1. Download and install

* Xampp 1.8.2: <http://www.apachefriends.org/download.html>

Download về cài đặt và khởi động xampp, bật apache và mysql

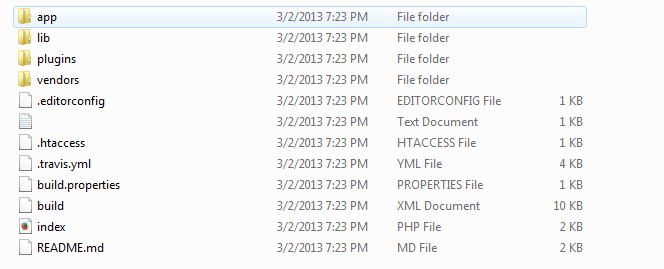
* Cakephp 2.3.1: <https://github.com/cakephp/cakephp/tags>

Download xong giải nén, bỏ vào xampp/htdocs/elearning

Download: DebugKit là 1 plugin debug cho cakephp: <https://github.com/cakephp/debug_kit>

Download boostrap css giao diện: <http://getbootstrap.com/getting-started/#download>

Trong folder elearning sẽ như thế này:



* SublimeText 2: <http://www.sublimetext.com/2>

Download xong cài đặt, khởi động sublimeText

Mở tab Project -> Add Folder to Project...

Chọn đường dẫn tới folder “elearning” vừa rồi

Folder “elearning” vừa được thêm vào bên khung bên trái của sublimeText

Bây giờ ta tiến hành code

1. Config cakephp và Tạo cơ sở dữ liệu

* Mở browser truy cập vào address sau: <http://localhost/elearning>
* Có rất nhiều thông báo lỗi xuất hiện, ta tiến hành fix lần lượt
* Quay lại sublimeText, mở file: elearning/app/Config/core.php
* Line203:

Configure::write('Security.salt', 'DYhG93b0qyJfIxfs2guVoUubWwvniR2G0FgaC9mi')

* Sửa lại thành:

Configure::write('Security.salt', 'DYhG93b0qyJfIxfs2guVoUubWwvniR2G0FgaC9miDAT')

* Line208:
* Configure::write('Security.cipherSeed', '76859309657453542496749683645')
* Sửa lại thành:

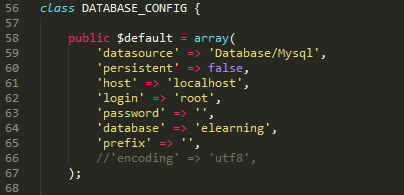
Configure::write('Security.cipherSeed', '768593096574535424967496836451991');

* 2 cái này dùng để mã hóa password theo hàm băm của Cakephp
* Giải nén plugin debug\_kit và đổi tên thành DebugKit, sau đó nhét vào “elearning/app/plugin”
* Giải nén bootstrap và copy vào “elearning/app/webroot”
* Tiếp tục mở file boostrap.php ở dòng 70 bỏ comment đi để Cakephp tự động load plugin của nó
* Refresh browser để thấy sự thay đổi
* Thấy thông báo chưa kết nối được với database. Ta sẽ làm như sau:
* Quay lại sublimeText, rename file:

elearning/app/Config/database.php.default

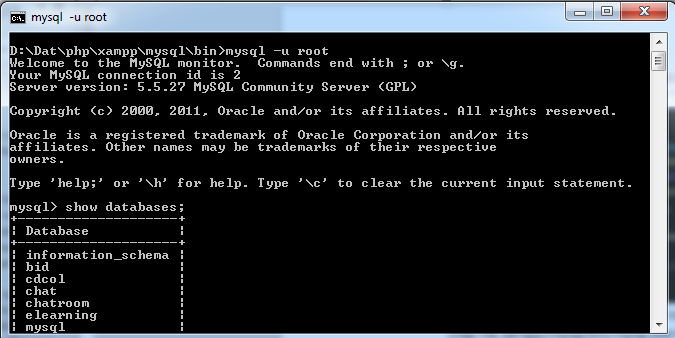
thành “database.php”

* Mở file database.php này ra, cấu hình lại như sau: (ở dòng 62 và 63 là user và password của mysql thường mặc định là “root” và “”)



* Bây giờ ta chạy mysql lên để tạo csdl có tên là “elearning” như sau:

chạy file xampp/mysql/bin/mysql.exe bằng dòng lệnh: mysql –u root –p pass



create database elearning;

use elearning;

|  |
| --- |
| CREATE TABLE users (  id INT UNSIGNED AUTO\_INCREMENT PRIMARY KEY,  username VARCHAR(50),  password VARCHAR(50),  created DATETIME DEFAULT NULL,  modified DATETIME DEFAULT NULL  ); |

|  |
| --- |
| insert into users(username,password,created,modified) values("quydat","123456",now(), now());  insert into users(username,password,created,modified) values("vankhoi","123456",now(), now());  insert into users(username,password,created,modified) values("trunghieu","123456",now(), now()); |

* Refresh browser để thấy thông báo toàn màu xanh. Tức là ok.



* Quy tắc đặt tên cơ sở dữ liệu:

bảng “users” sẽ tương ứng với Model “User” trong Cakephp

1. Code (tạo Model, Controller, View)

* Tạo Model User.php

|  |
| --- |
| <?php  class User extends AppModel {    }  ?> |

* Tạo Controller UsersController.php

|  |
| --- |
| <?php  class UsersController extends AppController {  public function index(){  $usersList = $this->User->find('all'); //get all user in User table  $this->set('usersList', $usersList); //gửi đến View/Users/index.ctp  }  public function addUser(){  if($this->request->is('post')){ //nếu có yêu cầu Add 1 User từ phía View/User/add\_user.ctp  $this->User->create(); //Tạo 1 bảng ghi mới trong User table  if($this->User->save($this->data)){ //Nếu newUser đã được lưu thành công vào csdl thì chuyển tới trang View/Users/index.ctp  $this->redirect(array('controller' => 'users', 'action' => 'index'));  }  else{  //thông báo lỗi ko tạo thành công tài khoản mới  }  }  }  public function editUser($id = null) {  //nếu có yêu cầu editUser từ phía View với type là post  if ($this->request->is('post')) {  $this->User->id = $id;  if ($this->User->save($this->request->data)) {  return $this->redirect(array('action' => 'index'));  }  }  //Nếu không có dữ liệu gửi lên  if (!$this->request->data) {  $user = $this->User->findById($id); //tìm theo id  $this->request->data = $user;  $this->set('user', $user); //gửi tới View/User/edit\_user.ctp  }  }  public function deleteUser($id) {  if ($this->User->delete($id)) { //gọi câu lệnh này của cake sẽ xóa bỏ trong csdl với id tương ứng  return $this->redirect(array('action' => 'index'));  }  }  } |

* Tạo View:

Trong View, tạo folder Users

Trong Users, tạo 3 file ctp tương ứng với 3 function trong UsersController là index(), addUser(), editUser():

index.ctp

|  |
| --- |
| <h1>User manager</h1>  <p><?php echo $this->Html->link("Add User", array('action' => 'addUser')); ?></p>  <table>  <tr>  <th>id</th>  <th>username</th>  <th>password</th>  <th>created</th>  <th>modified</th>  <th>function</th>  </tr>  <!-- $user配列をループして、投稿記事の情報を表示 -->  <?php foreach ($usersList as $user): ?>  <tr>  <td><?php echo $user['User']['id']; ?></td>  <td>  <?php echo $user['User']['username']; ?>  </td>  <td>  <?php echo $user['User']['password']; ?>  </td>  <td>  <?php echo $user['User']['created']; ?>  </td>  <td>  <?php echo $user['User']['modified']; ?>  </td>  <td>  <?php echo $this->Html->link('Edit', array('action' => 'editUser', $user['User']['id'])); ?>  <?php echo $this->Html->link('Del', array('action' => 'deleteUser', $user['User']['id'])); ?>  </td>  </tr>  <?php endforeach; ?>  </table> |

add\_user.ctp

|  |
| --- |
| <h1>Add User</h1>  <?php  echo $this->Form->create('User', array('action'=>'addUser'));  echo $this->Form->input('username');  echo $this->Form->input('password');  echo $this->Html->link("Back", array('controller' => 'users', 'action' => 'index'));  echo $this->Form->end("Add");  ?> |

edit\_user.ctp

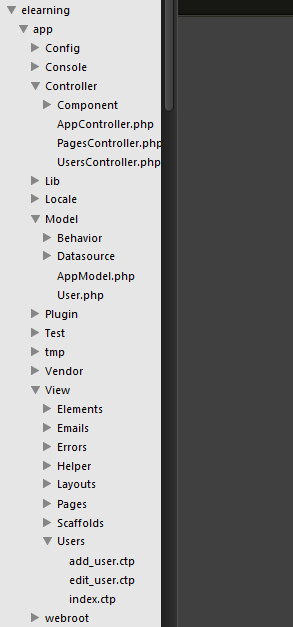
|  |
| --- |
| <h1>Edit User</h1>  <?php  echo $this->Form->create('User', array('action'=>'editUser'));  echo $this->Form->input('username', array('value' => $user['User']['username']));  echo $this->Form->input('password', array('value' => $user['User']['password']));  echo $this->Form->input('id', array('type' => 'hidden'));  echo $this->Html->link("Back", array('controller' => 'users', 'action' => 'index'));  echo $this->Form->end('Save');  ?> |

Function deleteUser() thì không cần giao diện

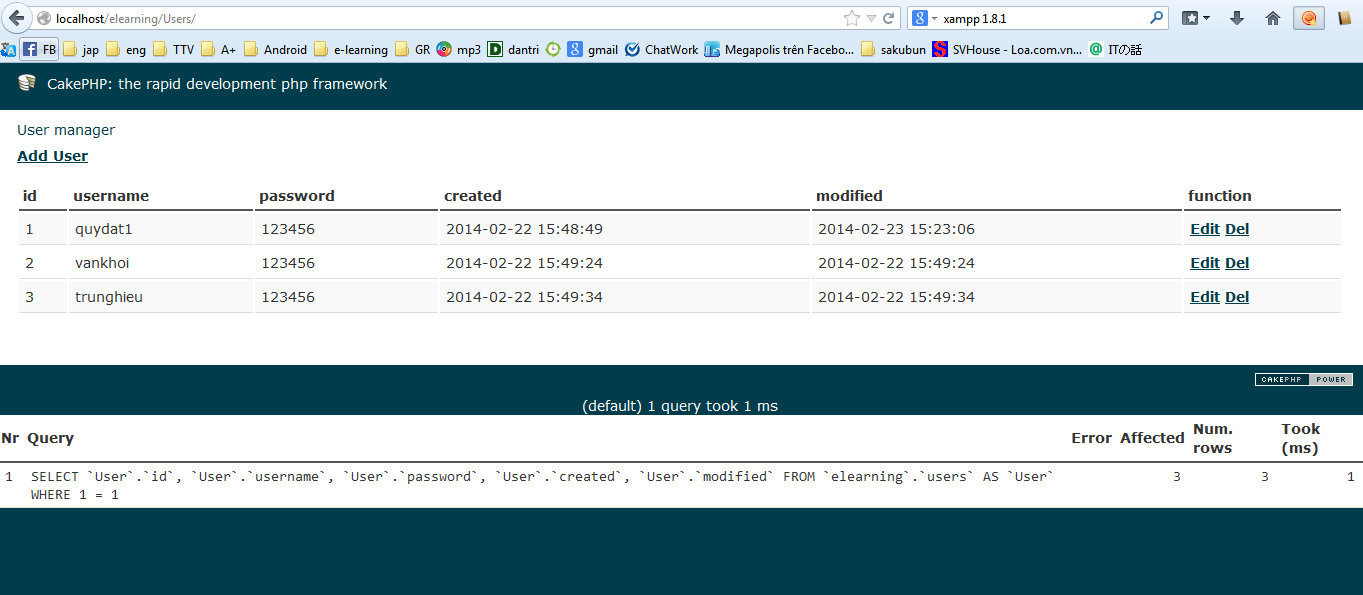
Như ta thấy function addUser() sẽ truyền và nhận dữ liệu từ/đến giao diện add\_user.ctp

Đấy cũng là 1 quy tắc đặt tên trong cakephp 2.3.1

OK. Cấu trúc project của chúng ta sẽ như sau:



OK. Bây giờ xem kết quả: <http://localhost/elearning/Users/>



1. Git tutorial

Download cái này: <http://msysgit.googlecode.com/files/msysGit-fullinstall-1.9.0-preview20140217.exe>

Cài đặt xong thì thiết lập biến môi trường cho window như sau:

Edit the **System Variable** called **PATH**.

Thêm vào dòng:

C:\msysgit\bin\;C:\msysgit\mingw\bin\

Ok.Bây giờ ta có thể sử dụng lệnh “git” trong cmd của window. Mở cmd và gõ thử “git” để xem kết quả.

* Đăng ký tài khoản github: <https://github.com/>
* Login vào <https://github.com/>
* Tạo mới 1 repository với tên là “elearning”
* Vào thư mục elearning trong xampp/htdocs/elearning
* Nhấn “Shift + right mouse” -> chọn “Open command here”
* Gõ các lệnh sau:

|  |
| --- |
| git init  git add \*  git config --global  git commit –m “first commit”  git remote add origin <https://github.com/ngquydat/elearning.git>  git pull  git push –u origin master  git config credential.helper store //ko yêu cầu username và password nhiều lần  touch .gitkeep //commit empty folder upto github  git config –global push.default matching //config push default is master  git reset --hard HEAD^^ //quay trở lại thời điểm trước nữa  git reset --hard HEAD^ //quay trở lại thời điểm trước đó  git reset --hard HEAD //quay trở lại thời điểm commit gần nhất  git remote add brand-name url //add thêm 1 remote  git remote –v //liệt kê tất cả những remote mà repo đó kết nối  git remote rename remote-source-name remote-destination-name  //đổi tên 1 remote  git remote rm remote-destination-name //xóa 1 remote  git branch –v //liệt kê các nhánh hiện có trong repo  git branch branch-name //tạo 1 nhánh mới  git checkout branch-name //chuyển qua nhánh mới  git checkout A //chuyển qua nhánh A  git merge B //hợp nhất nhánh B vào nhánh A  git pull remote-name branch-name //lấy dữ liệu về  git push remote-name branch-name //đẩy dữ liệu lên server  conflic  <<<<<<<  // những dòng code mà tôi đã thay đổi  =======  // những dòng code có sự thay đổi trên server  >>>>>>> |