**BỘ** **CÔNG** **THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**



**NGUYỄN TRANG NHUNG**

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

NGÀNH HỆ THỐNG THÔNG TIN

|  |
| --- |
| **NGHIÊN CỨU VÀ ỨNG DỤNG SELENIUM WEBDRIVER TRONG KIỂM THỬ WEBSITE ĐẶT ĐỒ ĂN TRỰC TUYẾN** |

**HỆ THỐNG THÔNG TIN**

|  |  |
| --- | --- |
| **CBHD** | **: ThS.Trần Việt Hà** |
| **Sinh viên** | **: Nguyễn Trang Nhung** |
| **Mã sinh viên** | **: 2021606577** |

**Hà Nội – Năm 2025**

|  |
| --- |
| **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
| ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC  NGÀNH HỆ THỐNG THÔNG TIN |
|  |

|  |
| --- |
| **NGHIÊN CỨU VÀ ỨNG DỤNG SELENIUM WEBDRIVER TRONG KIỂM THỬ WEBSITE ĐẶT ĐỒ ĂN TRỰC TUYẾN** |

|  |  |
| --- | --- |
| **CBHD** | **: ThS.Trần Việt Hà** |
| **Sinh viên** | **: Nguyễn Trang Nhung** |
| **Mã sinh viên** | **: 2021606577** |

|  |
| --- |
| **Hà Nội – Năm 2025** |

LỜI CẢM ƠN

Trong thời gian thực hiện đồ án tốt nghiệp ngành hệ thống thông tin, em đã nhận được nhiều sự hỗ trợ, giúp đỡ và khuyến khích từ rất nhiều người. Đó là một phần quan trọng để giúp em có thể thực hiện thành công đồ án tốt nghiệp của mình. Lời đầu tiên, em xin được gửi đến các thầy cô trường Công nghệ thông tin và Truyền thông thuộc trường Đại học Công nghiệp Hà Nội – những người đã truyền dạy rất nhiều kiến thức và kỹ năng cần thiết cho em trong suốt thời gian tham gia học tập và rèn luyện tại trường để em có thể thực hiện tốt đồ án tốt nghiệp cũng như trong nghề nghiệp tương lai. Đặc biệt, em xin được gửi lời cảm ơn đến thầy ThS.Trần Việt Hà là người hướng dẫn, đồng hành cùng em trong suốt quá trình thực hiện đồ án tốt nghiệp. Nhờ có sự chỉ bảo tận tình và giúp đỡ của thầy, em đã có thể xây dựng và hoàn thành đồ án tốt nghiệp của mình một cách tốt nhất.

Em cũng muốn gửi lời cảm ơn tới Công ty Cổ phần DevPro Việt Nam đã tạo cơ hội để em được học tập và thực hành trong lĩnh vực kiểm thử phần mềm, giúp em có nhiều thêm kiến thức, kỹ năng và định hướng phát triển cho nghề nghiệp sau này.

Trong quá trình thực hiện đồ án tốt nghiệp, em đã cố gắng hết mình để cố gắng đạt được kết quả tốt nhất, nhưng em vẫn rất mong nhận được thêm sự góp ý từ thầy cô và các bạn để tiếp tục hoàn thiện hơn trong tương lai.

Em xin chân thành cảm ơn!

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc198037328)

[MỤC LỤC ii](#_Toc198037329)

[DANH MỤC HÌNH VẼ iii](#_Toc198037330)

[DANH MỤC BẢNG BIỂU iv](#_Toc198037331)

[MỞ ĐẦU 1](#_Toc198037332)

[CHƯƠNG 1. TỔNG QUAN VỀ KIỂM THỬ PHẦN MỀM 4](#_Toc198037333)

[1.1 Khái niệm và mục tiêu của kiểm thử phần mềm 4](#_Toc198037334)

[1.2 Các nguyên lý cơ bản của kiểm thử 5](#_Toc198037335)

[1.3 Quy trình kiểm thử phần mềm 6](#_Toc198037336)

[1.4 Các mức độ và loại hình kiểm thử 9](#_Toc198037337)

[1.5 Các kỹ thuật kiểm thử 10](#_Toc198037338)

[1.6 Kiểm thử trên nền tảng web 17](#_Toc198037339)

[CHƯƠNG 2. CÔNG CỤ KIỂM THỬ SELENIUM WEBDRIVER 19](#_Toc198037340)

[2.1 Giới thiệu về công cụ Selenium WebDriver 19](#_Toc198037341)

[2.2 Công cụ kiểm thử tự động Selenium WebDriver 21](#_Toc198037342)

[2.3 Các công cụ và thư viện hỗ trợ Selenium WebDriver 25](#_Toc198037343)

[2.4 Các bước cài đặt Selenium WebDriver sử dụng ngôn ngữ Java 27](#_Toc198037344)

[CHƯƠNG 3. ỨNG DỤNG SELENIUM WEBDRIVER KIỂM THỬ ỨNG DỤNG WEB 38](#_Toc198037345)

[3.1 Giới thiệu về website đặt đồ ăn trực tuyến Haui Food 38](#_Toc198037346)

[3.2 Phân tích yêu cầu phần mềm 41](#_Toc198037347)

[3.3 Kế hoạch kiểm thử 69](#_Toc198037348)

[3.4 Thiết kế kịch bản kiểm thử 72](#_Toc198037349)

[3.5 Thực thi kiểm thử 99](#_Toc198037350)

[3.6 Báo cáo và đánh giá kết quả kiểm thử 101](#_Toc198037351)

[KẾT LUẬN 104](#_Toc198037352)

[TÀI LIỆU THAM KHẢO 105](#_Toc198037353)

# DANH MỤC HÌNH VẼ

[Hình 1.1: Quy trình kiểm thử phần mềm 7](#_Toc198037354)

[Hình 1.2: Các mức độ kiểm thử 9](#_Toc198037355)

[Hình 1.3: Phân vùng tương đương 12](#_Toc198037356)

[Hình 1.4: Phân tích giá trị biên 13](#_Toc198037357)

[Hình 1.5: Mô hình sơ đồ chuyển trạng thái 15](#_Toc198037358)

[Hình 2.1: Thành phần của Selenium 20](file:///C:\Users\trangnhung\NguyenTrangNhung\BaoCaoDATN_NguyenTrangNhung.docx#_Toc198037359)

[Hình 2.2: Màn hình trang tải về Selenium WebDriver 28](#_Toc198037360)

[Hình 2.3: Thao tác tạo một project mới trong IntelliJ 28](file:///C:\Users\trangnhung\NguyenTrangNhung\BaoCaoDATN_NguyenTrangNhung.docx#_Toc198037361)

[Hình 2.4: Chọn cấu hình JDK đúng với phiên bản Java ở IntelliJ 29](#_Toc198037362)

[Hình 2.5: Thao tác tải Selenium Java 30](#_Toc198037363)

[Hình 2.6: Chọn phiên bản cho Selenium Java 30](#_Toc198037364)

[Hình 2.7: Dependency để khai khai báo thư viện Selenium Java 31](#_Toc198037365)

[Hình 2.8: Thực hiện khởi động lại Maven sau khi thêm dependency 31](#_Toc198037366)

[Hình 2.9: Hoàn thiện cài đặt thư viện Selenium 32](file:///C:\Users\trangnhung\NguyenTrangNhung\BaoCaoDATN_NguyenTrangNhung.docx#_Toc198037367)

[Hình 2.10: Dependency để khai báo thư viện TestNG 33](#_Toc198037368)

[Hình 2.11: Hoàn thiện cài đặt TestNG 33](file:///C:\Users\trangnhung\NguyenTrangNhung\BaoCaoDATN_NguyenTrangNhung.docx#_Toc198037369)

[Hình 2.12: Kết quả của test script khi chạy bằng terminal 34](#_Toc198037370)

[Hình 2.13: Kiểm tra phiên bản Allure 34](#_Toc198037371)

[Hình 2.14: Cấu hình plugin Allure trong Maven 35](#_Toc198037372)

[Hình 2.15: Cấu hình Listener và các annotation của Allure 36](#_Toc198037373)

[Hình 2.16: Giao diện báo cáo Allure 37](#_Toc198037374)

[Hình 3.1: Màn hình Đăng ký 39](#_Toc198037375)

[Hình 3.2 Màn hình Đăng nhập 39](#_Toc198037376)

[Hình 3.3:Màn hình Trang chủ 39](#_Toc198037377)

[Hình 3.4: Màn hình Xem chi tiết sản phẩm 40](#_Toc198037378)

[Hình 3.5: Màn hình Đặt hàng 40](#_Toc198037379)

[Hình 3.6: Màn hình Trang cá nhân 40](#_Toc198037380)

[Hình 3.7: Biểu đồ Use case phía người dùng 41](#_Toc198037381)

[Hình 3.8: Biểu đồ Use case phía Quản trị viên 42](#_Toc198037382)

[Hình 3.9:Tổng quan kết quả kiểm thử website Haui Food 100](#_Toc198037383)

[Hình 3.10: Biểu đồ kết quả kiểm thử theo trạng thái, mức độ và thời gian 100](#_Toc198037384)

[Hình 3.11: Chi tiết về những test case thất bại 101](#_Toc198037385)

# DANH MỤC BẢNG BIỂU

[Bảng 1.1: Ví dụ về Bảng quyết dịnh 14](#_Toc198036964)

[Bảng 1.2: Ví dụ về Pairwise testing 15](#_Toc198036965)

[Bảng 3.1: Yêu cầu phi chức năng của website Haui Food 68](#_Toc198036966)

[Bảng 3.2: Kế hoạch thực hiện kiểm thử 70](#_Toc198036967)

[Bảng 3.3: Rủi ro và giải pháp khi thực hiện kiểm thử 71](#_Toc198036968)

[Bảng 3.4: Test case chức năng Đăng ký 72](#_Toc198036969)

[Bảng 3.5: Test case cho chức năng Đăng nhập 74](#_Toc198036970)

[Bảng 3.6: Test case cho chức năng Đăng xuất 76](#_Toc198036971)

[Bảng 3.7: Test case cho chức năng Tìm kiếm sản phẩm 77](#_Toc198036972)

[Bảng 3.8: Test case cho chức năng Xem chi tiết sản phẩm 78](#_Toc198036973)

[Bảng 3.9: Test case cho cụm chức năng Quản lý giỏ hàng 79](#_Toc198036974)

[Bảng 3.10: Test case cho chức năng Đặt hàng 82](#_Toc198036975)

[Bảng 3.11: Test case cho chức năng Hủy đơn hàng 85](#_Toc198036976)

[Bảng 3.12: Test case cho chức năng Xem danh sách đơn hàng 86](#_Toc198036977)

[Bảng 3.13: Test case cho chức năng Xem thông tin tài khoản 87](#_Toc198036978)

[Bảng 3.14: Test case cho chức năng Cập nhật thông tin cá nhân 88](#_Toc198036979)

[Bảng 3.15: Test case cho chức năng Đổi mật khẩu 90](#_Toc198036980)

[Bảng 3.16: Test case cho chức năng Quản lý đơn hàng 92](#_Toc198036981)

[Bảng 3.17: Test case cho chức năng Quản lý người dùng 96](#_Toc198036982)

1. MỞ ĐẦU
2. **Tên đề tài:**

Nghiên cứu và sử dụng Selenium Webdriver trong kiểm thử website đặt đồ ăn trực tuyến

1. **Lý do chọn đề tài**

Trong bối cảnh công nghệ phát triển mạnh mẽ, các website hay ứng dụng ngày càng trở nên phổ biến, đặc biệt là lĩnh vực thương mại điện tử. Sự phát triển nhanh chóng đó đặt ra yêu cầu cao về chất lượng phần mềm, đòi hỏi các hệ thống phải vận hành ổn định, chính xác và đáp ứng tốt nhu cầu người dùng.

Trong lĩnh vực phát triển phần mềm, kiểm thử đóng vai trò quan trọng không thể thiết trong việc đảm bảo chất lượng. Kiểm thử thủ công giúp đánh giá tính trực quan, trải nghiệm người dùng và xử lý những tình huống khóa tự động hóa nhưng sẽ dễ bỏ sót lỗi khi hệ thống ngày càng phức tạp. Vì thế cần kết hợp với kiểm thử tự động giúp tăng tốc độ kiểm thử, giảm thiểu sai sót khi thực hiện các bài kiểm thử lặp đi lặp lại và đảm bảo tính ổn định của hệ thống trên nhiều môi trường khác nhau.

Hiện nay, có rất nhiều công cụ phổ biến để kiểm thử tự động, tùy thuộc vào loại ứng dụng và mục đích kiểm thử. Trong đó, theo Stack Overflow Developer Survey 2023, Selenium là một trong những công cụ kiểm thử hàng đầu với hơn 50% lập trình viên, kiểm thử viên sử dụng trong lĩnh vực kiểm thử tự động.

Với mục tiêu tìm hiểu cách áp dụng kiểm thử tự động trong thực tế, em đã quyết định thực hiện đề tài “Nghiên cứu và sử dụng Selenium Webdriver trong kiểm thử website đặt đồ ăn trực tuyến”. Đề tài này giúp em nắm vững các kỹ thuật kiểm thử, cách thiết kế ca kiểm thử và kết hợp kiểm thử tự động với kiểm thử thủ công để tối ưu chất lượng phần mềm.

1. **Mục tiêu của đề tài**

* Hiểu được quy trình kiểm thử trong thực tế.
* Tìm hiểu và thực hành thiết kế ca kiểm thử và báo cáo kết quả nhằm đảm bảo chất lượng phần mềm.
* Tìm hiểu tính năng, cách cài đặt và sử dụng Selenium WebDriver kết hợp các công cụ và thư viện hỗ trợ khác trong kiểm thử tự động.
* Áp dụng các kỹ thuật kiểm thử và thực thi các kịch bản kiểm thử bằng Selenium WebDriver để kiểm thử chức năng cho trang web đặt đồ ăn trực tuyến Haui Food.

1. **Nội dung nghiên cứu**

Đề tài tập trung vào nghiên cứu và áp dụng các phương pháp kiểm thử phần mềm trong việc kiểm thử wbsite Haui Food với công cụ Selenium Webdriver. Các nội dung chính bao gồm:

* Lý thuyết kiểm thử phần mềm: Tìm hiểu quy trình kiểm thử phần mềm, các loại kiểm thử như kiểm thử chức năng, kiểm thử giao diện người dùng cũng như các kỹ thuật kiểm thử như kiểm thử hộp đen, kiểm thử hộp trắng.
* Phương pháp kiểm thử phần mềm tự động
* Công cụ hỗ trợ kiểm thử: Selenium Webdriver kết hợp với thư viện TestNG, Allure Report, Maven
* Ứng dụng: Vận dụng các kiến thức kiểm thử vào việc xây dựng và thực hiện kiểm thử cho website đặt đồ ăn trực tuyến Haui Food

1. **Giới hạn và phạm vi của đề tài**

Đề tài được thực hiện trong thời gian từ 17/03/2025 đến 17/05/2025.

Phạm vi tập trung vào kiểm thử website đặt đồ ăn trực tuyến do nhóm sinh viên thực hiện với các chức năng cơ bản với công cụ Selenium WebDriver trên nền tảng web.

1. **Phương pháp thực hiện**

* Phương pháp phân tích tài liệu: Tìm hiểu các kiến thức tổng quan về kiểm thử phần mềm, kiểm thử ứng dụng web, kiểm thử tự động. Tìm hiểu cụ thể về công cụ Selenium WebDriver cũng như việc sử dụng công cụ này trong kiểm thử phần mềm
* Phương pháp nghiên cứu định tính: Tìm hiểu các dự án thực tế đã sử dụng Selenium WebDriver để kiểm thử ứng dụng web, phân tích cách thức áp dụng và hiệu quả của phương pháp này trong kiểm thử tự động.
* Phương pháp nghiên cứu định lượng: Thực hiện kiểm thử tự động trên hệ thống đặt đồ ăn trực tuyến với Selenium WebDriver.
* Phương pháp phân tích số liệu: Sử dụng các phương pháp thống kể để xử lý dữ liệu kiểm thử, tổng hợp kết quả một cách trực quan và đánh giá tính hiệu quả từ đó đưa ra đề xuất và hướng phát triển sau này.

1. **Bố cục đề tài**

**Chương 1:**

**Chương 2:**

**Chương 3:**

# TỔNG QUAN VỀ KIỂM THỬ PHẦN MỀM

## Khái niệm và mục tiêu của kiểm thử phần mềm

Kiểm thử phần mềm được định nghĩa theo nhiều tiêu chuẩn khác nhau. Theo IEEE Standard 610.12-1990, kiểm thử phần mềm là “quá trình phân tích một sản phẩm phần mềm để phát hiện ra sự khác biệt giữa điều kiện hiện tại và điều kiện được yêu cầu (tức là lỗi) và đánh giá các tính năng của sản phẩm phần mềm đó.

Trong khi đó, theo ISTQB Glossary, kiểm thử phần mềm lại được định nghĩa như sau: kiểm thử phần mềm là một quá trình bao gồm tất cả các hoạt động trong vòng đời, bao gồm cả tĩnh và động, liên quan đến việc lập kế hoạch, chuẩn bị và đánh giá sản phẩm phần mềm cũng như các sản phẩm liên quan để xác định rằng chúng đáp ứng yêu cầu đã xác định, chứng mình rằng chúng phù hợp với mục đích sử dụng và phát hiện lỗi.

Từ đó có thể thấy, cả hai định nghĩa đều coi kiểm thử phần mềm là một quá trình quan trọng nhằm xác định lỗi và đảm bảo sản phẩm phần mềm đáp ứng chính xác, đầy đủ và đúng theo yêu cầu của khách hàng, yêu cầu sản phẩm đã đặt ra. Người thực hiện kiểm thử được gọi là kiểm thử viên (Tester). Dựa vào chiến lược kiểm thử thì kiểm thử phần mềm bao gồm:

* Kiểm thử thủ công: Là quá trình kiểm thử phần mềm mà kiểm thử viên thực hiện kiểm thử mọi thứ bằng tay, từ viết ca kiểm thử đến thực hiện test
* Kiểm thử tự động: Thực hiện một cách tự động các bước trong kịch bản kiểm thử bằng cách dùng một công cụ trợ giúp. Kiểm thử tự động nhằm tiết kiệm thời gian kiểm thử.

Mục đích chính của kiểm thử phần mềm bao gồm:

* Tìm ra được càng nhiều lỗi càng tốt trong điều kiện về thời gian đã định và nguồn lực sẵn có
* Chứng minh rằng sản phẩm phần mềm phù hợp với các đặc tả của nó
* Xác thực chất lượng kiểm thử phần mềm đã dùng chi phí và nỗ lực tối thiểu
* Thiết kế tài liệu kiểm thử một cách có hệ thống và thực hiện nó sao cho có hiệu quả, tiết kiệm được thời gian công sức

## Các nguyên lý cơ bản của kiểm thử

Trong quá trình phát triển phần mềm, kiểm thử đóng vai trò quan trọng nhằm đảm bảo chất lượng sản phẩm trước khi đưa vào sử dụng. Tuy nhiên, kiểm thử không thể phát hiện mọi lỗi và cũng không thể thực hiện một cách ngẫu nhiên mà cần có phương pháp tiếp cận khoa học. Để tối ưu hóa quá trình kiểm thử, ngành kiểm thử phần mềm đã xây dựng và tuân theo một số nguyên lý cơ bản giúp định hướng chiến lược kiểm thử, nâng cao hiệu quả và tối ưu nguồn lực. Dưới đây là 7 nguyên lý cơ bản của kiểm thử:

Thứ nhất, kiểm thử toàn bộ là không thể. Thay vì kiểm thử toàn bộ, chúng ta có thể dựa vào đánh giá rủi ro, độ ưu tiên và sử dụng kỹ thuật thiết kế kịch bản kiểm thử phù hợp để kiểm thử.

Thứ hai, sự phân cụm của lỗi (defect clustering). Nguyên lý này cho rằng một số ít các mô-đun chứa hầu hết các lỗi được phát hiện. Đây là việc áp dụng nguyên lý Pareto (80-20 Rule) để kiểm thử phần mềm: khoảng 80% các vẫn đề được tìm thấy trong 20% các mô-đun.

Thứ ba là nguyên lý thuốc trừ sâu. Theo nguyên lý này, một bộ dữ liệu kiểm thử được sử dụng lặp đi lặp lại sẽ không tìm được ra các lỗi mới. Để khắc phục điều này, các trường hợp kiểm thử cần phải được xem xét và sửa đổi thường xuyên, bổ sung thêm các trường hợp kiểm thử mới và khác để giúp tìm ra nhiều lỗi hơn. Người kiểm thử không chỉ đơn giản là phụ thuộc vào các kỹ thuật kiểm thử hiện có mà cần phải nhìn ra liên tục để cải thiện các phương pháp hiện có để làm cho kiểm thử hiệu quả hơn. Khi một thay đổi được thực hiện, việc kiểm thử hồi quy là rất cần thiết.

Thứ tư, kiểm thử chỉ ra sự hiện diện của các lỗi. Kiểm thử chỉ ra sự tồn tại của các lỗi nhưng không thể chứng minh rằng phần mềm không có lỗi. Kiểm thử làm giảm xác suất của các lỗi chưa được phát hiện trong phần mềm.

Thứ năm, sự sai lầm về việc phần mềm không có lỗi. Có thể phần mềm 99% không có lỗi vẫn không sử dụng được nếu hệ thống được xây dựng trên yêu cầu sai. Kiểm thử không chỉ là tìm ra các lỗi, mà còn để kiểm tra xem phần mềm có đáp ứng yêu cầu nghiệp vụ không.

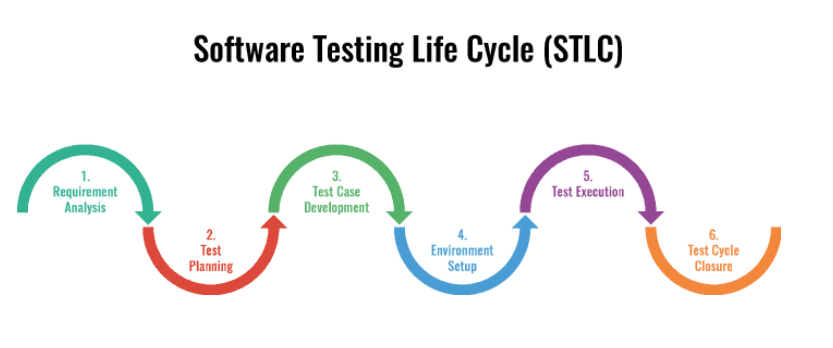
Thứ sáu, nên thực hiện kiểm thử càng sớm càng tốt. Kiểm thử nên bắt đầu càng sớm càng tốt trong chu kỳ vòng đời phát triển phần mềm. Lỗi được phát hiện sớm sẽ có chi phí sửa lỗi thấp hơn. Nhưng nên kiểm thử sớm như thế nào còn phụ thuộc vào việc sản phẩm phần mềm đó đang được phát triển theo quy trình như thế nào.

Thứ bảy, kiểm thử phụ thuộc vào ngữ cảnh. Cách tiếp cận, phương pháp, kỹ thuật và các loại kiểm thử tùy thuộc vào ứng dụng bơi tất cả các phần mềm đã phát triển đều không giống nhau. Ví dụ: Kiểm thử một trang web thương mại điện tử sẽ khác so với kiểm thử một trang web giới thiệu về một công ty.

## Quy trình kiểm thử phần mềm

Tùy vào từng tổ chức, hệ thống, ngữ cảnh, mức độ rủi ro của phần mềm mà quy trình kiểm thử phần mềm có thể gồm nhiều bước khác nhau. Nhưng nhìn chung mọi quy trình kiểm thử đều có những bước cơ bản sau:

* Phân tích yêu cầu (Requirement Analysis).
* Lập kế hoạch kiểm thử (Test Planning)
* Thiết kế kịch bản kiểm thử (Test Case Development)
* Thiết lập môi trường kiểm thử (Environment Setup)
* Thực hiện kiểm thử (Test Execution)
* Kết thúc chu kỳ kiểm thử **(**Test Cycle Closure)



Hình 1.1: Quy trình kiểm thử phần mềm

### Phân tích yêu cầu

Phân tích yêu cầu là giai đoạn đầu tiên trong quy trình kiểm thử phần mềm, gồm các hoạt động sau:

* + - * Tìm hiểu, nghiên cứu và phân tích cụ thể các yêu cầu trong tài liệu đặc tả của dự án hoặc tài liệu khách hàng.
      * Đưa ra câu hỏi với các bên liên quan như chuyên viên phân tích nghiệp vụ, người quản lý dự án, trưởng nhóm và khách hàng để hiểu chính xác hơn về yêu cầu của sản phẩm.

### Lập kế hoạch kiểm thử

Lập kế hoạch kiểm thử theo các bước quan trọng sau:

* + - * Xác định phạm vi, rủi ro và mục đích của hoạt động kiểm thử.
      * Xác định cách tiếp cận kiểm thử.
      * Xác định quy định kiểm thử hoặc chiến lược kiểm thử.
      * Xác định yêu cầu về nguồn nhân lực như con người, môi trường kiểm thử, thiết bị,...
      * Lên lịch trình cho việc phân tích kiểm thử và thiêt kế các trường hợp. kiểm thử, thực thi kiểm thử và đánh giá kết quả kiểm thử.
      * Xác định các tiêu chí kết thúc việc kiểm thử.

### Thiết kế kịch bản kiểm thử

Thiết kế kịch bản kiểm thử là một bước hết sức quan trọng giúp đảm bảo rằng hệ thống được kiểm tra một cách có hệ thống và đầy đủ. Một kịch bản kiểm thử được xây dựng tốt sẽ giúp xác định các tình huống cụ thể, bao gồm đầu vào, hành động, kết quả mong đợi và điều kiện tiên quyết. Thiết kế kịch bản kiểm thử gồm các hoạt động sau:

* + - * Kiểm tra lại tất cả các tài liệu để xác định công việc cần làm, các công việc có khác gì so với dự án trước khách hàng đưa cho.
      * Viết kịch bản kiểm thử (test case hoặc checklist).
      * Chuẩn bị dữ liệu kiểm thử.
      * Thành viên trong đội kiểm thử đánh giá lại test case/checklist.

### Thiết lập môi trường kiểm thử

Thiết lập môi trường kiểm thử nhằm đảm bảo rằng quá trình kiểm thử được thực hiện trong điều kiện tương đồng với môi trường thực tế, giúp phát hiện và khắc phục lỗi một cách chính xác. Việc cài đặt môi trường kiểm thử gồm các hoạt động sau:

* + - * Quyết định môi trường kiểm thử.
      * Chuẩn bị một số ca kiểm thử để kiểm tra xem môi trường cài đặt đã sẵn sàng cho việc kiểm thử hay chưa.

### Thực hiện kiểm thử

Thực hiện kiểm thử gồm các hoạt động sau:

* + - * Thực hiện các ca kiểm thử như thiết kế và mức độ ưu tiên đã đưa ra trên môi trường đã được cài đặt.
      * So sánh với kết quả mong đợi sau báo cáo các lỗi xảy ra lên công cụ quản lý lỗi.
      * Thực hiện kiểm thử lại và kiểm thử hồi quy để xác minh các lỗi đã được sửa.
      * Đo và phân tích tiến độ.
      * Báo cáo thường xuyên cho người quản lý dự án và khách hàng về tình hình thực hiện dự án.

### Đóng chu trình kiểm thử

Đây là giai đoạn cuối cùng trong quy trình kiểm thử phần mềm, gồm các hoạt động sau:

* + - * Tổng kết, báo cáo kết quả về việc thực thi test case.
      * Đánh giá các tiêu chí hoàn thành như phạm vi kiểm tra, chất lượng, chi phí, thời gian, mục tiêu kinh doanh quan trọng.
      * Thảo luận tất cả những điểm tốt, điểm chưa tốt và rút ra bài học kinh nghiệm.

## Các mức độ và loại hình kiểm thử

* Các mức độ kiểm thử:



Hình 1.2: Các mức độ kiểm thử

* Kiểm thử đơn vị (Unit testing): Nhằm kiểm tra đơn vị thiết kế nhỏ nhất – một module phần mềm. Người tiến hành kiểm thử đơn vị thường là lập trình viên. Kỹ thuật kiểm thử đơn vị chủ yếu là kiểm thử hộp trắng, trong các trường hợp cần thiết có thể sử dụng thêm kỹ thuật kiểm thử hộp đen.
* Kiểm thử tích hợp (Intergation testing): Nhằm xác minh rằng các module hoặc thành phần riêng lẻ của hệ thống có thể hoạt động cùng nhau một cách chính xác sau khi được tích hợp. Người thực hiện thường là nhóm phát triển phần mềm.
* Kiểm thử hệ thống (System test): Nhằm tìm kiếm các lỗi, nhưng trọng tâm là đánh giá về hoạt động, thao tác, sự tin cậy và các yêu cầu khác liên quan đến chất lượng của toàn hệ thống. Người thực hiện thường là kiểm thử viên. Kỹ thuật kiểm thử hệ thống chủ yếu là kiểm thử hộp đen.
* Kiểm thử chấp nhận (Acceptance test): Là cấp độ kiểm thử cuối cùng trong các cấp độ kiểm thử, được thực hiện để xác nhận rằng hệ thống đáp ứng đầy đủ các yêu cầu của khách hàng hoặc người dùng cuối trước khi đưa vào sử dụng chính thức.
* Các loại hình kiểm thử:
* Kiểm thử chức năng (Functional test): Quy trình cố gắng tìm ra các khác biệt giữa đặc tả bên ngoài của phần mềm và thực tế mà phần mềm cung cấp.
* Kiểm thử phi chức năng (Non-funtional test): Tập trung vào kiểm thử phần mề, hệ thống phần mềm có những đặc tính tốt như thế nào, tức là kiểm thử tập trung vào mặt chất lượng. Kiểm thử phi chức năng có thể được sử dụng ở mọi cấp độ kiểm thử nhưng thường được sử dụng hiệu quả nhất trong cấp độ kiểm thử hệ thống và kiểm thử chấp nhận.
* Kiểm thử cấu trúc (Structural test): Nhằm đánh giá sự bao phủ của code, tìm ra các lỗi tiềm ẩn trong cấu trúc logic của chương trình, xác minh xem mã nguồn có được viết đúng theo yêu cầu hay không. Kỹ thuật được sử dụng thường là kiểm thử hộp trắng và kiểm thử hộp xám.
* Kiểm thử liên quan đến sự thay đổi (Change related test): Kiểm thử xem sản phẩm có vận hành trơn tru qua các lần thay đổi hay không. Hai kỹ thuật sử dụng chủ yếu là kiểm thử lại và kiểm thử hồi quy.

## Các kỹ thuật kiểm thử

Trong kiểm thử phần mềm có 3 kỹ thuật chính bao gồm: Kiểm thử hộp đen, kiểm thử hộp trắng và kiểm thử hộp xám. Trong đề tài này, kỹ thuật được sử dụng là kiểm thử hộp đen.

### Kỹ thuật kiểm thử hộp đen

Kiểm thử hộp đen là một phương pháp kiểm thử phần mềm mà ở đó việc thực hiện kiểm tra các chức năng của một phần mềm mà không quan tâm cấu trúc bên trong, thiết kế hoặc mã hóa của nó, tập trung vào kiểm tra hệ thống làm gì chứ không kiểm tra hệ thống làm như thế nào. Nó còn được gọi là kiểm thử hướng dữ liệu hay là kiểm thử hướng vào/ra. Thường được sử dụng trong cấp độ kiểm thử hệ thống, kiểm thử chấp nhận người dùng và loại hình kiểm thử chức năng.

Ưu điểm: Kiểm thử hộp đen dễ tiếp cận, áp dụng được cho nhiều mức độ kiểm thử và có thể thực hiện mà không cần biết mã nguồn hoặc cấu trúc nội bộ của phần mềm.

Nhược điểm: Khó bao quát hết tất cả các trường hợp kiểm thử và không tối ưu cho kiểm thử hiệu năng hoặc bảo mật.

Các phương pháp kiểm thử hộp đen được sử dụng bao gồm:

* ***Phân vùng tương đương****:* là một kỹ thuật kiểm thử phần mềm có liên quan đến phân chia các giá trị đầu vào thành các phân vùng hợp lệ và không hợp lệ, sau đó chúng ta sẽ viết ra các kịch bản kiểm thử cho từng phần, chọn giá trị đại diện từ mỗi phân vùng làm dữ liệu thử nghiệm.

Phân tích cách làm gồm 2 bước:

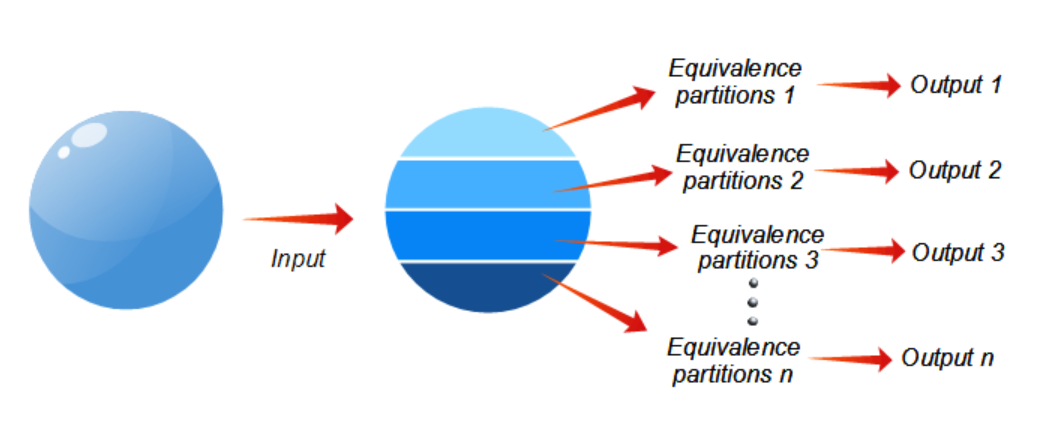
Bước 1: Xác định các lớp tương đương theo đầu vào hoặc đầu ra. Với mỗi điều kiện đầu vào hoặc đầu ra được mô tả trong đặc tả yêu cầu thì lấy ra: lớp thỏa mãn điều kiện và lớp không thỏa mãn điều kiện.

Bước 2: Lựa chọn giá trị đại diện

Ví dụ 1: Viết test case cho form Update giỏ hàng

Điều kiện ràng buộc: Trường Qty (Quantity) chỉ cho phép nhập số nguyên từ 1 đến 99. Trong đó từ 1-10 dành cho khách hàng mua lẻ (không được chiết khấu), từ 11-99 dành cho khách hàng mua sỉ (có chiết khấu).

* Ta có thể chia thành 5 phân vùng như sau khi kiểm thử:
* < 1
* 1 -10
* 11-99
* > 99
* Nhập vào không phải số nguyên



Hình 1.3: Phân vùng tương đương

* ***Phân tích giá trị biên***: là một kỹ thuật kiểm thử phần mềm có liên quan đến việc xác định biên (ranh giới) của điều kiện mô tả cho các giá trị đầu vào và chọn giá trị ở biên và bên cạnh giá trị biên làm dữ liệu kiểm thử.

Phân tích cách làm gồm 2 bước:

Bước 1: Xác định các điểm biên [a, b]

Bước 2: Thiết kế test case dựa vào các điểm biên

Test Case 1: giá trị a – 1 => kết quả mong đợi: Không hợp lệ

Test Case 2: giá trị a => Kết quả mong đợi: Hợp lệ

Test Case 3: giá trị b => Kết quả mong đợi: Hợp lệ

Test Case 4: giá trị b + 1 => Kết quả mong đợi: Không hợp lệ

Ví dụ 2: Phân tích giá trị biên cho điểm nằm trong khoảng [0,100]:

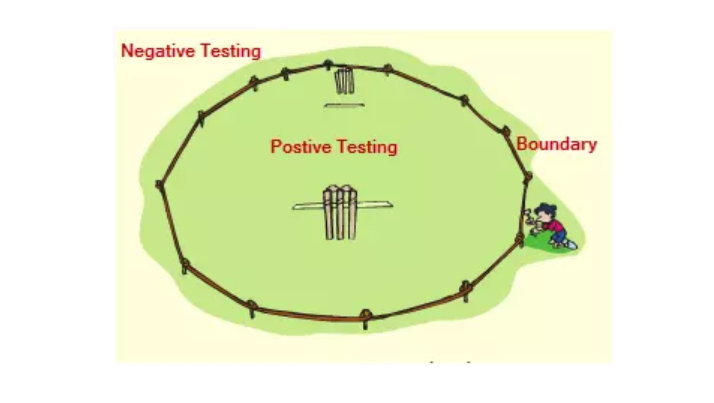
+ Giá trị nhỏ nhất: 0

+ Giá trị lớn nhất: 100

+ Giá trị nhỏ hơn giá trị nhỏ nhất: -1

+ Giá trị lớn hơn giá trị lớn nhất: 101

+ Giá trị trung bình: 50



Hình 1.4: Phân tích giá trị biên

* ***Bảng quyết định****:* Là một kỹ thuật hộp đen được sử dụng để kiểm tra các hành vi hệ thống với các cách kết hợp input đầu vào khác nhau. Đây là một cách tiếp cận có hệ thống, kết quả của các kết hợp và hành vi hệ thống tương ứng của chúng (output) sẽ được ghi lại dưới dạng bảng (Cause-Effect table). Đặc điểm của bảng là giá trị của các conditions chỉ có 2 giá trị True hoặc False và các cột (Rules) tương ứng với số lượng test case. Phương pháp này thường được áp dụng với các chức năng cần test có logic phức tạp và sự kết hợp nhiều điều kiện.

Phân tích cách làm gồm 4 bước:

Bước 1: Xác định các conditions (điều kiện đầu vào/input)

Bước 2: Tính toán số rules = 2^condition

Bước 3: Đặt toàn bộ các kết hợp vào bảng

Bước 4: Giảm số lượng các kết hợp và quyết định test case

Các nguyên tắc xác định condition:

Nguyên tắc 1: Cần xác định mục đích cuối cùng mà bài toán hướng tới là gì? Những điều kiện ảnh hưởng đến mục đích cuối cùng của bài toán thì chính là conditions.

Nguyên tắc 2: Không lấy chiều ngược lại của 1 condition đã chọn.

Nguyên tắc 3: Chỉ lấy các condition đơn lẻ, không được gộp các condition với nhau.

Ví dụ 3: Xác định test case cho bài toán mua vé tàu có được giảm giá hay không với các điều kiện sau: Nếu bạn có thẻ đường sắt "over 60s" thì được giảm giá 34% trên tất cả các vé bạn mua. Nếu bạn có thẻ "family rail card”, nếu bạn đi cùng với trẻ em (dưới 16 tuổi), thì bạn sẽ được giảm 50%, ngược lại bạn sẽ được giảm 10%. Bạn chỉ được sử dụng 1 loại thẻ đường sắt

Bảng quyết định của bài toán trên được thể hiện như sau:

Bảng 1.1: Ví dụ về Bảng quyết dịnh

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Điều kiện | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 |
| Thẻ “over 60s” | T | T | T | T | F | F | F | F |
| Thẻ “family rail” | T | T | F | F | F | F | T | T |
| Đi cùng trẻ em | T | F | T | F | T | F | T | F |
| **Giảm giá?** | **50%** | **34%** | **34%** | **34%** | **0** | **0** | **50%** | **10%** |

* ***Pairwise testing:*** Là một kỹ thuật kiểm thử hộp đen được sử dụng để kiểm tra các sự kết hợp của các cặp điều kiện có thể từ các tham số liên quan. Thường được áp dụng cho các chức năng cần test có logic phức tạp có sự kết hợp của quá nhiều các tổ hợp điều kiện.

Phân tích cách làm gồm 5 bước:

Bước 1: Liệt kê toàn bộ điều kiện đầu vào/Input

Bước 2: Xác định số lượng test case thủ công = số lượng vùng giá trị lớn nhất của các biến \* số lượng vùng giá trị lớn thứ 2 trong số các biến

Bước 3: Thực hiện kết hợp 2 giá trị đầu vào lớn nhất bảng

Bước 4: Thực hiện mapping các giá trị cho đến khi bảng được điền hết

Bước 5: Tối ưu hóa các cặp điều kiện

*Chú ý:* Thứ tự các điều kiện sắp xếp giảm dần theo số lượng vùng giá trị. Số lần lặp lại các giá trị của conditions lớn nhất sẽ phụ thuộc vào số lượng vùng giá trị của điều kiện lớn thứ hai.

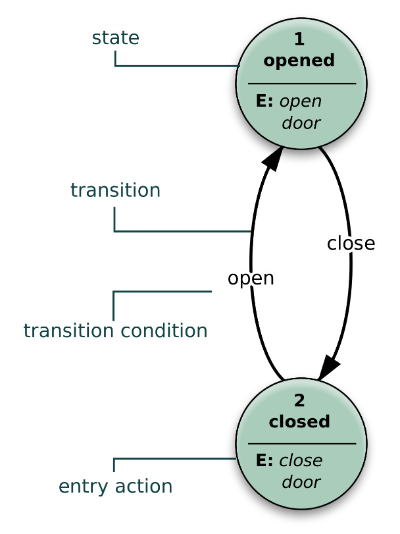
Ví dụ 4: Một ứng dụng gồm có: Listbox đơn giản với 10 phần tử (giả sử 0,1,2,3,4,5,6,7,8,9); Checkbox; Radio Nút; Textbox có thể nhận những giá trị nguyên dương từ 1 đến 100 và OK nút

Áp dụng pairwise testing và phân vùng tương đương ta có:

Bảng 1.2: Ví dụ về Pairwise testing

|  |  |  |  |
| --- | --- | --- | --- |
| **Texbox** | **List box** | **Checkbox** | **Radio Nút** |
| Int Valid | 0 | Checked | On |
| Int Valid | Other | Unchecked | Off |
| Int Invalid | 0 | Unchecked | On |
| Int Invalid | Other | Checked | Off |
| AlphaSpecialCharacter | 0 | Checked | Off |
| AlphaSpecialCharacter | Other | Unchecked | On |

* ***Sơ đồ chuyển đổi trạng thái:*** Là một kỹ thuật giúp thiết kế test case dựa vào mô hình (model) hoặc lược đồ (diagram) mô tả các trạng thái (state) và chuyển đổi (transition) của chúng. Thường áp dụng đối với các trường hợp có sự chuyển đổi trạng thái của các flow (thường sử dụng đối với việc kiểm thử End-To-End). Mô hình một sơ đồ chuyển trạng thái gồm 4 phần cơ bản:
* Các trạng thái hữu hạn
* Các sự dịch chuyển trạng thái
* Các sự kiện kích hoạt sự dịch chuyển trạng thái
* Các hành động là kết quả của sự dịch chuyển



Hình 1.5: Mô hình sơ đồ chuyển trạng thái

### Kỹ thuật kiểm thử hộp trắng

Kiểm thử hộp trắng là một phương pháp kiểm thử phần mềm dựa trên việc phân tích cấu trúc bên trong, thiết kế và mã nguồn để làm rõ đầu vào, đầu ra của phần mềm. Chính vì thế nên kỹ thuật kiểm thử hộp trắng thường được thực hiện bởi các lập trình viên và được áp dụng chủ yếu cho mức độ kiểm thử đơn vị và kiểm thử tích hợp. Kỹ thuật kiểm thử hộp trắng thuộc loại hình kiểm thử cấu trúc.

Ưu điểm: Có thể cô lập được lỗi dẫn đến việc sửa lỗi dễ dàng hơn và có thể thực hiện kiểm thử sớm hơn.

Nhược điểm: Không phù hợp với tất cả mọi người, không phải kiểm thử viên nào cũng có thể thực hiện được.

### Kỹ thuật kiểm thử hộp xám

Kiểm thử hộp xám là phương pháp kiểm thử khá mới mẻ, mới hình thành và đòi hỏi trình độ cao. Kiểm thử hộp xám là sự kết hợp của 2 loại kiểm thử hộp đen và hộp trắng. Với kỹ thuật này, người kiểm thử cần phải có một phần sự hiểu biết về cấu trúc của hệ thống cũng như là quyền truy cập vào cơ sở dữ liệu. Kiểm thử hộp xám thường được sử dụng cho mức độ kiểm thử tích hợp. Trong quá trình tiến hành kiểm thử hộp xám, người kiểm thử sẽ tạo đầu vào từ phía front-end, sau đó xác minh dữ liệu từ phía back-end.

Ưu điểm: Tiếp cận từ góc độ người dùng, giúp đảm bảo tính toàn diện của kiểm thử. Dựa trên các đặc tả chức năng, mô tả người dùng và sơ đồ kiến trúc, giúp xác nhận yêu cầu sớm. Kết hợp lợi ích của cả kiểm thử hộp đen và hộp trắng.

Nhược điểm: Có thể tốn nhiều thời gian để kiểm tra từng luồng, đôi khi không thực tế. Phạm vi kiểm tra thường thấp hơn so với kiểm thử hộp đen và hộp trắng riêng biệt.

## Kiểm thử trên nền tảng web

### Khái quát về kiểm thử trên nền tảng web

Trong bối cảnh kinh doanh hiện đại, việc triển khai các ứng dụng web là yếu tố quan trọng giúp doanh nghiệp mở rộng phạm vi tiếp cận khách hàng và tối ưu hóa cơ hội phát triển. Các ứng dụng web đóng vai trò then chốt trong việc kết nối, trao đổi thông tin và hỗ trợ hoạt động của nhiều tổ chức.

Để đạt được hiệu quả cao, ứng dụng web cần đảm bảo chất lượng, hiệu suất tối ưu cũng như mang lại trải nghiệm người dùng tốt. Không giống như các ứng dụng di động hay desktop, ứng dụng web phải hoạt động trên nhiều nền tảng khác nhau, đòi hỏi sự linh hoạt trong cấu hình và thao tác sử dụng. Điều này đặt ra nhiều thách thức đối với các nhà phát triển trong việc đảm bảo tính ổn định và tương thích của sản phẩm.

Do đó, việc xây dựng một chiến lược kiểm thử toàn diện là điều cần thiết để giảm thiểu rủi ro và nâng cao chất lượng ứng dụng web, giúp sản phẩm hoạt động ổn định trên nhiều môi trường khác nhau.

### Các hạng mục kiểm thử Website

Kiểm thử chức năng: Kiểm tra tính đúng đắn và hiệu quả của các chức năng trong ứng dụng web, bao gồm các liên kết, form nhập liệu, tìm kiếm và các tính năng khác. Mục tiêu là phát hiện và báo cáo về bất kỳ lỗi nào xuất hiện khi sử dụng các chức năng này.

Kiểm thử giao diện: So sánh thiết kế UI với thực tế trên trình duyệt, kiểm tra mỗi thành phần trên trang web để đảm bảo đúng với bản thiết kế. Kiểm tra các liên kết và menu, đảm bảo chúng hoạt động đúng và không trỏ về chính nó.

Kiểm thử cookie và session: Kiểm tra tính an toàn của việc đăng nhập bằng cách thử các tập tin cookie khác nhau và kiểm tra khả năng bảo mật của ứng dụng khi xóa các tập tin cookie.

Kiểm thử nội dung đa ngôn ngữ: Đảm bảo dữ liệu được dịch sang các ngôn ngữ khác nhau mà không mất đi ý nghĩa hoặc gây ra lỗi.

Kiểm thử cơ sở dữ liệu: Kiểm tra kết nối và truy vấn cơ sở dữ liệu để đảm bảo dữ liệu chính xác và các chức năng hoạt động đúng.

### Một số công cụ hỗ trợ kiểm thử website

* Một số công cụ kiểm thử chức năng:
  + BrowserStack: Kiểm thử chức năng của ứng dụng web trên nhiều trình duyệt khác nhau và kiểm tra khả năng hiển thị responsive.
  + Ranorex: Công cụ tự động kiểm thử cho ứng dụng web, desktop và di động.
  + Selenium: Công cụ mạnh mẽ cho kiểm thử tự động ứng dụng web, được phát triển và hỗ trợ bởi Selenium team từ Google.
* Một số công cụ kiểm thử hiệu năng:
  + WebLoad: Cho phép kiểm tra khả năng chịu tải và độ chịu lỗi của ứng dụng web bằng nhiều công nghệ khác nhau.
  + Apache JMeter: Công cụ mã nguồn mở để kiểm tra hiệu năng và chức năng của ứng dụng web.
  + NeoLoad: Đo và phân tích hiệu suất của ứng dụng web bằng cách tăng lưu lượng truy cập.
* Một số công cụ kiểm thử bảo mật:
  + Burp Suite: Công cụ kiểm tra lỗ hổng bảo mật cho ứng dụng web với nhiều tính năng tích hợp như Spider và Intruder.
  + OWASP Zed Attack Proxy: Đánh giá an ninh mạng và bảo mật của các ứng dụng web.
  + Nikto: Máy quét lỗ hổng máy chủ web mã nguồn mở để phát hiện cài đặt và cấu hình lỗi.

# CÔNG CỤ KIỂM THỬ SELENIUM WEBDRIVER

## Giới thiệu về công cụ Selenium WebDriver

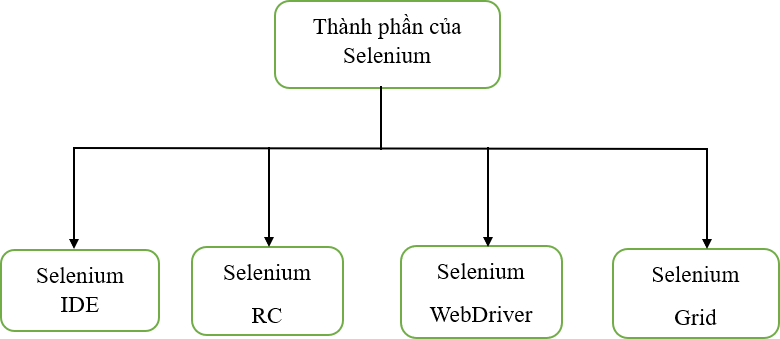
### Giới thiệu chung về Selenium

Selenium (thường được viết tắt là SE) là một bộ kiểm thử tự động mã nguồn mở được phát triển bởi Jason Huggins, sau đó tiếp tục phát triển bởi nhóm ThoughtWorks vào năm 2004. Selenium là công cụ miễn phí cho các ứng dụng web trên các trình duyệt và nền tảng khác nhau.

Selenium hỗ trợ kiểm tra hầu hết trên các trình duyệt phổ biến hiện nay như Chrome, Internet Explorer, Firefox, Safari, v.v cũng như các hệ điều hành phổ biến như Windows, MacOS, Linux. Selenium hỗ trợ một số lớn các ngôn ngữ lập trình như Java, C#, Perl, PHP, Python, Ruby, v.v giúp tăng linh hoạt trong quá trình kiểm thử. Không những vậy, Selenium có thể kết hợp thêm với một số công cụ khác để nâng cao hiệu quả kiểm thử tự động như:

* TestNG, Junit: Hỗ trợ quản lý và thực thi các kịch bản kiểm thử.
* Maven, Gradle: Quản lý thư viện và tự động hóa quy trình build dự án.
* Allure Report, Extent Reports, Cucumber Report: Tạo báo cáo kiểm thử chi tiết và trực quan.
* Jenkins, GitHub Actions: Tích hợp CI/CD để tự động chạy kiểm thử khi có thay đổi mã nguồn.
* Appium: Hỗ trợ kiểm thử tự động ứng dụng di động sử dụng cùng cú pháp với Selenium.
* Docker: Chạy kiểm thử song song trên nhiều trình duyệt và môi trường khác nhau.
* BrowserStack, Sauce Labs: Kiểm thử trên đám mây, hỗ trợ nhiều trình duyệt và thiết bị.

### Thành phần của Selenium

Selenium không chỉ là một công cụ duy nhất mà nó là một bộ phần mềm, mỗi bộ nó cung cấ các nhu cầu thử nghiệm khác nhau của một tổ chức. Selenium gồm có 4 thành phần sau:

Hình 2.1: Thành phần của Selenium

* Selenium Intergrated Development Environment (Selenium IDE): là framework đơn giản nhất trong bộ Selenium. Nó là một browser plugin có thể cài đặt dễ dàng như các plugin khác. Đây là một công cụ cho phép ghi (record) và phát lại (playback) một test script. Selenium IDE cho phép xuất ra kịch bản đã thu dưới nhiều loại ngôn ngữ lập trình khác nhau như Java, PHP, C#, Ruby, Perl hay Python.
* Selenium Remote Control (Selenium RC): là một framework kiểm thử cho phép thực hiện nhiều hơn và tuyến tính các hành động trên trình duyệt. Nó cho phép các nhà phát triển tự động hóa kiểm thử sử dụng một ngôn ngữ lập trình cho tính linh hoạt tối đa và mở rộng trong việc phát triển logic thử nghiệm.
* Selenium WebDriver: là sự kế thừa từ Selenium RC, nó thực hiện một cách tiếp cận hiện đại và ổn định hơn trong việc tự động hóa các hành động của trình duyệt. WebDriver không giống như Selenium RC, không dựa vào JavaScript cho tự động hóa. Nó kiểm soát trình duyệt bằng cách giao tiếp trực tiếp với nó.
* Selenium Grid: là một hệ thống hỗ trợ người dùng thực thi test script trên nhiều trình duyệt một cách song song mà không cần phải chỉnh sửa test script. Thực hiện phương pháp kiểm tra phân bổ, phối hợp nhiều Selenium RC để có thể thực thi trên nhiều trình duyệt Web khác nhau trong cùng một lúc nhằm giảm thiểu thời gian thực hiện.

## Công cụ kiểm thử tự động Selenium WebDriver

### Tổng quan về Selenium WebDriver

Selenium WebDriver là một trong những thành phần quan trọng và được sử dụng rộng rãi nhất trong bộ công cụ Selenium. Đây là một công cụ kiểm thử tự động mạnh mẽ, cho phép tương tác trực tiếp với trình duyệt web mà không cần thông qua bất kỳ thành phần trung gian nào. WebDriver được phát triển nhằm thay thế Selenium RC, khắc phục những hạn chế của phiên bản trước và mang lại hiệu suất kiểm thử cao hơn.

Không giống như Selenium IDE, vốn chỉ hoạt động trên một số trình duyệt nhất định, Selenium WebDriver hỗ trợ kiểm thử trên nhiều trình duyệt và nền tảng hiện đại như Chrome, Firefox, Edge, Safari và Opera. Đồng thời, WebDriver không yêu cầu phải khởi động máy chủ Selenium trước khi thực thi các tập lệnh kiểm thử như Selenium RC, giúp đơn giản hóa quá trình kiểm thử và nâng cao tốc độ thực thi.

Với khả năng mô phỏng hành vi người dùng một cách chân thực, Selenium WebDriver cho phép tự động hóa các thao tác như nhập liệu, nhấp chuột, cuộn trang, xác minh nội dung trang web và thực hiện các kịch bản kiểm thử phức tạp. Nhờ đó, WebDriver trở thành công cụ lý tưởng cho việc kiểm thử chức năng, kiểm thử giao diện và kiểm thử hồi quy trên các ứng dụng web.

### Kiến trúc của Selenium WebDriver

Selenium WebDriver là một công cụ kiểm thử tự động mạnh mẽ cho các ứng dụng web, hoạt động theo mô hình client – server. Kiến trúc của nó được thiết kế để cung cấp khả năng kiểm soát trực tiếp và hiệu quả đối với trình duyệt web.

Selenium WebDriver hoạt động bằng cách giao tiếp trực tiếp với trình duyệt thông qua trình điều khiển trình duyệt (browser driver). Trình điều khiển trình duyệt là một chương trình riêng biệt hoạt động như một cầu nối giữa Selenium WebDriver và trình duyệt web. Mỗi trình duyệt (Chrome, Firefox, Safari, v.v.) đều có trình điều khiển trình duyệt riêng. Các thành chính trong kiến trúc Selenium WebDriver bao gồm:

* Thư viện client của Selenium (Selenium Client Libraries): Đây là các thư viện ngôn ngữ lập trình (Java, C#, Python, v.v.) cung cấp các API để viết các kịch bản kiểm thử. Các thư viện này cho phép người dùng tương tác với trình duyệt, thực hiện các hành động như nhấp chuột, nhập liệu, v.v.
* WebDriver Language Bindings: Là các thư viện hỗ trợ đa ngôn ngữ lập trình cho phép người dùng có thể viết kịch bản kiểm thử bằng nhiều ngôn ngữ khác nhau.
* Trình điều khiển trình duyệt (Browser Driver): Đây là các trình điều khiển cụ thể cho từng trình duyệt, giúp Selenium WebDriver giao tiếp với trình duyệt. Ví dụ ChromeDriver cho Chrome, GeckoDriver cho Firefox, v.v.
* Trình duyệt (Browser): Selenium WebDriver hỗ trợ nhiều trình duyệt web phổ biến, cho phép kiểm thử trên nhiều nền tảng khác nhau.

Cách thức hoạt động của Selenium WebDriver như sau: Khi một kịch bản kiểm thử được thực thi, các lệnh Selenium được gửi đến trình điều khiển trình duyệt. Trình điều khiển trình duyệt diễn giải các lệnh này và giao tiếp với trình duyệt tương ứng. Trình duyệt thực hiện các hành động được chỉ định và trả kết quả về cho trình điều khiển trình duyệt. Trình điều khiển trình duyệt sau đó chuyển kết quả trở lại kịch bản kiểm thử.

### Các tính năng của Selenium WebDriver

* **Hỗ trợ đa trình duyệt:** Selenium WebDriver được thiết kế để hỗ trợ kiểm thử trên nhiều trình duyệt web phổ biến như Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Edge và cả trình duyệt Internet Explorer (dù Selenium đã chuyển sang sử dụng Edge Chromium từ phiên bản 3.141.59). Điều này đảm bảo rằng cách kịch bản kiểm thử có thể chạy trên cách trình duyệt khác nhau một cách đồng nhất và đáng tin cậy, giúp đảm bảo tính tương thích của ứng dụng web trên các nền tảng khác nhau.
* **Hỗ trợ đa ngôn ngữ lập trình:** Selenium WebDrver cung cấp khả năng viết kịch bản kiểm thử bằng một loạt các ngôn ngữ lập trình phổ biến. Các ngôn ngữ lập trình được Selenium WebDriver hỗ trợ: Java, C#, Python, PHP, Pearl, Ruby, v.v. Điều này tạo điều kiện linh hoạt cho các nhà phát triển và kiểm thử viên lựa chọn ngôn ngữ phù hợp với nhu cầu cụ thể của dự án hoặc tổ chức.
* **Tương tác trực tiếp với trình duyệt:** WebDriver giao tiếp trực tiếp với trình duyệt web, cho phép người dùng mô phỏng các hành động như nhấp chuột, nhập liệu, cuộn trang và gửi biểu mẫu.
* **Khả năng định vị phần tử linh hoạt:** Selenium WebDriver cung cấp nhiều phương pháp để định vị các phần tử web, bao gồm ID, tên, lớp, thẻ, CSS selector và Xpath. Điều này giúp ta dễ dàng tìm và tương tác với các phần tử trên trang web.
* **Kiểm tra tính tương tác của người dùng:** Công cụ cho phép ta kiểm tra tính tương tác của người dùng với các phần tử trên trang web, chẳng hạn như kiểm tra xem một nút có nhấp vào hay không hoặc một trường văn bản có được nhập liệu chính xác hay không.
* **Chụp ảnh màn hình:** Selenium WebDriver có thể chụp ảnh màn hình của trang web, giúp người kiểm thử ghi lại trạng thái của ứng dụng web trong quá trình kiểm tra.
* **Tích hợp với Selenium Grid:** Selenium WebDriver có thể được tích hợp với Selenium Grid, cho phép chạy các kiểm tra song song trên nhiều máy và trình duyệt khác nhau. Như thế sẽ giúp tăng tốc độ kiểm thử và giảm thời gian thực hiện.
* **Hỗ trợ kiểm thử trên thiết bị di động:** Công cụ có thể dùng để kiểm thử trên các trình duyệt di động bằng cách sử dụng Appium, một thư viện mã nguồn mở cho phép Selenium tương tác với các ứng dụng di động.

Tóm lại, Selenium WebDriver là một công cụ kiểm thử tự động mạnh mẽ và linh hoạt, cung cấp nhiều tính năng để kiểm tra các ứng dụng web trên nhiều trình duyệt và nền tảng khác nhau.

### Ưu điểm và nhược điểm của Selenium WebDriver

**Ưu điểm**

* + Mã nguồn mở: Selenium WebDriver là một phần mềm miền phí, dễ sử dụng và mang lại lợi ích trong việc kiểm thử phần mềm tự động.
* Tương thích nhiều trình duyệt: Selenium WebDriver tương thích với gần như tất cả các trình duyệt web phổ biến hiện nay, đảm bảo khả năng kiểm thử đa nền tảng.
* Hỗ trợ đa ngôn ngữ lập trình: Selenium WebDriver hỗ trợ gần hết các ngôn ngữ lập trình phổ biến hiện nay (C#, Java, Python, Ruby, Perl, v.v).
* Thân thiện, dễ sử dụng: Xét về độ thân thiện với người dùng, Selenium WebDriver vượt trội so với các thành phần khác của Selenium. Những câu lệnh của nó khá dễ nhớ và tiện dụng. Selenium cũng cung cấp cho người dùng nhiều tài liệu hướng dẫn cần thiết.
* Tốc độ thực thi nhanh: Selenium WebDriver tận dụng khá tốt khả năng hỗ trợ tự động hóa của các trình duyệt web. Tốc độ thực thi của Selenium WebDriver sẽ nhanh hơn nhiều nếu so với các công cụ khác trong bộ Selenium.

**Nhược điểm**

* Chỉ hỗ trợ ứng dụng web: Selenium WebDriver chỉ hỗ trợ kiểm thử trên các ứng dụng web, hạn chế sự đa dạng trong kiểm thử ứng dụng desktop hoặc di động.
* Đòi hỏi kinh nghiệm lập trình: Sử dụng Selenium WebDriver đòi hỏi tester phải có kiến thức vững về một ngôn ngữ lập trình cụ thể và kinh nghiệm sử dụng các công cụ kiểm thử tự động.
* Thiếu tính năng báo cáo tự động: Là cung cụ kiểm thử nhưng Selenium Webdriver chưa có sẵn chức năng báo cáo tự động nên việc khắc phục sự cố và sửa lỗi khó khăn hơn. Điều này đồng nghĩa với việc kiểm thử viên sẽ mất nhiều thời gian hơn để viết báo cáo kiểm thử.

## Các công cụ và thư viện hỗ trợ Selenium WebDriver

Selenium WebDriver là một công cụ mạnh mẽ để tự động hóa kiểm thử ứng dụng web. Tuy nhiên, để tối ưu hóa quá trình kiểm thử và nâng cao hiệu qua, có thể kết hợp Selenium WebDriver với các công cụ và thư viện sau:

* **Thư viện TestNG:**
* TestNG (Testing Next Generation) là một framework kiểm thử Java phổ biến, cung cấp nhiều tính năng hữu ích cho việc quản lý và thực thi các trường hợp kiểm thử.
* Cung cấp các annotation như @Test, @BeforeMethod, @AfterMethod để đánh dấu các phương thức là các bước kiểm thử, trước hoặc sau toàn bộ bộ kiểm thử hoặc nhóm kiểm thử.
* Quản lý thời gian: Có thể cấu hình TestNG để thực hiện các bước kiểm thử theo thứ tự nhất định hoặc song song, cũng như đặt thời gian timeout cho mỗi kiểm thử.
* Báo cáo và ghi nhật ký: TestNG tạo ra báo cáo chi tiết về kết quả kiểm thử, bao gồm thông tin về số lượng kiểm thử thành công và thất bại, thời gian chạy, và các chi tiết về lỗi nếu có.
* **Công cụ IntelliJ IDEA:**
* IntelliJ IDEA là một môi trường phát triển tích hợp mạnh mẽ cho Java.
* Cung cấp các công cụ hỗ trợ viết mã, kiểm tra lỗi cú pháp và tích hợp dễ dàng với TestNG, Maven.
* Hỗ trợ chạy kiểm thử trực tiếp trong IDE mà không cần thiết lập môi trường phức tạp.
* Tích hợp tốt với Git, Github, giúp quản lý mã nguồn hiệu quả
* Plugin: Hỗ trợ nhiều plugin hữu ích cho kiểm thử tự động hóa, ví dụ như plugin TestNG, Maven và Selenium.
* **Maven (Công cụ quản lý):**
* Maven là một công cụ quản lý dự án và xây dựng phần mềm, giúp tự động hóa quá trình xây dựng, kiểm thử và triển khai dự án. Nó sử dụng một file xml (pom.xml) để định nghĩa cấu hình dự án, bao gồm các phụ thuộc, plugin và cấu hình xây dựng.
* Maven hỗ trợ tự động hóa quá trình biên dịch, đóng gói và triển khải dự án.
* Hỗ trợ chạy các trường hợp kiểm thử tự động hóa và quản lý vòng đời dự án: Định nghĩa các giai đoạn vòng đời dự án, ví dụ như biên dịch, kiểm thử và triển khai.
* Ứng dụng thực tế: Quản lý các thư viện Selenium WebDriver và TestNG, tự động hóa quá trình xây dựng và chạy các trường hợp kiểm thử Selenium, tích hợp với các công cụ CI/CD để tự động hóa quá trình kiểm thử.
* **Trình điều khiển trình duyệt ChromeDriver:**
* ChromeDriver là một trình điều khiển web cho phép Selenium WebDriver điều khiển trình duyệt Google Chrome. Nó hoạt động như một cầu nối giữa Selenium WebDriver và Chrome, cho phép tự động hóa các thao tác trên trình duyệt Chrome, ví dụ như mở trang web, nhập dữ liệu, nhấp chuột và kiểm tra các phần tử trên trang.
* ChromeDriver tương thích tốt với Selenium WebDriver, cho phép viết các trường hợp kiểm thử tự động hóa dễ dàng.
* Được cập nhật thường xuyên để hỗ trợ các phiên bản mới nhất của Chrome.
* **Allure Report:**
* Allure Report là một framework tạo báo cáo kiểm thử linh hoạt và dễ hiểu, giúp trực quan hóa kết quả kiểm thử. Nó cung cấp các báo cáo chi tiết về kết quả kiểm thử bao gồm các bước kiểm thử, ảnh chụp màn hình và video.
* Cung cấp giao diện đồ họa trực quan và dễ hiểu, giúp theo dõi tiến độ kiểm thử và xác định các vấn đề một cách nhanh chóng.
* Cho phép đính kèm ảnh chụp màn hình và video vào báo cáo kiểm thử, giúp ghi lại các bước kiểm thử và lỗi.
* Allure Report hỗ trợ nhiều ngôn ngữ lập trình và framework kiểm thử, bao gồm Java, Python, JavaScript và thư viện TestNG.
* Tích hợp tốt với các công cụ CI/CD, cho phép tự động hóa quá trình tạo báo cáo kiểm thử.

Ngoài các công cụ và thư viện đã đề cập, Selenium WebDriver có thể kết hợp với nhiều công cụ và thư viện khác để tạo hiệu quả cao trong quá trình tự động hóa kiểm thử phần mềm như Selenium Grid, Cucumber, framework Junit, Docker, v.v.

## Các bước cài đặt Selenium WebDriver sử dụng ngôn ngữ Java

**Bước 1:** Cài đặt Chrome Brower

* + Vào trang web <https://www.google.com/intl/vi_vn/chrome/> và nhấn nút “Tải xuống tại đây”.
  + Thực hiện cài đặt Chrome.

**Bước 2:** Cài đặt Chrome Driver

 Vào trang web <https://chromedriver.chromium.org/downloads> và tải phiên bản phù hợp với Chrome Browser đã tải ở bước 1.

**Bước 3:** Cài đặt thư viện Selenium Webdriver sử dụng ngôn ngữ Java vào IntelliJ IDEA.

Trước khi cài đặt ta phải có điều kiện sử dụng Java 8 trở lên và có công cụ hỗ trợ code (Intellij IDEA, Eclipse,..). Ở đây ta thực hiện cài đặt Selenium WebDriver với công cụ IntelliJ IDEA.



Hình 2.2: Màn hình trang tải về Selenium WebDriver

Sau đó, mở công cụ IntelliJ IDEA lên và tạo một project mới.

Hình 2.3: Thao tác tạo một project mới trong IntelliJ

Chọn cấu hình JDK đúng với phiên bản Java



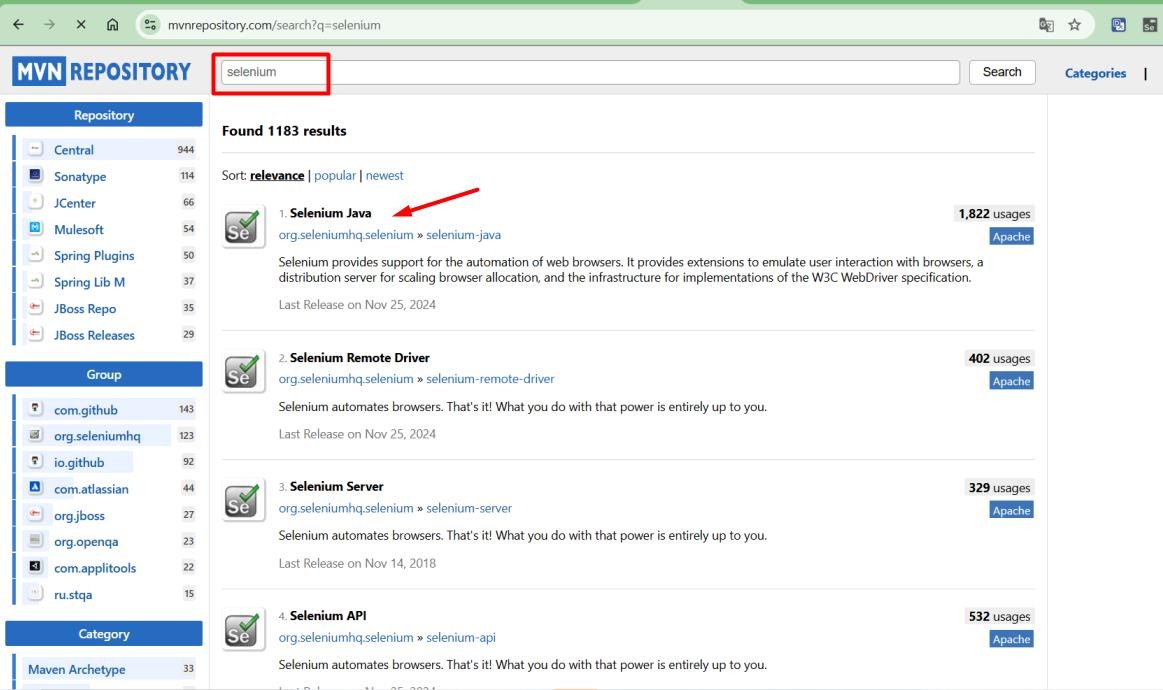
Hình 2.4: Chọn cấu hình JDK đúng với phiên bản Java ở IntelliJ

Sau khi tạo project Java xong, tới một bước quan trọng là cài đặt thư viện của Selenium WebDriver vào project để nó có thể chạy được các yêu cầu của WebDriver.

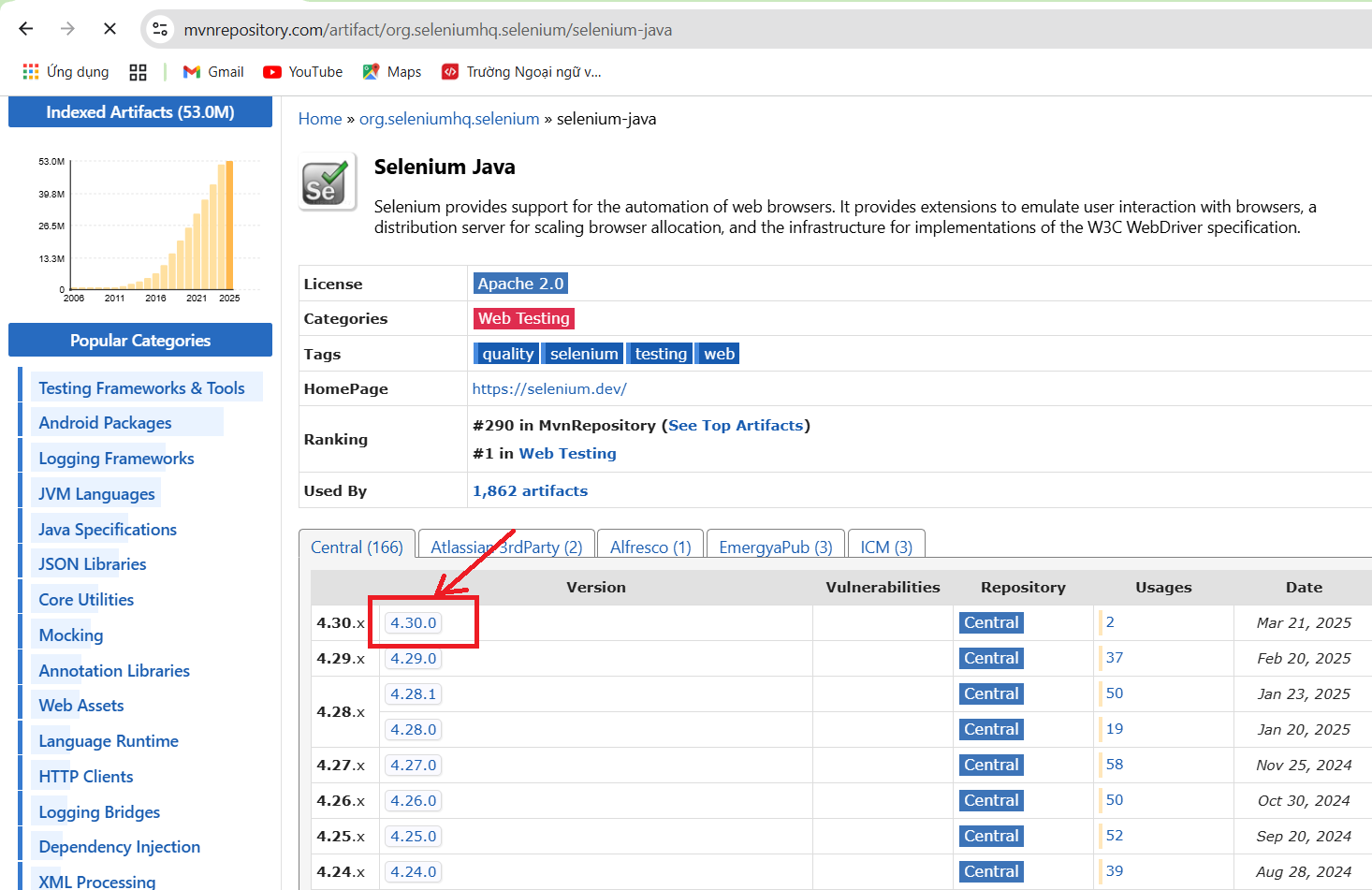
**Bước 4:** Cài đặt thư viện Selenium vào Maven project trên IntelliJ

Để cài đặt thư viện Selenium vào Maven project thì sẽ cài nó vào file pom.xml

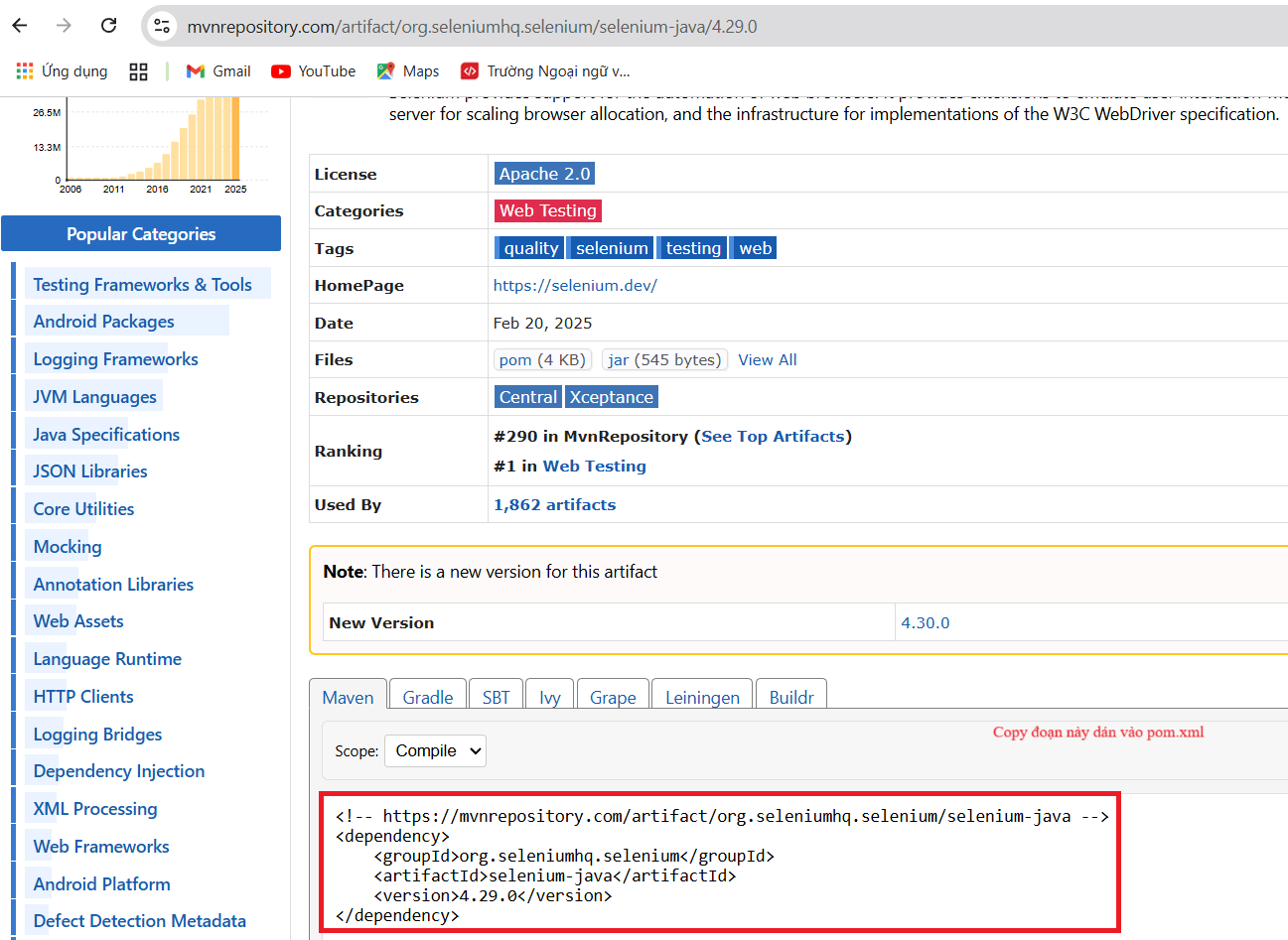
Thư viện được lấy tại trang web: <https://mvnrepository.com/>. Ta cần tìm Selenium Java, chọn phiên bản phù hợp. Sau đó, cần lấy ra dependency (phụ thuộc) dành cho Maven, được sử dụng để khai báo thư viện Selenium Java trong dự án Java sử dụng Maven.



Hình 2.5: Thao tác tải Selenium Java



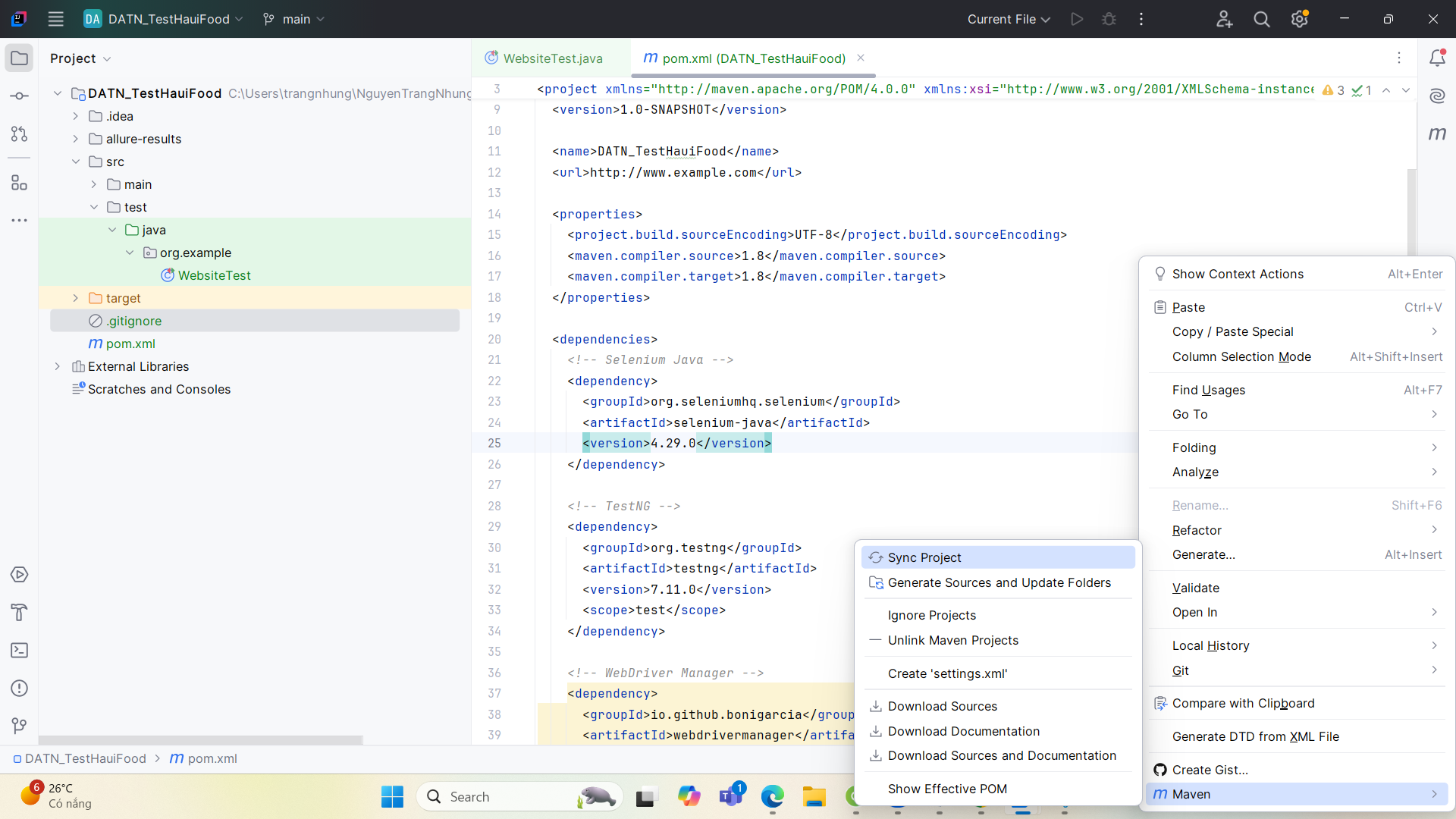
Hình 2.6: Chọn phiên bản cho Selenium Java



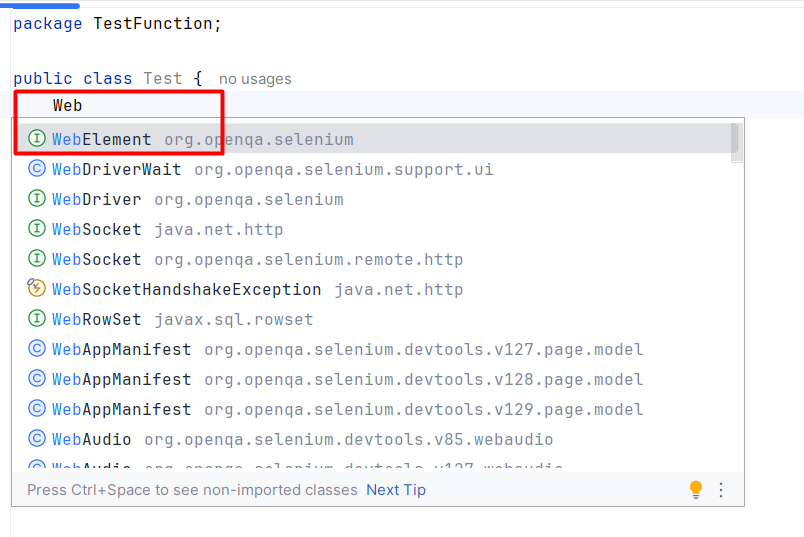
Hình 2.7: Dependency để khai khai báo thư viện Selenium Java

Sau khi lấy được dependency của phiên bản phù hợp như trên hình 2.7, ta cần thực hiện dán vào file pom.xml trên IntelliJ và phải dán đoạn mã trong cặp thẻ <dependencies> </dependencies>.

Khi đã thêm dependency vào pom.xml, cần khởi động lại Maven bằng cách nhấn vào biểu tượng Reload hoặc nhấp chuột phải > Maven > Reload project.



Hình 2.8: Thực hiện khởi động lại Maven sau khi thêm dependency

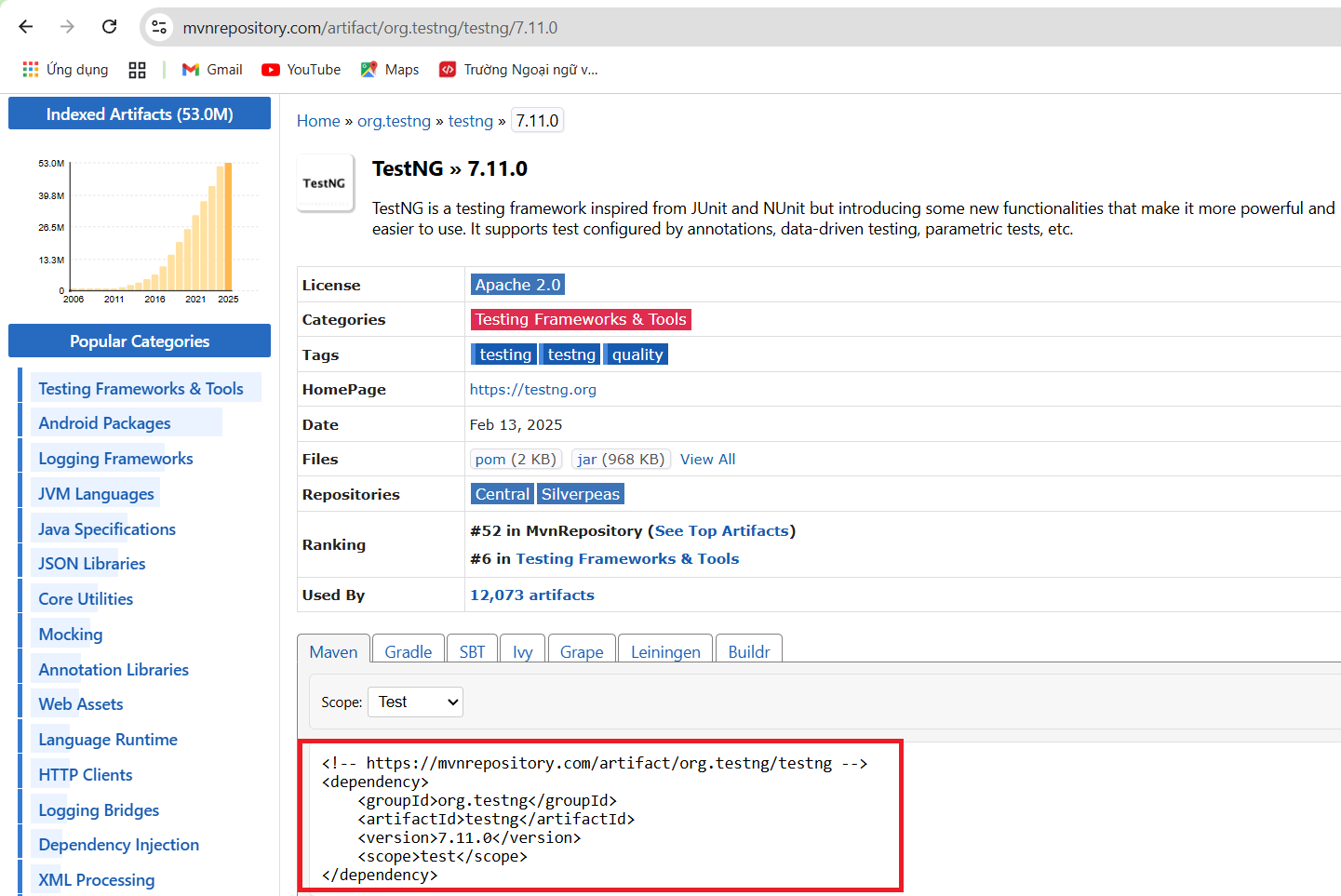
Kiểm tra Selenium đã cài đặt vào maven project: Nếu gợi ý hiển thị lên như hình 2.9 dưới đây là đã cài đặt được Selenium thành công.

Hình 2.9: Hoàn thiện cài đặt thư viện Selenium

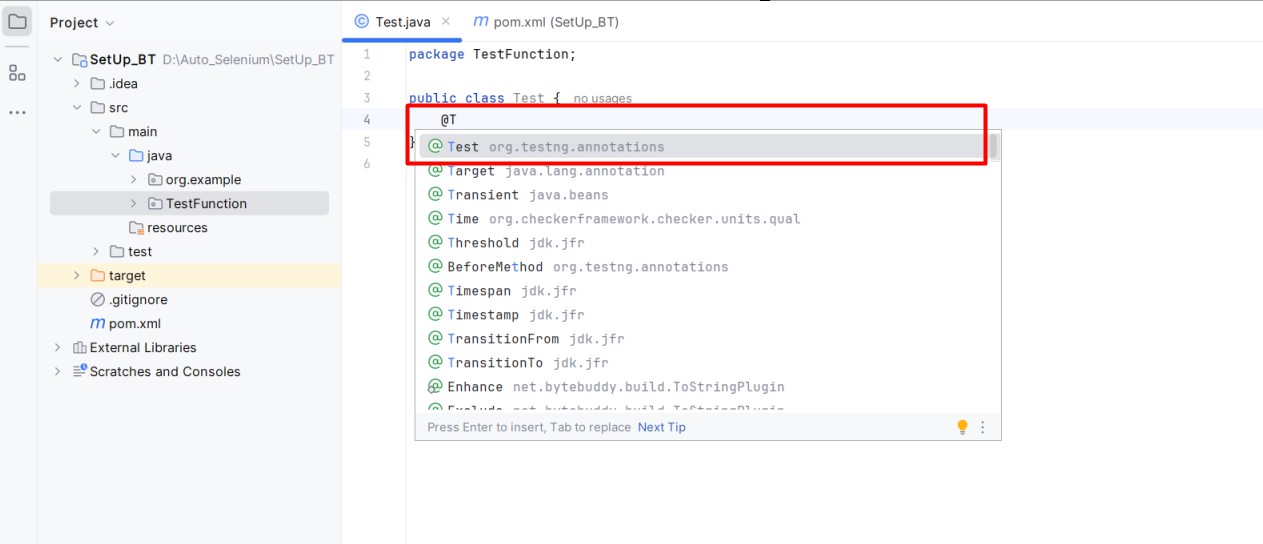
**Bước 4:** Cài đặt thư viện TestNG vào IntelliJ

Để thực hiện thêm thư viện TestNG vào file pom.xml cần tiến hành tương tự như Selnium Java:

Trên trang Maven Repository thực hiện tìm kiếm TestNG > Chọn phiên bản phù hợp > Lấy thông tin dependency (phụ thuộc) để thêm vào pom.xml của maven project đã tạo.



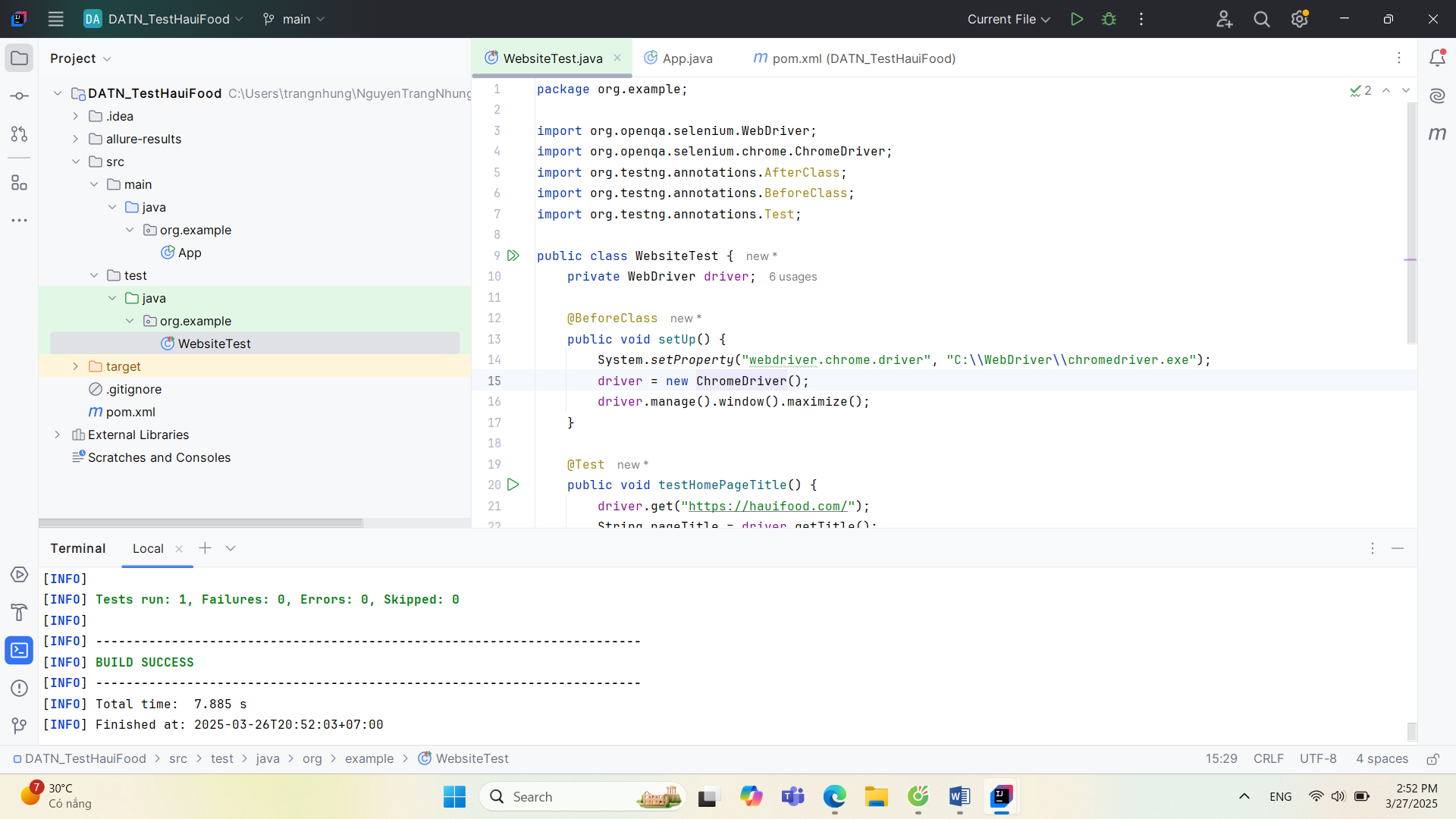
Hình 2.10: Dependency để khai báo thư viện TestNG

Sau đó, thêm dependency để khai báo thư viện TestNG vào file pom.xml và thực hiện reload Maven để hoàn tất cài đặt. Để kiểm tra xem thư viện TestNG đã được cài đặt thành công hay chưa, ta cần kiểm tra gợi ý hiển thị lên như hình dưới đây.

Hình 2.11: Hoàn thiện cài đặt TestNG

**Bước 5:** Viết test script đầu tiên

* Tạo class kiểm thử: Để viết một test script ta cần tạo class kiểm thử. Trước tiên, trong IntelliJ, tạo một package mới như sau: src/test/java/org/example. Sau đó tạo một file Java mới, ví dụ: WebsiteTest.java
* Viết script kiểm thử theo kịch bản kiểm thử
* Chạy test script: Có 2 cách để thực hiện:
* Nhấn chuột phải vào file cần chạy test script và chọn Run
* Chạy bằng terminal với lệnh sau: mvn test hoặc mvn clean test.

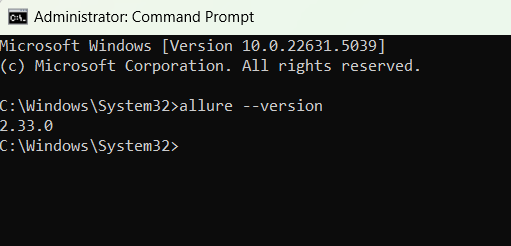


Hình 2.12: Kết quả của test script khi chạy bằng terminal

**Bước 6:** Tạo báo cáo Allure Report

1. Cài đặt Allure:

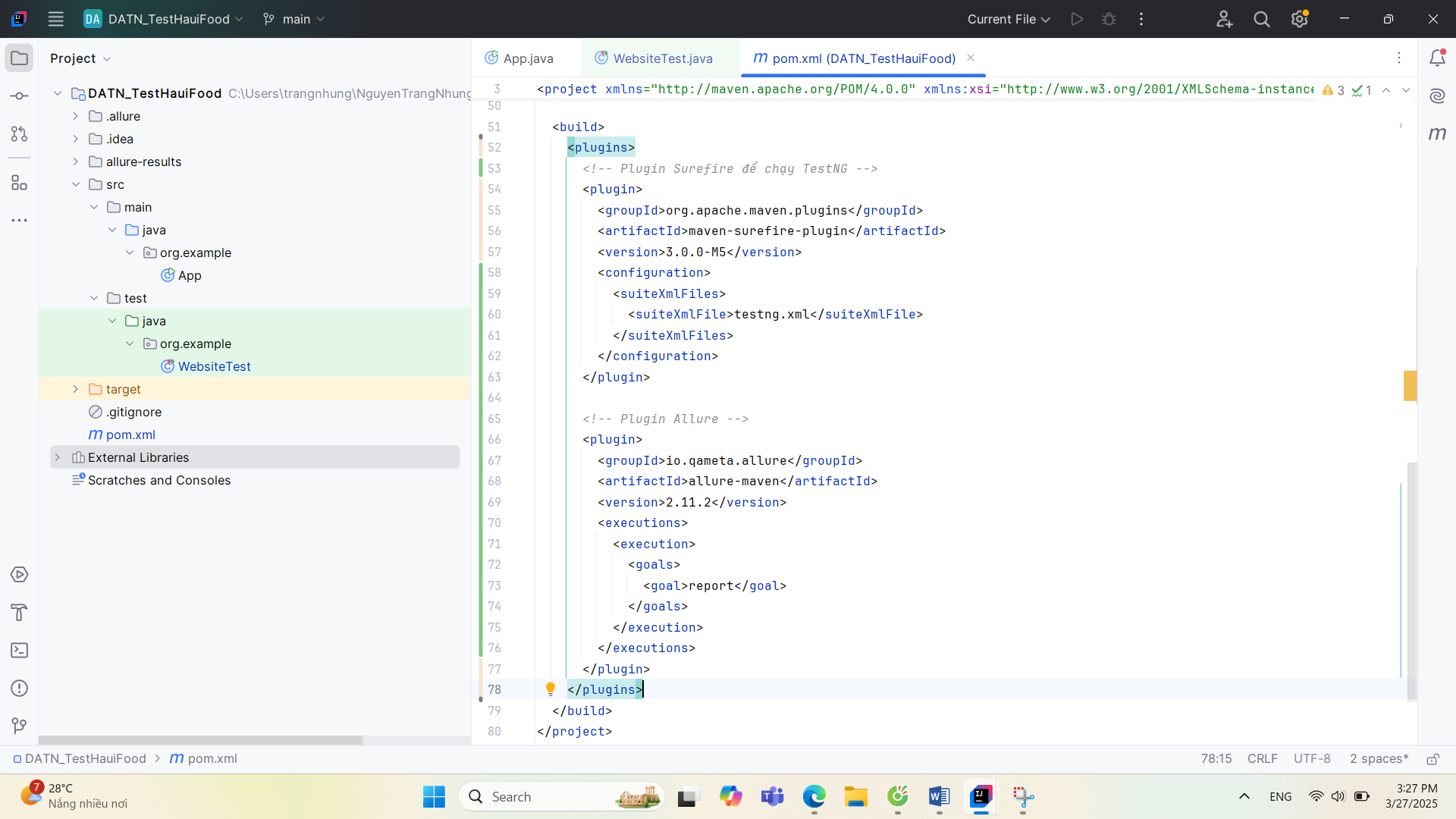
* Tải và cài đặt Allure từ <https://github.com/allure-framework/allure2>
* Kiểm tra Allure đã cài đặt thành công bằng lệnh: allure - -version



Hình 2.13: Kiểm tra phiên bản Allure

1. Cấu hình plugin allure trong Maven:

* Thêm vào file pom.xml phần plugin (lưu ý: cần thêm trong cặp thẻ <plugins></plugins>)

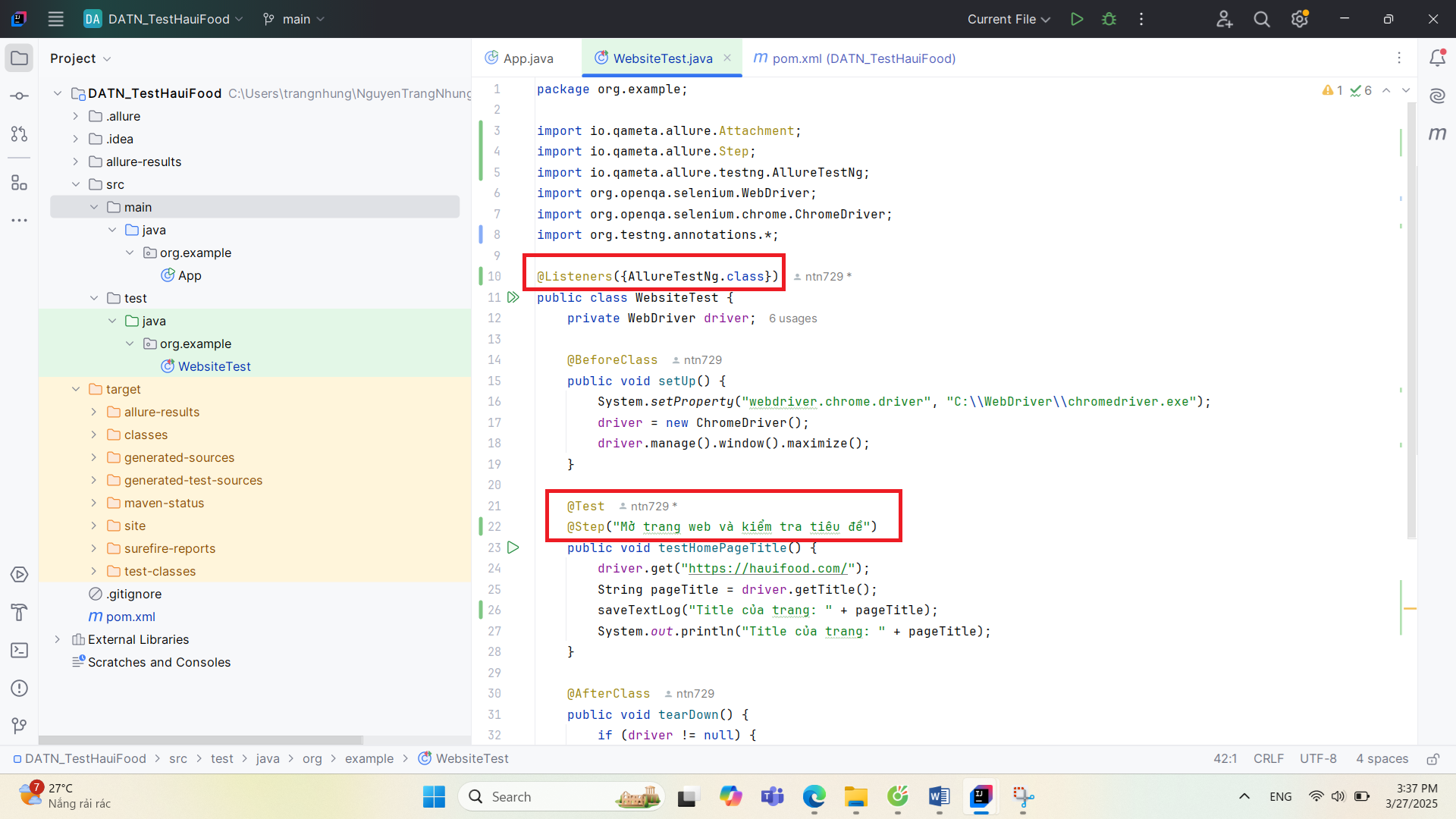


Hình 2.14: Cấu hình plugin Allure trong Maven

* Thực hiện reload lại maven

1. Cấu hình Listener của Allure:

* Cần thêm @Listeners vào class test để Allure có thể ghi nhận kết quả.
* Có thể thêm các annotation thuộc thư viện Allure để mô tả, phân loại và đánh giá mức độ quan trọng của các test case trong báo cáo Allure Report. Các annotation thường dùng bao gồm:
* @Description: Mô tả test case
* @Epic: Xác định test case liên quan đến một tính năng lớn trong hệ thống
* @Feature: Xác định tính năng nhỏ trong hệ thống
* @Story: Xác định user story (tình hướng cụ thể)
* @Severity: Xác định mức độ quan trọng của test case
* @Step: Các bước thực hiện



Hình 2.15: Cấu hình Listener và các annotation của Allure

1. Chạy kiểm thử và sinh báo cáo

* Chạy test script trên terminal bằng lệnh: mvn test –Dtest = “ten\_thu muc.ten\_class” để chạy script chỉ định hoặc mvn clean test để chạy toàn bộ script.
* Sinh báo cáo Allure bằng lệnh: mvn allure:report
* Xem báo cáo Allure bằng: allure serve target/allure-results
* Kết quả mong đợi: Allure sẽ tự động tạo một báo cáo HTML từ dữ liệu từ trong thư mục target/allure-results. Một trình duyệt web sẽ mở ra, hiển thị giao diện báo cáo của Allure.



Hình 2.16: Giao diện báo cáo Allure

# ỨNG DỤNG SELENIUM WEBDRIVER KIỂM THỬ ỨNG DỤNG WEB

## Giới thiệu về website đặt đồ ăn trực tuyến Haui Food

Haui Food (<https://hauifood.vn/>) là website mà em lựa chọn để thực hiện kiểm thử tự động trong đồ án. Đây là một sản phẩm được phát triển bởi nhóm sinh viên trường Đại học Công nghiệp Hà Nội trong môn Thực tập chuyên ngành, với mục tiêu ứng dụng các công nghệ hiện đại để xây dựng một nền tảng đặt đồ ăn trực tuyến. Dự án này được triển khai dựa trên các công nghệ tiên tiến như ReactJS cho giao diện người dùng, NodeJS và JavaScript để xử lý logic phía server, MongoDB để quản lý cơ sở dữ liệu, đồng thời áp dụng kiến trúc Microservices để tăng tính linh hoạt và hiệu suất của hệ thống.

Website Haui Food ra đời nhằm tối ưu hóa quy trình đặt món ăn trực tuyến, giúp nâng cao trải nghiệm người dùng thông qua một nền tảng tiện lợi, nhanh chóng và hiệu quả. Hệ thống hướng đến đối tượng là cán bộ, nhân viên và sinh viên của trường Đại học Công nghiệp Hà Nội, giúp họ dễ dàng tiếp cận với các dịch vụ ẩm thực mà không cần mất nhiều thời gian di chuyển hay chờ đợi. Ứng dụng này cung cấp đầy đủ các tính năng cần thiết của một nền tảng đặt đồ ăn trực tuyến, bao gồm:

* Đăng ký và đăng nhập tài khoản giúp người dùng cá nhân hóa trải nghiệm.
* Tìm kiếm và lựa chọn món ăn theo danh mục hoặc khẩu vị cá nhân.
* Đặt hàng trực tuyến với giao diện thân thiện, dễ sử dụng.
* Thanh toán linh hoạt qua nhiều phương thức như ví điện tử, thẻ ngân hàng hoặc thanh toán khi nhận hàng.
* Theo dõi trạng thái đơn hàng theo thời gian thực, giúp người dùng dễ dàng quản lý và nhận đơn hàng đúng thời điểm.

Giao diện một số chức năng của trang web:

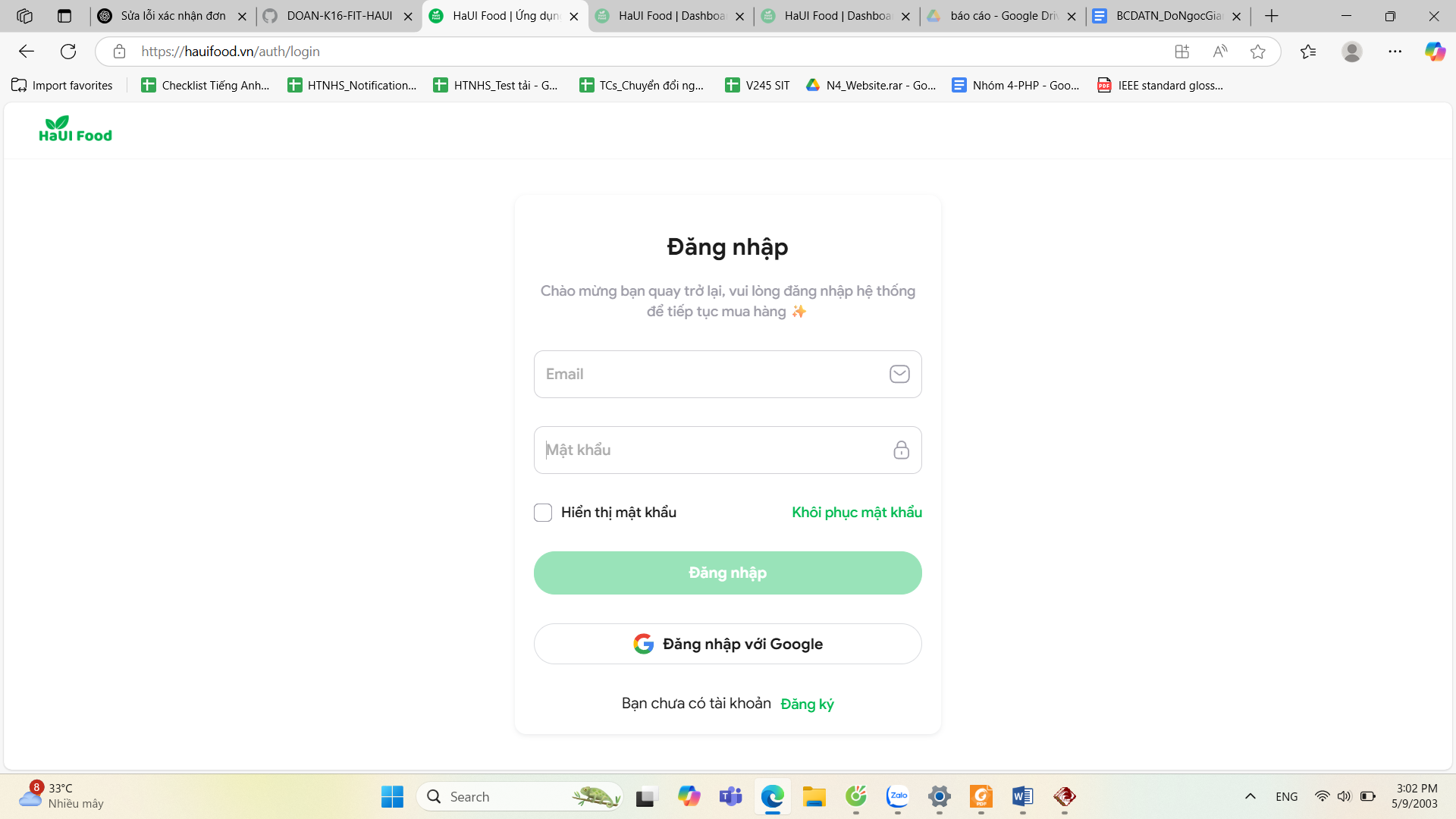
* Giao diện chức năng Đăng ký:

**A screenshot of a login form

AI-generated content may be incorrect.**

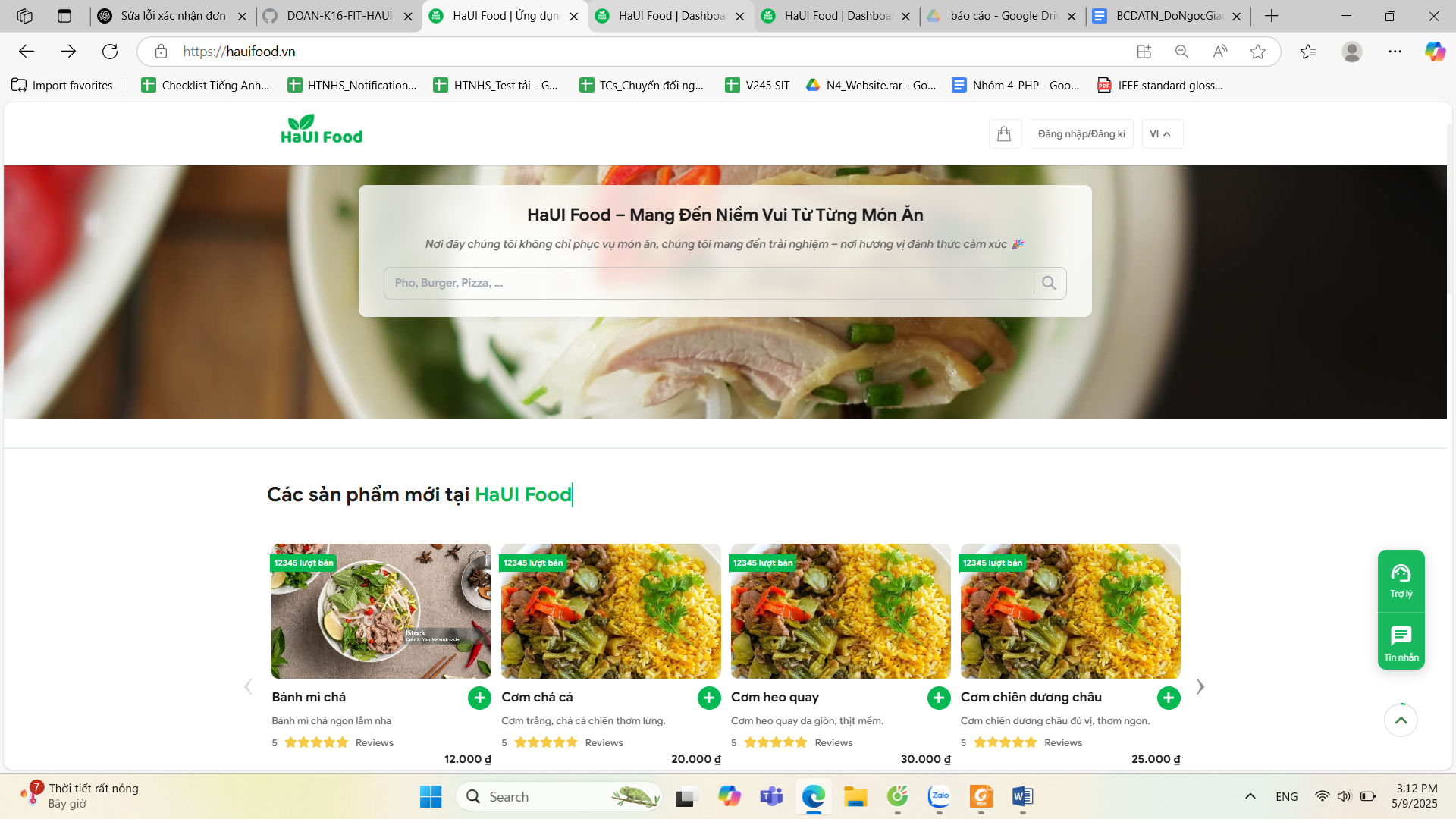
Hình 3.1: Màn hình Đăng ký

* Giao diện chức năng Đăng nhập:



Hình 3.2 Màn hình Đăng nhập

* Giao diện trang chủ:



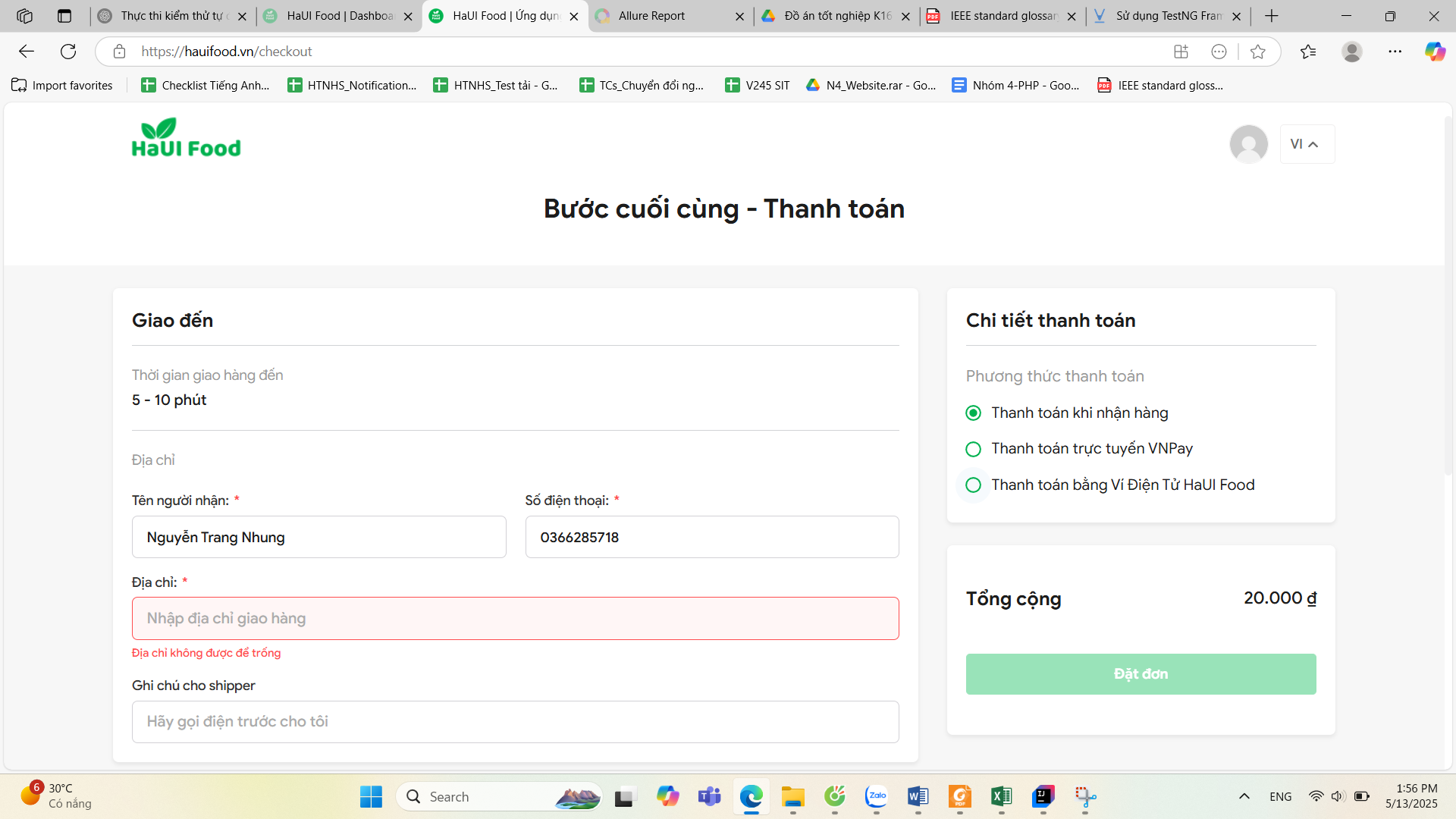
Hình 3.3:Màn hình Trang chủ

* Giao diện chức năng Xem chi tiết sản phẩm:



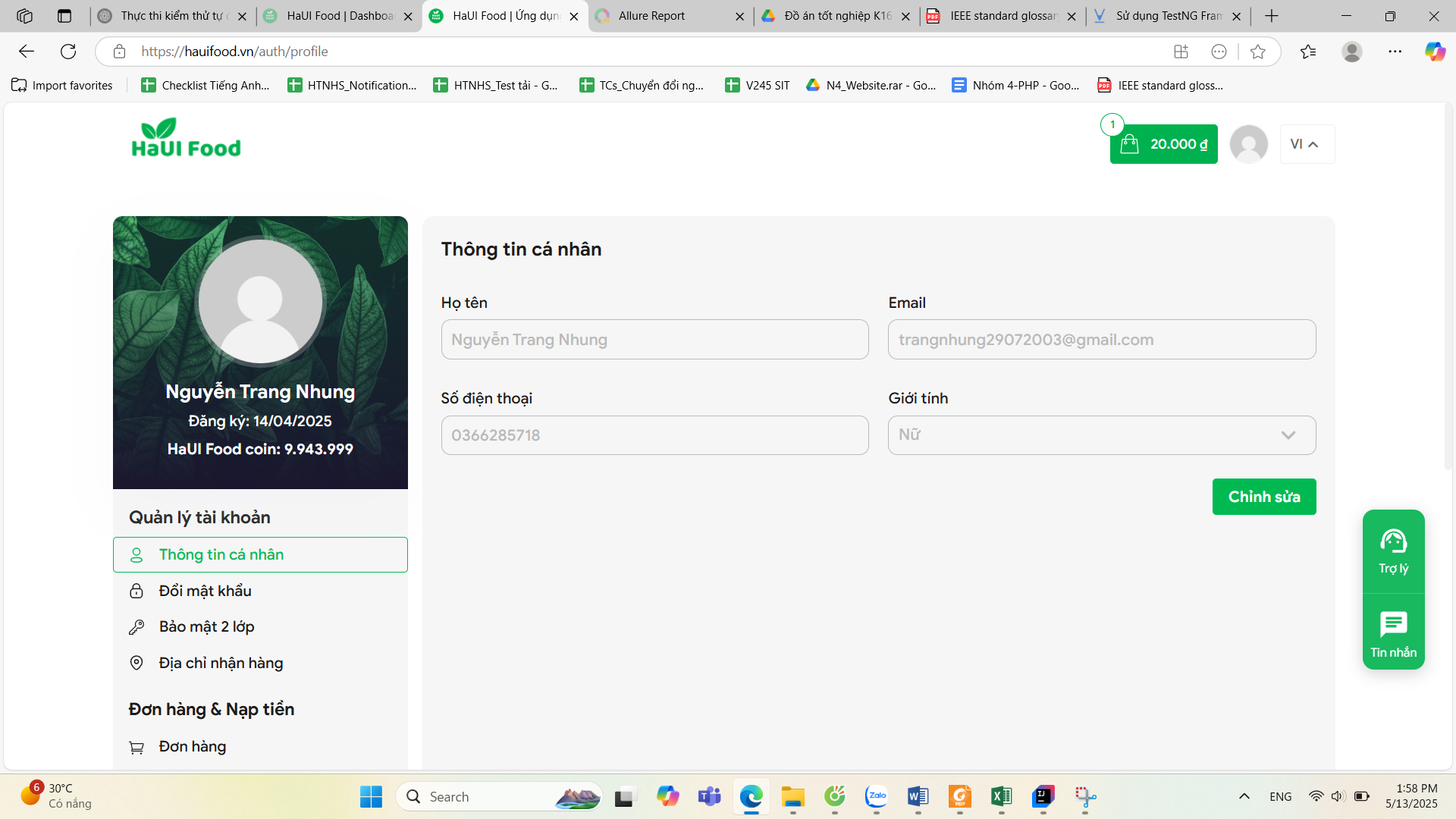
Hình 3.4: Màn hình Xem chi tiết sản phẩm

* Giao diện chức năng Đặt hàng:



Hình 3.5: Màn hình Đặt hàng

* Giao diện màn Trang cá nhân:



Hình 3.6: Màn hình Trang cá nhân

## Phân tích yêu cầu phần mềm

### Biểu đồ use case



Hình 3.7: Biểu đồ Use case phía người dùng



Hình 3.8: Biểu đồ Use case phía Quản trị viên

**Mô tả vắn tắt:**

* Đăng ký: Chức năng này cho phép người dùng có thể đăng ký tài khoản mới trên trang web.
* Đăng nhập: Chức năng này cho phép người dùng/ quản trị viên có thể đăng nhập tài khoản hợp lệ.
* Đăng xuất: Chức năng này cho phép người dùng đăng xuất khỏi phiên đăng nhập khi không cần sử dụng nữa
* Tìm kiếm: Chức năng này cho phép người dùng thực hiện tìm kiếm bằng tên sản phẩm hoặc cửa hàng.
* Xem chi tiết sản phẩm: Chức năng này cho phép người dùng có thể xem thông tin chi tiết về một sản phẩm cụ thể.
* Đặt hàng: Chức năng này cho phép người dùng thanh toán và hoàn tất quá trình mua hàng trên trang web.
* Hủy đơn hàng: Chức năng này cho phép người dùng thực hiện hủy đơn hàng đã đặt trước đó khi ở trạng thái chờ xác nhận.
* Xem danh sách đơn hàng: Chức năng này cho phép người dùng xem danh sách các đơn hàng đã mua trước đó.
* Thêm sản phẩm vào giỏ hàng: Chức năng này cho phép người dùng thêm một sản phẩm vào giỏ hàng
* Xóa sản phẩm khỏi giỏ hàng: Chức năng này cho phép người dùng xóa sản phẩm khỏi giỏ hàng mà người dùng đã thêm trước đó.
* Bật/tắt xác thực 2 lớp: Chức năng này cho phép người dùng bật hoặc tắt trạng thái bảo mật 2 lớp của tài khoản.
* Đổi mật khẩu: Chức năng này cho phép người dùng thực hiện đổi mật khẩu mới sau khi đã đăng nhập.
* Quên mật khẩu: Chức năng này cho phép người dùng khôi phục mật khẩu ở trang đăng nhập khi quên mật khẩu để đăng nhập.
* Xem thông tin tài khoản: Chức năng này cho phép xem chi tiết thông tin tài khoản cá nhân.
* Cập nhật thông tin cá nhân: Chức năng này cho phép người dùng cập nhật mới các thông tin cá nhân của tài khoản.
* Nạp tiền tài khoản: Chức năng này cho phép người dùng nạp tiền vào tài khoản của mình để có thể thực hiện mua hàng.
* Chat với cửa hàng: Chức năng này cho phép người dùng nhắn tin trực tiếp với các cửa hàng.
* Quản lý đơn hàng: Chức năng này cho phép quản trị viên xem, tìm kiếm và cập nhật trạng thái các đơn hàng.
* Quản lý người dùng: Chức năng này cho phép quản trị viên xem, tìm kiếm, thêm, sửa, xóa, khóa/mở khóa người dùng.
* Quản lý loại sản phẩm: Chức năng này cho phép quản trị viên xem, tìm kiếm, thêm, sửa, xóa các loại sản phẩm.
* Quản lý sản phẩm: Chức năng này cho phép quản trị viên xem, tìm kiếm, thêm, sửa, xóa các sản phẩm.
* Thống kế: Chức năng này cho phép quản trị viên thực hiện xem các thống kê chi tiết về doanh thu, số đơn hàng, tỉ lệ trạng thái đơn hàng trên dashboard.

### Mô tả chi tiết use case

#### Mô tả use case Đăng nhập

Use case này cho phép người dùng đăng nhập tài khoản đã đăng ký của mình để thực hiện các chức năng khác của hệ thống.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người kích vào biểu tượng “Đăng nhập” trên thanh menu. Hệ thống sẽ hiển thị màn hình đăng nhập và yêu cầu người dùng nhập tài khoản đăng nhập.
2. Người dùng nhập tài khoản và mật khẩu sau đó kích vào nút “Đăng nhập”. Hệ thống sẽ kiểm tra tài khoản và mật khẩu đã nhập ở bảng USERS và hiển thị trang chủ hệ thống với tài khoản đã đăng nhập. Use case kết thúc.

* **Luồng rẽ nhánh:**
  1. Tại bước 2 trong luồng cơ bản, nếu người dùng nhập tài khoản sai, hệ thống sẽ hiển thị thông báo “Tài khoản hoặc mật khẩu không chính xác”. Người dùng có thể kích vào ô nhập trên màn hình và nhập lại tài khoản và mật khẩu để đăng nhập lại. Use case kết thúc.
  2. Tại bước 2 trong luồng cơ bản nếu tài khoản người dùng chưa xác nhận khi đăng ký, hệ thống sẽ hiển thị thông báo “Vui lòng xác nhận tài khoản”.
  3. Tại bước 2 trong luồng cơ bản nếu tài khoản bật xác thực 2 lớp, hệ thống sẽ điều hướng người dùng sang trang nhập mã xác nhận. Trường hợp người dùng nhập đúng mã hệ thống trả về “Đăng nhập thành công” use case kết thúc trường hợp ngược lại “Mã xác nhận không chính xác”.
  4. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo “Không có dữ liệu”. Use case kết thúc.
* **Các yêu cầu đặc biệt:** Không có.
* **Tiền điều kiện:** Người dùng đã có tài khoản được lưu trong cơ sở dữ liệu.
* **Hậu điều kiện:** Không có.
* **Điểm mở rộng:** Không có.

#### Mô tả use case Đăng ký

Use case này cho phép người dùng đăng ký tài khoản trên hệ thống.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case bắt đầu khi người dùng nhấn chuột vào nút “Đăng ký” trên màn hình đăng nhập. Hệ thống sẽ hiển thị màn hình nhập các thông tin tài khoản để đăng ký.
2. Người dùng nhập các nội dung mà trang web yêu cầu để đăng ký tài khoản sử dụng: Họ tên, email, mật khẩu.
3. Sau khi tạo tài khoản thành công, website đưa bạn đến phần đăng nhập tài khoản.

* **Luồng rẽ nhánh:**

1. Tại bước 2 trong luồng cơ bản, người dùng nhập họ tên, email hay mật khẩu không đúng với form mẫu (không sử dụng kí tự đặc biệt), hệ thống sẽ báo lỗi và yêu cầu sinh viên cần nhập lại.
2. Tại bước 3 trong luồng cơ bản, nếu người dùng đăng ký tài khoản mới trùng với email đã tồn tại trên hệ thống, hệ thống sẽ trả về thông báo “Email đã tồn tại”. Use case kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo “Không có dữ liệu”. Use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Không có
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Tìm kiếm

Use case này cho phép người dùng tìm kiếm các sản phẩm và cửa hàng của trang web.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case bắt đầu khi người dùng nhấn vào ô tìm kiếm sau đó nhập tên sản phẩm hoặc cửa hàng cần tìm kiếm và nhấn vào nút tìm kiếm ở màn hình trang chủ.
2. Hệ thống sẽ hiển thị danh sách các sản phẩm hoặc các cửa hàng có liên quan đến từ khóa người dùng nhập vào từ ô tìm kiếm.

* **Luồng rẽ nhánh:**

1. Tại bước 1 trong luồng cơ bản, người dùng tìm kiếm bằng cách nhập từ bàn phím nếu trong bảng PRODUCTS và bảng USERS không có dữ liệu về sản phẩm hoặc cửa hàng, hệ thống sẽ hiển thị thông báo “Không tìm thấy sản phẩm hoặc cửa hàng”
2. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo “Không có dữ liệu” và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Không có
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Xem thông tin tài khoản

Use case này cho phép người dùng có thể xem được thông tin cá nhân của mình.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case bắt đầu khi người dùng lựa chọn vào mục “Trang cá nhân” ở avatar menu hệ thống sẽ lấy thông tin chi tiết của khách hàng bảo gồm: tên, ngày sinh, email, số điện thoại, mã sinh viên, ngày sinh, giới tính, ảnh avatar, ngày đăng ký, hauifood coin từ cơ sở dữ liệu và hiển thị lên màn hình.
2. Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện nếu use case không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống trước khi thực hiện use case
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Cập nhật thông tin cá nhân

Use case này cho phép người dùng cập nhật thông tin cá nhân như là tên, số điện thoại, mã sinh viên, ngày sinh, giới tỉnh, ảnh avatar.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case bắt đầu khi người dùng kích vào mục “Thông tin cá nhân” trong trang cá nhân hoặc khi truy cập vào trang cá nhân hệ thống sẽ lấy thông tin chi tiết của khách hàng bảo gồm: tên, ngày sinh, email, số điện thoại, mã sinh viên, ngày sinh, giới tính, ảnh avatar, ngày đăng ký, hauifood coin từ cơ sở dữ liệu và hiển thị lên màn hình.
2. Người dùng kích vào nút chỉnh sửa hệ thống sẽ cho phép người dùng chỉnh sửa các trường họ tên, số điện thoại, mã sinh viên, ngày sinh, giới tính, ảnh avatar.
3. Người dùng chỉnh sửa lại thông tin và bấm nút cập nhật hệ thống sẽ cập nhật thông tin đã chỉnh sửa và hiển thị thông tin được chỉnh sửa thành công lên màn hình.
4. Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 3 trong luồng cơ bản khi người dùng không thay đổi thông tin mà nhấn nút cập nhật thì hệ thống sẽ hiển thị thông báo “Không có thay đổi nào được thực hiện”.
2. Tại bước 3 trong luồng cơ bản khi người dùng kích nút hủy thì mọi thay đổi trước đó sẽ bị xóa và hiển thị thông tin ban đầu, use case kết kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo. Use case kết thúc.

* **Yêu cầu đặc biệt**: Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống trước khi thực hiện use case
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Đổi mật khẩu

Use case này cho phép người dùng thay đổi mật khẩu của tài khoản.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng bấm vào mục “Đổi mật khẩu” trong cá nhân hệ thống sẽ hiển thị biểu mẫu gồm các trường mật khẩu cũ, mật khẩu mới, xác nhận mật khẩu mới.
2. Người dùng nhập mật khẩu cũ, mật khẩu mới, xác nhận mật khẩu mới và kích vào nút cập nhật hệ thống sẽ cập nhật lại mật khẩu và hiển thị thông báo trên màn hình.
3. Use case kết thúc

* **Luồng rẽ nhánh:**

1. Tại bước 2 trong luồng cơ bản khi người dùng không nhập đầy đủ các trường hoặc trường mật khẩu mới và xác nhận mật khẩu mới không khớp với nhau mà nhấn nút cập nhật thì hệ thống sẽ hiện thông báo.
2. Tại bước 2 trong luồng cơ bản khi người dùng kích nút hủy thì mọi thông tin nhập vào sẽ bị xóa và hiển thị các trường ở trạng thái trống, use case kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo.
4. Use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống trước khi thực hiện use case
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Xem danh sách đơn hàng

Use case này cho phép người dùng xem lại tất cả đơn hàng mà mình đã đặt.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng bấm vào mục “Đơn mua” trong trang cá nhân hệ thống sẽ mặc định lấy thông tin tất cả đơn hàng bạn đã mua, các đơn hàng gồm tên cửa hàng, trạng thái đơn hàng, danh sách các sản phẩm, địa chỉ nhận hàng, ghi chú, thời gian tạo, phương thức thanh toán, tổng tiền trong cơ sở dữ liệu và hiển thị lên màn hình.
2. Người dùng kích vào một trong các trạng thái trên thanh trạng thái gồm “Tất cả”, “Chờ xác nhận”, “Đã xác nhận”, “Đang giao”, “Hoàn thành”, “Đã hủy”, “Bị từ chối” hệ thống sẽ lấy thông tin các đơn hàng ứng với trạng thái đã chọn, các đơn hàng gồm tên cửa hàng, trạng thái đơn hàng, danh sách các sản phẩm, địa chỉ nhận hàng, ghi chú, thời gian tạo, phương thức thanh toán, tổng tiền trong cơ sở dữ liệu và hiển thị lên màn hình.
3. Use case kết thúc.
   * **Luồng rẽ nhánh:**
     1. Tại bước 1 và 2 trong luồng cơ bản nếu không có đơn hàng nào trong cơ sở dữ liệu hệ thống sẽ hiển thị thông báo “Bạn hiện không có đơn hàng nào”, use case kết thúc.
     2. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống trước khi thực hiện use case
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Đặt hàng

Use case này cho phép người dùng đặt hàng các sản có trong giỏ hàng.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng chọn hay bỏ chọn các ô checkbox ở từng sản phẩm hoặc từng cửa hàng hoặc giữ nguyên mặc định tất cả đều được check và kích vào nút “Thanh toán”. Hệ thống sẽ điều hướng người dùng sang trang checkout để chuyển sang bước tiếp theo.
2. Hệ thống lấy thông tin các sản phẩm được chọn và hiển thị lên màn hình cùng với thông tin của đơn hàng gồm: Thời gian giao hàng, địa chỉ, ghi chú, tổng giá, phí áp dụng, phương thức thanh toán, QR. Tại đây người dùng cần thêm thông tin thanh toán:
   1. Người dùng thêm địa chỉ giao hàng bằng cách kích vào nút “Tòa nhà”, “Tầng”, “Phòng”, hệ thống sẽ hiển thị ra danh sách các tòa nhà, tầng, phòng để người dùng chọn.
   2. Người dùng có thể thêm ghi chú cho người giao hàng bằng cách nhập vào input “Ghi chú cho shipper”.
   3. Người dùng thay đổi phương thức thanh toán (nếu có) hoặc để nguyên mặc định của hệ thống “Thanh toán khi nhận hàng”.
3. Người dùng kích nút “Đặt đơn”, hệ thống sẽ kiểm tra và ghi nhận đơn hàng vào hệ thống và điều hướng người dùng đến trang Thông tin cá nhân.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 2a trong luồng cơ bản nếu người dùng chưa chọn đủ cả 3 trường tòa nhà, tầng, phòng học hệ thống sẽ không cho người dùng kích nút “Đặt đơn”.
2. Tại bước 2c trong luồng cơ bản, nếu người dùng chọn Thanh toán chuyển khoản hoặc Thanh toán bằng ví điện tử HaUI Food mà số dư trong ví điện tử không đủ hoặc chuyển khoản thất bại, tại bước 3 hệ thống sẽ thông báo lỗi ra màn hình. Use case kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi ra màn hình. Use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống, có sản phẩm trong giỏ hàng trước khi thực hiện các use case
* **Hậu điều kiện:** Nếu use case kết thúc thành công thì thông tin về đơn hàng sẽ được cập nhật thêm vào trong cơ sở dữ liệu.
* **Điểm mở rộng:** Không có

#### Mô tả use case Đăng xuất

Use case này cho phép người dùng thoát khỏi hệ thống sau khi đã đăng nhập.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng nhấp chuột vào ảnh đại diện. Khi đó hệ thống hiển thị 2 mục lựa chọn: Trang cá nhân và Đăng xuất.
2. Khi người dùng chọn Đăng xuất thì hệ thống thực hiện đăng xuất người dùng, hiển thị thông báo “Đăng xuất thành công” và điều hướng màn hình về trang chủ.

* **Luồng rẽ nhánh:**

1. Tại bước 2 trong luồng cơ bản, nếu xảy ra lỗi hệ thống, đăng xuất không thành công và người dùng vẫn duy trì trạng thái đang trong phiên đăng nhập.
2. Nếu người dùng cố gắng truy cập trang yêu cầu đăng nhập sau khi đăng xuất, hệ thống hiển thị thông báo yêu cầu đăng nhập lại.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống, có sản phẩm trong giỏ hàng trước khi thực hiện các use case
* **Hậu điều kiện:** Người dùng được đăng xuất thành công, không thể thực hiện các chức năng yêu cầu đăng nhập nếu chưa đăng nhập lại.
* **Điểm mở rộng:** Không có

#### Mô tả use case Xem chi tiết sản phẩm

**Mô tả vắn tắt:** Use case này cho phép người dùng xem thông tin chi tiết sản phẩm.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng kích vào chọn 1 sản phẩm. Hệ thống lấy ra thông tin về sản phẩm bao gồm: tên sản phẩm, mô tả sản phẩm, giá tiền, ảnh minh họa từ bảng PRODUCTS và hiển thị lên màn hình. Use case kết thúc.

* **Luồng rẽ nhánh:**
  + - 1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu thì hệt hống sẽ hiển thị thông báo lỗi và use case kết thúc.
* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống, có sản phẩm trong giỏ hàng trước khi thực hiện các use case
* **Hậu điều kiện:** Người dùng được đăng xuất thành công, không thể thực hiện các chức năng yêu cầu đăng nhập nếu chưa đăng nhập lại.
* **Điểm mở rộng:** Không có

#### Mô tả use case Thêm sản phẩm vào giỏ hàng

Use case này cho phép người dùng thêm sản phẩm vào giỏ hàng của mình.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người quản trị kích vào nút “+” trên sản phẩm muốn thêm vào giỏ hàng. Hệ thống lấy thông tin chi tiết của sản phẩm gồm: tên sản phẩm, mô tả, giá từ bảng SANPHAM trong cơ sở dữ liệu và hiển thị lên màn hình.
2. Tại đây người dùng có thể thay đổi số lượng sản phẩm muốn thêm vào giỏ hàng bằng cách kích vào nút “+” để tăng số lượng hoặc “-” để giảm số lượng. Sau khi thay đổi số lượng, người dùng kích vào nút “Thêm vào giỏ hàng”, hệ thống sẽ lưu thông tin sản phẩm gồm: id, tên sản phẩm, mô tả, giá sản phẩm, số lượng, tổng giá vào bảng GIOHANG.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 2 nếu người dùng kích vào nút “X” hoặc kích ra ngoài mà không kích nút “Thêm giỏ hàng” thì use case sẽ kết thúc.
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sử dữ liệu hoặc có lỗi phát sinh thì hệ thống sẽ hiển thị một thông báo ra màn hình và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng cần đăng nhập hệ thống trước khi có thể thực hiện use case
* **Hậu điều kiện:** Nếu use case kết thúc thành công thì thông tin sản phẩm sẽ được thêm và cập nhật vào trong cơ sở dữ liệu
* **Điểm mở rộng:** Không có

#### Mô tả use case Xóa sản phẩm khỏi giỏ hàng

Use case này cho phép người dùng xóa các sản phẩm đã thêm ra khỏi giỏ hàng của mình.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng kích vào nút “Giỏ hàng” trên thanh menu. Hệ thống sẽ lấy thông tin chi tiết của giỏ hàng bao gồm: Tên cửa hàng, tên sản phẩm, giá sản phẩm, số lượng, tổng giá sản phẩm, tổng giá giỏ hàng, thời gian giao và hiển thị lên màn hình.
2. Tại đây người dùng có thể kích nút “Xóa” ở sản phẩm cần xóa hoặc kích vào số lượng sản phẩm rồi kích vào nút “-” để giảm số lượng về 0 và kích nút “Cập nhật”. Hệ thống sẽ xóa sản phẩm ra khỏi giỏ hàng và hiển thị lại giỏ hàng.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sử dữ liệu hoặc có lỗi thì hệ thống sẽ hiển thị một thông báo ra màn hình và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng cần đăng nhập hệ thống trước khi có thể thực hiện use case
* **Hậu điều kiện:** Nếu use case kết thúc thành công thì thông tin giỏ hàng sẽ được cập nhật trong cơ sở dữ liệu
* **Điểm mở rộng:** Không có

#### Mô tả use case Hủy đơn hàng

Use case này cho phép người dùng hủy đơn hàng mà mình đã đặt

* **Luồng các sự kiện:**
* **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng bấm vào mục “Đơn mua” trong trang cá nhân hệ thống sẽ mặc định lấy thông tin tất cả đơn hàng bạn đã mua, các đơn hàng gồm tên cửa hàng, trạng thái đơn hàng, danh sách các sản phẩm, địa chỉ nhận hàng, ghi chú, thời gian tạo, phương thức thanh toán, tổng tiền trong cơ sở dữ liệu và hiển thị lên màn hình.
2. Người dùng tìm đơn hàng mình muốn hủy và nhấn nút hủy hệ thống sẽ cập nhật lại trạng thái đơn hàng là hủy và hiển thị thông báo.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 1 trong luồng cơ bản nếu không có đơn hàng nào trong cơ sở dữ liệu hệ thống sẽ hiển thị thông báo “Bạn hiện không có đơn hàng nào”, use case kết thúc.
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống trước khi thực hiện use case.
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Bật/tắt xác thực 2 lớp

Use case này cho phép người dùng bật tắt trạng thái bảo mật 2 lớp của tài khoản.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng bấm vào mục “Bảo mật 2 lớp” trong trang cá nhân hệ thống sẽ lấy khóa bí mật, mã QR từ cơ sở dữ liệu và hiển thị lên màn hình.
2. Nếu người dùng chưa bật bảo mật 2 lớp thì hệ thống yêu cầu nhập mã OTP ứng với khóa bí mật và nhấn vào nút bật hệ thống sẽ cập nhật lại trạng thái bảo mật của tài khoản là “Bật” và hiển thị thông báo.
3. Nếu người dùng đã bật bảo mật 2 lớp thì nút bật sẽ thay bằng nút tắt, người dùng nhấn vào nút tắt thì hệ thống sẽ cập nhật lại trạng thái bảo mật của tài khoản là “Tắt” và hiển thị thông báo.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 2 trong luồng cơ bản nếu nếu không nhập mã OTP hoặc nhập sai mã OTP mà nhấn nút bật thì hệ thống sẽ hiển thị thông báo.
2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu mất kết nối với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống trước khi thực hiện use case
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Quên mật khẩu

Use case này cho phép người dùng cấp lại mật khẩu mới khi quên mật khẩu cũ.

* **Luồng sự kiện:**
  + **Luồng cơ bản:**

1. Use case bắt đầu khi người dùng kích chọn “Khôi phục mật khẩu” tại màn hình Đăng nhập. Hệ thống điều hướng người dùng sang trang quên mật khẩu. Tại đây người dùng nhập email cần cấp lại mật khẩu mới.
2. Sau khi nhập email và mã captcha xong người dùng kích chọn “Gửi mã xác nhận” hệ thống sẽ gửi đến email một OTP 6 số, và điều hướng người dùng sang trang nhập mật khẩu mới.
3. Tại trang nhập mật khẩu, người dùng nhập mật khẩu mới cần cấp lại và xác nhận lại mật khẩu đó. Sau đó kích vào “Đặt lại” để cập nhật lại mật khẩu.

* **Luồng rẽ nhánh:**

1. Tại bước 2 trong luồng cơ bản, người dùng nhập email chưa tồn tại trong hệ thống, hệ thống sẽ hiển thị thông báo “Email không tồn tại”.
2. Tại bước 2 trong luồng cơ bản, nếu người dùng nhập sai mã OTP không giống OTP tại email thì hệ thống sẽ thông báo “OTP không chính xác”.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo “Không có dữ liệu”. Use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có.
* **Tiền điều kiện:** Không có.
* **Hậu điều kiện:** Không có.
* **Điểm mở rộng:** Không có.

#### Mô tả use case Nạp tiền vào tài khoản

Use case này cho phép người dùng nạp tiền vào tài khoản của mình.

* **Luồng các sự kiện:**
* **Luồng cơ bản:**

1. Use case này bắt đầu khi người dùng bấm vào mục “Nạp tiền” trong trang cá nhân hệ thống sẽ lấy nội dung chuyển khoản, mã QR từ cơ sở dữ liệu và hiển thị lên màn hình.
2. Người dùng chuyển số tiền muốn nạp với nội dung chuyển khoản được hiển thị trên màn hình đến số tài khoản được được hiển thị trên màn hình.
3. Hệ thống sẽ kiểm tra lịch sử giao dịch nếu giao dịch hợp lệ sẽ hiển thị thông báo nạp tiền thành công vào cập nhật lại số tiền.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 1 trong luồng cơ bản nếu không kết nối được với cơ sở dữ liệu hệ thống sẽ hiển thị thông báo và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống trước khi thực hiện use case.
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có.

#### Mô tả use case Chat với cửa hàng

Use case cho phép người dùng nhắn tin với các cửa hàng.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case bắt đầu khi người dùng nhấn vào icon tin nhắn trong một cửa hàng hoặc nút “Tin nhắn” trên trang chủ, hệ thống hiển thị danh sách các cuộc trò chuyện.
2. Khi người dùng kích chọn 1 cuộc trò chuyện, hệ thống sẽ lấy ra danh sách tin nhắn của cuộc trò chuyện đó và hiển thị lên màn hình.
3. Người dùng kích chọn một chức năng khác hoặc kích ra ngoài khung trò chuyện. Hệ thống sẽ điều hướng về trang chức năng khác và ẩn khung trò chuyện.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 1 khi người dung nhấn vào nút “Tin nhắn” mà chưa đăng nhập hệ thống sẽ chuyển hướng đến trang đăng nhập.
2. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo “Không có dữ liệu” và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không có
* **Tiền điều kiện:** Người dùng cần đăng nhập vào trang web trước khi sử dụng
* **Hậu điều kiện:** Không có
* **Điểm mở rộng:** Không có

#### Mô tả use case Quản lý loại sản phẩm

Use case cho phép quản trị viên thực hiện xem, tìm kiếm, thêm, sửa, xóa các loại sản phẩm.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case bắt đầu khi quản trị viên nhấn vào “Quản lý loại sản phẩm” tại trang quản lý. Hệ thống hiển thị danh sách loại sản phẩm bao gồm: mã thể loại, tên thể loại, ảnh, số lượng sản phẩm thuộc thể loại.
2. Xem chi tiết loại sản phẩm: Quản trị viên nhấn vào nút biểu tượng Xem chi tiết trên một dòng loại sản phẩm muốn xem. Hệ thống sẽ điều hướng màn hình sang trang Chi tiết thể loại và hiển thị các thông tin chi tiết của loại sản phẩm bao gồm: mã thể loại, tên thể loại, thời gian tạo, số lượng sản phẩm, mô tả, thời gian cập nhật, ảnh thể loại.
3. Thêm loại sản phẩm:
   * + - 1. Quản trị viên nhấn vào nút “Thêm thể loại”, Hệ thống điều hướng màn hình sang trang Thêm thể loại, hiển thị các trường nhập thông tin cho loại sản phẩm mới.
         2. Quản trị viên nhập vào thông tin cho loại sản phẩm mới bao gồm: mã thể loại, tên thể loại, mô tả, ảnh (tùy chọn) và nhấn vào nút “Thêm thể loại”. Hệ thống sẽ hiển thị thông báo thêm thể loại sản phẩm thành công, lưu thông tin loại sản phẩm mới vào bảng categories và cập nhật danh sách thể loại sản phẩm.
4. Sửa loại sản phẩm:
5. Quản trị viên nhấn vào nút biểu tượng sửa trên một dòng loại sản phẩm muốn thực hiện. Hệ thống điều hướng màn hình sang trang chỉnh sửa thể loại và lấy thông tin cũ của loại sản phẩm bao gồm: mã thể loại, tên thể loại, mô tả, ảnh hiển và hiển thị lên màn hình.
6. Quản trị viên nhập mới thông tin cho loại sản phẩm gồm: tên thể loại, mô tả, ảnh và nhấn vào nút “Chỉnh sửa”. Hệ thống hiển thị thông báo chỉnh sửa loại sản phẩm thành công, cập nhật lại trong bảng categories và hiển thị thông tin loại sản phẩm mới chỉnh sửa.
7. Xóa loại sản phẩm:
   1. Quản trị viên nhấn vào nút biểu tượng Xem chi tiết trên một dòng loại sản phẩm muốn xem. Hệ thống sẽ điều hướng màn hình sang trang Chi tiết thể loại và hiển thị các thông tin chi tiết của loại sản phẩm lên màn hình.
   2. Quản trị viên nhấn vào nút “Xóa” trên trang Chi tiết thể loại. Hệ thống sẽ hiển thị màn hình xác nhận xóa.
   3. Quản trị viên nhấn vào nút “Xác nhận”. Hệ thống sẽ hiển thị thông báo xóa loại sản phẩm thành công, xóa loại sản phẩm khỏi bảng categories và cập nhật lại danh sách hiển thị.
8. Tìm kiếm sản phẩm: Quản trị viên nhập vào từ khóa tìm kiếm và nhấn vào nút “Tìm kiếm”. Hệ thống sẽ hiển thị danh sách các loại sản phẩm phù hợp.

Use case kết thúc.

* **Luồng rẽ nhánh:**
  1. Tại bước 3b trong luồng cơ bản, nếu quản trị viên nhập vào mã loại đã tồn tại, hệ thống sẽ hiển thị thông báo “Mã loại đã tồn tại” và yêu cầu nhập lại. Quản trị viên có thể nhập lại để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc.
  2. Tại bước 3b hoặc 4b trong luồng cơ bản, nếu quản tị viên nhập thiếu một trong các trường thì hệ thống sẽ hiển thị thông báo “Vui lòng nhập đầy đủ mã và tên” và yêu cầu nhập lại. Quản trị viên có thể nhập lại để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc.
  3. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo “Không có dữ liệu” và use case kết thúc.
* **Các yêu cầu đặc biệt:** Use case này chỉ cho phép một số vai trò như quản trị viên, người chủ hệ thống thực hiện
* **Tiền điều kiện:** Quản trị viên cần đăng nhập vào trang quản trị trước khi thực hiện
* **Hậu điều kiện:** Nếu use case kết thúc thành công thì thông tin về loại sản phẩm sẽ được cập nhật trong cơ sở dữ liệu
* **Điểm mở rộng:** Không có

#### Mô tả use case Quản lý sản phẩm

Use case cho phép quản trị viên thực hiện xem, tìm kiếm, thêm, sửa, xóa các sản phẩm.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Use case bắt đầu khi quản trị viên nhấn vào “Quản lý sản phẩm” tại trang quản lý. Hệ thống hiển thị danh sách sản phẩm bao gồm: mã sản phẩm, tên sản phẩm, ảnh, mô tả, thể loại, số lượng bán, giá, trạng thái, ngày tạo.
2. Xem chi tiết sản phẩm: Quản trị viên nhấn vào nút biểu tượng Xem chi tiết trên một dòng sản phẩm muốn xem. Hệ thống sẽ điều hướng màn hình sang trang Chi tiết sản phẩm và hiển thị các thông tin chi tiết của sản phẩm bao gồm: mã sản phẩm, tên sản phẩm, giá sản phẩm, thương hiệu, loại sản phẩm, mô tả sản phẩm và ảnh.
3. Thêm sản phẩm:
   * + - 1. Quản trị viên nhấn vào nút “Thêm sản phẩm”, Hệ thống điều hướng màn hình sang trang Thêm sản phẩm, hiển thị các trường nhập thông tin cho sản phẩm mới.
         2. Quản trị viên nhập vào thông tin cho sản phẩm mới bao gồm: mã sản phẩm, tên sản phẩm, giá sản phẩm, mô tả sản phẩm, ảnh và chọn loại sản phẩm từ danh sách, sau đó nhấn vào nút “Thêm sản phẩm”. Hệ thống sẽ hiển thị thông báo thêm sarn phẩm thành công, lưu thông tin sản phẩm mới vào bảng products và cập nhật danh sách sản phẩm.
4. Sửa sản phẩm:
   1. Quản trị viên nhấn vào nút biểu tượng sửa trên một dòng sản phẩm muốn thực hiện. Hệ thống điều hướng màn hình sang trang Chỉnh sửa sản phẩm và lấy thông tin cũ của sản phẩm hiển thị lên màn hình bao gồm: mã sản phẩm, tên sản phẩm, giá sản phẩm, loại sản phẩm, thương hiệu, mô tả sản phẩm và ảnh.
   2. Quản trị viên nhập mới thông tin cho sản phẩm gồm: tên sản phẩm, giá sản phẩm, loại sản phẩm, thương hiệu, mô tả sản phẩm và ảnh và nhấn vào nút “Chỉnh sửa”. Hệ thống hiển thị thông báo chỉnh sửa sản phẩm thành công, cập nhật lại trong bảng products và hiển thị thông tin sản phẩm mới chỉnh sửa.
5. Xóa sản phẩm:
   1. Quản trị viên nhấn vào nút biểu tượng xóa trên một dòng sản phẩm muốn thực hiện. Hệ thống sẽ hiển thị màn hình xác nhận xóa sản phẩm.
   2. Quản trị viên nhấn vào nút “Xác nhận”. Hệ thống sẽ hiển thị thông báo xóa sản phẩm thành công, xóa sản phẩm khỏi bảng products và cập nhật lại danh sách hiển thị.
6. Tìm kiếm sản phẩm: Quản trị viên nhập vào từ khóa tìm kiếm hoặc chọn theo thể loại và nhấn vào nút “Tìm kiếm”. Hệ thống sẽ hiển thị danh sách các sản phẩm phù hợp.

Use case kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 3b trong luồng cơ bản, nếu quản trị viên nhập vào mã sản phẩm đã tồn tại, hệ thống sẽ hiển thị thông báo “Mã sản phẩm đã tồn tại” và yêu cầu nhập lại. Quản trị viên có thể nhập lại để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc.
2. Tại bước 3b hoặc 4b trong luồng cơ bản, nếu quản tị viên nhập thiếu một trong các trường thì hệ thống sẽ hiển thị thông báo “Vui lòng nhập đầy đủ thông tin” và yêu cầu nhập lại. Quản trị viên có thể nhập lại để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc.
3. Tại các bước 3b, 4b hoặc 5b trong luồng cơ bản, nếu quản trị viên kích vào nút “Hủy bỏ” thì hệ thống sẽ bỏ qua thao tác đang thực hiện và kết thúc.
4. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo “Không có dữ liệu” và use case kết thúc.

* **Các yêu cầu đặc biệt:** Use case này chỉ cho phép một số vai trò như quản trị viên, người chủ hệ thống thực hiện
* **Tiền điều kiện:** Quản trị viên cần đăng nhập vào trang quản trị trước khi thực hiện
* **Hậu điều kiện:** Nếu use case kết thúc thành công thì thông tin về sản phẩm sẽ được cập nhật trong cơ sở dữ liệu
* **Điểm mở rộng:** Không có

#### Quản lý người dùng

Use case cho phép quản trị viên thực hiện xem, tìm kiếm, thêm, sửa, khóa/mở khóa, xóa người dùng.

* **Luồng sự kiện:**
* **Luồng cơ bản:**
  1. Use case bắt đầu khi quản trị viên nhấn vào “Quản lý người dùng” tại trang quản lý. Hệ thống hiển thị danh sách người dùng bao gồm: họ và tên, email, số điện thoại, vai trò, giới tính, trạng thái, ngày tạo.
  2. Xem chi tiết người dùng: Quản trị viên nhấn vào nút biểu tượng Xem chi tiết trên một dòng người dùng muốn xem. Hệ thống sẽ điều hướng màn hình sang trang Chi tiết người dùng và hiển thị các thông tin chi tiết của đơn hàng bao gồm: tên người dùng, email, số điện thoại, giới tính, trạng thái khóa, vai trò và ảnh đại diện.
  3. Thêm người dùng:
  4. Quản trị viên nhấn vào nút “Thêm người dùng”, Hệ thống điều hướng màn hình sang trang Thêm người dùng, hiển thị các trường nhập thông tin cho người dùng mới.
  5. Quản trị viên nhập vào thông tin cho người dùng mới bao gồm: tên người dùng, email, mật khẩu, số điện thoại, chọn trạng thái khóa và vai trò, sau đó nhấn vào nút “Thêm người dùng”. Hệ thống sẽ hiển thị thông báo thêm người dùng thành công, lưu thông tin sản phẩm mới vào bảng users và cập nhật danh sách người dùng.
  6. Sửa thông tin người dùng:
     1. Quản trị viên nhấn vào nút biểu tượng sửa trên một dòng người dùng muốn thực hiện. Hệ thống điều hướng màn hình sang trang Chỉnh sửa người dùng và lấy thông tin cũ của người dùng hiển thị lên màn hình bao gồm: tên người dùng, email, mật khẩu, số điện thoại, trạng thái khóa và vai trò.
     2. Quản trị viên nhập mới thông tin cho người dùng gồm: tên người dùng, số điện thoại, chọn trạng thái khóa và vai trò và nhấn vào nút “Chỉnh sửa”. Hệ thống hiển thị thông báo chỉnh sửa người dùng thành công, cập nhật lại trong bảng users và hiển thị thông tin người dùng mới chỉnh sửa.
  7. Xóa người dùng:
     1. Quản trị viên nhấn vào nút biểu tượng xóa trên một dòng người dùng muốn thực hiện. Hệ thống sẽ hiển thị màn hình xác nhận xóa người dùng.
     2. Quản trị viên nhấn vào nút “Xác nhận”. Hệ thống sẽ hiển thị thông báo xóa người dùng thành công, xóa người dùng khỏi bảng users và cập nhật lại danh sách hiển thị.
  8. Khóa/Mở khóa:
     1. Quản trị viên nhấn vào nút biểu khóa/mở khóa trên một dòng người dùng muốn thực hiện. Hệ thống sẽ hiển thị màn hình xác nhận khóa/mở khóa người dùng.
     2. Quản trị viên nhấn vào nút “Xác nhận”. Hệ thống sẽ hiển thị thông báo khóa/mở khóa người dùng thành công và cập nhật trạng thái tài khoản trong bảng users và cập nhật lại danh sách hiển thị.
  9. Tìm kiếm người dùng: Quản trị viên nhập vào từ khóa tìm kiếm hoặc chọn theo vai trò, giới tính hoặc trạng thái tài khoản và nhấn vào nút “Tìm kiếm”. Hệ thống sẽ hiển thị danh sách người dùng phù hợp.

Use case kết thúc.

* **Luồng rẽ nhánh:**
  1. Tại bước 3b trong luồng cơ bản, nếu quản trị viên nhập vào mã sản phẩm đã tồn tại, hệ thống sẽ hiển thị thông báo “Mã sản phẩm đã tồn tại” và yêu cầu nhập lại. Quản trị viên có thể nhập lại để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc.
  2. Tại bước 3b hoặc 4b trong luồng cơ bản, nếu quản tị viên nhập thiếu một trong các trường thì hệ thống sẽ hiển thị thông báo “Vui lòng nhập đầy đủ thông tin” và yêu cầu nhập lại. Quản trị viên có thể nhập lại để tiếp tục hoặc kích vào nút “Hủy bỏ” để kết thúc.
  3. Tại bước 3b, 4b, 5b hoặc 6b trong luồng cơ bản, nếu quản trị viên kích vào nút “Hủy bỏ” thì hệ thống sẽ bỏ qua các thao tác đang thực hiện và kết thúc.
  4. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo “Không có dữ liệu” và use case kết thúc.
* **Các yêu cầu đặc biệt:** Use case này chỉ cho phép một số vai trò như quản trị viên, người chủ hệ thống thực hiện
* **Tiền điều kiện:** Quản trị viên cần đăng nhập vào trang quản trị trước khi thực hiện
* **Hậu điều kiện:** Nếu use case kết thúc thành công thì thông tin về người dùng sẽ được cập nhật trong cơ sở dữ liệu
* **Điểm mở rộng:** Không có

#### Mô tả use case Quản lý đơn hàng

Use case cho phép quản trị viên thực hiện xem, tìm kiếm, cập nhật trạng thái đơn hàng.

* **Luồng sự kiện:**
* **Luồng cơ bản:**
  1. Use case bắt đầu khi quản trị viên nhấn vào “Quản lý đơn hàng” tại trang quản lý. Hệ thống hiển thị danh sách đơn hàng bao gồm: mã đơn hàng, người nhận, số điện thoại, tổng đơn hàng, người giao hàng, phương thức thanh toán, trạng thái thanh toán, ngày tạo, trạng thái đơn hàng.
  2. Xem chi tiết đơn hàng: Quản trị viên nhấn vào nút biểu tượng Xem chi tiết trên một dòng đơn hàng muốn xem. Hệ thống sẽ điều hướng màn hình sang trang Chi tiết đơn hàng và hiển thị các thông tin chi tiết của đơn hàng bao gồm: mã đơn hàng, trạng thái, hình thức thanh toán, trạng thái thanh toán, tên người nhận, số điện thoại, địa chỉ, tổng tiền, chú tích và bảng danh sách sản phẩm trong đơn hàng.
  3. Cập nhật trạng thái đơn hàng:
     1. Quản trị viên nhấn vào nút biểu tượng cập nhật trạng thái (xác nhận/ đang giao/ hoàn thành/ từ chối) trên một dòng đơn hàng. Hệ thống sẽ hiển thị một màn hình xác nhận cập nhật trạng thái đơn hàng.
     2. Quản trị viên nhấn vào nút “Xác nhận”. Hệ thống sẽ hiển thị thông báo thành công, cập nhật trong cơ sở dữ liệu và cập nhật danh sách hiển thị.
  4. Tìm kiếm đơn hàng: Quản trị viên nhập vào từ khóa tìm kiếm hoặc chọn theo trạng thái đơn hàng, phương thức thanh toán, trạng thái thanh toán và nhấn vào nút “Tìm kiếm”. Hệ thống sẽ hiển thị danh sách đơn hàng phù hợp.

Use case kết thúc.

* **Luồng rẽ nhánh:**
  1. Tại bước 3b trong luồng cơ bản, nếu quản trị viên kích vào nút “Hủy bỏ” thì hệ thống sẽ bỏ qua các thao tác đang thực hiện và kết thúc.
  2. Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo “Không có dữ liệu” và use case kết thúc.
* **Các yêu cầu đặc biệt:** Use case này chỉ cho phép một số vai trò như quản trị viên, người chủ hệ thống thực hiện
* **Tiền điều kiện:** Quản trị viên cần đăng nhập vào trang quản trị trước khi thực hiện
* **Hậu điều kiện:** Nếu use case kết thúc thành công thì thông tin về đơn hàng sẽ được cập nhật trong cơ sở dữ liệu
* **Điểm mở rộng:** Không có

#### Mô tả use case Thống kê

Use case cho phép quản trị viên thực hiện xem thống kê doanh thu, số đơn hàng, tỉ lệ trạng thái đơn hàng.

* **Luồng sự kiện:**
* **Luồng cơ bản:**
  1. Use case bắt đầu khi quản trị viên nhấn vào “Thống kê dashboard” tại trang quản lý. Hệ thống hiển thị các biểu đồ thống kê bao gồm: doanh thu theo tháng, tỉ lệ sản phẩm bán chạy, lượt truy cập, phản hổi của khách hàng, đơn hàng theo khu vực, thị phần thương hiệu.

Use case kết thúc.

* **Luồng rẽ nhánh:**
  1. Trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo “Không có dữ liệu” và use case kết thúc.
* **Các yêu cầu đặc biệt:** Use case này chỉ cho phép một số vai trò như quản trị viên, người chủ hệ thống thực hiện
* **Tiền điều kiện:** Quản trị viên cần đăng nhập vào trang quản trị trước khi thực hiện
* **Hậu điều kiện:**
* **Điểm mở rộng:** Không có

### Các yêu cầu phi chức năng

Bảng 3.1: Yêu cầu phi chức năng của website Haui Food

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Nội dung** | **Mô tả** | **Ghi chú** |
| NF01 | Giao diện | - Ngôn ngữ tiếng Việt, Anh, Trung  - Cung cấp một giao diện đơn giản, gần gũi, trực quan và dễ sử dụng đối với người dùng |  |
| NF02 | Bảo mật | - Xử lý hầu hết các trường hợp lỗi, có hệ thống ghi log để xử lý thêm |  |
| NF03 | Tốc độ xử lý | - Nhanh chóng đáp ứng yêu cầu người dùng |  |
| NF04 | Thời gian hoạt động | - Có thể hoạt động hiệu quả 24/7 |  |
| NF05 | Yêu cầu thực thi | - Người dùng có thể hoạt động đồng thời với nhiều vai trò khác nhau |  |

## Kế hoạch kiểm thử

#### Giới thiệu:

* Tên dự án: Website đặt đồ ăn trực tuyến Haui Food
* Mô tả: Website thương mại điện tử dành cho các cán bộ nhân viên và sinh viên trường Đại học Công nghiệp Hà Nội tìm kiếm, xem chi tiết và đặt hàng đồ ăn, thức uống.
* Mục tiêu kiểm thử:

+ Đảm bảo các chức năng chính của website hoạt động theo đúng yêu cầu

+ Kiểm tra tính ổn định và hiệu suất của hệ thống

+ Xác định lỗi tiềm ẩn trước khi triển khai

* **Phạm vi kiểm thử:**
* Các chức năng kiểm thử tự động:

+ Đăng ký, đăng nhập, đăng xuất

+ Quản lý tài khoản cá nhân (xem thông tin tài khoản, cập nhật thông tin, đổi mật khẩu)

+ Tìm kiếm sản phẩm, xem chi tiết sản phẩm

+ Thêm/xóa sản phẩm vào giỏ hàng

+ Đặt hàng, hủy đơn hàng

+ Xem danh sách đơn hàng

* Ngoài phạm vi kiểm thử:

+ Nạp tiền vào tài khoản

+ Chat với cửa hàng

+ Quên mật khẩu

+ Bật/tắt xác thực 2 lớp

* **Chiến lược kiểm thử:**
* Phương pháp kiểm thử: Kiểm thử tự động với Selenium WebDriver
* Mức độ kiểm thử: Kiểm thử hệ thống
* Loại hình kiểm thử: Kiểm thử chức năng
* **Môi trường kiểm thử:**
* Hệ điều hành: Windows 11
* Trình duyệt: Chrome
* Công cụ kiểm thử: Selenium WebDriver, thư viện TestNG, Allure Report
* Ngôn ngữ sử dụng: Java
* **Kế hoạch thực hiện:**

Bảng 3.2: Kế hoạch thực hiện kiểm thử

|  |  |  |  |
| --- | --- | --- | --- |
| **Tuần** | **Thời gian** | **Công việc** | **Mô tả** |
| 1 | 31/03/2025 – 06/04/2025 | * Thiết kế ca kiểm thử * Chuẩn bị môi trường vả thiết lập framework | * Thiết kế test case dựa trên yêu cầu phần mềm đã phân tích * Cài đặt Selenium WebDriver, thư viện TestNG, Allure Report * Cấu trúc thư mục theo mô hình Page Object Model (POM) * Kết nối GitHub |
| 2 | 07/04/2025 –  13/04/2025 | Viết script và chạy test cho nhóm chức năng xác thực người dùng | Viết script cho chức năng Đăng ký, Đăng nhập, Đăng xuất |
| 3 | 14/04/2025 –  20/04/2025 | Viết script và chạy test cho các chức năng mua hàng | Viết script cho chức năng Tìm kiếm, Xem chi tiết sản phẩm, Thêm/xóa sản phẩm khỏi giỏ hàng, Đặt hàng, Hủy đơn hàng |
| 4 | 21/04/2025 –  27/04/2025 | Viết script và chạy test cho nhóm chức năng theo dõi đơn hàng và quản lý tài khoản cá nhân | Viết script cho chức năng Xem danh sách đơn hàng, Xem thông tin tài khoản, Cập nhật thông tin tài khoản, Đổi mật khẩu, Bật/tắt bảo mật 2 lớp |
| 5 | 28/04/2025 –  04/05/2025 | Chạy kiểm thử toàn bộ hệ thống, báo cáo kết quả và đánh giá | Chạy lại toàn bộ test case, tối ưu script  Tạo báo cáo kiểm thử và đánh giá kết quả |

* **Rủi ro và giải pháp:**

Bảng 3.3: Rủi ro và giải pháp khi thực hiện kiểm thử

|  |  |  |
| --- | --- | --- |
| **STT** | **Rủi ro** | **Giải pháp** |
| 1 | Website thay đổi giao diện gây lỗi script | Dùng XPath linh hoạt, cập nhật code thường xuyên |
| 2 | Selenium không nhận diện được một số phần tử | Chờ element bằng WebDriverWait |
| 3 | Môi trường kiểm thử không ổn định | Chạy trên nhiều trình duyệt khác nhau |

## Thiết kế kịch bản kiểm thử

### Test case cho chức năng Đăng ký

Bảng 3.4: Test case chức năng Đăng ký

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | | |
| **Test Case** | | | | | | | | |
| **Mức độ ưu tiên:** Cao | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | | |
| **Tên chức năng:** Đăng ký | | | | **Ngày thiết kế:** 01/04/2025 | | | | |
| **Mô tả:** Kiểm thử chức năng đăng ký tài khoản mới cho người dùng | | | | **Ngày thực hiện:** 07/04/2025 | | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| DK-1 | Kiểm tra truy cập màn hình Đăng ký |  | 1.Tại trang chủ, nhấn nút “Đăng nhập/Đăng ký”  2. Tại trang Đăng nhập, nhấn vào “Đăng ký” | |  | 2. Điều hướng màn hình sang trang Đăng ký | Đạt |  |
| DK-2 | Đăng ký tài khoản thành công khi nhập hợp lệ tất cả các trường | Email đăng ký chưa tồn tại trong cơ sở dữ liệu | Tại trang Đăng ký:  1. Nhập họ tên  2. Nhập Email  3. Nhập mật khẩu  4. Nhấn Đăng ký | | Họ tên: Nhung  Email: nhungnt@gmail.com  Mật khẩu: #Nhung123 | 4. Hiển thị thông báo Đăng ký thành công | Đạt |  |
| DK-3 | Đăng ký thất bại khi nhập Email đã đăng ký | Email đã đăng ký tài khoản và tồn tại trong cơ sở dữ liệu | Tại trang Đăng ký 1. Nhập họ tên 2. Nhập Email 3. Nhập mật khẩu 4. Nhấn nút Đăng ký | | Họ tên: Nhung  Email: kimngan@gmail.com  Mật khẩu: #Nhung124 | 4. Hiển thị thông báo “Email đã tồn tại” | Đạt |  |
| DK-4 | Kiểm tra thông báo lỗi khi bỏ trống các trường |  | 1. Nhấn chuột vào trường Họ tên  2. Để trống Họ tên -> Nhấn chuột vào trường Email  3. Để trống Email -> Nhấn chuột vào trường Mật khẩu  4. Để trống Mật khẩu -> Nhấn chuột vào Hiển thị mật khẩu | |  | 2. Hiển thị thông báo "Vui lòng nhập họ tên"  3. Hiển thị thông báo "Vui lòng nhập email"  4. Hiển thị thông báo "Vui lòng nhập mật khẩu" | Đạt |  |
| DK-5 | Kiểm tra thông báo lỗi khi nhập email sai định dạng |  | 1. Nhập email sai định dạng  2. Nhấn chuột vào trường Mật khẩu | | Email: abc.com (hoặc: 123/ abc!@#) | 2. Hiển thị thông báo "Email không đúng, vui lòng nhập lại" | Đạt |  |
| DK-6 | Kiểm tra thông báo lỗi khi nhập mật khẩu sai định dạng |  | 1. Nhập họ tên  2. Nhập email  3. Nhập mật khẩu sai định dạng  4. Nhấn Hiển thị mật khẩu | | Mật khẩu: m | 4. Hiển thị thông báo "Mật khẩu phải chứa ít nhất 8 ký tự và 1 ký tự đặc biệt @-\_" | Đạt |  |

### Test case cho chức năng Đăng nhập

Bảng 3.5: Test case cho chức năng Đăng nhập

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | | |
| **Test Case** | | | | | | | | |
| **Mức độ ưu tiên:** Cao | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | | |
| **Tên chức năng:** Đăng nhập | | | | **Ngày thiết kế:** 01/04/2025 | | | | |
| **Mô tả:** Kiểm thử chức năng đăng nhập cho người dùng | | | | **Ngày thực hiện:** 09/04/2025 | | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| DN-1 | Đăng nhập thành công với email và mật khẩu hợp lệ | Tài khoản đã tồn tại trong cơ cở dữ liệu | Tại trang Đăng nhập:  1. Nhập email  2. Nhập mật khẩu  3. Nhấn nút Đăng nhập | | Email: trangnhung29072003@gmail.com  Mật khẩu: #Nhung123 | 3.1 User đăng nhập thành công, hiển thị thông báo "Đăng nhập thành công"  3.2 Điều hướng màn hình về trang chủ | Đạt |  |
| DN-2 | Đăng nhập thất bại khi nhập sai email | Email không tồn tại trong cơ sở dữ liệu | 1. Nhập sai email  2. Nhập mật khẩu  3. Nhấn nút Đăng nhập | | Họ tên: Nhung  Email: n@gmail.com  Mật khẩu: #Nhung123 | 3. Hiển thị thông báo "Tài khoản hoặc mật khẩu không chính xác" | Đạt |  |
| DN-3 | Đăng nhập thất bại khi nhập đúng email nhưng sai mật khẩu | Email đã tồn tại trong cơ sở dữ liệu | 1. Nhập email  2. Nhập sai mật khẩu  3. Nhấn nút Đăng nhập | | Email: trangnhung29072003@gmail.com  Mật khẩu: #Nhung12 | 3. Hiển thị thông báo "Tài khoản hoặc mật khẩu không chính xác" | Đạt |  |
| DN-4 | Kiểm tra thông báo lỗi khi bỏ trống trường email |  | 1. Nhấn chuột vào trường Email  2. Để trống Email -> Nhấn chuột vào trường Mật khẩu | |  | 2. Hiển thị thông báo "Vui lòng nhập email" | Đạt |  |
| DN-5 | Kiểm tra thông báo lỗi khi bỏ trống mật khẩu |  | 1. Nhập email  2.Nhấn chuột vào trường Mật khẩu  3. Để trống Mật khẩu -> Nhấn ra khỏi form | |  | 3. Hiển thị thông báo "Vui lòng nhập mật khẩu" | Đạt |  |
| DN-6 | Kiểm tra thông báo lỗi khi nhập email sai định dạng |  | 1. Nhập email sai định dạng  2. Nhấn chuột vào trường Mật khẩu | | Email: abc.com (hoặc: 123/ abc!@#) | 2. Hiển thị thông báo "Email không đúng, vui lòng nhập lại" | Đạt |  |
| DN-7 | Kiểm tra thông báo lỗi khi nhập mật khẩu sai định dạng |  | 1. Nhập email  2. Nhập mật khẩu sai định dạng  3. Nhấn Hiển thị mật khẩu | | Email: trangnhung29072003@gmail.com  Mật khẩu: m | 3. Hiển thị thông báo "Mật khẩu phải chứa ít nhất 8 ký tự và 1 ký tự đặc biệt @-\_" | Đạt |  |

### Test case cho chức năng Đăng xuất

Bảng 3.6: Test case cho chức năng Đăng xuất

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | | |
| **Test Case** | | | | | | | | |
| **Mức độ ưu tiên:** Trung bình | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | | |
| **Tên chức năng:** Đăng xuất | | | | **Ngày thiết kế:** 01/04/2025 | | | | |
| **Mô tả:** Kiểm thử chức năng đăng xuất khỏi phiên đăng nhập cho người dùng | | | | **Ngày thực hiện:** 11/04/2025 | | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| DX-1 | Đăng xuất thành công khỏi phiên đăng nhập | Đã đăng nhập thành công và phiên còn hiệu lực | 1. Nhấn icon user trên góc phải màn hình  2. Nhấn Đăng xuất | | Email: trangnhung29072003@gmail.com  Mật khẩu: #Nhung123 | 2. Hiển thị thông báo "Đăng xuất thành công", điều hướng sang màn hình trang chủ | Đạt |  |

### Test case cho chức năng Tìm kiếm sản phẩm

Bảng 3.7: Test case cho chức năng Tìm kiếm sản phẩm

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | | |
| **Test Case** | | | | | | | | |
| **Mức độ ưu tiên:** Trung bình | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | | |
| **Tên chức năng:** Tìm kiếm sản phẩm | | | | **Ngày thiết kế:** 01/04/2025 | | | | |
| **Mô tả:** Kiểm thử chức năng tìm kiếm cho người dùng khi nhập vào thông tin tìm kiếm | | | | **Ngày thực hiện:** 14/04/2025 | | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| TK-1 | Tìm kiếm món ăn thành công khi nhập đúng tên món | Tên món tồn tại trong cơ sở dữ liệu | 1. Trên thanh tìm kiếm, nhập đúng thông tin món ăn cần tìm  2. Nhấn nút Tìm kiếm | | Nhập: Bánh mì chả | 2. Hiển thị danh sách các sản phẩm có thông tin phù hợp | Đạt |  |
| TK-2 | Tìm kiếm món ăn thành công khi nhập gần đúng tên món ăn |  | 1. Nhập một từ của thông tin món ăn cần tìm vào ô tìm kiếm  2. Nhấn nút Tìm kiếm | | Nhập: Bánh mì | 2. Hiển thị danh sách các sản phẩm có thông tin phù hợp | Đạt |  |
| TK-3 | Tìm kiếm món ăn không thành công khi món ăn không tồn tại | Món ăn không tồn tại trong cơ sở dữ liệu | 1. Nhập thông tin món ăn cần tìm vào ô tìm kiếm  2. Nhấn nút Tìm kiếm | | Nhập: bánh mỳ | 2. Hiển thị thông báo "Rất tiếc, hiện không có món ăn nào" | Đạt |  |

### Test case cho chức năng Xem chi tiết sản phẩm

Bảng 3.8: Test case cho chức năng Xem chi tiết sản phẩm

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | | |
| **Test Case** | | | | | | | | |
| **Mức độ ưu tiên:** Trung bình | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | | |
| **Tên chức năng:** Xem chi tiết sản phẩm | | | | **Ngày thiết kế:** 01/04/2025 | | | | |
| **Mô tả:** Kiểm thử chức năng tìm kiếm cho người dùng khi nhập vào thông tin tìm kiếm | | | | **Ngày thực hiện:** 15/04/2025 | | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| CT-1 | Xem chi tiết món ăn thành công |  | 1. Nhấn vào một món ăn đang hiển thị | |  | 1. Hiển thị đúng thông tin món ăn đã chọn bao gồm: tên sản phẩm, mô tả, giá tiền, ảnh minh họa | Đạt |  |

### Test case cho cụm chức năng Quản lý giỏ hàng

Bảng 3.9: Test case cho cụm chức năng Quản lý giỏ hàng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Cao | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Cụm chức năng Quản lý giỏ hàng | | | | **Ngày thiết kế:** 02/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng quản lý giỏ hàng cho người dùng bao gồm: xem giỏ hàng, thêm/xóa sản phẩm khỏi giỏ hàng | | | | **Ngày thực hiện:** 16/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| GH-1 | Xem giỏ hàng thành công khi giỏ hàng trống | Đã đăng nhập vào hệ thống | Tại trang chủ website:  1. Nhấn vào biểu tượng Giỏ hàng | Email: haui@gmail.com  Mật khẩu: #Haui123 | 1. Hiển thị màn hình xem giỏ hàng và hiển thị text "Giỏ hàng trống" | Đạt |  |
| GH-2 | Xem giỏ hàng khi chưa đăng nhập |  | Tại trang chủ website:  1. Nhấn vào biểu tượng Giỏ hàng |  | 1. Hiển thị thông báo "Chưa đăng nhập" | Đạt |  |
| GH-3 | Thêm sản phẩm vào giỏ hàng thành công | Đã đăng nhập vào hệ thống | 1. Nhấn xem chi tiết món ăn  2. Nhấn vào nút "Thêm vào giỏ hàng"  3. Nhấn vào biểu tượng giỏ hàng | Email: kimngan@gmail.com  Mật khẩu: #Ngan123 | 2. Hiển thị thông báo "Thêm sản phẩm vào giỏ hàng thành công"  3.1 Biểu tượng giỏ hàng có số lượng mặt hàng trong giỏ  3.2 Hiển thị màn chi tiết giỏ hàng | Đạt |  |
| GH-4 | Thêm sản phẩm vào giỏ thất bại khi chưa đăng nhập |  | 1. Nhấn xem chi tiết món ăn  2. Nhấn vào nút "Thêm vào giỏ hàng" |  | 2. Hiển thị thông báo "Vui lòng đăng nhập để thêm vào giỏ hàng" | Đạt |  |
| GH-5 | Xóa một sản phẩm khỏi giỏ hàng | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn vào biểu tượng giỏ hàng  2. Nhấn nút Xóa trên 1 dòng sản phẩm | Email: domai@gmail.com  Mật khẩu: #Mai12345 | 2.1 Hiển thị thông báo "Thành công"  2.2 Số lượng sản phẩm trong giỏ giảm đi 1, tổng tiền giảm đúng theo sản phẩm đã xóa | Đạt |  |
| GH-6 | Xóa toàn bộ giỏ hàng thành công | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn vào biểu tượng giỏ hàng  2. Nhấn nút Xóa tất cả  3. Nhấn vào biểu tượng giỏ hàng | Email: trangnhung29072003@gmail.com  Mật khẩu: #Nhung123 | 2. Hiển thị thông báo "Thành công"  3. Hiển thị "giỏ hàng trống" | Đạt |  |
| GH-7 | Cập nhật số lượng sản phẩm trong giỏ hàng | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn vào biểu tượng giỏ hàng  2. Nhấn nút số lượng sản phẩm  3. Nhấn vào nút "+"/"-"  4. Nhấn nút "Cập nhật" | Email: hado@gmail.com  Mật khẩu: #Nhung123 | 4.1 Điều hướng về màn chi tiết giỏ hàng  4.2 Hiển thị thông báo "Thành công"  4.3 Số lượng sản phẩm và tổng tiền trong giỏ hàng được cập nhật theo | Đạt |  |
| GH-8 | Cập nhật số lượng sản phẩm trong giỏ nhưng giữ nguyên số lượng hiện tại | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn vào biểu tượng giỏ hàng  2. Nhấn nút số lượng sản phẩm  3. Nhấn nút "Cập nhật" | Email: hado@gmail.com  Mật khẩu: #Nhung123 | 3. Hiển thị thông báo "Số lượng không thay đổi" | Thất bại |  |
| GH-9 | thêm sản phẩm thất bại khi chọn số lượng vượt quá số lượng có thể cung cấp | Đã đăng nhập vào hệ thống | Tại trang chủ website:  1. Nhấn xem chi tiết món ăn  2. Nhấn nút "+" để tăng số lượng sản phẩm lên tới 100 |  | 2. Hiển thị thông báo "Số lượng sản phẩm không được vượt quá 100 sản phẩm" | Đạt |  |

### Test case cho chức năng Đặt hàng

Bảng 3.10: Test case cho chức năng Đặt hàng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Cao | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Đặt hàng | | | | **Ngày thiết kế:** 04/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng đặt hàng cho người dùng bao gồm: tiến hành mua hàng từ giỏ hàng, lựa chọn hình thức thanh toán và thanh toán | | | | **Ngày thực hiện:** 19/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| DH-1 | Mua hàng thành công với hình thức thanh toán khi nhận hàng | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn xem giỏ hàng  2. Nhấn nút "Thanh toán"  3. Nhập thông tin địa chỉ giao hàng  4. Chọn hình thức thanh toán khi nhận hàng  5. Nhấn nút "Đặt hàng" | Email: test@gmail.com  Mật khẩu: #Haui123 | 5. Hiển thị thông báo Đặt hàng thành công | Đạt |  |
| DH-2 | Mua hàng thành công khi thanh toán bằng ví điện tử Haui Food | Đã đăng nhập vào hệ thống,giỏ hàng có sản phẩm, số dư lớn hơn giá trị đơn hàng | 1. Nhấn xem giỏ hàng  2. Nhấn nút "Thanh toán"  3. Nhập thông tin địa chỉ giao hàng  4. Chọn hình thức thanh toán bằng ví Haui Food  5. Nhấn nút "Đặt hàng" | Email: trangnhung29072003@gmail.com  Mật khẩu: #Nhung123 | 5.1 Hiển thị thông báo Đặt hàng thành công  5.2 Cập nhật đúng số dư ví Haui Food sau khi thanh toán | Đạt |  |
| DH-3 | Mua hàng thất khi thanh toán bằng ví điện tử Haui Food nhưng số dư không đủ | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn xem giỏ hàng  2. Nhấn nút "Thanh toán"  3. Nhập thông tin địa chỉ giao hàng  4. Chọn hình thức thanh toán bằng ví Haui Food  5. Nhấn nút "Đặt hàng" | email: test@gmail.com mật khẩu: #Nhung123 | 5. Hiển thị thông báo "Số dư tài khoản không đủ để thực hiện thanh toán" | Đạt |  |
| DH-4 | Mua hàng thành công khi thanh toán chuyển khoản | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn xem giỏ hàng  2. Nhấn nút "Thanh toán"  3. Nhập thông tin địa chỉ giao hàng  4. Chọn hình thức thanh toán trực tuyến VNPay  5. Nhấn nút "Đặt hàng" | email: test@gmail.com mật khẩu: #Nhung123 | 5.1 Điều hướng sang màn hình lựa chọn phương thức thanh toán  5.2 Thanh toán thành công sẽ điều hướng về màn hình Đặt hàng thành công | Thất bại |  |
| DH-5 | Kiểm tra thông báo khi nhập số điện thoại không hợp lệ về độ dài | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn xem giỏ hàng  2. Nhấn nút "Thanh toán"  3. Nhập sđt không hợp lệ | Nhập số điện thoại: 0 | 3. Hiển thị thông báo "Số điện thoại không hợp lệ" | Đạt |  |
| DH-6 | Kiểm tra thông báo khi nhập số điện thoại không hợp lệ | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn xem giỏ hàng  2. Nhấn nút "Thanh toán"  3. Nhập sđt không hợp lệ | Nhập số điện thoại: 1234567890 | 3. Hiển thị thông báo "Số điện thoại không hợp lệ, vui lòng kiểm tra lại" | Thất bại |  |
| DH-7 | Kiểm tra thông báo khi bỏ trống các trường thông tin đặt hàng | Đã đăng nhập vào hệ thống và giỏ hàng có sản phẩm | 1. Nhấn xem giỏ hàng  2. Nhấn nút "Thanh toán"  3. Để trống các trường thông tin đặt đơn |  | 3.1 Trường tên hiển thị thông báo "Tên người nhận không được để trống"  3.2 Trường sđt hiển thị thông báo "Số điện thoại không được để trống"  3.3 Trường địa chỉ hiển thị thông báo "Địa chỉ không được để trống" | Đạt |  |

### Test case cho chức năng Hủy đơn hàng

Bảng 3.11: Test case cho chức năng Hủy đơn hàng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Cao | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Hủy đơn hàng | | | | **Ngày thiết kế:** 04/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng hủy đơn hàng cho người dùng khi đơn hàng ở trạng thái chờ xác nhận | | | | **Ngày thực hiện:** 21/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| HD-1 | Hủy đơn hàng thành công | Đã đăng nhập vào hệ thống và có đơn đang ở trạng thái chờ xác nhận | 1. Nhấn vào icon User ở góc phải màn hình  2. Chọn Trang cá nhân  3. Chọn đơn mua  4. Chọn tab Chờ xác nhận  5. Nhấn nút Hủy  6. Nhập lý do hủy  7. Nhấn Xác nhận | Email: trangnhung29072003@gmail.com  Mật khẩu: #Haui123 | 7.1 Hiển thị thông báo "Thành công"  7.2 Nếu đơn hàng bị hủy được thanh toán bằng ví Haui Food thì sẽ được hoàn lại đúng số tiền giá trị đơn hàng | Đạt |  |
| HD-2 | Kiểm tra thông báo khi bỏ trống Lý do hủy đơn hàng | Đã đăng nhập vào hệ thống và có đơn đang ở trạng thái chờ xác nhận | 1. Nhấn vào icon User ở góc phải màn hình  2. Chọn Trang cá nhân  3. Chọn đơn mua  4. Chọn tab Chờ xác nhận  5. Nhấn nút Hủy  6. Tại màn Xác nhận hủy đơn hàng, bỏ trống lý do hủy  7. Nhấn Xác nhận | Email: trangnhung29072003@gmail.com  Mật khẩu: #Nhung123 | 7. Hiển thị thông báo lỗi "Không được để trống lý do hủy đơn hàng, vui lòng nhập lại" | Thất bại |  |

### Test case cho chức năng Xem danh sách đơn hàng

Bảng 3.12: Test case cho chức năng Xem danh sách đơn hàng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Trung bình | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Xem danh sách đơn hàng | | | | **Ngày thiết kế:** 05/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng danh sách đơn hàng cho người dùng bao gồm: khi người dùng chưa mua đơn hàng nào và khi đã mua đơn hàng | | | | **Ngày thực hiện:** 21/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| DS-1 | Xem danh sách đơn hàng khi chưa mua đơn hàng nào | Đã đăng nhập vào hệ thống và chưa mua hàng | 1. Nhấn vào icon user  2. Nhấn chọn Trang cá nhân  3. Chọn "Đơn hàng" | Email: thanhnt@gmail.com  Mật khẩu: #Thanh123 | 3. Hiển thị thông báo "Bạn hiện không có đơn nào" | Đạt |  |
| DS-2 | Xem danh sách đơn mua thành công khi đã mua hàng trước đó | Đã đăng nhập vào hệ thống và đã mua hàng | 1. Nhấn vào icon user  2. Nhấn chọn Trang cá nhân  3. Chọn "Đơn hàng"  4. Chọn các tab cần xem danh sách đơn hàng: Chờ xác nhận, Đã xác nhận, Đang giao, Hoàn thành, Đã hủy, Từ chối | Email: trangnhung29072003@gmail.com  Mật khẩu: #Nhung123 | 3. Hiển thị thông tin chi tiết đơn hàng đã mua  4. Các tab hiển thị đúng các đơn hàng ở trạng thái đó | Đạt |  |

### Test case cho chức năng Xem thông tin tài khoản

Bảng 3.13: Test case cho chức năng Xem thông tin tài khoản

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Thấp | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Xem thông tin tài khoản | | | | **Ngày thiết kế:** 05/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng xem thông tin tài khoản của người dùng | | | | **Ngày thực hiện:** 22/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| XTK-1 | Xem thông tin tài khoản thành công | Đã đăng nhập vào hệ thống | 1. Nhấn vào icon user  2. Nhấn chọn Trang cá nhân  3. Chọn "Thông tin cá nhân" | Email: thanhnt@gmail.com  Mật khẩu: #Thanh123 | 3. Hiển thị màn thông tin cá nhân gồm: họ tên, email, số điện thoại, giới tính | Đạt |  |
| XTK-2 | Xem thông tin tài khoản thất bại khi chưa đăng nhập |  | 1. Truy cập link của trang thông tin cá nhân khi chưa đăng nhập |  | 1. Hiển thị thông báo yêu cầu đăng nhập | Đạt |  |

### Test case cho chức năng Cập nhật thông tin cá nhân

Bảng 3.14: Test case cho chức năng Cập nhật thông tin cá nhân

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Trung bình | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Xem danh sách đơn hàng | | | | **Ngày thiết kế:** 05/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng cập nhật thông tin tài khoản của người dùng | | | | **Ngày thực hiện:** 25/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| CN-1 | Cập nhật thành công thông tin cá nhân | Đã đăng nhập vào hệ thống | 1. Truy cập trang Xem thông tin tài khoản  2. Nhấn nút Chỉnh sửa  3. Nhập thông tin cần chỉnh sửa  4. Nhấn nút Cập nhật | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Trường email ở trạng thái disable không thể chỉnh sửa, các trường khác chuyển sang trạng thái enable để chỉnh sửa  4. Hiển thị thông báo "Cập nhật thành công" | Đạt |  |
| CN-2 | Hủy chỉnh sửa khi thực hiện cập nhật | Đã đang nhập vào hệ thống | 1. Truy cập trang Xem thông tin tài khoản  2. Nhấn nút Chỉnh sửa  3. Nhấn nút Hủy | Email: haui@gmail.com  Mật khẩu: #Haui123 | 3. Các trường dữ liệu trả về thông tin cũ của tài khoản (không thay đổi) | Thất bại |  |
| CN-3 | Kiểm tra thông báo khi không có thông tin nào được thay đổi | Đã đang nhập vào hệ thống | 1. Truy cập trang Xem thông tin tài khoản  2. Nhấn nút Chỉnh sửa  3. Nhấn nút Cập nhật | Email: haui@gmail.com  Mật khẩu: #Haui123 | 3. Hiển thị thông báo "Không có thay đổi nào được thực hiện" | Thất bại |  |
| CN-4 | Kiểm tra thông báo khi nhập vào số điện thoại không hợp lệ | Đã đang nhập vào hệ thống | 1. Truy cập trang Xem thông tin tài khoản  2. Nhấn nút Chỉnh sửa  3. Nhập số điện thoại không hợp lệ  4. Nhấn nút Cập nhật | Email: haui@gmail.com  Mật khẩu: #Haui123 | 4. Hiển thị thông báo lỗi "Vui lòng kiểm tra lại thông tin và nhập đúng định dạng." | Đạt |  |

### Test case cho chức năng Đổi mật khẩu

Bảng 3.15: Test case cho chức năng Đổi mật khẩu

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Trung bình | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Đổi mật khẩu | | | | **Ngày thiết kế:** 05/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng đổi mật khẩu mới cho người dùng khi đã đăng nhập vào hệ thống | | | | **Ngày thực hiện:** 27/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| PW-1 | Đổi mật khẩu thành công | Đã đăng nhập vào hệ thống | 1. Nhấn vào icon user ở góc phải màn hình  2. Chọn Trang cá nhân  3. Nhấn "Đổi mật khẩu"  4. Nhập mật khẩu cũ, mật khẩu mới, xác nhận mật khẩu mới hợp lệ  5. Nhấn Cập nhật | Email: haui@gmail.com  Mật khẩu: #Haui123  Mật khẩu mới: #Haui321 | 5. Hiển thị thông báo "Đổi mật khẩu thành công" | Đạt |  |
| PW-2 | Đổi mật khẩu thất bại khi nhập sai mật khẩu cũ | Đã đang nhập vào hệ thống | 1. Nhấn vào icon user ở góc phải màn hình  2. Chọn Trang cá nhân  3. Nhấn "Đổi mật khẩu"  4. Nhập sai mật khẩu cũ và nhập mật khẩu mới, xác nhận mật khẩu mới hợp lệ  5. Nhấn Cập nhật | email: haui@gmail.com  Mật khẩu sai: #Nhung123 | 5. Hiển thị thông báo "Mật khẩu cũ không chính xác" | Đạt |  |
| PW-3 | Đổi mật khẩu thất bại khi nhập xác nhận mật khẩu mới không khớp | Đã đang nhập vào hệ thống | 1. Nhấn vào icon user ở góc phải màn hình  2. Chọn Trang cá nhân  3. Nhấn "Đổi mật khẩu"  4. Nhập mật khẩu cũ và nhập mật khẩu mới  5. Nhập sai xác nhận mật khẩu mới  6. Nhấn Cập nhật | Email: haui@gmail.com  Mật khẩu: #Haui123 | 5. Hiển thị thông báo "Mật khẩu xác nhận không khớp"  6. Hiển thị thông báo "Vui lòng kiểm tra lại thông tin và nhập đúng định dạng" | Đạt |  |
| PW-4 | Kiểm tra thông báo khi nhập mật khẩu mới không đúng định dạng | Đã đang nhập vào hệ thống | 1. Nhấn vào icon user ở góc phải màn hình  2. Chọn Trang cá nhân  3. Nhấn "Đổi mật khẩu"  4. Nhập mật khẩu cũ  5. Nhập mật khẩu mới và xác nhận không đúng định dạng  6. Nhấn Cập nhật | email: trangnhung29072003@gmail.com  mật khẩu: #Nhung123  mật khẩu mới: 123/abc | 6. Hiển thị thông báo "Vui lòng kiểm tra lại thông tin và nhập đúng định dạng" | Đạt |  |

### Test case cho chức năng Quản lý đơn hàng

Bảng 3.16: Test case cho chức năng Quản lý đơn hàng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Cao | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Quản lý đơn hàng | | | | **Ngày thiết kế:** 29/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng quản lý đơn hàng về phía quản trị viên bao gồm: xem, tìm kiếm, cập nhật các trạng thái đơn hàng | | | | **Ngày thực hiện:** 30/04/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| QLDH-1 | Đăng nhập vào trang quản trị thành công | Tài khoản có quyền truy cập trang quản lý | Truy cập trang quản lý Haui Food:  1. Nhập email  2. Nhập mật khẩu | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2.1 Hiển thị thông báo "Đăng nhập thành công"  2.2 Điều hướng màn hình sang trang thống kê | Đạt |  |
| QLDH-2 | Đăng nhập vào trang quản trị thất bại | Tài khoản không có quyền truy cập quản lý | Truy cập trang quản lý Haui Food:  1. Nhập email  2. Nhập mật khẩu | email: domai@gmail.com  Mật khẩu: #Mai1234 | 2. Hiển thị thông báo "Email or password is invalid" | Đạt |  |
| QLDH-3 | Đăng xuất khỏi phiên đăng nhập thành công | Đã đang nhập vào hệ thống | 1. Nhấn vào icon user ở góc phải màn hình  2. Nhấn Đăng xuất | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị thông báo “Bạn đã đăng xuất thành công” | Đạt |  |
| QLDH-4 | Xem các đơn đặt hàng | Đã đang nhập vào hệ thống | 1. Nhấn chọn “Quản lý đơn hàng” trên thanh menu  2. Kiểm tra bảng quản lý đơn hàng  3. Nhấn nút biểu tượng xem chi tiết trên 1 dòng đơn hàng | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị bảng quản lý đơn hàng có dữ liệu  3. Điều hướng sang trang Chi tiết đơn hàng và hiển thị đầy đủ dữ liệu | Thất bại |  |
| QLDH-5 | Lọc và xem danh sách đơn hàng | Đã đăng nhập vào hệ thống | 1. Nhấn chọn “Quản lý đơn hàng”  2. Tại bộ lọc, nhấn chọn các ô điều kiện muốn xem  3. Nhấn nút “Tìm kiếm” | Email: haui@gmail.com  Mật khẩu: #Haui123 | 3. Hiển thị danh sách các đơn hàng theo đúng điều kiện đã chọn | Đạt |  |
| QLDH-6 | Xác nhận đơn hàng đang ở trạng thái chờ xác nhận | Đã đăng nhập vào hệ thống, có ít nhất 1 đơn hàng ở trạng thái chờ xác nhận | 1.Nhấn chọn “Quản lý đơn hàng”  2. Tại 1 dòng đơn hàng ở trạng thái chờ xác nhận, nhấn vào nút biểu tượng Xác nhận  3. Tại màn Xác nhận, nhấn nút “Xác nhận”  4. Truy cập vào trang Xem đơn hàng về phía người dùng  5. Kiểm tra đơn hàng đã được xác nhận | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị màn Xác nhận  3.1 Hiển thị thông báo thành công  3.2 Đơn hàng chuyển sang trạng thái Đang chuẩn bị  5. Đơn hàng được hiển thị ở tab Đã xác nhận của phía người dùng | Đạt |  |
| QLDH-7 | Cập nhật trạng thái giao đơn hàng cho đơn hàng đang chuẩn bị | Đã đăng nhập vào hệ thống, có ít nhất 1 đơn hàng đang chuẩn bị | 1.Nhấn chọn “Quản lý đơn hàng”  2. Tại 1 dòng đơn hàng ở trạng thái Đang chuẩn bị, nhấn vào nút biểu tượng Giao hàng  3. Tại màn Xác nhận, chọn người giao hàng và nhấn nút “Xác nhận”  4. Truy cập vào trang Xem đơn hàng về phía người dùng  5. Kiểm tra đơn hàng đang giao | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị màn Xác nhận  3.1 Hiển thị thông báo thành công  3.2 Đơn hàng chuyển sang trạng thái Đang giao  5. Đơn hàng được hiển thị ở tab Đang giao của phía người dùng | Đạt |  |
| QLDH-8 | Từ chối đơn hàng đang ở trạng thái Chờ xác nhận | Đã đăng nhập vào hệ thống, có ít nhất 1 đơn hàng ở trạng thái chờ xác nhận | 1.Nhấn chọn “Quản lý đơn hàng”  2. Tại 1 dòng đơn hàng ở trạng thái chờ xác nhận, nhấn vào nút biểu tượng Từ chối  3. Tại màn Xác nhận, nhập lý do và nhấn nút “Xác nhận”  4. Truy cập vào trang Xem đơn hàng về phía người dùng  5. Kiểm tra đơn hàng đã được từ chối | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị màn Xác nhận  3.1 Hiển thị thông báo thành công  3.2 Đơn hàng chuyển sang trạng thái Từ chối  5. Đơn hàng được hiển thị ở tab Từ chối của phía người dùng | Thất bại |  |
| QLDH-9 | Kiểm tra thông báo lỗi khi bỏ trống lý do hủy đơn hàng | Đã đăng nhập vào hệ thống, có ít nhất 1 đơn hàng ở trạng thái chờ xác nhận | 1.Nhấn chọn “Quản lý đơn hàng”  2. Tại 1 dòng đơn hàng ở trạng thái chờ xác nhận, nhấn vào nút biểu tượng Từ chối  3. Tại màn Xác nhận, bỏ trống lý do và nhấn nút “Xác nhận” | Email: haui@gmail.com  Mật khẩu: #Haui123 | 3. Hiển thị thông báo “Lý do không được bỏ trống” | Đạt |  |
| QLDH-10 | Cập nhật trạng thái hoàn thành cho đơn hàng | Đã đăng nhập vào hệ thống | 1.Nhấn chọn “Quản lý đơn hàng”  2. Tại 1 dòng đơn hàng, nhấn vào nút biểu tượng để cập nhật các trạng thái: đang chuẩn bị, đang giao, hoàn thành  3. Kiểm tra lại trạng thái đơn hàng ở phía người dùng | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị thông báo thành công  3. Đơn hàng được cập nhật đúng trạng thái | Đạt |  |

### Test case cho chức năng Quản lý người dùng

Bảng 3.17: Test case cho chức năng Quản lý người dùng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tên dự án: Website đặt đồ ăn trực tuyến Haui Food** | | | | | | | |
| **Test Case** | | | | | | | |
| **Mức độ ưu tiên:** Cao | | | | **Người thiết kế:** Nguyễn Trang Nhung | | | |
| **Tên chức năng:** Quản lý người dùng | | | | **Ngày thiết kế:** 29/04/2025 | | | |
| **Mô tả:** Kiểm thử chức năng quản lý người dùng về phía quản trị viên bao gồm: xem, lọc thêm, sửa, khóa/mở khóa người dùng | | | | **Ngày thực hiện:** 05/05/2025 | | | |
| **ID** | **Mô tả** | **Tiền điều kiện** | **Bước thực hiện** | **Dữ liệu đầu vào** | **Kết quả mong đợi** | **Kết quả** | **Ghi chú** |
| QLND-1 | Xem thông tin chi tiết một người dùng | Đã đăng nhập vào hệ thống | 1.Nhấn nút “Quản lý người dùng”  2. Tại bảng danh sách người dùng, nhấn chọn nút biểu tượng xem chi tiết trên 1 dòng người dùng | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2.1 Điều hướng màn hình sang trang Chi tiết người dùng  2.2 Hiển thị thông tin chi tiết người dùng | Thất bại |  |
| QLND-2 | Lọc và xem danh sách người dùng | Đã đăng nhập vào hệ thống | 1. Nhấn chọn “Quản lý người dùng”  2. Tại bộ lọc, nhấn chọn các ô điều kiện muốn xem  3. Nhấn nút “Tìm kiếm” | Email: haui@gmail.com  Mật khẩu: #Haui123 | 3. Hiển thị danh sách người dùng theo đúng điều kiện lọc | Đạt |  |
| QLND-3 | Thêm người dùng mới thành công | Đã đang nhập vào hệ thống | 1.Nhấn chọn “Quản lý người dùng”  2. Nhấn nút “Thêm người dùng”  3. Nhập thông tin hợp lệ cho người dùng mới  4. Nhấn nút “Thêm người dùng” | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Điều hướng sang trang nhập liệu thêm người dùng  4.1 Hiển thị thông báo thành công  4.2 Hiển thị người dùng trong danh sách | Đạt |  |
| QLND-4 | Chỉnh sửa thông tin người dùng | Đã đang nhập vào hệ thống | 1.Nhấn nút “Quản lý người dùng”  2. Tại bảng danh sách người dùng, nhấn chọn nút biểu tượng Chỉnh sửa trên 1 dòng người dùng  3. Nhập thông tin chỉnh sửa mới  4. Nhấn nút “Chỉnh sửa” | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Điều hướng sang trang chỉnh sửa người dùng và hiển thị thông tin cũ  4.1 Hiển thị thông báo thành công  4.2 Cập nhật lại thông tin người dùng trong danh sách | Đạt |  |
| QLND-5 | Khóa một người dùng | Đã đăng nhập vào hệ thống, có người dùng ở trạng thái mở khóa | 1.Nhấn nút “Quản lý người dùng”  2. Tại bảng danh sách người dùng, nhấn chọn nút biểu tượng Khóa trên 1 dòng người dùng  3. Nhấn nút “Xác nhận”  4. Thử đăng nhập lại ở phía người dùng | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị màn xác nhận Khóa  3. Hiển thị thông báo thành công  4. Hiển thị thông báo “Tài khoản đã bị khóa, vui lòng liên hệ Quản trị viên” | Đạt |  |
| QLND-6 | Mở khóa một người dùng | Đã đang nhập vào hệ thống, có người dùng ở trạng thái khóa | 1.Nhấn nút “Quản lý người dùng”  2. Tại bảng danh sách người dùng, nhấn chọn nút biểu tượng Mở khóa trên 1 dòng người dùng  3. Nhấn nút “Xác nhận”  4. Thử đăng nhập lại ở phía người dùng | Email: haui@gmail.com  Mật khẩu: #Haui123 | 2. Hiển thị màn xác nhận  3. Hiển thị thông báo thành công  4. Người dùng đăng nhập thành công | Đạt |  |

## Thực thi kiểm thử

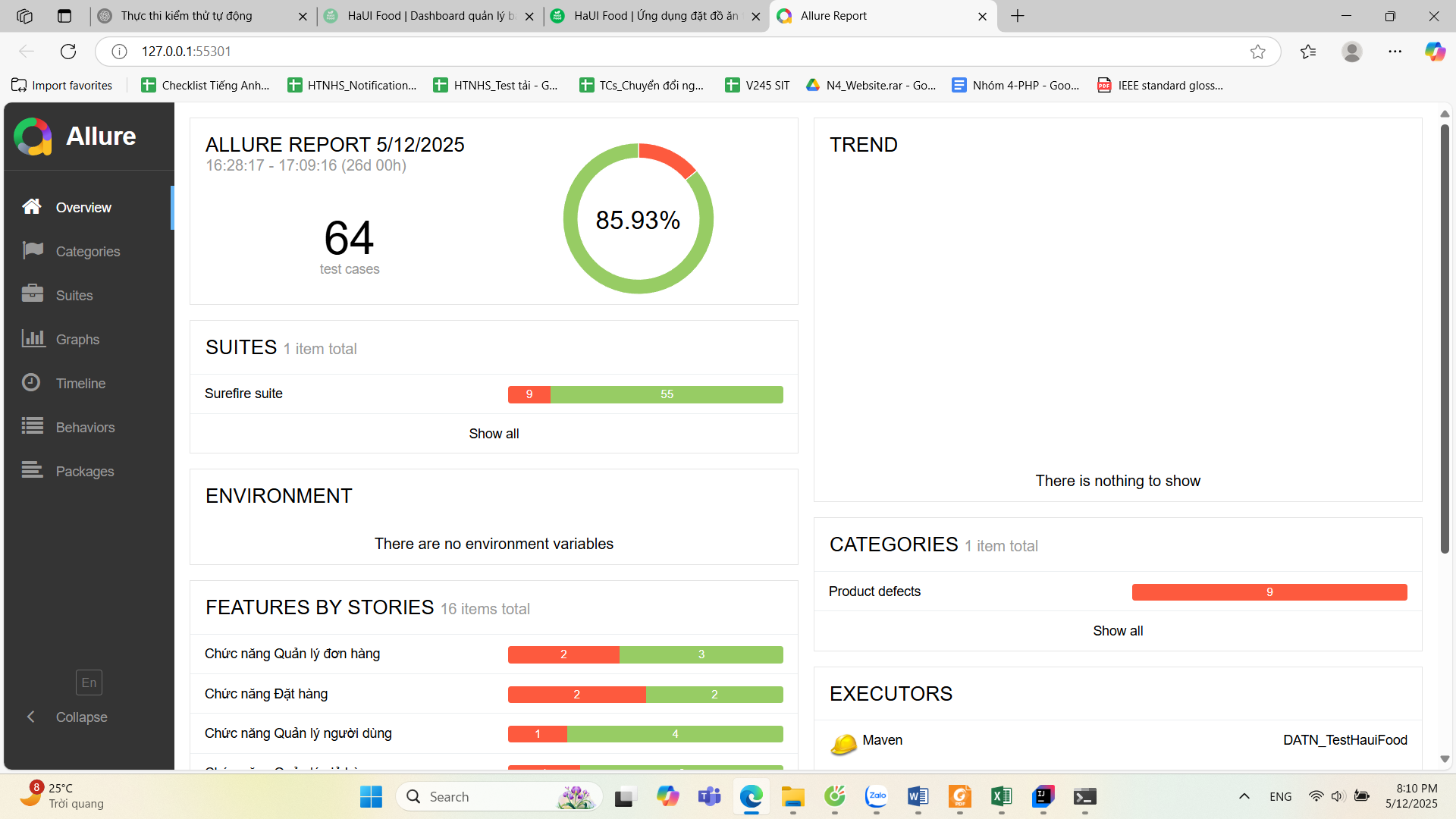
Quá trình thực thi kiểm thử được thực hiện trong môi trường phần cứng và phần mềm như sau:

* Hệ điều hành: Windows 11 (64-bit)
* IDE sử dụng: IntelliJ IDEA
* Ngôn ngữ lập trình: Java
* Framework kiểm thử: Selenium WebDriver + Test NG
* Trình quản lý phụ thuộc: Maven
* Trình duyệt được kiểm thử: Google Chrome (phiên bản 136.0)
* Công cụ báo cáo: Allure Report
* Cách thực thi: Tiến hành qua dòng lệnh bằng Maven và sinh báo cáo tự động bằng Allure

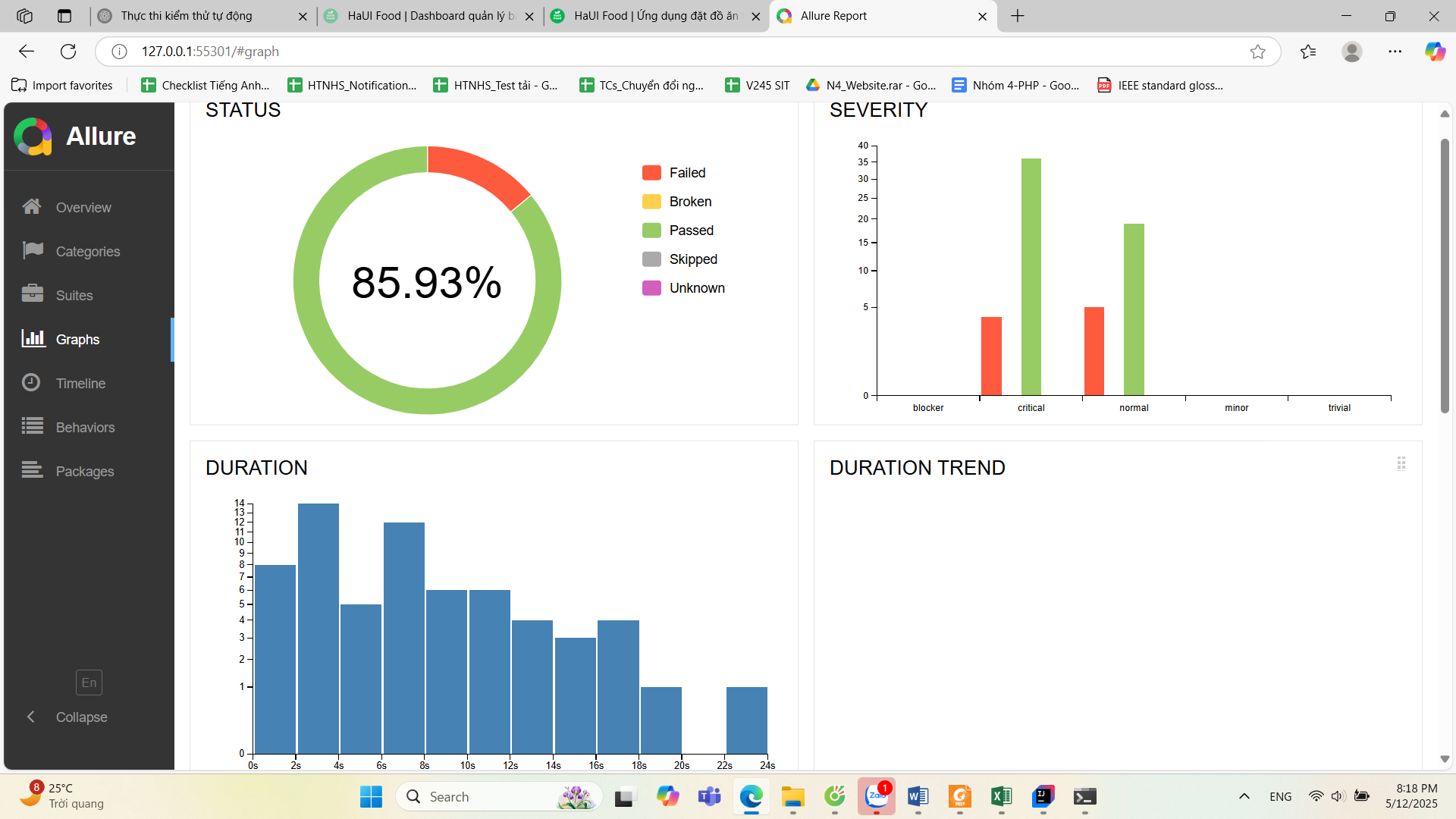
Cách thức thực thi kiểm thử: Các test case được thiết kế và tổ chức theo mô hình Page Object Model (POM). Việc thực thi được tiến hành như sau:

* Chạy test qua dòng lệnh bằng Maven
* Sau mỗi lần chạy test, kết quả sẽ được lưu trong thư mục target/allure-results.
* Sau đó, sử dụng lệnh allure serve để sinh báo cáo dưới dạng giao diện đồ họa HTML
* Với các test case thất bại, hệ thống sẽ tự động chụp ảnh màn hình và lưu trong thư mục target/allure-results.

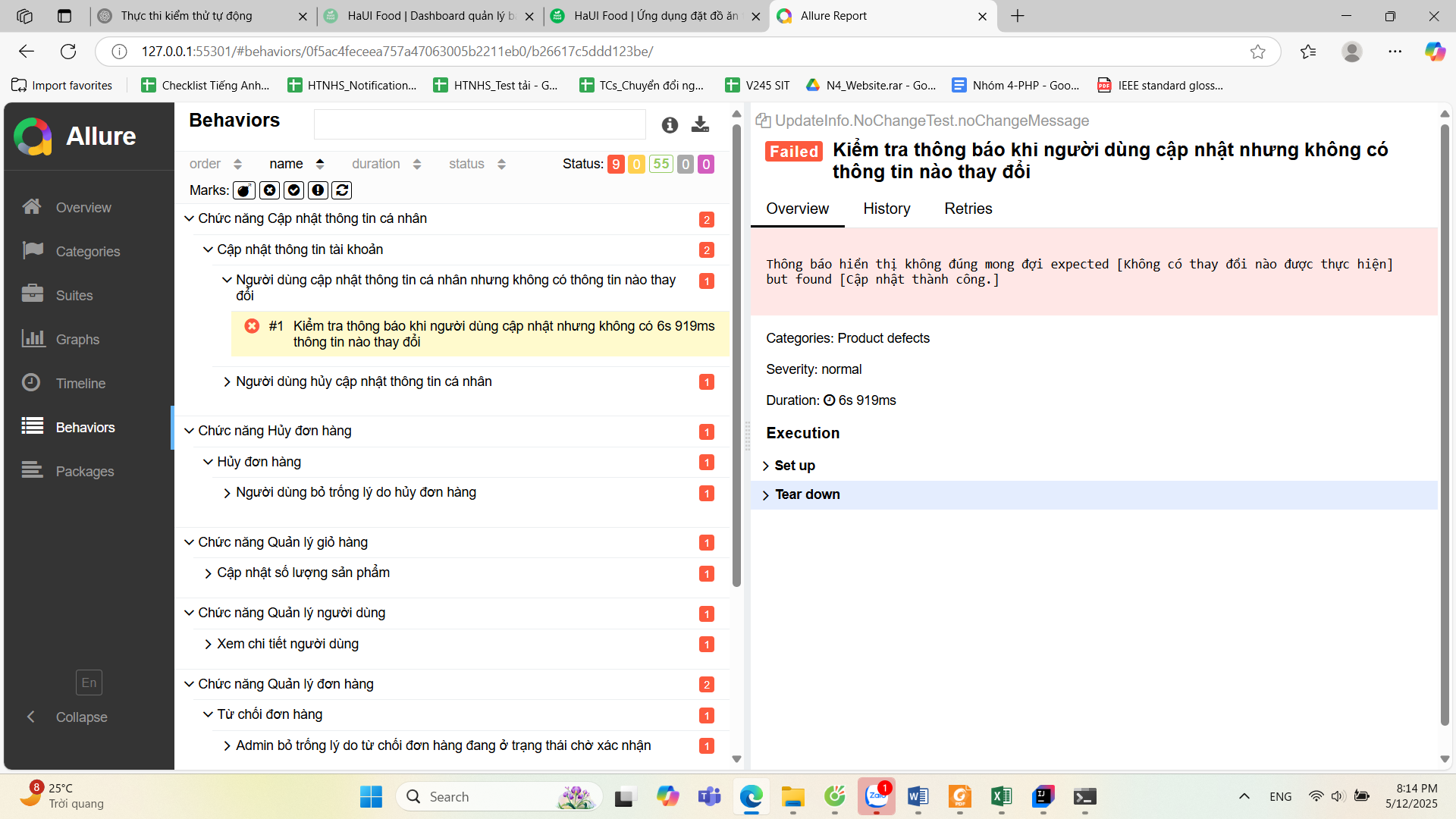
Tổng quan kết quả thực thi kiểm thử: Tổng số test case thực hiện: 64, trong đó: 55 test case ở trạng thái đạt, 9 test case ở trạng thái thất bại



Hình 3.9:Tổng quan kết quả kiểm thử website Haui Food



Hình 3.10: Biểu đồ kết quả kiểm thử theo trạng thái, mức độ và thời gian



Hình 3.11: Chi tiết về những test case thất bại

## Báo cáo và đánh giá kết quả kiểm thử

Sau khi thực thi toàn bộ các test case, hệ thống đã sinh báo cáo kết quả kiểm thử bằng công cụ Allure Report. Kết quả ghi nhận như sau:

* Tổng số test case: 64
* Số lượng passed: 55 (chiếm 85.93%)
* Số lượng failed: 9 (chiếm 14.07%)
* Skipped/Broken/Unknown: 0

Thông qua phân loại lỗi trong báo cáo Allure, các lỗi được phân chia theo mức độ nghiêm trọng như sau:

* Nghiêm trọng (4 lỗi): Các lỗi nghiêm trọng ảnh hưởng đến chức năng cốt lõi: đặt hàng, quản lý đơn hàng, quản lý người dùng. Những lỗi này cần được ưu tiên xử lý sớm nhất.
* Bình thường (3 lỗi): Lỗi ở các tính năng phụ như hiển thị thông báo, refresh trang không đúng lúc.
* Cực nghiêm trọng, Nhẹ, Không đáng kể: Không ghi nhận lỗi ở các mức này.

Về hiệu suất và độ ổn định kiểm thử:

* Tốc độ thực thi: Biểu đồ Duration cho thấy hầu hết các test case được thi thi trong khoảng 2 đến 20 giây, chứng tỏ script kiểm thử hoạt động hiệu quả và hệ thống phản hồi tốt.
* Ổn định kiểm thử: Không có test case nào bị “Skipped” hoặc “Broken”, thể hiện môi trường kiểm thử được thiết lập ổn định, không có xung đột phụ thuộc hay lỗi hạ tầng gây ngắt luồng kiểm thử.

Tiêu chí kiểm thử bao gồm các điều kiện sau:

* 100% test case được thực thi
* Tỷ lệ pass trên 80%
* Không còn lỗi ở mức cực nghiêm trọng
* Tất cả các lỗi phát hiện đã được phân tích và đánh giá
* Tài liệu kiểm thử hoàn tất
* Nhận định tổng quan:
* Tích cực:
* Phần lớn các chức năng chính và kiểm thử dữ liệu biên được kiểm thử thành công
* Tỷ lệ đạt khá cao và không ghi nhận lỗi cực nghiêm trọng
* Các lỗi được ghi nhận đều có thể tái hiện rõ ràng và ghi log đầy đủ thông qua Allure
* Hạn chế:
* Một số test case ở các chức năng quan trọng như đặt hàng vẫn gặp lỗi ở mức nghiêm trọng
* Cần kết hợp với kiểm thử thủ công để đánh giá các chức năng không thể thực hiện kiểm thử tự động như bật/tắt xác thực 2 lớp, quên mật khẩu. Ngoài ra cần kiểm thử hiệu năng và kiểm thử bảo mật để đánh giá toàn diện hơn.
* Đề xuất cải tiến:
* Cần ưu tiên xử lý các lỗi nghiêm nghiêm trọng trước khi triển khai hệ thống thực tế.
* Tích hợp kiểm thử liên tục (CI/CD) bằng GitHub Action để tự động kiểm thử sau mỗi lần cập nhật mã nguồn
* Mở rộng kiểm thử sang các môi trường khác nhau (trình duyệt, độ phân giải màn hình,…) để tăng độ bao phủ kiểm thử.

# KẾT LUẬN

Đề tài **"** Nghiên cứu và sử dụng Selenium Webdriver trong kiểm thử website đặt đồ ăn trực tuyến**"** xuất phát từ nhu cầu nâng cao chất lượng phần mềm và đảm bảo hoạt động chính xác của hệ thống bán hàng trực tuyến. Trong quá trình thực hiện đề tài tác giả đã nghiên cứu, tìm hiểu, áp dụng các kỹ thuật kiểm thử hiện và đạt được những kết quả sau:

* Trình bày được các kiến thức tổng quan về kiểm thử phần mềm, kiểm thử tự động.
* Phân tích yêu cầu phần mềm, xây dựng kế hoạch kiểm thử và thiết kế các kịch bản kiểm thử cho hầu hết các chức năng chính của website.
* Sử dụng công cụ Selenium WebDriver kết hợp với TestNG và ngôn ngữ lập trình Java để viết và tổ chức script kiểm thử tự động.
* Tích hợp Allure Report để tạo báo cáo kiểm thử trực quan, phục vụ cho việc đánh giá chất lượng phần mềm.
* Thực thi kiểm thử thành công với tỷ lệ test case pass đạt 85.93%. Các lỗi được phát hiện được phân loại theo mức độ nghiêm trọng giúp hỗ trợ cải tiến phần mềm.

Do thời gian thực hiện hạn chế và kinh nghiệm thực tế chưa nhiều, kết quả kiểm thử vẫn còn một số điểm có thể chưa hoàn thiện. Tác giải rất mong nhận được ý kiến đóng góp từ thầy cô và các bạn để sản phẩm được hoàn thiện hơn.

Về phương hướng phát triển trong tương lai:

* Tích hợp kiểm thử hiệu năng bằng cách sử dụng công cụ như JMeter hoặc Gatling để đánh giá tốc độ phản hồi, khả năng chịu tải của hệ thống khi có nhiều người dùng truy cập cùng lúc.
* Tự động hóa toàn bộ quy trình CI/CD như GitHub Actions hoặc Jenkins để kiểm thử được kích hoạt tự động mỗi khi cập nhật mã nguồn.
* Chuẩn hóa, tối ưu mã kiểm thử để dễ bảo trì, mở rộng và tái sử dụng khi hệ thống cập nhật tính năng mới.

# TÀI LIỆU THAM KHẢO

1. Thạc Bình Cường (2008), *Kiểm thử và đảm bảo chất lượng phần mềm*, Đại học Bách khoa Hà Nội, Hà Nội.
2. Nguyễn Thị Thanh Huyền (2011), *Giáo trình phân tích thiết kế hệ thống*, Nhà xuất bản Giáo dục Việt Nam, Hà Nội.
3. ISTQB (2018),  *Giáo trình Kiểm thử phần mềm cơ bản,* ISTQB, Bressels.
4. AnhTester (n.d.), *“Cài đặt Selenium WebDriver Java”, AnhTester Blog,* <https://anhtester.com/blog/cai-dat-selenium-webdriver-java-b256.html>
5. AnhTester (n.d.), *“Tạo project Selenium Java và TestNG Framework với Maven trên IntelliJ IDE”, AnhTester Blog,* <https://anhtester.com/blog/selenium-java/tao-project-selenium-java-va-testng-framework-voi-maven-tren-intellij-ide>
6. Nguyễn Thị Hoa (2020), *“Sử dụng TestNG Framework trong tạo kịch bản Selenium”, Viblo,* <https://viblo.asia/p/su-dung-testng-framework-trong-tao-kich-ban-selenium-V3m5W03EKO7>
7. Got It Vietnam (n.d.), *“Selenium WebDriver là gì? Top những ưu nhược điểm mà bạn cần biết”, Got It AI,* <https://vn.got-it.ai/blog/selenium-webdriver-la-gi-top-nhung-uu-nhuoc-diem-ma-ban-can-biet>
8. STC Admin (2013), *“Software Testing Life Cycle STLC”, Software Testing Class*, <https://www.softwaretestingclass.com/software-testing-life-cycle-stlc/>