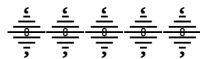


TRƯỜNG ĐẠI HỌC PHENIKAA
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO TỔNG QUAN

Đề tài: “Kiểm định trang web Tính thuế thu nhập cá nhân”

20010812	Nguyễn Thị Trang Nhung	20010812@st.phenikaa-uni.edu.vn
20010831	Hoàng Minh Thu	20010831@st.phenikaa-uni.edu.vn
20010795	Âu Thị Quý	20010795@st.phenikaa-uni.edu.vn
20010823	Hoàng Đình Thái	20010823@st.phenikaa-uni.edu.vn
20010744	Tổng Minh Khang	20010744@st.phenikaa-uni.edu.vn

Giảng viên hướng dẫn : Mai Xuân Tráng
Khoa: Công nghệ thông tin

Hà Nội, 06/2023

MỤC LỤC

1. Giới thiệu.....	6
1.1 Mục đích.....	7
1.2 Tổng quan.....	8
1.3 Phạm vi.....	9
1.4 Các định nghĩa và các từ viết tắt được dùng.....	10
1.4.1 Các định nghĩa.....	10
1.4.2 Các từ viết tắt.....	10
1.5 Những người sử dụng tài liệu này.....	10
2. Tài liệu tham khảo.....	10
3. Lịch trình công việc.....	11
4. Những yêu cầu về tài nguyên.....	12
4.1 Phần cứng.....	12
4.2 Phần mềm.....	12
4.3 Công cụ kiểm thử.....	12
4.4 Môi trường kiểm thử.....	13
4.5. Nhân sự.....	14
4.5.1 Vai trò và trách nhiệm.....	14
5. Phạm vi kiểm thử.....	14
5.1. Các yêu cầu chức năng.....	14
5.1 Những chức năng được kiểm thử.....	18
5.2 Những chức năng chưa được kiểm thử.....	23
6. Chiến lược kiểm thử.....	27
6.1 Test Stages (Các cấp độ kiểm thử).....	27
6.2 Các loại kiểm thử.....	29
7. Điều kiện chấp nhận.....	39
8. Defect Tracking.....	40
8.1 Phân loại lỗi.....	40
8.2 Quy trình xử lý lỗi.....	41
9. Test Deliverables.....	42
9.1 Test Cases.....	42
9.2 Test Reports - Báo cáo kiểm thử.....	44

BẢNG

Bảng 1: Các định nghĩa.....	9
Bảng 2: Lịch trình công việc.....	10
Bảng 3:Phân cứng.....	11
Bảng 4: Phần mềm.....	11
Bảng 5: Công cụ kiểm thử.....	11
Bảng 6: Vai trò trách nhiệm.....	13
Bảng 7.1: Use Case Login.....	18
Bảng 7.2: Mô tả data Login.....	18
Bảng 8.1: Use Case Register.....	19
Bảng 8.2: Mô phỏng data Register.....	20
Bảng 9.1: Use Case tính thuế.....	22
Bảng 9.2: Mô phỏng data tính thuế.....	22
Bảng 10.1: Use Case Cấp mã thuế.....	24
Bảng 10.2: Mô phỏng data Cấp mã thuế.....	24
Bảng 11.1: Use Case đóng thuế.....	26
Bảng 11.2: Mô phỏng data đóng thuế.....	26
Bảng 12: Test report.....	38
Bảng 13: Test case Login.....	44
Bảng 14: Test case Register.....	45

HÌNH ẢNH

Hình 1.1: Biểu đồ use case tổng quan.....	15
Hình 1.2: Biểu đồ cấp mã số thuế.....	16
Hình 1.3 :biểu đồ chức năng đóng thuế.....	16
Hình 2. Các cấp độ kiểm thử phần mềm.....	28
Hình 3. Quy trình kiểm thử phần mềm.....	30
Hình 5:Auto Test Selenium test case tính thuế.....	46
Hình 5:Giao diện đăng ký.....	47
Hình 6:Giao diện đăng nhập.....	47
Hình 7:Giao diện Web default.....	48
Hình 8:Giao diện Web Result.....	48
Hình 9:Config test hiệu năng Login.....	49
Hình 10:result test hiệu năng Login.....	49

1. Giới thiệu

Thuế thu nhập cá nhân (TNCN) ra đời rất lâu trong lịch sử thuế khóa và được coi là loại thuế quan trọng nhất để thực hiện mục tiêu công bằng xã hội. Cho đến nay, thuế TNCN đã trở thành một loại thuế cơ bản, giữ vai trò trung tâm trong hệ thống thuế của các quốc gia trên thế giới, đặc biệt là các nước phát triển. Bởi ngoài mục tiêu phân phối lại, thuế TNCN còn là hình thức thuế có tỷ trọng đóng góp cao trong tổng nguồn thu ngân sách của nhà nước (NSNN).

Tính quan trọng của tính thuế thu nhập cá nhân: Thuế thu nhập cá nhân là một khía cạnh quan trọng trong hệ thống thuế của mỗi quốc gia. Nắm vững cách tính thuế và đảm bảo tính chính xác của việc tính toán thuế là rất quan trọng để người dân và doanh nghiệp tuân thủ đúng quy định pháp luật và tránh vi phạm.

Sự phổ biến của trang web tính thuế: Trang web tính thuế thu nhập cá nhân được cung cấp bởi các cơ quan thuế hoặc công ty phần mềm thuế là một công cụ thông thường được sử dụng bởi người dân và doanh nghiệp để tính toán thuế. Đảm bảo tính chính xác và hiệu quả của trang web này là rất quan trọng để đảm bảo sự tin cậy và sự hài lòng của người sử dụng.

Tính chính xác và tính pháp lý: Tính chính xác của việc tính thuế là yếu tố cốt lõi. Kiểm định trang web tính thuế thu nhập cá nhân giúp xác định xem trang web có tính toán đúng và tuân thủ đúng quy định pháp luật về thuế hay không. Nếu có sai sót hoặc không chính xác, điều này có thể gây ra hậu quả pháp lý và tài chính cho người sử dụng.

Sự tiện lợi và tiết kiệm thời gian: Trang web tính thuế thu nhập cá nhân giúp người dân và doanh nghiệp tiết kiệm thời gian và công sức trong

việc tính toán thuế. Tuy nhiên, nếu trang web không hoạt động đúng cách hoặc tính toán không chính xác, người dùng có thể gặp rủi ro phát sinh sai sót trong việc tính thuế và phải đối mặt với những hậu quả không mong muốn.

Nhu cầu kiểm định và cải thiện: Kiểm định trang web tính thuế thu nhập cá nhân giúp xác định các điểm mạnh và điểm yếu của trang web hiện tại. Nếu phát hiện ra những vấn đề, người ta có thể đề xuất các cải tiến để cải thiện tính.

Vì tầm quan trọng của trang web tính thuế thu nhập cá nhân đó, nhóm em đã quyết định chọn đề tài nghiên cứu cách test cho trang web.

1.1 Mục đích

- Tài liệu kế hoạch kiểm thử này đưa ra các mục đích sau:
 - Xác định thông tin cơ bản về dự án và các thành phần chức năng được kiểm thử và không được kiểm thử.
 - Liệt kê những yêu cầu cho việc kiểm thử (Test Requirements).
 - Những chiến lược kiểm thử nên được sử dụng.
 - Ước lượng những yêu cầu về tài nguyên và chi phí cho việc kiểm thử.
 - Những tài liệu được lập sau khi hoàn thành việc kiểm thử.
- Mục đích của test plan hệ thống tính thuế là xác nhận rằng hệ thống tính thuế đang hoạt động chính xác và đáp ứng được các yêu cầu và tiêu chuẩn chất lượng đã được định sẵn. Test plan sẽ bao gồm các kịch bản thử nghiệm được thiết kế để kiểm tra tính năng và hiệu suất của hệ thống, đảm bảo rằng nó đáp ứng được các yêu cầu về độ chính xác, độ tin cậy, khả năng mở rộng và tương thích với các hệ thống khác. Kết quả của test plan sẽ cung cấp thông tin về các lỗi và

vấn đề trong hệ thống và giúp cho các nhà phát triển có thể sửa chữa và cải tiến hệ thống để đáp ứng được các tiêu chuẩn và yêu cầu của khách hàng và người dùng cuối.

1.2 Tổng quan

- Phần mềm hệ thống tính thuế thu nhập cá nhân là một hệ thống phần mềm tính toán và quản lý thuế thu nhập cá nhân cho các cá nhân có thu nhập đến từ các nguồn khác nhau. Phần mềm này được thiết kế để hỗ trợ các nhân viên kế toán và cán bộ thuế trong quá trình tính toán và nộp thuế thu nhập cá nhân đầy đủ và chính xác.
- Yêu cầu của phần mềm này bao gồm:
 - + Tính toán các khoản thuế thu nhập cá nhân phù hợp với các quy định thuế của nhà nước.
 - + Cập nhật và lưu trữ thông tin thuế thu nhập cá nhân của các cá nhân trong hệ thống.
 - + Hỗ trợ các công cụ quản lý và tạo báo cáo về các khoản thuế đã nộp và chưa nộp, số tiền thuế phải nộp, người đóng thuế và các thông tin liên quan khác.
 - + Có khả năng thực hiện kiểm tra và phân tích các lỗi trong quá trình tính toán thuế và cung cấp thông báo lỗi cho người sử dụng.
- Các chức năng của phần mềm bao gồm:
 - + Tính toán và hiển thị các khoản thuế phù hợp với thu nhập cá nhân của người sử dụng.

- + Cho phép người sử dụng nhập và lưu trữ thông tin thuế thu nhập cá nhân của mình vào hệ thống.
- + Cung cấp các công cụ quản lý và tạo báo cáo liên quan đến các khoản thuế và người đóng thuế.
- + Cung cấp thông tin về các quy định thuế mới nhất và các hướng dẫn về việc nộp thuế.
- + Hỗ trợ các tính năng bảo mật để đảm bảo an toàn và bảo mật thông tin người dùng.
- Các việc cần kiểm thử bao gồm:
 - + Kiểm tra tính chính xác của các kết quả tính toán thuế.
 - + Kiểm tra tính năng và khả năng mở rộng của hệ thống.
 - + Kiểm tra tính tương thích với các hệ thống khác.
 - + Kiểm tra tính đáp ứng và hiệu suất của hệ thống.
 - + Kiểm tra tính bảo mật và độ tin cậy của hệ thống.
 - + Kiểm tra các trường hợp đặc biệt và các tình huống lỗi.

1.3 Phạm vi

- Tài liệu kế hoạch kiểm thử này được áp dụng cho việc kiểm thử những chức năng áp dụng trong bài tập lớn
- Các dịch vụ, chức năng cần kiểm tra:
 - Tìm kiếm cơ bản
 - Tạo tài khoản và đăng nhập
 - Quản lý hồ sơ cá nhân
 - Các chức năng chính của phần mềm

1.4 Các định nghĩa và các từ viết tắt được dùng

1.4.1 Các định nghĩa

Thuật ngữ	Định nghĩa
Tên phần mềm	Hệ thống tính thuế thu nhập cá nhân
Thông tin	Các thông tin dữ liệu có trong phần mềm
Chức năng chính	Mô tả các chức năng chính
Liên kết	Liên kết hoặc có thể đăng nhập hệ thống khác

Bảng 1.1: Các định nghĩa

1.4.2 Các từ viết tắt

Thuật ngữ	Định nghĩa
TNCN	Thuế thu nhập cá nhân
NSNN	ngân sách của nhà nước

Bảng 1.1: Các từ viết tắt

1.5 Những người sử dụng tài liệu này

Các thầy cô, giảng viên và sinh viên sử dụng tài liệu

2. Tài liệu tham khảo

<https://viblo.asia/p/kiem-thu-phan-mem-la-gi-quy-trinh-kiem-thu-phan-mem-1VgZvayYKAw>

<https://viblo.asia/p/quy-trinh-kiem-thu-phan-mem-software-testing-life-cycle-stlc-Qbq5QLvmlD8>

<https://viblo.asia/p/php-unit-test-101-gioi-thieu-ve-phpunit-63vKjJEyK2R>

3. Lịch trình công việc

Milestone	Duration	Start Date	End Date
Đặc tả usecase	1 hour	May 29, 2023	May 29,2023
Xây dựng hệ thống tính thuế	3 days	May 29,2023	June 2,2023
Lập kế hoạch kiểm thử	1 day	June 2,2023	June 3,2023
Xem lại các tài liệu	2 days	June 3,2023	June 5,2023
Thiết kế các test case	1 day	June 5,2023	June 6,2023
Xem lại các testcase	2 days	June 6,2023	June 7,2023
Thực thi các testcase	1 day	June 7,2023	June 7,2023
Ghi nhận và đánh giá kết quả kiểm thử	1 day	June 8,2023	June 8,2023

Bảng 2: Lịch trình công việc

4. Những yêu cầu về tài nguyên

4.1 Phần cứng

- Máy tính cá nhân cần có kết nối mạng Internet

CPU	RAM	HDD	Architecture
Intel Core i5 / i7, 2.3Ghz	8GB	1TB	64 bit

Bảng 3: Phần cứng

4.2 Phần mềm

Tên phần mềm	Phiên bản	Loại
Chrome	111	Trình duyệt web

Bảng 4: Phần mềm

4.3 Công cụ kiểm thử

Hoạt động	Công cụ	Nhà cung cấp	Phiên bản
Quản lý test case	Microsoft Office Excel	Microsoft	2019
Quản lý Configuration	Microsoft Office Word	Microsoft	2019
Defect Tracking	Microsoft Office Excel	Microsoft	2019
Quản lý tiến độ kiểm thử	Microsoft Project Professional	Microsoft	2019

Bảng 5: Công cụ kiểm thử

4.4 Môi trường kiểm thử

Môi trường kiểm thử được sử dụng để đảm bảo rằng phần mềm được kiểm thử hoạt động chính xác trong môi trường thực tế, trước khi được triển khai cho người dùng cuối. Nó cũng giúp đảm bảo rằng các tính năng của phần mềm hoạt động đúng theo kỳ vọng và đáp ứng yêu cầu của khách hàng.

Một số thành phần chính của môi trường kiểm thử bao gồm:

Phần mềm kiểm thử: Bao gồm các công cụ và phần mềm được sử dụng để thực hiện kiểm thử, bao gồm các công cụ tự động hóa kiểm thử, các công cụ quản lý kiểm thử và các công cụ báo cáo kiểm thử.

Dữ liệu kiểm thử: Bao gồm các tập dữ liệu được sử dụng để thực hiện kiểm thử, bao gồm các tập dữ liệu kiểm thử cơ bản và các tập dữ liệu kiểm thử phức tạp hơn.

Cơ sở dữ liệu: Bao gồm các cơ sở dữ liệu được sử dụng để thực hiện kiểm thử, bao gồm cả cơ sở dữ liệu đơn giản và phức tạp.

Phần cứng: Bao gồm các thiết bị phần cứng được sử dụng để thực hiện kiểm thử, bao gồm cả máy tính và các thiết bị ngoại vi khác.

Mạng: Bao gồm các thiết bị mạng được sử dụng để thực hiện kiểm thử, bao gồm các máy chủ và các thiết bị kết nối mạng khác.

Các yêu cầu của môi trường kiểm thử cũng có thể khác nhau tùy thuộc vào phần mềm cụ thể được kiểm thử. Việc xác định các yêu cầu này là quan trọng để đảm bảo rằng môi trường kiểm thử được thiết lập đúng cách và đáp ứng các yêu cầu của kiểm

4.5. Nhân sự

4.5.1 Vai trò và trách nhiệm

Thành viên	Nhiệm vụ
Tổng Minh Khang	Xây dựng hệ thống Web result , Thực hiện các test case, test hệ thống với TestComplete
Hoàng Đình Thái	Xây dựng hệ thống Web default, Thực hiện các test case, test hệ thống với TestComplete
Hoàng Minh Thu	Xây dựng hệ thống chức năng đăng ký, Làm báo cáo, Vẽ use-case, mô tả chức năng hệ thống
Âu Thị Quý	Xây dựng hệ thống chức năng đăng ký, Làm báo cáo, Vẽ use-case, mô tả chức năng hệ thống
Nguyễn Thị Trang Nhung	Xây dựng hệ thống chức năng Log in, Làm báo cáo, Vẽ use-case, mô tả chức năng hệ thống

Bảng 6: Vai trò trách nhiệm

5. Phạm vi kiểm thử

5.1. Các yêu cầu chức năng

Các tác nhân:

Hệ thống thuế thu nhập cá nhân là một hệ thống phức tạp, bao gồm nhiều tác nhân và yếu tố ảnh hưởng:

- Chính phủ: Chính phủ là tác nhân chủ chốt trong hệ thống thuế thu nhập cá nhân. Chính phủ quyết định về các quy định thuế, bao gồm cả mức thuế, các khoản miễn thuế và các quy định về khai thuế và thu thuế.
- Người nộp thuế: Người lao động là tác nhân chịu trách nhiệm nộp thuế. Họ

phải đóng các khoản tiền thuế dựa trên thu nhập của mình.

- Nhà tuyển dụng: Nhà tuyển dụng là tác nhân có trách nhiệm khấu trừ thuế từ lương nhân viên và nộp cho chính phủ.

- Các cơ quan thuế: Các cơ quan thuế có trách nhiệm xác định các khoản thuế phải nộp của người lao động và đảm bảo việc khai thuế đúng và kịp thời.

- Các chuyên gia tài chính và luật sư: Các chuyên gia tài chính và luật sư cung cấp cho người nộp thuế thông tin và hỗ trợ trong việc hiểu và thực hiện các quy định thuế.

- Các tổ chức phi lợi nhuận và các nhà tài trợ: Các tổ chức phi lợi nhuận và các nhà tài trợ có thể đóng góp vào các quy định thuế và các chính sách tài chính để đảm bảo rằng hệ thống thuế thu nhập cá nhân là công bằng và hiệu quả.

Tất cả các tác nhân này đóng vai trò quan trọng trong việc xây dựng và duy trì hệ thống thuế thu nhập cá nhân.

Các chức năng của hệ thống

1. Đăng ký và cập nhật thông tin người đóng thuế: Hệ thống thuế thu nhập cá nhân cần có usecase để cho phép người đóng thuế đăng ký và cập nhật thông tin cá nhân của họ bao gồm: địa chỉ, SĐT, và thông tin thu nhập.

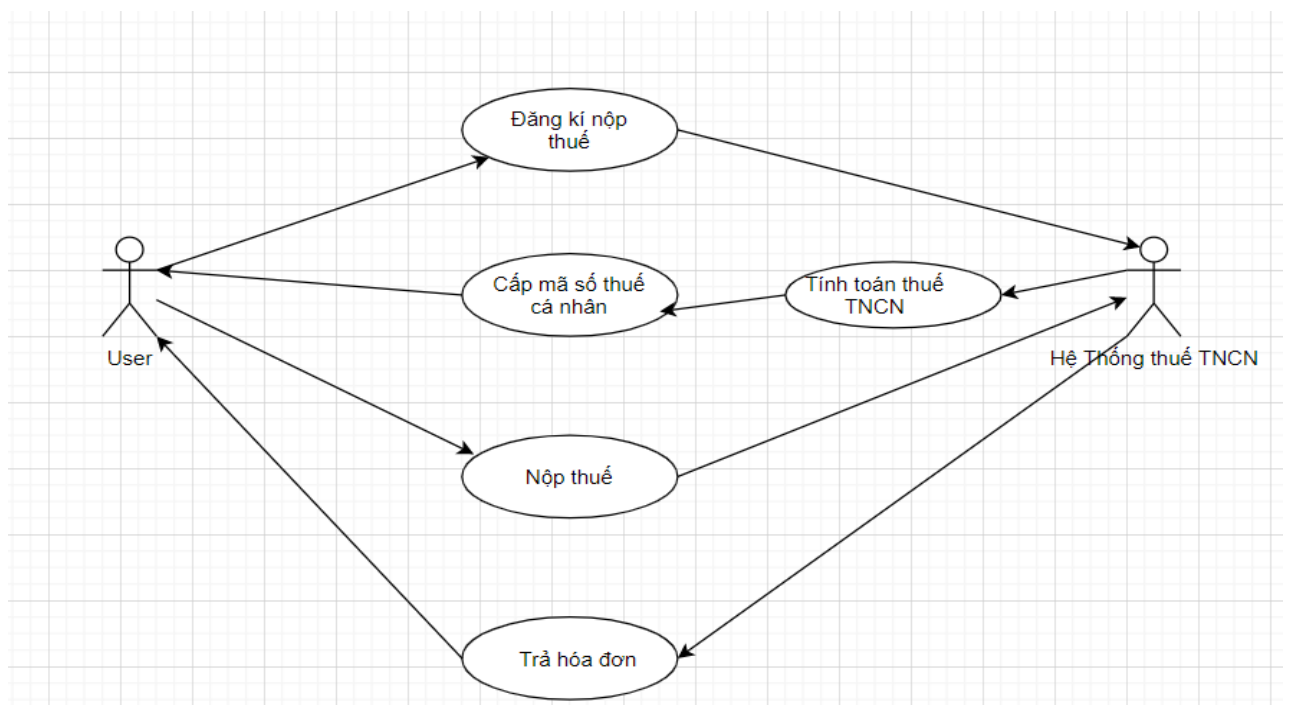
2. Tính toán và đăng ký thuế: hệ thống cần có usecase để tính toán số tiền thuế phải đóng dựa trên thông tin thu nhập của người đóng thuế và các quy định thuế hiện hành. Sau đó , hệ thống cần cho phép người đóng thuế đăng ký và nộp thuế.

3. Xử lý các nghiệp vụ liên quan đến thuế: hệ thống cần có usecase để xử lý các nghiệp vụ liên quan đến thuế bao gồm phát hành các báo cáo thuế, cập nhật thông tin thuế của người đóng thuế và giải quyết các thắc mắc của người đóng thuế.

4.Quản lý và bảo vệ thông tin thuế: hệ thống cần có usecase để quản lý và bảo vệ thông tin thuế của người đóng thuế, bao gồm lưu trữ và truy xuất thông tin thuế đảm bảo an toàn thông tin và bảo vệ quyền riêng tư của người đóng thuế.

5.Cung cấp các chương trình khuyến mãi thuế : hệ thống cần có usecase để cung cấp các chương trình khuyến mãi thuế, bao gồm các khoản miễn giảm để khuyến khích các hoạt động kinh doanh đầu tư

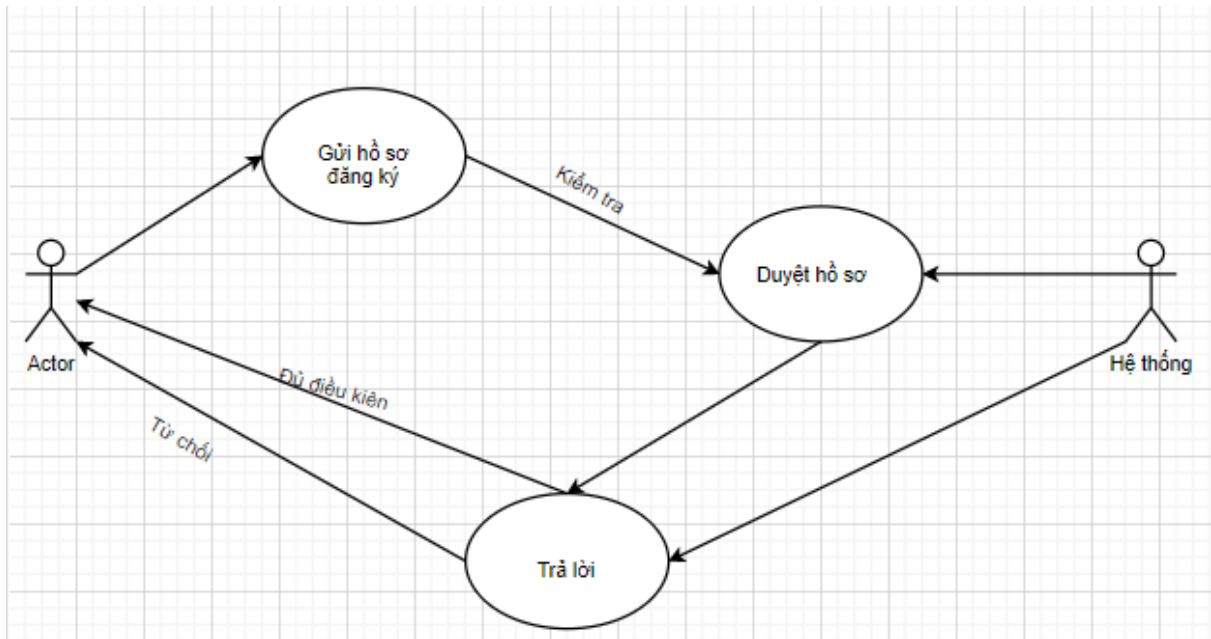
Biểu đồ use-case tổng quát



Hình 1.1: Biểu đồ use case tổng quan

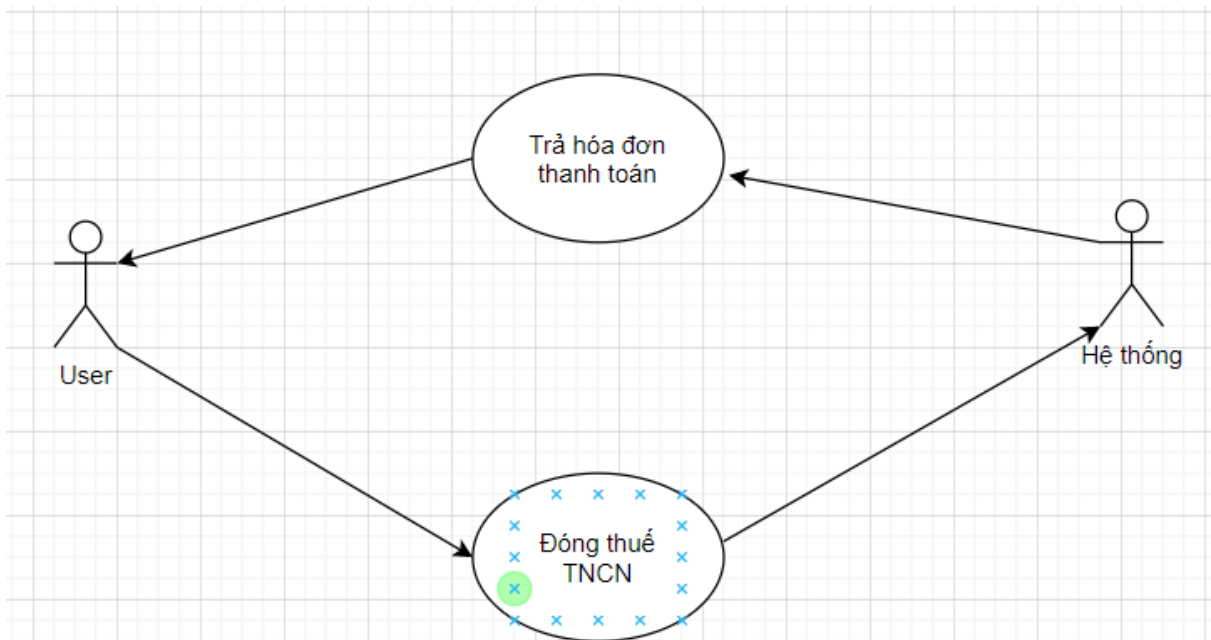
Biểu đồ use-case phân rã các chức năng:

1.1.1 Phân rã use-case cho cấp mã thuế TNCN



Hình 1.2: Biểu đồ cấp mã số thuế

1.1.2 Phân rã use-case cho đóng thuế



Hình 1.3 :biểu đồ chức năng đóng thuế

5.1 Những chức năng được kiểm thử

Usecase 1: Đăng ký tài khoản:

Mã Use case	UC001	Tên Use case	Đăng ký
Tác nhân	user		
Mô tả	Tác nhân đăng ký tài khoản để sử dụng chức năng của hệ thống		
Sự kiện kích hoạt	Click vào nút đăng ký trên thanh tiêu đề		
Tiền điều kiện	Không		
Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1	User	Chọn chức năng Đăng ký
	2	Hệ thống	Hiển thị giao diện đăng ký
	3	User	Nhập các thông tin tài khoản (mô tả phía dưới *)
	4	User	Yêu cầu đăng ký
	5	Hệ thống	Kiểm tra xem khách đã nhập các trường bắt buộc nhập hay chưa
	6	Hệ thống	Kiểm tra địa chỉ email của khách có hợp lệ không
	7	Hệ thống	Kiểm tra mật khẩu nhập lại và mật khẩu có trùng nhau hay không
	8	Hệ thống	Kiểm tra mật khẩu có đủ mức độ an toàn hay không
	9	Hệ thống	Lưu thông tin tài khoản và thông báo đăng ký thành công
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	6a.	Hệ thống	Thông báo lỗi: Cần nhập các trường bắt buộc nhập nếu khách nhập thiếu

	7a.	Hệ thống	Thông báo lỗi: Địa chỉ email không hợp lệ nếu địa chỉ email không hợp lệ
	8a.	Hệ thống	Thông báo lỗi: Mật khẩu xác nhận không trùng với Mật khẩu nếu hai mật khẩu không trùng nhau
	9a.	Hệ thống	Thông báo lỗi: Mật khẩu cần đảm bảo độ an toàn nếu mật khẩu không đảm bảo độ an toàn được quy định bởi hệ thống (có ít nhất 6 ký tự)
Hậu điều kiện		Tài khoản được tạo và lưu trữ vào hệ thống	

Bảng 7.1: Use Case Login

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1	Username	Input username field	Có	Username bao gồm cả chữ và số	abc123
2	Mật khẩu	Password field	Có	Ít nhất 6 ký tự	123456abc

Bảng 7.2: Mô tả data Login

Usecase 2: Đăng nhập hệ thống:

Mã Use case	UC002	Tên Use case	cấp mã thuế TNCN
Tác nhân	User		
Mô tả	Tác nhân đăng nhập vào hệ thống để sử dụng các chức năng hệ thống		
Sự kiện kích hoạt	Click vào nút log in trên giao diện website		
Tiền điều kiện	User đã có tài khoản trên hệ thống		

Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1	User	Chọn chức năng Đăng nhập
	2	Hệ thống	Hiển thị giao diện đăng nhập
	3	User	Nhập email và mật khẩu (mô tả phía dưới *)
	4	User	Yêu cầu đăng nhập
	5	Hệ thống	Kiểm tra xem khách đã nhập các trường bắt buộc nhập hay chưa
	6	Hệ thống	Kiểm tra email và mật khẩu có hợp lệ do người dùng nhập trong hệ thống hay không
	7	Hệ thống	Hiển thị chức năng tương ứng đối với Người dùng
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	6a.	Hệ thống	Thông báo lỗi: Cần nhập các trường bắt buộc nhập nếu khách nhập thiếu
	7a.	Hệ thống	Thông báo lỗi: Email và/hoặc mật khẩu chưa đúng nếu không tìm thấy email và mật khẩu trong hệ thống
Hậu điều kiện	Tác nhân đăng nhập được vào hệ thống		

Bảng 8.1: Use Case Register

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1	Username	Input username field	Có	Username bao gồm cả chữ và số	abc123

				Không trùng với username đã tồn tại	
2	Password	Password field	Có	Ít nhất 8 ký tự	123456abc
3	Fullname	Input fullname field	Có	Fullname bao gồm chuỗi tối đa 255 ký tự	Abc
4	Address	Input address field	Có	Address bao gồm chuỗi tối đa 255 ký tự	
5	Birthday	Select datetime	Có		
6	Gender	Select radio button	Có		
7	Email	Input email field	Có	Địa chỉ email hợp lệ	abc@gmail. com
8	Phone Number	Input phonenumber field	Có	Gồm chuỗi đủ 10 số bắt đầu bằng số 0	037****16 2

Bảng 8.2: Mô phỏng data Register

Usecase 3: Tính thuế:

Mã Use case	UC003	Tên Use case	Tính thuế TNCN
Tác nhân	User		
Mô tả	Tác nhân tính thuế TNCN		
Sự kiện kích hoạt	Click vào nút xác nhận		

Tiền điều kiện	User đã có tài khoản trên hệ thống		
Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1	User	Chọn chức năng Tính thuế TNCN
	2	Hệ thống	Hiển thị giao diện Tính thuế
	3	User	Nhập các thông tin cá nhân
	4	User	Nhập các mức thu nhập cá nhân theo từng tháng
	5	User	Xác nhận yêu cầu tính thuế TNCN
	6	Hệ thống	Kiểm tra xem người dùng đã nhập các trường bắt buộc nhập hay chưa
	7	Hệ thống	Kiểm tra thông tin người dùng nhập có chính xác hay chưa
	8	Hệ thống	Kiểm tra số liệu nhập có hợp lệ không
	9	Hệ thống	Tiến hành tính thuế TNCN và hiển thị thông tin ra màn hình
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	6a.	Hệ thống	Thông báo lỗi: Cần nhập các trường bắt buộc nhập nếu khách nhập thiếu
	7a.	Hệ thống	Thông báo lỗi: Nhập sai thông tin cá nhân nếu thông tin không khớp với thông tin người dùng đăng ký
	8a.	Hệ thống	Thông báo lỗi: Nếu số nhập vào không hợp lệ
	9a.	Hệ thống	Thông báo lỗi: Hệ thống gặp sự cố kỹ thuật hoặc lỗi, hệ thống sẽ hiển thị thông báo lỗi và hướng dẫn người dùng thử lại sau một thời gian
Hậu điều kiện	Hiển thị thông tin cá nhân và bảng tính thuế thu nhập cá nhân		

Bảng 9.1: Use Case tính thuế

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1	Năm	Input email field	Có	4 ký tự số	2002
2	Họ và tên	Input text field	Có	Chuỗi ký tự	Nguyễn Văn A
3	Bảo hiểm xã hội	Select box	Có	Có/Không	Có
4	Mức miễn trừ với người lao động	Default	Không	11	
5	Mức miễn trừ với người phụ thuộc	Default	Không	4.4	
6	Số người phụ thuộc	Input text field	Không	chuỗi tối đa 255 ký tự số	1
7	Lương theo các tháng	Input text field	Không	chuỗi tối đa 255 ký tự số	30

Bảng 9.2: Mô phỏng data tính thuế

5.2 Những chức năng chưa được kiểm thử

Usecase 4 : Cấp mã thuế:

Mã Use case	UC004	Tên Use case	Cấp mã thuế TNCN
Tác nhân	User		
Mô tả	Tác nhân cấp mã thuế TNCN		
Sự kiện kích hoạt	Click vào nút trên thanh tiêu đề		

Tiền điều kiện	User đã có tài khoản trên hệ thống		
Luồng sự kiện chính (Thành công)	STT	Thực hiện bởi	Hành động
	1	User	Chọn chức năng Cấp mã thuế
	2	Hệ thống	Hiển thị giao diện Cấp mã thuế
	3	User	Người dùng nhập thông tin cá nhân và thông tin liên quan để yêu cầu cấp mã thuế.
	4	Hệ thống	Hệ thống kiểm tra tính hợp lệ của thông tin và tiến hành quy trình cấp mã thuế.
	5	Hệ thống	Hệ thống tạo và hiển thị mã thuế mới cho người dùng.
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	4a.	Hệ thống	Thông báo lỗi: Nếu người dùng nhập thông tin không hợp lệ, ví dụ như thông tin cá nhân không đúng, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập lại thông tin đúng.
	5a.	Hệ thống	Thông báo lỗi: Nếu hệ thống gặp sự cố kỹ thuật hoặc lỗi trong quá trình cấp mã thuế, hệ thống sẽ hiển thị thông báo lỗi và hướng dẫn người dùng thử lại sau một thời gian
	5b.	Hệ thống	Thông báo lỗi: Nếu người dùng đã yêu cầu cấp mã thuế trước đó và mã thuế đã được cấp, hệ thống sẽ hiển thị thông báo rằng người dùng đã có mã thuế và không thể cấp thêm mã thuế mới.
Hậu điều kiện	Người dùng nhận được mã thuế và sử dụng cho các mục đích liên quan.		

Bảng 10.1: Use Case Cấp mã thuế

S T T	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1	Email	Input email field	Có	Địa chỉ email hợp lệ	thu@gmail.com
2	Họ và tên	Input text field	Có	Chuỗi ký tự	Nguyễn Văn A
3	Ngày sinh	Select box	Có	Male/Female	Male
4	Các thông tin liên quan đến thuế	Input text field	Có	Các thông tin cần thiết để xác định loại thuế, địa phương, ngành nghề hoặc các thông tin khác liên quan đến quá trình cấp mã thuế	thuế VAT
5	Mã xác nhận/Captcha	Input text field	Có	Yêu cầu 6 ký tự	123456

Bảng 10.2: Mô phỏng data Cấp mã thuế

Usecase 5: Đóng thuế:

Mã Use case	UC005	Tên Use case	Đóng thuế TNCN
Tác nhân	User		
Mô tả	Tác nhân đóng thuế TNCN		
Sự kiện kích hoạt	Click vào nút đóng thuế		
Tiền điều kiện	User đã có tài khoản trên hệ thống		
Luồng sự kiện	STT	Thực hiện bởi	Hành động

chính (Thành công)	1	User	Chọn chức năng đóng thuế TNCN
	2	Hệ thống	Hiển thị giao diện đóng thuế
	3	User	Nhập các thông tin cá nhân
	4	User	Nhập các thông tin liên quan đến thuế, chẳng hạn như mã số thuế, kỳ đóng thuế, số tiền cần đóng.
	5	User	Xác nhận yêu cầu đóng thuế TNCN
	6	Hệ thống	Hệ thống kiểm tra tính hợp lệ của thông tin và tiến hành quy trình đóng thuế.
	7	Hệ thống	Hệ thống xác nhận thành công việc đóng thuế và cung cấp thông báo cho người dùng.
Luồng sự kiện thay thế	STT	Thực hiện bởi	Hành động
	6a.	Hệ thống	Thông báo lỗi: Nếu người dùng nhập thông tin không hợp lệ, ví dụ như mã số thuế không chính xác hoặc kỳ đóng thuế không hợp lệ, hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng nhập lại thông tin đúng.
	6b.	Hệ thống	Thông báo lỗi: Nếu người dùng không nhập đủ thông tin cần thiết, hệ thống sẽ hiển thị thông báo lỗi yêu cầu người dùng cung cấp đầy đủ thông tin để đóng thuế.
	7a.	Hệ thống	Thông báo lỗi: Nếu hệ thống gặp sự cố kỹ thuật hoặc lỗi trong quá trình đóng thuế, hệ thống sẽ hiển thị thông báo lỗi và hướng dẫn người dùng thử lại sau một thời gian
	7b.	Hệ thống	Thông báo lỗi: Hệ thống gặp sự cố kỹ thuật hoặc lỗi, hệ thống sẽ hiển thị thông báo lỗi và hướng dẫn người dùng thử lại sau một thời gian

Hậu điều kiện	Xác nhận người dùng đã đóng thuế thành công.
---------------	--

Bảng 11.1: Use Case đóng thuế

STT	Trường dữ liệu	Mô tả	Bắt buộc?	Điều kiện hợp lệ	Ví dụ
1	Mã số thuế	Input text field	Có	6 ký tự	abc123
2	Họ và tên	Input text field	Có	Chuỗi ký tự	Nguyễn Văn A
3	Kỳ đóng thuế	Input text field	Có	Chuỗi ký tự	6/2023
4	Số tiền	Input text field	Có	Chuỗi ký tự	100000
5	Loại thuế	Input text field	Có	Chuỗi ký tự	VAT

Bảng 11.2: Mô phỏng data đóng thuế

6. Chiến lược kiểm thử

6.1 Test Stages (Các cấp độ kiểm thử)

Test Stages (các giai đoạn kiểm thử) là các pha hoặc giai đoạn khác nhau trong quá trình thực hiện kiểm thử để đảm bảo tính chính xác, hoạt động và hiệu suất của một hệ thống hoặc ứng dụng. Các giai đoạn này thường được thực hiện theo một trình tự nhất định và có mục tiêu cụ thể để kiểm tra các khía cạnh khác nhau của hệ thống.

Các giai đoạn kiểm thử phổ biến bao gồm:

- Unit Testing (Kiểm thử đơn vị): Giai đoạn này tập trung vào việc kiểm tra các phần tử nhỏ nhất trong hệ thống như hàm, module hoặc các thành phần đơn lẻ. Mục tiêu là đảm bảo tính chính xác và hoạt động của các phần tử này độc lập.

- **Integration Testing (Kiểm thử tích hợp):** Giai đoạn này kiểm tra các thành phần đã được đơn vị kiểm thử kết hợp với nhau để đảm bảo tính chính xác và tương tác giữa chúng. Mục tiêu là xác minh tích hợp chính xác của các thành phần.
- **System Testing (Kiểm thử hệ thống):** Giai đoạn này tập trung vào việc kiểm tra hệ thống hoàn chỉnh với mục tiêu kiểm tra tính chính xác, hoạt động và hiệu suất của hệ thống theo các yêu cầu và kịch bản kiểm thử đã được xác định.
- **Acceptance Testing (Kiểm thử chấp nhận):** Giai đoạn này được thực hiện để kiểm tra xem hệ thống đã đáp ứng được yêu cầu của người dùng hay không. Mục tiêu là đảm bảo tính chính xác, hoạt động và hiệu suất của hệ thống được chấp nhận bởi người dùng cuối.

Mỗi giai đoạn kiểm thử có mục tiêu và phạm vi riêng, và chúng thường được thực hiện tuần tự để đảm bảo việc kiểm tra toàn diện và liên tục của hệ thống. Tuy nhiên, phụ thuộc vào quy trình phát triển và yêu cầu cụ thể của dự án, các giai đoạn kiểm thử có thể thay đổi hoặc điều chỉnh.

LEVELS OF TESTING		
1	Unit Testing	Done by Developers
2	Integration Testing	Done by Testers
3	System Testing	Done by Testers
4	Acceptance Testing	Done by End Users

Hình 2. Các cấp độ kiểm thử phần mềm

6.2 Các loại kiểm thử

Kiểm thử phần mềm là gì?

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

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

Kiểm thử phần mềm đóng vai trò rất quan trọng :

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

Mục đích của kiểm thử phần mềm:

- Kiểm thử phần mềm để đánh giá phần mềm có đạt yêu cầu mong đợi hay có sai sót nào không?
- Phần mềm có làm việc như mong muốn không?
- Phần mềm có giải quyết được yêu cầu của khách hàng không? Nó làm được gì mà người dùng mong đợi?
- Người dùng có thích nó không?
- Nó có tương thích với các hệ thống khác của chúng ta hay không?

1. Phân loại kỹ thuật kiểm thử

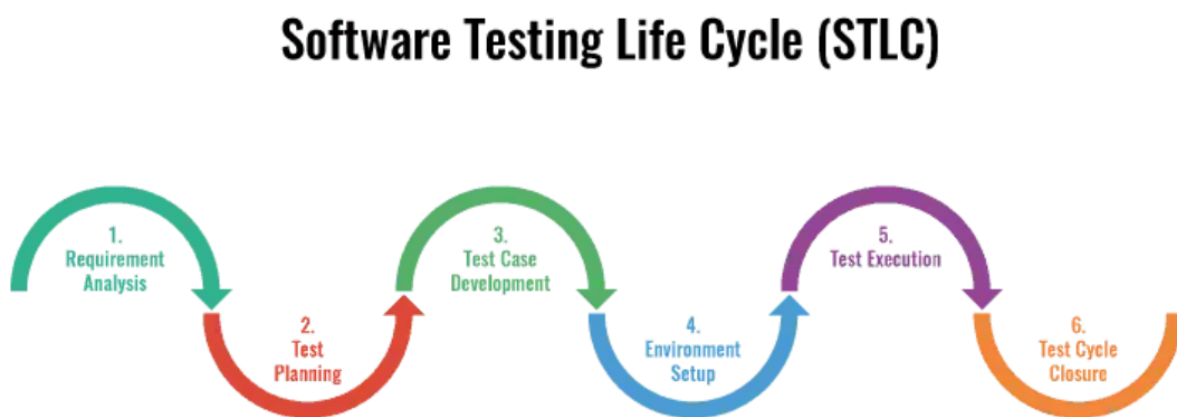
Ta phân loại kiểm thử dựa vào yếu tố: Chiến lược kiểm thử, phương pháp kiểm thử và kỹ thuật kiểm thử.

- Dựa vào chiến lược kiểm thử ta có thể phân chia kiểm thử thành 2 loại: kiểm thử thủ công và kiểm thử tự động
- Theo phương pháp tiến hành kiểm thử ta chia kiểm thử thành 2 loại: Kiểm thử tĩnh và kiểm thử động.
- Dựa vào kỹ thuật kiểm thử ta có thể phân chia kiểm thử thành 3 loại:

- Kiểm thử hộp đen, kiểm thử hộp trắng, kiểm thử hộp xám

2. Quy trình kiểm thử phần mềm

Quy trình kiểm thử bao gồm những giai đoạn sau:



Hình 3. Quy trình kiểm thử phần mềm

a. Requirement analysis - Phân tích yêu cầu

-Đầu vào:

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

-Hoạt động:

- Phân tích yêu cầu là giai đoạn đầu tiên trong quy trình kiểm thử phần mềm.
- QA team sẽ thực hiện đọc hiểu, nghiên cứu và phân tích cụ thể các yêu cầu trong tài liệu đặc tả của dự án hoặc tài liệu khách hàng. Qua hoạt động này, QA team sẽ nắm bắt được các yêu cầu mà dự án đưa ra bao gồm yêu cầu kiểm thử chức năng/ phi chức năng nào.

- Ngoài ra, trong quá trình phân tích, nghiên cứu tài liệu, nếu có câu hỏi phát sinh hay đề xuất giải quyết, QA team sẽ đưa ra câu hỏi với các bên liên quan như BA(Business Analysis), PM(Project Manager), team leader, khách hàng để hiểu chính xác hơn về yêu cầu của sản phẩm. Những câu hỏi này sẽ được lưu trữ vào file Q&A(Question and Answer). Các câu hỏi nên được đưa ra dưới dạng Yes/No question hoặc các lựa chọn để tiết kiệm thời gian trả lời cũng như hỗ trợ đưa ra những gợi ý hay để xây dựng sản phẩm ngay từ đầu. Như vậy, đương nhiên là chúng ta không nên nêu ra những câu hỏi dạng là gì, như thế nào, tại sao,... Những câu hỏi như thế thường mất thời gian để giải thích và cũng khó có thể giải thích một cách chi tiết nhất có thể. Hơn nữa, đối với khách hàng không có sự hiểu biết về lĩnh vực phần mềm mà họ yêu cầu thì càng không thể trả lời những câu hỏi mang tính chuyên môn cao. Chính chúng ta sẽ là người hỗ trợ và đưa ra giải pháp thích hợp cho khách hàng lựa chọn.

-Đầu ra:

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

b. Test planning - Lập kế hoạch kiểm thử

Đầu vào

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

Hoạt động

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

- Xác định phạm vi(Scope) dự án: Dự án thực hiện trong thời gian bao lâu? Bao gồm những công việc gì cho từng khoảng thời gian xác định? Từ đó

đưa ra lịch trình thực hiện cho từng công việc nhỏ sao cho phù hợp với toàn bộ đội dự án.

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

Đầu ra

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

c. Test case development - Thiết kế kịch bản kiểm thử

Đầu vào

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

Hoạt động

- Review tài liệu: Đầu tiên, các kiểm thử viên cần review lại tất cả các tài liệu để xác định công việc cần làm, các công việc có khác gì so với dự án trước khách hàng đưa cho, chức năng nào cần test, chức năng nào không cần test lại nữa. Từ đó, vừa có thể tiết kiệm thời gian mà vẫn đưa ra được một kịch bản kiểm thử đầy đủ và hiệu quả.
- Viết test case/ checklist: Sau đó, tester bắt tay vào việc viết test case chi tiết dựa vào kế hoạch đã đưa ra và vận dụng các kỹ thuật thiết kế kịch bản kiểm thử. Test case cần bao phủ được tất cả các trường hợp kiểm thử có thể xảy ra cũng như đáp ứng đầy đủ các tiêu chí của sản phẩm. Đồng thời tester cũng cần đánh giá mức độ ưu tiên cho từng test case.
- Chuẩn bị dữ liệu kiểm thử: Cùng với việc tạo ra các test case chi tiết, đội kiểm thử cũng cần chuẩn bị trước các dữ liệu kiểm thử cho các trường hợp cần thiết như test data, test script.
- Review test case/ checklist: Sau khi hoàn thành, các thành viên trong đội kiểm thử hoặc test leader cũng cần review lại test case đã tạo để có thể bổ sung, hỗ trợ lẫn nhau nhằm tránh những sai sót trong thiết kế test case và rủi ro về sau.

Đầu ra

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

d. Test environment setup - Thiết lập môi trường kiểm thử

Đầu vào

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

Hoạt động

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

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

Đầu ra

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

e. Test execution - Thực hiện kiểm thử

Đầu vào

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

Hoạt động

- Thực hiện các test case như thiết kế và mức độ ưu tiên đã đưa ra trên môi trường đã được cài đặt.
- So sánh với kết quả mong đợi sau báo cáo các bug xảy ra lên tool quản lý lỗi và theo dõi trạng thái của lỗi đến khi được sửa thành công.
- Thực hiện retest để verify các bug đã được fix và regression test khi có sự thay đổi liên quan.
- Trong quá trình thực hiện kiểm thử, kiểm thử viên cũng có thể hỗ trợ, đề xuất cho cả đội dự án để có giải pháp hợp lý và kết hợp công việc hiệu quả.
- Đo và phân tích tiến độ: kiểm thử viên cũng cần kiểm soát chặt chẽ tiến độ công việc của mình bằng cách so sánh tiến độ thực tế với kế hoạch, nếu chậm cần phải điều chỉnh sao cho kịp tiến độ dự án, nếu nhanh cũng cần điều chỉnh vì có thể test lead lên kế hoạch chưa sát với thực tế dự án. Từ đó có thể sửa chữa test plan cần điều chỉnh để phù hợp với tiến độ dự án đưa ra.
- Report thường xuyên cho PM và khách hàng về tình hình thực hiện dự án: Cung cấp thông tin trong quá trình kiểm thử đã làm được những chức năng nào, còn chức năng nào, hoàn thành được bao nhiêu phần trăm công

việc, báo cáo các trường hợp phát sinh sớm, tránh ảnh hưởng tiến độ công việc của cả ngày.

Đầu ra

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

f. Test cycle closure - Đóng chu trình kiểm thử

Đầu vào

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

Hoạt động

- Đây là giai đoạn cuối cùng trong quy trình kiểm thử phần mềm.
- Ở giai đoạn này, QA team thực hiện tổng kết, báo cáo kết quả về việc thực thi test case, bao nhiêu case pass/ fail, bao nhiêu case đã được fix, mức độ nghiêm trọng của lỗi, bao nhiêu lỗi cao/ thấp, lỗi còn nhiều ở chức năng nào, dev nào nhiều lỗi. Chức năng nào đã hoàn thành test/ chưa hoàn thành test/ trễ tiến độ bàn giao.
- Đánh giá các tiêu chí hoàn thành như phạm vi kiểm tra, chất lượng, chi phí, thời gian, mục tiêu kinh doanh quan trọng.
- Ngoài ra, giai đoạn này cũng thảo luận tất cả những điểm tốt, điểm chưa tốt và rút ra bài học kinh nghiệm cho những dự án sau, giúp cải thiện quy trình kiểm thử.

Đầu ra

Đầu ra của giai đoạn này bao gồm các tài liệu: Test report, Test results(final)

Quy trình	Đầu vào	Các hoạt động	Đầu ra
Phân tích yêu cầu	Tài liệu SRS, tài liệu thiết kế, bản prototype	<p>Đọc hiểu, nghiên cứu phân tích các yêu cầu có trong các bản tài liệu</p> <p>Đưa ra các câu hỏi còn thắc mắc về yêu cầu phần mềm với BA, team, leader, khách hàng để hiểu rõ hơn về yêu cầu sản phẩm</p>	File Q & A
Lập kế hoạch	Các tài liệu đã được cập nhật thông qua file Q & A trong giai đoạn phân tích yêu cầu	<p>Xác định phạm vi kiểm thử: thời gian, lịch trình cho các công việc.</p> <p>Xác định phương pháp tiếp cận.</p> <p>Xác định nguồn lực: con người và thiết bị.</p> <p>Lên kế hoạch thiết kế công việc test: các chức năng cần kiểm thử, cái nào cần thực</p>	Test plan, checklist

		hiện trước, sau, ai là người thực hiện...	
Thiết kế kiểm thử	Test plan, checklist và các tài liệu đặc tả đã được cập nhật	<p>Review tài liệu: xác định công việc cần làm.</p> <p>Viết test case/checklist.</p> <p>Chuẩn bị dữ liệu kiểm thử: test data, test script.</p> <p>Review test case/checklist: tránh rủi ro trong thiết kế test case</p>	Test design, test case, checklist, test data, test automation script
Chuẩn bị môi trường	Test plan, smoke test case, test data	Thực thi các smoke test case để kiểm tra môi trường kiểm thử đã sẵn sàng cho việc test chưa	Môi trường đã được chuẩn bị sẵn sàng cho việc test và các kết quả của smoke test case
Thực hiện kiểm thử	Test design, test case, checklist, test data, test automation script	<p>Thực hiện test theo kịch bản kiểm thử.</p> <p>So sánh kết quả thực tế với mong đợi và log bug lên tool quản lý lỗi, theo dõi quá trình xử lý lỗi.</p>	Test results, defect reports

Kết thúc	Tất cả các tài liệu được tổng hợp từ giai đoạn đầu tiên	Tổng kết báo cáo kết quả về việc thực thi test, chức năng nào hoàn thành/ chức năng chưa hoàn thành, lỗi còn nhiều ở chức năng nào, dev nào còn nhiều lỗi, lỗi có nghiêm trọng hay không...	Test report, test results final
----------	---	---	---------------------------------

Bảng 12: Test report

7. Điều kiện chấp nhận

Điều kiện chấp nhận (Acceptance Criteria) là các tiêu chí cụ thể và rõ ràng mà hệ thống hoặc ứng dụng cần đáp ứng để được chấp nhận là hoàn thành và đáp ứng yêu cầu của khách hàng hoặc người sử dụng cuối.

Các điều kiện chấp nhận thường được xác định trong quá trình lập kế hoạch và phát triển dự án, và chúng có vai trò quan trọng trong việc xác định phạm vi và tiêu chí thành công của dự án. Các điều kiện chấp nhận cung cấp một phương châm rõ ràng để kiểm tra và đánh giá xem hệ thống hoặc ứng dụng đã đáp ứng các yêu cầu và mong đợi của người sử dụng.

Một số ví dụ về điều kiện chấp nhận có thể bao gồm:

Đúng kết quả: Hệ thống hoặc ứng dụng phải đưa ra kết quả chính xác theo yêu cầu và logic kinh doanh đã xác định.

Tính ổn định: Hệ thống hoặc ứng dụng cần đảm bảo tính ổn định và không gây ra lỗi hoặc sự cố không mong muốn.

Hiệu suất: Hệ thống hoặc ứng dụng phải đáp ứng các yêu cầu về hiệu suất, bao gồm thời gian phản hồi, thời gian xử lý và khả năng xử lý số lượng lớn người dùng.

Tương thích: Hệ thống hoặc ứng dụng cần tương thích với các môi trường, nền tảng và trình duyệt khác nhau mà người sử dụng có thể sử dụng.

Giao diện người dùng: Giao diện người dùng cần thân thiện, dễ sử dụng và cung cấp trải nghiệm tốt cho người dùng.

Bảo mật: Hệ thống hoặc ứng dụng cần đảm bảo tính bảo mật của dữ liệu và thông tin cá nhân của người dùng.

Các điều kiện chấp nhận phải được xác định một cách rõ ràng, cụ thể và có thể đo lường để có thể kiểm tra và đánh giá một cách khách quan xem hệ thống đã đáp ứng được hay chưa. Chúng thường được sử dụng như một phương tiện để thống nhất và tạo sự hiểu biết chung giữa các bên liên quan, bao gồm cả khách hàng, nhóm phát triển và nhóm kiểm thử. Điều kiện chấp nhận cũng cung cấp một tiêu chuẩn đánh giá để xác định khi nào hệ thống hoặc ứng dụng được coi là đáp ứng yêu cầu và có thể được chấp nhận.

Các điều kiện chấp nhận thường được mô tả dưới dạng các câu mô tả rõ ràng và đo lường được. Ví dụ: "Hệ thống phải xử lý 100 giao dịch trong vòng 5 giây" hoặc "Giao diện người dùng phải hỗ trợ đa ngôn ngữ và có thiết kế đáp ứng".

Trong quá trình kiểm thử và đánh giá, các điều kiện chấp nhận được sử dụng để so sánh kết quả thực tế với kỳ vọng và xác định xem hệ thống hoặc ứng dụng đã đáp ứng được hay chưa. Nếu các điều kiện chấp nhận không được đạt được, quá trình kiểm thử và phát triển có thể cần điều chỉnh hoặc bổ sung để đảm bảo đáp ứng yêu cầu của khách hàng và người sử dụng cuối.

Quá trình xác định và quản lý điều kiện chấp nhận đóng vai trò quan trọng trong việc đảm bảo rằng hệ thống hoặc ứng dụng được chấp nhận và đáp ứng yêu cầu của khách hàng.

8. Defect Tracking

8.1 Phân loại lỗi

Lỗi tính toán thuế: Các lỗi này xảy ra khi tính toán thuế trên trang web không chính xác. Ví dụ, việc áp dụng sai thuế, tính sai tổng thuế, hoặc sử dụng công thức tính toán không đúng.

Lỗi hiển thị: Đây là các lỗi liên quan đến cách thông tin thuế được hiển thị trên trang web. Ví dụ, thông tin thuế không được hiển thị đầy đủ, thông tin sai hoặc mâu thuẫn, hoặc giao diện không rõ ràng về cách tính thuế.

Lỗi xử lý dữ liệu: Các lỗi này xảy ra khi dữ liệu đầu vào không được xử lý chính xác. Ví dụ, nhập sai thông tin liên quan đến thuế, không kiểm tra dữ liệu đầu vào hoặc không xử lý các tình huống đặc biệt như miễn thuế, giảm thuế, hay thuế đặc biệt.

Lỗi tính năng: Đây là các lỗi liên quan đến tính năng của trang web tính thuế. Ví dụ, không hỗ trợ các loại thuế cụ thể, không cung cấp tính năng tính toán tự động, hoặc không thể xử lý các trường hợp phức tạp như thuế đa quốc gia hay thuế tùy chỉnh.

Lỗi bảo mật: Đây là các lỗi liên quan đến an ninh thông tin và bảo mật dữ liệu trên trang web. Ví dụ, thiếu các biện pháp bảo mật như mã hóa dữ liệu, kiểm tra xác thực người dùng hoặc bảo vệ khỏi các cuộc tấn công như tấn công từ chối dịch vụ (DDoS) hay xâm nhập.

8.2 Quy trình xử lý lỗi

Ghi nhận lỗi: Ghi lại chi tiết về lỗi bao gồm mô tả, bước tái tạo lỗi, thông tin môi trường (trình duyệt, hệ điều hành, phiên bản), và bất kỳ thông tin nào hữu ích khác.

Ưu tiên và phân loại lỗi: Xác định mức độ ưu tiên của lỗi dựa trên tác động và sự quan trọng của nó. Phân loại lỗi thành các danh mục như lỗi nghiêm trọng, lỗi ưu tiên cao, lỗi trung bình và lỗi ưu tiên thấp.

Ghi lại quá trình tái tạo lỗi: Ghi lại lại các bước tái tạo lỗi, bao gồm dữ liệu đầu vào, tình huống và bất kỳ điều nào liên quan đến việc tái tạo lỗi một cách chính xác. Điều này giúp cho nhóm phát triển hoặc người quản lý hiểu rõ về vấn đề và cách để tái tạo nó.

Đánh giá và xác định nguyên nhân: Phân tích lỗi để xác định nguyên nhân gây ra nó. Điều này có thể bao gồm kiểm tra mã nguồn, xem xét các yếu tố xung quanh, và tìm hiểu về quy trình xử lý dữ liệu và tính toán thuế.

Xử lý và sửa lỗi: Dựa trên nguyên nhân đã xác định, phát triển và triển khai các biện pháp sửa lỗi. Điều này có thể bao gồm việc thay đổi mã nguồn, cập nhật tính năng, sửa lại công thức tính toán, hoặc cải thiện các biện pháp bảo mật.

Kiểm tra và xác minh: Sau khi sửa lỗi, tiến hành kiểm tra lại để đảm bảo rằng lỗi đã được khắc phục một cách thành công. Kiểm tra tính toàn vẹn của các tính năng, tính đúng đắn của tính toán thuế, và bảo mật dữ liệu.

Ghi lại kết quả và báo cáo: Ghi lại kết quả của quá trình sửa lỗi và tạo báo cáo. Báo cáo này cung cấp thông tin về lỗi đã xảy ra, nguyên nhân và biện pháp đã thực hiện để sửa chữa. Nó có thể được chia sẻ với nhóm phát triển, quản lý và bất kỳ bên liên quan nào khác để đảm bảo việc sửa lỗi được thực hiện đầy đủ và hiệu quả.

Quy trình xử lý lỗi có thể được tùy chỉnh phù hợp với yêu cầu và quy trình phát triển của mỗi dự án. Điều quan trọng là có một quy trình xử lý lỗi rõ ràng và hiệu quả để đảm bảo rằng các lỗi được phát hiện được giải quyết một cách nhanh chóng và chính xác.

9. Test Deliverables

9.1 Test Cases

Test Cases (các trường hợp kiểm thử) là các kịch bản hoặc tình huống cụ thể được thiết kế để kiểm tra tính chính xác, hoạt động và hiệu suất của một hệ thống hoặc ứng dụng. Mỗi test case tập trung vào một khía cạnh cụ thể của hệ thống và định nghĩa các đầu vào, bước thực hiện, kết quả mong đợi và tiên điều kiện.

Các test case thường được xác định dựa trên yêu cầu, kịch bản sử dụng và các trường hợp sử dụng tích cực và tiêu cực của hệ thống. Mục tiêu của các test case là tìm ra lỗi, xác minh tính năng và đảm bảo tính đáng tin cậy của hệ thống.

Một test case bao gồm các phần sau:

- Tiêu đề: Đặt tên cho test case để mô tả mục tiêu kiểm thử.
- Mô tả: Mô tả ngắn gọn về trường hợp kiểm thử và mục tiêu của nó.

- Đầu vào: Xác định các giá trị đầu vào cần thiết để thực hiện test case, bao gồm các tập tin, dữ liệu hoặc thao tác cần thiết.
- Bước thực hiện: Liệt kê các bước cụ thể để thực hiện test case, bao gồm các hành động, thao tác hoặc sự tương tác với hệ thống.
- Kết quả mong đợi: Xác định kết quả mong đợi khi test case được thực hiện một cách chính xác, bao gồm các giá trị đầu ra, trạng thái hoặc hành vi dự kiến của hệ thống.
- Tiền điều kiện: Xác định các điều kiện hoặc trạng thái trước khi thực hiện test case, chẳng hạn như cài đặt, dữ liệu sẵn có hoặc trạng thái của hệ thống.
- Kết quả: Ghi lại kết quả thực tế khi test case được thực hiện, bao gồm các giá trị thực tế của đầu ra, trạng thái hoặc hành vi của hệ thống.
- Kết luận: Đưa ra nhận xét hoặc đánh giá về kết quả của test case, bao gồm việc xác định lỗi (nếu có) và đánh giá tính chính xác của hệ thống.

Test cases được thiết kế để tăng tính toàn vẹn và đáng tin cậy của hệ thống. Chúng cung cấp một cách cụ thể để kiểm tra các tính năng và chức năng của hệ thống, xác định các lỗi, sự cố và các vấn đề khác mà người dùng có thể gặp phải.

Khi xác định và thiết kế các test case, quan trọng để đảm bảo rằng chúng:

- Đầy đủ: Đảm bảo rằng các trường hợp kiểm thử được thiết kế để kiểm tra tất cả các khía cạnh quan trọng của hệ thống và các yêu cầu của khách hàng.
- Rõ ràng và cụ thể: Các test case cần được mô tả một cách rõ ràng và chi tiết, đảm bảo rằng những ai thực hiện kiểm thử có thể hiểu và thực hiện chúng một cách chính xác.
- Tương tự và độc lập: Các test case nên được thiết kế để xác minh các tính năng và chức năng riêng biệt của hệ thống một cách độc lập và không phụ thuộc vào nhau.
- Có kết quả mong đợi rõ ràng: Mỗi test case cần định nghĩa rõ ràng các kết quả mong đợi, giúp người thực hiện kiểm thử xác định xem hệ thống hoạt động đúng hay sai.
- Đa dạng: Đảm bảo rằng các test case kiểm tra các trường hợp bình thường, giới hạn và biên, cũng như các tình huống không mong đợi và lỗi có thể xảy ra.

- Dễ dàng thực hiện và lặp lại: Các test case nên được thiết kế để thực hiện một cách dễ dàng và có thể lặp lại, giúp tiết kiệm thời gian và tài nguyên trong quá trình kiểm thử.
- Chính xác và đáng tin cậy: Các test case cần đảm bảo tính chính xác và đáng tin cậy trong việc xác định các lỗi và sự cố của hệ thống.

Quá trình xác định và thiết kế các test case là một phần quan trọng trong quá trình kiểm thử hệ thống. Chúng giúp đảm bảo rằng các tính năng và chức năng của hệ thống được kiểm tra một cách toàn diện và đáng tin cậy, từ đó đảm bảo chất lượng và sự thành công của dự án.

9.2 Test Reports - Báo cáo kiểm thử

Test Case Description : Check Login

STEP	STEP DETAILS	TEST DATA	EXPECTED RESULTS	A RESULT
1	Điều hướng tới http://localhost/thuc/hoc/auth/login/index.php		Trang đăng nhập được mở	pass
2	Nhập user ,password	Userid = nhungnguyen Pass = 123456789	Thông tin đăng nhập được nhập	pass

3	Click Sign in	Userid = nhungnguyen Pass = 123456789	User được đăng nhập và vào trang tiếp theo	pass
4	Click Sign in	Userid = nhung Pass = 123456789	Hiện bảng thông báo tài khoản hoặc mật khẩu không chính xác	pass
5	Chỉ nhập user/password và click sign in		Hiện bảng thông báo vui lòng điền thông tin vào password/user	pass
6	Chức năng remember me		Có thể bật	pass
7	Click Sign up		Trang đăng kí được mở ra	pass

Bảng 13: Test case Login

Test Case Description : Check Signup

STEP	STEP DETAILS	TEST DATA	EXPECTED RESULTS	A RESULT
1	Điều hướng tới http://localhost/thue/register/index.php		Trang đăng ký được mở	pass
2	Nhập username ,password, fullname, address, birthday, gender, email, phone number	username = abc123 password =123456789 fullname = abc address = Hà Nội birthday = 01/01/2002 gender = Male email = abc@gmail.com	Thông tin đăng nhập được nhập	pass

		phone number = 037****162		
3	Click Đăng Ký		Đăng ký thành công	pass
4	Đăng ký account mới trùng username Click Sign Up	username = abc123 password =123456789 fullname = abc address = Hà Nội birthday = 01/01/2002 gender = Male email = abc@gmail.com phone number = 037****162	Hiện bảng thông báo username đã tồn tại	pass
5	Nhập không đủ các trường thông tin Hoặc không nhập thông tin	username = abc123 password =123456789 fullname = abc address = Hà Nội birthday = 01/01/2002 gender = email = phone number =	Hiện bảng thông báo (tên trường thông tin) không được để trống	pass

Bảng 14: Test case Register

Test Case Description : Check Tính thuế

Selenium IDE - test*

Project: test*

Tests +

Search tests...

http://localhost

	Command	Target	Value
1	open	/SQA_N02_NHOM5/index.php	
2	set window size	1080x864	
3	click	name=year	
4	type	name=year	2023
5	click	name=name	
6	type	name=name	KhangTM
7	click	name=baohiem	
8	click	name=trulaodong	
9	type	name=trulaodong	11
10	click	name=truphuthuoc	
11	click	name=songuoipt	
12	type	name=songuoipt	3
13	click	name=thang1	
14	type	name=thang1	30
15	click	css=row:nth-child(2) > .col-sm-6:nth-child(1)	
16	click	name=thang2	
17	type	name=thang2	27

Command: add selection

Target:

Value:

Description:

Log Reference

Hình 5: AutoTest Selenium test case tính thuế

DEMO

Đăng Ký

<div>Username</div> <div></div>	<div>Password</div> <div></div>
<div>Fullname</div> <div></div>	<div>Address</div> <div></div>
<div>Birthday</div> <div><div></div><div></div></div>	<div>Gender</div> <div><input checked="" type="radio"/> Male <input type="radio"/> Female</div>
<div>Email</div> <div></div>	<div>Phone Number</div> <div></div>

Đăng Ký

[Bạn đã có tài khoản?](#)

Hình 5: Giao diện đăng ký

Sign In

f

USERNAME

Username

PASSWORD

Password

Sign In

☒ Remember Me ☐ Về Trang Chủ

Welcome to login

Don't have an account?

Sign Up

Hình 6: Giao diện đăng nhập

[Đăng nhập](#)

Thông tin cá nhân

Năm:

Họ và tên:

Bảo hiểm xã hội:
☒ Có ☐ Không

Mức miễn trừ với người lao động:

Mức miễn trừ với người phụ thuộc:

Số người phụ thuộc:

Thu nhập cá nhân

Tháng 1: <input type="text" value="30"/>	Tháng 7: <input type="text" value="27"/>
Tháng 2: <input type="text" value="27"/>	Tháng 8: <input type="text" value="28"/>
Tháng 3: <input type="text" value="29"/>	Tháng 9: <input type="text" value="27"/>
Tháng 4: <input type="text" value="26"/>	Tháng 10: <input type="text" value="29"/>
Tháng 5: <input type="text" value="31"/>	Tháng 11: <input type="text" value="33"/>
Tháng 6: <input type="text" value="33"/>	Tháng 12: <input type="text" value="34"/>

Xác nhận

Hình 7:Giao diện Web default

Bảng thuế cá nhân

Tháng	Thu nhập	Thuế thu nhập
1	30	0.29464
2	27	0.12232
3	29	0.22232
4	26	0.07232
5	31	0.39464
6	33	0.59464
7	27	0.12232
8	28	0.17232
9	27	0.12232
10	29	0.22232
11	33	0.59464
12	34	0.69464

Thông tin cá nhân

Năm	Họ và tên	Bảo hiểm xã hội
2023	KhangTM	Yes

Kết quả

Đã nộp	Thực tế	Nhận lại
3.62944	2.96784	0.6616

Tính lại

Hình 8:Giao diện Web Result

HTTP Request

Name: Login

Comments:

Basic Advanced

Web Server

Protocol [http]: Server Name or IP: localhost Port Number:

HTTP Request

GET Path: SQA_N02_NHOM5/auth/login/index.php Content encoding:

☐ Redirect Automatically ☒ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data ☐ Browser-compatible headers

Parameters Body Data Files Upload

Send Parameters With the Request:

Name:	Value	URL Encode?	Content-Type	Include Equals?
username	nguyennhung	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
password	12345678	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Hình 9: Config test hiệu năng Login

test_login.jmx (E:\jmeter\test_login.jmx) - Apache JMeter (5.5)

File Edit Search Run Options Tools Help

Thread Group

- Login
- View Results Tree
- Summary Report

View Results Tree

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: ☐ Errors ☐ Successes Configure

Search: ☐ Case sensitive ☐ Regular exp. Search Reset

Text

Sampler result Request Response data

Request Body Request Headers

```
1 GET http://localhost/SQA_N02_NHOM5/auth/login/index.php?username=nguyennhung&password=12345678
2
3 GET data:
4
5
6 no cookies
7
```

Raw HTTP

Hình 10: result test hiệu năng Login