|  |  |
| --- | --- |
| 1-02 | BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HỒ CHÍ MINH** |

**ĐỒ ÁN MÔN HỌC**

**KIỂM THỬ VÀ ĐẢM BẢO CHẤT LƯỢNG PHẦN MỀM**

**KIỂM THỬ WEBSITE** **SHOP ACCOUNT GAME**

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên ngành : **CÔNG NGHỆ PHẦN MỀM**

Giảng viên hướng dẫn : **Bùi Mạnh Toàn**

Sinh viên thực hiện :

Họ và tên MSSV

**Hồ Đại Minh**  **2180604350**

**Phan Thị Hằng** **2180605032**

**Liêu Đông Huy** **2180606424**

**Phạm Tấn Phát** **2180606267**

**Mã Hoàng Thái 2180606816**

Lớp: **21DTHC1**

TP. Hồ Chí Minh, 2024

# LỜI NÓI ĐẦU

Ngày nay, ứng dụng công nghệ thông tin vào đời sống được xem là một trong những yếu tố mang tính quyết định trong hoạt động của hầu hết tất cả mọi người, nó đóng vai trò hết sức quan trọng, có thể tạo ra những bước đột phá mạnh mẽ.

Cùng với sự phát triển không ngừng về kỹ thuật máy tính và mạng điện tử, công nghệ thông tin cũng được những công nghệ có đẳng cấp và lần lượt chinh phục hết đỉnh cao này đến đỉnh cao khác. Mạng Internet là một trong những sản phẩm có giá trị hết sức lớn lao và ngày càng trở nên một công cụ không thể thiếu, là nền tảng chính cho sự truyền tải, trao đổi thông tin trên toàn cầu. Giờ đây, mọi việc liên quan đến thông tin trở nên thật dễ dàng cho người sử dụng,…

Bằng Internet, chúng ta đã thực hiện được nhiều công việc với tốc độ nhanh hơn và chi phí thấp hơn nhiều so với cách thức truyền thống. Trong hoạt động sản xuất, kinh doanh, giờ đây, thương mại điện tử đã khẳng định được vai trò xúc tiến và thúc đẩy sự phát triển của doanh nghiệp. Đối với một trang website online, việc quảng bá và giới thiệu đến khách hàng các sản phẩm mới đáp ứng được nhu cầu của khách hàng sẽ là cần thiết. Vì vậy, chúng em đã thực hiện đồ án xây dựng “WEBSITE SHOP ACCOUNT GAME”.

MỤC LỤC

[LỜI NÓI ĐẦU 2](#_Toc28901)

[MỤC LỤC 3](#_Toc529)

[**CHƯƠNG 1: GIỚI THIỆU KIỂM THỬ 4**](#_Toc21387)

[1.1 Khái niệm 4](#_Toc5775)

[1.2 Các quy tắc trong kiểm thử 4](#_Toc5081)

[1.3 Các phương pháp trong kiểm thử 5](#_Toc17129)

[**CHƯƠNG 2: THIẾT KẾ TEST CASE 9**](#_Toc4894)

[2.1 Các phương pháp thiết kế test-case (Black-box Testing) 9](#_Toc21443)

[2.2 Kiểm thử hộp trắng (White-box Testing) 11](#_Toc5232)

[2.3 Mô tả đề tài và kế hoạch kiểm thử 11](#_Toc7494)

[*2.3.1 Mô tả đề tài* 11](#_Toc31267)

[*2.3.2 Kế hoạch kiểm thử* 12](#_Toc7386)

[**CHƯƠNG 3: TRIỂN KHAI KIỂM THỬ 13**](#_Toc27264)

[3.1 Mô tả các chức năng 13](#_Toc31519)

[3.2 Kiểm thử hộp đen 19](#_Toc10785)

[*3.2.1 Phương pháp phân vùng tương đương*  **Error! Bookmark not defined.**](#_Toc7757)

[*3.2.2 Kiểm thử phân tích giá trị biên*  20](#_Toc334)

[*3.2.3 Kiểm thử bảng quyết định*  23](#_Toc14059)

[*3.2.4 Kiểm thử dịch chuyển trạng thái*  25](#_Toc25256)

[3.1 Kiểm thử hộp trắng 27](#_Toc107)

[*3.1.1 Phương pháp phủ câu lệnh*  27](#_Toc28707)

[*3.1.2 Phương pháp phủ nhánh*  28](#_Toc549)

[*3.1.3 Phương pháp phủ đường dẫn*  29](#_Toc13323)

[*3.1.4 Phương pháp phủ điều kiện*  31](#_Toc1094)

[**CHƯƠNG 4: KIỂM THỬ TỰ ĐỘNG 33**](#_Toc18624)

[4.1 Công cụ kiểm thử tự động Selenium 33](#_Toc26374)

[4.2 TESTCASE **Error! Bookmark not defined.**](#_Toc10671)

[4.3 BUGSUMARY **Error! Bookmark not defined.**](#_Toc12831)

[**CHƯƠNG 5: KẾT LUẬN 49**](#_Toc18677)

[TÀI LIỆU THAM KHẢO 50](#_Toc20720)

# CHƯƠNG 1: GIỚI THIỆU KIỂM THỬ

## Khái niệm

Kiểm thử phần mềm là quá trình thực thi một hệ thống phần mềm để xác định xem phần mềm có đúng với đặc tả không và thực hiện trong môi trường như mong đợi hay không. Mục đích của kiểm thử phần mềm là để kiểm tra xem phần mềm có đáp ứng với nhu cầu của khách hàng và phù hợp với đặc tả yêu cầu, đảm bảo chất lượng và tính chính xác của ứng dụng

## Các quy tắc trong kiểm thử

Trong kiểm thử phần mềm có 7 nguyên tắc kiểm thử. Tìm hiểu chúng là 1 điều rất quan trọng bởi vì nó giúp tiết kiệm thời gian cũng như công sức truy lùng bug ẩn trong các ứng dụng của bạn. Có một câu hỏi đặt ra như thế này: "Bạn có thật sự hiểu thế nào là nguyên tắc?".

Về cơ bản, nguyên tắc chính là những quy định hoặc là luật mà chúng ta phải tuân theo. Hoặc bạn cũng có thể hiểu nguyên tắc là điều thiết yếu cần phải theo đuổi để phát triển hệ thống tốt nhất. Việc bỏ qua bất kỳ nguyên tắc nào cũng có thể làm giảm hiệu quả của việc thực hiện hệ thống. Vậy chúng ta sẽ nói về 7 nguyên tắc trong kiểm thử phần mềm:

* Kiểm thử đưa ra lỗi: Kiểm thử có thể cho thấy rằng phần mềm đang có lỗi, nhưng không thể chứng minh rằng phần mềm không có lỗi. Kiểm thử được thực hiện bằng những kĩ thuật khác nhau. Kiểm thử làm giảm xác suất lỗi chưa tìm thấy vẫn còn trong phần mềm, ngay cả khi đã kiểm thử nghiêm ngặt phần mềm vẫn có thể còn lỗi. Vì vậy chúng ta phải tìm được càng nhiều lỗi càng tốt.
* Kiểm thử cạn kiệt là không thể: Nguyên tắc này nói rằng kiểm tra mọi thứ trong phần mềm một cách trọn vẹn là 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: Thông thường, lỗi tập trung vào những module, thành phần chức năng chính của hệ thống. Nếu 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.
* Sự tập trung của lỗi: Thông thường, lỗi tập trung vào những module, thành phần chức năng chính của hệ thống. Nếu 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 lí thuốc trừ sâu: Nếu bạn sử dụng cùng một tập hợp các trường hợp kiểm thử liên tục, sau đó một thời gian các trường hợp kiểm thử không tìm thấy lỗi nào mới. Hiệu quả của các trường hợp kiểm thử bắt đầu giảm xuống sau một số lần thực hiện, vì vậy luôn luôn chúng ta phải luôn xem xét và sửa đổi các trường hợp kiểm thử trên một khoảng thời gian thường xuyên.
* Kiểm thử phụ thuộc vào ngữ cảnh: Theo nguyên tắc này thì việc kiểm thử phụ thuộc vào ngữ cảnh và chúng ta phải tiếp cận kiểm thử theo nhiều ngữ cảnh khác nhau Nếu bạn đang kiểm thử ứng dụng web và ứng dụng di động bằng cách sử dụng chiến lược kiểm thử giống nhau, thì đó là sai. Chiến lược để kiểm thử ứng dụng web sẽ khác với kiểm thử ứng dụng cho thiết bị di động của Android.
* Không có lỗi - Sai lầm: 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.

## Các phương pháp trong kiểm thử

* + 1. *Kiểm thử hộp đen*

Kiểm thử hộp đen là một phương pháp kiểm thử mà các tester không cần quan tâm đến các hoạt động bên trong hệ thống chạy ra sao, không cần quan tâm đến các dòng lệnh bên trong hệ thống hệ thống như thế nào. mà chỉ cần tập trung vào các giá trị đầu vào và các giá trị đầu ra của hệ thống có đúng với kết quả mong đợi của các trường hợp kiểm thử không để từ đó đánh giá chất lượng hệ thống. Chính vì cơ chế như vậy nên phương pháp này có các ưu nhược điểm như sau:

Ưu điểm:

* Được độ bao phủ lớn nhất khi thực hiện thiết kế các trường hợp kiểm thử sau này

- Hiệu quả khi số lượng các dòng lệnh của hệ thống lớn.

- Không cần truy cập vào các dòng lệnh

- Phân biệt được rõ quan điểm của người dùng với quan điểm của nhà phát triển

- Không đòi hỏi những kiến thức về ngôn ngữ lập trình ở các tester

Nhược điểm:

- Giới hạn ở độ bao phủ của các trường hợp kiểm thử

- Không hiệu quả bởi các tester bị giới hạn kiến thức về hệ thống

- Độ bao phủ thiếu vì tester không kiểm tra được các đoạn lệnh hoặc tập trung vào các dòng lệnh dễ xảy ra lỗi

- Khó để có thể thiết kế đầy đủ các trường hợp kiểm thử

* + 1. *Kiểm thử hộp trắng*

Kiểm thử hộp trắng là việc nghiên cứu cụ thể chi tiết từng luồng hoạt động cũng như các dòng lệnh bên trong hệ thống. Kiểm thử hộp trắng cũng được gọi dưới các cái tên khác như: Glass testing hay open-box testing. Kiểm thử hộp trắng đòi hỏi tester phải có kiến thức về ngôn ngữ lập trình. Tester sẽ cần phải nghiên cứu vào bên trong hê thống cụ thể là các dòng lệnh để tìm hiểu chúng có chạy đúng hay không. Dưới đây là các ưu nhược điểm của phương pháp này:

Ưu điểm:

* Các tester có kiến thức về ngôn ngữ lập trình sẽ dễ dàng phát hiện ra những lỗi ở các dòng lệnh
* Tối ưu hóa các dòng lệnh của hệ thống
* Các dòng lệnh không cần thiết hoặc các dòng lệnh có khả năng mang tới lỗi tiềm ẩn sẽ bị loại bỏ
* Các tester có kiến thức lập trình sau khi đã thực hiện phương pháp này sẽ dễ dàng đạt được độ bao phủ lớn nhất khi thực hiện thiết kế các trường hợp kiểm thử sau này

Nhược điểm:

* Việc sử dụng các tester có kiến thức về ngôn ngữ lập trình sẽ làm tăng giá thành để phát triển phần mềm
* Đôi lúc không khả thi khi kiểm tra chi tiết từng dòng lệnh để có thể từ đó phát hiện ra các lỗi tiềm ẩn của hệ thống, có rất nhiều các luồng không thể kiểm tra được
* Khó có thể duy trì phương pháp này liên tục, cần phải có tool chuyên dụng như tool về phân tích code hay tool về phát hiện và sửa lỗi
  + 1. *Kiểm thử hộp xám*

Kiểm thử hộp xám là một phương pháp kiểm thử mà đòi hỏi tester phải có một lượng kiến thức nhất định về các luồng hoạt động ở bên trong hệ thống. Khác với kiểm thử hộp đen, phương pháp mà tester chỉ quan tâm duy nhất để việc kiểm thử thông qua giao diện người dùng, kiểm thử hộp xám đòi hỏi tester phải truy cập vào các tài liệu thiết kế hệ thống cũng như hệ thống cơ sở dữ liệu của hệ thống. Do đó mà tester có thể chuẩn bị tốt hơn những dữ liệu cho việc kiểm thử cũng như các trường hợp kiểm thử trong quá trình lên kế hoạch kiểm thử hệ thống.

Ưu điểm:

* Vì là sự kết hợp giữa kiểm thử hộp trắng và kiểm thử hộp đen nên có được ưu điểm của cả hai phương pháp này

Trong phương pháp này các tester có thể thiết kế nên những trường hợp kiểm thử đặc biệt xung quanh các giao thức kết nối và các loại dữ liệu khác nhau.

- Việc kiểm thử được hoàn thành từ góc nhìn của người dùng chứ không phải từ nhà thiết kế

- Các tester sử dụng phương pháp này không dựa vào các dòng lệnh của hệ thống mà chủ yếu dựa trên các tài liệu định nghĩa giao diện cũng như các tài liệu đặc tả chức năng.

Nhược điểm:

* Vì phương pháp này không dựa trên việc truy cập code của hệ thống nên sẽ không tránh được việc độ bao phủ của các trường hợp kiểm thử bị giới hạn.
* Khi sử dụng phương pháp này thì nhiều trường hợp kiểm thử có thể bị dư thừa nếu mà những nhà thiết kế phần mềm tốn thời gian là vừa phải.
* Không thích hợp để kiểm tra các thuật toán trong hệ thống.
* Các miền dữ liệu có thể sẽ được test nếu các tester có kiến thức về nó.
  + 1. *So sánh các phương pháp kiểm thử*

|  |  |  |
| --- | --- | --- |
| **Kiểm thử hộp đen** | **Kiểm thử hộp trắng** | **Kiểm thử hộp xám** |
| Không cần quan tâm đến các luồng hoạt động trong hệ thống. | Cần nắm được toàn bộ các luồng hoạt động bên trong hệ thống. | Cần có kiến thức nhất định về các luồng hoạt động bên trong hệ thống |
| Được biết đến với các tên gọi khác như: closed-box testing, data-driven testing hoặc functional testing. | Được biết đến với các tên gọi khác như: clear-box testing hoặc code-based  testing. | Được biết đến với các tên gọi khác như: translucent testing. |
| Được thực hiện bởi người dùng cuối, kiểm thử viên và lập trình viên. | Thường thì được hoàn thành bởi kiểm thử viên và lập trình viên. | Được thực hiện bởi người dùng cuối, kiểm thử viên và lập trình viên. |

# CHƯƠNG 2: THIẾT KẾ TEST CASE

## 2.1 Các phương pháp thiết kế test-case (Black-box Testing)

## 2.1.1 Kỹ thuật phân vùng tương đương (Equivalence Partitioning)

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

* Phân vùng tương đương: là kỹ thuật thực hiện test theo từng class đồng giá trị (tập hợp điều kiện cùng một thao tác).
* Tập hợp giá trị input có cùng một kết quả xử lý, tập hợp thời gian có cùng một kết quả xử lý, tập hợp kết quả export được xử lý cùng một giá trị nhập.
* Mục đích : Giảm đáng kể số lượng test case cần phải thiết kế vì với mỗi lớp tương đương ta chỉ cần test trên các phần tử đại diện.
* Chọn tối thiểu một giá trị đại diện từ các class đồng giá trị để tiến hành test.
* Thiết kế ca kiểm thử cho phân vùng tương đương dựa trên sự đánh giá về các vùng tương đương với một điều kiện vào. Vùng tương đương biểu thị một tập cho các trạng thái hợp lệ hay không hợp lệ đối với điều kiện vào
* Thiết kế test case bằng phân vùng tương đương tiến hành theo hai bước:
  + Xác định các lớp tương đương
  + Xác định các ca kiểm thử

Nếu có lỗi xảy ra thì các giá trị khác trong class đồng giá trị cũng sẽ có lỗi giống

nhau.

## 2.1.2 Kỹ thuật phân tích giá trị biên (Boundary Value Analysis)

Phân tích giá trị biên (Boundary Value Analysis): là một kỹ thuật kiểm thử phần mềm có liên quan đến việc xác định biên (ranh giới) của điều kiện mô tả cho các giá trị đầu vào và chọn giá trị ở biên và bên cạnh giá trị biên làm dữ liệu kiểm thử. Phương pháp phân tích giá trị biên sẽ đưa ra các giá trị đặc biệt, bao gồm loại dữ liệu, giá trị lỗi, bên trong, bên ngoài biên giá trị, lớn nhất và nhỏ nhất.

* Test giá trị biên được thực hiện theo trình tự dưới đây:
  + Tìm ra đường biên
  + Quyết định giá trị biên
  + Quyết định giá trị để test
  + Giá trị biên.
  + Dưới giá trị biên. (Nếu là class đồng giá trị)
  + Trên 1 giá trị biên. (Nếu là class đồng giá trị)
* Boundary values:
  + Min – 1
  + Min
  + Max
  + Max + 1

### *2.1.3 Bảng quyết định (Decision Table)*

* Làm giảm số lượng test case không cần thiết so với 2 kỹ thuật trên vì nó loại trừ các phép kết hợp không cần thiết giữa các giá trị biến đầu vào.
* Liệt kê nguyên nhân (cause) – kết quả (result) trong một ma trận. Mỗi cột ma trận đại diện cho 1 phép kết hợp giữa các cause trong trong việc tạo ra 1 result.
* Các bước để tạo bảng quyết định
  + Liệt kê các nguyên nhân trong bảng quyết định
  + Tính tổng số lượng kết hợp giữa các cause
  + Điền vào các cột với tất cả các kết hợp có thể có
  + Rút bớt số lượng các phép kết hợp dư thừa
  + Kiểm tra các phép kết hợp có bao phủ hết mọi trường hợp hay không
  + Bổ sung kết quả vào bảng quyết định

### *2.1.4 Dịch chuyển trạng thái*

Là một kỹ thuật thiết kế kiểm thử phần mềm liên quan đến việc xác định các trường hợp (điều kiện đầu vào) và các hiệu ứng (điều kiện đầu ra). Vì các hệ thống hiện nay đều được phát triển trên nền tảng OOP, do đó, chúng ta có thể có được một đồ thị các đối tượng mà hệ thống định nghĩa và kết nối. Từ đồ thị này, chúng ta dễ dàng biết các mối quan hệ của những đối tượng mà hệ thống xử lý, từ đó sẽ cho chúng ta các kịch bản kiểm thử phù hợp.

## 2.2 Kiểm thử hộp trắng (White-box Testing)

Kiểm thử hộp trắng (White box testing) là một kỹ thuật xác minh giúp các kỹ sư phần mềm có thể sử dụng để kiểm tra mã code của họ hoạt động như dự kiến.

Có các loại white box testing đang tồn tại như sau:

* API testing (application programming interface) – Kiểm thử ứng dụng bằng cách sử dụng các hàm API public và private
* Code coverage – Là việc tạo các trường hợp test để thỏa mãn một số điều kiện bao phủ code - code coverage (ví dụ như, người thiết kế test có thể tạo ra các trường hợp test sao cho tất cả các câu lệnh của chương trình đều được thực thi ít nhất 1 lần)
* Fault injection methods – cải tiến bao phủ một trường hợp bằng cách đưa một số lỗi vào để test các đường dẫn code.
* Mutation testing methods
* Static testing - White box testing bao gồm tất cả các phương pháp kiểm thử tĩnh (ví dụ review code).

*Kiểm thử độ bao phủ*

Phương pháp kiểm thử white box cũng có thể được sử dụng để ước lượng tính trọn vẹn đầy đủ của các tập hợp kiểm thử (test suit) đã được tạo ra bằng phương pháp kiểm thử black box. Điều này cho phép nhóm sản xuất phần mềm xem xét lại các phần của hệ thống ít được test nhất và để chắc chắn rằng các chức năng quan trọng nhất đã được tập trung test kỹ.

*Hai hình thức chung của kiểm thử độ bao phủ code:*

* Bao phủ chức năng - Function coverage, dựa trên việc thực thi các chức năng.
* Bao phủ câu lệnh - Statement coverage, dựa trên số lượng các dòng lệnh đã được thực thi để hoàn thành kiểm thử.

## 2.3 Mô tả đề tài và kế hoạch kiểm thử

### *2.3.1 Mô tả đề tài*

Trong thời đại công nghệ phát triển mạnh mẽ, việc giải trí qua các nền tảng số đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày. Đề tài phát triển một website GameShop trực tuyến không chỉ đáp ứng nhu cầu giải trí đa dạng của người dùng mà còn mang lại giá trị kinh tế và cơ hội kinh doanh lớn cho các nhà phát triển.

Website ShopGame sẽ cung cấp một nền tảng nơi người dùng có thể dễ dàng truy cập và chọn lựa được các game thuộc nhiều thể loại khác nhau, từ hành động nhập vai, kinh dị đến giải đố.

Mục tiêu của đề tài là tạo ra một Website thân thiện với người dùng, có giao diện đẹp mắt và dễ sử dụng, đáp ứng tốt nhu cầu giải trí của mọi người. Đây không chỉ là một dự án công nghệ mà còn là một cơ hội kinh doanh tiềm năng, mang lại lợi ích cho cả người dùng và nhà phát triển.

### *2.3.2 Kế hoạch kiểm thử*

Kiểm thử hộp đen

|  |  |
| --- | --- |
| **Kỹ thuật kiểm thử** | **Chức năng phần mềm** |
| * Phân vùng tương đương | * Xem chi tiết game |
| * Phân tích giá trị biên | * Đăng ký |
| * Bảng quyết định | * Mua tài khoản game |
| * Dịch chuyển trạng thái | * Đăng nhập |

Kiểm thử hộp trắng

|  |  |
| --- | --- |
| **Kỹ thuật kiểm thử** | **Chức năng phần mềm** |
| * Phủ câu lệnh | * Đăng nhập |
| * Phủ nhánh | * Chỉnh sửa thông tin tài khoản game |
| * Phủ đường | * Đăng ký |
| * Phủ điều kiện | * Thêm acc game mới vào hệ thống |

Kiểm thử tự động

|  |  |
| --- | --- |
| **Loại kiểm thử** | **Công cụ kiểm thử** |
| * Kiểm thử chức năng (Đăng nhập) | * Katalon |
| * Kiểm thử chức năng (Đăng ký) | * Katalon |
| * Kiểm thử giao diện | * Katalon |

# CHƯƠNG 3: ÁP DỤNG

## 3.1 Đặc tả

## 3.1.1 Đăng ký

**Mục tiêu**: Cung cấp cho người dùng khả năng tạo tài khoản cá nhân, đảm bảo tính bảo mật và dễ sử dụng. Việc đăng ký giúp hệ thống quản lý người dùng, lưu trữ lịch sử giao dịch.

**Các bước thực hiện:**

Bước 1: Nhập thông tin cá nhân

Email:

* Yêu cầu định dạng đúng của email (ví dụ: admin@gmail.com).
* Độ dài tối đa không quá 30 ký tự.
* Email phải là duy nhất trong hệ thống. Nếu email đã được đăng ký, hệ thống sẽ báo lỗi yêu cầu người dùng nhập một email khác.

Mật khẩu:

* Độ dài từ 8 đến 16 ký tự.
* Mật khẩu phải bao gồm ít nhất:
  + 1 ký tự chữ hoa (A-Z).
  + 1 ký tự chữ thường (a-z).
  + 1 chữ số (0-9).
  + 1 ký tự đặc biệt (ví dụ: !, @, #).
* Người dùng cần nhập mật khẩu hai lần để xác nhận và đảm bảo không có lỗi nhập liệu. Nếu mật khẩu không khớp, hệ thống sẽ yêu cầu nhập lại.

Bước 2: Xác nhận tính duy nhất của tài khoản

Hệ thống kiểm tra xem email đã được sử dụng hay chưa. Nếu email đã tồn tại trong hệ thống, thông báo lỗi sẽ được hiển thị với yêu cầu nhập email khác.

Bước 3: Kiểm tra tính hợp lệ của mật khẩu

Hệ thống sẽ xác nhận rằng mật khẩu thỏa mãn tất cả các tiêu chí bảo mật. Nếu mật khẩu không đáp ứng, người dùng sẽ nhận được thông báo yêu cầu thay đổi mật khẩu theo các tiêu chí đã đề ra.

Bước 4: Xác thực tài khoản qua Email (nếu có)

* Sau khi hoàn tất đăng ký, hệ thống sẽ gửi một email xác thực tới địa chỉ email đã đăng ký. Người dùng cần kiểm tra hộp thư và nhấn vào đường dẫn xác nhận để kích hoạt tài khoản.
* Khi xác thực thành công, tài khoản sẽ chính thức được kích hoạt và người dùng có thể đăng nhập vào hệ thống.

**Các ràng buộc bảo mật:**

* Tất cả các mật khẩu sẽ được mã hóa trước khi lưu trữ trong cơ sở dữ liệu để bảo vệ thông tin cá nhân người dùng.
* Các thông tin đăng ký chỉ được chấp nhận khi đáp ứng đầy đủ yêu cầu về định dạng và bảo mật.

|  |  |
| --- | --- |
| **Tên Use-Case** | Đăng ký tài khoản |
| **Actor** | Người dùng. |
| **Mô tả** | Use-case cho phép user tạo tài khoản để truy cập vào hệ thống để thực hiện những chức năng của mình. |
| **Điều kiện kích hoạt** | Khi người dùng chọn chức năng đăng ký tài khoản từ trang chủ của hệ thống. |
| **Hậu điều kiện** | Người dùng đăng ký thành công. |
| **Luồng sự kiện chính** | 1. Hệ thống hiển thị màn hình đăng ký. 2. Người dùng nhập thông tin cá nhân như tên , năm sinh email, tên đăng nhập và mật khẩu. 3. Hệ thống hiển thị kiểm tra thông tin đăng ký. 4. Nếu thành công hệ thống hiển thị thành công, tạo tài khoản mới và truy cập vào trang chủ bằng tài khoản vừa tạo. 5. Kết thúc Use-Case. |
| **Luồng sự kiện phụ** | **A1-Tài khoản không hợp lệ**: khi người dùng nhập sai tên thông tin cá nhân  1.Hệ thống hiển thị lại màn hình đăng nhập để người dùng nhập lại thông tin và màn hình sẽ thông báo sai thông tin cá nhân  2.Quay lại bước 2 trong luồng sự kiện chính. |

## 3.1.2 Đăng nhập

**Mục tiêu:** Cung cấp cho người dùng quyền truy cập vào tài khoản đã đăng ký để sử dụng các tính năng riêng như lưu trữ lịch sử, danh sách tài khoản game, và quản lý tài khoản cá nhân. Đăng nhập phải bảo đảm tính bảo mật cao và dễ dàng thao tác.

**Các bước thực hiện**:

Bước 1: Nhập thông tin đăng nhập

Email:

* Yêu cầu đúng định dạng email đã đăng ký và không quá 30 ký tự.
* Email phải tồn tại trong cơ sở dữ liệu của hệ thống.

Mật khẩu:

* Độ dài mật khẩu từ 8 đến 16 ký tự.
* Mật khẩu phải khớp với mật khẩu đã đăng ký.

Bước 2: Xác thực thông tin đăng nhập

* Hệ thống sẽ kiểm tra xem email có tồn tại trong cơ sở dữ liệu hay không. Nếu email không tồn tại, hệ thống sẽ thông báo rằng email chưa được đăng ký và yêu cầu người dùng tạo tài khoản mới.
* Hệ thống xác nhận tính chính xác của mật khẩu đã nhập:
* Nếu mật khẩu sai, hệ thống sẽ hiển thị thông báo yêu cầu nhập lại.

Bước 3: Đăng nhập thành công

Sau khi xác thực thành công, người dùng sẽ được chuyển hướng đến trang chính của hệ thống và có thể truy cập các tính năng cá nhân.

**Xử lý khi đăng nhập thất bại:**

Yêu cầu nhập lại.

**Các ràng buộc bảo mật:**

Mã hóa thông tin đăng nhập: Hệ thống mã hóa mật khẩu và các thông tin bảo mật khác để bảo vệ dữ liệu.

|  |  |
| --- | --- |
| **Tên Use-Case** | Đăng Nhập |
| **Actor** | Quản trị viên và người dùng. |
| **Mô tả** | Use-case cho phép admin hoặc user truy cập vào hệ thống để thực hiện những chức năng của mình. |
| **Điều kiện kích hoạt** | Khi người dùng chọn chức năng đăng nhập từ trang chủ của hệ thống. |
| **Tiền điều kiện** | Người dùng phải có tài khoản trên hệ thống. |
| **Hậu điều kiện** | Người dùng đăng nhập thành công. |
| **Luồng sự kiện chính** | 1. Hệ thống hiển thị màn hình đăng nhập. 2. Người dùng nhập tên đăng nhập và mật khẩu. 3. Hệ thống hiển thị kiểm tra thông tin đăng nhập. 4. Nếu thành công hệ thống hiển thị thành công và truy cập đến trang chủ. 5. Kết thúc Use-Case. |
| **Luồng sự kiện phụ** | **A1-Mật khẩu không hợp lệ**: khi người dùng nhập sai tên đăng nhập và mật khẩu  1.Hệ thống hiển thị lại màn hình đăng nhập để người dùng nhập lại thông tin và màn hình sẽ thông báo mật khẩu hoặc tài khoản bị sai.  2.Quay lại bước 2 trong luồng sự kiện chính.  **A2-Quên mật khẩu:** khi người dùng chọn chức năng quên mật khẩu trên màn hình đăng nhập  1.Hệ thống hiển thị màn hình để người dùng nhập email.  2.Người dùng nhập email và chọn chức năng lấy lại mật khẩu.  3.Hệ thống kiểm tra email hợp lệ và gửi liên kết để reset mật khẩu cho người dùng qua email.  4.Hệ thống hiển thị màn hình thông báo thành công  5.Use-case kết thúc. |

## 3.1.4 Mua bán acc game

Đây là tính năng cốt lõi của website ShopGame

* Giao dịch tài khoản trực tuyến: Người dùng có thể xem chi tiết tài khoản, bao gồm thông tin game, cấp độ, vật phẩm, và hình ảnh minh họa trước khi mua.
* Điều chỉnh hiển thị tài khoản: Hệ thống hỗ trợ sắp xếp, lọc tài khoản theo game, mức giá, cấp độ, và trạng thái (còn hàng, đã bán) để người dùng dễ tìm kiếm.
* Lịch sử giao dịch: Hệ thống lưu lại lịch sử mua bán tài khoản, giúp người dùng dễ dàng theo dõi các giao dịch đã thực hiện và tra cứu lại thông tin khi cần..

|  |  |
| --- | --- |
| **Tên Use-Case** | Mua bán acc game |
| **Actor** | Người dùng. |
| **Mô tả** | Use-case cho phép user truy cập vào website để chọn và mua |
| **Điều kiện kích hoạt** | Khi người dùng chọn acc game muốn mua từ trang chủ của hệ thống. |
| **Tiền điều kiện** | Người dùng phải có tài khoản trên hệ thống. |
| **Hậu điều kiện** | Người dùng mua acc game thành công. |
| **Luồng sự kiện chính** | 1. Hệ thống hiển thị màn hình trang chủ. 2. Người dùng chọn vào acc game cần mua 3. Hệ thống hiển thị kiểm tra thông tin acc game 4. Nếu thành công hệ thống hiển thị acc game 5. Kết thúc Use-Case. |

## 3.1.5 Quản lý trang Admin

Trang admin là công cụ quan trọng để quản trị viên quản lý toàn bộ acc game và tương tác của người dùng, đảm bảo chất lượng dịch vụ và bảo mật.

Quản lý tài khoản game:

* Cho phép admin thêm mới, xóa, và chỉnh sửa thông tin acc game bao gồm tiêu đề, mô tả, thể loại.

Báo cáo và Thống kê:

* Cung cấp báo cáo tổng quan về lượng người truy cập, số lượng tài khoản đã bán và các tài khoản game bán chạy nhất.
  + Thống kê doanh thu, lượng giao dịch theo thời gian, theo thể loại game, giúp admin đánh giá hiệu quả hoạt động và định hướng phát triển nội dung phù hợp.

Trang admin không chỉ là trung tâm quản lý nội dung mà còn giúp duy trì chất lượng, bảo mật cho hệ thống và tạo trải nghiệm tốt nhất cho người dùng.

|  |  |
| --- | --- |
| **Tên Use-Case** | Quản lý tài khoản game |
| **Actor** | Quản trị viên (Admin) |
| **Mô tả** | Use-case cho phép Admin:  1. Thêm mới tài khoản game vào hệ thống  2. Chỉnh sửa thông tin tài khoản game đã có  3. Xóa tài khoản game khỏi hệ thống |
| **Điều kiện kích hoạt** | Khi Admin chọn chức năng: Thêm mới, Cập nhật, hoặc Xóa acc game trên giao diện quản lý |
| **Tiền điều kiện** | Admin phải đăng nhập thành công vào hệ thống bằng tài khoản quản trị |
| **Hậu điều kiện** | Admin đã thêm mới, cập nhật hoặc xóa tài khoản game thành công |
| **Luồng sự kiện chính** | 1. Hệ thống hiển thị danh sách tất cả tài khoản game đã có. 2. Admin chọn một tài khoản game để thực hiện thao tác Thêm mới / Cập nhật / Xóa.   **2.1 Admin chọn chức năng Thêm mới:**  Hệ thống hiển thị biểu mẫu nhập thông tin tài khoản game.  Admin nhập các thông tin như: Tên game, server, rank, cấp độ, giá bán, mô tả, hình ảnh.  Hệ thống kiểm tra dữ liệu nhập vào.  Nếu hợp lệ, hiển thị thông báo “Thêm tài khoản thành công”.  **2.2 Admin chọn chức năng Cập nhật:**  Hệ thống hiển thị biểu mẫu chứa thông tin hiện tại của tài khoản game.  Admin chỉnh sửa thông tin cần cập nhật.  Hệ thống kiểm tra dữ liệu mới.  Nếu hợp lệ, hiển thị thông báo “Cập nhật thành công”.  **2.3 Admin chọn chức năng Xóa:**  Hệ thống hiển thị thông báo xác nhận xóa.  Admin nhấn xác nhận.  Hệ thống xóa tài khoản game khỏi hệ thống và hiển thị thông báo “Xóa thành công”.  Kết thúc use-case. |
| **Luồng sự kiện phụ** | A1 – Thêm mới thất bại:   * Khi Admin nhập thiếu thông tin hoặc sai định dạng. * Hệ thống hiển thị thông báo lỗi → Quay lại bước 2.1.   A2 – Cập nhật thất bại:   * Khi Admin nhập thiếu thông tin hoặc sai định dạng. * Hệ thống hiển thị thông báo lỗi → Quay lại bước 2.2. |

## 3.2 Kiểm thử hộp đen

*3.2.1 Phương pháp phân vùng tương đương (Xem tài khoản game)*

Các Test Case:

• Test Case 1: Xem chi tiết tài khoản khi đã đăng nhập

→ Tài khoản game hiển thị đầy đủ thông tin (tên game, rank, skin, giá, mô tả), hình ảnh rõ nét, không bị lỗi hoặc thiếu nội dung.

• Test Case 2: Xem tài khoản khi không có kết nối mạng

→ Hiển thị thông báo: "Không có kết nối mạng, vui lòng thử lại sau", không hiển thị chi tiết tài khoản.

• Test Case 3: Xem tài khoản khi mạng yếu hoặc chậm

→ Trang hiển thị chậm, hệ thống hiện thông báo "Đang tải...". Hình ảnh có thể nén lại để tải nhanh hơn.

• Test Case 4: Xem tài khoản có hình ảnh chất lượng cao

→ Hình ảnh tài khoản hiển thị sắc nét, đầy đủ, không bị vỡ hay mờ.

• Test Case 5: Xem tài khoản có hình ảnh chất lượng thấp

→ Hình ảnh hiển thị mờ hơn nhưng tải nhanh, vẫn đảm bảo đầy đủ thông tin cần thiết.

• Test Case 6: Xem tài khoản không tồn tại

→ Hiển thị thông báo: "Tài khoản không tồn tại hoặc đã bị gỡ", không hiển thị thông tin.

• Test Case 7: Xem tài khoản khi chưa đăng nhập (yêu cầu đăng nhập)

→ Hiển thị thông báo: "Vui lòng đăng nhập để xem chi tiết tài khoản" và chuyển đến trang đăng nhập.

• Test Case 8: Tiếp tục xem tài khoản đã xem trước đó

→ Hệ thống tự động chuyển đến đúng tài khoản trong lịch sử mà người dùng đã xem, không cần tìm lại.

• Test Case 9: Xem tài khoản bị giới hạn quyền truy cập

→ Hiển thị thông báo: "Bạn cần nâng cấp tài khoản để xem thông tin này" và không hiển thị chi tiết.

• Test Case 10: Xem tài khoản có mô tả tiếng Việt

→ Hiển thị mô tả tiếng Việt rõ ràng, đúng ngữ pháp, không lỗi font.

• Test Case 11: Xem tài khoản không có mô tả

→ Hiển thị mặc định thông báo: "Chưa có mô tả cho tài khoản này".

• Test Case 12: Xem tài khoản với thông tin đa ngôn ngữ

→ Cho phép chọn ngôn ngữ hiển thị khác (ví dụ: tiếng Anh), hệ thống hiển thị đúng ngôn ngữ đã chọn..

### *3.2.2 Kiểm thử phân tích giá trị biên (Đăng ký)*

L   : Trang Login

H  : Trang Home

M1: Vui lòng nhập username

M2: Vui lòng nhập password

M3: Username không hợp lệ

M4: Password không hợp lệ

M5: Vui lòng nhập SDT

M6: SDT không hợp lệ

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | **C1** | **C2** | **C3** | **C4** | **C5** | **C6** | **C7** |
| **Điều kiện** | Username |  | I | V | V | V | V | V |
| Password | - | - | B | I | V | V | V |
| SDT | - | - | - | - | B | I | V |
| **Hành động** | Thông báo lỗi | M1 | M3 | M2 | M4 | M5 | M6 |  |
| Chuyển trang | L | L | L | L | L | L | H |

Bảng quyết định:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Cột** | **Đầu vào** | | | **Hành động** | |
| **username** | **password** | **SDT** | **Thông báo** | **Chuyển trang** |
| **C1** |  | Aa12345@ | 0232335643 | M1 | L |
| **C2** | minh@gmail.com | Aa12345@ | 0232335643 | M3 | L |
| minh@gmail.com | Aa12345@ | 0232335643 | M3 | L |
| minh@gmail.com | Aa12345@ | 0232335643 | M3 | L |
| minh@gmail.com | Aa12345@ | 0232335643 | M3 | L |
| **C3** | minh@gmail.com |  | 0232335643 | M2 | L |
| **C4** | minh@gmail.com | Aa123@  (6 ký tự) | 0232335643 | M4 | L |
| minh@gmail.com | Aa34567890123456@  ( 17 ký tự) | 0232335643 | M4 | L |
| minh@gmail.com | **ABCDEFGH**  (chỉ chữ Hoa) | 0232335643 | M4 | L |
| minh@gmail.com | **!@#$%^&\***  (chỉ  ký tự đặc biệt) | 0232335643 | M4 | L |
| minh@gmail.com | **Abcdefgh**  (Chỉ chữ thường) | 0232335643 | M4 | L |
| minh@gmail.com | **12345678**  (chỉ số) | 0232335643 | M4 | L |
| minh@gmail.com | **ABCDabcd**  (chữ Hoa và chữ thường) | 0232335643 | M4 | L |
| minh@gmail.com | **ABCD!@#$**  (chữ Hoa và ký tự đặc biết) | 0232335643 | M4 | L |
| minh@gmail.com | **ABCD1234**  (chữ Hoa và số) | 0232335643 | M4 | L |
| minh@gmail.com | **abcd!@#$**  (chữ thường và ký tự đặc biệt) | 0232335643 | M4 | L |
| minh@gmail.com | **abcd1234**  (chữ thường và số) | 0232335643 | M4 | L |
| minh@gmail.com | **!@#$1234**  (ký tự đặc biệt và số) | 0232335643 | M4 | L |
| minh@gmail.com | **ABCabc!@**  (chữ Hoa, chữ thường và ký tự đặc biệt) | 0232335643 | M4 | L |
| minh@gmail.com | **ABCabc12**  (chữ Hoa , chữ thường và số ) | 0232335643 | M4 | L |
| minh@gmail.com | **ABC!@#12**  (chữ Hoa, ký tự đặc biệt và số) | 0232335643 | M4 | L |
| minh@gmail.com | **abc!@#12**  (chữ thường, ký tự đặc biệt và số) | 0232335643 | M4 | L |
| **C5** | minh@gmail.com (đã tồn tại) | Aa12345@  (8 ký tự) | - | M5 | L |
| **C6** | minh@gmail.com (18 ký tự) | Aa12345@  (8 ký tự) | 0232335643434 | M6 | L |
| minh@gmail.com (30 ký tự) | Aa3456789012345@  (16 ký tự) | 0232335 | M6 | L |
| **C7** | minh@gmail.com (18 ký tự) | Aa12345@  (8 ký tự) | 0232335643 |  | H |
| minh@gmail.com (30 ký tự) | Aa3456789012345@  (16 ký tự | 0232335643 |  | H |

### *3.2.3 Kiểm thử bảng quyết định (Tìm kiếm tài khoản game)*

B   : Rỗng (không nhập từ khoá)

I : Nhập từ khoá không hợp lệa

V1: Nhập từ khoá hợp lệ (có phim trong csdl)

V2: Nhập từ khoá hợp lệ (không có phim trong csdl)

M1: Nhập thông tin phim cần tìm

M2: Từ khoá không hợp lệ

M3: Không có phim cần tìm

T   : Trang tìm kiếm phim

H  : Trang Home

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **C1** | **C2** | **C3** | **C4** |
| **Điều kiện** | B | I | V1 | V2 |
| **Hành động** | M1 | M2 |  | M3 |
| H | H | T | H |

*Bảng quyết định:*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Cột** | **Đầu vào** | **Hành động** | | | | |
| M1 | M2 | M3 | **H** | **T** |
| **C1** | “ ” | v | x | x | v | x |
| **C2** | "a" (1 ký tự) | x | v | x | v | x |
| “abcd…” (51 ký tự) | x | v | x | v | x |
| “@@@@” | x | v | x | v | x |
| “a@” | x | v | x | v | x |
| “A@” | x | v | x | v | x |
| “1@” | x | v | x | v | x |
| “aA@” | x | v | x | v | x |
| “1a@” | x | v | x | v | x |
| “Aa1@” | x | v | x | v | x |
| “1A@” | x | v | x | v | x |
| “xyz…” (60 ký tự) | x | v | x | v | x |
| **C3** | “Do” (2 ký tự) | x | x | x | x | v |
| “Doa” (3 ký tự) | x | x | x | x | v |
| “Do…” (49 ký tự) | x | x | x | x | v |
| “The…” (50 ký tự) | x | x | x | x | v |
| “do” | x | x | x | x | v |
| “Do” | x | x | x | x | v |
| “DO” | x | x | x | x | v |
| “1010” | x | x | x | x | v |
| “E123” | x | x | x | x | v |
| “e123” | x | x | x | x | v |
| “Ab123” | x | x | x | x | v |
| “Doraemon” (23 ký tự) | x | x | x | x | v |
| **C4** | “zabc” | x | x | v | v | x |
| “xy” | x | x | v | v | x |
| “xyz” | x | x | v | v | x |
| “xyz…” (49 ký tự) | x | x | v | v | x |
| “xyz…” (50 ký tự) | x | x | v | v | x |

### *3.2.4 Kiểm thử dịch chuyển trạng thái (đăng nhập)*

Các khía cạnh của hệ thống được mô tả thông qua lược đồ trạng thái. Hệ thống sẽ có nhiều trạng thái khác nhau, sự dịch chuyển từ một trạng thái này sang trạng thái khác được quyết định bởi một sự kiện nào đó.

**Các trạng thái:**

* Logged Out (Đã đăng xuất / chưa đăng nhập) - LO
* Logging In (Đang đăng nhập) - LingI
* Logged In (Đã đăng nhập) -LedI
* Login Failed (Đăng nhập thất bại) - LF

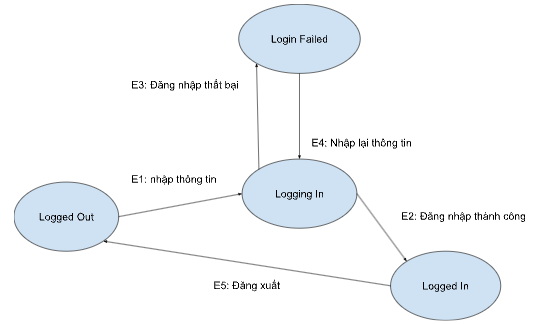
**Sự dịch chuyển:**

* Logged Out  → Logging In
* Logging In → Logged In, Login Failed
* Login Failed → Logging In
* Logged In → Logged Out

**Các sự kiện:**

* E1: Nhập thông tin đăng nhập
* E2: Thông tin chính xác → đăng nhập thành công
* E3: Thông tin không chính xác → đăng nhập thất bại
* E4: Nhập lại thông tin
* E5: Đăng xuất

**Các hành động:**



Sơ đồ lược đồ dịch chuyển

**Bảng lược đồ dịch chuyển:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Trạng thái** | **E1** | **E2** | **E3** | **E4** | **E5** |
| **LO** | LingI |  |  |  |  |
| **LingI** |  | LedI | LF |  |  |
| **LedI** |  |  |  |  | LO |
| **LF** |  |  |  | LingI |  |

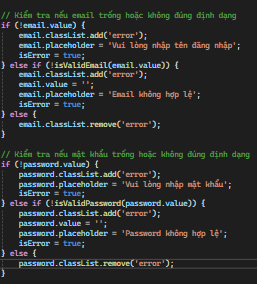
**Các TestCase:**

* Logged Out→ Logging In,
* Logging In→ Logged In,
* Logging In→ Login Failed,
* Login Failed→ Logging In,
* Logged In→ Logged Out

## 3.1 Kiểm thử hộp trắng

### *3.1.1 Phương pháp phủ câu lệnh (Đăng nhập)*

Phủ câu lệnh (statement coverage): mỗi câu lệnh được thực thi ít nhất một lần.



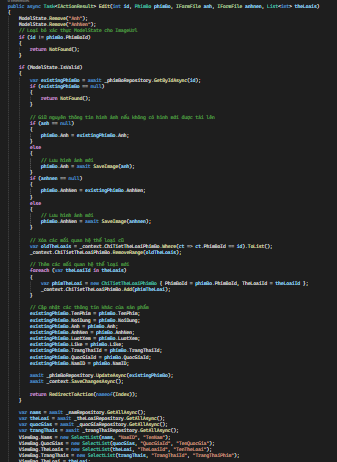
Chức năng đăng nhập

Bảng Test Case:

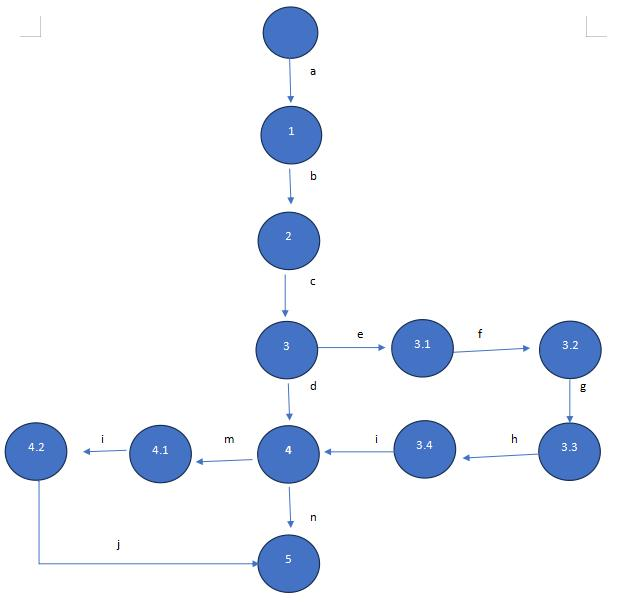
|  |  |  |
| --- | --- | --- |
| **Đầu vào** | | **Đầu ra mong muốn** |
| **Tài khoản** | **Mật khẩu** |
| admin@gmail.com | admin@123 | Đăng nhập thành công |
| admin@gmail.com | admin@124 | Mật khẩu không hợp lệ |
| admin | - | Email không hợp lệ |
| x | - | Vui lòng nhập tên đăng nhập |
| admin@gmail.com | Abcdxyz1298@LHK | Password không hợp lệ |
| admin@gmail.com | x | Vui lòng nhập mật khẩu |

### *3.1.2 Phương pháp phủ nhánh (Chỉnh sửa phim)*

Phủ nhánh có nghĩa là các cạnh được đi qua ít nhất một lần. Để phủ nhánh phải thiết kế dữ liệu kiểm thử sao cho mỗi nút vị từ (predicate) xảy ra tất cả các kết quả (true/false) có thể của nó, nên phủ nhánh còn gọi là phủ quyết định (decision coverage).



Chức năng chỉnh sửa phim



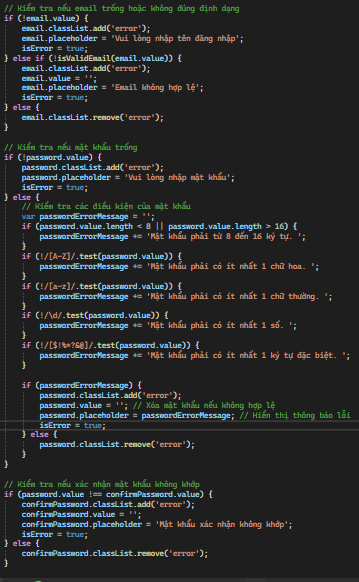
Sơ đồ kiểm thử phủ nhánh

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Đường dẫn** | **Đầu vào** | | | | **Đầu ra mong muốn** |
| **phimBo** | **anh** | **anhnen** | **theloais** |
| a,b,c,m,n | x | - | - | - | Không tìm được |
| A,,b,d | - | - | - | - | Không hợp lệ |
| A,b,c,d,l,,n,m | v | v | v | v | Thành công |
| A,b,c,l.n | x | x | x | - | Không thành công |
| A,b,c,n,m | v | v | v | - | Cập nhật thành công |
| A,b | v | x | x | x | Không update thể loại |

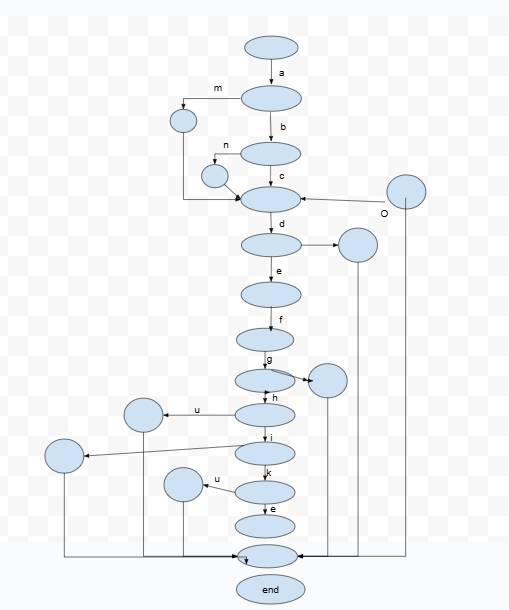
Bảng Test Case:

### *3.1.3 Phương pháp phủ đường dẫn (Đăng ký)*

Phủ đường dẫn (path coverage): mỗi đường dẫn qua ít nhất một lần. Đường là một tập các nhánh, nên phủ đường chắc chắn phủ nhánh, nhưng ngược lại chưa chắc đúng.



Chức năng đăng ký



Sơ đồ kiểm thử phủ đường dẫn

Bảng Test Case:

|  |  |  |  |
| --- | --- | --- | --- |
| **Đầu vào** | | | **Đầu ra mong muốn** |
| **Tài khoản** | **Mật khẩu** | **Xác nhận mk** |
| admin@gmail.com | admin@123 | admin@123 | Đăng ký thành công |
| admin@gmail.com | admin@123 | admin@124 | Mật khẩu xác nhận không khớp |
| admin | - | - | Email không hợp lệ |
| x | - | - | Vui lòng nhập tài khoản |
| admin@gmail.com | Abcdxyz1298@LHK | - | Mật khẩu từ 8 -16 ký tự |
| a@gmail.com | Abcde123 | - | Mật khẩu có ít nhât 1 chữ in hoa |
| b@gmail.com | ABCD1234 | - | Mật khẩu có ít nhất 1 chữ thường |
| c@gmail.com | ABCDabcd | - | Mật khẩu có ít nhất 1số |
| d@gmail.com | ABCD1234 | - | Mật khẩu có ít nhất 1 ký tự đặt biệt |
| e@gmail.com | x | - | Vui lòng nhập mật khẩu |

### *3.1.4 Phương pháp phủ điều kiện (Thêm phim)*

Phủ điều kiện (condition coverage): mỗi điều kiện trong các vị từ được thực hiện ít nhất một lần cho cả trường hợp true và false (không bắt buộc các kết hợp giữa chúng). Phủ điều kiện chứa chắc đảm bảo phủ các nhánh.

Bảng Test Case:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Đường dẫn** | **Đầu vào** | | | | **Đầu ra mong muốn** |
| **phimBo** | **anh** | **anhnen** | **theloais** |
| a,c,e,g,h,i,k,l,j,o | V | V | V | V | Trả về trang Index |
| a,c,d,i,k,l,j,m,n | V | I | V | V | Hiển thị thông báo lỗi do ảnh không hợp lệ |
| a,c,e,f,i,k,l,j,m,n | V | V | I | V | Hiển thị thông báo lỗi do ảnh nền không hợp lệ |
| a,c,e,g,h,i,,j,m,n | V | V | V | I | Hiển thị thông báo lỗi do danh sách thể loại phim không hợp lệ |
| a,b,n | I | - | - | - | Trả về trang thêm với dữ liệu đã nhập |

# CHƯƠNG 4: TEST CASE VÀ BUGSUMARY

## 4.1 Công cụ kiểm thử tự động katalon

Kiểm thử chức năng Đăng nhập, mua acc game

### *4.1.1 Giới thiệu*

Katalon Là một bộ công cụ toàn diện cho kiểm thử tự động hóa ứng dụng trên web và điện thoại di động. Công cụ này bao gồm một gói đầy đủ các tính năng mạnh mẽ giúp vượt qua những thách thức phổ biến trong tự động hóa thử nghiệm giao diện web, ví dụ như pop-up, iFrame và wait-time. Giải pháp thân thiện và linh hoạt này giúp tester thực hiện công tác kiểm tra tốt hơn, làm việc nhanh hơn và khởi chạy phần mềm chất lượng cao nhờ vào sự hông minh mà nó cung cấp cho toàn bộ quá trình tự động hóa kiểm thử.

*4.1.2 Các tính năng chính*

Tích hợp sẵn các công cụ test: Katalon tích hợp sẵn Selenium và Appium, giúp viết test nhanh mà không cần cài đặt rời rạc.

Giao diện kéo – thả dễ dùng: Hỗ trợ record & playback, người dùng không cần biết lập trình vẫn có thể tạo test case.

Hỗ trợ kiểm thử đa nền tảng: Kiểm thử được cả web, desktop, mobile (Android & iOS), và API trên cùng một nền tảng.

Tạo test suite dễ dàng: Dễ dàng nhóm nhiều test case thành một bộ test suite và chạy tự động theo lịch.

Hỗ trợ dữ liệu kiểm thử (Data-driven testing): Có thể kết nối file Excel, CSV, hoặc database để test với nhiều dữ liệu khác nhau.

Hệ thống báo cáo chi tiết: Tự động tạo report test bằng hình ảnh, log lỗi, biểu đồ kết quả trực quan.

## 4.2 TestCase:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Testcase ID** | **Sub-Module** | **Testcase Summary** | **Priority** | **Pre-condition** | **Pre-steps** | **Step by step** | **Expected Result** | **Actual Result** | **Test Result** | **Note** |
| TC1 | Đăng nhập | Đăng nhập bằng email và mật khẩu hợp lệ | 1 | Người dùng đã có tài khoản | Mở trang đăng nhập | 1. Vào web 2. Nhập email hợp lệ 3. Nhập mật khẩu hợp lệ 4. Nhấn nút "Đăng nhập" | Hệ thống đăng nhập thành công và chuyển đến trang chủ |  | Passed | Cần kết nối internet |
| TC2 | Trang chủ | Tải thành công trang chủ với video nền | 2 | Truy cập internet | Mở trang chủ | 1. Truy cập trang chủ | Video nền hiển thị & phát Hiển thị đúng menu và giao diện |  | Passed |  |
| TC3 | Trang chủ | Kiểm tra âm thanh nền tự động phát (ẩn) | 3 | Browser cho phép autoplay | Mở trang chủ | 1. Vào trang chủ | Âm thanh nền phát tự động nếu được phép |  | Passed | Một số browser chặn |
| TC4 | Trang chủ | Chọn danh mục "Account VIP" từ trang chủ | 2 | Người dùng đã đăng nhập thành công | Truy cập trang chủ | 1. Vào website 2. Nhấn “GENSHIN IMPACT ACCOUNTS”, chọn “Account VIP” | Điều hướng tới trang danh sách tài khoản VIP |  | Passed | Kiểm tra điều hướng đúng |
| TC5 | Danh sách account | Truy cập chi tiết tài khoản VIP từ danh sách | 2 | Đã ở trang danh sách tài khoản VIP | Truy cập danh sách tài khoản VIP | 1. Xem danh sách tài khoản 2. Nhấn “Xem Chi Tiết” tài khoản đầu tiên | Điều hướng đến trang chi tiết tài khoản |  | Passed |  |
| TC6 | Chi tiết tài khoản | Nhấn nút “Mua Ngay” để chuyển sang giao dịch | 1 | Người dùng đang xem chi tiết tài khoản | Truy cập trang chi tiết | 1. Nhấn nút “Mua Ngay” | Điều hướng đến trang mua tài khoản |  | Passed |  |
| TC7 | Giao dịch | Giao dịch mua tài khoản khi số dư không đủ | 1 | Người dùng không đủ tiền trong tài khoản | Đang ở trang mua tài khoản | 1. Nhấn 'Mua Ngay' tài khoản giá cao hơn số dư 2. Quan sát thông báo | Hệ thống báo ❌ 'Không đủ tiền' Hiện link nạp tiền |  | Failed | Thông báo đúng |
| TC8 | Giao dịch | Kiểm tra nút 'Nạp thêm tiền' hoạt động đúng | 2 | Đang hiển thị thông báo lỗi không đủ tiền | Đã thực hiện TC5 | 1. Nhấn vào 'nạp thêm tiền' | Điều hướng đến trang nap\_tien.php |  | Passed | Kiểm tra link hoạt động |
| TC9 | Giao dịch | Kiểm tra nút 'Lịch sử giao dịch' từ lỗi mua | 3 | Đang ở trang báo lỗi mua tài khoản | Đã thực hiện TC5 | 1. Nhấn 'Xem lịch sử giao dịch' | Điều hướng đến history.php |  | Passed |  |
| TC10 | Trang chủ | Truy cập trang chủ từ trang con | 3 | Đang ở bất kỳ trang con nào | VD: trang VIP hoặc mua tài khoản | 1. Nhấn '🏠 Quay về trang chủ' | Quay lại index.php |  | Passed |  |
| TC11 | Trang chủ → Nạp tiền | Truy cập từ trang chủ và chọn nạp tiền | 1 | Đã đăng nhập tài khoản | Mở trình duyệt, đăng nhập | 1. Truy cập index.php 2. Nhấn vào nút hoặc menu 'Nạp tiền' (nếu có hoặc click logo → top up) 3. Kiểm tra điều hướng đúng | Điều hướng đến đúng trang nap\_tien.php và hiển thị form nạp tiền |  | Passed | Có thể cần thiết kế thêm nút 'Nạp tiền' rõ ràng trên trang chủ |
| TC12 | Nạp bằng thẻ cào | Chọn nạp tiền bằng thẻ Viettel - đúng định dạng | 1 | Đã vào nap\_tien.php | Nhập thông tin đầy đủ | 1. Chọn hình thức 'Thẻ cào' 2. Chọn Viettel 3. Chọn mệnh giá 100.000đ 4. Nhập mã thẻ: 1234567890123 5. Nhập số seri: 987654321 6. Nhấn 'Nạp tiền' | Hiển thị thông báo nạp tiền thành công (hoặc đang xử lý) |  | Passed | Test xử lý backend sẽ xác nhận sau |
| TC13 | Nạp bằng thẻ cào lỗi | Nhập mã thẻ sai hoặc để trống | 2 | Đang ở nap\_tien.php | Chọn đúng nhà mạng nhưng không nhập mã | 1. Chọn Viettel 2. Chọn 20.000đ 3. Để trống mã hoặc nhập sai định dạng 4. Nhấn 'Nạp tiền' | Hiển thị lỗi 'Mã thẻ không hợp lệ' hoặc cảnh báo nhập thiếu |  | Passed |  |
| TC14 | Nạp bằng chuyển khoản STK | Nạp tiền bằng chuyển khoản - chọn đúng STK | 1 | Đã chuyển sang chế độ bank\_transfer | Thực hiện thao tác đầy đủ | 1. Chọn 'Chuyển khoản ngân hàng' 2. Chọn hình thức 'STK' 3. Chọn mệnh giá 160.000đ 4. Kiểm tra hiển thị thông tin STK 5. Nhấn 'Nạp tiền' | Hiển thị hướng dẫn chuyển khoản & thông báo đã gửi yêu cầu nạp |  | Passed | Có thể gửi form để admin xác minh |
| TC15 | Nạp bằng QR Code | Kiểm tra hiển thị mã QR sau khi chọn QR | 1 | Đã chọn bank\_transfer | Kiểm tra thao tác thay đổi option | 1. Chọn 'Chuyển khoản ngân hàng' 2. Chọn 'QR Code' 3. Quan sát phần mã QR hiển thị | Ảnh QR Code hiện đúng, thông tin rõ ràng |  | Passed |  |
| TC16 | Chuyển đổi qua lại giữa các phương thức | Kiểm tra việc ẩn/hiện form khi đổi loại nạp | 2 | Đã truy cập nap\_tien.php | Chuyển qua lại các combo box | 1. Chọn lần lượt các hình thức 2. Quan sát giao diện thay đổi 3. Đảm bảo không bị chồng form | Các block hiển thị đúng theo loại được chọn, không lỗi giao diện |  | Passed | Quan trọng với UI test |
| TC17 | Chọn game | Kiểm tra các nút chọn game hiển thị và hoạt động đúng | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Kiểm tra sự xuất hiện của 4 nút game (Genshin Impact, Honkai, Zenless Zone Zero, Wuthering Waves). 2. Kiểm tra nút 'Genshin Impact' có lớp active. 3. Nhấn vào từng nút và kiểm tra URL. | Các nút chọn game hiển thị đúng, nút 'Genshin Impact' sẽ có lớp active mặc định, nhấn vào các nút sẽ thay đổi URL tương ứng. |  | Passed |  |
| TC18 | Form thêm tài khoản | Kiểm tra form thêm tài khoản có kiểm tra và gửi đúng | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Điền đầy đủ dữ liệu hợp lệ (Tên đăng nhập, Mật khẩu, Giá tiền, Ảnh, Loại, Mô tả). 2. Nhấn 'Thêm'. 3. Kiểm tra xem tài khoản mới có xuất hiện trong bảng. | Tài khoản mới sẽ được thêm vào bảng và form sẽ được reset. |  | Passed |  |
| TC19 | Bảng tài khoản | Kiểm tra bảng tài khoản hiển thị đúng dữ liệu | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Kiểm tra bảng tài khoản có hiển thị các cột (ID, Tên đăng nhập, Mật khẩu, Giá tiền, Loại, Ảnh, Mô tả, Hành động). | Bảng tài khoản sẽ hiển thị đúng dữ liệu (ID, Tên đăng nhập, Mật khẩu, v.v.). |  | Passed |  |
| TC20 | Bảng tài khoản | Kiểm tra nút 'Sửa' hoạt động đúng | 1 | Người dùng đã vào trang quản lý tài khoản và có tài khoản trong bảng | Mở trang quản lý tài khoản và tìm tài khoản | 1. Nhấn vào nút 'Sửa' của tài khoản. 2. Kiểm tra xem trang sửa tài khoản có mở với các dữ liệu của tài khoản được điền sẵn. | Trang sửa tài khoản sẽ mở với các dữ liệu của tài khoản đã được điền sẵn. |  | Passed |  |
| TC21 | Bảng tài khoản | Kiểm tra nút 'Xóa' hoạt động đúng và có xác nhận | 1 | Người dùng đã vào trang quản lý tài khoản và có tài khoản trong bảng | Mở trang quản lý tài khoản và tìm tài khoản | 1. Nhấn vào nút 'Xóa' của tài khoản. 2. Xác nhận việc xóa tài khoản. 3. Kiểm tra tài khoản có bị xóa khỏi bảng. | Tài khoản sẽ bị xóa sau khi xác nhận và không còn xuất hiện trong bảng. |  | Passed |  |
| TC22 | Kiểm tra validation form | Kiểm tra xác thực form khi gửi dữ liệu thiếu | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Để trống một hoặc nhiều trường bắt buộc (Ví dụ: Tên đăng nhập, Mật khẩu, Giá tiền). 2. Nhấn 'Thêm'. | Hệ thống sẽ hiển thị thông báo lỗi yêu cầu các trường bắt buộc. |  | Passed |  |
| TC23 | Kiểm tra validation form | Kiểm tra trường 'Giá tiền' chỉ chấp nhận số hợp lệ | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập giá trị không hợp lệ (ví dụ: chữ cái hoặc ký tự đặc biệt) vào trường 'Giá tiền'. 2. Nhấn 'Thêm'. | Hệ thống sẽ hiển thị thông báo lỗi yêu cầu nhập giá trị hợp lệ cho trường 'Giá tiền'. |  | Passed |  |
| TC24 | Gửi form thêm tài khoản | Kiểm tra form gửi và thêm tài khoản thành công | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Điền đầy đủ dữ liệu hợp lệ. 2. Nhấn 'Thêm'. 3. Kiểm tra tài khoản mới có xuất hiện trong bảng. | Tài khoản mới sẽ được thêm vào bảng với dữ liệu đã nhập. |  | Passed |  |
| TC25 | Bảng tài khoản | Kiểm tra hiển thị ảnh tài khoản | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Kiểm tra tài khoản có ảnh sẽ hiển thị đúng ảnh. 2. Tài khoản không có ảnh sẽ hiển thị 'Không có ảnh'. | Tài khoản có ảnh sẽ hiển thị ảnh, tài khoản không có ảnh sẽ hiển thị 'Không có ảnh'. |  | Passed |  |
| TC26 | Chọn game | Kiểm tra URL thay đổi khi chuyển đổi game | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhấn vào từng nút game (ví dụ: Genshin Impact, Honkai). 2. Kiểm tra xem URL có thay đổi với tham số game= tương ứng. | URL sẽ thay đổi để phản ánh game đã chọn, ví dụ: '?game=genshin'. |  | Passed |  |
| TC27 | Gửi form thêm tài khoản | Kiểm tra trường "Ảnh" nhập tên file hợp lệ | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập tên file hợp lệ vào trường "Ảnh". 2. Nhấn "Thêm". 3. Kiểm tra xem ảnh có hiển thị đúng không. | Hệ thống sẽ hiển thị ảnh đúng nếu tên file hợp lệ, nếu không sẽ không hiển thị ảnh. |  | Passed |  |
| TC28 | Chọn game | Kiểm tra việc chọn game "Genshin Impact" hiển thị tài khoản đúng | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhấn vào nút "Genshin Impact". 2. Kiểm tra bảng tài khoản hiển thị các tài khoản thuộc game "Genshin Impact". | Bảng tài khoản sẽ chỉ hiển thị các tài khoản thuộc game "Genshin Impact". |  | Passed |  |
| tc29 | Bảng tài khoản | Kiểm tra chức năng tìm kiếm trong bảng tài khoản | 2 | Người dùng đã có tài khoản trong bảng | Mở trang quản lý tài khoản | 1. Nhập từ khóa vào ô tìm kiếm. 2. Kiểm tra kết quả tìm kiếm có hiển thị tài khoản tương ứng với từ khóa. | Kết quả tìm kiếm sẽ hiển thị tài khoản tương ứng với từ khóa nhập vào. |  | Passed |  |
| TC30 | Bảng tài khoản | Kiểm tra xem hệ thống có phân trang khi danh sách tài khoản quá dài không | 1 | Người dùng đã có nhiều tài khoản trong bảng | Mở trang quản lý tài khoản | 1. Kiểm tra số lượng tài khoản hiển thị trên một trang. 2. Kiểm tra có nút phân trang không. | Hệ thống sẽ hiển thị phân trang và không hiển thị hết tất cả tài khoản trên một trang duy nhất. |  | Passed |  |
| TC31 | Bảng tài khoản | Kiểm tra nút "Xóa" khi tài khoản đã bị xóa không xuất hiện trong bảng | 1 | Người dùng đã xóa tài khoản trước đó | Mở trang quản lý tài khoản | 1. Xóa tài khoản trong bảng. 2. Kiểm tra tài khoản đó có còn xuất hiện trong bảng không. | Tài khoản đã bị xóa sẽ không còn xuất hiện trong bảng tài khoản. |  | Passed |  |
| TC32 | Gửi form thêm tài khoản | Kiểm tra trường "Giá tiền" nhập giá trị âm | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập giá trị âm vào trường "Giá tiền". 2. Nhấn "Thêm". | Hệ thống sẽ hiển thị thông báo lỗi yêu cầu nhập giá trị hợp lệ cho trường "Giá tiền". |  | Passed |  |
| TC33 | Gửi form thêm tài khoản | Kiểm tra trường "Username" trùng với tài khoản đã có | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập tên đăng nhập đã tồn tại vào trường "Username". 2. Nhấn "Thêm". | Hệ thống sẽ hiển thị thông báo lỗi tên đăng nhập đã tồn tại. |  | Failed |  |
| TC34 | Gửi form thêm tài khoản | Kiểm tra trường "Mô tả" có thể nhập văn bản dài | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập mô tả dài vào trường "Mô tả". 2. Nhấn "Thêm". | Hệ thống sẽ chấp nhận mô tả dài và hiển thị đúng trong bảng tài khoản. |  | Passed |  |
| TC35 | Bảng tài khoản | Kiểm tra nút "Sửa" với tài khoản đã được thay đổi thông tin | 1 | Người dùng đã vào trang quản lý tài khoản và có tài khoản trong bảng | Mở trang quản lý tài khoản và chọn tài khoản cần sửa | 1. Nhấn vào nút "Sửa" của tài khoản. 2. Thay đổi thông tin (ví dụ: tên đăng nhập, giá tiền). 3. Nhấn "Lưu". 4. Kiểm tra thông tin trong bảng có được cập nhật đúng. | Thông tin tài khoản sẽ được cập nhật đúng trong bảng sau khi nhấn "Lưu". |  | Passed |  |
| TC36 | Hệ thống thông báo | Kiểm tra thông báo xác nhận khi xóa tài khoản | 1 | Người dùng đã vào trang quản lý tài khoản và có tài khoản trong bảng | Mở trang quản lý tài khoản | 1. Nhấn vào nút "Xóa" của tài khoản. 2. Kiểm tra thông báo xác nhận có xuất hiện không. | Thông báo xác nhận sẽ xuất hiện yêu cầu người dùng xác nhận hành động xóa tài khoản. |  | Passed |  |
| TC37 | Gửi form thêm tài khoản | Kiểm tra trường "Ảnh" nhập tên file hợp lệ | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập tên file hợp lệ vào trường "Ảnh". 2. Nhấn "Thêm". 3. Kiểm tra xem ảnh có hiển thị đúng không. | Hệ thống sẽ hiển thị ảnh đúng nếu tên file hợp lệ, nếu không sẽ không hiển thị ảnh. |  | Passed |  |
| TC38 | Chọn game | Kiểm tra việc chọn game "Genshin Impact" hiển thị tài khoản đúng | 1 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhấn vào nút "Genshin Impact". 2. Kiểm tra bảng tài khoản hiển thị các tài khoản thuộc game "Genshin Impact". | Bảng tài khoản sẽ chỉ hiển thị các tài khoản thuộc game "Genshin Impact". |  | Passed |  |
| TC39 | Bảng tài khoản | Kiểm tra chức năng tìm kiếm trong bảng tài khoản | 2 | Người dùng đã có tài khoản trong bảng | Mở trang quản lý tài khoản | 1. Nhập từ khóa vào ô tìm kiếm. 2. Kiểm tra kết quả tìm kiếm có hiển thị tài khoản tương ứng với từ khóa. | Kết quả tìm kiếm sẽ hiển thị tài khoản tương ứng với từ khóa nhập vào. |  | Passed |  |
| TC40 | Bảng tài khoản | Kiểm tra xem hệ thống có phân trang khi danh sách tài khoản quá dài không | 1 | Người dùng đã có nhiều tài khoản trong bảng | Mở trang quản lý tài khoản | 1. Kiểm tra số lượng tài khoản hiển thị trên một trang. 2. Kiểm tra có nút phân trang không. | Hệ thống sẽ hiển thị phân trang và không hiển thị hết tất cả tài khoản trên một trang duy nhất. |  | Failed |  |
| TC41 | Bảng tài khoản | Kiểm tra nút "Xóa" khi tài khoản đã bị xóa không xuất hiện trong bảng | 1 | Người dùng đã xóa tài khoản trước đó | Mở trang quản lý tài khoản | 1. Xóa tài khoản trong bảng. 2. Kiểm tra tài khoản đó có còn xuất hiện trong bảng không. | Tài khoản đã bị xóa sẽ không còn xuất hiện trong bảng tài khoản. |  | Passed |  |
| TC42 | Gửi form thêm tài khoản | Kiểm tra trường "Giá tiền" nhập giá trị âm | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập giá trị âm vào trường "Giá tiền". 2. Nhấn "Thêm". | Hệ thống sẽ hiển thị thông báo lỗi yêu cầu nhập giá trị hợp lệ cho trường "Giá tiền". |  | Failed |  |
| TC43 | Gửi form thêm tài khoản | Kiểm tra trường "Username" trùng với tài khoản đã có | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập tên đăng nhập đã tồn tại vào trường "Username". 2. Nhấn "Thêm". | Hệ thống sẽ hiển thị thông báo lỗi tên đăng nhập đã tồn tại. |  | Passed |  |
| TC44 | Gửi form thêm tài khoản | Kiểm tra trường "Mô tả" có thể nhập văn bản dài | 2 | Người dùng đã vào trang quản lý tài khoản | Mở trang quản lý tài khoản | 1. Nhập mô tả dài vào trường "Mô tả". 2. Nhấn "Thêm". | Hệ thống sẽ chấp nhận mô tả dài và hiển thị đúng trong bảng tài khoản. |  | Passed |  |
| TC45 | Bảng tài khoản | Kiểm tra nút "Sửa" với tài khoản đã được thay đổi thông tin | 1 | Người dùng đã vào trang quản lý tài khoản và có tài khoản trong bảng | Mở trang quản lý tài khoản và chọn tài khoản cần sửa | 1. Nhấn vào nút "Sửa" của tài khoản. 2. Thay đổi thông tin (ví dụ: tên đăng nhập, giá tiền). 3. Nhấn "Lưu". 4. Kiểm tra thông tin trong bảng có được cập nhật đúng. | Thông tin tài khoản sẽ được cập nhật đúng trong bảng sau khi nhấn "Lưu". |  | Passed |  |
| TC46 | Hệ thống thông báo | Kiểm tra thông báo xác nhận khi xóa tài khoản | 1 | Người dùng đã vào trang quản lý tài khoản và có tài khoản trong bảng | Mở trang quản lý tài khoản | 1. Nhấn vào nút "Xóa" của tài khoản. 2. Kiểm tra thông báo xác nhận có xuất hiện không. | Thông báo xác nhận sẽ xuất hiện yêu cầu người dùng xác nhận hành động xóa tài khoản. |  | Passed |  |
| TC47 | Lịch sử giao dịch | Kiểm tra hiển thị đúng lịch sử giao dịch | 1 | Người dùng chưa đăng | Mở trang lịch sử giao dịch | 1. Mở web. 2. Click trang "Lịch sử giao dịch". 3. Kiểm tra xem chuyển về trang đăng nhập không. | Chuyển sang trang đăng nhập. |  | Failed |  |
| TC48 | Lịch sử giao dịch | Kiểm tra nút "Hiện" mật khẩu trong lịch sử giao dịch | 1 | Người dùng đã đăng nhập và có giao dịch | Mở trang lịch sử giao dịch | 1. Vào trang "Lịch sử giao dịch". 2. Nhấn nút "Hiện" mật khẩu. 3. Kiểm tra xem mật khẩu có hiển thị không. | Mật khẩu sẽ hiển thị đúng khi nhấn nút "Hiện". |  | Passed |  |
| TC49 | Lịch sử giao dịch | Kiểm tra dữ liệu trong bảng giao dịch | 1 | Người dùng đã đăng nhập và có lịch sử giao dịch | Mở trang lịch sử giao dịch | 1. Vào trang "Lịch sử giao dịch". 2. Kiểm tra bảng dữ liệu có đầy đủ các trường ID, Tài khoản, Mật khẩu, Ngày giao dịch. | Bảng giao dịch sẽ hiển thị đầy đủ các thông tin về giao dịch như ID, tài khoản, mật khẩu, ngày giao dịch mà người đã mua |  | Failed |  |
| TC50 | Lịch sử giao dịch | Kiểm tra chức năng quay lại trang chủ | 1 | Người dùng đã đăng nhập và có giao dịch | Mở trang lịch sử giao dịch | 1. Vào trang "Lịch sử giao dịch". 2. Nhấn vào nút "Quay về trang chủ". 3. Kiểm tra xem có quay về trang chủ thành công không. | Hệ thống sẽ quay lại trang chủ thành công. |  | Passed |  |

## 4.3 BUGSUMARY:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Bug Summary** | **Clear & Understandable** | **Valid Bug** | **Good Writing** | **Assign Member** | **Review Member** | **Note** |
| 1 | Clear" Button Only Clears One Field at a Time | yes | yes | yes | Hồ Đại Minh | Phan Thị Hằng | Button should clear all fields at once |
| 2 | System Allows Negative Amount Transfers | yes | yes | yes | Hồ Đại Minh | Phan Thị Hằng | System should prevent negative transactions |
| 3 | Transfer Allows Overdraft Without Restriction | yes | yes | yes | Hồ Đại Minh | Phan Thị Hằng | System should prevent overdraft transactions |
| 4 | Combo Box 'Confirm by Letter' Not Working | yes | yes | yes | Phan Thị Hằng | Hồ Đại Minh | Combo box functionality needs fixing |
| 5 | Missing Login Button | yes | yes | yes | Phan Thị Hằng | Hồ Đại Minh | UI needs fixing to include a Login button |
| 6 | System Allows Account Creation with Negative Balance | yes | yes | yes | Phan Thị Hằng | Hồ Đại Minh | System should prevent negative balances at account creation |

# CHƯƠNG 5: KẾT LUẬN

## 5.1 Kết luận:

### *5.1.1 Kết quả đạt được:*

- Trình bày được các kiến thức cơ bản về kiểm thử phần mềm nói chung và kiểm

thử phần mềm tự động cho các ứng dụng Web nói riêng.

- Hiểu được cách cài đặt, thiết lập và sử dụng công cụ kiểm thử tự động.

- Áp dụng các kiến thức đã tìm hiểu vào thực hiện kiểm thử tự động các chức

năng chính của Website xem phim.

### *5.1.2 Hạn chế:*

- Phần mềm chưa đạt trạng thái tốt nhất.

- Chỉ mới kiểm thử được một vài ứng dụng nhỏ.

- Chưa nghiên cứu được nhiều phương pháp kiểm thử khác.

- Về phương diện chung, phần mềm có thể đi vào hoạt động nhưng cần bổ sung thêm ràng buộc và các tính năng yêu cầu. m thử tự động để có thể vận dụng kiểm thử các ứng dụng lớn hơn trong thực tiễn.

## 5.2 Nhận xét

Kiểm thử phần mềm là một hoạt động quan trọng trong đảm bảo chất lượng phần mềm để đảm bảo độ tin cậy và chất lượng của phần mềm. Mục đích chính của kiểm thử là phát hiện ra các lỗi phần mềm để từ đó khắc phục và sửa chữa. Việc kiểm thử không thể khẳng định được rằng các chức năng của sản phẩm đúng trong mọi điều kiện, mà chỉ có thể khẳng định rằng nó không hoạt động đúng trong những điều kiện cụ thể.

Trong thời gian tìm hiểu về kiểm thử phần mềm nói chung và công cụ kiểm thử nói riêng nhóm em đã hiểu được kiểm thử phần mềm là giai đoạn rất quan trọng trong quy trình sản xuất phần mềm.

# TÀI LIỆU THAM KHẢO

[1] <https://github.com/microsoft/playwright>

[2] https://playwright.dev/docs/api/class-playwright

[3] https://www.zaproxy.org/docs/

[4] https://cheatsheetseries.owasp.org

[5] [https://community.smartbear.c](https://www.youtube.com/watch?v=Dre2iVOCqhY)om

[6] [https://smartbear.com/academy](https://www.youtube.com/watch?v=Dre2iVOCqhY)/

[7] https://github.com/SeleniumHQ/selenium

[8] [https://www.selenium.dev/documentation/webdriver](https://www.youtube.com/watch?v=Dre2iVOCqhY)/

[9] <https://github.com/LambdaTest>

[10] [https://www.lambdatest.com/learning-hub](https://github.com/LambdaTest)/