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 TIN HỌC QUẢN LÝ**

**ĐỀ TÀI : KIỂM THỬ THỦ CÔNG CHO**

**ỨNG DỤNG WEB SCTV**

Sinh viên thực hiện : **Lê Thị Sâm**

Lớp : **47K14**

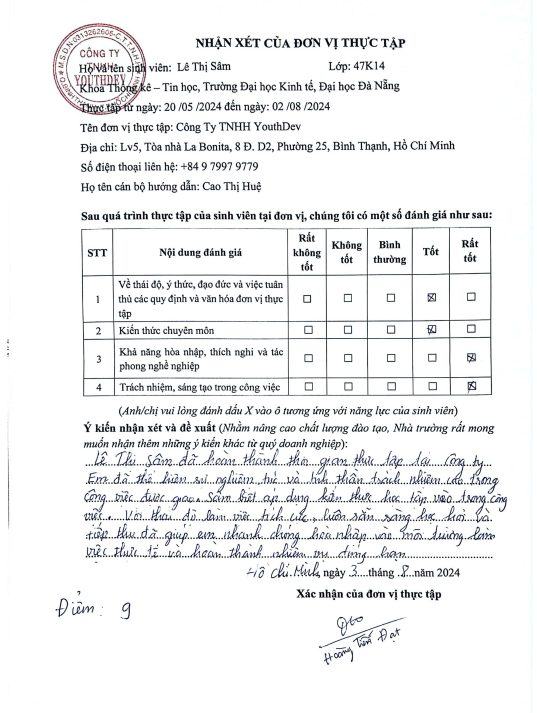
Đơn vị thực tập : **Công Ty TNHH YouthDev**

Cán bộ hướng dẫn : **Cao Thị Huệ**

Giảng viên hướng dẫn : **TS. Đặng Trung Thành**

***Đà Nẵng, 8/******2024***

### NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP



### LỜI CẢM ƠN

Lời cám ơn đầu tiên em xin gửi đến Công Ty TNHH YouthDev – đơn vị đã tiếp nhận, hỗ trợ nhiệt tình và tạo mọi điều kiện thuận lợi cho em tiếp cận thực tế, được trau dồi kiến thức và kinh nghiệm.

Em xin cảm ơn đến quý Thầy Cô Khoa Thống kê- Tin học trường Đại Học Kinh Tế Đà Nẵng đã tạo điều kiện để em có điều kiện thực tập. Đặc biệt, em xin cảm ơn thầy Đặng Trung Thành, người đã tận tình hướng dẫn em hoàn thành bài báo cáo này.

Cuối cùng em xin cảm ơn chị Cao Thị Huệ người hướng dẫn em tại công ty TNHH YouthDev đã giúp đỡ, tạo điều kiện, cung cấp những tài liệu cho em trong quá trình thực tập về mảng Tester để hoàn thành được kỳ thực tập tốt hơn.

Vì thời gian và kiến thức còn hạn hẹp nên bài báo cáo không thể tránh khỏi những thiếu sót, Em rất mong nhận được những ý kiến đóng góp và phản hồi từ quý thầy cô để em có thể khắc phục được những sai sót cũng như rút ra được những bài học cho mình và trau dồi thêm những kiến thức mới. Em xin chân thành cảm ơn!

### LỜI CAM ĐOAN

Em xin cam đoan đề tài: “Nghiên cứu và ứng dụng kiểm thử tự động cho website SCTV ” là một công trình nghiên cứu độc lập không có sự sao chép. Đề tài là một sản phẩm mà em đã hoàn thành sau quá trình nỗ lực nghiên cứu dưới sự hướng dẫn của thầy Đặng Trung Thành và chị mentor Cao Thị Huệ.

Trong quá trình làm báo cáo có tham khảo một số tài liệu có nguồn gốc rõ ràng.

Các số liệu và kết quả thực hành trong đề tài là trung thực và hoàn toàn không sao chép hay sử dụng kết quả nghiên cứu đề tài của người khác. Nếu phát hiện có gì sai sót trong đề tài, em xin chịu hoàn toàn trách nhiệm.

### MỤC LỤC

NHẬN XÉT CỦA ĐƠN VỊ THỰC TẬP i

LỜI CẢM ƠN ii

LỜI CAM ĐOAN iii

MỤC LỤC iv

DANH MỤC CÁC HÌNH ẢNH vii

DANH MỤC BẢNG BIỂU viii

DANH MỤC CÁC TỪ VIẾT TẮT ix

LỜI MỞ ĐẦU 1

CHƯƠNG 1. TỔNG QUAN VỀ ĐƠN VỊ THỰC TẬP 2

1.1. Giới thiệu tổng quát về doanh nghiệp thực tập 2

1.1.1. Giới thiệu Công Ty TNHH YouthDev 2

1.1.2. Tầm nhìn và sứ mệnh 2

1.1.3. Phương châm hoạt động 2

1.1.4. Cơ cấu tổ chức công ty 3

1.2. Tổng quan về vị trí việc làm 3

1.2.1. Mô tả về vị trí tester 3

a. Chuyên viên kiểm thử phần mềm: 3

b. Manual Test: 3

c. Nhân viên kiểm thử phần mềm QC: 3

d. Nhân viên kiểm thử phần mềm QA: 4

1.2.2. Các kỹ năng cần có của một tester [4] 4

1.2.3. Cơ hội nghề nghiệp và mức lương của vị trí Tester 5

CHƯƠNG 2. TỔNG QUAN VỀ LÝ THUYẾT 6

2.1. Tổng quan về kiểm thử phần mềm 6

2.1.1. Khái niệm về kiểm thử [6] 6

2.1.2. Tại sao kiểm thử phần mềm lại cần thiết? 6

2.2. Các nguyên tắc kiểm thử phần mềm [6] 6

2.3. Vòng đời kiểm thử phần mềm 7

Giai đoạn 1: Phân tích yêu cầu. 8

Giai đoạn 2: Lập kế hoạch phân tích . 8

Giai đoạn 3: Phát triển trường hợp kiểm thử. 8

Giai đoạn 4: Thiết lập môi trường. 8

Giai đoạn 5: Thực hiện kiểm thử. 9

Giai đoạn 6: Đóng kiểm thử. 9

2.4. Các phương pháp quản lý dự án 10

2.4.1. Phương pháp Waterfall Model [9] 10

2.4.2. Phương pháp Scrum [10] 11

2.4.3. Phương pháp V-mode. 11

2.4.4. Phương pháp Agile 12

2.5. Một số loại kiểm thử  [13] 13

a. Kiểm thử chức năng ( Functionality Testing) 13

b. Kiểm thử phi chức năng  (non-functional) 13

c. Kiểm thử cấu trúc 13

d. Kiểm thử thay đổi 13

2.6. Test case 14

2.6.1. Khái niệm về test case 14

2.6.2. Hình thức kiểm thử phần mềm 14

2.7. Các kỹ thuật kiểm thử phần mềm 14

2.8. Phương pháp kiểm thử phần mềm 15

2.8.1. Kiểm thử hộp đen 15

2.8.2. Kiểm thử hộp trắng 16

2.8.3. Kiểm thử hộp xám 16

2.9. Error/ Fault/Failure/Bug/Defect 16

2.10. Quy trình quản lí lỗi. [15] 17

CHƯƠNG 3. GIỚI THIỆU VỀ WEBSITE SCTV 19

3.1. Giới thiệu web SCTV 19

3.2. Sơ đồ use case tổng quát 19

3.3. Đặc tả, Phân tích Use case cho “Đăng Ký”, “Đăng nhập”, Phương thức thanh toán ” 20

3.3.1. Use case đăng ký 20

a. Đăng ký tài khoản 21

b. Đăng ký Các gói dịch vụ 23

3.3.2. Use case “ Đăng nhập ” 26

3.3.3. Use case “Phương thức thanh toán ” 28

CHƯƠNG 4. TRIỂN KHAI VÀ THỰC HIỆN TEST CASE 32

4.1.1. Test case cho chức năng 'Đăng ký và Đăng nhập cho WEB SCTV" 32

4.1.2. Test case cho chức năng " Phương Thức Thanh Toán cho WEB SCTV" 34

CHƯƠNG 5. KẾT QUẢ THỰC HIỆN TEST WEB 38

5.1. Kết quả kiểm thử cho chức năng 38

5.1.1. Kết quả kiểm thử cho chức năng “Đăng Ký”, “Đăng Nhập”, “Phương Thức Thanh Toán ” cho WEB SCTV 38

5.2. Bảng kết quả kiểm thử 38

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 40

TÀI LIỆU THAM KHẢO 42

CHECK LIST CỦA BÁO CÁO 44

PHỤ LỤC 45

### DANH MỤC CÁC HÌNH ẢNH

[Hình 1.1. Logo Công Ty TNHH YOUTHDEV 2](#_Toc173865548)

[Hình 2.1 Các giai đoạn của STLC 7](#_Toc173865549)

[Hình 2.2. Mô hình Waterfall 10](#_Toc173865550)

[Hình 2.3. Mô hình Scrum 11](#_Toc173865551)

[Hình 2.4. Mô hình V 12](#_Toc173865552)

[Hình 2.5. Mô hình Agile 12](#_Toc173865553)

[Hình 2.6. Bảng quyết định 15](#_Toc173865554)

[Hình 2.7. Quy trình quản lí lỗi 17](#_Toc173865555)

[Hình 3.1. Use case tổng quát hệ thống 20](#_Toc173865556)

[Hình 3.2. Sơ đồ hoạt động " Đăng ký tài khoản" 22](#_Toc173865557)

[Hình 3.3. Màn hình Đăng Ký Tài khoản 23](#_Toc173865558)

[Hình 3.4. Sơ đồ hoạt động "Đăng ký cho các gói dịch vụ". 25](#_Toc173865559)

[Hình 3.5. Màn hình đăng ký các gói dịch vụ 26](#_Toc173865560)

[Hình 3.6. Sơ đồ hoạt động "Đăng nhập" 27](#_Toc173865561)

[Hình 3.7. Màn hình đăng nhập 28](#_Toc173865562)

[Hình 3.8. Sơ đồ hoạt động" Phương thức thanh toán" 30](#_Toc173865563)

[Hình 3.9. Màn hình phương thức thanh toán 31](#_Toc173865564)

[Hình 4.1. Thiết kế Test case “Đăng ký, Đăng nhập” 34](#_Toc173865565)

[Hình 4.2. Thiết kế test case " Phương thức thanh toán" 37](#_Toc173865566)

[Hình 5.1. Bug report “Đăng ký, Đăng nhập” 38](#_Toc173865567)

[Hình 5.2. Bug report “Phương thức thanh toán” 38](#_Toc173865568)

### DANH MỤC BẢNG BIỂU

[Hình 1.1. Logo Công Ty TNHH YOUTHDEV 2](#_Toc173865569)

[Hình 2.1 Các giai đoạn của STLC 7](#_Toc173865570)

[Hình 2.2. Mô hình Waterfall 10](#_Toc173865571)

[Hình 2.3. Mô hình Scrum 11](#_Toc173865572)

[Hình 2.4. Mô hình V 12](#_Toc173865573)

[Hình 2.5. Mô hình Agile 12](#_Toc173865574)

[Hình 2.6. Bảng quyết định 15](#_Toc173865575)

[Hình 2.7. Quy trình quản lí lỗi 17](#_Toc173865576)

[Hình 3.1. Use case tổng quát hệ thống 20](#_Toc173865577)

[Hình 3.2. Sơ đồ hoạt động " Đăng ký tài khoản" 22](#_Toc173865578)

[Hình 3.3. Màn hình Đăng Ký Tài khoản 23](#_Toc173865579)

[Hình 3.4. Sơ đồ hoạt động "Đăng ký cho các gói dịch vụ". 25](#_Toc173865580)

[Hình 3.5. Màn hình đăng ký các gói dịch vụ 26](#_Toc173865581)

[Hình 3.6. Sơ đồ hoạt động "Đăng nhập" 27](#_Toc173865582)

[Hình 3.7. Màn hình đăng nhập 28](#_Toc173865583)

[Hình 3.8. Sơ đồ hoạt động" Phương thức thanh toán" 30](#_Toc173865584)

[Hình 3.9. Màn hình phương thức thanh toán 31](#_Toc173865585)

[Hình 4.1. Thiết kế Test case “Đăng ký, Đăng nhập” 34](#_Toc173865586)

[Hình 4.2. Thiết kế test case " Phương thức thanh toán" 37](#_Toc173865587)

[Hình 5.1. Bug report “Đăng ký, Đăng nhập” 38](#_Toc173865588)

[Hình 5.2. Bug report “Phương thức thanh toán” 38](#_Toc173865589)

### DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Từ viết đầy đủ** |
| 1 | QC | Quality Control |
| 2 | QA | Quality Assurance |
| 3 | N/A | Not available |

### LỜI MỞ ĐẦU

**1. Mục tiêu của đề tài**

Tìm hiểu về web SCTV và triển khai một phương pháp kiểm thử thủ công để đảm bảo chính xác và độ tin cậy của hệ thông.

Tìm hiểu về web SCTV: Nghiên cứu và hiểu rõ về cấu trúc, chức năng và các dịch vụ mà web SCTV cung cấp, bao gồm các chương trình VIP và các video miễn phí.

Phương pháp kiểm thử thủ công: Xây dựng và triển khai một phương pháp kiểm thử thủ công để kiểm tra tính chính xác và độ tin cậy của hệ thống web SCTV.

Cung cấp một quy trình kiểm thử thủ công đáng tin cậy để đảm bảo chất lượng của web SCTV.

**2. Phương pháp nghiên cứu**

* Nghiên cứu lý thuyết từ tài liệu sẵn có.
* Sử dụng các công cụ kiểm thử thủ công để thực hành kiểm thử.

**3. Đối tượng và phạm vi nghiên cứu**

* Đối tượng nghiên cứu: Web SCTV
* Phạm vi nghiên cứu: Nghiên cứu lý thuyết cơ bản về kiểm thử phần mềm và các công cụ kiểm thử phần mềm thủ công, sau đó thực hiện kiểm thử.

**4 . Kết cấu của đề tài**

Đề tài được tổ chức gồm phần mở đầu, 4 chương nội dung và phần kết luận:

* Mở đầu
* Chương 1: Tổng quan về đơn vị thực tập.
* Chương 2: Tổng quan về cơ sở lý thuyết.
* Chương 3: Triển khai và thiết kế test case web SCTV.
* Chương 4: Kết quả.
* Kết luận và hướng phát triển.

### TỔNG QUAN VỀ ĐƠN VỊ THỰC TẬP

#### Giới thiệu tổng quát về doanh nghiệp thực tập

##### Giới thiệu Công Ty TNHH YouthDev

Công ty TNHH YouthDev được thành lập năm 2014, là một công ty gia công và giải pháp công nghệ cao tại TP.HCM. Công ty tập trung chủ yếu vào việc áp dụng các công nghệ cung cấp dịch vụ phát triển ứng dụng web, di động và các hệ thống phần mềm tùy chỉnh theo nhu cầu của khách hàng, mang đến cho khách hàng những giải pháp tiên tiến và hiệu quả. [1]



Hình 1.1. Logo Công Ty TNHH YOUTHDEV

##### Tầm nhìn và sứ mệnh

YouthDev hướng đến trở thành công ty hàng đầu trong lĩnh vực gia công và giải pháp công nghệ cao, không chỉ tại TP.HCM mà còn trên toàn thị trường công nghệ trong và ngoài nước. Công ty đặt mục tiêu dẫn đầu trong việc áp dụng các công nghệ và ý tưởng mới, góp phần vào sự phát triển bền vững của khách hàng và cộng đồng.

YouthDev cam kết cung cấp các giải pháp công nghệ tiên tiến, giúp doanh nghiệp tối ưu hóa quy trình làm việc và nâng cao hiệu quả hoạt động. Với đội ngũ chuyên gia giàu kinh nghiệm và nhân viên nhiệt huyết, công ty đảm bảo mang lại các sản phẩm và dịch vụ chất lượng cao, luôn đặt lợi ích của khách hàng lên hàng đầu. YouthDev không ngừng nỗ lực để khẳng định vị thế của mình thông qua những dự án thành công và các giải thưởng uy tín trong ngành công nghệ.

##### Phương châm hoạt động

YouthDev luôn sẵn sàng giải quyết mọi vấn đề và cam kết mang lại giá trị thực sự cho khách hàng dựa trên sự tin tưởng và khả năng công nghệ vững chắc của chúng tôi.

##### Cơ cấu tổ chức công ty

Những lãnh đạo cấp cao của công ty:

* Hứa Phước Trường: Chủ tịch HĐQT/ Tổng giám đốc
* Toàn bộ công ty với hơn 20 nhân viên. YouthDev hội tụ đông đảo các chuyên gia phát triển phần mềm giàu kinh nghiệm, vững chuyên môn, đã từng tham gia nhiều dự án các doanh nghiệp trong và ngoài nước.

#### **Tổng quan về vị trí việc làm**

##### Mô tả về vị trí tester

Vị trí tester là một vai trò quan trọng trong quá trình phát triển và triển khai phần mềm. Tester là người chịu trách nhiệm đảm bảo chất lượng của ứng dụng hoặc sản phẩm phần mềm trước khi nó được ra mắt tới người dùng.

Tester thường là thành viên của nhóm phát triển phần mềm và có vai trò kiểm tra, đánh giá và xác minh tính toàn vẹn, chất lượng, hiệu suất và tính năng của ứng dụng. Công việc của tester bao gồm tạo ra các kịch bản kiểm thử, thực hiện các bài kiểm tra, tìm, báo cáo các lỗi, sự cố, đảm bảo rằng ứng dụng hoạt động một cách đáng tin cậy và đáp ứng yêu cầu của người dùng. [2]

###### Chuyên viên kiểm thử phần mềm:

Chuyên viên kiểm thử phần mềm, hay còn gọi là Tester, đóng vai trò quan trọng trong quá trình phát triển phần mềm. Họ đảm bảo rằng sản phẩm cuối cùng đáp ứng đầy đủ các yêu cầu của khách hàng và hoạt động một cách ổn định, hiệu quả.

###### Manual Test:

Manual Testing là quá trình kiểm thử phần mềm được thực hiện thủ công bởi các tester mà không sử dụng các công cụ tự động. Quá trình này bao gồm việc thực hiện các test case và kịch bản kiểm thử để phát hiện các lỗi hoặc vấn đề trong phần mềm. [3]

###### Nhân viên kiểm thử phần mềm QC:

Nhân viên kiểm thử phần mềm QC là người chịu trách nhiệm đảm bảo chất lượng của sản phẩm phần mềm thông qua việc thực hiện các hoạt động kiểm thử và kiểm soát chất lượng. Họ làm việc chặt chẽ với đội phát triển và các bên liên quan để đảm bảo rằng sản phẩm đáp ứng đầy đủ các yêu cầu và tiêu chuẩn đã đề ra.

###### Nhân viên kiểm thử phần mềm QA:

Nhân viên kiểm thử phần mềm QA chịu trách nhiệm đảm bảo chất lượng của sản phẩm phần mềm thông qua việc thiết lập và duy trì các quy trình và tiêu chuẩn chất lượng. Họ làm việc để đảm bảo rằng toàn bộ quá trình phát triển phần mềm đáp ứng các yêu cầu chất lượng và sản phẩm cuối cùng hoạt động ổn định, hiệu quả.

##### Các kỹ năng cần có của một tester [4]

Tester là một trong những vai trò quan trọng trong quá trình phát triển phần mềm. Công việc tester đảm bảo các hệ thống, chương trình và ứng dụng hoạt động như mong đợi trước khi ra mắt. Để làm được công việc này, tester cần có một số kỹ năng sau:

**Kỹ năng phân tích: C**ần 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. Kiểm thử phần mềm cho bạn cơ hội sử dụng khả năng sáng tạo, phân tích để tìm ra những thứ mà người khác không thấy được.

**Kỹ năng học hỏi nhanh:** Công nghệ ngày càng phát triển, nó phát triển từng ngày, mỗi ngày đều có nhiều điều mới mẻ. Nếu chúng ta không cập nhật thông tin thì chúng ta sẽ tụt hậu. Do vậy, chúng ta phải không ngừng học hỏi, trao dồi kiến thức, nổ lực tìm kiếm, cập nhật mọi sự thay đổi để phục vụ cho công việc, lĩnh vực mà chúng ta đang ngày đêm gắn bó cùng nó.

**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 vàn, hàng tỉ thứ nhỏ nhặt khác. Sản phẩm hoàn thiện, chất lượng cao sẽ tạo thêm niềm tin và uy tín của công ty với đối tác. Do đó, tester bắt buộc phải có tính cẩn thận, tỉ mỉ, nhạy bén. Nếu đã qua khâu kiểm tra mà sản phẩm vẫn bị lỗi, tester phải chịu trách nhiệm hoàn toàn.

**Kỹ năng giao tiếp:** Là một tester 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, còn cần phải diễn giải tốt hơn và có thể cung cấp dữ liệu để giải quyết vấn đề.

**Tiếng Anh, ngoại ngữ tốt là một ưu thế: Đối với** dân 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.

**Nên có tính “support”:** Một tester không cần phải là ngôi sao sáng nhất trong dự án, nhưng họ sẵn sàng đóng góp vượt qua trách nhiệm của mình để đảm bảo chất lượng phần mềm tốt nhất. Họ không ngại khó khăn, sẵn sàng đối mặt với thử thách và không ngần ngại học hỏi để trau dồi kiến thức. Tính chất này mang lại lợi thế lớn cho nghề tester, giúp họ không chỉ hoàn thành công việc của mình mà còn đóng góp tích cực vào thành công chung của dự án.

##### Cơ hội nghề nghiệp và mức lương của vị trí Tester

Với sự phát triển của ngành công nghệ, nhu cầu kiểm thử phần mềm ngày càng tăng. Các công ty luôn cần những chuyên gia kiểm thử để đảm bảo chất lượng sản phẩm. Tester có thể làm việc trong nhiều lĩnh vực khác nhau như tài chính, y tế, thương mại điện tử, game, viễn thông, và nhiều lĩnh vực khác. Với kinh nghiệm và kỹ năng, tester có thể thăng tiến lên các vị trí cao hơn như Test Lead, Test Manager, QA Manager hoặc các vị trí liên quan đến quản lý dự án. Nhiều cơ hội làm việc tại các công ty nước ngoài hoặc tham gia vào các dự án quốc tế với mức lương và phúc lợi hấp dẫn. Kỹ năng kiểm thử phần mềm cũng giúp tester dễ dàng chuyển đổi sang các vị trí khác trong ngành công nghệ thông tin như phát triển phần mềm, phân tích kinh doanh (Business Analyst), hoặc quản lý dự án.

Mức lương của Tester phụ thuộc vào nhiều yếu tố như kinh nghiệm, vị trí công việc, lĩnh vực hoạt động và địa điểm làm việc.

**Tại Việt Nam:**

* Junior Tester: Mức lương trung bình từ 8 - 12 triệu VNĐ/tháng.
* Mid-level Tester: Mức lương trung bình từ 12 - 20 triệu VNĐ/tháng.
* Senior Tester/Test Lead: Mức lương trung bình từ 20 - 35 triệu VNĐ/tháng.
* Test Manager/QA Manager: Mức lương trung bình từ 35 - 50 triệu VNĐ/tháng. [5]

### TỔNG QUAN VỀ LÝ THUYẾT

#### Tổng quan về kiểm thử phần mềm

##### Khái niệm về kiểm thử [6]

Kiểm thử phần mềm là một quá trình kiểm tra để đưa ra những đánh giá về chức năng của một ứng dụng phần mềm với mục đích xem xét liệu phần mềm đó đã được phát triển theo đúng tiêu chuẩn hay không, có đáp ứng được các yêu cầu cụ thể không. Bên cạnh đó, việc xác định được chi tiết lỗi đảm bảo rằng sản phẩm khi cho ra đời sẽ không có khuyết điểm, hướng đến mục tiêu tạo ra một sản phẩm phần mềm hoàn chỉnh và chất lượng chỉ.

Kiểm thử phần mềm sẽ được test theo 2 hướng: Positive và Negative

##### Tại sao kiểm thử phần mềm lại cần thiết?

Kiểm thử phần mềm thực sự được yêu cầu để chỉ ra các khiếm khuyết và lỗi đã được thực hiện trong các giai đoạn phát triển.

Điều cần thiết là vì nó đảm bảo rằng khách hàng thấy tổ chức đáng tin cậy và sự hài lòng của họ trong ứng dụng được duy trì.

#### Các nguyên tắc kiểm thử phần mềm [6]

**Nguyên tắc 1: Kiểm thử chứng minh sự hiện diện của lỗi**

Kiểm thử chỉ có thể chứng minh được rằng sản phẩm có lỗi. Kiểm thử phần mềm không thể chứng mình rằng sản phẩm không còn lỗi. Có nghĩa là khi bạn test không ra lỗi không đồng nghĩa với việc phần mềm đúng hoàn toàn.

**Nguyên tắc 2: Kiểm thử toàn diện là không thể**

Kiểm thử toàn diện tất cả các trường hợp có thể xảy ra là không khả thi, do số lượng tổ hợp các điều kiện kiểm thử là vô hạn. Do đó, cần tập trung vào các trường hợp kiểm thử quan trọng nhất và sử dụng các phương pháp kiểm thử dựa trên rủi ro để tối ưu hóa quá trình kiểm thử.

**Nguyên tắc 3: Kiểm tra càng sớm càng tốt**

Lỗi phần mềm càng được phát hiện sớm trong quá trình phát triển, chi phí sửa chữa lỗi càng thấp. Việc kiểm thử sớm giúp phát hiện và sửa lỗi trước khi chúng ảnh hưởng đến các giai đoạn sau của dự án.

**Nguyên tắc 4: Lỗi thường được phân bố tập trung**

Thường thì các lỗi phần mềm không được phân bố đều mà tập trung vào một số module hoặc chức năng cụ thể. Do đó, cần tập trung kiểm thử kỹ lưỡng vào các khu vực dễ xảy ra lỗi.

**Nguyên tắc 5: Nguyên lý thuốc trừ sâu**

Nếu cùng một bộ kiểm thử được sử dụng lặp đi lặp lại, các lỗi tương tự sẽ không còn được phát hiện. Do đó, cần thường xuyên xem xét và cải tiến các bộ kiểm thử để phát hiện các lỗi mới.

**Nguyên tắc 6: Kiểm thử phụ thuộc vào ngữ cảnh**

Các phương pháp và kỹ thuật kiểm thử phụ thuộc vào loại phần mềm và mục đích sử dụng của nó. Ví dụ, các phương pháp kiểm thử cho ứng dụng web sẽ khác so với kiểm thử hệ thống nhúng.

**Nguyên tắc 7: Sự sai lầm về việc không có lỗi**

Việc không phát hiện ra lỗi không có nghĩa là phần mềm hoàn toàn đúng. Một sản phẩm phần mềm có thể không có lỗi nhưng nếu không đáp ứng được các yêu cầu của người dùng, thì nó vẫn là một sản phẩm không thành công.

#### Vòng đời kiểm thử phần mềm

Vòng đời kiểm thử phần mềm (STLC) là quá trình kiểm thử được thực hiện một cách có hệ thống và có kế hoạch. Trong quá trình STLC, các hoạt động khác nhau được thực hiện để cải thiện chất lượng sản phầm. [7]

Các giai đoạn của STLC: có 6 giai đoạn



Hình 2.1 Các giai đoạn của STLC

###### Giai đoạn 1: Phân tích yêu cầu.

Hiểu rõ các yêu cầu của phần mềm để thiết kế các trường hợp kiểm thử phù hợp.

• Rà soát các tài liệu yêu cầu và đặc tả.

• Xác định các mục tiêu kiểm thử.

###### Giai đoạn 2: Lập kế hoạch phân tích .

Kế hoạch kiểm thử là một tài liệu mô tả các mục tiêu, phạm vi, phương pháp tiếp cận, và tập trung vào nỗ lực kiểm thử phần mềm. Cấu trúc chung của một test plan:

• Tên dự án.

• Danh sách các module cần test.

• Ngày bắt đầu, ngày kết thúc.

• Danh sách các test case.

• Nhân sự tham gia.

• Tài nguyên sử dụng (Severs, Workstations, Printers).

• Kế hoạch thực hiện.

###### Giai đoạn 3: Phát triển trường hợp kiểm thử.

Thiết kế và phát triển các trường hợp kiểm thử dựa trên yêu cầu phần mềm.

Ở giai đoạn này thực hiện các nhiệm vụ chủ yếu:

• Rà soát các yêu cầu cần thiết trước khi tiến hành kiểm thử như: tài liệu đặc tả, tài liệu đặc tả, tài liệu thiết kế.

• Xác định các điều kiện kiểm thử

• Thiết kế test case

• Đánh giá tính khả thi trong việc kiểm thử cũng như yêu cầu của hệ thống.

• Chuẩn bị môi trường test cũng như xác định các yêu cầu về cơ sở hạ tầng cũng như các công cụ kiểm thử tương ứng.

###### Giai đoạn 4: Thiết lập môi trường.

Một môi trường thử nghiệm là một thiết lập của phần mềm và phần cứng cho các đội kiểm thử để tiến hành kiểm tra các test case. Nói cách khác, nó hỗ trợ thực hiện kiểm thử với phần cứng, phần mềm và cấu hình mạng.

Đối với môi trường kiểm thử, khu vực trọng điểm để thiết lập bao gồm:

* Hệ thống và các ứng dụng.
* Dữ liệu kiểm thử.
* Máy chủ cơ sở dữ liệu.
* Môi trường để chạy Font end.
* Hệ điều hành clienc.
* Trình duyệt.
* Phần cứng bao gồm hệ điều hành máy cmá.
* Mạng.
* Các tài liệu cần thiết như tài liệu tham khảo/hướng dẫn cấu hình/hướng dẫn cài đặt/hướng dẫn sử dụng.

###### Giai đoạn 5: Thực hiện kiểm thử.

Khi developers đã code và đưa sản phẩm lên môi trường kiểm thử, tester sẽ thực thi dựa trên Test Case đã viết. Trong quá trình test, nếu phát hiện ra bug (lỗi) thì tester sẽ log (viết) lên các tool quản lý lỗi. Bug của lập trình viên nào sẽ giao lại cho người đấy xử lý. Khi nào developers fix bug xong, tester sẽ nhận lại và tiến hành kiểm thử.

Nếu lỗi đã được sửa, tính năng hoạt động ổn định, tester sẽ đổi trạng thái thành Close Bug. Trường hợp lỗi vẫn chưa được fix thành công, trạng thái sẽ được đổi thành Re-open để developers thực hiện fix lại. Khi nào bug được fix thành công mới được đóng lại việc test tính năng đấy.

Trong cả quá trình kiểm thử phần mềm, tester ưu tiên kiểm tra chức năng chính trước, chức năng phụ và giao diện sẽ thực hiện test sau. Quá trình kiểm thử phần mềm bắt buộc phải tuân thủ thời gian đã đề ra, mọi người trong team đôn đốc nhau để kịp tiến độ bàn giao sản phẩm. Cuối cùng, tester thực hiện làm báo cáo tùy theo yêu cầu của dự án để đánh giá việc kết thúc quy trình kiểm thử phần mềm.

###### Giai đoạn 6: Đóng kiểm thử.

Ở giai đoạn cuối cùng, tester chuẩn bị báo cáo kết thúc kiểm thử, tổng hợp lại các chỉ số trong quá trình test. Cả team phát triển sẽ ngồi họp để đánh giá toàn bộ các tiêu chí xác định kiểm thử đã đủ hay chưa. Những tiêu chí này khác nhau tùy theo từng dự án, thông thường bao gồm:

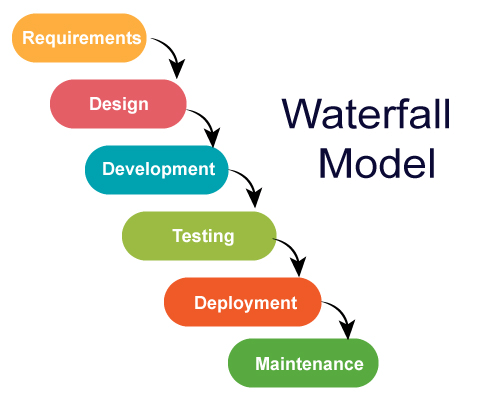
* Số lượng test case tối đa được thực thi Passed.
* Tỷ lệ lỗi giảm xuống dưới mức nhất định.
* Deadline được chốt từ giai đoạn làm kế hoạch kiểm thử.

Quy trình kiểm thử phần mềm thường chỉ được kết thúc khi sản phẩm được bàn giao cho khách hàng. Ngoài ra, hoạt động kiểm thử có thể kết thúc trong các trường hợp sau:

* Khi 1 dự án bị hủy bỏ.
* Khi các mục tiêu chính đã hoàn thành.
* Khi việc bảo trì hoặc cập nhật đã hoàn thành.

#### Các phương pháp quản lý dự án

##### Phương pháp Waterfall Model [9]



Hình 2.2. Mô hình Waterfall.

Mô hình Waterfall hay mô hình thác nước là một mô hình dùng để mô tả các bước phát triển phần mềm theo trình tự và logic chặt chẽ. Đây là mô hình đầu tiên được sử dụng rộng rãi trong ngành công nghiệp phần mềm và được chia thành các giai đoạn, đầu ra của giai đoạn trước sẽ trở thành đầu vào của giai đoạn tiếp theo. [5]

Thường được áp dụng cho các dự ánkhông thường xuyên bị thay đổi về yêu cầu.

##### Phương pháp Scrum [10]



Hình 2.3. Mô hình Scrum

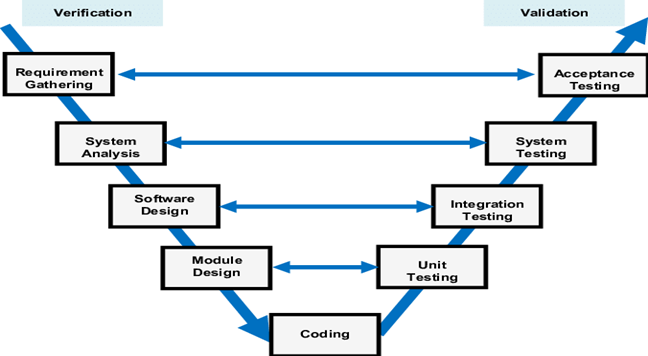
Scrum là 1 dạng của mô hình Agile và là framework phổ biến nhất khi thực hiện mô hình Agile. Scrum là mô hình phát triển lặp đi lặp lại, những khoảng lặp cố định thường kéo dài 1 đến 2 tuần được gọi là Sprint.

Scrum trong Kiểm thử phần mềm là một phương pháp để xây dựng các ứng dụng phần mềm phức tạp. Nó cung cấp các giải pháp dễ dàng để thực hiện các tác vụ phức tạp.

Scrum giúp nhóm phát triển tập trung vào mọi khía cạnh của quá trình phát triển sản phẩm phần mềm như chất lượng, hiệu suất, khả năng sử dụng, v.v. Nó cung cấp tính minh bạch, kiểm tra và thích ứng trong quá trình phát triển phần mềm để tránh sự phức tạp.

Thường phù hợp với những dự án có sự thay đổi về yêu cầu hoặc nghiệp vụ của khách hàng, làm theo giai đoạn ngắn, có thể nhìn thấy những rủi ro, hay những điểm chưa phù hợp để thay đổi.

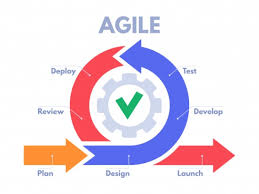
##### Phương pháp V-mode.



Hình 2.4. Mô hình V

Mô hình chữ V (V-Model) là một phần mở rộng của mô hình thác nước (Waterfall Model) và nó nhấn mạnh sự kiểm thử trong suốt quá trình phát triển phần mềm. Đây là một mô hình có tính kỷ luật cao, trong đó giai đoạn tiếp theo chỉ bắt đầu sau khi hoàn thành giai đoạn trước. Với V-Model, công việc kiểm thử được tham gia ngay từ đầu, giúp đảm bảo rằng mỗi giai đoạn phát triển đều được kiểm thử kỹ lưỡng trước khi chuyển sang giai đoạn kế tiếp. [11]

##### Phương pháp Agile



Hình 2.5. Mô hình Agile

Agile là một phương pháp phát triển phần mềm linh hoạt để làm sao đưa sản phẩm đến tay người dùng càng nhanh càng tốt và được xem như là sự cải tiến so với những mô hình cũ. [12]

Mô hình này được ứng dụng với bất kỳ loại hình dự án nào, nhưng cần sự tham gia và tính tương tác của khách hàng. Được sử dụng khi khách hàng yêu cầu chức năng sẵn sàng trong khoảng thời gian ngắn.

#### Một số loại kiểm thử  [13]

###### Kiểm thử chức năng ( Functionality Testing)

Kiểm thử chức năng được các Tester thực hiện để kiểm tra hoạt động của một chức năng nào đó có được hoạt động ổn định cũng như tốc độ phản hồi của chức năng có ổn định hay không.

###### Kiểm thử phi chức năng  (non-functional)

Kiểm thử phi chức năng được thực hiện nhằm trả lời câu hỏi: “Phần mềm có hoạt động tốt không?”. Kiểm thử phi chức năng chú trọng nhiều hơn vào những khía cạnh khác của phần mềm, như là độ bảo mật và khả năng tải của phần mềm đó.

###### Kiểm thử cấu trúc

Kiểm thử cấu trúc thường được gọi là “kiểm thử hộp trắng“(white-box) hoặc “hộp thủy tinh” (glass-box) vì chúng quan tâm đến những gì đang xảy ra bên trong hộp.

Kiểm thử cấu trúc là kiểm thử dựa trên phân tích cấu trúc bên trong của thành phần hoặc hệ thống. Nó thường được sử dụng như một cách đo lường của kiểm thử, thông qua độ bao phủ của một tập hợp các yếu tố cấu trúc.

###### Kiểm thử thay đổi

Mục đích của kiểm thử thay đổi là để kiểm tra xem phần mềm có vận hành trơn tru sau những lần sửa lỗi hay không.

Kiểm thử thay đổi gồm 2 loại chính:

* Kiểm thử xác nhận: Thường sẽ diễn ra sau khi lỗi trong phần mềm đã được xác nhận và được sửa. Lúc này, vai trò của Kiểm thử xác nhận là để xem lỗi đã thực sự được sửa hay chưa. Các tester sẽ tiến hành bằng cách cho một input giống hệt ban đầu và test xem output có ra được như mong muốn hay không.
* Regression Testing (Kiểm thử hồi quy): Mục đích của kiểm thử hồi quy để xác nhận rằng các thay đổi trong phần mềm hoặc môi trường không gây ra bất lợi ngoài mong muốn và hệ thống vẫn đáp ứng các yêu cầu.

#### Test case

##### Khái niệm về test case

Test case ( Kịch bản kiểm thử ) là một tập hợp các hành động được thực thi để xác minh một chức năng, một hệ thống phần mềm có hoạt động đúng hay không. [10]

##### Hình thức kiểm thử phần mềm

***Kiểm thử thủ công:*** Là hình thức kiểm thử phần mềm trong đó các trường hợp kiểm thử được người kiểm thử thực hiện theo cách thủ công mà không sử dụng bất kỳ công cụ tự động nào. Có các giai đoạn khác nhau để kiểm thử thủ công như kiểm thử đơn vị, kiểm thử tích hợp, kiểm thử hệ thông và kiểm thử chấp nhận người dùng.

***Kiểm thử tự động:*** Là hình thức kiểm thử phần mềm thực hiện bằng cách sử dụng các công cụ phần mềm kiểm thử tự động đặc biệt để thực thi một trường hợp kiểm thử. Kiểm thử tự động được dùng để chạy lại các kịch bản kiểm thử mà đã được thực hiện một cách thủ công, nhanh chóng và lặp đi lặp lại.

#### Các kỹ thuật kiểm thử phần mềm

**Phân vùng tương đương:**

Là kỹ thuật chia đầu vào thành những nhóm tương đương nhau. Nếu 1 giá trị trong nhóm hoạt động đúng thì tất cả các giá trị trong nhóm đó cũng hoạt động đúng và ngược lại. Các giá trị đầu vào chia thành các vùng tương đương sau:

* Vùng tương đương hợp lệ: Tập hợp các giá trị kiểm thử thỏa mãn điều kiện của hệ thống.
* Vùng tương đương không hợp lệ: Tập hợp các giá trị kiểm thử mô tả các trạng thái khác nhau của hệ thống: Sai, thiếu, không đúng, để trống.

Thiết kế test case bằng phân vùng tương đương tiến hành theo 2 bước:

* Xác định các lớp tương đương.
* Xây dựng các ca kiểm thử tương ứng

Khi thực hiện kỹ thuật phân vùng tương đương, đầu vào sẽ được chia theo nguyên tắc:

* 1 lớp các giá trị lớn hơn
* 1 lớp các giá trị nhỏ hơn
* 1 lớp các giá trị hợp lệ

**Phân tích giá trị biên:**

Là phương pháp kiểm thử tất cả các giá trị ở vùng biên của dữ liệu vào và dữ liệu ra. Chúng ta sẽ tập trung vào các giá trị biên chứ không kiểm thử toàn bộ dữ liệu. Khi áp dụng kỹ thuật phân tích giá trị biên, người kiểm thử sẽ chọn các giá trị:

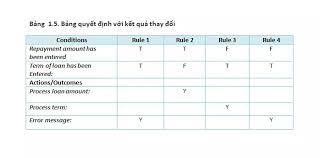
* Giá trị nhỏ nhất
* Giá trị ngay dưới giá trị nhỏ nhất
* Giá trị bình thường
* Giá trị lớn nhất
* Giá trị ngay trên giá trị lớn nhất

**Bảng quyết định:**

Đối với kiểm thử hành vi của hệ thống với nhiều trường dữ liệu, bảng quyết định sẽ giúp chúng ta phân loại và định hình được kịch bản kiểm thử một cách chính xác và rõ ràng hơn.

Có 4 bước để người kiểm thử tạo được bảng quyết định:

* Liệt kê tất cả Conditions/Inputs.
* Tính số lượng kết hợp có thể (Rules).
* Đặt tất cả các kết hợp trong bảng.
* Giảm thiểu các case kết hợp và quyết định Test Case



Hình 2.6. Bảng quyết định

#### Phương pháp kiểm thử phần mềm

##### Kiểm thử hộp đen

Là phương pháp kiểm thử mà người kiểm thử không cần quan tâm đến các hoạt động bên trong hệ thống chạy ra sao, không cần quan tâm đến các dòng lệnh bên trong hệ thống như thế nào mà chỉ cần tập trung và các giá trị đầu vào và các giá trị đầu ra của hệ thống có đúng với kết quả mong đợi của các trường hợp kiểm thử không để từ đó đánh giá chất lượng hệ thống.

##### Kiểm thử hộp trắng

Kiểm thử hộp trắng (được biết đến như là kiểm thử tính rõ ràng của hộp, kiểm thử hộp kính, kiểm thử hộp trong suốt và kiểm thử cấu trúc) giúp kiểm thử được cấu trúc nội bộ hoặc hoạt động của một chương trình, như tương phản với chức năng được bộc lộ của người dùng cuối.

##### Kiểm thử hộp xám

Ngoài 2 phương pháp đã được nhắc đến: Black box test và White box test, thì có  Gray box test là sự kết hợp giữa Black box test và White box test.

Kiểm thử hộp xám là một phương pháp kiểm thử phần mềm để kiểm tra một sản  phẩm hoặc ứng dụng phần mềm với một phần kiến thức về cấu trúc bên trong của ứng  dụng. Mục đích của kiểm thử hộp xám là tìm kiếm và xác định các lỗi do cấu trúc mã không  đúng hoặc sử dụng ứng dụng không đúng cách.

Muốn kiểm thử hộp xám, các Tester phải có kiến thức cơ bản về các luồng hoạt động bên trong hệ thống. Khi kiểm tra hộp xám, Tester có thể truy cập vào các tài liệu thiết kế hệ thống và cơ sở dữ liệu của hệ thống. Trong kiểm thử hộp xám, Tester chỉ nhìn thấy được một phần bên trong hệ thống. Vì hộp xám trong kiểm thử được thiết kế kiểu bán trong suốt, nhìn từ ngoài vào chỉ thấy một phần bên trong.

Các kỹ thuật kiểm thử hộp xám

* Kiểm thử ma trận (Matrix Testing).
* Kiểm thử hồi quy (Regression Testing).
* Kiểm thử mảng trực giao (Orthogonal Array Testing or OAT).
* Kiểm thử mẫu (Pattern Testing).

#### Error/ Fault/Failure/Bug/Defect

* Error: Là hành động của con người dẫn đến kết quả sai.
* Fault: Là trạng thái phần mềm gây ra bởi Error.
* Failure: Lỗi khi có kết quả sai lệch so với yêu cầu đặc tả, là sự khác biệt giữa kết quả thực tế trên màn hình và kết quả mong đợi của một thành phần hoặc một hệ thống nào đó.
* Bug: Là một khiếm khuyết trong một thành phần hoặc hệ thống mà nó có thể làm cho thành phần hoặc hệ thống này không thực hiện đúng chức năng yêu cầu của nó.
* Defect: Là lỗi trong quá trình phát triển hoặc lỗi logic làm cho chương trình hoạt động sai yêu cầu đề ra.

#### Quy trình quản lí lỗi. [15]



Hình 2.7. Quy trình quản lí lỗi

**Khám phá:**

Trong giai đoạn khám phá, các nhóm dự án phải phát hiện ra càng nhiều lỗi càng tốt trước khi khách hàng cuối có thể phát hiện ra. Một lỗi được cho là được phát hiện và thay đổi thành trạng thái "accepted" khi nó được các developer thừa nhận và chấp nhận.

**Phân loại:**

Phân loại lỗi giúp các nhà phát triển ưu tiên các nhiệm vụ của họ, giúp các nhà phát triển ưu tiên sửa được những lỗi quan trọng trước tiên.

Lỗi thường được phân loại bởi Test manager:

* Mức độ nghiêm trọng:
* Critical
* Major
* Medium
* Low
* Mức độ ưu tiên:
* High
* Medium
* Low

**Giải pháp:**

Sau khi lỗi được phát hiện và phân loại thành công thì sẽ chuyển sang giải quyết lỗi. Giải quyết lỗi trong kiểm thử phần mềm là quá trình từng bước sửa lỗi. Quá trình giải quyết lỗi bắt đầu bằng việc gán lỗi cho nhà phát triển, sau đó nhà phát triển lên lịch sửa lỗi theo mức độ ưu tiên, sau đó lỗi được sửa và cuối cùng nhà phát triển gửi báo cáo giải pháp cho người quản lý kiểm thử. Quá trình này giúp sửa chữa và theo dõi lỗi một cách dễ dàng. Có thể theo các bước sau để khắc phục lỗi:

* Assignment (Giao việc)
* Schedule fixing (Lên lịch sửa chữa)
* Fix the defect ( Sữa lỗi)
* Report the resolution (Báo cáo giải pháp)

**Xác minh:**

Sau khi nhà phát triển đã sửa lỗi và báo cáo lỗi, nhóm Tester xác minh rằng các lỗi thực sự được giải quyết..

**Đóng lỗi:**

Khi lỗi đã được sửa và xác minh, lỗi đó được thay đổi trạng thái là closed. Nếu không, bạn cần gửi thông báo cho developer để kiểm tra lại lỗi.

**Báo cáo lỗi:**

Báo cáo lỗi là một quá trình trong đó người quản lý kiểm thử chuẩn bị và gửi báo cáo lỗi cho nhóm quản lý để lấy phản hồi về quy trình quản lý lỗi và trạng thái lỗi.

### GIỚI THIỆU VỀ WEBSITE SCTV

#### Giới thiệu web SCTV

**Web SCTV** là dịch vụ trực tuyến của SCTV, cho phép người dùng xem TV trực tuyến, video, các show truyền hình, phim bộ và nhiều nội dung khác ngay trên web. Dịch vụ này cung cấp đa dạng kênh truyền hình, phim ảnh và các chương trình giải trí với chất lượng tuyệt đỉnh trên nhiều nền tảng, mang đến trải nghiệm xem linh hoạt và tiện lợi.

**Tính năng chính của hệ thống:**

* Đăng nhập, Đăng xuất, Tạo tài khoản.
* Xem phim, truyền hình miễn phí và trả phí.
* Phương thức thanh toán.

**Tác nhân liên quan đến hệ thống:**

User: Là người sử dụng website SCTV. Người dùng có hoặc không có tài khoản để truy cập và sử dụng các tính năng của trang web. Người dùng được thực hiện những chức năng sau:

* Đăng kí, Đăng nhập, Đăng xuất
* Mua gói bằng nhiều phương thức thanh toán
* Thêm vào yêu thích
* Xem video free
* Xem video vip
* Tìm kiếm, lọc
* Quản lí tài khoản

Trong bài báo cáo này, em tiến hành lập kế hoạch và thực hiện kiểm thử test case cho các tính năng “Đăng ký”, “Đăng nhập”, “Phương thức thanh toán”.

#### Sơ đồ use case tổng quát



Hình 3.1. Use case tổng quát hệ thống

#### Đặc tả, Phân tích Use case cho “Đăng Ký”, “Đăng nhập”, Phương thức thanh toán ”

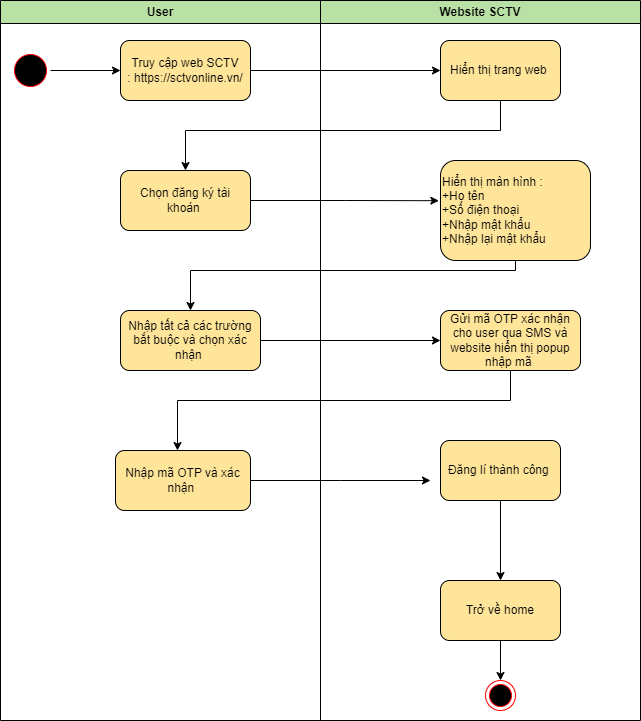
##### Use case đăng ký

###### Đăng ký tài khoản

|  |  |
| --- | --- |
| Use case name | Đăng ký |
| Description | Là người dùng, tôi muốn đăng ký tài khoản để vào web xem chương trình. |
| Actors | Người dùng |
| Priority | High |
| Triggers | Người dùng chọn chức năng đăng ký |
| Pre-conditions |  |
| Post-conditions |  |
| Main flow | 1. Truy cập Web SCTV 2. Hiển thị trang web 3. Chọn đăng ký tài khoản 4. Hiển thị màn hình : + Họ tên   + Số điện thoại  + Nhập mật khẩu  + Nhập lại mật khẩu   1. Nhập tất cả các trường bắt buộc và chọn xác nhận 2. Gửi mã OTP xác nhận về cho user qua SMS và website hiển thị popup nhập mã 3. Nhập mã OTP và xác nhận 4. Đăng ký thành công 5. Trở về Home |
| Alternative flows | N/A |
| Exception flows | N/A |
| Business rules | N/A |
| Non-functional requirements | N/A |

Bảng 1. Đăc tả đăng ký tài khoản

- Sơ đồ hoạt động :



Hình 3.2. Sơ đồ hoạt động " Đăng ký tài khoản"



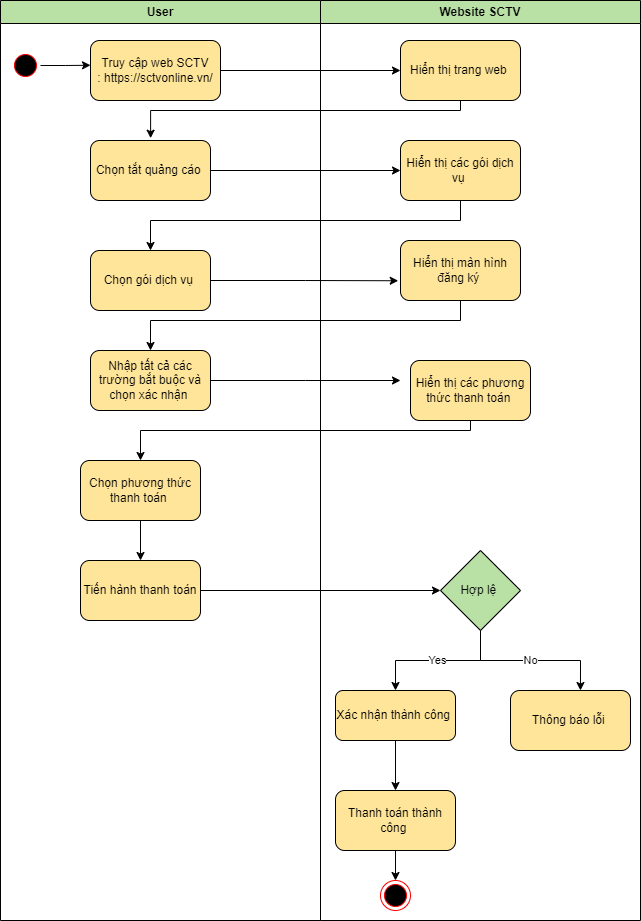
Hình 3.3. Màn hình Đăng Ký Tài khoản

###### Đăng ký Các gói dịch vụ

|  |  |
| --- | --- |
| Use case name | Đăng ký |
| Description | Là người dùng, tôi muốn đăng ký để vào web mua các gói dịch vụ |
| Actors | Người dùng |
| Priority | High |
| Triggers | Người dùng chọn chức năng đăng ký để mua gói |
| Pre-conditions |  |
| Post-conditions |  |
| Main flow | 1. Truy cập Web SCTV 2. Hiển thị trang web 3. Chọn tắt quảng cáo 4. Hiển thị các gói dịch vụ 5. Chọn gói dịch vụ 6. Hiển thị màn hình đăng ký 7. Nhập tất cả các trường bắt buộc và chọn xác nhận 8. Hiển thị các phương thức thanh toán 9. Chọn phương thức thanh toán 10. Tiến hành thanh toán 11. Thanh toán thành công |
| Alternative flows | N/A |
| Exception flows | 10.a Nếu thanh toán không hợp lệ thì hệ thống thông báo “Lỗi” mình quay lại bước 8 |
| Business rules | N/A |
| Non-functional requirements | N/A |

Bảng 2. Đắc tả đăng ký gói dịch vụ

**Sơ đồ hoạt động :**

****

Hình 3.4. Sơ đồ hoạt động "Đăng ký cho các gói dịch vụ".



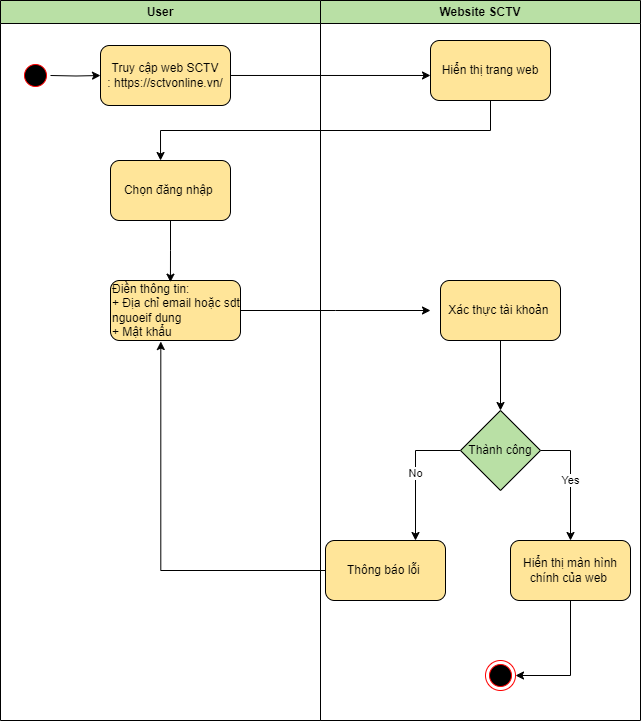
Hình 3.5. Màn hình đăng ký các gói dịch vụ

##### Use case “ Đăng nhập ”

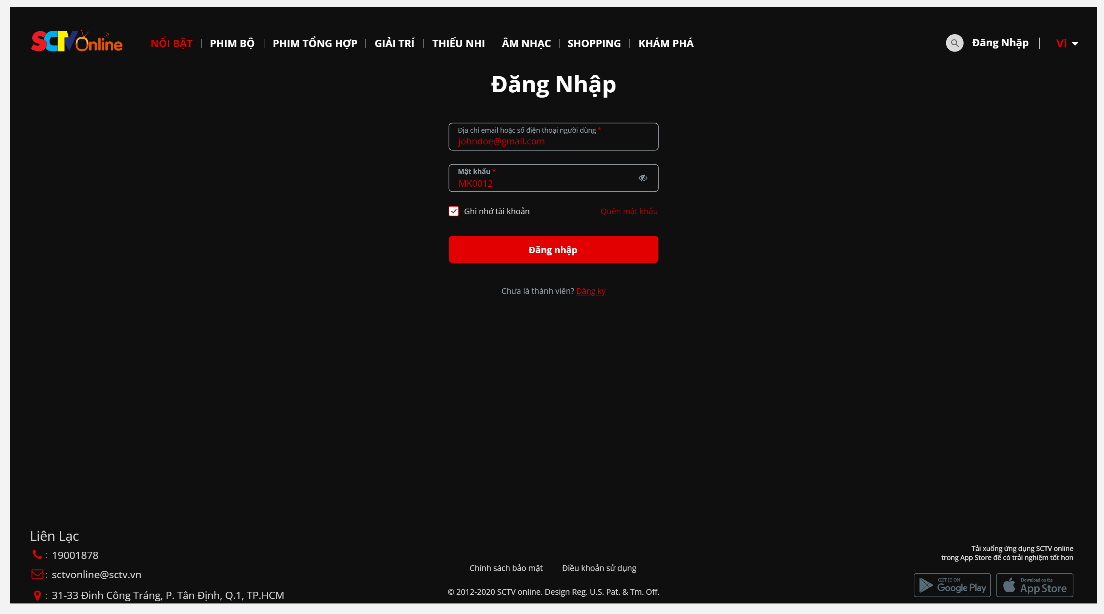
|  |  |
| --- | --- |
| Use case name | Đăng nhập |
| Description | Là người dùng, tôi muốn đăng nhập để vào web xem chương trình |
| Actors | Người dùng |
| Priority | High |
| Triggers | Người dùng chọn chức năng đăng nhập |
| Pre-conditions | Đăng ký thành công |
| Post-conditions | Đăng nhập được vào hệ thống |
| Main flow | 1. Truy cập Web SCTV 2. Hiển thị trang web 3. Chọn Đăng nhập 4. Điền thông tin : + Địa chỉ email hoặc sdt người dùng   + Mật khẩu   1. Xác thực tài khoản 2. Hiển thị màn hình của web |
| Alternative flows | N/A |
| Exception flows | 5a. Nếu xác thực tài khoản không thành công thì thông báo “Lỗi” và quay lại bước 4 |
| Business rules | N/A |
| Non-functional requirements | N/A |

Bảng 3. Đặc tả đăng nhập

**Sơ đồ hoạt động:**



Hình .. Sơ đồ hoạt động "Đăng nhập"



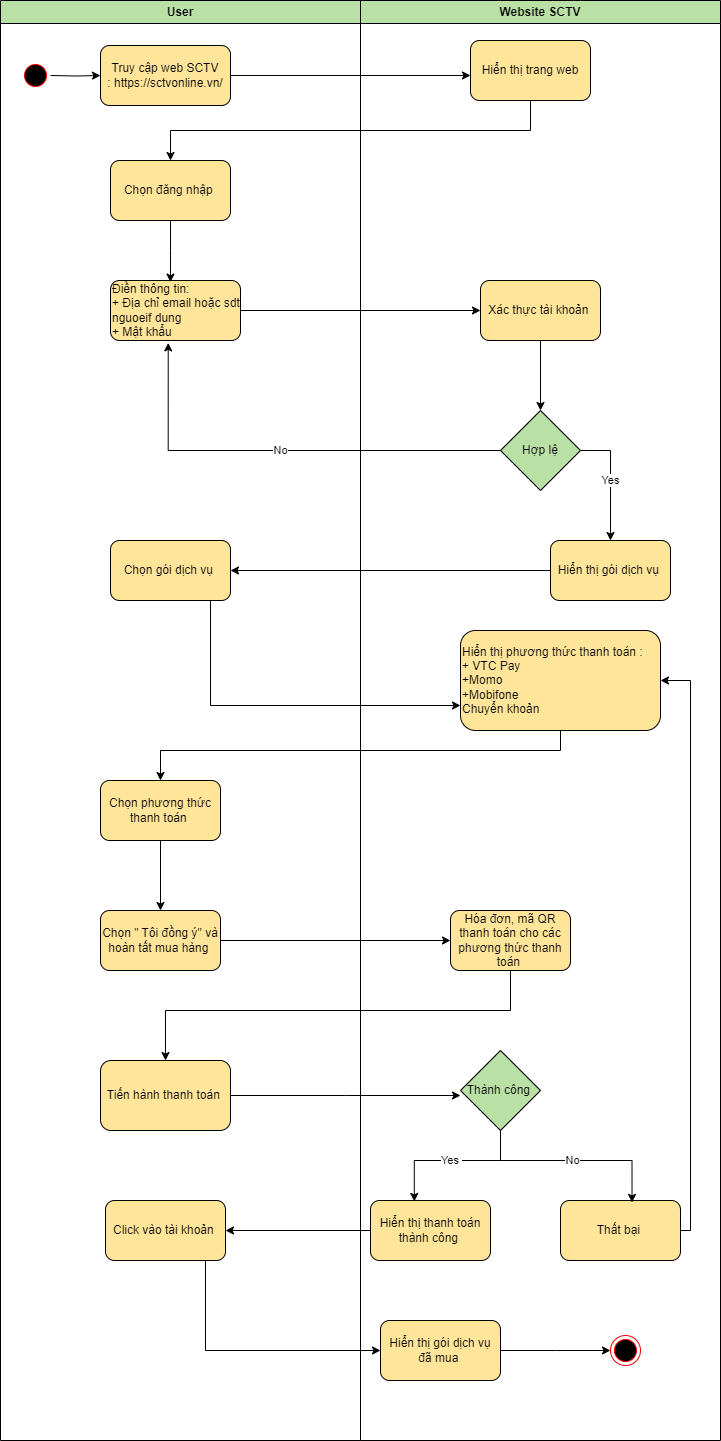
Hình .. Màn hình đăng nhập

##### Use case “Phương thức thanh toán ”

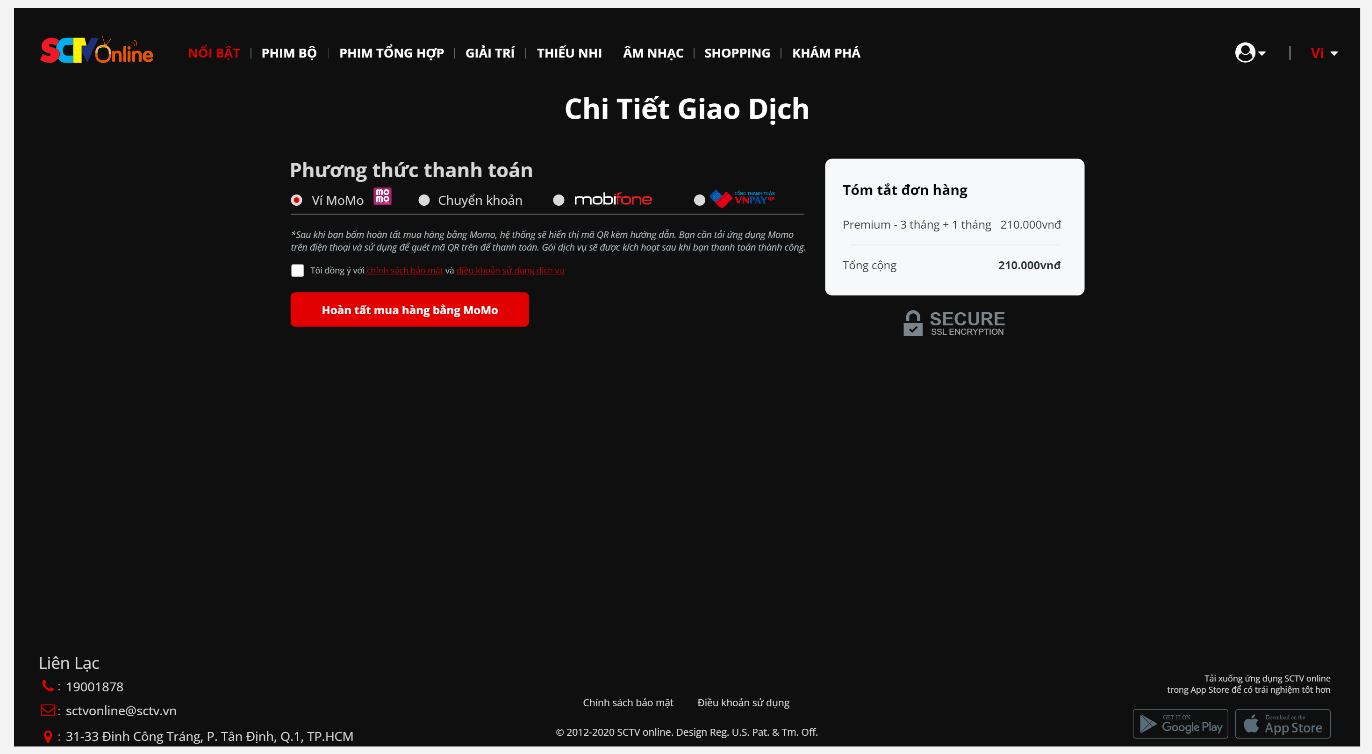
|  |  |
| --- | --- |
| Use case name | Phương thức thanh toán |
| Description | Là người dùng, tôi muốn thực hiện thanh toán dịch vụ SCTV để tôi có thể kích hoạt các gói VIP/PREMIUM để xem các chương trình. |
| Actors | Người dùng/ hệ thống |
| Priority | High |
| Triggers | Người dùng chọn chức năng Đăng Nhập |
| Pre-conditions | Người dùng đã đăng ký tài khoản trên SCTV. |
| Post-conditions | Gói dịch vụ VIP/ PREMIUM được kích hoạt trên tài khoản người dùng sau khi xác nhận thanh toán thành công |
| Main flow | 1. Truy cập Web SCTV 2. Hiển thị trang web 3. Chọn Đăng nhập 4. Điền thông tin 5. Xác thực tài khoản 6. Hiển thị gói dịch vụ 7. Chọn gói dịch vụ 8. Hiển thị phương thức thanh toán:   + VTC Pay  + Momo  + Mobifone  + Chuyển khoản   1. Chọn phương thức thanh toán 2. Chọn “ Tôi đồng ý” và hoàn tất mua hàng 3. Hóa đơn, mã QR thanh toán cho các phương thức thanh toán 4. Tiến hành thanh toán 5. Hiển thị thanh tóan thành công 6. Click vào tài khoản 7. Hiển thị gói dịch vụ đã mua |
| Alternative flows | N/A |
| Exception flows | 5a. Nếu xác thực tài khoản không hợp lệ thì mình quay lại bước 4  11a. Nếu thanh toán không thành công hiển thị thông báo “ Thất bại” rồi mình quay lại bước 8 |
| Business rules | N/A |
| Non-functional requirements | N/A |

Bảng . Đặc tả phương thức thanh toán

**Sơ đồ hoạt động :**



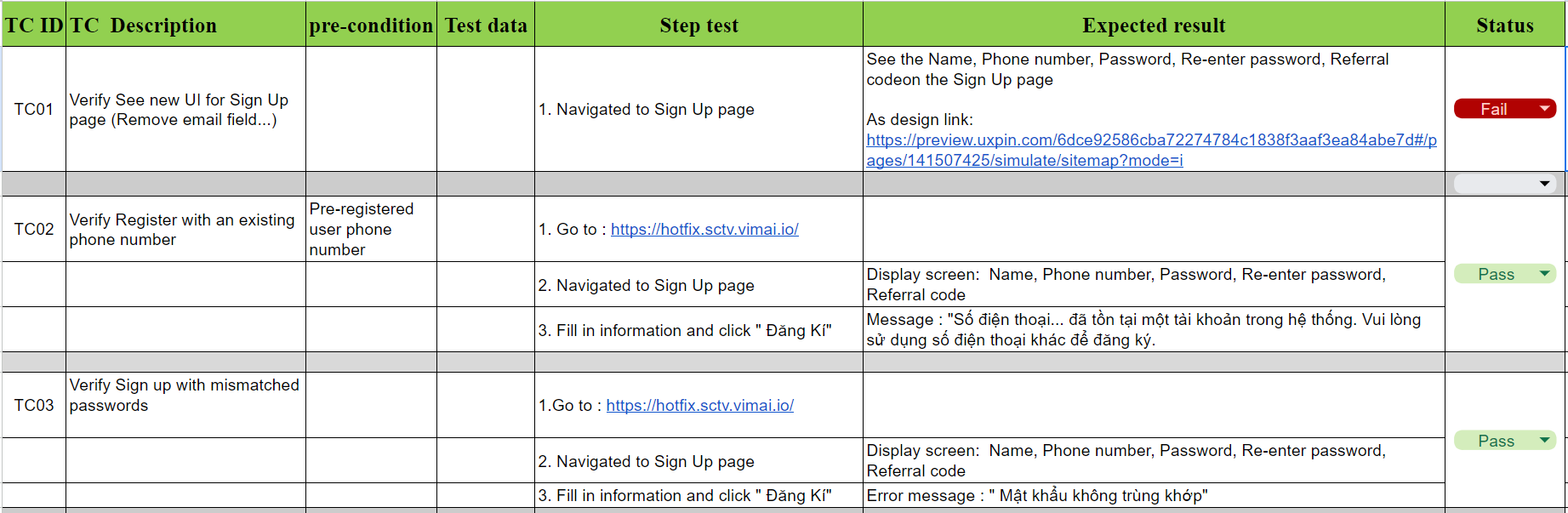
Hình .. Sơ đồ hoạt động" Phương thức thanh toán"



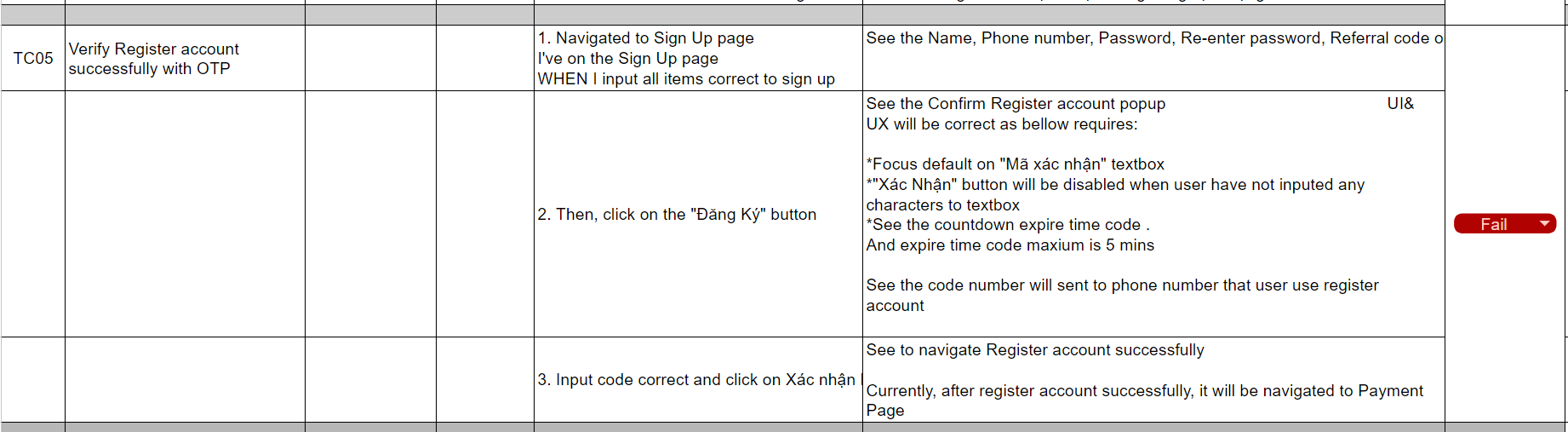
Hình .. Màn hình phương thức thanh toán

### TRIỂN KHAI VÀ THỰC HIỆN TEST CASE

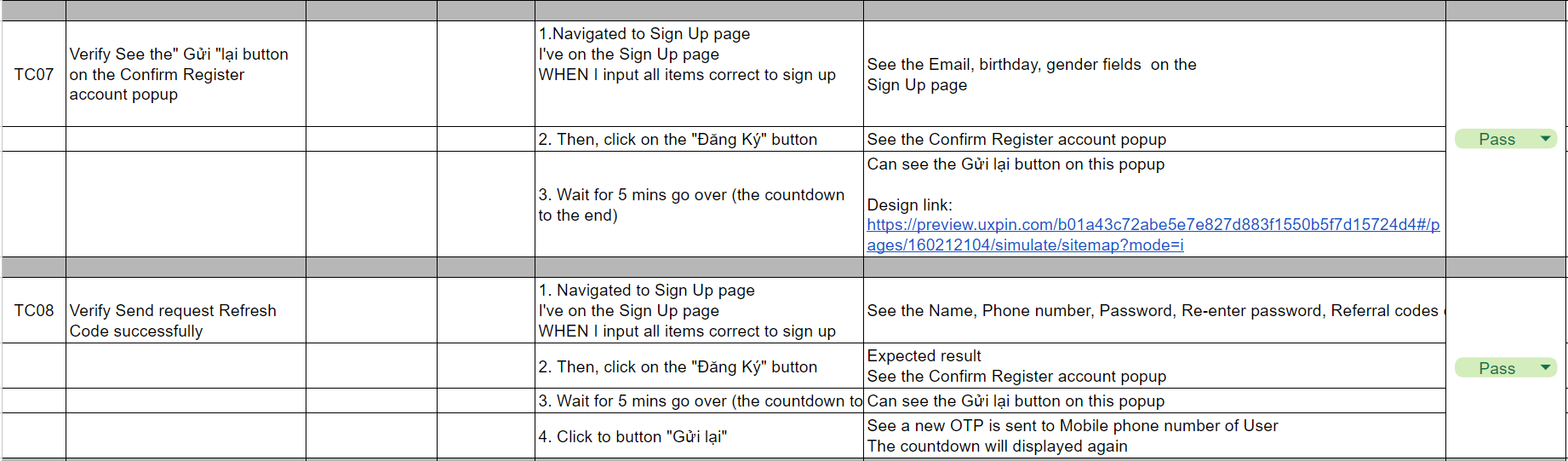
##### Test case cho chức năng 'Đăng ký và Đăng nhập cho WEB SCTV"

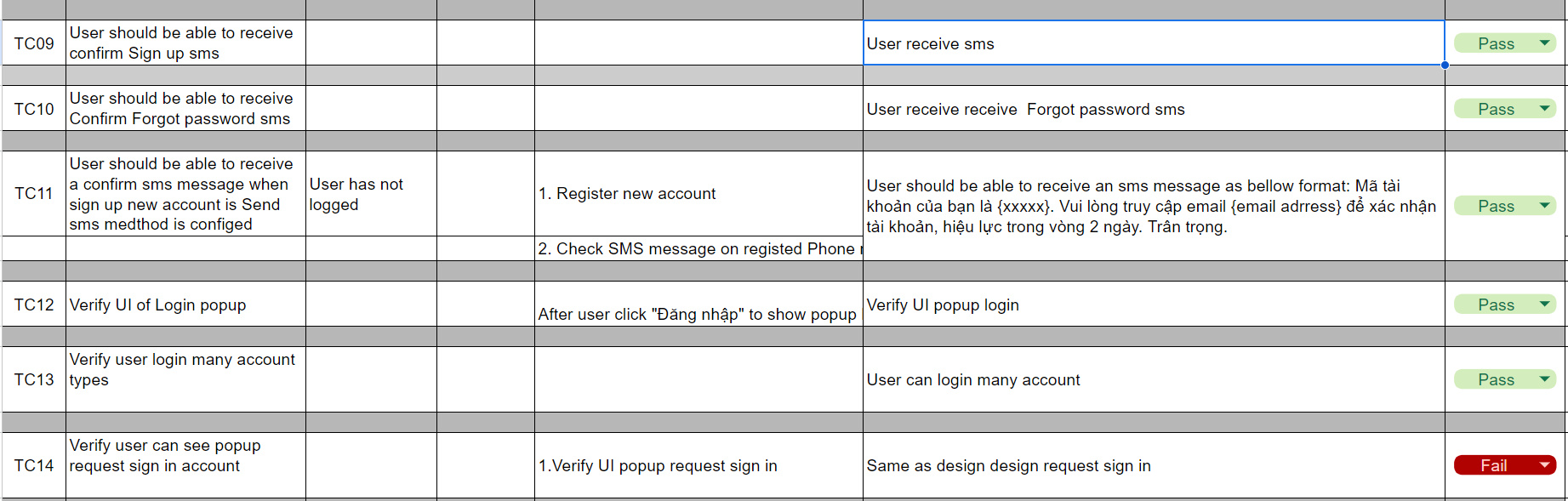


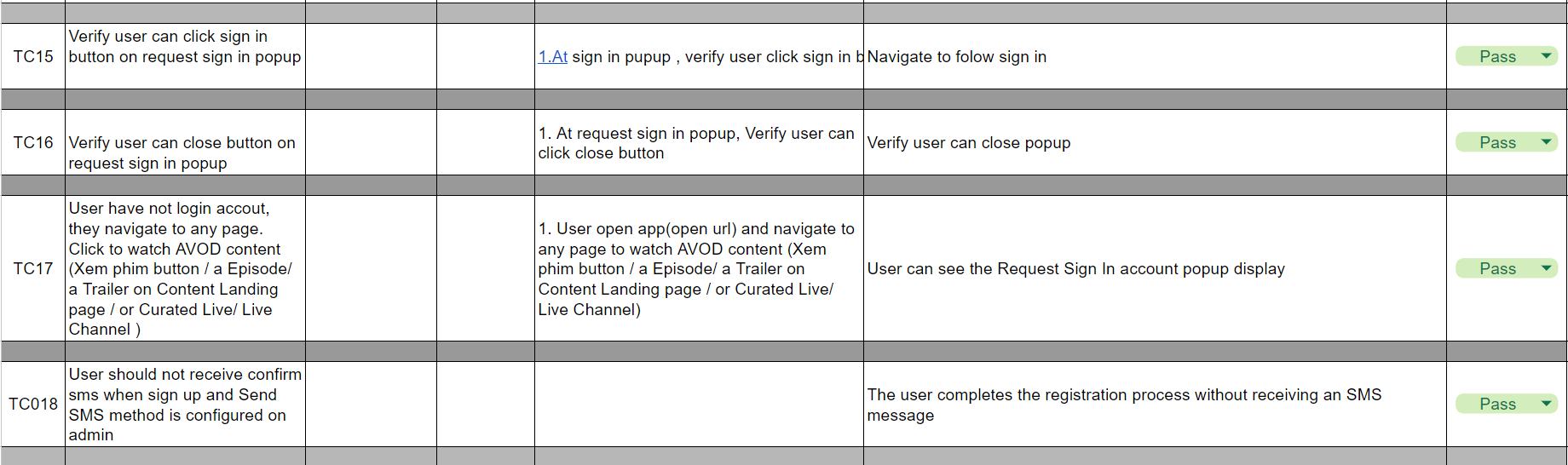


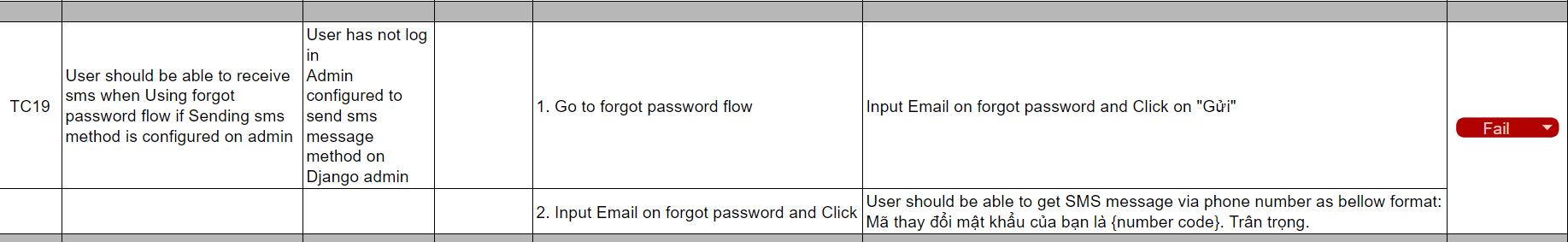




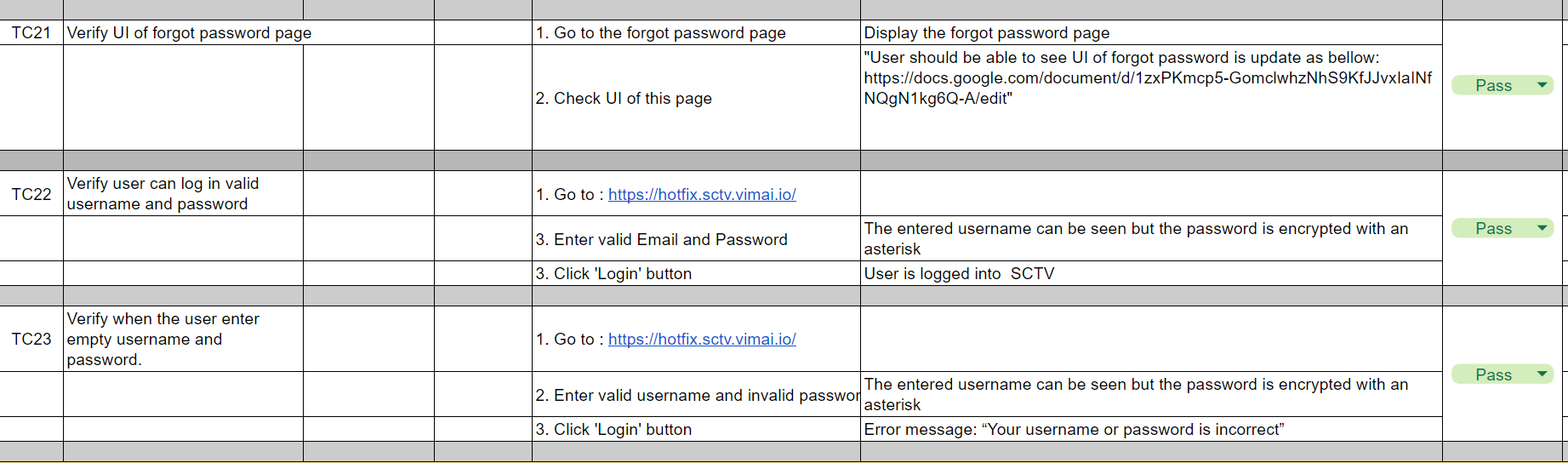






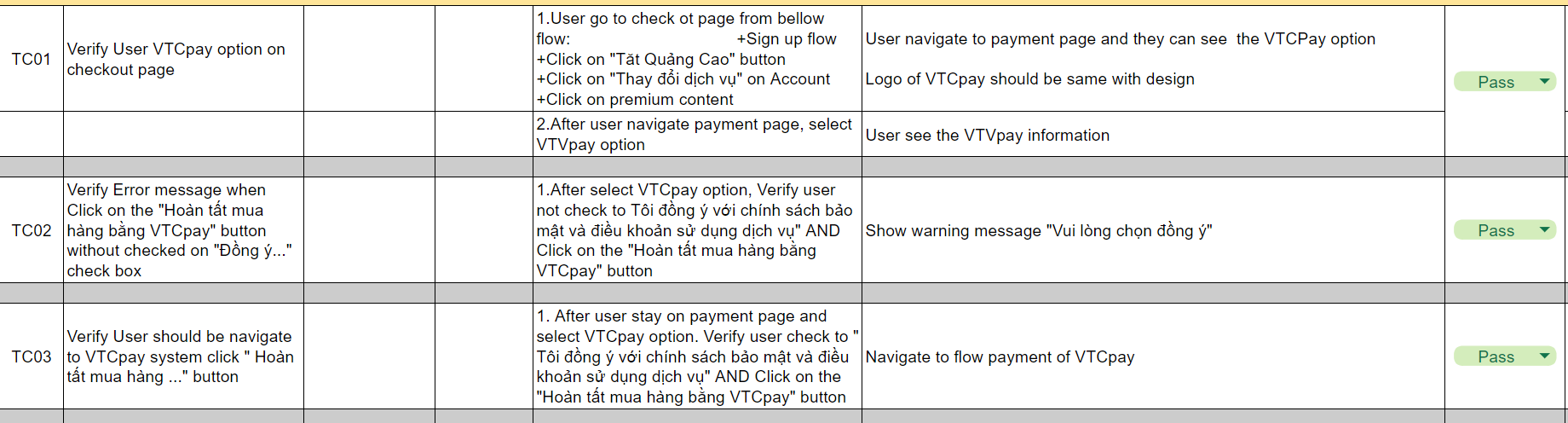


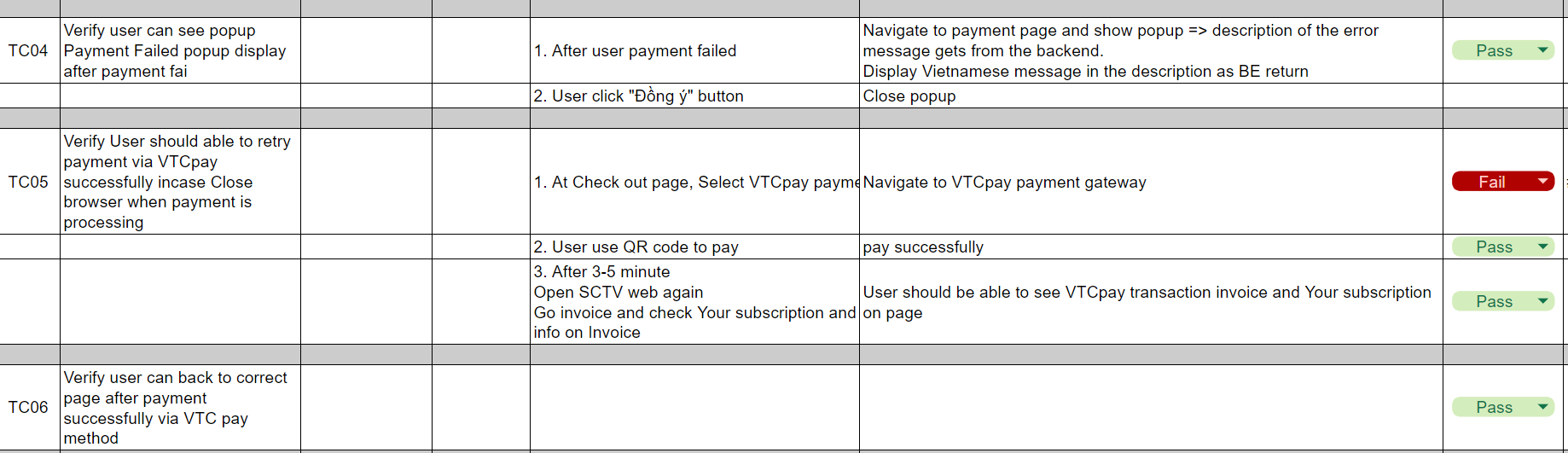


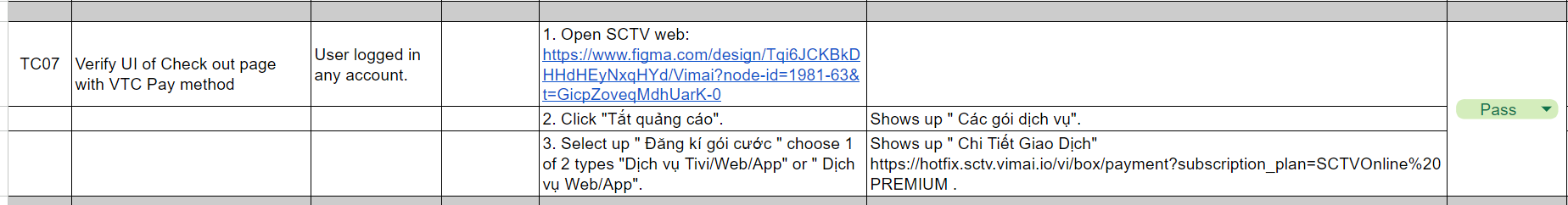


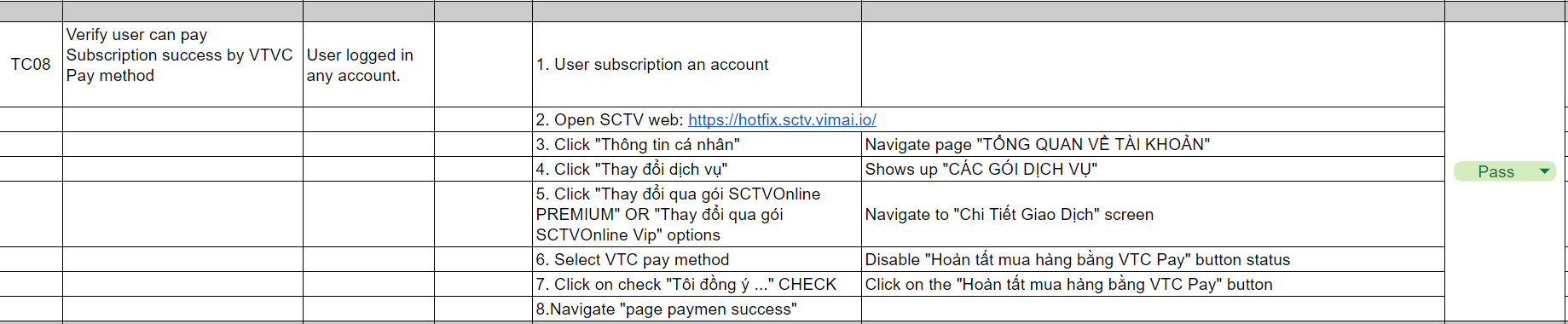
Hình 4.1. Thiết kế Test case “Đăng ký, Đăng nhập”

##### Test case cho chức năng " Phương Thức Thanh Toán cho WEB SCTV"

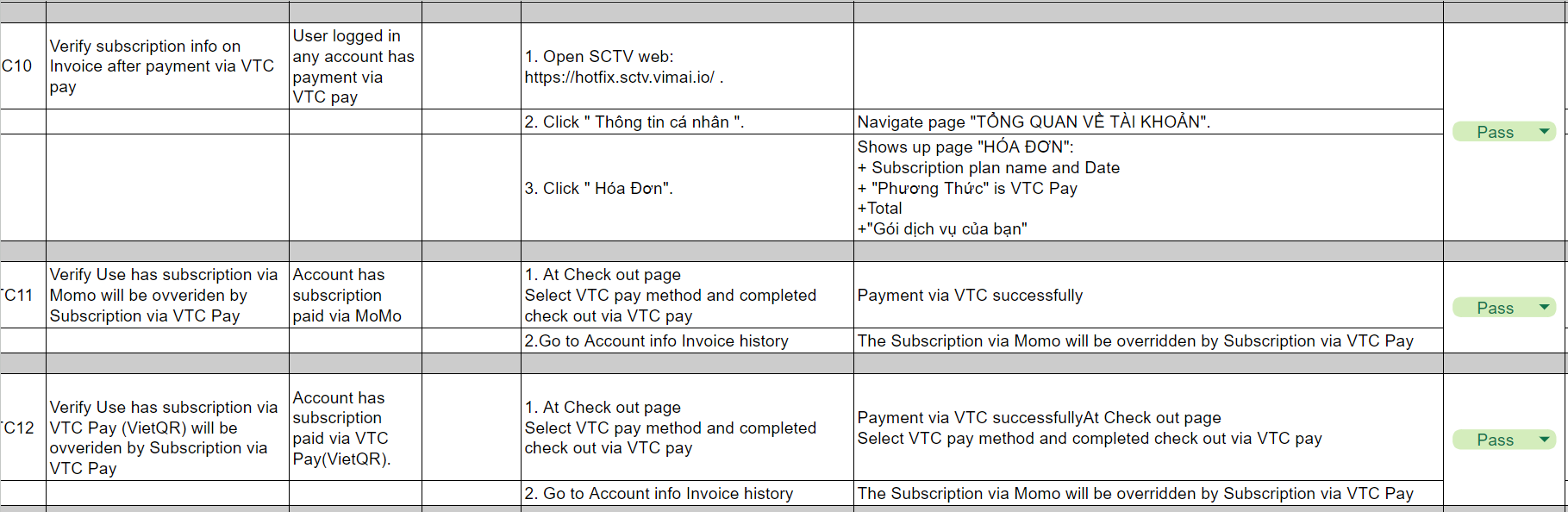


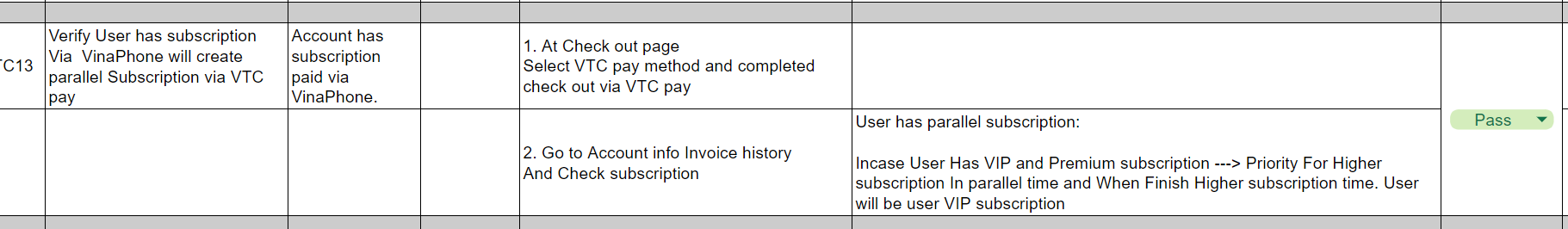


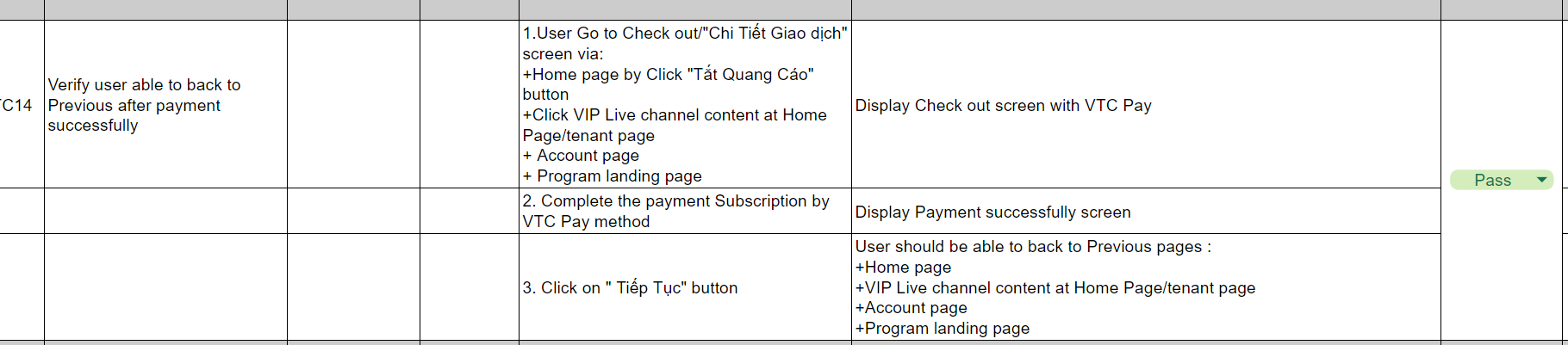


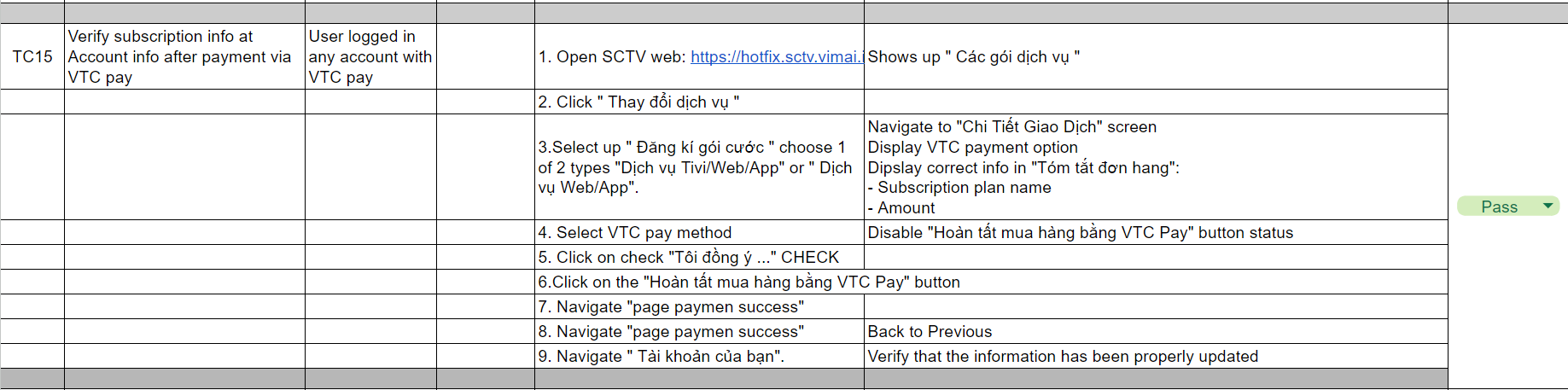


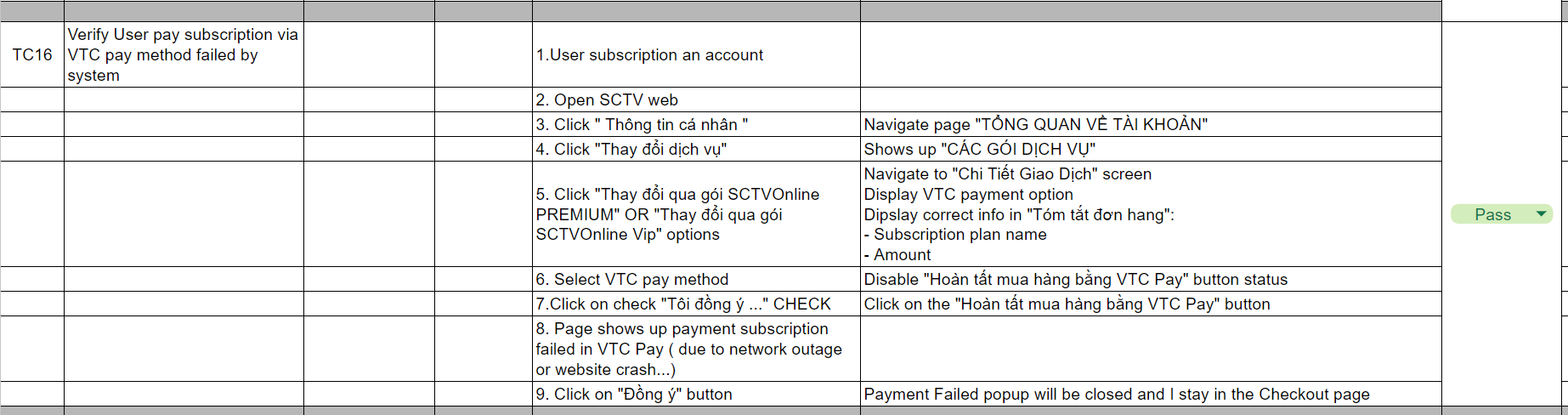


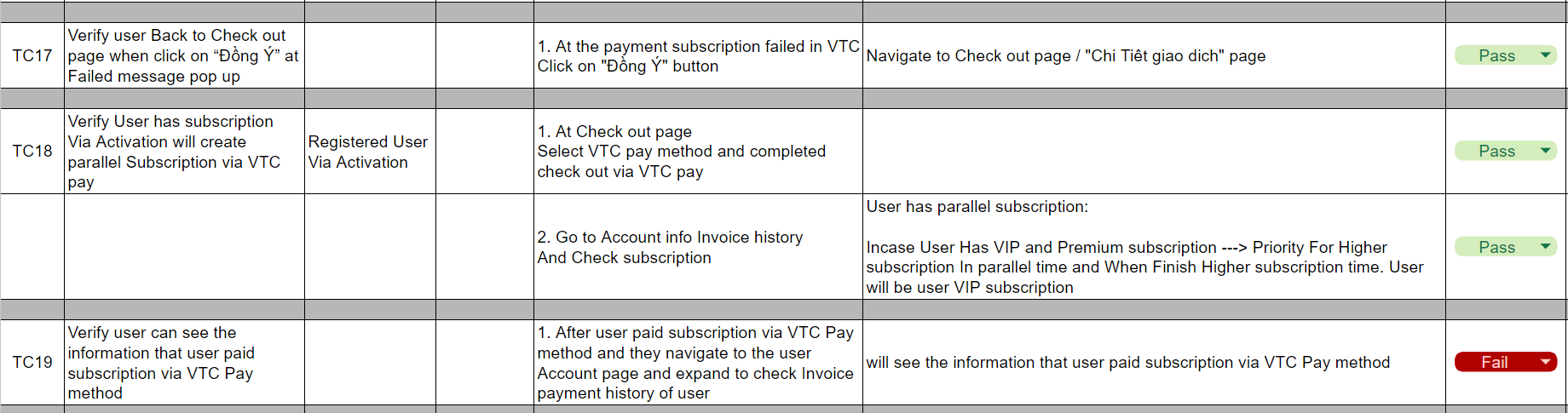


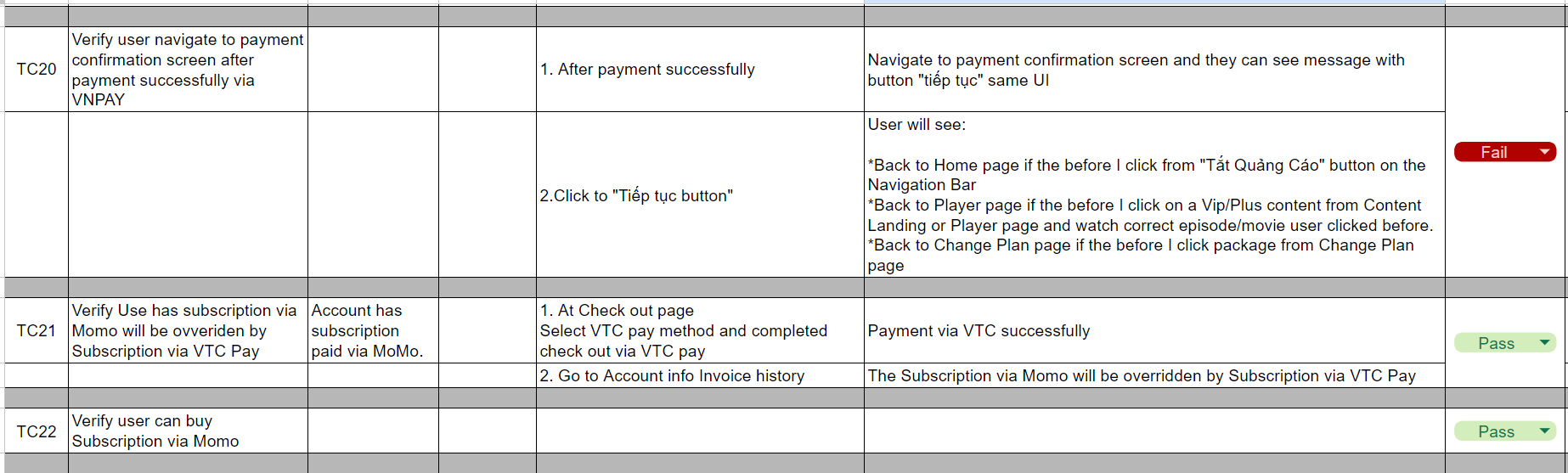


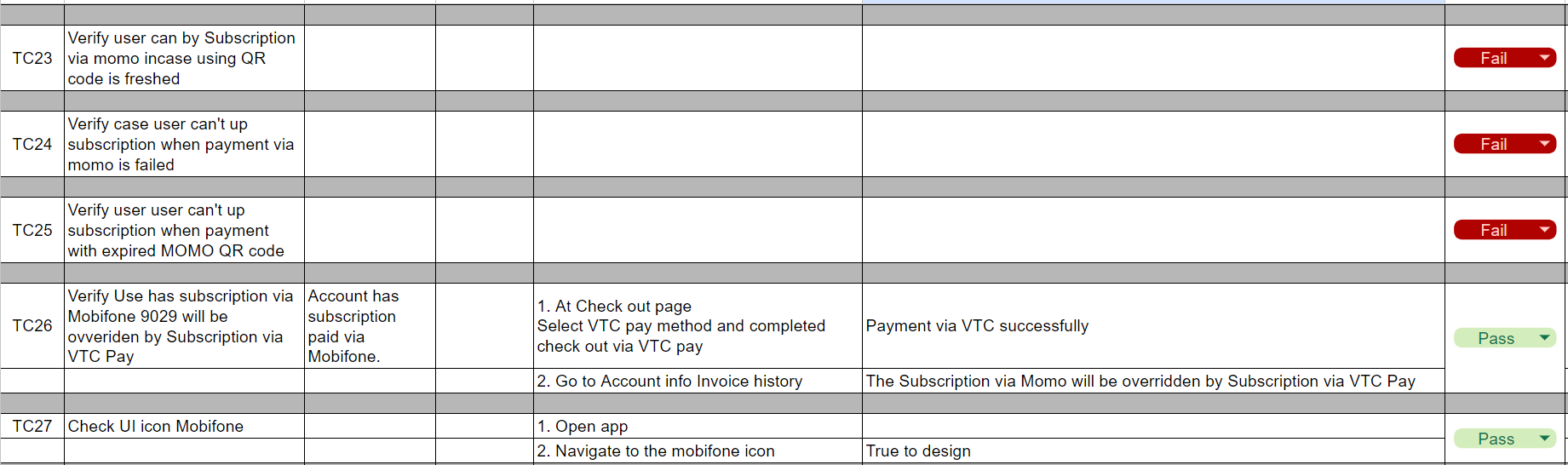
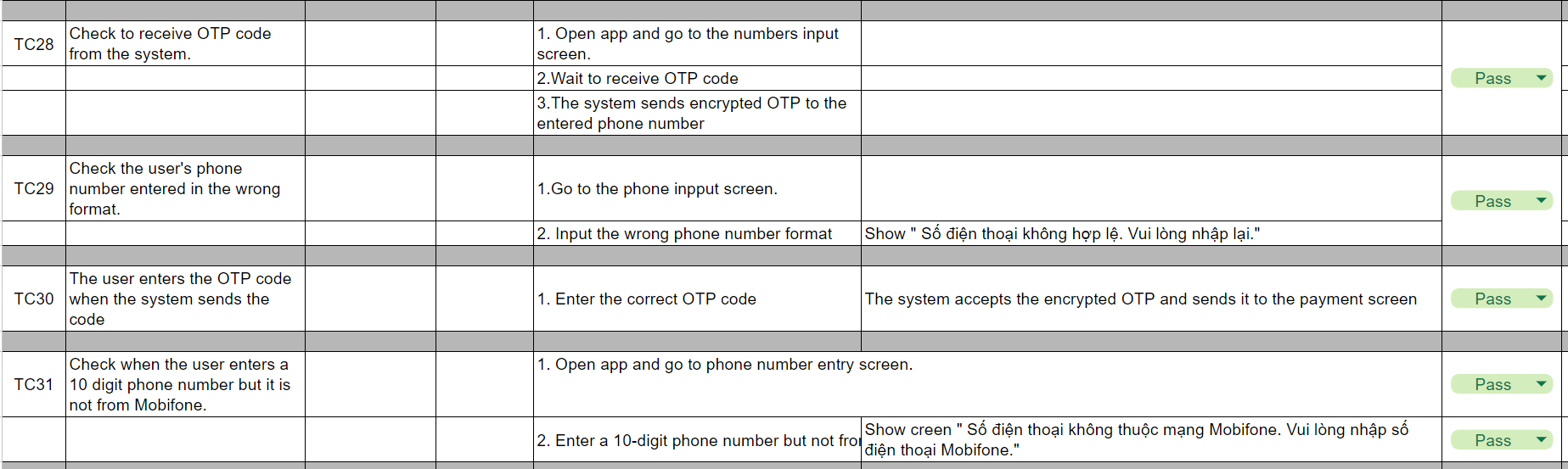


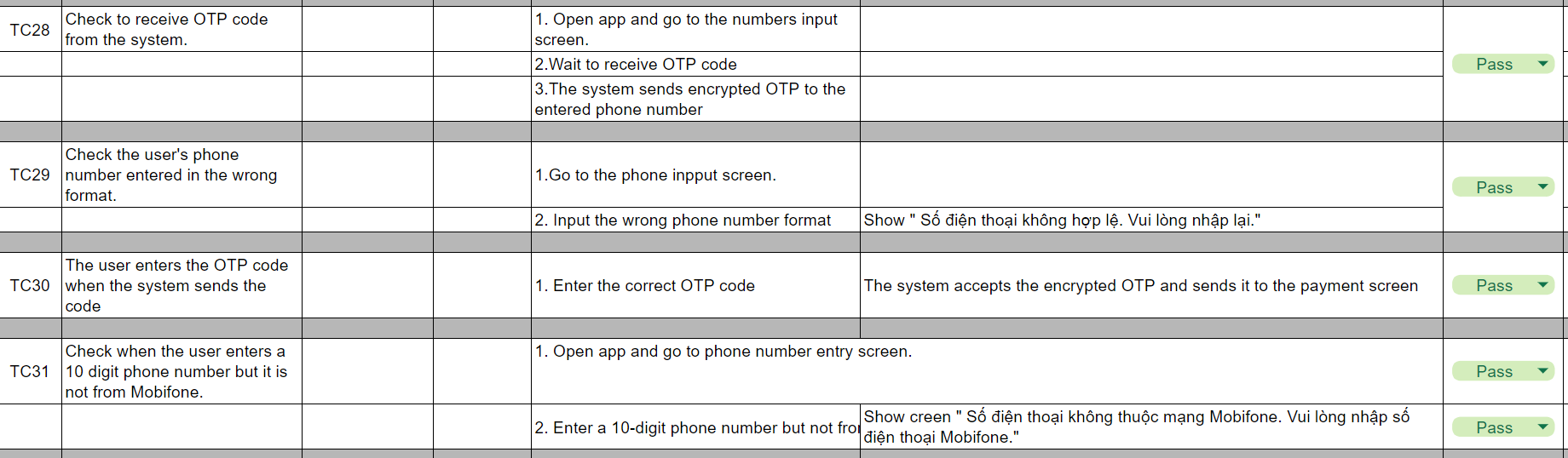


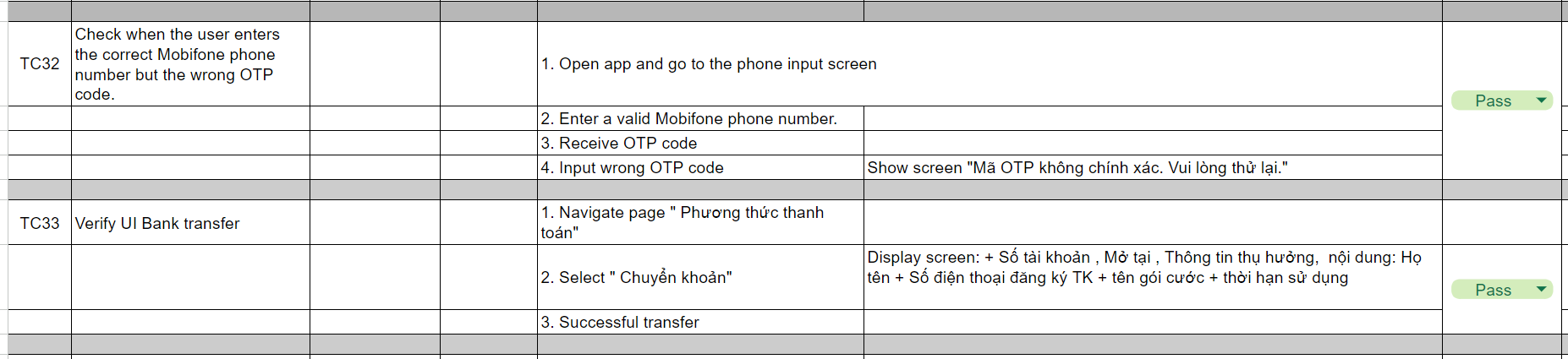












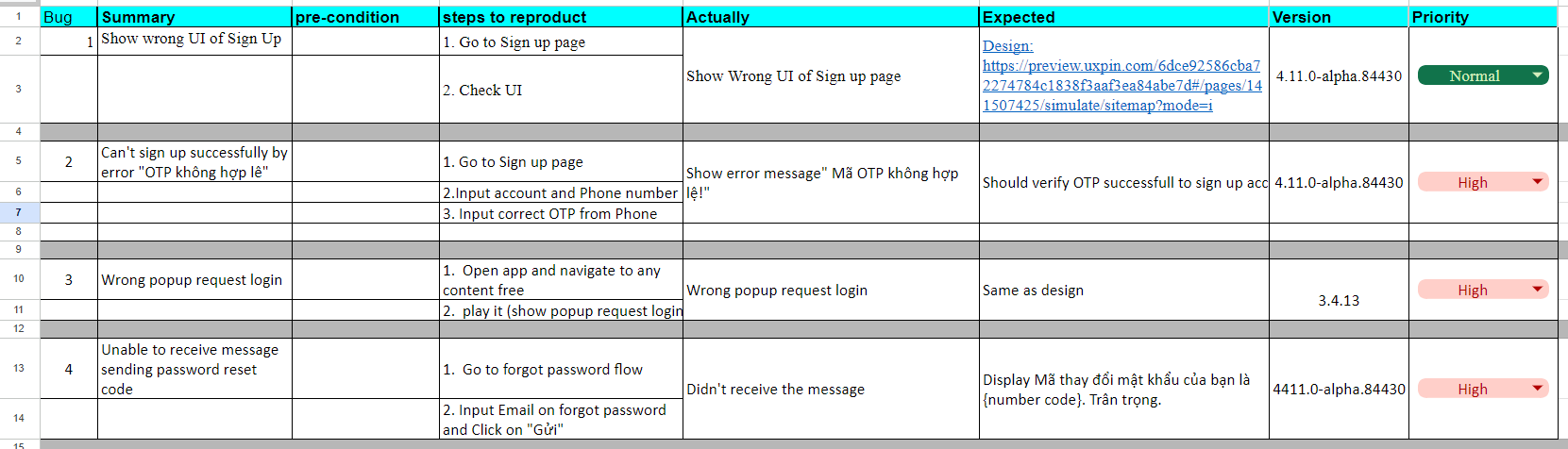
Hình 4.2. Thiết kế test case " Phương thức thanh toán"

### KẾT QUẢ THỰC HIỆN TEST WEB

#### Kết quả kiểm thử cho chức năng

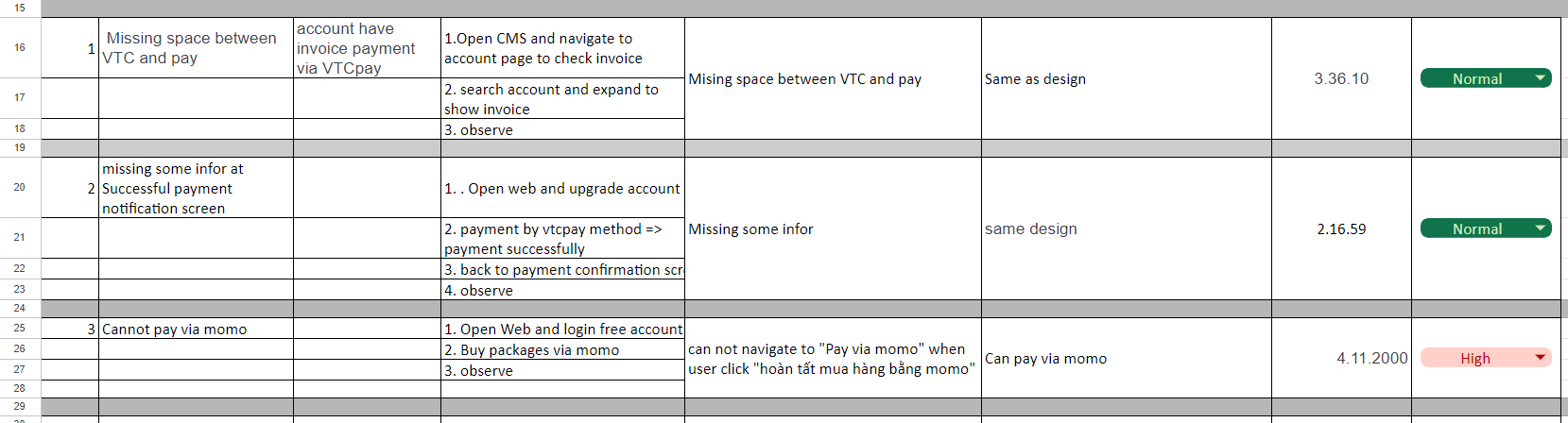
##### Kết quả kiểm thử cho chức năng “Đăng Ký”, “Đăng Nhập”, “Phương Thức Thanh Toán ” cho WEB SCTV

**Đăng ký, Đăng nhập**



Hình 5.1. Bug report “Đăng ký, Đăng nhập”

**Phương thức thanh toán**

****

Hình 5.2. Bug report “Phương thức thanh toán”

#### Bảng kết quả kiểm thử

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Total Test case | Pass | Fail | N/A | Block | Skip |
| Đăng ký, Đăng nhập | 23 | 19 | 6 | 0 | 0 | 0 |
| Phương thức thanh toán | 33 | 30 | 3 | 0 | 0 | 0 |

Bảng 5. .Bảng kết quả thực hiện kiểm thử

**Trạng thái của Test case:**

**Passed:** Nhập nội dung này nếu testcase đã được thực hiện và kết quả đã kiểm thử đúng như kết quả mong đợi.

**Failed**: Nhập nội dung này nếu testcase đã được thực hiện và kết quả đã kiểm thử không đúng như kết quả mong đợi.

**N/A**: Nhập nội dung này nếu test case thuộc đối tượng cần phải thực hiện trong đợt kiểm thử, tuy nhiên vì lý do thiếu môi trường kiểm thử, thiếu thiết bị ... nên đã không thể thực hiện được.

**Block:** Test case không thể thực hiện do bị chặn bởi một lỗi khác hoặc một vấn đề nào đó chưa được giải quyết.

**Skip:** Test case đã được bỏ qua và không thực hiện trong đợt kiểm thử hiện tại.

### KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

**Kết quả đạt được :**

* Cải thiện khả năng đọc hiểu tài liệu tiếng anh.
* Nắm vững quy trình kiểm thử thủ công và các phương pháp kiểm thử phổ biến.
* Có kiến thức về test case, lock bug
* Kỹ năng giao tiếp, ứng xử, hòa nhập thích ứng với môi trường làm việc.
* Phát triển và hoàn thiện bản thân cả trong tính cách lẫn công việc như chủ động hơn trong công việc.

**Kết luận :**

* Việc thực hiện kiểm thử được thực hiện theo một quy trình chặt chẽ và có sự liên kết giữa các bước thực hiện.
* Thiết kế test case phù hợp sẽ nâng cao được hiệu quả của kiểm thử tự động.
* Kiểm thử test sớm trong quy trình phát triển phần mềm có thể giúp giảm chi phí , giảm thiểu rủi ro và phát hiện lỗi sớm

**Hạn chế:**

* Thiếu kinh nghiệm trong việc viết testcase
* Thiếu sự kỹ càng và chi tiết trong quá trình xây dựng testcase.

**Hướng phát triển**

**Qua quá trình học tập và thực tập tại doanh nghiệp, em đã tích lũy được nhiều kiến thức và kinh nghiệm quý báu về lĩnh vực Tester. Những kiến thức này đã giúp em định hướng rõ ràng hơn về công việc tương lai và có thêm động lực để trau dồi kỹ năng chuyên môn.**

**Em đã nhận thức được tầm quan trọng của việc kết hợp giữa kiến thức nền tảng và kinh nghiệm thực tế trong công việc Tester. Do đó, em sẽ tiếp tục củng cố kiến thức đã học trên ghế nhà trường, đồng thời tích cực tham gia các khóa học chuyên sâu và cập nhật những xu hướng mới nhất trong lĩnh vực kiểm thử phần mềm.**

**Bên cạnh đó, em cũng ý thức được rằng kỹ năng mềm đóng vai trò quan trọng không kém kỹ năng chuyên môn. Do đó, em sẽ rèn luyện các kỹ năng như giao tiếp, làm việc nhóm, tư duy phản biện và giải quyết vấn đề để có thể thích ứng tốt với môi trường làm việc năng động và đòi hỏi cao.**

**Ngoài ra, em cũng sẽ dành thời gian để tìm hiểu và học hỏi về Automation Testing, đây là một lĩnh vực đang ngày càng phát triển và có nhu cầu cao về nhân lực. Việc trang bị cho bản thân kiến thức về Automation Testing sẽ giúp em có nhiều cơ hội việc làm hơn trong tương lai.**

### TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | "youthdev," [Online]. Available: https://youthdev.net/vi/. |
| [2] | "topcv," [Online]. Available: https://www.topcv.vn/tester-la-gi. |
| [3] | "manual-testing," [Online]. Available: https://anhtester.com/blog/manual-testing/5-ky-thuat-kiem-thu-phan-mem-thuong-su-dung. |
| [4] | "ky nang can co," [Online]. Available: https://iviettech.vn/blog/12075-tester-la-gi-cac-ky-nang-cua-mot-tester-gioi.html. |
| [5] | "luong tester," [Online]. Available: https://www.topcv.vn/luong-tester. |
| [6] | "kiem-thu-phan-mem-la-gi-quy-trinh-kiem-thu-phan-mem," viblo.asia, [Online]. Available: 1https://viblo.asia/p/kiem-thu-phan-mem-la-gi-quy-trinh-kiem-thu-phan-mem-1VgZvayYKAw. |
| [7] | "7-nguyen-tac-kiem-thu-phan-mem," testing.vn, [Online]. Available: https://www.testing.vn/7-nguyen-tac-kiem-thu-phan-mem/. |
| [8] | "vong-doiquy-trinh-kiem-thu-phan-mem," viblo.asia, [Online]. Available: https://viblo.asia/p/software-testing-life-cycle-vong-doiquy-trinh-kiem-thu-phan-mem-WAyK8MBWZxX. |
| [9] | "mo-hinh-waterfal," amis.misa.vn, [Online]. Available: https://amis.misa.vn/57939/mo-hinh-waterfall/. |
| [10] | "scrum," aws.amazon.com, [Online]. Available: https://aws.amazon.com/vi/what-is/scrum/. |
| [11] | " V-mode," viblo.asia, [Online]. Available: https://viblo.asia/p/tim-hieu-v-model-trong-kiem-thu-phan-mem-GrLZDJbn5k0. |
| [12] | "phuong-phap-agile," businesswiki, [Online]. Available: https://businesswiki.codx.vn/phuong-phap-agile/. |
| [13] | "type," guru99, [Online]. Available: https://www.guru99.com/vi/types-of-software-testing.html. |
| [14] | "test-case," topdev.vn, [Online]. Available: https://topdev.vn/blog/test-case-la-gi/. |
| [15] | "quy-trinh-kiem-thu-phan-mem," itnavi.com.vn, [Online]. Available: https://itnavi.com.vn/blog/quy-trinh-kiem-thu-phan-mem. |
| [16] | "tai-sao-kiem-thu-phan-mem-can-thiet," viblo.asia, [Online]. Available: https://viblo.asia/p/why-is-software-testing-necessary-tai-sao-kiem-thu-phan-mem-can-thiet-DZrGNoAjeVB. |
| [17] | "moi-truong-test-kiem-thu-trong-kiem-thu-phan-mem," viblo.asia, [Online]. Available: https://viblo.asia/p/moi-truong-test-kiem-thu-trong-kiem-thu-phan-mem-924lJbxalPM. |

### CHECK LIST CỦA BÁO CÁO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Nội dung công việc** | **Có** | **Không** | **Ghi chú** |
| 1 | Báo cáo được trình bày (định dạng) đúng với yêu cầu. | X |  |  |
| 2 | Báo cáo có số lượng trang đáp ứng đúng yêu cầu (30-50 trang) | X |  |  |
| 3 | Báo cáo trình bày được phần mở đầu bao gồm: Mục tiêu, Phạm vi và đối tượng, kết cấu … | X |  |  |
| 4 | Báo cáo trình bày về công ty, vị trí việc làm (công việc đó làm gì, kiến thức và kỹ năng cần thiết là gì, con đường phát triển sự nghiệp (career path)), cơ sở lý thuyết phù hợp với nội dung của đề tài (Tối đa 10-12 trang) | X |  |  |
| 5 | Báo cáo có sản phẩm cụ thể phù hợp với mục tiêu đặt ra của đề tài | X |  |  |
| 6 | Báo cáo có phần kết luận và hướng phát triển của đề tài | X |  |  |

### PHỤ LỤC

**Link file test case:** [**Link**](https://docs.google.com/spreadsheets/d/1WjCCxS6y_ctYRqgMzUOYPER89EzKCCsG4HWWsVi0H2A/edit?gid=0#gid=0)

**Link file bug:** [**link**](https://docs.google.com/spreadsheets/d/1uYqZx8TWNs95_fbgVsvvJVEzCpKj6vyc/edit?gid=811065015#gid=811065015)