

BỘ GIÁO DỤC VÀ ĐÀO TẠO

BÁO CÁO TỔNG KẾT

**ĐỀ TÀI THAM GIA XÉT TẶNG GIẢI
THƯỜNG KHOA HỌC VÀ CÔNG NGHỆ DÀNH
CHO SINH VIÊN TRONG CƠ SỞ GIÁO DỤC ĐẠI
HỌC NĂM 2025**

XÂY DỰNG ỨNG DỤNG CHỈ ĐƯỜNG SỬ DỤNG TÍN HIỆU WIFI

Thuộc nhóm ngành khoa học: Khoa học Kỹ thuật và Công nghệ

Chuyên ngành cụ thể của lĩnh vực khoa học và công nghệ: Công nghệ thông tin

MỤC LỤC

DANH MỤC HÌNH ẢNH	I
DANH MỤC BẢNG	II
DANH MỤC VIẾT TẮT	III
MỞ ĐẦU	1
1. Tổng quan tình hình nghiên cứu của đề tài	1
2. Lý do chọn đề tài	3
3. Mục tiêu đề tài	3
4. Phương pháp nghiên cứu	3
5. Đối tượng nghiên cứu	5
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT	6
1.1 Hệ thống định vị trong nhà	6
1.1.1 Khái niệm	6
1.1.2 Ứng dụng	6
1.2. Các phương pháp định vị trong nhà	9
1.2.1 Phương pháp học máy	9
1.2.2 Phương pháp học sâu	10
1.3 Các thư viện hỗ trợ xây dựng ứng dụng định vị và chỉ đường trong nhà ..	12
1.3.1 Dart	12
1.3.2 Flutter	13
1.3.3. FastAPI	14
1.3.4. Google Colab	15
1.3.5. MongoDB Atlas	16
CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	18
2.1 Phân tích chức năng ứng dụng	18
2.2. Sơ đồ Use-case	18
2.3. Kiến trúc ứng dụng	20
CHƯƠNG 3. PHƯƠNG PHÁP ĐỀ XUẤT	22
3.1. Hệ thống định vị thông qua Wifi	22

3.2. Mô tả tập dữ liệu	22
3.2. Tiền xử lý dữ liệu	24
3.3. Bản đồ định vị và chỉ số quy đổi	26
3.4. Tiến hành huấn luyện mô hình	28
3.4.1. Thuật toán định vị tất định	28
3.4.2. Mô hình học sâu	29
3.5. Kết quả đánh giá và so sánh các mô hình	32
CHƯƠNG 4:XÂY DỰNG ỨNG DỤNG	37
4.1. Cấu trúc thư mục	37
4.1.1. Frontend	37
4.2.2 Backend	38
4.2. Xây dựng API	40
4.3. Lựa chọn thuật toán dẫn đường	41
4.3.1 Thuật toán Dijkstra	41
4.3.2 Thuật toán A*	42
4.3.3 Thuật toán A*	43
4.4. Kết quả đạt được	44
4.4.1. Welcome Screen	44
4.4.2. Giao diện chính	46
4.4.3. Giao diện Map	47
4.4.4. Thông tin địa điểm	48
4.4.5. Chỉ đường	49
4.4.5. Loại địa điểm	50
4.4.6. Tìm đường đi tự chọn	51
4.4.7. Giới thiệu địa điểm	52
KẾT LUẬN	53
LỜI CẢM ƠN	54
DANH MỤC TÀI LIỆU THAM KHẢO	55

DANH MỤC HÌNH ẢNH

Hình 1. Khoảng cách giữa hai điểm trong không gian 2 chiều. [3]	10
Hình 2. k-Nearest Neighbor [3]	10
Hình 3. Mô hình định vị trong nhà cơ bản dựa trên học máy.[2]	11
Hình 4. Sơ đồ Use-case	18
Hình 5. Kiến trúc hệ thống	20
Hình 6. Hệ thống định vị thông qua WiFi	22
Hình 7. Sơ đồ chỉ số quy đổi	26
Hình 8. Sơ đồ mô hình theo bài báo [6]	30
Hình 9. Sơ đồ hệ thống chung của mô hình	31
Hình 10. Sơ đồ mô hình xây dựng	32
Hình 11. Sơ đồ mô hình WKNN - CNN	32
Hình 12. Sơ đồ mô hình WKNN - LSTM	32
Hình 13. Sơ đồ mô hình WKNN - LSTM Regression	32
Hình 14. Kết quả mô hình phân lớp	34
Hình 15. Kết quả mô hình phân lớp WKNN - CNN	34
Hình 16. Kết quả mô hình phân lớp WKNN - LSTM	35
Hình 17. Kết quả mô hình phân lớp WKNN - LSTM Regression	35
Hình 18. Biểu đồ thể hiện độ chính xác của phương pháp và mô hình	35
Hình 19. Thống kê độ sai khoảng cách trong thực tế	36
Hình 20. Cấu trúc thư mục FrontEnd	37
Hình 21. Cấu trúc thư mục BackEnd	38
Hình 22. Tập tin requirement.txt	39
Hình 23. Mã giả tập tin main.py	39
Hình 24. Mã giả xử lý giao tiếp BackEnd & Model	40
Hình 25. Thống kê API của dự án	41
Hình 26. Thuật toán Dijkstra	42
Hình 27. Thuật toán A*	43
Hình 28. Welcome Page	45
Hình 29. Giao diện chính	46
Hình 30. Giao diện Map	47
Hình 31. Thông tin POI khi tương tác trên Map	48
Hình 32. Hình ảnh chức năng nút "Chỉ đường"	49
Hình 33. Hình ảnh chức năng "Loại địa điểm"	50
Hình 34. Hình ảnh chức năng "Tìm kiếm địa điểm"	51
Hình 35. Hình ảnh trang thông tin	52

DANH MỤC BẢNG

Bảng 1. Thống kê cột dữ liệu mẫu	24
Bảng 2. So sánh kết quả định vị trước và sau chuẩn hóa	25
Bảng 3. Tập dữ liệu đầu vào có 37 đặc trưng	26
Bảng 4. Thống kê điểm thu thập dữ liệu theo ma trận	27
Bảng 5. Thống kê tham số của thuật toán áp dụng	30

DANH MỤC VIẾT TẮT

STT	Chữ viết tắt	Viết đầy đủ
1	GPS	Global Positioning System
2	IPS	Indoor Positioning System
3	AI	Artificial Intelligence
4	DL	Deep Learning
5	ML	Machine Learning
6	CNTT	Công nghệ thông tin
7	CNN	Convolution Neural Network
8	KNN	Recurrent Neural Networks
9	WKNN	Weighted k-Nearest Neighbors
10	LSTM	Long Short-Term Memory Networks

MỞ ĐẦU

1. Tổng quan tình hình nghiên cứu của đề tài

Hiện nay, hệ thống định vị toàn cầu (Global Positioning System - GPS) có thể xác định vị trí của một người/vật thể gần như trên toàn bộ bề mặt Trái Đất với sai số chỉ từ 3-5 m. Tuy nhiên, hệ thống GPS không thể xác định được một cách chính xác vị trí của người dùng trong môi trường kín, ví dụ như các tòa nhà, siêu thị, bệnh viện, v.v. Các bức tường, trần nhà, đồ đạc trong nhà là các yếu tố gây ra các hiệu ứng đa đường, phản xạ, tán xạ, và làm suy yếu tín hiệu truyền từ vệ tinh [1]. Chính vì vậy, việc nghiên cứu và phát triển các hệ thống định vị trong môi trường kín (Indoor Positioning System - IPS) như trong nhà, bệnh viện, trường học, v.v. là thực sự cần thiết.

Nhiều kết quả nghiên cứu trong và ngoài nước đã được công bố liên quan đến IPS. Các kỹ thuật định vị điển hình dựa trên thông tin cường độ tín hiệu nhận được (RSSI), thời gian đến (ToA), chênh lệch thời gian đến (TDoA), góc đến (AoA) và thông tin trạng thái kênh (CSI) đã được đề xuất bằng cách sử dụng nhiều công nghệ truy cập khác nhau như WiFi, Bluetooth, băng tần cực rộng (UWB), và thẻ nhận dạng tần số vô tuyến (RFID) để định vị trong nhà. Hầu hết các kỹ thuật định vị đều yêu cầu ít nhất ba nút neo đã biết để tính toán vị trí của mục tiêu chưa biết. Một số kỹ thuật không cần định vị như phương pháp Centroid và kỹ thuật DV hop cũng được nghiên cứu trong tài liệu. Tất cả các phương pháp này đều gặp phải vô số vấn đề như độ chính xác kém, độ phức tạp cao và không đáng tin cậy trong khi hầu hết các thiết bị định vị đều thiếu khả năng xử lý mạnh [2].

Kỹ thuật WiFi Fingerprinting là một trong các kỹ thuật phổ biến khi sử dụng công nghệ WiFi phục vụ cho việc định vị trong môi trường kín. Kỹ thuật này bao gồm hai giai đoạn chính là giai đoạn ngoại tuyến (offline phase) và giai đoạn trực tuyến (online phase). Ở giai đoạn ngoại tuyến, các giá trị RSS từ những AP khác nhau sẽ được thu thập tại mỗi điểm tham chiếu RP trong khu vực cần định vị. Các điểm RP được thiết lập với các tọa độ được biết trước. Các tọa độ này có thể được chọn theo cách phân bố đều, tức là khoảng cách giữa hai điểm RP liên tiếp là bằng nhau, hoặc chọn ngẫu nhiên các vị trí khác nhau. Tất cả các giá trị RSS và tọa độ tại từng RP sẽ được lưu trữ thành một bộ dữ liệu ngoại tuyến. Ở giai đoạn trực tuyến, vị trí chưa xác định của người dùng (hay điểm thực nghiệm TP) sẽ được tính toán dựa trên việc so sánh tập giá trị RSS của từng RP trong bộ dữ liệu với các giá trị RSS tại vị trí mà thiết bị di động của người dùng thu thập được. Quá trình so sánh trên có thể được thực hiện bằng cách sử

dụng những thuật toán định vị khác nhau như thuật toán tất định, thuật toán xác suất và thuật toán học máy.

Wang và các cộng sự [1] đề xuất một phương pháp định vị trong nhà kết hợp giữa autoencoder xếp chồng (Stacked autoencoder - SAE) và mạng thần kinh tích chập CNN. SAE được sử dụng để trích xuất các đặc trưng quan trọng từ dữ liệu RSS. Đồng thời, một phương pháp mã hóa nhiệt đặc biệt được áp dụng để liên kết tòa nhà với các tầng. Dữ liệu sau đó được đưa vào mạng CNN để huấn luyện. Hệ thống được đề xuất đã được đánh giá trên bộ dữ liệu UJIIndoorLoc (một bộ dữ liệu mã nguồn mở dành cho định vị trong nhà) và hiệu năng của nó đã được so sánh với một số phương pháp truyền thống. Kết quả cho thấy hệ thống đề xuất đạt được độ chính xác tốt ở cả cấp độ xác định tòa nhà và cấp độ xác định tầng với độ chính xác 96,73% dành cho dự đoán tầng, độ chính xác cho tòa nhà là 100%, và sai số định vị MSE là 11.56 m.

Trong nghiên cứu [2], Duong-Bao và các cộng sự đã tiến hành thực hiện quá trình thu thập dữ liệu, triển khai, và đánh giá hai thuật toán định vị phổ biến là KNN và WKNN trên bộ dữ liệu thu thập tại phòng làm việc trong 4 tháng. Nhóm tác giả nghiên cứu ảnh hưởng của điều kiện môi trường đến độ chính xác định vị, bao gồm mật độ người, thiết bị, nhiệt độ, chiều cao thiết bị, hướng thu thập, và thời gian thu thập. Bộ dữ liệu có 205 vị trí thu thập, sắp xếp theo lưới với khoảng cách 0.5 m giữa các điểm, mỗi vị trí thu thập 100 lần từ năm AP cố định, tổng cộng 20,500 mẫu cho giai đoạn ngoại tuyến và 410 mẫu cho giai đoạn trực tuyến cho hai trường hợp kiểm tra khác nhau. Nhóm tác giả áp dụng thử nghiệm hai thuật toán KNN và WKNN với các giá trị K thay đổi nhằm chọn ra giá trị K tối ưu. Kết quả cho thấy WKNN có kết quả tốt hơn KNN. Kết quả sai số tại $K=3$ là thấp nhất với sai số trung bình ở trường hợp 1 là $KNN = 1.22$ m, $WKNN = 1.17$ m và trường hợp 2 là $KNN = 1.32$ m, $WKNN = 1.26$ m. Nhóm tác giả kết luận rằng kỹ thuật WiFi Fingerprinting mặc dù cho kết quả định vị tương đối tốt, tuy nhiên hiệu năng của kỹ thuật bị ảnh hưởng bởi sự thay đổi môi trường khiến giá trị RSS dao động và độ chính xác định vị giảm.

Trong nghiên cứu [3], Moghtadaiee và các cộng sự đã thực nghiệm định vị trên bộ dữ liệu với số lượng AP khác nhau và kết quả cho thấy chỉ cần năm giá trị RSS thu được từ năm AP đã đủ cho việc định vị. Kết quả này cho thấy không phải tất cả các RSS thu được từ các AP đều cần thiết cho việc định vị. Vì vậy, nhiều nhóm nghiên cứu [4-5] đã đề xuất các giải pháp chọn AP dựa trên giá trị của RSS, nhằm tăng chất lượng định vị. Tuy nhiên, tác động của hiệu ứng đa đường và suy giảm tín hiệu có thể làm cho giá trị RSS của cùng một AP ở cùng một vị trí tại các thời điểm khác nhau không giống nhau.

Ngô và các cộng sự [6] sử dụng bộ dữ liệu UJIIndoorLoc, vốn là một tập dữ liệu lớn, để tiến hành xử lý dữ liệu nhằm giảm thời gian tính toán và tăng chất lượng. Các tác giả chỉ ra rằng bài toán phân lớp giá trị giữa các điểm càng chênh lệch lớn thì càng dễ phân lớp, chẳng hạn như tòa nhà được phân lớp tốt hơn nếu các đặc trưng có nhiều thông tin của tòa nhà, từ đó tiến hành việc loại bỏ các đặc trưng có sự khác biệt thấp, trùng nhau. Quá trình tiền xử lý dữ liệu được áp dụng cho pha kiểm thử của mô hình phân lớp. Đồng thời, nhóm tác giả đề xuất một phương pháp xây dựng mô hình hai giai đoạn bằng cách cho dữ liệu RSS đi qua phương pháp WKNN. Sau đó, các dự đoán của WKNN được chuyển đổi thành các vec-tơ gộp vào bộ dữ liệu ban đầu rồi mới huấn luyện mô hình CNN. Kết quả cho thấy rằng mô hình trên đạt tỷ lệ phân lớp vượt trội so với kết quả của các mô hình dùng các thuật toán độc lập khác trên cùng bộ dữ liệu UJIIndoorLoc.

2. Lý do chọn đề tài

Trong bối cảnh nhu cầu sử dụng công nghệ để cải thiện trải nghiệm di chuyển ngày càng cao, việc phát triển các ứng dụng chỉ đường trở nên cần thiết. Đề tài "Xây dựng ứng dụng chỉ đường sử dụng tín hiệu Wi-Fi" được hình thành với mục tiêu cung cấp một giải pháp hiện đại và tiện ích cho người dùng trong việc tìm đường và xác định vị trí chính xác.

Ứng dụng này sử dụng tín hiệu Wi-Fi để xác định vị trí của người dùng và tính toán lộ trình di chuyển, thay vì phụ thuộc vào các phương pháp truyền thống như GPS. Bằng cách tận dụng mạng WiFi sẵn có, hệ thống sẽ giúp người dùng dễ dàng tìm kiếm địa điểm, vẽ đường đi, và nhận chỉ dẫn một cách nhanh chóng và chính xác.

Ngoài tính năng chỉ đường, ứng dụng còn hỗ trợ người dùng theo dõi các điểm quan tâm, như khu vực tự học, cửa ra vào hay phòng tạp chí gần đó. Hệ thống được thiết kế linh hoạt, dễ sử dụng và có khả năng mở rộng trong tương lai.

3. Mục tiêu đề tài

Cung cấp một giao diện trực quan và tiện lợi cho người dùng, giúp họ có thể nắm bắt được cấu trúc, thông tin địa điểm và đường đi trong thư viện. Ứng dụng di động này sẽ hỗ trợ cho việc tìm kiếm đường đi và thông tin địa điểm tại tầng 1 của tòa nhà thư viện Trường Đại học Đà Lạt.

4. Phương pháp nghiên cứu

Nghiên cứu tham khảo: Tiến hành nghiên cứu tham khảo về các ứng dụng web tương tự đã được phát triển trong lĩnh vực quản lý di sản văn hóa hoặc các lĩnh vực

tương đồng. Việc này giúp hiểu rõ các tiêu chuẩn, các tính năng phổ biến và tiềm năng của ứng dụng trong lĩnh vực này.

Phân tích yêu cầu: Sau khi có kiến thức cơ bản từ nghiên cứu tham khảo, tiến hành phân tích yêu cầu cho ứng dụng chỉ đường. Điều này bao gồm tìm hiểu và ghi nhận các yêu cầu cụ thể từ người sử dụng và các bên liên quan như.

Thiết kế giao diện người dùng: Tiếp theo, tiến hành thiết kế giao diện người dùng cho ứng dụng web. Quá trình này sẽ tạo ra một giao diện trực quan, dễ sử dụng và tương thích trên nhiều thiết bị khác nhau. Thiết kế giao diện sẽ dựa trên yêu cầu và phản hồi từ người sử dụng để đảm bảo sự hài lòng và tương tác thuận tiện.

Xây dựng cơ sở dữ liệu: Xác định và xây dựng cơ sở dữ liệu để lưu trữ thông tin về các địa điểm đặc biệt. Các thông tin này bao gồm tên địa điểm, thông tin địa điểm, tọa độ địa lý và các tài liệu liên quan. Cơ sở dữ liệu sẽ được thiết kế để hỗ trợ các chức năng quản lý và tra cứu hiệu quả của ứng dụng.

Phát triển ứng dụng di động: Tiếp theo, thực hiện quá trình phát triển ứng dụng web theo thiết kế đã được xác định. Quá trình này bao gồm việc xây dựng giao diện người dùng, kết nối và tương tác với cơ sở dữ liệu, xử lý dữ liệu và triển khai các tính năng hiển thị địa điểm, vẽ đường đi và phát ra âm thanh dẫn đường.

Kiểm thử và đánh giá: Cuối cùng, sẽ tiến hành kiểm thử và đánh giá ứng dụng di động. Việc này bao gồm kiểm tra tính năng, độ tin cậy, hiệu suất của ứng dụng. Chúng ta sẽ sử dụng các phương pháp kiểm thử phù hợp để đảm bảo ứng dụng hoạt động một cách ổn định và đáp ứng yêu cầu của người sử dụng.

Đối với việc xây dựng một ứng dụng chỉ đường, chúng ta có rất nhiều sự lựa chọn về ngôn ngữ lập trình, server – máy chủ API, Database, ... Đối với những vấn đề này, em đã thống nhất chọn những phần mềm và công cụ sử dụng như sau:

- Hệ QTCSDL: MongoDB
- Server: FastAPI
- Giao diện: Dart
- Framework: Flutter
- Các công cụ hỗ trợ: Git

5. Đối tượng nghiên cứu

- Nghiên cứu về định vị trong nhà (Indoor Position System-IPS)
- Nghiên cứu và xây dựng bộ dữ liệu cho định vị trong nhà
- Nghiên cứu và xây dựng ứng dụng chỉ đường.

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1.1 Hệ thống định vị trong nhà

1.1.1 Khái niệm

Trong nghiên cứu [2], Duong-Bao và các cộng sự định vị trong nhà là quá trình xác định vị trí của thiết bị hoặc người dùng trong môi trường trong nhà như trong căn hộ, trường học, trung tâm thương mại, bệnh viện,... Hệ thống định vị toàn cầu GPS (Global Positioning System) và hệ thống định vị trong nhà IPS (Indoor Positioning System) là hai nguồn cung cấp thông tin vị trí để xây dựng nên các dịch vụ dựa trên vị trí.

Hệ thống GPS hiện nay có thể định vị các đối tượng trên bề mặt Trái Đất, nhưng không thể định vị một cách chính xác các đối tượng ở môi trường trong nhà. Thêm vào đó, việc xác định vị trí trên bề mặt Trái Đất với khoảng cách lớn sẽ dẫn tới độ chính xác thấp, có thể sai lệch từ vài chục mét lên đến vài chục kilômét so với vị trí thực, mà đối với môi trường trong nhà, diện tích nhỏ hơn rất nhiều thì việc xác định vị trí bằng hệ thống GPS là không thể chấp nhận với độ sai lệch lớn như vậy.

Thêm vào đó, môi trường trong nhà phức tạp hơn môi trường ngoài trời rất nhiều, với nhiều yếu tố liên quan tác động trực tiếp hoặc gián tiếp đến kết quả định vị. Chẳng hạn, môi trường trong nhà có diện tích nhỏ, nhiều vật cản, nhiều đồ vật, đối tượng cùng xuất hiện trong một không gian nhỏ, v.v.

Các hệ thống định vị trong nhà hay IPS đã được nghiên cứu trong vài thập kỷ gần đây. Các nghiên cứu tập trung chủ yếu trong công nghiệp liên quan đến mạng lưới cảm biến và công nghệ rô-bốt. Sự phát triển mạnh mẽ của các thiết bị điện thoại di động thông minh và các thiết bị giao tiếp không dây đã tạo ra tiền đề để xây dựng IPS và các ứng dụng liên quan đến vị trí người dùng.

1.1.2 Ứng dụng

Ứng dụng điều hướng (Navigation)

Điều hướng là chức năng cơ bản và quan trọng nhất của một hệ thống IPS. Điều hướng tức là đưa ra các chỉ dẫn để người dùng có thể đến nơi họ mong muốn. Khác với GPS, tập trung vào điều hướng cho các phương tiện giao thông như xe mô tô, ô tô, máy bay, v.v. thì IPS tập trung điều hướng cho người đi bộ trong phạm vi tòa nhà, khu triển lãm, v.v. Ví dụ, khi tổ chức các sự kiện tại khu vực "Hội trường thư viện" nếu chưa làm quen được với cấu trúc thư viện người , IPS có chức năng điều hướng sẽ đưa ra chỉ dẫn đường đi từ vị trí hiện tại của người đó đến khán phòng một cách tiện lợi và nhanh chóng thông qua ứng dụng trên điện thoại thông minh của người đó.

Một chức năng khác có thể được tích hợp vào ứng dụng điều hướng là cập nhật thông tin về thay đổi cấu trúc của tòa nhà. Chẳng hạn, một lối đi đang sửa chữa sẽ được thông báo thông qua ứng dụng điện thoại để tránh mọi người đi vào trong, đồng thời hướng dẫn mọi người di chuyển bằng lối đi khác.

Ngoài ra, dựa trên thông tin vị trí mà IPS cung cấp, có thể tạo ra ứng dụng chỉ dẫn cho người lao động đến đúng vị trí khu vực được phân công. Điều này áp dụng vào những khu vực lớn như công trường, khu vui chơi, trung tâm thương mại, v.v. nơi người lao động được phân công vị trí làm việc theo ca của họ.

Ứng dụng theo dõi (Tracking)

Ngoài việc điều hướng người dùng, IPS cần phải theo dõi liên tục vị trí của người dùng trong quá trình di chuyển để có thể điều chỉnh lối đi. Việc theo dõi người dùng nhằm cung cấp thông tin vị trí như người đó đang đứng ở phòng nào, gần những khu vực nào, v.v.

Các LBS sẽ khai thác những thông tin theo dõi vị trí người dùng để cung cấp dịch vụ gần đó. Các nhà hàng có thể quảng cáo thông tin về thực đơn, giá cả cho người dùng khi họ đứng gần đó hay cửa hàng thời trang có thể thông báo đến người dùng về chương trình khuyến mãi, v.v.

Một số IPS được sử dụng trong việc quản lý và giám sát những tài sản quý giá. Ví dụ, các thiết bị, máy móc trong bệnh viện rất đắt tiền, nhiều trong số chúng có kích

thước nhỏ và dễ dàng bị lấy cắp. Các hệ thống IPS có nhiệm vụ theo dõi, giám sát vị trí và có thể báo động nếu như các thiết bị này nằm ngoài vùng theo dõi của hệ thống. Hơn nữa, việc sử dụng IPS để theo dõi vị trí các tài sản có thể hỗ trợ trong việc quản lý một cách số hóa những tài sản này. Chẳng hạn, người dùng có thể kiểm tra số lượng của mỗi loại tài sản thông qua ứng dụng trên điện thoại di động dựa trên thông tin về vị trí của tài sản đó.

Ứng dụng bảo vệ (Security)

Việc theo dõi vị trí có thể ngăn chặn những sự xâm nhập trái phép vào các khu vực cấm. Các khu quân sự, phòng thí nghiệm là những nơi nguy hiểm và có nhiều khu vực hạn chế xâm nhập. Các IPS dễ dàng xác định được vị trí của đối tượng và cảnh báo nếu đối tượng xâm nhập vào các khu vực cấm. Thông thường, các IPS dựa trên thị giác máy tính sẽ phù hợp với việc này hoặc có thể trang bị cho đối tượng các thiết bị theo dõi vị trí, mặc dù có thể không hiệu quả nếu đối tượng tháo bỏ thiết bị.

Ứng dụng giám sát an toàn (Monitoring for Safety)

Ngoài khả năng xác định vị trí, các IPS còn có thể theo dõi một số chuyển động của cơ thể người thông qua các cảm biến gia tốc, cảm biến góc, v.v. tích hợp trên thiết bị di động thông minh như điện thoại, đồng hồ, máy tính bảng, v.v.

Những người lao động làm việc trong môi trường nguy hiểm, có rủi ro tai nạn cao cần có những công cụ bảo vệ nghiêm ngặt. Một số IPS có khả năng phát hiện các tình huống trượt chân và ngã của người lao động. Ví dụ, các công nhân xây dựng phải làm việc trong môi trường rất nguy hiểm và thường ở độ cao rất lớn. Việc phát hiện tình huống trượt chân và ngã của người lao động kịp thời sẽ giảm đi tổn thất về người.

Ngoài ra, người cao tuổi và những người không có khả năng tự chăm sóc bản thân là những đối tượng cần được theo dõi. Phát hiện kịp thời các tình huống tai nạn sẽ giúp những người này được trợ giúp y tế một cách nhanh chóng.

Ứng dụng cứu hộ khẩn cấp (Emergency)

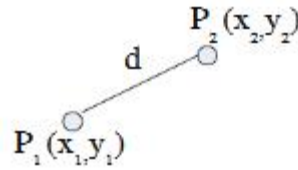
Các sự cố tai nạn là điều không mong muốn và cũng không thể tránh khỏi. Việc giải cứu nạn nhân và giảm thiểu thiệt hại là hướng giải quyết cần thiết khi xảy ra tai nạn. IPS có thể hỗ trợ trong việc xác định vị trí của nạn nhân để giúp các nhân viên cứu hộ nhanh chóng giải cứu. Ví dụ, một tòa nhà cao tầng xảy ra hỏa hoạn, nhiều người sẽ bị mắc kẹt một số vị trí khó đoán, IPS sẽ giải quyết vấn đề xác định vị trí nạn nhân.

1.2. Các phương pháp định vị trong nhà

1.2.1 Phương pháp học máy

Học máy (Machine Learning) là một nhánh của trí tuệ nhân tạo (AI), cho phép hệ thống học hỏi từ dữ liệu và cải thiện hiệu suất mà không cần phải lập trình chi tiết cho từng tác vụ cụ thể. Học máy có thể chia thành ba loại chính: học có giám sát, học không giám sát và học bán giám sát. Trong học có giám sát, mô hình học từ dữ liệu đã được gán nhãn để dự đoán kết quả cho dữ liệu mới. Học không giám sát không yêu cầu dữ liệu có nhãn và giúp phân nhóm hoặc tìm ra các mẫu trong dữ liệu. Học bán giám sát là sự kết hợp của hai phương pháp trên, sử dụng một lượng nhỏ dữ liệu có nhãn và nhiều dữ liệu không nhãn để huấn luyện mô hình. Học máy được ứng dụng rộng rãi trong các lĩnh vực như nhận dạng hình ảnh, phân tích ngôn ngữ tự nhiên, dự đoán tài chính và nhiều ứng dụng khác.

Theo Hằng và các cộng sự [3] thuật toán k láng giềng gần nhất là một thuật toán học máy và phân lớp có giám sát cơ bản nhất. Thuật toán này sử dụng kết quả đo mức độ tương đồng để dự đoán giá trị của các điểm dữ liệu mới. Điều này có nghĩa là, điểm dữ liệu mới sẽ được phân loại vào một lớp đối tượng dựa trên mức độ tương đồng của nó với các điểm trong tập huấn luyện. Mức độ tương đồng giữa dữ liệu mới và dữ liệu huấn luyện có thể được tính toán bằng cách khoảng cách Euclide, Manhattan hoặc Hamming. Trong đó, khoảng cách đó là Euclide được sử dụng phổ biến nhất. Hình 1 biểu diễn khoảng cách giữa hai điểm trong không gian 2 chiều và công thức tính khoảng cách Euclid được biểu diễn theo công thức (1).



Hình 1. Khoảng cách giữa hai điểm trong không gian 2 chiều. [3]

Theo Xuesheng Peng và các cộng sự [10] weighted k-Nearest Neighbor (WKNN) là một phiên bản cải tiến của thuật toán k-Nearest Neighbor (KNN) – một thuật toán học có giám sát trong máy học, có thể được sử dụng cho cả bài toán phân loại và hồi quy. KNN giả định rằng dữ liệu trong cùng một cụm (cluster) là giống nhau và có cùng đặc điểm.

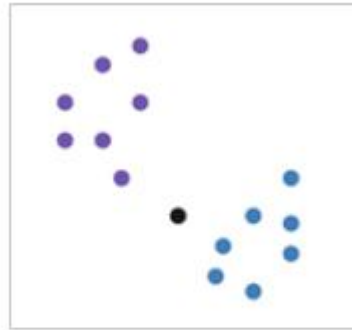


Figure 1: k-Nearest Neighbor

Hình 2. k-Nearest Neighbor [3]

Trong nghiên cứu [6] của Haizhao Lu và cộng sự, khác với thuật toán KNN truyền thống – vốn gán trọng số như nhau cho tất cả các điểm lân cận gần nhất – thuật toán WKNN nâng cao khả năng đóng góp cục bộ giữa các điểm lân cận bằng cách gán trọng số khác nhau cho các mẫu dựa trên khoảng cách, từ đó có thể cải thiện độ chính xác.

1.2.2 Phương pháp học sâu

Học sâu (Deep Learning) là một nhánh con của học máy và trí tuệ nhân tạo, sử dụng mạng nơ-ron nhân tạo nhiều lớp để mô phỏng cách thức học hỏi của não bộ con người. Mạng nơ-ron trong học sâu có nhiều lớp ẩn, giúp mô hình học được các đặc

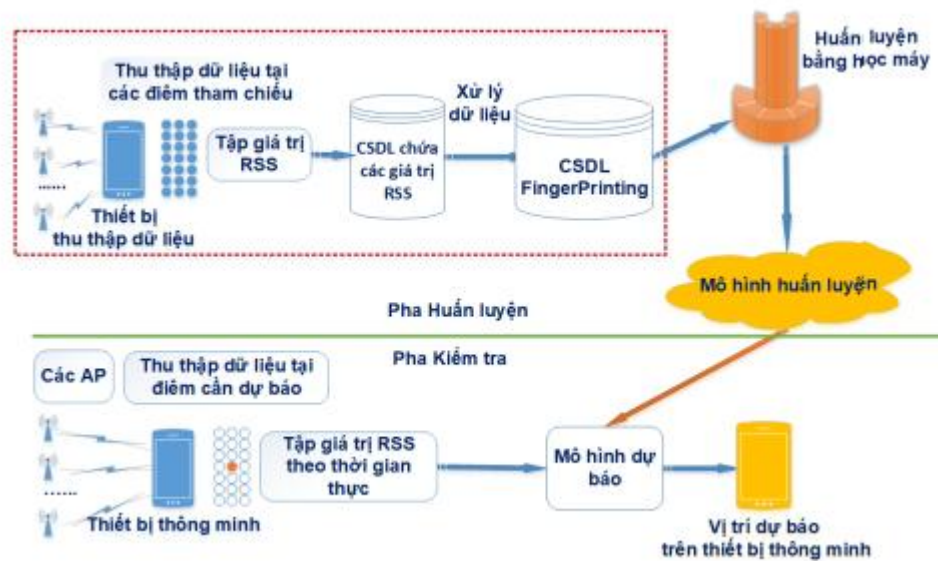
trung phức tạp và trừu tượng từ dữ liệu mà không cần sự can thiệp của con người trong quá trình trích xuất đặc trưng. Học sâu yêu cầu một lượng dữ liệu lớn và tài nguyên tính toán mạnh mẽ, đặc biệt là GPU để xử lý các phép toán phức tạp.

Các phương pháp truyền thống như xác suất, rừng ngẫu nhiên (random forest), máy vector hỗ trợ (support vector machine - SVM), và k-lân cận gần nhất (KNN) gặp phải vấn đề độ phức tạp tính toán cao.

Trong khi đó, các phương pháp học sâu dựa trên mạng nơ-ron tích chập (CNN) có thể cải thiện độ chính xác định vị và đơn giản hóa quá trình tính toán.

Các phương pháp lọc và nội suy (interpolation) giúp xử lý dao động tín hiệu và làm phong phú tập dữ liệu. Trong nghiên cứu [7], đề xuất một hệ thống định vị trong nhà dựa trên dấu vân tay WiFi sử dụng mạng nơ-ron tích chập (CNN).

Mô hình được đề xuất có một hệ thống khung hoàn chỉnh như minh họa trong Hình 1, bao gồm giai đoạn huấn luyện dữ liệu ngoại tuyến và giai đoạn định vị trực tuyến trong quá trình định vị.



Hình 3. Mô hình định vị trong nhà cơ bản dựa trên học máy.[2]

Trong giai đoạn ngoại tuyến, mô hình sử dụng tập dữ liệu mã nguồn mở UJIIndoorLoc, bao gồm tập huấn luyện và tập kiểm tra.

Một phần tư mẫu dấu vân tay từ tập huấn luyện được chọn ngẫu nhiên làm tập kiểm thử.

Sau khi loại bỏ các mẫu không phát hiện được cường độ tín hiệu RSSI, dữ liệu sẽ được chuẩn hóa để thu được tập huấn luyện và tập kiểm thử.

Hiện tại, mô hình thực hiện định vị vị trí của mỗi yêu cầu bằng cách so khớp dữ liệu dấu vân tay nhận được và trả về kết quả định vị cho người yêu cầu.

1.3 Các thư viện hỗ trợ xây dựng ứng dụng định vị và chỉ đường trong nhà

1.3.1 Dart

Dart, là một ngôn ngữ lập trình mạnh mẽ và hiệu quả. được phát triển bởi Google được thiết kế với mục tiêu đơn giản, tiện ích và dễ sử dụng, nhưng vẫn đảm bảo hiệu suất cao và khả năng mở rộng.

Ưu điểm	Nhược điểm
- Hệ sinh thái: Có thư viện cốt lõi và hệ sinh thái gồm nhiều package - Hiệu suất: Cung cấp tối ưu hóa việc biên dịch giúp dự đoán hiệu suất cao, khởi động nhanh trên các thiết bị. - Cú pháp: Dễ gần, có thể tiếp cận nhanh chóng nhờ vào cú pháp và định hướng đối tượng. - Nền tảng: Có thể áp dụng trên nhiều nền tảng như thiết bị di	Ít phổ biến: So với JavaScript, Python thì Dart chưa phổ biến rộng rãi. Backend: Có các framework tuy nhiên chưa được tối ưu bằng với NodeJS, Java hay Python trong việc xây dựng REST API, WebSocker.

động Android và cả IOS, website chuyển mã từ Dart sang JavaScript.	
--	--

Cú pháp Dart rõ ràng và súc tích, công cụ đơn giản nhưng mạnh mẽ. Dart hỗ trợ việc viết Reactive thông qua việc cung cấp Reactive Dart, để quản lý các phương thức, chẳng hạn như các Widget UI, bằng việc phân bổ phương thức nhanh và GC. Dart còn hỗ trợ cho việc xây dựng bằng phương thức không đồng bộ và đối tượng Future, Stream thông qua API.

1.3.2 Flutter

Flutter là một bộ công cụ phát triển ứng dụng đa nền tảng, được giới thiệu bởi Google, sử dụng ngôn ngữ Dart để lập trình. Các ứng dụng được xây dựng với Flutter có thể chạy trên các nền tảng phổ biến như Android, iOS, Raspberry Pi,...

Ưu điểm	Nhược điểm
● Hot reload – Các thay đổi được sửa đổi trên ứng dụng được hiển thị ngay lập tức giúp cho các nhà phát triển chỉnh sửa giao diện tiện lợi và nhanh chóng. Tuy nhiên, những sửa đổi cần thời gian để được hiển thị. ● Phát triển đa nền tảng – Flutter hỗ trợ cho việc phát triển ứng dụng đa nền tảng. Trong trường hợp này, ứng dụng có thể được lập trình sau đó một dự án có thể triển	Giới hạn chỉ dành cho thiết bị di động – Flutter hiện tại vẫn chỉ hỗ trợ chủ yếu cho thiết bị di động, việc xây dựng ứng dụng trên web vẫn chưa được tối ưu. Rất ít thư viện – Flutter không có thư viện. Một số thư viện hữu ích vẫn chưa được tích hợp vào, chỉ có những thư viện phổ biến là được Google quyết định thêm vào.

khai trên nhiều nền tảng khác. ● Widgets – Hỗ trợ cho việc xây dựng UI nhanh chóng, tiện lợi và ít tốn thời gian	
---	--

Flutter giúp việc phát triển ứng dụng cho đa nền tảng mà chỉ cần viết một mã nguồn duy nhất, tiết kiệm thời gian và công sức. Thứ hai, Flutter dễ dàng tích hợp với các dịch vụ bên ngoài, chẳng hạn như xác định vị trí qua Wi-Fi và tính toán đường đi.

Flutter cũng có hiệu suất tốt nhờ vào khả năng biên dịch trực tiếp thành mã máy, giúp ứng dụng hoạt động mượt mà. Giao diện của Flutter rất linh hoạt và đẹp, giúp người dùng dễ dàng sử dụng. Hơn nữa, với cộng đồng lớn và nhiều tài liệu hỗ trợ, việc phát triển ứng dụng trở nên dễ dàng hơn.

Cuối cùng, Flutter cho phép tích hợp các dịch vụ bản đồ như Google Maps, giúp hiển thị đường đi và các điểm quan tâm trong ứng dụng. Với tất cả những ưu điểm này, Flutter là một lựa chọn tuyệt vời cho ứng dụng chỉ đường sử dụng tín hiệu Wi-Fi.

1.3.3. FastAPI

Theo Tiangolo.(n.d.). FastAPI-tiangolo.FastAPI. FastAPI là một web framework hiện đại, hiệu suất cao giúp thiết kế web APIs với tiêu chuẩn Python type hints thông qua ngôn ngữ lập trình Python.

Một số tính năng chính của FastAPI bao gồm:

- Nhanh chóng: Hiệu suất cao so với NodeJS và Go là một framework Python nhanh nhất.
- Tốc độ phát triển: Code tính năng nhanh chóng.
- Hỗ trợ tốt hơn: Các trình soạn thảo hỗ trợ tốt hơn. Hạn chế thời gian gỡ lỗi.
- Tiện lợi: Giảm bớt thời gian đọc tài liệu, có thể học cách sử dụng nhanh chóng.

- **Tính năng:** Tính năng được tích hợp khi thiết lập tham số, hạn chế lỗi và trùng lặp code.
- **Tăng tốc:** Có được sản phẩm cùng với tài liệu (được tự động tạo) có thể tương tác.

Các tham số sau khi được khai báo là tham số của hàm sử dụng theo kiểu dữ liệu của Python, không phải sử dụng cú pháp mới, phương thức hay class của một thư viện nào mà chỉ sử dụng theo dữ liệu của chuẩn Python.

Được đánh là một trong những Python framework nhanh nhất, chỉ đứng sau Starlette và Uvicorn (được sử dụng trong FastAPI), Independent TechEmpower benchmarks chỉ ra rằng khi sử dụng FastAPI chạy dưới Uvicorn là hiệu quả nhất.

1.3.4. Google Colab

Google Colab là ứng dụng cho phép người dùng có thể viết và chia sẻ mã Python dễ dàng mà không cần cài đặt hay thiết lập cấu hình đẽm phức tạp trên máy tính. Hiểu đơn giản, thì là mọi thứ trên Colab đều có thể chạy và triển khai trên trình duyệt web, dữ liệu cũng sẽ được cập nhật tự động và lưu trữ trên nền tảng đám mây, đảm bảo sự thuận tiện và linh hoạt.

Về cơ bản, thì giao diện của Colab cũng khá giống với Jupyter Notebook, cho phép người dùng có thể thực thi các câu lệnh Python một cách trực quan và dễ dàng theo sát kết quả. Ngoài ra, Colab còn được tích hợp thêm khả năng tính toán “mạnh mẽ” trên tiện ích đám mây của Google, cùng với đó là kho dữ liệu và các thư viện học máy có sẵn, giúp tạo ra một môi trường lập trình linh hoạt và hỗ trợ tốt nhất cho trải nghiệm người dùng.

Một số tính năng chính của Google Colab bao gồm:

- **Tạo sổ tay ghi chép trực tuyến**

Google Colab cho phép người dùng có thể tạo và chạy các sổ tay ghi chép trực tuyến mà không cần phải cài đặt phần mềm. Cùng với đó là các dịch vụ sử dụng tài

nguyên tính toán miễn phí trên nền tảng đám mây, hỗ trợ người dùng làm việc hiệu quả mọi lúc, mọi nơi, và trên bất kỳ thiết bị nào có kết nối Internet.

- **GPU và TPU tăng tốc xử lý**

Hỗ trợ GPU và TPU tăng tốc “miễn phí” cũng là một điểm cộng của Colab. Theo đó, với phiên bản miễn phí, Colab cho phép người dùng có thể truy cập “free” vào thời gian chạy GPU và TPU trong tối đa 12h để tăng tốc xử lý dữ liệu và tối ưu thời gian tốt nhất.

- **Kết nối với Google Drive**

Với Colab, người dùng cũng có thể dễ dàng lưu trữ các tệp tài liệu trong quá trình lập trình lên Drive, và ngược lại, cũng rất thuận tiện khi tải tệp từ Drive lên Colaboratory. Điều này giúp việc lưu trữ và quản lý các tệp, dự án lập trình liên quan trở nên dễ dàng và linh hoạt hơn bao giờ hết.

Colab cung cấp các thư viện tài nguyên được cài đặt sẵn, chẳng hạn như NumPy, Matplotlib, Pandas, PyTorch, Keras, TensorFlow, để hỗ trợ người dùng hiệu quả trong việc viết mã và thử nghiệm. Không những vậy, ứng dụng còn cho phép người dùng cài đặt các thư viện không phải Colab như MySQL, GCP, SQL, AWS S3, nhằm tạo ra một môi trường thuận lợi và tối ưu nhất cho việc lập trình và nghiên cứu.

Ngoài thư viện tài nguyên học máy có sẵn, Colab còn cung cấp miễn phí cho người dùng tính năng truy cập đến GPU và TPU, giúp cho việc tăng tốc xử lý dữ liệu và huấn luyện mô hình trở nên hiệu quả và tối ưu thời gian hơn so với việc sử dụng CPU thông thường. Theo đó, đây có thể xem như giải pháp hoàn hảo cho các tác vụ tính phức tạp về mặt số học trong lĩnh vực học máy và deep learning.

1.3.5. MongoDB Atlas

MongoDB (Mongo Database) là một cơ sở dữ liệu opensource (mã nguồn mở) thuộc NoSQL hỗ trợ cho nhiều framework lập trình và giúp cho việc lập trình theo

hướng đối tượng. Những bảng dữ liệu trong Mongo Database (các Collection) thiết kế với mục đích là dữ liệu được sao lưu mà không phân phải theo định dạng cấu trúc nào.

Một số tính năng chính của MongoDB Atlas bao gồm:

Hỗ trợ lưu trữ các tệp dữ liệu

Mongo Database sử dụng hệ thống tệp thông qua việc lưu trữ tệp dữ liệu hỗ trợ việc sao chép dữ liệu.

Tại sao lựa chọn sử dụng công nghệ này:

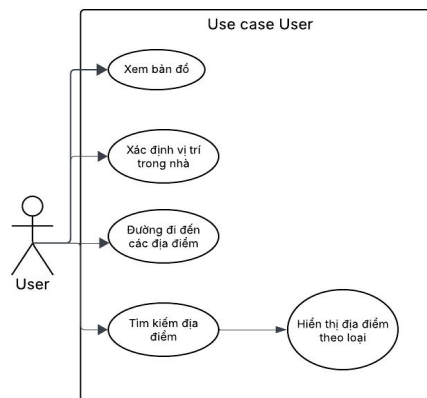
- Thông qua việc sử dụng dữ liệu dưới dạng JSON nên mỗi Collection có thiết kế kích thước và được lưu trữ trong nhiều document riêng biệt. Khi người dùng muốn thêm nhiều dữ liệu khác chỉ cần insert việc đó giúp cho quá trình lưu trữ linh hoạt hơn.
- Với mục đích NOSQL, dữ liệu không bị ràng buộc với nhau, không cần khóa chính, khóa phụ nên khi thêm, sửa, xóa dữ liệu sẽ đơn giản hơn so với việc kiểm tra mối quan hệ giữa các bảng.

CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1 Phân tích chức năng ứng dụng

Thông tin		
STT	Chức năng	Mô tả
1	Hiển thị các vị trí tại tầng 1 thư viện trên bản đồ	Hiển thị toàn bộ các địa điểm. Người dùng có thể xem các thông tin ngắn gọn, hình ảnh về các địa điểm,
2	Tìm kiếm các điểm theo loại	Lọc các loại của địa điểm làm nổi bật các địa điểm liên quan giúp người dùng dễ dàng tìm thấy trên bản đồ
3	Định vị vị trí hiện tại của người dùng	Hiển thị vị trí người dùng trên sơ đồ tầng 1 thư viện
4	Chỉ dẫn đường đi tới các điểm tại tầng 1 thư viện	Hiển thị chỉ đường người dùng đến vị trí được chọn
5	Thông tin địa điểm	Hiển thị chi tiết, hình ảnh thông tin địa điểm trong thư viện,

2.2. Sơ đồ Use-case



Hình 4. Sơ đồ Use-case

Tác nhân: Người dùng

Mô tả use case: Cho phép người dùng thực hiện được các chức năng.

Điều kiện trước: Không có

Dòng sự kiện chính:

Người dùng có thể thực hiện các chức năng sau trên website:

- Xem sơ đồ hiển thị danh sách các địa điểm tầng 1 thư viện trường Đại học Đà Lạt
- Xem thông tin các địa điểm
- Xem bài viết chi tiết về các địa điểm
- Định vị vị trí hiện tại của bản thân
- Đường đi đến các địa điểm
- Tìm kiếm địa điểm thông qua loại.
- Dẫn đường phát ra âm thanh

Mô tả:

Hệ thống ứng dụng chỉ đường gồm ba chức năng chính: Định vị vị trí, dẫn đường giữa người dùng và địa điểm được đánh dấu trên bản đồ, xem thông tin địa điểm.

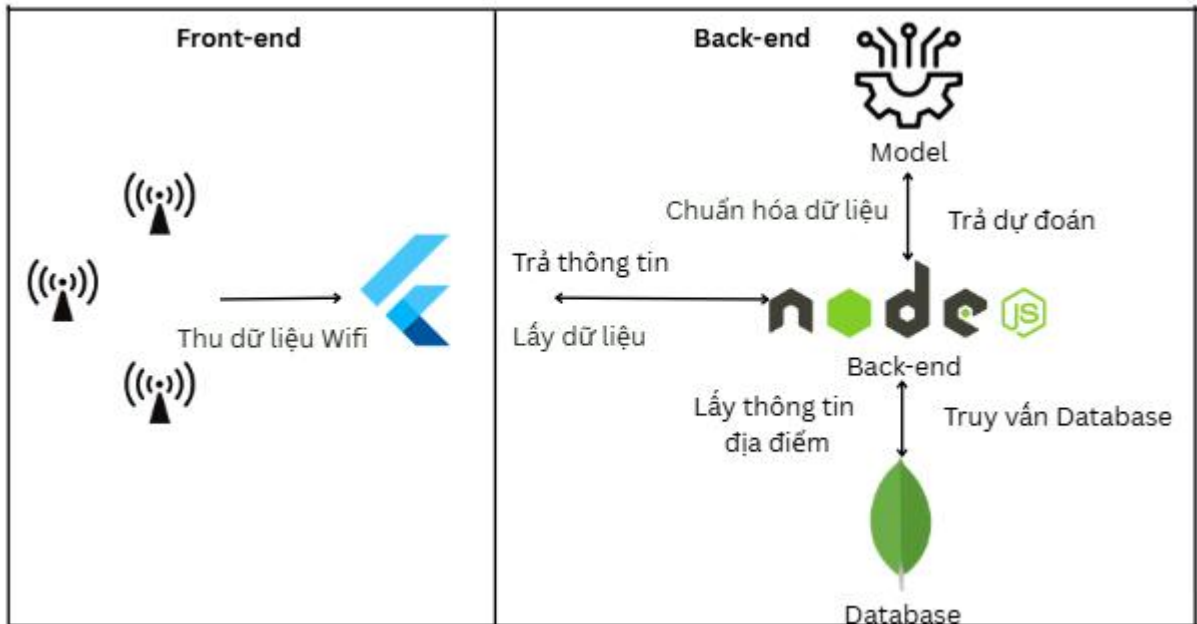
Định vị vị trí cung cấp chức năng giúp người dùng có thể định vị được hiện tại mình đang ở điểm nào trên bản đồ.

Dẫn đường đến địa điểm giúp cho người dùng có thể thấy được đường đi đến địa điểm mình mong muốn, ứng dụng sẽ phát ra âm thanh chỉ dẫn phương hướng đồng thời khoảng cách đến điểm đó.

Xem thông tin địa điểm khi người dùng chọn vào địa điểm đó, thông tin bao gồm tên địa điểm, mô tả địa điểm và hình ảnh của địa điểm đó.

2.3. Kiến trúc ứng dụng

Ứng dụng chỉ đường bằng tín hiệu wifi sử dụng mô hình Client-Server:



Hình 5. Kiến trúc hệ thống

Web server có nhiệm vụ kết nối với database và nơi chứa các phương thức xử lý với dữ liệu như lấy dữ liệu, nhận dữ liệu và trả dữ liệu với mô hình nhận diện.

Để giải quyết bài toán lưu trữ file hoàn toàn trên database, dự án đã quyết định sử dụng kết hợp MongoDB Atlas với Fast API. Trong đó, MongoDB Atlas là database chính, có nhiệm vụ lưu trữ các thông tin của đối tượng để chương trình có thể sử dụng và hiển thị trên giao diện. Các thông tin này bao gồm: tên file, thông tin đặc điểm của các địa điểm (RSSID, Name, Coordinare, ...). Web server chỉ được sử dụng để lưu trữ các file mà người dùng upload lên hệ thống như file mô hình, file giao thức, file chuyển hóa geojson thành package json.

Fast API cung cấp một tập hợp các phương thức và giao thức chuẩn để Client và Server giao tiếp. Client sử dụng phương thức API để gửi yêu cầu tới Server, ví dụ như yêu cầu lấy dữ liệu, tạo mới tài nguyên. Server sẽ nhận các yêu cầu này thông qua API, xử lý chúng và trả lại kết quả cho Client.

API được xây dựng theo kiến trúc RESTful API (Representational State Transfer) sử dụng giao thức HTTP để truyền tải dữ liệu và cho phép tương tác giữa Client và Server thông qua các phương thức chuẩn như GET, POST, PUT và DELETE. Dưới đây là một số điểm chi tiết về RESTful API:

Định dạng dữ liệu: RESTful API sử dụng các định dạng dữ liệu như JSON (JavaScript Object Notation) hoặc XML (eXtensible Markup Language) để truyền tải thông tin giữa Client và Server. JSON là định dạng phổ biến để sử dụng và đây là định dạng được sử dụng trong API này.

Trạng thái HTTP (HTTP Status): RESTful API sử dụng các mã trạng thái HTTP để chỉ định kết quả của một yêu cầu. Các mã phổ biến bao gồm 200 (OK), 201 (Created), 400 (Bad Request), 404 (Not Found), 500 (Internal Server Error), v.v. Mã trạng thái giúp Client hiểu kết quả của yêu cầu và thực hiện các hành động phù hợp.

Cách gọi API: RESTful API được gọi bằng cách sử dụng URL (Uniform Resource Locator) đại diện cho tài nguyên và sử dụng các phương thức HTTP. Ví dụ, để lấy danh sách người dùng, Client có thể gửi một yêu cầu GET tới URL: /user, để lấy danh sách di sản, Client có thể gửi một yêu cầu GET tới URL: /heritage và kết quả trả về sẽ có dạng JSON.

CHƯƠNG 3. PHƯƠNG PHÁP ĐỀ XUẤT

3.1. Hệ thống định vị thông qua Wifi

Hệ thống định vị thông qua tín hiệu WiFi là một phần quan trọng trong việc tự động hóa quá trình định vị người dùng tại khu vực thư viện tầng 1. Dưới đây là phần báo cáo mô tả cụ thể về cách hoạt động của hệ thống:

Hệ thống được thiết kế để tự động nhận dữ liệu RSS thu được từ thiết bị trong thực tế và trả kết quả đầu ra là vị trí dự đoán được dưới dạng file Json để Frontend hiển thị lên trên bản đồ. Quá trình này bao gồm các bước sau:

Hình 6. Hệ thống định vị thông qua WiFi

Bước 1: Dữ liệu RSS dưới dạng file Json sau khi được Backend giải mã sẽ được hệ thống Backend lưu lại dưới dạng mảng.

Bước 2: Dữ liệu RSS sau khi được đưa về dạng mảng sẽ được hệ thống xem số lượng phần tử trong mảng có đáp ứng đủ số lượng để làm dữ liệu đầu vào cho mô hình không, sau đó chuẩn hóa mảng về dưới dạng chuẩn hóa [0 1] nhằm đảm bảo dữ liệu đầu vào phù hợp với mô hình phân lớp.

Bước 3: Mô hình phân lớp sau khi nhận được mảng dữ liệu đầu vào sẽ dựa vào RSS, hệ thống sẽ so khớp với các vector đặc trưng và trả về cho backend ID RP địa điểm dự đoán được.

Bước 4: Sau khi mô hình dự đoán được địa điểm, server sẽ tham chiếu ID RP với cơ sở dữ liệu được lưu trong MongoDB Atlas và trả về cho Frontend thông tin địa điểm với file Json bao gồm các thuộc tính như ID RP, tên của RP, vị trí tọa độ của RP.

3.2. Mô tả tập dữ liệu

Tại tầng 1 của thư viện trường Đại học Đà Lạt số lượng AP có thể phát hiện được tại 1 vị trí có thể từ 3AP trở lên. Để đảm bảo chất lượng của quá trình định vị, hiện tượng đa lường tăng khi số lượng và mật độ AP tăng. Trong [7] tác giả Ngô Văn

Bình trích lục từ bài báo của [9] Vadih Moghtadaiee và các cộng sự đã thử nghiệm trên bộ dữ liệu với số lượng AP khác nhau và kết quả cho thấy chỉ cần 5 RSS thu được từ 5 AP đã đủ cho việc định vị.

Từ kết quả này, cho thấy không phải tất cả các RSS thu được từ các AP đều cần thiết cho việc định vị. Vì vậy, trong [5-6] nhiều nhóm nghiên cứu đã đề xuất các giải pháp chọn AP dựa trên giá trị của RSS, nhằm tăng chất lượng định vị. Tuy nhiên, tác động của hiệu ứng đa lường và suy giảm tín hiệu có thể làm cho giá trị RSS của cùng một AP ở cùng một vị trí tại các thời điểm khác nhau không giống nhau.

Dựa trên các nghiên cứu trên chúng tôi thực hiện việc thu thập dữ liệu RSS ở các thời điểm tại các vị trí khác nhau trong thời điểm khác nhau bao gồm khi đông người, khi vắng người. Các giá trị RSSI được thu thập ở 40 vị trí khác nhau tại cùng một tầng tại tòa nhà thư viện, khoảng cách trung bình giữa hai vị trí tham chiếu lân cận là 7 mét. Dữ liệu thực hiện 80 lần đo theo 4 hướng 0^0 , 90^0 , 180^0 và 270^0 , tương ứng 20 mẫu tại một hướng tại mỗi vị trí vào các thời điểm khác nhau trong ngày. Tổng số AP xuất hiện trong tập dữ liệu thu thập được là 3 đến 40 AP. Như vậy, tập dữ liệu bao gồm 3200 vector đại diện cho 3200 phép đo sử dụng cho cả quá trình huấn luyện ngoại tuyến và thử nghiệm trực tuyến. Khi thực hiện mô phỏng, tập dữ liệu training cho giai đoạn offline tương ứng với 3200 mẫu và tập dữ liệu testing cho giai đoạn online tương ứng với 160 mẫu được dành cho giai đoạn thử nghiệm. Dữ liệu được thu trên 4 thiết bị android khác nhau, mỗi chu kỳ để có thể cập nhật được dữ liệu mới là 30 giây.

Mỗi mẫu gồm 13 thông tin, giá trị RSS nằm trong khoảng từ -15dB đến -96dB, giá trị RSS=-100 được đặt mặc định với các AP không phát hiện được khi lấy mẫu. Đặc trưng còn lại là các nhãn gồm thời gian lấy dữ liệu, số lượt lấy dữ liệu, thứ tự WiFi quét được, tổng số lượng WiFi thu thập được, Wifi SSID, Wifi BSSID, Wifi RSSI (dBm), kinh độ (Longitude), vĩ độ (Latitude), mã người thu thập dữ liệu (ID), mã điện thoại thu thập dữ liệu (PhoneID),...

Bộ dữ liệu mẫu thu được tại tòa nhà thư viện tóm gọn như sau:

Time	2025:01:24:07:46:22
WiFi Fingerprinting serial	1
WiFi AP serial number	1,... 20
Total number of AP scanned	34
WiFi SSID	DLU-Teacher
WiFi BSSID	88:dc:**:**:cf
WiFi RSSI (dBm)	-57
GPS Longitude (°)	108.4452251
GPS Latitude (°)	11.957152
Phone Model	SM-S908E
Student ID	2212343
Reference Point ID	RP01

Bảng 1. Thống kê cột dữ liệu mẫu

3.2. Tiền xử lý dữ liệu

Dựa theo [7], dữ liệu cần tiền xử lý dữ liệu như sau:

Thứ nhất, số lượng các AP nhận được tại một RP nhỏ nhất là 5, lớn nhất là 40 và trung bình là xấp xỉ 15, trong khi bài toán định vị trong nhà chỉ cần xấp xỉ 3 AP là có thể xác định được vị trí và theo thống kê thì số RP có AP nhỏ hơn 6 (bao gồm cả các RP không có AP nào) là 0, do đó dự án này quyết định loại bỏ các RP có nhỏ hơn , điều này hầu như không ảnh hưởng đến chất lượng định nhưng lại tăng tốc độ tính toán.

Thứ hai, tại một RP có thể nhận được giá trị RSS từ AP của các thiết bị khác, các AP này không có tác dụng cho định vị thậm chí làm giảm chất lượng, do đó tệp dữ liệu cũng loại bỏ các AP này.

Trong [2] nhóm tác giả Ngô Bình Văn, sử dụng bộ dữ liệu UJIIndoorLoc là một tập dữ liệu lớn nên tác giả tiến hành xử lý dữ liệu nhằm giảm thời gian tính toán và tăng chất lượng. Tác giả chỉ ra rằng bài toán phân lớp giá trị giữa các điểm càng chênh lệch lớn thì càng dễ phân lớp, chẳng hạn như tòa nhà được phân lớp tốt hơn nếu các đặc trưng có nhiều thông tin của tòa nhà. Tiến hành việc loại bỏ các đặc trưng có sự khác biệt thấp, trùng nhau. Tiền xử lý dữ liệu được áp dụng cho pha kiểm thử của mô hình phân lớp.

Trước khi đưa vào huấn luyện, dữ liệu RSS được tiền xử lý bằng cách loại bỏ các AP có số lần xuất hiện thấp trong các lần thu thập dữ liệu, ví dụ như giá trị RSS thu được từ các thiết bị di động đóng vai trò như Hotspot tạm thời. Đồng thời, dữ liệu RSS được chuẩn hóa về khoảng $[0, 1]$ bằng phương pháp Min-Max Scaling nhằm làm tăng kết quả định vị như thể hiện trong Bảng 2 khi so sánh kết quả định vị của một số thuật toán khi thực hiện chuẩn hóa và không chuẩn hóa dữ liệu.

Độ chính xác (%)	Random Forest	WKNN	KNN
Không chuẩn hóa	88.76	73.91	78.52
Chuẩn hóa	97.16	88.70	89.50

Bảng 2. So sánh kết quả định vị trước và sau chuẩn hóa

Sau khi loại bỏ các đặc trưng có độ chênh lệch thấp bằng phương pháp đã áp dụng, tiến hành gộp các mảng lại để tạo thành tập dữ liệu đầu vào có 37 đặc trưng.

ID_RP	X	Y	WiFi BSSID 1	...	WiFi BSSID 34
1	-16	0	-57	...	-50
1	-16	0	-78	...	-46
...	...	..	...	...	...

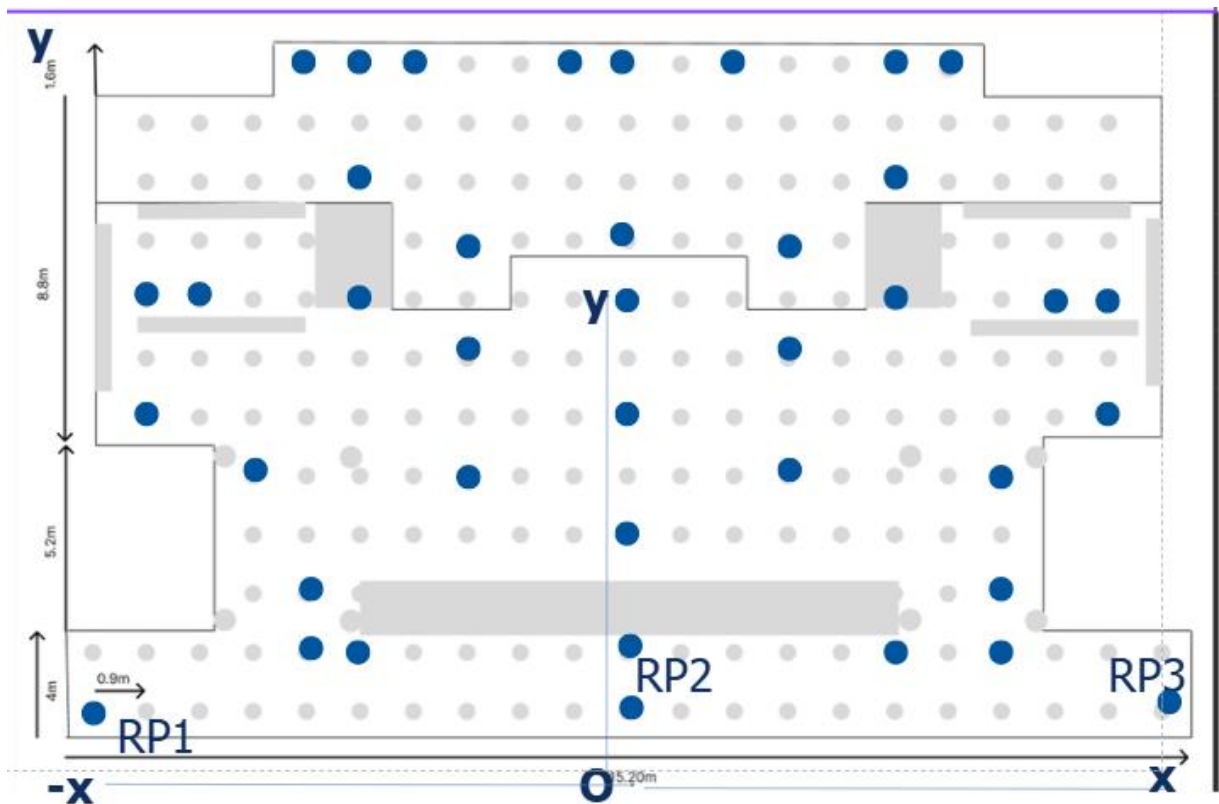
40	29	52	-98	...	-58
40	29	52	-65	...	-73

Bảng 3. Tập dữ liệu đầu vào có 37 đặc trưng

Công đoạn này giúp việc học và dự đoán của mô hình hiệu quả hơn đáng kể. Nhóm cũng đã tiến hành thực hiện huấn luyện và đánh giá thực nghiệm điều này, kết quả thu được là độ chính xác của việc tiền xử lý dữ liệu RSS có chuẩn hóa cao hơn khi chưa chuẩn hóa.

3.3. Bản đồ định vị và chỉ số quy đổi

Mặt bằng thực nghiệm tạo thành một lưới cách đều như hình.



Hình 7. Sơ đồ chỉ số quy đổi

Tất cả các thử nghiệm được nhóm thực hiện trên thiết bị là Samsung Galaxy Tab S6lite. Do tính chất của điện thoại có màn hình độ phân giải khác nhau như sau: ,nên nhóm thiết kế ảnh bản đồ dựa trên bản đồ thực tế, mục đích là:

- Dễ dàng thay đổi bản đồ chỉ đường
- Hỗ trợ hiển thị nhiều loại thiết bị có màn hình khác nhau nếu mở rộng.

Sau khi đo đạc và chia tỉ lệ bản đồ theo hệ trục tọa độ cục bộ (X,Y), giá trị dùng để quy đổi theo:

Vị trí thu thập (RP)	Ma trận (X,Y)	Vị trí thu thập (RP)	Ma trận (X,Y)
01-03	(-16, 0), (0.0, 0.0), (16, 0.0)	20-21	(-22, 34), (22, 34)
04-07	(-30, 10),(-8,10),(8,10), (30, 0)	22-25	(-43, 35),(-36, 35), (36, 35), (43, 35)
08-09	(-30, 14), (30, 14)	26-27	(3.0, 4.5); (7.0, 4.5)
10-11	(-15, 18), (0, 18)	28	(0, 41)
12-15	(-26, 22), (-13, 22), (13, 22), (26, 22)	29-32	(-22, 45), (-13, 45), (13, 45), (22, 45)
16-17	(-42, 24), (42, 27)	33-40	(-30, 52), (-23, 52), (-15, 52),(0, 52), (9,52),(14, 52), (22,52), (29, 52)
18-19	(-13, 33), (00, 33)		

Bảng 4. Thống kê điểm thu thập dữ liệu theo ma trận

Dựa trên tỉ lệ quy đổi, sai lệch giữa vị trí dự đoán và vị trí thực tế sẽ được tính bằng đơn vị mét (m). Cụ thể, gọi (X train), (Y train) là tọa độ vị trí thực tế, (X predict) và (Y predict) là tọa độ vị trí được định vị bởi hệ thống. Sai lệch vị trí định vị được tính bằng

m từ tọa độ vị trí [(X train), (Y train)] đến [(X predict), (Y predict)] theo [3] công thức tính lỗi tuyệt đối trung bình MAE (Mean Absolute Error) :

$$MAE = \frac{1}{S} \sum_{i=1}^S |\hat{Y}_{predict} - \hat{Y}_{train}| \quad [3]$$

3.4. Tiến hành huấn luyện mô hình

3.4.1. Thuật toán định vị tất định

Trong nghiên cứu này, chúng tôi thực hiện việc thử nghiệm hai thuật toán tất định (deterministic) được sử dụng rộng rãi trong việc định vị là KNN và WKNN. Ý tưởng của thuật toán KNN là chọn ra K điểm tham chiếu RP có khoảng cách Euclidean giữa vec-tơ RSS thu thập tại vị trí cần tìm và các vec-tơ RSS trong bộ dữ liệu ngoại tuyến là nhỏ nhất. Vị trí điểm cần tìm là tọa độ trung bình tính từ tọa độ của K điểm RP được chọn. Khác với KNN, WKNN tính thêm trọng số cho các điểm RP thông qua khoảng cách Euclidean. Ý tưởng là một điểm RP có khoảng cách RSS giữa nó và điểm cần tìm càng nhỏ thì có trọng số càng lớn, tức là RP đó được xem như nằm gần vị trí cần tìm hơn các RP khác.

Giá trị K sẽ được chọn từ phương pháp thử-sai nhằm xác định thuật toán và siêu tham số K tối ưu nhất. Thử nghiệm cho thấy WKNN cho kết quả sai số định vị thấp hơn và ổn định hơn KNN. Do đó, phương pháp WKNN được chọn làm nền tảng cho hệ thống định vị, sau kết hợp với mô hình học sâu trong phần tiếp theo nhằm cải thiện độ chính xác tổng thể.

Đặt K là số lượng láng giềng gần nhất (số RP) cần xét trong thuật toán WKNN, A_train là ma trận dữ liệu RSS huấn luyện có kích thước (số mẫu huấn luyện, số đặc trưng), A_test là ma trận dữ liệu RSS kiểm tra có kích thước (số mẫu kiểm tra, số đặc trưng), y_train là nhãn tương ứng (ID của AP) với các mẫu huấn luyện, và num_classes là tổng số lớp cần phân loại (ID của RP). Đầu ra (output) preds_prob là ma trận xác suất dự đoán có kích thước (số mẫu kiểm tra, số lớp). Mỗi hàng là một vec-tơ xác suất phân bố trên tất cả các lớp, biểu thị khả năng mỗi mẫu kiểm tra thuộc về từng lớp tương ứng. Thuật toán KNN và WKNN được mô tả chi tiết như dưới đây.

Thuật toán KNN: Tạo ra các vec-tơ theo xác suất

```

1: function GET_KNN_PREDICTIONS(K, A_train, A_test, y_train, num_classes)
2:   preds_class ← ∅
3:   for each RSS_test ∈ A_test do
4:     distances ← Euclidean distance between
       RSS_test and all samples in A_train

```

```

5: nearest_idx ← indices of K smallest distances
6: classes ← y_train[nearest_idx]
7: class_count ← zero vector of length num_classes
8: for each label in classes do
9:     class_count[label] ← class_count[label] + 1
10: end for
11: predicted_class ← argmax(class_count)
12: Append predicted_class to preds_class
13: end for
14: return preds_class
15: end function

```

Thuật toán WKNN: Tạo ra các vec-tơ theo xác suất

```

1: function GET_WKNN_PROBABILITIES(K, A_train, A_test, y_train,
num_classes)
2: c ← 0.01 //small constant to avoid division by zero
3: preds_prob ← ∅
4: for each RSS_test ∈ A_test do
5:     distances ← Euclidean distance between
        RSS_test and all samples in A_train
6:     nearest_idx ← indices of K smallest distances
7:     weights ← 1 / (distances[nearest_idx] + c)
8:     classes ← y_train[nearest_idx]
9:     class_weight_sum ← zero vector of length
        num_classes
10:    for each (label, w) in (classes, weights) do
11:        class_weight_sum[label] ←
            class_weight_sum[label] + w
12:    end for
13:    class_prob ← class_weight_sum /
        sum(class_weight_sum)
14:    Append class_prob to preds_prob
15: end for
16: return preds_prob
17: end function

```

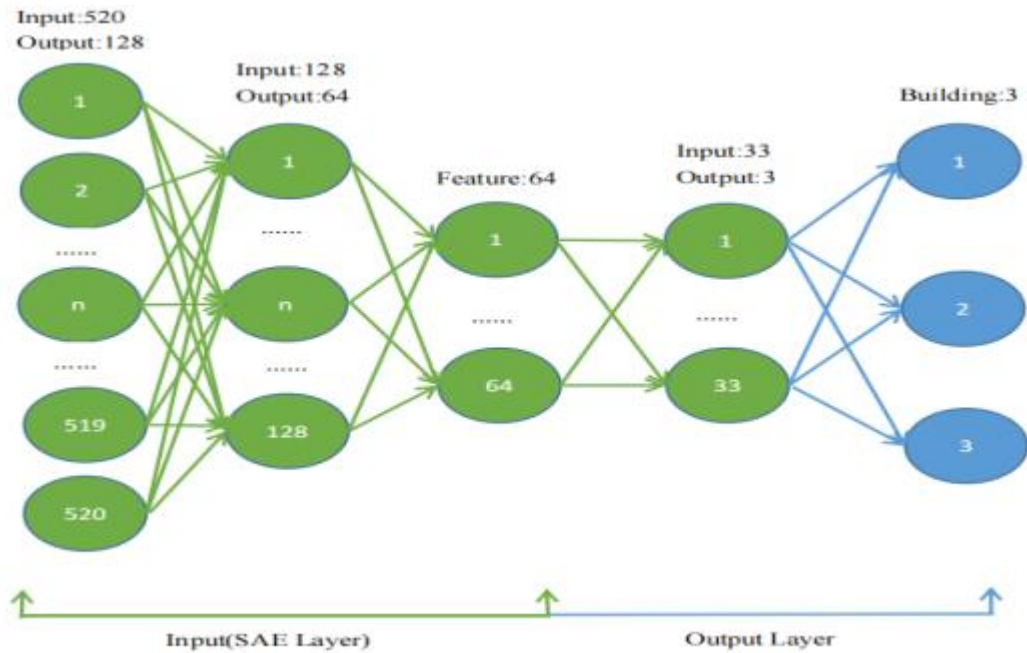
3.4.2. Mô hình học sâu

Sau khi chuẩn bị đầy đủ tệp dữ liệu đầu vào, ta tiến hành quá trình thực hiện các phương pháp định vị trong nhà sử dụng kỹ thuật Wifi Fingerprinting: KNN, WKNN, Random Forest và mô hình phân lớp CNN.

KNN	K = 3	K = 5	K = 7
WKNN	K = 3	K = 5	K = 7
Random Forest	n_estimators=100, random_state = 42	n_estimators=200, random_state = 42	n_estimators=500, random_state = 42

Bảng 5. Thống kê tham số của thuật toán áp dụng

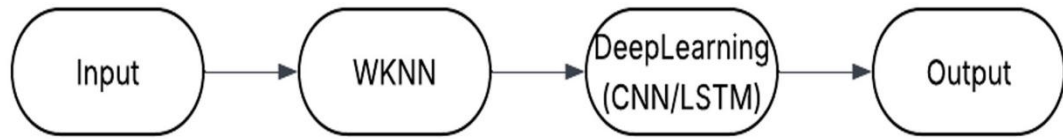
Dựa theo [7] J. -W. Jang, and S. -N. Hong mô hình phân lớp được xây dựng mới mạng nơon quen thuộc bao gồm các lớp mã hóa tự động, lớp fully connected hidden hay còn gọi là lớp dense và cuối cùng là lớp phân lớp được thể hiện qua hình sau:



Hình 8. Sơ đồ mô hình theo bài báo [6]

Trong bài báo này, chúng tôi đề xuất và so sánh các mô hình học sâu khác nhau để so sánh hiệu năng định vị của từng mô hình, từ đó chọn ra mô hình đạt độ chính xác định vị cao nhất khi thực nghiệm trên cùng một bộ dữ liệu WiFi Fingerprinting. Mô hình cơ bản được xây dựng kết hợp giữa kết quả định vị từ thuật toán định vị tất định phổ biến và hiệu quả WKNN và mạng thần kinh tích chập một chiều CNN hoặc mạng bộ nhớ dài ngắn hạn LSTM (Long Short-Term Memory) nhằm mục tiêu tính toán vị trí

của người dùng dựa trên dữ liệu RSS thu thập từ các điểm tham chiếu RP, thể hiện tại Hình 1.



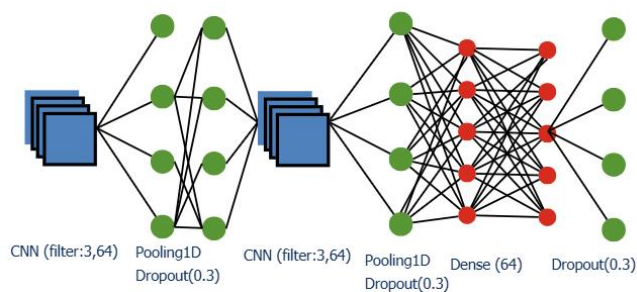
Hình 9. Sơ đồ hệ thống chung của mô hình

Cấu trúc của mô hình bao gồm các lớp chính như sau:

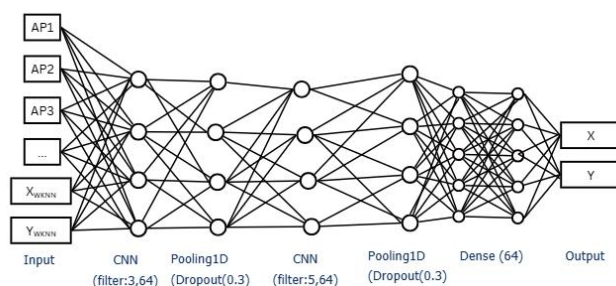
- Lớp Conv1D với 64 bộ lọc có nhiệm vụ trích xuất các đặc trưng cục bộ từ chuỗi tín hiệu RSSI đầu vào.
- Lớp Pooling1D có chức năng giảm chiều dữ liệu và giữ lại những đặc trưng quan trọng nhất.
- Lớp Dense với 64 nơ-ron được sử dụng để kết nối và tổng hợp các đặc trưng đã học được từ các lớp trước.
- Để tăng cường khả năng khái quát hóa của mô hình và hạn chế hiện tượng quá khớp (overfitting), hai lớp Dropout được thêm vào: một lớp với tỷ lệ dropout là 0.3 được chèn sau khối CNN/LSTM, hoặc một lớp khác với tỷ lệ dropout là 0.5 được thêm sau khối CNN/LSTM.
- Lớp đầu ra của mô hình là một lớp Dense gồm 2 nơ-ron, tương ứng với tọa độ (x, y), đại diện cho vị trí cần tìm của đối tượng trong không gian hai chiều.

Nhằm tìm ra mô hình mô hình hai giai đoạn có độ chính xác định vị cao nhất, bài báo xây dựng các mô hình học sâu và học sâu kết hợp khác nhau, bao gồm: mô hình phân lớp CNN cơ bản và mô hình hai giai đoạn WKNN-CNN, WKNN-LSTM, WKNN-LSTM Regression thể hiện trong hình 10 đến 13.

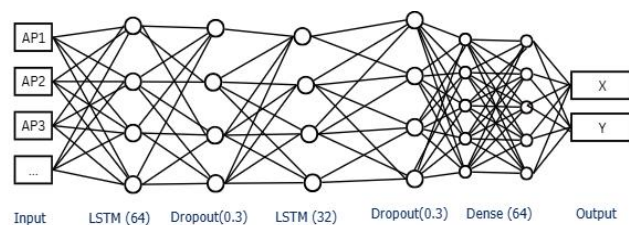
Thông qua mô hình phân lớp trên dự án xây dựng mô hình cơ bản của bài toán định vị trong nhà bằng fingerprinting dùng sóng Wifi dựa trên học máy được thể hiện trong [1]



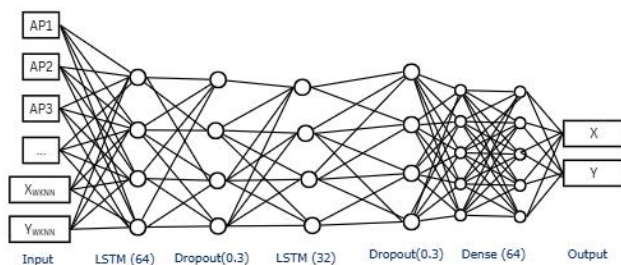
Hình 10. Sơ đồ mô hình xây dựng



Hình 11. Sơ đồ mô hình WKNN - CNN



Hình 12. Sơ đồ mô hình WKNN - LSTM



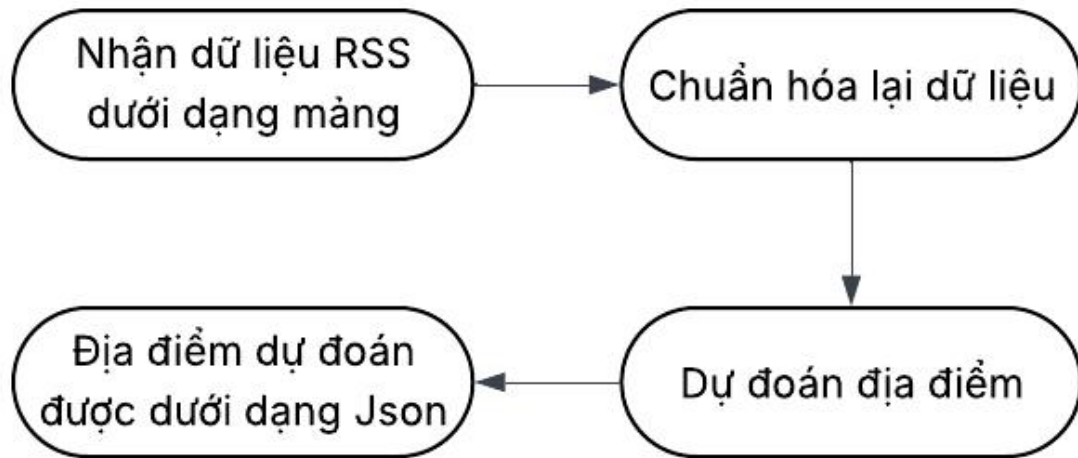
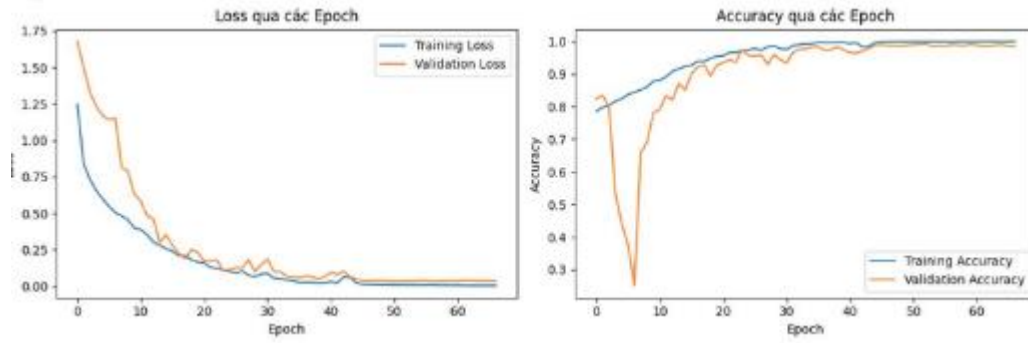
Hình 13. Sơ đồ mô hình WKNN - LSTM Regression

3.5. Kết quả đánh giá và so sánh các mô hình

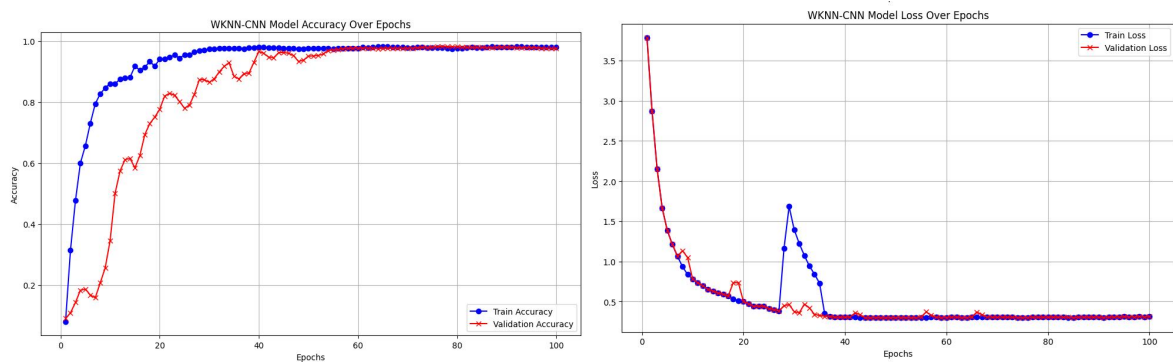
Trong phần này, bài báo cung cấp các kết quả thực nghiệm thể hiện độ chính xác định vị của các mô hình học sâu được đề xuất. Các mô hình cũng này được so sánh với các

thuật toán định vị phổ biến khác. Cụ thể, bài báo so sánh kết quả định vị khi sử dụng hai thuật toán tất định phổ biến là KNN và WKNN, $K=3$. Ngoài ra, thuật toán Random Forest cũng được cài đặt do thuật toán này đã được chứng minh là đạt độ chính xác cao trong một nghiên cứu trước đây [9]. Kết quả định vị của các phương pháp thể hiện trong Hình 10. Kết quả cho thấy hai thuật toán tất định cho kết quả định vị chính xác thấp nhất. Tuy nhiên, điểm mạnh của các thuật toán tất định là đơn giản và dễ cài đặt, có khả năng áp dụng dễ dàng trong các ứng dụng thực tế. Kết quả phân lớp của WKNN-CNN là cao nhất với khả năng phân lớp chính xác tới 98.75%.

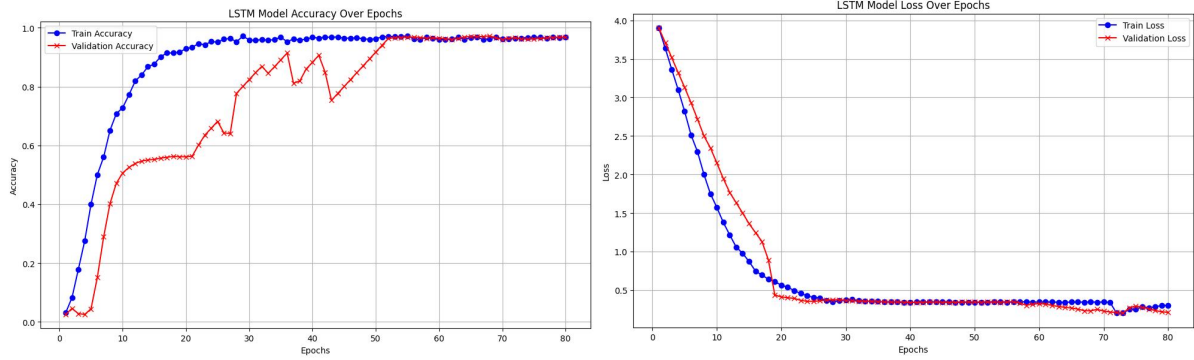
Tiếp theo, để đánh giá hiệu quả của mô hình trong việc dự đoán lỗi khoảng cách giữa vị trí thực và vị trí ước tính, bài báo sử dụng công thức tính lỗi tuyệt đối trung bình MAE (Mean Absolute Error). Kết quả sai số định vị MAE của các mô hình được thể hiện trong Hình 11. Mô hình học sâu hai pha được đề xuất WKNN-CNN cho kết quả sai số thấp nhất với chỉ 2.8 m. Trong khi đó, KNN và WKNN cho kết quả sai số cao nhất với sai số đều từ 8 m trở lên. Các sai số định vị tăng dần tiếp theo được sắp xếp theo thứ tự từ WKNN-LSTM, WKNN-LSTM Regression, CNN, và Random Forest, với sai số lần lượt là 3.02 m, 4.02 m, 5 m, và 6.02 m.



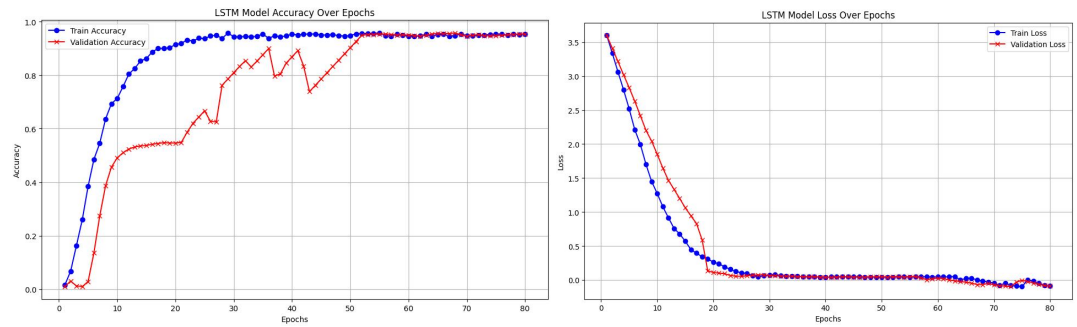
Hình 14. Kết quả mô hình phân lớp



Hình 15. Kết quả mô hình phân lớp WKNN - CNN

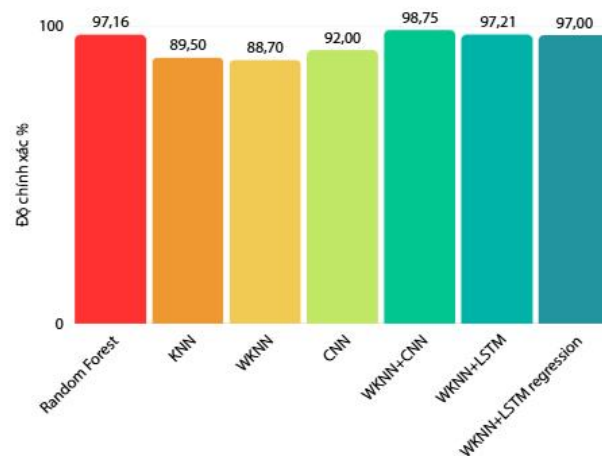


Hình 16. Kết quả mô hình phân lớp WKNN - LSTM



Hình 17. Kết quả mô hình phân lớp WKNN – LSTM Regression

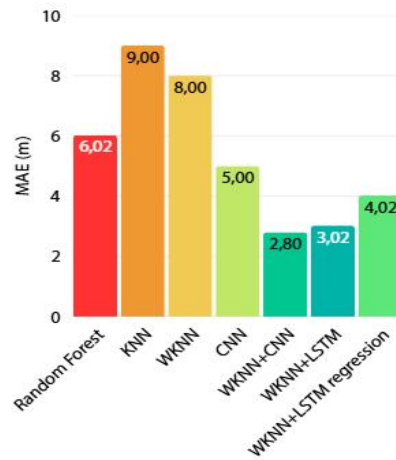
Các kết quả của các thuật toán độc lập được tổng hợp trong bảng.



Hình 18. Biểu đồ thể hiện độ chính xác của phương pháp và mô hình

Để đánh giá hiệu quả của mô hình trong việc dự đoán lỗi khoảng cách giữa vị trí thực và vị trí ước lượng, dự án sẽ sử dụng công thức tính lỗi tuyệt đối trung bình MAE (Mean Absolute Error)

Dựa vào [1-8], dự án thực hiện tiến hành giai đoạn Online phase, model CNN, tuy nhiên trong quá trình thử nghiệm thực tế, trên thiết bị Galaxy Tab S6 lite, model có sự nhầm lẫn với khoảng cách MSE là từ 0 đến 8m tại thời điểm đông người tại khu vực tầng 1 và có sự giao động của nhiều RSSI. Các phương pháp đạt được kết quả phân lớp tốt các điểm RP với MSE như sau:

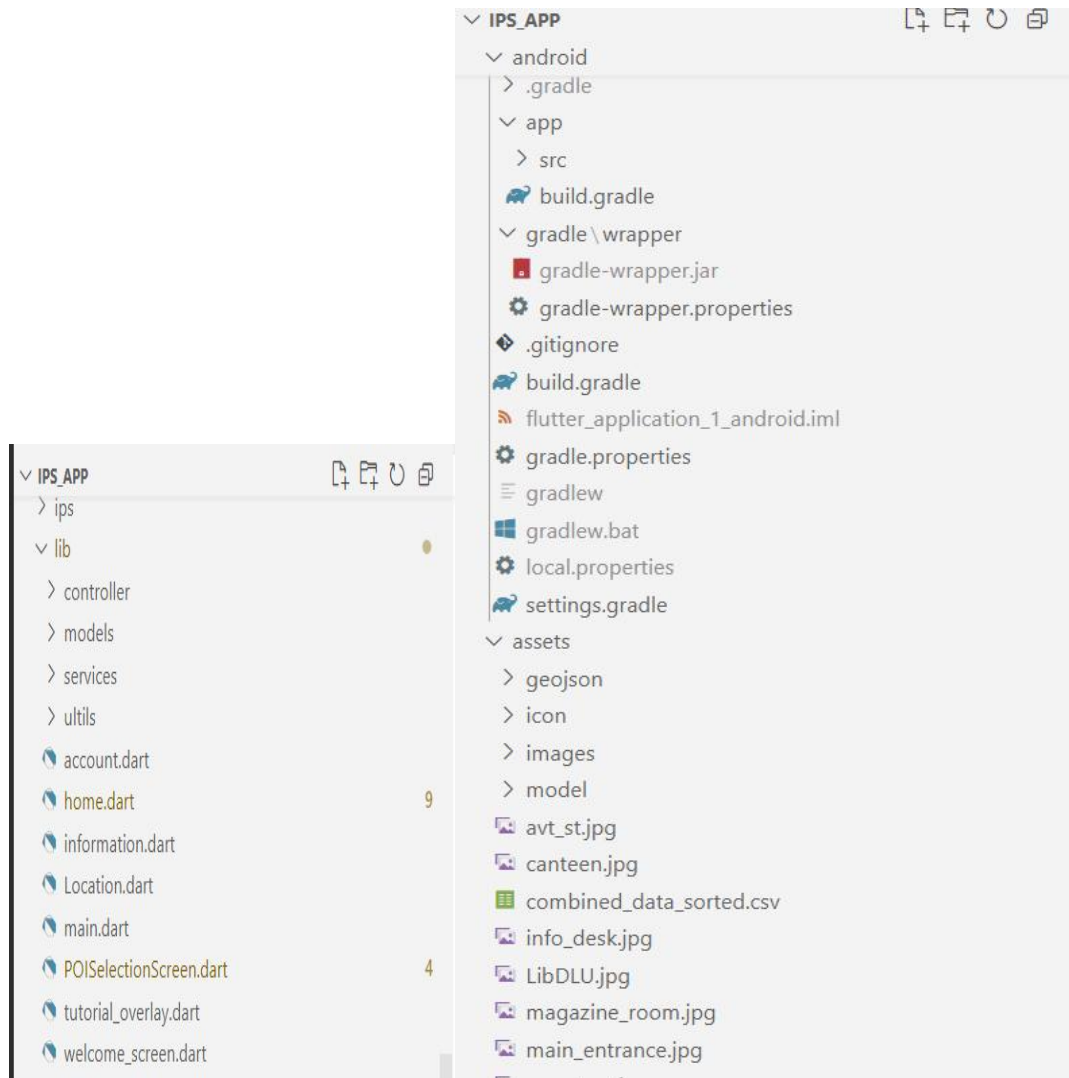


Hình 19. Thống kê độ sai khoảng cách trong thực tế

CHƯƠNG 4: XÂY DỰNG ỨNG DỤNG

4.1. Cấu trúc thư mục

4.1.1. Frontend



Hình 20. Cấu trúc thư mục FrontEnd

Cấu trúc chương trình có thư mục như sau:

.Android: Đây là thư mục tùy chọn, được sử dụng để lưu trữ cấu hình liên quan đến trình biên dịch và công cụ hỗ trợ cho môi trường phát triển Visual Studio Code. Thư mục này có thể chứa các tệp tin cấu hình như build.gradle hoặc các package liên quan đến Flutter.

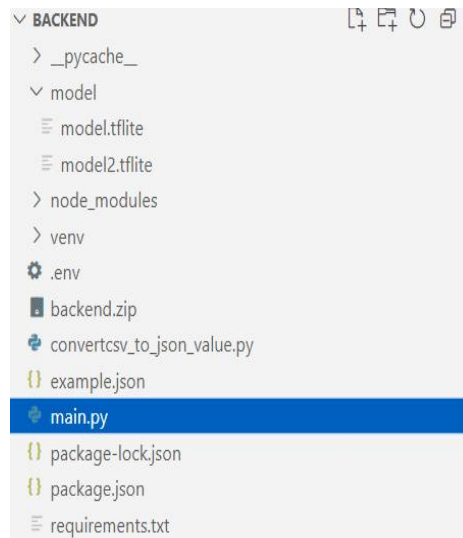
.assets: Thư mục “assets” chứa các tệp tin tĩnh không thay đổi của ứng dụng như hình ảnh, tệp tin khác được truy cập trực tiếp từ ứng dụng.

.lib: Thư mục “lib” chứa mã nguồn chính của ứng dụng Flutter. Đây là nơi chúng ta sẽ làm việc chủ yếu để xây dựng ứng dụng của chúng ta. Thư mục “lib” thường có cấu trúc tương tự như sau:

Các page chính:

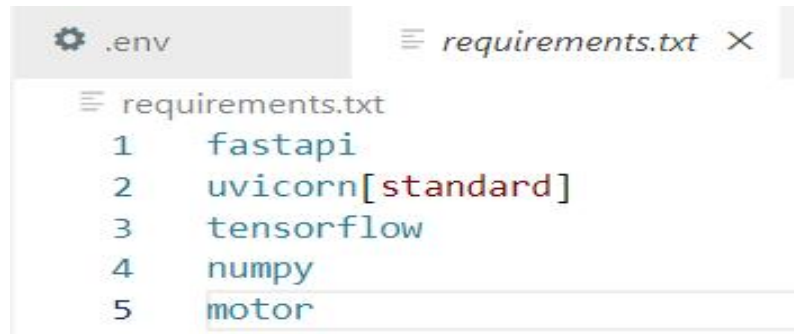
- controller: Thư mục “controller” chứa các tệp tin để xử lý các yêu cầu quyền cấp quyền và kiểm tra quyền
- models: Thư mục “models” chứa các tệp tin liên quan đến danh mục, thông tin chung của người dùng, xử lý dữ liệu và logic liên quan đến bản đồ.
- utils: Thư mục “utils” chứa các tệp tin liên quan đến các hàm tiện ích quyền truy cập, quét wifi, thông tin liên quan đến người dùng.
- services: Thư mục “services” chứa các tệp tin liên quan đến các dịch vụ ngoại vi như API, xử lý giao tiếp với máy chủ, hoặc xử lý dữ liệu từ các nguồn bên ngoài.

4.2.2 Backend



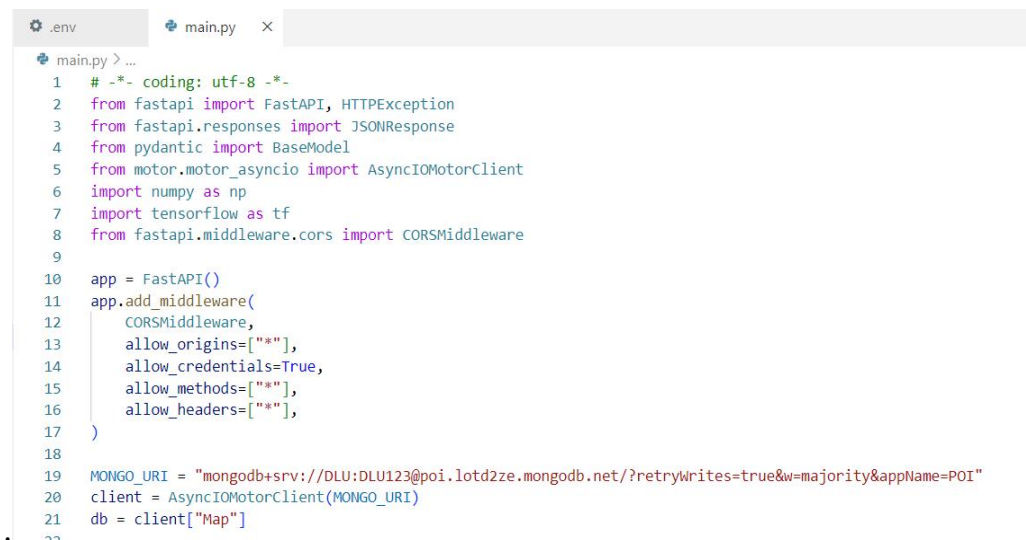
Hình 21. Cấu trúc thư mục BackEnd

Thư mục model chứa các mô hình được huấn luyện.



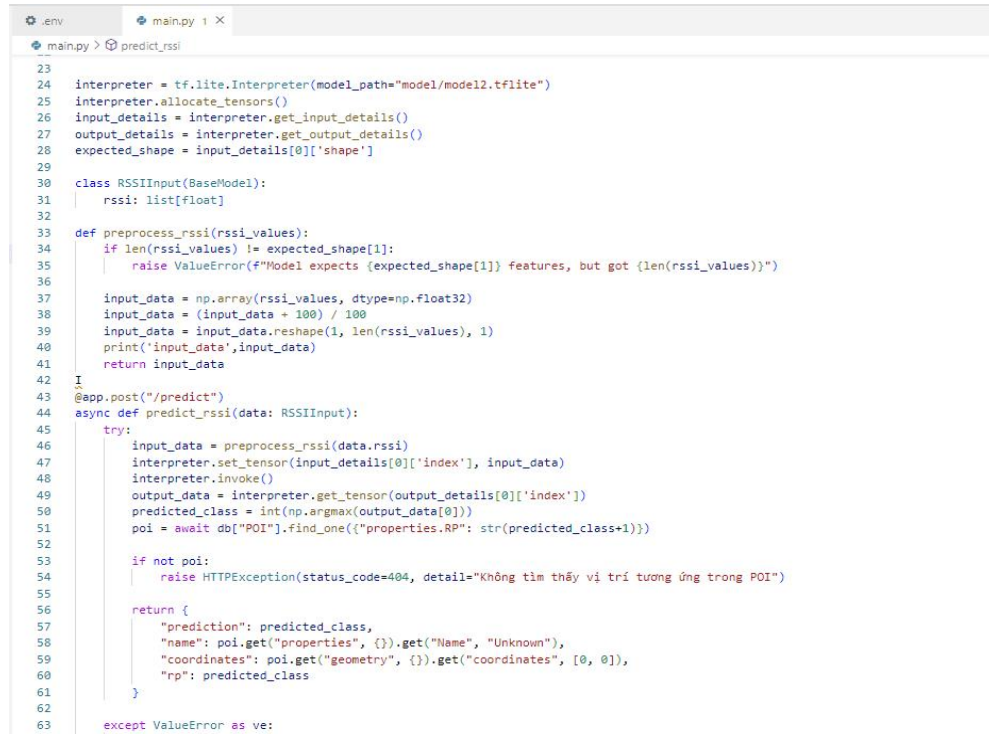
Hình 22. Tập tin requirement.txt

Tập tin requirement.txt: chứa các thư viện mà dự án cần cài đặt để hoạt động



Hình 23. Mã giả tập tin main.py

Tập tin main.py: chứa các các đoạn mã liên quan đến cấu hình, khởi tạo, route và khởi chạy server. Dưới đây là đoạn mã liên quan đến việc load model để phân lớp và chuẩn hóa lại dữ liệu đầu vào trước khi đưa vào model dự đoán điểm RP.



```

23
24 interpreter = tf.lite.Interpreter(model_path="model/model2.tflite")
25 interpreter.allocate_tensors()
26 input_details = interpreter.get_input_details()
27 output_details = interpreter.get_output_details()
28 expected_shape = input_details[0]['shape']
29
30 class RSSIInput(BaseModel):
31     rssi: list[float]
32
33 def preprocess_rssi(rssi_values):
34     if len(rssi_values) != expected_shape[1]:
35         raise ValueError(f"Model expects {expected_shape[1]} features, but got {len(rssi_values)}")
36
37     input_data = np.array(rssi_values, dtype=np.float32)
38     input_data = (input_data + 100) / 100
39     input_data = input_data.reshape(1, len(rssi_values), 1)
40     print("input_data", input_data)
41     return input_data
42
43 @app.post("/predict")
44 async def predict_rssi(data: RSSIInput):
45     try:
46         input_data = preprocess_rssi(data.rssi)
47         interpreter.set_tensor(input_details[0]['index'], input_data)
48         interpreter.invoke()
49         output_data = interpreter.get_tensor(output_details[0]['index'])
50         predicted_class = int(np.argmax(output_data[0]))
51         poi = await db["POI"].find_one({"properties.RP": str(predicted_class+1)})
52
53         if not poi:
54             raise HTTPException(status_code=404, detail="Không tìm thấy vị trí tương ứng trong POI")
55
56         return {
57             "prediction": predicted_class,
58             "name": poi.get("properties", {}).get("Name", "Unknown"),
59             "coordinates": poi.get("geometry", {}).get("coordinates", [0, 0]),
60             "rp": predicted_class
61         }
62     except ValueError as ve:
63

```

Hình 24. Mã giả xử lý giao tiếp BackEnd & Model

Dữ liệu RSSI trước khi được đưa vào model sẽ được định dạng lại thành dạng mảng dưới dạng file Json. Tiếp theo dữ liệu sẽ được chuẩn hóa lại về dạng số thập phân, xử lý dữ liệu về dạng chuẩn hóa [0 1], Min-Max Scaling để đảm bảo được độ nhất quán của dữ liệu đầu vào.

4.2. Xây dựng API

Hệ thống cung cấp các endpoint (điểm cuối) để truy cập vào dữ liệu và chức năng của nó. Điều này cho phép cung cấp các dịch vụ thông qua giao thức HTTP và nhận dữ liệu thông qua JSON. Bảng mô tả chi tiết các API Endpoint (xem phụ lục)

default			^
Method	Endpoint	Description	
POST	/predict	Predict Rssi	✓
GET	/geojson/POI	Get Poi Geojson	✓
GET	/geojson/Doors	Get Doors Geojson	✓
GET	/geojson/Hallways	Get Hallways Geojson	✓
GET	/geojson/Paths	Get Paths Geojson	✓
GET	/geojson/Room	Get Room Geojson	✓
GET	/geojson/Stair	Get Stair Geojson	✓

Hình 25. Thống kê API của dự án

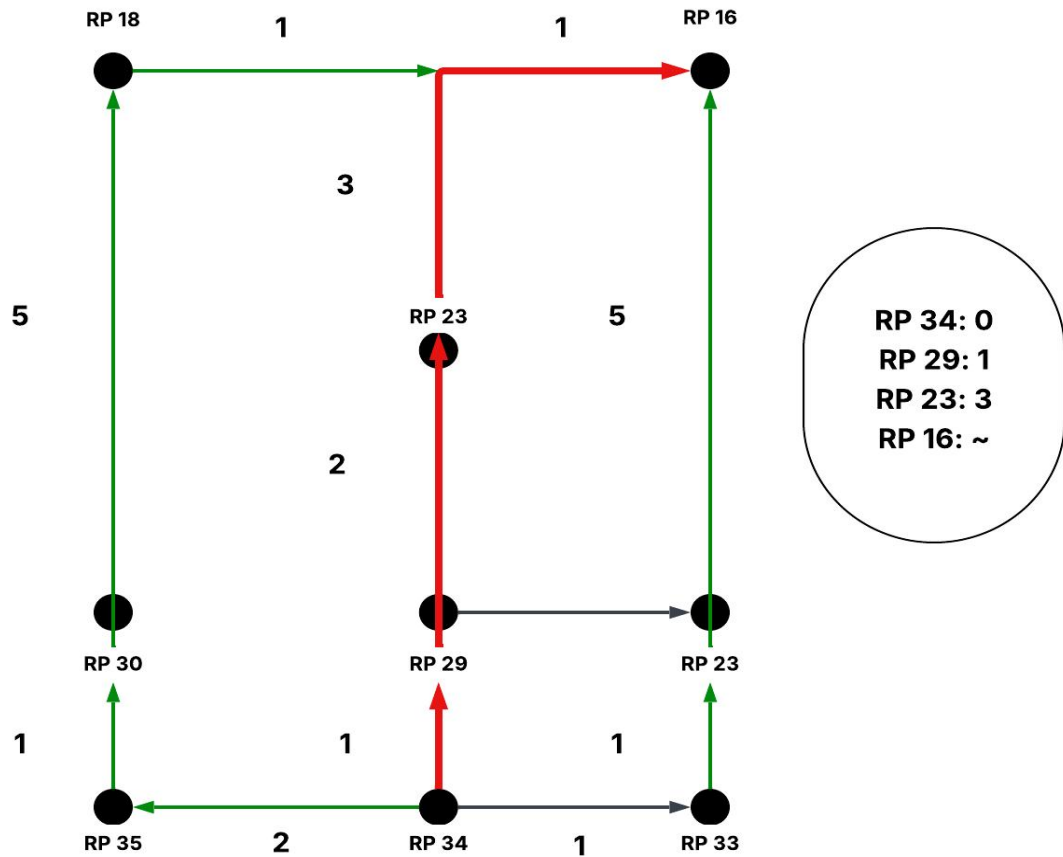
4.3. Lựa chọn thuật toán dẫn đường

4.3.1 Thuật toán Dijkstra

Thuật toán Dijkstra có tác dụng chính là thay thế việc tìm đường đến đích ngắn nhất, ví dụ thực tế có thể kể đến là Google Map của Mỹ hay Baidu Map của Trung Quốc cũng sử dụng thuật toán Dijkstra.

Thuật toán Dijkstra giải quyết bài toán đường đến đích ngắn nhất đồng thời cũng đơn giản hóa đường đi bằng cách xét các cạnh và so sánh 2 đường đi sẵn có với đường qua cả 3 đỉnh.

Quá trình tính toán như sau: Cho tất cả các đỉnh và cạnh đều là đường đi ngắn nhất. Qua việc tính toán so sánh, từ đó rút ra được đường đi nào là tối ưu nhất thông qua việc chọn đỉnh, các bước được thực hiện cho đến khi tất cả đỉnh đều được chọn.



Hình 26. Thuật toán Dijkstra

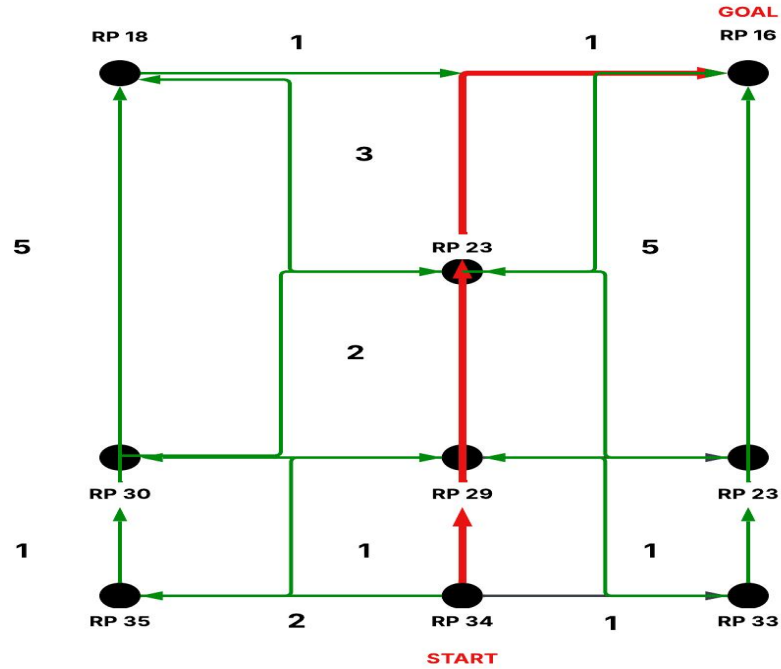
4.3.2 Thuật toán A*

Thuật toán A* là một trong những công cụ mạnh mẽ và phổ biến nhất để giải quyết các bài toán tối ưu hóa và tìm kiếm. Được phát triển vào đầu những năm 1960, thuật toán A* đã chứng minh khả năng vượt trội trong việc tìm kiếm đường đi tối ưu trong các bài toán đồ thị.

Thuật toán A* bắt đầu bằng việc đưa điểm bắt đầu vào danh sách mở (open list). Mỗi điểm trong danh sách này sẽ chứa các thông tin về chi phí $g(n)$, hàm heuristic $h(n)$ và tổng chi phí $f(n)$.

Điều khiến thuật toán A* trở nên hiệu quả và mạnh mẽ chính là việc sử dụng hàm heuristic $h(n)$. Nếu hàm này được thiết kế hợp lý, thuật toán A* sẽ có thể tìm ra

đường đi tối ưu trong thời gian rất ngắn. Đặc biệt, nếu hàm heuristic là “thực tế (admissible)”, tức là không bao giờ ước lượng chi phí quá cao so với giá trị thực tế, thuật toán A* sẽ luôn tìm ra được giải pháp tối ưu.



Hình 27. Thuật toán A*

4.3.3 Thuật toán A*

Nhóm thực hiện so sánh và đánh giá hai thuật toán dẫn đường được sử dụng rộng rãi trong việc xây dựng ứng dụng là thuật toán Dijkstra và A*. Đề tài quyết định thuật toán A* vì A* sử dụng hàm heuristic để ước lượng khoảng cách còn lại đến đích, từ đó giúp việc mở rộng các node, tập trung vào tìm đường tới điểm đến tạo ra đường đi trực quan khi biểu diễn. Quãng đường đi được thể hiện ở hình 22 như sau:



Hình 22. Khoảng cách và chỉ dẫn đường đi

4.4. Kết quả đạt được

Dưới đây là một số hình ảnh về các chức năng của ứng dụng chỉ đường tại khu vực thư viện tầng 1 trường Đại học Đà Lạt.

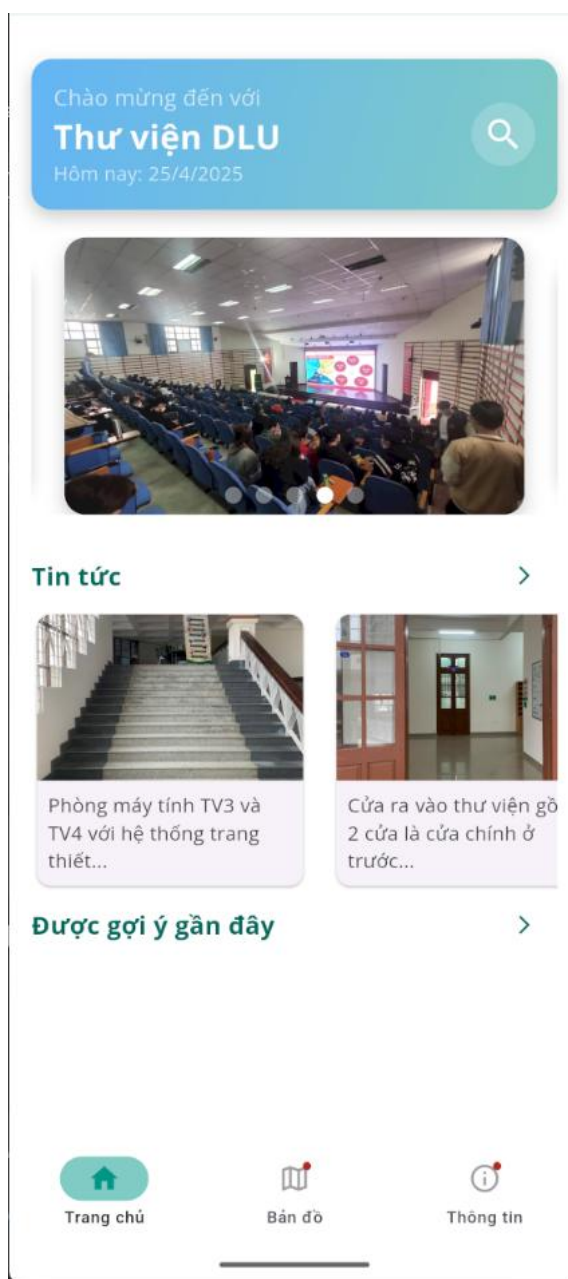
4.4.1. Welcome Screen

Phần giới thiệu Thư viện Trường Đại học Đà Lạt



Hình 28. Welcome Page

4.4.2. Giao diện chính



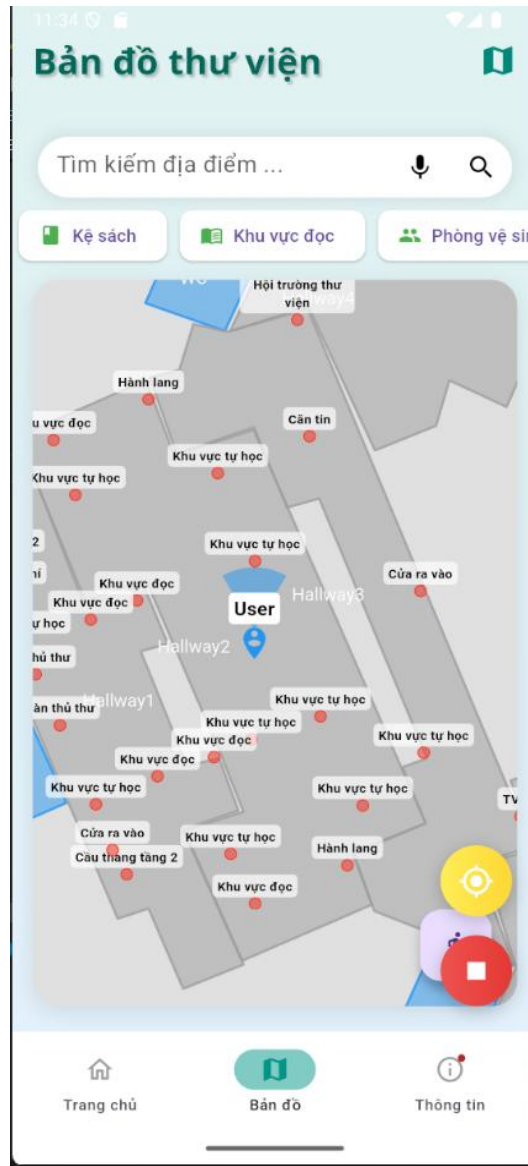
Hình 29. Giao diện chính

Thanh BottomNavbar hiển thị các trang mà người dùng có thể truy cập. Khi bấm vào, ứng dụng sẽ chuyển hướng người dùng đến các trang tương ứng.

Ở góc trên cùng của header, người dùng có thể chuyển hướng đến trang giúp người dùng có thể chọn điểm bắt đầu và chỉ đường đến các điểm mình mong muốn. Bên dưới

là các nút khi người dùng chọn sẽ đánh dấu nổi bật các điểm có loại địa điểm mong muốn.

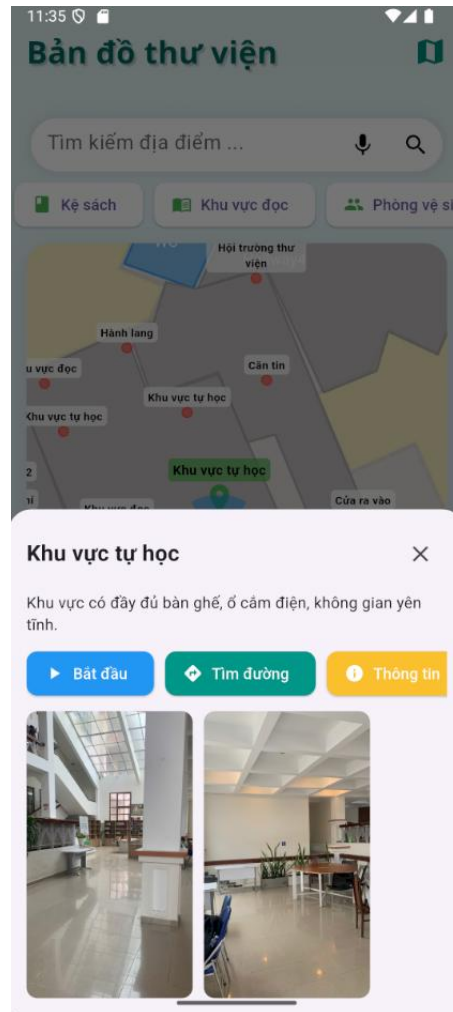
4.4.3. Giao diện Map



Hình 30. Giao diện Map

Giao diện chính của ứng dụng bao gồm việc hiển thị thanh tìm kiếm địa điểm, các nút thống kê loại địa điểm và giao diện bản đồ sơ đồ tầng 1 thư viện.

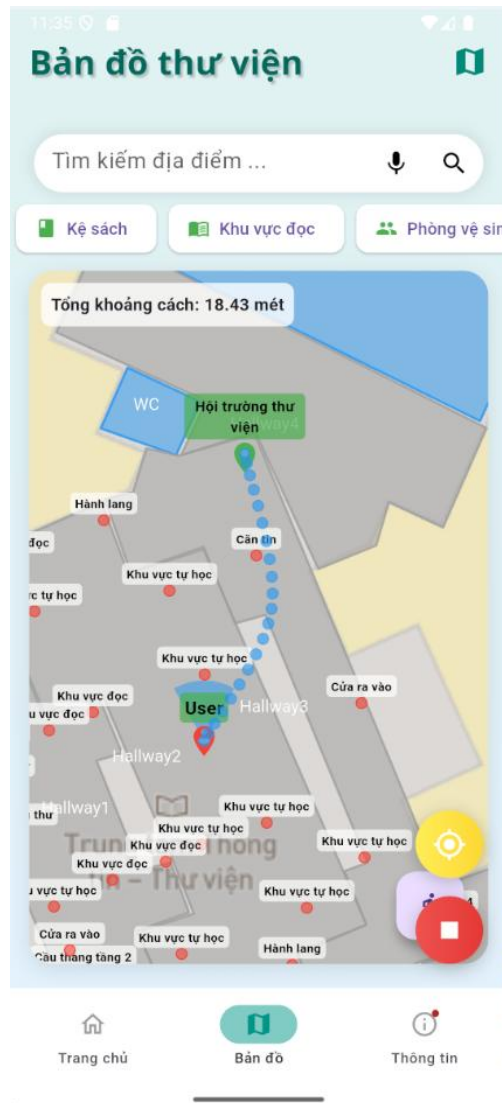
4.4.4. Thông tin địa điểm



Hình 31. Thông tin POI khi tương tác trên Map

Khi người dùng chọn vào một địa điểm trên bản đồ sẽ hiển thị Popup thông tin địa điểm, bao gồm tên địa điểm, mô tả địa điểm và các nút chức năng “Bắt đầu”, “Tìm đường”, “Thông tin”.

4.4.5. Chỉ đường

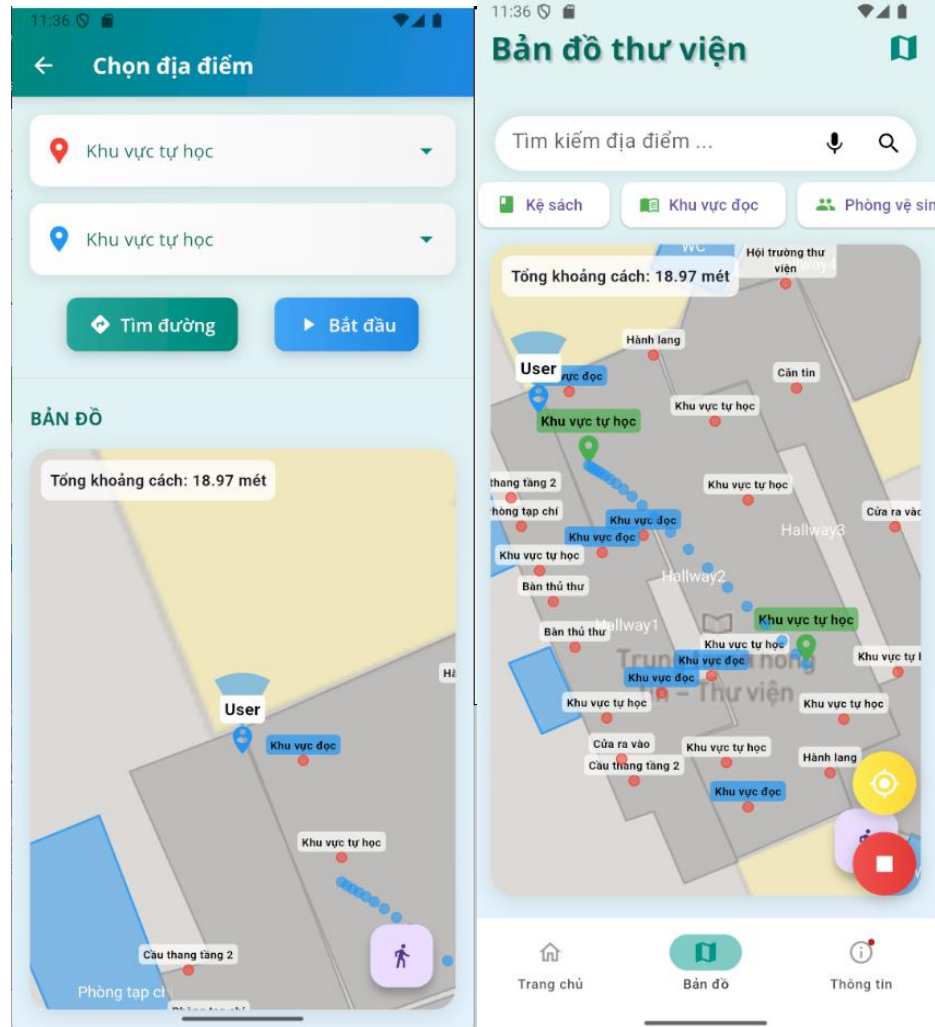


Hình 32. Hình ảnh chức năng nút "Chỉ đường"

Khi người dùng chọn nút “Chỉ đường” hệ thống sẽ chỉ đường người dùng từ vị trí bản thân đến địa điểm mong muốn được thể hiện bằng các làm nổi bật “User” bằng icon màu đỏ và điểm được chọn bằng màu xanh với đường đi bằng các chấm tròn màu xanh.

Với nút “Bắt đầu” hệ thống vẫn sẽ chỉ đường cho người dùng như “Chỉ đường” tuy nhiên hệ thống sẽ phát ra âm thanh chỉ dẫn hướng, khoảng cách đến điểm được lựa chọn.

4.4.6. Tìm đường đi tự chọn



Hình 34. Hình ảnh chức năng "Tìm kiếm địa điểm"

Người dùng khi nhấn vào “Tìm kiếm địa điểm” ở phần Header có thể lựa chọn địa điểm mình bắt đầu và kết thúc theo mong muốn bằng cách dùng DropDown, hệ thống sẽ trả về đường đi từ điểm bắt đầu và kết thúc ở trang chủ.

Hệ thống sẽ hiển thị hai địa điểm bằng cách làm nổi bật địa điểm bằng màu xanh lá cây và đường đi như hình trên.

4.4.7. Giới thiệu địa điểm

Người dùng khi nhấn vào nút “Thông tin” sẽ được dẫn đến bài viết giới thiệu về thông tin của địa điểm



Hình 35. Hình ảnh trang thông tin

KẾT LUẬN

Đề tài đã tìm hiểu và sử dụng thêm những kỹ thuật và phương pháp định vị sử dụng WiFi cho website như kỹ thuật WiFi Fingerprinting, sử dụng FastAPI để lập trình backend qua cách thức hoạt động của mô hình client-server để xây dựng một ứng dụng chỉ đường có các chức năng chính như sau:

Người dùng có thể xem được các địa điểm có trong tầng 1 thư viện trường Đại học Đà Lạt, đường đi đến các địa điểm có trong tầng, phân loại các địa điểm theo loại, xem thông tin địa điểm của các địa điểm.

Tóm lại, ứng dụng chỉ đường này cũng đóng vai trò quan trọng trong việc giúp người dùng có thể tìm kiếm các địa điểm, dễ dàng phổ biến cấu trúc thư viện. Dự án áp dụng được các kỹ thuật định vị trong nhà sử dụng kỹ thuật WiFi fingerprinting đồng thời đó tạo được một bộ dữ liệu 3200 mẫu về 40 địa điểm trong tầng 1 tòa nhà thư viện trường Đại học Đà Lạt, mô hình CNN đạt kết quả là 92% với độ lệch chuẩn MSE là trung bình từ 0m đến 8m khi vào giai đoạn kiểm thử trong thực tế.

Ứng dụng đã thực hiện được các chức năng chính bao gồm như định vị người dùng, chỉ dẫn đường đi tuy nhiên vẫn gặp một số hạn chế trong việc load bản đồ hiển thị vẫn còn chậm, một số điểm chưa được định vị chính xác vì sự ảnh hưởng với RSS trong quá trình định vị tại một số vị trí trung tâm của thư viện như RP 20,RP11,..

Ứng dụng chỉ đường vẫn còn một số điểm hạn chế cần phải khắc phục và khó khăn gặp phải như tài liệu hướng dẫn không chi tiết điều này buộc nhóm phải tự tìm hiểu và học hỏi từ kinh nghiệm từ các thầy, cô. Vì vậy, nên nếu có thêm thời gian em sẽ hướng đến các mục tiêu sau để hoàn thiện ứng dụng của mình như thêm trang web quản lý thông tin địa điểm cho admin, đăng nhập tài khoản đối với phân hệ admin, xây dựng bản đồ 3D tăng tốc độ load của ứng dụng thay vì dùng bản đồ thực tế từ nguồn khác, phát triển chức năng theo dõi vị trí người dùng. khi di chuyển.

LỜI CẢM ƠN

Bài báo được hỗ trợ bởi Đề tài nghiên cứu khoa học sinh viên 2025 (Quyết định số 64/QĐ-ĐHĐL), Đề tài nghiên cứu khoa học cấp trường trọng điểm năm 2024 của trường Đại học Đà Lạt (Quyết định số 1518/QĐ-ĐHĐL), và Đề tài nghiên cứu khoa học cấp Bộ Giáo Dục năm 2024 (Mã số B2024-DLA-04).

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] A, NESSA, et al. "'A Survey of Machine Learning for Indoor Positioning,.'" *IEEE Access*, vol. 8, ed., no. pp. 214945 - 214965, 2020.
- [2] Bình, Ngô Văn, and Hiệu Văn Vũ. "Một kỹ thuật định vị trong nhà bằng WiFi hiệu quả sử dụng học máy kết hợp." *Journal on Information Technologies & Communications*, 11 2022, DOI:10.32913/mic-ict-research-vn.v2022.n2.1124.
- [3] Dương, Hằng Thị, et al. "Giải pháp định vị trong nhà độ chính xác cao sử dụng thuật toán kNN và LSTM." *Journal of Military Science and Technology*, no. 86:48-55, 4 2023, DOI:10.54939/1859-1043.j.mst.86.2023.48-55.
- [4] D. Foad, A. Ghifari, M. B. Kusuma, et al., "A systematic literature review of A* pathfinding," *Procedia Computer Science*, vol. 179, pp. 507-514, 2021.
- [5] Tiangolo. (n.d.). *FastAPI - tiangolo*. FastAPI. <https://fastapi.tiangolo.com>
- [6] H. Lu, L. Zhang, H. Chen, et al., "An Indoor Fingerprint Positioning Algorithm Based on WKNN and Improved XGBoost," *Sensors*, vol 23, no. 8, p. 3952, 2023. <https://doi.org/10.3390/s23083952>.
- [7] Y. Wang, L. Cheng, H. Yuan, and X. Li, "A wifi fingerprint indoor location framework based on convolutional neural network," In proceeding of 2023 International Conference on Intelligent Communication and Networking (ICN), pp. 34-37, 2023, doi: 10.1109/ICUFN.2018.8436598.
- [8] N. T. Lương, D. B. Ninh, P. Q. Huy and N. H. Khánh, "WiFi Fingerprinting-based Indoor Positioning with Machine Learning Algorithms," *The 2022 Information and Communication Technology (ICT) Conference*, pp. 67-71, 2022.
- [9] V. B. Ngô, "Một kỹ thuật định vị trong nhà bằng WiFi hiệu quả sử dụng học máy kết hợp," *Journal on Information Technologies & Communications*, vol 11, pp. 1-8, 2022.
- [10] N. Duong-Bao, J. He, T. Vu-Thanh, L. N. Thi, L. Do Thi, and K. Nguyen-Huu, "A Multi-Condition WiFi Fingerprinting Dataset for Indoor Positioning," In proceeding of International Conference on Artificial Intelligence and Big Data in Digital Era, pp. 601-613, 2021.

- [11] Xuesheng Peng, et al. “An Improved Weighted K-Nearest Neighbor Algorithm for Indoor Localization.” *Electronics*, no. 9(12), 11 12 2020, <https://doi.org/10.3390/electronics9122117>.
- [12] Vahideh Moghtadaiee and Andrew Graham Dempster. “Design protocol and performance analysis of indoor fingerprinting positioning systems”. In: *Phys. Commun.* 13 (2014), pp. 17–30.