Using external code

Package repositories, PSR standards, and social coding provide us with lots of high-quality reusable libraries and other components with free licenses. We can just install any external component in project instead of reengineering them from scratch. It improves development performance and makes for higher- quality code.

Getting ready

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

How to do it...

In this recipe we will try to attach some libraries manually and via Composer.

Installing a library via Composer

When you use NoSQL or other databases without autoincrement primary keys, you must generate unique identifiers manually. For example, you can use Universally Unique Identifier (UUID) instead of a numerical one. Let’s do it:

1. Install <https://github.com/ramsey/uuid>component via Composer:

composer require ramsey/uuid

1. Create a demonstration console controller:

<?php

namespace app\commands;

use Ramsey\Uuid\Uuid; use yii\console\Controller;

class UuidController extends Controller {

public function actionGenerate()

{

PHP\_EOL);

php\_eol); php\_eol); php\_eol); php\_eol);

$this->stdout(Uuid::uuid4()->toString()

$this->stdout(uuid::uuid4()->toString()

$this->stdout(uuid::uuid4()->toString()

$this->stdout(uuid::uuid4()->toString()

$this->stdout(uuid::uuid4()->toString()

}

}

1. And just run it:

./yii uuid/generate

1. If successful, you’ll see the following output:

25841e6c-6060-4a81-8368-4d99aa3617dd

fcac910a-a9dc-4760-8528-491c17591a26