

« Day 6 »

La semaine dernière, nous avons utilisé github et corrigé notre code avec du checkstyle. Aujourd'hui c'est le dernier jour de ce module, vous devez me rendre votre rendu final ! Rappel du trello <https://trello.com/b/VGsD5zc1/sf-2018>

Un peu plus de détails sur l'aspect couverture des tests avec Codeception. Le reste du temps de la journée étant alloué à finir votre projet pour le rendre à **17h45 dernier délai**.

Pour me communiquer vos projets : envoyer un mail à florent.coquil@gmail.com avec en titre :

« MMI SYMFONY 2018 – <NOM> <PRENOM> »

Dans le mail, simplement un lien vers votre projet github, pensez à soigner la mise en forme avec un beau README.md sur github...

Les tests, commençons par un test unitaire...

Partons sur un framework comme <https://codeception.com/>

Suivez la procédure pour installer Codeception.

<https://codeception.com/for/symfony>

Installation avec composer :

```
composer require "codeception/codeception" --dev
```

Pour initialiser les tests

```
Php ./vendor/bin/codecept bootstrap
```

Pour créer un premier test

```
Php ./vendor/bin/codecept generate:test unit Example
```

Pour lancer vos tests

```
Php ./vendor/bin/codecept run unit
```

Les tests unitaires servent à tester votre modèle de données.

Réaliser des tests unitaires pour votre modèle qui va tester vos getters et setters.

Voici un exemple qui n'a rien à voir avec votre projet :

```

/**
 * Test for AdminStat Document.
 */
public function testModelAdminStat()
{
    $entity = new AdminStat();
    $now = new \DateTime();

    $this->assertEmpty($entity->getId());

    $entity->setCurrentTime($now);
    $entity->setTotalDbSize(11);
    $entity->setTotalJob(12);
    $this->assertEquals($now, $entity->getCurrentTime());
    $this->assertEquals(11, $entity->getTotalDbSize());
    $this->assertEquals(12, $entity->getTotalJob());

    $entity->setTotalDbSize(10000000000000000);
    $entity->setTotalJob(20000000000000000);
    $this->assertEquals(10000000000000000, $entity->getTotalDbSize());
    $this->assertEquals(20000000000000000, $entity->getTotalJob());

    $entity->setTotalDbSize(-1);
    $entity->setTotalJob(-2);
    $this->assertEquals(-1, $entity->getTotalDbSize());
    $this->assertEquals(-2, $entity->getTotalJob());
}

```

Maintenant on va tenter un test fonctionnel. Pour cela, comme indiqué <https://codeception.com/quickstart>

Faisons une suite dédiée :

```
Php ./vendor/bin/codecept generate:cest acceptance First
```

Comme indiqué dans le répertoire test, on crée un fichier **acceptance.suite.yml**

Avec

```

actor: AcceptanceTester

modules:
    enabled:
        - PhpBrowser:
            url: {YOUR APP'S URL}
        - \Helper\Acceptance

```

Maintenant on édite le test :

acceptance/FirstCest.php

```
<?php
class FirstCest
{
    public function frontpageWorks(AcceptanceTester $I)
    {
        $I->amOnPage('/');
        $I->see('Home');
    }
}
```

Nous avons plus qu'à le lancer :

```
Php ./vendor/bin/codecept run acceptance
```

Terminé !

Fin de ce module.