**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

\_\_\_\_\_\_\_\_\_\_\_\_\_🙠⬩🙢\_\_\_\_\_\_\_\_\_\_\_\_\_



**BÁO CÁO BÀI TẬP LỚN**

MÔN HỌC: KIỂM THỬ & ĐẢM BẢO CHẤT LƯỢNG PHẦN MỀM

ĐỀ TÀI:KIỂM THỬ WEBSITE BÁN HÀNG ĐIỆN TỬ

Sinh viên thực hiện : Hoàng Ngọc Luân

Dương Công Long

Nguyễn Thị Diễm Quỳnh

Lưu Thị Quyến

Phùng Ngọc Lâm

Lớp : KTPM – K20E

Giảng viên hướng dẫn : Nguyễn Lan Oanh

Thái Nguyên, ngày 1 tháng 8 năm 2024

**MỤC LỤC**

**[CHƯƠNG 1 TỔNG QUAN VỀ KIỂM THỬ PHẦN MỀM 3](#_Toc3460)**

**[1.1 Kiểm thử phần mềm là gì 3](#_Toc4882)**

**[1.2 Lý do phải kiểm thử phần mềm 3](#_Toc20250)**

**[1.3 Vai trò của kiểm thử phần mềm: 3](#_Toc11931)**

**[1.4 Mục tiêu của kiểm thử phần mềm 4](#_Toc10304)**

**[1.5 Các phương pháp kiểm thử phần mềm 4](#_Toc14856)**

**[1.6 Nguyên tắc kiểm thử phần mềm 6](#_Toc29432)**

**[1.7 Các loại hình kiểm thử phần mềm 8](#_Toc32255)**

**[1.8 Các chiến lược kiểm thử phần mềm 8](#_Toc31060)**

**[1.9 Quy trình kiểm thử phần mềm 10](#_Toc14403)**

**[CHƯƠNG 2 LẬP KẾ HOẠCH TEST 13](#_Toc5666)**

**[2.1 Giới thiệu phần mềm 13](#_Toc25654)**

**[2.2 Nhân sự 13](#_Toc23508)**

**[2.3 Công cụ 15](#_Toc17561)**

**[2.4 Phân tích đánh giá rủi do 15](#_Toc31712)**

**[2.5 kinh phí 15](#_Toc8181)**

**CHƯƠNG 3 GIỚI THIỆU CÔNG CỤ KATALON STUDIO VÀ JMERTER**

**[3.1 Công cụ Katalon Studio 16](#_Toc14830)**

[3.1.1 Giới thiệu về kiểm thử tự động 16](#_Toc28044)

[3.1.2 Giới thiệu về công cụ kiểm thử Katalon 16](#_Toc20867)

[3.1.3 Các tính năng chính của Katalon 16](#_Toc24282)

[3.1.4 Ưu điểm 17](#_Toc10772)

[3.1.5 Nhược điểm 17](#_Toc20951)

[3.1.6 Hướng dẫn cài đặt 18](#_Toc21670)

[3.1.7 Hướng dẫn sử dụng 19](#_Toc23203)

[Cách tạo một Test case với Katalon Studio 22](#_Toc26462)

**[3.2 Công cụ Jmeter 23](#_Toc18247)**

[3.2.1 Giới thiệu về kiểm thử trên Jmeter 23](#_Toc11956)

[3.2.2 Các tính năng chính của Jmeter 23](#_Toc22178)

[3.2.3 Ưu điểm 24](#_Toc10537)

[3.2.4 Nhược điểm 25](#_Toc25548)

[3.2.5. Hướng dẫn cài đặt 26](#_Toc10666)

**[CHƯƠNG 4 GIỚI THIỆU WEBSITE BÁN HÀNG ĐIỆN TỬ T&T 27](#_Toc4460)**

**[4.1 Mô tả chung về sản phẩm phần mềm 27](#_Toc7685)**

**[4.2 Đặc tả chức năng và demo 27](#_Toc7376)**

**CHƯƠNG 1 TỔNG QUAN VỀ KIỂM THỬ PHẦN MỀM**

**1.1 Kiểm thử phần mềm là gì**

Theo IEEE, phần mềm bao gồm các chương trình máy tính, các thủ tục, các  
tài liệu và dữ liệu có thể liên quan đến hoạt động của hệ thống máy tính. Kiểm thử phần mềm là quá trình vận hành hệ thống hoặc thành phần dưới những điều kiện xác định, quan sát hoặc ghi nhận kết quả và đưa ra đánh giá về hệ thống hoặc thành phần đó.

**1.2 Lý do phải kiểm thử phần mềm**

Kiểm thử phần mềm là một phần quan trọng trong quá trình phát triển phần mềm vì có các lý do sau đây:

Đảm bảo chất lượng: Kiểm thử giúp đảm bảo rằng phần mềm đáp ứng các yêu cầu và chức năng của nó và hoạt động đúng cách.

Phát hiện lỗi và sửa chữa: Kiểm thử giúp phát hiện và loại bỏ lỗi và khuyết điểm trong phần mềm trước khi nó được triển khai, giúp giảm thiểu các sự cố sau khi ứng dụng đã đi vào hoạt động.

Tăng độ tin cậy: Kiểm thử giúp cải thiện độ tin cậy của ứng dụng, đảm bảo rằng người dùng có thể tin tưởng và sử dụng nó một cách an toàn.

Giảm rủi ro: Kiểm thử giúp giảm rủi ro liên quan đến việc sử dụng phần mềm, bảo vệ dữ liệu và thông tin của người dùng.

Tiết kiệm thời gian và nguồn lực: Mặc dù kiểm thử đòi hỏi sử dụng thời gian và nguồn lực, nhưng nó giúp tiết kiệm thời gian và nguồn lực dài hạn bằng cách ngăn chặn các sự cố và lỗi ở giai đoạn sau của quá trình phát triển.

**1.3 Vai trò của kiểm thử phần mềm:**

Xác định lỗi: Vai trò chính của kiểm thử là xác định lỗi và khuyết điểm trong phần mềm. Điều này giúp sửa chữa chúng trước khi phần mềm được triển khai.

Đảm bảo chất lượng: Kiểm thử đảm bảo rằng phần mềm hoạt động đúng cách và đáp ứng các yêu cầu chất lượng.

Đánh giá hiệu suất: Kiểm thử hiệu năng đánh giá khả năng của phần mềm xử lý tải công việc và thời gian phản hồi.

Bảo vệ thông tin và dữ liệu: Kiểm thử bảo vệ thông tin và dữ liệu của người dùng bằng cách đảm bảo tính bảo mật của ứng dụng.

Giúp ra quyết định: Kết quả kiểm thử cung cấp thông tin quan trọng cho các bên liên quan, giúp họ ra quyết định về việc triển khai phần mềm.

**1.4 Mục tiêu của kiểm thử phần mềm**

Mục tiêu của kiểm thử phần mềm:

Tìm ra được càng nhiều lỗi càng tốt trong điều kiện về thời gian đã định cũng như là 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 ít nhất.
  + Thiết kế tài liệu kiểm thử một cách có hệ thống, thực hiện nó sao cho có hiệu quả

**1.5 Các phương pháp kiểm thử phần mềm**

**Kiểm thử hộp đen:**

Kỹ thuật kiểm thử hộp đen coi hệ thống giống một “hộp đen” mà cấu trúc bên trong của chương trình khơng thể nhìn thấy. Người kiểm thử chỉ kiểm tra các chức năng của phần mềm không cần quan tâm vào cấu trúc bên trong hay hoạt động của nó.  
Kiểm thử hộp đen có các đặc trưng như:  
˗ Nhằm đảm bảo các chức năng đủ và vận hành đúng.  
˗ Thực hiện các phép thử qua giao diện.  
Mục tiêu cảu kiểm thử phần mềm là tìm ra các loại sai:  
 Thiếu chức năng hoạc chức năng không đúng đắn.  
 Giao diện sai.  
 Sai trong cấu trúc hoặc truy cập dữ liệu ngoài.  
 Thực thi chức năng sai.  
 Sai khởi đầu hoặc kết thúc module.  
Các phương pháp kiểm thử hộp đen  
 Phương pháp phân hoạch tương đương  
 Phương pháp phân tích giá trị biên  
 Bảng hỗ trợ quyết định  
Ưu và nhược điểm của kiểm thử hộp đen  
Về ưu điểm:

Người kiểm thử có thể đánh giá phần mềm mềm một cách khách quan mà không cần biết lập trình, tách biệt với quan điểm của lập trình viên.  
Nhược điểm:

Độ bao phủ hạn chế do chỉ có phần nhỏ các test case được thực hiện.

Do người thực hiện kiểm thử khơng biết cấu trúc bên trong chương trình nên việc kiểm thử khơng hiệu quả.

Người kiểm thử có hiểu biết hạn chế về chương trình.

**Kiểm thử hộp trắng:**

Kiểm thử hộp trắng sử dụng để kiểm tra các đoạn mã chương trình phần mềm xem nó có vận hành đúng theo thiết kế hay không.  
Đặc điểm của kiểm thử hộp trắng:

Là chiến lược giải thuật phụ thuộc vào giải thuật, cấu trúc bên trong chương trình phần mềm.

Người kiểm thử phải có kiến thức nhất định về ngơn ngữ lập trình được dùng, hiểu giải thuật được sử dụng bên trong chương trình.

Việc kiểm thử được tiến hành kiểm tra xem giải thuật, câu lệnh có vận hành như thiết kế hay không.

Phương pháp này bắt buộc phải viết test case bao phủ các nhánh trong giải thuật, đảm bảo thực hiện đầy đủ. Mục đích của kiểm thử hộp trắng là bao phủ hết các câu lệnh, điều kiện, các rẽ nhánh trong mã nguồn chương trình.

Ưu và nhược điểm của while-box testing  
Ưu điểm:  
 Phù hợp để tìm kiếm lỗi và các vấn đề giải thuật trong mã lệnh.  
 Có thể tìm ra được các lỗi tiềm ẩn bên trong chương trình.  
 Lập trình viên có thể tự kiểm tra đoạn mã của mình.  
 Việc kiểm thử rà sốt lỗi hiệu quả nhất vì u cầu kiến thức cấu trúc bên trong của chương trình.  
Nhược điểm:  
 Kiểm thử hộp trắng khơng thể tìm thấy tính năng chưa thực hiện hoặc bỏ sót.  
 Người thiết kế test case phải hiểu sâu về cấu trúc bên trong và đánh giá,được chương trình.  
 Yêu cầu truy suất mã lệnh bên trong chương trình.

**1.6 Nguyên tắc kiểm thử phần mềm**

**Kiểm thử chứng mình sự hiện diện của lỗi**

Kiểm thử chỉ có thể chứng minh được rằng sản phẩm có lỗi. Kiểm thử phần mềm không thể chứng mình rằng sản phẩm không còn lỗi. Nghĩa là sản phẩm luôn có lỗi cho dù có kiểm thử nhiều bao nhiêu. Do đó, điều quan trọng là chúng ta phải thiết kế các trường hợp kiểm thử (test case) sao cho có thể tìm được càng nhiều lỗi càng tốt.

**Kiểm thử toàn bộ là không thể**

Trừ khi sản phẩm được kiểm thử quá đơn giản cũng như không có nhiều giá trị đầu vào (chẳng hạn như “Hello World”) thì việc chứng minh sản phẩm không còn bug cho dù có kiểm thử nhiều đến đâu là không khả thi. Hầu hết các sản phẩm ngày nay rất đa dạng và phức tạp do được phát triển trên nhiều nền tảng, công nghệ phong phú cũng như khả năng lưu trữ kết nối dữ liệu lớn, khiến việc kiểm thử trở nên khó khăn và việc kiểm thử toàn bộ là gần như không thể. Kiểm thử với tất cả các kết hợp đầu vào và đầu ra, với tất cả các kịch bản là không thể trừ phi nó chỉ bao gồm ít trường hợp thì có thể kiểm thử toàn bộ. Thay vì kiểm thử toàn bộ, việc phân tích rủi ro và dựa trên sự mức độ ưu tiên chúng ta có thể tập trung việc kiểm thử vào một số điểm cần thiết, có nguy cơ lỗi cao hơn.

**Kiểm thử càng sớm càng tốt**

Nguyên tắc này yêu cầu bắt đầu thử nghiệm phần mềm trong giai đoạn đầu của vòng đời phát triển phần mềm. Các hoạt động kiểm thử phần mềm từ giai đoạn đầu sẽ giúp phát hiện bug sớm hơn. Nó cho phép chuyển giao phần mềm theo yêu cầu đúng thời gian với chất lượng dự kiến. Ngoài ra ai làm phần mềm cũng biết được rằng việc phát hiện lỗi càng trể bao nhiêu thì chi phí để sửa lỗi càng cao bấy nhiêu. Tương tự, việc thay đổi yêu cầu không đúng ngay từ đầu thường tốn ít chi phí thay đổi tính năng trong hệ thống.

**Lỗi thường được phân bố tập trung**

Thông thường, phần lớn lỗi tập trung vào những module, thành phần chức năng chính của hệ thống. Điều này cũng thuận theo nguyên lý Pareto: 80% số lượng lỗi được tìm thấy trong 20% tính năng của hệ thống. Nếu bạn thành công xác định được điều này, bạn sẽ tập trung vào tìm kiếm lỗi quanh khu vực được xác định. Nó được coi là một trong những cách hiệu quả nhất để thực hiện kiểm tra hiệu quả

**Nghịch lý thuốc trừ sâu**

Trong kiểm thử phần mềm, nếu bạn cứ thực thi lặp đi lặp lại một bộ test case thì có khả năng rất thấp bạn sẽ tìm được lỗi từ những trường hợp kiểm thử này. Nguyên nhân là do khi hệ thống ngày càng hoàn thiện, những lỗi được tìm thấy lúc trước đã được sửa trong khi những trường hợp kiểm thử đã cũ. Do đó, khi một lỗi được sửa hay một tính năng mới được thêm vào, chúng ta nên tiến hành làm regression (kiểm thử hồi qui) nhằm mục đích đảm bảo những thay đổi này không ảnh hưởng đến những vùng khác của sản phẩm.

**Kiểm thử phụ thuộc vào ngữ cảnh**

Theo nguyên tắc này thì nếu bạn đang kiểm thử ứng dụng web và ứng dụng di động bằng cách sử dụng chiến lược kiểm thử giống nhau, thì điều đó là sai lầm. Chiến lược để kiểm thử nên khác nhau và phụ thuộc vào chính ứng dụng đó. Chiến lược cho test web application phải khác với ứng dụng android mobile.

**Quan niệm sai lầm về việc “hết lỗi”**

Việc không tìm thấy lỗi trên sản phẩm không đồng nghĩa với việc sản phẩm đã sẵn sàng để tung ra thị trường. Việc không tìm thấy lỗi cũng có thể là do bộ trường hợp kiểm thử được tạo ra chỉ nhằm kiểm tra những tính năng được làm đúng theo yêu cầu thay vì nhằm tìm kiếm lỗi mới.

**1.7 Các loại hình kiểm thử phần mềm**

Các loại hình kiểm thử gồm có: kiểm thử chức năng, kiểm thử phi chức năng, kiểm thử liên quan đến sự thay đổi.  
 Mục tiêu: đảm bảo đúng mục tiêu của kiểm thử chức năng nhập dữ liệu – xử lý – lấy và kiểm tra kết quả trả về.  
 Các loại kiểm thử chức năng:  
 Kiểm thử chức năng của hệ thống.  
 Kiểm thử tích hợp dữ liệu và cơ sở dữ liệu.  
 Kiểm thử vịng lặp cơng việc.  
 Kiểm thử truy cập.  
 Kiểm thử giao diện.  
Kiểm thử phi chức năng.  
 Tập trung vào kiểm thử sản phẩm, hệ thống phần mềm cần kiểm thử có những đặc tính tốt như thế nào.  
 Sử dụng hiệu quả nhất trong kiểm thử hệ thống và kiểm thử chấp nhận sản phẩm.  
 4 loại kiểm thử phi chức năng thường dùng:  
 Kiểm thử hiệu năng  
 Kiểm thử tải trọng  
 Kiểm thử tập trung.  
 Kiểm thử với dữ liệu lớn.  
Kiểm thử liên quan đến sự thay đổi:  
 Thực hiện hoạt động kiểm thử khi có sự thay đổi trên hoặc trong sản phẩm phần mềm.  
 Sự thay đổi của sản phẩm phần mềm có thể là:  
 Sửa chữa các lỗi tìm được.  
 Sản phẩm được nâng cấp được thay đổi về chức năng.  
 Gồm: kiểm thử lại và kiểm thử hồi quy.

**1.8 Các chiến lược kiểm thử phần mềm**

Kiểm thử chức năng (Functional Testing): Chiến lược này kiểm tra tính đúng đắn của các chức năng cụ thể trong phần mềm. Kiểm thử viên sử dụng tài liệu yêu cầu chức năng để tạo các kịch bản kiểm thử và kiểm tra xem phần mềm có hoạt động đúng đắn theo các chức năng được định nghĩa hay không.

Kiểm thử không chức năng (Non-Functional Testing): Trong chiến lược này, kiểm thử viên kiểm tra các khía cạnh không chức năng của phần mềm như hiệu suất, bảo mật, sức chịu đựng, tương tác người dùng, và tính sẵn sàng.

Kiểm thử tự động (Automated Testing): Kiểm thử tự động sử dụng các công cụ và kịch bản kiểm thử tự động để thực hiện các bài kiểm tra. Điều này giúp tăng tính lặp lại và hiệu suất của kiểm thử.

Kiểm thử đa nền tảng (Cross-Platform Testing): Trong chiến lược này, phần mềm được kiểm tra để đảm bảo tính tương thích và hoạt động đúng trên nhiều nền tảng như Windows, macOS, iOS, Android, và các trình duyệt web khác nhau.

Kiểm thử tích hợp (Integration Testing): Kiểm thử tích hợp kiểm tra tích hợp của các thành phần phần mềm. Kiểm thử viên kiểm tra sự tương tác giữa các thành phần để đảm bảo tích hợp đúng đắn.

Kiểm thử hợp đồng (Contract Testing): Kiểm thử hợp đồng kiểm tra xem các thành phần phần mềm tuân thủ các giao diện và hợp đồng đã được xác định trước đối với các dịch vụ mà chúng liên kết.

Kiểm thử liên tục (Continuous Testing): Chiến lược này tích hợp kiểm thử vào quy trình phát triển và triển khai liên tục để phát hiện lỗi ngay lập tức sau khi chúng xuất hiện.

Kiểm thử bảo mật (Security Testing): Kiểm thử bảo mật kiểm tra các lỗ hổng bảo mật trong phần mềm để đảm bảo tính an toàn và ngăn chặn các cuộc tấn công.

Kiểm thử tương tác người dùng (Usability Testing): Chiến lược này kiểm tra trải nghiệm người dùng để đảm bảo tính dễ sử dụng và thân thiện của phần mềm.

Kiểm thử hiệu suất (Performance Testing): Kiểm thử hiệu suất kiểm tra khả năng của phần mềm hoạt động dưới tải công việc khác nhau để đảm bảo tính nhanh chóng và ổn định.

Kiểm thử tự động giao diện người dùng (GUI Testing): Kiểm thử giao diện người dùng đồ họa để đảm bảo tính tương thích và hiệu suất.

Kiểm thử đa ngôn ngữ (Internationalization and Localization Testing): Kiểm thử đa ngôn ngữ kiểm tra tính đa ngôn ngữ và tính cơ địa hóa của phần mềm, đảm bảo rằng nó hoạt động đúng đắn và hiển thị đúng đắn trong các ngôn ngữ và vùng miền khác nhau.

**1.9 Quy trình kiểm thử phần mềm**

Xác định Mục tiêu kiểm thử: Bước đầu tiên trong quy trình là xác định mục tiêu kiểm thử. Điều này bao gồm việc hiểu rõ yêu cầu của khách hàng, mục tiêu chất lượng, và các tính năng cần kiểm tra.

**Lập Kế Hoạch Kiểm Thử (Test Planning):**

Xác định mục tiêu kiểm thử và phạm vi kiểm thử.

Xác định nguồn lực cần thiết (nhân sự, máy tính, phần mềm kiểm thử).

Xây dựng lịch trình kiểm thử.

Xác định môi trường kiểm thử (máy chủ, cơ sở dữ liệu, phần cứng).

Lập kế hoạch kiểm thử tự động (nếu áp dụng).

**2. Lập Kế Hoạch Kiểm Thử Chi Tiết (Detailed Test Planning):**

Xác định các tài liệu kiểm thử cần được xây dựng, chẳng hạn như kịch bản kiểm thử, dữ liệu kiểm thử, và báo cáo kiểm thử.

Chuẩn bị dữ liệu kiểm thử và môi trường kiểm thử.

Xác định các trường hợp kiểm thử cụ thể dựa trên yêu cầu chức năng và phi chức năng của phần mềm.

**3. Thiết Kế Kịch Bản Kiểm Thử (Test Script Design):**

Tạo các kịch bản kiểm thử dựa trên trường hợp kiểm thử đã xác định.

Xác định các bước kiểm thử cụ thể, bao gồm dữ liệu kiểm thử đầu vào và các dự đoán kết quả mong đợi.

**4. Chuẩn Bị Môi Trường Kiểm Thử (Test Environment Setup):**

Cài đặt và cấu hình môi trường kiểm thử, bao gồm máy chủ, cơ sở dữ liệu, và các phần mềm liên quan khác.

Chuẩn bị dữ liệu kiểm thử và thiết lập cơ sở dữ liệu kiểm thử.

**5. Thực Hiện Kiểm Thử (Test Execution):**

Chạy các kịch bản kiểm thử đã thiết kế trước đó.

Ghi lại kết quả kiểm thử và các lỗi được tìm thấy.

Theo dõi tiến trình kiểm thử và báo cáo về tiến độ cho nhóm quản lý và phát triển.

**6. Ghi Lỗi (Bug Reporting):**

Nếu phát hiện lỗi trong quá trình kiểm thử, ghi lại thông tin chi tiết về lỗi, bao gồm các bước để tái tái hiện lỗi và thông tin về môi trường kiểm thử.

Gửi báo cáo lỗi cho nhóm phát triển để khắc phục.

**7. Kiểm Thử Được Phê Duyệt (Test Review):**

Duyệt xét và xác nhận kết quả kiểm thử.

Đảm bảo rằng các lỗi đã được khắc phục một cách đầy đủ.

**8. Kiểm Thử Hội Đàm (User Acceptance Testing - UAT):**

Kiểm tra với người dùng cuối hoặc khách hàng để đảm bảo tính đáp ứng yêu cầu của họ.

**9. Kiểm Thử Tự Động (Test Automation):**

Tự động hóa các bước kiểm thử sử dụng các công cụ kiểm thử tự động (nếu cần).

**10. Phát Hành Và Triển Khai (Release and Deployment):**

Nếu phần mềm đã qua kiểm thử và được chấp nhận, nó có thể được phát hành và triển khai cho người dùng cuối.

**11. Theo Dõi Và Bảo Trì (Monitoring and Maintenance):**

Theo dõi hiệu suất của phần mềm trong môi trường sản xuất.

Xử lý các lỗi hoặc vấn đề nếu chúng xuất hiện trong sản phẩm đã triển khai.

**CHƯƠNG 2 LẬP KẾ HOẠCH TEST**

**2.1 Giới thiệu phần mềm**

Tên dự án: Kiểm thử website Bán hàng điện tử trực tuyến

Ngày bắt đầu: 26/01/2024

Ngày kết thúc dự kiến:

Người quản lý dự án: Hoàng Ngọc Luân

Giáo viên hướng dẫn: Th. S Nguyễn Lan Oanh

Phạm vi test:

- Các chức năng không test: Lấy toàn bộ sản phẩm, Lấy thông tin chi tiết của một sản phẩm, Lấy danh sách sản phẩm theo danh mục, Đăng kí, Đăng nhập, thống kê và in hóa đơn.

Các chức năng sẽ được kiểm thử trong hệ thống bao gồm:

* Tìm kiếm, sửa, xóa thương hiệu sản phẩm
* Thêm bình luận
* Thêm sản phẩm
* Thêm thương hiệu
* Thêm nhân viên
* Thêm sản phẩm vào giỏ hàng
* Thêm loại sản phẩm
* Mua hàng tại cửa hàng
* Đặt hàng
* Duyệt đơn hàng
* Cập nhât giỏ hàng

**2.2 Nhân sự**

|  |  |  |
| --- | --- | --- |
| Nhân sự | Nhiệm vụ | Kết quả |
| Hoàng Ngọc Luân | -Quản lý dự án  -Lập kế hoạch kiểm thử  -Tìm hiểu công cụ Jmeter  -Tìm hiểu công cụ Katalon Studio  -Kiểm thử chức năng (Thêm sản phẩm)  -Kiểm thử hiệu năng  -Kiểm thử tự động  -Viết báo cáo kiểm thử |  |
| Dương Cong Long | -Tìm hiểu lý thuyết kiểm thử  -Kiểm thử chức năng (Thêm thương hiệu sản phẩm)  -Kiểm thử chức năng(Thêm loại sản phẩm, Mua hàng tại cửa hàng) |  |
| Lưu Thị Quyến | -Kiểm thử chức năng(Thêm nhân viên, Duyệt đơn hàng) |  |
| Nguyễn Thị Diễm Quỳnh | -Kiểm thử chức năng ( Thêm sản phẩm vào giỏ hàng, Cập nhật giỏ hàng ) |  |
| Phùng Ngọc Lâm | -Kiểm thử chức năng ( Tìm kiếm,Sửa, Xóa thương hiệu sản phẩm )  -Kiểm thử chức năng( Thêm bình luận, Đặt hàng ) |  |

**2.3 Công cụ**

- Công cụ và môi trường test: Katalon Studio, Jmeter, Google Chrome

**2.4 Phân tích đánh giá rủi do**

**2.5 kinh phí**

- Kinh phí để thực hiện dự án ước tính là 2.000.000 USD

**CHƯƠNG 3 GIỚI THIỆU CÔNG CỤ KATALON STUDIO VÀ JMERTER**

**3.1 Công cụ Katalon Studio**

**3.1.1 Giới thiệu về kiểm thử tự động**

- Kiểm thử tự động là quá trình sử dụng các công cụ và phần mềm để thực hiện các bước kiểm thử một cách tự động, thay vì thủ công. Mục tiêu của kiểm thử tự động là tối ưu hóa quá trình kiểm thử bằng cách giảm thiểu sự phụ thuộc vào tay công việc và tăng cường hiệu suất, đồng thời giảm thiểu sai sót con người.

**3.1.2 Giới thiệu về công cụ kiểm thử Katalon**

- Katalon là một công cụ kiểm thử tự động dựa trên nền tảng mã nguồn mở, được phát triển để hỗ trợ việc kiểm thử tự động ứng dụng web và di động. Được ra mắt vào năm 2015, Katalon nhanh chóng trở thành một trong những công cụ phổ biến được sử dụng trong cộng đồng kiểm thử phần mềm.

**3.1.3 Các tính năng chính của Katalon**

1. Giao diện người dùng thân thiện: Katalon cung cấp một giao diện người dùng đơn giản và trực quan, giúp người dùng dễ dàng tạo và quản lý các kịch bản kiểm thử.

1. Hỗ trợ đa nền tảng: Katalon hỗ trợ kiểm thử tự động cho ứng dụng web, ứng dụng di động (Android và iOS) và dịch vụ web. Điều này cho phép bạn kiểm tra và đảm bảo chất lượng cho các ứng dụng chạy trên nhiều nền tảng khác nhau.
2. Ghi và tái sử dụng kịch bản kiểm thử: Katalon cho phép bạn ghi lại các hành động của mình trên ứng dụng và tạo kịch bản kiểm thử tự động. Bạn có thể tái sử dụng các kịch bản này để thực hiện kiểm thử trên nhiều phiên bản và môi trường khác nhau.
3. Hỗ trợ nhiều ngôn ngữ lập trình: Katalon hỗ trợ nhiều ngôn ngữ lập trình như Java, C#, Groovy, Python, và Ruby. Điều này cho phép bạn sử dụng ngôn ngữ lập trình ưa thích của mình để viết các kịch bản kiểm thử.

5. Tích hợp với công cụ kiểm thử và quản lý dự án: Katalon tích hợp với các công cụ kiểm thử và quản lý dự án phổ biến như Git, JIRA, Jenkins và Azure DevOps. Điều này giúp tăng cường khả năng làm việc nhóm và quản lý dự án hiệu quả.

6. Báo cáo tự động: Katalon cung cấp các báo cáo tự động về kết quả kiểm thử, bao gồm cả báo cáo dạng đồ thị và báo cáo chi tiết về các lỗi và kết quả kiểm thử. Điều này giúp bạn đánh giá hiệu suất và chất lượng của ứng dụng.

7. Hỗ trợ cộng đồng: Katalon có một cộng đồng lớn và năng động, nơi bạn có thể trao đổi kiến thức, chia sẻ kịch bản và nhận hỗ trợ từ những người dùng khác.

**3.1.4 Ưu điểm**

1. Giao diện người dùng thân thiện: Katalon cung cấp một giao diện người dùng đơn giản và trực quan, giúp người dùng dễ dàng tạo và quản lý các kịch bản kiểm thử.

1. Hỗ trợ đa nền tảng: Katalon hỗ trợ kiểm thử tự động cho ứng dụng web, ứng dụng di động (Android và iOS) và dịch vụ web. Điều này cho phép bạn kiểm tra và đảm bảo chất lượng cho các ứng dụng chạy trên nhiều nền tảng khác nhau.
2. Tích hợp dễ dàng: Katalon tích hợp tốt với các công cụ kiểm thử và quản lý dự án phổ biến như Git, JIRA, Jenkins và Azure DevOps. Điều này giúp tăng cường khả năng làm việc nhóm và quản lý dự án hiệu quả.
3. Ghi và tái sử dụng kịch bản kiểm thử: Katalon cho phép bạn ghi lại các hành động của mình trên ứng dụng và tạo kịch bản kiểm thử tự động. Bạn có thể tái sử dụng các kịch bản này để thực hiện kiểm thử trên nhiều phiên bản và môi trường khác nhau.
4. Hỗ trợ cộng đồng lớn: Katalon có một cộng đồng năng động và rộng lớn, nơi bạn có thể trao đổi kiến thức, chia sẻ kịch bản và nhận hỗ trợ từ những người dùng khác.

**3.1.5 Nhược điểm**

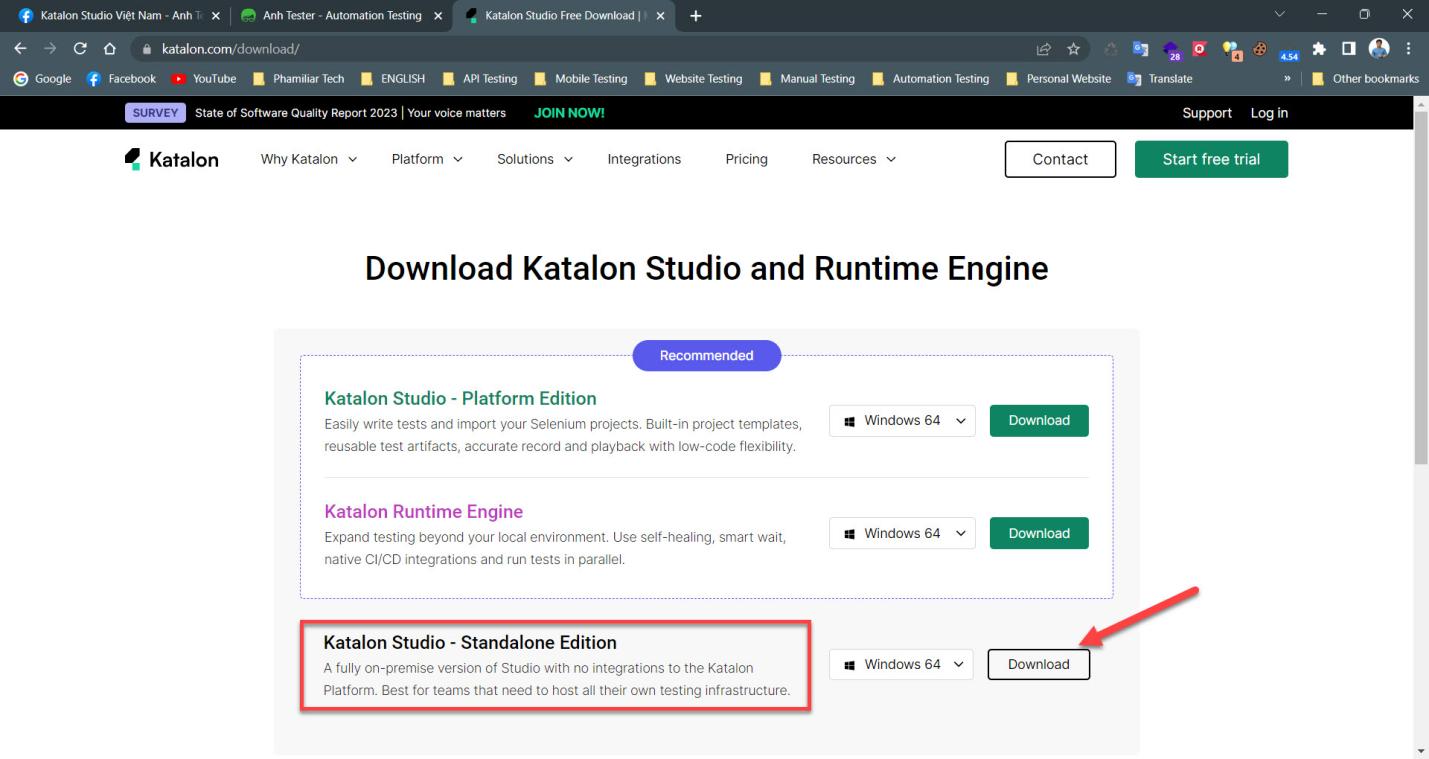
1. Hạn chế trong việc tùy chỉnh: Mặc dù Katalon cung cấp một số tính năng tùy chỉnh, nhưng nó có thể hạn chế so với một số công cụ kiểm thử tự động khác. Điều này có thể gây khó khăn cho việc đáp ứng các yêu cầu đặc biệt và phức tạp của dự án.

1. Framework mới nổi với một cộng đồng phát triển nhanh chóng.
2. Ngôn ngữ kịch bản hạn chế: chỉ hỗ trợ cho Java/Groovy
3. Hiệu suất: Trong một số trường hợp, Katalon có thể gặp vấn đề với hiệu suất khi thực hiện kiểm thử trên ứng dụng lớn và phức tạp. Việc xử lý dữ liệu và tương tác với ứng dụng có thể mất thời gian và gây ra sự chậm trễ.
4. Khả năng mở rộng: Khi thực hiện kiểm thử cho các dự án lớn và phức tạp, Katalon có thể gặp khó khăn trong việc mở rộng và quản lý quy mô lớn của các kịch bản kiểm thử.

**3.1.6 Hướng dẫn cài đặt**

**Download Katalon tại link sau:** [https://katalon.com/download/](https://katalon.com/download/" \t "https://www.anhtester.com/blog/katalon-tool/_blank).

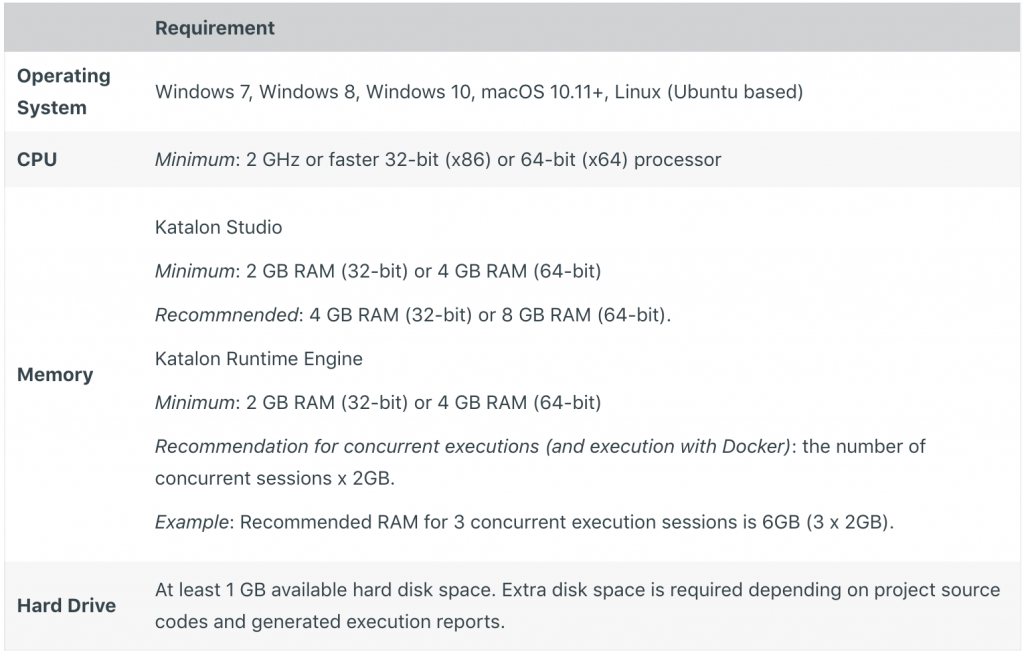
Tại đây, sẽ có 3 phiên bản khác nhau được phân biệt như sau:



**Katalon Studio (Standalone Edition)**: Được sử dụng như một IDE để viết test case, scripting bình thường nhất  
  
**Katalon Studio (Platform Edition)**: Được sử dụng như một IDE để viết test case, scripting nhưng connect tới dịch vụ Git để quản lý chạy Integrations sau này tiện lợi hơn. (chổ này buộc tạo nhánh Repository trên Git trước sau đó connect vào TestOps của Katalon)  
  
**Katalon Runtime Engine (KRE)**: Hỗ trợ chạy test trong chế độ Command Line Interface (CLI)  
  
*Tùy vào mục đích sử dụng, bạn download version phù hợp với mình nhé*

Hiện tải có các phiên bản có thể download: MacOS, Linux, Win7, Win8, Win10.

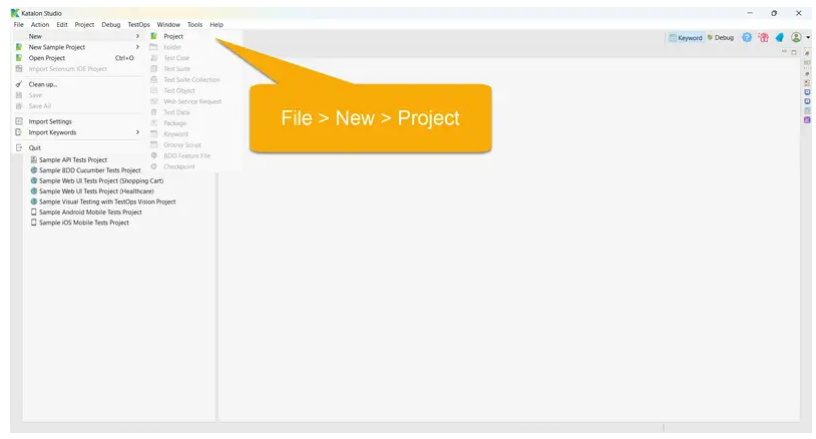
**Cấu hình cần thiết cho cài đặt**

****

**3.1.7 Hướng dẫn sử dụng**

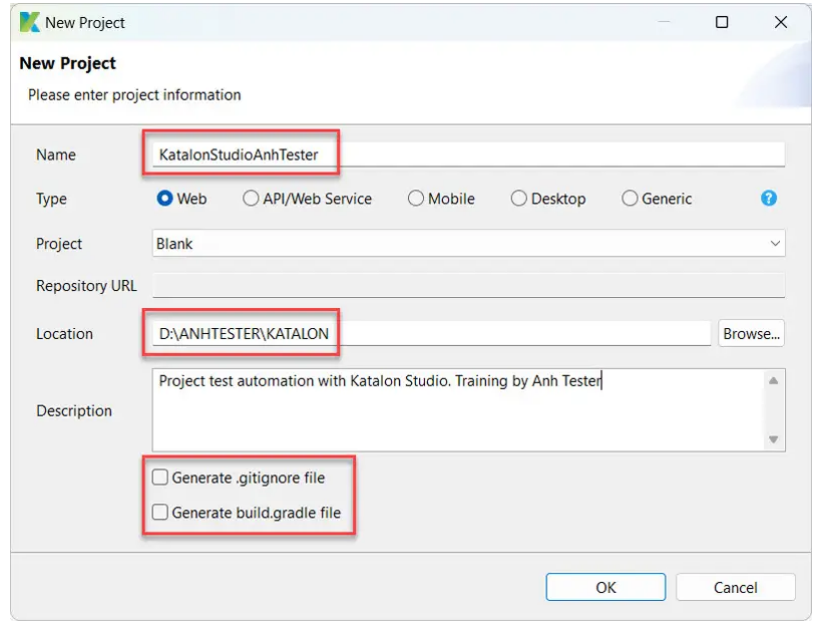
Sau khi đăng nhập vào Katalon Studio thành công thì bạn tiến hành tạo Project đầu tiên như sau:

**Bước 1**: Bạn chọn vào **File** => **New** => **Project**(hoặc nhấn vào **New Project**tại phần**Test Explorer**bên trái) của giao diện chính.

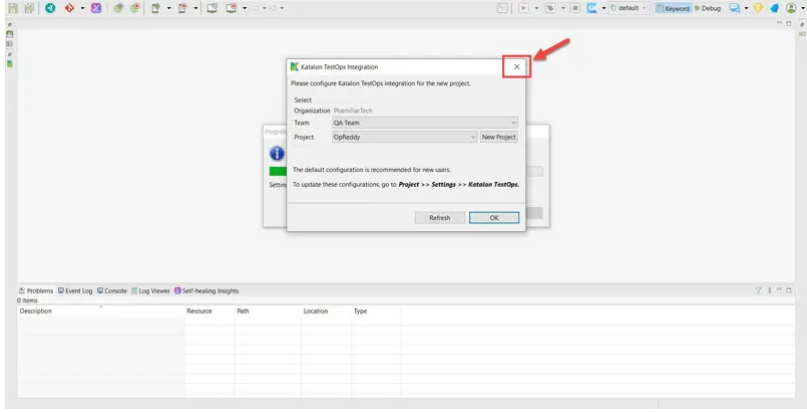


**Bước 2**: Ở Tab **New Project** thì bạn tiến hành đặt tên cho dự án còn chọn vị tí lưu dự án. Chú ý không nên để cùng foder với file cài **Katalon Studio**.

**Bước 3**: Tại phần **Type**có các loại ứng dụng kiểm thử cho bạn chọn gồm **Web, API/Web Service, Mobile hay Desktop** thì ở đây bạn sẽ tick chọn vào **Web**. Ngoài ra bạn có thể thay đổi các thông số khác hoặc để mặc định. Ở 2 ô**Generate** phía dưới mục **Description** thì bạn bỏ tick.

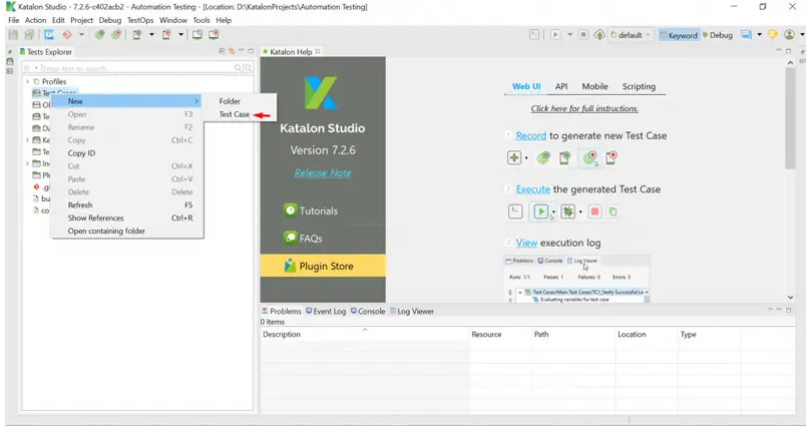


**Bước 4**: Khi tab **Katalon TestOps Integration**hiện ra thì tức là bạn đã tạo dự án thành công. Ở đây là nơi bạn upload các kết quả kiểm thử hay **Test Report**lên **TestOps cloud.**

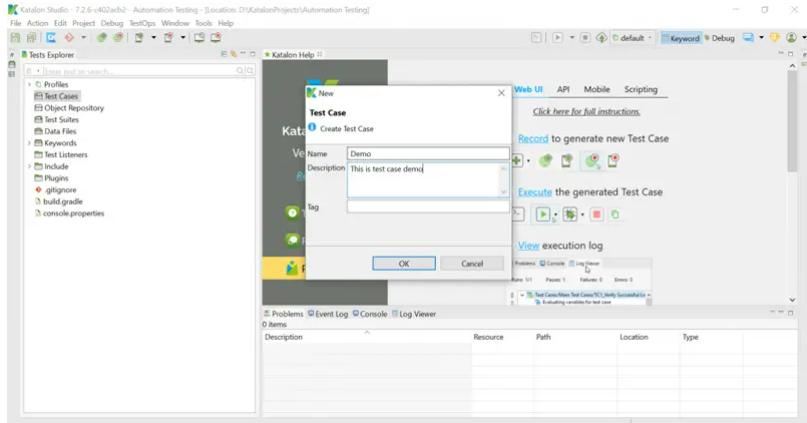


## Cách tạo một Test case với Katalon Studio

Bước 1: Trước tiên bạn chọn **File** => **New** => **Test Case**hoặc tại **Tests Explorer** bạn nhấn chuột phải vào mục **Test Cases**chọn **New => Test case**.



Bước 2: Kế tiếp bạn tiến hành đặt tên cho **Test case** và ấn **OK**. Sau đó màn hình sẽ hiển thị các tab: Manual mode, Script mode, Variables, Integration, Properties. Tại đây bạn có thể thực hiện tùy chỉnh với các Tab này.



**3.2 Công cụ Jmeter**

**3.2.1 Giới thiệu về kiểm thử trên Jmeter**

JMeter là một công cụ mã nguồn mở và miễn phí được Apache Software Foundation phát triển. Nó được sử dụng rộng rãi để kiểm thử hiệu năng và tải trong các ứng dụng web và dịch vụ web. JMeter cho phép bạn tạo, thực thi và phân tích các kịch bản kiểm thử để đo lường và đánh giá hiệu suất của hệ thống.

**3.2.2 Các tính năng chính của Jmeter**

1.Giao diện người dùng đồ họa (GUI): JMeter cung cấp một giao diện người dùng đồ họa thân thiện và trực quan, giúp người dùng dễ dàng tạo và cấu hình các kịch bản kiểm thử. Bạn có thể tạo các yêu cầu HTTP, FTP, JDBC và nhiều loại khác thông qua giao diện GUI.

1. Hỗ trợ nhiều giao thức: JMeter hỗ trợ nhiều giao thức như HTTP, HTTPS, FTP, JDBC, SOAP, REST, JMS và nhiều giao thức khác. Điều này cho phép bạn kiểm tra hiệu suất của các ứng dụng web, dịch vụ web và hệ thống cơ sở dữ liệu.

1. Thiết kế kịch bản kiểm thử linh hoạt: Bằng cách sử dụng JMeter, bạn có thể tạo ra các kịch bản kiểm thử phức tạp với nhiều yêu cầu song song, thời gian chờ ngẫu nhiên, định thời, vòng lặp và điều kiện logic. Điều này giúp mô phỏng hành vi người dùng thực tế và kiểm tra khả năng chịu tải của hệ thống.
2. Tạo tải và kiểm thử tải lớn: JMeter có thể tạo tải lớn bằng cách mô phỏng số lượng người dùng đồng thời và tốc độ yêu cầu trong một khoảng thời gian nhất định. Điều này cho phép bạn kiểm tra khả năng chịu tải của ứng dụng và xác định ngưỡng tải mà hệ thống có thể xử lý.
3. Thu thập và phân tích kết quả: JMeter cung cấp các Listener để thu thập kết quả từ quá trình kiểm thử. Bạn có thể xem thông tin chi tiết về thời gian phản hồi, số lỗi, thống kê và các thông số khác. Bên cạnh đó, JMeter cũng cung cấp khả năng xuất báo cáo dưới dạng HTML, CSV và XML để phân tích kết quả.

6.Mở rộng và tùy chỉnh: JMeter là một công cụ mở rộng mà bạn có thể tùy chỉnh và mở rộng theo nhu cầu của mình. Bạn có thể viết các plugin, mở rộng chức năng và tích hợp với các công cụ và khung làm việc khác.

**3.2.3 Ưu điểm**

Ưu điểm của JMeter:

1. Miễn phí và mã nguồn mở: JMeter là một công cụ mã nguồn mở và miễn phí, giúp tiết kiệm chi phí và cho phép người dùng tùy chỉnh và mở rộng theo nhu cầu.

1. Hỗ trợ đa nền tảng: JMeter hỗ trợ trên nhiều nền tảng, bao gồm Windows, Linux và macOS, cho phép bạn chạy kiểm thử trên các môi trường khác nhau.

1. Hỗ trợ nhiều giao thức: JMeter hỗ trợ nhiều giao thức như HTTP, HTTPS, FTP, JDBC, SOAP, REST, JMS và nhiều giao thức khác. Điều này giúp bạn kiểm tra hiệu suất của các ứng dụng web, dịch vụ web và hệ thống cơ sở dữ liệu.
2. Giao diện người dùng đồ họa (GUI) thân thiện: JMeter cung cấp giao diện người dùng đồ họa dễ sử dụng và trực quan, giúp người dùng tạo và cấu hình các kịch bản kiểm thử một cách dễ dàng.
3. Linh hoạt và mở rộng: JMeter cho phép tạo ra các kịch bản kiểm thử phức tạp với nhiều tham số tùy chỉnh như số lượng luồng, thời gian chờ, điều kiện logic và vòng lặp. Bạn cũng có thể viết các plugin và mở rộng chức năng của JMeter theo nhu cầu của mình.
4. Phân tích kết quả chi tiết: JMeter cung cấp các Listener để thu thập và hiển thị kết quả kiểm thử chi tiết như thời gian phản hồi, số lỗi, thống kê và các thông số khác. Bạn cũng có thể xuất báo cáo để phân tích kết quả.

**3.2.4 Nhược điểm**

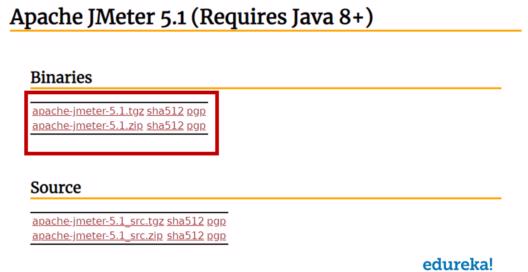
Nhược điểm của JMeter:

1. Giao diện người dùng phức tạp: Một số người dùng có thể gặp khó khăn trong việc sử dụng giao diện người dùng của JMeter, đặc biệt là khi làm việc với các tính năng phức tạp. Điều này có thể đòi hỏi thời gian và đào tạo để làm quen và tận dụng được toàn bộ tiềm năng của công cụ.
2. Yêu cầu tài nguyên hệ thống lớn: Khi chạy các kiểm thử có tải lớn hoặc phức tạp, JMeter có thể yêu cầu nhiều tài nguyên hệ thống như bộ nhớ và CPU. Điều này có thể làm giảm hiệu suất của hệ thống và hạn chế khả năng chạy đồng thời nhiều luồng kiểm thử.
3. Khả năng mở rộng: Mặc dù JMeter có khả năng mở rộng thông qua việc viết plugin và mở rộng chức năng, quá trình phát triển và triển khai các plugin có thể phức tạp và đòi hỏi kiến thức sâu về JMeter và Java.
4. Độ tin cậy và ổn định: Trong một số trường hợp, JMeter có thể gặp phải vấn đề về độ tin cậy và ổn định, đặc biệt khi chạy các kiểm thử có tải lớn. Điều này có thể làm mất mát dữ liệu hoặc giảm chính xác của kết quả kiểm thử.
5. Hạn chế trong việc kiểm thử các ứng dụng phức tạp: Mặc dù JMeter hỗ trợ nhiều giao thức và có khả năng linh hoạt, nó có thể gặp hạn chế khi kiểm thử các ứng dụng phức tạp với các tương tác phức tạp và yêu cầu chức năng đặc biệt.
6. Học tập và đào tạo: JMeter có một số khái niệm và tính năng phức tạp, việc học và làm quen với công cụ này có thể đòi hỏi một thời gian và sự đào tạo ban đầu. Điều này có thể làm tăng độ khó cho người mới sử dụng JMeter.
7. Khả năng quản lý phiên bản và tích hợp: JMeter không cung cấp các tính năng quản lý phiên bản tích hợp, điều này có thể gây khó khăn trong việc quản lý mã nguồn và hợp tác nhóm trong các dự án phức tạp.

**3.2.5. Hướng dẫn cài đặt**

Bước 1. Cài đặt Java

Bước 2 . Tải xuống



Bước 3. Cài đặt

**CHƯƠNG 4 GIỚI THIỆU WEBSITE BÁN HÀNG ĐIỆN TỬ T&T**

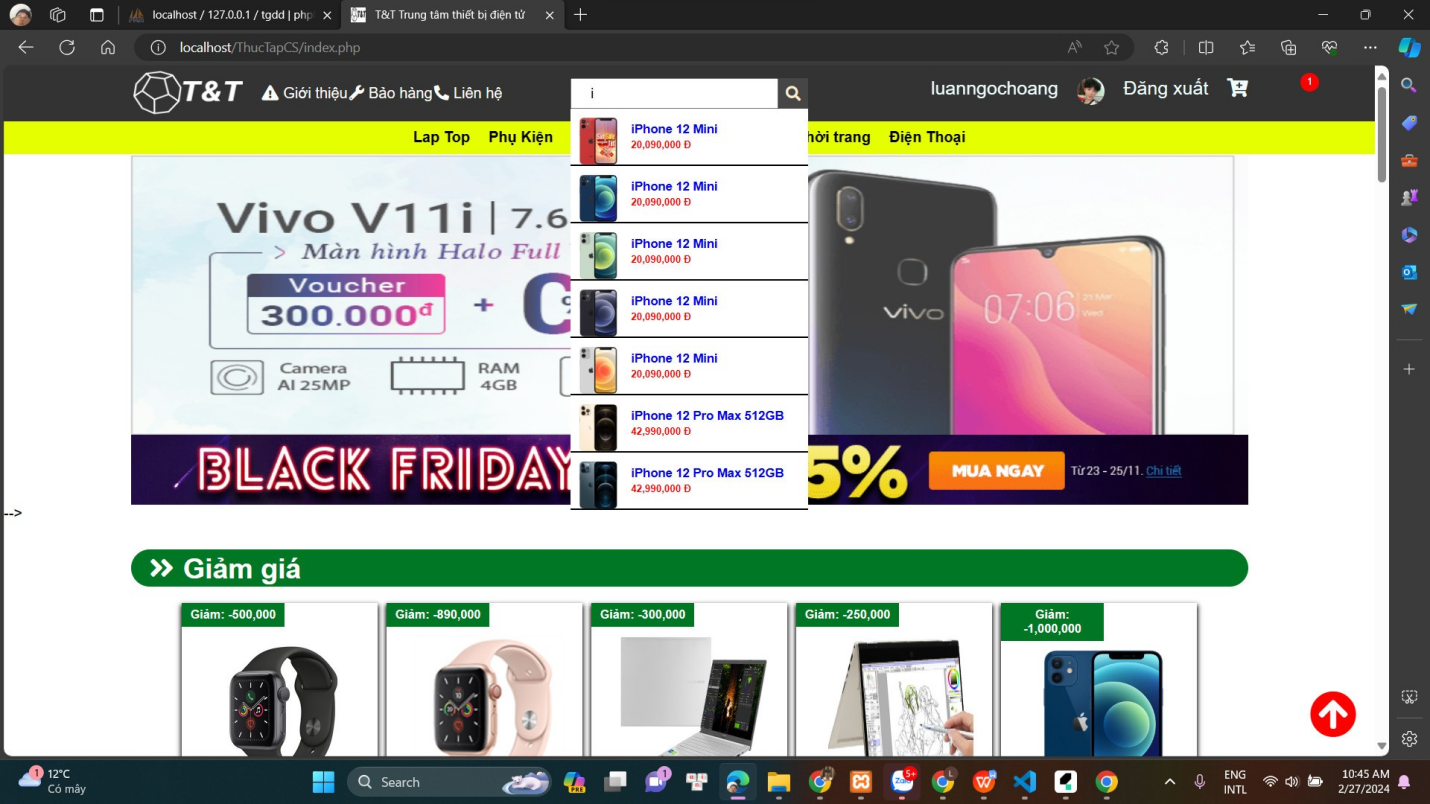
**4.1 Mô tả chung về sản phẩm phần mềm**

Website bán hàng điện tử T&T là một trang web chuyên cung cấp các sản phẩm điện tử và công nghệ thông tin cho người tiêu dùng.  Đây là một website bán hàng điện tử, là một nền tảng trực tuyến cho phép người dùng mua sắm và tiến hành các giao dịch mua bán trực tuyến. Qua website này, người dùng có thể tìm kiếm, so sánh và mua các sản phẩm và dịch vụ đa dạng một cách thuận tiện và an toàn.

**4.2 Đặc tả chức năng và demo**

|  |  |  |
| --- | --- | --- |
| **Tên chức năng** | **Công dụng** | **Demo giao diện** |
| **Chức năng tìm kiếm** | - Tìm kiếm theo từ khóa  - Lọc kết quả  - Gợi ý tìm kiếm |  |
| **Chức năng thêm sản phẩm** | Đây là quy trình cho phép người bán thêm các sản phẩm mới vào danh sách sản phẩm trên trang web:  - Điền thông tin sản phẩm  - Danh mục sản phẩm |  |
| **Chức năng bình luận** | Cho phép khách hàng để lại ý kiến, bình luận, đánh giá phản hồi về sản phẩm |  |
| **Chức năng đặt hàng** | Cho phép khách hàng lên đơn và đặt hàng thành công sản phẩm |  |
| **Chức năng** |  |  |

* **Giao diện tìm kiếm sản phầm**

****

**Đặc tả chức năng**

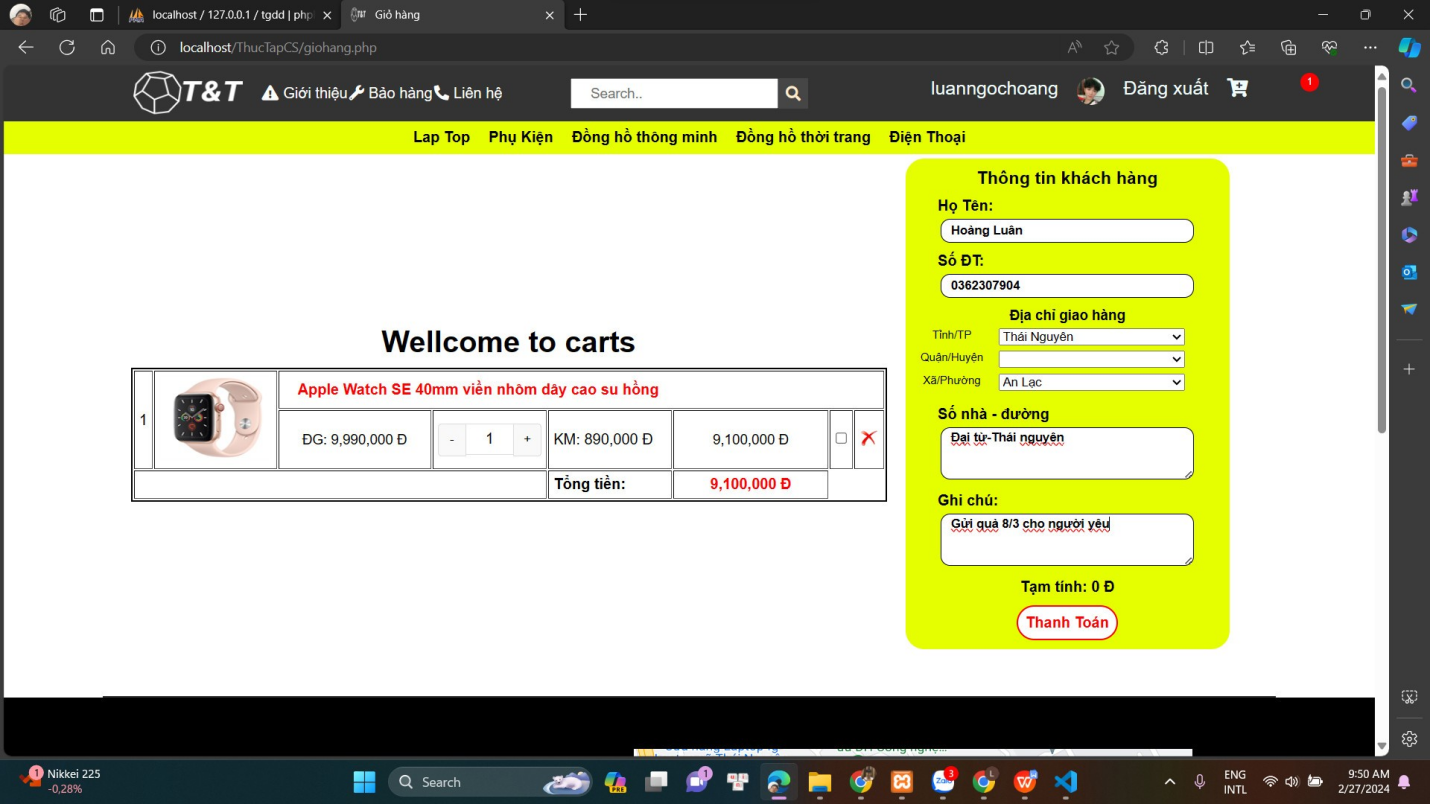
**Tìm kiếm sản phẩm**

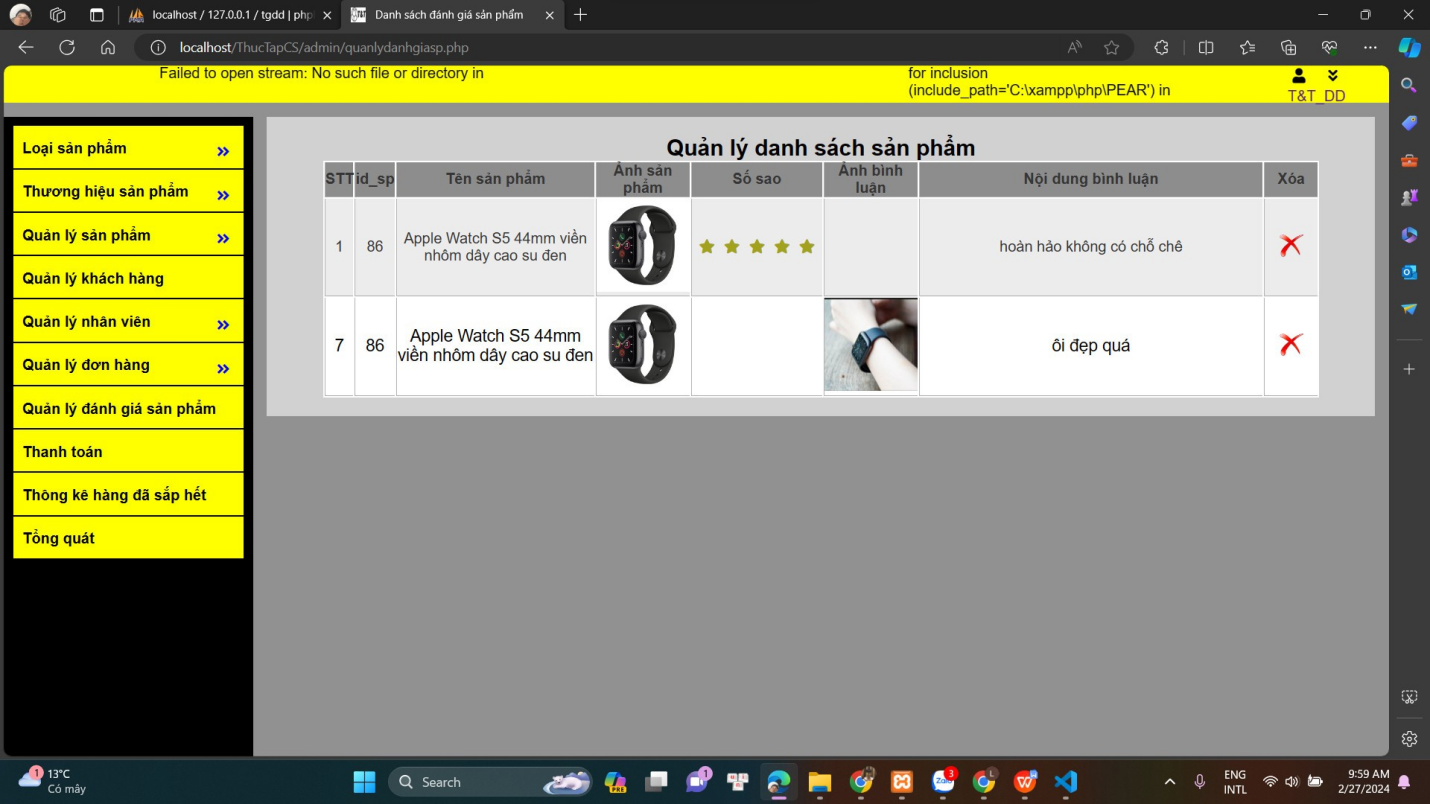
khách hàng nhập từng kí tự mà có chứa trong tên sản phẩm. Hệ thống sẽ tự động gợi ý sản phẩm mà khách hàng cần tìm.

Nếu có sản phẩm được gợi ý thì hiện thị tên sản phẩm, hình ảnh sản phẩm và giá sản phẩm ở dưới chỗ nhập tìm kiếm.

Nếu không có hiển thị sản phẩm gợi ý ở bên dưới thì không có sản phẩm mà bạn cần tìm.

* **Giao diện giỏ hàng và thanh toán**

****

* **Giao diện bình luận sản phẩm** ****

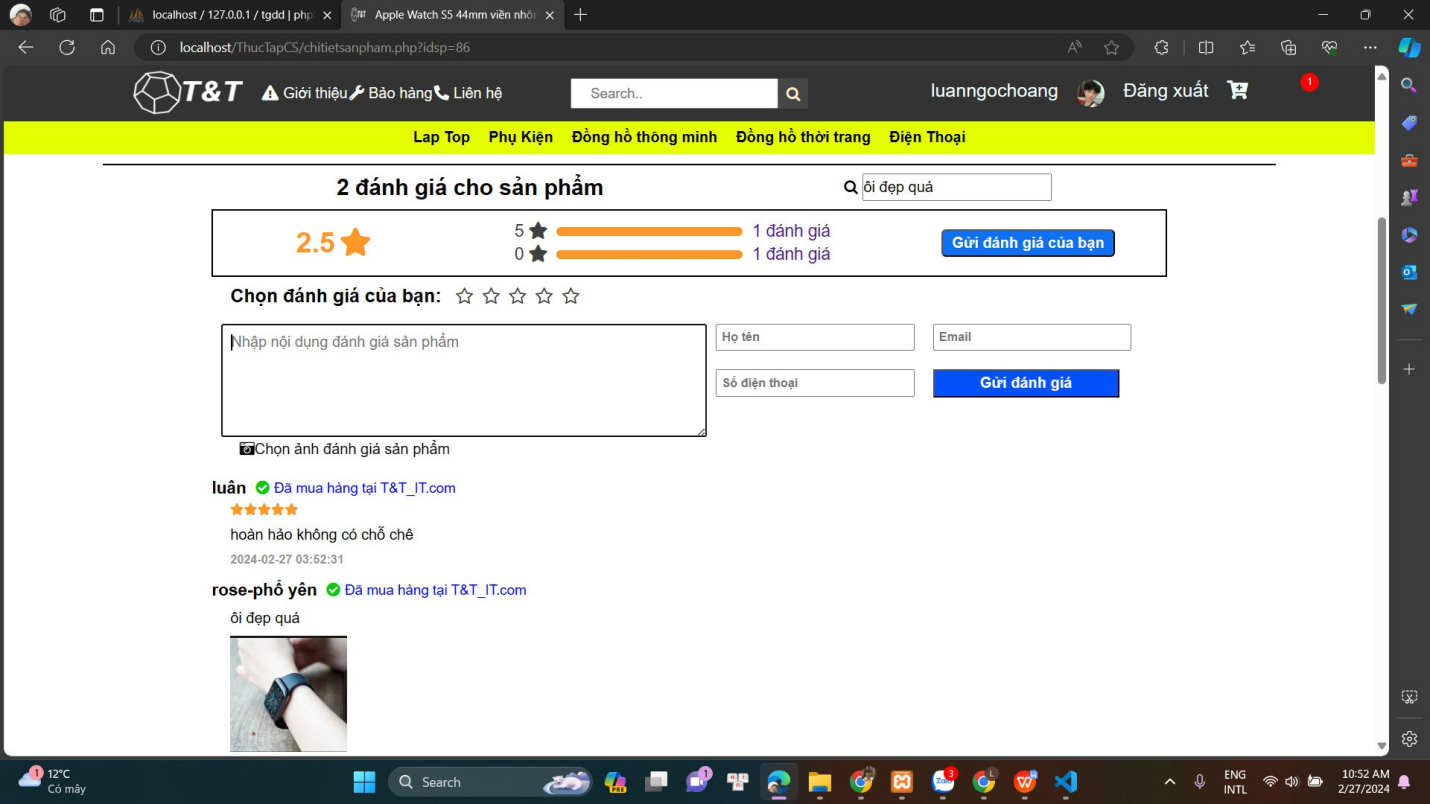
**Đặc tả chức năng**

**Thêm bình luận:**

Khách hàng sẽ đánh giá số sao về sản phẩm (5 mức đánh giá sao) và cũng như có thể bình luận, tải ảnh sản phẩm.

Nếu đáp ứng đủ yêu cầu mà hệ thống yêu cầu khách hàng đánh giá thì thêm bình luận thành công ngược lại thì thông báo thêm bình luận không thành công.

* **Giao diện tìm kiếm bình luận**

****

**Đặc tả chức năng**

**Tìm kiếm đánh giá sản phẩm (bình luận):**

Admin nhập vào ô tìm kiếm những từ tích cực hay tiêu cực để tìm kiếm bình luận.

Admin xóa bình luận đã tìm được.

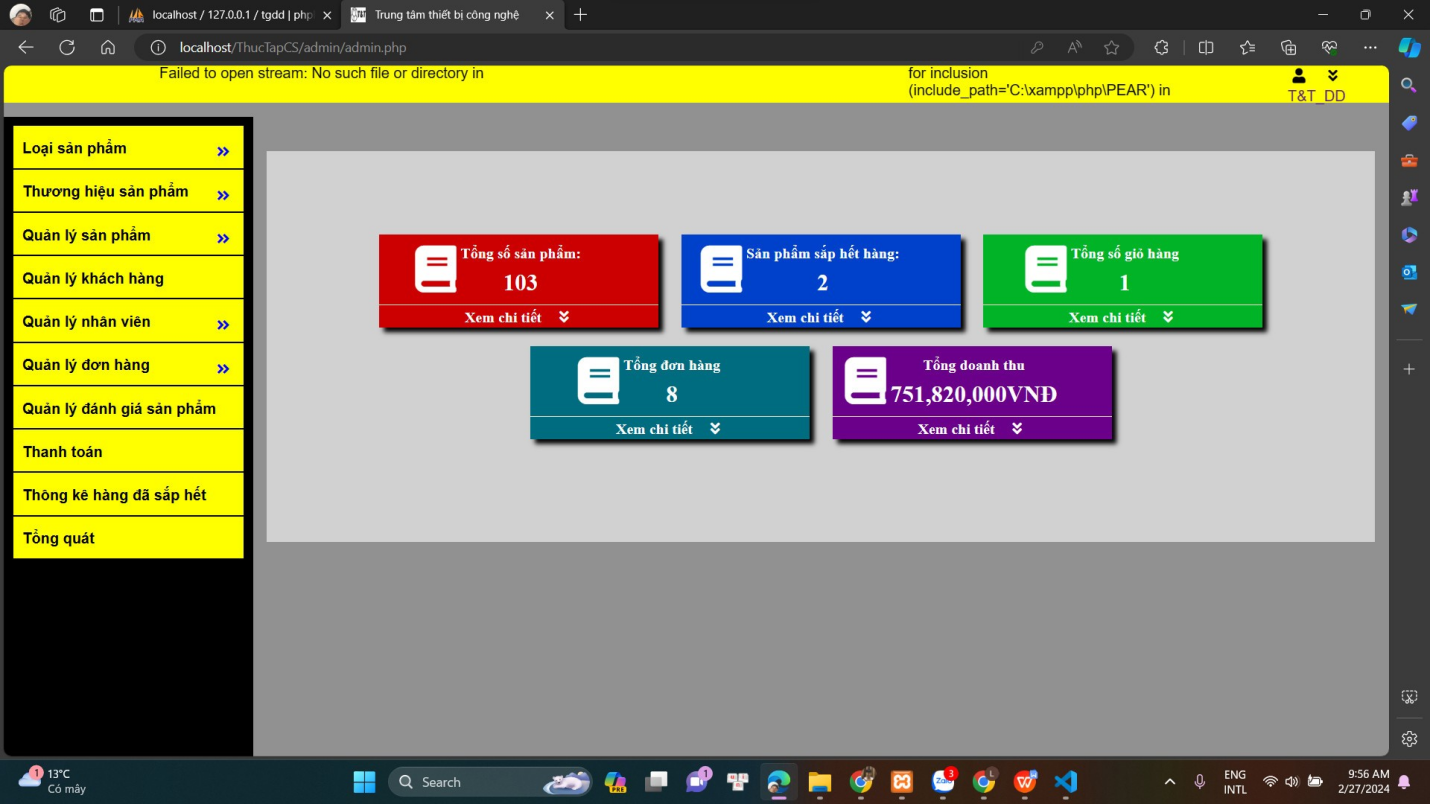
Tìm kiếm danh mục sản phẩm:

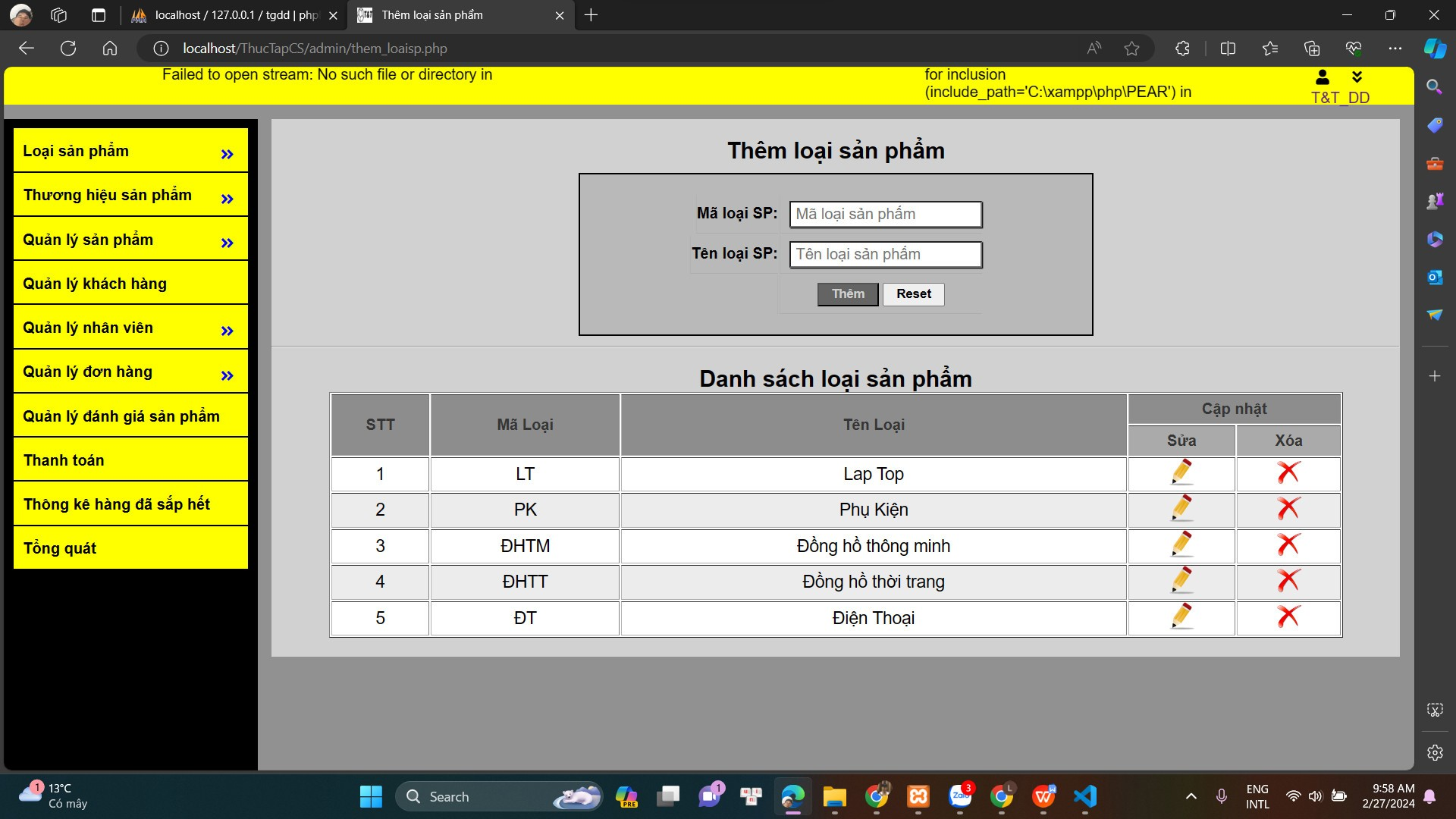
Admin nhập từng kí tự vào ô tìm kiếm liên quan đến loại sản phẩm hay tên sản phẩm để tìm kiếm.

Nếu tìm kiếm thành công sẽ hiện thị danh sách bảng ở dưới ô tìm kiếm.

Nếu tìm kiếm không thành công sẽ hiện thị ra dòng chữ “Không có loại sản phẩm bạn cần tìm” .

**4.2.1 Giao diện phía Admin**

* **Giao diện trang chủ**
* ****
* **Giao diện thêm,sửa ,xóa loại sản phẩm**



**Đặc tả chức năng**

**Thêm danh mục sản phẩm:**

Admin nhập thông tin cần thiết mà hệ thống yêu cầu.

Nếu nhập thành công thì sẽ thông báo thêm danh mục sản phẩm thành công và tự động hiện thị danh mục sản phẩm vừa mới nhập.

Nếu không thành công thì thông báo thêm danh mục sản phẩm không thành công và yêu cầu admin nhập lại thông tin.

**Sửa danh mục sản phẩm:**

Admin chọn những danh mục sản phẩm mà admin cần sửa.

Hệ thống sẽ đi đến trang sửa thông tin danh mục sản phẩm yêu cầu admin sửa thông tin danh mục sản phẩm mà admin cần sửa.

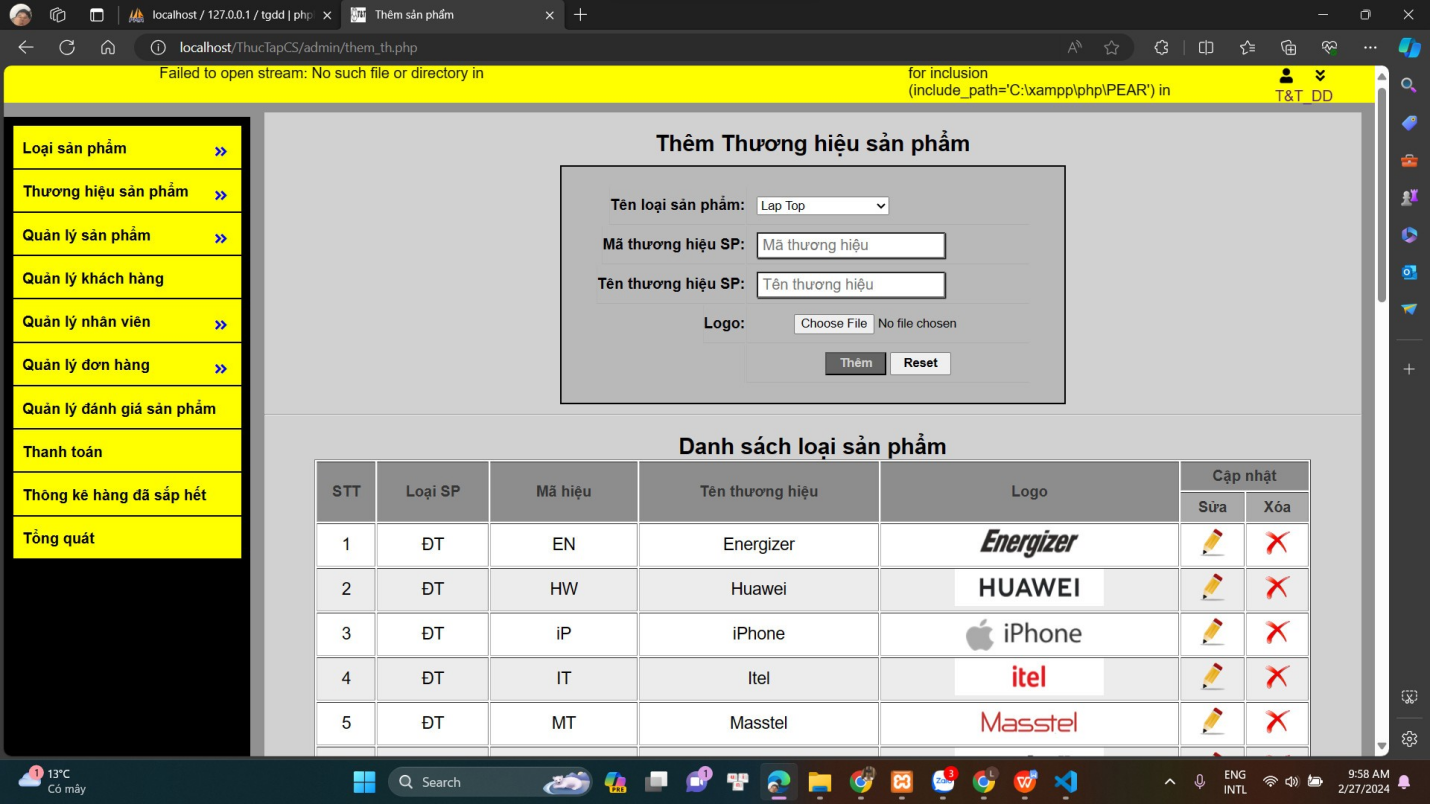
Nếu sứa thành công hệ thống sẽ tự động cập nhật ngay bên dưới để admin dễ kiểm tra danh mục đã sửa thành công như mong muốn chưa.

**Xóa danh mục sản phẩm:**

Admin chọn những danh mục sản phẩm cần xóa.

Nếu xóa thành công thì hệ thống xóa danh mục sản phẩm mà bạn đã chọn sau đó hệ thống cập nhật lại trang web hiện thị lại danh sách danh mục sản phẩm.

**Giao diện thêm, sửa, xóa thương hiệu**

****

**Đặc tả chức năng**

**Tìm kiếm thương hiệu:**

Admin và nhân viên nhập từng ký tự vào ô tìm kiếm để tìm những tên thương hiệu hay là nhập mã loại sản phẩm.

Nếu tồn tài những mã loại sản phẩm hay là tên thương hiệu thì hệ thống tự động hiển thị những thương hiệu mà bạn cần tìm.

Nếu không tồn tại thì hệ thống hiển thị dòng chữ ở dưới ô nhập tìm kiếm là ‘Không có thương hiệu nào bạn cần tìm’.

**Thêm thương hiệu:**

Admin chọn thêm thương hiệu để đi đến trang thêm thương hiệu.

Khi đi đến trang thương hiệu hệ thống yêu cầu nhập đầy đủ thông tin như mã loại sản phẩm, mã thương hiệu, tên thương hiệu và hình ảnh thương hiệu.

Nếu thêm thành công thì hệ thống sẽ cập nhật lại trang và hiện thị thông báo thêm thương hiệu thành công.

Nếu thêm không thành công thì hệ thống sẽ thông báo thêm thương hiệu thất bại và báo lỗi cần cập nhật.

**Sửa thương hiệu:**

Admin chọn thương hiệu cần sửa trong danh sách thương hiệu.

Sau khi chọn xong hệ thống đưa đến trang sửa thương hiệu sau đó admin cần chỉnh sửa những thông tin mà admin cần sửa.

Nếu sửa thành công thì hệ thống sẽ hiển thị thương hiệu đã sửa ngay dưới để bạn có thể kiểm tra ngay.

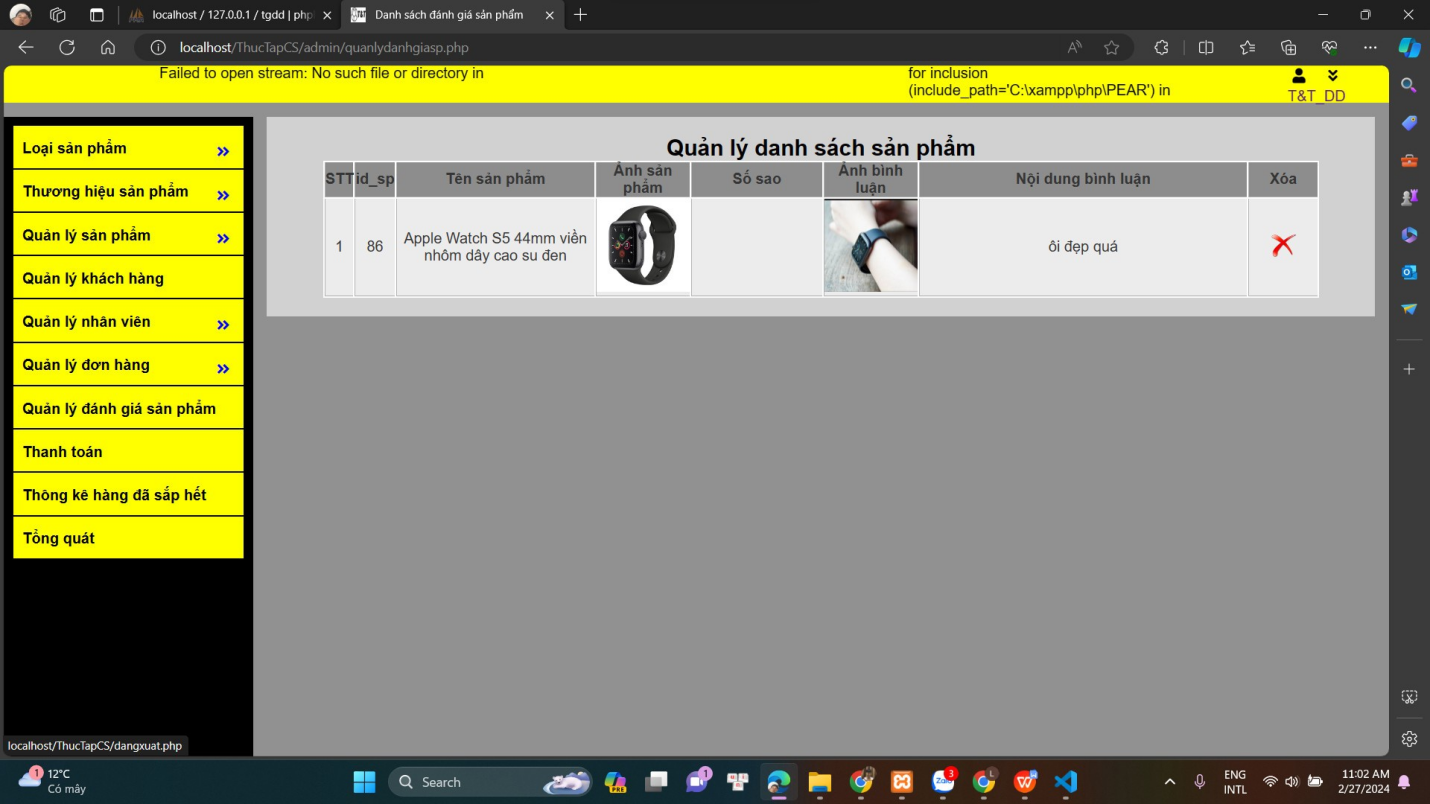
Nếu sửa không thành công thì hệ thống thống báo sửa không thành công.

**Xóa thương hiệu:**

Admin nhấp chuột vào icon xóa trong danh sách thương hiệu sản phẩm.

Sau khi nhấp chuột vào hệ thống sẽ cập nhật lại danh sách thương hiệu để admin kiểm tra có xóa được thương hiệu sản phẩm hay không

**Giao diện xóa bình luận**

****

**Đặc tả chức năng**

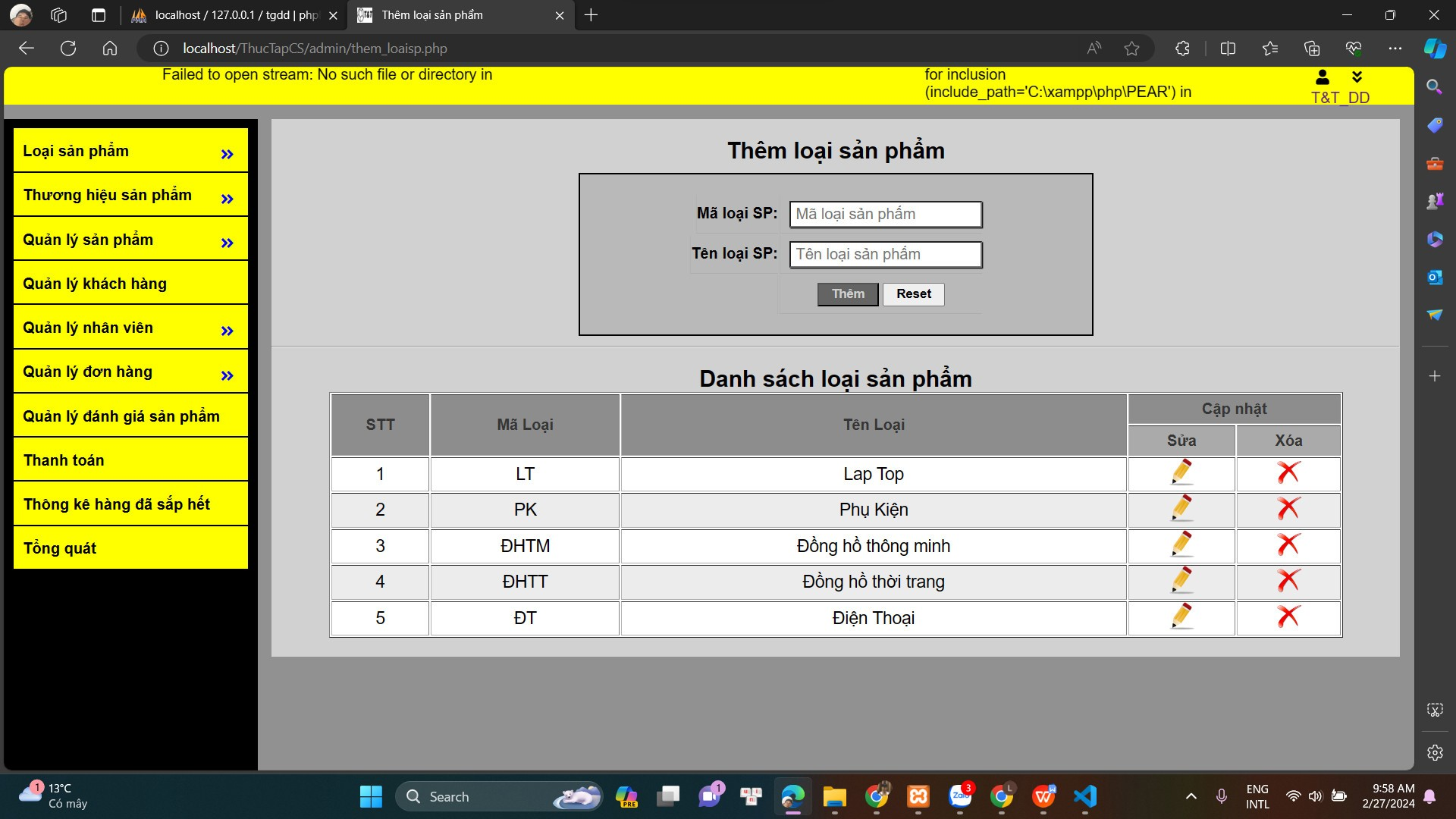
**Xóa đánh giá sản phẩm (bình luận):**

Admin tìm kiếm những bình luận có những lời tiêu cực đến sản phẩm và chọn những bình luận mà admin cần xóa.

Nếu xóa thành công thì thông báo xóa thành công và hệ thống tải lại trang bình luận.

Nếu xóa không thành công thì hệ thống thông báo xóa bình luận không thành công

**Giao diện thêm, sửa, xóa sản phẩm**

****

**Đặc tả chức năng**

**Tìm kiếm sản phẩm:**

Admin và nhân viên nhập từng ký tự vào ô tìm kiếm để tìm những tên sản phẩm hay là id sản phẩm.

Nếu tồn tài những sản phẩm hay id sản phẩm thì hệ thống tự động hiển thị những sản phẩm mà bạn cần tìm.

Nếu không tồn tại thì hệ thống hiển thị dòng chữ ở dưới ô nhập tìm kiếm là ‘Không có sản phẩm nào bạn cần tìm’.

**Thêm sản phẩm:**

Admin chọn thêm sản phẩm để đi đến trang sản phẩm.

Khi đi đến trang thêm sản phẩm hệ thống yêu cầu nhập đầy đủ thông tin như mã loại sản phẩm, mã thương hiệu, tên sản phẩm,… và hình ảnh sản phẩm cần phải có.

Nếu thêm thành công thì hệ thống sẽ cập nhật lại trang và hiện thị thông báo thêm sản phẩm thành công.

Nếu thêm không thành công thì hệ thống sẽ thông báo thêm sản phẩm thất bại và báo lỗi cần cập nhật.

**Sửa sản phẩm:**

Admin chọn sản phẩm cần sửa trong danh sách sản phẩm.

Sau khi chọn xong hệ thống đưa đến trang sửa sản phẩm sau đó admin cần chỉnh sửa những thông tin mà admin cần sửa.

Nếu sửa thành công thì hệ thống sẽ hiển thị sản phẩm đã sửa ngay dưới để bạn có thể kiểm tra ngay.

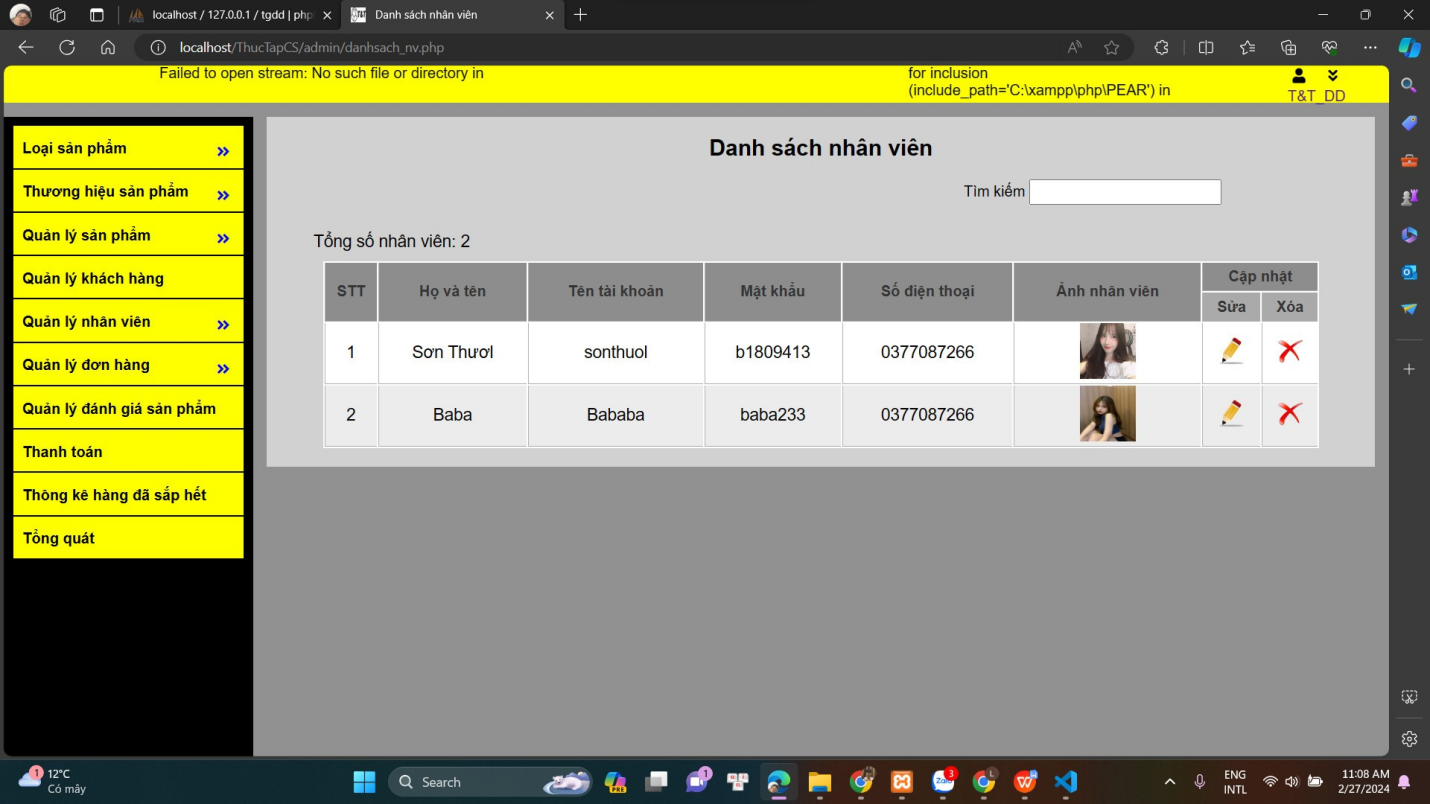
Nếu sửa không thành công thì hệ thống thống báo sửa không thành công.

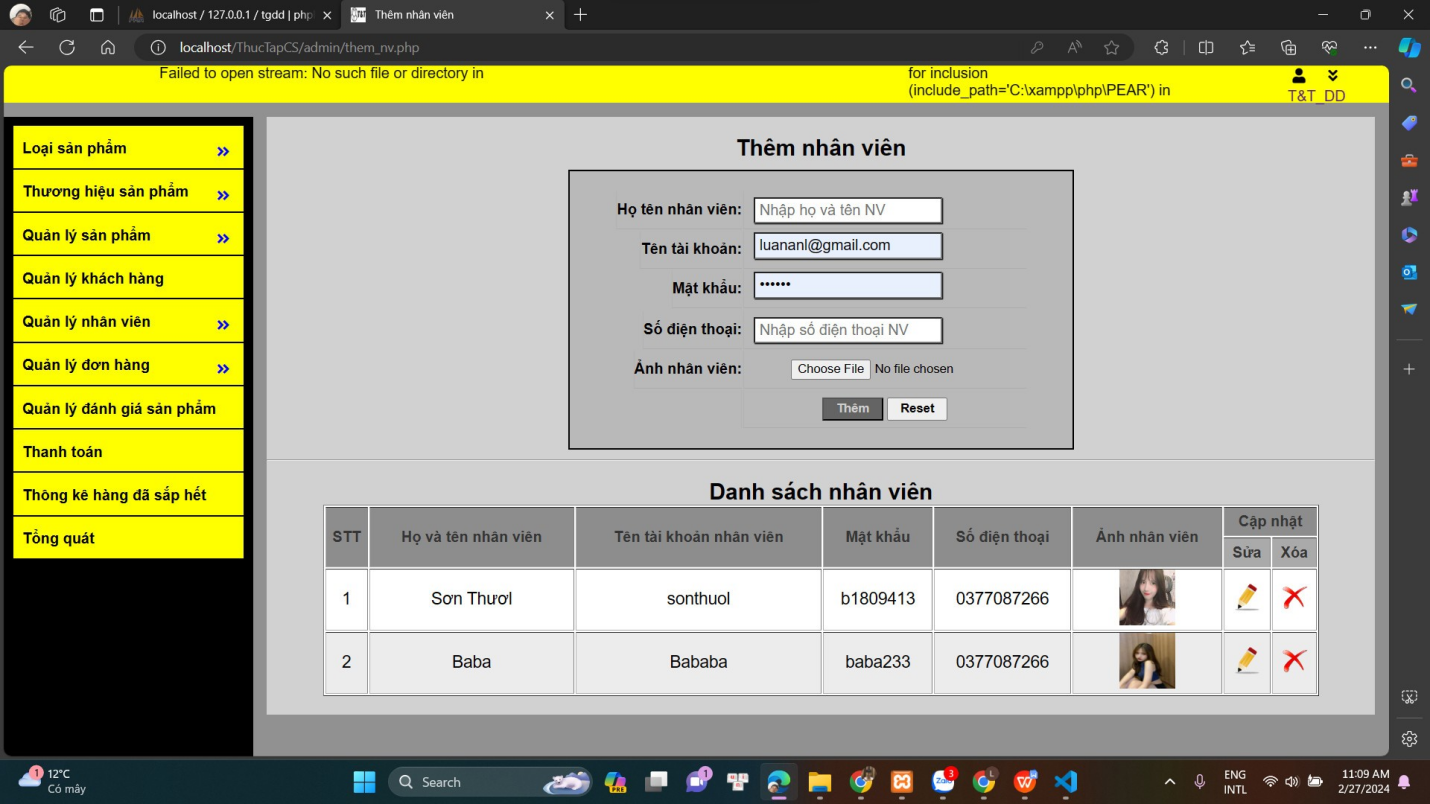
**Xóa sản phẩm:**

Admin nhấp chuột vào icon xóa trong danh sách sản phẩm.

Sau khi nhấp chuột vào hệ thống sẽ cập nhật lại danh sách để admin kiểm tra có xóa được sản phẩm hay không.

**Giao diện thêm,sửa,xóa nhân viên**

****

****

**Tìm nhân viên:**

Admin nhập mã nhân viên hay nhập tên nhân viên vào ô tìm kiếm. Hệ thống tự động tìm kiếm và hiển thị ngay dưới ô tìm kiếm.

Nếu tìm kiếm không thành công hệ thống sẽ hiển thị thông tin ‘Không có nhân viên nào mà bạn cần tìm’.

**Thêm nhân viên:**

Admin chọn chức năng thêm nhân viên trong trang quản lý nhân viên

Admin sẽ nhập thông tin cần thiết mà hệ thống yêu cầu.

Nếu thêm thành công hệ thống sẽ thông báo thêm nhân viên thành công.

Nếu thêm thất bại hệ thống sẽ thống báo thêm nhân viên không thành công.

**Xóa nhân viên:**

Admin vào trang quản lý nhân viên sau đó chọn nhân viên mà admin cần xoá.

Nếu xoá thành công hệ thống tự động cập nhật lại trang danh sách nhân viên để admin kiểm tra nhân viên đó đã xoá chưa.

Nếu xoá thất bại thì hệ thống thông báo cho admin biết xoá nhân viên thất bại.