

Audit

Samir Mehal
ToDo & Co
Openclassrooms - Projet 8
France

12 février 2024

Table des matières

1 - Résumé du projet	3
2 - Audit	3
2 -.1 Codacy	3
2 -.1.1 Projet avec Symfony 3	3
2 -.1.2 Projet avec Symfony 7 avant modification	5
2 -.1.3 Projet avec Symfony 7 après modification	6
2 -.1.4 Comparaison	7
2 -.2 Profiler de Symfony	7
2 -.2.1 Projet avec Symfony 7 avant modification	7
2 -.2.2 Projet avec Symfony 7 après modification	10
2 -.2.3 Comparaison	13
Conclusion	14

1 - Résumé du projet

Le projet ToDo & Co est un site web qui permet de créer et de modifier des utilisateurs ainsi que de créer et de modifier des tâches. Le but de ce document est de réaliser un audit du site web tel qu'il est ainsi qu'un audit du site web après ajout des modifications demandées ainsi que des tests. Le site web a été refait from scratch car les versions de PHP et de Symfony utilisées étaient trop anciennes. Nous sommes passés de la version 5.5.9 de PHP à la version 8.2, et de la version 3.1 de Symfony à la version 7.0.

2 - Audit

Un audit d'un site web est une analyse de la qualité de son code et de ses performances. Pour auditer la qualité du code, nous utiliserons Codacy de Symfony et pour auditer la performance nous utiliserons le profiler de Symfony.

2 - .1 Codacy

Codacy automatise les révisions de code et surveille la qualité du code à chaque demande de validation et d'extraction. Il rapporte l'impact de chaque demande de validation ou d'extraction sur les nouveaux problèmes concernant le style de code, les meilleures pratiques, la sécurité et bien d'autres.

Codacy s'utilise sur le repository de Github, ici sur le projet ToDo & Co.

2 - .1.1 Projet avec Symfony 3

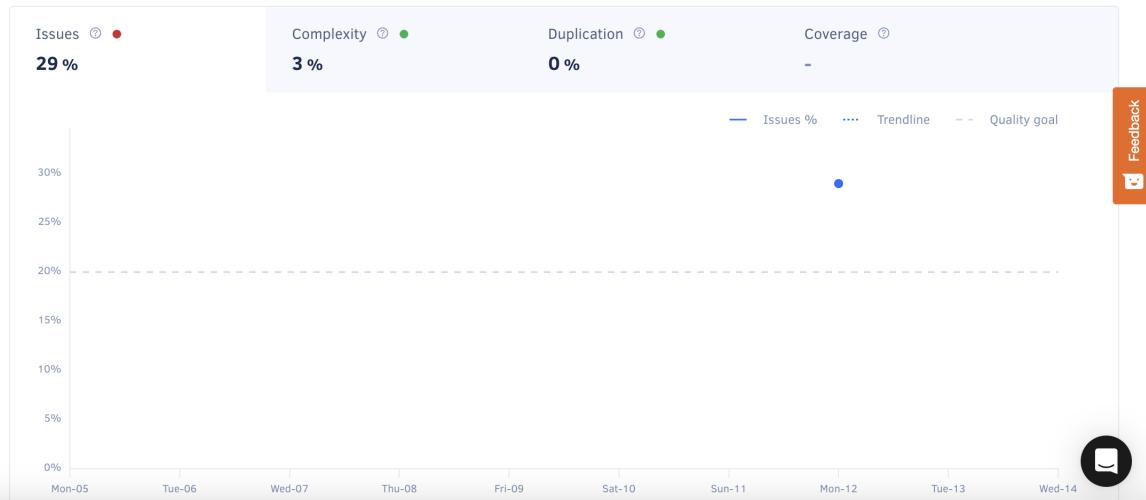
En utilisant Codacy sur le repository du projet avec Symfony 3, nous obtenons le résultat suivant :

Quality evolution

Last 3 months

Last 31 days

Last 7 days



Feedback

Issues breakdown

286 total issues

Category	Total
Code Style	253
Security	30
Unused Code	3
Best Practice	0
Comprehensibility	0
Error Prone	0
Performance	0
Compatibility	0
Code Complexity	0
Documentation	0

[See all issues](#)

Coverage



Set up coverage

See [how to generate coverage reports](#) and start [sending coverage results](#) to Codacy.

Open pull requests



All clear!

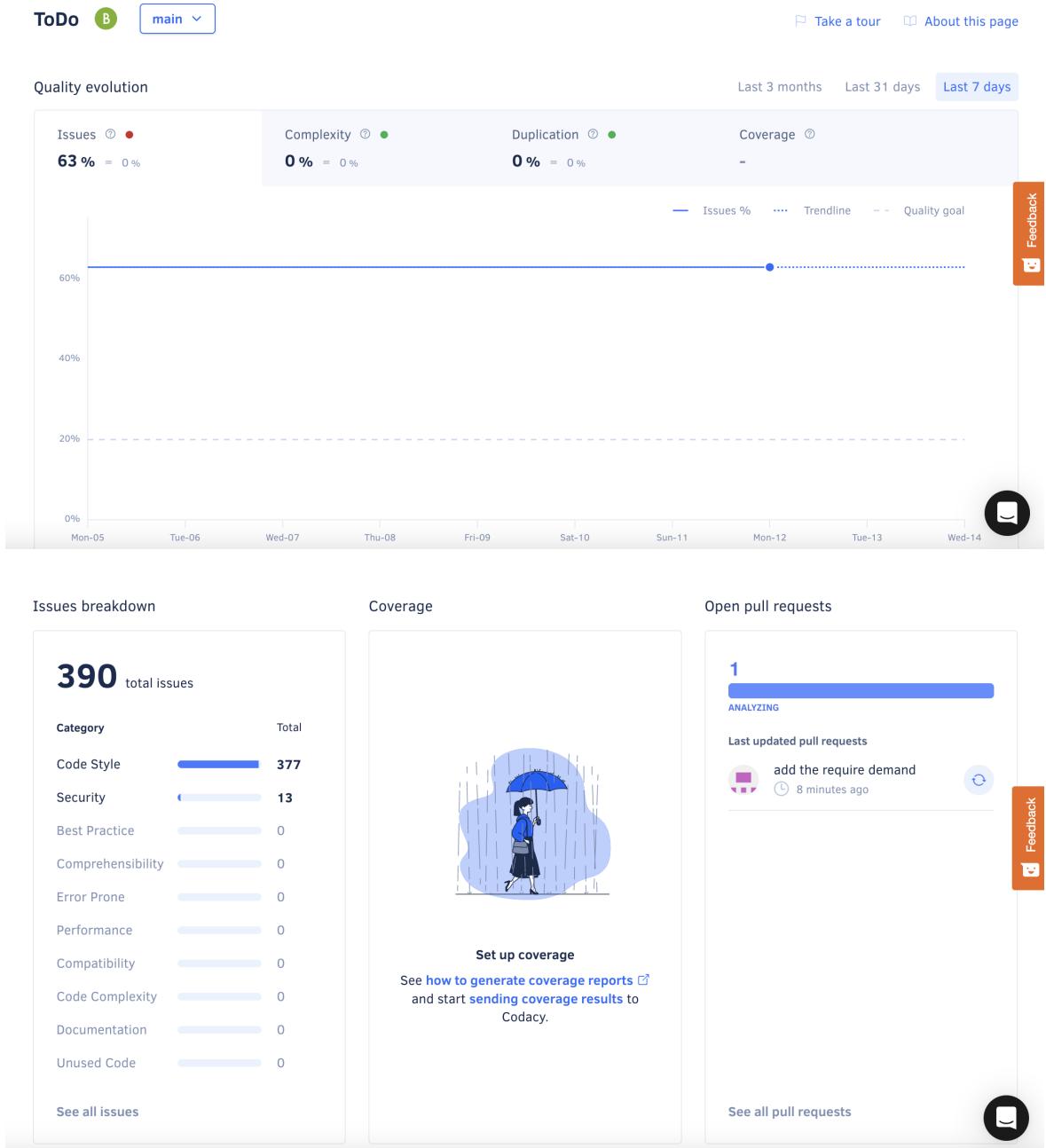
When pull requests are open, you'll be able to see here an overview of the status.

Feedback



2 .1.2 Projet avec Symfony 7 avant modification

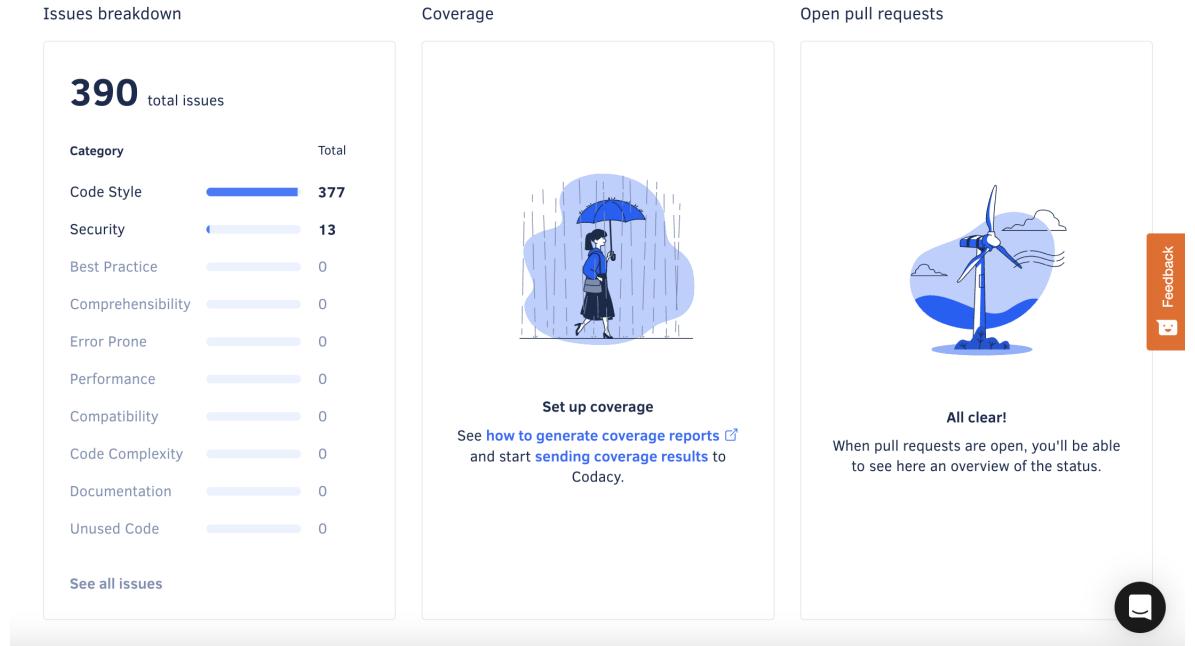
En utilisant Codacy sur le repository du projet avec Symfony 7 avant d'ajouter les modifications, nous obtenons le résultat suivant :



2 .1.3 Projet avec Symfony 7 après modification

En utilisant Codacy sur le repository du projet avec Symfony 7 après avoir ajouter les modifications et écrit les tests unitaires et fonctionnels, nous obtenons le résultat suivant :





2 -.1.4 Comparaison

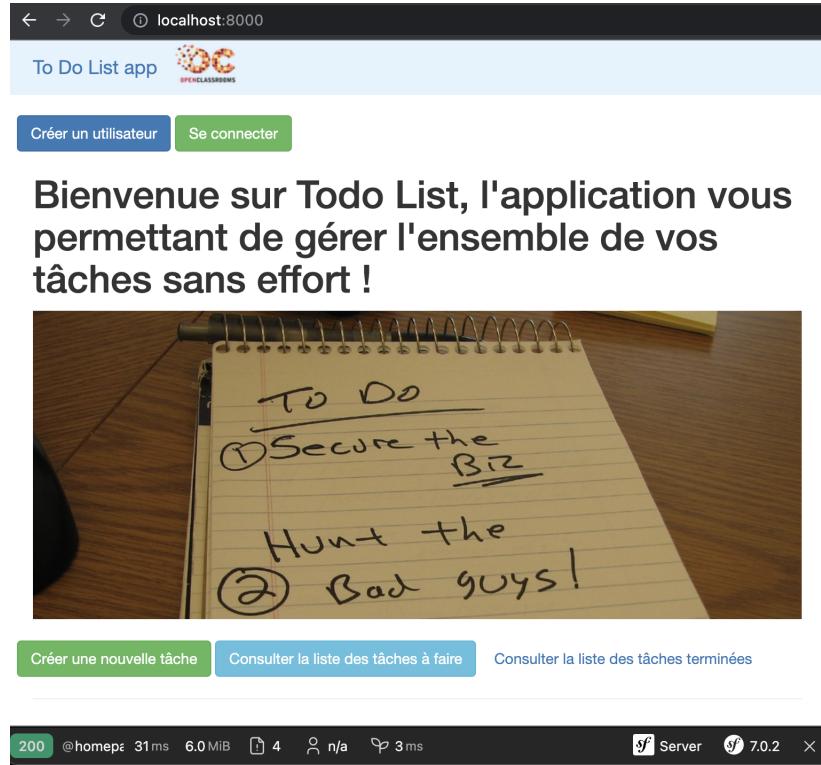
Plus le projet est récent et plus il possède de modifications, ce qui explique que le pourcentage d'issues augmente. Le nombre d'issue augmente aussi mais dans le projet avec Symfony 7 et après modification, des fichiers correspondant aux fichiers internent à Symfony ont été ignoré pour améliorer l'analyse de Codacy et donc la note globale. Le projet reste globalement le même, seul deux trois ajouts ont été réalisés (ajout d'un auteur aux tâches, seul ceux qui ont créé une tâche peuvent la supprimer, seul les administrateur peuvent accéder aux pages utilisateurs et choix du rôle lors de la création et de la modification d'un utilisateur).

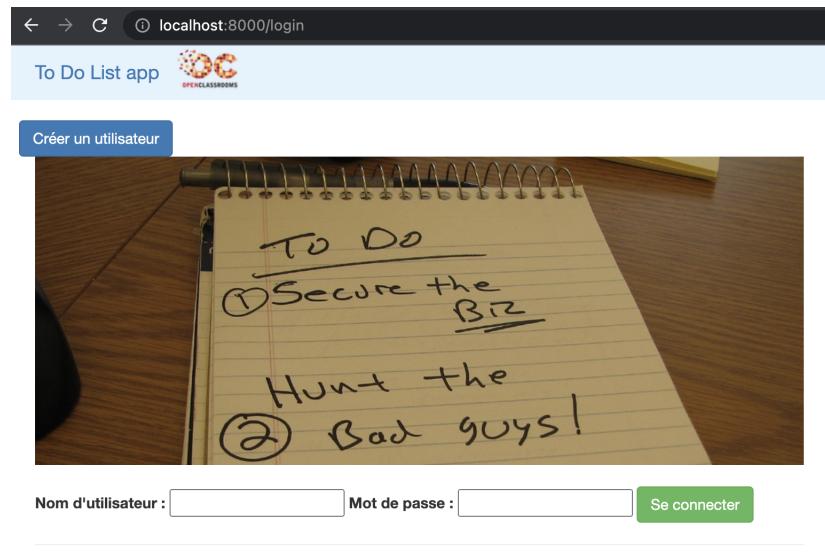
2 -.2 Profiler de Symfony

Le Web Profiler est une application Symfony livrée avec le framework Standard Edition et disponible quand nous travaillons en environnement de développement. Il permet de collecter de nombreuses informations sur notre application durant son exécution comme le temps de rendu de la page, la consommation de mémoire, le nombre d'appels au cache, etc.

2 -.2.1 Projet avec Symfony 7 avant modification

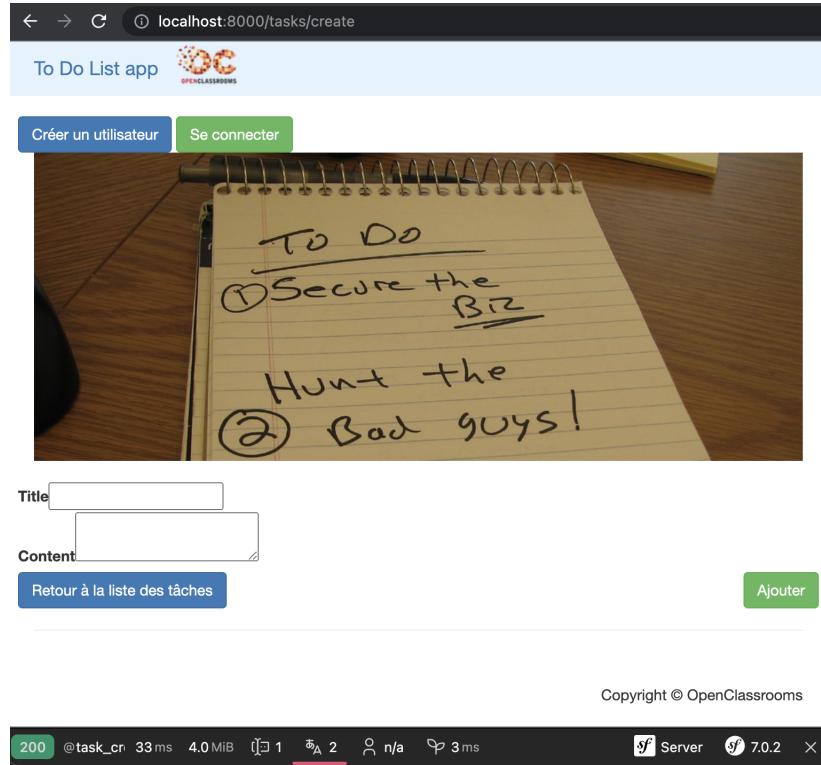
En utilisant le Profiler de Symfony sur le repository du projet avec Symfony 7 avant d'ajouter les modifications, nous obtenons le résultat suivant :





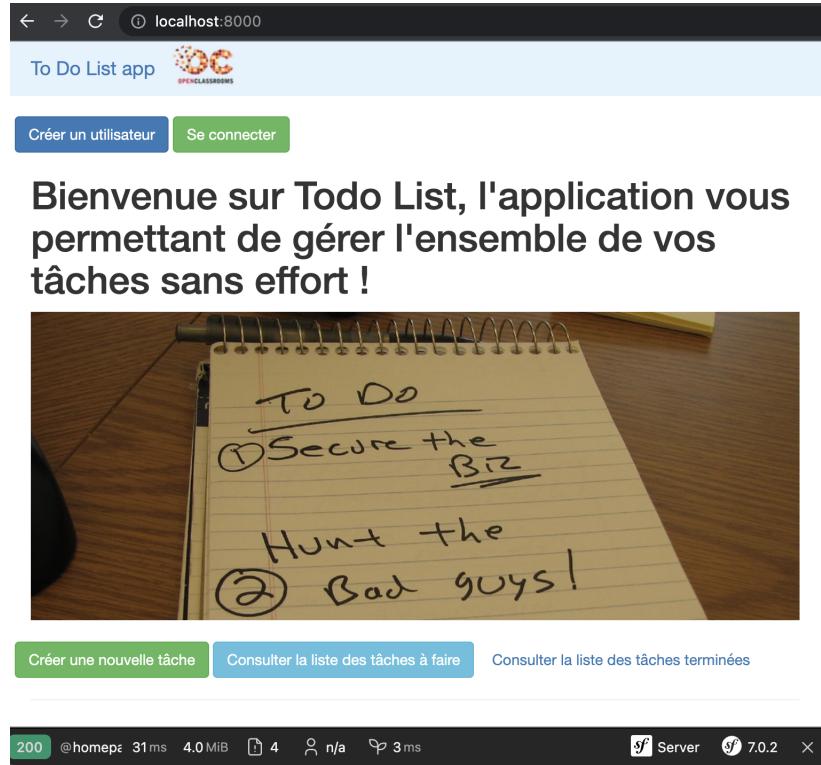
Copyright © OpenClassrooms

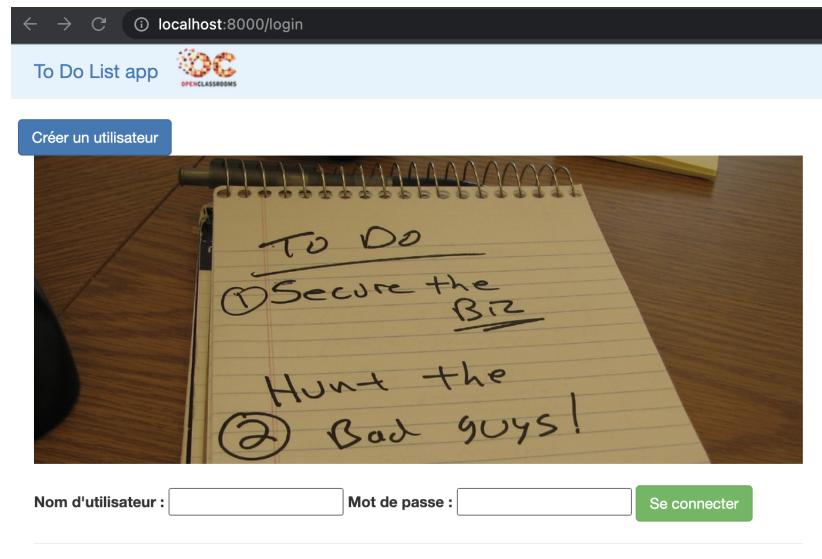


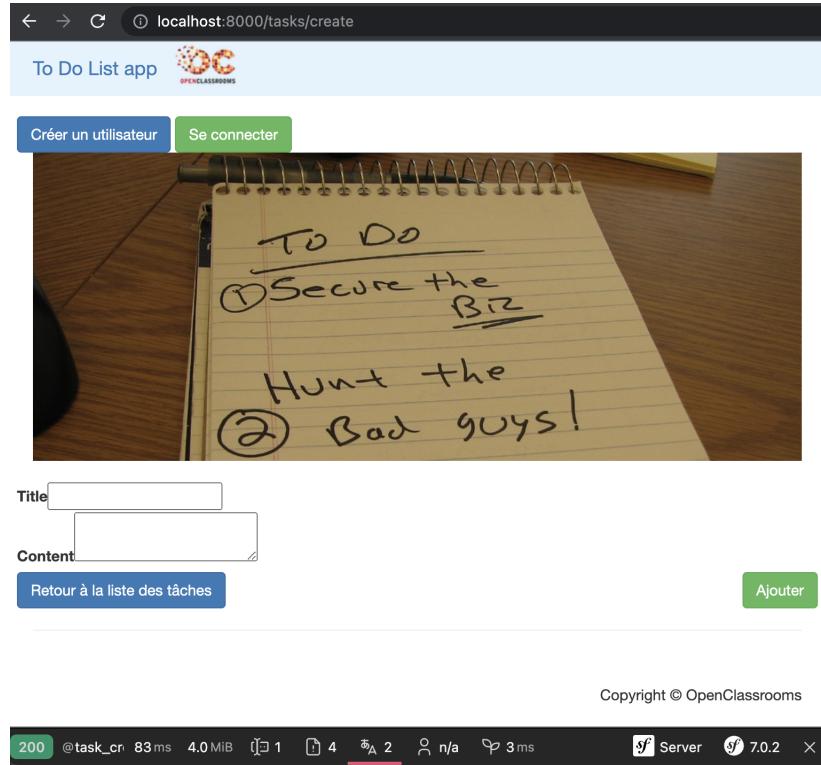


2 -.2.2 Projet avec Symfony 7 après modification

En utilisant le Profiler de Symfony sur le repository du projet avec Symfony 7 après avoir ajouter les modifications et écrit les tests unitaires et fonctionnels, nous obtenons le résultat suivant :







2 -2.3 Comparaison

Globalement, le temps de rendu de la page est entre 30 et 85 millisecondes, il peut varier en fonction des performances de la machine mais il reste similaire entre le projet avant modification et après modification.

La consommation de la mémoire a, quand à elle, diminué après modification, sans doute grâce à de l'optimisation de code.

Le temps de rendu des pages est aussi similaire entre le projet avant modification et le projet après modification. Il est entre 3 et 7 millisecondes pour le projet avant modification et d'environ 2-3 millisecondes pour le projet après modification.

Conclusion

Le projet a été réalisé from scratch pour repartir sur la dernière version de Symfony, puis le code a été copier coller en changeant certaines fonctions pour correspondre à la dernière version de Symfony grâce à la documentation officielle de Symfony. L'audit et la performance ont donc été réalisé avant et après modification sur la dernière version de Symfony.

L'audit et la performance du projet, avant et après modification, est à peu près similaire. Cela est du au fait qu'il n'y a pas beaucoup de modification apportées au projet.