**Kế Hoạch Phát Triển**

**Phần Mềm**

**cho**

**Website Thông tin nhà trọ**

Phiên bản 1.0 được phê chuẩn

**Được chuẩn bị bởi**

Lương Đức Duy 111280

Ngô Minh Phương 1111327

Nguyễn Hoàng Đông 1111282

Đại học Cần Thơ

18/08/2014

# Mục lục

[Mục lục 2](#_Toc396152716)

[Theo dõi phiên bản tài liệu 3](#_Toc396152717)

[1. Tổ chức nhóm phát triển phần mềm 4](#_Toc396152718)

[2. Quản lý chất lượng phần mềm 4](#_Toc396152719)

[3. Quản lý rủi ro 4](#_Toc396152720)

[4. Quản lý cấu hình 4](#_Toc396152721)

# Theo dõi phiên bản tài liệu

|  |  |  |  |
| --- | --- | --- | --- |
| Tên | Ngày | Lý do thay đổi | Phiên bản |
| Nguyễn Hoàng Đông | 19/08/2014 | Cập nhật phần Tổ chức nhóm phát triển và phần quản lý rủi ro. | 1.0 |
| Lương Đức Duy | 19/08/2014 | Cập nhật phần Quản lý chất lượng phần mềm | 1.1 |

# Tổ chức nhóm phát triển phần mềm

**Cơ cấu tổ chức**

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Họ và tên | Chức vụ | Vai trò |
| 1 | Lương Đức Duy | Nhóm trưởng | - Phân công công việc, lịch họp cũng như xúc tiến các thành viên khác trong nhóm hoàn thành đúng tiến độ công việc.  -Thiết kế viên, lập trình viên, viết tài liệu. |
| 2 | Ngô Minh Phương | Thành viên | -Thiết kế viên, lập trình viên, viết tài liệu. |
| 3 | Nguyễn Hoàng Đông | Thành viên | -Thiết kế viên, lập trình viên, viết tài liệu. |

**Giao tiếp**

Nhóm liên lạc với nhau, với khách hàng, người sử dụng, người quản lý bằng các phương tiện như: email, điện thoại, Trello và các buổi làm việc trực tiếp.

# Quản lý chất lượng phần mềm

## Tổ chức nhóm quản lý chất lượng

|  |  |  |
| --- | --- | --- |
| Họ và tên | Chức vụ | Vai trò |
| Lương Đức Duy | Trưởng nhóm | Quản lý chung, đảm bảo chất lượng thiết kế kiến trúc. |
| Nguyễn Hoàng Đông | Thành viên | Đảm bảo chất lượng tài liệu. |
| Ngô Minh Phương | Thành viên | Đảm bảo chất lượng code, kiểm thử. |

## Đề xuất các thủ tục, chuẩn, quy định về chất lượng

|  |  |
| --- | --- |
| Thành phần | Quy ước |
| Tài liệu | Quy tắc đặt tên: NL04\_TenTaiLieu\_vx.x với x.x là số phiên bản, nếu có tên người tạo file thì đặt thêm hậu tố là tên người, ví du: NL04\_TenTaiLieu\_TenTacGiaVietLienKhongDau\_vx.x.  Font chữ: Arial.  Kích thước: 12px.  Màu chữ: đen.  Các bảng phải được chỉnh cho lấp đầy phần ngang không gian.  Các mục heading phải được đánh số, viết thường, hoa đầu dòng.  Cấu trúc tài liệu:   * Trang cover: Ghi tên tài liệu, tên dự án, người chuẩn bị tài liệu, người phê chuẩn, ngày chuẩn bị, số phiên bản. Font-size: 14 – 32, màu chữ: đen. * Trang nội dung:   + Trang mục lục: chữ mục lục được định dạng heading nhưng không đánh số thứ tự. Số trang được đánh bằng số la mã, viết thường.   + Trang theo dõi phiên bản tài liệu: ghi rõ ngày sửa đổi, người sửa đổi, lý do, phiên bản. Tiêu đề được định dạng heading nhưng không đánh số thứ tự. Số trang được đánh bằng số la mã, viết thường.   + Trang nội dung khác: Các đề mục được format ở dạng heading. Đánh số theo số Ả rập: 1, 2, 3…. Các mục con nhỏ hơn được đánh số theo dạng X.X. Ví dụ : 1.1, 1.2, 1.3. Số trang được đánh số bắt đầu từ 1, được đánh số Ả-rập.   Nếu là file excel thì các trang được đổi thành các sheet với nội dung và định dạng tương tự. |
| Các quy ước thiết kế | Thiết kế theo mô hình MVC   * Các file Model: đặt tên theo quy tắc đặt tên lớp. * Các file View: đặt tên theo chức năng hiển thị, ngắn gọn, theo quy tắc Camel, ví dụ: AddAgreement, ListStudent, … * Các file Controller: đặt tên theo chức năng mà nó quản lý, ngắn gọn, theo quy tắc Camel, có hậu tố là Controller.Có thể đặt tên theo tên lớp mà nó quản lý, viết ở dạng số nhiều, ví dụ: StudentsController, RegisterCoursesController, v.v |
| Các quy ước lập trình | * Tên biến: tên biến được đặt ngắn gọn, thể hiện được mục đích khi sử dụng biến, thường là danh từ, viết theo quy tắc Camel, chữ đầu tiên viết thường. Ví dụ: numberOfStudents, isVisitted,…. Có thể sử dụng tiền tố để chỉ ra kiểu của biến, ví dụ: strName, intNumberOfWays, …. * Tên hàm: tên hàm bắt buộc phải là động từ, ngắn gọn, thể hiện mục đích của hàm, viết theo quy tắc Camel, chữ đầu tiên viết thường, ví dụ: createAgreement, editAgreement, solveEquation, updateStudentInformation, …. * Tên lớp: bắt buộc là danh từ, viết theo quy tắc Camel, ngắn gọn, thể hiện đúng bản chất của lớp, viết dưới dạng số ít, ví dụ: Student, Home, Factory, … * Các dòng phải được đánh indent phù hợp để thể hiện rõ cấu trúc chương trình. |
| Các quy ước về chú thích | * Khối chú thích:   /\*  \* Here is a block comment.  \*/  Khối chú thích được sử dụng để mô tả thông tin files, phương thức, cấu trúc dữ liệu và giải thuật. Khối chú thích phải được đặt ở đầu file, đầu các phương thức, cấu trúc dữ liệu, giải thuật, …   * Chú thích dòng:   if (condition) {  // Handle the condition.  ...  }  Chú thích dòng được sử dụng để phân mức cho code, trước chú thích dòng nên là 1 khoảng trắng. Nếu số lượng chữ quá dài để viết chú thích dòng thì nên sử dụng khối chú thích.   * Chú thích theo sau:   if (a == 2) {  return TRUE; // special case  } else {  return isPrime(a); // works only for odd a  }  Chú thích này được sử dụng để chú thích cho các dòng code, phải viết ngắn gọn, khoảng cách đủ xa để phân biệt với đoạn code.   * Doc comment:   /\*\*  \* The Example class provides ...  \*/  public class Example {  ...  Chú thích dạng này dùng để chú thích lớp, giao diện, phương thức, được đặt trong /\*\*…\*/ Ghi rõ chức năng của lớp, đầu vào, đầu ra, nội dung xử lý của phương thức, v.v |
| Chuẩn kiểm thử | * Kiểm thử theo mô hình chữ V, qua 4 giai đoạn. |

## Kiểm soát chất lượng

|  |  |
| --- | --- |
| Tài liệu | Yêu cầu |
| Software Requirement Specification | * Mô tả hệ thống. Có thể có sơ đồ chức năng của tổ chức. * Sơ đồ usecase. * Đặc tả usecase: Tên usecase, chức năng usecase, người sử dụng, tiền điều kiện, yêu cầu chức năng. * Yêu cầu phi chức năng. |
| High Level Design | * Sơ đồ usecase: phải thể hiện được chức năng của hệ thống, phân rã chi tiết đủ để thể hiện được các chức năng. * Sơ đồ dòng dữ liệu: thể hiện được các luồng dữ liệu vào ra trong hệ thống, phân rã vừa đủ để thể hiện rõ ràng được các chức năng * Cơ sở dữ liệu phải được vẽ bằng hình ảnh, thể hiện được các cột, trường, kiểu dữ liệu, mối quan hệ, ràng buộc giữa các bảng. |
| Detail Design Document | * Sơ đồ lớp và sơ đồ trình tự phải được vẽ bằng hình ảnh. * Sơ đồ lớp: Các lớp phải được chia thành các pakage. * Sơ đồ trình tự phải thể hiện ít nhất 1 trường hợp cho mỗi chức năng trong hệ thống. |
| Testing document | * Các trường hợp kiểm thử, yêu cầu các trường hợp kiểm thử phải có độ bao phủ trên 90%. |

## Công cụ, kỹ thuật và phương pháp đảm bảo chất lượng

Sử dụng các công cụ:

* CheckStyle: dùng để kiểm tra nội dung code có đạt chuẩn quy tắc code.
* FindBugs: dùng để tìm các bug trong file Java như bắt các trường hợp có thể sinh ra exception.
* Junit: kiểm thử các hàm trong code.
* Quick Test Pro: kiểm thử tích hợp, giao diện, v.v

# Quản lý rủi ro

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Yếu tố rủi ro | Mức độ rủi ro | Chiến lược làm giảm rủi ro | Hướng giải quyết khi xảy ra rủi ro |
| Nhóm yêu tố rủi ro liên quan đến khách hàng và người sử dụng | | | |  |
| 1 | Khách hàng thay đổi yêu cầu, chức năng của sản phẩm. | Cao | Phỏng vấn cẩn thận từng yêu cầu của khách hàng đối với hệ thống. Đưa ra các gợi ý về những chức năng cần thiết cho khách hàng chọn. | Họp nhóm phân tích lại yêu cầu của khách hàng để hoàn thiện phần mềm. |
| 2 | Người dùng không hiểu rõ về hệ thống cũng như các chức năng của nó. | Trung bình | Cần có mục hướng dẫn để giúp khách hàng dễ dàng trong việc sử dụng. | Tổ chức buổi hướng dẫn khách hàng sử dụng chương trình. |
| Nhóm yếu tố liên quan đến phạm vi và các yêu cầu. | | | |  |
| 1 | Thay đổi, mở rộng phạm vi dự án. | Thấp | Phỏng vấn kỹ lưỡng về quy mô và nhu cầu mở rộng quy mô của khách hàng. | Họp nhóm, dự trù lại trinh phí, phân tích lại yêu cầu để hoàn thiện tài liệu. |
| 2 | Yêu cầu không rõ ràng gây ra sự hiểu sai. | Trung bình | Phỏng vấn khách hàng một cách tỉ mỉ về các yêu cầu của họ để tránh hiểu sai ý khách hàng. | Thay đổi ngay và lập tức các yêu cầu sai lệch. Thiết kế lại chức năng tương ứng, kiểm tra xem nó có phù hợp với các chức năng khác hay không? |
| Nhóm yếu tố liên quan đến sự thực hiện. | | | |  |
| 1 | Không đồng nhất trong việc đặt tên biến cũng như tên hàm… | Trung bình | Cần có quy ước đặt tên hàm, biến do nhóm quy định(có thể tham khảo các chuẩn đặt tên ….) | Cần tập trung nhóm lại thay đổi và thống nhất triệt để các vấn đề trong việc lập trình. |
| 2 | Thiếu kinh phí | Trung bình | Cần dự trù kinh phí và chi phí phát sinh trong quá trình phát triển tránh trường hợp vượt kinh phí dự kiến. | Vận động kinh phí từ phía các đối tác. |
| 3 | Chậm trễ trong tiến độ do vấn đề cá nhân( về quê, bệnh…..) | Thấp | Các thành viên ở xa có thể làm việc bằng hình thức online. | Nhóm trưởng xúc tiến hoặc phân chia lại công việc nhầm tăng tiến độ các thành viên để kế hoạch hoàn thành đúng thời hạn. |
| 4 | Phản ứng chậm trể các vấn đề đã thông báo. | Trung bình | Thông tin phản hồi phải được thông báo trong một thời gian nhất định kể từ khi thông báo. | Liên hệ trực tiếp đến thành viên đó để thông báo và phê bình. |
| Nhóm yếu tố liên quan đến môi trường. | | | |  |
| 1 | Các thành viên ở xa dẫn đến không thuận lợi trong việc đi lại cũng như họp nhóm làm ảnh hưởng đến tiến độ công việc. | Thấp | Cần lên kế hoạch và nội dung họp nhóm một cách cụ thể. Các thành viên trong nhóm cần phải chủ động thời gian. | Họp nhóm phân chia lại công việc đáp ứng tiến độ của kế hoạch. |

# Quản lý cấu hình