**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ ĐÔNG Á**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI TẬP LỚN**

**HỌC PHẦN: ĐỒ ÁN CHUYÊN NGÀNH CNPM**

**CHỦ ĐỀ 1: XÂY DỰNG WEBSITE, WEB APPLICATION**

**TÊN ĐỀ TÀI 47: XÂY DỰNG WEBSITE KINH DOANH DỊCH**

**VỤ HOSTING**

**LỚP TÍN CHỈ: ĐACNPM.03.K13.10.LH.C04.1\_LT**

**Giảng viên hướng dẫn: ThS Mai Văn Linh**

**Danh sách sinh viên thực hiện: Nhóm 19**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Mã sinh viên** | **Sinh viên thực hiện** | **Lớp hành chính** |
| **1** | 20223155 | Nguyễn Trí Dũng | DCCNTT 13.10.16 |
| **2** | 20222996 | Trần Văn Nam | DCCNTT 13.10.16 |
| **3** | 20223110 | Nguyễn Văn Duy | DCCNTT 13.10.16 |

**Bắc Ninh - 2025**

**MỤC LỤC**

[DANH MỤC TỪ VIẾT TẮT iii](#_Toc27410)

[DANH MỤC HÌNH VẼ v](#_Toc19580)

[DANH MỤC BẢNG BIỂU viii](#_Toc31918)

[MỞ ĐẦU 1](#_Toc1255)

[CHƯƠNG I: CƠ SỞ LÝ THUYẾT 4](#_Toc21990)

[1.1. Tổng quan về lĩnh vực Web 4](#_Toc24311)

[1.1.1. Tổng quan 4](#_Toc15363)

[1.1.2. Bối cảnh phát triển 4](#_Toc14037)

[1.1.3. Tương lai của công nghệ web 5](#_Toc24693)

[1.1.4. Ưu điểm của hệ thống web 5](#_Toc25986)

[1.1.5. Hạn chế của hệ thống web 5](#_Toc8195)

[1.2. Tổng quan về công nghệ 6](#_Toc25909)

[1.2.1. Công nghệ thiết kế web 6](#_Toc621)

[1.2.2. Ngôn ngữ lập trình – PHP 7](#_Toc17290)

[1.2.3. Công nghệ cơ sở dữ liệu – MySQL 7](#_Toc22610)

[1.2.4. Các công nghệ nền tảng hỗ trợ khác 8](#_Toc27214)

[1.2.5. Đánh giá chung 14](#_Toc31052)

[CHƯƠNG II: THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH 15](#_Toc7086)

[2.1 Giới thiệu bài toán 15](#_Toc11240)

[2.1.1 Phát biểu bài toán 15](#_Toc9418)

[2.1.2. Danh sách tác nhân (Actor) 19](#_Toc20069)

[2.1.3 Danh sách các ca sử dụng (use case) 21](#_Toc8703)

[2.2 Thiết kế hệ thống 24](#_Toc20062)

[2.2.1 Biểu đồ Use case 24](#_Toc19047)

[2.2.2 Biểu đồ hoạt động 92](#_Toc11979)

[2.2.3 Biểu đồ trạng thái 99](#_Toc9304)

[2.2.4 Cấu trúc các bảng Database 104](#_Toc5880)

[2.2.5 Biểu đồ lớp 109](#_Toc21231)

[CHƯƠNG III: GIAO DIỆN HỆ THỐNG 110](#_Toc31344)

[3.1 Giao diện người dùng 110](#_Toc27886)

[3.2 Giao diện quản trị viên 114](#_Toc10467)

[CHƯƠNG IV: KIỂM THỬ HỆ THỐNG 119](#_Toc3946)

[4.1 Giới thiệu kiểm thử 119](#_Toc20110)

[4.1.1 Lý thuyết về kiểm thử hệ thống 119](#_Toc13095)

[4.1.2 Các loại kiểm thử hệ thống 120](#_Toc28859)

[4.2 Quy trình thực hiện 121](#_Toc16786)

[4.3 Thiết kế các Test-case 124](#_Toc3910)

[4.3.1 Test case theo module Đăng ký 124](#_Toc3760)

[4.3.2 Test case theo module Đăng nhập 128](#_Toc6993)

[4.3.3 Test case theo module Đổi mật khẩu 132](#_Toc18311)

[4.3.4 Test case theo module Mua hàng & Thanh toán 135](#_Toc13448)

[4.3.5 Test case theo module Quản lý hosting (Thêm, Sửa, Xóa) 140](#_Toc8394)

[4.3.6 Test case theo module Quản lý thành viên (Thêm, Sửa, Xóa) 146](#_Toc27050)

[KẾT LUẬN 152](#_Toc12228)

[TÀI LIỆU THAM KHẢO 154](#_Toc24114)

# DANH MỤC TỪ VIẾT TẮT

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **TỪ VIẾT TẮT** | **TÊN ĐẦY ĐỦ** | **GIẢI THÍCH** |
| **1** | API | Application Programming Interface | Giao diện lập trình ứng dụng, cho phép các phần mềm giao tiếp với nhau. |
| **2** | CMS | Content Management System | Hệ thống quản trị nội dung (ví dụ: WordPress). |
| **3** | CNPM | Công Nghệ Phần Mềm | Tên chuyên ngành. |
| **4** | CSS | Cascading Style Sheets | Ngôn ngữ định dạng kiểu cho trang web. |
| **5** | DNS | Domain Name System | Hệ thống phân giải tên miền. |
| **6** | FTP | File Transfer Protocol | Giao thức truyền tệp tin. |
| **7** | HTML | HyperText Markup Language | Ngôn ngữ đánh dấu siêu văn bản để tạo trang web. |
| **8** | HTTP/HTTPS | HyperText Transfer Protocol (Secure) | Giao thức truyền tải siêu văn bản (Phiên bản bảo mật). |
| **9** | IP | Internet Protocol | Giao thức Internet, định danh địa chỉ máy chủ. |
| **10** | JS | JavaScript | Ngôn ngữ lập trình kịch bản cho web. |
| **11** | PHP | PHP: Hypertext Preprocessor | Ngôn ngữ lập trình kịch bản phía máy chủ. |
| **12** | SQL | Structured Query Language | Ngôn ngữ truy vấn có cấu trúc cho cơ sở dữ liệu. |
| **13** | SSD | Solid-State Drive | Ổ đĩa thể rắn, công nghệ lưu trữ cho máy chủ. |
| **14** | SSL/TLS | Secure Sockets Layer / Transport Layer Security | Giao thức bảo mật, mã hóa kết nối giữa máy chủ và trình duyệt. |
| **15** | UI/UX | User Interface / User Experience | Giao diện người dùng / Trải nghiệm người dùng. |
| **16** | VPS | Virtual Private Server | Máy chủ riêng ảo. |
| **17** | WWW | World Wide Web | Mạng lưới thông tin toàn cầu. |
| **18** | WHM/cPanel | Web Host Manager / control Panel | Bảng điều khiển quản lý hosting phổ biến. |

# DANH MỤC HÌNH VẼ

[Hình 1. 1: Công nghệ web 4](#_Toc3810)

[Hình 1. 2: Ngôn ngữ PHP 7](#_Toc18283)

[Hình 1. 3: Cơ sở dữ liệu - MySQL 8](#_Toc5317)

[Hình 1. 4: Logo ứng dụng postman 9](#_Toc20812)

[Hình 1. 5: Logo ứng dụng VSCode 9](#_Toc20902)

[Hình 1. 6: Logo ứng dụng laragon 10](#_Toc950)

[Hình 1. 7: Logo thư viện bootstrap 11](#_Toc5467)

[Hình 1. 8: Logo thư viện Font Awesome 11](#_Toc25951)

[Hình 1. 9: Logo thư viện jQuery 12](#_Toc12586)

[Hình 1. 10: Logo thư viện SweetAlert 12](#_Toc31284)

[Hình 1. 11: Logo thư viện Ajax 13](#_Toc14854)

[Hình 1. 12: Logo thư viện Dastone 14](#_Toc2349)

[Hình 2. 1: Phân cấp chức năng 17](#_Toc4497)

[Hình 2. 2: Use case tổng quát 24](#_Toc5167)

[Hình 2. 3: Use case đăng nhập 24](#_Toc5707)

[Hình 2. 4: Use case Tài khoản & Bảo mật 30](#_Toc7290)

[Hình 2. 5: Use case Mua hàng & Thanh toán 37](#_Toc1401)

[Hình 2. 6: Use case quản lý dịch vụ đã mua 44](#_Toc29935)

[Hình 2. 7: Use case ticket hỗ trợ 50](#_Toc23611)

[Hình 2. 8: Use case lịch sự hoạt động 54](#_Toc3185)

[Hình 2. 9: Use case quản lý thành viên 58](#_Toc15840)

[Hình 2. 10: Use case quản lý hosting 64](#_Toc21769)

[Hình 2. 11: Use case Thống kê & Báo cáo 69](#_Toc9859)

[Hình 2. 12: Use case quản lý mã giảm giá 76](#_Toc31441)

[Hình 2. 13: Use case quản lý bài viết 80](#_Toc24929)

[Hình 2. 14: Use case quản lý hỗ trợ (ticket) 86](#_Toc13413)

[Hình 2. 15: Sơ đồ trạng thái chức năng Thêm bài viết 99](#_Toc16957)

[Hình 2. 16: Sơ đồ trạng thái chức năng Sửa bài viết 99](#_Toc30044)

[Hình 2. 17: Sơ đồ trạng thái chức năng Xóa bài viết 100](#_Toc11082)

[Hình 2. 18: Sơ đồ trạng thái chức năng Thêm mã giảm giá 100](#_Toc17300)

[Hình 2. 19: Sơ đồ trạng thái chức năng Sửa mã giảm giá 101](#_Toc5288)

[Hình 2. 20: Sơ đồ trạng thái chức năng Xóa mã giảm giá 101](#_Toc15106)

[Hình 2. 21: Sơ đồ trạng thái chức năng Thêm thành viên 102](#_Toc1248)

[Hình 2. 22: Sơ đồ trạng thái chức năng Sửa thành viên 102](#_Toc4978)

[Hình 2. 23: Sơ đồ trạng thái chức năng Xóa thành viên 103](#_Toc31337)

[Hình 2. 24: Sơ đồ trạng thái chức năng Thêm hosting 103](#_Toc18416)

[Hình 2. 25: Sơ đồ trạng thái chức năng Sửa hosting 104](#_Toc26821)

[Hình 2. 26: Sơ đồ trạng thái chức năng Xóa hosting 104](#_Toc30712)

[Hình 2. 27: Biểu đồ lớp 109](#_Toc9881)

[Hình 3. 1: Giao diện đăng ký hệ thống 110](#_Toc22484)

[Hình 3. 2: Giao diện đăng nhập hệ thống 110](#_Toc2657)

[Hình 3. 3: GIao diện thông tin cá nhân 111](#_Toc29550)

[Hình 3. 4: Giao diện bán hosting 111](#_Toc31702)

[Hình 3. 5: Giao diện thanh toán đơn hàng 112](#_Toc1763)

[Hình 3. 6: Giao diện đơn đặt hàng 112](#_Toc27682)

[Hình 3. 7: Giao diện hóa đơn 113](#_Toc4001)

[Hình 3. 8: Giao diện 1 bài viết cụ thể 113](#_Toc14370)

[Hình 3. 9: Lịch sử hoạt động 114](#_Toc8085)

[Hình 3. 10: Giao diện quản lý thành viên 114](#_Toc14472)

[Hình 3. 11: Giao diện câp nhật thông tin thành viên 115](#_Toc26939)

[Hình 3. 12: Giao diện quản lý hosting 115](#_Toc26666)

[Hình 3. 13: Giao diện thêm hosting mới 116](#_Toc27957)

[Hình 3. 14: Giao diện sửa thông tin hosting 116](#_Toc31169)

[Hình 3. 15: Giao diện xóa hosting 117](#_Toc23944)

[Hình 3. 16: Giao diện quản lý mã giảm giá 117](#_Toc16544)

[Hình 3. 17: Giao diện thêm mã giảm giá 118](#_Toc11260)

[Hình 3. 18: Giao diện sửa mã giảm giá 118](#_Toc11312)

[Hình 3. 19: Giao diện xóa mã giảm giá 119](#_Toc9257)

# **DANH MỤC BẢNG BIỂU**

[Bảng 2. 1: Danh sách các ca sử dụng 21](#_Toc21388)

[Bảng 2. 2: UC01 – Đăng nhập 25](#_Toc16457)

[Bảng 2. 3: Luồng sự kiện phụ UC01 26](#_Toc25704)

[Bảng 2. 4: UC02 – Đăng ký 27](#_Toc6490)

[Bảng 2. 5: Luồng sự kiện phụ UC02 29](#_Toc25479)

[Bảng 2. 6: UC03 - Đổi mật khẩu 30](#_Toc15962)

[Bảng 2. 7: Luồng sự kiện phụ UC03 32](#_Toc329)

[Bảng 2. 8: UC04 - Cập nhật thông tin cá nhân 32](#_Toc21163)

[Bảng 2. 9: Luồng sự kiện phụ UC04 34](#_Toc18468)

[Bảng 2. 10: UC05 - Thiết lập bảo mật 35](#_Toc15816)

[Bảng 2. 11: Luồng sự kiện phụ UC05 36](#_Toc24672)

[Bảng 2. 12: UC06 - Chọn gói dịch vụ 38](#_Toc29161)

[Bảng 2. 13: Luồng sự kiện phụ UC06 39](#_Toc10356)

[Bảng 2. 14: UC07 - Nhập mã giảm giá 39](#_Toc9247)

[Bảng 2. 15: Luồng sự kiện phụ UC07 41](#_Toc22890)

[Bảng 2. 16: UC08 - Thanh toán 42](#_Toc28178)

[Bảng 2. 17: Luồng sự kiện phụ UC08 43](#_Toc18988)

[Bảng 2. 18: UC09 - Kiểm tra trạng thái đơn hàng 45](#_Toc29259)

[Bảng 2. 19: Luồng sự kiện phụ UC09 46](#_Toc16050)

[Bảng 2. 20: UC10 - Gia hạn hoặc hủy dịch vụ 46](#_Toc5061)

[Bảng 2. 21: Luồng sự kiện chính UC10.1 47](#_Toc18852)

[Bảng 2. 22: Luồng sự kiện chính UC10.2 48](#_Toc7410)

[Bảng 2. 23: Luồng sự kiện phụ UC10 49](#_Toc18361)

[Bảng 2. 24: UC11 - Gửi yêu cầu hỗ trợ kỹ thuật 50](#_Toc331)

[Bảng 2. 25: Luồng sự kiện phụ UC11 52](#_Toc13627)

[Bảng 2. 26: UC12 - Trao đổi trực tiếp 52](#_Toc3938)

[Bảng 2. 27: Luồng sự kiện phụ UC12 54](#_Toc7096)

[Bảng 2. 28: UC13 - Lịch sử mua hàng và thanh toán 54](#_Toc26643)

[Bảng 2. 29: Luồng sự kiện phụ UC13 56](#_Toc14775)

[Bảng 2. 30: UC14 - Lịch sử đăng nhập nhập 56](#_Toc3424)

[Bảng 2. 31: Luồng sự kiện phụ UC14 58](#_Toc7575)

[Bảng 2. 32: UC15 - Thêm thành viên 59](#_Toc23895)

[Bảng 2. 33: Luồng sự kiện phụ UC15 60](#_Toc15533)

[Bảng 2. 34: UC16 - Sửa thành viên 60](#_Toc26985)

[Bảng 2. 35: Luồng sự kiện phụ UC16 62](#_Toc29483)

[Bảng 2. 36: UC17 - Cấm thành viên 62](#_Toc5495)

[Bảng 2. 37: Luồng sự kiện phụ UC17 63](#_Toc24976)

[Bảng 2. 38: UC18 - Thêm hosting 64](#_Toc3976)

[Bảng 2. 39: Luồng sự kiện phụ UC18 65](#_Toc29991)

[Bảng 2. 40: UC19 - Sửa hosting 66](#_Toc18791)

[Bảng 2. 41: Luồng sự kiện phụ UC19 67](#_Toc4928)

[Bảng 2. 42: UC20 - Xóa hosting 68](#_Toc8216)

[Bảng 2. 43: Luồng sự kiện phụ UC20 69](#_Toc16342)

[Bảng 2. 44: UC21 - Đơn hàng 70](#_Toc24325)

[Bảng 2. 45: Luồng sự kiện phụ UC21 71](#_Toc22598)

[Bảng 2. 46: UC22 - Doanh thu 71](#_Toc2841)

[Bảng 2. 47: Luồng sự kiện phụ UC22 72](#_Toc11662)

[Bảng 2. 48: UC23 - Hosting bán được 73](#_Toc4436)

[Bảng 2. 49: Luồng sự kiện phụ UC23 74](#_Toc20674)

[Bảng 2. 50: UC24 - Thành viên 74](#_Toc11688)

[Bảng 2. 51: Luồng sự kiện phụ UC24 75](#_Toc29462)

[Bảng 2. 52: Use case Quản lý mã giảm giá 76](#_Toc539)

[Bảng 2. 53: Luồng sự kiện chính UC25 77](#_Toc8105)

[Bảng 2. 54: Luồng sự kiện chính UC26 78](#_Toc16429)

[Bảng 2. 55: Luồng sự kiện chính UC27 78](#_Toc12338)

[Bảng 2. 56: Luồng sự kiện phụ use case quản lý mã giảm giá 79](#_Toc3876)

[Bảng 2. 57: UC28 - Thêm bài viết 80](#_Toc2082)

[Bảng 2. 58: Luồng sự kiện phụ UC28 82](#_Toc13321)

[Bảng 2. 59: UC29 - Sửa bài viết 82](#_Toc9780)

[Bảng 2. 60: Luồng sự kiện phụ UC29 83](#_Toc30206)

[Bảng 2. 61: UC30 - Xóa bài viết 84](#_Toc15341)

[Bảng 2. 62: Luồng sự kiện phụ UC30 85](#_Toc4526)

[Bảng 2. 63: UC31 - Xem, phân công, phản hồi ticket 86](#_Toc22541)

[Bảng 2. 64: Luồng sự kiện phụ UC31 87](#_Toc5049)

[Bảng 2. 65: UC32 - Cập nhật trạng thái 88](#_Toc20123)

[Bảng 2. 66: Luồng sự kiện phụ UC31 89](#_Toc17655)

[Bảng 2. 67: UC33 - Đóng và lưu trữ ticket 89](#_Toc23241)

[Bảng 2. 68: Luồng sự kiện phụ UC33 91](#_Toc9420)

[Bảng 2. 69: Chức năng "đăng nhập" 92](#_Toc458)

[Bảng 2. 70: Chức năng "Đăng ký" 93](#_Toc24942)

[Bảng 2. 71: Chức năng "Đổi mật khẩu" 94](#_Toc10137)

[Bảng 2. 72: Biểu đồ hoạt động quản lý hosting 95](#_Toc2819)

[Bảng 2. 73: Biểu đồ hoạt động quản lý mã giảm giá 96](#_Toc20344)

[Bảng 2. 74: Biểu đồ hoạt động quản lý bài viết 97](#_Toc17582)

[Bảng 2. 75: Biểu đồ hoạt động quản lý thành viên 98](#_Toc12607)

[Bảng 2. 76: Cấu trúc bảng discount (Mã giảm giá) 104](#_Toc28663)

[Bảng 2. 77: Cấu trúc bảng history\_buy\_hosting (Lịch sử mua hosting) 105](#_Toc21831)

[Bảng 2. 78: Cấu trúc bảng his\_login (Lịch sử đăng nhập) 106](#_Toc21041)

[Bảng 2. 79: Cấu trúc bảng package\_hosting (Gói Hosting) 107](#_Toc13587)

[Bảng 2. 80: Cấu trúc bảng posts (Bài viết/Blog) 107](#_Toc7930)

[Bảng 2. 81: Cấu trúc bảng users (Người dùng) 108](#_Toc2896)

[Bảng 4. 1: Module Đăng ký 124](#_Toc23612)

[Bảng 4. 2: Module Đăng nhập 128](#_Toc24777)

[Bảng 4. 3: Module Đổi mật khẩu 132](#_Toc7047)

[Bảng 4. 4: Module Mua hàng & Thanh toán 135](#_Toc16138)

[Bảng 4. 5: Module Quản lý hosting 140](#_Toc4731)

[Bảng 4. 6: Module Quản lý thành viên 146](#_Toc29353)

# MỞ ĐẦU

**1. Giới thiệu tổng quan đề tài**

Trong thời đại công nghệ số phát triển mạnh mẽ, các hoạt động thương mại, truyền thông, giáo dục và giải trí đều đang chuyển dịch lên môi trường trực tuyến. Website trở thành công cụ không thể thiếu đối với mọi tổ chức, doanh nghiệp nhằm giới thiệu sản phẩm, dịch vụ cũng như tương tác với khách hàng. Tuy nhiên, để một website hoạt động ổn định, nhanh chóng và bảo mật, yếu tố quan trọng nhất chính là hệ thống lưu trữ – hay còn gọi là dịch vụ hosting.

Trước nhu cầu ngày càng tăng về việc xây dựng và duy trì website, thị trường kinh doanh dịch vụ hosting tại Việt Nam đang phát triển mạnh mẽ với nhiều mô hình và giải pháp khác nhau. Trong bối cảnh đó, nhóm em thực hiện đề tài “Xây dựng Website Kinh Doanh Dịch Vụ Hosting” với mục tiêu thiết kế và phát triển một nền tảng web cung cấp dịch vụ lưu trữ trực tuyến cho người dùng.

Hệ thống hướng tới việc mô phỏng mô hình kinh doanh thực tế, bao gồm các chức năng như đăng ký tài khoản, lựa chọn gói dịch vụ, thanh toán, quản lý đơn hàng, và hỗ trợ khách hàng trực tuyến.

Thông qua đề tài này, nhóm sinh viên mong muốn không chỉ tạo ra một sản phẩm web hoàn chỉnh về mặt kỹ thuật, mà còn thể hiện được sự hiểu biết về mô hình kinh doanh thương mại điện tử trong lĩnh vực hạ tầng công nghệ thông tin.

**2. Lý do chọn đề tài**

Lý do chính để nhóm lựa chọn đề tài này xuất phát từ nhu cầu thực tiễn và tính ứng dụng cao trong ngành công nghệ phần mềm. Trước hết, dịch vụ hosting là nền tảng thiết yếu cho bất kỳ ứng dụng web nào. Việc nghiên cứu và xây dựng hệ thống kinh doanh hosting giúp sinh viên hiểu rõ hơn về cơ chế hoạt động của các nhà cung cấp dịch vụ Internet (ISP), mô hình quản lý tài nguyên máy chủ, cũng như quy trình đăng ký, thanh toán và hỗ trợ khách hàng.

Bên cạnh đó, đề tài còn mang ý nghĩa rèn luyện kỹ năng phát triển phần mềm toàn diện (full-stack), khi sinh viên phải đảm nhiệm cả vai trò phân tích yêu cầu, thiết kế hệ thống, lập trình front-end, back-end, cũng như kiểm thử và triển khai ứng dụng. Mặt khác, đề tài cũng giúp sinh viên tiếp cận các công nghệ hiện đại như PHP, MySQL, AJAX, JavaScript, HTML/CSS, framework Bootstrap, và các mô hình kiến trúc web MVC.

Cuối cùng, việc chọn đề tài này còn thể hiện mong muốn đóng góp giải pháp thực tế cho các doanh nghiệp vừa và nhỏ tại Việt Nam, giúp họ dễ dàng xây dựng nền tảng kinh doanh dịch vụ số mà không phụ thuộc hoàn toàn vào các hệ thống nước ngoài.

**3. Mục tiêu của đề tài**

Đề tài hướng đến mục tiêu xây dựng một hệ thống website hoàn chỉnh phục vụ hoạt động kinh doanh dịch vụ hosting. Các mục tiêu cụ thể bao gồm:

Thiết kế giao diện người dùng (UI/UX) hiện đại, dễ sử dụng, đảm bảo trải nghiệm thân thiện cho khách hàng.

Xây dựng hệ thống chức năng đầy đủ cho quy trình mua hàng trực tuyến: từ đăng ký tài khoản, lựa chọn gói hosting, nhập mã giảm giá, thanh toán, cho đến quản lý dịch vụ đã mua.

Phát triển trang quản trị (Admin Dashboard) cho phép nhà quản lý theo dõi thông tin khách hàng, đơn hàng, doanh thu, và xử lý yêu cầu hỗ trợ kỹ thuật.

Đảm bảo bảo mật và tính ổn định của hệ thống, thông qua các kỹ thuật mã hóa mật khẩu, xác thực người dùng, và phân quyền truy cập.

Hỗ trợ khả năng mở rộng và nâng cấp trong tương lai, có thể tích hợp thêm các module như quản lý domain, VPS hoặc dịch vụ điện toán đám mây.

Việc đạt được các mục tiêu này không chỉ giúp sinh viên củng cố kỹ năng lập trình và thiết kế hệ thống, mà còn góp phần hình thành tư duy quản lý dự án phần mềm thực tế.

**4. Nội dung và phạm vi nghiên cứu**

***4.1. Nội dung nghiên cứu***

Nội dung nghiên cứu của đề tài tập trung vào việc phân tích, thiết kế và phát triển một hệ thống web phục vụ kinh doanh dịch vụ hosting. Cụ thể bao gồm các nội dung chính sau:

Phân tích yêu cầu hệ thống: Xác định rõ các nhóm người dùng (khách hàng và quản trị viên), các chức năng chính và phụ, luồng xử lý dữ liệu, và các yêu cầu phi chức năng (bảo mật, hiệu năng, khả năng mở rộng).

Thiết kế cơ sở dữ liệu: Xây dựng mô hình dữ liệu quan hệ nhằm lưu trữ thông tin người dùng, gói dịch vụ, đơn hàng, và lịch sử giao dịch.

Phát triển các module chức năng: Bao gồm module đăng ký/đăng nhập, mua gói hosting, quản lý dịch vụ đã mua, ticket hỗ trợ kỹ thuật, và thống kê doanh thu.

Thiết kế giao diện website: Ứng dụng các nguyên tắc thiết kế giao diện hiện đại, tương thích với đa nền tảng (desktop, tablet, mobile).

Kiểm thử và đánh giá: Thực hiện các phương pháp kiểm thử hộp đen, kiểm thử giao diện, và kiểm thử hiệu năng để đảm bảo hệ thống hoạt động ổn định.

***4.2. Phạm vi nghiên cứu***

Do giới hạn về thời gian và phạm vi học phần, hệ thống được xây dựng ở mức mô phỏng hoạt động kinh doanh hosting. Dữ liệu về gói dịch vụ, đơn hàng và thanh toán được xử lý qua cơ sở dữ liệu nội bộ thay vì tích hợp với các nhà cung cấp hosting thực tế.

Ngoài ra, đề tài tập trung vào việc hoàn thiện mô hình logic, giao diện và chức năng quản trị, chưa triển khai các yếu tố mở rộng như tự động cấp phát tài nguyên máy chủ hoặc kết nối API của các hệ thống quản lý domain quốc tế. Tuy nhiên, cấu trúc hệ thống được thiết kế mở để có thể dễ dàng phát triển thêm các tính năng này trong tương lai.

**5. Kết luận**

Phần mở đầu đã trình bày tổng quan về đề tài, lý do lựa chọn, mục tiêu, nội dung và phạm vi nghiên cứu. Đây là nền tảng quan trọng để triển khai các chương tiếp theo, bao gồm cơ sở lý thuyết, phân tích hệ thống, thiết kế mô hình dữ liệu, và xây dựng sản phẩm thực nghiệm.

Đề tài kỳ vọng sẽ góp phần giúp nhóm sinh viên nâng cao năng lực chuyên môn và mang lại một sản phẩm ứng dụng có giá trị thực tiễn cao trong lĩnh vực kinh doanh dịch vụ trực tuyến.

# CHƯƠNG I: CƠ SỞ LÝ THUYẾT

## 1.1. Tổng quan về lĩnh vực Web

### 1.1.1. Tổng quan

Web (World Wide Web) là một hệ thống siêu văn bản (hypertext) hoạt động trên nền Internet, cho phép người dùng truy cập, chia sẻ và trao đổi thông tin thông qua các trình duyệt web. Kể từ khi ra đời vào đầu thập niên 1990, web đã trở thành nền tảng trọng yếu của hầu hết các hoạt động trong đời sống hiện đại — từ thương mại điện tử, giáo dục, truyền thông cho đến quản lý doanh nghiệp và dịch vụ công.

Ngày nay, web không chỉ là nơi hiển thị thông tin tĩnh mà đã phát triển thành các ứng dụng web (Web Application) phức tạp, có khả năng xử lý dữ liệu, tương tác người dùng và kết nối hệ thống thông minh, tạo nên nền tảng cho hầu hết các mô hình kinh doanh trực tuyến.

### 1.1.2. Bối cảnh phát triển

Trong kỷ nguyên chuyển đổi số (Digital Transformation), việc ứng dụng web vào hoạt động sản xuất, kinh doanh và quản lý trở thành xu hướng tất yếu. Các doanh nghiệp không chỉ sử dụng web để quảng bá hình ảnh mà còn để tự động hóa quy trình bán hàng, chăm sóc khách hàng, và quản trị dữ liệu.



Hình 1. 1: Công nghệ web

Đặc biệt, sự phát triển của các công nghệ mới như trí tuệ nhân tạo (AI), điện toán đám mây (Cloud Computing), Internet vạn vật (IoT) và Blockchain đã mở rộng khả năng ứng dụng của web trong nhiều lĩnh vực khác nhau. Hầu hết các nền tảng công nghệ hiện nay đều tích hợp hoặc hoạt động dựa trên giao diện web, khiến việc phát triển ứng dụng web trở thành một kỹ năng thiết yếu trong ngành này.

### 1.1.3. Tương lai của công nghệ web

Công nghệ web trong tương lai hướng đến sự thông minh, cá nhân hóa và bảo mật cao hơn. Các xu hướng nổi bật bao gồm:

Web 3.0: Tập trung vào việc phi tập trung hóa dữ liệu, tăng quyền kiểm soát thông tin cho người dùng.

Progressive Web App (PWA): Ứng dụng web hoạt động như ứng dụng di động, có thể chạy offline và cài đặt trực tiếp.

AI-driven Web: Tích hợp trí tuệ nhân tạo để tối ưu hóa trải nghiệm người dùng, gợi ý nội dung và phân tích hành vi truy cập.

Tối ưu hiệu năng và an toàn: Đặc biệt quan trọng đối với các hệ thống giao dịch và thương mại điện tử.

### 1.1.4. Ưu điểm của hệ thống web

Tính linh hoạt cao: Người dùng có thể truy cập ở bất kỳ đâu, chỉ cần có Internet.

Dễ bảo trì và nâng cấp: Mọi thay đổi chỉ cần thực hiện trên máy chủ, không cần cập nhật từng thiết bị.

Khả năng mở rộng: Dễ dàng mở rộng quy mô người dùng và dữ liệu thông qua hạ tầng đám mây.

Tương thích đa nền tảng: Hoạt động tốt trên cả máy tính, điện thoại và máy tính bảng và các phần mềm khác nhau.

Tối ưu chi phí vận hành: Không cần cài đặt phần mềm phức tạp, giảm chi phí bảo trì và ổn định.

### 1.1.5. Hạn chế của hệ thống web

Phụ thuộc Internet: Khi mất kết nối, người dùng không thể sử dụng.

Hiệu năng có thể bị giới hạn: So với ứng dụng cài đặt trực tiếp (native app), tốc độ và khả năng xử lý có thể thấp hơn.

Rủi ro bảo mật: Dễ bị tấn công nếu không có cơ chế mã hóa và xác thực chặt chẽ và nguy cơ ảnh hưởng trức tiếp tới khách hàng.

Độ trễ khi tải dữ liệu: Phụ thuộc vào tốc độ mạng và tối ưu của máy chủ.

## 1.2. Tổng quan về công nghệ

### 1.2.1. Công nghệ thiết kế web

a) Khái niệm  
 Thiết kế web là quá trình tạo ra giao diện và cấu trúc của một website nhằm đảm bảo khả năng tương tác, hiển thị nội dung hợp lý, và trải nghiệm người dùng (UX) tốt nhất. Quá trình này bao gồm cả phần giao diện người dùng (Front-end) và phần xử lý logic dữ liệu (Back-end).

b) Back-end là gì  
 Back-end là phần “hậu trường” của một website, nơi diễn ra các hoạt động xử lý logic, lưu trữ và truy xuất dữ liệu. Nó bao gồm máy chủ (server), cơ sở dữ liệu (database) và mã nguồn xử lý (server-side code). Back-end chịu trách nhiệm giao tiếp giữa giao diện và dữ liệu, đảm bảo mọi yêu cầu của người dùng được thực thi chính xác.

c) Front-end là gì  
 Front-end là phần giao diện mà người dùng trực tiếp tương tác — bao gồm các thành phần như bố cục, biểu mẫu, hình ảnh, nút bấm và hiệu ứng hiển thị. Front-end thường được xây dựng bằng HTML, CSS và JavaScript, kết hợp với các thư viện hoặc framework như Bootstrap, jQuery, React, hoặc Vue.js.

#### d) Nguyên tắc thiết kế web

* Đơn giản – Tối ưu – Dễ sử dụng.
* Đảm bảo trải nghiệm người dùng (UX) và tính thẩm mỹ (UI).
* Thiết kế tương thích trên nhiều thiết bị (Responsive Design).
* Tối ưu tốc độ tải trang và SEO.
* Tuân thủ tiêu chuẩn bảo mật web.

### 1.2.2. Ngôn ngữ lập trình – PHP

PHP (Hypertext Preprocessor) là ngôn ngữ lập trình kịch bản chạy phía máy chủ (server-side scripting) được sử dụng phổ biến nhất trong phát triển web. PHP nổi bật với khả năng tích hợp trực tiếp vào mã HTML, dễ học, dễ triển khai và có cộng đồng hỗ trợ rất lớn. Ưu điểm:

Mã nguồn mở, miễn phí, cài đặt dễ dàng.

Tương thích với nhiều hệ điều hành và máy chủ web.

Tích hợp mạnh mẽ với cơ sở dữ liệu MySQL.

Hỗ trợ nhiều framework như Laravel, CodeIgniter, Symfony,… giúp phát triển nhanh và chuẩn hóa mã nguồn.



Hình 1. 2: Ngôn ngữ PHP

PHP là lựa chọn lý tưởng cho các website thương mại điện tử, blog, hệ thống quản lý nội dung (CMS) và các ứng dụng quản trị doanh nghiệp vừa và nhỏ.

### 1.2.3. Công nghệ cơ sở dữ liệu – MySQL

MySQL là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở phổ biến nhất hiện nay. Nó cho phép lưu trữ, truy xuất và quản lý dữ liệu một cách hiệu quả thông qua ngôn ngữ SQL (Structured Query Language). Đặc điểm nổi bật:

Tốc độ xử lý nhanh, ổn định, phù hợp với ứng dụng web.



Hình 1. 3: Cơ sở dữ liệu - MySQL

Hỗ trợ bảo mật, phân quyền người dùng và sao lưu dữ liệu.

Tương thích tốt với PHP và các ngôn ngữ khác như Python, Java.

Dễ triển khai trên các nền tảng hosting hoặc cloud.

### 1.2.4. Các công nghệ nền tảng hỗ trợ khác

Trong quá trình xây dựng và phát triển website kinh doanh dịch vụ hosting, nhóm đã lựa chọn và sử dụng nhiều công nghệ, công cụ hỗ trợ khác nhau nhằm tối ưu hóa quy trình phát triển, kiểm thử, thiết kế giao diện và triển khai hệ thống.

Những công cụ này không chỉ giúp tiết kiệm thời gian lập trình mà còn đảm bảo tính ổn định, trực quan và thân thiện với người dùng. Việc kết hợp hợp lý giữa các công nghệ nền tảng là một trong những yếu tố quan trọng giúp hệ thống hoạt động hiệu quả và chuyên nghiệp hơn. Dưới đây là các công nghệ tiêu biểu được sử dụng trong đề tài:

#### 1.2.4.1. Postman

Postman là công cụ mạnh mẽ dùng để kiểm thử API (Application Programming Interface). Nó cho phép lập trình viên dễ dàng gửi các yêu cầu HTTP (GET, POST, PUT, DELETE) đến máy chủ và xem kết quả phản hồi một cách trực quan.

* Điểm hay:

Giúp phát hiện lỗi logic trong luồng xử lý back-end nhanh chóng.

Giao diện thân thiện, dễ sử dụng cho cả người mới bắt đầu.

Cho phép lưu trữ và chia sẻ bộ sưu tập API, thuận tiện khi làm việc nhóm.



Hình 1. 4: Logo ứng dụng postman

* Vai trò trong đề tài: Nhóm sử dụng Postman để kiểm tra tính chính xác của các

API xử lý dữ liệu như đăng nhập, thanh toán, gửi ticket hỗ trợ,… giúp đảm bảo hệ thống vận hành ổn định và phản hồi đúng dữ liệu.

#### 1.2.4.2. Visual Studio Code (VSCode)

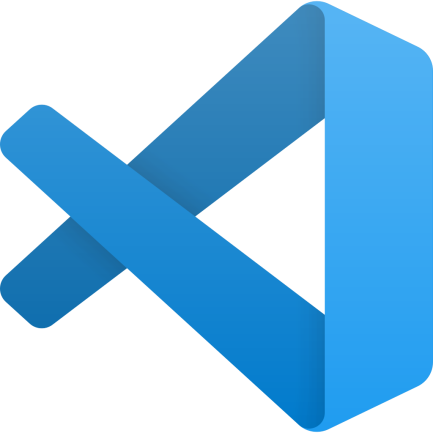
VSCode là môi trường phát triển tích hợp (IDE) phổ biến, miễn phí và nhẹ, hỗ trợ nhiều ngôn ngữ lập trình như PHP, HTML, CSS, JavaScript, Python,...

* Điểm hay:

Tích hợp nhiều tiện ích mở rộng (extensions) giúp tăng tốc quá trình lập trình.

Hỗ trợ debug, tự động hoàn thành mã và gợi ý cú pháp thông minh.

Giao diện gọn gàng, có thể tùy chỉnh linh hoạt.



Hình 1. 5: Logo ứng dụng VSCode

* Vai trò trong đề tài: VSCode được sử dụng làm công cụ lập trình chính cho toàn

bộ dự án, giúp nhóm dễ dàng phát triển, chỉnh sửa và quản lý mã nguồn tập trung, đảm bảo tính đồng bộ trong nhóm.

1.2.4.3. Laragon

Laragon là phần mềm tạo môi trường máy chủ ảo (local server) giúp chạy các ứng dụng PHP, MySQL, Apache hoặc Nginx ngay trên máy tính cá nhân.

* Điểm hay:

Cài đặt nhanh, cấu hình đơn giản chỉ trong vài phút.

Hỗ trợ nhiều ngôn ngữ và framework khác nhau (PHP, Node.js, Python,…).

Khởi động và dừng dịch vụ linh hoạt, tiêu tốn ít tài nguyên hệ thống.



Hình 1. 6: Logo ứng dụng laragon

* Vai trò trong đề tài: Laragon được nhóm dùng để mô phỏng máy chủ web nội bộ,

phục vụ cho việc chạy thử website, kết nối cơ sở dữ liệu và kiểm thử chức năng trước khi triển khai chính thức.

#### 1.2.4.4. Bootstrap

Bootstrap là framework front-end nổi tiếng giúp thiết kế giao diện website nhanh chóng, đẹp mắt và tương thích trên nhiều thiết bị.

* Điểm hay:

Cung cấp sẵn các thành phần giao diện (UI components) như form, button, navbar, modal,…

Hỗ trợ thiết kế responsive, giúp website hiển thị tốt trên điện thoại, máy tính bảng và máy tính để bàn.

Dễ dàng tùy chỉnh màu sắc, bố cục và phong cách theo nhu cầu.



Hình 1. 7: Logo thư viện bootstrap

* Vai trò trong đề tài: Bootstrap giúp nhóm rút ngắn thời gian thiết kế giao diện, đồng thời đảm bảo website có tính thẩm mỹ cao, nhất quán và chuyên nghiệp.

#### 1.2.4.5. Font Awesome

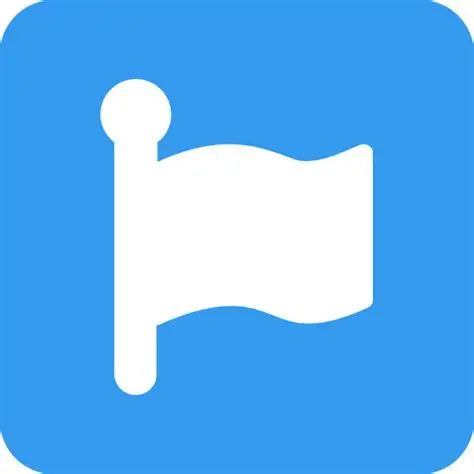
Font Awesome là thư viện biểu tượng vector (icon) được sử dụng rộng rãi trong thiết kế web.

* Điểm hay:

Cung cấp hàng nghìn biểu tượng miễn phí, dễ tích hợp chỉ với một dòng mã.

Có thể tùy chỉnh kích thước, màu sắc, hiệu ứng dễ dàng.

Giúp giao diện trở nên trực quan và sinh động hơn.



Hình 1. 8: Logo thư viện Font Awesome

* Vai trò trong đề tài: Nhóm sử dụng Font Awesome để hiển thị các biểu tượng

trong menu, nút chức năng và thông báo, giúp người dùng dễ nhận biết và thao tác nhanh hơn.

#### 1.2.4.6. jQuery

jQuery là thư viện JavaScript mã nguồn mở giúp đơn giản hóa việc thao tác với DOM, xử lý sự kiện và tạo hiệu ứng động.

* Điểm hay:

Cú pháp ngắn gọn, dễ học.

Tương thích tốt với nhiều trình duyệt.

Hỗ trợ thực hiện Ajax, hiệu ứng động và xử lý form nhanh gọn.



Hình 1. 9: Logo thư viện jQuery

* Vai trò trong đề tài: jQuery được nhóm dùng để tạo các hiệu ứng giao diện, xác

thực dữ liệu form và thực hiện các yêu cầu Ajax mà không cần tải lại trang.

#### 1.2.4.7. SweetAlert

SweetAlert là thư viện giúp hiển thị các thông báo (alert) đẹp mắt, thay thế cho hộp thoại mặc định của trình duyệt.

* Điểm hay:

Giao diện trực quan, dễ nhìn.

Cho phép tùy chỉnh biểu tượng, màu sắc, và nút hành động.

Hỗ trợ xử lý logic sau khi người dùng xác nhận.



Hình 1. 10: Logo thư viện SweetAlert

* Vai trò trong đề tài: SweetAlert được dùng để hiển thị thông báo kết quả đăng ký,

đăng nhập, thanh toán hoặc cảnh báo người dùng khi thao tác sai, giúp nâng cao trải nghiệm người dùng.

#### 1.2.4.8. Ajax

Ajax (Asynchronous JavaScript and XML) là kỹ thuật giúp gửi và nhận dữ liệu từ máy chủ mà không cần tải lại toàn bộ trang web.

* Điểm hay:

Giúp website phản hồi nhanh hơn, tạo cảm giác mượt mà.

Tối ưu băng thông và thời gian tải trang.

Dễ dàng tích hợp với PHP và jQuery.



Hình 1. 11: Logo thư viện Ajax

* Vai trò trong đề tài: Ajax được ứng dụng trong các chức năng như đăng nhập,

thêm vào giỏ hàng, gửi ticket hỗ trợ, giúp website xử lý dữ liệu theo thời gian thực mà không cần refresh.

#### 1.2.4.9. Dastone

Dastone là mẫu giao diện quản trị (Admin Template) được xây dựng bằng HTML, CSS và Bootstrap.

* Điểm hay:

Cung cấp sẵn các thành phần biểu đồ, bảng, thanh điều hướng, giúp xây dựng trang quản trị nhanh chóng.

Giao diện hiện đại, tối ưu cho thiết bị di động.

Dễ tùy biến, hỗ trợ tích hợp biểu đồ thống kê và biểu mẫu quản trị.

IMG_256

Hình 1. 12: Logo thư viện Dastone

* Vai trò trong đề tài: Dastone được sử dụng làm nền tảng cho trang quản trị Admin,

nơi hiển thị thông tin người dùng, đơn hàng, doanh thu và trạng thái dịch vụ một cách trực quan, chuyên nghiệp.

### 1.2.5. Đánh giá chung

Sự kết hợp giữa các công nghệ trên mang lại một hệ sinh thái phát triển web hoàn chỉnh và hiệu quả. PHP và MySQL đóng vai trò xương sống cho phần xử lý dữ liệu, trong khi Bootstrap, jQuery và SweetAlert đảm nhiệm phần giao diện người dùng hiện đại.

Laragon và VSCode giúp thiết lập môi trường phát triển nhanh, thuận tiện, còn Postman hỗ trợ kiểm thử API đảm bảo tính chính xác. Việc tích hợp Ajax giúp tối ưu trải nghiệm người dùng, giảm độ trễ khi thao tác.

Điểm nổi bật của hệ thống là khả năng hoạt động ổn định, tốc độ cao, dễ mở rộng và thân thiện với người dùng. Sự phối hợp giữa các công nghệ front-end và back-end mang lại hiệu quả tổng thể, giúp đề tài không chỉ mang tính học thuật mà còn có tính ứng dụng thực tế cao.

# CHƯƠNG II: THIẾT KẾ VÀ XÂY DỰNG CHƯƠNG TRÌNH

## 2.1 Giới thiệu bài toán

### 2.1.1 Phát biểu bài toán

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, nhu cầu sử dụng dịch vụ lưu trữ website (hosting) ngày càng gia tăng, đặc biệt là với các cá nhân, doanh nghiệp vừa và nhỏ. Việc kinh doanh dịch vụ hosting đòi hỏi một hệ thống trực tuyến giúp người dùng có thể lựa chọn, mua gói dịch vụ, thanh toán và quản lý các thông tin liên quan một cách dễ dàng.

Tuy nhiên, trên thực tế, nhiều doanh nghiệp nhỏ chưa có nền tảng tự động hóa việc bán và quản lý dịch vụ hosting; các quy trình thường được xử lý thủ công, gây tốn thời gian, dễ xảy ra sai sót và thiếu minh bạch. Để khắc phục những hạn chế này, cần xây dựng một website kinh doanh dịch vụ hosting có khả năng quản lý người dùng, đơn hàng, gói dịch vụ, thanh toán và hỗ trợ khách hàng trực tuyến một cách thống nhất.

* Bài toán đặt ra là:

“Xây dựng một hệ thống website có chức năng giới thiệu, bán và quản lý dịch vụ hosting, cho phép khách hàng đăng ký – đăng nhập tài khoản, lựa chọn gói dịch vụ phù hợp, tiến hành thanh toán và theo dõi trạng thái dịch vụ; đồng thời, cung cấp giao diện quản trị cho nhân viên kỹ thuật và quản lý để theo dõi hoạt động kinh doanh, xử lý yêu cầu hỗ trợ và quản lý người dùng.”

* Mục tiêu của bài toán

Hệ thống được xây dựng nhằm mô phỏng mô hình hoạt động của một doanh nghiệp cung cấp dịch vụ hosting, cho phép khách hàng thao tác trực tuyến trên giao diện website và giúp quản trị viên giám sát, điều hành toàn bộ hoạt động nội bộ. Cụ thể, hệ thống phải đáp ứng các mục tiêu sau:

Cho phép người dùng đăng ký, đăng nhập, bảo mật tài khoản, và quản lý dịch vụ đã mua.

Hỗ trợ mua gói hosting, nhập mã giảm giá, thanh toán trực tuyến và gia hạn dịch vụ.

Tích hợp AI giúp giải quyết nhanh những vấn đề của khách hàng đặt ra trong giới hạn cho phép.

Cung cấp hệ thống hỗ trợ khách hàng (ticket) giúp trao đổi, giải đáp và xử lý sự cố cũng như khắc phục sự cố.

Tạo giao diện quản trị (admin) để quản lý thành viên, đơn hàng, dịch vụ, báo cáo doanh thu, và cấu hình hệ thống.

Tối ưu hiệu suất, bảo mật dữ liệu và tạo trải nghiệm sử dụng trực quan, hiện đại.

* Đối tượng hướng đến của hệ thống

Nhóm khách hàng sử dụng dịch vụ hosting: Bao gồm các cá nhân, tổ chức, doanh nghiệp nhỏ và vừa, các đơn vị thiết kế website, các chủ shop online, blog cá nhân, doanh nghiệp mới khởi nghiệp… có nhu cầu thuê dịch vụ lưu trữ website, tên miền và các dịch vụ đi kèm. Nhóm đối tượng này cần một nền tảng trực tuyến cho phép tìm hiểu thông tin gói dịch vụ, so sánh, đặt mua, thanh toán và quản lý dịch vụ một cách nhanh chóng, minh bạch và dễ sử dụng, đồng thời khai thác các tính năng AI nhằm giảm tải công việc cho nhân viên và giảm thời gian chờ của người dùng.

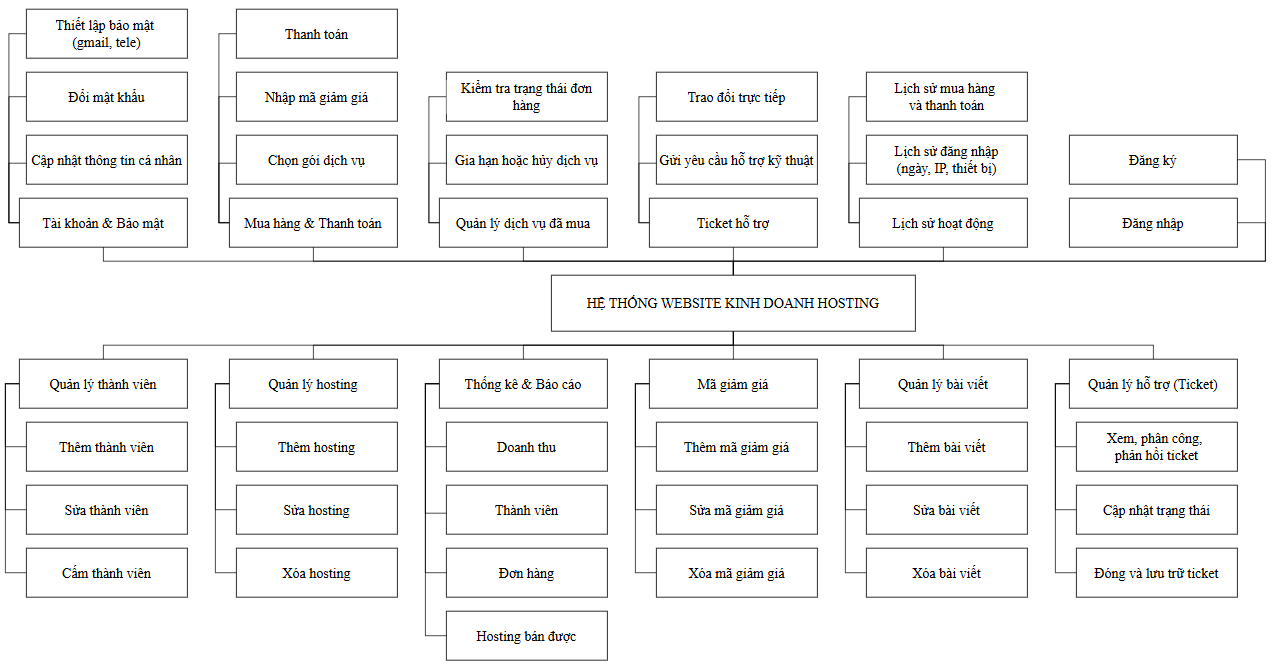
Nhóm quản trị viên phía nhà cung cấp dịch vụ: Là người chịu trách nhiệm vận hành và quản lý toàn bộ hệ thống kinh doanh dịch vụ hosting. Đối tượng này cần một công cụ tập trung để quản lý gói dịch vụ, khách hàng, đơn hàng, hóa đơn, thống kê báo cáo và hỗ trợ khách hàng.

* Điều quan trọng nhất của đề tài

Điều quan trọng nhất của đề tài “Xây dựng website kinh doanh dịch vụ hosting” là xây dựng được một hệ thống bán dịch vụ hosting trực tuyến tự động hóa tối đa quy trình kinh doanh (từ tiếp nhận nhu cầu, đặt dịch vụ, thanh toán, xuất hóa đơn cho đến thống kê, báo cáo theo ngày, tuần, tháng) và tích hợp trí tuệ nhân tạo (AI) để hỗ trợ khách hàng cũng như hỗ trợ quản trị viên.

Nhờ đó, hệ thống không chỉ dừng lại ở việc giới thiệu và bán các gói hosting đơn thuần, mà còn giúp nâng cao trải nghiệm người dùng, giảm phụ thuộc vào nhiều loại nhân viên vận hành, tập trung mọi hoạt động vào vai trò quản trị viên và tối ưu hiệu quả quản lý, theo dõi hoạt động kinh doanh của nhà cung cấp dịch vụ hosting.

* Cấu trúc phân cấp chức năng của hệ thống



Hình 2. 1: Phân cấp chức năng

Dựa trên yêu cầu nghiệp vụ, hệ thống được chia thành các nhóm chức năng tiêu biển chính như sau:

* Nhóm chức năng dành cho khách hàng (Customer)

Tài khoản và bảo mật: Chức năng cho phép khách hàng quản lý thông tin tài khoản cá nhân (họ tên, email, số điện thoại, địa chỉ…), thay đổi mật khẩu và cấu hình các tùy chọn bảo mật (ví dụ: xác thực nhiều lớp nếu được triển khai), nhằm đảm bảo tài khoản được bảo vệ an toàn trong quá trình sử dụng.

Đăng ký: Cho phép người dùng mới tạo tài khoản trên hệ thống bằng cách cung cấp các thông tin cơ bản như họ tên, email, mật khẩu,… Sau khi đăng ký thành công, người dùng có thể đăng nhập và sử dụng các chức năng dành cho khách hàng.

Đăng nhập: Cho phép người dùng đã có tài khoản truy cập vào hệ thống bằng tên đăng nhập (hoặc email) và mật khẩu. Sau khi đăng nhập, hệ thống xác thực thông tin và phân quyền sử dụng các chức năng tương ứng với vai trò khách hàng.

Mua hàng và thanh toán: Chức năng cho phép khách hàng lựa chọn gói dịch vụ hosting, cấu hình các tùy chọn đi kèm, áp dụng mã giảm giá (nếu có), xác nhận đơn hàng và thực hiện thanh toán trực tuyến thông qua các phương thức thanh toán được hỗ trợ. Sau khi thanh toán thành công, đơn hàng và dịch vụ tương ứng sẽ được ghi nhận trong hệ thống.

Quản lý dịch vụ đã mua: Cho phép khách hàng theo dõi danh sách các dịch vụ hosting đã đăng ký, xem chi tiết cấu hình, trạng thái dịch vụ, thời hạn sử dụng, thực hiện gia hạn hoặc gửi yêu cầu hủy dịch vụ trong trường hợp không còn nhu cầu.

Ticket hỗ trợ: Khách hàng có thể gửi các yêu cầu hỗ trợ (kỹ thuật, thanh toán, sử dụng dịch vụ…), mô tả vấn đề gặp phải, đính kèm thông tin liên quan. Hệ thống cho phép theo dõi trạng thái xử lý, trao đổi với bộ phận hỗ trợ hoặc hệ thống AI hỗ trợ, và nhận phản hồi từ quản trị viên.

Lịch sử hoạt động: Chức năng ghi nhận và hiển thị lại các hoạt động quan trọng của khách hàng trên hệ thống như lịch sử đăng nhập, lịch sử đơn hàng, lịch sử thanh toán và các giao dịch liên quan. Điều này giúp khách hàng dễ dàng tra cứu thông tin và tăng tính minh bạch trong quá trình sử dụng dịch vụ.

* Nhóm chức năng dành cho quản trị viên (Administrator)

Quản lý thành viên: Cho phép quản trị viên xem danh sách tài khoản khách hàng, tra cứu thông tin chi tiết, khóa/mở tài khoản trong các trường hợp vi phạm hoặc theo yêu cầu, hỗ trợ điều chỉnh một số thông tin cần thiết để đảm bảo hệ thống người dùng được quản lý chặt chẽ.

Quản lý hosting: Chức năng dùng để định nghĩa và quản lý các gói dịch vụ hosting: thêm mới, chỉnh sửa thông tin, thay đổi giá, cấu hình tài nguyên, ẩn/hiển thị hoặc xóa gói dịch vụ. Qua đó, quản trị viên có thể linh hoạt xây dựng danh mục sản phẩm phù hợp với chiến lược kinh doanh.

Thống kê và báo cáo: Cung cấp các báo cáo tổng hợp về tình hình hoạt động của hệ thống theo ngày, tuần, tháng: số lượng đơn hàng, doanh thu, số dịch vụ đang hoạt động, số lượng khách hàng mới, các gói dịch vụ được sử dụng nhiều,… Các báo cáo có thể được thể hiện dưới dạng bảng số liệu và biểu đồ, giúp quản trị viên dễ dàng đánh giá và ra quyết định.

Quản lý đơn hàng và thanh toán: Cho phép quản trị viên xem danh sách và chi tiết các đơn hàng, kiểm tra trạng thái thanh toán, xác nhận hoặc hủy đơn, xử lý các trường hợp thanh toán lỗi, hoàn tiền (nếu được hỗ trợ trong phạm vi hệ thống). Chức năng này đảm bảo quy trình xử lý đơn hàng diễn ra minh bạch và chính xác.

Mã giảm giá: Chức năng giúp quản trị viên tạo mới, cập nhật, xóa các mã giảm giá, thiết lập điều kiện áp dụng, thời gian hiệu lực và theo dõi tình trạng sử dụng. Qua đó, hệ thống hỗ trợ triển khai các chiến dịch marketing linh hoạt.

Quản lý bài viết: Chức năng dùng tạo ra các bài viết và quản lý các gói dịch vụ bài viết: thêm mới, chỉnh sửa thông tin, xóa bài viết. Qua đó, quản trị viên có thể linh hoạt xây dựng các bài viết để hỗ trợ cũng như quảng bá trang website.

Quản lý hỗ trợ: Chức năng này cho phép quản trị viên theo dõi và xử lý các ticket hỗ trợ: tiếp nhận yêu cầu, xem chi tiết nội dung, phản hồi cho khách hàng, cập nhật trạng thái (mới, đang xử lý, đã xử lý, đóng…) và phối hợp với hệ thống AI hỗ trợ để tự động trả lời các câu hỏi lặp lại, giúp giảm tải khối lượng công việc thủ công.

Những nhóm chức năng này được thể hiện trực quan trong sơ đồ phân cấp chức năng (Hình 2.1), giúp minh họa mối quan hệ giữa các module trong hệ thống. Mỗi chức năng chính được chia nhỏ thành các chức năng con, tạo nên cấu trúc rõ ràng, thuận lợi cho việc thiết kế chi tiết và lập trình.

### 2.1.2. Danh sách tác nhân (Actor)

Trong hệ thống Website Kinh Doanh Dịch Vụ Hosting, các tác nhân (actor) là những đối tượng bên ngoài tương tác trực tiếp với hệ thống để thực hiện các chức năng cụ thể.

Dựa trên sơ đồ Use Case tổng thể, hệ thống có hai tác nhân chính: Customer (khách hàng – người dùng có tài khoản) và Administrator (quản trị viên hệ thống).

Các thao tác của người truy cập chưa đăng nhập (khách vãng lai) như xem trang chủ, xem thông tin giới thiệu, xem danh sách gói dịch vụ… được xem là một phần trong luồng tương tác chung, không được tách thành một actor riêng trong phạm vi đề tài.

*1. Customer (Khách hàng – người dùng có tài khoản)*

**Mô tả:** Customer là người dùng đã đăng ký tài khoản và đăng nhập vào hệ thống. Đây là đối tượng trực tiếp sử dụng các dịch vụ hosting mà hệ thống cung cấp. Trước khi trở thành Customer, người dùng có thể truy cập website như khách vãng lai để tham khảo thông tin, sau đó thực hiện đăng ký tài khoản.

* Các chức năng tương tác:

Quản lý tài khoản & bảo mật: Quản lý thông tin cá nhân, cập nhật hồ sơ, đổi mật khẩu, thiết lập các tùy chọn bảo mật.

Mua hàng & thanh toán: Lựa chọn gói dịch vụ hosting phù hợp, áp dụng mã giảm giá, tiến hành thanh toán trực tuyến thông qua các phương thức được hệ thống hỗ trợ.

Quản lý dịch vụ đã mua: Theo dõi trạng thái các dịch vụ đang hoạt động, xem thông tin chi tiết cấu hình, thực hiện gia hạn hoặc yêu cầu hủy dịch vụ khi cần.

Hỗ trợ & ticket: Gửi yêu cầu hỗ trợ kỹ thuật hoặc hỗ trợ sử dụng dịch vụ; theo dõi trạng thái xử lý ticket; trao đổi với hệ thống hỗ trợ (ví dụ: chatbot AI) và với quản trị viên trong các trường hợp cần can thiệp.

Lịch sử hoạt động: Xem lịch sử đăng nhập, lịch sử đơn hàng, các giao dịch thanh toán và hóa đơn tương ứng.

* Vai trò: Là người sử dụng dịch vụ chính, tạo ra doanh thu cho hệ thống. Đưa ra

các phản hồi, đánh giá và yêu cầu hỗ trợ, góp phần cải thiện chất lượng dịch vụ và trải nghiệm người dùng.

*2. Administrator (Quản trị viên hệ thống)*

Mô tả: Administrator là người chịu trách nhiệm vận hành, quản lý và giám sát toàn bộ hệ thống website. Đây là tác nhân có quyền cao nhất, có thể can thiệp vào các module quản lý người dùng, dịch vụ, đơn hàng, hóa đơn, ticket hỗ trợ và các báo cáo thống kê.

Trong phạm vi đề tài, hệ thống chỉ sử dụng một vai trò quản trị viên duy nhất, không phân tách thêm các loại nhân viên khác (nhân viên kỹ thuật, nhân viên chăm sóc khách hàng, nhân viên kinh doanh,…). Một số tác vụ hỗ trợ được giảm tải thông qua việc tích hợp AI.

* Các chức năng tương tác:

Quản lý người dùng: Xem danh sách khách hàng, khóa/mở tài khoản trong các trường hợp vi phạm hoặc theo yêu cầu, hỗ trợ cập nhật một số thông tin cần thiết.

Quản lý gói hosting: Thêm mới, chỉnh sửa, ẩn/hiển thị hoặc xóa các gói dịch vụ hosting; cập nhật giá, thông số kỹ thuật, thời hạn và các chính sách đi kèm.

Quản lý đơn hàng & thanh toán: Xem chi tiết đơn hàng, theo dõi trạng thái thanh toán, xác nhận hoặc hủy đơn trong các trường hợp cần thiết, xử lý hoàn tiền (nếu phạm vi hệ thống cho phép).

Quản lý mã giảm giá: Tạo, cập nhật, xóa mã giảm giá; thiết lập thời gian hiệu lực, điều kiện áp dụng và theo dõi tình trạng sử dụng.

Quản lý bài viết: Thêm mới, chỉnh sủa bài viết hoặc xóa bài viết nhằm cung cấp thông tin đầy đủ và chính xác cho khách hàng.

Quản lý hỗ trợ (ticket): Tiếp nhận và xem chi tiết các yêu cầu hỗ trợ, phản hồi khách hàng, cập nhật trạng thái xử lý ticket; phối hợp với module AI hỗ trợ để xử lý các yêu cầu lặp lại hoặc đơn giản.

Thống kê & báo cáo: Theo dõi các báo cáo tổng hợp về doanh thu, số lượng đơn hàng, số lượng dịch vụ đang hoạt động, lượng khách hàng mới, tần suất sử dụng các gói dịch vụ theo ngày/tuần/tháng, phục vụ việc đánh giá hiệu quả kinh doanh và ra quyết định.

* Vai trò: Là người điều hành và kiểm soát toàn bộ hoạt động của hệ thống. Đảm

bảo dữ liệu được quản lý chính xác, an toàn và bảo mật. Kết hợp với hệ thống AI để nâng cao hiệu quả vận hành và tối ưu trải nghiệm của khách hàng.

### 2.1.3 Danh sách các ca sử dụng (use case)

Bảng 2. 1: Danh sách các ca sử dụng

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã use case** | **Chức chính** | | **Actor** |
| UC01 | Đăng nhập | | User |
| UC02 | Đăng ký | | User |
| UC03 | Đổi mật khẩu | | User |
| UC04 | Cập nhật thông tin cá nhân | | User |
| UC05 | Thiết lập bảo mật | | User |
| UC06 | Chọn gói dịch vụ | | User |
| UC07 | Nhập mã giảm giá | | User |
| UC08 | Thanh toán | | User |
| UC09 | Kiểm tra trạng thái đơn hàng | | User |
| UC10 (Gia hạn hoặc hủy dịch vụ) | UC10.1 | Gia hạn dịch vụ | User |
| UC10.2 | Hủy dịch vụ |
| UC11 | Gửi yêu cầu hỗ trợ kỹ thuật | | User |
| UC12 | Trao đổi trực tiếp | | User |
| UC13 | Lịch sử mua hàng và thanh toán | | User |
| UC14 | Lịch sử đăng nhập nhập | | User |
| UC15 | Thêm thành viên | | Administrator |
| UC16 | Sửa thành viên | | Administrator |
| UC17 | Cấm thành viên | | Administrator |
| UC18 | Thêm hosting | | Administrator |
| UC19 | Sửa hosting | | Administrator |
| UC20 | Xóa hosting | | Administrator |
| UC21 | Đơn hàng | | Administrator |
| UC22 | Doanh thu | | Administrator |
| UC23 | Hosting bán được | | Administrator |
| UC24 | Thành viên | | Administrator |
| UC25 | Thêm mã giảm giá | | Administrator |
| UC26 | Sửa mã giảm giá | | Administrator |
| UC27 | Xóa mã giảm giá | | Administrator |
| UC28 | Thêm bài viết | | Administrator |
| UC29 | Sửa bài viết | | Administrator |
| UC30 | Xóa bài viết | | Administrator |
| UC31 | Xem, phân công, phản hồi ticket | | Administrator |
| UC32 | Cập nhật trạng thái | | Administrator |
| UC33 | UC33 - Đóng và lưu trữ ticket | | Administrator |

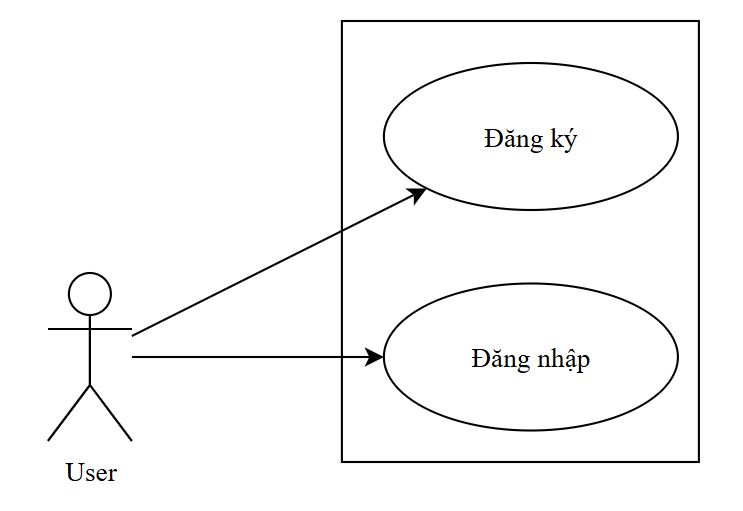
## 2.2 Thiết kế hệ thống



Hình 2. 2: Use case tổng quát

### 2.2.1 Biểu đồ Use case

#### a, Use case đăng nhập vào hệ thống



Hình 2. 3: Use case đăng nhập

* Mô tả chi tiết Use Case: Đăng nhập hệ thống

Bảng 2. 2: UC01 – Đăng nhập

|  |  |
| --- | --- |
| **ID** | UC01 |
| **Tên Use Case** | Đăng nhập |
| **Tác nhân chính** | Người dùng (Customer hoặc Administrator) |
| **Mục tiêu** | Cho phép người dùng truy cập vào hệ thống bằng thông tin tài khoản hợp lệ, xác thực qua cơ sở dữ liệu và khởi tạo phiên làm việc an toàn để sử dụng các chức năng theo quyền hạn. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường.  - Cơ sở dữ liệu users khả dụng và chứa thông tin tài khoản hợp lệ. |
| **Điều kiện thực hiện** | - Người dùng đã có tài khoản hợp lệ trong hệ thống.  - Thiết bị có kết nối Internet ổn định.- Trình duyệt hỗ trợ session hoặc JWT token. |
| **Sự kiện kích hoạt** | Người dùng nhập thông tin đăng nhập (tên đăng nhập và mật khẩu) và nhấn nút “Đăng nhập”. |
| **Luồng sự kiện chính** | B1: Người dùng mở giao diện đăng nhập.  B2: Người dùng nhập username và password, sau đó nhấn nút “Đăng nhập”.  B3: Hệ thống nhận yêu cầu POST với action = LOGIN, thực hiện lọc dữ liệu đầu vào (anti-XSS).  B4: Hệ thống kiểm tra dữ liệu đầu vào:  • Nếu trường username hoặc password rỗng → [A1] và kết thúc use case.  • Nếu tài khoản không tồn tại trong bảng users → [A2] và kết thúc use case.  => Nếu hợp lệ, tiếp tục bước B5.  B5: Hệ thống xác thực mật khẩu bằng hàm verify\_password():  • Nếu không khớp → [A3] và kết thúc use case.  => Nếu khớp, tiếp tục bước B6.  B6: Hệ thống sinh chuỗi token ngẫu nhiên $ss\_token, bắt đầu giao dịch begin\_transaction(), cập nhật thông tin đăng nhập (IP, vị trí, thời gian, token phiên) vào bảng users và ghi lịch sử đăng nhập vào bảng his\_login.  • Nếu bất kỳ thao tác nào trong giao dịch thất bại → [A4] kết thúc use case.  => Nếu thành công, tiếp tục bước B7.  B7: Hệ thống commit() giao dịch, lưu session token vào $\_SESSION['session'].  B8: Hệ thống phản hồi json\_Msg('success', 'Đăng nhập thành công') và chuyển người dùng đến giao diện chính.  B9 (ngoại lệ mạng): Nếu trong quá trình xử lý có sự cố mất kết nối hoặc lỗi máy chủ → [A5] và kết thúc use case. |

* Luồng sự kiện phụ (Alternative / Exception Flow):

Bảng 2. 3: Luồng sự kiện phụ UC01

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Thiếu thông tin đầu vào (username hoặc password rỗng) | Hiển thị thông báo “Vui lòng nhập đủ thông tin.” |
| **A2** | Tài khoản không tồn tại trong bảng users | Hiển thị thông báo “Tài khoản không tồn tại.” |
| **A3** | Mật khẩu không chính xác (verify\_password không khớp) | Hiển thị thông báo “Mật khẩu không chính xác.” |
| **A4** | Ghi nhật ký hoặc cập nhật thất bại trong giao dịch | Hệ thống rollback() và thông báo “Đăng nhập thất bại.” |
| **A5** | Mất kết nối mạng hoặc lỗi máy chủ | Hiển thị thông báo “Không thể kết nối đến hệ thống, vui lòng thử lại.” |

* Mô tả chi tiết Use Case: Đăng ký

Bảng 2. 4: UC02 – Đăng ký

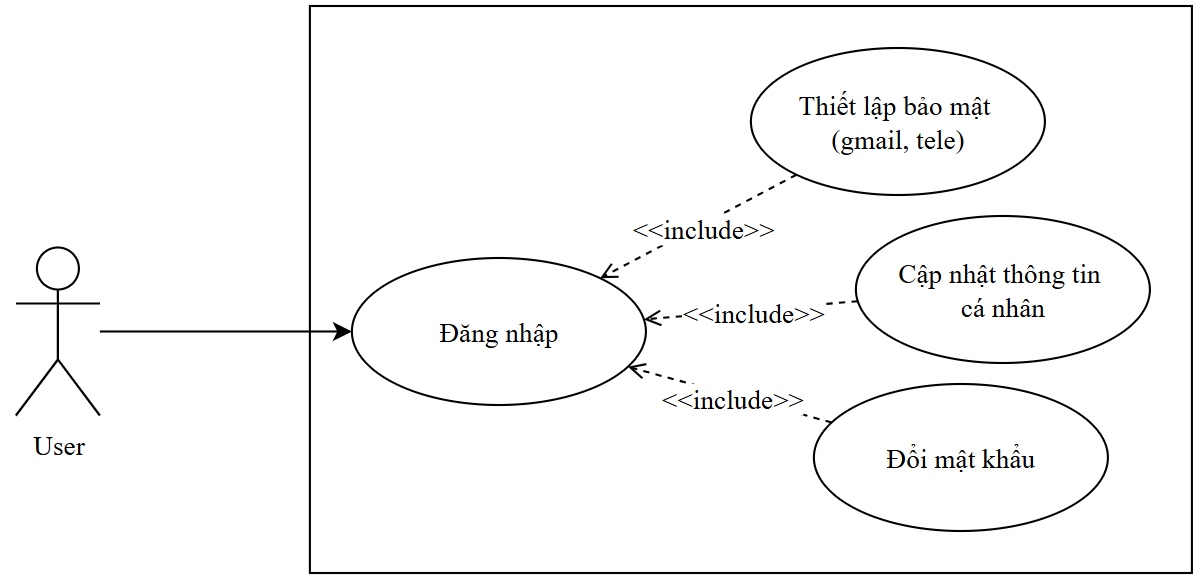
|  |  |
| --- | --- |
| **ID** | UC02 |
| **Tên Use Case** | Đăng ký |
| **Tác nhân chính** | Người dùng mới (Customer chưa có tài khoản) |
| **Mục tiêu** | - Cho phép người dùng tạo tài khoản mới bằng cách cung cấp thông tin hợp lệ (tên đăng nhập, mật khẩu, email, số điện thoại).  - Hệ thống kiểm tra tính hợp lệ, đảm bảo không trùng lặp và lưu thông tin vào cơ sở dữ liệu để người dùng có thể đăng nhập và sử dụng dịch vụ. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu users khả dụng. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng truy cập đúng giao diện đăng ký của hệ thống. |
| **Sự kiện kích hoạt** | Người dùng nhập đầy đủ thông tin đăng ký (username, email, password, confirm password, phone) và nhấn nút “Đăng ký”. |
| **Luồng sự kiện chính** | B1: Người dùng mở giao diện đăng ký.  B2: Người dùng nhập thông tin: username, email, password, confirm password, phone và nhấn “Đăng ký”.  B3: Hệ thống nhận yêu cầu POST với action = REGISTER, lọc dữ liệu đầu vào bằng antixss().  B4: Hệ thống kiểm tra tính hợp lệ của dữ liệu:  • Nếu thiếu bất kỳ trường nào → [A1] và kết thúc use case.  • Nếu username chứa ký tự đặc biệt hoặc dấu → [A2] và kết thúc use case.  • Nếu username đã tồn tại trong bảng users → [A3] và kết thúc use case.  • Nếu username quá ngắn (<6 ký tự) → [A4] và kết thúc use case.  • Nếu email đã tồn tại → [A5] và kết thúc use case.  • Nếu email không hợp lệ hoặc không phải định dạng Gmail → [A6] và kết thúc use case.  => Nếu tất cả điều kiện hợp lệ, tiếp tục bước B5.  B5: Hệ thống sinh chuỗi token ngẫu nhiên $ss\_token, mã hóa mật khẩu bằng TypePassword(), bắt đầu giao dịch begin\_transaction().  B6: Thực hiện thêm bản ghi mới vào bảng users với thông tin: username, email, phone, password, session\_token, ip\_adr, address, createdate, time, level, money, band.  • Nếu quá trình thêm dữ liệu hoặc giao dịch gặp lỗi → [A7] và kết thúc use case.  => Nếu thành công, tiếp tục bước B7.  B7: Hệ thống commit() giao dịch, lưu token phiên vào $\_SESSION['session'], phản hồi json\_Msg('success', 'Đăng ký thành công').  B8: Hệ thống tự động đăng nhập người dùng vừa tạo và chuyển đến giao diện chính (dashboard).  B9 (ngoại lệ mạng): Nếu trong quá trình xử lý xảy ra lỗi kết nối hoặc lỗi máy chủ → [A8] và kết thúc use case. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 5: Luồng sự kiện phụ UC02

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Thiếu thông tin đầu vào | “Vui lòng điền đầy đủ thông tin.” |
| **A2** | Tên đăng nhập có ký tự đặc biệt hoặc dấu | “Tên đăng nhập không gồm ký tự đặc biệt và dấu.” |
| **A3** | Tên đăng nhập đã tồn tại | “Người dùng đã tồn tại.” |
| **A4** | Tên đăng nhập quá ngắn (<6 ký tự) | “Tên đăng nhập phải tối thiểu 6 ký tự.” |
| **A5** | Email đã được sử dụng | “Email đã được sử dụng.” |
| **A6** | Email không hợp lệ hoặc không phải định dạng Gmail | “Email không hợp lệ.” |
| **A7** | Lỗi khi chèn dữ liệu hoặc lỗi giao dịch | “Đăng ký thất bại.” |
| **A8** | Mất kết nối hoặc lỗi máy chủ | “Không thể kết nối đến hệ thống, vui lòng thử lại.” |

#### b, Use case Tài khoản & Bảo mật



Hình 2. 4: Use case Tài khoản & Bảo mật

* Mô tả chi tiết Use Case: Đổi mật khẩu

Bảng 2. 6: UC03 - Đổi mật khẩu

|  |  |
| --- | --- |
| **ID** | UC03 |
| **Tên Use Case** | Đổi mật khẩu |
| **Tác nhân chính** | Người dùng đã đăng nhập (Customer) |
| **Mục tiêu** | Cho phép người dùng thay đổi mật khẩu của tài khoản bằng cách cung cấp mật khẩu cũ chính xác, mật khẩu mới và xác nhận mật khẩu mới. Hệ thống kiểm tra tính hợp lệ và cập nhật mật khẩu mới vào cơ sở dữ liệu. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu users khả dụng. - Người dùng PHẢI đã đăng nhập vào hệ thống. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng truy cập đúng giao diện đổi mật khẩu trong phần tài khoản cá nhân. |
| **Sự kiện kích hoạt** | Người dùng nhập đầy đủ thông tin (mật khẩu cũ, mật khẩu mới, xác nhận mật khẩu mới) và nhấn nút "Đổi mật khẩu". |
| **Luồng sự kiện chính** | B1: Người dùng mở giao diện đổi mật khẩu trong khu vực quản lý tài khoản.  B2: Người dùng nhập thông tin: password\_old, password\_new, password\_cf và nhấn nút "Đổi mật khẩu".  B3: Hệ thống nhận yêu cầu POST với action = 'CHANGE\_PASS' và lọc dữ liệu đầu vào bằng antixss().  B4: Hệ thống gọi hàm checkUser($username, $session) để xác thực quyền của người dùng.  • Nếu không hợp lệ → [A1] và kết thúc use case.  B5: Hệ thống kiểm tra tính hợp lệ của dữ liệu:  • Nếu bất kỳ trường nào (password\_old, password\_new, password\_cf) bị bỏ trống → [A2] và kết thúc use case.  • Nếu password\_old không khớp với mật khẩu hiện tại trong cơ sở dữ liệu → [A3] và kết thúc use case.  • Nếu password\_new hoặc password\_cf có độ dài nhỏ hơn 6 ký tự → [A4] và kết thúc use case.  • Nếu password\_new không khớp với password\_cf → [A5] và kết thúc use case.  => Nếu tất cả điều kiện hợp lệ, tiếp tục bước B6.  B6: Hệ thống mã hóa mật khẩu mới bằng TypePassword($password\_cf) và bắt đầu giao dịch begin\_transaction().  B7: Hệ thống thực hiện cập nhật mật khẩu mới vào bảng users cho người dùng hiện tại.  • Nếu quá trình cập nhật thất bại → [A6] và kết thúc use case.  => Nếu thành công, tiếp tục bước B8.  B8: Hệ thống commit() giao dịch và phản hồi json\_Msg('success', 'Đổi mật khẩu thành công').  B9: Hệ thống có thể yêu cầu người dùng đăng nhập lại hoặc thông báo thành công và giữ nguyên trạng thái đăng nhập. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 7: Luồng sự kiện phụ UC03

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Người dùng không có quyền thực hiện hành động (chưa đăng nhập hoặc session không hợp lệ). | "Bạn không có quyền thực hiện hành động này" |
| **A2** | Có trường thông tin bị bỏ trống. | "Vui lòng nhập đủ thông tin" |
| **A3** | Mật khẩu cũ không chính xác. | "Mật khẩu cũ không chính xác" |
| **A4** | Mật khẩu mới quá ngắn (dưới 6 ký tự). | "Độ dài mật khẩu phải lớn hơn 6 kí tự" |
| **A5** | Mật khẩu mới và xác nhận mật khẩu không khớp. | "Nhập lại mật khẩu không khớp" |
| **A6** | Lỗi hệ thống khi cập nhật cơ sở dữ liệu (lỗi giao dịch, lỗi kết nối). | "Đổi mật khẩu thất bại" |

* Mô tả chi tiết Use Case: Cập nhật thông tin cá nhân

Bảng 2. 8: UC04 - Cập nhật thông tin cá nhân

|  |  |
| --- | --- |
| **ID** | UC05 |
| **Tên Use Case** | Cập nhật thông tin cá nhân |
| **Tác nhân chính** | Người dùng đã đăng nhập (Customer) |
| **Mục tiêu** | Cho phép người dùng cập nhật thông tin cá nhân của mình (email, số điện thoại, địa chỉ) trong hệ thống. Hệ thống kiểm tra tính hợp lệ và cập nhật vào cơ sở dữ liệu. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu users khả dụng. - Người dùng PHẢI đã đăng nhập vào hệ thống. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng truy cập đúng trang thông tin cá nhân. |
| **Sự kiện kích hoạt** | Người dùng chỉnh sửa thông tin cá nhân và nhấn nút "Lưu thay đổi". |
| **Luồng sự kiện chính** | B1: Người dùng truy cập trang thông tin cá nhân trong tài khoản.  B2: Người dùng chỉnh sửa các thông tin: email, phone, address và nhấn nút "Lưu thay đổi".  B3: Hệ thống nhận yêu cầu POST với action = 'UPDATE\_PROFILE' và lọc dữ liệu đầu vào bằng antixss().  B4: Hệ thống xác thực quyền người dùng thông qua session.  • Nếu không hợp lệ → [A1] và kết thúc use case.  B5: Hệ thống kiểm tra tính hợp lệ của dữ liệu:  • Nếu email trống → [A2] và kết thúc use case.  • Nếu email không đúng định dạng → [A3] và kết thúc use case.  • Nếu email đã được sử dụng bởi người dùng khác → [A4] và kết thúc use case.  • Nếu phone không đúng định dạng (nếu có validate) → [A5] và kết thúc use case.  => Nếu tất cả điều kiện hợp lệ, tiếp tục bước B6.  B6: Hệ thống bắt đầu giao dịch begin\_transaction().  B7: Hệ thống thực hiện cập nhật thông tin vào bảng users cho người dùng hiện tại.  • Nếu quá trình cập nhật thất bại → [A6] và kết thúc use case.  => Nếu thành công, tiếp tục bước B8.  B8: Hệ thống commit() giao dịch và phản hồi json\_Msg('success', 'Cập nhật thông tin thành công').  B9: Hệ thống hiển thị thông báo thành công và cập nhật lại thông tin trên giao diện. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 9: Luồng sự kiện phụ UC04

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Người dùng không có quyền thực hiện hành động (session không hợp lệ). | "Bạn không có quyền thực hiện hành động này" |
| **A2** | Email bị bỏ trống. | "Email không được để trống" |
| **A3** | Email không đúng định dạng. | "Email không hợp lệ" |
| **A4** | Email đã được sử dụng bởi tài khoản khác. | "Email đã được sử dụng bởi tài khoản khác" |
| **A5** | Số điện thoại không đúng định dạng. | "Số điện thoại không hợp lệ" |
| **A6** | Lỗi hệ thống khi cập nhật cơ sở dữ liệu. | "Cập nhật thông tin thất bại" |

* Mô tả chi tiết Use Case: Thiết lập bảo mật

Bảng 2. 10: UC05 - Thiết lập bảo mật

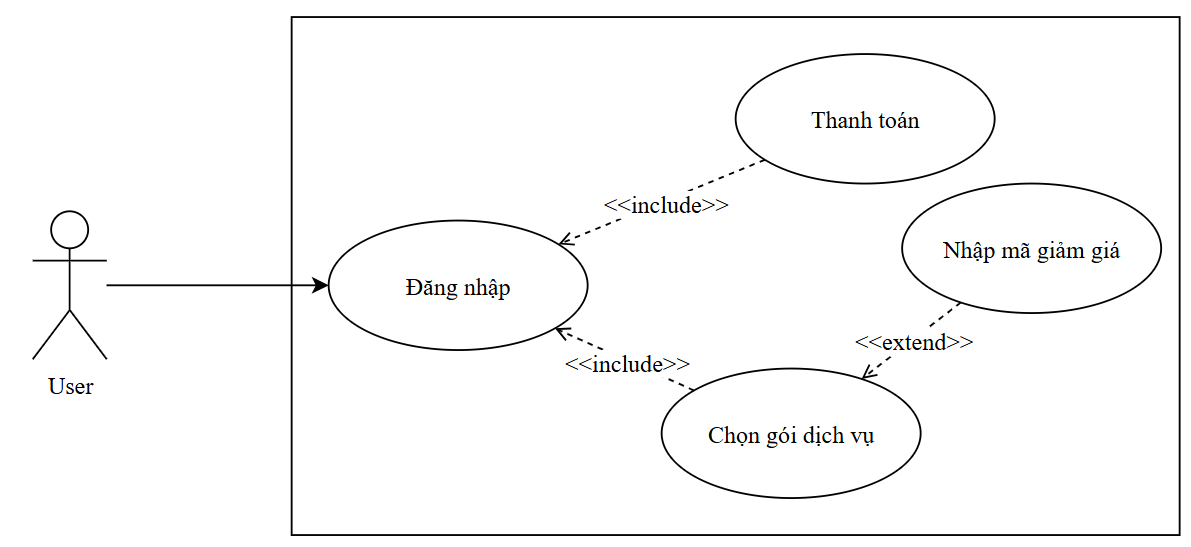
|  |  |
| --- | --- |
| **ID** | UC05 |
| **Tên Use Case** | Thiết lập bảo mật (Gmail, Telegram) |
| **Tác nhân chính** | Người dùng đã đăng nhập (Customer) |
| **Mục tiêu** | Cho phép người dùng thiết lập và liên kết tài khoản Gmail, Telegram để nhận thông báo bảo mật, cảnh báo đăng nhập và khôi phục tài khoản khi cần thiết. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu users khả dụng. - Người dùng PHẢI đã đăng nhập vào hệ thống. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng có tài khoản Gmail và Telegram hợp lệ. - Người dùng truy cập đúng trang cài đặt bảo mật. |
| **Sự kiện kích hoạt** | Người dùng nhập thông tin Gmail/Telegram và nhấn nút "Liên kết" hoặc "Lưu cài đặt". |
| **Luồng sự kiện chính** | B1: Người dùng truy cập trang thiết lập bảo mật trong tài khoản.  B2: Người dùng chọn loại thiết lập:  • Liên kết Gmail để nhận cảnh báo bảo mật  • Liên kết Telegram để nhận thông báo real-time  B3: Đối với Gmail:  - Người dùng nhập địa chỉ Gmail và nhấn "Gửi mã xác minh"  - Hệ thống gửi mã xác minh 6 chữ số đến Gmail  - Người dùng nhập mã xác minh và nhấn "Xác nhận"  B4: Đối với Telegram:  - Người dùng nhập Telegram ID/@username  - Hệ thống gửi tin nhắn xác minh đến Telegram bot  - Người dùng xác nhận liên kết qua tin nhắn Telegram  B5: Hệ thống nhận yêu cầu POST với action = 'SETUP\_SECURITY' và lọc dữ liệu đầu vào bằng antixss().  B6: Hệ thống xác thực quyền người dùng thông qua session.  • Nếu không hợp lệ → [A1] và kết thúc use case.  B7: Hệ thống kiểm tra tính hợp lệ:  • Đối với Gmail: kiểm tra định dạng email, mã xác minh khớp → [A2] nếu lỗi  • Đối với Telegram: kiểm tra Telegram ID tồn tại, xác minh thành công → [A3] nếu lỗi  B8: Hệ thống bắt đầu giao dịch begin\_transaction().  B9: Hệ thống cập nhật thông tin bảo mật vào bảng users hoặc user\_security: security\_email, telegram\_id, security\_status.  • Nếu thất bại → [A4] và kết thúc use case.  B10: Hệ thống commit() giao dịch và phản hồi json\_Msg('success', 'Thiết lập bảo mật thành công').  B11: Hệ thống gửi thông báo xác nhận đến Gmail/Telegram đã liên kết. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 11: Luồng sự kiện phụ UC05

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Người dùng không có quyền thực hiện hành động (session không hợp lệ). | "Bạn không có quyền thực hiện hành động này" |
| **A2** | Lỗi Gmail: - Email không hợp lệ - Mã xác minh không khớp - Không gửi được email | "Email không hợp lệ" "Mã xác minh không chính xác" "Không thể gửi mã xác minh" |
| **A3** | Lỗi Telegram: - Telegram ID không tồn tại - Không kết nối được với Telegram Bot - Xác minh thất bại | "Telegram ID không tồn tại" "Không thể kết nối Telegram" "Xác minh Telegram thất bại" |
| **A4** | Lỗi hệ thống khi cập nhật cơ sở dữ liệu. | "Thiết lập bảo mật thất bại" |
| **A5** | Email/Telegram đã được sử dụng bởi tài khoản khác. | "Email/Telegram này đã được sử dụng" |
| **A6** | Quá thời gian xác minh (mã hết hạn). | "Mã xác minh đã hết hạn" |

#### c, Use case Mua hàng & Thanh toán



Hình 2. 5: Use case Mua hàng & Thanh toán

* Mô tả chi tiết Use Case: Chọn gói dịch vụ

Bảng 2. 12: UC06 - Chọn gói dịch vụ

|  |  |
| --- | --- |
| **ID** | UC06 |
| **Tên Use Case** | Chọn gói dịch vụ Hosting |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng xem danh sách các gói hosting đang có, so sánh thông số kỹ thuật và lựa chọn gói dịch vụ phù hợp trước khi tiến hành thanh toán. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu package\_hosting khả dụng. - Các gói hosting đã được quản trị viên thiết lập và có trạng thái "on". |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng đã đăng nhập vào hệ thống (theo sơ đồ). - Người dùng truy cập trang dịch vụ hosting. |
| **Sự kiện kích hoạt** | Người dùng truy cập trang "Dịch vụ Hosting" hoặc "Mua hosting" từ menu chính. |
| **Luồng sự kiện chính** | B1: Người dùng truy cập trang danh sách gói dịch vụ hosting.  B2: Hệ thống truy vấn danh sách các gói hosting từ bảng package\_hosting với điều kiện status = 'on' và sắp xếp theo mức giá hoặc tính năng.  B3: Hệ thống hiển thị danh sách gói dịch vụ với đầy đủ thông tin:  - Tên gói (name\_host)  - Giá tiền (money)  - Dung lượng lưu trữ (disk)  - Số domain phụ (other\_domain)  - Số alias domain (alias\_domain)  - Máy chủ (server\_host)  B4: Người dùng xem và so sánh các gói dịch vụ.  B5: Người dùng chọn một gói dịch vụ phù hợp bằng cách nhấn nút "Chọn" hoặc "Đăng ký" trên gói đó.  B6: Hệ thống lưu thông tin gói đã chọn vào session hoặc giỏ hàng tạm thời.  B7: Hệ thống chuyển hướng người dùng đến trang thanh toán để tiếp tục quy trình. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 13: Luồng sự kiện phụ UC06

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Không có gói dịch vụ nào khả dụng (tất cả đều status = 'off'). | Hiển thị thông báo: "Hiện tại không có gói dịch vụ nào khả dụng. Vui lòng quay lại sau." |
| **A2** | Người dùng chưa đăng nhập. | Chuyển hướng đến trang đăng nhập với thông báo: "Vui lòng đăng nhập để xem dịch vụ" |
| **A3** | Lỗi kết nối cơ sở dữ liệu. | Hiển thị thông báo: "Hệ thống đang bảo trì. Vui lòng thử lại sau." |
| **A4** | Người dùng hủy bỏ việc chọn gói. | Quay lại trang danh sách gói dịch vụ mà không lưu thông tin. |

* Mô tả chi tiết Use Case: Nhập mã giảm giá

Bảng 2. 14: UC07 - Nhập mã giảm giá

|  |  |
| --- | --- |
| **ID** | UC07 |
| **Tên Use Case** | Nhập mã giảm giá |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng nhập mã giảm giá hợp lệ để được giảm giá trên tổng số tiền thanh toán cho gói hosting đã chọn. |
| **Điều kiện tiên quyết** | - Người dùng đã đăng nhập. - Người dùng đã chọn gói dịch vụ hosting (UC06). - Hệ thống đang ở trang thanh toán. - Kết nối đến cơ sở dữ liệu discount khả dụng. |
| **Điều kiện thực hiện** | - Người dùng có mã giảm giá hợp lệ. - Trang thanh toán có trường nhập mã giảm giá. |
| **Sự kiện kích hoạt** | Người dùng nhập mã giảm giá vào trường nhập và nhấn nút "Áp dụng" hoặc "Xác nhận". |
| **Luồng sự kiện chính** | B1: Tại trang thanh toán, người dùng nhập mã giảm giá vào ô "Mã giảm giá".  B2: Người dùng nhấn nút "Áp dụng" hoặc "Xác nhận".  B3: Hệ thống nhận yêu cầu (AJAX/POST) với action = 'APPLY\_DISCOUNT' và mã giảm giá từ người dùng.  B4: Hệ thống lọc và làm sạch dữ liệu đầu vào bằng antixss().  B5: Hệ thống truy vấn cơ sở dữ liệu để kiểm tra mã giảm giá trong bảng discount với các điều kiện:  - code = mã nhập vào  - status = 'on'  - amount > used  - enddate > thời gian hiện tại HOẶC enddate = 0 (không giới hạn)  - createdate <= thời gian hiện tại  B6: Nếu mã hợp lệ:  - Hệ thống kiểm tra điều kiện áp dụng:  • Tổng tiền đơn hàng >= min (nếu có)  • Tổng tiền giảm <= max (nếu có, đối với loại percent)  - Tính toán số tiền được giảm:  • Nếu type = 'fixed': giảm trực tiếp discount  • Nếu type = 'percent': giảm = (tổng tiền \* discount / 100)  B7: Hệ thống cập nhật tổng tiền thanh toán mới và hiển thị thông tin giảm giá.  B8: Hệ thống phản hồi json\_Msg('success', 'Áp dụng mã giảm giá thành công') cùng với thông tin chi tiết về số tiền được giảm. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 15: Luồng sự kiện phụ UC07

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Mã giảm giá không tồn tại. | "Mã giảm giá không tồn tại" |
| **A2** | Mã giảm giá đã hết hạn sử dụng. | "Mã giảm giá đã hết hạn" |
| **A3** | Mã giảm giá đã được sử dụng hết lượt. | "Mã giảm giá đã hết lượt sử dụng" |
| **A4** | Mã giảm giá chưa đến thời gian áp dụng. | "Mã giảm giá chưa đến thời gian áp dụng" |
| **A5** | Đơn hàng không đạt giá trị tối thiểu để áp dụng. | "Đơn hàng cần tối thiểu [min] để áp dụng mã này" |
| **A6** | Số tiền giảm vượt quá mức tối đa cho phép. | "Số tiền giảm tối đa cho mã này là [max]" |
| **A7** | Mỗi người dùng chỉ được sử dụng mã này một lần (nếu có quy định). | "Bạn đã sử dụng mã giảm giá này trước đây" |
| **A8** | Lỗi hệ thống khi kiểm tra mã. | "Lỗi hệ thống, vui lòng thử lại" |

* Mô tả chi tiết Use Case: Thanh toán

Bảng 2. 16: UC08 - Thanh toán

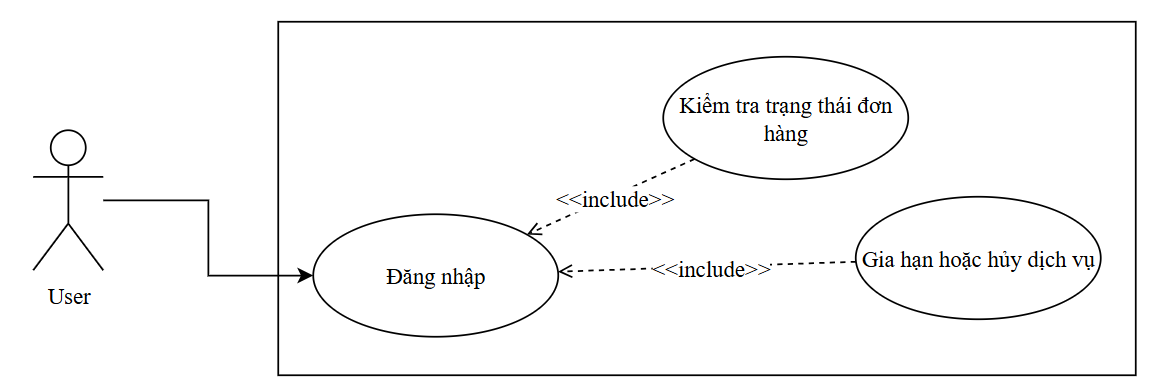
|  |  |
| --- | --- |
| **ID** | UC08 |
| **Tên Use Case** | Thanh toán gói dịch vụ Hosting |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng hoàn tất quá trình thanh toán cho gói hosting đã chọn, kích hoạt dịch vụ và nhận thông tin truy cập. |
| **Điều kiện tiên quyết** | - Người dùng đã đăng nhập (UC01). - Người dùng đã chọn gói dịch vụ hosting (UC06). - Thông tin giỏ hàng đã được thiết lập. - Kết nối đến cổng thanh toán khả dụng. |
| **Điều kiện thực hiện** | - Người dùng có đủ số dư/tài khoản thanh toán. - Người dùng đang ở trang xác nhận thanh toán. |
| **Sự kiện kích hoạt** | Người dùng nhấn nút "Thanh toán" hoặc "Xác nhận đặt hàng" sau khi đã xem xét thông tin đơn hàng. |
| **Luồng sự kiện chính** | B1: Hệ thống hiển thị trang xác nhận thanh toán với đầy đủ thông tin:  - Thông tin gói hosting đã chọn  - Giá gốc và giá sau giảm (nếu có mã giảm giá)  - Phương thức thanh toán khả dụng  B2: Người dùng xác nhận thông tin đơn hàng và chọn phương thức thanh toán (Ví dụ: Ví điện tử, Chuyển khoản ngân hàng, Thẻ cào).  B3: Người dùng nhấn nút "Xác nhận thanh toán".  B4: Hệ thống nhận yêu cầu POST với action = 'PROCESS\_PAYMENT' và kiểm tra:  - Tính hợp lệ của session người dùng  - Tính toàn vẹn của thông tin giỏ hàng  - Sự khả dụng của gói dịch vụ  B5: Hệ thống bắt đầu giao dịch begin\_transaction().  B6: Xử lý thanh toán:  - Ghi nhận giao dịch vào bảng transactions  - Trừ số dư tài khoản người dùng (nếu thanh toán qua ví)  - Cập nhật số lượt sử dụng mã giảm giá (nếu có)  B7: Kích hoạt dịch vụ:  - Tạo tài khoản hosting mới trong bảng user\_hosting  - Tạo thông tin đăng nhập (username/password)  - Gán tài nguyên theo gói đã chọn  B8: Hệ thống commit() giao dịch.  B9: Hệ thống gửi email xác nhận thanh toán và thông tin dịch vụ đến người dùng.  B10: Hệ thống chuyển hướng đến trang "Thanh toán thành công" và hiển thị thông tin truy cập hosting. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 17: Luồng sự kiện phụ UC08

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Số dư tài khoản không đủ để thanh toán. | "Số dư không đủ. Vui lòng nạp thêm tiền vào tài khoản." |
| **A2** | Kết nối đến cổng thanh toán thất bại. | "Hệ thống thanh toán đang bảo trì. Vui lòng thử lại sau." |
| **A3** | Mã giảm giá đã hết lượt sử dụng (kiểm tra cuối cùng). | "Mã giảm giá đã hết lượt sử dụng. Vui lòng kiểm tra lại." |
| **A4** | Gói dịch vụ không còn khả dụng. | "Gói dịch vụ không còn khả dụng. Vui lòng chọn gói khác." |
| **A5** | Lỗi hệ thống khi kích hoạt dịch vụ hosting. | "Kích hoạt dịch vụ thất bại. Vui lòng liên hệ hỗ trợ." |
| **A6** | Người dùng hủy thanh toán giữa chừng. | "Bạn đã hủy quá trình thanh toán." |
| **A7** | Timeout phiên đăng nhập. | "Phiên đăng nhập đã hết hạn. Vui lòng đăng nhập lại." |

#### d, Use case quản lý dịch vụ đã mua



Hình 2. 6: Use case quản lý dịch vụ đã mua

* Mô tả chi tiết Use Case: Kiểm tra trạng thái đơn hàng

Bảng 2. 18: UC09 - Kiểm tra trạng thái đơn hàng

|  |  |
| --- | --- |
| **ID** | UC09 |
| **Tên Use Case** | Kiểm tra trạng thái đơn hàng/dịch vụ |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng theo dõi trạng thái các đơn hàng, dịch vụ hosting đã đăng ký và thực hiện các thao tác quản lý như gia hạn hoặc hủy dịch vụ. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu transactions và user\_hosting khả dụng. - Người dùng có ít nhất một đơn hàng hoặc dịch vụ đã đăng ký. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng đã đăng nhập vào hệ thống. |
| **Sự kiện kích hoạt** | Người dùng truy cập trang "Quản lý đơn hàng" hoặc "Dịch vụ của tôi" từ menu tài khoản. |
| **Luồng sự kiện chính** | B1: Người dùng truy cập trang quản lý đơn hàng/dịch vụ.  B2: Hệ thống truy vấn danh sách các đơn hàng và dịch vụ từ người dùng hiện tại từ các bảng:  - transactions (cho lịch sử thanh toán)  - user\_hosting (cho dịch vụ hosting đang hoạt động)  B3: Hệ thống hiển thị danh sách với các thông tin:  - Mã đơn hàng/Dịch vụ  - Tên gói dịch vụ  - Ngày đăng ký  - Trạng thái (Đang chờ, Đã kích hoạt, Đã hủy)  - Thời hạn sử dụng  - Nút hành động (Xem chi tiết, Gia hạn, Hủy)  B4: Người dùng xem danh sách và chọn một đơn hàng/dịch vụ để xem chi tiết.  B5: Hệ thống hiển thị trang chi tiết với đầy đủ thông tin:  - Thông tin thanh toán  - Thông số kỹ thuật hosting  - Lịch sử gia hạn  - Log hoạt động dịch vụ  B6: Người dùng kiểm tra trạng thái và đưa ra quyết định quản lý. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 19: Luồng sự kiện phụ UC09

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Người dùng chưa đăng nhập. | Chuyển hướng đến trang đăng nhập với thông báo: "Vui lòng đăng nhập để xem đơn hàng" |
| **A2** | Không có đơn hàng/dịch vụ nào. | Hiển thị thông báo: "Bạn chưa có đơn hàng nào. Hãy đăng ký dịch vụ để bắt đầu." |
| **A3** | Lỗi kết nối cơ sở dữ liệu. | Hiển thị thông báo: "Hệ thống đang bảo trì. Vui lòng thử lại sau." |
| **A4** | Không tìm thấy thông tin chi tiết đơn hàng. | Hiển thị thông báo: "Không tìm thấy thông tin đơn hàng." |

* Mô tả chi tiết Use Case: Gia hạn hoặc hủy dịch vụ

Bảng 2. 20: UC10 - Gia hạn hoặc hủy dịch vụ

|  |  |
| --- | --- |
| **ID** | UC10.1 – Gia hạn dịch vụ  UC10.2 – Hủy dịch vụ |
| **Tên Use Case** | Gia hạn hoặc hủy dịch vụ Hosting |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng chủ động gia hạn để tiếp tục sử dụng dịch vụ hosting hoặc hủy dịch vụ khi không còn nhu cầu sử dụng. |
| **Điều kiện tiên quyết** | - Người dùng đã đăng nhập (UC01). - Người dùng có ít nhất một dịch vụ hosting đang hoạt động hoặc sắp hết hạn. - Dịch vụ tồn tại và có thể thao tác được. |
| **Điều kiện thực hiện** | - Người dùng đang ở trang quản lý dịch vụ chi tiết. - Đối với gia hạn: dịch vụ chưa bị hủy và còn khả dụng. - Đối với hủy: dịch vụ không ở trạng thái đã hủy. |
| **Sự kiện kích hoạt** | Người dùng nhấn nút "Gia hạn" hoặc "Hủy dịch vụ" trên giao diện quản lý dịch vụ chi tiết. |

* Luồng sự kiện chính (Main Flow)

Bảng 2. 21: Luồng sự kiện chính UC10.1

|  |  |
| --- | --- |
| **Bước** | **Mô tả hành động** |
| **B1** | Từ trang chi tiết dịch vụ, người dùng nhấn nút "Gia hạn". |
| **B2** | Hệ thống kiểm tra điều kiện gia hạn: - Dịch vụ có tồn tại và thuộc quyền sở hữu - Dịch vụ chưa bị hủy - Gói dịch vụ vẫn còn được cung cấp |
| **B3** | Hệ thống hiển thị form gia hạn với các tùy chọn: - Thời gian gia hạn (1 tháng, 6 tháng, 1 năm) - Tổng tiền cần thanh toán - Phương thức thanh toán khả dụng |
| **B4** | Người dùng chọn thời gian gia hạn và nhấn "Xác nhận gia hạn". |
| **B5** | Hệ thống chuyển hướng đến trang thanh toán (tương tự UC08) để xử lý khoản phí gia hạn. |
| **B6** | Sau khi thanh toán thành công, hệ thống cập nhật thời hạn dịch vụ trong bảng user\_hosting: - end\_date = end\_date + thời gian gia hạn - status = 'active' (nếu đã hết hạn) |
| **B7** | Hệ thống ghi nhận giao dịch gia hạn vào bảng renewal\_history. |
| **B8** | Hệ thống gửi email xác nhận gia hạn thành công và hiển thị thông báo cho người dùng. |

* Luồng sự kiện chính (Main Flow)

Bảng 2. 22: Luồng sự kiện chính UC10.2

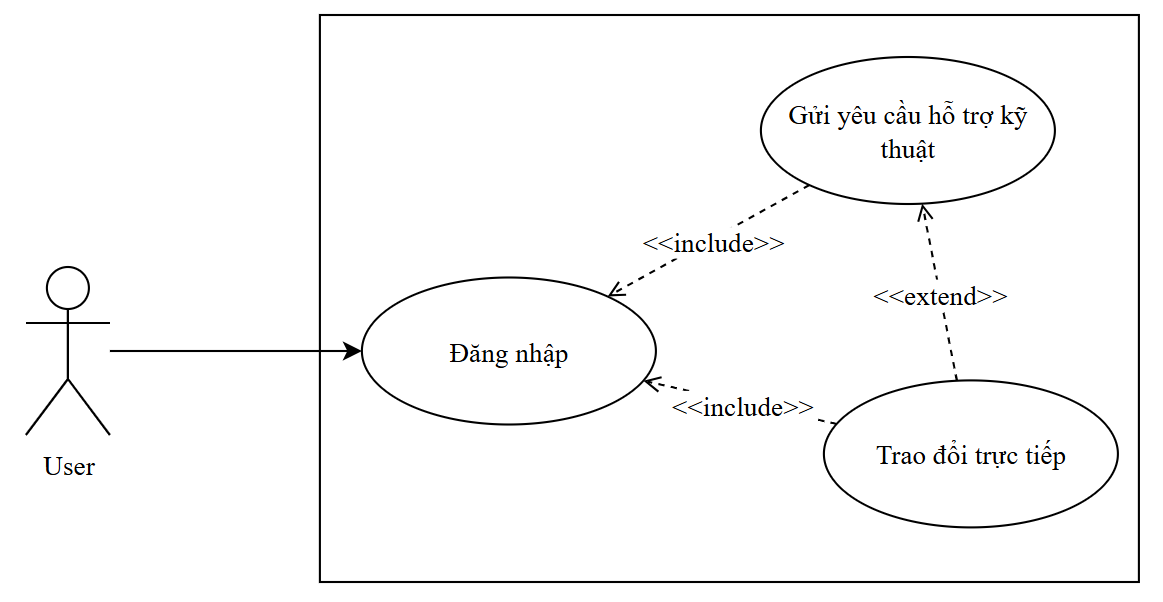
|  |  |
| --- | --- |
| **Bước** | **Mô tả hành động** |
| **B1** | Từ trang chi tiết dịch vụ, người dùng nhấn nút "Hủy dịch vụ". |
| **B2** | Hệ thống kiểm tra điều kiện hủy: - Dịch vụ tồn tại và đang hoạt động - Dịch vụ chưa bị hủy trước đó - Không có giao dịch đang chờ xử lý |
| **B3** | Hệ thống hiển thị popup xác nhận với cảnh báo quan trọng: - Tất cả dữ liệu sẽ bị xóa vĩnh viễn - Không thể khôi phục dịch vụ sau khi hủy - Chính sách hoàn tiền (nếu có) |
| **B4** | Người dùng nhập lý do hủy (tùy chọn) và xác nhận hủy dịch vụ. |
| **B5** | Hệ thống bắt đầu giao dịch begin\_transaction(). |
| **B6** | Hệ thống cập nhật trạng thái dịch vụ: - status = 'cancelled' trong bảng user\_hosting - cancelled\_at = thời gian hiện tại - cancellation\_reason = lý do hủy |
| **B7** | Hệ thống thu hồi tài nguyên và lên lịch xóa dữ liệu. |
| **B8** | Hệ thống commit() giao dịch. |
| **B9** | Hệ thống gửi email xác nhận hủy dịch vụ và hiển thị thông báo thành công. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 23: Luồng sự kiện phụ UC10

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Dịch vụ không tồn tại hoặc không thuộc quyền sở hữu. | "Không tìm thấy dịch vụ hoặc bạn không có quyền gia hạn." |
| **A2** | Dịch vụ đã bị hủy trước đó. | "Không thể gia hạn dịch vụ đã hủy." |
| **A3** | Gói dịch vụ không còn được cung cấp. | "Gói dịch vụ này không còn khả dụng để gia hạn." |
| **A4** | Thanh toán thất bại. | "Thanh toán thất bại. Vui lòng thử lại hoặc liên hệ hỗ trợ." |
| **A5** | Dịch vụ không tồn tại hoặc không thuộc quyền sở hữu. | "Không tìm thấy dịch vụ hoặc bạn không có quyền hủy." |
| **A6** | Dịch vụ đã bị hủy trước đó. | "Dịch vụ này đã được hủy trước đó." |
| **A7** | Có giao dịch đang chờ xử lý. | "Không thể hủy khi có giao dịch đang chờ xử lý." |
| **A8** | Người dùng hủy bỏ xác nhận. | "Đã hủy bỏ thao tác hủy dịch vụ." |

#### e, Use case ticket hỗ trợ



Hình 2. 7: Use case ticket hỗ trợ

* Mô tả chi tiết Use Case: Gửi yêu cầu hỗ trợ kỹ thuật

Bảng 2. 24: UC11 - Gửi yêu cầu hỗ trợ kỹ thuật

|  |  |
| --- | --- |
| **ID** | UC11 |
| **Tên Use Case** | Gửi yêu cầu hỗ trợ kỹ thuật |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng gửi yêu cầu hỗ trợ kỹ thuật về các vấn đề liên quan đến dịch vụ hosting để được đội ngũ kỹ thuật giải quyết. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu support\_tickets khả dụng. - Người dùng có ít nhất một dịch vụ hosting đang hoạt động. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng đã đăng nhập vào hệ thống. |
| **Sự kiện kích hoạt** | Người dùng gặp sự cố kỹ thuật và truy cập trang "Hỗ trợ kỹ thuật" để gửi yêu cầu. |
| **Luồng sự kiện chính** | B1: Người dùng truy cập trang "Hỗ trợ kỹ thuật" từ menu tài khoản.  B2: Người dùng nhấn nút "Tạo yêu cầu hỗ trợ mới".  B3: Hệ thống hiển thị form tạo yêu cầu với các trường:  - Tiêu đề yêu cầu  - Dịch vụ liên quan (chọn từ danh sách hosting)  - Mức độ ưu tiên (Cao, Trung bình, Thấp)  - Mô tả chi tiết vấn đề  - Tệp đính kèm (log, screenshot)  B4: Người dùng điền đầy đủ thông tin và nhấn nút "Gửi yêu cầu".  B5: Hệ thống nhận yêu cầu POST với action = 'CREATE\_SUPPORT\_TICKET' và kiểm tra:  - Tính hợp lệ của session người dùng  - Thông tin bắt buộc đã được nhập  B6: Hệ thống bắt đầu giao dịch begin\_transaction().  B7: Hệ thống tạo bản ghi mới trong bảng support\_tickets:  - Mã ticket tự động sinh  - Thông tin người gửi, dịch vụ liên quan  - Trạng thái: "Mới"  - Mức độ ưu tiên  - Nội dung chi tiết  B8: Hệ thống commit() giao dịch.  B9: Hệ thống gửi email xác nhận đến người dùng và thông báo đến đội ngũ hỗ trợ.  B10: Hệ thống hiển thị thông báo thành công và mã ticket cho người dùng. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 25: Luồng sự kiện phụ UC11

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Người dùng chưa đăng nhập. | Chuyển hướng đến trang đăng nhập với thông báo: "Vui lòng đăng nhập để gửi yêu cầu hỗ trợ" |
| **A2** | Thiếu thông tin bắt buộc (tiêu đề, mô tả). | "Vui lòng điền đầy đủ thông tin bắt buộc" |
| **A3** | Không có dịch vụ hosting nào đang hoạt động. | "Bạn cần có ít nhất một dịch vụ hosting đang hoạt động để gửi yêu cầu hỗ trợ" |
| **A4** | Lỗi hệ thống khi tạo ticket. | "Tạo yêu cầu thất bại. Vui lòng thử lại sau." |

* Mô tả chi tiết Use Case: Trao đổi trực tiếp

Bảng 2. 26: UC12 - Trao đổi trực tiếp

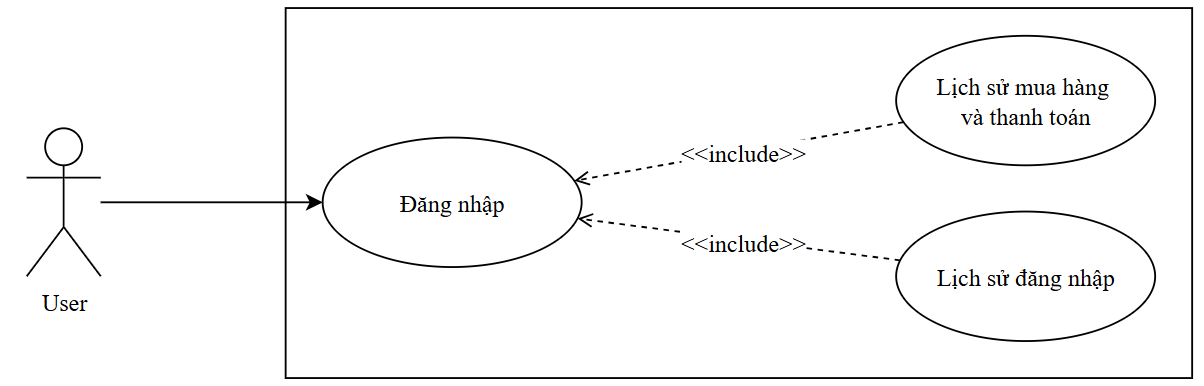
|  |  |
| --- | --- |
| **ID** | UC12 |
| **Tên Use Case** | Trao đổi trực tiếp (Live Chat) |
| **Tác nhân chính** | Khách hàng (Customer), Nhân viên hỗ trợ (Support Staff) |
| **Mục tiêu** | Cho phép khách hàng và nhân viên hỗ trợ trao đổi trực tiếp qua chat để giải quyết nhanh các vấn đề kỹ thuật. |
| **Điều kiện tiên quyết** | - Hệ thống chat hoạt động bình thường. - Có nhân viên hỗ trợ trực tuyến. - Người dùng đã có ticket hỗ trợ đang mở (từ UC11). |
| **Điều kiện thực hiện** | - Người dùng đang xem chi tiết ticket hỗ trợ. - Tính năng chat khả dụng. |
| **Sự kiện kích hoạt** | Người dùng nhấn nút "Chat trực tiếp" hoặc nhân viên hỗ trợ bắt đầu phiên chat. |
| **Luồng sự kiện chính** | B1: Từ trang chi tiết ticket hỗ trợ, người dùng nhấn nút "Chat trực tiếp".  B2: Hệ thống kiểm tra tính khả dụng của nhân viên hỗ trợ.  B3: Hệ thống tạo phiên chat mới hoặc kết nối đến phiên chat hiện có.  B4: Hệ thống hiển thị giao diện chat với:  - Thông tin ticket liên quan  - Lịch sử trao đổi (nếu có)  - Khung nhập tin nhắn  - Danh sách nhân viên hỗ trợ đang tham gia  B5: Người dùng và nhân viên hỗ trợ trao đổi thông tin qua lại.  B6: Hệ thống lưu tự động lịch sử chat vào cơ sở dữ liệu.  B7: Khi vấn đề được giải quyết, nhân viên hỗ trợ hoặc người dùng kết thúc phiên chat.  B8: Hệ thống lưu trữ toàn bộ lịch sử chat và cập nhật trạng thái ticket. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 27: Luồng sự kiện phụ UC12

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Không có nhân viên hỗ trợ trực tuyến. | "Hiện không có nhân viên hỗ trợ trực tuyến. Vui lòng thử lại sau hoặc để lại tin nhắn." |
| **A2** | Kết nối chat bị gián đoạn. | "Mất kết nối. Đang thử kết nối lại..." |
| **A3** | Ticket không tồn tại hoặc đã đóng. | "Không thể chat do ticket không tồn tại hoặc đã đóng." |
| **A4** | Lỗi hệ thống chat. | "Hệ thống chat đang bảo trì. Vui lòng sử dụng hình thức hỗ trợ khác." |

#### f, Use case lịch sự hoạt động



Hình 2. 8: Use case lịch sự hoạt động

* Mô tả chi tiết Use Case: Lịch sử mua hàng và thanh toán

Bảng 2. 28: UC13 - Lịch sử mua hàng và thanh toán

|  |  |
| --- | --- |
| **ID** | UC13 |
| **Tên Use Case** | Xem lịch sử mua hàng và thanh toán |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng theo dõi toàn bộ lịch sử mua hàng, thanh toán các gói hosting và dịch vụ đã sử dụng trong tài khoản. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu transactions, user\_hosting khả dụng. - Người dùng có ít nhất một giao dịch hoặc dịch vụ đã đăng ký. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng đã đăng nhập vào hệ thống. |
| **Sự kiện kích hoạt** | Người dùng truy cập trang "Lịch sử mua hàng" hoặc "Lịch sử thanh toán" từ menu tài khoản. |
| **Luồng sự kiện chính** | B1: Người dùng truy cập trang "Lịch sử mua hàng" từ menu tài khoản.  B2: Hệ thống truy vấn lịch sử giao dịch từ bảng transactions với điều kiện user\_id = người dùng hiện tại.  B3: Hệ thống sắp xếp kết quả theo thời gian giao dịch giảm dần (mới nhất lên đầu).  B4: Hệ thống hiển thị danh sách lịch sử với các thông tin:  - Mã giao dịch  - Ngày giờ giao dịch  - Tên gói dịch vụ  - Số tiền  - Phương thức thanh toán  - Trạng thái giao dịch  - Nút "Xem chi tiết"  B5: Người dùng có thể:  - Xem danh sách tổng quan  - Lọc theo khoảng thời gian  - Tìm kiếm theo mã giao dịch  - Xem chi tiết từng giao dịch  B6: Khi người dùng nhấn "Xem chi tiết", hệ thống hiển thị đầy đủ thông tin hóa đơn. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 29: Luồng sự kiện phụ UC13

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Người dùng chưa đăng nhập. | Chuyển hướng đến trang đăng nhập với thông báo: "Vui lòng đăng nhập để xem lịch sử mua hàng" |
| **A2** | Không có giao dịch nào. | Hiển thị thông báo: "Bạn chưa có giao dịch nào." kèm nút "Đăng ký dịch vụ" |
| **A3** | Lỗi kết nối cơ sở dữ liệu. | Hiển thị thông báo: "Hệ thống đang bảo trì. Vui lòng thử lại sau." |

* Mô tả chi tiết Use Case: Lịch sử đăng nhập nhập

Bảng 2. 30: UC14 - Lịch sử đăng nhập nhập

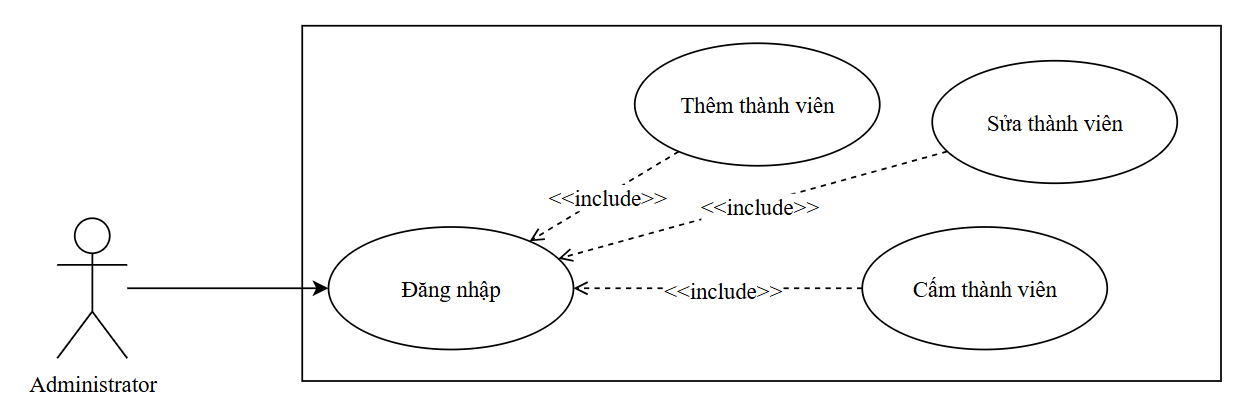
|  |  |
| --- | --- |
| **ID** | UC14 |
| **Tên Use Case** | Xem lịch sử đăng nhập |
| **Tác nhân chính** | Khách hàng (Customer) |
| **Mục tiêu** | Cho phép khách hàng theo dõi lịch sử đăng nhập vào tài khoản, phát hiện các hoạt động đăng nhập bất thường để bảo vệ tài khoản. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu login\_history khả dụng. - Người dùng đã từng đăng nhập ít nhất một lần. |
| **Điều kiện thực hiện** | - Người dùng có thiết bị kết nối Internet ổn định. - Người dùng đã đăng nhập vào hệ thống. |
| **Sự kiện kích hoạt** | Người dùng truy cập trang "Lịch sử đăng nhập" hoặc "Hoạt động tài khoản" từ menu bảo mật. |
| **Luồng sự kiện chính** | B1 Người dùng truy cập trang "Lịch sử đăng nhập" từ menu tài khoản.  B2 Hệ thống truy vấn lịch sử đăng nhập từ bảng login\_history với điều kiện user\_id = người dùng hiện tại.  B3 Hệ thống sắp xếp kết quả theo thời gian đăng nhập giảm dần.  B4 Hệ thống hiển thị danh sách lịch sử với các thông tin:  - Thời gian đăng nhập  - Địa chỉ IP  - Thiết bị & trình duyệt  - Vị trí (dựa trên IP)  - Trạng thái (thành công/thất bại)  B5 Người dùng có thể:  - Xem các lần đăng nhập gần đây  - Phát hiện đăng nhập từ thiết bị/địa điểm lạ  - Lọc theo khoảng thời gian  B6 Nếu phát hiện hoạt động đáng ngờ, người dùng có thể thực hiện:  - Đổi mật khẩu  - Đăng xuất toàn bộ thiết bị  - Báo cáo với quản trị viên |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 31: Luồng sự kiện phụ UC14

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Người dùng chưa đăng nhập. | Chuyển hướng đến trang đăng nhập với thông báo: "Vui lòng đăng nhập để xem lịch sử đăng nhập" |
| **A2** | Không có lịch sử đăng nhập nào. | Hiển thị thông báo: "Chưa có lịch sử đăng nhập nào được ghi nhận." |
| **A3** | Lỗi kết nối cơ sở dữ liệu. | Hiển thị thông báo: "Hệ thống đang bảo trì. Vui lòng thử lại sau." |

#### h, Use case quản lý thành viên



Hình 2. 9: Use case quản lý thành viên

* Mô tả chi tiết Use Case: Thêm thành viên

Bảng 2. 32: UC15 - Thêm thành viên

|  |  |
| --- | --- |
| **ID** | UC15 |
| **Tên Use Case** | Thêm thành viên mới |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên tạo tài khoản mới cho người dùng (thành viên) trong hệ thống, thiết lập quyền và thông tin ban đầu. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu users khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. |
| **Điều kiện thực hiện** | - Quản trị viên có quyền thêm người dùng. - Truy cập đúng trang quản lý thành viên. |
| **Sự kiện kích hoạt** | Quản trị viên nhấn nút "Thêm thành viên mới" trên trang quản lý. |
| **Luồng sự kiện chính** | B1: Quản trị viên truy cập trang quản lý thành viên và nhấn "Thêm mới".  B2: Hệ thống hiển thị form thêm thành viên với các trường:  - Username (bắt buộc)  - Email (bắt buộc)  - Mật khẩu (bắt buộc)  - Số điện thoại  - Địa chỉ  - Cấp độ (level): User, Admin, v.v.  - Trạng thái (status)  B3: Quản trị viên nhập thông tin và nhấn "Lưu".  B4: Hệ thống nhận yêu cầu POST với action='ADD\_USER', lọc dữ liệu bằng antixss().  B5: Kiểm tra hợp lệ:  - Username không trùng  - Email không trùng  - Mật khẩu đủ mạnh (>6 ký tự)  - Email hợp lệ  B6: Mã hóa mật khẩu bằng TypePassword().  B7: Bắt đầu giao dịch begin\_transaction().  B8: Thêm bản ghi mới vào bảng users với các thông tin đã nhập.  B9: commit() giao dịch.  B10: Phản hồi thành công và hiển thị thông báo. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 33: Luồng sự kiện phụ UC15

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Username đã tồn tại. | "Username đã tồn tại." |
| **A2** | Email đã được sử dụng. | "Email đã được sử dụng." |
| **A3** | Thiếu thông tin bắt buộc. | "Vui lòng nhập đủ thông tin bắt buộc." |
| **A4** | Lỗi hệ thống khi thêm dữ liệu. | "Thêm thành viên thất bại." |

* Mô tả chi tiết Use Case: Sửa thành viên

Bảng 2. 34: UC16 - Sửa thành viên

|  |  |
| --- | --- |
| **ID** | UC16 |
| **Tên Use Case** | Chỉnh sửa thông tin thành viên |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên chỉnh sửa thông tin, quyền hạn và trạng thái của một thành viên đã có trong hệ thống. |
| **Điều kiện tiên quyết** | - Thành viên cần sửa đã tồn tại trong hệ thống. - Quản trị viên có quyền chỉnh sửa thông tin thành viên. |
| **Sự kiện kích hoạt** | Quản trị viên chọn một thành viên từ danh sách và nhấn nút "Sửa". |
| **Luồng sự kiện chính** | B1: Quản trị viên chọn thành viên từ danh sách, nhấn "Sửa".  B2: Hệ thống hiển thị form chỉnh sửa với thông tin hiện tại.  B3: Quản trị viên thay đổi các thông tin có thể chỉnh sửa:  - Email  - Số điện thoại  - Địa chỉ  - Cấp độ (level)  - Số dư (money)  - Trạng thái (status)  B4: Quản trị viên nhấn "Cập nhật".  B5: Hệ thống nhận POST với action='UPDATE\_USER', lọc dữ liệu.  B6: Kiểm tra hợp lệ:  - Email không trùng với thành viên khác (nếu thay đổi)  - Các trường hợp lệ  B7: Bắt đầu giao dịch begin\_transaction().  B8: Cập nhật bản ghi trong bảng users.  B9: commit() giao dịch.  B10: Phản hồi thành công. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 35: Luồng sự kiện phụ UC16

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Thành viên không tồn tại. | "Thành viên không tồn tại." |
| **A2** | Email mới trùng với thành viên khác. | "Email đã được sử dụng bởi thành viên khác." |
| **A3** | Lỗi hệ thống khi cập nhật. | "Cập nhật thất bại." |

* Mô tả chi tiết Use Case: Cấm thành viên

Bảng 2. 36: UC17 - Cấm thành viên

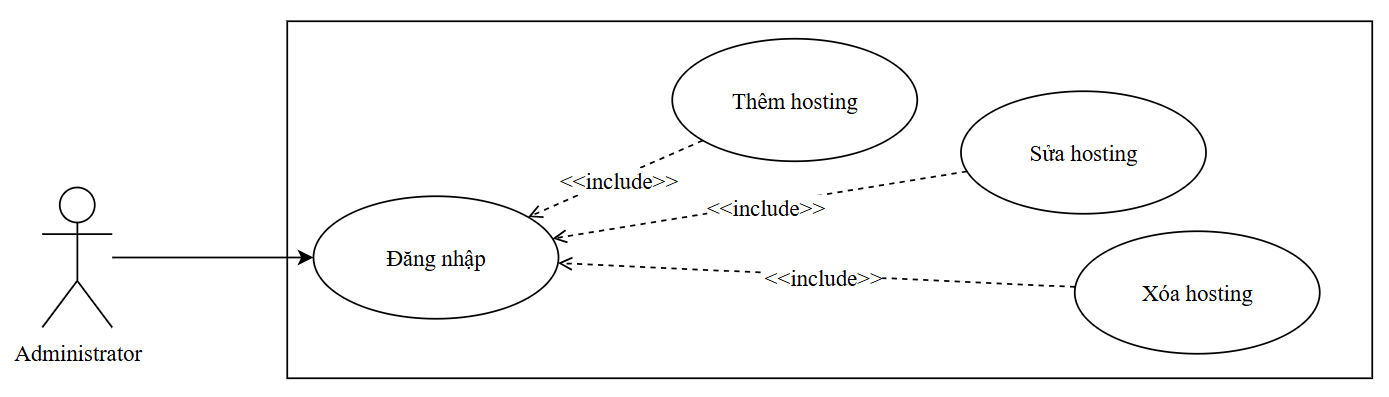
|  |  |
| --- | --- |
| **ID** | UC17 |
| **Tên Use Case** | Cấm/Khóa thành viên |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên vô hiệu hóa (cấm) tài khoản của một thành viên, ngăn không cho đăng nhập và sử dụng dịch vụ. |
| **Điều kiện tiên quyết** | - Thành viên cần cấm đang ở trạng thái hoạt động. - Quản trị viên có quyền cấm thành viên. |
| **Sự kiện kích hoạt** | Quản trị viên chọn một thành viên và nhấn nút "Cấm" hoặc "Khóa tài khoản". |
| **Luồng sự kiện chính** | B1: Quản trị viên chọn thành viên từ danh sách, nhấn "Cấm".  B2: Hệ thống hiển thị popup xác nhận, yêu cầu nhập lý do cấm (tùy chọn).  B3: Quản trị viên xác nhận và có thể nhập lý do.  B4: Hệ thống nhận POST với action='BAN\_USER'.  B5: Bắt đầu giao dịch begin\_transaction().  B6: Cập nhật trường band (hoặc status) trong bảng users thành giá trị thể hiện "bị cấm". Ví dụ: band = 'yes' hoặc status = 'banned'.  B7: Ghi log hành động cấm vào bảng admin\_logs (nếu có).  B8: commit() giao dịch.  B9: Hệ thống tự động đăng xuất thành viên đó nếu họ đang online.  B10: Phản hồi thành công. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 37: Luồng sự kiện phụ UC17

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Thành viên không tồn tại. | "Thành viên không tồn tại." |
| **A2** | Thành viên đã bị cấm trước đó. | "Thành viên này đã bị cấm." |
| **A3** | Không thể cấm tài khoản admin cấp cao. | "Không thể cấm tài khoản quản trị viên cấp cao." |
| **A4** | Lỗi hệ thống. | "Thao tác thất bại." |

#### i, Use case quản lý hosting



Hình 2. 10: Use case quản lý hosting

* Mô tả chi tiết Use Case: Thêm hosting

Bảng 2. 38: UC18 - Thêm hosting

|  |  |
| --- | --- |
| **ID** | UC18 |
| **Tên Use Case** | Thêm gói hosting mới |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên thêm mới một gói hosting vào hệ thống với đầy đủ thông số kỹ thuật và giá cả. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu package\_hosting khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. |
| **Điều kiện thực hiện** | - Quản trị viên có quyền quản lý gói hosting. - Truy cập đúng trang quản lý gói hosting. |
| **Luồng sự kiện chính** | B1: Quản trị viên truy cập trang quản lý gói hosting và nhấn "Thêm mới".  B2: Hệ thống hiển thị form thêm gói hosting với các trường:  - Tên gói (name\_host)  - Máy chủ (server\_host)  - Giá tiền (money)  - Dung lượng (disk)  - Số domain phụ (other\_domain)  - Số alias domain (alias\_domain)  - Trạng thái (status)  B3: Quản trị viên nhập đầy đủ thông tin và nhấn "Lưu".  B4: Hệ thống nhận yêu cầu POST với action='ADD\_PACKAGE\_HOSTING', lọc dữ liệu bằng antixss().  B5: Kiểm tra hợp lệ:  - Các trường bắt buộc không trống  - Giá tiền là số và >0  B6: Chuẩn hóa dữ liệu:  - name\_host\_upper = chuyển name\_host thành chữ hoa, thay space bằng underscore  - code\_host = chuyển name\_host thành chữ thường, thay space bằng underscore  B7: Kiểm tra trùng lặp: Kiểm tra code\_host đã tồn tại trong DB chưa.  B8: Bắt đầu giao dịch begin\_transaction().  B9: Thêm bản ghi mới vào bảng package\_hosting với các thông tin đã chuẩn hóa.  B10: commit() giao dịch.  B11: Phản hồi json\_Msg('success', 'Thêm gói hosting thành công'). |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 39: Luồng sự kiện phụ UC18

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Thiếu thông tin bắt buộc. | "Vui lòng nhập đầy đủ thông tin gói hosting" |
| **A2** | Giá tiền không hợp lệ (không phải số hoặc ≤ 0). | "Giá tiền không hợp lệ" |
| **A3** | Tên gói đã tồn tại (code\_host trùng). | "Tên gói này đã tồn tại trong hệ thống (code trùng)" |
| **A4** | Lỗi hệ thống khi thêm dữ liệu. | "Thêm gói hosting thất bại" |

* Mô tả chi tiết Use Case: Sửa hosting

Bảng 2. 40: UC19 - Sửa hosting

|  |  |
| --- | --- |
| **ID** | UC19 |
| **Tên Use Case** | Chỉnh sửa thông tin gói hosting |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên chỉnh sửa thông tin, cấu hình và trạng thái của một gói hosting đã có trong hệ thống. |
| **Điều kiện tiên quyết** | - Gói hosting cần sửa đã tồn tại trong hệ thống. - Quản trị viên có quyền chỉnh sửa gói hosting. |
| **Sự kiện kích hoạt** | Quản trị viên chọn một gói hosting từ danh sách và nhấn nút "Sửa". |
| **Luồng sự kiện chính** | B1: Quản trị viên chọn gói hosting từ danh sách, nhấn "Sửa".  B2: Hệ thống hiển thị form chỉnh sửa với thông tin hiện tại của gói.  B3: Quản trị viên thay đổi các thông tin:  - Tên gói (name\_host)  - Máy chủ (server\_host)  - Giá tiền (money)  - Dung lượng (disk)  - Số domain phụ (other\_domain)  - Số alias domain (alias\_domain)  - Trạng thái (status)  B4: Quản trị viên nhấn "Cập nhật".  B5: Hệ thống nhận POST với action='UPDATE\_PACKAGE\_HOSTING', lọc dữ liệu.  B6: Kiểm tra hợp lệ:  - ID gói hosting hợp lệ và tồn tại  - Các trường bắt buộc không trống  - Giá tiền là số và >0  B7: Chuẩn hóa dữ liệu (tương tự UC18).  B8: Bắt đầu giao dịch begin\_transaction().  B9: Cập nhật bản ghi trong bảng package\_hosting với thông tin mới.  B10: commit() giao dịch.  B11: Phản hồi json\_Msg('success', 'Cập nhật gói hosting thành công'). |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 41: Luồng sự kiện phụ UC19

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | ID gói hosting không hợp lệ (≤ 0). | "Thiếu ID gói hosting" |
| **A2** | Gói hosting không tồn tại. | "Gói hosting không tồn tại" |
| **A3** | Lỗi hệ thống khi cập nhật. | "Cập nhật thất bại" |

* Mô tả chi tiết Use Case: Xóa hosting

Bảng 2. 42: UC20 - Xóa hosting

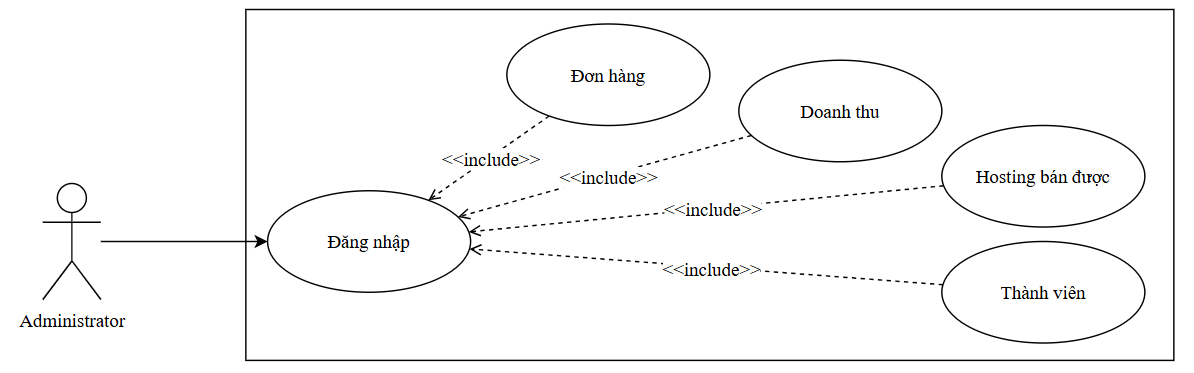
|  |  |
| --- | --- |
| **ID** | UC17 |
| **Tên Use Case** | Cấm/Khóa thành viên |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên vô hiệu hóa (cấm) tài khoản của một thành viên, ngăn không cho đăng nhập và sử dụng dịch vụ. |
| **Điều kiện tiên quyết** | - Thành viên cần cấm đang ở trạng thái hoạt động. - Quản trị viên có quyền cấm thành viên. |
| **Sự kiện kích hoạt** | Quản trị viên chọn một thành viên và nhấn nút "Cấm" hoặc "Khóa tài khoản". |
| **Luồng sự kiện chính** | B1: Quản trị viên chọn gói hosting từ danh sách, nhấn "Xóa".  B2: Hệ thống hiển thị popup xác nhận với cảnh báo về việc xóa vĩnh viễn.  B3: Quản trị viên xác nhận muốn xóa.  B4: Hệ thống nhận POST với action='DELETE\_PACKAGE\_HOSTING'.  B5: Kiểm tra điều kiện:  - ID hợp lệ (≥ 0)  - Gói hosting tồn tại  - Không có người dùng nào đang sử dụng gói này  B6: Bắt đầu giao dịch begin\_transaction().  B7: Xóa bản ghi trong bảng package\_hosting theo điều kiện id.  B8: commit() giao dịch.  B9: Phản hồi json\_Msg('success', 'Đã xóa gói hosting thành công'). |

Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 43: Luồng sự kiện phụ UC20

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | ID không hợp lệ (≤ 0). | "Thiếu ID cần xóa" |
| **A2** | Gói hosting không tồn tại. | "Gói hosting không tồn tại" |
| **A3** | Có người dùng đang sử dụng gói này. | "Không thể xóa gói hosting đang có người dùng sử dụng" |
| **A4** | Lỗi hệ thống khi xóa. | "Xóa gói hosting thất bại" |
| **A5** | Người dùng hủy bỏ xác nhận. | "Đã hủy thao tác xóa" |

#### k, Use case Thống kê & Báo cáo



Hình 2. 11: Use case Thống kê & Báo cáo

* Mô tả chi tiết Use Case: Đơn hàng

Bảng 2. 44: UC21 - Đơn hàng

|  |  |
| --- | --- |
| **ID** | UC21 |
| **Tên Use Case** | Quản lý và xem danh sách đơn hàng |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên xem, tìm kiếm, lọc và quản lý tất cả các đơn hàng/dịch vụ đã được người dùng đăng ký và thanh toán trong hệ thống. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu transactions, user\_hosting khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. |
| **Điều kiện thực hiện** | - Có ít nhất một đơn hàng trong hệ thống. - Truy cập đúng trang quản lý đơn hàng. |
| **Luồng sự kiện chính** | B1: Quản trị viên đăng nhập và truy cập trang "Quản lý đơn hàng".  B2: Hệ thống truy vấn danh sách đơn hàng từ các bảng transactions, user\_hosting, kết hợp với users và package\_hosting.  B3: Hệ thống hiển thị danh sách với các thông tin:  - Mã đơn hàng/Mã dịch vụ  - Tên khách hàng  - Gói dịch vụ  - Số tiền  - Phương thức thanh toán  - Trạng thái (Thành công/Thất bại/Đang chờ)  - Ngày tạo  - Ngày hết hạn  B4: Quản trị viên có thể:  - Tìm kiếm theo mã đơn, tên khách hàng  - Lọc theo trạng thái, khoảng thời gian  - Xem chi tiết từng đơn hàng  - Xuất danh sách ra file Excel  B5: Hệ thống cập nhật danh sách theo các bộ lọc và hiển thị kết quả. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 45: Luồng sự kiện phụ UC21

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Quản trị viên chưa đăng nhập. | Chuyển hướng đến trang đăng nhập. |
| **A2** | Không có đơn hàng nào. | Hiển thị thông báo "Chưa có đơn hàng nào trong hệ thống." |
| **A3** | Lỗi kết nối cơ sở dữ liệu. | Hiển thị thông báo "Lỗi hệ thống, vui lòng thử lại sau." |

* Mô tả chi tiết Use Case: Doanh thu

Bảng 2. 46: UC22 - Doanh thu

|  |  |
| --- | --- |
| **ID** | UC22 |
| **Tên Use Case** | Xem báo cáo tổng quan doanh thu |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên xem báo cáo tổng hợp doanh thu của hệ thống theo các khoảng thời gian khác nhau, với biểu đồ trực quan và số liệu chi tiết. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu transactions khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. |
| **Điều kiện thực hiện** | - Có dữ liệu giao dịch trong hệ thống. - Truy cập đúng trang báo cáo doanh thu. |
| **Luồng sự kiện chính** | B1: Quản trị viên đăng nhập và truy cập trang "Báo cáo doanh thu".  B2: Hệ thống mặc định hiển thị báo cáo doanh thu của tháng hiện tại.  B3: Hệ thống truy vấn và tổng hợp dữ liệu từ bảng transactions với trạng thái thành công.  B4: Hệ thống hiển thị báo cáo với:  - Tổng doanh thu (theo ngày, tuần, tháng, năm)  - Biểu đồ đường doanh thu theo thời gian  - So sánh doanh thu kỳ này với kỳ trước  - Top 5 khách hàng có giá trị đơn hàng cao nhất  B5: Quản trị viên chọn khoảng thời gian cụ thể (Từ ngày - Đến ngày) để xem báo cáo chi tiết.  B6: Hệ thống tải lại báo cáo với dữ liệu trong khoảng thời gian đã chọn.  B7: Quản trị viên có thể xuất báo cáo ra file Excel hoặc PDF. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 47: Luồng sự kiện phụ UC22

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Quản trị viên chưa đăng nhập. | Chuyển hướng đến trang đăng nhập. |
| **A2** | Không có dữ liệu giao dịch nào trong khoảng thời gian đã chọn. | Hiển thị thông báo "Không có dữ liệu doanh thu trong khoảng thời gian này." |
| **A3** | Lỗi khi truy vấn tổng hợp dữ liệu. | Hiển thị thông báo "Lỗi khi tải dữ liệu báo cáo." |

* Mô tả chi tiết Use Case: Hosting bán được

Bảng 2. 48: UC23 - Hosting bán được

|  |  |
| --- | --- |
| **ID** | UC23 |
| **Tên Use Case** | Xem thống kê số lượng hosting bán được |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên thống kê số lượng các gói hosting đã bán được, phân tích hiệu suất bán hàng của từng gói dịch vụ. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu user\_hosting, package\_hosting khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. |
| **Điều kiện thực hiện** | - Có ít nhất một gói hosting đã được bán. - Truy cập đúng trang thống kê hosting. |
| **Luồng sự kiện chính** | B1: Quản trị viên đăng nhập và truy cập trang "Thống kê hosting bán được".  B2: Hệ thống truy vấn dữ liệu từ các bảng user\_hosting và package\_hosting để thống kê số lượng bán theo từng gói.  B3: Hệ thống hiển thị thống kê với:  - Số lượng từng gói hosting đã bán  - Doanh thu tương ứng từng gói  - Biểu đồ cột so sánh các gói  - Tỷ lệ phần trăm thị phần của từng gói  B4: Quản trị viên có thể lọc thống kê theo khoảng thời gian cụ thể.  B5: Hệ thống cập nhật thống kê theo bộ lọc.  B6: Quản trị viên có thể xuất dữ liệu thống kê ra file Excel. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 49: Luồng sự kiện phụ UC23

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Quản trị viên chưa đăng nhập. | Chuyển hướng đến trang đăng nhập. |
| **A2** | Không có gói hosting nào được bán. | Hiển thị thông báo "Chưa có gói hosting nào được bán." |
| **A3** | Lỗi truy vấn dữ liệu. | Hiển thị thông báo "Lỗi khi tải dữ liệu thống kê." |

* Mô tả chi tiết Use Case: Thành viên

Bảng 2. 50: UC24 - Thành viên

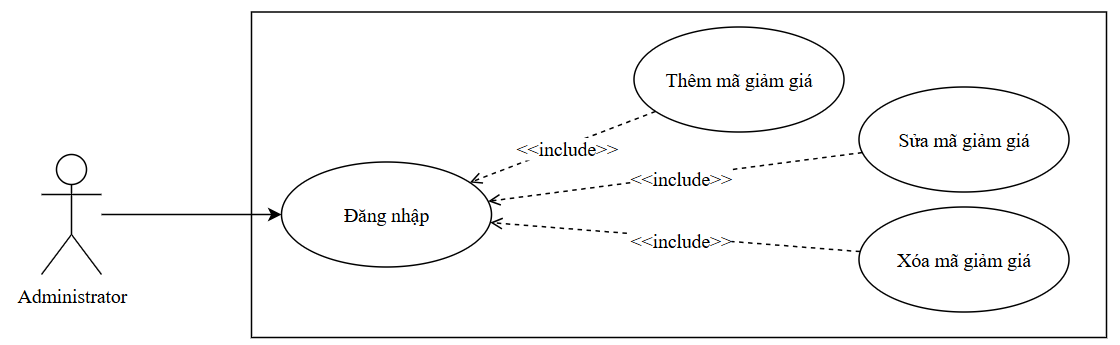
|  |  |
| --- | --- |
| **ID** | UC24 |
| **Tên Use Case** | Xem thống kê tổng quan về thành viên |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên xem các chỉ số về thành viên: tổng số, tăng trưởng, tỷ lệ hoạt động, và phân tích xu hướng đăng ký. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu users khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. |
| **Điều kiện thực hiện** | - Có ít nhất một thành viên trong hệ thống. - Truy cập đúng trang thống kê thành viên. |
| **Luồng sự kiện chính** | B1: Quản trị viên đăng nhập và truy cập trang "Thống kê thành viên".  B2: Hệ thống truy vấn dữ liệu từ bảng users.  B3: Hệ thống hiển thị các chỉ số chính:  - Tổng số thành viên  - Số thành viên mới (hôm nay, tuần này, tháng này)  - Số thành viên đang hoạt động (có dịch vụ)  - Số thành viên bị cấm  - Biểu đồ tăng trưởng thành viên theo thời gian  - Tỷ lệ thành viên có ít nhất một dịch vụ  B4: Quản trị viên có thể lọc theo khoảng thời gian để xem xu hướng đăng ký.  B5: Hệ thống cập nhật thống kê theo bộ lọc.  B6: Quản trị viên có thể xuất báo cáo thành viên. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 51: Luồng sự kiện phụ UC24

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Quản trị viên chưa đăng nhập. | Chuyển hướng đến trang đăng nhập. |
| **A2** | Không có thành viên nào trong hệ thống. | Hiển thị thông báo "Chưa có thành viên nào trong hệ thống." |
| **A3** | Lỗi truy vấn dữ liệu. | Hiển thị thông báo "Lỗi khi tải dữ liệu thống kê." |

#### l, Use case quản lý mã giảm giá



Hình 2. 12: Use case quản lý mã giảm giá

* Mô tả chi tiết use case: Quản lý mã giảm giá

Bảng 2. 52: Use case Quản lý mã giảm giá

|  |  |
| --- | --- |
| **ID** | UC25 - Thêm mã giảm giá  UC26 - Sửa mã giảm giá  UC27 - Xóa mã giảm giá |
| **Tên Use Case** | Quản lý Mã Giảm Giá (Thêm/Sửa/Xóa) |
| **Tác nhân chính** | Quản trị viên (Admin) |
| **Mục tiêu** | Cho phép quản trị viên thực hiện các thao tác CRUD (Create, Read, Update, Delete) trên các mã giảm giá trong hệ thống. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu discount khả dụng. - Người dùng PHẢI có quyền Quản trị viên và đã đăng nhập. |
| **Điều kiện thực hiện** | - Quản trị viên có thiết bị kết nối Internet ổn định. - Quản trị viên truy cập đúng trang quản lý mã giảm giá. |
| **Sự kiện kích hoạt** | Quản trị viên thực hiện một trong các hành động: Thêm mới, Chỉnh sửa hoặc Xóa mã giảm giá. |

* Luồng sự kiện chính (Main Flow)

Bảng 2. 53: Luồng sự kiện chính UC25

|  |  |
| --- | --- |
| **Bước** | **Mô tả hành động** |
| **B1** | Quản trị viên truy cập giao diện thêm mã giảm giá. |
| **B2** | Quản trị viên nhập thông tin: code, amount, discount, type, min, max, createdate, enddate và nhấn nút "Thêm mới". |
| **B3** | Hệ thống nhận yêu cầu POST với action = 'ADD\_DISCOUNT' và lọc dữ liệu đầu vào bằng antixss(). |
| **B4** | Hệ thống chuẩn hóa dữ liệu: - type = "percent" nếu được chọn, ngược lại là "fixed" - Chuyển đổi createdate và enddate từ datetime-local sang timestamp |
| **B5** | Hệ thống kiểm tra tính hợp lệ: • Nếu code trống hoặc amount ≤ 0 → [A1] và kết thúc. • Nếu discount ≤ 0 → [A2] và kết thúc. • Nếu code đã tồn tại trong DB → [A3] và kết thúc. • Nếu hợp lệ, tiếp tục B6. |
| **B6** | Hệ thống bắt đầu giao dịch begin\_transaction(). |
| **B7** | Hệ thống thực hiện thêm bản ghi mới vào bảng discount với status = "on" và used = 0. • Nếu thất bại → [A4] và kết thúc. • Nếu thành công, tiếp tục B8. |
| **B8** | Hệ thống commit() giao dịch và phản hồi json\_Msg('success', 'Thêm mã giảm giá thành công'). |

* Luồng sự kiện chính (Main Flow)

Bảng 2. 54: Luồng sự kiện chính UC26

|  |  |
| --- | --- |
| **Bước** | **Mô tả hành động** |
| **B1** | Quản trị viên chọn mã giảm giá cần sửa từ danh sách. |
| **B2** | Quản trị viên chỉnh sửa thông tin và nhấn nút "Cập nhật". |
| **B3** | Hệ thống nhận yêu cầu POST với action = 'UPDATE\_DISCOUNT' và lọc dữ liệu. |
| **B4** | Hệ thống kiểm tra: • Nếu id ≤ 0 hoặc không tồn tại → [A5] và kết thúc. • Nếu hợp lệ, tiếp tục B5. |
| **B5** | Hệ thống kiểm tra tính hợp lệ: • Nếu code trống → [A6] và kết thúc. • Nếu amount < 0 → [A7] và kết thúc. • Nếu used < 0 → đặt used = 0. • Nếu used > amount → [A8] và kết thúc. • Nếu discount ≤ 0 → [A2] và kết thúc. • Nếu code trùng với bản ghi khác → [A9] và kết thúc. |
| **B6** | Hệ thống bắt đầu giao dịch begin\_transaction(). |
| **B7** | Hệ thống cập nhật bản ghi trong bảng discount. • Nếu thất bại → [A10] và kết thúc. • Nếu thành công, tiếp tục B8. |
| **B8** | Hệ thống commit() và phản hồi json\_Msg('success', 'Cập nhật mã giảm giá thành công'). |

* Luồng sự kiện chính (Main Flow)

Bảng 2. 55: Luồng sự kiện chính UC27

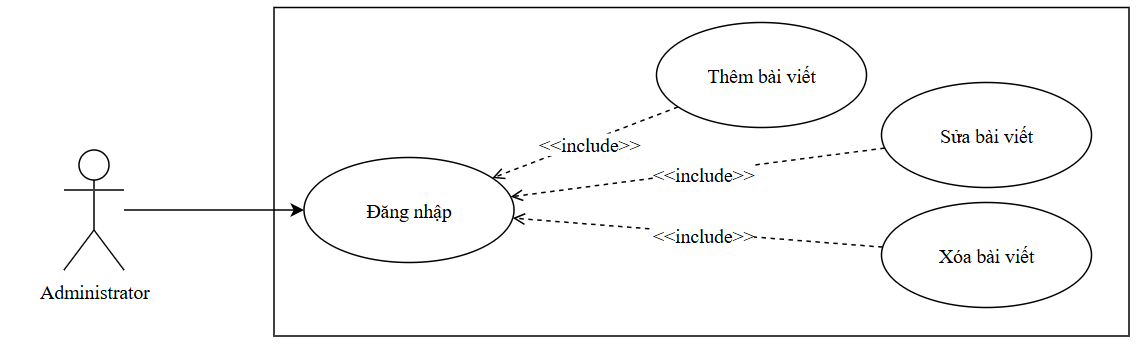
|  |  |
| --- | --- |
| **Bước** | **Mô tả hành động** |
| **B1** | Quản trị viên chọn mã giảm giá cần xóa và xác nhận hành động. |
| **B2** | Hệ thống nhận yêu cầu POST với action = 'DELETE\_DISCOUNT'. |
| **B3** | Hệ thống kiểm tra: • Nếu id ≤ 0 hoặc không tồn tại → [A5] và kết thúc. • Nếu hợp lệ, tiếp tục B4. |
| **B4** | Hệ thống bắt đầu giao dịch begin\_transaction(). |
| **B5** | Hệ thống xóa bản ghi trong bảng discount. • Nếu thất bại → [A11] và kết thúc. • Nếu thành công, tiếp tục B6. |
| **B6** | Hệ thống commit() và phản hồi json\_Msg('success', 'Đã xóa mã giảm giá thành công'). |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 56: Luồng sự kiện phụ use case quản lý mã giảm giá

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Thiếu mã hoặc số lượng không hợp lệ. | "Vui lòng nhập đầy đủ mã giảm giá và số lượng hợp lệ" |
| **A2** | Giá trị giảm không hợp lệ. | "Giá trị giảm không hợp lệ" |
| **A3** | Mã giảm giá đã tồn tại. | "Mã giảm giá này đã tồn tại trong hệ thống" |
| **A4** | Lỗi hệ thống khi thêm dữ liệu. | "Thêm mã giảm giá thất bại" |
| **A5** | ID không hợp lệ hoặc mã không tồn tại. | "Mã giảm giá không tồn tại" |
| **A6** | Mã giảm giá trống. | "Mã giảm giá không được để trống" |
| **A7** | Số lượng không hợp lệ. | "Số lượng không hợp lệ" |
| **A8** | Số đã dùng lớn hơn số lượng. | "Số đã dùng không được lớn hơn số lượng" |
| **A9** | Mã trùng với bản ghi khác. | "Mã giảm giá này đã được sử dụng cho bản ghi khác" |
| **A10** | Lỗi hệ thống khi cập nhật. | "Cập nhật mã giảm giá thất bại" |
| **A11** | Lỗi hệ thống khi xóa dữ liệu. | "Xóa mã giảm giá thất bại" |

#### m, Use case quản lý bài viết



Hình 2. 13: Use case quản lý bài viết

* Mô tả chi tiết Use Case: Thêm bài viết

Bảng 2. 57: UC28 - Thêm bài viết

|  |  |
| --- | --- |
| **ID** | UC28 |
| **Tên Use Case** | Thêm bài viết mới |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên tạo bài viết mới (tin tức, hướng dẫn, blog) để đăng lên website, cung cấp thông tin cho người dùng. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu posts hoặc articles khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. |
| **Điều kiện thực hiện** | - Quản trị viên có quyền thêm bài viết. - Truy cập đúng trang quản lý bài viết. - Có sẵn trình soạn thảo văn bản (WYSIWYG). |
| **Luồng sự kiện chính** | B1: Quản trị viên truy cập trang quản lý bài viết và nhấn "Thêm mới".  B2: Hệ thống hiển thị form soạn thảo bài viết với các trường:  - Tiêu đề (bắt buộc)  - Mô tả ngắn (excerpt)  - Nội dung đầy đủ (sử dụng WYSIWYG editor)  - Danh mục (category)  - Hình ảnh đại diện (featured image)  - Từ khóa (tags)  - Trạng thái (draft, published, pending)  - SEO: slug, meta title, meta description  B3: Quản trị viên nhập đầy đủ thông tin và nhấn "Lưu nháp" hoặc "Xuất bản".  B4: Hệ thống nhận yêu cầu POST với action='ADD\_POST', lọc dữ liệu bằng antixss().  B5: Kiểm tra hợp lệ:  - Tiêu đề không trống  - Nội dung có độ dài tối thiểu  - Slug là duy nhất (nếu tự động sinh từ tiêu đề)  B6: Hệ thống bắt đầu giao dịch begin\_transaction().  B7: Thêm bản ghi mới vào bảng posts với thông tin đã nhập, kèm thông tin tác giả (author\_id) là admin hiện tại.  B8: commit() giao dịch.  B9: Phản hồi json\_Msg('success', 'Thêm bài viết thành công') và hiển thị thông báo. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 58: Luồng sự kiện phụ UC28

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Thiếu tiêu đề hoặc nội dung. | "Vui lòng nhập tiêu đề và nội dung bài viết." |
| **A2** | Slug đã tồn tại. | "Đường dẫn bài viết đã tồn tại, vui lòng chọn slug khác." |
| **A3** | Lỗi khi upload hình ảnh. | "Upload hình ảnh thất bại. Vui lòng thử lại." |
| **A4** | Lỗi hệ thống khi lưu bài viết. | "Lưu bài viết thất bại. Vui lòng thử lại sau." |

* Mô tả chi tiết Use Case: Sửa bài viết

Bảng 2. 59: UC29 - Sửa bài viết

|  |  |
| --- | --- |
| **ID** | UC29 |
| **Tên Use Case** | Chỉnh sửa bài viết |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên chỉnh sửa nội dung, thông tin và trạng thái của một bài viết đã có trong hệ thống. |
| **Điều kiện tiên quyết** | - Bài viết cần sửa đã tồn tại trong hệ thống. - Quản trị viên có quyền chỉnh sửa bài viết (có thể là tác giả hoặc admin cấp cao). |
| **Sự kiện kích hoạt** | Quản trị viên chọn một bài viết từ danh sách và nhấn nút "Sửa". |
| **Luồng sự kiện chính** | B1: Quản trị viên chọn bài viết từ danh sách, nhấn "Sửa".  B2: Hệ thống hiển thị form chỉnh sửa với thông tin hiện tại của bài viết.  B3: Quản trị viên thay đổi các thông tin: tiêu đề, mô tả, nội dung, danh mục, hình ảnh, tags, trạng thái, SEO.  B4: Quản trị viên nhấn "Cập nhật".  B5: Hệ thống nhận POST với action='UPDATE\_POST', lọc dữ liệu.  B6: Kiểm tra hợp lệ:  - Bài viết tồn tại  - Tiêu đề không trống  - Slug mới là duy nhất (nếu thay đổi)  B7: Hệ thống bắt đầu giao dịch begin\_transaction().  B8: Cập nhật bản ghi trong bảng posts với thông tin mới, đồng thời cập nhật thời gian sửa (updated\_at).  B9: commit() giao dịch.  B10: Phản hồi json\_Msg('success', 'Cập nhật bài viết thành công'). |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 60: Luồng sự kiện phụ UC29

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Bài viết không tồn tại. | "Bài viết không tồn tại." |
| **A2** | Quản trị viên không có quyền sửa bài viết này. | "Bạn không có quyền chỉnh sửa bài viết này." |
| **A3** | Slug mới đã tồn tại (trùng với bài khác). | "Đường dẫn này đã được sử dụng cho bài viết khác." |
| **A4** | Lỗi hệ thống khi cập nhật. | "Cập nhật bài viết thất bại." |

* Mô tả chi tiết Use Case: Xóa bài viết

Bảng 2. 61: UC30 - Xóa bài viết

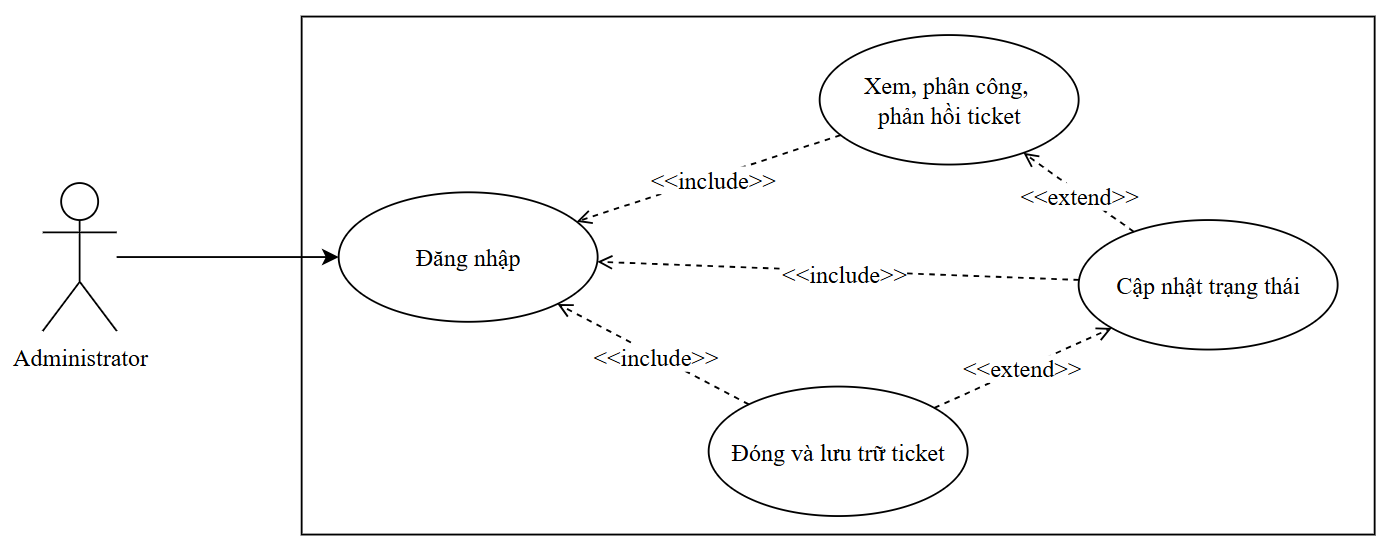
|  |  |
| --- | --- |
| **ID** | UC30 |
| **Tên Use Case** | Xóa bài viết |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên xóa vĩnh viễn hoặc chuyển vào thùng rác một bài viết không còn cần thiết. |
| **Điều kiện tiên quyết** | - Bài viết cần xóa tồn tại trong hệ thống. - Quản trị viên có quyền xóa bài viết. |
| **Sự kiện kích hoạt** | Quản trị viên chọn một bài viết và nhấn nút "Xóa". |
| **Luồng sự kiện chính** | B1: Quản trị viên chọn bài viết từ danh sách, nhấn "Xóa".  B2: Hệ thống hiển thị popup xác nhận: "Bạn có chắc chắn muốn xóa bài viết này? Hành động này không thể hoàn tác."  B3: Quản trị viên xác nhận muốn xóa.  B4: Hệ thống nhận POST với action='DELETE\_POST' và ID bài viết.  B5: Hệ thống kiểm tra:  - Bài viết tồn tại  - Quyền của admin  B6: Hệ thống bắt đầu giao dịch begin\_transaction().  B7: Xóa bản ghi trong bảng posts (hoặc cập nhật deleted\_at nếu xóa mềm).  B8: commit() giao dịch.  B9: Phản hồi json\_Msg('success', 'Đã xóa bài viết thành công'). |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 62: Luồng sự kiện phụ UC30

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Bài viết không tồn tại. | "Bài viết không tồn tại." |
| **A2** | Quản trị viên không có quyền xóa bài viết. | "Bạn không có quyền xóa bài viết này." |
| **A3** | Lỗi hệ thống khi xóa. | "Xóa bài viết thất bại." |
| **A4** | Người dùng hủy bỏ xác nhận. | "Đã hủy thao tác xóa." |

#### n, Use case quản lý hỗ trợ (ticket)



Hình 2. 14: Use case quản lý hỗ trợ (ticket)

* Mô tả chi tiết Use Case: Xem, phân công, phản hồi ticket

Bảng 2. 63: UC31 - Xem, phân công, phản hồi ticket

|  |  |
| --- | --- |
| **ID** | UC31 |
| **Tên Use Case** | Xem danh sách, phân công và phản hồi ticket hỗ trợ |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên xem toàn bộ ticket hỗ trợ, phân công ticket cho nhân viên hỗ trợ cụ thể và phản hồi trực tiếp đến người dùng. |
| **Điều kiện tiên quyết** | - Hệ thống hoạt động bình thường. - Kết nối đến cơ sở dữ liệu support\_tickets, users khả dụng. - Quản trị viên đã đăng nhập với quyền Admin. - Có ít nhất một ticket hỗ trợ trong hệ thống. |
| **Điều kiện thực hiện** | - Quản trị viên có quyền quản lý ticket hỗ trợ. - Truy cập đúng trang quản lý hỗ trợ. |
| **Luồng sự kiện chính** | B1: Quản trị viên đăng nhập và truy cập trang "Quản lý Ticket hỗ trợ".  B2: Hệ thống hiển thị danh sách tất cả ticket với các bộ lọc:  - Trạng thái (Mới, Đang xử lý, Đã trả lời, v.v.)  - Mức độ ưu tiên  - Người phụ trách  - Khoảng thời gian  B3: Quản trị viên xem danh sách, chọn một ticket để xem chi tiết.  B4: Phân công ticket:  - Quản trị viên chọn "Phân công"  - Chọn nhân viên hỗ trợ từ danh sách  - Hệ thống cập nhật assigned\_to và gửi thông báo cho nhân viên  B5: Phản hồi ticket:  - Quản trị viên nhập phản hồi vào khung chat/comment  - Có thể đính kèm file, hình ảnh  - Nhấn "Gửi phản hồi"  - Hệ thống lưu phản hồi và gửi email thông báo cho người dùng  B6: Ticket được cập nhật trạng thái tự động (ví dụ: từ "Mới" thành "Đang xử lý").  B7: Hệ thống lưu lịch sử phản hồi và phân công. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 64: Luồng sự kiện phụ UC31

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Không có ticket nào trong hệ thống. | Hiển thị thông báo "Chưa có ticket hỗ trợ nào." |
| **A2** | Nhân viên được phân công không tồn tại hoặc không có quyền. | "Nhân viên không hợp lệ hoặc không có quyền xử lý ticket." |
| **A3** | Lỗi khi gửi phản hồi hoặc email. | "Gửi phản hồi thất bại, nhưng đã lưu nội dung." |
| **A4** | Ticket đã bị đóng. | "Không thể phản hồi ticket đã đóng." |

* Mô tả chi tiết Use Case: UC32 - Cập nhật trạng tháit

Bảng 2. 65: UC32 - Cập nhật trạng thái

|  |  |
| --- | --- |
| **ID** | UC32 |
| **Tên Use Case** | Cập nhật trạng thái ticket hỗ trợ |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên thay đổi trạng thái của ticket hỗ trợ để phản ánh tiến độ xử lý (ví dụ: từ "Mới" sang "Đang xử lý", "Đang chờ phản hồi", v.v.). |
| **Điều kiện tiên quyết** | - Ticket hỗ trợ tồn tại và chưa bị đóng/lưu trữ. - Quản trị viên có quyền thay đổi trạng thái ticket. |
| **Sự kiện kích hoạt** | Quản trị viên chọn thay đổi trạng thái từ dropdown trong trang chi tiết ticket. |
| **Luồng sự kiện chính** | B1: Trong trang chi tiết ticket, quản trị viên chọn trạng thái mới từ dropdown (ví dụ: "Đang chờ phản hồi từ khách hàng").  B2: Hệ thống hiển thị popup xác nhận (tùy chọn) và yêu cầu nhập ghi chú (tùy chọn).  B3: Quản trị viên xác nhận thay đổi.  B4: Hệ thống nhận POST với action='UPDATE\_TICKET\_STATUS'.  B5: Hệ thống kiểm tra:  - Ticket tồn tại  - Trạng thái mới hợp lệ  - Quyền của admin  B6: Hệ thống bắt đầu giao dịch begin\_transaction().  B7: Cập nhật trường status trong bảng support\_tickets.  B8: Ghi log lịch sử thay đổi trạng thái vào bảng ticket\_history.  B9: commit() giao dịch.  B10: Gửi email/thông báo cho người dùng về thay đổi trạng thái (nếu cần).  B11: Phản hồi thành công và cập nhật giao diện. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

Bảng 2. 66: Luồng sự kiện phụ UC31

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Ticket không tồn tại. | "Ticket không tồn tại." |
| **A2** | Trạng thái mới không hợp lệ. | "Trạng thái không hợp lệ." |
| **A3** | Ticket đã đóng, không thể thay đổi trạng thái. | "Không thể thay đổi trạng thái ticket đã đóng." |
| **A4** | Lỗi hệ thống khi cập nhật. | "Cập nhật trạng thái thất bại." |

* Mô tả chi tiết Use Case: Đóng và lưu trữ ticket

Bảng 2. 67: UC33 - Đóng và lưu trữ ticket

|  |  |
| --- | --- |
| **ID** | UC33 |
| **Tên Use Case** | Đóng ticket và lưu trữ |
| **Tác nhân chính** | Quản trị viên (Administrator) |
| **Mục tiêu** | Cho phép quản trị viên đóng ticket khi vấn đề đã được giải quyết và lưu trữ ticket để tham khảo sau này, giải phóng danh sách ticket đang mở. |
| **Điều kiện tiên quyết** | - Ticket tồn tại và đang ở trạng thái có thể đóng (ví dụ: "Đã giải quyết"). - Quản trị viên có quyền đóng ticket. |
| **Sự kiện kích hoạt** | Quản trị viên nhấn nút "Đóng ticket" trong trang chi tiết ticket. |
| **Luồng sự kiện chính** | B1: Trong trang chi tiết ticket, quản trị viên nhấn nút "Đóng ticket".  B2: Hệ thống hiển thị form đóng ticket:  - Yêu cầu chọn lý do đóng (Đã giải quyết, Không phản hồi, Trùng lặp, v.v.)  - Nhập ghi chú cuối cùng (tùy chọn)  - Đánh giá mức độ hài lòng (nếu có)  B3: Quản trị viên điền thông tin và xác nhận đóng ticket.  B4: Hệ thống nhận POST với action='CLOSE\_TICKET'.  B5: Hệ thống kiểm tra:  - Ticket tồn tại, chưa đóng  - Đã có phản hồi từ nhân viên hỗ trợ (tùy chọn)  B6: Hệ thống bắt đầu giao dịch begin\_transaction().  B7: Cập nhật ticket:  - status = 'closed'  - closed\_at = thời gian hiện tại  - closed\_by = admin ID  - closure\_reason = lý do đóng  B8: Di chuyển ticket sang bảng lưu trữ (hoặc đánh dấu archived = 1).  B9: commit() giao dịch.  B10: Gửi email thông báo đóng ticket và khảo sát hài lòng cho người dùng.  B11: Phản hồi thành công và chuyển hướng về danh sách ticket. |

* Luồng sự kiện phụ (Alternative / Exception Flow)

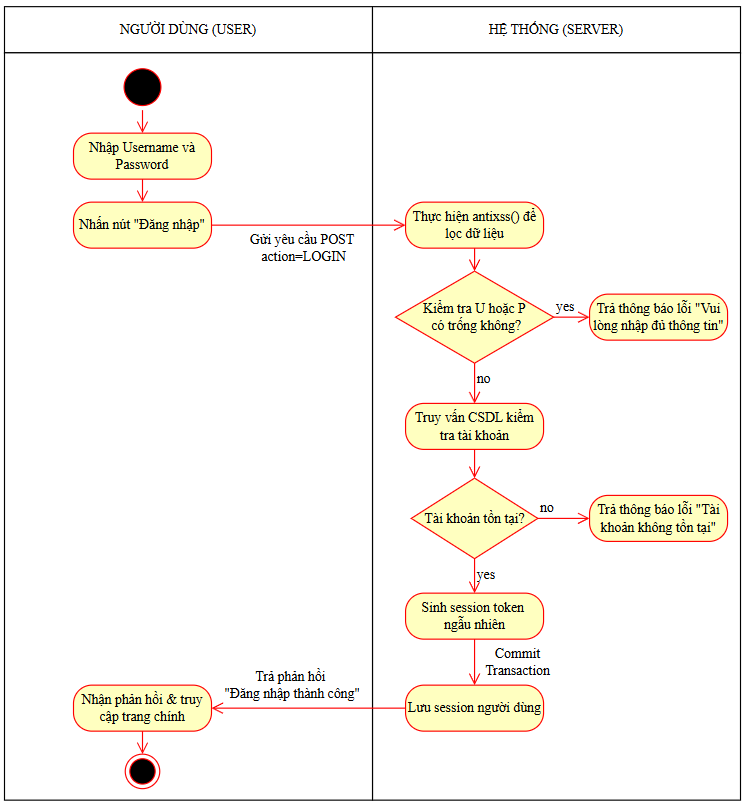
Bảng 2. 68: Luồng sự kiện phụ UC33

|  |  |  |
| --- | --- | --- |
| **Mã lỗi** | **Điều kiện** | **Phản hồi hệ thống** |
| **A1** | Ticket không tồn tại. | "Ticket không tồn tại." |
| **A2** | Ticket đã đóng trước đó. | "Ticket đã được đóng trước đó." |
| **A3** | Chưa có phản hồi nào từ nhân viên hỗ trợ. | "Cần có ít nhất một phản hồi từ nhân viên trước khi đóng ticket." |
| **A4** | Lỗi hệ thống khi đóng/lưu trữ. | "Đóng ticket thất bại." |
| **A5** | Người dùng hủy thao tác. | "Đã hủy thao tác đóng ticket." |

### 2.2.2 Biểu đồ hoạt động

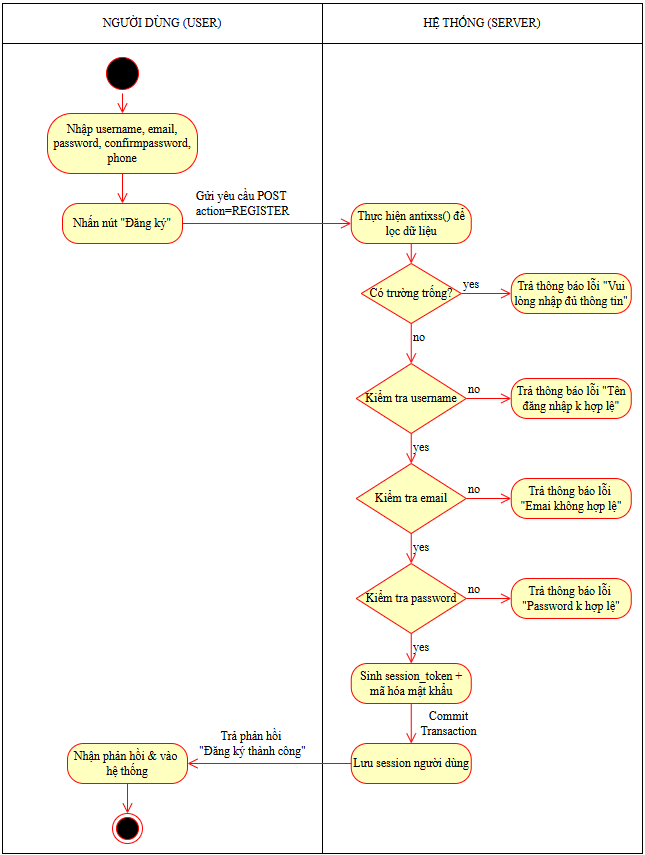
#### a, Biểu đồ hoạt động đăng nhập

Bảng 2. 69: Chức năng "đăng nhập"



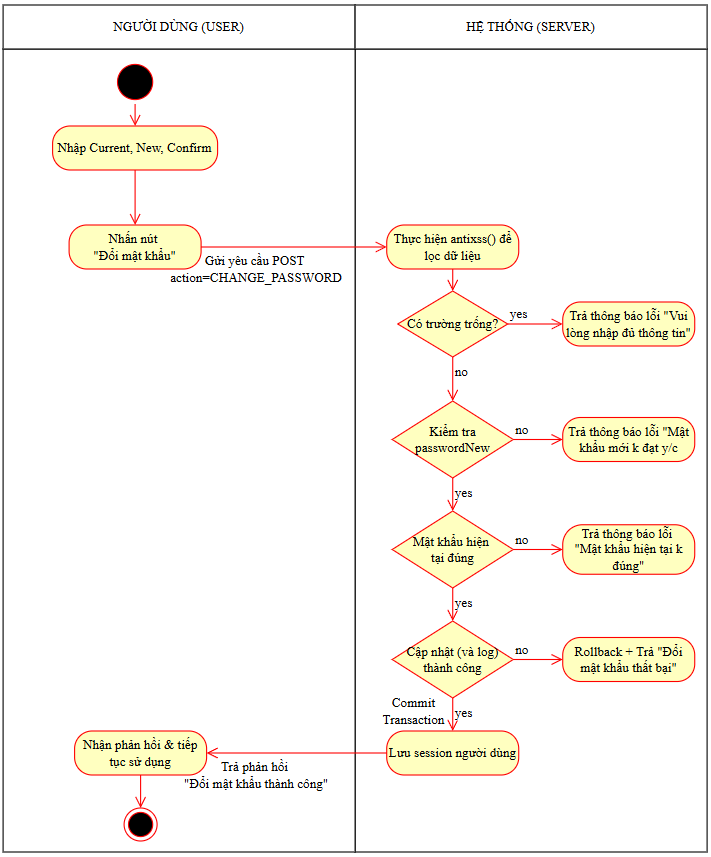
#### b, Biểu đồ hoạt động đăng ký

Bảng 2. 70: Chức năng "Đăng ký"



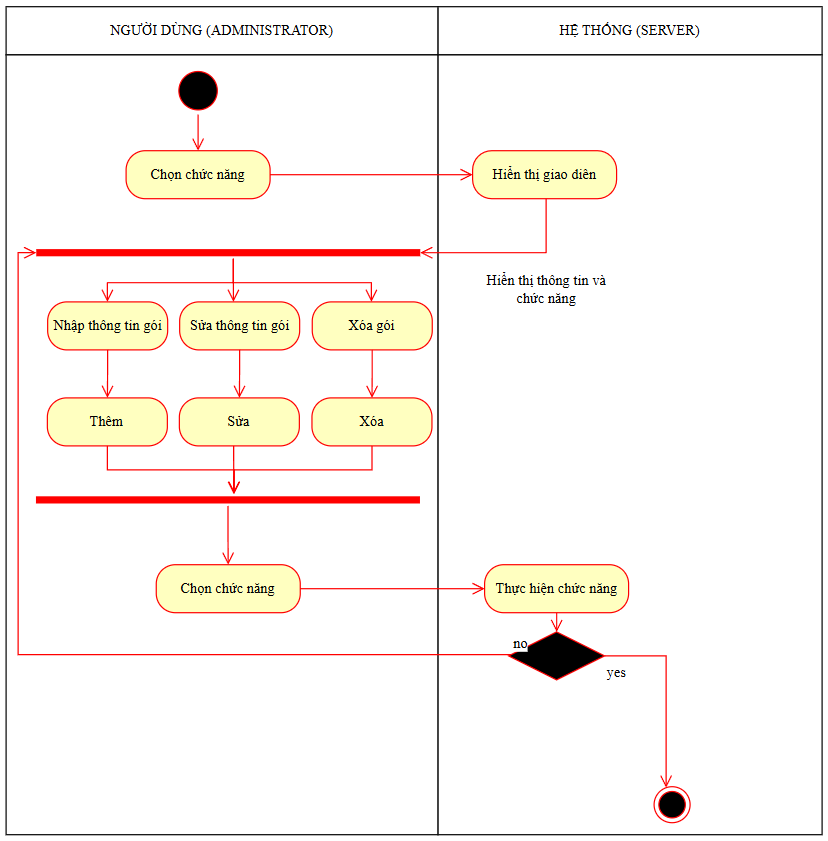
#### c, Biểu đồ hoạt động đổi mật khẩu

Bảng 2. 71: Chức năng "Đổi mật khẩu"



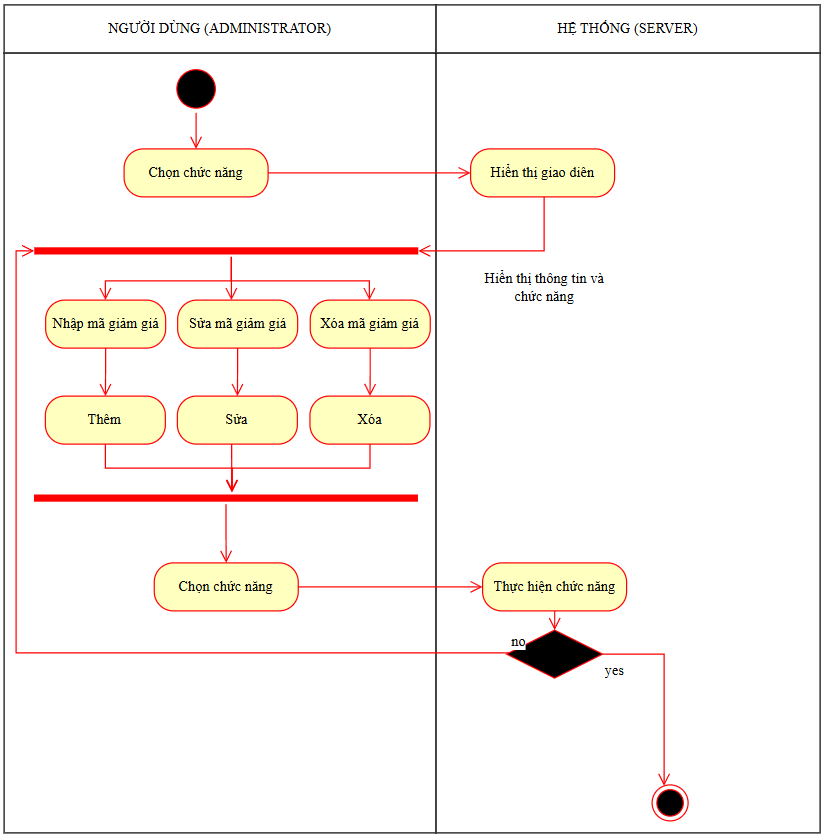
#### d, Biểu đồ hoạt động quản lý hosting

Bảng 2. 72: Biểu đồ hoạt động quản lý hosting



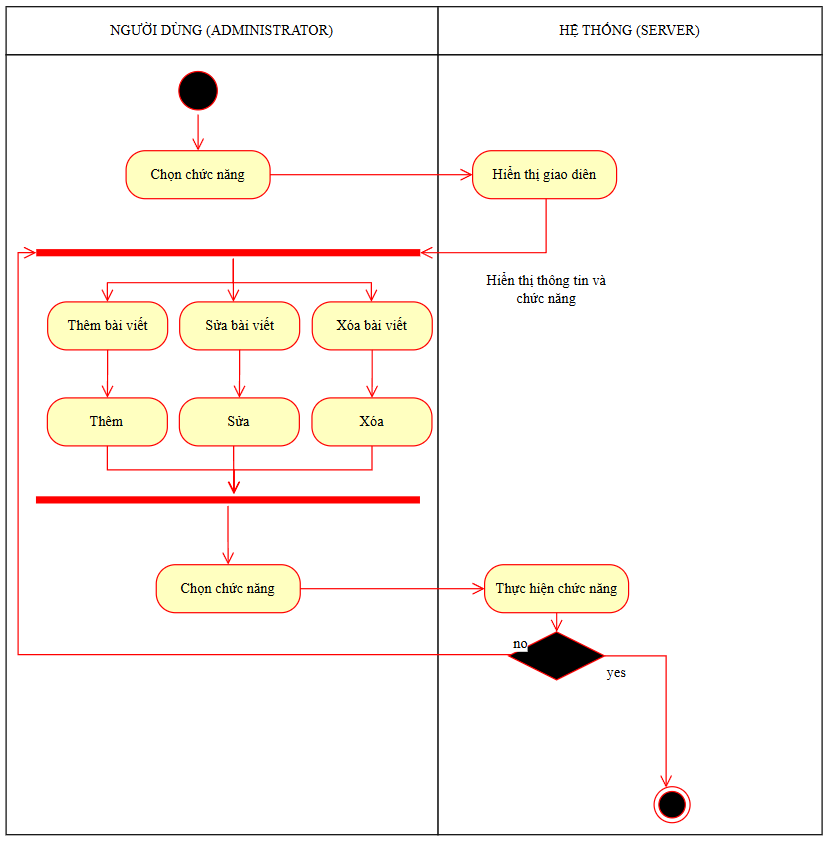
#### e, Biểu đồ hoạt động quản lý mã giảm giá

Bảng 2. 73: Biểu đồ hoạt động quản lý mã giảm giá



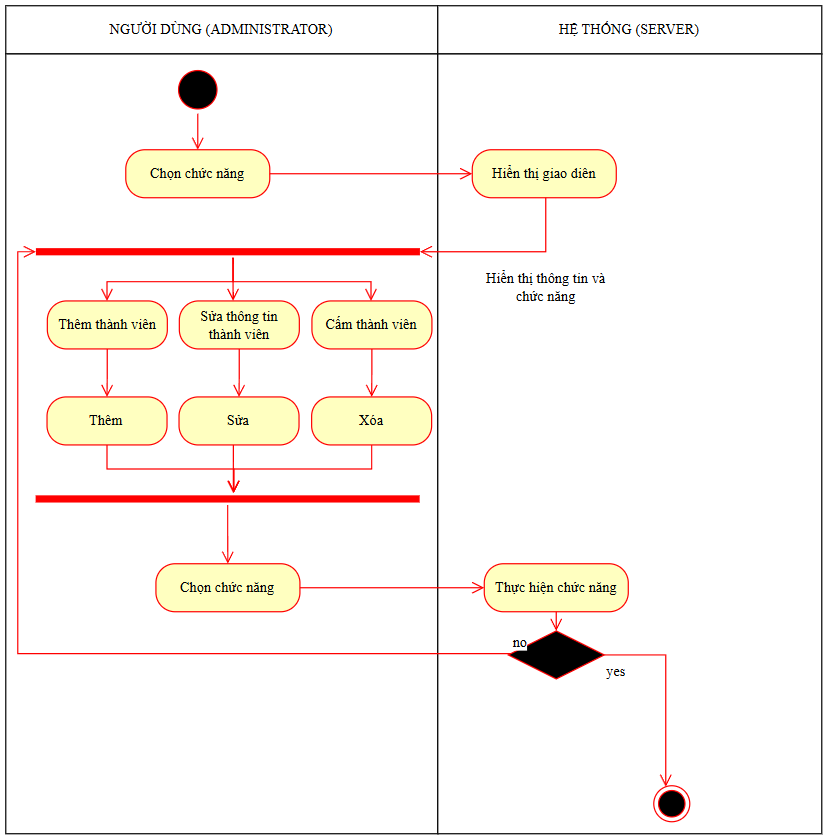
#### f, Biểu đồ hoạt động quản lý bài viết

Bảng 2. 74: Biểu đồ hoạt động quản lý bài viết



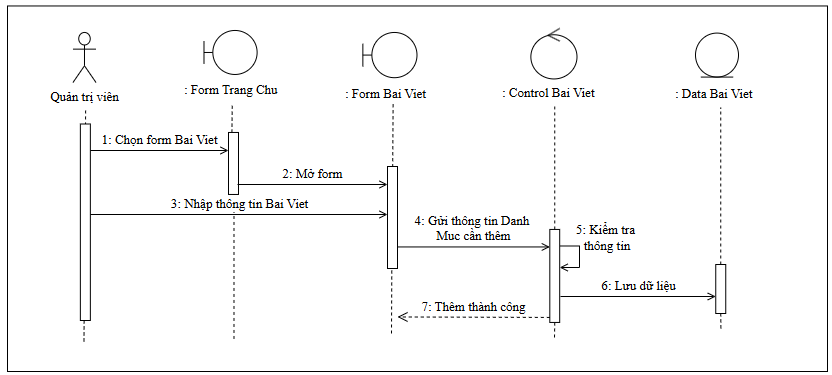
#### g, Biểu đồ hoạt động quản lý thành viên

Bảng 2. 75: Biểu đồ hoạt động quản lý thành viên

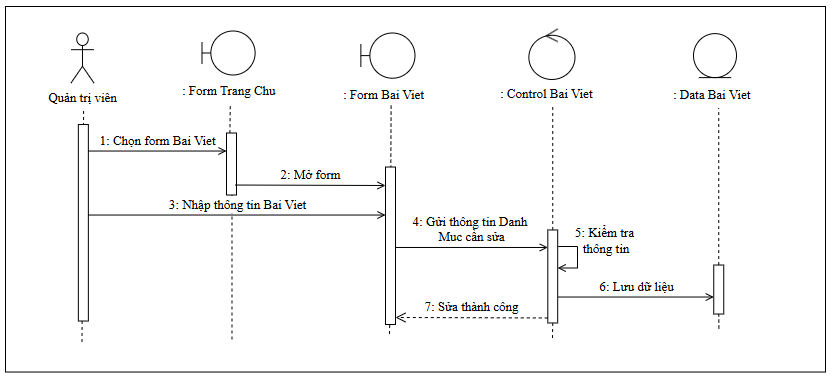


### 2.2.3 Biểu đồ trạng thái

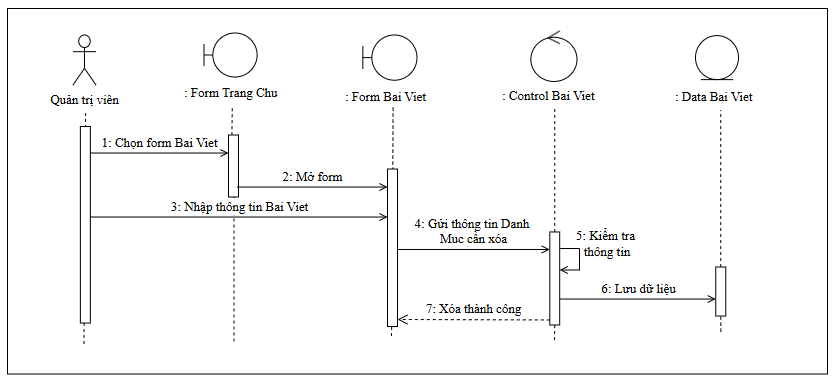
#### a, Biểu đồ trạng thái của quản lý bài viết



Hình 2. 15: Sơ đồ trạng thái chức năng Thêm bài viết

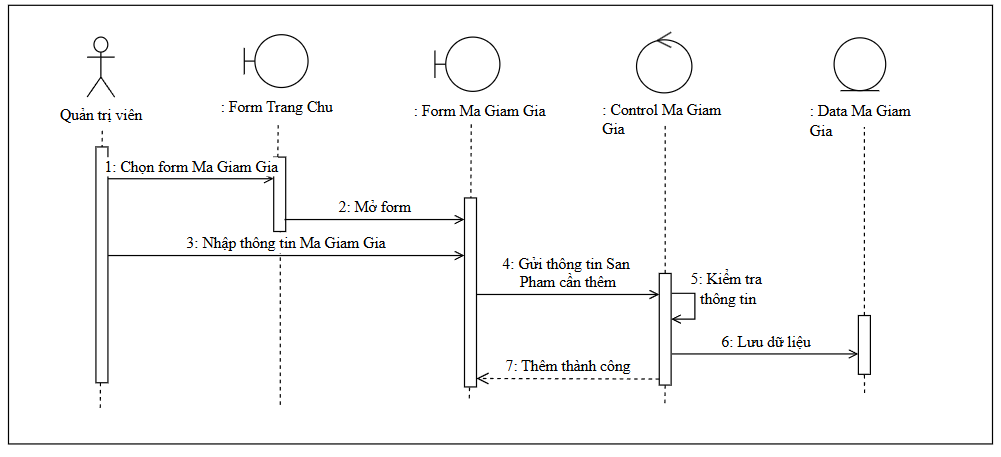


Hình 2. 16: Sơ đồ trạng thái chức năng Sửa bài viết

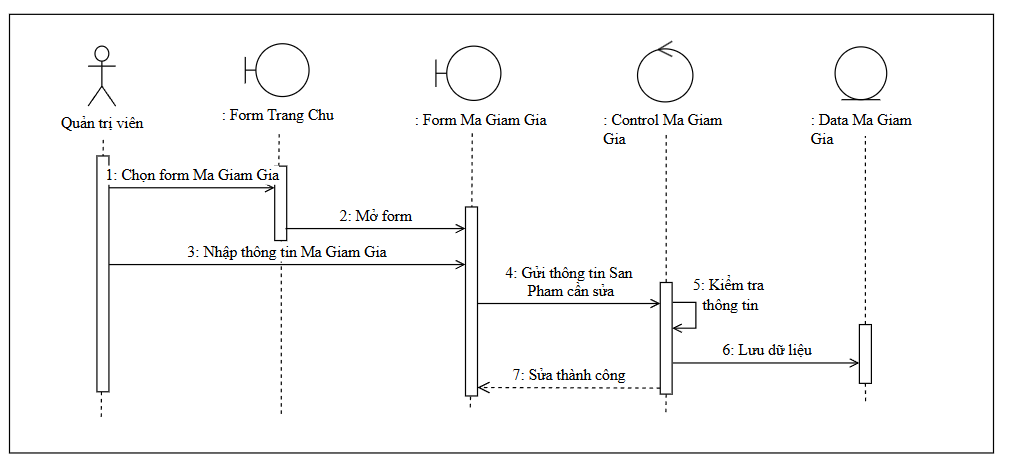


Hình 2. 17: Sơ đồ trạng thái chức năng Xóa bài viết

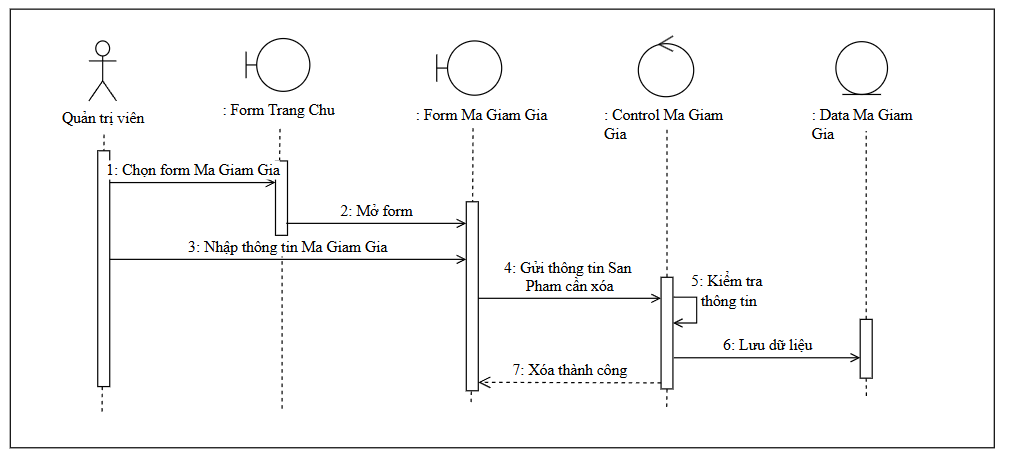
#### b, Biểu đồ trạng thái của quản lý mã giảm giá



Hình 2. 18: Sơ đồ trạng thái chức năng Thêm mã giảm giá

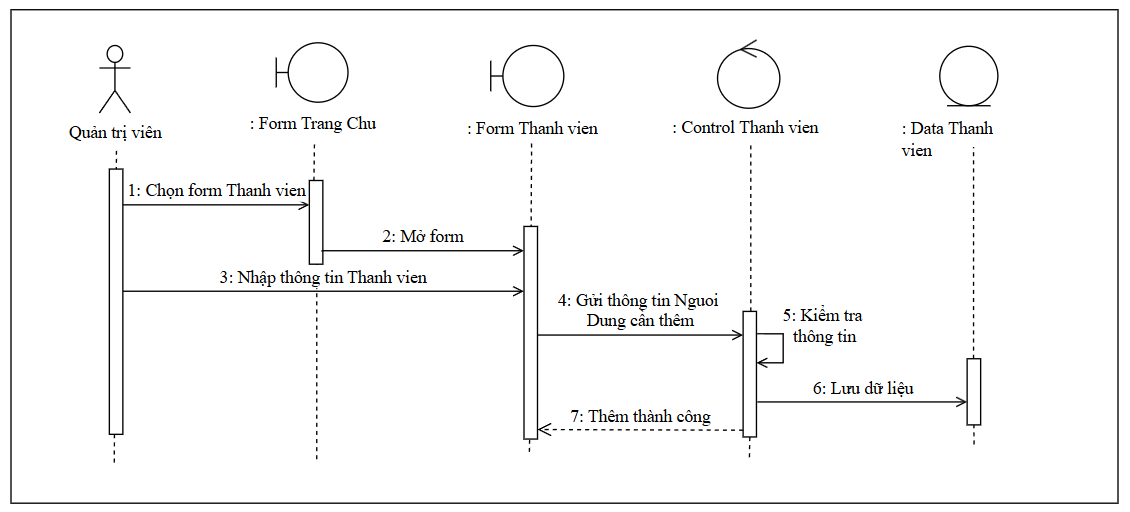


Hình 2. 19: Sơ đồ trạng thái chức năng Sửa mã giảm giá

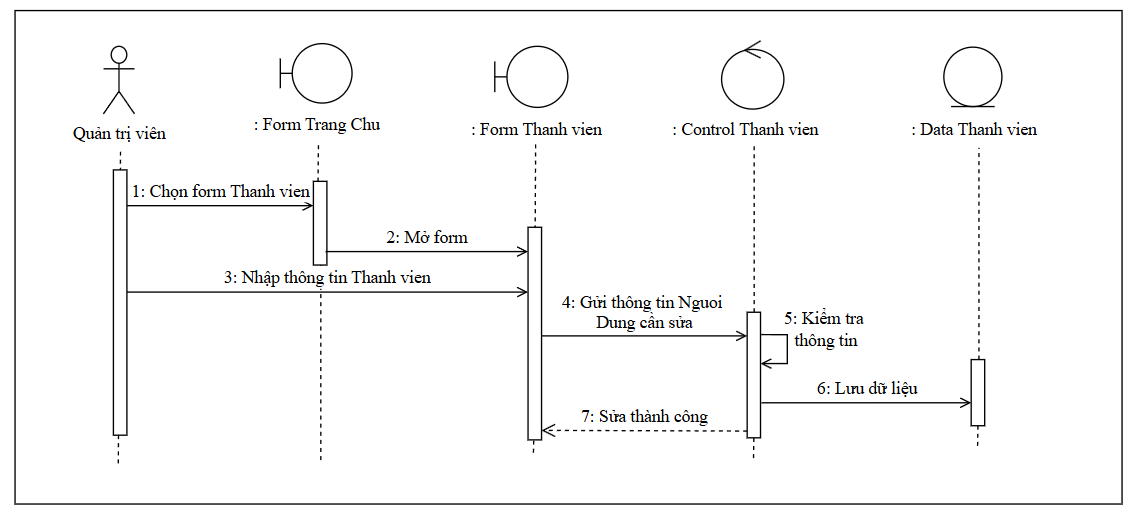


Hình 2. 20: Sơ đồ trạng thái chức năng Xóa mã giảm giá

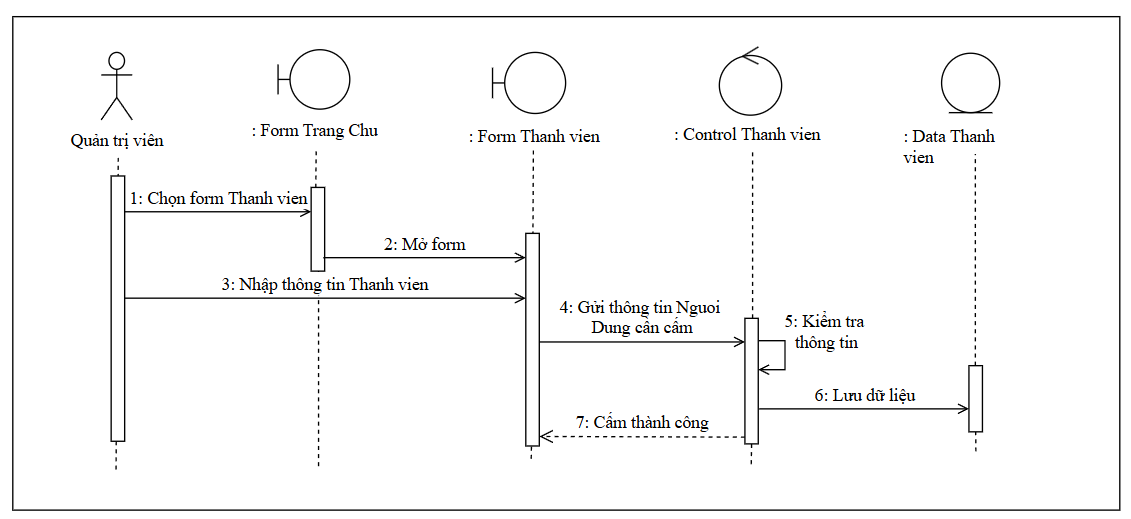
#### c, Biểu đồ trạng thái của quản lý thành viên



Hình 2. 21: Sơ đồ trạng thái chức năng Thêm thành viên

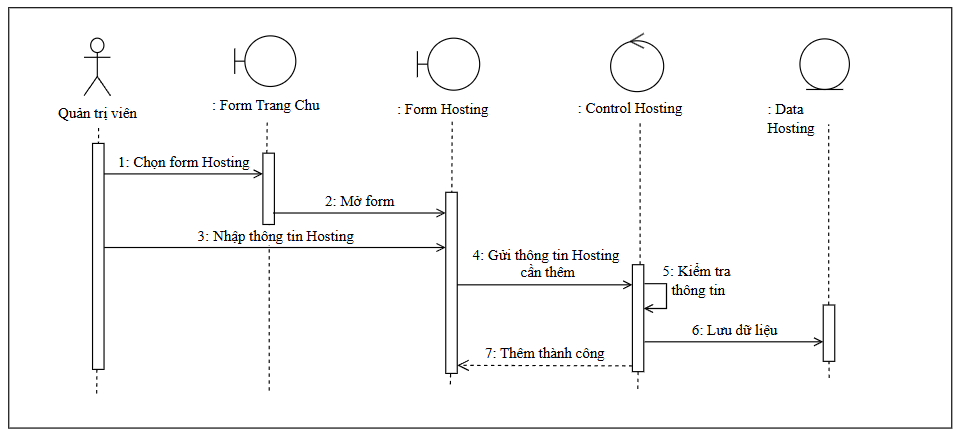


Hình 2. 22: Sơ đồ trạng thái chức năng Sửa thành viên

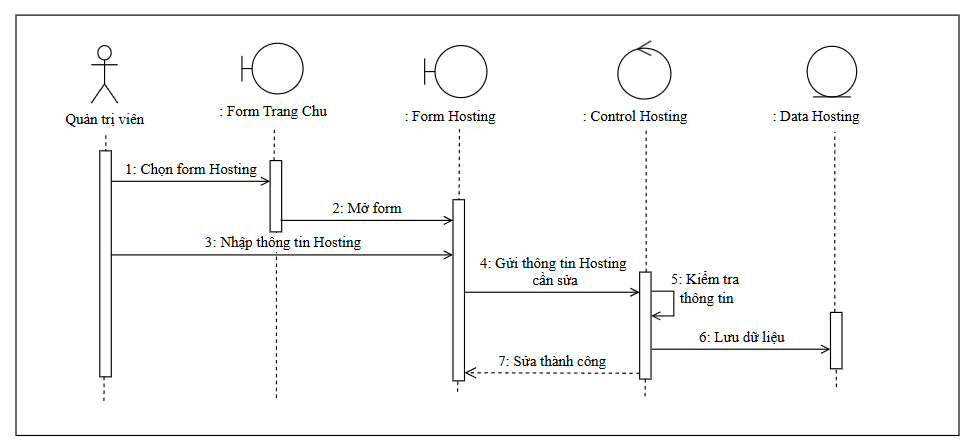


Hình 2. 23: Sơ đồ trạng thái chức năng Xóa thành viên

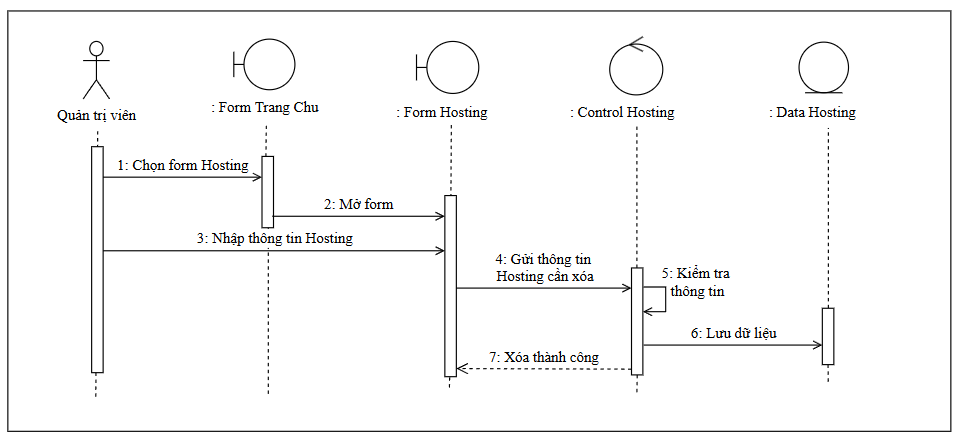
d, Biểu đồ trạng thái của quản lý hosting



Hình 2. 24: Sơ đồ trạng thái chức năng Thêm hosting



Hình 2. 25: Sơ đồ trạng thái chức năng Sửa hosting



Hình 2. 26: Sơ đồ trạng thái chức năng Xóa hosting

### 2.2.4 Cấu trúc các bảng Database

Bảng 2. 76: Cấu trúc bảng discount (Mã giảm giá)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Cho phép NULL** | **Khóa chính** | **Mô tả** |
| id | INT | - | NO | ✓ | ID tự động tăng |
| code | VARCHAR | 100 | YES | - | Mã giảm giá |
| amount | INT | - | NO | - | Số lượng |
| used | INT | - | NO | - | Đã sử dụng |
| discount | INT | - | NO | - | Giá trị giảm |
| createdate | VARCHAR | 100 | YES | - | Ngày tạo |
| enddate | VARCHAR | 100 | YES | - | Ngày hết hạn |
| min | INT | - | NO | - | Giá trị tối thiểu |
| max | INT | - | NO | - | Giá trị tối đa |
| type | VARCHAR | 100 | YES | - | Loại (fixed/percent) |
| status | VARCHAR | 50 | NO | - | Trạng thái |

Bảng 2. 77: Cấu trúc bảng history\_buy\_hosting (Lịch sử mua hosting)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Cho phép NULL** | **Khóa chính** | **Mô tả** |
| id | INT | - | NO | ✓ | ID tự động tăng |
| username | VARCHAR | 100 | YES | - | Tên đăng nhập |
| domain | VARCHAR | 100 | YES | - | Tên miền |
| pk\_host | VARCHAR | 100 | YES | - | Mã gói hosting |
| sv\_host | INT | - | YES | - | ID server hosting |
| money | INT | - | YES | - | Giá |
| total\_money | INT | - | YES | - | Tổng tiền |
| account | TEXT | - | YES | - | Tài khoản (mã hóa) |
| password | TEXT | - | YES | - | Mật khẩu (mã hóa) |
| creatAt | VARCHAR | 100 | YES | - | Ngày tạo |
| endAt | VARCHAR | 100 | YES | - | Ngày hết hạn |
| status | VARCHAR | 100 | YES | - | Trạng thái |
| note | VARCHAR | 100 | YES | - | Ghi chú |
| time | VARCHAR | 100 | YES | - | Thời gian |

Bảng 2. 78: Cấu trúc bảng his\_login (Lịch sử đăng nhập)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Cho phép NULL** | **Khóa chính** | **Mô tả** |
| id | INT | - | NO | ✓ | ID tự động tăng |
| username | VARCHAR | 100 | YES | - | Tên đăng nhập |
| title | VARCHAR | 100 | YES | - | Tiêu đề |
| browser | VARCHAR | 100 | YES | - | Trình duyệt |
| device | VARCHAR | 100 | YES | - | Thiết bị |
| ip | VARCHAR | 100 | YES | - | Địa chỉ IP |
| address | VARCHAR | 100 | YES | - | Địa chỉ |
| time | VARCHAR | 100 | YES | - | Thời gian |

Bảng 2. 79: Cấu trúc bảng package\_hosting (Gói Hosting)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Cho phép NULL** | **Khóa chính** | **Mô tả** |
| id | INT | - | NO | ✓ | ID tự động tăng |
| name\_host | VARCHAR | 100 | YES | - | Tên gói hosting |
| code\_host | VARCHAR | 100 | YES | - | Mã gói hosting |
| server\_host | INT | - | YES | - | ID server hosting |
| money | INT | - | YES | - | Giá |
| disk | INT | - | YES | - | Dung lượng |
| other\_domain | VARCHAR | 100 | YES | - | Domain khác |
| alias\_domain | VARCHAR | 100 | YES | - | Alias domain |
| status | VARCHAR | 100 | YES | - | Trạng thái |

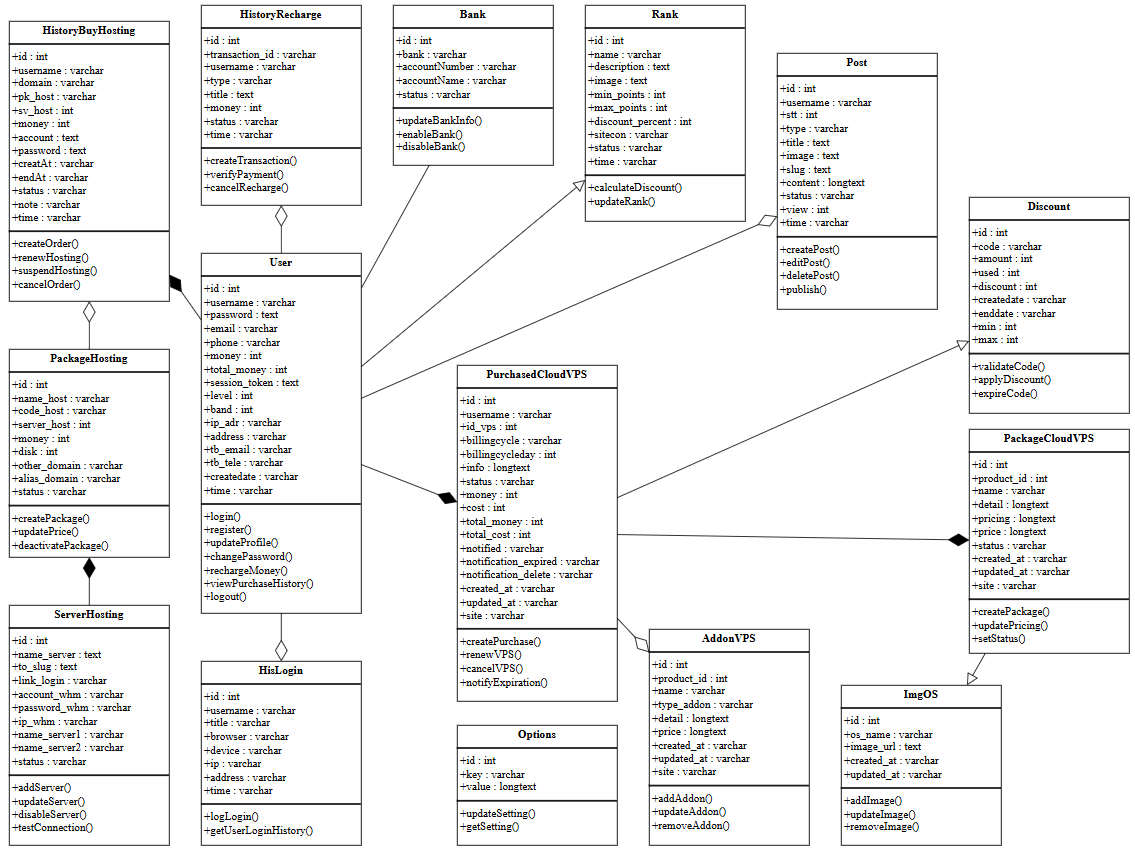
Bảng 2. 80: Cấu trúc bảng posts (Bài viết/Blog)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Cho phép NULL** | **Khóa chính** | **Mô tả** |
| id | INT | - | NO | ✓ | ID tự động tăng |
| username | VARCHAR | 100 | YES | - | Tên đăng nhập |
| stt | INT | - | NO | - | Số thứ tự |
| type | VARCHAR | 100 | YES | - | Loại bài viết |
| title | TEXT | - | YES | - | Tiêu đề |
| image | TEXT | - | YES | - | Hình ảnh |
| slug | TEXT | - | YES | - | Slug URL |
| content | LONGTEXT | - | YES | - | Nội dung |
| status | VARCHAR | 100 | NO | - | Trạng thái |

Bảng 2. 81: Cấu trúc bảng users (Người dùng)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Kích thước** | **Cho phép NULL** | **Khóa chính** | **Mô tả** |
| id | INT | - | NO | ✓ | ID tự động tăng |
| fullname | TEXT | - | YES | - | Họ tên |
| username | VARCHAR | 100 | NO | - | Tên đăng nhập |
| password | TEXT | - | NO | - | Mật khẩu (mã hóa) |
| email | VARCHAR | 100 | NO | - | Email |
| phone | VARCHAR | 100 | NO | - | Số điện thoại |
| money | INT | - | NO | - | Số dư |
| total\_money | INT | - | NO | - | Tổng tiền đã nạp |
| session\_token | TEXT | - | NO | - | Token phiên |
| level | INT | - | NO | - | Cấp độ (0-user, 1-admin) |
| band | INT | - | NO | - | Trạng thái khóa (0/1) |
| ip\_adr | VARCHAR | 100 | YES | - | Địa chỉ IP |
| address | VARCHAR | 100 | YES | - | Địa chỉ |
| tb\_email | VARCHAR | 100 | YES | - | Thông báo email |
| tb\_tele | VARCHAR | 100 | YES | - | Thông báo Telegram |
| id\_tele | VARCHAR | 100 | YES | - | ID Telegram |
| token | TEXT | - | YES | - | Token xác thực |
| veri\_otp | INT | - | YES | - | OTP xác thực |
| veri\_email | VARCHAR | 100 | YES | - | Xác thực email |
| createdate | VARCHAR | 100 | YES | - | Ngày tạo |

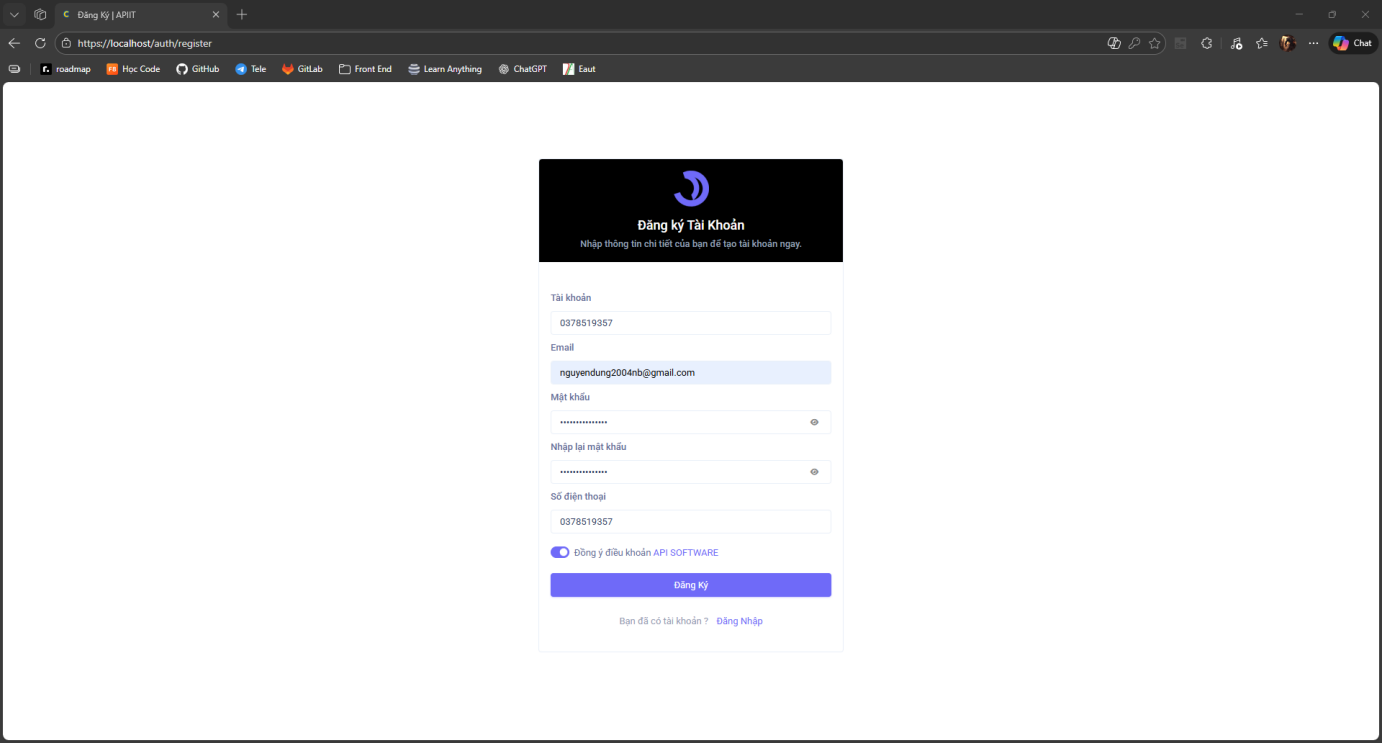
### 2.2.5 Biểu đồ lớp



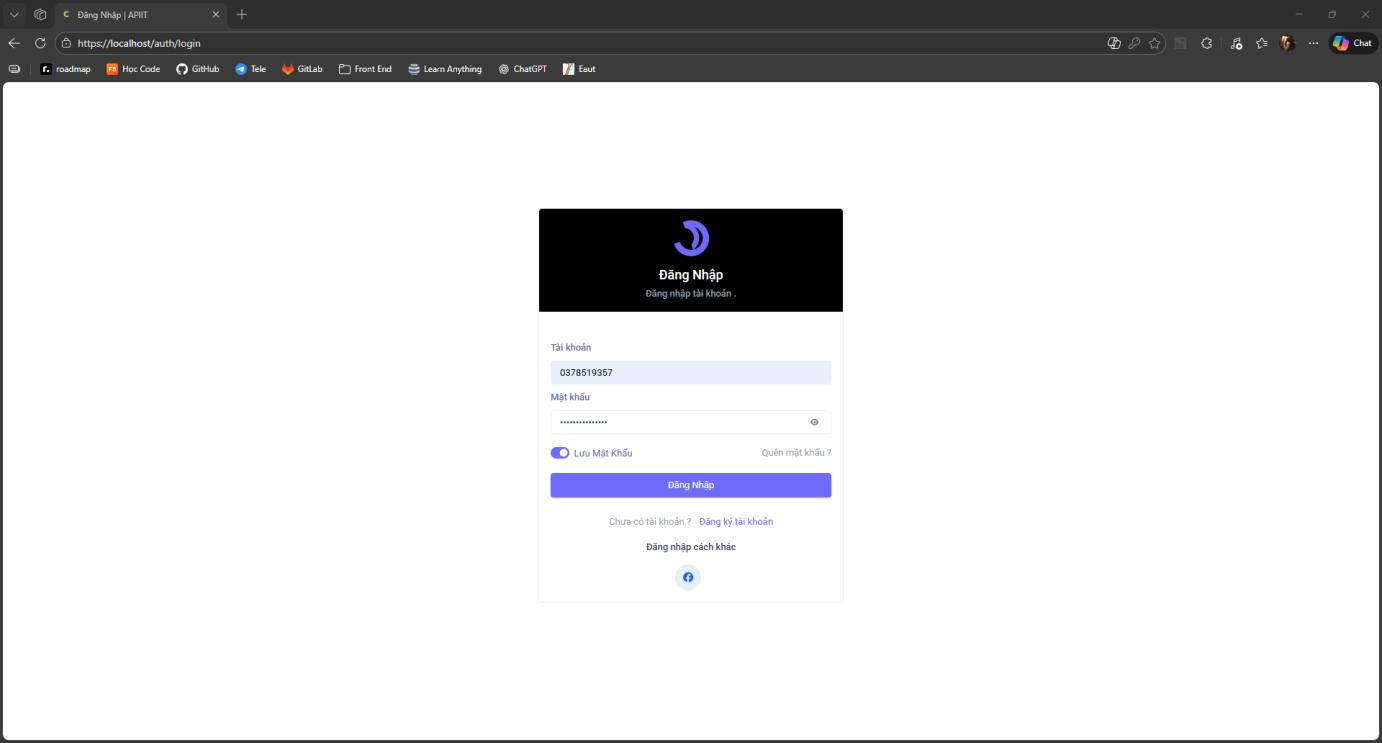
Hình 2. 27: Biểu đồ lớp

# CHƯƠNG III: GIAO DIỆN HỆ THỐNG

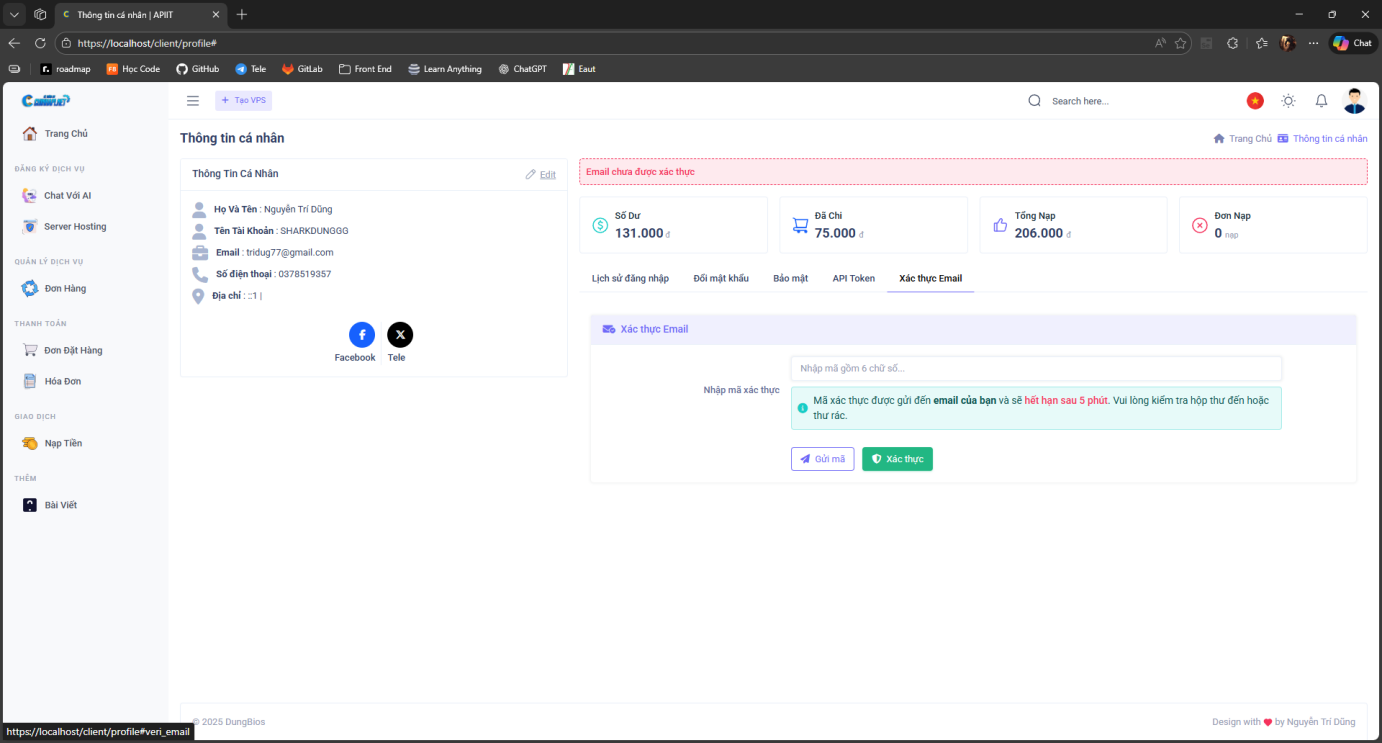
## 3.1 Giao diện người dùng



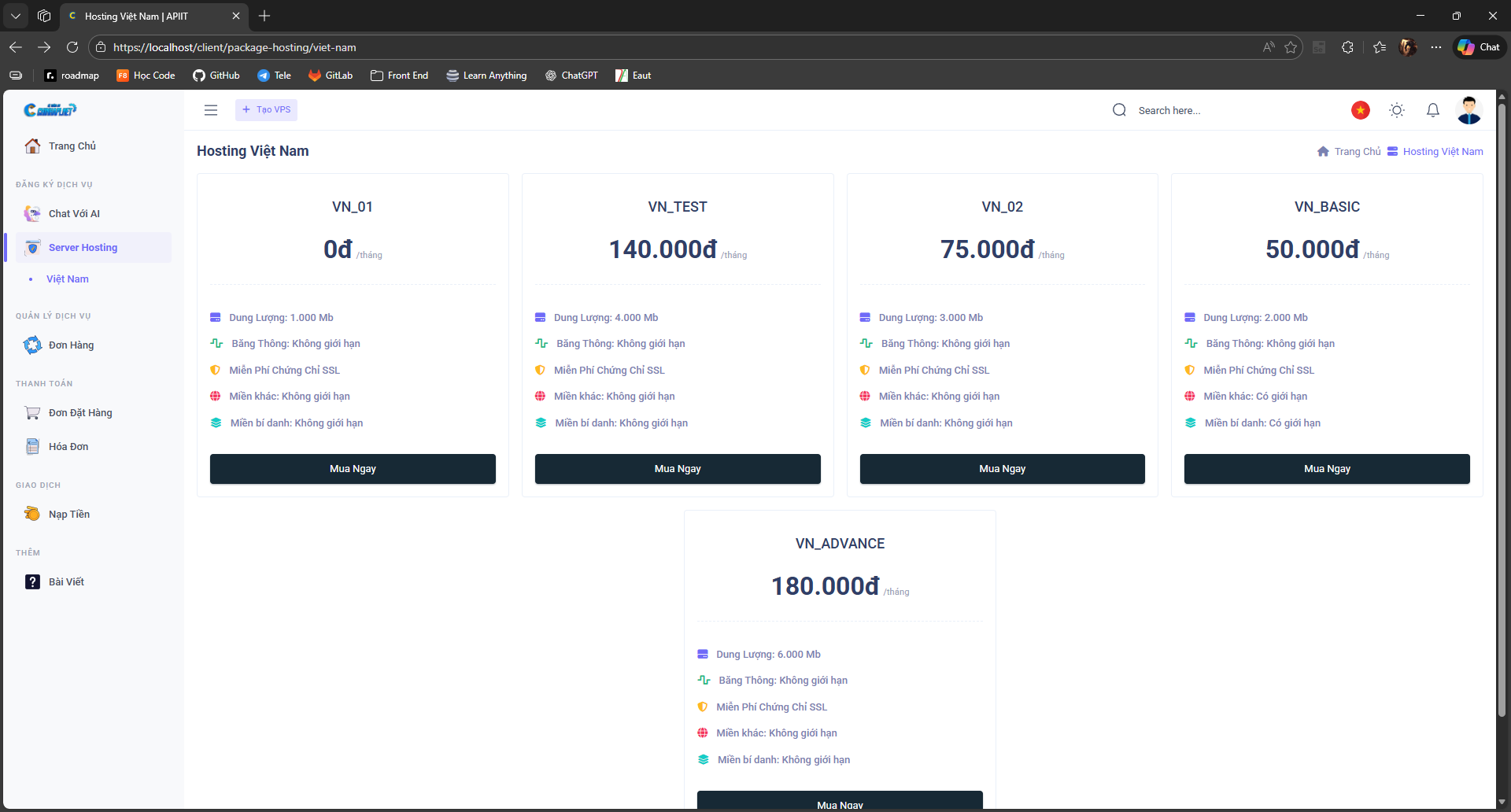
Hình 3. 1: Giao diện đăng ký hệ thống



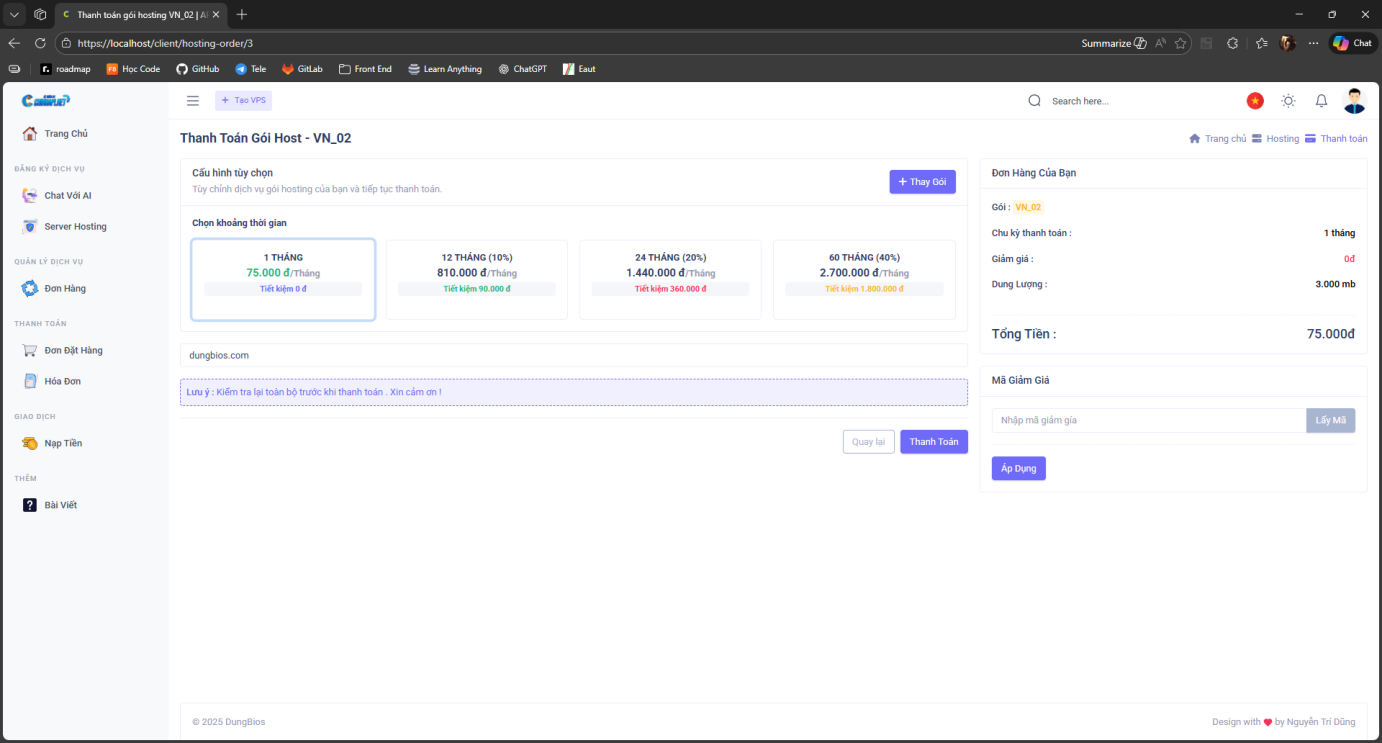
Hình 3. 2: Giao diện đăng nhập hệ thống



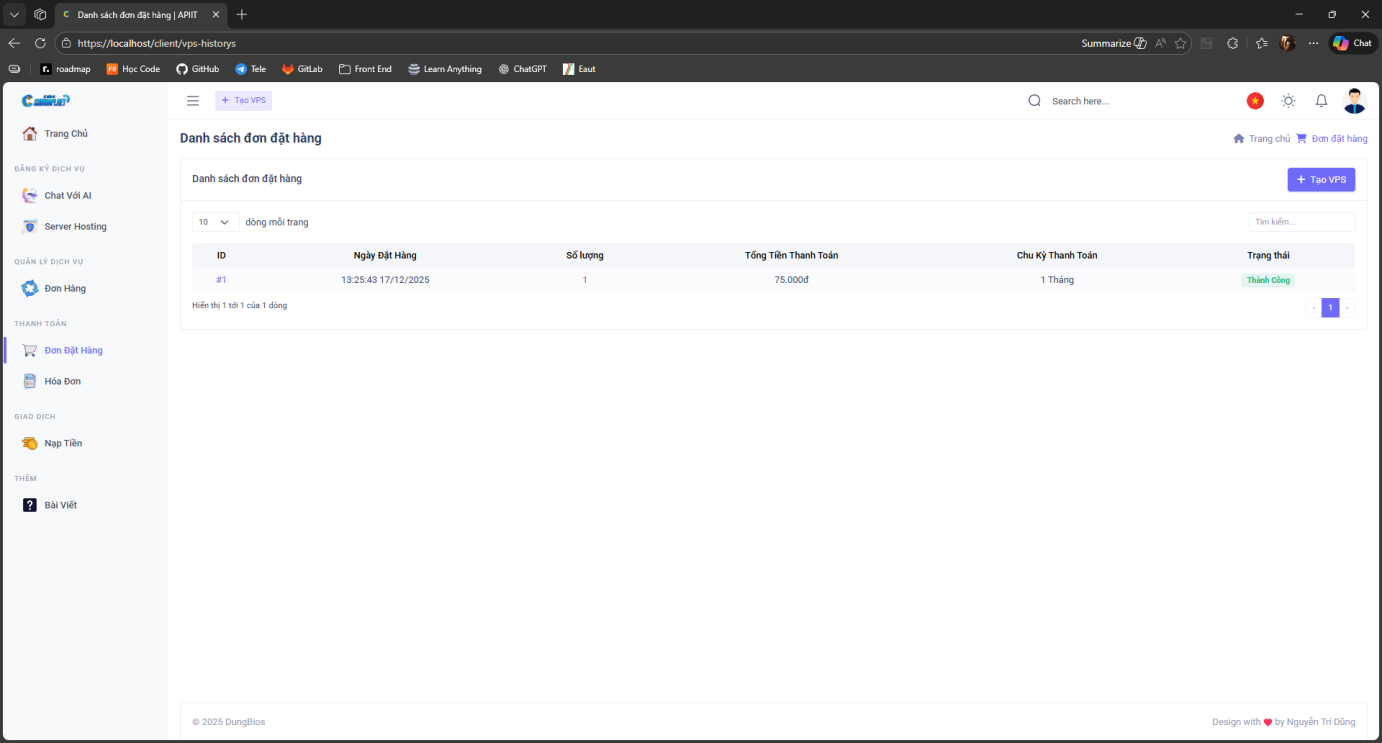
Hình 3. 3: GIao diện thông tin cá nhân



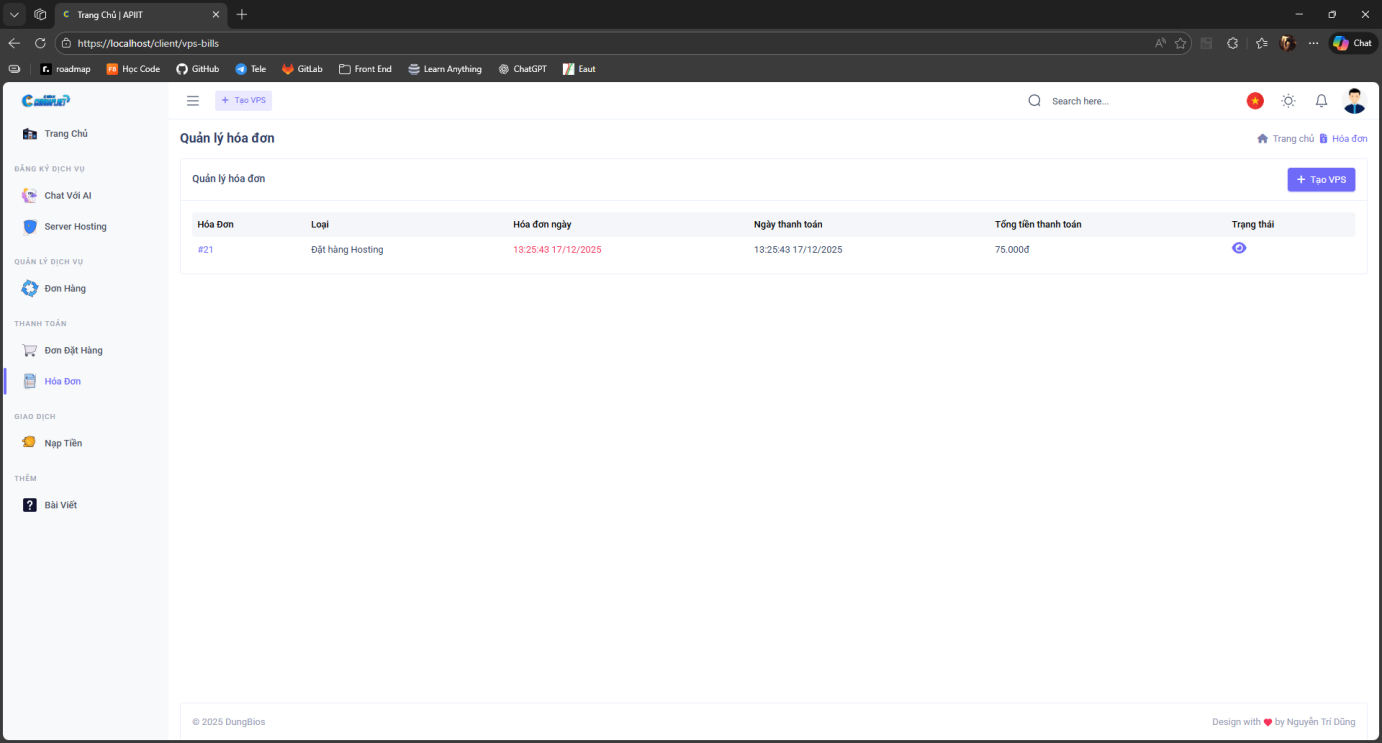
Hình 3. 4: Giao diện bán hosting



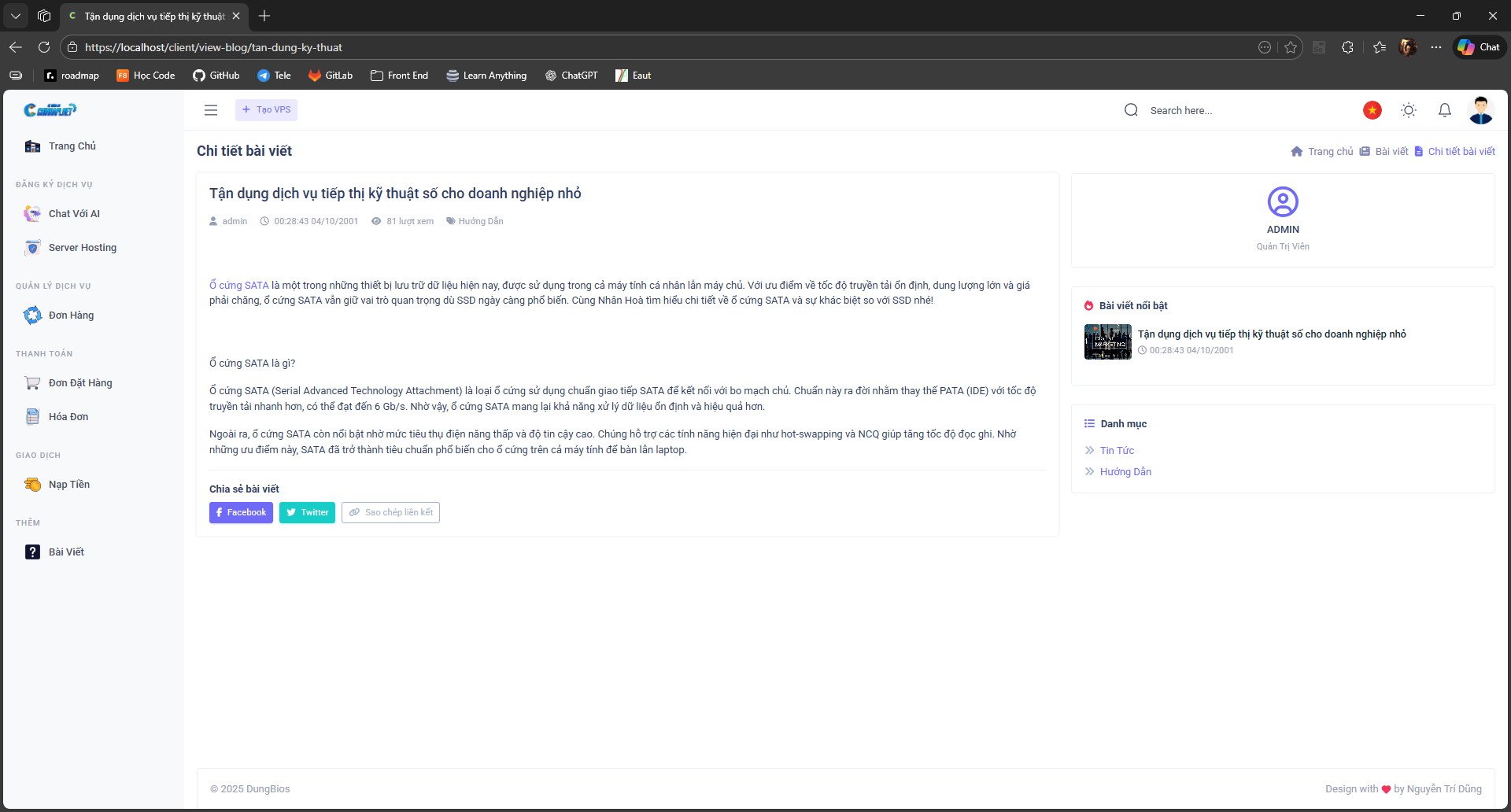
Hình 3. 5: Giao diện thanh toán đơn hàng



Hình 3. 6: Giao diện đơn đặt hàng

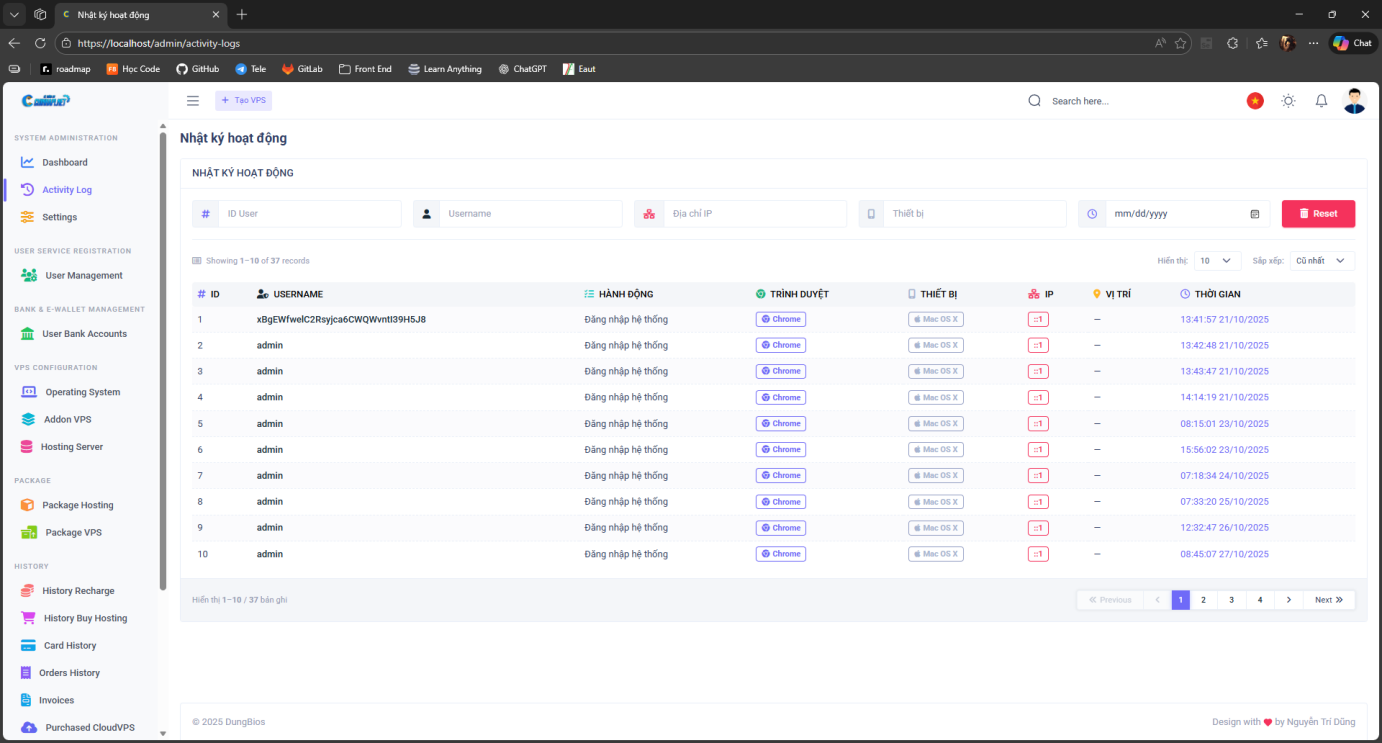


Hình 3. 7: Giao diện hóa đơn

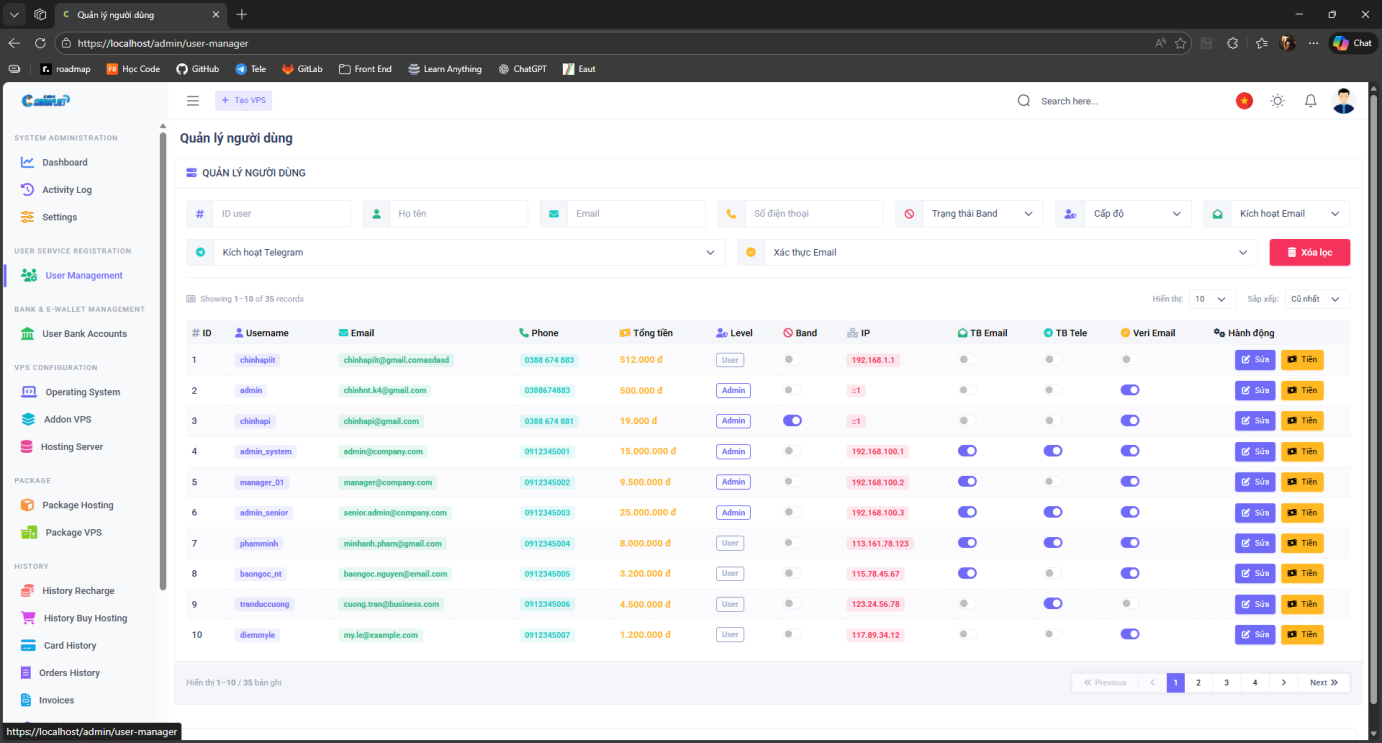


Hình 3. 8: Giao diện 1 bài viết cụ thể

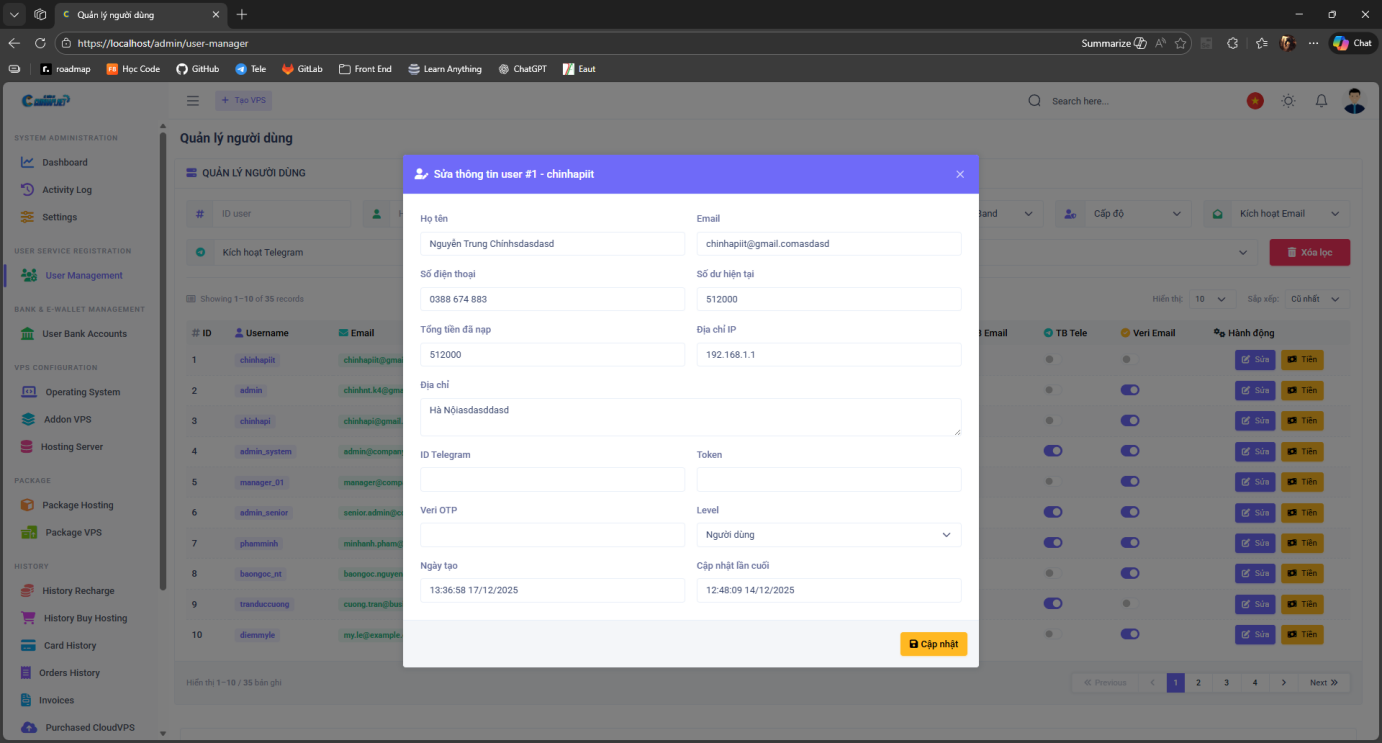
## 3.2 Giao diện quản trị viên



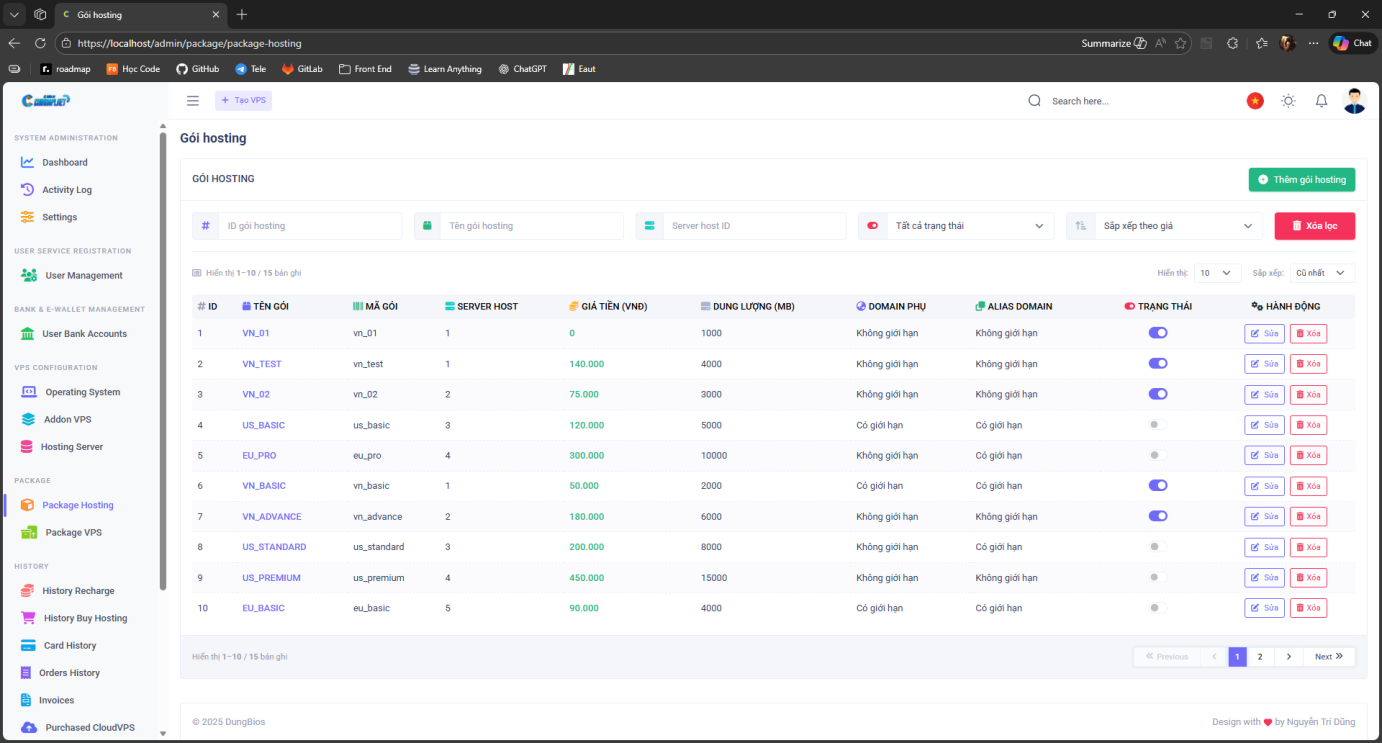
Hình 3. 9: Lịch sử hoạt động



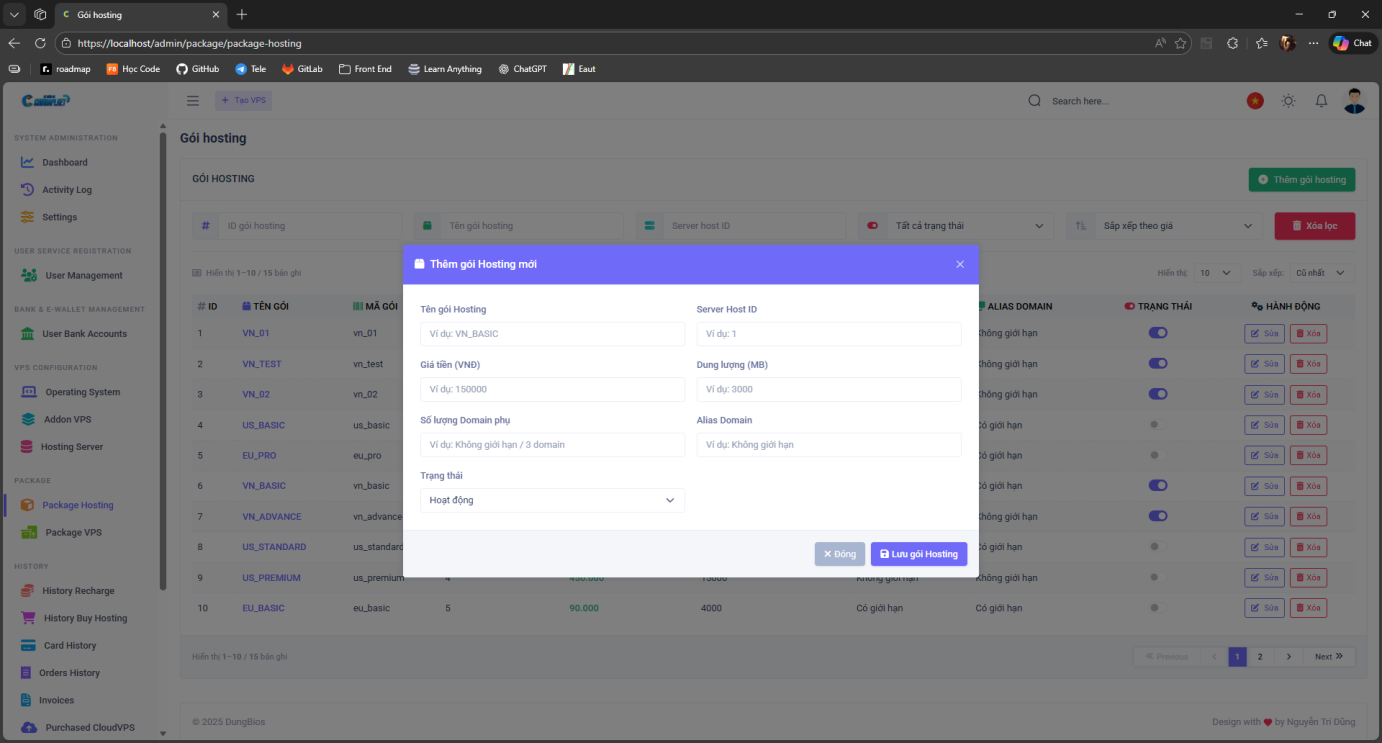
Hình 3. 10: Giao diện quản lý thành viên



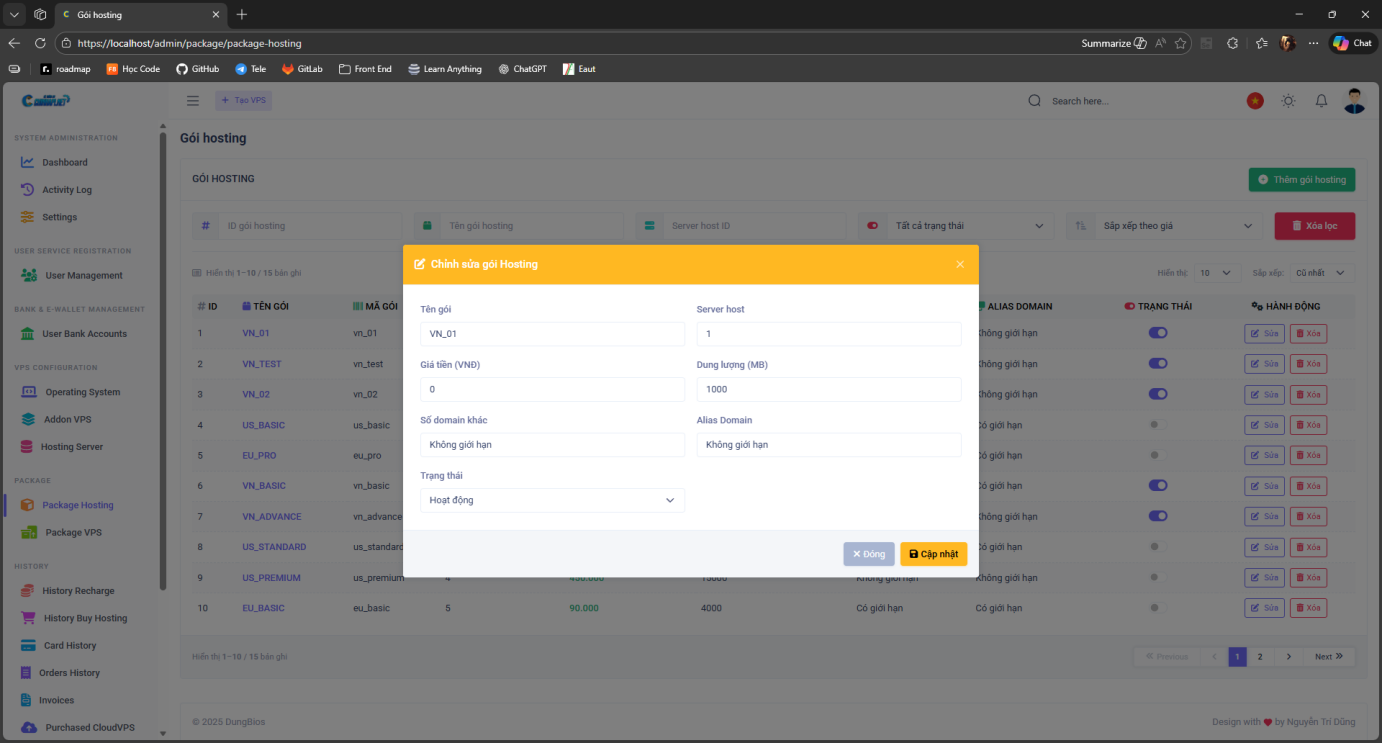
Hình 3. 11: Giao diện câp nhật thông tin thành viên



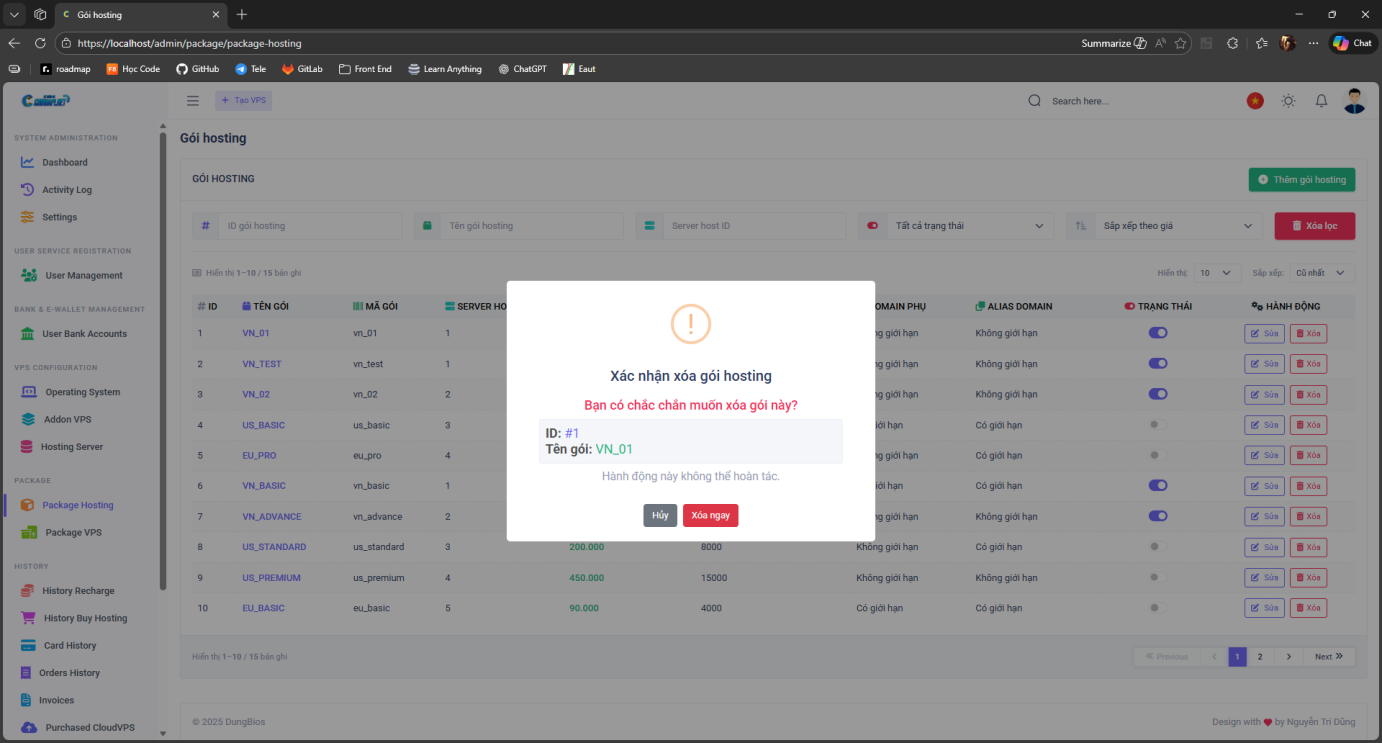
Hình 3. 12: Giao diện quản lý hosting



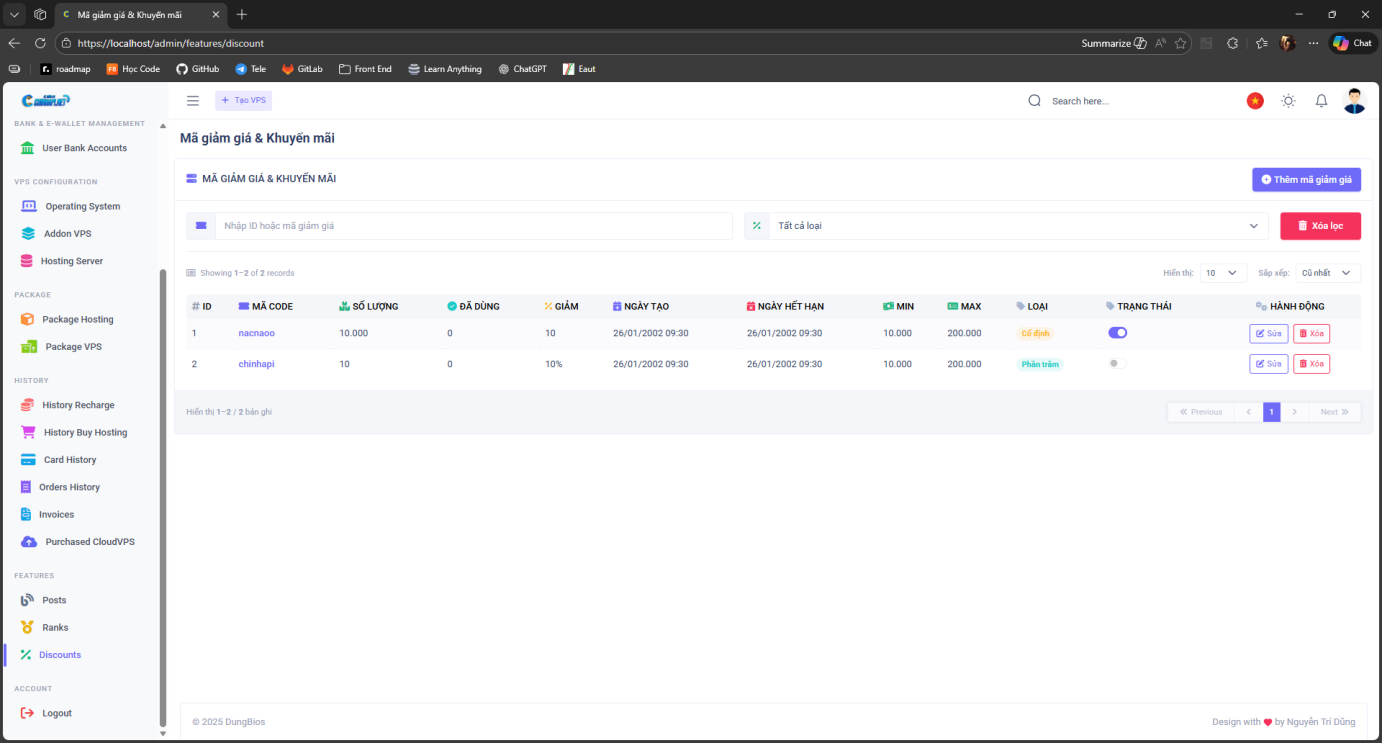
Hình 3. 13: Giao diện thêm hosting mới



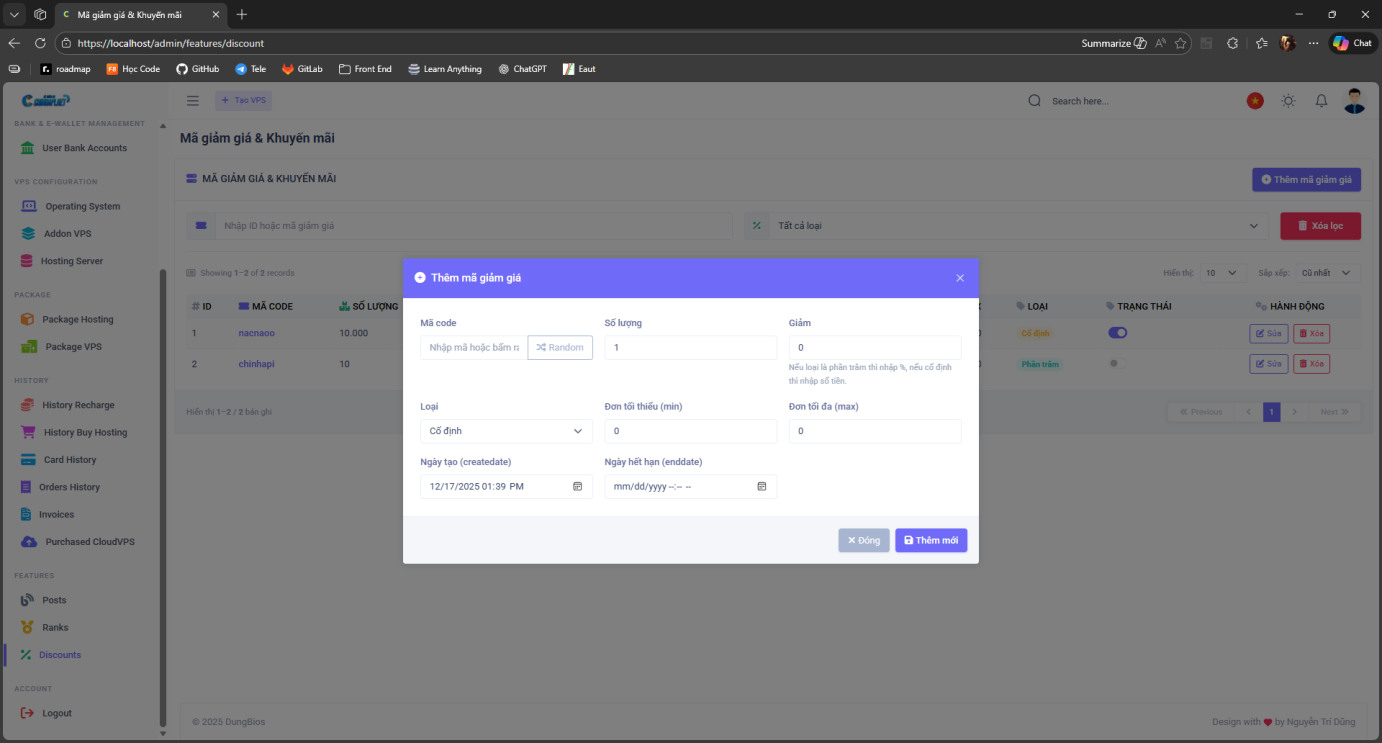
Hình 3. 14: Giao diện sửa thông tin hosting



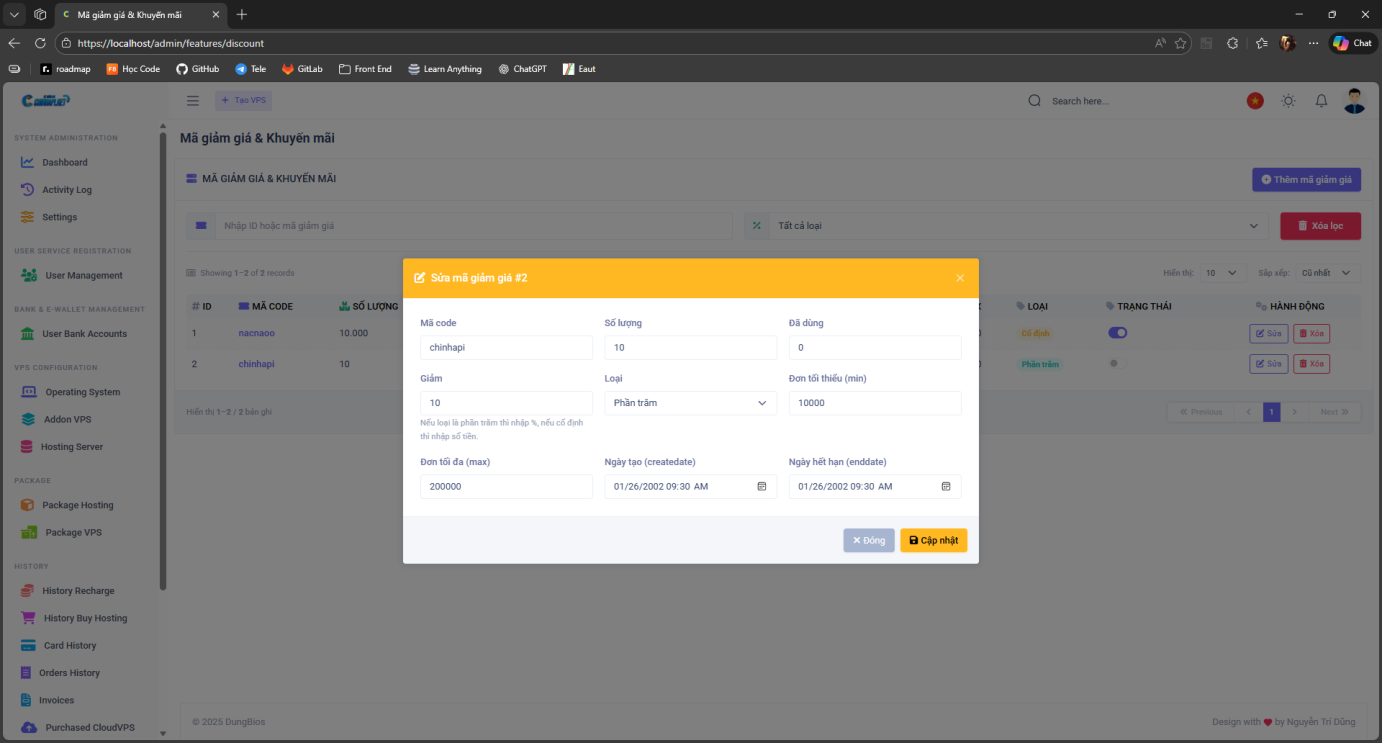
Hình 3. 15: Giao diện xóa hosting



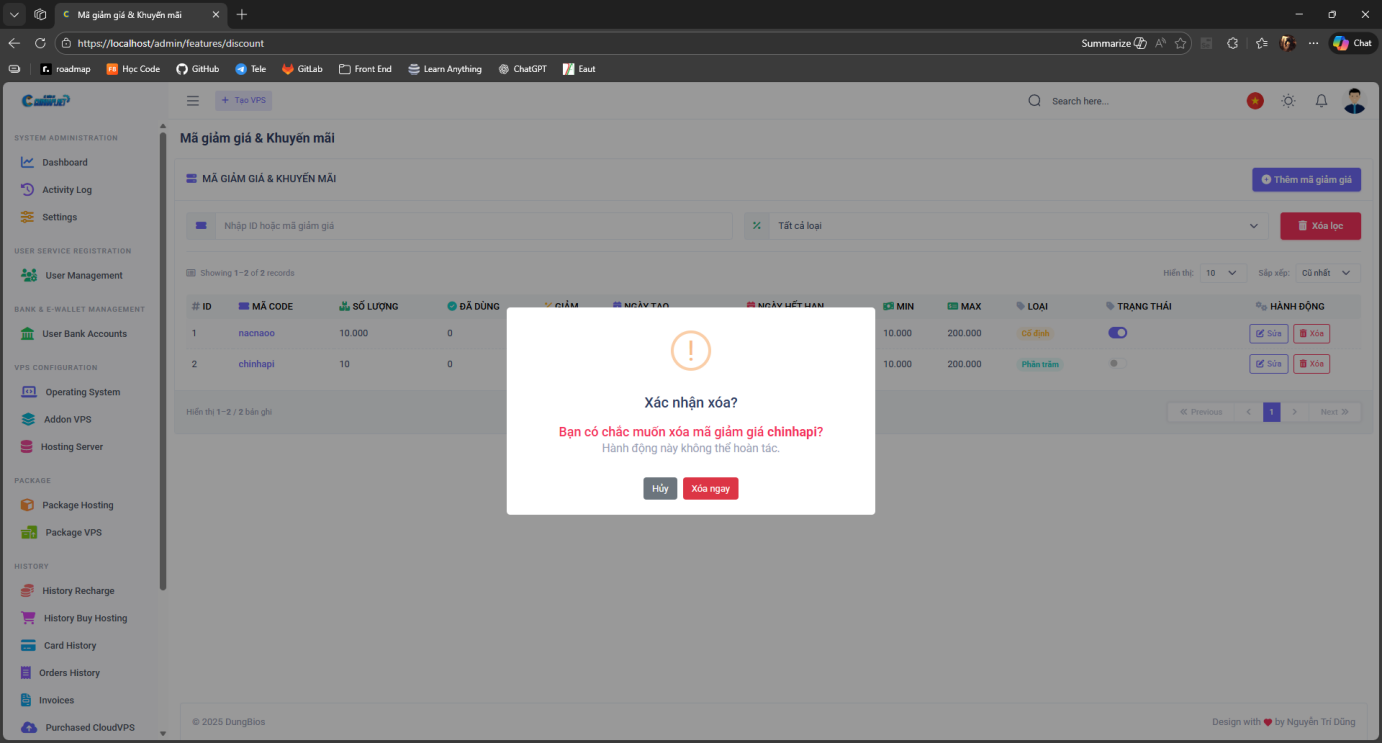
Hình 3. 16: Giao diện quản lý mã giảm giá



Hình 3. 17: Giao diện thêm mã giảm giá



Hình 3. 18: Giao diện sửa mã giảm giá



Hình 3. 19: Giao diện xóa mã giảm giá

# CHƯƠNG IV: KIỂM THỬ HỆ THỐNG

## 4.1 Giới thiệu kiểm thử

Kiểm thử hệ thống (System Testing) là giai đoạn quan trọng trong chu trình phát triển phần mềm, được thực hiện sau khi tích hợp đầy đủ các module. Mục tiêu của kiểm thử hệ thống là đánh giá toàn bộ hệ thống "Xây dựng website kinh doanh dịch vụ Hosting" dựa trên đặc tả yêu cầu ban đầu, đảm bảo hệ thống hoạt động chính xác, ổn định và đáp ứng đầy đủ nhu cầu nghiệp vụ thực tế trước khi đưa vào triển khai chính thức.

Trong đồ án này, nhóm sử dụng Selenium IDE như một công cụ hỗ trợ chính cho việc kiểm thử tự động giao diện người dùng (UI), giúp tăng hiệu quả, độ bao phủ và khả năng tái sử dụng các bài kiểm thử.

### 4.1.1 Lý thuyết về kiểm thử hệ thống

Kiểm thử hệ thống là mức kiểm thử cao nhất, nơi một hệ thống phần mềm hoàn chỉnh được kiểm tra trong môi trường mô phỏng hoặc giống với thực tế nhất có thể. Khác với kiểm thử đơn vị (Unit Test) hay kiểm thử tích hợp (Integration Test) tập trung vào từng phần nhỏ, kiểm thử hệ thống xem xét toàn bộ sản phẩm như một thể thống nhất.

**Phạm vi:** Toàn bộ hệ thống website, bao gồm tất cả các module (quản lý người dùng, đăng ký dịch vụ, thanh toán, quản trị hosting...), cơ sở dữ liệu, giao diện và các tương tác với môi trường bên ngoài (nếu có).

* Mục đích chính:

Xác nhận (Validation): Đảm bảo hệ thống xây dựng đúng với yêu cầu người dùng và mục tiêu kinh doanh.

Phát hiện lỗi: Tìm kiếm các lỗi tiềm ẩn về chức năng, hiệu năng, bảo mật và khả năng tương thích.

Đánh giá chất lượng tổng thể: Kiểm tra tính ổn định, độ tin cậy và khả năng sẵn sàng của hệ thống.

Quy trình kiểm thử điển hình: Lập kế hoạch kiểm thử → Thiết kế test case → Thiết lập môi trường kiểm thử → Thực thi kiểm thử → Phân tích và báo cáo kết quả.

### 4.1.2 Các loại kiểm thử hệ thống

Để đảm bảo chất lượng toàn diện, hệ thống website dịch vụ hosting được kiểm thử qua nhiều loại hình khác nhau, chủ yếu tập trung vào hai nhóm chính:

1. ***Kiểm thử chức năng (Functional Testing):***

**Mục tiêu:** Đảm bảo tất cả các chức năng của website hoạt động đúng theo đặc tả yêu cầu.

* Các bài kiểm tra tiêu biểu trong đồ án:

Kiểm thử giao diện người dùng (UI Testing): Sử dụng Selenium IDE để tự động hóa việc kiểm tra luồng đăng ký tài khoản, đăng nhập, chọn gói hosting, thêm vào giỏ hàng.

Kiểm thử tính năng thanh toán: Xác minh luồng thanh toán thành công/thất bại, tính toán tổng tiền chính xác.

Kiểm thử nghiệp vụ quản trị: Kiểm tra các chức năng CRUD (Thêm, Sửa, Xóa, Xem) đối với danh sách khách hàng, đơn hàng, dịch vụ trong khu vực Admin.

Kiểm thử tính hợp lệ của dữ liệu: Kiểm tra các trường nhập liệu, ràng buộc, và thông báo lỗi (ví dụ: email không hợp lệ, mật khẩu yếu).

1. ***Kiểm thử phi chức năng (Non-Functional Testing):***

**Mục tiêu:** Đánh giá các thuộc tính chất lượng của hệ thống như hiệu suất, độ tin cậy, khả năng sử dụng.

* Các bài kiểm tra được thực hiện:

Kiểm thử hiệu năng (Performance Testing): Đánh giá thời gian phản hồi của trang chủ, trang đăng nhập dưới tải bình thường. Kiểm tra khả năng xử lý của trang quản trị khi hiển thị danh sách lớn các đơn hàng.

Kiểm thử khả năng sử dụng (Usability Testing): Đánh giá trải nghiệm người dùng thông qua thử nghiệm với một nhóm người dùng mẫu, tập trung vào tính dễ hiểu của giao diện, tính nhất quán và khả năng điều hướng.

Kiểm thử tính tương thích (Compatibility Testing): Kiểm tra hiển thị và hoạt động của website trên các trình duyệt phổ biến (Chrome, Firefox, Edge) và trên các kích thước màn hình khác nhau (desktop, tablet, mobile).

## 4.2 Quy trình thực hiện

**Bước 1: Lập Kế Hoạch Kiểm Thử & Phân Tích Yêu Cầu**(Mục tiêu của bước này là xác định phạm vi, phương pháp và nguồn lực cho toàn bộ hoạt động kiểm thử.)

**Đầu vào:** Tài liệu Đặc Tả Yêu Cầu (SRS), Thiết Kế Hệ Thống, Các User Story.

* Hoạt động chính:

Xác định phạm vi kiểm thử: Liệt kê các tính năng chính cần kiểm tra (Đăng ký/Kích hoạt hosting, Quản lý tên miền, Hệ thống thanh toán, Dashboard quản trị...), đồng thời xác định các tính năng không thuộc phạm vi (ví dụ: tích hợp với cổng thanh toán thật ở giai đoạn này).

Xác định chiến lược kiểm thử và áp dụng kết hợp. Kiểm thử thủ công là các luồng nghiệp vụ phức tạp, kiểm thử khả năng sử dụng và giao diện; Kiểm thử tự động với Selenium IDE tập trung vào các luồng nghiệp vụ cốt lõi, có tính lặp lại cao (Regression Test). Ví dụ: tự động hóa việc đăng nhập, duyệt gói dịch vụ, thêm vào giỏ hàng.

Lập kế hoạch nguồn lực & tiến độ: Phân công thành viên phụ trách từng mảng kiểm thử (chức năng, giao diện, hiệu năng), ước lượng thời gian và xác định các mốc quan trọng (Milestone).

**Bước 2: Thiết Kế Test Case & Chuẩn Bị Môi Trường Kiểm Thử**(Bước này chuyển hóa các yêu cầu thành các tình huống kiểm thử cụ thể, có thể thực thi được.)

**Đầu vào:** Tài liệu từ Bước 1.

* Hoạt động chính:

Thiết kế Test Case: Tạo các test case chi tiết cho kiểm thử chức năng, bao gồm: ID Test Case, Mô tả, Các bước thực hiện (Step-by-Step), Dữ liệu đầu vào (Test Data), Kết quả mong đợi (Expected Result); Thiết kế các kịch bản kiểm thử tự động (Test Script) cho Selenium IDE: Ghi lại các thao tác trên trình duyệt (record) cho các luồng chính, sau đó chỉnh sửa và thêm các lệnh kiểm tra (assertion) để xác nhận kết quả; Xây dựng checklist cho kiểm thử phi chức năng: Các tiêu chí về hiệu năng (thời gian phản hồi < 3s), khả năng tương thích trình duyệt (Chrome, Firefox, Edge).

Xác định & Chuẩn bị Môi Trường Kiểm Thử: Môi trường kiểm thử (Testing Environment): Được sao chép từ môi trường phát triển, chạy trên máy chủ riêng (VPS/local server) với cấu hình tương tự production. Bao gồm: Web Server (Apache/Nginx), PHP, Database (MySQL), và toàn bộ mã nguồn của website.

**Bước 3: Thiết Lập Môi Trường & Dữ Liệu Kiểm Thử (**Bước chuẩn bị cuối cùng trước khi thực thi kiểm thử.**)**

Triển khai hệ thống lên môi trường kiểm thử: Cài đặt và cấu hình đầy đủ các thành phần.

Tạo và nạp dữ liệu kiểm thử (Test Data): Sử dụng database có sẵn kết hợp với tự sinh dữ liệu (mock data) để tạo các tình huống đa dạng. Ví dụ: 50 tài khoản khách hàng với các gói dịch vụ khác nhau, 100 đơn hàng ở nhiều trạng thái.

Cài đặt và cấu hình công cụ: Cài đặt Selenium IDE trên trình duyệt, thiết lập các project test suite riêng biệt cho từng module.

**Bước 4: Thực Thi Kiểm Thử & Ghi Nhận Kết Quả (**Tiến hành chạy các test case đã thiết kế và ghi lại kết quả một cách trung thực.**)**

Thực thi Test Case thủ công: Tester thực hiện từng bước theo test case đã viết, kiểm tra kết quả thực tế so với kết quả mong đợi.

Chạy Test Script tự động: Thực thi các bộ test script trong Selenium IDE. Công cụ sẽ tự động chạy, so sánh và báo cáo thành công (Pass) hay thất bại (Fail).

Kiểm thử phi chức năng: Sử dụng công cụ (ví dụ: Lighthouse, GTmetrix) để đo hiệu suất trang; kiểm tra thủ công giao diện trên nhiều trình duyệt và kích thước màn hình.

Ghi nhận kết quả & Báo lỗi: Mọi kết quả được ghi vào bảng tổng hợp. Khi phát hiện lỗi, tester tạo một Báo Cáo Lỗi (Bug Report) đầy đủ thông tin:

Tiêu đề lỗi: Mô tả ngắn gọn.

Môi trường: Trình duyệt, phiên bản, hệ điều hành.

Mức độ nghiêm trọng (Severity): Critical, Major, Minor.

Các bước tái hiện lỗi (Step to Reproduce).

Kết quả thực tế & mong đợi.

Ảnh chụp màn hình/Video.

**Bước 5: Phân Tích, Báo Cáo & Đánh Giá Kết Quả** (Bước cuối cùng tổng kết toàn bộ hoạt động kiểm thử và đưa ra khuyến nghị.)

Phân tích tổng hợp: Thống kê số lượng test case đã thực thi, tỉ lệ Pass/Fail, số lượng bug được tìm thấy và phân loại theo mức độ nghiêm trọng và theo module.

Đánh giá tiêu chí hoàn thành (Exit Criteria): Hệ thống chỉ được coi là vượt qua kiểm thử khi: Tất cả test case cho các chức năng cốt lõi đều Pass; Không còn lỗi Critical (Làm sập hệ thống, mất dữ liệu) hoặc Major (Chức năng chính không hoạt động) nào mở.

Các chỉ số hiệu năng cơ bản đạt yêu cầu.

Viết Báo Cáo Kiểm Thử (Test Summary Report): Tổng kết toàn bộ quy trình, kết quả, các vấn đề tồn đọng và khuyến nghị chính thức: Hệ thống ĐẠT hay KHÔNG ĐẠT để chuyển sang giai đoạn Kiểm Thử Chấp Nhận Sản Phẩm (UAT) hoặc cần sửa lỗi và kiểm thử lại.

## 4.3 Thiết kế các Test-case

Test-case được thiết kế dựa trên phương pháp Phân lớp tương đương (Equivalence Partitioning) và Phân tích giá trị biên (Boundary Value Analysis), đảm bảo độ bao phủ tối ưu với số lượng test-case hợp lý. Mỗi test-case đều tuân thủ cấu trúc "Mục tiêu - Hành động - Xác nhận".

### 4.3.1 Test case theo module Đăng ký

Bảng 4. 1: Module Đăng ký

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TC ID** | **Mô tả** | **Các bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** |
| REG-01 | Đăng ký thành công với thông tin hợp lệ | 1. Truy cập trang đăng ký 2. Điền đầy đủ thông tin 3. Click nút đăng ký | username: "user12345" email: "user123@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo "Đăng ký thành công" Backend: Trả về JSON: {"status":"success","msg":"Đăng ký thành công"}, tạo session, insert vào DB với level=1, band=0 |
| REG-02 | Đăng ký thất bại  - Thiếu thông tin bắt buộc | 1. Truy cập trang đăng ký 2. Để trống 1 trong các trường bắt buộc 3. Click đăng ký | username: "" (trống) email: "test@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Vui lòng điền đầy đủ thông tin"} |
| REG-03 | Đăng ký thất bại  - Username chứa ký tự đặc biệt | 1. Truy cập trang đăng ký 2. Nhập username có ký tự đặc biệt 3. Click đăng ký | username: "user@123" email: "test@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Tên đăng nhập không gồm ký tự đặc biệt và dấu"} |
| REG-04 | Đăng ký thất bại  - Username đã tồn tại | 1. Truy cập trang đăng ký 2. Nhập username đã có trong DB 3. Click đăng ký | username: "admin" (đã tồn tại) email: "new@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Người dùng đã tồn tại"}, kiểm tra query SELECT từ bảng users |
| REG-05 | Đăng ký thất bại  - Username quá ngắn (<5) | 1. Truy cập trang đăng ký 2. Nhập username 4 ký tự 3. Click đăng ký | username: "user" email: "test@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Tên đăng nhập phải tối thiểu 5 ký tự và tối đa 20 ký tự"} |
| REG-06 | Đăng ký thất bại  - Username quá dài (>20) | 1. Truy cập trang đăng ký 2. Nhập username 21 ký tự 3. Click đăng ký | username: "usernameverylong123456" email: "test@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Tên đăng nhập phải tối thiểu 5 ký tự và tối đa 20 ký tự"} |
| REG-07 | Đăng ký thất bại  - Email đã tồn tại | 1. Truy cập trang đăng ký 2. Nhập email đã có trong DB 3. Click đăng ký | username: "newuser123" email: "existing@gmail.com" (đã có) password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Email đã được sử dụng"}, kiểm tra query SELECT từ bảng users |
| REG-08 | Đăng ký thất bại  - Email không phải @[gmail.com](https://gmail.com/" \t "https://chat.deepseek.com/a/chat/s/_blank) | 1. Truy cập trang đăng ký 2. Nhập email không kết thúc bằng @[gmail.com](https://gmail.com/" \t "https://chat.deepseek.com/a/chat/s/_blank) 3. Click đăng ký | username: "newuser123" email: "user@yahoo.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Email không hợp lệ"} |
| REG-09 | Đăng ký thất bại  - Email có ký tự đặc biệt không cho phép trước @ | 1. Truy cập trang đăng ký 2. Nhập email có ký tự không nằm trong regex 3. Click đăng ký | username: "newuser123" email: "user!#$@[gmail.com](https://gmail.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả về JSON: {"status":"error","msg":"Email không hợp lệ"} |
| REG-10 | Đăng ký thành công  - Kiểm tra dữ liệu được lưu đúng | 1. Đăng ký thành công 2. Kiểm tra database | username: "testuser123" email: "testuser123@gmail.com" password: "Test@123" confirmpassword: "Test@123" phone: "0912345679" | Backend: Dữ liệu trong bảng users: - password được mã hóa (TypePassword) - money = 0, total\_money = 0 - session\_token = 32 ký tự - level = 1, band = 0 - veri\_email = 'off' - createdate và time = timestamp hiện tại |
| REG-11 | Kiểm tra chống XSS | 1. Truy cập trang đăng ký 2. Nhập dữ liệu có script 3. Click đăng ký | username: "user<script>alert(1)</script>" email: "test@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Backend: Hàm antixss() xử lý và loại bỏ/thay thế mã độc, không thực thi script |
| REG-12 | Đăng ký thất bại  - Không gửi action REGISTER | 1. Truy cập trang đăng ký 2. Gửi POST request không có action='REGISTER' | POST data không có action hoặc action khác | Backend: Không xử lý đăng ký, không có response hoặc response mặc định |
| REG-13 | Kiểm tra transaction rollback khi insert thất bại | 1. Tạo lỗi DB (ví dụ: tên bảng sai) 2. Thử đăng ký | username: "rollbacktest" email: "rollback@gmail.com" password: "Pass@123" confirmpassword: "Pass@123" phone: "0912345678" | Backend: Thực hiện rollback(), không có dữ liệu nào được insert, trả về JSON: {"status":"error","msg":"Đăng ký thất bại"} |

### 4.3.2 Test case theo module Đăng nhập

Bảng 4. 2: Module Đăng nhập

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TC ID** | **Mô tả** | **Các bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** |
| LOGIN-01 | Đăng nhập thành công với username và password đúng | 1. Truy cập trang đăng nhập 2. Nhập username và password hợp lệ 3. Click nút đăng nhập | username: "admin" (tồn tại) password: "Abc@123" (khớp với DB) | Frontend: Hiển thị "Đăng nhập thành công" Backend: Trả JSON success, cập nhật session\_token, ghi log his\_login, tạo session |
| LOGIN-02 | Đăng nhập thất bại  - Thiếu thông tin | 1. Truy cập trang đăng nhập 2. Để trống username hoặc password 3. Click đăng nhập | username: "" password: "Abc@123" | Frontend: Hiển thị thông báo lỗi Backend: Trả JSON: {"status":"error","msg":"Vui lòng nhập đủ thông tin"} |
| LOGIN-03 | Đăng nhập thất bại - Username không tồn tại | 1. Truy cập trang đăng nhập 2. Nhập username không có trong DB 3. Click đăng nhập | username: "nonexistentuser" password: "AnyPassword123" | Frontend: Hiển thị "Tài khoản không tồn tại" Backend: Trả JSON: {"status":"error","msg":"Tài khoản không tồn tại"} |
| LOGIN-04 | Đăng nhập thất bại  - Mật khẩu sai | 1. Truy cập trang đăng nhập 2. Nhập username đúng, password sai 3. Click đăng nhập | username: "admin" (tồn tại) password: "WrongPassword123" | Frontend: Hiển thị "Mật khẩu không chính xác" Backend: Trả JSON: {"status":"error","msg":"Mật khẩu không chính xác"} |
| LOGIN-05 | Đăng nhập thất bại  - Tài khoản bị band (nếu có logic) | 1. Truy cập trang đăng nhập 2. Nhập username của tài khoản bị khóa 3. Click đăng nhập | username: "banneduser" (band=1) password: "CorrectPass123" | Frontend: Hiển thị thông báo tài khoản bị khóa (nếu có) Backend: Không có kiểm tra band trong code hiện tại - đây là lỗ hổng |
| LOGIN-06 | Kiểm tra chống XSS ở đầu vào | 1. Truy cập trang đăng nhập 2. Nhập dữ liệu có script 3. Click đăng nhập | username: <script>alert(1)</script> password: "AnyPassword" | Backend: Hàm antixss() xử lý, không thực thi script, đăng nhập thất bại do username không tồn tại |
| LOGIN-07 | Kiểm tra SQL Injection | 1. Truy cập trang đăng nhập 2. Nhập payload SQL 3. Click đăng nhập | username: ' OR '1'='1 password: "any" | Backend: Không bị SQLi do sử dụng hàm antixss() và query không concatenate trực tiếp (cần xem lại hàm get\_row) |
| LOGIN-08 | Đăng nhập thất bại  - Không gửi action LOGIN | 1. Truy cập trang đăng nhập 2. Gửi POST request không có action='LOGIN' | POST data không có action hoặc action khác | Backend: Không xử lý đăng nhập, không có response |
| LOGIN-09 | Kiểm tra update thông tin khi đăng nhập thành công | 1. Đăng nhập thành công 2. Kiểm tra bảng users | username: "testuser" password: "CorrectPass123" | Backend: Cập nhật các trường: ip\_adr, address, time, session\_token trong bảng users |
| LOGIN-10 | Kiểm tra ghi log lịch sử đăng nhập | 1. Đăng nhập thành công 2. Kiểm tra bảng his\_login | username: "testuser" password: "CorrectPass123" | Backend: Insert 1 bản ghi vào his\_login với: username, title="Đăng nhập hệ thống", browser, device, ip, address, time |
| LOGIN-11 | Kiểm tra transaction rollback khi update thất bại | 1. Tạo lỗi khi update (ví dụ: tên bảng sai) 2. Thử đăng nhập | username: "testuser" password: "CorrectPass123" | Backend: Rollback transaction, không update users, không insert his\_login, trả về "Đăng nhập thất bại" |
| LOGIN-12 | Đăng nhập thành công  - Kiểm tra session được tạo | 1. Đăng nhập thành công 2. Kiểm tra session server-side | username: "testuser" password: "CorrectPass123" | Backend: $\_SESSION['session'] được set với session\_token 32 ký tự, có thể sử dụng cho các request sau |
| LOGIN-13 | Đăng nhập với tài khoản có session\_token cũ | 1. Tài khoản có session\_token cũ trong DB 2. Đăng nhập thành công | username: "existinguser" password: "CorrectPass123" | Backend: session\_token được thay thế bằng giá trị mới 32 ký tự, session cũ bị vô hiệu hóa |
| LOGIN-14 | Kiểm tra đăng nhập nhiều lần liên tiếp | 1. Đăng nhập thành công từ 2 thiết bị khác nhau 2. Kiểm tra session | username: "testuser" password: "CorrectPass123" | Backend: Mỗi lần đăng nhập tạo session\_token mới, ghi đè session\_token cũ, chỉ phiên đăng nhập cuối cùng có hiệu lực |

### 4.3.3 Test case theo module Đổi mật khẩu

Bảng 4. 3: Module Đổi mật khẩu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TC ID** | **Mô tả** | **Các bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** |
| CHPASS-01 | Đổi mật khẩu thành công với thông tin hợp lệ | 1. Đăng nhập hệ thống 2. Truy cập trang đổi mật khẩu 3. Nhập mật khẩu cũ đúng, mật khẩu mới hợp lệ 4. Click đổi mật khẩu | passwordOld: "Current@123" passwordNew: "NewPass@456" passwordCf: "NewPass@456" | Frontend: Hiển thị "Đổi mật khẩu thành công" Backend: Trả JSON success, update password trong DB, ghi đè bằng mã hóa mới |
| CHPASS-02 | Đổi mật khẩu thất bại - Không đăng nhập (không có quyền) | 1. Truy cập trang đổi mật khẩu khi chưa đăng nhập 2. Nhập thông tin 3. Click đổi mật khẩu | passwordOld: "any" passwordNew: "any" passwordCf: "any" | Frontend: Hiển thị lỗi Backend: Trả JSON: {"status":"error","msg":"Bạn không có quyền thực hiện hành động này"} (checkUser trả false) |
| CHPASS-03 | Đổi mật khẩu thất bại - Thiếu thông tin | 1. Đăng nhập 2. Truy cập trang đổi mật khẩu 3. Để trống 1 trong 3 trường 4. Click đổi mật khẩu | passwordOld: "" passwordNew: "NewPass@456" passwordCf: "NewPass@456" | Frontend: Hiển thị lỗi Backend: Trả JSON: {"status":"error","msg":"Vui lòng nhập đủ thông tin"} |
| CHPASS-04 | Đổi mật khẩu thất bại - Mật khẩu cũ sai | 1. Đăng nhập 2. Truy cập trang đổi mật khẩu 3. Nhập mật khẩu cũ không đúng 4. Click đổi mật khẩu | passwordOld: "WrongOldPass" passwordNew: "NewPass@456" passwordCf: "NewPass@456" | Frontend: Hiển thị "Mật khẩu cũ không chính xác" Backend: Trả JSON: {"status":"error","msg":"Mật khẩu cũ không chính xác"} (verify\_password trả false) |
| CHPASS-05 | Đổi mật khẩu thất bại - Mật khẩu mới quá ngắn (<6) | 1. Đăng nhập 2. Truy cập trang đổi mật khẩu 3. Nhập mật khẩu mới 5 ký tự 4. Click đổi mật khẩu | passwordOld: "Current@123" passwordNew: "New12" passwordCf: "New12" | Frontend: Hiển thị lỗi Backend: Trả JSON: {"status":"error","msg":"Độ dài mật khẩu phải lớn hơn 6 kí tự"} |
| CHPASS-06 | Đổi mật khẩu thất bại - Mật khẩu mới và xác nhận không khớp | 1. Đăng nhập 2. Truy cập trang đổi mật khẩu 3. Nhập mật khẩu mới và xác nhận khác nhau 4. Click đổi mật khẩu | passwordOld: "Current@123" passwordNew: "NewPass@456" passwordCf: "Different@789" | Frontend: Hiển thị "Nhập lại mật khẩu không khớp" Backend: Trả JSON: {"status":"error","msg":"Nhập lại mật khẩu không khớp"} |
| CHPASS-07 | Kiểm tra mật khẩu mới đúng độ dài biên (6 ký tự) | 1. Đăng nhập 2. Truy cập trang đổi mật khẩu 3. Nhập mật khẩu mới đúng 6 ký tự 4. Click đổi mật khẩu | passwordOld: "Current@123" passwordNew: "New12@" passwordCf: "New12@" | Frontend: Hiển thị thành công Backend: Trả JSON success, password được cập nhật (độ dài 6 ký tự hợp lệ) |
| CHPASS-08 | Kiểm tra mật khẩu mới độ dài lớn (>6) | 1. Đăng nhập 2. Truy cập trang đổi mật khẩu 3. Nhập mật khẩu mới 20 ký tự 4. Click đổi mật khẩu | passwordOld: "Current@123" passwordNew: "VeryLongPassword@123456" passwordCf: "VeryLongPassword@123456" | Frontend: Hiển thị thành công Backend: Trả JSON success, password được cập nhật (không giới hạn độ dài tối đa) |
| CHPASS-09 | Kiểm tra chống XSS | 1. Đăng nhập 2. Truy cập trang đổi mật khẩu 3. Nhập dữ liệu có script 4. Click đổi mật khẩu | passwordOld: "<script>alert(1)</script>" passwordNew: "NewPass@456" passwordCf: "NewPass@456" | Backend: Hàm antixss() xử lý và loại bỏ/thay thế mã độc, mật khẩu cũ sai (do đã bị thay đổi) |
| CHPASS-10 | Kiểm tra update database thành công | 1. Đăng nhập với user A 2. Đổi mật khẩu thành công 3. Kiểm tra DB | passwordOld: "OldPass123" passwordNew: "NewPass456" passwordCf: "NewPass456" | Backend: Field password trong bảng users được cập nhật bằng giá trị mới đã mã hóa (TypePassword) |
| CHPASS-11 | Kiểm tra transaction rollback khi update thất bại | 1. Đăng nhập 2. Tạo lỗi DB (ví dụ: tên bảng sai) 3. Thử đổi mật khẩu | passwordOld: "Current@123" passwordNew: "NewPass@456" passwordCf: "NewPass@456" | Backend: Rollback transaction, password không thay đổi, trả về "Đổi mật khẩu thất bại" |
| CHPASS-12 | Đổi mật khẩu thành công - Có thể đăng nhập bằng mật khẩu mới | 1. Đăng nhập user A 2. Đổi mật khẩu thành công 3. Đăng xuất 4. Thử đăng nhập bằng mật khẩu mới | passwordOld: "OldPass123" passwordNew: "NewPass456" passwordCf: "NewPass456" | Backend: Đăng nhập thành công với mật khẩu mới (verify\_password trả true), không thể đăng nhập bằng mật khẩu cũ |
| CHPASS-13 | Đổi mật khẩu thất bại - Không gửi action CHANGE\_PASS | 1. Đăng nhập 2. Gửi POST request không có action='CHANGE\_PASS' | POST data không có action hoặc action khác | Backend: Không xử lý đổi mật khẩu, không có response |

### 4.3.4 Test case theo module Mua hàng & Thanh toán

Bảng 4. 4: Module Mua hàng & Thanh toán

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TC ID** | **Mô tả** | **Các bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** |
| BUY-01 | Mua hosting thành công (1 tháng, không giảm giá) | 1. Đăng nhập 2. Chọn gói hosting 3. Nhập domain 4. Click thanh toán | id\_host: "1" (status='on') month: "1" discount: "" domain: "[example.com](https://example.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: 500.000đ Package price: 100.000đ/tháng | Frontend: Hiển thị "Thanh toán thành công" Backend: Trả JSON success, trừ tiền, tạo 3 bản ghi (history\_buy\_hosting, orders, invoices) |
| BUY-02 | Mua hosting thất bại - Chưa đăng nhập | 1. Truy cập trang mua hàng khi chưa đăng nhập 2. Nhập thông tin 3. Click thanh toán | id\_host: "1" month: "1" discount: "" domain: "[test.com](https://test.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" | Frontend: Hiển thị lỗi Backend: Trả JSON: {"status":"error","msg":"Bạn không có quyền thực hiện hành động này"} (checkUser trả false) |
| BUY-03 | Mua hosting thất bại - Thiếu thông tin | 1. Đăng nhập 2. Để trống 1 trong 4 trường 3. Click thanh toán | id\_host: "" month: "1" discount: "" domain: "[test.com](https://test.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" | Frontend: Hiển thị "Thiếu thông tin" Backend: Trả JSON: {"status":"error","msg":"Thiếu thông tin"} |
| BUY-04 | Mua hosting thất bại - Hosting không tồn tại | 1. Đăng nhập 2. Nhập id\_host không có trong DB 3. Click thanh toán | id\_host: "999" (không tồn tại) month: "1" discount: "" domain: "[test.com](https://test.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" | Frontend: Hiển thị "Hosting không tồn tại" Backend: Trả JSON: {"status":"error","msg":"Hosting không tồn tại"} |
| BUY-05 | Mua hosting thất bại - Hosting status không phải 'on' | 1. Đăng nhập 2. Chọn gói hosting có status='off' 3. Click thanh toán | id\_host: "2" (status='off') month: "1" discount: "" domain: "[test.com](https://test.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" | Frontend: Hiển thị "Hosting không tồn tại" Backend: Trả JSON: {"status":"error","msg":"Hosting không tồn tại"} (do điều kiện AND status='on') |
| BUY-06 | Mua hosting thất bại - Tên miền không hợp lệ (không có dấu chấm) | 1. Đăng nhập 2. Nhập domain không có dấu chấm 3. Click thanh toán | id\_host: "1" month: "1" discount: "" domain: "testdomain" | Frontend: Hiển thị "Tên miền bạn nhập không hợp lệ!!" Backend: Trả JSON: {"status":"error","msg":"Tên miền bạn nhập không hợp lệ!!"} |
| BUY-07 | Mua hosting thất bại - Số dư không đủ | 1. Đăng nhập với user có 50.000đ 2. Chọn gói 100.000đ 3. Click thanh toán | id\_host: "1" (giá 100.000đ) month: "1" discount: "" domain: "[test.com](https://test.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: 50.000đ | Frontend: Hiển thị "Số dư của bạn không đủ" Backend: Trả JSON: {"status":"error","msg":"Số dư của bạn không đủ"} |
| BUY-08 | Mua hosting thất bại - Tên miền đã tồn tại | 1. Đăng nhập 2. Nhập domain đã có trong history\_buy\_hosting 3. Click thanh toán | id\_host: "1" month: "1" discount: "" domain: "[existing.com](https://existing.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" (đã có) | Frontend: Hiển thị "Tên miền đã tồn tại" Backend: Trả JSON: {"status":"error","msg":"Tên miền đã tồn tại"} |
| BUY-09 | Mua hosting thành công - Giảm giá theo tháng (12 tháng -10%) | 1. Đăng nhập 2. Chọn gói 100.000đ 3. Chọn 12 tháng 4. Click thanh toán | id\_host: "1" (100.000đ) month: "12" discount: "" domain: "[test12.com](https://test12.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: đủ | Backend: Tính toán: 100.000 \* 12 = 1.200.000 - 10% = 1.080.000đ, trừ đúng số tiền này |
| BUY-10 | Mua hosting thành công - Giảm giá theo tháng (24 tháng -20%) | 1. Đăng nhập 2. Chọn gói 100.000đ 3. Chọn 24 tháng 4. Click thanh toán | id\_host: "1" (100.000đ) month: "24" discount: "" domain: "[test24.com](https://test24.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: đủ | Backend: Tính toán: 100.000 \* 24 = 2.400.000 - 20% = 1.920.000đ |
| BUY-11 | Mua hosting thành công - Giảm giá theo tháng (60 tháng -40%) | 1. Đăng nhập 2. Chọn gói 100.000đ 3. Chọn 60 tháng 4. Click thanh toán | id\_host: "1" (100.000đ) month: "60" discount: "" domain: "[test60.com](https://test60.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: đủ | Backend: Tính toán: 100.000 \* 60 = 6.000.000 - 40% = 3.600.000đ |
| BUY-12 | Mua hosting thành công - Áp dụng mã giảm giá | 1. Đăng nhập 2. Chọn gói 100.000đ 3. Nhập mã giảm giá hợp lệ 4. Click thanh toán | id\_host: "1" (100.000đ) month: "1" discount: "SAVE10" (giảm 10%) domain: "[testdiscount.com](https://testdiscount.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: đủ | Backend: Gọi hàm checkDiscount(100.000, "SAVE10") trả về 90.000đ, trừ đúng số tiền này |
| BUY-13 | Kiểm tra transaction rollback khi insert thất bại | 1. Đăng nhập 2. Tạo lỗi DB khi insert 3. Thử mua hosting | id\_host: "1" month: "1" discount: "" domain: "[rollback.com](https://rollback.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: đủ | Backend: Rollback transaction, không trừ tiền user, không tạo bản ghi nào, trả về "Thanh toán thất bại" |
| BUY-14 | Kiểm tra dữ liệu được lưu đúng trong history\_buy\_hosting | 1. Mua hosting thành công 2. Kiểm tra DB | id\_host: "1" month: "6" discount: "" domain: "[verify.com](https://verify.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" | Backend: Bảng history\_buy\_hosting có: - username đúng - domain: "[verify.com](https://verify.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" - pk\_host: code\_host từ package - account: encryptData(to\_slug(domain)) - password: encryptData(random 12 ký tự) - endAt: now + 6\*30\*24\*60\*60 |
| BUY-15 | Kiểm tra chống XSS | 1. Đăng nhập 2. Nhập domain có script 3. Click thanh toán | id\_host: "1" month: "1" discount: "" domain: "<script>alert(1)</script>.com" | Backend: Hàm antixss() xử lý, domain không hợp lệ (không có dấu chấm đúng vị trí) |
| BUY-16 | Mua hosting với tháng không có giảm giá (3 tháng) | 1. Đăng nhập 2. Chọn gói 100.000đ 3. Chọn 3 tháng 4. Click thanh toán | id\_host: "1" (100.000đ) month: "3" discount: "" domain: "[test3.com](https://test3.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: đủ | Backend: Tính toán: 100.000 \* 3 = 300.000đ (không giảm giá), trừ đúng số tiền này |
| BUY-17 | Kiểm tra tính năng trừ tiền khi update\_user thất bại | 1. Đăng nhập 2. Tạo lỗi khi trừ tiền 3. Thử mua hosting | id\_host: "1" month: "1" discount: "" domain: "[fail.com](https://fail.com/" \t "https://chat.deepseek.com/a/chat/s/_blank)" User money: đủ | Backend: Rollback transaction, trả về "Thanh toán thất bại", không trừ tiền |

### 4.3.5 Test case theo module Quản lý hosting (Thêm, Sửa, Xóa)

Bảng 4. 5: Module Quản lý hosting

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TC ID** | **Mô tả** | **Các bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** |
| HOST-ADD-01 | Thêm gói hosting thành công với đầy đủ thông tin hợp lệ | 1. Truy cập trang quản lý hosting 2. Click "Thêm mới" 3. Nhập đầy đủ thông tin 4. Click "Lưu" | name\_host: "VN ALO" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "100000" disk: "10 GB" other\_domain: "5" alias\_domain: "10" status: "on" | Frontend: Hiển thị "Thêm gói hosting thành công" Backend: Trả JSON success, insert vào bảng package\_hosting với name\_host="VN\_ALO", code\_host="vn\_alo" |
| HOST-ADD-02 | Thêm gói hosting thất bại  - Thiếu thông tin bắt buộc | 1. Truy cập trang thêm hosting 2. Để trống một trường bắt buộc 3. Click "Lưu" | name\_host: "" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "100000" disk: "10 GB" | Frontend: Hiển thị "Vui lòng nhập đầy đủ thông tin gói hosting" Backend: Trả JSON: {"status":"error","msg":"Vui lòng nhập đầy đủ thông tin gói hosting"} |
| HOST-ADD-03 | Thêm gói hosting thất bại  - Giá tiền không hợp lệ (không phải số) | 1. Truy cập trang thêm hosting 2. Nhập money không phải số 3. Click "Lưu" | name\_host: "TEST PACKAGE" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "abc" disk: "10 GB" | Frontend: Hiển thị "Giá tiền không hợp lệ" Backend: Trả JSON: {"status":"error","msg":"Giá tiền không hợp lệ"} |
| HOST-ADD-04 | Thêm gói hosting thất bại  - Giá tiền âm hoặc bằng 0 | 1. Truy cập trang thêm hosting 2. Nhập money ≤ 0 3. Click "Lưu" | name\_host: "TEST PACKAGE" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "0" disk: "10 GB" | Frontend: Hiển thị "Giá tiền không hợp lệ" Backend: Trả JSON: {"status":"error","msg":"Giá tiền không hợp lệ"} |
| HOST-ADD-05 | Thêm gói hosting thất bại  - Tên gói đã tồn tại (trùng code\_host) | 1. Truy cập trang thêm hosting 2. Nhập tên gói đã có trong hệ thống 3. Click "Lưu" | name\_host: "VN ALO" (đã tồn tại) server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "150000" disk: "15 GB" | Frontend: Hiển thị "Tên gói này đã tồn tại trong hệ thống (code trùng)" Backend: Trả JSON: {"status":"error","msg":"Tên gói này đã tồn tại trong hệ thống (code trùng)"} |
| HOST-ADD-06 | Kiểm tra chuẩn hóa tên gói (tự động chuyển đổi) | 1. Truy cập trang thêm hosting 2. Nhập name\_host có dấu cách 3. Click "Lưu" | name\_host: "Gói Cao Cấp" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "200000" disk: "20 GB" | Backend: Tự động chuyển đổi: - name\_host\_upper: "GÓI\_CAO\_CẤP" - code\_host: "gói\_cao\_cấp" Lưu đúng vào DB |
| HOST-ADD-07 | Thêm gói hosting với status mặc định "off" | 1. Truy cập trang thêm hosting 2. Không chọn status (mặc định) 3. Click "Lưu" | name\_host: "Gói Mặc Định" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "50000" disk: "5 GB" status: "" (sẽ thành "off") | Backend: Trả JSON success, status được lưu là "off" trong DB |
| HOST-ADD-08 | Kiểm tra chống XSS trong input | 1. Truy cập trang thêm hosting 2. Nhập dữ liệu có script 3. Click "Lưu" | name\_host: "<script>alert(1)</script>" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "100000" disk: "10 GB" | Backend: Hàm antixss() xử lý, loại bỏ script, lưu dữ liệu đã được làm sạch |
| HOST-ADD-09 | Kiểm tra transaction rollback khi insert thất bại | 1. Truy cập trang thêm hosting 2. Tạo lỗi DB khi insert 3. Thử thêm gói mới | name\_host: "Rollback Test" server\_host: "[server1.vn](https://server1.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "100000" disk: "10 GB" | Backend: Rollback transaction, trả về "Thêm gói hosting thất bại", không có bản ghi nào được insert |
| HOST-UPDATE-01 | Cập nhật gói hosting thành công | 1. Truy cập danh sách hosting 2. Click "Sửa" trên gói cần chỉnh 3. Thay đổi thông tin 4. Click "Cập nhật" | id: "5" (tồn tại) name\_host: "GÓI VIP" server\_host: "[server2.vn](https://server2.vn/" \t "https://chat.deepseek.com/a/chat/s/_blank)" money: "250000" disk: "25 GB" other\_domain: "10" alias\_domain: "20" status: "on" | Frontend: Hiển thị "Cập nhật gói hosting thành công" Backend: Trả JSON success, update bảng package\_hosting với ID=5 |
| HOST-UPDATE-02 | Cập nhật thất bại  - Thiếu ID | 1. Gửi request update không có ID 2. Click "Cập nhật" | id: "" hoặc 0 Các trường khác có giá trị | Frontend: Hiển thị "Thiếu ID gói hosting" Backend: Trả JSON: {"status":"error","msg":"Thiếu ID gói hosting"} |
| HOST-UPDATE-03 | Cập nhật thất bại  - Gói hosting không tồn tại | 1. Truy cập trang sửa hosting 2. Nhập ID không có trong DB 3. Click "Cập nhật" | id: "999" (không tồn tại) Các trường khác có giá trị | Frontend: Hiển thị "Gói hosting không tồn tại" Backend: Trả JSON: {"status":"error","msg":"Gói hosting không tồn tại"} |
| HOST-UPDATE-04 | Cập nhật thành công  - Chỉ thay đổi một số trường | 1. Truy cập trang sửa hosting 2. Chỉ thay đổi money và status 3. Click "Cập nhật" | id: "5" (tồn tại) money: "300000" (tăng từ 250000) status: "off" (chuyển từ on) | Backend: Trả JSON success, chỉ update 2 trường money và status, các trường khác giữ nguyên |
| HOST-UPDATE-05 | Kiểm tra update với giá tiền hợp lệ (số thập phân) | 1. Truy cập trang sửa hosting 2. Nhập money có số thập phân 3. Click "Cập nhật" | id: "5" money: "149.99" Các trường khác giữ nguyên | Backend: Trả JSON success, lưu "149.99" vào DB (nếu kiểu dữ liệu cho phép) |
| HOST-UPDATE-06 | Kiểm tra update với status không hợp lệ | 1. Truy cập trang sửa hosting 2. Nhập status không phải "on"/"off" 3. Click "Cập nhật" | id: "5" status: "invalid\_status" | Backend: Vẫn lưu giá trị "invalid\_status" (không có validate) - đây là lỗ hổng |
| HOST-UPDATE-07 | Kiểm tra transaction rollback khi update thất bại | 1. Truy cập trang sửa hosting 2. Tạo lỗi DB khi update 3. Thử cập nhật | id: "5" money: "350000" Các trường khác giữ nguyên | Backend: Rollback transaction, trả về "Cập nhật thất bại", dữ liệu không thay đổi |
| HOST-DELETE-01 | Xóa gói hosting thành công | 1. Truy cập danh sách hosting 2. Click "Xóa" trên gói cần xóa 3. Xác nhận xóa | id: "10" (tồn tại) | Frontend: Hiển thị "Đã xóa gói hosting thành công" Backend: Trả JSON success, xóa bản ghi với ID=10 khỏi package\_hosting |
| HOST-DELETE-02 | Xóa thất bại  - Thiếu ID | 1. Gửi request delete không có ID 2. Click "Xóa" | id: "" hoặc 0 | Frontend: Hiển thị "Thiếu ID cần xóa" Backend: Trả JSON: {"status":"error","msg":"Thiếu ID cần xóa"} |
| HOST-DELETE-03 | Xóa thất bại  - Gói hosting không tồn tại | 1. Truy cập trang xóa hosting 2. Nhập ID không có trong DB 3. Click "Xóa" | id: "999" (không tồn tại) | Frontend: Hiển thị "Gói hosting không tồn tại" Backend: Trả JSON: {"status":"error","msg":"Gói hosting không tồn tại"} |
| HOST-DELETE-04 | Kiểm tra xóa gói hosting đang có người dùng (khóa ngoại) | 1. Truy cập danh sách hosting 2. Chọn gói đang có khách hàng sử dụng 3. Click "Xóa" | id: "1" (đang có bản ghi trong history\_buy\_hosting) | Backend: Có thể lỗi constraint foreign key (nếu có), trả về "Xóa gói hosting thất bại" |
| HOST-DELETE-05 | Kiểm tra transaction rollback khi delete thất bại | 1. Truy cập trang xóa hosting 2. Tạo lỗi DB khi delete 3. Thử xóa | id: "5" (tồn tại) | Backend: Rollback transaction, trả về "Xóa gói hosting thất bại", bản ghi không bị xóa |

### 4.3.6 Test case theo module Quản lý thành viên (Thêm, Sửa, Xóa)

Bảng 4. 6: Module Quản lý thành viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TC ID** | **Mô tả** | **Các bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** |
| MEMBER-ADD-01 | Admin thêm thành viên mới thành công | 1. Đăng nhập admin 2. Truy cập trang quản lý thành viên 3. Click "Thêm thành viên" 4. Điền thông tin hợp lệ 5. Click "Lưu" | fullname: "Nguyễn Văn A" username: "newuser123" password: "Pass@123" email: "newuser@gmail.com" phone: "0912345678" level: 0 band: 0 | Frontend: Hiển thị "Thêm thành viên thành công" Backend: Insert vào bảng users, password mã hóa, các trường mặc định: money=0, total\_money=0, veri\_email='off' |
| MEMBER-ADD-02 | Thêm thành viên thất bại  - Thiếu thông tin bắt buộc | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Để trống username 4. Click "Lưu" | username: "" email: "test@gmail.com" password: "Pass@123" phone: "0912345678" | Frontend: Hiển thị thông báo lỗi Backend: Trả JSON: {"status":"error","msg":"Vui lòng điền đầy đủ thông tin"} |
| MEMBER-ADD-03 | Thêm thành viên thất bại  - Username đã tồn tại | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Nhập username đã có 4. Click "Lưu" | username: "admin" (đã tồn tại) email: "new@gmail.com" password: "Pass@123" phone: "0912345678" | Frontend: Hiển thị "Người dùng đã tồn tại" Backend: Trả JSON: {"status":"error","msg":"Người dùng đã tồn tại"} |
| MEMBER-ADD-04 | Thêm thành viên thất bại  - Email đã tồn tại | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Nhập email đã có 4. Click "Lưu" | username: "newuser456" email: "existing@gmail.com" (đã có) password: "Pass@123" phone: "0912345678" | Frontend: Hiển thị "Email đã được sử dụng" Backend: Trả JSON: {"status":"error","msg":"Email đã được sử dụng"} |
| MEMBER-ADD-05 | Thêm thành viên thất bại  - Username có ký tự đặc biệt | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Nhập username chứa ký tự đặc biệt 4. Click "Lưu" | username: "user@123" email: "test@gmail.com" password: "Pass@123" phone: "0912345678" | Frontend: Hiển thị "Tên đăng nhập không gồm ký tự đặc biệt và dấu" Backend: Trả JSON: {"status":"error","msg":"Tên đăng nhập không gồm ký tự đặc biệt và dấu"} |
| MEMBER-ADD-06 | Thêm thành viên thất bại  - Username quá ngắn/dài | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Nhập username 4 ký tự 4. Click "Lưu" | username: "user" (4 ký tự) email: "test@gmail.com" password: "Pass@123" phone: "0912345678" | Frontend: Hiển thị "Tên đăng nhập phải tối thiểu 5 ký tự và tối đa 20 ký tự" Backend: Trả JSON: {"status":"error","msg":"Tên đăng nhập phải tối thiểu 5 ký tự và tối đa 20 ký tự"} |
| MEMBER-ADD-07 | Thêm thành viên thất bại  - Email không hợp lệ | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Nhập email không phải @[gmail.com](https://gmail.com/" \t "https://chat.deepseek.com/a/chat/s/_blank) 4. Click "Lưu" | username: "newuser123" email: "user@yahoo.com" password: "Pass@123" phone: "0912345678" | Frontend: Hiển thị "Email không hợp lệ" Backend: Trả JSON: {"status":"error","msg":"Email không hợp lệ"} |
| MEMBER-ADD-08 | Thêm thành viên với level=1 (admin) | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Chọn level=1 4. Click "Lưu" | username: "newadmin" email: "adminnew@gmail.com" password: "Pass@123" phone: "0912345679" level: 1 | Backend: Insert thành công với level=1 (quyền admin) |
| MEMBER-ADD-09 | Thêm thành viên với band=1 (đã khóa) | 1. Đăng nhập admin 2. Truy cập trang thêm thành viên 3. Chọn band=1 4. Click "Lưu" | username: "lockeduser" email: "locked@gmail.com" password: "Pass@123" phone: "0912345680" band: 1 | Backend: Insert thành công với band=1 (tài khoản bị khóa) |
| MEMBER-UPDATE-01 | Cập nhật thông tin thành viên thành công | 1. Đăng nhập admin 2. Truy cập danh sách thành viên 3. Click "Sửa" trên user A 4. Thay đổi fullname, phone 5. Click "Cập nhật" | id: 5 (tồn tại) fullname: "Nguyễn Văn B" phone: "0987654321" | Frontend: Hiển thị "Cập nhật thành công" Backend: Trả JSON success, update các trường tương ứng |
| MEMBER-UPDATE-02 | Cập nhật thất bại  - Thành viên không tồn tại | 1. Đăng nhập admin 2. Gửi request update với ID không tồn tại | id: 999 (không tồn tại) fullname: "Test User" | Frontend: Hiển thị "Thành viên không tồn tại" Backend: Trả JSON: {"status":"error","msg":"Thành viên không tồn tại"} |
| MEMBER-UPDATE-03 | Cập nhật thất bại  - Email mới đã tồn tại (của user khác) | 1. Đăng nhập admin 2. Sửa user A 3. Đổi email thành email của user B 4. Click "Cập nhật" | id: 5 (user A) email: "userB@gmail.com" (đã có của user B) | Frontend: Hiển thị "Email đã được sử dụng" Backend: Trả JSON: {"status":"error","msg":"Email đã được sử dụng"} |
| MEMBER-UPDATE-04 | Cập nhật thành công - Không thay đổi email (giữ nguyên) | 1. Đăng nhập admin 2. Sửa user A 3. Giữ nguyên email cũ 4. Click "Cập nhật" | id: 5 email: "userA@gmail.com" (chính là email hiện tại) | Backend: Trả JSON success, không báo lỗi trùng email |
| MEMBER-UPDATE-05 | Thay đổi level thành viên (user → admin) | 1. Đăng nhập admin 2. Sửa user level=0 3. Đổi level=1 4. Click "Cập nhật" | id: 5 level: 1 | Backend: Trả JSON success, update level=1, user này có quyền admin |
| MEMBER-UPDATE-06 | Thay đổi trạng thái band (khóa tài khoản) | 1. Đăng nhập admin 2. Sửa user band=0 3. Đổi band=1 4. Click "Cập nhật" | id: 5 band: 1 | Backend: Trả JSON success, update band=1, tài khoản bị khóa |
| MEMBER-UPDATE-07 | Cập nhật số dư (money) cho thành viên | 1. Đăng nhập admin 2. Sửa user 3. Thay đổi số dư 4. Click "Cập nhật" | id: 5 money: 500000 | Backend: Trả JSON success, update money=500000 |
| MEMBER-UPDATE-08 | Cập nhật veri\_email (xác thực email) | 1. Đăng nhập admin 2. Sửa user 3. Đổi veri\_email='on' 4. Click "Cập nhật" | id: 5 veri\_email: "on" | Backend: Trả JSON success, update veri\_email='on' |
| MEMBER-DELETE-01 | Xóa thành viên thành công | 1. Đăng nhập admin 2. Truy cập danh sách thành viên 3. Click "Xóa" trên user A 4. Xác nhận xóa | id: 10 (tồn tại) | Frontend: Hiển thị "Xóa thành viên thành công" Backend: Trả JSON success, xóa bản ghi với ID=10 |
| MEMBER-DELETE-02 | Xóa thất bại  - Thành viên không tồn tại | 1. Đăng nhập admin 2. Gửi request delete với ID không tồn tại | id: 999 (không tồn tại) | Frontend: Hiển thị "Thành viên không tồn tại" Backend: Trả JSON: {"status":"error","msg":"Thành viên không tồn tại"} |
| MEMBER-DELETE-03 | Xóa thất bại  - Không thể xóa chính mình (admin) | 1. Đăng nhập admin A 2. Cố gắng xóa chính tài khoản admin A | id: 1 (ID của admin đang đăng nhập) | Frontend: Hiển thị "Không thể xóa chính mình" Backend: Trả JSON: {"status":"error","msg":"Không thể xóa chính mình"} |
| MEMBER-DELETE-04 | Xóa thành viên đang có hosting/dịch vụ | 1. Đăng nhập admin 2. Chọn user đang có hosting 3. Click "Xóa" | id: 5 (có bản ghi trong history\_buy\_hosting) | Backend: Có thể lỗi khóa ngoại, cần xử lý cascade hoặc không cho xóa |

# KẾT LUẬN

Đề tài “Xây dựng website kinh doanh dịch vụ Hosting” được nhóm thực hiện với mục tiêu thiết kế và phát triển một hệ thống thương mại điện tử chuyên biệt, đáp ứng đầy đủ nhu cầu quản lý và kinh doanh các gói dịch vụ lưu trữ web trực tuyến. Trong bối cảnh chuyển đổi số mạnh mẽ và nhu cầu về không gian mạng cho các cá nhân, doanh nghiệp ngày càng tăng cao, việc xây dựng một nền tảng chuyên nghiệp, tự động hóa và thân thiện với người dùng là hết sức cần thiết.

Qua quá trình nghiên cứu, phân tích yêu cầu và thiết kế hệ thống, đề tài đã đạt được những kết quả cụ thể sau:

Thứ nhất, hệ thống đã được phân tích và thiết kế theo mô hình phân tầng rõ ràng với hai nhóm người dùng chính: Khách hàng (Customer) và Quản trị viên (Administrator). Đối với khách hàng, hệ thống cung cấp một hành trình trải nghiệm đầy đủ và liền mạch: từ đăng ký, đăng nhập, tìm hiểu và lựa chọn gói hosting, thực hiện thanh toán trực tuyến, đến việc quản lý dịch vụ đã mua, gửi yêu cầu hỗ trợ và theo dõi lịch sử hoạt động. Các chức năng này được xây dựng nhằm đảm bảo tính tiện lợi, minh bạch và chủ động cho người dùng cuối.

Thứ hai, đối với quản trị viên, đề tài đã xây dựng một bộ công cụ quản lý toàn diện và mạnh mẽ, bao gồm: quản lý thành viên, quản lý danh mục gói hosting, theo dõi và xử lý đơn hàng - thanh toán, tạo và kiểm soát mã giảm giá, đăng tải và quản lý bài viết, xử lý ticket hỗ trợ khách hàng, cùng hệ thống báo cáo thống kê chi tiết. Nhóm chức năng này không chỉ giúp vận hành hệ thống trơn tru mà còn cung cấp dữ liệu quan trọng để đưa ra các chiến lược kinh doanh hiệu quả.

Thứ ba, đề tài không chỉ dừng lại ở các chức năng cốt lõi mà còn tích hợp những yếu tố nâng cao trải nghiệm và hiệu quả hoạt động, như: cơ chế bảo mật tài khoản, hệ thống hỗ trợ khách hàng (kết hợp giữa con người và AI hỗ trợ), và các báo cáo trực quan hóa bằng biểu đồ. Điều này cho thấy hệ thống được thiết kế với tầm nhìn hướng tới một nền tảng thương mại điện tử hiện đại, có khả năng cạnh tranh và mở rộng trong tương lai.

Quá trình thực hiện đề tài là cơ hội quý báu để nhóm vận dụng tổng hợp kiến thức đã học về Phân tích thiết kế hệ thống, Công nghệ phần mềm, Cơ sở dữ liệu, Lập trình Web và Quản trị dự án vào thực tiễn. Nhóm đã học được cách làm việc theo quy trình, từ phân tích nghiệp vụ, thiết kế kiến trúc, cho đến lập kế hoạch triển khai chi tiết.

Tuy nhiên, trong khuôn khổ một đồ án học phần, hệ thống vẫn còn có những hạn chế nhất định cần được phát triển thêm trong tương lai, như: tích hợp thêm nhiều cổng thanh toán, phát triển ứng dụng di động đi kèm, tối ưu hóa thuật toán gợi ý sản phẩm, hoặc nâng cao khả năng tự động hóa triển khai và quản lý dịch vụ hosting (server).

Tóm lại, đề tài “Xây dựng website kinh doanh dịch vụ Hosting” đã hoàn thành các mục tiêu đề ra ban đầu, cung cấp một giải pháp phần mềm khả thi và có tính ứng dụng thực tiễn cao. Sản phẩm không chỉ là một website bán hàng thông thường mà là một hệ sinh thái dịch vụ hoàn chỉnh, đáp ứng nhu cầu của cả người mua và người bán trong lĩnh vực hosting. Kết quả của đề tài là nền tảng quan trọng, vừa là minh chứng cho quá trình học tập và nghiên cứu của nhóm sinh viên, vừa mở ra hướng phát triển để có thể trở thành một sản phẩm thực tế, góp phần vào sự phát triển chung của ngành công nghệ thông tin và thương mại điện tử tại Việt Nam.

# TÀI LIỆU THAM KHẢO

1. Amazon Web Services. (2023, March 15). Best practices for designing scalable

web hosting architectures. AWS Cloud Computing Blog. https://aws.amazon.com/blogs/architecture/

1. Fowler, M. (2018). Refactoring: Improving the design of existing code (2nd ed.).

Addison-Wesley.

1. Laravel. (2024). Laravel documentation (Version

11.x). https://laravel.com/docs/11.x

1. Le, H. D., & Tran, T. M. (2021). Thiết kế cơ sở dữ liệu hiện đại. Nhà Xuất Bản

Giao Thông Vận Tải.

1. Mai, V. L. (2025). Bài giảng môn Đồ án chuyên ngành CNPM [Tài liệu học tập].

Khoa Công nghệ Thông tin, Trường Đại học Công nghệ Đông Á.

1. Nguyen, T. H., & Le, V. A. (2020). Phát triển ứng dụng web với PHP và MySQL.

Nhà Xuất Bản Khoa Học và Kỹ Thuật.

1. Pham, T. L. (2023). Nghiên cứu và xây dựng hệ thống bán hosting trực tuyến [Đồ

án tốt nghiệp kỹ sư]. Trường Đại học Bách Khoa Hà Nội.

1. Sommerville, I. (2016). Software engineering (10th ed.). Pearson.
2. Stripe. (2024). Accept online payments. https://stripe.com/docs/payments
3. Tran, M. D., & Pham, Q. V. (2024). A hybrid approach for optimizing e-

commerce recommendation systems. Trong Kỷ yếu Hội nghị Quốc tế về Khoa học Máy tính và Kỹ thuật Phần mềm (ICCSE 2024) (tr. 112-119). Nhà Xuất Bản Giao Thông Vận Tải.

1. World Wide Web Consortium. (2023, December 14). HTML living

standard. https://html.spec.whatwg.org/

1. Nhóm 6, Trường Đại học Công nghệ - ĐHQGHN. (2024). Bài tập lớn môn

Công nghệ phần mềm: Hệ thống quản lý rạp chiếu phim.

1. Nhóm 6, Trường Đại học Công nghiệp Hà Nội. (2023). Báo cáo bài tập lớn

nhập môn công nghệ phần mềm: Nghiên cứu mô hình bản mẫu và ứng dụng phần mềm quản lý bán laptop.

1. Trần Thị Ngát, Trường Đại học Mở Hà Nội. (2022). Bài tập lớn Thiết kế

website kinh doanh sản phẩm bằng bạc.

1. Trường Đại học Kinh Doanh và Công Nghệ Hà Nội. (2025). Danh sách bài tập

lớn các đề tài môn Kỹ nghệ phần mềm.

1. Trường Đại học Kinh tế Quốc dân. (2022). Bài tập lớn Kỹ nghệ phần mềm: Xây dựng phần mềm quản lý sinh viên.
2. Trường Đại học Tân Trào. (2024). Bài tập lớn Công nghệ web: "Thiết kế

website shop quần áo".

1. Nguyễn Văn Long & Đỗ Thị Thanh Tuyền. (2020). Phát triển ứng dụng Web

với PHP và MySQL. Nhà Xuất Bản Khoa Học và Kỹ Thuật.

1. Sommerville, I. (2016). Software Engineering (10th ed.). Pearson.
2. Vũ Đình Minh. (2023). Bài giảng môn Đồ án chuyên ngành CNPM. Khoa Công

nghệ Thông tin, Trường Đại học Công nghệ Đông Á.

1. Amazon Web Services. (2023). Best practices for designing scalable web

hosting architectures. AWS Cloud Computing Blog. https://aws.amazon.com/blogs/architecture/

1. Laravel. (2024). Laravel documentation (Version

11.x). https://laravel.com/docs/11.x

1. Stripe. (2024). Accept online payments. https://stripe.com/docs/payments