

PHP Frameworks

Topics

- What is the framework?
- Why framework?
- MVC Framework.
- Benefits and Drawbacks of MVC.
- PHP Frameworks
- Example application using one of framework
- Screenshots
- References

What is a framework ?

For example, in order for a program to get data from a mysql database, it has to undergo a list of actions:

1. Connect to the database server
2. Select a database
3. Query the database
4. Fetch the data
5. Use the Data

A framework may handle steps 1-4 for you, so that your responsibilities are reduced to:

1. Tell the framework to fetch the data
2. Use the data

Sample program in single tire architecture

Connect to database :

```
$db_host = "localhost";  
$db_name = "test";  
$db_username = "root";  
$db_password = "root";
```

```
$conn = mysql_connect($db_host,$db_username,$db_password) or die("Could not connect to Server" .mysql_error());  
mysql_select_db($db_name) or die("Could not connect to Database" .mysql_error());
```

```
<html>  
<head>  
</head>  
<body>
```

Edit :

```
<form:>  
    get code from databse and display at values of input boxes  
</form>
```

Display :

```
<table>
```

```
<?php  
$query="iselect * from users ";
```

```
$result = Mysql_query($query);
```

```
While($get = mysql_fetch_assoc($result))
```

```
{  
    ?>  
    <tr><td><?php echo $get['name']?></td></tr>  
<?php  
}  
?>  
</table>
```

Same program using two tire architecture

At the PHP file :

```
<?php

require 'libs/Smarty.class.php';
include "includes/functions.php";

$smarty = new Smarty;

$smarty->assign("title","Get data from Database");
$smarty->assign("keywords",get_data, database);
$smarty->assign("description","Get data from database process ")

$query= "select * from users ";
$result = Mysql_query("$query");
$getdata= mysql_fetch_array($result);
$smarty->assign("data",$data);
$smarty->display('userss.tpl');
```

At a .tpl file

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>

<meta http-equiv="content-type" content="text/html;
charset=utf-8" />
<title>{$title}</title>

<meta name="keywords" content="{ $keywords}" />
<meta name="description" content="{ $description}" />

<<h1 class="title">Smarty !</h1>

<ul class="list" >

{section name=rows loop=$data}
{$data[rows]}

{/section}

</dody>
</html>
```

Why framework?

Developing an application from scratch is easy than maintaining it . So after the development the good thing is that its easy to maintain and add more features.

Its easy to handle the complex problems easily.

1. Clean urls (SEO friendly urls)

2. We can standardization!

3 .Security

4. Extensibility

5. Fast Loading

6. Reusable of code

7. increase flexibility

MVC Framework

-> MVC is a method of separating the user interface of an application from its Domain Logic.

-> MVC stands for Model, View, and Controller. It aims to separate your app in these 3 components:

Model:

- help you retrieve, insert, and update information in your database.

View:

- the information that is being presented to a user.

Controller:

- an intermediary between the Model, the View
- any other resources needed to process the HTTP request and generate a web

page

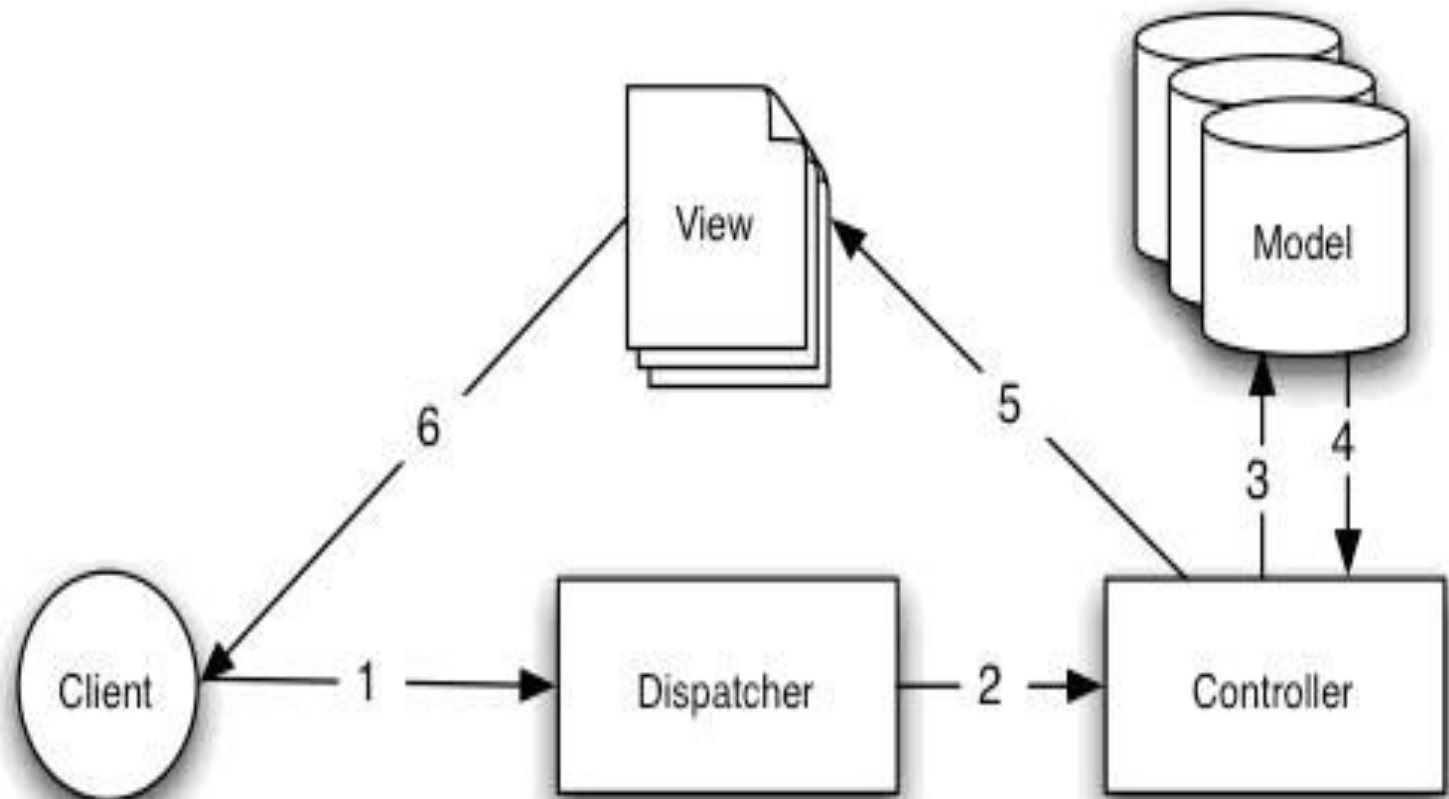
Benefits of using MVC

- Easier to Debug as we have multiple levels properly written in the application.
- Substitutable user interface :
- User interface components:
- Multiple simultaneous views of the same model
- Synchronized views
- Easier user interface changes
- Easier testing
- Development of the application becomes fast.
- Easy for multiple developers to collaborate and work together.

Drawbacks of using MVC

It is hard to understand the MVC architecture.
Must have strict rules on methods.




















MVC



Top 10 frameworks

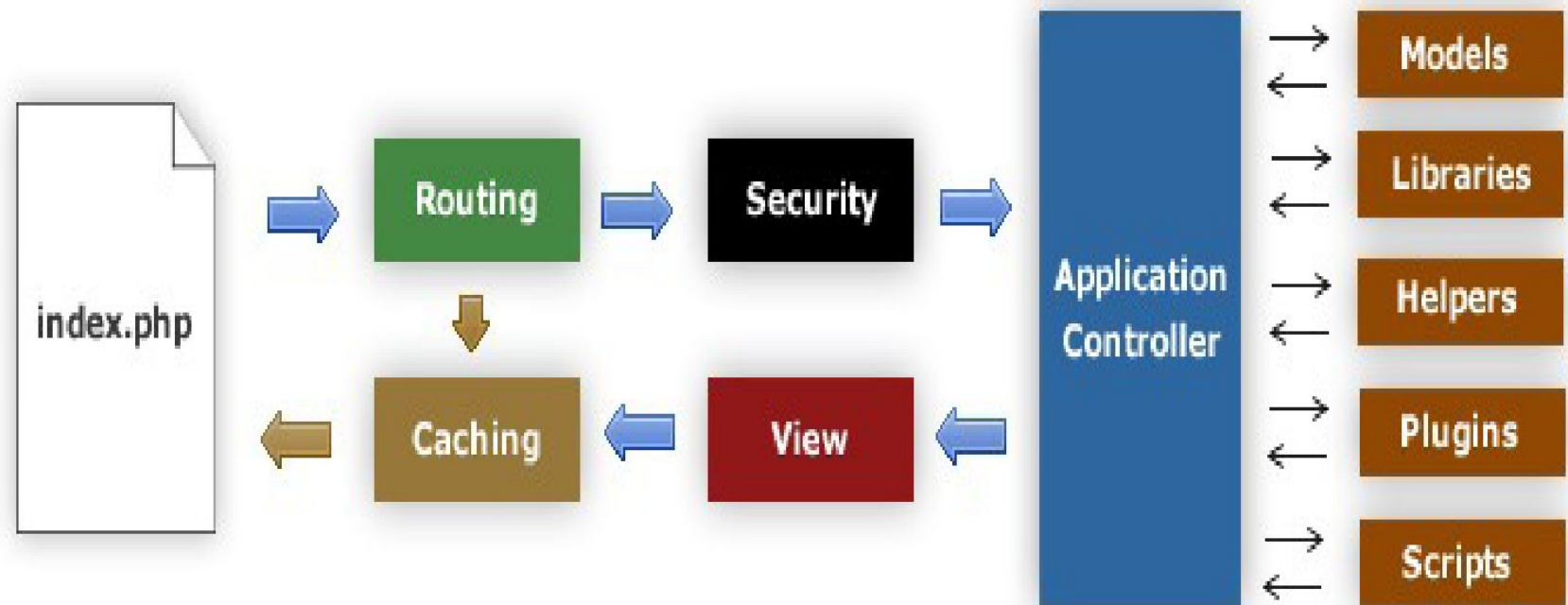
- **CodeIgniter**
- **Yii**
- **CakePHP**
- **Zend**
- **Symfony**
- **PHPDevShell**
- **Prado**
- **Akelos**
- **Zoop**
- **QPHP**

Comparison of frameworks

PHP Framework	PHP4	PHP5	MVC	Multiple DB's	ORM	DB Objects	Templates	Caching	Validation	Ajax	Auth Module	Modules	EDP
Akelos 	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
ash.MVC 	-	✓	✓	-	-	✓	✓	-	✓	-	✓	✓	-
CakePHP 	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	-
CodeIgniter 	✓	✓	✓	✓	-	✓	✓	✓	✓	-	-	-	-
DIY 	-	✓	✓	-	✓	✓	✓	✓	-	✓	-	-	-
eZ Components 	-	✓	-	✓	-	✓	✓	✓	✓	-	-	-	-
Fusebox 	✓	✓	✓	✓	-	-	-	✓	-	✓	-	✓	-
PHP on TRAX 	-	✓	✓	✓	✓	✓	-	-	✓	✓	-	✓	-
PHPDevShell 	-	✓	-	-	-	-	✓	-	-	✓	✓	✓	-
PhpOpenbiz 	-	✓	✓	✓	✓	✓	✓	-	✓	✓	✓	-	-
Prado 	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
QPHP 	✓	✓	✓	✓	-	✓	✓	-	✓	✓	✓	✓	✓
Seagull 	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	-
Symfony 	-	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	-
WACT 	✓	✓	✓	✓	-	✓	✓	-	✓	-	-	✓	-
WASP 	-	✓	✓	-	-	✓	✓	-	✓	✓	✓	✓	-
Yii 	-	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Zend 	-	✓	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	-
ZooP 	✓	✓	✓	✓	-	✓	✓	✓	✓	✓	✓	-	-

- **MVC:** Indicates whether the framework comes with inbuilt support for a Model-View-Controller setup.
- **Multiple DB's:** Indicates whether the framework supports multiple databases without having to change anything.
- **ORM:** Indicates whether the framework supports an object-record mapper, usually an implementation of ActiveRecord.
- **DB Objects:** Indicates whether the framework includes other database objects, like a TableGateWay.
- **Templates:** Indicates whether the framework has an inbuilt template engine.
- **Caching:** Indicates whether the framework includes a caching object or some way other way of caching.
- **Validation:** Indicates whether the framework has an inbuilt validation or filtering component.
- **Ajax:** Indicates whether the framework comes with inbuilt support for Ajax.
- **Auth Module:** Indicates whether the framework has an inbuilt module for handling user authentication.
- **Modules:** Indicates whether the framework has other modules, like an RSS feed parser, PDF module or anything else (useful).
- **EDP:** Event Driven Programming.New!

CodeIgniter Features



Configure the file:

system/application/config/

```
$config['base_url'] = 'http://localhost/';  
$config['index_page'] = '';
```

Default Settings :

```
$config['charset'] = "UTF-8";  
$config['cache_path'] = '';  
$config['permitted_uri_chars'] = 'a-z 0-9~%.:_-';  
$config['log_date_format'] = 'Y-m-d H:i:s';  
$config['global_xss_filtering'] = TRUE;
```

To configure the database:

application\config\database.php

```
$db['default']['hostname'] = ""; // Host Name  
$db['default']['username'] = ""; // User Name  
$db['default']['password'] = ""; // Password  
$db['default']['database'] = ""; // Database Name  
$db['default']['dbdriver'] = ""; // Databse driver.
```

CodeIgniter URLs

example.com/index.php/news/article/my_article

news – Controller

article – class function

my_article - any additional segments

If we add the below contents at .htaccess file

```
DirectoryIndex index.php
RewriteEngine on
RewriteCond $1 !^(index\.php|images|css|js|robots\.txt|favicon\.ico)
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ ./index.php/$1 [L,QSA]
```

Then URLs will change it into.

example.com/news/article/my_article

Controller

```
<?php

class Upload extends Controller {

    function Upload() {
        parent::Controller();
        /* $this->load->helper('form');*/
    }

    function index() {

        if ($this->session->userdata('logged_in') != TRUE)
        {

            redirect(base_url(). 'user/login');

        }
        else
        {
            //echo $this->session->userdata('name');
            $data['login']=$this->session->userdata('name') ;

        }

        $this->load->database();
        $data['title']="Welcome to CodeIgniter Upload Images";
        $this->load->view('header',$data);
        $this->load->view('upload_form');
        $this->load->view('footer');
    }

    function _createThumbnail($fileName) {
        $config['image_library'] = 'gd2';
        $config['source_image'] = 'uploads/' . $fileName;
        $config['create_thumb'] = TRUE;
        $config['maintain_ratio'] = TRUE;
        $config['width'] = 75;
        $config['height'] = 75;
        $this->load->library('image_lib', $config);
        if(!$this->image_lib->resize()) echo $this->image_lib->display_errors();
    }
}
```



```

function list_images() {
    $this->load->database();
    $this->load->library('pagination');
    $config['total_rows'] = $this->db->count_all('code_image');
    $config['per_page'] = '3';
    $config['full_tag_open'] = '<p>';
    $config['full_tag_close'] = '</p>';
    $config['base_url'] = base_url(). 'upload/list_images/';

    $this->pagination->initialize($config);
    //echo base_url();
    $this->load->model('code_image');
    $data['images'] = $this->code_image->get_images($config['per_page'],$this->uri->segment(3));

    // This gives us anchor() - see the view at the end

    $data1['login']=$this->session->userdata('name') ;

    $data1['title']="List of images in the Website";
    $this->load->view('header',$data1);
    $this->load->helper('url');
    $this->load->view('list_images', $data);
    $this->load->view('footer');
}

```

```

function view_image($image_id)
{
    $this->load->database();
    $this->load->model('code_image');
    $data['image'] = $this->code_image->get_image($image_id);
    $data1['login']=$this->session->userdata('name') ;
    $data1['title']="List of images in the Website";
    $this->load->view('header',$data1);
    $this->load->view('view_image', $data);
    $this->load->view('footer');
}

```

Model

```
<?php
class Code_image extends Model {
    function get_images($num, $offset) {

        $query = $this->db->get('code_image', $num, $offset);
        //$query = $this->db->get('code_image');
        foreach ($query->result_array() as $row)
        {
            $result[] = $row;
        }
        return $result;
    }

    function get_image($image_id) {
        $query = $this->db->where('image_id', $image_id)->get('code_image');
        $result = $query->row_array();
        return $result;
    }
}
?>
```

Views

```
<table width="900" cellpadding="0" cellspacing="0" border="0" class="content" >
<tr><td><p>List Out Photos </p></td></tr>
<tr>
<td width="400">
<table align="center">
<?php foreach ($images as $image): ?>
    <tr><td colspan="2" style=" border-top:1px solid #669966;">&nbsp;</td></tr>
    <tr >
        <td width="200"></td>
        <td width="200"><?=<anchor( base_url(). 'upload/view_image/' . $image['image_id'], 'View')?></td>
    </tr>
    <tr><td colspan="2" style=" border-bottom:1px solid #669966;">&nbsp;</td></tr>
<?php endforeach; ?>
<tr><td colspan="2" style=" border-bottom:1px solid #669966;" align="right">
<?php echo $this->pagination->create_links(); ?>
&nbsp;</td></tr>

</table>
</td>
</tr>
</table>
```

Particular Image

```
<table width="900" cellpadding="0" cellspacing="0" border="0" class="content" >
<tr><td><p>View Image </p></td></tr>
<tr>
<td width="400">
<table align="center">
    <tr><td colspan="2" style=" border-top:1px solid #669966;">&nbsp;</td></tr>
    <tr >
        <td width="400"></td>

    </tr>
    <tr><td colspan="2" style=" border-bottom:1px solid #669966;">&nbsp;</td></tr>
</table>
</td>
</tr>
</table>
```

Upload a image

```
<table width="900" height="200" cellpadding="0" cellspacing="0" border="0" class="content" >
```

```
<tr><td colspan="3" align="left"><h2>Upload an Image </h2></td></tr>
```

```
<?php echo form_open_multipart(base_url().'upload/doUpload'); ?>
```

```
<tr valign="top"><td colspan="3" align="center">
```

```
<table cellpadding="0" cellspacing="2" border="0">
```

```
<tr>
```

```
<td>&nbsp;</td>
```

```
<td>Image Name: </td>
```

```
<td><input type="file" name="userfile" /></td>
```

```
<td>&nbsp;</td>
```

```
<td><input type="image" src="<?=base_url()?>images/upload.png" value="Login" /></td>
```

```
</tr>
```

```
</table>
```

```
</td>
```

```
</tr>
```

```
<?php echo form_close(); ?>
```

```
</table>
```

List of images :

URL : http://localhost/codeigniter/upload/list_images



CodeIgniter Applications

[Home](#) | [Upload](#) | [List](#) | [User](#)

List Out Photos

	View
	View
	View

2 ≥

Particular Image

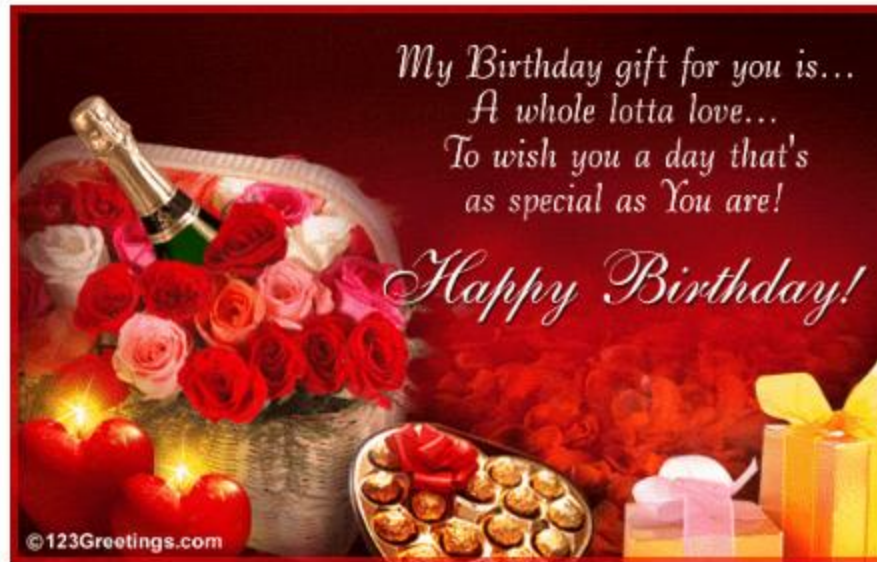
URL : http://localhost/codeigniter/upload/view_image/1



CodeIgniter Applications

[Home](#) | [Upload](#) | [List](#) | [User](#)

[View Image](#)



Upload an image



CodeIgniter Applications

[Home](#)

[Upload](#)

[List](#)

[Logout](#)

Upload an Image

Browse...

Upload

Upload success page



CodeIgniter Applications

[Home](#) | [Upload](#) | [List](#) | [Logout](#)

Congratulations. Your photo has been uploaded successfully.

File Name: logo.jpg

File Size: 29.72

File Extension: .jpg



stimulate solutions
social media marketing



References

- PHP framework comparison benchmarks
- <http://avnetlabs.com/php/php-framework-comparison-benchmarks>
- Why Should we use a PHP Framework?
- <http://www.ashesh.com.np/why-should-we-use-a-php-framework/>
- CakePHP official site
- <http://cakephp.org/>
- CodeIgniter Site
- <http://codeigniter.com>
- PHP frameworks lists
- <http://www.phpframeworks.com/>
- http://www.phpwact.org/pattern/model_view_controller