defined('YII\_ENV') or define('YII\_ENV', 'prod');

$dir = dirname($\_SERVER['SCRIPT\_FILENAME']);

require($dir . '/../vendor/autoload.php'); require($dir . '/../vendor/yiisoft/yii2/Yii.php');

$config = require($dir . '/../config/web.php');

(new yii\web\Application($config))->run();

Also, set the content of the share/yii file:

#!/usr/bin/env php <?php

defined('YII\_DEBUG') or define('YII\_DEBUG', false); defined('YII\_ENV') or define('YII\_ENV', 'prod');

$dir = dirname($\_SERVER['SCRIPT\_FILENAME']);

require($dir . '/vendor/autoload.php'); require($dir . '/vendor/yiisoft/yii2/Yii.php');

$config = require($dir. '/config/console.php');

$application = new yii\console\Application($config);

$exitCode = $application->run(); exit($exitCode);

Note

Note: We deliberately use the dirname($\_server [' script\_filename ' ]) code instead of the original dir constant because dir will return incorrect value when the file is included via symbolic link.

Note: If you use the yii2-app-advanced template you can redeclare only the config/main-local.php and config/params-local.php files of every (backend, frontend, console, and common) because web/index. php and yii files will be created automatically by the init command.

Step 4 - Trying to deploy

1. Come back to the localhost with the deploy .php file and run the deploy command: dep deploy prod
2. If successful, you will see the deployment report: