

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI



BÁO CÁO MÔN HỌC: IOT VÀ ỨNG DỤNG

Đề tài : Hệ thống giám sát sức khỏe

Nhóm sinh viên thực hiện:

STT	Họ và tên	MSSV
1	Nguyễn Tiến Nam	20194337
2	Ngô Trọng Nghĩa	20194341
3	Đào Xuân An	20190076
4	Nguyễn Quốc Tuấn	20194402

Giáo viên hướng dẫn : TS. PHẠM NGỌC HÙNG

Nhóm : 11

Hà Nội, ngày 26 tháng 07 năm 2022

Mục lục

CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI.....	2
1.1 Lý do chọn đề tài	2
1.2 Tổng quan về sản phẩm	3
1.3 Mục tiêu của đề tài	3
CHƯƠNG 2. CÔNG NGHỆ SỬ DỤNG	3
2.1 Giao thức kết nối.....	3
MQTT.....	3
Hypertext Transfer Protocol (HTTP).....	5
2.2. Nodejs	5
2.3. React Native	6
2.4 Reactjs	6
2.5 MongoDB	6
CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	6
3.1. Sơ đồ tổng quan.....	6
3.2 Biểu đồ use-case tổng quan cho các chức năng	7
- Biểu đồ use-case phân rã Quản lý thiết bị theo dõi.....	9
3.3 Thiết kế cơ sở dữ liệu.....	9
CHƯƠNG 4. Kết quả demo.....	12
4.1 Dữ liệu nhận được từ cảm biến	12
4.2 Giao diện web	12
4.3 Giao diện app android.....	15
CHƯƠNG 5. TỔNG KẾT, KẾT LUẬN VÀ ĐÁNH GIÁ	21
5.1. Bảng phân chia công việc của các thành viên và đánh giá.....	21
5.2. Những kết quả thu được	22
5.3. Hướng phát triển	22
CHƯƠNG 6. Tài liệu tham khảo	22

CHƯƠNG 1. TỔNG QUAN ĐỀ TÀI

1.1 Lý do chọn đề tài

- Cuộc sống thời hiện đại ngày càng bận rộn và áp lực khiến cho nhiều người không có đủ thời gian chăm sóc cho người thân
- Sản phẩm ra đời nhằm đáp ứng nhu cầu theo dõi sức khỏe định kỳ của người thân đặc biệt là người cao tuổi

1.2 Tổng quan về sản phẩm

- Người sử dụng đeo thiết bị lên cơ thể
- Thiết bị được bật và tính toán dữ liệu liên tục
- Dữ liệu gồm: nhiệt độ, nhịp tim, huyết áp,..
- Người giám hộ đăng nhập trên ứng dụng android hoặc web để theo dõi các thông tin về sức khỏe của người thân

1.3 Mục tiêu của đề tài

- Thiết kế thành công hệ thống giám sát sức khỏe.
- Triển khai được ứng dụng trên android và web
- Hệ thống ổn định, chính xác

CHƯƠNG 2. CÔNG NGHỆ SỬ DỤNG

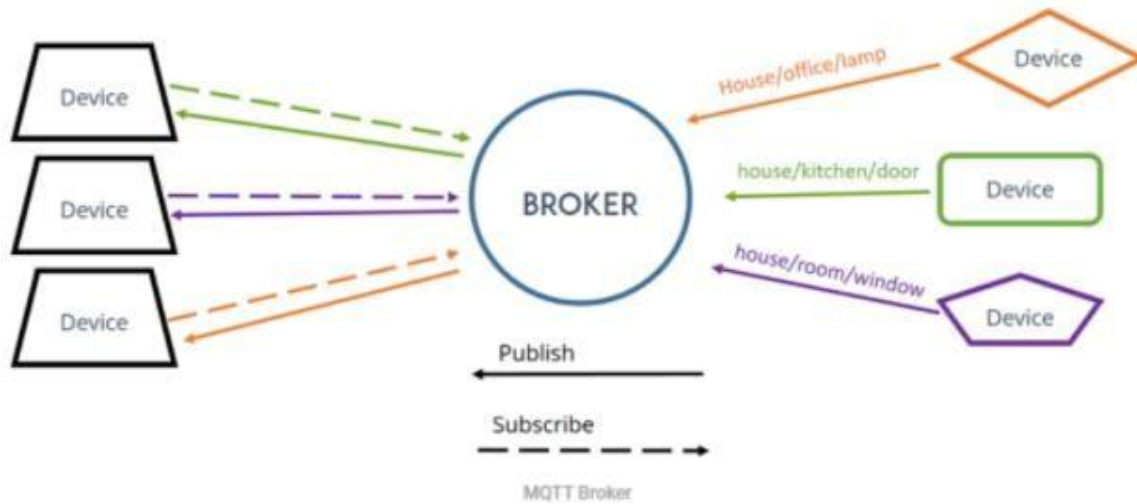
2.1 Giao thức kết nối

MQTT

MQTT (Message Queue Telemetry Transport) là một giao thức truyền thông điệp theo mô hình Publish/Subscribe sử dụng băng thông thấp, độ tin cậy cao và có khả năng hoạt động trong điều kiện đường truyền không ổn định.

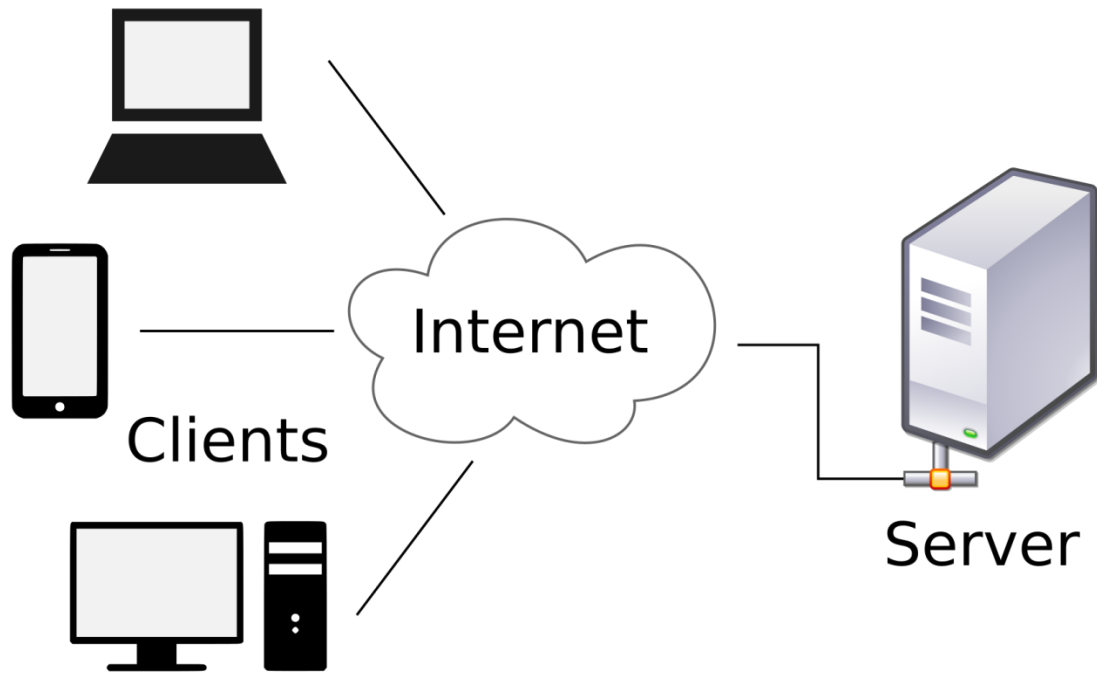
Kiến trúc của MQTT gồm 2 phần chính là Broker và Clients. Trong đó, broker được coi như là trung tâm, nó là điểm giao của tất cả các kết nối đến từ client. Nhiệm vụ chính của broker là nhận message từ publisher, xếp các message theo hàng đợi rồi chuyển chúng tới những địa chỉ cụ thể. Nhiệm vụ của broker là nó có thể đảm nhận thêm một vài tính năng liên quan tới quá trình truyền thông điệp như là: bảo mật message, lưu trữ message, logs,... Client thì được chia thành 2

nhóm là publisher và subscriber. Client là các thành phần phần mềm hoạt động trên các thiết bị hoặc các website nên chúng được thiết kế để có thể hoạt động một cách linh hoạt. Client chỉ làm ít nhất một trong 2 việc là publish các message lên một topic cụ thể hoặc subscribe một topic nào đó.



Hypertext Transfer Protocol (HTTP)

Giao thức hoạt động ở tầng application dùng để truyền các thông tin dạng siêu văn bản như HTML, hoạt động theo mô hình client – server



Được sử dụng rộng rãi trong môi trường web, là phương thức cực kì phổ biến giúp kết nối các trình duyệt web và các web servers trên khắp thế giới

2.2. Nodejs

Một mã nguồn mở, một môi trường cho các máy chủ và ứng dụng mạng Node.js sử dụng Google V8 JavaScript engine để thực thi mã, và một tỷ lệ lớn các mô-đun cơ bản được viết bằng JavaScript. Các ứng dụng node.js thì được viết bằng JavaScript.

Node.js cung cấp kiến trúc hướng sự kiện (event-driven) và non-blocking I/O API, tối ưu hóa thông lượng của ứng dụng và có khả năng mở rộng cao

Một vài ưu điểm của Nodejs:

- Có tốc độ xử lý nhanh nhờ cơ chế xử lý bất đồng bộ (non-blocking). Bạn có thể dễ dàng xử lý hàng ngàn kết nối trong khoảng thời gian ngắn nhất.

- Nhận và xử lý nhiều kết nối chỉ với một single-thread. Nhờ đó, hệ thống xử lý sẽ sử dụng ít lượng RAM nhất và giúp quá trình xử Nodejs lý nhanh hơn rất nhiều.

2.3. Android

Android đang là hệ điều hành chiếm lĩnh thị trường thiết bị di động hiện nay. Hàng triệu thiết bị điện thoại di động đều chạy trên nền android. Chính vì thế, nhóm đã lựa chọn ngôn ngữ java để lập trình android cho điện thoại di động

2.4 Reactjs

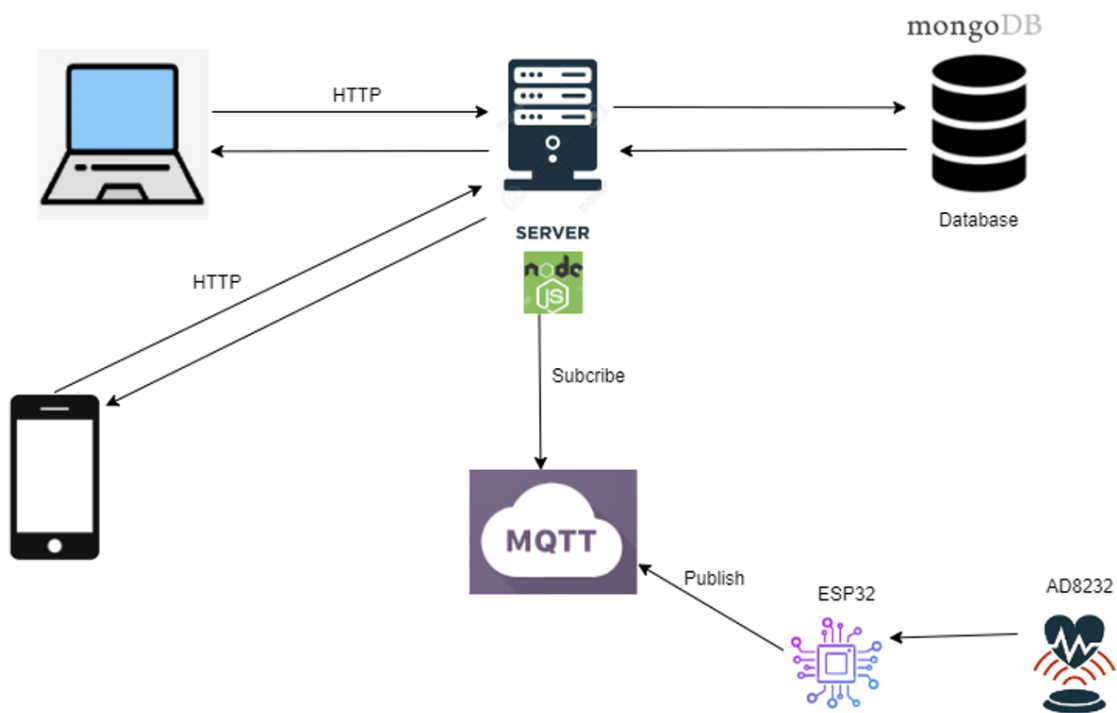
Reactjs là một opensource được phát triển bởi Facebook, ra mắt vào năm 2013, bản thân nó là một thư viện Javascript được dùng để xây dựng các tương tác với các thành phần trên website. Một trong những điểm nổi bật nhất của ReactJS đó là việc render dữ liệu không chỉ thực hiện được tầng Server mà còn ở Client nữa.

2.5 MongoDB

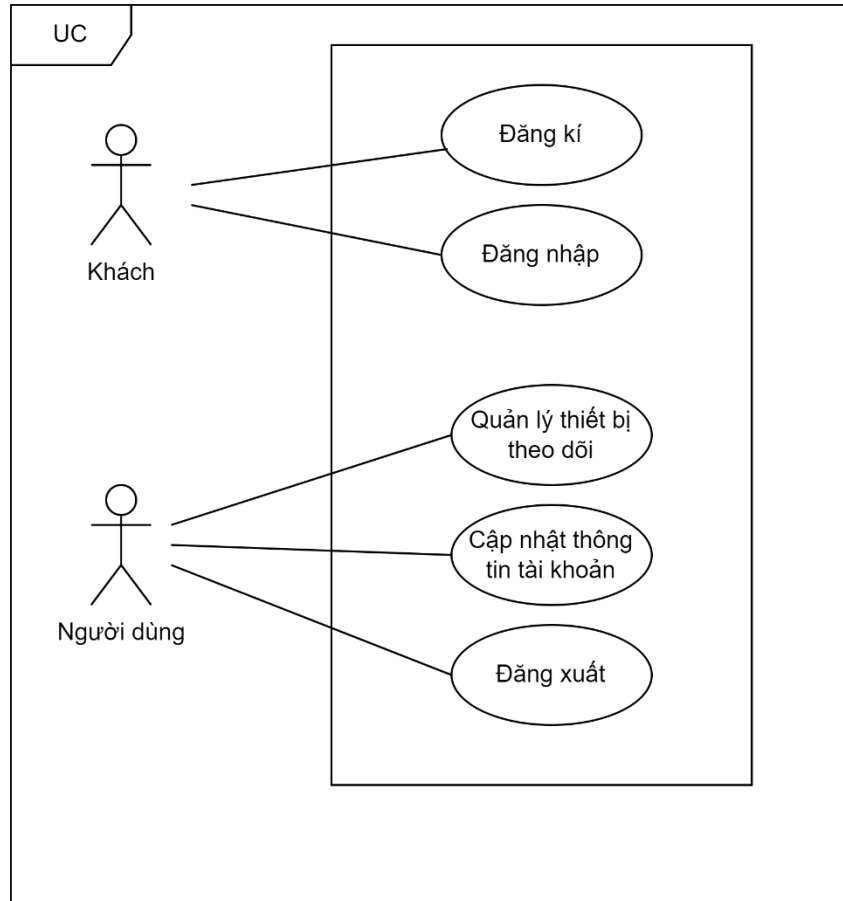
MongoDB là một chương trình cơ sở dữ liệu mã nguồn mở được thiết kế theo kiểu hướng đối tượng trong đó các bảng được cấu trúc một cách linh hoạt cho phép các dữ liệu lưu trên bảng không cần phải tuân theo một dạng cấu trúc nhất định nào. Chính do cấu trúc linh hoạt này nên MongoDB có thể được dùng để lưu trữ các dữ liệu có cấu trúc phức tạp và đa dạng và không cố định (hay còn gọi là Big Data)

CHƯƠNG 3. PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

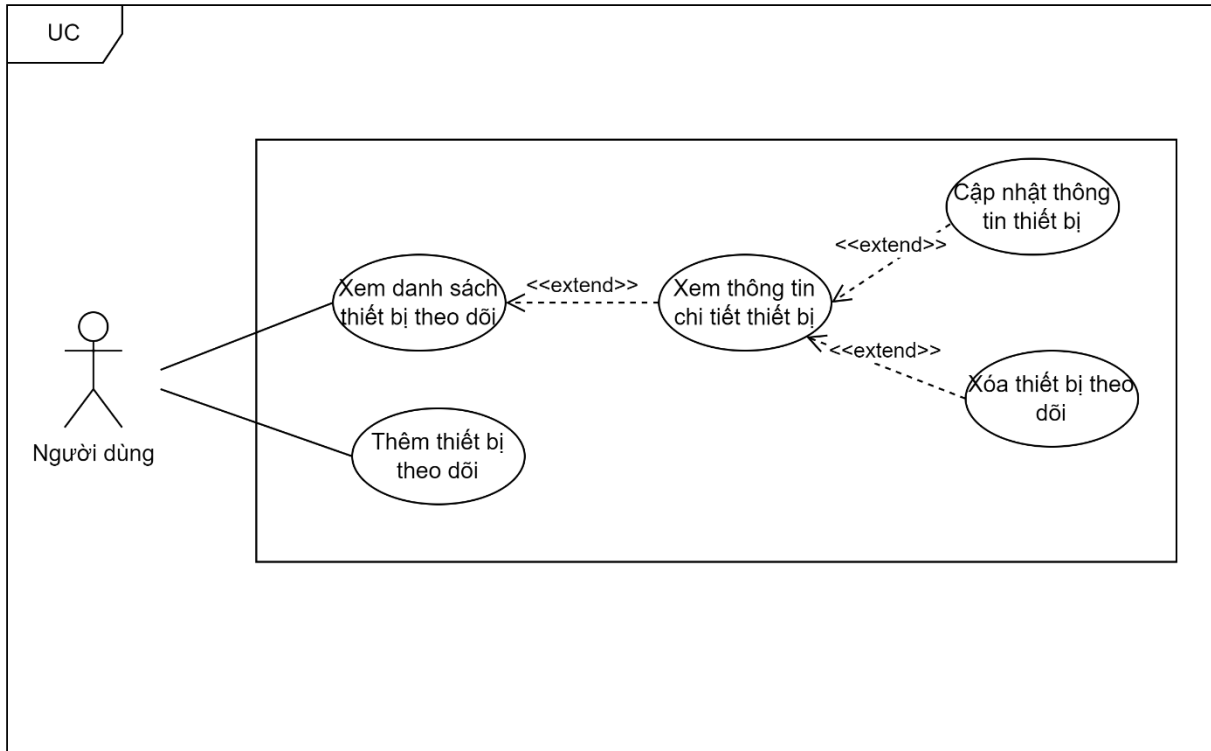
3.1. Sơ đồ tổng quan



3.2 Biểu đồ use-case tổng quan cho các chức năng



- Biểu đồ use-case phân rã Quản lý thiết bị theo dõi



3.3 Thiết kế cơ sở dữ liệu

- Công nghệ sử dụng: Hệ thống cơ sở dữ liệu MongoDB

Bảng User:

Trường	Kiểu dữ liệu	Kích thước	Khuôn dạng	Ghi chú
Id	string		ký tự	
name	string	200 ký tự	ký tự	
role	string	20 ký tự	ký tự	
username	string	200 ký tự	ký tự	
age	number		số nguyên	
weight	number		số thực	
height	number		số thực	
createdAt	datetime		thời gian	

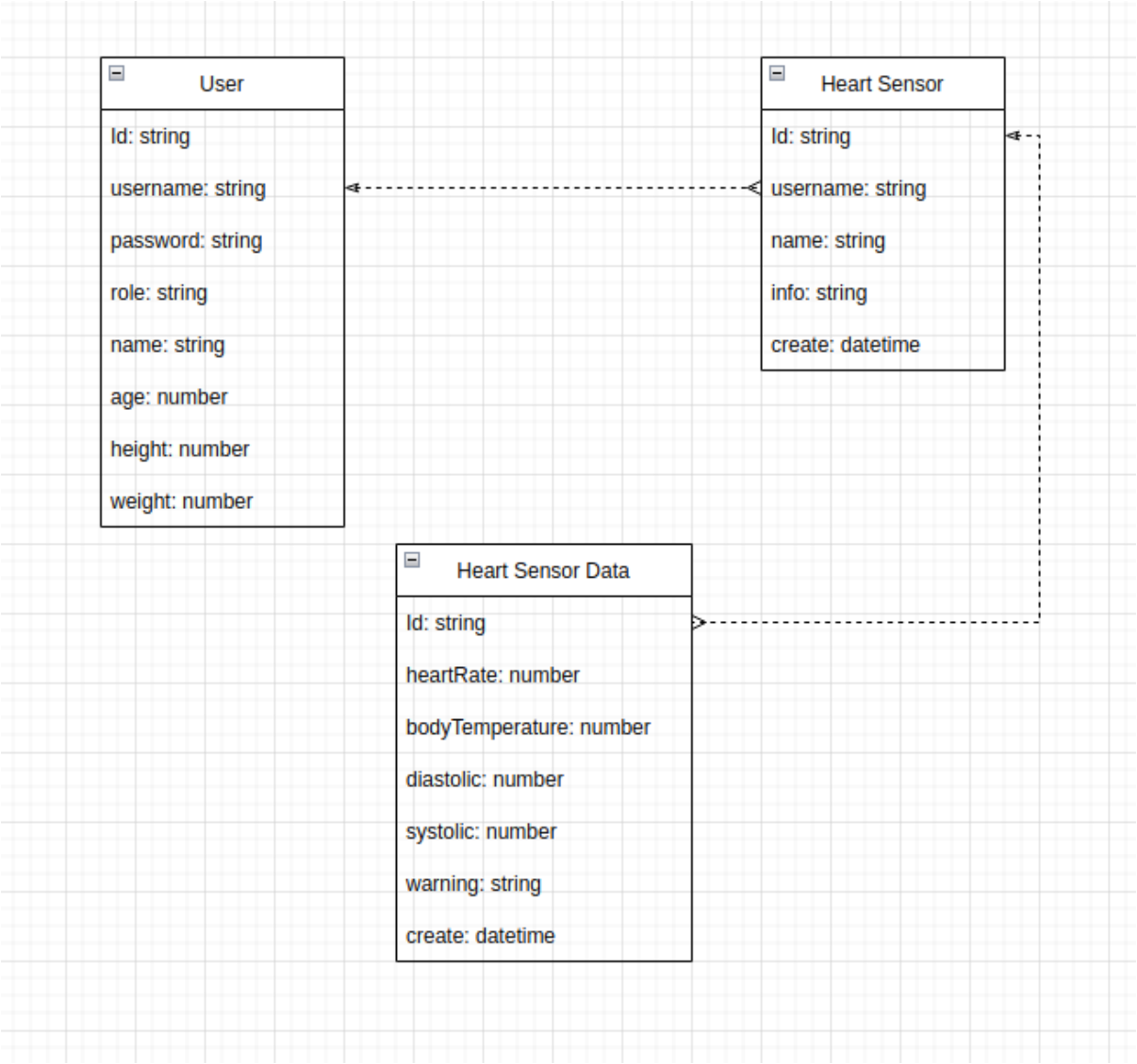
Bảng Heart Sensor:

Trường	Kiểu dữ liệu	Kích thước	Khuôn dạng	Ghi chú
ID	string		ký tự	
username	string		ký tự	
name	string		ký tự	
info	string		ký tự	
createdAt	datetime		thời gian	

Bảng Heart Sensor Data

Trường	Kiểu dữ liệu	Kích thước	Khuôn dạng	Ghi chú
Id	string		ký tự	
heartRate	number		số nguyên	
bodyTemperature	number		số nguyên	
diastolic	number		số nguyên	

systolic	number		số nguyên	
warning	string		ký tự	
createdAt	datetime		thời gian	



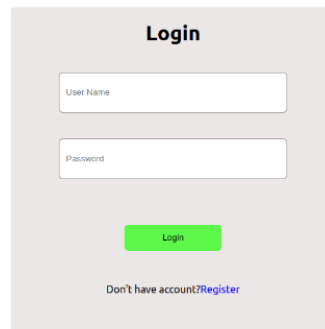
CHƯƠNG 4. Kết quả demo

4.1 Dữ liệu nhận được từ cảm biến

```
{"Id": "12345", "heartRate": "110", "bodyTemperature": "38", "systolic": "100", "diastolic": "90"}
```

4.2 Giao diện web

- Đăng nhập, đăng ký



The image shows a login form titled "Login" centered on a light gray background. It contains two input fields: "User Name" and "Password". Below these fields is a green "Login" button. At the bottom, there is a link that says "Don't have account? Register".

Register

User Name

Password

Name

Age

Weight

Height

[Register](#)

You have an account? [Login](#)

- Cập nhật thông tin cá nhân

Device chart

Nghia

Tuannq

tuannq

tuann3

tuann4

tuann5

tuann6

tuann7

t

tr

User Information

User Name

Name

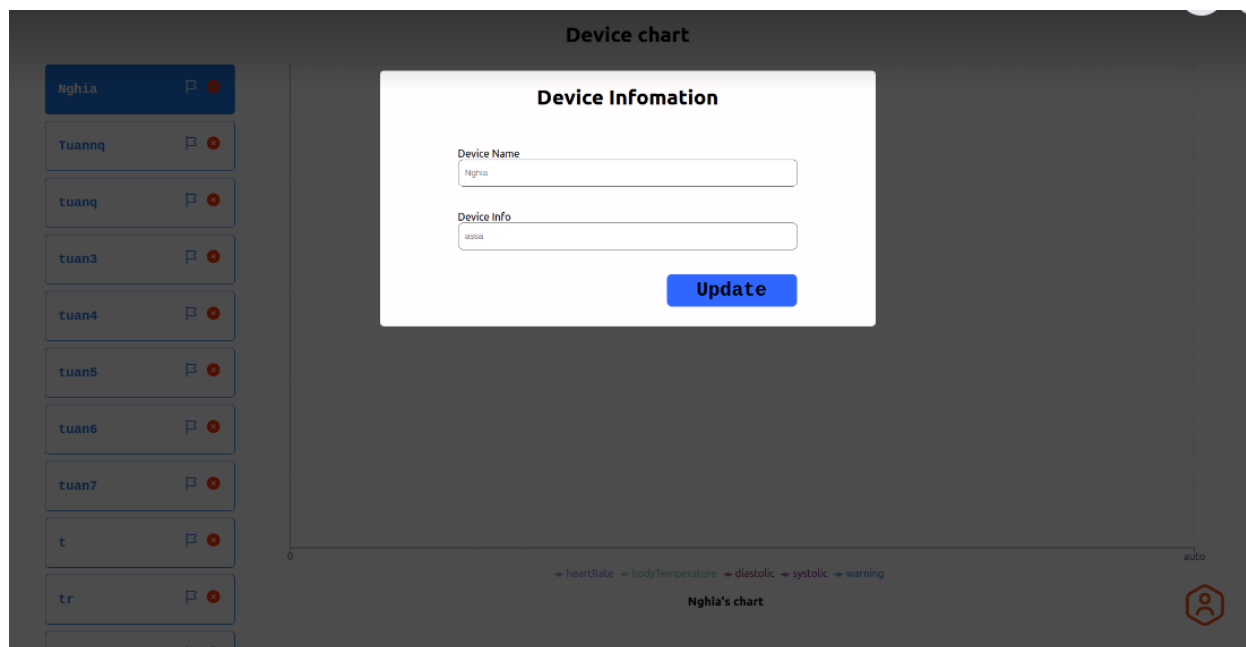
Age

Weight

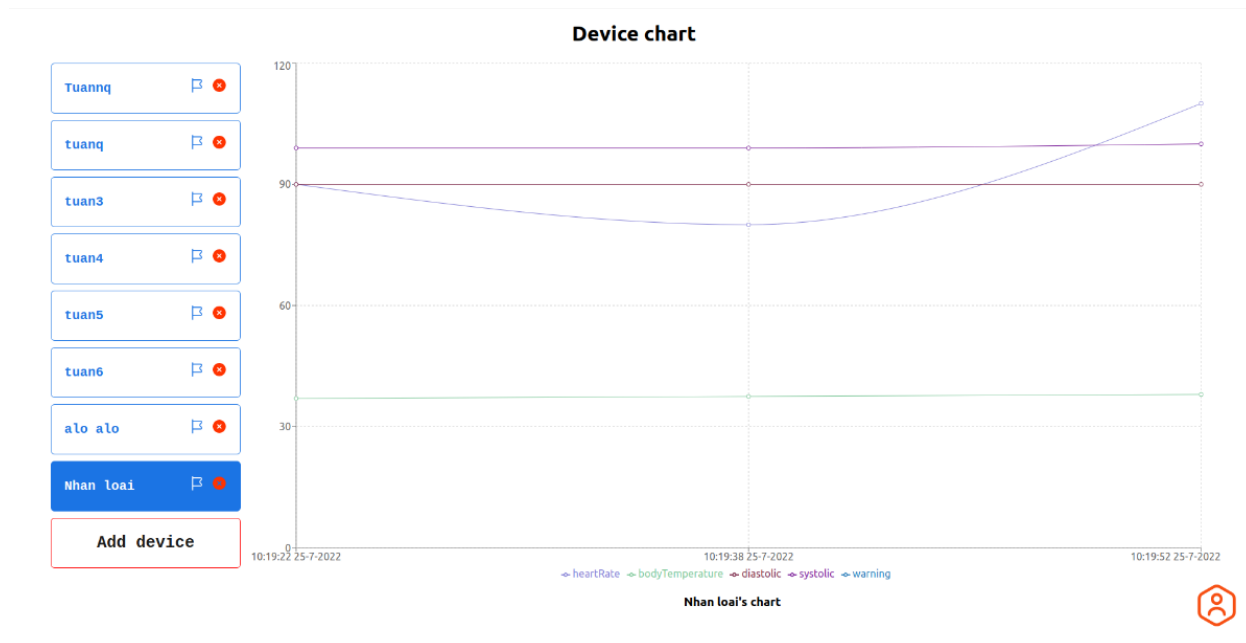
Height

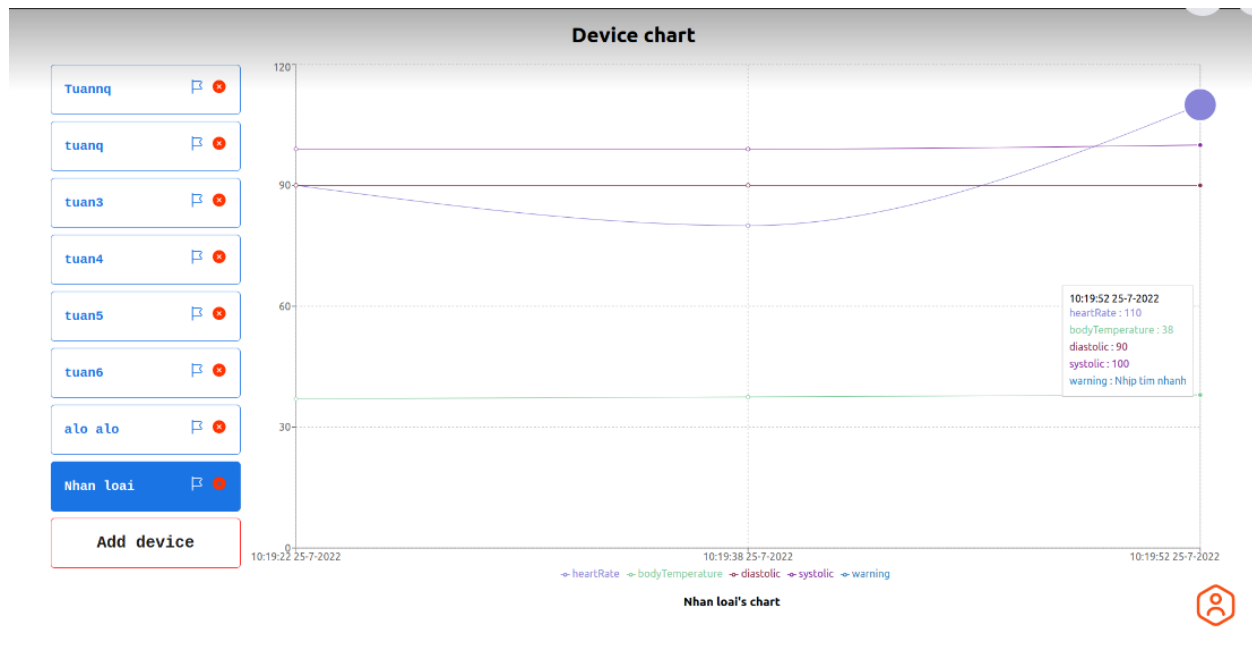
[Save](#) [Logout](#)

- Cập nhật thông tin thiết bị



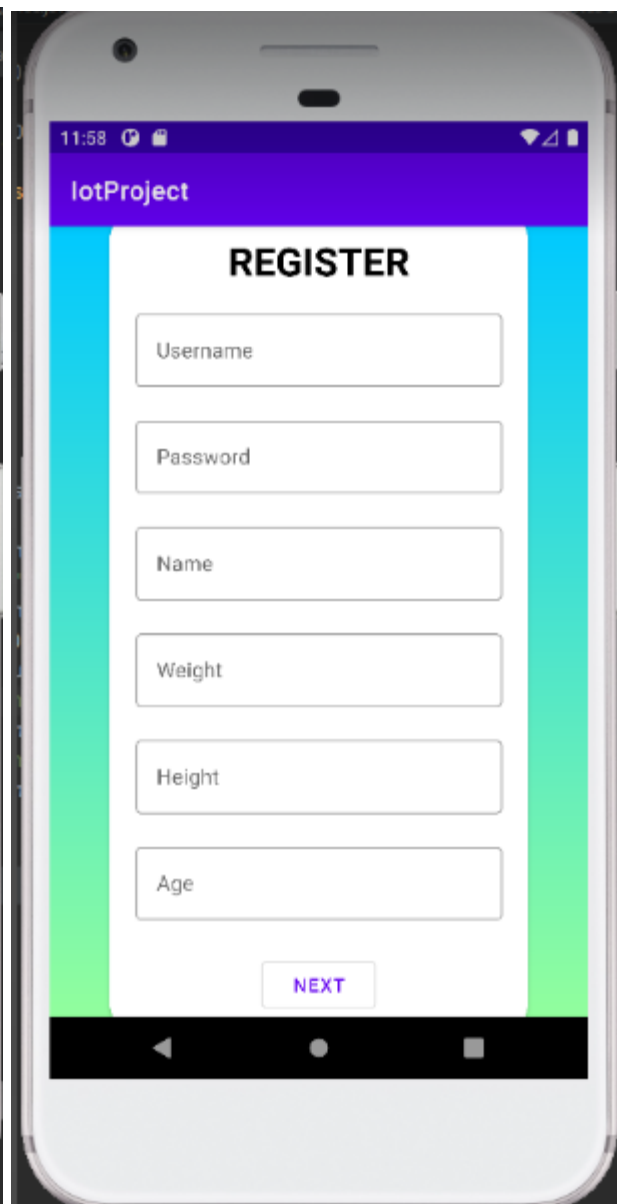
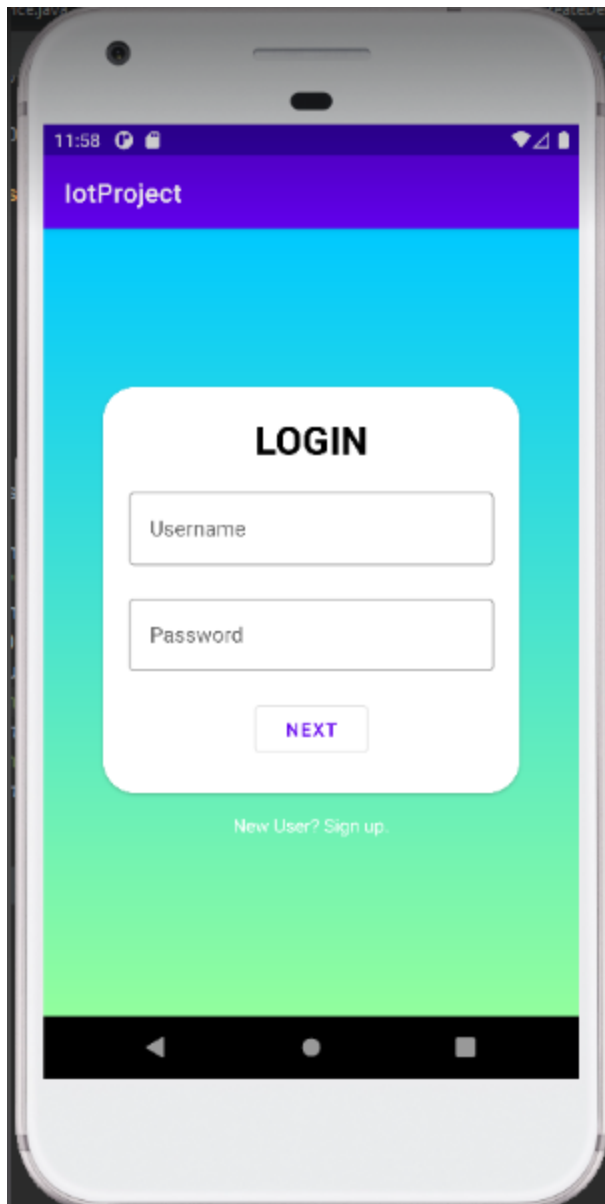
- Dữ liệu nhận được từ thiết bị





4.3 Giao diện app android

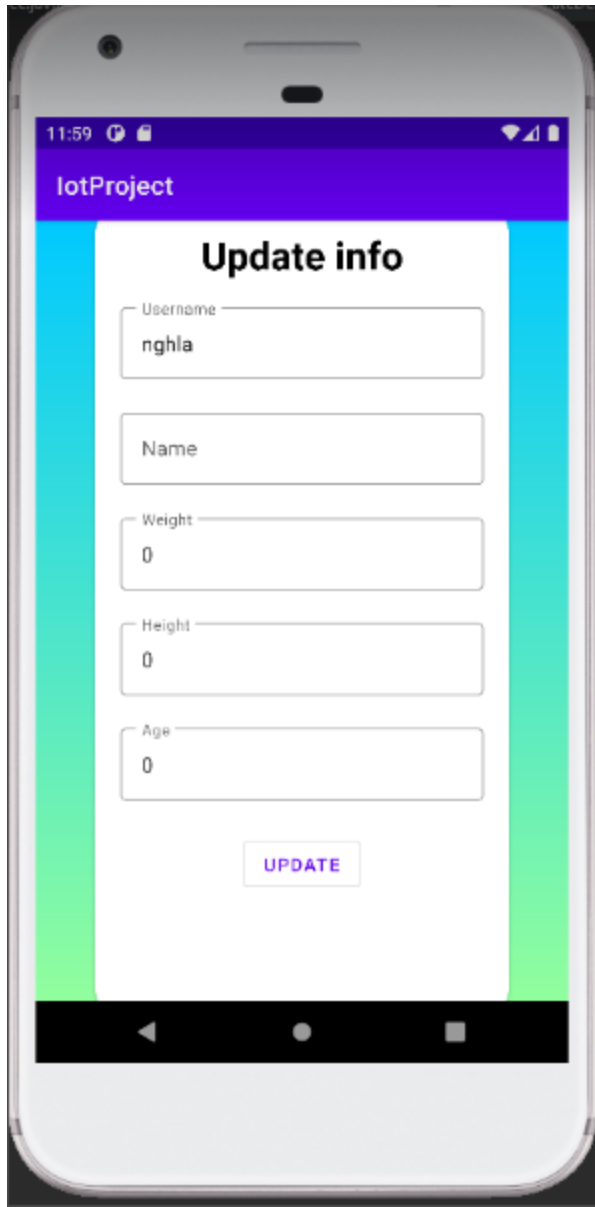
- Đăng kí, đăng nhập



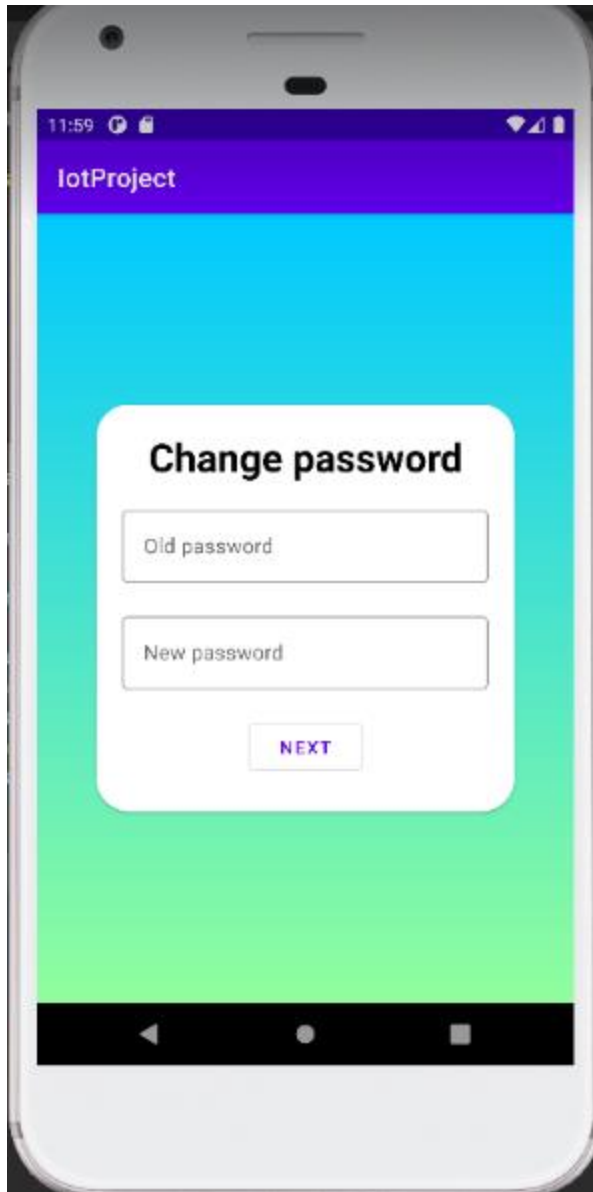
- Danh sách thiết bị



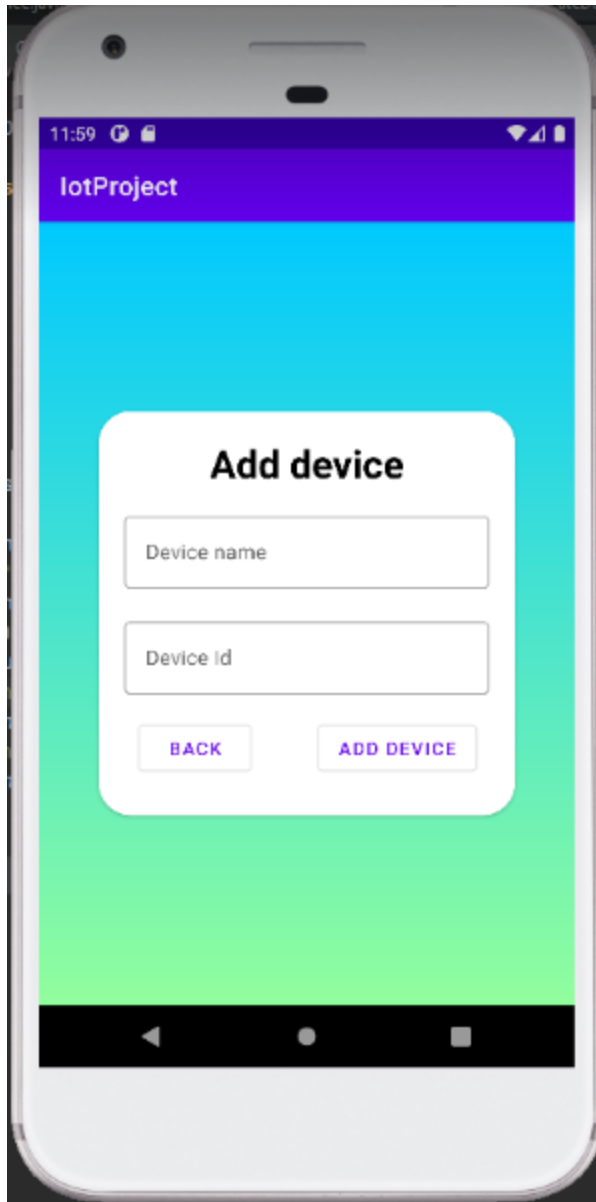
Cập nhật thông tin tài khoản



- Đổi mật khẩu



- Thêm thiết bị theo dõi



- Xem thông tin thiết bị theo dõi



CHƯƠNG 5. TỔNG KẾT, KẾT LUẬN VÀ ĐÁNH GIÁ

5.1. Bảng phân chia công việc của các thành viên và đánh giá

Công việc	Thành viên thực hiện
Server Nodejs, làm báo cáo	Ngô Trọng Nghĩa
Web frontend reactjs	Đào Xuân An

ESP32+sensor, làm slide	Nguyễn Tiến Nam
Android	Nguyễn Quốc Tuấn

5.2. Những kết quả thu được

- Server có cấu trúc rõ ràng, hoạt động chính xác trong việc xử lý các yêu cầu
- Giao diện web, app dễ sử dụng kể cả áp dụng kể cả với người ít tiếp xúc với công nghệ
- Sản phẩm đã đáp ứng được các yêu cầu đề ra, hoạt động khá ổn định

5.3. Hướng phát triển

- Thu gọn thiết bị phần cứng
- Đo được thêm nhiều loại dữ liệu đo được
- Có thêm nhiều cảnh báo liên quan đến sức khỏe

CHƯƠNG 6. Tài liệu tham khảo

1. TS. Phạm Ngọc Hưng, Slides môn học “IoT và Ứng dụng”, Trường CNTT&TT – Đại học Bách Khoa Hà Nội, 2021.
2. <https://reactjs.org/>
3. <https://nodejs.org/en/>
4. <https://developer.android.com/courses/fundamentals-training/overview-v2>
5. <https://developer.android.com/docs>