LỜI NÓI ĐẦU

Ngày nay, ngành công nghệ thông tin đang phát triển vô cùng mạnh mẽ. Nó đã chứng tỏ được sức mạnh của mình trong mọi lĩnh vực của cuộc sống. Trong thực tế đã có rất nhiều cơ quan công sở, xí nghiệp, trường học, bệnh viên đã đưa tin học vào ứng dụng. Trong đó việc quản lí thông tin là một ứng dụng cần thiết và mang lại nhiều lời ích. Nó giúp cho các cơ quan, trường học, bệnh viện tiết kiệm thời gian, tiền bạc, nhân lực và có độ chính xác, tin cậy cao.

Việc áp dụng tin học, đưa máy tính vào trợ giúp quản lý ký túc xá sinh viên là rất cần thiết, nó giúp cho người quản lý thu hẹp không gian lưu trữ, tránh được thất lạc dữ liệu, xử lý nhanh một khối lượng lớn về thông tin, tra cứu tìm kiếm một cách nhanh chóng, tiết kiệm được nhiều thời gian và công sức người quản lý và sinh viên.

Xuất phát từ nhận thức trên, với mong muốn áp dụng công nghệ thông tin vào quản lý, dựa trên những hoạt động thực tiến em xin tìm hiểu và thực hiện đề tài “Hệ thống quản lý ký túc xá trường Đại học Bách Khoa Hà Nội” với các chức năng đăng ký phòng online và quản lý ký túc xá.

Trong quá trình thực hiện đề tài, mặc dù được sự hướng dẫn rất nhiệt tình của thầy Kiều Đức Hạnh cùng với sự nỗ lực cá nhân nhưng cũng không thể tránh khỏi những thiếu sót. Em rất mong nhận được những ý kiến đóng góp từ quý Thầy, Cô và bạn bè để em học thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt hơn.

Em xin chân thành cảm ơn!

Hà Nội, ngày tháng năm 2019

|  |  |  |
| --- | --- | --- |
| BỘ LAO ĐỘNG TB & XÃ HỘI |  | CỘNG HÒA XÃ HÔI CHỦ NGHĨA  VIỆT NAM |
| **TRƯỜNG CĐN BÁCH KHOA HÀ NỘI** |  | **Độc lập - Tự do - Hạnh phúc** |

**NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP**

Họ và tên sinh viên: Nguyễn Thị Hồng Duyên Mã số sinh viên: CD164125

Khoá: 8 Khoa: Công nghệ thông tin Nghề: Lập trình

1. Tên đề tài: Xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá.
2. Nội dung các phần thuyết minh và tính toán:

* Mục tiêu
* Tìm hiểu công nghệ PHP, CodeIgniter, MySQL, HTML, CSS.
* Sử dụng công nghệ đã tìm hiểu để xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá.
* Nhiệm vụ
* Tìm hiểu về công nghệ PHP, CodeIgniter, MySQL
* Hoàn thành hệ thống đăng ký trực tuyến và quản lý ký túc xá.

1. Họ tên giảng viên hướng dẫn: Thầy Kiều Đức Hạnh
2. Ngày giao nhiệm vụ đồ án:
3. Ngày hoàn thành đồ án:

Hà Nội, ngày tháng năm 2019

Chủ nhiệm khoa Giảng viên hướng dẫn

BẢN TÓM TẮT ĐỒ ÁN TỐT NGHIỆP

**TÊN ĐỀ TÀI: Xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá.**

Sinh viên thực hiện: NGUYỄN THỊ HỒNG DUYÊN

Lớp: LTMT1-K8

Giáo viên hướng dẫn: Thầy Kiều Đức Hạnh

**TÓM TẮT ĐỀ TÀI**

Hiện nay, các website, phần mềm hỗ trợ giáo dục đã xuất hiện và phát triển mạnh mẽ, nhằm đáp ứng nhu cầu của học sinh sinh viên, cũng như hỗ trợ quản lý cho cán bộ công nhân viên của nhà trường. Bên cạnh các website học tập, nhà trường cũng cần quản lý việc lưu trú của sinh viên tại ký túc xá. Hệ thống đăng ký trực tuyến và quản lý ký túc xá được xây dựng với 2 chức năng chính là đăng ký phòng trực tuyến dành cho sinh viên và quản lý ký túc dành cho cán bộ. Hệ thống hướng tới các giải pháp để khắc phục những hạn chế của hệ thống cũ và đem lại lợi ích cho cả sinh viên và cán bộ quản lý.

**CÁC MỤC TIÊU CHÍNH**

* Tìm hiểu công nghệ PHP, CodeIgniter, MySQL, HTML, CSS.
* Ứng dụng công nghệ để xây dựng hế thống đăng ký trực tuyến và quản lý ký túc xá.

**KẾT QUẢ DỰ KIẾN**

* Xây dựng thành công hệ thống và đưa vào sử dụng trong thực tế.
* Thành thạo ngôn ngữ PHP, framework CodeIgniter, MySQL,…
* Viết báo cáo tổng kết

MỤC LỤC

**MỞ ĐẦU**

Trong thời đại công nghệ 4.0 hiện nay, các website, phần mềm hỗ trợ giáo dục đã xuất hiện và phát triển mạnh mẽ, nhằm đáp ứng nhu cầu của học sinh sinh viên, cũng như hỗ trợ quản lý cho cán bộ công nhân viên của nhà trường. Trong đó, việc quản lý sinh viên đăng ký lưu trú tại ký túc xá của nhà trường cũng là một bài toán thu hút sự quan tâm của nhiều người.

1. Lý do chọn đề tài

Ngày nay, với sự phát triển của các trường học, trường đại học, cao đẳng về quy mô diện tích sẽ dẫn đến nhu cầu lưu trú tại trường của sinh viên tăng cao. Với cách làm truyền thống, đầu năm học hoặc đầu kỳ học, sinh viên sẽ cần đến ban quản lý của ký túc xá để đăng ký lưu trú. Điều đó dẫn đến sự quá tải cho ban quản lý ký túc vì sinh viên đăng ký liên tục. Sinh viên đăng ký lưu trú cần chuẩn bị giấy tờ thủ tục đầy đủ trước khi đăng ký cũng là một vấn đề gây ra sự chậm trễ cho việc sắp xếp của ban quản lý. Khi số lượng dữ liệu tăng dẫn đến việc quản lý trên sổ sách dễ gây nhầm lẫn, thất lạc giấy tờ. Bên cạnh đó, sinh viên sẽ phải chờ đợi để đăng ký phòng, nếu gặp trường hợp ký túc xá hết phòng, hết chỗ thì phải tìm ký túc xá khác để đăng ký. Việc sắp xếp phòng phụ thuộc vào ban quản lý ký túc khiến sinh viên khá bị động.

Hệ thống đăng ký trực tuyến và quản lý ký túc xá được xây dựng nhằm giải quyết các hạn chế trên của hệ thống cũ. Từ đó có thể tiết kiệm thời gian, công sức của cả sinh viên và cán bộ quản lý ký túc.

Với mục đích trên, em chọn đề tài “Xây dựng hệ thống đăng ký trực tuyến và quản lý ký túc xá” để giải quyết vấn đề đăng ký và quản lý ký túc xá làm đề tài tốt nghiệp của mình.

1. Đối tượng và phạm vi đề tài

* Đối tượng: sinh viên và cán bộ quản lý ký túc xá.
* Phạm vi: đăng ký phòng trực tuyến và quản lý sinh viên lưu trú.

1. Phương pháp nghiên cứu

* Khảo sát nhu cầu thực tế của sinh viên
* Tham khảo hệ thống quản lý truyền thống
* Tham khảo cách tính hóa đơn tại ký túc xá
* Xây dựng cơ sở dữ liệu
* Lập trình website và hoàn thiện

1. Nội dung đồ án

Nội dung đồ án gồm 3 chương:

***Chương 1: Tìm hiểu công cụ lập trình PHP, CodeIgniter, MySQL***

Nội dung chính của chương là tìm hiểu về PHP cơ bản, framework của PHP là CodeIgniter, hệ quản trị cơ sở dữ liệu MySQL.

***Chương 2: Phân tích và thiết kế hệ thống***

Nội dung của chương này là phân tích và thiết kế đối tượng, chức năng của hệ thống.

***Chương 3: Kết luận và đánh giá***

Chương nêu ra kết quả của đề tài và hướng phát triển của hệ thống đã xây dựng.

**CHƯƠNG I: TÌM HIỂU CÔNG CỤ LẬP TRÌNH PHP, FRAMEWORK CODEIGNITER, MYSQL**

1. **Ngôn ngữ lập trình PHP**
   1. Khái niệm PHP

PHP - viết tắt hồi quy của **"Hypertext Preprocessor"**, là một ngôn ngữ lập trình kịch bản được chạy ở phía server nhằm sinh ra mã html trên client. PHP đã trải qua rất nhiều phiên bản và được tối ưu hóa cho các ứng dụng web, với cách viết mã rõ rãng, tốc độ nhanh, dễ học nên PHP đã trở thành một ngôn ngữ [lập trình web](http://hoclaptrinhweb.org/hoc-thiet-ke-web/lap-trinh-web.html) rất phổ biến và được ưa chuộng.

* 1. Cách hoạt động của PHP

Khi người sử dụng gọi trang PHP, Web Server sẽ triệu gọi PHP Engine để thông dịch dịch trang PHP và trả kết quả cho người dùng như hình bên dưới.



Hình 1.2.1: Sơ đồ hoạt động của PHP

* 1. Cú pháp khai báo hằng và biến

**Biến**: Không giống như hằng, biến không cần định nghĩa, có thể thay đổi trong chương trình. Cách đặt tên của biến cũng giống như cách đặt tên của hằng. Biến phân biệt chữ hoa và chữ thường. Tất cả biến trong PHP đều bắt đầu bằng dấu "$". PHP tự động hiểu các kiểu biến nên người dùng không cần phải khai báo kiểu biến.

**Hằng**: Tương tự như trong những ngôn ngữ lập trình khác, hằng trong PHP cũng là một giá trị được khai báo trước khi sử dụng. Nó phải bắt đầu bằng chữ cái hoặc dấu gạch dưới \_ và không được bắt đầu bằng số, nó có thể là một chuỗi. Hằng trong PHP được định nghĩa bằng hàm define(). Nó không thay đổi và không được được định nghĩa lại.

* 1. Các kiểu dữ liệu

PHP có tổng cộng tám loại dữ liệu được sử dụng cho các biến, các kiểu dữ liệu này thường được hiểu ngầm dựa vào giá trị của biến mà không cần phải khai báo. Để có thể [học lập trình php](http://hoclaptrinhweb.org/hoc-thiet-ke-web/lap-trinh-web.html) chúng ta cần phải nắm vững các kiểu dữ liệu và toán tử của PHP.

* Integers: là kiểu số nguyên, không có dấu thập phân, ví dụ 1234.
* Doubles: là kiểu số thực, ví dụ 3.64576.
* Booleans: chỉ có hai giá trị true hoặc false.
* NULL: là một kiểu dữ liệu đặc biệt chỉ có một giá trị: NULL.
* Strings: là chuỗi ký tự, giống như 'tôi học php'.
* Array: kiểu dữ liệu mảng - là một tập hợp các phần tử có cùng kiểu dữ liệu với nhau.
* Object: là 1 thể hiện của 1 lớp đã được định nghĩa trước - được sử dụng trong lập trình hướng đối tượng.
* Resource: là kiểu dữ liệu đặc biệt dùng để lưu trữ tham chiếu đến các tài nguyên bên ngoài PHP chẳng hạn như kết nối đến database.
  1. Toán tử trong PHP

**Toán tử số học:**

|  |  |  |
| --- | --- | --- |
| Toán tử | Mô tả | Ví dụ |
| + | Cộng 2 số | A + B = 30 |
| - | Trừ 2 số | A - B = -10 |
| \* | Nhân 2 số | A \* B = 200 |
| / | Chia 2 số | B / A = 2 |
| % | Toán tử Modulo - lấy số dư còn lại sau khi chia 2 số | B % A = 0 |
| ++ | Cộng thêm 1 | A++ = 11 |
| -- | Trừ đi 1 | A-- = 9 |

**Toán tử so sánh:**

|  |  |  |
| --- | --- | --- |
| Toán tử | Mô tả | Ví dụ |
| == | Kiểm tra giá trị 2 toán hạng có bằng nhau hay không, nếu bẳng trả về true, ngược lại là false. | (A == B) is not true. |
| != | Kiểm tra giá trị 2 toán hạng có không bằng nhau hay không, nếu không bẳng trả về true, ngược lại là false. | (A != B) is true. |
| > | Kiểm tra toán hạng bên trái có lớn hơn bên phải hay không, nếu đúng trả về true. | (A > B) is not true. |
| < | Kiểm tra toán hạng bên trái có nhỏ hơn bên phải hay không, nếu đúng trả về true. | (A < B) is true. |
| >= | Kiểm tra toán hạng bên trái có lớn hơn hoặc bằng bên phải hay không, nếu đúng trả về true. | (A >= B) is not true. |
| <= | Kiểm tra toán hạng bên trái có nhỏ hơn hoặc bằng bên phải hay không, nếu đúng trả về true. | (A <= B) is true. |

**Toán tử gán:**

|  |  |  |
| --- | --- | --- |
| Toán tử | Mô tả | Ví dụ |
| = | Gán giá trị toán hạng bên phải sang toán hạng bên trái | C = A + B sẽ gán giá trị của A + B cho C |
| += | Cộng toán hạng bên trái với toán hạng bên phải và gán kết quả cho toán hạng bên trái | C += A tương đương với C = C + A |
| -= | Trừ toán hạng bên trái với toán hạng bên phải và gán kết quả cho toán hạng bên trái | C -= A tương đương với C = C - A |
| \*= | Nhân toán hạng bên trái với toán hạng bên phải và gán kết quả cho toán hạng bên trái | C \*= A tương đương với C = C \* A |
| /= | Chia toán hạng bên trái với toán hạng bên phải và gán kết quả cho toán hạng bên trái | C /= A tương đương với C = C / A |
| %= | Modulo toán hạng bên trái với toán hạng bên phải và gán kết quả cho toán hạng bên trái | C %= A tương đương với C = C % A |

* 1. Các lệnh điều khiển
* **Câu lệnh if else:** câu lệnh này cho phép chúng ta kiểm tra một điều kiện nào đó, nếu đúng(true) thì thực hiện tiếp những gì trong block của if, nếu sai thì thực hiện các câu lệnh trong else.

if (condition) {  
    #code;  
} else {  
    #code;  
}

* **Câu lệnh switch case**: nếu có nhiều điều kiện ràng buộc thay vì phải if else nhiều lần ta có thể sử dụng switch case để thay thế.

switch(n) {  
    case label1: #code; break;

case label2: #code; break;

. . .   
    default: #code;  
}

* **Câu lệnh while**: nếu biểu thức trong while trả về giá trị TRUE thì tiếp tục thực hiện câu lệnh, sau khi thực hiện câu lệnh thì kiểm tra lại biểu thức, nếu vẫn còn trả về giá trị TRUE thì lại tiếp tục thực hiện câu lệnh...cứ tiếp tục như vậy cho tới khi nào biểu thức trả về giá trị FALSE thì thoát khỏi vòng lặp.

while (condition is true) {  
    #code;  
}

* **Câu lệnh do while**: Vòng lặp này giống như vòng lặp while, nhưng thay vì kiểm tra điều kiện vào lúc đầu của đoạn lệnh cần lặp, thì nó lại kiểm tra giá trị điều kiện vào cuối vòng lặp. Điều này có nghĩa là nó luôn luôn thực hiện đoạn lệnh cần lặp ít nhất một lần.

do {  
    #code;  
} while (condition is true);

* Câu lệnh for: được sử dụng để lặp với số lần nhất định với điều kiện được định nghĩa trước.

for(*init counter;test counter;increment counter*) {  
  *code to be executed;*  
}

* Câu lệnh foreach: Câu lệnh foreach chỉ làm việc với mảng.

foreach ($*array*as$*value*) {  
    *code to be executed;*  
}

* **Câu lệnh break**: được sử dụng để dừng việc thực thi của các vòng lặp for, foreach, while, do-while và switch.

break;

* **Câu lệnh continue**: bỏ qua lần lặp hiện thời và tiếp tục thực hiện các lần lặp tiếp theo trong các vòng lặp for, while, do-while

continue;

* 1. Mảng

Mảng là một danh sách các phần tử có cùng kiểu dữ liệu. Mảng có thể là mảng một chiều hay nhiều chiều. Mảng có 2 thành phần là chỉ mục (key) và giá trị. Các loại mảng trong PHP:

* Mảng số (numeric array)
* Mảng  kết hợp (associative array)
* Mảng đa chiều
  1. Lập trình hướng đối tượng

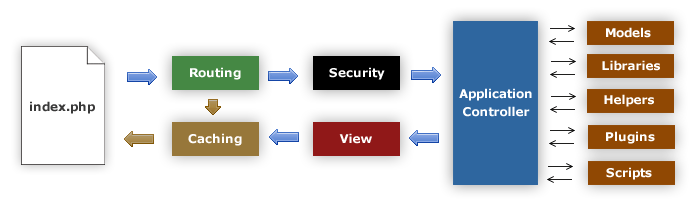
Lập trình hướng đối tượng (OOP) là kiểu viết mã sử dụng lớp (classes), quan hệ (relationships), thuộc tính (properties) và phương thức (methods) của đối tượng (objects) nhằm giảm thời gian viết mã và giúp đơn giản hóa việc lập trình. Các đặc điểm của lập trình hướng đối tượng:

* Tính trừu tượng (Abstraction )
* Tính đóng gói (Encapsulation)
* Tính kế thừa (Inheritance)
* Tính đa hình (Polymorphism)

1. **Framework CodeIgniter**
   1. CodeIgniter là gì?

CodeIgniter là nền tảng ứng dụng web nguồn mở được viết bằng ngôn ngữ PHP bởi Rick Ellis. Phiên bản đầu tiên được phát hành vào ngày 28/02/2006. Hiện tại CodeIgniter đang được phát triển bởi ExpressionEngine Development Team thuộc EllisLab, Inc.

* 1. Tổ chức dữ liệu trong CodeIgniter



* + Tập tin index.php đóng vai trò làm controller đầu vào, thiết lập các tài nguyên cần thiết cho hệ thống.
* Routing: Điều hướng giúp xác định các yêu cầu và hướng xử lý đối với chúng.
* Caching: Nếu dữ liệu được yêu cầu đã được lưu trong bộ đệm, CodeIgniter sẽ trả dữ liệu trong bộ đệm về phía client. Quá trình xử lý kết thúc.
* Security: Dữ liệu trước khi được chuyển đến các Controller sẽ được lọc để phòng chống XXS hoặc SQL Injection.
* Application Controller: Controller xử lý dữ liệu nhận được bằng cách gọi đến các Models, Libraries, Helpers, Plugins...có liên quan.
* View: Dữ liệu được chuyển qua View để hiển thị cho người dùng. Nếu chức năng caching được bật, dữ liệu sẽ được lưu trong cache cho những lần yêu cầu tiếp theo.
  1. Mô hình MVC trong CodeIgniter
  + Model là những lớp được xây dựng nhằm thực hiện việc trao đổi thông tin với cơ sở dữ liệu. Một lớp Model có thể thực hiện các tác vụ truy vấn, thêm, xóa, cập nhật dữ liệu.
  + View là những tập tin HTML được xây dựng nhằm thể hiện dữ liệu trong model thành các giao diện tương t|c với người dùng. View có thể là một trang web hoàn chỉnh, hay chỉ là một phần của trang web (header, footer, sidebar...).
* Controller là những lớp đóng vai trò trung gian giữa view và model. Controller nhận các yêu cầu từ phía người dùng, kiểm tra chúng trước khi chuyển qua cho model. Sau khi model xử lý yêu cầu và trả dữ liệu về, controller chuyển sang view để hiển thị dữ liệu cho người dùng.

1. **Hệ quản trị cơ sở dữ liệu MySQL**

**CHƯƠNG II: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

**CHƯƠNG III: KẾT LUẬN VÀ ĐÁNH GIÁ**