**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**TRẦN LÊ TUẤN KIỆT**

**PHÁT TRIỂN HỆ THỐNG ĐẶT VÉ XE OUBUS SỬ DỤNG DJANGO VÀ REACTJS**

**ĐỒ ÁN NGÀNH**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**TP. HỒ CHÍ MINH, 2024BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**TRẦN LÊ TUẤN KIỆT**

**PHÁT TRIỂN HỆ THỐNG ĐẶT VÉ XE OUBUS SỬ DỤNG DJANGO VÀ REACTJS**

**Mã số sinh viên: 2151053033**

**ĐỒ ÁN NGÀNH**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Giảng viên hướng dẫn: ThS. DƯƠNG HỮU THÀNH**

**TP. HỒ CHÍ MINH, 2024**

LỜI CẢM ƠN

Trước hết, em xin bày tỏ lòng biết ơn sâu sắc của mình dành cho Thầy ThS. Dương Hữu Thành vì Thầy đã hết lòng hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện môn Đồ án Ngành của mình.

Em xin chân thành cảm ơn tất cả các Thầy Cô thuộc khoa Công nghệ thông tin trường Đại học Mở Thành phố Hồ Chí Minh vì đã dìu dắt, dạy dỗ em về những kiến thức chuyên ngành và tinh thần trách nhiệm trong học tập để em có đủ khả năng thực hiện được môn Đồ án Ngành của mình.

Đồ án Ngành chính là kết quả của mọi sự cố gắng và nỗ lực của em trong quá trình học tập tại Trường. Tuy nhiên, đây là lần đầu tiên em thực hiện nên trong quá trình thực hiện đồ án ngành không thể tránh khỏi những thiếu sót, em rất mong sẽ nhận được những góp ý quý báu của tất cả các thầy cô để kết quả của em được hoàn thiện hơn.

Một lần nữa em xin chân thành cảm ơn.

Hồ Chí Minh, ngày 06 tháng 10 năm 2024

Sinh viên

Trần Lê Tuấn Kiệt

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

TÓM TẮT ĐỒ ÁN NGÀNH

Trong bối cảnh cơ sở 3 của trường Đại học Mở Thành phố Hồ Chí Minh được di chuyển xuống Huyện Nhà Bè, nhu cầu sử dụng dịch vụ xe bus của sinh viên trường ngày càng cao, nên việc xây dựng một hệ thống quản lý và đặt vé trực tuyến đảm bảo sinh viên có thể dễ dàng đặt vé và nhận thông báo nhanh chóng là vô cùng cần thiết.

Mục tiêu của đồ án là phát triển một hệ thống thân thiện với người sử dụng, cho phép sinh viên có thể đăng nhập, tra cứu lịch trình, đánh giá chuyến đi và đặt vé xe bus một cách tiện lợi. Bên cạnh đó, hệ thống cũng cung cấp công cụ quản lý cho nhân viên của nhà trường, giúp theo dõi và điều phối các chuyến xe một cách hiệu quả.

Giải pháp được đề xuất là sử dụng công nghệ Django để xử lý các logic nghiệp vụ và quản lý dữ liệu, kết hợp với công nghệ ReactJS nhằm cung cấp một giao diện trực quan, mượt mà. Quá trình phát triển hệ thống tuân thủ các nguyên tắc thiết kế module hóa, đảm bảo khả năng mở rộng và bảo trì dễ dàng.

Kết quả đạt được từ quá trình phát triển là một hệ thống có tính ổn định cao, cải thiện rõ rệt quy trình đặt vé, giúp tiết kiệm thời gian và nâng cao trải nghiệm của sinh viên khi sử dụng dịch vụ xe bus. Bên cạnh đó, việc sử dụng các công nghệ lập trình hiện đại của giúp hệ thống đạt hiệu suất tốt và dễ mở rộng hơn trong tương lai.

ABSTRACT

**MỤC LỤC**

[LỜI CẢM ƠN 1](#_Toc179138680)

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN 2](#_Toc179138681)

[TÓM TẮT ĐỒ ÁN NGÀNH 3](#_Toc179138682)

[ABSTRACT 4](#_Toc179138683)

[DANH MỤC TỪ VIẾT TẮT 8](#_Toc179138684)

[DANH MỤC HÌNH VẼ 9](#_Toc179138685)

[DANH MỤC BẢNG 11](#_Toc179138686)

[MỞ ĐẦU 12](#_Toc179138687)

[Chương 1. TỔNG QUAN ĐỀ TÀI 13](#_Toc179138688)

[1.1. Giới thiệu đề tài 13](#_Toc179138689)

[1.2. Lý do chọn đề tài 13](#_Toc179138690)

[1.3. Mục tiêu và phạm vi đề tài 14](#_Toc179138691)

[1.4. Phương pháp nghiên cứu 15](#_Toc179138692)

[1.5. Bố cục báo cáo 15](#_Toc179138693)

[Chương 2. CƠ SỞ LÝ THUYẾT 17](#_Toc179138694)

[2.1. Django 17](#_Toc179138695)

[2.1.1. Giới thiệu Django 17](#_Toc179138696)

[2.1.2. Ưu điểm của Django 18](#_Toc179138697)

[2.1.3. Django REST framework 18](#_Toc179138698)

[2.1.4. Cấu trúc project Django REST API 20](#_Toc179138699)

[2.2. ReactJS 23](#_Toc179138700)

[2.2.1. Giới thiệu ReactJS 23](#_Toc179138701)

[2.2.2. Một số đặc điểm của React 23](#_Toc179138702)

[2.2.3. Vòng đời ReactJS 24](#_Toc179138703)

[2.3. Các công cụ hỗ trợ 25](#_Toc179138704)

[2.3.1. Cloudinary 25](#_Toc179138705)

[2.3.2. MySQL 25](#_Toc179138706)

[2.3.3. PythonAnywhere 26](#_Toc179138707)

[Chương 3. HỆ THỐNG ĐẶT VÉ XE OUBUS 27](#_Toc179138708)

[3.1. Giới thiệu bài toán 27](#_Toc179138709)

[3.2. Phân tích hệ thống 28](#_Toc179138710)

[3.2.1. Chức năng Đăng nhập 28](#_Toc179138711)

[3.2.2. Chức năng Đăng ký 28](#_Toc179138712)

[3.2.3. Chức năng Đặt vé trực tuyến 29](#_Toc179138713)

[3.2.4. Chức năng Đăng ký gói Combo 30](#_Toc179138714)

[3.2.5. Chức năng Đánh giá chuyến xe 31](#_Toc179138715)

[3.2.6. Chức năng Tìm kiếm chuyến xe 33](#_Toc179138716)

[3.2.7. Chức năng Xem danh sách chuyến xe 35](#_Toc179138717)

[3.2.8. Chức năng Xem lịch sử đặt vé 36](#_Toc179138718)

[3.2.9. Chức năng Quản lý thông tin cá nhân 38](#_Toc179138719)

[3.2.10. Chức năng Quản lý đánh giá 39](#_Toc179138720)

[3.2.11. Chức năng Quản lý chuyến xe 41](#_Toc179138721)

[3.2.12. Chức năng Quản lý vé xe 42](#_Toc179138722)

[3.2.13. Chức năng Thống kê báo cáo 42](#_Toc179138723)

[3.3. Thiết kế hệ thống 44](#_Toc179138724)

[3.3.1. Sơ đồ tuần tự (Sequence Diagram) 44](#_Toc179138725)

[3.3.2. Sơ đồ hoạt động (Activity Diagram) 49](#_Toc179138726)

[3.3.3. Thiết kế dữ liệu 53](#_Toc179138727)

[3.4. Kiến trúc hệ thống 55](#_Toc179138728)

[3.5. Kết quả thực hiện 56](#_Toc179138729)

[3.5.1. Chức năng Xem danh sách chuyến xe 56](#_Toc179138730)

[3.5.2. Chức năng Đặt vé trực tuyến 57](#_Toc179138731)

[3.5.3. Chức năng Đăng ký gói Combo 59](#_Toc179138732)

[3.5.4. Chức năng Đánh giá chuyến xe 60](#_Toc179138733)

[3.5.5. Chức năng Xem lịch sử đặt vé 62](#_Toc179138734)

[3.5.6. Chức năng Quản lý thông tin cá nhân 63](#_Toc179138735)

[3.5.7. Chức năng Quản lý tuyến xe 64](#_Toc179138736)

[3.5.8. Chức năng Quản lý đánh giá 65](#_Toc179138737)

[3.5.9. Chức năng Thống kê báo cáo 67](#_Toc179138738)

[Chương 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 68](#_Toc179138739)

[4.1. Kết luận 68](#_Toc179138740)

[4.2. Hướng phát triển 69](#_Toc179138741)

[TÀI LIỆU THAM KHẢO 70](#_Toc179138742)

[PHỤ LỤC 71](#_Toc179138743)

DANH MỤC TỪ VIẾT TẮT

DANH MỤC HÌNH VẼ

[Hình 2.1: Mô hình MVT (Model-View-Template) 17](#_Toc179138744)

[Hình 2.2: Mô hình RESTful Web Services 19](#_Toc179138745)

[Hình 2.3: Cấu trúc chung của một project DRF 20](#_Toc179138746)

[Hình 2.4: Vòng đời hoạt động của React Component (nguồn: https://projects.wojtekmaj.pl/react-lifecycle-methods-diagram/) 24](#_Toc179138747)

[Hình 3.1: Usecase chức năng đăng nhập 28](#_Toc179138748)

[Hình 3.2: Usecase chức năng Đăng ký 29](#_Toc179138749)

[Hình 3.3: Usecase chức năng Đặt vé trực tuyến 29](#_Toc179138750)

[Hình 3.4: Usecase chức năng Đăng ký gói Combo 30](#_Toc179138751)

[Hình 3.5: Usecase chức năng Đánh giá chuyến xe 32](#_Toc179138752)

[Hình 3.6: Usecase chức năng Tìm kiếm chuyến xe 34](#_Toc179138753)

[Hình 3.7: Usecase chức năng Xem danh sách chuyến xe 35](#_Toc179138754)

[Hình 3.8: Usecase chức năng Xem lịch sử đặt vé 37](#_Toc179138755)

[Hình 3.9: Usecase chức năng Quản lý thông tin cá nhân 38](#_Toc179138756)

[Hình 3.10: Usecase chức năng Quản lý đánh giá 40](#_Toc179138757)

[Hình 3.11: Usecase Quản lý chuyến xe 41](#_Toc179138758)

[Hình 3.12: Usecase chức năng Quản lý vé xe 42](#_Toc179138759)

[Hình 3.13: Usecase chức năng Thống kê báo cáo 43](#_Toc179138760)

[Hình 3.14: Sơ đồ tuần tự cho usecase Đặt vé trực tuyến 45](#_Toc179138761)

[Hình 3.15: Sơ đồ tuần tự của usecase Đăng ký gói Combo 46](#_Toc179138762)

[Hình 3.16: Sơ đồ tuần tự của usecase Thêm đánh giá 47](#_Toc179138763)

[Hình 3.17: Sơ đồ tuần tự của usecase Xem lịch sử đặt vé 47](#_Toc179138764)

[Hình 3.18: Sơ đồ tuần tự của usecase Thêm chuyến xe 48](#_Toc179138765)

[Hình 3.19: Sơ đồ tuần tự usecase Thống kê số lượng vé 48](#_Toc179138766)

[Hình 3.20: Sơ đồ hoạt động của usecase Đặt vé trực tuyến 49](#_Toc179138767)

[Hình 3.21: Sơ đồ hoạt động của usecase Đăng ký gói Combo 50](#_Toc179138768)

[Hình 3.22: Sơ đồ hoạt động của usecase Thêm đánh giá 51](#_Toc179138769)

[Hình 3.23: Sơ đồ hoạt động của usecase Xem lịch sử đặt vé 51](#_Toc179138770)

[Hình 3.24: Sơ đồ hoạt động của usecase Thêm chuyến xe 52](#_Toc179138771)

[Hình 3.25: Sơ đồ hoạt động của usecse Thống kê số lượng vé 53](#_Toc179138772)

[Hình 3.26: Lược đồ cơ sở dữ liệu quan hệ của hệ thống đặt vé xe bus 53](#_Toc179138773)

[Hình 3.27: Kiến trúc của hệ thống đặt vé xe OU Bus 55](#_Toc179138774)

[Hình 3.28: Giao diện hiển thị danh sách chuyến xe với sinh viên 57](#_Toc179138775)

[Hình 3.29: Giao diện hiển thị danh sách chuyến xe với nhân viên 57](#_Toc179138776)

[Hình 3.30: Giao diện chi tiết thông tin của chuyến xe và các chỗ ngồi 58](#_Toc179138777)

[Hình 3.31: Giao diện hiển thị vé chưa thanh toán trong Lịch sử đặt vé 58](#_Toc179138778)

[Hình 3.32: Giao diện hiển thị các phương thức thanh toán khi mua vé 59](#_Toc179138779)

[Hình 3.33: Giao diện các danh sách các gói Combo 59](#_Toc179138780)

[Hình 3.34: Giao diện chọn phương thức thanh toán 60](#_Toc179138781)

[Hình 3.35: Giao diện hiển thị thông tin Combo được đăng ký bởi sinh viên 60](#_Toc179138782)

[Hình 3.36: Giao diện Lịch sử đặt vé 61](#_Toc179138783)

[Hình 3.37: Giao diện nhập đánh giá cho vé xe 61](#_Toc179138784)

[Hình 3.38: Giao diện thể hiện nội dung đánh giá của sinh viên và phản hồi của nhân viên 62](#_Toc179138785)

[Hình 3.39: Giao diện lịch sử đặt vé 62](#_Toc179138786)

[Hình 3.40: Giao diện thông tin của người dùng là sinh viên 63](#_Toc179138787)

[Hình 3.41: Giao diện thông tin của người dùng là nhân viên 63](#_Toc179138788)

[Hình 3.42: Giao diện thay đổi avatar. 64](#_Toc179138789)

[Hình 3.43: Giao diện danh sách tuyến xe 64](#_Toc179138790)

[Hình 3.44: Giao diện thông tin chi tiết tuyến xe 65](#_Toc179138791)

[Hình 3.45: Giao diện chỉnh sửa thông tin tuyến xe 65](#_Toc179138792)

[Hình 3.46: Giao diện danh sách các đánh giá của sinh viên 66](#_Toc179138793)

[Hình 3.47: Giao diện chi tiết đánh giá khi nhân viên chưa phản hồi 66](#_Toc179138794)

[Hình 3.48: Giao diện chi tiết đánh giá khi nhân viên đã phản hồi 67](#_Toc179138795)

[Hình 3.49: Giao diện thống kê 67](#_Toc179138796)

DANH MỤC BẢNG

[Bảng 3.1: Đặc tả usecase Đặt vé trực tuyến 30](#_Toc179138797)

[Bảng 3.2: Đặc tả chức năng đăng ký Combo 31](#_Toc179138798)

[Bảng 3.3: Đặc tả usecase Thêm đánh giá 32](#_Toc179138799)

[Bảng 3.4: Đặc tả usecase Xem đánh giá 33](#_Toc179138800)

[Bảng 3.5: Đặc tả usecase Xem đánh giá 33](#_Toc179138801)

[Bảng 3.6: Đặc tả chức năng Tìm kiếm theo tuyến xe 34](#_Toc179138802)

[Bảng 3.7: Đặc tả chức năng Tìm kiếm theo ngày khởi hành 34](#_Toc179138803)

[Bảng 3.8: Đặc tả chức năng Xem danh sách chuyến xe 36](#_Toc179138804)

[Bảng 3.9: Đặc tả chức năng Xem chi tiết chuyến xe 36](#_Toc179138805)

[Bảng 3.10: Đặc tả chức năng Xem lịch sử đặt vé 37](#_Toc179138806)

[Bảng 3.11: Đặc tả usecase chức năng Chỉnh sửa thông tin 38](#_Toc179138807)

[Bảng 3.12: Đặc tả usecase chức năng Đổi mật khẩu 39](#_Toc179138808)

[Bảng 3.13: Đặc tả usecase chức năng Phản hồi đánh giá 40](#_Toc179138809)

[Bảng 3.14: Đặc tả usecase chức năng Thêm chuyến xe 41](#_Toc179138810)

[Bảng 3.15: Đặc tả chức năng Thống kê số lượng vé 43](#_Toc179138811)

[Bảng 3.16: Đặc tả chức năng Thống kê đánh giá 43](#_Toc179138812)

MỞ ĐẦU

Trong bối cảnh công nghệ số đang phát triển với tốc độ vượt bậc, việc số hóa các dịch vụ đã trở thành xu hướng tất yếu của xã hội. Trường Đại học Mở Thành phố Hồ Chí Minh với sứ mệnh mang lại trải nghiệm học tập tốt nhất cho sinh viên, đã nhận thấy nhu cầu ngày càng cao về việc sử dụng các dịch vụ hỗ trợ như phương tiện di chuyển. Điều này đặc biệt rõ nét khi xét đến việc cơ sở 3 của trường hiện nằm ở vị trí khá xa so với trung tâm thành phố khiến cho việc di chuyển của sinh viên gặp nhiều khó khăn, nhất là trong những khung giờ cao điểm.

Nắm bắt được nhu cầu này, dự án ‘Phát triển hệ thống đặt vé xe OU BUS sử dụng Django và ReactJS’ đã được đề xuất. Hệ thống này không chỉ giúp sinh viên dễ dàng đặt vé xe bus mà còn mang lại sự tiện lợi trong việc quản lý các chuyến đi của họ. Bên cạnh đó, hệ thống sẽ cung cấp một giải pháp hiện đại, thân thiện với người dùng, giúp tối ưu hóa quy trình từ việc chọn tuyến xe, đặt ghế cho đến thanh toán trực tuyến. Việc triển khai hệ thống đặt vé này cũng giúp cho nhà trường có thể quản lý tốt hơn dịch vụ vận tải, đồng thời nâng cao chất lượng dịch vụ dành cho sinh viên, tạo nên một môi trường học tập chuyên nghiệp và hiệu quả.

# TỔNG QUAN ĐỀ TÀI

## Giới thiệu đề tài

Trường Đại học Mở Thành phố Hồ Chí Minh là trường đại học đa ngành trực thuộc Bộ Giáo dục và Đào tạo, hiện nay, trường có nhiều cơ sở tại các Quận 1, Quận 3, Huyện Nhà Bè …Đa số các ngành thuộc hệ đào tạo đại trà sẽ được học tại Quận Gò Vấp (cơ sở cũ). Tuy nhiên vì lý do khách quan, năm 2023, Trường đã đột ngột cho di chuyển cơ sở tại Gò Vấp về Nhà Bè. Huyện Nhà Bè là một địa điểm thuộc vị trí ngoại thành của TP. HCM nên sẽ cách xa nơi ở của nhiều sinh viên hiện tại, gây trở ngại khó khăn trong việc đi học của sinh viên. Sau khi lắng nghe được những khó khăn của sinh viên, nhà trường có đưa ra một số phương pháp hỗ trợ. Có thể kể đến, trường đã hỗ trợ sinh viên có nguyện vọng ở trọ ở những nơi gần trường. Bên cạnh đó, dịch vụ xe buýt OU Bus cũng được đề xuất dành cho các bạn sinh viên muốn di chuyển bằng xe buýt của trường từ các quận nội thành về huyện Nhà Bè mà không cần phải thay đổi nơi ở hiện tại và đề xuất này đã nhận được sự ủng hộ của rất nhiều sinh viên.

Tuy nhiên, đây là lần đầu tiên của trường tổ chức dịch vụ OU Bus nên sẽ không tránh khỏi những sai sót. Hệ thống đặt vé xe buýt OU Bus ban đầu của trường hiện tại chỉ phục vụ được các chức năng cơ bản, cho nên việc phát triển hệ thống trở nên chỉn chu, hỗ trợ sinh viên có trải nghiệm tốt hơn trong khi sử dụng dịch vụ tại trường là vô cùng cần thiết. Đó là lý do đề tài “Xây dựng hệ thống website đặt vé OU Bus” được đưa ra nhằm cải tiến hệ thống của nhà trường hiện đang được vận hành lúc bấy giờ.

## Lý do chọn đề tài

Dịch vụ OU Bus là dịch vụ còn mới mẻ với trường Đại học Mở TP. HCM vì đây là lần đầu tiên trường tổ chức loại dịch vụ này. Trên thực tế, các sinh viên hiện đang đăng ký vé xe trên website của trường, các chức năng chỉ dừng ở mức cơ bản. Trải nghiệm đặt vé của sinh viên chưa đạt được sự hài lòng của đại đa số người dùng.

Đối với nhân viên của trường, việc nắm bắt được những đánh giá của sinh viên khi sử dụng dịch vụ là một việc vô cùng quan trọng trong công tác hoàn thiện dịch vụ này. Tuy nhiên hiện nay, họ rất chưa thể nhận được những đánh giá của sinh viên về trải nghiệm sử dụng dịch vụ khi hệ thống đăng ký OU Bus hiện tại chưa cho phép sinh viên bình luận, hay đánh giá chuyến đi của họ. Bên cạnh đó, họ cũng gặp trở ngại trong việc quản lý vé bán ra và số lượng sinh viên trên mỗi chuyến xe để điều phối các chuyến xe sao cho hợp lý nhất.

Từ những thực tế trên và sự hiểu biết của bản thân, tôi đã lên ý tưởng và nghiên cứu phát triển đề tài: “Phát triển hệ thống đặt vé xe OU Bus sử dụng Django và ReactJS” để giúp cho sinh viên của trường có thể trải nghiệm dịch vụ OU Bus một cách tốt nhất. Từ đó góp phần giúp cho nhà trường có thể cải thiện được khó khăn khi gặp phải di chuyển vị trí cơ sở.

## Mục tiêu và phạm vi đề tài

Mục đích của đề tài “Phát triển hệ thống đặt vé xe OU Bus sử dụng Django và ReactJS” là tạo ra một nền tảng trực tuyến giúp sinh viên có thể đặt vé xe buýt thuộc hệ thống dịch vụ OU Bus do nhà trường phụ trách một cách tiện lợi và nhanh chóng.

Để đạt được mục đích đó, hệ thống cần đạt được những mục tiêu như sau:

Chức năng đăng ký: Sinh viên cần phải đăng ký khi chưa có tài khoản tại hệ thống này. Thông tin MSSV phải đúng thì tài khoản mới được khởi tạo.

Chức năng đăng nhập: Sinh viên cần phải đăng nhập để có thể truy cập vào hệ thống. Sinh viên chỉ có thể đăng nhập khi đã có tài khoản.

Xem danh sách chuyến xe: Sinh viên xem danh sách các chuyến xe, có thể lọc theo bến xe, ngày khởi hành của chuyến xe.

Chức năng đặt vé xe: Sinh viên có thể đặt vé xe riêng lẻ hoặc có thể đăng ký các gói combo. Hệ thống sẽ thông báo đến email của sinh viên khi gói combo gần hết hạn hoặc đã sử dụng gần hết vé. Gói combo sẽ tự động hủy khi nó hết hạn hoặc sinh viên đã sử dụng hết vé trong gói combo đó. Vé xe bus cần đặt trước 2 ngày kể từ ngày chuyến xe đó khởi hành. Sinh viên cần chọn chuyến xe, chọn chỗ ngồi và phải thanh toán online (Zalopay hoặc VNPAY) để hoàn tất thủ tục đặt vé.

Chức năng đánh giá chuyến xe**:** Sau mỗi lần hoàn thành chuyến đi, sinh viên có thể đánh giá chuyến đi đó thông qua bình luận và đánh giá theo sao (1-5 sao). Mỗi sinh viên chỉ có một lượt đánh giá cho mỗi chuyến xe. Nhân viên nhà trường sẽ phản hồi nếu sinh viên gặp vấn đề trong quá trình sử dụng dịch vụ.

Quản lý chuyến xe (thêm/sửa/xóa): Dựa vào số lượng vé đã đặt, nhân viên sẽ kiểm soát số lượng các chuyến xe sao cho phù hợp. Nhân viên sẽ cũng là người gán và thay đổi tài xế trên các chuyến xe.

Thống kê báo cáo: Chỉ nhân viên nhà trường mới có thể xem được các thống kê về lượt đánh giá của sinh viên, về số lượng vé xe bán ra theo tuần, theo tháng, theo năm.

Chat hỗ trợ: Sinh viên và nhân viên có thể chat với nhau nhằm mang lại trải nghiệm dịch vụ tốt nhất cho sinh viên.

Đối tượng sử dụng hệ thống: Sinh viên Trường Đại học Mở Thành phố Hồ Chí Minh và nhân viên của nhà trường được phân công quản lý hệ thống.

Thời gian thực hiện đề tài: 10 tuần (05/08/2024 – 16/10/2024).

Công nghệ được sử dụng để phát triển hệ thống: Django (Backend) và ReactJS (Frontend).

## Phương pháp nghiên cứu

Đề tài ‘Phát triển hệ thống đặt vé xe OU Bus sử dụng Django và ReactJS’ sử dụng phương pháp “Khảo sát nghiệp vụ” để tìm hiểu nhu cầu sử dụng dịch vụ OU Bus của sinh viên và nghiên cứu về các tài liệu công nghệ liên quan. Công nghệ được đề xuất sử dụng cho đề tài là Django, một framework của ngôn ngữ Python, chuyên xử lý các logic bên phía cơ sở dữ liệu và xây dựng các Restful API. Nhằm mang lại trải nghiệm của người dùng một cách tốt nhất, đề tài sẽ sử dụng công nghệ ReactJS, một trong những thư viện phổ biến trong việc phát triển các trang web động.

## Bố cục báo cáo

Chương 1: Tổng quan đề tài. Phần này giới thiệu về nội dung của đề tài, về mục tiêu và lý do thực hiện đề tài. Đề tài tập trung vào việc xây dựng một hệ thống đặt vé xe trực tuyến danh cho sinh viên trường Đại học, nhằm mang lại tiện lợi và nhanh chóng trong quá trình đặt vé, quản lý chuyến đi. Ngoài ra, phần này cũng trình bày về phạm vi nghiên cứu của đề tài.

Chương 2: Cơ sở lý thuyết. Trong chương này, đề tài trình bày các kiến thức nền tảng và công nghệ được sử dụng trong dự án. Bao gồm các khái niệm cơ bản về công nghệ phát triển web như Django, cụ thể là Django REST framework cho phần xử lý phía máy chủ (backend) và ReactJS cho phần giao diện người dùng (frontend). Cùng với đó là các lý thuyết cơ bản về các công cụ tích hợp hỗ trợ trong hệ thống.

Chương 3: Hệ thống đặt vé xe OU Bus. Phần này mô tả chi tiết quá trình phát triển của đồ án, từ giai đoạn phân tích hệ thống, thiết kế hệ thống, triển khai các chức năng chính của hệ thống, kiến trúc của hệ thống, đến những kết quả mà đề tài đã đạt được sau 10 tuần thực hiện.

Chương 4: Kết luận và hướng phát triển. Chương này tổng kết lại những kết quả đạt được của đề tài, nhấn mạnh vào tính khả thi và hiệu quả của hệ thống đặt vé xe OU Bus. Đồng thời, đề tài cũng đề xuất các hướng phát triển tiếp theo của hệ thống nhằm nâng cao trải nghiệm của người dùng và mở rộng tính năng của hệ thống trong tương lai.

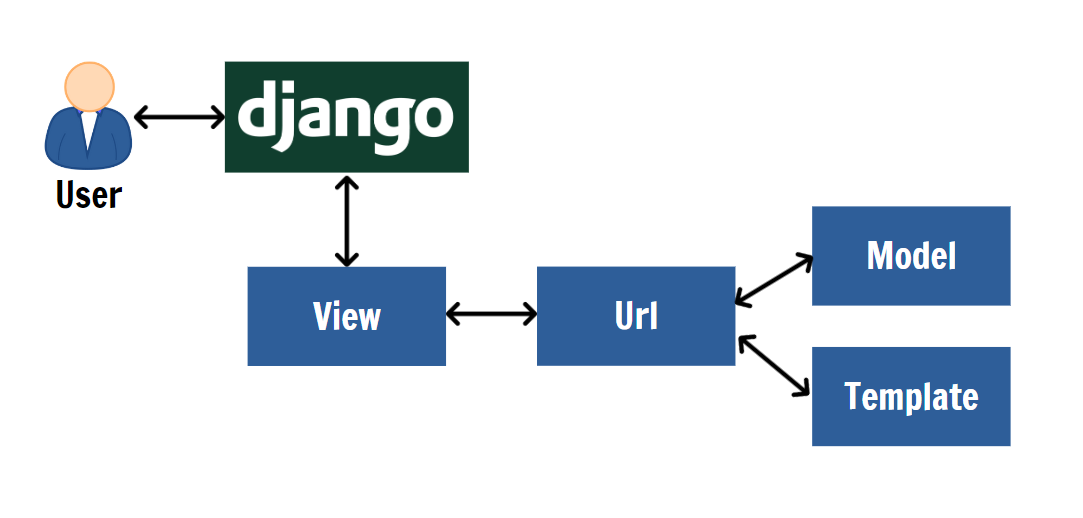
# CƠ SỞ LÝ THUYẾT

## Django

### Giới thiệu Django

Django là một framework phát triển web mã nguồn mở cấp cao, được viết bằng ngôn ngữ lập trình Python, giúp đơn giản hóa và tăng tốc quá trình xây dựng ứng dụng web.

Django tuân theo mô hình MVT (Model-View-Template), tương tự mô hình MVC (Model-View-Controller) truyền thống nhưng có một số khác biệt trong cách tổ chức mã nguồn, giúp tách biệt logic xử lý dữ liệu, giao diện và các hoạt động tương tác với người dùng.



Hình 2.1: Mô hình MVT (Model-View-Template)

Model chịu trách nhiệm quản lý và tương tác với cơ sở dữ liệu. Nó định nghĩa cấu trúc dữ liệu trong hệ thống, các bảng và quan hệ giữa các bảng trong cơ sở dữ liệu.

View là nơi xử lý các logic nghiệp vụ thông qua các url được định nghĩa. Khi nhận được request từ phía client, View sẽ có trách nhiệm thực hiện yêu cầu bằng cách tương tác với Model để lấy dữ liệu và chuyển dữ liệu này về Template để hiển thị [1].

Template là nơi chịu trách nhiệm hiển thị giao diện cho người dùng. Template của Django thường là các tệp HTML, trong đó có thể chèn các biến và logic đơn giản như vòng lặp, điều kiện.

### Ưu điểm của Django

Django cung cấp nhiều tính năng mặc định, giảm thiểu thời gian phải tự viết code từ đầu. Django tuân theo nguyên tắc DRY (Don’t repeat yourself – Không lặp lại) cho phép lập trình viên có thể tái sử dụng các đoạn code hiện có nhằm giảm số lượng code [2] và có tích hợp sẵn ORM (Object-Relational Mapping) giúp dễ dàng quản lý và thao tác với cơ sở dữ liệu mà không cần viết câu lệnh SQL phức tạp.

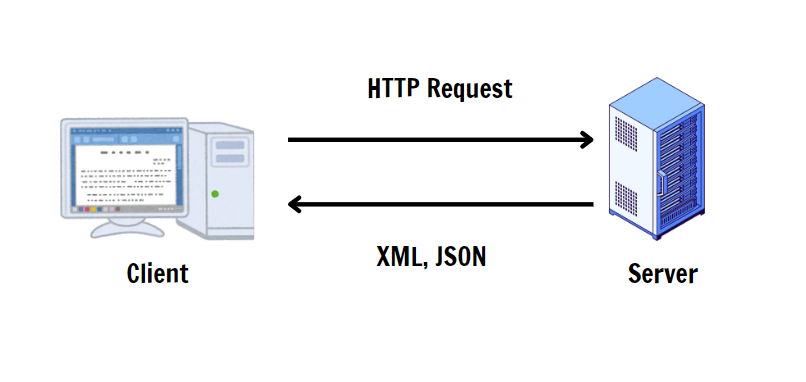
Django được xem là một trong những framework có hệ thống bảo mật tốt nhất hiện nay. Nó được thiết kế với các tính năng bảo mật tích hợp như ngăn chặn SQL Injection, XSS (Cross-Site Scripting), CSRF (Cross-Site Request Forgery).

Django sở hữu nhiều tính năng bổ sung và có thể dễ dàng mở rộng. Đồng thời, Django cũng hỗ trợ người dùng trên đa nền tảng và có thể hoạt động được với hầu hết các database chính. Chính những yếu tố này, framework này của Python có thể giúp ta giải quyết các dự án ở bất kỳ quy mô nào, từ những trang web nhỏ cho đến những hệ thống web high-load [2].

Bên cạnh đó, Django sở hữu một cộng đồng người dùng lớn và có nhiều tài liệu phong phú giúp người dùng có thể dễ dàng học hỏi và giải quyết vấn đề trong quá trình sử dụng framework.

### Django REST framework

REST (Representational State Transfer - Chuyển đổi trạng thái đại diện) là một dạng kiến trúc phần mềm hoạt động dành cho các dịch vụ web, cho phép các hệ thống giao tiếp với nhau dựa trên giao thức HTTP.



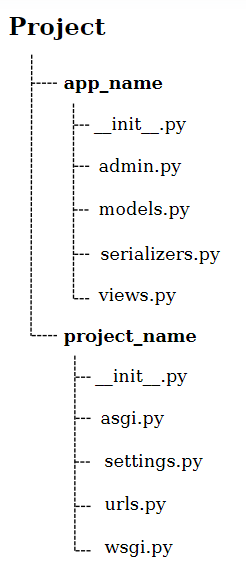
Hình 2.2: Mô hình RESTful Web Services

Django REST framework (DRF) là một framework mạnh mẽ và linh hoạt trong việc phát triển các ứng dụng web API được phát triển trên nền tảng của Django. Được thiết kế với mục tiêu giúp đơn giản hóa việc xây dựng API, DRF cung cấp các thành phần như serializers hỗ trợ chuyển đổi dữ liệu giữa Python và các định dạng dữ liệu như JSON, authentication để xác thực người dùng và các class-based views để xử lý các request một cách dễ dàng.

Bên cạnh việc giúp xây dựng các API cơ bản, Django REST framework còn hỗ trợ các chức năng khác như xử lý lỗi, quản lý permissions cho phép kiểm soát quyền truy cập tới từng API dựa trên vai trò của người dùng và tự động tạo các tài liệu API. Điều này giúp các lập trình viên có thể tập trung vào logic của ứng dụng mà không cần lo về kỹ thuật phức tạp trong việc xây dựng API.

Với những công cụ và các tính năng tiêu biểu trên, DRF đã trở thành một lựa chọn phổ biến cho việc phát triển các ứng dụng web API trong cộng đồng phát triển ứng dụng Python.

### Cấu trúc project Django REST API



Hình 2.3: Cấu trúc chung của một project DRF

Ở Model (models.py), đây là nơi ta định nghĩa các lớp tương ứng với các bảng trong cơ sở dữ liệu. Ở đoạn code sau, ta tạo một table ‘Bus’ vào cơ sở dữ liệu.

from django.db import models

class Bus(models.Model):

name = models.CharField(max\_length=100)

license\_plate = models.CharField(max\_lenght=20, unique=True)

Ta cần phải nhập python manage.py makemigrations để cập nhật những thay đổi trong models.py. Những thay đổi này sẽ được lưu thành các tệp migration trong thư mục migrations. Để áp dụng các thay đổi đó lên cơ sở dữ liệu ta cần: python manage.py migrate.

Serializer là một thành phần quan trọng trong DRF project, giúp chuyển đổi dữ liệu từ các đối tượng Python (ví dụ như model) sang các định dạng như JSON, và ngược lại.

BusSerializer bên dưới giúp chuyển đổi dữ liệu của model Bus thành định dạng JSON

from rest\_framework import serializers

from oubus.models import Bus

class BusSerializer(serializers.ModelSerializer):

class Meta:

model = Bus

fields = ‘\_\_all\_\_’

Ở views.py, đây là nơi định nghĩa các View hoặc Viewsets của API. Trong DRF, ta có thể sử dụng các class-based views hoặc viewset để xử lý logic của API. BusViewSet bên dưới được xây dựng để xử lý các request liên quan đến model Bus

from rest\_framework import viewsets, generics

from oubus.models import Bus

from oubus import serializers

class BusViewSet(viewsets.ViewSet, generics.ListCreateAPIView, generics.RetrieveUpdateDestroyAPIView):

queryset = Bus.objects.all()

serializer\_class = serializers.BusSerializer

def get\_permissions(self):

if self.request.method in ['POST', 'PATCH', 'DELETE']:

return [perms.IsStaff()]

return [permissions.IsAuthenticated()]

Đối với urls.py, tệp này sẽ định nghĩa các đường dẫn URL cho API của bạn. DRF cung cấp cơ chế routing mạnh mẽ thông qua routers.

from django.urls import path, include

from rest\_framework import routers

from oubus import views

r = routes.DefaultRouter()

r.register(‘buses’, views.BusViewSet, basename=’buses’)

urlpatterns = [

path(‘’, include(r.urls)),

]

Bên cạnh đó còn có tệp permissions.py, tệp này được sử dụng để định nghĩa các lớp phân quyền cho API. DRF cũng cung cấp nhiều tùy chọn phân quyền như dựa trên người dùng, nhóm, hoặc custom permission.

Sau khi định nghĩa các thành phần trên, bạn có thể khởi chạy server bằng lệnh: python manage.py runserver

## ReactJS

### Giới thiệu ReactJS

ReactJS hay React là một thư viện Javarscript mã nguồn mở được phát triển bởi Facebook, được dùng để xây dựng giao diện người dùng (UI) cho các ứng dụng web. ReactJS nổi bật với khả năng tạo ra các ứng dụng web đơn trang, nơi giao diện có thể thay đổi mà không cần phải tải lại toàn bộ trang [3].

### Một số đặc điểm của React

Kiến trúc component-based: React cho phép chia nhỏ giao diện người dùng thành các các component độc lập, và có thể tái sử dụng. Mỗi component hoạt động như một khối giao diện nhỏ, đều có riêng cho mình state and props. State là dữ liệu hoặc trạng thái nội bộ của một component, khi state thay đổi thì React sẽ tự động cập nhật lại component. Props được hiểu là dữ liệu truyền từ component cha sang component con. Props không thể thay đổi từ bên trong component con.

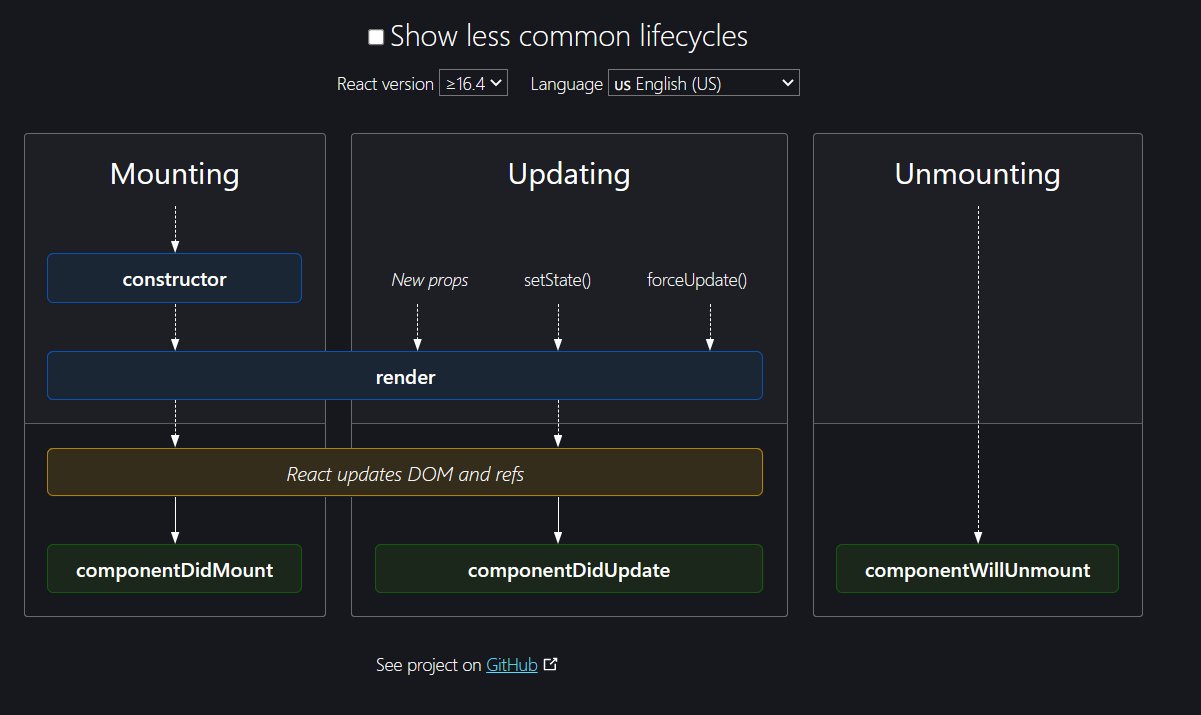
JSX (Javascript Syntax Extension): JSX là một cú pháp mở rộng của Javascript, cho phép những nhà phát triển có thể viết mã HTML trong cùng tệp với mã Javascript. JSX giúp việc viết mã dễ hiểu và tương tác hơn khi kết hợp giao diện và logic.

Virtual DOM: React sử dụng một kỹ thuật gọi là Virtual DOM, giúp tăng tốc độ và hiệu suất. Khi xuất hiện thay đổi trên giao diện, React không cập nhật trực tiếp vào DOM thật mà sẽ tạo một cây DOM ảo, chỉ cập nhật những phần đã thay đổi trên DOM thật.

One-way Data Binding hay còn được hiểu là ràng buộc dữ liệu một chiều. Dữ liệu trong React chỉ di chuyển theo 1 hướng từ component cha xuống component con. Ràng buộc này giúp dễ dàng theo dõi được luồng dữ liệu và kiểm soát các thay đổi của ứng dụng.

### Vòng đời ReactJS

Mỗi React Component đều có một vòng đời (lifestyle) hoạt động. Vòng đời của component được chia thành ba giai đoạn: Mounting, Updating, Unmounting.



Hình 2.4: Vòng đời hoạt động của React Component (nguồn: https://projects.wojtekmaj.pl/react-lifecycle-methods-diagram/)

Giai đoạn Mounting còn được hiểu là giai đoạn khởi tạo, đây là giai đoạn khi component được khởi tạo và đưa vào DOM. Phương thức quan trọng trong giai đoạn này bao gồm contructor(props). Đây là phương thức khởi tạo được gọi đầu tiên khi component được khởi tạo. Tại đây ta có thể gán các giá trị ban đầu cho state và các biến nếu muốn. Tiếp theo là phương thức render (), đây là phương thức bắt buộc trong mỗi component và nó trả về khối JSX để đổ vào DOM. Hàm render () luôn được gọi trong quá trình khởi tạo và bất cứ khi nào có sự thay đổi trong state và props. Cuối cùng, componentDidMount () được chạy tự động một lần gọi sau khi component được render xong, nhưng nếu về sau props, state có thay đổi thì hàm này sẽ không chạy lại. Hàm componentDidMount () cũng là nơi thích hợp để thực hiện các tác vụ liên quan đến thao tác DOM hoặc gọi API.

Giai đoạn Updating xảy ra khi props hoặc state của component thay đổi, và component cần phải render lại để phản ánh các thay đổi đó. Ở phương thức componentDidUpdate (prevDrops, prevState, snapshot), phương thức được gọi ngay sau khi component được mount. Đây là nơi ta có thể thực hiện các tác vụ sau khi DOM đã thay đổi hoặc gửi yêu cầu cập nhật dữ liệu khi các props thay đổi.

Giai đoạn Unmounting là giai đoạn khi component bị loại bỏ khỏi DOM. Phương thức chính trong giai đoạn này là componentWillUnmount (), phương thức này sẽ được gọi khi một component bị unmount và destroy. Thực hiện mọi thao tác cleanup cần thiết trong phương thức này như hủy bỏ các sự kiện, dừng các interval/timers hoặc hủy các kết nối API.

## Các công cụ hỗ trợ

### Cloudinary

Cloudinary là một nền tảng quản lý và tối ưu hóa các nội dung đa phương tiện như hình ảnh, video được lưu trữ trên điện toán đám mây. Nó cung cấp cho người sử dụng các công cụ mạnh mẽ để quản lý, lưu trữ, phân phối các dữ liệu đa phương tiện một cách hiệu quả, giúp người sử dụng dễ dàng tích hợp và sử dụng trong các ứng dụng web. Cloudinary cho phép người dùng có thể tải lên và lưu trữ các dữ liệu là hình ảnh hoặc video và các file đa phương tiện khác trên hạ tầng đám mây. Những dữ liệu đó được lưu trữ an toàn và có thể được truy cập từ bất kỳ nơi nào thông qua API. Các lập trình viên cũng có thể quản lý được tất cả các file đa phương tiện đã được gửi lên Cloudinary bao gồm việc chỉnh sửa hoặc xóa, hoặc là phân loại các tệp một cách thuận tiện.

Việc sử dụng Cloudinary giúp các lập trình viên có thể giảm tải cho server, thay vì phải lưu trữ các tệp đa phương tiện tại server thì Cloudinary có thể xử lý mọi thứ liên quan đến lưu trữ, tối ưu hóa và cải thiện hiệu suất trên đám mây. Đồng thời nó cũng dễ dàng để tích hợp vào hệ thống thông qua các tài liệu hướng dẫn chi tiết của Cloudinary.

### MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, được sử dụng rộng rãi trong việc tạo, lưu trữ, quản lý và truy vấn dữ liệu trên các ứng dụng web và phần mềm. MySQL sử dụng SQL (Structured Query Language), ngôn ngữ tiêu chuẩn để tương tác với cơ sở dữ liệu, nhằm quản lý các hoạt động bao gồm tạo, xóa, sửa đổi và truy xuất dữ liệu.

MySQL có khả năng xử lý khối lượng lớn dữ liệu với tốc độ cao, ngay cả khi hệ thống có nhiều yêu cầu truy vấn đồng thời. Đây là một trong những lý do mà MySQL được lựa chọn cho các ứng dụng web có quy mô lớn như Facebook, hay Youtube.

Bên cạnh đó, MySQL đã được phát triển và tối ưu hóa qua nhiều năm, nên có thể đảm bảo được độ tin cậy và tính ổn định cao cho các ứng dụng sử dụng nó. MySQL còn cung cấp các công cụ hỗ trợ cho sao lưu và phục hồi dữ liệu một cách an toàn, đảm bảo tính toàn vẹn của dữ liệu trong các trường hợp khẩn cấp.

Lợi ích mà MySQL mang lại khi sử dụng là đây là một phiên bản mã nguồn mở miễn phí, dễ dàng tiếp cận các sinh viên, học sinh khi có nhu cầu lưu trữ dữ liệu cho các dự án nhỏ của bản thân, đồng thời vẫn có các phiên bản trả phí cung cấp nhiều tính năng nâng cao hơn cho các doanh nghiệp lớn. Những người dùng mới của MySQL cũng có thể dễ dàng tiếp cận công cụ bởi SQL là một ngôn ngữ dễ học, MySQL cũng có rất nhiều tài liệu, diễn đàn và cộng đồng luôn sẵn sàng hỗ trợ khi người dùng gặp khó khăn.

### PythonAnywhere

PythonAnywhere là một nền tảng điện toán đám mây được thiết kế để hỗ trợ cho việc triển khai các ứng dụng web Python như Django, Flask một cách dễ dang mà không cần cấu hình server. Người dùng chỉ cần upload lên mã nguồn, cấu hình đơn giản, và ứng dụng sẽ sẵn sàng chạy.

Bên cạnh đó, PythonAnywhere hỗ trợ cho việc tích hợp các cơ sở dữ liệu như MySQL, PostgreSQL, và SQLite, giúp người sử dụng có thể dễ dàng lưu trữ và quản lý dữ liệu trong ứng dụng của bản thân.

PythonAnywhere sở hữu gói sử dụng miễn phí giúp các sinh viên tiếp cận với các dự án nhỏ hoặc các gói sử dụng trả phí với nhiều tính năng nâng cao hơn dành cho các dự án lớn hoặc chuyên nghiệp.

# HỆ THỐNG ĐẶT VÉ XE OUBUS

## Giới thiệu bài toán

Hệ thống “Đặt vé xe OU Bus” được phát triển nhằm hỗ trợ sinh viên Trường Đại học Mở Thành phố Hồ Chí Minh trong việc đăng ký và sử dụng dịch vụ xe buýt của trường. Sinh viên cần đăng nhập vào hệ thống với tài khoản hợp lệ, hoặc nếu chưa có tài khoản thì sinh viên phải đăng ký bằng mã số sinh viên do nhà trường cấp.

Vé xe có thể được mua lẻ hoặc theo gói combo: Combo 15 vé và Combo 30 vé, với mức giá ưu đãi hơn so với mua lẻ từng vé. Mỗi combo bao gồm số lượng vé nhất định và có hạn sử dụng. Khi đăng ký combo, sinh viên sẽ tiến hành thanh toán, sau đó có thể đặt vé mà không cần thanh toán thêm. Hệ thống sẽ gửi thông báo qua email khi vé trong gói combo sinh viên đăng ký hết. Mỗi sinh viên chỉ được sử dụng một combo tại một thời điểm, và combo sẽ tự động hủy khi hết hạn hoặc khi sinh viên đã sử dụng hết số vé trong gói combo.

Trong quá trình đặt vé, sinh viên chọn chuyến xe, chỗ ngồi và thanh toán trực tuyến (qua ZaloPay hoặc VNPay) để vé có hiệu lực. Sinh viên cũng có thể xem danh sách chuyến xe, tìm kiếm chuyến xe theo bến xe hay ngày khởi hành của chuyến xe để tìm kiếm chuyến xe phù hợp. Sau mỗi chuyến đi được hoàn thành, sinh viên có thể để lại đánh giá cho chuyến xe đó nếu sinh viên có trải nghiệm không tốt về chuyến xe đó.

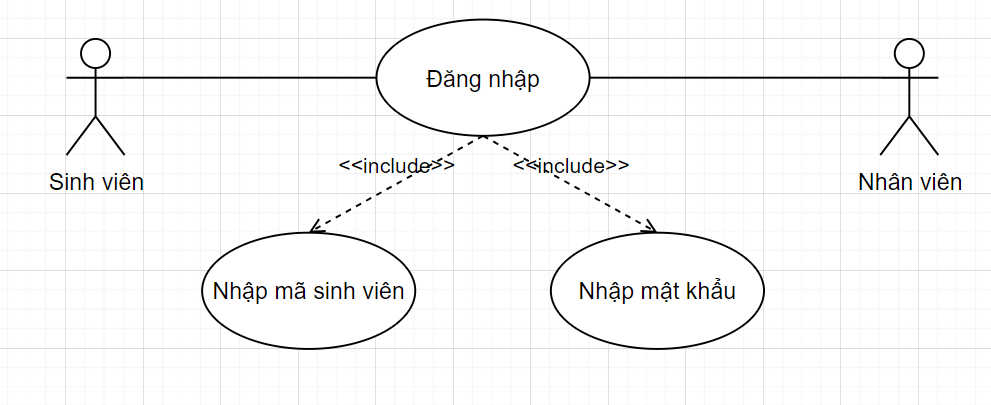
Nhân viên nhà trường sẽ tổng hợp và sẽ phản hồi lại đánh giá của sinh viên nếu nhận được các đánh giá không tốt. Dựa vào số lượng vé, nhân viên có thể thêm/sửa/xóa các chuyến xe sao cho phù hợp. Nhân viên cũng sẽ là người gán hoặc thay đổi tài xế cho các chuyến xe. Bên cạnh đó, nhân viên có thể xem được các thống kê báo cáo về số lượng đánh giá của sinh viên theo thời gian nhân viên chọn và thống kê số lượng vé đã được bán ra theo tuần, theo tháng, theo năm được phân theo từng tuyến xe.

Ngoài ra, sinh viên có thể trao đổi với nhân viên qua tính năng chat để giải quyết các vấn đề gặp phải khi sử dụng hệ thống.

## Phân tích hệ thống

### Chức năng Đăng nhập

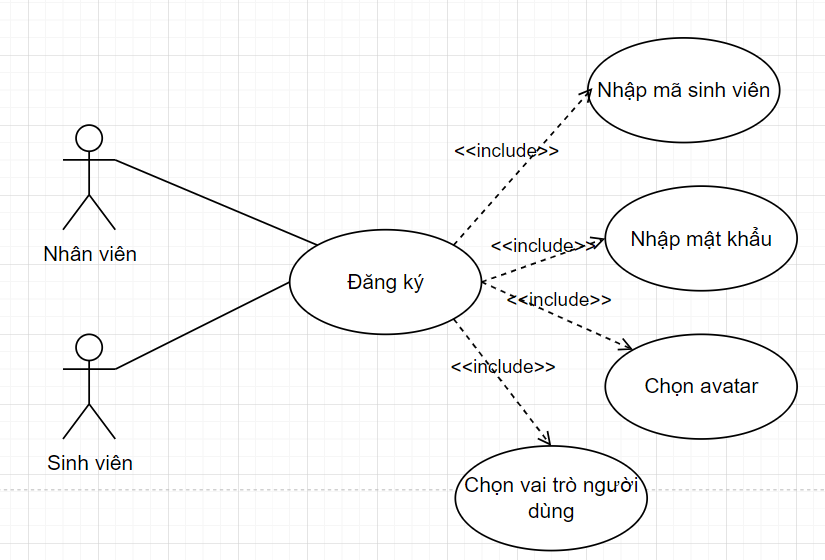
Hệ thống yêu cầu người dùng đăng nhập thông tin tài khoản, hệ thống sẽ kiểm tra tài khoản đó có tồn tại trong hệ thống hay chưa. Nếu đã tồn tại thì hệ thống sẽ hiển thị giao diện làm việc của tác nhân tương ứng. Nếu thông tin không đúng thì cho người dùng đăng nhập lại.



Hình 3.1: Usecase chức năng đăng nhập

### Chức năng Đăng ký

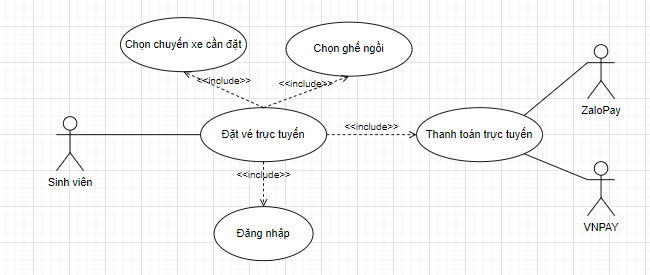
Hệ thống yêu cầu người dùng nhập thông tin tài khoản, hệ thống sẽ kiểm tra tài khoản đó đã tồn tại trên hệ thống chưa dựa trên mã sinh viên hoặc mã nhân viên tùy vào vai trò mà người dùng đăng ký. Nếu chưa có, hệ thống sẽ lưu tài khoản và thông báo tài khoản đăng ký thành công.



Hình 3.2: Usecase chức năng Đăng ký

### Chức năng Đặt vé trực tuyến

Sinh viên chọn chuyến xe muốn đi, chọn ghế ngồi và đặt vé xe. Nếu quyết định thanh toán vé, sinh viên chọn vé và chọn hình thức thanh toán, sinh viên tiến hành thanh toán để hoàn tất quá trình đặt vé.



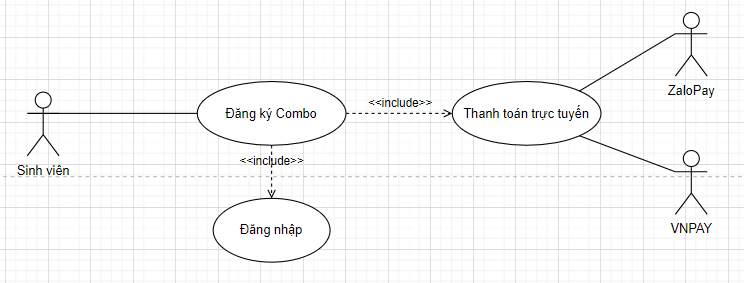
Hình 3.3: Usecase chức năng Đặt vé trực tuyến

Bảng 3.1: Đặc tả usecase Đặt vé trực tuyến

|  |  |
| --- | --- |
| **Tên usecase** | Đặt vé trực tuyến |
| **Mô tả** | Cho phép khách hàng thực hiện đặt vé trực tuyến trên hệ thống |
| **Actor chính** | Sinh viên |
| **Actor phụ** | ZaloPay, VNPAY |
| **Tiền điều kiện** | Đã đăng nhập, đã chọn vé cần đặt |
| **Hậu điều kiện** | Sinh viên cần thanh toán vé trực tuyến |
| **Luồng hoạt động** | 1. Sinh viên chọn vào chuyến xe cần đặt 2. Sinh viên chọn ghế ngồi 3. Sinh viên đặt vé 4. Sinh viên xác nhận thanh toán 5. Sinh viên chọn phương thức thanh toán 6. Hệ thống kiểm tra thông tin thanh toán 7. Hệ thống thông báo đặt vé thành công |
| **Luồng thay thế** | * Ở bước 3, nếu không đồng ý đặt vé thì có thể quay lại trang chủ. * Nếu bước 6, nếu xảy ra lỗi trong lúc thanh toán thì hệ thống sẽ thông báo thanh toán lỗi. |

### Chức năng Đăng ký gói Combo

Sinh viên có thể đăng ký gói Combo bằng cách chọn Đăng ký Combo, sau đó chọn gói Combo theo mong muốn và phải thanh toán trực tuyến thông qua ZaloPay hoặc VNPAY để gói Combo đó bắt đầu có hiệu lực.



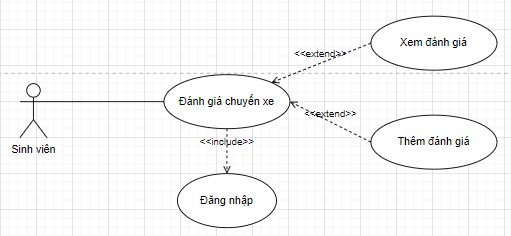
Hình 3.4: Usecase chức năng Đăng ký gói Combo

Bảng 3.2: Đặc tả chức năng đăng ký Combo

|  |  |
| --- | --- |
| **Tên usecase** | Đăng ký Combo |
| **Mô tả** | Cho phép sinh viên đăng ký gói vé Combo |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập |
| **Hậu điều kiện** | Thanh toán trực tuyến |
| **Luồng hoạt động** | Sinh viên chọn chức năng Đăng ký gói Combo  Sinh viên chọn gói vé mong muốn  Hệ thống kiểm tra tính hợp lệ của gói vé  Hệ thống yêu cầu thanh toán trực tuyến  Sinh viên xác nhận thanh toán.  Hệ thống kiểm tra thông tin thanh toán.  Hệ thống thông báo đặt gói Combo thành công. |
| **Luồng thay thế** | * Ở bước 1, nếu sinh viên hiện đã đăng ký gói Combo nào đó vẫn còn hiệu lực thì sẽ hiển thị thông tin của Combo hiện tại của sinh viên. * Ở bước 4, nếu hệ thống phát hiện đánh giá không hợp lệ thì hệ thống sẽ thông báo không thể thêm đánh giá và trở lại màn hình thêm đánh giá ở bước 3. |

### Chức năng Đánh giá chuyến xe

Sinh viên cần phải đăng nhập để thực hiện được chức năng đánh giá và sinh viên cần phải có vé mà chuyến xe đã được hoàn thành thì mới có thế đánh giá cho chuyến xe đó.



Hình 3.5: Usecase chức năng Đánh giá chuyến xe

Bảng 3.3: Đặc tả usecase Thêm đánh giá

|  |  |
| --- | --- |
| **Tên usecase** | Thêm đánh giá |
| **Mô tả** | Cho phép sinh viên đánh giá chuyến xe của mình |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập, đã hoàn thành chuyến xe đó. |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Sinh viên vào phần Lịch sử đặt vé.  Sinh viên chọn vé thuộc chuyến xe cần đánh giá.  Sinh viên thêm đánh giá bao gồm phần rating và phần bình luận.  Hệ thống sẽ kiểm tra tính hợp lệ của đánh giá.  Hệ thống lưu lại đánh giá và thông báo thêm đánh giá thành công. |
| **Luồng thay thế** | * Ở bước 2, nếu sinh viên đã đánh giá rồi thì hệ thống sẽ hiển thị đánh giá của sinh viên. * Ở bước 4, nếu hệ thống phát hiện đánh giá không hợp lệ thì hệ thống sẽ thông báo không thể thêm đánh giá và trở lại màn hình thêm đánh giá ở bước 3. |

Bảng 3.4: Đặc tả usecase Xem đánh giá

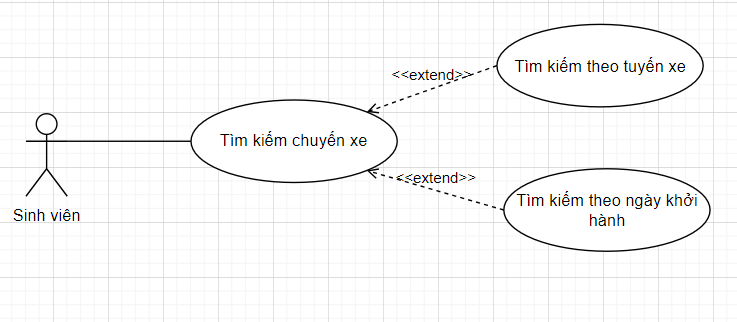
|  |  |
| --- | --- |
| **Tên usecase** | Xóa đánh giá |
| **Mô tả** | Cho phép sinh viên xóa đánh giá chuyến xe. |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập, đã có đánh giá. |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | 1. Sinh viên chọn vào Đánh giá. 2. Hệ thống hiển thị thông tin đánh giá của sinh viên và phản hồi của nhân viên nếu có. |
| **Luồng thay thế** | * Ở bước 2, nếu sinh viên chưa thực hiện đánh giá ở vé xe đó thì hệ thống sẽ hiển thị form để sinh viên nhập thông tin đánh giá |

Bảng 3.5: Đặc tả usecase Xem đánh giá

|  |  |
| --- | --- |
| **Tên usecase** | Xem đánh giá |
| **Mô tả** | Cho phép sinh viên xem đánh giá chuyến xe. |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập, đã có đánh giá. |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | 1. Sinh viên chọn mục Đánh giá 2. Hệ thống sẽ hiển thị danh sách các đánh giá của sinh viên. |

### Chức năng Tìm kiếm chuyến xe

Sinh viên có thể tìm kiếm chuyến xe bằng cách tìm theo tuyến xe hay theo ngày khởi hành.



Hình 3.6: Usecase chức năng Tìm kiếm chuyến xe

Bảng 3.6: Đặc tả chức năng Tìm kiếm theo tuyến xe

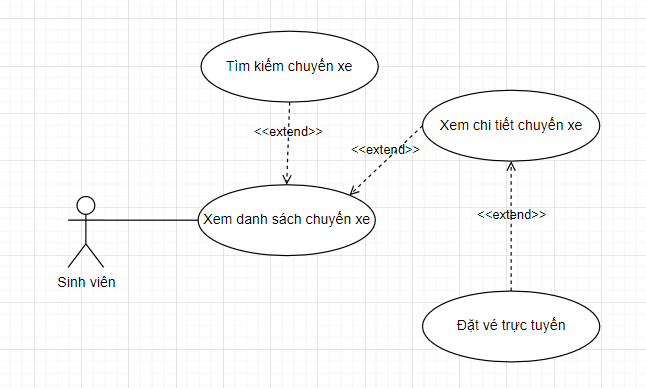
|  |  |
| --- | --- |
| **Tên usecase** | Tìm kiếm theo bến xe |
| **Mô tả** | Cho phép sinh viên có thể lọc danh sách chuyến xe theo tuyến chạy của xe bus |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Sinh viên chọn bến xe đi.  Sinh viên chọn bến xe đến.  Hệ thống sẽ kiếm tra chuyến đi trong CSDL.  Hệ thống sẽ hiển thị danh sách các chuyến đi tương ứng. |
| **Luồng thay thế** | * Ở bước 3, nếu không tồn tại chuyến đi nào trong CSDL thì hệ thống sẽ thông báo không tìm thấy chuyến xe. |

Bảng 3.7: Đặc tả chức năng Tìm kiếm theo ngày khởi hành

|  |  |
| --- | --- |
| **Tên usecase** | Tìm kiếm theo bến xe |
| **Mô tả** | Cho phép sinh viên có thể lọc danh sách chuyến xe theo ngày khởi hành của chuyến xe. |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | 1. Sinh viên chọn ngày khởi hành. 2. Hệ thống sẽ kiểm tra các chuyến đi trong CSDL. 3. Hệ thống hiển thị danh sách chuyến đi có ngày khởi hành tương ứng. |
| **Luồng thay thế** | * Ở bước 2, nếu chuyến đi không có trong CSDL, thì hệ thống sẽ thông báo không tìm thấy chuyến đi. |

### Chức năng Xem danh sách chuyến xe

Sinh viên có thể xem danh sách các chuyến xe trên hệ thống. Nếu sinh viên chọn xem thông tin của một chuyến xe cụ thể nào đó thì hệ thống sẽ hiển thị thông tin chi tiết của chuyến xe đó. Khi xem thông tin chi tiết của chuyến xe, sinh viên có thể tiến hành đặt vé cho chuyến xe đó.



Hình 3.7: Usecase chức năng Xem danh sách chuyến xe

Bảng 3.8: Đặc tả chức năng Xem danh sách chuyến xe

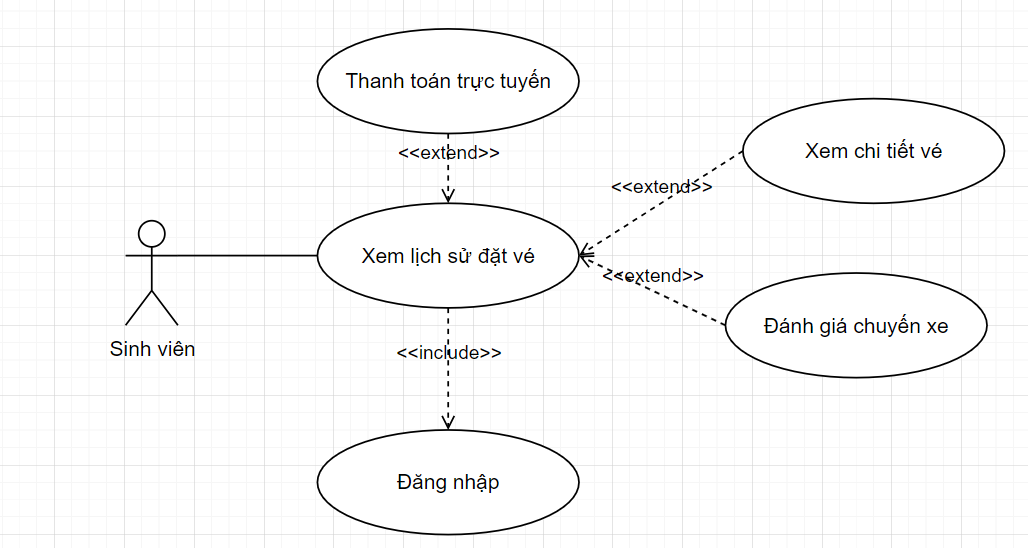
|  |  |
| --- | --- |
| **Tên usecase** | Xem danh sách chuyến xe |
| **Mô tả** | Cho phép sinh viên xem danh sách các chuyến xe đang được mở của hệ thống. |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Sinh viên gửi yêu cầu xem danh sách các chuyến xe  Hệ thống kiểm tra các chuyến xe hợp lệ trong CSDL  Hệ thống hiển thị danh sách chuyến xe. |
| **Luồng thay thế** | * Ở bước 2, nếu không tìm thấy các chuyến xe phù hợp thì hệ thống sẽ thông báo không tìm thấy chuyến xe. |

Bảng 3.9: Đặc tả chức năng Xem chi tiết chuyến xe

|  |  |
| --- | --- |
| **Tên usecase** | Xem chi tiết chuyến xe |
| **Mô tả** | Cho phép sinh viên xem thông tin chi tiết của một chuyến xe |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Đã đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | 1. Sinh viên chọn chuyến xe cần xem thông tin 2. Hệ thống sẽ hiển thị thông tin chi tiết của chuyến xe đó |

### Chức năng Xem lịch sử đặt vé

Sinh viên chọn chức năng Xem lịch sử đặt vé, hệ thống sẽ hiển thị danh sách các hóa đơn ghi nhận đặt vé của sinh viên, sinh viên có thể chọn xem chi tiết hóa đơn, hệ thống sẽ hiển thị thông tin chi tiết của hóa đơn đó. Nếu sinh viên chưa từng đặt vé thì hệ thống sẽ thông báo rằng chưa có hóa đơn nào.



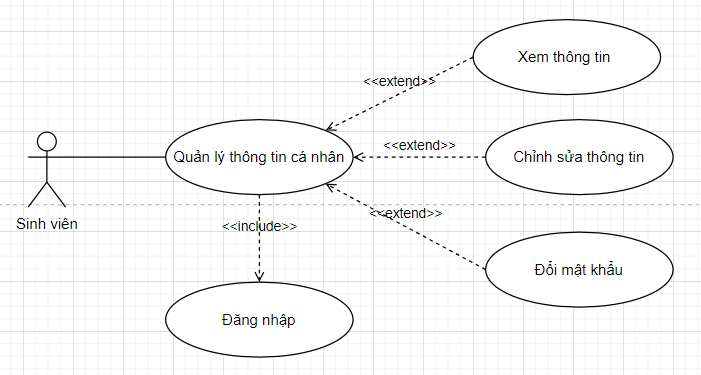
Hình 3.8: Usecase chức năng Xem lịch sử đặt vé

Bảng 3.10: Đặc tả chức năng Xem lịch sử đặt vé

|  |  |
| --- | --- |
| **Tên usecase** | Xem lịch sử đặt vé |
| **Mô tả** | Cho phép sinh viên xem lịch sử đặt vé xe của mình. |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Cần đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Sinh viên chọn chức năng xem lịch sử đặt vé  Hệ thống sẽ hiển thị danh sách các vé đã đặt của sinh viên trong CSDL  Sinh viên có thể thực hiện các chức năng như xem chi tiết vé, đánh giá chuyến xe hoặc thanh toán trực tuyến cho vé sau khi truy cập vào trang lịch sử đặt vé |
| **Luồng thay thế** | * Ở bước 2, nếu sinh viên chưa đặt vé xe lần nào thì hệ thống sẽ thông báo rằng sinh viên hiện chưa có vé nào. |

### Chức năng Quản lý thông tin cá nhân

Hệ thống có chức năng Quản lý thông tin cá nhân. Ở đây sinh viên có thể xem được những thông tin của tài khoản, có thể chỉnh sửa thông tin và đổi mật khẩu.



Hình 3.9: Usecase chức năng Quản lý thông tin cá nhân

Bảng 3.11: Đặc tả usecase chức năng Chỉnh sửa thông tin

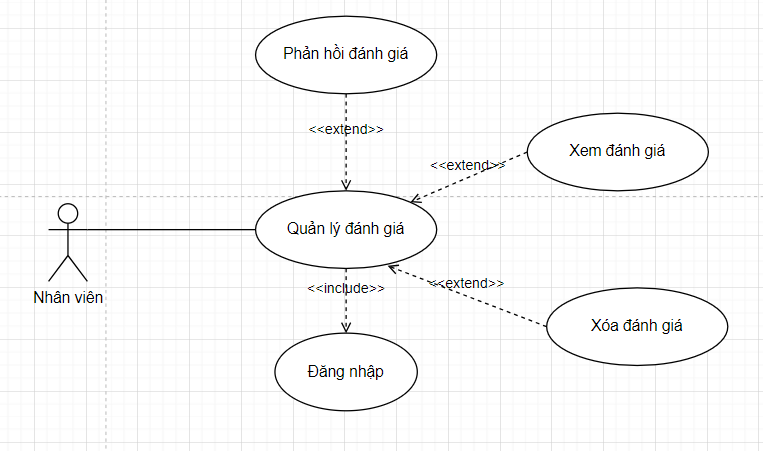
|  |  |
| --- | --- |
| **Tên usecase** | Chỉnh sửa thông tin |
| **Mô tả** | Cho phép sinh viên chỉnh sửa thông tin cá nhân của mình |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Cần đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Sinh viên chọn chức năng chỉnh sửa thông tin.  Sinh viên sẽ thay đổi thông tin theo nhu cầu  Sinh viên xác nhận thay đổi  Hệ thống lưu lại thông tin thay đổi. |
| **Luồng thay thế** | * Ở bước 2, nếu sinh viên không muốn thay đổi thì có thể chọn Thoát để thoát khỏi màn hình thay đổi thông tin, thông tin sẽ trở về như cũ. * Ở bước 3, nếu có lỗi thì sẽ quay lại màn hình chỉnh sửa thông tin. |

Bảng 3.12: Đặc tả usecase chức năng Đổi mật khẩu

|  |  |
| --- | --- |
| **Tên usecase** | Đổi mật khẩu |
| **Mô tả** | Cho phép sinh viên đổi mật khẩu cho tài khoản của mình |
| **Actor chính** | Sinh viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Cần đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | 1. Sinh viên chọn chức năng Đổi mật khẩu 2. Sinh viên nhập mật khẩu cũ và mật khẩu mới. 3. Sinh viên xác nhận thay đổi mật khẩu 4. Hệ thống lưu trữ mật khẩu mới |
| **Luồng thay thế** | * Ở bước 2, nếu sinh viên nhập mật khẩu cũ sai thì hệ thống sẽ yêu cầu nhập lại. * Ở bước 3, nếu sinh viên không muốn thay đổi mật khẩu thì chọn Thoát, quay về bước 1. |

### Chức năng Quản lý đánh giá

Nhân viên có nhiệm vụ quản lý danh sách các đánh giá của sinh viên. Nhân viên có thể xem được danh sách các đánh giá, có thể xóa đánh giá khi nội dung không được phù hợp.



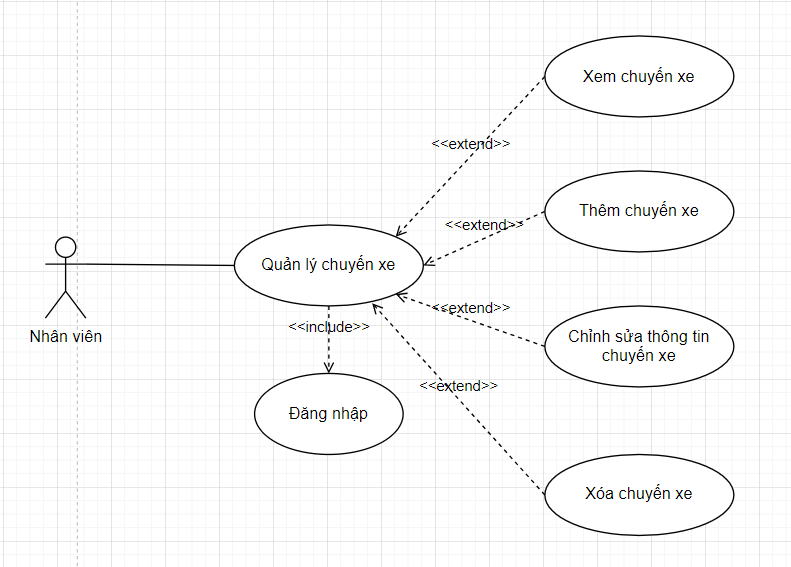
Hình 3.10: Usecase chức năng Quản lý đánh giá

Bảng 3.13: Đặc tả usecase chức năng Phản hồi đánh giá

|  |  |
| --- | --- |
| **Tên usecase** | Phản hồi đánh giá |
| **Mô tả** | Cho phép nhân viên có thể phản hồi đánh giá. |
| **Actor chính** | Nhân viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Cần đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Nhân viên chọn đánh giá cần được phản hồi.  Nhân viên nhập nội dung phản hồi.  Nhân viên xác nhận phản hồi.  Hệ thống sẽ lưu trữ và hiển thị phản hồi. |
| **Luồng thay thế** | * Ở bước 1, nếu nhân viên đã phản hồi đánh giá của sinh viên rồi thì hiển thị đánh giá của sinh viên và phản hồi của nhân viên, có thể thực hiện chỉnh sửa phản hồi. * Ở bước 3, nếu nhân viên không muốn phản hồi nữa thì có thể chọn Thoát, hệ thống sẽ quay trở lại màn hình sau khi thực hiện bước 1. * Ở bước 4, nếu hệ thống xảy ra lỗi trong quá trình lưu trữ thì hệ thống sẽ quay lại bước 2. |

### Chức năng Quản lý chuyến xe

Nhân viên có quyền Quản lý chuyến xe, họ có thể xem danh sách các chuyến xe, thêm chuyến, sửa thông tin của chuyến hoặc xóa chuyến xe khi không đủ lượng sinh viên đặt vé.



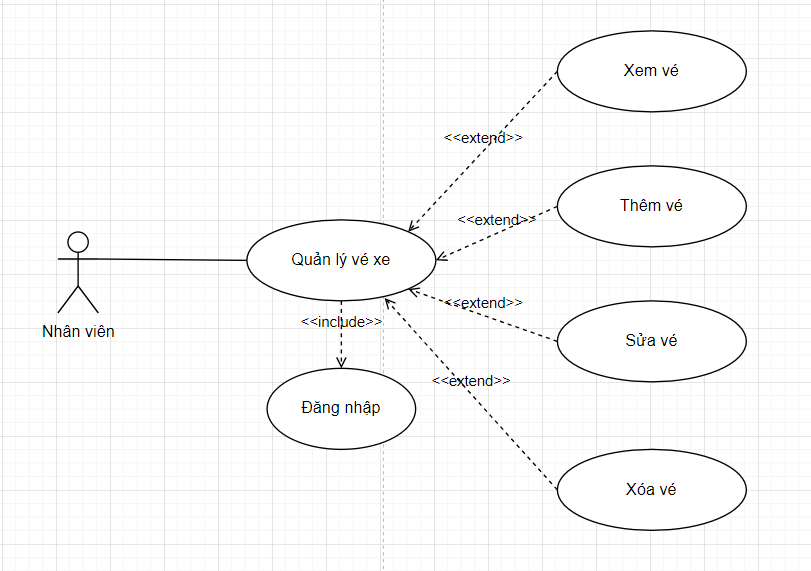
Hình 3.11: Usecase Quản lý chuyến xe

Bảng 3.14: Đặc tả usecase chức năng Thêm chuyến xe

|  |  |
| --- | --- |
| **Tên usecase** | Thêm chuyến xe |
| **Mô tả** | Cho phép nhân viên có thể thêm các chuyến xe. |
| **Actor chính** | Nhân viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Cần đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Nhân viên chọn chức năng Thêm chuyến xe.  Nhân viên nhập thông tin cho chuyến xe.  Nhân viên xác nhận tạo chuyến xe.  Hệ thống kiểm tra tính hợp lệ của chuyến xe.  Hệ thống thông báo chuyến xe được tạo thành công. |
| **Luồng thay thế** | * Ở bước 3, nếu nhân viên không muốn tạo chuyến xe nữa thì có thể chọn Thoát, hệ thống sẽ quay lại bước 1. * Ở bước 4, Hệ thống nhận thấy thông tin chuyến xe không hợp lệ thì sẽ thông báo lỗi cho nhân viên. |

### Chức năng Quản lý vé xe

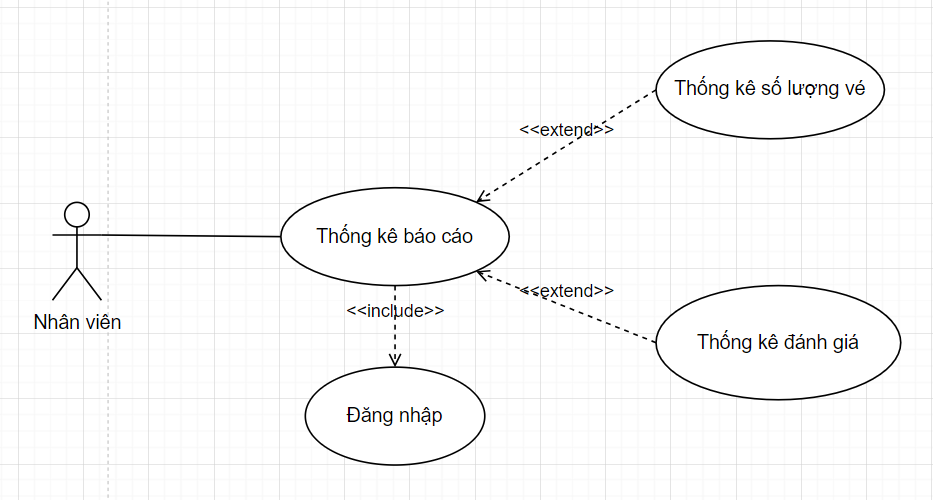
Chức năng chủ yếu phục vụ cho việc sinh viên đặt vé thông qua nhân viên chứ không đặt trực tiếp trên website.



Hình 3.12: Usecase chức năng Quản lý vé xe

### Chức năng Thống kê báo cáo

Thống kê báo cáo là chức năng vô cùng quan trọng trong một hệ thống. Đây là chức năng giúp nhân viên có kiểm soát được tình hình hoạt động của hệ thống đặt xe.



Hình 3.13: Usecase chức năng Thống kê báo cáo

Bảng 3.15: Đặc tả chức năng Thống kê số lượng vé

|  |  |
| --- | --- |
| **Tên usecase** | Thống kê số lượng vé |
| **Mô tả** | Cho phép nhân viên xem được các thống kê về số lượng vé bán ra. |
| **Actor chính** | Nhân viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Cần đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | Nhân viên chọn chức năng Thống kê theo số lượng vé  Nhân viên chọn khoảng thời gian cần thống kê  Hệ thống sẽ hiển thị kết quả thống kê |
| **Luồng thay thế** | * Ở bước 2, nếu nhân viên không chọn khoảng thời gian cần thống kê thì hệ thống sẽ mặc định hiển thị thống kê trong một tháng gần nhất. |

Bảng 3.16: Đặc tả chức năng Thống kê đánh giá

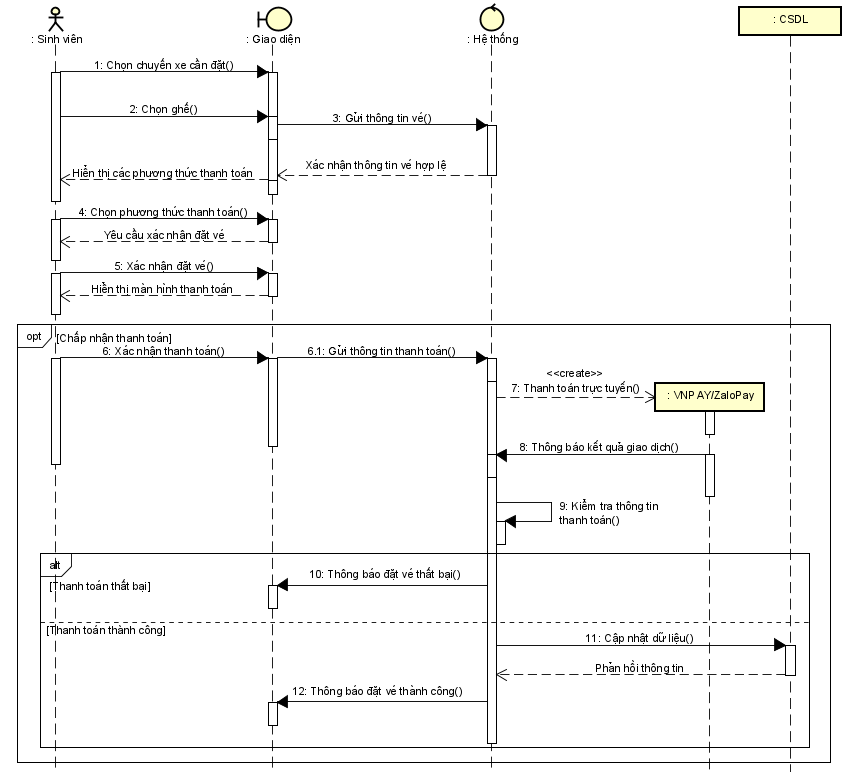
|  |  |
| --- | --- |
| **Tên usecase** | Thống kê mật độ chuyến xe |
| **Mô tả** | Cho phép nhân viên xem được các thống kê về các đánh giá của sinh viên |
| **Actor chính** | Nhân viên |
| **Actor phụ** | Không có |
| **Tiền điều kiện** | Cần đăng nhập |
| **Hậu điều kiện** | Không có |
| **Luồng hoạt động** | 1. Nhân viên chọn chức năng Thống kê đánh giá. 2. Nhân viên chọn khoảng thời gian cần thống kê. 3. Hệ thống sẽ hiển thị giao diện thống kê. |
| **Luồng thay thế** | * Ở bước 2, nếu nhân viên không chọn khoảng thời gian cần thống kê thì hệ thống sẽ mặc định hiển thị thống kê trong một tuần gần nhất. |

## Thiết kế hệ thống

### Sơ đồ tuần tự (Sequence Diagram)

1. Đặt vé trực tuyến

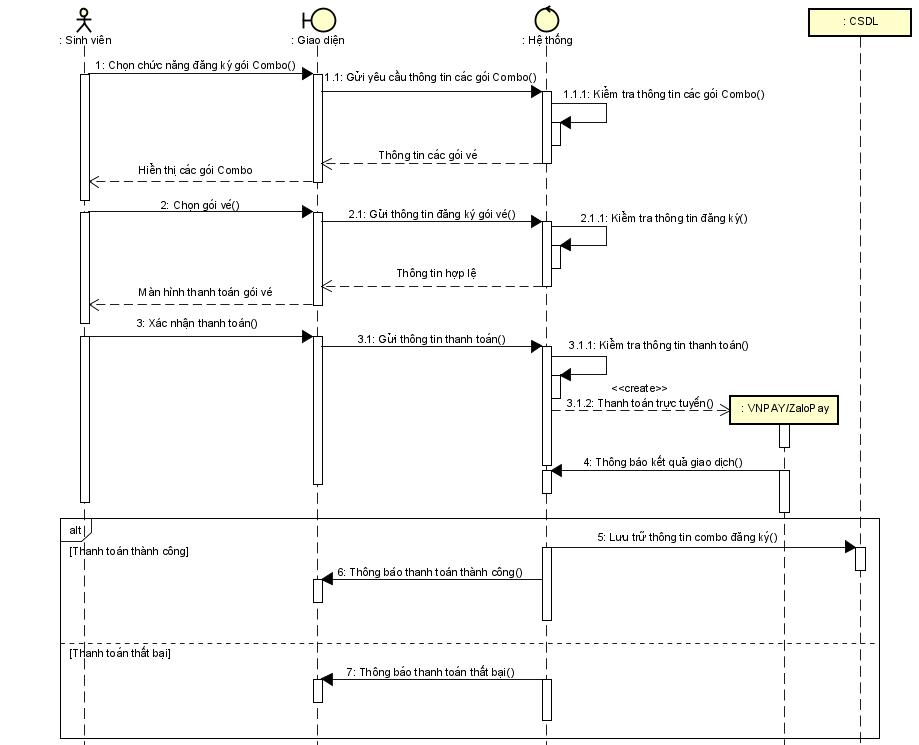
Sơ đồ tuần tự của chức năng Đặt vé trực tuyến cho biết quá trình tương tác giữa các đối tượng tham gia vào chức năng Đặt vé trực tuyến như Sinh Viên, Giao diện, Hệ thống, Cơ sở dữ liệu và có sự tích hợp thêm các ví điện tử ZaloPay, VNPAY để phục vụ cho chức năng thanh toán vé.



Hình 3.14: Sơ đồ tuần tự cho usecase Đặt vé trực tuyến

1. Đăng ký gói Combo

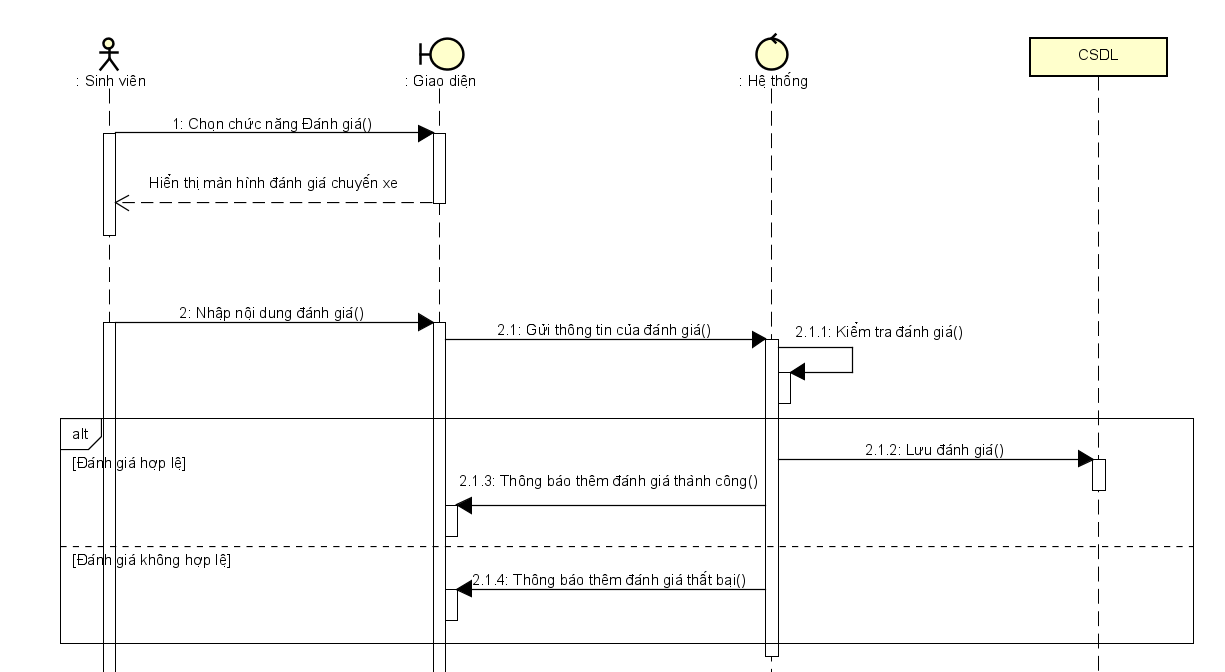
Chức năng Đăng ký gói Combo có sự tham gia của các đối tượng tương tự với chức năng Đặt vé trực tuyến như Sinh viên, Giao diện, Hệ thống, Cơ sở dữ liệu và có sự tích hợp với ví điện tử như ZaloPay, VNPAY để hỗ trợ hoàn thành thanh toán.



Hình 3.15: Sơ đồ tuần tự của usecase Đăng ký gói Combo

1. Thêm đánh giá

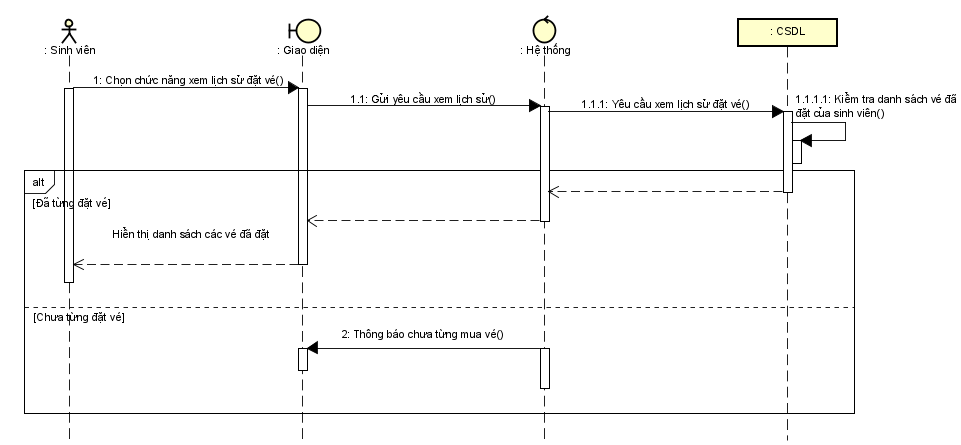
Sơ đồ tuần tự của chức năng Thêm đánh giá có các đối tượng như Sinh Viên, Giao diện, Hệ thống và Cơ sở dữ liệu. Sơ đồ mô tả các tương tác giữa các đối tượng trong quá trình thực hiện chức năng Thêm đánh giá, từ việc chọn chức năng đánh giá, thêm đánh giá đến việc hệ thống phản hồi lại kết quả.



Hình 3.16: Sơ đồ tuần tự của usecase Thêm đánh giá

1. Xem lịch sử đặt vé

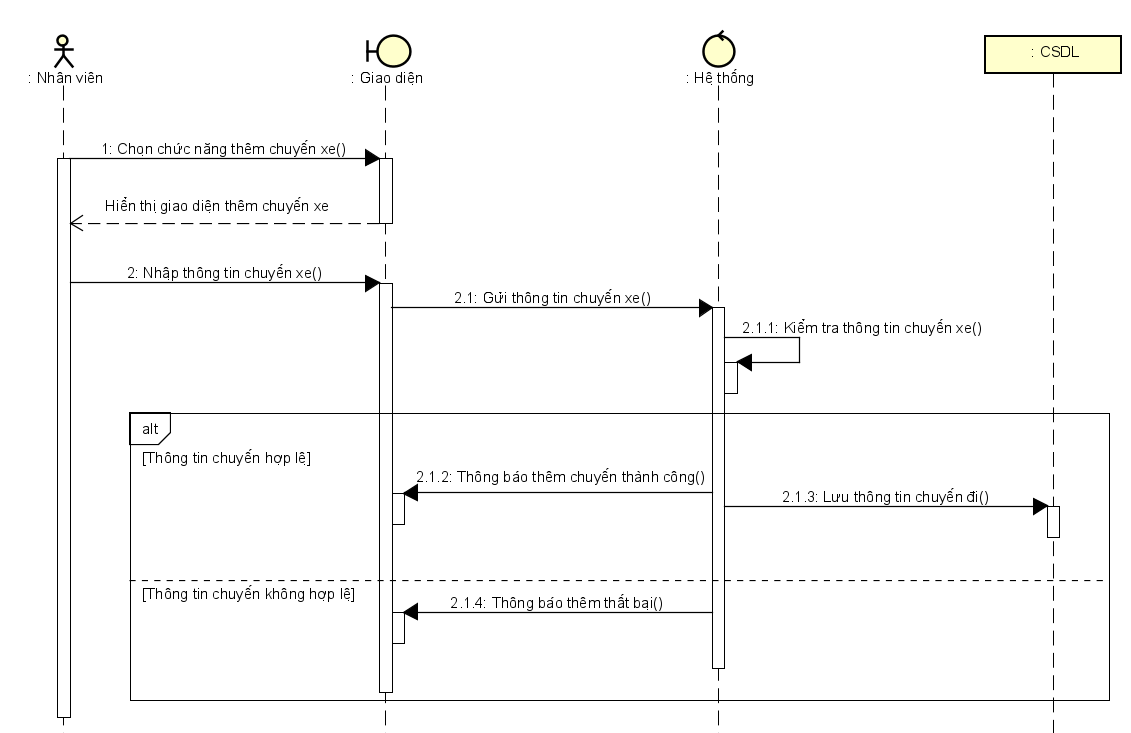
Sơ đồ tuần tự sau mô tả các tương tác giữa các đối tượng như Sinh viên, Giao diện, Hệ thống, Cơ sở dữ liệu của chức năng Xem lịch sử đặt vé.



Hình 3.17: Sơ đồ tuần tự của usecase Xem lịch sử đặt vé

1. Thêm chuyến xe

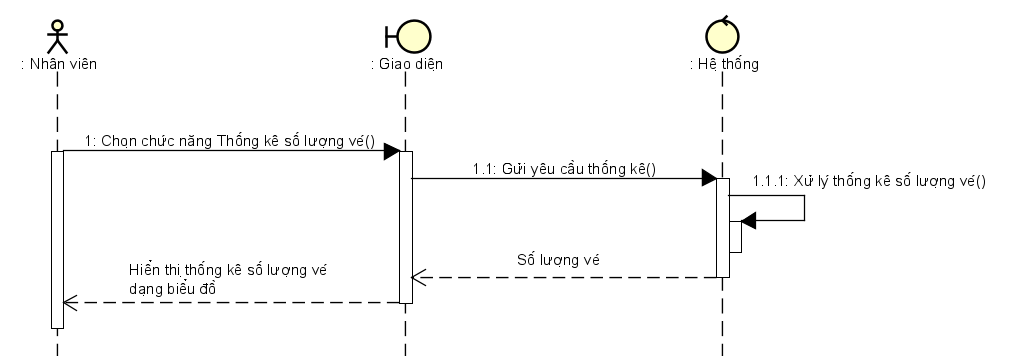
Chức năng Thêm chuyến xe sẽ có sự tham gia của các đối tượng như Nhân viên, Giao diện, Hệ thống và CSDL. Sơ đồ tuần tự sau sẽ mô tả chi tiết cách thức các đối tượng tương tác với nhau để hình thành nên chức năng Thêm chuyến xe.



Hình 3.18: Sơ đồ tuần tự của usecase Thêm chuyến xe

1. Thống kê số lượng vé

Sơ đồ tuần tự của chức năng Thống kê số lượng vé cho biết sự tương tác lẫn nhau của các đối tượng như Nhân viên, Giao diện và Hệ thống trong quá trình thực hiện chức năng Thống kê số lượng vé.

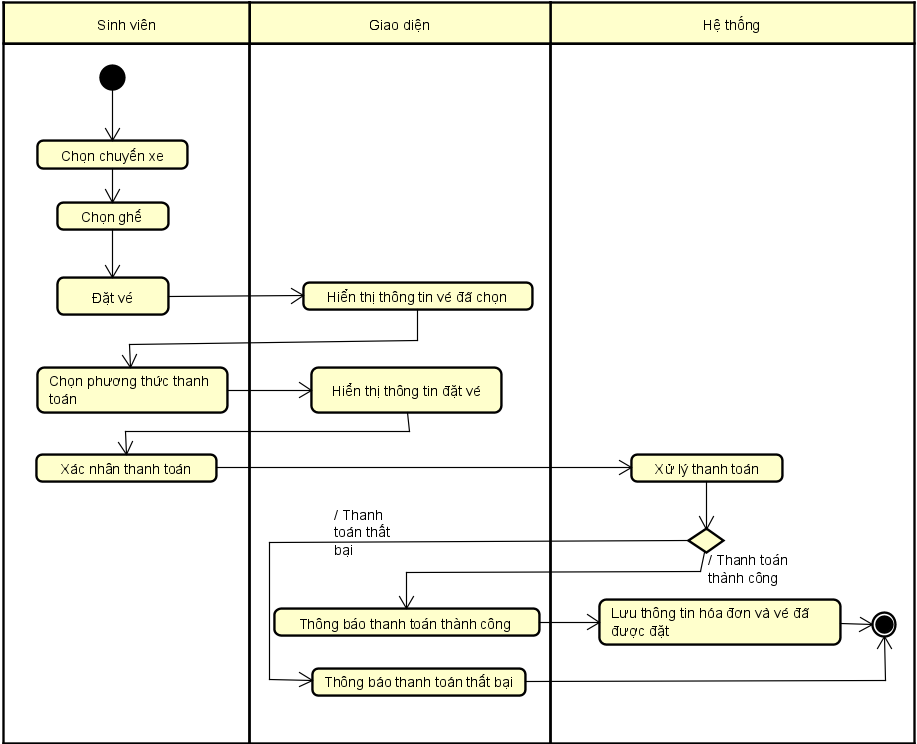


Hình 3.19: Sơ đồ tuần tự usecase Thống kê số lượng vé

### Sơ đồ hoạt động (Activity Diagram)

1. Đặt vé trực tuyến

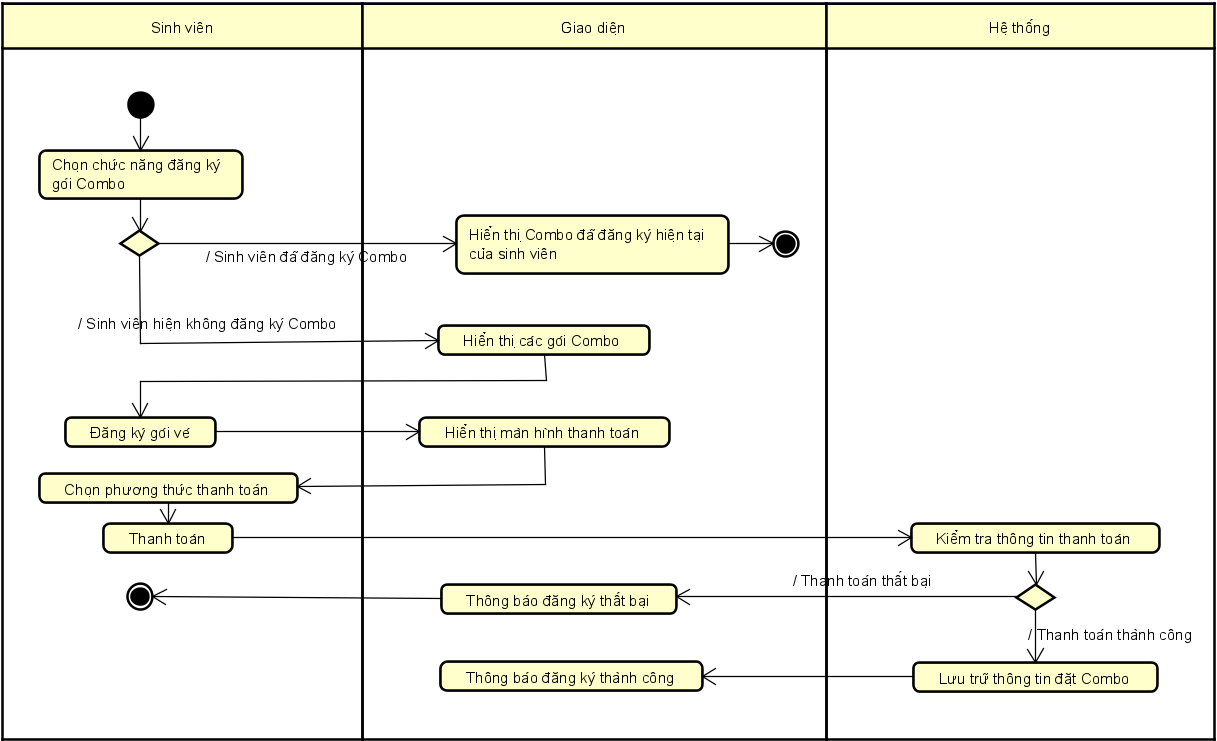
Sơ đồ hoạt động của chức năng Đặt vé trực tuyến trong hệ thống này mô tả quá trình mà sinh viên thực hiện từ khi chọn chuyến xe đến khi thanh toán thành công hoặc thất bại. Sơ đồ này giúp hình dung chi tiết các bước trong quá trình đặt vé và xử lý thanh toán của hệ thống.



Hình 3.20: Sơ đồ hoạt động của usecase Đặt vé trực tuyến

1. Đăng ký gói Combo

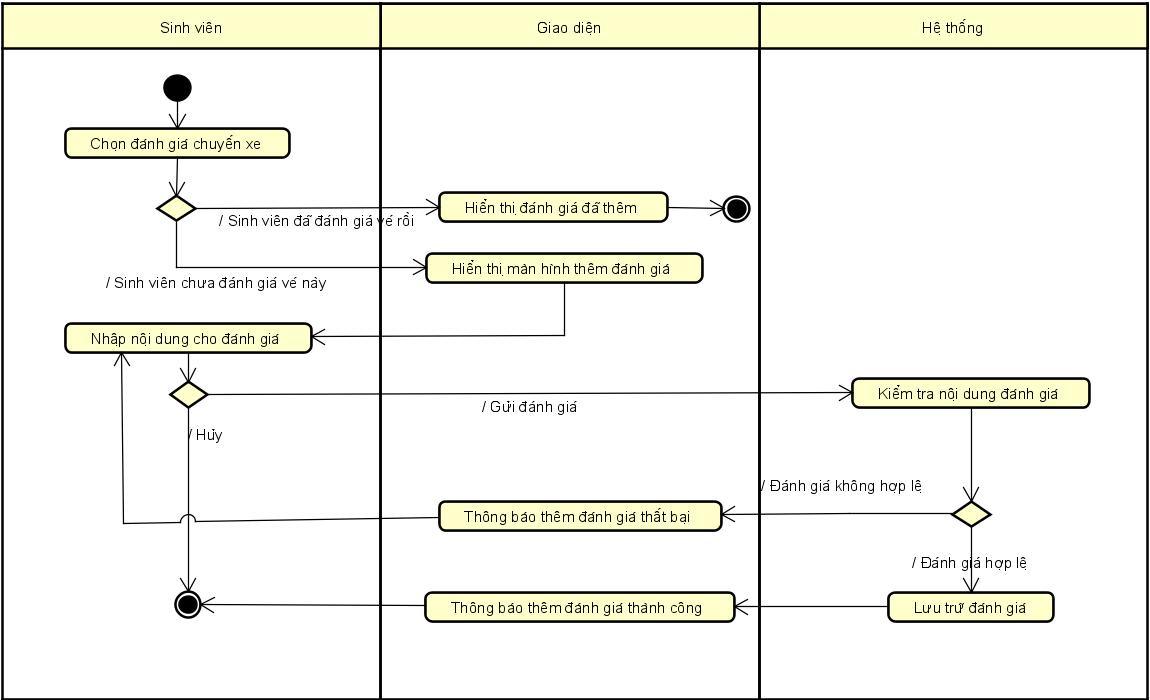
Sơ đồ hoạt động của chức năng Đăng ký gói Combo mô tả quá trình mà sinh viên thực hiện từ khi chọn vào chức năng Đăng ký gói Combo cho đến khi thanh toán thành công hoặc thất bại. Nếu sinh viên hiện đã sử dụng gói Combo nào đó thì hệ thống sẽ không cho đăng ký và sẽ hiển thị trạng thái Combo hiện tại mà sinh viên đã đăng ký.



Hình 3.21: Sơ đồ hoạt động của usecase Đăng ký gói Combo

1. Thêm đánh giá

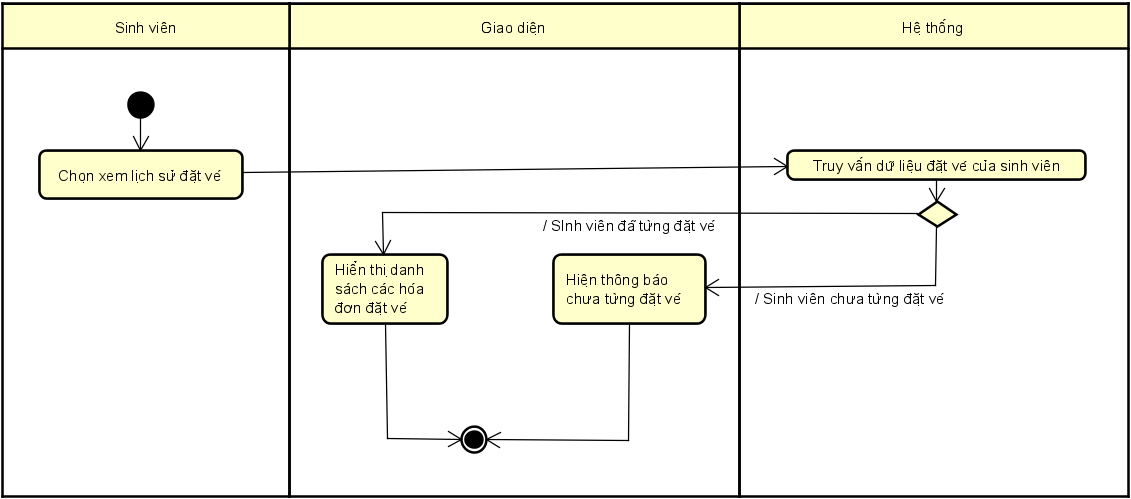
Sơ đồ hoạt động của chức năng Thêm đánh giá mô tả quá trình mà sinh viên thực hiện từu khi chọn chức năng Chọn đánh giá chuyến xe đến khi hệ thống thông báo thêm đánh giá thành công hoặc thất bại. Nếu sinh viên đã đánh giá chuyến xe này rồi thì hệ thống sẽ hiển thị nội dung đánh giá và sẽ hiển thị phản hồi của nhân viên nếu có, và không cho phép thêm đánh giá ở chuyến xe đó nữa.



Hình 3.22: Sơ đồ hoạt động của usecase Thêm đánh giá

1. Xem lịch sử đặt vé

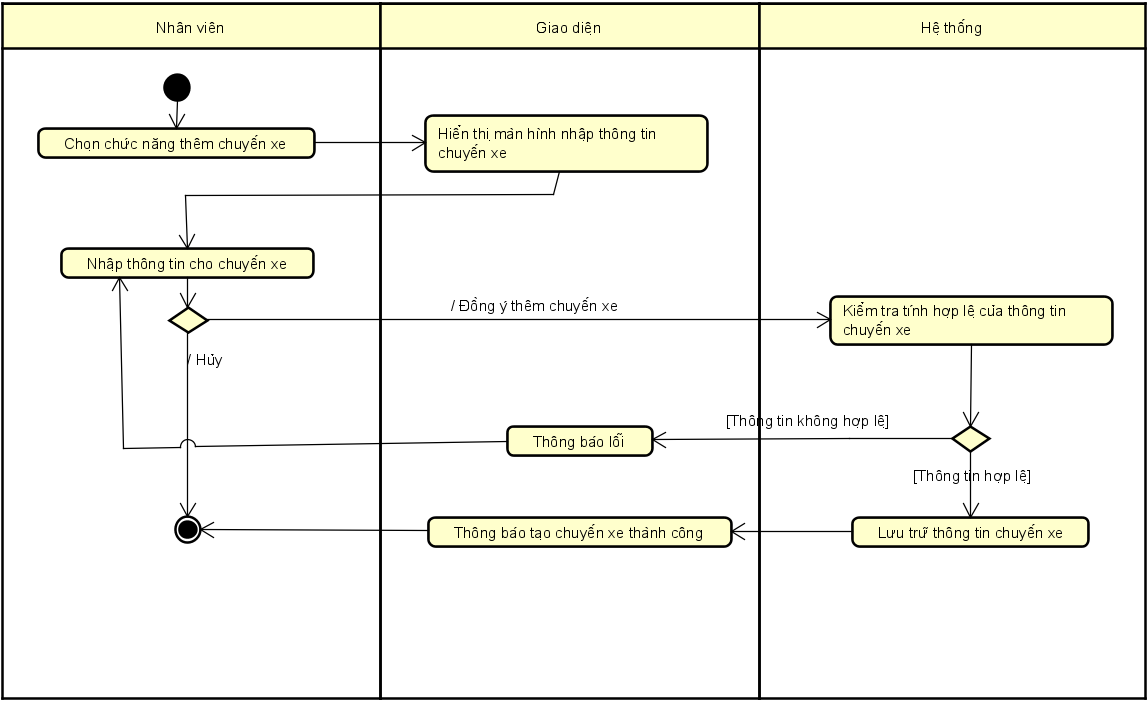
Sơ đồ hoạt động của chức năng Xem lịch sử đặt vé mô tả quá trình mà sinh viên chọn chức năng Xem lịch sử đặt vé đến khi hệ thống hiển thị danh sách các vé mà sinh viên đã đặt hoặc nếu sinh viên chưa từng đặt vé thì hệ thống sẽ hiển thị thông báo chưa từng đặt vé.



Hình 3.23: Sơ đồ hoạt động của usecase Xem lịch sử đặt vé

1. Thêm chuyến xe

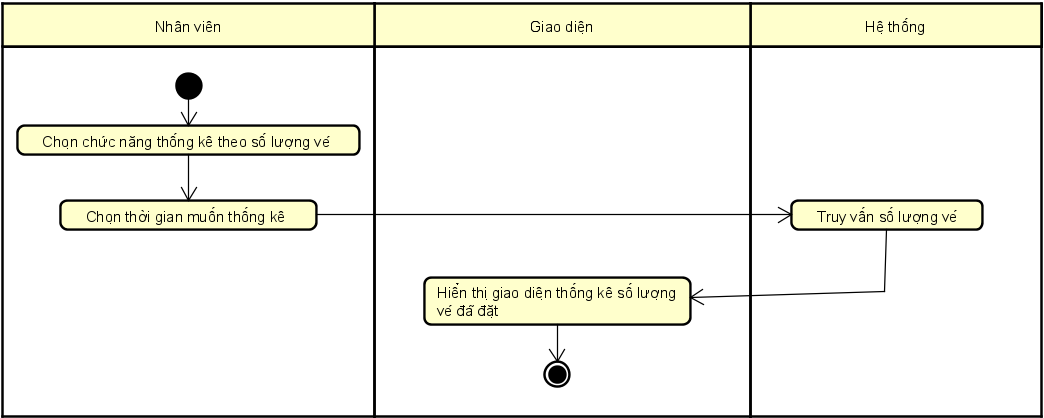
Sơ đồ hoạt động này mô tả quá trình thực hiện thêm chuyến xe của nhân viên từ lúc nhân viên chọn chức năng thêm chuyến xe đến khi hệ thống kiểm tra tính hợp lệ thông tin của chuyến xe mới. Nếu thông tin hợp lệ thì chuyến xe mới sẽ được thêm, nếu không hợp lệ thì hệ thống sẽ thông báo lỗi và quay trở lại giao diện nhập thông tin.



Hình 3.24: Sơ đồ hoạt động của usecase Thêm chuyến xe

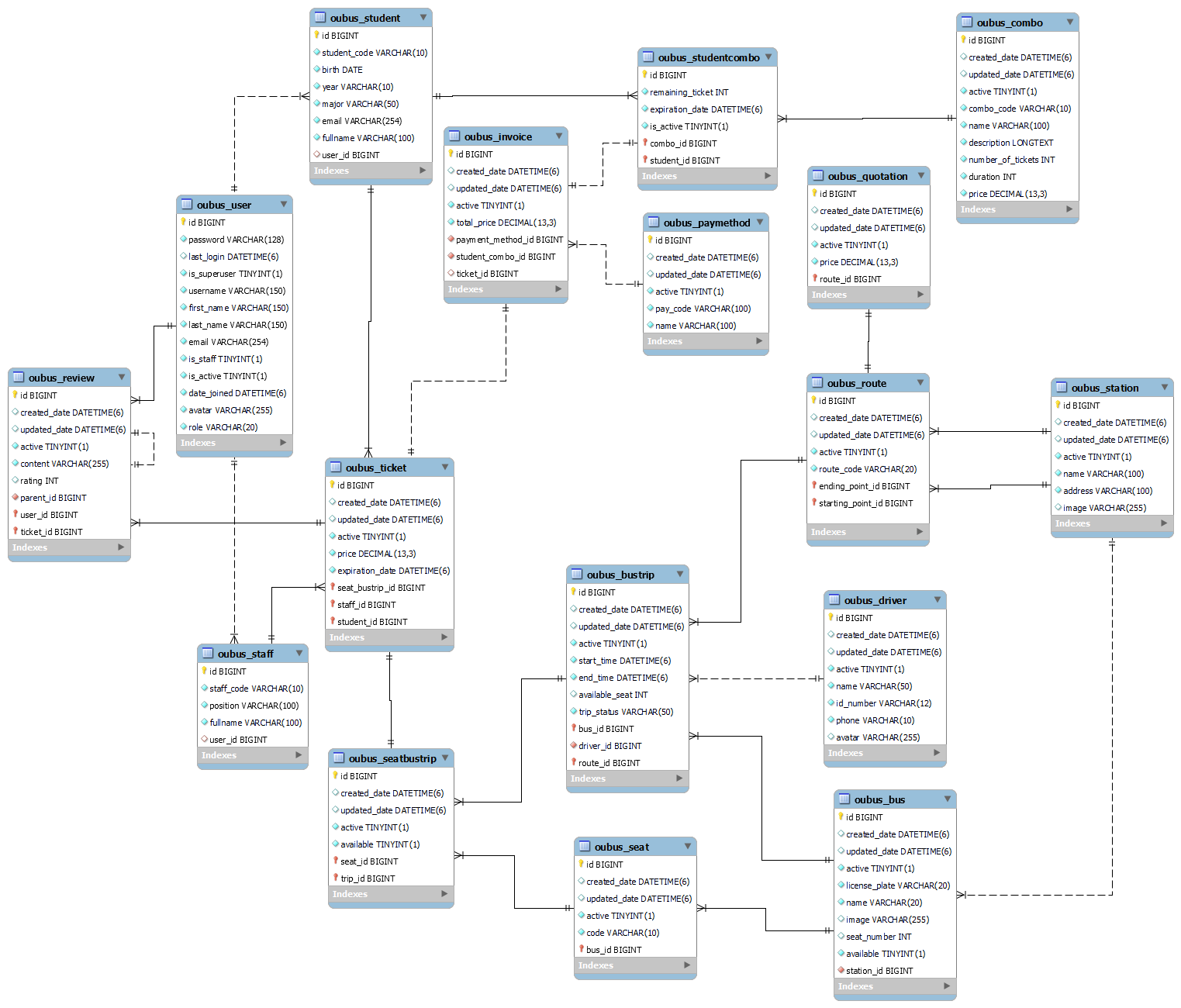
1. Thống kê số lượng vé

Sơ đồ hoạt động này mô tả quá trình thực hiện xem thống kê số lượng vé mà sinh viên đã đặt của nhân viên. Nhân viên có thể tùy chọn thời gian để hệ thống thống kê.



Hình 3.25: Sơ đồ hoạt động của usecse Thống kê số lượng vé

### Thiết kế dữ liệu



Hình 3.26: Lược đồ cơ sở dữ liệu quan hệ của hệ thống đặt vé xe bus

Student – Ticket: Mối quan hệ giữa sinh viên và vé xe là mối quan hệ 1- n. Một sinh viên có thể đặt được nhiều vé xe, mỗi vé xe sẽ chỉ lưu trữ thông tin của 1 sinh viên. Ghi nhận thông tin của sinh viên vào vé khi sinh viên thực hiện chức năng đặt vé.

Staff – Ticket: Mối quan hệ giữa nhân viên và vé xe là mối quan hệ 1 – n. Mối quan hệ giúp ghi nhận thông tin của nhân viên lập vé. Một nhân viên có thể lập nhiều vé xe, mỗi vé xe đều lưu trữ thông tin của 1 nhân viên xác lập vé, nếu sinh viên đặt trực tuyến thì nhân viên được lưu trữ trên vé là Hệ thống.

User – Review: Mối quan hệ giữa người dùng và đánh giá là mối quan hệ 1 – n. Mối quan hệ này giúp lưu lại thông tin của chủ sở hữu đánh giá. Mỗi người dùng có thể để lại nhiều đánh giá, mỗi đánh giá lưu trữ thông tin của 1 người dùng đã tạo đánh giá đó.

Ticket – Review: Mối quan hệ giữa vé xe và đánh giá là mối quan hệ 1 – n. Một vé xe có thể có tối đa 2 đánh giá. Một đánh giá đến từ sinh viên, và một phản hồi đánh giá đến từ nhân viên. Mỗi đánh giá đều lưu trữ thông tin của một vé mà nó thuộc về.

Review – Review: Mối quan hệ giữa đánh giá của sinh viên và phản hồi của nhân viên là mối quan hệ 1 – 1. Mối quan hệ này giúp giải quyết cho chức năng phản hồi lại đánh giá của nhân viên. Mỗi đánh giá của sinh viên chỉ được phản hồi với 1 phản hồi của nhân viên. Và mỗi phản hồi của nhân viên chỉ phản hồi cho 1 đánh giá của sinh viên.

Student – Combo: Mối quan hệ giữa sinh viên và gói Combo là mối quan hệ n – n, sinh ra bảng StudentCombo. Mối quan hệ này mô tả cho chức năng sinh viên đăng ký gói Combo. Mỗi sinh viên đều có đăng ký nhiều gói Combo khác nhau ở các thời điểm khác nhau, mỗi gói Combo khi đăng ký cũng sẽ lưu trữ thông tin của sinh viên đã đăng ký. Dữ liệu sinh viên và trạng thái Combo mà sinh viên đang đăng ký sẽ được lưu trữ tại bảng StudentCombo.

Route – Bustrip: Mối quan hệ giữa tuyến xe chạy và chuyến xe là mối quan hệ 1 – n.

Mỗi tuyến xe sẽ có nhiều chuyến xe hoạt động, mỗi một chuyến xe chỉ chạy 1 tuyến đường duy nhất.

Bustrip – Seat: Mối quan hệ giữa chuyến xe và ghế ngồi là mối quan hệ n – n, phát sinh ra bảng SeatBustrip. Mối quan hệ được tạo thành khi mỗi chuyến xe được tạo thành, các vị trí ngồi trên chiếc xe bus đó sẽ được lưu lại trạng thái (sẵn sàng/ không sẵn sàng) để có thể thể phục vụ cho việc đặt vé xe. Trạng thái đấy sẽ được lưu trữ tại bảng SeatBustrip.

Ticket – SeatBustrip: Mối quan hệ giữa vé xe và ghế ngồi trên chuyến xe là mối quan hệ 1 – 1. Một vé xe sẽ bao gồm thông tin của vị trí ngồi trên chuyến xe mà sinh viên đã đặt, và chỗ ngồi của chuyến xe đó chỉ thuộc về 1 vé xe.

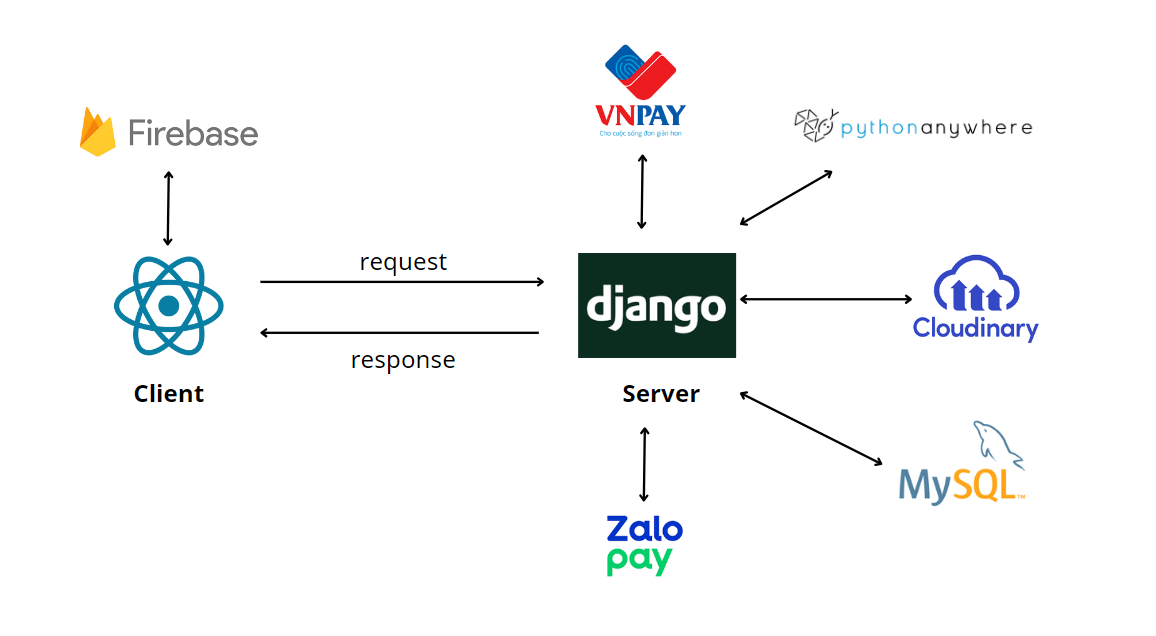
Invoice – Ticket: Mối quan hệ giữa hóa đơn và vé xe là mối quan hệ 1 – 1. Một vé xe chỉ tồn tại một hóa đơn khi vé đó đã được sinh viên thanh toán, và một hóa đơn chỉ lưu trữ thông tin 1 vé xe mà nó chịu trách nhiệm xác minh thanh toán.

Invoice – StudentComBo: Mối quan hệ giữa hóa đơn và gói Combo mà sinh viên đã đăng ký là mối quan hệ 1 – 1. Một hóa đơn chỉ lưu trữ thông tin của gói Combo mà sinh viên đã đăng ký, và gói Combo đó chỉ được xác minh thanh toán với một hóa đơn duy nhất.

Route – Quotation: Mối quan hệ giữa tuyến xe chạy và bảng giá vé xe là mối quan hệ 1 – 1. Mỗi một tuyến xe sẽ có một mức giá duy nhất, và nhờ vào thông tin này để có thể gán giá trị tiền cho vé xe tùy theo các tuyến xe đã được định giá tiền.

Thông tin chi tiết của các bảng và các cột sẽ được ghi ở phần PHỤ LỤC

## Kiến trúc hệ thống



Hình 3.27: Kiến trúc của hệ thống đặt vé xe OU Bus

ReactJS được sử dụng để xây dựng giao diện của người dùng (UI). Đây là một thư viện Javascripts mạnh mẽ cho phép tạo các ứng dụng web tương tác và linh hoạt. Ở phía client sử dụng ReactJS để tương tác với các API của Django thông qua các HTTP request (GET, POST, PUT, DELETE) tới các endpoint của Django Rest Framework. Ở phía client cũng sẽ tương tác trực tiếp với Firebase để thực hiện chức năng chat realtime.

Về phía server sử dụng Django Rest Framework chịu trách nhiệm cung cấp các API RESTful, xử lý logic nghiệp vụ và quản lý dữ liệu giữa client và cơ sở dữ liệu. Các endpoint API do DRF cung cấp cho phép client lấy, gửi, chỉnh sửa và xóa dữ liệu bên trong hệ thống, phản hồi lại request của client. Bên cạnh đó, hệ thống sử dụng Oauth2 làm cơ chế ủy quyền (authorization), giúp xác thực các yêu cầu từ phía người dùng và có thời hạn.

Hai cổng thanh toán trực tuyến ZaloPay và VNPAY được tích hợp vào hệ thống ở phía server. Django sẽ tương tác với các API của hai cổng thanh toán trên để xử lý các yêu cầu thanh toán của người dùng, bao gồm việc tạo giao dịch vào nhận kết quả thanh toán.

Đồng thời, phía server có sử dụng PythonAnywhere, đây là một nền tảng đám mây được xây dựng giúp dễ dàng lưu trữ và triển khai ứng dụng Python, bao gồm API Django và cơ sở dữ liệu MySQL.

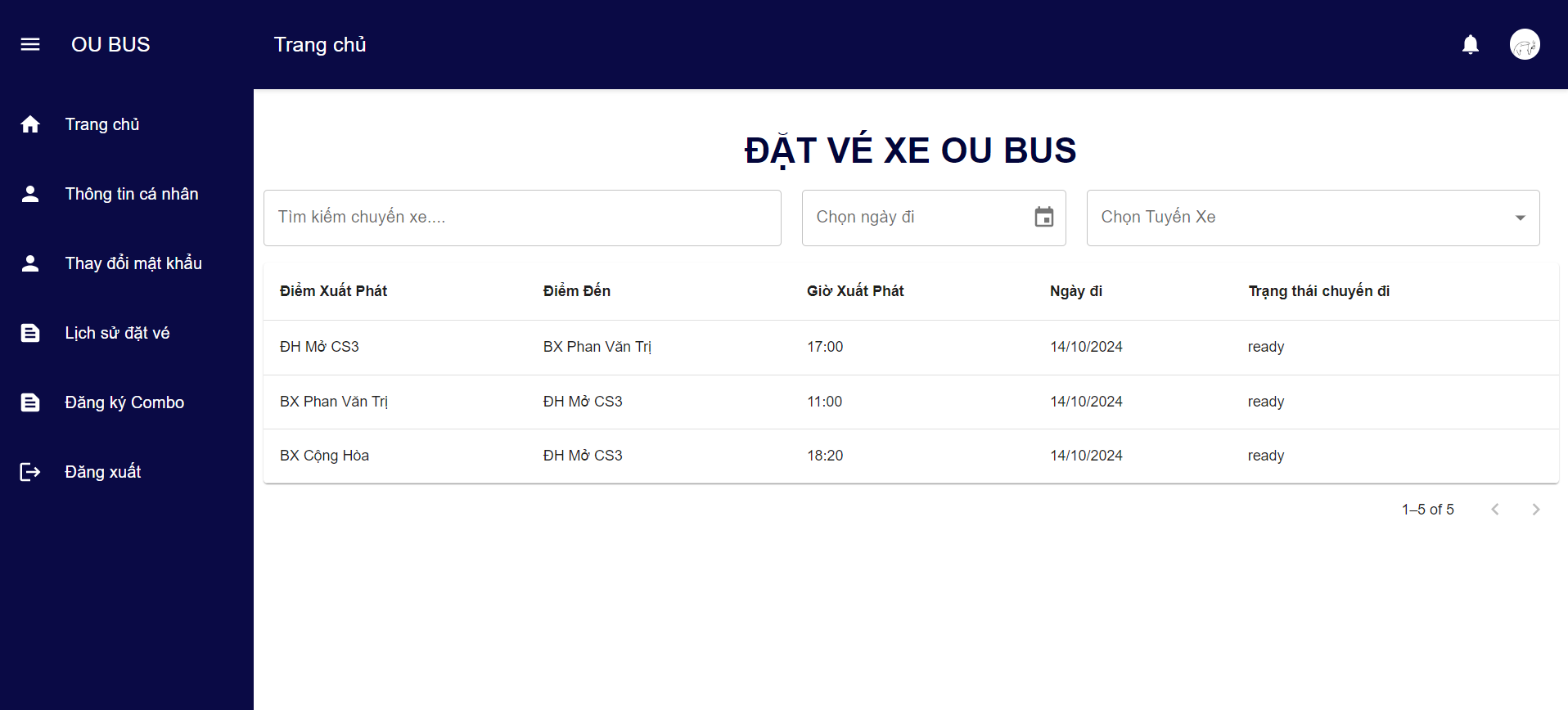
Tại server, hệ thống sử dụng hệ quản trị cơ sở dữ liệu MySQL để lưu trữ toàn bộ dữ liệu của hệ thống. Bên cạnh đó, Django sử dụng ORM để kết nối và tương tác với MySQL, giúp quản lý các truy vấn tới MySQL để đọc hoặc ghi dữ liệu dựa trên các yêu cầu từ client.

Phía server cũng sử dụng Cloudinary, đây là nơi lưu trữ các dữ liệu là hình ảnh trong hệ thống. Các ảnh được upload từ client (ReactJS) thông qua API của Django Rest Framework gửi tới Cloudinary, sử dụng CloudinaryField để lưu đường dẫn tới ảnh trong cơ sở dữ liệu MySQL.

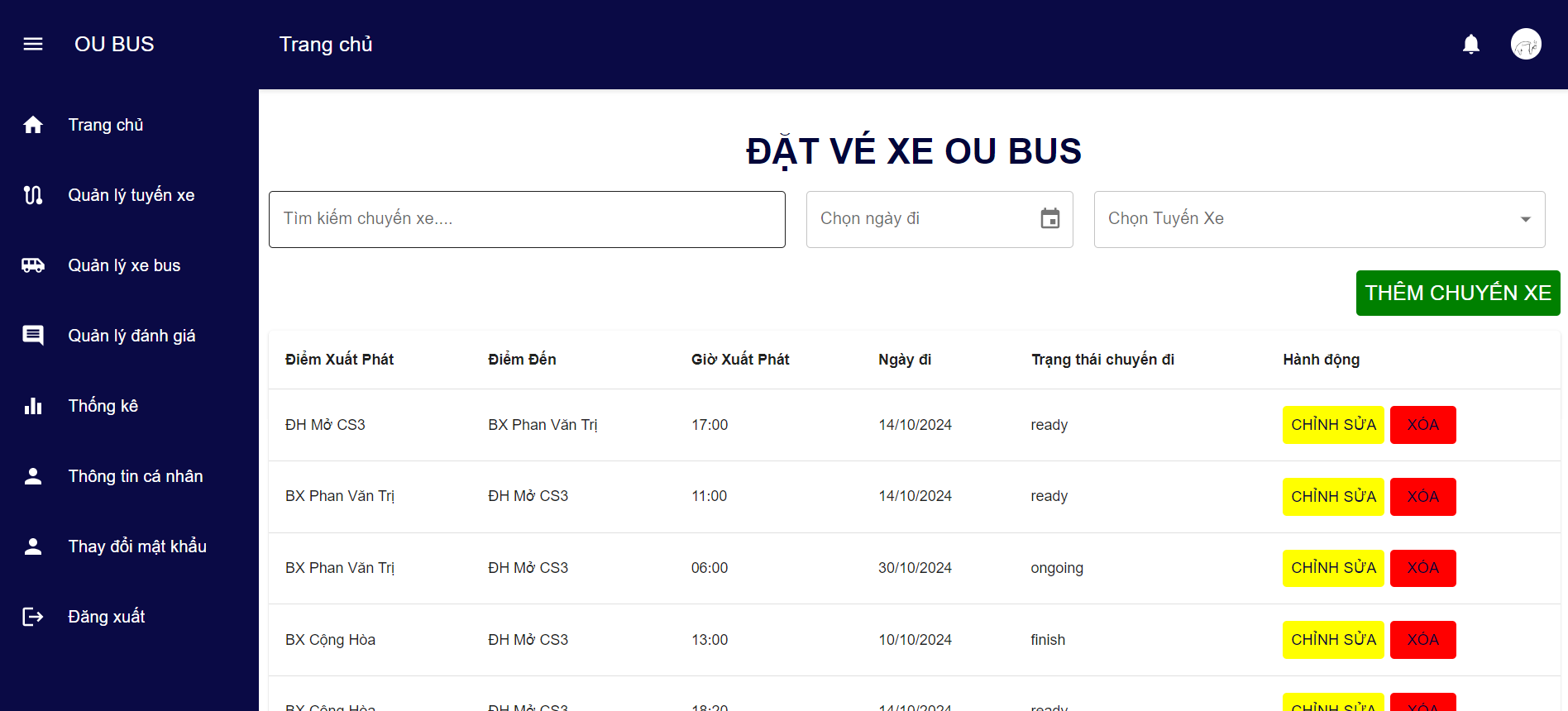
## Kết quả thực hiện

### Chức năng Xem danh sách chuyến xe

Ở chức năng xem danh sách chuyến xe, nếu người sử dụng hệ thống là sinh viên sẽ hiển thị danh sách các chuyến xe hiện đang sẵn sàng để sinh viên đặt vé. Sinh viên có thể tìm kiếm chuyến xe theo từ khóa dựa vào tên của bến xe, hoặc có thể lọc theo các tuyến xe chạy hay ngày mà chuyến xe khởi hành. Nếu người dùng là nhân viên thì sẽ hiển thị thêm các nút như chỉnh sửa, xóa hay thêm chuyến xe để nhân viên có thể quản lý được các chuyến xe trên hệ thống.



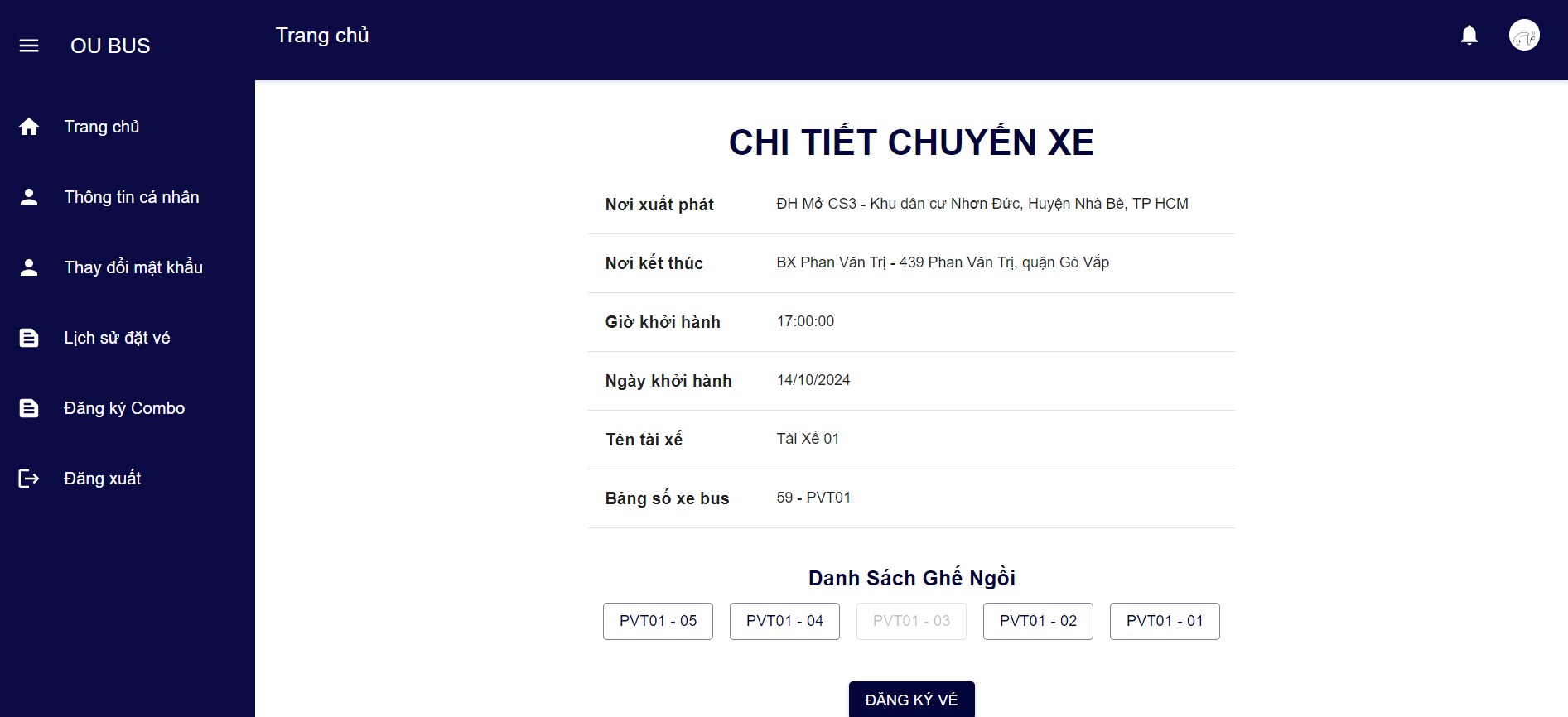
Hình 3.28: Giao diện hiển thị danh sách chuyến xe với sinh viên



Hình 3.29: Giao diện hiển thị danh sách chuyến xe với nhân viên

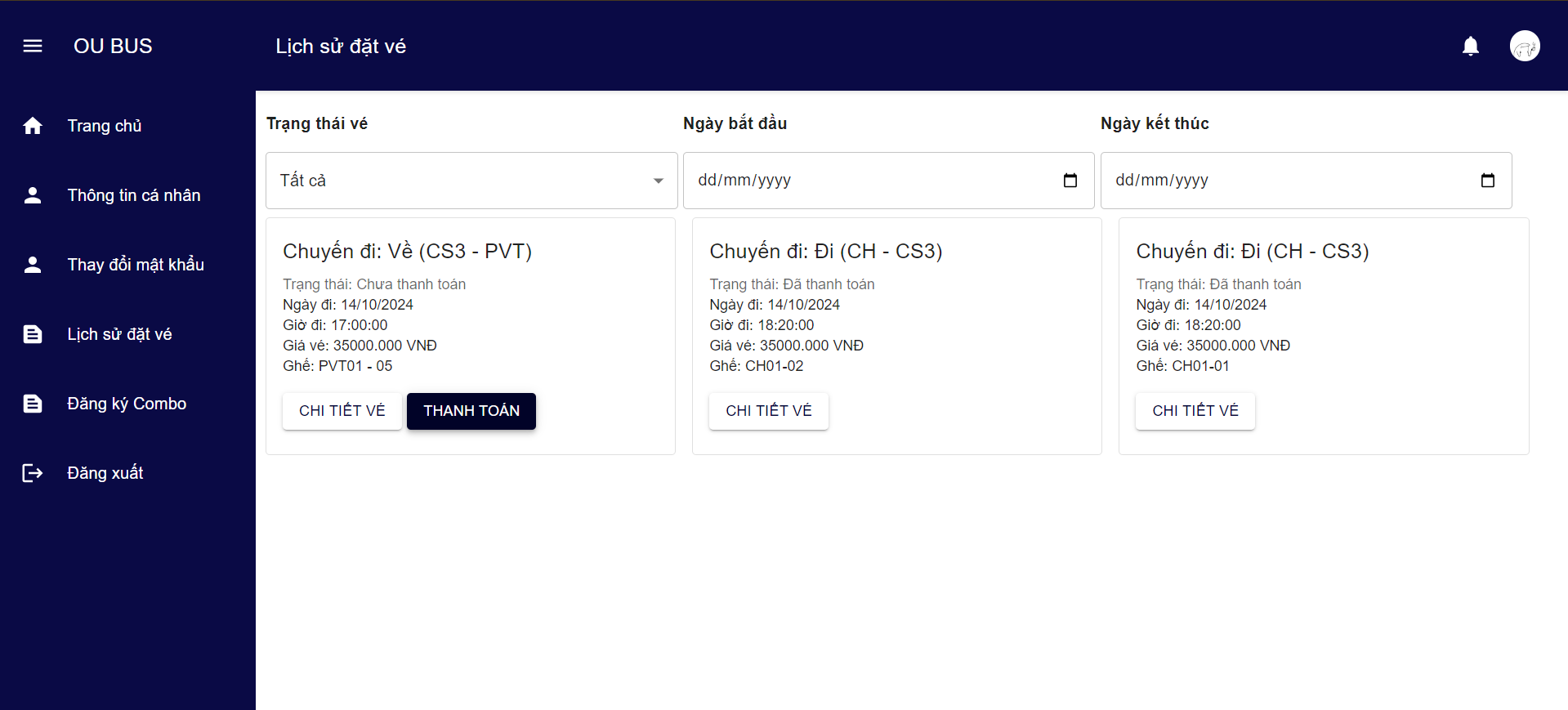
### Chức năng Đặt vé trực tuyến

Ở chức năng Đặt vé trực tuyến, sinh viên đầu tiên cần chọn chuyến xe theo mong muốn để có thể xem thông tin chi tiết của chuyến xe đó, sau đó chọn vị trí ngồi trên xe, sau đó có thể chọn Đăng ký vé.



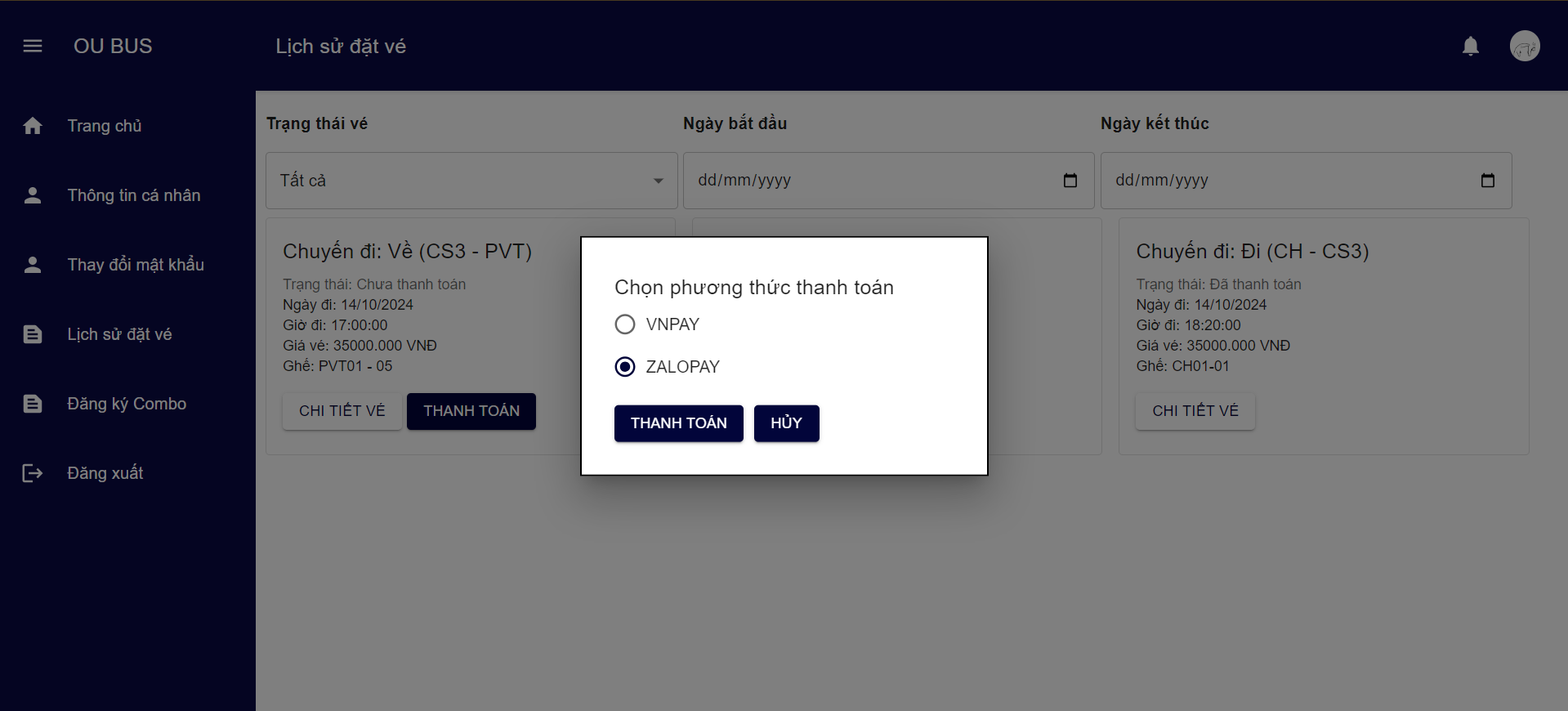
Hình 3.30: Giao diện chi tiết thông tin của chuyến xe và các chỗ ngồi

Sau khi đã đăng ký vé, vé của sinh viên sẽ hiển thị trong mục Lịch sử đặt vé, sinh viên có thể chọn thanh toán vé tại đây để vé có hiệu lực.



Hình 3.31: Giao diện hiển thị vé chưa thanh toán trong Lịch sử đặt vé

Khi chọn thanh toán, sinh viên phải chọn phương thức thanh toán để tiến hành thanh toán.



Hình 3.32: Giao diện hiển thị các phương thức thanh toán khi mua vé

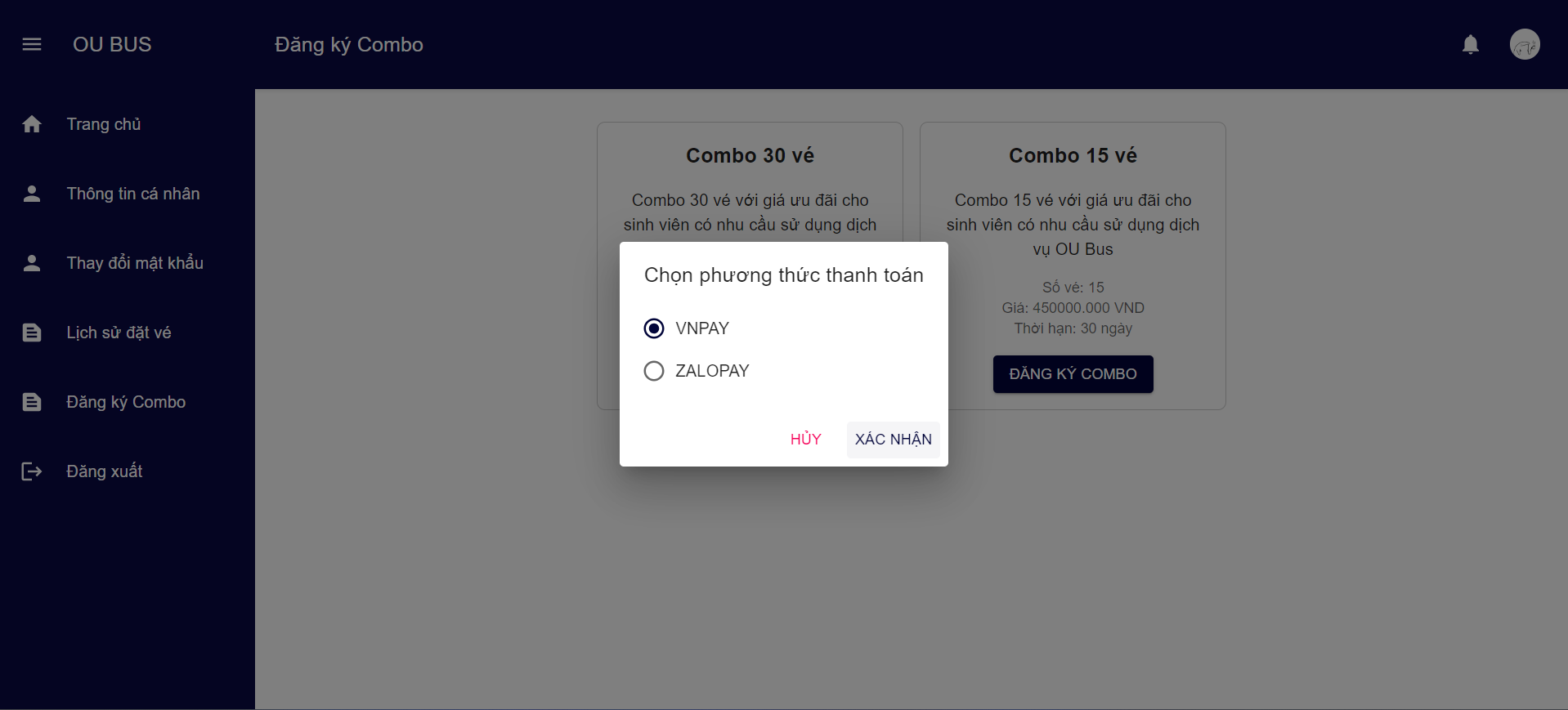
### Chức năng Đăng ký gói Combo

Khi sinh viên chọn chức năng Đăng ký Combo ở menu bên trái màn hình, nếu sinh viên hiện tại chưa đăng ký gói Combo nào thì hệ thống sẽ hiển thị danh sách các gói Combo hiện có tại hệ thống. Nếu sinh viên đã đăng ký rồi thì sẽ hiển thị thông tin gói Combo mà sinh viên đã đăng ký.

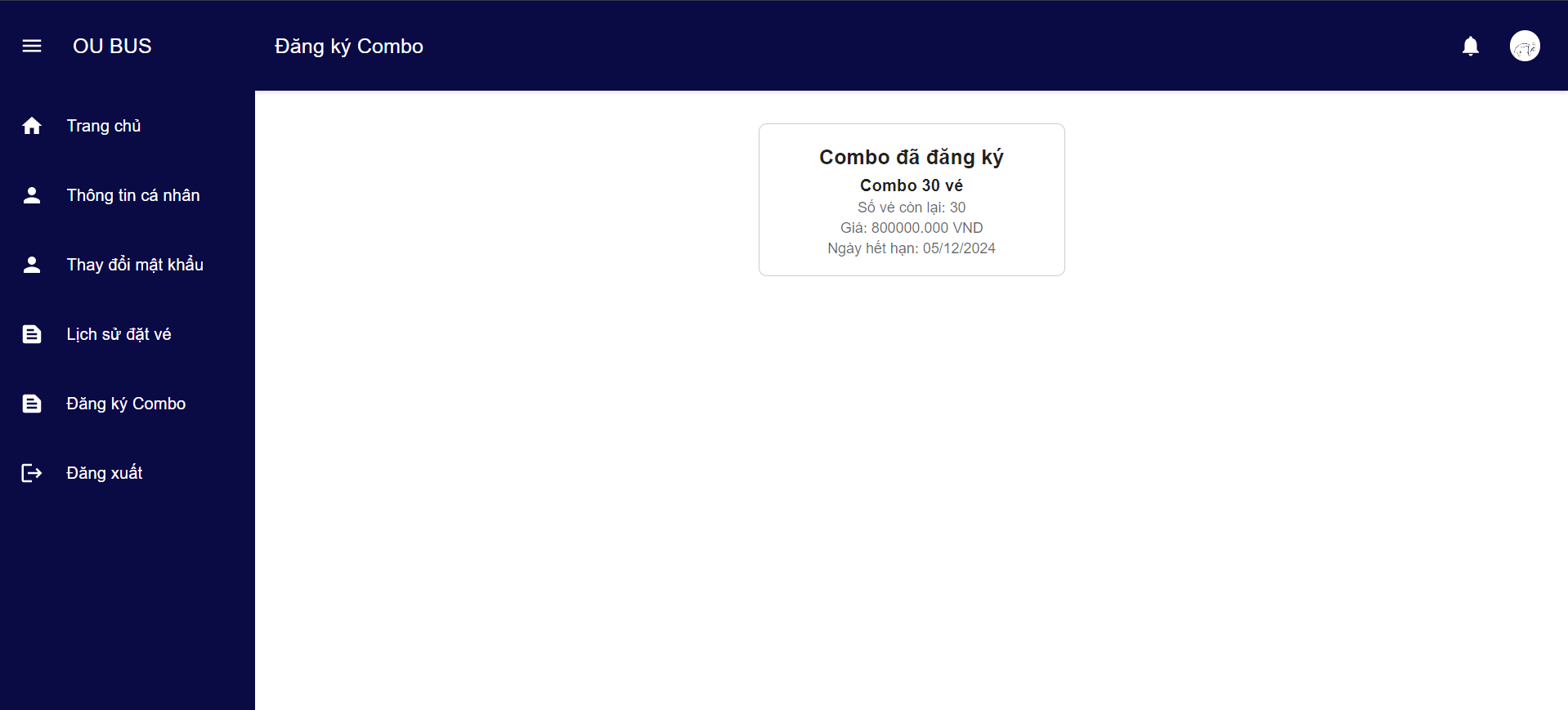


Hình 3.33: Giao diện các danh sách các gói Combo

Sinh viên nếu có thể đăng ký 1 gói Combo nếu sinh viên có mong muốn mua nhiều vé với giá ưu đãi hơn. Khi sinh viên chọn Đăng ký Combo, thì màn hình sẽ hiển thị các phương thức để sinh viên có thể thanh toán gói vé đó.



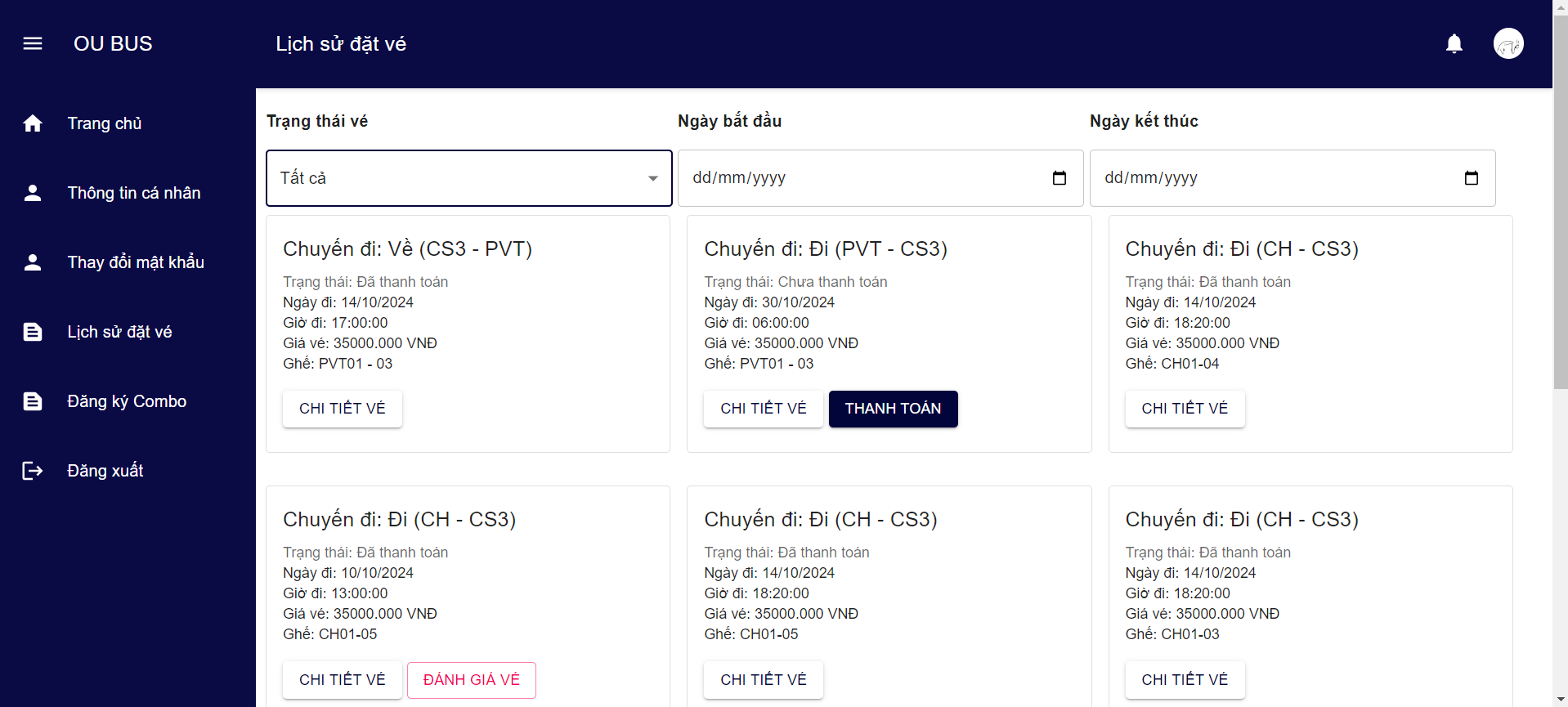
Hình 3.34: Giao diện chọn phương thức thanh toán



Hình 3.35: Giao diện hiển thị thông tin Combo được đăng ký bởi sinh viên

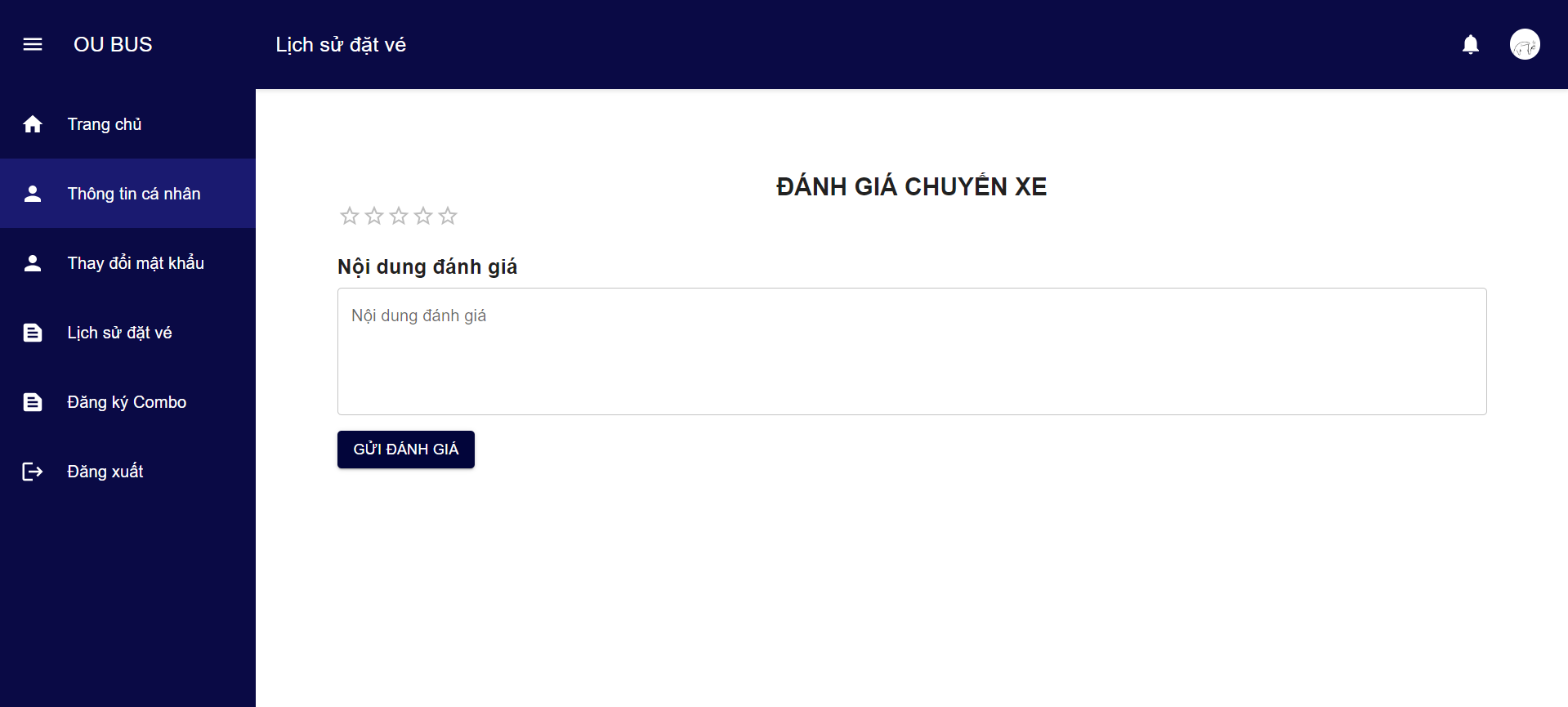
### Chức năng Đánh giá chuyến xe

Sinh viên chỉ thấy chức năng đánh giá khi mà chuyến xe mà vé của sinh viên đã đăng ký hoàn thành. Sinh viên cần chọn vào mục Lịch sử đặt vé để xem được các vé xe nào có thể đánh giá.



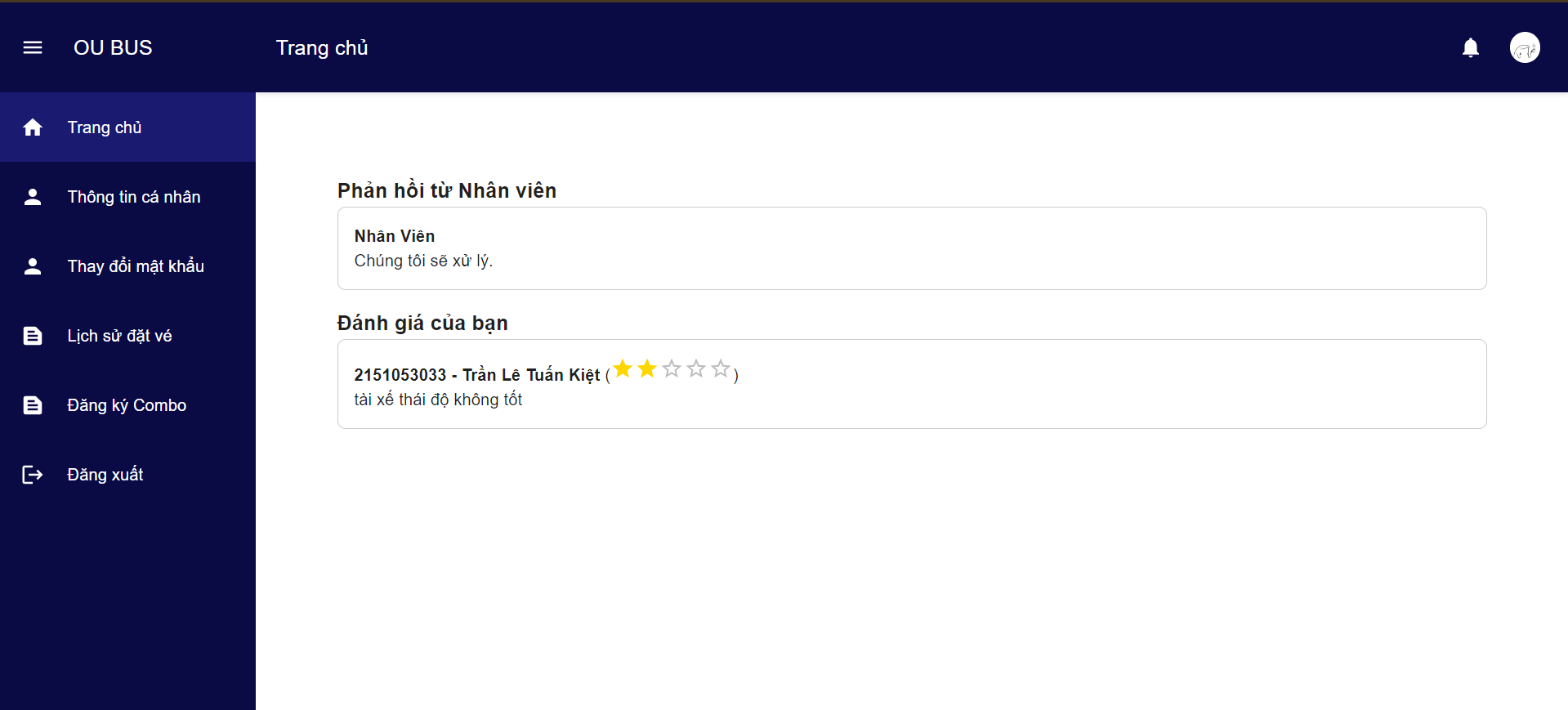
Hình 3.36: Giao diện Lịch sử đặt vé

Sinh viên chọn vào nút Đánh giá vé để thực hiện đánh giá. Nếu sinh viên chưa đánh giá cho vé xe đó thì hệ thống sẽ hiển thị ô nhập liệu cho sinh viên để sinh viên có thể thêm đánh giá.



Hình 3.37: Giao diện nhập đánh giá cho vé xe

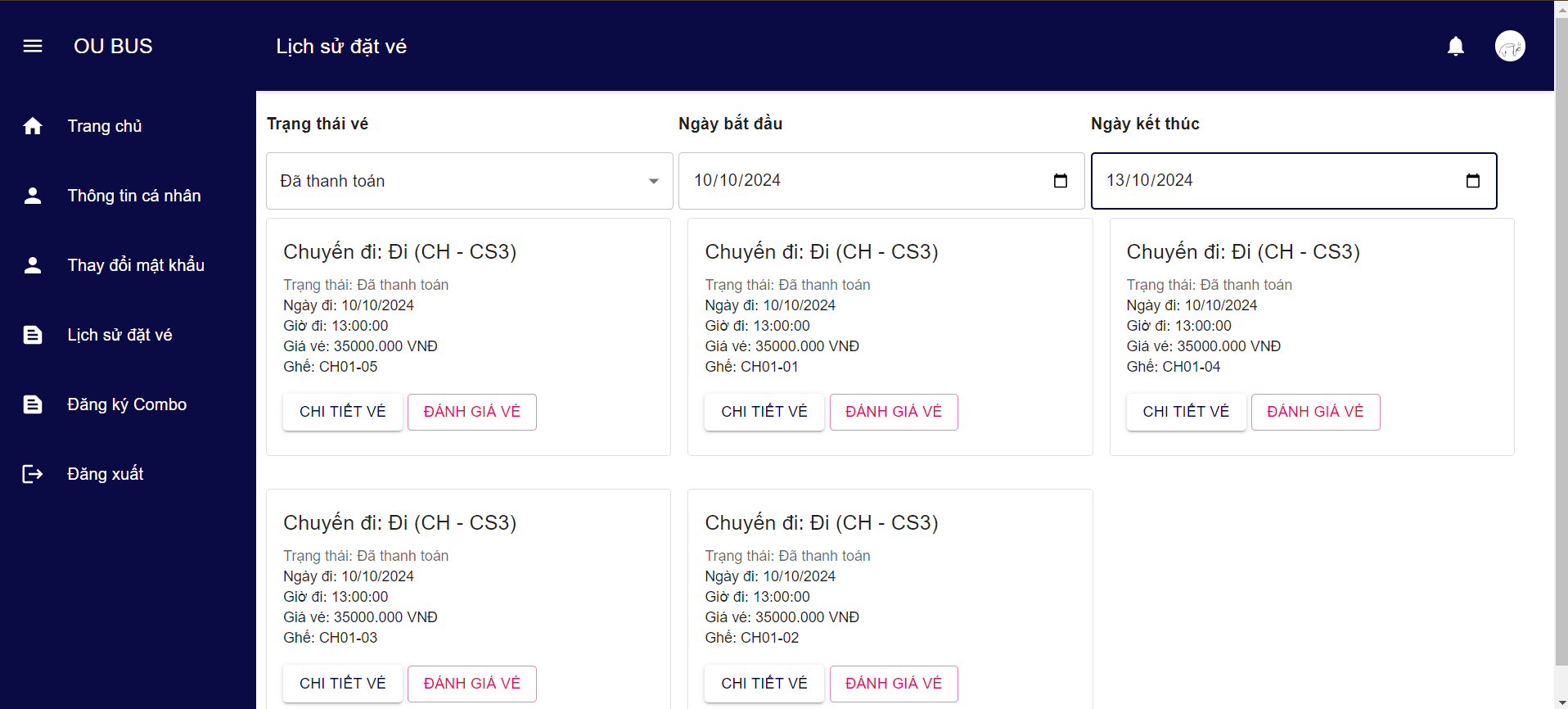
Nếu sinh viên đã thực hiện đánh giá rồi thì màn hình sẽ hiển thị nội dung của đánh giá đó và phản hồi từ phía nhân viên nếu nhân viên đã phản hồi.



Hình 3.38: Giao diện thể hiện nội dung đánh giá của sinh viên và phản hồi của nhân viên

### Chức năng Xem lịch sử đặt vé

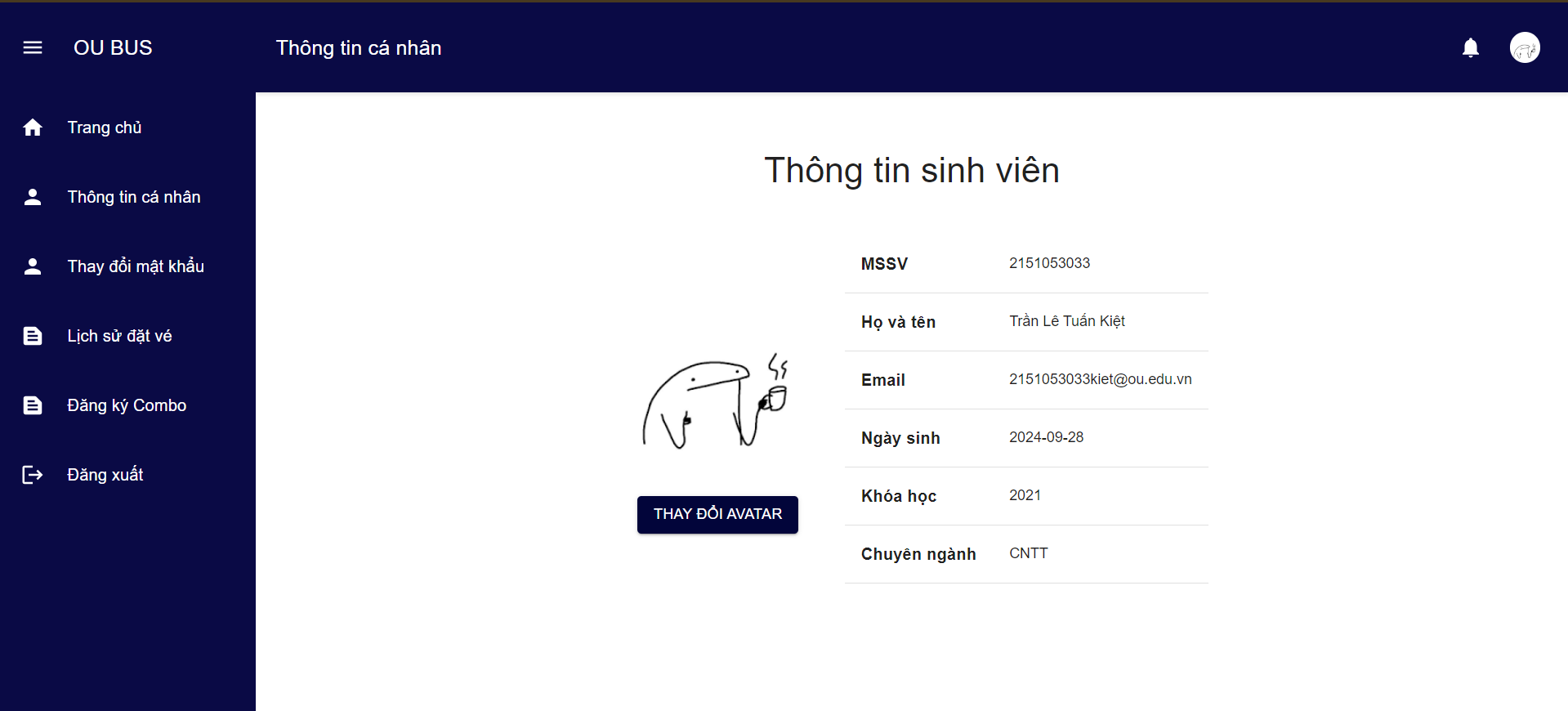
Ở chức năng Xem lịch sử đặt vé, sinh viên có thể xem được danh sách vé xe mà sinh viên đã đăng ký. Ở giao diện này, sinh viên có thể thực hiện thanh toán hoặc để lại đánh giá nếu vé của bạn thõa đủ điều kiện. Sinh viên cũng có thể thực hiện lọc các vé theo tình trạng đã thanh toán hoặc chưa thanh toán, và cũng có thể lọc các vé theo khoảng thời gian mà sinh viên chọn.



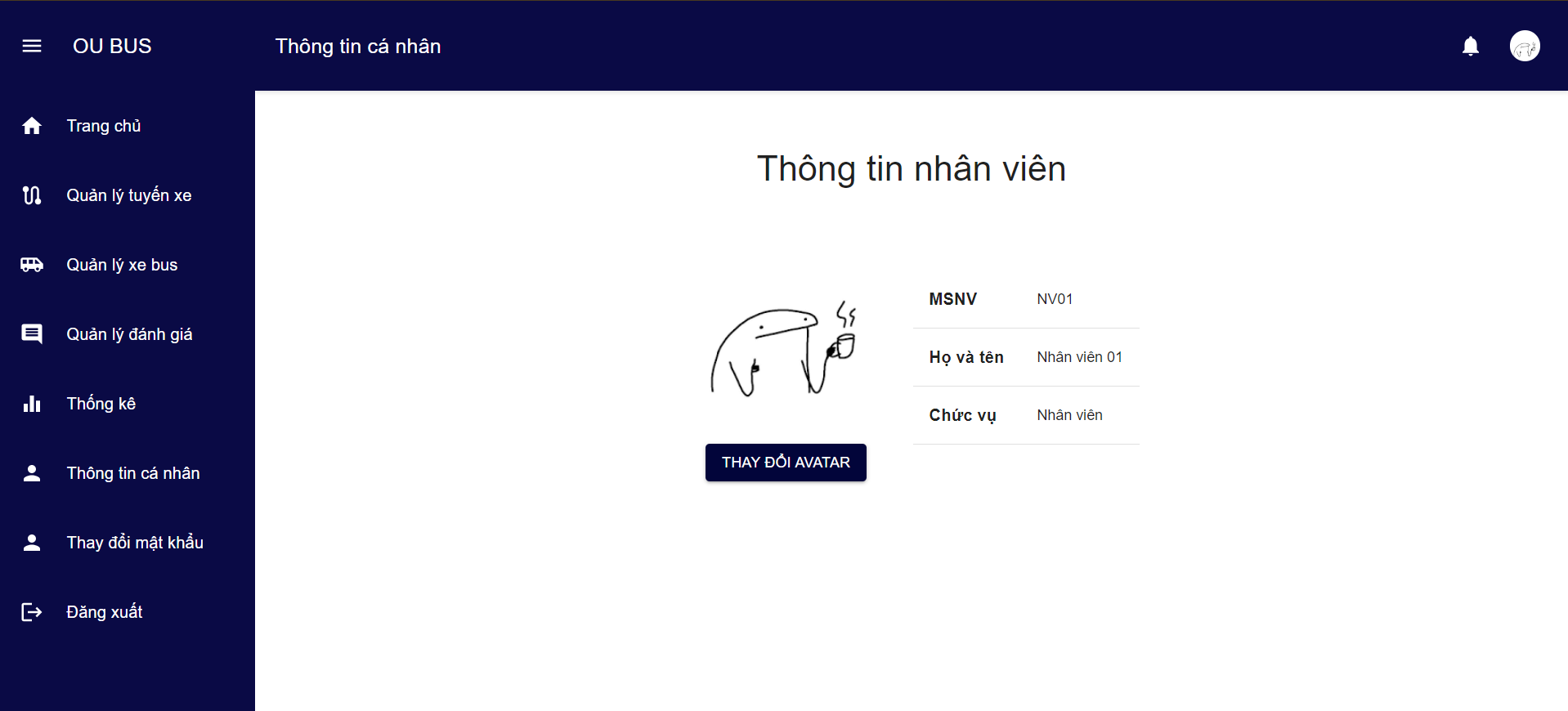
Hình 3.39: Giao diện lịch sử đặt vé

### Chức năng Quản lý thông tin cá nhân

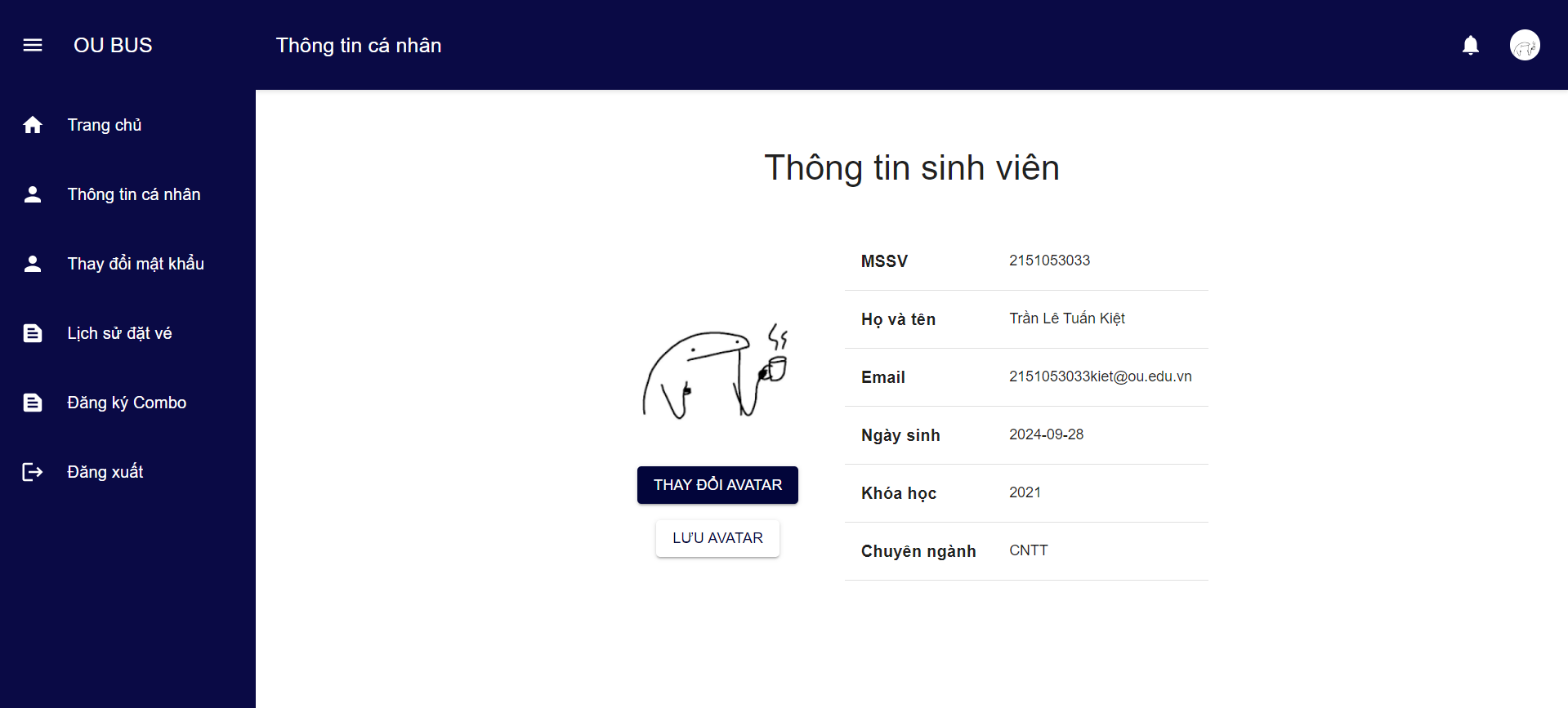
Sinh viên hoặc nhân viên có thể xem hồ sơ thông tin của bản thân tại chức năng Quản lý thông tin cá nhân. Các thông tin của người dùng được nhà trường quản lý nên người dùng không thể tự mình thay đổi. Người dùng chỉ có thể thay đổi avatar của mình bằng cách chọn nút Thay đổi avatar.



Hình 3.40: Giao diện thông tin của người dùng là sinh viên



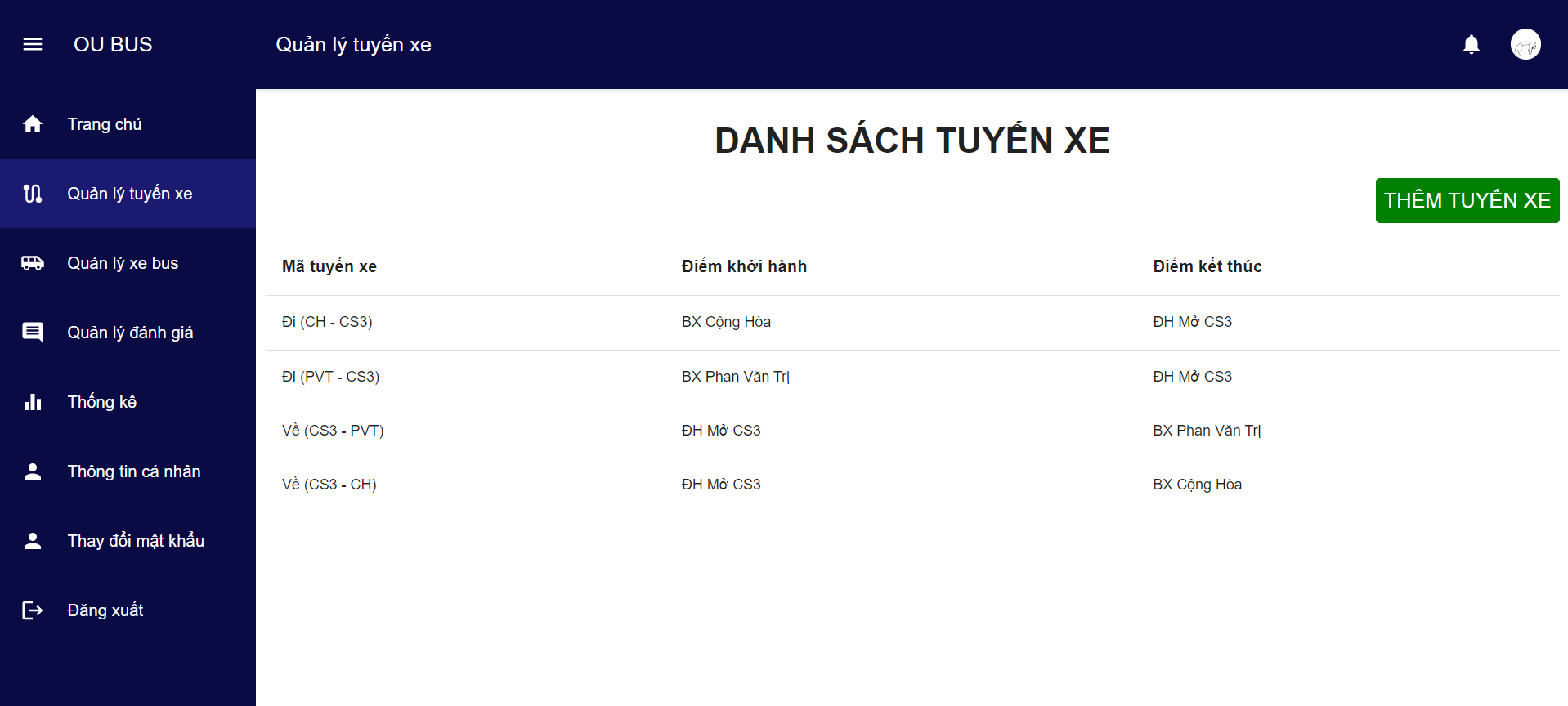
Hình 3.41: Giao diện thông tin của người dùng là nhân viên



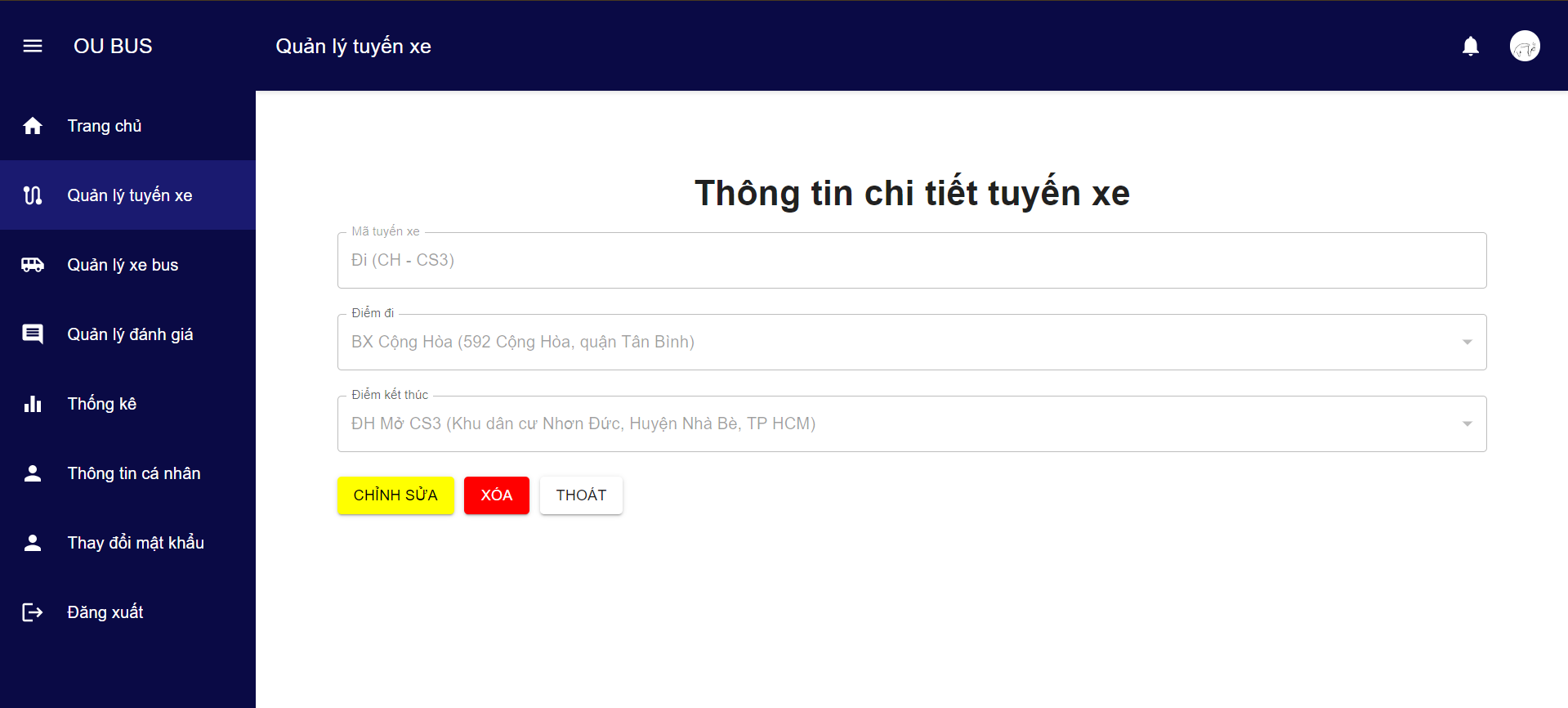
Hình 3.42: Giao diện thay đổi avatar.

### Chức năng Quản lý tuyến xe

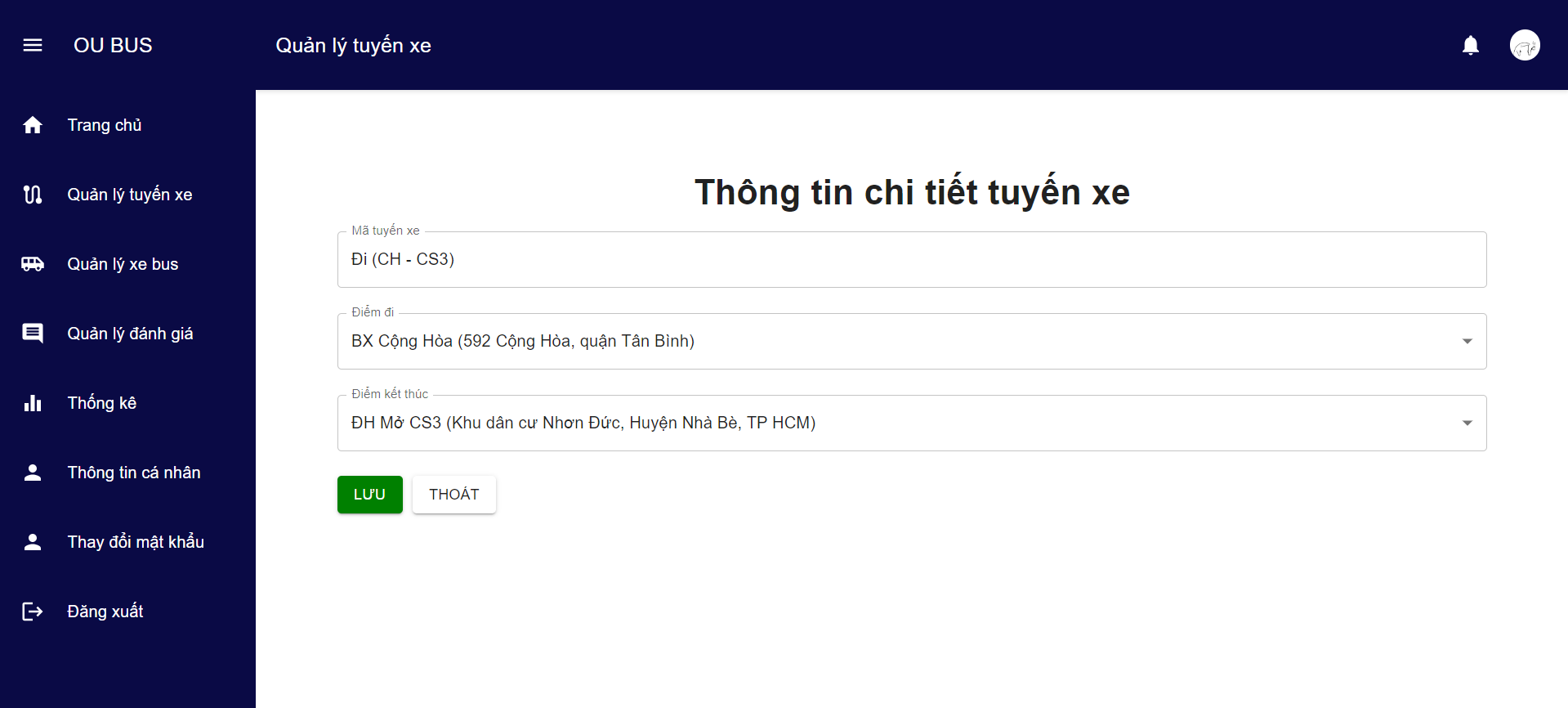
Nhân viên có trách nhiệm quản lý các tuyến xe chạy của hệ thống. Nhân viên có thể thêm tuyến xe, hoặc có thể chỉnh sửa thông tin hoặc xóa tuyến xe.



Hình 3.43: Giao diện danh sách tuyến xe



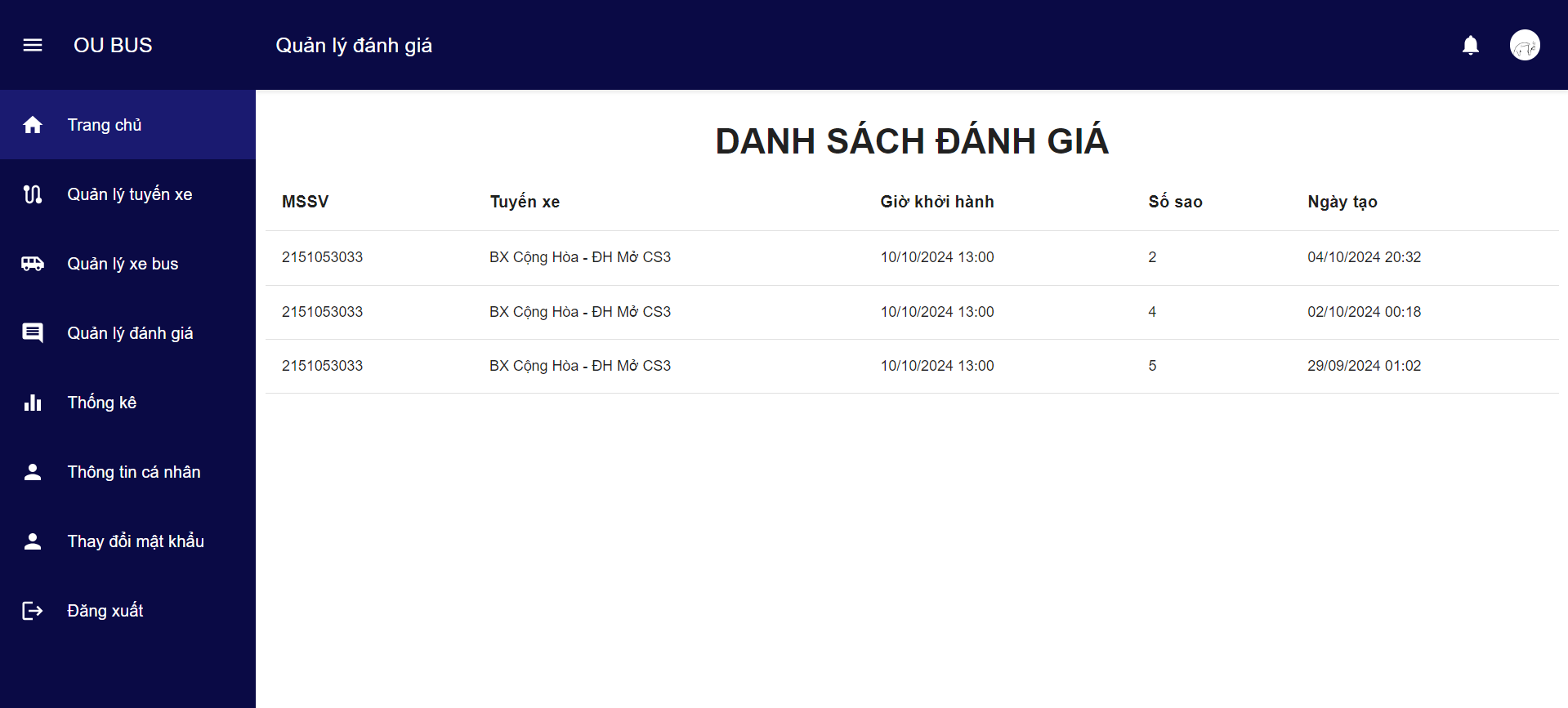
Hình 3.44: Giao diện thông tin chi tiết tuyến xe



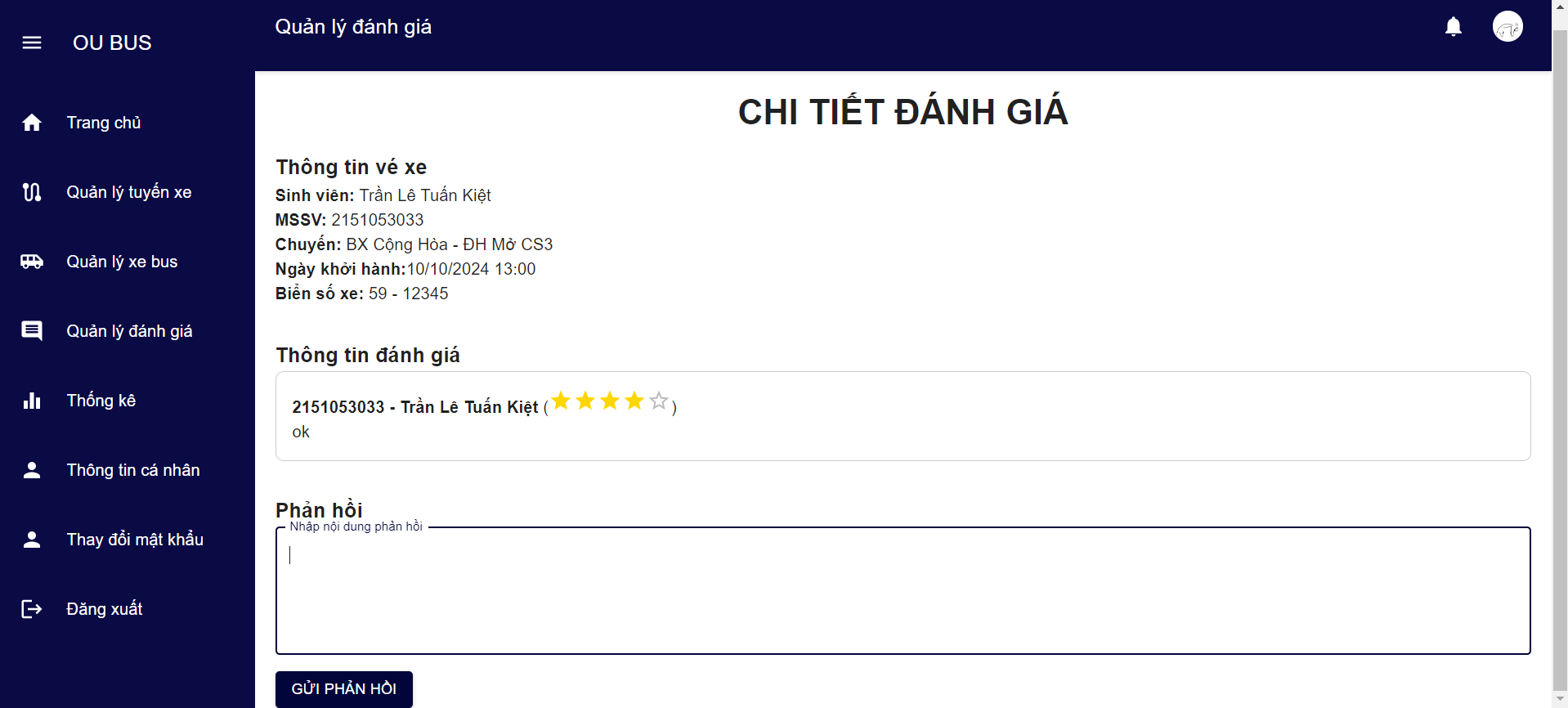
Hình 3.45: Giao diện chỉnh sửa thông tin tuyến xe

### Chức năng Quản lý đánh giá

Nhân viên có thể xem danh sách các đánh giá của các sinh viên. Bên cạnh đó, họ có thể phản hồi lại với các đánh giá của sinh viên.

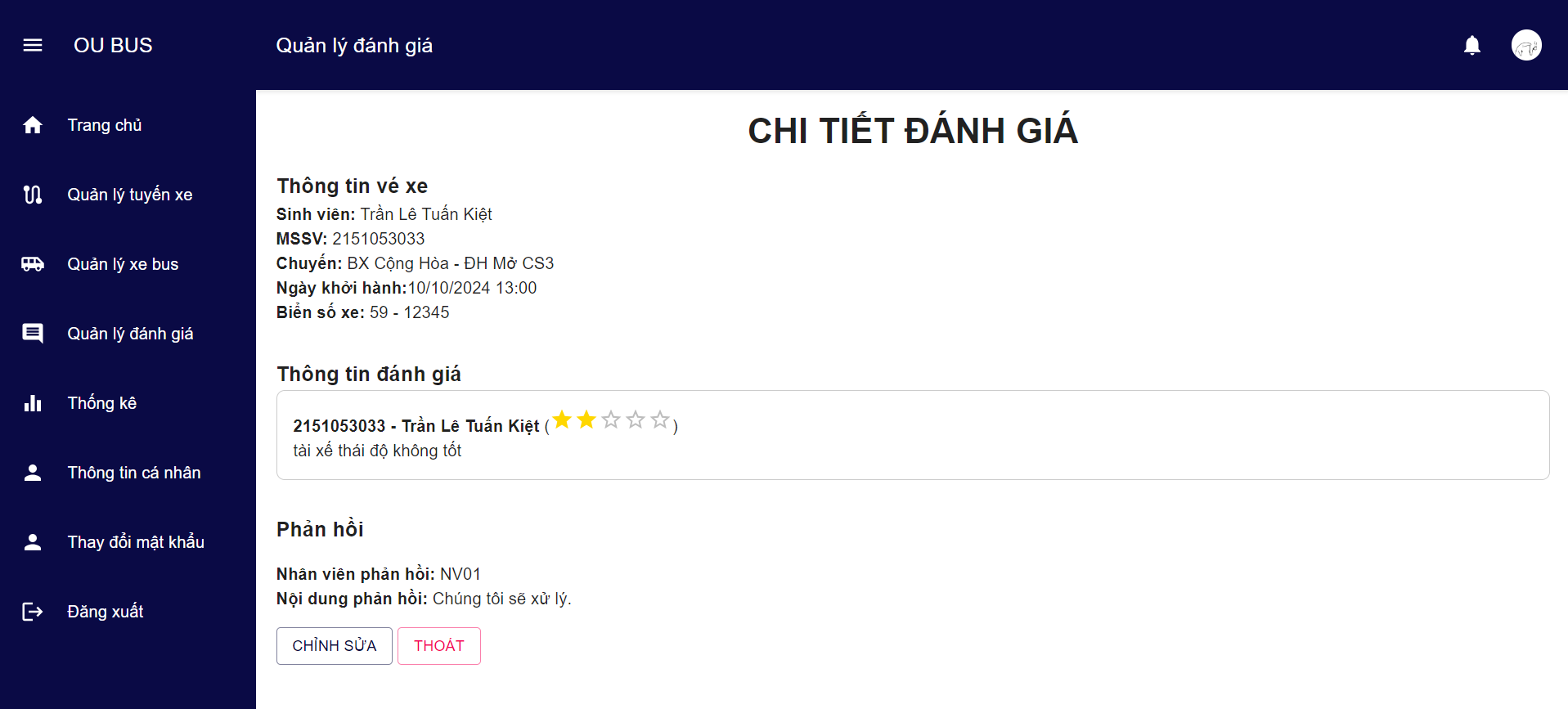


Hình 3.46: Giao diện danh sách các đánh giá của sinh viên



Hình 3.47: Giao diện chi tiết đánh giá khi nhân viên chưa phản hồi

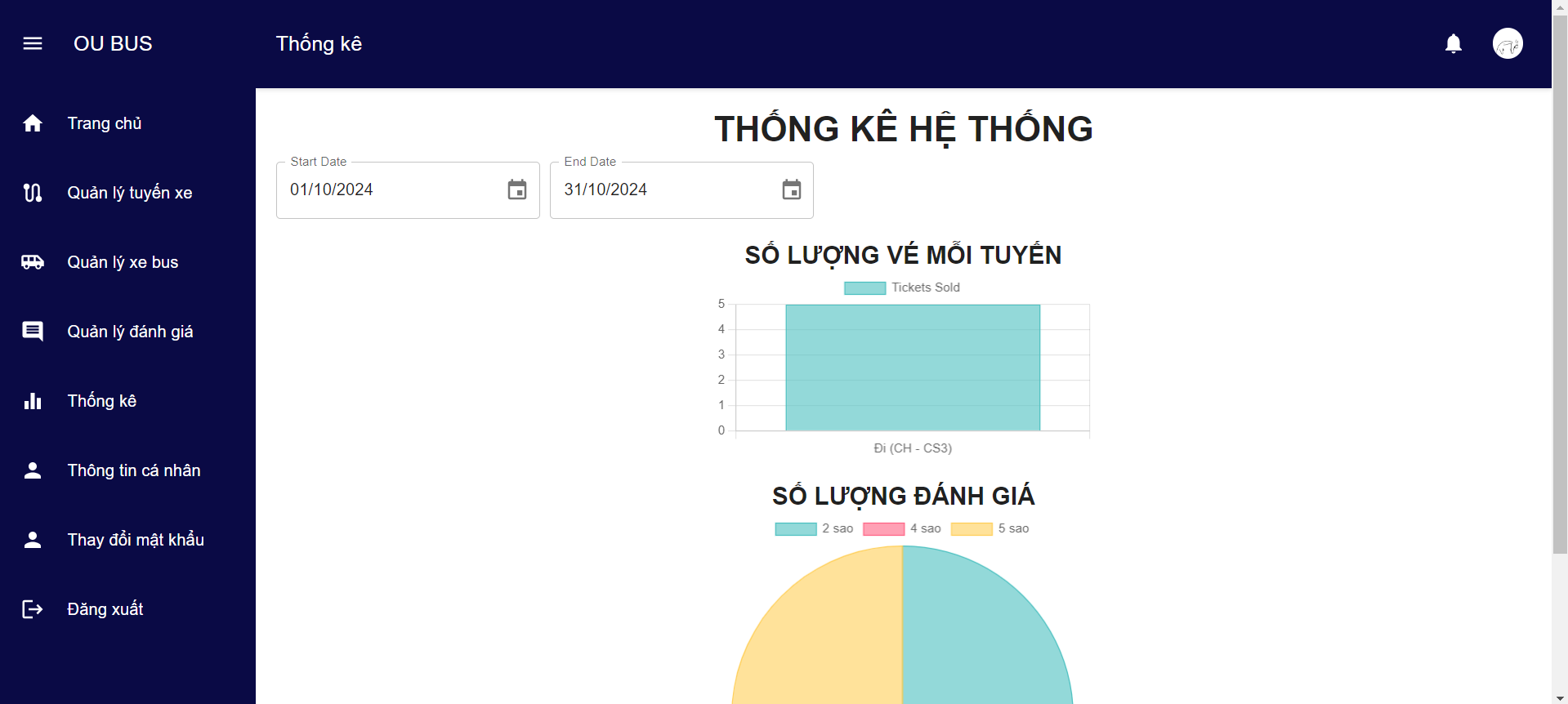
Nếu nhân viên đã phản hồi rồi thì hệ thống sẽ hiển thị nội dung của phản hồi đó.



Hình 3.48: Giao diện chi tiết đánh giá khi nhân viên đã phản hồi

### Chức năng Thống kê báo cáo

Nhân viên có thể xem được các thống kê về số lượng vé hay các lượt đánh giá từ sinh viên ở dạng biểu đồ ở chức năng Thống kê báo cáo này. Nhân viên có thể chọn khoảng thời gian mà nhân viên muốn thống kê.



Hình 3.49: Giao diện thống kê

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Sau 10 tuần triển khai và phát triển, hệ thống đã đáp ứng được các mục tiêu ban đầu, bao gồm các chức năng đăng nhập, đăng ký và phân quyền cho người dùng. Sinh viên có thể thực hiện chức năng đặt vé trực tuyến, tuy nhiên hệ thống vẫn chỉ mới hỗ trợ thanh toán qua VNPAY, nên vẫn còn hạn chế. Sau mỗi chuyến xe hoàn thành, sinh viên có thể đánh giá chuyến đi của mình, và nhân viên sẽ tiếp nhận, phản hồi các đánh giá để nhanh chóng xử lý các vấn đề nhằm nâng cao trải nghiệm dịch vụ OU Bus. Ngoài ra, nhân viên cũng có thể tham gia sử dụng hệ thống, quản lý chuyến xe, tuyến xe, và theo dõi các đánh giá từ sinh viên. Thống kê về số lượng vé và các đánh giá của sinh viên cũng được hiển thị dưới dạng biểu đồ để thêm trực quan. Người dùng bao gồm sinh viên và nhân viên có thể thực hiện thay đổi avatar hay thay đổi mật khẩu của tài khoản khi cần thiết. Sinh viên cũng có thể chat với nhân viên thông qua Firebase khi gặp khó khăn trong quá trình sử dụng dịch vụ.

Quá trình thực hiện dự án không chỉ là cơ hội để em học hỏi và hiểu biết sâu hơn về công nghệ như Django và ReactJS, mà còn giúp em có thể rèn luyện được khả năng kết hợp chúng lại trong một hệ thống thực tế. Từ việc xây dựng API, xử lý dữ liệu phức tạp ở backend ở Django, đến việc phát triển giao diện người dùng trực quan, mượt mà với ReactJS. Ngoài ra, việc triển khai dự án trên môi trường thực tế như PythonAnywhere, quản lý cơ sở dữ liệu với MySQL, và xử lý hình ảnh thông qua Cloudinary đã mang đến cho em những kinh nghiệm quý báu trong việc quản lý và duy trì một ứng dụng web hoàn chỉnh.

Bên cạnh đó, do khả năng quản lý thời gian còn chưa tốt nên hệ thống vẫn còn một số hạn chế. Giao diện của hệ thống chưa thực sự là tối ưu và thân thiện với người dùng. Tính năng thanh toán trực tuyến qua ví điện tử ZaloPay chưa được tích hợp, khiến chức năng đặt vé trực tuyến chưa thật sự hoàn thiện. Hệ thống có hỗ trợ chat trực tuyến, tuy nhiên vẫn còn đơn giản, điều đó làm giảm khả năng tương tác một cách trọn vẹn giữa sinh viên và nhân viên khi phát sinh vấn đề.

Những điểm chưa hoàn thiện này sẽ là mục tiêu tiếp theo để nâng cao hệ thống trong các giai đoạn phát triển sắp tới.

## Hướng phát triển

Để hoàn thiệc chức năng đặt vé trực tuyến, hệ thống cần tích hợp các cổng thanh toán phổ biến như ZaloPay hay các phương thức thanh toán khác. Việc này sẽ giúp sinh viên dễ dàng thanh toán trực tiếp ngay trên hệ thống hoặc mở rộng, nâng cao trải nghiệm của người dùng và tạo sự thuận tiện trong quá trình mua vé.

Hiện tại, giao diện hệ thống chưa được tối ưu hóa về mặt trải nghiệm và thẩm mỹ. Việc tối ưu và nâng cao giao diện sẽ giúp hệ thống trở nên thân thiện hơn với người dùng, đồng thời tăng sự tương tác và hài lòng khi sử dụng dịch vụ.

Để giúp hệ thống nâng cao tính tương tác và hỗ trợ người dùng một cách nhanh chóng, kịp thời, trọn vẹn, việc phát triển chức năng chat trực tuyến giữa sinh viên và nhân viên một cách chỉnh chu hơn là điều cần thiết.

Hệ thống cần tích hợp thêm công cụ ước lượng được khoảng cách, để khi người dùng đưa ra vị trí của họ thì hệ thống có thể đưa ra đề xuất bến xe gần nhất, tạo sự thuận tiện trong trải nghiệm sử dụng dịch vụ của sinh viên.

Và phát triển thêm một phiên bản ứng dụng di động sẽ có thể tăng tính tiện lợi, tạo điều kiện cho người dùng dễ dàng tiếp cận và sử dụng dịch vụ mọi lúc, mọi nơi.

TÀI LIỆU THAM KHẢO

[1] Adrian Holovaty và Jacob Kaplan-Moss, ‘Chapter 1: Introduction to Django’, trong *The Django Book*, USA: Apress, 2009.

[2] Ironhack Việt Nam, ‘Django là gì? Cách hiệu quả nhất để học Django Python là gì?’, 18/06/2021. [Trực tuyến]. Địa chỉ: <https://ironhackvietnam.edu.vn/django-la-gi/>. [Truy cập 30/08/2024].

[3] Chris Minnick, ‘Introduction’, trong *ReactJS Foundations Building User Interfaces with ReactJS,* Canada: John Willy & Sons, 2022, xxvii.

PHỤ LỤC

Bảng User

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | Int | Not | - | Khóa chính |
| username | varchar (50) | Not | Tên tài khoản | - |
| password | varchar (50) | Not | Mật khẩu | - |
| avatar | varchar (255) | - | Avatar người dùng | - |
| role | varchar (50) | Not | Vai trò người dùng | - |

Bảng Student

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| student\_code | varchar (10) | Not | Mã số sinh viên | - |
| birth | date | Not | Ngày sinh | - |
| year | varchar (10) | - | Năm học | - |
| major | varchar (50) | Not | Chuyên ngành | - |
| fullname | varchar (100) | Not | Họ tên sinh viên | - |
| email | varchar (254) | Not | Email của sinh viên | - |
| user\_id | int | - | Mã tài khoản | Khóa ngoại (1-1) |

Bảng Staff

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| staff\_code | varchar (10) | Not | Mã số nhân viên | - |
| position | varchar (100) | Not | Vị trí | - |
| fullname | varchar (100) | Not | Họ tên nhân viên | - |
| user\_id | int | - | Mã tài khoản | Khóa ngoại (1-1) |

Bảng Station

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| name | varchar (100) | Not | Tên bến xe | - |
| address | varchar (100) | Not | Địa chỉ bến xe | - |
| image | varchar (255) | - | Hình ảnh bến xe | - |

Bảng Bus

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| license\_plate | varchar (20) | Not | Biển số xe | - |
| name | varchar (20) | Not | Tên xe bus | - |
| image | varchar (255) | - | Hình ảnh xe bus | - |
| seat\_number | int | - | Số ghế trên xe | - |
| available | boolean | - | Trạng thái xe | - |
| station\_id | int | - | Mã bến xe | Khóa ngoại |

Bảng Route

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| route\_code | varchar (20) | Not | Mã tuyến xe | - |
| starting\_point | int | Not | Điểm đi | Khóa ngoại |
| ending\_point | int | Not | Điểm đến | Khóa ngoại |

Bảng Seat

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| code | varchar (10) | Not | Mã ghế | - |
| bus\_id | int | Not | Tên xe bus | Khóa ngoại |

Bảng Driver

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| id\_number | varchar (12) | Not | Số căn cước tài xế | - |
| name | varchar (20) | Not | Tên tài xế | - |
| phone | varchar (10) | Not | Số điện thoại | - |
| avatar | varchar (255) | - | Hình ảnh tài xế | - |

Bảng BusTrip

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| start\_time | datetime | Not | Thời gian khởi hành | - |
| end\_time | datetime | Not | Thời gian kết thúc | - |
| available\_seat | int | - | Số chỗ ngồi trống | - |
| route | int | Not | Tuyến xe chạy | Khóa ngoại |
| bus | int | Not | Xe bus | Khóa ngoại |
| driver | int | Not | Tài xế | Khóa ngoại |
| trip\_status | varchar (50) | - | Trạng thái chuyến xe | - |

Bảng SeatBustrip

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| seat | int | Not | Mã ghế | Khóa ngoại |
| trip | Int | Not | Mã chuyến xe | Khóa ngoại |
| available | boolean | - | Trạng thái chỗ ngồi | - |

Bảng Quotation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| route | int | Not | Mã tuyến | Khóa ngoại |
| price | decimal (13, 3) | Not | Giá vé |  |

Bảng Ticket

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| student | int | Not | Mã sinh viên | Khóa ngoại |
| staff | int | Not | Mã nhân viên | Khóa ngoại |
| seat\_bustrip | int | Not | Mã ghế | Khóa ngoại |
| price | decimal (13, 3) | Not | Giá vé | - |
| expiration\_date | datetime | Not | Ngày hết hạn | - |

Bảng Combo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| combo\_code | varchar (10) | Not | Mã combo | - |
| name | varchar (100) | Not | Tên gói Combo | - |
| description | varchar (255) | Not | Mô tả Combo | - |
| number\_of\_tickets | int | Not | Số vé | - |
| duration | int | Not | Thời hạn sử dụng | - |
| price | decimal (13, 3) | Not | Giá vé combo | - |

Bảng StudentCombo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| student | int | Not | Mã sinh viên | Khóa ngoại |
| combo | int | Not | Mã combo | Khóa ngoại |
| remaining\_ticket | int | Not | Số vé còn lại |  |
| expiration\_date | datetime | Not | Ngày hết hạn |  |
| is\_active | boolean | Not | Trạng thái |  |

Bảng Review

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| user | int | Not | Mã người dùng | Khóa ngoại |
| ticket | int | Not | Mã vé | Khóa ngoại |
| content | varchar (255) | Not | Nội dung đánh giá |  |
| rating | int | Not | Số sao đánh giá |  |
| parent | int | Not | Mã phản hồi | Khóa ngoại |

Bảng PayMethod

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| pay\_code | varchar (100) | Not | Mã phương thức | - |
| name | varchar (100) | Not | Tên phương thức | - |

Bảng Invoice

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Type | Null | Chú thích | Khóa |
| id | int | Not | - | Khóa chính |
| ticket | int | Not | Mã vé của sinh viên | Khóa ngoại |
| student\_combo | int | Not | Mã combo của sinh viên | Khóa ngoại |
| payment\_method | int | Not | Mã phương thức thanh toán | Khóa ngoại |
| total\_price | decimal (13,3) | Not | Ngày hết hạn |  |