|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC CẦN THƠ **KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG BỘ MÔN CÔNG NGHỆ PHẦN MỀM** | |
| Trường Đại học Cần Thơ – Wikipedia tiếng Việt | |
| **NIÊN LUẬN NGÀNH KỸ THUẬT PHẦN MỀM**  **KẾ HOẠCH PHÁT TRIỂN**  **CTU PC SHOP CAN THO UNIVERSITY PC SHOP ONLINE**  Phiên bản 1.0 | |
| **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 dụng 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 |
|  |  |  |  |

# 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 vài 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](https://github.com/daomtthuan/ctu-pc-shop).

# QUẢN LÝ CHẤT LƯỢNG PHẦN MỀM

## 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.

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

### 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 câu của giảng viên hướng dẫn.
* Thư mực 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\_KeHoachPhatTrienPhanMem.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 phần/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.

### 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.

### Các chuẩn và quy ước trong lập trình

#### 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.

#### 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 lệ rẻ 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 vao đó sử dụng kiểu khai báo biến cục bộ let.

#### Quy ước về cách viết chú thích:

Nên viết chú thích ở những đoạn mã nguồn mà bản than 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 khan 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 @parm 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, môi tả lỗi được ném ra

#### Các chuẩn và thực tiển kiểm thử

Viet gi do o day

## KIỂM SOÁT CHẤT LƯỢNG

### Kế hoạch phát triển

Nôi 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.

### 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.
* Tinh 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.

### 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.
* Thiếu 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.

### Kế hoạch kiểm thử

### Tài liệu kiểm thử

### Tài liệu người dung

### Tài liệu báo cáo

### Tài liệu khác

## QUẢN LÝ RỦI RO

# QUẢN LÝ CẤU HÌNH

## TỔ CHỨC QUẢN LÝ CẤU HÌNH

## NHẬN DẠNG CẤU HÌNH

## QUẢN LÝ PHÁT HÀNH VÀ PHÂN PHỐI

## TÀI NGUYÊN QUẢN LÝ CẤU HÌNH