



**KHOA CÔNG NGHỆ THÔNG TIN**  
**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  
**ĐẠI HỌC QUỐC GIA TP HCM**

**Firefly-iii**  
**Kế hoạch test**

**Mã dự án: 007**

**Mã tài liệu: 007**

**TPHCM, ngày 09 tháng 01 năm 2024**



**TRANG KÝ****Người lập:** \_\_\_\_\_

&lt;Chức vụ&gt;

&lt;Ngày&gt; \_\_\_\_\_

**Người xem xét:** \_\_\_\_\_

&lt;Chức vụ&gt;

&lt;Ngày&gt; \_\_\_\_\_

\_\_\_\_\_

&lt;Chức vụ&gt;

&lt;Ngày&gt; \_\_\_\_\_

**Người phê duyệt:** \_\_\_\_\_

&lt;Chức vụ&gt;

&lt;Ngày&gt; \_\_\_\_\_

## MỤC LỤC

<b>1</b>	<b>GIỚI THIỆU .....</b>	<b>5</b>
1.1	Mục đích .....	5
1.2	Thông tin chung .....	6
1.3	Tài liệu liên quan .....	7
1.4	Phạm vi test.....	7
1.5	Ràng buộc .....	8
1.6	Liệt kê các mạo hiểm .....	8
<b>2</b>	<b>CÁC YÊU CẦU CHO TEST (LÀM NHÓM).....</b>	<b>9</b>
<b>3</b>	<b>CHIẾN LƯỢC TEST .....</b>	<b>11</b>
3.1	Các kiểu test.....	11
3.1.1	Test chức năng (Functional Testing) .....	11
3.1.2	Test hiệu suất (Performance testing) .....	12
3.2	Giai đoạn test .....	13
3.3	Các công cụ test .....	14
3.4	Môi trường test.....	14
<b>4</b>	<b>TÀI NGUYÊN .....</b>	<b>15</b>
4.1	Nhân lực.....	15
4.2	Hệ thống.....	16
<b>5</b>	<b>CÁC MỐC KIỂM SOÁT CỦA GIAI ĐOẠN TEST (TEST MILESTONES) .....</b>	<b>16</b>
<b>6</b>	<b>CÁC SẢN PHẨM.....</b>	<b>17</b>

# 1 GIỚI THIỆU

## 1.1 Mục đích

Mục đích của tài liệu.

- Xác định chiến lược kiểm thử để đảm bảo rằng phần mềm được phát triển đáp ứng đầy đủ yêu cầu kỹ thuật và các đặc tả ban đầu.
- Đặt ra mục tiêu và phạm vi cụ thể của quá trình kiểm thử để đảm bảo chất lượng và độ tin cậy của sản phẩm.
- Xây dựng kế hoạch quản lý kiểm thử, bao gồm các phương pháp và quy trình quản lý để giảm thiểu rủi ro và tối ưu hóa hiệu suất kiểm thử.
- Đề xuất các yêu cầu về phần cứng, phần mềm, và công cụ kiểm thử cần thiết để thực hiện quá trình kiểm thử một cách hiệu quả.
- Xác định rõ các đặc điểm và chức năng cụ thể sẽ được kiểm thử để đảm bảo tất cả các khía cạnh của sản phẩm được đánh giá.
- Định danh và đánh giá nguy cơ tiềm ẩn trong quá trình phát triển, cung cấp phương tiện để phòng tránh và giải quyết các vấn đề xuất phát.
- Lên kế hoạch và giao việc kiểm thử, đặt ra các tiêu chí đánh giá cụ thể và phân công công việc chi tiết cho từng thành viên trong nhóm kiểm thử.

Tổ chức của tài liệu: Gồm 6 phần chính:

- Giới thiệu chung: mô tả mục đích, phạm vi test, các tài liệu liên quan, phạm vi test, các kiểu test, các ràng buộc, các mạo hiểm.
- Các yêu cầu cho test: tình huống test, các yêu cầu chức năng và phi chức năng.
- Chiến lược test: giới thiệu phương án tiếp cận để test các mục tiêu test.
- Tài nguyên: về nhân lực và kỹ thuật, đào tạo huấn luyện (nếu có)
- Các mốc kiểm soát của giai đoạn test: lập thời biểu, thời gian hoàn thành, phân công công việc.
- Các sản phẩm: testcases, test report.

## 1.2 Thông tin chung

"Firefly III" là trình quản lý (tự lưu trữ) cho tài chính cá nhân của bạn. Nó có thể giúp bạn theo dõi chi phí và thu nhập của mình để bạn có thể chi tiêu ít hơn và tiết kiệm nhiều hơn.

Quản lý tài chính cá nhân khá khó khăn và mọi người đều có cách tiếp cận riêng. Một số người lập ngân sách, những người khác hạn chế dòng tiền của họ bằng cách rút bỏ thẻ tín dụng, những người khác cố gắng tăng dòng tiền hiện tại của họ. Có rất nhiều cách để tiết kiệm và kiếm tiền. Firefly III hoạt động theo nguyên tắc nếu bạn biết tiền của mình sẽ đi đâu thì bạn có thể ngăn nó đi đến đó.

Bằng cách theo dõi chi phí và thu nhập của mình, bạn có thể lập ngân sách phù hợp và tiết kiệm tiền. Hãy ngừng sống từ đồng lương này đến đồng lương khác mà hãy dành cho mình khoảng trống tài chính mà bạn cần.

Danh sách các tính năng:

- Tạo các giao dịch định kỳ để quản lý tiền của bạn.
- Xử lý giao dịch dựa trên quy tắc với khả năng tạo quy tắc của riêng bạn
- Hệ thống sổ sách kế toán kép
- Lưu trữ, chỉnh sửa và xóa các khoản rút tiền, gửi tiền và chuyển khoản. Điều này cho phép bạn quản lý tài chính đầy đủ.
- Bạn có thể quản lý các loại tài khoản khác nhau.
- Tài khoản tài sản - Asset accounts.
- Tài khoản tài sản chung (tài khoản hộ gia đình) - Shared asset accounts (household accounts).
- Tài khoản tiết kiệm - Saving accounts.
- Thẻ tín dụng.
- Nợ phải trả (các khoản vay, thế chấp).
- Có thể tạo, thay đổi và quản lý tiền bằng cách sử dụng ngân sách.
- Tổ chức các giao dịch theo danh mục.
- Tiết kiệm để đạt được mục tiêu bằng cách sử dụng heo đất
- Dự đoán hóa đơn.

- Xem báo cáo thu nhập và chi phí.

Mục đích kiểm thử phần mềm này:

- Functional Testing: Kịp thời phát hiện lỗi trong quá trình phát triển phần mềm.
- Performance Testing: Tăng hiệu suất hoạt động cho ứng dụng.
- GUI and Usability Testing: Điều chỉnh UI/UX phù hợp với người dùng.

### 1.3 Tài liệu liên quan

STT	Tên tài liệu	Nguồn	Ghi chú
1	Mẫu báo cáo Test Plan	Moodle khoa Công nghệ thông tin	

### 1.4 Phạm vi test

Các giai đoạn test :

- System test

Các kiểu test:

- Functional Testing.
- GUI and Usability testing.
- Performance Testing.

Các chức năng được test (làm nhóm):

- Tạo piggy bank.
- Tạo Expense Transaction.
- Tạo Transfers Transaction
- Tạo Bill
- Tạo Expense Account
- Tạo Revenue Account
- Tạo Asset Account
- Tạo Recurring Transaction

Các chức năng không được test:

- Đăng nhập, Đăng ký.
- Tạo webook.
- Classification.
- Report.
- Export Data.
- Options.

Các giả thiết trong quá trình lập kế hoạch:

- Đảm bảo tiến độ thực hiện theo kế hoạch đề ra được duy trì đúng như dự kiến.
- Đảm bảo nguồn nhân lực và vật chất kỹ thuật đáp ứng đầy đủ và hiệu quả.
- Xây dựng các kênh giao tiếp ban đầu mạnh mẽ để minh họa rõ vai trò và trách nhiệm của từng thành viên trong dự án.
- Bảo đảm không có sự xung đột giữa các thành viên tham gia, đảm bảo sự hòa thuận và hiệu suất tốt nhất của nhóm.

### 1.5 Ràng buộc

- Đảm bảo trang web hoạt động tốt trên trình duyệt Google Chrome, hệ điều hành Windows, Linux, MacOS.
- Chỉ kiểm thử một số đơn vị tiền tệ nhất định khi tạo Account, trong đó có Euro, Pound, ...
- Người dùng đăng nhập thành công.

### 1.6 Liệt kê các mạo hiểm

Stt	Mạo hiểm	Phương án khắc phục & phòng ngừa	Mức độ ảnh hưởng (MD)
1	Thời gian đáp ứng mỗi khi phát hiện lỗi bên phía developer có thể chậm.	Thường phải có feedback quy định trong khoản thời gian nào đó.	Không nghiêm trọng



2	Không kiểm thử hết tất cả các đơn vị tiền tệ hiện có nên có một số đơn vị tiền tệ hoạt động không chính xác, gây lỗi khi test.	Chỉ Enable một số đơn vị tiền tệ hoạt động ổn định.	Không nghiêm trọng
3	Rủi ro về mặt nhân sự trong quá trình thực hiện.	Cần phải có phương án dự phòng trong lịch biểu.	Nghiêm trọng

## 2 CÁC YÊU CẦU CHO TEST (LÀM NHÓM)

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.

- Tạo piggy bank:
  - Tạo được piggy bank khi các trường dữ liệu được nhập hợp lệ, dữ liệu được cập nhật vào database.
  - Xuất hiện các dòng thông báo khi tạo thành công hoặc có thông báo lỗi khi tạo không thành công. Có cụ thể lỗi ở dưới Text Field của dữ liệu.
  - Giao diện Tạo piggy bank phải bố trí để người dùng biết mình phải làm gì.
- Tạo Expense Transaction:
  - Tạo được Expense Transaction khi các trường dữ liệu được nhập hợp lệ, dữ liệu được cập nhật vào database.
  - Xuất hiện các dòng thông báo khi tạo thành công hoặc có thông báo lỗi khi tạo không thành công. Có cụ thể lỗi ở dưới Text Field của dữ liệu.
  - Giao diện Tạo piggy bank phải bố trí để người dùng biết mình phải làm gì.
- Tạo Transfers Transaction:
  - Tạo được Transfer Transaction khi các trường dữ liệu được nhập hợp lệ, dữ liệu được cập nhật vào database.
  - Xuất hiện các dòng thông báo khi tạo thành công hoặc có thông báo lỗi khi tạo không thành công. Có cụ thể lỗi ở dưới Text Field của dữ liệu.

- Giao diện Tạo Transfer Transaction phải bố trí để người dùng biết mình phải làm gì.
- Tạo Bill:
  - Tạo được Bill khi các trường dữ liệu được nhập hợp lệ, dữ liệu được cập nhật vào database.
  - Xuất hiện các dòng thông báo khi tạo thành công hoặc có thông báo lỗi khi tạo không thành công. Có cụ thể lỗi ở dưới Text Field của dữ liệu.
  - Giao diện Tạo Bill phải bố trí để người dùng biết mình phải làm gì.
- Tạo Expense Account:
  - Tạo Expense Account thành công khi các trường nhập liệu hợp lệ, dữ liệu sẽ được cập nhật vào database.
  - Xuất hiện thông báo khi tạo thành công hoặc thông báo lỗi chỗ trường nhập liệu không hợp lệ.
  - Giao diện Tạo Expense Account phải bố trí để người dùng biết mình phải làm gì.
- Tạo Renvenue Account:
  - Tạo Renvenue Account thành công khi các trường nhập liệu hợp lệ, dữ liệu sẽ được cập nhật vào database.
  - Xuất hiện thông báo khi tạo thành công hoặc thông báo lỗi chỗ trường nhập liệu không hợp lệ.
  - Giao diện Tạo Renvenue Account phải bố trí để người dùng biết mình phải làm gì.
- Tạo Asset Account:
  - Tạo Asset Account thành công khi các trường nhập liệu hợp lệ, dữ liệu sẽ được cập nhật vào database.
  - Xuất hiện thông báo khi tạo thành công hoặc thông báo lỗi chỗ trường nhập liệu không hợp lệ.
  - Giao diện Tạo Aseset Account phải bố trí để người dùng biết mình phải làm gì.
- Tạo Recurring Transaction:

- Tạo Recurring Transaction thành công khi các trường nhập liệu hợp lệ , dữ liệu sẽ được cập nhật vào database.
- Xuất hiện thông báo khi tạo thành công hoặc thông báo lỗi chỗ trường nhập liệu không hợp lệ.
- Giao diện Tạo Recurring Transaction phải bố trí để người dùng biết mình phải làm gì.

### 3 CHIẾN LƯỢC TEST

Phương án tiếp cận test được thiết kế để đảm bảo sự toàn vẹn và chất lượng của phần mềm thông qua việc áp dụng kỹ thuật test hiệu quả và thiết lập các điều kiện rõ ràng để đánh giá việc hoàn thành của quá trình kiểm thử. Dưới đây là các khía cạnh chi tiết của phương án:

#### 3.1 Các kiểu test

##### 3.1.1 Test chức năng (Functional Testing)

##### 3.1.1.1 Test chức năng (Function Testing)

<b>Mục đích test:</b>	Đảm bảo các chức năng được kiểm tra hoạt động chính xác theo yêu cầu, bao gồm định hướng, dữ liệu đầu vào, xử lý và dữ liệu nhận được
<b>Cách thực hiện:</b>	Với mỗi UC sử dụng dữ liệu hợp lệ và không hợp lệ để kiểm tra: <ul style="list-style-type: none"> <li>- Kết quả mong đợi với dữ liệu hợp lệ.</li> <li>- Lỗi thích hợp hoặc thông báo hiển thị khi dữ liệu không hợp lệ.</li> <li>- Mỗi qui tắc nghiệp vụ đều được áp dụng đúng.</li> </ul>
<b>Điều kiện hoàn thành:</b>	<ul style="list-style-type: none"> <li>- Toàn bộ kế hoạch test đã được thực hiện.</li> <li>- Toàn bộ các lỗi phát hiện ra đều đã được ghi nhận.</li> </ul>
<b>Các vấn đề đặc biệt:</b>	<ul style="list-style-type: none"> <li>- Liệt kê và xác định hết tất cả các vấn đề phát sinh trong quá trình kiểm thử.</li> </ul>

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

<b>Mục đích test:</b>	<p>Kiểm tra:</p> <ul style="list-style-type: none"> <li><input type="checkbox"/> 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).</li> <li><input type="checkbox"/> Các đối tượng và thuộc tính màn hình như menus, size, position, state, và tập trung vào việc tương thích với chuẩn.</li> </ul>
<b>Cách thực hiện:</b>	<p>- Tạo và Chỉnh Sửa Test:</p> <p>Phát triển và điều chỉnh các bộ test cho mỗi màn hình để kiểm tra việc sử dụng đúng cách.</p> <p>Đảm bảo rằng mọi chức năng của màn hình được kiểm thử một cách đầy đủ.</p> <p>- Kiểm Tra Đối Tượng và Thuộc Tính Màn Hình:</p> <p>Kiểm tra menus, size, position, state và các thuộc tính khác của màn hình.</p> <p>Tập trung vào việc đảm bảo tương thích với các chuẩn và hướng dẫn thiết kế giao diện người sử dụng.</p>
<b>Điều kiện hoàn thành:</b>	Mỗi màn hình được kiểm tra thành công đúng với phiên bản kiểm tra hoặc phạm vi chấp nhận được.
<b>Các vấn đề đặc biệt:</b>	<p>- Truy Cập Đối Tượng:</p> <p>Kiểm tra xem toàn bộ các thuộc tính của các đối tượng có thể truy cập được hay không.</p> <p>Điều này đặc biệt quan trọng để đảm bảo tính tương tác và sử dụng thuận lợi cho người dùng.</p>

### 3.1.2 Test hiệu suất (Performance testing)

#### 3.1.1.3 Load Testing

<b>Mục tiêu test:</b>	Kiểm tra hiệu suất về thời gian cho các giao dịch hoặc tình huống nghiệp vụ đã thiết kế dưới nhiều điều kiện làm việc (workload) khác nhau. Mục tiêu là đảm bảo rằng hệ thống có
-----------------------	--

	khả năng xử lý một lượng lớn giao dịch mà vẫn duy trì thời gian đáp ứng chấp nhận được.
<b>Cách thực hiện:</b>	<p>Sử dụng các bộ test đã được xây dựng cho test chức năng và chu trình nghiệp vụ.</p> <p>Sửa lại file dữ liệu để tăng số lượng giao dịch hoặc số lượng người dùng để mô phỏng các điều kiện làm việc tải cao.</p> <p>Thực hiện test dưới nhiều điều kiện workload khác nhau để đánh giá hiệu suất thời gian của hệ thống.</p>
<b>Điều kiện hoàn thành:</b>	<p>Nếu kiểm tra được thực hiện thành công mà không có lỗi đáng kể.</p> <p>Thời gian đáp ứng của hệ thống vẫn nằm trong khoảng thời gian chấp nhận được.</p>
<b>Các vấn đề đặc biệt:</b>	<p>- Thời Gian Thực Hiện Load Testing:</p> <p>Load testing cần phải được thực hiện trong khoảng thời gian chuyên dụng hoặc vào những giờ chuyên biệt để đảm bảo đo đạc đầy đủ và chính xác.</p> <p>- Kích Thước Cơ Sở Dữ Liệu:</p> <p>Cơ sở dữ liệu sử dụng cho load testing cần phải có kích thước thực tế hoặc được đo đạc bằng cách đảm bảo rằng dung lượng của nó phản ánh môi trường thực tế.</p>

### 3.2 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			X	X

Kiểu test	Giai đoạn test			
	Unit	Integration	System	Acceptance
(Load Test)				
Reliability (Integrity, Structure)	X	X		

### 3.3 Các công cụ test

Mục đích	Công cụ	Nhà cung cấp/Tự xây dựng	Phiên bản
Sử dụng để thực hiện automation testing	Katalon Studio	Nhà cung cấp	V9
Sử dụng để làm tool management để xuất report, test case, test suit	Katalon TestOps	Nhà cung cấp	V9
Chứa source code và run source code trên localhost	Docker	Nhà cung cấp	4.26.1

### 3.4 Môi trường test

- Giai đoạn Unit test:
  - Môi trường chạy: chạy trực tiếp trên code.
  - Database: Xây dựng database độc lập, đề cao thiết kế cơ sở dữ liệu nhỏ và nhẹ để tối ưu hiệu suất trong quá trình kiểm thử đơn vị.
  - Điều kiện mạng: Sử dụng mạng LAN để tạo điều kiện kiểm thử nhanh chóng và hiệu quả (độ trễ thấp).
- Giai đoạn Intergration test:
  - Môi trường chạy: chạy trên bản dịch chương trình để kiểm tra tính tương tác giữa các thành phần của hệ thống.
  - Database: Xây dựng database độc lập, đề cao thiết kế cơ sở dữ liệu phản ánh gần với môi trường thực tế và dữ liệu kiểm thử phức tạp.

- Điều kiện mạng: Sử dụng mạng LAN hoặc mô phỏng môi trường mạng trong thực tế.
- Giai đoạn System test:
  - Môi trường chạy: chạy trên bản dịch để đảm bảo tính toàn vẹn và tương thích của toàn bộ hệ thống.
  - Database: Sử dụng database độc lập, thiết kế phản ánh chính xác môi trường chạy thực tế, với dữ liệu lớn và đa dạng.
  - Điều kiện mạng: Mô phỏng môi trường mạng trong thực tế.
- Giai đoạn Acceptance test:
  - Môi trường chạy: chạy trên bản dịch để đảm bảo tính toàn vẹn và chấp nhận được cho người dùng cuối.
  - Database: Sử dụng database giống hệt với môi trường thực tế, dữ liệu thử nghiệm chân thực nhất.
  - Điều kiện mạng: Sử dụng môi trường mạng giống thực tế.

## 4 TÀI NGUYÊN

### 4.1 Nhân lực

Bảng sau mô tả nguồn lực test cho dự án.

Họ tên	Trách nhiệm/Ghi chú
Nguyễn Phước Hải	Test chức năng Create Expense Account Test chức năng Create Revenue Account
Hồ Sĩ Đức	Test chức năng Create Asset Account Test chức năng Create Recurring Transaction
Nguyễn Hải Đăng	Test chức năng Create Expense Transaction Test chức năng Create Piggy Bank Kiểm thử hiệu năng.
Nguyễn Văn Hiếu	Test chức năng Create Bill Test chức năng Create Transaction

## 4.2 Hệ thống

1. Phần cứng: 4 máy tính.
  - a. 1 máy cài Windows 10.
  - b. 1 máy cài Windows 11.
  - c. 1 máy cài Ubuntu 22.04.
2. Phần mềm:
  - a. Katalon Studio v9 – Katalon TestOps (bản quyền phía khoa).
  - b. Jmeter (để kiểm thử hiệu năng).
  - c. AI Heat Map (để kiểm thử GUI & Usability).

## 5 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
Tìm hiểu Project	All	25/12/2023	28/12/2023
Viết test plan	All	27/12/2023	28/12/2023
Kiểm thử chức năng, kiểm thử GUI & Usability: Create Expense Transaction, Create Piggy Bank	Đăng	29/12/2023	04/01/2024
Kiểm thử chức năng, kiểm thử GUI & Usability: Create Expense Account, Create Revenue Account	Hải	29/12/2023	04/01/2024
Kiểm thử chức năng, kiểm thử GUI & Usability: Create Asset Account, Create Recurring Transaction	Đức	29/12/2024	04/01/2024



Kiểm thử chức năng, kiểm thử GUI & Usability: Create Bill, Create Transaction	Hiếu	29/12/2024	04/01/2024
Kiểm thử hiệu năng	Đăng	05/01/2024	06/01/2024
Viết các báo cáo: Sinh dữ liệu, kiểm thử chức năng, kiểm thử hiệu năng, kiểm thử tự động	All	07/01/2024	09/01/2024

## 6 CÁC SẢN PHẨM

STT	Sản phẩm	Ngày bàn giao	Người bàn giao	Người nhận bàn giao
1	Test plan	28/12/2023	All	Đăng
2	Test Case	07/01/2024	All	Đăng
3	Bug Report	09/01/2024	All	Đăng
4	Video Test	09/01/2024	All	Đăng