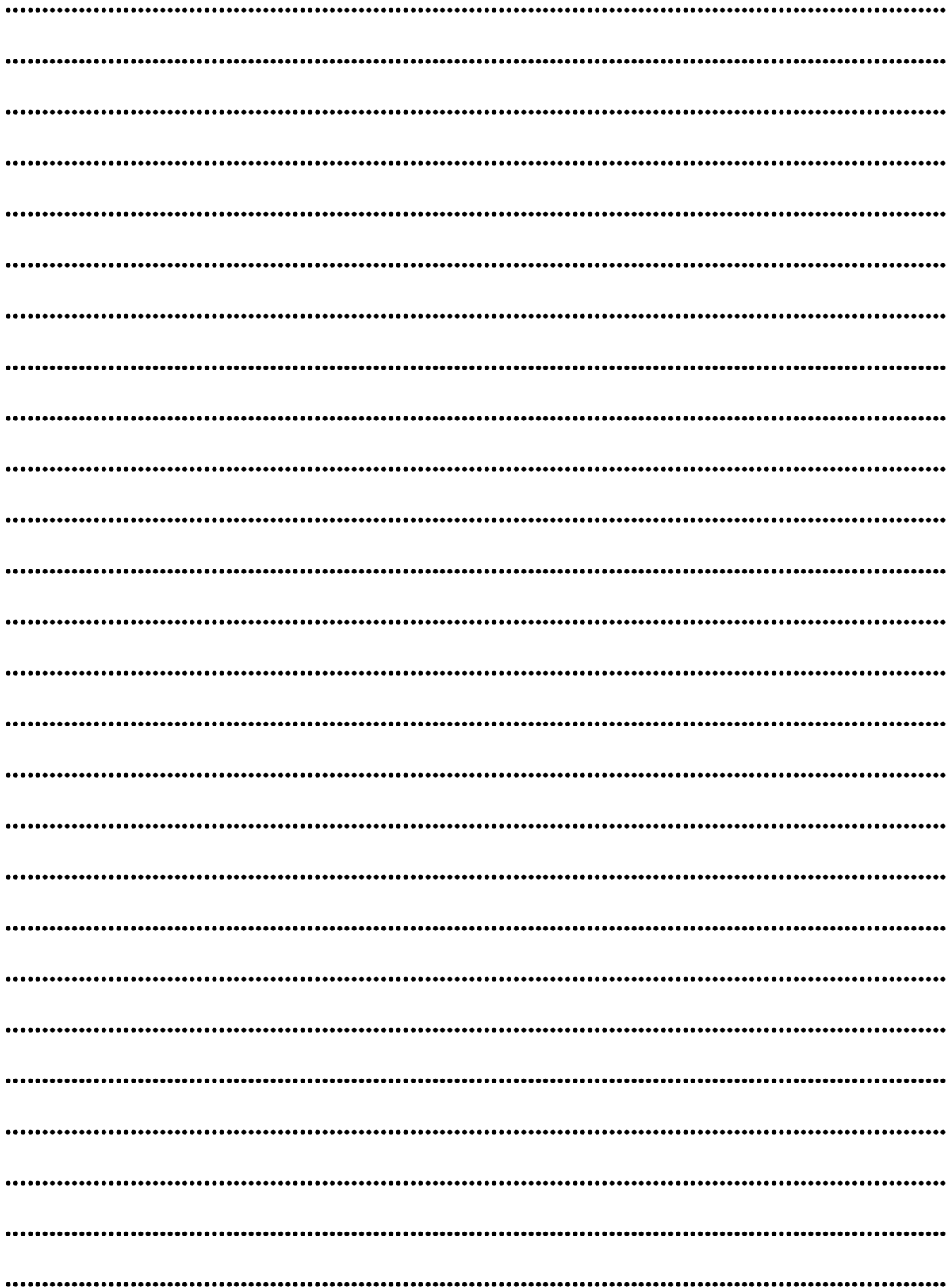
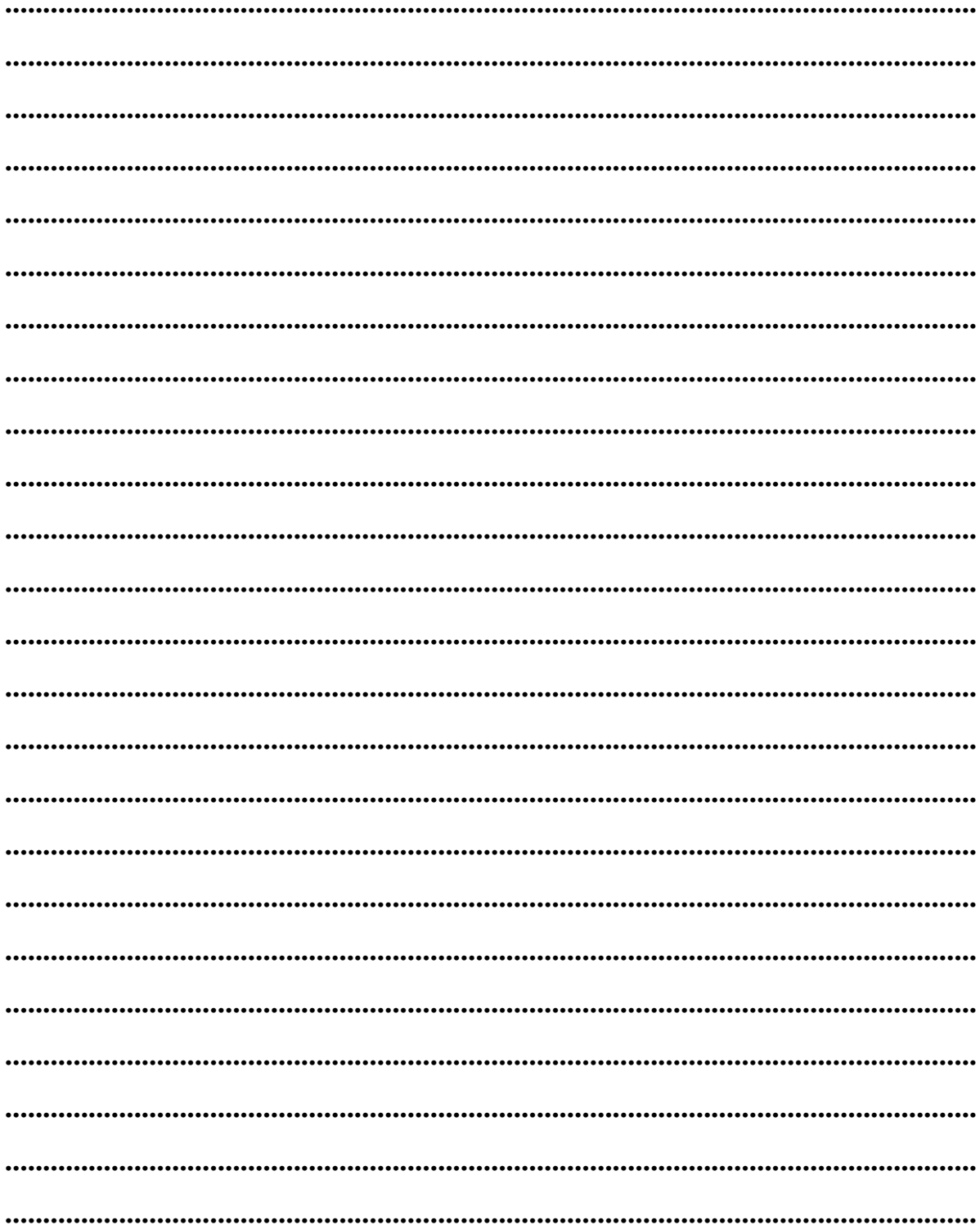


[illegible]



[illegible]



NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

TT	Họ và tên	Mã số SV	Lớp	Khoa
1	Trần Thế Đầu	106150160	15DT3	Điện tử - Viễn thông
2	Lê Hữu Phú	106150194	15DT3	Điện tử - Viễn thông
3	Nguyễn Ngọc Quang	106150128	15DT2	Điện tử - Viễn thông

1. Tên đề tài đồ án: **Hệ thống nhắc nhở uống thuốc dựa trên nền tảng IOTs.**
2. Đề tài thuộc diện: ☐ Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện
3. Các số liệu và dữ liệu ban đầu:
 - Thông tin về các ngôn ngữ lập trình.
 - Thông tin về các giao thức.
 - Thông tin về các linh kiện điện tử, các module chức năng.
4. Nội dung các phân thuyết minh và tính toán:
.....
.....
.....
.....
.....
5. Họ tên người hướng dẫn: ThS. Hồ Viết Việt
6. Ngày giao nhiệm vụ đồ án:/...../2020
7. Ngày hoàn thành đồ án:/...../2020

Đà Nẵng, ngày ... tháng ... năm 2020.

Trưởng Bộ môn

Người hướng dẫn

ThS. Hồ Viết Việt

LỜI NÓI ĐẦU

Chúng tôi xin chân thành cảm ơn thầy Hồ Viết Việt và các thầy cô trong Khoa Điện tử - Viễn thông, trường Đại học Bách Khoa - Đại học Đà Nẵng đã truyền đạt những kiến thức quý báu trong những năm học vừa qua và nhất là tạo điều kiện thuận lợi cho chúng tôi học tập thực hiện đồ án tốt nghiệp này.

Để có được kết quả như ngày hôm nay, chúng tôi rất biết ơn gia đình bạn bè đã động viên, khích lệ, tạo mọi điều kiện thuận lợi nhất về vật chất lẫn tinh thần trong suốt quá trình học tập cũng như quá trình thực hiện đồ án tốt nghiệp.

Mặc dù đã cố gắng hoàn thành đồ án trong phạm vi và khả năng cho phép nhưng chắc chắn sẽ không tránh khỏi những thiếu sót. Chúng tôi rất mong nhận được sự thông cảm, góp ý và tận tình chỉ bảo của quý thầy cô và các bạn.

CAM ĐOAN

Kính gửi: Hội đồng bảo vệ Đồ án tốt nghiệp khoa Điện tử - Viễn thông, trường Đại học Bách Khoa – Đại học Đà Nẵng.

Tôi: TRẦN THẾ ĐẦU

LÊ HỮU PHÚ

NGUYỄN NGỌC QUANG

Sinh viên ngành Kỹ thuật máy tính, khoa Điện tử - Viễn thông, trường Đại học Bách Khoa – Đại học Đà Nẵng.

Chúng tôi xin cam đoan nội dung của đồ án này không phải là bản sao chép nào của bất cứ Đồ án hoặc công trình đã có từ trước. Nếu vi phạm, tôi xin chịu mọi hình thức kỷ luật của Khoa hoặc Nhà trường.

Đà Nẵng, ngày tháng năm 2020

Trần Thế Đầu

Lê Hữu Phú

Nguyễn Ngọc Quang

MỤC LỤC

NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN	1
NHẬN XÉT CỦA NGƯỜI PHẢN BIỆN	3
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP	5
LỜI NÓI ĐẦU.....	i
CAM ĐOAN.....	ii
MỤC LỤC	iii
DANH SÁCH CÁC BẢNG, HÌNH VẼ.....	v
DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT	vii
MỞ ĐẦU.....	1
CHƯƠNG 1: TỔNG QUAN HỆ THỐNG NHẮC NHỞ UỐNG THUỐC DỰA TRÊN NỀN TẢNG IOTS	2
1.1 Giới thiệu chương	2
1.2 Giải pháp	2
1.2.1 Sơ đồ hệ thống	2
1.2.2 Kiến trúc chức năng và vận hành hệ thống.....	2
1.2.3 Chức năng và lựa chọn linh kiện chính cho từng khối	4
1.3 Kết luận chương	5
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ TÌM HIỂU CÁC THÀNH PHẦN.....	6
2.1 Giới thiệu chương	6
2.2 Giao thức	6
2.2.1 Giao thức MQTT	6
2.2.2 Giao thức I2C.....	7
2.2.3 Giao thức HTTP.....	7
2.3 Ngôn ngữ lập trình	7
2.3.1 Ngôn ngữ HTML	7
2.3.2 Ngôn ngữ PHP	7
2.3.3 Ngôn ngữ Javascript	8
2.3.4 Ngôn ngữ Java	8
2.4 Giới thiệu tổng quan về Server	8
2.5 Giới thiệu tổng quan về linh kiện.....	8
2.5.1 Kit RF WiFi + Bluetooth ESP32	8
2.5.2 Màn hình LCD 1602	9

2.5.3 Mạch chuyển đổi I2C cho LCD.....	9
2.5.4 Module cảm biến nhiệt độ và độ ẩm DHT11	9
2.5.5 Còi chip 5V.....	10
2.5.6 Nguồn sử dụng.....	10
2.6 Kết luận chương.....	10
CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ PHẦN CỨNG, PHẦN MỀM	11
3.1 Giới thiệu chương	11
3.2 Thiết kế phần cứng.....	11
3.2.1 Khối vi điều khiển trung tâm.....	11
3.2.2 Khối điều khiển	12
3.2.3 Khối còi báo hiệu.....	12
3.2.4 Khối hiển thị	13
3.3 Thiết kế phần mềm.....	14
3.3.1 Khối vi điều khiển trung tâm.....	14
3.3.2 Database server.....	20
3.3.3 Khối web server.....	21
3.3.4 Khối ứng dụng trên điện thoại.....	22
3.4 Kết luận chương.....	28
CHƯƠNG 4: THI CÔNG, KIỂM TRA KẾT QUẢ VÀ ĐÁNH GIÁ HỆ THỐNG	29
4.1 Giới thiệu chương	29
4.2 Thi công mạch.....	29
4.2.1 Về mạch PCB	29
4.2.2 Mạch thực tế	30
4.3 Thi công mô hình	30
4.4 Kết quả thực nghiệm và đánh giá hệ thống.....	31
4.4.1 Kết quả thực nghiệm.....	31
4.4.2 Đánh giá hệ thống.....	32
4.5 Kết luận chương.....	32
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	33
TÀI LIỆU THAM KHẢO	34

DANH SÁCH CÁC BẢNG, HÌNH VẼ

Bảng 3.1 Bảng danh sách nhắc nhở	20
Bảng 3.2 Bảng độ ẩm	21
Bảng 3.3 Bảng nhiệt độ	21
Bảng 3.4 Các topic đăng ký và nhiệm vụ tương ứng	25
Hình 1.1 Mô hình tổng quan	2
Hình 1.2 Giảm đồ Usecase	3
Hình 1.3 Giảm đồ Activity	4
Hình 2.1 Kiến trúc mức cao của MQTT	6
Hình 2.2 Mô hình Publish-Subscribe	6
Hình 2.3 Kit RF WiFi + Bluetooth ESP32	8
Hình 2.4 Màn hình LCD 1602	9
Hình 2.5 Module I2C	9
Hình 2.6 Module DHT11	9
Hình 2.7 Còi chip 5V	10
Hình 2.8 Pin Lithium Battery 3.7V, 2200mah	10
Hình 3.1 Sơ đồ nguyên lý khối VĐK trung tâm	11
Hình 3.2 Sơ đồ nguyên lý khối điều khiển	12
Hình 3.3 Sơ đồ nguyên lý khối còi báo hiệu	13
Hình 3.4 Sơ đồ nguyên lý khối hiển thị	13
Hình 3.5 Sơ đồ thuật toán chương trình chính	14
Hình 3.6 Sơ đồ thuật toán chương trình khởi tạo hệ thống	15
Hình 3.7 Sơ đồ thuật toán chương trình hiển thị độ ẩm, nhiệt độ	16
Hình 3.8 Sơ đồ thuật toán chương trình cài đặt giờ	17
Hình 3.9 Sơ đồ thuật toán chương trình cài đặt phút	18
Hình 3.10 Sơ đồ thuật toán chương trình nhắc nhở	19
Hình 3.11 Sơ đồ thuật toán chương trình tìm hộp thuốc	20
Hình 3.12 Sơ đồ thuật toán chương trình của web server	21
Hình 3.13 Giao diện chính	22
Hình 3.14 Hiển thị thông tin về độ ẩm và nhiệt độ	22
Hình 3.15 Sơ đồ thuật toán ứng dụng trên điện thoại (1)	23
Hình 3.16 Sơ đồ thuật toán ứng dụng trên điện thoại (2)	24

Hình 3.17 Giao diện quản lý chung	26
Hình 3.18 Giao diện khi muốn thêm một báo thức mới	27
Hình 3.19 Giao diện khi đến giờ uống thuốc.....	28
Hình 4.1 Mạch xử lý trung tâm	29
Hình 4.2 Mạch xử lý trung tâm thực tế.....	30
Hình 4.3 Mô hình thực tế mặt trước và sau	30
Hình 4.4 Giao diện mặc định	31
Hình 4.5 Giao diện cài đặt giờ.....	31
Hình 4.6 Giao diện cài đặt phút	31
Hình 4.7 Giao diện nhắc nhở	32
Hình 4.8 Giao diện tìm hộp thuốc.....	32

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

VĐK	Vi điều khiển
GPIO	General Purpose Input/Output
UART	Universal Asynchronous Receiver-Transmitter
HTTP	HyperText Transfer Protocol
PHP	Hypertext Preprocessor
HTML	HyperText Markup Language
SQL	Structured Query Language
MQTT	Message Queuing Telemetry Transport
I2C	Inter-Integrated Circuit
RF	Radio Frequency
LCD	Liquid-Crystal Display
WiFi	Wireless Fidelity
App	Application
PCB	Printed Circuit Board
LED	Light Emitting Diode
DC	Direct Current
I/O	Input/Output
BJT	Bipolar Junction Transistor
GND	Ground
IC	Integrated Circuit

MỞ ĐẦU

Mục đích thực hiện đề tài là xây dựng một hệ thống nhắc nhở uống thuốc đa nền tảng với xương sống là nền tảng IOTs.

Mục tiêu đề tài là phục vụ cho người sử dụng một cách thuận tiện nhất, nơi mà người dùng có thể thiết lập các nhắc nhở trên bất cứ thiết bị nào có trong tay mà không cần quan tâm đến công việc đồng bộ dữ liệu.

Phạm vi của đề tài là khu vực nhà ở hoặc bệnh viện có thể kết nối được WiFi. Đối tượng hướng tới là người sử dụng cá nhân có nhu cầu được nhắc nhở uống thuốc.

Cấu trúc của đồ án này gồm có bốn chương:

Chương 1: Tổng quan hệ thống nhắc nhở uống thuốc dựa trên nền tảng IOTs.

Chương 2: Các giao thức, ngôn ngữ và linh kiện.

Chương 3: Thiết kế phần cứng, phần mềm.

Chương 4: Thi công, kiểm tra kết quả, đánh giá hệ thống.

Nhóm thực hiện đồ án xin chân thành cảm ơn ThS. Hồ Viết Việt đã tận tình hướng dẫn, hỗ trợ về kiến thức và hướng hoàn thiện đồ án trong suốt quá trình nghiên cứu thực hiện đồ án.

Nhóm xin chân thành cảm ơn!

CHƯƠNG 1: TỔNG QUAN HỆ THỐNG NHẮC NHỞ UỐNG THUỐC DỰA TRÊN NỀN TẢNG IOTS

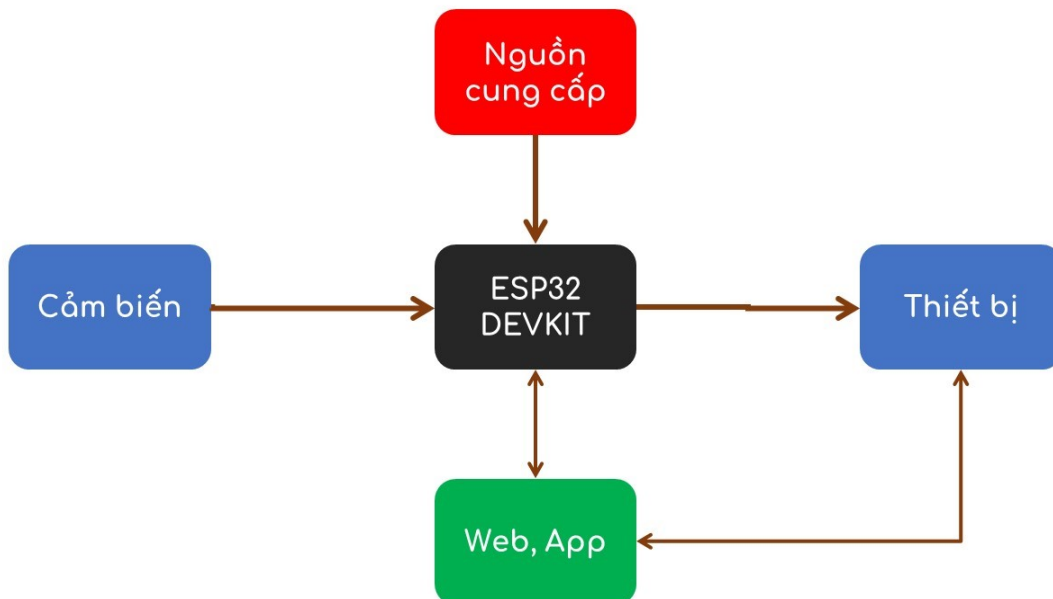
1.1 Giới thiệu chương

Xuất phát từ thực tế hiện nay trên thị trường chỉ là những sản phẩm hộp thuốc đơn giản, không có thông báo tự động hay nhắc nhở người dùng uống thuốc. Do đó, chúng tôi đã nảy sinh ra ý tưởng tích hợp nền tảng IOTs vào hộp thuốc, nhằm đưa ra một sản phẩm mới có khả năng tự động nhắc nhở, thông báo cho người dùng uống thuốc đúng giờ. Sản phẩm này mang lại tính tiện dụng và thân thiện với người dùng.

Chương này chúng tôi sẽ trình bày tổng quan về hệ thống “***Nhắc nhở uống thuốc dựa trên nền tảng IOTs***”, sơ đồ tổng quát của hệ thống, các giản đồ usecase và activity, cũng như chức năng và linh kiện cho từng khối của hệ thống nêu trên.

1.2 Giải pháp

1.2.1 Sơ đồ hệ thống



Hình 1.1 Mô hình tổng quan

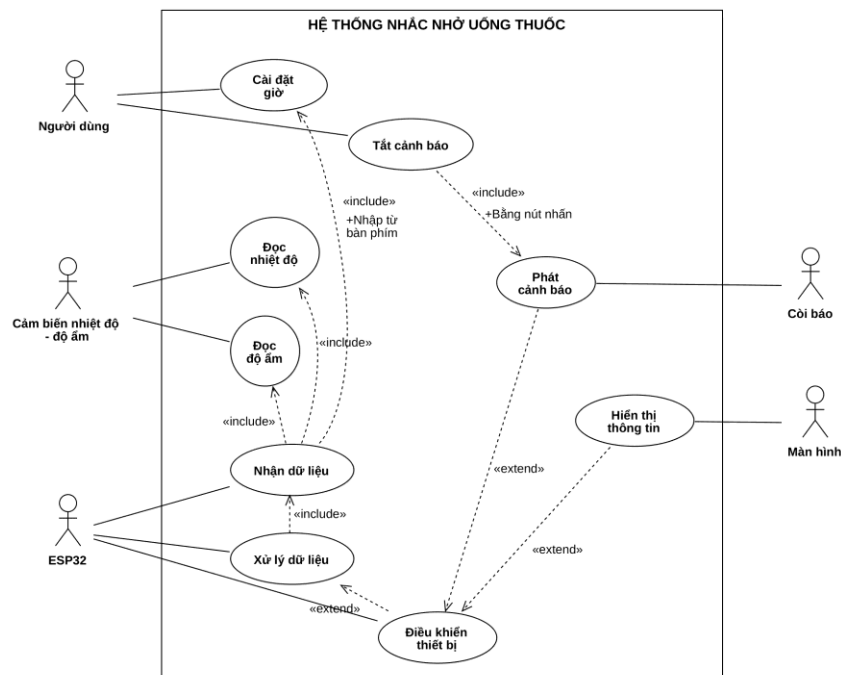
1.2.2 Kiến trúc chức năng và vận hành hệ thống

❖ Giản đồ Use Case

Hình 1.1 giới thiệu về kiến trúc chức năng và cách vận hành của “Hệ thống nhắc nhở uống thuốc”. Nhằm bắt được các yêu cầu từ hệ thống, mô tả được sự liên quan của

người dùng từ bên ngoài tác động như thế nào vào hệ thống, chỉ ra các đối tượng của hệ thống và các sự kiện diễn ra trong quá trình hoạt động.

Use Case::Medication Reminder System



Hình 1.2 Giải đồ Usecase

Khi hệ thống bắt đầu hoạt động:

Cảm biến nhiệt độ - độ ẩm: Đọc thông tin nhiệt độ, độ ẩm của môi trường bằng cảm biến chuyên dụng.

Người dùng: Thực hiện thao tác cài đặt giờ nhắc nhở uống thuốc thông qua các phím nhấn chức năng. Cũng như tắt cảnh báo khi cần.

ESP32: Có nhiệm vụ nhận dữ liệu từ cảm biến và người dùng để tiến hành xử lý dữ liệu nhận được, từ đó điều khiển các thiết bị.

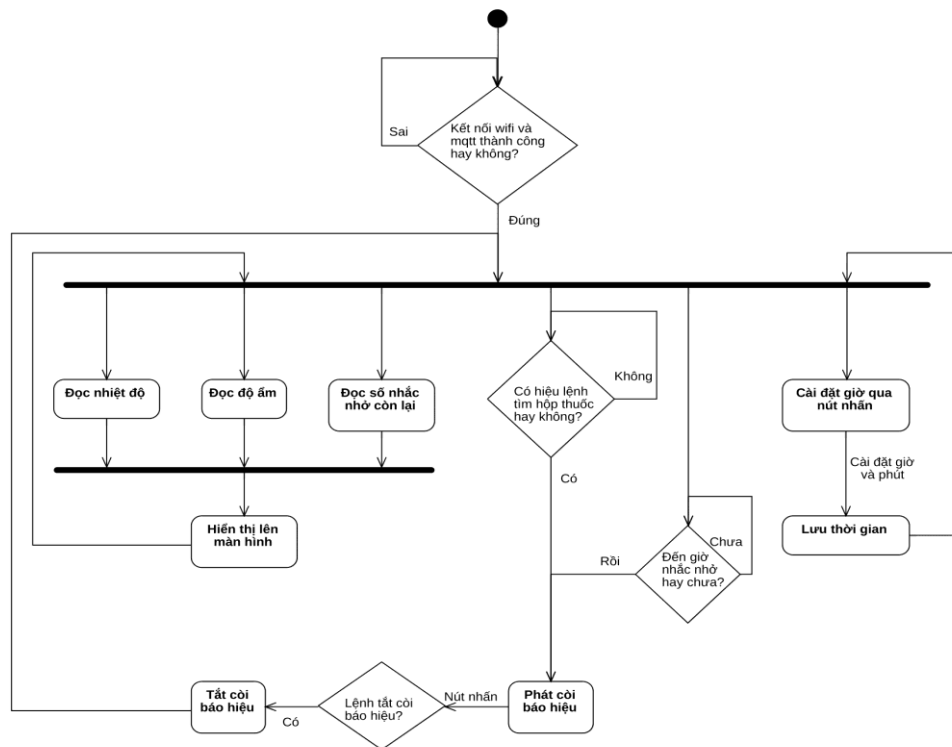
Màn hình: Hiển thị thông tin về nhiệt độ, độ ẩm, số lần nhắc nhở còn lại.

Còi báo: Phát hoặc tắt cảnh báo nếu có hiệu lệnh điều khiển.

❖ **Giải đồ Activity**

Hình 1.2 mô tả quy trình vận hành của hệ thống.

Activity1::Medication Reminder System



Hình 1.3 Giải đồ Activity

Giải thích giải đồ: Khi hệ thống khởi động, VDK sẽ kiểm tra đã kết nối mạng WiFi và MQTT hay chưa. Nếu đã kết nối thành công thì sẽ tiến đến các tiến trình đọc thông số về nhiệt độ, độ ẩm và số nhắc nhở còn lại sau đó hiển thị lên màn hình. Bên cạnh đó, người dùng có thể cài đặt giờ để nhắc nhở uống thuốc theo giờ được cài, hệ thống sau khi nhận được thông tin về thời gian sẽ lưu lại nội dung. Ngoài ra, nếu có hiệu lệnh tìm hộp thuốc từ app hoặc đã đến giờ nhắc nhở (được cài đặt trước đó) thì hộp thuốc sẽ phát ra còi cảnh báo đồng thời hiển thị nội dung trên màn hình để người dùng nhận biết, người dùng có thể tắt còi báo thông qua nút nhấn.

1.2.3 Chức năng và lựa chọn linh kiện chính cho từng khối

❖ **Khối vi điều khiển trung tâm:**

Chức năng: VDK trung tâm để kết nối các thiết bị và xử lý dữ liệu. Đồng thời truyền nhận dữ liệu giữa VDK trung tâm - Server thông qua WiFi và giao thức MQTT.

Linh kiện: Kit RF WiFi + Bluetooth ESP32.

❖ **Khối Server database:**

Chức năng: Đóng vai trò là máy chủ server xử lý các dữ liệu từ khối VDK trung tâm gửi lên.

Linh kiện: Laptop Dell.

❖ **Khởi còi báo hiệu:**

Chức năng: Báo hiệu khi có lệnh tìm hộp thuốc hoặc đã đến giờ nhắc nhở người dùng uống thuốc.

Linh kiện: Còi chip 5V.

❖ **Khởi nút nhấn:**

Chức năng: Thiết lập thời gian, tắt cảnh báo. Bao gồm có 4 nút nhấn.

Linh kiện: Nút nhấn 4 chân cao.

❖ **Khởi cảm biến:**

Chức năng: Thu thập thông tin của môi trường để gửi về khối VĐK trung tâm.

Linh kiện: Module cảm biến nhiệt độ - độ ẩm DHT11.

❖ **Khởi hiển thị:**

Chức năng: Hiển thị các thông tin cần thiết lên màn hình.

Linh kiện: LCD 1602 và module giao tiếp I2C.

❖ **Khởi nguồn:**

Chức năng: Nguồn 7.4VDC cung cấp cho VĐK trung tâm, khởi hiển thị.

Linh kiện: Pin sạc 18650 2200mAh.

1.3 Kết luận chương

Trên đây chúng tôi đã trình bày một các tổng quan về hệ thống, giải pháp cho vấn đề đã nêu ra. Ở chương sau chúng tôi sẽ đi vào chi tiết của hệ thống, các thiết bị được sử dụng.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ TÌM HIỂU CÁC THÀNH PHẦN

2.1 Giới thiệu chương

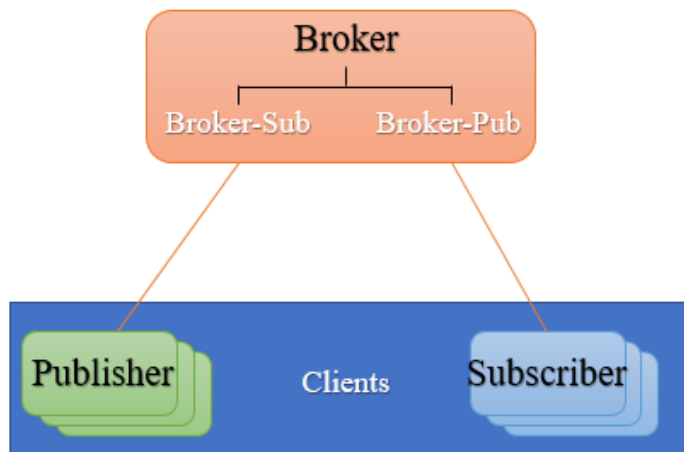
Sau khi trình bày tổng quan hệ thống, chương này nhóm chúng tôi sẽ đi vào chi tiết, giới thiệu, nêu các giao thức, ngôn ngữ lập trình có sử dụng trong hệ thống. Giới thiệu và biểu diễn sơ đồ khối của cảm biến và điều khiển, trình bày các chức năng cảm biến và điều khiển.

2.2 Giao thức

2.2.1 Giao thức MQTT

❖ **Khái niệm:**

MQTT là từ viết tắt của Message Queue Telemetry Transport. Đây là một giao thức truyền thông điệp (message) theo mô hình xuất bản/đăng ký (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.



Hình 2.1 Kiến trúc mức cao của MQTT

❖ **Kiến trúc mức cao của MQTT gồm 2 phần chính là Broker và Clients:**



Hình 2.2 Mô hình Publish-Subscribe

❖ **Publish, subscribe:**

Mô hình publish/subscribe là một mô hình được sử dụng thay thế cho mô hình khách – chủ truyền thống. Trong mô hình giao thức MQTT, các client có thể trao đổi các gói tin với nhau bằng cách xuất bản gói tin (gọi là publisher) hoặc đăng ký nhận gói tin (gọi là subscriber), tuy nhiên các client không hề biết đến sự tồn tại của nhau. Có một thành phần thứ ba, đóng vai trò như một server trung tâm (được gọi là broker) là nơi mà các client có thể biết đến và đóng vai trò trung gian có khả năng lọc các gói tin nhận từ các publisher để gửi đến các subscriber phù hợp.

❖ **Broker:**

Đối tác của các MQTT client là MQTT broker, đây chính là trái tim của bất kỳ một giao thức tổ chức theo mô hình publish/subscribe. Tùy thuộc vào các công việc cụ thể, một broker có thể xử lý hàng ngàn kết nối đồng thời từ các client. MQTT broker chịu trách nhiệm chính trong việc nhận tất cả các gói tin, lọc gói tin và quyết định gửi gói tin đến các client đã đăng ký nhận các gói tin đó.

2.2.2 Giao thức I2C

I2C là từ viết tắt Tiếng Anh của Inter-Integrated Circuit. Đây là giao thức truyền thông nối tiếp đồng bộ được phát triển bởi Philips Semiconductors và phổ biến nhất hiện nay.

I2C được sử dụng rộng rãi trong việc kết nối nhiều IC lại với nhau, hay kết nối giữa IC và các ngoại vi với tốc độ thấp. Giao thức I2C sử dụng 2 dây để kết nối là SCL (Serial Clock) và SDA (Serial Data). Trong đó, dây SCL có tác dụng đồng bộ hóa giữa các thiết bị khi truyền dữ liệu, còn SDA là dây truyền dữ liệu.

2.2.3 Giao thức HTTP

HTTP (HyperText Transfer Protocol) là giao thức truyền tải siêu văn bản được sử dụng trong www dùng để truyền tải dữ liệu giữa Server đến các trình duyệt Web và ngược lại. Giao thức này sử dụng cổng 80 (port 80) là chủ yếu.

Hay có thể hiểu khi gõ 1 địa chỉ vào trình duyệt Web, lúc này trình duyệt Web sẽ gửi 1 yêu cầu qua giao thức HTTP đến Web server. Web server sẽ nhận yêu cầu này và trả lại kết quả cho trình duyệt Web.

GET và POST là hai phương thức của giao thức HTTP.

2.3 Ngôn ngữ lập trình

2.3.1 Ngôn ngữ HTML

HTML là viết tắt của Hyper Text Markup Language. Nó dùng để định dạng bố cục, các thuộc tính liên quan đến cách hiển thị của một đoạn text và được hiển thị trên một chương trình đặc biệt ta gọi là trình duyệt. Trong đồ án này tôi sử dụng html để tạo bố cục văn bản, chèn đường dẫn.

2.3.2 Ngôn ngữ PHP

PHP (viết tắt hồi quy "PHP: Hypertext Preprocessor") là một ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yếu được dùng để phát triển các ứng dụng viết cho máy chủ, mã nguồn mở, dùng cho mục đích tổng quát. Nó rất thích hợp với web và có thể dễ dàng nhúng vào trang HTML. Trong đồ án này tôi sử dụng php để kết nối server với cơ sở dữ liệu Mysql. MySQL là một hệ quản trị cơ sở dữ liệu dùng để lưu trữ dữ liệu.

2.3.3 Ngôn ngữ Javascript

Javascript là một ngôn ngữ lập trình kịch bản dựa vào đối tượng phát triển có sẵn hoặc tự định nghĩa ra, javascript được sử dụng rộng rãi trong các ứng dụng Website. Javascript được hỗ trợ hầu như trên tất cả các trình duyệt. Trong đồ án này javascript được dùng để tải về một phần của trang web mà không cần phải tải lại trang sử dụng kỹ thuật AJAX.

2.3.4 Ngôn ngữ Java

Java là một ngôn ngữ lập trình, được phát triển bởi Sun Microsystem vào năm 1995, là ngôn ngữ kế thừa trực tiếp từ C/C++ và là một ngôn ngữ lập trình hướng đối tượng.

2.4 Giới thiệu tổng quan về Server

Máy chủ (hay server) được dùng để xử lý các gói tin hay truy cập được gửi từ máy khách (client) thông qua giao thức MQTT hoặc HTTP. Các truy cập HTTP này thường được gửi từ các chương trình duyệt web trên máy tính hoặc ứng dụng điện thoại hệ điều hành android.

Ngoài ra, server còn thực thi chương trình xử lý tính toán thời gian do người dùng cài đặt được lưu trong cơ sở dữ liệu, sử dụng ngôn ngữ Python. Python là một ngôn ngữ lập trình bậc cao, đa mục đích, cú pháp được đơn giản hóa, dễ đọc hiểu. Hiện nay, Python là một trong những ngôn ngữ lập trình mới và phổ biến.

2.5 Giới thiệu tổng quan về linh kiện

2.5.1 Kit RF WiFi + Bluetooth ESP32

- Module trung tâm: WiFi BLE SoC ESP32 ESP-WROOM-32.
- Nguồn sử dụng: 5VDC từ cổng micro USB hoặc 7V-12V (khuyến dùng) vào chân Vin trên Kit.
- Tích hợp mạch nạp và giao tiếp UART CP2102.
- Ra chân đầy đủ module ESP32, chuẩn chân cắm 2.54mm.
- Tích hợp LED Status, nút BOOT và ENABLE.
- Kích thước: 28.33 x 51.45mm.



Hình 2.3 Kit RF WiFi + Bluetooth ESP32

2.5.2 Màn hình LCD 1602

- Điện áp: -0.3V – 7V.
- Hoạt động ổn định: 2.7V – 5.5V.
- Điện áp ra mức cao: >2.4V.
- Điện áp ra mức thấp: <0.4V.
- Dòng điện cấp nguồn: 350 μ A - 600 μ A.
- Nhiệt độ hoạt động: -30° - 75°C.
- Hiển thị 2 dòng, mỗi dòng 16 ký tự.
- Chữ đen, nền xanh lá.
- Kích thước: 80 x 36 x 12.5mm.



Hình 2.4 Màn hình LCD 1602

2.5.3 Mạch chuyển đổi I2C cho LCD

- Điện áp hoạt động: 3 – 6V.
- Chuẩn giao tiếp: I2C.
- Dải địa chỉ I2C: 0x20 đến 0x27.
- Địa chỉ mặc định: 0x27.
- Có điều chỉnh độ tương phản bằng vi trở.
- Kích thước: 41.5 x 19 x 15.3mm.



Hình 2.5 Module I2C

2.5.4 Module cảm biến nhiệt độ và độ ẩm DHT11

- Điện áp hoạt động: 3 – 5V.
- Dải độ ẩm hoạt động: 20 – 90%, sai số $\pm 5\%$.
- Dải nhiệt độ hoạt động: 0 – 50°C, sai số $\pm 2^\circ\text{C}$.
- Khoảng cách truyền tối đa: 20m.
- Tần số lấy mẫu tối đa: 1Hz (1 giây/ 1 lần).
- Kích thước: 28 x 12 x 10mm.



Hình 2.6 Module DHT11

2.5.5 Còi chip 5V

- Điện áp hoạt động: 3.5 – 5.5V.
- Dòng điện tiêu thụ: < 25mA.
- Tần số cộng hưởng: 2300Hz \pm 500Hz.
- Biên độ âm thanh: > 80dB.
- Nhiệt độ hoạt động: -20 – 70°C.
- Kích thước: Đường kính 12mm, cao 9.7mm.



Hình 2.7 Còi chip 5V

2.5.6 Nguồn sử dụng

- Điện áp: 3.7V (Khi đầy pin lên đến 4.2V).
- Pin được sử dụng nhiều lần.
- Kiểu: 18650.
- Dung lượng: 2200mAh.
- Khuyến cáo: Không sử dụng pin < 2.5V.
- Kích thước: 18 x 65mm.
- Giá: 35,000 VNĐ/ viên.



Hình 2.8 Pin Lithium Battery 3.7V, 2200mah

2.6 Kết luận chương

Chương này chúng tôi đã trình bày chi tiết về các giao thức thiết bị sử dụng trong hệ thống, sơ đồ và chức năng của bộ phận điều khiển thiết bị điện. Chương tiếp theo chúng tôi sẽ trình bày về thiết lập database và xây dựng website để quản lý hệ thống.

CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ PHẦN CỨNG, PHẦN MỀM

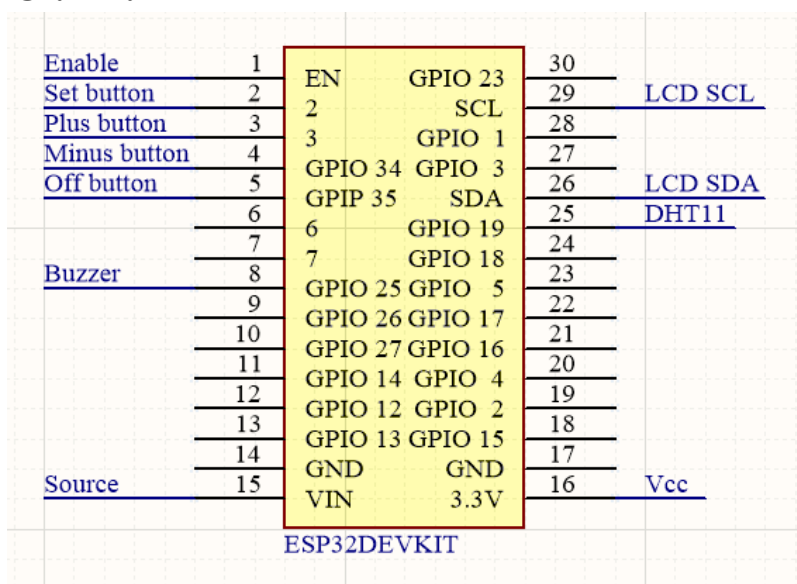
3.1 Giới thiệu chương

Chương này nhóm chúng tôi giới thiệu về phần cứng của hệ thống, lưu đồ và các thành phần liên quan.

3.2 Thiết kế phần cứng

3.2.1 Khối vi điều khiển trung tâm

❖ Sơ đồ nguyên lý:



Hình 3.1 Sơ đồ nguyên lý khối VDK trung tâm

Như thông tin đã đề cập ở chương 2, nguồn cấp vào ở chân Vin cho Kit ESP32 là 7 - 12V, ở đây chúng tôi dùng nguồn 7.4VDC.

Chân EN của VDK sử dụng để nạp chương trình tự động vào VDK.

Chân số 2 (GPIO36), 3 (GPIO39), 4 (GPIO34), 5 (GPIO35) lần lượt kết nối với các nút nhấn Set, Plus, Minus, Off để sử dụng cho khối điều khiển.

Chân số 8 (GPIO25) sử dụng kết nối tới khối còi báo hiệu.

Chân số 16 có mức điện áp đầu ra 3.3V, được sử dụng làm nguồn chung, cung cấp điện áp cho các khối điều khiển, khối còi báo hiệu, khối cảm biến hoạt động.

Chân số 26 (SDA) và 29 (SCL) kết nối tới khối hiển thị để truyền tải thông tin cần hiển thị lên màn hình.

❖ Những lưu ý:

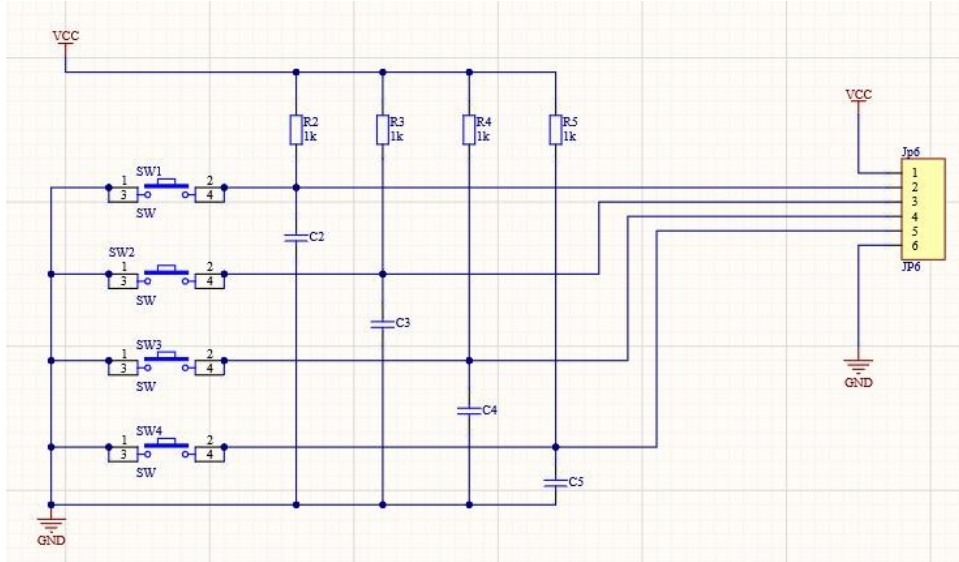
GPIO34, 35, 36, 39 chỉ dùng được chức năng INPUT, không dùng được chức năng OUTPUT.

GPIO1, 3 được kết nối đến bộ UART0 và ESP32 cũng nạp chương trình thông qua bộ UART này, nên tránh sử dụng 2 chân này cho chức năng I/O.

GPIO0, 2, 4, 5, 12, 15 đã được định sẵn mức logic bên trong module ESP32 để phục vụ quá trình nạp chương trình, nên tránh sử dụng các chân GPIO này.

3.2.2 Khối điều khiển

❖ Sơ đồ nguyên lý:



Hình 3.2 Sơ đồ nguyên lý khối điều khiển

Có 4 nút nhấn, mỗi nút nhấn kết nối tới 4 chân của VĐK. Đồng thời, mỗi nút nhấn đều có trở kéo lên nguồn và tụ để chống nhiễu cho nút nhấn.

Nút nhấn 1: Nút Set, có chức năng đi vào giao diện cài đặt thời gian, xác nhận thời gian, lưu thời gian được cài đặt.

Nút nhấn 2: Nút Plus, có chức năng tăng thời gian lên 1 đơn vị. Nếu là giờ thì tăng lên 1 giờ, nếu là phút thì tăng lên 1 phút. Trường hợp 23 giờ thì khi nhấn nút Plus sẽ về lại 0 giờ hoặc 59 phút thì sẽ về 0 phút.

Nút nhấn 3: Nút Minus, tương tự như nút Plus nhưng là trừ đi 1 đơn vị. Trường hợp 0 giờ thì khi nhấn nút Minus sẽ thành 23 giờ hoặc 0 phút thì sẽ thành 59 phút.

Nút nhấn 4: Nút Off, có chức năng tắt các cảnh báo khi đang có cảnh báo.

3.2.3 Khối còi báo hiệu

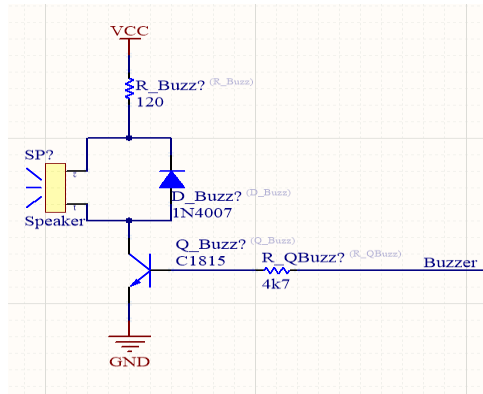
❖ Sơ đồ nguyên lý:

VĐK trung tâm dùng chân Buzzer để điều khiển tắt/mở BJT Q_Buzz.

Q_Buzz hoạt động ở chế độ bão hòa.

Nếu chân Buzzer ở mức cao (3.3V), Q_Buzz được mở, có dòng điện đổ từ nguồn Vcc (3.3V) qua loa, qua Q_Buzz về GND. Khi đó, loa phát âm thanh.

Nếu chân Buzzer ở mức thấp (0V), BJT Q_Buzz bị tắt, không có dòng chạy qua loa. Khi đó loa tắt, không có âm thanh.



Hình 3.3 Sơ đồ nguyên lý khối còi báo hiệu

Diode D_Buzz được mắc song song với loa để tránh dòng điện cảm ứng do loa tạo ra lúc hoạt động chạy ngược về BJT làm hỏng BJT.

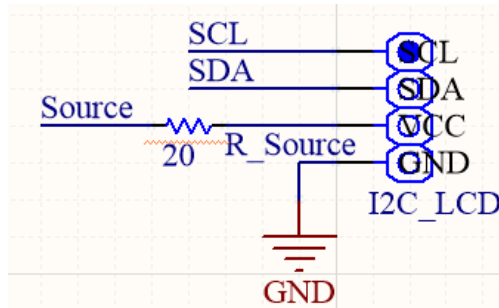
3.2.4 Khối hiển thị

❖ Sơ đồ nguyên lý:

Kết nối màn hình LCD với VĐK thông qua module giao tiếp I2C để tiết kiệm chân VĐK. Chỉ cần sử dụng 2 chân tín hiệu là SDA và SCL.

Chân SDA và SCL trên module I2C kết nối tương ứng đến chân SDA và SCL ở khối VĐK trung tâm. Hai chân này có chức năng nhận dữ liệu từ VĐK để giải mã và hiển thị lên màn hình LCD.

Chân Vcc ở module được nối vào khối nguồn qua một điện trở nhỏ.

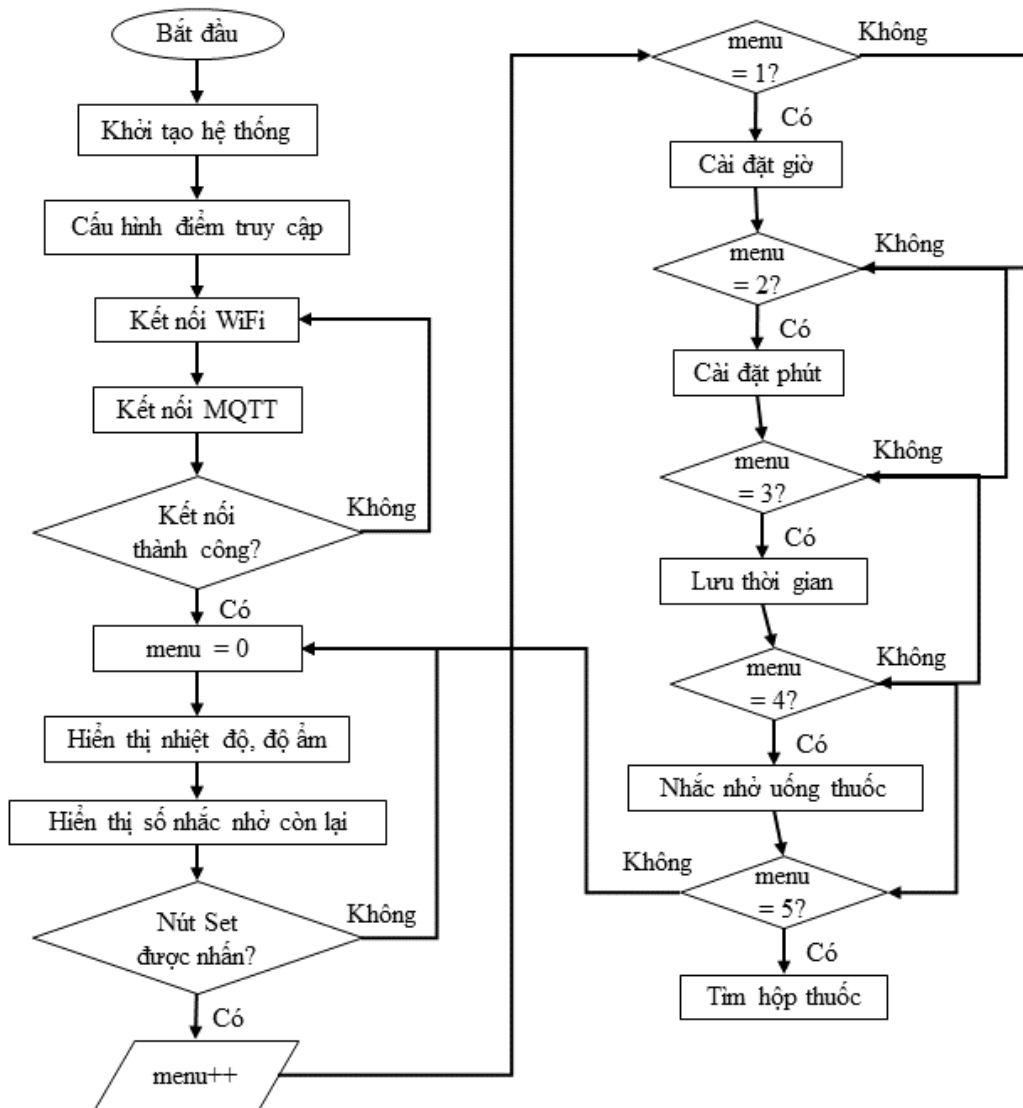


Hình 3.4 Sơ đồ nguyên lý khối hiển thị

3.3 Thiết kế phần mềm

3.3.1 Khối vi điều khiển trung tâm

❖ Chương trình chính:



Hình 3.5 Sơ đồ thuật toán chương trình chính

❖ Giải thích hình ảnh:

Khi được cấp nguồn, VDK trung tâm tiến hành khởi tạo hệ thống mặc định ban đầu, sau đó khởi tạo điểm truy cập để kết nối WiFi và MQTT, nếu kết nối không thành công thì sẽ kết nối lại đến lúc thành công.

Khi menu = 0, chương trình hiển thị nhiệt độ - độ ẩm và số nhắc nhở còn lại được thực thi.

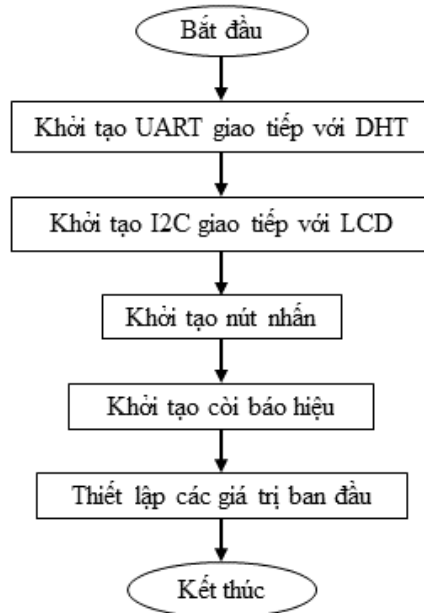
Nếu nút Set được nhấn thì tăng giá trị của menu lên 1 đơn vị.

Kiểm tra xem menu bằng bao nhiêu, nếu menu = 1 thì chương trình cài đặt giờ được thực thi, nếu menu = 2 thì chương trình cài đặt phút được thực thi, nếu menu = 3 thì chương trình lưu thời gian được thực thi, nếu menu = 4 thì chương trình phát cảnh

báo nhắc nhở uống thuốc được thực thi và cuối cùng nếu menu = 5 thì chương trình tìm hộp thuốc được thực thi. Nếu không, trở về lại chương trình menu = 0.

❖ **Chương trình con:**

Thuật toán chương trình con khởi tạo hệ thống:

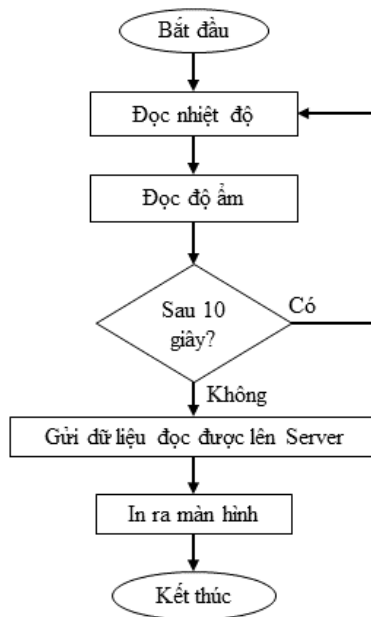


Hình 3.6 Sơ đồ thuật toán chương trình khởi tạo hệ thống

❖ **Giải thích hình ảnh:**

Hình 3.6 mô phỏng thuật toán khởi tạo hệ thống. VĐK sẽ cấu hình các chân GPIO theo chuẩn truyền thông nối tiếp UART để giao tiếp với các ngoại vi (DHT, LCD), chân vào/ra cho các nút nhấn và còi đi vào hoạt động ổn định. Tiếp đến, VĐK sẽ thiết lập các giá trị mặc định ban đầu.

Thuật toán hiển thị nhiệt độ, độ ẩm:

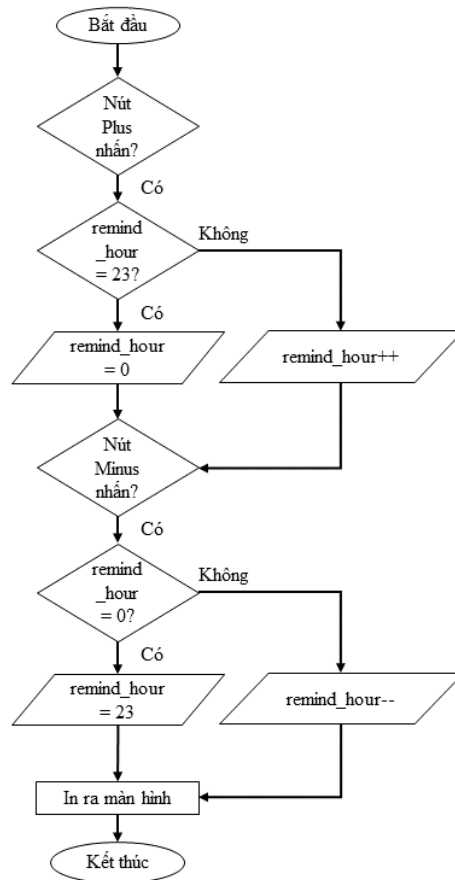


Hình 3.7 Sơ đồ thuật toán chương trình hiển thị độ ẩm, nhiệt độ

❖ **Giải thích hình ảnh:**

Hình 3.7 là thuật toán đọc và hiển thị nhiệt độ, độ ẩm lên màn hình LCD. Cảm biến DHT11 đọc thông tin nhiệt độ và độ ẩm sau đó gửi về cho VĐK, cứ sau 10 giây đọc một lần. Thông tin về nhiệt độ và độ ẩm được VĐK gửi lên Server và in ra màn hình LCD.

Thuật toán cài đặt giờ:



Hình 3.8 Sơ đồ thuật toán chương trình cài đặt giờ

❖ **Giải thích hình ảnh:**

Hình 3.8 là thuật toán cài đặt giờ (khi menu = 1).

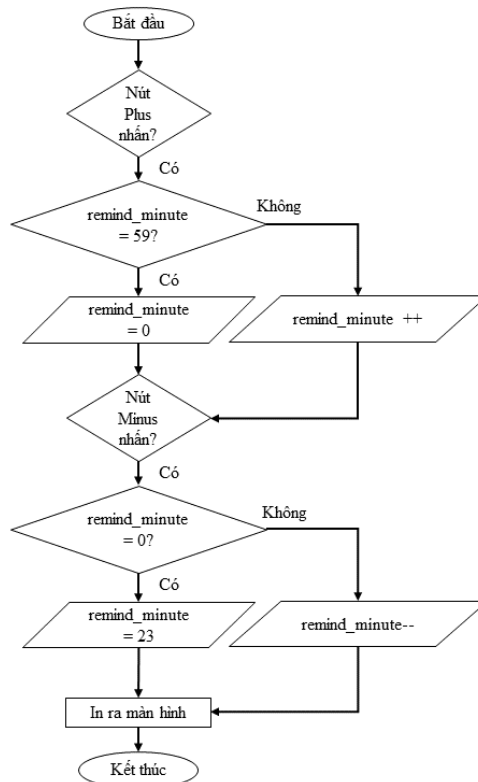
Khi vào tiến trình cài đặt giờ (menu = 1):

Nếu nút Plus được nhấn thì tăng biến remind_hour (là biến lưu giá trị giờ) lên 1 đơn vị (nếu remind_hour đang có giá trị 23 thì khi nhấn nút Plus sẽ về 0).

Ngược lại, nếu nút Minus được nhấn thì giảm biến remind_hour đi 1 đơn vị (nếu remind_hour đang có giá trị 0 thì khi nhấn nút Plus sẽ về 23).

Sau đó, in giá trị của biến remind_hour ra màn hình LCD.

Thuật toán cài đặt phút:



Hình 3.9 Sơ đồ thuật toán chương trình cài đặt phút

❖ **Giải thích hình ảnh:**

Hình 3.9 là thuật toán chương trình con cài đặt phút.

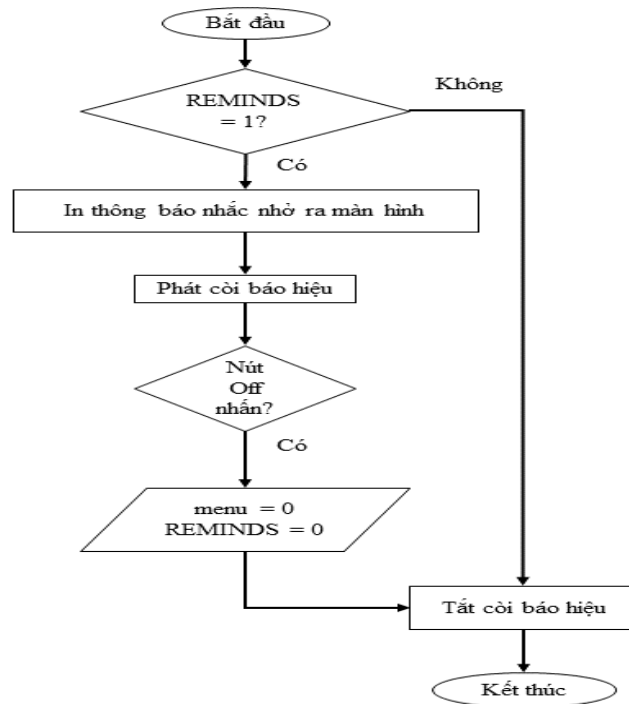
Cũng tương tự như chương trình cài đặt giờ. Khi menu = 2 thì chương trình con cài đặt phút được thực thi:

Nếu nút Plus được nhấn thì tăng biến remind_minute (là biến lưu giá trị phút) lên 1 đơn vị (nếu remind_minute đang có giá trị 59 thì khi nhấn nút Plus sẽ về 0).

Ngược lại, nếu nút Minus được nhấn thì giảm biến remind_minute đi 1 đơn vị (nếu remind_minute đang có giá trị 0 thì khi nhấn nút Plus sẽ về 59).

Sau đó, in giá trị của biến remind_minute ra màn hình LCD.

Thuật toán nhắc nhở uống thuốc:



Hình 3.10 Sơ đồ thuật toán chương trình nhắc nhở

❖ **Giải thích hình ảnh:**

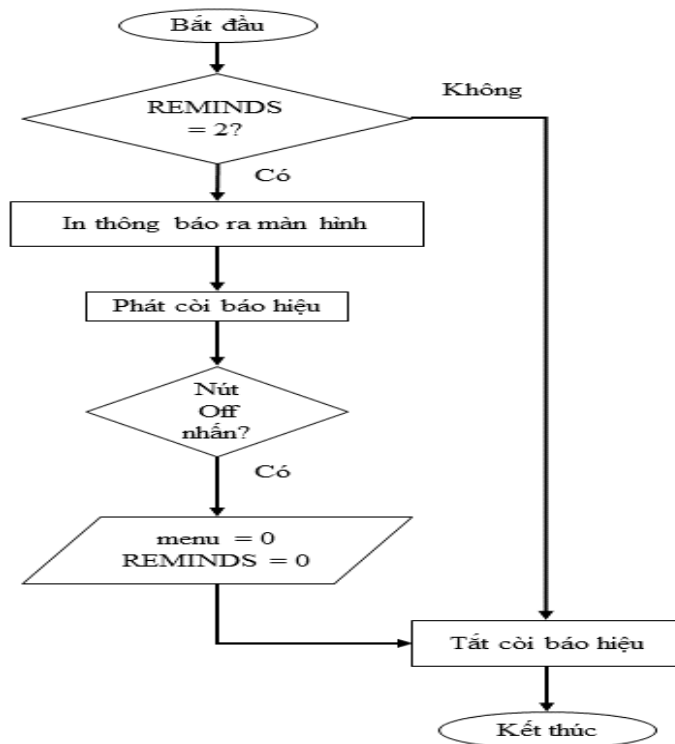
Hình 3.10 là thuật toán chương trình con nhắc nhở uống thuốc theo thời gian đã cài đặt.

VĐK sẽ tiếp nhận giá trị biến REMINDS từ Server thông qua giao thức MQTT và kết nối WiFi.

Nếu REMINDS = 1 thì sẽ in thông báo nhắc nhở uống thuốc “Time to take medicines!” lên màn hình LCD, đồng thời phát cảnh báo bằng còi báo hiệu.

Bên cạnh đó, VĐK cũng sẽ lắng nghe xem nút nhấn Off có được nhấn hay không, nếu có thì sẽ tắt trạng thái nhắc nhở.

Thuật toán tìm hộp thuốc:



Hình 3.11 Sơ đồ thuật toán chương trình tìm hộp thuốc

❖ **Giải thích hình ảnh:**

Hình 3.11 là thuật toán chương trình con tìm vị trí hộp thuốc.

VĐK sẽ tiếp nhận giá trị biến REMINDS từ Server thông qua giao thức MQTT và kết nối WiFi.

Nếu REMINDS = 2 thì chương trình tìm hộp thuốc được thực thi và sẽ in dòng tin nhắn “I’m here!” lên màn hình LCD, đồng thời phát cảnh báo bằng còi báo hiệu.

Bên cạnh đó, VĐK cũng sẽ lắng nghe xem nút nhấn Off có được nhấn hay không, nếu có thì sẽ tắt trạng thái báo hiệu.

3.3.2 Database server

Database server là nơi lưu trữ lại dữ liệu của toàn bộ hệ thống, mục đích của việc sử dụng này là để đồng bộ giữa các client với nhau với chỉ một nguồn dữ liệu duy nhất.

❖ **Cấu trúc các bảng trong cơ sở dữ liệu “day”**

Bảng 3.1 Bảng danh sách nhắc nhở

Số thứ tự	Tên trường	Giải thích	Kiểu	Ghi chú
1	Id	Số thứ tự	Int(11)	Khóa chính
2	Hour	Giờ	Int(255)	
3	Minute	Phút	Int(255)	
4	Seconds	Giây	Int(255)	

Bảng 3.2 Bảng độ ẩm

Số thứ tự	Tên trường	Giải thích	Kiểu	Ghi chú
1	Id	Số thứ tự	Int(11)	Khóa chính
2	Humidity	Độ ẩm	Int(255)	
3	Time	Thời gian	Timestamp	

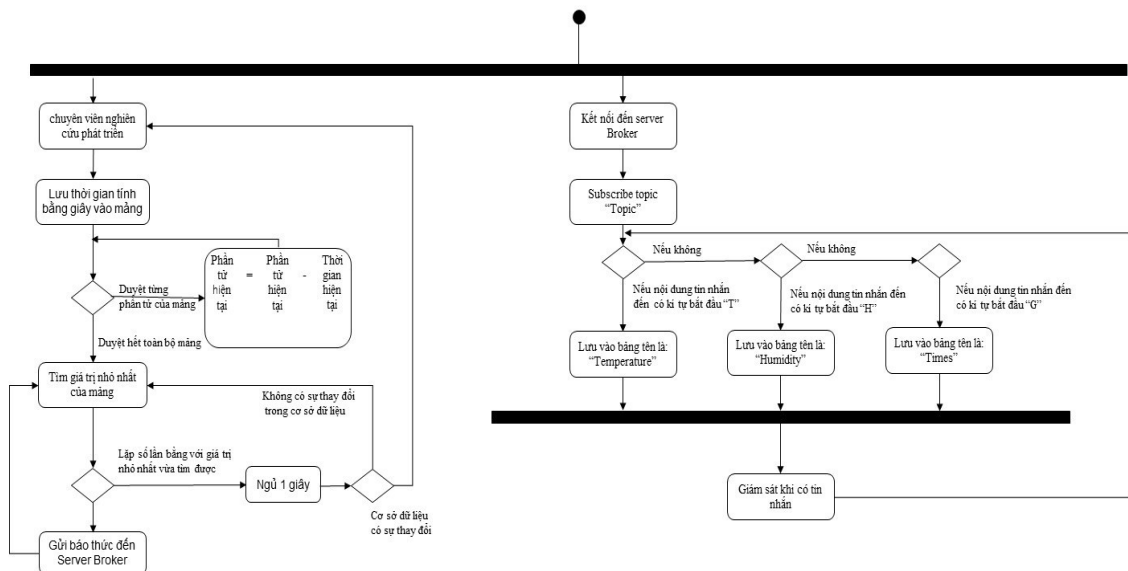
Bảng 3.3 Bảng nhiệt độ

Số thứ tự	Tên trường	Giải thích	Kiểu	Ghi chú
1	Id	Số thứ tự	Int(11)	Khóa chính
2	Temperature	Nhiệt độ	Int(255)	
3	Time	Thời gian	Timestamp	

3.3.3 Khỏi web server

Server chính là phần đầu mối tổng, có nhiệm vụ nhận gửi gói tin từ broker lưu vào trong cơ sở dữ liệu. Xử lý dữ liệu thời gian để đưa ra thông báo uống thuốc khi đến giờ bằng cách gửi thông báo về broker.

❖ Activity diagram



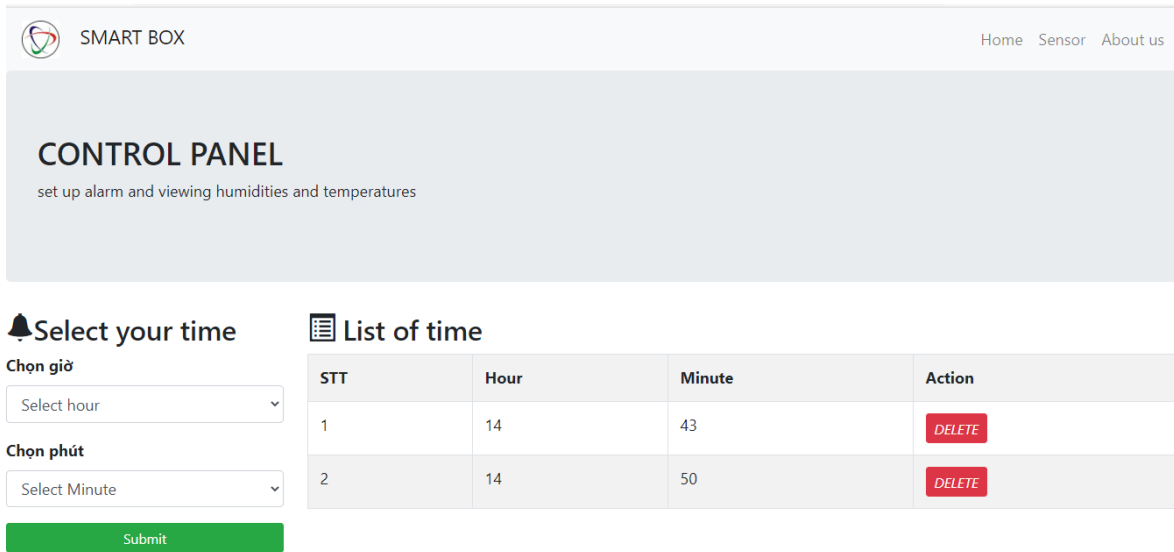
Hình 3.12 Sơ đồ thuật toán chương trình của web server

• Giải thích hình ảnh:

Bằng cách sử dụng đa luồng trong quá trình xử lý server vừa nhận được thông tin đến từ các MQTT Client đồng thời vẫn có thể xử lý được thời gian do người dùng cài đặt được lưu trong cơ sở dữ liệu để có thể đưa ra báo thức đúng thời điểm.

Web server là nơi lưu trữ thông tin của trang web và xử lý các yêu cầu từ người dùng. Web server trong đề án này sử dụng để hiển thị danh sách nhắc nhở, đồng thời người dùng vẫn có thể thiết lập các báo thức để nhắc nhở việc báo thức cho bản thân. Nơi đây còn hỗ trợ việc hiển thị thông tin nhiệt độ ẩm và có thay đổi màu sắc khi hai

đôi tượng đó thay đổi vượt quá định mức. Do có sử dụng kỹ thuật AJAX nên không lo đến việc khi người dùng đang thiết lập nhắc nhở trên giao diện thì trang bị tải lại.



SMART BOX Home Sensor About us

CONTROL PANEL

set up alarm and viewing humidities and temperatures

Select your time

Chọn giờ
Select hour

Chọn phút
Select Minute

Submit

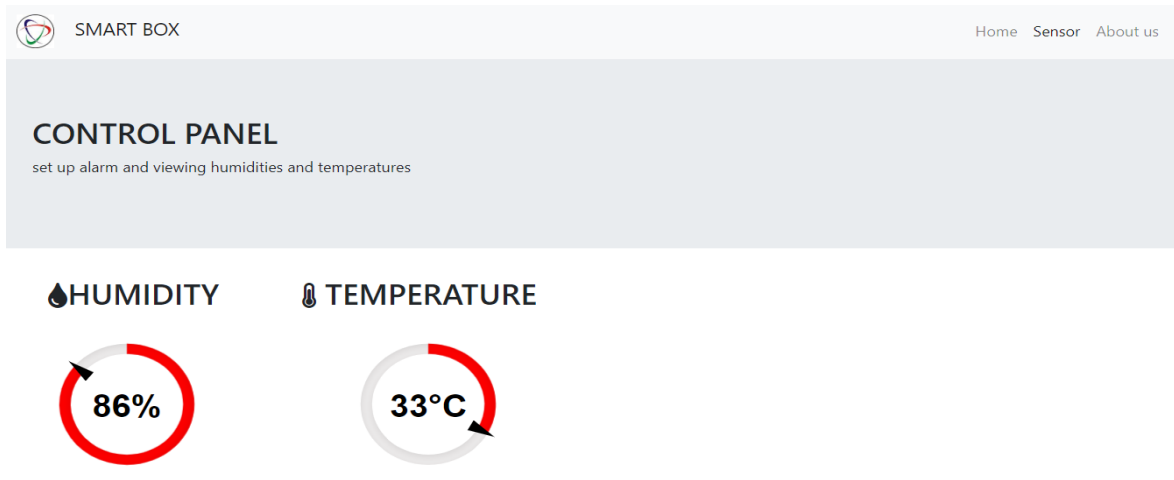
List of time

STT	Hour	Minute	Action
1	14	43	DELETE
2	14	50	DELETE

Hình 3.13 Giao diện chính

- **Giải thích về hình ảnh:**

Theo mặc định khi truy cập vào trang chủ thì người dùng sẽ thấy giao diện như trên hình. Tại đây người dùng có thể thêm mới báo thức ngay trên giao diện của web sau đó trên danh sách sẽ hiện ra những nhắc nhở mà người dùng đã cài đặt.



Hình 3.14 Hiện thị thông tin về độ ẩm và nhiệt độ

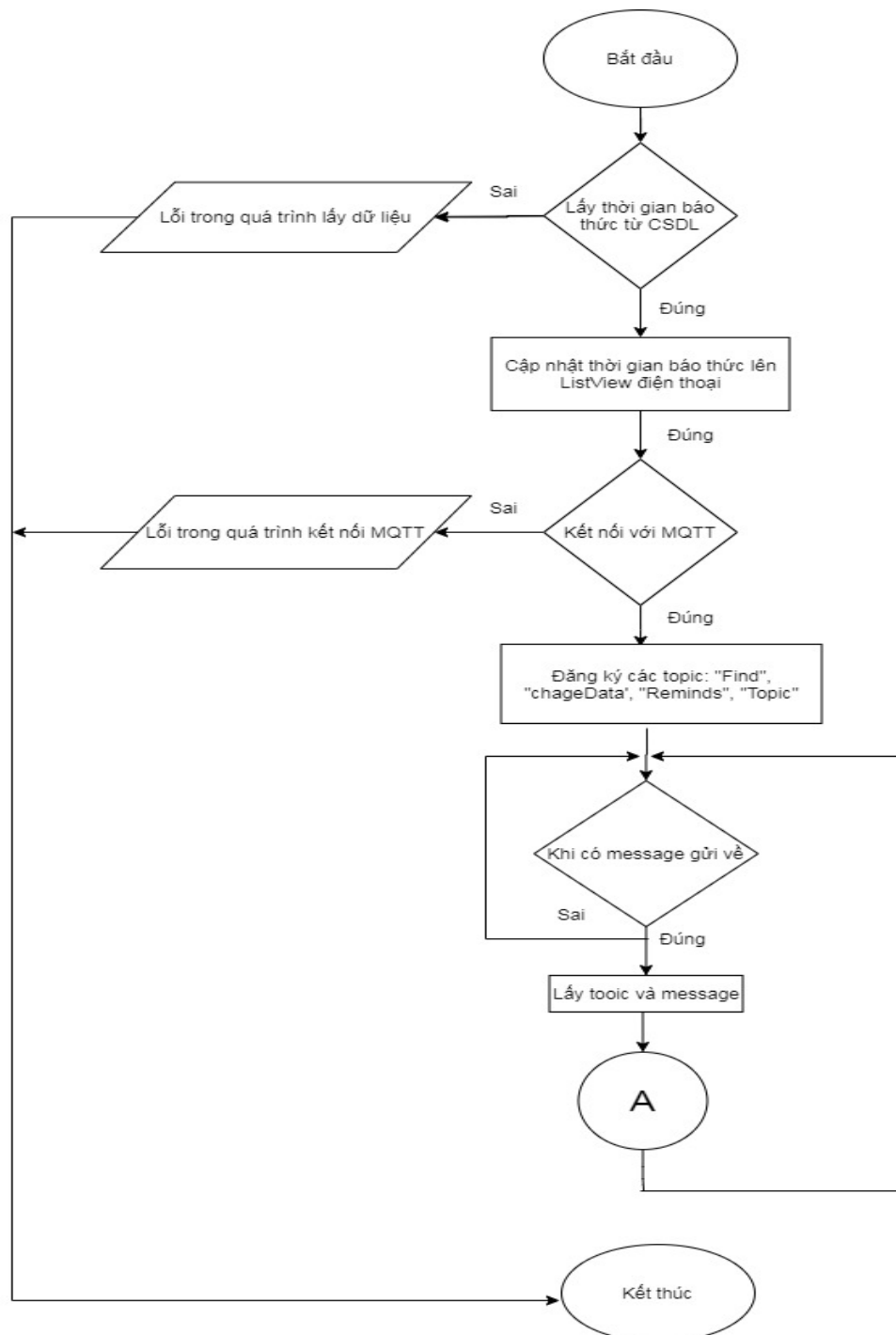
- **Giải thích về hình ảnh:**

Khi nhấn vào thanh menu trên cùng ta sẽ thấy thẻ Sensor đây là nơi người dùng có thể quan sát thông tin về độ ẩm và nhiệt độ. Hệ thống cũng đưa ra cảnh báo bằng màu nếu độ ẩm và nhiệt độ vượt quá định mức, tiện lợi cho việc bảo quản thuốc.

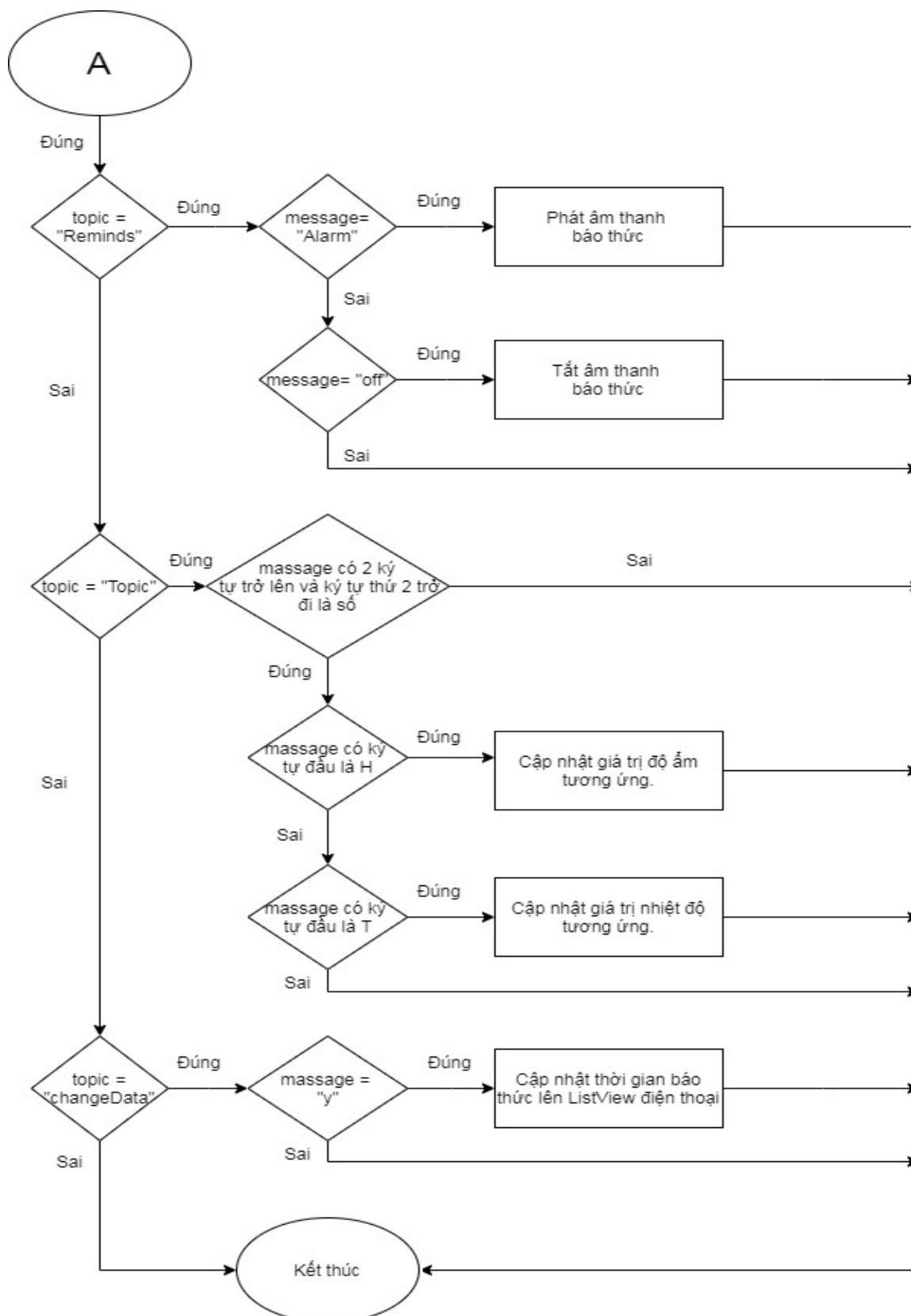
3.3.4 Khối ứng dụng trên điện thoại

Ứng dụng hỗ trợ cài đặt trên hệ điều hành Android. Ứng dụng có chức năng cài đặt báo thức và hiển thị nhiệt độ, độ ẩm kết hợp với tìm kiếm hộp đựng và phát ra thông báo nhắc nhở đến cho người dùng ngay cả khi điện thoại đang tắt.

❖ **Lưu đồ thuật toán:**



Hình 3.15 Sơ đồ thuật toán ứng dụng trên điện thoại (1)



Hình 3.16 Sơ đồ thuật toán ứng dụng trên điện thoại (2)

- Nguyên lý hoạt động:

Khi ứng dụng vừa được khởi động thì nó sẽ kết nối với database.

Nếu kết nối thành công thì lấy dữ liệu thời gian ở database và đổ ra ListView của ứng dụng.

Nếu kết nối không thành công thì sẽ báo lỗi “Lỗi trong quá trình lấy dữ liệu”.

- Kết nối với MQTT:

Nếu kết nối thành công thì đăng ký các topic để phục vụ cho việc đồng bộ dữ liệu và tìm kiếm hộp thuốc.

Nếu kết nối không thành công thì sẽ báo lỗi “Lỗi trong quá trình kết nối MQTT”.

Khi có dữ liệu gửi về từ các topic đã đăng ký ở trên thì xử lý.

Nếu topic là “Reminds”.

Nếu message là “Alarm” thì phát âm thanh báo thức.

Nếu message là “off” thì tắt âm thanh báo thức.

Nếu topic là Topic.

Nếu message có 2 ký tự trở lên và ký tự thứ 2 trở đi là số.

Nếu ký tự đầu tiên là “H” thì cập nhật giá trị độ ẩm.

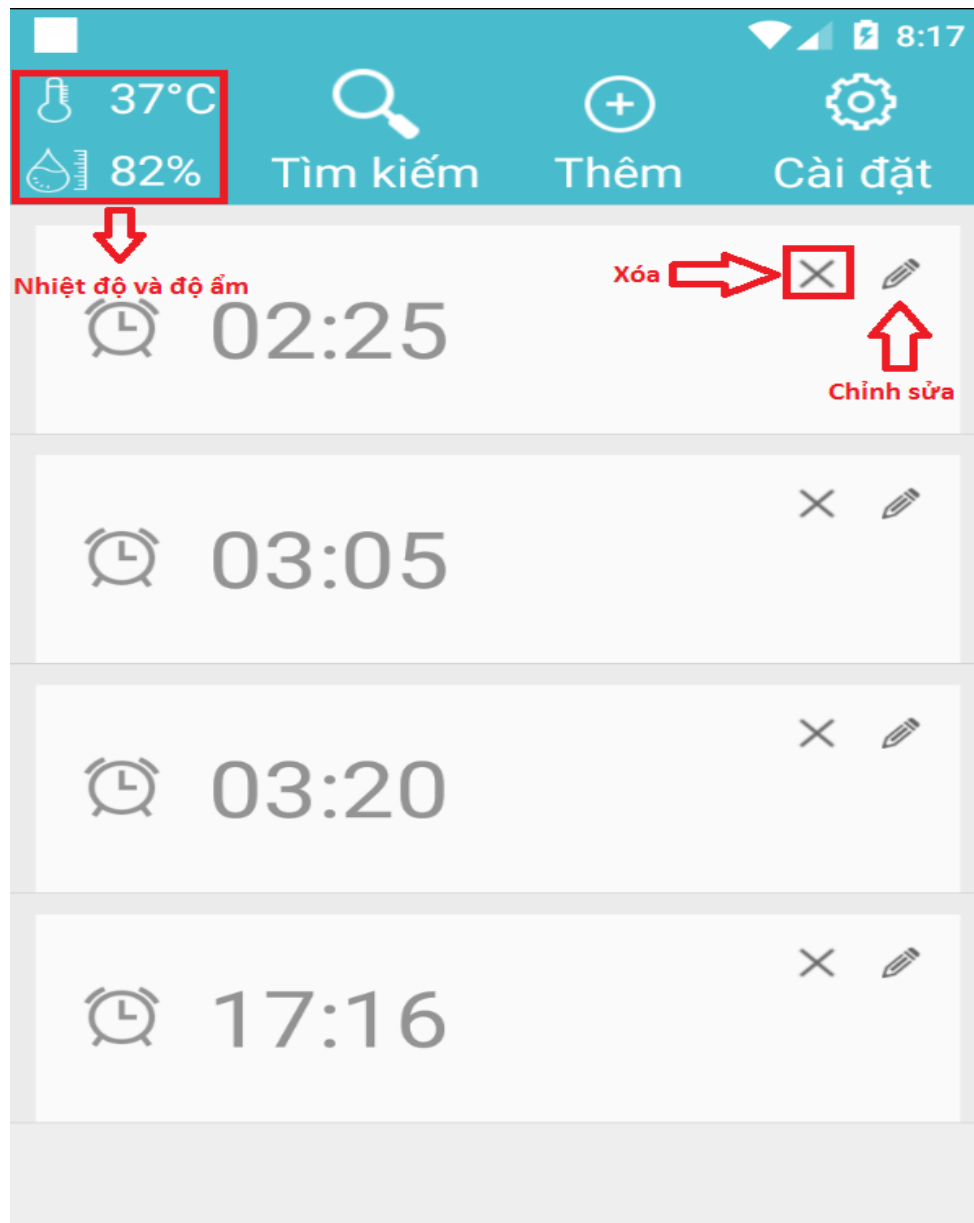
Nếu ký tự đầu tiên là “T” thì cập nhật giá trị nhiệt độ.

Nếu topic là changeData.

Nếu message là “y” thì đồng bộ dữ liệu ở database và ListView.

Bảng 3.4 Các topic đăng ký và nhiệm vụ tương ứng

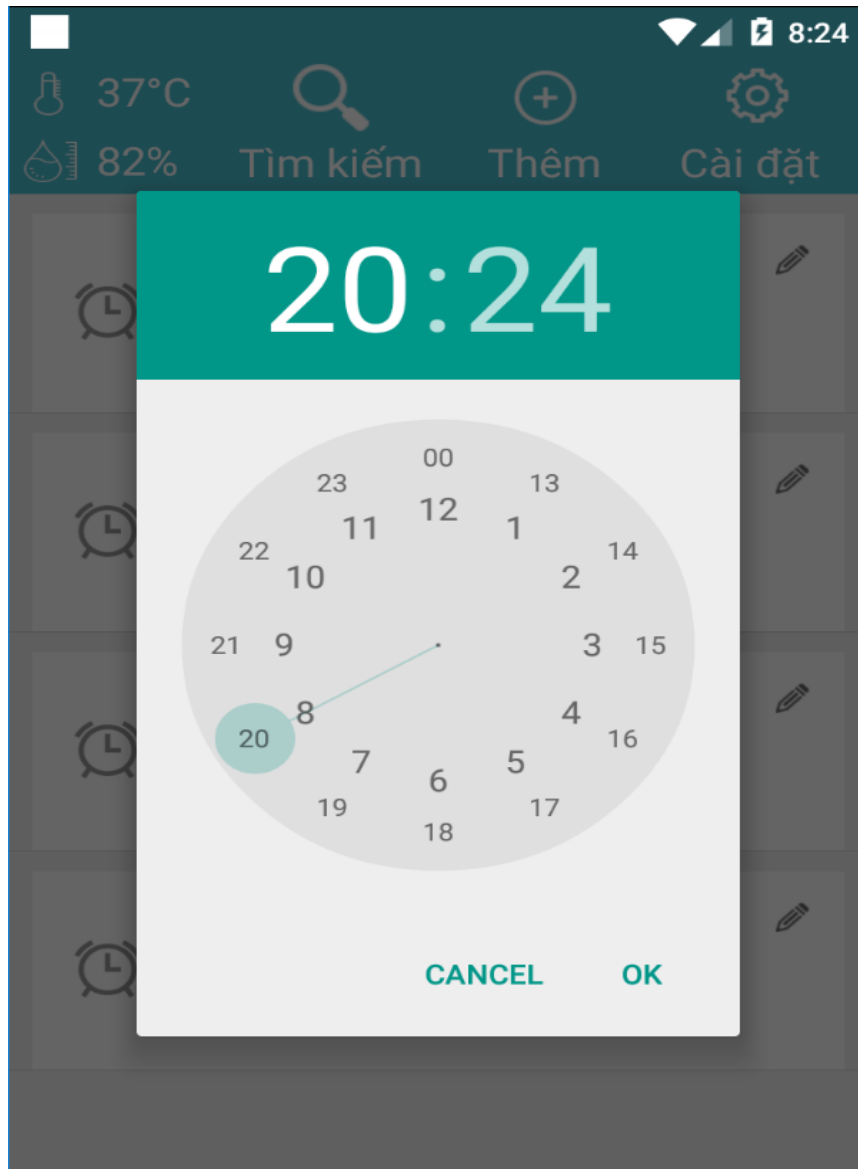
Topic	Nhiệm vụ
Find	Tìm kiếm hộp thuốc.
changeData	Cập nhật lại ListView khi thay đổi database.
Reminds	Bật, tắt báo thức.
Topic	Nhiệt độ và độ ẩm được gửi về.



Hình 3.17 Giao diện quản lý chung

- **Giải thích về hình ảnh:**

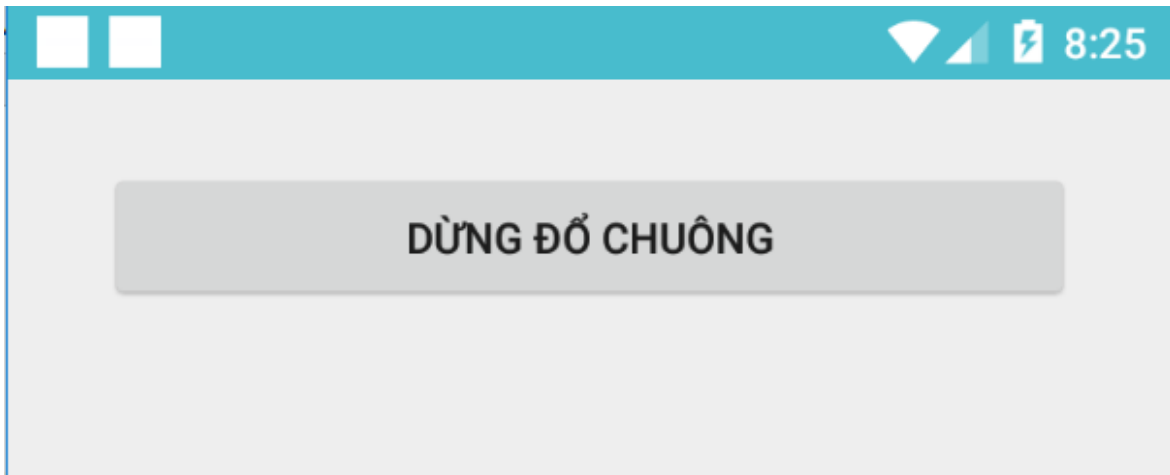
Người dùng có thể thêm, xóa, chỉnh sửa thời gian thông báo, tìm kiếm hộp thuốc và xem thông tin về nhiệt độ và độ ẩm hiện tại ở bên trong hộp thuốc.



Hình 3.18 Giao diện khi muốn thêm một báo thức mới

- **Giải thích về hình ảnh:**

Giao diện người dùng muốn thêm báo thức. Nếu nhấn OK thì báo thức sẽ được thêm vào database và sẽ cập nhật lên giao diện listview của điện thoại. Nếu nhấn vào CANCEL hoặc chạm thay vào vị trí ngoài khung lựa chọn giờ thì sẽ hủy việc thêm.



Hình 3.19 Giao diện khi đến giờ uống thuốc

- **Giải thích về hình ảnh:**

Khi đến giờ uống thuốc, server sẽ gửi thông báo về điện thoại, lúc đó điện thoại sẽ chuyển sang màn hình bên dưới kết hợp với đồ chuông.

3.4 Kết luận chương

Trên đây nhóm chúng tôi đã trình bày xong về phần cứng của hệ thống, lưu đồ và các thành phần liên quan. Chương kế tiếp chúng tôi sẽ trình bày về thi công, kiểm tra kết quả, đánh giá hệ thống.

CHƯƠNG 4: THI CÔNG, KIỂM TRA KẾT QUẢ VÀ ĐÁNH GIÁ HỆ THỐNG

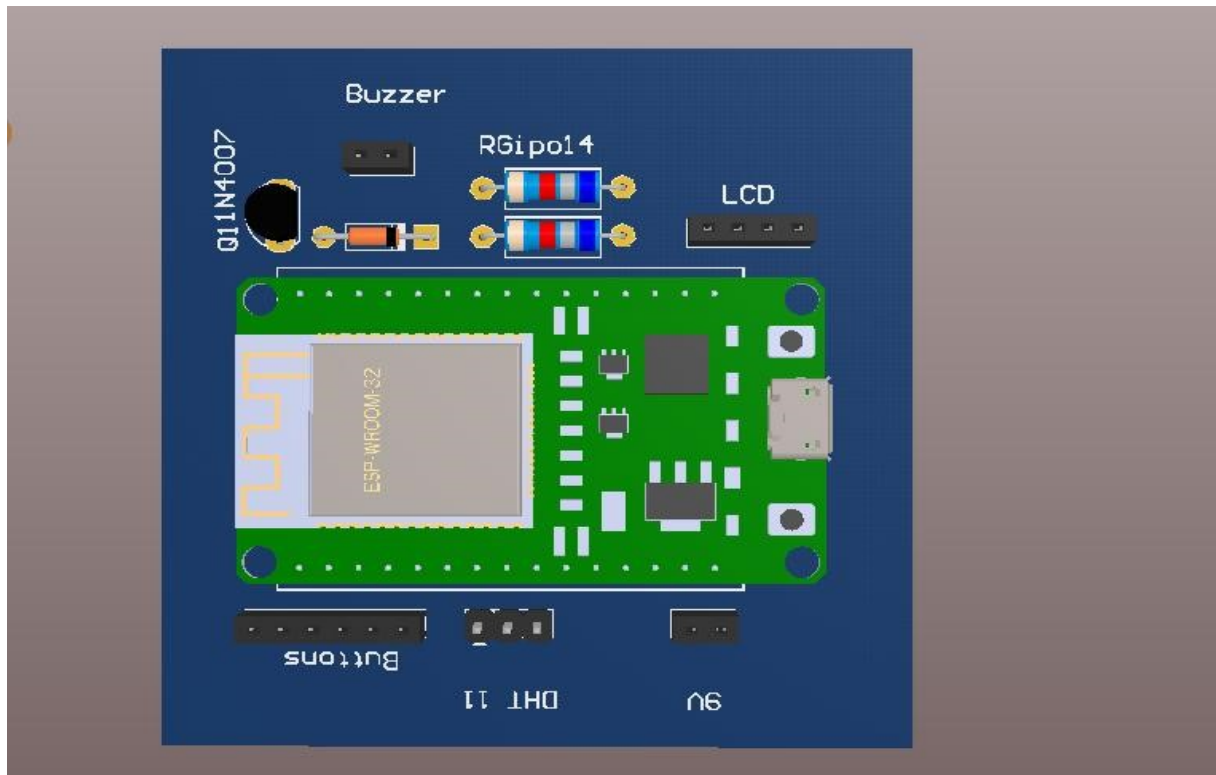
4.1 Giới thiệu chương

Trong chương này sẽ trình bày kết quả thi công của hệ thống gồm phần cứng, phần mềm; đánh giá và đưa ra hướng phát triển của đề tài.

4.2 Thi công mạch

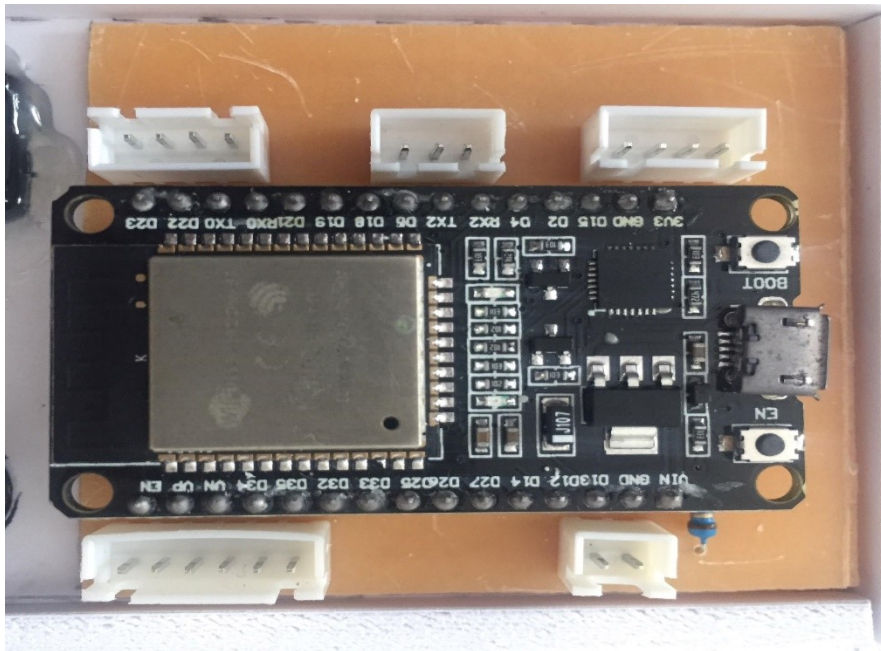
4.2.1 Vẽ mạch PCB

Mạch được vẽ bằng phần mềm Altium Designer (phiên bản 18.1.6). Dưới đây là hình ảnh 3D của mạch:



Hình 4.1 Mạch xử lý trung tâm

4.2.2 Mạch thực tế



Hình 4.2 Mạch xử lý trung tâm thực tế

4.3 Thi công mô hình



Hình 4.3 Mô hình thực tế mặt trước và sau

4.4 Kết quả thực nghiệm và đánh giá hệ thống

4.4.1 Kết quả thực nghiệm

❖ Màn hình chính:



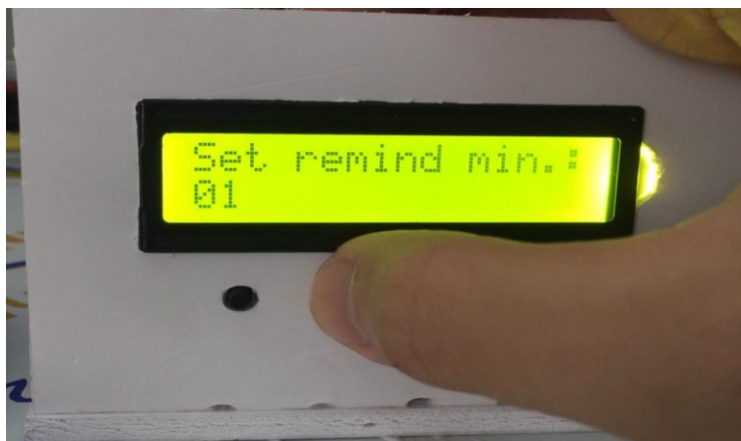
Hình 4.4 Giao diện mặc định

❖ Cài đặt giờ:



Hình 4.5 Giao diện cài đặt giờ

❖ Cài đặt phút:



Hình 4.6 Giao diện cài đặt phút

❖ **Nhắc nhở uống thuốc:**



Hình 4.7 Giao diện nhắc nhở

❖ **Tìm hộp thuốc:**



Hình 4.8 Giao diện tìm hộp thuốc

4.4.2 Đánh giá hệ thống

❖ **Ưu điểm:**

Hoàn thành hết chức năng ban đầu đặt ra cho hệ thống.

Hệ thống hoạt động khá ổn định, tốc độ xử lý nhanh do sử dụng giao thức MQTT.

Giao diện website đơn giản, bảng biểu trực quan, sinh động, thông báo rõ ràng, khách hàng dễ dàng sử dụng.

❖ **Nhược điểm:**

Thời gian thực hiện đề tài còn hạn chế nên một số chức năng không thể tránh khỏi thiếu sót nhất định.

4.5 Kết luận chương

Như vậy là nhóm chúng tôi đã trình bày xong toàn bộ những điểm chính của đề tài qua bốn chương. Tiếp theo là phần kết luận và hướng phát triển.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

❖ Kết luận:

Với đề tài “*Nhắc nhở uống thuốc dựa trên nền tảng IOTs*” nếu có thể tạo nhiều điều kiện tốt để nghiên cứu, phát triển hoàn thiện cùng với những chức năng cần thiết, đề tài này sẽ rất có triển vọng, nhất là đối với những cá nhân luôn phải mang theo thuốc bên mình.

Kết quả đạt được: Xây dựng được website tương tác với người dùng. Xây dựng được ứng dụng hỗ trợ thao tác thiết lập báo thức. Xây dựng được mô hình phần cứng, mạch nhỏ gọn.

❖ Hướng phát triển của đề tài:

Thu nhỏ kích cỡ của hộp thuốc hơn nữa để tiện lợi khi mang theo bên mình.

Sử dụng một phương thức truyền khác song song thay vì chỉ sử dụng được WiFi, đề phòng trường hợp sử dụng tại địa điểm không có sóng WiFi.

Tối ưu phần mềm hơn và thêm một số chức năng thiết yếu khác sau quá trình nghiên cứu thêm.

Toàn bộ dữ liệu về đề tài này sẽ được đưa lên trang GitHub cá nhân với địa chỉ: <https://github.com/lhp1507/medication-reminder-system> để phục vụ cho mục đích nghiên cứu.

TÀI LIỆU THAM KHẢO

- [1] <https://stackoverflow.com/>
- [2] https://www.bogotobogo.com/python/Multithread/python_multithreading_subclassing_Timer_Object.php
- [3] <https://getbootstrap.com/docs/4.5/getting-started/introduction/>
- [4] <https://github.com/bblanchon/ArduinoJson>
- [5] <https://espeasy.readthedocs.io/en/latest/Reference/GPIO.html#best-pins-to-use-on-esp32>
- [6] <https://bandardalat.com/2018/02/22/p5-dong-ho-hen-gio-bao-thuc/>
- [7] https://github.com/khoih-prog/ESP_WiFiManager
- [8] <https://tapit.vn/chuc-nang-nhap-xuat-tin-hieu-gpio-tren-nodemcu-esp32-dev-kit-va-nhung-luu-y-khi-su-dung/>
- [9] <http://arduino.vn/>
- [10] <https://www.arduino.cc/>
- [11] <https://how2electronics.com/internet-clock-esp32-lcd-display-ntp-client/>
- [12] <https://lastminuteengineers.com/esp32-arduino-ide-tutorial/>
- [13] <https://randomnerdtutorials.com/>
- [14] <https://www.slideshare.net/trongthuy2/luan-van-he-thong-giam-sat-nong-nghiep-bang-cong-nghe-iot-hay>
- [15] <https://roboticadiy.com/how-to-make-arduino-alarm-clock/>
- [16] <https://roboindia.com/tutorials/lcd-alarm-clock/>