

NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

TÀI LIỆU KIỂM THỬ

*Yêu cầu nhóm sinh viên hoàn thành
tài liệu kiểm thử cho đồ án đã được
giao theo biểu mẫu đính kèm.*



Bộ môn Công nghệ phần mềm
Khoa Công nghệ thông tin
Đại học Khoa học tự nhiên TP

HCM

MỤC LỤC

1	Thông tin nhóm.....	2
2	Kế hoạch kiểm thử (Test Plan).....	3
3	Test case.....	4
3.1	Danh sách các test case.....	4
3.2	Đặc tả các test case.....	4
3.2.1	Test case 1.....	4
3.2.2	Test case 2.....	4

TÀI LIỆU KIỂM THỬ

Tài liệu tập trung vào các chủ đề:

- ✓ Tạo ra tài liệu kiểm thử phần mềm.
- ✓ Hoàn chỉnh tài liệu kiểm thử phần mềm với các nội dung:
 - Kế hoạch kiểm thử (Test plan)
 - Các test case
- ✓ Đọc hiểu tài liệu kiểm thử phần mềm.

1 Thông tin nhóm

Website / **Facebook** **nhóm:**

<https://www.facebook.com/groups/678818555815551/>

MSSV	Họ Tên	Email	Điện thoại
1612001	Hoàng Dân An (team leader)	anaizhu98@gmail.com	0772 1088 66
1612043	Phan Ngọc Thanh Bình	binhphan2509@gmail.com	0162 8646 767
1612008	Lê Trần Lâm An	ghostkey.war@gmail.com	0397 3541 85
1612619	Nguyễn Bá Thắng	ka123kara@gmail.com	0162 7901 357

2 Kế hoạch kiểm thử (Test plan)

Dự định kiểm thử

1. Các yêu cầu chức năng thực hiện test

- **Giao diện:** Login, Home Page, thông số tính toán, giao diện, Phân quyền người dùng, ngôn ngữ và đồng bộ hóa
- **Code và backend bên dưới:** Quản lý Users, thống kê chi tiêu, báo cáo quản lý túi tiền, các chức năng phần mềm

2. Môi trường thực hiện test:

- **Device:** máy ảo Genymotion trên PC, phiên bản android 6.0 trở lên và API > 21, Android Studio 3.2

Công cụ kiểm thử

Lập báo cáo Test: MSWord

Công cụ hỗ trợ test giao diện: Genymotion Emulator

Framework hỗ trợ Unit Tests: Robolectric, JUnit 4

Chiến lược kiểm thử

1. Chiến lược xây dựng Test case:

- Quá trình kiểm thử được thực hiện sau khi hoàn thành mã nguồn, và giao diện cơ bản
- Toàn bộ phần giao diện sẽ xây dựng theo thống kê để kiểm tra vì yêu cầu không rõ ràng và có thể bị thay đổi nhiều
- Toàn bộ phần chức năng của Frontend và Backend đều có testcase
- Xây dựng testcase theo hình thức kiểm tra động dựa vào kịch bản kiểm thử kết hợp thanh tra phần mềm
- Các chức năng chính được cụ thể hóa theo kịch bản kiểm thử
- Nghiệm thu Alpha triển khai trong nội bộ nhóm chạy thử, sử dụng phần mềm

2. Test Browser và các thiết bị phân theo giai đoạn sau:

- Integration test: Chỉ test trên PC
- System test: Test trên PC và các thiết bị còn lại
- Acceptance test: các thành viên trong nhóm

3. Chiến lược về nguồn lực:

- Hoàng Dân An: Chịu trách nhiệm test chính phần code và logic bên dưới database
- Lê Trần Lâm An: Chịu trách nhiệm test chính phần Chức năng, Giao diện và lỗi
- Thực hiện review chéo kết quả sau từng round

4. Thứ tự ưu tiên test:

- Ưu tiên test giao diện hoàn thiện sớm để confirm với KH.
- Danh sách các chức năng ưu tiên test theo quá trình người dùng sẽ gặp

3 Test case

3.1 Danh sách các test case

[Liệt kê tên các test case, các đối tượng test và diễn giải]

STT	Tên test case	Đối tượng test	Nội dung
1	T001	đăng nhập	Nhập đúng, chọn đăng nhập
2	T002	đăng nhập	Không nhập gì, chọn đăng nhập
3	T003	đăng nhập	nhập mỗi tên tài khoản
4	T004	đăng nhập	nhập mỗi mật khẩu
5	T005	đăng nhập	nhập tên tài khoản có khoảng trắng (ký tự đặc biệt)
6	T006	đăng nhập	nhập mật khẩu sai
7	T007	đăng nhập	nhập mật khẩu sai quá 3 lần
8	T008	đăng nhập	nhập tên đăng nhập sai (tài khoản chưa đăng ký)
9	T009	đăng nhập	nhập tên đăng nhập sai quá 3 lần
10	T010	Quên mật khẩu, đăng ký	Không nhập gì, chọn xác nhận
11	T011	Quên mật khẩu, đăng ký	Nhập sai email, email không tồn tại

12	T012	Quên mật khẩu, đăng ký	Nhập đúng thông tin
13	T013	Thêm thu nhập	Nhập đúng kiểu dữ liệu
14	T014	Thêm thu nhập	Nhập sai kiểu dữ liệu
15	T015	Thêm chi tiêu	Nhập đúng kiểu dữ liệu
16	T016	Thêm chi tiêu	Nhập sai kiểu dữ liệu
17	T017	Thêm khoản ngân sách (kế hoạch thu chi)	Nhập đúng kiểu dữ liệu
18	T018	Thêm khoản ngân sách (kế hoạch thu chi)	Nhập sai kiểu dữ liệu
19	T019	Thêm và rút khoản tiết kiệm(sổ tiết kiệm)	Nhập đúng kiểu dữ liệu
20	T020	Thêm và rút khoản tiết kiệm(sổ tiết kiệm)	Nhập sai kiểu dữ liệu
21	T021	Xem các khoản thu chi theo ngày, số dư trong ví	Xem thông tin số tiền đã chi và còn lại
22	T022	Thu nhập, chi tiêu	Chỉnh sửa dữ liệu
23	T023	Phân loại thu chi	Chọn mục điều kiện phân loại
24	T024	Thống kê phân loại	Biểu đồ thống kê theo phân loại đã chọn
25	T025	Thống kê theo thời gian	Biểu đồ bar theo khoảng thời gian đã chọn
26	T026	Back up	Lưu dữ liệu lên firebase
27	T027	Thêm loại thu nhập, chi tiêu	Lựa chọn icon và tên phân loại mới
28	T028	Thông báo cho người dùng	Chọn/bỏ chọn chế độ nhắc nhở
29	T029	Xem thông tin phần mềm	Thông tin sơ lược ở mục about
30	T030	Xem tài khoản đang dùng	Xem thông tin email, tài khoản ở phần setting
31	T031	Xem hướng dẫn sử dụng	Xem ở phần

			setting
32	T032	Đăng nhập offline	Không sử dụng kết nối mạng

3.2 Đặc tả các test case

[Sinh viên chọn và trình bày đặc tả của vài (9-10) test case quan trọng nhất]

Test case 1

Test case	T008
Related Use case	U013
Context	Người dùng đăng nhập bằng tài khoản chưa xác thực, chưa được đăng ký
Input Data	Tên đăng nhập, mật khẩu của tài khoản
Expected Output	Đăng nhập không thành công, xuất hiện thông báo
Test steps	Mở app -> chọn đăng nhập -> điền thông tin đăng nhập -> chọn nút “đăng nhập”
Actual Output	Như dự kiến
Result	Passed

Test case 2

Test case	T011
Related Use case	U013
Context	Người dùng đăng ký tài khoản mới với email không hợp lệ
Input Data	Tên đăng nhập, mật khẩu của tài khoản
Expected Output	Đăng ký không thành công, xuất hiện thông báo
Test steps	Mở app -> chọn đăng ký -> điền thông tin tài khoản : +bỏ trống +không phải email +ký tự đặc biệt -> chọn nút “đăng ký”
Actual Output	Như dự kiến
Result	Passed

Test case 3

Test case	T013, T015
<i>Related Use case</i>	<i>U001</i>
<i>Context</i>	<i>Nhập liệu thu nhập và chi tiêu</i>
<i>Input Data</i>	<i>Số tiền thu, chi và nội dung cần ghi nhớ, chú thích về nguồn thu chi tương ứng</i>
<i>Expected Output</i>	<i>Danh sách thu, chi sẽ hiện ra tại màn hình home</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> chọn mục thêm thu chi (nằm kế bên nút home) -> chọn tab thu hoặc chi tương ứng -> điền thông tin thu chi : +Số tiền thu được, chi trả +chú thích thêm, thông tin về nguồn thu (văn bản) -> chọn nút “thêm ”</i>
<i>Actual Output</i>	<i>Như dự kiến</i>
<i>Result</i>	<i>Passed</i>

Test case 4

Test case	T019
<i>Related Use case</i>	<i>U007</i>
<i>Context</i>	<i>Sổ tiết kiệm</i>
<i>Input Data</i>	<i>Khoản tiền thu nhập người dùng muốn tiết kiệm</i>
<i>Expected Output</i>	<i>Nhập liệu thành công, thông tin khoản tiết kiệm được lưu vào danh sách tiết kiệm</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> chọn mục sổ tiết kiệm -> nhập khoản tiết kiệm : + số tiền + thông tin, nội dung cần ghi chú ->chọn nút “thêm”</i>
<i>Actual Output</i>	<i>Như dự kiến</i>
<i>Result</i>	<i>Passed</i>

Test case 5

Test case	T21
<i>Related Use case</i>	<i>U002, U005</i>
<i>Context</i>	<i>xem các mục thu, chi trong ngày và số dư trong ví</i>
<i>Input Data</i>	<i>Dữ liệu về thông tin thu chi trong cùng ngày</i>
<i>Expected Output</i>	<i>Danh sách thu, chi trong ngày tại màn hình home không bị thiếu sót hay sai lệch</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> xem thông tin tại màn hình home hoặc chọn nút home nếu như vẫn duy trì đăng nhập</i>
<i>Actual Output</i>	<i>Như dự kiến</i>
<i>Result</i>	<i>Passed</i>

Test case 6

Test case	T22
<i>Related Use case</i>	<i>U004</i>
<i>Context</i>	<i>Chỉnh sửa thông tin chi tiêu</i>
<i>Input Data</i>	<i>Dữ liệu về thông tin thu chi trong cùng ngày</i>
<i>Expected Output</i>	<i>Danh sách thu, chi được sửa đổi nội dung</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> vào lịch sử -> chọn mục cần chỉnh sửa -> chỉnh sửa nội dung -> chọn lưu lại</i>
<i>Actual Output</i>	<i>Như dự kiến</i>
<i>Result</i>	<i>Passed</i>

Test case 7

Test case	T023
<i>Related Use case</i>	<i>U008</i>
<i>Context</i>	<i>Phân loại thu chi</i>

<i>Input Data</i>	<i>Dữ liệu các khoản thu chi đã nhập liệu trong lịch sử tài khoản</i>
<i>Expected Output</i>	<i>Lọc dữ liệu các khoản thu, chi theo kiểu phân loại</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> chọn phân loại -> lựa chọn kiểu phân loại -> lưu lại</i>
<i>Actual Output</i>	<i>Tính năng chưa cập nhật</i>
<i>Result</i>	<i>Passed/Failed</i>

Test case 8

Test case	T025
<i>Related Use case</i>	<i>U010</i>
<i>Context</i>	<i>Thống kê thu chi theo thời gian</i>
<i>Input Data</i>	<i>Dữ liệu các khoản thu chi đã nhập liệu trong lịch sử tài khoản</i>
<i>Expected Output</i>	<i>Biểu đồ thống kê các khoản thu chi theo khoảng thời gian đã chọn</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> chọn thống kê -> chọn khoảng thời gian bắt đầu và kết thúc -> xác nhận</i>
<i>Actual Output</i>	<i>Tính năng chưa cập nhật</i>
<i>Result</i>	<i>Passed/Failed</i>

Test case 9

Test case	T026
<i>Related Use case</i>	<i>U011</i>
<i>Context</i>	<i>Back up dữ liệu</i>
<i>Input Data</i>	<i>Dữ liệu các khoản thu chi đã nhập liệu trong lịch sử tài khoản, tài khoản google để đồng bộ hóa</i>
<i>Expected Output</i>	<i>Dữ liệu được lưu lại để đồng bộ sau này</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> chọn back up</i>
<i>Actual Output</i>	<i>Như dự kiến</i>
<i>Result</i>	<i>Passed</i>

Test case 10

Test case	T027
<i>Related Use case</i>	<i>U012</i>
<i>Context</i>	<i>PIN lock</i>
<i>Input Data</i>	<i>Mã PIN để bảo mật thông tin khi truy cập bằng tài khoản</i>
<i>Expected Output</i>	<i>Khóa nội dung thông tin thu chi trong app, chỉ có thể xem sau khi nhập đúng mã PIN</i>
<i>Test steps</i>	<i>Mở app -> đăng nhập -> chọn PIN lock -> nhập mã PIN -> xác nhận</i>
<i>Actual Output</i>	<i>Tính năng chưa cập nhật</i>
<i>Result</i>	<i>Passed/Failed</i>

3.3 Unit tests

Unit Test là phần để test các hàm xử lý và thao tác với database bên dưới giao diện.

Các file test nằm ở thư mục:

04_Implement/02_Source/app/src/test/com/example/thang/smartmoney/

3.3.1 Công cụ và framework để test:

- Môi trường: Android Studio 3.2
- Framework test: Robolectric & JUnit 4

3.3.2 Kết quả các unit tests

3.3.3 Coverage reports

Các class thao tác với database có kết quả đúng gần hết

50% classes, 51% lines covered in package 'com.example.thang.smartmoney.data...'

Element	Class, %	Method, %	Line, %
Database	50% (1/2)	66% (8/12)	62% (40/64)
DBGiaoDich	100% (1/1)	91% (11/12)	90% (39/43)
DBNganSach	100% (1/1)	100% (12/12)	100% (66/66)
DBTietKiem	100% (1/1)	50% (5/10)	43% (17/39)
DBVi	100% (1/1)	37% (3/8)	25% (7/28)
FirebaseSync	0% (0/4)	0% (0/23)	0% (0/90)

Các class thao tác logic và cung cấp dữ liệu cho giao diện cũng đã được test unit nhưng chưa đủ hoàn toàn

80% classes, 59% lines covered in package 'com.example.thang.smartmoney.mod...'

Element	Class, %	Method, %	Line, %
ClassCategory	0% (0/2)	0% (0/9)	0% (0/48)
ClassExpense	100% (1/1)	66% (2/3)	85% (6/7)
ClassGiaoDich	100% (1/1)	83% (5/6)	94% (35/37)
ClassIcon	100% (1/1)	75% (3/4)	68% (31/45)
ClassIncome	100% (1/1)	33% (1/3)	28% (2/7)
ClassNganSach	100% (1/1)	81% (13/16)	82% (34/41)
ClassTietKiem	100% (2/2)	80% (4/5)	47% (11/23)
ClassVi	100% (1/1)	100% (2/2)	100% (12/12)