

Table des matières

[I-       Suivi de qualité. 2](#_Toc87264757)

[1-      Outils mis en place. 2](#_Toc87264758)

[II-     Etude de la dette technique. 3](#_Toc87264759)

[III-         Suivi des performances. 4](#_Toc87264760)

# I-                  Suivi de qualité

 Deux outils ont été mis en place afin d’assurer le suivi de la qualité du code

Codacy

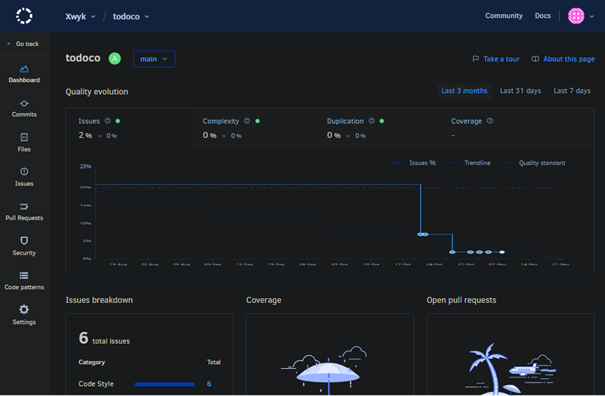
                Analyse :

-          Les vulnérabilités

-          Les mauvaises pratiques

-          Le code inutilisé

-          La duplication de code



SonarQube (<https://www.sonarqube.org/>)

Analyse :

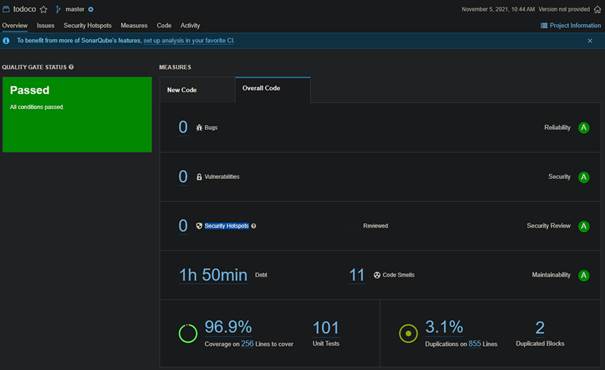
-          La fiabilité du code (recherche de bugs)

-          Les vulnérabilités

-          Les « Security Hotspots », morceaux de code potentiellement risqués point de vue sécurité, nécessitant une vérification de la part du développeur

-          Les mauvaises pratiques présentes dans le code

L’outil permet également l’interprétation des résultats PHPUnit, ainsi que la vérification de duplication de code afin de centraliser les informations sur un unique dashboard.



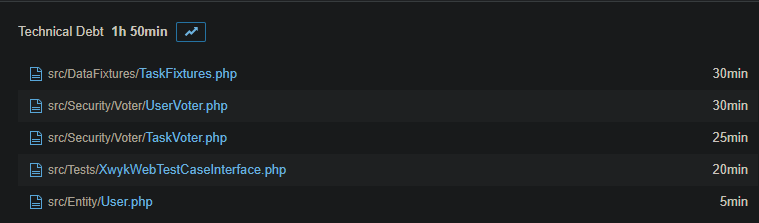
Pour ces deux outils, les standards stipulés dans le CONTRIBUTING.md sont analysés (PSR1-2-4)

En local sur le projet, installés via Composer, PHPStan et PHP-cs-fixer ont également permis de suivre la qualité du code, et des mauvaises pratiques de syntaxe lors du développement. PHP-cs-fixer est inclus dans la CI github, et termine l’action si la sortie est en erreur. Il est donc important de le lancer avant de pusher les modifications sur le repo.

# II-                Etude de la dette technique

La dette technique représente les bugs présents dans l’application, liés ou non au non-respect des règles établies.

Ici, la dette technique est estimée par l’outil de suivi de qualité SonarQube à 1h50.



Sont donc répertoriées toutes les modifications à effectuer afin de réduire la dette. Par exemple :



On peut ici voir qu’une des modifications recommandée par l’outil est de rassembler les deux vérifications en une seule.

La réduction de la dette technique de l’application d’une manière générale passe par :

* L’établissement de règles de codage (ici, respect des PSR1-2-4 et best practices pour Symfony 5.3)
* Le suivi de la qualité du code au fur et à mesure du projet
* Le développement de tests unitaires et fonctionnels afin de valider chaque modification de code

# III-             Suivi des performances