|  |  |
| --- | --- |
| **E:\LOP 12CTT3\HOC KY VII\Quan Ly Du An\templates\1140.jpg** | **ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  **KHOA CÔNG NGHỆ THÔNG TIN**  **MÔN QUẢN LÝ DỰ ÁN PHẦN MỀM** |

**Kế hoạch kiểm thử**

**DỊCH MỘT CÂU TỪ TIẾNG ANH SANG TIẾNG VIỆT**

**Phiên bản 2.0**

**NHÓM 6**

**TP HỒ CHÍ MINH 2015**

**Bản ghi nhận thay đổi phiên bản**

|  |  |  |  |
| --- | --- | --- | --- |
| **Ngày** | **Phiên bản** | **Mô tả** | **Tác giả** |
| **08/11/2015** | **1.0** | **Phiên bản đầu tiên** | **Trần Minh Vương, Nguyễn Trương Vương, Trần Văn Việt** |
| **09/12/2015** | **2.0** | **Phiên bản cập nhật** | **Trần Minh Vương.** |
|  |  |  |  |

MỤC LỤC

[1 GIỚI THIỆU 3](#_Toc437503220)

[1.1 Mục đích 3](#_Toc437503221)

[1.2 Thông tin chung 4](#_Toc437503222)

[1.3 Tài liệu liên quan 4](#_Toc437503223)

[1.4 Phạm vi test 4](#_Toc437503224)

[1.5 Ràng buộc 5](#_Toc437503225)

[1.6 Liệt kê các mạo hiểm 5](#_Toc437503226)

[2 CÁC YÊU CẦU CHO Test 5](#_Toc437503227)

[3 CHIẾN LƯỢC TEST 5](#_Toc437503228)

[3.1 Các kiểu test 6](#_Toc437503229)

[3.1.1 Test chức năng (Functional Testing) 6](#_Toc437503230)

[3.1.2 Test hiệu suất (Performance testing) 9](#_Toc437503231)

[3.2 Giai đoạn test 10](#_Toc437503232)

[3.3 Các công cụ test 10](#_Toc437503233)

[3.4 Môi trường test 11](#_Toc437503234)

[4 TÀI NGUYÊN 11](#_Toc437503235)

[4.1 Nhân lực 11](#_Toc437503236)

[4.2 Hệ thống 12](#_Toc437503237)

[5 CÁC MỐC KIỂM SOÁT CỦA GIAI ĐOẠN TEST (TEST MILESTONES) 12](#_Toc437503238)

[6 CÁC SẢN PHẨM 12](#_Toc437503239)

# GIỚI THIỆU

## Mục đích

* Tài liệu được biên soạn dùng để:
  + Xác định, đặc tả các module, thành phần dự án cần được kiểm thử dựa vào mục đích của phần mềm cũng như các chức năng đã được hiện thực trong phần mềm.
  + Liệt kê những yêu cầu kiểm thử.
  + Nêu ra những phương pháp, chiến lược cần sử dụng.
  + Xác định nguồn lực và phân công các module cụ thể cho từng thành viên cũng như lên kế hoạch từng bước cho từng cá nhân.
  + Đề ra những tiêu chuẩn có thể chấp nhận được để kết luận kết quả kiểm thử.
  + Liệt kê những kết quả, tài liệu thu được khi tiến hành kiểm thử.
* Xác định môi trường và test case sẽ thực hiện.
* Chỉ ra khối lượng test sẽ thực hiện.
* Giúp xác định được khi nào kết thúc giai đoạn test.

## Thông tin chung

* Mục đích kiểm thử phần mềm từ điển dịch một câu tiếng Anh sang tiếng Việt: EVTranslator nhằm mục đích tìm ra càng nhiều lỗi càng tốt để phục vụ cho việc phát triển phần mềm dựa trên yêu cầu đã mô tả:
* Phạm vi test bao gồm:
  + Test chức năng:
  + Test hiệu suất.

## Tài liệu liên quan

* Tài liệu đặc tả yêu cầu.
* Tài liệu đặc tả kiểm thử.
* Tài liệu kế hoạch quản lí cấu hình.
* Tài liệu proof of contept.
* Tài liệu thiết kế phần mềm.

## Phạm vi test

* Các giai đoạn kiểm thử được thực hiện:
  + Unit test: kiểm thử mức đơn vị.
    - Đảm bảo thông tin được xử lý và kết quả là chính xác giữa dữ liệu nhập và chức năng của phần mềm.
    - Kiểm tra từng đơn vị, thành phần nhỏ nhất của phần mềm bao gồm: các hàm, thủ tục, lớp, phương thức.
    - Việc thực hiện unit test sẽ tiết kiệm thời gian và chi phí cho việc kiểm thử và sửa lỗi ở các mức sau.
    - Unit test yêu cầu tester phải hiểu về thiết kế và code của chương trình.
  + Integration test: kiểm thử tích hợp.
    - Thực hiện trên những unit đã được kiểm tra cẩn thận bằng unit test và các lỗi ở mức unit đã được fix.
    - Integration test kết hợp các thành phần của ứng dụng và kiểm thử như một ứng dụng đã hoàn thiện.
    - Bao gồm 2 mục đích: phát hiện lỗi giao tiếp giữa các unit, tích hợp các thành phần thành hệ thống nhỏ và thành hệ thống hoàn chỉnh phục vụ kiểm thử ở mức hệ thống.

## Ràng buộc

* Phần mềm EVTranslator (dịch một câu tiếng Anh sang tiếng Việt) được xây dựng trên hệ điều hành android và dùng cơ sở dữ liệu là SQLite do đó phải có kiến thức về lập trình di động cũng như nắm rõ cách vận hành SQLite.
* Mọi thành viên trong nhóm phải hoàn thành đúng tiến độ trong test plan.
* Mọi vấn đề phát sinh phải báo cáo với nhóm trưởng để tìm giải pháp và phải báo cáo thường xuyên về vấn đề này.

## Liệt kê các mạo hiểm

* Thiếu nguồn nhân lực am hiểu về lập trình di động, các hình thức test (Unit test, Integration test, system test).
  + Các thành viên tự trau dồi kiến thức của bản thân
  + Tổ chức các buổi họp để các thành viên trao đổi và trau dồi kiến thức với nhau.
* Thiếu tài nguyên về phần cứng lẫn phần mềm.
  + Chọn lựa phần cứng, phần mềm đủ khả năng phục vụ cho quá trình test.
* Không hoàn thành đúng tiến độ.
  + Dùng phần mềm quản lí công việc.
  + Giám sát nhắc nhở các thành viên.
  + Bổ sung nhân lực nếu cần thiết.
* Mâu thuẫn giữa các thành viên xảy ra.
  + Tạo ra các buổi gặp mặt để giải quyết vấn đề.

# CÁC YÊU CẦU CHO Test

* Tìm hiểu kỹ về chương trình các chức năng và cách làm việc để hiểu sâu được ứng dụng.
* Nếu cần thiết thì tester cần phải tìm hiểu các công cụ hỗ trợ test để nâng cao độ chính xác của test cases.
* Thực hiện đầy đủ các test case cho dự án.
* Kiểm soát tốt các công việc tránh sự hỗn loạn giữa các thành viên.
* Khi gặp bug thì cần viết bug report, up lên github cho nhóm dev biết và fix bug.

# CHIẾN LƯỢC TEST

Các chiến lược :

* Assert : Là một macro dùng để phát hiện lỗi của phần mềm trong quá trình phát triển bằng cách thực thi biểu thức trong Assert. Dùng đẻ tìm kiếm các lỗi do người lập trình vô tình gây ra.(tham số không hợp lệ, sai thuật toán …)
* Mock Obiect : Là một đối tượng ảo mô phỏng các tính chất và hình vi giống hệt như đối tượng thật , được dùng để kiểm tra tính đúng đắng của một đơn vị chương trình.

Đơn giản hơn Đối tượng thực nhưng vẫn giữ được sự tương tác với các đối tượng khác.

+ Đảm bảo công việc không bị gián đoạn.

+ Giúp tiếp cận công việc tốt hơn.

+ Giúp phát hiện Interface cần tách ở một số lớp.

+ Đẽ dàng cho việc kiểm nghiệm.

* Test driven Development : Là một chiến lược phát triển sử dụng kĩ thuật UI theo nguyên tắc tạo ra các công đoạn kiểm nghiệm trước khi xây dựng mã.

Lợi ích : Định hình ý tưởng thiết kế, lập trình đôi,vận dụng tốt các phương pháp hướng đối tượng.

## Các kiểu test

### Test chức năng (Functional Testing)

#### Test chức năng (Function Testing)

Mục đích của test chức năng là tập trung vào các yêu cầu của phần mềm. 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 yêu cầu trong đặc tả. 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:

VD: Chức năng coppy nội dung đã dịch thành công, paste nội dung thành công sang văn bản bất kỳ.

|  |  |
| --- | --- |
| Mục đích test: | Kiểm tra các yêu cầu chức năng cho đáp ứng được đúng như trong đặc tả yêu cầu hay không. Các kết quả đầu ra cho được kết quả đúng như mong đợi. |
| Cách thực hiện: | Thực hiện mỗi UC, 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 theo mô tả usecase |
| Điều kiện hoàn thành: | * Toàn bộ kế hoạch test case đã được thực hiện. * Toàn bộ các lỗi phát hiện ra đã được ghi nhận. * Các test case có độ ưu tiên đều thực hiện cẩn thận. |
| Các vấn đề đặc biệt: | Có thể dừng test khi lập trình viên không thực hiện unit test.  Cần có dữ liệu test cơ bản |

#### 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.

VD: Font, size văn bản có dễ nhìn, dễ đọc không?

|  |  |
| --- | --- |
| Mục đích test: | Kiểm tra:   * Việc sử dụng thông qua mục tiêu test phản ánh đúng các chức năng và yêu cầu ứng dụng, bao gồm màn hình chính, màn hình dịch, … * 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 cần hoặc không cần thông qua giao diện người dùng để giao tiếp với dữ liệu. Ở phạm vi đồ án này, nhóm chỉ cho phép người dùng thêm dữ liệu câu đã dịch, chứ chưa thêm database.

|  |  |
| --- | --- |
| Mục đích test: | Đảm việc xuất dữ liệu và thêm dữ liệu chính xác. |
| Cách thực hiện: | Thực hiện các test cases. Đảm bảo toàn vẹn dữ liệu người dùng thêm vào.  Thực hiện các test cases trên cơ sở dữ liệu mà các chức năng chương trình có để xem kết quả.  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 chức năng xử lý đều giống như thiết kế và không có sai lệch dữ liệu. |
| Các vấn đề đặc biệt: | Các xử lý phải được thực hiện bằng tay. |

### Test hiệu suất (Performance testing)

#### Performance Profiling

Performance profiling là một dạng test hiệu suất trong đó thời gian phản hồi và các yêu cầu phụ thuộc thời gian khác được đo đạc và đánh giá. Mục đích của Performance Profiling là kiểm tra thời gian của các yêu cầu có đạt được hay không.

VD: Thời gian dịch trung bình của một câu là bao nhiêu? Tối đa thời gian cho phép.

|  |  |
| --- | --- |
| Mục đích test: | Kiểm tra thời gian dịch.  Kiểm tra thời gian load ứng dụng,… |
| Cách thực hiện: | Sử dụng các test cases.  Dùng công cụ đo thời gian thực hiện của chức năng. |
| Điều kiện hoàn thành: | Chức năng thực hiện thành công với thời gian cho phép, theo đặc tả yêu cầu. |
| Các vấn đề đặc biệt: | Vì app nhóm tự thiết kế nên thời gian không tốt như app hiện có là google tranlate. |

#### 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.

VD: Mở được bao nhiêu lần ứng dụng liên tục.

|  |  |
| --- | --- |
| Mục đích test: | Tìm kiếm giới hạn về word flow của ứng dụng như: app mở liên tục được bao nhiêu lần, bấm nút dịch bao nhiêu lần thì app bị crash. |
| Cách thực hiện: | Sử dụng các test đã xây dựng để thực hiện Stress Testing. |
| Điều kiện hoàn thành: | Các test case pass. ứng dụng bắt hết các lỗi không đáng có của ứng dụng như crash ứng dụng khi dịch. |
| 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. |

#### 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.

VD: câu dịch dài tối đa là bao nhiêu ký tự? Một lần dịch được bao nhiêu câu?

|  |  |
| --- | --- |
| Mục đích test: | Kiểm tra giới hạn của input nhập vào. |
| Cách thực hiện: | Thực hiện test cases đã xây dựng. |
| 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 theo các yêu cầu phi chức năng. |
| Các vấn đề đặc biệt: | Tuy theo trường hợp, môi trường mà giới hạn input khác nhau. |

## Giai đoạn test

| Kiểu test | Giai đoạn test | | | |
| --- | --- | --- | --- | --- |
| Unit | Integration | System | Acceptance |
| Test chức năng | X | X | X | X |
| Test hiệu suất. |  |  | X | X |

## 

## Các công cụ test

|  |  |  |  |
| --- | --- | --- | --- |
| Mục đích | Công cụ | Nhà cung cấp/Tự xây dựng | Phiên bản |
| Quản lý hoạt động kiểm thử | Android Studio | Google | 1.4 |
| Kiểm thử trên máy thật | ASUS Zenfone | ASUS | 4.5 |
| Báo cáo lỗi | Microsoft Word | Microsoft | 2013 |
| Test cases | Microsoft Excel | Microsoft | 2013 |

## Môi trường test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Giai đoạn test | Môi trường | Công cụ | Database | Cấu hình |
| Unit test | Code | Android studio 1.4 | Cùng với database phát triển. |  |
| Intergration test | Code | Android studio 1.4 | Cùng với database phát triển. |  |
| System test | Code | Android studio 1.4 | Cùng với database phát triển.. |  |
| Acceptance test | Máy thật | ASUS 4.5 | Database riêng. | Android 5.0  Ram 1Gb.  Memory 4 Gb. |

# TÀI NGUYÊN

## Nhân lực

Bảng sau mô tả nguồn lực test cho dự án.

|  |  |
| --- | --- |
| Họ tên | Trách nhiệm/Ghi chú |
| Trần Minh Vương | Test Manager |
| Trần Văn Việt | Tester |
| Nguyễn Trương Vương | Tester |
| Trần Hữu Vũ | Tester |

## Hệ thống

* Môi trường phần cứng cần thiết:
  + Thiết bị android hoặc thiết bị máy tính cài máy ảo.
* Hệ thống phần mềm cần thiết:
  + IDE để lập trình là Android studio 1.4.
* Công cụ hỗ trợ cho quá trình test
  + Công cụ quản lí dự án: Microsoft Offiec, facebook, …

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

|  |  |  |  |
| --- | --- | --- | --- |
| Milestone Task | Nguồn lực | Ngày bắt đầu | Ngày kết thúc |
| Lập kế hoạch kiểm thử | M.Vương, T.Vương, Việt. | 22/102015 | 12/11/2015 |
| Tiến hành kiểm thử | M.Vương, T.Vương, Việt, Vũ | 03/12/2015 | 10/12/2015 |
| Báo cáo kết quả test | M.Vương | 09/12/2015 | 10/12/2015 |

# CÁC SẢN PHẨM

| STT | Sản phẩm | Ngày bàn giao | Người bàn giao | Người nhận bàn giao |
| --- | --- | --- | --- | --- |
| 1 | Test plan | 12/11/2015 | T.M.Vương | T.T.T.Tùng |
| 2 | Test cases | 19/11/2015 | T.M.Vương | T.T.T.Tùng |
| 3 | Test report | 09/12/2015 | T.M.Vương | T.T.T.Tùng |
| 4 | Bug report | 09/12/2015 | T.M.Vương | T.T.T.Tùng |