Page Models

In prior iterations of the framework you would define your page model in the code behind page and instantiate it within the page model file.  
  
page.code.php

Class page extends BasePage

{

Public function Init()

{

$this->MasterPage = MASTERPAGE\_DEFAULT;

}

}

$PageModel = new page($Request, $Session, $Cookie, $CurrentUser);

Then in page.php you would put

<?php /\* @var $PageModel page \*/ ?>

At the top of your file and PHPStorm would know what page model you were using. However, that recently changed and PHPStorm would no longer know what class $PageModel was outside of the first code black. And with large projects with hundreds of pages, now you have the global $PageModel and PHPStorm doesn’t know which of the 100 is being used on page.php. While that may be fixed in a later version of PHPStorm (it was not an issue before a certain version), it prompted a change to how WebView works.

Rather than instantiating the class in the code behind, you use a define to tell the framework which class you are using and how to use it. It now supports static and instance based page models.

For an instance based page model (using $this in the various methods) you replace

$PageModel = new page($Request, $Session, $Cookie, $CurrentUser);

With define(‘PAGE\_MODEL’,’page’);

The original method is still supported but not recommended.

You can also define you page model as a static class

Class page extends BasePage

{

Public static function DoInit()

{

Self::$MasterPage = MASTERPAGE\_DEFAULT;

}

}

Note that “Do” is now in front of Init. The same is for Get and Post. When using an instance of the page model, use Get and Post. When using the static option use DoGet and DoPost

For an static based page model (using self:: in the various methods) you replace

$PageModel = new page($Request, $Session, $Cookie, $CurrentUser);

With define(‘PAGE\_MODEL\_STATIC’,’page’);

QuickDRY’s WebView file will figure out how to load the page model and configure it. $Request, $Session, $Cookie and $Server are no longer globals. You access them in the global scope through $Web->Request, etc.

Which is not recommended. Ideally, the only global in your project should be $Web

This largely takes care of the issue of PHPStorm not knowing what page model you’re referring to in your view. Of course, if you have multiple page models with the same name, there’s not much you can do about it. Try to avoid it.

At this time, QuickDRY doesn’t support using namespaces for Page Models.