

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC KINH TẾ
KHOA THỐNG KÊ - TIN HỌC



BÁO CÁO THỰC TẬP NGHỀ NGHIỆP
NGÀNH HỆ THỐNG THÔNG TIN QUẢN LÝ
CHUYÊN NGÀNH QUẢN TRỊ HỆ THỐNG THÔNG TIN
TÊN ĐỀ TÀI:
AUTOMATION TESTING FOR THE SHOPPING WEBSITE
WITH ROBOT FRAMEWORK

Sinh viên thực hiện : Trần Xuân Thiện
Lớp : 46K21.3
Đơn vị thực tập : TMA Solutions
Cán bộ hướng dẫn : Từ Minh Hoàng
Giảng viên hướng dẫn : TS. Hoàng Thị Thanh Hà

Đà Nẵng, tháng 08 năm 2023

CÔNG TY TNHH GIẢI PHÁP
PHẦN MỀM TƯỜNG MINH
BÌNH ĐỊNH

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP

Họ và tên sinh viên: TRẦN XUÂN THIÊN Lớp: 46K21.3

Khoa Thống kê – Tin học, Trường Đại học Kinh tế, Đại học Đà Nẵng

Thực tập từ ngày: 19/6/2023 đến ngày 25/8/2023

Tên đơn vị thực tập: Công ty TNHH Giải pháp Phần mềm Tường Minh Bình Định (TMA Solutions Bình Định)

Địa chỉ: 12 Đại lộ Khoa học, KV 2, phường Ghềnh Ráng, thành phố Quy Nhơn

Số điện thoại liên hệ: (0256) 389 8979 – Máy lẻ: 7222 | Hotline: 0977 465 083

Họ tên cán bộ hướng dẫn: Từ Minh Hoàng

Sau quá trình thực tập của sinh viên tại đơn vị, chúng tôi có một số đánh giá như sau:

STT	Nội dung đánh giá	Rất không tốt	Không tốt	Bình thường	Tốt	Rất tốt
1	Về thái độ, ý thức, đạo đức và việc tuân thủ các quy định và văn hóa đơn vị thực tập	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
2	Kiến thức chuyên môn	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	Khả năng hòa nhập, thích nghi và tác phong nghề nghiệp	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4	Trách nhiệm, sáng tạo trong công việc	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

(Anh/chị vui lòng đánh dấu X vào ô tương ứng với năng lực của sinh viên)

Ý kiến nhận xét và đề xuất (Nhằm nâng cao chất lượng đào tạo, Nhà trường rất mong muốn nhận thêm những ý kiến khác từ quý doanh nghiệp):

Nhận xét:

- Kiến thức về chuyên môn ở mức khá.
- Có sự tự tin và khả năng thuyết trình tốt.
- Có khả năng tìm hiểu và học hỏi những kiến thức mới.
- Hoàn thành tốt công việc được giao.
- Có khả năng làm việc nhóm và độc lập ở mức tốt.
- Cần cải thiện thêm khả năng nghe, nói tiếng Anh và một số kỹ năng khác để đáp ứng nhu cầu công việc.

Đề xuất: Không

Xác nhận của đơn vị thực tập



Lâm Thị Thanh Thảo

Quy Nhơn, ngày 25 tháng 8 năm 2023

Người hướng dẫn



Từ Minh Hoàng

LỜI CẢM ƠN

Kính gửi quý thầy/cô,

Em xin gửi lời cảm ơn chân thành đến quý thầy/cô đã dành thời gian quý báu để đọc và đánh giá báo cáo thực tập của em. Bởi vì sự hướng dẫn và phản hồi của quý thầy/cô đã giúp tôi hoàn thành báo cáo một cách tốt nhất.

Em xin cảm ơn sự tận tâm và những kiến thức quý báu mà quý thầy/cô đã truyền đạt trong suốt quá trình thực tập của em. Quý thầy/cô đã cho em những cơ hội quý giá để áp dụng những kiến thức đã học và trải nghiệm thực tế trong lĩnh vực của mình. Điều này không chỉ giúp em nắm vững kiến thức chuyên ngành mà còn định hình và phát triển khả năng làm việc nhóm, tư duy phân tích, và giải quyết vấn đề.

Lời cảm ơn đặc biệt dành cho các thành viên của công ty TMA Solutions Bình Định đã tạo điều kiện thuận lợi cho em thực hiện thực tập. Sự giúp đỡ và sự hỗ trợ từ các đồng nghiệp và đồng đội của em đã tạo ra một môi trường làm việc tích cực và động lực để em phát triển và đạt được mục tiêu của mình trong thời gian thực tập.

Một lần nữa, xin chân thành cảm ơn quý thầy/cô và tất cả những người đã giúp đỡ em trong thực tập. Em xin cam đoan rằng báo cáo này là công trình nghiên cứu của riêng em, được hoàn thành dựa trên kiến thức và kinh nghiệm cá nhân. Tất cả các nguồn tài liệu và thông tin từ các tác giả khác đã được trích dẫn đúng quy định và không có vi phạm bản quyền hay nguyên tắc đạo đức nghiêm trọng.

Xin chân thành cảm ơn và kính chúc quý thầy/cô sức khỏe và hạnh phúc!

Trân trọng,

Trần Xuân Thiện

LỜI CAM ĐOAN

Em xin cam đoan:

1. Nội dung trong báo cáo thực tập nghề nghiệp với đề tài “**Automation testing for the shopping website with robot framework**” là kết quả thực hiện dưới sự hướng dẫn của T.S Hoàng Thị Thanh Hà và mentor Từ Minh Hoàng không có sự sao chép từ bất kỳ nguồn nào khác.
2. Ngoài ra, trong bài báo cáo có sử dụng một số nguồn tài liệu tham khảo đã được trích dẫn nguồn và chú thích rõ ràng.

Em xin hoàn toàn chịu trách nhiệm trước bộ môn, khoa và nhà trường về sự cam đoan này.

Đà Nẵng, ngày 20 tháng 8 năm 2023

MỤC LỤC

LỜI CẢM ƠN	iv
LỜI CAM ĐOAN	v
DANH MỤC HÌNH ẢNH	ix
DANH MỤC BẢNG BIỂU	xi
LỜI MỞ ĐẦU	xii
CHƯƠNG 1: TỔNG QUAN VỀ CÔNG TY TMA VÀ VỊ TRÍ TESTER	1
1.1. Tổng quan về Công ty TMA Bình Định.....	1
1.1.1. Giới thiệu về công ty	1
1.1.2. Tầm nhìn và sứ mệnh	1
1.1.3. Giá trị cốt lõi.....	2
1.2. Tổng quan về vị trí Tester	2
1.2.1. Mô tả về vị trí Tester	2
1.2.2. Các kỹ năng cần có của Tester	2
1.2.3. Cơ hội nghề nghiệp.....	3
1.2.4. Mức lương của vị trí Tester	4
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VỀ KIỂM THỬ PHẦN MỀM	6
2.1. Tổng quan về kiểm thử phần mềm	6
2.1.1. Giới thiệu về kiểm thử phần mềm và mục tiêu kiểm thử.....	6
2.1.2. Nguyên tắc của kiểm thử phần mềm	6
2.1.3. Vòng đời phát triển phần mềm (SDLC)	8
2.2. Loại kiểm thử phần mềm.....	10
2.2.1. Manual testing (Kiểm thử thủ công).....	10
2.2.2. Automation testing (Kiểm thử tự động)	11
2.2.3. Security testing (Kiểm thử bảo mật).....	11
2.2.4. API testing (Kiểm thử API)	12
2.3. Phương pháp kiểm thử	12

2.3.1. Static testing (Kiểm thử tĩnh)	12
2.3.2. Dynamic testing (Kiểm thử động)	13
2.3.3. White box testing (Kiểm thử hộp trắng).....	13
2.3.4. Black box testing (Kiểm thử hộp đen).....	14
2.4. Cấp độ của kiểm thử.....	15
2.4.1. Unit testing (Kiểm thử đơn vị)	15
2.4.2. Integration testing (Kiểm thử tích hợp)	15
2.4.3. System testing (Kiểm thử hệ thống)	15
2.4.4. Acceptance Testing (Kiểm thử người dùng)	16
2.5. Tổng quan về Python	16
2.5.1. Python là gì?	16
2.5.2. Ứng dụng của Python	16
2.6. Tổng quan về Robot Framework.....	17
2.6.1. Robot Framework là gì ?	17
2.6.2. Ưu điểm và nhược điểm của Robot Framework	18
CHƯƠNG 3: TRIỂN KHAI DỰ ÁN “AUTOMATION TESTING FOR THE SHOPPING WEBSITE WITH ROBOT FRAMEWORK”	19
3.1. Tổng quan hệ thống “Shopping Website”	19
3.2. Đặc tả yêu cầu.....	20
3.2.1. Chức năng tại màn hình “Đăng ký (Register)”	20
3.2.1.1. Đặc tả màn hình “Đăng ký (Register)”	20
3.2.1.2. Usecase “Đăng ký (Register)”	21
3.2.1.3. Đặc tả usecase “Đăng ký (Register)”	21
3.2.2. Chức năng tại màn hình “Đăng nhập (Login)”	21
3.2.2.1. Đặc tả màn hình “Đăng nhập (Login)”	21
3.2.2.2. Usecase “Đăng nhập (Login)”	21
3.2.2.3. Đặc tả usecase “Đăng nhập (Login)”	22
3.2.3. Chức năng tại màn hình “Home”	22

3.2.3.1. Đặc tả màn hình “Home”	22
3.2.3.2. Usecase màn hình “Home”	23
3.2.3.3. Đặc tả Usecase “Home”	23
3.2.4. Chức năng tại màn hình “Giỏ hàng (Items)”	24
3.2.4.1. Đặc tả chức năng tại màn hình “Giỏ hàng (Items)”	24
3.2.4.2. Usecase màn hình “Giỏ hàng (Items)”	25
3.3. Thiết kế Testcase	26
3.3.1. Testcase chức năng tại màn hình “Đăng ký (Register)”	26
3.3.2. Testcase chức năng tại màn hình “Đăng nhập (Login)”	29
3.3.3. Testcase chức năng tại màn hình “Home”	33
3.3.4. Testcase chức năng tại màn hình “Items”	38
3.4. Chạy code test tự động.....	44
3.4.1. Keywords của code.....	44
3.4.2. Testcase trên Robot framework.....	50
3.4.2.1. Testcase màn hình “Register”	50
3.4.2.2. Testcase màn hình “Login”	51
3.4.2.3. Testcase màn hình “Home”	52
3.4.2.4. Testcase màn hình “Items”	54
3.5. Kết quả test	57
3.5.1. Kết quả test màn hình “Register”	57
3.5.2. Kết quả test màn hình “Login”	57
3.5.3. Kết quả test màn hình “Home”	58
3.5.4. Kết quả test màn hình “Items”	59
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	60
TÀI LIỆU THAM KHẢO.....	61
PHỤ LỤC	62

DANH MỤC HÌNH ẢNH

Hình 2. 1. 1: Quy trình SDLC	9
Hình 3. 1. 1: Mô hình hệ thống	20
Hình 3. 2. 1: Màn hình đăng nhập đăng ký	20
Hình 3. 2. 2: Màn hình Home.....	23
Hình 3. 2. 3: Màn hình giỏ hàng.....	25
Hình 3. 4. 1: Keywords chung của code.....	44
Hình 3. 4. 2: Keywords của màn hình đăng ký “Register”	45
Hình 3. 4. 3: Keywords của màn hình đăng ký “Register”	45
Hình 3. 4. 4: Keywords của màn hình đăng ký “Login”	46
Hình 3. 4. 5: Keywords của màn hình đăng ký “Login”	46
Hình 3. 4. 6: Keywords của màn hình đăng ký “Home”	47
Hình 3. 4. 7: Keywords của màn hình đăng ký “Home”	47
Hình 3. 4. 8: Keywords của màn hình đăng ký “Login”	48
Hình 3. 4. 9: Keywords của màn hình đăng ký “Items”	48
Hình 3. 4. 10: Keywords của màn hình đăng ký “Items”	49
Hình 3. 4. 11: Keywords của màn hình đăng ký “Items”	49
Hình 3. 4. 12: Testcase 1 đăng ký tài khoản thành công.....	50
Hình 3. 4. 13: Testcase 2 đăng ký tài khoản thất bại.....	50
Hình 3. 4. 14: Testcase 3 đăng ký tài khoản thất bại khi bỏ trống email	50
Hình 3. 4. 15: Testcase 4 đăng ký tài khoản thất bại khi bỏ trống mật khẩu	50
Hình 3. 4. 16: Testcase 5 đăng ký tài khoản thất bại khi bỏ trống email và mật khẩu.....	51
Hình 3. 4. 17: Testcase 1 đăng nhập thành công.....	51
Hình 3. 4. 18: Testcase 2 đăng nhập thất bại khi tài khoản và mật khẩu sai.....	51
Hình 3. 4. 19: Testcase 3 đăng nhập thất bại khi để trống mật khẩu	51
Hình 3. 4. 20: Testcase 4 đăng nhập thất bại khi để trống tài khoản	52
Hình 3. 4. 21: Testcase 5 đăng nhập thất bại khi để trống tài khoản và mật khẩu	52
Hình 3. 4. 22: Testcase 6 che giấu mật khẩu khi đăng nhập	52
Hình 3. 4. 23: Testcase 7 phân biệt chữ hoa ở mật khẩu.....	52
Hình 3. 4. 24: Testcase 1 home page chỉ có 3 sliders.....	53
Hình 3. 4. 25: Testcase 2 home page chỉ có 3 arrivals	53
Hình 3. 4. 26: Testcase 3 xem thông tin chi tiết của sản phẩm	53
Hình 3. 4. 27: Testcase 4 xem đánh mô tả sản phẩm	53
Hình 3. 4. 28: Testcase 5 xem đánh giá sản phẩm	54
Hình 3. 4. 29: Testcase 6 thêm sản phẩm vào giỏ hàng	54
Hình 3. 4. 30: Testcase 7 thêm sản phẩm thất bại khi đầu vào số lượng bằng	54
Hình 3. 4. 31: Testcase 1 xóa sản phẩm ra khỏi giỏ hàng	55
Hình 3. 4. 32: : Testcase 2 khôi phục sản phẩm đã xóa	55
Hình 3. 4. 33: Testcase 3 cập nhật giỏ hàng.....	55
Hình 3. 4. 34: Testcase 4 cập nhật sản phẩm với số lượng bằng 0	56
Hình 3. 4. 35: Testcase 5 áp dụng mã giảm giá lớn hơn 450srp	56

Hình 3. 4. 36: Testcase 6 áp dụng mã giảm giá nhỏ hơn hoặc bằng 450rsp	56
Hình 3. 4. 37: Testcase 7 bỏ trống ô mã giảm giá.....	57
Hình 3. 5. 1: Kết quả testcase màn hình “Register”	57
Hình 3. 5. 2: Kết quả testcase màn hình “Login”	58
Hình 3. 5. 3: Kết quả testcase màn hình “Home”.....	59
Hình 3. 5. 4: Kết quả testcase màn hình “Items”	59

DANH MỤC BẢNG BIỂU

Bảng 1: Testcase màn hình “Register”	29
Bảng 2: Testcase màn hình “Login”	33
Bảng 3: Testcase màn hình “Home”	38
Bảng 4: Testcase màn hình “Items”	43
Bảng 5: Bảng kết quả màn hình “Register”	57
Bảng 6: Bảng kết quả màn hình “Login”	58
Bảng 7: Bảng kết quả testcase màn hình “Home”	58
Bảng 8: Bảng kết quả testcase màn hình “Items”	59

LỜI MỞ ĐẦU

1. Mục tiêu nghiên cứu của đề tài

Kiểm thử (Testing) với mục đích xác định lỗi hỏng, các lỗi trên hệ thống hoặc các yêu cầu đang bị thiếu hay trái ngược với các yêu cầu thực tế đã được đề ra.

2. Nhiệm vụ của đề tài

- Nghiên cứu lý thuyết: kiểm thử phần mềm, các công cụ được sử dụng hỗ trợ cho việc kiểm thử tự động

- Tìm hiểu cách xây dựng test case trong kiểm thử phần mềm

- Tìm hiểu công cụ Robot Framework trong kiểm thử phần mềm

- Áp dụng những kiến thức được học để ứng dụng thực hiện kiểm thử lên dự án thực tế

3. Phương pháp nghiên cứu

- Tìm hiểu lý thuyết trên Internet

- Được mentor hướng dẫn

- Tham khảo ý kiến và sự hướng dẫn của giáo viên

4. Đối tượng và phạm vi nghiên cứu

- Đối tượng: Trang Web (<https://practice.automationtesting.in/>)

- Thời gian: Nghiên cứu được thực hiện trong 10 tuần (19/06/2023 – 25/08/2023)

- Địa điểm: Công ty TMA Solutions Bình Định

- Lĩnh vực nghiên cứu: Kiểm thử tự động một số chức năng trên trang Web

5. Kết cấu của đề tài

Báo cáo gồm 3 chương:

- Chương 1: Tổng quan về Công ty TMA và vị trí Tester
- Chương 2: Cơ sở lý thuyết về kiểm thử phần mềm
- Chương 3: Triển khai dự án

CHƯƠNG 1: TỔNG QUAN VỀ CÔNG TY TMA VÀ VỊ TRÍ TESTER

1.1. Tổng quan về Công ty TMA Bình Định

1.1.1. Giới thiệu về công ty

Được thành lập năm 1997, TMA là tập đoàn công nghệ hàng đầu Việt Nam với 3500 kỹ sư và khách hàng là những tập đoàn công nghệ cao hàng đầu thế giới từ 30 quốc gia. TMA hiện có 7 chi nhánh tại Việt Nam (6 tại Tp.HCM và 1 ở Tp. Quy Nhơn) cùng 6 chi nhánh ở nước ngoài (Mỹ, Úc, Canada, Đức, Nhật, Singapore).

Tháng 6 năm 2018, TMA đã mở chi nhánh tại Bình Định. Sau 4 năm, TMA Bình Định đã phát triển nhanh chóng với hơn 400 kỹ sư, trong đó có nhiều kỹ sư đang làm việc tại TP.HCM đã trở về làm việc tại quê hương.

Tháng 8 năm 2018, TMA đã khởi công xây dựng Công viên Sáng tạo TMA Bình Định (TMA Innovation Park – TIP) trên 10 hecta tại Thung lũng Sáng tạo Quy Nhơn (Quy Nhơn Innovation Park – SNIVY) với vốn đầu tư hàng trăm tỷ đồng.



1.1.2. Tầm nhìn và sứ mệnh

Là trung tâm phần mềm đầu tiên tại Thung lũng Sáng tạo Quy Nhơn, Công viên Sáng tạo TMA mang sứ mệnh trở thành trung tâm phát triển phần mềm và công nghệ cao hàng đầu tại miền Trung, góp phần quan trọng đưa Thung lũng sáng tạo Quy Nhơn trở thành một điểm đến của công nghệ 4.0 tại Việt Nam. Công viên Sáng tạo TMA bao gồm Trung tâm Phát triển Phần Mềm, Xưởng Phần mềm, Trung tâm R&D, Trung tâm Khoa học Dữ liệu, Học viện Công Nghệ...

1.1.3. Giá trị cốt lõi

- **Sự tôn trọng:** đối xử với người khác theo cách bạn muốn được đối xử
- **Trung thực:** trung thực với những người khác và chính bạn
- **Sự cam kết:** chúng tôi chuyển đổi lời hứa thành hiện thực

1.2. Tổng quan về vị trí Tester

1.2.1. Mô tả về vị trí Tester

Nhân viên Tester là người kiểm thử phần mềm, chịu trách nhiệm phát triển chất lượng và triển khai phần mềm. Họ tham gia vào việc thực hiện các thử nghiệm tự động và thủ công để đảm bảo phần mềm do lập trình viên viết phù hợp với mục đích sử dụng. Một số nhiệm vụ bao gồm phân tích phần mềm và hệ thống, giảm thiểu rủi ro và ngăn ngừa sự cố phần mềm.

Vị trí tester đòi hỏi kiến thức về kiểm thử phần mềm, quy trình phát triển phần mềm, kỹ năng ghi nhận lỗi và báo cáo, cũng như hiểu biết về các công cụ kiểm thử và tự động hóa kiểm thử.

1.2.2. Các kỹ năng cần có của Tester

- **Kỹ năng phân tích:** Muốn trở thành một Tester giỏi, bạn cần có 1 kỹ năng phân tích nhanh nhạy và chính xác, rèn luyện được khả năng nhìn nhận vấn đề từ nhiều góc độ, nhiều hướng. Để có góc nhìn đó, bạn cần phải phân tích yêu cầu khách hàng đưa, phân tích trong từng function nhỏ mà bạn đang test, tìm ra được những điểm mấu chốt, quan trọng, dự đoán được những vùng có thể xảy ra bug nhiều và phải kết hợp phân tích trong các function liên kế function mà bạn đang test.

- **Chi tiết, tỉ mỉ, cẩn thận, kiên trì:** Để test hiệu quả, nâng cao chất lượng của sản phẩm đến mức tối ưu thì một người tester phải rèn luyện cho mình khả năng cẩn thận, kiên trì, tỉ mỉ và chi tiết. Chúng ta sẽ không thể bỏ qua từng cái dấu chấm hay dấu phẩy, icon có bị méo xệch hay hàng vàng, hàng tỉ thứ nhỏ nhặt khác. Nhưng tất cả đều vô cùng quan trọng, nó hướng đến sự thân thiện với người dùng, đáp ứng nhu cầu khách hàng. Sản phẩm hoàn thiện, chất lượng cao sẽ tạo thêm niềm tin và uy tín của công ty với đối tác.

- **Có nền tảng kỹ thuật** Công việc của một tester là vô cùng quan trọng, do đó người làm kiểm thử phần mềm cũng đòi hỏi những kỹ năng chuyên môn mà không phải ai cũng có thể sở hữu hoặc trang bị trong một sớm một chiều. Các tester cần phải hiểu và thông thạo ít nhất một ngôn ngữ lập trình. Vì thế kiến thức chuyên môn về lập trình là điều đầu tiên cần có của một tester. Họ còn phải có được những kỹ năng thiết kế, lập trình, phân tích và hiểu biết về các ứng dụng khác nhau của các phần mềm vì kỹ sư kiểm định phần mềm cũng giống như bác sĩ chẩn bệnh, phải nắm vững kiến thức mới có thể chẩn đoán chính xác.

- **Kỹ năng giao tiếp:** Một tester giỏi là người luôn tỏ ra thoải mái với các cuộc xung đột. Nếu nhiều lần, nó sẽ trở nên khó khăn trong việc xác định nguồn gốc của lỗi. Là một tester bạn cần giao tiếp với nhiều người trong một công ty như trưởng nhóm, lập trình viên, BA,... Tìm ra sự thiếu sót là không đủ, bạn cần phải diễn giải tốt hơn và có thể cung cấp dữ liệu để giải quyết vấn đề. Vũ khí lớn nhất của bạn là "Giao tiếp", hãy trò chuyện mà không có sự e ngại, Email mà không có lỗi nào và trở thành một bậc thầy về giao tiếp. Giao tiếp - cả nói và viết - là một kỹ năng quan trọng giúp bạn trao đổi chặt chẽ với các developers, những người phân tích chức năng hay những bên liên quan khác trong dự án. Bạn phải đảm bảo được là hiểu rõ yêu cầu của dự án, mô tả được những tiêu chí để kiểm tra và giải thích các bước mô tả vấn đề.

- **Tiếng Anh, ngoại ngữ tốt là một ưu thế:** Đối với IT nói chung và Tester nói riêng thì ngoại ngữ là điều không thể thiếu, nó giúp bạn đọc hiểu tài liệu, update công nghệ, giao tiếp với khách hàng,... điều đó cũng có nghĩa bạn có nhiều cơ hội thăng tiến hơn. Trong kiểm thử phần mềm, giao tiếp đóng vai trò cực kỳ quan trọng. Đối với các dự án outsource thì việc giao tiếp bằng tiếng Anh gần như là không thể tránh khỏi. Nếu dự án của bạn là dự án nội bộ thì việc trau dồi tiếng Anh cũng sẽ giúp bạn tiếp cận và cập nhật kiến thức về kiểm thử phần mềm tốt hơn.

1.2.3. Cơ hội nghề nghiệp

Vị trí tester có rất nhiều cơ hội nghề nghiệp, bởi vì kiểm thử phần mềm là một lĩnh vực quan trọng và ngày càng được coi trọng trong quy trình phát triển phần mềm. Tester có thể làm các công việc như:

- **Tester phần mềm:** Tester có thể làm việc trong các công ty phát triển phần mềm, thử nghiệm sản phẩm và ứng dụng trước khi chúng được đưa ra thị trường. Vai trò này đảm bảo rằng sản phẩm đáp ứng chất lượng và yêu cầu của người dùng.
- **Tester tự động hóa:** Tester có khả năng tự động hóa kiểm thử sẽ tìm được nhiều cơ hội trong công việc. Họ sử dụng các công cụ tự động hóa để thực hiện các kịch bản kiểm thử, giúp tăng hiệu suất và chính xác.
- **Tester kiểm tra hiệu suất:** Chuyên viên kiểm tra hiệu suất tập trung vào đánh giá và cải thiện hiệu suất của phần mềm trong điều kiện khác nhau. Công việc này rất quan trọng để đảm bảo ứng dụng hoạt động mượt mà và đáp ứng được lượng người dùng lớn.
- **Tester bảo mật:** Tester bảo mật tập trung vào kiểm tra tích hợp các biện pháp bảo mật vào phần mềm. Họ đảm bảo rằng ứng dụng không có lỗ hổng bảo mật và thông tin của người dùng được bảo vệ.

Ngoài ra, tester cũng có thể thăng tiến lên các vị trí quản lý trong lĩnh vực kiểm thử phần mềm hoặc chuyển sang các vai trò khác trong quy trình phát triển phần mềm như chuyên viên kiểm thử chất lượng, chuyên viên chất lượng sản phẩm, chuyên viên dự án, hay BA. Cơ hội nghề nghiệp của tester là rộng mở và tiềm năng phát triển trong tương lai là rất tích cực.

1.2.4. Mức lương của vị trí Tester

- Mức lương của tester (người kiểm thử phần mềm) ở Việt Nam có thể thay đổi tùy theo nhiều yếu tố như vị trí, kỹ năng, kinh nghiệm làm việc, khu vực địa lý và quy mô của công ty.

- **Lương Intern Tester:** với sinh viên mới ra trường, bạn trái ngành nhảy việc muộn theo nghề tester chưa có kinh nghiệm làm việc thì mức lương sẽ tương đối thấp. Trung bình sẽ rơi vào **khoảng 3-6 triệu đồng mỗi tháng**. Đó là mức lương khởi điểm cho vị trí nhân viên thử việc, thực tập sinh, khi bạn trở thành nhân viên chính thức sẽ được điều chỉnh lương theo quyết định của quản lý.
- **Lương Fresh Tester:** Fresh Tester là những bạn sinh viên mới ra trường và vừa kết thúc thời gian thử việc. Họ đã nắm được những kiến thức cơ bản nhưng chưa

có nhiều kinh nghiệm làm việc hay chuyên môn sâu. Bởi vậy các Fresher Tester vẫn cần chỉ dạy nhiều hơn và mức lương tester mới ra trường trung bình nằm trong **khoảng 6-8 triệu đồng/tháng**.

- **Lương Junior Tester:** Junior Tester là những người đã có nhiều kiến thức, kinh nghiệm và trình độ chuyên môn tốt. Họ có thể giải quyết các test case phức tạp, có sự chủ động và sáng tạo trong công việc. Vậy nên mức lương trung bình cũng cao hơn hẳn 2 cấp bậc kể trên là **khoảng 15 triệu đồng/tháng**.
- **Lương Senior Tester:** Đối với vị trí Senior Tester hay còn có thể hiểu là những người lãnh đạo, quản lý với mức độ hiểu biết sâu rộng cùng kinh nghiệm phong phú. Bởi vậy nên họ có thể quản lý các cấp dưới của mình, điều hành và phối hợp hoạt động của nhân viên để mang về kết quả cao nhất. Mức lương của Senior Tester vì thế thường rất cao, **khoảng 22 triệu đồng/tháng**.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VỀ KIỂM THỬ PHẦN MỀM

2.1. Tổng quan về kiểm thử phần mềm

2.1.1. Giới thiệu về kiểm thử phần mềm và mục tiêu kiểm thử

- Kiểm thử phần mềm là một quá trình thực thi một chương trình hoặc ứng dụng với mục đích tìm ra các lỗi của phần mềm. Và được sử dụng để xác định tính đúng đắn, đầy đủ và chất lượng của phần mềm máy tính được phát triển.

- Kiểm thử phần mềm là cần thiết bởi vì tất cả chúng ta đều phạm sai lầm. Một số sai lầm đó không quan trọng, nhưng một số lại rất tốn kém hoặc nguy hiểm. Chúng ta cần kiểm tra mọi thứ tránh sai sót tối đa để giao sản phẩm cho khách hàng đảm bảo rằng khách hàng thấy tổ chức đáng tin cậy và sự hài lòng của họ đối với ứng dụng được duy trì.

- Mục tiêu kiểm thử:

- Đảm bảo chất lượng của sản phẩm
- Phòng ngừa và phát hiện lỗi
- Sẵn sàng tích hợp và sửa đổi thành phần
- Cung cấp thông tin để ra quyết định cho giai đoạn tiếp theo
- Công việc nên được thực hiện theo quy trình được xác định trước và SRS
- Xác minh và xác thực yêu cầu người dùng
- Thảo luận về việc tạo các trường hợp thử nghiệm
- Tạo sự tự tin trong công việc
- Tìm lỗi từ phần mềm trước khi khách hàng phát hiện ra
- Lỗi nhận được bản sửa lỗi từ nhà phát triển

2.1.2. Nguyên tắc của kiểm thử phần mềm

- **Testing shows presence of defects (Kiểm thử sẽ chỉ ra các lỗi hiện diện hay việc kiểm thử sẽ tìm ra lỗi):** Kiểm thử phần mềm có thể chỉ ra rằng có lỗi tồn tại, có sự hiện diện của lỗi trong sản phẩm phần mềm đang thực hiện kiểm thử chứ không thể chứng minh rằng sản phẩm đó là không có lỗi, sản phẩm đó hoạt động đúng. Kiểm thử phần mềm làm giảm xác suất lỗi chưa tìm thấy vẫn còn trong một sản phẩm phần mềm khi sử dụng nhưng kể cả khi

người dùng không tìm thấy lỗi nào trong quá trình dùng sản phẩm thì đó cũng không phải là bằng chứng sản phẩm đó hoạt động chính xác và không có lỗi.

- **Exhaustive testing is impossible (Thực hiện kiểm thử tất cả các trường hợp kết hợp là không thể):** Trong thực tế các dự án thường diễn ra, đối với các giai đoạn gấp rút và điều kiện không cho phép. Kiểm thử mọi thứ (tất cả các tổ hợp của điều kiện input đầu vào và điều kiện trước) là không thể thực hiện được bởi vì thời gian và chi phí sẽ bị tăng lên theo cấp số nhân. Thay vì kiểm thử toàn bộ, việc phân tích rủi ro và dựa trên sự mức độ ưu tiên chúng ta có thể tập trung việc kiểm thử vào một số điểm cần thiết.

- **Early testing (Kiểm thử sớm):** kiểm thử sớm như thiết kế các tình huống test và thực hiện các hoạt động review sớm, tìm ra các lỗi sớm thì chi phí để sửa chữa, khắc phục các lỗi đó sẽ rẻ hơn. Các hoạt động kiểm thử nên bắt đầu càng sớm càng tốt trong quy trình phát triển (vòng đời phát triển) phần mềm hoặc hệ thống và nên tập trung vào các mục tiêu đã định ra từ trước.

- **Defect clustering (Sự tập trung của lỗi):** Trong quá trình thực hiện kiểm thử, bạn sẽ thấy rằng có lỗi tập trung ở một số ít các module chức năng trong sản phẩm bạn đang kiểm thử. Thông thường đó là những chức năng chính của sản phẩm, những chức năng đòi hỏi xử lý phức tạp, hay những chức năng do những người có ít kinh nghiệm thực hiện. Chân lý này tương tự như nguyên tắc 80/20: thông thường 20% chức năng quan trọng trong một chương trình có thể gây ra đến 80% tổng số lỗi phát hiện được trong chương trình đó.

- **Pesticide paradox (Nghịch lý thuốc trừ sâu):** Bug ở đây được ví von như những chú sâu, còn những testcase (tình huống kiểm thử) phát hiện ra bug thì được coi như một loại thuốc trừ sâu. Nếu bạn dùng một loại thuốc quá lâu thì sẽ dẫn đến việc nhờn thuốc. Trong kiểm thử cũng vậy, khi việc test tương tự nhau được lặp đi lặp lại nhiều lần thì cuối cùng sẽ có một số trường hợp kiểm thử (test case) sẽ không còn tìm thấy bất kỳ lỗi mới nào. Để khắc phục "nghịch lý thuốc trừ sâu" này, các tình huống kiểm thử cần phải được xem xét và sửa đổi thường xuyên, và cần phải viết các test case mới và khác nhau để thực hiện nhiều phần khác nhau của phần mềm hoặc hệ thống để tìm ra lỗi tiềm ẩn nhiều hơn nữa.

- **Testing is context dependent (Kiểm thử theo ngữ cảnh):** Chân lý này thể hiện rằng với mỗi sản phẩm khác nhau, trong những lĩnh vực công việc khác nhau thì việc kiểm thử cũng sẽ khác nhau.

- **Absence of error fallacy (Sự sai lầm về việc không có lỗi):** Chúng ta vẫn thường nghĩ giản đơn về việc kiểm thử chính là sửa hết tất cả những lỗi sai trong sản phẩm mà chúng ta tìm ra. Nhưng việc tìm và sửa chữa lỗi sẽ không giúp được gì nếu hệ thống được xây dựng xong mà không thể dùng được và không đáp ứng được nhu cầu và sự mong đợi của người dùng. Nghĩa là nhà sản xuất sản phẩm phần mềm nếu sau khi làm đầy đủ các công đoạn để phát triển ra một sản phẩm phần mềm hoàn chỉnh (như requirement, design, code, test rồi sửa lỗi) nhưng sản phẩm đưa ra cho người dùng thật sự dùng thì phát hiện ra là sản phẩm đó không thể sử dụng hoặc không đáp ứng được nhu cầu thực sự của người sử dụng. Khi đó, dự án phần mềm coi như thất bại mặc dù các hoạt động đảm bảo chất lượng phần mềm, hoạt động kiểm thử vẫn được thực hiện đầy đủ.

2.1.3. Vòng đời phát triển phần mềm (SDLC)

- Các giai đoạn SDLC (Vòng đời phát triển phần mềm) có thể được giải thích là một loạt các bước hoặc giai đoạn mà các dự án phần mềm trải qua từ khi bắt đầu đến khi hoàn thành. Các giai đoạn này cung cấp một cách tiếp cận có cấu trúc để hướng dẫn quá trình phát triển.

- **Quy trình SDLC:**



Hình 2. 1. 1: Quy trình SDLC

1. Thu thập yêu cầu

Ở pha này bộ phận phân tích yêu cầu sẽ đi gặp và trao đổi với khách hàng cũng như làm rõ các chức năng, các yêu cầu mà khách hàng mong muốn xây dựng trong phần mềm của mình. Trong thực tế khi ở pha này các đơn vị phần mềm sẽ có bộ câu hỏi chung dành cho các dự án cũng như câu hỏi riêng cho đặc thù của dự án cần làm. Đây là pha quan trọng ảnh hưởng đến việc xây dựng và phát triển phần mềm trong thời gian tới do vậy đội ngũ phân tích yêu cầu thường là những người đã có nhiều kinh nghiệm giúp trong quá trình trao đổi với khách hàng sẽ làm rõ và hiểu đúng được các yêu cầu bài toán của khách hàng cũng như thu thập các biểu mẫu thông tin liên quan đến dự án về phục vụ phân tích ở giai đoạn kế tiếp.

2. Phân tích

Đây là pha sẽ được thực hiện sau khi đã ghi nhận các yêu cầu của khách hàng về bộ phận phân tích sẽ thực hiện làm rõ các yêu cầu và hiện thực hóa bằng một tài liệu **SRS** gọi là “Tài liệu đặc tả”. Các bộ phận liên quan như lập trình, kiểm thử viên,... sẽ thực hiện công việc dựa trên mô tả các chức năng chi tiết trong tài liệu và nó sẽ trả lời câu hỏi “Phần mềm sẽ làm gì?”.

3. Pha thiết kế

Ở pha này sau khi căn cứ vào tài liệu đặc tả, bộ phận thiết kế sẽ thiết kế đưa ra giao diện chung cũng như bộ phận lập trình sẽ thiết kế giao diện mức chi tiết theo từng chức năng của

phần mềm. Sau đó sử dụng bản khung phần mềm này để trao đổi và thống nhất với khách hàng về giao diện, bố cục. Khi khách hàng đồng ý với thiết kế phần mềm theo prototype đó sẽ mới chuyển sang giai đoạn tiếp theo nếu không sẽ ghi nhận ý kiến đóng góp và thực hiện chỉnh sửa cho đến khi khách hàng đồng thuận với bản prototype phần mềm.

4. Pha lập trình

Trong giai đoạn này SDLC bắt đầu phát triển thực tế và sản phẩm được xây dựng. Pha lập trình là pha hiện thực hóa các chức năng của phần mềm sau khi khách hàng đã thống nhất prototype của phần mềm. Ở pha này các lập trình viên (developer) sẽ lập trình xử lý chức năng, module theo yêu cầu được giao giao sau đó sẽ chuyển cho kiểm thử viên thực hiện kiểm tra các chức năng theo testcase được xây dựng dựa trên tài liệu đặc tả.

5. Pha kiểm thử

Ở pha này, các kiểm thử viên sẽ nhận được các bàn giao chức năng thực hiện từ lập trình viên. Sau đó các kiểm thử viên sẽ thực hiện kiểm tra các chức năng này theo các testcase được xây dựng. Trong quá trình kiểm thử theo testcase nếu có phát sinh lỗi, kiểm thử viên sẽ thực hiện báo bug lỗi để lập trình viên xử lý chức năng đó fix lỗi.

Quá trình kiểm thử chức năng, kiểm tra lại, báo bug lỗi, báo cáo sẽ thực hiện đi thực hiện lại cho đến khi các chức năng lập trình đã thực hiện đúng theo tài liệu đặc tả hay yêu cầu của khách hàng.

Sau khi hoàn thiện các chức năng cũng như đạt yêu cầu về kiểm thử, phần mềm sẽ được triển khai thử nghiệm trên môi trường của khách hàng. Nếu có yêu cầu chỉnh sửa đội ngũ phát triển phần mềm sẽ thực hiện bổ sung, sửa lỗi để có thể nghiệm thu phần mềm.

6. Pha triển khai, bảo trì

Một khi sản phẩm sau khi được kiểm thử và sẵn sàng triển khai, sản phẩm được phát hành chính thức trên thị trường thích hợp. Trong quá trình sử dụng phần mềm của khách hàng, công ty phát triển phần mềm sẽ phải hỗ trợ, xử lý lỗi nếu có phát sinh trong quá trình sử dụng.

2.2. Loại kiểm thử phần mềm

2.2.1. Manual testing (Kiểm thử thủ công)

Manual testing (Kiểm thử thủ công) là quá trình thực hiện kiểm tra các tính năng, chức năng và yêu cầu của phần mềm một cách thủ công, không sử dụng các công cụ tự động hóa. Manual testing đóng vai trò quan trọng trong việc xác định chất lượng của phần mềm trước khi phát hành sản phẩm cho người dùng cuối. Manual testing đòi hỏi sự chú ý đáng kể từ các kiểm thử viên và có thể tốn nhiều thời gian. Tuy nhiên, nó vẫn là một phần quan trọng trong quá trình phát triển phần mềm, bởi vì nó cho phép các kiểm thử viên tập trung vào việc hiểu rõ nhu cầu của người dùng cuối cùng và khám phá các lỗi không rõ ràng mà công cụ tự động hóa có thể bỏ sót.

2.2.2. Automation testing (Kiểm thử tự động)

Automation testing (Kiểm thử tự động) là một phương pháp thực hiện kiểm thử phần mềm bằng sự hỗ trợ của các công cụ và kịch bản thực thi tự động thay vì thực hiện kiểm thử thủ công. Điều này giúp tăng hiệu quả, độ tin cậy và tiết kiệm thời gian trong quá trình kiểm thử. Kiểm thử tự động là một yếu tố quan trọng trong quá trình phát triển phần mềm hiện đại, giúp tăng cường chất lượng sản phẩm, giảm chi phí và thời gian thực hiện kiểm thử. Tuy nhiên, nó cần được triển khai một cách khôn ngoan và kết hợp với kiểm thử thủ công để đảm bảo tính hiệu quả và đáng tin cậy trong việc kiểm thử phần mềm.

2.2.3. Security testing (Kiểm thử bảo mật)

- **Security testing (Kiểm thử bảo mật)** là quá trình thực hiện kiểm thử để đảm bảo rằng các tính năng và chức năng của phần mềm được bảo mật và an toàn trước những mối đe dọa tiềm ẩn từ bên ngoài.

- **Mục tiêu chính của security testing** là xác định các lỗ hổng bảo mật, các thiếu sót về bảo mật và khả năng kiểm soát truy cập trái phép vào hệ thống. Đảm bảo rằng phần mềm đáp ứng các yêu cầu bảo mật và bảo vệ khỏi các cuộc tấn công từ bên ngoài. Nó đảm bảo rằng các dữ liệu và thông tin quan trọng của người dùng cuối và doanh nghiệp được bảo mật và an toàn trong môi trường kỹ thuật số ngày càng phức tạp và đầy thách thức.

- **Các hình thức của Security Testing:**

Penetration Testing: Còn được gọi là Ethical Hacking, là quá trình thử thách hệ thống bằng cách giả lập các cuộc tấn công từ bên ngoài để tìm ra các điểm yếu và lỗ hổng bảo mật. Mục tiêu là tìm ra các lỗ hổng có thể tấn công được từ kẻ tấn công thực sự.

Vulnerability Scanning: Sử dụng các công cụ tự động để quét và phát hiện các lỗ hổng bảo mật phổ biến trong hệ thống, ví dụ như các cổng mạng mở, các thiếu sót bảo mật trong phần mềm hay cấu hình hệ thống không an toàn.

Security Code Review: Kiểm tra mã nguồn của phần mềm để phát hiện các lỗ hổng bảo mật như các điểm dừng tiềm tàng, sử dụng bất hợp lý của hàm API và các vấn đề bảo mật khác.

2.2.4. API testing (Kiểm thử API)

- **API testing (Application Programming Interface testing)** là quá trình kiểm thử các giao diện lập trình ứng dụng của phần mềm, có nhiệm vụ đảm bảo tính đúng đắn, đáng tin cậy và hiệu quả của các API. API là các giao diện mà phần mềm sử dụng để giao tiếp với các thành phần khác, cho phép trao đổi dữ liệu và thực hiện các chức năng.

- **Manual Testing:** Kiểm thử thủ công của các API bằng cách sử dụng các công cụ như Postman hoặc cURL để gửi các yêu cầu và xác nhận kết quả trả về.

- **Automated Testing:** Sử dụng các công cụ tự động hóa để thực hiện kiểm thử tự động của các API, giúp tiết kiệm thời gian và tăng tính chính xác.

2.3. Phương pháp kiểm thử

2.3.1. Static testing (Kiểm thử tĩnh)

- **Static testing (Kiểm thử tĩnh)** là các kiểm thử được thực hiện mà không cần phải chạy hoặc thực thi các đối tượng kiểm thử - test object.

- Static testing có thể được thực hiện bằng cách review và static analysis:

- Review là hoạt động kiểm tra sản phẩm được thực hiện bởi con người. Người kiểm tra sẽ đọc kỹ sản phẩm cần kiểm tra (ví dụ tài liệu, source code,...) để tìm ra lỗi.

- Static analysis là hoạt động kiểm tra sản phẩm tĩnh (như tài liệu, source code) bằng một công cụ - tool (ví dụ: để kiểm tra xem có lỗi về từ gõ sai – spelling chẳng hạn thì điều chỉnh thiết lập để Microsoft Word có thể hỗ trợ, hoặc một số công cụ hỗ trợ việc review source code như tìm ra các câu lệnh, các biến mà không được sử dụng, hay đoạn code đó không được lập trình theo các quy định, quy ước về lập trình – coding convention có sẵn).

Ví dụ: hoạt động review các tài liệu, source code sẽ là các hoạt động kiểm thử tĩnh, sản phẩm cần kiểm thử chỉ là các trang tài liệu tĩnh, người thực hiện việc kiểm tra không cần chạy sản phẩm đó.

2.3.2. Dynamic testing (Kiểm thử động)

Kiểm thử động - Dynamic testing ngược với kiểm thử tĩnh, nghĩa là kiểm thử mà cần phải chạy hoặc thực thi các đối tượng kiểm thử - test object.

Ví dụ: khi một nhân viên kiểm thử – tester thực hiện kiểm thử cho một chức năng nào đó của sản phẩm hoặc hệ thống, họ sẽ phải chạy chức năng đó và giao tiếp với sản phẩm cần kiểm thử qua giao diện, đó chính là kiểm thử động.

2.3.3. White box testing (Kiểm thử hộp trắng)

- **White box testing (Kiểm thử hộp trắng)** là một kỹ thuật kiểm thử trong đó cấu trúc bên trong, thiết kế và mã hóa của phần mềm được kiểm tra để xác minh luồng đầu vào đầu ra và cải thiện thiết kế, khả năng sử dụng và bảo mật. Trong thử nghiệm hộp trắng, mã được hiển thị cho người thử nghiệm, vì vậy nó còn được gọi là Thử nghiệm hộp rõ ràng, Thử nghiệm hộp mở, Thử nghiệm hộp trong suốt, Thử nghiệm dựa trên mã và Thử nghiệm hộp thủy tinh.

- **Kỹ thuật kiểm thử hộp trắng:**

- **Statement Testing (Kiểm thử tất cả câu lệnh):** Statement testing tập trung vào kiểm tra từng câu lệnh trong mã nguồn để đảm bảo rằng mỗi câu lệnh được thực thi một cách đúng đắn và không bị bỏ sót. Mục tiêu là đạt đến 100% vị phạm câu lệnh trong quá trình kiểm thử.

- **Decision Testing (Kiểm thử tất cả các quyết định, các rẽ nhánh):** Decision testing tập trung vào kiểm tra tất cả các quyết định và các rẽ nhánh trong mã nguồn để đảm bảo rằng mọi quyết định đều được xử lý chính xác và các rẽ nhánh điều kiện đều được thực hiện đúng. Mục tiêu của decision testing là đạt đến 100% vị phạm của các điều kiện và các quyết định trong quá trình kiểm thử.

- **Condition Determination Testing (Kiểm thử các điều kiện):** Condition determination testing tập trung vào kiểm tra hiệu quả của các điều kiện riêng lẻ trong quyết định tổng thể. Mục tiêu là đảm bảo rằng mỗi điều kiện đơn trong các quyết định hoạt động đúng đắn và không bị ảnh hưởng bởi các điều kiện khác.

- **Path Testing (Kiểm thử các đường đi):** Path testing tập trung vào kiểm tra tất cả các đường đi trong mã nguồn để đảm bảo rằng mỗi đường đi trong hàm hay thủ tục đều được thực hiện đúng đắn. Mục tiêu là đạt đến 100% vị phạm của các đường đi trong quá trình kiểm thử.

2.3.4. Black box testing (Kiểm thử hộp đen)

- **Black box testing (Kiểm thử hộp đen)** là một phương pháp kiểm thử phần mềm trong đó các chức năng của ứng dụng phần mềm được kiểm tra mà không cần biết về cấu trúc mã nội bộ, chi tiết triển khai và đường dẫn nội bộ. Kiểm thử hộp đen chủ yếu tập trung vào đầu vào và đầu ra của các ứng dụng phần mềm và nó hoàn toàn dựa trên các yêu cầu và thông số kỹ thuật của phần mềm.

- **Kỹ thuật kiểm thử hộp đen:**

- **Kiểm thử lớp tương đương:** là một kỹ thuật thiết kế các tình huống kiểm thử bằng cách phân chia hệ thống thành nhiều nhóm, nhiều lớp mà trong mỗi nhóm/lớp đó hệ thống sẽ được kỳ vọng là được kiểm soát theo cùng một cách và có cùng xử lý giống nhau. Các nhóm/lớp này được gọi là các nhóm/lớp ngang bằng hay tương đương.

- **Kiểm tra giá trị ranh giới:** Kiểm tra giá trị ranh giới tập trung vào các giá trị tại các ranh giới. Kỹ thuật này xác định xem một phạm vi giá trị nhất định có được hệ thống chấp nhận hay không. Nó rất hữu ích trong việc giảm số lượng các

trường hợp thử nghiệm. Nó phù hợp nhất cho các hệ thống có đầu vào nằm trong phạm vi nhất định.

- **Kiểm tra bảng quyết định :** Bảng quyết định đặt các nguyên nhân và tác động của chúng trong một ma trận. Các tình huống kiểm thử được thiết kế dựa trên việc kết hợp các đầu vào và/hoặc các nguyên nhân được thể hiện ở trong bảng quyết định.

2.4. Cấp độ của kiểm thử

2.4.1. Unit testing (Kiểm thử đơn vị)

- **Unit testing** là phần nhỏ nhất có thể kiểm tra của hệ thống hoặc ứng dụng có thể được biên dịch, thích, tải và thực thi. Loại thử nghiệm này giúp kiểm tra từng mô-đun riêng biệt.

- Mục đích là để kiểm tra từng phần của phần mềm bằng cách tách nó ra. Nó kiểm tra xem thành phần đó có hoàn thành các chức năng hay không. Loại thử nghiệm này được thực hiện bởi các nhà phát triển.

2.4.2. Integration testing (Kiểm thử tích hợp)

- Kiểm tra xem các thành phần của hệ thống có được tích hợp theo đúng như thiết kế tổng thể, thiết kế kiến trúc hệ thống (architecture design) hay không.

Ví dụ: Trong giai đoạn thử nghiệm này, các mô-đun phần mềm khác nhau được kết hợp và thử nghiệm theo nhóm để đảm bảo rằng hệ thống tích hợp sẵn sàng cho thử nghiệm hệ thống.

- Kiểm thử tích hợp kiểm tra luồng dữ liệu từ mô-đun này sang mô-đun khác. Loại thử nghiệm này được thực hiện bởi những người thử nghiệm.

2.4.3. System testing (Kiểm thử hệ thống)

- **System testing** được thực hiện trên một hệ thống hoàn chỉnh, tích hợp. Nó cho phép kiểm tra sự tuân thủ của hệ thống theo yêu cầu. Nó kiểm tra sự tương tác tổng thể của các thành phần. Nó liên quan đến kiểm tra tải, hiệu suất, độ tin cậy và bảo mật.

- Thử nghiệm hệ thống thường là thử nghiệm cuối cùng để xác minh rằng hệ thống đáp ứng các đặc điểm kỹ thuật. Nó đánh giá cả nhu cầu chức năng và phi chức năng cho thử nghiệm.

2.4.4. Acceptance Testing (Kiểm thử người dùng)

- **Acceptance Testing** là một thử nghiệm được tiến hành để tìm hiểu xem các yêu cầu của một đặc điểm kỹ thuật hoặc hợp đồng có được đáp ứng khi giao hàng hay không.

- Kiểm tra chấp nhận về cơ bản được thực hiện bởi người dùng hoặc khách hàng.

2.5. Tổng quan về Python

2.5.1. Python là gì?

- **Python** là một ngôn ngữ lập trình cao cấp, mạnh mẽ, đa năng và dễ học. Nó được tạo ra bởi Guido van Rossum và được công bố lần đầu tiên vào năm 1991. Python là một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới và được sử dụng trong nhiều lĩnh vực, bao gồm phát triển phần mềm, trí tuệ nhân tạo (AI), khoa học dữ liệu, web development, và nhiều ứng dụng khác.

- Một trong những điểm mạnh của Python là cú pháp đơn giản và dễ đọc, giúp cho việc viết mã trở nên rõ ràng và dễ hiểu hơn so với nhiều ngôn ngữ khác. Python hỗ trợ nhiều phong cách lập trình, bao gồm lập trình hướng đối tượng, lập trình hàm, và lập trình cấu trúc.

- Python có một cộng đồng lớn và đầy đủ tài liệu hỗ trợ, bao gồm nhiều thư viện và module tiện ích đã được viết sẵn, giúp nhà phát triển tiết kiệm thời gian và công sức trong việc xây dựng ứng dụng và giải quyết các vấn đề phức tạp. Điều này đã làm cho Python trở thành một lựa chọn phổ biến cho các dự án lớn và nhỏ trên toàn thế giới.

2.5.2. Ứng dụng của Python

Python có rất nhiều ứng dụng khác nhau và được sử dụng trong nhiều lĩnh vực như:

- **Phát triển phần mềm:** Python được sử dụng rộng rãi để phát triển các ứng dụng và phần mềm đa dạng, bao gồm ứng dụng desktop, ứng dụng di động, ứng dụng web, và hơn thế nữa.

- **Trí tuệ nhân tạo và học máy:** Python được ưa chuộng trong lĩnh vực trí tuệ nhân tạo (AI) và học máy (machine learning). Thư viện như TensorFlow, PyTorch và scikit-learn cung cấp các công cụ mạnh mẽ cho việc xây dựng và huấn luyện mô hình máy học.
- **Khoa học dữ liệu và phân tích:** Python có nhiều thư viện mạnh mẽ như Pandas, NumPy và Matplotlib, giúp cho việc xử lý và phân tích dữ liệu trở nên dễ dàng hơn.
- **Web development:** Python được sử dụng để xây dựng các ứng dụng web và trang web thông qua các framework như Django và Flask.
- **Game development:** Python có thư viện Pygame cho phép nhà phát triển tạo các trò chơi đơn giản và 2D.

2.6. Tổng quan về Robot Framework

2.6.1. Robot Framework là gì ?

- **Robot Framework** là một công cụ kiểm thử tự động mã nguồn mở được sử dụng để tự động hóa kiểm thử phần mềm. Nó cung cấp một cách dễ dàng để viết các bài kiểm thử sử dụng một cú pháp đơn giản và dễ hiểu.

- Robot Framework được viết bằng Python và hỗ trợ các kiểu kiểm thử như kiểm thử chức năng, kiểm thử giao diện người dùng, kiểm thử dịch vụ web, và kiểm thử ứng dụng di động.

- **Một số đặc điểm chính của Robot Framework bao gồm:**

- **Cú pháp đơn giản:** Robot Framework sử dụng cú pháp thân thiện, dễ đọc và dễ viết, dựa trên cấu trúc từ khóa (keyword-driven). Điều này giúp cho người dùng dễ dàng tạo và quản lý các bài kiểm thử.
- **Hỗ trợ đa ngôn ngữ:** Robot Framework hỗ trợ nhiều ngôn ngữ lập trình, bao gồm Python và Java, để người dùng có thể sử dụng các thư viện và module từ nhiều nguồn.
- **Dễ mở rộng:** Người dùng có thể viết các thư viện tùy chỉnh và tích hợp chúng vào Robot Framework để mở rộng khả năng kiểm thử.

- **Tích hợp liên kết:** Robot Framework tích hợp được với nhiều công cụ kiểm thử và báo cáo, giúp người dùng dễ dàng tích hợp vào quy trình kiểm thử tổng thể.
- **Hỗ trợ các loại kiểm thử:** Ngoài kiểm thử chức năng, Robot Framework còn hỗ trợ kiểm thử giao diện người dùng, kiểm thử dịch vụ web và kiểm thử ứng dụng di động.

2.6.2. Ưu điểm và nhược điểm của Robot Framework

- Ưu điểm:

- **Dễ dàng sử dụng:** Robot Framework có cú pháp rõ ràng và dễ đọc, giúp người dùng dễ dàng viết và duyệt các bài kiểm thử mà không cần có kiến thức sâu về lập trình.
- **Hỗ trợ đa nền tảng:** Robot Framework hỗ trợ các hệ điều hành khác nhau, giúp người dùng thực hiện kiểm thử trên nhiều môi trường khác nhau.
- **Dễ mở rộng:** Robot Framework cho phép người dùng tạo các thư viện tùy chỉnh và tích hợp chúng vào quy trình kiểm thử, giúp mở rộng khả năng kiểm thử của nó.
- **Hỗ trợ nhiều loại kiểm thử:** Robot Framework không chỉ hỗ trợ kiểm thử chức năng, mà còn kiểm thử giao diện người dùng, kiểm thử dịch vụ web và kiểm thử ứng dụng di động.

- Nhược điểm:

- **Hạn chế cho các kiểm thử phức tạp:** Mặc dù Robot Framework phù hợp với các kiểm thử đơn giản và trung bình, nhưng nó có thể gặp khó khăn trong việc thực hiện các kiểm thử phức tạp và lớn.
- **Yêu cầu kiến thức cơ bản về lập trình:** Dù cú pháp của Robot Framework dễ đọc, nhưng vẫn cần một số kiến thức cơ bản về lập trình và tư duy logic để viết các tập lệnh kiểm thử hiệu quả.
- **Hiệu suất tương đối chậm:** So với một số công cụ kiểm thử tự động khác, Robot Framework có thể chạy chậm hơn trong việc thực thi các bài kiểm thử.

CHƯƠNG 3: TRIỂN KHAI DỰ ÁN “AUTOMATION TESTING FOR THE SHOPPING WEBSITE WITH ROBOT FRAMEWORK”

3.1. Tổng quan hệ thống “Shopping Website”

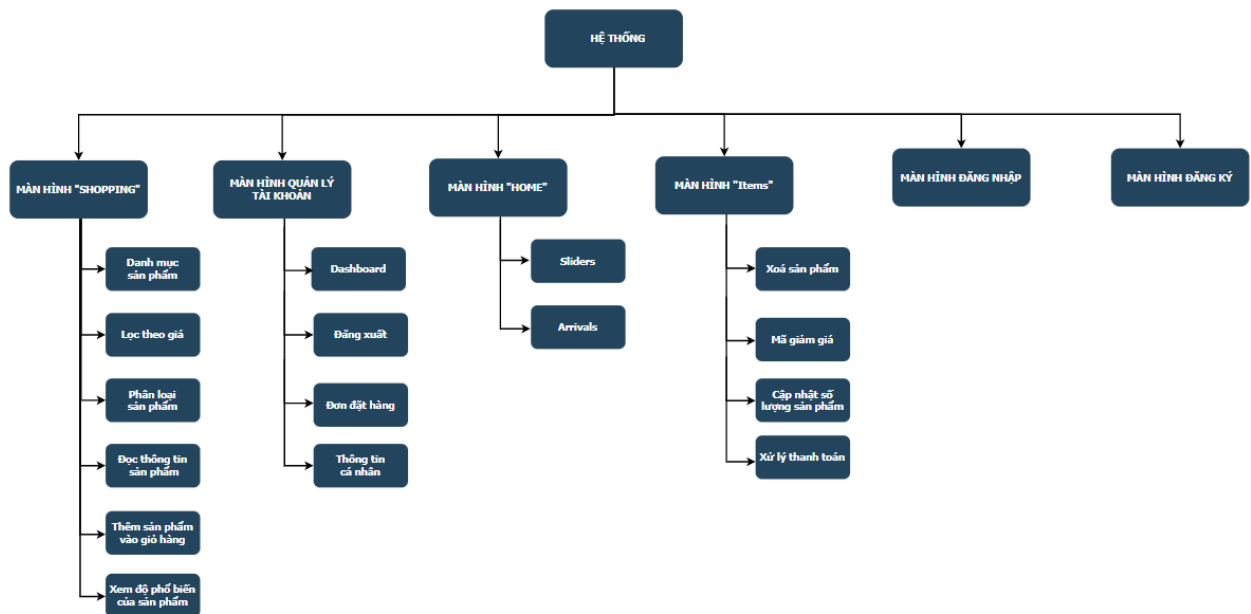
- Website “Shopping” là trang web được thiết kế với môi trường mua sắm các sản phẩm về sách lập trình. Người mua có thể thêm sách bản thân yêu thích vào giỏ hàng và tự động giúp người mua tự động tính tiền. Trang web sẽ phân theo các loại sách, người dùng có thể vào các thể loại mà bản thân muốn để dễ dàng lựa chọn, có thể xem bình luận về sách cũng như thông tin chi tiết của sách. Ngoài ra, trang web cho phép người dùng sắp xếp mức độ phổ biến của sách và lựa chọn khoảng giá theo mong muốn mang đến trải nghiệm tốt nhất cho người dùng.

- Đối tượng sử dụng: Tất cả khách hàng có nhu cầu sử dụng sách

- Khách hàng có quyền:

- Tạo tài khoản cá nhân
- Thêm sách cần mua vào giỏ hàng
- Sửa hoặc xóa sách đã được thêm vào giỏ hàng
- Sắp xếp các mức độ phổ biến của sách
- Giới hạn giá của sách hiện đang được bán
- Viết bình luận để đánh giá về sách

- Mô hình tổng quát của hệ thống



Hình 3. 1. 1: Mô hình hệ thống

3.2. Đặc tả yêu cầu

3.2.1. Chức năng tại màn hình “Đăng ký (Register)”

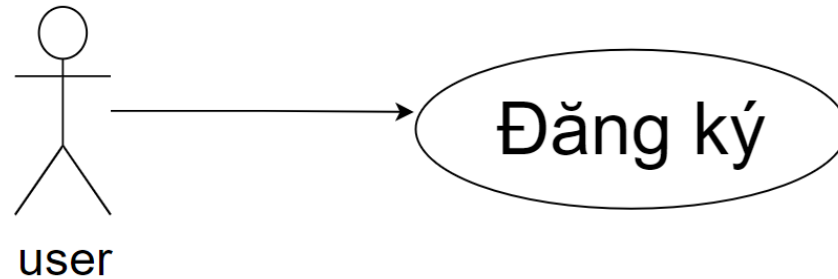
3.2.1.1. Đặc tả màn hình “Đăng ký (Register)”

- Khi khách hàng truy cập vào web, hệ thống sẽ hiển thị như Hình 3.2.1, từ màn hình khách hàng điền thông tin địa chỉ email, mật khẩu vào ô đăng ký phía bên phải màn hình.

- Sau khi đăng ký thành công hệ thống sẽ chuyển sang màn hình My account.

Hình 3. 2. 1: Màn hình đăng nhập đăng ký

3.2.1.2. Usecase “Đăng ký (Register)”



3.2.1.3. Đặc tả usecase “Đăng ký (Register)”

- **Tác nhân:** User
- **Điều kiện kích hoạt:** Người dùng truy cập vào hệ thống
- **Mô tả chi tiết usecase:**
 - Bước 1: Hệ thống hiển thị màn hình đăng ký
 - Bước 2: Người dùng nhập email và password cần tạo
 - Bước 3: Click vào nút “**Register**” hệ thống chuyển sang trang chủ

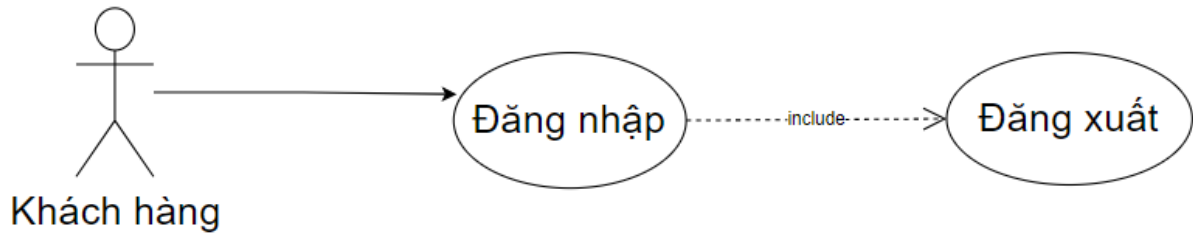
3.2.2. Chức năng tại màn hình “Đăng nhập (Login)”

3.2.2.1 Đặc tả màn hình “Đăng nhập (Login)”

- Khi khách hàng truy cập vào web, hệ thống sẽ hiển thị như *Hình 3.2.1*, từ màn hình khách hàng điền thông tin địa chỉ email, mật khẩu đã được đăng ký vào ô đăng nhập phía bên trái màn hình.

- Sau khi đăng nhập thành công hệ thống sẽ chuyển sang màn hình My account.

3.2.2.2. Usecase “Đăng nhập (Login)”



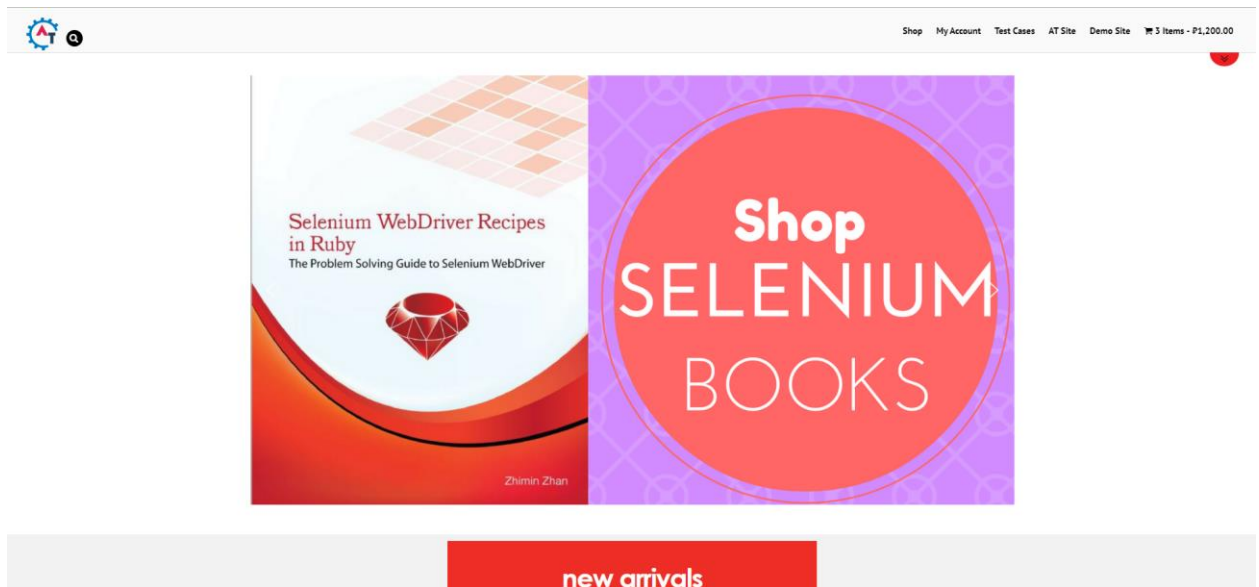
3.2.2.3. Đặc tả usecase “Đăng nhập (Login)”

- **Tác nhân:** User
- **Điều kiện kích hoạt:** Người dùng truy cập vào hệ thống
- **Mô tả chi tiết Usecase:**
 - **Bước 1:** Hệ thống hiển thị màn hình đăng nhập
 - **Bước 2:** Người dùng nhập email và password đã được đăng ký
 - **Bước 3:** Click vào nút “**Login**” hệ thống chuyển sang trang My account

3.2.3. Chức năng tại màn hình “Home”

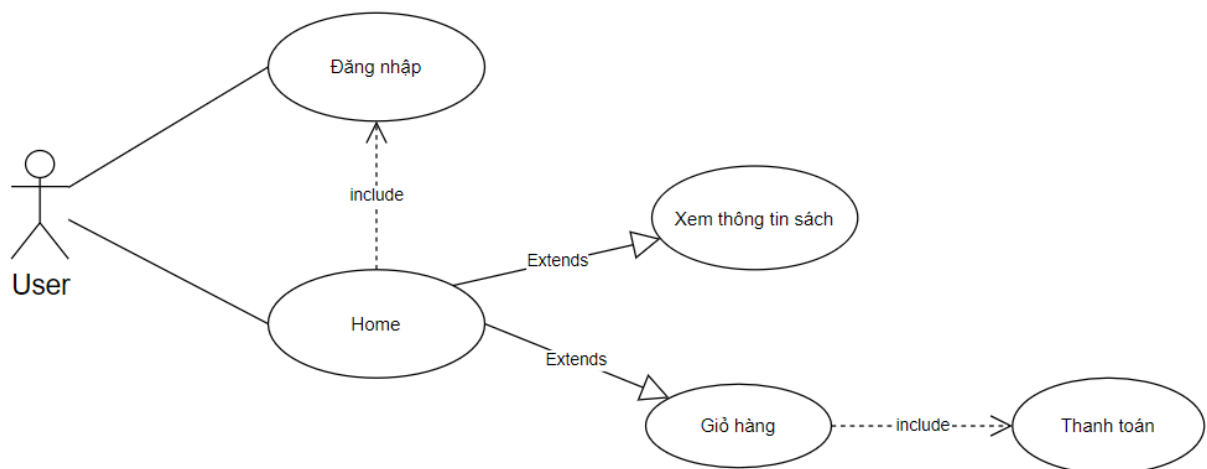
3.2.3.1. Đặc tả màn hình “Home”

- Sau khi đăng nhập thành công, người dùng click vào “Home” để xem sách như *Hình 3.2.2*, sau đó click vào từng cuốn sách để xem thông tin sách, giá, bình luận về sách. Sau đó, người dùng có thể thêm sách vào giỏ hàng, click vào link điều hướng đến giỏ hàng để xem lại giá, cập nhật số lượng sách cần mua, cũng như áp dụng phiếu giảm giá cho sách và đặt mua sách nếu muốn.



Hình 3. 2. 2: Màn hình Home

3.2.3.2. Usecase màn hình “Home”



3.2.3.3. Đặc tả Usecase “Home”




- **Tác nhân:** User
- **Điều kiện kích hoạt:** Người dùng đăng nhập vào hệ thống
- **Mô tả chi tiết Usecase:**
 - **Bước 1:** Hệ thống hiển thị màn hình đăng nhập
 - **Bước 2:** Người dùng nhập email và password đã được đăng ký

- **Bước 3:** Click vào nút **“Login”** hệ thống chuyển sang trang My account
- **Bước 4:** Người dùng click vào **“Home”** để xem sản phẩm
- **Bước 5:** Click vào loại sách ưa thích để xem thông tin chi tiết cuốn sách, cũng như bình luận của sách
 - **Bước 6:** Nếu muốn cho sách vào giỏ hàng thì người dùng click **“Add to basket”** hoặc thêm sách vào giỏ hàng trực tiếp tại màn hình **“Home”**. Sách sẽ được thêm thành công nếu trong kho có đủ số lượng
 - **Bước 7:** Nếu muốn xem lại giỏ hàng khách hàng sẽ vào menu **“Items”** để xem giỏ hàng, chỉnh sửa thông tin như số lượng sách, xoá các cuốn sách không có nhu cầu mua hay thêm mã giảm giá.
 - **Bước 8:** Click **“Update basket”** để cập nhật lại thông tin sản phẩm
 - **Bước 9:** Nếu muốn mua sách người dùng click **“Proceed to checkout”** và kiểm tra lại thông tin cá nhân địa chỉ, số lượng của sách, lựa chọn hình thức thanh toán và click **“Place order”** để đặt hàng.

3.2.4. Chức năng tại màn hình “Giỏ hàng (Items)”

3.2.4.1. Đặc tả chức năng tại màn hình “Giỏ hàng (Items)”

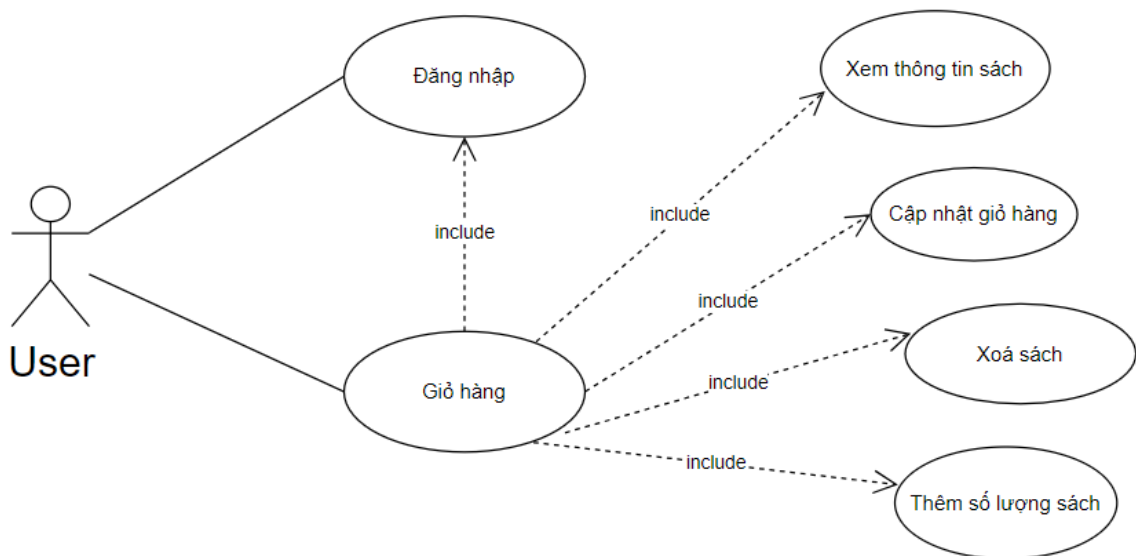
- Sau khi đăng nhập thành công, người dùng thêm sách mình thích vào giỏ hàng tại trang “Shop” hoặc “Home”, sau đó chuyển sang màn hình giỏ hàng tại menu “Items” như *Hình 3.2.3* để điều chỉnh số lượng mua, thêm hoặc xoá các loại sách theo nhu cầu và sau đó tiến hành thanh toán.

				Shop	My Account	Test Cases	AT Site	Demo Site	2 Items - ₹700.00
Product	Price	Quantity	Total						
 Android Quick Start Guide	₹450.00	1	₹450.00						
 Functional Programming in JS	₹250.00	1	₹250.00						
Coupon cod: <input type="text"/> <input type="button" value="APPLY COUPON"/>				<input type="button" value="UPDATE BASKET"/>					

Basket Totals	
Subtotal	₹700.00
Tax	₹14.00
Total	₹714.00

Hình 3. 2. 3: Màn hình giỏ hàng

3.2.4.2. Usecase màn hình “Giỏ hàng (Items)”



3.2.4.3. Đặc tả Usecase “Giỏ hàng”

- Tác nhân: User
- Điều kiện kích hoạt: Người dùng đăng nhập vào hệ thống
- Mô tả chi tiết Usecase:
 - **Bước 1:** Hệ thống hiển thị màn hình đăng nhập

- **Bước 2:** Người dùng nhập email và password đã được đăng ký
- **Bước 3:** Click vào nút “**Login**” hệ thống chuyển sang trang My account
- **Bước 4:** Người dùng click vào “Shop” để xem sản phẩm
- **Bước 5:** Click “**ADD TO BASKET**” để thêm sản phẩm vào giỏ hàng
- **Bước 6:** Click “Items” để chuyển sang màn hình thông tin giỏ hàng
- **Bước 7:** Khách hàng loại bỏ sản phẩm không mong muốn hoặc cập nhật lại số lượng
- **Bước 8:** Click “**UPDATE BASKET**” để cập nhật lại thông tin sản phẩm

3.3. Thiết kế Testcase

- Viết testcase cho 4 chức năng:

- Đăng ký (Register)
- Đăng nhập (Login)
- Trang chủ (Home)
- Giỏ hàng (Items)

- Link testcase: https://docs.google.com/spreadsheets/d/1wDp60FWGox4fW-nQzOpP1s71s_nSdVXp/edit?usp=sharing&ouid=112963174031465478907&rtpof=true&sd=true

3.3.1. Testcase chức năng tại màn hình “Đăng ký (Register)”

No.	Test name	Pre condition	Test step	Data	Expected Result	Verify
1	Check account registration is successful	Successful access to the website	1) Open the website 2) Click on My Account Menu 3) Enter registered Email Address	Email address: tranx2222@gmail.com Password: Thien123@Thien123	User successfully registered The website will switch to the “Home page”	Pass

			in Email-Address textbox 4) Enter your own password in password textbox 5) Click on Register button			
2	Check registration when email is invalid	Successful access to the website	1) Open the website 2) Click on My Account Menu 3) Enter invalid Email Address in Email-Address textbox 4) Enter your own password in password textbox 5) Click on Register button	Email address: tranx2222g mail.com Password: Thien123@ Thien123	Registration failed The screen showed the error "Please include '@' in email address. 'tranx2222gm ail.com' is missing '@'."	Pass
3	Check registration when email address is empty	Successful access to the website	1) Open the website 2) Click on My Account Menu 3) Enter empty	Email address: Empty	Registration failed The screen showed the error "Error:	Pass

			<p>Email Address in Email-Address textbox</p> <p>4) Enter your own password in password textbox</p> <p>5) Click on Register button</p>	<p>Password: Thien123@Thien123</p>	<p>Please provide a valid email address.”</p>	
4	<p>Check registration when password is empty</p>	<p>Successful access to the website</p>	<p>1) Open the browser</p> <p>2) Enter the URL “http://practice.automationtesting.in/”</p> <p>3) Click on My Account Menu</p> <p>4) Enter valid Email Address in Email-Address textbox</p> <p>5) Enter empty password in password textbox</p>	<p>Email address: tranx2222@gmail.com</p> <p>Password: Empty</p>	<p>Registration failed</p> <p>The screen showed the error “Error: Please enter an account password.”</p>	<p>Pass</p>

			6) Click on Register button			
5	Check registration when email address and password are blank	Successful access to the website	1) Open the website 2) Click on My Account Menu 3) Enter empty Email Address in Email-Address textbox 4) Enter empty password in password textbox 5) Click on Register button	Email address: Empty Password: Empty	Registration failed The screen showed the error “Error: Please provide a valid email address.”	Pass

Bảng 1: Testcase màn hình “Register”

3.3.2. Testcase chức năng tại màn hình “Đăng nhập (Login)”

No.	Test name	Pre condition	Test step	Data	Expected Result	Verify
1	Check login successful	Successful access to the website	1) Open the website 2) Click on My Account Menu 3) Enter registered username in username textbox	Username or email address: tranx2222@gmail.com	User successfully registered	Pass

		Account is registered	4) Enter password in password textbox 5) Click on login button	Password: Thien123@Thien123	The website will switch to the “My account”	
2	Check login when username and password are incorrect	Successful access to the website Account is registered	1) Open the website 2) Click on My Account Menu 3) Enter incorrect username in username textbox 4) Enter incorrect password in password textbox. 5) Click on login button	Username or email address: tranien2@gmail.com Password: ThienThi123thien222	Login failed The screen showed the error “Error: A user could not be found with this email address.”	Pass
3	Check login when	Successful access	1) Open the website 2) Click on My Account Menu	Username or email address: tranx2222@gmail.com	Login failed The screen	Pass

	password is blank	to the website Account is registered	3) Enter valid username in username textbox 4) Now enter empty password in the password textbox 5) Click on login button.	Password: Empty	showed the error “Error: Password is required.”	
4	Check login when username is blank	Successful access to the website Account is registered	1) Open the website 2) Click on My Account Menu 3) Enter empty username in username textbox 4) Now enter valid password in the password textbox 5) Click on login button.	Username or email address: Empty Password: Thien@123Thien123	Login failed The screen showed the error “Error: Username is required.”	Pass
5	Check login when username and password	Successful access to the website Account is registered	1) Open the website 2) Click on My Account Menu 3) Enter empty username in username textbox	Username or email address: Empty Password: Empty	Login failed The screen showed the error	Pass



	are empty		4) Now enter valid password in the password textbox 5) Click on login button.		“Error: Username is required.”	
6	Check password masking when logging in	Successful access to the website Account is registered	1) Open the website 3) Click My Account Menu 4) Enter the password field with some characters.	Username or email address: tranx2222@gmail.com Password: Thien123@Thien123	The password field only shows the characters in asterisks and does not show the password on the screen	Pass
7	Check password case sensitive function	Successful access to the website	1) Open website 3) Click My Account Menu 4) Enter username field	Username or email address: tranx2222@gmail.com	Login failed The screen showed the error	Pass

		Account is registered	5) Password field in all caps 6) Click the "login" button	Password: THIEN@123T HIEN123	“Error: The password you entered for the username tranxthien2222@gmail.com is incorrect. Lost your password?”	
--	--	-----------------------	--	------------------------------------	--	--

Bảng 2: Testcase màn hình “Login”

3.3.3. Testcase chức năng tại màn hình “Home”

No.	Test name	Pre condition	Test step	Data	Expected Result	Verify
1	Check login successful	Website is logged in	1) Open the website 2) Click on Shop Menu 3) Now click on Home menu button		The Home page must contains	Pass

			<p>4) Click on the arrow as shown</p> 		only three sliders	
2	Home page with three Arrivals only	Website is logged in	<p>1) Open the website</p> <p>2) Click on Shop Menu</p> <p>3) Now click on Home menu button</p> <p>4) Watch the screen</p>		<p>The Home page must contains only three Arrivals as shown below</p> 	Pass
3	Check image in arrivals navigate to page where user can add products to basket	Website is logged in	<p>1) Open the website</p> <p>2) Click on Shop Menu</p> <p>3) Now click on Home menu button</p> <p>4) Test whether the Home page has Three Arrivals only</p> <p>5) The Home page must contains only three Arrivals</p>		<p>The image should be clickable and navigate to the next page where</p>	Pass

			6) Now randomize the image in Arrivals		the user can add the book to the basket	
4	Check arrivals navigate to cart and show description of book	Website is logged in	1) Open website 2) Click Shop Menu 3) Now click on the Home menu button 4) Check if Homepage has only Three arrivals 5) The homepage can only contain three Destinations 6) Now random click on image in Arrivals and will navigate to next page where user can add that book to basket 7) Click on the Description tab		Displays a description of the book the user clicked on	Pass
5	Check arrivals navigate to cart	Website is logged in	1) Open website 2) Click Shop Menu		Show reviews of books the user	Pass

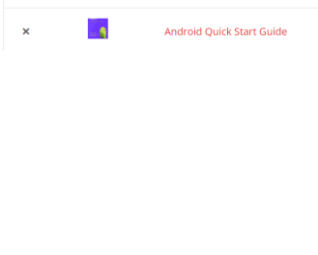
	and show review of book		<p>3) Now click on the Home menu button</p> <p>4) Check if Homepage has only Three arrivals</p> <p>5) The homepage can only contain three Destinations</p> <p>6) Now random click on image in Arrivals and will navigate to next page where user can add that book to cart</p> <p>7) Click on the review tab</p>		clicked on	
6	Check arrivals navigating to cart and added books failed when over quantity in stock	Website is logged in	<p>1) Open the website</p> <p>2) Click Shop Menu</p> <p>3) Now click on the Home menu button</p> <p>4) Check if Homepage has only Three arrivals</p> <p>5) The homepage can only contain three Destinations</p>	The inventory has 20 books, enter the number of books as 21 books.	Gives an error prompt "you must enter a value between 1 and 20"	Pass

			<p>6) Now random click on the image in Arrivals</p> <p>7) Navigate to the next page where the user can add that book to his cart</p> <p>8) Select more books than there are in stock.</p> <p>9) Click the add to cart button</p>			
7	Check arrivals navigate to basket and add books to basket	Website is logged in	<p>1) Open the website</p> <p>2) Click Shop Menu</p> <p>3) Now click on the Home menu button</p> <p>4) Check if Homepage has only Three arrivals</p> <p>5) The homepage can only contain three Destinations</p> <p>6) Now random click on the image in Arrivals and navigate to the next page where the user can add that book to his cart</p>	Input number of the book is 0	The screen displays the message " Giá trị lớn hơn hoặc bằng 1."	Pass

			7) Select books is 0 8) Click the "add to basket" button			
--	--	--	---	--	--	--

Bảng 3: Testcase màn hình “Home”

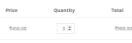
3.3.4. Testcase chức năng tại màn hình “Items”

No .	Test name	Pre condition	Test step	Data	Expected Result	Verify
1	Check the function of removing products from basket	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Randomly add any product to basket 4) Click the “Items” Menu 5) Click the delete product button as shown in the image below 		The product has been removed from the cart Showing the message “Functional Programming in JS removed . Undo?”	Pass
2	Check the function of	Website is logged in	1) Open the website 2) Click on the “Shop” menu		Deleted products	Pass

	restoring deleted products when clicking “Undo?”		3) Randomly add any product to basket 4) Click Menu “Items” 5) Click the delete product button 6) The screen shows a message that the product has been removed 7) Click “Undo?”		are restored	
3	Check the function to update the number of books when clicking the “Update basket” button	Website is logged in	1) Open the website 2) Click on the “Shop” menu 3) Randomly add any product to basket 4) Click Menu “Items” 5) In the “quantity” column write the quantity to buy 6) Click “Update basket” button	Input number of the book is 10	The system updates the number of books according to the number of user input Showing the message “Basket updated.”	Pass

4	Check display when entering product quantity is 0	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Randomly add any product to basket 4) Click the “Items” Menu 5) In the column “quantity” write the quantity as 0 6) Click the “Update Cart” button	Input number of the book is 0	The system displays "Your basket is currently empty."	Pass
5	Check to apply discount code successfully when the book price is over 450 rps	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Add a product to cart that costs more than 450 rps 4) Click Menu "Items" 5) at the checkbox “coupon code” enter the discount code 6) Click the “apply coupon” button	Book: Selenium Ruby Discount code: krishnasakinala	Book price reduced by 50rps displayed in the box “basket totals” Message display “Coupon code applied”	Pass

					successfully.”	
6	Check to apply discount code fails when the price of the book is less than or equal to 450 rps	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Add products to cart for less than 450 rps 4) Click Menu “Items” 5) At the checkbox “Coupon code” enter the discount code 6) Click “Apply Coupon” button	Book: Functional Programming in JS Discount code: krishnasakinala	The screen shows the error: “Minimum spend for this coupon is 450.00rps”	Pass
7	Blank the discount code blank in the “coupon code” box.	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Add any product to basket 2) Click Menu “Items” 3) Do not enter the discount code	Book: Functional Programming in JS Discount code: Empty	The screen shows the error: “Please enter a coupon code.”	Pass

			4) Click the “Apply Coupon” button			
8	Check the function of increasing number of books when clicking on the up arrow as shown below 	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Add any product to cart 4) Click Menu “Items” 5) At the quantity column, press the button with the arrow pointing up	Book: Functional Programming in JS Quantity: 15	The number of books increases by one unit with each click	Pass
9	Check the function to update the number of books decreasing when clicking on the up arrow as	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Add any product to basket 4) Click Menu “Items” 5) In the quantity column, press the button with the arrow pointing down	Book: Functional Programming in JS Quantity: 2	The number of books decreases by one unit with each click	Pass

	shown below					
10	Check out the payment in the shopping basket	Website is logged in	1) Open website 2) Click on the “Shop” menu 3) Add any product to cart 2) Click Menu “Items” 4) Click “Proceed to checkout” button 5) Now user can fill his details in billing details form and can opt any payment in the payment gateway like Direct bank transfer,cheque,cash or paypal. 6) Now click on "Place Order" button to complete process	Book: Functional Programming in JS Quantity: 2	Navigate to the "order details" page and display the message : "Thank you. Your order has been received ."	Pass

Bảng 4: Testcase màn hình “Items”

3.4. Chạy code test tự động

3.4.1. Keywords của code

```
Load in Interactive Console
*** Settings ***
Library SeleniumLibrary
Library FakerLibrary

Load in Interactive Console
*** Variables ***
${url} http://practice.automationtesting.in/
${browser} chrome
${email} tranxthien22223380@gmail.com
${password} Thien123@Thien123
${My account} https://practice.automationtesting.in/my-account/

*** Keywords ***
# Chung #
Load in Interactive Console
Open website
    Open Browser    ${url}    ${browser}

Load in Interactive Console
Navigate to the "My account" page
    Click Element    xpath://a[@href="https://practice.automationtesting.in/my-account/"]

Load in Interactive Console
Navigate to the "Shop" page
    Click Element    xpath://*[@id="menu-item-40"]/a

Load in Interactive Console
Navigate to the "Items" menu to see the product has been added to the basket
    Click Element    xpath://a[@href="https://practice.automationtesting.in/basket/"]
```

Hình 3. 4. 1: Keywords chung của code


```

# Register #
Load in Interactive Console
Input username and password
[Arguments]   ${email}   ${password}
${rantEmail}  Email      domain=@gmail.com
Wait For Condition    return document.readyState == "complete"
Input Text    id:reg_email   ${email}
# IF   ${EMPTY} != ${password}
Input Text    id:reg_password   ${password}
Press Keys    id:reg_password    ESC
Sleep         3s
# END

Load in Interactive Console
Fill out the information and click the register button
#Wait Until Element Is Enabled    //input[@name="register"]    timeout=25s
Click Element    //input[@name="register"]
Sleep           3s

Load in Interactive Console
User account registration successful
Wait Until Page Contains Element    /*[@id="content"]
Page Should Contain Element    /*[@id="page-36"]/div/div[1]/div

Load in Interactive Console
Invalid email registration failed
${validationText}    Get Element Attribute    //input[@id='reg_email']    validationMessage
Log    ${validationText}
Should Be Equal As Strings    ${validationText}    Please include an '@' in the email address. 'tranx222gmail.com' is missing an '@'.
Sleep    10s

Load in Interactive Console
Registration failed when email is blank and shows error message
Page Should Contain    Error: Please provide a valid email address.
Sleep    10s

```

Hình 3. 4. 2: Keywords của màn hình đăng ký “Register”

```

Load in Interactive Console
Registration fails when password is blank and shows error message
Wait Until Page Contains Element    xpath://ul[@class="woocommerce-error"]    timeout=30s
Page Should Contain Element    xpath://ul[@class="woocommerce-error"]
Sleep    10s
Load in Interactive Console
Registration failed when empty email, password blank and showing error message
Page Should Contain    Error: Please provide a valid email address.
Sleep    10s

```

Hình 3. 4. 3: Keywords của màn hình đăng ký “Register”

```

# Login #
Load in Interactive Console
Username and password login
  [Arguments]    ${email}    ${password}
  Input Text     id:username  ${email}
  Input Text     id:password  ${password}

Load in Interactive Console
After filling in the information click the login button
  Click Button   xpath://input[@name="login"]

Load in Interactive Console
Successful login with correct email and password
  Wait Until Page Contains    Hello tranxthien2222 (not tranxthien2222? Sign out)
  Page Should Contain        Hello tranxthien2222 (not tranxthien2222? Sign out)

Load in Interactive Console
Login failed when entering wrong email and password
  Wait Until Page Contains    A user could not be found with this email address.
  Page Should Contain        A user could not be found with this email address.

Load in Interactive Console
Login failed when password is blank
  Wait Until Page Contains    Error: Password is required.
  Page Should Contain        Error: Password is required.

Load in Interactive Console
Login failed when username is blank
  Wait Until Page Contains    Error: Username is required.
  Page Should Contain        Error: Username is required.
  Sleep                      5s

Load in Interactive Console
Login failed when username and password are blanks
  Wait Until Page Contains    Error: Username is required.
  Page Should Contain        Error: Username is required.

```

Hình 3. 4. 4: Keywords của màn hình đăng ký “Login”

```

Load in Interactive Console
Password has been masked
  Execute Javascript    document.getElementById('password').type = 'password'
# Execute Javascript document.getElementById('password').type = 'password': thực thi mã JavaScript để thay đổi kiểu trường nhập liệu mật khẩu t
# ẩn mật khẩu bằng các dấu sao hoặc chấm trên trình duyệt.

Load in Interactive Console
Login failed when password is capitalized
  Wait Until Page Contains    Error: The password you entered for the username tranxthien2222@gmail.com is incorrect. Lost your password?
  Page Should Contain        Error: The password you entered for the username tranxthien2222@gmail.com is incorrect. Lost your password?

```

Hình 3. 4. 5: Keywords của màn hình đăng ký “Login”

```

# Home #
Load in Interactive Console
Navigate from shop page to home page
  Click Element    xpath://*[@id="menu-item-40"]/a
  Click Element    xpath://*[@id="content"]/nav/a

Load in Interactive Console
The Home page must contains only three sliders
  ${slider_count}  Get WebElements    xpath://*[@class="n2-ss-slider-3"]//img
  Length Should Be  ${slider_count}    3
  Sleep            5s

Load in Interactive Console
The Home page must contains only three Arrivals
  ${arrivals_count}  Get WebElements    xpath://ul[@class='products']
  Length Should Be   ${arrivals_count}    3
  Sleep              5s

Load in Interactive Console
Click Random Arrival Image
  ${images} =      Get WebElements    xpath=//ul[@class='products']/li//img
  ${random_index} = Evaluate    random.randint(1, len($images)-1)
  #${random_index} = Pyint    min_value=1    max_value=len($images)    step=1
  Click Element    ${images[${random_index}]}
  Sleep            5s

```

Hình 3. 4. 6: Keywords của màn hình đăng ký “Home”

```

Load in Interactive Console
Navigate to the details page of the success book
  Wait Until Page Contains Element    xpath://button[@class="single_add_to_cart_button button alt"]
  Page Should Contain Element    xpath://button[@class="single_add_to_cart_button button alt"]
  Sleep    5s

Load in Interactive Console
Click the Description tab to view the book's details
  Click Element    xpath://li[@class="description_tab active"]

Load in Interactive Console
Successful display of book information
  Wait Until Page Contains Element    xpath://a[@href="#tab-description"]
  Page Should Contain Element    xpath://a[@href="#tab-description"]
  Sleep    10s

Load in Interactive Console
Click on the Reviews tab to view the book's details
  Click Element    xpath://li[@class="reviews_tab"]

Load in Interactive Console
Successful display of book reviews
  Wait Until Page Contains Element    xpath://a[@href="#tab-reviews"]
  Page Should Contain Element    xpath://a[@href="#tab-reviews"]

Load in Interactive Console
Enter product quantity
  Input Text    xpath://input[@name='quantity']    21

Load in Interactive Console
Enter product quantity equal to 0
  Input Text    xpath://input[@name='quantity']    0

```

Hình 3. 4. 7: Keywords của màn hình đăng ký “Home”

```

Load in Interactive Console
Click Add to basket
  Wait Until Element Is Enabled    xpath://button[@class="single_add_to_cart_button button alt"]    timeout=20s
  Click Element    xpath://button[@class="single_add_to_cart_button button alt"]
  Sleep    3s

Load in Interactive Console
Successfully added product to basket
  Wait Until Page Contains    21  "Selenium Ruby" have been added to your basket.
  Page Should Contain    21  "Selenium Ruby" have been added to your basket.

Load in Interactive Console
Add product fails and shows error message
  ${validation_Text}    Get Element Attribute    //input[@name='quantity']    validationMessage
  Log    ${validation_Text}
  Should Be Equal As Strings    ${validation_Text}    Value must be greater than or equal to 1.
  Sleep    10s

```

Hình 3. 4. 8: Keywords của màn hình đăng ký “Login”

```

# Items #

Load in Interactive Console
Click Random product
  ${products}    Get WebElements    xpath://ul[@class='products masonry-done']/li/img
  ${product_index} =    Evaluate    random.randint(0, len($products)-1)
  Click Element    ${products[${product_index}]}
  Sleep    5s

Load in Interactive Console
Click the "x" to remove the product from the basket
  Click Element    xpath://td[@class="product-remove"]

Load in Interactive Console
Display message successfully removed product from basket
  Page Should Contain Element    xpath://*[@id="page-34"]/div/div[1]/div[1]

Load in Interactive Console
Click the "undo" button to restore the deleted product
  Click Element    xpath://*[@id="page-34"]/div/div[1]
  Sleep    5s

Load in Interactive Console
Product has been restored successfully
  Wait Until Page Contains Element    //*[@id="layout"]
  Page Should Contain Element    //*[@id="layout"]
  Sleep    5s

Load in Interactive Console
Enter product quantity in Items menu
  Input Text    xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[1]/td[5]/div/input    10

Load in Interactive Console
Enter "update basket" to update the quantity you just filled in
  Click Element    xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[2]/td/input[1]
  Sleep    5s

```

Hình 3. 4. 9: Keywords của màn hình đăng ký “Items”

```
Load in Interactive Console
The product has been successfully updated
Wait Until Page Contains    Basket updated.
Page Should Contain        Basket updated.

Load in Interactive Console
Show empty cart when update quantity equal to 0
Wait Until Page Contains    Your basket is currently empty.
Page Should Contain        Your basket is currently empty.

Load in Interactive Console
Enter product quantity in Items menu (0)
Input Text    xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[1]/td[5]/div/input    0

Load in Interactive Console
Add book to basket
Click Element    xpath://a[@href="https://practice.automationtesting.in/product/selenium-ruby/"]
Click Element    xpath://button[@class="single_add_to_cart_button button alt"]
Wait Until Page Contains Element    xpath://*[@id="content"]/div[1]
Page Should Contain Element    xpath://*[@id="content"]/div[1]

Load in Interactive Console
Input coupon code
Input Text    xpath://*[@id="coupon_code"]    krishnasakinala

Load in Interactive Console
Click the apply coupon button
Click Element    xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[2]/td/div/input[2]

Load in Interactive Console
Successful discount when applying the code with books larger than 450srp
Page Should Contain Element    xpath://*[@id="page-34"]/div/div[1]/div[1]
Sleep    5s
```

Hình 3. 4. 10: Keywords của màn hình đăng ký “Items”

```
Load in Interactive Console
Add books with price less than or equal to 450
Click Element    xpath://a[@href="https://practice.automationtesting.in/product/functional-programming-in-js/"]
Click Element    xpath://button[@class="single_add_to_cart_button button alt"]
Wait Until Page Contains Element    xpath://*[@id="content"]/div[1]
Page Should Contain Element    xpath://*[@id="content"]/div[1]

Load in Interactive Console
Discount failed when applying code with books larger than 450
Wait Until Page Contains Element    xpath://*[@id="page-34"]/div/div[1]/ul
Page Should Contain Element    xpath://*[@id="page-34"]/div/div[1]/ul
Sleep    5s

Load in Interactive Console
Click apply coupon
Click Element    xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[2]/td/div/input[2]

Load in Interactive Console
Applying discount code failed when the "coupon code" box was blank.
Wait Until Page Contains Element    xpath://*[@id="page-34"]/div/div[1]/ul
Page Should Contain Element    xpath://*[@id="page-34"]/div/div[1]/ul
Sleep    5s
```

Hình 3. 4. 11: Keywords của màn hình đăng ký “Items”

3.4.2. Testcase trên Robot framework

3.4.2.1. Testcase màn hình “Register”

```
Run Suite | Debug Suite | Load in Interactive Console
*** Settings ***
Resource    ../Robot_BC_CK/Keywords.robot
Load in Interactive Console
*** Variables ***

*** Test Cases ***
# Register #
Run | Debug | Run in Interactive Console
TC1 user account registration successful
    Open website
    Navigate to the "My account" page
    Input username and password    ${email}    ${password}
    Fill out the information and click the register button
    User account registration successful
```

Hình 3. 4. 12: Testcase 1 đăng ký tài khoản thành công

```
TC2 registration failed
    Open website
    Navigate to the "My account" page
    Input username and password    email=tranx2222gmail.com    password=Thien@123Thien123
    Fill out the information and click the register button
    Invalid email registration failed
    Close Browser
```

Hình 3. 4. 13: Testcase 2 đăng ký tài khoản thất bại

```
TC3 empty email
    Open website
    Navigate to the "My account" page
    Input username and password    ${EMPTY}    ${password}
    Fill out the information and click the register button
    Registration failed when email is blank and shows error message
    Close Browser
```

Hình 3. 4. 14: Testcase 3 đăng ký tài khoản thất bại khi bỏ trống email

```
TC4 empty password
    Open website
    Navigate to the "My account" page
    Input username and password    ${email}    ${EMPTY}
    Fill out the information and click the register button
    Registration fails when password is blank and shows error message
    Close Browser
```

Hình 3. 4. 15: Testcase 4 đăng ký tài khoản thất bại khi bỏ trống mật khẩu

```

TC5 empty password and email
Open website
Navigate to the "My account" page
Input username and password    ${EMPTY}    ${EMPTY}
Fill out the information and click the register button
Registration failed when empty email, password blank and showing error message
Close Browser

```

Hình 3. 4. 16: Testcase 5 đăng ký tài khoản thất bại khi bỏ trống email và mật khẩu

3.4.2.2. Testcase màn hình “Login”

```

Run | Debug | Run in Interactive Console
TC1 login success
Open website
Navigate to the "My account" page
Username and password login    ${email}    ${password}
After filling in the information click the login button
Successful login with correct email and password
Close Browser

```

Hình 3. 4. 17: Testcase 1 đăng nhập thành công

```

TC2 incorrect password and username
Open website
Navigate to the "My account" page
Username and password login    email=thie@gmail.com    password=Thienj@123ghid
After filling in the information click the login button
Login failed when entering wrong email and password
Close Browser

```

Hình 3. 4. 18: Testcase 2 đăng nhập thất bại khi tài khoản và mật khẩu sai

```

TC3 empty password
Open website
Navigate to the "My account" page
Username and password login    ${email}    ${EMPTY}
After filling in the information click the login button
Login failed when password is blank
Close Browser

```

Hình 3. 4. 19: Testcase 3 đăng nhập thất bại khi để trống mật khẩu

```
Run | Debug | Run in Interactive Console
TC4 empty username
  Open website
  Maximize Browser Window
  Navigate to the "My account" page
  Username and password login    ${EMPTY}    ${password}
  After filling in the information click the login button
  Login failed when username is blank
  Close Browser
```

Hình 3. 4. 20: Testcase 4 đăng nhập thất bại khi để trống tài khoản

```
Run | Debug | Run in Interactive Console
TC5 empty password and username
  Open website
  Navigate to the "My account" page
  Username and password login    ${EMPTY}    ${EMPTY}
  After filling in the information click the login button
  Login failed when username and password are blanks
  Close Browser
```

Hình 3. 4. 21: Testcase 5 đăng nhập thất bại khi để trống tài khoản và mật khẩu

```
Run | Debug | Run in Interactive Console
TC6 mask password
  Open website
  Navigate to the "My account" page
  Username and password login    ${email}    ${password}
  Password has been masked
  Close Browser
```

Hình 3. 4. 22: Testcase 6 che giấu mật khẩu khi đăng nhập

```
Run | Debug | Run in Interactive Console
TC7 handles case sensitive
  Open website
  Maximize Browser Window
  Navigate to the "My account" page
  Username and password login    ${email}    ${password.upper()}
  After filling in the information click the login button
  Login failed when password is capitalized
  Close Browser
```

Hình 3. 4. 23: Testcase 7 phân biệt chữ hoa ở mật khẩu

3.4.2.3. Testcase màn hình “Home”


```
Run | Debug | Run in Interactive Console
TC1 home page has 3 sliders
  Open website
  Navigate from shop page to home page
  The Home page must contains only three sliders
  Close Browser
```

Hình 3. 4. 24: Testcase 1 home page chỉ có 3 sliders

```
Run | Debug | Run in Interactive Console
TC2 three arrivals
  Open website
  Navigate from shop page to home page
  The Home page must contains only three Arrivals
  Close Browser
```

Hình 3. 4. 25: Testcase 2 home page chỉ có 3 arrivals

```
Run | Debug | Run in Interactive Console
TC3 add products to basket
  Open website
  Navigate from shop page to home page
  The Home page must contains only three Arrivals
  Click Random Arrival Image
  Navigate to the details page of the success book
  Close Browser
```

Hình 3. 4. 26: Testcase 3 xem thông tin chi tiết của sản phẩm

```
Run | Debug | Run in Interactive Console
TC4 show description page
  Open website
  Navigate from shop page to home page
  The Home page must contains only three Arrivals
  Click Random Arrival Image
  Navigate to the details page of the success book
  Click the Description tab to view the book's details
  Successful display of book information
  Close Browser
```

Hình 3. 4. 27: Testcase 4 xem đánh mô tả sản phẩm

```
Run | Debug | Run in Interactive Console
TC5 show reviews page
  Open website
  Navigate from shop page to home page
  The Home page must contains only three Arrivals
  Click Random Arrival Image
  Navigate to the details page of the success book
  Click on the Reviews tab to view the book's details
  Successful display of book reviews
  Close Browser
```

Hình 3. 4. 28: Testcase 5 xem đánh giá sản phẩm

```
TC6 add products
  Open website
  Navigate from shop page to home page
  The Home page must contains only three Arrivals
  Click Random Arrival Image
  Navigate to the details page of the success book
  Enter product quantity
  Click Add to basket
  Successfully added product to basket
  Close Browser
```

Hình 3. 4. 29: Testcase 6 thêm sản phẩm vào giỏ hàng

```
Run | Debug | Run in Interactive Console
TC7 add products with quantity equal to 0
  Open website
  Navigate from shop page to home page
  The Home page must contains only three Arrivals
  Click Random Arrival Image
  Navigate to the details page of the success book
  Enter product quantity equal to 0
  Click Add to basket
  Add product fails and shows error message
  Close Browser
```

Hình 3. 4. 30: Testcase 7 thêm sản phẩm thất bại khi đầu vào số lượng bằng

3.4.2.4. Testcase màn hình “Items”

```
Run | Debug | Run in Interactive Console
TC1 Remove product from basket
  Open website
  Navigate to the "Shop" page
  Click Random product
  Click Add to basket
  Navigate to the "Items" menu to see the product has been added to the basket
  Click the "x" to remove the product from the basket
  Display message successfully removed product from basket
```

Hình 3. 4. 31: Testcase 1 xoá sản phẩm ra khỏi giỏ hàng

```
Run | Debug | Run in Interactive Console
TC2 restoring deleted products
  Open website
  Navigate to the "Shop" page
  Click Random product
  Click Add to basket
  Navigate to the "Items" menu to see the product has been added to the basket
  Click the "x" to remove the product from the basket
  Display message successfully removed product from basket
  Click the "undo" button to restore the deleted product
  Product has been restored successfully
  Close Browser
```

Hình 3. 4. 32: : Testcase 2 khôi phục sản phẩm đã xoá

```
Run | Debug | Run in Interactive Console
TC3 update basket
  Open website
  Navigate to the "Shop" page
  Click Random product
  Click Add to basket
  Navigate to the "Items" menu to see the product has been added to the basket
  Enter product quantity in Items menu
  Enter "update basket" to update the quantity you just filled in
  The product has been successfully updated
  Close Browser
```

Hình 3. 4. 33: Testcase 3 cập nhật giỏ hàng

```

Close Browser
Run | Debug | Run in Interactive Console
TC4 product quantity is 0
  Open website
  Navigate to the "Shop" page
  Click Random product
  Click Add to basket
  Navigate to the "Items" menu to see the product has been added to the basket
  Enter product quantity in Items menu (0)
  Enter "update basket" to update the quantity you just filled in
  Show empty cart when update quantity equal to 0
  Close Browser

```

Hình 3. 4. 34: Testcase 4 cập nhật sản phẩm với số lượng bằng 0

```

Close Browser
Run | Debug | Run in Interactive Console
TC5 discount code greater than 450
  Open website
  Navigate to the "Shop" page
  Add book to basket
  Navigate to the "Items" menu to see the product has been added to the basket
  Input coupon code
  Click the apply coupon button
  Successful discount when applying the code with books larger than 450srp
  Close Browser

```

Hình 3. 4. 35: Testcase 5 áp dụng mã giảm giá lớn hơn 450srp

```

Run | Debug | Run in Interactive Console
TC6 discount code less than or equal to 450
  Open website
  Navigate to the "Shop" page
  Add books with price less than or equal to 450
  Navigate to the "Items" menu to see the product has been added to the basket
  Input coupon code
  Click the apply coupon button
  Discount failed when applying code with books larger than 450
  Close Browser

```

Hình 3. 4. 36: Testcase 6 áp dụng mã giảm giá nhỏ hơn hoặc bằng 450rsp

```
Close browser
Run | Debug | Run in Interactive Console
TC7 blank coupon code
Open website
Navigate to the "Shop" page
Click Random product
Click Add to basket
Navigate to the "Items" menu to see the product has been added to the basket
Click the apply coupon button
Applying discount code failed when the "coupon code" box was blank.
Close Browser
```

Hình 3. 4. 37: Testcase 7 bỏ trống ô mã giảm giá

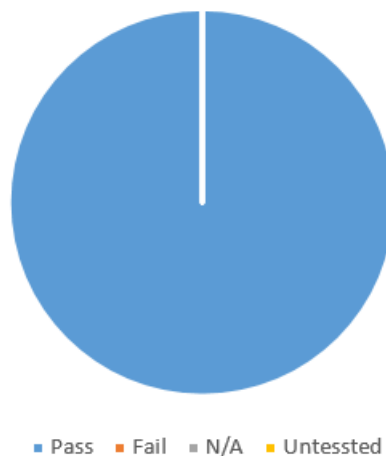
3.5. Kết quả test

3.5.1. Kết quả test màn hình “Register”

	Pass	Fail	N/A	Untessted	Tổng cộng
Tab exam	5	0	0	0	5
	100%	0%	0%	0%	100%

Bảng 5: Bảng kết quả màn hình “Register”

Kết quả chạy testcase màn hình "Register"



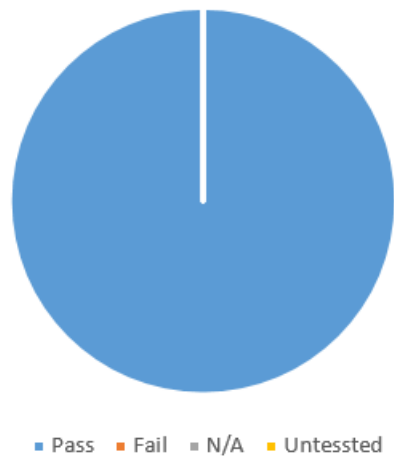
Hình 3. 5. 1: Kết quả testcase màn hình “Register”

3.5.2. Kết quả test màn hình “Login”

	Pass	Fail	N/A	Untessted	Tổng cộng
Tab exam	7	0	0	0	7
	100%	0%	0%	0%	100%

Bảng 6: Bảng kết quả màn hình “Login”

Kết quả testcase màn hình "Login"



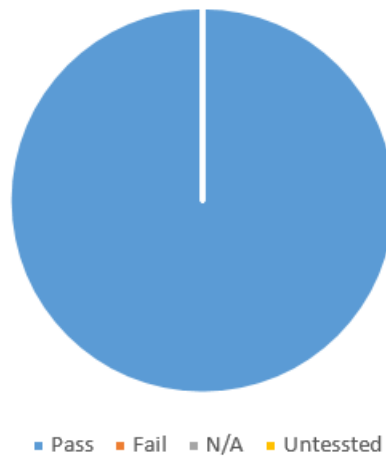
Hình 3. 5. 2: Kết quả testcase màn hình “Login”

3.5.3. Kết quả test màn hình “Home”

	Pass	Fail	N/A	Untessted	Tổng cộng
Tab exam	7	0	0	0	7
	100%	0%	0%	0%	100%

Bảng 7: Bảng kết quả testcase màn hình “Home”

Kết quả chạy testcase màn hình "Home"



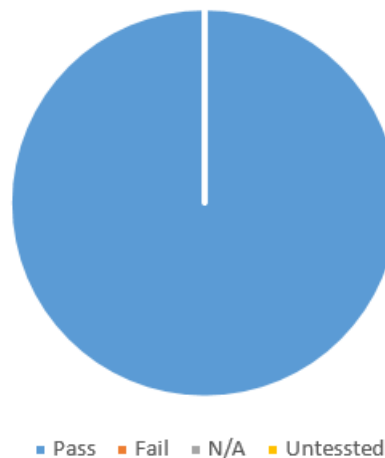
Hình 3. 5. 3: Kết quả testcase màn hình “Home”

3.5.4. Kết quả test màn hình “Items”

	Pass	Fail	N/A	Untessted	Tổng cộng
Tab exam	7	0	0	0	7
	100%	0%	0%	0%	100%

Bảng 8: Bảng kết quả testcase màn hình “Items”

Kết Quả Chạy Test Màn Hình "Items"



Hình 3. 5. 4: Kết quả testcase màn hình “Items”

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

- Kết quả đạt được:

Trong thời gian thực hiện tập tin tại công ty TMA Bình Định, qua quá trình tìm hiểu, nghiên cứu cơ sở lý thuyết và phát triển ứng dụng công nghệ, em đã đạt được những kết quả sau:

- Nắm cơ sở lý thuyết của Kiểm thử.
- Các loại kiểm thử phần mềm.
- Biết cách viết testcase và sử dụng cơ bản robot framework.
- Áp dụng kiến thức đã học để test tự động cho trang web.
- Hiểu được cách làm việc của một tester

- Hạn chế:

- Chưa viết đầy đủ các trường hợp kiểm thử
- Khả năng viết Automation còn hạn chế

- Hướng phát triển:

- Thực hiện kiểm thử trên nhiều trình duyệt web khác nhau.
- Bổ sung các ràng buộc trong chương trình kiểm thử.

TÀI LIỆU THAM KHẢO

<https://robotframework.org/SeleniumLibrary/SeleniumLibrary.html>

<https://viblo.asia/p/kiem-thu-phan-mem-cac-loai-kiem-thu-test-types-va-kiem-thu-bao-tri-XL6lAvER5ek>

<https://viblo.asia/p/phuong-phap-kiem-thu-phan-mem-nguyen-ly-kiem-thu-phan-mem-OeVKB3kJZkW>

<https://viblo.asia/p/tim-hieu-ve-cac-cap-do-kiem-thu-test-levels-4P856drAZY3>

<https://www.guru99.com/software-testing.html>

<https://www.w3schools.com/python/default.asp>

<https://docs.robotframework.org/docs>

PHỤ LỤC

1. Keywords

```
*** Settings ***
Library      SeleniumLibrary
Library      FakerLibrary

*** Variables ***
${url}       http://practice.automationtesting.in/
${browser}   chrome
${email}     tranxthien22223380@gmail.com
${password}   Thien123@Thien123
${My account} https://practice.automationtesting.in/my-account/

*** Keywords ***
# Chung #
Open website
    Open Browser    ${url}    ${browser}

    Navigate to the "My account" page
        Click
    Element    xpath://a[@href="https://practice.automationtesting.in/my-account/"]

    Navigate to the "Shop" page
        Click Element    xpath://*[@id="menu-item-40"]/a

    Navigate to the "Items" menu to see the product has been added to the basket
        Click
    Element    xpath://a[@href="https://practice.automationtesting.in/basket/"]
```

```
# Register #
Input username and password
    [Arguments]    ${email}    ${password}
    #${rantEmail}   Email    domain=@gmail.com
    Wait For Condition    return document.readyState == "complete"
    Input Text    id:reg_email    ${email}
#    IF    ${EMPTY} != ${password}
    Input Text    id:reg_password    ${password}
    Press Keys    id:reg_password    ESC
    Sleep    3s
#    END

Fill out the information and click the register button
    #Wait Until Element Is Enabled    //input[@name="register"]    timeout=25s
    Click Element    //input[@name="register"]
```

```

Sleep      3s

User account registration successful
Wait Until Page Contains Element    /*[@id="content"]
Page Should Contain Element        /*[@id="page-36"]/div/div[1]/div

Invalid email registration failed
    ${validationText}    Get Element
Attribute    //input[@id='reg_email']    validationMessage
Log    ${validationText}
    Should Be Equal As Strings    ${validationText}    Please include an '@' in the
email address. 'tranx2222gmail.com' is missing an '@'.
Sleep      10s

Registration failed when email is blank and shows error message
Page Should Contain    Error: Please provide a valid email address.
Sleep      10s

Registration fails when password is blank and shows error message
Wait Until Page Contains Element    xpath://ul[@class="woocommerce-
error"]    timeout=30s
Page Should Contain Element    xpath://ul[@class="woocommerce-error"]
Sleep      10s

Registration failed when empty email, password blank and showing error message
Page Should Contain    Error: Please provide a valid email address.
Sleep      10s

```

```

# Login #
Username and password login
    [Arguments]    ${email}    ${password}
    Input Text    id:username    ${email}
    Input Text    id:password    ${password}

After filling in the information click the login button
Click Button    xpath://input[@name="login"]

Successful login with correct email and password
Wait Until Page Contains    Hello tranxthien2222 (not tranxthien2222? Sign out)
Page Should Contain    Hello tranxthien2222 (not tranxthien2222? Sign out)

Login failed when entering wrong email and password
Wait Until Page Contains    A user could not be found with this email address.
Page Should Contain    A user could not be found with this email address.

Login failed when password is blank
Wait Until Page Contains    Error: Password is required.

```

```

    Page Should Contain    Error: Password is required.

Login failed when username is blank
    Wait Until Page Contains    Error: Username is required.
    Page Should Contain    Error: Username is required.
    Sleep    5s

Login failed when username and password are blanks
    Wait Until Page Contains    Error: Username is required.
    Page Should Contain    Error: Username is required.

Password has been masked
    Execute Javascript    document.getElementById('password').type = 'password'
# Execute Javascript document.getElementById('password').type = 'password': thực
thi mã JavaScript để thay đổi kiểu trường nhập liệu mật khẩu thành "password".
# ẩn mật khẩu bằng các dấu sao hoặc chấm trên trình duyệt.

Login failed when password is capitalized
    Wait Until Page Contains    Error: The password you entered for the username
tranxthien2222@gmail.com is incorrect. Lost your password?
    Page Should Contain    Error: The password you entered for the username
tranxthien2222@gmail.com is incorrect. Lost your password?

```

```

# Home #
Navigate from shop page to home page
    Click Element    xpath://*[@id="menu-item-40"]/a
    Click Element    xpath://*[@id="content"]/nav/a

The Home page must contains only three sliders
    ${slider_count}    Get WebElements    xpath://*[@class="n2-ss-slider-
3"]//img
    Length Should Be    ${slider_count}    3
    Sleep    5s

The Home page must contains only three Arrivals
    ${arrivals_count}    Get WebElements    xpath://ul[@class='products']
    Length Should Be    ${arrivals_count}    3
    Sleep    5s

Click Random Arrival Image
    ${images} =    Get WebElements    xpath=//ul[@class='products']/li//img
    ${random_index} =    Evaluate    random.randint(1, len($images)-1)
    #${random_index} =    Pyint    min_value=1    max_value=len($images)    step=1
    Click Element    ${images[${random_index}]}
    Sleep    5s

# Evaluate: thực thi các biểu thức Python trong lệnh Robot Framework.

```

```

# random.randint(0, len($elements)-1): biểu thức Python. random.randint(a, b) sẽ
tạo một số nguyên ngẫu nhiên
# trong khoảng từ a đến b (bao gồm cả a và b).
# ${random_index}: Đây là biến của Robot Framework để lưu giá trị của biểu thức
Python. Sau khi lệnh thực hiện,
# random_index sẽ giữ giá trị số nguyên ngẫu nhiên.
# Click Element ${elements[${random_index}]}
# Click Element: Đây là keyword của thư viện SeleniumLibrary để thực hiện việc nhấn
vào một phần tử trên trang web.
# ${elements[${random_index}]}: Đây là cách bạn truyền một phần tử từ danh sách
${elements} bằng cách sử dụng chỉ số ${random_index}.
# Điều này sẽ làm cho Robot Framework nhấn vào phần tử ngẫu nhiên đã chọn.

Navigate to the details page of the success book
    Wait Until Page Contains
Element    xpath://button[@class="single_add_to_cart_button button alt"]
    Page Should Contain Element    xpath://button[@class="single_add_to_cart_button
button alt"]
    Sleep    5s
# chờ đến khi một phần tử xuất hiện trên trang web. Nút xpath "add to basket" khi
nhấn vào sách

Click the Description tab to view the book's details
    Click Element    xpath://li[@class="description_tab active"]

Successful display of book information
    Wait Until Page Contains Element    xpath://a[@href="#tab-description"]
    Page Should Contain Element    xpath://a[@href="#tab-description"]
    Sleep    10s

Click on the Reviews tab to view the book's details
    Click Element    xpath://li[@class="reviews_tab"]

Successful display of book reviews
    Wait Until Page Contains Element    xpath://a[@href="#tab-reviews"]
    Page Should Contain Element    xpath://a[@href="#tab-reviews"]

Enter product quantity
    Input Text    xpath://input[@name='quantity']    21

Enter product quantity equal to 0
    Input Text    xpath://input[@name='quantity']    0

Click Add to basket
    Wait Until Element Is
Enabled    xpath://button[@class="single_add_to_cart_button button
alt"]    timeout=20s

```

```

Click Element    xpath://button[@class="single_add_to_cart_button button alt"]
Sleep           3s

Successfully added product to basket
Wait Until Page Contains    21 x "Selenium Ruby" have been added to your
basket.
Page Should Contain        21 x "Selenium Ruby" have been added to your basket.

Add product fails and shows error message
${validation_Text}    Get Element
Attribute    //input[@name='quantity']    validationMessage
Log    ${validation_Text}
Should Be Equal As Strings    ${validation_Text}    Value must be greater than
or equal to 1.
Sleep    10s
# Items #
Click Random product
${products}    Get WebElements    xpath://ul[@class='products masonry-
done']/li//img
${product_index} =    Evaluate    random.randint(0, len($products)-1)
Click Element    ${products[${product_index}]}
Sleep    5s

Click the "x" to remove the product from the basket
Click Element    xpath://td[@class="product-remove"]

Display message successfully removed product from basket
Page Should Contain Element    xpath://*[@id="page-34"]/div/div[1]/div[1]

Click the "undo" button to restore the deleted product
Click Element    xpath://*[@id="page-34"]/div/div[1]
Sleep    5s

Product has been restored successfully
Wait Until Page Contains
Element    //*[@id="layout"]    #xpath://*[@id="pagewrap"]
Page Should Contain
Element    //*[@id="layout"]    #xpath://*[@id="pagewrap"]
Sleep    5s

Enter product quantity in Items menu
Input Text    xpath://*[@id="page-
34"]/div/div[1]/form/table/tbody/tr[1]/td[5]/div/input    10

Enter "update basket" to update the quantity you just filled in
Click Element    xpath://*[@id="page-
34"]/div/div[1]/form/table/tbody/tr[2]/td/input[1]

```

```

Sleep      5s

The product has been successfully updated
Wait Until Page Contains      Basket updated.
Page Should Contain      Basket updated.

Show empty cart when update quantity equal to 0
Wait Until Page Contains      Your basket is currently empty.
Page Should Contain      Your basket is currently empty.
Enter product quantity in Items menu (0)
Input Text      xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[1]/td[5]/div/input      0

Add book to basket
Click
Element      xpath://a[@href="https://practice.automationtesting.in/product/selenium-ruby/"]
Click Element      xpath://button[@class="single_add_to_cart_button button alt"]
Wait Until Page Contains Element      xpath://*[@id="content"]/div[1]
Page Should Contain Element      xpath://*[@id="content"]/div[1]

Input coupon code
Input Text      xpath://*[@id="coupon_code"]      krishnasakinala

Click the apply coupon button
Click Element      xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[2]/td/div/input[2]

Successful discount when applying the code with books larger than 450srp
Page Should Contain Element      xpath://*[@id="page-34"]/div/div[1]/div[1]
Sleep      5s

Add books with price less than or equal to 450
Click
Element      xpath://a[@href="https://practice.automationtesting.in/product/functional-programming-in-js/"]
Click Element      xpath://button[@class="single_add_to_cart_button button alt"]
Wait Until Page Contains Element      xpath://*[@id="content"]/div[1]
Page Should Contain Element      xpath://*[@id="content"]/div[1]

Discount failed when applying code with books larger than 450
Wait Until Page Contains Element      xpath://*[@id="page-34"]/div/div[1]/ul
Page Should Contain Element      xpath://*[@id="page-34"]/div/div[1]/ul
Sleep      5s

Click apply coupon

```

```
Click Element    xpath://*[@id="page-34"]/div/div[1]/form/table/tbody/tr[2]/td/div/input[2]
```

Applying discount code failed when the "coupon code" box was blank.

```
Wait Until Page Contains Element    xpath://*[@id="page-34"]/div/div[1]/ul
Page Should Contain Element    xpath://*[@id="page-34"]/div/div[1]/ul
Sleep    5s
```

2. Testcase

```
*** Settings ***
Resource    ../Robot_BC_CK/Keywords.robot
*** Variables ***

*** Test Cases ***
# Register #
TC1 user account registration successful
    Open website
    Navigate to the "My account" page
    Input username and password    ${email}    ${password}
    Fill out the information and click the register button
    User account registration successful
TC2 registration failed
    Open website
    Navigate to the "My account" page
    Input username and
password    email=tranx2222gmail.com    password=Thien@123Thien123
    Fill out the information and click the register button
    Invalid email registration failed
    Close Browser
TC3 empty email
    Open website
    Navigate to the "My account" page
    Input username and password    ${EMPTY}    ${password}
    Fill out the information and click the register button
    Registration failed when email is blank and shows error message
    Close Browser
TC4 empty password
    Open website
    Navigate to the "My account" page
    Input username and password    ${email}    ${EMPTY}
    Fill out the information and click the register button
    Registration fails when password is blank and shows error message
    Close Browser
TC5 empty password and email
```



```
Open website
Navigate to the "My account" page
Input username and password    ${EMPTY}    ${EMPTY}
Fill out the information and click the register button
Registration failed when empty email, password blank and showing error
message
Close Browser
```

Login

TC1 login success

```
Open website
Navigate to the "My account" page
Username and password login    ${email}    ${password}
After filling in the information click the login button
Successful login with correct email and password
Close Browser
```

TC2 incorrect password and username

```
Open website
Navigate to the "My account" page
Username and password login    email=thie@gmail.com    password=Thienj@123ghid
After filling in the information click the login button
Login failed when entering wrong email and password
Close Browser
```

TC3 empty password

```
Open website
Navigate to the "My account" page
Username and password login    ${email}    ${EMPTY}
After filling in the information click the login button
Login failed when password is blank
Close Browser
```

TC4 empty username

```
Open website
Maximize Browser Window
Navigate to the "My account" page
Username and password login    ${EMPTY}    ${password}
After filling in the information click the login button
Login failed when username is blank
Close Browser
```

TC5 empty password and username

```
Open website
Navigate to the "My account" page
Username and password login    ${EMPTY}    ${EMPTY}
After filling in the information click the login button
Login failed when username and password are blanks
Close Browser
```

TC6 mask password

```

    Open website
    Navigate to the "My account" page
    Username and password login    ${email}    ${password}
    Password has been masked
    Close Browser
TC7 handles case sensitive
    Open website
    Maximize Browser Window
    Navigate to the "My account" page
    Username and password login    ${email}    ${password.upper()}
    After filling in the information click the login button
    Login failed when password is capitalized
    Close Browser

```

```

# Home page #
TC1 home page has 3 sliders
    Open website
    Navigate from shop page to home page
    The Home page must contains only three sliders
    Close Browser
TC2 three arrivals
    Open website
    Navigate from shop page to home page
    The Home page must contains only three Arrivals
    Close Browser
TC3 add products to basket
    Open website
    Navigate from shop page to home page
    The Home page must contains only three Arrivals
    Click Random Arrival Image
    Navigate to the details page of the success book
    Close Browser
TC4 show description page
    Open website
    Navigate from shop page to home page
    The Home page must contains only three Arrivals
    Click Random Arrival Image
    Navigate to the details page of the success book
    Click the Description tab to view the book's details
    Successful display of book information
    Close Browser
TC5 show reviews page
    Open website
    Navigate from shop page to home page
    The Home page must contains only three Arrivals
    Click Random Arrival Image

```

```

    Navigate to the details page of the success book
    Click on the Reviews tab to view the book's details
    Successful display of book reviews
    Close Browser
TC6 add products
    Open website
    Navigate from shop page to home page
    The Home page must contains only three Arrivals
    Click Random Arrival Image
    Navigate to the details page of the success book
    Enter product quantity
    Click Add to basket
    Successfully added product to basket
    Close Browser
TC7 add products with quantity equal to 0
    Open website
    Navigate from shop page to home page
    The Home page must contains only three Arrivals
    Click Random Arrival Image
    Navigate to the details page of the success book
    Enter product quantity equal to 0
    Click Add to basket
    Add product fails and shows error message
    Close Browser

```

```

# Items #
TC1 Remove product from basket
    Open website
    Navigate to the "Shop" page
    Click Random product
    Click Add to basket
    Navigate to the "Items" menu to see the product has been added to the basket
    Click the "x" to remove the product from the basket
    Display message successfully removed product from basket
TC2 restoring deleted products
    Open website
    Navigate to the "Shop" page
    Click Random product
    Click Add to basket
    Navigate to the "Items" menu to see the product has been added to the basket
    Click the "x" to remove the product from the basket
    Display message successfully removed product from basket
    Click the "undo" button to restore the deleted product
    Product has been restored successfully
    Close Browser
TC3 update basket

```

```

Open website
Navigate to the "Shop" page
Click Random product
Click Add to basket
Navigate to the "Items" menu to see the product has been added to the basket
Enter product quantity in Items menu
Enter "update basket" to update the quantity you just filled in
The product has been successfully updated
Close Browser
TC4 product quantity is 0
Open website
Navigate to the "Shop" page
Click Random product
Click Add to basket
Navigate to the "Items" menu to see the product has been added to the basket
Enter product quantity in Items menu (0)
Enter "update basket" to update the quantity you just filled in
Show empty cart when update quantity equal to 0
Close Browser
TC5 discount code greater than 450
Open website
Navigate to the "Shop" page
Add book to basket
Navigate to the "Items" menu to see the product has been added to the basket
Input coupon code
Click the apply coupon button
Successful discount when applying the code with books larger than 450srp
Close Browser
TC6 discount code less than or equal to 450
Open website
Navigate to the "Shop" page
Add books with price less than or equal to 450
Navigate to the "Items" menu to see the product has been added to the basket
Input coupon code
Click the apply coupon button
Discount failed when applying code with books larger than 450
Close Browser
TC7 blank coupon code
Open website
Navigate to the "Shop" page
Click Random product
Click Add to basket
Navigate to the "Items" menu to see the product has been added to the basket
Click the apply coupon button
Applying discount code failed when the "coupon code" box was blank.
Close Browser

```