

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



Lương Minh Tuấn

**NGHIÊN CỨU XÂY DỰNG ỨNG DỤNG HỖ TRỢ KHÁCH DU
LỊCH TẠI VIỆT NAM**

Chuyên ngành : KHOA HỌC MÁY TÍNH

Mã số : 8.48.01.01

TÓM TẮT LUẬN VĂN THẠC SĨ KỸ THUẬT

HÀ NỘI – NĂM 2020

Luận văn được hoàn thành tại:
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: TS. NGUYỄN ĐÌNH HÓA

Phản biện 1:

.....

Phản biện 2:

.....

Luận văn sẽ được bảo vệ trước Hội đồng chấm luận văn thạc sĩ tại Học viện
Công nghệ Bưu chính Viễn thông

Vào lúc: giờ ngày tháng năm 2020

Có thể tìm hiểu luận văn tại:

- Thư viện của Học viện Công nghệ Bưu chính Viễn thông

PHẦN MỞ ĐẦU

Trong những năm qua, du lịch Việt Nam đang trên đà phát triển, lượng khách quốc tế đến cũng như khách du lịch nội địa ngày càng tăng. Du lịch Việt Nam ngày càng được biết đến nhiều hơn trên thế giới, nhiều điểm đến trong nước được bình chọn là địa chỉ yêu thích của du khách quốc tế. Du lịch đang ngày càng nhận được sự quan tâm của toàn xã hội. Chất lượng và tính cạnh tranh của du lịch là những vấn đề nhận được nhiều sự chú ý và thảo luận rộng rãi. Một cách tiếp cận đa chiều về đánh giá chất lượng du lịch sẽ góp phần hình thành các giải pháp đúng đắn nâng cao chất lượng và tính cạnh tranh của du lịch Việt Nam.

Ứng dụng công nghệ thông tin trong ngành Du lịch là một trong những giải pháp đột phá để nâng cao tính cạnh tranh của du lịch Việt Nam, từ đó tạo ra nền tảng du lịch thông minh trong thời đại cách mạng công nghiệp 4.0[4]. Trong những năm gần đây điện thoại di động thông minh (smartphone) đã được sử dụng phổ biến và trở thành phương tiện giao tiếp và giải trí của đông đảo người dùng nhờ giá smartphone ngày càng rẻ, cung cấp nhiều tính năng phong phú và dễ sử dụng. Nhằm đáp ứng nhu cầu du lịch của du khách nước ngoài tại Việt Nam, tác giả đã đề xuất ra ý tưởng nghiên cứu xây dựng một ứng dụng giúp di động cho du khách nước ngoài du lịch tại Việt Nam có thể có được trải nghiệm tốt nhất, hiểu thêm về văn hóa, lịch sử cũng như con người Việt Nam.

Để xây dựng nên ứng dụng có thể hỗ trợ cho người du lịch và đem lại những trải nghiệm tốt nhất thì tác giả cần thiết kế giao diện cho ứng dụng phù hợp; xây dựng được lượng dữ liệu bao gồm các địa danh; thiết lập, cài đặt các thuật toán sao cho phù hợp nhất với bài toán, giúp cho khách du lịch có những trải nghiệm tốt nhất.

Hiện nay, trên thế giới đã có nhiều nghiên cứu về các ứng dụng phục vụ cho du lịch. Các ứng dụng thường thường gặp là:

TripIt là ứng dụng sẽ cung cấp cho bạn những thông tin cơ bản về nơi bạn sắp đến. Bên cạnh đó là những thông tin về các chuyến bay, khách sạn... giúp bạn lên kế hoạch tổ chức và sắp xếp cho chuyến đi của mình một cách tốt nhất. **TripAdvisor** cũng sẽ giúp bạn có thể dễ dàng thực hiện việc đặt phòng, đăng ký các chuyến bay cũng như đặt chỗ tại nhà hàng, nơi mà bạn sắp đến. Nhưng các nghiên cứu này thường chỉ cung cấp các nội dung về: Cẩm nang đi du lịch, giới thiệu về các địa điểm du lịch bằng văn bản,... Không mang đến cho người dùng những trải nghiệm đi du lịch thực tế.

Với mục đích nâng cao chất lượng trải nghiệm khi đi du lịch tại Việt Nam, góp phần đưa hình ảnh, đất nước, con người Việt Nam đến với bạn bè quốc tế, thu hút đông đảo khách du lịch đến nước ta, tôi đã đề xuất triển khai đề tài “**Nghiên cứu xây dựng ứng dụng hỗ trợ khách du lịch tại Việt Nam**” thực hiện trong khuôn khổ luận văn thạc sĩ chuyên ngành “Khoa học máy tính”.

Nội dung luận văn được trình bày trong 3 chương cùng các phần mở đầu và kết luận.

Chương 1 của luận văn nêu lên tổng quan về du lịch Việt Nam và các hướng phát triển du lịch hiện nay.

Chương 2 trình bày các nội dung cơ bản về hệ điều hành iOS, giới thiệu về các thư viện và thuật toán được sử dụng để xây dựng nên sản phẩm của luận văn. Ngoài ra, nội dung của chương 2 cũng bao gồm phân tích thiết kế ứng dụng hỗ trợ khách du lịch tại Việt Nam.

Chương 3 bao gồm các bước và quá trình cài đặt ứng dụng hỗ trợ khách du lịch tại Việt Nam cùng các đánh giá tính hiệu quả của hệ thống. Một số kết quả thử nghiệm chính của sản phẩm đồ án được trình bày tại chương này.

Phần kết luận tóm tắt lại các nội dung đã đạt được của luận văn, và nêu lên một số gợi ý về hướng phát triển tiếp theo của luận văn.

CHƯƠNG 1. TỔNG QUAN VỀ DU LỊCH TẠI VIỆT NAM VÀ CÁC HƯỚNG PHÁT TRIỂN DU LỊCH HIỆN NAY

1.1. Giới thiệu tổng quan về du lịch tại Việt Nam

Việt Nam là một quốc gia có tiềm năng du lịch đa dạng và phong phú với nhiều thế mạnh:

Di sản Việt Nam

Cả nước có hơn 40.000 di tích và thắng cảnh, trong đó hơn 3.000 di tích được xếp hạng di tích quốc gia, 5.000 di tích được xếp hạng cấp tỉnh. Đồng bằng sông Hồng là khu vực có mật độ và số lượng di tích chiếm nhiều nhất với tỷ lệ lên đến 70%. Ngoài ra, tính trên địa bàn toàn quốc thì Việt Nam còn có 117 bảo tàng - nơi lưu giữ quá trình lịch sử hào hùng của dân tộc [6].

Việt Nam là một trong những số ít quốc gia trên thế giới được UNESCO công nhận nhiều di sản đến vậy. Tính đến nay, nước ta đã có 8 di sản được UNESCO công nhận: Di tích Hoàng thành Thăng Long, phố cổ Hội An, quần thể danh thắng Tràng An, cố đô Huế, thành nhà Hồ, thánh địa Mỹ Sơn, vườn quốc gia Phong Nha - Kẻ Bàng, vịnh Hạ Long. Đây là một trong những tiềm năng du lịch Việt Nam thu hút khách quốc tế.

Danh lam thắng cảnh

Theo UNESCO, Khu dự trữ sinh quyển là “những khu vực hệ sinh thái bờ biển và trên cạn giúp thúc đẩy các giải pháp điều hòa việc bảo tồn sự đa dạng sinh học với việc sử dụng bền vững khu vực đó”. Việt Nam đã được UNESCO công nhận 8 khu dự trữ sinh quyển thế giới, bao gồm: Châu thổ sông Hồng, Cát Bà, Tây Nghệ An, Đồng Nai, Cù Lao Chàm, Cần Giờ, Cà Mau và biển Kiên Giang.

Văn hóa và Ẩm thực

Văn hóa và Ẩm thực chính là hai trong các tiềm năng du lịch cần được gìn giữ và phát triển. Việt Nam có 54 dân tộc anh em, mỗi vùng miền, mỗi dân tộc đều có bản sắc văn hóa, phong tục tập quán, lối sống riêng khác nhau tạo thành nét cuốn hút riêng. Không chỉ vậy, Việt Nam còn có di sản văn hóa phi vật thể nổi tiếng: Nhã nhạc cung đình Huế, không gian văn hóa cồng chiêng Tây Nguyên, ca trù, quan họ, hát xoan, hội Gióng, tín ngưỡng thờ cúng Hùng Vương (10/03 Âm lịch)... để thu hút khách du lịch [1].

1.2. Các hướng phát triển du lịch hiện nay

Xu hướng hội nhập, hợp tác, cạnh tranh toàn cầu, giao lưu mở rộng và tăng cường ứng dụng khoa học công nghệ trong nền kinh tế tri thức trên thế giới và khu vực đã và đang tạo những cơ hội đồng thời cũng là thách thức đối với phát triển du lịch Việt Nam. Trước bối cảnh và xu hướng đó, định hướng phát triển Du lịch Việt Nam phải đáp ứng được những yêu cầu mới của thời đại về tính chuyên nghiệp, tính hiện đại, hội nhập và hiệu quả đồng thời bảo tồn và phát huy bản sắc dân tộc, yếu tố truyền thống để phát triển bền vững, tương xứng với tiềm năng, lợi thế của đất nước và đáp ứng yêu cầu cạnh tranh quốc tế.

Đầu tư phát triển du lịch có trọng tâm, trọng điểm theo định hướng ưu tiên; tiếp tục đầu tư nâng cao năng lực và chất lượng cung ứng du lịch, kết cấu hạ tầng, xúc tiến quảng bá và xây dựng thương hiệu du lịch.

1.2.1. Phát triển đầu tư hạ tầng du lịch

Việt Nam với 3.260km bờ biển, 125 bãi biển lớn nhỏ, hàng ngàn cảnh quan thiên nhiên và di tích văn hoá trải rộng khắp cả nước... nếu được đầu tư tốt về hạ tầng, chắc chắn ngành du lịch Việt Nam sẽ cất cánh. Đầu tư vào du lịch tăng cả về số lượng và chất lượng với nhiều dự án lớn từ các nhà đầu tư chiến lược như SunGroup, VinGroup, FLC, Vina Capital... Nhiều doanh nghiệp du lịch được thế giới bình chọn, vinh danh là khách sạn, khu nghỉ dưỡng sang trọng nhất thế giới.

Thành tựu thực tế du lịch mang lại thu nhập, tạo việc làm, thúc đẩy lan tỏa phát triển các ngành, lĩnh vực liên quan, góp phần chuyển dịch cơ cấu kinh tế, nâng cao đời sống xã hội, tăng cường hữu nghị, bảo tồn và phát huy giá trị văn hóa dân tộc và bảo vệ môi trường đang từng bước khẳng định du lịch dần trở thành ngành kinh tế mũi nhọn.

1.2.2. Phát triển quảng bá du lịch

Hiện nay, với sự phổ biến rộng rãi của Internet, xu thế cuộc Cách mạng công nghiệp lần thứ 4 (CMCN 4.0) đang lan tỏa và tác động mạnh mẽ đến mọi lĩnh vực, trong đó có du lịch. Từ đó, nhiều du khách đã chủ động tìm kiếm thông tin, lựa chọn dịch vụ của các nhà cung cấp cũng như phương thức mua sắm.

Theo đó, du lịch đứng trước yêu cầu chủ động nắm bắt thời cơ và vượt qua thách thức, xác định hướng đi và giải pháp phù hợp để phát triển bứt phá. Vì vậy, chúng ta cũng cần phải

phân tích, tìm hiểu những đặc tính của CMCN 4.0 cho công tác quảng bá xúc tiến du lịch. Cần phải có chuyên gia, bởi công nghệ đang ngày càng phát triển, chỉ cần một vài tháng đã thay đổi hết.

Với những thành tựu vượt bậc của công nghệ thông tin, du lịch có cơ hội tối ưu hóa hiệu quả công tác xúc tiến quảng bá và mở rộng thị trường, đẩy mạnh phát triển du lịch trực tuyến và thương mại điện tử. Đồng thời, tạo ra nhiều sản phẩm du lịch mới hấp dẫn hơn, góp phần nâng cao năng lực cạnh tranh và xây dựng thương hiệu du lịch.

1.2.3. Phát triển các ứng dụng hỗ trợ du lịch

Sự phát triển mạnh mẽ của xu hướng mobile cũng là một tiền đề quan trọng góp phần thúc đẩy khía cạnh cá nhân hóa hoạt động du lịch. Trên thực tế, khi việc tra cứu địa điểm và bản đồ, tìm kiếm và khớp lệnh dịch vụ được thực hiện nhanh và dễ dàng trên thiết bị di động thì người đi du lịch sẽ tự tin để tự thiết lập và lựa chọn các nội dung và dịch vụ du lịch theo ý muốn cá nhân.

Với những phần mềm, ứng dụng hỗ trợ du lịch như đưa ra thông tin của các điểm du lịch bao gồm các di tích, nhà hàng, khách sạn, điểm mua sắm, ... dần giúp cho du lịch Việt Nam được biết đến nhiều hơn với mọi người

1.3. Kết luận chương

Ứng dụng CNTT trong du lịch không phải là điều quá mới mẻ khi cũng đã có nhiều ứng dụng phát triển quảng bá du lịch. Với những trải nghiệm du lịch mới khi đi bất cứ đâu cũng luôn có ứng dụng như một hướng dẫn viên du lịch luôn đồng hành cùng người dùng vừa tiết kiệm được chi phí thuê hướng dẫn viên du lịch mà lại vừa đem lại những khoảng không gian riêng tư cho người dùng.

Việc ứng dụng CNTT trong du lịch Việt Nam góp phần quảng bá du lịch Việt Nam rộng rãi hơn, mang lại nhiều trải nghiệm tốt hơn cho người dùng cho bạn bè trong nước và quốc tế. Góp phần thúc đẩy sự phát triển của du lịch nước nhà.

CHƯƠNG 2. XÂY DỰNG ỨNG DỤNG HỖ TRỢ KHÁCH DU LỊCH TẠI VIỆT NAM

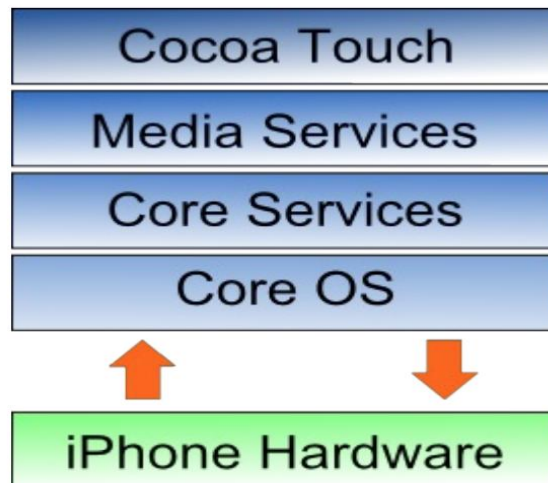
2.1. Giới thiệu về hệ điều hành iOS

2.1.1. Hệ điều hành iOS là gì?

iOS (trước đây là iPhone OS) là hệ điều hành trên các thiết bị di động của Apple. Đây là hệ điều hành chạy trên các sản phẩm iPhone, iPad, và iPod Touch và là hệ điều hành phổ biến thứ 2 trên toàn cầu, sau Android của Google.

Phiên bản mới nhất là iOS 13 và iPadOS 13 được ra mắt chính thức vào tháng 9 năm 2019 cùng với bộ 3 iPhone 11, iPhone 11 Pro và 11 Pro Max, hỗ trợ cập nhật cho các thiết bị từ iPhone 6s trở lên, iPod Touch (thế hệ 7), iPad (2017) trở lên, iPad Air (thế hệ thứ hai) trở lên, iPad Mini (thế hệ thứ 4) trở lên và iPad Pro [7].

2.1.2. Kiến trúc của hệ điều hành iOS



Hình 2.1. Kiến trúc hệ điều hành iOS

Lớp **Media** cung cấp cho hệ điều hành iOS các dịch vụ đa phương tiện như âm thanh, video, hình ảnh, đồ họa, bao gồm một số thành phần sau:

- Core Graphics Framework
- Quartz Core Framework
- ES Framework
- Audio
- AV Foundation Framework

- Core Audio Framework
- Open Audio Library
- Media Player Framework

Lớp **Core Services** cung cấp một số dịch vụ cơ bản cho hệ điều hành và các lớp khác, bao gồm:

- Address book Framework
- Core Data Framework
- Core Foundation Framework
- Foundation Framework
- Core Location Framework
- Store Kit Framework
- SQLite Library

Lớp **Core OS** cung cấp một số dịch vụ hệ thống như memory manager, networking, file system, ... bao gồm:

- CFNetwork Framework
- External Accessory Framework
- Security Framework

System Lớp **Cocoa Touch** cung cấp các framework chung cho các lập trình viên phát triển ứng dụng, bao gồm:

- UIKit Framework
- Map Kit Framework
- Push Notification Service
- Message UI Framework
- Address Book UI Framework
- Game Kit Framework

2.2. Giới thiệu các thư viện và thuật toán sử dụng

2.2.1. Tổng quan về thư viện Mapkit

Trước khi giới thiệu về thư viện Mapkit, tác giả sẽ đưa ra các nội dung cơ bản sau:

Hệ thống Định vị Toàn cầu (tiếng Anh: Global Positioning System – GPS) là hệ thống xác định vị trí dựa trên vị trí của các vệ tinh nhân tạo, do Bộ Quốc phòng Hoa Kỳ thiết kế, xây dựng, vận hành và quản lý. Trong cùng một thời điểm, tọa độ của một điểm trên mặt đất sẽ được xác định nếu xác định được khoảng cách từ điểm đó đến ít nhất ba vệ tinh. Tuy được quản lý bởi Bộ Quốc phòng Hoa Kỳ, chính phủ Hoa Kỳ cho phép mọi người trên thế giới sử dụng một số chức năng của GPS miễn phí, bất kể quốc tịch nào.

Core Location là framework của Apple được tích hợp sẵn vào trong iOS. Giúp cho ứng dụng của bạn có thể lấy được vị trí người dùng thông qua GPS. Tìm kiếm các địa điểm thông qua tọa độ, hoặc ngược lại. Cung cấp dữ liệu để làm việc với các Framework cho Map (Google Map hay MapKit).

Một số tính năng từ Core Location:

- Theo dõi vị trí hiện tại của người dùng
- Chạy ngầm ở chế độ background
- Tự động cập nhật vị trí người dùng mới
- Tự động điều chỉnh cấu hình để tiết kiệm pin khi phải thường xuyên theo dõi GPS
- Cung cấp các API để chuyển đổi từ tọa độ sang địa chỉ và ngược lại

Ta có các class đặc trưng của Core Location

CLLocationManager

- Lớp quản lý các thao tác khi làm việc với Core Location
- Cấu hình để lấy vị trí người dùng được chính xác hơn
- Yêu cầu cấp quyền từ phía người dùng

CLLocation

- Chứa thông tin của 1 vị trí
- Latitude : kinh độ
- Longitude : vĩ độ

CLGeocoder

- Giúp cho việc chuyển đổi từ tọa độ thành địa chỉ và ngược lại
- Tìm kiếm dựa trên tọa độ, địa chỉ ... của địa điểm

Bản đồ là một phần quan trọng của cuộc sống. Chúng ta sử dụng chúng hằng ngày để tìm kiếm địa điểm và những chỉ dẫn. MapKit framework làm việc này trở nên đơn giản cho những người lập trình để tích hợp đến những ứng dụng.

MapKit

- Là framework chính chủ của Apple
- Được xây dựng trên các API và data của Apple Map
- Cung cấp cho các lập trình viên một tập các công cụ để thao tác và tích hợp Map vào ứng dụng iOS của họ

MKMapView: MKMapView là một UI control mà nó được sử dụng để hiển thị bản đồ trên những thiết bị iOS Annotation:

Annotations là thông tin liên quan đến một địa điểm cụ thể nào đó trên map. Những cái này bao gồm: cửa hàng, tạp hóa, sân ga, mall hay bất kì thông tin custom nào được insert bởi người dùng.

AnnotationView: là một thể hiện của Annotations, mà nó được định nghĩa bằng cách sử dụng MKAnnotationView class. Mặc định thì những annotations được hiển thị bằng cách sử dụng view pin annotation.

Overlay: Là các đối tượng mà dùng để vẽ lên MapView, như: hình tròn, vuông, chữ nhật, đa giác, ảnh ...

Map Kit cung cấp 3 hệ tọa độ để xác định điểm trên bản đồ:

Map coordinate: là cách cơ bản để xác định địa điểm trên trái đất, biểu diễn kinh độ và vĩ độ trên trái đất. Sử dụng:

Map point: là giá trị của x và y trên bản đồ được chiếu theo phép chiếu Mecartor.

2.2.2. Giới thiệu về SQLite

SQLite là một hệ thống quản lý cơ sở dữ liệu quan hệ nhúng. Hầu hết các cơ sở dữ liệu quan hệ (điển hình như Oracle hay MySQL) chạy trên máy chủ, chạy độc lập, cho phép các ứng dụng yêu cầu truy cập cơ sở dữ liệu kết nối vào. Tuy nhiên SQLite được gọi là hệ cơ sở dữ liệu nhúng vì nó được cung cấp dưới dạng một thư viện liên kết bên trong ứng dụng. Như vậy, hệ cơ sở dữ liệu SQLite không có máy chủ cơ sở dữ liệu nào, tất cả các hoạt động truy xuất cơ sở dữ liệu được xử lý nội bộ trong ứng dụng thông qua các hàm của thư viện SQLite.

SQLite là một hệ cơ sở dữ liệu mã nguồn mở và hiện tại đã trở thành giải pháp cơ sở dữ liệu triển khai rộng rãi.

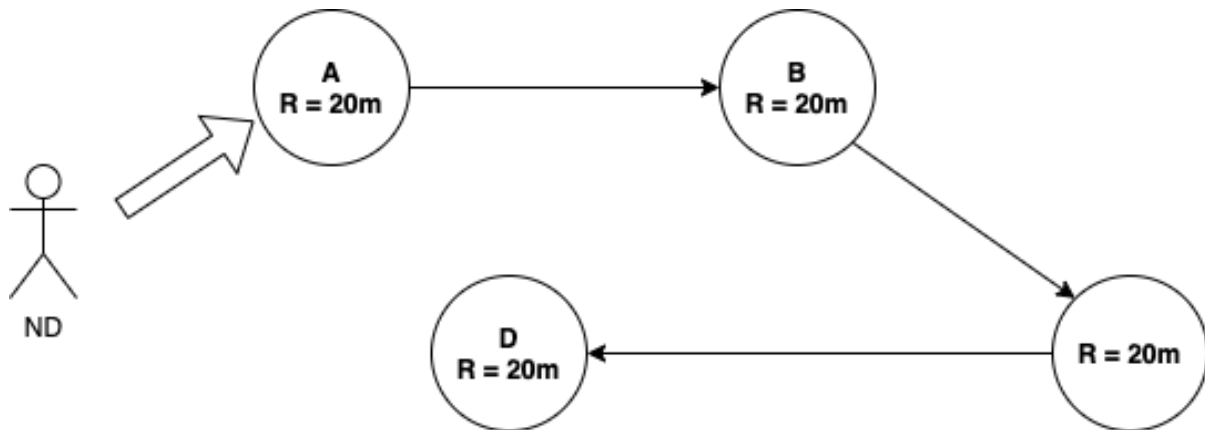
SQLite được viết bằng ngôn ngữ C bởi vậy để sử dụng SQLite trong Swift hoặc là cần sử dụng những cú pháp C phức tạp, các kiểu dữ liệu, con trỏ của ngôn ngữ C, hoặc cách tiếp cận dễ dàng hơn là sử dụng một SQLite wrapper như một lớp đứng giữa SQLite và Swift với cú pháp đơn giản dễ đọc hơn.

Structured Query Language (SQL): Dữ liệu được truy vấn trong SQLite sử dụng một ngôn ngữ cấp cao được gọi là ngôn ngữ truy vấn có cấu trúc (thường được viết tắt là SQL). SQL là một chuẩn ngôn ngữ sử dụng bởi hầu hết các hệ cơ sở dữ liệu quan hệ.

2.2.3. Nghiên cứu một số thuật toán tính toán

Việc nghiên cứu ra các thuật toán rất quan trọng trong ngành khoa học máy tính nói chung, công nghệ thông tin và lập trình phần mềm nói riêng. Các thuật toán được trình bày trong phần này sẽ giúp chọn ra một địa điểm có khoảng cách người dùng ngắn nhất trong tập hợp các địa điểm trong cơ sở dữ liệu một cách nhanh và chính xác nhất khi người dùng liên tục thay đổi vị trí của mình giúp cho ứng dụng có thể vận hành một cách tự động. Thuật toán được nghiên cứu giúp giải quyết các vấn đề được đưa ra trong bài toán sau đây.

Giới thiệu về bài toán: Tìm điểm có khoảng cách tới người dùng nhỏ nhất.



Hình 2.2. Tổng quan bài toán “Tìm điểm có khoảng cách ngắn nhất tới người dùng”

Để đáp ứng được nhu cầu của ứng dụng, bài toán được đưa ra như sau:

Đầu vào của bài toán là tập hợp các địa điểm có trong cơ sở dữ liệu (Tất cả các địa điểm trong tập các địa điểm đều có một giá trị bán kính riêng, vị trí khác nhau – vị trí của địa

điểm được xác định trên bản đồ bằng kinh độ và vĩ độ) và vị trí hiện tại của người dùng liên tục thay đổi.

Đầu ra của bài toán là một điểm có khoảng cách tới vị trí hiện tại của người dùng ngắn nhất.

Chúng ta sẽ đi tìm một địa điểm trong tập các địa điểm có khoảng cách ngắn nhất tới vị trí hiện tại của người dùng bằng cách tính khoảng cách từ người dùng đến tất cả các địa điểm trong tập các địa điểm. Sau khi có tìm được địa điểm có khoảng cách ngắn nhất tới người dùng, chúng ta so sánh khoảng cách của người dùng đến địa điểm đó, nếu khoảng cách của người dùng đến địa điểm đó nhỏ hơn hoặc bằng giá trị bán kính của điểm đó thì sẽ phát lên file âm thanh giới thiệu về địa điểm đó. Địa điểm gần nhất này sẽ thay đổi khi người dùng di chuyển. Mỗi file âm thanh giới thiệu về địa điểm sẽ chỉ được nghe tự động một lần.

Để giải quyết được bài toán nêu trên, tác giả đã nghiên cứu và sáng tạo nên thuật toán “Tìm điểm có khoảng cách tới người dùng nhỏ nhất” và thuật toán “Bật file âm thanh khi người dùng nằm trong bán kính của địa danh”

2.2.3.1. Thuật toán: Tìm điểm có khoảng cách tới người dùng nhỏ nhất.

Tổng quan về thuật toán

```
var distanceMin : Double
var scoreVal : Double
var locationChoose : Location
for location : Location in locationS{
    let distanceTemp = distanceUserToPlace(location);
    distanceMin = distanceTMP;
    scoreVal = location.scoreLocation;
    if distanceTemp < distanceMin {
        distanceMin = distanceTemp;
        locationChoose = location;
        scoreVal = location.scoreLocation;
    }
}
```

Giải thích thuật toán:

Đầu tiên ta khai báo ba biến

- **distanceMin**: Khoảng cách ngắn nhất từ người dùng tới địa điểm

- **scoreVal**: Bán kính của điểm có khoảng cách ngắn nhất
- **locationChoose**: Điểm có khoảng cách ngắn nhất tới người dùng

Sử dụng vòng lặp for với mỗi địa danh trong tập các địa danh ta đều tính được khoảng cách tới người dùng dựa vào hàm **distanceUserToPlace(location)**. Ta sẽ gán giá trị trả về của hàm **distanceUserToPlace(location)** cho biến **distanceTemp** và bán kính của địa điểm **location.scoreLocation** cho **scoreVal**. Sau đó so sánh 2 biến **distanceTemp** và **distanceMin**. Nếu như **distanceTemp < distanceMin** thì ta lại gán giá trị của biến **distanceTemp** cho **distanceMin**, gán địa điểm đó **location** cho **locationChoose** và giá trị của **location.scoreLocation** cho **scoreVal**.

Sau khi xác định được địa danh có khoảng cách ngắn nhất tới người dùng và bán kính của địa danh đó ta sẽ cần một thuật toán để có thể phát file âm thanh giới thiệu về địa danh đó sao cho mỗi địa danh chỉ nghe tự động một lần. Mỗi địa danh sẽ có 1 bán kính riêng. Khi khoảng cách từ người dùng đến một địa danh bất kỳ nhỏ hơn hoặc bằng với bán kính của địa danh đó ứng dụng sẽ phát bài giới thiệu về địa danh đó. Khi người dùng đã nghe được bài giới thiệu về địa danh nào thì chúng ta sẽ lưu địa danh đó vào một mảng các địa danh đã nghe. Từ yêu cầu trên ta sẽ có hai thuật toán để giải bài toán.

2.2.3.2. Thuật toán: Bật file âm thanh khi người dùng nằm trong bán kính của địa danh

```
if (distance(ND -> A) <= RA) {
    playAudio(A)
    isPlayAudio = true
    addToArrayPlayed(A)
}
```

Sau khi xác định được điểm A có khoảng cách ngắn nhất tới người dùng. Đây là thuật toán trên lý thuyết so sánh khoảng cách giữa người dùng tới điểm A và bán kính của điểm A. Khi khoảng cách người dùng đến điểm A nhỏ hơn hoặc bằng bán kính điểm A thì hàm bật file âm thanh(**playAudio(A)**) sẽ được gọi và thêm điểm A vào danh sách đã được nghe(**addToArrayPlayed(A)**), sẽ không được phát lại cho đến khi người dùng khởi động lại ứng dụng.

So với ý tưởng ban đầu (thuật toán sử dụng bán kính của người dùng thay vì địa điểm) thì thuật toán mới này đem lại độ chính xác cao hơn. Giúp đem lại trải nghiệm tốt hơn tới

người dùng. Một điểm mới của thuật toán đó là khi người dùng nghe file âm thanh này rồi thì file âm thanh sẽ không bật lên trong lần tới, không gây cảm giác nhàm chán cho người dùng

2.2.3.3. Thuật toán: Kiểm tra bật file âm thanh khi người dùng đến gần địa danh khác

```
if (distance(ND -> B) <= RB && isPlayAudio = false {
    playAudio(B);
} else {
    playAudio(A);
}
```

Đây là thuật toán trên lý thuyết để kiểm tra bật file âm thanh khi người dùng đến gần địa điểm khác. Khi người dùng nghe hết bài giới thiệu về địa danh thì biến isPlayAudio = false. Khi người dùng di chuyển đến điểm B. mà khoảng cách từ người dùng đến điểm B nhỏ hơn khoảng cách người dùng đến điểm A.

Khi người dùng đến điểm B mà vẫn đang nghe bài giới thiệu về điểm A. Đến khi nghe hết bài giới thiệu về điểm A mà vẫn trong bán kính của điểm B thì người dùng sẽ được tiếp tục nghe giới thiệu về điểm B. Còn nếu người dùng nằm trong bán kính của điểm C thì sẽ phát bài giới thiệu của điểm C. Khi người dùng quay lại điểm B sẽ chỉ nghe được bài giới thiệu của điểm B mà không phải nghe lại hai bài giới thiệu về điểm A và C trước đó.

2.2.4. Áp dụng các thuật toán vào ứng dụng cụ thể

2.2.4.1. Thuật toán: Cập nhập vị trí người dùng

Để áp dụng được trong thực tế ứng dụng ta cần một hàm chức năng có khả năng cập nhập lại vị trí của người dùng tới các địa điểm để xem có thay đổi về địa điểm gần nhất hay không.

```
var latUser : String!
var longUser : String!
Func sẽ được cập nhập sau mỗi 5 giây
func updateLocation(location: CLLocation) {
    latUser = location.coordinate.latitude
    longUser = location.coordinate.longitude
    getMindistance ()
}
```

Hàm này sẽ cập nhập giá trị lat, long của người dùng sau mỗi 5 giây

2.2.4.2. Thuật toán: Tính khoảng cách người dùng đến một địa danh

Trên lý thuyết chúng ta sẽ tính khoảng cách của người dùng đến các điểm. Nếu ứng dụng vào bản đồ, ta sẽ tính khoảng cách của người dùng đến các điểm dựa trên kinh độ, vĩ độ của người dùng và các địa điểm.

```
func distanceToPlace(latUserVal:Double, longUserVal:Double, latPlace:Double,
longPlace:Double) -> Double {
    let myLocation = CLLocation(latitude: latUserVal, longitude: longUserVal)
    let myPlacesLocation = CLLocation(latitude: latPlace, longitude: longPlace)
    let distance = myLocation.distance(from: myPlacesLocation)
    return distance
}
```

Thuật toán tính toán khoảng cách của người dùng đến một địa danh dựa trên kinh độ, vĩ độ của người dùng và địa danh bất kỳ.

2.2.4.3. Thuật toán: Tìm địa danh có khoảng cách với người dùng ngắn nhất

```
var distanceMin: Double = -1
var placeObj : LocationMap
var locationMaps = [LocationMap]()
var isPlayAudio : Bool = false
func getMindistance(){
    for locationMap : LocationMap in locationMaps{
        distanceMin = -1 // Reset giá trị
        let distanceTemp = distanceToPlace(latUserVal:latUser, longUserVal:longUser,
latPlace:locationMap.lat, longPlace: locationMap.long)
        if ( distanceMin == -1 ){ //Initial
            distanceMin = distanceTemp
            placeObj = locationMap
            self.scoreVal = locationMap.locScore // locScore là bán kính điểm
        } else {
            if (distanceTemp < distanceMin){
                if(distanceTmp == 0){
                    distanceMin = distanceTemp
                    placeObj = locationMap
                    self.scoreVal = locationMap.locScore
                    break
                } else {
                    distanceMin = distanceTemp
                    placeObj = locationMap
                    self.scoreVal = locationMap.locScore
                }
            }
        }
    }
}
```


Dựa vào thuật toán “Tính toán khoảng cách người dùng đến một địa danh”, thuật toán “Tìm địa danh có khoảng cách với người dùng gần nhất” sẽ tính khoảng cách từ người dùng đến các địa danh có trong CSDL, từ đó lưu trữ địa danh có khoảng cách gần nhất tới người dùng lại.

2.2.4.4. Thuật toán: Kiểm tra khoảng cách để bật file âm thanh

```
func checkToPlayAudio() {
    if ((distanceMin <= scoreVal) && (isPlayAudio == false)){
        playSound(soundName: placeObj.loc_audio)
    }
}
```

2.2.4.5. Thuật toán: Bật file âm thanh

```
var arrLocAudio : [String] = []

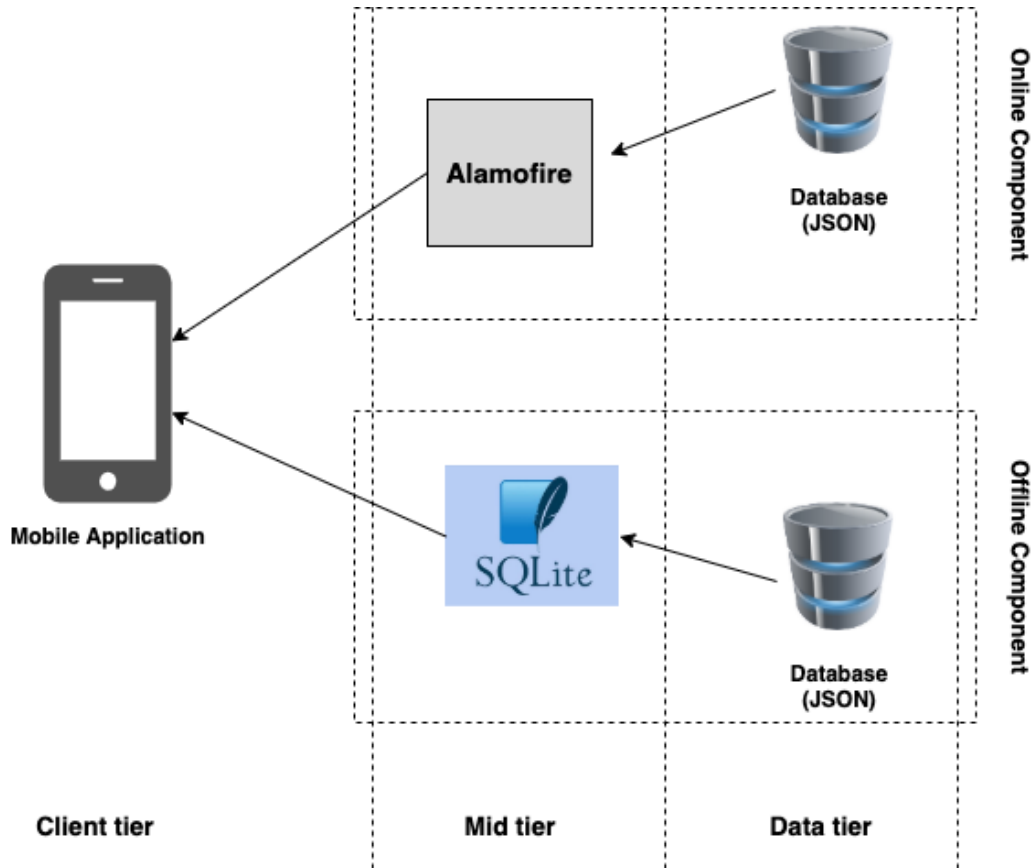
func playSound(soundName : String) {
    if (self.checkExistArrayValWithKey(“ArrAudioPlayed”) == false){
        self.arrLocAudio.append(soundName)
        self.saveArrayValWithKey(“ArrAudioPlayed”, self.arrLocAudio)
    } else {
        self.arrLocAudio = self.getArrayValWithKey(“ArrAudioPlayed”)
        if let object = self.arrLocAudio.filter({ $0 == soundName }).first {
            print("Audio đã nghe", object)
        } else {
            self.arrLocAudio.append(soundName)
            self.saveArrayValWithKey(“ArrAudioPlayed”), arrLocAudio)
            playAudio(soundName)
            self.isPlayAudio = true
        }
    }
}
```

Sau khi xác định được địa danh gần nhất với người dùng, thuật toán “Bật file âm thanh” sẽ kiểm tra xem người dùng đã nghe file âm thanh giới thiệu địa danh này chưa nhờ vào mảng danh sách các địa danh đã được lưu khi người dùng nghe trước đó. Nếu địa danh chưa có

trong mảng các địa danh đã được nghe thì file âm thanh của địa danh đó sẽ được phát và lưu địa danh đó vào mảng các địa danh đã phát.

2.3. Phân tích thiết kế ứng dụng hỗ trợ khách du lịch tại Việt Nam

2.3.1. Sơ đồ kiến trúc tổng thể



Hình 2.3. Sơ đồ kiến trúc tổng thể của toàn hệ thống

Mobile application: Ứng dụng di động được xây dựng dựa trên rất nhiều Framework con (sử dụng ngôn ngữ lập trình Objective-C và Swift). Ứng dụng di động sử dụng Alamofire, SQLite để tương tác với cơ sở dữ liệu

Alamofire: Thư viện Networking kết nối các hàm API trên Web Service để lấy dữ liệu

SQLite: Là một thư viện phần mềm mà triển khai một SQL Database Engine, không cần máy chủ, không cần cấu hình, khép kín và nhỏ gọn. Nó là một cơ sở dữ liệu, không cần cấu hình, có nghĩa là giống như các cơ sở dữ liệu khác mà bạn không cần phải cấu hình nó trong hệ thống của mình.

Database: Chứa dữ liệu dưới dạng JSON

2.3.2. Phân tích yêu cầu

2.3.2.1. Yêu cầu về dữ liệu

Bảng 2.1. Dữ liệu về địa danh

Dữ liệu về địa danh		
Tên biến	Kiểu dữ liệu	Mô tả
post_id	Int	ID của địa danh
name	String	Tên địa danh
another_name	String	Tên gọi khác của địa danh
address	String	Địa chỉ địa danh
service_time	String	Giờ mở cửa
featured_image	String	Ảnh địa danh
content	String	Bài giới thiệu về địa danh
ticket_price	String	Giá vé vào cửa
phone	String	Số điện thoại địa danh
content_voice	String	Đường dẫn đến file âm thanh
long	String	Kinh độ của địa danh
lat	String	Vĩ độ của địa danh
loc_type	Int	Loại địa danh
zone_id	Int	ID khu vực
zone_name	String	Tên khu vực

Bảng 2.2. Dữ liệu về người dùng

Dữ liệu về người dùng		
Tên biến	Kiểu dữ liệu	Mô tả
id	Int	ID của người dùng
email	String	Email của người dùng
facebook_id	String	ID Facebook của người dùng
device_id	String	ID điện thoại của người dùng
phone	String	Số điện thoại của người dùng
name	String	Tên của người dùng
birthday	String	Ngày sinh nhật của người dùng
avatar	String	Đường dẫn đến file hình ảnh đại diện
address	String	Địa chỉ của người dùng
gender	String	Giới tính của người dùng
lang	String	Ngôn ngữ của người dùng

2.3.2.2. *Yêu cầu chức năng đăng ký*

Giới thiệu: Chức năng này cho phép người dùng đăng ký tài khoản để đăng nhập vào hệ thống

Yêu cầu: Ứng dụng phải cung cấp các hình thức đăng ký khác nhau để người dùng lựa chọn

- Đăng ký tài khoản riêng;
- Đăng ký tài khoản bằng tài khoản Facebook;
- Đăng ký tài khoản bằng tài khoản Google;

2.3.2.3. *Yêu cầu chức năng đăng nhập*

Giới thiệu: Chức năng này giúp người dùng đăng nhập vào hệ thống để sử dụng các chức năng khác của hệ thống

Yêu cầu: Ứng dụng phải cung cấp các hình thức đăng nhập khác nhau để người dùng lựa chọn

- Đăng nhập nhanh: Giúp người dùng đăng nhập dễ dàng vào hệ thống dựa trên ID của thiết bị;
- Đăng nhập với tài khoản Facebook;
- Đăng nhập bằng tài khoản Google;
- Đăng nhập bằng tài khoản riêng đã đăng ký.

2.3.2.4. *Yêu cầu chức năng khám phá các địa danh*

Giới thiệu: Chức năng này giúp người dùng có thể biết thêm thông tin về các địa điểm du lịch như: Địa chỉ, Giờ, Giá vé, Số điện thoại, Giới thiệu địa điểm bằng chữ, Giới thiệu địa điểm bằng âm thanh, Các địa điểm thăm quan xung quanh, Các địa điểm mua sắm, Các địa điểm ăn uống.

Yêu cầu: Trong dữ liệu phải cung cấp đầy đủ các thông tin về địa điểm.

2.3.2.5. *Yêu cầu chức năng nghe giới thiệu về địa danh*

Giới thiệu: Chức năng này giúp người dùng khi đến một địa danh có lưu trong dữ liệu, ứng dụng sẽ phát một bài giới thiệu về địa danh đó dựa vào vị trí của người dùng và vị trí của địa danh đó.

Yêu cầu: Trong dữ liệu có lưu thông tin về kinh độ, vĩ độ của địa danh và ứng dụng phải xác định được kinh độ, vĩ độ của người dùng. Từ đó tính khoảng cách giữa vị trí người dùng và địa danh để kích hoạt được tính năng.

2.3.2.6. Yêu cầu chức năng xem bản đồ các địa danh

Giới thiệu: Chức năng này cho phép người dùng có thể xem được tất cả các địa danh trên bản đồ.

Yêu cầu: Ứng dụng cung cấp các địa danh lên bản đồ bằng việc lấy kinh độ, vĩ độ, tên của địa danh.

2.3.2.7. Yêu cầu chức năng xem lịch sử

Giới thiệu: Chức năng này cho phép người dùng xem lại các địa danh mình đã đi qua.

Yêu cầu: Ứng dụng lưu lại các địa danh mà người đó đã đi qua để hiển thị lên cho người dùng.

2.3.2.8. Yêu cầu chức năng xem thông báo

Giới thiệu: Chức năng thông báo cho người dùng các tin tức mới về ứng dụng.

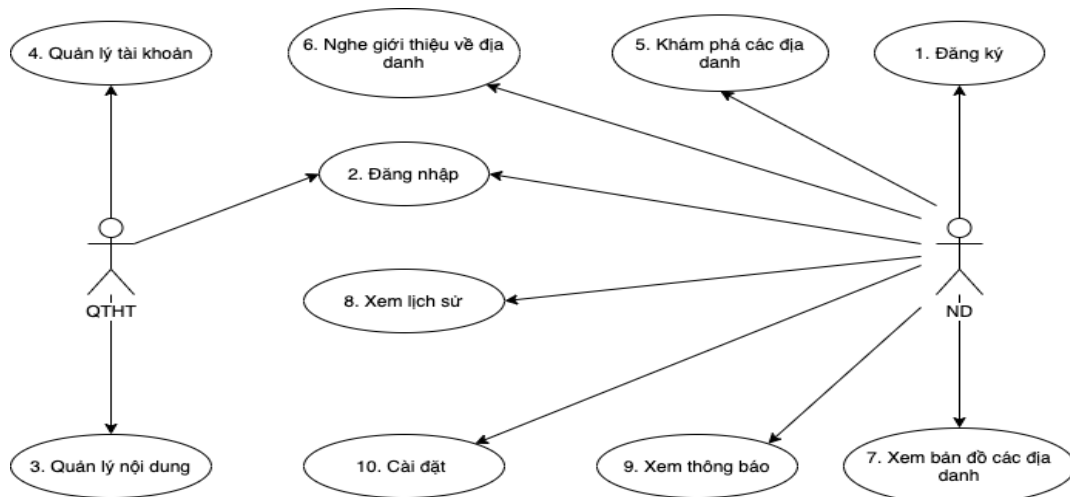
Yêu cầu: Ứng dụng gửi các thông báo mới cho người dùng.

2.3.2.9. Yêu cầu chức năng cài đặt

Giới thiệu: Người dùng có thể tùy chỉnh thông tin người dùng, chuyển đổi ngôn ngữ

Yêu cầu: Ứng dụng cung cấp cho người dùng các phương thức để thay đổi được thông tin người dùng và danh sách các ngôn ngữ hỗ trợ.

2.3.2.10. Sơ đồ use-case các chức năng chính



Hình 2.4. Sơ đồ Use-Case tổng quan các chức năng của ứng dụng

2.3.2.11. Các tác nhân chính tham gia hệ thống

Quản trị hệ thống (QTHT): Là tác nhân có quyền đăng nhập vào hệ thống quản lý tài khoản và quản lý nội dung.

Người dùng (ND): Tác nhân này có thể đăng nhập vào hệ thống, sử dụng hệ thống để khám phá thêm các địa danh nổi tiếng, nghe giới thiệu về địa danh, xem bản đồ các địa danh, xem lịch sử các địa danh đã đi qua, xem thông báo của hệ thống và cài đặt, tùy chỉnh các tiện ích khác.

2.3.3. Thiết kế chi tiết

2.3.3.1. Chức năng đăng ký

2.3.3.2. Chức năng đăng nhập

2.3.3.3. Chức năng khám phá các địa danh

2.3.3.4. Chức năng nghe giới thiệu về địa danh

2.3.3.5. Chức năng xem bản đồ các địa danh

2.3.3.6. Chức năng xem lịch sử

2.3.3.7. Chức năng xem thông báo

2.4. Kết luận chương

Dựa trên việc phân tích, thiết kế, phân rã các chức năng cũng như nghiên cứu cải tiến các thuật toán cho ra các kết quả chính xác hơn, giúp cho trải nghiệm của người dùng được tốt hơn, tối ưu hóa hiệu suất hoạt động của ứng dụng.

Với việc tích hợp sử dụng ứng dụng trong môi trường không có internet giúp người dùng không cần phải lo lắng khi hết dung lượng 4G hay không có wifi.

CHƯƠNG 3. CÀI ĐẶT VÀ ĐÁNH GIÁ

3.1. Cài đặt ứng dụng

3.1.1. Môi trường cài đặt và các yêu cầu phần cứng, phần mềm

3.1.1.1. Môi trường cài đặt

Đường link tải ứng dụng: <https://testflight.apple.com/join/JO0WueHK>

Quá trình triển khai và xây dựng phần mềm đã áp dụng các giải pháp công nghệ sau:

1. Ngôn ngữ lập trình: Swift version 5.0;
 2. IDE (Môi trường phát triển tích hợp): Xcode version 11.4.1 (11E503a);
 3. Bộ công cụ quản lý mã nguồn mở cho iOS: CocoaPods;
 4. Các thư viện iOS bên thứ ba (Third Party):
 - RxSwift: RxSwift là framework sử dụng cho ngôn ngữ Swift theo kỹ thuật reactive;
 - Alamofire: Thư viện Networking kết nối các hàm API trên Web Service để lấy dữ liệu;
 - Kingfisher: Là một thư viện nhẹ để tải xuống và lưu trữ hình ảnh từ web
 - AV Foundation: Là framework đầy đủ tính năng để làm việc với phương tiện nghe nhìn trên iOS, macOS, watchOS và tvOS.
 - Core Location: Là framework trong iOS cung cấp các service liên quan tới vị trí của thiết bị, độ cao, hướng.
 - FacebookCore: Thư viện giúp sử dụng Facebook để đăng nhập.
 - GoogleSignIn: Thư viện giúp dùng Gmail để đăng nhập.
 5. Công cụ cung cấp thư viện hình ảnh icon: Icons 8 Lite;
 6. Công cụ render kích thước ảnh cho App Icon: Iconical;
 7. Công cụ soạn thảo văn bản: Microsoft Office Word for Mac;
 8. Công cụ tạo báo cáo thuyết trình: Microsoft Office Power Point for Mac;
 9. Công cụ vẽ sơ đồ: draw.io.
- #### 3.1.1.2. Yêu cầu phần cứng, phần mềm
- Phần cứng: iPhone 6 trở lên
 - Phần mềm: iOS 11 trở lên
 - Tài khoản người dùng:

- + Tài khoản khách;
- + Tài khoản Facebook;
- + Tài khoản gmail.

3.1.2. Giới thiệu một số mô đun chương trình

3.1.2.1. *Mô đun khảo sát*

Giao diện khảo sát này giúp cho người dùng có thể tùy chọn được ngôn ngữ và địa danh chuẩn bị đi du lịch. Giúp lưu trữ dữ liệu khi người dùng không có internet được tốt hơn.

3.1.2.2. *Mô đun đăng ký*

Người dùng đăng ký mới có thể đăng ký tài khoản của hệ thống.

3.1.2.3. *Mô đun đăng nhập*

Người dùng có thể đăng nhập bằng tài khoản đăng ký của hệ thống, đăng nhập nhanh, đăng nhập với tài khoản Facebook, đăng nhập với tài khoản Google.

3.1.2.4. *Mô đun khám phá*

Ở giao diện khám phá người dùng được cung cấp thông tin tổng quan về địa điểm, các địa điểm nên khám phá khi đến địa danh, và danh sách các khu vực khác.

3.1.2.5. *Mô đun lịch sử*

Mỗi khi người dùng đi đến địa danh nào. Lịch sử của người dùng đều được ghi lại ở giao diện lịch sử.

3.1.2.6. *Mô đun bản đồ*

Người dùng có thể biết chính xác được các địa danh trên bản đồ và có thể xem được danh sách các địa danh đó. Các địa danh được chia làm 3 loại: Di tích, ăn uống và mua sắm. Người dùng có thể tìm kiếm các địa danh.

3.1.2.7. *Mô đun cài đặt*

Người dùng có thể cập nhập thông tin người dùng tại giao diện cài đặt. Người dùng có thể tùy chọn ngôn ngữ cho ứng dụng.

3.2. Một số kết quả thử nghiệm

3.2.1. Kết quả thử nghiệm độ chính xác nhận biết địa điểm khi người dùng đến nơi và khi chuẩn bị chuyển sang địa điểm khác.

Khi người dùng đến gần địa danh thì sẽ phát một file âm thanh giới thiệu về địa danh đó. Hình ảnh chỉ mang tính chất minh họa cho chức năng hoạt động đúng như kỳ vọng. Vì ứng dụng có sử dụng âm thanh nên khi demo ứng dụng sẽ chi tiết hơn về độ chính xác.

3.2.2. Kết quả thử nghiệm lưu lại lịch sử khi người dùng đặt chân đến các địa danh.

Mỗi khi người dùng đi đến địa danh nào. Lịch sử của người dùng đều được ghi lại ở giao diện lịch sử.

3.3. Đánh giá tính hiệu quả

3.3.1.1. *Tính năng nghe giới thiệu về địa danh*

Tính năng nghe giới thiệu về địa danh chủ yếu sử dụng GPS để xác định vị trí người dùng. Do độ chính xác của thiết bị định vị phụ thuộc vào rất nhiều yếu tố, đặc biệt là tỉ lệ giữa

tín hiệu và nhiễu, vị trí vệ tinh, thời tiết và các vật cản như nhà cao tầng hay các ngọn núi, hầm xe của các tòa nhà hay đường hầm. Những yếu tố này có thể gây lỗi cho vị trí cần được xác định. Nhiễu tín hiệu thường tạo ra sai số khoảng 1-10 mét. Các ngọn núi, nhà cao tầng hay các vật cản khác chặn đường tín hiệu giữa bộ phận thu và vệ tinh có thể gây ra sai số lớn gấp ba lần sai số gây ra do nhiễu tín hiệu. Vì vậy trong điều kiện thời tiết lý tưởng khi người dùng trải nghiệm, ứng dụng đáp ứng tốt nhu cầu của người dùng.

3.3.1.2. *Tính năng khám phá địa danh*

Với nội dung phong phú có chọn lọc, tính năng này mang đến cho người dùng tất cả thông tin cần có của một địa điểm du lịch như: Tên địa danh, địa chỉ địa danh, giờ mở cửa của địa danh, giá vé vào cổng, số điện thoại, chức năng chỉ đường tới địa danh, bài giới thiệu về địa danh bằng âm thanh cũng như là bằng văn bản và các địa điểm gợi ý khác khi đến địa danh.

3.4. *Nhận xét*

Các thuật toán sử dụng trong ứng dụng đã đem đến rất nhiều hiệu quả cũng như hiệu năng tốt đối với ứng dụng. Khi thực nghiệm đã cho ra các kết quả chính xác.

Bỏ qua những sai số chấp nhận được do sử dụng tính năng GPS để xác định vị trí người dùng cùng với địa điểm ảnh hưởng bởi thời tiết cũng như địa hình thì khả năng ứng dụng trên thực tế của ứng dụng rất cao khi các kết quả thực nghiệm đều đem đến độ chính xác gần như là tuyệt đối.

3.5. *Kết luận chương*

Trong quá trình thực hiện đề tài, cùng với việc thiết kế giao diện thân thiện, dễ sử dụng, tác giả cũng nghiên cứu áp dụng, nâng cấp các thuật toán để giúp cho ứng dụng mang lại nhiều trải nghiệm tốt hơn cho người dùng. Với mong muốn ứng dụng đến được với đông đảo bạn bè trong nước và quốc tế, tác giả luôn đón nhận những ý kiến góp ý từ tất cả mọi người để phát triển ứng dụng tốt nhất có thể.

Hiện tại ứng dụng đang trong giai đoạn thử nghiệm và có sẵn phiên bản thử nghiệm dành cho hệ điều hành iOS tại địa chỉ:

<https://testflight.apple.com/join/JO0WueHK>

Tác giả rất mong nhận được nhiều phản hồi về ứng dụng.

KẾT LUẬN

Luận văn “*Nghiên cứu xây dựng ứng dụng hỗ trợ khách du lịch tại Việt Nam*” với mục đích xây dựng nên ứng dụng hỗ trợ cho khách du lịch tại Việt Nam cũng như góp phần quảng bá du lịch Việt Nam đã đạt được những kết quả chính sau:

Lý thuyết:

Về mặt lý thuyết, luận văn đã giới thiệu về hệ điều hành iOS; giới thiệu về các thư viện sử dụng trong luận văn bao gồm thư viện Mapkit và thư viện SQLite; luận văn cũng đưa ra các nghiên cứu một số thuật toán tính áp dụng cho bài toán được đưa ra và áp dụng các thuật toán đó vào ứng dụng; ngoài ra luận văn còn xây dựng sơ đồ kiến trúc tổng thể, thiết kế tổng thể và thiết kế chi tiết của ứng dụng

Ứng dụng:

Trên cơ sở nghiên cứu lý thuyết đã xây dựng được một ứng dụng hỗ trợ cho khách du lịch tại Việt Nam. Hiện tại ứng dụng đang trong giai đoạn thử nghiệm để tiếp thu những ý kiến đóng góp của người dùng giúp cho ứng dụng được hoàn thiện hơn.

Hướng nghiên cứu tiếp theo:

Hướng nghiên cứu tiếp theo của luận văn sẽ tập trung vào việc nghiên cứu các phương pháp nâng cao độ chính xác của các thuật toán thu thập dữ liệu người dùng để đưa ra các gợi ý tốt nhất cho người dùng; Nâng cao hiệu quả để ứng dụng chạy nhanh hơn, bổ sung thêm nhiều tính năng mới trong ứng dụng và mở rộng ứng dụng trên nhiều nền tảng khác nhau: Android, Windows Phone.