

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

ĐỒ ÁN TỐT NGHIỆP

Xây dựng hệ thống quản lý vận chuyển hành khách

NGUYỄN ĐÌNH DƯƠNG

duong.nd173071@sis.hust.edu.vn

Ngành Công nghệ thông tin

Chuyên ngành Công nghệ thông tin

Giảng viên hướng dẫn: PGS.TS. Trần Đình Khang _____

Bộ môn: Hệ thống thông tin

Viện: Công nghệ thông tin – Truyền thông

HÀ NỘI, 02/2022

Lời cam kết

Họ và tên sinh viên: Nguyễn Đình Dương

Điện thoại liên lạc: 0352748300

Email: duong.nd173071@sis.hust.edu.vn

Lớp: CNTT.09 - K62

Hệ đào tạo: Cử nhân kỹ thuật

Tôi – *Nguyễn Đình Dương* – cam kết Đồ án Tốt nghiệp (ĐATN) là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *PGS.TS Trần Đình Khang*. Các kết quả nêu trong ĐATN là trung thực, là thành quả của riêng tôi, không sao chép theo bất kỳ công trình nào khác. Tất cả những thông tin tham khảo trong ĐATN – bao gồm hình ảnh, bảng biểu, số liệu – đều được ghi rõ ràng và đầy đủ nguồn gốc trong danh mục tài liệu tham khảo. Tôi xin hoàn toàn chịu trách nhiệm nếu có những sai phạm vi phạm quy chế của nhà trường.

Hà Nội, ngày 09 tháng 02 năm 2022

Tác giả ĐATN

Nguyễn Đình Dương

Lời cảm ơn

Bốn năm đại học trôi qua một cách thật nhanh, tuy không dài nhưng cũng là một chặng đường đủ để bản thân em có những trải nghiệm mới, được tiếp xúc với những con người mới, tư duy mới, và hơn cả là những bài học kinh nghiệm trong cuộc sống.

Qua Đồ án tốt nghiệp này, em muốn gửi lời cảm ơn tới tất cả mọi người, tới các thầy cô, bạn bè ở Viện Công nghệ thông tin và Truyền thông – nay là Trường Công nghệ thông tin và Truyền thông – Đại học Bách khoa Hà Nội đã chỉ bảo cũng như cho em những bài học quý giá trong hơn bốn năm qua, em xin gửi lời cảm ơn đặc biệt tới gia đình - những người đã luôn ở bên, đồng hành cùng em qua mọi thử thách, mọi buồn vui. Cuối cùng, em xin gửi lời cảm ơn chân thành nhất tới thầy Trần Đình Khang đã giúp em trong quá trình thực hiện đồ án, thầy luôn có những hướng dẫn, chỉ dạy kịp thời với những thắc mắc, khó khăn mà em đã gặp phải.

Em xin chân thành cảm ơn.

Tóm tắt

Cuộc sống hiện nay ngày càng hiện đại, internet cũng như các thiết bị thông minh được ứng dụng trong rất nhiều lĩnh vực đời sống. Bên cạnh đó, dân số ngày càng đông, chính vì thế nhu cầu đi lại của người dân cũng ngày càng lớn. Tuy nhiên hiện nay đa phần người dân khi muốn di chuyển đến những tỉnh thành khác cũng như các công ty vận tải vẫn giữ hệ thống, phương pháp cũ từ ngày trước là khách sẽ gọi đặt vé thông qua số điện thoại nhà xe/phòng vé hoặc ra tận bến xe để mua vé. Chính từ những điều thực tế đó đã đặt ra cần có một hệ thống giúp các công ty vận tải quản lý hành khách, chuyển đi một cách thuận tiện hơn cũng như giúp hành khách có thể đặt vé thông qua chiếc điện thoại thông minh của mình.

Hệ thống quản lý dành cho các công ty vận tải được xây dựng trên nền tảng web, các nhân viên công ty có thể truy cập bất cứ lúc nào. Hệ thống đặt vé cho hành khách được xây dựng trên nền tảng Android và trong tương lai sẽ mở rộng thành đa nền tảng, giúp người dùng có những trải nghiệm tốt nhất, thuận tiện nhất.

Mục lục

Lời cam kết	ii
Mục lục	iv
Danh mục hình vẽ.....	vii
Danh mục bảng biểu	ix
Chương 1 Giới thiệu đề tài	1
1.1 Đặt vấn đề	1
1.2 Mục tiêu và phạm vi đề tài.....	1
1.3 Định hướng giải pháp	2
1.4 Bố cục đồ án	2
Chương 2 Khảo sát và phân tích yêu cầu	3
2.1 Khảo sát hiện trạng	3
2.2 Tổng quan chức năng.....	5
2.2.1 Biểu đồ use case tổng quan	5
2.2.2 Biểu đồ use case phân rã	6
2.2.3 Quy trình nghiệp vụ.....	11
2.3 Đặc tả chức năng.....	14
2.3.1 Đặc tả use case Đăng nhập bằng số điện thoại	14
2.3.2 Đặc tả use case Đặt vé.....	15
2.3.3 Đặc tả use case Thực hiện chuyến đi.....	17
2.3.4 Đặc tả use case Tạo chuyến đi.....	17
2.3.5 Đặc tả use case Đặt lịch chuyến đi	19
2.3.6 Đặc tả use case Đánh giá chuyến đi	20

2.3.7	Đặc tả use case Hủy vé.....	21
2.3.8	Đặc tả use case Thêm mới xe	22
2.3.9	Đặc tả use case Phản hồi khách hàng	23
2.3.10	Đặc tả use case Thêm mới công ty vận tải	24
2.4	Giải pháp ghép chuyến đi cho các công ty vận tải.....	26
2.4.1	Đặt vấn đề.....	26
2.4.2	Giải pháp đưa ra	26
2.5	Yêu cầu phi chức năng.....	28
2.5.1	Yêu cầu về bảo mật	28
2.5.2	Yêu cầu giao diện	28
2.5.3	Yêu cầu khác	29
Chương 3 Công nghệ sử dụng.....		30
3.1	Server	30
3.1.1	NodeJS.....	30
3.1.2	MySQL.....	30
3.2	Website	31
3.2.1	VueJS.....	31
3.3	App mobile	31
3.3.1	Java.....	31
3.3.2	RESTful API	32
3.3.3	Firebase	33
Chương 4 Phát triển và triển khai ứng dụng		34
4.1	Thiết kế kiến trúc	34
4.1.1	Lựa chọn kiến trúc phần mềm	34
4.1.2	Thiết kế tổng quan	36
4.2	Thiết kế chi tiết	39
4.2.1	Thiết kế giao diện	39

4.2.2 Thiết kế lớp.....	41
4.2.3 Thiết kế cơ sở dữ liệu	41
4.3 Xây dựng ứng dụng	48
4.3.1 Thư viện và công cụ sử dụng	48
4.3.2 Kết quả đạt được.....	49
4.3.3 Minh hoạ các chức năng chính	49
4.4 Kiểm thử	56
4.4.1 Kiểm thử tương thích	56
4.4.2 Kiểm thử chức năng	56
4.5 Triển khai	57
Chương 5 Kết luận và hướng phát triển	59
5.1 Kết luận.....	59
5.2 Hướng phát triển	59
Tài liệu tham khảo	60

Danh mục hình vẽ

Hình 1 Biểu đồ usecase tổng quan	5
Hình 2 Biểu đồ usecase phân rã Quản lý thông tin cá nhân.....	6
Hình 3 Biểu đồ usecase phân rã Theo dõi lịch sử hoạt động	7
Hình 4 Biểu đồ usecase phân rã Tìm kiếm	8
Hình 5 Biểu đồ usecase phân rã Đặt vé.....	9
Hình 6 Biểu đồ usecase phân rã Quản lý chuyến đi.....	10
Hình 7 Biểu đồ usecase phân rã Quản lý người dùng	11
Hình 8 Biểu đồ quy trình nghiệp vụ Đặt vé	12
Hình 9 Biểu đồ quy trình nghiệp vụ Quản lý chuyến đi	13
Hình 10 Hình minh họa chức năng “Ghép chuyến đi”	26
Hình 11 Sơ đồ thuật toán ghép xe	28
Hình 12 Mô hình Client-Server.....	31
Hình 13 Tổng quan kiến trúc phần mềm.....	34
Hình 14 Sơ đồ gói cho Front-end Web	36
Hình 15 Biểu đồ gói cho Front-end app mobile.....	37
Hình 16 Biểu đồ gói cho Back-end/Server	38
Hình 17. Giao diện màn hình app mobile	39
Hình 18. Giao diện website.....	40
Hình 19. Sơ đồ thiết kế cơ sở dữ liệu.....	41
Hình 20. Giao diện đăng nhập app mobile.....	49
Hình 21. Giao diện tìm kiếm vé xe	50

Hình 22. Giao diện chọn chuyến đi.....	51
Hình 23. Giao diện chọn vị trí ngồi.....	52
Hình 24. Giao diện chọn điểm lên/xuống	53
Hình 25. Giao diện Quản lý người dùng và Quản lý công ty vận tải của Admin hệ thống	54
Hình 26. Giao diện quản lý xe	54
Hình 27. Giao diện quản lý chuyến đi.....	55
Hình 28. Giao diện Ghép chuyến đi.....	55
Hình 29. Giao diện Đánh giá từ khách hàng và phản hồi	56

Danh mục bảng biểu

Bảng 1. Dữ liệu đầu vào use case đăng nhập bằng số điện thoại	15
Bảng 2. Dữ liệu đầu vào use case Đặt vé	16
Bảng 3. Dữ liệu đầu vào chức năng Thêm mới chuyến đi	18
Bảng 4. Dữ liệu đầu vào chức năng Đặt lịch chuyến đi	20
Bảng 5. Dữ liệu đầu vào chức năng Đánh giá chuyến đi	21
Bảng 6. Dữ liệu đầu vào chức năng Đánh giá chuyến đi	23
Bảng 7. Dữ liệu đầu vào chức năng Phản hồi khách hàng	24
Bảng 8. Dữ liệu đầu vào chức năng Thêm mới công ty vận tải	25
Bảng 9. Thiết kế chi tiết bảng users	41
Bảng 10. Thiết kế chi tiết bảng transportations.....	42
Bảng 11. Thiết kế chi tiết bảng coach_type	43
Bảng 12. Thiết kế chi tiết bảng coaches	43
Bảng 13. Thiết kế chi tiết bảng seats.....	44
Bảng 14. Thiết kế chi tiết bảng trip_seat.....	44
Bảng 15. Thiết kế chi tiết bảng provinces	45
Bảng 16. Thiết kế chi tiết bảng stations	45
Bảng 17. Thiết kế chi tiết bảng trips	45
Bảng 18. Thiết kế chi tiết bảng payments	46
Bảng 19. Thiết kế chi tiết bảng tickets	46
Bảng 20. Thiết kế chi tiết bảng reviews	47
Bảng 21. Thiết kế chi tiết bảng notifications.....	48
Bảng 22. Danh sách thư viện và công cụ sử dụng.....	48

Bảng 23. Thống kê kiểm thử tương thích.....	56
Bảng 24. Kiểm thử chức năng Đặt vé của người dùng	57
Bảng 25. Kiểm thử chức năng quản lý phía công ty vận tải	57
Bảng 26. Thống kê các thiết bị kiểm thử cho hệ thống.....	58

Chương 1 Giới thiệu đề tài

1.1 Đặt vấn đề

Từ xưa đến nay, nhu cầu đi lại của con người lúc nào cũng có và nhu cầu ấy càng ngày càng tăng cùng với sự phát triển của xã hội. Công nghệ được ứng dụng nhiều lĩnh vực trong cuộc sống và đặc biệt trong nhiều năm trở lại đây, công nghệ thông tin được ứng dụng trong ngành Giao thông vận tải, điều đó làm cho cuộc sống của con người tiện lợi hơn.

Ngày trước nếu muốn di chuyển trong tỉnh/thành phố, người ta có thể gọi taxi hay ra đường bắt xe ôm thì bây giờ chỉ cần một chiếc điện thoại, bạn có thể đặt ngay cho mình một chuyến xe đến đúng nơi yêu cầu, thậm chí, tiện lợi hơn bạn có thể trả tiền trong tài khoản mà không cần tiền mặt, hay việc bạn có thông tin đầy đủ của tài xế, vị trí của xe cũng làm cho mỗi chuyến đi trở nên an toàn hơn.

Tuy là tiện lợi như vậy nhưng để phát triển thành mô hình lớn không phải điều dễ. Xuất phát từ những điều trên DATN của em sẽ xây dựng một hệ thống cho những chuyến đi đường dài – khi hiện nay đa số người dân di chuyển qua các tỉnh vẫn dùng phương thức cũ là ra bến xe mua vé hay gọi điện cho phòng vé. Hệ thống sẽ giúp khách hàng có thể đặt vé bằng điện thoại cá nhân mà không cần phải ra bến, các nhà xe có thể giảm bớt nhân công khi khâu đặt vé đã có hệ thống hỗ trợ,...

1.2 Mục tiêu và phạm vi đề tài

Nhìn vào tình hình hiện nay, ở Việt Nam, chúng ta đã có một số bên triển khai việc mua bán vé xe đường dài online như Vexere, Vetaugiare24h, Megabus,... hay các công ty lớn như Be, Zalo cũng phát triển chức năng này và tích hợp luôn trong app. Tuy nhiên, về phía người dùng, loại hình này chưa thực sự được khách hàng quan tâm mà chỉ một số bộ phận nhỏ, chủ yếu là sinh viên biết đến và sử dụng thử. Còn về phía các công ty vận tải, hiện nay nhiều nhà xe đã hợp tác với các app và sử dụng phần mềm quản lý bán vé của bên cung cấp dịch vụ nên việc đồng bộ hệ thống mua/bán vé cũng dễ dàng hơn.

Với những vấn đề đặt ra ở mục 1.1 cũng như tình hình thực tế nêu ở trên, mục tiêu của đồ án này là xây dựng được hệ thống mua bán vé xe đường dài bao gồm phần mềm cho nhà xe để quản lý việc bán vé, doanh thu,... và phần mềm cho người dùng sử dụng để đặt vé theo nhu cầu.

1.3 Định hướng giải pháp

Từ mục tiêu đã đề ra ở mục 1.2, giao diện quản lý cho các công ty vận tải sẽ được xây dựng trên nền tảng web, với web, nhân viên nhà xe sẽ dễ dàng sử dụng, đặc biệt với các thiết bị màn hình lớn có bố cục rộng rãi sẽ giúp việc quản lý được hiệu quả, tránh những sai sót không cần thiết hơn so với làm việc qua màn hình nhỏ.

Bên cạnh đó, giao diện cho người dùng/khách hàng sẽ được xây dựng trên app mobile, cụ thể trong ĐATN này sẽ xây dựng trên Android vì với tình hình hiện nay, mỗi người đa số đều có cho mình một chiếc smartphone, nên việc xây dựng app mobile sẽ giúp phần mềm tiếp cận được với nhiều khách hàng hơn và chính nó cũng mang lại sự tiện lợi cho người dùng khi chỉ cần vài thao tác đơn giản là đã có ngay vé xe cho mình.

ĐATN sử dụng Nodejs để xây dựng phía server, MySQL làm cơ sở dữ liệu, giao diện web xây dựng bằng PHP, Android Java để xây dựng app mobile, RESTful API để giao tiếp client-server và Firebase để xác thực người dùng. Ngoài ra em sử dụng Github để quản lý mã nguồn một cách an toàn và hiệu quả.

1.4 Bố cục đồ án

Dưới đây là bố cục báo cáo đồ án tốt nghiệp này cho những phần còn lại.

Chương 2 em trình bày về khảo sát của em với các ứng dụng, hệ thống cho phép người dùng đặt vé xe đường dài. Sau đó đưa ra các phân tích, đặc tả các chức năng, nghiệp vụ sẽ xây dựng trong hệ thống của mình.

Trong Chương 3, em sẽ giới thiệu về các công nghệ em sử dụng trong đồ án tốt nghiệp này, các công nghệ để xây dựng giao diện web, giao diện app mobile, back-end và triển khai hệ thống.

Đến Chương 4, em sẽ trình bày chi tiết về phân tích thiết kế hệ thống, các thiết kế kiến trúc, database, giao diện,...

Chương 5 trình bày về những giải pháp, đóng góp nổi bật của mình trong suốt quá trình làm đồ án.

Cuối cùng, Chương 6 là chương sẽ nêu lên những thứ em đã đạt được sau khi làm ĐATN, những thiếu sót, hạn chế của hệ thống cũng như của bản thân để từ đó rút ra kinh nghiệm và có định hướng phát triển sản phẩm thêm trong tương lai.

Chương 2 Khảo sát và phân tích yêu cầu

2.1 Khảo sát hiện trạng

Hiện nay trên thị trường có một số ứng dụng cho phép người dùng đặt vé xe đường dài và dưới đây em xin đưa ra một vài so sánh giữa các ứng dụng này.

	Vexere	Be	Vetaugiare24h	Megabus
Giao diện app mobile cho người dùng	Có	Có	Không	Không
Tích hợp khai báo thông tin y tế	Không	Không	Không	Không
Thanh toán điện tử	Có	Có	Có	Có
Người dùng được chọn vị trí ghế/giường	Có	Có	Không	Có
Cho phép người dùng đăng ký gửi thêm hàng hóa, xe cộ	Không	Không	Có	Không
Cho phép người dùng đánh giá trải nghiệm của mình về các nhà xe	Có	Không	Không	Có
Thông báo cho khách hàng khi sắp đến giờ khởi hành	Có	Có	Không	Không
Phần mềm quản lý cho các nhà xe	Có	Không	Không	Không

Vexere là hệ thống thương mại dịch vụ dành cho khách hàng muốn đặt vé và cả các nhà xe (công ty vận tải) hay đại lý bán vé cho nhà xe. Đây là một hệ thống khá hoàn chỉnh và nổi

lên trong vài năm trở lại đây, tuy nhiên qua quá trình sử dụng vẫn có thể thấy một số chức năng còn phức tạp hoặc chưa được phát triển như việc tích hợp khai báo y tế thay vì sau khi khách lên xe rồi phải ghi vào giấy khai báo của nhà xe.

Be khởi điểm ban đầu là dịch vụ xe ôm, taxi điện tử. Về sau phát triển thêm nhiều chức năng khác tạo thành hệ sinh thái riêng. Tuy nhiên khi phát triển chức năng đặt vé xe đường dài thì còn chưa được chú trọng, chức năng này chỉ phát triển đơn giản là cho người dùng đặt chỗ và thanh toán tiền vé. Người dùng không có cái đánh giá khách quan về các nhà xe, bên phía nhà xe cũng chưa được hỗ trợ hệ thống quản lý.

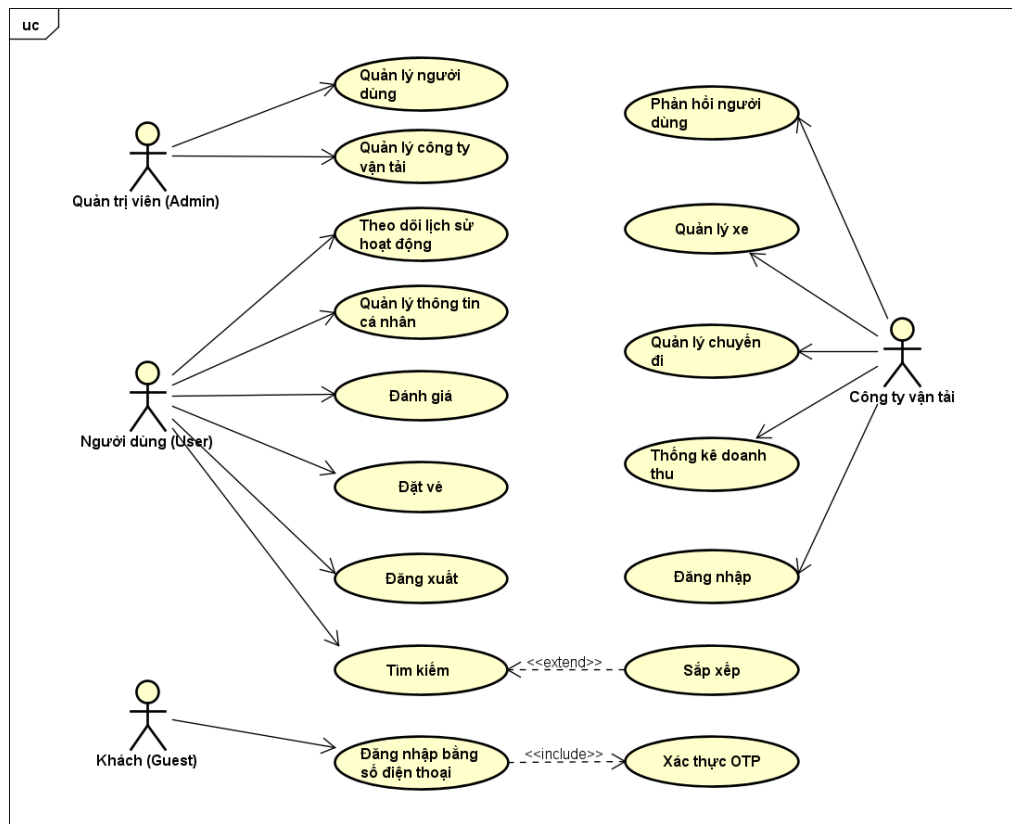
Vetaure24h là hệ thống đặt vé xe được phát triển cách đây khá lâu, vì vậy cho đến thời điểm hiện tại thì hệ thống hơi lỗi thời, giao diện nhìn chưa bắt mắt và cũng chưa có app mobile cho người dùng, khách hàng đặt vé không được chọn chỗ ngồi.

Megabus là hệ thống cho phép người dùng đặt vé xe, vé máy bay, phòng nghỉ trên cả nước. Các chức năng cơ bản vận hành tốt tuy nhiên chưa có giao diện app mobile cho người dùng, chưa tích hợp khai báo y tế trong mùa dịch, các nhà xe đăng ký còn hạn chế dẫn đến việc tìm chuyến đi còn khó khăn ở một số tuyến.

Qua tìm hiểu, so sánh những ưu, nhược điểm của các hệ thống đặt vé khác, bên cạnh các chức năng cơ bản như đặt vé, đăng nhập, đăng ký,... hệ thống của em sẽ phát triển thêm một số chức năng hỗ trợ cho người dùng như thông báo cho người dùng khi sắp đến giờ khởi hành, tích hợp khai báo thông tin y tế bằng cách quét mã QR, cho phép người dùng đánh giá các chuyến đi của mình để người dùng sau có cái nhìn khách quan hơn trước khi quyết định đặt vé của nhà xe nào,...

2.2 Tổng quan chức năng

2.2.1 Biểu đồ use case tổng quan



Hình 1 Biểu đồ usecase tổng quan

Người dùng (user) là những người đăng nhập vào hệ thống bằng số điện thoại, người dùng sẽ được quyền tìm kiếm, đặt vé xe và thực hiện một số chức năng khác.

Quản trị viên (Admin) là những tài khoản đặc biệt của hệ thống, quản trị viên sẽ là bên quản lý hệ thống: quản lý người dùng, quản lý các công ty vận tải.

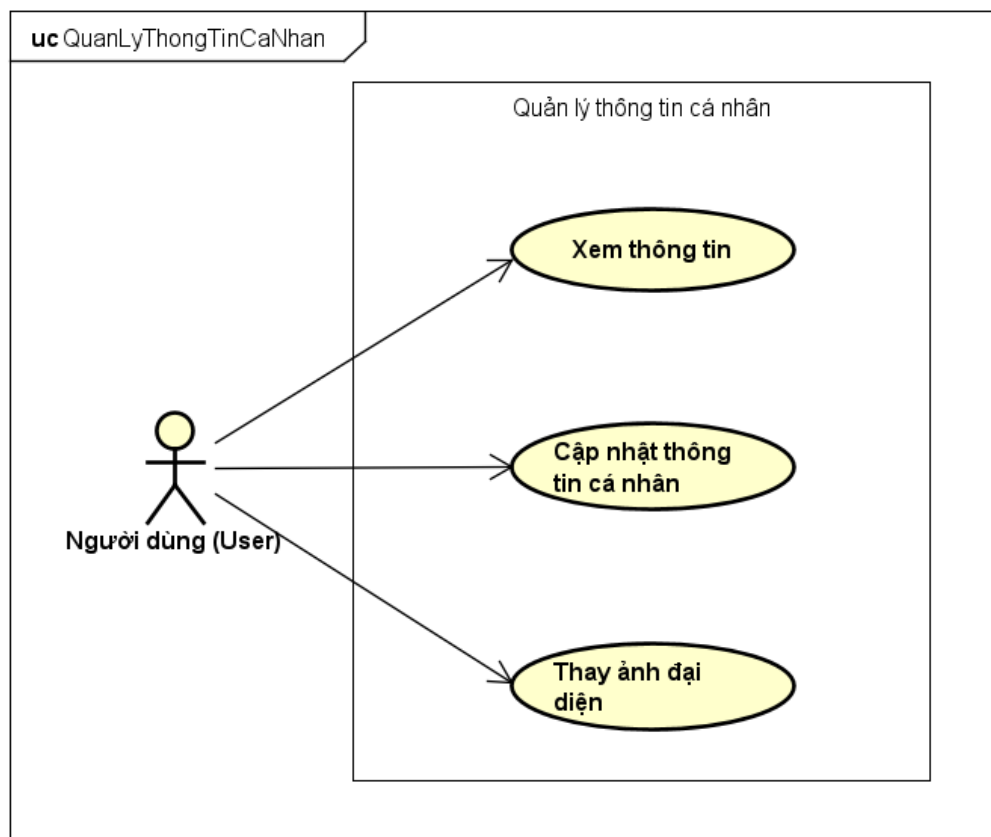
Khách (Guest) là những người truy cập vào hệ thống nhưng chưa đăng nhập, để có thể thực hiện được các chức năng khác của hệ thống, khách cần thực hiện chức năng đăng nhập.

Công ty vận tải là các nhà xe, được cung cấp tài khoản để quản lý, phản hồi người dùng, cập nhật tình hình vé xe,...

Hệ thống có chức năng chính là Đặt vé, cho phép người dùng đặt vé xe của một nhà xe bất kỳ. Bên cạnh đặt vé, người dùng cũng có thể thực hiện các chức năng liên quan như gửi phản hồi, đánh giá của bản thân. Đối với bên các công ty vận tải, chức năng chính là Quản lý chuyến đi, chức năng cho phép nhà xe tạo/sửa/hoãn các chuyến xe, ghép các chuyến xe ít người để có thể tiết kiệm chi phí.

2.2.2 Biểu đồ use case phân rã

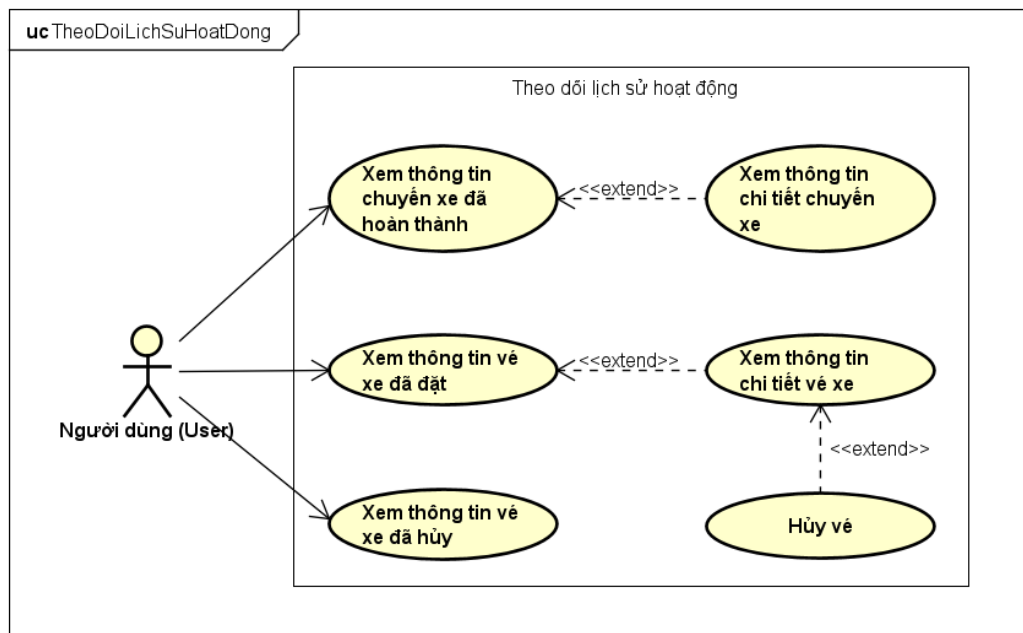
2.2.2.1 Biểu đồ use case phân rã Quản lý thông tin cá nhân



Hình 2 Biểu đồ usecase phân rã Quản lý thông tin cá nhân

Hình 2 mô tả chức năng quản lý thông tin cá nhân, với 1 tác nhân là Người dùng(User), người dùng có thể xem thông tin, chỉnh sửa/cập nhật thông tin bản thân, thay đổi ảnh đại diện của tài khoản.

2.2.2.2 Biểu đồ usecase phân rã Theo dõi lịch sử hoạt động



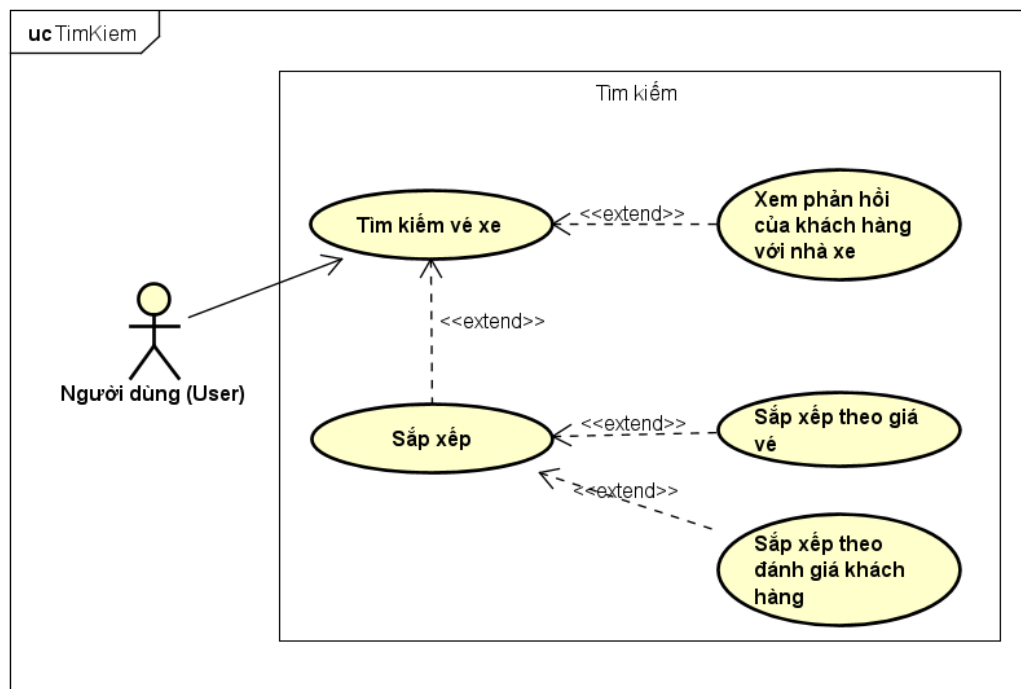
Hình 3 Biểu đồ usecase phân rã Theo dõi lịch sử hoạt động

Hình 3 mô tả chức năng theo dõi lịch sử hoạt động của người dùng.

Người dùng xem thông tin các vé xe mình vừa đặt và chi tiết từng vé đó, nếu vẫn còn trong thời gian hiệu lực người dùng có thể hủy vé đó theo chính sách của hệ thống.

Người dùng xem thông tin các chuyến xe mình đã đi khi đặt vé qua hệ thống, từ đó có thể đánh giá về chuyến đi nếu chưa đánh giá. Ngoài ra, người dùng cũng có thể xem thông tin các vé xe mình đã hủy.

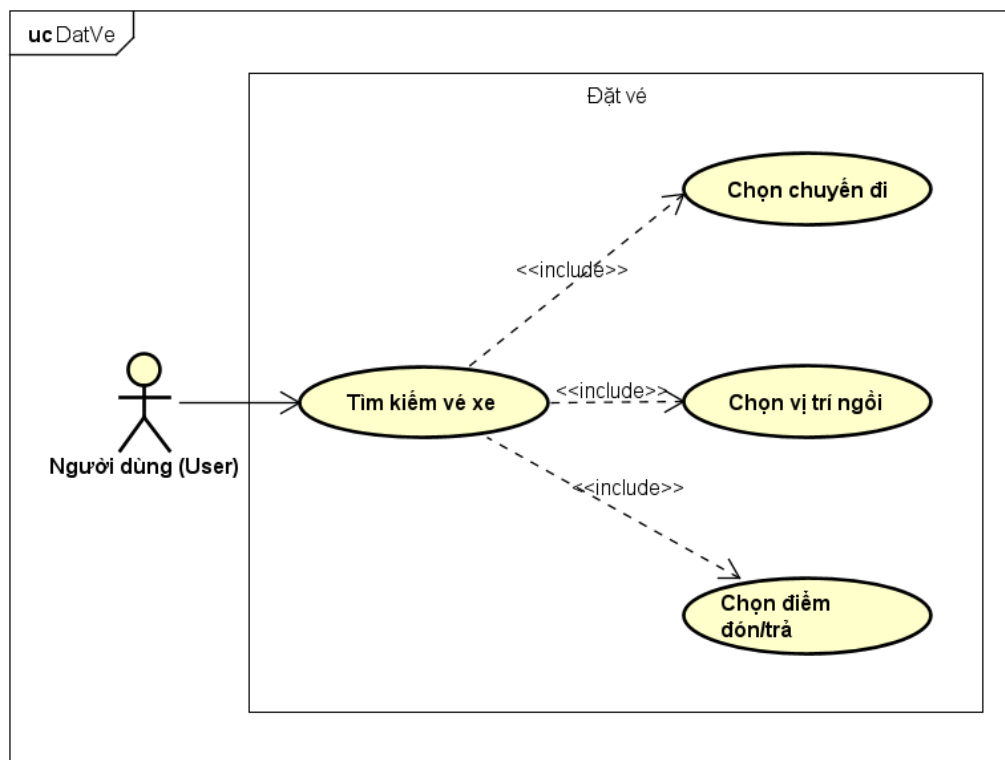
2.2.2.3 Biểu đồ usecase phân rã Tìm kiếm



Hình 4 Biểu đồ usecase phân rã Tìm kiếm

Hình 4 mô tả chức năng tìm kiếm vé xe với 2 tác nhân là người dùng và khách, chức năng này cho phép tìm kiếm vé xe theo một vài thông số nhất định: nơi đi/đến, thời gian. Từ những kết quả đó, người dùng có thể xem phản hồi của những người dùng khác về dịch vụ của từng nhà xe. Ngoài ra, người dùng có thể lọc kết quả theo giá vé (tăng dần/giảm dần) hoặc lọc theo số điểm khách hàng đánh giá.

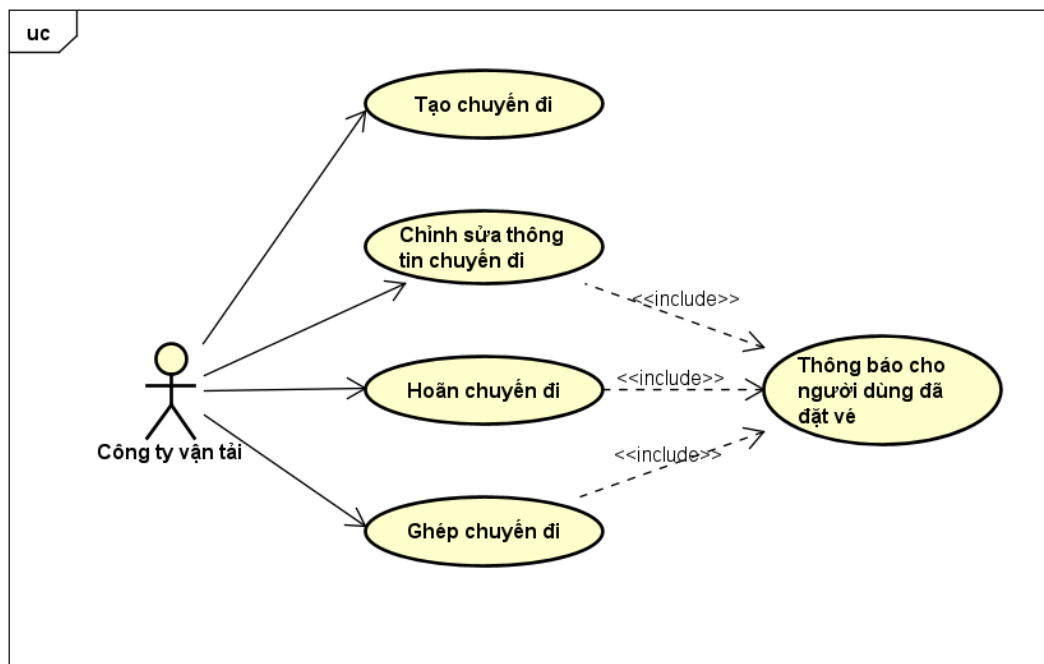
2.2.2.4 Biểu đồ usecase phân rã Đặt vé



Hình 5 Biểu đồ usecase phân rã Đặt vé

Hình 5 mô tả chức năng đặt vé của người dùng. Khách sau khi tìm kiếm nếu muốn đặt vé thì phải đăng nhập mới thực hiện được chức năng này. Chức năng đặt vé bao gồm: Chọn chuyến đi từ những kết quả tìm kiếm được, chọn vị trí ngồi trên xe, chọn điểm đón/trả (điểm đón/trả sẽ là vị trí chi tiết hơn so với nơi đến/đi, có thể là quận/huyện, xã hay một địa điểm nổi tiếng ở địa phương).

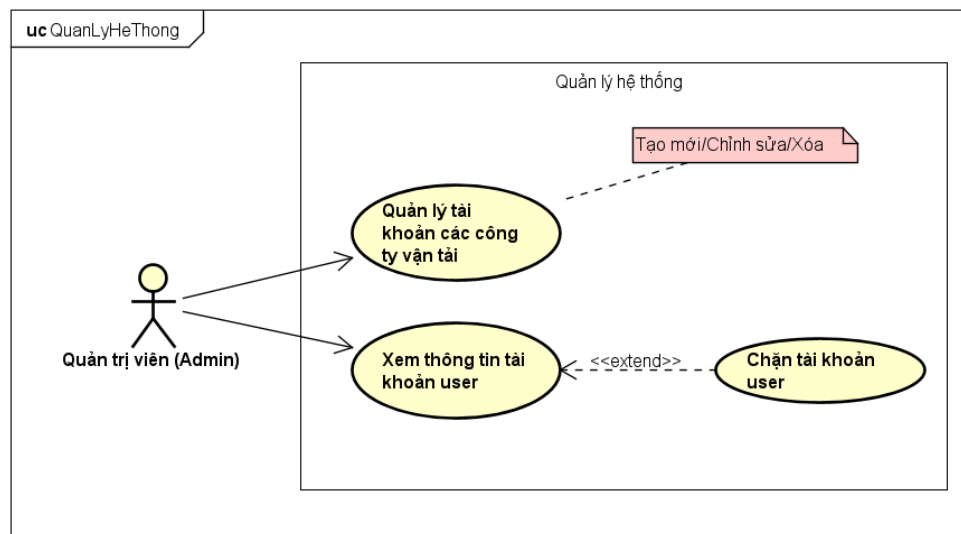
2.2.2.5 Biểu đồ usecase phân rã Quản lý chuyến đi



Hình 6 Biểu đồ usecase phân rã Quản lý chuyến đi

Hình 6 mô tả chức năng quản lý chuyến đi của Công ty vận tải, chức năng này bao gồm tạo chuyến đi mới – các nhà xe có thể để hệ thống tạo tự động đối với những chuyến xe có tần suất lặp lại đều đặn, chỉnh sửa thông tin chuyến đi (đối với thông tin quan trọng như thời gian nếu thay đổi phải đưa ra thông báo cho người dùng đã đặt vé chuyến đi đó), hoãn chuyến đi (vì những lý do đặc biệt như có Thông tư của thành phố cấm các dịch vụ vận tải, khi hoãn phải thông báo cho người dùng đã đặt vé), với những chuyến đi trong ngày nếu có ít người đặt vé thì có thể ghép các chuyến với nhau để tiết kiệm chi phí.

2.2.2.6 Biểu đồ usecase phân rã Quản lý hệ thống



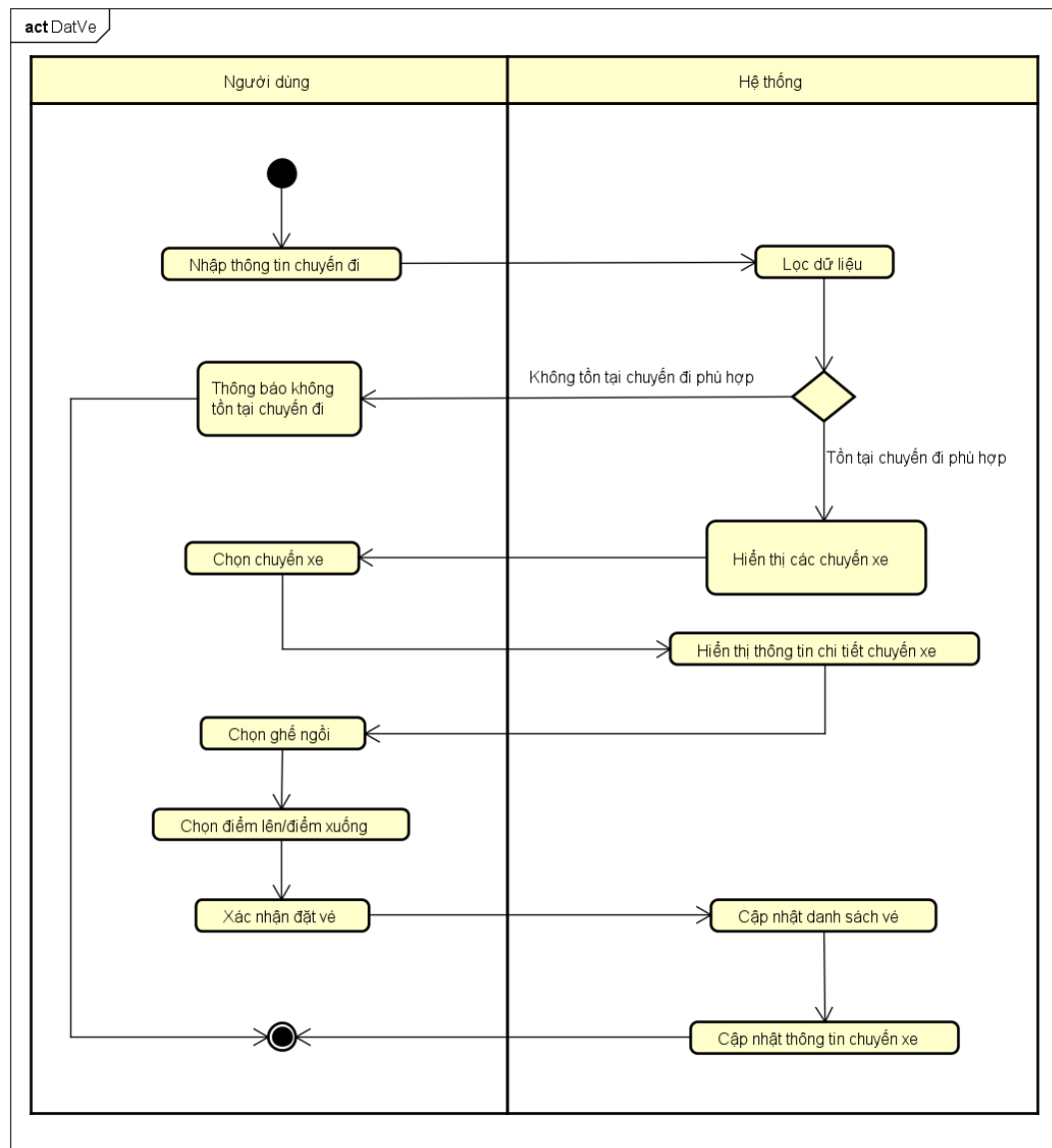
Hình 7 Biểu đồ usecase phân rã Quản lý người dùng

Hình 7 mô tả chức năng quản lý của Quản trị viên hệ thống, Quản trị viên có thể tạo mới, chỉnh sửa, xóa tài khoản cho các công ty vận tải hợp tác với hệ thống, xem thông tin tài khoản của người dùng trong hệ thống, đối với những người dùng bỏ chuyển không thanh toán quá số lần quy định sẽ bị chặn khỏi hệ thống.

2.2.3 Quy trình nghiệp vụ

Đồ án tốt nghiệp này có khá nhiều nghiệp vụ cho cả ba bên: Admin, công ty vận tải, người dùng để có thể hoàn thiện hệ thống. Tuy nhiên ở đây lược bớt những nghiệp vụ ngắn và tập trung trình bày hai nghiệp vụ chính, lớn nhất của hệ thống: Quy trình Đặt vé của người dùng và quy trình Quản lý chuyến đi của công ty vận tải.

2.2.3.1 Quy trình nghiệp vụ Đặt vé

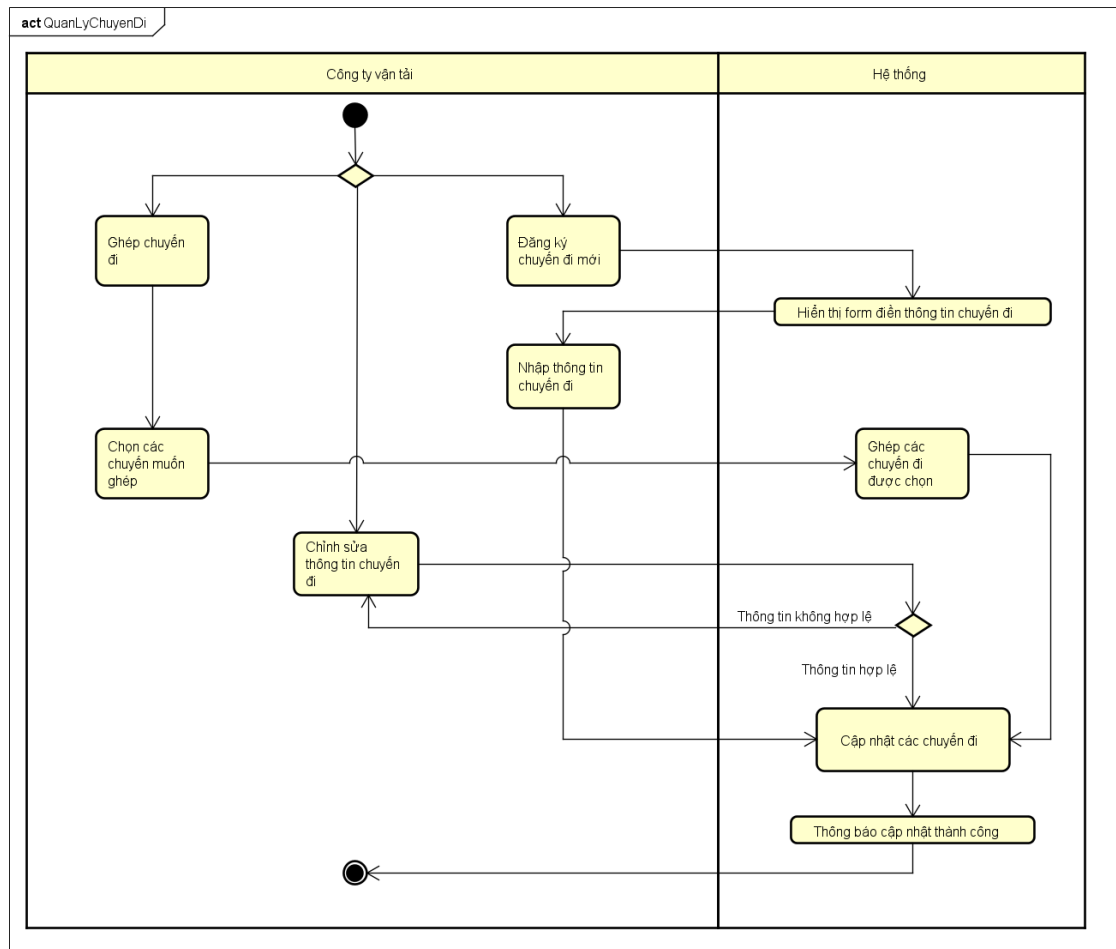


Hình 8 Biểu đồ quy trình nghiệp vụ Đặt vé

Hình 8 mô tả quy trình nghiệp vụ Đặt vé, đây là một nghiệp vụ quan trọng của hệ thống bên phía user. Để đặt vé, khách hàng cần đăng nhập tài khoản cá nhân, tìm kiếm chuyến đi phù hợp với nhu cầu của bản thân thông qua các thông tin: Nơi đến/đi (tỉnh/thành), thời gian đi. Dựa vào thông tin đó, hệ thống sẽ đưa ra các chuyến đi trùng thông tin người dùng tìm kiếm, tại đây người dùng có thể lọc các kết quả tìm được theo giá vé, điểm đánh giá. Sau khi chọn được chuyến đi, người dùng chọn các thông tin chi tiết hơn: vị trí ngồi, điểm lên/xuống (các điểm trung chuyển của nhà xe nếu có, quận/huyện/xã hay ở một địa điểm nổi tiếng tại địa phương), hệ thống cho phép người dùng sử dụng phương thức thanh toán bằng tiền mặt, khách hàng sẽ thanh toán khi lên xe hay tại quầy vé bến xe. Trong tương lai, sẽ phát triển để người dùng có thể thanh toán qua các ví điện tử.

Những khách hàng đặt vé nhưng bỏ chuyến sẽ có cảnh báo từ hệ thống tới tài khoản người dùng, nếu bỏ chuyến 3 lần hệ thống sẽ khóa tài khoản.

2.2.3.2 Quy trình nghiệp vụ Quản lý chuyến đi



Hình 9 Biểu đồ quy trình nghiệp vụ Quản lý chuyến đi

Hình 9 mô tả quy trình nghiệp vụ Quản lý chuyến đi, đây là quy trình quan trọng của hệ thống bên phía công ty vận tải.

Với các chuyến đi mới, nhân viên nhà xe tạo theo form thông tin hệ thống cho sẵn hoặc có thể đặt lịch tạo tự động với những chuyến lặp lại định kỳ.

Với các chuyến đi đã tạo và cần chỉnh sửa thông tin, nhân viên có thể chỉnh sửa một số thông tin nhất định: có thể chỉnh sửa giờ xuất phát nhưng không được đổi ngày xuất phát nếu đã có khách hàng đặt vé ở chuyến đi đó, đổi xe,...

Với các chuyến đi có quá ít khách đặt, nhà xe có thể cân nhắc để ghép các chuyến đi của mình. Tuy nhiên, với việc ghép chuyến đi này, nhà xe sẽ phải có thông báo sớm với khách đặt vé để khách có thể cân nhắc. Chức năng này sẽ được mô tả chi tiết hơn ở mục 2.4.

2.3 Đặc tả chức năng

2.3.1 Đặc tả use case Đăng nhập bằng số điện thoại

Mã Use case	UC001	Tên Use case	Đăng nhập bằng số điện thoại
Tác nhân	Khách		
Tiền điều kiện	Không		
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Khách	Yêu cầu form đăng nhập
	2.	Hệ thống	Hiển thị màn hình đăng nhập
	3.	Khách	Nhập số điện thoại và xác nhận
	4.	Hệ thống	Gửi mã OTP tới số điện thoại khách vừa nhập
	5.	Hệ thống	Hiển thị màn hình nhập mã OTP
	6.	Khách	Nhập mã OTP và xác nhận
	7.	Hệ thống	Kiểm tra mã OTP khách vừa nhập
	8.	Hệ thống	Thông báo đăng nhập thành công và hiển thị màn hình chính
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	3.a.	Hệ thống	Thông báo nhập sai số điện thoại
	3.b.	Khách	Thực hiện luồng sự kiện 3
	7.a.	Hệ thống	Thông báo mã OTP chưa đúng
Hậu điều kiện	Không		

* Dữ liệu đầu vào của form đăng nhập gồm các trường dữ liệu sau:

Bảng 1. Dữ liệu đầu vào use case đăng nhập bằng số điện thoại

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Số điện thoại	Nội dung dạng văn bản	Có	Gồm 9 ký tự nếu ký tự đầu tiên không phải “0” Gồm 10 ký tự nếu ký tự đầu tiên là “0”	37954456 037954456

2.3.2 Đặc tả use case Đặt vé

Mã Use case	UC002	Tên Use case	Đặt vé
Tác nhân	Người dùng		
Tiền điều kiện			
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Người dùng	Nhập thông tin chuyến đi và tìm kiếm
	2.	Hệ thống	Hiển thị kết quả tìm kiếm
	3.	Người dùng	Chọn chuyến đi
	4.	Hệ thống	Hiển thị màn hình chọn vị trí ngồi
	5.	Người dùng	Chọn vị trí ngồi và xác nhận
	6.	Hệ thống	Hiển thị màn hình chọn vị trí đón/trả
	7.	Người dùng	Chọn vị trí đón/trả
	8.	Hệ thống	Hiển thị màn hình chọn phương thức thanh toán
	9.	Người dùng	Chọn phương thức thanh toán bằng tiền mặt

	10.	Người dùng	Xác nhận đặt vé
	11.	Hệ thống	Thông báo đặt vé thành công
	12.	Hệ thống	Lưu lại thông tin vé và cập nhật thông tin chuyến đi
Luồng sự kiện thay thế			
Hậu điều kiện	Không		

* Dữ liệu đầu vào các thông tin khi đặt vé gồm các trường dữ liệu sau:

Bảng 2. Dữ liệu đầu vào use case Đặt vé

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Vị trí ngồi	Tập hợp các toggle button, mỗi button thể hiện cho một vị trí trên xe	Có	Có ít nhất 1 button được chọn	
2.	Vị trí đón/trả	Các radio button, 1 ô điền nội dung văn bản với mỗi lựa chọn đón và trả	Có	Với mỗi điểm đón/trả đều tích 1 lựa chọn	
3.	Chọn phương thức thanh toán	1 lựa chọn: Tiền mặt	Có		

2.3.3 Đặc tả use case Thực hiện chuyến đi

Mã Use case	UC003	Tên Use case	Thực hiện chuyến đi																								
Tác nhân	Người dùng																										
Tiền điều kiện	Người dùng đã đặt vé thành công																										
Luồng sự kiện chính (Luồng sự kiện thành công)	<table><tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr><tr><td>1.</td><td>Hệ thống</td><td>Thông báo trước khi đến giờ xuất phát 1 tiếng</td></tr><tr><td>2.</td><td>Hệ thống</td><td>Thông báo đến giờ khởi hành</td></tr><tr><td>3.</td><td>Hệ thống</td><td>Hiển thị màn hình kết thúc chuyến đi khi đến giờ xuống xe</td></tr><tr><td>4.</td><td>Người dùng</td><td>Xác nhận đã hoàn thành chuyến đi</td></tr><tr><td>5.</td><td>Hệ thống</td><td>Hiển thị màn hình đánh giá chuyến đi</td></tr><tr><td>6.</td><td>Người dùng</td><td>Đánh giá, phản hồi của bản thân về chuyến đi, nhà xe, nhân viên,...</td></tr><tr><td>7.</td><td>Hệ thống</td><td>Cập nhật chuyến đi, phản hồi</td></tr></table>			STT	Thực hiện bởi	Hành động	1.	Hệ thống	Thông báo trước khi đến giờ xuất phát 1 tiếng	2.	Hệ thống	Thông báo đến giờ khởi hành	3.	Hệ thống	Hiển thị màn hình kết thúc chuyến đi khi đến giờ xuống xe	4.	Người dùng	Xác nhận đã hoàn thành chuyến đi	5.	Hệ thống	Hiển thị màn hình đánh giá chuyến đi	6.	Người dùng	Đánh giá, phản hồi của bản thân về chuyến đi, nhà xe, nhân viên,...	7.	Hệ thống	Cập nhật chuyến đi, phản hồi
STT	Thực hiện bởi	Hành động																									
1.	Hệ thống	Thông báo trước khi đến giờ xuất phát 1 tiếng																									
2.	Hệ thống	Thông báo đến giờ khởi hành																									
3.	Hệ thống	Hiển thị màn hình kết thúc chuyến đi khi đến giờ xuống xe																									
4.	Người dùng	Xác nhận đã hoàn thành chuyến đi																									
5.	Hệ thống	Hiển thị màn hình đánh giá chuyến đi																									
6.	Người dùng	Đánh giá, phản hồi của bản thân về chuyến đi, nhà xe, nhân viên,...																									
7.	Hệ thống	Cập nhật chuyến đi, phản hồi																									
Luồng sự kiện thay thế	<table><tr><th>STT</th><th>Thực hiện bởi</th><th>Hành động</th></tr><tr><td>6.a.</td><td>Người dùng</td><td>Bỏ qua bước đánh giá và sẽ đánh giá sau</td></tr><tr><td>6.b.</td><td>Hệ thống</td><td>Cập nhật chuyến đi</td></tr></table>			STT	Thực hiện bởi	Hành động	6.a.	Người dùng	Bỏ qua bước đánh giá và sẽ đánh giá sau	6.b.	Hệ thống	Cập nhật chuyến đi															
STT	Thực hiện bởi	Hành động																									
6.a.	Người dùng	Bỏ qua bước đánh giá và sẽ đánh giá sau																									
6.b.	Hệ thống	Cập nhật chuyến đi																									
Hậu điều kiện	Không																										

2.3.4 Đặc tả use case Tạo chuyến đi

Mã Use case	UC004	Tên Use case	Tạo chuyến đi
Tác nhân	Công ty vận tải		

Tiền điều kiện			
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Công ty vận tải	Chọn chức năng Quản lý chuyến đi
	2.	Hệ thống	Hiển thị màn hình Quản lý chuyến đi
	3.	Công ty vận tải	Chọn Thêm mới chuyến đi
	4.	Hệ thống	Hiển thị màn hình Thêm mới chuyến đi
	5.	Công ty vận tải	Điền thông tin chuyến đi mới
	6.	Hệ thống	Cập nhật chuyến đi
Luồng sự kiện thay thế			
Hậu điều kiện	Không		

* Dữ liệu đầu vào use case Tạo chuyến đi gồm các trường dữ liệu sau:

Bảng 3. Dữ liệu đầu vào chức năng Thêm mới chuyến đi

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Xe	Dropdown các xe mà công ty vận tải sở hữu	Có		
2.	Nơi đi	Dropdown các tỉnh thành trong nước – là nơi xuất phát của chuyến đi	Có		

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
3.	Nơi đến	Dropdown các tỉnh thành trong nước – là nơi kết thúc của chuyến đi	Có		
4.	Ngày giờ xuất phát	DateTime Picker	Có		

2.3.5 Đặc tả use case Đặt lịch chuyến đi

Mã Use case	UC005	Tên Use case	Đặt lịch chuyến đi
Tác nhân	Công ty vận tải		
Tiền điều kiện			
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Công ty vận tải	Chọn chức năng Quản lý chuyến đi
	2.	Hệ thống	Hiển thị màn hình Quản lý chuyến đi
	3.	Công ty vận tải	Chọn Đặt lịch chuyến đi
	4.	Hệ thống	Hiển thị màn hình Đặt lịch chuyến đi
	5.	Công ty vận tải	Điền thông tin chuyến đi
	6.	Hệ thống	Cập nhật các chuyến đi
Luồng sự kiện thay thế			

Hậu điều kiện	Không
----------------------	-------

Bảng 4. Dữ liệu đầu vào chức năng Đặt lịch chuyến đi

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Xe	Dropdown button gồm các xe mà công ty vận tải sở hữu	Có		
2.	Nơi đi	Dropdown các tỉnh thành trong nước – là nơi xuất phát của chuyến đi	Có		
3.	Nơi đến	Dropdown các tỉnh thành trong nước – là nơi kết thúc của chuyến đi	Có		
4.	Giờ xuất phát	DatePicker	Có		
5.	Tần suất	Input số tự nhiên lớn hơn 0 – là số ngày sẽ lặp lại chuyến đi ấy.	Có		

2.3.6 Đặc tả use case Đánh giá chuyến đi

Mã Use case	UC006	Tên Use case	Đánh giá chuyến đi
--------------------	-------	---------------------	--------------------

Tác nhân	Người dùng		
Tiền điều kiện	Người dùng đã hoàn thành chuyến đi		
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Hệ thống	Hiển thị màn hình kết thúc chuyến đi
	2.	Người dùng	Chọn Viết đánh giá
	3.	Người dùng	Nhập các thông tin đánh giá của bản thân về chuyến đi
	4.	Hệ thống	Cập nhật đánh giá
Luồng sự kiện thay thế			
Hậu điều kiện	Không		

Bảng 5. Dữ liệu đầu vào chức năng Đánh giá chuyến đi

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Điểm đánh giá	Các radio button giá trị từ 1 đến 5	Có		
2.	Nội dung	Textfield cho phép người dùng nhập văn bản	Có		

2.3.7 Đặc tả use case Hủy vé

Mã Use case	UC007	Tên Use case	Hủy vé
--------------------	-------	---------------------	--------

Tác nhân	Người dùng		
Tiền điều kiện	Người dùng đã đặt vé thành công		
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Người dùng	Chọn Quản lý vé
	2.	Người dùng	Chọn Vé đang đặt
	3.	Người dùng	Nhấn vào vé và chọn Hủy vé
	4.	Hệ thống	Cập nhật thông tin vé
Luồng sự kiện thay thế			
Hậu điều kiện	Không		

2.3.8 Đặc tả use case Thêm mới xe

Mã Use case	UC008	Tên Use case	Thêm mới xe
Tác nhân	Công ty vận tải		
Tiền điều kiện			
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Công ty vận tải	Chọn Quản lý xe
	2.	Công ty vận tải	Chọn Thêm mới
	3.	Công ty vận tải	Nhập các thông tin của xe mới
	4.	Hệ thống	Kiểm tra trùng lặp thông tin
	5.	Hệ thống	Cập nhật thông tin xe

Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	5.a.	Hệ thống	Thông báo thông tin chưa đúng/bị trùng. Yêu cầu nhập lại.
Hậu điều kiện	Không		

Bảng 6. Dữ liệu đầu vào chức năng Đánh giá chuyến đi

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Loại xe	Dropdown các loại xe	Có		
2.	Biển số xe	Biển số xe	Có		
3.	Số điện thoại	Số điện thoại liên lạc của xe – thường sẽ là của tài xế hoặc lơ xe	Có		

2.3.9 Đặc tả use case Phản hồi khách hàng

Mã Use case	UC009	Tên Use case	Phản hồi khách hàng
Tác nhân	Công ty vận tải		
Tiền điều kiện			
Luồng sự kiện chính (Luồng sự kiện)	STT	Thực hiện bởi	Hành động
	1.	Công ty vận tải	Chọn Khách hàng đánh giá

thành công)	2.	Công ty vận tải	Viết phản hồi
	3.	Công ty vận tải	Gửi
	4.	Hệ thống	Cập nhật thông tin
Luồng sự kiện thay thế			
Hậu điều kiện	Không		

Bảng 7. Dữ liệu đầu vào chức năng Phản hồi khách hàng

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Nội dung	Nội dung phản hồi của công ty vận tải với nhà xe	Không		

2.3.10 Đặc tả use case Thêm mới công ty vận tải

Mã Use case	UC010	Tên Use case	Thêm mới công ty vận tải
Tác nhân	Admin		
Tiền điều kiện			
Luồng sự kiện chính (Luồng sự kiện thành công)	STT	Thực hiện bởi	Hành động
	1.	Admin	Chọn Quản lý công ty vận tải
	2.	Admin	Chọn Thêm mới
	3.	Hệ thống	Hiển thị form thông tin

	4.	Admin	Nhập thông tin công ty vận tải
	5.	Hệ thống	Kiểm tra thông tin hợp lệ
	6.	Hệ thống	Cập nhật thông tin xe
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	5.a.	Hệ thống	Thông báo thông tin chưa hợp lệ/bị trùng. Yêu cầu nhập lại.
Hậu điều kiện	Không		

Bảng 8. Dữ liệu đầu vào chức năng Thêm mới công ty vận tải

STT	Trường dữ liệu	Mô tả	Bắt buộc	Điều kiện hợp lệ	Ví dụ
1.	Tên công ty vận tải	Input để nhập tên công ty vận tải	Có		
2.	Số điện thoại	Số điện thoại/hotline của công ty vận tải	Có		
3.	Slogan/Mô tả	Câu slogan hoặc mô tả của công ty về bản thân	Có		
4.	Logo	File Upload – Chọn ảnh từ máy cá nhân	Có		

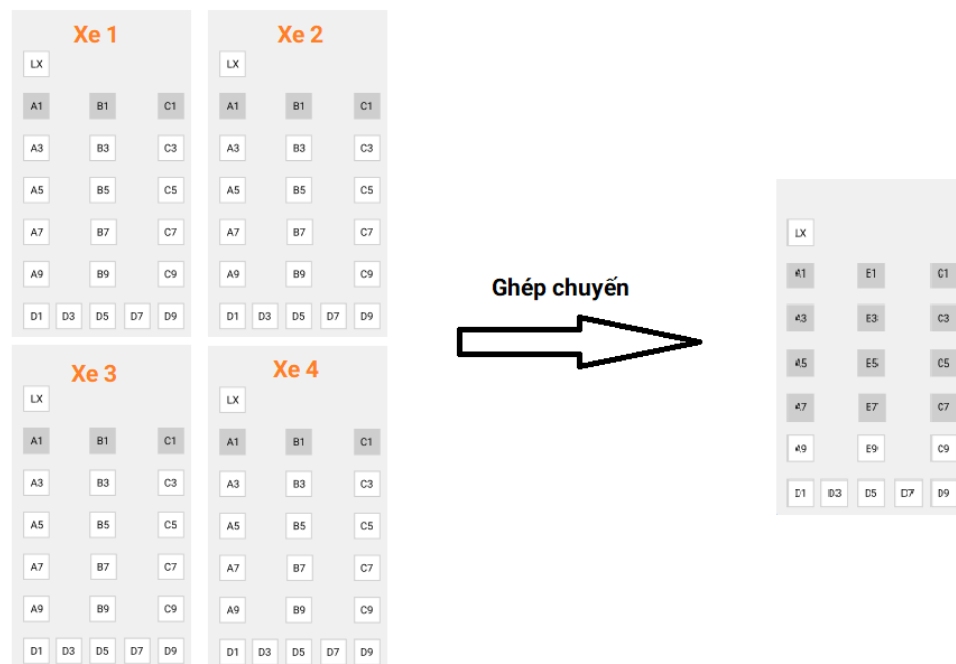
2.4 Giải pháp ghép chuyến đi cho các công ty vận tải

2.4.1 Đặt vấn đề

Với những hệ thống đặt vé xe hiện nay, mỗi nhà xe sẽ có khá nhiều chuyến đi trong 1 ngày, vì mô hình kinh doanh bây giờ của đa số các công ty vận tải khá lớn, không gói gọn lại trong một vài chiếc xe mà có rất nhiều xe thuộc sở hữu của công ty, nên việc sắp xếp nhiều chuyến trong một ngày cũng xem như là một giải pháp để nhà xe có thể chở được nhiều khách hơn, khách hàng có nhiều lựa chọn khung giờ đi trong ngày hơn, qua đó doanh thu của nhà xe cũng tốt hơn. Nhưng cũng từ đó, nảy sinh ra một vấn đề ảnh hưởng khá lớn đến lợi nhuận của các công ty vận tải, mặc dù có thể doanh thu cao hơn nhưng việc có nhiều chuyến đi trong ngày cũng làm tăng chi phí vận hành hệ thống tăng lên nhiều lần. Chính vì thế, với những ngày có lượng khách ít, rất có thể có nhiều chuyến có khách đặt nhưng mỗi chuyến xe chỉ có 2-3 người đặt, dẫn đến việc nhà xe bắt buộc phải hủy chuyến hoặc chấp nhận chạy xe chỉ với số ít khách. Cả hai phương án này đều khiến cho phía công ty vận tải chịu lỗ khá nặng cho mỗi ngày như vậy.

2.4.2 Giải pháp đưa ra

Để giải quyết vấn đề vừa nêu trên, hệ thống của em đã phát triển thêm một tính năng “Ghép chuyến đi”. Thay vì hủy chuyến hay chấp nhận chạy chuyến ít người, các nhà xe có thể ghép những chuyến đi ít người đẩy lại với nhau, tạo thành một chuyến có đông người hơn, qua đó cũng tiết kiệm chi phí cũng như đảm bảo lợi nhuận hơn rất nhiều.



Hình 10 Hình minh họa chức năng “Ghép chuyến đi”

Để đảm bảo sự trải nghiệm của khách hàng, những chuyến xe được ghép cùng nhau phải thỏa mãn: các chuyến xe cùng ngày, cùng một loại xe (ghế ngồi, giường nằm, limousine,...), cùng nhà xe. ⁽¹⁾ Nhân viên nhà xe sẽ tự chủ động chọn chức năng ghép chuyến đi khi cần thiết (nên trước khoảng 10-12 tiếng so với thời gian xuất phát chuyến xe sớm nhất). Sau khi các chuyến xe được ghép, hệ thống sẽ gửi thông báo tới người dùng để người dùng nắm được thông tin.

Dưới đây là cách mà hệ thống sẽ tính toán và thực hiện ghép chuyến ngay sau khi có yêu cầu từ phía nhà xe.

Giả sử nhà xe A có n chuyến xe chưa xuất phát trong ngày hôm nay, dĩ nhiên cả n chuyến xe này đều thỏa mãn các tính chất (1) vừa nêu ở trên. Xe thứ i ($i = 1, 2, 3, \dots, n$), sẽ có a_i chỗ ngồi/giường và b_i vé đã được đặt.

Ta có thể tính lợi nhuận của nhà xe bằng công thức:

$$\text{Lợi nhuận} = \sum_{i=1}^n (b_i \times \text{giá vé}) - \sum_{i=1}^n (\text{chi phí vận hành})$$

Vì cùng một loại xe, cùng loại dịch vụ nên giá vé của mỗi xe đều giống nhau.

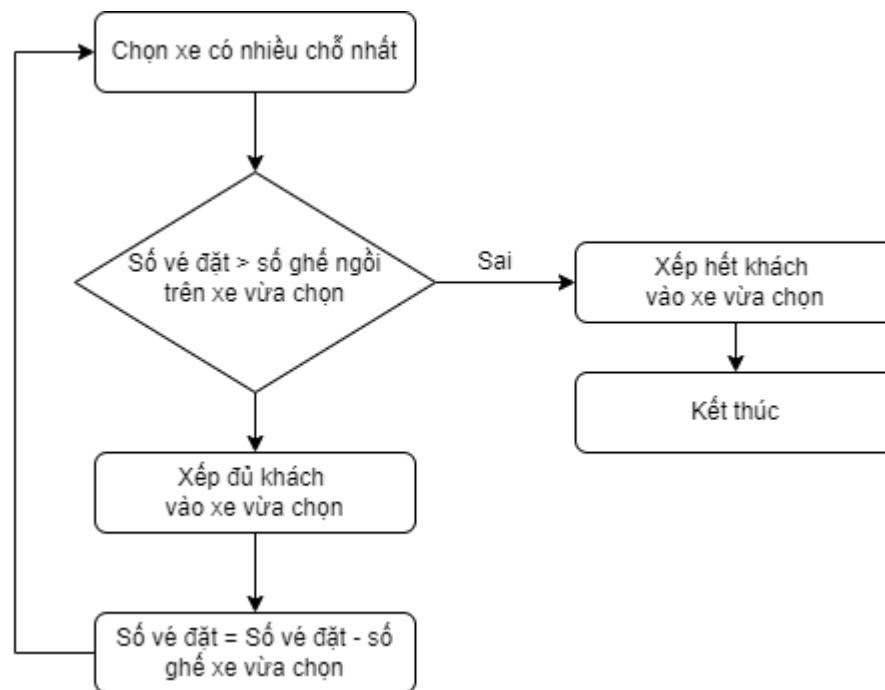
Chi phí vận hành bao gồm tiền xăng, lương tài xế, nhân viên, chi phí phát sinh,... Đây là những khoản nó sẽ thay đổi với từng xe, từng chuyến đi khác nhau và rất khó để có thể ước lượng chính xác được thông số này nên để đơn giản hơn chúng ta sẽ ước lượng chi phí vận hành trung bình mỗi xe (điều này sẽ dễ hơn rất nhiều cho các công ty vận tải).

$$\text{Lợi nhuận} = \sum_{i=1}^n (b_i \times \text{giá vé}) - n \times \text{chi phí vận hành trung bình mỗi xe}$$

Vì với số lượng khách đã đăng ký vé (b_i) là cố định nên để có lợi nhuận tốt, ta cần bớt đi lượng $n \times \text{chi phí vận hành trung bình mỗi xe}$ hay một cách tường minh hơn, chúng ta cần giảm số xe cần chạy xuống mức tối thiểu.

Tuy nhiên trong các xe có nhiều chỗ thường sẽ có dãy ghế sau cùng là dãy ghế ít được người dùng lựa chọn vì lý do đó là những ghế sát nhau (thường là 3-5 ghế) không có khoảng cách giữa các ghế nên việc đó khiến cho khách hàng rất không thoải mái, đặc biệt là trong mùa dịch này. Ngoài ra, người ta cũng hạn chế chọn ghế sau cùng vì nó gần động cơ ở đuôi xe cũng như ngồi càng cuối xe sẽ dễ bị say xe hơn. Cũng có trường hợp khách hàng sẽ chọn hàng ghế cuối nếu họ đi cùng nhóm bạn bè và muốn ngồi gần nhau trên cả chuyến đi. Do đó, hệ thống sẽ chia các xe thành 2 khu vực ghế là trước và sau. Những khách hàng đặt vé ở

khu trước, khi ghép sẽ luôn được vé ở khu trước và những khách đặt vé ở khu sau cũng sẽ luôn được vé ở khu sau.



Hình 11 Sơ đồ thuật toán ghép xe

Ta sẽ lần lượt xếp khách đã đặt vé vào các xe có số chỗ nhiều nhất, vào từng khu vực phù hợp từng vé cho đến khi hết khách.

2.5 Yêu cầu phi chức năng

2.5.1 Yêu cầu về bảo mật

Để đảm bảo tính bảo mật, chính chủ cũng như thuận tiện cho phía người dùng, hệ thống chỉ cho phép người dùng đăng nhập bằng số điện thoại thông qua xác thực OTP.

Về phía tài khoản của admin và các công ty vận tải, mật khẩu sẽ được mã hóa bằng MD5 và lưu vào database chuỗi đã được mã hóa đó.

2.5.2 Yêu cầu giao diện

Ứng dụng, website cần có giao diện thân thiện, dễ nhìn, dễ sử dụng đối với người dùng. App mobile cần phù hợp với các điện thoại sử dụng hệ điều hành Android hiện nay. Sử dụng các hình ảnh, icon liên quan đến chức năng của hệ thống để tạo sự đơn giản, dễ liên tưởng.

2.5.3 Yêu cầu khác

Hệ thống cần được xây dựng để dễ dàng sửa đổi, mở rộng, linh động theo nhu cầu thực tế của người dùng, của các công ty vận tải.

Chương 3 Công nghệ sử dụng

3.1 Server

3.1.1 NodeJS

NodeJS là một nền tảng (platform) được xây dựng dựa trên V8 Javascript Engine – trình thông dịch thực thi mã Javascript. Đây là công cụ mạnh mẽ để xây dựng server hay các ứng dụng web một cách nhanh chóng, hiệu quả. NodeJS được viết bằng C++ do đó nó cho tốc độ xử lý và hiệu năng khá cao.

NodeJS có thể chạy được trên nhiều hệ điều hành: Linux, Windows, Mac OS.

NodeJS có một cộng đồng lập trình viên sử dụng lớn, đó là một lợi thế, khi sử dụng NodeJS có thể tham khảo hoặc được giải đáp những thắc mắc, những lỗi gặp phải trong quá trình sử dụng.

3.1.2 MySQL

MySQL là một trong những hệ thống quản lý cơ sở dữ liệu(CSDL) mã nguồn mở quan hệ SQL sử dụng trên web phổ biến nhất hiện nay.

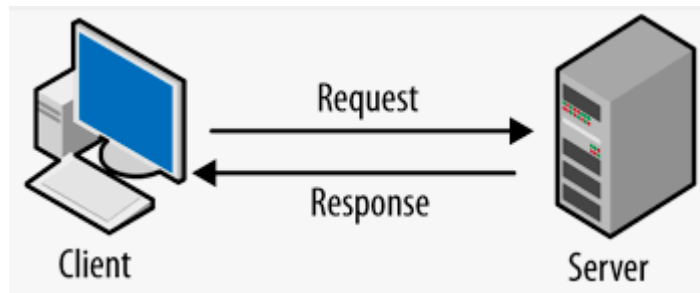
MySQL được sử dụng rộng rãi bởi khả năng dễ sử dụng, hoạt động rất nhanh và tốt với các tập dữ liệu lớn và được sử dụng với các trang web lớn như Google, Twitter, Facebook,...

Tính năng nổi bật của MySQL:

- MySQL được phát hành theo giấy phép mã nguồn mở.
- MySQL lý tưởng cho tất cả ứng dụng lớn và nhỏ.
- MySQL có tốc độ xử lý, truy vấn nhanh, an toàn, dễ mở rộng và sử dụng.
- MySQL hoạt động trên đa nền tảng với hầu hết các ngôn ngữ lập trình (PHP, JavaScript, Java, C, Kotlin,...)
- MySQL hỗ trợ cơ sở dữ liệu lớn, lên tới 50 triệu hàng trong một bảng, giới hạn kích thước mặc định một bảng là 4GB và có thể tăng giới hạn đến 8 triệu TB.
- MySQL có tiêu chuẩn bảo mật mã hóa thông tin cao, đảm bảo an toàn dữ liệu

Các tính năng này khiến MySQL phù hợp cho sản phẩm ĐATN này.

MySQL hoạt động dựa trên mô hình Client-Server được mô tả ở hình dưới:



Hình 12 Mô hình Client-Server

Máy client sẽ liên lạc với server trong mạng nhất định, MySQL tạo ra bảng để lưu trữ, định nghĩa dữ liệu. Client sẽ gửi các yêu cầu đến server và server sẽ phản hồi lại kết quả sau khi truy vấn dữ liệu từ database.

3.2 Website

3.2.1 VueJS

Vue.js là một framework Javascript được tạo bởi Evan You – một kỹ sư làm việc tại Google, có thể coi là sự kết hợp giữa 2 công nghệ cũng đang nổi hiện nay là React và Angular, thích hợp để xây dựng UI cho các single-page application (SPA). Các trang web viết bằng Vue có khả năng lấy dữ liệu từ server mà không cần tải lại trang giúp tăng trải nghiệm người dùng.

Vue có tiện ích CLI riêng biệt, cung cấp khả năng xây dựng hiệu quả hoặc thử nghiệm ý tưởng mới. Ngoài ra, Vue có thể thực hiện nhiều tác vụ cùng lúc, nó có thể giải quyết tốt các quá trình phức tạp như tạo ứng dụng web hay ổn định các ứng dụng di động. Vue thường phù hợp cho những project thiên về giao diện như logistic, bán hàng,... Nó có tính liên kết dữ liệu hai chiều cũng như có thể linh động điều chỉnh các quy trình cho nên kể cả với những project có giao diện đơn giản nhưng quy trình khá phức tạp, Vue vẫn hoàn toàn có thể vận hành tốt.

Vue.js là một framework được thiết kế từ đầu theo hướng cho phép và khuyến khích việc phát triển ứng dụng theo từng bước. Khi phát triển lớp giao diện, người dùng chỉ cần dùng thư viện lõi của Vue, vốn rất dễ học và tích hợp với các thư viện hoặc dự án có sẵn.

3.3 App mobile

3.3.1 Java

Java là một ngôn ngữ lập trình hiện đại, bậc cao, hướng đối tượng, bảo mật và mạnh mẽ. Java ban đầu được phát triển bởi Sun Microsystems do James Gosling khởi xướng và phát

hành vào năm 1995. Mặc dù càng ngày càng xuất hiện nhiều ngôn ngữ lập trình mới đi kèm những thư viện giúp công việc của lập trình viên bớt nặng nhọc tuy nhiên Java vẫn đang là một ngôn ngữ lập trình được rất đông lập trình viên trên thế giới sử dụng.

Một số điểm nổi bật của Java:

- **Hướng đối tượng:** Trong Java, mọi thứ đều là một Object. Java có thể dễ dàng mở rộng và bảo trì vì nó được xây dựng dựa trên mô hình Object.
- **Bảo mật:** Với tính năng an toàn của Java, nó cho phép phát triển các hệ thống không có virus, giả mạo. Các kỹ thuật xác thực dựa trên mã hoá khóa công khai.
- **Đa luồng:** Với tính năng đa luồng của Java có thể viết các chương trình có thể thực hiện nhiều tác vụ đồng thời. Tính năng thiết kế này cho phép các nhà phát triển xây dựng các ứng dụng tương tác có thể chạy trơn tru hơn.
- **Thông dịch:** Mã byte Java được dịch trực tiếp tới các máy tính gốc và không được lưu trữ ở bất cứ đâu.
- **Hiệu năng cao:** Với việc sử dụng trình biên dịch Just-In-Time, Java cho phép thực hiện hiệu năng cao.
- **Tính an toàn:** Ngôn ngữ lập trình Java yêu cầu chặt chẽ về kiểu dữ liệu, dữ liệu phải được khai báo tường minh, không sử dụng con trỏ và các phép toán với con trỏ, Java kiểm soát chặt chẽ việc truy nhập đến mảng, chuỗi, không cho phép sử dụng các kỹ thuật tràn do đó các truy nhập sẽ không vượt quá kích thước của mảng hoặc chuỗi. Quá trình cấp phát và giải phóng bộ nhớ được thực hiện tự động. Cơ chế xử lý lỗi giúp việc xử lý và phục hồi lỗi dễ dàng hơn.

3.3.2 RESTful API

RESTful API là một tiêu chuẩn trong việc thiết kế API cho các ứng dụng client-server để thuận tiện cho việc quản lý tài nguyên và truyền tải qua phương thức HTTP.

Thành phần chính của RESTful API:

- **API (Application Programming Interface):** là tập hợp các quy tắc để tương tác giữa các thành phần ứng dụng. API trả về dạng dữ liệu JSON hoặc XML.
- **REST (Representational State Transfer):** là dạng chuyển đổi cấu trúc để viết API. Nó tạo tương tác giữa các máy với nhau qua HTTP và quy định việc sử dụng các phương thức HTTP và dạng URL cho các ứng dụng.

Trong đồ án này thì dữ liệu trả về sẽ luôn là dạng JSON.

3.3.3 Firebase

Firebase là một dịch vụ hệ thống backend được Google cung cấp sẵn cho ứng dụng. Dịch vụ này giúp phát triển, triển khai và mở rộng quy mô của ứng dụng Mobile nhanh hơn, dù là Android hay IOS, Firebase vẫn có thể hỗ trợ tốt. Trong một số dự án, Firebase là các API đơn giản, không cần backend hay server. Tuy nhiên, đề án này chỉ sử dụng một số tính năng của firebase như xác thực số điện thoại.

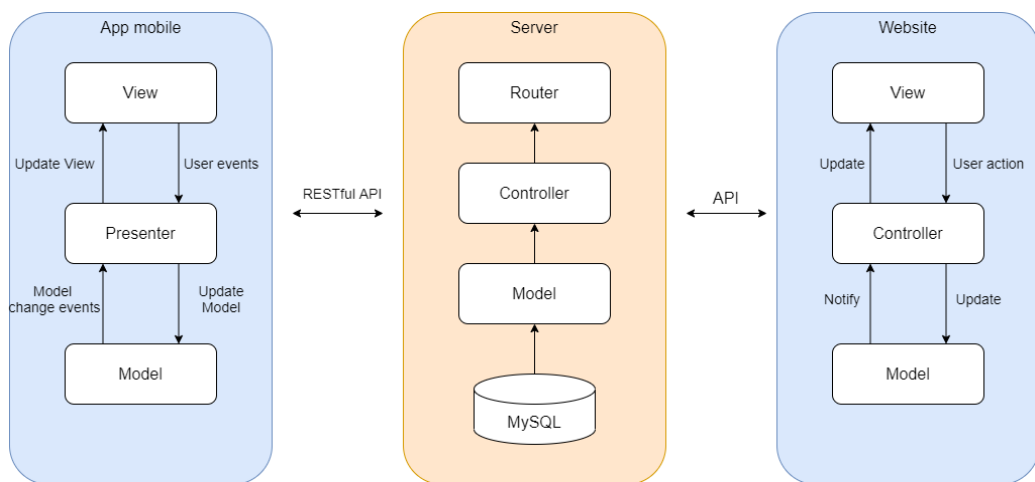
Ưu điểm:

- Triển khai ứng dụng nhanh chóng: Firebase tiết kiệm rất nhiều thời gian quản lý và đồng bộ tất cả dữ liệu cho người dùng. Đó là nhờ người dùng không phải quan tâm đến phần backend cùng các API tốt, hỗ trợ đa nền tảng.
- Sự ổn định: Được viết dựa trên nền tảng cloud cung cấp bởi Google, các công cụ luôn đảm bảo độ ổn định tối đa. Bên cạnh đó, quá trình nâng cấp hay bảo trì Server cũng diễn ra nhanh và đơn giản hơn.

Chương 4 Phát triển và triển khai ứng dụng

4.1 Thiết kế kiến trúc

4.1.1 Lựa chọn kiến trúc phần mềm



Hình 13 Tổng quan kiến trúc phần mềm

Hệ thống được triển khai như mô hình được mô tả ở **Hình 13**. Server sẽ sử dụng (truy xuất, thêm, sửa, xóa) dữ liệu từ database thông qua các model, các chức năng logic được thực thi tại controller và router sẽ định tuyến, cung cấp phương thức giao tiếp cho app cũng như web. Giao diện app mobile (cho khách hàng) và web (cho công ty vận tải) hiển thị dữ liệu được nhận từ server, đồng thời nhận các yêu cầu của người dùng gửi về server để xử lý. App mobile được xây dựng theo mô hình MVP, web được xây dựng theo mô hình MVC.

4.1.1.1 Mô hình MVC

MVC là từ viết tắt bởi 3 từ Model – View – Controller. Đây là mô hình thiết kế sử dụng trong kỹ thuật phần mềm. Mô hình chia source code thành 3 phần, tương ứng mỗi từ. Mỗi từ tương ứng với một hoạt động tách biệt trong một mô hình.

Model là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng, là cầu nối giữa 2 thành phần bên dưới: View và Controller. Model thể hiện dưới hình thức là một cơ sở dữ liệu hoặc có khi chỉ đơn giản là một file XML. Model thể hiện rõ các thao tác với cơ sở dữ liệu như cho phép xem, thêm/sửa/xóa, truy xuất, xử lý hay lọc dữ liệu,...

View là phần giao diện dành cho người sử dụng. Nơi mà người dùng có thể lấy được thông tin dữ liệu của MVC thông qua các thao tác truy vấn như tìm kiếm hoặc sử dụng thông qua các website. View có chức năng ghi nhận hoạt động của người dùng để tương tác với Controller. Tuy nhiên, View không có mối quan hệ trực tiếp với Controller, cũng không được lấy dữ liệu từ Controller mà chỉ hiển thị yêu cầu chuyển cho Controller.

Controller là bộ phận có nhiệm vụ xử lý các yêu cầu người dùng đưa đến thông qua view, từ đó đưa ra dữ liệu phù hợp với người dùng. Bên cạnh đó, Controller còn có chức năng kết nối với model.

4.1.1.2 Mô hình MVP

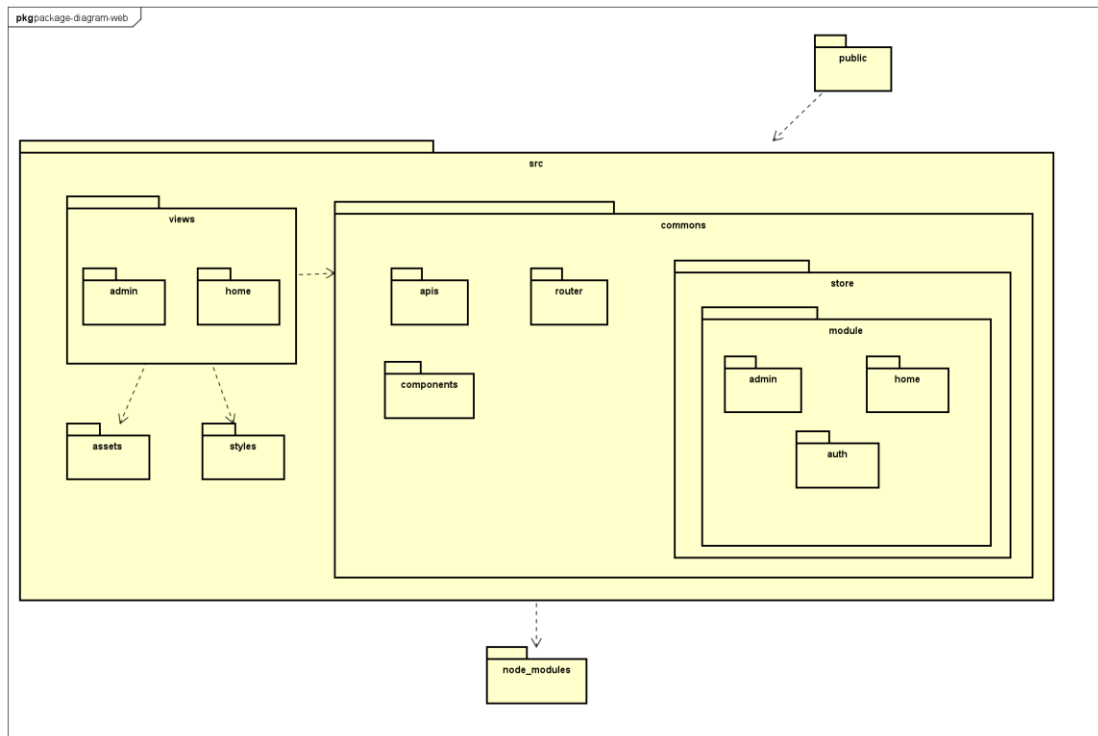
MVP là một mô hình kiến trúc lập trình gần giống như MVC. MVP có nhiều điểm được kế thừa từ MVC trong đó Controller được thay bằng Presenter. Mô hình này chia ứng dụng thành 3 phần chính: Model, View và Presenter. MVP là một mô hình kiến trúc hướng giao diện người dùng, được thiết kế để tạo thuận lợi cho việc kiểm thử đơn vị (unit testing) và tăng tính tách biệt giữa tầng dữ liệu và tầng hiển thị dữ liệu trong mô hình MVC.

Model, tương tự như ở mô hình MVC, là bộ phận có chức năng lưu trữ toàn bộ dữ liệu của ứng dụng. Model bao gồm các lớp mô tả business logic, định nghĩa business rules cho dữ liệu.

View là thành phần tương tác trực tiếp và nhận yêu cầu từ người dùng như XML, Activity, fragments. Nó không bao gồm bất kỳ việc xử lý logic nào.

Presenter sẽ nhận input của người dùng thông qua View, rồi xử lý dữ liệu của người dùng với sự trợ giúp của Model và trả kết quả về View. Presenter giao tiếp với View qua interface. Interface được định nghĩa trong lớp Presenter. Trong cấu trúc MVP, Presenter thao túng Model và cập nhật ở View. View và Presenter tách biệt với nhau hoàn toàn và giao tiếp với nhau qua thông qua interface. Vì nếu tách riêng từng phần ở View sẽ dễ dàng cho việc kiểm thử ứng dụng ở MVP.

4.1.2 Thiết kế tổng quan



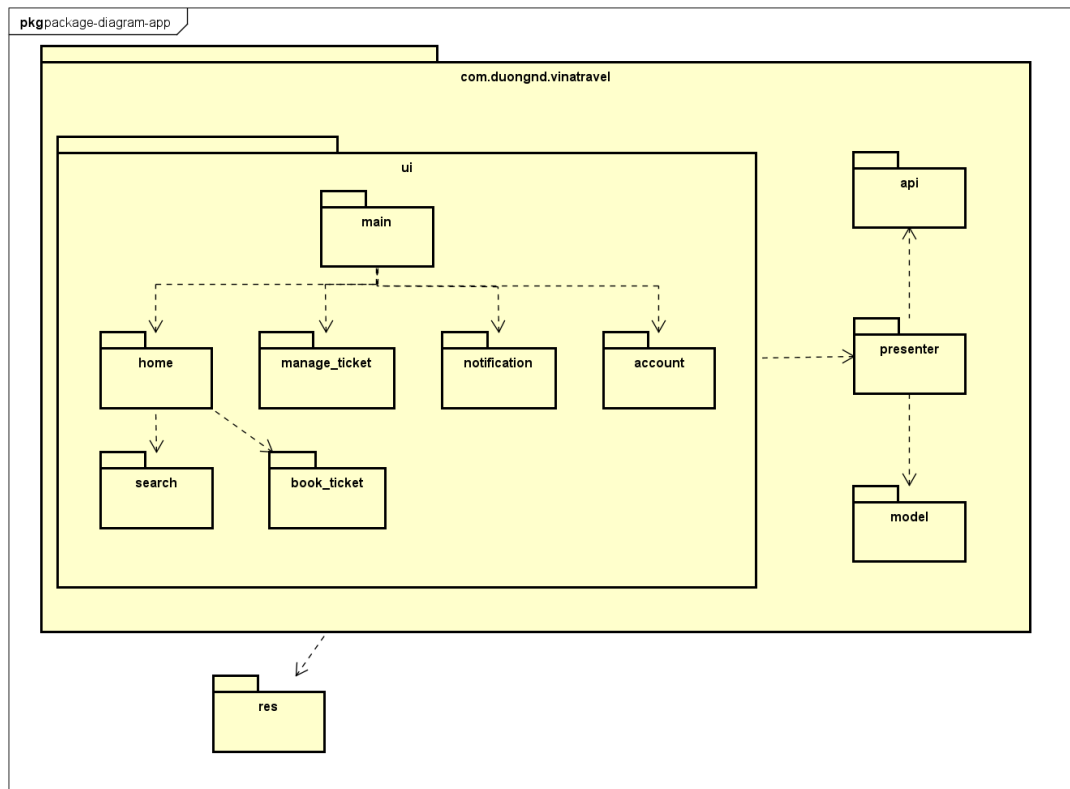
Hình 14 Sơ đồ gói cho Front-end Web

Front-end web của hệ thống sử dụng Vue và quản lý state bằng Vuex. Cấu trúc của project gồm có:

- node_modules là gói chứa những thư viện javascript được import và sử dụng
- gói src chứa mã nguồn chính của project
- gói public chứa tất cả tài nguyên công khai mà người dùng có thể truy cập được trên browser

Mã nguồn của project được chia thành các package nhỏ hơn phù hợp với từng chức năng:

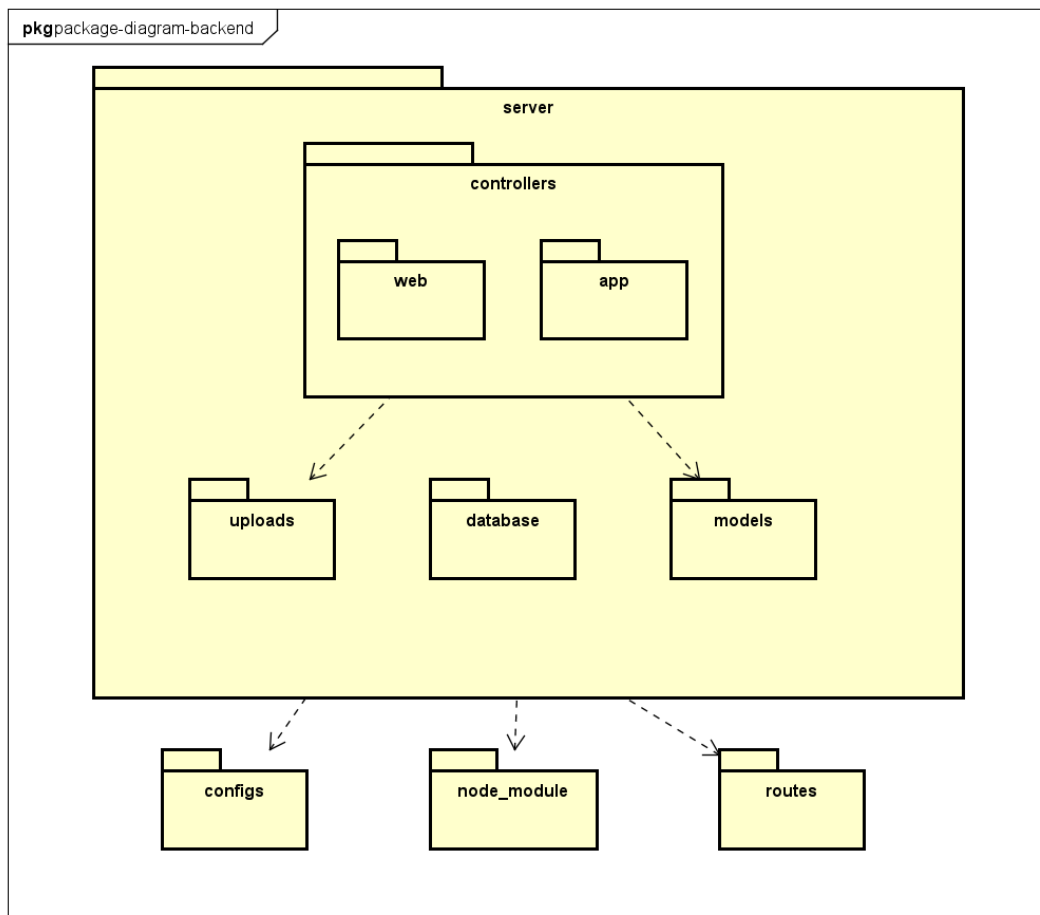
- Gói views chứa các trang chính của hệ thống bao gồm các trang của admin và các trang home cho công ty vận tải
- Gói commons chứa các tài nguyên dùng chung như: api, route, các components và store. Store là gói sẽ quản lý các state dùng chung của cả hệ thống hay có thể hiểu là nơi quản lý dữ liệu dùng chung của cả hệ thống.



Hình 15 Biểu đồ gói cho Front-end app mobile

Hình 15 là biểu đồ gói cho app mobile, được thiết kế theo chuẩn thiết kế ứng dụng android, gồm hai thư mục chính:

- Thư mục res chứa giao diện hiển thị lên phía người dùng.
- Thư mục com.duongnd.vinatravel là thư mục chứa phần code chính trong đó: (i) ui là chứa các code liên quan đến giao diện người dùng, (ii) api chứa class kết nối tới server và gọi các api, (iii) model chứa các class thực thể và (iv) presenter để trao đổi thông tin giữa ui và model



Hình 16 Biểu đồ gói cho Back-end/Server

Package models chứa các class thực thể.

Package database chứa class giúp liên kết đến hệ cơ sở dữ liệu.

Package uploads chứa các file ảnh, media của hệ thống.

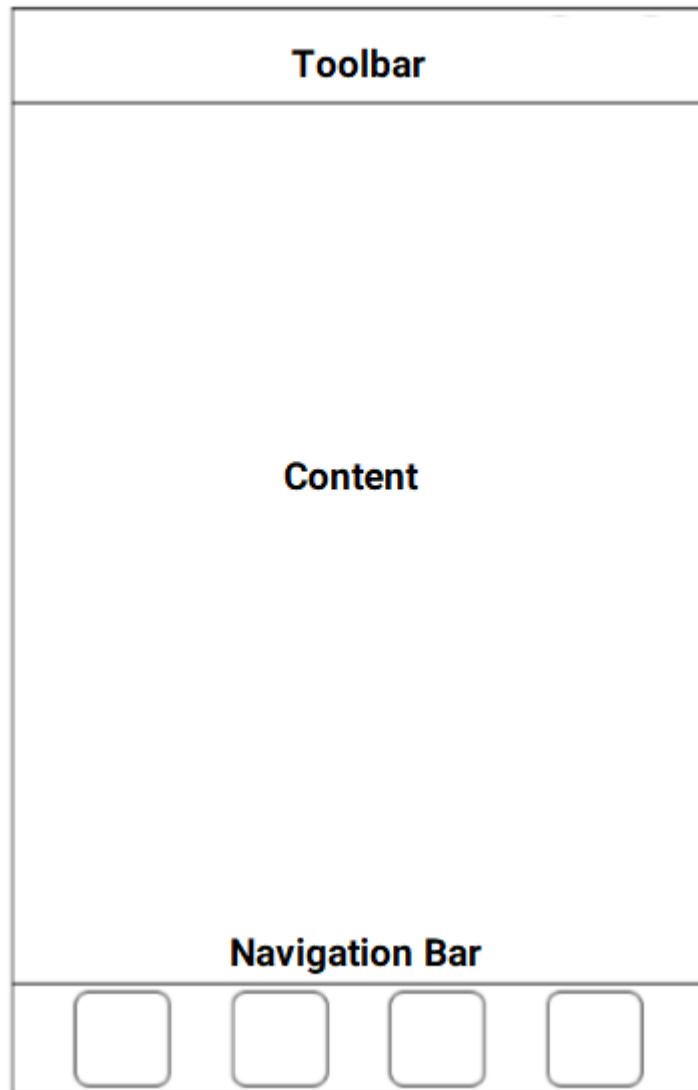
Package controllers chứa các Controller của hệ thống, thực hiện các logic nghiệp vụ.

Package node_module là gói chứa những thư viện javascript được import và sử dụng.

Package routes là gói chứa những file định tuyến đường dẫn api cho phía client gọi.

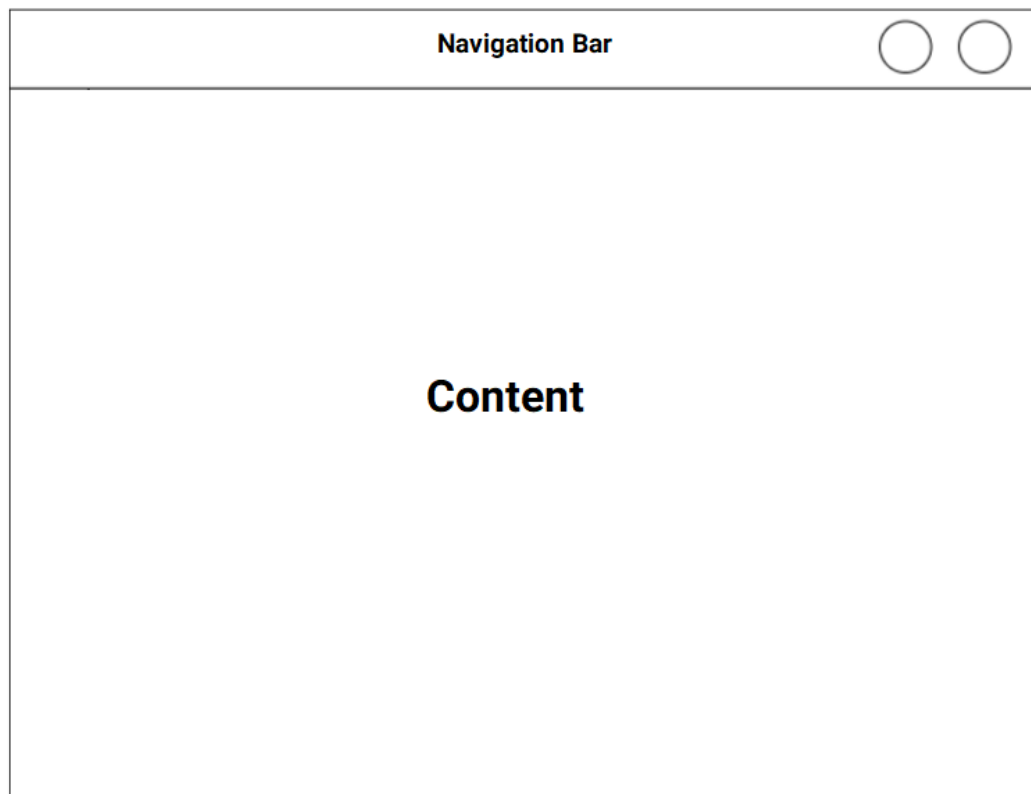
4.2 Thiết kế chi tiết

4.2.1 Thiết kế giao diện



Hình 17. Giao diện màn hình app mobile

App mobile được thiết kế trên nền tảng Android, phù hợp với hầu hết màn hình điện thoại android hiện tại, nhưng tương thích nhất màn hình có độ phân giải là 1280x720 pixels. Bố cục gồm ba phần chính là Toolbar, Content, và Navigation Bar. Toolbar chứa thanh tiêu đề của màn hình, đồng thời có thể có các giao diện tương tác nhanh tại đây. Content là nội dung chính, là các fragment hiển thị các thông tin chính của từng màn hình. Navigation Bar là thành phần cố định điều hướng các màn hình. Content và Toolbar thay đổi phù hợp với từng nội dung hiển thị.



Hình 18. Giao diện website

Giao diện web được thiết kế responsive, phù hợp với nhiều kích thước màn hình khác nhau.

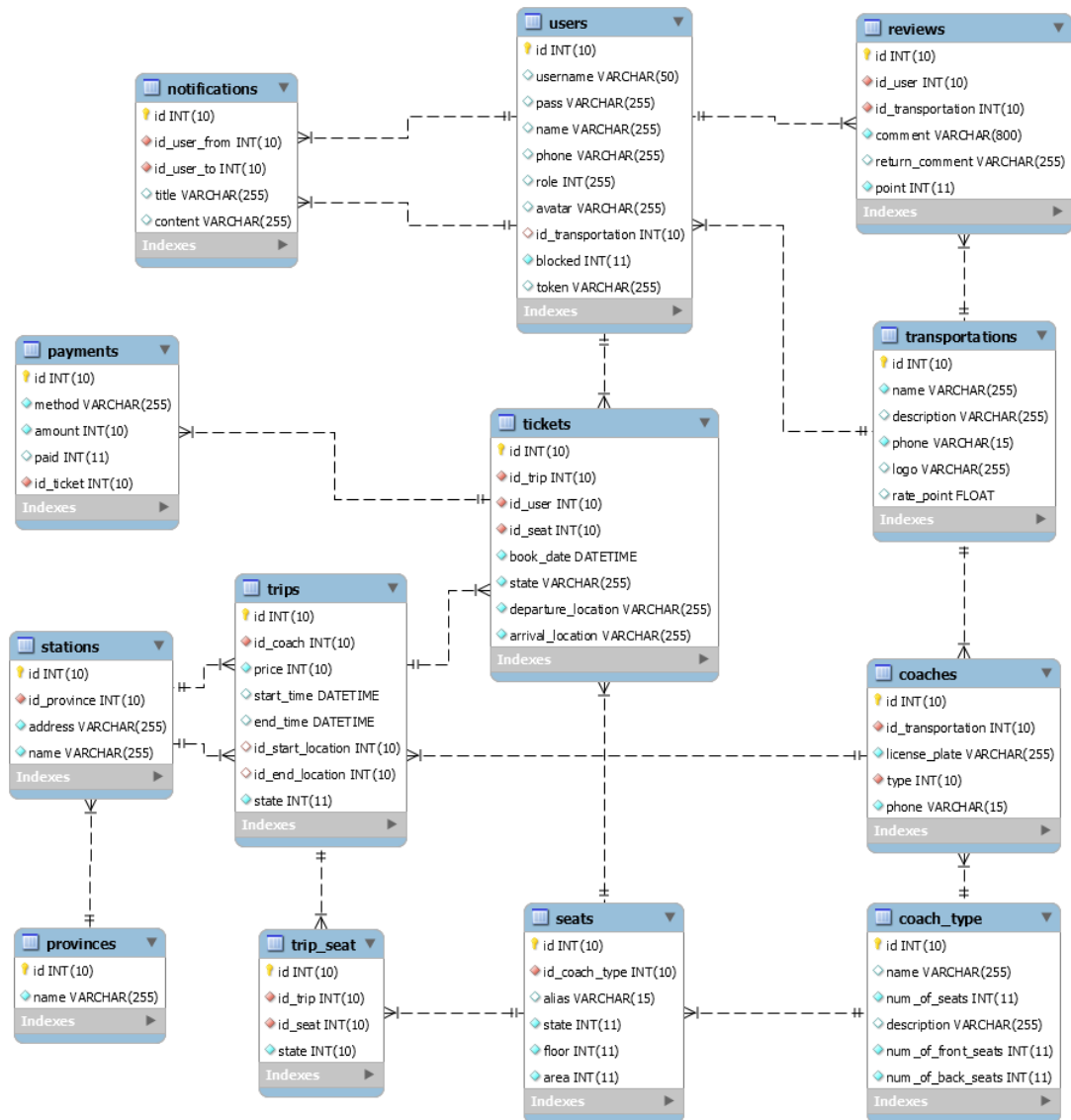
Website không yêu cầu load lại trang sau mỗi thao tác nên có tính tương tác cao với người dùng. Ngoài ra, giao diện ứng dụng sử dụng các icon thể hiện các chức năng một cách trực quan, giúp người dùng có thể dễ dàng làm quen và sử dụng hệ thống.

Tông màu chủ đạo của ứng dụng là xanh dương, mang lại cảm giác dễ chịu cho người sử dụng, không bị loá mắt khi sử dụng lâu dài.

Bố cục đơn giản với 2 phần chính: Navigation Bar là thanh điều hướng giúp người dùng có thể chuyển qua các chức năng khác nhanh chóng, thuận tiện. Content là nội dung ứng với từng chức năng.

4.2.2 Thiết kế lớp

4.2.3 Thiết kế cơ sở dữ liệu



Hình 19. Sơ đồ thiết kế cơ sở dữ liệu

Chi tiết các bảng trong cơ sở dữ liệu:

Bảng 9. Thiết kế chi tiết bảng users

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh người dùng
username	varchar(50)		Tên đăng nhập

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
pass	varchar(255)		Mật khẩu
phone	varchar(255)		Số điện thoại
name	varchar(255)		Họ tên người dùng
role	int(50)		Chức năng của người dùng: 0 – admin 1 – tài khoản công ty vận tải 2 – người dùng
id_transportation	int	Khóa ngoại	Định danh công ty vận tải mà người dùng là nhân viên (chỉ có với user có role là tài khoản công ty vận tải)
blocked	int		Trạng thái tài khoản người dùng: 0 – bình thường 1 – bị chặn
token	varchar(255)		Token dùng để xác thực tài khoản

Bảng 10. Thiết kế chi tiết bảng transportations

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh công ty vận tải
name	varchar(255)		Tên công ty vận tải

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
description	varchar(255)		Thông tin thêm về công ty vận tải
phone	varchar(15)		Số điện thoại
logo	varchar(255)		Đường dẫn file logo công ty vận tải
rate_point	float		Điểm trung bình do người dùng đánh giá

Bảng 11. Thiết kế chi tiết bảng coach_type

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh loại xe
name	varchar(255)		Tên loại xe
description	varchar(255)		Mô tả
num_of_seats	int		Số chỗ ngồi trên xe
num_of_front_seats	int		Số chỗ ngồi ở đầu xe
num_of_back_seats	int		Số chỗ ngồi ở cuối xe

Bảng 12. Thiết kế chi tiết bảng coaches

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh xe
id_transportation	int	Khóa ngoại	Định danh công ty vận tải
license_plate	varchar(255)		Biển số xe

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
phone	varchar(15)		Số điện thoại
type	int	Khóa ngoại	Loại xe

Bảng 13. Thiết kế chi tiết bảng seats

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh ghế
alias	varchar(15)		Tên ghế
id_coach_type	int	Khóa ngoại	Định danh loại xe
state	int		Trạng thái ghế: 0 – đang trống 1 – đã được đặt
floor	int		Tầng của ghế (tầng 1/ tầng 2)
area	int		Khu vực ghế (trước/sau)

Bảng 14. Thiết kế chi tiết bảng trip_seat

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	
id_trip	int	Khóa ngoại	Định danh chuyến đi
id_seat	int	Khóa ngoại	Định danh ghế
state	int		Trạng thái ghế: 0 – đang trống

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
			1 – đã được đặt

Bảng 15. Thiết kế chi tiết bảng provinces

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh tỉnh/thành
name	varchar(255)		Tên tỉnh/thành

Bảng 16. Thiết kế chi tiết bảng stations

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh địa điểm
id_province	int	Khóa ngoại	Định danh tỉnh/thành
address	varchar(255)		Địa chỉ
name	varchar(255)		Tên địa điểm

Bảng 17. Thiết kế chi tiết bảng trips

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh chuyến đi
id_coach	int	Khóa ngoại	Định danh xe
price	int		Giá vé
start_time	date		Thời gian xuất phát
end_time	date		Thời gian kết thúc dự kiến

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id_start_location	int	Khóa ngoại	Định danh địa điểm xuất phát
id_end_location	int	Khóa ngoại	Định danh địa điểm kết thúc
state	int		Trạng thái chuyển đi: 0 – sẵn sàng 1 – đã thực hiện 2 – đã hoãn

Bảng 18. Thiết kế chi tiết bảng payments

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh đơn thanh toán
method	varchar(255)		Phương thức thanh toán
amount	int		Tổng tiền thanh toán
paid	int		Đã thanh toán/Chưa thanh toán
id_ticket	int	Khóa ngoại	Định danh vé xe

Bảng 19. Thiết kế chi tiết bảng tickets

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh vé
id_trip	int	Khóa ngoại	Định danh chuyến đi

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id_user	int	Khóa ngoại	Định danh người dùng
id_seat	int	Khóa ngoại	Định danh ghế
book_date	date		Thời gian đặt vé
state	varchar(255)		Trạng thái vé
departure_location	varchar(255)		Địa điểm trung chuyển đón (nếu có)
arrival_location	varchar(255)		Địa điểm trung chuyển xuống (nếu có)

Bảng 20. Thiết kế chi tiết bảng reviews

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh đánh giá
id_user	int	Khóa ngoại	Định danh người dùng
id_transportation	int	Khóa ngoại	Định danh công ty vận tải
comment	varchar(255)		Bình luận đánh giá
return_comment	varchar(255)		Bình luận phản hồi của công ty vận tải
point	int		Điểm đánh giá

Bảng 21. Thiết kế chi tiết bảng notifications

Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
id	int	Khóa chính	Định danh thông báo
id_user_from	int	Khóa ngoại	Định danh tài khoản gửi thông báo
id_user_to	int	Khóa ngoại	Định danh tài khoản nhận thông báo
tittle	varchar(255)		Tiêu đề thông báo
content	varchar(255)		Nội dung thông báo

4.3 Xây dựng ứng dụng

4.3.1 Thư viện và công cụ sử dụng

Thư viện và công cụ được sử dụng cho việc hoàn thành đồ án được mô tả trong **Bảng 17** dưới đây:

Bảng 22. Danh sách thư viện và công cụ sử dụng

Mục đích	Công cụ	Địa chỉ URL
IDE lập trình	Visual Studio Code	https://code.visualstudio.com/
IDE lập trình	Android Studio	https://developer.android.com/
Xây dựng backend	NodeJS	https://nodejs.org/
Xây dựng frontend web	VueJS	https://vuejs.org
Database hệ thống	MySQL	https://www.mysql.com/
Xác thực số điện thoại người dùng	Firebase	https://firebase.google.com/

4.3.2 Kết quả đạt được

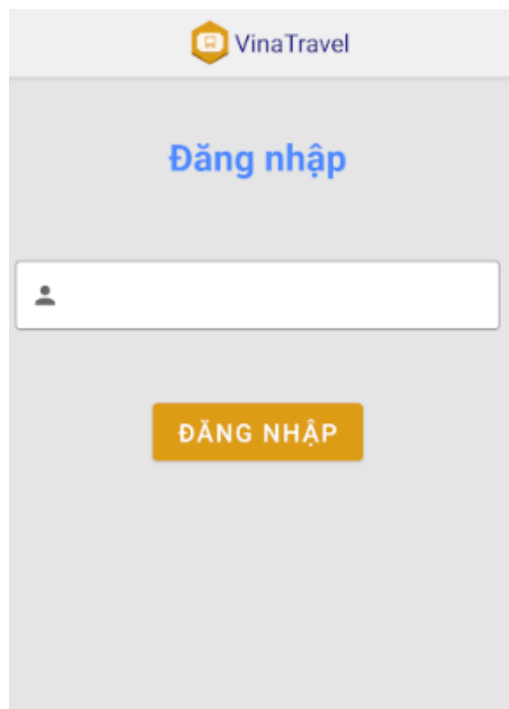
Qua quá trình tìm hiểu, em đã xây dựng được một hệ thống đặt vé xe đường dài, hệ thống giúp các công ty vận tải có thể tự quản lý các chuyến đi, doanh số bán hàng,... và từ những thông tin của các nhà xe đó, người dùng có thể đặt vé một cách nhanh chóng, thuận tiện.

Dưới đây là một số thống kê về các thông số của source code hệ thống:

Thông tin	Số liệu
Số dòng code	Khoảng 15000 dòng
Số file mã nguồn	Khoảng 300 files
Dung lượng mã nguồn	Front-end Web: 3MB, Mobile: 3MB, Back-end: 6MB

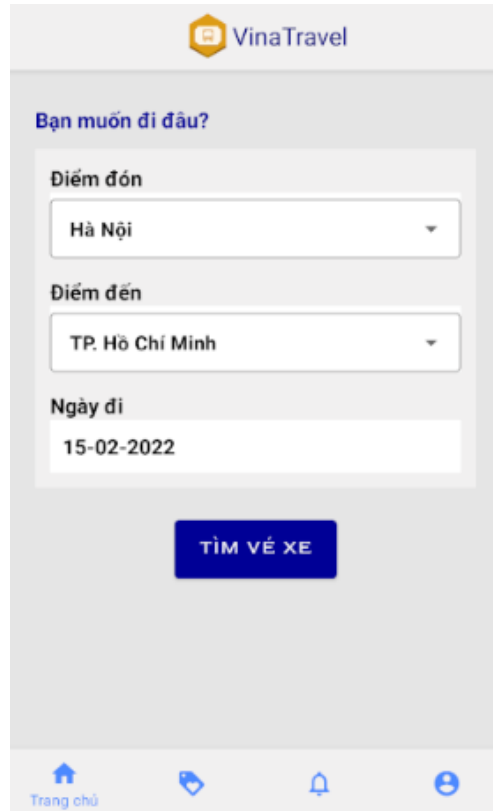
4.3.3 Minh họa các chức năng chính

4.3.3.1 App mobile



Hình 20. Giao diện đăng nhập app mobile

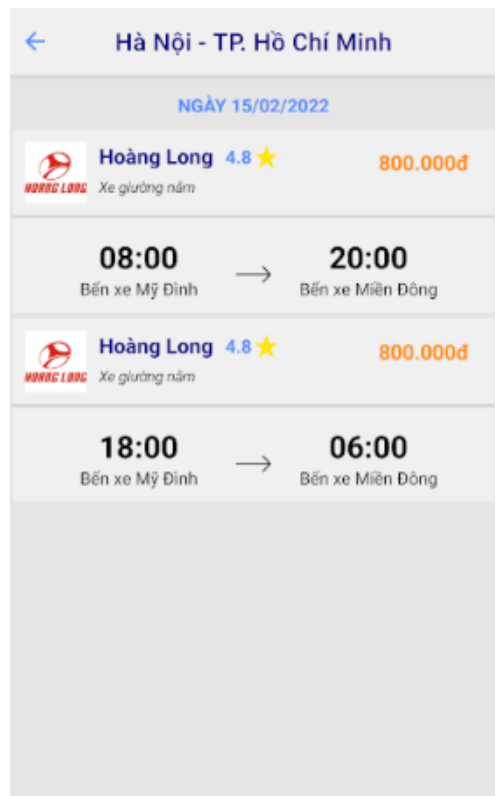
Ứng dụng cho phép người dùng đăng nhập bằng số điện thoại, sau khi nhập số điện thoại, một mã OTP sẽ được gửi về số người dùng đã nhập để có thể xác thực người dùng đang sở hữu số điện thoại đó. Với cách đăng nhập này, người dùng có thể bớt đi nỗi lo sợ khi quên mật khẩu như ở các hệ thống khác. Giao diện được thiết kế tinh giản, không màu mè, tạo cảm giác dễ chịu, dễ nhìn cho người dùng.



Hình 21. Giao diện tìm kiếm vé xe

Sau khi đăng nhập thành công, người dùng sẽ được chuyển đến màn hình trang chủ, cũng chính là màn hình tìm kiếm vé xe – một trong những chức năng quan trọng nhất của app. Ở đây, người dùng sẽ nhập 3 thông tin cơ bản của chuyến xe cần tìm: điểm đón, điểm đến và ngày đi, từ đó, hệ thống sẽ hiển thị ra những kết quả phù hợp với yêu cầu của người dùng.

Với giao diện trang chủ này, người dùng cũng có thể chuyển sang các tab khác bằng cách click vào biểu tượng ở thanh điều hướng cuối màn hình để thực hiện các chức năng khác như: quản lý vé xe, thông báo, thông tin cá nhân.



Hình 22. Giao diện chọn chuyến đi

Hệ thống sẽ tìm kiếm những chuyến đi phù hợp với yêu cầu của người dùng và hiển thị lên màn hình “Chọn chuyến đi”, thông tin của từng chuyến đi cũng sẽ được hiển thị ở đây, bao gồm cả thông tin chi tiết chuyến đi và điểm đánh giá của người dùng với công ty vận tải tương ứng. Để chọn chuyến đi người dùng chỉ cần click vào chuyến đi tương ứng hiển thị trên màn hình.

Chọn ghế

☐ Ghế trống ☒ Ghế đang chọn
☐ Ghế không thể chọn

Tầng 1					Tầng 2				
A1	B1	C1	A2	B2	C2				
A3	B3	C3	A4	B4	C4				
A5	B5	C5	A6	B6	C6				
A7	B7	C7	A8	B8	C8				
A9	B9	C9	A10	B10	C10				
D1	D3	D5	D7	D9	D2	D4	D6	D8	D10

Mã ghế: A6

Tổng số tiền: 800000

TIẾP TỤC

Hình 23. Giao diện chọn vị trí ngồi

Sau khi người dùng chọn được chuyến đi phù hợp, hệ thống sẽ hiển thị thông tin về các vị trí ngồi trên xe thực hiện chuyến đi đó. Sơ đồ các vị trí trên xe cũng sẽ được hiển thị để người dùng dễ hình dung một cách rõ ràng nhất có thể.

Những ghế có nền xám đậm sẽ là những ghế đã được đặt trước đó và người dùng không thể chọn chúng. Ghế có nền trắng là những ghế trống, người dùng có thể chọn và những ghế được chọn thì sẽ chuyển sang nền vàng nổi bật. Người dùng có thể bỏ chọn ghế bằng cách click vào ghế đã chọn một lần nữa.

Với những ghế đã chọn, thông tin tên ghế và tổng tiền phải thanh toán sẽ được cập nhật ở bên dưới màn hình.

←

Chọn điểm lên

Bến xe Mỹ Đình

Vị trí khác

Mã ghếA6

Tổng số tiền800000

TIẾP TỤC

←

Chọn điểm xuống

Bến xe Miền Đông

Vị trí khác

Mã ghếA6

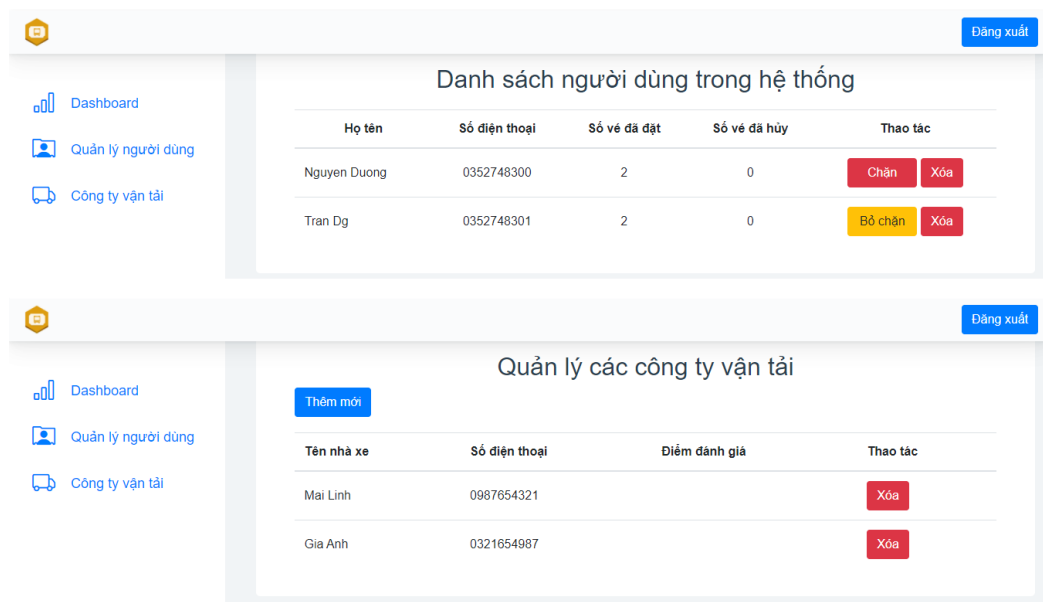
Tổng số tiền800000

TIẾP TỤC

Hình 24. Giao diện chọn điểm lên/xuống

Sau khi chọn được vị trí ghế, người dùng sẽ chọn các điểm lên/xuống để hoàn tất quá trình đặt vé. Ngoài hai địa điểm công ty vận tải sẽ thiết lập mặc định sẵn, người dùng có thể nhập địa điểm khác trên lộ trình của chuyến đi để có thể được xuống gần điểm mong muốn hơn.

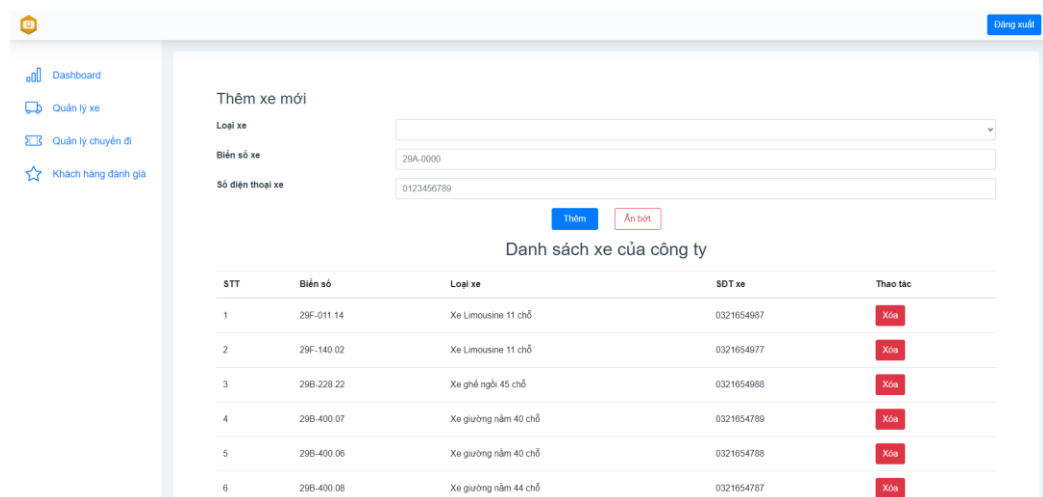
4.3.3.2 Web cho Admin hệ thống



Hình 25. Giao diện Quản lý người dùng và Quản lý công ty vận tải của Admin hệ thống

Với các công ty vận tải, Admin có thể quản lý các thông tin, thêm mới hoặc xóa một công ty bất kỳ khỏi hệ thống. Còn đối với người dùng, ngoài chức năng xem, xóa người dùng, Admin có thể chặn/bỏ chặn người dùng đó, với những người dùng có số lần hủy vé, bùng vé nhiều hệ thống sẽ tự động chặn, và để khôi phục người dùng phải liên lạc với hệ thống qua email rồi từ đó, admin sẽ xem xét tình huống và đưa ra quyết định bỏ chặn hay không. Những tài khoản bị chặn sẽ không thể đăng nhập cũng như sử dụng các chức năng của hệ thống.

4.3.3.3 Web cho phía công ty vận tải



Hình 26. Giao diện quản lý xe

Các công ty vận tải sẽ có trách nhiệm quản lý các xe của mình, mỗi xe sẽ có biển số xe riêng để phân biệt và có số điện thoại riêng để khi gần đến giờ xuất phát hoặc khi có thắc mắc, người dùng có thể gọi cho tài xế hoặc lơ xe của chuyến họ đã đặt.

STT	Tuyến đường	Thời gian	Biển số xe	Giá vé	Vé đã đặt	Thao tác
1	Hà Nội - Hà Tĩnh	21.00 07/02/2022 - 04.00 08/02/2022	29B-400.07	210000	27/40	Xem Hoàn
2	Hà Tĩnh - Hà Nội	13.00 21/02/2022 - 20.00 21/02/2022	29B-400.07	250000	33/40	Xem Hoàn
3	Hà Nội - Hà Tĩnh	13.00 22/01/2023 - 20.00 22/01/2023	29B-400.07	240000	15/40	Xem Hoàn
4	Hà Nội - Hà Tĩnh	13.00 25/01/2022 - 20.00 25/01/2022	29B-400.07	250000	23/40	Xem Hoàn
5	Hà Nội - Hà Tĩnh	01.00 28/02/2020 - 08.00 28/02/2020	29B-400.07	250000	27/40	Xem Hoàn
6	Hà Nội - Hà Tĩnh	06.00 28/02/2020 - 13.00 28/02/2020	29B-400.06	250000	30/44	Xem Hoàn
7	Hà Nội - Hà Tĩnh	11.00 28/02/2020 - 18.00 28/02/2020	29B-400.08	250000	10/42	Xem Hoàn
8	Hà Nội - Hà Tĩnh	16.00 28/02/2020 - 23.00 28/02/2020	29B-400.09	250000	27/46	Xem Hoàn

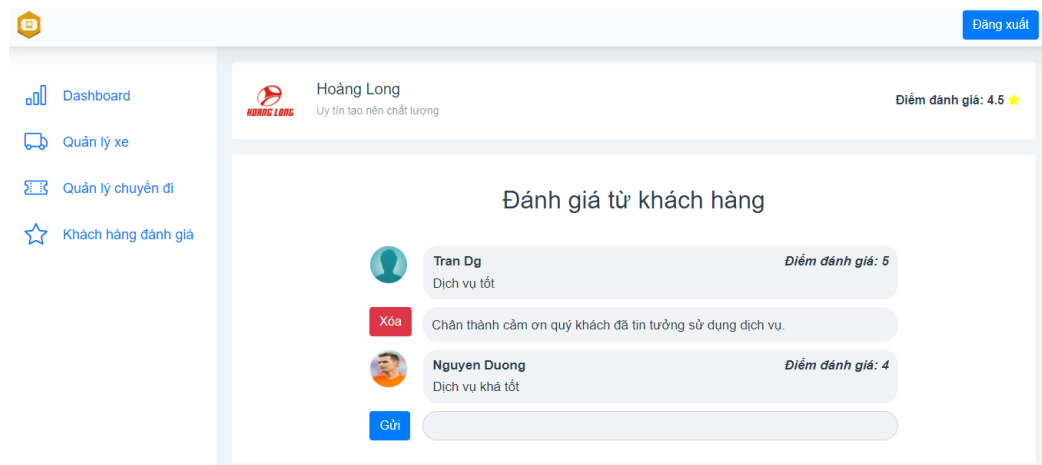
Hình 27. Giao diện quản lý chuyến đi

Các công ty vận tải sẽ là bên tạo các chuyến đi và danh sách những chuyến đi đó sẽ được hiển thị ở màn hình Quản lý chuyến đi. Với những chuyến đi gần ngày giờ xuất phát và số vé được đặt ở mức ổn hay nói cách khác chuyến đi đó đã sẵn sàng thì sẽ có nền màu xanh. Những chuyến đi có số vé đặt ở mức còn thấp, gây lỗi cho công ty vận tải sẽ được hiển thị với nền màu đỏ để công ty có thể cân nhắc ghép chuyến, hoãn chuyến hay vẫn cho xuất phát.

STT	Tuyến đường	Thời gian	Biển số xe	Giá vé	Vé đã đặt	Chọn
1	Hà Nội - Hà Tĩnh	21.00 07/02/2022 - 04.00 08/02/2022	29B-400.07	210000	27/40	<input checked="" type="checkbox"/>
2	Hà Tĩnh - Hà Nội	13.00 21/02/2022 - 20.00 21/02/2022	29B-400.07	250000	33/40	<input type="checkbox"/>
3	Hà Nội - Hà Tĩnh	13.00 22/01/2023 - 20.00 22/01/2023	29B-400.07	240000	27/40	<input type="checkbox"/>
4	Hà Nội - Hà Tĩnh	13.00 25/01/2022 - 20.00 25/01/2022	29B-400.07	250000	23/40	<input type="checkbox"/>
5	Hà Nội - Hà Tĩnh	01.00 28/02/2020 - 08.00 28/02/2020	29B-400.07	250000	37/40	<input type="checkbox"/>
6	Hà Nội - Hà Tĩnh	06.00 28/02/2020 - 13.00 28/02/2020	29B-400.06	250000	30/44	<input type="checkbox"/>
7	Hà Nội - Hà Tĩnh	11.00 28/02/2020 - 18.00 28/02/2020	29B-400.08	250000	10/42	<input checked="" type="checkbox"/>
8	Hà Nội - Hà Tĩnh	16.00 28/02/2020 - 23.00 28/02/2020	29B-400.09	250000	15/46	<input checked="" type="checkbox"/>

Hình 28. Giao diện Ghép chuyến đi

Hiện thị tương tự với màn hình Quản lý chuyến đi, tuy nhiên ở đây các công ty vận tải có thể chọn những chuyến đi gây lỗi (có nền màu đỏ) để tiến hành ghép các chuyến đi đó thành những chuyến đi tốt hơn cho công ty.



Hình 29. Giao diện Đánh giá từ khách hàng và phản hồi

Công ty vận tải có thể xem các đánh giá từ khách hàng sau khi sử dụng dịch vụ bên mình và phản hồi những đánh giá đó.

4.4 Kiểm thử

4.4.1 Kiểm thử tương thích

Bảng 23. Thống kê kiểm thử tương thích

Thiết bị	Thông số	Giao diện	Chức năng
Oppo F3	Màn hình: IPS LCD 5.5" Full HD RAM: 4GB	Đạt	Đạt
Google Pixel 3	Màn hình: P-OLED 5.5" Full HD+ RAM: 4GB	Đạt	Đạt

4.4.2 Kiểm thử chức năng

Để kiểm thử các chức năng chính của hệ thống, em tiến hành kiểm thử bằng phương pháp kiểm thử hộp đen.

Bảng 24. Kiểm thử chức năng Đặt vé của người dùng

Chức năng	Test case	Kết quả
Đặt vé	Tìm kiếm vé xe	Đạt
	Chọn chỗ trên xe, cập nhật giá tiền	Đạt
	Chọn điểm lên, xuống, điền thêm thông tin khác	Đạt
Quản lý vé	Cập nhật thông tin vé đã đặt, đã đi, đã hủy	Đạt
	Hủy vé xe	Đạt

Bảng 25. Kiểm thử chức năng quản lý phía công ty vận tải

Chức năng	Test case	Kết quả
Quản lý xe	Thêm xe mới	Đạt
	Cập nhật thông tin xe	Đạt
Quản lý chuyến đi	Thêm chuyến đi	Đạt
	Ghép chuyến đi	Đạt
	Hoãn chuyến đi	Đạt
Thông báo	Gửi thông báo tới người dùng	Đạt

4.5 Triển khai

Ứng dụng mobile được triển khai trên thiết bị di động chạy hệ điều hành Android. Server được triển khai trên localhost tại laptop cá nhân. Bảng dưới đây mô tả thông tin chi tiết các thiết bị đã được dùng để triển khai hệ thống.

Bảng 26. Thống kê các thiết bị kiểm thử cho hệ thống

Thiết bị	Vai trò	Cấu hình
Oppo F3	App mobile cho người dùng	Màn hình: IPS LCD 5.5" Full HD RAM: 4GB
Laptop HP 15-ay169TX Z6X61PA	Server	Intel Core i7-7500U (2.70 GHz Up to 3.50GHz, 4MB) 8GB RAM DDR4
Laptop HP 15-ay169TX Z6X61PA	Giao diện web	Màn hình: 15.6" HD 8GB RAM DDR4

Chương 5 Kết luận và hướng phát triển

5.1 Kết luận

Sau thời gian một kỳ học tìm hiểu, phân tích, nghiên cứu với sự hướng dẫn của thầy Trần Đình Khang, em đã xây dựng được cơ bản một hệ thống đặt vé xe đường dài. Hệ thống đã giải quyết được một số vấn đề lớn trong thời buổi hiện nay như (i) cung cấp môi trường quản lý dễ dàng, tiện lợi cho các công ty vận tải, (ii) cung cấp ứng dụng giúp người dùng đặt vé một cách nhanh chóng, thuận tiện thay vì đặt vé như cách truyền thống, (iii) cung cấp giải pháp cho vấn đề ghép các chuyến đi để tránh lãng phí tài nguyên.

Trong quá trình phát triển hệ thống, em cũng được tiếp xúc và học tập, trau dồi về nhiều công nghệ, kiến thức mới như một số framework phát triển frontend, backend: VueJS, Android, NodeJS, RESTful API, quy trình triển khai, xây dựng một hệ thống,...

Tuy nhiên, vì thời gian có hạn cũng như kiến thức, khả năng của bản thân còn hạn chế nên hệ thống còn có nhiều thiếu sót: giao diện chưa thực sự hài hòa về bố cục hay màu sắc, trải nghiệm người dùng ở một vài chức năng còn chưa tốt,...

5.2 Hướng phát triển

Trong tương lai, hệ thống sẽ được nâng cấp và hoàn thiện hơn nữa về các tính năng cũng như trải nghiệm cho người dùng. Hệ thống sẽ có thể tích hợp thêm chức năng thanh toán từ các ngân hàng hay ví điện tử, tích hợp bản đồ Google/Apple giúp người dùng có thể định vị vị trí của xe, của bản thân trong thời gian thực.

Ngoài ra, ứng dụng mobile sẽ được phát triển trên các nền tảng khác, cũng như nghiệp vụ của các công ty vận tải sẽ được thay đổi để phù hợp với nhu cầu thực tế hơn nữa.

Tài liệu tham khảo

[1] B. Burd, Java Programming for Android Developers for Dummies, 2nd Edition, 2016.

[2] B. A. Syed, Beginning Node.js, 2014.

[3] "Introduction: Vue.js," [Online]. Available: vuejs.org.