Rapport de couverture de code

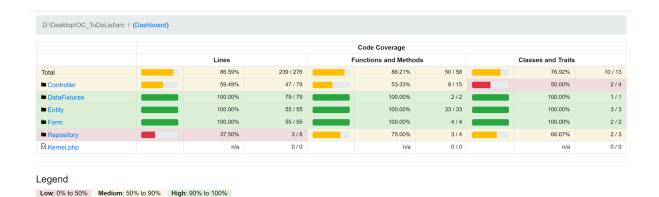
Introduction

Ce rapport présente une analyse détaillée de la couverture de code du projet OC ToDoList.

La couverture de code est une métrique essentielle qui mesure le pourcentage de code source exécuté lors des tests automatisés.

Un taux de couverture minimum et supérieur à 70% est généralement considéré comme un indicateur de qualité, garantissant que les fonctionnalités critiques du code sont bien testées.

Ce rapport met en évidence la couverture actuelle, identifie les zones nécessitant des améliorations, et fournit des recommandations pour optimiser la qualité du code et assurer une maintenance efficace.



Résumé général

Couverture totale: 86,59 %

Lignes : 239 / 276

Fonctions & Méthodes: 86,21 % (50 /

Generated by php-code-coverage 9.2.32 using PHP 8.3.11 and PHPUnit 9.6.21 at Wed Oct 30 16:48:08 UTC 2024

58)

Classes & Traits: 76,92 % (10 / 13)

Partie	Couverture (%)	Lignes
Controller	59,49%	47 / 79
DataFixtures	100%	79 / 79
Entity	100%	55 / 55
Form	100%	55 / 55
Repository	37,50%	3/8

Analyse

Couverture globale:

La couverture globale de 86.59% est bien au-dessus de l'objectif de 70%. Cela indique que la majorité du code est testé.

Secteurs d'amélioration :

Les contrôleurs sont en dessous de la moyenne avec 59,49 %, il serait bénéfique d'écrire des tests supplémentaires pour les contrôleurs afin d'améliorer cette couverture.

Les repository ont un faible taux de couverture (37,50 %) et devraient être priorisées pour des tests supplémentaires.

Secteurs à maintenir :

Les DataFixtures, Entity, et Form affichent une couverture de 100%, ce qui est excellent.

Ces parties sont bien testées et doivent être maintenues ainsi.

```
Windows PowerShell
\label{psi} PS \ D:\Desktop\C_ToDoList> php \ bin/phpunit --coverage-html coverage/ --testdox \ PHPUnit 9.6.21 by Sebastian Bergmann and contributors.
Testing
Default Controller (App\Tests\Controller\DefaultController)
Security Controller (App\Tests\Controller\SecurityController)

√Login page
Task Controller (App\Tests\Controller\TaskController)

\( \sqrt{reate} \) task as authenticated user

\( \sqrt{task} \) as authenticated user

\( \sqrt{reate} \) task as authenticated user

\( \sqrt{reate} \) task as authenticated user

\( \sqrt{gogle} \) task as authenticated user

\( \sqrt{pelete} \) task as authenticated user

\( \sqrt{pelete} \) task as unauthorized user

\( \sqrt{get} \) anonymous user creates if not exists
App Fixtures (App\Tests\DataFixtures\AppFixtures)

VLoad fixtures
Task (App\Tests\Entity\Task)

Set author and get author

Set author as anonymous

Toggle is done
User (App\Tests\Entity\User)

Get id

Set and get email

Set and get username

Set and get password

Set and get roles

Get roles default

Set roles enforce user role

Set roles only admin

Get user identifier

Erase credentials
User Type (App\Tests\Form\UserType)

*Build form with creation option

*Invalid form with mismatched passwords

*Configure options
 Tag_Repository_(App\Tests\Repository\TagRepository)

✓Add tag
  ✓Add tag
✓Find tag by id
Time: 00:33.007, Memory: 80.00 MB
Generating code coverage report in HTML format ... done [00:00.357] PS D:\Desktop\OC_ToDoList> \mid
```

Conclusion

Nous avons atteint et dépassé l'objectif de 70% de couverture de code avec une couverture globale de 86.59%.

Cependant, un effort supplémentaire est nécessaire pour augmenter la couverture des sections Controller et Repository.