**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

-----o0o----

Ảnh có chứa văn bản, thiết bị

Mô tả được tạo tự động

CTT503 – KIỂM CHỨNG PHẦN MỀM

DANH: Kiểm thử ứng dụng WEB

Mã dự án: CTT503

Nhóm: Máy quét Oracel





**Hồ Chí Minh – 22/12/2023**

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 |
| 22/12/2023 |  |  |  |  | Tạo tài liệu | 1.0 |
| 32/12/2023 |  | Thêm nội dung |  | 1.0 | Thêm nội dung | 1.1 |
| 08/01/2024 |  | Thêm nội dung |  | 1.1 | Thêm nội dung | 1.2 |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

TRANG KÝ

**Người lập:** 20120468 – Nguyễn Văn Hải 22/12/2023

Trưởng nhóm

**Người xem xét:** 18120419 – Phạm Trường Khoa 22/12/2023

Thành viên

18120410 – Phạm Quốc Khánh 22/12/2023

Thành viên

**Người phê duyệt:** 20120159 – Nguyễn Sỹ Phong 22/12/2023

Thành viên

Mục lục

[1 GIỚI THIỆU 5](#_Toc154226160)

[1.1 Mục đích 5](#_Toc154226161)

[1.2 Thông tin chung 5](#_Toc154226162)

[1.3 Tài liệu liên quan 6](#_Toc154226163)

[1.4 Phạm vi test 7](#_Toc154226164)

[1.5 Ràng buộc 8](#_Toc154226165)

[1.6 Liệt kê các mạo hiểm 9](#_Toc154226166)

[2 CÁC YÊU CẦU CHO Test 10](#_Toc154226167)

[3 CHIẾN LƯỢC TEST 11](#_Toc154226168)

[3.1 Các kiểu test 11](#_Toc154226169)

[3.1.1 Test chức năng (Functional Testing) 14](#_Toc154226170)

[3.1.2 Test hiệu suất (Performance testing) 17](#_Toc154226171)

[3.1.3 Test Bảo mật và Kiểm soát truy cập (Security and Access Control Testing) 21](#_Toc154226172)

[3.1.4 Test hồi qui (Regression Testing) 21](#_Toc154226173)

[3.2 Giai đoạn test 22](#_Toc154226174)

[3.3 Các công cụ test 23](#_Toc154226175)

[3.4 Môi trường test 23](#_Toc154226176)

[4 TÀI NGUYÊN 24](#_Toc154226177)

[4.1 Nhân lực 24](#_Toc154226178)

[4.2 Hệ thống 25](#_Toc154226179)

[5 CÁC MỐC KIỂM SOÁT CỦA GIAI ĐOẠN TEST (TEST MILESTONES) 26](#_Toc154226180)

[6 CÁC SẢN PHẨM 27](#_Toc154226181)

# GIỚI THIỆU

## Mục đích

Mục đích của đồ án là thực hiểm thử trên 1 trang web, ứng dụng đã được đi vào sử dụng chính thức bằng các phương pháp: Functional testing, GUI and Visibility testing, Automation testing và Performance testing. Từ đó học hỏi về quy trình để thực hiện kiểm thử ở các dự án thực tết, cách tổ chức và ghi chép lại các tài liệu.

Các phần của tài liệu:

* Giới thiệu: giới thiệu chung về tài liệu, dự án.
* Các yêu cầu cho test: danh sách liệu kê các thành phần, yêu cầu cho việc test.
* Chiến lược test: liệu kê ra các kiểu test, các giai đoạn, công cụ và môi trường.
* Tài nguyên: Nhân lực và và hệ thống của nhóm.
* Các mốc kiểm soát giai đoạn test: Liệt kê ra các mốc trong test.
* Các sản phẩm: Các test được hoàn thành và chuyển giao.

## Thông tin chung

Dự án này nhằm kiểm chứng, thử nghiệm các chứng năng chính của nền tảng quản lí tài chính cá nhận firefly III.

Firefly III là một ứng dụng quản lý tài chính cá nhân mã nguồn mở, được thiết kế để giúp người dùng quản lý và theo dõi tài chính cá nhân của họ. Dưới đây là một tóm tắt về chức năng, tính năng, kiến trúc và lịch sử của dự án Firefly III:

Chức năng và Tính năng Chính:

* Quản lý Tài khoản và Giao dịch: Cho phép người dùng theo dõi tất cả các giao dịch tài chính từ nhiều nguồn khác nhau, bao gồm tài khoản ngân hàng, ví điện tử, thẻ tín dụng và tiền mặt.
* Tự động hóa Giao dịch: Cung cấp khả năng tự động hoá việc nhập dữ liệu giao dịch thông qua các cổng kết nối API với các ngân hàng và dịch vụ tài chính khác.
* Quản lý Ngân sách: Cho phép người dùng thiết lập và theo dõi ngân sách cá nhân, theo dõi các khoản chi tiêu và thu nhập.
* Báo cáo và Thống kê: Tạo ra báo cáo chi tiết, biểu đồ và thống kê về tình hình tài chính cá nhân để người dùng có thể hiểu rõ hơn về lối sống tài chính của mình.
* Tính năng Lập kế hoạch Tài chính: Hỗ trợ người dùng lập kế hoạch tài chính dựa trên các mục tiêu cụ thể như tiết kiệm, trả nợ, đầu tư.

**Kiến Trúc:** Firefly III được xây dựng dựa trên nền tảng PHP, sử dụng các công nghệ và framework như Laravel và MySQL để lưu trữ dữ liệu. Giao diện người dùng được xây dựng với HTML, CSS và JavaScript.

**Lịch Sử Dự Án:**

* Firefly III được phát triển bắt đầu từ năm 2016, ban đầu được tạo ra như một dự án cá nhân với mục tiêu giúp tác giả quản lý tài chính cá nhân.
* Do nhu cầu từ cộng đồng người dùng, Firefly III đã phát triển mở rộng và trở thành một dự án mã nguồn mở, được cải tiến liên tục thông qua cộng đồng các nhà phát triển.
* Dự án liên tục cập nhật các phiên bản mới với các tính năng mới, sửa lỗi và cải thiện hiệu suất để cung cấp trải nghiệm người dùng tốt nhất.
* Firefly III đã thu hút sự quan tâm của người dùng muốn có một cách tự quản lý tài chính linh hoạt, đồng thời cung cấp khả năng tùy chỉnh cao để đáp ứng nhu cầu cụ thể của từng người dùng.

## Tài liệu liên quan

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên tài liệu | Nguồn | Ghi chú |
| 1 | Bao cao | 20120468 - Nguyễn Văn Hải | Đánh giá quá trình làm việc của team. |
| 2 |  |  |  |
|  |  |  |  |

## Phạm vi test

**Giai đoạn Test:**

* Unit Testing: Kiểm thử từng đơn vị nhỏ nhất của mã nguồn để đảm bảo chúng hoạt động đúng và không có lỗi.
* Integration Testing: Kiểm thử tích hợp để kiểm tra sự tương tác giữa các thành phần khi chúng được kết hợp lại với nhau.
* System Testing: Kiểm thử hệ thống để đảm bảo rằng hệ thống hoạt động đúng và đáp ứng các yêu cầu của người dùng.

**Kiểu Test trong Kế Hoạch:**

* Functional Testing: Kiểm thử chức năng để đảm bảo rằng mọi chức năng hoạt động đúng theo yêu cầu.
* Performance Testing: Kiểm thử hiệu suất để đánh giá khả năng hoạt động và hiệu suất của hệ thống dưới các điều kiện khác nhau.
* GUI and Usability Testing: Kiểm thử giao diện người dùng và tính khả dụng để đảm bảo trải nghiệm người dùng tốt nhất.
* Automation Testing: Sử dụng các công cụ và kịch bản tự động hóa để thực hiện các bài kiểm tra một cách nhanh chóng và hiệu quả.

**Tính năng và Chức năng sẽ được hoặc không được test:**

* Được test: Quản lý tài khoản và giao dịch, tự động hóa giao dịch, quản lý ngân sách, báo cáo và thống kê.
* Không được test: Tính năng không liên quan đến quản lý tài chính cá nhân (ví dụ: tính năng xã hội, tính năng giải trí).

**Độ Ưu Tiên cho Chức Năng Được Test:** Quản lý tài khoản và giao dịch có thể được ưu tiên cao vì đây là trọng tâm của ứng dụng quản lý tài chính.

**Giả Thiết Trong Quá Trình Lập Kế Hoạch:**

* Giả thiết về sự chuẩn bị và sẵn sàng của môi trường test, cài đặt và tài nguyên cần thiết.
* Giả thiết về sự hỗ trợ và sự tham gia tích cực từ các bên liên quan và cộng đồng người dùng.

**Điều Kiện để Test Hồi Quy:**

* Xác định lại các chức năng quan trọng và quan trọng nhất để test hồi quy.
* Xác định chu kỳ và phạm vi của các bản nâng cấp để tập trung test hồi quy.

Số Lỗi Dự Kiến: Ước tính có thể là khoảng 20-30% của tổng số lượng lỗi đã phát hiện.

## Ràng buộc

**Ràng Buộc liên quan đến Môi Trường Test:**

* Tài Nguyên Hệ Thống Ngoài: Cần thiết lập một môi trường test giống với môi trường sản xuất để kiểm tra tính tương thích và tính ổn định của hệ thống. Nếu có các hệ thống bên ngoài mà hệ thống cần giao tiếp, việc thiếu các tài nguyên này có thể ảnh hưởng đến quá trình kiểm thử.
* Dữ Liệu Test: Cần có dữ liệu test đủ và đa dạng để kiểm tra các trường hợp sử dụng khác nhau của hệ thống. Việc thiếu hụt dữ liệu test có thể làm giảm hiệu suất và độ chính xác của quá trình kiểm thử.

**Ràng Buộc liên quan đến Tài Nguyên và Lịch Trình:**

* Nguồn Lực: Ràng buộc về nguồn lực bao gồm nhân lực, phần cứng, và phần mềm cần thiết cho quá trình kiểm thử. Nếu thiếu hụt nhân lực hoặc công cụ test, có thể gây trì hoãn hoặc hạn chế khả năng thực hiện các loại kiểm thử cần thiết.
* Lịch Trình: Ràng buộc về lịch trình có thể là một yếu tố quan trọng, đặc biệt là trong việc hoàn thành các giai đoạn kiểm thử trong khoảng thời gian cụ thể. Thiếu hụt thời gian có thể ảnh hưởng đến chất lượng của quá trình kiểm thử hoặc khiến cho các bước test bị bỏ qua.

**Ràng Buộc từ Tài Liệu Yêu Cầu Hệ Thống (SRS):**

* Yêu Cầu Giao Tiếp: Nếu có ràng buộc từ tài liệu Yêu Cầu Hệ Thống (SRS) về việc hệ thống cần phải tương tác với các hệ thống khác, việc thiếu các tài nguyên hoặc môi trường để kiểm tra sự tương tác này có thể làm giảm tính toàn vẹn của quá trình kiểm thử.
* Yêu Cầu Cụ Thể: SRS có thể mô tả các yêu cầu cụ thể về môi trường test, dữ liệu test, hoặc các yêu cầu khác có thể tạo ra ràng buộc trong quá trình kiểm thử.
* Yêu Cầu Hiệu Suất: Nếu SRS đề cập đến yêu cầu về hiệu suất hoặc khả năng mở rộng của hệ thống, việc thiếu các công cụ test hoặc môi trường thử nghiệm có thể làm giảm hiệu suất của quá trình kiểm thử này.

## Liệt kê các mạo hiểm

|  |  |  |  |
| --- | --- | --- | --- |
| **Stt** | **Mạo hiểm** | **Phương án khắc phục & phòng ngừa** | **Mức độ ảnh hưởng (MD)** |
| 1 | Thiếu Dữ Liệu Test đa dạng và đủ lượng để phủ các trường hợp sử dụng đa dạng | Tạo dữ liệu test đa dạng từ các nguồn khác nhau hoặc sử dụng công cụ tự động tạo dữ liệu test. | Trung bình - Cao |
| 2 | Thiếu Tài Nguyên Tư Cách và Quyền Truy Cập | Xác định rõ quyền truy cập và cung cấp tài nguyên tư cách để thực hiện các kiểm thử cần thiết. | Trung bình |
| 3 | Rủi ro về Sự Tương Thích | Kiểm tra tương thích trên các trình duyệt, thiết bị và phiên bản phần mềm khác nhau. | Trung bình - Cao |
| 4 | Thiếu Hỗ Trợ từ Các Bên Liên Quan | Tạo liên kết mạnh mẽ và thảo luận rõ ràng với các bên liên quan, yêu cầu hỗ trợ và sự tham gia tích cực. | Trung bình |
| 5 | Thiếu Công Cụ và Tài Nguyên Test | Đầu tư vào các công cụ test và tài nguyên phù hợp, cung cấp đào tạo cho nhân viên về việc sử dụng công cụ test. | Trung bình |
| 6 | Thiếu Thời Gian Cho Giai Đoạn Kiểm Thử | Xác định rõ lịch trình, ưu tiên hoạt động và phân chia thời gian cho các giai đoạn kiểm thử. | Trung bình - Cao |

# CÁC YÊU CẦU CHO Test

**Yêu Cầu Chức Năng:**

*Quản Lý Tài Khoản và Giao Dịch:*

* Tạo, sửa đổi và xóa các tài khoản và giao dịch.
* Phân loại giao dịch và thêm các nhãn để phân biệt.
* Tính năng sao lưu và khôi phục dữ liệu tài khoản.

*Tự Động Hóa Giao Dịch:*

* Kết nối với các ngân hàng hoặc dịch vụ tài chính để tự động tải dữ liệu giao dịch.
* Đảm bảo tính chính xác và an toàn trong việc tự động hóa này.

*Quản Lý Ngân Sách:*

* Thiết lập và theo dõi ngân sách cá nhân, đặt cảnh báo khi vượt quá ngân sách.
* Tính năng phân tích chi tiêu so với ngân sách đặt ra.

*Báo Cáo và Thống Kê:*

* Tạo và xuất các báo cáo về tài chính cá nhân.
* Biểu đồ và thống kê cụ thể về lịch sử giao dịch và xu hướng tài chính.

**Yêu Cầu Phi Chức Năng:**

*Bảo Mật và Quản Lý Truy Cập:*

* Xác thực an toàn và quản lý quyền truy cập dữ liệu cá nhân.
* Tính bảo mật của hệ thống và dữ liệu người dùng.

*Hiệu Suất và Tương Thích:*

* Đảm bảo hệ thống hoạt động ổn định và hiệu suất với dữ liệu lớn.
* Tương thích với các trình duyệt web và thiết bị di động phổ biến.

*Tính Khả Dụng và Giao Diện Người Dùng:*

* Đánh giá trải nghiệm người dùng, khả năng sử dụng và khả năng thích ứng với giao diện người dùng.

*Hỗ Trợ và Tài Liệu:*

* Tính khả dụng và hiệu quả của tài liệu hỗ trợ và trợ giúp người dùng.

**Quan Trọng của Các Yêu Cầu:**

* Các yêu cầu quản lý tài khoản và giao dịch, quản lý ngân sách cùng với báo cáo và thống kê có thể được ưu tiên cao vì chúng là trọng tâm của ứng dụng quản lý tài chính.
* Bảo mật và hiệu suất cũng là những yêu cầu không thể thiếu, vì đó là yếu tố quan trọng đối với mọi dự án phần mềm.

# CHIẾN LƯỢC TEST

## Các kiểu test

1. *Functional Testing*

Kỹ Thuật:

* Phạm Vi Test: Kiểm tra tính năng và chức năng của ứng dụng dựa trên yêu cầu.
* Hoạt Động Chính: Tạo test case dựa trên các yêu cầu chức năng và phi chức năng. Thực hiện các test case để xác minh tính đúng đắn và hoạt động của các chức năng.
* Đánh Giá Kết Quả: Sử dụng kết quả từ việc thực hiện test case để xác định xem các chức năng hoạt động đúng theo mong đợi hay không.

Điều Kiện Hoàn Thành:

* Tất cả test case đã được thực hiện và không có lỗi nào chưa được giải quyết.
* Tất cả yêu cầu chức năng đã được kiểm tra và chấp nhận.

Các Vấn Đề Đặc Biệt:

* Sự sẵn sàng của cơ sở dữ liệu test để hỗ trợ việc tạo, cập nhật và làm mới dữ liệu test.
* Đảm bảo rằng mục tiêu test phù hợp hoặc đồng bộ với hệ thống thực tế để test chức năng.

1. *Automation Testing*

Kỹ Thuật:

* Phạm Vi Test: Tự động hóa việc thực hiện test case và kiểm tra.
* Hoạt Động Chính: Sử dụng các công cụ tự động hóa test để thực hiện các test case đã thiết lập trước đó.
* Đánh Giá Kết Quả: Phân tích kết quả từ các công cụ tự động hóa để xác định sự thành công và tính đúng đắn của các test case.

Điều Kiện Hoàn Thành:

* Tất cả test case đã được thực hiện bằng công cụ tự động hóa.
* Số lượng lỗi còn lại sau quá trình kiểm tra không vượt quá một ngưỡng nhất định.

Các Vấn Đề Đặc Biệt:

* Sự sẵn sàng và ổn định của công cụ tự động hóa test để thực hiện các test case.
* Xác định và giải quyết các lỗi không mong muốn từ công cụ tự động hóa test.

1. *GUI and Usability Testing*

Kỹ Thuật:

* Phạm Vi Test: Đánh giá giao diện người dùng và trải nghiệm người dùng.
* Hoạt Động Chính: Thực hiện test để kiểm tra sự dễ sử dụng, thân thiện của giao diện người dùng, và kiểm tra tính hợp lý, trực quan của nó.
* Đánh Giá Kết Quả: Sử dụng phản hồi từ người dùng thử nghiệm để đánh giá trải nghiệm người dùng và hiệu suất của giao diện.

Điều Kiện Hoàn Thành:

* Tất cả các vấn đề liên quan đến giao diện và trải nghiệm người dùng đã được giải quyết hoặc cải thiện.

Các Vấn Đề Đặc Biệt:

* Đồng nhất hóa mục tiêu test với hệ thống thực tế để test giao diện người dùng và tính thích ứng.

1. *Performance Testing:*

Kỹ Thuật:

* Phạm Vi Test: Đánh giá hiệu suất hệ thống và xác định các thông số về tải.
* Hoạt Động Chính: Tạo và thực hiện các test case để đánh giá hiệu suất với các điều kiện tải khác nhau.
* Đánh Giá Kết Quả: Sử dụng kết quả từ việc thực hiện test case để xác định thời gian phản hồi và các chỉ số hiệu suất khác của hệ thống.

Điều Kiện Hoàn Thành:

* Thời gian phản hồi và hiệu suất hệ thống đáp ứng được các tiêu chí đã định.

Các Vấn Đề Đặc Biệt:

* Sử dụng máy chủ mạng phù hợp để thực hiện test hiệu suất trong các điều kiện không ảnh hưởng đến các giao dịch khác.
* Đồng nhất hóa mục tiêu test với hệ thống thực tế để test hiệu suất.

1. *Ghi Chú:*

Các điều kiện hoàn thành cần phải được định rõ để đảm bảo việc đánh giá được chính xác và hoàn thiện.

Các vấn đề đặc biệt cần được quan tâm để giảm thiểu các rủi ro có thể xảy ra trong quá trình kiểm thử và đảm bảo nguồn lực được sử dụng một cách hiệu quả.

### Test chức năng (Functional Testing)

#### Test chức năng (Function Testing)

|  |  |
| --- | --- |
| Mục đích test: | Đảm bảo tính đúng đắn của các chức năng trong Firefly III, bao gồm định hướng, dữ liệu đầu vào, xử lý và dữ liệu nhận được. |
| Cách thực hiện: | Xác định Phạm Vi Test: Định rõ các chức năng cần kiểm tra và các Use Case cần được thực hiện.  Chuẩn Bị Dữ Liệu: Tạo dữ liệu hợp lệ và không hợp lệ để sử dụng trong quá trình kiểm thử.  Thực Hiện Test: Thực hiện từng Use Case hoặc chu trình UC với dữ liệu hợp lệ và không hợp lệ. Kiểm tra kết quả mong đợi với dữ liệu hợp lệ và phát hiện lỗi khi dữ liệu không hợp lệ. Áp dụng các qui tắc nghiệp vụ để đảm bảo tính đúng đắn của chức năng.  Ghi Nhận Kết Quả: Ghi chép chi tiết về các lỗi phát hiện và kết quả kiểm thử. |
| Điều kiện hoàn thành: | Toàn Bộ Kế Hoạch Test Đã Được Thực Hiện: Tất cả các Test Cases đã được thực hiện theo kế hoạch đã xác định.  Tất Cả Các Lỗi Đã Được Ghi Nhận: Mọi lỗi hoặc sự không thể thực hiện chức năng phải được ghi nhận và báo cáo. |
| Các vấn đề đặc biệt: | Dữ Liệu Test: Đảm bảo sẵn có đủ dữ liệu để thực hiện test một cách toàn diện và đáng tin cậy.  Qui Trình Tương Tác: Đôi khi, các chức năng có thể phụ thuộc vào những hành động trước đó. Đảm bảo các bước tương tác trước đó được thực hiện một cách chính xác.  Lập Trình Nội Bộ: Sự thay đổi trong mã nguồn có thể ảnh hưởng đến chức năng, vì vậy cần kiểm tra lại sau mỗi lần thay đổi.  Điều Kiện Môi Trường: Môi trường kiểm thử cần phải phản ánh môi trường sản phẩm cuối để đảm bảo tính nhất quán trong việc kiểm tra. |

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

|  |  |
| --- | --- |
| Mục đích test: | Đảm bảo tính chính xác, dễ sử dụng và thân thiện của giao diện người dùng (UI) của Firefly III.  Kiểm tra tính tương thích và hiển thị đúng đắn trên các nền tảng và thiết bị khác nhau.  Xác minh các chức năng và sự tương tác của người dùng với giao diện. |
| Cách thực hiện: | Kiểm Tra Định Dạng và Hiển Thị: Kiểm tra việc hiển thị dữ liệu và thành phần giao diện trên các trình duyệt và thiết bị khác nhau.  Chức Năng Tương Tác: Kiểm tra các chức năng nhấn, kéo, thả, nhập liệu và lựa chọn trên giao diện.  Tương Thích Nền Tảng: Kiểm tra giao diện trên các nền tảng khác nhau: web, di động, desktop.  Đơn Vị Của Giao Diện: Kiểm tra từng thành phần giao diện như nút, ô nhập liệu, thanh công cụ, menu dropdown.  Thẩm Định Thông Tin: Xác nhận tính chính xác của thông tin hiển thị trên giao diện so với cơ sở dữ liệu. |
| Điều kiện hoàn thành: | Toàn bộ kiểm thử giao diện người sử dụng đã được thực hiện.  Các lỗi và vấn đề phát hiện ra đã được ghi chép và báo cáo. |
| Các vấn đề đặc biệt: | Đa Nền Tảng: Đảm bảo giao diện hoạt động một cách nhất quán trên nhiều nền tảng khác nhau (Web, iOS, Android, Desktop).  Phản Hồi Nhanh: Kiểm tra thời gian phản hồi của giao diện khi người dùng tương tác.  Tương Thích Trình Duyệt: Xác định và kiểm tra tương thích với các trình duyệt phổ biến như Chrome, Firefox, Safari.  Đa Ngôn Ngữ: Đảm bảo giao diện hiển thị đúng với các ngôn ngữ khác nhau được hỗ trợ.  Tính Tương Tác: Đảm bảo giao diện đáp ứng đúng các yêu cầu tương tác của người dùng và không gây hiểu lầm. |

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

|  |  |
| --- | --- |
| Mục đích test: | Đảm bảo tính toàn vẹn, chính xác và nhất quán của dữ liệu trong cơ sở dữ liệu của Firefly III.  Kiểm tra quá trình tích hợp dữ liệu và xác thực tính chính xác của dữ liệu được lưu trữ. |
| Cách thực hiện: | Kiểm Tra Dữ Liệu: Xác minh rằng dữ liệu được nhập vào hệ thống là chính xác và đầy đủ.  Kiểm Tra Tích Hợp Dữ Liệu: Kiểm tra tính hợp nhất của dữ liệu từ nhiều nguồn khác nhau (nếu có) vào cơ sở dữ liệu chính.  Kiểm Tra Toàn Vẹn Dữ Liệu: Xác minh rằng dữ liệu không bị mất mát và được lưu trữ một cách an toàn.  Kiểm Tra Nhất Quán Dữ Liệu: Đảm bảo tính nhất quán giữa các bảng, mối quan hệ giữa chúng và các ràng buộc dữ liệu.  Kiểm Tra Thông Tin Nhạy Cảm: Đảm bảo rằng dữ liệu nhạy cảm được bảo vệ một cách an toàn và không thể truy cập trái phép. |
| Điều kiện hoàn thành: | Tất cả các kịch bản kiểm thử về dữ liệu và tích hợp dữ liệu đã được thực hiện.  Tất cả các lỗi và vấn đề phát hiện ra đã được ghi chép và báo cáo. |
| Các vấn đề đặc biệt: | Bảo Mật Dữ Liệu: Đảm bảo rằng dữ liệu nhạy cảm được mã hóa và bảo mật đúng cách.  Backup và Recovery: Đảm bảo quá trình sao lưu và khôi phục dữ liệu được thực hiện một cách hiệu quả và đảm bảo tính nhất quán của dữ liệu.  Tính Nguyên Vẹn Dữ Liệu: Kiểm tra tính nguyên vẹn của dữ liệu sau quá trình tích hợp và xử lý dữ liệu.  Tương Tác Giữa Các Bảng Dữ Liệu: Kiểm tra và đảm bảo rằng thông tin giữa các bảng dữ liệu tương tác đúng như mong đợi.  Hiệu Suất Cơ Sở Dữ Liệu: Kiểm tra hiệu suất của cơ sở dữ liệu khi có lượng dữ liệu lớn và các tác vụ cập nhật/truy vấn. |

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

|  |  |
| --- | --- |
| Mục đích test: | Đảm bảo tính toàn vẹn, hiệu suất và đáp ứng của hệ thống Firefly III trong các chu trình hoạt động kinh doanh hay nghiệp vụ cụ thể.  Kiểm tra tính chính xác và hiệu suất của toàn bộ quy trình làm việc kinh doanh. |
| Cách thực hiện: | Xác Định Chu Trình Nghiệp Vụ: Định nghĩa và phân tích các chu trình hoạt động kinh doanh cần kiểm tra.  Thực Hiện Chu Trình Nghiệp Vụ: Thực hiện toàn bộ chu trình nghiệp vụ từ đầu đến cuối theo kịch bản đã được xác định.  Kiểm Tra Đáp Ứng và Hiệu Suất: Đánh giá sự đáp ứng của hệ thống Firefly III trong mỗi bước của chu trình nghiệp vụ. Đo lường hiệu suất trong môi trường thực tế.  Kiểm Tra Tính Toàn Vẹn Dữ Liệu: Kiểm tra dữ liệu được tạo ra hoặc thay đổi trong quá trình thực hiện chu trình.  Kiểm Tra Kết Quả và Báo Cáo: Xác nhận tính chính xác của kết quả và báo cáo được tạo ra bởi hệ thống sau mỗi chu trình. |
| Điều kiện hoàn thành: | Tất cả các chu trình nghiệp vụ đã được kiểm thử và đánh giá.  Các lỗi và vấn đề phát hiện ra đã được ghi chép và báo cáo. |
| Các vấn đề đặc biệt: | Xử Lý Ngoại Lệ: Kiểm tra khả năng của hệ thống trong việc xử lý các tình huống ngoại lệ hoặc lỗi không dự đoán.  Tương Tác Hệ Thống Ngoài: Đảm bảo tính liên kết của hệ thống Firefly III với các hệ thống bên ngoài trong các chu trình nghiệp vụ.  Tính Nguyên Vẹn Dữ Liệu: Kiểm tra tính toàn vẹn và đúng đắn của dữ liệu sau mỗi chu trình nghiệp vụ.  Hiệu Suất và Tải Trọng: Đánh giá hiệu suất và khả năng chịu tải của hệ thống trong điều kiện hoạt động thực tế.  Phản Hồi Người Dùng: Thu thập phản hồi từ người dùng về trải nghiệm và chất lượng của chu trình nghiệp vụ. |

### Test hiệu suất (Performance testing)

#### Performance Profiling

|  |  |
| --- | --- |
| Mục đích test: | Đánh giá và xác định các điểm yếu trong mã nguồn và tìm ra các khu vực cần tối ưu hóa để cải thiện hiệu suất của ứng dụng Firefly III. |
| Cách thực hiện: | Xác định Nhiệm Vụ và Kịch Bản Test: Xác định chức năng cụ thể hoặc kịch bản mà Performance Profiling sẽ được áp dụng.  Thu Thập Dữ Liệu Hiệu Suất: Sử dụng các công cụ đo lường hiệu suất để thu thập dữ liệu liên quan đến thời gian thực thi, tài nguyên tiêu thụ, và chi tiết về hoạt động của ứng dụng.  Phân Tích và Đánh Giá Dữ Liệu: Phân tích dữ liệu thu thập để xác định các vấn đề hiệu suất, điểm nghẽn, và các khu vực cần cải thiện.  Tối Ưu Mã Nguồn và Tinh chỉnh Hiệu Suất: Dựa trên phân tích, thực hiện tối ưu mã nguồn và các thay đổi cần thiết để cải thiện hiệu suất của ứng dụng. |
| Điều kiện hoàn thành: | Điều Kiện Chính: Các chỉ số hiệu suất đã được thu thập và phân tích.  Tối Ưu Hóa: Cải thiện và tối ưu hóa mã nguồn để giảm thời gian thực thi và tăng hiệu suất. |
| Các vấn đề đặc biệt: | Thiếu Các Công Cụ Hiệu Suất Phù Hợp: Nếu không có các công cụ hiệu suất phù hợp, việc đánh giá và tối ưu hóa có thể gặp khó khăn.  Phức Tạp Của Mã Nguồn: Mã nguồn phức tạp hoặc không tối ưu có thể làm gia tăng khó khăn trong việc xác định và cải thiện hiệu suất.  Tính Nhạy Cảm của Dữ Liệu: Nếu dữ liệu là nhạy cảm, việc thu thập và sử dụng dữ liệu có thể đòi hỏi sự cẩn trọng và tuân thủ quy định bảo mật.  Công Cụ và Tài Nguyên Cần Thiết: Sử dụng các công cụ phân tích và tối ưu hóa hiệu suất yêu cầu kiến thức chuyên sâu và có thể cần tài nguyên mạnh mẽ để thực hiện. tế hoặc đo bằng nhau |

#### Load Testing

|  |  |
| --- | --- |
| Mục tiêu test: | Đánh giá hiệu suất và khả năng của hệ thống Firefly III dưới các điều kiện tải lớn, đồng thời xác định khả năng của hệ thống để xử lý workload cao mà vẫn duy trì được hiệu suất mong đợi. |
| Cách thực hiện: | Xác Định Kịch Bản Tải: Định nghĩa các kịch bản tải có thể phù hợp với ứng dụng Firefly III dựa trên các giao dịch nghiệp vụ logic và tình huống thực tế.  Thực Hiện Tải Lớn: Thực hiện kiểm thử với tải lớn đồng thời hoặc tăng dần số lượng người dùng/volume giao dịch để kiểm tra khả năng chịu tải của hệ thống.  Đo Lường và Thu Thập Dữ Liệu: Thu thập dữ liệu về thời gian phản hồi, tỷ lệ giao dịch và các chỉ số hiệu suất khác trong quá trình tải lớn.  Đánh Giá Hiệu Suất: Phân tích dữ liệu thu thập để xác định hiệu suất của hệ thống Firefly III dưới tải lớn, nhận biết điểm yếu và mức độ đáp ứng. |
| Điều kiện hoàn thành: | Thực Hiện Tải Lớn: Tất cả các kịch bản tải đã được thực hiện theo kế hoạch.  Đánh Giá Hiệu Suất: Dữ liệu đã được phân tích và đánh giá, nhằm xác định các vấn đề hiệu suất và đáp ứng của hệ thống. |
| Các vấn đề đặc biệt: | Chuẩn Bị Dữ Liệu Tải: Dữ liệu thực tế và tập hợp tình huống sử dụng gần với điều kiện thực tế để tạo ra các kịch bản tải chân thực và hợp lý.  Thời Gian Thực Thi: Thời gian thực thi để thực hiện load testing có thể tăng lên đáng kể tùy thuộc vào kích thước của tải và khả năng chịu tải của hệ thống.  Quản Lý Tài Nguyên: Tài nguyên hệ thống như CPU, bộ nhớ và băng thông mạng cần được quản lý chặt chẽ để thực hiện tải lớn mà không ảnh hưởng đến môi trường khác.  Định Lượng Và Đánh Giá Kết Quả: Việc thu thập và đánh giá dữ liệu từ load testing cần sự chính xác để xác định chính xác hiệu suất của hệ thống. |

#### Stress Testing

|  |  |
| --- | --- |
| Mục đích test: | Xác định và đánh giá khả năng của hệ thống Firefly III dưới áp lực và tải nặng, nhằm tìm ra các vấn đề tiềm ẩn khi tài nguyên không đủ hoặc khi có sự cạnh tranh về tài nguyên. |
| Cách thực hiện: | Tạo Tải Nặng: Tạo điều kiện tải nặng đồng thời hoặc tăng dần tải để thử nghiệm khả năng chịu tải của hệ thống Firefly III.  Đánh Giá Sức Chịu Đựng: Đánh giá khả năng của hệ thống trong những điều kiện áp lực cao, bao gồm việc kiểm tra sự ổn định, đáp ứng và khả năng phục hồi sau khi trải qua tải nặng.  Đo Lường Tối Đa Có Thể Điều Khiển: Xác định tải làm việc tối đa mà hệ thống có thể xử lý mà vẫn duy trì được hiệu suất ổn định và không gặp phải các vấn đề nghiêm trọng. |
| Điều kiện hoàn thành: | Đạt Được Giới Hạn Tải Chịu Đựng: Xác định được giới hạn tải mà hệ thống có thể chịu đựng mà không gặp phải các vấn đề nghiêm trọng. |
| Các vấn đề đặc biệt: | Quản Lý Tài Nguyên Hệ Thống: Stress testing yêu cầu việc quản lý chặt chẽ các tài nguyên hệ thống như bộ nhớ, CPU, và băng thông để đảm bảo rằng các vấn đề không phát sinh từ thiếu tài nguyên.  Xác Định Lỗi Tài Nguyên: Tìm ra các lỗi có thể phát sinh từ sự cạnh tranh về tài nguyên như cạnh tranh hoặc chia sẻ tài nguyên như khóa cơ sở dữ liệu hoặc băng thông mạng.  Thiết Lập Giới Hạn Tải: Việc xác định giới hạn tải mà hệ thống có thể chịu đựng là một yếu tố quan trọng để đảm bảo rằng nó có thể hoạt động ổn định trong điều kiện cực đoan. |

#### Volume Testing

|  |  |
| --- | --- |
| Mục đích test: | Kiểm tra và xác định giới hạn của phần mềm Firefly III khi xử lý dữ liệu với lượng lớn, nhằm đảm bảo tính ổn định và độ chính xác của phần mềm trong điều kiện tải nặng. |
| Cách thực hiện: | Tạo Lượng Dữ Liệu Lớn: Tạo các tập dữ liệu lớn để kiểm tra xem phần mềm có xử lý được dữ liệu này một cách chính xác và hiệu quả không.  Thực Hiện Các Chức Năng Quan Trọng: Thực hiện các chức năng quan trọng mà yêu cầu xử lý dữ liệu lớn, đồng thời đánh giá hiệu suất và khả năng của hệ thống trong điều kiện này.  Đánh Giá Độ Ổn Định: Đánh giá xem phần mềm có duy trì được hiệu suất và ổn định không khi xử lý dữ liệu với lượng lớn. |
| Điều kiện hoàn thành: | Dữ Liệu Xử Lý Đúng: Đảm bảo rằng phần mềm xử lý dữ liệu lớn một cách chính xác và không gặp phải các lỗi dữ liệu. |
| Các vấn đề đặc biệt: | Quản Lý Tài Nguyên: Cần quản lý tài nguyên hệ thống chặt chẽ để đảm bảo rằng việc xử lý dữ liệu lớn không gây ra các vấn đề về tài nguyên.  Kiểm Tra Độ Ổn Định: Việc kiểm tra độ ổn định và đáng tin cậy của phần mềm khi xử lý dữ liệu lớn là quan trọng để đảm bảo rằng nó hoạt động như mong đợi. |

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

|  |  |
| --- | --- |
| Mục đích test: | Kiểm tra và đảm bảo tính bảo mật của ứng dụng Firefly III ở cả mức ứng dụng và mức hệ thống. |
| Cách thực hiện: | Bảo Mật Mức Ứng Dụng:  Kiểm tra việc hạn chế quyền truy cập vào các chức năng và dữ liệu phù hợp với vai trò của người dùng.  Đảm bảo rằng người dùng chỉ có thể thực hiện các hành động mà họ được cấp quyền.  Bảo Mật Mức Hệ Thống:  Xác định và kiểm tra các cổng truy cập vào hệ thống và ứng dụng.  Đảm bảo rằng chỉ người dùng được phép mới có thể truy cập hệ thống thông qua các cổng truy cập hợp lệ. |
| Điều kiện hoàn thành: | Kiểm Tra Các Điều Kiện Bảo Mật:  Đảm bảo rằng mọi điều kiện bảo mật đã được kiểm tra và xác minh. |
| Các vấn đề đặc biệt: | Kiểm Tra Điều Kiện Truy Cập:  Kiểm tra xem hệ thống có thể phân biệt và áp dụng đúng các quyền truy cập cho từng nhóm người dùng khác nhau không.  Kiểm Tra Tính An Toàn Của Dữ Liệu:  Đảm bảo rằng dữ liệu được bảo mật phù hợp với cấp độ truy cập của từng người dùng.  Kiểm Tra Các Cổng Truy Cập Hệ Thống:  Xác minh rằng chỉ có các cổng truy cập được cho phép mới có thể truy cập vào hệ thống. |

### Test hồi qui (Regression Testing)

|  |  |
| --- | --- |
| Mục đích test: | Xác nhận rằng việc thay đổi code hoặc thực hiện cải tiến mới không gây ra ảnh hưởng tiêu cực đến các chức năng đã được kiểm thử trước đó. |
| Cách thực hiện: | Chuẩn Bị Dữ Liệu Test:  Xác định tập dữ liệu và kịch bản test cần thiết để kiểm tra các chức năng cụ thể sau khi thực hiện thay đổi.  Thực Hiện Test Hồi Qui:  Chạy lại các bộ test chức năng hoặc các kịch bản test liên quan đến các phần được thay đổi.  So Sánh Kết Quả:  So sánh kết quả của việc thực hiện test với kết quả ban đầu để kiểm tra sự ổn định và xác nhận không có sự thay đổi không mong muốn. |
| Điều kiện hoàn thành: | Không Có Sự Thay Đổi Không Mong Muốn:  Đảm bảo rằng sau khi thực hiện test hồi qui, không có sự thay đổi không mong muốn trong các chức năng đã được kiểm tra trước đó. |
| Các vấn đề đặc biệt: | Phạm Vi Test Hồi Qui:  Xác định phạm vi test hồi qui để tập trung vào các phần quan trọng nhất và liên quan đến các thay đổi gần đây nhất.  Cập Nhật Kịch Bản Test:  Cập nhật các kịch bản test để đảm bảo rằng chúng phản ánh đúng các thay đổi mới và tương thích với phiên bản mới.  Chạy Test Đa Dạng:  Thực hiện test hồi qui trên nhiều môi trường khác nhau để đảm bảo tính nhất quán và ổn định. |

## Giai đoạn test

| 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

|  |  |  |  |
| --- | --- | --- | --- |
| Mục đích | Công cụ | Nhà cung cấp/Tự xây dựng | Phiên bản |
| Kiểm thử tự động | Katalon Studio | Nhà cung cấp | Phiên bản 8.x |
| Quản lý dữ liệu | Microsoft Excel | Tự xây dựng | Phiên bản mới nhất |

## Môi trường test

Unit Test:

* Khi test chạy chương trình: Thông thường chạy trên code.
* Cơ sở dữ liệu: Sử dụng cơ sở dữ liệu độc lập hoặc có thể sử dụng cơ sở dữ liệu phát triển.
* Điều kiện về mạng: Thường sử dụng mạng LAN.
* Mô hình cài đặt: Thiết lập như khi lập trình, với ít máy chủ hoặc máy trạm.

Integration Test:

* Khi test chạy chương trình: Thường chạy trên code hoặc bản dịch.
* Cơ sở dữ liệu: Cần thiết lập database riêng và tương tự hoặc gần giống với môi trường chạy thực tế.
* Điều kiện về mạng: Sử dụng hệ thống đường truyền giống như hoặc gần giống với môi trường chạy thực tế.
* Mô hình cài đặt: Cần phải chú ý thiết lập sao cho gần giống mô hình sẽ chạy trong thực tế nhất.

System Test:

* Khi test chạy chương trình: Chạy trên bản dịch.
* Cơ sở dữ liệu: Sử dụng database tương tự hoặc giống với môi trường chạy thực tế.
* Điều kiện về mạng: Sử dụng hệ thống đường truyền tương tự hoặc gần giống với môi trường chạy thực tế.
* Mô hình cài đặt: Phải thiết lập sao cho gần giống mô hình sẽ chạy trong thực tế nhất.

Acceptance Test:

* Khi test chạy chương trình: Chạy trên bản dịch.
* Cơ sở dữ liệu: Sử dụng database giống với môi trường chạy thực tế.
* Điều kiện về mạng: Sử dụng hệ thống đường truyền giống với môi trường chạy thực tế.
* Mô hình cài đặt: Thiết lập để phản ánh mô hình sẽ chạy trong môi trường thực tế.

# TÀI NGUYÊN

## Nhân lực

|  |  |
| --- | --- |
| Họ tên | Trách nhiệm/Ghi chú |
| Nguyễn Văn Hải | Test plane và thực hiện test 2 chức năng trong kiểm thử (Đăng nhập và account) |
| Phạm Trường Khoa | Viết kịch bản cho performance testing và thực hiện test 2 chức năng trong kiểm thử (Piggy banks và Classification) |
| Phạm Quốc Khánh | Tìm hiểu, set up test management tool và thực hiện test 2 chức năng trong kiểm thử (Report và Bill) |
| Nguyễn Sỹ Phong | Thực hiện generate data và thực hiện test 2 chức năng trong kiểm thử (Budget và Transaction) |

## Hệ thống

**Yêu cầu về Phần Cứng:**

Máy tính:

* Có thể chạy các ứng dụng và công cụ kiểm thử một cách mượt mà và hiệu quả.
* Đủ tài nguyên để cài đặt các phần mềm liên quan và chạy các chương trình kiểm thử.

Môi trường Network:

* Kết nối mạng ổn định để truy cập vào các tài nguyên cần thiết và kiểm thử chức năng liên quan đến mạng.

Các thiết bị phần cứng đặc biệt (nếu có):

* Nếu có yêu cầu kiểm thử cụ thể về tính năng cần thiết đối với các thiết bị cụ thể (ví dụ: thiết bị di động, máy đọc thẻ, thiết bị giao tiếp khác), cần có các thiết bị này để kiểm tra tính năng tương ứng.

**Yêu Cầu về Phần Mềm:**

Hệ điều hành:

* Cung cấp hệ điều hành tương thích với các công cụ phát triển và kiểm thử.
* Có thể là Windows, macOS, Linux hoặc các phiên bản khác tùy thuộc vào sự tương thích với công cụ cụ thể.

Công cụ Kiểm thử:

* Automation Testing Tools: Có thể sử dụng Katalon Studio, Selenium, hoặc các công cụ tự động hóa khác để thực hiện kiểm thử tự động.
* Công cụ Quản lý Test Case và Report: Cần có các công cụ để quản lý test case, theo dõi kết quả kiểm thử và tạo báo cáo.

Trình duyệt Web và Ứng dụng:

* Đảm bảo sẵn có các trình duyệt web phổ biến như Google Chrome, Firefox, Safari, Microsoft Edge, để kiểm tra tính tương thích của ứng dụng web trên các nền tảng khác nhau.
* Nếu kiểm thử ứng dụng di động, cần cài đặt và kiểm tra trên các thiết bị và trình giả lập di động khác nhau.

Các Framework và thư viện liên quan:

* Có thể cần cài đặt các framework và thư viện cụ thể phụ thuộc vào ngôn ngữ lập trình và công nghệ sử dụng để phát triển Firefly.

Cơ sở dữ liệu:

* Yêu cầu cài đặt cơ sở dữ liệu phù hợp (ví dụ: MySQL, PostgreSQL, MongoDB) nếu kiểm thử liên quan đến cơ sở dữ liệu.
* Để đảm bảo hiệu quả kiểm thử, việc tuân thủ và cung cấp các yêu cầu về phần cứng và phần mềm trên là cực kỳ quan trọng để đảm bảo team kiểm thử có môi trường phù hợp để thực hiện các hoạt động kiểm thử một cách hiệu quả.

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

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone Task | Nguồn lực | Ngày bắt đầu | Ngày kết thúc |
| Tìm hiểu về firefly III và cài đặt trên máy | Nhóm test | 26/12/24 | 2/1/24 |
| Cài đặt, tìm hiểu test manager tool | Nhóm test | 26/12/24 | 2/1/24 |
| Tìm hiểu test plane, generate data để test | Nhóm test | 26/12/24 | 2/1/24 |
| Viết scenario và hoàn thành functional testing | Nhóm test | 26/12/24 | 2/1/24 |
| Hoàn thành GUI and Usability testing | Nhóm test | 26/12/24 | 2/1/24 |
| Hoàn thành Automation testing | Nhóm test, Automation Engineer | 26/12/24 | 2/1/24 |
| Hoàn thành các document liên quan và Performance testing | Nhóm test, Technical Writer | 2/1/24 | 9/1/24 |

# CÁC SẢN PHẨM

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