

TRƯỜNG ĐẠI HỌC CẦN THƠ
KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
BỘ MÔN CÔNG NGHỆ PHẦN MỀM



NIÊN LUẬN
NGÀNH KỸ THUẬT PHẦN MỀM
KẾ HOẠCH PHÁT TRIỂN

CTU PC SHOP
CAN THO UNIVERSITY PC SHOP ONLINE

Phiên bản 0.5

Giảng viên hướng dẫn

Ths. Võ Huỳnh Trâm

Sinh viên thực hiện

Đào Minh Trung Thuận – B1704855 – K43

Nguyễn Quốc Hưng – B1704735 – K43

Học kỳ 1, năm học 2020 – 2021

THEO DÕI PHIÊN BẢN TÀI LIỆU

Tên	Ngày	Lý do thay đổi	Phiên bản
Đào Minh Trung Thuận Nguyễn Quốc Hưng	24-09-2020	Viết kế hoạch phát triển	0.1
Đào Minh Trung Thuận	25-09-2020	Bổ sung nội dung quản lý chất lượng (còn thiếu mục các chuẩn và thực tiễn kiểm thử) Thêm mục quản lý chất lượng, mục kiểm soát quản lý (còn thiếu mục kế hoạch kiểm thử, tài liệu kiểm thử, tài liệu người dùng, tài liệu báo cáo)	0.2
Nguyễn Quốc Hưng	26-09-2020	Thêm mục quản lý chất lượng, mục kế hoạch kiểm thử, tài liệu kiểm thử, tài liệu người dùng, tài liệu báo cáo.	0.3
Đào Minh Trung Thuận	27-09-2020	Hoàn thành các mục còn thiếu, sửa lỗi chính tả, định dạng	0.5

MỤC LỤC

1	TỔ CHỨC NHÓM PHÁT TRIỂN PHẦN MỀM	3
2	QUẢN LÝ CHẤT LƯỢNG PHẦN MỀM.....	3
2.1	TỔ CHỨC NHÓM QUẢN LÝ CHẤT LƯỢNG.....	3
2.2	CÁC THỦ TỤC, CHUẨN, QUY ĐỊNH VỀ CHẤT LƯỢNG.....	4
2.2.1	Các chuẩn, quy ước về tài liệu.....	4
2.2.2	Chuẩn thiết kế:	5
2.2.3	Các chuẩn và quy ước trong lập trình	5
2.3	KIỂM SOÁT CHẤT LƯỢNG	8
2.3.1	Kế hoạch phát triển.....	8
2.3.2	Tài liệu đặc tả yêu cầu.....	9
2.3.3	Tài liệu thiết kế	9
2.3.4	Kế hoạch kiểm thử.....	9
2.3.5	Tài liệu kiểm thử.....	10
2.3.6	Tài liệu người dùng	11
2.3.7	Tài liệu báo cáo.....	11
2.4	QUẢN LÝ RỦI RO	11
3	QUẢN LÝ CẤU HÌNH	13
3.1	TỔ CHỨC QUẢN LÝ CẤU HÌNH	13
3.2	NHẬN DẠNG CẤU HÌNH.....	13
3.3	QUẢN LÝ PHÁT HÀNH VÀ PHÂN PHỐI	14
3.4	TÀI NGUYÊN QUẢN LÝ CẤU HÌNH.....	15

1 TỔ CHỨC NHÓM PHÁT TRIỂN PHẦN MỀM

Tổ chức nhóm:

Họ và tên	Mã số sinh viên	Vai trò
Đào Minh Trung Thuận	B1704855	Trưởng nhóm
Nguyễn Quốc Hưng	B1704735	Thành viên

Trưởng nhóm có vai trò trong việc quản lý thành viên, phân công và kiểm duyệt công việc thành viên trong nhóm. Trưởng nhóm chịu trách nhiệm, quản lý đảm bảo chất lượng phần mềm. Giữ liên lạc, báo cáo tiến độ dự án định kỳ cho giảng viên hướng dẫn.

Tất cả thành viên nhóm có vai trò đóng góp xây dựng phần mềm, bao gồm: lên ý tưởng, phát triển phần mềm, kiểm thử, quản lý tài nguyên và môi trường phát triển. Mỗi thành viên có trách nhiệm trong việc đảm bảo hoàn thành đúng tiến độ công việc mà trưởng nhóm phân công.

Thành viên nhóm giao tiếp với nhau thông qua hai hình thức:

- Ngoại tuyến: Chiều 13 giờ 00 thứ năm mỗi tuần ở sảnh khoa Công nghệ thông tin và Truyền thông.
- Trực tuyến: Giao tiếp online thông qua nhóm Zalo, họp nhóm và hướng dẫn phân công, trình bày công việc qua Google Meet. Trưởng nhóm liên lạc với giảng viên hướng dẫn thông qua Gmail.

Mã nguồn dự án được lưu trữ trên Github: github.com/daomtthuan/ctu-pc-shop.

2 QUẢN LÝ CHẤT LƯỢNG PHẦN MỀM

2.1 TỔ CHỨC NHÓM QUẢN LÝ CHẤT LƯỢNG

Trưởng nhóm có vai trò đảm bảo, kiểm duyệt mã nguồn phần mềm, phê duyệt các công việc đã hoàn thành của các thành viên trong nhóm. Đảm bảo, quản lý và chịu trách nhiệm chung các tài sản, môi trường, mã nguồn của dự án.

Tất cả thành viên chịu trách nhiệm quản lý, hoàn thành các công việc, tài sản, mã nguồn đã được phân công. Ngoài ra, đóng góp sửa lỗi chung cho mã nguồn dự án.

2.2 CÁC THỦ TỤC, CHUẨN, QUY ĐỊNH VỀ CHẤT LƯỢNG

2.2.1 Các chuẩn, quy ước về tài liệu

Tất cả tài liệu được lưu trong thư mục docs của dự án, bên trong gồm các thư mục con:

- Thư mục teacher: lưu các tài liệu liên quan đến mẫu, yêu cầu của giảng viên hướng dẫn.
- Thư mục work: lưu các tài liệu của nhóm trong quá trình thực hiện quản lý, phát triển dự án.

Cách đặt tên tài liệu: viết tên tài liệu không dấu, không khoảng cách, viết hoa mỗi từ, có thể chú thích bằng dấu gạch dưới.

VD: `SoftwareDevelopmentPlan_KeHoachPhatTrieuPhanMem.docx`

Thành phần một tài liệu phải bao gồm:

- Trang bìa gồm có tên trường, khoa, logo trường, tên tài liệu, tên dự án, phiên bản, tên giảng viên hướng dẫn, tên các thành viên, thời gian được viết của tài liệu.
- Trang quản lý từng phiên bản của tài liệu gồm có tên người thực hiện, ngày sửa đổi, nội dung sửa và phiên bản.
- Tài liệu phải có mục lục trang, mục lục hình, bảng, sơ đồ... (nếu có).

Cấu trúc chuẩn tài liệu:

- Tất cả tài liệu được viết bằng phần mềm Microsoft Word hiện hành với đuôi tệp (phần mở rộng tệp) là .docx.
- Sử dụng bảng mã tiếng Việt Unicode UTF-8 để soạn thảo.
- Tài liệu được viết bắt buộc không có lỗi chính tả và sai ngữ pháp.

Trình bày tài liệu:

- Tiêu đề phân/chương (Heading 1): font Arial, Size 14, Bold, chữ hoa.
- Tiêu đề mục la mã (Heading 2): font Arial, Size 13, Bold, chữ hoa.
- Tiêu đề mục nhỏ (Heading 3): font Arial, Size 13, Bold, chữ thường.
- Nội dung (normal): font Times New Roman, size 13, chữ thường.
- Nội dung mã nguồn (Code): font Consolas, size 11.
- Sử dụng format/style để định nghĩa các style trên.
- Mục lục trình bày đến 3 cấp (heading 1, 2, 3). Sử dụng Insert/Index and Table/ Table of contents để làm mục lục tự động.

- Quy cách đánh số chương mục theo kiểu Outline.
- Các thành phần tài liệu phải được ngắt Section.
- Mỗi trang phải được đánh số trang và ghi tên dự án và tên tài liệu.
- Nội dung canh lề đều hai bên, với nội dung trong bảng: canh lề phải với nội dụng số và canh đều với nội dụng chữ, tiêu đề cột canh giữa in đậm.

2.2.2 Chuẩn thiết kế:

Tất cả các mẫu thiết kế được lưu trong thư mục design của dự án:

- Thiết kế tài sản (assets) cho dự án bằng phần mềm Adobe Illustrator và Adobe Photoshop. Hình ảnh được lưu dưới dạng .png.
- Thiết kế mô hình dự án bằng các chuẩn mô hình như UML, DFD, FlowChart... Sử dụng phần mềm Draw.io để thiết kế. Mô hình được xuất ra dưới định dạng hình ảnh với phần mở rộng .png.

2.2.3 Các chuẩn và quy ước trong lập trình

2.2.3.1 Quy ước tổ chức thư mục

Mã nguồn để đặt trong thư mục src của thư mục dự án, được phân làm hai thành phần con:

- Thư mục client: chứa mã nguồn cho phía client.
- Thư mục server: chứa mã nguồn web service của phía server.

Trong thư mục src có file .env chứa biến môi trường, cấu hình của dự án.

2.2.3.2 Quy ước đặt tên:

Đặt tên biến, tên hàm với tên có ý nghĩa (tên biến thường là danh từ, tên hàm thường là động từ), kiểu đặt tên theo kiểu camelCase (các từ nối liền, không ký tự nối, từ đầu viết thường chữ cái đầu, các từ còn lại viết hoa chữ cái đầu).

VD: Với TypeScript hoặc JavaScript:

```
let numberProducts: Number = 1;
function doSomethings(param: Number) {
    // Do somethings
}
```

VD: Với PHP:

```
$numberProducts = 1;
function doSomething(string $var = null)
{
```

```
# code...  
}
```

Hằng số, biến môi trường, biến cấu hình được đặt tên có ý nghĩa (thường là danh từ) và đặt tên theo kiểu SNAKE_CASE_ALL_CAPS (tất các từ được in hoa và đối với nhau bằng dấu gạch dưới). Riêng các hằng định nghĩa giá trị cho hệ thống thì tên bắt đầu và kết thúc bằng hai dấu gạch dưới.

VD: Với TypeScript hoặc JavaScript:

```
const BASE_URL = '/';
```

VD: Với PHP

```
define('__ROOT__', __DIR__);
```

VD: Với biến môi trường

```
DATABASE_DNS = 'mysql:host=localhost:3306;dbname=ctu_cp_shop'
```

Đặt tên không gian có ý nghĩa, mang hàm ý chỉ sự bao đóng tập hợp và được đặt theo kiểu 'PascalCase' (không có ký tự nối, các từ viết liền nhau và được viết hoa chữ cái đầu). Tên NameSpace phải trùng với tên thư mục.

VD: Controller, Model

Đặt tên thư mục trùng với tên NameSpace, tên thư mục đặt theo kiểu kebab-case (các từ viết thường được nối với nhau bằng dấu gạch nối).

VD: core, application

Đặt tên lớp trùng với tên tệp, tên lớp có ý nghĩa, thường là danh từ và đặt theo kiểu PascalCase.

VD: Student, Product, SystemException

Khi lập trình, nếu sử dụng câu lệnh rẽ nhánh (if ... else ...), thì nên ưu tiên các trường hợp thoát hàm, thoát chương trình lên đầu kiểm tra trước.

Tên của lớp Model phải trùng với tên bảng trong cơ sở dữ liệu, tên trường của lớp phải trùng với tên trường của bảng.

Khi lập trình với TypeScript, JavaScript hạn chế khai báo biến toàn cục var, thay vào đó sử dụng kiểu khai báo biến cục bộ let.

2.2.3.3 Quy ước về cách viết chú thích:

Nên viết chú thích ở những đoạn mã nguồn mà bản thân cảm thấy sẽ gây mơ hồ và khó hiểu cho các lập trình viên khác nhưng hạn chế viết chú thích tràn lan, không cần thiết gây khó khăn trong việc bảo trì, phát triển.

Viết chú thích cho các thuộc tính và hàm public bằng kiểu comment doc (/** */).

VD: TypeScript, JavaScript

```
/** User */
class User {
  /** User Id */
  private id: string;

  /**
   * Create new User instance
   * @param {string} id User Id
   */
  constructor(id: string) {
    this.id = id;
  }

  /**
   * Do somethings
   * @return {number} Result
   */
  public doSomethings(): number {
    return 1;
  }
}
```

VD: Với PHP:

```
/** Student */
class Student {
  /** Student name */
  private string $name;

  /**
   * Create new Student
   * @param string $name Student name
   */
  public function __construct(string $name) {
```

```
        $this->name = $name;
    }

    /**
     * Get Student name
     * @return string Student name
     */
    public function getName(): string {
        return $this->name;
    }
}
```

Thành phần chú thích của một hàm gồm có:

- Nội dung, mô tả hoạt động
- Đối số, tham số đầu vào (nếu có) bắt đầu bằng @param gồm kiểu, tên tham số, mô tả tham số
- Giá trị trả về (nếu có) bắt đầu bằng @return gồm kiểu, mô tả giá trị trả về
- Ném lỗi (nếu có) bắt đầu bằng @throws gồm kiểu, mô tả lỗi được ném ra

2.2.3.4 Các chuẩn và thực tiễn kiểm thử

Viet gi do o day

2.3 KIỂM SOÁT CHẤT LƯỢNG

2.3.1 Kế hoạch phát triển

Nội dung kế hoạch phát triển bao gồm các thông tin cách thức tổ chức về nhân sự, về quản lý chất lượng, quản lý rủi ro và quản lý cấu hình, tài nguyên của dự án. Cụ thể:

- Cách thức tổ chức nhân sự: Mô tả cấu trúc nhóm làm việc, thành phần, phân công nhiệm vụ thành viên, xác định rõ vai trò, trách nhiệm thành viên trong tổ chức. Các thức hoạt động cũng như thông tin liên lạc.
- Quản lý về chất lượng: Nêu rõ cấu trúc tổ chức quản lý chất lượng dự án. Các đề xuất về thủ tục, quy định quy ước về kỹ thuật, về tài liệu, bố trí tài nguyên. Liệt kê các kỹ thuật và công nghệ được sử dụng để đảm bảo về chất lượng.
- Quản lý rủi ro: Liệt kê nhận dạng các rủi ro xảy ra trong quá trình phát triển. Bên cạnh đó là các chiến lược giảm thiểu rủi ro và hướng giải quyết khi rủi ro xảy ra.
- Quản lý cấu hình: Nêu rõ cấu trúc tổ chức để quản lý cấu hình bao gồm trách nhiệm, vai trò cũng như từng thành phần chính trong tổ chức và cách thức

nhận dạng cấu hình, quản lý và tổ chức phát hành phân phối, quản lý các tài nguyên cấu hình, môi trường phát triển.

2.3.2 Tài liệu đặc tả yêu cầu

Bao gồm tài liệu thông tin về dự án, mô tả tổng quan về yêu cầu về chức năng và phi chức năng. Mô tả tính năng sẽ và cách thức hoạt động của chúng trong hệ thống. Mô tả các yêu cầu về giao tiếp. Tài liệu bao gồm các lưu đồ để làm rõ vấn đề yêu cầu.

- Mô tả tổng quan: mục tiêu, phạm vi, bối cảnh, các chức năng yêu cầu, đặc điểm của người sử dụng, môi trường vận hành, các ràng buộc về thực thi, thiết kế và các giả định ràng buộc.
- Yêu cầu giao tiếp: mô tả giao diện người dùng, giao tiếp các thành phần hệ thống gồm giao tiếp phần cứng, phần mềm và truyền tải thông tin.
- Tính năng hệ thống: liệt kê các tính năng, yêu cầu có trong hệ thống, bao gồm mức độ ưu tiên, tác nhân, chuỗi đáp ứng và làm rõ yêu cầu.
- Yêu cầu phi chức năng: những yêu cầu phi chức năng từ phía người dùng về thực thi, độ an toàn, bảo mật, tốc độ, thiết kế và các quy tắc nghiệp vụ.

2.3.3 Tài liệu thiết kế

Mô tả về mục tiêu, phạm vi thiết kế. Làm rõ thiết kế hệ thống từ mức ngữ cảnh để sâu hơn. Khai quát cấu trúc hệ thống, thiết kế dữ liệu, thiết kế chức năng.

- Kiến trúc hệ thống: Đưa ra cấu trúc hệ thống và các kho dữ liệu cũng như các mối liên hệ giữa các thành phần trong hệ thống. Mô tả về sự phân rã các thành phần và cơ sở thiết kế.
- Thiết kế dữ liệu: Mô tả dữ liệu, mô hình hoá dữ liệu
- Thiết kế chức năng: Mô tả mục đích, giao diện, dữ liệu, các thức hoạt động, các ràng buộc của từng chức năng.

2.3.4 Kế hoạch kiểm thử

Mô tả chi tiết các mục tiêu, thị trường đích, nhóm beta nội bộ và các quy trình cho một thử nghiệm beta cụ thể cho sản phẩm. Kế hoạch này chứa đựng hiểu biết chi tiết về luồng làm việc sẽ xảy ra. Kế hoạch kiểm thử bao gồm các phần chính như sau:

- Mục đích và phạm vi kiểm thử: Đặc tả mục đích và phạm vi sản phẩm hỗ trợ.
- Cách tiếp cận và các chiến lược được dùng: Đề cập các cấp độ kiểm thử và đặc tả về phương pháp luận kiểm thử sẽ được dùng.
- Các tính chất cần được kiểm thử: Chứa tất cả testcase cũng như tất cả trạng thái cơ bản.

- Các tính chất không cần được kiểm thử: Chứa danh sách các vùng được loại trừ khỏi kiểm thử và các testcase được định nghĩa không cần kiểm thử.
- Rủi ro và các sự cố bất ngờ: Danh sách rủi ro có thể xảy ra và phương pháp cần thực hiện để tối thiểu hóa hay sống chung với rủi ro.
- Tiêu chí đình chỉ và phục hồi kiểm thử: Các điều kiện để quá trình kiểm thử dừng lại hoặc tiếp tục việc kiểm thử đã dừng trước đó.
- Môi trường kiểm thử: Đặc tả đầy đủ về các môi trường kiểm thử.
- Lịch kiểm thử: Lịch kiểm thử ở dạng ước lượng, nên chứa các thông tin: các cột mốc với ngày xác định + Kết quả phân phối của từng cột mốc.
- Tiêu chí dừng kiểm thử và chấp nhận: Bất kỳ chuẩn chất lượng mong muốn nào mà phần mềm phải thỏa mãn cho việc phân phối đến khách hàng.
- Nhân sự: Danh sách vai trò của từng thành viên, trách nhiệm, danh sách huấn luyện và các công tác huấn luyện.
- Các kết quả phân phối: Danh sách tất cả tài liệu dự định phân phối nội bộ sau khi mỗi cột mốc kết thúc hay sau khi dự án kết thúc.

2.3.5 Tài liệu kiểm thử

Mô tả và lập kế hoạch kiểm tra tài liệu, thiết kế thử nghiệm, thực hiện thử nghiệm, kết quả thí nghiệm được rút ra từ quá trình kiểm tra. Một số tài liệu kiểm thử quan trọng:

- Test policy: Tài liệu cung cấp mô tả các nguyên tắc, phương pháp và mục tiêu kiểm thử quan trọng.
- Test strategy: Xác định mức kiểm tra được thực thi.
- Test plan: Lập kế hoạch hoàn chỉnh của hoạt động kiểm tra.
- Requirments Traceability atrix: Kết nối yêu cầu với các trường hợp kiểm thử.
- Test Scenario: Kịch bản thử nghiệm được xác minh bằng một hoặc nhiều trường hợp kiểm thử.
- Test case: Nhóm các giá trị đầu vào, điều kiện tiên quyết thực hiện, hậu quả dự kiến và kết quả.
- Test Data: Dữ liệu thoát ra trước quá trình kiểm thử, được dùng để thực hiện các trường hợp thử nghiệm.
- Defect Report: Báo cáo tài liệu về bất kỳ lỗi nào trong hệ thống.
- Test summary report: Tóm tắt các hoạt động kiểm tra được thực hiện và kết quả.

2.3.6 Tài liệu người dùng

Hỗ trợ người dùng cuối bằng cách cung cấp cho họ thông tin rõ ràng và dễ hiểu về sản phẩm hoặc dịch vụ cụ thể. Bao gồm các thành phần sau:

- Hướng dẫn từng bước.
- Hướng dẫn sử dụng.
- Câu hỏi thường gặp.
- Trình diễn video / hướng dẫn.
- Hỗ trợ nhúng Cổng thông tin hỗ trợ, trò chuyện trực tuyến, vv

2.3.7 Tài liệu báo cáo

Tập hợp các thông tin dưới dạng văn bản tài liệu nhằm thực hiện mục đích chuyển tiếp, tường trình các sự kiện trong quá trình phát triển, kiểm thử sản phẩm đồng thời thể hiện kiến nghị, đề xuất. Nội dung chính của báo cáo:

- Đánh giá tình hình hoặc mô tả sự việc, hiện tượng xảy ra.
- Phân tích nguyên nhân, điều kiện của sự việc, hiện tượng, đánh giá tình hình, xác định những công việc cần tiếp tục giải quyết.
- Nêu những phương hướng, nhiệm vụ chính để tiếp tục giải quyết các biện pháp tổ chức thực hiện, những tồn tại đang cần giải quyết, những nhiệm vụ tiếp tục đặt ra.
- Thu thập tài liệu, số liệu để đưa vào báo cáo.
- Chọn lọc tài liệu, tổng hợp sự kiện và số liệu phục vụ các yêu cầu trọng tâm của báo cáo.
- Đánh giá tình hình qua tài liệu, số liệu một cách khái quát.
- Dự kiến những đề xuất kiến nghị.

2.4 QUẢN LÝ RỦI RO

STT	Yếu tố rủi ro	Mức độ rủi ro	Chiến lược làm giảm rủi ro	Giải quyết khi xảy ra rủi ro
1	Chức năng hệ thống không đúng yêu cầu	Cao	Lấy ý kiến khách hàng, trao đổi thường xuyên khi đi đến thống nhất mới bắt đầu thiết kế phát triển chức năng. Tài liệu thiết	Trao đổi và thu thập lại ý kiến khách hàng, thay đổi thiết kế, phát triển lại chức năng hệ thống nào

			kế phải viết rõ ràng, dễ hiểu.	không đúng yêu cầu khách hàng.
2	Lỗi trong quá trình sử dụng hệ thống	Cao	Mỗi chức năng phải được kiểm thử kỹ càng, nhiều trường hợp kiểm thử, kiểm thử trên nhiều môi trường tương đồng nhất ở phía khách hàng. Tài liệu người dùng phải viết rõ ràng cho người sử dụng dễ hiểu.	Tìm hiểu nguyên nhân lỗi, nếu lỗi từ phía chủ quan khách thì thay đổi tài liệu người sử dụng để người sử dụng không thao tác sai khi sử dụng hệ thống. Nếu lỗi về chức năng, thì kiểm tra lại lỗi, tìm cách khắc phục, sửa lỗi.
3	Môi trường không thể đáp ứng hệ thống	Cao	Thu thập thông tin môi trường khách hàng, tiến hành thiết kế hệ thống phù hợp với môi trường khách hàng.	Thiết kế lại hệ thống sao cho phù hợp với môi trường khách hàng.
4	Hệ thống bị tấn công, bị phá hoại, dữ liệu người dùng bị xâm phạm	Cao	Khi thiết kế hệ thống phải được đảm bảo an toàn, bảo mật. Tổ chức lưu trữ, sao chép, khôi phục dữ liệu. Bảo mật về tài liệu thiết kế cũng như dữ liệu của khách hàng.	Phân tích, tìm nguyên nhân, lỗ hổng, tiến hành sửa chữa, khôi phục dữ liệu (nếu bị mất), bồi thường thiệt hại nếu lỗ hổng đó là từ phía bên nhà phát triển.
5	Lỗi trong quá trình triển khai, cài đặt	Trung bình	Kiểm thử cài đặt triển khai trên nhiều môi trường tương đồng phía khách hàng.	Kiểm tra nguyên nhân lỗi, sửa lỗi. Nếu lỗi do môi trường khách hàng, giải quyết như mục 3, nếu

				lỗi do hệ thống thì giải quyết như mục 2.
6	Thay đổi yêu cầu, khó khăn trong quá trình thực hiện nghiệp vụ	Trung bình	Tìm hiểu rõ yêu cầu, phân tích đúng, thiết kế hệ thống phù hợp với quy trình nghiệp vụ phía khách hàng. Luôn giữ góc nhìn là người sử dụng để thiết kế và phát triển hệ thống thuận tiện và phù hợp.	Phân tích, thu thập lại yêu cầu khách hàng, quy trình nghiệp vụ, thiết kế lại hệ thống, chức năng cho phù hợp và tiến hành phát triển lại.
7	Chức năng hệ thống không còn phù hợp, lạc hậu, hoặc có công nghệ công hơn để xử lý.	Thấp	Liên lạc, lấy ý kiến khách hàng thường xuyên, tiến hành bảo trì hệ thống định kì.	Trao đổi với khách hàng, tiến hành nâng cấp, bảo trì hệ thống

3 QUẢN LÝ CẤU HÌNH

3.1 TỔ CHỨC QUẢN LÝ CẤU HÌNH

Nhóm trưởng lập kế hoạch quản lý cấu hình, định danh, đánh số các mục cấu hình đã phân công, kiểm soát sự thay đổi của từng cấu hình, tiến hành gộp cấu hình khi các cấu hình đã được hoàn tất. Tiến hành kiểm soát phiên bản, đánh số phiên bản. Giữ vai trò và chịu trách nhiệm trong mọi thay đổi lớn cấu hình, phê duyệt, kiểm tra mỗi khi có sự thay đổi cấu hình. Tiến hành lưu trữ, sao chép khôi phục nếu có xử cố, thất thoát.

Thành viên có vai trò tuân thủ nghiêm các quy trình trong kế hoạch quản lý cấu hình đã đề ra. Chịu trách nhiệm về sự thay đổi cấu hình mà mình đã được phân công. Đồng thời phối hợp cùng nhóm trưởng và cách thành viên còn lại tìm nguyên nhân khi xảy ra thất thoát, hoặc sự cố xảy ra liên quan.

3.2 NHẬN DẠNG CẤU HÌNH

Một cấu hình sẽ được định danh như sau:

- Phần tên cấu hình: bao gồm ngày thay đổi, loại thay đổi, đối tượng thay đổi, phiên bản thay đổi.
- Nội dung: mô tả chi tiết sự thay đổi của cấu hình

VD: [27-09-2020]: FIX DatabaseAccess 1.0-draft

Nội dung: Sửa lỗi ... khi truy cập cơ sở dữ liệu

Các định danh trên cho biết:

- Thời gian thay đổi: 27-09-2020
- Loại thay đổi: FIX
- Đối tượng thay đổi: DatabaseAccess
- Phiên bản: 1.0-draft
- Nội dung: Sửa lỗi ... khi truy cập cơ sở dữ liệu

3.3 QUẢN LÝ PHÁT HÀNH VÀ PHÂN PHỐI

Lập kế hoạch phát hành và phân phối. Chuẩn bị kĩ càng trong khâu kiểm thử, hỗ trợ khách hàng như hệ thống được phát hành và phân phối chính thức. Thực hiện định danh phiên bản cho từng đợt phát hành để dễ dàng quản lý.

Tên phiên bản: được đánh số theo công thức sau:

Major.Minor.[Build [.Revision]]

VD: Version 1.0.1-alpha-test

Trong đó:

- Major: Chuỗi phiên bản chính.
- Minor: Chuỗi phiên bản phụ
- Build: Chuỗi phiên bản cấu tạo, đánh dấu sự khác nhau trong cùng 1 phiên bản phụ, 2 chữ số.
- Revision: Lần sửa đổi, đánh dấu lần sửa đổi của mã nguồn.

Chỉ số major sẽ tăng mỗi khi: Có sự thay đổi lớn trong “nhân hệ thống” mà theo đó hệ thống mới có thể khác 1 phần hay hoàn toàn hệ thống cũ.

Chỉ số minor sẽ tăng mỗi khi: Có sự thay đổi phần “core” của hệ thống mà không làm mất đi hoàn toàn tính tương thích trong cùng phiên bản chính.

Chỉ số build sẽ tăng mỗi khi: Có đóng gói gửi đi ra ngoài đội code (đội phát triển) nhằm các mục đích phát hành hay thử nghiệm.

Chỉ số revision có thể được sử dụng mỗi khi: Cần thay thế mã nguồn phát hành trước đó mà chưa cần thiết phải thay tên phiên bản.

Mỗi một phiên bản phát hành, cần ghi rõ thông tin về phiên bản bao gồm các thay đổi, sửa chữa, các tính năng mới, các tính năng bị loại bỏ cũng như bị thay đổi.

3.4 TÀI NGUYÊN QUẢN LÝ CẤU HÌNH

Sử dụng Git để quản lý cấu hình dự án, cụ thể sử dụng dụng cụ Github. Nhóm trưởng sẽ tạo repository dự án trên Github và mời các thành viên vào để đóng góp mã nguồn.

Các thành viên tiến hành tạo tài khoản Github và clone dự án về máy cá nhân. Khi viết xong mã nguồn đã được phân công tiến hành commit và push. Tên commit được đóng sẽ là tên định danh cấu hình cùng nội dung cấu hình sẽ được viết trong phần mô tả. Mọi sự thay đổi về mã nguồn sẽ được thông báo về Email của từng thành viên trong nhóm. Nhóm trưởng sẽ phê duyệt lại các thay đổi và tiến hành merge mã nguồn. Khi dự án hoàn tất, nhóm trưởng tiến hành phát hành phân phối.