

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH CÔNG NGHỆ THÔNG TIN
KIỂM THỬ BẰNG CÔNG CỤ SELENIUM VÀ ỨNG DỤNG KIỂM
THỬ WEBSITE BÁN KHÓA HỌC ONLINE

CBHD: TS.Nguyễn Mạnh Cường

Sinh viên: Phạm Thị Hạnh

Mã số sinh viên: 2020606268

Lớp: CNTT05 – K15

Hà Nội – 2024

MỤC LỤC

MỤC LỤC.....	1
DANH MỤC HÌNH ẢNH	4
DANH MỤC BẢNG BIỂU	5
LỜI CẢM ƠN	6
MỞ ĐẦU.....	7
CHƯƠNG 1.TỔNG QUAN VỀ KIỂM THỬ.....	9
1.1. Khái niệm cơ bản về kiểm thử phần mềm	9
1.1.1 Kiểm thử phần mềm là gì?.....	9
1.1.2 Lợi ích của Kiểm thử phần mềm là gì?	10
1.1.3 Các loại kiểm thử phần mềm	11
1.1.3.1 Kiểm thử chức năng (Functional testing)	11
1.1.3.2 Kiểm thử phi chức năng (Non-functional testing)	13
1.1.3.3. Kiểm thử cấu trúc (Structural testing)	14
1.1.3.4 Kiểm thử liên quan đến các thay đổi (Change related testing)	15
1.1.4 Quy trình kiểm thử phần mềm.....	16
1.1.4.1 Lập kế hoạch và kiểm soát	17
1.1.4.2 Phân tích và thiết kế.....	18
1.1.4.3 Thực hiện kiểm thử.....	18
1.1.4.4 . Đánh giá tiêu chí hoàn thành và báo cáo	19
1.1.4.5 Hoàn tất kiểm thử	19
1.1.5 Các phương pháp kiểm thử.....	20
1.1.5.1 Kiểm thử hộp trắng (White box testing).....	20

1.1.5.2 Kiểm thử hộp đen (Black box testing)	21
1.1.5.3 Kiểm thử hộp xám (Gray box testing).....	22
1.1.6 Các kỹ thuật viết test case.....	23
1.1.6.1 Phân vùng tương đương (Equivalence Class Partitioning)	23
1.1.6.2 Phân tích giá trị biên (Boundary Value Analysis).....	25
1.1.6.3 Bảng quyết định (Decision Table).....	27
1.1.6.4 Đoán lỗi (Error Guessing)	28
1.2 Phân loại kiểm thử	29
1.2.1 Kiểm thử thủ công	29
1.2.2 Kiểm thử tự động	29
1.2.3 Sự khác biệt kiểm thử thủ công và kiểm thử tự động.....	30
1.2.4 Một số công cụ kiểm thử tự động	31
CHƯƠNG 2. CÔNG CỤ KIỂM THỬ SELENIUM	38
2.1 Công cụ Selenium WebDriver	38
2.1.1 Khái niệm.....	38
2.1.2 Ưu và nhược điểm của Selenium WebDriver.....	38
2.1.3 Cách lấy xpath của element trên website.....	40
2.1.4 Một vài khái niệm cần biết trong Selenium WebDriver.....	41
2.2 Ngôn ngữ lập trình Java.....	43
2.2.1 Khái niệm.....	43
2.2.2 Mục đích sử dụng Java	43
2.2.3 Đặc điểm cơ bản của Java.....	44
2.2.4 Các thành phần của Java SE Platform	46

2.3 Nền tảng lập trình JUnit.....	46
2.3.1 Khái niệm.....	46
2.3.2 Vai trò của việc ứng dụng JUnit	46
2.3.3 Những thuật ngữ cần biết khi áp dụng JUnit	47
2.3.4 Giới thiệu các phiên bản của JUnit.....	49
2.4 Môi trường phát triển tích hợp Eclipse	50
2.4.1 Khái niệm.....	50
2.4.2 Ưu và nhược điểm của Eclipse	50
CHƯƠNG 3. ỨNG DỤNG CÔNG CỤ KIỂM THỬ CHO PHẦN MỀM BÁN	
KHÓA HỌC ONLINE.....	52
3.1 Tổng quan về hệ thống kiểm thử	52
3.1.1 Giới thiệu chung	52
3.1.2 Đặc tả use case	53
3.2 Thiết kế các ca kiểm thử	61
3.3 Thực hiện kiểm thử.....	89
3.4 Tổng hợp kết quả kiểm thử	96
3.5 Đánh giá thực nghiệm.....	97
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	99
TÀI LIỆU THAM KHẢO.....	101

DANH MỤC HÌNH ẢNH

Hình 1.1: Kiểm thử phần mềm [1]	10
Hình 1.2: Quy trình kiểm thử phần mềm [1]	17
Hình 2.1: Đường dẫn của Xpath [1].....	40
Hình 2.2: Quy trình xử lý dữ liệu trong Selenium WebDriver [1]	41
Hình 3.1: Màn hình tạo Maven project.....	90
Hình 3.2: Màn hình Class Constant	90
Hình 3.3: Màn hình Class Helper.....	91
Hình 3.4:Màn hình Class CreateAccountScreen trong package Screen.....	92
Hình 3.5: Màn hình Class CreateAccount trong Package TestCase.....	93
Hình 3.6: Màn hình trỏ nút chạy chương trình trên Eclipse	94
Hình 3.7: Màn hình khi Selenium WebDriver chạy ca kiểm thử	94
Hình 3.8: Màn hình kết quả sau khi chạy kiểm thử xong.....	95

DANH MỤC BẢNG BIỂU

Bảng 1.1: Bảng quyết định chức năng Đăng nhập.....	28
Bảng 3.1: Đặc tả use case “Tạo tài khoản”	53
Bảng 3.2: Đặc tả use case “Đăng nhập”	55
Bảng 3.3: Đặc tả use case “Tìm kiếm thông tin”	56
Bảng 3.4: Đặc tả use case “Quản lý giỏ hàng”	57
Bảng 3.5: Đặc tả use case “Đăng ký khóa học”	58
Bảng 3.6: Đặc tả use case “Cập nhật thông tin cá nhân”	59
Bảng 3.7: Đặc tả use case “Đăng xuất”	61
Bảng 3.8: Bảng các ca kiểm thử cho “Tạo tài khoản”	62
Bảng 3.9: Bảng các ca kiểm thử cho “Đăng nhập”	71
Bảng 3.10: Bảng các ca kiểm thử cho “Tìm kiếm thông tin”	73
Bảng 3.11: Bảng các ca kiểm thử cho “Quản lý giỏ hàng”	75
Bảng 3.12: Bảng các ca kiểm thử cho “Đăng ký khóa học”	76
Bảng 3.13: Bảng các ca kiểm thử cho “Cập nhật thông tin cá nhân”	82
Bảng 3.14: Bảng các ca kiểm thử cho “Đăng xuất”	89
Bảng 3.15: Bảng kết quả kiểm thử thủ công	96
Bảng 3.16: Bảng kết quả kiểm thử tự động	96
Bảng 3.17: Bảng so sánh thời gian thực hiện kiểm thử thủ công và tự động	97
Bảng 3.18: Bảng so sánh ưu nhược điểm của kiểm thử thủ công và tự động ...	98

LỜI CẢM ƠN

Em xin chân thành gửi lời cảm ơn đến các thầy cô khoa Công nghệ thông tin trường Đại học Công nghiệp Hà Nội đã tận tình hướng dẫn, hỗ trợ em trong thời gian học tập tại trường, giúp em hoàn thành tốt đồ án tốt nghiệp.

Em xin gửi lời cảm ơn đến thầy Nguyễn Mạnh Cường một người giảng viên tuyệt vời, thầy đã giúp đỡ chỉ bảo tận tình em trong suốt quá trình làm đồ án tốt nghiệp. Giúp em có hướng đi đúng và những ý tưởng mới.

Mặc dù đã cố gắng với tất cả nỗ lực của bản thân để hoàn thiện đồ án, nhưng do thời gian có hạn, năng lực và kinh nghiệm còn hạn chế nên đồ án không thể tránh khỏi những thiếu sót. Kính mong nhận được sự đóng góp ý kiến từ phía thầy cô, bạn bè để em có thể nâng cao kiến thức của bản thân, hoàn thiện đồ án được tốt hơn.

Sau cùng, em xin kính chúc các thầy cô trong khoa Công nghệ thông tin trường Đại học Công nghiệp Hà Nội dồi dào sức khỏe, niềm tin để tiếp tục thực hiện sứ mệnh cao đẹp của mình là truyền đạt kiến thức cho thế hệ mai sau.

Em xin trân trọng cảm ơn!

Hà Nội, ngày tháng năm 2024

Sinh viên thực hiện

Phạm Thị Hạnh

MỞ ĐẦU

Trong thời đại hiện nay, việc tự động hóa đang trở thành một xu hướng quan trọng và được áp dụng rộng rãi trong nhiều lĩnh vực khác nhau, đặc biệt là trong lĩnh vực công nghệ phần mềm và kiểm thử phần mềm. Sự kết hợp giữa kiểm thử thủ công và các công cụ tự động hóa không chỉ mang lại hiệu quả đáng kinh ngạc mà còn tạo ra sự hoàn hảo hơn đáng kể. Điều này không chỉ giúp tăng cường chất lượng sản phẩm mà còn giúp tiết kiệm thời gian và chi phí trong quá trình phát triển và kiểm thử phần mềm.

Hiện nay, với sự phát triển của công nghệ, các công cụ và phần mềm kiểm thử tự động đã trở nên ngày càng mạnh mẽ và linh hoạt hơn, giúp tạo ra những kịch bản kiểm thử phức tạp và tự động hóa quy trình kiểm thử một cách hiệu quả. Những công cụ như Selenium đã trở thành một trong những lựa chọn phổ biến nhất cho việc kiểm thử tự động các ứng dụng web, với khả năng tương tác và kiểm thử trên nhiều trình duyệt khác nhau một cách linh hoạt.

Trong bối cảnh này, nhu cầu tìm hiểu và áp dụng các công cụ và phương pháp kiểm thử tự động ngày càng trở nên quan trọng. Việc nắm vững quy trình kiểm thử phần mềm và tiếp cận các công cụ hỗ trợ kiểm thử sẽ giúp các nhà phát triển và kiểm thử nắm bắt được những lợi ích to lớn từ việc tự động hóa quy trình kiểm thử. Đồng thời, việc này cũng đòi hỏi sự rèn luyện và phát triển kỹ năng cho các nhà phát triển và kiểm thử để có thể sử dụng hiệu quả các công cụ và phương pháp kiểm thử tự động.

Dựa trên nhận thức này, việc lựa chọn đề tài “*Kiểm thử bằng công cụ Selenium và ứng dụng kiểm thử website bán khoá học online*” là một bước đi hợp lý và cần thiết. Báo cáo này sẽ tập trung vào việc nghiên cứu, áp dụng và đánh giá hiệu quả của việc sử dụng công cụ Selenium trong quá trình kiểm thử một

website bán khoá học online. Với nội dung được phân chia rõ ràng thành ba chương:

- Chương 1 : Tổng quan về kiểm thử phần mềm. Nơi sẽ trình bày về các phương pháp kiểm thử và sự quan trọng của việc tự động hóa quy trình kiểm thử.

- Chương 2 : Công cụ kiểm thử Selenium. Nơi sẽ đào sâu vào việc tìm hiểu và lựa chọn các công nghệ, đặc biệt là công cụ Selenium, để áp dụng vào quá trình kiểm thử.

- Chương 3 : Ứng dụng công cụ kiểm thử cho phần mềm bán khóa học online. Nơi sẽ trình bày về quá trình thực hiện kiểm thử tự động bằng Selenium và đánh giá kết quả thu được.

Qua việc tập trung nghiên cứu và thực hiện đề tài này, em đã thu được nhiều kinh nghiệm quý báu và nhận ra sự quan trọng của việc áp dụng tự động hóa trong quy trình kiểm thử phần mềm. Báo cáo này không chỉ là kết quả của sự nỗ lực cá nhân mà còn là một nguồn thông tin hữu ích cho cộng đồng chuyên ngành.

CHƯƠNG 1. TỔNG QUAN VỀ KIỂM THỬ

1.1. Khái niệm cơ bản về kiểm thử phần mềm

1.1.1 Kiểm thử phần mềm là gì?

Kiểm thử phần mềm là phương pháp kiểm tra xem sản phẩm phần mềm đó trên thực tế có phù hợp với các yêu cầu đã đặt ra hay không, và đảm bảo rằng không có lỗi hay khiếm khuyết. Nó bao gồm việc kiểm tra, phân tích, quan sát và đánh giá các khía cạnh khác nhau của sản phẩm. Người kiểm thử phần mềm (Tester) sử dụng kết hợp các công cụ thủ công và tự động. Sau khi tiến hành kiểm thử, Tester báo cáo kết quả cho team phát triển. Mục đích là xác định các lỗi, khiếm khuyết hoặc các yêu cầu còn thiếu so với yêu cầu thực tế.

Cần hiểu được tầm quan trọng của việc kiểm thử đối với mỗi công ty phát triển phần mềm. Với kiểm thử phần mềm, nếu có bất kỳ lỗi nào, nó có thể được xác định sớm và giải quyết trước khi giao sản phẩm.

Nhiều công ty phát triển phần mềm thường bỏ qua bước này vì ngân sách eo hẹp và cho rằng nó sẽ không dẫn đến hậu quả lớn. Nhưng để tạo những trải nghiệm tốt nhất cho khách hàng, chất lượng sản phẩm cần phải được đặt lên hàng đầu. Và vì vậy, việc kiểm thử sản phẩm để tìm lỗi là điều gần như bắt buộc. Doanh nghiệp chỉ có thể mang đến giá trị cho khách hàng khi sản phẩm cung cấp được coi là lý tưởng. Và để đạt được điều đó, các công ty phải đảm bảo rằng người dùng không gặp phải bất kỳ vấn đề nào khi sử dụng sản phẩm của mình. Cách tốt nhất để làm điều đó là tạo ra sản phẩm không có lỗi.



Hình 1.1: Kiểm thử phần mềm [1]

Vai trò của kiểm thử phần mềm rất quan trọng trong quá trình phát triển.

Thêm nữa, khi khách hàng sử dụng sản phẩm, họ rất có thể phải tiết lộ một số thông tin cá nhân. Để ngăn chặn tin tặc nắm được dữ liệu này, việc kiểm tra bảo mật là điều bắt buộc trước khi phần mềm đến tay người dùng. Sản phẩm phần mềm được kiểm thử kỹ càng qua quy trình phù hợp sẽ đảm bảo độ tin cậy, bảo mật, giúp tiết kiệm thời gian, chi phí, mang đến sự hài lòng cho khách hàng.

Một lý do nữa khiến việc kiểm thử ngày càng trở nên quan trọng đó là phát hiện khả năng tương thích với các thiết bị và nền tảng khác nhau. Giả sử khi phát triển một trang web, Tester phải kiểm tra xem trang web có chạy trên độ phân giải thiết bị khác nhau, các trình duyệt khác nhau hay không? Những gì hoạt động tốt trên Chrome có thể không chạy tốt trên Safari hoặc Internet Explorer. Điều này làm phát sinh nhu cầu kiểm tra trình duyệt chéo, bao gồm kiểm tra tính tương thích của ứng dụng trên các trình duyệt khác nhau.

1.1.2 Lợi ích của Kiểm thử phần mềm là gì?

- Hiệu quả về chi phí: Đây là một trong những lợi ích quan trọng của kiểm thử phần mềm. Thực tế cho thấy rằng các lỗi thiết kế khó có thể được loại trừ

hoàn toàn đối với bất kỳ hệ thống nào. Đó không phải là lỗi bất cẩn của Developer mà đôi khi do sự phức tạp của hệ thống. Nếu các vấn đề về thiết kế không được phát hiện, thì việc tìm ra và sửa các lỗi/khiếm khuyết sẽ trở nên khó khăn và tốn kém hơn. Kiểm thử bất kỳ dự án IT nào cũng sẽ giúp công ty tiết kiệm, việc xác định lỗi trong giai đoạn đầu sẽ giúp quá trình sửa chữa tốn ít chi phí hơn.

- Bảo mật: Đây là điểm nhạy cảm và dễ bị tấn công nhất của kiểm thử phần mềm. Kiểm thử giúp loại bỏ các rủi ro và vấn đề trong sản phẩm. Cùng với đó, tất cả khách hàng đều đang tìm kiếm những sản phẩm đáng tin cậy.

- Chất lượng sản phẩm: Đây là yêu cầu thiết yếu của bất kỳ sản phẩm phần mềm nào. Kiểm thử phần mềm giống như việc củng cố danh tiếng công ty bằng cách cung cấp các sản phẩm chất lượng cho khách hàng.

- Sự hài lòng của khách hàng: Trong bất kỳ hoạt động kinh doanh sản phẩm nào, mục tiêu cuối cùng đều là mang đến cho khách hàng trải nghiệm tốt nhất. Sự hài lòng của khách hàng rất quan trọng trong quá trình hợp tác lâu dài.

1.1.3 Các loại kiểm thử phần mềm

Kiểm thử phần mềm không phải là một việc đơn lẻ. Nó có nhiều hình thức khác nhau và được phân loại theo một số tiêu chí. Về cơ bản, kiểm thử phần mềm được chia làm 4 loại:

1.1.3.1 Kiểm thử chức năng (Functional testing)

Kiểm thử chức năng là xác minh hệ thống hoạt động theo đúng theo các yêu cầu nghiệp vụ. Hình thức kiểm thử này có thể được thực hiện từ hai khía cạnh: dựa trên yêu cầu (requirements-based) và dựa trên quy trình nghiệp vụ (business – process – based).

Trong kiểm thử dựa trên yêu cầu, các yêu cầu được ưu tiên tùy thuộc vào tiêu chí rủi ro. Điều này sẽ đảm bảo những phần quan trọng nhất sẽ được test đầy đủ. Mặt khác, kiểm thử dựa trên quy trình nghiệp vụ sẽ sử dụng những kiến thức tương ứng. Quy trình nghiệp vụ mô tả các việc liên quan đến nghiệp vụ hằng ngày của hệ thống.

Kiểm thử chức năng bao gồm 5 bước:

- Xác định các chức năng mà phần mềm sẽ thực hiện.
- Tạo các dữ liệu đầu vào dựa trên các tài liệu đặc tả kỹ thuật của các chức năng.
- Xác định các kết quả đầu ra dựa trên các tài liệu đặc tả kỹ thuật của các chức năng.
- Thực hiện các trường hợp kiểm thử.
- So sánh kết quả thực tế và kết quả mong muốn.

Trong đó, kiểm thử chức năng còn được chia nhỏ ra thành các loại:

- Kiểm thử đơn vị (Unit testing)
- Smoke Testing
- Sanity Testing
- Kiểm thử giao diện (Interface testing)
- Kiểm thử tích hợp (Integration testing)
- Kiểm thử hệ thống (System testing)
- Kiểm thử hồi quy (Regression testing)

- Kiểm thử chấp nhận (Acceptance testing)

Ưu điểm của kiểm thử chức năng:

- Hình thức kiểm thử này mô phỏng việc sử dụng hệ thống thực tế
- Được thực hiện trong các điều kiện gần với điều kiện của khách hàng
- Không có giả định nào về cấu trúc hệ thống được đưa ra trong khi kiểm thử chức năng
- Rất dễ dàng để thực hiện test thủ công

Ngược lại, kiểm thử chức năng có những giới hạn sau:

- Khả năng cao xảy ra tình trạng test dư thừa
- Các lỗi logic trong phần mềm có thể bị bỏ sót trong khi kiểm thử chức năng

Kiểm thử phần mềm có nhiều hình thức khác nhau và được phân loại theo một số tiêu chí.

1.1.3.2 Kiểm thử phi chức năng (Non-functional testing)

Kiểm thử phi chức năng là kiểm tra các đặc tính chất lượng của hệ thống. Ví dụ, kiểm tra xem bao nhiêu người có thể đăng nhập đồng thời vào một phần mềm. Kiểm tra phi chức năng cũng quan trọng không kém như kiểm tra chức năng và ảnh hưởng đến sự hài lòng của khách hàng.

Tương tự, kiểm thử phi chức năng cũng được chia thành các loại:

- Kiểm thử độ ổn định (Stability testing): đánh giá phần mềm có thể liên tục hoạt động tốt trong hoặc ngay trên khoảng thời gian có thể chấp nhận hay không

- Kiểm thử khả năng chịu tải (Load testing): đánh giá hoạt động của hệ thống khi khối lượng công việc ngày càng tăng
- Kiểm thử áp lực (Stress testing): ước tính hoạt động của hệ thống ở trong hoặc vượt quá giới hạn khối lượng công việc dự kiến
- Kiểm thử tính khả dụng (Usability testing): sản phẩm được test về tính thân thiện với người dùng
- Kiểm thử bảo trì (Maintainability testing): kiểm tra mức độ đánh giá, thay đổi và test sản phẩm
- Kiểm thử độ tin cậy (Reliability testing): sử dụng công cụ để tìm, ngăn chặn và loại bỏ lỗi trước khi hệ thống được triển khai
- Kiểm thử tính tương thích (Portability testing): xác định mức độ dễ dàng hoặc khó khăn mà phần mềm có thể di chuyển từ môi trường này sang môi trường khác

1.1.3.3. Kiểm thử cấu trúc (Structural testing)

Kiểm thử cấu trúc thường được gọi là “hộp trắng” hoặc “hộp thủy tinh” bởi vì phương pháp này quan tâm đến việc tìm kiếm những gì đang xảy ra bên trong, kiểm tra dựa trên phân tích cấu trúc bên trong của thành phần hoặc hệ thống. Nó thường được sử dụng như một cách đo lường của kiểm thử, thông qua độ bao phủ của một tập hợp các yếu tố cấu trúc. Kiểm thử cấu trúc chủ yếu được áp dụng ở kiểm thử thành phần, kiểm thử tích hợp.

Các mục tiêu chính của kiểm thử cấu trúc bao gồm:

- Nhận ra những điểm bất cập
- Test chức năng bổ sung

- Xác định những phần bị thiếu trong bộ kiểm thử

Ưu điểm của kiểm thử cấu trúc:

- Loại bỏ code chết
- Có khả năng tìm ra lỗi ở giai đoạn đầu
- Đảm bảo kiểm tra phần mềm kỹ lưỡng hơn
- Tiết kiệm thời gian

Bên cạnh đó, nhược điểm của kiểm thử cấu trúc:

- Kiểm tra kết cấu khá tốn kém
- Yêu cầu kiến thức về code
- Đòi hỏi kiến thức vững chắc về công cụ được sử dụng để test

Kiểm thử cấu trúc đảm bảo kiểm tra kỹ lưỡng hơn và tiết kiệm thời gian.

1.1.3.4 Kiểm thử liên quan đến các thay đổi (Change related testing)

❖ Kiểm thử xác nhận (Confirmation testing)

Khi kiểm thử gặp lỗi, Tester phải xác định nguyên nhân lỗi là do lỗi phần mềm. Sau khi Tester phát hiện lỗi và báo cho Developer để sửa thì phần mềm sau đó sẽ cập nhật phiên bản vá lỗi. Cuối cùng, Tester cần thực hiện kiểm tra thêm một lần nữa để xác định rằng lỗi thực sự đã được giải quyết.

Khi thực hiện kiểm tra xác nhận, điều quan trọng nhất là phải đảm bảo rằng các trường hợp kiểm thử phải được thực hiện chính xác giống như lần đầu tiên, sử dụng cùng một đầu vào, dữ liệu và môi trường kiểm thử để đảm bảo rằng các lỗi đã được sửa. Tester cần phải biết rằng trong lần kiểm thử sau khi vá lỗi khả năng sinh ra lỗi khác trong phần mềm là điều hoàn toàn có thể xảy ra. Vì vậy

kiểm thử chính xác ở phiên bản hiện tại của phần mềm là chưa đủ. Cách phát hiện các điểm ngoài ý muốn của việc kiểm lỗi là thực hiện kiểm thử hồi quy.

❖ Kiểm thử hồi quy (Regression testing)

Tương tự như kiểm thử xác nhận thì kiểm thử hồi quy liên quan đến việc lặp lại các trường hợp kiểm thử đã được thực hiện trước đó. Kiểm thử hồi quy được thực hiện khi phần mềm thay đổi do sửa lỗi, chức năng mới.

Mục đích của kiểm thử hồi quy để xác minh rằng các sửa đổi trong phần mềm hoặc môi trường không gây ra bất lợi ngoài ý muốn, ảnh hưởng hoặc làm hư các chức năng và hệ thống vẫn đáp ứng các yêu cầu của phần mềm. Tất cả các trường hợp trong quá trình kiểm thử hồi quy sẽ được thực hiện mỗi khi một phiên bản vá lỗi của phần mềm được release, và điều này khiến chúng trở nên lý tưởng cho tự động hóa.

1.1.4 Quy trình kiểm thử phần mềm

Vậy là chúng ta đã tìm hiểu xong Kiểm thử phần mềm là gì, lợi ích của nó và các loại kiểm thử. Tiếp theo, hãy cùng đi đến quy trình kiểm thử nhé. Có rất nhiều quy trình khác nhau như: mô hình chữ V, mô hình thác nước, mô hình xoắn ốc, ... hoặc có thể là mô hình kết hợp những mô hình trên.

Tùy thuộc vào quy mô của phần mềm và tính cấp thiết của dự án, quy trình kiểm thử sẽ khác nhau. Tuy nhiên, trong đồ án này sẽ giới thiệu quy trình kiểm thử với 5 bước, bao gồm công việc lập kế hoạch và công việc sau đánh giá như sau:



Hình 1.2: Quy trình kiểm thử phần mềm [1]

Quy trình kiểm thử phần mềm được chia thành 5 bước :

1.1.4.1 Lập kế hoạch và kiểm soát

Lập kế hoạch kiểm thử là việc tạo ra một tài liệu mô tả tiếp cận tổng thể và các mục tiêu cần test. Bao gồm xem xét cơ sở test, xác định các điều kiện dựa trên phân tích các mục thử test, viết các trường hợp và thiết kế môi trường test. Tiêu chí hoàn thành được chỉ định để biết khi nào việc kiểm thử hoàn tất (ở bất kỳ giai đoạn nào).

Kiểm soát là hoạt động so sánh tiến độ thực tế so với kế hoạch và báo cáo tình trạng, bao gồm cả những sai lệch so với kế hoạch. Nó liên quan đến việc thực hiện các hành động cần thiết để đáp ứng mục tiêu của dự án.

Mục đích của bước này là:

- Xác định phạm vi, rủi ro và các mục tiêu test
- Xác định các tài nguyên test cần thiết như con người, môi trường, ...
- Lên lịch trình cho các nhiệm vụ phân tích và thiết kế, thực hiện, và đánh giá test

1.1.4.2 Phân tích và thiết kế

Phân tích và thiết kế kiểm thử có các nhiệm vụ chính sau:

- Xem xét cơ sở test – thông tin dựa trên các trường hợp test, chẳng hạn như yêu cầu, đặc điểm thiết kế, phân tích rủi ro, kiến trúc và giao diện
- Xác định các điều kiện test
- Thiết kế các bài test
- Thiết kế môi trường thử test, thiết lập và xác định cơ sở hạ tầng và công cụ cần thiết

1.1.4.3 Thực hiện kiểm thử

Thực hiện kiểm thử là việc test chỉ định trên hệ thống máy tính theo cách thủ công hoặc sử dụng công cụ test tự động. Việc triển khai test có nhiệm vụ chính sau:

- Tiến hành các trường hợp test bằng cách sử dụng các kỹ thuật và tạo dữ liệu cho các thử nghiệm đó
- Tạo các bộ kiểm thử từ các trường hợp test để thực hiện hiệu quả. Bộ kiểm thử là tập hợp các trường hợp test được sử dụng để kiểm thử phần mềm
- Thực hiện lại các trường hợp test không thành công trước đó để xác nhận bản sửa lỗi
- Ghi lại kết quả của việc thực hiện test. Ở đó nhật ký kiểm thử ghi lại trạng thái của trường hợp test. (đạt/ không đạt)
- So sánh kết quả thực tế với kết quả mong đợi

1.1.4.4 . Đánh giá tiêu chí hoàn thành và báo cáo

Đánh giá tiêu chí hoàn thành là quá trình xác định thời điểm dừng kiểm thử. Nó phụ thuộc vào phạm vi của mã code, chức năng hoặc rủi ro. Ngoài ra cũng phụ thuộc vào rủi ro business, chi phí, thời gian và sự khác nhau giữa các dự án. Đánh giá tiêu chí hoàn thành có các nhiệm vụ chính sau:

- Đánh giá xem có cần test thêm hoặc tiêu chí hoàn thành đã chỉ định có cần thay đổi hay không
- Viết một báo cáo tóm tắt kiểm thử cho các bên liên quan

1.1.4.5 Hoàn tất kiểm thử

Quy trình hoàn tất kiểm thử được thực hiện khi phần mềm sẵn sàng được bàn giao. Ngoài ra, kiểm thử có thể bị dừng lại vì các lý do khác như:

- Khi dự án bị hủy bỏ
- Khi đạt được một số mục tiêu
- Khi bản cập nhật hoặc release bảo trì hoàn thành
 - Bước này có các nhiệm vụ chính sau:
 - Kiểm tra xe sản phẩm được bàn giao chưa, theo kế hoạch nào, và để đảm bảo rằng tất cả các báo cáo sự cố đã được giải quyết
 - Hoàn thiện và lưu trữ phần mềm kiểm thử như scripts, môi trường test, ... để sử dụng lại sau này
 - Bàn giao phần mềm kiểm thử cho bên bảo trì
 - Đánh giá cách test đã thực hiện và rút kinh nghiệm cho các bản release và dự án trong tương lai

Có thể rất khó để thực hiện mọi thứ trong quá trình từ đầu đến cuối một cách hoàn hảo, từ lập kế hoạch đến thực hiện và hậu kỳ test. Tuy nhiên, việc xác định quy trình và cải thiện test là một bước rất quan trọng để kiểm soát chất lượng sản phẩm. Kiểm thử phần mềm sẽ trở nên đơn giản hơn nếu hiểu mục đích, nắm rõ các bước của quy trình và tuân theo.

1.1.5 Các phương pháp kiểm thử

1.1.5.1 Kiểm thử hộp trắng (White box testing)

Kiểm thử hộp trắng là một phương pháp kiểm thử phần mềm trong đó kiểm thử viên có kiến thức về cấu trúc, nội bộ và mã nguồn của phần mềm. Kiểm thử hộp trắng được thực hiện bằng cách kiểm tra mã nguồn của phần mềm, các biến và hàm, các điều kiện và vòng lặp để đảm bảo rằng các thành phần của phần mềm hoạt động chính xác và hiệu quả.

Kiểm thử hộp trắng được chia thành nhiều loại khác nhau, bao gồm:

- API Testing là kiểm thử ứng dụng bằng cách dùng hàm API private và public.
- Code Coverage là việc tạo trường hợp test nhằm thỏa mãn điều kiện bao phủ code – code coverage.
- Fault Injection Methods sẽ đưa một số lỗi vào để test đường dẫn code nhằm cải tiến bao phủ.
- Mutation Testing Methods là một trong những loại kiểm thử hộp trắng nhằm thay đổi câu lệnh trong source code, kiểm tra test case có khả năng tìm ra lỗi hay không.
- Static testing cho phép các tester không cần hiểu biết về mã lệnh mà chỉ cần căn cứ vào tài liệu đặc tả để xử lý bất cứ chức năng nào.

❖ Ưu điểm:

- Giúp hệ thống tối ưu hóa các dòng lệnh một cách đơn giản
- Giúp các tester phát hiện lỗi dễ dàng trong mỗi dòng lệnh
- Loại bỏ nhanh chóng dòng lệnh có lỗi tiềm ẩn hoặc không quan trọng
- Tester sau khi thực hiện phương pháp kiểm thử phần mềm này sẽ dễ dàng hơn để đạt được độ bao phủ lớn nhất trong các trường hợp kiểm thử sau

❖ Nhược điểm:

- Khi tìm lỗi tiềm ẩn của hệ thống, có nhiều luồng không thể kiểm tra được bằng cách kiểm tra chi tiết từng dòng lệnh
- Cần nhiều tool chuyên biệt, như tool phát hiện/sửa lỗi, phân tích code để có thể duy trì phương pháp kiểm thử hộp trắng
- Tester cần có chuyên môn cao và dày dặn kinh nghiệm
- Giá thành phần mềm sẽ tăng lên đáng kể nếu sử dụng các tester có chuyên môn cao

1.1.5.2 Kiểm thử hộp đen (Black box testing)

Kiểm thử hộp đen là một phương pháp kiểm thử phần mềm trong đó kiểm thử viên không có kiến thức về cấu trúc, nội bộ và mã nguồn của phần mềm. Thay vào đó, kiểm thử viên tập trung vào các đầu vào và đầu ra của phần mềm để đảm bảo rằng chúng hoạt động chính xác và đáp ứng các yêu cầu chức năng và phi chức năng của phần mềm.

❖ Ưu điểm:

- Tester không cần phải truy cập vào từng dòng lệnh

- Hiệu quả và phù hợp với hệ thống có số lượng lớn dòng lệnh
- Phân biệt một cách rõ ràng quan điểm của nhà phát triển và người dùng
- Không đòi hỏi tester phải có kiến thức về ngôn ngữ lập trình khi kiểm thử phần mềm

❖ **Nhược điểm:**

- Bị giới hạn bởi độ bao phủ của các trường hợp kiểm thử
- Khó khăn trong việc thiết kế đầy đủ mọi trường hợp kiểm thử
- Thực tế không mang lại hiệu quả cao bởi các tester bị giới hạn kiến thức về hệ thống
- Các tester chỉ tập trung vào dòng lệnh dễ xảy ra lỗi, khó có thể kiểm tra tất cả đoạn lệnh của hệ thống

1.1.5.3 Kiểm thử hộp xám (Gray box testing)

Kiểm thử hộp xám là phương pháp kiểm thử phần mềm kết hợp giữa kiểm thử hộp trắng và hộp đen. Trong kiểm thử hộp xám, kiểm thử viên có một số kiến thức về cấu trúc và nội bộ của phần mềm để phát hiện lỗi và kiểm tra tính toàn vẹn của phần mềm.

❖ **Ưu điểm:**

- Phương pháp kiểm thử phần mềm tối ưu bởi nó là sự kết hợp giữa 2 phương pháp còn lại
- Các tester chỉ cần dựa vào các tài liệu đặc tả chức năng, định nghĩa giao diện mà không cần dựa vào các dòng lệnh

- Với phương pháp này, kịch bản thử nghiệm có thể được thiết kế phức tạp, thông minh và hiệu quả hơn

- Việc kiểm thử sẽ được hoàn thiện từ góc nhìn của người sử dụng

❖ Nhược điểm:

- Kiểm thử hộp xám cho một ứng dụng có hệ thống phân tán thường gặp khó khăn khi liên kết lỗi

- Độ bao phủ của các trường hợp kiểm thử bị giới hạn do kiểm thử hộp xám không dựa trên việc truy cập code

- Các luồng đầu vào của hệ thống bị giới hạn về mặt thời gian kiểm thử, cho nên, có rất nhiều luồng hoạt động của hệ thống không được kiểm tra

1.1.6 Các kỹ thuật viết test case

1.1.6.1 Phân vùng tương đương (Equivalence Class Partitioning)

- Phân vùng lớp tương đương cho phép bạn phân chia tập hợp các điều kiện kiểm tra thành một phân vùng nên được coi là giống nhau.

- Phương pháp kiểm thử phần mềm này chia miền đầu vào của chương trình thành các lớp dữ liệu mà từ đó các trường hợp kiểm thử nên được thiết kế.

- Với các giá trị đầu vào chia thành các vùng tương đương:

- + Vùng tương đương hợp lệ: tập hợp các giá trị kiểm thử thỏa mãn điều kiện của hệ thống

- + Vùng tương đương không hợp lệ: Tập hợp các giá trị kiểm thử mô tả trạng thái khác của hệ thống: sai, thiếu, không đúng, ...

- 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.

- Thiết kế Test-case bằng phân lớp tương đương tiến hành theo 2 bước:

- + Xác định các lớp tương đương
- + Xác định các ca kiểm thử

- Nguyên tắc:

- + 1 lớp các giá trị lớn hơn
- + 1 lớp các giá trị nhỏ hơn
- + n lớp các giá trị hợp lệ

Ví dụ: Thiết kế testcase cho ô text chỉ cho nhập số nguyên với độ dài ký tự thuộc [1-10] hoặc [20-30]

- Với yêu cầu trên ta có các vùng:

- + nhỏ hơn 1 : vùng không hợp lệ
- + [1-10] : vùng hợp lệ
- + lớn hơn 10 và nhỏ hơn 20 : vùng không hợp lệ
- + [20-30] : vùng hợp lệ
- + lớn hơn 30: vùng không hợp lệ
- + Nhập các ký tự không phải số nguyên : vùng không hợp lệ

=> Vì vậy ta có các case sau:

- Case hợp lệ:

- + Nhập 5 ký tự
- + Nhập 25 ký tự số

- Case không hợp lệ:

- + Không nhập vào trường
- + Nhập 15 ký tự
- + Nhập số thập phân

- + Nhập 35 kí tự
- + Nhập ký tự chữ: Tiếng việt, Tiếng anh, Full-size, Half-size
- + Nhập ký tự đặc biệt, space, kí tự Enter
- + Nhập câu lệnh SQL injection, HTML, XSS

1.1.6.2 Phân tích giá trị biên (Boundary Value Analysis)

- Phân tích giá trị biên dựa trên việc kiểm thử tại các ranh giới giữa các phân vùng, Chúng ta sẽ tập trung vào các giá trị biên chứ không test toàn bộ dữ liệu. Thay vì chọn nhiều giá trị trong lớp đương tương để làm đại diện, phân tích giá trị biên yêu cầu chọn một hoặc vài giá trị là các cạnh của lớp tương đương để làm điều kiện test.

- Chúng ta thường thấy rằng một số lượng lớn lỗi xảy ra tại các ranh giới của các giá trị đầu vào được xác định thay vì các giá trị giữa, còn được gọi là các giá trị biên. Từ đó đưa ra lựa chọn các test cases thực hiện giá trị đầu vào các giá trị biên.

- Kỹ thuật thiết kế test cases này bổ sung cho phân vùng tương đương. Kỹ thuật kiểm thử phần mềm này dựa trên nguyên tắc: Nếu một hệ thống hoạt động tốt với các giá trị biên thì nó sẽ hoạt động tốt cho tất cả các giá trị nằm giữa hai giá trị biên.

- Phân tích giá trị biên sẽ chọn các giá trị:

- + Giá trị ngay dưới giá trị nhỏ nhất
- + Giá trị nhỏ nhất
- + Giá trị ngay trên giá trị nhỏ nhất
- + Giá trị ngay dưới giá trị lớn nhất
- + Giá trị lớn nhất
- + Giá trị ngay trên giá trị lớn nhất

Ví dụ: Với ví dụ trên ta có các case test sau:

- + Không nhập ký tự nào (Giá trị ngay dưới giá trị nhỏ nhất vùng 1)
- + Nhập 1 ký tự (giá trị nhỏ nhất vùng 1)
- + Nhập 2 ký tự (giá trị ngay trên giá trị nhỏ nhất vùng 1)
- + Nhập 9 ký tự (giá trị ngay dưới giá trị lớn nhất vùng 1)
- + Nhập 10 ký tự (giá trị lớn nhất vùng 1)
- + Nhập 11 ký tự (giá trị ngay trên giá trị lớn nhất vùng 1)
- + Nhập 19 ký tự (giá trị ngay dưới giá trị nhỏ nhất vùng 2)
- + Nhập 20 ký tự (giá trị nhỏ nhất vùng 2)
- + Nhập 21 ký tự (giá trị ngay trên giá trị nhỏ nhất vùng 2)
- + Nhập 29 ký tự (giá trị ngay dưới giá trị lớn nhất vùng 2)
- + Nhập 30 ký tự (giá trị lớn nhất vùng 2)
- + Nhập 31 ký tự (giá trị ngay trên giá trị lớn nhất vùng 2)

=> Kết hợp kỹ thuật phân vùng tương đương với phân tích giá trị biên ta có các case test sau:

- + Không nhập ký tự nào
- + Nhập 1 ký tự
- + Nhập 5 ký tự
- + Nhập 10 ký tự
- + Nhập 11 ký tự
- + Nhập 19 ký tự
- + Nhập 20 ký tự
- + Nhập 21 ký tự
- + Nhập 25 ký tự

- + Nhập 30 ký tự
- + Nhập 31 ký tự
- + Nhập số thập phân
- + Nhập ký tự chữ: Tiếng việt, Tiếng anh, Full-size, Half-size
- + Nhập ký tự đặc biệt, space, kí tự Enter
- + Nhập câu lệnh SQL injection, HTML, XSS

1.1.6.3 Bảng quyết định (Decision Table)

- Bảng quyết định còn được gọi là bảng Nguyên nhân – Kết quả (Cause-Effect). Kỹ thuật kiểm thử phần mềm này được sử dụng cho các chức năng cần sự kết hợp của các yếu tố đầu vào các biến.

Ví dụ: Nút Submit phải được enable nếu người dùng đã nhập tất cả các trường bắt buộc

- Đầu tiên là xác định đầu ra của các chức năng có phụ thuộc vào sự kết hợp của các đầu vào. Nếu có tập hợp kết hợp đầu vào lớn, thì hãy chia nó thành các tập hợp nhỏ hơn hữu ích cho việc quản lý bảng quyết định.

- Đối với mọi chức năng, cần tạo một bảng và liệt kê tất cả các loại kết hợp đầu vào và đầu ra tương ứng. Điều này giúp xác định các điều kiện bị tester bỏ qua.

- Các bước để tạo bảng quyết định:

- + Nhập đầu vào theo hàng.
- + Nhập tất cả các quy tắc trong cột.
- + Điền vào bảng với sự kết hợp của đầu vào.
- + Trong hàng cuối cùng, ghi chú đầu ra so với kết hợp đầu vào.

Ví dụ: Nút Submit chỉ được enable khi tất cả các đầu vào được nhập bởi người dùng cuối.

Bảng 1.1: Bảng quyết định chức năng Đăng nhập

	Role 1	Role 2	Role 3	Role 4
Email	T	T	F	F
Password	T	F	T	F
Submit	H	Error	Error	Error

Gồm 2 trường đầu vào: Email và Password

Chú thích:

- T: Email hoặc Password đúng
- F: Email hoặc Password sai
- H: Đăng nhập thành công, chuyển đến màn hình trang chủ
- Error: Hiện thị thông báo lỗi, đăng nhập không thành công

1.1.6.4 Đoán lỗi (Error Guessing)

- Đoán lỗi là một kỹ thuật kiểm thử phần mềm dựa trên việc đoán lỗi có thể chiếm ưu thế trong code. Đây là một kỹ thuật dựa trên kinh nghiệm, trong đó nhà phân tích kiểm thử sử dụng kinh nghiệm của mình để đoán phần có vấn đề hoặc có lỗi của ứng dụng kiểm thử.

- Kỹ thuật xác định danh sách các lỗi có thể xảy ra hoặc các tình huống dễ xảy ra lỗi. Sau đó, người kiểm thử viết test cases để tìm kiếm những lỗi đó. Để thiết kế các test cases dựa trên kỹ thuật kiểm thử phần mềm này, nhà phân tích có thể sử dụng các kinh nghiệm trong quá khứ để xác định các điều kiện.

- Cách đoán lỗi:

- + Tester nên sử dụng kinh nghiệm trước đây để kiểm thử các ứng dụng tương tự.
- + Hiểu biết về hệ thống đang kiểm thử.

- + Kiến thức về các lỗi thực hiện điển hình.
- + Nhớ những chức năng phức tạp trước đây.
- + Đánh giá lịch sử dữ liệu và kết quả kiểm thử.

1.2 Phân loại kiểm thử

1.2.1 Kiểm thử thủ công

Kiểm thử thủ công là kiểm thử phần mềm trong đó các kiểm thử được thực hiện thủ công bởi QA Analyst. Kiểm thử thủ công được thực hiện để tìm ra các lỗi trong phần mềm đang được phát triển.

Trong kiểm thử thủ công, tester sẽ kiểm thử tất cả các tính năng thiết yếu của ứng dụng hoặc phần mềm. Tester thực hiện các test cases và tạo báo cáo kiểm thử mà không cần sự trợ giúp của bất kỳ công cụ kiểm thử phần mềm tự động nào.

Đây là một phương pháp kiểm thử cổ điển, giúp tìm lỗi trong các hệ thống phần mềm, thường được tiến hành bởi một tester có kinh nghiệm và thực hiện theo quy trình kiểm thử phần mềm.

1.2.2 Kiểm thử tự động

Trong Kiểm thử tự động, tester viết code/ test scripts để tự động thực hiện kiểm thử. Tester sử dụng các công cụ tự động hóa phù hợp để xây dựng các kịch bản kiểm thử và xác minh phần mềm hoạt động đúng hay không. Mục tiêu là hoàn thành kiểm thử trong một khoảng thời gian ngắn hơn.

Kiểm thử tự động hoàn toàn dựa vào kiểm thử trước khi chạy tự động kịch bản để so sánh kết quả thực tế với kết quả dự kiến. Điều này giúp tester xác định xem một ứng dụng có hoạt động đúng như mong đợi hay không.

Kiểm thử tự động cho phép tester thực hiện kiểm thử lặp lại và hồi quy mà không cần sự can thiệp của người kiểm thử thủ công (Manual tester). Mặc dù tất cả các quy trình được thực hiện tự động nhưng vẫn đòi hỏi phải thực hiện kiểm thử thủ công trước khi thực hiện kiểm thử tự động để tạo các kịch bản kiểm thử ban đầu.

1.2.3 Sự khác biệt kiểm thử thủ công và kiểm thử tự động

Tiêu chí	Kiểm thử thủ công	Kiểm thử tự động
Định nghĩa	Testcase được thực hiện thủ công bởi tester	Tester phải viết test script và lựa chọn công cụ để tự động hóa việc test
Thời gian xử lý	Cần nhiều thời gian và nhân lực	Thời gian kiểm thử nhanh hơn so với manual testing
Kiểm thử khám phá	Kiểm thử khám phá được thực hiện	Không cho phép kiểm thử khám phá
Thay đổi UI	Sự thay đổi nhỏ như ID, Class hoặc 1 button nhưng không ảnh hưởng đến thực thi test	Chỉ 1 vài thay đổi nhỏ trong UI, người dùng phải update script để đảm bảo có kết quả như mong đợi
Độ tin cậy	Kết quả kiểm thử không đáng tin cậy vì có khả năng xảy ra lỗi do con người	Do được thực thi bằng tool và scripts nên kết quả đáng tin cậy hơn
Đầu tư	Cần nhiều nguồn nhân lực	Bắt buộc phải đầu tư tool để test và những kỹ sư auto

Báo cáo	Manual test thường lưu lại kết quả ở Excel, Word, ...	Tất cả stakeholders có thể đăng nhập vào hệ thống auto và kiểm tra lại kết quả test
Sự quan sát của con người	Cần có sự quan sát của con người để giúp cho hệ thống thân thiện với người dùng	Không có sự quan sát của con người
Kiểm thử hiệu năng	Không thực hiện được Kiểm thử hiệu năng	Kiểm thử hiệu năng phải được thực hiện bởi 1 tool phù hợp
Kiến thức lập trình	Không cần có khả năng code	Phải có kiến thức về lập trình để tạo ra các test script
Cách tiếp cận tốt	Manual testing hữu ích khi chúng ta chạy lại bộ testcase 1 hoặc 2 lần	Auto test rất hữu ích khi ta chạy lại bộ script nhiều lần
Sử dụng khi nào?	Kiểm thử thủ công phù hợp cho test khám phá, khả năng sử dụng và kiểm thử dựa vào thực tế	Test auto thích hợp cho kiểm thử hồi quy, hiệu năng hoặc các trường hợp có khả năng lặp lại nhiều lần

1.2.4 Một số công cụ kiểm thử tự động

❖ Selenium

Selenium là một công cụ kiểm thử phần mềm tự động mã nguồn mở miễn phí cho các ứng dụng web trên nhiều trình duyệt và nền tảng khác nhau như Windows, Mac và Linux. Selenium giúp Tester thực hiện kiểm thử bằng nhiều ngôn ngữ lập trình khác nhau như Java, PHP, C#, Python, Groovy, Ruby và Perl.

Selenium hiện có 3 loại: Selenium Webdriver, Selenium IDE, Selenium Grid. Tùy vào kỹ năng, nền tảng và yêu cầu mà bạn có thể lựa chọn sử dụng loại Selenium phù hợp.

Công cụ này phổ biến với tất cả các trình duyệt nổi tiếng hiện tại như Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari, Opera. Vì vậy, Selenium chắc chắn là nền tảng cho hầu hết các công cụ kiểm thử phần mềm khác.

❖ **TestingWhiz**

TestingWhiz là công cụ kiểm thử phần mềm tự động với phiên bản Enterprise cung cấp một gói hoàn chỉnh gồm nhiều giải pháp test tự động khác nhau. Trong đó bao gồm: test web, test phần mềm, test database (cơ sở dữ liệu), test API, test ứng dụng di động, bảo trì bộ kiểm tra hồi quy, tối ưu hóa và tự động hóa cũng như kiểm thử trên nhiều trình duyệt.

Ngoài ra, TestingWhiz cung cấp nhiều tính năng quan trọng khác nhau như:

- Kiểm thử theo hướng từ khóa (key-word driven), theo hướng dữ liệu (data driven) và kiểm thử phân tán (distributed)
- Kiểm thử tiện ích mở rộng trong trình duyệt
- Object Eye Internal Recorder
- SMTP Integration
- Tích hợp với các công cụ theo dõi lỗi như Jira, Mantis, TFS và FogBugz
- Tích hợp với các công cụ quản lý kiểm thử như HP Quality Center, Zephyr, TestRail và Microsoft VSTS
- Centralized Object Repository (Kho lưu trữ đối tượng tập trung)

- Version Control System Integration (Tích hợp hệ thống kiểm soát phiên bản)

- Customized Recording Rule (Quy tắc ghi tùy chỉnh)

❖ **HPE Unified Functional Testing (HP – UFT formerly QTP)**

HPE UFT cung cấp tính năng tự động hóa kiểm thử để kiểm thử chức năng và kiểm thử hồi quy cho các ứng dụng phần mềm. Ngôn ngữ script Visual Basic Scripting Edition được ứng dụng bởi công cụ này để đăng ký các quá trình kiểm thử và vận hành các đối tượng và điều khiển khác nhau trong việc test các ứng dụng.

Ngoài ra, QTP cung cấp các tính năng khác như:

- Tích hợp với Mercury Business Process Testing và Mercury Quality Center

- Nhận dạng Unique Smart Object

- Cơ chế xử lý lỗi

- Tạo các tham số cho đối tượng, checkpoint và bảng điều hướng dữ liệu

- Tài liệu tự động

❖ **TestComplete**

TestComplete là một nền tảng kiểm thử chức năng cung cấp các giải pháp khác nhau để tự động kiểm thử. Công cụ này sử dụng cho máy tính để bàn, web và các ứng dụng di động.

TestComplete cung cấp các tính năng sau:

- GUI testing

- Hỗ trợ ngôn ngữ test – JavaScript, Python, VBScript, JScript, DelphiScript, C++ Script & C# Script

- Kiểm thử trình hiển thị
- Kiểm thử theo script (Scripted testing)
- Kiểm thử ghi và phát lại (Test recording and playback)

❖ **Ranorex**

Ranorex Studio cung cấp các công cụ tự động hóa testing khác nhau bao gồm việc test tất cả các ứng dụng máy tính để bàn, web và thiết bị di động.

Cụ thể hơn, Ranorex cung cấp các tính năng sau:

- Kiểm thử GUI
- Có thể tái sử dụng test code
- Phát hiện bug
- Tích hợp với nhiều công cụ khác nhau
- Ghi và phát lại

❖ **Sahi**

Sahi là một công cụ kiểm thử phần mềm tự động hóa áp dụng cho việc test các ứng dụng web. Mã nguồn mở Sahi được viết bằng ngôn ngữ lập trình Java và JavaScript.

Sahi cung cấp các tính năng sau:

- Thực hiện kiểm thử nhiều trình duyệt cùng lúc

- Hỗ trợ các framework ExtJS, ZK, Dojo, YUI, ...
- Ghi lại và phát lại khi test trình duyệt

❖ **Watir**

Watir là một công cụ kiểm thử mã nguồn mở được tạo thành từ các thư viện Ruby để tự động kiểm thử ứng dụng web.

Công cụ này cung cấp các tính năng sau:

- Kiểm thử bất kỳ ứng dụng web dựa trên ngôn ngữ nào
- Kiểm thử trên nhiều trình duyệt
- Tương thích với các công cụ phát triển theo định hướng kinh doanh như RSpec, Cucumber và Test/ Unit
- Kiểm thử các nút, biểu mẫu, liên kết và phản hồi của chúng trên trang web

❖ **Tosca Testsuite**

Tosca Testsuite là một công cụ phần mềm để thực hiện tự động kiểm thử phần mềm chức năng và hồi quy. Ngoài chức năng tự động hóa thử nghiệm, TOSCA bao gồm quản lý kiểm thử tích hợp, giao diện người dùng đồ họa (GUI), giao diện dòng lệnh (CLI) và giao diện lập trình ứng dụng (API).

Tosca Testsuite đi kèm với các tính năng sau:

- Lập kế hoạch và thiết kế trường hợp thử nghiệm
- Kiểm tra cung cấp dữ liệu
- Dịch vụ mạng ảo hóa

- Kiểm tra ứng dụng di động
- Quản lý tích hợp
- Bảo hiểm rủi ro

❖ **Telerik TestStudio**

Telerik TestStudio cung cấp giải pháp để tự động kiểm thử ứng dụng trên máy tính để bàn, web và thiết bị di động bao gồm kiểm thử giao diện người dùng, load và hiệu suất.

Công cụ này cung cấp nhiều khả năng tương thích khác nhau như:

- Hỗ trợ các ngôn ngữ lập trình như HTML, AJAX, ASP.NET, JavaScript, Silverlight, WPF và MVC
- Tích hợp với Visual Basic Studio 2010 và 2012
- Ghi và phát lại
- Kiểm thử trên nhiều trình duyệt
- Kiểm thử thủ công
- Tích hợp với các công cụ theo dõi bug

❖ **Katalon Studio**

Katalon Studio là một công cụ kiểm thử phần mềm tự động hóa miễn phí được phát triển bởi Katalon LLC. Công cụ này được xây dựng dựa trên các framework tự động hóa mã nguồn mở Selenium, Appium với giao diện IDE chuyên biệt để kiểm tra API, web và thiết bị di động. Công cụ này bao gồm một gói đầy đủ các tính năng mạnh mẽ giúp dễ dàng tự động hóa kiểm thử giao diện người dùng web.

Katalon Studio bao gồm các tính năng sau:

- Kho lưu trữ đối tượng tích hợp, XPath, nhận dạng lại đối tượng
- Hỗ trợ các ngôn ngữ script Java/ Groovy
- Hỗ trợ tích hợp cho kiểm thử dựa trên hình ảnh
- Hỗ trợ các công cụ Tích hợp liên tục như Jenkins & TeamCity
- Hỗ trợ Duel-editor Interface
- Quy trình thực thi có thể tùy chỉnh.

CHƯƠNG 2. CÔNG CỤ KIỂM THỬ SELENIUM

2.1 Công cụ Selenium WebDriver

2.1.1 Khái niệm

WebDriver là một giao diện lập trình ứng dụng (API) cho phép thực hiện thao tác trên các trình duyệt web như mở trang web, nhập dữ liệu, nhấn nút, kéo thả, chụp ảnh màn hình ... WebDriver được ra đời vào năm 2006 bởi Simon Stewart, một kỹ sư phần mềm. Năm 2008, WebDriver được kết hợp với Selenium RC để tạo ra Selenium 2.0, phiên bản hiện tại của Selenium.

Selenium WebDriver là một thành phần của Selenium, bao gồm cả Selenium IDE, Selenium Grid và Selenium RC. Selenium WebDriver là phiên bản nâng cấp của Selenium RC, với khả năng tương tác trực tiếp với trình duyệt và thực hiện các thao tác như click, nhập liệu, kéo thả, chụp ảnh màn hình, ... Selenium WebDriver hỗ trợ nhiều ngôn ngữ lập trình và nhiều trình duyệt khác nhau rất thân thiện và dễ sử dụng.

2.1.2 Ưu và nhược điểm của Selenium WebDriver

- *Ưu điểm:*

- + Mã nguồn mở: Selenium WebDriver là một dự án mã nguồn mở, có nghĩa là bất kỳ ai cũng có thể sử dụng và đóng góp cho dự án. Điều này giúp Selenium WebDriver được phát triển và cập nhật thường xuyên, đảm bảo luôn đáp ứng được các nhu cầu của người dùng.

- + Tương thích nhiều trình duyệt: Selenium WebDriver hỗ trợ nhiều trình duyệt phổ biến, bao gồm Chrome, Firefox, Edge, Safari, ... Điều này giúp bạn có thể sử dụng Selenium WebDriver để kiểm thử các ứng dụng web trên nhiều trình duyệt khác nhau.

+ Hỗ trợ đa ngôn ngữ lập trình: Selenium WebDriver hỗ trợ nhiều ngôn ngữ lập trình phổ biến, bao gồm Java, Python, C#, ... Điều này giúp bạn có thể sử dụng ngôn ngữ lập trình mà mình mong muốn để tạo các kịch bản kiểm thử tự động.

+ Tốc độ thực thi nhanh: Selenium Webdriver có tốc độ thực thi nhanh và hiệu quả.

+ Nâng cao chất lượng ứng dụng: Selenium WebDriver có thể giúp bạn nâng cao chất lượng ứng dụng web của mình bằng cách phát hiện và khắc phục các lỗi sớm.

- *Nhược điểm:*

+ Chỉ hỗ trợ ứng dụng web: Selenium WebDriver chỉ hỗ trợ kiểm thử tự động các ứng dụng web. Điều này có nghĩa là nếu bạn muốn kiểm tra tốc độ ứng dụng trên Windows, bạn sẽ gặp nhiều khó khăn vì Selenium WebDriver không hỗ trợ đầy đủ các tính năng và công cụ tiện ích trên hệ điều hành này. Đây là một nhược điểm lớn của Selenium WebDriver khi làm việc với Windows.

+ Đòi hỏi kinh nghiệm lập trình: Để sử dụng Selenium WebDriver, bạn cần có kiến thức về lập trình. Điều này có thể là một trở ngại lớn đối với những bạn không có kinh nghiệm lập trình.

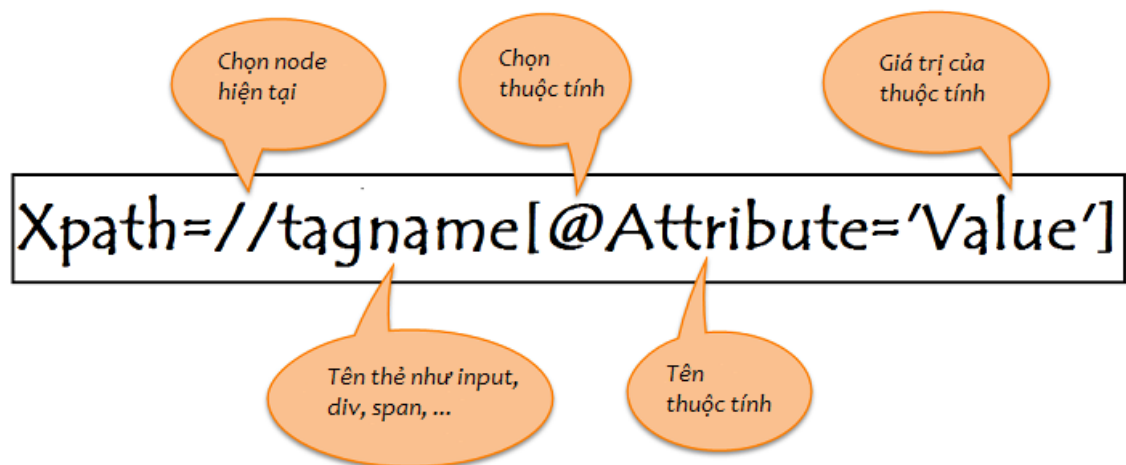
+ Khó bảo trì và mở rộng: Các kịch bản kiểm thử Selenium WebDriver có thể trở nên phức tạp và khó bảo trì, đặc biệt là đối với các ứng dụng web phức tạp. Ngoài ra, Selenium WebDriver cũng có thể khó mở rộng để đáp ứng các nhu cầu cụ thể của bạn.

+ Không có hỗ trợ kỹ thuật: Bạn không có sự hỗ trợ kỹ thuật chính thức từ nhà phát triển của Selenium WebDriver. Bạn phải tự tìm kiếm và giải quyết các vấn đề gặp phải khi sử dụng Selenium WebDriver. Bạn có thể tham khảo các tài liệu, diễn đàn, blog hay video trên mạng để học hỏi và giải đáp các thắc mắc của mình.

+ Thiếu tính năng báo cáo tự động: Bạn không có tính năng báo cáo tự động khi sử dụng Selenium WebDriver. Bạn phải tự viết mã để tạo ra các báo cáo chi tiết về kết quả kiểm thử, số lượng lỗi, thời gian thực thi, ... Bạn cũng có thể sử dụng các công cụ khác như TestNG, JUnit, Allure, ... để hỗ trợ việc tạo báo cáo.

2.1.3 Cách lấy xpath của element trên website

Xpath được sử dụng để xác định các phần tử web dựa trên path XML của nó. Đây được xem là kỹ thuật hiệu quả nhất, có thể xác định tất cả các phần tử trên trang web thường là các element không định vị được bằng id, name, class. Tuy nhiên khi có sự thay đổi trong cấu trúc trang thì bạn sẽ cần phải update lại thông tin các element liên quan.



Hình 2.1: Đường dẫn của Xpath [1]

Có 2 loại Xpath: Xpath tuyệt đối và Xpath tương đối.

- Xpath tuyệt đối:

Xpath tuyệt đối bắt đầu bằng dấu gạch chéo đơn `/`, cho phép xác định một đường dẫn tuyệt đối đến đối tượng UI.

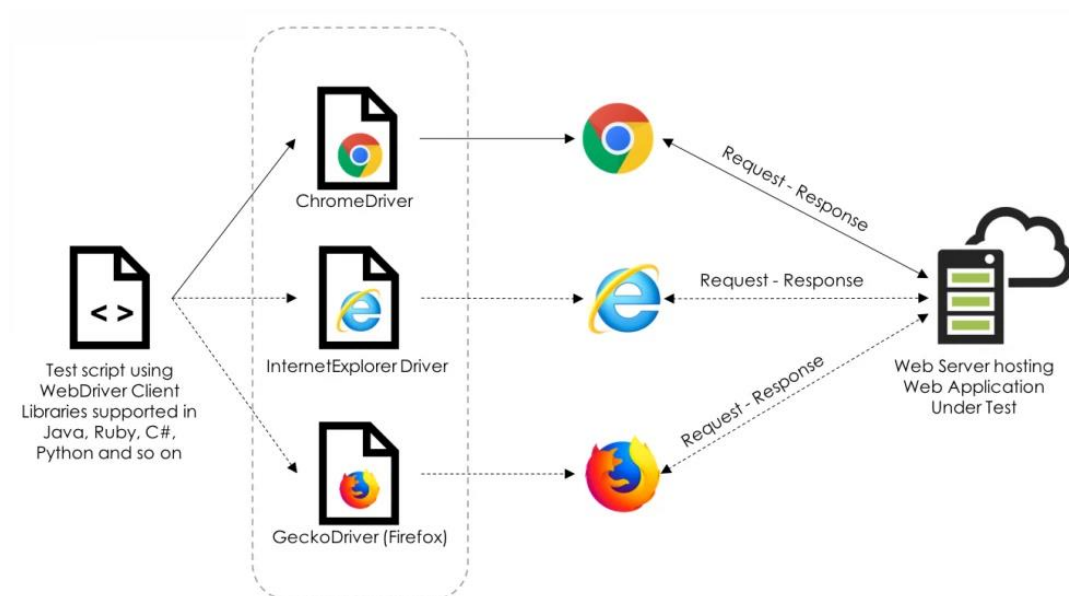
Xpath tuyệt đối được xem là là cách tìm kiếm phần tử dễ dàng nhất, tuy nhiên nhược điểm của nó là nếu có bất kỳ thay đổi nào trên đường dẫn của element thì Xpath lấy sẽ sai.

- Xpath tương đối: Xpath tương đối bắt đầu bằng 2 dấu gạch chéo "/", cho phép xác định một đối tượng UI ở bất kỳ đâu trên trang web, không cần bắt đầu bởi thẻ html trong đường dẫn.

Cách lấy Xpath từ Inspect của Chrome:

- Bước 1: Chạy và khởi động trình duyệt.
- Bước 2: Trỏ chuột vào vị trí muốn lấy. Sau đó, bấm chuột phải rồi chọn Inspect.
- Bước 3: Kiểm tra lại xem vị trí đã được xác định đúng chưa. Sau đó, bấm chuột phải rồi chọn Copy và chọn Copy Xpath.

2.1.4 Một vài khái niệm cần biết trong Selenium WebDriver



Hình 2.2: Quy trình xử lý dữ liệu trong Selenium WebDriver [1]

- *Session:*

Session là một kết nối giữa WebDriver và trình duyệt. Mỗi session có một ID duy nhất, được sử dụng để xác định các lệnh và kết quả. Một session được tạo ra khi bạn khởi tạo một đối tượng WebDriver trong mã của bạn. Một session được kết thúc khi bạn đóng trình duyệt hoặc gọi phương thức `quit()` của đối tượng WebDriver.

- *Element:*

Element là đối tượng đại diện cho một thành phần của giao diện người dùng trên trang web, ví dụ như một nút, một liên kết, một ô nhập liệu, ... Bạn có thể tìm kiếm các Element trên trang web bằng cách sử dụng các phương thức `findElementBy...()` của đối tượng WebDriver, hoặc bằng cách sử dụng các phương thức `findElementsBy...()` để tìm kiếm nhiều Element cùng một lúc. Bạn có thể tương tác với các Element bằng cách sử dụng các phương thức như `click()`, `sendKeys()`, `clear()`, ...

- *Locator:*

Locator là một cách để xác định vị trí của một Element trên trang web, bằng cách sử dụng một thuộc tính hoặc một biểu thức của Element đó. Có nhiều loại locator khác nhau, ví dụ như ID, name, class name, tag name, CSS selector, XPath, ... Bạn có thể sử dụng Locator để truyền vào các phương thức `findElementBy...()` hoặc `findElementsBy...()` để tìm kiếm các Element mong muốn.

- *Wait:*

Wait là một kỹ thuật cho phép bạn đợi một khoảng thời gian nhất định hoặc cho đến khi một điều kiện nào đó được thỏa mãn trước khi tiếp tục thực hiện các lệnh tiếp theo. Có hai loại Wait chính là Implicit Wait và Explicit Wait.

Implicit Wait là một cài đặt toàn cục cho phép bạn đặt một khoảng thời gian tối đa mà WebDriver sẽ đợi cho mỗi lần tìm kiếm Element. Explicit Wait là một cài đặt cục bộ cho phép bạn đặt một khoảng thời gian tối đa và một điều kiện cụ thể mà WebDriver sẽ đợi cho một Element nào đó.

- *Alert:*

Alert là hộp thoại xuất hiện trên trang web, yêu cầu người dùng nhập liệu hoặc xác nhận một hành động nào đó. Có ba loại Alert chính là Simple Alert, Confirmation Alert và Prompt Alert. Bạn có thể xử lý các Alert bằng cách sử dụng lớp Alert và các phương thức như accept(), dismiss(), sendKeys(), ...

2.2 Ngôn ngữ lập trình Java

2.2.1 Khái niệm

Java là một ngôn ngữ lập trình, để lập trình các ứng dụng (web app, android app, ...). Java được phát triển bởi Sun Microsystem vào năm 1995 và là ngôn ngữ lập trình hướng đối tượng.

2.2.2 Mục đích sử dụng Java

- Lập trình các ứng dụng cho các thiết bị android (android app): ứng dụng trên điện thoại di động, máy tính bảng..., thiết bị sử dụng hệ điều hành android.

- Lập trình các ứng dụng web (web app): có thể một trong những trang web bạn truy cập hằng ngày được lập trình bằng Java.

- Lập trình các ứng dụng trên hệ điều hành window (win app): ứng dụng trên máy tính, thiết bị sử dụng hệ điều hành microsoft window.

- Phát triển nhiều loại ứng dụng khác nhau: Cơ sở dữ liệu, mạng, Internet, viễn thông, giải trí, ...

2.2.3 Đặc điểm cơ bản của Java

- *Java là ngôn ngữ lập trình hướng đối tượng:*

- + Mọi thực thể trong chương trình đều là một đối tượng (1 class xác định).
- + Các biến, hàm đều nằm trong một class nào đó.

- *Đơn giản:*

- + Loại bỏ con trỏ.
- + Loại bỏ lệnh goto.
- + Không cho phép đa kế thừa (chuyển sang sử dụng interface).

- *Độc lập nền:*

Khác với các ngôn ngữ lập trình khác, thay vì biên dịch mã nguồn thành mã máy hoặc thông dịch mã nguồn khi chạy, Java được thiết kế để biên dịch mã nguồn thành bytecode, bytecode sau đó sẽ được môi trường thực thi (runtime environment). Do đó một chương trình viết bằng Java có thể chạy trên nhiều thiết bị, nhiều hệ điều hành khác nhau, với điều kiện ở đó có cài sẵn máy ảo Java.

- *Mạnh mẽ:*

- + Quá trình cấp phát, giải phóng bộ nhớ được thực hiện tự động.
- + Yêu cầu chặt chẽ khi khai báo dữ liệu, ép kiểu dữ liệu.
- + Tự động phát hiện lỗi lúc biên dịch.
- + Không sử dụng con trỏ hoặc các phép toán con trỏ.

- *Đa nhiệm:*

Java hỗ trợ lập trình đa luồng (multithreading): cho phép hoạt động đa tiến trình, tiến trình có thể chạy song song cùng một thời điểm và có thể tương tác với nhau.

- *Cơ chế thu gom rác tự động:*

Khi các đối tượng được tạo ra trong Java, chúng sẽ được JRE tự động cấp phát không gian bộ nhớ cho các đối tượng trên heap.

- *Tính an toàn và bảo mật:*

- + Ngôn ngữ lập trình Java yêu cầu chặt chẽ về kiểu dữ liệu.
- + Dữ liệu phải được khai báo tường minh.
- + Không sử dụng con trỏ và các phép toán với con trỏ.
- + Java kiểm soát chặt chẽ việc truy nhập đến mảng, chuỗi. Không cho phép sử dụng các kỹ thuật tràn. Do đó các truy nhập sẽ không vượt quá kích thước của mảng hoặc chuỗi.
- + Quá trình cấp phát và giải phóng bộ nhớ được thực hiện tự động.
- + Cơ chế xử lý lỗi giúp việc xử lý và phục hồi lỗi dễ dàng hơn.
- + Java cung cấp một môi trường quản lý chương trình với nhiều mức khác nhau:

- Mức 1 : Chỉ có thể truy xuất dữ liệu cũng như phương thức thông qua giao diện mà lớp cung cấp.
- Mức 2 : Trình biên dịch kiểm soát các đoạn mã sao cho tuân thủ các quy tắc của ngôn ngữ lập trình Java trước khi thông dịch.
- Mức 3 : Trình thông dịch sẽ kiểm tra mã bytecode xem các đoạn mã này có đảm bảo được các quy định, quy tắc trước khi thực thi.
- Mức 4: Java kiểm soát việc nạp các lớp vào bộ nhớ để giám sát việc vi phạm giới hạn truy xuất trước khi nạp vào hệ thống.

2.2.4 Các thành phần của Java SE Platform

Java Platform gồm có 3 thành phần chính:

- Java Virtual Machine (Java VM): Máy ảo Java.
- Java Application Programming Interface (Java API).
- Java Development Kit (JDK) gồm trình biên dịch, thông dịch, trợ giúp, soạn tài liệu, ... và các thư viện chuẩn.

2.3 Nền tảng lập trình JUnit

2.3.1 Khái niệm

JUnit là một framework phổ biến được sử dụng cho công việc viết và thực hiện các bài kiểm tra đơn vị trong việc phát triển phần mềm bằng ngôn ngữ lập trình Java. Nền tảng cung cấp các cơ chế thông dụng để viết các đơn vị kiểm tra bài kiểm tra, đảm bảo tính đúng đắn trong các phần nhỏ của nguồn mã.

Nền tảng JUnit cung cấp cú pháp đơn giản và dễ sử dụng. Đây là một yếu tố không thể thiếu trong quy trình phát triển phần mềm phổ biến sử dụng ngôn ngữ lập trình Java. Công nghệ có tác dụng tăng cường chất lượng của mã nguồn thông qua quá trình cung cấp phản hồi nhanh chóng về mã nguồn và duy trì tính ổn định của sản phẩm phần mềm.

2.3.2 Vai trò của việc ứng dụng JUnit

Việc sử dụng JUnit mang lại nhiều lợi ích quan trọng trong quá trình phát triển phần mềm, bao gồm:

- Đảm bảo nguồn mã chất lượng

JUnit đảm bảo nguồn mã hoạt động được mong đợi thông qua công việc viết và tự động kiểm tra bài kiểm tra. Hệ thống có đủ khả năng phát hiện lỗi sớm và giữ cho nguồn mã luôn ổn định.

Bằng cách viết và chạy bài kiểm tra thường xuyên, JUnit giúp tăng cường độ tin cậy của nguồn mã hóa và giảm thiểu rủi ro trong quá trình phát triển, duy trì mã nguồn.

- Tiết kiệm thời gian và tài nguyên

Quá trình tự động hóa kiểm tra với JUnit trực tiếp giảm thiểu công việc thủ công và tiết kiệm thời gian cho hoạt động kiểm tra mã nguồn. Nhờ vào cách hình thành các bài kiểm tra sẽ giúp người dùng hiểu rõ hơn về cách hoạt động và sử dụng mã nguồn.

JUnit tích hợp tốt với CI/CD quy trình nhằm tự động hóa quá trình kiểm tra nguồn mã trong quá trình phát triển khai báo và duy trì nguồn mã hóa.

2.3.3 Những thuật ngữ cần biết khi áp dụng JUnit

Trong JUnit, người dùng nên chú ý đến một số thuật ngữ dưới đây để sử dụng framework kiểm tra hiệu quả. Dưới đây là một số khái niệm quan trọng:

- *Test Case (Bài kiểm tra)*: Đây là trường hợp cụ thể mà bạn muốn kiểm tra trong mã nguồn của bạn. Một bài kiểm tra thường tương ứng với một phương thức trong lớp kiểm tra.

- *Test Suite (Bộ kiểm tra)*: Đây là nhóm các bài kiểm tra mà bạn muốn chạy cùng nhau. Bằng cách sử dụng Test Suite, bạn có thể chỉ định các bài kiểm tra cụ thể nào cần phải chạy.

- *Annotations (Chú thích)*: JUnit sử dụng các chú thích để đánh dấu các phương thức kiểm tra và thiết lập trước/sau khi chạy kiểm tra.

Ví dụ: `@Test`, `@Before`, `@After`, `@BeforeClass`, và `@AfterClass` là những annotation quan trọng trong JUnit.

- *Assertions (Các khả năng so sánh)*: JUnit cung cấp một số phương thức assertions để kiểm tra kết quả của bài kiểm tra.

Ví dụ, assertEquals, assertTrue và assertFalse là những phương thức assertions thường được sử dụng.

- *Test Runner (Bộ chạy kiểm tra)*: Đây là công cụ hoặc lớp mà bạn sử dụng để chạy bài kiểm tra của bạn. JUnit có các Test Runner mặc định và cũng cho phép tùy chỉnh Test Runner.

- *Parameterized Tests (Bài kiểm tra tham số hóa)*: Đây là một tính năng của JUnit cho phép bạn chạy cùng một bài kiểm tra với nhiều bộ tham số khác nhau.

- *Mocking and Stubbing (Mock và Stub)*: JUnit thường được kết hợp với các framework khác như Mockito để tạo ra các đối tượng giả lập (mock objects) và đặt giá trị trả về (stubs) trong các bài kiểm tra.

- *Fixtures*: Thuật ngữ đề cập đến những điều kiện ban đầu cần thiết để thực hiện bài kiểm tra một cách đáng tin cậy. Trong đó bao gồm hoạt động thiết lập dữ liệu ban đầu, tạo ra các đối tượng cần thiết và chuẩn bị môi trường thực thi cho bài kiểm tra.

Trong JUnit, bạn có thể sử dụng các annotations như @Before để thiết lập fixtures trước khi chạy mỗi bài kiểm tra và @After để dọn dẹp fixtures sau khi bài kiểm tra kết thúc.

- *Mock*: Giả lập đề cập đến việc tạo các cài đặt đối tượng để thay thế các thực thi đối tượng trong bài kiểm tra. Mô hình đối tượng giúp bạn mô phỏng hoạt động của các đối tượng một cách chính xác và kiểm soát dữ liệu đầu vào và đầu ra của chúng.

Việc sử dụng các framework như Mockito trong JUnit giúp tạo và quản lý các đối tượng mô phỏng một cách hiệu quả.

- *JUnit Class*: Thuật ngữ thường đề cập đến các lớp (lớp) chứa các bài kiểm tra (trường hợp kiểm tra) và các đồ đặc tương ứng. Mỗi phương thức trong lớp kiểm tra đều được đánh dấu bằng chú thích `@Test` để xác định rằng đó là một bài kiểm tra.

JUnit Class cũng có thể chứa các phương thức được sử dụng cho công việc cài đặt và dọn dẹp các cố định bằng cách sử dụng các chú thích `@Before` và `@After`.

2.3.4 Giới thiệu các phiên bản của JUnit

Hiện tại, JUnit đã có nhiều phiên bản dành cho sự phát triển và cải tiến theo thời gian. Trong phần tiếp theo giới thiệu một số phiên bản chính của JUnit:

- JUnit 3:

Đây là phiên bản đầu tiên của JUnit ra mắt vào năm 1997. JUnit 3 sử dụng kỹ thuật kế thừa để tạo các bài kiểm tra và không sử dụng các chú thích như trong các phiên bản sau này. Cấu hình và quản lý bài kiểm tra trong JUnit 3 diễn ra tốn kém hơn và thủ công hơn.

- JUnit 4:

JUnit 4 ra đời vào năm 2006 và đánh dấu một bước tiến lớn với sự giới thiệu của các chú thích và người chạy. Chú thích và trình chạy giúp JUnit 4 trở nên dễ sử dụng hơn, giảm thiểu công việc lặp đi lặp lại và tăng cường tính tùy biến. Điều này đã làm nên sự phổ biến của JUnit 4 trong cộng đồng phát triển phần mềm.

- JUnit 5:

JUnit 5 hay còn được gọi là JUnit Jupiter. Đây là một phiên bản hoàn toàn mới của JUnit và ra đời vào năm 2017. JUnit 5 mang đến một số cải tiến kể về hiệu suất, linh hoạt và tính năng. Nền tảng hỗ trợ các tính năng hiện tại như lambdas, Generics và Stream cũng như sở hữu mô-đun cấu trúc tốt hơn cho tính năng mở rộng, tùy chỉnh.

2.4 Môi trường phát triển tích hợp Eclipse

2.4.1 Khái niệm

Eclipse là một công cụ lập trình mã nguồn mở mạnh mẽ, được thiết kế như một môi trường phát triển tích hợp chủ yếu cho Java, nhưng cũng hỗ trợ lập trình bằng nhiều ngôn ngữ lập trình khác như C/C++, PHP, Ruby, ... Cùng với khả năng mở rộng mã nguồn thông qua việc tích hợp các plugin cho dự án, Eclipse là sự lựa chọn linh hoạt cho các nhà phát triển. Điều đặc biệt là Eclipse hoàn toàn miễn phí và đi kèm với nhiều plugin hữu ích.

2.4.2 Ưu và nhược điểm của Eclipse

- *Ưu điểm:*

- + Sử dụng cho hai hệ điều hành là Windows và Linux.
- + Hỗ trợ nhiều ngôn ngữ lập trình khác như Java, C/C++, PHP.
- + Dễ dàng làm việc với cả GUI và non-GUI.
- + Tốc độ tải nhanh với SWT/JFace.
- + Đa dạng plugin hữu ích như Testing, Code formatter...
- + Phù hợp cho các dự án lớn và triển khai trên máy tính cũng như đám mây lưu trữ.
- + Mã nguồn mở giúp cá nhân hóa và tạo cấu trúc đầy đủ tính năng cho các công cụ tích hợp.

- *Nhược điểm:*

Nhược điểm bao gồm quá trình cài đặt khá phức tạp, đòi hỏi tài nguyên phần cứng và bộ nhớ máy lớn. Việc tích hợp nhiều plugin có thể dẫn đến sự thiếu nhất quán. Tuy nhiên, để khắc phục những khó khăn này, nhà cung cấp đã phát triển phiên bản Eclipse Luna mới, không đòi hỏi quá trình cài đặt phức tạp. Điều này giúp tiết kiệm tài nguyên máy tính, phiên bản mới này vẫn đang trong quá trình được chấp nhận rộng rãi.

CHƯƠNG 3. ỨNG DỤNG CÔNG CỤ KIỂM THỬ CHO PHẦN MỀM BÁN KHÓA HỌC ONLINE

3.1 Tổng quan về hệ thống kiểm thử

3.1.1 Giới thiệu chung

- Website bán khóa học ColorMe được xây dựng với mục tiêu cung cấp một nền tảng thuận tiện và hiệu quả cho cá nhân, giáo viên, và tổ chức giáo dục để tiếp cận và phục vụ học viên một cách dễ dàng. Trang web này được tạo ra để giúp giáo viên và các tổ chức giáo dục mở rộng phạm vi đào tạo mà không gặp khó khăn về tài chính và quản lý nguồn lực nhân sự.

- Người bán khóa học bao gồm giáo viên cá nhân, nhóm giáo viên, và tổ chức giáo dục có nhu cầu cung cấp các khóa học chất lượng cao cho học viên. Họ có thể sử dụng các dịch vụ như đăng ký khóa học, quản lý nội dung, đánh giá học viên, và nhiều tính năng khác.

- Học viên của trang web này là cá nhân và tổ chức có nhu cầu học tập và nâng cao kỹ năng chuyên môn. Họ có thể tìm kiếm, so sánh và đăng ký các khóa học phù hợp với mục tiêu và mong muốn của mình.

- Trang web cung cấp các tính năng như đăng ký khóa học, quản lý nội dung, đánh giá và phản hồi để đáp ứng đa dạng nhu cầu của giáo viên và học viên. Đồng thời, nó tạo cơ hội cho việc học tập thuận lợi và hiệu quả, mang lại lợi ích cho cả người bán khóa học và học viên trong quá trình truyền đạt kiến thức và kỹ năng.

3.1.2 Đặc tả use case

Bảng 3.1: Đặc tả use case “Tạo tài khoản”

Use case	Tạo tài khoản	
Tác nhân	Học viên	
Mô tả	Học viên thực hiện đăng ký tài khoản	
Tiền điều kiện	Học viên truy cập vào hệ thống	
Hậu điều kiện	Học viên tạo tài khoản thành công	
Luồng sự kiện chính	Học viên	Hệ thống
	1. Nhấn chọn Đăng nhập	2. Hiện thị trang Đăng nhập
	3. Nhấn nút Tạo tài khoản	4. Hiện thị trang Tạo tài khoản
	5. Nhập các thông tin như Họ và tên, Email, Mật khẩu, Nhập lại mật khẩu, Số điện thoại và nhấn nút Đăng ký	6. Kiểm tra thông tin đã nhập. Nếu hợp lệ, lưu lại thông tin vào CSDL và hiện thị thông báo “Color Me đã gửi email xác nhận vào hòm thư “{Email}”. Bạn vui lòng kiểm tra email để hoàn thành đăng kí”
Ngoại lệ	Học viên	Hệ thống
	5.1 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Họ và tên và nhấn nút Đăng ký	6.1 Hiện thị thông báo lỗi “Vui lòng nhập họ và tên”

	5.2 Nhập ký tự khác chữ cho trường Họ và tên và nhấn nút Đăng ký	6.2 Hiện thị thông báo lỗi “Vui lòng nhập Họ và tên theo đúng định dạng”
	5.3 Không nhập hoặc nhập toàn ký tự khoảng trắng hoặc nhập email không đúng định dạng cho trường Email và nhấn nút Đăng ký	6.3 Hiện thị thông báo lỗi “Vui lòng nhập Email theo đúng định dạng”
	5.4 Nhập email đã tạo tài khoản cho trường Email và nhấn nút Đăng ký	6.4 Hiện thị thông báo lỗi “Email đã tồn tại”
	5.5 Không nhập trường Mật khẩu và nhấn nút Đăng ký	6.5 Hiện thị thông báo lỗi “Vui lòng nhập mật khẩu”
	5.6 Không nhập trường Nhập lại mật khẩu và nhấn nút Đăng ký	6.6 Hiện thị thông báo lỗi “Vui lòng nhập lại mật khẩu”
	5.7 Nhập thông tin cho 2 trường Mật khẩu và Nhập lại mật khẩu không khớp và nhấn nút Đăng ký	6.7 Hiện thị thông báo lỗi “Mật khẩu không trùng khớp”
	5.8 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Số điện thoại và nhấn	6.8 Hiện thị thông báo lỗi “Vui lòng nhập số điện thoại liên lạc”

	nút Đăng ký	
	5.9 Nhập số điện thoại không đúng định dạng cho trường Số điện thoại và nhấn nút Đăng ký	6.9 Hiển thị thông báo lỗi “Vui lòng nhập Số điện thoại theo đúng định dạng”
	5.10 Nhập số điện thoại đã đăng ký tài khoản cho trường Số điện thoại và nhấn nút Đăng ký	6.10 Hiển thị thông báo lỗi “Vui lòng nhập Số điện thoại chưa được tạo tài khoản”

Bảng 3.2: Đặc tả use case “Đăng nhập”

Use case	Đăng nhập	
Tác nhân	Học viên	
Mô tả	Học viên thực hiện đăng nhập vào hệ thống	
Tiền điều kiện	Học viên đã có tài khoản để đăng nhập	
Hậu điều kiện	Học viên đăng nhập vào hệ thống thành công	
Luồng sự kiện chính	Học viên	Hệ thống
	1. Nhấn chọn Đăng nhập	2. Hiển thị trang Đăng nhập
	3. Nhập các thông tin như Tên đăng nhập/Email, Mật khẩu và nhấn nút Đăng nhập	4. Kiểm tra thông tin đã nhập. Nếu hợp lệ, đăng nhập vào hệ thống và hiển thị Trang chủ

Ngoại lệ	Học viên	Hệ thống
	3.1 Không nhập hoặc nhập thiếu thông tin	4.1 Nút Đăng nhập bị disable
	3.2 Nhập email không đúng định dạng cho trường Tên đăng nhập/Email và nhấn nút Đăng nhập	4.2 Hiện thị thông báo lỗi “Vui lòng nhập Email theo đúng định dạng”
	3.3 Nhập thông tin không chính xác và nhấn nút Đăng nhập	4.3 Hiện thị thông báo lỗi “Email hoặc Password không đúng”

Bảng 3.3: Đặc tả use case “Tìm kiếm thông tin”

Use case	Tìm kiếm thông tin	
Tác nhân	Học viên	
Mô tả	Học viên thực hiện tìm kiếm thông tin	
Tiền điều kiện	Học viên truy cập vào hệ thống	
Hậu điều kiện	Học viên tìm kiếm thông tin thành công	
Luồng sự kiện chính	Học viên	Hệ thống
	1. Nhập từ khóa và nhấn nút Tìm kiếm	2. Tìm kiếm người dùng/bài viết theo từ khóa và hiển thị danh sách người dùng/bài viết tìm thấy
Ngoại lệ	Học viên	Hệ thống
	1.1 Nhập từ khóa ≤ 4	2.1 Hiện thị thông báo lỗi “Từ khóa tìm

	ký tự và nhấn Enter	kiểm phải > 4 ký tự”
	1.2 Nhập từ khóa không có trong hệ thống và nhấn Enter	2.2 Hiện thị thông báo lỗi “Không tìm thấy người dùng”

Bảng 3.4: Đặc tả use case “Quản lý giỏ hàng”

Use case	Quản lý giỏ hàng	
Tác nhân	Học viên	
Mô tả	Học viên thực hiện thêm vào giỏ hàng	
Tiền điều kiện	Học viên đã đăng nhập vào hệ thống	
Hậu điều kiện	Học viên thêm vào giỏ hàng thành công	
Luồng sự kiện chính	Học viên	Hệ thống
	1. Nhấn chọn sản phẩm bất kỳ	2. Hiện thị trang Thông tin chi tiết sản phẩm
	3. Nhấn nút Thêm vào giỏ hàng	4. Kiểm tra thông tin đã nhập. Nếu hợp lệ, lưu lại thông tin vào CSDL và hiện thị thông báo “Đã thêm khóa học vào giỏ hàng”
Ngoại lệ	Không có	

Bảng 3.5: Đặc tả use case “Đăng ký khóa học”

Use case	Đăng ký khóa học	
Tác nhân	Học viên	
Mô tả	Học viên thực hiện đăng ký khóa học	
Tiền điều kiện	Học viên đã đăng nhập vào hệ thống	
Hậu điều kiện	Học viên đăng ký khóa học thành công	
Luồng sự kiện chính	Học viên	Hệ thống
	1. Nhấn nút Đăng kí tại khóa học bất kỳ	2. Hiện thị trang Đăng ký khóa học
	3. Nhập các thông tin như Địa chỉ email, Họ và tên, Số điện thoại, Mã ưu đãi và nhấn nút Đăng ký	4. Kiểm tra thông tin đã nhập. Nếu hợp lệ, lưu lại thông tin vào CSDL và hiện thị thông báo “Đăng kí thành công, bạn vui lòng check mail để kiểm tra thông tin”
Ngoại lệ	Học viên	Hệ thống
	3.1 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Địa chỉ email và nhấn nút Đăng ký	4.1 Hiện thị thông báo lỗi “Vui lòng nhập email”
	3.2 Nhập email không đúng định dạng cho trường Địa chỉ email và nhấn nút Đăng ký	4.2 Hiện thị thông báo lỗi “Vui lòng nhập đúng email”

	3.3 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Họ và tên và nhấn nút Đăng ký	4.3 Hiện thị thông báo lỗi “Vui lòng nhập tên”
	3.4 Nhập tên khác ký tự chữ cho trường Họ và tên và nhấn nút Đăng ký	4.4 Hiện thị thông báo lỗi “Vui lòng nhập đúng tên”
	3.5 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Số điện thoại và nhấn nút Đăng ký	4.5 Hiện thị thông báo lỗi “Vui lòng nhập số điện thoại”
	3.6 Nhập số điện thoại không đúng định dạng cho trường Số điện thoại và nhấn nút Đăng ký	4.6 Hiện thị thông báo lỗi “Vui lòng nhập đúng số điện thoại”
	3.7 Nhập mã khuyến mại không chính xác và nhấn nút Đăng ký	4.7 Hiện thị thông báo lỗi “Mã khuyến mãi không hợp lệ”

Bảng 3.6: Đặc tả use case “Cập nhật thông tin cá nhân”

Use case	Cập nhật thông tin cá nhân	
Tác nhân	Học viên	
Mô tả	Học viên thực hiện cập nhật thông tin cá nhân	
Tiền điều kiện	Học viên đã đăng nhập vào hệ thống	
Hậu điều kiện	Học viên cập nhật thông tin cá nhân thành công	
Luồng sự kiện chính	Học viên	Hệ thống
	1. Nhấn chọn Tên đăng nhập	2. Hiện thị menu sổ xuống có

		chứa Trang cá nhân
	3. Nhấn chọn Trang cá nhân	4. Hiện thị màn hình Thông tin cá nhân
	5. Nhấn nút Sửa thông tin	6. Hiện thị màn hình Sửa thông tin cá nhân
	7. Nhập các thông tin như Mô tả về bạn, Chọn username riêng, Trường học, Nơi làm việc và Địa điểm và nhấn nút Lưu	8. Kiểm tra thông tin đã nhập. Nếu hợp lệ, lưu lại thông tin vào CSDL và hiện thị thông báo “Cập nhật thành công”
Ngoại lệ	Học viên	Hệ thống
	7.1 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Mô tả về bạn và nhấn nút Lưu	8.1 Hiện thị thông báo lỗi “Vui lòng nhập mô tả về bạn”
	7.2 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Chọn username riêng và nhấn nút Lưu	8.2 Hiện thị thông báo lỗi “Vui lòng nhập username riêng”
	7.3 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Trường học và nhấn nút Lưu	8.3 Hiện thị thông báo lỗi “Vui lòng nhập trường học”
	7.4 Không nhập hoặc nhập toàn ký tự khoảng trắng cho trường Nơi làm việc và nhấn nút Lưu	8.4 Hiện thị thông báo lỗi “Vui lòng nhập nơi làm việc”
	7.5 Không nhập hoặc nhập toàn ký	8.5 Hiện thị thông báo lỗi “Vui

	tự khoảng trắng cho trường Địa điểm và nhấn nút Lưu	lòng nhập địa điểm”
--	---	---------------------

Bảng 3.7: Đặc tả use case “Đăng xuất”

Use case	Đăng xuất	
Tác nhân	Học viên	
Mô tả	Học viên thực hiện đăng xuất khỏi hệ thống	
Tiền điều kiện	Học viên đã đăng nhập vào hệ thống	
Hậu điều kiện	Học viên đăng xuất khỏi hệ thống thành công	
Luồng sự kiện chính	Học viên	Hệ thống
	1. Nhấn chọn Tên đăng nhập	2. Hiện thị menu sổ xuống có chứa Đăng xuất
	3. Nhấn chọn Đăng xuất	4. Đăng xuất khỏi hệ thống và hiển thị Trang chủ
Ngoại lệ	Không có	

3.2 Thiết kế các ca kiểm thử

Bảng 3.8: Bảng các ca kiểm thử cho “Tạo tài khoản”

ID	Mục tiêu kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
FUNC_01_01	Kiểm tra không nhập trường Họ và tên	1. Không nhập trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập họ và tên"	Pass
FUNC_01_02	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Họ và tên	1. Nhập toàn ký tự khoảng trắng cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập họ và tên"	Fail
FUNC_01_03	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Họ và tên	1. Nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass

FUNC_01_04	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Họ và tên	1. Nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hòm thư "{Email}". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass
FUNC_01_05	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Họ và tên	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Họ và tên theo đúng định dạng"	Fail
FUNC_01_06	Kiểm tra nhập giá trị có chứa ký tự số cho trường Họ và tên	1. Nhập giá trị có chứa ký tự số cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Họ và tên theo đúng định dạng"	Fail
FUNC_01_07	Kiểm tra không nhập trường Email	1. Không nhập trường Email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Pass

FUNC_01_08	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Email	1. Nhập toàn ký tự khoảng trắng cho trường Email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Pass
FUNC_01_09	Kiểm tra nhập Email không đúng định dạng	1. Nhập Email không đúng định dạng 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Pass
FUNC_01_10	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Email	1. Nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Fail
FUNC_01_11	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Email	1. Nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Pass

FUNC_01_12	Kiểm tra nhập giá trị có chứa ký tự đặc biệt (trừ '@' và '.') cho trường Email	<ol style="list-style-type: none"> 1. Nhập giá trị có chứa ký tự đặc biệt (trừ '@' và '.') cho trường Email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản 	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Fail
FUNC_01_13	Kiểm tra nhập giá trị có chứa ký tự số cho trường Email	<ol style="list-style-type: none"> 1. Nhập giá trị có chứa ký tự số cho trường Email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản 	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass
FUNC_01_14	Kiểm tra nhập Email đã đăng ký	<ol style="list-style-type: none"> 1. Nhập Email đã đăng ký 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản 	3. Hiện thị thông báo lỗi "Email đã tồn tại"	Pass
FUNC_01_15	Kiểm tra không nhập trường Mật khẩu	<ol style="list-style-type: none"> 1. Không nhập trường Mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản 	3. Hiện thị thông báo lỗi "Vui lòng nhập mật khẩu"	Pass

FUNC_01_16	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Mật khẩu	1. Nhập toàn ký tự khoảng trắng cho trường Mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập mật khẩu"	Fail
FUNC_01_17	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng cho trường Mật khẩu	1. Nhập giá trị có chứa ký tự khoảng trắng cho trường Mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass
FUNC_01_18	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Mật khẩu	1. Nhập giá trị có chứa ký tự chữ cho trường Mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass
FUNC_01_19	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Mật khẩu	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass

FUNC_01_20	Kiểm tra nhập giá trị có chứa ký tự số cho trường Mật khẩu	1. Nhập giá trị có chứa ký tự số cho trường Mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư "{Email}". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass
FUNC_01_21	Kiểm tra không nhập trường Nhập lại mật khẩu	1. Không nhập trường Nhập lại mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập lại mật khẩu"	Fail
FUNC_01_22	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Nhập lại mật khẩu	1. Nhập toàn ký tự khoảng trắng cho trường Nhập lại mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập lại mật khẩu"	Fail
FUNC_01_23	Kiểm tra nhập Nhập lại mật khẩu không khớp với Mật khẩu	1. Nhập Nhập lại mật khẩu không khớp với Mật khẩu 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Mật khẩu không trùng khớp"	Pass

FUNC_01_24	Kiểm tra không nhập trường Số điện thoại	1. Không nhập trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập số điện thoại liên lạc"	Pass
FUNC_01_25	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Số điện thoại	1. Nhập toàn ký tự khoảng trắng cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập số điện thoại liên lạc"	Fail
FUNC_01_26	Kiểm tra nhập toàn số 0 cho trường Số điện thoại	1. Nhập toàn số 0 cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Số điện thoại theo đúng định dạng"	Fail
FUNC_01_27	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Số điện thoại	1. Nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass

FUNC_01_28	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Số điện thoại	<p>1. Nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Số điện thoại</p> <p>2. Nhập dữ liệu hợp lệ cho các trường khác</p> <p>3. Nhấn nút Tạo tài khoản</p>	3. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hòm thư "{Email}". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass
FUNC_01_29	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Số điện thoại	<p>1. Nhập giá trị có chứa ký tự đặc biệt cho trường Số điện thoại</p> <p>2. Nhập dữ liệu hợp lệ cho các trường khác</p> <p>3. Nhấn nút Tạo tài khoản</p>	3. Hiện thị thông báo lỗi "Vui lòng nhập Số điện thoại theo đúng định dạng"	Fail
FUNC_01_30	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Số điện thoại	<p>1. Nhập giá trị có chứa ký tự chữ cho trường Số điện thoại</p> <p>2. Nhập dữ liệu hợp lệ cho các trường khác</p> <p>3. Nhấn nút Tạo tài khoản</p>	3. Hiện thị thông báo lỗi "Vui lòng nhập Số điện thoại theo đúng định dạng"	Fail
FUNC_01_31	Kiểm tra nhập giá trị chứa 9 ký tự số cho trường Số điện thoại	<p>1. Nhập giá trị chứa 9 ký tự số cho trường Số điện thoại</p> <p>2. Nhập dữ liệu hợp lệ cho các trường khác</p> <p>3. Nhấn nút Tạo tài</p>	3. Hiện thị thông báo lỗi "Vui lòng nhập Số điện thoại theo đúng định dạng"	Fail

		khoản		
FUNC_01_32	Kiểm tra nhập giá trị chứa 11 ký tự số cho trường Số điện thoại	1. Nhập giá trị chứa 11 ký tự số cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Số điện thoại theo đúng định dạng"	Fail
FUNC_01_33	Kiểm tra nhập Số điện thoại đã đăng ký	1. Nhập Số điện thoại đã đăng ký 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Tạo tài khoản	3. Hiện thị thông báo lỗi "Vui lòng nhập Số điện thoại chưa được tạo tài khoản"	Fail
FUNC_01_34	Kiểm tra đăng ký thành công	1. Nhập dữ liệu hợp lệ cho các trường 2. Nhấn nút Tạo tài khoản	2. Hiện thị thông báo "Color Me đã gửi email xác nhận vào hộp thư \"{Email}\". Bạn vui lòng kiểm tra email để hoàn thành đăng ký"	Pass

Bảng 3.9: Bảng các ca kiểm thử cho “Đăng nhập”

ID	Mục tiêu kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
FUNC_02_01	Kiểm tra không nhập trường Tên đăng nhập/Email	1. Không nhập trường Tên đăng nhập/Email 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Nút Đăng nhập bị disable	Fail
FUNC_02_02	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Tên đăng nhập/Email	1. Nhập toàn ký tự khoảng trắng cho trường Tên đăng nhập/Email 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Hiện thị thông báo lỗi "Email hoặc Password không đúng"	Pass
FUNC_02_03	Kiểm tra nhập Tên đăng nhập/Email không đúng định dạng	1. Nhập Tên đăng nhập/Email không đúng định dạng 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Fail
FUNC_02_04	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Tên đăng nhập/Email	1. Nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Tên đăng nhập/Email 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Chuyển đến màn hình trang chủ tương ứng với tài khoản đăng nhập	Pass

FUNC_02_05	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Tên đăng nhập/Email	1. Nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Tên đăng nhập/Email 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Fail
FUNC_02_06	Kiểm tra nhập giá trị có chứa ký tự đặc biệt (trừ '@' và '.') cho trường Tên đăng nhập/Email	1. Nhập giá trị có chứa ký tự đặc biệt (trừ '@' và '.') cho trường Tên đăng nhập/Email 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Hiện thị thông báo lỗi "Vui lòng nhập Email theo đúng định dạng"	Fail
FUNC_02_07	Kiểm tra nhập giá trị có chứa ký tự số cho trường Tên đăng nhập/Email	1. Nhập giá trị có chứa ký tự số cho trường Tên đăng nhập/Email 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Chuyển đến màn hình trang chủ tương ứng với tài khoản đăng nhập	Pass
FUNC_02_08	Kiểm tra nhập Tên đăng nhập/Email chưa đăng ký	1. Nhập Tên đăng nhập/Email chưa đăng ký 2. Nhập dữ liệu hợp lệ cho trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Hiện thị thông báo lỗi "Email hoặc Password không đúng"	Pass
FUNC_02_09	Kiểm tra không nhập trường Mật khẩu	1. Nhập dữ liệu hợp lệ cho trường Tên đăng nhập/Email 2. Không nhập trường Mật khẩu 3. Nhấn nút Đăng nhập	3. Nút Đăng nhập bị disable	Pass

FUNC_02_10	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Mật khẩu	<ol style="list-style-type: none"> 1. Nhập dữ liệu hợp lệ cho trường Tên đăng nhập/Email 2. Nhập toàn ký tự khoảng trắng cho trường Mật khẩu 3. Nhấn nút Đăng nhập 	3. Hiện thị thông báo lỗi "Email hoặc Password không đúng"	Pass
FUNC_02_11	Kiểm tra nhập Mật khẩu không chính xác	<ol style="list-style-type: none"> 1. Nhập dữ liệu hợp lệ cho trường Tên đăng nhập/Email 2. Nhập Mật khẩu không chính xác 3. Nhấn nút Đăng nhập 	3. Hiện thị thông báo lỗi "Email hoặc Password không đúng"	Pass
FUNC_02_12	Kiểm tra đăng nhập thành công	<ol style="list-style-type: none"> 1. Nhập dữ liệu hợp lệ cho các trường 2. Nhấn nút Đăng nhập 	2. Chuyển đến màn hình trang chủ tương ứng với tài khoản đăng nhập	Pass

Bảng 3.10: Bảng các ca kiểm thử cho “Tìm kiếm thông tin”

ID	Mục tiêu kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
FUNC_03_01	Kiểm tra tìm kiếm khi không nhập từ khóa	<ol style="list-style-type: none"> 1. Không nhập từ khóa 2. Nhấn Enter 	2. Hiện thị thông báo lỗi "Từ khóa tìm kiếm phải > 4 ký tự"	Pass
FUNC_03_02	Kiểm tra tìm kiếm khi nhập từ khóa toàn ký tự khoảng trắng	<ol style="list-style-type: none"> 1. Nhập từ khóa toàn ký tự khoảng trắng 2. Nhấn Enter 	2. Hiện thị thông báo lỗi "Từ khóa tìm kiếm phải > 4 ký tự"	Fail

FUNC_03_03	Kiểm tra tìm kiếm khi nhập từ khóa chứa 4 ký tự	1. Nhập từ khóa chứa 4 ký tự 2. Nhấn Enter	2. Hiện thị thông báo lỗi "Từ khóa tìm kiếm phải > 4 ký tự"	Fail
FUNC_03_04	Kiểm tra tìm kiếm khi nhập từ khóa chứa 5 ký tự	1. Nhập từ khóa chứa 5 ký tự 2. Nhấn Enter	2. Hiện thị danh sách người dùng và bài viết có chứa từ khóa	Pass
FUNC_03_05	Kiểm tra tìm kiếm khi nhập từ khóa chứa 6 ký tự	1. Nhập từ khóa chứa 6 ký tự 2. Nhấn Enter	2. Hiện thị danh sách người dùng và bài viết có chứa từ khóa	Pass
FUNC_03_06	Kiểm tra tìm kiếm chính xác	1. Nhập từ khóa chính xác 2. Nhấn Enter	2. Hiện thị danh sách người dùng và bài viết trùng khớp với từ khóa	Pass
FUNC_03_07	Kiểm tra tìm kiếm tương đối	1. Nhập từ khóa tương đối 2. Nhấn Enter	2. Hiện thị danh sách người dùng và bài viết có chứa từ khóa	Pass
FUNC_03_08	Kiểm tra tìm kiếm không phân biệt chữ hoa, chữ thường	1. Nhập từ khóa là chữ hoa có trong CSDL 2. Nhấn Enter	2. Hiện thị danh sách người dùng và bài viết có chứa từ khóa	Pass
FUNC_03_09	Kiểm tra tìm kiếm với từ khóa có chứa ký tự khoảng trắng ở đầu và cuối	1. Nhập từ khóa có chứa ký tự khoảng trắng ở đầu và cuối 2. Nhấn Enter	2. Hiện thị danh sách người dùng và bài viết có chứa từ khóa	Pass
FUNC_03_10	Kiểm tra tìm kiếm không thành công	1. Nhập từ khóa không có trong CSDL 2. Nhấn Enter	2. Hiện thị thông báo "Không tìm thấy người dùng" - Không bài viết nào	Pass

			được hiển thị	
--	--	--	---------------	--

Bảng 3.11: Bảng các ca kiểm thử cho “Quản lý giỏ hàng”

ID	Mục tiêu kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
FUNC_04_01	Kiểm tra thêm khóa học vào giỏ hàng khi giỏ hàng trống	1. Nhấn nút Thêm vào giỏ hàng của khóa học bất kỳ	1. Hiện thị thông báo "Đã thêm khóa học vào giỏ hàng"	Pass
FUNC_01_02	Kiểm tra thêm khóa học vào giỏ hàng khi khóa học đã có trong giỏ hàng	2. Nhấn nút Thêm vào giỏ hàng của khóa học ở bước 1	2. Không hiện thị thông báo "Đã thêm khóa học vào giỏ hàng"	Pass
FUNC_01_03	Kiểm tra thêm khóa học vào giỏ hàng khi giỏ hàng đã có khóa học khác	3. Nhấn nút Thêm vào giỏ hàng của khóa học khác với khóa học ở bước 1	3. Hiện thị thông báo "Đã thêm khóa học vào giỏ hàng"	Pass
FUNC_01_04	Kiểm tra xóa khóa học khỏi giỏ hàng khi giỏ hàng chỉ có 2 khóa học	4. Nhấn biểu tượng Xóa của khóa học bất kỳ	4. Trong giỏ hàng chỉ còn 1 khóa học	Pass
FUNC_01_05	Kiểm tra xóa khóa học khỏi giỏ hàng khi giỏ hàng chỉ có 1 khóa học	5. Nhấn biểu tượng Xóa của khóa học	5. Hiện thị thông báo "Giỏ hàng của bạn đang trống"	Pass

Bảng 3.12: Bảng các ca kiểm thử cho “Đăng ký khóa học”

ID	Mục tiêu kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
FUNC_05_01	Kiểm tra không nhập trường Địa chỉ email	1. Không nhập trường Địa chỉ email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập email"	Pass
FUNC_05_02	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Địa chỉ email	1. Nhập toàn ký tự khoảng trắng cho trường Địa chỉ email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập email"	Fail
FUNC_05_03	Kiểm tra nhập Địa chỉ email không đúng định dạng	1. Nhập Địa chỉ email không đúng định dạng 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng email"	Pass
FUNC_05_04	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho	1. Nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Địa chỉ email 2. Nhập dữ liệu hợp lệ cho	3. Hiện thị thông báo "Đăng ký thành công, bạn vui lòng check	Fail

	trường Địa chỉ email	các trường khác 3. Nhấn nút Đăng ký	mail để kiểm tra thông tin"	
FUNC_05_05	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Địa chỉ email	1. Nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Địa chỉ email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng email"	Pass
FUNC_05_06	Kiểm tra nhập giá trị có chứa ký tự đặc biệt (trừ '@' và '.') cho trường Địa chỉ email	1. Nhập giá trị có chứa ký tự đặc biệt (trừ '@' và '.') cho trường Địa chỉ email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng email"	Fail
FUNC_05_07	Kiểm tra nhập giá trị có chứa ký tự số cho trường Địa chỉ email	1. Nhập giá trị có chứa ký tự số cho trường Địa chỉ email 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo "Đăng ký thành công, bạn vui lòng check mail để kiểm tra thông tin"	Fail
FUNC_05_08	Kiểm tra không nhập trường Họ và tên	1. Không nhập trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập tên"	Pass

FUNC_ 05_09	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Họ và tên	1. Nhập toàn ký tự khoảng trắng cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập tên"	Fail
FUNC_ 05_10	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Họ và tên	1. Nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo "Đăng ký thành công, bạn vui lòng check mail để kiểm tra thông tin"	Pass
FUNC_ 05_11	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Họ và tên	1. Nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo "Đăng ký thành công, bạn vui lòng check mail để kiểm tra thông tin"	Fail
FUNC_ 05_12	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Họ và tên	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng tên"	Fail

FUNC_ 05_13	Kiểm tra nhập giá trị có chứa ký tự số cho trường Họ và tên	<ol style="list-style-type: none"> 1. Nhập giá trị có chứa ký tự số cho trường Họ và tên 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký 	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng tên"	Fail
FUNC_ 05_14	Kiểm tra không nhập trường Số điện thoại	<ol style="list-style-type: none"> 1. Không nhập trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký 	3. Hiện thị thông báo lỗi "Vui lòng nhập số điện thoại"	Pass
FUNC_ 05_15	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Số điện thoại	<ol style="list-style-type: none"> 1. Nhập toàn ký tự khoảng trắng cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký 	3. Hiện thị thông báo lỗi "Vui lòng nhập số điện thoại"	Fail
FUNC_ 05_16	Kiểm tra nhập toàn số 0 cho trường Số điện thoại	<ol style="list-style-type: none"> 1. Nhập toàn số 0 cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký 	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng số điện thoại"	Fail
FUNC_ 05_17	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở	<ol style="list-style-type: none"> 1. Nhập giá trị có chứa ký tự khoảng trắng ở đầu và cuối cho trường Số điện thoại 	3. Hiện thị thông báo "Đăng ký thành công, bạn	Pass

	đầu và cuối cho trường Số điện thoại	2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	vui lòng check mail để kiểm tra thông tin"	
FUNC_05_18	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Số điện thoại	1. Nhập giá trị có chứa ký tự khoảng trắng ở giữa cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo "Đăng ký thành công, bạn vui lòng check mail để kiểm tra thông tin"	Fail
FUNC_05_19	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Số điện thoại	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng số điện thoại"	Fail
FUNC_05_20	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Số điện thoại	1. Nhập giá trị có chứa ký tự chữ cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng số điện thoại"	Fail
FUNC_05_21	Kiểm tra nhập giá trị chứa 9 ký tự số cho trường Số điện thoại	1. Nhập giá trị chứa 9 ký tự số cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng số điện thoại"	Fail

		3. Nhấn nút Đăng ký		
FUNC_05_22	Kiểm tra nhập giá trị chứa 11 ký tự số cho trường Số điện thoại	1. Nhập giá trị chứa 11 ký tự số cho trường Số điện thoại 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Vui lòng nhập đúng số điện thoại"	Fail
FUNC_05_23	Kiểm tra không nhập trường Mã ưu đãi	1. Không nhập trường Mã ưu đãi 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo "Đăng ký thành công, bạn vui lòng check mail để kiểm tra thông tin"	Pass
FUNC_05_24	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Mã ưu đãi	1. Nhập toàn ký tự khoảng trắng cho trường Mã ưu đãi 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo "Đăng ký thành công, bạn vui lòng check mail để kiểm tra thông tin"	Pass
FUNC_05_25	Kiểm tra nhập dữ liệu không hợp lệ cho trường Mã ưu đãi	1. Nhập dữ liệu không hợp lệ cho trường Mã ưu đãi 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Đăng ký	3. Hiện thị thông báo lỗi "Mã khuyến mãi không hợp lệ"	Pass
FUNC_05_26	Kiểm tra đăng ký khóa học thành	1. Nhập dữ liệu hợp lệ cho các trường	2. Hiện thị thông báo "Đăng ký	Pass

	công	2. Nhấn nút Đăng ký	thành công, bạn vui lòng check mail để kiểm tra thông tin"	
--	------	---------------------	--	--

Bảng 3.13: Bảng các ca kiểm thử cho “Cập nhật thông tin cá nhân”

ID	Mục tiêu kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
FUNC_06_01	Kiểm tra không nhập trường Mô tả về bạn	1. Không nhập trường Mô tả về bạn 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập mô tả về bạn"	Fail
FUNC_06_02	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Mô tả về bạn	1. Nhập toàn ký tự khoảng trắng cho trường Mô tả về bạn 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập mô tả về bạn"	Fail
FUNC_06_03	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Mô tả về bạn	1. Nhập giá trị có chứa ký tự chữ cho trường Mô tả về bạn 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass

FUNC_06_04	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng cho trường Mô tả về bạn	1. Nhập giá trị có chứa ký tự khoảng trắng cho trường Mô tả về bạn 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_05	Kiểm tra nhập giá trị có chứa ký tự số cho trường Mô tả về bạn	1. Nhập giá trị có chứa ký tự số cho trường Mô tả về bạn 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_06	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Mô tả về bạn	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Mô tả về bạn 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_07	Kiểm tra không nhập trường Chọn username riêng	1. Không nhập trường Chọn username riêng 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập username riêng"	Fail
FUNC_06_08	Kiểm tra nhập toàn ký tự khoảng trắng	1. Nhập toàn ký tự khoảng trắng cho trường Chọn username riêng 2. Nhập dữ liệu hợp lệ cho các	3. Hiện thị thông báo lỗi "Vui lòng	Fail

	cho trường Chọn username riêng	trường khác 3. Nhấn nút Lưu	nhập username riêng"	
FUNC_06_09	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Chọn username riêng	1. Nhập giá trị có chứa ký tự chữ cho trường Chọn username riêng 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_10	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng cho trường Chọn username riêng	1. Nhập giá trị có chứa ký tự khoảng trắng cho trường Chọn username riêng 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_11	Kiểm tra nhập giá trị có chứa ký tự số cho trường Chọn username riêng	1. Nhập giá trị có chứa ký tự số cho trường Chọn username riêng 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_12	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Chọn username riêng	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Chọn username riêng 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass

FUNC_06_13	Kiểm tra không nhập trường Trường học	1. Không nhập trường Trường học 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập trường học"	Fail
FUNC_06_14	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Trường học	1. Nhập toàn ký tự khoảng trắng cho trường Trường học 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập trường học"	Fail
FUNC_06_15	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Trường học	1. Nhập giá trị có chứa ký tự chữ cho trường Trường học 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_16	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng cho trường Trường học	1. Nhập giá trị có chứa ký tự khoảng trắng cho trường Trường học 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_17	Kiểm tra nhập giá trị có chứa ký tự số cho	1. Nhập giá trị có chứa ký tự số cho trường Trường học 2. Nhập dữ liệu hợp lệ cho các	3. Hiện thị thông báo "Cập nhật	Pass

	trường Trường học	trường khác 3. Nhấn nút Lưu	thành công"	
FUNC_06_18	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Trường học	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Trường học 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_19	Kiểm tra không nhập trường Nơi làm việc	1. Không nhập trường Nơi làm việc 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập nơi làm việc"	Fail
FUNC_06_20	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Nơi làm việc	1. Nhập toàn ký tự khoảng trắng cho trường Nơi làm việc 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập nơi làm việc"	Fail
FUNC_06_21	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Nơi làm việc	1. Nhập giá trị có chứa ký tự chữ cho trường Nơi làm việc 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass

FUNC_06_22	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng cho trường Nơi làm việc	1. Nhập giá trị có chứa ký tự khoảng trắng cho trường Nơi làm việc 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_23	Kiểm tra nhập giá trị có chứa ký tự số cho trường Nơi làm việc	1. Nhập giá trị có chứa ký tự số cho trường Nơi làm việc 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_24	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Nơi làm việc	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Nơi làm việc 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_25	Kiểm tra không nhập trường Địa điểm	1. Không nhập trường Địa điểm 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo lỗi "Vui lòng nhập địa điểm"	Fail
FUNC_06_26	Kiểm tra nhập toàn ký tự khoảng trắng cho trường Địa	1. Nhập toàn ký tự khoảng trắng cho trường Địa điểm 2. Nhập dữ liệu hợp lệ cho các trường khác	3. Hiện thị thông báo lỗi "Vui lòng nhập địa	Fail

	điểm	3. Nhấn nút Lưu	điểm"	
FUNC_06_27	Kiểm tra nhập giá trị có chứa ký tự chữ cho trường Địa điểm	1. Nhập giá trị có chứa ký tự chữ cho trường Địa điểm 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_28	Kiểm tra nhập giá trị có chứa ký tự khoảng trắng cho trường Địa điểm	1. Nhập giá trị có chứa ký tự khoảng trắng cho trường Địa điểm 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_29	Kiểm tra nhập giá trị có chứa ký tự số cho trường Địa điểm	1. Nhập giá trị có chứa ký tự số cho trường Địa điểm 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass
FUNC_06_30	Kiểm tra nhập giá trị có chứa ký tự đặc biệt cho trường Địa điểm	1. Nhập giá trị có chứa ký tự đặc biệt cho trường Địa điểm 2. Nhập dữ liệu hợp lệ cho các trường khác 3. Nhấn nút Lưu	3. Hiện thị thông báo "Cập nhật thành công"	Pass

Bảng 3.14: Bảng các ca kiểm thử cho “Đăng xuất”

ID	Mục tiêu kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
FUNC_07_01	Kiểm tra đăng xuất thành công	1. Nhấn vào tên đăng nhập 2. Nhấn chọn Đăng xuất	2. Đăng xuất thành công	Pass

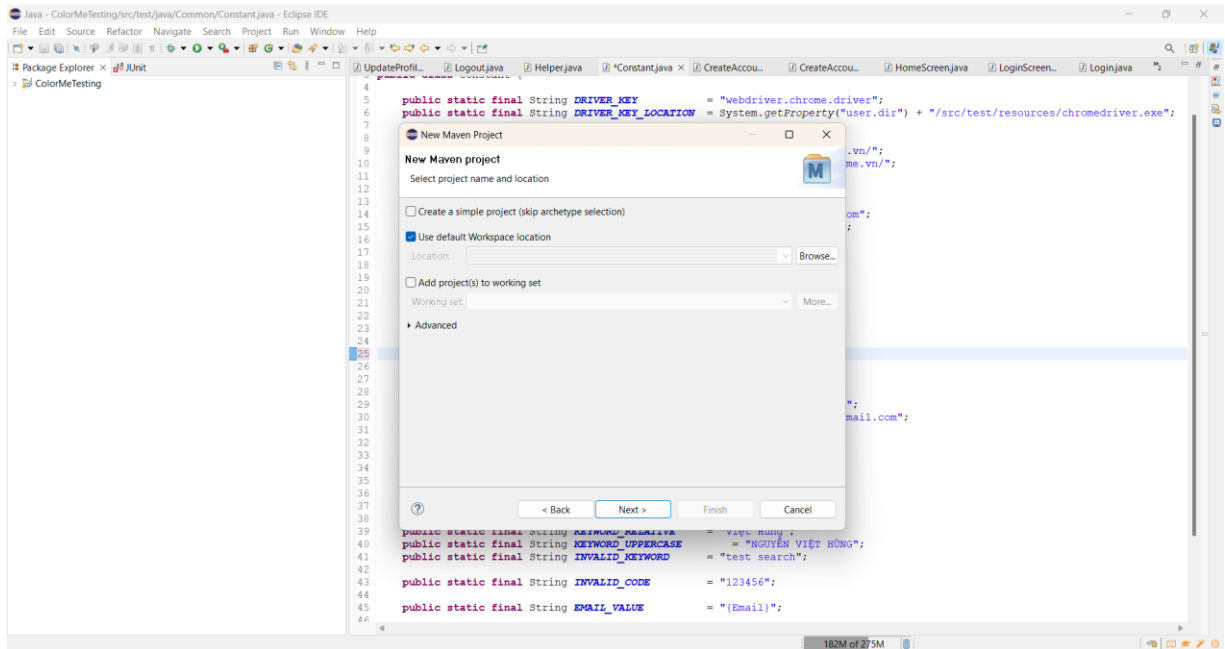
3.3 Thực hiện kiểm thử

Để xây dựng test script, ban đầu người dùng cần phải tải xuống bản webdriver tương thích với mỗi browser để có thể chạy được các auto testscript. Download công cụ Eclipse để thực hiện xây dựng và thực thi câu lệnh (tùy vào nhu cầu mỗi người dùng, có thể viết và thực thi câu lệnh trên các công cụ khác nhau có hỗ trợ ngôn ngữ và thư viện java).

Với mỗi browser sẽ có một bản webdriver riêng như với Chrome - ChromeDriver, với Firefox - GeckoDriver, ...

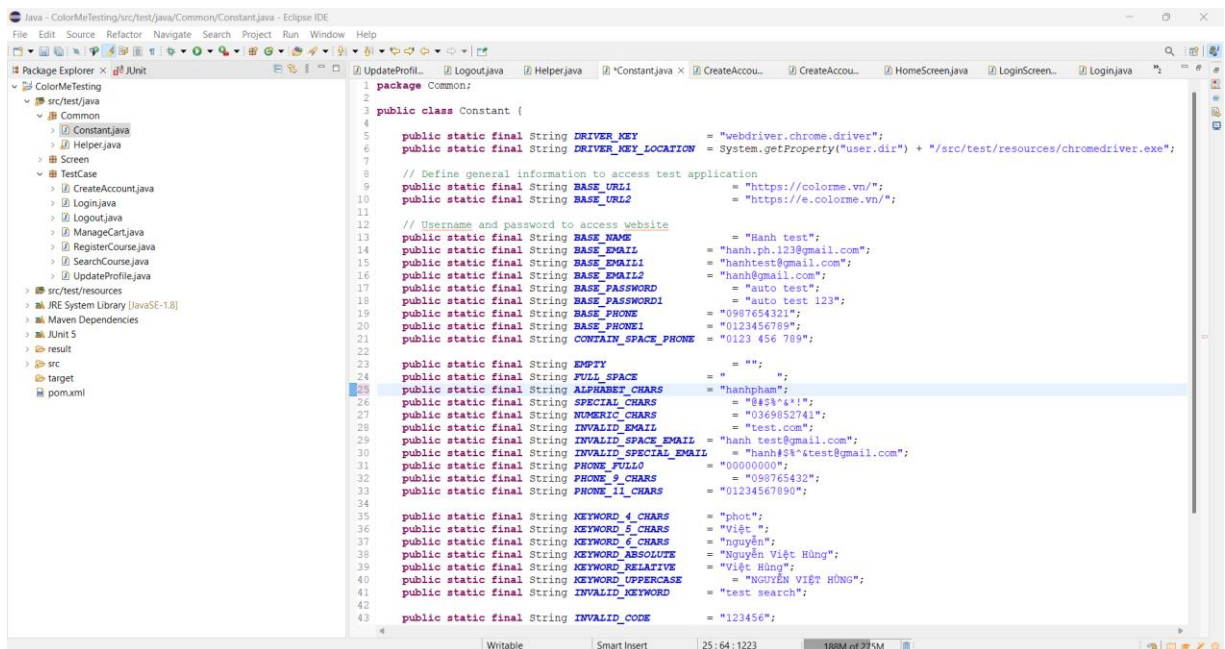
❖ Các bước triển khai:

- Bước 1: Tạo mới một project: chọn File → New → Project → Thư mục Maven → Maven Project → Nhập các thông tin → Nhấn Finish



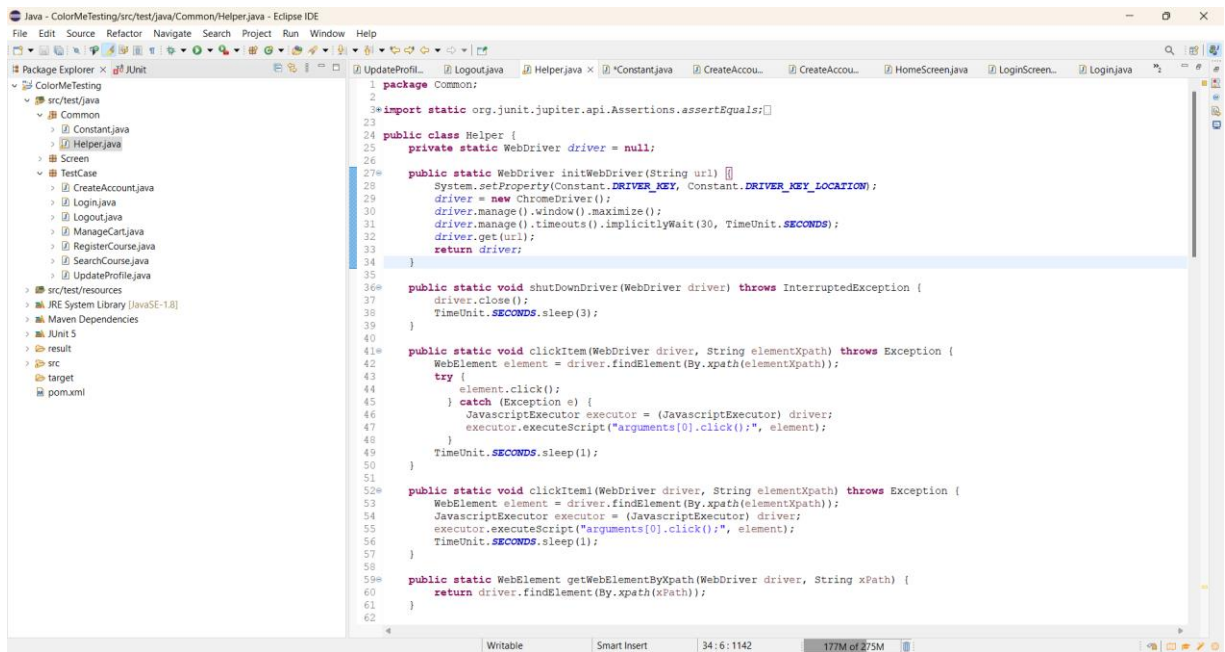
Hình 3.1: Màn hình tạo Maven project

- Bước 2: Tạo package Base để lưu trữ các class Constant chứa hằng số chung và class Helper chứa phương thức chung



Hình 3.2: Màn hình Class Constant

Class constant này được tạo ra nhằm mục đích lưu giữ dữ liệu đầu vào cho website Colorme.vn. Để cho phép công cụ Selenium truy xuất dữ liệu kiểm thử, dữ liệu này có thể được sửa đổi mà không cần thay đổi mã nguồn trong các tệp khác để giúp hệ thống chạy hiệu quả hơn.

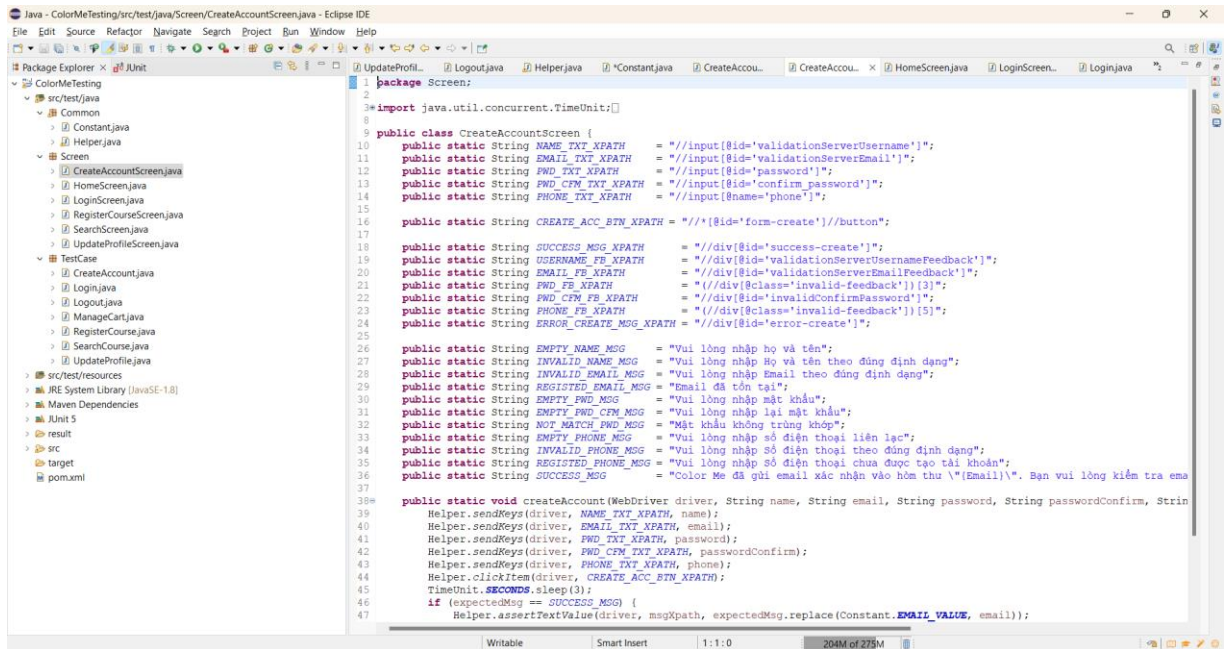


Hình 3.3: Màn hình Class Helper

- Class Helper được tạo ra để chứa các phương thức chung như là xử lý chuỗi, xử lý ngày tháng, chuyển đổi dữ liệu, tạo định dạng, kiểm tra điều kiện, và các tác vụ chung khác giúp tăng tính rõ ràng, ngắn gọn và dễ hiểu của các câu lệnh trong mã nguồn.

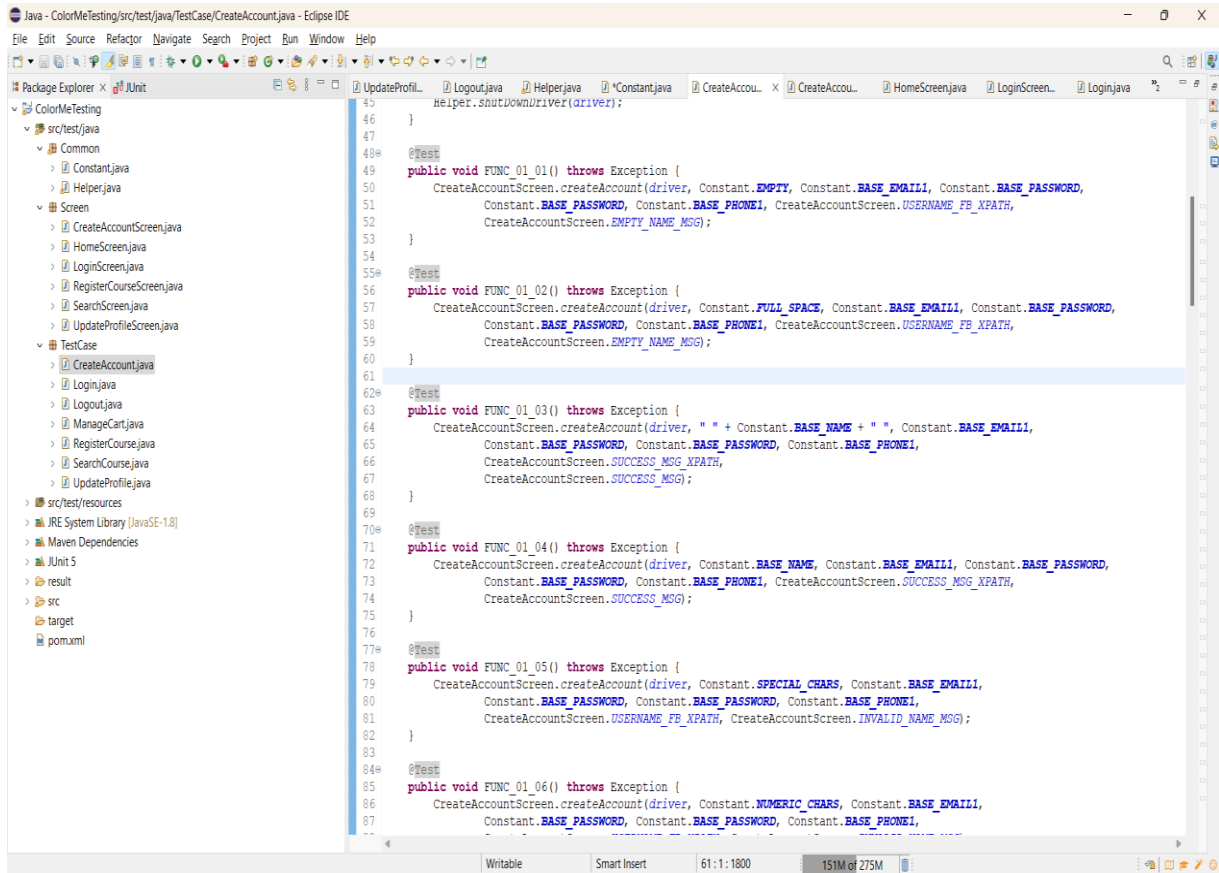
- Thông qua việc đóng gói các phương thức chung vào một class Helper, chúng ta có thể tái sử dụng các phương thức đó ở nhiều nơi khác nhau trong mã nguồn mà không cần phải viết lại từ đầu.

- Bước 3: Tạo package Screen để lưu trữ các class của từng màn hình mà chúng ta sẽ thao tác để chứa các đường dẫn xpath và hàm thao tác kiểm tra website trên màn hình tương ứng




Hình 3.4: Màn hình Class CreateAccountScreen trong package Screen

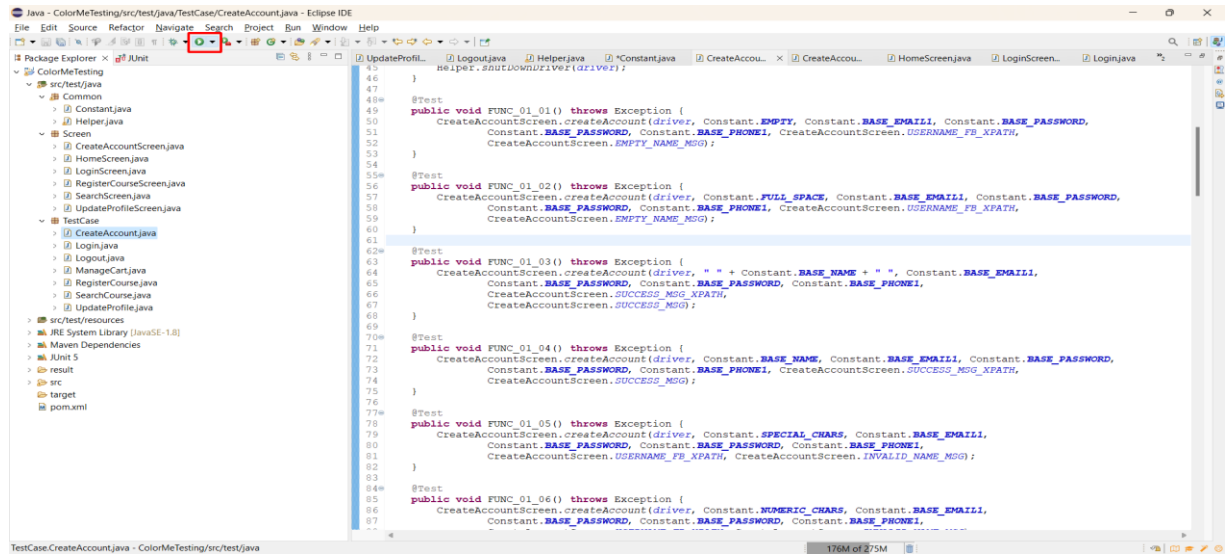
- Mục tiêu của việc sử dụng Package Screen là để lưu trữ các chức năng liên quan đến từng màn hình thành các lớp riêng biệt và tái sử dụng chúng trong các kịch bản kiểm thử. Điều này giúp tách biệt logic kiểm thử từ giao diện người dùng, cung cấp một cấu trúc rõ ràng cho kiểm thử và tăng tính bảo trì và tái sử dụng của mã kiểm thử.
- Bước 4: Tạo package TestCase để lưu trữ các class để chứa các ca kiểm thử của từng module



Hình 3.5: Màn hình Class CreateAccount trong Package TestCase

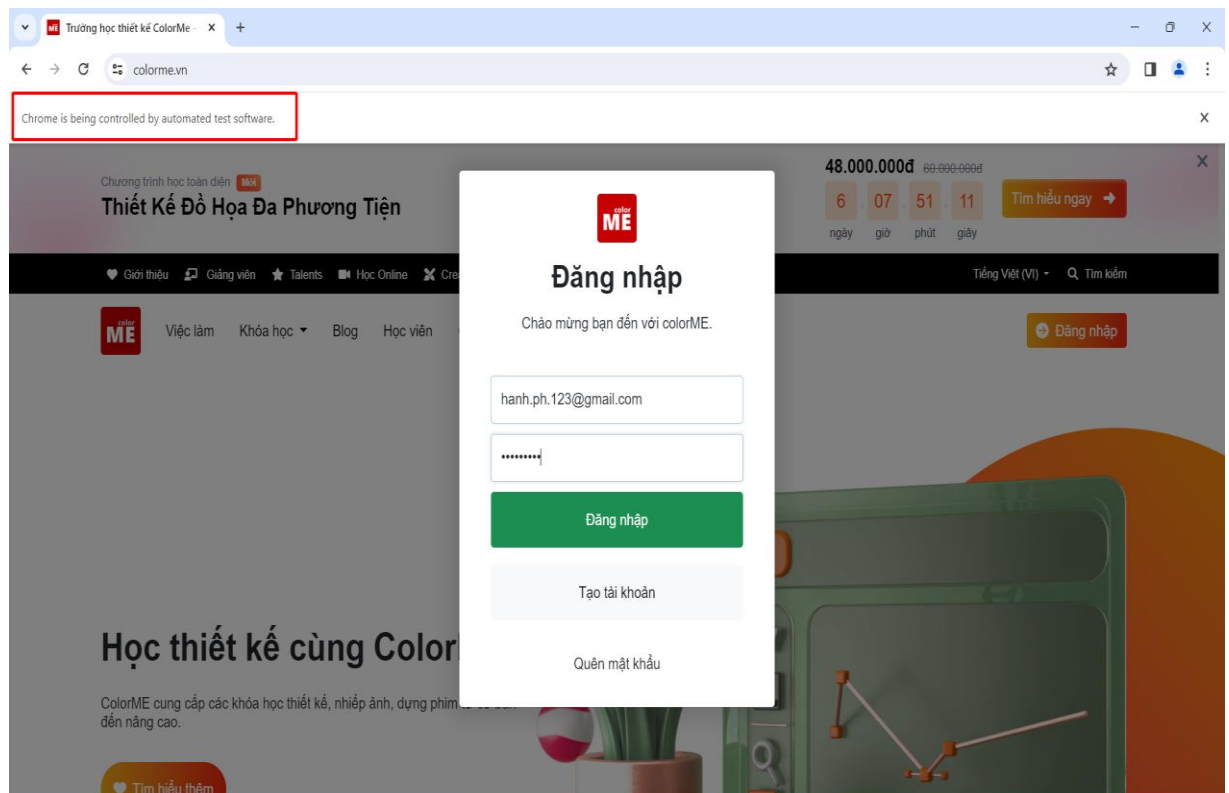
Mục tiêu của việc sử dụng Package TestCase và các lớp chức năng là để tạo ra một bộ kiểm thử toàn diện cho hệ thống, bao gồm các ca kiểm thử đại diện cho các chức năng chính. Điều này giúp đảm bảo tính đáng tin cậy và chất lượng của hệ thống thông qua việc xác định và khắc phục các lỗi tiềm ẩn và đảm bảo rằng chức năng hoạt động như mong đợi trong mọi trường hợp.

- Bước 5: Để thực hiện kiểm thử chức năng thì mở file chứa bộ ca kiểm thử của chức năng cần chạy rồi nhấn nút  trên thanh menu

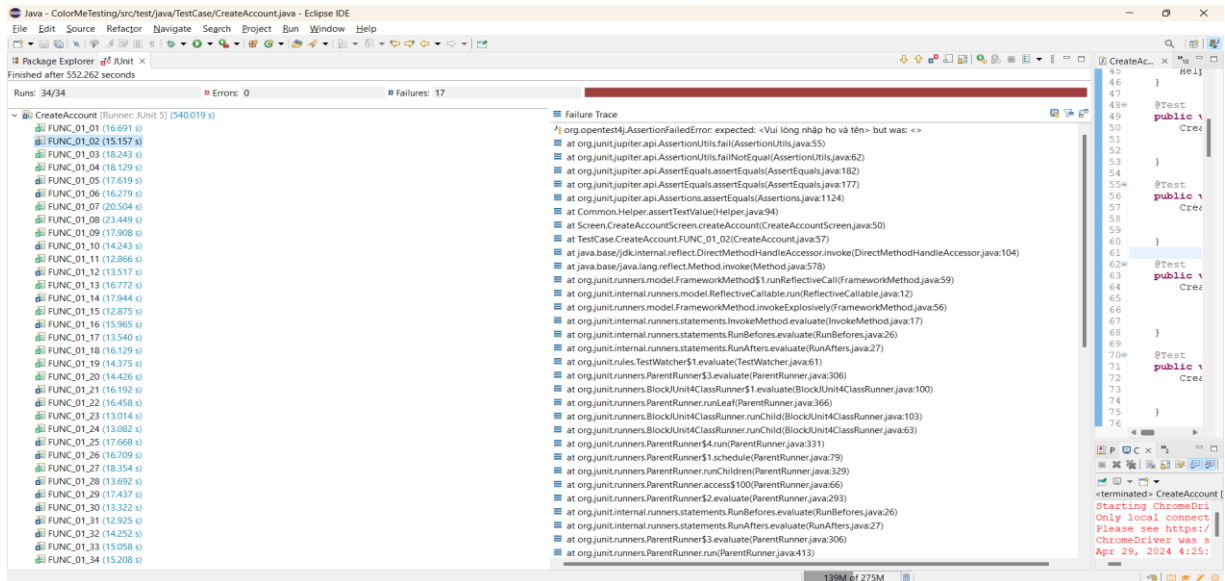


Hình 3.6: Màn hình trở nút chạy chương trình trên Eclipse

Sau khi chạy kiểm thử xong, sẽ có báo cáo kết quả kiểm thử như sau:



Hình 3.7: Màn hình khi Selenium WebDriver chạy ca kiểm thử



Hình 3.8: Màn hình kết quả sau khi chạy kiểm thử xong

- Sau khi hoàn thành bộ ca kiểm thử, màn hình kết quả được hiển thị. Trên màn hình này, chúng ta sẽ thấy các thông tin tổng quan về kết quả của các ca kiểm thử đã được thực hiện. Màn hình kết quả cung cấp một cái nhìn tổng quan về tình trạng và hiệu suất của hệ thống sau quá trình kiểm thử.

- Màn hình kết quả bao gồm các thông tin sau:

- Số lượng ca kiểm thử đã chạy, số lượng ca kiểm thử thất bại và số lượng ca kiểm thử lỗi.
- Thời gian tổng cộng để chạy bộ ca kiểm thử.
- Chi tiết kết quả của từng ca kiểm thử, bao gồm thông tin về thành công, thất bại, hoặc lỗi (nếu có).
- Báo cáo tổng quan về hiệu suất của hệ thống, bao gồm thời gian phản hồi, tài nguyên sử dụng, và các thông số khác liên quan đến độ tin cậy và hiệu suất của ứng dụng.

3.4 Tổng hợp kết quả kiểm thử

Sau khi thực hiện kiểm thử toàn bộ test case đã thiết kế ở trên thì em đã tổng hợp lại được hai bảng kết quả kiểm thử bằng phương pháp thủ công và phương pháp tự động như sau:

Bảng 3.15: Bảng kết quả kiểm thử thủ công

	Tổng số TCs	Số TCs Pass	Số TCs Fail
Tạo tài khoản	34	19	15
Đăng nhập	12	8	4
Tìm kiếm thông tin	10	8	2
Quản lý giỏ hàng	5	5	0
Đăng ký khóa học	26	13	13
Cập nhật thông tin cá nhân	30	20	10
Đăng xuất	1	1	0
TỔNG	118	74	44

Bảng 3.16: Bảng kết quả kiểm thử tự động

	Tổng số TCs	Số TCs Pass	Số TCs Fail
Tạo tài khoản	34	19	15
Đăng nhập	12	8	4
Tìm kiếm thông tin	10	8	2

Quản lý giỏ hàng	5	5	0
Đăng ký khóa học	26	13	13
Cập nhật thông tin cá nhân	30	20	10
Đăng xuất	1	1	0
TỔNG	118	74	44

3.5 Đánh giá thực nghiệm

Sau khi thực hiện kiểm thử một vài chức năng chính của website bán khóa học thì em đã tổng hợp được bảng so sánh thời gian thực hiện kiểm thử thủ công và tự động như sau:

Bảng 3.17: Bảng so sánh thời gian thực hiện kiểm thử thủ công và tự động

	Số test case	Thời gian thực hiện thủ công	Thời gian thực hiện tự động
Tạo tài khoản	34	3 giờ 30 phút	15 phút
Đăng nhập	12	1 giờ 15 phút	5 phút
Tìm kiếm thông tin	10	1 giờ	5 phút
Quản lý giỏ hàng	5	30 phút	2 phút
Đăng ký khóa học	26	2 giờ 30 phút	12 phút
Cập nhật thông tin cá nhân	30	3 giờ	13 phút
Đăng xuất	1	5 phút	30 giây

Sau cả quá trình kiểm thử thì bản thân em đã nhận thấy được một số ưu nhược điểm của cả hai cách kiểm thử thủ công và tự động như sau:

Bảng 3.18: Bảng so sánh ưu nhược điểm của kiểm thử thủ công và tự động

	Kiểm thử thủ công	Kiểm thử tự động
Ưu điểm	Phát hiện các lỗi giao diện nhanh	Thời gian test ngắn
	Thực hiện được các case exception	Tiết kiệm nguồn lực
	Không yêu cầu tester có kỹ năng lập trình	Có khả năng tái sử dụng nhiều lần
	Các thay đổi nhỏ của cấu trúc website như class, id không làm ảnh hưởng đến quá trình kiểm thử	Sử dụng công cụ test không biết mệt mỏi
Nhược điểm	Tốn thời gian chạy testcase, tiến độ bị chậm	Đòi hỏi tester phải có kiến thức lập trình
	Tốn nhân lực	Không thích hợp với dự án nhỏ, số lượng testcase ít
	Lỗi do con người thao tác sai dẫn đến kết quả test không chính xác	Dễ bị ảnh hưởng bởi source code của sản phẩm

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

❖ Kết luận

Qua thời gian thực hiện, dưới sự hướng dẫn tận tình của thầy Nguyễn Mạnh Cường cùng sự giúp đỡ rất nhiều của các quý thầy cô trong khoa, em đã cố gắng hoàn thành đồ án đúng theo yêu cầu và thời gian quy định. Trong đồ án, em đã thực hiện được những công việc sau:

Tìm hiểu về quy trình kiểm thử phần mềm, vai trò, mục tiêu và các kỹ thuật kiểm thử phần mềm.

Tìm hiểu về kiểm thử chức năng, các cách tiếp cận, quy trình và phương pháp kiểm thử chức năng.

Tạo các test case và kiểm thử chức năng thủ công và tự động cho *Website bán khóa học*.

Qua quá trình thử nghiệm kiểm thử thực tiễn đã giúp em hiểu rõ hơn được về ưu và nhược điểm giữa manual testing và automation testing. Từ đó, biết định hướng sắp xếp, đan xen và kết hợp giữa 2 loại kiểm thử với nhau, giúp tăng năng suất trong công việc hơn.

❖ Hướng phát triển

Hướng phát triển của em trong tương lai gần là tìm hiểu về các loại test (ví dụ như kiểm thử api, kiểm thử giao diện, kiểm thử hiệu năng, ...); áp dụng các kiểu kiểm thử khác nhau vào các sản phẩm thực tiễn, mở rộng thêm lĩnh vực kiểm thử từ đó đúc rút thêm được kinh nghiệm kiểm thử phần mềm cũng như kinh nghiệm làm việc.

Tìm hiểu chuyên sâu hơn và thực hiện kiểm thử tự động để có thể hoàn thành tốt việc kiểm thử của một trang website hoàn chỉnh hơn và đạt được các kết quả tốt hơn. Từ đó, ngày càng cải thiện và trau dồi các kỹ năng viết test

script, kỹ thuật về kiểm thử chức năng cũng như trau dồi kinh nghiệm sử dụng các công cụ kiểm thử tự động và tăng thêm kiến thức cho bản thân.

Trở thành một chuyên viên kiểm thử phần mềm về cả hai mảng manual testing và automation testing.

TÀI LIỆU THAM KHẢO

- [1] VIBLO (2019). “Các kỹ thuật kiểm thử phần mềm”. Truy cập từ: <https://viblo.asia/p/cac-ky-thuat-kiem-thu-phan-mem-gGJ599eG5X2> ngày 30/04/2024.
- [2] QA acharya - School Of Software Testing (2024). Testcase Examples. Truy cập từ <https://www.qaacharya.in/> ngày 30/04/2024.
- [3] Tài liệu môn học Kiểm thử phần mềm, NXB Đại học Công Nghiệp Hà Nội
- [4] TS.Nguyễn Mạnh Cường “Tài liệu hướng dẫn làm báo cáo tốt nghiệp khoa Công nghệ thông tin”, Trường Đại học Công Nghiệp Hà Nội, năm 2024.