**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC VĂN HIẾN**

**KHOA CÔNG NGHỆ - THÔNG TIN**

****

**Kiểm thử phần mềm**

**ĐỀ TÀI:**

**KIỂM THỬ TỰ ĐỘNG WEBSITE TIKI**

**GVHD: Nguyễn Thị Phương Trang**

**SVTH: Lê Nguyễn Hoàn Huy**

**MSSV: 211A290034**

**SVTH: Đặng Vương Thái Đăng**

**MSSV: 211A010010**

**SVTH: Phạm Dương Công Phúc**

**MSSV: 211A010356**

**SVTH: Nguyễn Tuấn Kiệt**

**MSSV: 211A290020**

**LỚP HỌC PHẦN: INT47401**

***Tp.Hồ Chí Minh, tháng 12 năm 2024***

**NHẬN XÉT CỦA GIẢNG VIÊN**

…………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………..…………………………………………………………..……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………....

**Mục lục**

[I. Giới thiệu 5](#_Toc184914560)

[1.1 Mô tả ứng dụng kiểm thử 5](#_Toc184914561)

[1.2 Mục tiêu đồ án 5](#_Toc184914562)

[1.3 Các chức năng/module thực hiện kiểm thử 5](#_Toc184914563)

[1.4 Môi trường, công cụ và các giả định 5](#_Toc184914564)

[1.5 Nhân sự thực hiện 6](#_Toc184914565)

[1.6 Kế hoạch thực hiện 6](#_Toc184914566)

[1.6.1 Phân tích yêu cầu và chuẩn bị tài liệu 6](#_Toc184914567)

[1.6.2 Chuẩn bị môi trường kiểm thử 6](#_Toc184914568)

[1.6.3 Kết quả kiểm thử 6](#_Toc184914569)

[1.7 Thông tin nhóm 6](#_Toc184914570)

[II. Chiến lược kiểm thử 7](#_Toc184914571)

[2.1 Yêu cầu và chức năng 7](#_Toc184914572)

[2.2 Ưu tiên thiết kế Test Case 7](#_Toc184914573)

[2.3 Thực hiện kiểm thử tự động 8](#_Toc184914574)

[2.3.1 Kỹ thuật sinh test case 8](#_Toc184914575)

[2.3.1.1 Đăng nhập 8](#_Toc184914576)

[a. Phân vùng tương đương 8](#_Toc184914577)

[b. Giá trị biên 8](#_Toc184914578)

[c. Bảng quyết định 9](#_Toc184914579)

[d. Test case 9](#_Toc184914580)

[2.3.1.2 Quản lý tài khoản 10](#_Toc184914581)

[a. Phân vùng tương đương 10](#_Toc184914582)

[b. Giá trị biên 11](#_Toc184914583)

[c. Test case 12](#_Toc184914584)

[2.3.1.3 Tìm kiếm và lọc sản phẩm 13](#_Toc184914585)

[a. Phân vùng tương đương 13](#_Toc184914586)

[b. Test case 14](#_Toc184914587)

[2.3.1.4 Thêm sản phẩm vào giỏ hàng 14](#_Toc184914588)

[a. Phân vùng tương đương 14](#_Toc184914589)

[b. Giá trị biên 15](#_Toc184914590)

[c. Test Case 15](#_Toc184914591)

[III. Thực hiện kiểm thử 16](#_Toc184914592)

[3.1 Quy trình thực hiện kiểm thử tự động 16](#_Toc184914593)

[3.1.1 Môi trường kiểm thử 16](#_Toc184914594)

[3.1.2 Viết Code 16](#_Toc184914595)

[3.1.2.1 Đăng nhập 18](#_Toc184914596)

[3.1.2.2 Quản lý tài khoản 19](#_Toc184914597)

[3.1.2.3 Tìm kiếm và lọc sản phẩm 20](#_Toc184914598)

[3.1.2.4 Thêm sản phẩm vào giỏ hàng 21](#_Toc184914599)

[3.2 Chạy testcase tự động 21](#_Toc184914600)

[IV. Kết quả đạt được 22](#_Toc184914601)

[4.1 Kết quả kiểm tra của chức năng đăng nhập 22](#_Toc184914602)

[4.2 Kết quả kiểm tra của chức năng quản lý tài khoản 23](#_Toc184914603)

[4.3 Kết quả kiểm tra của chức năng tìm kiếm và lọc sản phẩm 23](#_Toc184914604)

[4.4 Kết quả kiểm tra của chức năng thêm sản phẩm vào giỏ hàng 24](#_Toc184914605)

[V. Kết quả kiểm thử 25](#_Toc184914606)

[5.1. Kết quả kiểm thử 25](#_Toc184914607)

[5.2. Danh sách lỗi tìm được 25](#_Toc184914608)

[VI. Nhận xét, đánh giá và đề xuất 26](#_Toc184914609)

[6.1. Nhận xét: 26](#_Toc184914610)

[6.2. Đánh giá: 26](#_Toc184914611)

[6.3. Đề xuất: 26](#_Toc184914612)

# I. ****Giới thiệu****

## ****1.1 Mô tả ứng dụng kiểm thử****

**Website Tiki một trong những nền tảng thương mại điện tử tại Việt Nam Tiki cung cấp các dịch vụ mua sắm trực tuyến, bao gồm việc tìm kiếm sản phẩm, thêm sản phẩm vào giỏ hàng, thanh toán và các chức năng quản lý tài khoản người dùng**

* + **Website Tiki được xây dựng sẵn, do đó quá trình kiểm thử sẽ tập trung vào hành vi của người dùng**

## ****1.2 Mục tiêu đồ án****

Mục tiêu của đồ án là thực hiện kiểm thử tự động cho website Tiki Kiểm thử sẽ sử dụng Selenium và C# để kiểm thử tự động các chức năng

## ****1.3 Các chức năng/module thực hiện kiểm thử****

* + - **Đăng nhập** kiểm thử chức năng đăng nhập của người dùng
    - **Quản lý tài khoản** cập nhập thông tin cá nhân xem, chỉnh sữa danh sách địa chỉ giao hàng
    - **Tìm kiếm và lọc sản phẩm** kiểm thử tính năng tìm kiếm và lọc sản phẩm của website tiki
    - **Thêm sản phẩm vào giỏ hàng và chỉnh sửa số lượng** kiểm thử chức năng thêm sản phẩm vào giỏ hàng và chỉnh sửa số lượng sản phẩm khi thêm sản phẩm và chỉnh sữa số lượng trong giỏ hàng

## ****1.4 Môi trường, công cụ và các giả định****

* **Môi trường kiểm thử:** Sử dụng trình duyệt Chrome và ChromeDriver.
* **Công cụ kiểm thử:** Selenium WebDriver để thực hiện kiểm thử tự động giao diện người dùng, lập trình bằng ngôn ngữ lập trình C#.
* **Giả định:** Môi trường ổn định, mạng không gián đoạn, hệ thống website không gặp lỗi nghiêm trọng

## ****1.5 Nhân sự thực hiện****

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Họ và Tên** | **MSSV** | **Trách nhiệm** |
| 1 | **Phạm Dương Công Phúc** | **211A010356** | Kiểm thử tính năng thêm sản phẩm và sữa số lượng sản phẩm trong giỏ hàng |
| 2 | **Đặng Vương Thái Đăng** | **211A010010** | quản lý tài khoản Cập nhật thông tin cá nhân (họ tên, số điện thoại, email, địa chỉ), xem chỉnh sửa địa chỉ giao hàng |
| 3 | **Nguyễn Tuấn Kiệt** | **211A290020** | Kiểm thử tính năng tìm kiếm và lọc sản phẩm xác định kết quả tìm kiếm đúng với thông tin tìm kiếm |
| 4 | **Lê Nguyễn Hoàn Huy** | **211A290034** | Kiểm thử tính năng đăng nhập |

## ****1.6 Kế hoạch thực hiện****

### ****1.6.1 Phân tích yêu cầu và chuẩn bị tài liệu****

* Tìm hiểu các chức năng cần kiểm thử
  + Thêm sản phẩm vào giỏ hàng và chỉnh sửa số lượng
  + Tìm kiếm và lọc sản phẩm
  + Đăng nhập
  + Quản lý tài khoản (chỉnh sửa thông tin cá nhân và địa chỉ giao hàng)
* Xác định các trường hợp kiểm thử (test case)
* Dữ liệu bao gồm tài khoản người dùng, sản phẩm cần kiểm thử

### ****1.6.2 Chuẩn bị môi trường kiểm thử****

* + Cài đặt Selenium WebDriver
  + Cài đặt Visual Studio, ngôn ngữ C#
  + Cài đặt ChromeDriver tương thích với trình duyệt Chrome

### ****1.6.3 Kết quả kiểm thử****

* + **Số lượng test case đã thực hiện**
  + **Số lượng test case Pass/Fail**
  + **Các lỗi phát hiện trong quá trình kiểm thử**

## ****1.7 Thông tin nhóm****

Thông tin liên lạc của nhóm

**Lê Nguyễn Hoàn Huy**

* **SDT:** 0901647655
* **EMAIL:** hoanhuy12@gmail.com

# II. ****Chiến lược kiểm thử****

## 2.1 Yêu cầu và chức năng

Xác định yêu cầu Dựa trên các chức năng đã phân công, nhóm tiến hành phân tích và liệt kê các yêu cầu kiểm thử cho từng chức năng

* Đăng nhập
* Quản lý tài khoản
* Tìm kiếm và lọc sản phẩm
* Thêm sản phẩm vào giỏ hàng và chỉnh sửa số lượng

Xác định giao diện người dùng sử dụng công cụ DevTools của Chrome để xác định các phần tử cần tương tác (XPath, CSS Selector)

Xác định danh sách testcase cần thực hiện cho từng chức năng để không bỏ sót test case nào

## 2.2 Ưu tiên thiết kế Test Case

**Thứ tự ưu tiên**

* + **Đăng nhập:** Chức năng cơ bản để thực hiện các tính năng khác của trang web nếu người dùng không đăng nhập họ không thể thực hiện các thao tác như quản lý thông tin tài khoản, thêm sản phẩm, mua hàng…
  + **Quản lý tài khoản:** Dùng cập nhật thông tin cá nhân và lưu trữ địa chỉ giao hàng là một chức năng quản trộng để thực hiện thanh toán và giao hàng
  + **Tìm kiếm và lọc sản phẩm:** Tính năng giúp người dùng tìm thấy sản phẩm mong muốn dễ dàng việc kiểm thử chức năng mông muốn kết quả tìm kiếm trả về đúng với mông muốn của từ khóa tìm kiếm
  + **Thêm sản phẩm vào giỏ hàng và chỉnh sửa số lượng sản phẩm:** Chức năng quan trong của 1 website về mua sấm online giúp người dùng lưu trữ và xác mình sản phẩm đứng với nhu cầu của bản thân kiểm thử đảm bảo hệ thống xử lý đúng số lượng và thông tin sản phẩm trong giỏ hàng

## 2.3 Thực hiện kiểm thử tự động

### 2.3.1 Kỹ thuật sinh test case

#### ****2.3.1.1 Đăng nhập****

### ****a. Phân vùng tương đương****

Đầu vào số điện thoại, mật khẩu

Các nhóm hợp lệ và không hợp lệ

* **Số điện thoại**
  + **Hợp lệ**
    - 0901647555
  + **Không hợp lệ**
    - taikhoan312 (có chứa ký tự)
    - Mã mạng không tồn tài (khác 03x, 05x, 07x, 08x, 09x)
    - Trống ("")
    - Quá ngắn (12345)
    - Quá dài (chứa hơn 11 số)
* **Mật khẩu**
  + **Hợp lệ**
    - matkhau123.
  + **Không hợp lệ**
    - Trống ("")
    - Quá ngắn (1h34)
    - Chỉ có số (12345678)
    - Chỉ có chữ (gawgaghn)

### ****b. Giá trị biên****

* + **Số điện thoại** tối thiểu 10 số tối đa 10 số
    - **Hợp lệ:** 0901234567
    - **Không hợp lệ:** 
      * 090123456 (9 số)
      * 09876543211 (11 số)
  + **Mật khẩu** tối thiểu 8 ký tự, tối đa 32 ký tự bao gồm số và chữ
    - **Hợp lệ:** matk1234
    - **Không hợp lệ:** 
      * Huy123 (<8 ký tự)
      * 123ghagw123tg1232g2g2g23g23g2g238 (>32 ký tự)

### ****c. Bảng quyết định****

|  |  |  |
| --- | --- | --- |
| **Số điện thoại** | **Mật khẩu** | **Output** |
| F | F | F |
| T | F | F |
| T | T | T |

### ****d. Test case****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Input** | | **Output** | **Ghi chú** |
|  | **Số điện thoại** | **Mật khẩu** |  |  |
| 1 | “ “ | … | F (Số điện thoại không được để trống) |  |
| 2 | 09876543211 | … | F | Không thể nhập vào số thứ 11 |
| 3 | 098765432 (9 số) | … | F (Số điện thoại không đúng định dạng) | Số điện thoại quá ngắn 9 số |
| 4 | %@!^ | … | F | Không thể nhập vào ký tự đặc biệt |
| 5 | abcdefghij | … | F | Không thể nhập vào ký tự |
| 6 | 1111647655 | … | F (Số điện thoại không đúng định dạng) | SDT Mã mạng không tồn tại ở VN |
| 7 | 0901647652 | … | F | Số điện thoại không có trong hệ thống tiền hành đăng ký |
| 8 | 0901647655 | … | T | Đúng SDT chuyển qua nhập mật khẩu |
| 9 | 0901647655 | “ “ | F (Mật khẩu không được để trống) |  |
| 10 | 0901647655 | Gaw212315 | F (Đăng nhập thất bại mật khẩu không đúng) |  |
| 11 | 0901647655 | Huy12345 | T | Hiện lại trang chủ tiki và có thông tin tài khoản |

#### ****2.3.1.2 Quản lý tài khoản****

### ****a. Phân vùng tương đương****

* **Cập nhập tên**
  + Đầu vào: Trường fullName.
  + Các nhóm hợp lệ và không hợp lệ:
    - Hợp lệ:
      * Tên: Thai Dang.
      * Tên chứa dấu: Thái Đăng.
    - Không hợp lệ:
      * Tên rỗng: "".
      * Tên quá dài (Tối Đa 64 kí tự) : "Thai Dang AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA".
      * Tên chứa ký tự đặc biệt không cho phép: "Thai Dang @#$%^&\*()".
      * Tên chứa Số không cho phép: "Thai Dang 1101"
* **Cập nhập ngày sinh**
  + Đầu vào: Trường ngày, tháng, năm sinh.
  + Các nhóm hợp lệ và không hợp lệ:
    - Hợp lệ:
      * Ngày: 11.
      * Tháng: 1.
      * Năm: 2003.
    - Không hợp lệ:
      * Ngày vượt biên: 32, 0.
      * Tháng vượt biên: 13, 0.
      * Năm vượt biên: 1900, 2024.
* **Cập nhật quốc tịch**
  + Đầu vào: Quốc tịch trong dropdown.
  + Các nhóm hợp lệ và không hợp lệ:
    - Hợp lệ:
      * Vietnamese.
      * American.
    - Không hợp lệ:
      * Quốc tịch không tồn tại: "Alien".
      * Rỗng: ""
* **Cập nhật giới tính**
  + Đầu vào: Trường radio chọn giới tính.
  + Các nhóm hợp lệ và không hợp lệ:
    - Hợp lệ:
      * Nam.
      * Nữ.
      * Khác.
    - Không hợp lệ:
      * Rỗng (không chọn).
* **Cập nhật Nickname**
  + Đầu vào: Trường userName ….
  + Các nhóm hợp lệ và không hợp lệ:
    - Hợp lệ:
      * Tên: “ dalziel “
      * Tên có kí tự : “Dang\_dep\_trai”
    - Không hợp lệ:
      * Tên rỗng: "".
      * Tên quá dài (Tối Đa 64 kí tự) : "Thai Dang AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA".
      * Tên chứa ký tự đặc biệt không cho phép: "Thai Dang @#$%^&\*()".
      * Tên chứa Số không cho phép: "Thai Dang 1101".
      * Tên chứa dấu: “ Thái Đăng ”

### ****b. Giá trị biên****

* **Cập nhập tên**
  + Giới hạn độ dài của trường tên: Tối thiểu 1 ký tự, tối đa 50 ký tự.
    - Hợp lệ:
      * "D" (1 ký tự).
      * "Thai Dang" (8 ký tự).
      * "Dang Vuong Thai Dang" (50 ký tự).
    - Không hợp lệ:
      * "" (0 ký tự).
      * "Thai Dang AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA" (>50 ký tự)
* **Cập nhập ngày sinh**
  + Ngày sinh giới hạn hợp lệ từ 01/01/1900 đến ngày hiện tại.
    - Hợp lệ:
      * 01/01/1900.
      * 31/12/2023.
    - Không hợp lệ:
      * 31/12/1899.
      * 01/01/2024.
* **Cập nhật Nickname**
  + Giới hạn độ dài của trường tên: Tối thiểu 1 ký tự, tối đa 50 ký tự.
    - Hợp lệ:
      * "D" (1 ký tự).
      * "Thai Dang" (8 ký tự).
      * "Dang Vuong Thai Dang" (50 ký tự).
    - Không hợp lệ:
      * "" (0 ký tự).
      * "Thai Dang AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA" (>50 ký tự).
      * "Thai Dang @#$%^&\*()".(ký tự đặt biệt )
      * “Thai Dang 11“ ( tên chứa số )
      * “ Thái Đăng ” (tên chứa dấu )

### ****c. Test case****

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Điều Kiện** | **Tên** | **NickName** | **Ngày Sinh** | **Giới tính** | **Quốc Tịch** | **Kết quả** |
| Hợp lệ | Thai Dang | dangcute | 11/01/2003 | Nam | Việt Nam | Cập nhật thành công |
| Tên rỗng | “ ” | dangcute | 11/01/2003 | Nam | Việt Nam | Báo lỗi” Tên không hợp lệ…” |
| Nickname đã tồn tại | Thai Dang | Dal | 11/01/2003 | Nam | Việt Nam | Báo lỗi “Nickname đã tồn tại trong hệ thống hoặc đã có người sử dụng” |

#### ****2.3.1.3 Tìm kiếm và lọc sản phẩm****

### ****a. Phân vùng tương đương****

* **Tìm kiếm theo keyword**
  + **Từ khóa hợp lệ (valid input)**
    - **Từ khóa hợp lệ và có kết quả**: Ví dụ, tìm kiếm với từ khóa "Laptop" hoặc "iPhone".
    - **Từ khóa hợp lệ nhưng không có kết quả**: Ví dụ, tìm kiếm với từ khóa "abcdxyz" mà không có sản phẩm nào phù hợp.
  + **Từ khóa không hợp lệ (invalid input)**
    - **Chứa ký tự đặc biệt hoặc khoảng trắng**: Ví dụ, tìm kiếm với ký tự không hợp lệ như "@#$%"
    - **Chuỗi quá ngắn hoặc không có ký tự**: Ví dụ, tìm kiếm với từ khóa chỉ có một ký tự hoặc chuỗi trống.
* **Tìm kiếm theo theo bộ lộc**
  + **Lọc giá hợp lệ:**
    - **Trường hợp 1**: Lọc giá hợp lệ có sản phẩm phù hợp (ví dụ: "Dưới 1 triệu" hoặc "1 triệu - 5 triệu")
      * **Phân vùng**: Các khoảng giá hợp lệ
      * **Kiểm thử**: Xác minh rằng các sản phẩm nằm trong khoảng giá này được hiển thị chính xác
    - **Trường hợp 2**: Lọc giá hợp lệ nhưng không có sản phẩm phù hợp (ví dụ: "50 triệu - 100 triệu").
      * **Phân vùng**: Các khoảng giá hợp lệ nhưng không có sản phẩm
      * **Kiểm thử**: Kiểm tra rằng hệ thống hiển thị thông báo "Không tìm thấy sản phẩm phù hợp".
  + **Lọc giá không hợp lệ:**
    - **Trường hợp 1**: Giá âm (ví dụ: "-1 triệu") .
      * **Phân vùng**: Các khoảng giá không hợp lệ (giá âm).
      * **Kiểm thử**: Xác minh rằng hệ thống sẽ không cho phép người dùng chọn hoặc sẽ thông báo lỗi
    - **Trường hợp 2**: Giá sai định dạng (ví dụ: "abc" hoặc "100 triệu đến 50 triệu ")
      * **Phân vùng**: Đầu vào không hợp lệ (dữ liệu không phải là số hoặc sai cú pháp).
      * **Kiểm thử**: Kiểm tra xem hệ thống hiển thị thông báo lỗi yêu cầu người dùng nhập lại đúng định dạng.
  + **Lọc giá đặc biệt:**
    - **Trường hợp 1**: Chọn khoảng giá lớn hơn giá sản phẩm cao nhất (ví dụ: "Trên 200 triệu").
      * **Phân vùng**: Các khoảng giá vượt quá mức giá của sản phẩm.
      * **Kiểm thử**: Kiểm tra xem hệ thống hiển thị thông báo "Không tìm thấy sản phẩm phù hợp".
    - **Trường hợp 2**: Nhập khoảng giá tùy chỉnh đúng định dạng (ví dụ: "5 triệu - 10 triệu") .
      * **Phân vùng**: Các khoảng giá hợp lệ do người dùng tùy chỉnh.
      * **Kiểm thử**: Kiểm tra xem danh sách sản phẩm trong khoảng giá này có được hiển thị chính xác không
    - **Trường hợp 3**: Nhập khoảng giá tùy chỉnh sai định dạng hoặc để trống (ví dụ: "10 triệu - ", "abc - xyz").
      * **Phân vùng**: Các khoảng giá sai định dạng.
      * **Kiểm thử**: Kiểm tra xem hệ thống sẽ hiển thị thông báo lỗi yêu cầu người dùng nhập lại khoảng giá hợp lệ.

### ****b. Test case****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Loại từ khoá** | **Trường hợp** | **Dữ liệu** | **Kết quả mong đợi** |
| 1 | Từ khoá hợp lệ | Từ khoá hợp lệ và có kết quả | “iphone” | Hiển thị danh sách phù hợp với từ khoá tìm kiếm |
| 2 | Từ khoá hợp lệ | Từ khoá hợp lệ nhưng có kết quả tìm kiếm | “abzfjl”  “"@#$%".”  “a” | Hiển thị danh sách nhưng không có sản phẩm khớp với keyword |
| 3 | Giá hợp lệ | Giá trong khoảng có sẵn và sản phẩm phù hợp | "Dưới 22.000.000” | Hiển thị danh sách sản phẩm nằm trong khoảng giá đã chọn. |
| 4 | Giá hợp lệ | Giá trong khoảng người dùng nhập vào | "10 triệu - 20 triệu" | Hiển thị danh sách sản phẩm nằm trong khoảng người dùng nhập vào |

#### ****2.3.1.4 Thêm sản phẩm vào giỏ hàng****

### ****a. Phân vùng tương đương****

* Thêm sản phẩm vào giỏ hàng
  + Hợp lệ: Sản phẩm tồn tại, còn hàng
  + Không hợp lệ: Sản phẩm hết hàng, không tồn tại, hoặc dữ liệu thiếu
* Cập nhật số lượng sản phẩm
  + Hợp lệ: Số lượng trong giới hạn tồn kho
  + Không hợp lệ: Số lượng vượt quá tồn kho, số âm, hoặc không phải số
* Xóa sản phẩm khỏi giỏ hàng
  + Hợp lệ: Sản phẩm có trong giỏ hàng
  + Không hợp lệ: Sản phẩm không tồn tại trong giỏ hàng hoặc giỏ hàng trống

### ****b. Giá trị biên****

* Thêm sản phẩm vào giỏ hàng
  + Sản phẩm tồn kho = 1 (giới hạn tối thiểu).
  + Sản phẩm tồn kho = 100 (gới hạn tối đa).
* Cập nhật số lượng sản phẩm
  + Giới hạn dưới: số lượng = 0 (xóa sản phẩm)
  + Giới hạn trên: số lượng = tồn kho tối đa
* Xóa sản phẩm khỏi giỏ hàng
  + Giỏ hàng có 1 sản phẩm (trạng thái tối thiểu)
  + Giỏ hàng đầy (trạng thái tối đa)

### ****c. Test Case****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Hành động** | **Trường hợp** | **Số lượng thêm vào** | **Kết quả mong đợi** |
| 1 | Thêm sản phẩm hợp lệ | Sản phẩm hợp lệ, số lượng trong tồn kho | 1 | Số lượng sản phẩm trong giỏ hàng đúng với số lượng đã thêm |
| 2 | Thêm sản phẩm với số lượng bằng 0 | Số lượng bằng 0 | 0 | Không thể nhập 0 hoặc giảm không trong số lượng |
| 3 | Thêm sản phẩm với số lượng âm | Số lượng âm | -1 | Không thể nhập -1 hoặc giảm không trong số lượng |
| 4 | Thêm sản phẩm vượt quá tồn kho | Số lượng lớn hơn tồn kho | 101 (chỉ có 100 tồn kho) | Không thể nhập số lượng vướt quá hàng tồn kho |
| 5 | Thêm sản phẩm nhiều lần để tăng số lượng | Thêm 1 sản phẩm nhiều lần | Thêm 1 nhiều lần | Số lượng sản phẩm trong giỏ hàng tăng đúng với số lượng đã thêm |

# III. ****Thực hiện kiểm thử****

## 3.1 Quy trình thực hiện kiểm thử tự động

* Chuẩn bị môi trường công cụ Selenium WebDriver, ChromeDriver, Visual Studio
* Tạo script kiểm thử
  + Khởi động trình duyệt và điều hướng đến website Tiki
  + Tạo các file kiểm thử theo từng chức năng (thêm sản phẩm, đăng nhập, tìm kiếm)
* Chạy script và ghi nhận kết quả

### 3.1.1 Môi trường kiểm thử

* **Các công cụ:**
  + Selenium WebDriver để thực hiện kiểm thử tự động
  + Visual Studio với C# dùng để
  + ChromeDriver đảm bảo tương thích với trình duyệt Chrome
* **Tài khoản kiểm thử:** Sữ dụng tài khoản có sản hoặc là tạo 1 tài khoản để kiểm thử

### 3.1.2 Viết Code

Sử dụng ngôn ngữ lập trình C# để viết các test script kiểm thử tự động

Test script được chia nhỏ theo từng chức năng, bao gồm

* **Khởi tạo trình duyệt (OneTimeSetup)**

Phương thức **OneTimeSetup** là một phương thức khởi tạo được thực hiện một lần duy nhất trước khi tất cả các test case trong lớp test được chạy để tránh tải sữ dụng nhiều lẫn cho mỗi test case để tiết kiệm thời gian và bộ nhớ

Trình duyệt Chrome được khởi động thông qua ChromeDriver với các tuỳ chọn mặc định

Dòng lệnh

**driver.Manage().Timeouts().ImplicitWait = TimeSpan.FromSeconds(20);**

để chắc chắn WebDriver sẽ đợi 5 giây trước khi ném lỗi nếu không tìm thấy …

**driver.Manage().Window.Maximize()**

dùng để mỡ trình duyệt với toàn màng hình

* **Khởi tạo môi trường trước khi chạy test case (SetUp)**

Điều hướng trình duyệt đến trang web Tiki bằng

driver.Navigate().GoToUrl("https://tiki.vn/");

Thread.Sleep(2000);

lệnh này sẽ đợi trang web tải trong 2s để tiếp tục chức năng khác

try

{

driver.FindElement(By.CssSelector("img[alt='close-icon']")).Click();

Thread.Sleep(1500);

}

catch (NoSuchElementException) {

//nếu không có quảng cáo thì bỏ qua

}

Sữ dụng try catch và dòng lệnh bên trong dùng để tìm và đống quảng cáo nếu không tìm thầy thì ko thực hiện

* + - * **Tiếp theo đăng nhập dùng cho chức năng thêm sản phẩm và sữa thông tin tài khoản**

Lệnh này để tìm kiếm đượng dẫn mỡ form đăng nhập

// Mở form đăng nhập

driver.FindElement(By.CssSelector("div[data-view id='header\_header\_account\_container']")).Click();

* + - * **Từ phần này danh cho chỉnh sữa thông tin và giỏ hàng**

Tìm trường nhập số điện thoại số điện thoại và nhập

driver.FindElement(By.Name("tel")).SendKeys("…");

Tìm và bấm nút tiếp tục

driver.FindElement(By.XPath("//button[contains(text(), 'Tiếp Tục')]")).Click();

Tìm trường nhập mật khẩu nhập vật khẩu

driver.FindElement(By.XPath("//input[@placeholder='Mật khẩu']")).SendKeys("Mật khẩu");

Tìm và bấm nút đăng nhập

driver.FindElement(By.XPath("//button[contains(text(), 'Đăng Nhập')]")).Click();

Sau khi đăng nhập thì chương trình sẽ đợi vài giây để cho người test tự giải capcha

#### 3.1.2.1 Đăng nhập

* + Kiểm tra trường hợp đăng nhập thành công và thất bại với các dữ liệu đầu vào khác nhau
  + Gồm 2 đầu vào là số điện thoại và mật khẩu

Hàm nhập số điện thoại

private void EnterPhoneNumber(string phoneNumber)

{

var phone = driver.FindElement(By.Name("tel"));

phone.Clear();

phone.SendKeys(phoneNumber);

}

Hàm nhập mật khẩu

private void EnterPass(string passWord)

{

var pass =

driver.FindElement(By.XPath("//input[@placeholder='Mật khẩu']"));

pass.Clear();

pass.SendKeys(passWord);

}

* 2 hàm này dùng để thực hiện tìm đăng nhập số điện thoại và mật khẩu nhiều lần

Sau đó sẽ lần lược tạo testcase theo danh sách test case ví dụ với trường hợp test số điện thoại

var errorMessage = driver.FindElements(By.CssSelector("span.error-mess"));

Nếu có lỗi xuất hiện thì biến errorMessage sẽ tìm đến đường dẫn chứa thông báo lỗi sau đó thì kiểm tra lỗi dưới đây là 1 câu lệnh ví dụ cho trường hợp để số điện thội trống

if (errorMessage.Count > 0 &&

errorMessage[0].Text == "Số điện thoại không được để trống")

{

Assert.Pass("Kết quả: Số điện thoại trống | Thông báo đúng");

}

else

{

Assert.Fail("Kết quả: Thông báo đã có gì đó được nhập vào trong SDT");

}

#### 3.1.2**.2 Quản lý tài khoản**

// In thông tin trước khi reload

Console.WriteLine($"Trước khi reload - FullName: {fullName}, UserName: {userName}, Day: {dayValue}, Month: {monthValue}, Year: {yearValue}");

Thread.Sleep(5000);

// Reload lại trang

driver.Navigate().Refresh();

// Kiểm tra giá trị sau khi reload

try

{

var nameAfterReload = driver.FindElement(By.CssSelector("input[name='fullName']")).GetAttribute("value");

var userNameAfterReload = driver.FindElement(By.CssSelector("input[name='userName']")).GetAttribute("value");

var dayAfterReload = new SelectElement(driver.FindElement(By.CssSelector("select[name='day']"))).SelectedOption.GetAttribute("value");

var monthAfterReload = new SelectElement(driver.FindElement(By.CssSelector("select[name='month']"))).SelectedOption.GetAttribute("value");

var yearAfterReload = new SelectElement(driver.FindElement(By.CssSelector("select[name='year']"))).SelectedOption.GetAttribute("value");

// In thông tin sau khi reload

Console.WriteLine($"Sau khi reload - FullName: {nameAfterReload}, UserName: {userNameAfterReload}, Day: {dayAfterReload}, Month: {monthAfterReload}, Year: {yearAfterReload}");

// So sánh giá trị

if (nameAfterReload == fullName &&

userNameAfterReload == userName &&

dayAfterReload == dayValue &&

monthAfterReload == monthValue &&

yearAfterReload == yearValue)

{

Console.WriteLine("Dữ liệu đã được lưu thành công!");

}

else

{

Console.WriteLine("Dữ liệu không được lưu chính xác.");

}

}

catch (Exception ex)

{

Console.WriteLine($"Lỗi khi kiểm tra giá trị sau reload: {ex.Message}");

}

}

#### 3.1.2.3 Tìm kiếm và lọc sản phẩm

Ô tìm kiếm sản phẩm được xác định qua

CssSelector("input[data-view-id='main\_search\_form\_input']"));

và từ khóa "MacBook" được nhập vào ô tìm kiếm.

Tiếp theo, nút tìm kiếm được nhấn thông qua

CssSelector("button[data-view-id='main\_search\_form\_button']"))

để gửi yêu cầu tìm kiếm.

**+Xử lý kết quả tìm kiếm** Sau khi nhấn nút tìm kiếm, chương trình chờ thêm 3 giây

Thread.Sleep(3000)

để kết quả tìm kiếm có thời gian tải.

Danh sách sản phẩm được luu bằng lệnh

CssSelector("span.style\_\_StyledItem-sc-139nb47-0.fxtnPx")

Vào biến **productList**

Sau đó kiếm tra kết quả tìm kiếm sản phẩm có khớp với keyword đã tìm kiếm không ví dụ TestCase 1

foreach (var product in productList)

{

// Lấy tên sản phẩm

var productName = product.FindElement(By.CssSelector("h3.style\_\_NameStyled-sc- 139nb47-8.ibOlar")).Text;

if (!productName.ToLower().Trim().Contains("macbook"))

{

Assert.Fail($"Sản phẩm không liên quan: {productName}");

}

else

{

Assert.Pass("Sản phẩm tìm thấy khớp với từ khóa");

}

}

#### 3.1.2.4 Thêm sản phẩm vào giỏ hàng

Sau khi đăng nhập thành công mỡ trang chứa sản phẩm

driver.Navigate().GoToUrl("https://tiki.vn/nuoc-giat-earth-choice-pure-clean-1l-p276237692.html?spid=276237693");

Sau đó nhấp số lượng vào ô số lượng

var inputElement = driver.FindElement(By.CssSelector("input.input"));

inputElement.Clear();

inputElement.SendKeys("1");

Và nhấn nút thêm sản phẩm

driver.FindElement(By.CssSelector("button[data-view-id='pdp\_add\_to\_cart\_button']")).Click();

Sau đó vào trang giỏ hàng

driver.FindElement(By.XPath("//img[@alt='header\_header\_img\_Cart']")).Click();

Và thức hiện kiểm tra sản phẩm trong giỏ hàng

if (emptyMessageElements.Count > 0 && emptyMessageElements[0].Text.Contains("Giỏ hàng trống"))

{

Assert.Fail("Thêm sản phẩm vào giỏ hàng thất bại. Thông báo: 'Giỏ hàng trống");

}

else if (productNameElements.Count > 0)

{

Assert.Pass("Sản phẩm đã được thêm vào giỏ hàng đúng với kết quả mông đợi");

}

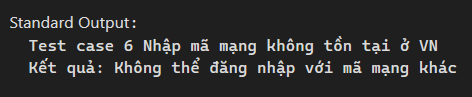
else

{

Assert.Fail("Sản phẩm đã thêm vào giỏ hàng không đúng với kết quả mông đợi");

}

## 3.2 Chạy testcase tự động

* Kết nối ChromeDriver để thực hiện các code trên trình duyệt Chrome
* Chạy các test code bằng **Test Explorer** trong Visual Studio
* Ghi nhận kết quả từ Standrad Output: khi test
  + Ví dụ:

# IV. ****Kết quả đạt được****

* **Kết quả kiểm thử**:
  + Tổng số test case: 11
  + Pass: 11
  + Fail: 0

## 4.1 Kết quả kiểm tra của chức năng đăng nhập

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Trường hợp kiểm thử** | **Ưu tiên** | **Bước thực hiện** | **Kết quả mong đợi** | **Kết quả thực tế** | **Kết quả kiểm thử** |
| 1 | Đăng nhập | Trường hợp để trống số điện thoại | 1 | Không điền số điện thoại | Thông báo số điện thoại không được để trống | Thông báo lỗi hiển thị đúng | **Pass** |
| 2 | Đăng nhập | Trường hợp nhập số điện thoại quá 11 số | 1 | Nhập số điện thoại 11 chữ số | Thông báo không thể nhập 11 số điện thoại | Chỉ có 10 số được nhập | **Pass** |
| 3 | Đăng nhập | Trường hợp nhập số điện thoại quá ngắn | 1 | Nhập số điện thoại dưới 10 chữ số | Thông báo số điện thoại không đúng định dạng | Thông báo lỗi hiển thị đúng | **Pass** |
| 4 | Đăng nhập | Trường hợp nhập số điện thoại chứa ký tự đặc biệt | 1 | Nhập số điện thoại có chứa ký tự đặc | Không thể nhập vào | Không có gì được nhập vào | **Pass** |
| 5 | Đăng nhập | Trường hợp nhập số điện thoại chứa chữ cái | 1 | Nhập số điện thoại có chứa chữ cái | Không thể nhập vào | Không có gì được nhập vào | **Pass** |
| 6 | Đăng nhập | Trường hợp nhập có định dạng mạng không có ở việt nam | 1 | Nhập số điện thoại không có ở việt nam | Thông báo số điện thoại không đúng định dạng | Thông báo lỗi hiển thị đúng | **Pass** |
| 7 | Đăng nhập | Trường hợp nhập số điện thoại không tồn tại trong hệ thống | 1 | Nhập số điện thoại không tồn | Chuyển sang Form đang ký | Kết quả hiện thị đúng | **Pass** |
| 8 | Đăng nhập | Nhập đúng số điện thoại | 1 | Nhập số điện thoại đã được đăng ký | Chuyển sang Form nhập mật khẩu | Kết quả hiện thị đúng | **Pass** |
| 9 | Đăng nhập | Trường hợp để trống mật khẩu | 2 | Để trống mật khẩu | Thông báo mật khẩu không được để trống | Thông báo lỗi hiển thị đúng | **Pass** |
| 10 | Đăng nhập | Trường hợp nhập sai mật khẩu | 2 | Nhập sai mật khẩu | Thông báo sai mật khẩu | Thông báo hông tin đăng nhập không đúng | **Pass** |
| 11 | Đăng nhập | Trường hợp nhập đúng mật khẩu | 2 | Nhập đứng mật khẩu | Đăng nhập thành công | Tắt Form đăng nhập và có thể xem thông tin cá nhân | **Pass** |

## Kết quả kiểm tra của chức năng quản lý tài khoản

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Trường hợp kiểm thử** | **Ưu tiên** | **Bước thực hiện** | **Kết quả mong đợi** | **Kết quả thực tế** | **Kết quả kiểm thử** |
| 1 | Quản lí tài khoản | Để trống name “ “ | **1** | Không điền tên | Báo lỗi” Tên không hợp lệ…” | Thông báo lỗi hiển thị đúng | **Pass** |
| 2 | Quản lí tài khoản | Điền tên + só | **1** | Điền tên : Thai Dang 01 | Thông báo lỗi: Tên không hợp lệ, vui lòng nhập tên không có kí tự và số | Thông báo lỗi hiển thị đúng | **Pass** |
| 3 | Quản lí tài khoản | Nhập nickname | **1** | Nhập nickname : Dal | Báo lỗi “Nickname đã tồn tại trong hệ thống hoặc đã có người sử dụng” | Thông báo lỗi hiển thị đúng | **Pass** |

## 4.3 Kết quả kiểm tra của chức năng tìm kiếm và lọc sản phẩm

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Trường hợp kiểm thử** | **Ưu tiên** | **Bước thực hiện** | **Kết quả mong đợi** | **Kết quả thực tế** | **Kết quả kiểm thử** |
| 1 | Tìm kiếm | Từ khoá hợp lệ và có kết quả | **1** | * Điền từ khóa“MacBook” * Tìm sản phẩm khớp với từ khóa | Không tìm thấy sản phẩm không liên quan đến từ khóa | Trả về danh sách khớp với từ khóa | **Pass** |
| 2 | Tìm kiếm | Từ khoá hợp lệ nhưng có kết quả tìm kiếm | **1** | * Điền từ khóa “abzfjl” hoặc “"@#$%".” hoặc “a” * Và tìm sản phẩm khớp với từ khóa | Không có sản phẩm nào liên quan đến từ khóa | Trã về danh sách không có sản phẩm liên quan đến từ khóa | **Pass** |
| 3 | Tìm kiếm | Giá trong khoảng có sẵn và sản phẩm phù hợp | 1 | * Điền từ khóa iphone và nhấn tìm kiếm * Nhấn vào bộ lộc và nhấn giá dưới 22.000.000 | Hiển thị danh sách sản phẩm nằm trong khoảng giá đã chọn. | Trã về danh sách dưới 22tr | **Pass** |
| 4 | Tìm kiếm | Giá trong khoảng người dùng nhập vào | 1 | * Điền từ khóa iphone và nhấn tìm kiếm * Nhấn vào bộ lộc nhập giá vào ô từ - đến | Hiển thị danh sách sản phẩm nằm trong khoảng người dùng nhập vào | Trã về danh sách trong khoản người dùng nhập vào | **Pass** |

## 4.4 Kết quả kiểm tra của chức năng thêm sản phẩm vào giỏ hàng

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STT** | **Chức năng** | **Trường hợp kiểm thử** | **Ưu tiên** | **Bước thực hiện** | **Kết quả mong đợi** | **Kết quả thực tế** | **Kết quả kiểm thử** |
| 1 | Thêm sản phẩm vào giỏ hàng | Thêm sản phẩm hợp lệ | 1 | * Nhấn nút thêm sản phẩm * Sau đó vào giỏ hàng kiểm tra số lượng | Số lượng sản phẩm trong giỏ hàng đúng với số lượng đã thêm | Đúng với mong đợi | **pass** |
| 2 | Thêm sản phẩm vào giỏ hàng | Thêm sản phẩm với số lượng bằng 0 | **1** | Sữa số lượng thêm sản phẩm vào là 0 | Không thể thêm sản phẩm | Không thể nhập 0 hoặc giảm không trong số lượng | **pass** |
| 3 | Thêm sản phẩm vào giỏ hàng | Thêm sản phẩm với số lượng âm | 1 | Sữa số lượng thêm sản phẩm vào là âm | Không thể thêm sản phẩm | Không thể nhập -1 hoặc giảm không trong số lượng | **pass** |
| 4 | Thêm sản phẩm vào giỏ hàng | Thêm sản phẩm vượt quá tồn kho | 1 | Sữa số lượng thành 100 và thêm sản phẩm | Không thể nhập số lượng vướt quá hàng tồn kho | Thông báo số lượng tồn kho khi thêm sản phẩm và không thể thêm | **pass** |
| 5 | Thêm sản phẩm vào giỏ hàng | Thêm sản phẩm nhiều lần để tăng số lượng | 1 | * Nhấn nút thêm sản phẩm * Sau đó vào giỏ hàng kiểm tra số lượng * Lập lại nhiều lần | Số lượng sản phẩm trong giỏ hàng tăng đúng với số lượng đã thêm | Đứng với mong đợi | **pass** |

# V. Kết quả kiểm thử

## 5.1. Kết quả kiểm thử

Tổng số test case đã thực hiện gồm 22 test case

* 11 Test Case Đăng nhập
* 3 Test Case quảng lý tài khoản
* 4 Test Case tìm kiếm và lộc sản phẩm
* 5 Test Case thểm sản phẩm vào giỏ hàng

## 5.2. Danh sách lỗi tìm được

**Không tìm thấy lỗi trong quá trình kiểm thử. Các chức năng hoạt động đúng như kỳ vọng**

# ****VI.**** Nhận xét, đánh giá và đề xuất

## 6.1. Nhận xét:

* Các chức năng chính của website Tiki đã được kiểm thử với các test case được phân chia. Các chức năng như đăng nhập, tìm kiếm, lọc sản phẩm, và quản lý tài khoản đều hoạt động đúng như mong đợi
* Hệ thống không gặp phải lỗi lớn nào trong quá trình kiểm thử, các chức năng hoạt động ổn định và đúng như thiết kế

## 6.2. Đánh giá:

* Các chức năng chính của website Tiki, bao gồm đăng nhập, tìm kiếm sản phẩm, và lọc sản phẩm, đều đáp ứng được yêu cầu và hoạt động chính xác. Điều này cho thấy website Tiki có chất lượng tốt và khả năng sử dụng ổn định.
* Tài nguyên kiểm thử (tài khoản, sản phẩm) đủ để kiểm thử các chức năng chính

## 6.**3. Đề xuất:**

* Mặc dù không phát hiện lỗi trong quá trình kiểm thử, nhưng vẫn nên thường xuyên kiểm tra lại