Redis database driver

This extension allows you to use Redis key-value storage in any project on the Yii2 framework. It contains the Cache and Session storage handlers, as well as the extension, which implements the ActiveRecord pattern for access to the Redis database records.

Getting ready

1. Create a new application by using composer, as described in the official guide at <http://www.yiiframework.com/doc-2.0/guide-start-installation.html>.
2. Install the storage: <http://redis.io>.
3. Install all migrations with the following command:

composer require yiisoft/yii2-redis

How to do it...

First of all, configure the Connection class in your configuration file:

return [

//....

'components' => [

'redis' => [

'class' => 'yii\redis\Connection',

'hostname' => 'localhost',

'port' => 6379,

'database' => 0,

],

]

];

Direct usage

For low-level working with Redis commands, you can use the executeCommand method of the connection component:

Yii::$app->redis->executeCommand('hmset', ['test\_collection', 'keyl', 'vall', 'key2',

'val2']);

You can also use simplified shortcuts instead of executeCommand calls: Yii::$app->redi

s->hmset('test\_collection', 'key1', 'val1', 'key2', 'val2')

Using ActiveRecord

For access to Redis records via the ActiveRecord pattern, your record class needs to extend from the yii\redis\ActiveRecord base class and implement the attributes() method:

class Customer extends \yii\redis\ActiveRecord {

public function attributes()

{