

**SERVICE DIRECTORY**

**KẾ HOẠCH KIỂM THỬ**

|  |  |
| --- | --- |
| Mã dự án | SD\_SOF303 |
| Mã tài liệu | TP\_V1.0 |
| Ngày | 01/01/2018 |

**Nhóm: Spring**

**Tên thành viên Trần Văn A**

**Nguyễn Văn C**

**Trần Thị D**

**Hà Nội - 01-2018**

BẢN GHI NHẬN THAY ĐỔI TÀI LIỆU

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Ngày thay đổi | Vị trí thay đổi | Lý do | Nguồn gốc | Phiên bản cũ | Mô tả thay đổi | Phiên bản mới |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

TRANG KÝ

**Người lập:** <Ngày>

<Chức vụ>

**Người xem xét:** <Ngày>

<Chức vụ>

<Ngày>

<Chức vụ>

**Người phê duyệt:** <Ngày>

<Chức vụ>

MỤC LỤC

[1 GIỚI THIỆU 4](#_Toc160956423)

[1.1 Mục đích 4](#_Toc160956424)

[1.2 Thông tin chung 4](#_Toc160956425)

[1.3 Tài liệu tham khảo 4](#_Toc160956426)

[1.4 Phạm vi test 4](#_Toc160956427)

[1.5 Ràng buộc 5](#_Toc160956428)

[1.6 Liệt kê các mạo hiểm 5](#_Toc160956429)

[2 CÁC YÊU CẦU CHO Test 6](#_Toc160956430)

[3 CHIẾN LƯỢC TEST 6](#_Toc160956431)

[3.1 Các kiểu test 6](#_Toc160956432)

[3.1.1 Test chức năng (Functional Testing) 9](#_Toc160956433)

[3.1.2 Test hiệu suất (Performance testing) 12](#_Toc160956434)

[3.1.3 Test Bảo mật và Kiểm soát truy cập (Security and Access Control Testing) 16](#_Toc160956435)

[3.1.4 Test hồi qui (Regression Testing) 17](#_Toc160956436)

3.1.5 **Recovery Testing :** kiểm thử phục hồi là kiểm thử được thực hiện sau khi có các sự cố hệ thống dẫn đến chương trình lỗi, không hoạt động được. Kiểm thử phục hồi được thực hiện để đảm bảo chương trình sau khi khắc phục lỗi trên không xảy ra bug

[3.2 Giai đoạn test 18](#_Toc160956437)

[3.3 Các công cụ test 18](#_Toc160956438)

[3.4 Môi trường test 19](#_Toc160956439)

[4 TÀI NGUYÊN 19](#_Toc160956440)

[4.1 Nhân lực 19](#_Toc160956441)

[4.2 Hệ thống 19](#_Toc160956442)

[5 CÁC MỐC KIỂM SOÁT CỦA GIAI ĐOẠN TEST (TEST MILESTONES) 20](#_Toc160956443)

[6 CÁC SẢN PHẨM 20](#_Toc160956444)

# GIỚI THIỆU

## Mục đích

* + - Tài liệu kế hoạch kiểm thử này đưa ra các mục đích sau:
      * Xác định thông tin cơ bản về dự án và các thành phần chức năng được kiểm thử.
      * Liệt kê các danh sách yêu cầu cho test.
      * Những chiến lược kiểm thử được sử dụng trong dự án.
      * Đưa ra các yêu cầu về tài nguyên phù hợp cho việc kiểm thử.

## Thông tin chung

* Hệ thống kho dữ liệu trung tâm Service Directory của công ty AB là một hệ thống có các chức năng dùng để đăng nhập, lưu trữ dữ liệu về tổ chức của hệ thống, lưu trữ về dịch vụ, về chương trình, cơ sở, tài nguyên, con người và dữ liệu địa lý trong hệ thống. Hệ thống sau khi đăng nhập sẽ vào màn hình hiển thị hệ thống với 4 mục chức năng: Organisation, Service, Geograph, Premises và Contact. Trong các mục chức năng có những chức năng liên hệ, thêm liên hệ, tìm kiếm liên lạc và tra cứu Type of Business. Tại mục Contact sẽ hiển thị danh sách, thêm mới hoặc sửa chữa các liên hệ các nhân trong hệ thống. Dữ liệu liên hệ sẽ hiển thị 1 cửa sổ và lấy dữ chi tiết hiển thị từ tổ chức, chi nhánh … khi điền liên hệ của 1 cá nhân.

## Tài liệu tham khảo

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên tài liệu | Nguồn | Ghi chú |
| 1 | Tài liệu SRS\_AB\_TV |  |  |
| 2 |  |  |  |
|  |  |  |  |

## Phạm vi test

Tài liệu này là kế hoạch kiểm thử được áp dụng cho việc kiểm thử những chức năng hệ thống kho trung tâm dữ liệu Service Directory của công ty AB.

Các chức năng cần kiểm tra:

* + - Kiểm tra đăng nhập hệ thống: người dùng cần đăng nhập vào user và password để đăng nhập hệ thống
      * Nếu để trống trường user và password sẽ đưa ra thông báo yêu cầu nhập lại
      * Khi user hoặc password không đúng sẽ đưa ra yêu cầu đăng nhập lại
      * Khi người dùng quên password sẽ vào phần tính năng “ Quên mật khẩu ” cho phép người dùng lấy lại password khi nhập email. Nếu email không khớp sẽ đưa ra thông báo và ngược lại hệ thống sẽ gửi mật khẩu đến email để nhập vào.
      * Kiểm tra đăng nhập với tài khoản vừa đổi mật khẩu.
    - Chức năng liên hệ: người dùng nhập liệu vào First Name và Surname từ đây sẽ kiểm tra có hiển thị danh sách Contact về thông tin những người có First Name và Surname.
    - Chức năng thêm liên hệ: người dùng có thể tạo liên lạc với việc bấm lệnh Create, kiểm tra tính năng Save khi:
      * Nếu First Name và Surname trống đưa ra thống báo và yêu cầu người dùng nhập lại
      * Đúng định dạng Office Phone, Mobile Phone, ST Home Phone phải là số nếu không đưa ra thống báo và yêu cầu người dùng nhập lại.
      * Tính năng Lookup hiển thị các khung tìm kiếm liên lạc để lựa chọn liên lạc.
    - Chức năng chỉnh sửa liên lạc: người dùng có thể chỉnh sửa liện lạc với việc bấm lệnh Edit. Kiểm tra hệ thống có đưa hiển thị đúng người cần sửa.
    - Chức năng tìm kiếm liên lạc: sẽ cung cấp Address, Post code, Town, County, Country từ dữ liệu nhập vào. Kiểm tra các tính năng:
      * Tính năng Search Address hiển thị danh sách tất cả các địa chỉ sau khi người dùng nhập Postcode, Street, Town.
      * Tính năng None sẽ đóng khung màn hình và tất cả các trường địa chỉ.
      * Tính năng Clear xóa toàn bộ điều kiện mà người dùng nhập vào.
      * Tính năng Close màn hình sẽ đóng.

## Ràng buộc

* + - Yêu cầu liên quan đến hiệu năng:
      * Môi trường thử nhiệm cho các công ty nước ngoài của FPT như sau: máy chủ: CPU – Intel Pentium4 3.0GHz, RAM – 4GB, HDD-160GB, OS-Window Server 2003 SP1 và máy khách: CPU – Intel Pentium4 2.4GHz, RAM – 1GB, HDD-40GB, OS-Window XP SP2.
      * Thời gian phát triên phần mềm đảm bảo các yêu cầu sau: Thêm chức năng(giả sử có 30 trường cần được cập nhật vào dữ liệu) Yêu cầu thời gian thực thi khoảng từ 3-5 giây. Lần đầu tiên truy vấn trang, cần nhiền thời gian hơn 1 chút, khoảng 10 giây. Các lần gọi tiếp theo ít hơn 5 giây. Chức năng hiển thị danh sách ( giả sử có khoảng 1000 bản ghi hiển thị) thời gian phản hổi hiển thị là 7-10 giây. Cho lần đầu tiên khoảng 15 giây. Cho các lần tiếp theo ít hơn 10 giây Với tất cả các xác thực về logic dữ liệu, thời gian phản hồi không quá 2 giây.
      * Với máy chủ được đề cập ở trên, hệ thống Service Directory cần đảm bảo làm việc cho cùng 1 lúc 20-50 người dùng.
    - Yêu cầu liên quan đến khả năng mở rộng:
      * Hệ thống này cần có khả năng mở rộng hướng đến số lượng người dùng.
    - Bảo mật:
      * Liên quan đến cơ chế xác thực phân cơ sở và xác thực cho người dùng chuẩn.
    - Tính linh động:
      * Làm việc trên trình duyệt IE 6.0 trở lên và FireFox ở bản phát hành tới
    - Yêu cầu liên quan đến xử lý lỗi:
      * Chủ động thông báo các vấn đề. Hệ thống cần cung cấp đầy đủ các ngữ cánh theo thông báo để hỗ trợ chẩn đoán và sửa chữa các vẫn đề. Xác định mức độ thông báo là cần thiết cho các lớp khác nhau của từng trường hợp lỗi: lỗi đăng nhập các tệp tin, hiển thị và gửi email.
    - Yêu cầu liên quan đến cơ sở hạ tầng:
      * Tất cả các dịch vụ bên trong của Cẩm nang dịch vụ đến hướng tới máy chủ ở giao thức https nhằm mục đích bảo mật. Các ủy quyền cần được cung cấp cho cổng giao thức SSL.
    - Yêu cầu liên quan đến cảm quan và hình thức:
      * Xem mục 6. Giao diện tương tự như màn hình mẫu gợi ý với FRS cho những CID và giống với phần mền MS Dynamics.
    - Yêu cầu liên quan đến tài liệu người dùng:
      * Biểu tượng trợ giúp sẽ nằm ở góc màn hình và có thể chọn khi chạy sang màn hình tiếp theo. Tài liệu trợ giúp được lưu trữ trong mục pháp lý. Nó nằm trong phạm vi của Phần 2.
    - Yêu cầu liên quan đến trợ giúp:
      * Hỗ trợ đầy đủ trong quá trình kiểm thử chấp nhận (UAT) và 3 tháng hỗ trợ sau đó
    - Yêu cầu liên quan đến độ tin cậy:
      * Tính sẵn sàng: Cẩm nang hướng dẫn dịch vụ được định hướng chạy 24h một ngày và 7 ngày trong tuần mà không phát sinh lỗi phản hồi. Cần đảm bảo không có ngoại lệ nào phát sinh bên trong dịch vụ.
    - Yêu cầu liên quan đến thiết kế:
      * Thiết kế cần đảm bảo rằng yêu cầu này tương thích với bất cứ nội dung nào mà hệ thống có thể hỗ trợ.
      * Thiết kế cần tuân theo những ý chính trong thiết kế của hệ thống hiển thị trung tâm CID.

## Liệt kê các mạo hiểm

<Liệt kê các mạo hiểm/rủi ro và phương án khắc phục, phòng ngừa có thể ảnh hưởng đến việc thiết kế, phát triển và thực hiện test. Khi lập tài liệu thì cần xoá dòng hướng dẫn trên đi>

|  |  |  |  |
| --- | --- | --- | --- |
| **Stt** | **Mạo hiểm** | **Phương án khắc phục & phòng ngừa** | **Mức độ ảnh hưởng (MD)** |
| 1 | NA | NA | NA |
| 2 | NA | NA | NA |
|  |  |  |  |

# CÁC YÊU CẦU CHO Test

Danh sách dưới đây xác định các thành phần (tình huống test, các yêu cầu chức năng và phi chức năng) được xác định như mục tiêu test. Các thành phần liệt kê trong danh sách này sẽ được test.

<Liệt kê danh sách các yêu cầu chính cho test>

# CHIẾN LƯỢC TEST

<Chiến lược test giới thiệu phương án tiếp cận để test các mục tiêu test.

Những vấn đề chính trong chiến lược test là các kỹ thuật được áp dụng và điều kiện **để biết khi nào việc test được hoàn thành**.

Mô tả các kiểu test dùng trong dự án.

Có thể liệt kê với mỗi kiểu test tương ứng test cho chức năng nào[[1]](#footnote-1).

Việc test có thể dừng khi nào.

Ví dụ:

* Nó không còn hữu ích
* Nó đòi hỏi một phạm vi nhất định
* Nó đòi hỏi một số lỗi nhất định phải tìm được
* Hết thời gian
* Tester trả lại gói phần mềm cho LTV khi chưa sửa lỗi

>

## Các kiểu test

<Đối với mỗi kiểu test phải giải thích **kỹ thuật, điều kiện hoàn thành và các vấn đề đặc biệt liên quan**.

**Kỹ thuật**: Kỹ thuật phải mô tả việc test được thực hiện như thế nào, bao gồm cả những gì sẽ được test, các hoạt động chính sẽ được thực hiện trong quá trình test và các phương pháp dùng để đánh giá kết quả.

**Điều kiện hoàn thành**: Điều kiện hoàn thành được phát biểu nhằm hai mục đích:

* Xác định chất lượng sản phẩm được chấp nhận
* Xác định thời điểm mà các nỗ lực test được thực hiện thành công

Một điều kiện hoàn thành được phát biểu rõ ràng phải bao gồm:

* Chức năng, hoạt động hoặc các điều kiện được tính toán
* Phương pháp tính toán

Điều kiện hoặc mức độ thích ứng với phép đo

**Các vấn đề đặc biệt**: Phần này phải chỉ ra các ảnh hưởng hoặc phụ thuộc có thể tác động hoặc ảnh hưởng đến nguồn lực test mô tả trong chiến lược. Các ảnh hưởng có thể bao gồm: Nhân công (ví dụ sự sẵn sàng hoặc cần thiết của các nguồn lực khác test để hỗ trợ/tham gia trong test); các ràng buộc (ví dụ hạn chế về thiết bị hoặc sự sẵn sàng hoặc cần thiết/thiếu các thiết bị đặc biệt); các yêu cầu đặc biệt (ví dụ lịch test hoặc truy cập vào hệ thống)

**Một ví dụ về mô tả kiểu test:**

***Kỹ thuật***:

* *Functional Test*

*Đối với chu trình sự kiện của mỗi UC, sẽ xác định một tập các giao dịch đại diện cho mỗi hành động của tác nhân khi thực hiện UC.*

*Tối thiểu phải có 2 TC cho mỗi giao dịch, một TC để phản ánh điều kiện tích cực và một phản ánh điều kiện tiêu cực (không được chấp nhận)*

*Trong giai đoạn đầu tiện, các UC 1-4 và 12 sẽ được test, theo hình thức sau:*

*UC 1 bắt đầu với tác nhân đã truy cập thành công vào ứng dụng và tại cửa sổ chính, và kết thúc khi người dùng xác định SAVE.*

*Mỗi TC sẽ được tiến hành và thực hiện bằng cách sử dụng Rational Robot.*

*Việc kiểm tra và đánh giá việc thực hiện mỗi TC sẽ được thực hiện theo phương pháp sau:*

*Thực hiện Test script (Mỗi test script có được thực hiện thành công như mong muốn không?)*

*Tình trạng Window hoặc phương pháp kiểm tra Object Data (tiến hành trong các test script) sẽ được dùng để kiểm tra sự hiển thị của các màn hình chính và dữ liệu được xác định được nắm bắt/hiển thị bởi mục tiêu test trong khi thực hiện test.*

*Cơ sở dữ liệu của các mục tiêu test (sử dụng Microsoft Access) sẽ được kiểm tra trước khi test và kiểm tra lại sau khi test để kiểm chứng rằng các thay đổi thực hiện trong quá trình test đã được phản ánh chính xác trong dữ liệu.*

* *Performance Test:*

*Với mỗi UC, xác định một tập các giao dịch, như định nghĩa trong tài liệu phân tích workload, sẽ được tiến hành và thực hiện bằng Rational Suite PerformanceStudio và Rational Robot (GUI scripts)*

*Ít nhất 3 workloaf được phản ánh trong test script và lịch trình thực hiện test, bao gồm:*

*Stressed workload: 750 người dùng (15 % quản lý, 50 % bán hàng, 35 % marketing)*

*Peak workload: 350 người dùng (10 % quản lý, 60 % bán hàng, 30 % marketing)*

*Nominal workload: 150 người dùng (2 % quản lý, 75% bán hàng, 23 % marketing)*

*Test script dùng để thực hiện mỗi giao dịch sẽ bao gồm bộ đếm thời gian tương tự để đo thời gian phản hồi, ví dụ tổng thời gian giao dịch (như định nghĩa trong tài liệu phân tích workload), và các hoạt động giao dịch chính hoặc thời gian xử lý.*

*Test script sẽ thực hiện các workload trong 1 giờ (trừ phi được ghi chú khác trong tài liệu phân tích workload).*

*Kiểm tra và đánh giá việc thực hiện mỗi thực hiện test (của một workload) bao gồm:*

*Thực hiện test được theo dõi bằng biểu đồ trạng thái (để xác định rằng việc test và workload được thực hiện như mong muốn)*

*Thực hiện test script (mỗi test script có được thực hiện thành công như mong đợi không?)*

*Ghi nhận và đánh giá thời gian phản hồi đã định nghĩa bằng các báo cáo sau:*

* + *Performance Percentile*
  + *Response Time*

***Điều kiện hoàn thành****:*

Tất cả các TC có trong kế hoạch đều đã được thực hiện

Tất cả các lỗi được xác định phải được ghi nhận vào một giải pháp đã thỏa thuận (All identified defects have been addressed to an agreed upon resolution)

Tất cả các TC có trong kế hoạch đã được thực hiện lại và toàn bộ các lỗi mở đã được ghi nhận như đã thỏa thuận và không có lỗi mới nào được phát hiện

Hoặc

Toàn bộ các TC đặt mức ưu tiên cao đều đã được thực hiện

Toàn bộ các lỗi tìm thấy đều được ghi nhận vào một giải pháp đã thỏa thuận

Toàn bộ các lỗi có trọng số 1 và 2 đều được giải quyết

Tất cả các TC có mức ưu tiên cao đều đã được thực hiện lại và toàn bộ các lỗi mở đã được ghi nhận như đã thỏa thuận và không có lỗi mới nào được phát hiện

**Các vấn đề đặc biệt**

* Cơ sở dữ liệu test yêu cầu người thiết kế hoặc quản trị CSDL hỗ trợ để tạo mới, cập nhật và làm tươi dữ liệu test
* Việc test hiệu suất hệ thống sử dụng máy chủ trong mạng hiện tại (có hỗ trợ cả các giao dịch khác không thuộc việc test). Việc test sẽ phải được lập lịch vào những giờ không còn các giao dịch khác trên mạng.
* Mục tiêu test phải đồng nhất với hệ thống hợp lệ (hoặc giả lập đồng bộ) để việc test chức năng có thể được tiến hành và thực hiện
* Việc test có thể bị dừng khi <số lỗi vượt quá norm, ...>
* Cán bộ test có thể dừng test khi lập trình viên không thực hiện unit test, ...

>

### Test chức năng (Functional Testing)

#### Test chức năng (Function Testing)

<Mục đích của test chức năng là tập trung vào các yêu cầu test có thể được lưu vết trực tiếp trong các UC hoặc các chức năng và qui tắc nghiệp vụ. Mục tiêu của kiểu test này là kiểm tra tính đúng đắn của các dữ liệu, qui trình và báo cáo cũng như việc thực hiện đúng những qui tắc nghiệp vụ. Kiểu test này dựa vào kỹ thuật black box, tức là kiểm tra ứng dụng và các xử lý nội tại bằng cách tương tác với ứng dụng thông qua giao diện người sử dụng và phân tích các kết quả hoặc đầu ra. Bảng sau liệt kê một số gợi ý đối với mỗi ứng dụng:

|  |  |
| --- | --- |
| Mục đích test: | <Đảm bảo mục tiêu test đúng đắn của chức năng, bao gồm định hướng, dữ liệu đầu vào, xử lý và dữ liệu nhận được> |
| Cách thực hiện: | <Thực hiện mỗi UC, chu trình UC hoặc chức năng, sử dụng dữ liệu hợp lệ và không hợp lệ để kiểm tra:  -    Kết quả mong đợi với dữ liệu hợp lệ.  -    Lỗi thích hợp hoặc thông báo hiển thị khi dữ liệu không hợp lệ.  -    Mỗi qui tắc nghiệp vụ đều được áp dụng đúng> |
| Điều kiện hoàn thành: | -   <Toàn bộ kế hoạch test đã được thực hiện.  -     Toàn bộ các lỗi phát hiện ra đã được ghi nhận.> |
| Các vấn đề đặc biệt: | <Xác định hoặc mô tả các vấn đề (nội bộ hoặc bên ngoài) ảnh hưởng đến việc test chức năng> |

>

#### Test giao diện người sử dụng (User Interface Testing)

<Test giao diện người dùng (UI) kiểm tra các tương tác của người dùng với phần mềm. Mục tiêu của test UI là để đảm bảo rằng giao diện người dùng cung cấp cho người sử dụng cách truy cập và sử dụng thích hợp thông qua các chức năng trong mục tiêu test. Ngoài ra, test UI còn để đảm bảo rằng các đối tượng trong phạm vi chức năng UI giống như mong đợi và phù hợp với tổ chức hoặc chuẩn ngành.>

|  |  |
| --- | --- |
| Mục đích test: | <Kiểm tra:     Việc sử dụng thông qua mục tiêu test phản ánh đúng các chức năng và yêu cầu nghiệp vụ, bao gồm màn hình đến màn hình, trường đến trường và sử dụng các phương pháp truy cập (phím tabs, di chuột, tổ hợp phím)      Các đối tượng và thuộc tính màn hình như menus, size, position, state, và tập tring vào việc tương thích với chuẩn> |
| Cách thực hiện: | <Tạo ra và chỉnh sửa test cho mỗi màn hình để kiểm tra việc sử dụng đúng cách và tình trạng các đối tượng cho mỗi màn hình và đối tượng của ứng dụng> |
| Điều kiện hoàn thành: | <Mỗi màn hình được kiểm tra thành công đúng với phiên bản kiểm tra hoặc phạm vi chấp nhận được> |
| Các vấn đề đặc biệt: | <Không phải toàn bộ các thuộc tính của các đối tượng đều truy cập được> |

#### Test dữ liệu và tích hợp dữ liệu (Data and Database Integrity Testing)

<Cơ sở dữ liệu và xử lý cơ sở dữ liệu phải được test như một hệ thống con trong dự án. hệ thống con này phải được test không cần thông qua giao diện người dùng để giao tiếp với dữ liệu. Nghiên cứu thêm về DBMS để xác định các công cụ và kỹ thuật có thể có giúp hỗ trợ cho việc test:

|  |  |
| --- | --- |
| Mục đích test: | <Đảm bảo rằng các phương pháp truy cập và chức năng xử lý là đúng và không có sai lệch dữ liệu> |
| Cách thực hiện: |      <Thực hiện từng phương pháp truy cập và xử lý, thử từng trường hợp với dữ liệu hợp lệ và không hợp lệ hoặc các yêu cầu dữ liệu.      Kiểm tra cơ sở dữ liệu để đảm bảo rằng dữ liệu được lưu trữ như mong đợi, toàn bộ các sự kiện với cơ sở dữ liệu xảy ra đều đúng, hợc xem xét các dữ liệu trả về để đảm bảo rằng đã nhận được dữ liệu đúng cho các lý do đúng> |
| Điều kiện hoàn thành: | <Tất cả các phương pháp truy cập và chức năng xử lý đều giống như thiết kế và không có sai lệch dữ liệu> |
| Các vấn đề đặc biệt: |     <Việc test có thể đòi hỏi phải môi trường phát triển DBMS hoặc drivers để truy cập hoặc sửa dữ liệu trực tiếp trong cơ sở dữ liệu.      Các xử lý phải được thực hiện bằng tay.     Cơ sở dữ liệu có kích thước nhỏ hoặc tối thiểu (giới hạn số bản ghi) phải được dùng để làm rõ thêm các sự kiện không được phép chấp nhận> |

>

#### Test chu trình nghiệp vụ (Business Cycle Testing)

<Test chu trình nghiệp vụ phải thực hiện các hoạt động trong dự án qua thời gian. Phải xác định một chu kỳ, ví dụ một năm, và các giao dịch và hoạt động có thể xảy ra trong chu kỳ của năm đó phải được thực hiện. Việc này bao gồm cả các chu kỳ hàng ngày, hàng tuần hoặc hàng tháng và các sự kiện là ảnh hưởng bởi ngày tháng, ví dụ như ứng dụng ngân hàng>

|  |  |
| --- | --- |
| Mục đích test: | <Đảm bảo mục đích của test là đúng đắn và các tiến trình chạy ngầm thực hiện đúng yêu cầu về mô hình nghiệp vụ và lịch trình> |
| Cách thực hiện: | <Việc test sẽ giả lập vài chu trình nghiệp vụ bằng cách thực hiện các công việc sau:     Các test dùng cho việc test chức năng sẽ được sửa lại hoặc nâng cấp để tăng số lần mỗi chức năng được thực hiện để giả lập một số người dùng khác nhau trong chu kỳ đã định.      Toàn bộ các chức năng theo ngày tháng sẽ được thực hiện với dữ liệu hợp lệ và không hợp lệ hoặc chu kỳ thời gian     Toàn bộ các chức năng xảy ra trong lịch trình chu kỳ sẽ được thực hiện vào thời gian thích hợp     Việc test sẽ bao gồm cả dữ liệu hợp lệ và không hợp lệ để kiểm tra:     - Kết quả xảy ra khi dữ liệu hợp lệ.        - Lỗi tương tự hoặc cảnh báo hiển thị khi dữ liệu không hợp lệ.       Mỗi qui tắc nghiệp vụ đều được áp dụng. |
| Điều kiện hoàn thành: |  <Toàn bộ kế hoạch test đã được thực hiện.   Toàn bộ các lỗi phát hiện ra đều được ghi nhận> |
| Các vấn đề đặc biệt: |    <Ngày và các sự kiện của hệ thống có thể đòi hỏi các hoạt động hỗ trợ đặc biệt     Mô hình nghiệp vụ đòi hỏi xác định các yêu cầu và thủ tục test thích hợp> |

### Test hiệu suất (Performance testing)

#### Performance Profiling

<Performance profiling là một dạng test hiệu suất trong đó thời gian phản hồi, tỷ lệ giao dịch và các yêu cầu phụ thuộc thời gian khác được đo đạc và đánh giá. Mục đích của Performance Profiling là kiểm tra các yêu cầu về hiệu suất có đạt được hay không. Performance profiling là tiến hành và thực hiện để mô tả sơ lược và điều chỉnh các hành vi hiệu suất của mục tiêu test như một hàm của các điều kiện ví dụ workload hoặc cấu hình phần cứng.

Chú ý: Các giao dịch dưới đây tham chiếu đến “các giao dịch nghiệp vụ logic”. Các giao dịch này được định nghĩa như xác định các UC mà tác nhân của hệ thống hy vọng được thực hiện bằng cách sử dụng mục tiêu test, như thêm mới hoặc sửa một hợp đồng>

|  |  |
| --- | --- |
| Mục đích test: | <Kiểm tra các biểu hiện về hiệu suất cho các giao dịch hoặc chức năng nghiệp vụ đã thiết kế theo những điều kiện sau:      workload bình thường đã biết trước (normal anticipated workload)      workload xấu đã biết trước (anticipated worst case workload)> |
| Cách thực hiện: |     <Sử dụng các thủ tục test cho test chức năng và chu trình nghiệp vụ      Chỉnh sửa file dữ liệu để tăng số lượng các giao dịch hoặc scripts để tăng số tương tác xảy ra trong mỗi giao dịch      Scripts phải được chạy trên một máy (trường hợp tốt nhất để đánh giá người dùng đơn lẻ, giao dịch đơn lẻ) và phải lặp lại trên nhiều máy trạm (ảo hoặc thực, xem các vấn đề đặc biệt dưới đây)> |
| Điều kiện hoàn thành: |     <Giao dịch đơn lẻ hoặc người dùng đơn lẻ: Thực hiện thành công test script không có lỗi và trong phạm vi mong đợi hoặc thời gian phản hồi cho mỗi giao dịch>      <Nhiều giao dịch hoặc nhiều người dùng: Thực hiện thành công test script không có lỗi và trong thời gian chấp nhận được> |
| Các vấn đề đặc biệt: | <Việc test hiệu suất toàn diện bao gồm phải có một workload nền trên máy chủ.  Có một số phương pháp để thực hiện, bao gồm:     “Drive transactions” trực tiếp đến máy chủ, thường trong các form gọi SQL.      Tạo các người dùng ảo để giả lập nhiều máy trạm, thường là vài trăm. Sử dụng công cụ Remote Terminal Emulation để thực hiện việc load này, kỹ thuật này còn được dùng để load giao dịch trên mạng      Sử dụng nhiều người dùng, mỗi người chạy một test script để load lên hệ thống  Test hiệu suất phải được thực hiện trên máy chuyên dụng hoặc thời gian chuyên dùng. Điều đó cho phép việc tính toán được đầy đủ và chính xác.  Cơ sở dữ liệu sử dụng để test hiệu suất phải có kích thước thực tế hoặc đo bằng nhau> |

#### Load Testing

<Load testing là một kiểu test hiệu suất mà mục tiêu là kiểm tra workload để tính toán và đánh giá hiệu suất và khả năng của mục đích test để tiếp tục thực hiện các chức năng thích hợp với các workload khác. Mục đích của load testing là xác định và đảm bảo các chức năng hệ thống thích hợp với nhiều nhất các workload. Ngoài ra load testing còn đánh giá các tính năng hiệu suất như thời gian phản hồi, tỉ lệ giao dịch và các vấn đề liên quan đến thời gian khác.>

<Chú ý: Các giao dịch dưới đây tham chiếu đến “các giao dịch nghiệp vụ logic”. Các giao dịch này được định nghĩa như các chức năng xác định mà người dùng cuối của hệ thống mong muốn thực hiện thông qua ứng dụng như thêm hoặc sửa các thông tin hợp đồng>

|  |  |
| --- | --- |
| Mục tiêu test: | <Kiểm tra hiệu suất về thời gian cho các giao dịch hoặc tình huống nghiệp vụ đã thiết kế với nhiều điều kiện workload> |
| Cách thực hiện: |     <Sử dụng các test đã xây dựng cho test chức năng và chu trình nghiệp vụ.      Sửa lại file dữ liệu để tăng số lượng giao dịch hoặc test nhằm tăng thêm số lần thực hiện mỗi giao dịch> |
| Điều kiện hoàn thành: | <Nhiều giao dịch hoặc nhiều người dùng: Thực hiện thành công việc test không có lỗi và trong thời gian chấp nhận được> |
| Các vấn đề đặc biệt: |     <Load testing phải được thực hiện trên máy chuyên dụng hoặc vào những giờ chuyên biệt. Nó cho phép đo đạc đầy đủ và chính xác.      Cơ sở dữ liệu dùng cho load testing phải có kích thước thực tế hoặc đo bằng nhau> |

#### Stress Testing

<Stress testing là một kiểu test hiệu suất được thực hiện để tìm ra các lỗi trong trường hợp thiếu tài nguyên hoặc cạnh tranh về tài nguyên. Bộ nhớ hoặc dung lượng đĩa ít có thể làm xuất hiện lỗi trong mục đích test mà nó không xuất hiện dưới điều kiện bình thường. Các lỗi khác có thể là kết quả của việc cạnh tranh hoặc chia sẻ tài nguyên như khóa cơ sở dữ liệu hoặc băng thông mạng. Stress testing cũng được dùng để xác định wordload tối đa mà mục đích test có thể điều khiển được.>

<Chú ý: Tham khảo các giao dịch dưới đây tham chiếu đến các giao dịch nghiệp vụ logic>

|  |  |
| --- | --- |
| Mục đích test: | <Kiểm tra các chức năng của mục đích test là đúng đắn và không có lỗi với những điều kiện sau:      Có ít hoặc không có bộ nhớ phù hợp trên máy chủ (RAM và DASD)      Số lượng máy trạm tối đa trong thực tế hoặc giả lập kết nối vào máy chủ      Nhiều người dùng thực hiện cùng một giao dịch với cùng dữ liệu hoặc account      Độ lớn các giao dịch xấu hoặc hỗn hợp (xem phần Performance Testing ở trên).  Chú ý: Mục đích của Stress Testing có thể được phát biểu rõ và ghi ra các điều kiện mà hệ thống có thể lỗi, không thể tiếp tục thực hiện các chức năng một cách thích hợp> |
| Cách thực hiện: |     <Sử dụng các test đã xây dựng để thực hiện Performance Profiling hoặc Load Testing.      Để test việc hạn chế tài nguyên, test phải chạy trên máy đơn lẻ và RAM và DASD trên máy chủ phải giảm đi hoặc hạn chế      Để thực hiện các stress tests khác phải sử dụng nhiều người dùng cùng chạy một TC hoặc bổ sung các test để thực hiện độ lớn giao dịch xấy hoặc hỗn hợp. |
| Điều kiện hoàn thành: | <Toàn bộ kế hoạch test được thực hiện và các hạn chế của hệ thống được xác định thỏa mãn các điều kiện tối thiểu đã đặt ra hoặc chỉ sai trong trong hợp các điều kiện không nằm trong điều kiện đã xác định> |
| Các vấn đề đặc biệt: |     <Việc test Stressing mạng có thể đòi hỏi những công cụ để load mạng với nhiều thông báo hoặc gói dữ liệu.      DASD dùng cho hệ thống phải tạm thời giảm xuống để hạn chế khả năng chỗ trống cho tăng trưởng cơ sở dữ liệu.      Đồng bộ hóa các máy trạm đồng thời truy cập vào cùng một bản ghi hoặc các account dữ liệu> |

#### Volume Testing

<Mục tiêu của Volume Testing là để kiểm tra giới hạn của độ lớn của dữ liệu có thể làm phần mềm bị sai. Volume Testing cũng xác định load lớn nhất liên tục hoặc độ lớn mà mục đích test có thể điều khiển được trong chu kỳ đã cho. Ví dụ, nếu mục đích test là xử lý một tập các bản ghi để tạo báo cáo, Volume Test có thể dùng một cơ sở dữ liệu test lớn và kiểm tra xem phần mềm có chạy bình thường và cho ra báo cáo đúng không>

|  |  |
| --- | --- |
| Mục đích test: | <Kiểm tra xem mục tiêu test có thực hiện thành công các chức năng theo những điều kiện sau không:      Số máy trạm lớn nhất kết nối (thực tế hoặc vật lý – có thể), hoặc giả lập, tất cả đều thực hiện cùng một chức năng nghiệp vụ trong một chu kỳ mở rộng.      Kích thước cơ sở dữ liệu lớn nhất có thể (thực tế hoặc đo được) và nhiều query hoặc giao dịch báo cáo được thực hiện đồng thời.> |
| Cách thực hiện: |     <Sử dụng các test đã xây dựng cho Performance Profiling hoặc Load Testing.      Có thể dùng nhiều người dùng, chạy cùng một test hoặc bổ sung các test để thực hiện trường hợp giao dịch volume hoặc hỗn hợp xấu nhất (xem Stress Testing ở trên) trong một chu kỳ mở rộng.      Tạo ra cơ sở dữ liệu lớn nhất (thực tế, qui đổi, hoặc lọc các dữ liệu đại diện) và nhiều người dùng chạy các query và giao dịch báo cáo đồng thời trong một chu kỳ mở rộng> |
| Điều kiện hoàn thành: |     <Toàn bộ kế hoạch test được thực hiện và các giới hạn của hệ thống được xác định là đạt tới hoặc xử lý mà không có lỗi> |
| Các vấn đề đặc biệt: | <Chu kỳ thời gian như thế nào là chấp nhận được cho điều kiện cơ sở dữ liệu lớn, như đã nói ở trê?> |

### Test Bảo mật và Kiểm soát truy cập (Security and Access Control Testing)

<Test bảo mật và kiểm soát truy cập tập trung vào hai lĩnh vực bảo mật chính:

* Bảo mật ở mức ứng dụng, bao gồm truy cập dữ liệu và các chức năng nghiệp vụ
* Bảo mật ở mức hệ thống, bao gồm truy cập vào hệ thống hoặc truy cập từ xa

Bảo mật mức ứng dụng đảm bảo rằng, dựa trên bảo mật đã yêu cầu, người dùng bị hạn chế sử dụng một số chức năng hoặc tình huống sử dụng, hoặc bị hạn chế trong giới hạn dữ liệu phù hợp với họ. Ví dụ, mọi người có thể được phép nhập dữ liệu để tạo account nhưng chỉ có người quản lý có thể xóa chúng. Nếu là bảo mật ở mức dữ liệu, việc test đảm bảo rằng “người dùng nhóm 1” có thể nhìn thấy các thông tin khách hàng, bao gồm dữ liệu tài chính, tuy nhiên “người dùng nhóm 2” chỉ nhìn thấy các thông tin chung chung cho cùng một khách hàng.

Bảo mật mức hệ thống đảm bảo rằng chỉ những người dùng được cho quyền truy cập vào hệ thống mới có khả năng truy cập vào ứng dụng và chỉ bằng các cổng thích hợp

>

|  |  |
| --- | --- |
| Mục đích test: |          Bảo mật mức ứng dụng: Đảm bảo rằng một người dùng chỉ có thể truy cập vào những chức năng hoặc dữ liệu mà nhóm người dùng đó được phép           Bảo mật mức hệ thống: Đảm bảo rằng chỉ những người được phép truy cập hệ thống và ứng dụng được phép truy cập chúng |
| Cách thực hiện: |          Bảo mật ứng dụng: Xác định và liệt kê từng nhóm người dùng và các chức năng hoặc dữ liệu mà họ được phép truy cập      Tạo test case cho mỗi nhóm người dùng và kiểm tra từng quyền bằng cách tạo các giao dịch xác định cho mỗi nhóm      Sửa lại nhóm người dùng và chạy lại tình huống test cho cùng những người dùng. Với mỗi trường hợp, kiểm tra các chức năng thêm vào hoặc dữ liệu có đúng không hay bị từ chối.           Truy cập mức hệ thống: tham khảo các điều kiện đặc biệt dưới đây |
| Điều kiện hoàn thành: | <Với mỗi nhóm người dùng đều có các chức năng hoặc dữ liệu thích hợp, và toàn bộ các chức năng giao dịch đều như dự kiến và chạy trong các test chức năng ứng dụng trước đó> |
| Các vấn đề đặc biệt: | <Truy cập vào hệ thống phải được xem xét hoặc thảo luận với quản trị hệ thống hoặc quản trị mạng, có thể không cần nếu nó là chức năng của quản trị mạng hoặc quản trị hệ thống> |

### Test hồi qui (Regression Testing)

<Test hồi qui là một hoạt động cấn thiết để chỉ ra rằng việc thay đổi code không gây ra những ảnh hưởng bất lợi>

|  |  |
| --- | --- |
| Mục đích test: | Test hồi qui dùng để kiểm tra các phần được sửa chữa trong phần mềm, để đảm bảo rằng những sự thay đổi đó không gây ra lỗi trong những phần khác |
| Cách thực hiện: |     <Tái sử dụng các TC từ những phần test trước để test các module đã được sửa chữa>.     <Sử dụng công cụ Rational Robot: Tạo một số test script về chức năng. Định nghĩa lịch thực hiện tự động cho chúng**>**     <80% các TC được chọn ngẫu nhiên>   <Xây dựng một chương trình phân tích sơ sở hạ tầng. Chúng ta dựng một cơ sở hạ tầng có thể mở rộng được để thực hiện và đánh giá chương trình phân tích. Dựa vào kết quả phân tích chúng ta xác định phạm vi cần test hồi qui.> |
| Điều kiện hoàn thành: |     <Toàn bộ các TC được thực hiện và đạt yêu cầu>      <Toàn bộ các TC được chọn được thực hiện và đạt yêu cầu> |
| Các vấn đề đặc biệt: |  |

## Giai đoạn test

<Làm rõ trạng thái của giai đoạn sẽ thực hiện test. Bảng sau liệt kê các giai đoạn mà việc test thường được thực hiện>

| Kiểu test | Giai đoạn test | | | |
| --- | --- | --- | --- | --- |
| Unit | Integration | System | Acceptance |
| <Functional Tests  (Function, User Interface)> | X | X | X | X |
| <Performance Tests  (Performance profiles of individual components)> | X | X |  |  |
| <Performance Tests  (Load, Stress, Contention)> |  |  | X | X |
| <Reliability  (Integrity, Structure)> | X | X |  |  |

## Các công cụ test

<Liệt kê các công cụ sẽ áp dụng cho dự án>

|  |  |  |  |
| --- | --- | --- | --- |
| Mục đích | Công cụ | Nhà cung cấp/Tự xây dựng | Phiên bản |
|  |  |  |  |

## Môi trường test

<Chỉ rõ môi trường sẽ xây dựng để thực hiện test qua các giai đoạn Unit test, Intergration test, System test, Acceptance test…. Với mỗi giai đoạn, hãy xác định các yếu tố để xây dựng môi trường test như thế nào, sử dụng như môi trường mà chương trình sẽ chạy thật hay tạo môi trường giả lập gần giống với môi trường chạy thật của chương trình. Các yếu tố về môi trường như:

- Khi test chạy chương trình bằng bản dịch hay chạy trên code. Thông thường, các giai đoạn System test, Acceptance test phải chạy trên bản dịch.

- Các database sẽ sử dụng độc lập hay dùng chung với database phát triển. Thông thường, từ Intergration test, nhóm test phải thiết lập database riêng và thiết lập các thông số cho database gần giống hoặc giống hệt như khi chương trình sẽ chạy thật.

- Điều kiện về mạng: sẽ sử dụng mạng LAN hay Dial up… Thông thường, khi Unit test, có thể sử dụng mạng LAN nhưng khi System test trở đi thì nên sử dụng hệ thống đường truyền giống như hoặc gần giống như môi trường chạy thật.

- Mô hình sẽ cài đặt chương trình test: số lượng máy chủ, máy trạm; việc chia tách các server, các máy trạm, việc cài đặt các domain … Thông thường, trong Unit test có thể sử dụng viếc thiết lập như khi lập trình, nhưng khi System test trở đi, phải chú ý thiết lập sao cho gần giống mô hình sẽ chạy trong thực tế nhất >.

# TÀI NGUYÊN

## Nhân lực

Bảng sau mô tả nguồn lực test cho dự án.

|  |  |
| --- | --- |
| Họ tên | Trách nhiệm/Ghi chú |
|  |  |

## Hệ thống

<Liệt kê các yêu cầu về phần cứng và phần mềm>

# CÁC MỐC KIỂM SOÁT CỦA GIAI ĐOẠN TEST (TEST MILESTONES)

Test v1.0 phải phối hợp các hoạt động test cho nguồn lực test được xác định trong phần trước. Độc lập với milestone của dự án, phải xác định để thông tin về tình trạng hoàn thành của dự án

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone Task | Nguồn lực | Ngày bắt đầu | Ngày kết thúc |
|  |  |  |  |

# CÁC SẢN PHẨM

| STT | Sản phẩm | Ngày bàn giao | Người bàn giao | Người nhận bàn giao |
| --- | --- | --- | --- | --- |
|  | <Test cases> |  |  |  |
|  | <Test procedures> |  |  |  |
|  | <Defect log> |  |  |  |
|  | <Defect reports> |  |  |  |

1. Chỉ dành cho tester FIS-HCM khi lập tài liệu kế hoạch test [↑](#footnote-ref-1)