DMIT2503 – ADVANCED WEB CONCEPTS

CODEIGNITER MVC PHP FRAMEWORK – DAY 1

- Using the basic build provided, upload the whole folder *ci-2020* to your webserver.
- Go to that URL, and you should see the standard CodeIgniter (CI) welcome message page.
    - Let's use this URL as our example as we will be working with URL segments quite a bit: http://username.dmitstudent.ca/dmit2503/ci-2020

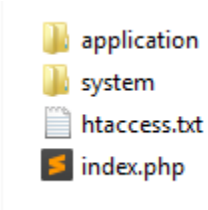
HTACCESS MOD REWRITE

Let's try seeing how CI deals with it's segment-based navigation: Now let's try manually navigating to the ***default controller*** which is called Welcome. Type in ***/welcome*** after your URL:
http://username.dmitstudent.ca/dmit2503/ci-2020/welcome

- We may see a **Not Found** error.
- Now try appending both */index.php/* and */welcome/* to the URL:
http://username.dmitstudent.ca/dmit2503/ci-2020/index.php/welcome
- Note that WITH the index.php in the URL we can see the main welcome page. However, we would like to get rid of the index.php in the URL, so let's look at the provided htaccess.
- In this build, we have included the necessary htaccess file. Please change ***htaccess.txt*** to *.htaccess* (note the dot in front) on your web server.
- Now do the same test: type in the URL with ***/welcome*** and no ***index.php***. It should now resolve to the welcome page ☺
- *Note*: This is not the standard htaccess file that comes with CI. I found this in a quick Google search and it has tested well on dmitstudent. You may have to do some more work depending on your hosting environment. If your server doesn't do the mod rewrite, then live with the index.php before all segments.

FILE STRUCTURE

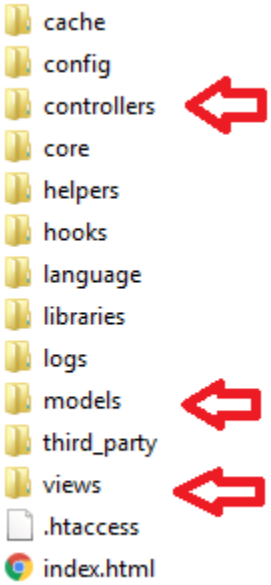
Open up the local folder and let's see what we're working with:



Note the 2 folders:

- **Application:** This is where we build our app.
- **System:** This is the engine. Don't even think of touching this.

Open the **Application** folder:



Note the many folders. Also note what is being pointed out:

- **Controllers:** The logic of your app. Also, in CI this is the segment based URL's.
- **Models:** The database interaction of your app. DB queries go here.
- **Views:** What the user sees. This will the client-side (HTML, etc.) plus whatever is created by the controllers and models.

Generally, we will only be working in these three folders. We will have some config files to edit, and may delve into some libraries, but pretty much all of our work is done in these three folders.

CONFIGURATION

Although the base functionality of CI will work out of the box, we will need to change some configuration settings to enable much of the framework. Please save and upload these files after you edit them.

Open the **application/config/config.php** file and edit these settings:

- $config['base_url'] = '';
  - Copy/paste the root URL of your app from the browser to this.
  - **$config['base_url'] = 'http://username.dmitstudent.ca/dmit2503/ci-2020';**
  - This is mission critical. We will be using this to resolve all links, so get it right please.

- $config['index_page'] = 'index.php';
    - Make this empty. Since the htaccess mod rewrite is working, we don't want index.php appended.
    - **$config['index_page'] =’’;**

- $config['encryption_key'] = '';
    - Put anything you like in here. It's your encryption key so it can be anything. CI will encrypt sessions, etc. so this helps with security.
    - **$config['encryption_key'] = 'phil-is-a-groovy-cat';**

DATABASE

First, please go into your Virtuemin account (https://studentweb2.sicet.ca:10000/) and create a new database. Call it “ci_2020” or anything you like.

Now, open the **application/config/database.php** file and edit these settings. If you get an error, then restore these to the defaults and leave them for now.

- 'hostname' => 'localhost',
- 'username' => '',
- 'password' => '',
- 'database' => '',

OTHER CONFIG FILES

Keep these 2 files in mind as we will come back to them:

- **Autoload:** We need to load certain libraries and helpers as we do various things. If we want them loaded as we start the app (as opposed to only being loaded for parts of the app), we can do that here.
- **Routes :** Routes here is not a needy as Laravel, Angular, or other frameworks. We will change the default controller at some point however.

DMIT2503 – ADVANCED WEB CONCEPTS

CONTROLLERS & VIEWS

Ok, time to start creating an app.

- In an editor of your choice, open the following file: **application/controllers/Welcome.php.**
- Save this as a new file. Same folder, but call it **Home.php.** Note that the file MUST be mixed-case (uppercase first character).
- Change the code to the following:

Note: Pasting code into and out of MSWord or PDF docs may change certain characters (especially double quotes, etc.), so please the time to re-type things if they don't work or don't look correct.

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');

class Home extends CI_Controller {


	public function index()
	{
		echo "This is home";

	}
}
```

Please note the following rules:

1. Even though we are working with PHP and you see an opening PHP block (<?php), there is NO closing PHP block (?>). This is because the user will never go directly to the file; the file is parsed by the CI system.
2. The class name has changed from Welcome to Home.
3. Always best to copy/paste controllers and models (Save As >> New filename) as they follow a set syntax.
4. The echo is just to test. Never best practice to write something from the controller, but it can be done.

Save and upload your file.

Now manually browse to your URL, but add the segment called /home/ to it:
http://username.dmitstudent.ca/dmit2503/ci-2020/home.

Viola. You should see a simple echo of "This is home". You have created a new controller and learned about CI's segment based navigation.

In CodeIgniter, the controller name becomes the first segment of the URL. Links can be made to navigate to these.

DMIT2503 – ADVANCED WEB CONCEPTS

VIEWS

Ok, time to create our own view file.

- Create a new file from scratch, name it ***home_view.php,*** and save it in the *application/views* folder.
- Create an H1 heading (nothing else, no template HTML) with <h1>Home, Sweet Home!</h1>
- In your **home controller**, remove the echo and add a line to load this view.

```
// Home.php controller
public function index()
{
	//echo "This is home";

	$this->load->view('home_view');
}
```

```
// home_view.php in Views folder

<h1>Home, Sweet Home!</h1>
```

- Save, upload, and go to the home segment: http://username.dmitstudent.ca/dmit2503/ci-2020/home.
- You should now see a controller loading a view.

Note: While controllers and models must follow the set syntax, views don't. They can be lowercase, but must retain the ".php" extension assuming we want to use PHP in them (we do). However, you will open and close all PHP blocks in views. Generally, some devs (me) will call them *something_view.php* so that when we have multiple files open in an editor, we know what this one is.

DEFAULT ROUTE

Since we are now creating our own app, we can do away with the Welcome controller and views.

- Open the **application/config/routes** file and change the default controller to home.
  - $route['default_controller'] = 'home';
- Now, go to the root of your app: http://username.dmitstudent.ca/dmit2503/ci-2020/
- Voila. The Home controller and view are the home page.

DMIT2503 – ADVANCED WEB CONCEPTS

OUR APP TEMPLATE

While many MVC frameworks use a templating engine, and there are some 3rd party ones for CodeIgniter, we will use a simple header/footer files for our app template.

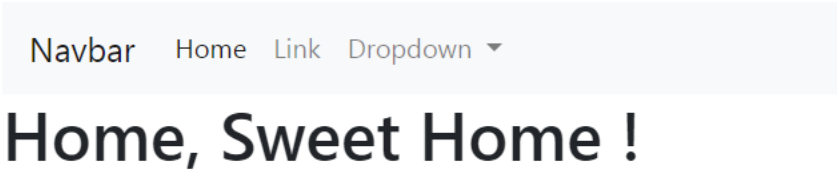
- In your application/views folder, create a subfolder called */includes/*.
- Now, create 2 blank files in that folder:
  - header.php
  - footer.php
- Put anything you like in the 2 files, then load them on your controller before and after your home view, save, upload, and have a look.

```
// Home.php controller
public function index()
{
        $this->load->view('includes/header');
        $this->load->view('home_view');
        $this->load->view('includes/footer');
}
```

Ok. Now let's prettify this a bit. We'll use Bootstrap 4 as a quick and easy UI to skin our app.

- On your own, go to https://getbootstrap.com/docs/4.4/getting-started/introduction/ and create a simple template of front end code that you can split between the two files.
  - Everything from the doctype until the <body> goes in header.
  - Everything from the optional JS to the end of the html goes in footer.
- Within both your includes, create a in header (the goes in footer).
- Now, let's try putting a Bootstrap Navbar in the header. This should be just within the which is itself just within the <body>.
  - https://getbootstrap.com/docs/4.0/components/navbar/
  - In the navbar template code, feel free to play around and remove some links, search etc.
  - Let's leave in a link or two and the dropdown.

DMIT2503 – ADVANCED WEB CONCEPTS



Ok. That is a great start to our CodeIgniter app. Let's review a few basic concepts:

- All work is done in the application folder.
- Controllers must be mixed case (uppercase first character) and the file name and class name are the same. The closing PHP tag is never present.
- This controller/class name also becomes the way we navigate through the app as it becomes the first segment.
- Views can be mostly HTML, but with PHP mix-ins, so they should have the ".php" extension.
- Views are loaded from the controller. We refer to them with their filename without their extensions.

ONE LAST LITTLE TEST

Add a new function to your Home controller like this.

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class Home extends CI_Controller {
	public function index()
	{
		$this->load->view('includes/header');
		$this->load->view('home_view');
		$this->load->view('includes/footer');
	}
	public function test()
	{
		$this->load->view('includes/header');
		$this->load->view('test_view');
		$this->load->view('includes/footer');
		//echo "<h1>This is TEST function in Home controller</h1>";
	}
}
```

- Create a new view with an *<h1>This is TEST function in Home controller</h1>.*

Now, how can we navigate to see this function within a class?
https://codeigniter.com/user_guide/general/urls.html?highlight=segment

See you next time folks ☺