

**BỘ GIÁO DỤC ĐÀO TẠO
TRƯỜNG ĐẠI HỌC ĐẠI NAM**



ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG APP ĐẶT TAXI TRÊN NỀN TẢNG ANDROID

SINH VIÊN THỰC HIỆN : NGUYỄN DUY KHƯƠNG

MÃ SINH VIÊN 1351020062

KHOA : CÔNG NGHỆ THÔNG TIN

HÀ NỘI - 2023

**BỘ GIÁO DỤC ĐÀO TẠO
TRƯỜNG ĐẠI HỌC ĐẠI NAM**



NGUYỄN DUY KHUỜNG

**XÂY DỰNG APP ĐẶT TAXI TRÊN NỀN
TẢNG ADNROID**

CHUYÊN NGÀNH : CÔNG NGHỆ THÔNG TIN
MÃ SINH VIÊN 1351020062

NGƯỜI HƯỚNG DẪN : TS. TRẦN ĐĂNG CÔNG

HÀ NỘI - 2023

LỜI CAM ĐOAN

Tôi xin cam đoan rằng đồ án tốt nghiệp với đề tài... là nghiên cứu độc lập của tôi. Đồng thời đây là kết quả nghiên cứu hoàn toàn trung thực, không sao chép từ bất kì một công trình nghiên cứu khác nào. Những tài liệu trích dẫn đều đã được ghi rõ nguồn gốc. Tôi xin chịu hoàn toàn trách nhiệm trước nhà trường nếu trường hợp phát hiện ra bất cứ sai phạm hay vấn đề sao chép nào trong đề tài này.

Hà Nội, ngày 7 tháng 6 năm 2023

Sinh viên thực hiện

LỜI MỞ ĐẦU

Cơ sở vật chất của Khoa Công Nghệ Thông Tin được nâng cấp liên tục trong nhiều năm qua nhưng vẫn chưa theo kịp sự phát triển mạnh và liên tục của Ngành Công Nghệ Thông Tin. Mặc dù vậy, nhờ sự hỗ trợ của các Thầy, Cô của khoa đã tạo điều kiện cho chúng em được làm đồ án thông qua kỳ tốt nghiệp này. Bản thân là sinh viên Khoa công nghệ thông tin, tuy nhiên kinh nghiệm sống cũng như làm việc của bản thân em là chưa nhiều, chính vì vậy việc tổ chức làm đồ án tốt nghiệp này là cơ hội tốt cho chúng em được trải nghiệm kiến thức thực tế và áp dụng công việc một cách khách quan nhất.

Tốt nghiệp là quá trình rèn luyện suốt 4 năm học đại học của bản thân em, hoàn thiện những kiến thức chưa đầy đủ, rõ ràng trong khi được học tập tại trường. Đồ án tốt nghiệp là cơ hội để sinh viên đem những kiến thức đã học suốt 4 năm để thể hiện, làm quen với công việc của mình trong tương lai, giảm đi khoảng cách lý thuyết và thực tế, giữa nhà trường và xã hội. Làm đồ án giúp sinh viên có thêm những kỹ năng mới: kỹ năng làm việc liên tục, học hỏi liên tục, khả năng chịu đựng áp lực công việc, kỹ năng phân tích, kỹ năng tổng hợp...

Sau nhiều ngày làm đồ án thì em đã tiếp thu được những kiến thức thực tế rất đáng quý. Qua đó nhận ra được trình độ kiến thức của mình hiện tại đang nằm ở đâu và có định hướng để cố gắng cho tương lai của mình. Trong thời gian làm đồ án nội dung chủ yếu của em là xây dựng app đặt taxi trên nền tảng android.

Để củng cố kiến thức đã học và ứng dụng trong một số công việc cụ thể nên em đã được giao tìm hiểu về App đặt taxi. Em xin chân thành cảm ơn Giảng viên (TS) Trần Đăng Công đã tận tình chỉ dạy và hướng dẫn em hoàn thiện đồ án này.

LỜI CẢM ƠN

Lời đầu tiên cho Em xin gửi lời cảm ơn đến toàn thể Ban giám Hiệu khoa Công Nghệ Thông Tin–Trường Đại Học Đại Nam đã tạo điều kiện cho em được tham gia làm đồ án và viết bài báo cáo này, đây là cơ hội tốt để cho em có thể đúc kết các kỹ năng, kiến thức học trên lớp và cũng giúp ích rất lớn để em ngày càng tự tin về bản thân mình.

Em xin gửi lời cảm ơn chân thành đến Thầy Giáo – Giảng viên hướng dẫn Trần Đăng Công trong suốt thời gian vừa qua đã nhiệt tình chỉ dạy, giúp đỡ để Em có thể hoàn thiện tốt đồ án tốt nghiệp này.

Mặc dù đã cố gắng hoàn thành tốt nhất bản đồ án, tuy nhiên do hạn chế về mặt thời gian và kinh nghiệm nên trong đồ án có nhiều thiếu sót. Do vậy kính mong thầy bỏ qua và góp ý để bản báo cáo được hoàn thiện hơn.

Một lần nữa, em xin được bày tỏ lòng cảm ơn chân thành nhất!

Hà Nội, Ngày 25 Tháng 05 Năm 2023

Sinh viên

MỤC LỤC

CHƯƠNG 1. GIỚI THIỆU VỀ LẬP TRÌNH MOBILE TRÊN NỀN TẢNG

ANDROID.....	1
1.1. Giới thiệu về hệ điều hành Android	1
1.1.1. Android là một hệ điều hành dựa trên nền tảng Linux.	1
1.1.2. Ưu điểm của hệ điều hành Android	2
1.1.3. Nhược điểm của hệ điều hành Android	2
1.1.4. Các đặc điểm của Android.....	3
1.1.5. Các phiên bản Android	4
1.1.6. Các thiết bị khác sử dụng hệ điều hành Android	6
1.2. Giới thiệu về lập trình Android và ứng dụng Android studio.....	7
1.2.1. Lập trình android	7
1.2.2. Ứng dụng Android studio	7
1.3. Giới thiệu về hệ quản trị cơ sở dữ liệu Firebase	8
1.3.1. Firebase là gì?	8
1.3.2. Ưu điểm khi sử dụng Firebase	8
1.3.3. Nhược điểm của Firebase	9
CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ BÀI TOÁN ĐẶT XE TAXI	10
2.1. Phát biểu bài toán	10
ĐỐI VỚI QUẢN LÝ	10
ĐỐI VỚI KHÁCH HÀNG	10
ĐỐI VỚI TÀI XẾ	11
2.2. Mô tả bài toán	11
2.3. Thiết kế mô hình hệ thống	11
2.3.1. Xác định các tác nhân (Actor) và chức năng (Usecase).....	11
2.3.2. Biểu đồ tổng quát	13
2.3.3. Use case dành cho các tác nhân	13
2.3.4. Biểu đồ use case chi tiết	14

2.4. Các bước tạo cơ sở dữ liệu trên Firebase	27
2.4.1. Tạo một project mới trên Firebase	27
2.4.2. Cấu hình Firebase lên Android studio.....	28
2.4.3. Đọc, ghi dữ liệu Firebase trên Android	30
CHƯƠNG 3. KẾT QUẢ ĐẠT ĐƯỢC TRONG PHẦN MỀM.....	32
3.1. Sơ đồ công năng chính của phần mềm	32
3.2. Chi tiết về kết quả	32
3.2.1. Giao diện của người dùng	32
3.2.2. Giao diện đăng kí người dùng	34
3.2.3. Giao diện trang chủ	36
3.2.4. Giao diện đăng nhập Tài xế.....	38
3.2.6. Giao diện đăng kí loại xe.....	41
CHƯƠNG 4. KẾT LUẬN.....	42
4.1. Bài học kinh nghiệm	42
4.2. Về kỹ năng	42
4.3. Về phẩm chất đạo đức nghề nghiệp	42
4.4. Những thu hoạch bổ ích nhất với bản thân.....	43
4.5. Những hạn chế của bản thân.....	43
4.6. Dự kiến tiếp theo.....	43
TÀI LIỆU THAM KHẢO	44

DANH MỤC HÌNH ẢNH

Hình 2.1 Sơ đồ User tổng quát	22
Hình 2.2: Sơ đồ Use case dành cho tác nhân “Khách hàng”	23
Hình 2.3: Sơ đồ use case dành cho tác nhân “Tài Xế”	23
Hình 2.4: Sơ đồ Use case chi tiết dành cho tác nhân ”Đăng kí tài khoản”	24
Hình 2.5: Biểu đồ hoạt động đăng ký	25
Hình 2.6: Sơ đồ Use case dành cho tác nhân đăng nhập	26
Hình 2.7: Biểu đồ hoạt động đăng nhập	27
Hình 2.8: Sơ đồ use case tác nhân tìm kiếm	28
Hình 2.9: Biểu đồ hoạt động tìm kiếm	30
Hình 2.10: Sơ đồ use case tác nhân đặt xe.....	30
Hình 2.11: Biểu đồ hoạt động đặt xe	32
Hình 2.12: Sơ đồ use case tác nhân nhận chuyển	32
Hình 2.13: Biểu đồ hoạt động nhận chuyển	34
Hình 2.14 Trang chủ của Firebase	35
Hình 2.15 Tạo mới một Project	35
Hình 2.16: Tạo mới một Project	36
Hình 2.17: Tạo mới một Project	36
Hình 2.18: Tạo mới một Project	37
Hình 2.19: Tạo mới một Project	38
Hình 3.1: Sơ đồ công năng chính của phần mềm	40
Hình 3.2: Giao diện đăng nhập người dùng.....	41
Hình 3.3: Giao diện đăng kí người dùng	42
Hình 3.4: Giao diện trang chủ người dùng	45
Hình 3.5: Giao diện đăng nhập tài xế	46
Hình 3.6: Giao diện đăng kí tài xế	47
Hình 3.7: Giao diện đăng kí loại xe	49

DANH MỤC BẢNG BIỂU

Bảng 1.1: Các đặc điểm của Android	13
Bảng 1.2: Các phiên bản Android	15
Bảng 2.1: Bảng xác định tác nhân và chức năng.....	21
Bảng 2.2: Bảng đặc tả chức năng đăng ký.....	25
Bảng 2.3: Bảng đặc tả chức năng đăng nhập.....	27
Bảng 2.4: Bảng đặc tả chức năng tìm kiếm.....	29
Bảng 2.5: Bảng đặc tả chức năng đặt xe.....	31
Bảng 2.6: Bảng đặc tả chức năng nhận chuyển.....	33

CHƯƠNG 1. GIỚI THIỆU VỀ LẬP TRÌNH MOBILE TRÊN NỀN TẢNG ANDROID

1.1. Giới thiệu về hệ điều hành Android

1.1.1. Android là một hệ điều hành dựa trên nền tảng Linux.

Hệ điều hành Android được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh, máy tính bảng và các thiết bị khác.

Hệ điều hành Android được phát triển bởi Tổng công ty dưới sự hỗ trợ về tài chính của Google. Đến năm 2005, Google chính thức mua lại Android.

Android ra mắt vào năm 2007 cùng với tuyên bố thành lập Liên minh thiết bị cầm tay mở: một hiệp hội gồm các công ty phần cứng, phần mềm, và viễn thông với mục tiêu đẩy mạnh các tiêu chuẩn mở cho các thiết bị di động. Chiếc điện thoại đầu tiên chạy Android được bán vào năm 2008.

Android có mã nguồn mở và Google phát hành mã nguồn theo Giấy phép Apache. Chính mã nguồn mở cùng với một giấy phép không có nhiều ràng buộc đã cho phép các nhà phát triển thiết bị, mạng di động và các lập trình viên nhiệt huyết được điều chỉnh và phân phối Android một cách tự do.

Ngoài ra, Android còn có một cộng đồng lập trình viên đông đảo chuyên viết các ứng dụng để mở rộng chức năng của thiết bị, bằng một loại ngôn ngữ lập trình Java có sửa đổi. Vào tháng 10 năm 2012, có khoảng 700.000 ứng dụng trên Android, và số lượt tải ứng dụng từ Google Play, cửa hàng ứng dụng chính của Android, ước tính khoảng 25 tỷ lượt.

Những yếu tố này đã giúp Android trở thành nền tảng điện thoại thông minh phổ biến nhất thế giới, vượt qua Symbian OS vào quý 4 năm 2010, 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. Bản chất mở của Android cũng khích lệ một đội ngũ đông đảo lập trình viên và những

người đam mê sử dụng mã nguồn mở để tạo ra những dự án do cộng đồng quản lý. Những dự án này bổ sung các tính năng cao cấp cho những người dùng thích tìm tòi hoặc đưa Android vào các thiết bị ban đầu chạy hệ điều hành khác.

Android chiếm 87,7% thị phần điện thoại thông minh trên toàn thế giới vào thời điểm quý 2 năm 2017, với tổng cộng 2 tỷ thiết bị đã được kích hoạt và 1,3 triệu lượt kích hoạt mỗi ngày. Sự thành công của hệ điều hành cũng khiến nó trở thành mục tiêu trong các vụ kiện liên quan đến bằng phát minh, góp mặt trong cái gọi là “cuộc chiến điện thoại thông minh” giữa các công ty công nghệ.

1.1.2. Ưu điểm của hệ điều hành Android

- Giao diện dễ sử dụng, khả năng tùy biến cao, bạn có thể tùy ý chỉnh sửa mọi thứ trên hệ điều hành android, trừ các phân vùng trong hệ thống.
- Android có sự phổ biến rất cao bởi hệ điều hành này len lỏi ở mọi thiết bị di động, từ điện thoại cao cấp, trung cấp cho đến những sản phẩm có giá phổ thông.
- Kho ứng dụng Google khổng lồ, cập nhật liên tục, với đủ mọi ứng dụng mà bạn thích, khám phá.
- Hệ điều hành Android ngày nay có khả năng tối ưu hóa hiệu suất smartphone đem đến trải nghiệm mượt mà nhất cho người dùng.
- Giá thành phải chăng.
- Sử dụng nhiều ứng dụng cùng lúc, đa nhiệm cao.

1.1.3. Nhược điểm của hệ điều hành Android

- Dễ bị nhiễm các phần mềm độc hại cùng các mã độc, do hệ điều hành Android cho phép bạn cài đặt các ứng dụng bên ngoài Google Play.
- Quá nhiều ứng dụng trên Google Play dẫn đến kiểm soát chất lượng rất khó.
- Sự phân mảnh lớn của hệ điều hành android.
- Khả năng cập nhật hệ điều hành chậm chạp.
- Thường xuyên bị đầy bộ nhớ đệm RAM.

1.1.4. Các đặc điểm của Android

Đặc điểm	Mô tả
<i>UI đẹp</i>	Màn hình Android OS cơ bản cung cấp một giao diện người dùng đẹp và có tính thẩm mỹ cao
<i>Connectivity</i>	GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC và WiMAX.
<i>Lưu trữ</i>	SQLite, một relational database gọn nhẹ, được sử dụng cho mục đích lưu trữ dữ liệu
<i>Hỗ trợ Media</i>	H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG PNG, GIF, và BMP
<i>Thông báo</i>	SMS và MMS
<i>Trình duyệt Web</i>	Dựa trên thiết bị WebKit mã nguồn mở, đi kèm với thiết bị V8 JavaScript của Chrome hỗ trợ HTML5 và CSS3
<i>Multi-touch</i>	Android hỗ trợ cho multi-touch mà đã được tạo ban đầu có sẵn cho các Handset như HTC Hero
<i>Đa nhiệm</i>	Người dùng có thể nhảy từ một tác vụ tới tác vụ khác và nhiều ứng dụng đa dạng có thể chạy đồng thời cùng một lúc

<i>Widget tùy chỉnh</i>	Widgets có thể thay đổi kích cỡ, vì thế người dùng có thể mở rộng để hiển thị nhiều nội dung hơn, hoặc thu nhỏ để tiết kiệm không gian
<i>Đa ngôn ngữ</i>	Hỗ trợ text đơn hướng và song hướng
<i>GCM</i>	Google Cloud Messaging (GCM) là một dịch vụ cho phép lập trình viên gửi thông điệp dữ liệu ngắn tới người dùng trên thiết bị Android, mà không cần một Sync Solution
<i>Wi-Fi Direct</i>	Một công nghệ cho phép các ứng dụng dò tìm và ghép cặp một cách trực tiếp, thông qua một kết nối peer-to-peer
<i>Android Beam</i>	Một công nghệ dựa trên NFC phổ biến cho phép người dùng chia sẻ tức thì, chỉ cần kích hoạt NFC của hai điện thoại với nhau

Bảng 1.1: Các đặc điểm của Android

1.1.5. Các phiên bản Android

Tỷ lệ sử dụng các phiên bản khác nhau tính đến tháng 4 năm 2014. Phần lớn các thiết bị Android cho tới nay vẫn chạy hệ điều hành phiên bản 4.1.x Jelly Bean được phát hành ngày 9 tháng 7 năm 2012 nhờ tính ổn định và hỗ trợ tốt các máy có cấu hình thấp.

Phiên bản	Tên mã	Ngày phát hành	Cấp API	Phân bố (20 tháng 7 năm 2014)
5.0	Lollipop	7/2014	20	Dành cho người phát hành
4.4	KitKat	10/2013	19	17,9%
4.3	Jelly Bean	25/7/2013	18	10,5%
4.2.x	Jelly Bean	13/11/2012	17	18,8%
4.1.x	Jelly Bean	9/7/2012	16	25,2%
4.0.x	Ice Cream Sandwich	16/12/2011	15	11,4%
3.2	Honeycomb	15/7/2011	13	0%
3.1	Honeycomb	10/ 5/2011	12	0%
2.3.3– 2.3.7	Gingerbread	9/2/2011	10	13%
2.3– 2.3.2	Gingerbread	6/12/2010	9	0,5%
2.2	Froyo	20/5/2010	8	0,7%

Phiên bản	Tên mã	Ngày phát hành	Cấp API	Phân bố (20 tháng 7 năm 2014)
2.0–2.1	Eclair	26/10/2009	7	0%
1.6	Donut	15/9/2009	4	0%

Bảng 1.2: Các phiên bản Android

1.1.6. Các thiết bị khác sử dụng hệ điều hành Android

Bản chất mở và cho phép thay đổi của Android giúp nó xuất hiện trên các thiết bị điện tử khác, như laptop và netbook, smartbook TV thông minh (Google TV) và máy ảnh (Nikon Coolpix S800c và Galaxy Camera).

Hơn thế nữa, hệ điều hành Android còn được ứng dụng trong kính mắt thông minh (Project Glass), đồng hồ đeo tay, tai nghe, đầu CD và DVD cho xe hơi, gương soi, máy nghe nhạc bỏ túi và điện thoại để bàn và VoIP. Ouya, một máy trò chơi điện tử chạy Android, đã trở thành một trong những chiến dịch khởi động thành công nhất, gây quỹ được 8,5 triệu đô la Mỹ để phát triển, tiếp sau đó là các máy trò chơi điện tử dựa trên Android như Project Shield của NVIDIA.

Vào năm 2011, Google đã trình diễn “Android@Home”, một công nghệ tự động hóa gia đình, sử dụng Android để điều khiển nhiều thiết bị gia dụng như công tắc điện, ổ cắm và thiết bị điều khiển nhiệt độ trong nhà.

Chiếc đèn mẫu được quảng cáo là có thể được điều khiển từ điện thoại hoặc máy tính bảng Android, nhưng trưởng nhóm Android Andy Rubin vẫn cảnh trọng cho rằng “tắt mở bóng đèn không phải là việc gì mới,” ám chỉ nhiều dịch vụ tự động hóa gia đình đã gặp thất bại trước đây. Ông nói rằng Google

có suy nghĩ tham vọng hơn và dự định của công ty là sử dụng vị trí của mình như một nhà cung cấp dịch vụ đám mây để mang sản phẩm Google đến gia đình của khách hàng.

1.2. Giới thiệu về lập trình Android và ứng dụng Android studio

1.2.1. Lập trình android

- Tính đến đầu năm 2016, có hơn một tỷ người tích cực sử dụng Android, một hệ điều hành di động giống Unix chạy trên điện thoại, máy tính bảng, TV thông minh, đồng hồ thông minh và hàng loạt thiết bị khác. Mặc dù lập trình Android được Google phát triển và bảo trì, nhưng đây là phần mềm miễn phí và nguồn mở có thể được sử dụng trong lập trình di động và tùy chỉnh bởi bất kỳ ai. Do đó, các thiết bị Android được sản xuất bởi một số nhà sản xuất thiết bị lớn, như Samsung, LG, Sony và HTC.
- Tính đến đầu năm 2016, có hơn một tỷ người tích cực sử dụng Android, một hệ điều hành di động giống Unix chạy trên điện thoại, máy tính bảng, TV thông minh, đồng hồ thông minh và hàng loạt thiết bị khác. Mặc dù lập trình Android được Google phát triển và bảo trì, nhưng đây là phần mềm miễn phí và nguồn mở có thể được sử dụng trong lập trình di động và tùy chỉnh bởi bất kỳ ai. Do đó, các thiết bị Android được sản xuất bởi một số nhà sản xuất thiết bị lớn, như Samsung, LG, Sony và HTC.

1.2.2. Ứng dụng Android studio

Tháng 5 năm 2013, Google công bố Android Studio, một môi trường phát triển ứng dụng tích hợp (IDE) dành riêng cho Android, mã nguồn mở, dựa trên IDE Java IntelliJ của hãng JetBrains (đối thủ với Eclipse và Netbeans, vốn khá quen thuộc với dân lập trình Java). Android Studio chạy trên Windows, Mac và Linux, nhằm thay thế cho Eclipse Android Development Tool (ADT) vốn được sử dụng làm IDE chính trong các năm trước đó.

Tính năng nổi bật

- Bộ công cụ build ứng dụng dựa trên Gradle (thay vì Maven).

- Chức năng dò và sửa lỗi nhanh, hướng Android.
- Công cụ chỉnh sửa màn hình dạng kéo thả tiện lợi.
- Các wizard tích hợp nhằm giúp lập trình viên tạo ứng dụng từ mẫu có sẵn.
- Tích hợp Google Cloud Platform, dễ dàng tích hợp với Google Cloud Messaging và App Engine của Google.

Và đi kèm với Android Studio là Android SDK. Nếu Android Studio là trình soạn thảo code (IDE) thì Android SDK là bộ tổng hợp các công cụ để build app, các bản mẫu máy ảo Android (sử dụng để test app) cần thiết để làm ra một ứng dụng Android hoàn chỉnh.

1.3 Giới thiệu về hệ quản trị cơ sở dữ liệu Firebase

1.3.1. Firebase là gì?

Firebase là một nền tảng giúp phát triển các ứng dụng di động trong web. Bên cạnh đó, Firebase còn được hiểu là một dịch vụ cơ sở dữ liệu hoạt động trên nền tảng đám mây cloud với hệ thống máy chủ mạnh mẽ của Google. Firebase chứa cơ sở dữ liệu mang đến khả năng code nhanh và thuận tiện hơn. Lập trình viên có thể dễ dàng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu sẵn có.

1.3.2. Ưu điểm khi sử dụng Firebase

Hiện nay, Firebase được rất nhiều lập trình viên lựa chọn vì đem lại nhiều lợi ích với các ưu điểm hơn hẳn nhiều nền tảng khác. Những ưu điểm của Firebase là:

- Sử dụng miễn phí và thuận tiện
- Dễ sử dụng và tích hợp
- Đáp ứng nhu cầu của người dùng
- Cập nhật liên tục và đa nền tảng

1.3.3 Nhược điểm của Firebase

Bên cạnh những ưu điểm trên, Firebase vẫn còn tồn tại những nhược điểm nhất định khiến các nhà phát triển khó có thể xây dựng ứng dụng một cách hoàn thiện nhất

- Giới hạn về quy mô ứng dụng
- Khả năng tùy chỉnh hạn chế
- Bảo mật còn yếu.

CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ BÀI TOÁN ĐẶT XE TAXI

2.1. Phát biểu bài toán

Nhận thấy được ứng dụng công nghệ thông tin vào đời sống thường ngày, nhu cầu di chuyển của người dùng. Mục đích cụ thể là để xây dựng một ứng dụng giúp cho việc đặt xe taxi để đi lại của người dùng được nhanh chóng, ổn định và ứng dụng trên smart phone. Hiện nay dịch vụ vận chuyển rất được phát triển với quy mô rộng lớn và nhu cầu đặt xe taxi online đang ngày càng phổ biến. Và thay vì phải ra đường vẫy xe taxi hoặc phải xin số người khác để gọi, giá cước đắt. Vì vậy chúng ta cần xây dựng thêm về các app, web đặt taxi trên nền tảng online để dễ dàng, thuận tiện hơn trong việc thu thập ý kiến cũng như đáp ứng nhu cầu di chuyển của khách hàng và tài xế.

Dựa trên nhu cầu đó em đã phát triển app đặt taxi trên nền tảng Android. App có các công năng cơ bản sau:

ĐỐI VỚI QUẢN LÝ

- Quản lý được tài khoản của khách hàng .
- Quản lý được tài khoản của tài xế.
- Thêm, sửa, xóa được tài khoản khách hàng.
- Thêm, sửa, xóa tài khoản tài xế.
- Liên hệ với tài xế.
- Liên hệ với khách hàng.
- Xem được lịch sử chuyến đi của khách hàng và tài xế.

ĐỐI VỚI KHÁCH HÀNG

- Khách hàng mới cũng như người dùng cũ có thể đặt taxi mọi lúc mọi nơi, thuận tiện cho người bận công việc.
- Khách hàng có thể xem thông tin về quãng đường di chuyển, số tiền phải thanh toán, thông tin cơ bản của tài xế như số điện thoại, họ tên, xe và biển số.
- Khi khách hàng đặt xe, khách hàng cần làm theo hướng dẫn của app

ĐỐI VỚI TÀI XẾ

- Xem thông tin quãng đường đón, trả khách, cước phí được nhận sau khi đã khấu trừ.
- Biết thông tin khách hàng gồm họ tên khách hàng, số điện thoại khách hàng, thanh toán bằng tiền mặt hay thẻ.
- Nắm rõ được thu nhập của mình theo ngày, tuần, tháng.

2.2. Mô tả bài toán

Dựa trên các công năng được mô tả. Em đã phát triển chính vào các công năng sau của app:

Các chức năng cơ bản của app:

- **Đăng nhập:** Cho phép tài xế đăng nhập vào hệ thống, khi đăng nhập thành công người quản trị sẽ sử dụng được các chức năng của hệ thống.
- **Đăng kí:** Cho phép tài xế đăng kí sau đó đăng nhập vào hệ thống, khi đăng nhập thành công người quản trị sẽ sử dụng được các chức năng của hệ thống.
- **Tìm kiếm:** Tìm kiếm các vị trí trên bản đồ...
- **Quản lý doanh thu:** Xem doanh thu trong ngày tuần, tháng.
- **Liên hệ với khách hàng:** Cho phép xem thông tin của khách hàng gửi tới như họ tên, số điện thoại khách hàng.

Giao diện khách hàng:

- + Có chức năng đăng nhập, đăng kí tài khoản trên hệ thống của app.
- + Hiện thị vị trí đang đứng trên map.
- + Tìm kiếm các vị trí trên map, các ưu đãi.
- + Cho phép đặt xe, biết quãng đường di chuyển và giá cước.
- + Cho phép gửi liên hệ, thắc mắc tới tài xế hoặc nhân viên chăm sóc khách hàng của công ty app.

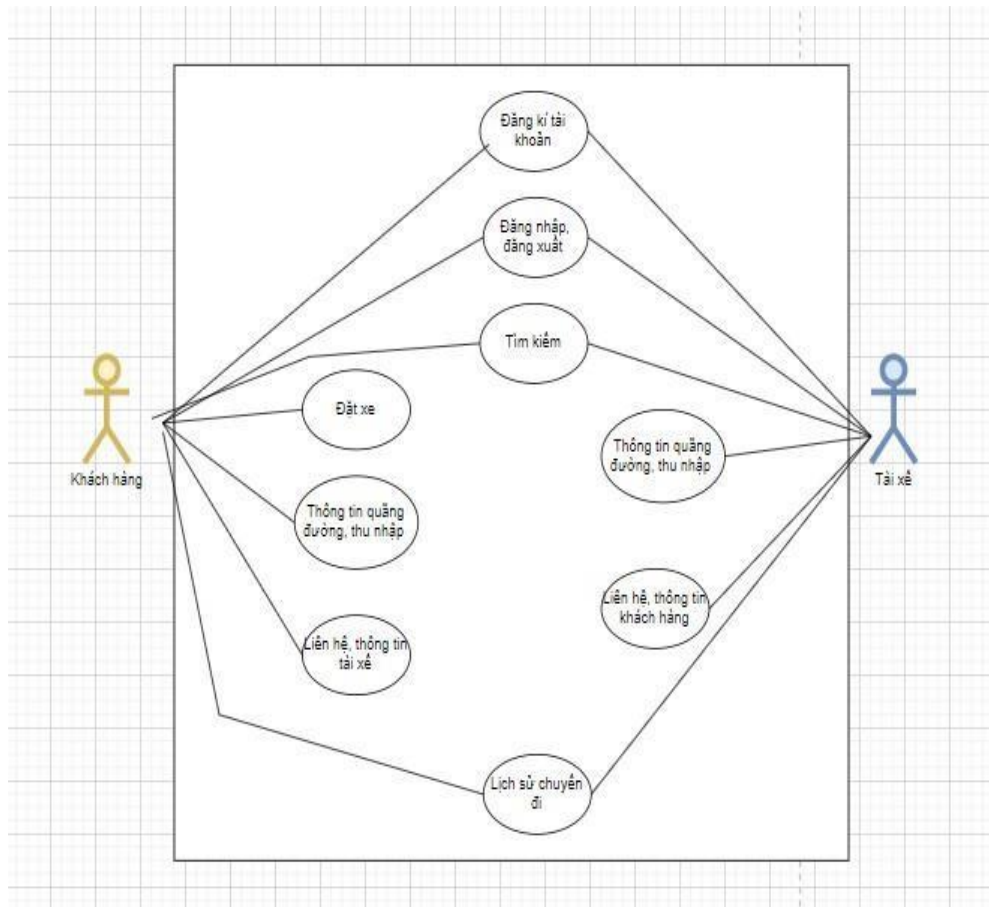
2.3. Thiết kế mô hình hệ thống

2.3.1 Xác định các tác nhân (Actor) và chức năng (Usecase)

STT	Tác nhân	Use case
1	Khách hàng	<ul style="list-style-type: none"> - Đăng kí - Đăng nhập - Đăng xuất - Đặt xe - Thông tin quãng đường, giá cước - Liên hệ với tài xế, thông tin tài xế - Tìm kiếm - Xem lịch sử chuyến đi
2	Tài xế	<ul style="list-style-type: none"> - Đăng ký - Đăng nhập - Đăng xuất - Xem chi tiết quãng đường, thu nhập - Xem thông tin khách hàng, liên hệ - Xem lịch sử chuyến đi - Tìm kiếm

Bảng 2.1: Bảng xác định tác nhân và chức năng

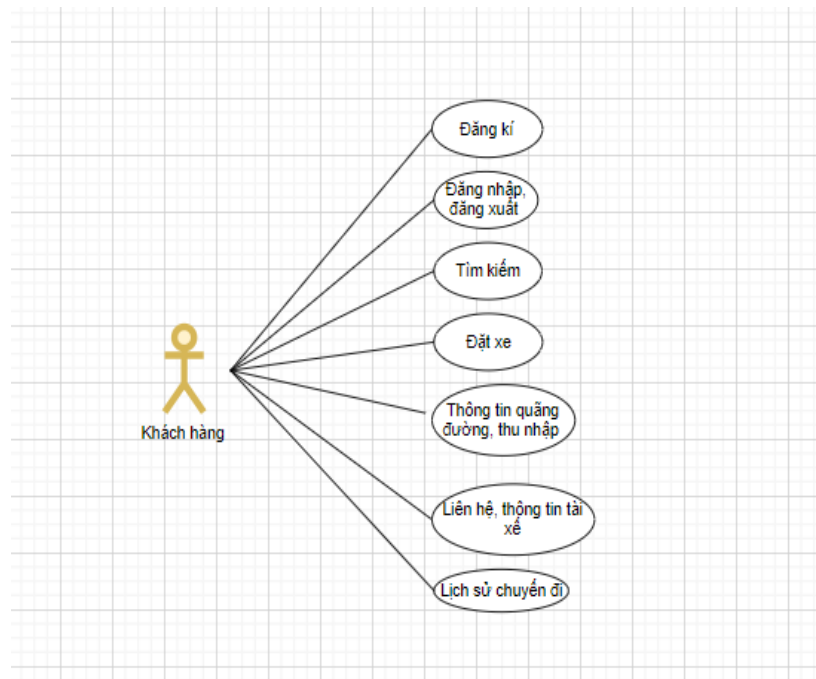
2.3.2 Biểu đồ tổng quát



Hình 2.1 Sơ đồ User tổng quát

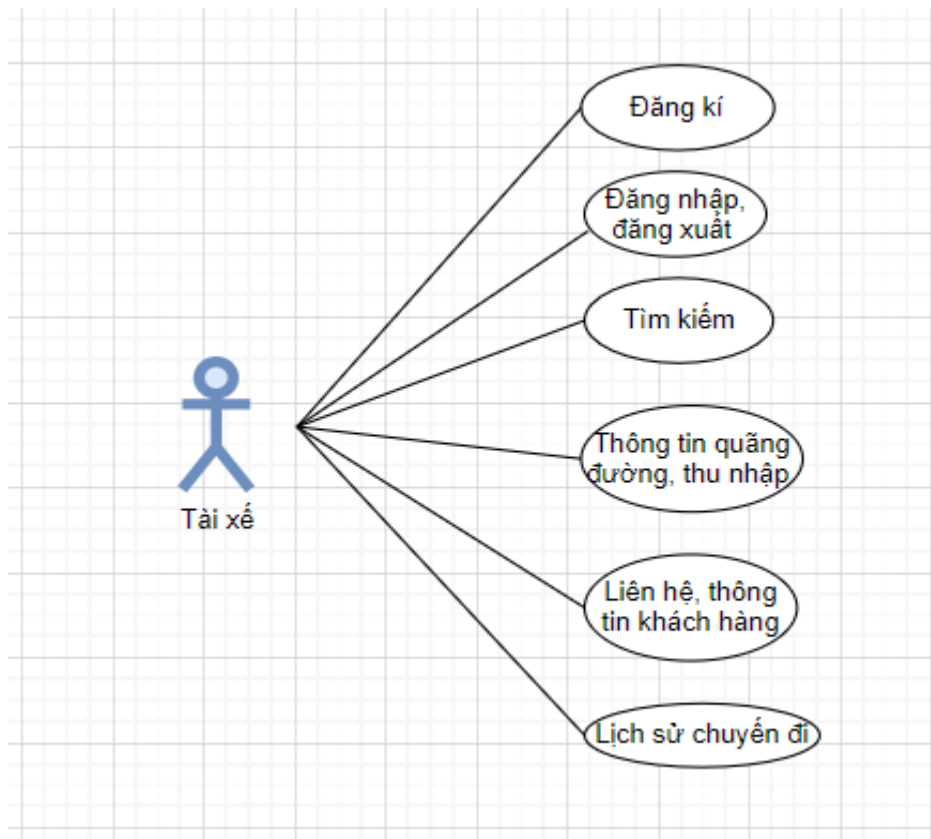
2.3.3 Use case dành cho các tác nhân

Use case dành cho tác nhân “Khách hàng”



Hình 2.2: Sơ đồ Use case dành cho tác nhân "Khách hàng"

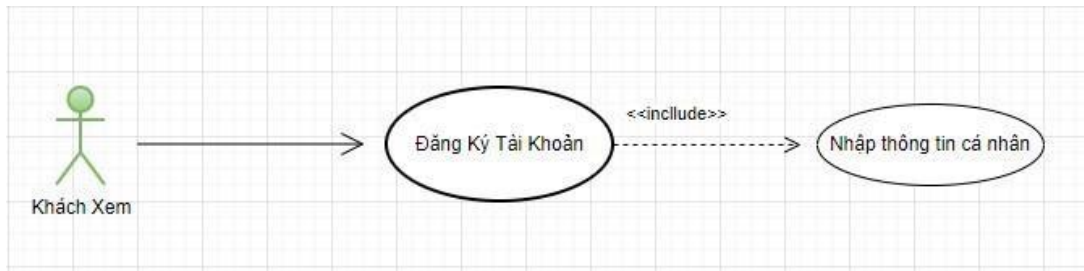
Use case dành cho tài xế



Hình 2.3: Sơ đồ use case dành cho tác nhân "Tài xế"

2.3.4 Biểu đồ use case chi tiết

Biểu đồ use case chi tiết đăng kí tài khoản



Hình 2.4: Sơ đồ Use case chi tiết dành cho tác nhân “Đăng ký tài khoản”

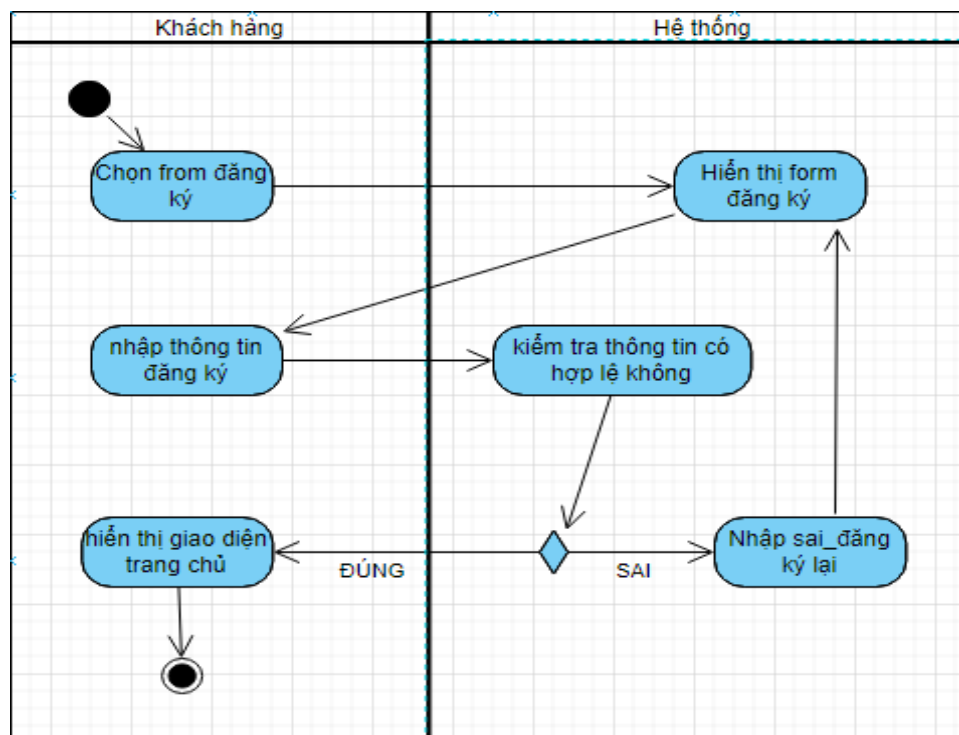
- Đặc tả chức năng

Tên use case	Đăng ký
Tác nhân	Khách hàng, tài xế (chưa có tài khoản)
Mục đích	Đăng ký tài khoản để đăng nhập vào app
Mô tả chung	Khách hàng đăng ký làm thành viên của app Tài xế đăng ký làm tài xế của hãng
Điều kiện	Khách hàng chưa có tài khoản Tài xế chưa có tài khoản
Luồng sự kiện	<ul style="list-style-type: none"> -Tại trang chủ chọn chức năng đăng ký - Form đăng ký sẽ hiển thị lên - Nhập thông tin đăng nhập gồm tên tài khoản và mật khẩu, số điện thoại vào form đăng ký - Hệ thống sẽ tiến hành kiểm tra các thông tin nếu đúng thì đăng ký thành công và hệ thống sẽ lưu

	thông tin khách hàng vào CSDL và hiển thị trang chủ của app
Ngoại lệ	+ nhập đầy đủ các thông tin yêu cầu + bỏ trống thông tin, yêu cầu nhập đầy đủ

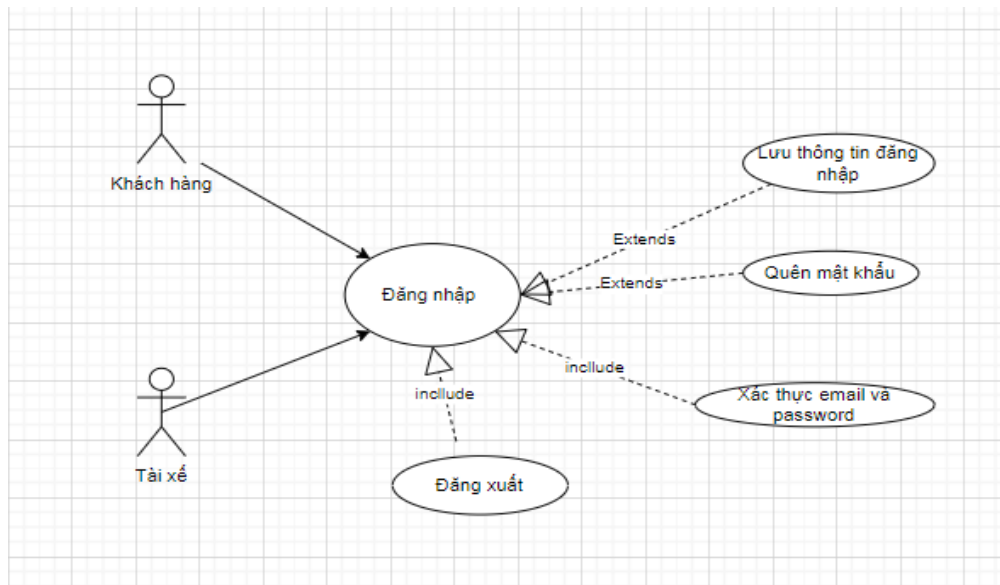
Bảng 2.2 Bảng đặc tả chức năng đăng ký

- Biểu đồ hoạt động



Hình 2.5: Biểu đồ hoạt động đăng ký

Biểu đồ chi tiết use case đăng nhập



Hình 2.6: Sơ đồ Use case dành cho tác nhân “Đăng nhập”

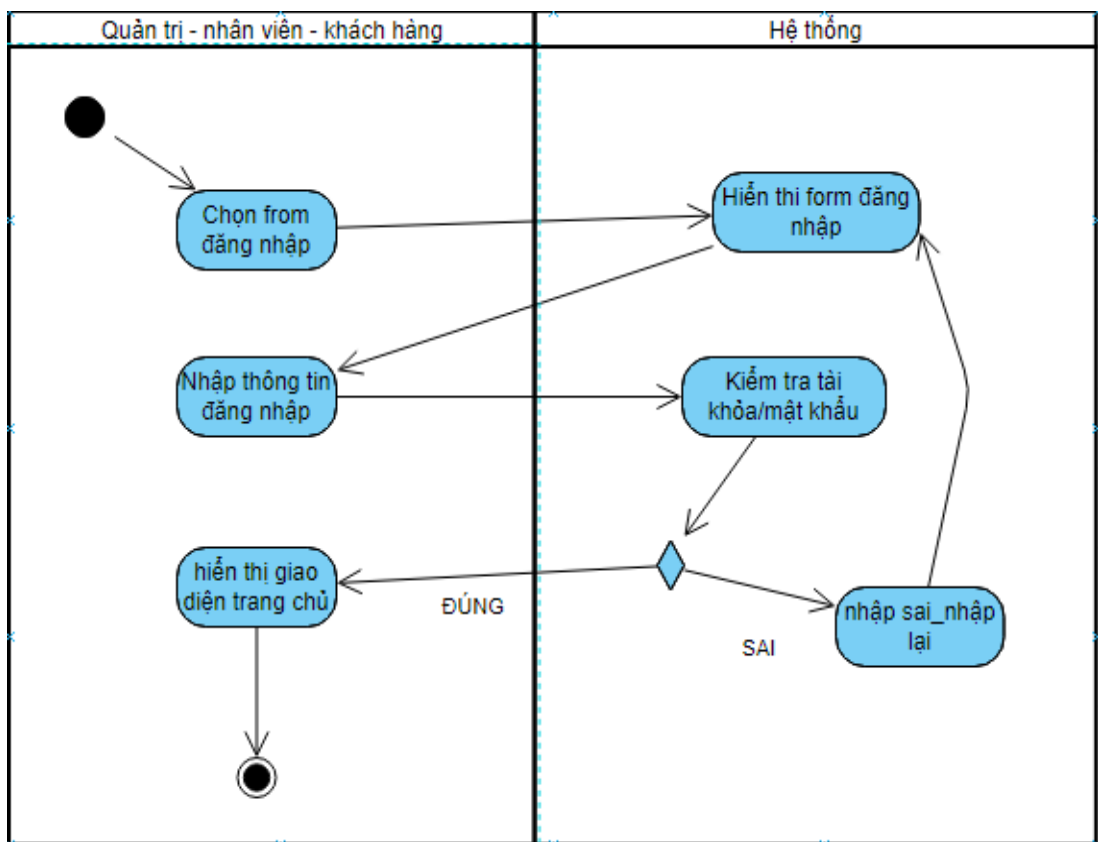
Đặc tả chức năng

Tên Use case	Đăng nhập
Tác nhân	Quản lý, tài xế, khách hàng
Mục đích	Cho phép người dùng truy cập vào hệ thống
Điều kiện	Đã có tài khoản
Mô tả chung	Khách hàng đăng nhập vào hệ thống bằng account và password
Luồng sự kiện	<p>tại trang chủ chọn chức năng đăng nhập</p> <p>form đăng nhập sẽ hiển thị</p> <p>Tiến hành nhập thông tin đăng nhập account và password</p>

	hệ thống sẽ kiểm tra, nếu đúng thì cho phép truy cập và sử dụng các chức năng
Ngoại lệ	Nhập sai account và password, đăng nhập thất bại

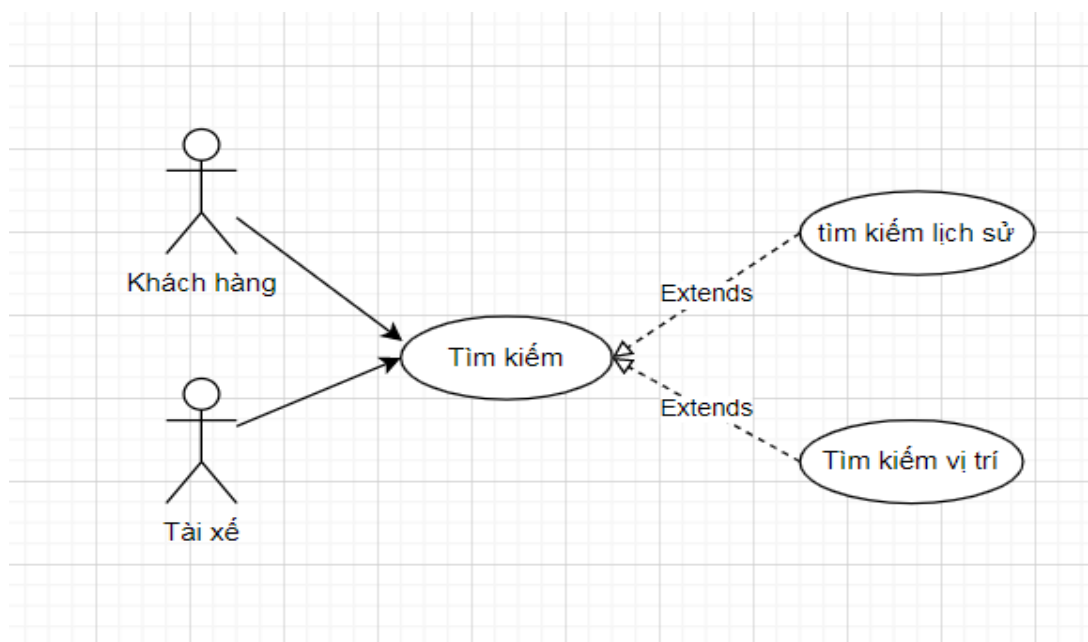
Bảng 2.3: Bảng đặc tả chức năng đăng nhập

- Biểu đồ hoạt động



Hình 2.7: Biểu đồ hoạt động đăng nhập

Biểu đồ use case chi tiết chức năng tìm kiếm



Hình 2.8: Sơ đồ use case tác nhân tìm kiếm

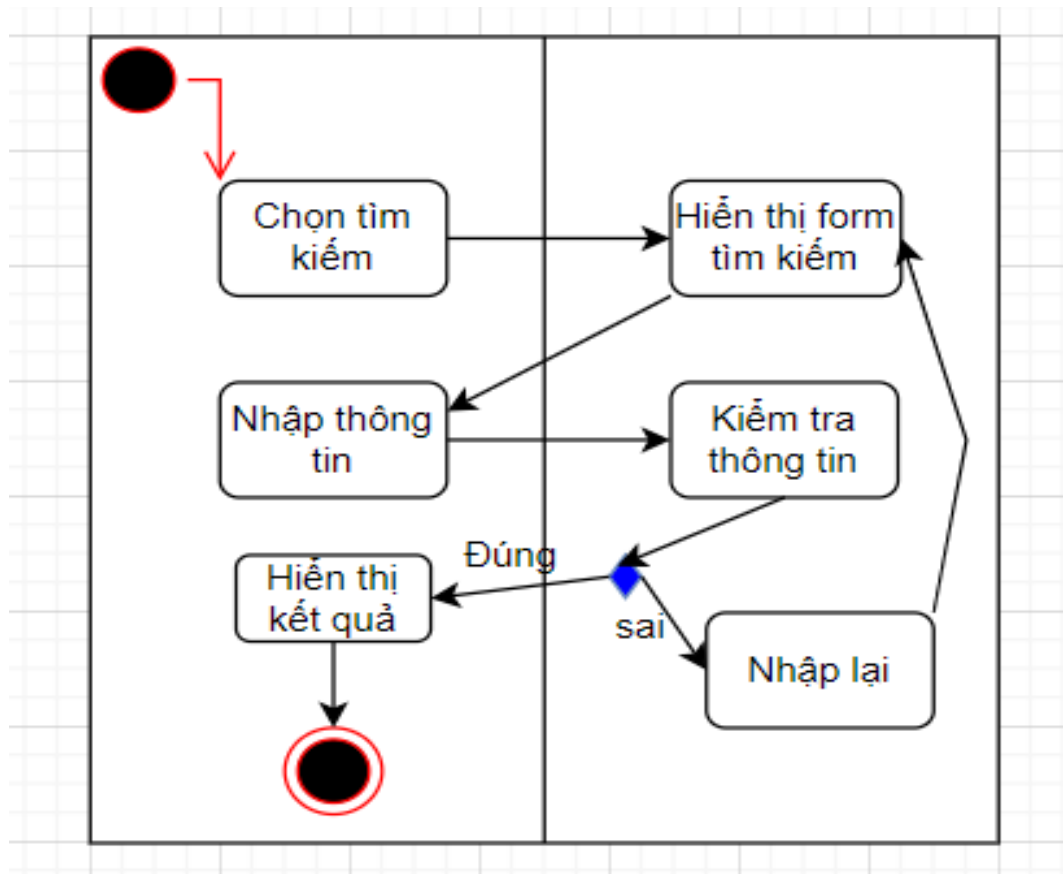
Đặc tả chức năng

Tên Use case	Tìm kiếm
Tác nhân	Quản lý, tài xế, khách hàng
Mục đích	Cho phép người dùng tìm kiếm
Điều kiện	Đã có tài khoản
Mô tả chung	Khách hàng và tài xế sử dụng tìm kiếm
Luồng sự kiện	-tại trang chủ chọn chức năng tìm kiếm

	<ul style="list-style-type: none"> form tìm kiếm sẽ hiển thị Tiến hành nhập thông tin
Ngoại lệ	Nhập sai thông tin

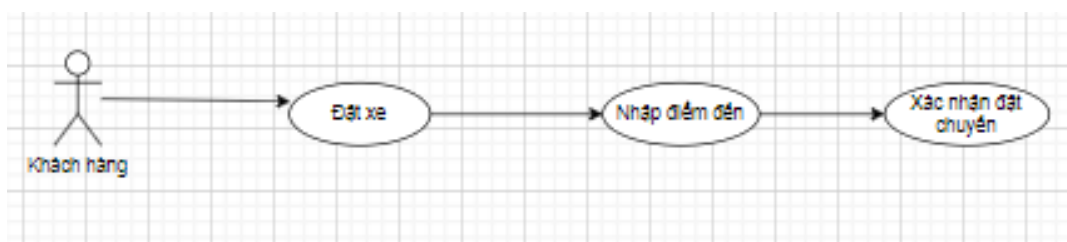
Bảng 2.4: Bảng đặc tả chức năng tìm kiếm

- Biểu đồ hoạt động



Hình 2.9: Biểu đồ hoạt động tìm kiếm

Biểu đồ use case chi tiết chức năng đặt xe



Hình 2.10: Sơ đồ use case tác nhân đặt xe

Đặc tả chức năng

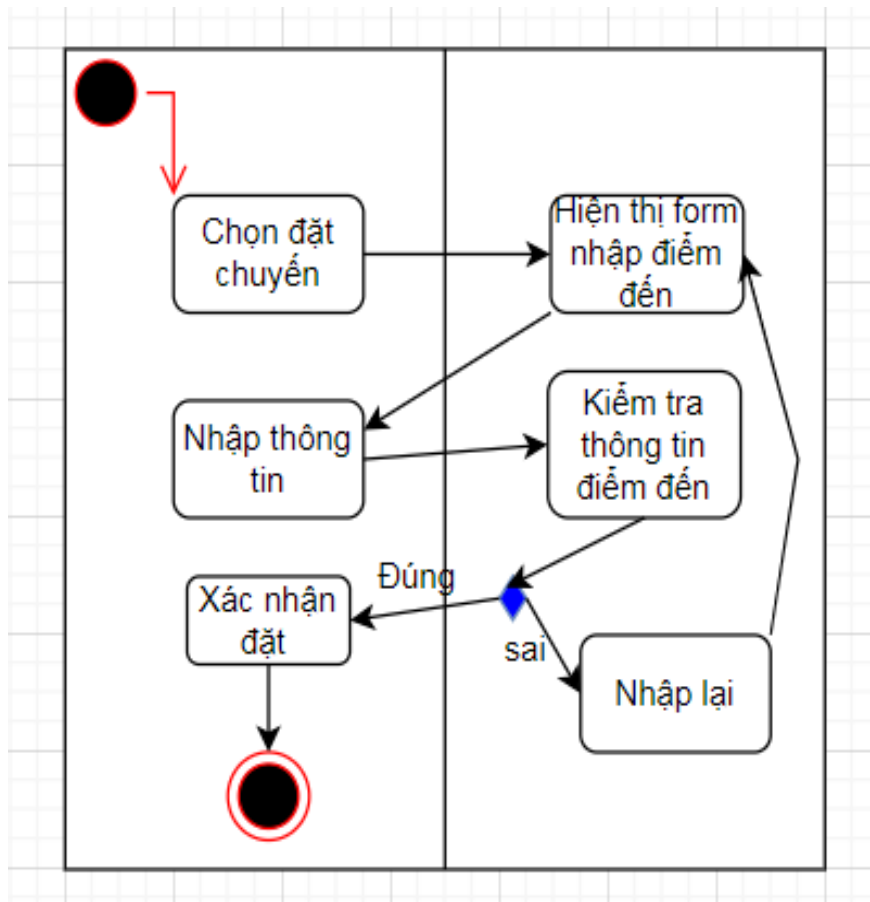
Tên Use case	Đặt xe
Tác nhân	khách hàng

Mục đích	Cho phép người dùng đặt chuyển
-------------	-----------------------------------

Điều kiện	Đã có tài khoản
Mô tả chung	Khách hàng đặt xe để di chuyển
Luồng sự kiện	-tại trang chủ chọn đặt xe - form điền điểm đến sẽ hiển thị - Tiến hành nhập thông tin - Xác nhận đặt
Ngoại lệ	Nhập sai thông tin

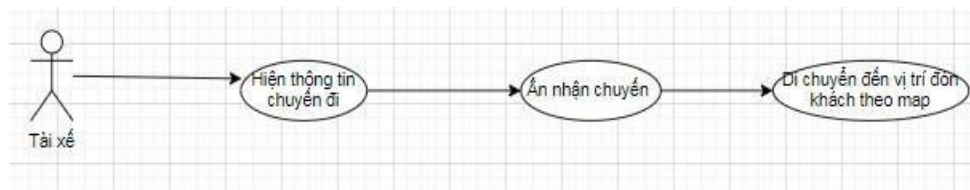
Bảng 2.5: Bảng đặc tả chức năng đặt xe

- Biểu đồ hoạt động



Hình 2.11: Biểu đồ hoạt động đặt xe

Biểu đồ use case chi tiết chức năng nhận chuyển



Hình 2.12: Sơ đồ use case tác nhân nhận chuyển

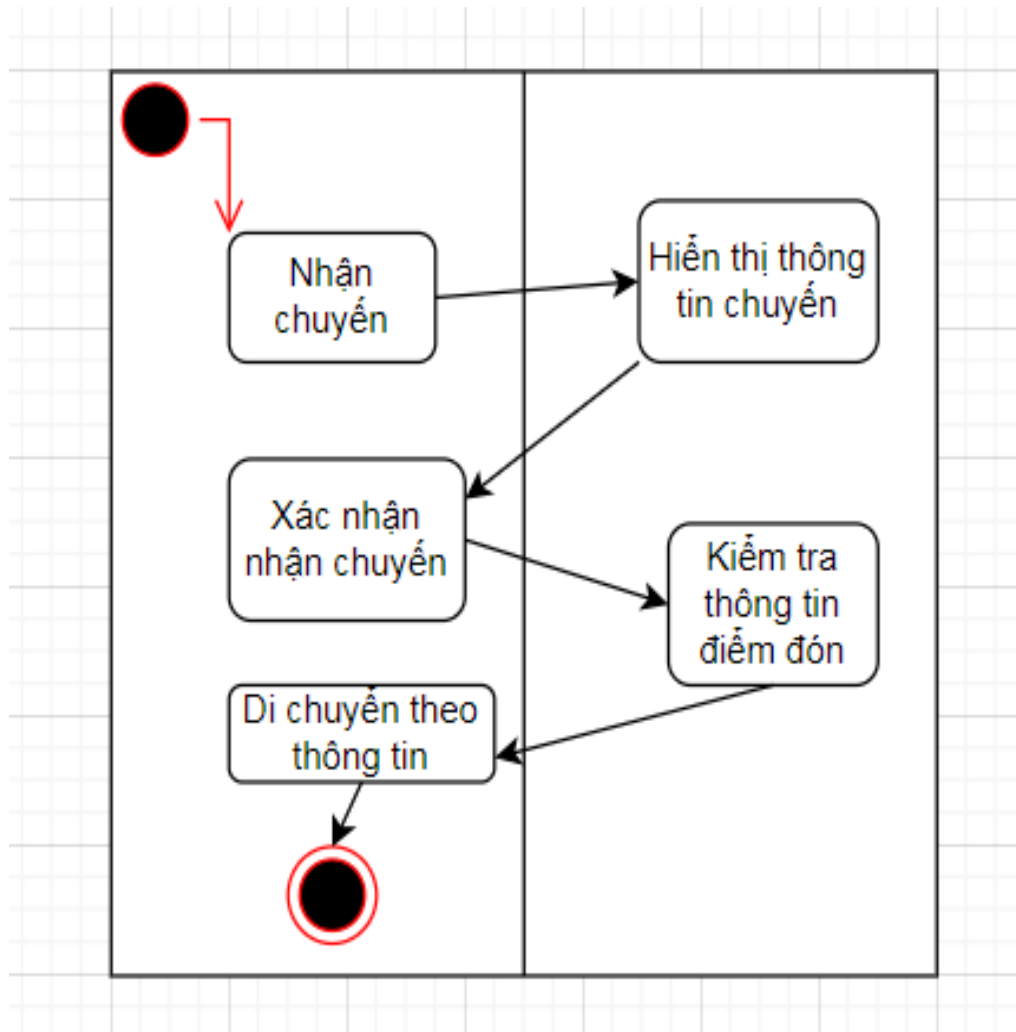
Đặc tả chức năng

Tên Use case	Nhận chuyển
Tác nhân	tài xế
Mục đích	Cho phép tài xế nhận chuyển

Điều kiện	Đã có tài khoản tài xế
Mô tả chung	tài xế nhận nhận chuyển
Luồng sự kiện	<ul style="list-style-type: none"> - tài xế nhận xác nhận nhận chuyển - địa chỉ người đặt sẽ hiện lên - Tiến hành đón khách theo địa chỉ
Ngoại lệ	

Bảng 2.6: Bảng đặc tả chức năng nhận chuyển

- Biểu đồ hoạt động

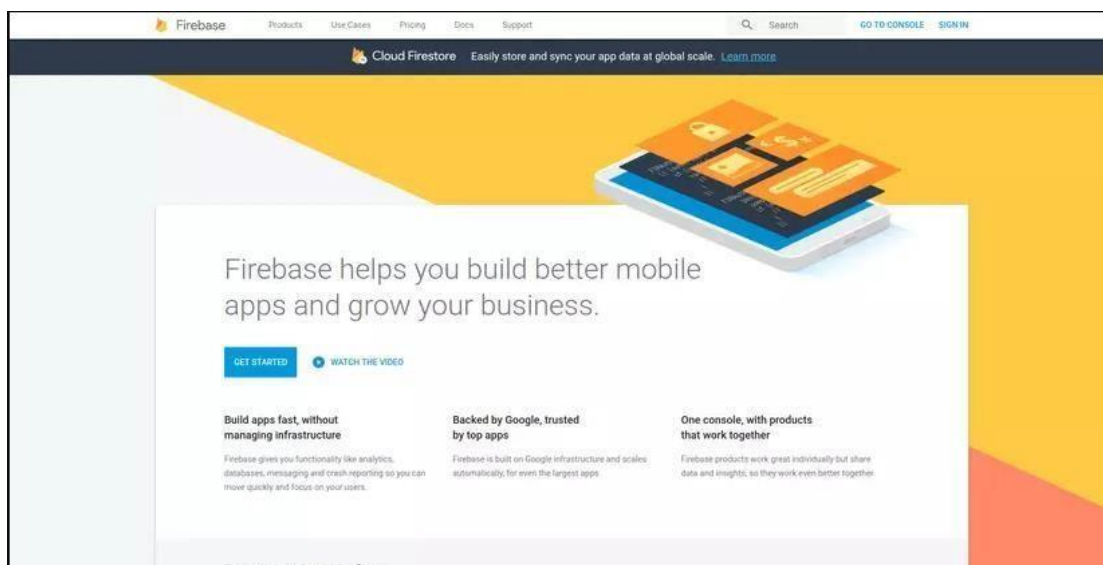


Hình 2.13: Biểu đồ hoạt động nhận chuyển

2.4. Các bước tạo cơ sở dữ liệu trên Firebase

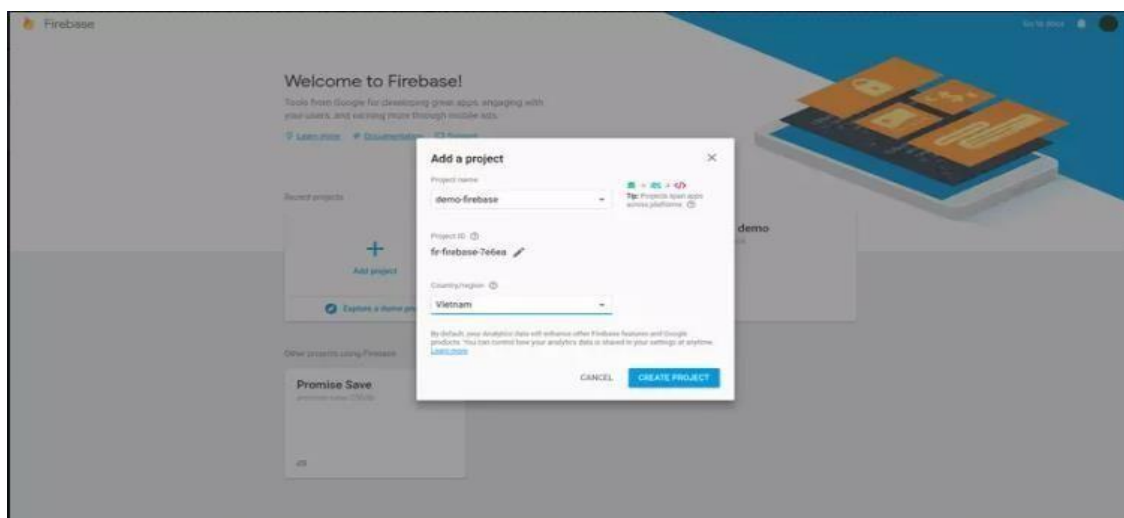
2.4.1 Tạo một project mới trên Firebase

Bước 1: Tạo một tài khoản trên Firebase hoặc bạn có thể login vào Firebase bằng tài khoản gmail.



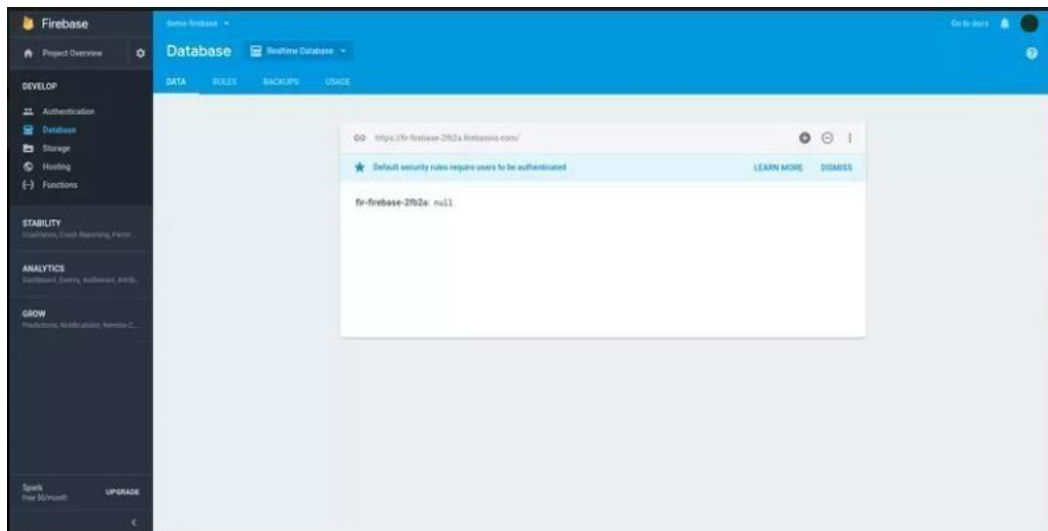
Hình 2.14 Trang chủ của Firebase

Bước 2: Sau khi đã đăng nhập tài khoản Firebase. Chúng ta tạo một project mới trên Firebase. Ở đây mình tạo một project có tên demo-firebase. Chúng ta viết vào dòng Project name và chọn CREAT PROJECT.



Hình 2.15 Tạo mới một Project

Sau khi đã tạo project mới chúng ta sẽ được chuyển vào trang dashboard của project vừa mới tạo. Phía menu bên trái, chúng ta chọn Develop -> Database



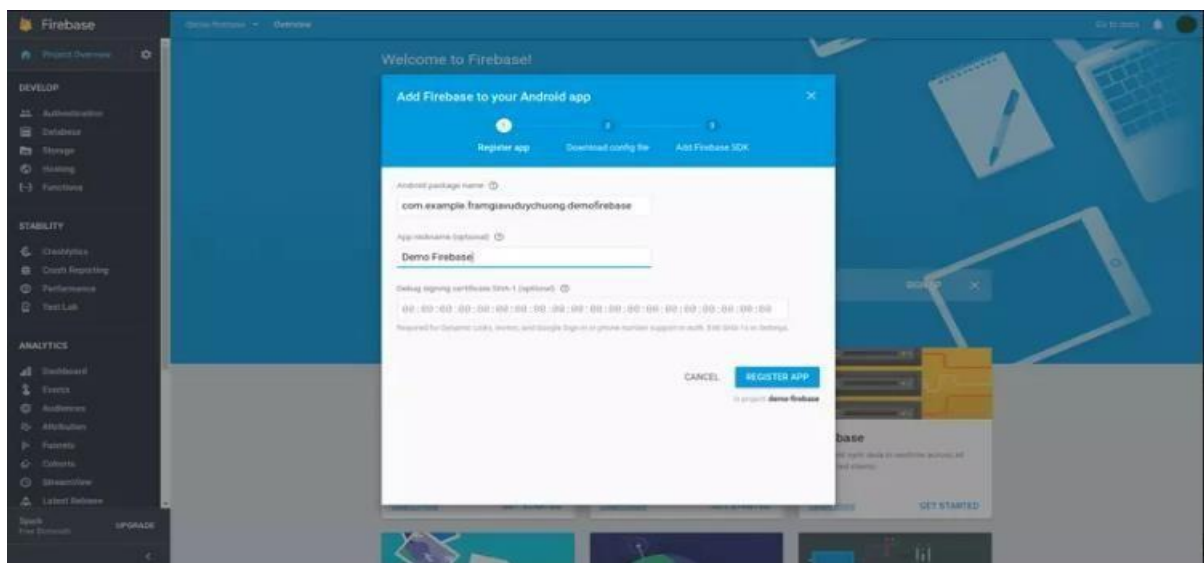
Hình 2.16: Tạo mới một Project

Toàn bộ dữ liệu của project sẽ được hiển thị ở phần Data. Như vậy chúng ta đã tạo xong một project mới trên Firebase.

2.4.2 Cấu hình Firebase lên Android studio

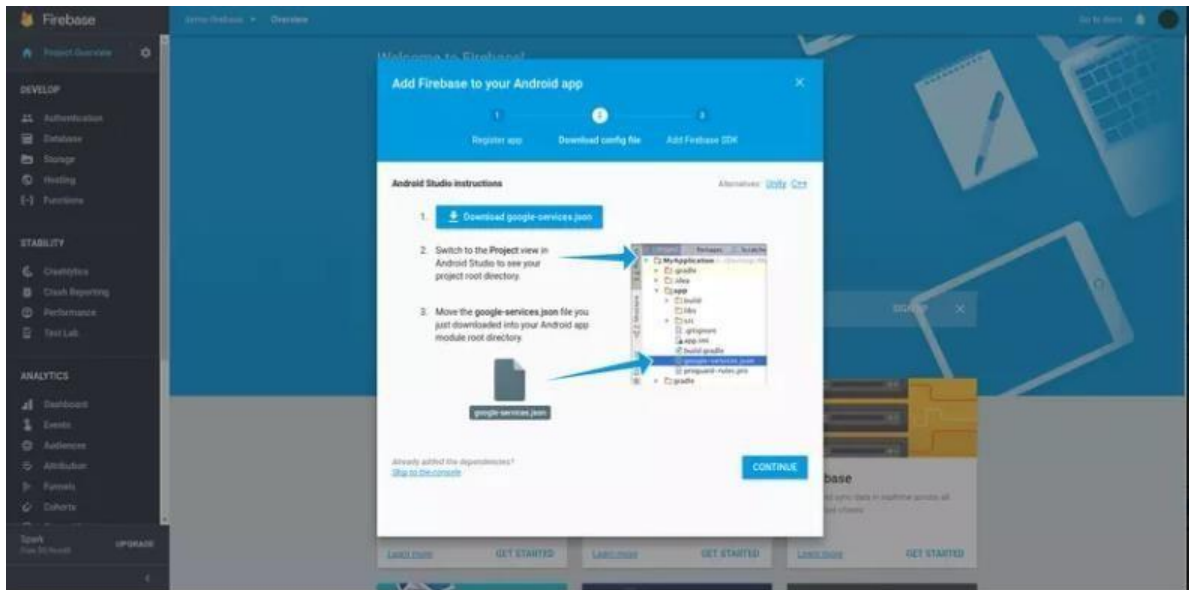
Bước 1: Mở Android Studio, tạo một project mới và chọn MainActivity dạng Empty Activity.

Bước 2: Tại console của Firebase, vào mục project overview sau đó ấn vào nút chọn tích hợp vào Android. Sau đó có một sổ hiển thị để chúng ta nhập package name và project name. Nhập vào tên package và tên project giống như project vừa tạo ở Android Studio.



Hình 2.17: Tạo mới một Project

Sau đó tải file `google-services.json` và copy vào trong thư mục app ở Android Studio



Hình 2.18: Tạo mới một Project

Bước 3: Thực hiện cấu hình thư viện trên Android Studio Trong Android Studio, mở file `build.gradle` (project) và thêm dòng sau:

```
buildscript {  
    dependencies {  
        // Add this line  
        classpath 'com.google.gms:google-services:3.1.0'  
    }  
}
```

Trong file `build.gradle` (app) thêm dòng sau vào cuối:

```
dependencies {  
    //Add this line  
    compile 'com.google.firebase:firebase-database:11.8.0'  
}  
apply plugin: 'com.google.gms.google-services'
```

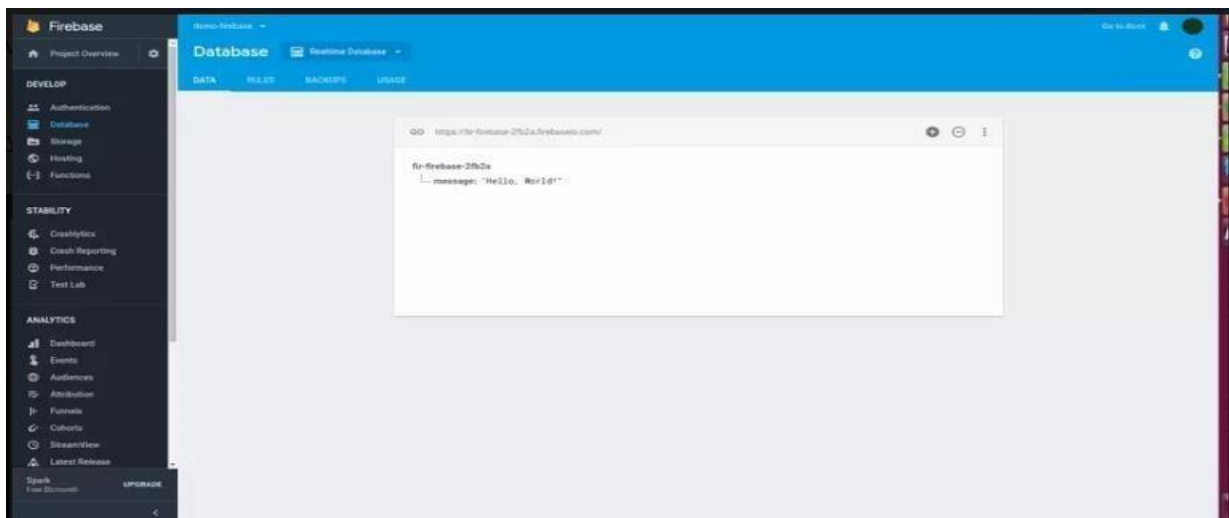
Tại phía góc trên bên phải của Android Studio ấn **Sync Now** Vậy là ta đã hoàn thành cấu hình FireBase lên Android studio. Tiếp theo chúng ta sẽ code một ví dụ về lấy dữ liệu từ Firebase về và add lên ListView trong Androidstudio.

2.4.3 Đọc, ghi dữ liệu Firebase trên Android

- Ghi dữ liệu. Trên Android studio ta vào MainActivity. Tại hàm onCreate ta gọi hàm

```
// Write a message to the database
FirebaseDatabase database = FirebaseDatabase.getInstance();
DatabaseReference myRef =
database.getReference("message");myRef.setValue("Hello,
```

Sau đó chạy thử chương trình thì sẽ thấy ở trên database đã được thêm dữ liệu vào



Hình 2.19: Tạo mới một Project

Trong trường hợp dữ liệu không được ghi vào thì chúng ta có thể sửa lại RULES trong tab RULES để có thể ghi được mà không cần Authentication

```
{
  "rules": {
    ".read": true,
    ".write": true
```

```
}  
}
```

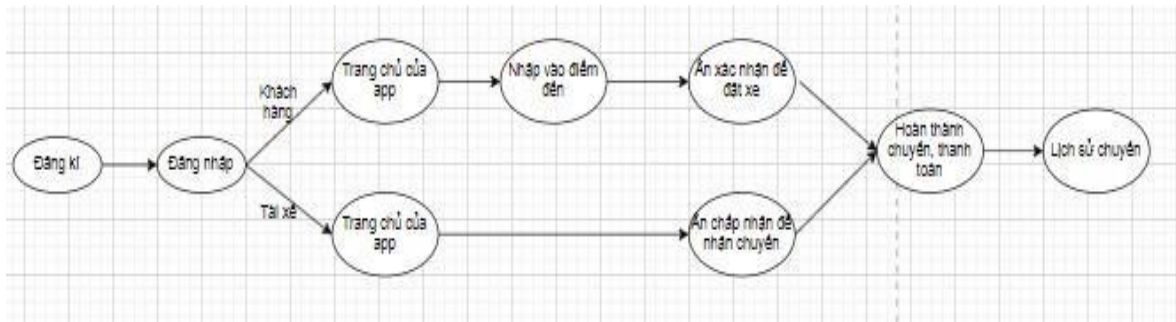
- Đọc dữ liệu Chúng ta có thể đọc dữ liệu bằng cách thêm ValueEventListener vào reference

```
FirebaseDatabase database = FirebaseDatabase.getInstance();  
DatabaseReference myRef =  
    database.getReference("message");  
  
// Read from the database  
myRef.addValueEventListener(new ValueEventListener()  
{  
    @Override  
    public void onDataChange(DataSnapshot dataSnapshot) {  
        // This method is called once with the initial value and again  
        // whenever data at this location is updated.  
        String value =  
            dataSnapshot.getValue(String.class); Log.d(TAG,  
                "Value is: " + value);  
    }  
  
    @Override  
    public void onCancelled(DatabaseError error) {  
        // Failed to read value  
        Log.w(TAG, "Failed to read value.", error.toException());  
    }  
}
```


CHƯƠNG 3. KẾT QUẢ ĐẠT ĐƯỢC TRONG PHẦN MỀM

3.1. Sơ đồ công năng chính của phần mềm

Dưới đây là sơ đồ công năng sử dụng chính của phần mềm

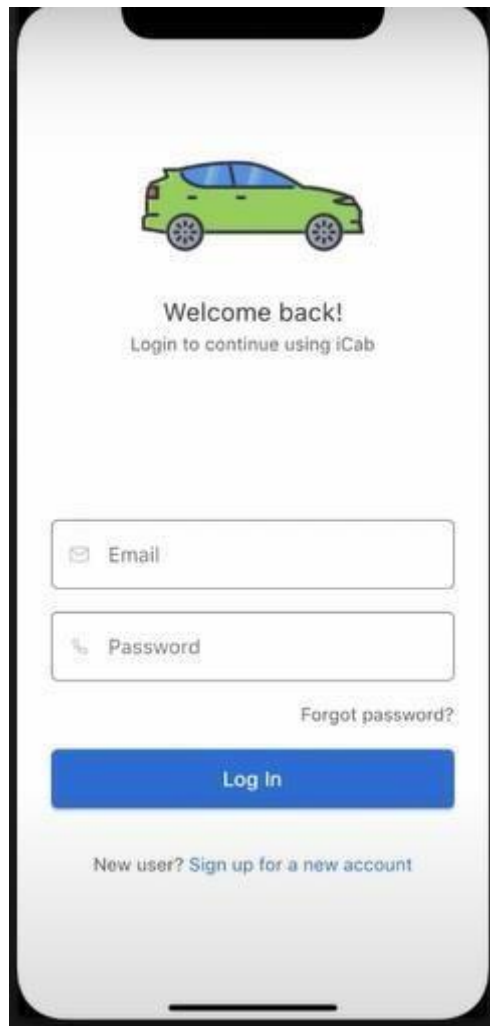


Hình 3.1: Sơ đồ công năng chính của phần mềm

3.2. Chi tiết về kết quả

3.2.1 Giao diện của người dùng

Khi bước vào ứng dụng thì phần đầu tiên người dùng sẽ thấy là phần đăng nhập. Ở đây chúng ta sẽ thấy có Email và Pass Word cho người dùng đã có tài khoản nhập



Hình 3.2: Giao diện đăng nhập người dùng

Module code của đăng nhập

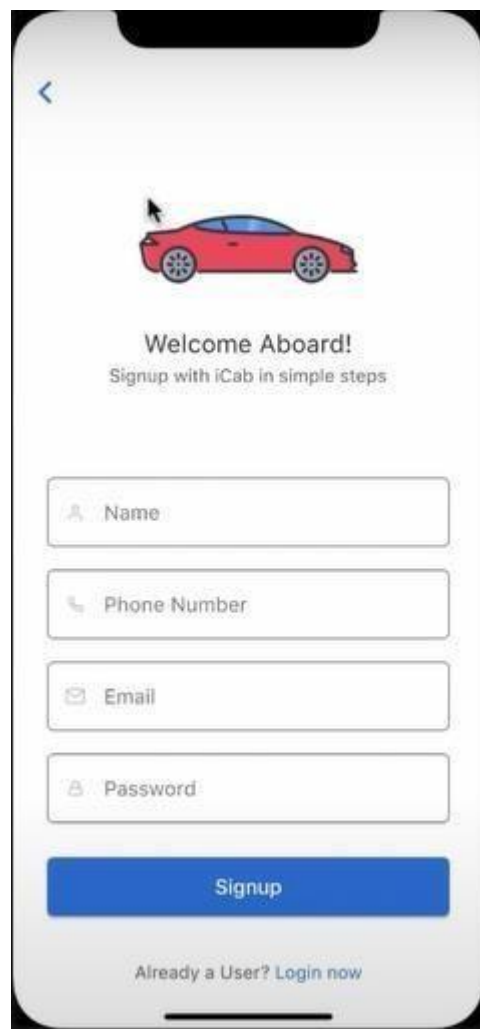
```
isActive: data['is_active'] ??
false,firstname: data['firstname']
?? '',
lastname: data['lastname'] ?? '',
email: data['email'] ??
'',createdAt:
DateTime.fromMillisecondsSinceEpoch(data['createdAt'].millisecondsSinceEpoch),
isVerified: data['is_verified'] ?? false,
licensePlate: data['license_plate'] ??
'',phone: data['phone'] ?? '',
vehicleType: data['vehicle_type'] ??
'',
vehicleColor: data['vehicle_color'] ?? ''
```

```
latlng: data['latlng'] == null ? LatLng(0, 0) :  
LatLng(data['latlng']['lat'],data['latlng']['lng']),
```

Trên đây là đoạn code ở trang đăng nhập của người dùng. Đoạn code trên sẽ hiện thị ra form đăng ký để người dùng nhập vào. Gồm mã số, họ tên, email, số điện thoại. Và đoạn code này có chức năng check xem người dùng có nhập sai form hoặc sai thông tin hay không, nếu sai sẽ nhắc nhở nhập lại luôn.

3.2.2 Giao diện đăng kí người dùng

Tiếp đó nếu người dùng chưa có tài khoản thì người dùng sẽ nhấn vào phần đăng kí mới, ở đây người dùng sẽ nhập các thông tin các nhân vào form bên dưới



Hình 3.3: Giao diện đăng kí người dùng

Module code của đăng ký

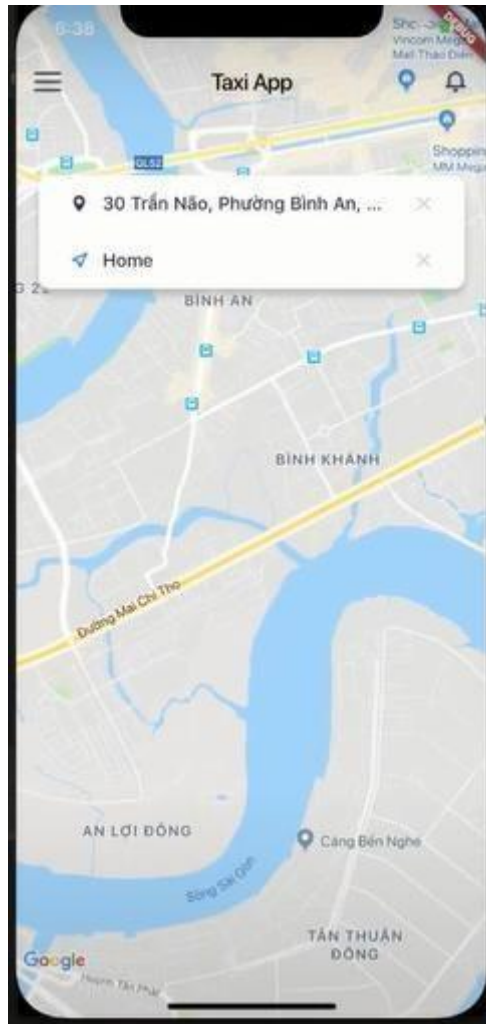
```
crossAxisAlignment: CrossAxisAlignment.start,
children: [
  const SizedBox(height: kToolbarHeight * 0.6),
  Text(
    'Enter Code',
    style: Theme.of(context).textTheme.headline5,
  ).paddingBottom(CityTheme.elementSpacing),
  Text(
    'A 6 digit code has been sent to',
    style: Theme.of(context).textTheme.bodyText1,
  ).paddingBottom(8),
  Text(
    '+234 ${state.phoneController.text}',
    style: Theme.of(context).textTheme.headline6!.copyWith(fontWeight:
FontWeight.bold),
  ).paddingBottom(CityTheme.elementSpacing),
  CityTextField(
    controller: state.otpController,
    label: 'O T P',
    keyboardType: TextInputType.phone,
  ),
  Spacer(),
  state.phoneAuthState == PhoneAuthState.loading
    ? Text(
        'Verifying...',
        style: Theme.of(context).textTheme.bodyText1!.copyWith(fontWeight:
FontWeight.w400),
      ).paddingBottom(8)
    : SizedBox.shrink(),
  state.phoneAuthState == PhoneAuthState.codeSent
    ? Row(
        children: [
          Text(
            'Resend code in ',
            style: Theme.of(context).textTheme.bodyText1!.copyWith(fontWeight:
FontWeight.w400),
          ),
```

```
Text(  
    '0:${state.timeOut}',  
    style:  
Theme.of(context).textTheme.subtitle1!.copyWith(fontWeight:  
FontWeight.bold),  
)  
],  
)paddingBottom(8)
```

Đoạn code trên là khi chúng ta đăng ký sẽ có mã otp gửi về số điện thoại mà chúng ta dùng để đăng kí. Mã OTP đó có 6 chữ số. Ở đây được sử dụng cho trường hợp bảo mật cá nhân, hoặc chúng ta có thể hiểu được mục đích của chức năng này là để xác thực người dùng.

3.2.3. Giao diện trang chủ

Sau khi đăng nhập thành công sẽ hiện ra trang chủ người dùng



Hình 3.4: Giao diện trang chủ người dùng

- Code trang chủ, menu, google map

```
class GoogleMapKey {
    static const String key = 'AizaSyCbIWmCy0PNQI7TD4xxOBq15S8aMNI0ry4';
}
```

Đoạn code này là Call API key google map. Có chức năng cho phép phần mềm sử dụng google map.

```
Id: data['id'] ?? '',
price: data['price'] ?? 0.0,
timeOfArrival: data['time_of_arrival'] ??
DateTime.now(),title: data['ride_type'],
icon: getIcon(data['id']),
```

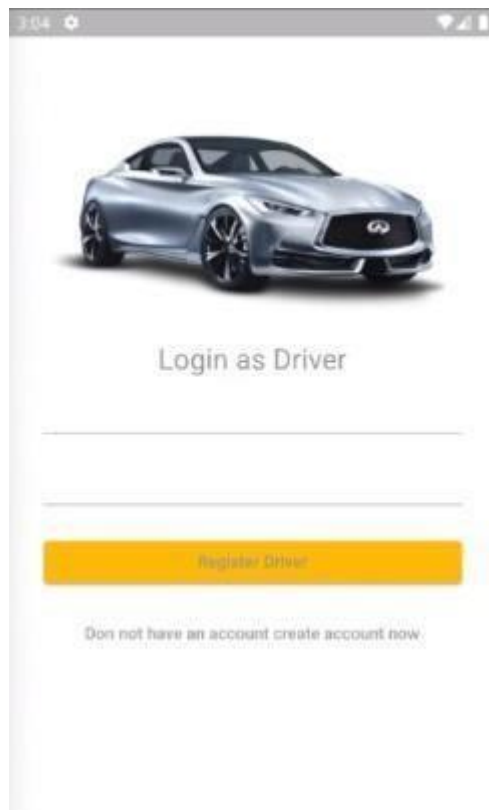
Đoạn code trên đây là thông tin của tài xế, khách hàng thấy sau khi được nhận chuyển, sẽ hiển thị ra mã số chuyển đi, giá tiền, thời gian dự kiến.

```
id: data['id'],  
city: data['city'] ?? '',  
country: data['country'] ?? '',  
latLng: LatLng(data['latlng']['lat'],  
data['latlng']['lng']),polylines: [],  
postcode: data['post_code'] ?? '',  
state: data['state'] ?? '',  
street: data['street'] ?? '',
```

Đoạn code này dùng để triển khai các địa điểm, đường phố, địa chỉ, mã số, hiển thị thông tin địa điểm trên phần mềm.

3.2.4. Giao diện đăng nhập Tài xế

Khi tài xế nhấp vào ứng dụng sẽ hiện ra trang đăng nhập đầu tiên



Hình 3.5: Giao diện đăng nhập tài xế

Module code đăng nhập tài xế

```
CustomTextFormFieldWidget(controller: emailController, hintText: "Your Email",  
obscureText: false, keyboardType: TextInputType.emailAddress,),
```

```
const SizedBox(height: 12,),  
CustomTextFormFieldWidget(controller: passwordController, hintText: "Your  
Password", obscureText: true, keyboardType: TextInputType.text,),
```

Dòng code trên là dòng code quan trọng để tạo UI cho trang đăng nhập. Dùng để tạo form nhập thông tin Email và Password.

3.2.5 Giao diện Đăng kí tài xế



Hình 3.6: Giao diện đăng kí tài xế

Module code quan trọng

```
TextEditingController nameController = TextEditingController();  
TextEditingController emailController = TextEditingController();  
TextEditingController passwordController = TextEditingController();  
TextEditingController phoneController = TextEditingController();
```

Code khai báo các form có trong đăng kí tài xế mới. Gồm có tên tài xế, email tài xế, password và số điện thoại tài xế


```
CustomTextFormFieldWidget(controller:  
  nameController,hinText: “Your name”,  
  obscureText: false,  
  keyboardType: TextInputType.text,  
),
```

Code khai báo thành phần tên tài xế, dạng text có nghĩa là khi tài xế nhấn vào form tên thì sẽ hiện ra phần chữ trên bảng chữ điện thoại.

```
CustomTextFormFieldWidget(controller:  
  emailController,hinText: “Your Email”,  
  obscureText: false,  
  keyboardType: TextInputType.emailAddress,  
),
```

Code khai báo email, TextInputType.emailAddress, sẽ gợi ý người dùng điền @ ngay khi mở bàn phím điện thoại lên.

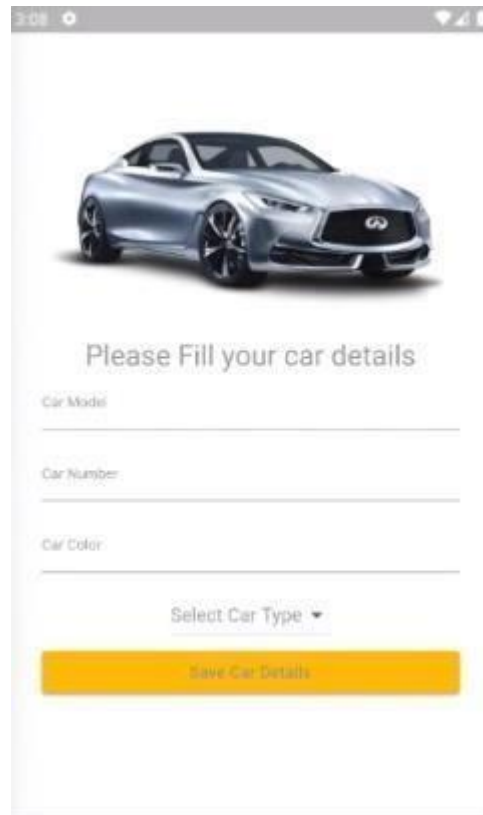
```
CustomTextFormFieldWidget(controller:  
  phoneController,hinText: “Your Phone”,  
  obscureText: false,  
  keyboardType: TextInputType.phone,  
),
```

Code có chức năng khai báo số điện thoại, dùng TextInputType.phone, có chức năng hiện bàn phím dạng số khi người dùng nhấn vào form Phone.

```
CustomTextFormFieldWidget(controller:  
  passwordController,hinText: “Your Password”,  
  obscureText: true,  
  keyboardType: TextInputType.text,  
),
```

Chức năng của đoạn code trên là khai báo password email, và obscureText: true, có chức năng làm ẩn đi dòng chữ khi người dùng nhập, nhằm tránh bị ăn cắp tài khoản do sơ ý.

3.2.6. Giao diện đăng kí loại xe



Hình 3.7: Giao diện đăng kí loại xe

Module code đăng kí xe

```
TextEditingController carModelController = TextEditingController();  
TextEditingController carNumberController =  
TextEditingController();TextEditingController carTypeController =  
TextEditingController(); TextEditingController carColorController =
```

Code khai báo các chức năng của trang đăng kí xe. Người dùng sẽ đăng kí mẫu xe, biển số xe, kiểu xe, màu xe.

CHƯƠNG 4. KẾT LUẬN

4.1. Bài học kinh nghiệm

Sau khi thực hiện đồ án xây dựng ứng dụng di động, em đã thu được rất nhiều kinh nghiệm quý báu. Đầu tiên, em đã hiểu rõ hơn về quy trình tạo ra một ứng dụng di động. Em đã học cách xác định các yếu tố chính để xây dựng app như: tính năng, giao diện, tốc độ và bảo mật.

Thứ hai, em đã nhận thấy tầm quan trọng của việc tương tác với khách hàng. Em đã học được cách lắng nghe và đáp ứng yêu cầu của khách hàng, từ đó cải thiện được sản phẩm và dịch vụ mà mình cung cấp.

Chính vì vậy, để trở thành một lập trình viên giỏi, cần phải có kiến thức về từng nghiệp vụ cụ thể trong hệ thống và tìm hiểu kỹ các ứng dụng được sử dụng để xây dựng phần mềm.

4.2. Về kỹ năng

Những kỹ năng đã được rèn luyện qua đợt làm đồ án:

- Kỹ năng làm liên tục
- Kỹ năng học hỏi liên tục
- Kỹ năng Thuyết trình
- Kỹ năng khi thiết kế giao diện app

Công việc cụ thể mà thông qua đó SV được rèn luyện các kỹ năng trên:

- Xây dựng app đặt taxi.
- Thuyết trình bảo vệ dự án của mình.
- Được tìm hiểu về ngôn ngữ lập trình mới.

4.3. Về phẩm chất đạo đức nghề nghiệp

Những phẩm chất, đạo đức nghề nghiệp được rèn luyện trong đợt thực tập:

- Tính kỷ luật, đúng giờ, chấp hành nội quy doanh nghiệp, tác phong công

việc, yêu nghề, tuân thủ luật về CNTT, an ninh mạng, chịu khó tìm tòi.

- Có tinh thần trách nhiệm với công việc được giao
- Học được tính kiên nhẫn khi gặp vấn đề mới.
- Học được cách quản lý thời gian và cách sử dụng thời gian sao cho hiệu quả.

4.4. Những thu hoạch bổ ích nhất với bản thân

- Hiểu biết hơn về cách thức làm việc của công ty.
- Hiểu biết hơn môi trường làm việc khắc nghiệt của các công ty.
- Hiểu biết thêm về nền tảng Android.
- Được tranning các kiến thức về quá trình tạo ra sản phẩm phần mềm.

4.5. Những hạn chế của bản thân

- Kiến thức chuyên môn chưa đủ đáp ứng như cầu làm việc.
- Kỹ năng tiếng anh chưa tốt.
- Kỹ năng lập trình còn trung bình.
- Mắc những lỗi tiêu chuẩn của một app.

4.6. Dự kiến tiếp theo

- Tích lũy thêm kiến thức để làm tốt hơn cho công việc tương lai.
- Định hướng rõ ràng hơn về hướng đi trong ngành.
- Cố gắng trang bị đầy đủ tư liệu, kinh nghiệm, kiến thức khi ra trường.

TÀI LIỆU THAM KHẢO

Danh mục các website tham khảo:

- [1] <https://flutter.dev/>
- [2] <https://www.youtube.com/watch?v=yYHU1qH9Cao>
- [3] https://www.gosell.vn/blog/thiet-ke-mobile-app/#Cung_cap_dieu_huong_don_gian

Tiếng Anh

- [1] Flutter Succinctly (2019) by Ed Freitas, American.
- [2] Flutter UI Succinctly (2021) by Ed Freitas, American.