Bộ Giáo Dục Và Đào Tạo

Trường Đại Học Ngoại Ngữ - Tin Học Thành Phố Hồ Chí Minh

**Khoa Công Nghệ Thông Tin**



**MÔN HỌC : BẢO ĐẢM CHẤT LƯỢNG PHẦN MỀM**

**ĐỀ TÀI : Bảo Đảm Chất Lượng**

**Cho Website Quản Lý Nhà Sách**

**Giáo Viên Hướng Dẫn** :Ths. Tiếu Phùng Mai Sương

**Thành Viên :**

1. Phan Anh Đô – MSSV: 22DH110808

*Tp. Hồ chí minh, Ngày 1 tháng 4 năm 2025*

**Nhận xét của giảng viên**

Giảng viên 01

Giảng viên 02

**Lời cảm ơn**

Kính gửi Thạc Sĩ Tiếu Phùng Mai Sương,

Em xin chân thành cảm ơn thầy vì những kiến thức bổ ích và tâm huyết đã chia sẻ trong môn Kiểm thử chất lượng phần mềm. Sự tận tâm và chuyên nghiệp của thầy không chỉ giúp em nắm vững kiến thức chuyên môn mà còn truyền cảm hứng để em nỗ lực, sáng tạo và phát triển trong lĩnh vực này.

Em rất trân trọng những buổi học và những lời khuyên chân thành của thầy, qua đó em nhận ra tầm quan trọng của việc kiểm thử và đảm bảo chất lượng phần mềm trong mọi dự án. Em hy vọng sẽ tiếp tục học hỏi và áp dụng những bài học quý báu này vào thực tiễn.

Một lần nữa, em xin gửi lời cảm ơn sâu sắc và chúc thầy luôn mạnh khỏe, thành công trong sự nghiệp giảng dạy cũng như trong công việc của mình.

Trân trọng,

**Mục lục**

[I. Giới thiệu chung 10](#_Toc194401643)

[II. Lý thuyết 11](#_Toc194401644)

[1. Kiểm thử phần mềm 11](#_Toc194401645)

[2. Selenium 12](#_Toc194401646)

[III. Kiểm thử hệ thống ABC 14](#_Toc194401647)

[1. Đặc tả các chức năng hệ thống (System Functions) 14](#_Toc194401648)

[2. Xác định mục tiêu, các chuẩn chất lượng cần đạt của hệ thống 16](#_Toc194401649)

[3. Phạm vi kiểm thử của hệ thống (Scope) 18](#_Toc194401650)

[4. Thiết kế testcases/ test checklists 18](#_Toc194401651)

[5. Test Scripts 18](#_Toc194401652)

[6. Test Run/Report 27](#_Toc194401653)

[IV. Kết luận 28](#_Toc194401654)

[1. Đánh giá kết quả sau khi Kiểm Thử 28](#_Toc194401655)

[2. Kết luận 29](#_Toc194401656)

**Bảng phân công công việc**

|  |  |  |
| --- | --- | --- |
| **Họ tên** | **Công việc** | **Hoàn thành** |
| Phan Anh Đô | Viết báo cáo, làm test scenario, testcase, test script | 100% |

1. Giới thiệu chung

Tầm Quan Trọng của Kiểm Thử Phần Mềm

1. Đảm Bảo Chất Lượng Sản Phẩm

Kiểm thử phần mềm giúp đảm bảo rằng sản phẩm phần mềm hoạt động đúng theo yêu cầu và không có lỗi trước khi phát hành. Điều này đặc biệt quan trọng trong ngành thương mại điện tử, nơi mà trải nghiệm người dùng và sự tin cậy của hệ thống là yếu tố then chốt. Một lỗi nhỏ có thể dẫn đến mất mát doanh thu và uy tín của doanh nghiệp1.

2. Giảm Thiểu Rủi Ro

Trong quá trình phát triển phần mềm, việc phát hiện và sửa chữa lỗi sớm sẽ giúp giảm thiểu rủi ro và chi phí phát sinh sau này. Nếu lỗi được phát hiện sau khi sản phẩm đã ra mắt, chi phí sửa chữa sẽ cao hơn nhiều và có thể gây gián đoạn dịch vụ, ảnh hưởng đến trải nghiệm khách hàng2.

3. Bảo Mật Dữ Liệu

Kiểm thử phần mềm giúp phát hiện các lỗ hổng bảo mật, đảm bảo rằng dữ liệu khách hàng được bảo vệ an toàn. Trong ngành thương mại điện tử, bảo mật dữ liệu là yếu tố sống còn, vì các cuộc tấn công mạng có thể gây thiệt hại lớn về tài chính và uy tín3.

4. Tăng Trải Nghiệm Người Dùng

Một sản phẩm phần mềm được kiểm thử kỹ lưỡng sẽ mang lại trải nghiệm người dùng tốt hơn. Giao diện thân thiện, hiệu suất ổn định và tính năng hoạt động mượt mà sẽ giúp giữ chân khách hàng và tăng doanh thu cho doanh nghiệp4.

5. Đáp Ứng Nhu Cầu Thị Trường

Trong bối cảnh cạnh tranh khốc liệt của ngành thương mại điện tử, việc kiểm thử phần mềm giúp doanh nghiệp nhanh chóng đưa ra thị trường những sản phẩm chất lượng cao, đáp ứng nhu cầu ngày càng cao của khách hàng. Điều này giúp doanh nghiệp duy trì và mở rộng thị phần5.

Kết Luận

Kiểm thử phần mềm không chỉ là một bước quan trọng trong quy trình phát triển phần mềm mà còn là yếu tố quyết định đến sự thành công của sản phẩm trên thị trường. Đặc biệt trong ngành thương mại điện tử, nơi mà sự hài lòng của khách hàng và bảo mật dữ liệu là ưu tiên hàng đầu, kiểm thử phần mềm đóng vai trò then chốt trong việc đảm bảo chất lượng và hiệu quả của sản phẩm.

1. Lý thuyết
   1. Kiểm thử phần mềm

Kiểm thử phần mềm là quá trình kiểm tra, phát hiện và loại trừ các lỗi trông phần mềm trước khi ra mắt sản phẩm đến tay người dung. Dưới đây là một tóm tắt về kiểm thử:

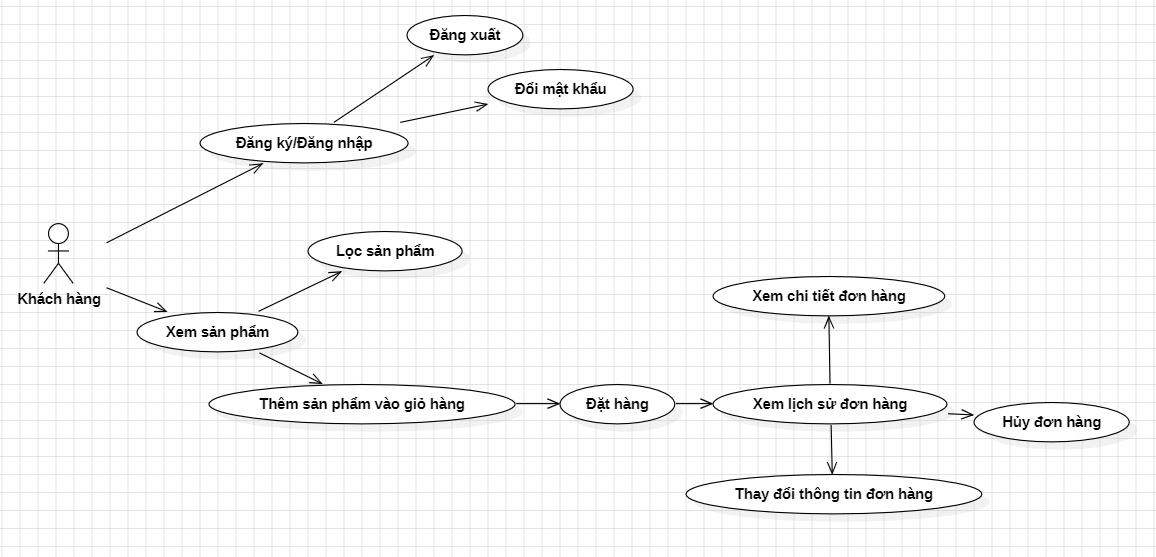
* Mục tiêu chính:
* Mục đích của kiểm thử là đảm bảo phần mềm hoạt động đúng theo yêu cầu đề ra và loại trừ các lỗi phần mềm để nâng cao chất lượng sản phẩm.
* Thõa màn người dung và những bên liên qua đảm bảo phần cuộc sống của họ.
* Loại kiểm thử:
* Kiểm thử chức năng: Tập trung vào kiểm tra tính năng và chức năng của sản phẩm.
* Kiểm thử phi chức năng: Kiểm tra các khía cạnh không liên quan đến tính năng, như hiệu suất, bảo mật và sự ổn định của hệ thống.
* Kiểm thử tự động: Sử dụng các công cụ và kịch bản kiểm thử để tự động hóa quá trình kiểm thử.
* Kiểm thử thủ công: Thực hiện kiểm thử bằng tay bằng cách thử nghiệm từng chức năng một.
* Kiểm thử liên tục:
* Quá trình kiểm thử không nên chỉ diễn ra ở giai đoạn cuối của dự án.
* Kiểm thử nên được tích hợp và thực hiện liên tục trong quá trình phát triển để phát hiện và sửa lỗi sớm.
* Tư duy kiểm thử:
* Kiểm thử đòi hỏi một tư duy chặt chẽ, sẵn sàng tìm kiếm lỗi và kiên nhẫn trong việc xác minh tính đúng đắn của sản phẩm.
* Quản lý lỗi:
* Các lỗi phát hiện được trong quá trình kiểm thử cần được ghi lại, theo dõi, và quản lý để đảm bảo rằng chúng được sửa và kiểm tra lại.
* Các kỹ thuật phổ biến:
* Kiểm thủ hộp tráng dựa trên mã nguồn; kiểm thử hộp đen dựa trên yêu cầu; kiểm thử chức năng, tải, hiệu năng, tích hợp.
* Kết quả kiểm thử sẽ giúp đánh giá chất lượng phần mềm, phát hiện và khắc phục kịp thời các lỗi trước khi bàn giao sản phẩm.
* Kiểm thử là một phần quan trọng của quá trình phát triển và bảo trì phần mềm, đảm bảo rằng sản phẩm đáp ứng yêu cầu và đáng tin cậy trong hoạt động.
  1. Selenium
* **Selenium** là một bộ công cụ kiểm thử tự động (automation testing tool) mã nguồn mở được thiết kế dành riêng cho các ứng dụng web. Nó cho phép người kiểm thử giả lập các thao tác của người dùng trên trình duyệt (như click, nhập liệu, di chuyển trang) và hỗ trợ chạy trên nhiều trình duyệt (Chrome, Firefox, IE, …) cũng như trên nhiều hệ điều hành khác nhau. Ngoài ra, Selenium hỗ trợ nhiều ngôn ngữ lập trình như Java, Python, C#, Ruby, PHP,… giúp tích hợp dễ dàng vào quy trình kiểm thử hiện đại.
* **Selenium IDE**: Selenium IDE là một plugin (trước đây chủ yếu cho Firefox, hiện nay cũng có phiên bản cho Chrome) cho phép ghi lại và phát lại các thao tác kiểm thử trên trình duyệt.
* Ưu điểm:
* **Dễ cài đặt và sử dụng:** Chỉ cần cài đặt plugin, bạn có thể ghi lại các thao tác mà không cần phải viết mã.
* **Phù hợp cho người mới:** Không yêu cầu kiến thức lập trình, rất thích hợp để tạo mẫu (prototype) và kiểm thử các trường hợp đơn giản.
* **Tích hợp báo cáo cơ bản:** Cung cấp một số chức năng báo cáo và cho phép export các test case sang các ngôn ngữ khác để phát triển thêm.
* Nhược điểm:
* **Giới hạn về tính năng:** Chỉ phù hợp với các test case đơn giản; khi cần logic phức tạp, tính toán hay điều kiện rẽ nhánh, IDE gặp hạn chế.
* **Hiệu năng và mở rộng:** Không thể xử lý tốt các kịch bản kiểm thử phức tạp và tốc độ thực thi chậm hơn so với WebDriver.
* **Chỉ chạy trên trình duyệt:** Do là plugin trình duyệt, nên nó chỉ hoạt động trong môi trường trình duyệt, không phù hợp cho tự động hóa các ứng dụng desktop.
* **Selenium WebDriver**: Selenium WebDriver là thư viện giúp tương tác trực tiếp với trình duyệt thông qua các driver riêng (ChromeDriver, FirefoxDriver, …). Nó gửi lệnh trực tiếp từ mã kiểm thử đến trình duyệt, giúp mô phỏng chính xác các hành động của người dùng.

Ưu điểm

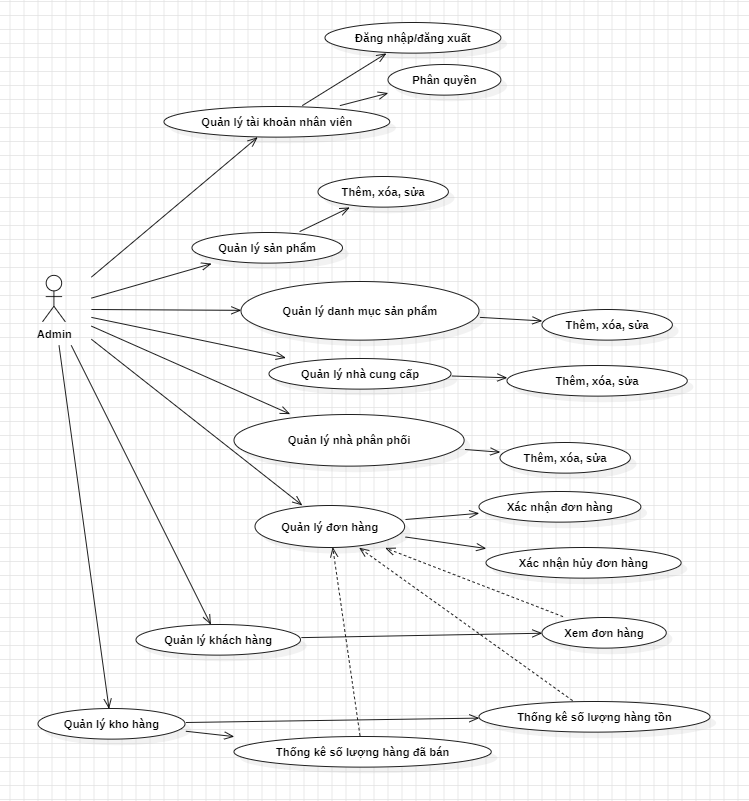
* Đa ngôn ngữ và đa nền tảng: Hỗ trợ nhiều ngôn ngữ lập trình và chạy trên hầu hết các trình duyệt, hệ điều hành (Windows, Mac, Linux, Android, iOS).
* Tốc độ nhanh và ổn định: Giao tiếp trực tiếp với trình duyệt giúp giảm độ trễ và tăng tốc độ thực thi.
* Linh hoạt: Cho phép xây dựng các kịch bản kiểm thử phức tạp, tích hợp với các framework kiểm thử khác (TestNG, JUnit, Allure,…).
* Cộng đồng mạnh: Được sử dụng rộng rãi trên toàn thế giới, có rất nhiều tài liệu và sự hỗ trợ từ cộng đồng.
* Nhược điểm
* Yêu cầu kiến thức lập trình: Người dùng cần nắm vững một ngôn ngữ lập trình để viết và bảo trì các test script.
* Cài đặt phức tạp: Quá trình cài đặt ban đầu và cấu hình môi trường có thể phức tạp hơn so với Selenium IDE.
* Chỉ hỗ trợ kiểm thử web: Không thể tự động hóa các ứng dụng desktop hoặc các trường hợp ngoài môi trường web.

1. Kiểm thử hệ thống ABC
   1. Đặc tả các chức năng hệ thống (System Functions)

* **Khách hàng:**
* Đăng ký tài khoản.
* Đăng nhập.
* Đăng xuất.
* Chọn danh mục sản phẩm và lọc sản phẩm.
* Xem sản phẩm.
* Thêm vào giỏ hàng.
* Mua hàng.
* Xem lịch sử đơn hàng.
* Xem chi tiết đơn hàng.

****

* **Admin:**
* Quản lý tài khoản quản trị.
* Quản lý sản phẩm.
* Quản lý chủng loại sản phẩm.
* Quản lý nhà cung cấp.
* Quản lý nhà phân phối.
* Quản lý đơn đặt hàng, chi tiết đơn đặt hàng.
* Quản lý tài khoản khách hàng.
* Quản lý kho hàng.

****

* 1. Xác định mục tiêu, các chuẩn chất lượng cần đạt của hệ thống

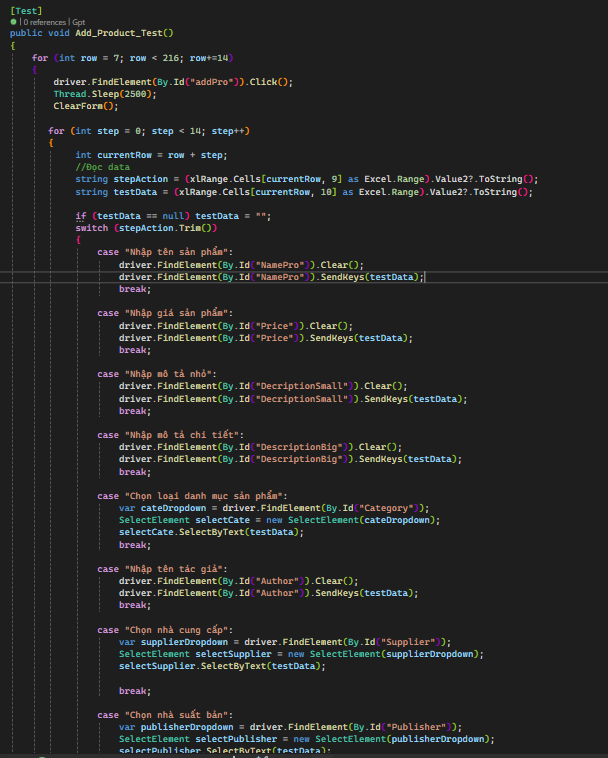
1. **Mục tiêu**

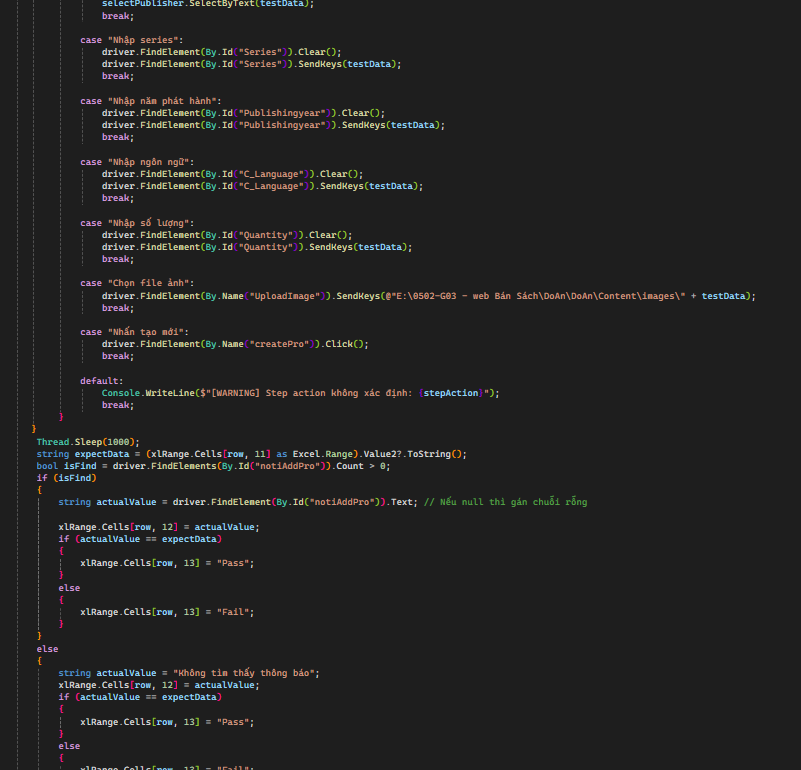
* Dễ theo dõi đơn hàng, dễ tìm kiếm.
* Đối với người QL: quy trình bán hàng đơn giản hiệu quả nhanh chóng, dễ dàng theo dõi được lượng hàng bán ra.
* Xác định thông tin cơ bản về dự án và các thành phần chức năng được kiểm thử và không được kiểm thử.
* Liệt kê những yêu cầu cho việc kiểm thử (Test Requirements).
* Những chiến lược kiểm thử nên được sử dụng.
* Ước lượng những yêu cầu về tài nguyên và chi phí cho việc kiểm thử.
* Những tài liệu được lập sau khi hoàn thành việc kiểm thử.
* Đảm bảo mục tiêu test đúng đắn của chức năng, bao gồm định hướng, dữ liệu đầu vào và dữ liệu nhận được.
* Kiểm tra tính đúng đắn của các dữ liệu, quy trình và báo cáo cũng như việc thực hiện đúng những quy tắc nghiệp vụ.

1. **Các chuẩn cần đạt của hệ thống: iso 9126 có 6 chuẩn**

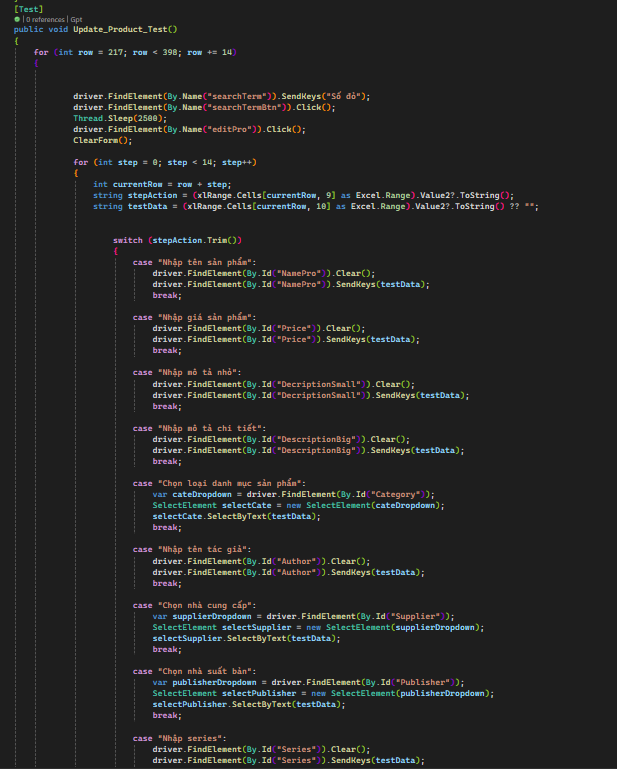
* Tính chức năng: Khả năng của phần mềm cung cấp các chức năng đáp ứng được nhu cầu sử dụng khi phần mềm làm viêc trong điều kiện cụ thể. Bao gồm:
* Tính phù hợp.
* Tính chính xác.
* Khả năng hợp tác làm việc.
* Tính an toàn.
* Tính phù hợp.
* Tính tin cậy: là khả năng của phần mềm có thể hoạt động ổn định trong những điều kiện cụ thể.
* Tính hoàn thiện.
* Khả năng chịu lỗi.
* Khả năng phục hồi.
* Tính tin cậy phù hợp.
* Tính khả dụng: là khả năng của phần mềm có thể hiểu được, học được, sử dụng được và hấp dẫn người sử dụng trong từng trường hợp sử dụng cụ thể.
* Có thể hiểu được.
* Có thể học được.
* Có thể sử dụng được.
* Tính hấp dẫn.
* Tính khả năng phù hợp.
  1. Phạm vi kiểm thử của hệ thống (Scope)
* **Tác vụ khách hàng**
* Đăng ký tài khoản (Kiểm thử tự động).
* Đăng nhập.
* Đăng xuất.
* Chọn danh mục sản phẩm.
* Lọc sản phẩm.
* Xem giỏ hàng.
* Đặt hàng.
* **Tác vụ Admin**
* Quản lý sản phẩm (Kiểm thử tự động).
* **Đa tác vụ**
* Đăng ký tài khoản khách hàng, đặt hàng, kiểm tra đơn hàng
  1. Thiết kế testcases/ test checklists
* [**Link Test Scenario- Test Case**](https://hufliteduvn-my.sharepoint.com/:x:/r/personal/suongtpm_huflit_edu_vn/_layouts/15/Doc.aspx?sourcedoc=%7B961FCE1C-14D7-410C-8169-79533E738788%7D&file=4217_Web%20Qu%E1%BA%A3n%20L%C3%BD%20Nh%C3%A0%20S%C3%A1ch_TestScenarioTestCase.xlsx&action=default&mobileredirect=true)
  1. Test Scripts

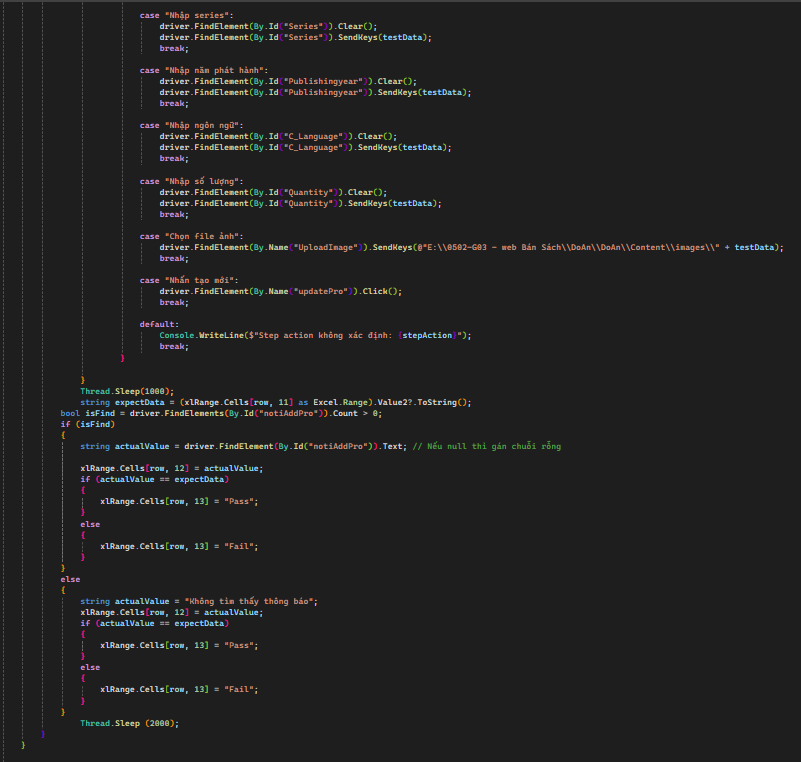
1. **Thêm sản phẩm**

****

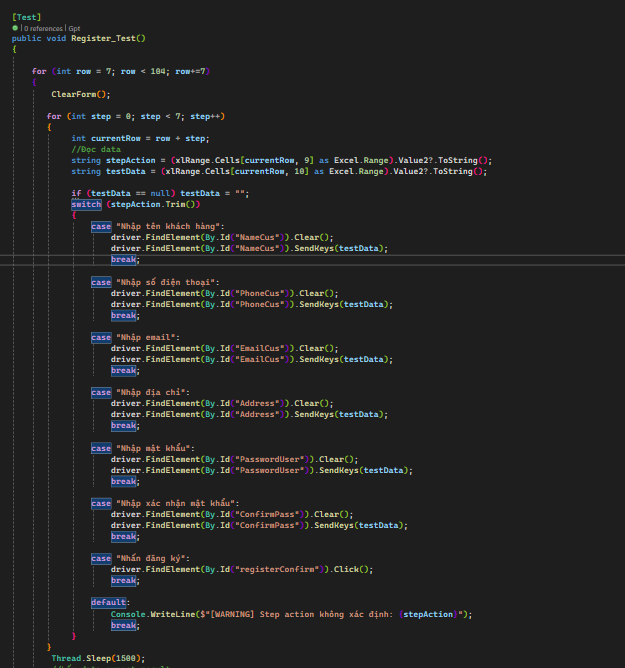
****

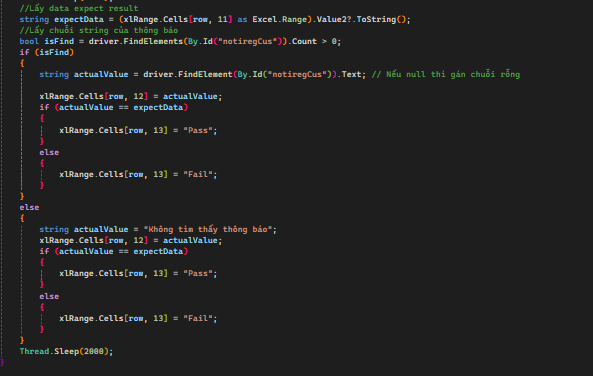
1. **Thay đổi thông tin sản phẩm**

****

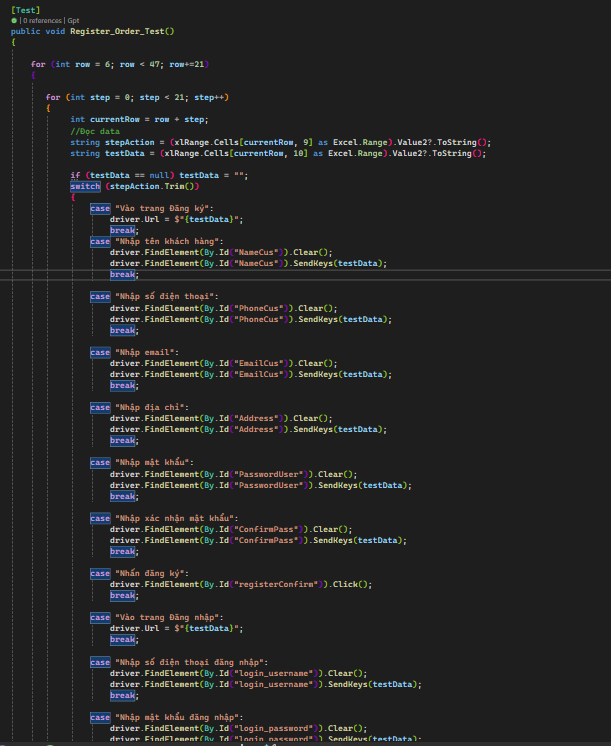
****

1. **Đăng ký tài khoản khách hàng**

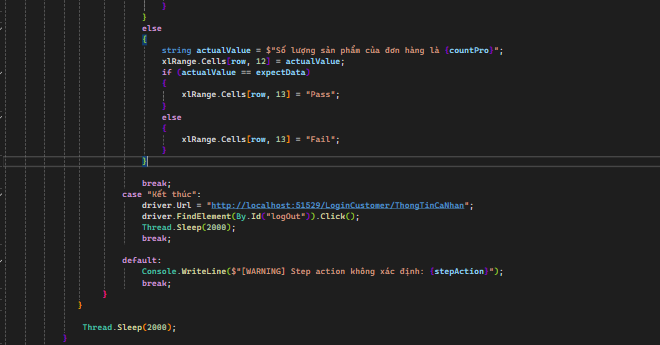
****

****

1. **Đa chức năng**

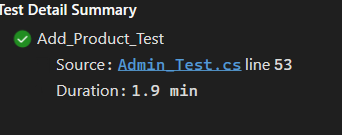
****

****

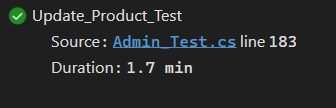
****

* 1. Test Run/Report

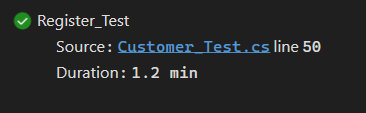
1. **Thêm sản phẩm**

****

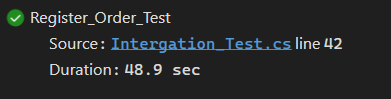
1. **Thay đổi thông tin sản phẩm**

****

1. **Đăng ký tài khoản khách hàng**

****

1. **Đa chức năng**

****

1. Kết luận
   1. Đánh giá kết quả sau khi Kiểm Thử

* Quá trình kiểm thử hệ thống website quản lý nhà sách đã được thực hiện , bao gồm nhiều kịch bản kiểm thử khác nhau để đảm bảo rằng các chức năng được kiểm tra của hệ thống hoạt động đúng đắn và hiệu quả. Các kịch bản kiểm thử bao gồm kiểm thử chức năng đăng nhập, đăng ký, giỏ hàng, thanh toán, quản lý sản phẩm, và nhiều chức năng khác. Mỗi kịch bản kiểm thử đều được mô tả rõ ràng với các bước thực hiện, dữ liệu kiểm thử, kết quả mong đợi và kết quả thực tế.
* Những Khó Khăn Gặp Phải
  + - Lỗi Giao Diện Người Dùng (UI)
* Mô tả: Một số lỗi liên quan đến giao diện người dùng như không hiển thị đúng thông tin, lỗi định dạng, hoặc không phản hồi khi người dùng thực hiện các thao tác.
* Giải quyết: Thực hiện kiểm thử giao diện người dùng (UI testing) kỹ lưỡng hơn, sử dụng các công cụ tự động hóa kiểm thử giao diện để phát hiện và sửa lỗi sớm.
  + - Lỗi Thông Báo Lỗi Khi Cập Nhật Thông Tin Sản Phẩm Và Đăng Ký Tài Khoản
* Mô tả: Thông báo lỗi của chức năng cập nhật thông tin sản phẩm hiển thị lên giao diện không đúng với yêu cầu, thông báo lỗi của phần đăng ký tài khoản khách hàng không hiển thị trên giao diện.
* Giải quyết: Kiểm thử chi tiết đoạn mã phụ trách cho phần hiển thị thông báo của hai chức năng.
  + - Lỗi Tương Thích Trình Duyệt
* Mô tả: Một số chức năng không hoạt động đúng trên các trình duyệt khác nhau hoặc các phiên bản trình duyệt cũ.
* Giải quyết: Thực hiện kiểm thử tương thích trình duyệt (cross-browser testing) để đảm bảo rằng hệ thống hoạt động tốt trên tất cả các trình duyệt phổ biến và các phiên bản khác nhau.
  + - Lỗi Hiệu Suất Hệ Thống
* Mô tả: Hệ thống gặp vấn đề về hiệu suất như thời gian tải trang lâu, phản hồi chậm khi có nhiều người dùng truy cập cùng lúc.
* Giải quyết: Thực hiện kiểm thử hiệu suất (performance testing) để xác định các điểm nghẽn và tối ưu hóa mã nguồn, cơ sở dữ liệu và cấu hình máy chủ.
  + - Lỗi Bảo Mật
* Mô tả: Các lỗ hổng bảo mật như SQL injection, XSS (cross-site scripting), và các vấn đề liên quan đến bảo mật dữ liệu người dùng.
* Giải quyết: Thực hiện kiểm thử bảo mật (security testing) để phát hiện và khắc phục các lỗ hổng bảo mật, áp dụng các biện pháp bảo mật như mã hóa dữ liệu, xác thực hai yếu tố (2FA), và kiểm tra bảo mật định kỳ.
  1. Kết luận

Qua quá trình kiểm thử toàn diện hệ thống website quản lý nhà sách, chúng ta nhận thấy rằng các chức năng cơ bản như đăng nhập, đăng ký, quản lý sản phẩm, giỏ hàng và thanh toán đều hoạt động đúng theo yêu cầu chức năng. Tuy nhiên, trong quá trình kiểm thử cũng đã phát hiện một số lỗi cần được khắc phục như sau:

* **Lỗi giao diện người dùng (UI):** Một số lỗi định dạng, hiển thị thông tin chưa đúng và phản hồi không kịp thời khi người dùng thao tác. Để khắc phục, cần thực hiện kiểm thử UI kỹ lưỡng hơn và sử dụng các công cụ tự động hóa để phát hiện lỗi sớm.
* **Lỗi thông báo lỗi:** Một số chức năng như cập nhật thông tin sản phẩm và đăng ký tài khoản chưa hiển thị thông báo lỗi một cách rõ ràng, dẫn đến khả năng gây nhầm lẫn cho người dùng. Việc kiểm thử chi tiết đoạn mã phụ trách hiển thị thông báo đã giúp xác định nguyên nhân và đưa ra hướng khắc phục.
* **Lỗi tương thích trình duyệt:** Một số chức năng không hoạt động đúng trên các trình duyệt hoặc phiên bản cũ, cho thấy cần thực hiện kiểm thử cross-browser để đảm bảo trải nghiệm người dùng nhất quán trên nhiều nền tảng.
* **Lỗi hiệu suất hệ thống:** Hệ thống gặp vấn đề về thời gian tải trang và phản hồi khi có nhiều người dùng truy cập đồng thời. Điều này chỉ ra sự cần thiết phải tối ưu hóa mã nguồn, cơ sở dữ liệu và cấu hình máy chủ thông qua các bài kiểm thử hiệu suất.
* **Lỗi bảo mật:** Các lỗ hổng bảo mật như SQL injection và XSS được phát hiện, đòi hỏi hệ thống cần áp dụng các biện pháp bảo mật mạnh mẽ như mã hóa dữ liệu, xác thực và kiểm tra định kỳ để bảo vệ thông tin người dùng.

Nhìn chung, quá trình kiểm thử đã giúp phát hiện và khắc phục được nhiều vấn đề quan trọng, góp phần nâng cao chất lượng và độ tin cậy của hệ thống. Qua đó, có thể kết luận rằng hệ thống đang ở mức ổn định và sẵn sàng cho việc triển khai, tuy nhiên vẫn cần tiếp tục giám sát và thực hiện kiểm thử định kỳ để đảm bảo hệ thống luôn đáp ứng tốt các yêu cầu kỹ thuật và bảo mật trong môi trường thực tế.

**Tài liệu tham khảo**