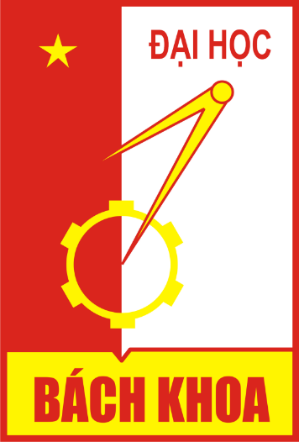
**Trường Đại Học Bách Khoa Hà Nội**

**Viện Điện Tử - Viễn Thông**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



BÁO CÁO ĐỒ ÁN 3

**Xây dựng website quản lý tiến độ công việc dùng cakephp**

Giáo viên hướng dẫn: thầy Nguyễn Tiến Hòa

Sinh viên thực hiện:

Tạ Anh Tú 20134509 ĐTTT 04 – K58

Trần Văn Thanh Toàn 20134027 ĐTTT 03 – K58

*Hà Nội, 12/2017*

Mục lục

[1. Lập kế hoạch 3](#_Toc502873553)

[1.1. Giới thiệu đề tài 3](#_Toc502873554)

[1.2. Phân tích khả thi 3](#_Toc502873555)

[1.3. Phân công công việc 4](#_Toc502873556)

[2. Phân tích 4](#_Toc502873557)

[2.1. Xác định yêu cầu 4](#_Toc502873558)

[2.1.1. Yêu cầu chức năng 4](#_Toc502873559)

[2.1.2. Yêu cầu phi chức năng 5](#_Toc502873560)

[2.2. Mô hình hóa chức năng 5](#_Toc502873561)

[2.2.1. Activity diagram 5](#_Toc502873562)

[2.2.2. Use case diagram 12](#_Toc502873563)

[2.3. Mô hình hóa cấu trúc 13](#_Toc502873564)

[2.3.1. Class diagram 13](#_Toc502873565)

[3. Thiết kế 13](#_Toc502873566)

[3.1. Mô hình thực thể liên kết 13](#_Toc502873567)

[3.2. Thiết kế cơ sở dữ liệu 14](#_Toc502873568)

[3.3. Giới thiệu về framework Cakephp 15](#_Toc502873569)

[3.4. Thiết kế sản phẩm 18](#_Toc502873570)

[Tài liệu tham khảo 19](#_Toc502873571)

# 1. Lập kế hoạch

## 1.1. Giới thiệu đề tài

Do nhu cầu thực tế cần quản lý các công việc hiện tại của từng thành viên trong 1 tổ chức, công ty hay phòng lab nghiên cứu, nhóm đã thống nhất chọn đề tài xây dựng hệ thống quản lý tiến độ công việc. Hệ thống có thể áp dụng cho 1 nhóm, 1 phòng lab với quy mô nhỏ.

## 1.2. Phân tích khả thi

1.2.1. Khả thi về kỹ thuật

* Về mặt ứng dụng:

Dễ dàng sử dụng, tra cứu nhanh chóng, tiết kiệm thời gian cho nhân viên, sinh viên, giáo viên…

* Về mặt công nghệ:

Nhóm dự tính sẽ sử dụng ngôn ngữ lập trình php, với framework CakePHP để thiết kế giao diện website.

* Yêu cầu kĩ thuật:

+ Phần cứng: cấu hình tối thiểu:

* Ram 1Gb
* CPU 1,7Ghz

+ Hệ điều hành: Window/Linux/MacOS/IOS/Android/Java… Yêu cần cần có trình duyệt: Chrome/Firefox/Safari/Opera/IE…

* Tính tương thích:

Có khả năng tương thích tốt với hệ thống hiện tại, sử dụng cơ sở dữ liệu sẵn có.

1.2.2. Khả thi về tổ chức

* Đánh giá khả năng:

+ Hệ thống cuối cùng sẽ đáp ứng tốt người dùng.

+ Hệ thống giúp đơn giản hóa công việc, giúp công tác quản lý khoa học và quy củ.

* Tổ chức con người: Hệ thống (nếu có thể) sẽ được sử dụng bởi phòng lab C9-421.

## 1.3. Phân công công việc

Sau đây là bảng phân công công việc của nhóm

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Task group** | | **Task name** | **Person In Charge** |
|
|
|  |  |  |  |  |
| 1 | Lập kế hoạch | Khởi tạo dự án | Lập nhóm, đề xuất ý tưởng và tổng hợp các đề tài | Cả nhóm |
| 2 | Đánh giá, lựa chọn đề tài | Cả nhóm |
| 3 | Nhận biết dự án | Cả nhóm |
| 4 | Phân tích tính khả thi về kỹ thuật, tổ chức | Toàn |
| 5 | Quản trị dự án | Xác định quy mô dự án | Cả nhóm |
| 6 | Xây dựng và tạo bảng kế hoạch | Tú |
| 7 | Phân công công việc | Tú |
| 8 | Phân tích | Xác định rõ yêu cầu | Phân tích các yêu cầu chức năng, phi chức năng | Toàn |
| 9 | Mô hình hóa chức năng | Xây dựng biểu đồ hoạt động (Activity diagram) | Toàn |
| 10 | Xây dựng biểu đồ kịch bản sử dụng (Use case) | Toàn |
| 11 | Mô hình hóa cấu trúc | Xây dựng biểu đồ lớp (Class diagram) | Tú |
| 12 | Thiết kế |  | Xây dựng mô hình thực thể và liên kết (Entity Relationship Model) | Toàn |
| 13 | Thiết kế CSDL | Toàn |
| 14 | Thiết kế giao diện người dùng (dùng CSS) | Toàn |
| 15 | Triển khai |  | Cài đặt và vận hành | Cả nhóm |
| 16 | Testing, debug | Cả nhóm |
| 17 | Phát hành sản phẩm | Tú |

# 2. Phân tích

## 2.1. Xác định yêu cầu

### 2.1.1. Yêu cầu chức năng

Người dùng của hệ thống được chia thành 2 loại: giáo viên và sinh viên. Với mỗi loại người dùng hệ thống có các yêu cầu chức năng sau:

* Với giáo viên (GV)
  + Đăng nhập, đăng xuất, đổi mật khẩu
  + Thêm sinh viên (tạo tài khoản cho SV) (nghĩa là giáo viên sẽ quyết định sinh viên nào sẽ được sử dụng hệ thống). SV không thể tự đăng ký 1 tài khoản để sử dụng hệ thống này!
  + Xem tiến độ công việc của từng SV
  + Giải đáp câu hỏi của SV
  + Quản lý bài đăng, xóa bài đăng
  + Xem nhật ký các công việc của tất cả sinh viên
* Với sinh viên (SV)
  + Đăng nhập, đăng xuất, đổi mật khẩu
  + Tạo công việc mới (để GV theo dõi, giám sát)
  + Cập nhật tiến độ công việc của mình
  + Hỏi đáp khi gặp vấn đề khó
  + Đăng bài, bình luận (giống fb, edmodo)

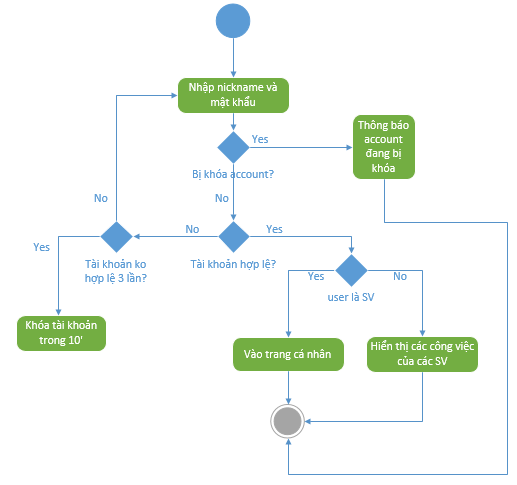
### 2.1.2. Yêu cầu phi chức năng

* Hệ thống chạy ổn định, ít lỗi
* Tương thích với các loại hệ điều hành khác nhau
* Dễ dàng sử dụng, bảo trì, nâng cấp
* Đáp ứng nhanh, phục vụ nhiều người dùng mà ko bị treo, lag
* Hiển thị các thông tin cần thiết rõ ràng, đơn giản

## 2.2. Mô hình hóa chức năng

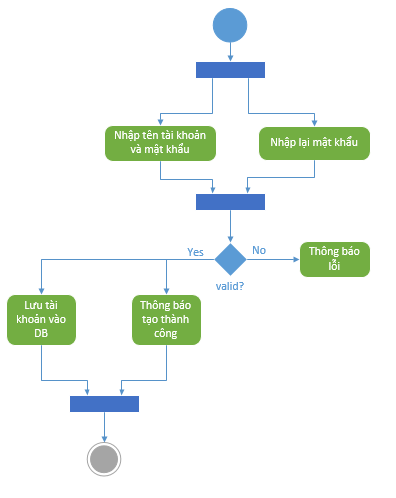
### 2.2.1. Activity diagram

* Login

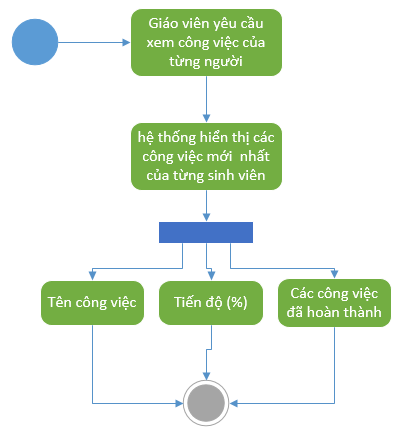


a) Giáo viên

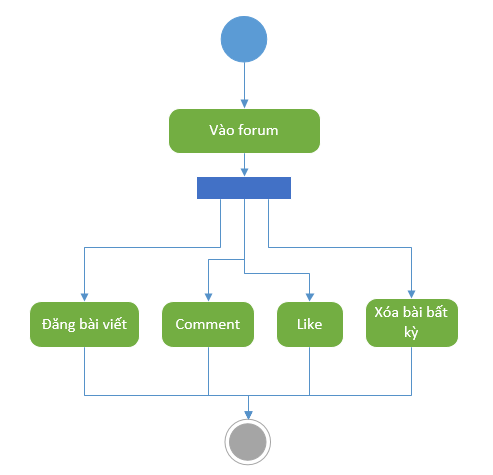
* Tạo tài khoản cho SV



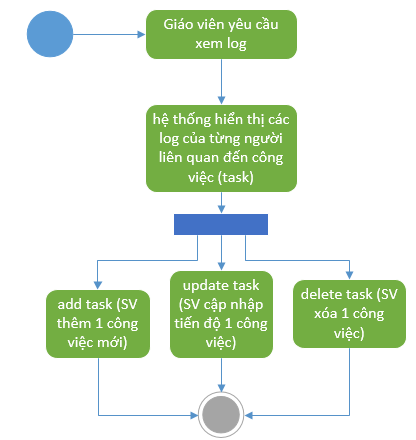
* Xem tiến độ công việc của từng thành viên



* Các hoạt động trên forum

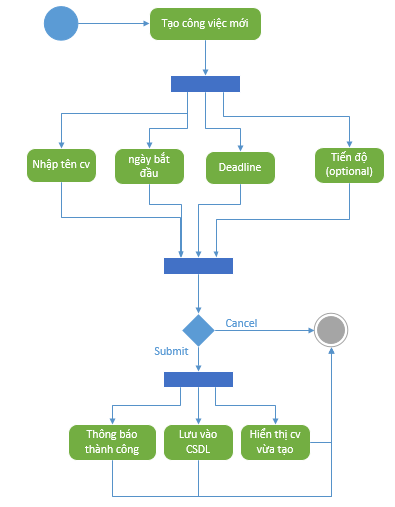


* Xem log

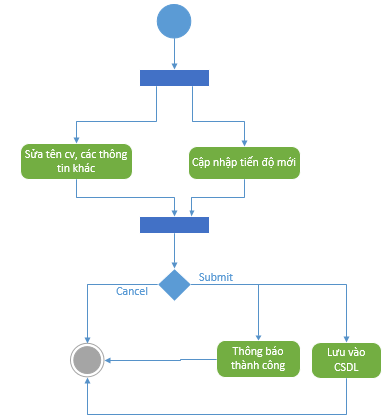


b) Sinh viên

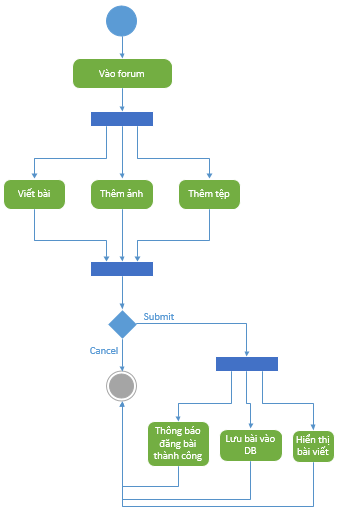
* Login: giống giáo viên
* Thêm cv



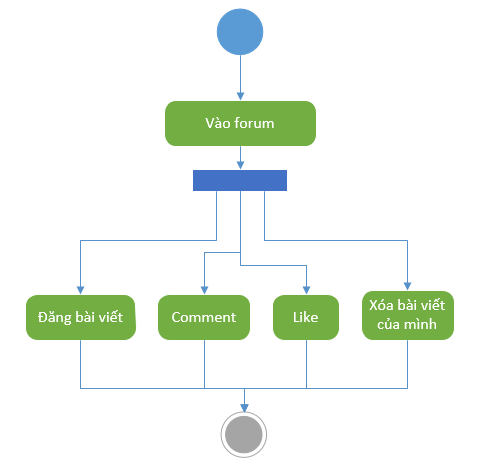
* Cập nhật công việc, tiến độ



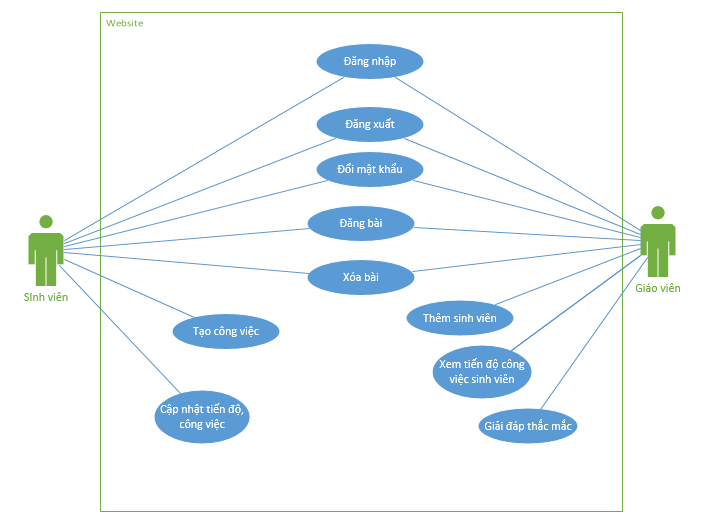
* Đăng bài viết



* Các hoạt động trên forum

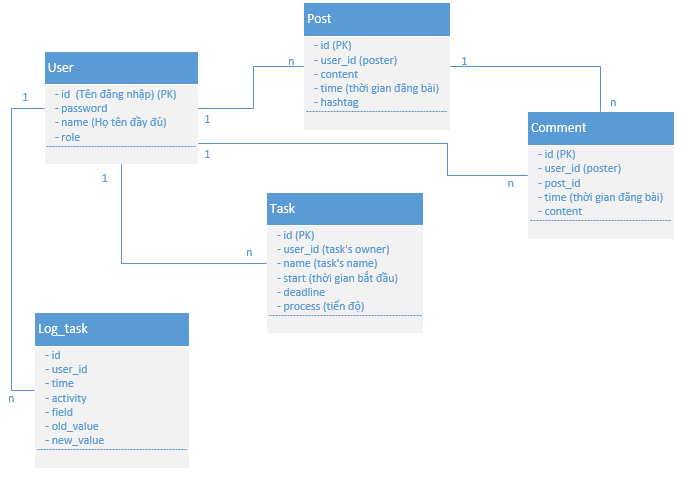


### 2.2.2. Use case diagram



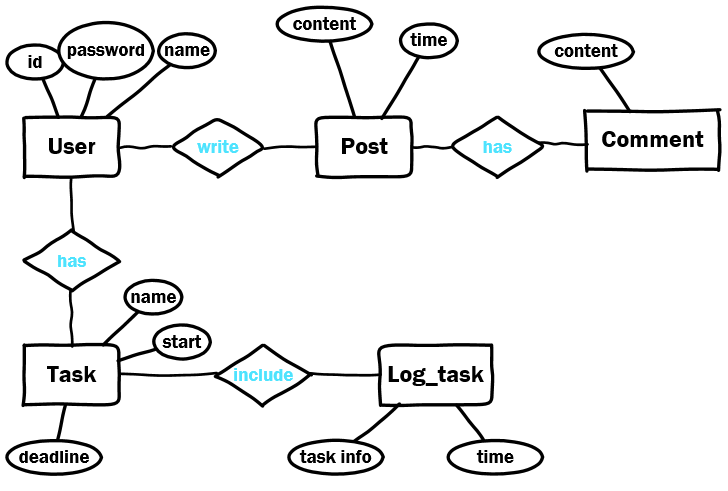
## 2.3. Mô hình hóa cấu trúc

### 2.3.1. Class diagram



# 3. Thiết kế

## 3.1. Mô hình thực thể liên kết



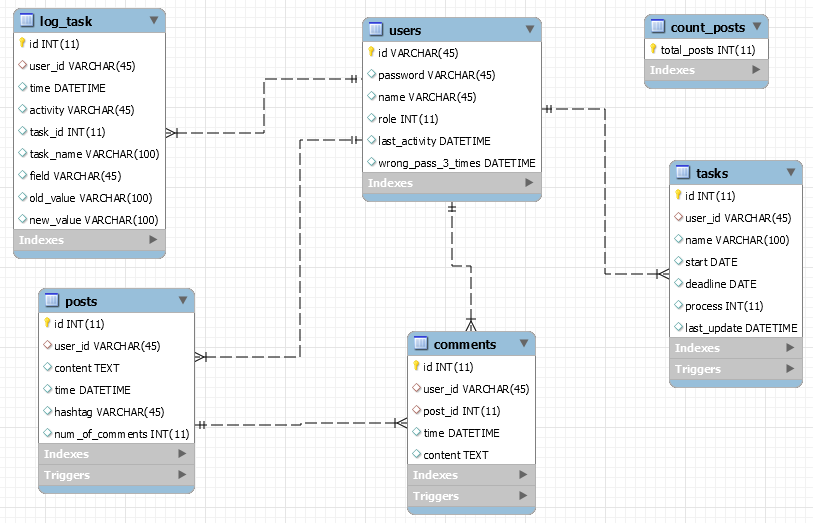
## 3.2. Thiết kế cơ sở dữ liệu

- Sau đây là từng bảng:

* `user`: chứa tài khoản của người dùng
* `task`: chứa công việc của các sinh viên
* `posts`: chứa những bài đăng trên forum
* `comments: chứa các comment của từng post
* `log\_task`: nhật ký ghi lại các hoạt động của sinh viên với các công việc của mình
* `count\_posts`: chứa 1 giá trị duy nhất: tổng số bài đăng trong forum

- Chú ý:

* Ko tạo được FK cho field `task\_id` của bảng `log\_task` tham chiếu tới id của `task`, vì nếu có 1 task bị xóa thì id của task đó vẫn có trong 1 vài record của bảng này
* 1 số chỗ vi phạm dạng chuẩn 3 trong thiết kế cơ sở dữ liệu, chẳng hạn, field task\_name trong bảng `log\_task` là thừa, nhưng vẫn để như vậy để tăng tốc truy vấn



## 3.3. Giới thiệu về framework Cakephp

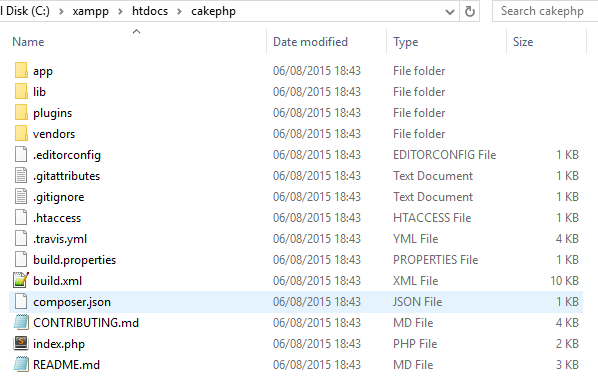
a) Giới thiệu chung

- CakePHP là một framework cho php, mục đích của nó là cung cấp một framework cho người sử dụng php để phát triển các ứng dụng web một cách nhanh chóng, dễ dàng, linh hoạt. Nó còn là một OpenSource. Để sử dụng và phát triển đòi hỏi người sử dụng phải có am hiểu về PHP và HTML, một số kiến thức trong lập trình hướng đối tượng, mô hình MVC.

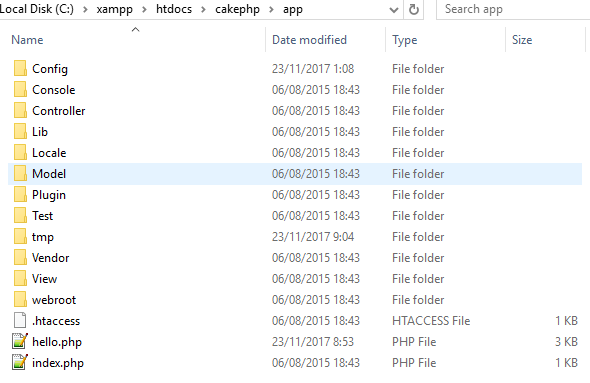
- Tính năng

* Nguồn mở, được miễn phí và có hỗ trợ rộng lớn từ cộng đồng.
* Theo mô hình đang được sử dụng rất nhiều MVC.
* Hỗ trợ sinh code tự động.
* Tương thích với PHP4 và PHP5
* Có hỗ trọ caching.
* Phân quyền (ACL).
* Đa ngôn ngữ, đa giao diện.
* Hỗ trợ kiểm tra các ràng buộc dữ liệu.
* Xây dựng nhiêu thư viện hỗ trợ cho cả view, controller.
* Dễ xây dựng thư viện của riêng mình, liên kết với các ứng dụng khác.
* Hỗ trợ nhiều hệ quản trị cơ sở dữ liệu.

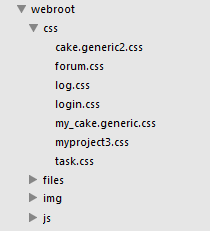
b) Cấu trúc thư mục của framework



* Ý nghĩa các thư mục:
  + app: chứa tất cả source code của ứng dụng (là thư mục quan trọng nhất)
  + lib: chứa thư viện của cakephp
  + plugins: chứa các thành phần mở rộng của ứng dụng
  + vendors: là các ứng dụng bên thứ 3 có thể là phpmail, facebook…
* Trong thư mục app:



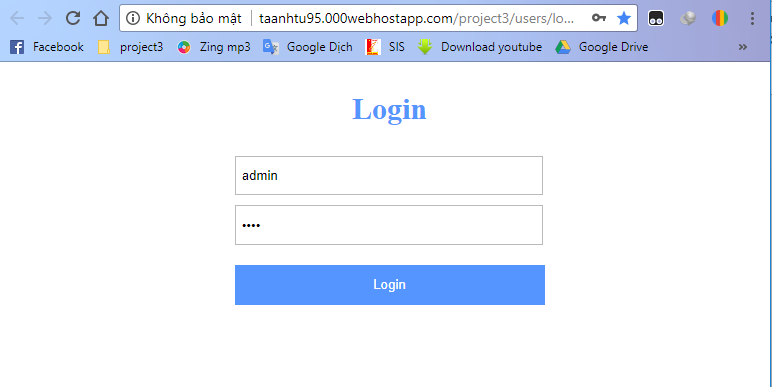
* Sau đây là nội dung các thư mục quan trọng
  + Config: chứa file cấu hình của hệ thống như database, routes…
  + Controller: chứa file AppController.php, các controller và Component(tự tạo)
  + Model: chứa file Model và behavor, datasource
  + tmp: đây là nơi chứa các tập tin tạm được tự động sinh ra của ứng dụng
  + View: chứa các file giao diện hiển thị và Helper
  + webroot: chứa các tập tin tài liệu về hình ảnh, javascript, css… Ví dụ:

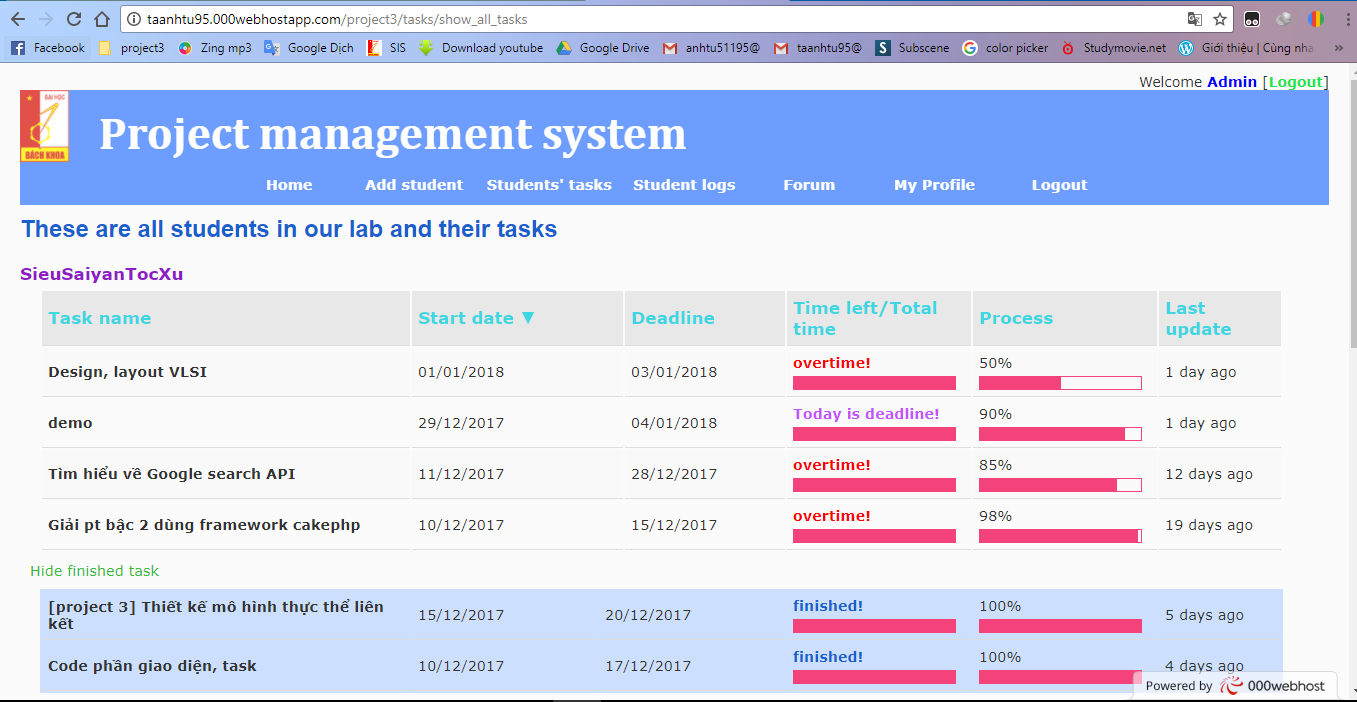


* Ngoài các thư mục trên còn có các thư mục khác có ý nghĩa như sau:
  + Console: hỗ trợ tự động sinh code qua command line
  + Lib: thư viện tự tạo của chúng ta
  + Locale: chứa file ngôn ngữ, sử dụng cho ứng dụng đa ngôn ngữ.
  + Plugin: chứa các gói mở rộng cho ứng dụng.
  + Test: dùng để test chương trình
  + Vendor: chứa ứng dụng của bên thứ 3 giống thư mục vendors ở bên ngoài
* Ngoài ra còn 2 thư mục khác cũng khá quan trọng đó là:
  + Controller\Component: Component là những thành phần mở rộng trong CakePHP, cho phép người sử dụng một cách linh hoạt và có thể tùy biến lại. Trong Cakephp có hỗ trợ một số component mà ta thường dùng là: ACL, mail, time…. chúng được sử dụng trong controller. Ngoài ra chúng ta cũng có thể viết một component của riêng mình để sử dụng.
  + View\Helper: là tập hợp các thư viện, công thức… để có thể sử dụng bất cứ đâu trong view. Cakephp hỗ trợ sẳn các helper như: form, html, ajax, number, session…Cũng như compoment ta cũng có thể viết helper cho riêng mình
* Trong project này chủ yếu code trong các thư mục: app/Controller, app/Model, app/View, app/webroot

## 3.4. Thiết kế sản phẩm

* Sản phẩm hoàn thiện và đã deploy trên web thật
* Có thể vào xem sản phẩm tại: <http://taanhtu95.000webhostapp.com/project3/users/login>
* Username: admin, password: 1111, hoặc: username: toan, password: 1111.





# Tài liệu tham khảo

* Download cakephp framework:

<https://github.com/cakephp/cakephp/tags>

* Cài localhost xampp:

<https://thachpham.com/thu-thuat/cai-dat-localhost-xampp.html>

* Cấu hình xampp cho phpmyadmin:

<https://stackoverflow.com/questions/11096045/mysql-said-documentation-1045-access-denied-for-user-rootlocalhost-usi>

* Cấu trúc thư mục của cakephp:

<http://nongdanit.info/php-mysql/cau-truc-thu-muc-va-cac-quy-uoc-dat-ten-trong-cakephp.html>