TRƯỜNG ĐẠI HỌC KINH TẾ

**KHOA THỐNG KÊ – TIN HỌC**

–––––––––––––––––––––––––––––––



**BÁO CÁO THỰC TẬP NGHỀ NGHIỆP**

**THỰC HIỆN KIỂM THỬ DỰ ÁN PHẦN MỀM QUẢN LÝ HỢP ĐỒNG ĐIỆN TỬ**

**Sinh viên thực hiện**: Huỳnh Ngọc Trâm

**Lớp:** 44K21.2

**Giảng viên hướng dẫn:** Cao Thị Nhâm

***Đà Nẵng, 7/2021***

**LỜI CẢM ƠN**

Chuyên đề thực tập nghề nghiệp chuyên ngành Quản trị hệ thống thông tin với đề tài “THỰC HIỆN KIỂM THỬ DỰ ÁN PHẦN MỀM QUẢN LÝ HỢP ĐỒNG ĐIỆN TỬ” tại công ty VNPT-IT3 là kết quả của quá trình cố gắng của bản thân và sự giúp đỡ tận tình của đơn vị thực tập cũng như sự quan tâm động viên của thầy cô. Qua đây em xin gửi lời cảm ơn đến những người đã giúp đỡ em trong quá trình thực tập ngắn này.

Em xin cảm ơn cô Cao Thị Nhâm. Cô đã tận tình hướng dẫn và cung cấp tài liệu cần thiết, góp ý cũng như giám sát tiến độ kĩ lưỡng, góp ý để em có thể hoàn thành báo cáo một cách tốt nhất.

Em xin chân thành cảm ơn ban lãnh đạo nhà trường, đặc biệt là khoa Thông kê – Tin học đã tạo ra những kì thực tập dù ngắn nhưng đem lại đầy kinh nghiệm cũng như hiểu biết về môi trường làm việc của công ty.

Em xin gửi lời cảm ơn đến công ty VNPT\_IT3, đến chị Huỳnh Thùy Trang là người theo sát em trong quá trình thực tập và các anh chị trong team Tester đã hỗ trợ cả vật chất lẫn tinh thần trong kì thực tập này. Em đã có thể hiểu về văn hóa doanh nghiệp và cả những quy tắc hay sự giao tiếp, làm việc nhóm với team là thế nào.

Lần đầu tiên được tiếp xúc với môi trường công sở, còn nhiều bỡ ngỡ và cả những hạn chế nên không tránh khỏi thiếu xót, rất mong sự góp ý của thầy cô để có thể hoàn thiện hơn về kiến thức và cải thiện những thiếu xót của chính mình.

Em xin chân thành cảm ơn!

**Sinh viên Huỳnh Ngọc Trâm**

**LỜI CAM ĐOAN**

Chúng em xin cam đoan dự án **THỰC HIỆN KIỂM THỬ DỰ ÁN PHẦN MỀM QUẢN LÝ HỢP ĐỒNG ĐIỆN TỬ** là kết quả nghiên cứu độc lập dưới sự hướng dẫn của giáo viên hướng dẫn TS. Cao Thị Nhâm. Ngoài ra không có bất cứ sự sao chép nào của người khác.

Dự án, nội dung báo cáo là sản phẩm mà em đã nỗ lực nghiên cứu trong quá trình học tập cũng như tìm hiểu trên ở nhiều phương tiện. Các dữ liệu, kết quả trình bày trong báo cáo là hoàn toàn trung thực, em xin chịu hoàn toàn trách nhiệm, kỷ luật của bộ môn và nhà trường đề ra nếu như có vấn đề xảy ra.

**Sinh viên thực hiện**

**Huỳnh Ngọc Trâm**

# MỤC LỤC

[**MỤC LỤC i**](#_Toc76397997)

[**DANH MỤC HÌNH ẢNH ii**](#_Toc76397998)

[**DANH MỤC BẢNG BIỂU iii**](#_Toc76397999)

[**DANH MỤC CÁC TỪ VIẾT TẮT iv**](#_Toc76398000)

[**LỜI MỞ ĐẦU 1**](#_Toc76398001)

[**CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 2**](#_Toc76398002)

[**1.1 Tổng quan về Testing 2**](#_Toc76398003)

[**1.1.1 Tài liệu SRS 2**](#_Toc76398004)

[**1.1.2 Test Type (Loại kiểm thử) 2**](#_Toc76398005)

[**1.1.3 Test Level (Mức độ kiểm thử) 6**](#_Toc76398006)

[**1.1.4 Test Method (Phương pháp kiểm thử) 12**](#_Toc76398007)

[**1.1.4.1.1 Định nghĩa 12**](#_Toc76398008)

[**1.1.4.1.2 Đối tượng được kiểm thử 12**](#_Toc76398009)

[**1.1.4.1.3 Phương pháp thử nghiệm 13**](#_Toc76398010)

[**1.1.4.2.1 Định nghĩa 13**](#_Toc76398011)

[**1.1.4.2.2 Đối tượng kiểm thử 13**](#_Toc76398012)

[**1.1.4.2.3 Phương pháp thử nghiệm 14**](#_Toc76398013)

[**1.1.4.3.2 Phương pháp thử nghiệm 14**](#_Toc76398014)

[**1.1.5 Kĩ thuật kiểm thử hộp đen 15**](#_Toc76398015)

[**1.2 Tổng quan về nghề nghiệp 17**](#_Toc76398016)

[**1.2.1 Vị trí làm việc 17**](#_Toc76398017)

[**1.2.2 Mô tả công việc của nhân viên Tester 17**](#_Toc76398018)

[**1.2.3 Mức lương 18**](#_Toc76398019)

[**CHƯƠNG 2. PHÂN TÍCH VÀ THIẾT KẾ ỨNG DỤNG 19**](#_Toc76398020)

[**2.1 Mô tả bài toán 19**](#_Toc76398021)

[**2.2 Thiết kế các chức năng 19**](#_Toc76398022)

[**2.3 Thiết kế cơ sở dữ liệu 19**](#_Toc76398023)

[**CHƯƠNG 3. KẾT QUẢ 20**](#_Toc76398024)

[**3.1 Kết quả thực hiện thiết kế testcase 20**](#_Toc76398025)

[**3.2 Kết quả test hệ thống 20**](#_Toc76398026)

[**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 21**](#_Toc76398027)

[**TÀI LIỆU THAM KHẢO 22**](#_Toc76398028)

[**PHỤ LỤC (nếu có) 23**](#_Toc76398029)

# DANH MỤC HÌNH ẢNH

[Hình 1: Các loại kiểm thử 9](#_Toc76391368)

[Hình 2: Phương pháp kiểm thử bigbang 14](#_Toc76391369)

[Hình 3: Phương pháp kiểm thử Top down 15](#_Toc76391370)

[Hình 4: Phương pháp kiểm thử Botton up 15](#_Toc76391371)

# DANH MỤC BẢNG BIỂU

[Bảng 1: So sánh kiểm thử chức năng và kiểm thử phi chức năng 11](#_Toc76391383)

[Bảng 2: So sánh các mức độ kiểm thử 18](#_Toc76391384)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| **Viết tắt** | **Viết đầy đủ** |
| TPPM | Thành phần phần mềm |

# LỜI MỞ ĐẦU

1. **Mục tiêu của đề tài**

* Nghiên cứu

1. **Nhiệm vụ của đề tài**
2. **Kết cấu của đề tài**

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

## Tổng quan về Testing

### Tài liệu SRS

#### Khái niệm

Tài liệu SRS là từ viết tắt của Software Requirement Specification, được dịch ra tiếng việt là tài liệu đặc tả yêu cầu. SRS là tài liệu được sử dụng để mô tả chi tiết các yêu cầu chức năng và phi chức năng của hệ thống. Tài liệu này sẽ hỗ trợ đưa ra các tính năng của hệ thống hay dùng cho việc đọc hiểu hệ thống của bên thứ ba liên quan đến công ty. Đây là một tài liệu quan trọng cho đội phát triển (system analyst, business analyst, code) và kiểm thử (tester).

#### Tầm quan trọng của tài liệu SRS

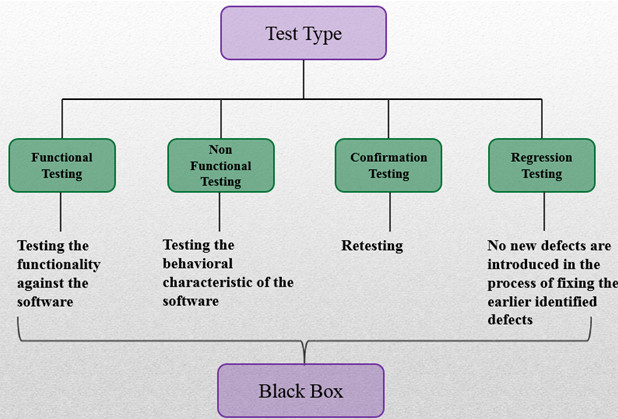
SRS là tài liệu đặc tả vô cùng quan trọng trong quá trình phát triển phần mềm, nó có vai trò:

* Giúp cho các bên thứ ba – stakeholders đều hiểu được hệ thống theo cùng một hướng, tránh trường hợp mỗi người một ý.
* Giúp cho đội phát triển xây dựng hệ thống một cách chính xác, đặc tả được các tính năng, không đi lạc hướng so với yêu cầu của khách hàng.
* SRS giúp nhà kiểm thử hệ thống đọc hiểu từ đó xây dựng nên kịch bản kiểm thử chi tiết nhất.
* Giúp cho việc bảo trì hệ thống và cải tiến những chức năng của hệ thống một cách nhanh chóng và dễ dàng.

### Test Type (Loại kiểm thử)

Có tất cả 4 loại Test Types:

1. Testing of function  (Functional testing)
2. Testing of software product characteristics (Non – Functional testing)
3. Testing of software structure/architecture ( Structural testing)
4. Testing related to changes (Confirmation and regression testing)

[](https://tanhongit.com/wp-content/uploads/2020/05/cac-loai-kiem-thu-test-types.jpg)

Hình 1: Các loại kiểm thử

#### Testing of function (Functional testing)\_ Kiểm thử chức năng

Functional testing hay còn gọi kiểm thử chức năng được các Tester(người kiểm thử) 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 chức năng có thể thực hiện theo 2 quan điểm

Đó chính là requirements-based và business – process – based.

1. Requirements – based: Dùng đặc tả yêu cầu dùng để làm cơ sở để thiết kế các test case (design test).Chúng ta nên xét độ ưu tiên của yêu cầu dựa trên các tiêu chí rủi ro và sử dụng độ ưu tiên để kiểm thử. Điều này sẽ đảm bảo những phần quan trọng nhất sẽ được kiểm tra đầy đủ.
2. Business – process – based: sử dụng các kiến thức về quy trình nghiệp vụ. Quy trình nghiệp vụ mô tả các kịch bản liên quan đến nghiệp vụ hằng ngày của hệ thống.

Kiểm thử chức năng bao gồm 5 bước:

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

#### Testing of software product characteristics – Kiểm thử phi chức năng (Non – Functional testing)

Kiểm thử phi chức năng(Non – Functional testing) là kiểm tra các đặc tính chất lượng của hệ thống, các tester sẽ quan tâm đến việc mọi chức hoạt động tốt như thế nào, tốc độ nhanh như thế nào, phản hồi và số lượng người truy cập vào cùng một thời điểm cũng như xem mức độ sử dụng CPU, Ram, Pin,… của phần mềm,…

Kiểm thử phi chức năng bao gồm:

1. Kiểm thử hiệu năng (Performance testing)
2. Kiểm thử khả năng chịu tải (Load testing)
3. Kiểm thử áp lực (Stress testing)
4. Kiểm thử tính khả dụng (Usability testing)
5. Kiểm thử bảo trì (Maintainability testing)
6. Kiểm thử độ tin cậy (Reliability testing)
7. Kiểm thử tính tương thích (Portability testing)

|  |  |
| --- | --- |
| **Kiểm thử chức năng** | **Kiểm thử phi chức năng** |
| Kiểm thử chức năng được thực hiện sử dụng kỹ thuật cung cấp bởi người dùng và xác thực yêu cầu của hệ thống | Kiểm thử phi chức năng kiểm tra hiệu suất, độ tin cậy, khả năng mở rộng và các khía cạnh khác của hệ thống phần mềm |
| Kiểm thử chức năng được thực thi đầu tiên | Thực hiện sau thực hiện chức năng |
| Công cụ sử dụng: test bằng tay hoặc test tự động | Sử dụng các công cụ |
| Đầu vào là yêu cầu nghiệp vụ | Đầu vào là tốc độ , khả năng mở rộng , những tham số về hiệu suất |
| Đầu vào là yêu cầu nghiệp vụ | Mô tả hiệu quả của sản phẩm |
| Dễ dàng test bằng tay | Khó kiểm tra bằng tay |
| Các loại kiểm thử chức năng: Kiểm thử đơn vị, Kiểm thử sơ lược, Kiểm thử độ bền, Kiểm thử tích hợp, Kiểm thử hộp đen, Kiểm thử hộp trắng, Kiểm thử người dùng chấp nhận, Kiểm thử hồi quy | Các loại kiểm thử phi chức năng: Kiểm thử hiệu suất, Kiểm thử việc tải, Kiểm thửu khả năng chịu tải Kiểm thử an toàn, Kiểm thử cài đặt, Kiểm thử áp lực, Kiểm thử thâm nhập, Kiểm thử tính tương thích, Kiểm thử di chuyển |

Bảng 1: So sánh kiểm thử chức năng và kiểm thử phi chức năng

#### Testing of software structure/architecture ( Structural testing)

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 cấu trúc chủ yếu được áp dụng ở Kiểm Thử Thành Phần, Kiểm Thử Tích Hợp.

Ở cấp độ thành phần, và mức thấp hơn trong kiểm thử tích hợp thành phần có hỗ trợ công cụ tốt để đo mức độ bao phủ của code. Các công cụ đo lường mức độ bao phủ đánh giá tỉ lệ phần trăm thực thi sau khi được thực hiện bởi một bộ kiểm thử. Nếu bộ bao phủ ko phải tuyệt đối thì các kiểm thử bổ sung có thể cẩn được viết và chạy để có thể bao phủ hết các phần còn lại.

Các kỹ thuật được dùng để kiểm tra cấu trúc là kỹ thuật kiểm thử  “hộp trắng”, các mô hình luồng điều khiển thông thường được dùng để kiểm thử các cấu trúc.

Testing related to changes (Confirmation and regression testing)

##### Confirmation testing (Kiểm thử xác nhận)

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

Khi thực hiện kiểm tra xác nhận điều quan trọng nhất là phải đảm bảo rằng các trường hợp thử phải được thực hiện chính xác giống như lần đầu tiên, sử dụng cùng một đầu vào, dữ liệu và môi trường để kiểm thử đảm bảo rằng các lỗi đã được sửa.

Tester cần phải biết rằng trong lần kiểm thử sau khi được vá lỗi khả năng sinh ra lỗi khác trong phần mềm là điều hiển nhiên vì vậy kiểm chính xác ở phiên bản hiện tại của một phần mềm là chưa đủ. Cách phát hiện các bất lợi ngoài ý muốn của việc kiểm lỗi là thực hiện kiểm thử hồi quy.

##### Regression testing (Kiểm thử hồi quy)

Giống như kiểm thử xác nhận thì bên kiểm thử hồi quy liên quan đến việc thực hiện các trường hợp kiểm thử đã được thực hiện trước đó.

Mục đích của kiểm thử hồi quy là để xác minh rằng các sửa đổi trong phần mềm hoặc môi trường không gây ra bất lợi ngoài ý muốn, ảnh hưởng hoặc làm hư các chức năng liên quan không và hệ thống vẫn đáp ứng các yêu cầu của phần mềm.

Bộ kiểm thử hồi quy hoặc gói kiểm tra hồi quy là một tập hợp các trường hợp kiểm thử được sử dụng đặc biệt để kiểm tra hồi quy.  Bộ kiểm thử hồi quy được thiết kế để thực hiện kiểm hầu hết các chức năng trong một hệ thống nhưng không kiểm chi tiết bất kỳ chức năng nào. Tất cả các trường hợp trong bộ kiểm thử hồi quy sẽ được thực thi mỗi khi một phiên bản vá lỗi của phần mềm được phát hành và điều này làm cho chúng trở nên lý tưởng cho tự động hóa.

Kiểm thử hồi quy được thực hiện khi phần mềm thay đổi, do sửa lỗi, chức năng mới.

### Test Level (Mức độ kiểm thử)

Tất cả các giai đoạn của quá trình phát triển phầm mềm đều trải qua quá trình kiểm thử phần mềm. Có 4 cấp độ kiểm thử phần mềm là:

* Kiểm thử đơn vị (Unit Testing)
* Kiểm thử tích hợp (Integration Testing)
* Kiểm thử hệ thống (System Testing)
* Kiểm thử chấp nhận (Acceptance Testing)

#### Unit Testing( Kiểm thử đơn vị)

##### Định nghĩa

Unit Testing là một mức kiểm thử phần mềm với mục đích để xác nhận từng unit của phần mềm được phát triển đúng như được thiết kế. Unit testing là mức test nhỏ nhất trong bất kỳ phần mềm nào. các hàm (Function), thủ tục (Procedure), lớp (Class), hoặc các phương thức (Method) đều có thể được xem là Unit. Nó thường có một hoặc vài đầu vào nhưng đầu ra là duy nhất.

##### Khi nào thực hiện Unit Testing?

Unit testing là mức kiểm thử đầu tiên trong các mức kiểm thử phần mềm. Nó được thực hiện trước khi Integration Testing. Kiểm thử “Hộp trắng”

##### Ai thực hiện Unit Testing?

Unit Testing thường do lập trình viên thực hiện. Unit test được thực hiện càng sớm càng tốt trong giai đoạn viết code và xuyên suốt quá trình phát triển phần mềm.

##### Mục đích

* Tăng sự đảm bảo khi có sự thay đổi mã
* Code dễ sử dụng, dễ hiểu, có thể tái sử dụng nhiều hơn
* Phát triển nhanh hơn
* Chi phí sửa lỗi thấp hơn so với các mức kiểm thử giai đoạn sau
* Debug dễ dàng

#### Integration Testing( Kiểm thử tích hợp)

##### Định nghĩa

Kiểm thử tích hợp là một mức của kiểm thử phần mềm kiểm tra một nhóm các module nhỏ liên quan đến nhau xem chúng có hoạt động đúng chức năng như trong thiết kế hay không. Theo ISTQB ( International Software Testing Qualifications Board):

* Kiểm thử tích hợp được thực hiện để phát hiện các lỗi về giao diện hoặc trong tương tác giữa các thành phần hoặc hệ thống tích hợp
* Kiểm thử tích hợp thành phần: kiểm tra sự tương tác giữa các thành phần với điều kiện các thành phần đã pass ở phần kiểm thử thành phần trước đó
* Kiểm thử tích hợp hệ thống: kiểm tra sự tương tác giữa các hệ thống con khác nhau và các hệ thống này đã pass ở lần kiểm thử trước đó

##### Khi nào thực hiện Integration Testing?

Kiểm thử tích hợp là mức thứ 2 trong các mức kiểm thử phần mềm. Nó được thực hiện sau Unit Testing và trước System testing. Kiểm thử mọi hộp (đen, trắng, xám)

##### Ai thực hiện Integration Testing?

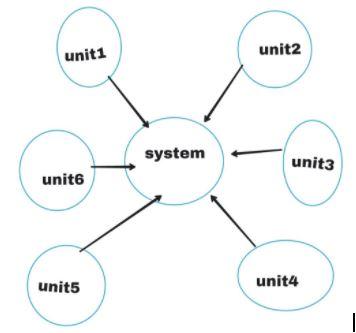
Kiểm thử integration có thể được thực hiện bởi developer, một test team chuyên biệt hay một nhóm chuyên developer/kiểm thử viên tích hợp bao gồm cả kiểm thử phi chức năng.

##### Mục đích

Kiểm tra sự tích hợp 1 nhóm các thành phần riêng lẻ có liên quan xem chúng có hoạt động đúng như mong đợi hay không.

##### Một số phương pháp kiểm thử tích hợp

###### Bigbang

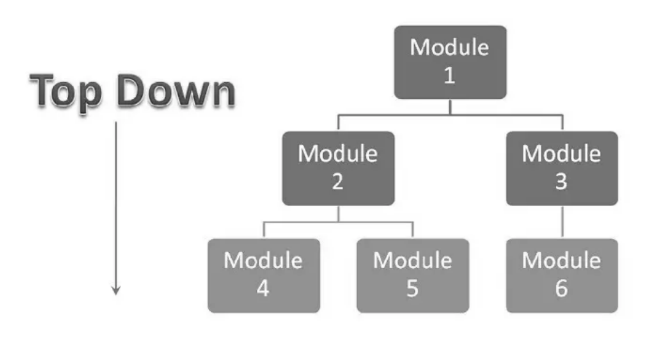


Hình 2: Phương pháp kiểm thử Bigbang

Đây là phương pháp test tích hợp mà tất cả hoặc hầu hết các unit được kết hợp với nhau và cùng được kiểm thử. Phương pháp này được thực hiện khi team kiểm thử nhận được toàn bộ phần mềm.

Như vậy big bang testing là kiểm tra sự tương tác giữa các unit , còn system test là sự tương tác của cả hệ thống.

###### Top down



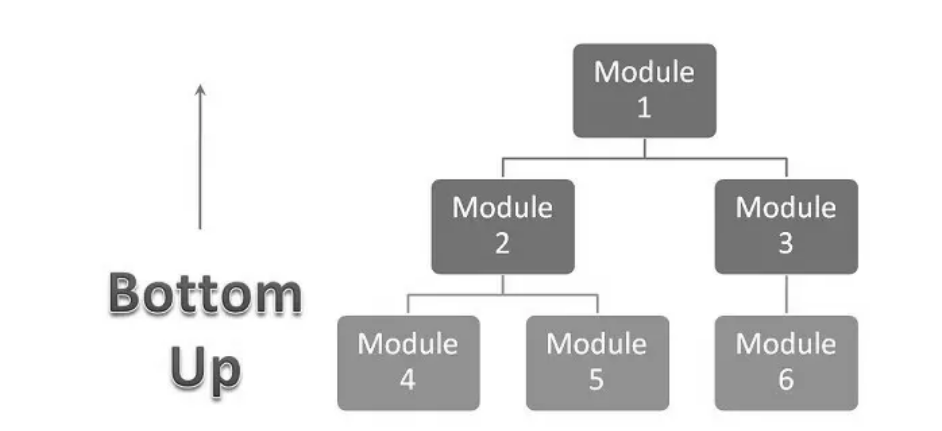
Hình 3: Phương pháp kiểm thử Top down

Đơn vị cao nhất được kiểm thử đầu tiền, đơn vị thấp hơn được kiểm thử sau đó một các tuần tự.

Vd: các chức năng chung, menu chính,… (ví dụ: bắt đầu từ GUI hoặc menu chính).

Modules quan trọng đang được thử nghiệm trên mức ưu tiên; lỗi trong thiết kế lớn có thể được tìm thấy và cố định đầu tiên.

###### Bottom up



Hình 4: Phương pháp kiểm thử Botton up

Đơn vị dưới cùng được kiểm thử đầu tiên, đơn vị cao hơn được kiểm thử tuần tự sau đó.

Vd: kiểm tra các module nhỏ hơn sau đó gộp vs module lớn hơn,…

Mỗi module ở mức thấp hơn được thử nghiệm với các module cao hơn cho đến khi tất cả các module đều được kiểm tra.

###### Sanwich

Sandwich là một cách tiếp cận để kiểm thử tích hợp, đó là sự kết hợp của các phương pháp Top Down và Bottom Up.

##### Một số lưu ý trước khi thực hiện kiểm thử tích hợp

* Đảm bảo rằng bạn có tài liệu thiết kế chi tiết phù hợp trong đó các tương tác giữa mỗi đơn vị được xác định rõ ràng.
* Đảm bảo rằng bạn có một hệ thống quản lý cấu hình phần mềm mạnh mẽ tại chỗ. Hoặc nếu không, bạn sẽ có một thời gian khó khăn theo dõi phiên bản phù hợp của từng đơn vị, đặc biệt là nếu số lượng đơn vị được tích hợp là rất lớn.
* Đảm bảo rằng mỗi đơn vị được kiểm thử đơn vị trước khi bạn bắt đầu Kiểm thử tích hợp.

#### System Testing( Kiểm thử hệ thống)

##### Định nghĩa

Kiểm thử hệ thống là một mức của kiểm thử phần mềm . Giai đoạn này sẽ hoàn thiện và hợp nhất phần mềm để kiểm thử. Theo ISTQB định nghĩa: quy trình của kiểm thử tích hợp hệ thống để xác nhận xem hệt hống phần mềm có đáp ứng đúng theo đặc tả yêu cầu.

##### Khi nào thực hiện System Testing?

Kiểm thử hệ thống là mức kiểm thử thứ 3 trong các mức kiểm thử phần mềm được thực hiện sau kiểm thử tích hợp và trước kiểm thử chấp nhận. Kiểm thử “hộp đen”

##### Ai thực hiện System Testing?

Thông thường, các tester thực hiện kiểm thử hệ thống.

##### Mục đích

Mục đích của giai đoạn này là để đánh giá sự hoạt động của hệ thống có đúng theo như tài liệu đặc tả.

#### Acceptance Testing( Kiểm thử chấp nhận)

##### Định nghĩa

Theo ISTQB định nghĩa: Kiểm thử chấp nhận chính thức liên quan đến yêu cầu và quy trình kinh doanh để xác định liệu hệ thống có đáp ứng tiêu chí chấp nhận hay không và cho phép người dùng, khách hàng hoặc tổ chức được ủy quyền khác xác định có chấp nhận hệ thống hay không.

##### Khi nào thực hiện Acceptance Testing?

Kiểm thử chấp nhận là mức thứ 4 được thực hiện sau khi hoàn thành kiểm thử hệ thống và trước khi đưa sản phẩm vào sử dụng chính thức.

##### Ai thực hiện Acceptance Testing?

Kiểm thử chấp nhận được chia thành 2 mức khác nhau

* Kiểm thử alpha: được thực hiện bởi những người trong tổ chức nhưng không tham gia phát triển phần mềm. Kiểm thử “hộp đen và trắng”
* Kiểm thử beta: được thực hiện bởi khách hàng/ người dùng cuối tại địa điểm của người dùng cuối. Kiểm thử “hộp đen”

##### Mục đích

Đảm bảo phần mềm đáp ứng đúng yêu cầu của khách hàng. Sản phẩm nhận được sự chấp nhận từ khách hàng/ người dùng cuối.

##### Kết luận

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Các mức kiểm thử** | **Unit Testing** | **Integration Testing** | **System testing** | **Alpha Testing** | **Beta Testing** |
| Định nghĩa | Là mức kiểm thử nhỏ nhất, test từng module nhỏ trong hệ thống | Kiểm thử tích hợp một nhóm các module riêng lẻ có liên quan với nhau | Mức kiểm thử toàn bộ, hợp nhất tất cả các chức năng của phần mềm | Hoạt động kiểm thử chấp nhận nội bộ | Kiểm thử chấp nhận khi phát hành, bàn giao cho khách hàng. |
| Người thực hiện | Do chính dev viết module đó | Kiểm thử độc lập hoặc do lập trình viên | Tester | Người dùng/ nhóm test độc lập( không phải người phát triển phần mềm) thực hiện tại nơi sản xuất phần mềm | Thực hiện bởi khách hàng/ người dùng cuối tại địa điểm của họ |
| Thời gian thực hiện | \* Trong quá trình build module đó. \* Phải hoàn thành trước khi chuyển sang Integration Testing | Sau khi các module nhỏ liên quan đã được unit test | Sau Integration Testing và trước Acceptance Testing | Sau system Testing , và trước Beta Testing | Sau alpha testing |
| Mục đích | Đảm bảo mỗi module nhỏ của phần mềm được thực hiện đúng với thiêt kế | Kiểm tra chức năng, độ tin cậy và hiệu suất của hệ thống khi tích hợp | Đánh giá toàn bộ hệ thống hoạt động đúng theo đặc tả yêu cầu hay không | Đảm bảo phần mềm hoạt động đúng trước khi bàn giao cho khách hàng | Đảm bảo khách hàng xác nhận phần mềm hoạt động đúng như yêu cầu của họ |
| Ưu điểm | \* Tăng sự đảm bảo khi có sự thay đổi mã. \* Code dễ sử dụng, dễ hiểu, có thể tái sử dụng nhiều hơn. \* Phát triển nhanh hơn. \* Chi phí sửa lỗi thấp hơn so với các mức kiểm thử giai đoạn sau. \* Debug dễ dàng | \* Phát hiện lỗi ở đơn vị thấp của sản phẩm. \* Thuận tiện với các dự án nhỏ | Kiểm tra được tổng thể hoạt động của toàn hệ thống trước khi phát hành | Các vấn đề xuất hiện có thể được giải quyết ngay bởi bên sản xuất | Chỉ đòi hỏi thực hiện trong thời gian ngắn |
| Nhược điểm | Dev ít kinh nghiệm sẽ mất khá nhiều thời gian để fix bug | \* Các lỗ hổng về thiết kế ở cấp cao hơn sẽ được phát hiện chỉ ở cuối vòng đời. \* Có thể bỏ qua các bug giao diện nhỏ trong quá trình tìm bug | Chỉ có thể thực hiện khi tất cả các module riêng lẻ đã hoàn tất và không có lỗi lớn | Có thể kéo dài lâu hơn | \* Giới hạn số lượng người dùng. \* Chi phí sửa lỗi cao hơn nếu có |
| Kỹ thuật kiểm thử | Kiểm thử hộp trắng | Kiểm thử hộp trắng, đen và xám | Kiểm thử hộp đen | Kiểm thử hộp trắng và đen | Kiểm thử hộp đen |

Bảng 2: So sánh các mức độ kiểm thử

### Test Method (Phương pháp kiểm thử)

#### BACK BOX TEST

##### **Định nghĩa**

Kiểm tra hộp đen (Black box testing) là một phương pháp kiểm thử phần mềm mà việc kiểm tra các chức năng của một ứng dụng không cần quan tâm vào cấu trúc nội bộ hoặc hoạt động của nó.

##### **Đối tượng được kiểm thử**

Là thành phần phần mền (TPPM) có thể là 1 hàm chức năng, 1 modul chức năng, 1 phân hệ chức năng...

##### **Phương pháp thử nghiệm**

Dựa vào chức năng Kiểm thử hộp đen (Black box test) có thể được áp dụng hầu như đến mọi cấp độ của kiểm thử phần mềm:

* Kiểm thử đơn vị (Unit test)
* Kiểm thử tích hợp (Intergration test)
* Kiểm thử hệ thống (System test)
* Kiểm thử chấp nhận (Acceptance test).

Tuy nhiên, Black box test được sử dụng thích hợp nhất trong kiểm thử hệ thống (System test) và Kiểm thử chấp nhận (Acceptance test)

##### Đặc điểm

* Là chiến lược kiểm thử TPPM dựa vào thông tin duy nhất là các đặc tả về yêu cầu chức năng của TPPM tương ứng.
* Người kiểm thử không cần thiết phải có kiến thức về việc mã hoá, cấu trúc bên trong của TPPM, cũng như không yêu cầu phải biết lâp trình phần mềm.
* Việc kiểm thử được tiến hành dựa vào việc kiểm thử TPPM làm được gì, có phù hợp với yêu cầu của người dùng hay không. Các tester nhập số liệu vào phần mềm và chỉ cần xem kết quả của phần mềm và các mục tiêu kiểm tra.
* Mức test này thường yêu cầu các tester phải viết test case đầy đủ trước khi test; khi test, đơn giản chỉ cần thực hiện theo các bước mô tả trong test case thao tác và nhập data vào, sau đó xem kết quả trả về hoặc hành vi của phần mềm, rồi so sánh với kết quả mong đọi được viết trong testcase

##### Tạo test case và Thực hiện test case

* Khi viết test case: Dựa vào yêu cầu và giao diện bên ngoài của chương trình (Không can thiệp vào bên trong code của chương trình)
* Khi thực hiện test: Thực hiện trên giao diện của chương trình (yêu cầu chương trình phải chạy được mới test được, không can thiệp vào code)

#### WHITE BOX TEST



##### **Định nghĩa**

Kiểm thử hộp trắng (While box test) là phương pháp thử nghiệm phần mềm, trong đó các thiết kế, cấu trúc giải thuật bên trong, và việc thực hiện các công việc đều được biết đến

##### **Đối tượng kiểm thử**

Là 1 thành phần của phần mềm (1 chức năng, 1 module chức năng, 1 phân hệ chức năng....)

##### **Phương pháp thử nghiệm**

Dựa vào thuật giải Kiểm thử hộp trắng dựa vào thuật giải cụ thể, vào cấu trúc dữ liệu bên trong của đơn vị phần mềm cần kiểm thử để xác định đơn vị phần mềm đó có thực hiện đúng không.

* Với những TPPM quá lớn sẽ tốn rất nhiều thời gian và công sức để kiểm thử nếu như dùng kiểm thử tích hợp (Integration test) hay kiểm thử chức năng (Functional test)).
* Kỹ thuật white box test thích hợp dùng để kiểm thử đơn vị (Unit test)

##### Đặc điểm

* Là chiến lược kiểm thử TPPM dựa vào giải thuật, cấu trúc bên trong chức năng của TPPM tương ứng.
* Người kiểm thử phải có kiến thức nhất định về việc mã hoá, cấu trúc bên trong của chức năng, biết lâp trình phần mềm.
* Việc kiểm thử được tiến hành dựa vào việc kiểm xem giải thuật, mã lệnh đã làm có đúng không.
* Mức test này thường yêu cầu các tester phải viết test case đầy đủ các nhánh trong code; khi test, sẽ set điều kiện và data để chạy vào đủ tất cả các nhánh trong giải thuật, đảm bảo thực hiện đầy đủ.

##### Tạo testcase và thực hiện test

* Khi viết test case: Dựa vào yêu cầu và nội dung Source Code (can thiệp vào bên trong Code của chương trình)
* Khi thực hiện test: Thực thi test trong code (không cần thực thi chương trình, vì thực hiện test white box sẽ sử dụng framework nào đó hỗ trợ (Ví dụ như test kiểu debug)
* Trong kiểm tra này, đòi hỏi người tester phải có kiến thức và kỹ năng nhất định về ngôn ngữ lập trình được dùng, hiểu thuật giải trong thành phần phần mềm, để có thể hiểu được chi tiết về đoạn code cần kiểm thử .

#### GRAY BOX TEST

Ngoài 2 kỹ thuật đã được nhắc đến: Black box test và white box test, thì có 1 kỹthuật, Gray box test là sự kết hợp giữa black box test và white box test.

##### Định nghĩa

Gray Box Testing là một phương pháp kiểm thử phần mềm được kết hợp giữa Phương pháp Kiểm thử Black Box (hộp đen) và White Box (hộp trắng). Trong Kiểm thử Hộp xám, cấu trúc bên trong sản phẩm chỉ được biết một phần

##### **Phương pháp thử nghiệm**

Dựa vào giải thuật và chức năng

Gray box test có thể được sử dụng ở nhiều mức kiểm thử khác nhau. Tuy nhiên, chủ yếu được ứng dụng trong Kiểm thử tích hợp (Intergration test)

##### Tạo testcase và thực hiện test

* Khi viết test case: Dựa vào yêu cầu và nội dung Source Code (can thiệp vào bên trong Code của chương trình)
* Khi thực hiện test: Thực hiện trên giao diện của chương trình (yêu cầu chương trình phải chạy được mới test được, không can thiệp vào code)

### Kĩ thuật kiểm thử hộp đen

Phân vùng tương đương (equivalence class)

Ý tưởng của kỹ thuật này là cố gắng phân các testcase ra thành nhiều nhóm khác nhau: các testcase trong mỗi nhóm xác định TPPM thực hiện cùng 1 hành vi.

Mỗi nhóm testcase thỏa mãn tiêu chuẩn trên được gọi là 1 lớp tương đương, ta chỉ cần xác định 1 testcase đại diện cho nhóm và dùng testcase này để kiểm thử TPPM. Như vậy ta đã giảm rất nhiều testcase cần định nghĩa và kiểm thử, nhưng chất lượng kiểm thử không bị giảm sút bao nhiêu so với vét cạn.

Điều này là dựa vào kỳ vọng:

* Nếu 1 testcase trong lớp tương đương nào đó gây lỗi TPPM thì các testcase trong lớp này cũng sẽ gây lỗi như vậy.
* Nếu 1 testcase trong lớp tương đương nào đó không gây lỗi TPPM thì các testcase trong lớp này cũng sẽ không gây lỗi.
* Với các giá trị không hợp lệ: Ta nên tạo 1 lớp tương đương đại diện các testcase chứa các giá trị không hợp lệ theo đặc tả để xem TPPM phản ứng như nào với những trường hợp này

Kỹ thuật phân tích các giá trị ở biên

Khi tạo testcase, ta chỉ dùng Kỹ thuật phân lớp tương đương thì hẳn là chưa đủ. Kinh nghiệm cho thấy rằng lỗi thường nằm ở biên (đầu hay cuối) của 1 khoảng liên tục nào đó (lớp tương đương).

Do đó với Kỹ thuật phân tích giá trị biên tập trung tạo các testcase ứng với những giá trị ở biên này.

Nên thông thường là có sự kết hợp cả 2 kỹ thuật: Phân lớp tương đương và Phân tích giá trị biên để viết các testcase.

Ý tưởng của kỹ thuật là chỉ định nghĩa các testcase ứng với các giá trị ngay trên biên hay lân cận biên của từng lớp tương đương. Do đó kỹ thuật này chỉ thích hợp với các lớp tương đương xác định bởi các giá trị liên tục (số nguyên, số thực), chứ nó không thích hợp với lớp tương đương được xác định bởi các giá trị liệt kê mà không có mối quan hệ lẫn nhau.

Quy trình cụ thể để thực hiện kiểm thử dựa trên các giá trị ở biên:

* Nhận dạng các lớp tương đương dựa trên đặc tả về yêu cầu chức năng của TPPM.
* Nhận dạng 2 biên của mỗi lớp tương đương. Tạo các testcase cho mỗi biên của mỗi lớp tương đương :
* 1 testcase tại giá trị biên.
* 1 testcase ngay dưới biên.
* 1 testcase ngay trên biên.

Ý nghĩa ngay trên và ngay dưới biên phụ thuộc vào đơn vị đo lường cụ thể : Số nguyên , số thập phân...

Kỹ thuật dùng bảng quyết định (decision table)

Bảng quyết định là 1 công cụ rất hữu ích để đặc tả các yêu cầu phần mềm hoặc để đặc tả bảng thiết kế hệ thống phần mềm. Nó miêu tả các qui tắc nghiệp vụ phức tạp mà phần mềm phải thực hiện dưới dạng dễ đọc và dễ kiểm soát :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Rule 1** | **Rule 2** | **...** | **Rule p** |
| **Conditions** |  |  |  |  |
| Conditions-1 |  |  |  |  |
| ... |  |  |  |  |
| Conditions-m |  |  |  |  |
| **Actions** |  |  |  |  |
| Actions-1 |  |  |  |  |
| ... |  |  |  |  |
| Actions-n |  |  |  |  |

Trong đó:

* Condition-1 tới Condition-m miêu tả m điều kiện dữ liệu nhập khác nhau có thể có.
* Action-1 tới Action-n miêu tả n hoạt động khác nhau mà hệ thống có thể thực hiện phụ thuộc vào tổ hợp điều kiện dữ liệu nhập nào.
* Mỗi cột miêu tả 1 luật cụ thể : tổ hợp điều kiện nhập cụ thể và các hoạt động cụ thể cần thực hiện. Lưu ý các hoạt động cần thực hiện không phụ thuộc vào thứ tự các điều kiện nhập, nó chỉ phụ thuộc vào giá trị các điều kiện nhập. Tương tự, các hoạt động cần thực hiện không phụ thuộc vào trạng thái hiện hành của TPPM, chúng cũng không phụ thuộc vào các điều kiện nhập đã có trước đó.

Sơ đồ chuyển đổi trạng thái (state transition testing)

Sơ đồ chuyển đổi trạng thái là một kỹ thuật được sử dụng để kiểm tra các trạng thái khác nhau của hệ thống. Trạng thái của hệ thống thay đổi tùy theo điều kiện hoặc sự kiện. Các sự kiện kích hoạt các trạng thái trở thành kịch bản và người kiểm tra cần kiểm tra chúng.

Một sơ đồ chuyển trạng thái cung cấp một cái nhìn rõ ràng về các thay đổi trạng thái nhưng chỉ có hiệu quả đối với các ứng dụng đơn giản. Các dự án phức tạp hơn có thể dẫn đến các sơ đồ chuyển đổi trạng thái phức tạp hơn do đó làm cho nó kém hiệu quả hơn.

## Tổng quan về nghề nghiệp



### Vị trí làm việc

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

### **Mô tả công việc của nhân viên Tester**

Là một nhân viên Tester, bạn sẽ là một trong những người đầu tiên tiếp xúc và kiểm thử phần mềm, thường xuyên được làm việc trong các dự án mới, thú vị. Đặc biệt, với vị trí Tester, cho dù chưa có kinh nghiệm bạn vẫn có thể ứng tuyển. Sự tập trung và khả năng làm việc độc lập là một trong những yếu tố quan trọng hàng đầu mà nhà tuyển dụng tìm kiếm.

Công việc của nhân viên Tester

* Là Tester cần phải hiểu rõ được các sản phẩm cần phải kiểm tra.
* Cần nắm rõ cạc lập kế hoạch các chiến lược thử nghiệm để tìm ra được những vấn đề cần giải quyết và thực hiện các thử nghiệm.

Thực hiện phân tích rõ ràng các ưu điểm, nhược điểm hoặc giải quyết dễ dàng hơn các rủi ro liên quan đến từng thành phần cũng như giao diện của sản phẩm.

* Cần hiểu rõ về code để có thể dễ dàng check cũng như kiểm tra lại chúng.
* Cần phải có các kỹ năng làm việc với các tập lệnh cũng như công cụ tự động hóa.
* Cần thường xuyên cập nhật cho mọi khía cạnh kỹ thuật của các cơ sở hạ tầng sản phẩm.
* Chú ý phân tích, ghi nhận được các vấn đề và có trách nhiệm cung cấp các phản hồi thích hợp nhất.

Yêu cầu trình độ và kỹ năng với nhân viên Tester

* Bằng cử nhân/cao đẳng các chuyên ngành khoa học máy tính, công nghệ thông tin hoặc liên quan.
* Kiến thức về thiết kế, kiểm thử phần mềm và thành thạo các phương pháp thử nghiệm.
* Thành thạo các kỹ thuật kiểm thử phần mềm cũng như đánh giá khả năng tương thích với các chương trình phần mềm khác nhau.
* Kiến thức về lập trình.
* Giao tiếp tốt và kỹ năng tư duy phản biện.
* Kỹ năng tổ chức tốt và tư duy định hướng chi tiết.

### Mức lương

Theo khảo sát thực tế, mức lương của nhân viên Tester có thể dao động trong khoảng **từ 4 đến 34 triệu/tháng**, tuỳ thuộc vào hình thức làm việc, kinh nghiệm, trình độ và hiệu suất.

# CHƯƠNG 2. PHÂN TÍCH HỆ THỐNG

## Tổng quan hệ thống:

Với nhiều chi nhánh trên toàn quốc và lượng khách hàng đáng kể. Hệ thống hợp đồng điện tử nhằm đáp ứng yêu cầu quản lý hợp đồng điện tử đáp ứng yêu cầu về pháp luật của nhà nước và quy định của tập đoàn VNPT, Tổng công ty Vinaphone. Để các bước trong việc kí kết hợp đồng được tối ưu về thời gian lẫn chi phí, việc quản lý cũng trở nên có logic hơn.

Các tác nhân của hệ thống gồm: Quản trị viên, nhân viên, khách hàng.

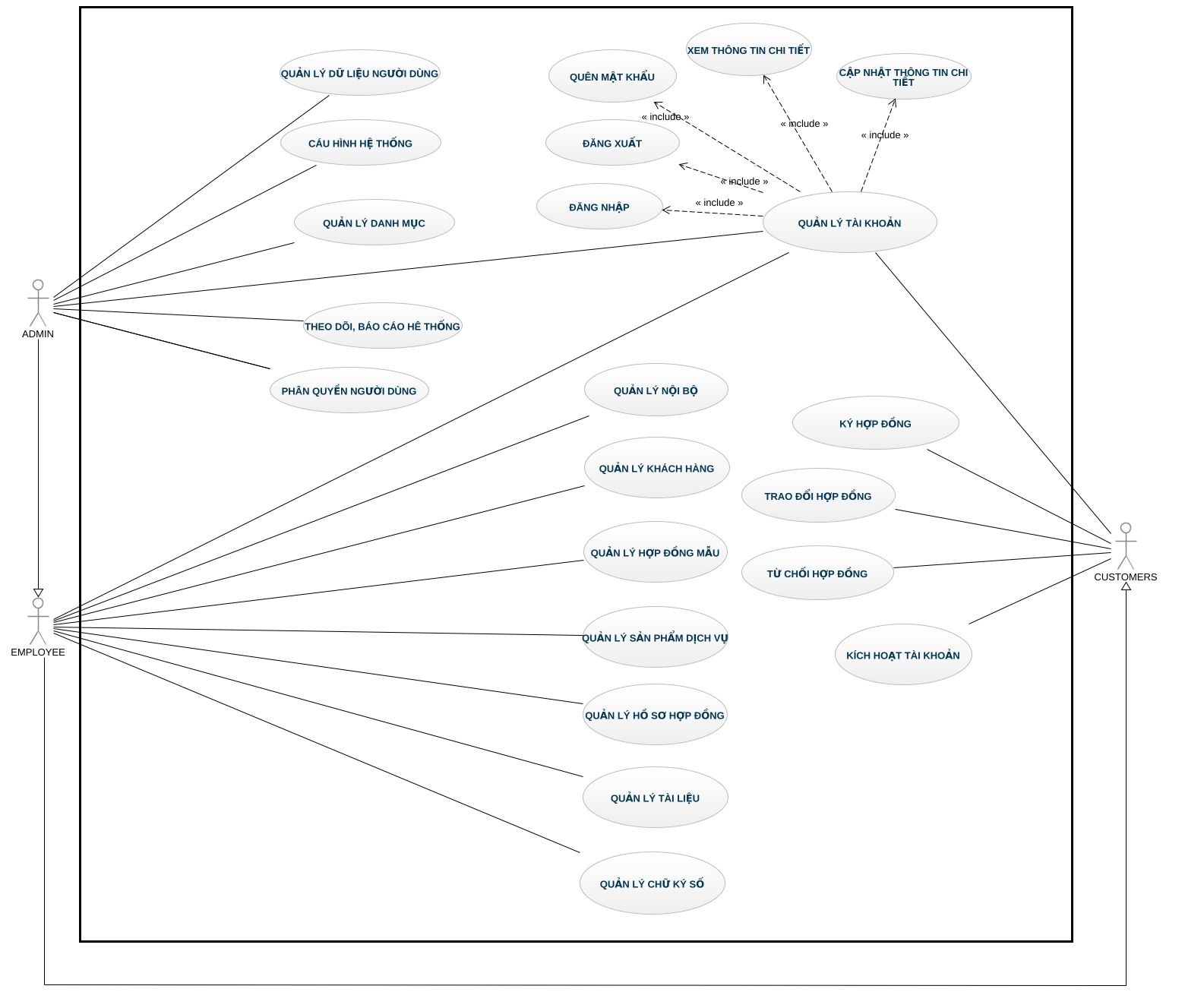
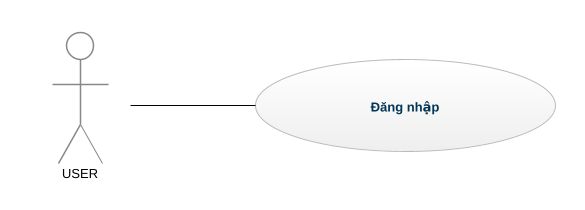
Quản trị viên (admin) có chức năng quản lý:

Nhân viên có quyền:

Khách hàng có quyền:

|  |  |  |
| --- | --- | --- |
| STT | Tác nhân | Ý nghĩa, nhiệm vụ của các tác nhân |
| 1 | Admin | Tác nhân admin là tác nhân quan trọng, giữ vai trò chính của hệ thống, quản lý, theo dõi, kiểm soát các hoạt động trong hệ thống.  Tác nhân admin có tất cả các chức năng trong hệ thống: |
| 2 | Nhân viên | Tác nhân nhân viên là tác nhân hoạt động chính trong hệ thống, liên kết với khách hàng để kí kết hợp đồng.  Tác nhân nhân viên có các chức năng sau: |
| 3 | Khách hàng | Tác nhân khách hàng là tác nhân có tác động qua lại với nhân viên.  Tác nhân khách hàng có các chức năng sau: |

Bảng : Tổng quan hệ thống



Hình : Sơ đồ Use Case tổng quan

## 2.2 Phân tích yêu cầu

### Chức năng “Đăng nhập”

#### Biểu đồ Usecase cho chức năng “Đăng nhập”

#### Đặc tả yêu cầu cho chức năng “Đăng nhập”

Hình : Sơ đồ Use case cho chức năng Đăng nhập

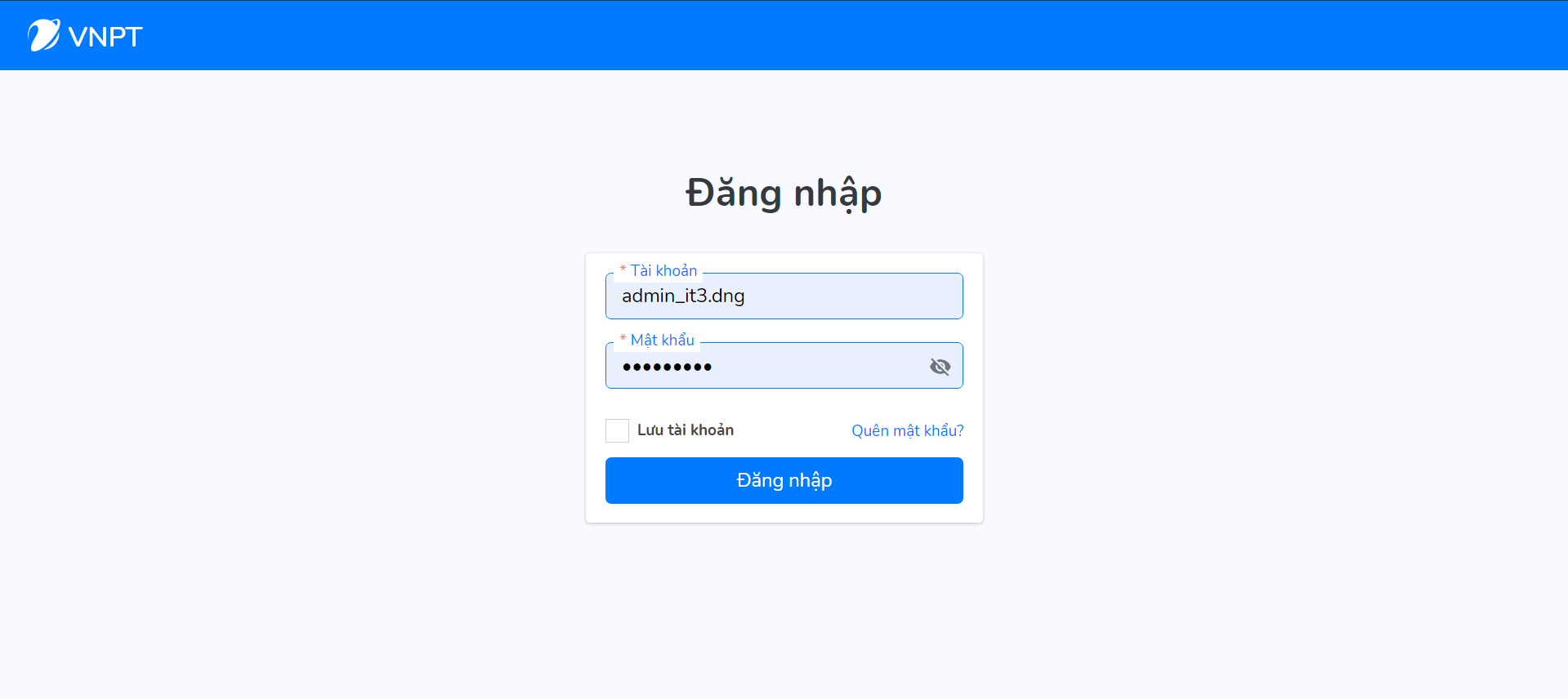
* Tác nhân: Admin, Nhân viên, Khách hàng
* Điều kiện tiên quyết:
* Admin đã có tài khoản được phân quyền
* Nhân viên và khách hàng đã nhận tài khoản và mật khẩu từ admin.
* Mô tra khái quát: Use case này thực hiện việc đăng nhập vào hệ thống. Tùy vào quyền của người dùng mà đăng nhập với tài khoản cho sẵn.
* Mô tả chi tiết Use case:
* Bước 1: Admin tạo 1 tài khoản cho nhân viên ở mục quản lý nhân viên hoặc khách hàng và cấp 1 mật khẩu nhất định.
* Bước 2: Nhân viên hoặc khách hàng vào bằng tài khoản và mật khẩu được cung cập bởi Admin:

|  |  |  |
| --- | --- | --- |
| Tên trường | Trường bắt buộc | Mục đích |
| Tên tài khoản | x | Người dùng nhập tên tài khoản để đăng nhập |
| Mật khẩu | x | Người dùng nhập mật khẩu để đăng nhập |

* Bước 3: Kiểm tra dữ liệu nhập vào:

|  |  |
| --- | --- |
| Tên trường | Yêu cầu kiểm tra |
| Tên tài khoản | * Tên tài khoản phái chính xác. * Nếu sai sẽ báo lỗi: “Tài khoản hoặc mật khẩu không đúng” * Để trống sẽ hiển thị: “Vui lòng nhập thông tin!” |
| Mật khẩu | * Mật khẩu phải chính xác * Nếu sai sẽ báo lỗi: “Tài khoản hoặc mật khẩu không đúng” * Để trống sẽ hiển thị: “Vui lòng nhập thông tin!” * Xem mật khẩu khi chưa mã hóa bằng icon “con mắt” |

* Bước 4: Người dùng lưu tài khoản click vào checkbox “Lưu tài khoản”
* Bước 5: Người dùng click “Đăng nhập”
* Bước 6: Hệ thống lưu tài khoản và mật khẩu của người dùng, hiển thị thông báo Lưu tài khoản thành công. Chuyển hướng đến màn hình Dashboard.
* Giao diện:

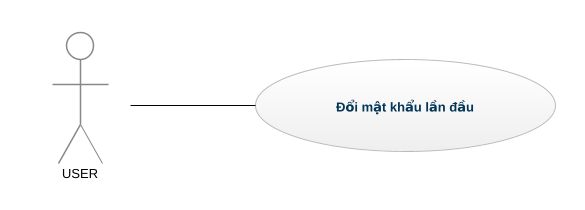


Hình : Giao diện màn hình Đăng nhập

### Chức năng “Đăng nhập lần đầu”

#### Biểu đồ Usecase cho chức năng “Đăng nhập lần đầu”

#### Đặc tả yêu cầu cho chức năng “Đăng nhập lần đầu”



Hình : Sơ đồ Use case cho chức năng Đổi mật khẩu lần đầu

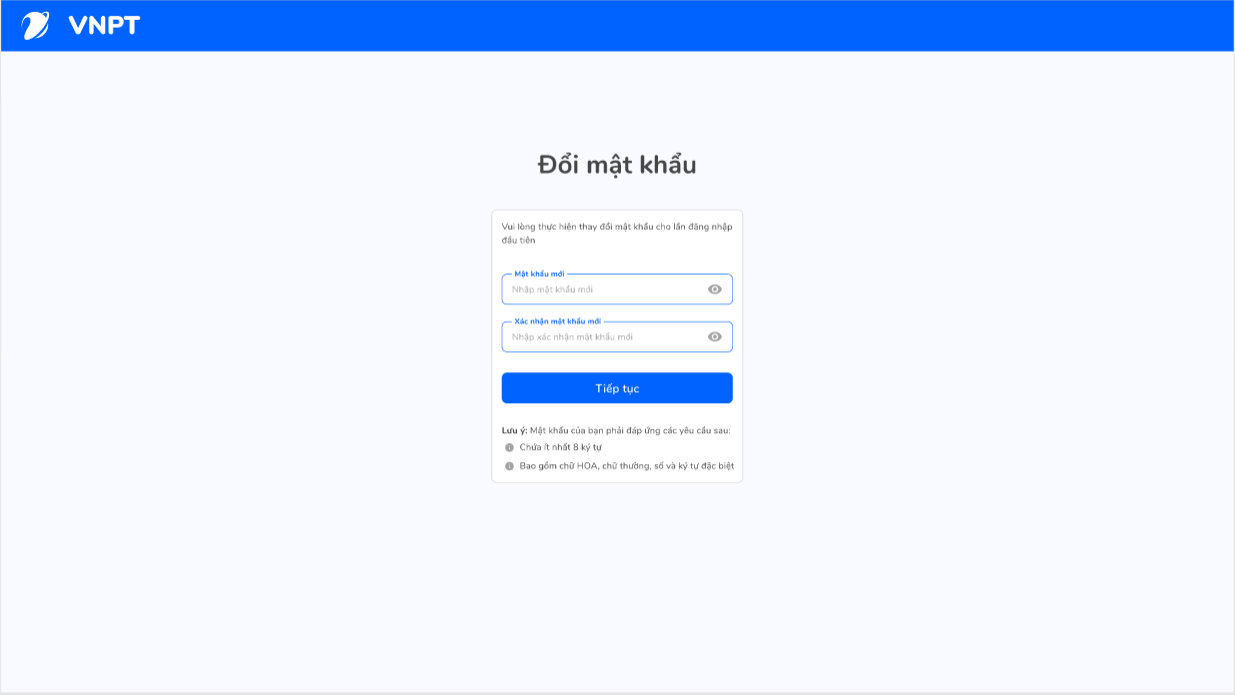
* Tác nhân: Admin, Nhân viên, Khách hàng
* Điều kiện tiên quyết: Người dùng đã nhận được tài khoản và đăng nhập thành công vào tài khoản.
* Mô tả khái quát Use case: Use case này thực hiện việc đổi mật khẩu lần đầu đăng nhập.
* Mô tả chi tiết:
  + Bước 1: Người dùng nhập các thông tin gồm:

|  |  |  |
| --- | --- | --- |
| Tên trường | Trường bắt buộc | Mục đích |
| Mật khẩu mới | X | Người dùng đổi mật khẩu để nâng cao bảo mật |
| Mật khẩu xác nhận mới | x | Để xác nhận lại mật khẩu có chính xác hay không. |

* Bước 2: Kiểm tra dữ liệu nhập vào:

|  |  |
| --- | --- |
| Tên trường | Yêu cầu kiểm tra |
| Mật khẩu mới | * Nhập mật khẩu mới thỏa mãn yêu cầu: “Chứa từ 8-50 ký tự. Phải bao gồm chữ HOA, thường, số và kí tự đặc biệt” * Nếu nhập sai: Yêu cầu hiện màu đỏ * Nếu nhập đúng: yêu cầu chuyển màu xanh * Xem mật khẩu dưới dạng không mã hóa bằng icon “con mắt” |
| Xác nhận mật khẩu mới | * Nhập xác nhận mật khẩu trùng với mật khẩu mới * Nếu sai sẽ thông báo “Mật khẩu mới và xác nhận mật khẩu mới không trùng khớp” * Xem xác nhận mật khẩu dưới dạng không mã hóa bằng icon “con mắt” * Nút “Tiếp tục” bị vô hiệu hóa |

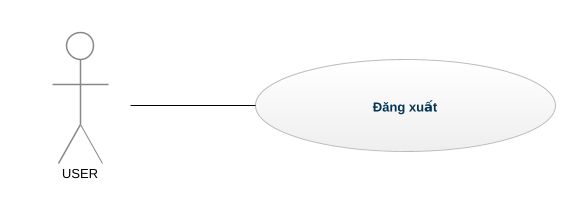
* Bước 3: Người dùng click “Tiếp tục” khi đã nhập hợp lệ
* Bước 4: Người dùng chuyển đến màn hình Dashboard
* Giao diện:



### Chức năng “Đăng xuất”

#### Biểu đồ Usecase cho chức năng “Đăng xuất”

#### Đặc tả yêu cầu cho chức năng “Đăng xuất”

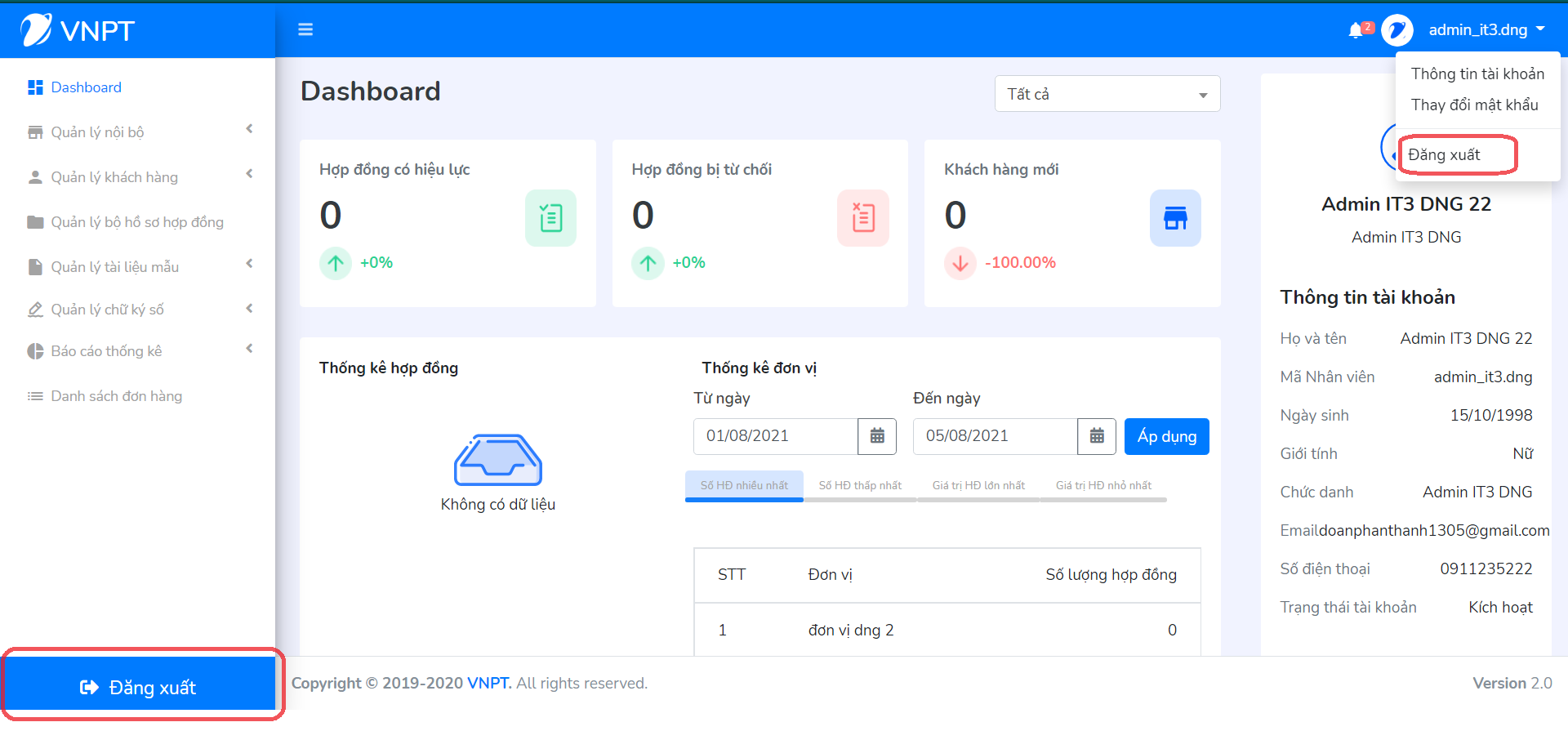


Hình : Sơ đồ Use case cho chức năng Đăng xuất

* Tác nhân: Admin, Nhân viên, Khách hàng
* Điều kiện tiên quyết: Đăng nhập thành công vào hệ thống, xuất hiện màn hình Dashboard
* Mô tra khái quát: Use case thực hiện việc đăng xuất tài khoản ra khỏi hệ thống.
* Mô tả chi tiết:
* Bước 1: Người dùng có thể thực hiện bằng 2 cách sau:

|  |  |
| --- | --- |
| Cách 1 | Click vào tên tài khoản đang login ở góc trên bên phải rồi click vào “Đăng xuất” |
| Cách 2 | Người dùng click vào “Đăng xuất” tại thanh menu bên trái màn hình |

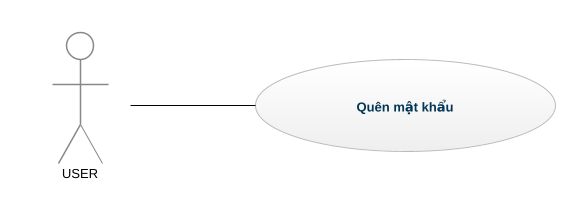
* Bước 2: Đăng xuất thành công. Chuyển hướng người dùng về màn hình “Đăng nhập”
* Giao diện:



### Chức năng “Quên mật khẩu”

#### Biểu đồ Usecase cho chức năng “Quên mật khẩu”

#### Đặc tả yêu cầu cho chức năng “Quên mật khẩu”



Hình : Sơ đồ Use case cho chức năng Quên mật khẩu

* Tác nhân: Admin, Nhân viên, Khách hàng
* Điều kiện tiên quyết: Truy cập thành công vào màn hình Đăng nhập
* Mô tra khái quát: Use case thực hiện việc thay đổi mật khẩu khi người dùng quên.
* Mô tả chi tiết:
* Bước 1: Nhấn vào đường link “Quên mật khẩu” ở màn hình Đăng nhập
* Bước 2: Người dùng được chuyển hướng đến màn hình Quên mật khẩu và nhập Tên tài khoản.
* Bước 3: Kiểm tra dữ liệu nhập vào:

|  |  |  |
| --- | --- | --- |
| Tên trường | Trường bắt buộc | Mục đích và yêu cầu |
| Tên tài khoản | X | * Để nhận biết tài khoản nào cần khôi phục và lấy dữ liệu về gmail và số điện thoại từ tên tài khoản đó. * Nếu sai: hiện thông báo “AbC không tồn tại" ( với AbC là tên tài khoản không tồn tại trên hệ thống)” * Tên tài khoản là tên đã tồn tại trong hệ thống. |

* Bước 4: Người dùng click “tiếp tục” chuyển hướng đến chọn phương thức khôi phục.

|  |  |  |
| --- | --- | --- |
| Tên trường | Trường bắt buộc | Mục đích và yêu cầu |
| Phương thức khôi phục | x | Lựa chọn phương thức cho việc khôi phục mật khẩu  Chọn 2 phương thức:   * Gửi đường link khôi phục tài khoản qua mail * Gửi mã xác nhận qua số điện thoại   Mặc định là đã chọn phương thức gửi qua mail. |

* Bước 5: Người dùng Click “Tiếp tục”:

Chuyển sang màn hình xác nhận tài khoản nếu chọn phương thức gửi mã qua số điện thoại.

Lưu ý: Người dùng chọn phương thức gửi mail sẽ hiện màn hình thông báo kiểm tra mail và truy cập đường link được gửi.

* Bước 6: Nhập mã hoặc truy cập link:

Đối với phương thức gửi mã, người dùng nhập mã được gửi qua số điện thoại lên hệ thống.

Đối với phương thức gửi mail, người dùng truy cập đường link được gửi.

* Bước 7: Kiểm tra dữ liệu nhập vào:

|  |  |  |
| --- | --- | --- |
| Tên trường | Trường bắt buộc | Yêu cầu |
| Mã xác nhận | x | Nhập mã từ tin nhắn vào hệ thống   * Nhập sai: Thông báo "Mã xác nhận không chính xác, vui lòng thử lại. Bạn còn 2 lần nhập mã xác nhận." * Nhập quá 3 lần sai: Hệ thống thông báo: “Hiển thi popup "Bạn đã sử dụng quá số lần nhập mã xác nhận. Vui lòng thực hiện lại" và quay lại màn hình ban đầu. * Mã xác nhận gửi quá 5 lần: Thông báo “Bạn đã sử dụng quá số lần gửi lại mã xác nhận. Vui lòng thử lại sau 30 phút.” * Mã xác nhận quá 5 phút: Thông báo “Mã xác nhận đã hết hạn. Vui lòng thực hiện gửi lại mã và nhập mã xác nhận mới.” |

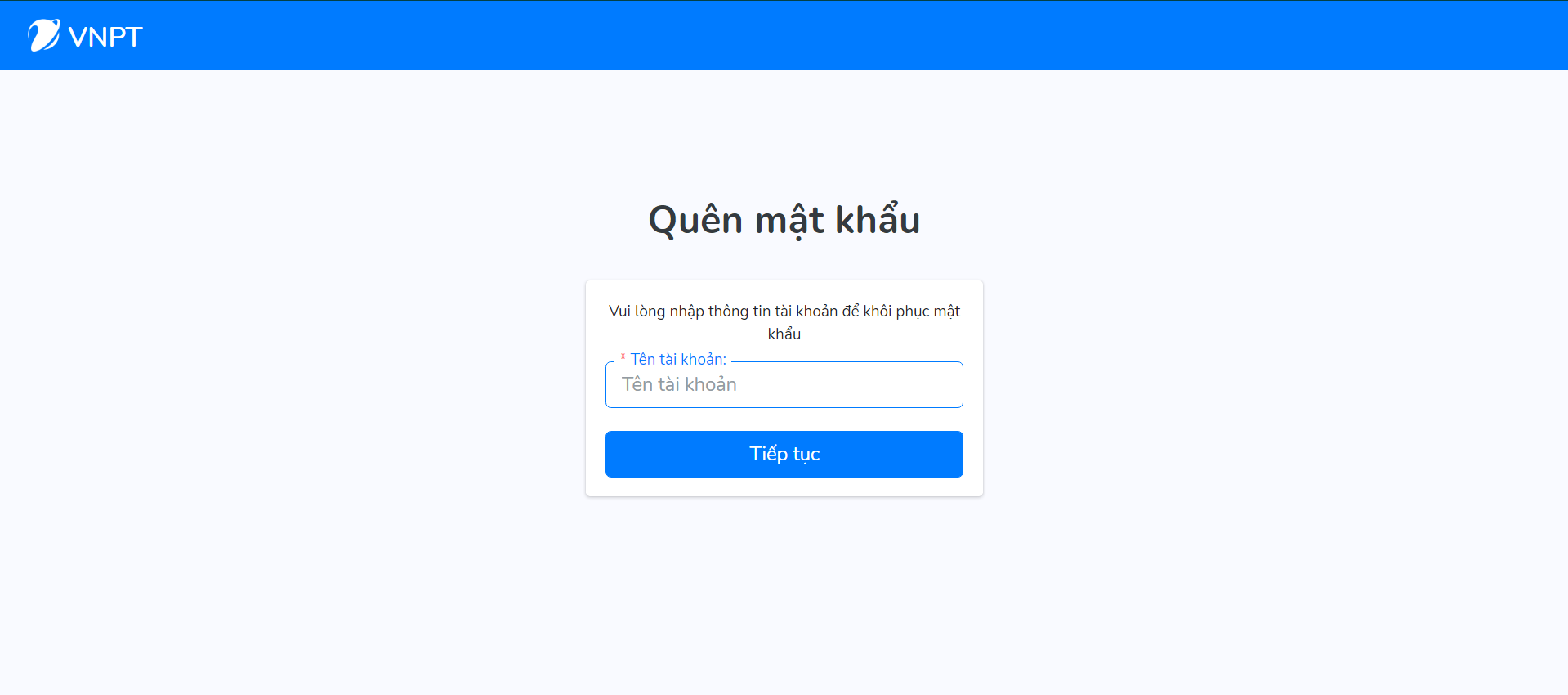
* Bước 7: Người dùng chuyển hướng sang màn hình Quên mật khẩu tiến hành đổi mật khẩu.
* Bước 8: Người dùng nhập các trường gồm:

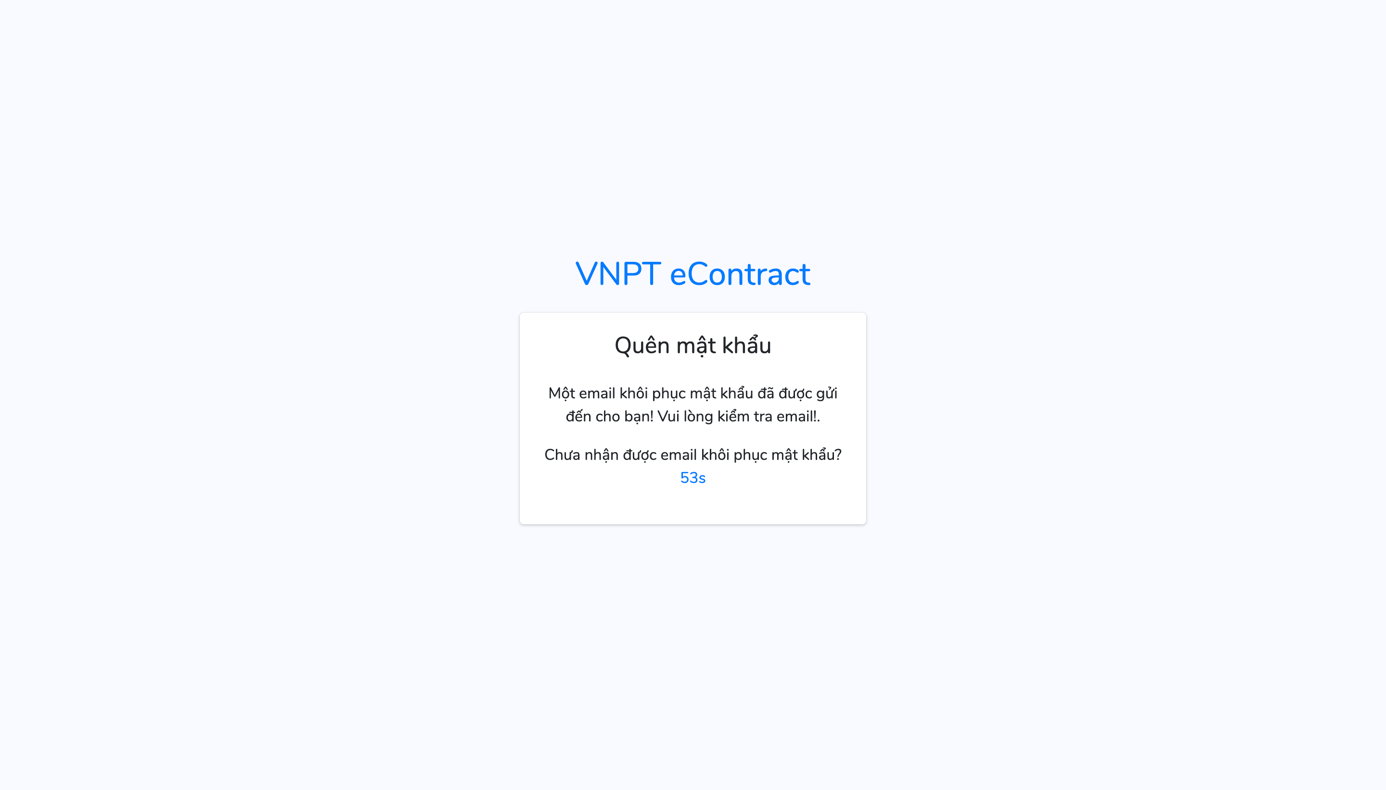
|  |  |  |
| --- | --- | --- |
| Tên trường | Trường bắt buộc | Mục đích |
| Mật khẩu mới | X | Người dùng đổi mật khẩu để nâng cao bảo mật |
| Nhập lại mật khẩu mới | x | Để xác nhận lại mật khẩu có chính xác hay không. |

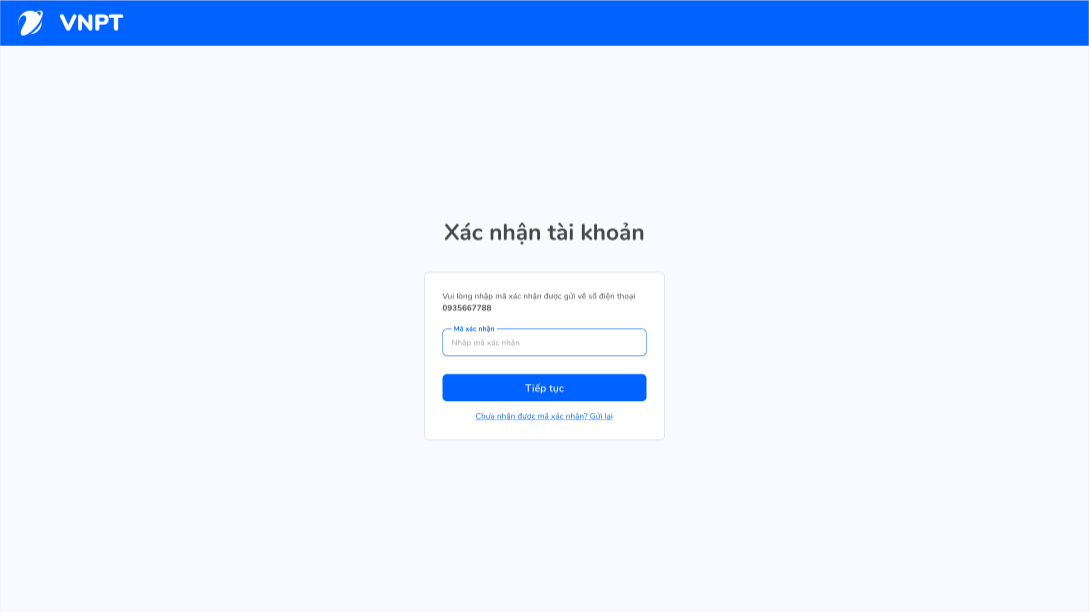
* Bước 9: Kiểm tra dữ liệu nhập vào:

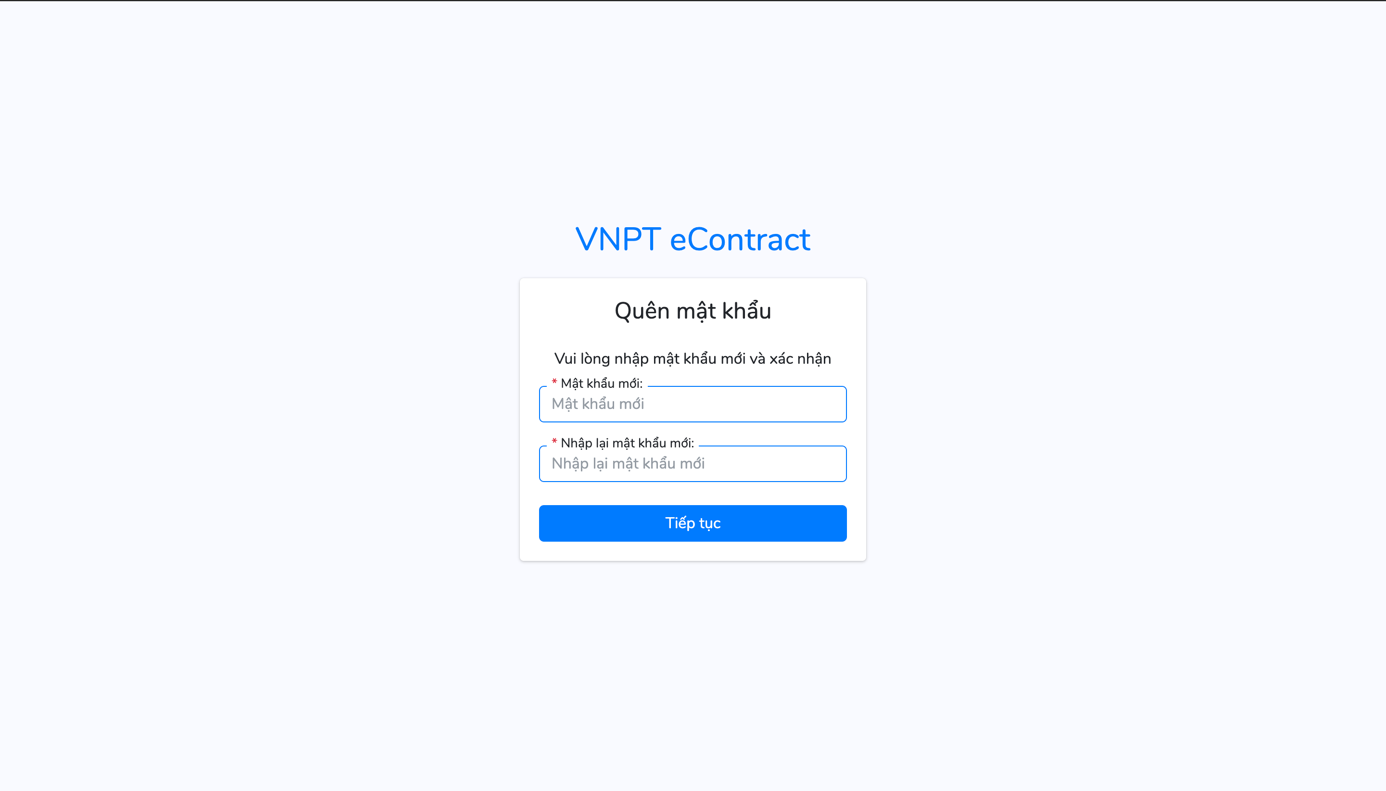
|  |  |
| --- | --- |
| Tên trường | Yêu cầu kiểm tra |
| Mật khẩu mới | * Nhập mật khẩu mới thỏa mãn yêu cầu: “Chứa từ 8-50 ký tự. Phải bao gồm chữ HOA, thường, số và kí tự đặc biệt” * Nếu nhập sai: Yêu cầu hiện màu đỏ * Nếu nhập đúng: yêu cầu chuyển màu xanh * Xem mật khẩu dưới dạng không mã hóa bằng icon “con mắt” |
| Nhập lại mật khẩu mới | * Nhập xác nhận mật khẩu trùng với mật khẩu mới * Nếu sai sẽ thông báo “Mật khẩu mới và Nhập lại mật khẩu mới không trùng khớp” * Xem nhập lại mật khẩu dưới dạng không mã hóa bằng icon “con mắt” |

* Bước 10: Người dùng click “Tiếp tục” khi đã nhập hợp lệ
* Bước 11: Người dùng chuyển đến màn hình Dashboard
* Giao diện:





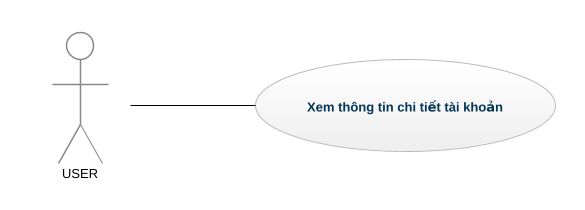




### Chức năng “Xem thông tin chi tiết tài khoản”

#### Biểu đồ Usecase cho chức năng “Xem thông tin chi tiết tài khoản”

#### Đặc tả yêu cầu cho chức năng “Xem thông tin chi tiết tài khoản”

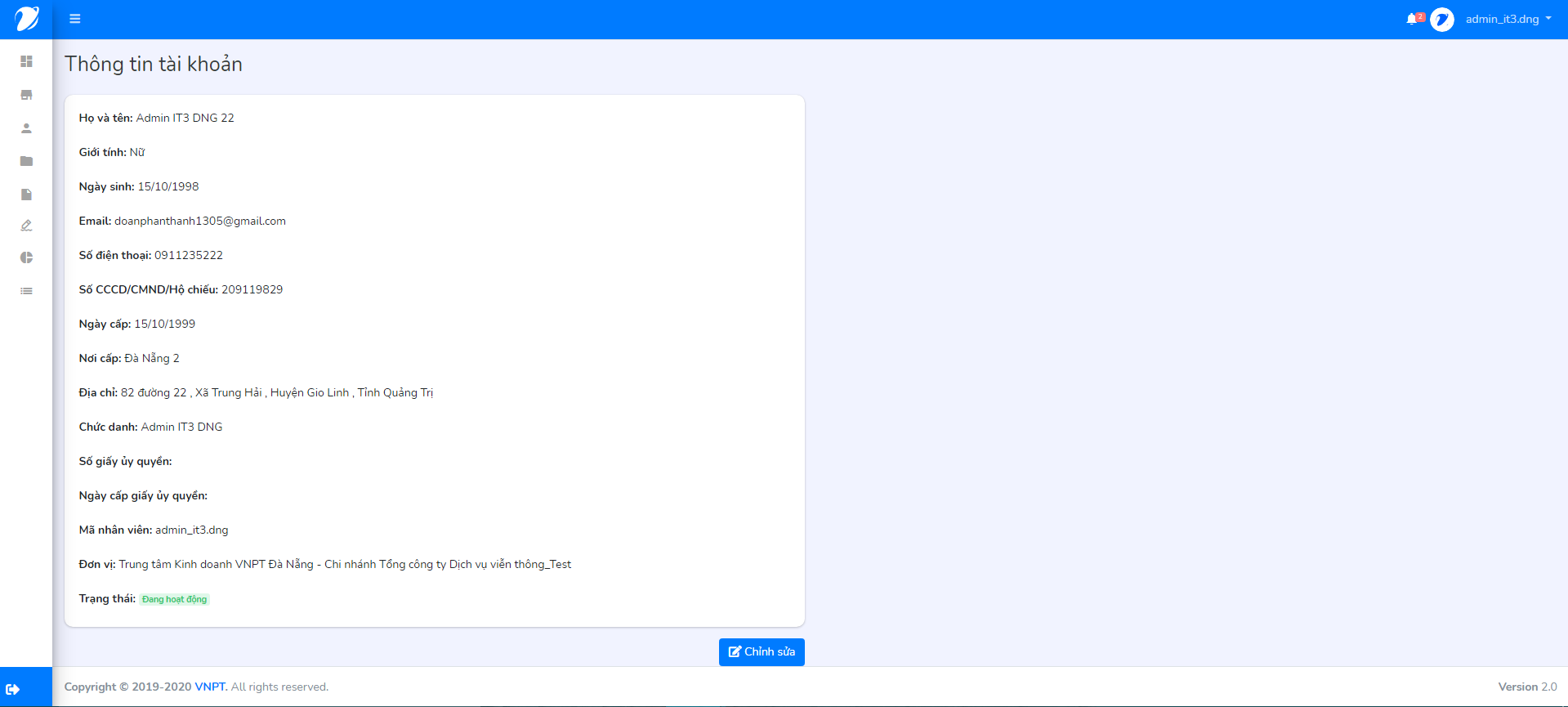


Hình : Sơ đồ Use case cho chức năng Xem thông tin chi tiết tài khoản

* Tác nhân: Admin, Nhân viên, Khách hàng
* Điều kiện tiên quyết: Đăng nhập thành công vào hệ thống.
* Mô tra khái quát: Use case thực hiện việc xem chi tiết thông tin tài khoản. Người dùng có thể xem tất cả thông tin của mình.
* Mô tả chi tiết:
* Bước 1: Người dùng click vào tên tài khoản đang login ở góc trên bên phải rồi click vào “Thông tin tài khoản”
* Bước 2: Người dùng được đưa đến trang “Thông tin tài khoản” và có các thông tin cần thiết gồm:

|  |  |
| --- | --- |
| Tên trường | Bắt buộc |
| Họ và tên | X |
| Giới tính | X |
| Ngày sinh | X |
| Email | X |
| Số điện thoại | X |
| Số CCCD/CMND/Hộ chiếu | X |
| Ngày cấp | X |
| Nơi cấp | X |
| Địa chỉ | X |
| Chức danh | X |
| Số giấy ủy quyền |  |
| Ngày cấp giấy ủy quyền |  |
| Mã nhân viên | X |
| Đơn vị | X |
| Trạng thái | X |

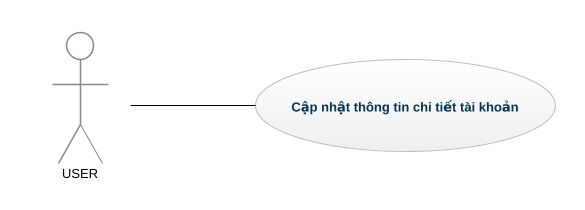
* Giao diện:



### Chức năng “Cập nhật thông tin chi tiết tài khoản”

#### Biểu đồ Usecase cho chức năng “Cập nhật thông tin chi tiết tài khoản”

#### Đặc tả yêu cầu cho chức năng “Cập nhật thông tin chi tiết tài khoản”



Hình : Sơ đồ Use case cho chức năng Cập nhật thông tin chi tiết tài khoản

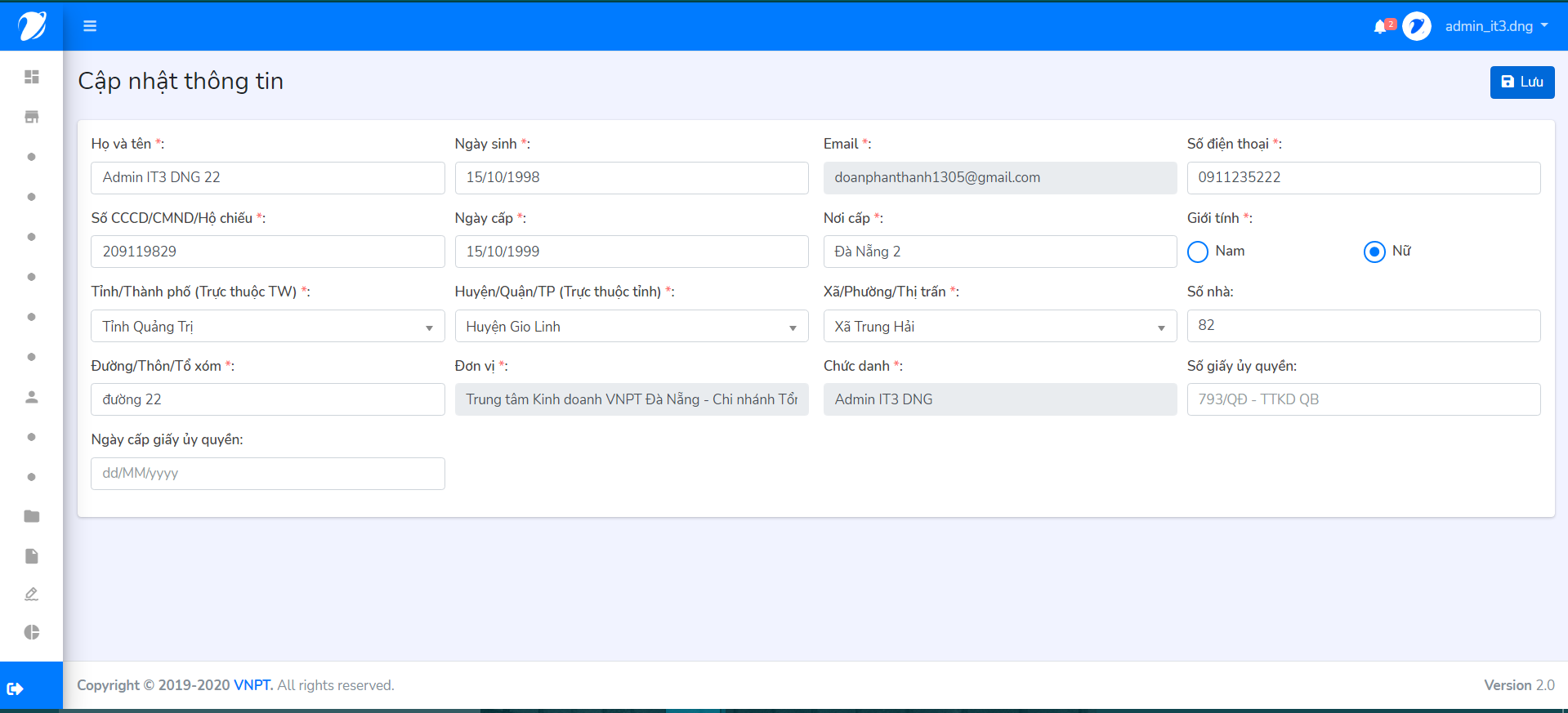
* Tác nhân: Admin, Nhân viên, Khách hàng
* Điều kiện tiên quyết: Đăng nhập thành công vào hệ thống, vào được trang “Thông tin tài khoản”.
* Mô tra khái quát: Use case thực hiện việc chỉnh sửa thông tin người dùng.
* Mô tả chi tiết:
* Bước 1: Người dùng click vào nút Chỉnh sửa để tiến hành cập nhật thông tin
* Bước 2: Người dùng tiến hành cập nhật thông tin. Các trường trong trang “Cập nhật thông tin” bao gồm:

|  |  |
| --- | --- |
| Tên trường | Bắc buộc |
| Họ và tên | X |
| Ngày sinh | X |
| Email | X |
| Số điện thoại | X |
| Số CCCD/CMND/Hộ chiếu | X |
| Ngày cấp | X |
| Nơi cấp | X |
| Giới tính | X |
| Tỉnh/ Thành phố (Trược thuộc TW) | X |
| Huyện/ Quận/TP (Trực thuộc tỉnh) | X |
| Xã/phường/Thị trấn | X |
| Số nhà |  |
| Đường/Thôn/Tổ xóm | X |
| Đơn vị | X |
| Chức danh | X |
| Số giấy ủy quyền |  |
| Ngày cấp giấy ủy quyền |  |

* Bước 3: Kiểm tra dữ liệu nhập vào:

|  |  |
| --- | --- |
| Tên trường | Yêu cầu |
| Họ và tên | * Chứa ký tự chữ, số, kí tự đặc biệt * Không được để trống * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Ngày sinh | * Ngày đúng theo định dạng dd/mm/yyyy * Sai định dạng như trên: Thông báo “Sai định dạng ngày tháng năm!” * Không được ghi ngày tương lai. Nếu sai thì hiện thông báo “Ngày sinh không được là ngày tương lai!” * Các ngày tồn tại theo ngày trong lịch * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Email | * Bị vô hiệu hóa, không thể chỉnh sửa |
| Số điện thoại | * Số điện thoại đủ 10 số * Đúng định dạng số điện thoại (bắt đầu bằng 0) * Nhập sai định dạng: Thông báo “Sai định dạng số điện thoại!” * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Số CCCD/CMND/Hộ chiếu | * Đủ 9 kí tự số (CMND) hoặc 8 kí tự số+chữ (Hộ chiếu) và 12 số (CCCD đúng định dạng * Nhập sai: Thông báo “Số CCCD/CMND/Hộ chiếu sai định dạng!” |
| Ngày cấp | * Ngày đúng theo định dạng dd/mm/yyyy * Sai định dạng như trên: Thông báo “Sai định dạng ngày tháng năm!” * Không được ghi ngày tương lai. Nếu sai thì hiện thông báo “Ngày sinh không được là ngày tương lai!” * Các ngày tồn tại theo ngày trong lịch * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Nơi cấp | * Có thể nhập chữ, số, kí tự đặc biệt * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Giới tính | * Có thể chọn 1 trong 2 * Không thể chọn cả 2 hay không chọn cả 2 |
| Tỉnh/ Thành phố (Trược thuộc TW) | * Nhập kí tự vào thanh tìm kiếm, có thể tìm theo từ hoặc theo kí tự * Chọn trong danh sách tỉnh/thành phố. * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Huyện/ Quận/ TP (Trực thuộc tỉnh) | * Nhập kí tự vào thanh tìm kiếm, có thể tìm theo từ hoặc theo kí tự * Chọn trong danh sách tỉnh/thành phố. * Chọn sau Tỉnh/Thành phố. * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Xã/ Phường/ Thị trấn | * Nhập kí tự vào thanh tìm kiếm, có thể tìm theo từ hoặc theo kí tự * Chọn trong danh sách tỉnh/thành phố. * Chọn sau Huyện/Quận/TP. * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Số nhà | * Chứa chữ, số, kí tự đặc biệt * Có thể không nhập |
| Đường/Thôn/Tổ xóm | * Chứa chữ, số, kí tự đặc biệt * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |
| Đơn vị | * Bị vô hiệu hóa, không thể chỉnh sửa |
| Chức danh | * Bị vô hiệu hóa, không thể chỉnh sửa |
| Số giấy ủy quyền | * Chứa chữ, số, kí tự đặc biệt * Có thể không nhập |
| Ngày cấp giấy ủy quyền | * Ngày đúng theo định dạng dd/mm/yyyy * Sai định dạng như trên: Thông báo “Sai định dạng ngày tháng năm!” * Không được ghi ngày tương lai. Nếu sai thì hiện thông báo “Ngày sinh không được là ngày tương lai!” * Các ngày tồn tại theo ngày trong lịch * Nếu để trống: Thông báo “Vui lòng nhập thông tin!” |

* Bước 4: Người dùng click nút Lưu xuất hiện popup xác nhận “Bạn có chắc chắn muốn lưu thông tin?”
* Bước 5: Người dùng click Đồng ý, hệ thông lưu thông tin.
* Giao diện:



# CHƯƠNG 3. KẾT QUẢ



## Kết quả thực hiện thiết kế testcase

### Kết quả viết checklist:

#### Checklist Đăng nhập

|  |  |  |  |
| --- | --- | --- | --- |
| **Checklist đăng nhập** | | | |
| **Stt** |  | **Mô tả test case** | **Kết quả mong đợi** |
| 1 | **Giao diện mặc định** | Kiểm tra mở màn hình [đăng nhập] | Màn hình [đăng nhập] hiển thị thông công |
| 2 | Kiểm tra hiển thị giao diện màn hình [đăng nhập] | 1. Hiển thị gồm các item:  - logo vnpt  - text "vnpt" - title "đăng nhập" - textbox [tài khoản] - textbox [mật khẩu]  + icon "con mắt" - checkbox [lưu mật khẩu] - hyperlink [quên mật khẩu?] - button [đăng nhập] |
| 3 | **Test textbox [tài khoản]** | Kiểm tra hiển thị placeholder | Hiển thị placeholder "tài khoản" |
| 5 | Kiểm tra hiển thị con trỏ chuột khi đưa đến textbox | Hiển thị i |
| 6 | Trường hợp nhập dữ liệu có space 2 đầu | Tự động bỏ ký tự khoảng cách ở 2 đầu |
| 7 | Kiểm tra trường hợp khi để trống [tên tài khoản] click [đăng nhập] | Hiển thị "vui lòng nhập thông tin!" dưới textbox [tài khoản] |
|  | Nhập chữ, số, kí tự đặc biệt | Hiển thị đầy đủ và đúng kí tự đã nhập |
| 8 | **Test textbox [mật khẩu]** | Kiểm tra hiển thị placeholder | Hiển thị placeholder "mật khẩu" |
| 10 | Kiểm tra hiển thị con trỏ chuột khi đưa đến textbox | Hiển thị i |
| 11 | Kiểm tra trường hợp không nhập | Hiển thị "vui lòng nhập thông tin!" dưới textbox [mật khẩu] |
| 12 | Kiểm tra các ký tự nhập vào ở mật khẩu được mã hóa | Các ký tự nhập vào được mã hóa:  " |
| 13 | Kiểm tra hiển thị dữ liệu khi dùng icon "con mắt" | Các ký tự ở dạng mã hóa được hiển thị ở dạng các ký tự |
| 14 | Kiểm tra khi nhập số, chữ, kí tự đặc biệt | Hiển thị đầy đủ và đúng kí tự đã nhập |
| 15 | **Đăng nhập thành công** | Nhập tài khoản và mật khẩu đúng click đăng nhập | Đăng nhập thành công, chuyển hướng đến màn hình [dashboard] |
| 16 | **Đăng nhập không thành công** | Nhập tài khoản sai click đăng nhập | Hệ thống báo lỗi "tài khoản hoặc mật khẩu không đúng" |
| 17 | Nhập mật khẩu sai click đăng nhập | Hệ thống báo lỗi "tài khoản hoặc mật khẩu không đúng" |
|  | **Test checkbox [lưu tài khoản]** | Kiểm tra mặc định của checkbox | không hiển thị dấu tick chọn tại checkbox [lưu tài khoản] |
| 18 | Trường hợp tick chọn checkbox [lưu tài khoản] | Hiển thị dấu tick chọn tại checkbox [lưu tài khoản] |
| 19 | Trường hợp tick chọn nhiều lần checkbox [lưu tài khoản] | không hiển thị dấu tick chọn tại checkbox [lưu tài khoản] |
| 20 | Kiểm tra chức năng của checkbox [lưu tài khoản] khi nhập đúng thông tin đăng nhập | - màn hình dashboard hiển thị. Đăng nhập vào hệ thống thành công bằng tài khoản đã lưu mật khẩu - hiển thị mật khẩu đã lưu trước đó ở textbox [mật khẩu] dưới dạng mã hóa |
|  | Kiểm tra chức năng của checkbox [lưu tài khoản] khi nhập sai thông tin đăng nhập |  |
| 21 | **Test hypelink [quên tài khoản]** | Kiểm tra hiển thị con trỏ chuột | Trạng thái của chuột sẽ chuyển thành ký tự bàn tay và highlight hyperlink |
| 22 | Kiểm tra trường hợp click vào hypelink có hoạt động | Chuyển đến màn hình [quên mật khẩu] với textbox [tên tài khoản] |

#### Checklist Đổi mật khẩu lần đầu:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Kịch bản kiểm thử đổi mật khẩu lần đầu** | | | | |
| Stt |  |  | Mô tả test case |  |
| 1 | Giao diện mặc định |  | Hiển thị màn hình [đổi mật khẩu] cho lần đầu |  |
| 2 |  | Kiểm tra hiển thị giao diện màn hình [đổi mật khẩu] | màn hình [đổi mật khẩu lần đầu đăng nhập] hiển thị gồm: - header hiển thị bao gồm các item từ trái sang phải:  + logo vnpt  + text "vnpt" - body hiển thị gồm các item:  + title "đổi mật khẩu"  + text "vui lòng thực hiện thay đổi mật khẩu cho lần đầu đăng nhập"  + textbox [mật khẩu mới], icon "con mắt"  + textbox [xác nhận mật khẩu mới], icon "con mắt"  + button [tiếp tục]  + text "lưu ý:mật khẩu của bạn phải đáp ứng các yêu cầu sau:"  + text "chứa từ 8-50 ký tự"  + text "phải bao gồm chữ hoa, chữ thường, số và ký tự đặc biệt" |
| 3 |  | Kiểm tra các ký tự nhập vào ở mật khẩu được mã hóa | Các ký tự nhập vào được mã hóa:  "\*\*\*\*\*\*\*\*\*\*\*" |
| 4 |  | Kiểm tra các ký tự nhập vào ở mật khẩu được hiển thị khi click icon [con mắt] | Các ký tự ở dạng mã hóa được hiển thị ở dạng các ký tự 'matkhau@123" |
| 5 | Test textbox [mật khẩu mới] |  | Kiểm tra hiển thị placehoder | Hiển thị placehoder "nhập mật khẩu mới" |
| 6 |  | Khi đưa chuột đến textbox | Hiển thị con trỏ chuột hiện i |
| 7 |  | Để trống textbox [mật khẩu mới] và click "tiếp tục | Button [tiếp tục] ở trạng thái disable |
| 8 |  | Trường hợp nhập 8 ký tự không hợp lệ: chữ thường + số + ký tự đặc biệt | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 9 | Trường hợp nhập 8 ký tự không hợp lệ: chữ hoa + số + ký tự đặc biệt | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 10 | Trường hợp nhập 8 ký tự không hợp lệ: chữ thường + chữ hoa + ký tự đặc biệt | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 11 | Trường hợp nhập 8 ký tự không hợp lệ: chữ thường + chữ hoa + số | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 12 | Kiểm tra hiển thị khi nhập ít hơn 8 kí tự | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 13 | Kiểm tra hiển thị khi nhập quá 50 ký tự | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 14 | Kiểm tra hiển thị khi nhập đủ 50 kí tự | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 15 | Kiểm tra hiển thị khi nhập đủ 8 kí tự | - hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu đỏ |
| 16 |  | Kiểm tra hiển thị mật khẩu khi nhập đúng yêu cầu | - hệ thống không hiển thị thông báo lỗi tại mục lưu ý, dòng yêu cầu :" chứa từ 8-50 ký tự - phải bao gồm chữ hoa, thường, số và kí tự đặc biệt" chuyển sang màu xanh |
| 17 | Test button icon con mắt | Kiểm tra con trỏ chuột đưa đến button | Hiển thị con trỏ hình bàn tay |
| 18 | Kiểm tra hiển thị kí tự của mật khẩu khi click icon | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| 19 | Kiểm tra hiển thị khi click icon [con mắt] nhiều lần | Phụ thuộc vào hoạt động bật/tắt con mắt mà kết quả hiển thị khác nhau |
| 20 | Test textbox [xác nhận mật khẩu mới] |  | Kiểm tra placehoder có hiển thị | Hiển thị "nhập xác nhận mật khẩu mới" |
| 21 |  | Để trống textbox và click "tiếp tục" | Hiển thị "vui lòng nhập thông tin!" |
| 22 |  | Trường hợp nhập không trùng với mật khẩu mới | Hệ thống báo lỗi "xác nhận mật khẩu mới và mật khẩu mới không trùng khớp!" dưới textbox [xác nhận mật khẩu mới] |
| 23 |  | Trường hợp nhập trùng với mật khẩu mới | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| 24 | Test button icon con mắt | Khi đưa chuột đến textbox con trỏ chuột hiện hình bàn tay | Hiển thị con trỏ hình bàn tay |
| 25 | Kiểm tra hiển thị kí tự của mật khẩu khi click icon | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| 26 | Kiểm tra hiển thị khi click icon [con mắt] nhiều lần | Phụ thuộc vào hoạt động bật/tắt con măt mà kết quả hiển thị khác nhau |
| 27 | Kiểm tra tính hợp lệ của button [tiếp tục] |  | Để trống tất cả các trường | Button [tiếp tục] disable |
| 28 |  | Nhập k hợp lệ | Button [tiếp tục] disable |
| 29 |  | Nhập mật khẩu mới và xác nhận mật khẩu k đồng nhất | Button [tiếp tục] disable |
| 30 |  | Nhập giá trị hợp lệ ở tất cả các trường | Button [tiếp tục] enable |
| 31 | Kiểm tra chức năng của button [tiếp tục] |  | Nhập [mật khẩu mới] và [xác nhận mật khẩu] click [tiếp tục] | Di chuyển đến màn hình [dashboard] |
| 32 | Kiểm tra đăng nhập lại thành công sau khi đã đổi mật khẩu mới |  | Đăng nhập vào tài khoản đã đăng ký | Đăng nhập thành công |

#### Checklist Đăng xuất

|  |  |  |  |
| --- | --- | --- | --- |
| **Kiểm thử phần đăng xuất** | | | |
| Stt |  | **Mô tả test case** |  |
| 1 | **Đăng xuất thành công** | Kiểm tra khi click chức năng [đăng xuất] tại tên tài khoản đang login | Đăng xuất thành công và chuyển hướng sang trang [đăng nhập] |
| 2 | Kiểm tra khi click vào button [đăng xuất] tại thanh bên trái màn hình | Đăng xuất thành công và chuyển hướng sang trang [đăng nhập] |
| 3 | Trường hợp chọn icon đăng xuất tại màn hình dashboard | Đăng xuất thành công và chuyển hướng sang trang [đăng nhập] |

#### Checklist Quên mật khẩu

|  |  |  |  |
| --- | --- | --- | --- |
| **Kiểm thử phần quên mật khẩu** | | | |
|  |  | **Mô tả test case** |  |
| **Màn hình [quên mật khẩu]** |  | Kiểm tra mở màn hình [quên mật khẩu] | Màn hình [quên mật khẩu] hiển thị thành công |
|  | Kiểm tra hiển thị giao diện màn hình [quên mật khẩu] | Hiển thị gồm các item: + logo "vnpt" + text "vnpt" - title "quên mật khẩu" - hiển thị text thông báo là "vui lòng nhập thông tin tài khoản để khôi phục mật khẩu" - textbox [tên tài khoản] - button [tiếp tục] |
| **Kiểm tra tính hợp lệ của textbox [tên tài khoản]** | Kiểm tra hiển thị con trỏ chuột | Hiển thị con trỏ hình i |
| Kiểm tra hiển thị placehoder | Hiển thị placehoder "tên tài khoản" |
| 1. Để trống textbox [tên tài khoản]  2. Click [tiếp tục] | Hệ thống báo lỗi "vui lòng nhập thông tin!" dưới textbox [tên tài khoản] |
| 1. Nhập chữ + số + ký tự đặc biệt tại textbox [tên tài khoản] 2. Click button [tiếp tục] | Hiển thị đúng và đầy đủ dữ liệu nhập |
| **Kiểm tra chức năng của button [tiếp tục]** | Trường hợp nhập tên tài khoản chưa tồn tại trên hệ thống | Hệ thống báo lỗi "abc không tồn tại" ( với abc là tên tài khoản không tồn tại trên hệ thống) |
| Trường hợp nhập tên tài khoản đã tồn tại trên hệ thống | Chuyển qua màn hình [chọn phương thức khôi phục] |
|  | Kiểm tra mở màn hình [quên mật khẩu] chọn phương thức khôi phục |  |
|  | Kiểm tra hiển thị giao diện màn hình [chọn phương thức khôi phục] | - hiển thị gồm các item: + logo "vnpt" + text "vnpt" + title "quên mật khẩu" - hiển thị text thông báo là "tài khoản của bạn đã thực hiện liên kết với email và số điện thoại. Vui lòng chọn hình thức xác thực tài khoản" - radio button [gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*.vn] - radio button [gửi mã xác nhận về số điện thoại 09\*\*\*\*\*\*\*8] - button [tiếp tục] |
| **Kiểm tra chức năng của radio button** | Kiểm tra hiển thị mặc định | Hiển thị mặc định chọn "gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*.vn" |
| Trường hợp thay đổi chọn 1 giá trị bất kỳ  (step: click chọn option [gửi mã xác nhận về số điện thoại 09\*\*\*\*\*\*\*8]) | Hệ thống hiển thị tick chọn radio "gửi mã xác nhận về số điện thoại 09\*\*\*\*\*\*\*8" |
| Trường hợp chọn nhiều giá trị cùng 1 lúc (step:  1. Click chọn option [gửi mã xác nhận về số điện thoại 09\*\*\*\*\*\*\*8] 2. Click chọn option [gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*\*\*.vn]) 3. Kiểm tra hiển thị | 3.  - không thể chọn 2 radio cùng 1 lúc - hệ thống hiển thị tick chọn tại option 'gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*\*\*.vn' |
| Trường hợp bỏ chọn radio button (step: 1. Hệ thống đang hiển thị giá trị tại option 'gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*.vn' 2.click chọn radio button tại giá trị 'gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*.vn' để bỏ check | - không thể bỏ chọn radio button - hệ thống hiển thị dấu tick tại option = 'gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*.vn' |
| **Kiểm tra button [tiếp tục]** | Chọn phương thức gửi mail và click button [tiếp tục] | Chuyển đến màn hình [thông báo kiểm tra email] |
| Chọn phương thức gửi số điện thoại và click button [tiếp tục] | Chuyển đến màn hình [xác nhận tài khoản] |
| **Màn hình [xác nhận tài khoản]** |  | ~~Kiểm tra mở màn hình [thông báo kiểm tra email]~~ |  |
|  | Kiểm tra hiển thị giao diện màn hình [thông báo kiểm tra email] | **Chuyển sang màn hình [xác nhận tài khoản] thành công hiển thị gồm các item: + logo "vnpt" + text "vnpt" - title "xác nhận tài khoản" - hiển thị text thông báo là "một email đã được gửi về địa chỉ n\*\*\*@\*\*.vn. Vui lòng truy cập đường dẫn trong email để khôi phục mật khẩu." (với "n\*\*\*@\*\*.vn" là tài khoản người dùng) - text [gửi lại email sau 01:59s] và sẽ tự động đếm ngược** |
|  | Trường hợp quá thời gian chờ kích hoạt mail | Tại màn hình [xác nhận tài khoản], hiển thị gồm các item:  + logo "vnpt"  + text "vnpt" - title "xác nhận tài khoản" - hiển thị text thông báo là "một email đã được gửi về địa chỉ n\*\*\*@\*\*\*\*.vn. Vui lòng truy cập đường dẫn trong email để khôi phục mật khẩu." (với "n\*\*\*@\*\*\*\*.vn" là tài khoản người dùng) - hypelink [chưa nhận được email? Gửi lại] |
| **Hypelink "gửi lại"** | Kiểm tra hiển thị con trỏ chuột | Hiển thị con trỏ hình bàn tay |
| Kiểm tra hiển thị khi click vào hypelink [chưa nhận được email? Gửi lại] | Hiển thị "gửi lại email sau 1:59s" và sẽ tự động đếm ngược |
|  | Kiểm tra mở màn hình [xác nhận tài khoản] |  |
|  | Kiểm tra hiển thị giao diện màn hình [xác nhận tài khoản] | Chuyển sang màn hình [xác nhận tài khoản] thành công hiển thị gồm các item: + logo "vnpt" + text "vnpt" - title "xác nhận mật khẩu" - hiển thị text thông báo là "vui lòng nhập mã xác nhận được gửi về số điện thoại 0901233444" (với "0901233488" là tài khoản người dùng) - textbox [mã xác nhận] - button [tiếp tục] - hypelink "chưa nhận được mã xác nhận? Gửi lại?" |
| **Kiểm tra textbox [mã xác nhận]** | Kiểm tra hiển thị placehoder | Hiển thị placehoder "nhập mã xác nhận" |
| Trường hợp nhập dữ liệu có sapce 2 đầu | Hiển thị thông báo lỗi: "mã otp không đúng định dạng!" |
| Nhập đúng mã đã nhận từ tin nhắn click [tiếp tục] | Chuyển đến màn hình [đổi mật khẩu] thành công |
| Nhập mã sai click [tiếp tục] | Hiển thi popup [thông báo] với: text: "mã xác nhận không chính xác, vui lòng thử lại. Bạn còn 2 lần nhập mã xác nhận." button [đóng] |
| Nhập quá 3 lần sai | Hiển thi popup [thông báo] với: - text: " bạn đã sử dụng quá số lần nhập mã xác nhận. Vui lòng thực hiện lại" - button [đóng] |
| Mã xác nhận hết hạn quá 5 phút | Hiển thi popup [thông báo] với - text:"mã xác nhận đã hết hạn. Vui lòng thực hiện gửi lại mã và nhập mã xác nhận mới." - button [đóng] |
| Mã xác nhận gửi lại quá 5 lần | Hiển thi popup [thông báo] với: -text: "bạn đã sử dụng quá số lần gửi lại mã xác nhận. Vui lòng thử lại sau 30 phút." - button [đóng] |
| **Kiểm tra [hypelink ]** | Trường hợp click hypelink [chưa nhận được mã xác nhận? Gửi lại] | Hiển thị "gửi lại mã xác nhận sau 01:53s" và sẽ tự động đếm ngược |
| **Màn hình [quên mật khẩu]** | Truy cập từ link của mail | Kiểm tra mở màn hình [quên mật khẩu] |  |
|  | Kiểm tra hiển thị giao diện màn hình [quên mật khẩu] | Hệ thống hiển thị đầy đủ tất cả các item:  màn hình [quên mật khẩu] hiển thị gồm: - header hiển thị bao gồm các item từ trái sang phải:  - logo "vnpt"  - text "vnpt" - body hiển thị gồm các item: - text [vui lòng nhập thông tin sau] - textbox [mật khẩu mới] - textbox [xác nhận mật khẩu mới] - button [tiếp tục] |
| **Test textbox [mật khẩu mới]** | Kiểm tra hiển thị placehoder | Hiển thị placehoder "mật khẩu mới" |
| Khi đưa chuột đến textbox | Hiển thị con trỏ chuột hiện i |
| Để trống textbox [mật khẩu mới] và click [tiếp tục] | Hiển thị "vui lòng nhập thông tin!" dưới textbox [mật khẩu mới] |
| Nhập đúng mật khẩu 8 kí tự bấm tiếp tục | Hiển thị đúng và đủ kí tự vừa nhập |
| Nhập đúng mật khẩu 50 kí tự bấm tiếp tục | Hiển thị đúng và đủ kí tự vừa nhập |
| Nhập chữ hoa + thường+ số + kí tự đặc biệt | Hiển thị đúng và đủ kí tự vừa nhập |
| Nhập chữ thường + số + kí tự đặc biệt | Hệ thống báo lỗi "mật khẩu có độ dài từ 8-50 ký tự bao gồm ký tự hoa, thường, số và ký tự đặc biệt!" |
| Nhập chữ hoa + số+ kí tự đặc biệt | Hệ thống báo lỗi "mật khẩu có độ dài từ 8-50 ký tự bao gồm ký tự hoa, thường, số và ký tự đặc biệt!" dưới textbox [mật khẩu mới] |
| Nhập chữ hoa + chữ thường + kí tự đặc biệt | Hệ thống báo lỗi "mật khẩu có độ dài từ 8-50 ký tự bao gồm ký tự hoa, thường, số và ký tự đặc biệt!" dưới textbox [mật khẩu mới] |
| Nhập chữ hoa+ chữ thường + số | Hệ thống báo lỗi "mật khẩu có độ dài từ 8-50 ký tự bao gồm ký tự hoa, thường, số và ký tự đặc biệt!" dưới textbox [mật khẩu mới] |
| Nhập sai mật khẩu dưới 8 kí tự bấm tiếp tục | Hệ thống báo lỗi "mật khẩu có độ dài từ 8-50 ký tự bao gồm ký tự hoa, thường, số và ký tự đặc biệt!" dưới textbox [mật khẩu mới] |
| Nhập sai mật khẩu trên 50 kí tự bấm tiếp tục | Hệ thống báo lỗi "mật khẩu có độ dài từ 8-50 ký tự bao gồm ký tự hoa, thường, số và ký tự đặc biệt!" |
| **Test textbox [nhập lại mật khẩu mới]** | Kiểm tra hiển thị placehoder | Hiển thị placehoder "nhập lại mật khẩu mới" |
| Khi đưa chuột đến textbox | Hiển thị con trỏ chuột hiện i |
| Để trống textbox [nhập lại mật khẩu mới] và click [tiếp tục] | Hệ thống báo lỗi "vui lòng nhập thông tin!" dưới textbox [nhập lại mật khẩu mới] |
| Trường hợp nhập không trùng với mật khẩu mới | Hệ thống báo lỗi "xác nhận mật khẩu mới và mật khẩu mới không trùng khớp!" dưới textbox [nhập lại mật khẩu mới] |
| Trường hợp nhập trùng với mật khẩu mới | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **Kiểm tra chức năng button [tiếp tục]** | Nhập mật khẩu mới, xác nhận mật khẩu => click [tiếp tục] | Di chuyển đến màn hình [dashboard] |

#### Checklist Xem thông tin chi tiết tài khoản

|  |  |  |
| --- | --- | --- |
| **KIỂM THỬ PHẦN XEM TT CHI TIẾT TK** | | |
|  | **MÔ TẢ TEST CASE** |  |
| **GIAO DIỆN MẶC ĐỊNH** | Kiểm tra hiển thị button [Thông tin tài khoản] |  |
| Kiểm tra mở màn hình [Thông tin tài khoản] |  |
| Kiểm tra hiển thị giao diện màn hình [Thông tin tài khoản] | - Header hiển thị bao gồm các item từ trái sang phải:  + Logo VNPT  + Text "VNPT" - Body hiển thị gồm các item:  + Title "Thông tin tài khoản"  + Lable chứa thông tin user   - [Họ và tên]: Text  - [Giới tính]: Nam/Nữ  - [Ngày sinh]: Date  - [Email]: Text  - [Số điện thoại]: Number  - [Số CCCD/CMND/Hộ chiếu]: Number  - [Ngày cấp]: Date  - [Nơi cấp]: Text   - [Địa chỉ]: Text  - [Chức danh]: Text   - [Đơn vị]: Text  + Button [Chỉnh sửa] |
| Kiểm tra hiển thị dữ liệu của Lable | Hiển thị đúng và đầy đủ dữ liệu (có các trường và dữ liệu hợp lệ):  - [Họ và tên]: Text  - [Giới tính]: Nam/Nữ  - [Ngày sinh]: Date  - [Email]: Text  - [Số điện thoại]: Number  - [Số CCCD/CMND/Hộ chiếu]: Number  - [Ngày cấp]: Date  - [Nơi cấp]: Text   - [Địa chỉ]: Text  - [Chức danh]: Text   - [Đơn vị]: Text |
| **BUTTON [THÔNG TIN TÀI KHOẢN]** | Kiểm tra hiển thị con trỏ chuột | Hiển thị con trỏ hình bàn tay |

#### Checklist Cập nhật thông tin chi tiết tài khoản

|  |  |  |  |
| --- | --- | --- | --- |
| **KIỂM THỬ PHẦN CẬP NHẬT TT CHI TIẾT TK** | | | |
|  |  | **MÔ TẢ TEST CASE** |  |
| **GIAO DIỆN MẶC ĐỊNH** |  | Kiểm tra tính năng button [Chỉnh sửa] |  |
|  | Kiểm tra mở màn hình [Cập nhật thông tin] |  |
|  | Kiểm tra hiển thị giao diện màn hình [Cập nhật thông tin] | Hệ thống hiển thị đầy đủ tất cả các item:  Màn hình [CẬP NHẬT THÔNG TIN CHI TIẾT] hiển thị gồm: - Header hiển thị bao gồm các item từ trái sang phải:  - Logo "VNPT"  - Text "VNPT" - Body hiển thị gồm các item:  - Button [Đăng xuất]  - Button [Lưu]  - Title "Cập nhật thông tin"   - Textbox [Họ và tên]  - Textbox [Ngày sinh]  - Textbox [Email]  - Textbox [Số điện thoại]  - Textbox [Số CCCD/CMND/Hộ chiếu]  - Textbox [Ngày cấp]  - Textbox [Nơi cấp]  - Textbox [Số nhà]  - Textbox [Đường/Thôn/Tổ xóm]  - Textbox [Đơn vị]  - Textbox [Chức danh]  - ComboBox [Tỉnh/Thành phố ( Trực thuộc TW)]  - ComboBox [Xã/Phường/Thị trấn]  - ComboBox [Huyện/Quận/TP (Trực thuộc tỉnh) ]  - Radio Button [Giới tính] |
| **TEXTBOX [HỌ VÀ TÊN]** |  | Kiểm tra hiển thị để trống Textbox [Họ và Tên] | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Kiểm tra hiển thị placehoder | Hiển thị placehoder "Họ và tên" |
|  | Kiểm tra hiển thị con trỏ chuột | Hiển thị hình I |
|  | Nhập chữ, số, kí tự đặc biệt | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
|  | Trường hợp nhập dữ liệu có space 2 đầu | Tự động bỏ ký tự khoảng cách ở 2 đầu |
| **TEXTBOX [NGÀY SINH]** |  | Kiểm tra hiển thị để trống Textbox [Ngày sinh] | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Kiểm tra hiển thị con trỏ chuột | Hiển thị hình I |
|  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehoder "Ngày sinh" |
|  | Kiểm tra hiển thị placehoder đã nhập nhập | Hiển thị ngày sinh |
|  | Nhập chữ+số+kí tự đặc biệt | - Hệ thống chỉ cho nhập ký tự số và ký tự "/" theo định dạng DD/MM/YYYY  - Chặn không cho nhập các ký tự đặc biệt trừ "/" và ký tự chữ |
|  | Nhập ngày 00 | tự động thành 01 |
|  | Nhập 01 đến 31 | Hiển thị đúng |
|  | Nhập >31 | Tự động về 31 |
|  | Nhập tháng 00 | tự động thành 01 |
|  | Nhập tháng 01 đến 12 | Hiển thị đúng |
|  | Nhập >12 | Tự động về 12 |
|  | Nhập 31 ngày cho tháng 4, 6, 9, 11 | Tự động về ngày 30 |
| Nhập 29 cho tháng 2 năm 2021 | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| Nhập 30 cho tháng 2 năm 2021 | Hiển thị ngày 29 cho tháng 2 |
| Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Không thể nhập quá kí tự |
| Nhập ngày tương lai | Hệ thống báo lỗi "Ngày sinh không được là ngày tương lai!" |
| Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Hệ thống báo lỗi "Sai định dạng ngày tháng năm!" |
|  | Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **TEXTBOX [EMAIL]** |  | Kiểm tra hiển thị mặc định | - Hiển thị thông tin email của tài khoản đang login - Ở trạng thái disable, không cho phép chỉnh sửa |
| **TEXTBOX [SỐ ĐIỆN THOẠI]** |  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehoder "Số điện thoại" |
|  | Kiểm tra hiển thị placehoder đã nhập | Hiển thị đúng và đủ |
|  | Kiểm tra hiển thị để trống textbox [Số điện thoại] | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Nhập chữ + số + kí tự đặc biệt | Hệ thống báo lỗi "Sai định dạng số điện thoại" |
|  | Trường hợp nhập dữ liệu có space 2 đầu | Tự động bỏ ký tự khoảng cách ở 2 đầu |
|  | Nhập nhỏ hơn 10 số | Hệ thống báo lỗi "Sai định dạng số điện thoại" |
|  | Nhập lớn hơn 10số | Hệ thống báo lỗi "Sai định dạng số điện thoại" |
| Nhập chữ, kí tự đặc biệt | Hệ thống báo lỗi "Sai định dạng số điện thoại" |
| Nhập đầu số không đúng ( bắt đầu bằng 1) | Hệ thống báo lỗi "Sai định dạng số điện thoại" |
| Trường hợp nhập dấu space giữa số điện thoại | Hệ thống báo lỗi "Sai định dạng số điện thoại" |
|  | Nhập đủ 10 số với đầu số là 0 | Hiển thị đủ và đúng 10số |
| **TEXTBOX [SỐ CCCD/CMND/HỘ CHIẾU]** |  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehoder "Số CCCD/CMND/HỘ CHIẾU" |
|  | Kiểm tra hiển thị placehoder đã nhập | Hiển thị đúng và đủ |
|  | Kiểm tra hiển thị để trống | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Nhập chữ, kí tự đặc biệt | Hệ thống báo lỗi "Sai định dạng số điện thoại" |
|  | Trường hợp nhập dữ liệu có space 2 đầu | Tự động bỏ ký tự khoảng cách ở 2 đầu |
|  | Nhập nhỏ hơn 9 số | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
|  | Nhập lớn hơn 9 số | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
| Nhập nhỏ hơn 12 số | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
| Nhập lớn hơn 12 số | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
| Nhập nhỏ hơn 8 kí tự | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
| Nhập lớn hơn 8 kí tự | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
| Nhập 8 kí tự sai định dạng | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
| Nhập ký tự +số + chữ | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
| Trường hợp nhập dấu space giữa số CCCD/CMND/Hộ chiếu | Hệ thống báo lỗi"Số CCCD/CMND/Hộ chiếu sai định dạng!" |
|  | Nhập đủ 9 số | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
|  | Nhập 8 kí tự đúng định dạng | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
|  | Nhập đủ 12 số | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **TEXTBOX [NGÀY CẤP]** |  | Kiểm tra hiển thị để trống Textbox [Ngày sinh] | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehoder "Ngày sinh" |
|  | Kiểm tra hiển thị placehoder đã nhập nhập | Hiển thị ngày sinh |
|  | Nhập chữ+số+kí tự đặc biệt | - Hệ thống chỉ cho nhập ký tự số và ký tự "/" theo định dạng DD/MM/YYYY  - Chặn không cho nhập các ký tự đặc biệt trừ "/" và ký tự chữ |
|  | Nhập ngày 00 | tự động thành 01 |
|  | Nhập 01 đến 31 | Hiển thị đúng |
|  | Nhập >31 | Tự động về 31 |
|  | Nhập tháng 00 | tự động thành 01 |
|  | Nhập tháng 01 đến 12 | Hiển thị đúng |
|  | Nhập >12 | Tự động về 12 |
|  | Nhập 31 ngày cho tháng 4, 6, 9, 11 | Tự động về ngày 30 |
|  | Nhập 29 cho tháng 2 năm 2021 | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
|  | Nhập 30 cho tháng 2 năm 2021 | Hiển thị ngày 29 cho tháng 2 |
|  | Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Không thể nhập quá kí tự |
|  | Nhập ngày tương lai | Hệ thống báo lỗi "Ngày sinh không được là ngày tương lai!" |
|  | Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Hệ thống báo lỗi "Sai định dạng ngày tháng năm!" |
|  | Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **TEXTBOX [NƠI CẤP]** |  | Kiểm tra hiển thị để trống | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Kiểm tra hiển thị con trỏ chuột | Hiển thị hình I |
|  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehoder |
|  | Kiểm tra hiển thị placehoder đã nhập | Hiển thị đủ và đúng 10số |
|  | Trường hợp nhập dữ liệu có space 2 đầu | Tự động bỏ ký tự khoảng cách ở 2 đầu |
|  | Nhập kí tự số + chữ+ ký tự đặc biệt | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **TEXTBOX [SỐ NHÀ]** |  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehoder "Số nhà" |
|  | Kiểm tra hiển thị placehoder đã nhập | Hiển thị đúng và đủ |
|  | Kiểm tra hiển thị để trống | Không báo lỗi |
|  | Nhập số + chữ + kí tự đặc biệt | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **TEXTBOX [ĐƯỜNG/THÔN/TỔ XÓM]** |  | Kiểm tra hiển thị để trống | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehodler "ĐƯỜNG/THÔN/TỔ XÓM' |
|  | Trường hợp nhập dữ liệu có space 2 đầu | Tự động bỏ ký tự khoảng cách ở 2 đầu |
|  | Nhập số + chữ + kí tự đặc biệt | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **TEXTBOX [ĐƠN VỊ]** |  | Kiểm tra hiển thị mặc định | Email lấy dữ liệu mặc định của tài khoản, ở trạng thái disable không cho chỉnh sửa |
| **TEXTBOX [CHỨC DANH]** |  | Kiểm tra hiển thị placehoder chưa chọn | Hiển thị placehoder "TỈNH/THÀNH PHỐ (TRỰC THUỘC TW)" |
| **COMBOBOX [TỈNH/THÀNH PHỐ (TRỰC THUỘC TW)]** |  | Kiểm tra hiển thị placehoder đã chọn | hiển thị đúng và đủ |
|  | Kiểm tra hiển thị mở đóng khi click vào ComboBox | Hiển thị danh sách tỉnh/thành phố (trực thuộc tw) |
|  | Kiểm tra list dữ liệu | đúng và đủ |
|  | di chuột đến list | Chột đến đâu đổi màu đến đó |
|  | Kiểm tra chọn 1 dữ liệu | Hiển thị dữ liệu đã chọn |
|  | Kiểm tra chọn nhiều dữ liệu | Hiển thị dữ liệu đã chọn gần nhất |
|  | Trường hợp nhập search LIKE tên tỉnh/ thành phố tại searchbox [Tỉnh/Thành phố (Trực thuộc TW) | Hiển thị đúng với kết quả tìm kiếm LIKE |
|  | Trường hợp nhập đúng tên tỉnh/ thành phố tại searchbox [Tỉnh/Thành phố (Trực thuộc TW) | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| Trường hợp nhập tên tỉnh/ thành phố tại searchbox [Tỉnh/Thành phố (Trực thuộc TW) không tồn tại | Kết quả không tồn tại |
| **COMBOBOX [XÃ/PHƯỜNG/THỊ TRẤN]** | Đã nhập Tỉnh/ thành phố | Kiểm tra hiển thị placehoder chưa chọn | Hiển thị placehoder [XÃ/PHƯỜNG/THỊ TRẤN] |
|  | Kiểm tra hiển thị placehoder đã chọn | hiển thị đúng và đủ |
|  | Kiểm tra hiển thị mở đóng khi click vào ComboBox | Hiển thị danh sách [XÃ/PHƯỜNG/THỊ TRẤN] |
|  | Kiểm tra list dữ liệu | đúng và đủ |
|  | di chuột đến list [XÃ/PHƯỜNG/THỊ TRẤN] | Chột đến đâu đổi màu đến đó |
|  | Kiểm tra chọn 1 dữ liệu | Hiển thị dữ liệu đã chọn |
| Kiểm tra chọn nhiều dữ liệu | Hiển thị dữ liệu đã chọn gần nhất |
|  | Trường hợp nhập search LIKE[XÃ/PHƯỜNG/THỊ TRẤN] | Hiển thị đúng với kết quả tìm kiếm LIKE |
|  |  | Trường hợp nhập đúng tên [XÃ/PHƯỜNG/THỊ TRẤN] | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **COMBOBOX [HUYỆN/QUẬN/THÀNH PHỐ (TRỰC THUỘC TỈNH)]** | Đã nhập Tỉnh/ thành phố, Xã/phường | Kiểm tra hiển thị placehoder chưa chọn | Hiển thị placehoder [HUYỆN/QUẬN/THÀNH PHỐ (TRỰC THUỘC TỈNH)] |
|  | Kiểm tra hiển thị placehoder đã chọn | hiển thị đúng và đủ |
|  | Kiểm tra hiển thị mở đóng khi click vào ComboBox | Hiển thị danh sách [HUYỆN/QUẬN/THÀNH PHỐ (TRỰC THUỘC TỈNH)] |
|  | Kiểm tra list dữ liệu | đúng và đủ |
| di chuột đến list [HUYỆN/QUẬN/THÀNH PHỐ (TRỰC THUỘC TỈNH)] | Chột đến đâu đổi màu đến đó |
|  | Kiểm tra chọn 1 dữ liệu | Hiển thị dữ liệu đã chọn |
|  | Kiểm tra chọn nhiều dữ liệu | Hiển thị dữ liệu đã chọn gần nhất |
|  | Trường hợp nhập search LIKE [HUYỆN/QUẬN/THÀNH PHỐ (TRỰC THUỘC TỈNH)] | Hiển thị đúng với kết quả tìm kiếm LIKE |
|  | Trường hợp nhập đúng tên [HUYỆN/QUẬN/THÀNH PHỐ (TRỰC THUỘC TỈNH)] | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **RADIO BUTTON [GIỚI TÍNH]** |  | Kiểm tra hiển thị mặc định | Hiển thị mặc định chọn "Gửi đường dẫn khôi phục mật khẩu về email n\*\*\*@\*\*.vn" |
|  | Trường hợp thay đổi chọn 1 giá trị bất kỳ (Step: click chọn option [Nam]) | Hệ thống hiển thị tick chọn radio [Nam] |
|  | Trường hợp chọn nhiều giá trị cùng 1 lúc (step:  1. click chọn option [Nam] 2. Click chọn option [Nữ]) 3. Kiểm tra hiển thị | - Không thể chọn 2 radio cùng 1 lúc - Hệ thống hiển thị tick chọn tại option [Nữ] |
|  | Trường hợp bỏ chọn radio button (Step: 1. Hệ thống đang hiển thị giá trị tại option 'Nữ' 2.Click chọn radio button tại giá trị 'Nữ' để bỏ check) | - Không thể bỏ chọn radio button - Hệ thống hiển thị dấu tick tại option 'Nữ" |
|  |  | Hiển thị hình bàn tay |
| **TEXTBOX [SỐ GIẤY ỦY QUYỀN]** |  | Kiểm tra hiển thị placehoder | Hiển thị placehoder "Số giấy uỷ quyền" |
|  | Kiểm tra mặc định | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Nhập chữ+số+kí tự đặc biệt | hiển thị đúng và đủ |
| **DATEPICKER [NGÀY CẤP GIẤY ỦY QUYỀN]** |  | Kiểm tra hiển thị để trống Textbox [Ngày cấp giấy ủy quyền] | Hệ thống báo lỗi" Vui lòng nhập thông tin" |
|  | Kiểm tra hiển thị placehoder chưa nhập | Hiển thị placehoder "Ngày cấp giấy ủy quyền" |
|  | Kiểm tra hiển thị placehoder đã nhập | Hiển thị Ngày cấp giấy ủy quyền |
|  | Nhập chữ+số+kí tự đặc biệt | - Hệ thống chỉ cho nhập ký tự số và ký tự "/" theo định dạng DD/MM/YYYY  - Chặn không cho nhập các ký tự đặc biệt trừ "/" và ký tự chữ |
|  | Nhập ngày 00 | tự động thành 01 |
|  | Nhập 01 đến 31 | Hiển thị đúng |
|  | Nhập >31 | Tự động về 31 |
|  | Nhập tháng 00 | tự động thành 01 |
|  | Nhập tháng 01 đến 12 | Hiển thị đúng |
|  | Nhập >12 | Tự động về 12 |
|  | Nhập 31 ngày cho tháng 4, 6, 9, 11 | Tự động về ngày 30 |
|  | Nhập 29 cho tháng 2 năm 2021 | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
|  | Nhập 30 cho tháng 2 năm 2021 | Hiển thị ngày 29 cho tháng 2 |
|  | Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Không thể nhập quá kí tự |
|  | Nhập ngày tương lai | Hệ thống báo lỗi "Ngày cấp giấy ủy quyền không được là ngày tương lai!" |
|  | Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Hệ thống báo lỗi "Sai định dạng ngày tháng năm!" |
|  | Trường hợp nhập đủ 8 ký tự (không tính ký tự "/") | Hiển thị đúng và đầy đủ thông tin đã nhập vào hệ thống |
| **BUTTON [LƯU]** | Kiểm tra hiển thị mặc định | Để trống textbox bấm Đồng ý | Hiển thị cửa sổ thông báo "Vui lòng nhập thông tin!" Quay về màn hình [Cập nhật thông tin] |
| Kiểm tra lưu thành công | Nhập hết thông tin vào textbox bấm nút Hủy | Bước 2: Hiển thị cửa sổ thông báo [Bạn có chắc chắn muốn lưu thông tin?] gồm: - Text "Bạn có chắc chắn muốn lưu thông tin?"  + Button [Hủy]  + Button [Đồng ý] Bước 3: Hiển thị lưu thành công Chuyển hướng đến màng hình Dashboard |
| Nhập thông tin bắt buộc vào textbox bấm nút Lưu |
| Kiểm tra lưu không thành công | Nhập hết thông tin vào textbox bấm nút Hủy | Bước 2: Hiển thị cửa sổ thông báo [Bạn có chắc chắn muốn lưu thông tin?] gồm: - Text "Bạn có chắc chắn muốn lưu thông tin?"  + Button [Hủy]  + Button [Đồng ý] Bước 3: Quay về màn hình Cập nhật |
| Nhập sai 1 thông tin | Bước 2: Hiển thị cửa sổ thông báo [Bạn có chắc chắn muốn lưu thông tin?] gồm: - Text "Bạn có chắc chắn muốn lưu thông tin?"  + Button [Hủy]  + Button [Đồng ý] Bước 3: Thông báo lỗi: "Vui lòng nhập thông tin!" |
| Hiển thị popup "Bạn có chắc muốn Lưu thông tin" | Kiểm tra hiển thị nút Lưu và Hủy | Nút Lưu Hiển thị popup thông báo, bao gồm:  - Text "Bạn có chắc chắn muốn lưu thông tin?"  - Button [Hủy]  - Button [Đồng ý] Nút Hủy - Cập nhật thông tin không thành công - Các trường dữ liệu không đưuọc reset, vẫn dữ nguyên dữ liệu đã nhập trước đó |

### Kết quả viết testcase:

## Kết quả test hệ thống

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN