Changing the Yii directory layout

By default, we have the Basic and Advanced Yii2 application skeletons with different directory structures.  
But these structures are not dogmatic, and we can customize them if required.

For example, we can move the runtime directory out of the project.

Getting ready

Create a new yii2 -app-basic application by using the Composer the package manager, as described in  
the official guide at <http://www.yiiframework.com/doc-2.0/guidestart-installation.html>.

How to do it...

Changing the location of the runtime directory

Open config/web.php and config/console.php and define the runtimePath parameter:

$config = [

'id' => ' basic',

'basePath' => dirname( DIR ),

'bootstrap' => ['log'],

'runtimePath' => '/tmp/runtime',

'components' => [

// ...

],

]

Move the runtime directory to the new location.

Changing the location of the vendor directory

1. Open config/web. php and config/console. php and define the vendorPath parameter:

$config = [

'id' => 'basic',

'basePath' => dirname( DIR ),

'bootstrap' => ['log'],

'vendorPath' => dirname( DIR ), '/../vendor,

'components' => [

// ...

],

]

1. Move the vendor directory with the composer.json and composer.lock files to the new location.
2. Open the web/index. php and yii files and find these rows:

require( DIR . '/../vendor/autoload.php');

require( DIR . '/../vendor/yiisoft/yii2/Yii.php');

1. Change the including paths.

Changing the location of the controllers

1. Rename the commands directory to console.
2. Change the namespace of app\commands\HelloController to app\console\HelloController.
3. Open config/console.php and redefine the controllerNamespace parameter:

$config = [

'id' => 'basic-console',

'basePath' => dirname( DIR ),

'bootstrap' => ['log'],

'controllerNamespace' => 'app\console,

'components' => [

// ...

],

]

Changing the locations of the views directory

1. Open config/web. php and define viewPath parameter:

$config = [

'id' => 'basic',

'basePath' => dirname( DIR ),

'bootstrap' => ['log'],

'viewPath' => '@app/myviews',

'components' => [

// ...

],

]

1. Rename your views directory.

How it works...

In the yii\base\Application : :preInit method our application defines basePath, runtimePath, and  
vendorPath parameters.

By default, these values lead to the root application directory, runtime and vendor path in the root  
respectively.

For example, you can redefine vendorPath if you want to share the vendor directory with some instances  
of the same projects. But take care of the package’s versions compatibility.

The yii\base\Application class extends yii\base\Module, which contains the controllerNamespace  
and viewPath parameters. The first one allows you to change the base namespace of the application and  
modules. It is helpful if you want to provide frontend and backend controllers in the same module  
directory. Just change the controllers directory to frontend and backend or create subdirectories and  
configure your frontend and backend applications:

return [

'id' => 'app-frontend',

'basePath' => dirname( DIR ),

'controllerNamespace' => frontend\controllers',

'bootstrap' => ['log'],

'modules' => [

'user' => [

'my\user\Module',

'controllerNamespace' => 'my\user\controllers\frontend',

]

],

// ...

]

return [

'id' => 'app-backend',

'basePath' => dirname( DIR ),

'controllerNamespace' => 'backend\controllers',

'bootstrap' => ['log'],

'modules' => [

'user' => [

'my\user\Module',

'controllerNamespace' => 'my\user\controllers\backend',

]

],

// ...

]

See also

[In order to learn more about application structures, refer to http://www.viiframework.com/doc-2.0/guide-](http://www.yiiframework.com/doc-2.0/guide-structure-applications.html)  
structure-applications.html.