**PHÂN TÍCH VÀ QUẢN LÝ YÊU CẦU PHẦN MỀM**

**CHẤT LƯỢNG SẢN PHẨM**

C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

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 và chương trình 2](#_Toc467154099)

[2 Yêu cầu chất lượng sản phẩm 3](#_Toc467154100)

[2.1 Độ tin cậy: 3](#_Toc467154101)

[2.2 Tính sẵn sàng: 3](#_Toc467154102)

[2.3 Độ hiệu quả: 4](#_Toc467154103)

[2.4 Kích cỡ: 4](#_Toc467154104)

[2.5 Tốc độ xử lý: 5](#_Toc467154105)

[2.6 Khả năng bảo trì: 5](#_Toc467154106)

[2.7 Sự đơn giản trong giao diện: 6](#_Toc467154107)

[2.8 Yêu cầu về bảo mật: 6](#_Toc467154108)

[2.9 Thời gian đồng bộ: 6](#_Toc467154109)

[2.10 Khả năng xử lý lượng người dùng lớn: 7](#_Toc467154110)

**CHẤT LƯỢNG SẢN PHẨM**

# Các nội dung chính

Mục tiêu tài liệu tập trung vào các chủ đề:

* Tạo ra tài liệu về chất lượng phần mềm.
* Hoàn chỉnh tài liệu với các nội dung:
  + Danh sách một số yêu cầu chất lượng.
  + Đặt vấn đề, nêu lý do, hoàn cảnh và hướng xử lý cho từng yêu cầu.
* Đọc hiểu tài liệu chất lượng phần mềm.

# Thông tin nhóm và chương trình

**Chương trình**: Quản lý chi tiêu cho sinh viên.

Platform: Android 4.4 trở lên, PC (window 8.0 trở lên)

**Website / Facebook nhóm:** <https://www.facebook.com/groups/196824944025798/?fref=ts>

|  |  |  |  |
| --- | --- | --- | --- |
| **MSSV** | **Họ Tên** | **Email** | **Điện thoại** |
| **1412544** | **Phạm Đức Tiên** | **1412544@student.hcmus.edu.vn** |  |
| 1412427 | Nguyễn Minh Quang | 1412427@student.hcmus.edu.vn |  |
| 1412465 | Nguyễn Đình Sơn | [1412465@student.hcmus.edu.vn](mailto:1412465@student.hcmus.edu.vn) |  |
| 1412484 | Vũ Thành Tâm | 1412484@student.hcmus.edu.vn |  |
| 1412542 | Nguyễn Hà Tiên | 1412542@student.hcmus.edu.vn |  |
| 1412564 | Trần Thùy Bích Trâm | 1412564@student.hcmus.edu.vn |  |

# Yêu cầu chất lượng sản phẩm

#### Độ tin cậy:

* **Đặt vấn đề:** Độ đồng bộ dữ liệu giữa các thiết bị.
* **Lý do:** Người dùng (sinh viên) có thể sử dụng nhiều thiết bị khác nhau và chương trình hỗ trợ di động lẫn PC, nên vấn đề đồng bộ dữ liệu cho đồng nhất sẽ xảy ra.
* **Hoàn cảnh xảy ra vấn đề:**

Xét tình cảnh thật, Việt Nam không phải nơi nào cũng có wifi và không phải sinh viên nào cũng có 3G. Nên việc đồng bộ hóa mọi thứ lên cloud là không thể.

Ngày 15/11/2016 là lần đồng bộ cuối dữ liệu về danh sách tiền bạc thu chi trên cloud.

Sau 3 ngày đi chơi, người dùng đã nhập rất nhiều khoản chi tiêu vào trong điện thoại di động.

Ngày 19/11/2016 sau khi đi chơi về nhà, người dùng quên mất việc đồng bộ dữ liệu từ di động lên cloud. Sau khi bật máy tính và bật chương trình để coi lại mình đã chi tiêu những gì thì không thấy dữ liệu.

* **Hướng xử lý:**

+ Sau khi đăng nhập, máy tính sẽ gửi một request lên server để hỏi các thiết bị được liên kết đã đồng bộ hóa đến thời gian hiện tại chưa.

+ Nếu như các thiết bị không phản hồi thì sẽ đưa ra thông báo khoảng thời gian được đồng bộ gần đầy nhất để người dùng có thể biết được khoảng thời gian mình mới làm chưa được đồng bộ.

+ Nếu như các thiết bị có phản hồi thì sẽ cập nhật lại dữ liệu đến dữ liệu mới nhất cho người dùng.

#### Tính sẵn sàng:

* **Đặt vấn đề:** Có thể nhập số tiền đã thu, chi trong một khoảng thời gian ngắn và ít thao tác nhất có thể.
* **Lý do:** Đối với việc mua đồ rồi nhập tiền đã tiêu thì sẽ xảy ra vấn đề khi người dùng lấy các thiết bị ra để nhập. Sự khó khăn xảy đến khi đồ đang cầm quá cồng kềnh hoặc đang ở nơi đông người khó lấy thiết bị ra do sợ trộm cướp.
* **Hoàn cảnh xảy ra vấn đề:**

Khi đang đi mua đồ ở chợ, mua giày ngoài đường. Chúng ta sẽ không dám lấy điện thoại ra để nhập liệu do sợ bị giựt điện thoại hoặc đồ mua được quá nhiều và không dám để xuống đất sợ dơ, nên người dùng không có đủ tay để nhập liệu. Nếu không nhập và đi mua trong 1 thời gian dài, chúng ta sẽ không thể nào nhớ được những gì đã mua (với những người có trí nhớ kém).

* **Hướng xử lý:**

Thiết kế cho chương trình chạy background, khi người dùng cần nhập có thể kích hoạt bằng các tổ hợp phím nút hoặc kích hoạt bằng phím nhanh, nhập liệu bằng giọng nói thông qua tai nghe.

#### Độ hiệu quả:

* **Đặt vấn đề:** Thiết kế chương trình chạy không tốn quá nhiều dung lượng pin của sản phẩm.
* **Lý do:** Do yêu cầu của 2.2 nên chương trình phải chạy background, khi chạy background thì sẽ tốn pin. Đối với các điện thoai di động tầm trung thì dung lượng pin của các máy khá thấp (ví dụ đối với model Ausus Zenfone 4.5 thì thời gian gọi thoại sẽ chỉ từ 4~6 tiếng và chưa kể dùng thời gian dài các thiết bị còn bị chai pin,…)
* **Hoàn cảnh xảy ra vấn đề:**

Người dùng sử dụng các model di động cũ, bị chai pin. Nếu như trong 1 khoảng thời gian mà chương trình sử dụng hết 30% dung lượng pin thì sao?

* **Hướng xử lý:**

Chương trình được tạo ra sau khi chạy phải đảm bảo trong 1 khoảng thời gian thử nghiệm – 1 tiếng (không bật các chương trình khác, chỉ có chương trình chạy background) phải đảm bảm việc chiếm dung lượng pin từ 3~5%

#### Kích cỡ:

* **Đặt vấn đề:** Chương trình sẽ phải lưu nhiều dữ liệu, nếu kích cỡ quá lớn sẽ gây bất tiện cho người dùng nếu như bộ nhớ của điện thoại quá ít.
* **Lý do:** Do khi ta quản lý chi tiêu thì dù ít hay nhiều ta cũng cần phải xem xét dữ liệu chi tiêu trong vòng 2 năm đổ lại. Nếu ngày nào ta cũng nhập chi tiêu thì sau 2 năm nguồn dữ liệu hiện có sẽ rất lớn.
* **Hoàn cảnh xảy ra vấn đề:**

Người dùng sử dụng chương trình từ năm 2014 đến năm 2016 và họ lưu hết tất cả mọi thứ trên máy. Kích cỡ chương trình của họ sẽ là bao nhiêu?

* **Hướng xử lý:**

Quy tất cả dữ liệu về dạng text. Mỗi khoản thu hoặc chi sẽ được đưa lên 1 dòng dữ liệu.

#### Tốc độ xử lý:

* **Đặt vấn đề:** Một lúc nào đó người dùng sẽ phải thống kê tất cả các khoản chi tiêu lại và thời gian duyệt dữ liệu để tạo ra thống kê là một vấn đề.
* **Lý do:** Nhu cầu tạo ra các bảng sơ đồ thống kê là một chức năng chính của chương tình chi tiêu.
* **Hoàn cảnh xảy ra vấn đề:**

Người dùng đã dùng quản lý chi tiêu trong 3 năm và họ muốn tạo một bảng thống kê chi tiêu tổng 3 năm. Thời gian xử lý để tạo ra là bao lâu?

* **Hướng xử lý:**

Duyệt tuần tự, giới hạn tốc độ xử lý 10,000 dòng dữ liệu trở xuống trong vòng 2~3s.

#### Khả năng bảo trì:

* **Đặt vấn đề:** Sau một thời gian, chương trình cần được update tính năng hoặc bỏ đi các tính năng không cần thiết.
* **Lý do:** Do nhu cầu của thị trường và người dùng đòi hỏi sửa chữa và thêm các tính năng mới.
* **Hướng xử lý:**

+ Khi bảo trì server chứa dữ liệu để đồng bộ hóa online thì sẽ thông báo cho người dùng tạm thời đừng đồng bộ.

+ Khi tạo thêm chức năng mới đảm bảo cơ sở dữ liệu olnline vẫn đồng bộ hóa được với ít nhất 3 phiên bản trước đó.

#### Sự đơn giản trong giao diện:

* **Đặt vấn đề:** Giao diện quá rắc rối thì sẽ tốn thời gian của người dùng và khó tập thói quen.
* **Lý do:** Do nhu cầu sử dụng liên tục việc cập nhật thu chi, nếu như thiết kế giao diện quá rắc rối sẽ khiến cho người dùng gặp ức chế tâm lý khi sử dụng.
* **Hướng xử lý:**

+ Giao diện phải đảm bảo quy tắc màu sắc của thiết kế.

+ Giao diện của chương trình khi làm ra nên hiện sẵn những chức năng mà người dùng chắc chắn sử dụng và cho phép người dùng “cá nhân hóa” chương trình.

+ Tông màu chủ đạo là trắng và đen.

#### Yêu cầu về bảo mật:

* **Đặt vấn đề:** Bảo mật thông tin của người dùng.
* **Lý do:** Do quản lý chi tiêu ghi lại hầu hết các thu chi người dùng sử dụng, thông tin rơi vào tay người xấu sẽ khôn lường.
* **Hướng xử lý:**

+ Người dùng phải đăng nhập vào phần mềm mới được thực hiện các chức năng của phần mềm.

+ Mỗi người dùng có một tên đăng nhập và mật khẩu tương ứng với sổ thu chi của mình.

+ Sau khi đăng nhập người dùng có thể thiết lập mật khẩu cấp 2 để bảo vệ thêm an toàn cho tài khoản.

#### Thời gian đồng bộ:

* **Đặt vấn đề:** Thời gian đồng bộ giữa các thiết bị có lâu quá không?
* **Lý do:** Do quản lý chi tiêu hỗ trợ máy tính lẫn điện thoại và người dùng chắc chắn sẽ có nhu cầu đổi điện thoại hoặc luân chuyển thiết bị qua lại, nếu tốc độ đồng bộ dữ liệu quá lâu thì người dùng sẽ ngại đồng bộ hóa và thậm chí không muốn dùng chương trình.
* **Hướng xử lý:**

+ Thời gian đồng bộ dữ liệu lên cloud tối đa 3s với các bộ dữ liệu <2 năm.

+ Thời gian đồng bộ dữ liệu 5~10s cho các bộ dữ liệu 2~6 năm.

#### Khả năng xử lý lượng người dùng lớn:

* **Đặt vấn đề:** Một người thì server xử lý được vậy cả trăm người thì sao?
* **Lý do:** Đây là một vấn đề trong tương lai, nhưng nên nghĩ trước về khả năng chương trình được nhiều người sử dụng, nếu có quá nhiều người sử dụng và server không chịu nổi tải khi đồng bộ hóa thì sẽ mất rất nhiều người dùng.
* **Hướng xử lý:**

+ Chỉ lưu trữ dữ liệu dạng text trên server.

+ Khi cần đồng bộ file sẽ chỉ giao động không quá 20 MB (nếu lớn hơn thì cắt nhỏ ra theo trình tự thời gian gần nhất đến xa dần).