chúng ta chỉ cần đối chiếu câu lệnh điều khiển của người dùng với những câu lệnh được hỗ trợ trong hệ thống, sau đó sử dụng câu lệnh để lấy ra được tín hiệu điều khiển từ các tệp cấu hình đã lưu tại server. Khi này, có thể gửi các tín hiệu điều khiển này thông qua đèn LED phát sóng hồng ngoại được gắn trên phần cứng của hệ thống là đã có thể thực hiện điều khiển.

Đối với việc thu thập dữ liệu từ các cảm biến, chỉ cần các cảm biến được đăng ký với MQTT topic gửi dữ liệu, hệ thống có thể nhận dữ liệu từ các topic này và tiến hành so sánh với topic đã được đăng ký. Nếu topic trùng nhau, dữ liệu sẽ được đưa vào lưu trữ trong hệ thống thông qua việc gọi giao thức http đến backend.

CHƯƠNG 5: KẾT QUẢ

5.1. Hình ảnh mô tả hệ thống về đăng ký thiết bị

5.1.1. Giao diện đăng nhập hệ thống

Đăng nhập

Tài khoản:

Mật khẩu:

☒ Nhớ tài khoản?

Đăng nhập

Hình 5-1. Giao diện đăng nhập hệ thống

5.1.2. Giao diện thống kê thiết bị

IOT Admin

Thống kê thiết bị

Thống số dữ liệu

Hồng ngoại

Điều khiển

Đăng xuất

Admin / Thiết bị

Tổng số:

2

Số lượng bản ghi:

5

Thêm

Thống kê thiết bị:

Id	Tên	Nguyên mẫu	Loại thiết bị	Control Topic	Output Topic	Loại switch	Ngày tạo	Mô tả	Trạng thái	Tương tác
1	nhệt độ phòng ngủ		sensor	iot/temp_11		0	2021-12-15 23:46:20			Chỉnh sửa Xóa
2	đèn phòng ngủ		switch	CyberLink/commands/1037600		2	2021-12-15 23:46:33			Chỉnh sửa Xóa

<

1

>

IOT admin @Designed by NvMe

Hình 5-2. Giao diện thống kê thiết bị

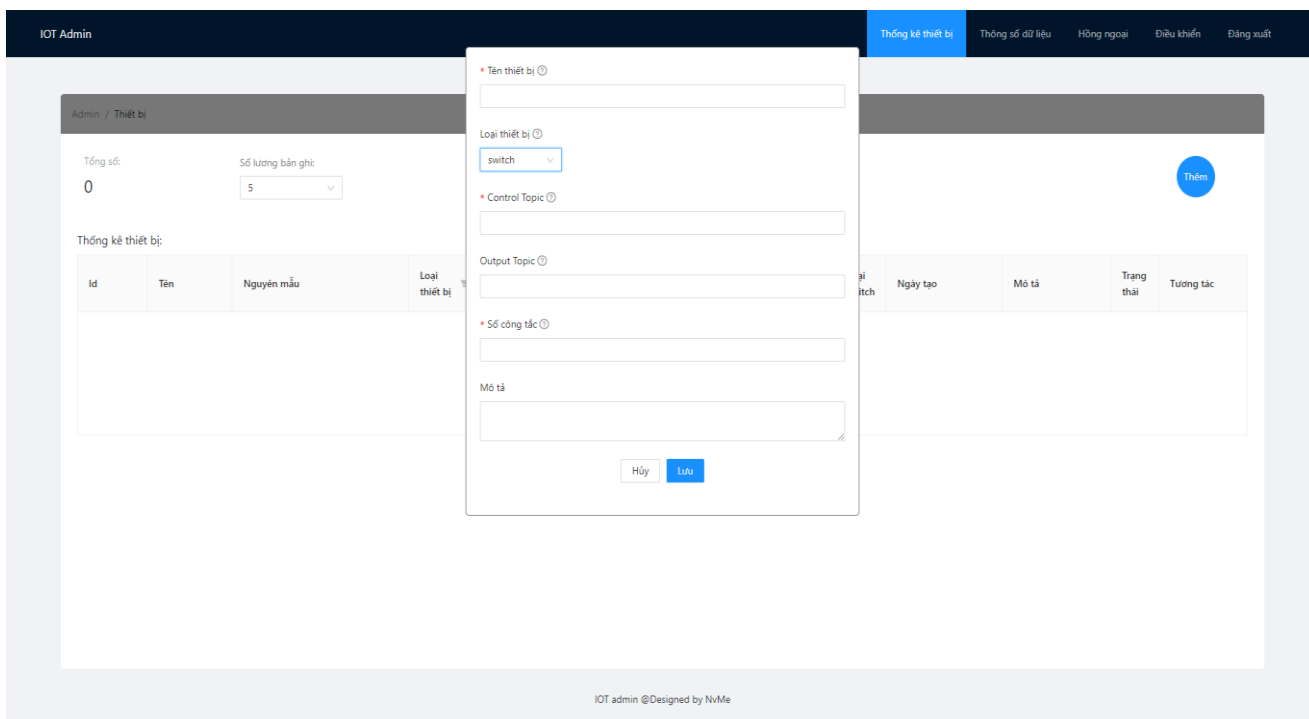
5.1.3. Giao diện thêm mới thiết bị

The screenshot displays the 'IOT Admin' web application. A modal window titled 'Thêm' (Add) is open, allowing the user to create a new device. The modal contains the following fields:

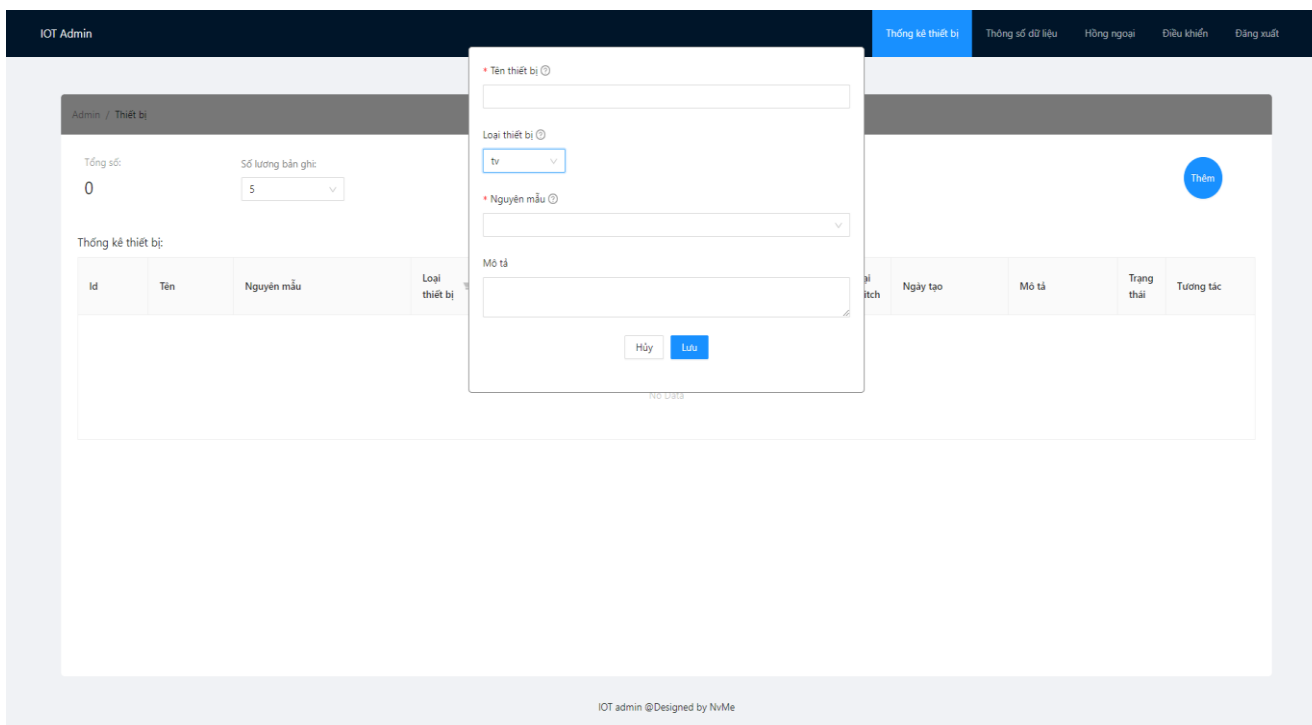
- Tên thiết bị** (Device Name): A text input field.
- Loại thiết bị** (Device Type): A dropdown menu with 'sensor' selected.
- Control Topic**: A text input field.
- Output Topic**: A text input field.
- Mô tả** (Description): A text area.

At the bottom of the modal are 'Hủy' (Cancel) and 'Lưu' (Save) buttons. In the background, the main interface shows a sidebar with 'Admin / Thiết bị' and a top navigation bar with links like 'Thống kê thiết bị', 'Thống số dữ liệu', 'Hồng ngoại', 'Điều khiển', and 'Đăng xuất'. A table with columns 'Id', 'Tên', 'Nguyên mẫu', and 'Loại thiết bị' is partially visible.

Hình 5-3. Giao diện thêm thiết bị sensor

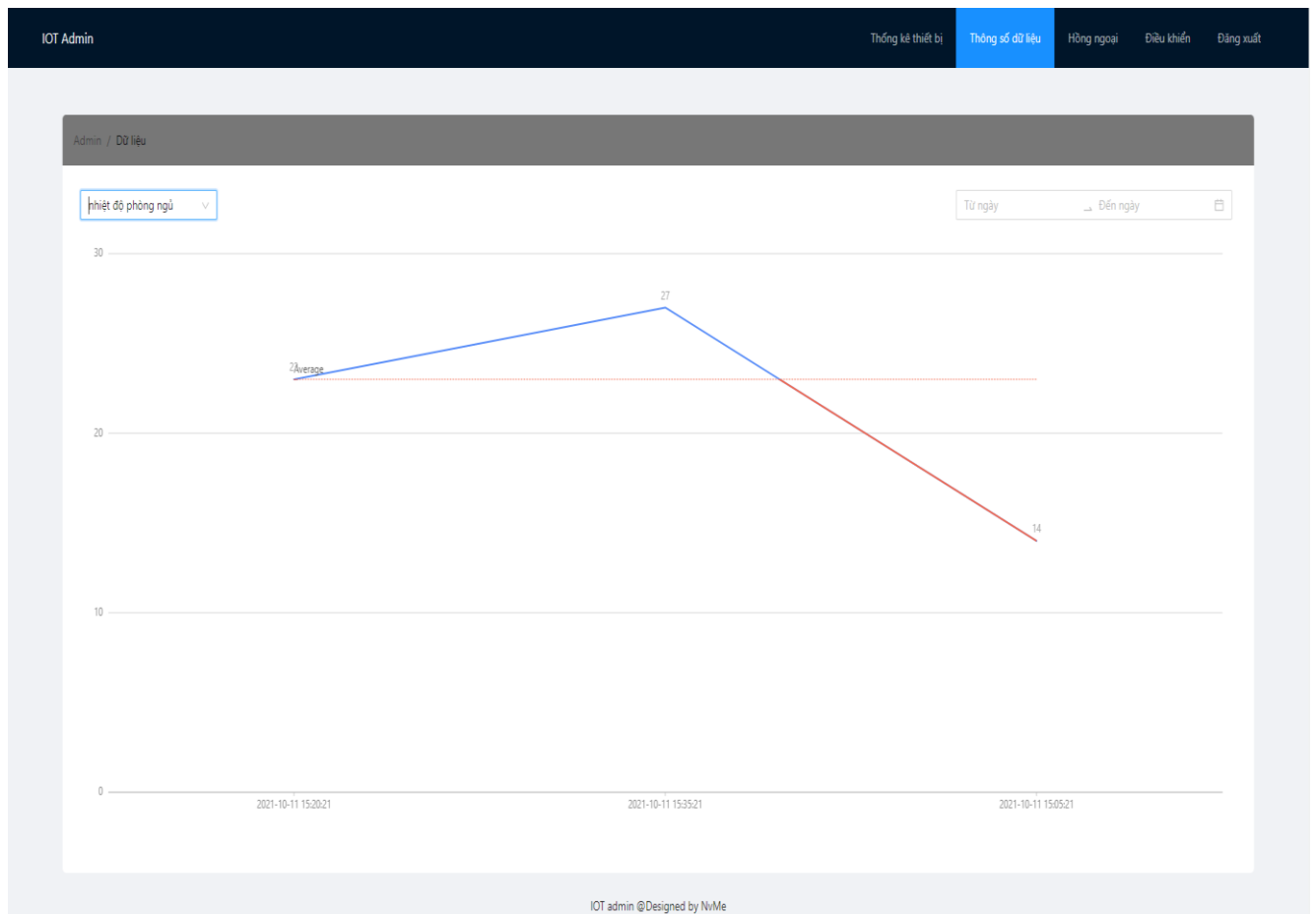


Hình 5-4. Giao diện thêm thiết bị switch

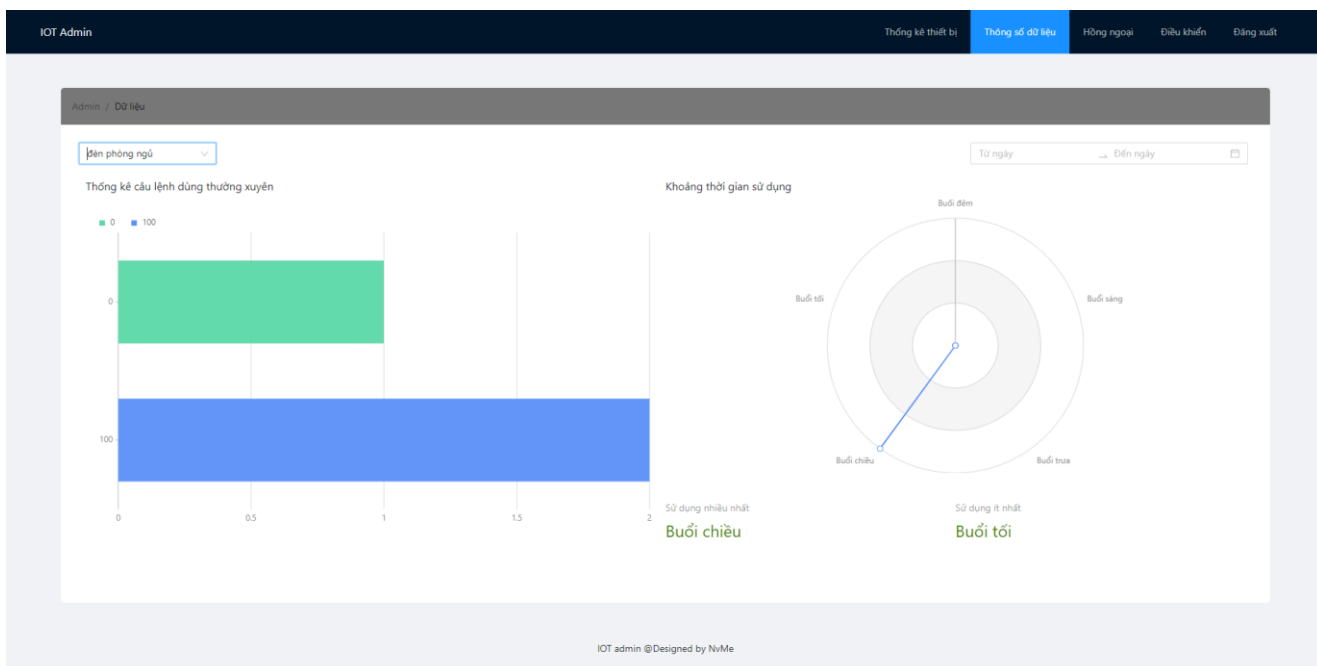


Hình 5-5. Giao diện thêm thiết bị với các thiết bị điều khiển bằng hồng ngoại

5.1.4. Giao diện thông số dữ liệu



Hình 5-6. Giao diện thông số dữ liệu với dữ liệu từ sensor

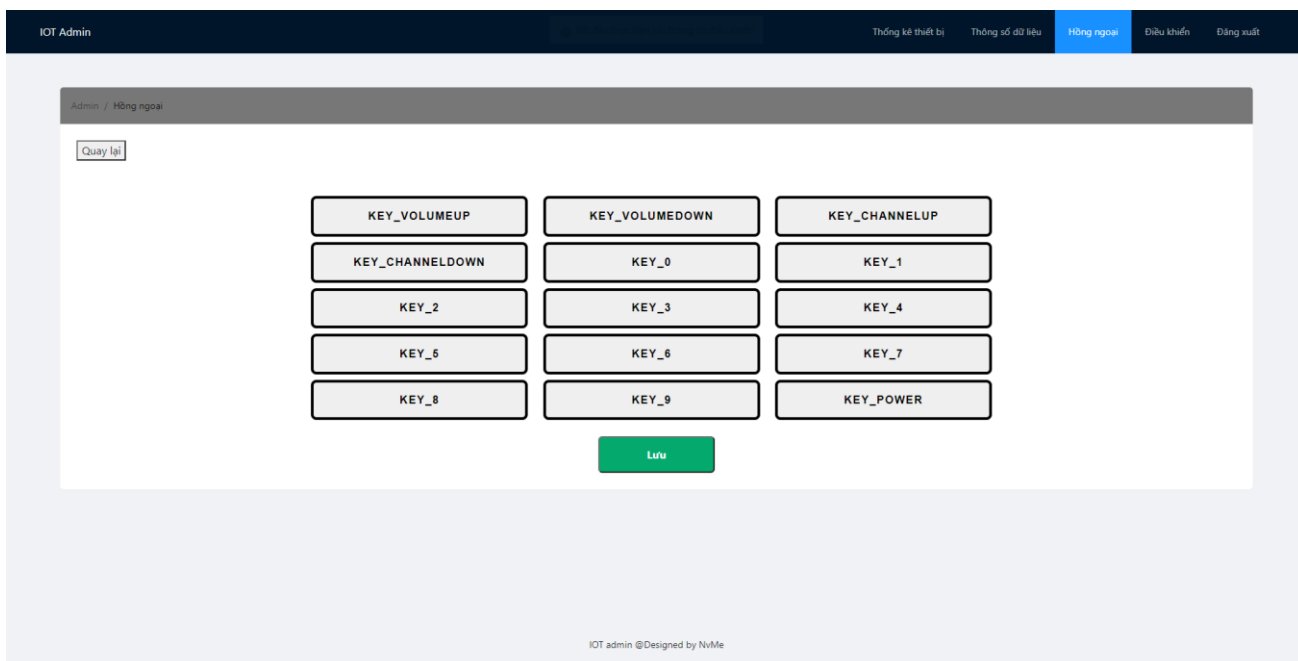


Hình 5-7. Giao diện thông số dữ liệu với các dữ liệu điều khiển

5.1.5. Giao diện thêm thông tin tệp cấu hình điều khiển

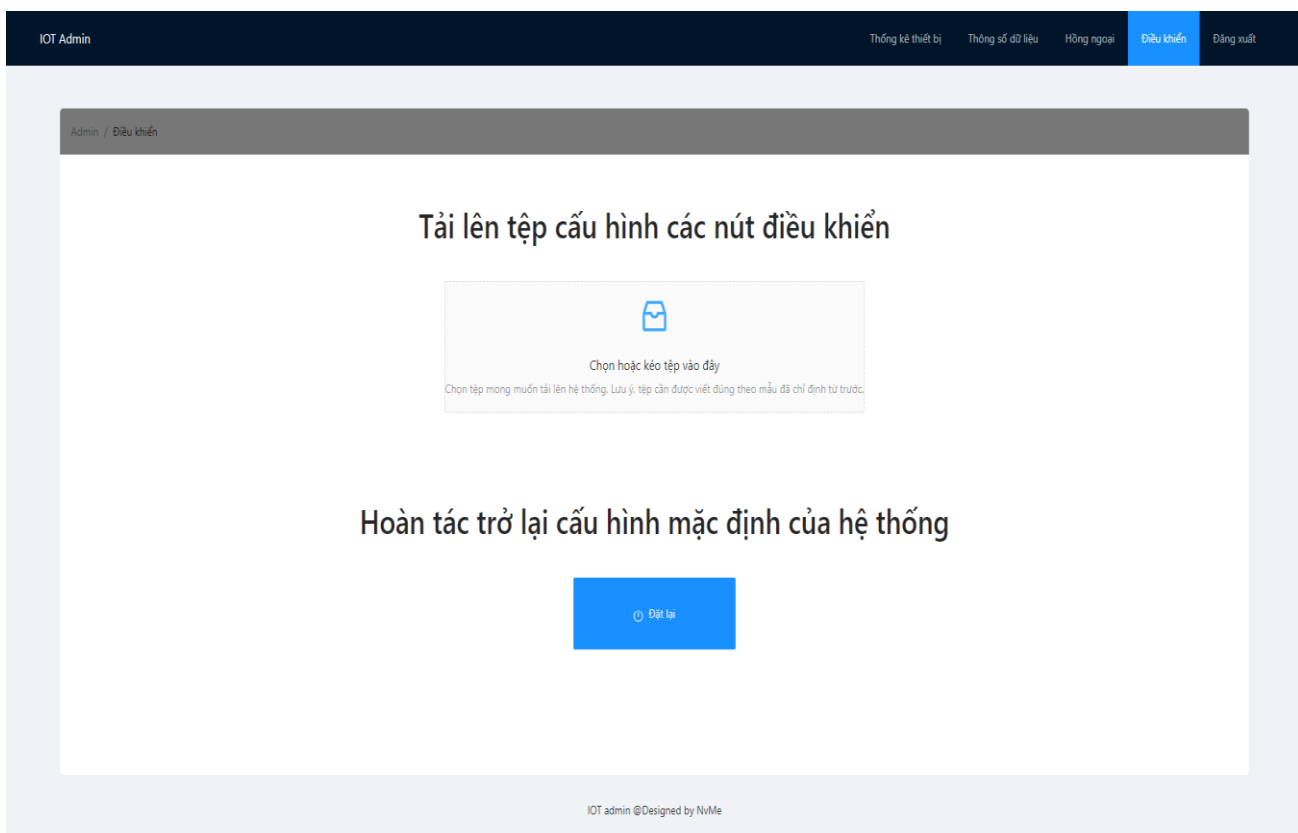
The screenshot shows the 'IOT Admin' interface. The top navigation bar includes links for 'Thống kê thiết bị', 'Thông số dữ liệu', 'Hồng ngoại' (highlighted), 'Điều khiển', and 'Đăng xuất'. The breadcrumb trail indicates 'Admin / Hồng ngoại'. The main heading is 'Thêm thông tin điều khiển thiết bị hồng ngoại'. Below this is a text input field labeled 'Tên thiết bị sẽ được lưu'. At the bottom, there are three buttons labeled 'Tivi', 'air-conditioner', and 'fan'. The footer text reads 'IOT admin @Designed by NvlMe'.

Hình 5-8. Giao diện thêm tệp cấu hình điều khiển thiết bị hồng ngoại 1



Hình 5-9. Giao diện thêm tệp cấu hình điều khiển thiết bị hồng ngoại 2

5.1.6. Giao diện thay đổi cấu hình các nút điều khiển thiết bị



Hình 5-10. Giao diện thay đổi thông tin các nút điều khiển hồng ngoại

CHƯƠNG 6: KẾT LUẬN VÀ PHƯƠNG HƯỚNG PHÁT TRIỂN

Trong chăm sóc sức khỏe bệnh nhân, việc áp dụng công nghệ làm thay đổi thói quen của con người theo hướng tích cực, giúp người giám hộ giảm tải bớt thời gian túc trực bên người bệnh và người bệnh cũng có thể tự mình chăm sóc bản thân. Vì những lý do đó, tương tác người máy mang đến những cơ hội, những tiềm năng khổng lồ trong việc thay đổi phương thức chăm sóc bệnh nhân. Mỗi một ứng dụng nhỏ cũng có thể mang lại lợi ích thiết thực cho không chỉ bệnh nhân mà còn những người xung quanh. Chính vì vậy, việc xây dựng một phần mềm, một hệ thống có thể hỗ trợ người bệnh suy giảm khả năng vận động trong việc tự chủ những mong muốn sinh hoạt đời thường cũng như bác sĩ, người nhà bệnh nhân có thể giám sát được sức khỏe của người bệnh là cực kì cần thiết và thiết thực.

Trong khóa luận trên đã trình bày quá trình ứng dụng nhận diện giọng nói vào điều khiển các thiết bị IOT và một số vấn đề thực tế trong quá trình xây dựng nên hệ thống như: Khảo sát bài toán thực tế, phân tích yêu cầu, việc lựa chọn công nghệ phù hợp, thiết kế kiến trúc, thiết kế các chức năng để giải quyết vấn đề, thiết kế cơ sở dữ liệu, ... Đến hiện tại, hệ thống đã có thể hỗ trợ tương tác giữa người dùng và các thiết bị trong nhà thông qua các loại hình điều khiển không dây. Trong khóa luận này, em đã nhận được rất nhiều kinh nghiệm thiết thực trong việc xây dựng và thiết kế phần mềm ở cả hai phía người dùng (frontend) lẫn máy chủ (backend) và cách thức ứng dụng điều khiển các thiết bị trong nhà thông qua phần cứng RPI4.

Hệ thống hiện tại đã có hầu hết các chức năng cơ bản để có thể mang đến cho người dùng những lợi ích, tuy nhiên vẫn có thể phát triển thêm nhiều tính năng mới. Đối với một hệ thống được đặt trên một phần cứng cố định, khó nâng cấp hay mở rộng tài nguyên thì việc quản lý tài nguyên là hết sức quan trọng. Chỉ một thay đổi nhỏ cũng có thể giúp hệ thống nhanh hơn hoặc làm hệ thống chậm đi. Trong tương lai, kế hoạch phát triển hệ thống đó là giảm bớt số lượng lượt request, giảm bớt độ lớn các gói tin được truyền qua lại trong hệ thống nhằm giảm tải lượng tiêu tốn tài nguyên, tăng cảm giác trải nghiệm người dùng khi hệ thống phần nào đó nhanh hơn và ổn định hơn. Hơn nữa việc thống kê thông tin hiện nay vẫn có thể được phát triển thêm nếu áp dụng các mô hình hay thuật toán để có thể phân tích dữ liệu và đưa ra tổng quan hơn về tình trạng người bệnh; đơn cử

như việc áp dụng các phương pháp data mining vào dữ liệu cảm biến cơ thể¹³[13]. Việc áp dụng các phương pháp này có thể đưa ra cho những người giám hộ thông tin tổng quan hơn về người bệnh khi phương pháp sử dụng chính những dữ liệu được thu thập từ tình trạng của người bệnh thông qua các cảm biến và tạo thành các gợi ý cho người giám hộ về những biểu hiện bất thường trong thông số của người bệnh.

¹³ **Mining Productive-Associated Periodic-Frequent Patterns in Body Sensor Data for Smart Home Care,**
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5461076/>

TÀI LIỆU THAM KHẢO

Tiếng việt

- [5] “MQTT và vai trò trong IOT - viblo”,
<https://viblo.asia/p/mqtt-la-gi-vai-tro-cua-mqtt-trong-iot-V3m5WL3bKO7>
- [10] “Spring Restful và mô hình 3 lớp – kipalog.com”,
<https://kipalog.com/posts/Spring-Boot--Spring-Restful-va-mo-hinh-3-lop>
- [11] “Lỗi hỏng log4j của java và mức độ ảnh hưởng – vnexpress.net”
<https://vnexpress.net/lo-hong-lon-nhat-thap-ky-co-the-anh-huong-nhieu-he-thong-o-viet-nam-4403132.html>

Tiếng anh

- [1] “Viet Nams first large scale national survey people disabilities – unicef.org”
<https://www.unicef.org/vietnam/press-releases/launch-key-findings-viet-nams-first-large-scale-national-survey-people-disabilities>
- [2] "Why You Should Use React.js For Web Development - freeCodeCamp." 18 thg 2. 2021,
<https://www.freecodecamp.org/news/why-use-react-for-web-development/>.
- [3] “What is Spring Boot – stackify.com”,
<https://stackify.com/what-is-spring-boot/>
- [4] “Mariadb – mariadb.org”
<https://mariadb.org/>
- [6] “What is mosquitto – mosquitto.org”,
<https://mosquitto.org/>
- [7] “What is LIRC – lirc.org”
<https://www.lirc.org/>
- [8] “Python Speech Recognition – github.com”
https://github.com/Uberi/speech_recognition#readme
- [9] “How to get started with google text-to-speech – towardsdatascience.com”,
<https://towardsdatascience.com/how-to-get-started-with-google-text-to-speech-using-python-485e43d1d544>
- [12] “What is JWT – jwt.io”
<https://jwt.io/introduction>
- [13] Walaa N. Ismail and Mohammad Mehedi Hassan, “Mining Productive-Associated Periodic-Frequent Patterns in Body Sensor Data for Smart Home Care”, ncbi.nlm.nih.gov, 2017, pp. 10-13”