

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN ĐIỆN TỬ
BỘ MÔN KỸ THUẬT MÁY TÍNH - VIỄN THÔNG



HCMUTE

ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG HỆ THỐNG HỖ TRỢ GIÁM SÁT
HỌC SINH TRÊN XE ĐUA ĐÓN ÚNG DỤNG AWS

NGÀNH HỆ THỐNG NHÚNG VÀ IOT

GVHD: ThS. TRƯỜNG QUANG PHÚC

SVTH: NGUYỄN ĐẶNG MAI THY

MSSV: 20139039

Tp. Hồ Chí Minh, tháng 06 năm 2024

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN – ĐIỆN TỬ
BỘ MÔN KỸ THUẬT MÁY TÍNH – VIỄN THÔNG



HCMUTE

ĐỒ ÁN TỐT NGHIỆP
**XÂY DỰNG HỆ THỐNG HỖ TRỢ GIÁM SÁT
HỌC SINH TRÊN XE ĐUA ĐÓN ỦNG DỤNG AWS
NGÀNH HỆ THỐNG NHÚNG VÀ IOT**

GVHD: ThS. TRƯƠNG QUANG PHÚC
SVTH: NGUYỄN ĐẶNG MAI THY
MSSV: 20139039

Tp. Hồ Chí Minh, tháng 06 năm 2024

Tp. HCM, ngày 03 tháng 07 năm 2024

BẢN GIẢI TRÌNH CHỈNH SỬA ĐỒ ÁN TỐT NGHIỆP

NGÀNH: HỆ THỐNG NHÚNG VÀ IOT

- Tên đề tài: Xây dựng hệ thống hỗ trợ giám sát học sinh trên xe đưa đón ứng dụng AWS
- Tên sinh viên: Nguyễn Đăng Mai Thy MSSV: 20139039
- GVHD: ThS. Trương Quang Phúc
- Hội đồng bảo vệ 06, phòng A3-401, ngày 25 tháng 06 năm 2024
- Giải trình chỉnh sửa báo cáo đồ án tốt nghiệp:

TT	Nội dung góp ý của Hội đồng	Kết quả chỉnh sửa, bổ sung	Ghi chú
1	Chỉnh sửa các lưu đồ giải thuật	Đã thực hiện chỉnh sửa các lưu đồ giải thuật ở chương 3, mục 3.4, hình 3.9, hình 3.12, hình 3.13, hình 3.14, hình 3.18.	
2	Đại từ nhân xưng, đánh giá độ tin cậy	Đã thực hiện chỉnh sửa đại từ nhân xưng ở tất cả các chương. Đã thực hiện bổ sung đánh giá độ tin cậy ở chương 4, mục 4.4.	

Xác nhận của GVHD
(Ký họ và tên)

Nhóm thực hiện báo cáo
(Ký họ và tên)

LỜI CẢM ƠN

Lời đầu tiên, tôi muốn bày tỏ lòng biết ơn sâu sắc đến các thầy cô trong Khoa Điện - Điện Tử và bộ môn Kỹ Thuật Máy Tính - Viễn Thông tại trường Đại học Sư Phạm Kỹ Thuật Thành phố Hồ Chí Minh. Cảm ơn thầy cô đã dành sự quan tâm, tâm huyết và tận tình dạy dỗ, truyền đạt kiến thức cho tôi từ những môn học đại cương đến chuyên ngành. Từ đó tạo nền tảng lý thuyết vững chắc, cũng như tạo điều kiện tốt nhất để tôi có thể phát triển trong suốt quá trình học tập.

Đặc biệt, tôi xin gửi lời cảm ơn chân thành Thầy ThS. Trương Quang Phúc, là giảng viên hướng dẫn và hỗ trợ tôi trong quá trình thực hiện đồ án tốt nghiệp. Tôi luôn biết ơn sự hỗ trợ tận tình, sự chỉ bảo chu đáo và những lời khuyên, gợi ý của thầy. Sự đồng hành của thầy đã giúp tôi vượt qua mọi khó khăn, thách thức trong quá trình thực hiện và hoàn thành tốt đồ án tốt nghiệp này.

Người thực hiện đề tài

Nguyễn Đặng Mai Thy

LỜI CAM ĐOAN

Tôi thực hiện đồ án tốt nghiệp với đề tài “Xây dựng hệ thống hỗ trợ giám sát học sinh trên xe đưa đón ứng dụng AWS” cam đoan không sao chép bất kỳ nội dung hay kết quả nào của các đồ án, công trình khác. Các nội dung được sử dụng để tham khảo xây dựng đề tài đã được trích dẫn đầy đủ ở phần tài liệu tham khảo cuối cùng.

Người thực hiện đề tài

Nguyễn Đặng Mai Thy

TÓM TẮT

Đề tài " Xây dựng hệ thống hỗ trợ giám sát học sinh trên xe đưa đón ứng dụng AWS " tập trung vào việc phát triển một giải pháp công nghệ thông minh nhằm hỗ trợ theo dõi và giám sát quá trình đưa đón học sinh.

Hệ thống được thiết kế để thu thập dữ liệu theo thời gian thực từ các chuyến xe đưa đón học sinh, bao gồm vị trí chuyến xe và thông tin về học sinh trên xe. Dữ liệu sẽ được lưu trữ trên các nền tảng điện toán đám mây là AWS và Firebase. Sau khi được xử lý, dữ liệu sẽ được truyền tới phần giao diện người dùng.

Giao diện người dùng của hệ thống bao gồm Phần mềm Quản lý và Ứng dụng điểm danh. Về Phần mềm Quản lý, người dùng có thể sử dụng để quản lý các thông tin liên quan đến chuyến xe, học sinh và tài xế. Về Ứng dụng điểm danh sẽ có hai ứng dụng con là Điểm danh lên xe và Điểm danh xuống trạm.

Điểm danh lên xe ứng dụng công nghệ nhận diện khuôn mặt để đảm bảo học sinh lên đúng xe và điểm danh xuống trạm sẽ là điểm danh hai lớp (bao gồm điểm danh khuôn mặt và so sánh vị trí) để chắc chắn việc học sinh đã xuống đúng trạm.

Nhìn chung, việc thiết kế và triển khai hệ thống hỗ trợ nhằm mục đích nâng cao hiệu quả quản lý, giảm thiểu rủi ro. Cung cấp phương tiện an toàn, đáng tin cậy cho việc đưa đón học sinh đến trường và về nhà. Điều này sẽ giúp tăng cường sự tiện lợi và an tâm cho phụ huynh cũng như an toàn cho học sinh.

Cuối cùng, về kết quả thực hiện hệ thống tôi đã hoàn thành mô hình có đầy đủ các chức năng mặc định theo yêu cầu người dùng. Định hướng được sự phát triển của hệ thống trong tương lai, áp dụng đề tài vào thực tế cũng như bổ sung thêm các tính năng mới nhằm tối ưu hóa hệ thống này.

MỤC LỤC

LỜI CẢM ƠN	I
LỜI CAM ĐOAN	II
TÓM TẮT	III
MỤC LỤC	IV
DANH MỤC HÌNH ẢNH	VII
DANH MỤC BẢNG BIỂU.....	X
DANH MỤC CÁC TỪ VIẾT TẮT.....	XI
CHƯƠNG 1 TỔNG QUAN	1
1.1 ĐẶT VÂN ĐÈ	1
1.2 MỤC TIÊU ĐÈ TÀI.....	2
1.3 TÌNH HÌNH NGHIÊN CỨU	2
1.3.1 <i>Tình hình nghiên cứu trong nước:</i>	3
1.3.2 <i>Tình hình nghiên cứu ngoài nước:</i>	3
1.4 PHƯƠNG PHÁP NGHIÊN CỨU.....	4
1.5 BỐ CỤC CỦA ĐÈ TÀI.....	5
CHƯƠNG 2 CƠ SỞ LÝ THUYẾT.....	7
2.1 TỔNG QUAN VỀ HỆ THỐNG.....	7
2.2 GIỚI THIỆU VỀ AMAZON WEB SERVICES (AWS).....	8
2.2.1 <i>Tổng quan về Amazon Web Services</i>	8
2.2.2 <i>Tổng quan về Amazon Lambda</i>	8
2.2.3 <i>Tổng quan về Amazon DynamoDB</i>	9
2.2.4 <i>Tổng quan về Amazon Rekognition</i>	10
2.2.5 <i>Tổng quan về Amazon API Gateway</i>	11
2.2.6 <i>Tổng quan về Amazon S3</i>	11

2.2.7 <i>Tổng quan về AWS Identity and Access Management</i>	12
2.2.8 <i>Tổng quan về Amazon CloudWatch</i>	13
2.3 GIỚI THIỆU VỀ FIREBASE	14
2.3.1 <i>Tổng quan về Firebase</i>	14
2.3.2 <i>Tổng quan về Firebase Realtime Database</i>	15
2.4 GIỚI THIỆU VỀ CÁC FRAMEWORK SỬ DỤNG	16
2.4.1 <i>Tổng quan về React.js</i>	16
2.4.2 <i>Tổng quan về Next.js</i>	16
2.4.3 <i>Tổng quan về Tailwind CSS</i>	17
2.4.4 <i>Tổng quan về Node.js</i>	18
2.5 GIỚI THIỆU VỀ PHẦN CỨNG SỬ DỤNG	19
2.5.1 <i>Tổng quan về Raspberry Pi 4</i>	19
2.5.2 <i>Tổng quan về Module GPS NEO-M8N</i>	20
2.5.3 <i>Tổng quan về Module Relay</i>	21
2.5.4 <i>Tổng quan về Motor</i>	21
CHƯƠNG 3 THIẾT KẾ VÀ THI CÔNG.....	23
3.1 YÊU CẦU HỆ THỐNG	23
3.2 ĐẶC TẢ HỆ THỐNG	24
3.2.1 <i>Chức năng của hệ thống</i>	24
3.2.2 <i>Mô hình tổng thể hệ thống</i>	24
3.2.3 <i>Sơ đồ khái niệm</i>	25
3.2.4 <i>Hoạt động của hệ thống</i>	26
3.3 THIẾT KẾ PHẦN CỨNG	27
3.3.1 <i>Khối xử lý trung tâm</i>	28
3.3.2 <i>Khối đọc định vị</i>	28
3.3.3 <i>Khối đóng/ mở cửa</i>	29
3.3.4 <i>Khối nguồn</i>	30
3.3.5 <i>Sơ đồ nguyên lý</i>	31

3.4	THIẾT KẾ PHẦN MỀM	32
3.4.1	<i>Phần mềm quản lý</i>	33
3.4.2	<i>Ứng dụng điểm danh</i>	38
3.5	THIẾT KẾ AMAZON WEB SERVICES	42
3.5.1	<i>Chương trình con Lấy dữ liệu từ AWS</i>	43
3.5.2	<i>Chương trình con Truyền dữ liệu đến AWS</i>	44
3.5.3	<i>Chương trình con Amazon Rekognition</i>	45
3.6	THIẾT KẾ FIREBASE	46
CHƯƠNG 4 KẾT QUẢ, ĐÁNH GIÁ HỆ THỐNG		48
4.1	KẾT QUẢ PHẦN CỨNG	48
4.2	KẾT QUẢ PHẦN MỀM	50
4.2.1	<i>Giao diện Phần mềm Quản lý</i>	50
4.2.2	<i>Giao diện Ứng dụng Điểm danh</i>	57
4.2.3	<i>Giao diện hiển thị trên các thiết bị khác</i>	62
4.3	KẾT QUẢ DỮ LIỆU LUU TRU	63
4.3.1	<i>Lưu trữ dữ liệu trên AWS</i>	63
4.3.2	<i>Lưu trữ dữ liệu trên Firebase</i>	68
4.4	ĐÁNH GIÁ HỆ THỐNG	68
CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN TRONG TƯƠNG LAI		71
5.1	KẾT LUẬN	71
5.2	HƯỚNG PHÁT TRIỂN TRONG TƯƠNG LAI	72
TÀI LIỆU THAM KHẢO		XIII

DANH MỤC HÌNH ẢNH

Hình 2.1: Hình ảnh minh họa về Amazon Web Services.....	8
Hình 2.2: Hình ảnh minh họa về Amazon Lambda	9
Hình 2.3: Hình ảnh minh họa về chức năng Amazon DynamoDB.....	9
Hình 2.4: Hình ảnh minh họa về Amazon Rekognition.....	10
Hình 2.5: Hình ảnh minh họa về chức năng của Amazon API Gateway.....	11
Hình 2.6: Hình ảnh minh họa về Amazon S3	12
Hình 2.7: Hình ảnh minh họa về AWS Identity and Access Management.....	13
Hình 2.8: Hình ảnh minh họa về Amazon CloudWatch	13
Hình 2.9: Hình ảnh minh họa về Firebase.....	14
Hình 2.10: Hình ảnh minh họa về ứng dụng Firebase Realtime Database API Web/ App.....	15
Hình 2.11: Hình ảnh minh họa về tính năng của React.js.....	16
Hình 2.12: Hình ảnh minh họa tính năng của Next.js.....	17
Hình 2.13: Hình ảnh minh họa về Tailwind CSS.....	18
Hình 2.14: Hình ảnh minh họa về kiến trúc Node.js.....	18
Hình 2.15: Hình ảnh minh họa về phần cứng Raspberry Pi 4.....	19
Hình 2.16: Hình ảnh minh họa về block diagram của module NEO-M8	20
Hình 2.17: Hình ảnh minh họa về kiến trúc của Relay	21
Hình 2.18: Hình ảnh minh họa về cấu trúc của động cơ (motor).....	22
Hình 3.1: Mô hình tổng thể hệ thống	24
Hình 3.2: Sơ đồ khái niệm hệ thống	25
Hình 3.3: Sơ đồ hoạt động của hệ thống	26
Hình 3.4: Sơ đồ khái chi tiết của phần cứng	27
Hình 3.5: Sơ đồ nguyên lý Raspberry Pi 4B	28
Hình 3.6: Sơ đồ nguyên lý kết nối module NEO-M8 với Raspberry Pi 4B.....	28
Hình 3.7: Sơ đồ nguyên lý kết nối Relay Module, Motor với Raspberry Pi 4.....	29
Hình 3.8: Sơ đồ nguyên lý kết nối phần cứng hệ thống.....	31

Hình 3.9: Lưu đồ giải thuật chính của phần mềm.....	32
Hình 3.10: Lưu đồ giải thuật của Phần mềm quản lý.....	33
Hình 3.11: Lưu đồ giải thuật của chương trình con Đăng Nhập.....	34
Hình 3.12: Lưu đồ giải thuật của chương trình con Giám sát chuyến xe.	35
Hình 3.13: Lưu đồ giải thuật của chương trình con Quản lý học sinh.....	36
Hình 3.14: Lưu đồ giải thuật của Chương trình con Quản lý tài xế.....	37
Hình 3.15: Lưu đồ giải thuật của Ứng dụng điểm danh.....	38
Hình 3.16: Lưu đồ giải thuật của chương trình con Điểm danh lên xe.....	39
Hình 3.17: Lưu đồ giải thuật của chương trình con Điểm danh xuống trạm.	40
Hình 3.18: Lưu đồ giải thuật của chương trình con So sánh dữ liệu vị trí.....	41
Hình 3.19: Sơ đồ nguyên lý của các dịch vụ AWS được sử dụng trong hệ thống....	42
Hình 3.20: Sơ đồ nguyên lý của chương trình con Lấy dữ liệu từ AWS	43
Hình 3.21: Sơ đồ nguyên lý của chương trình con Truyền dữ liệu đến AWS (Thông tin tài xế).....	44
Hình 3.22: Sơ đồ nguyên lý của chương trình con Truyền dữ liệu đến AWS (Thông tin học sinh)	44
Hình 3.23: Sơ đồ nguyên lý của chương trình con Amazon Rekognition.	45
Hình 3.24: Sơ đồ nguyên lý của hệ thống Firebase	46
Hình 3.25: Sơ đồ nguyên lý của chương trình con Lấy dữ liệu từ Firebase	46
Hình 3.26: Sơ đồ nguyên lý của chương trình con Truyền dữ liệu đến Firebase	47
Hình 4.1: Bản thiết kế hộp đựng mô hình hệ thống	48
Hình 4.2: Kết quả đi dây phần cứng hoàn chỉnh.....	49
Hình 4.3: Kết quả mô hình phần cứng của hệ thống.....	49
Hình 4.4: Giao diện màn hình chính	50
Hình 4.5: Giao diện Đăng nhập.....	50
Hình 4.6: Giao diện Đăng nhập nếu đăng nhập sai.....	51
Hình 4.7: Giao diện phần Tổng quan	52
Hình 4.8: Giao diện phần Giám sát chuyến xe.....	52

Hình 4.9: Tính năng lọc thông tin của phần Giám sát chuyến xe theo chuyến xe ...	53
Hình 4.10: Tính năng lọc thông tin của phần Giám sát chuyến xe theo trạng thái ...	53
Hình 4.11: Giao diện phần Quản lý học sinh	54
Hình 4.12: Tính năng lọc thông tin của phần Quản lý học sinh theo chuyến xe	55
Hình 4.13: Tính năng thêm thông tin, sửa thông tin học sinh của phần Quản lý học sinh.....	55
Hình 4.14: Giao diện phần Quản lý tài xế.....	55
Hình 4.15: Tính năng thêm thông tin, sửa thông tin tài xế của phần Quản lý tài xế	56
Hình 4.16: Giao diện màn hình chính của Ứng dụng điểm danh.....	57
Hình 4.17: Giao diện chính của phần Điểm danh lên xe	58
Hình 4.18: Giao diện sau khi tiến hành chụp ảnh chân dung.....	58
Hình 4.19: Các lựa chọn phương thức điều hướng	59
Hình 4.20: Giao diện trả về sau khi điểm danh lên xe thành công.	59
Hình 4.21: Giao diện trả về sau khi điểm danh lên xe KHÔNG thành công.....	60
Hình 4.22: Giao diện màn hình chính của phần Điểm danh xuống trạm.....	60
Hình 4.23: Giao diện trả về sau khi điểm danh xuống trạm thành công.....	61
Hình 4.24: Giao diện trả về sau khi điểm danh xuống trạm KHÔNG thành công. ..	61
Hình 4.25: Hình ảnh minh họa khi hiển thị giao diện trên các thiết bị khác nhau....	62
Hình 4.26: Kết quả của Amazon API Gateway sau khi thiết lập hệ thống	63
Hình 4.27: Kết quả của Amazon DynamoDB sau khi thiết lập hệ thống	64
Hình 4.28: Kết quả lưu thông tin tài xế trong bảng Drivers	64
Hình 4.29: Kết quả lưu thông tin học sinh trong bảng Students	65
Hình 4.30: Kết quả của Amazon Lambda Functions sau khi thiết lập hệ thống	66
Hình 4.31: Kết quả của Amazon S3 sau khi thiết lập hệ thống	67
Hình 4.32: Kết quả lưu trữ hình ảnh chân dung thông tin học sinh	67
Hình 4.33: Kết quả lưu trữ hình ảnh điểm danh của học sinh	67
Hình 4.34: Kết quả của Firebase Realtime Database sau khi thiết lập hệ thống	68

DANH MỤC BẢNG BIỂU

Bảng 3.1: Sơ đồ kết nối chân của module NEO-M8 với Raspberry Pi 4B	29
Bảng 3.2: Sơ đồ kết nối chân của module Relay, Motor với Raspberry Pi 4 và Nguồn 12V	30
Bảng 3.3: Thông số về dòng điện, điện áp và công suất tiêu thụ trung bình của Motor.	30
Bảng 3.4: Thông số về dòng điện, điện áp và công suất tiêu thụ của các linh kiện còn lại.....	30

DANH MỤC CÁC TỪ VIẾT TẮT

Viết tắt	Mô tả
API	Application Programming Interface
ARM	Advanced RISC Machines (processor architecture)
AWS	Amazon Web Services
C	Celsius
CPU	Central Processing Unit
CSS	Cascading Style Sheets
DOM	Document Object Model
GND	Ground
GPIO	General-Purpose Input/Output
GPS	Global Positioning System
GPU	Graphics Processing Unit
HTTP	Hypertext Transfer Protocol
IAM	Identity and Access Management
ID	Identification
IN	Input
JS	Java Script
JSON	JavaScript Object Notation
LCD	Liquid Crystal Display
NEO	Not Elsewhere Classified
NO	Normally Open
NoSQL	Not only SQL (database)
RAM	Random Access Memory
REST	Representational State Transfer
RFID	Radio Frequency Identification

Rx	Receive
SoC	System on a Chip
TTL	Transistor-Transistor Logic
Tx	Transmit
THCS	Trung học cơ sở
UART	Universal Asynchronous Receiver – Transmitter
VCC	Positive supply voltage

CHƯƠNG 1 TỔNG QUAN

1.1 ĐẶT VẤN ĐỀ

Giáo dục giữ vai trò then chốt trong sự phát triển của xã hội hiện đại, tạo nền tảng vững chắc cho tương lai của thế hệ trẻ. Mỗi gia đình đều mong muốn con em mình được học tập trong môi trường an toàn và chất lượng, bao gồm cả quá trình đi lại từ nhà đến trường. Tuy nhiên, việc vận hành và quản lý dịch vụ đưa đón học sinh của các trường vẫn đang gặp nhiều khó khăn, gây lo lắng và bất an cho phụ huynh và cộng đồng.

Một vấn đề đáng quan ngại gần đây là việc có trường hợp học sinh bị quên lại trên xe đưa đón sau khi kết thúc lịch trình, gây nguy hiểm cho sức khỏe và an toàn của các em. Bên cạnh đó, việc giám sát thông tin của học sinh và tài xế cũng chưa rõ ràng ảnh hưởng đến chất lượng và hiệu quả của dịch vụ.

Việc áp dụng các công nghệ thông minh vào hệ thống đưa đón nhằm hỗ trợ giám sát chuyến xe, học sinh là một giải pháp hữu ích để giải quyết các vấn đề trên. Các thiết bị cảm biến định vị giúp người dùng có thể giám sát và kiểm soát lịch trình chuyến xe một cách tự động và chính xác. Giúp người quản lý có thể phản ứng kịp thời khi có sự cố xảy ra. Ngoài ra, Ứng dụng điểm danh học sinh bằng khuôn mặt giúp hệ thống có thể theo dõi học sinh khi lên/xuống xe, đảm bảo học sinh lên đúng xe xuống đúng trạm và không có em nào bị quên lại.

Hệ thống hỗ trợ giám sát học sinh trên xe đưa đón mang lại nhiều lợi ích quan trọng, như giải quyết vấn đề đảm bảo an toàn cho học sinh và không một học sinh nào bị bỏ quên trên xe. Cải thiện việc theo dõi lịch trình, quản lý thông tin học sinh, tài xế một cách chính xác. Đảm bảo rằng con em mỗi gia đình đều được đưa đón an toàn đến trường, tạo nền tảng vững chắc cho một môi trường giáo dục chất lượng. Ngoài ra, việc áp dụng công nghệ vào các hệ thống này cũng giúp giảm thiểu nhân công, tự động hóa các công việc không quan trọng.

1.2 MỤC TIÊU ĐỀ TÀI

Đối với đề tài “Xây dựng hệ thống hỗ trợ giám sát học sinh trên xe đưa đón ứng dụng AWS” tôi đã đề ra 4 mục tiêu chính:

Thứ nhất: Thiết kế hệ thống định vị xe đưa đón, mở cửa xe buýt tự động:

- Sử dụng công nghệ định vị GPS để xác định vị trí chính xác của các xe đưa đón trong thời gian thực.
- Mở cửa xe buýt tự động khi học sinh xuống đúng trạm của mình.

Thứ hai: Thiết kế hệ thống điểm danh học sinh lên/xuống xe:

- Sử dụng công nghệ nhận diện khuôn mặt để theo dõi và điểm danh học sinh khi lên và xuống xe.
- Lưu trữ và phân tích dữ liệu về việc lên/xuống xe của học sinh.

Thứ ba: Xây dựng giao diện người dùng:

- Phần mềm quản lý thông tin chuyến xe, học sinh, tài xế và Ứng dụng điểm danh học sinh lên/xuống xe.
- Giao diện trực quan, dễ sử dụng sẽ giúp người quản lý dễ dàng theo dõi và giám sát

Thứ tư: Kiểm thử và đảm bảo chất lượng hệ thống:

- Tiến hành các bài kiểm thử toàn diện.
- Đảm bảo độ trễ truyền dữ liệu của hệ thống không quá 5 giây.

1.3 TÌNH HÌNH NGHIÊN CỨU

Về phần tình hình nghiên cứu, tôi muốn trình bày ngắn gọn nội dung của các công trình đã được nghiên cứu trước đó để khai thác để hỗ trợ cho việc xây dựng ý tưởng và phát triển đề tài từ đó rút ra được phương án thực hiện tốt nhất, tối ưu nhất cho đề tài. Góp phần hoàn thành chính chu nhất cho toàn bộ hệ thống.

1.3.1 Tình hình nghiên cứu trong nước:

Từ vụ việc bé trai lớp 1 Trường Tiểu học Quốc tế Gateway (quận Cầu Giấy, Hà Nội) tử vong, vẫn đề bảo đảm an toàn trong hoạt động đưa đón học sinh bằng xe ô tô của nhà trường đặt ra cấp thiết hơn. Sau nhiều văn bản siết chặt hoạt động này của các cấp quản lý được ban hành, dường như các bậc cha mẹ vẫn chưa hoàn toàn yên tâm với các giải pháp "trên giấy" [1].

Trước thực trạng này, Nguyễn Tiến An, học sinh lớp 9, Trường THCS Lương Thế Vinh (TT.Đăk Tô, H.Đăk Tô, Kon Tum) sau 2 tháng miệt mài nghiên cứu với hàng chục lần thử nghiệm, An đã chế tạo thành công thiết bị kiểm soát học sinh. Thiết bị kiểm soát có thể đếm số lượng học sinh trên xe thông qua việc quét thẻ ID vào thiết bị. Mỗi học sinh có một mã ID, tài xế chỉ cần quan sát số liệu đếm học sinh được hiển thị trên thiết bị trong quá trình lên và xuống xe [2].

Ngoài ra, Trung tâm Nghiên cứu phát triển và chuyển giao công nghệ định vị sử dụng vệ tinh - Đại học Bách khoa Hà Nội (Trung tâm NAVIS) đã thực hiện công trình nghiên cứu và cho ra đời hệ thống quản lý tối ưu lộ trình xe đưa đón học sinh có tên gọi “Safe school” - Trường học an toàn, hệ thống giám sát theo công nghệ định vị tiên tiến này giúp học sinh được cập nhật vị trí, thông tin gửi về cho phụ huynh và nhà trường để cùng giám sát, bảo đảm sự an toàn, tránh xảy ra những sự cố ngoài ý muốn hay các nguy cơ mất an toàn trong môi trường học đường và ngoài trường học [1].

1.3.2 Tình hình nghiên cứu ngoài nước:

A Real-time School Bus Tracking and Monitoring [3] một bài báo được xuất bản trên tạp chí khoa học IEEE đã đề xuất sử dụng hệ thống theo dõi thời gian thực sử dụng các kỹ thuật xử lý hình ảnh. Hệ thống này giám sát hình ảnh bên trong xe buýt và nhận diện khuôn mặt học sinh đang có mặt trên xe. Nhằm theo dõi số lượng và thông tin những học sinh ấy. Bên cạnh đó, nhận dạng khuôn

mặt cũng được sử dụng trong hệ thống điểm danh tự động, loại bỏ hầu hết các hạn chế của hệ thống điểm danh thủ công như việc làm giả dữ liệu điểm danh, điểm danh dùm và hệ thống điểm danh không an toàn.

Trong hệ thống này sử dụng công nghệ RFID, GPS kết nối với một máy chủ từ xa qua wifi sử dụng một vi điều khiển Arduino. Một mô-đun GPS được sử dụng để xác định tọa độ địa lý hiện tại của vị trí xe cũng như tốc độ di chuyển của xe. Một đầu đọc RFID để điểm danh mỗi học sinh khi họ lên hoặc xuống xe bằng cách đọc ID từ thẻ RFID của họ. Hệ thống còn tải thông tin từ các thiết bị ngoại vi lên cơ sở dữ liệu trên máy chủ web.

1.4 PHƯƠNG PHÁP NGHIÊN CỨU

Để đáp ứng các mục tiêu của đề tài, tôi sử dụng các phương pháp nghiên cứu như thu thập số liệu, thực nghiệm và phân tích tổng kết kinh nghiệm.

Phương pháp thu thập số liệu:

- Sử dụng phương pháp quan sát: Tiến hành quan sát trực tiếp các tình huống đưa đón học sinh bằng xe buýt và ghi chép số liệu về lịch trình, thời gian, vị trí, số lượng học sinh, phương thức giám sát và các thông tin liên quan khác.

Phương pháp thực nghiệm:

- Thiết kế và triển khai hệ thống: Thiết kế và triển khai hệ thống hỗ trợ giám sát học sinh trên xe đưa đón, bao gồm các thành phần như hệ thống định vị GPS, ứng dụng điểm danh bằng khuôn mặt và phần mềm quản lý chuyến xe, học sinh.
- Tiến hành thử nghiệm: Tiến hành các bài kiểm tra và thử nghiệm hệ thống trên thực tế, đánh giá khả năng theo dõi và báo cáo vị trí, độ chính xác và độ tin cậy của hệ thống

Phương pháp phân tích tổng kết kinh nghiệm:

- Đánh giá hiệu quả: Đánh giá hiệu quả của hệ thống hỗ trợ giám sát học sinh trên xe đưa đón bằng cách so sánh các kết quả thực nghiệm với mục tiêu cũng như các tiêu chuẩn đã được đề ra.
- Phân tích dữ liệu: Phân tích các dữ liệu đã được thu thập từ quá trình thử nghiệm trên nhằm xem xét xu hướng, mẫu dữ liệu và biểu đồ để hiểu rõ hơn về hiệu suất và khả năng của hệ thống.
- Tổng kết kinh nghiệm: Dựa trên kết quả phân tích dữ liệu để tổng kết kinh nghiệm và rút ra những kết luận thực tế về việc sử dụng hệ thống. Đề xuất các phương pháp cải tiến để nâng cao hiệu suất và tính ứng dụng của hệ thống.

Dựa trên các cơ sở này, tôi có thể thực hiện nghiên cứu và phát triển hệ thống hỗ trợ giám sát học sinh trên xe đưa đón một cách toàn diện và có căn cứ dữ liệu đáng tin cậy.

1.5 BỘ CỤC CỦA ĐỀ TÀI

Chương 1: Tổng quan

- Đặt vấn đề, giới thiệu khái quát và tổng quan về mục tiêu đề tài, tình hình nghiên cứu, các phương pháp nghiên cứu cũng như bộ cục của đề tài.

Chương 2: Cơ sở lý thuyết

- Giới thiệu, mô tả tổng quan về các cơ sở lý thuyết, nguyên lý hoạt động và các tính năng được ứng dụng vào hệ thống của các module phần cứng, các ứng dụng điện toán đám mây, các dịch vụ Amazon cũng như các Framework được sử dụng trong đề tài.

Chương 3: Thiết kế và thi công:

- Trình bày các sơ đồ khái niệm, nguyên lý hoạt động, lưu đồ giải thuật của các khái thành phần của toàn hệ thống.

Chương 4: Kết quả, đánh giá hệ thống:

- Trình bày và đánh giá các tính năng, mô hình, kết quả cuối cùng đạt được từ phần cứng và phần mềm của hệ thống.

Chương 5: Kết luận và hướng phát triển:

- Kết luận đề tài từ những kết quả đã đạt được, nêu lên các ưu điểm và nhược điểm của hệ thống. Từ đó, đề xuất các hướng phát triển trong tương lai cho hệ thống.

CHƯƠNG 2 CƠ SỞ LÝ THUYẾT

2.1 TỔNG QUAN VỀ HỆ THỐNG

Hệ thống hỗ trợ giám sát học sinh trên xe đưa đón ứng dụng AWS được thiết kế để cung cấp một giải pháp thông minh và hiệu quả cho việc quản lý và giám sát dịch vụ đưa đón học sinh bằng xe buýt của nhà trường.

Một số tính năng chính của hệ thống bao gồm:

- Thu thập và phân tích dữ liệu theo thời gian thực từ các chuyến xe đưa đón, bao gồm vị trí và thông tin của học sinh đang trên xe. Dữ liệu được lưu trữ trên nền tảng đám mây để xử lý và truyền đến giao diện người dùng.
- Sử dụng công nghệ định vị GPS để theo dõi vị trí của xe đón cũng như module relay để tự động hóa tính năng đóng/mở cửa xe buýt theo Úng dụng điểm danh.
- Thiết kế giao diện người dùng tích hợp Phần mềm quản lý và Úng dụng điểm danh dùng để quản lý thông tin học sinh, tài xế và chuyến xe. Cũng như ứng dụng công nghệ nhận diện khuôn mặt và so sánh vị trí để điểm danh học sinh.
- Thiết kế hệ thống sử dụng các dịch vụ của Amazon Web Services, Firebase để lưu trữ và xử lý dữ liệu liên quan đến quản lý thông tin và nhận diện khuôn mặt.

Nhìn chung, hệ thống này được thiết kế để tối ưu hóa quá trình quản lý, giám sát và đảm bảo an toàn cho học sinh trong quá trình đưa đón bằng xe buýt của nhà trường.

2.2 GIỚI THIỆU VỀ AMAZON WEB SERVICES (AWS)

2.2.1 Tổng quan về Amazon Web Services

AWS (Amazon Web Services) là một nền tảng điện toán đám mây hàng đầu, được cung cấp bởi Amazon. Nó cung cấp một loạt các dịch vụ đa dạng như storage, compute, databases, machine learning và nhiều dịch vụ khác với mô hình định giá trả theo mức sử dụng trên nền tảng Internet [4].

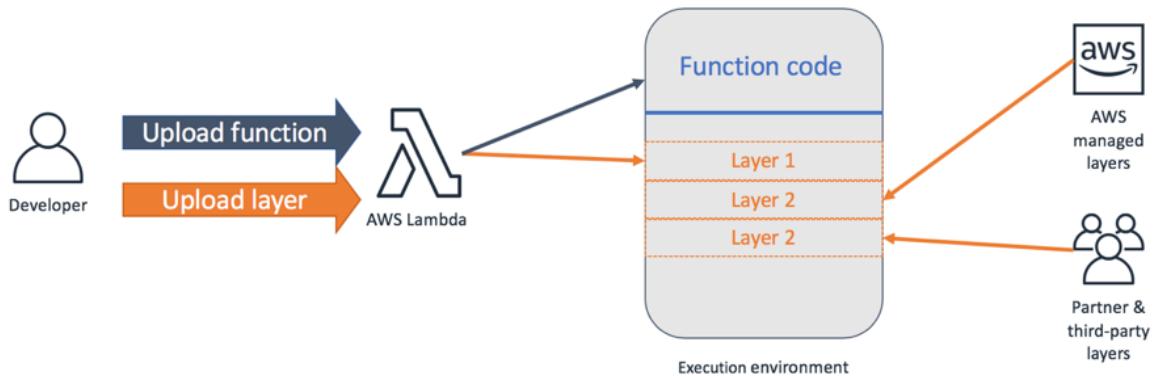


Hình 2.1: Hình ảnh minh họa về Amazon Web Services

Hệ thống hỗ trợ giám sát sử dụng 5 dịch vụ phổ biến của AWS bao gồm Lambda, DynamoDB, Rekognition, API Gateway, S3, IAM, CloudWatch để phát triển ứng dụng. AWS đóng vai trò quan trọng khi thực hiện các tính năng chính của hệ thống như lưu trữ, xử lý, nhận diện khuôn mặt trong hệ thống.

2.2.2 Tổng quan về Amazon Lambda

Amazon Lambda là một dịch vụ máy tính cho phép hệ thống chạy code mà không cần phải cung cấp hay quản lý máy chủ. Lambda chạy code trên cơ sở hạ tầng máy tính có khả năng sẵn sàng cao và thực hiện tất cả các công việc quản trị tài nguyên bao gồm bảo trì máy chủ và hệ điều hành, cung cấp khả năng mở rộng tự động và ghi nhật ký. Tất cả những gì cần làm là cung cấp code của hệ thống tại một trong những runtime ngôn ngữ mà Lambda hỗ trợ.



Hình 2.2: Hình ảnh minh họa về Amazon Lambda

Trong hệ thống Lambda Functions được dùng để thực hiện việc lấy dữ liệu, xử lý dữ liệu cũng như thực hiện chương trình nhận diện khuôn mặt sử dụng Amazon Rekognition. Bằng cách sử dụng các layer, một function sẽ có các layer khác nhau để thực hiện các nhiệm vụ khác nhau và lambda functions được phân theo các bảng trong DynamoDB và S3 buckets.

2.2.3 Tổng quan về Amazon DynamoDB

Amazon DynamoDB là một dịch vụ cơ sở dữ liệu không cấu trúc hoàn toàn được quản lý, cung cấp hiệu năng nhanh và dự đoán được với tính khả chuyển thích nghi. DynamoDB giúp hệ thống giảm bớt các gánh nặng quản trị vận hành và mở rộng cơ sở dữ liệu phân tán, cũng như cung cấp mã hóa dữ liệu tại chỗ, loại bỏ gánh nặng vận hành và tính phức tạp liên quan đến bảo vệ dữ liệu nhạy cảm.



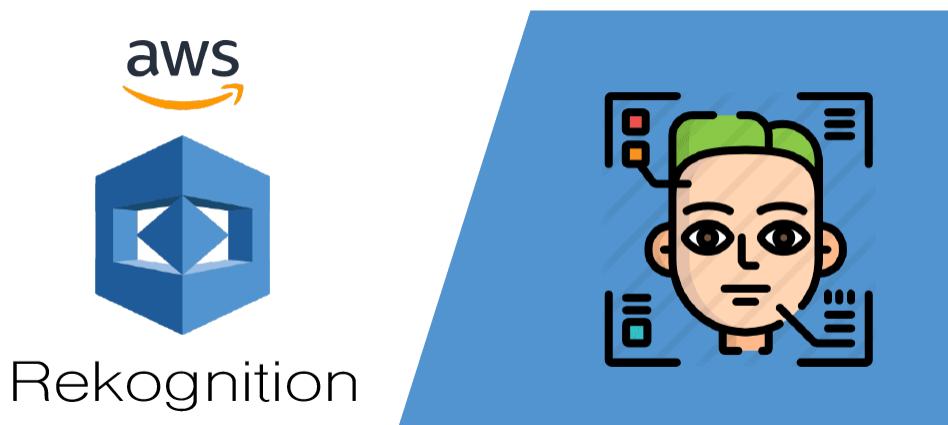
Hình 2.3: Hình ảnh minh họa về Amazon DynamoDB

Hệ thống sử dụng DynamoDB làm cơ sở dữ liệu để lưu trữ các thông tin của học sinh cũng như thông tin của tài xế. Bởi vì độ tiện lợi và có thể thao tác thông qua các dòng lệnh nên rất thích hợp để tích hợp vào phần mềm. Tối ưu hóa quá trình vận hành khi sử dụng các dịch vụ cùng một nền tảng.

2.2.4 Tổng quan về Amazon Rekognition

Amazon Rekognition là một dịch vụ phân tích ảnh và video đám mây với các tính năng thị giác máy tính tiên tiến. Dịch vụ này cung cấp một API đơn giản và dễ sử dụng, cho phép hệ thống nhanh chóng phân tích các tệp ảnh và video được lưu trữ trên Amazon S3. Rekognition dựa trên công nghệ học sâu đã được chứng minh, có khả năng mở rộng cao và được phát triển bởi các nhà khoa học của Amazon.

Các tính năng chính bao gồm tìm kiếm và phân loại ảnh/video, xác minh danh tính dựa trên khuôn mặt, nhận dạng khuôn mặt, phát hiện nội dung không an toàn, và nhận dạng văn bản. Những tính năng này mang lại nhiều khả năng ứng dụng như bảo mật, giám sát và phân tích nội dung truyền thông.



Hình 2.4: Hình ảnh minh họa về Amazon Rekognition

Dù không thao tác trực tiếp nhưng Rekognition được sử dụng thông qua các dòng lệnh được viết trong Lambda để gọi và thực hiện thao tác nhận diện khuôn mặt. Độ chính xác của model được sử dụng trong Rekognition rất vượt trội, hiệu quả cao hơn so với các model Computer Vision được publish. Ngoài ra

dịch vụ này còn được Amazon cho phép sử dụng miễn phí với giới hạn trong một năm nên Amazon Rekognition là sự lựa chọn tốt nhất cho sinh viên và các nhà đầu tư, công ty khởi nghiệp nhỏ lẻ.

2.2.5 Tổng quan về Amazon API Gateway

Amazon API Gateway là một dịch vụ của AWS cho phép các nhà phát triển tạo, công bố, duy trì, giám sát và bảo mật các API REST, HTTP và WebSocket. Với API Gateway, các nhà phát triển có thể tạo API để cung cấp quyền truy cập vào các dịch vụ AWS hoặc các dịch vụ web khác, cũng như dữ liệu được lưu trữ trong đám mây AWS. API Gateway tạo ra các API RESTful dựa trên HTTP và các API WebSocket tuân thủ giao thức WebSocket.



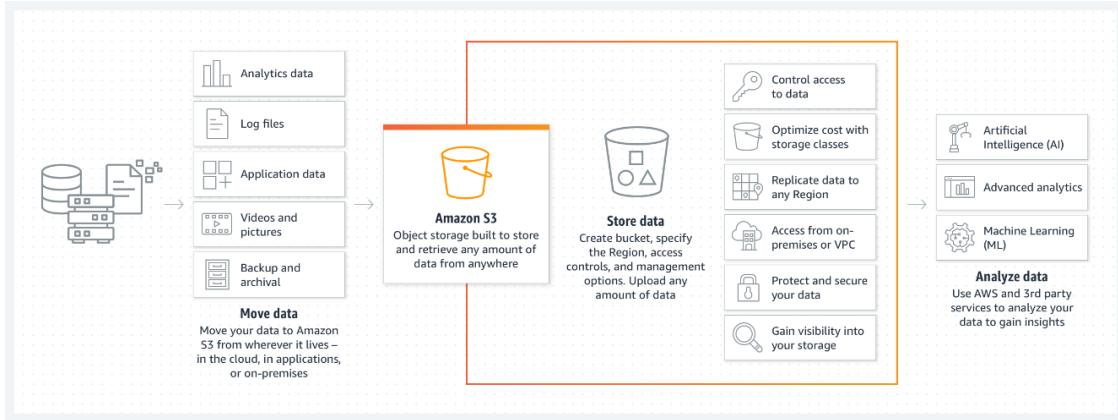
Hình 2.5: Hình ảnh minh họa về chức năng của Amazon API Gateway

Trong hệ thống, API Gateway được dùng để cấp phép cho phần mềm sử dụng các dịch vụ của Amazon Lambda như PUT, POST, GET cũng như duy trì và giám sát API REST tuân thủ giao thức WebSocket.

2.2.6 Tổng quan về Amazon S3

Amazon S3 là một dịch vụ lưu trữ dữ liệu, đối tượng có cung cấp khả năng mở rộng, sẵn có dữ liệu, bảo mật và hiệu suất hàng đầu. Khách hàng có thể sử dụng Amazon S3 để lưu trữ và bảo vệ bất kỳ dữ liệu nào cho nhiều

trường hợp sử dụng khác nhau. Amazon S3 cung cấp các tính năng quản lý để hệ thống có thể tối ưu hóa, tổ chức và định cấu hình truy cập vào dữ liệu của mình.



Hình 2.6: Hình ảnh minh họa về Amazon S3

Trong hệ thống, S3 được dùng để lưu trữ hình ảnh bao gồm hình ảnh chân dung của học sinh (thông tin học sinh) và hình ảnh điểm danh của học sinh (Ứng dụng điểm danh nhận diện khuôn mặt). Đa số các dữ liệu lưu trữ sẽ dưới dạng file .png, .jpg, .jpeg.

2.2.7 Tổng quan về AWS Identity and Access Management

AWS IAM Identity Center là dịch vụ AWS được khuyến nghị để quản lý quyền truy cập của người dùng vào các tài nguyên AWS. Đây là một nơi duy nhất để bạn có thể gán các người dùng trong tổ chức (được gọi là "các nhân sự"), cùng với việc quản lý nhất quán quyền truy cập của họ vào nhiều tài khoản và ứng dụng AWS.

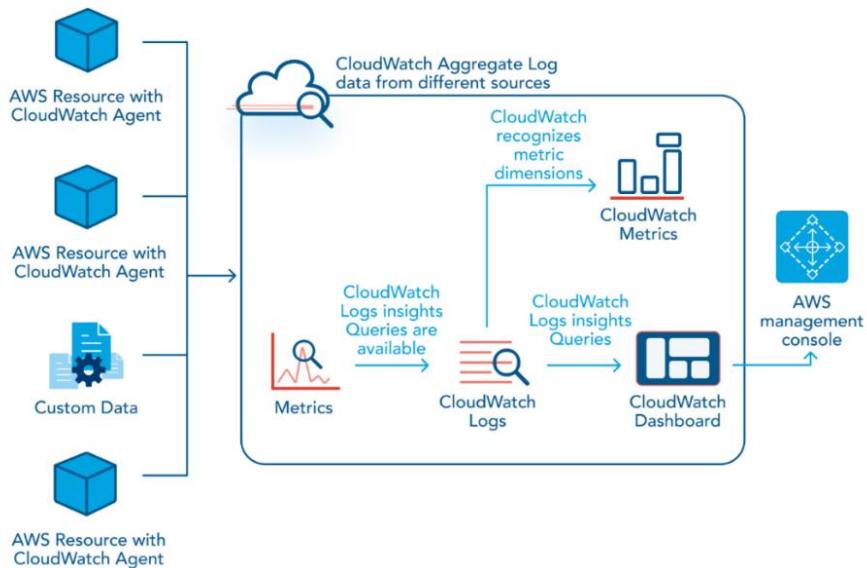
Hệ thống sử dụng IAM để cấp quyền truy cập cho người quản lý để việc quản lý trở nên nhất quán. Vì hệ thống sử dụng tận 5 ứng dụng của AWS nên việc cấp quyền là bắt buộc phải có. Đó là lí do hệ thống phải sử dụng tài khoản cá nhân có cung cấp thẻ visa vì ở tài khoản Academy không cho phép sử dụng IAM để cấp quyền truy cập người dùng.



Hình 2.7: Hình ảnh minh họa về AWS Identity and Access Management

2.2.8 Tổng quan về Amazon CloudWatch

Amazon CloudWatch theo dõi các tài nguyên Amazon Web Services (AWS) và các ứng dụng hệ thống chạy trên AWS theo thời gian thực. Có thể sử dụng CloudWatch để thu thập và theo dõi các chỉ số liên quan, là các biến hệ thống có thể đo lường cho các tài nguyên và ứng dụng. Trang chủ CloudWatch tự động hiển thị các chỉ số về mọi dịch vụ AWS mà hệ thống sử dụng, và cũng có thể tạo bảng điều khiển tùy chỉnh để hiển thị các chỉ số về các ứng dụng tùy chỉnh của riêng bạn.



Hình 2.8: Hình ảnh minh họa về Amazon CloudWatch

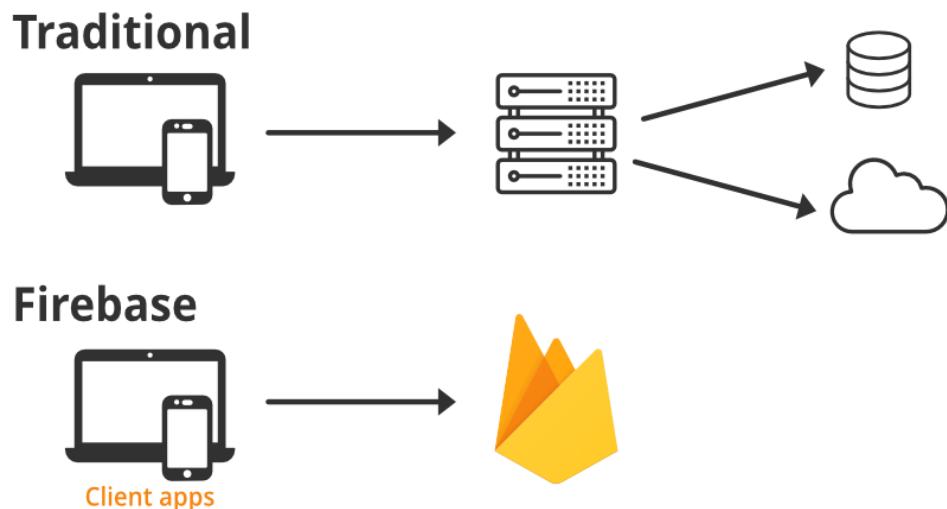
Trong hệ thống, sau khi gửi và cập nhập dữ liệu lên DynamoDB thông qua Lambda người quản lý có thể sử dụng CloudWatch logs để theo dõi trạng thái của các dòng lệnh đó. Để khi hệ thống gửi và cập nhập dữ liệu không thành công có thể truy cập để xem lỗi ở dòng lệnh nào trong lambda functions và khắc phục chúng.

2.3 GIỚI THIỆU VỀ FIREBASE

2.3.1 Tổng quan về Firebase

Firebase là một nền tảng phát triển ứng dụng của Google, giúp các nhà phát triển xây dựng, quản lý và phát triển ứng dụng của họ một cách nhanh chóng và an toàn. Không yêu cầu lập trình ở phía Firebase, điều này khiến việc sử dụng các tính năng của nó trở nên dễ dàng hơn.

Nó cung cấp các dịch vụ cho Android, iOS, Web và Unity. Là một giải pháp toàn diện để giúp các nhà phát triển xây dựng và quản lý ứng dụng một cách hiệu quả.

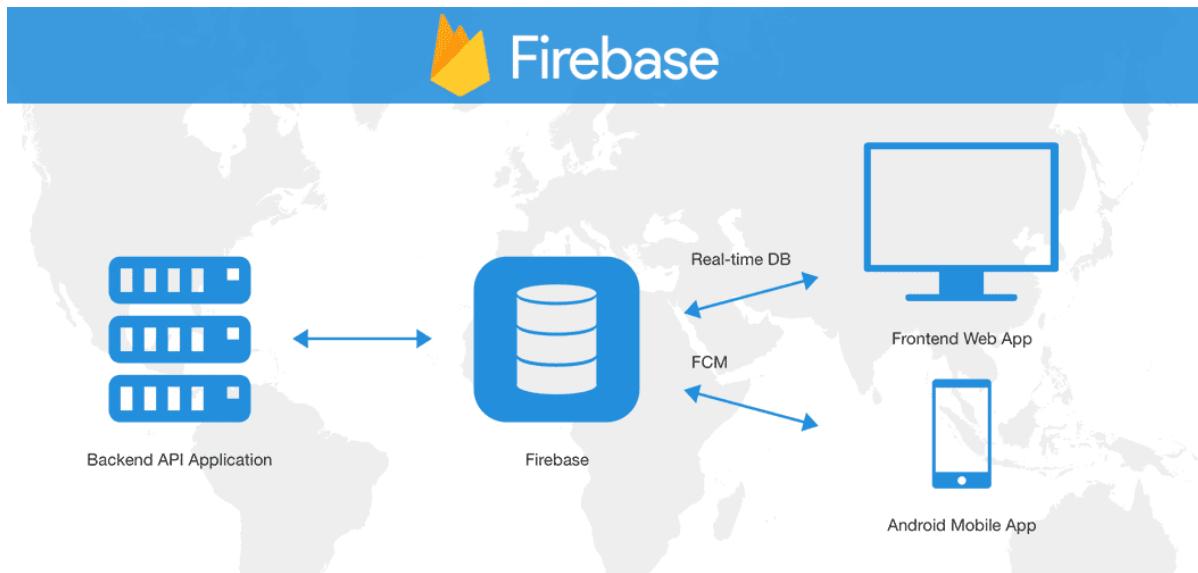


Hình 2.9: Hình ảnh minh họa về Firebase

Hệ thống sử dụng dịch vụ Realtime Database của Firebase nhằm lưu trữ dữ liệu thời gian thực cũng như làm ứng dụng trung gian cho phần cứng và phần mềm của hệ thống.

2.3.2 Tổng quan về Firebase Realtime Database

Firebase Realtime Database là một cơ sở dữ liệu được lưu trữ trên đám mây, lưu trữ dữ liệu dưới dạng JSON và được đồng bộ hóa theo thời gian thực với tất cả các client kết nối. Điều này cho phép xây dựng các ứng dụng đa nền tảng với dữ liệu được chia sẻ và cập nhật tức thời trên tất cả các client.



Hình 2.10: Hình ảnh minh họa về ứng dụng Firebase Realtime Database API Web/ App

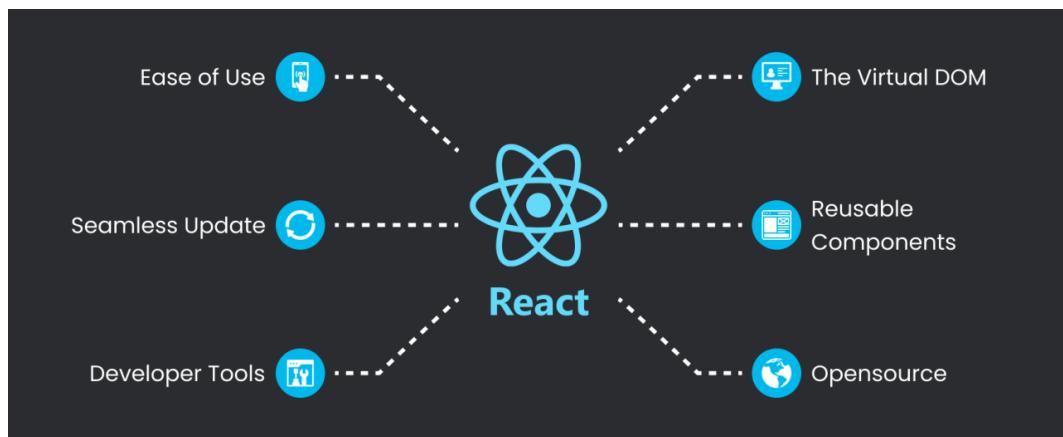
Firebase Realtime Database cung cấp một API Web giúp hệ thống tương tác với cơ sở dữ liệu trực tiếp từ trình duyệt hoặc ứng dụng web. Với API này, hệ thống có thể truy cập và thao tác dữ liệu trong cơ sở dữ liệu Realtime Database mà không cần xây dựng một server riêng. API Web hỗ trợ đồng bộ hóa dữ liệu theo thời gian thực, cho phép ứng dụng web vẫn có thể đọc, ghi và theo dõi dữ liệu ngay cả khi không có kết nối mạng.

Ngoài ra, API Web tích hợp sẵn các cơ chế xác thực và ủy quyền của Firebase, giúp quản lý quyền truy cập vào dữ liệu một cách an toàn. Với những tính năng này, API Web của Firebase Realtime Database trở thành một lựa chọn hấp dẫn cho các nhà phát triển web, cho phép họ xây dựng ứng dụng web với trải nghiệm người dùng tốt hơn và hiệu suất cao hơn.

2.4 GIỚI THIỆU VỀ CÁC FRAMEWORK SỬ DỤNG

2.4.1 Tổng quan về React.js

React là thư viện JavaScript phổ biến để xây dựng UI web. Nó dựa trên thành phần, khai báo, cho phép tái sử dụng. React tuân theo Virtual DOM, tối ưu hóa hiệu suất bằng cách giảm thiểu cập nhật DOM. Điều này giúp React hoạt động nhanh chóng và tốt với các công cụ, thư viện khác. Cụ thể, React tạo Virtual DOM thay vì trực tiếp thao tác DOM trình duyệt, thực hiện các thao tác cần thiết trong Virtual DOM trước khi áp dụng vào DOM thực, chỉ cập nhật những gì cần thiết.



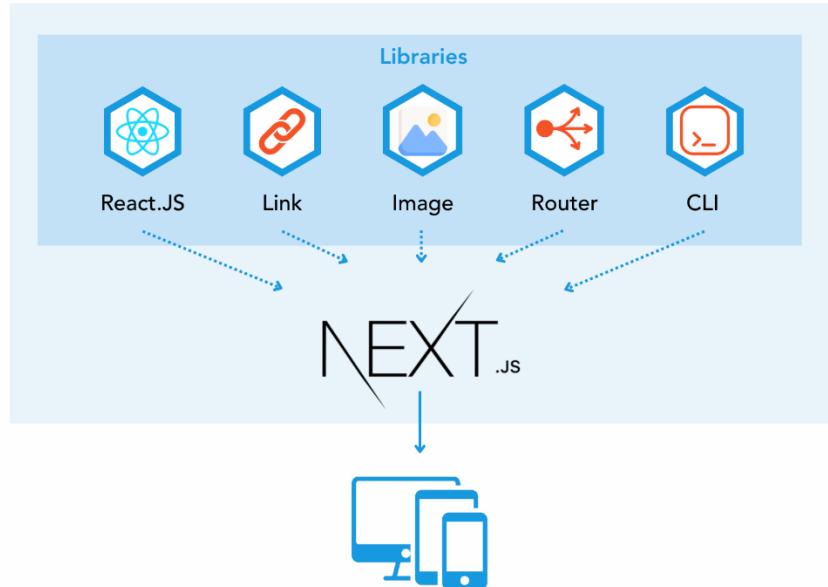
Hình 2.11: Hình ảnh minh họa về tính năng của React.js

Hệ thống được xây dựng trên nền tảng React.js, từ đó có thể tạo môi trường phát triển và tải các thư viện cần thiết để xây dựng các tính năng cơ bản của ứng dụng. Sau khi hoàn thành giai đoạn xây dựng trên React.js, hệ thống sẽ được triển khai bằng Next.js để tạo ra ứng dụng web server-rendered và tối ưu hóa SEO và hiệu suất của hệ thống.

2.4.2 Tổng quan về Next.js

Next.js là một framework của React.js dùng để xây dựng các ứng dụng web full-stack [5]. Nó cung cấp các tính năng và tối ưu hóa bổ sung, giúp hệ thống tập trung vào việc xây dựng ứng dụng thay vì mất thời gian với cấu hình. Một số tính năng chính của Next.js bao gồm định tuyến, rendering phía máy

khách và phía máy chủ, lấy dữ liệu, hỗ trợ các phương pháp kiểu, và tối ưu hóa về hình ảnh, font và script.



Hình 2.12: Hình ảnh minh họa tính năng của Next.js

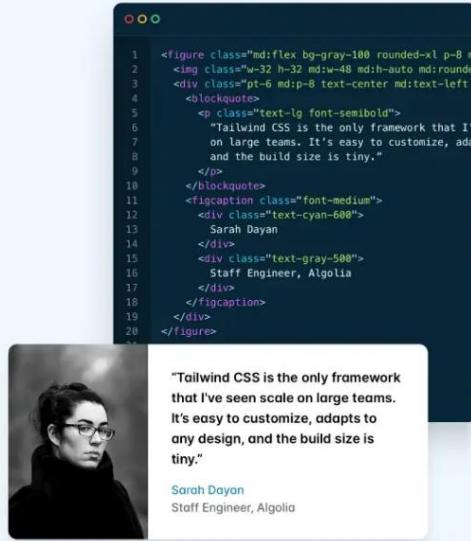
2.4.3 Tổng quan về Tailwind CSS

Tailwind CSS là một framework CSS theo phương pháp "utility-first" (tiện ích-trước), cung cấp một tập hợp các lớp tiện ích (utility classes) được thiết kế sẵn [6]. Những lớp này cho phép hệ thống tạo kiểu nhanh chóng mà không cần viết CSS tùy chỉnh, thúc đẩy sự nhất quán và khả năng mở rộng. Cách tiếp cận của Tailwind chuyển trọng tâm từ các thành phần CSS truyền thống sang các lớp chức năng, từ đó giúp các nhà phát triển xây dựng giao diện người dùng phản hồi và hấp dẫn một cách hiệu quả với nỗ lực tối thiểu.

Hệ thống được xây dựng trên nền tảng Next.js và sử dụng Tailwind CSS để phát triển phần giao diện người dùng. Là một ứng dụng web full-stack, với front-end và back-end được kết nối với các dịch vụ đám mây như Firebase và AWS. Việc tích hợp các nền tảng công nghệ hàng đầu như Next.js, Tailwind CSS, Firebase và AWS giúp hệ thống đảm bảo hiệu suất, tính bảo mật và khả năng mở rộng trong tương lai.



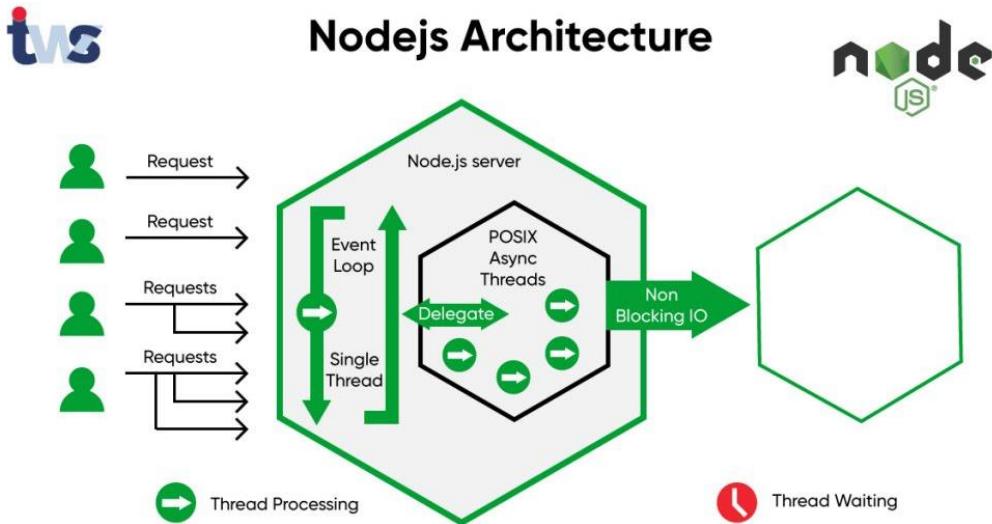
A UTILITY-FIRST CSS FRAMEWORK AND API FOR RAPID, RESPONSIVE DESIGN



Hình 2.13: Hình ảnh minh họa về Tailwind CSS

2.4.4 Tổng quan về Node.js

Node.js là một môi trường thời gian chạy JavaScript với mã nguồn mở, đa nền tảng, cho phép thực thi mã JavaScript bên ngoài trình duyệt web. Nó chạy trên công cơ JavaScript V8 của Google Chrome, nhưng hoạt động trong một quy trình duy nhất, tránh cần tạo luồng mới cho mỗi yêu cầu. Nhờ thiết kế bất đồng bộ I/O, Node.js có thể xử lý hiệu quả hàng ngàn kết nối đồng thời mà không gặp phải vấn đề với quản lý đồng thời luồng.



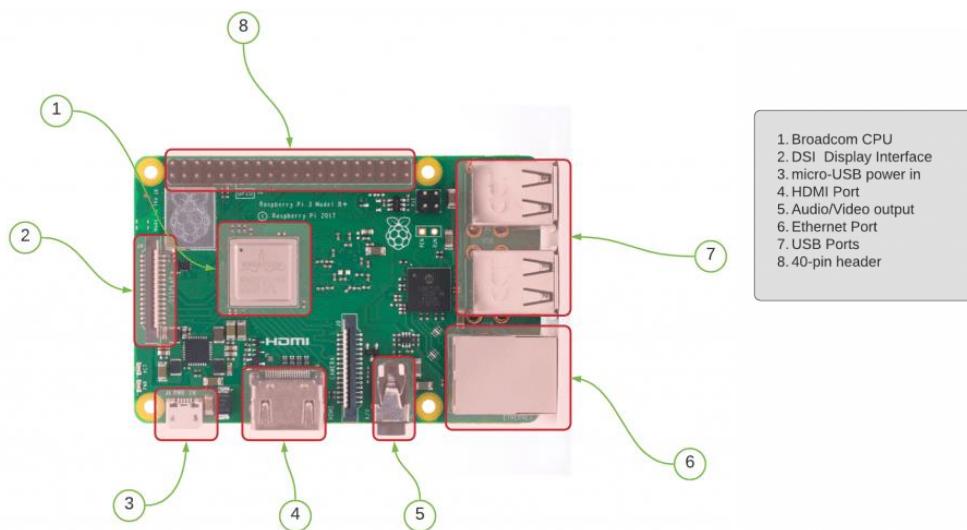
Hình 2.14: Hình ảnh minh họa về kiến trúc Node.js

Việc sử dụng Node.js trong hệ thống này mang lại nhiều lợi ích như tính đồng bộ, dễ tích hợp với JavaScript, hệ sinh thái phong phú và hỗ trợ tốt cho các yêu cầu về WebSocket, API và xử lý dữ liệu thời gian thực.

2.5 GIỚI THIỆU VỀ PHẦN CỨNG SỬ DỤNG

2.5.1 Tổng quan về Raspberry Pi 4

Raspberry Pi là một máy tính nhúng có thể kết nối với nhiều cảm biến và mô-đun khác, sử dụng vi xử lý Broadcom BCM2835 System on a Chip (SoC). Trên Raspberry Pi, chip Broadcom chứa một bộ vi xử lý ARM 700MHz và một GPU video core 4.



Hình 2.15: Hình ảnh minh họa về phần cứng Raspberry Pi 4

Raspberry Pi có rất nhiều ứng dụng, và do được rát phổ biến, số lượng cảm biến, mô-đun và thư viện mã của bên thứ ba đã phát triển vượt bậc. Các ứng dụng thực sự là vô tận, nhưng dưới đây là một số ứng dụng phổ biến:

- Máy chủ web
- Trung tâm mạng nội bộ cho Internet vạn vật (IoT)
- Điều khiển cảm biến
- Điều khiển robot
- Điều khiển thiết bị công nghiệp

- Hệ thống điều khiển tự động
- Xử lý dữ liệu

Hệ thống sử dụng Raspberry Pi 4 Module B làm khối xử lý trung tâm của phần cứng, giúp hệ thống đọc dữ liệu từ cảm biến và truyền đến các cơ sở dữ liệu cũng như lấy dữ liệu để điều khiển motor. Với tốc độ truyền tải cao, độ trễ thấp cũng như không bị ngắt khoảng nhiều, so với tầm giá raspberry là sự lựa chọn tốt nhất

2.5.2 Tổng quan về Module GPS NEO-M8N

Dòng module GPS NEO-M8 của u-blox được xây dựng trên bộ xử lý GNSS M8 hiệu suất cao. Các module NEO-M8 sử dụng khả năng tiếp nhận đồng thời tối đa ba hệ thống GNSS, cung cấp độ chính xác định vị nổi bật trong các kịch bản khó khăn, hỗ trợ QZSS, GAGAN và IMES, bảo vệ tính toàn vẹn của thông điệp, và phát hiện gian lận.

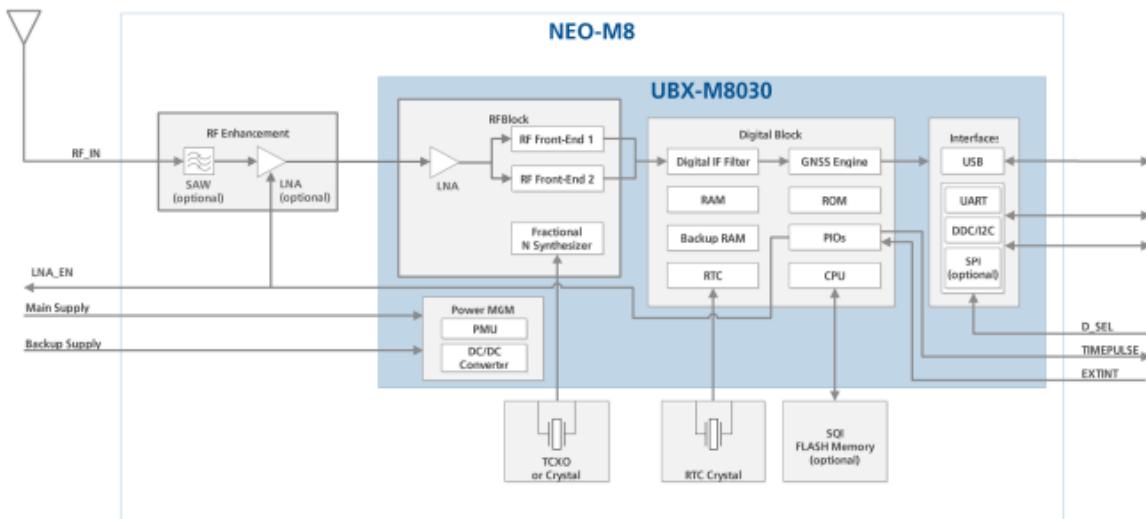


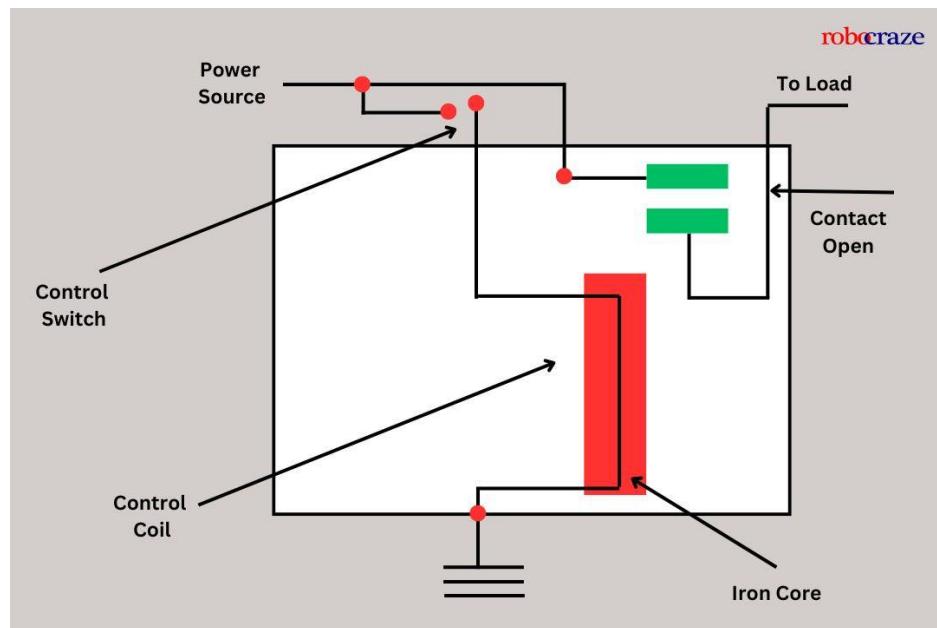
Figure 1: NEO-M8 block diagram

Hình 2.16: Hình ảnh minh họa về block diagram của module NEO-M8

So sánh với NEO-6M thì NEO-M8 nổi bật hơn với các tính năng nâng cao cùng mức giá hợp lý giúp hệ thống có thể hiệu quả hơn tối ưu hơn. Được xem là lựa chọn ưu việt khi phải cân nhắc giữa chất lượng và chi phí.

2.5.3 Tổng quan về Module Relay

Rơ le là một thiết bị có khả năng mở và đóng các mạch điện, cho phép điều khiển và tối ưu hóa dòng điện trong các ứng dụng khác nhau. Chúng hoạt động bằng cách phản ứng với các thay đổi về điện áp và cường độ dòng điện, sau đó kích hoạt các tiếp điểm để mở hoặc đóng mạch. Điều này giúp tăng tốc độ xử lý quy trình, độ chính xác và an toàn trong nhiều hệ thống.



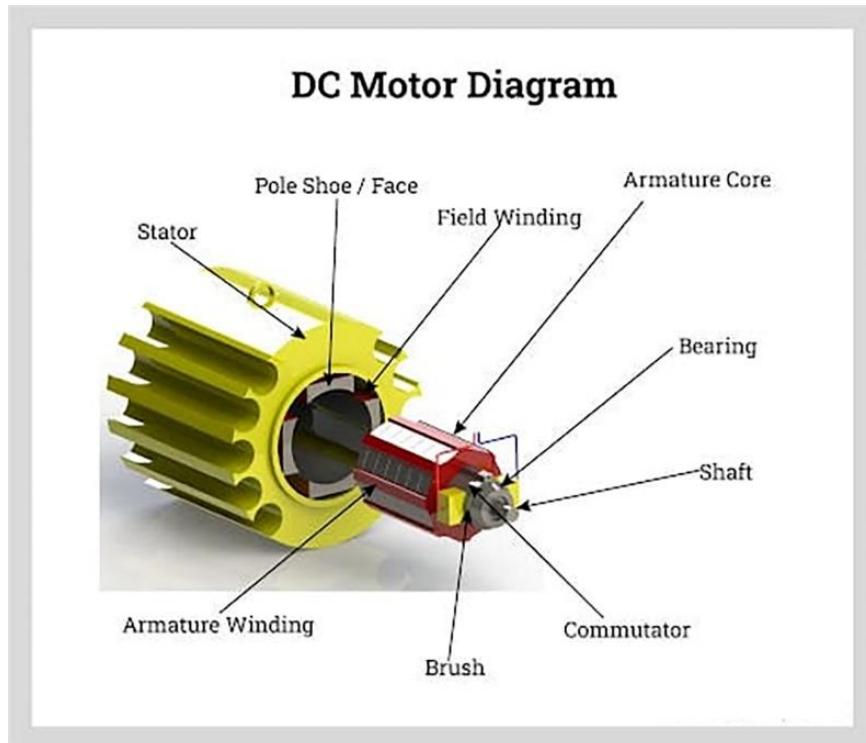
Hình 2.17: Hình ảnh minh họa về kiến trúc của Relay

Hệ thống sử dụng module relay để có thể điều khiển motor đóng mở cửa xe theo kết quả điểm danh bảo mật lớp hai (chỉ áp dụng cho phần Điểm danh xuống trạm). Thiết kế như vậy sẽ tạo ra một hệ thống điều khiển cửa xe an toàn, chỉ cho phép mở cửa khi có kết quả hợp lệ, đồng thời có thể giám sát sinh viên và cảnh báo các trường hợp không hợp lệ.

2.5.4 Tổng quan về Motor

Motor là một thiết bị điện cơ dùng để chuyển đổi năng lượng điện thành năng lượng cơ học, tạo ra chuyển động quay. Các loại motor phổ biến bao gồm động cơ DC, động cơ AC, động cơ bước và động cơ servo. Chúng được ứng dụng rộng rãi trong nhiều lĩnh vực như công nghiệp, giao thông, gia dụng,...

Các đặc tính quan trọng của motor bao gồm công suất, mô-men xoắn, tốc độ quay, hiệu suất và độ tin cậy.



Hình 2.18: Hình ảnh minh họa về cấu trúc của động cơ (motor)

Trong hệ thống này, motor được sử dụng như một mô hình tượng trưng cho cánh cửa xe đưa rước. Khi motor ngừng hoạt động, điều này tương ứng với việc cửa xe đã đóng lại. Ngược lại, khi motor bắt đầu chạy, điều này biểu trưng cho cánh cửa xe đang mở.

CHƯƠNG 3 THIẾT KẾ VÀ THI CÔNG

3.1 YÊU CẦU HỆ THỐNG

Hệ thống được thiết kế theo những yêu cầu sau đây:

- Thu thập và phân tích dữ liệu thời gian thực: Hệ thống sẽ thu thập dữ liệu từ các xe đua đón, bao gồm thông tin về vị trí và học sinh trên xe. Dữ liệu thời gian thực này sẽ được phân tích để cung cấp thông tin chi tiết cho phần hiển thị giao diện người dùng.
- Lưu trữ dữ liệu trên nền tảng đám mây: Các dữ liệu được thu thập sẽ được lưu trữ trên các nền tảng đám mây như AWS và Firebase. Điều này giúp cho hệ thống dễ dàng quản lý thông tin, xử lý dữ liệu và áp dụng chúng trong khái giao diện người dùng.
- Theo dõi vị trí bằng công nghệ GPS, đóng/ mở cửa xe tự động bằng relay: Hệ thống sẽ sử dụng công nghệ định vị GPS để theo dõi vị trí của các chuyến xe trong suốt quá trình đua đón. Dữ liệu vị trí thu thập còn được sử dụng trong chương trình con so sánh vị trí của điểm danh xuống trạm. Sau đó, relay sẽ đóng/ mở cửa xe tự động dựa theo kết quả đó.
- Tích hợp ứng dụng điểm danh và phần mềm quản lý vào khái giao diện người dùng: Hệ thống sẽ tích hợp ứng dụng điểm danh bằng nhận diện khuôn mặt hai lớp và phần mềm quản lý học sinh, tài xế và giám sát chuyến xe vào khái giao diện người dùng. Điều này giúp cho hệ thống tối ưu hơn, tiện dụng hơn giúp người dùng dễ dàng sử dụng hơn.
- Sử dụng các dịch vụ của Amazon Web Services để xử lý dữ liệu: Hệ thống sử dụng 5 dịch vụ của AWS, kết hợp chúng với nhau để thực hiện việc lưu trữ, xử lý dữ liệu theo yêu cầu về các tính năng chính của hệ thống. Bao gồm S3, DynamoDB, Rekognition, Lambda, IAM.

3.2 ĐẶC TÍCH HỆ THỐNG

3.2.1 Chức năng của hệ thống

Dựa vào những yêu cầu thiết kế hệ thống thì tôi đã xây dựng và phát triển hệ thống bao gồm những chức năng như sau:

- Chức năng đăng nhập, đăng xuất dựa trên tài khoản được cấp dành cho người quản lý.
- Chức năng thu thập dữ liệu về tọa độ vị trí của chuyến xe.
- Chức năng lưu trữ thông tin, hình ảnh của học sinh và tài xế.
- Chức năng xử lý thông tin hình ảnh để nhận diện khuôn mặt.
- Hệ thống có ứng dụng điểm danh với khóa bảo mật hai lớp đảm bảo việc học sinh lên đúng xe xuống đúng trạm, đúng vị trí nhà.
- Hệ thống tích hợp Phần mềm quản lý để người quản lý có thể giám sát chặt chẽ từng chuyến xe, số lượng học sinh có trên xe, vị trí xe để khi có trực trặc, sự cố bất ngờ xảy ra có thể thông báo ngay đến phụ huynh học sinh cũng như hỗ trợ ngay khi cần thiết. Đảm bảo được sự an toàn tuyệt đối với học sinh, phần nào giảm bớt được lo lắng cho phụ huynh.

3.2.2 Mô hình tổng thể hệ thống

Mô hình hoạt động tổng thể của hệ thống hỗ trợ giám sát học sinh trên xe đưa rước:

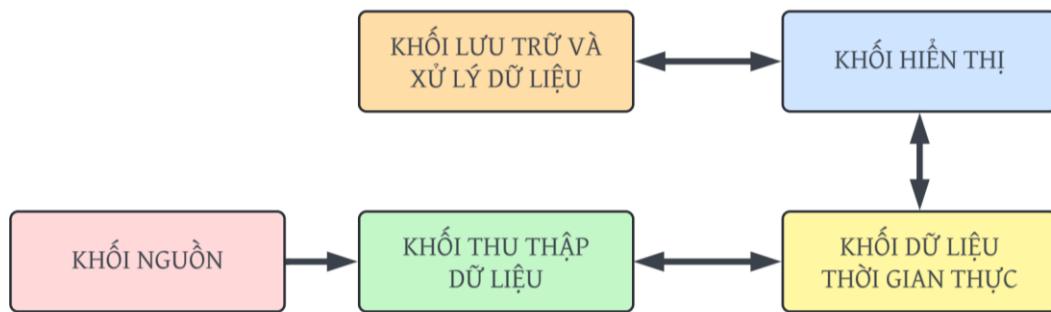


Hình 3.1: Mô hình tổng thể hệ thống

Dựa trên những yêu cầu chính của hệ thống thì mô hình tổng thể được chia thành:

- Raspberry Pi 4B: Thu thập dữ liệu GPS và truyền dữ liệu đến Firebase, lấy dữ liệu từ Firebase về để đóng/ mở cửa ra vào.
- Firebase: Lấy và truyền dữ liệu đến Raspberry và trang web
- Web: Cung cấp giao diện Phần mềm quản lý và Ứng dụng điểm danh
- Amazon Web Services: Lưu trữ, truyền và xử lý dữ liệu từ trang web

3.2.3 Sơ đồ khái niệm

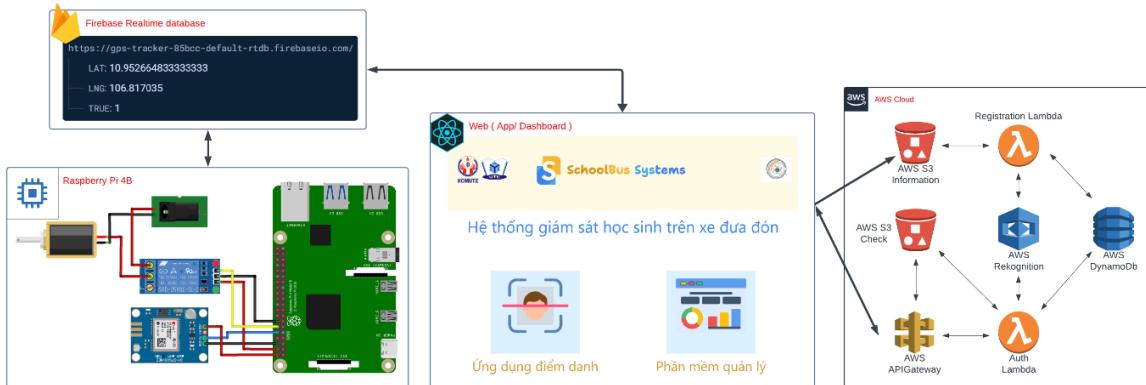


Hình 3.2: Sơ đồ khái niệm

- Khối nguồn: Cấp nguồn từ máy cấp nguồn cho khối thu thập dữ liệu (Raspberry).
- Khối thu thập dữ liệu (Raspberry Pi 4B): Sử dụng cảm biến GPS để định vị vị trí chuyến xe sau đó gửi dữ liệu đến khối dữ liệu thời gian thực, đồng thời lấy dữ liệu đúng/ sai của bảo mật lớp hai để đóng/ mở cửa xe.
- Khối lưu trữ dữ liệu thời gian thực (Firebase): Nhận và truyền dữ liệu đồng thời cho cả khái niệm hiển thị và khái niệm thu thập dữ liệu.
- Khối lưu trữ và xử lý dữ liệu (AWS): Nhận, truyền và xử lý dữ liệu hình ảnh, thông tin, thông số mà khái niệm hiển thị trả về. Sử dụng dịch vụ Amazon Rekognition có sẵn để nhận diện khuôn mặt làm bảo mật lớp một cho hệ thống.

- Khối hiển thị (Web): Hiển thị giao diện người dùng với hai tính năng là Ứng dụng điểm danh và Phần mềm quản lý, giúp người dùng có thể điểm danh học sinh với bảo mật hai lớp chính xác và luôn nắm bắt thông tin, quản lý giám sát chuyên xe và học sinh liên tục.

3.2.4 Hoạt động của hệ thống



Hình 3.3: Sơ đồ hoạt động của hệ thống

Đầu tiên người quản lý sẽ sử dụng Phần mềm quản lý, nhập tên đăng nhập và mật khẩu hợp lệ tại trang Đăng nhập để có thể truy cập vào trang quản lý. Sau đó người quản lý sẽ bắt đầu thêm thông tin của các học sinh cũng như thêm thông tin của tài xế. Dữ liệu đó sẽ được truyền đến AWS để lưu trữ, sau đó sẽ được truy xuất lại từ cơ sở dữ liệu của AWS và hiển thị trên các phần giao diện giám sát và quản lý.

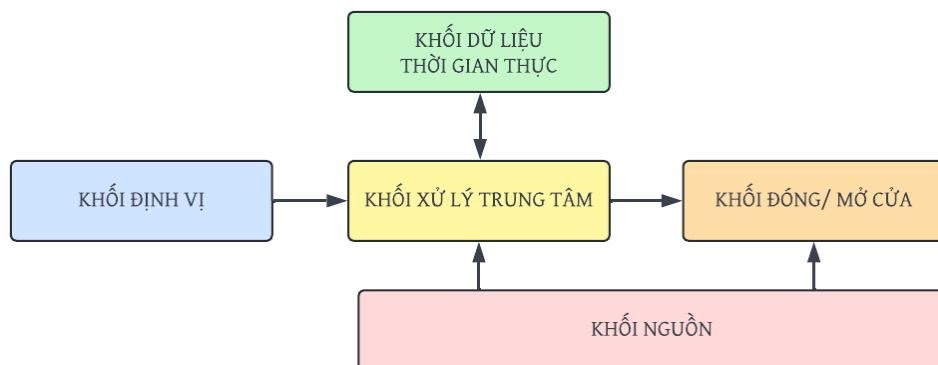
Tiếp đến người dùng sẽ thiết lập phần cứng trên phương tiện di chuyển (xe đưa rước), cấp nguồn cho thiết bị và bắt đầu thu thập dữ liệu từ cảm biến và truyền dữ liệu đó lên Firebase theo thời gian thực.

Sau đó, người dùng sẽ sử dụng Ứng dụng điểm danh trên các chuyến xe đưa rước thông qua các thiết bị di động có truy cập wifi. Ứng dụng điểm danh sẽ gửi dữ liệu về hình ảnh điểm danh của học sinh đến AWS đồng thời cập nhật trạng thái của học sinh trên AWS. Đối với Điểm danh xuống trạm sau khi truyền các dữ liệu đến AWS, phần mềm sẽ lấy dữ liệu vị trí của chuyến xe từ Firebase

và dữ liệu vị trí nhà của học sinh đã được nhận diện đúng khuôn mặt từ AWS và so sánh với nhau với sai số khoảng cách là 5m.

Cuối cùng là cập nhập kết quả bảo mật lớp hai lên Firebase để phần cứng có thể truy xuất để thực hiện đóng/ mở cửa xe.

3.3 THIẾT KẾ PHẦN CỨNG

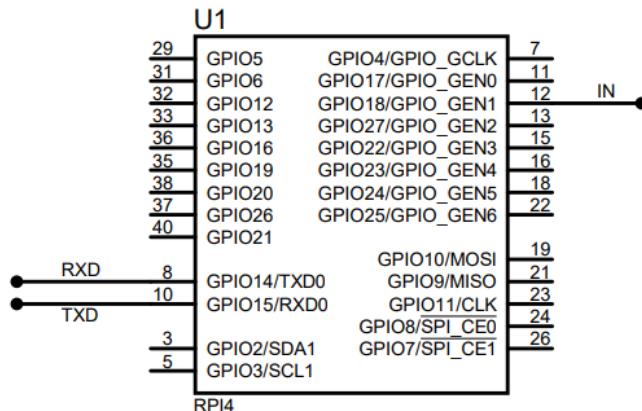


Hình 3.4: Sơ đồ khái niệm của phần cứng

- *Khối nguồn:* Cấp nguồn từ máy cấp nguồn cho khối thu thập dữ liệu (Raspberry) và khối đóng/ mở cửa.
- *Khối định vị:* Sử dụng module định vị GPS NEO-M8 với kích thước nhỏ gọn tiện lợi và độ nhạy cao chính xác thích hợp cho việc giám sát và định vị phương tiện.
- *Khối đóng/ mở cửa:* Sử dụng module relay 5V và motor 12V có kết nối nguồn 12V để đóng/ mở cửa khi khối xử lý trung tâm trả về tín hiệu.
- *Khối xử lý trung tâm:* Hệ thống sử dụng một máy tính nhúng thu nhỏ Raspberry Pi 4B để lấy dữ liệu từ khối định vị truyền đến khối dữ liệu thời gian thực cũng như lấy dữ liệu từ khối thời gian thực để tiến hành thực hiện khối đóng/ mở cửa.
- *Khối dữ liệu thời gian thực:* Sử dụng Firebase một nền tảng điện toán đám mây cho phép lưu trữ dữ liệu thời gian thực. Firebase sẽ cung cấp API để khối xử lý trung tâm có thể truy cập vào cơ sở dữ liệu và truyền dữ liệu, lấy dữ liệu trực tiếp.

3.3.1 Khối xử lý trung tâm

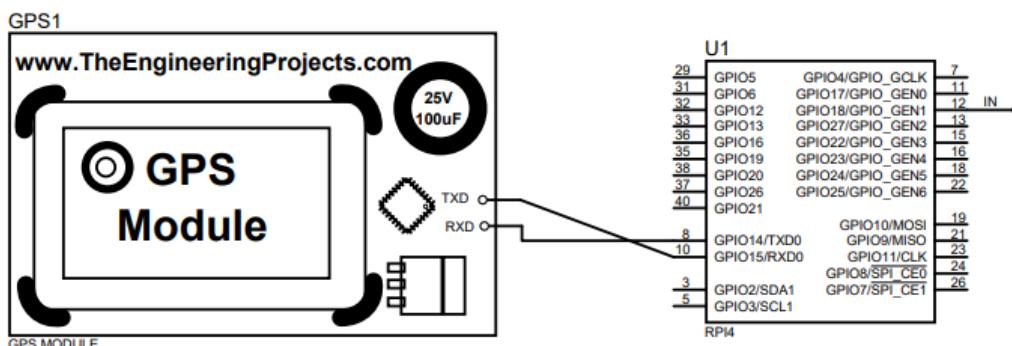
Hệ thống sử dụng Raspberry Pi 4B để làm khói xử lý trung tâm vì nó là một chiếc máy tính nhúng nhỏ gọn tiện lợi tích hợp nhiều tính năng cũng như hỗ trợ giao diện người dùng như một hệ điều hành cho phép cài đặt nhiều ứng dụng khác nhau. Có thể đọc và gửi dữ liệu liên tục với độ trễ khi truyền đi ít hơn 5 giây đảm bảo số liệu luôn được cập nhật chính xác và nhanh nhất.



Hình 3.5: Sơ đồ nguyên lý Raspberry Pi 4B

Ngoài ra Raspberry Pi 4B được trang bị CPU quad-core 64-bit ARM Cortex-A72 là dòng chip mạnh nhất so với các phiên bản khác cùng dòng, với RAM 4GB hỗ trợ xử lý nhiều tác vụ cùng lúc với độ nhạy, tốc độ xử lý nhanh hơn. Rất thích hợp để sử dụng trong hệ thống do không chỉ đọc dữ liệu và điều khiển các cảm biến mà còn truyền dữ liệu lên Firebase thời gian thực.

3.3.2 Khối đọc định vị



Hình 3.6: Sơ đồ nguyên lý kết nối module NEO-M8 với Raspberry Pi 4B

Hệ thống sử dụng module NEO-M8 GPS một dòng cảm biến GPS phổ biến nhất hiện nay. Giá thành sinh viên nhưng độ chính xác và tốc độ có thể nói là quá xuất sắc so với tầm giá. Module sử dụng hệ thống vệ tinh Mỹ và chuẩn giao tiếp UART/TTL để kết nối dù có sai số nhưng độ chênh lệch không cao nằm trong khoảng bán kính 5m.

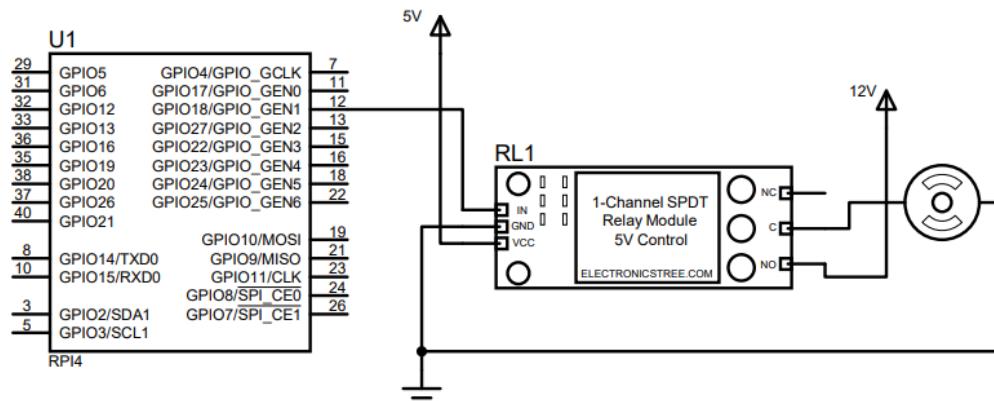
Bảng 3.1: Sơ đồ kết nối chân của module NEO-M8 với Raspberry Pi 4B

Module NEO-M8 GPS	Raspberry Pi 4B
Tx	GPIO15
Rx	GPIO14
5V	5V
GND	GND

NEO - M8 là một phiên bản cải tiến tốt hơn so với M6, theo trải nghiệm tôi thấy độ chính xác và độ nhạy tốt hơn so với M6 mà giá thành lại không chênh lệch quá nhiều rất thích hợp cho hệ thống.

3.3.3 Khởi động/ mở cửa

Hệ thống sử dụng một relay một kênh hoạt động ở mức điện áp 5V để điều khiển motor 12V. Cụ thể, khi relay được kích hoạt (đóng mạch), nó sẽ cung cấp nguồn điện 12V cho motor, khiến motor bắt đầu chạy và mô phỏng trạng thái cửa xe đang mở. Ngược lại, khi relay ngừng hoạt động (ngắt mạch), nguồn điện 12V sẽ bị cắt khỏi motor, khiến motor ngừng chạy và biểu trưng cho cửa xe đã đóng lại.



Hình 3.7: Sơ đồ nguyên lý kết nối Relay Module, Motor với Raspberry Pi 4

Việc sử dụng relay một kênh 5V để điều khiển motor 12V giúp tách biệt mạch điều khiển (5V) khỏi mạch động lực (12V), đảm bảo an toàn và tin cậy cho hệ thống. Khi nhận tín hiệu điều khiển từ các bộ phận khác, relay sẽ nhanh chóng đóng hoặc ngắt mạch động lực của motor, cho phép mô phỏng trạng thái mở/đóng cửa xe một cách chính xác.

Bảng 3.2: Sơ đồ kết nối chân của module Relay, Motor với Raspberry Pi 4 và Nguồn 12V

Relay	Raspberry Pi 4B	Motor	Nguồn 12V
IN	GPIO18		
GND	GND		
VCC	VCC		
C		VCC	
NO			12V
		GND	GND

3.3.4 Khởi nguồn

Hệ thống sử dụng 2 nguồn điện khác nhau:

+ Nguồn 12V được dùng để cung cấp điện cho motor và đầu ra của relay.

Bảng 3.3: Thông số về dòng điện, điện áp và công suất tiêu thụ trung bình của Motor.

Tên linh kiện	Điện áp hoạt động	Dòng điện tiêu thụ	Công suất tiêu thụ trung bình
Motor	6~18V	240mA	4W

+ Hệ thống còn sử dụng thêm một nguồn điện 5.1V để cấp điện cho Raspberry Pi 4B và các thiết bị kết nối với nó (NEO-M8 GPS và Relay).

Bảng 3.4: Thông số về dòng điện, điện áp và công suất tiêu thụ của các linh kiện còn lại.

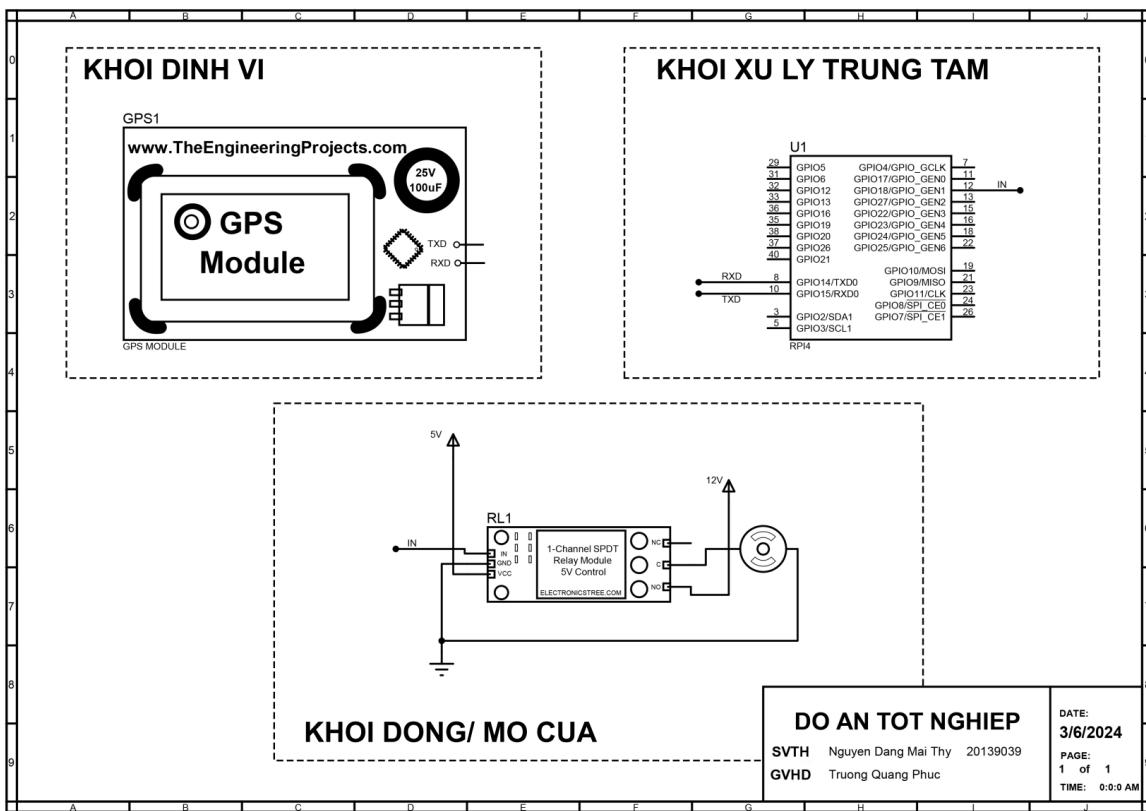
Tên linh kiện	Điện áp hoạt động	Dòng điện tiêu thụ	Công suất tiêu thụ trung bình
Raspberry Pi 4B	5.1V	3A	15.3W
Module NEO-M8 GPS	5V	45mA	0.225W
Module Relay	5V	80mA	0.4W

Việc sử dụng nguồn 5.1V từ adapter chính hãng của Raspberry Pi 4B giúp đảm bảo cấp điện ổn định và tin cậy cho các thành phần quan trọng của hệ thống, như bo mạch điều khiển chính, module GPS và phần điều khiển của relay.

+ Dựa trên các thông số của các linh kiện thì công suất tiêu thụ trung bình của toàn hệ thống là 4.981W.

Với mức công suất tiêu thụ trung bình này, hệ thống có thể hoạt động ổn định và hiệu quả. Các nguồn điện 12V, 5V và 5.1V được lựa chọn và thiết kế phù hợp để cung cấp đủ công suất cho các thành phần trong hệ thống. Điều này đảm bảo hệ thống có thể vận hành liên tục mà không bị gián đoạn do thiếu nguồn điện. Ngoài ra, việc tách biệt các mức điện áp cũng giúp tăng tính an toàn và bảo vệ các thành phần điện tử quan trọng.

3.3.5 Sơ đồ nguyên lý



Hình 3.8: Sơ đồ nguyên lý kết nối phần cứng hệ thống

Khối định vị (Module GPS NEO-M8)

- Khối này sẽ liên tục lấy dữ liệu về vị trí của xe đưa đón và gửi lên khối xử lý trung tâm.

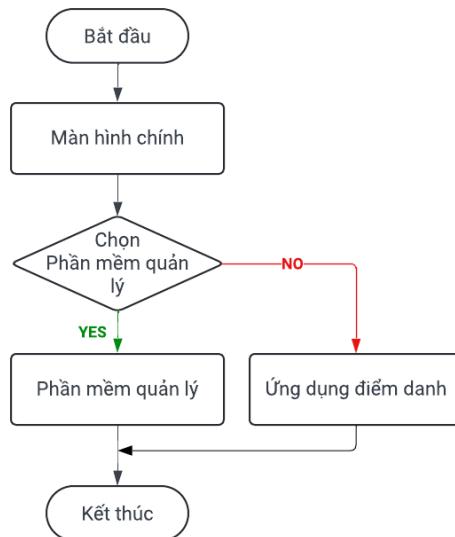
Khối đóng/mở cửa (Module Relay và Motor)

- Khối này sử dụng module relay 5V và motor 12V để khi khối xử lý trung tâm gửi tín hiệu lệnh, khối này sẽ tiến hành đóng/mở cửa tương ứng.

Khối xử lý trung tâm (Raspberry Pi 4B)

- Khối này đóng vai trò trung tâm, thu nhận dữ liệu từ khối định vị GPS và so sánh với dữ liệu thời gian thực.
- Nếu vị trí phương tiện khớp với vị trí trạm, khối xử lý sẽ gửi lệnh cho khối đóng/mở cửa để mở cửa cho phương tiện ra/vào.

3.4 THIẾT KẾ PHẦN MỀM

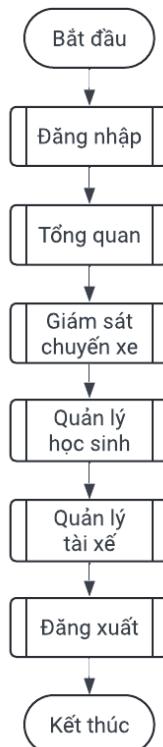


Hình 3.9: Lưu đồ giải thuật chính của phần mềm.

Phần mềm sẽ có hai phần chính đó là Phần mềm quản lý và Ứng dụng điểm danh. Nếu thiết bị được gắn trên xe sẽ mở Ứng dụng điểm danh còn khi người dùng muốn giám sát quản lý thì sẽ mở Phần mềm quản lý.

Bài báo cáo đang trình bày trường hợp nhận học sinh từ trường và đưa học sinh về nhà (ở trường hợp này mới phải cần chú trọng đến việc quản lý học sinh và cần điểm danh hai lớp do ở trường hợp này học sinh rất dễ xuống nhầm trạm và quên xuống trạm)

3.4.1 Phần mềm quản lý



Hình 3.10: Lưu đồ giải thuật của Phần mềm quản lý.

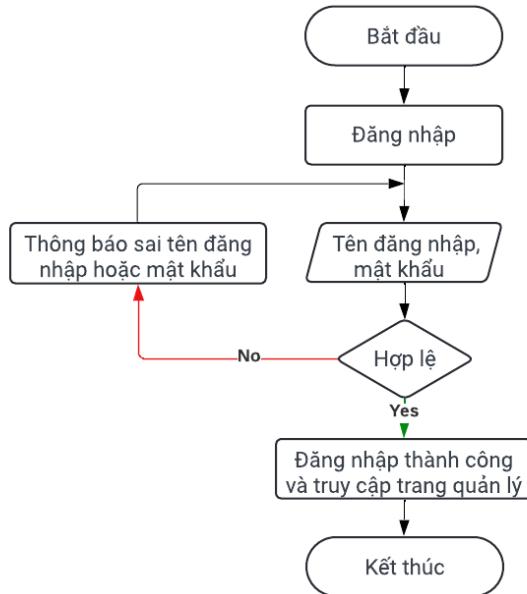
Phần mềm quản lý bao gồm nhiều chương trình con khác nhau. Khi người quản lý truy cập vào phần mềm, đầu tiên họ sẽ được hiển thị giao diện Đăng nhập. Sau khi đăng nhập thành công, người quản lý sẽ được chuyển đến giao diện chính của trang quản lý. Bao gồm các phần như Tổng quan, Giám sát chuyến xe, Quản lý học sinh, Quản lý tài xế, Đăng xuất.

Phần mềm quản lý cung cấp một giao diện trực quan và đa chức năng, giúp người quản lý dễ dàng theo dõi, quản lý và điều hành hoạt động của hệ thống một cách hiệu quả.

➤ Chương trình con Đăng nhập

Thiết lập lưu đồ giải thuật đăng nhập tài khoản cho người quản lý khi sử dụng Phần mềm quản lý, khi người dùng chọn vào giao diện Phần mềm quản lý thì trang Đăng nhập sẽ hiện ra trước, người dùng phải nhập đúng tên đăng nhập và mật khẩu để có thể truy cập vào trang quản lý.

Nếu tên đăng nhập hoặc mật khẩu người dùng nhập sai không hợp lệ thì trên giao diện sẽ hiển thị thông báo. Nếu người dùng nhập đúng các thông tin đó thì sẽ được chuyển sang giao diện trang quản lý, hiển thị giao diện Tổng quan, từ đó người quản lý có thể chọn đến các phần quản lý khác.



Hình 3.11: Lưu đồ giải thuật của chương trình con Đăng Nhập

➤ Chương trình con Tổng quan

Phần giao diện Tổng quan sẽ chỉ hiển thị thông tin cung cấp cho người quản lý những thông tin cơ bản và tổng quan về đồ án cũng như hệ thống quản lý vận tải học sinh. Đây sẽ là điểm khởi đầu để người quản lý, người dùng có thể nhanh chóng nắm bắt được cách hoạt động của hệ thống.

➤ Chương trình con Giám sát chuyến xe

Thiết lập lưu đồ giải thuật của phần giao diện giám sát chuyến xe, khi người quản lý ấn vào giao diện sẽ hiển thị thông tin học sinh, trạng thái của học sinh (đã lên xe, trên xe, đã xuống xe) liên tục thay đổi theo thời gian, dữ liệu này sẽ được lấy từ AWS.

Ngoài ra giao diện này còn hiển thị vị trí chuyến xe trên bản đồ sử dụng google maps theo thời gian thực với dữ liệu được lấy từ Firebase (Dữ liệu được lấy từ cảm biến gps truyền tới Firebase).



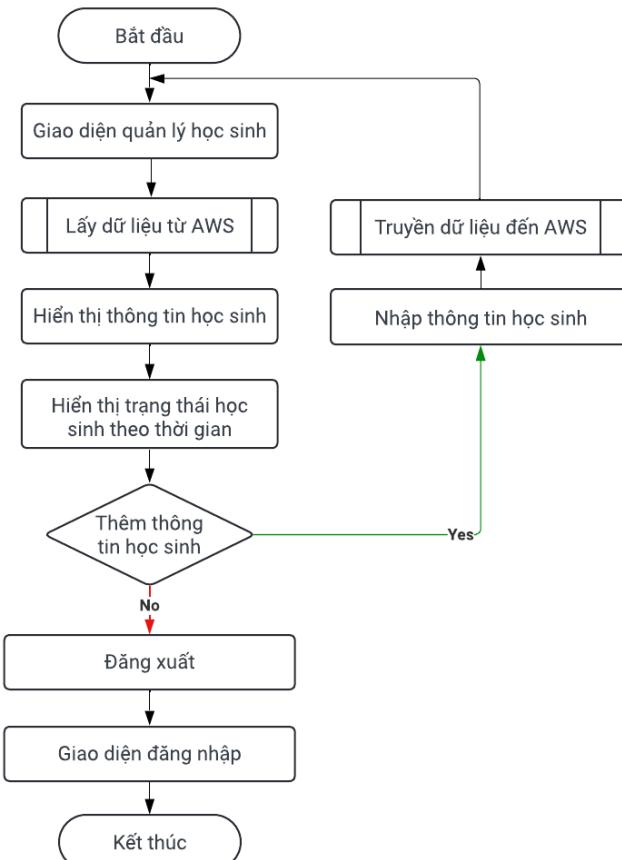
Hình 3.12: Lưu đồ giải thuật của chương trình con Giám sát chuyến xe.

Giao diện giám sát chuyến xe còn có tính năng cho phép người quản lý có thể lọc thông tin học sinh theo chuyến xe, theo trạng thái của học sinh giúp việc quản lý dễ dàng hơn, giám sát chặt chẽ hơn.

Như vậy, qua giao diện giám sát chuyến xe, người quản lý sẽ có thể theo dõi được thông tin chi tiết về từng học sinh, trạng thái của các tôi cũng như vị trí chuyến xe một cách chính xác và kịp thời. Điều này sẽ giúp việc quản lý và giám sát quá trình vận chuyển học sinh trở nên hiệu quả hơn.

➤ Chương trình con Quản lý học sinh

Thiết lập lưu đồ giải thuật của phần giao diện quản lý học sinh, khi người quản lý ấn vào giao diện sẽ hiển thị thông tin học sinh, trạng thái của học sinh (đã lên xe, trên xe, đã xuống xe) liên tục thay đổi theo thời gian, dữ liệu này cũng được lấy từ AWS.



Hình 3.13: Lưu đồ giải thuật của chương trình con Quản lý học sinh.

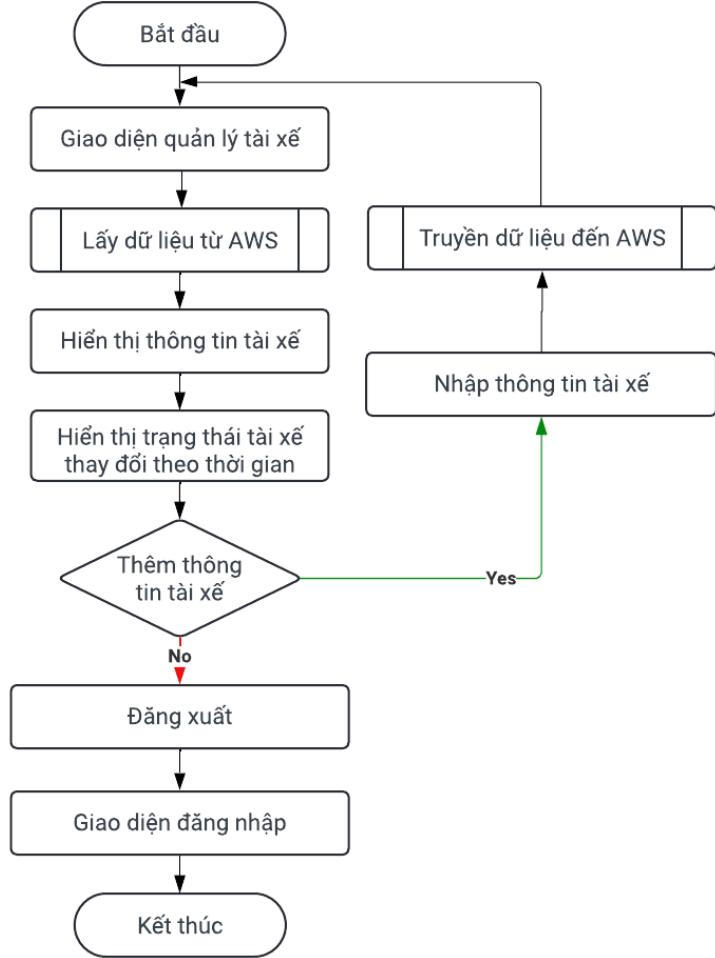
Ngoài ra giao diện này còn hỗ trợ tính năng thêm thông tin học sinh (hình ảnh và thông tin cá nhân) dữ liệu này sẽ được truyền đến AWS để lưu trữ và sẽ được truy xuất để hiển thị ở cả Ứng dụng điểm danh và Phần mềm quản lý.

Giao diện Quản lý học sinh còn có tính năng cho phép người quản lý có thể lọc thông tin học sinh theo chuyến xe giúp việc quản lý dễ dàng hơn.

Với những tính năng này, giao diện quản lý học sinh sẽ trở thành một công cụ hữu ích, giúp người quản lý nắm bắt và quản lý thông tin học sinh một cách hiệu quả.

➤ Chương trình con Quản lý tài xế

Tương tự Quản lý học sinh, lưu đồ giải thuật của phần giao diện quản lý tài xế cũng sẽ hiển thị thông tin của tài xế và trạng thái hoạt động của tài xế được cập nhật liên tục theo thời gian, dữ liệu này cũng được lấy từ AWS.



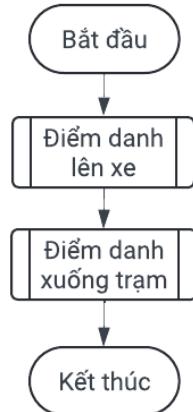
Hình 3.14: Lưu đồ giải thuật của Chương trình con Quản lý tài xế.

Phần giao diện này cũng được hỗ trợ tính năng thêm thông tin tài xế (chỉ bao gồm thông tin cá nhân) dữ liệu này sẽ được truyền đến AWS để lưu trữ và chỉ được truy xuất khi chuyến xe gấp ván đề, sự cố và phải có sự cho phép từ người quản lý.

➤ *Chương trình con Đăng xuất*

Khi người dùng ấn vào phần Đăng xuất thì sẽ đưa người dùng về lại phần giao diện Đăng nhập ban đầu. Việc này đảm bảo tính bảo mật, chỉ những người có quyền truy cập mới có thể sử dụng hệ thống. Ngoài ra, các thông tin, dữ liệu của người dùng cũng sẽ được giữ an toàn khi họ đăng xuất khỏi hệ thống.

3.4.2 Ứng dụng điểm danh



Hình 3.15: Lưu đồ giải thuật của Ứng dụng điểm danh.

Ứng dụng điểm danh có hai phần là Điểm danh lên xe và Điểm danh xuống trạm. Điểm danh lên xe được sử dụng khi học sinh rời khỏi trường và lên xe đưa đón, giúp ghi lại thời gian và cung cấp hình ảnh việc học sinh đã an toàn lên xe và bắt đầu chuyến đi. Tương tự, điểm danh xuống trạm được thực hiện khi học sinh đến điểm trả về nhà và xuống xe. Quá trình này ghi lại thời gian và hình ảnh cụ thể khi học sinh kết thúc chuyến đi.

Việc chia thành hai phần "Điểm danh lên xe" và "Điểm danh xuống trạm" giúp hệ thống quản lý di chuyển của học sinh một cách chi tiết và chính xác. Dữ liệu được thu thập từ hai thời điểm này sẽ cung cấp thông tin về toàn bộ hành trình của học sinh, từ lúc lên xe đến lúc xuống trạm.

Điều này không chỉ giúp theo dõi an toàn của học sinh mà còn hỗ trợ việc quản lý, báo cáo và phân tích dữ liệu di chuyển của học sinh một cách hiệu quả.

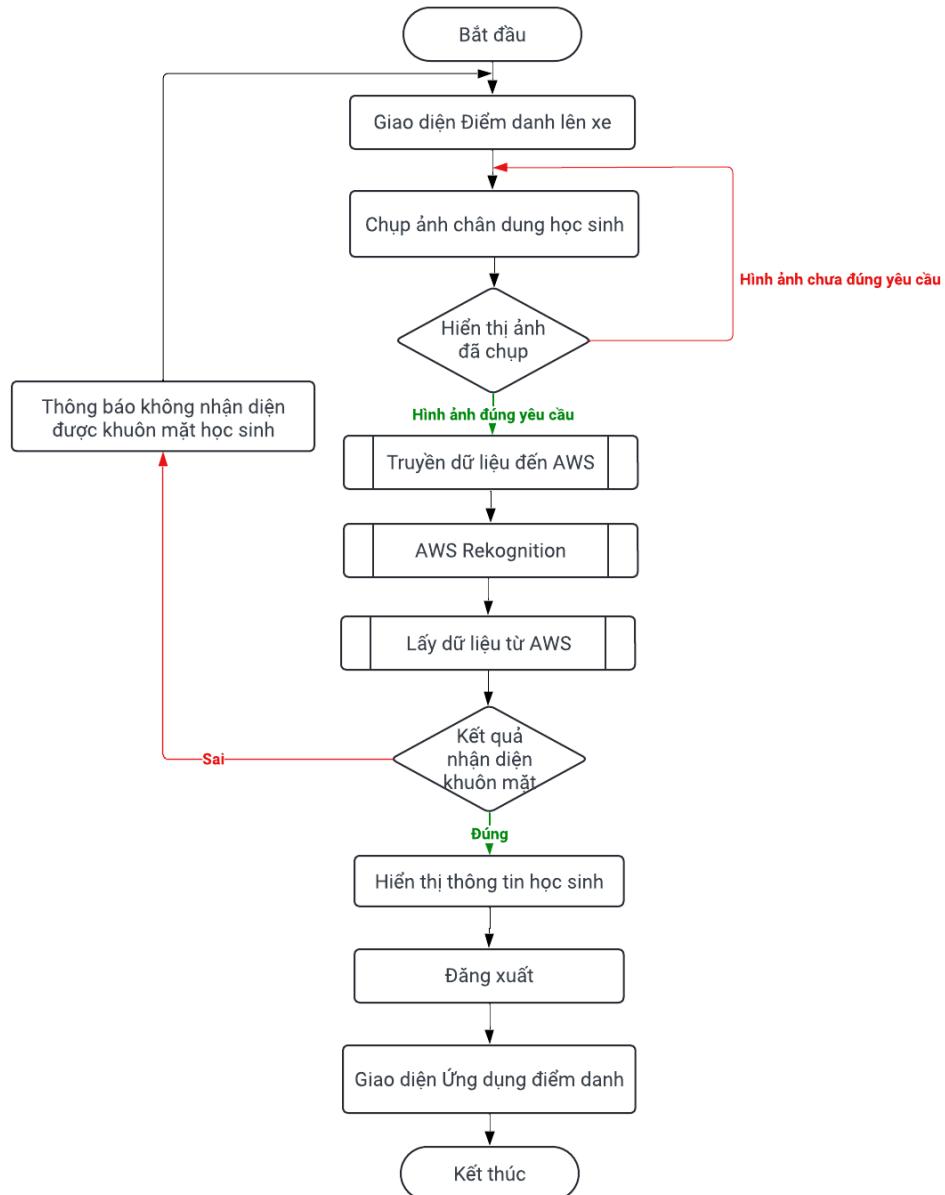
➤ *Chương trình con Điểm danh lên xe*

Bắt đầu chương trình con điểm danh lên xe học sinh sẽ phải tự thao tác chụp ảnh chân dung của mình.

Sau khi bấm nút chụp, ảnh sẽ được hiển thị lên giao diện nếu hình ảnh chưa đúng yêu cầu sẽ phải chụp lại ảnh, còn nếu hình ảnh đúng yêu cầu sẽ được

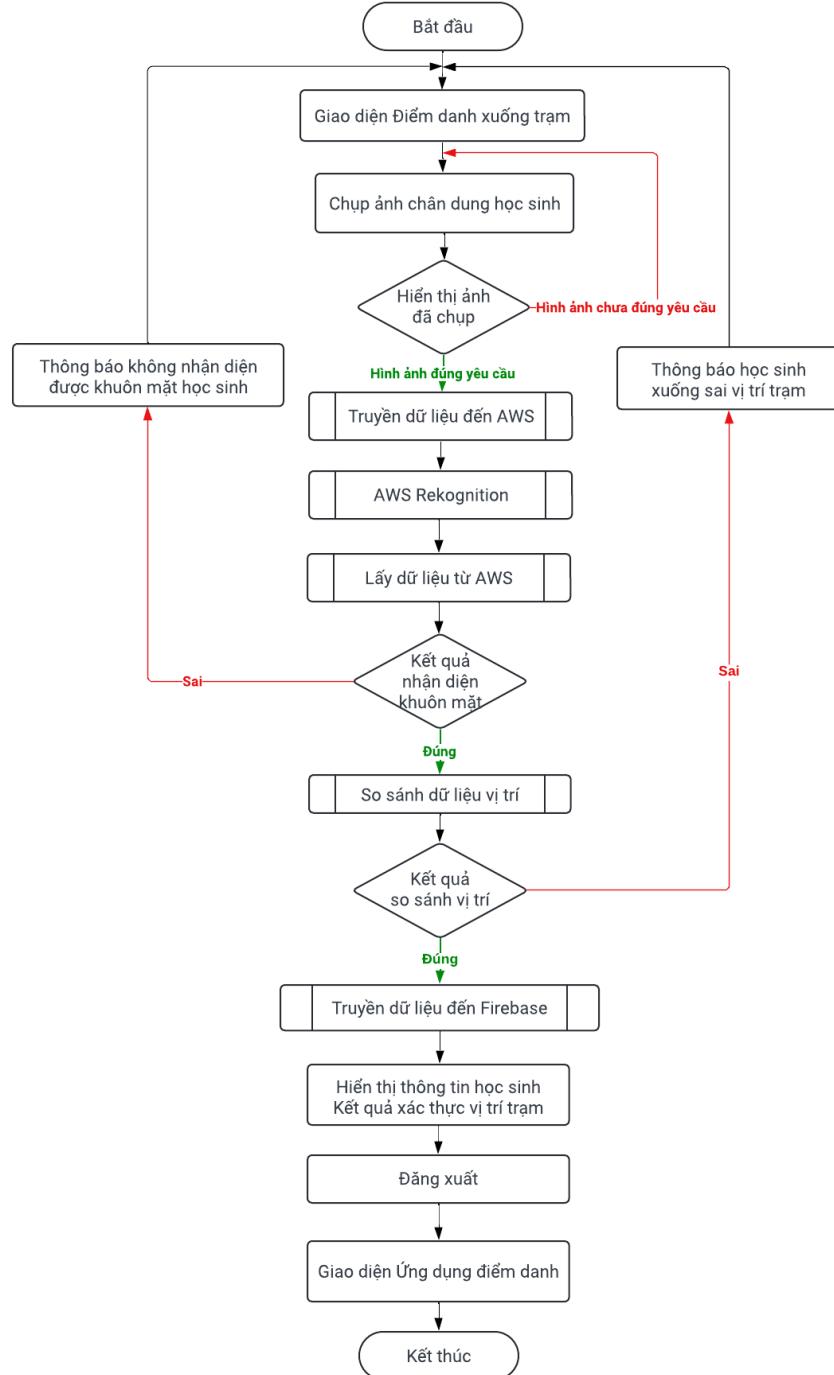
truyền đến AWS thông qua chương trình con Amazon Rekognition sẽ trả về kết quả của nhận diện khuôn mặt. Hệ thống sẽ lấy dữ liệu đó về giao diện phần mềm, nếu kết quả sai thì giao diện sẽ hiển thị thông báo và trả về giao diện ban đầu, nếu kết quả đúng giao diện sẽ hiển thị thông tin học sinh.

Sau khi tất cả học sinh đều lên xe, người quản lý có thể đăng xuất quay trở về giao diện Ứng dụng điểm danh ban đầu để truy cập vào phần Điểm danh xuống xe.



Hình 3.16: Lưu đồ giải thuật của chương trình con Điểm danh lên xe.

➤ Chương trình con Điểm danh xuống trạm



Hình 3.17: Lưu đồ giải thuật của chương trình con Điểm danh xuống trạm.

Từ khi chương trình con Điểm danh xuống trạm bắt đầu đến lúc trả về kết quả nhận diện khuôn mặt thì tất cả đều giống với Điểm danh lên xe, nhưng vì Điểm danh xuống trạm là điểm danh hai lớp nên khi kết quả nhận diện khuôn

mặt trả về là đúng thì sẽ tiến đến chương trình con So sánh dữ liệu vị trí.

Sau đó chương trình con này sẽ trả về kết quả so sánh vị trí, nếu kết quả là sai thì giao diện sẽ hiển thị thông báo và trả về giao diện ban đầu, nếu kết quả là đúng thì sẽ truyền dữ liệu đến Firebase đồng thời giao diện hiển thị thông tin học sinh cũng như kết quả đó.

Sau khi tất cả học sinh đều xuống xe, người quản lý có thể đăng xuất quay trở về giao diện Ứng dụng điểm danh ban đầu và tắt thiết bị.

➤ *Chương trình con So sánh dữ liệu vị trí*

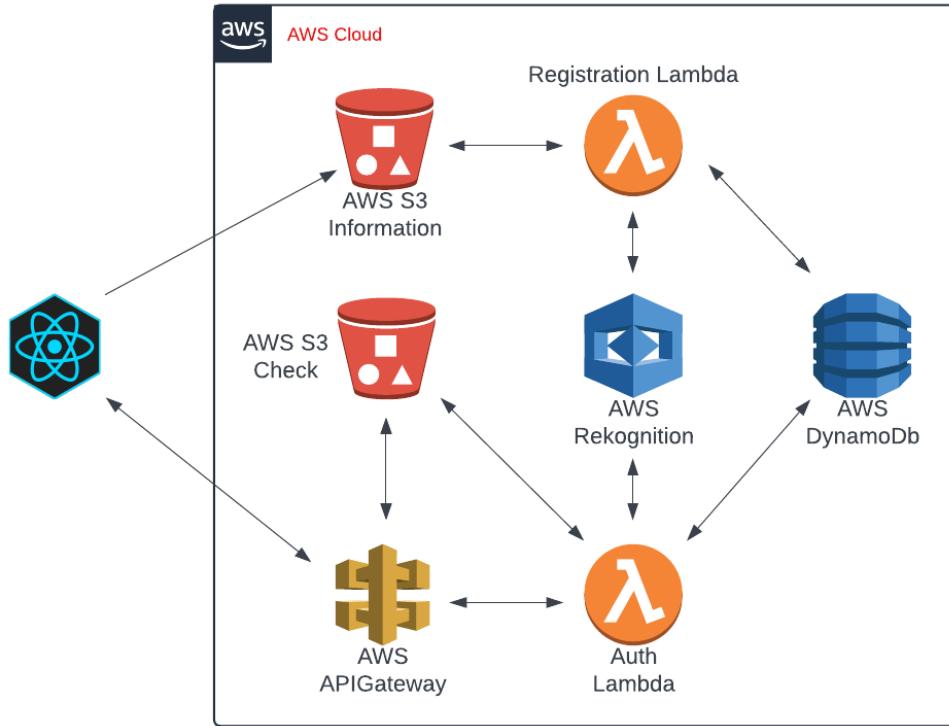
Khi bắt đầu chương trình, nó sẽ đồng thời lấy dữ liệu từ AWS (chứa thông tin vị trí của học sinh) và dữ liệu từ Firebase (chứa thông tin vị trí trực tiếp của chuyến xe). Sau đó, sẽ tiến hành so sánh vị trí trạm với vị trí của chuyến xe, với sai số cho phép là 5 mét.



Hình 3.18: Lưu đồ giải thuật của chương trình con So sánh dữ liệu vị trí.

Quá trình so sánh này sẽ diễn ra nhanh chóng để xác định xem chuyến xe có ở trong phạm vi 5 mét của trạm mà học sinh cần xuống hay không. Nếu vị trí nằm trong khoảng cách cho phép, chương trình sẽ trả về kết quả tương ứng. Sau khi hoàn thành việc so sánh và trả về kết quả, chương trình sẽ kết thúc.

3.5 THIẾT KẾ AMAZON WEB SERVICES



Hình 3.19: Sơ đồ nguyên lý của các dịch vụ AWS được sử dụng trong hệ thống.

Sơ đồ nguyên lý này bao gồm 2 luồng chính:

- *Luồng Lưu trữ dữ liệu hình ảnh (thông tin học sinh):*

Quy trình Lưu trữ dữ liệu hình ảnh (thông tin học sinh) bắt đầu bằng việc lưu trữ hình ảnh chân dung của học sinh trong S3 bucket. Hàm Registration Lambda sẽ sử dụng Amazon Rekognition để lập chỉ mục các ảnh học sinh.

Quá trình này sẽ tạo ra một khóa duy nhất, thông tin học sinh sẽ được lưu trữ trong DynamoDB với ID Rekognition tương ứng. Việc lưu trữ thông tin theo cách này sẽ giúp quản lý dữ liệu một cách hiệu quả và đảm bảo tính bảo mật.

- *Luồng Lưu, gửi và chỉnh sửa dữ liệu:*

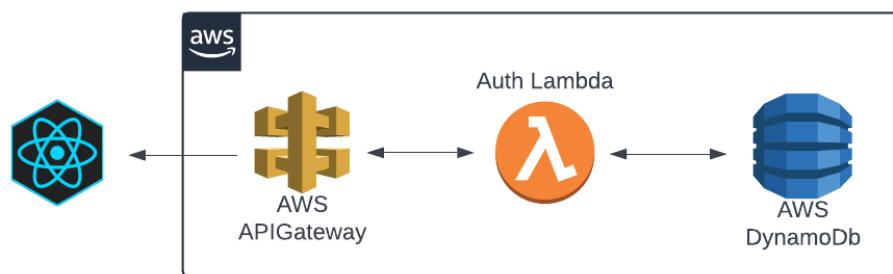
Quy trình Lưu, gửi và chỉnh sửa dữ liệu bắt đầu khi Ứng dụng điểm danh tải lên hình ảnh điểm danh của học sinh (Điểm danh lên xe và Điểm danh xuống trạm) thông qua giao diện phần mềm và gọi API Gateway.

Ảnh này sẽ được lưu trữ trong S3 bucket dành riêng cho hình ảnh điểm danh. Sau đó, hàm Auth Lambda sẽ sử dụng Amazon Rekognition để tạo ra một

khóa duy nhất từ ảnh đã tải lên. Tiếp theo, ID Rekognition này sẽ được kiểm tra với thông tin có sẵn trong bảng DynamoDB, nơi lưu trữ thông tin học sinh. Tiếp đó dữ liệu của học sinh đã được xác thực sẽ được truyền đến phần mềm để hiển thị trên giao diện.

Ngoài ra, Phần mềm quản lý còn có thể thêm, chỉnh sửa dữ liệu về thông tin học sinh và thông tin tài xế thông qua sự cho phép của API Gateway và cũng sử dụng hàm Auth Lambda để lưu trữ thông tin trong DynamoDB

3.5.1 Chương trình con Lấy dữ liệu từ AWS



Hình 3.20: Sơ đồ nguyên lý của chương trình con Lấy dữ liệu từ AWS

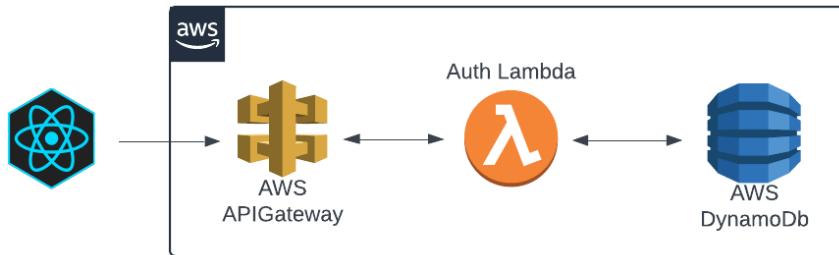
Đầu tiên, giao diện phần mềm sẽ gọi API Gateway để yêu cầu dữ liệu. API Gateway sau đó sẽ kích hoạt hàm Auth Lambda để truy vấn dữ liệu từ DynamoDB. Hàm Lambda sẽ thực hiện truy vấn dữ liệu từ DynamoDB và trả về kết quả dưới dạng JSON cho API Gateway.

Tiếp theo, API Gateway sẽ chuyển dữ liệu JSON trả về từ hàm Lambda đến giao diện phần mềm. Cuối cùng, giao diện phần mềm sẽ nhận dữ liệu từ API Gateway và hiển thị lên giao diện người dùng.

Để thực hiện được chương trình con này thì hệ thống phải thiết lập tại API Gateway một Stages để cấp quyền cho hệ thống được lấy và truyền dữ liệu đến DynamoDB thông qua dịch vụ Lambda. Các quyền có thể là GET, POST, PUT tương ứng với các dữ liệu khác nhau (thông tin học sinh, thông tin tài xế, hình ảnh chân dung học sinh, hình ảnh điểm danh) hệ thống muốn lưu trữ và xử lý.

3.5.2 Chương trình con Truyền dữ liệu đến AWS

➤ Trường hợp thông tin tài xế



Hình 3.21: Sơ đồ nguyên lý chương trình con Truyền dữ liệu đến AWS (Thông tin tài xế)

Đầu tiên, giao diện phần mềm sẽ gửi dữ liệu đến API Gateway thông qua một yêu cầu HTTP (POST). API Gateway sẽ kích hoạt hàm Auth Lambda để xử lý dữ liệu và lưu vào DynamoDB. Trong hàm Lambda, dữ liệu từ API Gateway sẽ được nhận và xử lý.

Sau đó, dữ liệu sẽ được lưu vào DynamoDB. Khi lưu dữ liệu thành công, hàm Lambda sẽ trả về một phản hồi thành công cho API Gateway. Tiếp theo, API Gateway sẽ chuyển phản hồi từ hàm Lambda trở lại cho giao diện.

Cuối cùng, giao diện phần mềm sẽ nhận được phản hồi thành công từ API Gateway, xác nhận rằng dữ liệu đã được lưu thành công vào DynamoDB.

➤ Trường hợp thông tin học sinh

Trước tiên, giao diện sẽ tải dữ liệu lên một S3 Bucket. Khi dữ liệu được tải lên thành công, một trigger từ S3 Bucket sẽ kích hoạt một hàm Lambda.



Hình 3.22: Sơ đồ nguyên lý chương trình con Truyền dữ liệu đến AWS (Thông tin học sinh)

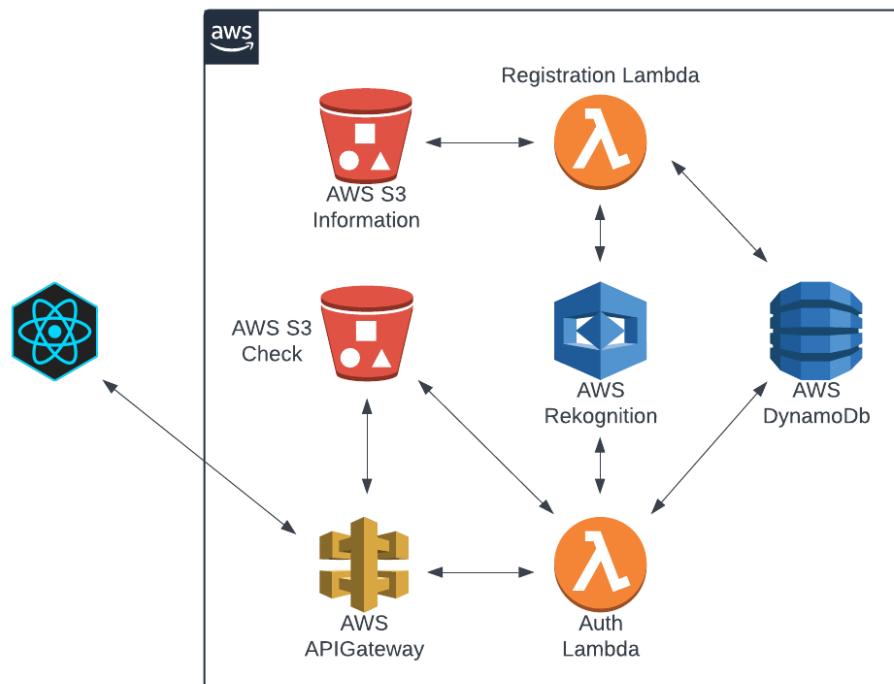
Hàm Registration Lambda này sẽ nhận thông báo về sự kiện tải lên từ S3 Bucket, sau đó đọc dữ liệu từ S3 và thực hiện các logic xử lý, chẳng hạn như

validate dữ liệu. Sau khi xử lý dữ liệu thành công, hàm Lambda sẽ lưu dữ liệu vào DynamoDB.

Cuối cùng, hàm Lambda sẽ trả về phản hồi thành công, thông báo rằng dữ liệu đã được lưu trữ thành công.

Ngoài ra, người quản lý có thể sử dụng Amazon CloudWatch Logs để giám sát, kiểm tra quá trình thực thi hàm Lambda. Để khi tải, chỉnh sửa dữ liệu không được thực hiện có thể kiểm tra, chỉnh sửa theo các lỗi đã được thông báo.

3.5.3 Chương trình con Amazon Rekognition



Hình 3.23: Sơ đồ nguyên lý của chương trình con Amazon Rekognition.

Quá trình nhận diện khuôn mặt sử dụng Amazon Rekognition bắt đầu bằng việc ứng dụng điểm danh sẽ tải lên hình ảnh chân dung điểm danh của học sinh lên Amazon S3 thông qua một yêu cầu HTTP (POST) của API Gateway.

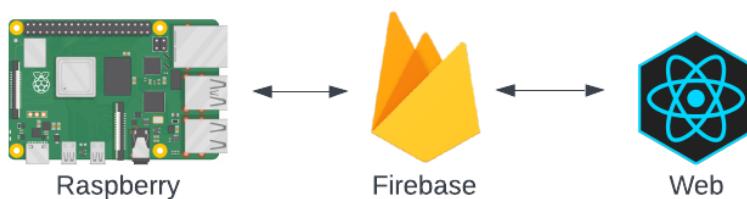
Sau đó, S3 sẽ kích hoạt hàm Auth Lambda để xử lý ảnh và thực hiện nhận diện khuôn mặt. Trong hàm Lambda, dữ liệu ảnh từ S3 sẽ được lấy ra và truyền vào API SearchFacesByImage của Amazon Rekognition để thực hiện nhận diện khuôn

mặt. Dịch vụ này sẽ trích xuất các đặc trưng của khuôn mặt để so sánh với các hình ảnh chân dung đã lưu trước đó trong S3 Information.

Kết quả nhận diện từ Amazon Rekognition sẽ được lưu vào DynamoDB để lưu trữ. Sau khi lưu kết quả vào DynamoDB thành công, hàm Lambda sẽ trả về một phản hồi thành công cho API Gateway. Và API Gateway sẽ chuyển phản hồi từ hàm Lambda trở lại cho giao diện phần mềm để hiển thị.

Bằng cách sử dụng Lambda, API Gateway và DynamoDB, quá trình nhận diện khuôn mặt trở nên linh hoạt, mở rộng và dễ quản lý. Giao diện phần mềm chỉ cần gửi ảnh lên, còn các tác vụ xử lý và lưu trữ được thực hiện trên nền tảng AWS.

3.6 THIẾT KẾ FIREBASE

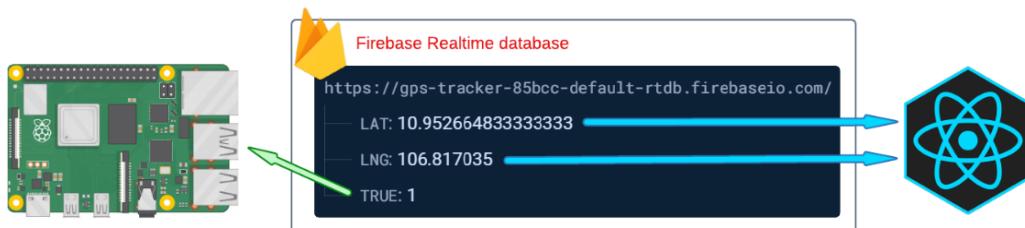


Hình 3.24: Sơ đồ nguyên lý của hệ thống Firebase

Firebase sẽ truyền và nhận dữ liệu liên tục theo thời gian thực từ phần mềm lẫn phần cứng, như một cầu nối trao đổi dữ liệu cho hai phần này.

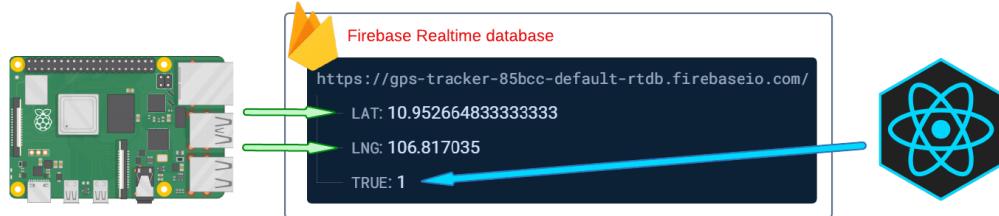
➤ Chương trình con Lấy dữ liệu từ Firebase

Phần cứng sẽ lấy kết quả trả về của chương trình con so sánh vị trí (điểm danh lớp hai của điểm danh xuống trạm) để điều khiển đóng mở cửa. Đồng thời, phần mềm sẽ lấy dữ liệu của cảm biến định vị vị trí để bắt đầu tiến hành chương trình con so sánh vị trí.



Hình 3.25: Sơ đồ nguyên lý của chương trình con Lấy dữ liệu từ Firebase

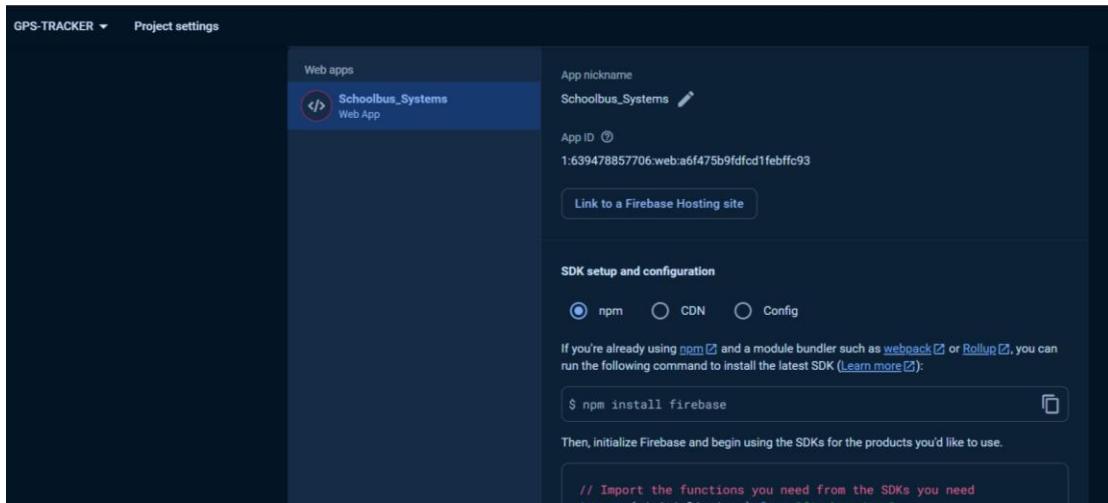
➤ Chương trình con Truyền dữ liệu đến Firebase



Hình 3.26: Sơ đồ nguyên lý của chương trình con Truyền dữ liệu đến Firebase

Phần cứng sẽ truyền dữ liệu của cảm biến định vị vị trí cùng lúc đó phần mềm cũng truyền đến dữ liệu trả về của chương trình con so sánh vị trí của ứng dụng điểm danh.

➤ Thiết lập Firebase Realtime Database Web API



Hình 3.27: Thiết lập Firebase Web API cho hệ thống

Sử dụng tính năng Web API của Firebase để phần mềm có thể liên kết với Firebase để truyền và lấy dữ liệu theo thời gian thực.

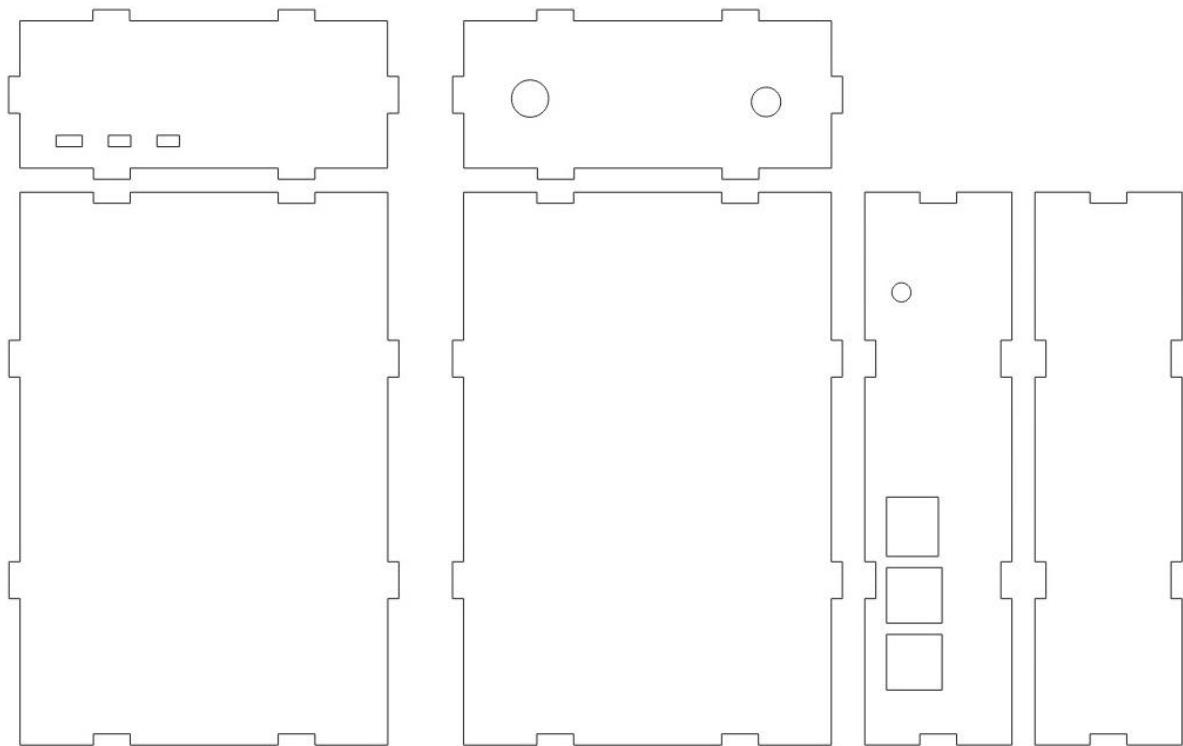
Hệ thống sử dụng Firebase để lưu trữ dữ liệu thời gian thực từ phần cứng và phần mềm vì để so sánh giữa Firebase và AWS trong lĩnh vực thành phố thì Firebase là tốt nhất. Do để lưu trữ dữ liệu thời gian thực trên AWS người dùng phải sử dụng bổ sung thêm các dịch vụ khác của AWS như IoT Core, TimeStreamDB, Imply,.. và kết hợp với giao thức MQTT khá là phức tạp trong khi đó Firebase chỉ cần API và thiết bị kết nối wifi khá là đơn giản.

CHƯƠNG 4 KẾT QUẢ, ĐÁNH GIÁ HỆ THỐNG

4.1 KẾT QUẢ PHẦN CỨNG

Sau khi lựa chọn các linh kiện phần cứng phù hợp với yêu cầu đã được đề ra của hệ thống, tiến hành các bước tiếp theo bao gồm bố trí hợp lý, đi dây lại các linh kiện để có thể ra được một sản phẩm gọn gàng, nhỏ gọn, tiện lợi.

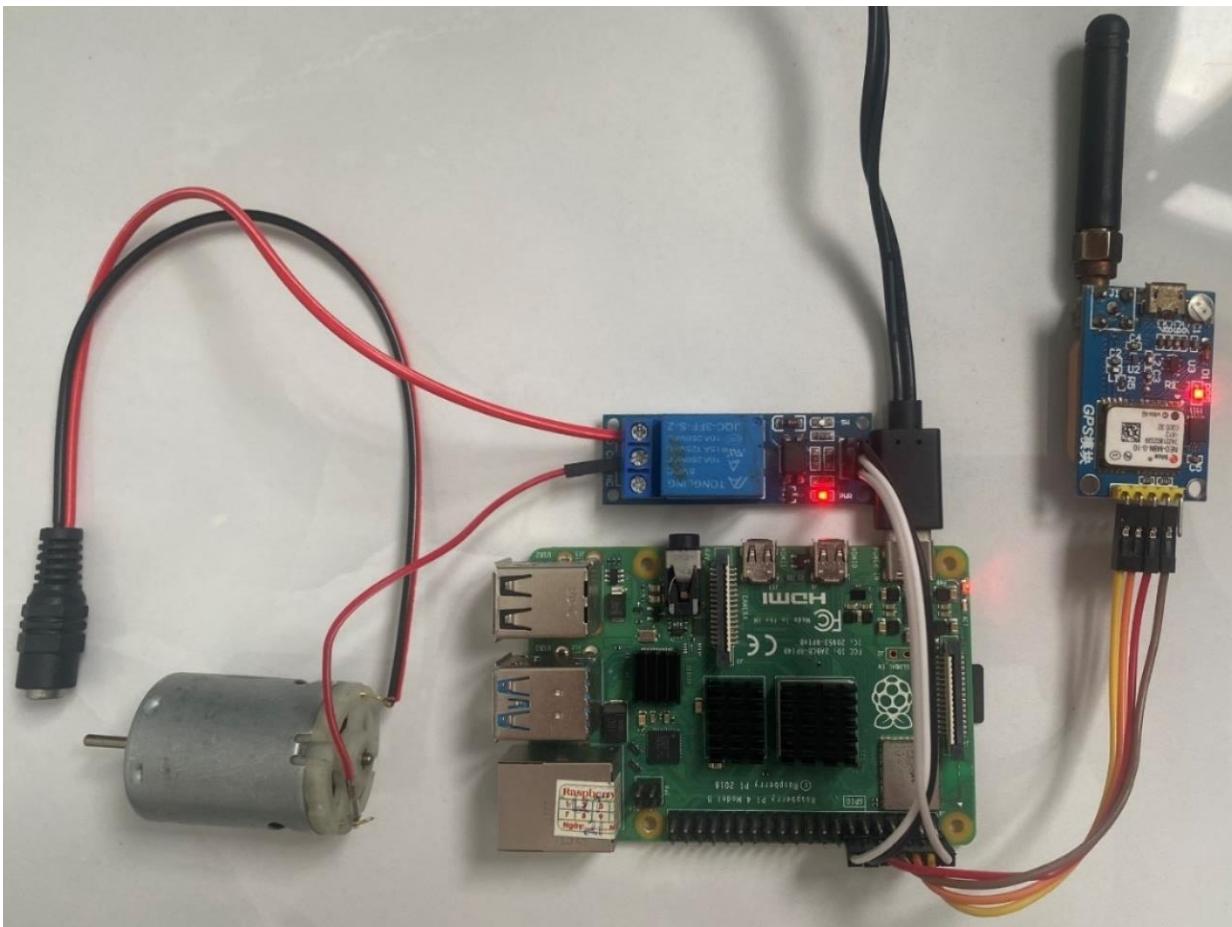
➤ *Thiết kế hộp đựng*



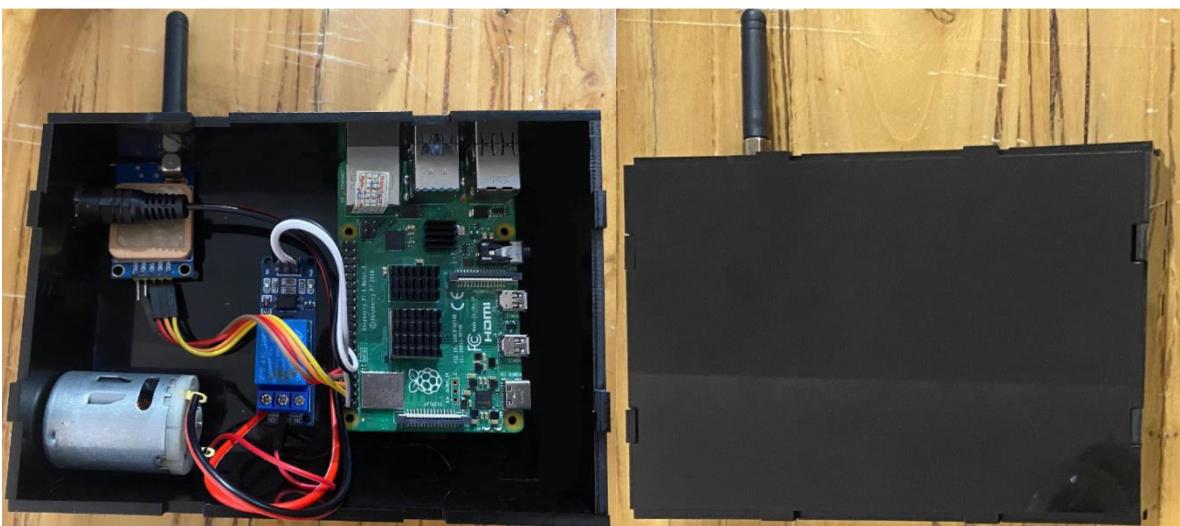
Hình 4.1: Bản thiết kế hộp đựng mô hình hệ thống

Phần hộp đựng mô hình hệ thống được thiết kế 2D sử dụng phần mềm CoreDraw x8. Sản phẩm được thiết kế để có thể đựng được các linh kiện phần cứng của hệ thống bao gồm Raspberry Pi 4B, module GPS NEO-M8, Ăng-ten bắt sóng của module GPS, module Relay 5V, Motor cũng như bổ sung các cổng để cấp nguồn cho Raspberry và Motor. Hộp đựng sử dụng mica nhằm tránh trường hợp thiết bị quá nhiệt làm hỏng thiết kế hộp.

➤ Mô hình phần cứng của hệ thống



Hình 4.2: Kết quả đi dây phần cứng hoàn chỉnh.

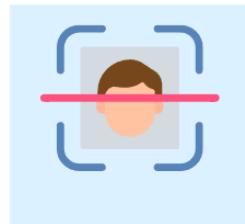


Hình 4.3: Kết quả mô hình phần cứng của hệ thống

4.2 KẾT QUẢ PHẦN MỀM



Hệ thống giám sát học sinh trên xe đưa đón



Ứng dụng điểm danh



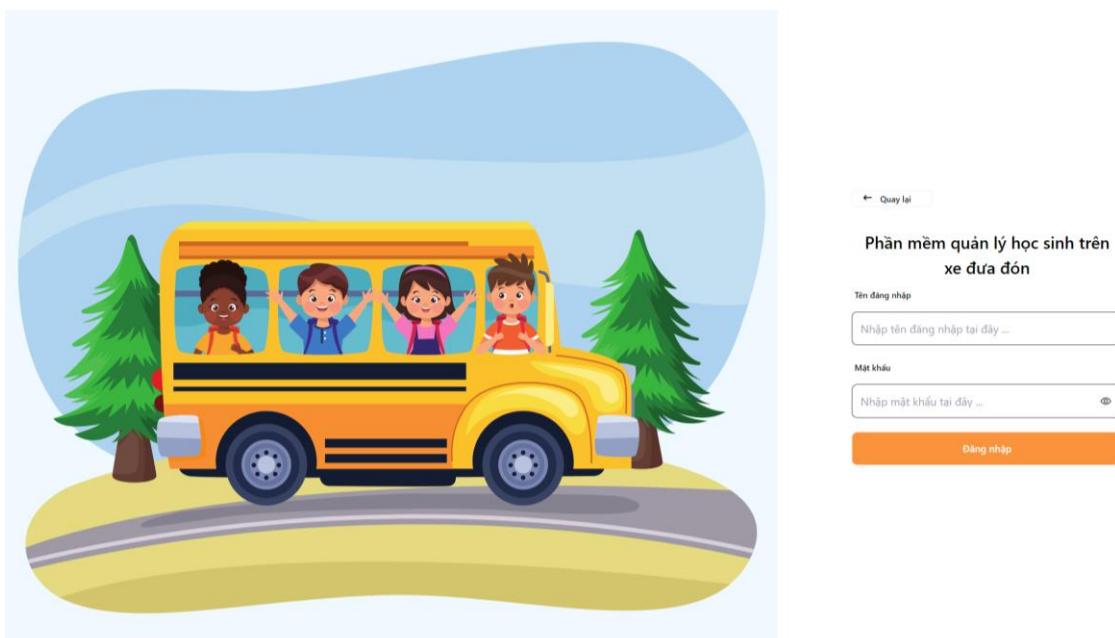
Phần mềm quản lý

Hình 4.4: Giao diện màn hình chính

Màn hình chính của phần mềm sẽ bao gồm hai phần tương đương với hai tính năng mà hệ thống cung cấp là Úng dụng điểm danh và Phần mềm quản lý.

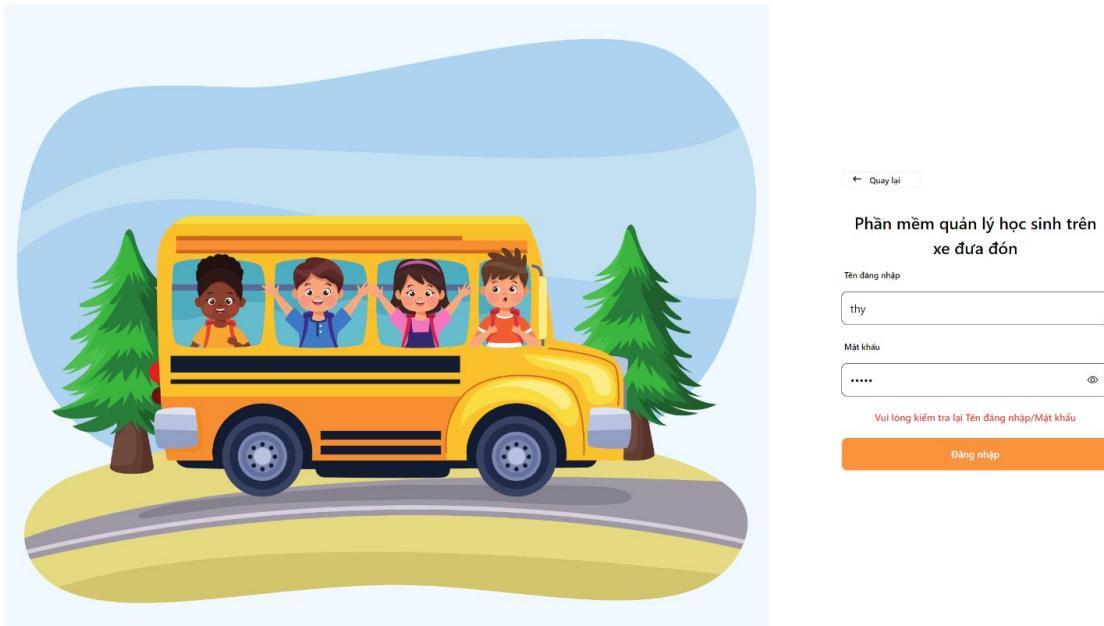
4.2.1 Giao diện Phần mềm Quản lý

- Giao diện Đăng nhập



Hình 4.5: Giao diện Đăng nhập

Việc hạn chế quyền truy cập chỉ dành cho người quản lý nhằm đảm bảo tính bảo mật và an toàn cho thông tin của học sinh và tài xế. Điều này giúp nâng cao chất lượng và hiệu quả hoạt động của hệ thống, tránh tình trạng thông tin bị rò rỉ hay thay đổi trái phép.



Hình 4.6: Giao diện Đăng nhập nếu đăng nhập sai

Nếu người dùng nhập sai tên đăng nhập hoặc mật khẩu, giao diện sẽ hiển thị thông báo lỗi như hình 4.6. Chỉ khi người dùng đúng cả tên đăng nhập và mật khẩu mới có thể truy cập vào trang quản lý.

Do tên đăng nhập và mật khẩu của người quản lý có giới hạn nên không cần phải tạo cơ sở dữ liệu để lưu trữ dữ liệu này. Chỉ cần mặc định sẵn trong phần mềm và chỉnh sửa thông qua mã code khi cần.

➤ Giao diện Tổng quan

Phần giao diện này sẽ hiển thị tên đề tài và tóm tắt giới thiệu về nội dung chính của đồ án và các hình ảnh minh họa kèm theo. Đây sẽ là phần giúp người xem nhanh chóng nắm bắt được mục tiêu và ý tưởng chính của sản phẩm.

Hình 4.7: Giao diện phần *Tổng quan*

➤ Giao diện *Giám sát chuyến xe*

Hình 4.8: Giao diện phần *Giám sát chuyến xe*

Giao diện phần giám sát chuyến xe được thiết kế khá toàn diện để đáp ứng các yêu cầu theo dõi và quản lý chuyến đi của học sinh.

Hiển thị vị trí chuyến xe trên bản đồ Google:

- Cho phép theo dõi chính xác vị trí hiện tại của chuyến xe.

Hiển thị thông tin và trạng thái của các học sinh:

- Liệt kê danh sách các học sinh trên chuyến xe, bao gồm các thông

tin cơ bản như tên, vị trí (latitude, longitude), trạng thái.

- Trạng thái cụ thể như "Có mặt trên xe", "Chưa lên xe", "Đã xuống trạm" sẽ giúp nắm bắt tình hình điểm danh của học sinh.
- Thời gian điểm danh lên xe và xuống trạm để theo dõi quá trình.

Với hai phần hiển thị này, người quản lý sẽ có được tổng quan đầy đủ về tình hình chuyến xe cũng như sự tham gia của từng học sinh. Điều này giúp việc giám sát trở nên hiệu quả và chính xác hơn.

Ngoài ra tất cả thông tin dữ liệu được hiển thị ở phần này sẽ được cập nhập liên tục theo thời gian thông qua việc lấy dữ liệu từ các cơ sở dữ liệu hệ thống sử dụng. Quá trình cập nhật này nhằm đảm bảo rằng người quản lý luôn có được những thông tin mới nhất, chính xác nhất.

Hình 4.9: Tính năng lọc thông tin của phần Giám sát chuyến xe theo chuyến xe

Hình 4.10: Tính năng lọc thông tin của phần Giám sát chuyến xe theo trạng thái

Ngoài ra, phần giao diện này còn hỗ trợ lọc thông tin học sinh theo chuyên xe và trạng thái giúp người quản lý dễ dàng giám sát hơn.

➤ Giao diện Quản lý học sinh

STT	Tên	Chuyên xe	Vị trí	XOÁ	
1	Nguyen Dang Mai Thy	1	10.952648_106.816997		
2	Bui Thi Thanh Ngan	1	10.855654_106.631087		
3	Nguyen Quoc Viet	1	10.850570_106.772055		
4	Huynh Thien Nhan	2	10.840788_106.808897		
5	Tran Ninh Hoang	2	10.770358_106.679110		

Hình 4.11: Giao diện phần Quản lý học sinh

Giao diện Quản lý học sinh bao gồm các tính năng như hiển thị thông tin học sinh, thêm thông tin học sinh, chỉnh sửa thông tin học sinh.

Các thông tin học sinh bao gồm tên, chuyên xe, vị trí. Những thông tin này được lấy trực tiếp khi truy xuất dữ liệu từ các bảng trong AWS DynamoDB.

Do hạn chế của AWS trong việc hỗ trợ lưu trữ dữ liệu có dấu tiếng Việt, các thông tin này sẽ được hiển thị không dấu. Đây là một hạn chế tạm thời của hệ thống, nhưng sẽ được cập nhật khi AWS cải thiện khả năng lưu trữ và xử lý dữ liệu tiếng Việt có dấu trong tương lai.

Ở tính năng thêm thông tin học sinh, sửa thông tin học sinh (hình 4.13) dữ liệu thêm và chỉnh sửa sau khi xác nhận sẽ được truyền đến AWS để insert/update items trong table của DynamoDB.

Ngoài ra ở phần giao diện này, không chỉ thêm và chỉnh sửa các thông tin cá nhân mà còn có thể cập nhập hình ảnh chân dung của học sinh. Với hình ảnh

được lấy trực tiếp từ máy tính cá nhân hay dữ liệu di động. Giúp người quản lý có thể linh động, thao tác dễ dàng trong việc quản lý thông tin học sinh.

Hình 4.12: Tính năng lọc thông tin của phần Quản lý học sinh theo chuyến xe

Tính năng lọc thông tin theo chuyến xe này tương tự với tính năng lọc thông tin của phần Giám sát chuyến xe

Hình 4.13: Tính năng thêm thông tin, sửa thông tin học sinh của phần Quản lý học sinh

➤ Giao diện Quản lý tài xe

Hình 4.14: Giao diện phần Quản lý tài xe

Tương tự với giao diện Quản lý học sinh, phần giao diện Quản lý tài xế cũng bao gồm ba tính năng chính là hiển thị thông tin tài xế, thêm thông tin tài xế và chỉnh sửa thông tin tài xế.

Các thông tin tài xế được hiển thị bao gồm:

- Tên tài xế
- Số điện thoại
- Chuyến xe phụ trách
- Trạng thái (hoạt động hoặc không hoạt động)

Hoạt động nếu chuyến xe đang chạy, không hoạt động nếu chuyến xe chưa bắt đầu chạy hoặc đã kết thúc

Việc tích hợp thông tin về chuyến xe và trạng thái hoạt động của tài xế trong cùng một giao diện quản lý giúp người quản lý dễ dàng giám sát và theo dõi tình trạng hoạt động của từng tài xế cũng như chuyến xe mà họ phụ trách.

Tăng tính đồng bộ và liên kết giữa quản lý tài xế và quản lý chuyến xe, vì trạng thái hoạt động của tài xế trực tiếp liên quan đến chuyến xe đang thực hiện. Giao diện trở nên đơn giản, dễ sử dụng hơn, không bị rườm rà với các tính năng thừa thãi.

The screenshot shows two side-by-side forms for managing driver information. Both forms have a header with a back arrow and a blue 'XÁC NHẬN THÊM' (Confirm Add) or 'CẤP NHẬT' (Update) button in the top right corner.

Thêm thông tin tài xế (Left Form):

Thông tin tài xế	
Tên	Võ Văn Tài
Số điện thoại	0123654789
Chuyến xe chạy	4
Trạng thái	Không hoạt động

Sửa thông tin tài xế (Right Form):

Thông tin tài xế	
Tên	Phùng Đức Hải
Số điện thoại	0123456789
Chuyến xe chạy	1
Trạng thái	Hoạt động

Hình 4.15: Tính năng thêm thông tin, sửa thông tin tài xế của phần Quản lý tài xế

Giống với các tính năng thêm, sửa thông tin học sinh thì thêm thông tin tài xế và sửa thông tin tài xế có cơ chế hoạt động giống nhau. Nhưng ở phần quản lý thông tin tài xế thì chỉ có các thông tin cá nhân không bao gồm hình ảnh chân dung.

4.2.2 Giao diện Ứng dụng Điểm danh



Ứng dụng điểm danh học sinh bằng khuôn mặt



Điểm danh lên xe



Điểm danh xuống trạm

Hình 4.16: Giao diện màn hình chính của Ứng dụng điểm danh

Giao diện Ứng dụng điểm danh bao gồm hai tính năng là điểm danh lên xe (sử dụng để quản lý học sinh lên đúng xe với điểm danh một lớp là nhận diện khuôn mặt) và điểm danh xuống trạm (được sử dụng để quản lý học sinh xuống đúng trạm với điểm danh hai lớp là nhận diện khuôn mặt và so sánh vị trí).

➤ Giao diện Điểm danh lên xe

Màn hình chính của phần Điểm danh lên xe gồm hai phần chính.

Phần đầu tiên hiển thị hình ảnh trực tiếp từ camera của thiết bị điện tử. Mục đích là để học sinh có thể chỉnh sửa góc nhìn của mình, đảm bảo không bị che khuất khi thực hiện quá trình điểm danh. Việc này sẽ giúp hệ thống có thể nhận diện và xác thực danh tính của học sinh một cách chính xác.

Phần còn lại của màn hình hiển thị nội dung hướng dẫn cụ thể cách thực hiện thao tác chụp ảnh khuôn mặt. Nội dung này sẽ cung cấp thông tin chi tiết về các bước cần thực hiện, cũng như giới thiệu các tính năng của các nút nhấn có

trong giao diện. Điều này sẽ giúp học sinh có thể thao tác đúng cách, đồng thời cũng giúp họ nắm rõ các tính năng có sẵn để có thể tương tác với hệ thống một cách hiệu quả.



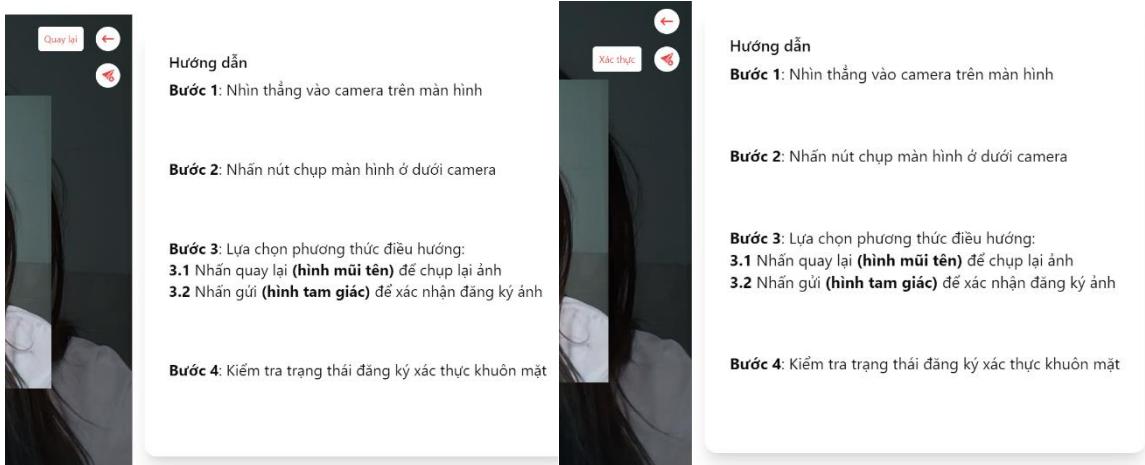
Hình 4.17: Giao diện chính của phần Điểm danh lên xe

Thiết kế màn hình chính như vậy nhằm mục đích hướng dẫn và hỗ trợ tối đa cho học sinh trong quá trình điểm danh lên xe, từ đó nâng cao hiệu quả và tính chính xác của toàn bộ quy trình.



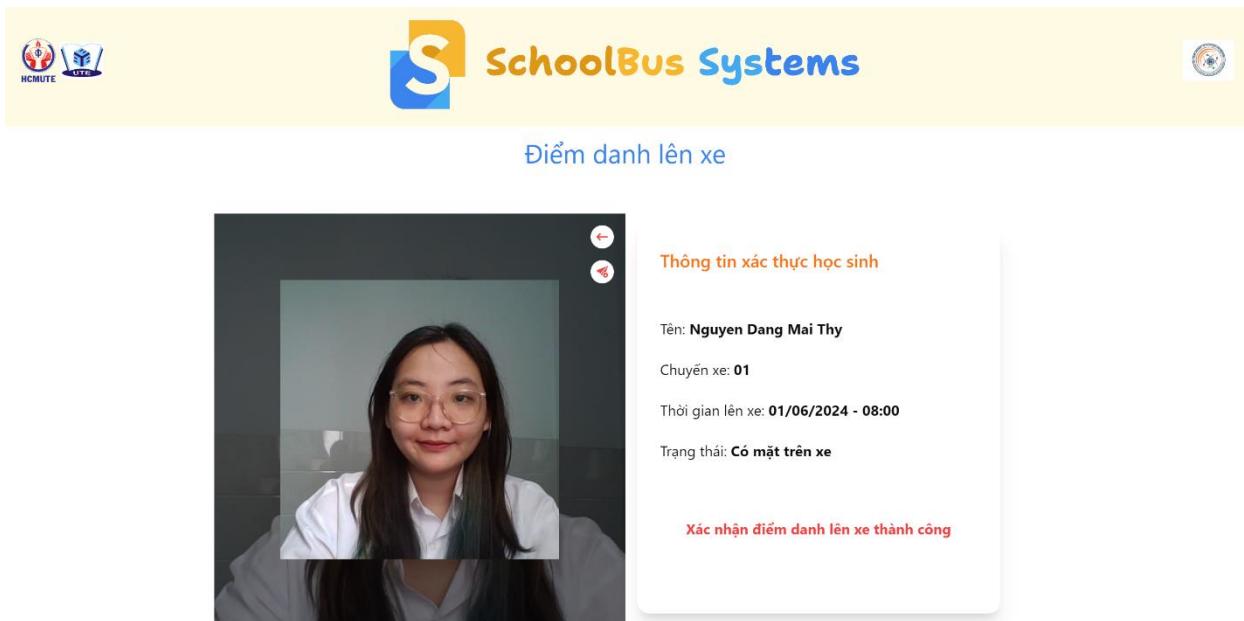
Hình 4.18: Giao diện sau khi tiến hành chụp ảnh chân dung.

Sau khi học sinh canh chỉnh góc mặt, nhìn thẳng vào camera như hướng dẫn và bấm nút chụp thì hình ảnh vừa chụp sẽ được hiển thị lại trên giao diện để học sinh xem đã hợp yêu cầu chưa.



Hình 4.19: Các lựa chọn phương thức điều hướng

Theo hướng dẫn thì học sinh sẽ có hai lựa chọn phương thức điều hướng là nhấn nút quay lại (nếu hình ảnh chưa đặt yêu cầu) để chụp lại ảnh sau đó giao diện sẽ trả về màn hình chính của Điểm danh lên xe. Hoặc nhấn nút gửi (nếu hình ảnh đã đạt yêu cầu) để xác nhận gửi ảnh và sau đó ảnh sẽ được gửi đến AWS bắt đầu nhận diện khuôn mặt.



Hình 4.20: Giao diện trả về sau khi điểm danh lên xe thành công.

Sau khi xác nhận gửi nếu kết quả nhận diện từ AWS trả về là đúng thì giao diện sẽ hiển thị thông tin của học sinh và thông báo thành công.



Hình 4.21: Giao diện trả về sau khi điểm danh lên xe KHÔNG thành công.

Nếu kết quả nhận diện từ AWS trả về là sai thì giao diện sẽ hiển thị tất cả thông tin là None và thông báo “KHÔNG” thành công.

➤ *Giao diện Điểm danh xuống trạm*

Giao diện của Điểm danh xuống trạm tương tự giao diện Điểm danh lên xe.



Hình 4.22: Giao diện màn hình chính của phần Điểm danh xuống trạm



Hình 4.23: Giao diện trả về sau khi điểm danh xuống trạm thành công.



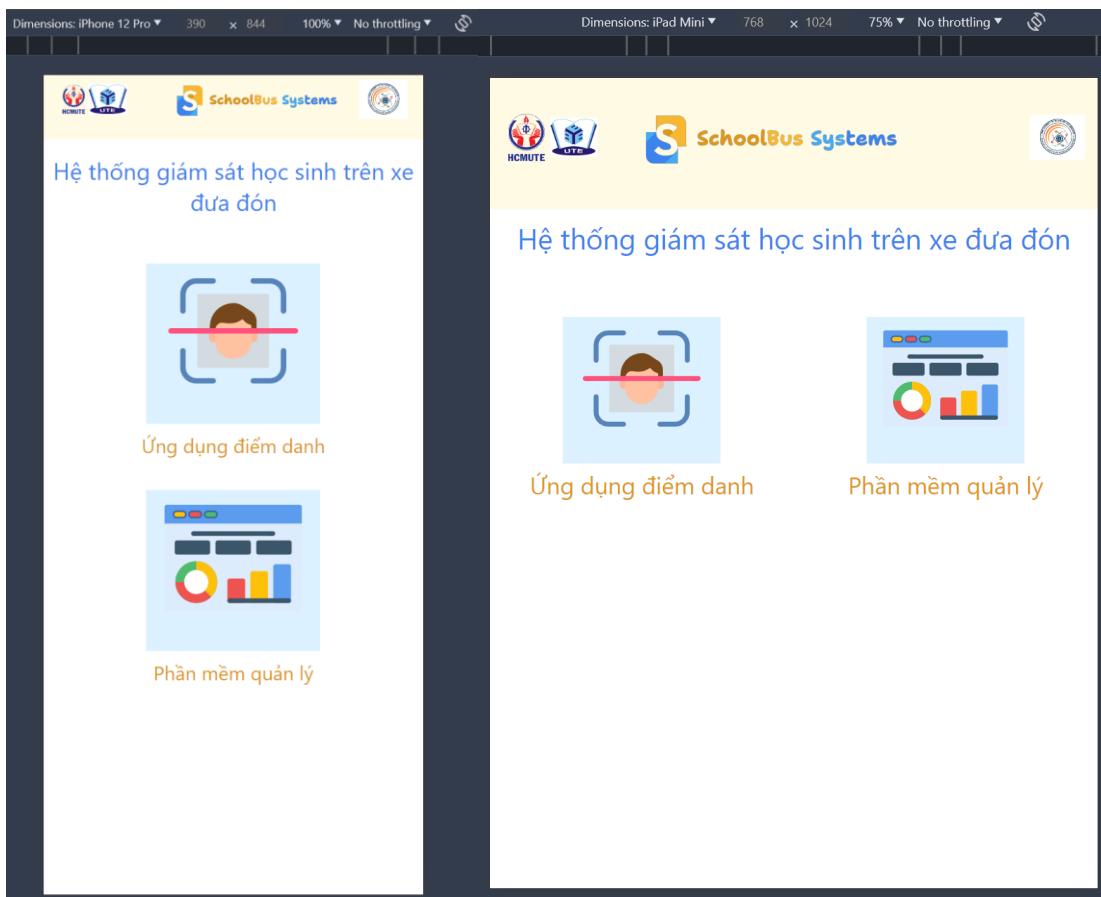
Hình 4.24: Giao diện trả về sau khi điểm danh xuống trạm KHÔNG thành công.

Nếu kết quả so sánh vị trí cũng chính xác, nghĩa là học sinh đang ở vị trí đúng, ứng dụng sẽ hiển thị toàn bộ thông tin của học sinh và thông báo "THÀNH CÔNG" cho quá trình điểm danh.

Tuy nhiên, nếu kết quả nhận diện từ AWS trả về là không chính xác, hoặc kết quả so sánh vị trí không đúng, ứng dụng sẽ hiển thị tất cả thông tin liên quan là "None" và thông báo "KHÔNG THÀNH CÔNG" cho quá trình điểm danh.

Quy trình kiểm tra kép này, bao gồm nhận diện khuôn mặt và xác minh vị trí, giúp đảm bảo tính chính xác và an toàn của toàn bộ hệ thống điểm danh. Nó giúp loại trừ các trường hợp nhầm lẫn hoặc giả mạo danh tính, từ đó nâng cao độ tin cậy và hiệu quả của ứng dụng.

4.2.3 Giao diện hiển thị trên các thiết bị khác



Hình 4.25: Hình ảnh minh họa khi hiển thị giao diện trên các thiết bị khác nhau

Hệ thống sẽ được thiết kế với giao diện người dùng (UI) responsive, có khả năng tự động điều chỉnh để phù hợp với nhiều loại thiết bị kết nối internet khác nhau, như smartphone, tablet,... thay vì chỉ có thể sử dụng trên laptop

Việc này sẽ mang lại nhiều lợi ích như tăng trải nghiệm người dùng (UX) thông qua việc truy cập dễ dàng trên nhiều thiết bị, mở rộng phạm vi sử dụng hệ thống, không bị giới hạn bởi loại thiết bị.

4.3 KẾT QUẢ DỮ LIỆU LUU TRỮ

Hệ thống sử dụng hai nền tảng lưu trữ dữ liệu khác nhau là AWS và Firebase. AWS cho thông tin cá nhân và hình ảnh chân dung còn Firebase dành cho các dữ liệu trực tiếp thay đổi liên tục từ phần cứng và phần mềm, là cầu nối giữa hai phần của hệ thống.

4.3.1 Lưu trữ dữ liệu trên AWS

➤ Kết quả thiết lập Amazon API Gateway

The screenshot shows the AWS API Gateway interface. In the top navigation bar, the path is: API Gateway > APIs > SchoolBus-Systems-API (1yg1s6dez6) > Stages. The main title is "Stages". On the left, there's a tree view of the API resources under the "dev" stage: / (with /drivers, GET, OPTIONS, POST, /student, GET, OPTIONS, POST, and /{bucket}/{filename}), and /{bucket}. On the right, the "Stage details" section for the "dev" stage is displayed. It includes fields for Stage name (dev), Rate (Info -), Cache cluster (Info - Inactive), Default method-level caching (Info - Inactive), Invoke URL (https://1yg1s6dez6.execute-api.us-east-1.amazonaws.com/dev), and Active deployment (qlaeh7 on May 31, 2024, 23:42 (UTC+07:00)). Below this is the "Logs and tracing" section, which lists CloudWatch logs (Info - Inactive), Detailed metrics (Info - Inactive), and X-Ray tracing (Info - Inactive). There are also sections for Custom access logging and another set of logs and tracing options.

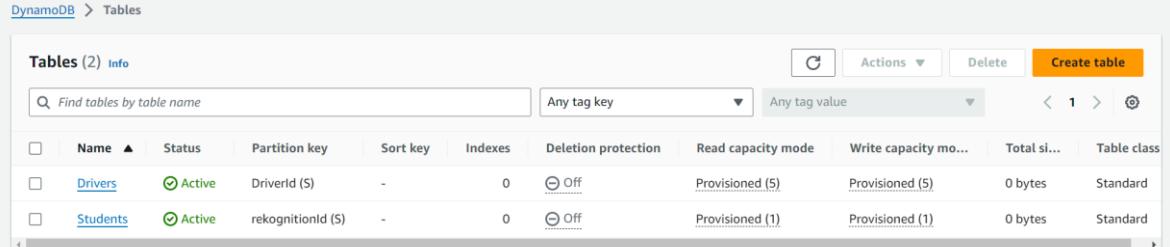
Hình 4.26: Kết quả của Amazon API Gateway sau khi thiết lập hệ thống

API Gateway đóng vai trò như một trung gian, quản lý và điều phối các yêu cầu như PUT, GET, POST dữ liệu từ phần mềm thông qua API. Cụ thể ở đây hệ thống bao gồm ba nguồn là drivers, student và bucket. Nguồn drivers được cấp phép có thể lấy, thêm và chỉnh sửa dữ liệu từ bảng Drivers trong DynamoDB. Tương tự như vậy nguồn student cũng được cấp phép để lấy, thêm và chỉnh sửa dữ liệu từ bảng Students. Trong khi đó nguồn bucket lại chỉ có chức

năng tải dữ liệu lên S3 bucket, dữ liệu được tải lên chia ra làm hai phần, phần hình ảnh (chân dung học sinh) và phần tên ảnh (ID học sinh).

Mỗi nguồn sẽ được kết nối với một Lambda function khác nhau để có thể xử lý dữ liệu đầu vào và hiển thị, lưu trữ trên DynamoDB và S3. Và khi gọi API chúng ta chỉ cần gọi một Invoke URL cho cả ba nguồn thay vì phải gọi ba cái khác nhau.

➤ Kết quả thiết lập Amazon DynamoDB

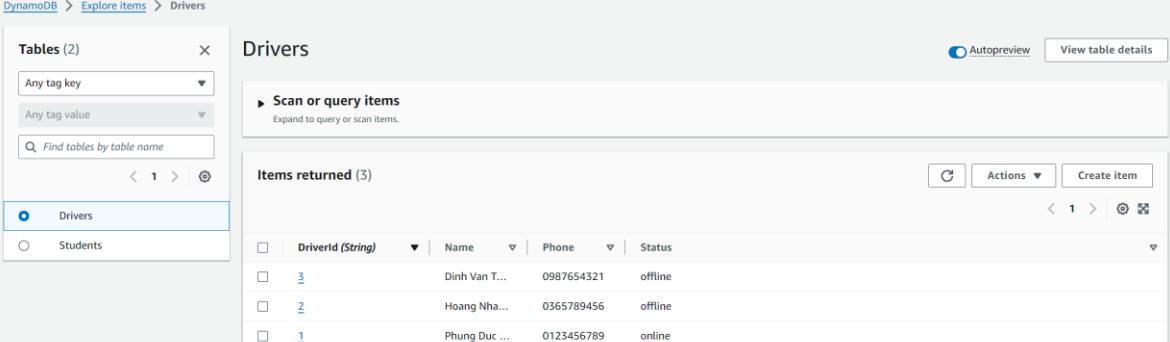


The screenshot shows the AWS DynamoDB 'Tables' page. At the top, there are filters for 'Find tables by table name', 'Any tag key', 'Any tag value', and pagination controls. Below the header, there is a table with the following data:

	Name	Status	Partition key	Sort key	Indexes	Deletion protection	Read capacity mode	Write capacity mode	Total size	Table class
<input type="checkbox"/>	Drivers	Active	DriverId (\$)	-	0	Off	Provisioned (5)	Provisioned (5)	0 bytes	Standard
<input type="checkbox"/>	Students	Active	rekognitionId (\$)	-	0	Off	Provisioned (1)	Provisioned (1)	0 bytes	Standard

Hình 4.27: Kết quả của Amazon DynamoDB sau khi thiết lập hệ thống

Hệ thống sẽ lưu trữ dữ liệu vào hai bảng (table) trong Amazon DynamoDB bao gồm bảng Drivers (dành cho phần quản lý tài xế) và bảng Students (dành cho phần quản lý học sinh).



The screenshot shows the AWS DynamoDB 'Explore items' page for the 'Drivers' table. On the left, there is a sidebar with a 'Tables (2)' section and a 'Scan or query items' button. The main area displays the 'Items returned (3)' table with the following data:

	DriverId (String)	Name	Phone	Status
<input type="checkbox"/>	3	Dinh Van T...	0987654321	offline
<input type="checkbox"/>	2	Hoang Nha...	0365789456	offline
<input type="checkbox"/>	1	Phung Duc ...	0123456789	online

Hình 4.28: Kết quả lưu thông tin tài xế trong bảng Drivers

Bảng Drivers sẽ chứa các thông tin của tài xế trong phần quản lý tài xế đã đề cập trước đó bao gồm DriverId (chuyến xe tài xế phụ trách), Name (tên của tài xế), Phone (số điện thoại tài xế), Status (trạng thái hoạt động của tài xế).

Do một trong những mục tiêu chính của hệ thống là giảm thiểu nhân công nên hệ thống mặc định mỗi chuyến xe sẽ chỉ có một tài xế phụ trách nên DriverId (chuyến xe tài xế phụ trách) nên được sử dụng làm Partition key.

Ngoài ra bảng dữ liệu này sẽ được cung cấp trực tiếp từ phần mềm ứng dụng quản lý thông qua Amazon API Gateway và Lambda do dữ liệu không chứa hình ảnh.

	rekognitionId (String)	Locate	Name	Status	Time
<input type="checkbox"/>	202	10.770358...	Tran Ninh H...	true	16:25
<input type="checkbox"/>	201	10.840788...	Huynh Thie...	up-wrong	none
<input type="checkbox"/>	103	10.850570...	Nguyen Qu...	down-true	17:05
<input type="checkbox"/>	102	10.855654...	Bui Thi Tha...	up-wrong	none
<input type="checkbox"/>	101	10.952648...	Nguyen Da...	true	14:30

Hình 4.29: Kết quả lưu thông tin học sinh trong bảng Students

Tương tự bảng Drivers bảng Student bao gồm các thông tin của học sinh đã được đề cập đến trong phần quản lý học sinh bao gồm rekognitionId (ID học sinh), Locate (vị trí trạm), Name (tên học sinh), Status (trạng thái học sinh) và Time (thời gian mà học sinh điền danh lê xe, điền danh xuống trạm).

Ở đây Status được mã hóa ngắn hơn để tiện xử lý khi cập nhập dữ liệu trong đó “true” = “Có mặt trên xe”, “up-wrong” = “Chưa lên xe”, “down-true” = “Đã xuống trạm”

Trong đó, rekognitionId (Partition key của bảng) chứa thông tin về ID của học sinh được mã hóa “Chuyến xe” “Số thứ tự”. Ví dụ rekognitionId là 102 có nghĩa là chuyến xe của học sinh này là chuyến số 1 và học sinh có số thứ tự là 02.

➤ Kết quả thiết lập Amazon Lambda Functions

Function name	Description	Package type	Runtime	Last modified
schoolbus-systems-registration	-	Zip	Python 3.8	17 minutes ago
schoolbus-systems-app	-	Zip	Python 3.8	8 minutes ago
schoolbus-systems-drivers	-	Zip	Python 3.8	44 seconds ago

Hình 4.30: Kết quả của Amazon Lambda Functions sau khi thiết lập hệ thống

Lambda Functions được dùng để chạy mã code mà không cần phải quản lý hay duy trì máy chủ giống EC2 Instance. Vì tính tiện lợi của nó ở đây hệ thống sử dụng ba lambda functions bao gồm schoolbus-systems-registration, schoolbus-systems-app, schoolbus-systems-drivers cho các tính năng khác nhau.

Function schoolbus-systems-registration được dùng để truyền, xử lý dữ liệu về thông tin học sinh (thêm, chỉnh sửa) từ phần mềm đến bảng Students trong DynamoDB. Bao gồm cả hình ảnh chân dung và dữ liệu cá nhân.

Function schoolbus-systems-app được dùng để truyền hình ảnh chân dung của học sinh khi nhận diện khuôn mặt đến DynamoDB và trả kết quả nhận diện về cho Ứng dụng điểm danh.

Function schoolbus-systems-drivers được dùng để truyền, xử lý dữ liệu về thông tin tài xế (thêm, chỉnh sửa) từ phần mềm đến bảng Drivers.

➤ Kết quả thiết lập Amazon S3 Buckets

Amazon S3 Buckets được dùng để lưu trữ các dạng dữ liệu như hình ảnh, tài liệu,... Vì vậy hệ thống sử dụng hai S3 buckets để lưu trữ hình ảnh bao gồm schoolbus-systems-images chứa hình ảnh chân dung học sinh và schoolbus-systems-check chứa hình ảnh điểm danh của học sinh.

Cả hai buckets đều được cấp quyền để có thể tải ảnh và xóa ảnh bằng các dòng lệnh, hỗ trợ cho việc phát triển phần mềm của hệ thống.

General purpose buckets (2)

Name	AWS Region	IAM Access Analyzer	Creation date
schoolbus-systems-check	US East (N. Virginia) us-east-1	View analyzer for us-east-1	May 18, 2024, 14:01:00 (UTC+07:00)
schoolbus-systems-images	US East (N. Virginia) us-east-1	View analyzer for us-east-1	May 18, 2024, 14:00:13 (UTC+07:00)

Hình 4.31: Kết quả của Amazon S3 sau khi thiết lập hệ thống

Objects (5)

Name	Type	Last modified	Size	Storage class
101.jpeg	jpeg	June 1, 2024, 00:28:08 (UTC+07:00)	1.1 MB	Standard
102.jpg	jpg	June 1, 2024, 00:28:06 (UTC+07:00)	15.5 KB	Standard
103.jpg	jpg	June 1, 2024, 00:28:04 (UTC+07:00)	766.2 KB	Standard
201.jpg	jpg	June 1, 2024, 00:28:02 (UTC+07:00)	647.3 KB	Standard
202.jpg	jpg	June 1, 2024, 00:27:58 (UTC+07:00)	814.7 KB	Standard

Hình 4.32: Kết quả lưu trữ hình ảnh chân dung thông tin học sinh

Hình ảnh chân dung thông tin học sinh sẽ được lưu với filename là rekognitionId.

Objects (6)

Name	Type	Last modified	Size	Storage class
239f3043-2da1-461d-901b-5e06073756da.jpeg	jpeg	June 1, 2024, 00:36:38 (UTC+07:00)	40.4 KB	Standard
24e5df66-7f4c-4c49-9fd7-7c99a28f10cb.jpeg	jpeg	June 1, 2024, 00:36:29 (UTC+07:00)	40.4 KB	Standard
2b5fbe5f-deef-4139-b2d9-87fc66dc2b40.jpeg	jpeg	June 1, 2024, 00:36:40 (UTC+07:00)	40.4 KB	Standard
9acc45a-96fd-4861-a451-effc16576e90.jpeg	jpeg	June 1, 2024, 00:36:32 (UTC+07:00)	40.4 KB	Standard
a2247701-8ead-4238-b30e-ccf08050fc9b.jpeg	jpeg	June 1, 2024, 00:36:34 (UTC+07:00)	40.4 KB	Standard
fe29b30a-4e46-4bb0-ac5c-5b5e3135a944.jpeg	jpeg	June 1, 2024, 00:36:37 (UTC+07:00)	40.4 KB	Standard

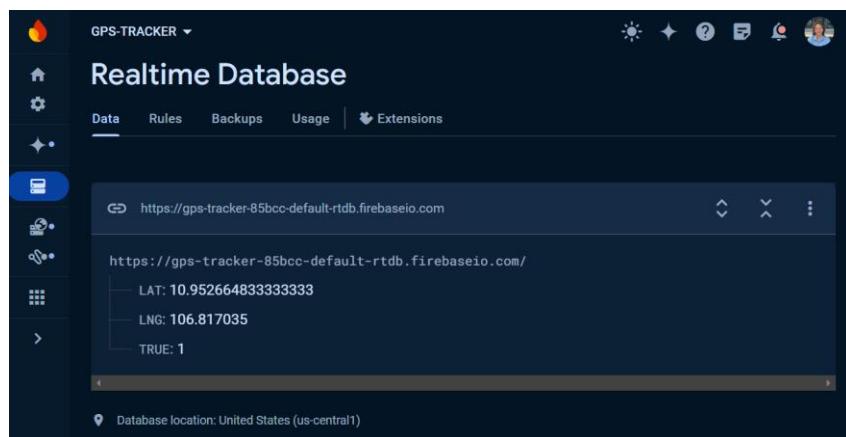
Hình 4.33: Kết quả lưu trữ hình ảnh điểm danh của học sinh

Còn hình ảnh điểm danh của học sinh sẽ được lưu với một dãy số mã hóa bất kì, những hình ảnh này chỉ dùng khi có sự cố xảy ra nên việc mã hóa sẽ giúp tránh lộ thông tin cá nhân hằng ngày của các bé.

4.3.2 Lưu trữ dữ liệu trên Firebase

Hệ thống sử dụng hai tính năng của firebase đó là Realtime Database và Web API để lưu trữ dữ liệu thời gian thực và gọi API về trang web, gọi đường dẫn về phần cứng để đọc dữ liệu.

➤ Kết quả thiết lập Firebase Realtime Database



Hình 4.34: Kết quả của Firebase Realtime Database sau khi thiết lập hệ thống

Phần Realtime Database hệ thống sử dụng ba biến dữ liệu thời gian thực bao gồm LAT (latitude) - vĩ độ hiện tại của chuyến xe, LNG (longitude) - kinh độ hiện tại của chuyến xe mà cảm biến vị trí gửi lên. Cuối cùng là TRUE, kết quả của so sánh vị trí trong Điểm danh xuống trạm mà Ứng dụng điểm danh trả về.

4.4 ĐÁNH GIÁ HỆ THỐNG

Các chức năng hoạt động của toàn bộ hệ thống đã thực thi một cách ổn định nhưng tôi vẫn chưa thực nghiệm hệ thống trên xe rước để đảm bảo độ chính xác và độ nhạy của cảm biến GPS cũng như tốc độ truyền dữ liệu của các phản liệu có ổn định hay không khi xe đưa rước di chuyển.

Hệ thống sử dụng dịch vụ Amazon Rekognition với độ chính xác trên 95% (theo lý thuyết) và khi thực nghiệm với một mẫu dữ liệu nhỏ thì đạt độ chính xác gần như tuyệt đối (100% cho 10 mẫu dữ liệu).

Độ chính xác của cảm biến GPS NEO-M8N trên lý thuyết ở mức CEP (Circular Error Probable) 2,5 mét trong điều kiện tốt, với SBAS (Satellite-Based Augmentation System) có thể cải thiện độ chính xác lên đến 1 mét. Và tốc độ cập nhật lên đến 18 Hz, độ nhảy cảm -167dBm. Kết quả thực nghiệm trên hệ thống cho thấy độ chính xác giao động từ 2-3 mét và cần phải được sử dụng trong môi trường thoáng, không bị chắn quá nhiều.

Về tốc độ phản hồi, hệ thống trả kết quả nhanh (khoảng 4 giây), đáp ứng yêu cầu độ trễ dưới 5 giây. Khi mới chạy trang web, các thao tác phản hồi hơi chậm, nhưng sau khi thực hiện hết các tính năng, thao tác trở nên mượt mà, phản hồi nhanh (khoảng 1,5 giây).

Tài khoản miễn phí AWS có nhiều giới hạn về dung lượng và tài nguyên như Amazon S3 miễn phí 5GB cho 12 tháng, Amazon DynamoDB miễn phí 25GB không giới hạn thời gian, AWS Lambda miễn phí 1 triệu requests mỗi tháng, Amazon Rekognition miễn phí 5000 ảnh mỗi tháng trong 1 năm và Amazon CloudWatch 10 custom metrics và alarms không giới hạn thời gian.

Hệ thống sử dụng Firebase ở Test Mode cho developing giúp cho cơ sở dữ liệu Realtime Database hoàn toàn mở ra Internet. Vì lựa chọn này làm cho hệ thống dễ bị tấn công nên các quy tắc bảo mật cơ sở dữ liệu được cấu hình để ngừng cho phép các yêu cầu sau 30 ngày đầu tiên. Nên cần có kế hoạch bảo trì, cập nhập API mỗi tháng để đảm bảo hệ thống luôn hoạt động.

Giao diện người dùng của hệ thống được tinh chỉnh để có thể truy cập bằng nhiều thiết bị khác nhau chỉ thông qua một đường link. Nhờ vào thiết kế đa thiết bị này, người dùng có thể dễ dàng truy cập vào hệ thống bất kể họ đang sử

dụng máy tính, điện thoại di động, máy tính bảng hay bất kỳ thiết bị nào khác.

Trong đồ án này chỉ mới ứng dụng hệ thống cho một chiều đưa rước đó là đưa học sinh từ trường về nhà (trường hợp này có khả năng xuống sai trạm và dễ bị bỏ quên trên xe nhất) mà chưa cài tiến hệ thống để có thể sử dụng cho cả hai chiều đưa rước.

Ngoài ra việc sử dụng API Google Maps để hiển thị vị trí của chuyến xe vẫn còn đang hạn chế do đó là một tính năng mất phí cũng như việc lưu trữ nhiều hình ảnh học sinh hơn, nhiều thông tin học sinh hơn cũng như thông tin tài xế thông tin chuyến xe vẫn còn rất hạn chế do AWS là một dịch vụ có trả phí, với tài khoản miễn phí một năm thì sẽ có giới hạn về số lượng.

Về phần đánh giá độ tin cậy của hệ thống, độ chính xác của xác thực khuôn mặt và hiển thị thông tin học sinh sau khi xác thực thành công gần như đạt ngưỡng 100% với tốc độ phản hồi là 4,3s đạt tiêu chuẩn với độ trễ cho phép. Tỷ lệ xác thực vị trí của học sinh khi xuống trạm so sánh với vị trí trạm theo thông tin học sinh với sai số khoảng cách là 5m có độ chính xác cao, đảm bảo an toàn cho học sinh và nâng cao độ tin cậy của hệ thống. Cuối cùng, tốc độ cập nhập dữ liệu trên AWS sau khi thao tác chỉnh sửa từ giao diện khá nhanh giúp người quản lý và phụ huynh luôn có được những thông tin chính xác nhất.

CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT

TRIỂN TRONG TƯƠNG LAI

5.1 KẾT LUẬN

Thông qua đề tài “Xây dựng hệ thống hỗ trợ giám sát học sinh trên xe đưa đón ứng dụng AWS” với mô hình có thể hoạt động bình thường và thực hiện được đầy đủ các chức năng được đề ra thông qua các yêu cầu hệ thống ban đầu. Từ đó rút ra các ưu điểm và nhược điểm như sau:

Ưu điểm:

- Hệ thống, mô hình hoạt động, thực hiện được các chức năng theo yêu cầu người dùng với độ chính xác tương đối tốt.
- Hệ thống hoạt động với các tính năng ổn định, đáp ứng được các tiêu chí yêu cầu về thời gian đối với một mô hình truyền gửi thực tế.
- Hệ thống có giao diện người dùng dễ sử dụng, các thao tác thực hiện cũng được tối ưu hóa về đơn giản nhất.
- Các linh kiện phần cứng mà hệ thống sử dụng không nhiều, không tiêu thụ quá nhiều điện năng. Phần giao diện cũng được phát triển để có thể truy cập trên mọi thiết bị điện tử mà không bị khó nhìn.
- Hệ thống đa số sử dụng các dịch vụ giá rẻ và được cung cấp số lượng miễn phí của AWS giúp người dùng không tốn quá nhiều chi phí khi triển khai hệ thống.

Nhược điểm:

- Hệ thống phải xài hai dịch vụ điện toán đám mây để lưu trữ dữ liệu.
- Do sử dụng các tính năng miễn phí trên AWS nên các bước xử lý trên nền tảng vẫn còn bị rườm rà chưa tối ưu triệt để.
- Hệ thống chỉ đang ứng dụng với một chiều đưa rước, từ trường về nhà.
- Hệ thống chưa hỗ trợ hiển thị cùng lúc vị trí nhiều chuyến xe trên

google maps cũng như chỉ đường cho tài xế.

- Chưa phát triển ứng dụng điện thoại để tiện ích hơn cho người dùng.
- Việc điểm danh còn phải thực hiện thủ công (bấm nút chụp hình).

5.2 HƯỚNG PHÁT TRIỂN TRONG TƯƠNG LAI

Một số hướng phát triển cho hệ thống trong tương lai:

- Tích hợp vào một nền tảng điện toán đám mây duy nhất (AWS) để đơn giản hóa và tối ưu hóa quá trình xử lý dữ liệu, giảm chi phí.
- Tối ưu hóa quy trình xử lý dữ liệu bằng cách sử dụng các tính năng nâng cao trên nền tảng điện toán đám mây, loại bỏ các quy trình không cần thiết để cải thiện hiệu suất và tăng tốc độ xử lý.
- Triển khai ứng dụng hỗ trợ cả hai chiều đưa rước, từ nhà đến trường và ngược lại, để nâng cao trải nghiệm người dùng và giảm rủi ro.
- Cải thiện tính năng hiển thị vị trí chuyến xe trên Google Maps và tích hợp chỉ đường, tìm đường đi ngắn nhất.
- Phát triển ứng dụng di động cho phụ huynh và tài xế để tăng tính tiện lợi và trải nghiệm người dùng.
- Tự động hóa quy trình điểm danh bằng công nghệ Computer Vision, detection face theo giây và tự động cắt ảnh hợp lệ gửi lên Rekognition.
- Mở rộng khả năng lưu trữ dữ liệu: Sử dụng các gói dịch vụ nâng cao hoặc tích hợp với các nền tảng lưu trữ khác nhau để tăng khả năng lưu trữ dữ liệu.
- Kiểm tra và cải thiện độ chính xác của cảm biến GPS trên xe để đảm bảo độ chính xác và ổn định của cảm biến GPS, đồng thời tối ưu hóa tốc độ truyền dữ liệu.

Bằng cách triển khai các hướng phát triển này, hệ thống sẽ trở nên hoàn thiện hơn, đáp ứng tốt hơn nhu cầu của người dùng, và nâng cao trải nghiệm sử dụng cho cả phụ huynh, tài xế và học sinh.

TÀI LIỆU THAM KHẢO

- [1] Bảo Hân, “Safe school - Hệ thống theo dõi, bảo đảm an toàn đưa đón học sinh,” Báo Hà nội mới, Aug 18, 2019. [Online]. Available: <https://hanoimoi.vn/safe-school-he-thong-theo-doi-bao-dam-an-toan-dua-don-hoc-sinh-536005.html>. [Accessed April 20, 2024]
- [2] Đức Nhật, “Học sinh chế tạo thiết bị kiểm soát trên xe đưa đón,” Báo Thanh niên, Sep 14, 2020. [Online]. Available: <https://thanhnien.vn/hoc-sinh-che-tao-thiet-bi-kiem-soat-tren-xe-dua-don-185993615.htm>. [Accessed April 20, 2024]
- [3] Khalid Ammar and Mohammad Jalmoud, “A Real-time School Bus Tracking and Monitoring System,” in *2019 IEEE 10th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 17-19 October 2019*, IEEE, 19 December 2019.
- [4] Sanika Zende, “AWS Compute and Storage Services,” Medium, Oct 5, 2023. [Online]. Available: <https://medium.com/@sanikazende27/aws-compute-and-storage-services-6b8ebf582ae6>. [Accessed April 24, 2024]
- [5] Purti Aggarwal, “Getting Start with NextJs,” Medium, Mar 24, 2024. [Online]. Available: <https://purtiaggarwal1997.medium.com/getting-start-with-nextjs-dfdae6f8f1c8>. [Accessed April 24, 2024]
- [6] Douglas Karr, “Tailwind CSS: A Utility-First CSS Framework and API for Rapid, Responsive Design,” Martech.zone, Sep 13, 2021. [Online]. Available: <https://martech.zone/tailwind-css-framework/>. [Accessed April 26, 2024]