

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN

NGUYỄN THỊ THANH HUYỀN

XÂY DỰNG FRAMEWORK KIỂM THỬ TỰ ĐỘNG VÀ
ỨNG DỤNG KIỂM THỬ CHO WEBSITE SIMPLILEARN

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

HƯNG YÊN - 2023

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HÙNG YÊN

NGUYỄN THỊ THANH HUYỀN

XÂY DỰNG FRAMEWORK KIỂM THỬ TỰ ĐỘNG VÀ ỨNG
DỤNG KIỂM THỬ CHO WEBSITE SIMPLILEARN

NGÀNH: CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: KỸ THUẬT PHẦN MỀM

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

NGƯỜI HƯỚNG DẪN
THS. ĐỖ THỊ THU TRANG

HÙNG YÊN - 2023

NHẬN XÉT

Nhận xét của giảng viên hướng dẫn:

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

GIẢNG VIÊN HƯỚNG DẪN

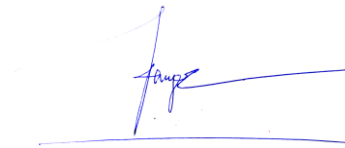
(Ký và ghi rõ họ tên)

LỜI CAM ĐOAN

Em xin cam đoan đồ án tốt nghiệp “Xây dựng framework kiểm thử tự động và ứng dụng kiểm thử cho website Simplilearn” là công trình nghiên cứu của bản thân. Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các số liệu, kết quả trình bày trong đồ án là hoàn toàn trung thực, nếu sai em xin chịu hoàn toàn trách nhiệm và chịu mọi kỷ luật của Bộ môn và Nhà trường đề ra.

Hưng Yên, ngày 26 tháng 06 năm 2023.

Sinh viên



Nguyễn Thị Thanh Huyền

LỜI CẢM ƠN

Để có thể hoàn thành đồ án này, đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Công nghệ phần mềm, Khoa Công nghệ thông tin - Trường Đại học Sư phạm Kỹ thuật Hưng Yên đã tạo điều kiện thuận lợi cho em thực hiện đồ án này.

Em muốn bày tỏ sự biết ơn và trân trọng đối với sự hướng dẫn và hỗ trợ mà cô Đỗ Thị Thu Trang đã cung cấp trong suốt quá trình thực hiện đồ án. Cô đã chia sẻ những kinh nghiệm thực tiễn, cung cấp tư vấn và hỗ trợ trong quá trình nghiên cứu, phân tích, thiết kế, và triển khai dự án. Nhờ sự tận tâm và kiến thức chuyên môn của cô, em đã có cơ hội tiếp cận và áp dụng những kiến thức Kiểm thử phần mềm vào dự án của mình một cách hiệu quả.

Đồng thời, Em xin gửi lời cảm ơn chân thành và lòng biết ơn sâu sắc đến Trường và tất cả các Thầy, các Cô vì luôn tạo điều kiện thuận lợi trong quá trình học tập của chúng em. Trường đã không ngừng đầu tư vào cơ sở vật chất, công nghệ và tài nguyên, tạo nên một môi trường học tập hiện đại và đáp ứng đầy đủ yêu cầu của ngành nghề. Sự tận tâm và trang bị kiến thức cần thiết, quý báu từ các Thầy, các Cô đã giúp em thực hiện thành công đồ án này.

Mặc dù em đã có cố gắng, nhưng với kiến thức còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy giáo, Cô giáo về những kết quả triển khai trong đồ án.

Em xin trân trọng cảm ơn!

MỤC LỤC

MỤC LỤC	5
DANH SÁCH CÁC THUẬT NGỮ	7
DANH SÁCH BẢNG BIỂU	8
CHƯƠNG 1: MỞ ĐẦU	11
1.1 Lý do chọn đề án.....	11
1.2 Mục tiêu của đề án.....	12
1.2.1 Mục tiêu tổng quát.....	12
1.2.2 Mục tiêu cụ thể	12
1.3 Giới hạn và phạm vi của đề án	13
1.3.1 Đối tượng nghiên cứu	13
1.3.2 Phạm vi nghiên cứu	13
1.4 Nội dung thực hiện	13
1.5 Phương pháp tiếp cận	14
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	15
2.1 Tổng quan về kiểm thử phần mềm	15
2.1.1. Các phương pháp kiểm thử phần mềm:.....	15
2.1.2. Vai trò của kiểm thử phần mềm	15
2.1.3. Quy trình kiểm thử phần mềm.....	15
2.2 Kiểm thử tự động phần mềm	20
2.3 Kiểm thử chức năng.....	27
2.4 Giới thiệu framework kiểm thử tự động	28
2.4.1 Tổng quan về framework kiểm thử tự động	28
2.4.2 Hybrid driven testing framework	29
2.5 Công nghệ áp dụng	31
CHƯƠNG 3: ĐẶC TẢ HỆ THỐNG PHẦN MỀM	36
3.1. Giới thiệu về hệ thống phần mềm	36
3.2. Các yêu cầu chức năng.....	36
3.2.1. Chức năng tổng quát	36
3.2.2. Chức năng đăng nhập	37

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

3.2.3.	Chức năng đăng ký	39
3.2.4.	Chức năng tìm kiếm.....	42
3.2.5.	Chức năng đăng ký khóa học.....	43
3.3.	Các yêu cầu phi chức năng	48
CHƯƠNG 4: TRIỂN KHAI KIỂM THỬ TỰ ĐỘNG		49
4.1.	Phương pháp xây dựng Framework.....	49
4.2.	Thiết kế các yêu cầu kiểm thử	50
4.3.	Xây dựng ca kiểm thử.....	56
4.4.	Xây dựng dữ liệu kiểm thử	67
4.5.	Thực thi và báo cáo kiểm thử.....	71
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....		73
TÀI LIỆU THAM KHẢO		75

DANH SÁCH CÁC THUẬT NGỮ

Từ viết tắt	Từ đầy đủ	Giải thích
HTML	HyperText Markup Language	Ngôn ngữ Đánh dấu Siêu văn bản
IDE	Integrated Development Environment	Môi trường phát triển tích hợp
IDE	Integrated Development Environment	Môi trường phát triển tích hợp
ISO	International Organization for Standardization	Tổ chức tiêu chuẩn hóa quốc tế
Q&A	Question and Answer	Câu hỏi và trả lời
QA	Quality Assurance	Đảm bảo chất lượng
QTP	QuickTest Professional	
SRS	Software Requirement Specification	Đặc tả yêu cầu phần mềm
UFT	Unified Functional Testing	

DANH SÁCH BẢNG BIỂU

<i>Bảng 3. 1 Mô tả cho usecase tổng quát.....</i>	<i>37</i>
<i>Bảng 3. 2 Dòng sự kiện chính- chức năng đăng nhập</i>	<i>38</i>
<i>Bảng 3. 3 Validate data chức năng đăng nhập</i>	<i>39</i>
<i>Bảng 3. 4 Nội dung thông báo chức năng đăng nhập.....</i>	<i>39</i>
<i>Bảng 3. 5 Dòng sự kiện chính chức năng đăng ký</i>	<i>40</i>
<i>Bảng 3. 6 Validate data chức năng đăng ký.....</i>	<i>41</i>
<i>Bảng 3. 7 Nội dung thông báo chức năng đăng ký</i>	<i>41</i>
<i>Bảng 3. 8 Dòng sự kiện chính chức năng tìm kiếm.....</i>	<i>43</i>
<i>Bảng 3. 9 Validate data chức năng tìm kiếm.....</i>	<i>43</i>
<i>Bảng 3. 10 Nội dung thông báo chức năng tìm kiếm</i>	<i>43</i>
<i>Bảng 3. 11 Dòng sự kiện chính chức năng đăng ký.....</i>	<i>44</i>
<i>Bảng 3. 12 Validate data chức năng đăng ký khóa học</i>	<i>45</i>
<i>Bảng 3. 13 Validate data biểu mẫu đăng ký khóa học</i>	<i>46</i>
<i>Bảng 3. 14 Nội dung thông báo chức năng đăng ký khóa học.....</i>	<i>47</i>
<i>Bảng 3. 15 Các yêu cầu phi chức năng.....</i>	<i>48</i>
<i>Bảng 4. 1 Trường hợp kiểm thử chức năng đăng ký.....</i>	<i>56</i>
<i>Bảng 4. 2 Dữ liệu kiểm thử chức năng đăng ký.....</i>	<i>67</i>
<i>Bảng 4. 3 Kịch bản kiểm thử chức năng đăng ký.....</i>	<i>69</i>

DANH SÁCH HÌNH VẼ

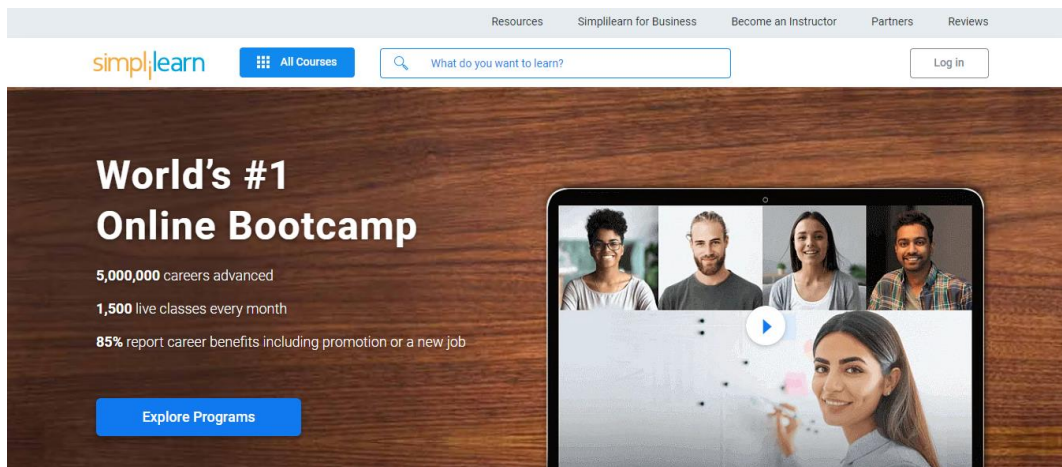
<i>Hình 1. 1 Trang chủ website Simplilearn.....</i>	<i>11</i>
<i>Hình 2. 1 Quy trình kiểm thử phần mềm.....</i>	<i>16</i>
<i>Hình 2. 2 Giới thiệu kiểm thử tự động phần mềm.....</i>	<i>20</i>
<i>Hình 2. 3 Quy trình kiểm thử chức năng.....</i>	<i>27</i>
<i>Hình 2. 4 Quy trình hoạt động Data Driven Testing Framework.....</i>	<i>29</i>
<i>Hình 2. 5 Quy trình hoạt động Keyword-Driven Testing Framework.....</i>	<i>30</i>
<i>Hình 2. 6 Giới thiệu tổng quan về Selenium WebDriver.....</i>	<i>31</i>
<i>Hình 2. 7 Giới thiệu tổng quan về TestNG.....</i>	<i>33</i>
<i>Hình 2. 8 Giới thiệu tổng quan về Eclipse</i>	<i>34</i>
<i>Hình 3. 1 Usecase tổng quát chức năng.....</i>	<i>37</i>
<i>Hình 3. 2 Use case chức năng đăng nhập.....</i>	<i>38</i>
<i>Hình 3. 3 Usecase chức năng đăng ký</i>	<i>40</i>
<i>Hình 3. 4 Use case chức năng tìm kiếm</i>	<i>42</i>
<i>Hình 3. 5 Usecase chức năng đăng ký khóa học.....</i>	<i>44</i>
<i>Hình 3. 6 Luồng nghiệp vụ chức năng đăng ký khóa học</i>	<i>47</i>
<i>Hình 4. 1 Cấu trúc Framework kiểm thử tự động.....</i>	<i>49</i>
<i>Hình 4. 2 Thiết kế kiểm thử tổng quan hệ thống</i>	<i>50</i>
<i>Hình 4. 3 Thiết kế kiểm thử chức năng đăng nhập.....</i>	<i>52</i>
<i>Hình 4. 4 Thiết kế kiểm thử chức năng đăng ký.....</i>	<i>53</i>
<i>Hình 4. 5 Thiết kế kiểm thử chức năng tìm kiếm.....</i>	<i>54</i>
<i>Hình 4. 6 Thiết kế kiểm thử hiệu năng.....</i>	<i>55</i>

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

<i>Hình 4. 7 Test report (ExtentReport.html) của chức năng đăng ký</i>	<i>71</i>
<i>Hình 4. 8 Ghi trạng thái testcase vào file log.....</i>	<i>72</i>
<i>Hình 4. 9 Ảnh chụp màn hình lưu tại folder Screenshots khi testcase fail.....</i>	<i>72</i>

CHƯƠNG 1: MỞ ĐẦU

1.1 Lý do chọn đề án



Hình 1. 1 Trang chủ website Simplilearn

Trong thế kỷ 21, sự phát triển của website, đặc biệt là các website dạy học trực tuyến, đã đem lại nhiều tiềm năng và thay đổi đáng kể cách chúng ta tiếp cận và tham gia vào quá trình học tập. Các website dạy học trực tuyến đã mở ra cơ hội học tập cho mọi người trên toàn cầu. Bất kể nơi bạn đang sinh sống, chỉ cần có kết nối internet, bạn có thể tiếp cận với các khóa học và tài liệu từ các giảng viên và chuyên gia hàng đầu trên thế giới. Điều này giúp phá vỡ các rào cản về địa lý và thời gian, tạo điều kiện thuận lợi cho việc học tập từ xa và tự học. Website dạy học trực tuyến cung cấp một loạt các khóa học và nguồn tài liệu đa dạng trong nhiều lĩnh vực khác nhau. Từ kỹ năng mềm đến chuyên môn kỹ thuật, từ ngôn ngữ nước ngoài đến khoa học dữ liệu, bạn có thể tìm thấy những khóa học phù hợp với sở thích và mục tiêu học tập của mình. Điều này tạo ra sự linh hoạt và lựa chọn cho người học. Một số website dạy học trực tuyến cung cấp các công cụ tương tác như video học, bài tập thực hành.

Simplilearn là nhà cung cấp dịch vụ đào tạo chứng chỉ hàng đầu trên thế giới. Simplilearn đã đào tạo hơn 1,000,000 chuyên gia và họ có hơn 2,000 giảng viên đủ điều kiện cùng với hơn 400 khóa học và hơn 40 chứng nhận toàn cầu.

Họ cung cấp đào tạo trực tuyến trong các lĩnh vực như an ninh mạng, điện toán đám mây, quản lý dự án, tiếp thị kỹ thuật số và khoa học dữ liệu, nơi công nghệ và các

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

phương pháp hay nhất đang phát triển nhanh chóng và nhu cầu về các ứng viên đủ điều kiện được đáp ứng một cách hoàn hảo.

Trong những năm gần đây, nhu cầu về học tập trực tuyến đã tăng mạnh. Việc xây dựng framework và kiểm thử tự động cho ứng dụng website Simplilearn sẽ giúp đảm bảo rằng nền tảng học tập này hoạt động một cách ổn định, mượt mà và mang lại trải nghiệm tốt cho người dùng. Một framework và quy trình kiểm thử tự động có thể giúp tăng cường hiệu suất và hiệu quả phát triển. Cho phép tự động hóa quy trình kiểm thử, giảm thời gian và công sức cần thiết để thực hiện kiểm thử thủ công. Điều này giúp tăng cường năng suất và giảm thời gian phát triển, đồng thời đảm bảo chất lượng của ứng dụng.

1.2 Mục tiêu của đề án

1.2.1 Mục tiêu tổng quát

Thực hiện xây dựng framework và kiểm thử tự động cho ứng dụng website Simplilearn. Mục tiêu chính là đảm bảo rằng website Simplilearn hoạt động một cách ổn định và tin cậy. Framework và quy trình kiểm thử tự động sẽ giúp phát hiện lỗi và vấn đề sớm, từ đó cải thiện tính ổn định của hệ thống và đảm bảo người dùng không gặp phải các sự cố không mong muốn, đảm bảo rằng các chức năng hoạt động đúng và đáp ứng yêu cầu của người dùng một cách chính xác và nhanh chóng.

1.2.2 Mục tiêu cụ thể

Nghiên cứu kiểm tra và đảm bảo rằng các chức năng của ứng dụng hoạt động chính xác theo yêu cầu, xác nhận rằng các chức năng của ứng dụng hoạt động một cách đúng và liên tục trong các tình huống khác nhau. Đảm bảo rằng các tính năng không gặp lỗi và hoạt động ổn định dưới tải công việc và tải người dùng.

Mục tiêu kiểm thử các chức năng sau:

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

- Kiểm thử tự động chức năng đăng nhập.
- Kiểm thử tự động chức năng đăng ký.
- Kiểm thử tự động chức năng tìm kiếm.
- Kiểm thử tự động chức năng đăng kí khóa học.

1.3 Giới hạn và phạm vi của đề án

1.3.1 Đối tượng nghiên cứu

- Đối tượng nghiên cứu: website Simplilearn.
- Khách thể nghiên cứu: khách hàng.
- Công nghệ kỹ thuật và Framework kiểm thử tự động hướng lai từ khóa và dữ liệu.

1.3.2 Phạm vi nghiên cứu

- Phạm vi không gian: ứng dụng kiểm thử tự động cho website Simplilearn
- Phạm vi thời gian: 03/2023 - 06/2023
- Ý nghĩa khoa học: Đề tài này cung cấp cơ hội để nghiên cứu và áp dụng các phương pháp, kỹ thuật, và công cụ kiểm thử phần mềm, đóng góp vào nghiên cứu và đánh giá về các phương pháp, quy trình, và tiêu chuẩn đảm bảo chất lượng phần mềm trong lĩnh vực giáo dục trực tuyến.
- Thực tiễn của đề tài: Thực hiện xây dựng được framework và kiểm thử tự động cho website Simplilearn giúp em áp dụng các kiến thức và kỹ năng đã học được trong lĩnh vực công nghệ thông tin, như kiểm thử phần mềm, phân tích yêu cầu, lập trình. Ngoài ra, thực hiện kiểm thử chức năng giúp đảm bảo trải nghiệm người dùng tốt hơn trên website Simplilearn.

1.4 Nội dung thực hiện

- Xác định các yêu cầu kiểm thử cho website bao gồm việc xác định các chức năng, kịch bản kiểm thử, test script, dữ liệu kiểm thử và các trường hợp kiểm thử.

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

- Xây dựng các thành phần của framework theo hướng Hybrid driven testing framework (là phương pháp kết hợp cả hai phương pháp là data-driven testing và keyword-driven testing) bao gồm việc xây dựng các thư viện, hàm hỗ trợ, và công cụ cần thiết để thực hiện kiểm thử. Thiết kế code xử lý chung có thể tái sử dụng để tăng tính hiệu quả và đơn giản hóa việc thực hiện kiểm thử.

- Áp dụng framework tự động thực hiện các bước kiểm thử, sử dụng dữ liệu và từ khóa đã được định nghĩa để kiểm thử các chức năng như: đăng ký khóa học, đăng ký tài khoản, đăng nhập tài khoản, tìm kiếm.

1.5 Phương pháp tiếp cận

➤ Cách tiếp cận:

Nghiên cứu tài liệu

Trải nghiệm

So sánh: so sánh với website khác

Nghiên cứu các chuẩn đảm bảo chất lượng phần mềm MCCALI 1977, ISO 9216, IEEE,

➤ Sử dụng các phương pháp kiểm thử:

Phương pháp phân tích giá trị biên.

Phương pháp phân vùng tương đương.

Phương pháp bảng quyết định.

➤ Sử dụng các công cụ hỗ trợ kiểm thử: Eclipse IDE

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Tổng quan về kiểm thử phần mềm

Kiểm thử phần mềm là hoạt động nhằm tìm kiếm và phát hiện ra các lỗi của phần mềm, đảm bảo phần mềm chính xác, đúng và đầy đủ theo yêu cầu của khách hàng, yêu cầu của sản phẩm đã đặt ra. Kiểm thử phần mềm cũng cung cấp mục tiêu, cái nhìn độc lập về phần mềm điều này cho phép đánh giá và hiểu rõ các rủi ro khi thực thi phần mềm.

2.1.1. Các phương pháp kiểm thử phần mềm:

- Kiểm thử hộp trắng (white box testing): Trong kiểm thử hộp trắng cấu trúc mã, thuật toán được đưa vào xem xét. Người kiểm thử truy cập vào mã nguồn của chương trình để có thể kiểm tra nó.
- Kiểm thử hộp đen (black box testing): Kiểm tra các chức năng của hệ thống dựa trên bản đặc tả yêu cầu.
- Kiểm thử hộp xám (gray box testing): Là sự kết hợp giữa black box testing và white box testing.

2.1.2. Vai trò của kiểm thử phần mềm

- Kiểm thử phần mềm là hoạt động đảm bảo chất lượng phần mềm và mang tính sống còn trong các dự án sản xuất phần mềm. Vì vậy nó đã trở thành quy trình bắt buộc trong các dự án phần mềm hiện nay.
- Kiểm thử phần mềm để tránh những rủi ro, lỗi phát sinh trong suốt quá trình tạo ra sản phẩm.
- Lỗi càng phát hiện ra sớm càng giúp tránh được rủi ro và chi phí.

2.1.3. Quy trình kiểm thử phần mềm



Hình 2. 1 Quy trình kiểm thử phần mềm

1. Phân tích yêu cầu

Đầu vào của giai đoạn phân tích yêu cầu bao gồm các tài liệu như: tài liệu đặc tả yêu cầu, tài liệu thiết kế hệ thống, tài liệu khách hàng yêu cầu về các tiêu chí chấp nhận của sản phẩm, bản prototype của khách hàng yêu cầu (nếu có), ...

Phân tích yêu cầu là giai đoạn đầu tiên trong quy trình kiểm thử phần mềm.

Đầu ra của giai đoạn phân tích yêu cầu bao gồm tài liệu chứa các câu hỏi và câu trả lời liên quan đến nghiệp vụ của hệ thống, tài liệu báo cáo tính khả thi, phân tích rủi ro của việc kiểm thử phần mềm.

2. Lập kế hoạch kiểm thử

Đầu vào của giai đoạn lập kế hoạch kiểm thử là các tài liệu đặc tả đã được cập nhật thông qua các câu hỏi và trả lời được đưa ra trong giai đoạn phân tích yêu cầu, tài liệu báo cáo tính khả thi, phân tích rủi ro của việc kiểm thử phần mềm.

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Dựa vào các tài liệu được cung cấp và cập nhật mới nhất, thông thường, test manager hoặc test leader sẽ là người lập kế hoạch kiểm thử cho cả QA team. Lập kế hoạch kiểm thử nhằm xác định một số yếu tố quan trọng sau:

- Xác định phạm vi(Scope) dự án: Dự án thực hiện trong thời gian bao lâu? Bao gồm những công việc gì cho từng khoảng thời gian xác định? Từ đó đưa ra lịch trình thực hiện cho từng công việc nhỏ sao cho phù hợp với toàn bộ đội dự án.
- Xác định phương pháp tiếp cận: Nói về cách tiếp cận để kiểm thử cho một đối tượng nào đó, thì phải dựa vào nhiều thứ, ví dụ như: Thời gian cho phép test có phù hợp với con số ước lượng, nhiều hay ít, yêu cầu chất lượng từ phía khách hàng thế nào? Cao, thấp hay khắc khe hay sao cũng được? Công nghệ / kỹ thuật sử dụng để phát triển ứng dụng này là gì? Lĩnh vực của hệ thống/sản phẩm đang được test (domain) là gì? ...Từ đó, test manager có thể đưa ra những phương pháp và kế hoạch phù hợp nhất cho cả quá trình thực hiện dự án sao cho đúng với các tiêu chí chấp nhận của sản phẩm và kịp tiến độ với các mốc thời gian bàn giao, phát hành.
- Xác định các nguồn lực
 - Con người: Bao nhiêu người tham gia dự án, ai sẽ test phần nào, bao nhiêu tester tham gia? Tester và nhóm phát triển có kinh nghiệm về lĩnh vực này không?
 - Thiết bị: số lượng server, version, máy tính, mobile để thực hiện test là bao nhiêu.
- Lên kế hoạch thiết kế công việc test:
 - Bản kế hoạch kiểm thử sẽ bao gồm các nội dung:
 - Liệt kê các chức năng cần kiểm thử.

- Để thực hiện test chức năng này thì cần làm những công việc gì, trong thời gian bao lâu, cái nào thực hiện trước, cái nào thực hiện sau, ai là người thực hiện.
- Xác định điều kiện bắt đầu: xác định những điều kiện tối thiểu để bắt đầu hoạt động kiểm thử cho từng chức năng.
- Xác định điều kiện kết thúc: khi có những điều kiện nào thì sẽ kết thúc việc kiểm thử.

Đầu ra của giai đoạn lập kế hoạch bao gồm các tài liệu như test plan, test estimation, test schedule.

3. Thiết kế kịch bản kiểm thử

Đầu vào của giai đoạn thiết kế kịch bản kiểm thử là test plan, test estimation, test schedule, các tài liệu đặc tả đã được cập nhật.

Review tài liệu: Đầu tiên, các kiểm thử viên cần review lại tất cả các tài liệu để xác định công việc cần làm, các công việc có khác gì so với dự án trước khách hàng đưa cho, chức năng nào cần test, chức năng nào không cần test lại nữa. Từ đó, vừa có thể tiết kiệm thời gian mà vẫn đưa ra được một kịch bản kiểm thử đầy đủ và hiệu quả.

Viết test case/ check list: Sau đó, tester bắt tay vào việc viết test case chi tiết dựa vào kế hoạch đã đưa ra và vận dụng các kỹ thuật thiết kế kịch bản kiểm thử. Test case cần bao phủ được tất cả các trường hợp kiểm thử có thể xảy ra cũng như đáp ứng đầy đủ các tiêu chí của sản phẩm. Đồng thời tester cũng cần đánh giá mức độ ưu tiên cho từng test case.

Chuẩn bị dữ liệu kiểm thử: Cùng với việc tạo ra các test case chi tiết, đội kiểm thử cũng cần chuẩn bị trước các dữ liệu kiểm thử cho các trường hợp cần thiết như test data, test script.

Review test case/ check list: Sau khi hoàn thành, các thành viên trong đội kiểm thử hoặc test leader cũng cần review lại test case đã tạo để có thể bổ sung, hỗ trợ lẫn nhau nhằm tránh những sai sót trong thiết kế test case và rủi ro về sau.

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Sau khi hoàn thành thiết kế kịch bản kiểm thử, đội kiểm thử sẽ có các tài liệu bao gồm: test design, test case, check list, test data, test automation script.

4. Thiết lập môi trường kiểm thử

Đầu vào của giai đoạn cài đặt môi trường kiểm thử là test plan, smoke test case, test data.

Việc cài đặt môi trường kiểm thử là giai đoạn cũng rất quan trọng trong vòng đời phát triển phần mềm. Môi trường kiểm thử sẽ được quyết định dựa trên những yêu cầu của khách hàng, hay đặc thù của sản phẩm ví dụ như server/ client/ network, ...

Tester cần chuẩn bị một vài test case để kiểm tra xem môi trường cài đặt đã sẵn sàng cho việc kiểm thử hay chưa. Đây chính là việc thực thi các smoke test case.

Đầu ra của giai đoạn này là môi trường đã được cài đặt đúng theo yêu cầu, sẵn sàng cho việc kiểm thử và kết quả của smoke test case.

5. Thực hiện kiểm thử

Tài liệu đầu vào của giai đoạn này là test plan, test design, test case, check list, test data, test automation script.

Thực hiện các test case như thiết kế và mức độ ưu tiên đã đưa ra trên môi trường đã được cài đặt.

So sánh với kết quả mong đợi sau báo cáo các bug xảy ra lên tool quản lý lỗi và theo dõi trạng thái của lỗi đến khi được sửa thành công.

Thực hiện re-test để verify các bug đã được fix và regression test khi có sự thay đổi liên quan.

Đầu ra của giai đoạn này là test results (kết quả kiểm thử), defect reports (danh sách các lỗi tìm được).

6. Đóng chu trình kiểm thử

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Đầu vào của giai đoạn đóng chu trình kiểm thử là bao gồm tất cả những tài liệu liên quan đã được tổng hợp, ghi chép và hoàn thiện đầy đủ trong suốt quy trình kiểm thử của dự án: tài liệu phân tích đặc tả yêu cầu, test plan, test results, defect reports, tài liệu Q&A, ...

Đây là giai đoạn cuối cùng trong quy trình kiểm thử phần mềm.

Ở giai đoạn này, QA team thực hiện tổng kết, báo cáo kết quả về việc thực thi test case, bao nhiêu case pass/ fail, bao nhiêu case đã được fix, mức độ nghiêm trọng của lỗi, bao nhiêu lỗi cao/ thấp, lỗi còn nhiều ở chức năng nào, dev nào nhiều lỗi. Chức năng nào đã hoàn thành test/ chưa hoàn thành test/ trễ tiến độ bàn giao.

Đánh giá các tiêu chí hoàn thành như phạm vi kiểm tra, chất lượng, chi phí, thời gian, mục tiêu kinh doanh quan trọng.

Ngoài ra, giai đoạn này cũng thảo luận tất cả những điểm tốt, điểm chưa tốt và rút ra bài học kinh nghiệm cho những dự án sau, giúp cải thiện quy trình kiểm thử.

2.2 Kiểm thử tự động phần mềm



Hình 2. 2 Giới thiệu kiểm thử tự động phần mềm

Automation testing (Kiểm thử tự động) là quá trình sử dụng các công cụ, script và phần mềm để thực hiện những trường hợp kiểm thử, bằng cách lặp lại những hành động được xác định trước. Automation testing tập trung vào việc thay thế hoạt động thủ công của con người bằng các hệ thống hoặc thiết bị.

- Ưu điểm:

- Độ tin cậy cao: công cụ kiểm thử tự động có sự ổn định cao hơn so với con người, đặc biệt trong trường hợp nhiều test cases, nên độ tin cậy cao hơn so với kiểm thử thủ công.

- Khả năng lặp: công cụ kiểm thử tự động ra đời là để giúp cho các tester không phải lặp đi lặp lại các thao tác (ví dụ: nhập dữ liệu, click, check kết quả...) 1 cách nhàm chán với độ tin cậy và ổn định cao.

- Khả năng tái sử dụng: với 1 bộ kiểm thử tự động, người ta có thể sử dụng cho nhiều phiên bản ứng dụng khác nhau, đây được gọi là tính tái sử dụng.

- Tốc độ cao: do thực thi bởi máy nên tốc độ của kiểm thử tự động nhanh hơn nhiều so với tốc độ của con người. Nếu cần 5 phút để thực thi một test case một cách thủ công thì có thể người ta chỉ cần khoảng 30s để thực thi một cách tự động.

- Chi phí thấp: nếu áp dụng kiểm thử tự động đúng cách, người ta có thể tiết kiệm được nhiều chi phí, thời gian và nhân lực, do kiểm thử tự động nhanh hơn nhiều so với kiểm thử thủ công, đồng thời nhân lực cần để thực thi và bảo trì scripts không nhiều.

- Nhược điểm

- Khó mở rộng, khó bảo trì: trong cùng một dự án, để mở rộng phạm vi cho kiểm thử tự động khó hơn nhiều so với kiểm thử thủ công vì cập nhật hay chỉnh sửa yêu cầu nhiều công việc như debug, thay đổi dữ liệu đầu vào và cập nhật code mới.

- Khả năng bao phủ thấp: do khó mở rộng và đòi hỏi nhiều kỹ năng lập trình nên độ bao phủ của kiểm thử tự động thấp xét trên góc nhìn toàn dự án.

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

- Vấn đề công cụ và nhân lực: hiện nay cũng có nhiều công cụ hỗ trợ kiểm thử tự động khá tốt nhưng chúng vẫn còn nhiều hạn chế. Ngoài ra nhân lực đạt yêu cầu (có thể sử dụng thành thạo các công cụ này) cũng không nhiều.

2.2.1. Quy trình kiểm thử tự động

- Lựa chọn công cụ kiểm thử

Lựa chọn Công cụ kiểm thử phần lớn phụ thuộc vào công nghệ mà Ứng dụng Kiểm thử được xây dựng.

- Xác định phạm vi tự động

Phạm vi tự động là phạm vi ứng dụng được Kiểm thử tự động hóa. Các điểm sau giúp xác định phạm vi tự động:

- Các tính năng quan trọng
- Kịch bản có lượng dữ liệu lớn
- Các chức năng chung trên ứng dụng
- Tính khả thi về mặt kỹ thuật
- Mức độ các thành phần được sử dụng lại
- Độ phức tạp của test cases
- Khả năng sử dụng các test cases tương tự để kiểm thử trên nhiều trình duyệt
- Lập kế hoạch, thiết kế và phát triển
 - Tester tạo một chiến lược & kế hoạch kiểm thử tự động, chi tiết như sau:
 - Công cụ kiểm thử tự động được chọn
 - Thiết kế Framework và các tính năng của Framework
 - Các mục trong phạm vi và ngoài phạm vi tự động hóa

- Chuẩn bị kiểm thử tự động hóa
- Lịch trình và mốc thời gian thực thi kịch bản kiểm thử
- Bàn giao sản phẩm được kiểm thử tự động
- Thực thi kiểm thử

Kịch bản kiểm thử tự động hóa được thực thi trong giai đoạn này. Các Scripts cần dữ liệu kiểm thử đầu vào trước khi được thiết lập để chạy. Sau khi thực hiện sẽ trả về các báo cáo kiểm thử chi tiết.

Việc thực thi có thể được thực hiện bằng cách sử dụng công cụ tự động trực tiếp hoặc thông qua công cụ Quản lý kiểm thử sẽ gọi công cụ tự động hóa.

- Bảo trì

Khi các chức năng mới được thêm vào Hệ thống được kiểm thử với các chu kỳ liên tiếp, Scripts tự động hóa cũng cần được thêm, xem xét và bảo trì cho mỗi chu kỳ phát hành. Bảo trì trở nên cần thiết để cải thiện hiệu quả của Scripts trong kiểm thử tự động hóa.

2.2.2. Một số công cụ hỗ trợ kiểm thử tự động

a. Selenium

Khái niệm: Nó là một công cụ kiểm tra phần mềm được sử dụng để kiểm tra hồi quy (Regression Testing). Đây là một công cụ kiểm tra mã nguồn mở cung cấp chức năng phát lại và thu âm để kiểm tra hồi quy. Các Selenium IDE chỉ hỗ trợ trình duyệt web Mozilla Firefox.

❖ Đặc điểm của Selenium:

Cung cấp các điều khoản để xuất khẩu ghi lại kịch bản trong các ngôn ngữ khác như Java, Ruby, RSpec, Python, C #, JUnit và TestNG + Nó có thể thực hiện nhiều bộ kiểm thử cùng một lúc

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Xác định phần tử sử dụng id, tên, đường dẫn X, v.v. + Lưu trữ các bộ kiểm thử như Ruby Script, HTML và bất kỳ định dạng nào khác + Hỗ trợ tệp tin người dùng selenium-extensions.js

Cho phép để chèn ý kiến ở giữa của kịch bản để hiểu rõ hơn nội dung và mục đích của kịch bản kiểm thử

b. QTP (HP UFT)

Khái niệm: QTP được sử dụng rộng rãi để kiểm tra chức năng (Functional Testing) và hồi quy (Regression Testing), giải quyết các ứng dụng phần mềm và môi trường. Để đơn giản hóa việc tạo và bảo trì thử nghiệm, nó sử dụng khái niệm kiểm tra từ khóa.

❖ Đặc điểm:

Được sử dụng dễ dàng hơn dành cho người kiểm thử viên không theo ngành kỹ thuật để thích ứng và tạo ra các trường hợp thử nghiệm làm việc

Sửa lỗi nhanh hơn bằng cách ghi lại và sao chép các lỗi cho nhà phát triển

Thu gọn tài liệu thử nghiệm tại một trang web

QTP hỗ trợ môi trường phát triển .NET

Có cơ chế xác định đối tượng kiểm thử tốt

c. Rational Function Tester

Khái niệm: Là 1 công cụ kiểm tra tự động hướng đối tượng có khả năng tự động kiểm tra dữ liệu, kiểm tra giao diện, và kiểm thử hồi quy (Regression Testing)

❖ Đặc điểm:

+ Hỗ trợ một loạt các giao thức và ứng dụng như Java, HTML, NET, Windows, SAP, Visual Basic ...

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

- + Có thể ghi lại và phát lại các hành động theo yêu cầu + Tích hợp tốt với các công cụ quản lý kiểm soát nguồn như Rational Clear Case và tích hợp Rational Team Concert
- + Cho phép các nhà phát triển tạo ra các kịch bản liên quan đến từ khóa để có thể được tái sử dụng
- + Bộ biên tập Công cụ Java Developer Toolkit của Eclipse tạo điều kiện cho nhóm tạo mã thử nghiệm các đoạn mã trong Java với Eclipse
- + Hỗ trợ điều khiển tùy chỉnh thông qua proxy SDK (Java / .Net)
- + Hỗ trợ kiểm soát phiên bản để cho phép phát triển song song các kịch bản thử nghiệm

d. WATIR

Khái niệm: Là một phần mềm kiểm tra mã nguồn mở để kiểm thử hồi quy (Regression Testing). Watir chỉ hỗ trợ khám phá Internet trên các cửa sổ trong khi Watir webdriver hỗ trợ Chrome, Firefox, IE, Opera,

❖ Đặc điểm:

Hỗ trợ nhiều trình duyệt trên các nền tảng khác nhau

Sử dụng một ngôn ngữ kịch bản hiện đại có đầy đủ tính năng

Hỗ trợ ứng dụng web được viết bởi bất kỳ ngôn ngữ nào

Cho phép bạn viết các test case dễ đọc và bảo trì

e. SilkTest

Khái niệm: Silk Test được thiết kế để thực hiện kiểm tra chức năng (Functional Testing) và hồi quy (Regression Testing). Nó là một ngôn ngữ hướng đối tượng giống như C ++. Nó sử dụng các khái niệm về đối tượng, các class và sự kế thừa.

❖ Đặc điểm:

Nó bao gồm tất cả các tập tin mã nguồn

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Chuyển đổi các lệnh script thành các lệnh GUI. Trên cùng một máy, các lệnh có thể được chạy trên một máy từ xa hoặc máy chủ

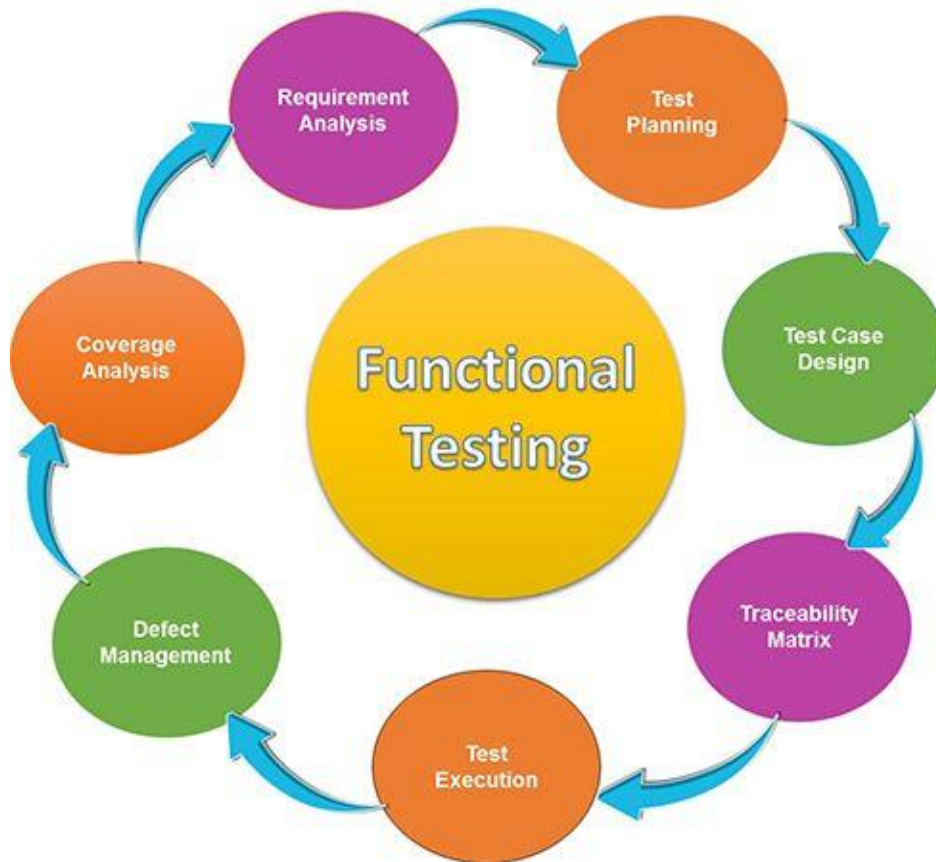
Để xác định chuyển động của con chuột cùng với các bấm phím, Silktest có thể được thực hiện. Nó có thể sử dụng cả phương pháp phát lại và ghi hoặc các phương pháp lập trình mô tả.

Xác định tất cả các điều khiển và cửa sổ của ứng dụng được thử dưới dạng các đối tượng và xác định tất cả thuộc tính và thuộc tính của mỗi đối tượng.

f. Apache JMeter

Khái niệm: Apache JMeter được dùng để kiểm thử khả năng chịu tải và kiểm thử hiệu năng cho các ứng dụng Web và một số ứng dụng khác. Công cụ này hỗ trợ kiểm thử hiệu năng của các mã nguồn được viết bằng các ngôn ngữ khác nhau như PHP, Java, ASP.NET, . . . Apache JMeter mô phỏng khả năng chịu tải của các máy chủ trên máy sử dụng để kiểm thử hệ thống. Công cụ này hỗ trợ giao diện đồ họa giúp phân tích tốt hiệu suất khi kiểm thử đồng thời nhiều ca kiểm thử. Ngoài ra, Apache JMeter còn hỗ trợ thêm nhiều tiện ích khác.

2.3 Kiểm thử chức năng



Hình 2. 3 Quy trình kiểm thử chức năng

Kiểm thử chức năng (Functional Testing) là một trong các quy trình đảm bảo chất lượng của lĩnh vực kiểm thử phần mềm. Đây là một loại kiểm thử hộp đen (black box testing), tức là các trường hợp nó cần xét đến sẽ dựa vào đặc tả của ứng dụng/phần mềm hoặc hệ thống đang thử nghiệm. Các chức năng sẽ được kiểm tra bằng cách nhập các giá trị đầu vào và sau đó sẽ kiểm tra, đánh giá các kết quả đầu ra mà không cần quan tâm đến các cấu trúc hay cài đặt bên trong của ứng dụng.

Kiểm thử chức năng là một quy trình so sánh sự khác biệt giữa đặc tả bên ngoài của phần mềm với các chức năng thực tế mà phần mềm cung cấp. Các đặc tả này phần nhiều sẽ dựa vào góc nhìn của người sử dụng về phần mềm, không liên quan đến các công nghệ sử dụng hay các thiết lập bên trong nó. Người sử dụng có thể là bất cứ ai, nếu họ tiếp xúc với phần mềm khi nó là thành phẩm có thể sử dụng được.

Kiểm thử chức năng đem lại khá nhiều lợi ích, chẳng hạn như tránh được việc kiểm thử dư thừa với các chức năng không cần thiết, hay ngăn chặn sự đa dạng lỗi tại cùng một thời điểm.

2.4 Giới thiệu framework kiểm thử tự động

2.4.1 Tổng quan về framework kiểm thử tự động

Framework là các đoạn code đã được viết sẵn, cấu thành nên một bộ khung và các thư viện lập trình được đóng gói. Chúng cung cấp các tính năng có sẵn như mô hình, API và các yếu tố khác để tối giản cho việc phát triển các ứng dụng web phong phú, năng động. Các framework giống như là chúng ta có khung nhà được làm sẵn nền móng cơ bản, bạn chỉ cần vào xây dựng và nội thất theo ý mình.

❖ Yêu cầu ở mức cao thì một Framework kiểm thử tự động phải thoả mãn ít nhất 3 tính chất:

- Tự động hoá việc thực thi kiểm thử:

Framework phải hỗ trợ hoàn toàn việc tự động hoá kiểm thử, nếu chỉ mỗi hỗ trợ việc thực thi ca kiểm thử thì chưa đủ. Nó còn phải hỗ trợ cả việc quản lý lỗi phát sinh, phân tích kết quả kiểm thử, quản lý báo cáo kiểm thử và ghi nhật ký việc kiểm thử một cách chi tiết.

- Dễ sử dụng:

Framework phải được sử dụng một cách dễ dàng và thuận tiện bởi kỹ sư kiểm thử hoặc những người có trách nhiệm kiểm thử chỉ với các kiến thức cơ bản về sử dụng máy tính và phần mềm Framework cũng cần phải hỗ trợ khả năng thiết kết, sửa đổi và quản lý kịch bản kiểm thử, ca kiểm thử, thực thi chúng và xem kết quả kiểm thử mà không cần thêm bất cứ kỹ năng lập trình nào.

- Bảo trì được:

Nó phải đủ dễ để bảo trì, sửa đổi, nâng cấp cả dữ liệu test lẫn mã nguồn của framework khi yêu cầu kiểm tra của hệ thống thay đổi hoặc được cập nhật hoặc do một lý do nào đó mà cần phải sửa đổi. Hơn thế nữa nó cũng phải dễ dàng cho phép thêm các tính năng mới vào framework.

❖ Các đặc tính của một framework có thể coi là các yêu cầu về mặt chắc năng mà framework đó phải có bao gồm các chức năng sau:

- Thực thi các ca kiểm thử:

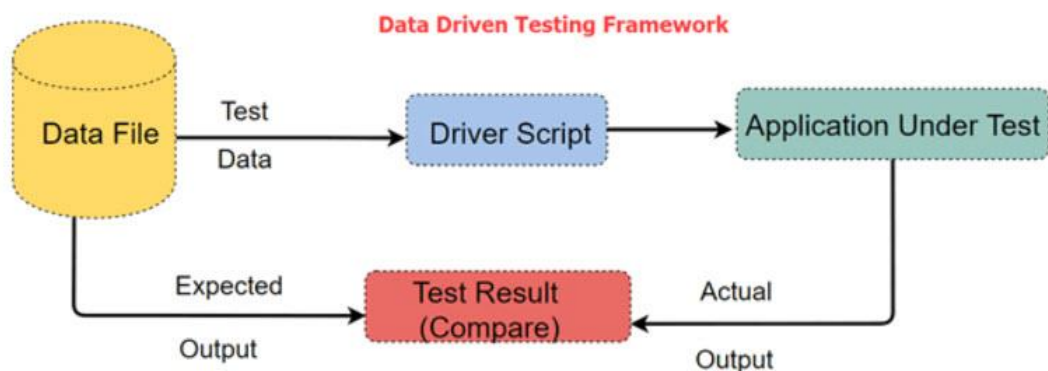
Framework phải có khả năng thực thi ca kiểm thử bằng một lệnh có thể thao tác trên giao diện đồ hoạ hoặc dòng lệnh. Điều này cũng đòi hỏi framework phải có khả năng thiết lập môi trường kiểm tra và hơn thế nữa trước khi thực thi một ca kiểm thử nó còn phải kiểm tra tiên điều kiện của ca kiểm thử đó đã được sẵn sàng hay chưa?

- Chức chạy và dừng ca kiểm thử: Framework phải hỗ trợ chức năng chạy từng ca kiểm thử hoặc dừng ca kiểm thử được chỉ định
- Chức năng quản lý lỗi
- Chắc năng xác minh kết quả kiểm thử
- Quản lý trạng thái ca kiểm thử
- Xử lý kết quả không mong đợi
- Ghi nhật ký chi tiết
- Tự động sinh báo cáo

2.4.2 Hybrid driven testing framework

Hybrid driven testing framework là một kiểu framework kiểm thử phần mềm kết hợp cả hai phương pháp là data-driven testing và keyword-driven testing. Nó nhằm tận dụng lợi ích của cả hai phương pháp để xây dựng một cách tiếp cận kiểm thử linh hoạt và hiệu quả.

➤ Data-Driven Testing Framework



Hình 2. 4 Quy trình hoạt động Data Driven Testing Framework

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Khung tự động hóa kiểm tra theo hướng dữ liệu tập trung vào việc tách logic các tập lệnh kiểm tra và dữ liệu kiểm tra khỏi nhau.

Nó cho phép chúng ta tạo các kịch bản tự động hóa thử nghiệm bằng cách chuyển các bộ dữ liệu thử nghiệm khác nhau.

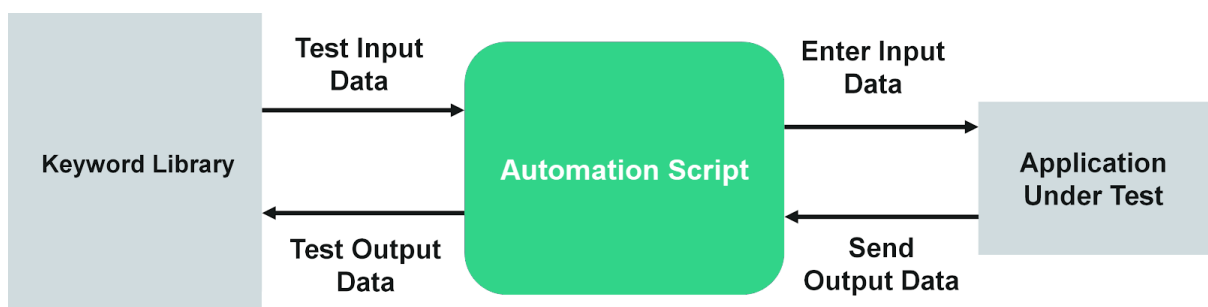
Tập dữ liệu thử nghiệm được lưu trong các tệp hoặc tài nguyên bên ngoài như Trang tính MS Excel, Bảng MS Access, Cơ sở dữ liệu SQL, tệp XML, v.v.,

Các kịch bản kiểm tra kết nối với các tài nguyên bên ngoài để lấy dữ liệu kiểm tra. Bằng cách sử dụng khuôn khổ này, chúng ta có thể dễ dàng làm cho các tập lệnh thử nghiệm hoạt động bình thường cho các bộ dữ liệu thử nghiệm khác nhau. Khuôn khổ này làm giảm đáng kể số lượng tập lệnh thử nghiệm so với khuôn khổ dựa trên mô-đun.

Khung này cung cấp phạm vi kiểm tra nhiều hơn với các bài kiểm tra có thể sử dụng lại và tính linh hoạt trong việc thực hiện các bài kiểm tra chỉ khi được yêu cầu và chỉ bằng cách thay đổi dữ liệu kiểm tra đầu vào.

Nó đáng tin cậy về mặt không ảnh hưởng đến các bài kiểm tra bằng cách thay đổi dữ liệu kiểm tra nhưng nó có những hạn chế riêng, chẳng hạn như người kiểm tra làm việc trên khuôn khổ này cần phải có kiến thức lập trình thực hành để phát triển các tập lệnh kiểm tra.

➤ Keyword-Driven Testing Framework



Hình 2. 5 Quy trình hoạt động Keyword-Driven Testing Framework

Nó còn được gọi là thử nghiệm theo hướng bảng hoặc thử nghiệm dựa trên từ hành động. Trong thử nghiệm theo hướng từ khóa, chúng tôi sử dụng định dạng bảng để xác định từ khóa hoặc từ hành động cho từng chức năng hoặc phương pháp mà chúng tôi sẽ thực thi. Nó thực hiện các kịch bản kiểm tra tự động hóa dựa trên các từ khóa được chỉ định trong bảng excel.

2.5 Công nghệ áp dụng

2.5.1 Selenium WebDriver



Hình 2. 6 Giới thiệu tổng quan về Selenium WebDriver

Selenium WebDriver (gọi tắt là WebDriver) là 1 automation framework của web, cho phép thực thi các test của bạn đối với các trình duyệt khác nhau, không chỉ Firefox, Chrome. WebDriver cũng cho phép bạn sử dụng các ngôn ngữ lập trình trong việc tạo test script của bạn. Bạn có thể sử dụng conditional operations như if- else hay là switch-

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

case hoặc vòng lặp do-while. WebDriver hỗ trợ nhiều ngôn ngữ lập trình khác nhau như Java, .Net, PHP, Python, Perl, Ruby. Không nhất thiết bạn phải am hiểu tất cả các ngôn ngữ trên nhưng để sử dụng WebDriver 1 cách hiệu quả hơn, bạn nên biết ít nhất một trong số những ngôn ngữ trên.

❖ Tại sao sử dụng Selenium Web Driver?

- Người dùng có thể dùng miễn phí
- Kiến trúc đơn giản:
 - + Điều khiển trình duyệt từ hệ điều hành.
 - + Bạn chỉ cần IDE và trình duyệt là xong.
- Ngôn ngữ lập trình hỗ trợ: Web Driver hỗ trợ bạn viết kịch bản kiểm thử bằng các ngôn ngữ khác nhau như Java, .NET, PHP, Python, Perl, Ruby và bạn có thể sử dụng các điều kiện if else, các vòng lặp để tăng tính chính xác cho kịch bản kiểm thử.
- Tốc độ: Khi so sánh với các công cụ khác của bộ Selenium, WebDriver là công cụ nhanh nhất trong số tất cả do tương tác trực tiếp từ hệ điều hành tới trình duyệt

❖ Ưu điểm của Selenium WebDriver

- Hỗ trợ 7 ngôn ngữ lập trình: JAVA, C #, PHP, Ruby, Perl, Python và .Net.
- Hỗ trợ thử nghiệm trên nhiều trình duyệt khác nhau như: Firefox, Chrome, IE, Safari
- Các thử nghiệm có thể được thực hiện trên các hệ điều hành khác nhau như: Windows, Mac, Linux, Android, iOS
- Khắc phục các hạn chế của Selenium v1 như tải lên tệp, tải xuống, cửa sổ bật lên và hộp thoại

❖ Nhược điểm của Selenium WebDriver

- Báo cáo kiểm thử chi tiết không thể được tạo ra
- Không thể kiểm tra hình ảnh

2.5.2 TestNG

TestNG là một testing Framework được xây dựng trên cảm hứng từ JUnit và Nunit nhưng đem đến một số tính năng mới làm cho nó mạnh hơn và dễ dàng sử dụng hơn. Là công cụ kiểm thử tự động cho phép tạo testcase thực thi các danh mục kiểm thử: Unit test, test chức năng, TestNG là một automation testing testing Framework, là phiên bản nâng cấp với nhiều tính năng mạnh mẽ và vượt trội dễ dàng sử dụng và là công cụ kiểm thử tự động được sử dụng phổ biến hiện nay. Nó loại bỏ hầu hết các hạn chế khuôn mẫu cũ và cho phép nhà phát triển khả năng viết các bài kiểm tra linh hoạt và mạnh mẽ với sự trợ giúp của các chú thích đơn giản.



Hình 2. 7 Giới thiệu tổng quan về TestNG

- Một số đặc điểm và tính năng chính của TestNG:
 - Hỗ trợ nhiều loại kiểm thử: TestNG hỗ trợ kiểm thử đơn vị (unit testing), kiểm thử tích hợp (integration testing), kiểm thử hệ thống (system testing) và kiểm thử chức năng (functional testing).
 - Định nghĩa và quản lý các bộ kiểm thử: TestNG cho phép bạn định nghĩa các bộ kiểm thử (test suite) để nhóm các test case lại với nhau. Bạn có thể tạo ra các bộ kiểm thử phức tạp bằng cách chỉ định các phụ thuộc và sắp xếp thứ tự thực thi các test case.

- Hỗ trợ đa luồng và song song: TestNG có khả năng thực thi các test case đồng thời và song song, giúp tăng tốc độ thực thi kiểm thử.
- Hỗ trợ các Annotation: TestNG sử dụng các Annotation để đánh dấu và quản lý các test case và phương thức kiểm thử. Các Annotation như `@Test`, `@BeforeTest`, `@AfterTest`, `@DataProvider` cho phép bạn dễ dàng xác định các bước chuẩn bị, tiền xử lý và hậu xử lý cho các test case.
- Tích hợp dễ dàng: TestNG tích hợp tốt với các công cụ phát triển phần mềm khác như Eclipse, IntelliJ IDEA và Maven. Nó cung cấp các plugin và tính năng tích hợp để dễ dàng thực hiện kiểm thử trong quy trình phát triển.

2.5.3 Eclipse

Eclipse là 1 công cụ hỗ trợ lập trình mã nguồn mở được phát triển bởi IBM. Eclipse như một môi trường phát triển Java tích hợp (IDE), với Eclipse chúng ta có thể mở rộng hơn mã nguồn bằng cách chèn thêm các plugins cho project (PDE- Plug-in Development Environment). Mặc dù Eclipse được viết bằng ngôn ngữ lập trình Java, nhưng việc sử dụng nó không hạn chế chỉ cho ngôn ngữ Java. Ví dụ, Eclipse hỗ trợ sẵn hoặc có thể cài thêm các plugins để hỗ trợ cho các ngôn ngữ lập trình như C/C++ và COBOL. Ngoài ra, còn rất nhiều ngôn ngữ khác như PHP, Groovy, ... Eclipse còn hỗ trợ cho lập trình viên code theo các mô hình phát triển như MVC, tạo thêm các lib hỗ trợ phát triển phần mềm.



Hình 2. 8 Giới thiệu tổng quan về Eclipse

➤ Ưu điểm

- Tạo thuận lợi cho tích hợp liên mạch các công cụ bên trong mỗi một và xuyên qua nhiều kiểu nội dung và các nhà cung cấp công cụ khác nhau.

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

- Hỗ trợ việc xây dựng nhiều công cụ.
- Hỗ trợ một tập hợp không hạn chế các nhà cung cấp công cụ, bao gồm cả các nhà cung cấp phần mềm độc lập (ISV).
- Hỗ trợ các công cụ thao tác các kiểu nội dung bất kỳ (bao gồm cả HTML, Java, C, JSP, EJB, XML, và GIF).
- Hỗ trợ cả môi trường phát triển ứng dụng GUI lẫn không dựa trên GUI.
- Chạy trên nhiều hệ điều hành, bao gồm cả Windows và Linux.
- Lợi dụng tính phổ biến của ngôn ngữ Java để viết các công cụ.
- Do sử dụng SWT/JFace nên có lẽ load nhanh hơn.

➤ Nhược điểm

- Cài đặt phức tạp
- Tốn phần cứng máy Eclipse cũng khá tốn bộ nhớ máy để cài đặt nên hiện nay eclipse đã khắc phục bằng phiên bản Eclipse Luna, không cần cài đặt
- Tốn bộ nhớ máy: chạy eclipse cũng khá tốn bộ nhớ máy tính nên để sử dụng nó máy của bạn cũng cần có cấu hình tương đối cao
- Thiếu tính nhất quán vì có nhiều plugins quá.

CHƯƠNG 3: ĐẶC TẢ HỆ THỐNG PHẦN MỀM

3.1. Giới thiệu về hệ thống phần mềm

Mục đích của tài liệu này để xác định chức năng và các yêu cầu khác về chương trình quản lý website giáo dục trực tuyến, bao gồm giới thiệu tổng quát về hệ thống, yêu cầu, tính ứng dụng, độ tin cậy và hiệu suất. Tài liệu này được dùng cho đội dự án để phát triển sản phẩm.

Phần 1: Giới thiệu chung về tài liệu.

Phần 2: Cung cấp thông tin tổng quát về Website Simplilearn.

Phần 3: Mô tả các yêu cầu cụ thể của các chức năng, bao gồm input, output, các xử lý của chương trình.

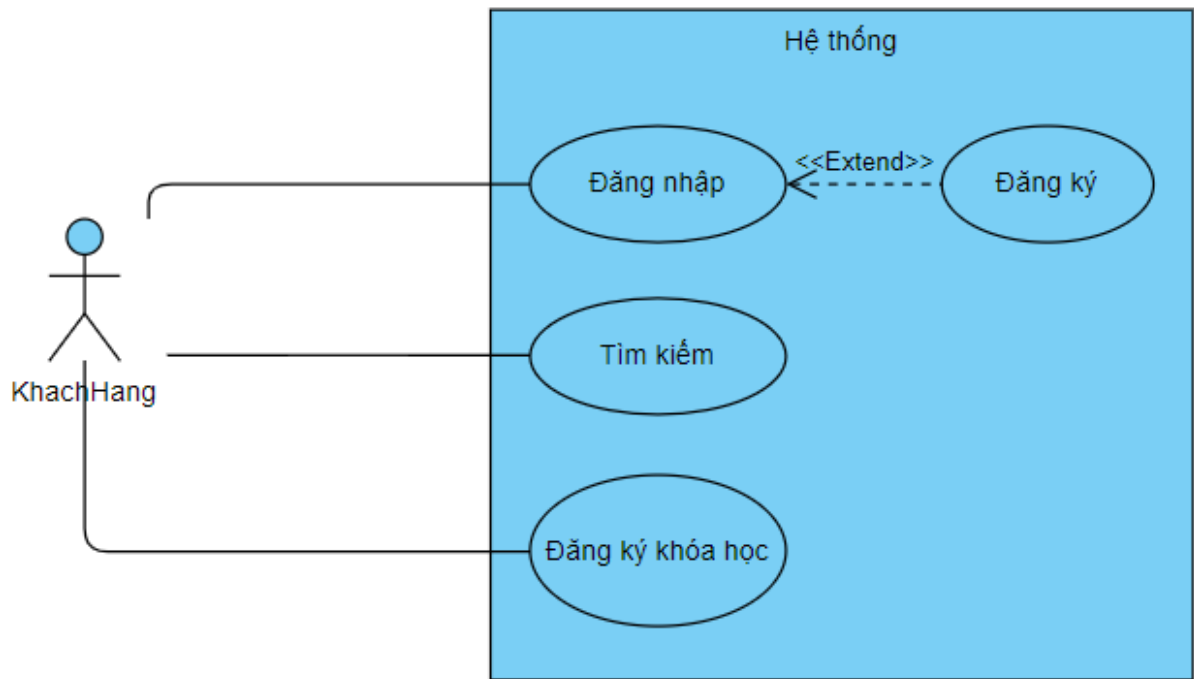
Phần 4: Mô tả các yêu cầu khác liên quan đến tính ứng dụng của hệ thống (tính thân thiện với end user).

Phần 5: Mô tả các yêu cầu về tính toàn vẹn của dữ liệu, hiệu suất

3.2. Các yêu cầu chức năng

3.2.1. Chức năng tổng quát

- Usecase tổng quát phân hệ người dùng



Hình 3. 1 Usecase tổng quát phân hệ người dùng

➤ Các usecase được mô tả như sau:

Bảng 3. 1 Mô tả cho usecase tổng quát

STT	Chức năng	Mô tả
1	Đăng nhập	Cho phép khách hàng đăng nhập bằng tài khoản email, Facebook, Google, ...
2	Đăng ký	Khách hàng phải cung cấp họ tên, email, mật khẩu để đăng ký
3	Tìm kiếm	Khách hàng nhập thông tin tìm kiếm vào thanh tìm kiếm để tìm kiếm thông tin. Khách hàng cũng có thể sử dụng bộ lọc tìm kiếm
4	Đăng kí khóa học	Cho phép khách hàng xem thông tin chi tiết và đăng kí khóa học.

3.2.2. Chức năng đăng nhập

1. Mục đích

Khách hàng đăng nhập vào hệ thống

2. Tác nhân liên quan

Khách hàng

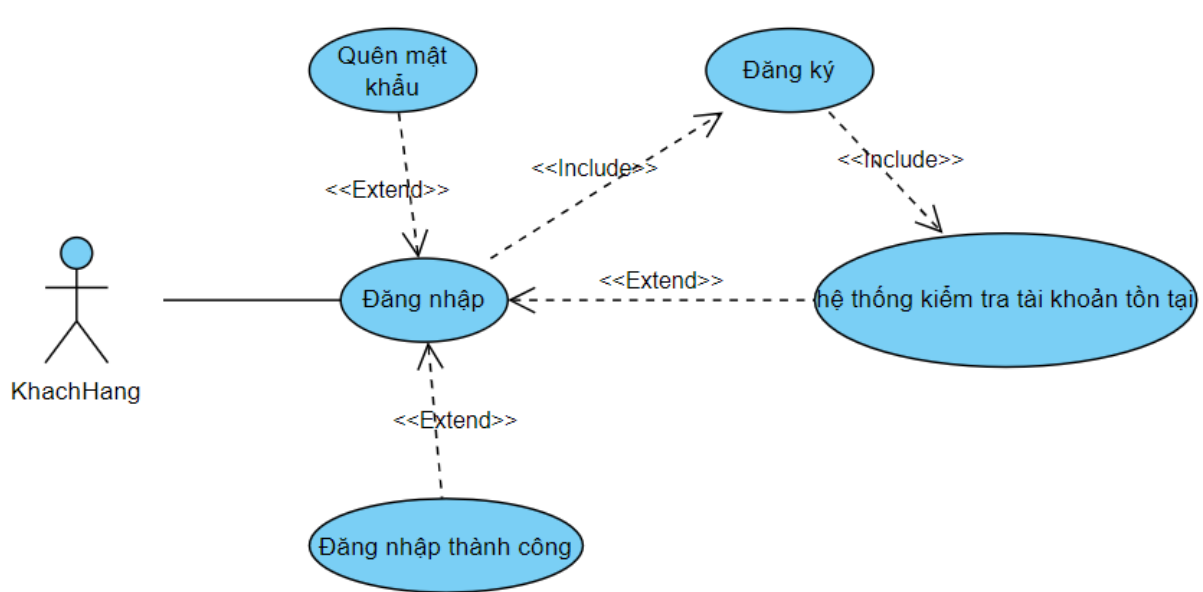
3. Điều kiện trước

Khách hàng phải đăng ký tài khoản và phải nhập tài khoản, mật khẩu chính xác

4. Điều kiện sau

Không

5. Biểu đồ use case



Hình 3. 2 Use case chức năng đăng nhập

6. Dòng sự kiện chính

Bảng 3. 2 Dòng sự kiện chính- chức năng đăng nhập

Hành động của tác nhân	Phản ứng của hệ thống
------------------------	-----------------------

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

1. Khách hàng chọn chức năng “Đăng nhập” vào hệ thống.	2. Form “Đăng nhập” hiển thị.
3. Khách hàng nhập tên tài khoản và mật khẩu sau đó nhấn button “Đăng nhập”.	4. Hệ thống kiểm tra dữ liệu các thông tin đã nhập. Nếu đúng sẽ lưu vào cơ sở dữ liệu thông tin về người dùng và hiển thị thông tin tài khoản.

7. Dòng sự kiện phụ

Không

8. Dòng sự kiện rẽ nhánh

NA

9. Yêu cầu validate data

Bảng 3. 3 Validate data chức năng đăng nhập

STT	Tên thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	Email	Nvarchar	10 - 50	Not null	
2	MatKhau	Nvarchar	8-128	Not null	

10. Nội dung thông báo

Bảng 3. 4 Nội dung thông báo chức năng đăng nhập

STT	Tình huống	Thông báo
1	Khi nhập email sai	Hiển thị thông báo “Please enter a valid email address”
2	Khi nhập mật khẩu sai	Hiển thị thông báo Password Criteria

3.2.3. Chức năng đăng ký

1. Mục đích

Khách hàng đăng ký tài khoản

2. Tác nhân liên quan

Khách hàng

3. Điều kiện trước

Khách hàng đăng ký tài khoản

4. Điều kiện sau

Không

6. Biểu đồ use case



Hình 3. 3 Usecase chức năng đăng ký

7. Dòng sự kiện chính

Bảng 3. 5 Dòng sự kiện chính chức năng đăng ký

Hành động của tác nhân	Phản ứng của hệ thống
1. Khách hàng chọn chức năng “Đăng ký” tài khoản	2. Form “Đăng ký” hiển thị.

3. Khách hàng nhập họ tên, địa chỉ email, mật khẩu,... sau đó nhấn button “Đăng ký”.	4. Hệ thống kiểm tra dữ liệu các thông tin đã nhập. Nếu thông tin hợp lệ sẽ lưu vào cơ sở dữ liệu thông tin về người dùng và quay lại trang đăng nhập.
--	--

8. Dòng sự kiện phụ

Không

9. Dòng sự kiện rẽ nhánh

NA

10. Yêu cầu validate data

Bảng 3. 6 Validate data chức năng đăng ký

STT	Tên thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	Tên	Nvarchar	10-20	Not null	
2	SDT	int	7-15	Not null	
3	Email	Nvarchar	10 - 50	Not null	
4	MatKhau	Nvarchar	8-125	Not null	
5	CountryCode	Nvarchar	1000		

11. Nội dung thông báo

Bảng 3. 7 Nội dung thông báo chức năng đăng ký

STT	Tình huống	Thông báo
1	Khi nhập email không tồn tại, sai định dạng	Hiện thị thông báo “Please enter a valid email address”

2	Khi nhập email đã đăng ký	Hiện thị thông báo” Email address already exists, please choose a different one”
3	Khi không nhập thông tin	Hiện thị thông báo “Please fill in all required fields”

3.2.4. Chức năng tìm kiếm

12. Mục đích

Khách hàng tìm kiếm thông tin theo yêu cầu

13. Tác nhân liên quan

Khách hàng

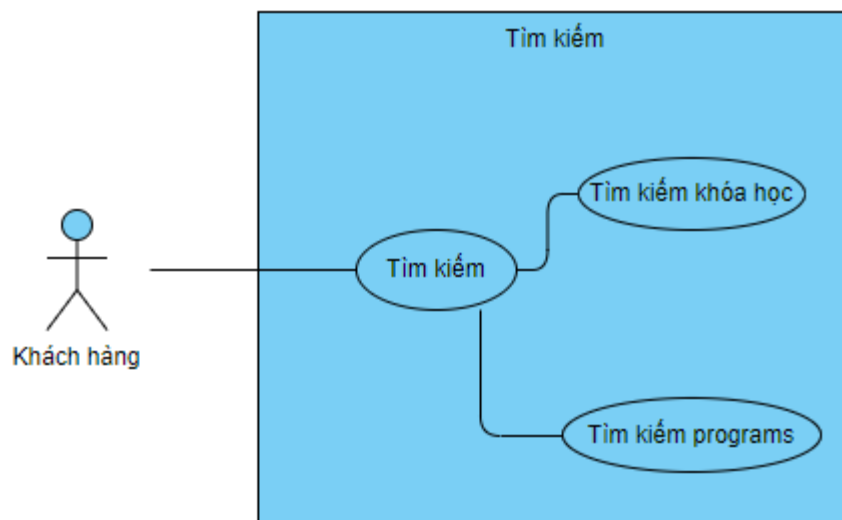
14. Điều kiện trước

Không

15. Điều kiện sau

Không

16. Biểu đồ use case



Hình 3. 4 Use case chức năng tìm kiếm

17. Dòng sự kiện chính

Bảng 3. 8 Dòng sự kiện chính chức năng tìm kiếm

Hành động của tác nhân	Phản ứng của hệ thống
1. Khách hàng nhập thông tin tìm kiếm sau đó ấn button “tìm kiếm” hoặc ấn phím enter	1. Hệ thống kiểm tra dữ liệu và trả về kết quả phù hợp với từ khóa đã được nhập

18. Dòng sự kiện phụ

Không

19. Dòng sự kiện rẽ nhánh

NA

20. Yêu cầu validate data

Bảng 3. 9 Validate data chức năng tìm kiếm

STT	Tên thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	Từ khóa tìm kiếm	Nvarchar	10-20	Not null	

21. Nội dung thông báo

Bảng 3. 10 Nội dung thông báo chức năng tìm kiếm

STT	Tình huống	Thông báo
1	Khi nhập từ khóa tìm kiếm không tồn tại	Hiện thị thông báo” Your search did not match any content.”

3.2.5. Chức năng đăng ký khóa học

1. Mục đích

Khách hàng đăng ký khóa học

2. Tác nhân liên quan

Khách hàng

3. Điều kiện trước

Không

4. Điều kiện sau

Không

6. Biểu đồ use case



Hình 3. 5 Usecase chức năng đăng ký khóa học

7. Dòng sự kiện chính

Bảng 3. 11 Dòng sự kiện chính chức năng đăng ký

Hành động của tác nhân	Phản ứng của hệ thống
1. Khách hàng chọn chức năng Đăng ký khóa học	1. Form “Đăng ký” hiển thị.

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

2. Khách hàng nhập họ tên, địa chỉ email, số điện thoại, số năm kinh nghiệm,... sau đó nhấn button đăng ký	2. Hệ thống kiểm tra dữ liệu các thông tin đã nhập. Nếu thông tin hợp lệ sẽ chuyển sang trang biểu mẫu để hoàn thành đăng ký
3. Khách hàng nhập thông tin chi tiết về giới tính, trình độ học vấn, tổng số năm kinh nghiệm... sau đó nhấn gửi	3. Hệ thống kiểm tra thông tin hợp lệ sẽ thông báo đăng ký thành công

8. Dòng sự kiện phụ

Không

9. Dòng sự kiện rẽ nhánh

NA

10. Yêu cầu validate data

Bảng 3. 12 Validate data chức năng đăng ký khóa học

STT	Tên thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1	Tên	Nvarchar	10-20	Not null	
2	SĐT	int	7-15	Not null	
3	Email	Nvarchar	10 - 50	Not null	
4	Total year experience	Nvarchar	8-125	Not null	

Bảng 3. 13 Validate data biểu mẫu đăng ký khóa học

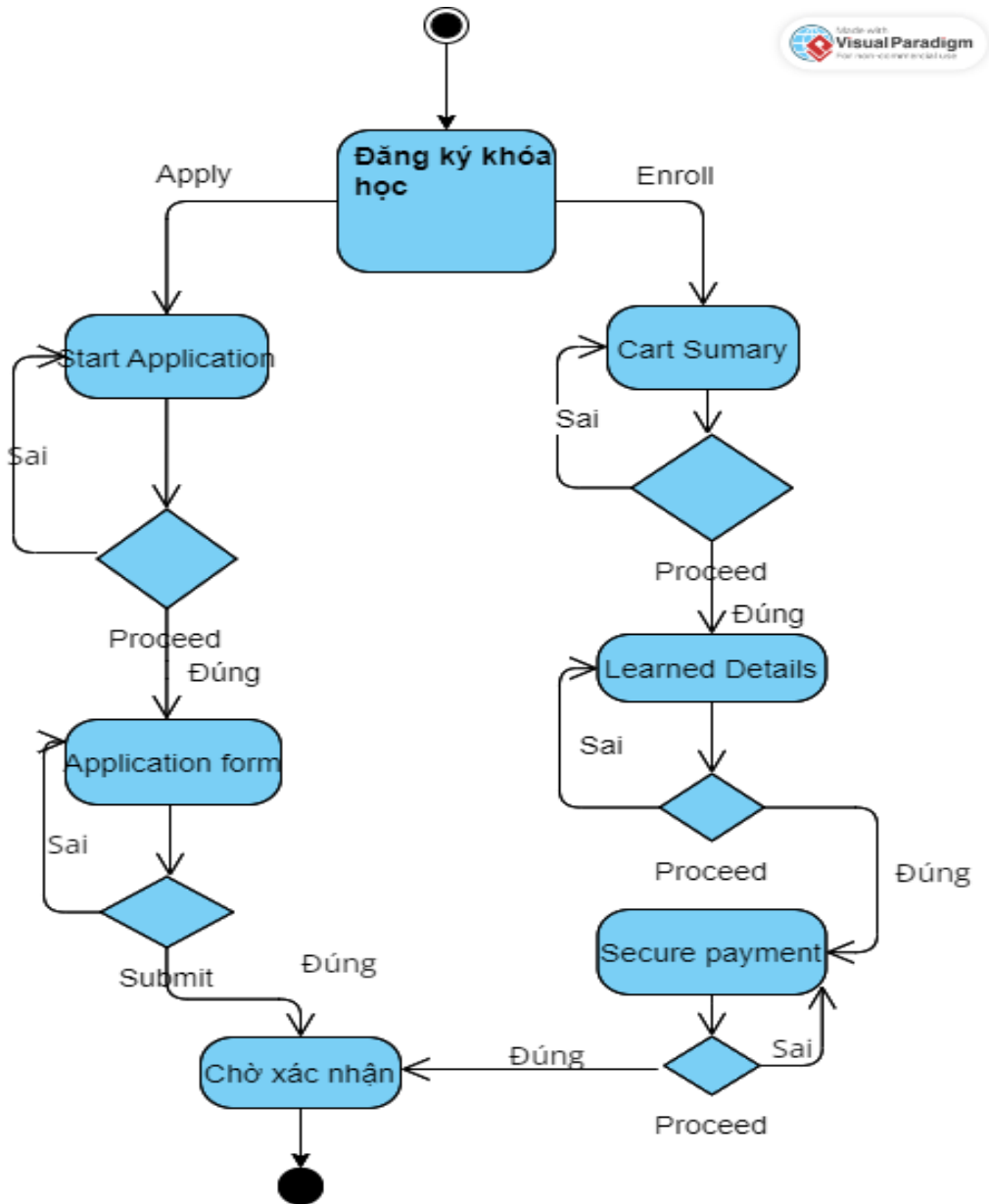
STT	Tên thuộc tính	Kiểu dữ liệu	Kích thước	Ràng buộc	Ghi chú
1.	Name	Nvarchar	10-20	Not null	
2.	Email	Nvarchar	10 - 50	Not null	
3.	Gender	Nvarchar	4-20	Not null	
4.	Date of Birth	Date		Not null	
5.	Location	Nvarchar	200	Not null	
6.	Highest Qualification	Nvarchar	200	Not null	
7.	Degree of Specialisation	Nvarchar	200	Not null	
8.	Name of the Institute	Nvarchar	200	Not null	
9.	Year of passing	int		Not null	
10.	Percentage	Nvarchar	200	Not null	
11.	Total years of work experience	int		Not null	
12.	Current Organization	Nvarchar	200		
13.	Designation	Nvarchar	200		
14.	Years of Relevant (Analytics/DevOps/Software) Experience	Nvarchar	200		
15.	state_of_purpose	Nvarchar	1-400	Not null	

11. Nội dung thông báo

Bảng 3. 14 Nội dung thông báo chức năng đăng ký khóa học

STT	Tình huống	Thông báo
1	Khi nhập thiếu trường dữ liệu	Hiện thị thông báo lỗi ngay dưới trường thông tin

12. Luồng nghiệp vụ



Hình 3. 6 Luồng nghiệp vụ chức năng đăng ký khóa học

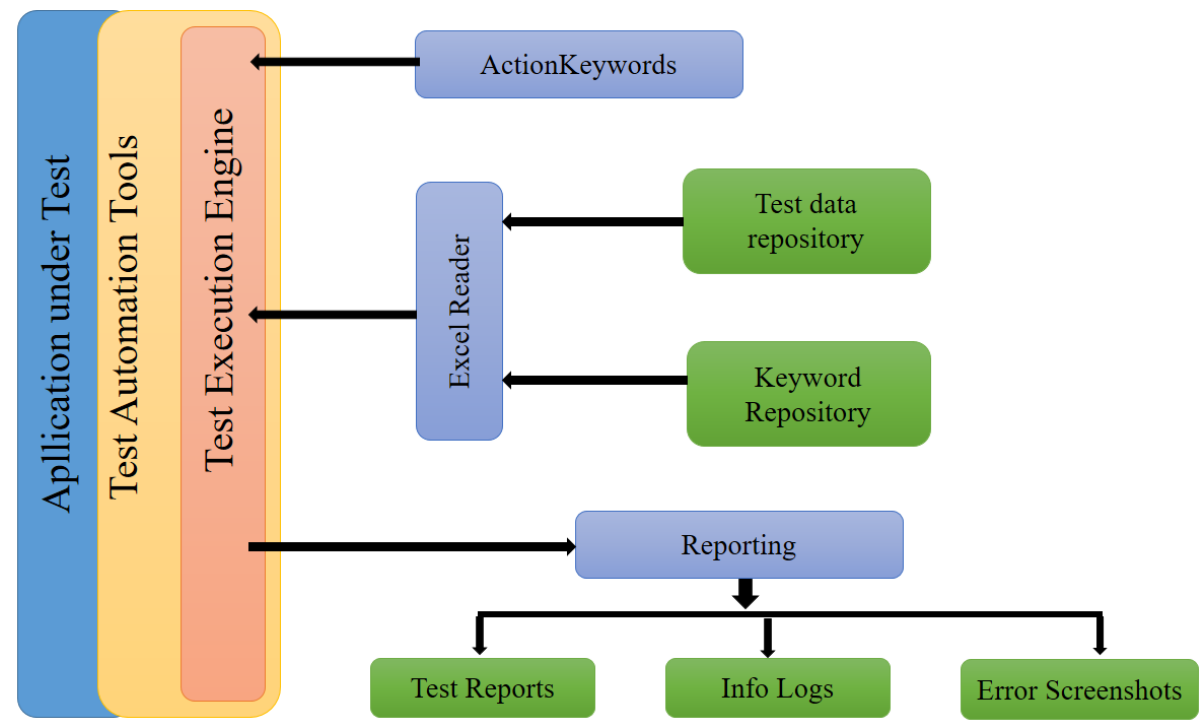
3.3. Các yêu cầu phi chức năng

Bảng 3. 15 Các yêu cầu phi chức năng

Mục	Tên yêu cầu	Mô tả yêu cầu
1.	Yêu cầu về hiệu năng	Thời gian yêu cầu mở 1 website con bất kì: <3s. Khả năng đáp ứng và mở rộng của hệ thống với lượng truy cập tăng vọt. Số kết nối tối đa mà hệ thống có thể đáp ứng khi có nhiều số người truy cập cùng thời điểm.
2.	Các ràng buộc thiết kế	Hệ thống được thiết kế theo web framework Next.js, ngôn ngữ viết phần mềm là Node.js. Các công cụ hỗ trợ phát triển: Sql Server, Visual Studio, Postman.
3.	Yêu cầu về độ tin cậy (Reliability)	Hệ thống có độ tin cậy cao. Thời gian khắc phục lỗi tối đa là một ngày.
4.	Yêu cầu về tính khả dụng (Availability)	Hệ thống có khả năng cung cấp điều kiện cho người dùng thực hiện các tác vụ một cách an toàn và hiệu quả trong khi sử dụng.
5.	Yêu cầu về bảo mật (Security)	Dữ liệu đăng nhập phải được mã hóa, các thao tác quản trị không được thực hiện khi chưa đăng nhập.
6.	Yêu cầu về bảo trì (Maintainability)	Hệ thống có thể dễ dàng bảo trì được khi yêu cầu người dùng thay đổi.
7.	Yêu cầu tính khả chuyển (Portability)	Hệ thống chạy được trên các môi trường khác nhau như mobile và desktop.
8.	Yêu cầu về tính thực tế	Phần mềm có thể phục vụ 1000 người dùng.

CHƯƠNG 4: TRIỂN KHAI KIỂM THỬ TỰ ĐỘNG

4.1. Phương pháp xây dựng Framework



Hình 4. 1 Cấu trúc Framework kiểm thử tự động

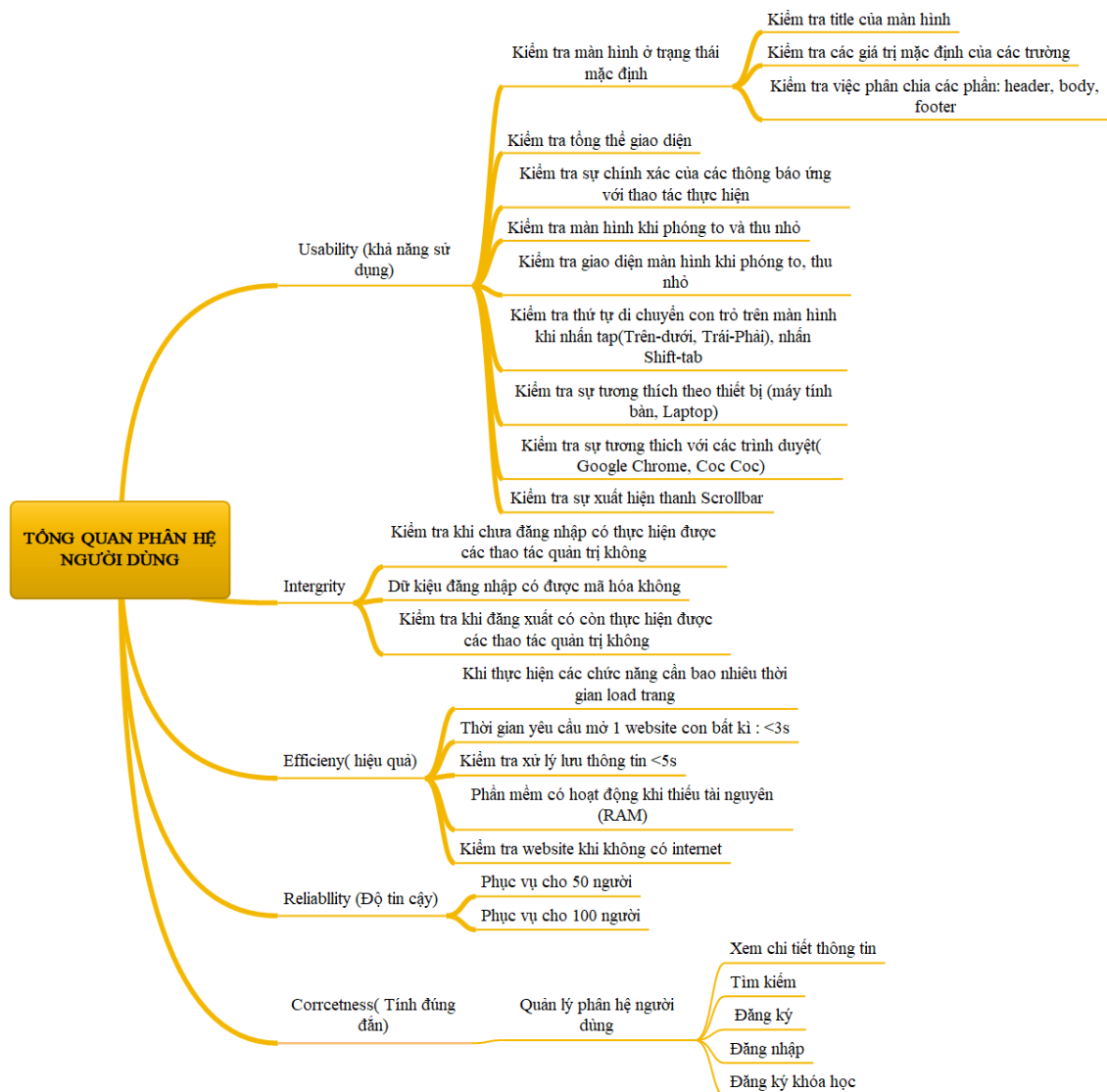
Framework kiểm thử tự động được xây dựng như sau:

1. Test Data Repository: lưu trữ các dữ liệu kiểm thử, bao gồm cả dữ liệu đầu vào và dữ liệu mong đợi, được lưu trữ trong file Excel.
2. Keyword Repository: lưu trữ các từ khóa (keywords) mô tả các hành động và chức năng của ứng dụng. Mỗi từ khóa tương ứng với một tác vụ cụ thể trong quá trình kiểm thử được lưu trữ trong file Excel.
3. ActionKeywords: lưu trữ các tập hợp các phương thức, các hàm được định nghĩa để thực hiện các hành động như điều hướng đến URL, nhập liệu, nhấp chuột, xác nhận, kiểm tra dữ liệu và các hành động khác liên quan đến ứng dụng.
4. Test Execution Engine: được xây dựng để thực thi các bước kiểm thử, sử dụng dữ liệu từ Test Data Repository và Keyword Repository để thực hiện kiểm thử theo luồng dữ liệu và các từ khóa đã được xác định trước.

5. Reporting and Logging: báo cáo và ghi lại kết quả sau khi thực hiện kiểm thử, tạo lập và phân tích kết quả kiểm thử, đồng thời ghi lại các lỗi phát sinh trong quá trình kiểm thử.

4.2. Thiết kế các yêu cầu kiểm thử

4.2.1. Thiết kế kiểm thử tổng quan hệ thống

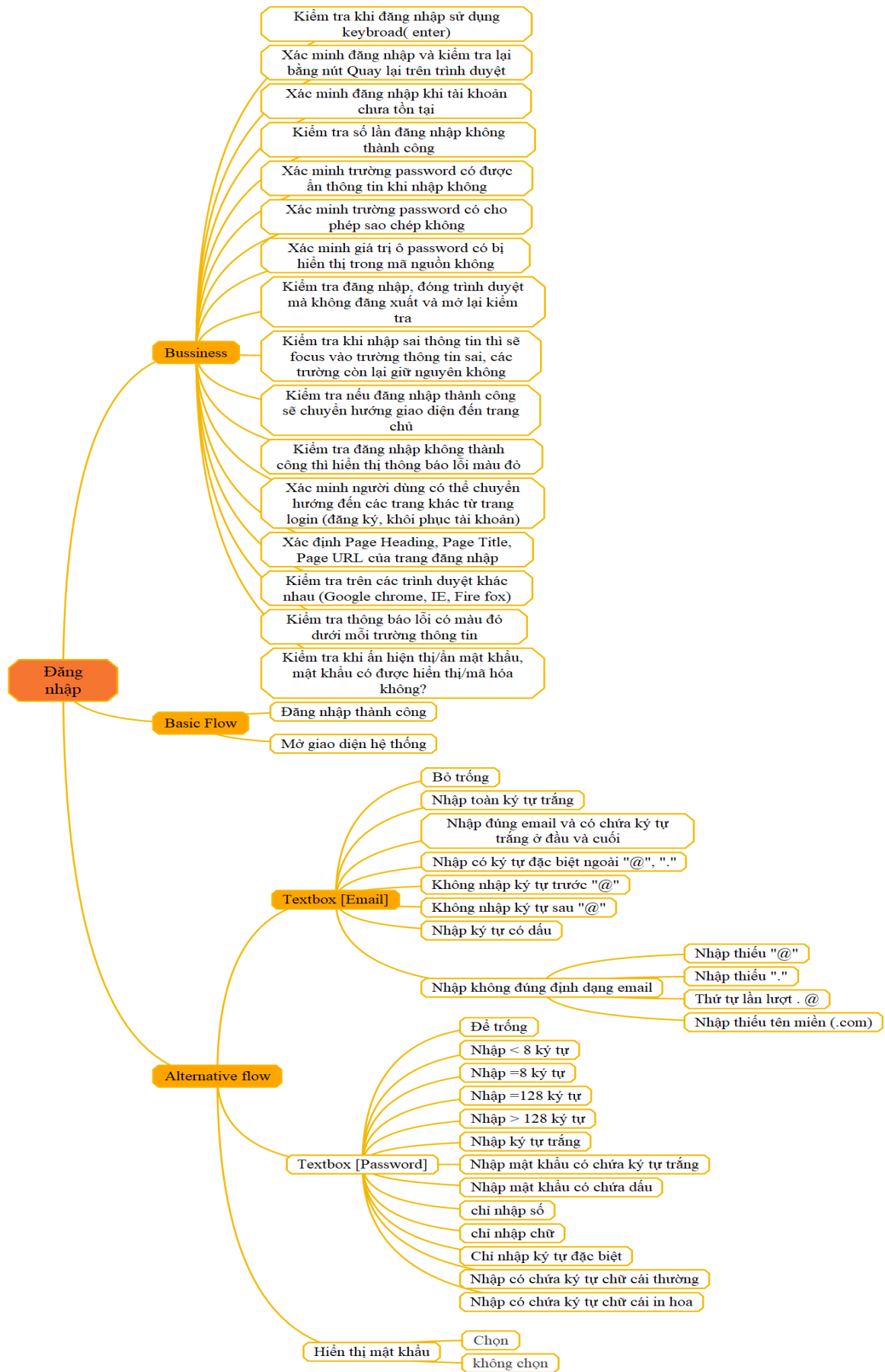


Hình 4. 2 Thiết kế kiểm thử tổng quan hệ thống

4.2.2. Thiết kế kiểm thử chức năng

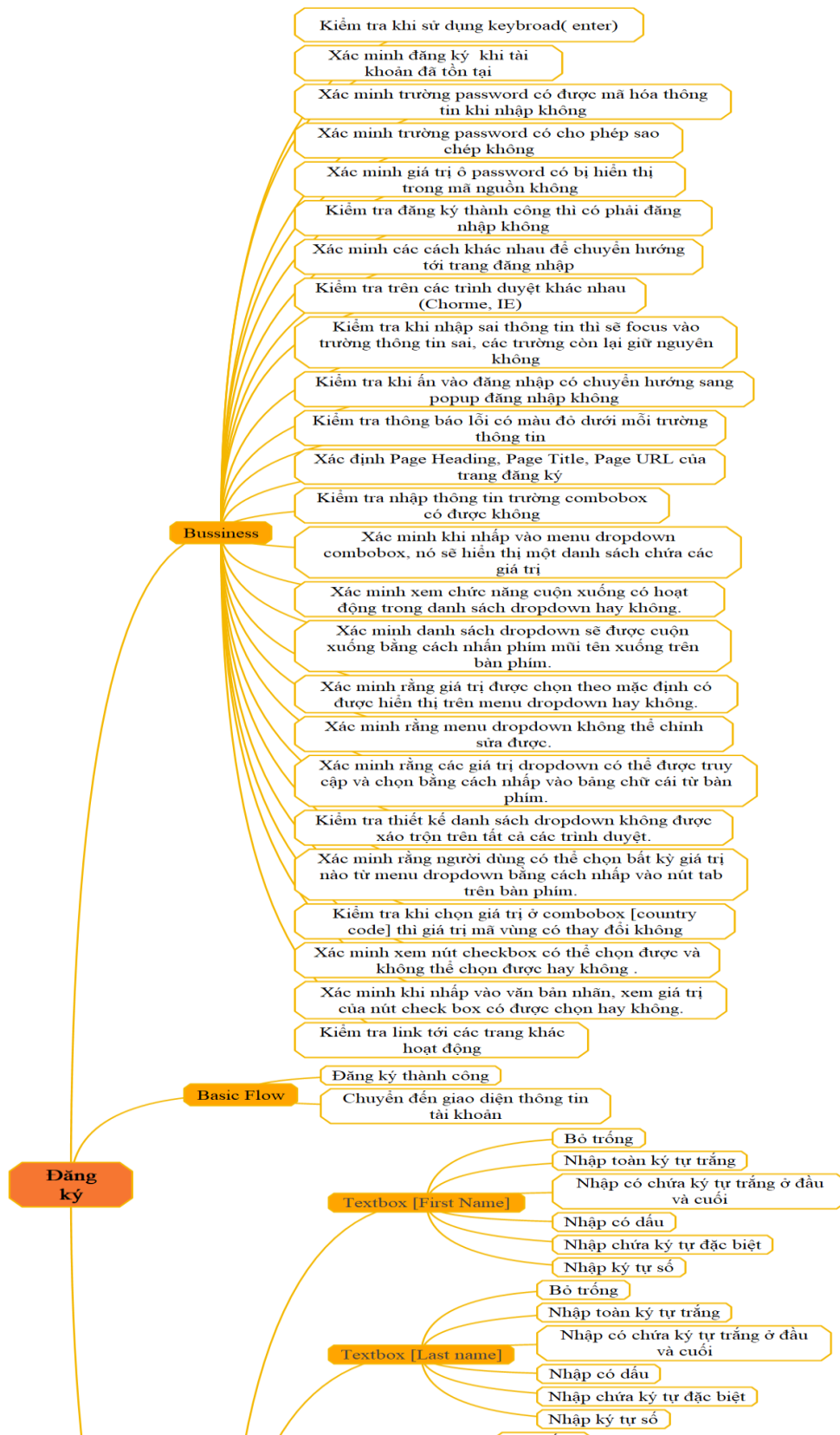
a. Thiết kế kiểm thử chức năng đăng nhập

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

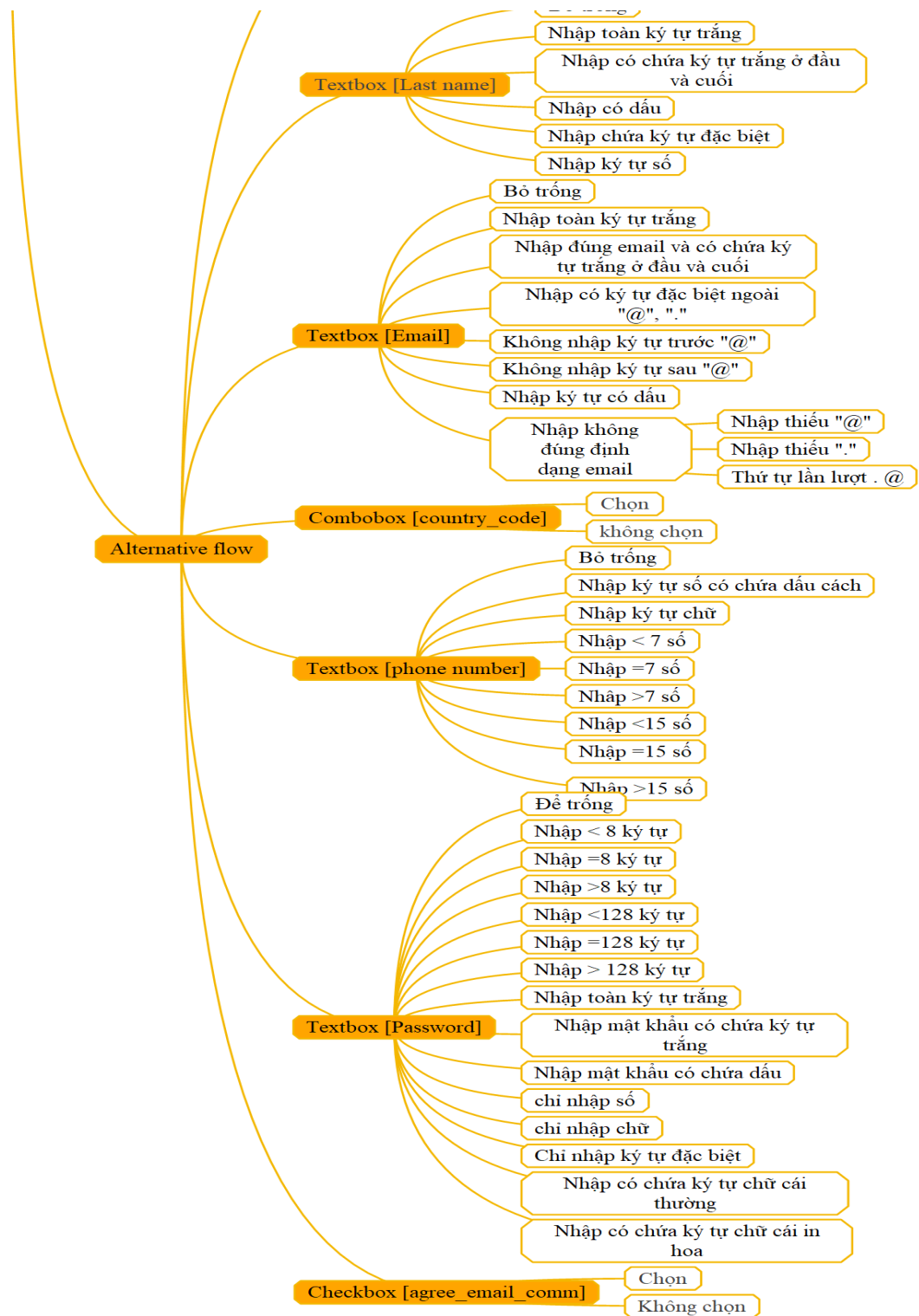


Hình 4.3 Thiết kế kiểm thử chức năng đăng nhập

b. Thiết kế kiểm thử chức năng đăng ký

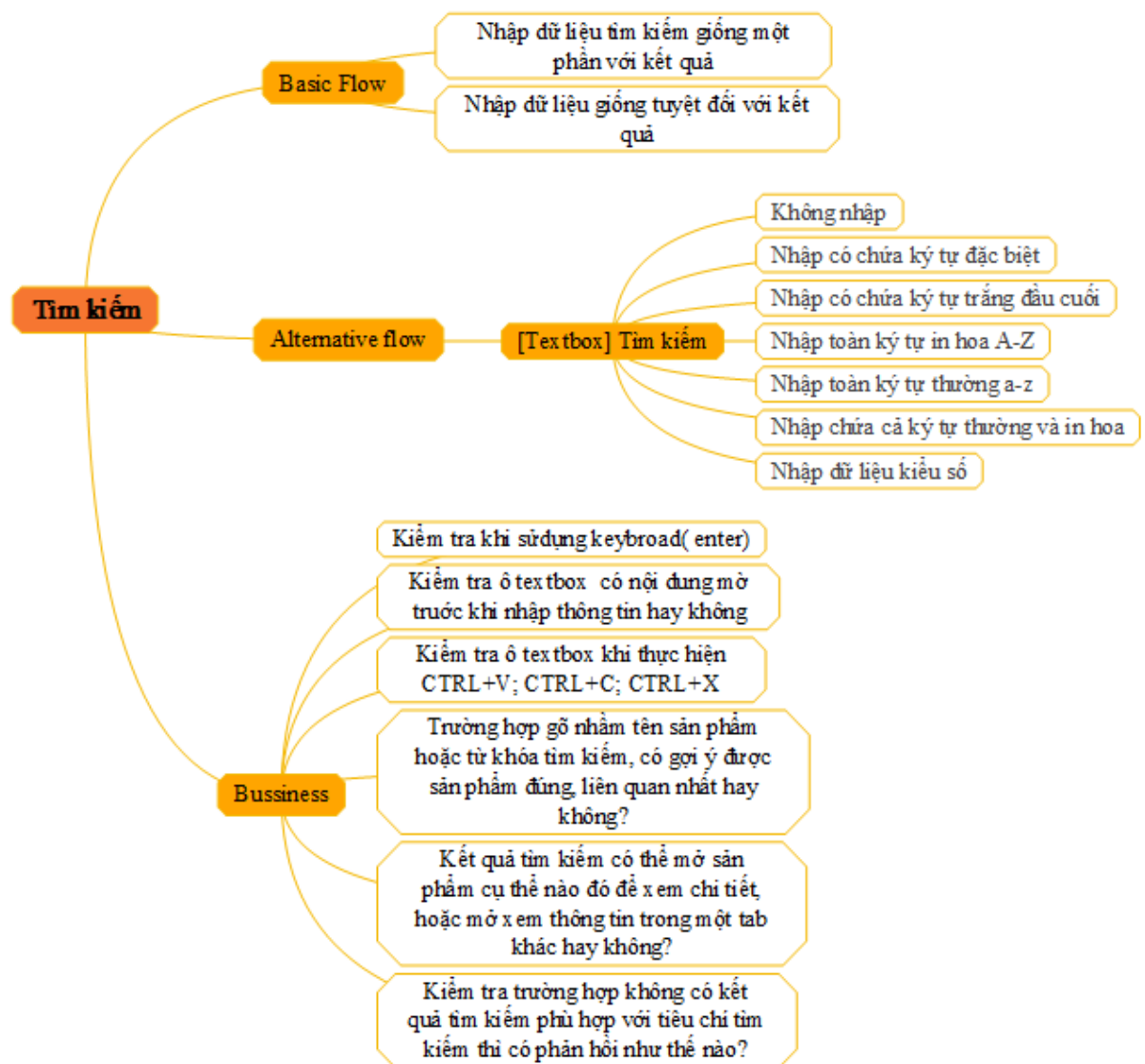


Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn



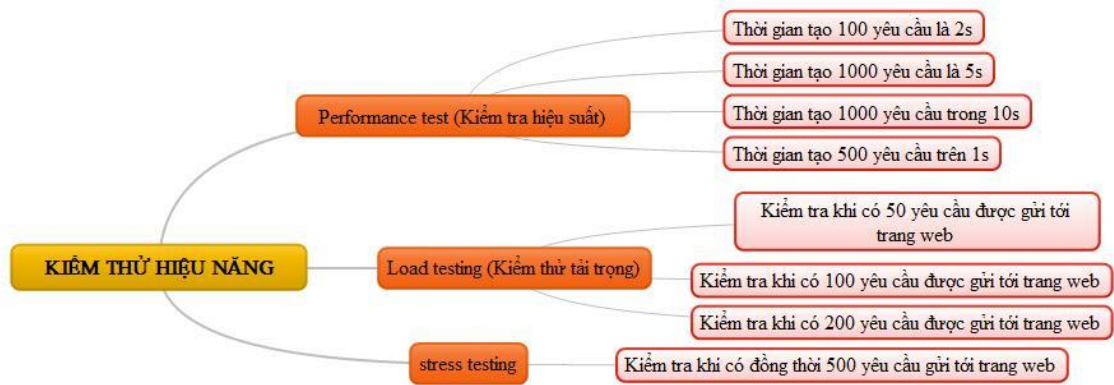
Hình 4. 4 Thiết kế kiểm thử chức năng đăng ký

c. Thiết kế kiểm thử chức năng tìm kiếm



Hình 4. 5 Thiết kế kiểm thử chức năng tìm kiếm

4.2.3. Thiết kế kiểm thử hiệu năng



Hình 4. 6 Thiết kế kiểm thử hiệu năng

4.3. Xây dựng ca kiểm thử

4.2.1 Trường hợp kiểm thử chức năng đăng ký

Bảng 4. 1 Trường hợp kiểm thử chức năng đăng ký

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
R-02	Bỏ trống Textbox First name		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "argee_email_comm" 6. Nhấn nút [Create Account]	Hiện thị thông báo "Please fill in all required fields"			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
R-03	Nhập [first name] toàn ký tự trắng		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "agree_email_comm" 6. Nhấn nút [Create Account]	Hiện thị thông báo "Please Provide a valid first name in input field."			
R-04	Nhập [first name] có chứa ký tự trắng ở đầu và cuối		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone	đăng ký thành công			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
			number","Password","argee_email_comm" 6. Nhấn nút [Create Account]				
R-05	Nhập [first name] chứa ký tự đặc biệt		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email","country_code","phone number","Password","argee_email_comm" 6. Nhấn nút [Create Account]	không cho phép nhập ký tự đặc biệt			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
R-06	Nhập [first name] hoặc [last name] là ký tự chữ số		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "argee_email_comm" 6. Nhấn nút [Create Account]	không cho phép nhập ký tự chữ số			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
R-07	Bỏ trống Textbox Last name		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "agree_email_comm" 6. Nhấn nút [Create Account]	thông báo "Please fill in all required fields"			
R-08	Nhập [Last name] toàn ký tự trắng		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone	Thông báo "Please Provide a valid last name in input field."			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
			number", "Password", "argee_email_comm" 6. Nhấn nút [Create Account]				
R-09	Nhập [Last name] có chứa ký tự trắng ở đầu và cuối		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "argee_email_comm" 6. Nhấn nút [Create Account]	đăng ký thành công			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
R-10	Nhập [Last name] chứa ký tự đặc biệt		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "agree_email_comm" 6. Nhấn nút [Create Account]	không cho phép nhập ký tự đặc biệt			
R-11	Bỏ trống textbox [Email Address]		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "country_code", "phone	thông báo "Please fill in all required fields"			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
			number","Password","argee_email_comm" 6. Nhấn nút [Create Account]				
R-12	Trường [Email Address] nhập toàn ký tự trắng, trường dữ liệu khác nhập hợp lệ		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email","country_code","phone number","Password","argee_email_comm" 6. Nhấn nút [Create Account]	Thông báo "Please enter a valid email address"			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
R-13	Trường [Email Address] nhập có chứa ký tự trắng đầu cuối, trường dữ liệu khác nhập hợp lệ		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "agree_email_comm" 6. Nhấn nút [Create Account]	đăng ký thành công			
R-14	Trường [Email Address] nhập có chứa ký tự đặc biệt, trường		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone	Thông báo "Please enter a valid email address"			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
	dữ liệu khác nhập hợp lệ		number","Password","argee_email_comm" 6. Nhấn nút [Create Account]				
R-15	Dấu chấm "." ở đầu giá trị [Email Address], trường dữ liệu khác nhập hợp lệ		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email","country_code","phone number","Password","argee_email_comm" 6. Nhấn nút [Create Account]	Hiển thị thông báo "Please enter a valid email address"			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
R-16	Nhập thừa @, trường dữ liệu khác nhập hợp lệ		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone number", "Password", "argee_email_comm" 6. Nhấn nút [Create Account]	Hiển thị thông báo "Please enter a valid email address"			
R-17	Thiếu tên miền cấp cao nhất (.com/ .net/ .org/ etc), trường dữ		1. Mở màn hình trang chủ 2. Nhấn nút [Login] 3. Nhấn [Sign up] 4. Nhấn nút [Sign up with Email] 5. Nhập dữ liệu vào ô "First name", "last name", "email", "country_code", "phone	Hiển thị thông báo "Please enter a valid email address"			

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

Test case ID	Test Case Name	Pre-condition	Test Case Steps	Expected Output	Actual Output	Test result	Bug ID
	liệu khác nhập hợp lệ		number", "Password", "argee_email_comm" 6. Nhấn nút [Create Account]				

4.4. Xây dựng dữ liệu kiểm thử

4.3.1 Dữ liệu kiểm thử chức năng đăng ký

Bảng 4. 2 Dữ liệu kiểm thử chức năng đăng ký

testcase ID	Testsuite	First name	Last name	Email Address	phone number	Password	expectedresult
R-01	TS1	Lan	Nguyen	akay05021@gmail.com	0945792644	Lankay33	Dashboard
R-02	TS_register01		Nguyen	nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Please fill in all required fields
R-03	TS_register04		Nguyen	nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Please Provide a valid first name in input field.

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

testcaseID	Testsuite	First name	Last name	Email Address	phone number	Password	expectedresult
R-04	TS1	Lan	Nguyen	nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Dashboard
R-07	TS_register01	Lan		nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Please fill in all required fields
R-08	TS_register04	Lan		nguyenlinh2s001hy@gmail.com	0945792644	Lankay34	Please Provide a valid last name in input field.
R-09	TS1	Lan	Nguyen	nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Dashboard
R-10	TS1	Lan	nguyen122#2@	nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Dashboard
R-11	TS_register01	Lan	Nguyen		0945792644	Lankay34	Please fill in all required fields
R-12	TS_register01	Lan	Nguyen		0945792644	Lankay34	Please enter a valid email address
R-13	TS_register01	Lan	Nguyen	nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Please enter a valid email address
R-14	TS_register01	Lan	Nguyen	akay#0502@gmail.com	0945792644	Lankay34	Please enter a valid email address
R-15	TS_register01	Lan	Nguyen	.nguyenlinh2001hy@gmail.com	0945792644	Lankay34	Please provide valid email address.
R-16	TS_register01	Lan	Nguyen	akay0502@@gmail.com	0945792644	Lankay34	Please enter a valid email address
R-17	TS_register01	Lan	Nguyen	akay0502@gmail	0945792644	Lankay34	Please enter a valid email address

4.3.2. Kịch bản kiểm thử cho chức năng đăng ký

Bảng 4. 3 Kịch bản kiểm thử chức năng đăng ký

TC_ID	Testsuit_ID	Description	Action Keyword	Locator or Type	Locator Value	Test Data
TC_01	TS_register01	Mở trình duyệt	openBrowser	null	null	Chrome
TC_01	TS_register01	Chuyển tới trang Web	navigate	null	null	https://accounts.simplilearn.com/user/register-email
TC_01	TS_register01	Nhập first name	setText	xpath	//input[@placeholder='First Name*']	varfirstNameregister
TC_01	TS_register01	Nhập last name	setText	xpath	//input[@placeholder='Last Name*']	varlastNameregister
TC_01	TS_register01	Nhập email address	setText	xpath	//input[@placeholder='Email Address*']	varEmailregister
TC_01	TS_register01	Chọn mã vùng	clickElement	xpath	//select[@id='country_code']	
TC_01	TS_register01	Chọn item	selectOptionByValue	id	country_code	VN
TC_01	TS_register01	Nhập phone number	setText	xpath	//input[@placeholder='Phone Number*']	varphoneNumberRegister
TC_01	TS_register01	Nhập password	setText	xpath	//input[@id='password']	varPasswordRegister

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

TC_ID	Testsuit_ID	Description	Action Keyword	Locat or Type	Locator Value	Test Data
TC_0 1	TS_register 01	Chọn Đồng ý điều khoản và dịch vụ	selectCheckbox	xpath	//*[contains(text(),'I agree to the')]	
TC_0 1	TS_register 01	Chọn Create Account	clickElement	id	btn_register	
TC_0 1	TS_register 01	So sánh kết quả mong đợi	verifyElementTex t	id	error_box	varresult
TC_0 1	TS_register 01	Đóng trình duyet	closeBrower			

Xây dựng framework kiểm thử tự động và ứng dụng kiểm thử cho website Simplilearn.

4.5. Thực thi và báo cáo kiểm thử

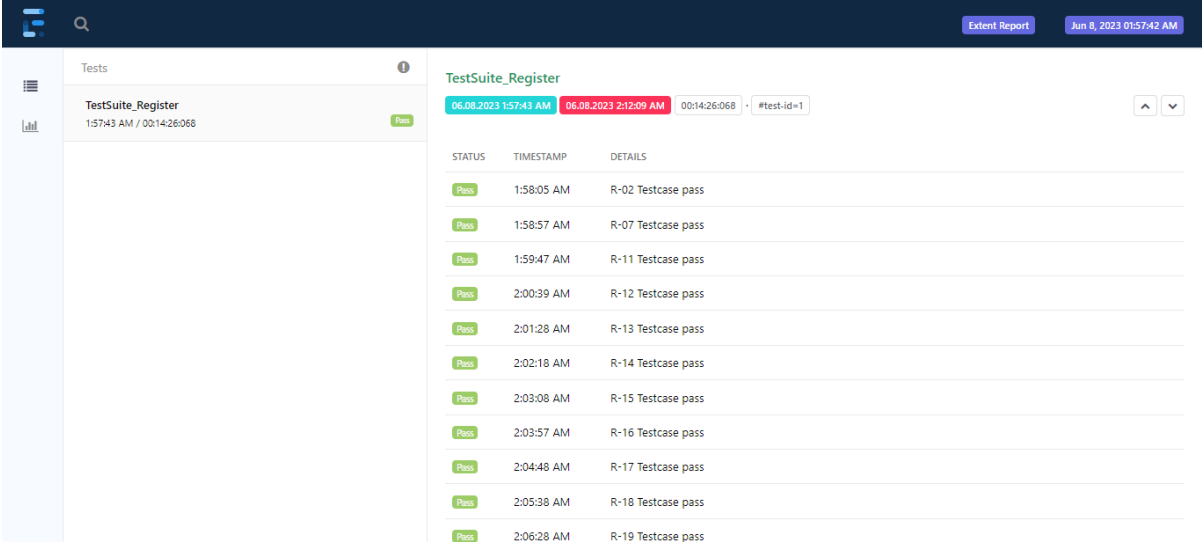
Để thực thi các yêu cầu kiểm thử, em đã thực hiện theo các bước sau:

1. Tạo một file ExecutionEngine.java để thực thi kiểm thử. Sử dụng Excel reader để đọc dữ liệu từ Test Data Repository và Keyword Repository. Xây dựng các keyword trong ActionKeyword.java để thực hiện các hành động kiểm thử.

2. Thực thi kiểm thử trong file ExecutionEngine.java, đọc dữ liệu từ Test Data Repository và Keyword Repository. Dựa trên các keyword được đọc, gọi các phương thức tương ứng trong ActionKeyword.java để thực hiện các hành động kiểm thử.

3. Sau mỗi testcase được thực thi, ghi lại trạng thái vào file logs, sử dụng các lớp hỗ trợ của log4j để ghi dữ liệu vào file. Trong trường hợp testcase có trạng thái fail, em sử dụng phương thức getScreenshotAs() của thư viện TakesScreenshot Selenium WebDriver để chụp lại màn hình giao diện tại thời điểm phát hiện lỗi vào folder Screenshots.

4. Sau khi thực thi kiểm thử hoàn tất, em sử dụng thư viện ExtentReports để tạo một file HTML report (ExtentReport.html) và lưu trạng thái của các testcase vào đó.



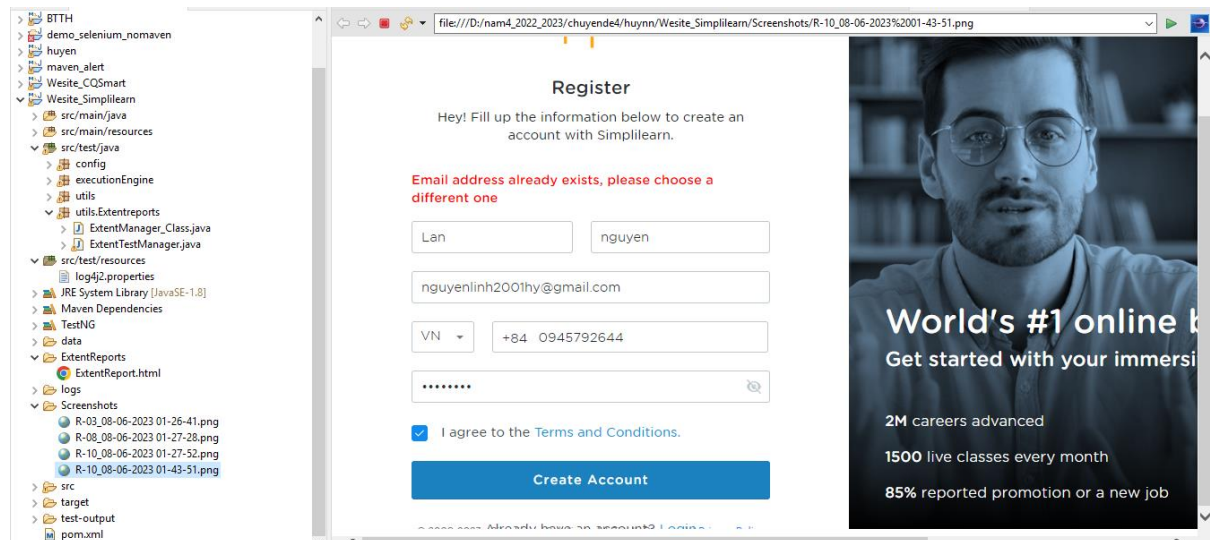
TestSuite_Register		
06.08.2023 1:57:43 AM 06.08.2023 2:12:09 AM 00:14:26:068 #test-id=1		
STATUS	TIMESTAMP	DETAILS
Pass	1:58:05 AM	R-02 Testcase pass
Pass	1:58:57 AM	R-07 Testcase pass
Pass	1:59:47 AM	R-11 Testcase pass
Pass	2:00:39 AM	R-12 Testcase pass
Pass	2:01:28 AM	R-13 Testcase pass
Pass	2:02:18 AM	R-14 Testcase pass
Pass	2:03:08 AM	R-15 Testcase pass
Pass	2:03:57 AM	R-16 Testcase pass
Pass	2:04:48 AM	R-17 Testcase pass
Pass	2:05:38 AM	R-18 Testcase pass
Pass	2:06:28 AM	R-19 Testcase pass

Hình 4. 7 Test report (ExtentReport.html) của chức năng đăng ký

Xây dựng framework kiểm thử tự động ứng dụng kiểm thử cho website Simplilearn

```
1 [INFO] 08-06-2023 01:57:42 [main] LogHelpers - Start testing Default test
2 [INFO] 08-06-2023 01:57:42 [main] LogHelpers - TestSuite_Register test is starting...
3 [INFO] 08-06-2023 01:58:05 [main] LogHelpers - R-02 Testcase pass
4 [INFO] 08-06-2023 01:58:57 [main] LogHelpers - R-07 Testcase pass
5 [INFO] 08-06-2023 01:59:47 [main] LogHelpers - R-11 Testcase pass
6 [INFO] 08-06-2023 02:00:39 [main] LogHelpers - R-12 Testcase pass
7 [INFO] 08-06-2023 02:01:28 [main] LogHelpers - R-13 Testcase pass
8 [INFO] 08-06-2023 02:02:18 [main] LogHelpers - R-14 Testcase pass
9 [INFO] 08-06-2023 02:03:08 [main] LogHelpers - R-15 Testcase pass
10 [INFO] 08-06-2023 02:03:57 [main] LogHelpers - R-16 Testcase pass
11 [INFO] 08-06-2023 02:04:48 [main] LogHelpers - R-17 Testcase pass
12 [INFO] 08-06-2023 02:05:38 [main] LogHelpers - R-18 Testcase pass
13 [INFO] 08-06-2023 02:06:28 [main] LogHelpers - R-19 Testcase pass
14 [INFO] 08-06-2023 02:07:18 [main] LogHelpers - R-20 Testcase pass
15 [INFO] 08-06-2023 02:08:07 [main] LogHelpers - R-21 Testcase pass
16 [INFO] 08-06-2023 02:08:58 [main] LogHelpers - R-24 Testcase pass
17 [INFO] 08-06-2023 02:09:19 [main] LogHelpers - R-30 Testcase pass
18 [INFO] 08-06-2023 02:10:08 [main] LogHelpers - R-31 Testcase pass
19 [INFO] 08-06-2023 02:10:28 [main] LogHelpers - R-36 Testcase pass
20 [INFO] 08-06-2023 02:10:48 [main] LogHelpers - R-37 Testcase pass
21 [INFO] 08-06-2023 02:11:48 [main] LogHelpers - R-39 Testcase pass
22 [INFO] 08-06-2023 02:12:08 [main] LogHelpers - R-40 Testcase pass
23 [INFO] 08-06-2023 02:12:09 [main] LogHelpers - TestSuite_Register test is passed.
24 [INFO] 08-06-2023 02:12:09 [main] LogHelpers - End testing Default test
25
```

Hình 4. 8 Ghi trạng thái testcase vào file log



Hình 4. 9 Ảnh chụp màn hình lưu tại folder Screenshots khi testcase fail

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết quả đạt được

Khi xây dựng framework kiểm thử tự động và ứng dụng kiểm thử website Simplilearn, tôi đã đạt được những kết quả quan trọng sau:

1. Xây dựng framework kiểm thử tự động hướng hybrid driven testing, kết hợp cả việc sử dụng các công cụ kiểm thử tự động Selenium WebDriver và viết các kịch bản kiểm thử bằng ngôn ngữ lập trình Java.
2. Xây dựng được bộ tài liệu chi tiết cho quá trình kiểm thử của ứng dụng web Simplilearn bao gồm các phần như thiết kế kiểm thử, tài liệu đặc tả, dữ liệu kiểm thử, kịch bản kiểm thử và testcase.
3. Xây dựng hoàn thiện chức năng đọc ghi file Excel, cho phép quản lý các bộ dữ liệu kiểm thử trong các tệp Excel dễ dàng và linh hoạt, có thể đọc dữ liệu kiểm thử từ các bảng trong file Excel và sử dụng để thực thi các kịch bản kiểm thử.
4. Xây dựng được báo cáo kiểm thử: Sau khi thực hiện các kịch bản kiểm thử, framework tự động tạo báo cáo chi tiết về kết quả kiểm thử, các lỗi và sự cố phát hiện được.

Hạn chế của đề tài

Tuy nhiên có một số hạn chế mà tôi gặp phải khi xây dựng framework kiểm thử tự động trên ứng dụng web Simplilearn:

1. Framework chưa tối ưu code, mã nguồn của framework chưa được tối ưu hóa hiệu quả. Điều này gây ra tình trạng tăng thời gian thực thi kiểm thử và tiêu thụ tài nguyên hệ thống không cần thiết.
2. Độ bao phủ chưa cao do chưa định nghĩa đầy đủ các trường hợp kiểm thử, chưa thực hiện đầy đủ các kịch bản kiểm thử cho mọi tình huống.

Hướng phát triển của đề tài

Tiếp tục nghiên cứu và phát triển framework để tối ưu hóa mã nguồn và hiệu suất, cải thiện thuật toán và cấu trúc dữ liệu và tăng cường việc xây dựng và thực hiện các kịch bản kiểm thử để đạt được độ bao phủ cao hơn.

Nghiên cứu và mở rộng để hỗ trợ nhiều nền tảng và trình duyệt khác tăng tính linh hoạt và khả năng tái sử dụng của framework trên các môi trường khác nhau.

Những hướng phát triển trên có thể giúp cải thiện framework kiểm thử tự động trên ứng dụng web Simplilearn, từ đó đảm bảo chất lượng và độ tin cậy của ứng dụng.

TÀI LIỆU THAM KHẢO

- [1] Bộ môn CNPM, đề cương “Kiểm thử phần mềm”, ĐHSPKT Hưng Yên, năm 2016
- [2] Bộ môn CNPM, Đề cương “Kiểm thử ứng dụng web và di động”, ĐHSPKT Hưng Yên, năm 2016.
- [3] <https://viblo.asia/p/cac-test-automation-framework-pho-bien-nhat-hien-nay-va-uu-nhuoc-diem-cua-tung-loai-4dbZNpkQ5YM>
- [4] <https://viblo.asia/p/data-driven-testing-framework-6J3Zg9Ng5mB>
- [5] <https://anhtester.com/lesson/selenium-java-bai-30-take-screenshot-va-record-video-sau-khi-chay-test>
- [6] <https://anhtester.com/lesson/selenium-java-bai-23-thuc-hanh-viet-ham-xu-ly-chung-nang-cao-de-dung-lai>
- [7] <https://anhtester.com/lesson/selenium-java-bai-32-su-dung-log4j-de-ghi-log-vao-file-trong-selenium-java>
- [8] <https://anhtester.com/blog/selenium-java/selenium-java-bai-31-cai-dat-va-su-dung-extent-report-allure-report>
- [9] <https://www.guru99.com/handling-iframes-selenium.html>
- [10] <https://stackoverflow.com/questions/9588827/how-to-switch-to-the-new-browser-window-which-opens-after-click-on-the-button>