[Cakephp] Bài 1 – Giới thiệu

Đây là bài đầu tiên trong Series về Cakephp, mình xin hướng dẫn các bạn những gì mình biết về Cakephp, cũng như cùng các bạn học thêm về nó. Đây là một framework mà mình được một người anh hướng dẫn và tự học thêm, nên có thể sẽ còn rất nhiều thứ không được hoàn thiện, nhưng cũng muốn chia sẽ và học hỏi từ các bạn.

Trong bài này mình xin giới thiệu về mục đích, tính năng và các yêu cầu trước khi cài đặt Cakephp Framework.

**Mục đích**

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**

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

**Yêu cầu:** Phải có hệ quản trị cơ sở dữ liệu ở đây mình dùng MYSQL, hệ điều hành windows hoặc linux( mình sử dụng windows). Tóm lại chỉ cần cài đặt [webserver – Xampp](http://nongdanit.info/php-mysql/huong-dan-cai-dat-webserver-xampp-server.html)là chúng ta có thể thoải mái sử dụng.

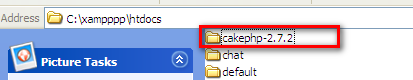
Mình xin kết thúc bài này ở đây, mục đích bài này chỉ giới thiệu thôi, nên các bạn có thể đọc cho biết là được, nhưng nếu được thì cũng nên nhớ tính năng của nó, phòng trường hợp đi phỏng vấn xin việc (mình cũng có gặp mấy lần ^^). Cảm ơn các bạn đã quan tâm.

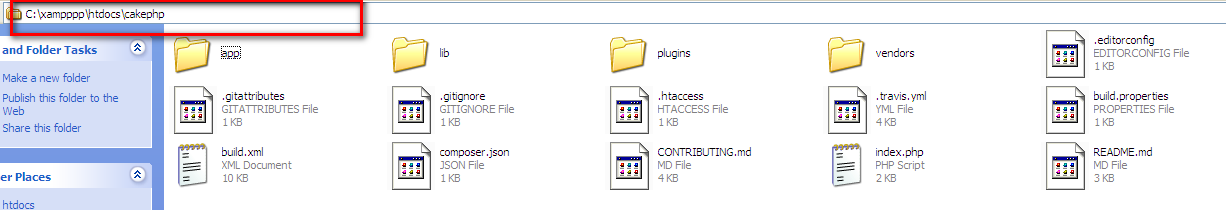
[Cakephp] Bài 2 – Hướng dẫn cài đặt

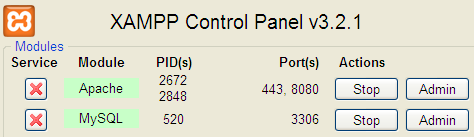
Tiếp theo Series Cakephp Framework là bài hướng dẫn cài đặt, hiện tại CakePHP đã có phiên bản 3.0 trở lên, nhưng mình xin hướng dẫn bản CakePHP 2.7.2 các bạn có thể download [tại đây](https://github.com/cakephp/cakephp/tags), và cần một webserver XAMPP tích hợp Mysql và Apache hướng dẫn cài đặt [tại đây](http://nongdanit.info/php-mysql/huong-dan-cai-dat-webserver-xampp-server.html).

Tiến hành cài đặt:

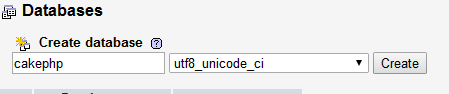
* Giải nén thư mục tải về, rồi copy vào htdocs của xampp.
* Đổi tên lại thành **cakephp**.

**[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_140118.png)**

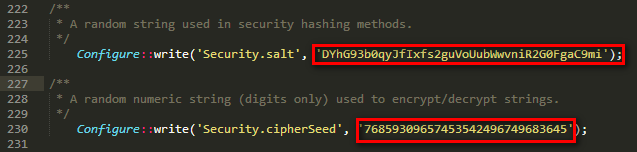
**[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_140736.png)**

**[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_140521.png)**

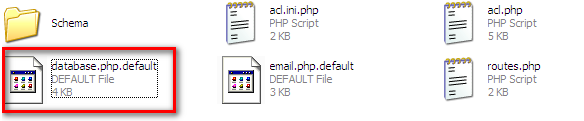
* Mở xampp và chạy 2 dịch vụ Apache, MySQL.
* Truy cập phpmyadmin(*http://localhost:8080/phpmyadmin*) để tạo một database cho project **cakephp**, ở đây mình tạo database có tên là ***cakephp***

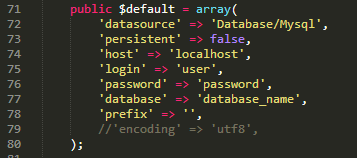
[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_141509.png)

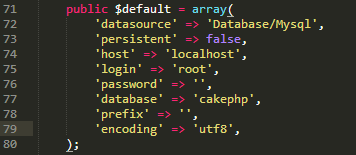
* Mở file *core.php*, trong thư mục project **cakephp** theo đường dẫn C:\xampppp\htdocs\cakephp\app\Config\, tìm kiếm 2 dòng như trong hình sau và thay đổi nó thành một giá trị khác hoặc xóa cũng được, sau khi thay đổi xong thì save lại.

[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_142440.png)

* Tiếp tục tại đường dẫn trên có một file tên là **database.php.default**, đổi tên file này lại thành **database.php**, rồi mở lên và thay đổi các thông tin về database.

**[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_143834.png)**

**[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_144047.png)**

**[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_145218.png)**

*Thay đổi các thông số sau:*

*-login(user database) là: root.*

*-password (pass database) là: ”(rỗng)*

*-database (tên database):****cakephp***

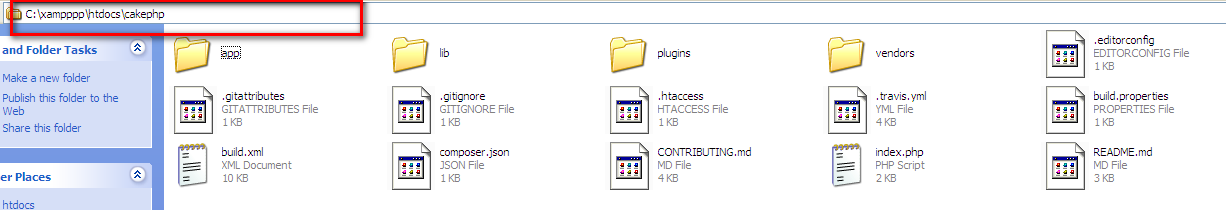
* Truy cập url trình duyệt http://localhost:8080/cakephp, nếu có hình như dưới thì chúc mừng bạn đã cài đặt thành công CakePHP 2.7.2

[Cakephp] Bài 3 – Cấu trúc thư mục và các quy ước đặt tên

You are here: [Nongdanit](http://nongdanit.info/) > [Học PHP & MySQL](http://nongdanit.info/php-mysql) > [Cakephp] Bài 3 – Cấu…

Chúng ta đã cài đặt thành công CakePHP qua bài viết [Hướng dẫn cài đặt CakePHP](http://nongdanit.info/php-mysql/huong-dan-cai-dat-cakephp.html), và trong bài viết này chúng ta sẽ cùng tìm hiểu về cấu trúc thư mục và một số quy ước đặt tên khi tạo file, tạo thư mục, các class… trong Cakephp framework.

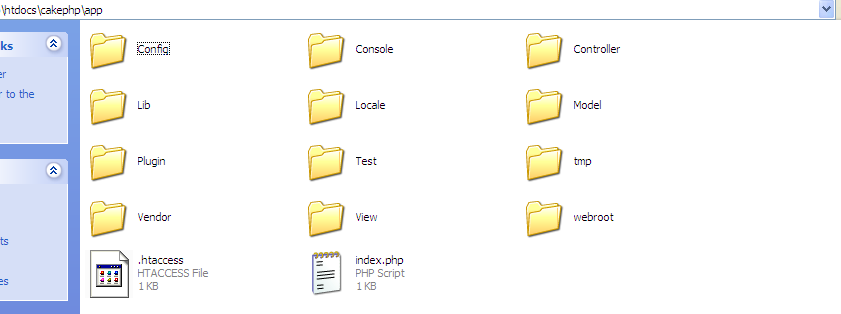
Chúng ta sử dụng lại project **cakephp** đã tạo ở bài [Hướng dẫn cài đặt CakePHP](http://nongdanit.info/php-mysql/huong-dan-cai-dat-cakephp.html)

[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-22_140736.png)

Như trong hình trên ta thấy có các thư mục (các file thì cứ để mặc định)như sau:

* *app: đây là phần quan trong nhất chứa tất cả các mã nguồn của chúng ta(*mục đích chính của bài viết này*)*
* *lib: trong này chứa phần mã nguồn, thư viện của Cakephp. Hiển nhiên chúng ta không nên đụng tới nếu không rõ về Cakephp.*
* *plugins: chứa các thành phần mở rộng của ứng dụng(sẽ nói rõ hơn ở một bài khác).*
* *vendors: là các ứng dụng bên thứ 3 có thể là phpmail, facebook…(sẽ nói rõ hơn ở một bài khác).*

Trong bài này ta tìm hiểu trong thư mục **app:**

[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-26_150410.png)

* ***Config****: thư mục chứa file cấu hình của hệ thống như: database, routes…*
* *Console: hỗ trợ tự động sinh code qua command line(sẽ có một bài riêng).*
* ***Controller****: chứa các controller và C****omponent****(tự tạo) quan trọng là file AppController.php*
* *Lib: thư viện tự tạo của chúng ta*
* *Locale: chứa file ngôn ngữ , sử dụng cho ứng dụng đa ngôn ngữ.*
* ***Model****: chứa file Model và behavor, datasource*
* *Plugin: chứa các gói mở rộng cho ứng dụng.*
* *Test: tên thư mục đã nói lên tất cả rồi^^*
* ***tmp****: đây là nơi chứa các tập tin tạm được tự động sinh ra của ứng dụng.*
* *Vendor: chứa ứng dụng của bên thứ 3 giống thư mục vendors ở bên ngoài.*
* ***View****: chứa các file giao diện hiển thị và****Helper***
* ***webroot****: chứa các tập tin tài liệu về hình ảnh, javascript, css, …*

Trong các thư mục trên các thư mục nào bôi đen nghĩa là quan trọng, chúng ta sẽ sử dụng thường xuyên với chúng khi viết một ứng dụng bằng Cakephp.

Ngoài ra còn 2 thư mục quan trọng nữa là **Component và Helper**

* 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(mình sẽ hướng dẫn một bài riêng).
* 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(sẽ có một bài riêng).

**Quy ước đặt tên trong CakePHP**

1. Tên file, tên class:  
   – Tên file sử dụng chữ thường, viết hoa chữ cái phân cách các từ: CategoriesController.php, DataComponent.php, Category.php, SessionHelper.php.  
   – Tên class viết hoa chữ cái đầu tiên của từ.  
   – Xem cách đặt tên ở bảng sau:

|  |  |  |
| --- | --- | --- |
|  | Tên file | Class |
| Controller | CategoriesController.php | CategoriesController |
| Component | DataComponent.php | DataComponent |
| Model | Category.php | Category |
| Helper | SessionHelper.php | Session |

1. Quy ước về cơ sở dữ liệu và Model:  
   – Tên các class trong Model phải ở dạng số ít và viết hoa chữ cái đầu tiên: Category, User…  
   – Tên các bảng trong database phải là số nhiều: categories, users…  
   – CakePHP không hỗ trợ khóa chính gồm nhiều cột. Nếu muốn kết nối (join) các bảng, hãy truy vấn trực tiếp hoặc thêm một khóa chính thay cho khóa nhiều cột.  
   – Khóa chính mặc định trong CakePHP là cột id, kiểu int, tự tăng. Tuy nhiên, có thể định nghĩa lại khóa chính thông qua biến $primaryKey trong model.  
   – Các khóa ngoại trong các quan hệ có tên mặc định là số ít của tên bảng kèm theo\_id ở cuối cùng. Ví dụ với 2 bảng categories và lessons, bảng lessons sẽ có khóa ngoại đến bảng categories là **category\_id.**
2. Quy ước về Controller:  
   – Tên lớp của controller phải ở dạng số nhiều viết hoa chữ cái đầu tiên, và phía sau cùng là chữ *Controller*.  
   – Ví dụ: ta có file CategoriesController.php thì tên class trong file này là:

|  |  |
| --- | --- |
| 1  2  3  4  5 | <?php    class CategoriesController extends AppController{    //Code here    }  ?> |

1. Quy ước về View: trong view được chia làm 2 cấp độ thư mục và file:  
   – Tên thư mục phải tương ứng với controller.  
   – Tên file của View phải tương ứng với các phương thức trong controller đó.  
   – Đuôi mở rộng của các file trong view là **.ctp**
2. Ví dụ : giả sử trong database ta có một bảng tên ***categories***ta sẽ có các Model, Controller, View như sau  
   – Model: sẽ có file Category.php  
   – Controller: sẽ có file CategoriesController.php  
   – View: sẽ có thư mục Categories  
   – Thêm 2 phương thức vào trong CategoriesController.php nội dung như sau:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | <?php  class CategoriesController extends AppController{     public function menu(){       }     public function index(){       }  }  ?> |

– Trong thư mục Categories chúng ta phải tạo thêm 2 file là index.ctp và menu.ctp

Mình xin kết thúc bài này tại đây, mục đích bài là giới thiệu về cấu trúc thư mục, file cũng như cách đặt tên của chúng trong CakePHP, bài tiếp theo sẽ là một demo đơn giản bằng cách tạo ra các Model, Controller và View

[Cakephp] Bài 4 – Tạo một Model – View – Controller

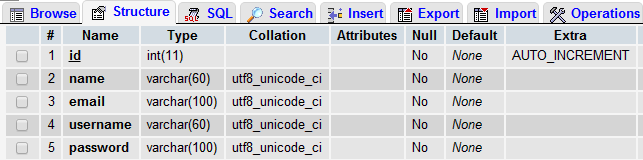
You are here: [Nongdanit](http://nongdanit.info/) > [Học PHP & MySQL](http://nongdanit.info/php-mysql) > [Cakephp] Bài 4 – Tạo…

Ở bài trước chúng ta đã tìm hiểu về [cấu trúc thư mục, các cách thức đặt tên file trong Cakephp](http://nongdanit.info/php-mysql/cau-truc-thu-muc-va-cac-quy-uoc-dat-ten-trong-cakephp.html). Bài này sẽ là một demo đơn giản để tạo ra các file model, view, controller và cách chạy ứng dụng theo tên project/controller/view.

Chúng ta sẽ sử dụng lại database và project **cakephp** đã tạo ra ở [Bài 2 – Hướng dẫn cài đặt CakePHP](http://nongdanit.info/php-mysql/huong-dan-cai-dat-cakephp.html)

* Tên database: **cakephp**
* Tên project: **cakephp**
* Link truy cập: http://localhost:8080/cakephp([đã đổi port](http://nongdanit.info/php-mysql/cach-doi-cong-va-su-dung-ten-mien-ao-localhost-xampp.html))

Bây giờ trong database **cakephp**, ta tiến hành tạo một table có tên là **users** với các columns:

[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-28_151356.png)

Khi ta có một table **users**, thì trong thư mục Model (C:\xampppp\htdocs\cakephp\app\Model) chúng ta sẽ tạo một file User.php, với nội dung:

|  |  |
| --- | --- |
| 1  2  3  4 | <?php  class User extends AppModel {      var $name = "User";//tên của model  } |

Có thể hiểu như sau:

* Tên file và tên class đều được đặt ở dạng số ít, kí tự đầu viết hoa.
* Tên model đặt giống với tên table nhưng ở dạng số ít.
* Ví dụ trên là bảng users, nên tên file Model là User.php và tên class là User

Tiếp theo đến thư mục Controller(C:\xampppp\htdocs\cakephp\app\Controller), ta tạo file tên  là UsersController.php có nội dung:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | <?php  class UsersController extends AppController{       public $name = "Users"; // tên của Controller User       function index(){        //code ...       }  } |

* Tên file và tên class được đặt ở dạng số nhiều viết hoa kí tự đầu theo sau cùng là chữ Controller.
* Tên controller đặt giống với tên table và ở dạng số nhiều.
* Theo ví dụ trên thì ta đặt file là UsersController.php, và class là UsersController

Sau cùng là thư mục view, vì có controller là UsersController nên trong C:\xampppp\htdocs\cakephp\app\View, ta tạo một thư mục Users trong thư mục này tạo file index.ctp(tương ứng với function index() trong controller), file index.ctp có nội dung như sau:

|  |  |
| --- | --- |
| 1  2  3 | <?php  echo "Xin chào các bạn";  ?> |

Đến bước này chúng ta đã tạo xong các file cần thiết để chạy ứng dụng(đương nhiên chúng ta vẫn chưa sử dụng dữ liệu trong model) để chạy ứng dụng chúng ta tiến hành truy cập url: http://localhost:8080/cakephp/users/index, nếu xuất hiện câu “Xin chào các bạn” thì chúng ta đã thành công.

Tiếp theo chúng ta sẽ sử dụng thêm model để hiển thị dữ liệu trong table **users**, đầu tiên chúng ta cần có dữ liệu trong table, bạn có thể sử dụng phpmyadmin để thêm dữ liệu hoặc chạy câu sql sau:

|  |  |
| --- | --- |
| 1  2  3  4  5 | INSERT INTO `users` VALUES ('1', 'Nguyễn Văn A', 'nqsu381@gmail.com', 'nongdanit1', MD5('123456'));  INSERT INTO `users` VALUES ('2', 'Nguyễn Văn B', 'nqsu381@gmail.com', 'nongdanit2', MD5('123456'));  INSERT INTO `users` VALUES ('3', 'Nguyễn Văn C', 'nqsu381@gmail.com', 'nongdanit3', MD5('123456'));  INSERT INTO `users` VALUES ('4', 'Nguyễn Văn D', 'nqsu381@gmail.com', 'nongdanit4', MD5('123456'));  INSERT INTO `users` VALUES ('5', 'Nguyễn Văn E', 'nqsu381@gmail.com', 'nongdanit5', MD5('123456')); |

Thay đổi nội dung file UsersController.php thành đoạn code sau:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | <?php  class UsersController extends AppController{       public $name = "Users"; // tên của Controller User       function index(){        $data = $this->User->find("all");        $this->set("data",$data);       }  } |

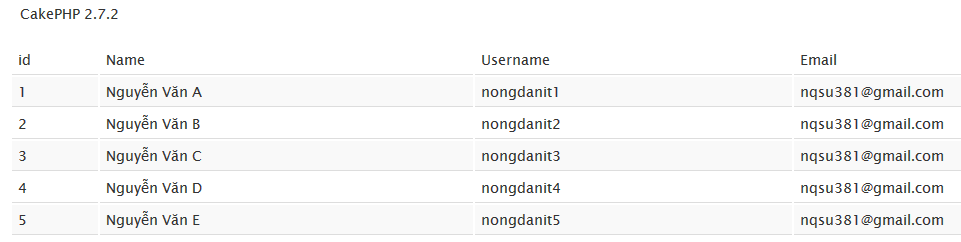
\* Trong đó *$this->User->find(“all”)* là gọi model User để hiển thị tất cả dữ liệu trong table **users**, dữ liệu lấy ra theo dạng mảng được gán cho mảng data các bạn có thể print\_r($data) để xem thành phần của mảng này, find() là cách lấy dữ liệu trong cakephp, chúng ta sẽ tìm hiểu về nó ở một bài khác.

\* *$this->set(“data”,$data)* là cách để gán giá trị mảng data vào biến data và hiển thị chúng ra view(index.ctp). Ở view thì chỉ cần foreach để hiển thị dữ liệu như sau:

Nội dung file index.ctp:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | <?php  if($data==NULL){   echo "<h2>Dada Empty</h2>";  }  else{   echo "<table>   <tr>     <td>id</td>     <td>Name</td>     <td>Username</td>     <td>Email</td>   </tr>";   foreach($data as $item){     echo "<tr>";     echo "<td>".$item['User']['id']."</td>";     echo "<td>".$item['User']['name']."</td>";     echo "<td>".$item['User']['username']."</td>";     echo "<td>".$item['User']['email']."</td>";     echo "</tr>";   }  }  ?> |

Tiến hành chạy lại ứng dụng sẽ được như hình bên dưới

[](http://nongdanit.info/wp-content/uploads/2015/08/2015-08-29_092603.png)

Đến đây thì ứng dụng đầu tiên của chúng ta đã hoàn tất, và bài hướng dẫn này cũng đến đây thôi, qua bài này điều cần là chúng ta biết cách tạo ra các file model, view, controller tương ứng với table được tạo trong database, và biết cách chạy ứng dụng mình theo kiểu **tên project/controller/view**