**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH**



**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**

**HỆ THỐNG QUẢN LÝ CHI PHÍ ĐI LẠI CỦA NHÂN VIÊN BÁN HÀNG TRONG CÔNG TY**

**HỘI ĐỒNG: 3**

**GVHD: ThS. Đặng Trần Trí**

**TS. Phan Trọng Nhân**

**GVPB: ThS. Nguyễn Thanh Tùng**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | MSSV |
| 1 | Đoàn Ngọc Tài | 51203208 |
| 2 | Nguyễn Công Bảy | 51200212 |
| 3 | Phạm Anh Nguyên Hồng | 51201293 |

*TP. HỒ CHÍ MINH, THÁNG 6 NĂM 2017*

# LỜI CAM ĐOAN

Chúng tôi cam đoan rằng các công việc và kết quả được trình bày trong luận văn này là do chính chúng tôi thực hiện, biên soạn và chưa hề được công bố ở bất kỳ đâu. Các thông tin, kết quả tham khảo được chúng tôi thu thập từ các nguồn khác nhau được ghi rõ trong mục Tài liệu tham khảo.

*TP.HCM, ngày 01 tháng 06 năm 2017*

Trân trọng

# LỜI CẢM ƠN

Với lòng biết ơn sâu sắc nhất, chúng tôi xin gửi đến quý Thầy Cô ở Trường Đại Học Bách Khoa – ĐHQG TP. Hồ Chí Minh đã truyền đạt tri thức và tâm huyết của mình cho chúng tôi trong suốt thời gian học tập tại trường.

Chúng tôi xin chân thành cảm ơn Thầy Đặng Trần Trí và Thầy Phan Trọng Nhân đã tận tâm theo sát chúng tôi trong suốt quá trình thực hiện luận văn, đóng góp những ý kiến quý báu cho chúng tôi qua từng buổi gặp mặt xuyên suốt quá trình tìm hiểu và thực hiện đề tài.

Chúng tôi cũng xin chân thành cảm ơn tác giả của các tài liệu mà chúng tôi đã tham khảo trong quá trình thực hiện luận văn.

*TP.HCM, ngày 01 tháng 06 năm 2017*

Trân trọng

# TÓM TẮT //TODO

Phương pháp xác thực là phương pháp kiểm tra xem người dùng đang tương tác với hệ thống có chính xác là chủ nhân thật sự hay không. Có rất nhiều hệ thống hiện nay dùng một chuỗi kí tự bí mật để làm mật khẩu xác thực. Đây là một phương pháp xác thực đơn giản. Tuy nhiên, nó cũng mang một số bất lợi. Thứ nhất, nó yêu cầu người dùng phải nhớ chính xác mật khẩu của mình tạo ra. Thứ hai, với những chuỗi mật khẩu quá đơn giản, ít kí tự thì hệ thống rất dễ bị tấn công. Chính vì vậy, để giải quyết những vấn đề trên, các phương pháp xác thực dùng đặc trưng sinh trắc ra đời. Một trong phương pháp xác thực người dùng rất được quan tâm hiện nay là xác thực dựa trên đặc trưng khuôn mặt.

Khuôn mặt đóng vai trò quan trọng trong quá trình giao tiếp giữa người với người, và cũng mang một lượng thông tin giàu có. Cùng với đó, sự phát triển của thiết bị di động ngày càng rộng rãi và phổ biến, các thông tin cá nhân được lưu trữ trên chúng càng nhiều dẫn đến việc bảo mật trở thành một vấn đề quan trọng. Do đó, xác thực khuôn mặt trên thiết bị di động là đề tài rất có tiềm năng và giúp ích được nhiều trong đời sống con người.

Mục tiêu của đề tài là nghiên cứu, tìm hiểu các cách rút trích đặc trưng gương mặt. Từ đó đánh giá ưu khuyết điểm của từng cách để áp dụng vào việc xác thực trên thiết bị di động. Đồng thời phải nâng cao tính an toàn khi sử dụng sinh trắc học bằng các phương pháp bảo mật.

Hệ thống đề xuất trong đề tài được phát triển trên nền tảng android với sự kết hợp của ba giải thuật PCA, Fisherface và Local binary pattern. Phương pháp bảo mật được tích hợp trong hệ thống là Secure Sketch. Hiệu suất của hệ thống được đo và thử nghiệm trên tập Face94 và hình ảnh thực tế thu được qua camera với kết quả giải thuật Eigenface nhanh nhất về tốc độ, giải thuật Fisherface có độ chính xác cao nhất với tỉ lệ FAR và FRR đều bằng 0% (không sử dụng bảo mật) và 2% (có sử dụng bảo mật)

# ABSTRACT //TODO

Authentication means verifying if the person interacting with the system is truly the device’s owner. Nowadays, there are many systems using a secret string as password, this is the simplest method. However, it has some drawbacks. First of all, users must remember exactly their password. Secondly, if the password is too simple, or contains few characters, the system will be vulnerable. Therefore, in order to solve the addressed problems, biometric authentication systems are born. One of the most concern methods right now is facial recognition.

Our faces play a critical role in human communication, moreover, they carry information about ourselves. And mobile devices have been growing rapidly and steadily for the last couple years, the more information we store on our devices, the stronger the need for security. Therefore, facial recognition on mobile devices is a promising approach that could contribute to human life.

The main objective of this thesis is researching on how to extract facial features, evaluate strengths and weaknesses of each approach to apply for facial recognition on mobile devices. Additionally, we have to keep the biometrics safe by template protection methods.

Our proposed system is based on Android platform with a combination of three algorithms PCA, Fisherface and Local Binary Pattern. The template protection technique used in our system is called Secure Sketch. The experiments are conducted on Faces94 database and photos taken by camera. According to the results, Eigenface gives us the best performance in term of running time, while Fisherface has the most accurate both rate FAR and FRR are 0% (no secure method), 2% (use secure method).

# MỤC LỤC//TODO

[LỜI CAM ĐOAN 2](#_Toc483664442)

[LỜI CẢM ƠN 3](#_Toc483664443)

[TÓM TẮT //TODO 4](#_Toc483664444)

[ABSTRACT //TODO 5](#_Toc483664445)

[MỤC LỤC//TODO 6](#_Toc483664446)

[DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT//TODO 8](#_Toc483664447)

[DANH MỤC CÁC HÌNH//TODO 9](#_Toc483664448)

[1. GIỚI THIỆU ĐỀ TÀI 11](#_Toc483664449)

[1.1. Giới thiệu đề tài 11](#_Toc483664450)

[1.2. Mục tiêu đề tài 11](#_Toc483664451)

[1.3. Phạm vi đề tài 11](#_Toc483664452)

[1.4. Cấu trúc báo cáo //todo 12](#_Toc483664453)

[2. CƠ SỞ LÝ THUYẾT 13](#_Toc483664454)

[2.1. Tổng quan về GPS 13](#_Toc483664455)

[2.1.1. Giới thiệu về GPS 13](#_Toc483664456)

[2.1.2. Cơ chế hoạt động của GPS 13](#_Toc483664457)

[2.1.3. Những hạn chế khi sử dụng GPS của thiết bị di động để định vị bản đồ 14](#_Toc483664458)

[2.2. Tổng quan về hệ điều hành Android 15](#_Toc483664459)

[2.2.1. Giới thiệu về hệ điều hành Android 15](#_Toc483664460)

[2.2.2. Kiến trúc của Android 16](#_Toc483664461)

[2.2.3. Các phiên bản phổ biến của hệ điều hành Android 18](#_Toc483664462)

[3. HỆ THỐNG ĐỀ XUẤT 20](#_Toc483664463)

[3.1. Mô tả hệ thống 20](#_Toc483664464)

[3.2. Website 21](#_Toc483664465)

[3.2.1. Sơ đồ chức năng (Use Case Diagram) 21](#_Toc483664466)

[a. Use case Scenario 23](#_Toc483664467)

[b. Sơ đồ sequence 26](#_Toc483664468)

[4. Các chức năng chính 30](#_Toc483664469)

[5. HIỆN THỰC //TODO 32](#_Toc483664470)

[1. Website 32](#_Toc483664471)

[1.1. Workflow 32](#_Toc483664472)

[1.2. Giao diện Website 32](#_Toc483664473)

[1.2.1. Đăng nhập 32](#_Toc483664474)

[1.2.2. Trang chủ 32](#_Toc483664475)

[c. Chi tiết các khối 39](#_Toc483664476)

[6. Kiểm định sản phẩm 43](#_Toc483664477)

[a. Danh sách Unit Test Case. 43](#_Toc483664478)

[b. Chi tiết Unit Test Case 43](#_Toc483664479)

[c. Báo cáo Unit Test 46](#_Toc483664480)

[7. ĐÁNH GIÁ KẾT QUẢ //TODO 48](#_Toc483664481)

[1. Phương pháp đánh giá 48](#_Toc483664482)

[Tiêu chí đánh giá 48](#_Toc483664483)

[Xây dựng tập thí nghiệm sử dụng để huấn luyện, đăng kí và kiểm tra 48](#_Toc483664484)

[Phương pháp đánh giá thực nghiệm 49](#_Toc483664485)

[Môi trường thí nghiệm 49](#_Toc483664486)

[8. Các thí nghiệm 49](#_Toc483664487)

[- Đánh giá hiệu suất giải thuật PCA trên tập thí nghiệm 49](#_Toc483664488)

[-Đánh giá hiệu suất giải thuật Fisherface trên tập thí nghiệm 50](#_Toc483664489)

[-Đánh giá hiệu suất giải thuật LBP trên tập thí nghiệm 52](#_Toc483664490)

[-Đánh giá hiệu suất kết hợp 3 giải thuật trên tập thí nghiệm 53](#_Toc483664491)

[9. Kết quả thí nghiệm 54](#_Toc483664492)

[10. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 55](#_Toc483664493)

[1. Kết luận 55](#_Toc483664494)

[11. Hướng phát triển 55](#_Toc483664495)

[THAM KHẢO 56](#_Toc483664496)

# DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT//TODO

PCA: Principal component analysis

LDA: Linear discriminant analysis

LBP: Local binary pattern

FLD: Fisher’s linear discriminant

FAR : False accept ratio

FRR: False rejection ratio

OpenCV: Open source computer vision

HMAC: Hash message authentication code

SHA-1: Secure hash algorithm 1

API: Application programming interface

UTCID: Unit testcase identify

VTR: Vector riêng

GTR: Trị riêng

# DANH MỤC CÁC HÌNH//TODO

[Hình 1. Các loại sinh trắc học phổ biến 14](#_Toc470817981)

[Hình 2. Xác thực bằng giấy phép lái xe hoặc hộ chiếu 15](#_Toc470817982)

[Hình 3. Nhận dạng bằng password 15](#_Toc470817983)

[Hình 4. Nhận dạng bằng mã PIN, khó để ghi nhớ 16](#_Toc470817984)

[Hình 5. Hồ sơ sinh trắc học của một người 16](#_Toc470817985)

[Hình 6. Tương tác con người và máy tính 17](#_Toc470817986)

[Hình 7. Biểu cảm khuôn mặt 1 17](#_Toc470817987)

[Hình 8. Biểu cảm khuôn mặt 2 18](#_Toc470817988)

[Hình 9. Ví dụ về phương pháp PCA 21](#_Toc470817989)

[Hình 10. Biểu diễn khuôn mặt thành vector 22](#_Toc470817990)

[Hình 11. Hình minh họa giải thuật PCA 24](#_Toc470817991)

[Hình 12. Sự kết hợp tuyến tính của khuôn mặt 24](#_Toc470817992)

[Hình 13. Hình ảnh so sánh PCA và FLD trong bài toán hai lớp, nơi dữ liệu cho mỗi lớp nằm gần một không gian con tuyến tính 27](#_Toc470817993)

[Hình 14. Minh họa giải thuật LBP 29](#_Toc470817994)

[Hình 15. Ví dụ về quá trình tính toán đặc trưng LBP 31](#_Toc470817995)

[Hình 16. Mô hình Secure Sketch 32](#_Toc470817996)

[Hình 17. Mô hình Codebook Secure Sketch 33](#_Toc470817997)

[Hình 18. Quá trình sinh sketch và khôi phục 35](#_Toc470817998)

[Hình 19. Tập huấn luyện Eigenface 37](#_Toc470817999)

[Hình 20. Khuôn mặt trung bình 37](#_Toc470818000)

[Hình 21. Các eigen vector của tập huấn luyện 38](#_Toc470818001)

[Hình 22. Hình ảnh người cần xác thực 39](#_Toc470818002)

[Hình 23. Tập train Fisherface 40](#_Toc470818003)

[Hình 24. Khuôn mặt trung bình 40](#_Toc470818004)

[Hình 25. Các vector sau khi train Fisherface 41](#_Toc470818005)

[Hình 26. Testcase 41](#_Toc470818006)

[Hình 27. Tập huấn luyện của LBP 42](#_Toc470818007)

[Hình 28. Ảnh xám của tập huấn luyện 42](#_Toc470818008)

[Hình 29. Ảnh LBP thu được sau khi tính toán 43](#_Toc470818009)

[Hình 30. Biểu đồ minh họa Histogram 43](#_Toc470818010)

[Hình 31. Ảnh cần xác thực 44](#_Toc470818011)

[Hình 32. Ảnh xác thực đã chuyển qua ảnh xám 44](#_Toc470818012)

[Hình 33. Ảnh xác thực LBP 45](#_Toc470818013)

[Hình 34. Hệ thống đề xuất 47](#_Toc470818014)

[Hình 35. Chi tiết hệ thống 48](#_Toc470818015)

[Hình 36. Hình mô tả kết hợp 3 giải thuật 49](#_Toc470818016)

[Hình 37. Mô hình usecase của hệ thống 49](#_Toc470818017)

[Hình 38. Sequence đăng ký 53](#_Toc470818018)

[Hình 39. Sequence cập nhật sinh trắc 53](#_Toc470818019)

[Hình 40. Sequence xem hướng dẫn 54](#_Toc470818020)

[Hình 41. Sequence xác thực 55](#_Toc470818021)

[Hình 42. Sequence tùy chỉnh 55](#_Toc470818022)

[Hình 43. Quá trình đăng ký dùng Secure Sketch 56](file:///C:\Users\dohoa\AndroidStudioProjects\wowauthentication\document\PHÁT-TRIỂN-HỆ-THỐNG-XÁC-THỰC-BẰNG-ĐẶC-TRƯNG-GƯƠNG-MẶT-TRÊN-ANDROID-CÓ-BẢO-VỆ-MẪU-ĐẶC-TRƯNG-SINH-TRẮC.docx#_Toc470818023)

[Hình 44. Quá trình xác thực dùng Secure Sketch 57](#_Toc470818024)

[Hình 45. Kiến trúc của Android 59](#_Toc470818025)

[Hình 46. Sơ đồ miêu tả sự phổ biến của các phiên bản android 61](#_Toc470818026)

[Hình 47. Tỷ lệ kết quả theo testcase 70](#_Toc470818027)

[Hình 48. Số loại testcase 70](#_Toc470818028)

[Hình 49. Cấu trúc tập thí nghiệm 71](#_Toc470818029)

[Hình 50. Độ chính xác của PCA với các ngưỡng 73](#_Toc470818030)

[Hình 51. Độ chính xác của PCA với các DELTA 73](#_Toc470818031)

[Hình 52. Độ chính xác của Fisherface với các ngưỡng 74](#_Toc470818032)

[Hình 53. Độ chính xác của Fisherface với các DELTA 74](#_Toc470818033)

[Hình 54. Độ chính xác của LBP với các ngưỡng 75](#_Toc470818034)

[Hình 55. Độ chính xác của LBP với các DELTA 76](#_Toc470818035)

# GIỚI THIỆU ĐỀ TÀI

## Giới thiệu đề tài

Ngày nay, các công ty đều có chính sách hỗ trợ nhân viên chi phí đi lại từ nơi ở tới công ty, từ công ty tới gặp khách hàng, … nhất là đối với các nhân viên bán hàng, họ rất thường xuyên di chuyển do nhu cầu công việc. Hiện tại, việc quản lý chi phí đi lại của nhân viên thường thông qua các số liệu do nhân viên cung cấp (hóa đơn, vé xe, do nhân viên tự khai báo, …) mà chưa được quản lý một cách hệ thống, tự động.

Các thiết bị di động thông minh (Smartphone) đang được sử dụng ngày càng phổ biến và rộng rãi. Ngoài các tính năng nghe gọi, nhắn tin, Smartphone còn có khả năng định vị với độ chính xác cao. Do đó, các dịch vụ dựa trên vị trí (Location-based service) ngày càng phát triển mạnh.

Đề tài này được phát triển để hỗ trợ công ty quản lý chi phí đi lại của nhân viên trong các hoạt động hàng ngày. Dữ liệu đi lại của nhân viên sẽ được thu thập và lưu trữ. Dữ liệu này sẽ cho phép công ty tính toán được chi phí đi lại của các nhân viên để đưa ra mức hỗ trợ phù hợp.

## Mục tiêu đề tài

Các mục tiêu của đề tài:

* Định vị và quản lý lịch trình của nhân viên:
* Định vị tự động dùng Smartphone.
* Tính toán chi phí đi lại của nhân viên.
* Bảo vệ tính riêng tư về vị trí, đường đi:
* Bảo vệ tính riêng tư về vị trí.
* Quản lý truy xuất đến dữ liệu lịch sử di chuyển.

Hệ thống gồm các thành phần:

* Một ứng dụng di động (trên nền tảng Android) dành cho nhân viên, để thu thập dữ liệu về vị trí, lộ trình, chi phí đi lại của nhân viên.
* Một Website hỗ trợ những người quản trị hệ thống, để quản lý dữ liệu của công ty.

## Phạm vi đề tài

Phạm vi nghiên cứu của đề tài:

* Đề tài này được phát triển dựa trên giả định là nhân viên trung thực, nghĩa là các thông tin được nhân viên cung cấp thông qua ứng dụng di động là trung thực, không sử dụng những phần mềm làm sai lệch các thông tin đó nên đề tài bỏ qua việc chống gian lận vị trí.
* Công thức tính toán chi phí đi lại sẽ phụ thuộc vào chính sách của mỗi công ty. Do đó, đề tài sẽ không nghiên cứu, sử dụng một công thức tính toán cụ thể nào.

## Cấu trúc báo cáo //todo

Báo cáo gồm có 6 chương với các nội dung:

* Chương I: Giới thiệu đề tài: Trình bày tổng quát về đề tài tính cần thiết và tiềm năng ứng dụng của nó; Làm rõ mục tiêu và phạm vi đề tài; Phương pháp nghiên cứu và hiện thực của nhóm; Cấu trúc báo cáo.
* Chương II: Cơ sở lý thuyết: Giới thiệu về sinh trắc học, ứng dụng của sinh trắc học và các tiêu chí đánh giá một sinh trắc học; Lý thuyết về các phương pháp rút trích Principal component analysis, Fisherface, Local binary pattern. Phương pháp bảo mật mẫu sau khi đã rút trích được đặc trưng; Phương pháp huấn luyện và cách xác thực một khuôn mặt đầu vào.
* Chương III: Hệ thống đề xuất: Mô tả về hệ thống của nhóm với các sơ đồ usecase, activity. Chi tiết về các chức năng chính của phần mềm.
* Chương IV: Hiện thực: Giới thiệu về nền tảng android, các thư viện nhóm sử dụng. Các khối chức năng chính và việc kiểm định sản phẩm.
* Chương V: Đánh giá kết quả: Trình bày các phương pháp đánh giá, các thí nghiệm sử dụng để đánh giá và kết quả thí nghiệm.
* Chương VI: Kết luận và hướng phát triển

# CƠ SỞ LÝ THUYẾT

## Tổng quan về GPS

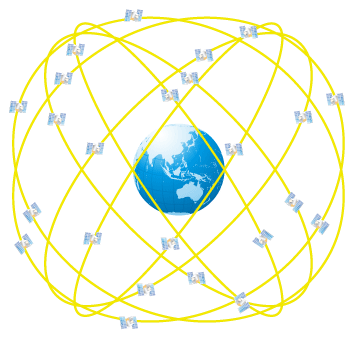
* + 1. **Giới thiệu về GPS**

GPS (viết tắt từ Global Positioning System – Hệ thống định vị toàn cầu) là hệ thống xác định vị trí toàn cầu được tài trợ và quản lý bởi Bộ Quốc phòng Mỹ (U. S. Department of Defense). Đây là dịch vụ định vị toàn cầu được cung cấp miễn phí, có khả năng cung cấp thông tin cần thiết để xác định vị trí của người dùng. [1]

* + 1. **Cơ chế hoạt động của GPS**

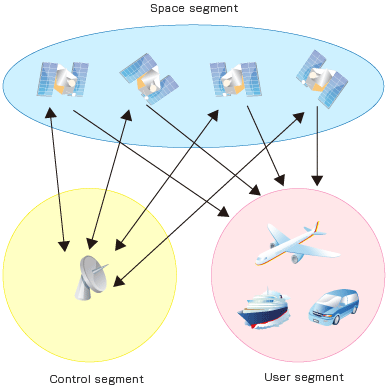
GPS bao gồm 3 thành phần: [3]

* Các vệ tinh GPS (Space segment): 24 vệ tinh GPS đang hoạt động được triển khai trên 6 quỹ đạo (khoảng 4 vệ tinh trên mỗi quỹ đạo) quay quanh trái đất với chu kỳ quỹ đạo khoảng 12 giờ ở độ cao khoảng 20 000 km so với mực nước biển.



Hình 2.1.2.1. Quỹ đạo của các vệ tinh GPS [2]

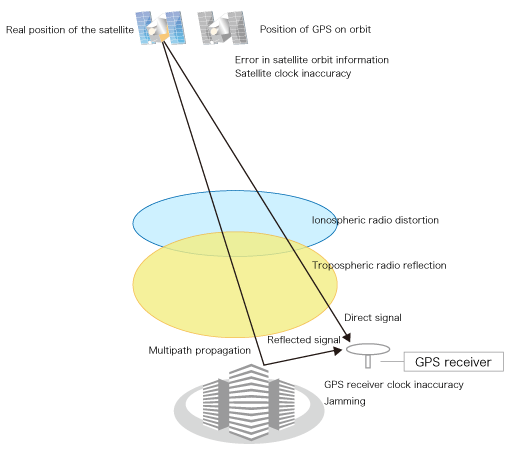
* Trạm điều khiển mặt đất (Control segment): Trạm điều khiển mặt đất đóng vai trò điều khiển, vận hành và duy trì hoạt động của các vệ tinh, đảm bảo chất lượng của tín hiệu GPS.
* Thiết bị thu GPS của người dùng (User segment): Thiết bị thu GPS của người dùng thu tín hiệu từ các vệ tinh GPS và chuyển đổi chúng thành những dữ liệu để ước tính vị trí người dùng. Cần kết hợp tín hiệu nhận được từ 4 vệ tinh GPS để tính toán vị trí người dùng (kinh độ, vĩ độ, cao độ và thời gian (time offset)).



Hình 2.1.2.2. Cấu trúc cơ bản của GPS [2]

* + 1. **Những hạn chế khi sử dụng GPS của thiết bị di động để định vị bản đồ**

Do nhiều yếu tố chủ quan và khách quan, sử dụng GPS của thiết bị di động để định vị bản đồ luôn có sai số không thể tránh khỏi (khoảng 1.4–7.1m ở những điều kiện bình thường). [3] Điều này ảnh hưởng đến độ chính xác của dữ liệu về vị trí và lộ trình của người dùng.



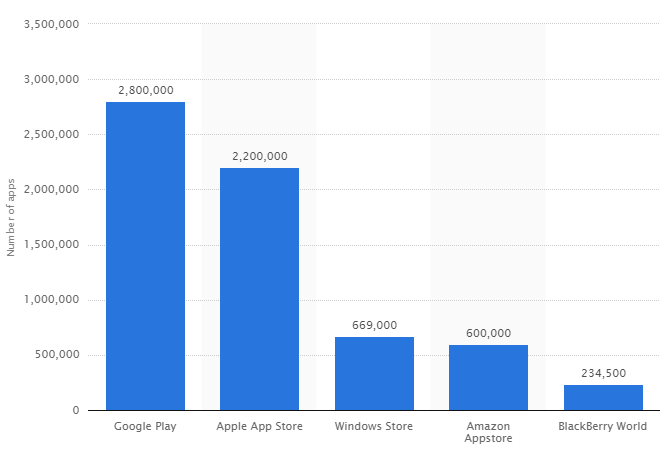
Hình 2.1.3.1. Một số yếu tố ảnh hưởng chất lượng định vị của GPS [2]

* 1. **Tổng quan về hệ điều hành Android**
     1. **Giới thiệu về hệ điều hành Android**

Android [4] là một hệ điều hành dựa trên nền tảng Linux được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh và máy tính bảng.

Ban đầu, Android được phát triển bởi công ty Android (Android, Inc.). Công ty Android được thành lập tại Palo Alto, California vào tháng 10 năm 2003 bởi Andy Rubin, Rich Miner, Nick Sears và Chris White để phát triển một hệ điều hành dành cho các thiết bị di động có thể nhận biết được ví trị và sở thích của người dùng. Google mua lại công ty Android vào ngày 17 tháng 8 năm 2005 và biến nó thành một bộ phận trực thuộc Google. Tại Google, nhóm do Rubin đứng đầu đã phát triển một nền tảng dành cho thiết bị di động trên nền nhân Linux. Hệ điều hành Android được ra mắt vào năm 2007. Năm 2008, chiếc điện thoại đầu tiên chạy hệ điều hành Android, HTC Dream, được bán ra.

Android là hệ điều hành mã nguồn mở, phát hành theo giấy phép Apache, một giấy phép không có nhiều ràng buộc. Điều này cho phép các nhà phát triển, các lập trình viên được tự do tinh chỉnh và phân phối Android. Bên cạnh đó, Android còn có một cộng đồng lập trình viên đông đảo, nhiệt huyết, chuyên viết các ứng dụng cho hệ điều hành này bằng ngôn ngữ lập trình Java. Theo thống kê từ Statista, đã có khoảng 2.8 triệu ứng dụng (tính đến tháng 3 năm 2017) [5] được phát hành trên Google Play, cửa hàng ứng dụng chính của Android, với 65 tỷ lượt tải về (từ tháng 8 năm 2010 đến tháng 5 năm 2016). [6]



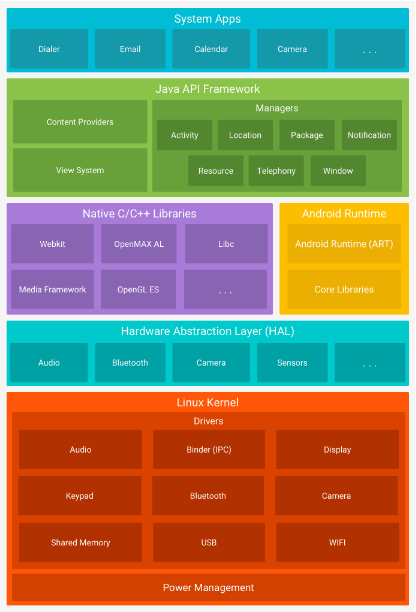
Hình 2.2.1.1. Thống kê số ứng dụng được phát hành trên các cửa hàng ứng dụng hàng đầu (tính đến tháng 3 năm 2017) [5]

Những yếu tố này đã giúp Android trở thành hệ điều hành điện thoại thông minh phổ biến nhất thế giới, và được các công ty công nghệ lựa chọn khi họ cần một hệ điều hành không nặng nề, có khả năng tinh chỉnh và giá rẻ chạy trên các thiết bị công nghệ cao thay vì tạo dựng từ đầu. Kết quả là mặc dù được thiết kế để chạy trên điện thoại và máy tính bảng, Android đã xuất hiện trên TV, máy chơi game và các thiết bị điện tử khác.

* + 1. **Kiến trúc của Android**

Kiến trúc của hệ điều hành Android bao gồm: [7]

* Linux Kernel : được dựa trên nhân Linux, là một thành phần cực kỳ quan trọng.
* Hardware Abstraction Layer (HAL): giúp các ứng dụng Android có thể tương tác với các thiết bị phần cứng.
* Native C/C++ Libraries: bao gồm các thư viện (viết bằng C/C++) hỗ trợ cho các ứng dụng thực hiện các chức năng mở rộng.
* Android Runtime: đảm nhận công việc biên dịch các tập tin ứng dụng, để chúng có thể chạy được trên nền tảng Android.
  + Core Libraries : tập các thư viện cơ bản của Android .
  + Dalvik Vitual Machine : bộ máy ảo của android, là nơi thực hiện các ứng dụng mà người dùng sử dụng trên thiết bị Android.
* Java API Framework: cung cấp toàn bộ những tính năng cần thiết của hệ điều hành Android cho lập trình viên sử dụng thông qua các API được viết bằng ngôn ngữ Java.
* System Apps: là tập hợp các ứng dụng và chức năng hệ thống cơ bản để tương tác với người sử dụng, hoặc để các lập trình viên phát triển ứng dụng.



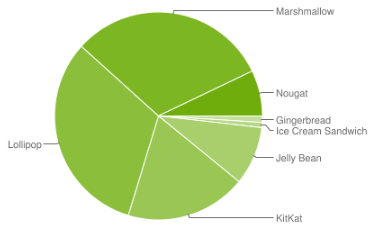
Hình 2.2.2.1. Sơ đồ kiến trúc của hệ điều hành Android [7]

* + 1. **Các phiên bản phổ biến của hệ điều hành Android**

Bảng sau đây sẽ liệt kê các phiên bản phổ biến nhất của hệ điều hành Android. Bảng sẽ không liệt kê những phiên bản chiếm ít hơn 0.1% thị phần của các thiết bị chạy hệ điều hành Android tính tới thời điểm được khảo sát (ngày 2 tháng 5 năm 2017).

| Phiên bản | Tên phiên bản | API level | Phân bố |
| --- | --- | --- | --- |
| 2.3.3 – 2.3.7 | Gingerbread | 10 | 1.0% |
| 4.0.3 – 4.0.4 | Ice Cream Sandwich | 15 | 0.8% |
| 4.1.x | Jelly Bean | 16 | 3.2% |
| 4.2.x | 17 | 4.6% |
| 4.3 | 18 | 1.3% |
| 4.4 | KitKat | 19 | 18.8% |
| 5.0 | Lollipop | 21 | 8.7% |
| 5.1 | 22 | 23.3% |
| 6.0 | Marshmallow | 23 | 31.2% |
| 7.0 | Nougat | 24 | 6.6% |
| 7.1 | 25 | 0.5% |

Bảng 2.2.3.1. Bảng liệt kê các phiên bản phổ biến của hệ điều hành Android [8]



Hình 2.2.3.2. Các phiên bản phổ biến của hệ điều hành Android

Nhóm quyết định chọn phiên bản Jelly Bean (API level: 16) là phiên bản thấp nhất được hệ thống quản lý chi phí đi lại của nhân viên bán hàng trong công ty hỗ trợ để cho phép được nhiều người dùng có thể sử dụng được hệ thống.

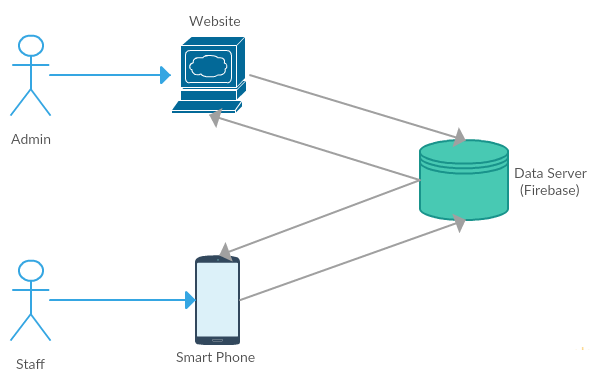
# HỆ THỐNG ĐỀ XUẤT

## Mô tả hệ thống

Hệ thống quản lý chi phí đi lại của nhân viên bán hàng trong công ty bao gồm một ứng dụng di động (trên nền tảng Android) dành cho nhân viên, để thu thập dữ liệu về vị trí, lộ trình, chi phí đi lại của nhân viên và một Website (trên nền tảng Web) hỗ trợ những người quản trị hệ thống, để quản lý dữ liệu của công ty (dữ liệu về thông tin nhân viên và vị trí, lộ trình, chi phí đi lại của nhân viên).

Website được những người quản trị hệ thống sử dụng để quản lý dữ liệu của công ty. Người quản trị sau khi đăng nhập có thể xem được danh sách toàn bộ nhân viên trong công ty và thống kê cá nhân của từng người (chi phí đi lại và quãng đường đi lại). Khi có nhân viên mới vào, người quản trị cần tạo tài khoản người dùng cho nhân viên đó dựa trên thông tin cá nhân của nhân viên cung cấp. Khi nhân viên nghỉ việc, người quản trị sẽ xóa dữ liệu người đó ra khỏi hệ thống (tài khoản đăng nhập, thông tin người dùng và các dữ liệu về vị trí, lộ trình, chi phí đi lại liên quan). Người quản trị có thể xem thống kê về tổng quãng đường và tổng chi phí đi lại của tất cả nhân viên công ty trong khoảng thời gian mong muốn (thống kê công ty). Tùy thuộc vào chính sách hỗ trợ của công ty, người quản trị sẽ nhập công thức tính toán chi phí cho từng phương tiện (là hàm số theo quãng đường đi được s và thời gian đi được t). Đây là những công thức được áp dụng trên ứng dụng di động của nhân viên để tính toán chi phí đi lại của nhân viên.

Ứng dụng di động được nhân viên cài đặt trên Smartphone cá nhân (hoặc Smartphone do công ty cấp). Nhân viên mỗi khi đi làm sẽ thực hiện đăng nhập vào hệ thống, lựa chọn phương tiện di chuyển và bắt đầu di chuyển. Hệ thống định vị tự động vị trí của nhân viên, vẽ lại lộ trình di chuyển của nhân viên, thu thập dữ liệu về quãng đường đi được và chi phí đi lại của nhân viên. Cứ sau 3 giây, hệ thống sẽ truy vấn vị trí của nhân viên một lần, nếu vị trí của nhân viên thay đổi so với vị trí lần trước (3 mét so với vị trí trước đó), hệ thống mới cập nhật vị trí cho nhân viên. Trước khi vị trí được lưu trên Data Server, dữ liệu vị trí sẽ được xử lý thông qua Web service Google Maps Snap to Roads (cứ nhận được 5 vị trí thì xử lý 1 lần) để giảm thiểu sai số do GPS gây ra. [3] Mỗi khi nhân viên thay đổi phương tiện di chuyển, nhân viên sẽ thực hiện lựa chọn thay đổi phương tiện trên ứng dụng. Đối với các phương tiện công cộng có sử dụng vé cho mỗi chuyến đi, nhân viên cần nhập giá vé và chụp hình vé đối chứng. Nhân viên có thể xem thống kê về chi phí đi lại và quãng đường đi lại của bản thân trong khoảng thời gian mong muốn.

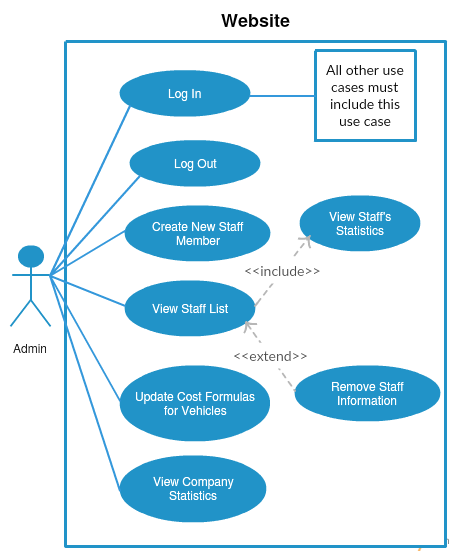


Hình 3.1.1. Tổng quan về hệ thống

* 1. **Website**
     1. **Sơ đồ chức năng (Use Case Diagram)**

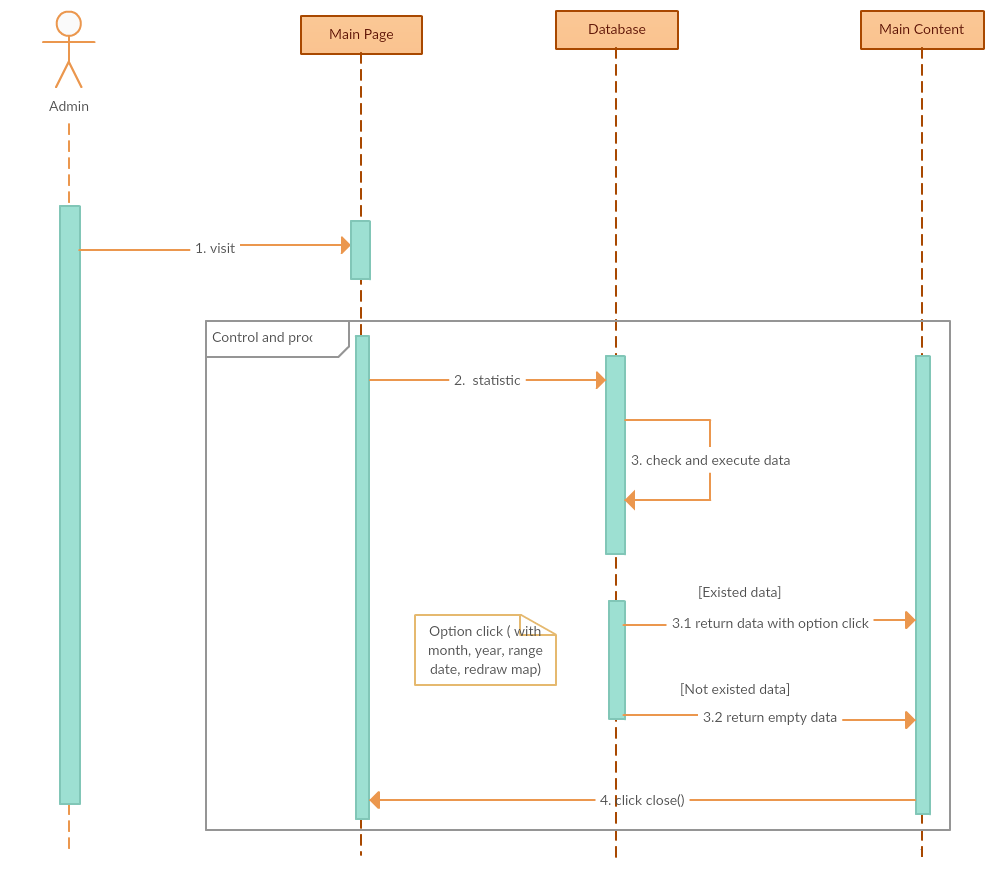
Website dành cho những người quản trị hệ thống gồm có 4 chức năng chính:

* Xem thống kê cá nhân: người quản trị có thể xem thông kế bằng số liệu và biểu đồ về chi phí và quãng đường đi lại của từng nhân viên trong khoảng thời gian mong muốn. Ngoài ra, người quản trị còn có thể xem lại lộ trình của nhân viên trong một ngày cụ thể.
* Tạo tài khoản mới: người quản trị sẽ tạo tài khoản người dùng mới cho nhân viên mới vào dựa trên thông tin cá nhân mà nhân viên cung cấp (họ tên, số điện thoại, email).
* Thiết lập công thức tính toán chi phí: người quản trị hệ thống sẽ nhập công thức tính toán chi phí đi lại riêng (là hàm số theo quãng đường đi được s và thời gian đi được t) cho từng phường tiện tùy theo chính sách hỗ trợ của công ty.
* Xem thống kê công ty: người quản trị có thể xem thông kế bằng số liệu và biểu đồ về chi phí và quãng đường đi lại của tất cả các nhân viên của công ty trong khoảng thời gian mong.



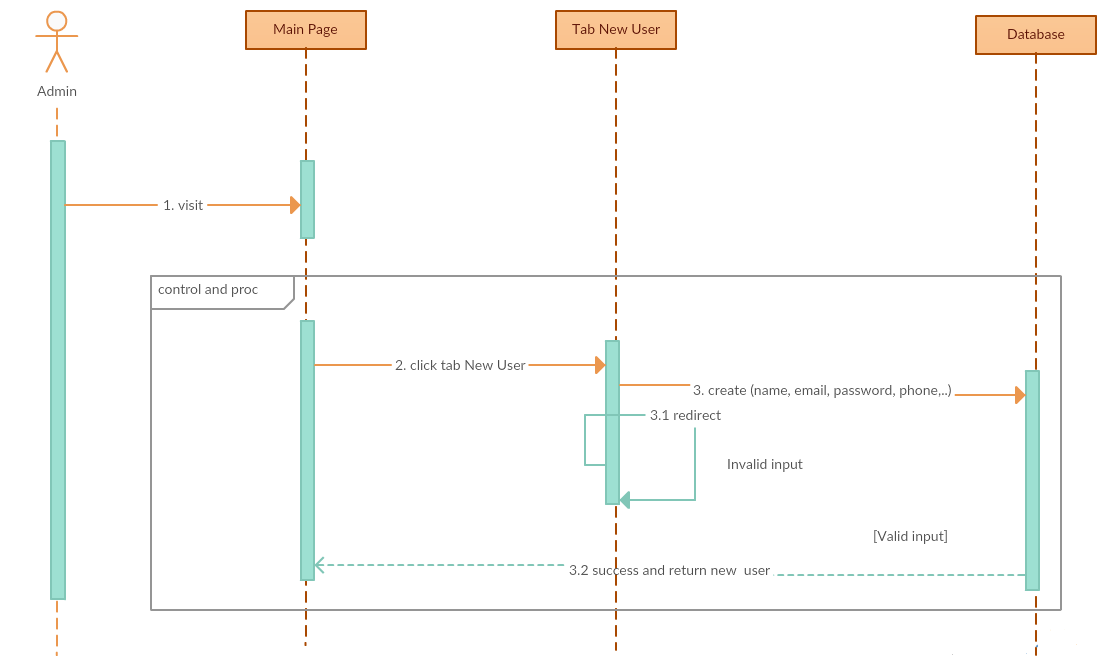
Hình 3.2.1.1. Sơ đồ chức năng của Website

* + 1. **Sơ đồ tuần tự (Sequence Diagram)**
       1. **Chức năng Xem thống kê cá nhân**

****

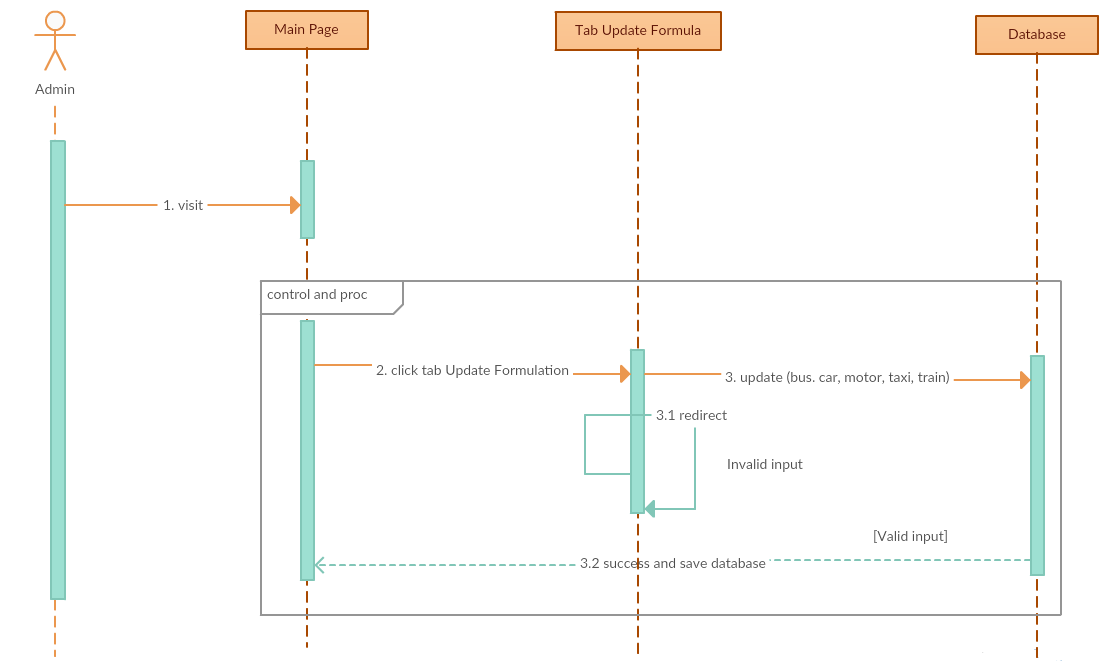
Hình 3.2.2.1.1. Sơ đồ tuần tự chức năng Xem thống kê cá nhân

* + - 1. **Chức năng Tạo tài khoản mới**

****

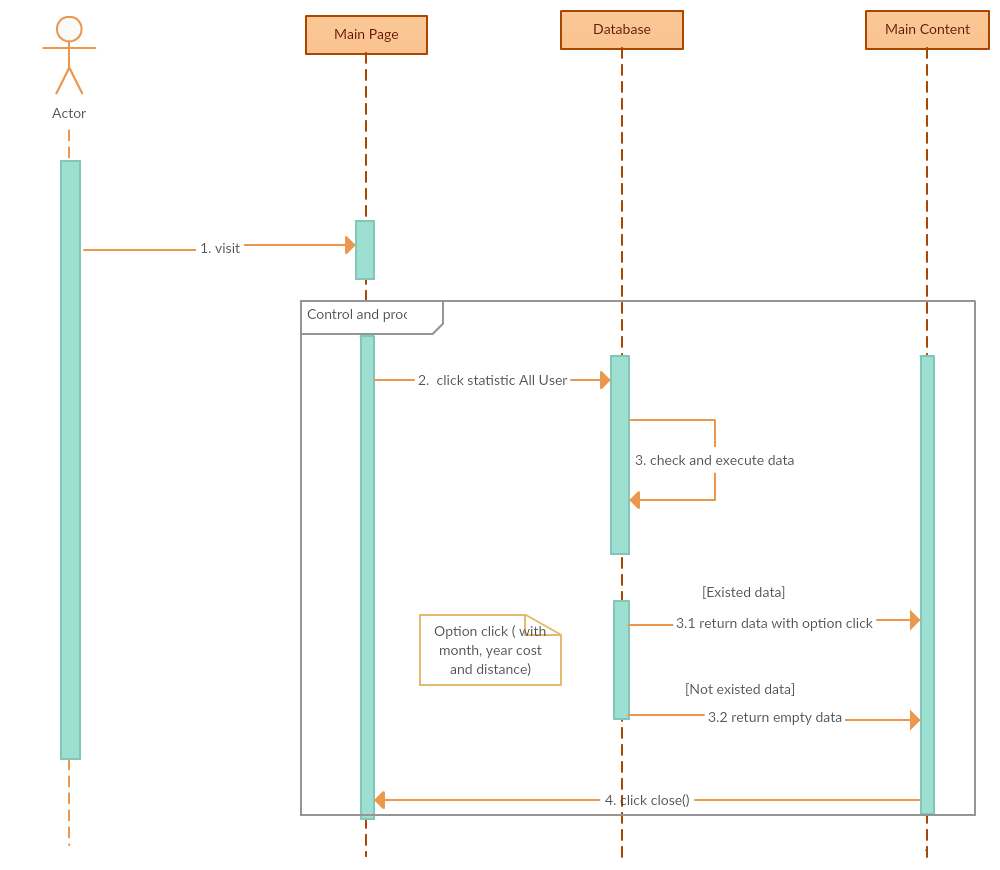
Hình 3.2.2.2.1. Sơ đồ tuần tự chức năng Tạo người dùng mới

* + - 1. **Chức năng Thiết lập công thức tính toán chi phí**

****

Hình 3.2.2.3.1. Sơ đồ tuần tự chức năng Thiết lập công thức tính toán chi phí

* + - 1. **Chức năng Xem thống kê công ty**

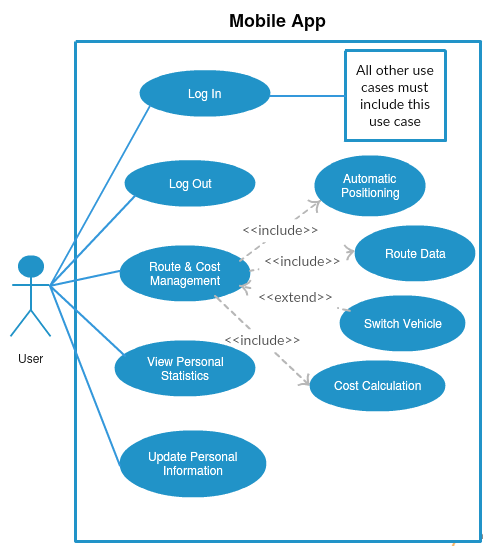
****

Hình 3.2.2.4.1. Sơ đồ tuần tự chức năng Xem thống kê cá nhân

* 1. **Ứng dụng di động**
     1. **Sơ đồ chức năng (Use Case Diagram)**

Ứng dụng di động dành cho nhân viên của công ty có 2 chức năng chính:

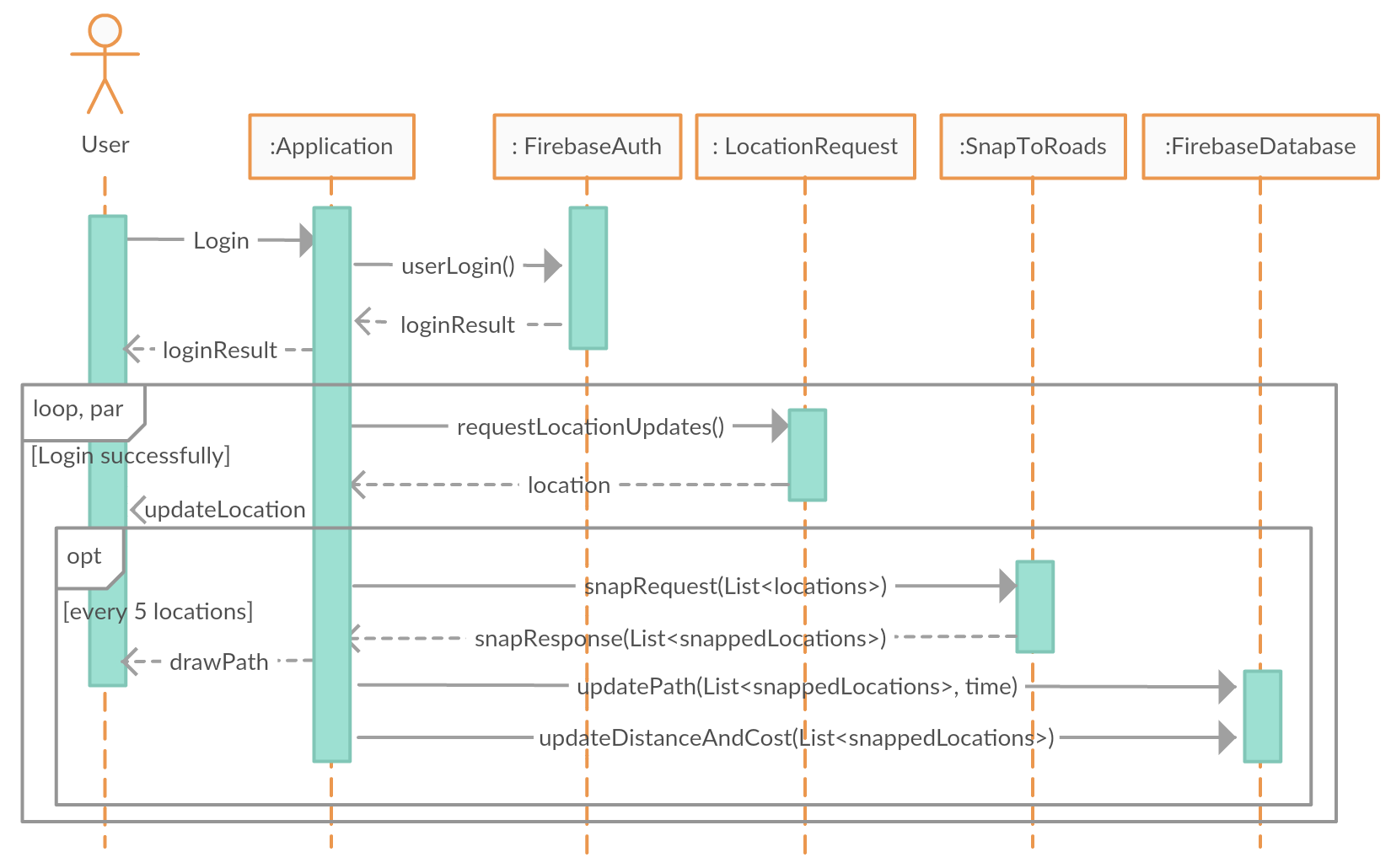
* Quản lý lộ trình, chi phí đi lại: hệ thống định vị tự động vị trí của nhân viên, vẽ lộ trình di chuyển của nhân viên theo phương tiện, thu thập dữ liệu về quãng đường đi được và chi phí đi lại của nhân viên (chi phí được tính theo công thức tính toán do người quản trị thiết lập).
* Xem thống kê cá nhân: nhân viên có thể xem thông kế bằng biểu đồ về chi phí và quãng đường đi lại của bản thân trong khoảng thời gian mong muốn. Ngoài ra, nhân viên còn có thể xem lại lộ trình của bản thân trong một ngày cụ thể.



Hình 3.3.1. Sơ đồ chức năng của Ứng dụng di động

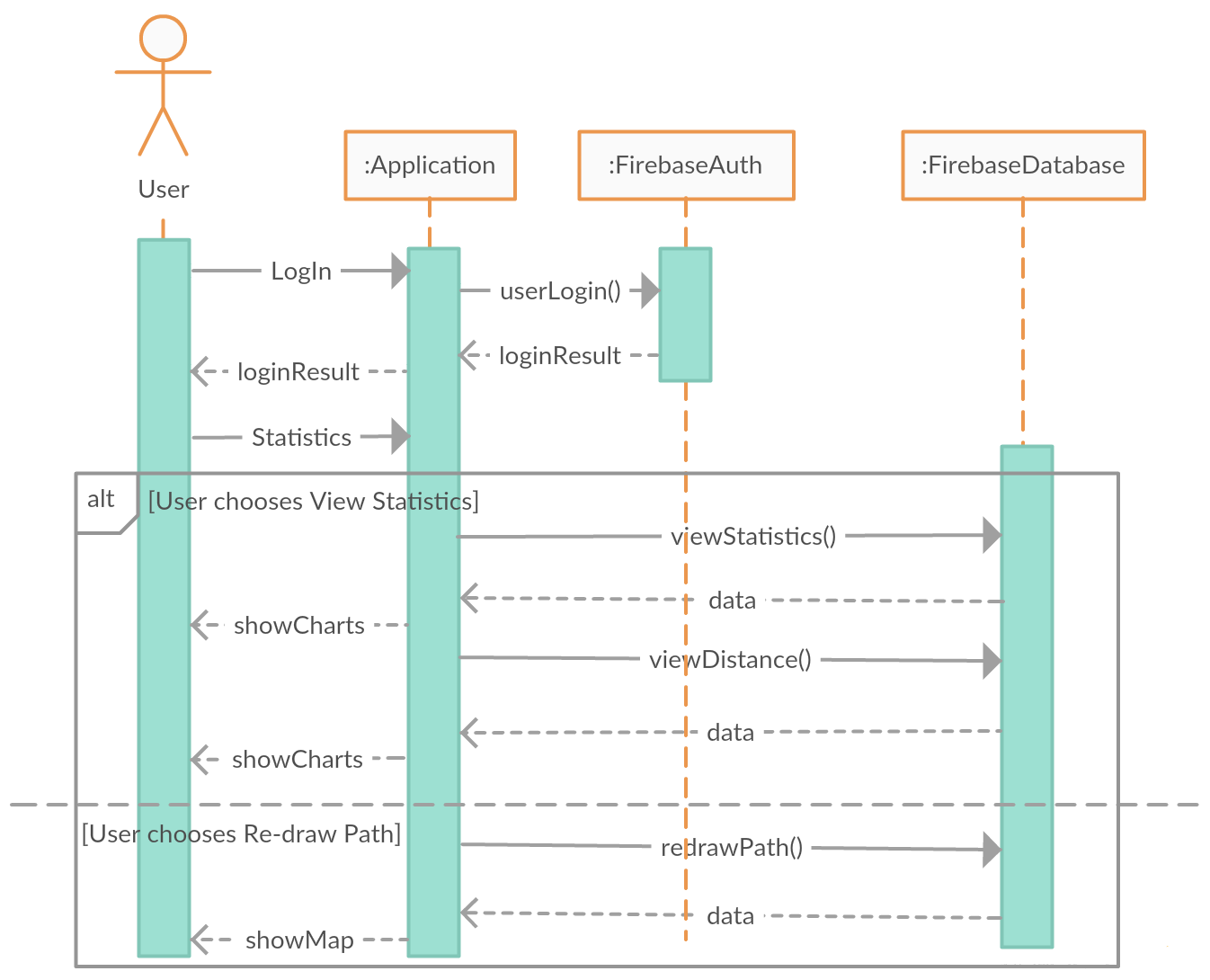
* + 1. **Sơ đồ tuần tự (Sequence Diagram)**

#### Chức năng Quản lý lộ trình, chi phí đi lại



Hình 3.3.2.1.1. Sơ đồ tuần tự chức năng Quản lý lộ trình, chi phí đi lại

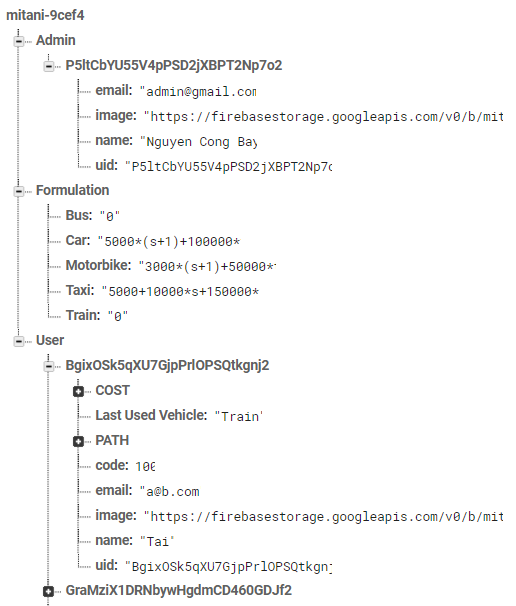
* + - 1. **Chức năng Xem thống kê cá nhân**

****

Hình 3.3.2.2.1. Sơ đồ tuần tự chức năng Xem thống kê cá nhân

* 1. **Cấu trúc cơ sở dữ liệu**
     1. **Tổng quan về cấu trúc cơ sở dữ liệu của hệ thống**

Hệ thống quản lý chi phí đi lại của nhân viên bán hàng trong công ty sử dụng dịch vụ Firebase của Google. Cơ sở dữ liệu của hệ thống bao gồm 1 node duy nhất (node “mitani-9cef4”), node này có tên được đặt dựa theo tên của công ty mà nhóm được giao đề tài. Tất cả các node con được tạo ra sau đó đều bắt nguồn từ node gốc này. Có 3 node con chính: Admin, Formulation và User.

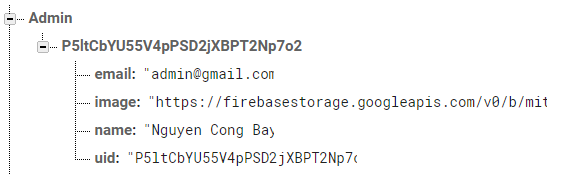


Hình 3.4.1.1. Tổng quan về cấu trúc dữ liệu của hệ thống

* + 1. **Node “Admin”**

Node “Admin” lưu trữ thông tin của toàn bộ tài khoản của người quản trị hệ thống, các tài khoản này thường được các công ty giao cho bộ phận quản lý nhân sự. Bộ phận này sẽ làm nhiệm vụ thêm tài khoản khi có nhân viên mới tới, xóa tài khoản khi có nhân viên rời đi, thiết lập công thức tính toán chi phí theo chí sách của của công ty. Theo sau node “Admin” là các node con được đặt tên bằng UID, là mã định danh người dùng, được tạo tự động từ hệ thống (ví dụ: [*P5ltCbYU55V4pPSD2jXBPT2Np7o2*](https://console.firebase.google.com/project/mitani-9cef4/database/data/Admin/P5ltCbYU55V4pPSD2jXBPT2Np7o2)). Mỗi node con này sẽ chứa thông tin của một tài khoản của người quản trị hệ thống khác nhau, bao gồm các node:

* “email”: địa chỉ e-mail dùng để đăng nhập.
* “image”: hình ảnh đại diện, hiển thị khi đăng nhập vào hệ thống, được lưu trữ dưới dạng đường dẫn đến tập tin hình ảnh được lưu trữ ở File Storage trên Data Server.
* “name”: họ và tên của chủ tài khoản.
* “uid”: UID của tài khoản, được tạo tự động từ hệ thống.

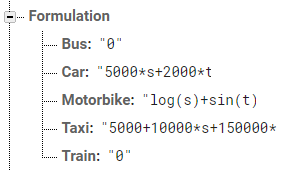


Hình 3.4.2.1. Cấu trúc node “Admin”

* + 1. **Node “Formulation”**

Node “Formulation” chứa công thức tính toán chi phí đi lại cho tất cả các loại phương tiện hiện có trong hệ thống, theo sau là các node con được đặt tên theo phương tiện, bao gồm: “Bus” (xe buýt), “Car” (xe ô tô), “Motorbike” (xe máy), “Taxi” (xe taxi), “Train” (tàu hỏa), có giá trị là các công thức tính toán chi phí dành cho phương tiện đó. Mỗi công thức có thể bao gồm:

* Các hằng số: hỗ trợ kiểu số thực.
* Các biến số: hỗ trợ 2 biến số chính:
* s (tính theo km): quãng đường mà phương tiện đó đã đi.
* t (tính theo giờ): thời gian mà nhân viên sử dụng phương tiện đó.
* Các toán tử: cộng (+), trừ (–), nhân (\*), chia (/), lũy thừa (^), modulo (%).
* Các hàm tính toán: abs (giá trị tuyện đối), cbrt (lập phương), ceil (giới hạn trên), cos (cos của một góc), sin (sin của một góc), log (logarit tự nhiên – logarit cơ số e), sqrt (căn bậc 2 của một số), …

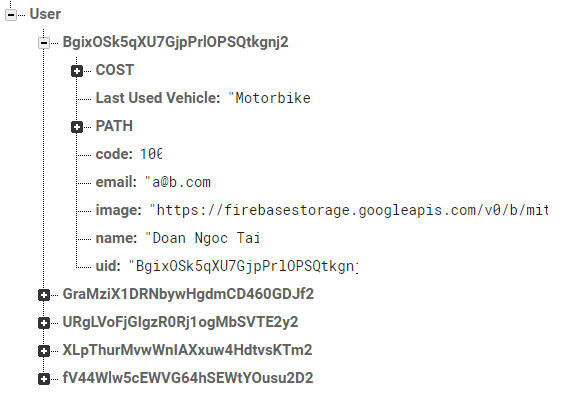


Hình 3.4.3.1. Cấu trúc node “Formulation”

* + 1. **Node “User”**

Node “User” lưu trữ thông tin của toàn bộ nhân viên trong công ty (người dùng của hệ thống). Theo sau node “User” là các node con được đặt tên bằng UID, là mã định danh người dùng, được tạo tự động từ hệ thống (ví dụ: [*P5ltCbYU55V4pPSD2jXBPT2Np7o2*](https://console.firebase.google.com/project/mitani-9cef4/database/data/Admin/P5ltCbYU55V4pPSD2jXBPT2Np7o2)). Mỗi node con này chứa thông tin của một tài khoản khác nhau. Thông tin của một tài khoản bao gồm các node:

* “Last Used Vehicle”: loại phương tiện gần nhất được sử dụng, node này sẽ được khởi tạo khi người dùng đăng nhập lần đầu tiên vào hệ thống và được gán giá trị khởi tạo là “Motorbike”.
* “code”: mã định danh riêng của người dùng, đơn giản hơn UID, được sử dụng trong thống kê toàn hệ thống thay cho UID.
* “email”: địa chỉ e-mail dùng để đăng nhập.
* “image”: hình ảnh đại diện, hiển thị khi đăng nhập vào hệ thống, được lưu trữ dưới dạng đường dẫn đến tập tin hình ảnh được lưu trữ ở File Storage trên Data Server.
* “name”: họ và tên của chủ tài khoản.
* “uid”: UID của tài khoản, được tạo tự động từ hệ thống.
* “COST”: lưu trữ thông tin về chi phí đi lại, được khởi tạo tự động khi người dùng đăng nhập vào hệ thống lần đầu.
* “PATH”: lưu trữ thông tin về quãng đường đã đi và danh sách các tọa độ đã đi qua, được khởi tạo tự động khi người dùng đăng nhập vào hệ thống lần đầu.

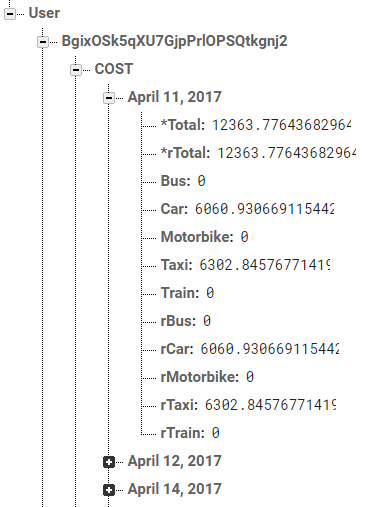
****

Hình 3.4.4.1. Cấu trúc Node “User”

* + - 1. **Node “User/[UID]/COST”**

Node “COST” lưu trữ toàn bộ thông tin về chi phí đi lại của một nhân viên, từ lúc bắt đầu sử dụng hệ thống cho đến hiện tại. Chi phí đi lại được chia theo từng ngày, lưu trữ trong các node con của node “COST”, được đặt tên bằng ngày mà dữ liệu đó được gửi lên (ví dụ: April 11. 2017). Chi phí mỗi ngày bao gồm các node con:

* “\*Total”: tổng chi phí ước tính, được tính bởi các công thức thành phần lấy từ node “Formulation”.
* “\*rTotal”: tổng chi phí thực tế, chính là tổng chi phí ước tính cộng thêm các chi phí phát sinh do người dùng gửi lên, luôn lớn hơn hoặc bằng tổng chi phí ước tính.
* “Bus”: chi phí sử dụng xe buýt ước tính.
* “Car”: chi phí sử dụng xe ô tô ước tính.
* “Motorbike”: chi phí sử dụng xe máy ước tính.
* “Taxi”: chi phí sử dụng xe taxi ước tính.
* “Train”: chi phí sử dụng tàu hỏa ước tính.
* “rBus”: chi phí sử dụng xe buýt thực tế.
* “rCar”: chi phí sử dụng xe ô tô thực tế.
* “rMotorbike”: chi phí sử dụng xe máy thực tế.
* “rTaxi”: chi phí sử dụng xe taxi thực tế.
* “rTrain”: chi phí sử dụng tàu hỏa thực tế

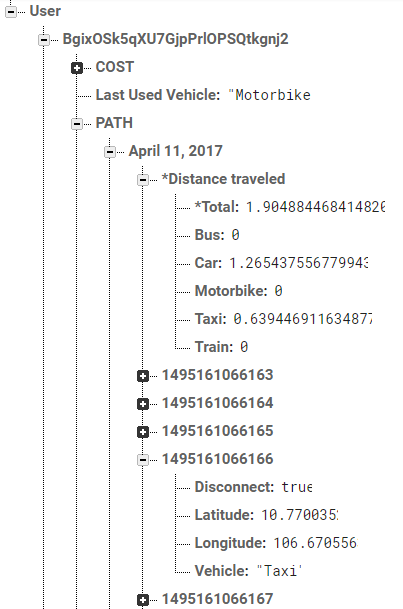


Hình 3.4.4.1.1. Cấu trúc node “COST”

* + - 1. **Node “User/[UID]/PATH”**

Node “PATH” lưu trữ toàn bộ thông tin về quãng đường và danh sách các tọa độ đã đi qua của một nhân viên. Thông tin được lữu trữ theo từng ngày, trong các node con của node “PATH”, được đặt tên bằng ngày mà dữ liệu đó được gửi lên (ví dụ: April 11. 2017). Thông tin này bao gồm các node con:

* Node “\*Distance traveled”: lưu trữ thông tin về quãng đường đã đi trong ngày (tính theo km), bao gồm các thông tin sau:
  + “\*Total”: tổng quãng đường đã di chuyển trong ngày.
  + “Bus”: quãng đường di chuyển bằng xe buýt.
  + “Car”: quãng đường di chuyển bằng xe ô tô.
  + “Motorbike”: quãng đường di chuyển bằng xe máy.
  + “Taxi”: quãng đường di chuyển bằng xe taxi.
  + “Train”: quãng đường di chuyển bằng tàu hỏa.
* Các node tọa độ: lưu trữ thông tin về một tọa độ mà người dùng đã đi qua, khởi tạo trong lúc người dùng di chuyển, được đặt tên bằng số mili giây tính từ ngày 1 tháng 1 năm 1970 đến thời điểm dữ liệu được gửi lên, ví dụ: “[*1495161066163*](https://console.firebase.google.com/project/mitani-9cef4/database/data/User/BgixOSk5qXU7GjpPrlOPSQtkgnj2/PATH/April%2011%2C%202017/1495161066163)” (nhằm đảm bảo dữ liệu được sắp xếp theo đúng thứ tự gửi lên). Tập hợp các node tọa độ sẽ cho ta biết toàn bộ quá trình di chuyển (quãng đường) của nhân viên trong ngày đó. Mỗi node tọa độ bao gồm các node con:
  + “Latitude”: vĩ độ.
  + “Longitude”: kinh độ.
  + “Vehicle”: loại phương tiện đang được sử dụng (ví dụ: “Taxi”).
* “Disconnect”: được khởi tạo tại thời điểm người dùng ngắt kết nối khỏi hệ thống (do đăng xuất, điện thoại hết pin, hết lưu lượng mạng, tắt app hoàn toàn, …), luôn có giá trị là “true”, tại các node tọa độ mà người dùng vẫn đang trong trạng thái kết nối thì node “Disconnect” sẽ không được khởi tạo.

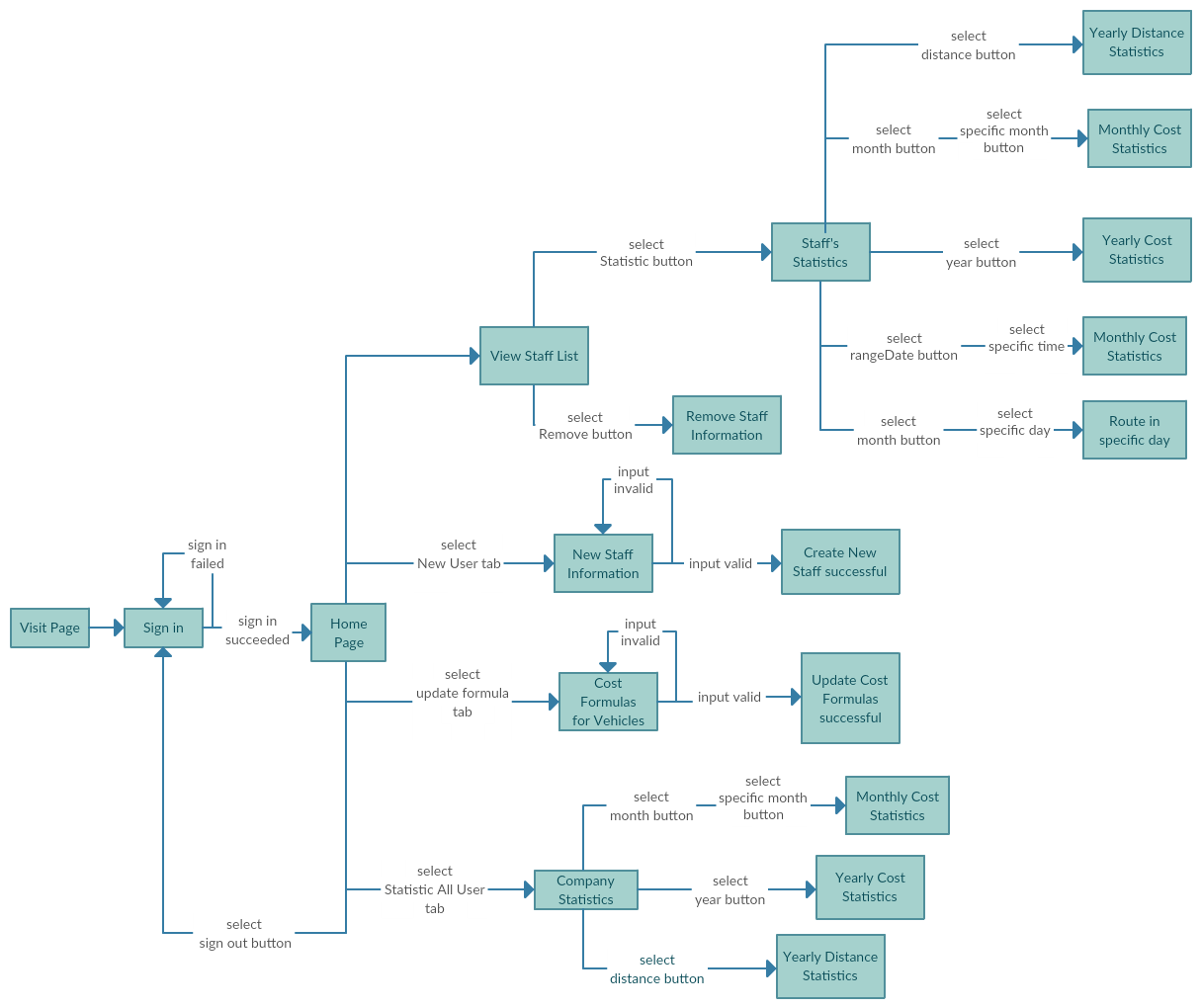


Hình 3.4.4.2.1. Cấu trúc node “PATH”

# HIỆN THỰC HỆ THỐNG

## Website

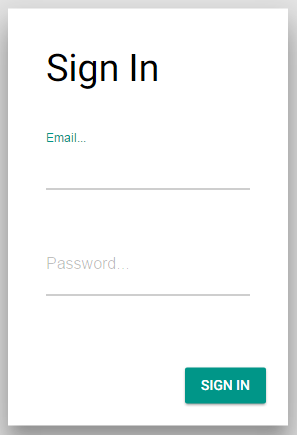
* + 1. **Workflow**

****

Hình 4.1.1.1. Workflow của Website

* + 1. **Giao diện Website**
       1. **Đăng nhập**

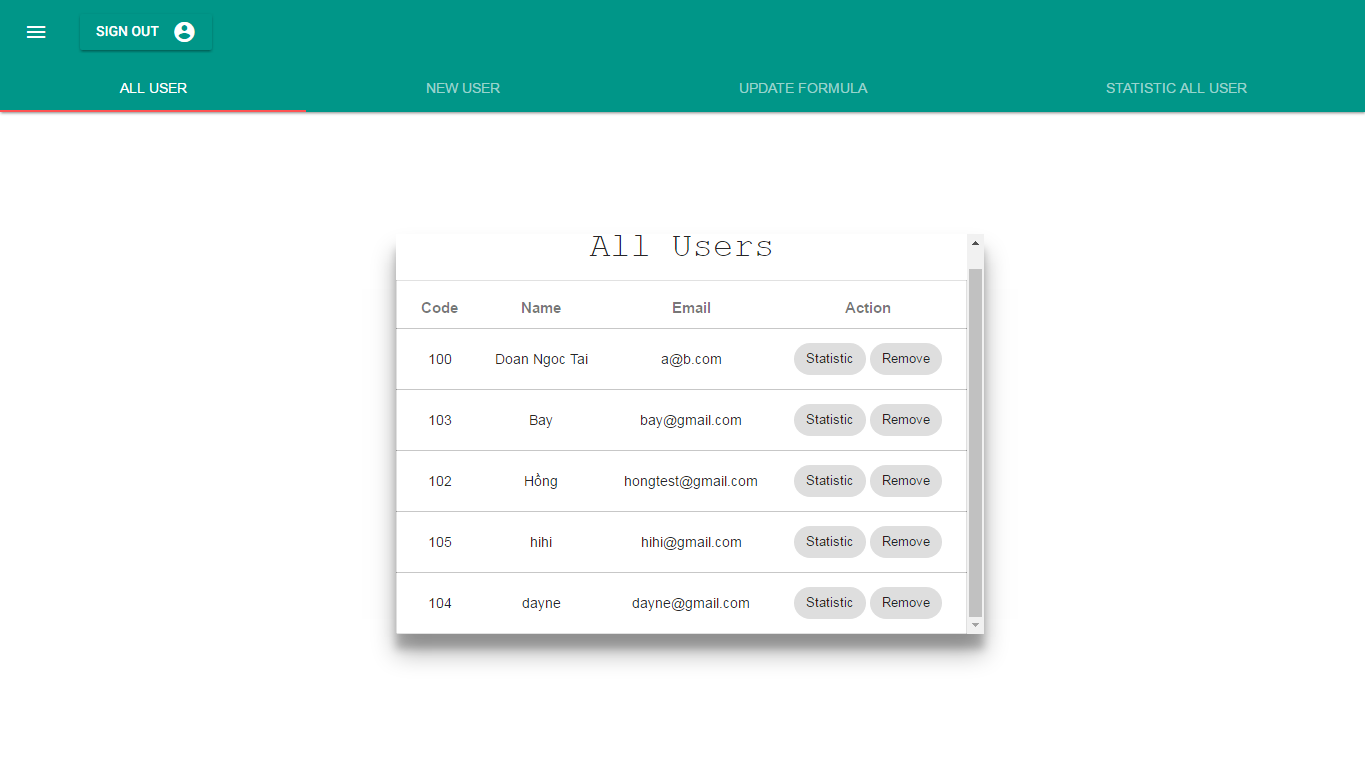
Người quản trị lần đầu vô Website trong ngày cần thực hiện đăng nhập Website bằng tài khoản của mình.



Hình 4.1.2.1.1. Trang Đăng nhập

* + - 1. **Trang chủ**

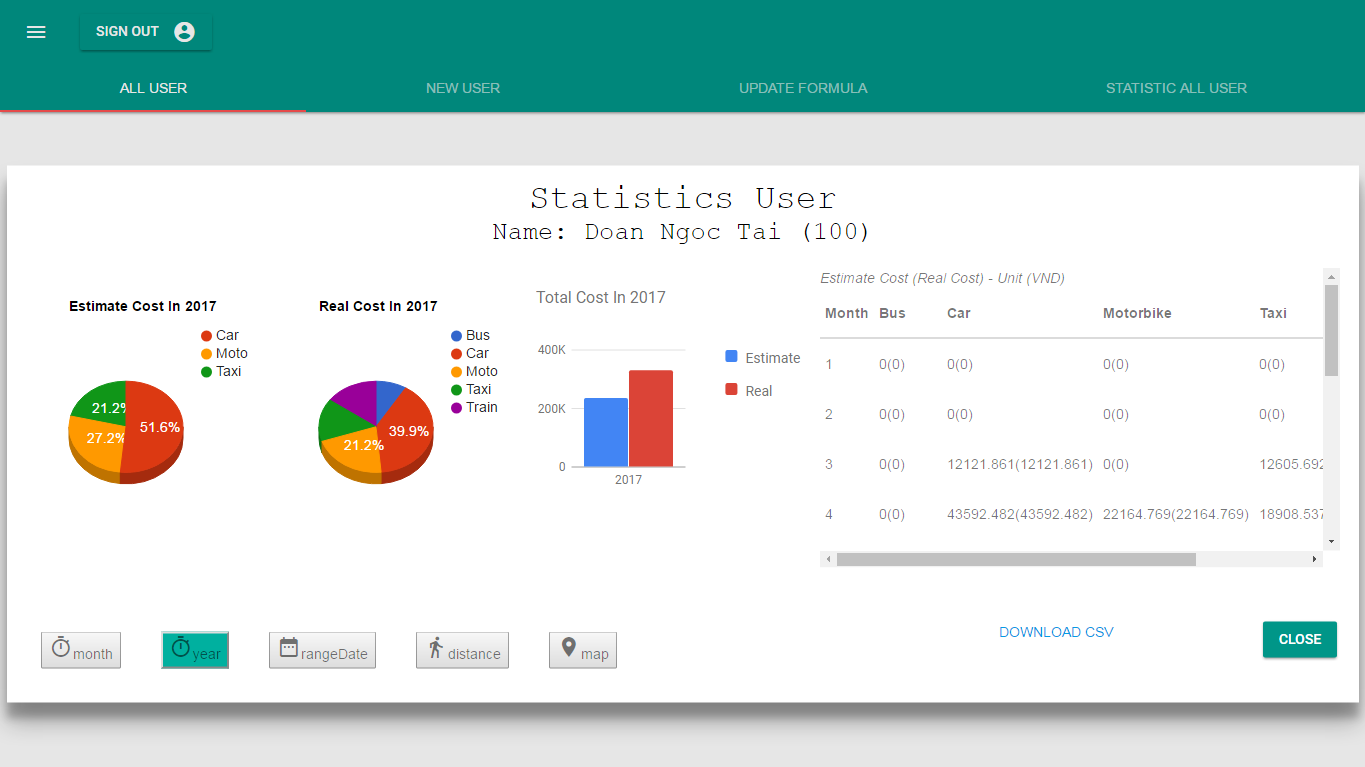
Sau khi đăng nhập, người quản trị sẽ vào trang chủ của Website. Ở trang chủ, người quản trị có thể xem được danh sách nhân viên của công ty. Người quản trị có thể lựa chọn xem thống kê cá nhân của nhân viên (tới trang xem thống kê cá nhân) hoặc xóa nhân viên.



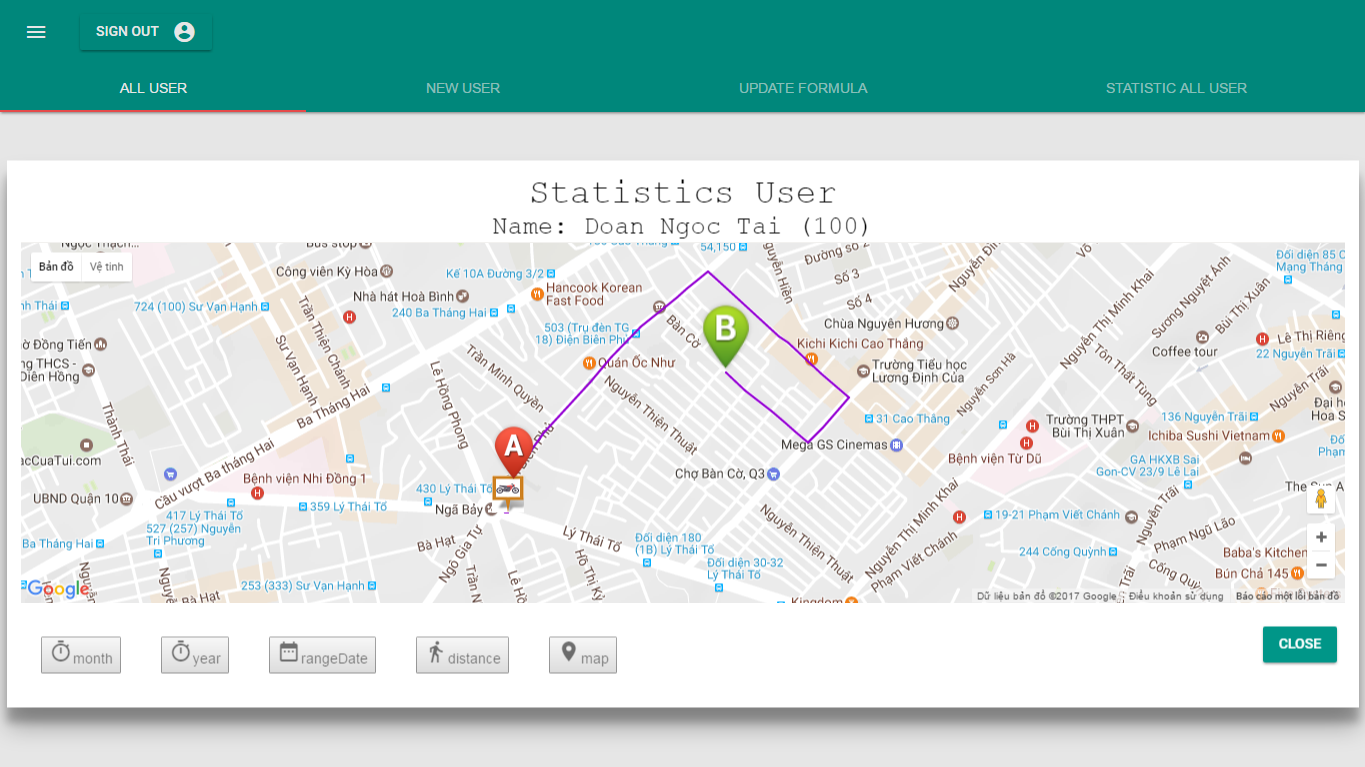
Hình 4.1.2.2.1. Trang chủ

* + - 1. **Trang thống kê cá nhân**

Từ trang chủ, người quản trị bấm nút “Statistic” của một nhân viên nào đó sẽ hiện ra trang thống kê cá nhân của nhân viên đó. Người quản trị có thể xem thống kê chi phí đi lại cá nhân của nhân viên theo khoảng thời gian mong muốn (theo tháng, theo năm, theo khoảng thời gian cụ thể), thống kê quãng đường đi lại của nhân viên trong năm hiện tại và quãng đường nhân viên di chuyển (bằng bản đồ).



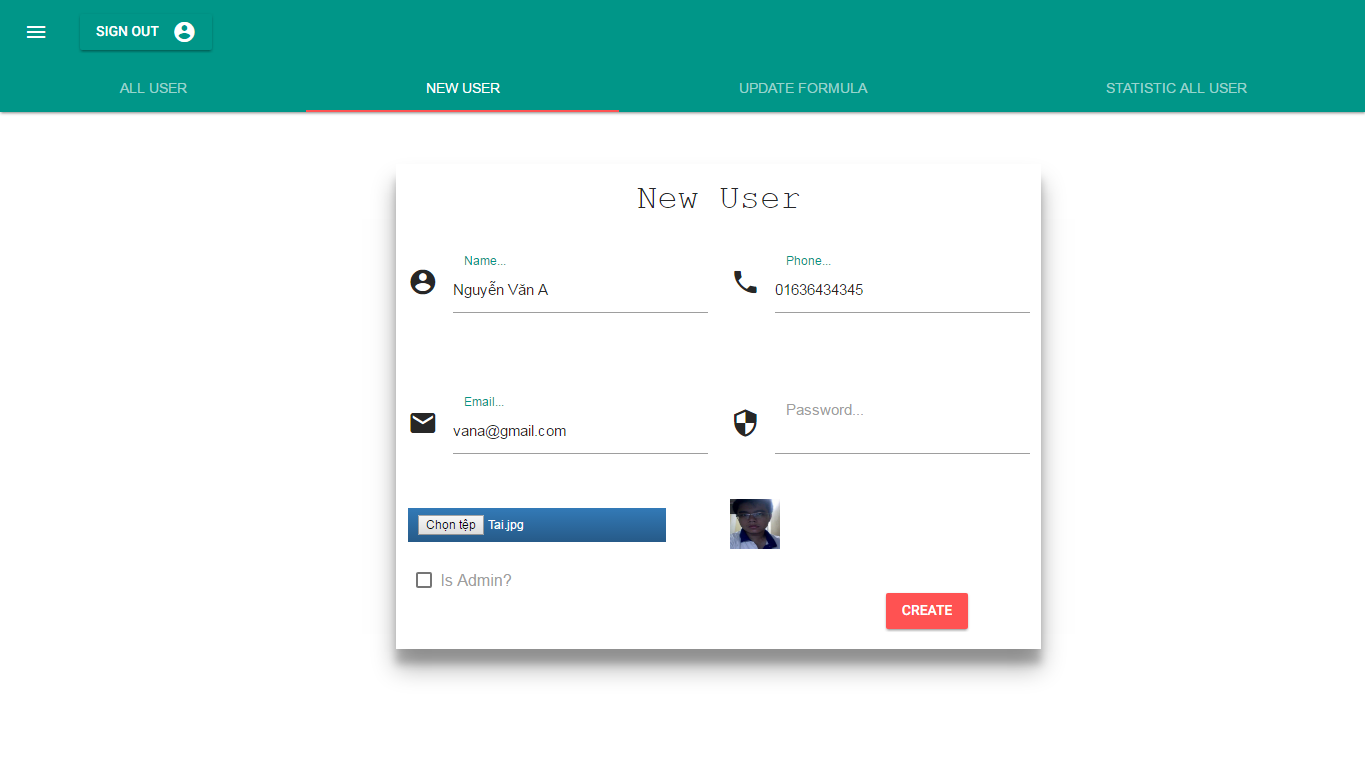
Hình 4.1.2.3.1. Trang thống kê cá nhân theo năm



Hình 4.1.2.3.2. Lộ trình của nhân viên trong một ngày cụ thể

* + - 1. **Trang tạo tài khoản mới**

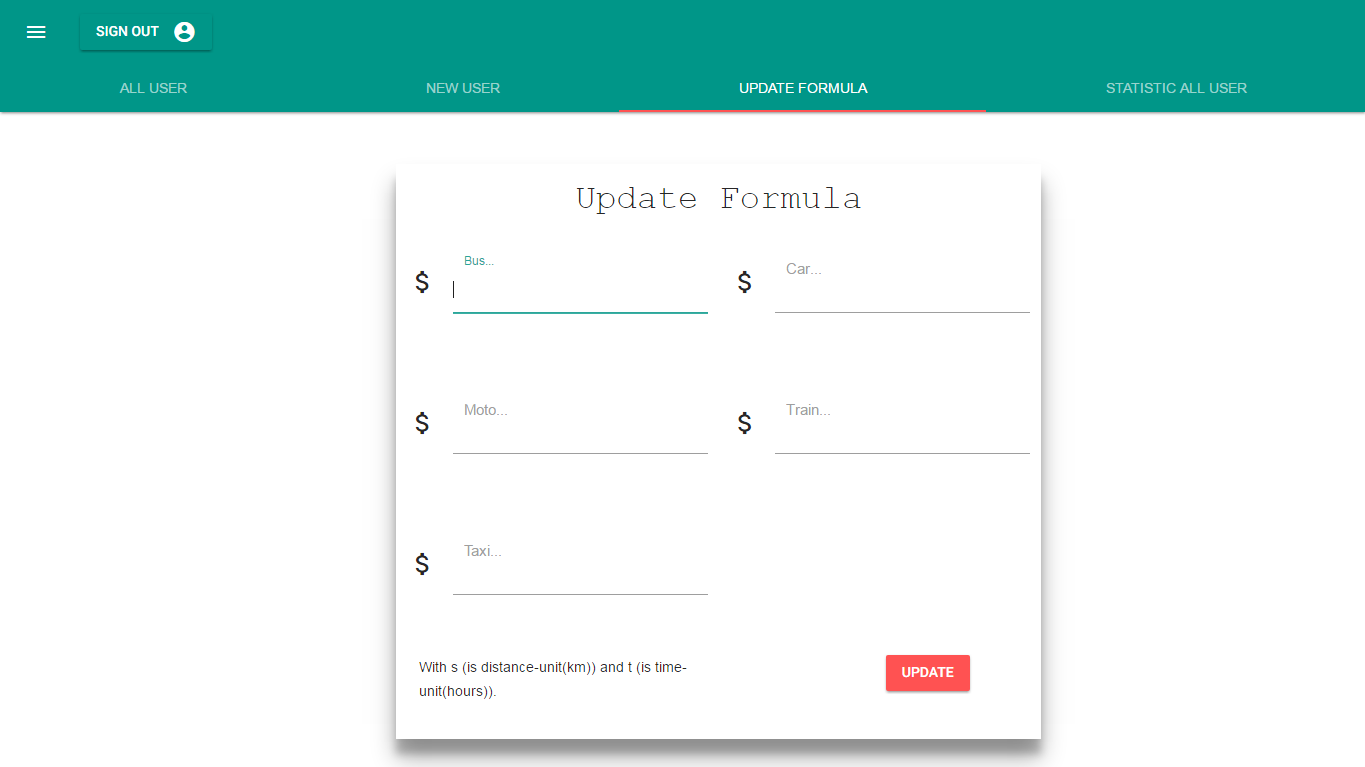
Từ trang chủ, người quản trị muốn thêm tài khoản cho nhân viên mới thì chọn tab “NEW USER”, điền đầy đủ thông tin của nhân viên, chọn ảnh đại diện của nhân viên rồi bấm nút “CREATE”



Hình 4.1.2.4.1. Trang tạo tài khoản mới

* + - 1. **Trang thiết lập công thức tính toán chi phí**

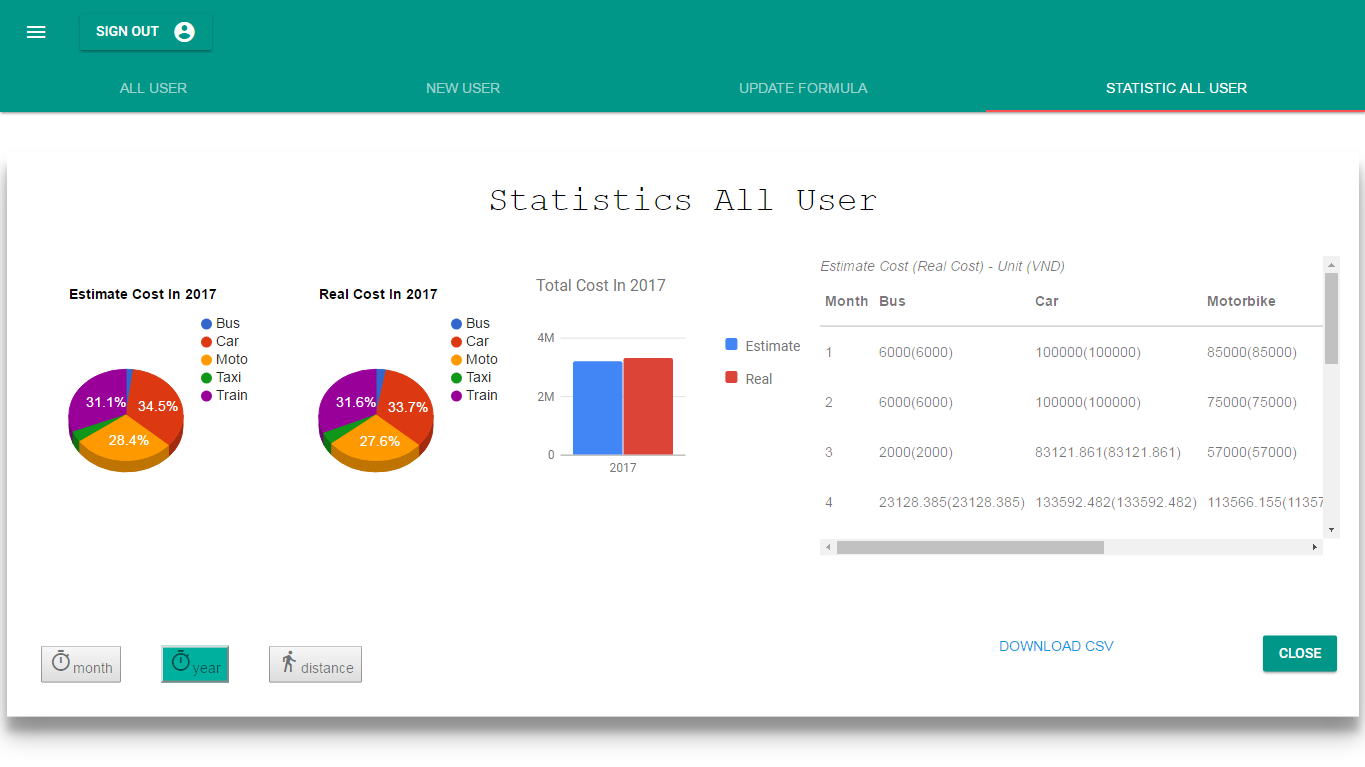
Khi muốn thiết lập công thức tính toán chi phí cho từng loại phương tiện, người quản trị chọn tab “UPDATE FORMULA”, điền công thức tính toán chi phí cho các phương tiện rồi bấm “UPDATE”



Hình 4.1.2.4.1. Trang thiết lập công thức tính toán chi phí đi lại của các loại phương tiện

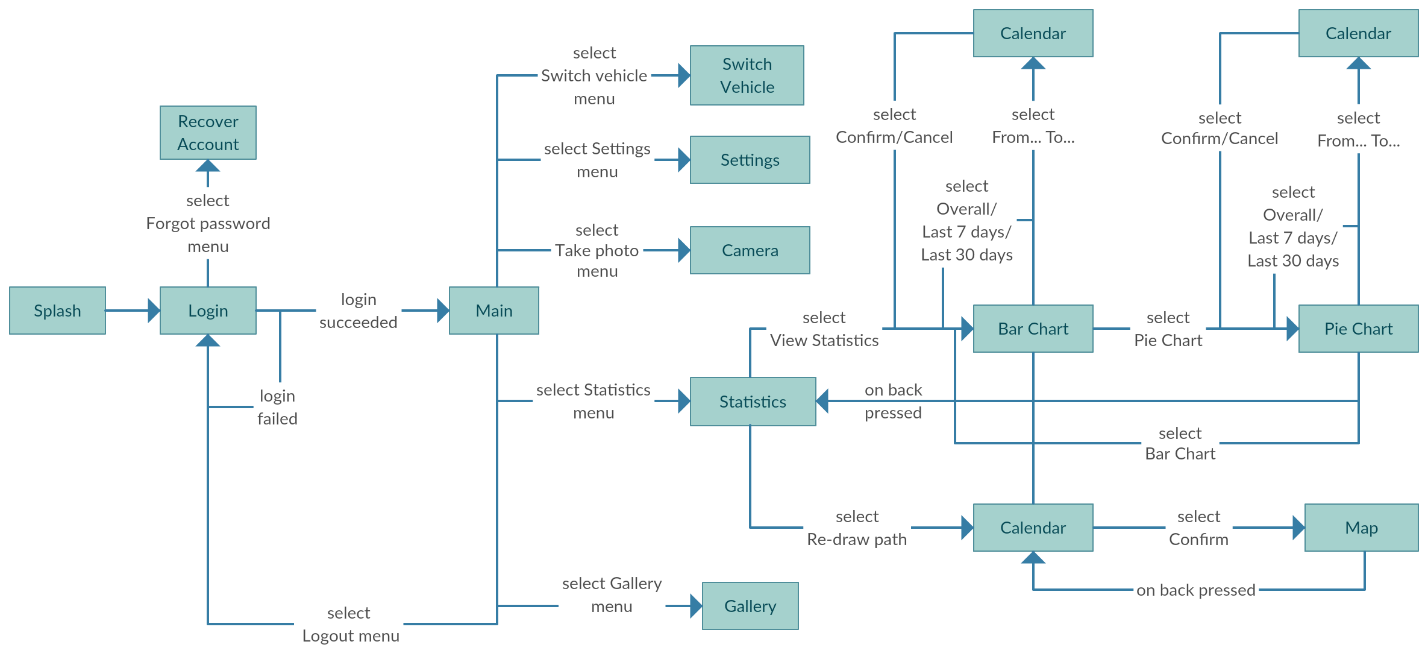
* + - 1. **Trang thống kê công ty**

Khi muốn xem thống kê của công ty, người quản trị chọn tab “STATISTIC ALL USER”, thống kê của tất cả các nhân viên trong công ty (toàn công ty) sẽ hiện lên. Người quản trị có thể xem thống kê về chi phí đi lại của toàn công ty theo khoảng thời gian mong muốn (theo tháng, theo năm), thống kê quãng đường đi lại của nhân viên trong năm hiện tại.



Hình 4.1.2.6.1. Trang thống kê công ty theo năm

* 1. **Ứng dụng di động**
     1. **Workflow**

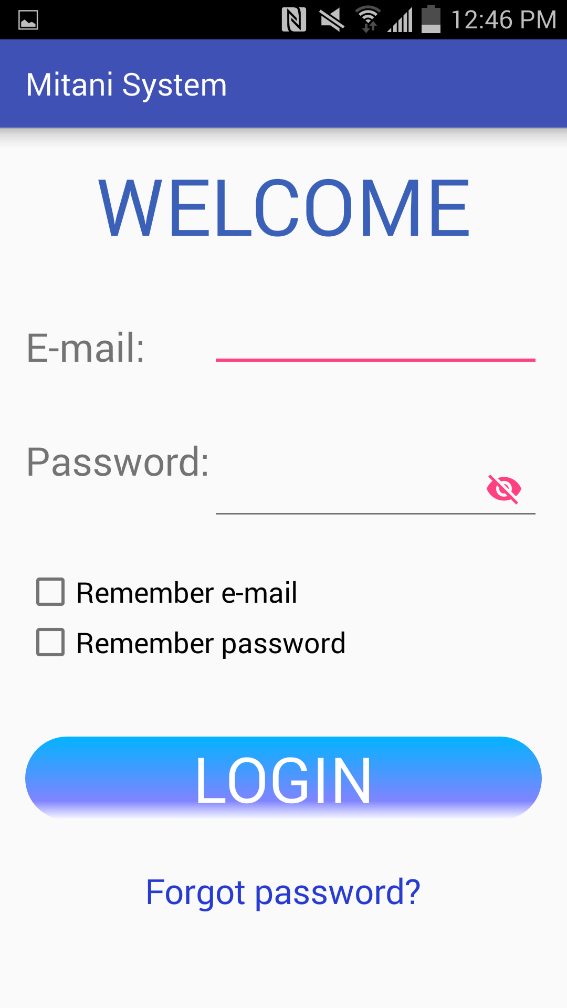


Hình 4.2.1.1. Workflow của ứng dụng di động

1. Ứng dụng di động
   1. Workflow

//todo

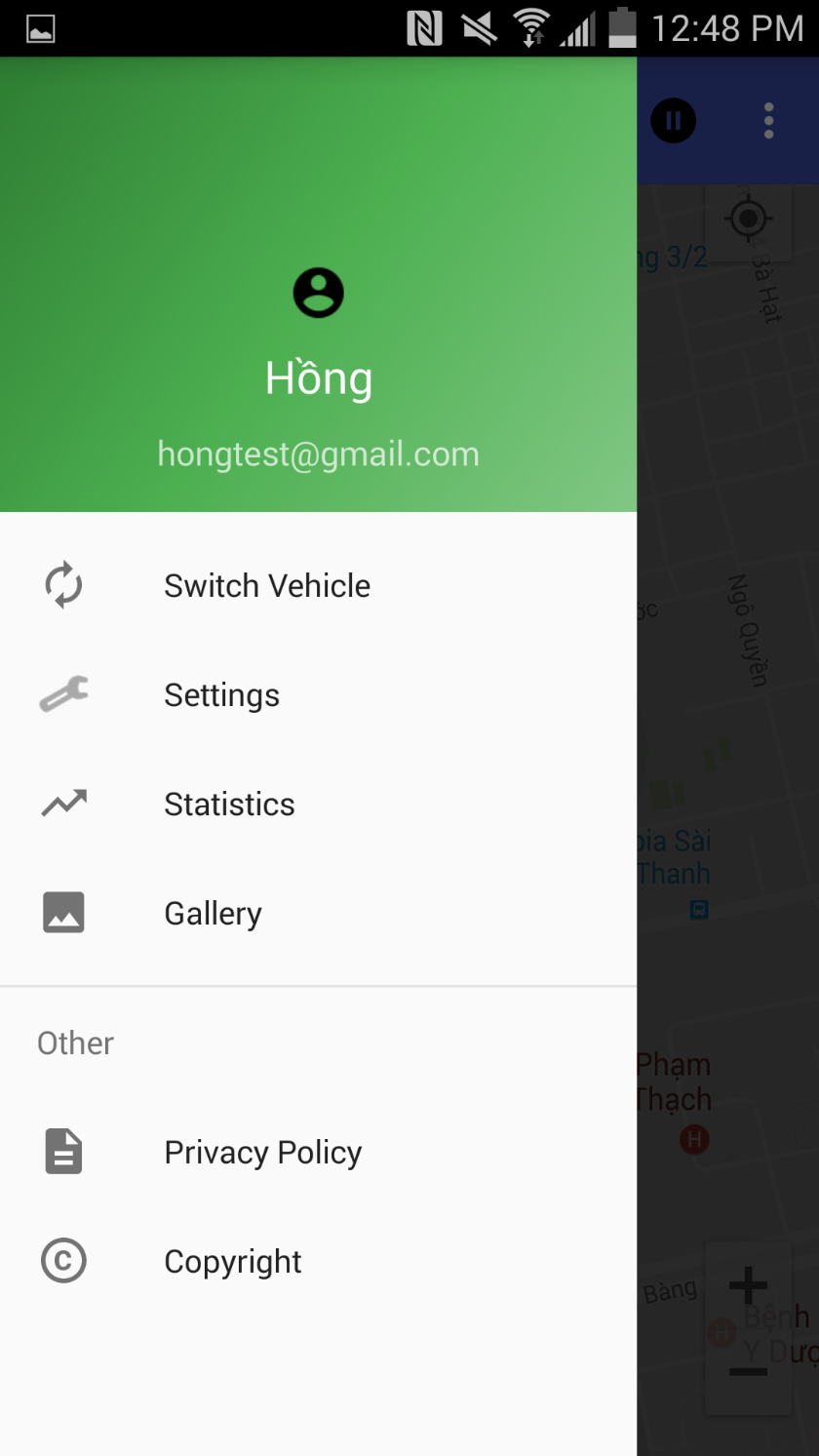
* 1. Giao diện ứng dụng di động
     1. Màn hình đăng nhập



* + 1. Màn hình



* + 1. Màn hình



Hệ thống có thể được chia thành các khối chức năng chính sau:

|  |  |
| --- | --- |
| Khối chức năng | Nhiệm vụ |
| UI | Gồm các Activity hiện thực giao diện để người dùng tương tác với phần mềm. |
| Camera | Hiện thực các hàm sử dụng camera, xác định khuôn mặt, tiền xử lý hình ảnh… |
| Feature Extraction | Hiện thực ba giải thuật rút trích đặc trưng Eigenface, Fisherface và Local Binary Pattern. |
| Security | Hiện thực chức năng bảo mật của chương trình: Codebook và HMAC. |
| Authentication | Hiện thực quá trình đăng ký và xác thực người dùng. |
| Utils | Gồm các hàm trợ giúp như xử lý dữ liệu đầu vào, lưu trữ các vector đặc trưng… |
| Settings | Cho phép người dùng xem hướng dẫn, chọn giải thuật xác thực. |

### Chi tiết các khối

**UI**

|  |  |
| --- | --- |
| Bảng tóm tắt các Activity | |
| WelcomeActivity | Activity khởi động của chương trình, hiển thị giới thiệu sơ lược về ứng dụng. Ở bước này, các tập tin chứa dữ liệu huấn luyện cũng sẽ được sao lưu vào máy để sử dụng cho những bước sau. |
| SignUpActivity | Activity hiển thị màn hình để người dùng đăng ký mới, đăng ký lại, hoặc kích hoạt tính năng xác thực khuôn mặt. |
| PictureActivity | Activity hỗ trợ chụp khuôn mặt của người dùng, và cắt hình ảnh theo kích thước định sẵn. |
| FaceTrackerActivity | Activity dùng cho giai đoạn đăng ký khuôn mặt, sau khi chụp ảnh, sẽ tiến hành huấn luyện lần lượt với cả ba giải thuật, và lưu sketch cũng như mẫu gốc đã được hash dưới dạng file xml. |
| AuthenticationActivity | Activity dùng cho giai đoạn xác thực khuôn mặt, sau khi chụp ảnh, sẽ tiến hành khôi phục mẫu gốc với mẫu vừa chụp và sketch được lưu lại ở bước trên, sau đó tiến hành so sánh kết quả. |
| SettingActivity | Activity cho phép người dùng xem hướng dẫn của chương trình và lựa chọn giữa các giải thuật xác thực. |

**Camera**

|  |  |
| --- | --- |
| Bảng tóm tắt các phương thức quan trọng | |
| CameraSourcePreview | Yêu cầu tính năng camera của điện thoại. |
| FaceGraphic | Xác định khuôn mặt của người dùng, vẽ khung hình chữ nhật bao quanh khuôn mặt trên camera. |
| PhotoHandle | Đọc, lưu, và cắt hình ảnh theo kích thước định sẵn. |
| PreprocessImage | Tiền xử lý hình ảnh trước khi áp dụng các giải thuật, làm mờ nền, chuyển sang Đa mức xám (Grayscale), cân chỉnh sáng. |

**Feature Extraction**

Ở đây, nhóm hiện thực rút trích đặc trưng với ba giải thuật Eigenface, Fisherface và LBP. Mỗi giải thuật có các đặc thù, tính toán khác nhau, nhưng có những điểm chung như sau:

|  |  |
| --- | --- |
| Bảng tóm tắt các phương thức quan trọng | |
| compute | Dữ liệu đầu vào là tập các hình ảnh và nhãn trong tập huấn luyện, ta sẽ tính toán ra hình ảnh trung bình và vector riêng (eigenvector). |
| doProject | Chiếu các mẫu đăng ký của người dùng vào không gian huấn luyện ở bước trên. |
| getAuthenMat | Sử dụng trong bước xác thực, chiếu mẫu xác thực vào không gian huấn luyện. |

**Security**

|  |  |
| --- | --- |
| Bảng tóm tắt các phương thức quan trọng | |
| calculateRFC2104HMAC | Dữ liệu đầu vào là nội dung thông điệp và khóa, dữ liệu đầu ra là thông điệp đã được mã hóa theo giải thuật HMAC SHA-1. |
| quantization | Dùng để chuyển giá trị của vector đặc trưng từ miền liên tục sang miền rời rạc. |
| sketchGeneration | Tạo ra sketch dựa trên vector đặc trưng. |
| featureRecovery | Dữ liệu đầu vào là mẫu xác thực và sketch, khôi phục lại mẫu ban đầu theo giải thuật codebook. |

**Authentication**

|  |  |
| --- | --- |
| Bảng tóm tắt các phương thức quan trọng | |
| codebookGenerator | Dữ liệu đầu vào là tập các vector đặc trưng sau khi đã chiếu vào không gian huấn luyện, áp dụng giải thuật codebook, dữ liệu đầu ra là tập các sketch ứng với các vector trên. |
| hashTemplate | Dữ liệu đầu vào cũng là tập các vector đặc trưng sau khi đã chiếu vào không gian huấn luyện, áp dụng HMAC, dữ liệu đầu ra là thông tin đã được mã hóa, dùng để so sánh trong bước xác thực. |
| eigenFace/fisherFace/LBPTraining | Đọc tập dữ liệu huấn luyện có sẵn và huấn luyện theo từng giải thuật, gọi hàm sinh sketch và băm mẫu ban đầu, lưu cả hai thông tin lại để dùng trong bước xác thực. |
| codebookRecover | Dữ liệu đầu vào là mẫu xác thực mới và tập các sketch đã lưu. Khôi phục lại mẫu ban đầu bằng cách áp dụng thuật toán tuần tự lên mẫu và từng sketch. Dữ liệu đầu ra là tập các mẫu được khôi phục. |
| predict | Dữ liệu đầu vào là tập các mẫu ban đầu và tập các mẫu vừa được khôi phục lại, vì lý do hiệu suất, nên không so sánh trùng khớp hoàn toàn, mà chỉ so sánh theo từng phần tử, |
| eigenFace/fisherFace/LBPAuthentication | Đọc tập sketch và mẫu ban đầu đã lưu, chiếu mẫu xác thực vào không gian huấn luyện, khôi phục lại mẫu ban đầu, và thực hiện so sánh. |

**Utils**

|  |  |  |
| --- | --- | --- |
| Bảng tóm tắt các phương thức quan trọng | | |
| InputHandle | Từ các hình được chụp, thực hiện giai đoạn tiền xử lý như cắt hình theo chuẩn, cân bằng sáng… |
| StoreModel | Hiện thực các hàm lưu và đọc những dữ liệu cần thiết cho các quá trình đăng ký và xác thực như tập huấn luyện, sketch, mẫu băm… |

**Settings**

Setting Actitivy cho phép người dùng lựa chọn một trong ba giải thuật để xác thực. Hoặc có thể kết hợp cả ba để tăng tính bảo mật.

## Kiểm định sản phẩm

### Danh sách Unit Test Case.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Stt** | **Tên lớp** | **Tên hàm** | **Mã hàm** | **Mô tả** | **Tiền điều kiện** |
| 1 | FaceTracerActivity | doSignUp | SignUp | Đăng ký mẫu sinh trắc với hệ thống. | -Đăng ký lần đầu.  -Đã có các file cần thiết.  -Chụp đủ 10 tấm hình khuôn mặt. |
| 2 | SignUpActivity | doReSignUp | Re-SignUp | Cập nhật mẫu sinh trắc (Đăng ký lại). | -Đã đăng ký  -Đã xác thực thành công. |
| 3 | AuthenticationActivity | doAuthentication | Authentication | Thực hiện xác thực. | -Đã đăng ký.  -Đã có các file cần thiết.  -Đã có hình để xác thực. |
| 4 | SettingActivity | onSharedPreferenceChanged | Setting | Các tùy chỉnh thuật toán, ngưỡng… | -Không. |
| 5 | WelcomeActivity | launchWelcomeActivity | Help | Thông tin về ứng dụng. | -Không. |

### Chi tiết Unit Test Case

* SignUp

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  | **UTCID01** |
| **Điều kiện** | **Tiền điều kiện** |  |  |  |
|  |  | **Đăng ký lần đầu** |  |  |
|  |  | **Có các file xml cần thiết** |  |  |
|  |  | **Có đủ 10 tấm hình khuôn mặt** |  |  |
| **Confirm** | **Return** |  |  |  |
|  | **Exception** |  |  |  |
|  | **Log message** |  |  |  |
|  |  |  | "Sign Up Successfully" | **O** |
| **Result** | Type(N : Normal, A : Abnormal, B : Boundary) | | | N | |
|  | Passed/Failed | | | P | |
|  | Executed Date | | | # | |
|  | Defect ID | | |  | |

* Re-SignUp

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **UTCID01** | |
| **Điều kiện** | **Tiền điều kiện** |  |  |  | |
|  |  | **Đã đăng ký** |  |  | |
|  |  | **Đã xác thực thành công** |  |  | |
| **Confirm** | **Return** |  |  |  | |
|  | **Exception** |  |  |  | |
|  | **Log message** |  |  |  | |
|  |  |  | "Re-SignUp Successfully" | **O** | |
| **Result** | Type(N : Normal, A : Abnormal, B : Boundary) | | | | N | |
|  | Passed/Failed | | | | P | |
|  | Executed Date | | | | # | |
|  | Defect ID | | | |  | |

* Authentication

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | **UTCID01** | | **UTCID02** | |
| **Điều kiện** | **Tiền điều kiện** |  |  |  | |  | |
|  |  | **Đã đăng ký** |  |  | |  | |
|  |  | **Đã có các file cần thiết** |  |  | |  | |
|  |  | **Đã có hình để xác thực** |  |  | |  | |
|  | **Hình xác thực đúng** |  |  | **O** | |  | |
|  | **Hình xác thực sai** |  |  |  | | **O** | |
| **Confirm** | **Return** |  |  |  | |  | |
|  |  |  | true | **O** | |  | |
|  |  |  | false |  | | **O** | |
|  | **Exception** |  |  |  | |  | |
|  | **Log message** |  |  |  | |  | |
| **Result** | Type(N : Normal, A : Abnormal, B : Boundary) | | | | N | | N | |
|  | Passed/Failed | | | | P | | P | |
|  | Executed Date | | | | # | | # | |
|  | Defect ID | | | |  | |  | |

* Setting

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **UTCID01** | **UTCID02** |
| **Điều kiện** | **Tiền điều kiện** |  |  |  |  |
|  | **Đổi thuật toán** |  |  | **O** |  |
|  | **Đổi ngưỡng** |  |  |  | **O** |
| **Confirm** | **Return** |  |  |  |  |
|  |  |  | true | **O** | **O** |
|  |  |  | false |  |  |
|  | **Exception** |  |  |  |  |
|  | **Log message** |  |  |  |  |
| **Result** | Type(N : Normal, A : Abnormal, B : Boundary) | | | N | N |
|  | Passed/Failed | | | P | P |
|  | Executed Date | | | # | # |
|  | Defect ID | | |  |  |

* Help

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | **UTCID01** | |
| **Điều kiện** | **Tiền điều kiện** |  |  |  | |
| **Confirm** | **Return** |  |  |  | |
|  | **Exception** |  |  |  | |
|  | **Log message** |  |  |  | |
|  |  |  | "Show Welcome Successfully" | **O** | |
| **Result** | Type(N : Normal, A : Abnormal, B : Boundary) | | | | N | |
|  | Passed/Failed | | | | P | |
|  | Executed Date | | | | # | |
|  | Defect ID | | | |  | |

### Báo cáo Unit Test

(N : Normal, A : Abnormal, B : Boundary)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Stt** | **Mã hàm** | **Passed** | **Failed** | **Untested** | **N** | **A** | **B** | **Tổng cộng** |
| 1 | [SignUp](file:///C:\Users\Quang%20Huy\Downloads\Group03_Unit-Test-Case.xls#HotStory!A1) | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 2 | [Re-SignUp](file:///C:\Users\Quang%20Huy\Downloads\Group03_Unit-Test-Case.xls#ChooseStory!A1) | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 3 | [Authentication](file:///C:\Users\Quang%20Huy\Downloads\Group03_Unit-Test-Case.xls#ChooseChapterToRead!A1) | 2 | 0 | 0 | 2 | 0 | 0 | 2 |
| 4 | [Setting](file:///C:\Users\Quang%20Huy\Downloads\Group03_Unit-Test-Case.xls#Setting!A1) | 2 | 0 | 0 | 2 | 0 | 0 | 2 |
| 5 | [Help](file:///C:\Users\Quang%20Huy\Downloads\Group03_Unit-Test-Case.xls#Login!A1) | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
|  | Sub total | 8 | 0 | 0 | 8 | 0 | 0 | 8 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test coverage** |  | **100** | % |
| **Test successful coverage** |  | **100** | % |
| **Normal case** |  | **100** | % |
| **Abnormal case** |  | **0** | % |
| **Boundary case** |  | **0** | % |

Hình . Tỷ lệ kết quả theo testcase

Hình . Số loại testcase

# ĐÁNH GIÁ KẾT QUẢ //TODO

## Phương pháp đánh giá

#### Tiêu chí đánh giá

Nhóm hiện thực và đánh giá theo các tiêu chí sau:

* Thời gian và độ chính xác khi xác thực không sử dụng giải thuật bảo mật
* Thời gian và độ chính xác khi xác thực sử dụng giải thuật bảo mật

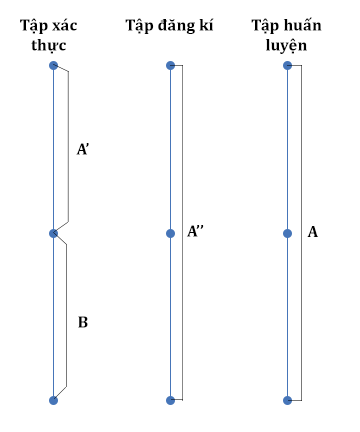
#### Xây dựng tập thí nghiệm sử dụng để huấn luyện, đăng kí và kiểm tra

Để đánh giá các tiêu chí trên, nhóm đã sử dụng tập dữ liệu dùng để nghiên cứ nhận dạng khuôn mặt là Face94. Từ tập Face94 nhóm chọn lọc các hình để tạo lên tập thí nghiệm gồm 3 phần chính:

- Tập huấn luyện A: Tập này dùng để làm không gian chiếu cho PCA và Fisherface. Nhóm chọn 100 người có cả đàn ông và phụ nữ. Mỗi người có 10 tấm hình khác nhau được đánh số theo cấu trúc tên [thứ tự của người đó]-[thứ tự ảnh của người đó]

- Tập đăng kí A’’: Tập này dùng để đăng ký 100 người với hệ thống để tiến hành xác thực. Nhóm chọn 100 người có cả đàn ông và phụ nữ, những người này là những người giống người ở tập A (về thứ tự nhưng khác hình). Mỗi người có 5 tấm hình khác nhau và khác hình ở tập A.

- Tập xác thực: Tập này là những hình đầu vào để xác thực và đo độ chính xác của giải thuật. Tập này gồm 100 người, mỗi người 1 tấm hình (ảnh đúng vị trí gồm 10 ảnh sáng, 10 ảnh tối, 30 ảnh bình thường và ảnh sai vị trí tương tự). Nhóm đã sắp thứ tự những hình này để tạo ra testcase so sánh. Testcase đúng sẽ là đăng nhập được với 50 người đầu (tập A’) và không đăng nhập được với 50 người sau (tập B).



Hình . Cấu trúc tập thí nghiệm

#### Phương pháp đánh giá thực nghiệm

Ta sẽ đánh giá dựa trên các tỷ lệ:

* FAR (False Accept Rate): số lần hệ thống trả về đăng nhập được khi kết quả chính xác của thí nghiệm là không đăng nhập được trên tổng số.
* FRR (False Reject Rate): số lần hệ thống trả về không đăng nhập được khi kết quả chính xác của thí nghiệm là đăng nhập được trên tổng số.

Từ FAR và FRR ta sẽ tìm ra điểm ngưỡng tối ưu mà ở đó tỉ lệ FAR và FRR chấp nhận được.

#### Môi trường thí nghiệm

Điện thoại gionee CTRL V4S với thông số phần cứng như sau:

-Camera trước 2.0 MP

-Hệ điều hành Android 5.0 (Lolillop)

-Chipset MTK 6582

-CPU Quad-core 1.3 GHz

-RAM 1GB

## Các thí nghiệm

#### - Đánh giá hiệu suất giải thuật PCA trên tập thí nghiệm

Hiệu suất giải thuật được đo về thời gian về độ chính xác của giải thuật PCA đối với tập thí nghiệm đã miêu tả ở trên.

Thời gian chạy (đơn vị mm:ss.sss):

|  |  |
| --- | --- |
| PCA | Tổng thời gian |
| Không bảo mật | 17:17.092 |
| Có bảo mật | 26:08.704 |

Độ chính xác:

Hình . Độ chính xác của PCA với các ngưỡng

Với lần chạy không sử dụng bảo mật trên tập thí nghiệm, giải thuật PCA đạt kết quả tốt nhất ở ngưỡng 2000 với tỉ lệ FAR bằng 0/50 và FRR bằng 1/50.

Hình . Độ chính xác của PCA với các DELTA

Với lần chạy sử dụng bảo mật Secure Sketch trên tập thí nghiệm, giải thuật PCA đạt kết quả tốt nhất ở giá trị DELTA bằng 50 với tỷ lệ FAR bằng 1/50 và FRR bằng 2/50.

#### -Đánh giá hiệu suất giải thuật Fisherface trên tập thí nghiệm

Hiệu suất giải thuật được đo về thời gian về độ chính xác của giải thuật Fisherface đối với tập thí nghiệm đã miêu tả ở trên

Thời gian chạy (đơn vị mm:ss.sss):

|  |  |
| --- | --- |
| Fisherface | Tổng thời gian |
| Không bảo mật | 21:27.597 |
| Có bảo mật | 29:56.123 |

Độ chính xác:

Hình . Độ chính xác của Fisherface với các ngưỡng

Với lần chạy không sử dụng bảo mật trên tập thí nghiệm, giải thuật Fisherface đạt hết quả tốt nhất ở ngưỡng 720 với tỉ lệ FAR bằng 0/50 và FRR bằng 0/50.

Hình . Độ chính xác của Fisherface với các DELTA

Với lần chạy sử dụng bảo mật Secure Sketch trên tập thí nghiệm, giải thuật Fisherface đạt kết quả tốt nhất ở giá trị DELTA bằng 42 với tỷ lệ FAR bằng 1/50 và FRR bằng 1/50.

#### -Đánh giá hiệu suất giải thuật LBP trên tập thí nghiệm

Hiệu suất giải thuật được đo về thời gian về độ chính xác của giải thuật LBP đối với tập thí nghiệm đã miêu tả ở trên

Thời gian chạy (đơn vị mm:ss.sss):

|  |  |
| --- | --- |
| PCA | Tổng thời gian |
| Không bảo mật | 15:12.466 |
| Có bảo mật | 21:10.141 |

Độ chính xác:

Hình . Độ chính xác của LBP với các ngưỡng

Với lần chạy không sử dụng bảo mật trên tập thí nghiệm, giải thuật LBP đạt kết quả tốt nhất ở ngưỡng 265 với tỉ lệ FAR bằng 0/50 và FRR bằng 6/50.

Hình . Độ chính xác của LBP với các DELTA

Với lần chạy sử dụng bảo mật Secure Sketch trên tập thí nghiệm, giải thuật LBP đạt kết quả tốt nhất ở giá trị DELTA bằng 3 với tỷ lệ FAR bằng 12/50 và FRR bằng 1/50.

#### -Đánh giá hiệu suất kết hợp 3 giải thuật trên tập thí nghiệm

Hiệu suất giải thuật được đo về thời gian về độ chính xác của 3 giải thuật đối với tập thí nghiệm đã miêu tả ở trên. Cụ thể, kết quả của sự kết hợp được tính là xác thực đúng nếu 2 trên 3 giải thuật cho kết quả xác thực đúng. Các thông số ngưỡng tối ưu và delta tối ưu của mỗi giải thuật được sử dụng để tiến hành thí nghiệm:

|  |  |  |
| --- | --- | --- |
| Thông số tối ưu | Ngưỡng | DELTA |
| PCA | 2000 | 50 |
| Fisherface | 720 | 42 |
| LBP | 265 | 3 |

Thời gian chạy (đơn vị mm:ss.sss) và độ chính xác:

|  |  |  |  |
| --- | --- | --- | --- |
| PCA, Fisherface, LBP | FAR | FRR | Tổng thời gian |
| Không bảo mật | 0/50 | 0/50 | 36:27.66 |
| Có bảo mật | 1/50 | 2/50 | 41:29.083 |

Với lần chạy không sử dụng bảo mật trên tập thí nghiệm, 3 giải thuật đạt kết quả tốt nhất với tỉ lệ FAR bằng 0/50 và FRR bằng 0/50.

Với lần chạy sử dụng Secure Sketch trên tập thí nghiệm, 3 giải thuật đạt kết quả tốt nhất với tỷ lệ FAR bằng 1/50 và FRR bằng 2/50.

## Kết quả thí nghiệm

Từ các kết quả thí nghiệm trên, ta rút ra một số kết luận sau:

* Thuật toán PCA hoạt động khá hiệu quả khi testcase có cùng điều kiện ánh sáng với hình ảnh trong tập huấn luyện, hướng của khuôn mặt là chính diện. Tốc độ của thuật toán khá nhanh trong việc huấn luyện, đăng ký và xác thực. Đối với những testcase có điều kiện ánh sáng thay đổi, cường độ sáng cao hoặc thấp. Tỷ lệ sai xót khá cao, việc xác thực cho kết quả không như mong đợi.
* Thuật toán Fisherface chạy tốt khi tập train và tập test có cùng điều kiện ánh sáng cũng như diện mạo. Đối với các ảnh sáng hoặc tối, giải thuật chạy không ổn định. Kết quả xác thực chỉ đúng và ổn định ở một số tập sáng đầu tiên.
* Giải thuật LBP chạy tốt ở các điều kiện sáng khác nhau. Độ chính xác của LBP không cao. Tốc độ chạy của LBP nhanh hơn so với PCA và Fisherface.
* Khi không sử dụng bảo mật, giải thuật Fisherface đạt độ chính xác cao nhất, sau đó tới PCA và cuối cùng là LBP. Khi sử dụng bảo mật Secure Sketch, giải thuật PCA đạt độ chính xác cao nhất, sau đó tới Fisherface và cuối cùng là LBP.
* Kết hơp 3 giải thuật cho kết quả tốt khi chạy trên tập thí nghiệm.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Đề tài của luận văn là “Phát triển hệ thống xác thực bằng đặc trưng gương mặt trên Android có bảo vệ mẫu đặc trưng sinh trắc”. Mục tiêu của hệ thống là thay thế màn hình khóa (lock screen) cổ điển bằng xác thực khuôn mặt người dùng, với trọng tâm là rút trích các đặc trưng sinh trắc, kết hợp với bảo mật.

Sau quá trình tìm hiểu và nghiên cứu, nhóm đã lựa chọn và hiện thực ba giải thuật rút trích đặc trưng sinh trắc là Eigenface, Fisherface và Local Binary Pattern. Mỗi giải thuật có ưu, nhược điểm khác nhau. Nhóm đã tiến hành phân tích, hiện thực, so sánh kết quả giữa các giải thuật này. Ngoài ra, nhóm cũng đề xuất sử dụng kết hợp giữa ba giải thuật để đạt được kết quả tốt nhất.

Tuy trọng tâm của đề tài là rút trích đặc trưng, để hoàn thiện hệ thống, nhóm đã hiện thực bảo mật mẫu sử dụng Secure Sketch theo mô hình Codebook và HMAC. Với tính năng này, cho dù thiết bị có bị đánh cắp hoặc tấn công, đặc trưng sinh trắc của người dùng cũng không bị lộ hoặc sử dụng trái phép. Ta cũng có thể dễ dàng đổi khóa của hàm băm để tạo ra mẫu mới, kẻ tấn công sẽ không đăng nhập vào hệ thống với mẫu cũ được.

Về kết quả, cả ba giải thuật có hiệu suất khá cao trên tập Faces94, cũng như khi sử dụng với các khuôn mặt ở ngoài. Điều này cho thấy các giải thuật rút trích được hiện thực khá tốt.

Điểm trừ của hệ thống đó là sau khi áp dụng thêm lớp bảo mật, hiệu suất giảm, đặc biệt là khi sử dụng trong thực tế. Lý do chính là vì các tác động bên ngoài như ánh sáng, góc mặt, kiểu tóc, phông nền… ảnh hưởng đến quá trình rút trích đặc trưng. Mặc dù Secure Sketch giúp sửa lỗi và khôi phục lại mẫu gốc, nhưng rất khó để khôi phục lại chính xác hoàn toàn mẫu gốc ban đầu. Tuy nhiên với tỷ lệ lỗi thấp, đặc biệt là FAR, chương trình vẫn đảm bảo sự an toàn tốt cho hệ thống.

## Hướng phát triển

Về hướng phát triển, có thể được chia thành hai hướng chính sau đây:

* Nâng cao giải thuật tiền xử lý cho ảnh: hiện tại, hệ thống cũng đã có chức năng giúp cân chỉnh sáng, xóa mờ phông nền cho ảnh sau khi chụp. Tuy nhiên, để cải thiện hiệu suất, cần có sự tìm hiểu kỹ hơn, và hiện thực các giải thuật tốt hơn.
* Bảo mật cho hệ thống: tính năng bảo mật cho hệ thống hiện tại còn đơn giản, tuy có khắc phục nhiễu nhưng chưa hoàn chỉnh. Tuy bảo mật không phải là trọng tâm của đề tài, nhưng nhóm cũng đã có tìm hiểu thêm các giải thuật sinh codebook khác nhau như Kekre’s Fast Code Book Generation kết hợp với K-Means. Do nhận thấy các giải thuật đó không phù hợp với các vector đặc trưng, nên nhóm đã không hiện thực. Do đó, cần giành nhiều thời gian nghiên cứu các giải thuật khác để đảm bảo cho hệ thống an toàn cũng như không làm giảm hiệu suất.

# THAM KHẢO

[1] Peter H. Dana (2000, May 2). *The Global Positioning System*. Retrieved from <http://www.colorado.edu/geography/gcraft/notes/gps/gps_f.html>

[2] *What is GPS?* (Data Accessed: 2017, May 26). Retrieved from <http://www.furuno.com/en/gnss/technical/tec_what_gps>

[3] Kaplan, E., & Hegarty, C. (2005). *Understanding GPS: principles and applications*. Artech house.

[4] *Android (Hệ điều hành)* (Data Accessed: 2017, May 26). Retrieved from <https://vi.wikipedia.org/wiki/Android_(hệ_điều_hành)>

[5] *App stores: number of apps in leading app stores 2017* (Data Accessed: 2017, May 26). Retrieved from <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>

[6] *Google Play: number of downloads 2010-2016* (Data Accessed: 2017, May 26). Retrieved from <https://www.statista.com/statistics/281106/number-of-android-app-downloads-from-google-play/>

[7] *Platform Architecture* (Data Accessed: 2017, May 26). Retrieved from <https://developer.android.com/guide/platform/index.html>

[8] *Dashboards* (Data Accessed: 2017, May 27). Retrieved from <https://developer.android.com/about/dashboards>