Creating Skeleton with Basic Routing System

This document defines the homework assignments for ["PHP Web Development Basic" Course @ Software University](https://softuni.bg/trainings/1746/php-web-developmentbasics-september-2017).

**Contents**

[1 Create a new file - .htaccess 2](#_Toc527998328)

[2 Create a new file – config.php 2](#_Toc527998329)

[3 Create a new file – functions.php 3](#_Toc527998330)

[3.1 Parsing Request 3](#_Toc527998331)

[3.2 Processing the parsed request 3](#_Toc527998332)

[3.3 Loading controllers 4](#_Toc527998333)

[4 Create a new file – index.php 4](#_Toc527998334)

[5 Create a new directory – controllers 5](#_Toc527998335)

[5.1 Create a new file – BaseController.php 5](#_Toc527998336)

[5.2 Create a new file – **HomeController.php** 7](#_Toc527998337)

[5.3 Create a new file - **UserController.php** 8](#_Toc527998338)

[6 Create a new directory: models 8](#_Toc527998339)

[6.1 Create a new file – **BaseModel.php** 8](#_Toc527998340)

[6.2 Create a new file – **HomeModel.php** 8](#_Toc527998341)

[6.3 Create a new file – **UsersModel.php** 9](#_Toc527998342)

[7 Create a new directory – views 9](#_Toc527998343)

[7.1 Create a new directory inside views named **\_layout** 9](#_Toc527998344)

[7.1.1 Create a new file – **footer.php** 9](#_Toc527998345)

[7.1.2 Create a new file – **header.php** 9](#_Toc527998346)

[7.1.3 Create a new file: fill-posted-fields.php 10](#_Toc527998347)

[7.1.4 Create a new file: show-validation-errors.php 10](#_Toc527998348)

[8 Create a new directory inside views named home 11](#_Toc527998349)

[8.1 Create a new file – **index.php** 11](#_Toc527998350)

[9 Create a new directory inside views named users 11](#_Toc527998351)

[9.1 Create a new file – **login.php** 11](#_Toc527998352)

[9.2 Create a new file – **register.php** 11](#_Toc527998353)

[10 Output 11](#_Toc527998354)

[10.1 Index.php 11](#_Toc527998355)

[10.2 Login 12](#_Toc527998356)

[10.3 Register 12](#_Toc527998357)

За начало си направете проект с име blog, като root папката да е xampp\htdocs.



След това добавете папките db и content в папка blog. След това следвайте стъпките за да създадете Basic Routing System.

# Create a new file - .htaccess

<**IfModule mod\_rewrite.c**>  
 **RewriteEngine On  
 RewriteCond %{REQUEST\_URI} !^(/[a-zA-Z0-9\_-]+)?/content/.\*$  
 RewriteRule ^ index.php**</**IfModule**>

# Create a new file – config.php

*define*(**'APP\_ROOT'**, **'/blog'**);  
  
*define*(**'DEFAULT\_CONTROLLER'**, **'home'**);  
*define*(**'DEFAULT\_ACTION'**, **'index'**);  
  
*define*(**'DB\_HOST'**, **'localhost'**);  
*define*(**'DB\_USER'**, **'root'**);  
*define*(**'DB\_PASS'**, **''**);  
*define*(**'DB\_NAME'**, **'blog'**);

# Create a new file – functions.php

## Parsing Request

*/\*\*  
 \* Parse REQUEST\_URI to extract request parts: {controller}/{action}/{params}  
 \** ***@return*** *array holding: 'controller' (string), 'action' (string) and  
 \* 'params' (array of strings). When controller / action is not available,  
 \* returns the default controller 'home' / action 'index'.  
 \*/*

**function** parseRequest() : **array**{  
 $requestPath = $\_SERVER[**'REQUEST\_URI'**]; *// holds /blog/home/index or /home/index* **if** (*substr*($requestPath, 0, *strlen*(***APP\_ROOT*** . **'/'**)) != ***APP\_ROOT*** . **'/'**) {  
 **die**(**'APP\_ROOT is incorrectly defined in config.php. Use "" or "/blog".'**);  
 }  
 $requestPath = *substr*($requestPath, *strlen*(***APP\_ROOT***)); *// remove APP\_ROOT prefix* $requestParts = *explode*(**'/'**, $requestPath);  
  
 *// Extract the controller from {controller}/{action}/{params}* $controller = ***DEFAULT\_CONTROLLER***;  
 **if** (*count*($requestParts) >= 2 && $requestParts[1] != **''**) {  
 $controller = $requestParts[1];  
 }  
  
 *// Extract the action from {controller}/{action}/{params}* $action = ***DEFAULT\_ACTION***;  
 **if** (*count*($requestParts) >= 3 && $requestParts[2] != **''**) {  
 $action = $requestParts[2];  
 }  
  
 *// Extract the action parameters from {controller}/{action}/{params}* $params = *array\_splice*($requestParts, 3);  
  
 $requestParsed = [  
 **'controller'** => $controller,  
 **'action'** => $action,  
 **'params'** => $params];  
 **return** $requestParsed;  
}

## Processing the parsed request

*/\*\*  
 \* Processes the parsed request /{controller}/{action}/{params}. Creates and  
 \* initializes the requested controller class, invokes the requested action  
 \* and renders its view.  
 \** ***@param*** *array $requestParsed associative array holding: 'controller' (string),  
 \* 'action' (string) and 'params' (array of strings).  
 \*/*

**function** processRequest(**array** $requestParsed)  
{  
 *// Create the controller class* $controller = $requestParsed[**'controller'**];  
 $action = $requestParsed[**'action'**];  
 $controllerClassName = *ucfirst*(*strtolower*($controller)) . **'Controller'**;  
 **if** (*class\_exists*($controllerClassName)) {  
 $controller = **new** $controllerClassName($controller, $action);  
 } **else** {  
 $controllerFileName = **"controllers/"** . $controllerClassName . **'.php'**;  
 **die**(**"Cannot find controller '**$controller**' in class '**$controllerFileName**'"**);  
 }  
  
 *// Invoke the requested action and renders its view (if not rendered)* **if** (*method\_exists*($controller, $action)) {  
 *// Invoke $controller->{$action}($params);* $params = $requestParsed[**'params'**];  
 *call\_user\_func\_array*(**array**($controller, $action), $params);  
 $controller->renderView();  
 } **else** {  
 **die**(**"Cannot find action '**$action**' in controller '**$controllerClassName**'"**);  
 }  
}

## Loading controllers

*/\*\*  
 \* Auto load the controller and model classes from their .php files.  
 \** ***@param*** *string $class\_name <p>the name of the class to load</p>  
 \*/*

**function** \_\_autoload(string $class\_name)  
{  
 **if** (*file\_exists*(**"controllers/**$class\_name**.php"**)) {  
 **include "controllers/**$class\_name**.php"**;  
 }  
 **if** (*file\_exists*(**"models/**$class\_name**.php"**)) {  
 **include "models/**$class\_name**.php"**;  
 }  
}

**function** cutLongText(string $text, int $maxSize=200, bool $htmlEscape = **true**) : string  
{  
 $append = **''**;  
 **if** (*strlen*($text) > $maxSize) {  
 $text = *substr*($text, 0, $maxSize);  
 $append = **'&hellip;'**;  
 }  
 **if** ($htmlEscape)  
 $text = *htmlspecialchars*($text);  
 **return** $text . $append;  
}

# Create a new file – index.php

**<?php  
  
require\_once**(**'config.php'**);  
**require\_once**(**'functions.php'**);  
*session\_start*();  
$requestParsed = parseRequest();  
processRequest($requestParsed);

# Create a new directory – controllers

## Create a new file – BaseController.php

**<?php  
  
abstract class** BaseController  
{  
 **protected $controllerName**;  
 **protected $actionName**;  
 **protected $isViewRendered** = **false**;  
 **protected $isPost** = **false**;  
 **protected $isLoggedIn** = **false**;  
 **protected $title** = **""**;  
 **protected $model**;  
 **protected $validationErrors** = [];  
  
 **function** \_\_construct(string $controllerName, string $actionName)  
 {  
 $this->**controllerName** = $controllerName;  
 $this->**actionName** = $actionName;  
  
 **if** ($\_SERVER[**'REQUEST\_METHOD'**] === **'POST'**) {  
 $this->**isPost** = **true**;  
 }  
  
 $this->**isLoggedIn** = **isset**($\_SESSION[**'username'**]);  
  
 *// Load the default model class for the current controller* $modelClassName = *ucfirst*(*strtolower*($controllerName)) . **'Model'**;  
 **if** (*class\_exists*($modelClassName)) {  
 $this->**model** = **new** $modelClassName();  
 }  
  
 $this->onInit();  
 }  
  
 **public function** onInit()  
 {  
 *// Implement initializing logic in the subclasses* }  
  
 **public function** index()  
 {  
 *// Implement the default action in the subclasses* }  
  
 **public function** renderView(string $viewName = **null**, bool $includeLayout = **true**)  
 {  
 **if** (!$this->**isViewRendered**) {  
 *ob\_start*();  
 **if** ($viewName == **null**) {  
 $viewName = $this->**actionName**;  
 }  
 $viewFileName = **'views/'** . $this->**controllerName** . **'/'** . $viewName . **'.php'**;  
 **include**($viewFileName);  
 $htmlFromView = *ob\_get\_contents*();  
 *ob\_end\_clean*();  
 **if** ($includeLayout) {  
 **include**(**'views/\_layout/header.php'**);  
 }  
 **echo** $htmlFromView;  
 **if** ($includeLayout) {  
 **include**(**'views/\_layout/footer.php'**);  
 }  
 $this->**isViewRendered** = **true**;  
 }  
 }  
  
 **public function** redirectToUrl(string $url)  
 {  
 *header*(**"Location: "** . $url);  
 **die**;  
 }  
  
 **public function** redirect(  
 string $controllerName, string $actionName = **null**, **array** $params = **null**)  
 {  
 $url = ***APP\_ROOT*** . **'/'** . *urlencode*($controllerName);  
 **if** ($actionName != **null**) {  
 $url .= **'/'** . *urlencode*($actionName);  
 }  
 **if** ($params != **null**) {  
 $encodedParams = *array\_map*($params, **'urlencode'**);  
 $url .= *implode*(**'/'**, $encodedParams);  
 }  
 $this->redirectToUrl($url);  
 }  
  
 **public function** authorize() {  
 **if** (! $this->**isLoggedIn**) {  
 $this->addErrorMessage(**"Please login first."**);  
 $this->redirect(**"users"**, **"login"**);  
 }  
 }  
  
 **function** addMessage(string $msg, string $type)  
 {  
 **if** (!**isset**($\_SESSION[**'messages'**])) {  
 $\_SESSION[**'messages'**] = **array**();  
 };  
 *array\_push*($\_SESSION[**'messages'**],  
 **array**(**'text'** => $msg, **'type'** => $type));  
 }  
  
 **function** addInfoMessage(string $infoMsg)  
 {  
 $this->addMessage($infoMsg, **'info'**);  
 }  
  
 **function** addErrorMessage(string $errorMsg)  
 {  
 $this->addMessage($errorMsg, **'error'**);  
 }  
  
 **function** setValidationError(string $fieldName, string $errorMsg)  
 {  
 $this->**validationErrors**[$fieldName] = $errorMsg;  
 }  
  
 **function** formValid() : bool  
 {  
 **return** *count*($this->**validationErrors**) == 0;  
 }  
}

## Create a new file – **HomeController.php**

<?php

**class** HomeController **extends** BaseController  
{  
 **function** index() {  
 *//* ***TODO: Load posts to be displayed here ...*** }  
   
 **function** view($id) {  
 *//* ***TODO: Load a post to be displayed here ...*** }  
}

## Create a new file - **UsersController.php**

**<?php  
  
class** UsersController **extends** BaseController  
{  
 **public function** register()  
 {  
 *//* ***TODO: your user registration functionality will come here ...*** }  
  
 **public function** login()  
 {  
 *//* ***TODO: your user login functionality will come here ...*** }  
  
 **public function** logout()  
 {  
 *//* ***TODO: your user logout functionality will come here ...*** }  
}

# Create a new directory: models

## Create a new file – **BaseModel.php**

**<?php  
  
abstract class** BaseModel  
{  
 **protected static** *$db*;  
  
 **public function** \_\_construct()  
 {  
 **if** (**self**::*$db* == **null**) {  
 **self**::*$db* = **new** mysqli(***DB\_HOST***, ***DB\_USER***, ***DB\_PASS***, ***DB\_NAME***);  
 **self**::*$db*->set\_charset(**"utf8"**);  
 **if** (**self**::*$db*->**connect\_errno**) {  
 **die**(**'Cannot connect to database'**);  
 }  
 }  
 }  
}

## Create a new file – **HomeModel.php**

**<?php  
  
class** HomeModel **extends** BaseModel  
{  
 *//* ***TODO: your database access functions for the home page will come here ...***}

## Create a new file – **UsersModel.php**

**<?php  
  
class** UsersModel **extends** BaseModel  
{  
 *//* ***TODO: your database access functions for the user register / login will come here ...***}

# Create a new directory – views

## Create a new directory inside views named **\_layout**

### Create a new file – **footer.php**

<**footer**>  
 (c) Blog System, 2018  
 </**footer**>  
  
 **<?php require\_once**(**'fill-posted-fields.php'**); **?>  
  
 <?php require\_once**(**'show-validation-errors.php'**); **?>**</**body**>  
  
</**html**>

### Create a new file – **header.php**

<!DOCTYPE **html**>  
<**html**>  
  
<**head**>  
 <**link rel="stylesheet" href="<?=*APP\_ROOT*?>/content/styles.css"** />  
 <**link rel="icon" href="<?=*APP\_ROOT*?>/content/images/favicon.ico"** />  
 <**script src="<?=*APP\_ROOT*?>/content/scripts/jquery-3.0.0.min.js"**></**script**>  
 <**script src="<?=*APP\_ROOT*?>/content/scripts/blog-scripts.js"**></**script**>  
 <**title**>**<?php if** (**isset**($this->**title**)) **echo** *htmlspecialchars*($this->**title**) **?>**</**title**>  
</**head**>  
  
<**body**>  
<**header**>  
 <**a href="<?=*APP\_ROOT*?>"**><**img src="<?=*APP\_ROOT*?>/content/images/site-logo.png"**></**a**>  
 <**a href="<?=*APP\_ROOT*?>/"**>Home</**a**>  
 **<?php if** ($this->**isLoggedIn**) : **?>** <**a href="<?=*APP\_ROOT*?>/posts"**>Posts</**a**>  
 <**a href="<?=*APP\_ROOT*?>/posts/create"**>Create Post</**a**>  
 <**a href="<?=*APP\_ROOT*?>/users"**>Users</**a**>  
 **<?php else**: **?>** <**a href="<?=*APP\_ROOT*?>/users/login"**>Login</**a**>  
 <**a href="<?=*APP\_ROOT*?>/users/register"**>Register</**a**>  
 **<?php endif**; **?>  
 <?php if** ($this->**isLoggedIn**) : **?>** <**div id="logged-in-info"**>  
 <**span**>Hello, <**b**>**<?=***htmlspecialchars*($\_SESSION[**'username'**])**?>**</**b**></**span**>  
 <**form method="post" action="<?=*APP\_ROOT*?>/users/logout"**>  
 <**input type="submit" value="Logout"**/>  
 </**form**>  
 </**div**>  
 **<?php endif**; **?>**</**header**>  
  
**<?php require\_once**(**'show-notify-messages.php'**); **?>**

### Create a new file: fill-posted-fields.php

<**script**>  
**<?php  
 foreach** ($\_POST **as** $fieldName => $fieldValue) {  
 $fieldJson = *json\_encode*($fieldName);  
 $valueJson = *json\_encode*($fieldValue);  
 **echo "setFieldValue(**$fieldJson**,** $valueJson**);\n"**;  
 }  
**?>**</**script**>

7.1.4. Create a new file: show-notify-messages.php

**<?php  
if** (**isset**($\_SESSION[**'messages'**])) {  
 **echo '<ul id="messages">'**;  
 **foreach** ($\_SESSION[**'messages'**] **as** $msg) {  
 **echo '<li class="'** . $msg[**'type'**] . **'">'**;  
 **echo** *htmlspecialchars*($msg[**'text'**]);  
 **echo '</li>'**;  
 }  
 **echo '</ul>'**;  
 **unset**($\_SESSION[**'messages'**]);  
}  
**?>**

### Create a new file: show-validation-errors.php

<**script**>  
 **<?php  
 foreach** (*array\_reverse*($this->**validationErrors**) **as** $fieldName => $errorMsg) {  
 $fieldJson = *json\_encode*($fieldName);  
 $errorMsgJson = *json\_encode*($errorMsg);  
 **echo "showValidationError(**$fieldJson**,** $errorMsgJson**);\n"**;  
 }  
 **?>**</**script**>

# Create a new directory inside views named home

## Create a new file – **index.php**

**<?php** $this->**title** = **'Welcome to My Blog'**; **?>**<**h1**>**<?=***htmlspecialchars*($this->**title**)**?>**</**h1**>  
  
*<!--* ***TODO: display the posts here*** *-->*

8.2. Create a new file – **view.php**

**<?php** $this->**title** = **"Post Title" ?>**<**h1**>**<?=***htmlspecialchars*($this->**title**)**?>**</**h1**>  
  
*<!--* ***TODO: display a single post here*** *-->*

# Create a new directory inside views named users

## Create a new file – **login.php**

**<?php** $this->**title** = **'Login'**; **?>**<**h1**>**<?=** *htmlspecialchars*($this->**title**) **?>**</**h1**>  
  
*<!--* ***TODO: login form will come here ...*** *-->*

## Create a new file – **register.php**

**<?php** $this->**title** = **'Register New User'**; **?>**<**h1**>**<?=** *htmlspecialchars*($this->**title**) **?>**</**h1**>  
  
*<!--* ***TODO: register form will come here ...*** *-->*

# Output

## Index.php



## Login



## Register

