

Documentation technique

ToDo and Co

Version : 2.0

Sommaire

1. [Introduction](#)
2. [Process de qualité du code](#)
 - 2.1. [Existant et correctif](#)
3. [Performance de l'application](#)
 - 3.1. [Blackfire](#)

Introduction

L'entreprise vient tout juste d'être montée, et l'application a dû être développée à toute vitesse pour permettre de montrer à de potentiels investisseurs que le concept est viable.

Le choix du développeur précédent a été d'utiliser le framework PHP Symfony.

Notre rôle ici est donc d'améliorer la qualité de l'application soit les tâches suivantes :

- l'implémentation de nouvelles fonctionnalités ;
- la correction de quelques anomalies ;
- et l'implémentation de tests automatisés.

Il est également demandé d'analyser le projet grâce à des outils vous permettant d'avoir une vision d'ensemble de la qualité du code et des différents axes de performance de l'application.

Il n'est pas demandé de corriger les points remontés par l'audit de qualité de code et de performance.

Process de qualité

1.1. Existant et correctif

Le code de base était sous **Symfony 3.1** qui n'est **plus maintenu**, il a fallu **migrer vers 5.4** qui est la dernière version **stable et sûre** (LTS) obligatoire pour le projet. Bien entendu plusieurs composants ont été adaptés pour faire tourner l'application sans erreur du au dépréciation et renommage de certains éléments.

Ensuite les **correctifs concernant l'attache d'une tâche à un utilisateur et le choix du rôle pour un utilisateur** on était intégré ainsi qu'une **nouvelle fonctionnalité** concernant les **restrictions selon le rôle de l'utilisateur**.

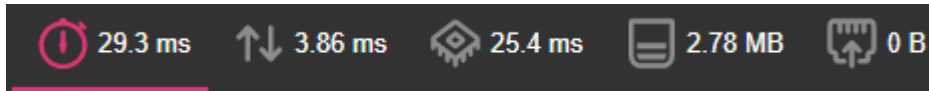
Enfin on à **couvert le code a 70% avec des tests automatisé** qui est le minimum requis pour ce projet en perspective des évolution avenir. Pour être sur de la qualité on à réalisé une **analyse de la qualité du code sur Codacy** cherchant une **note de B** au minimum. **L'outil Blackfire** nous a permis de **mesurer les performances puis de les adapter** au besoin de rapidité et de confort.

Pour plus d'information sur l'analyse du code et sur sa couverture ce rendre sur le README.md à la racine du projet.

Performance de l'application

1.2. Blackfire

L'outil **Blackfire** à l'avantage d'avoir une **empreinte minimale** et ne requiert **aucune modification** du code. Il nous a permis de **mesurer les performances puis de les adapter** au besoin de rapidité et de confort.



Nous retrouvons en haut de la page le nom du profil, ainsi que les métriques :

- le temps de génération de la page en PHP, découpé en temps d'accès au processeur et temps d'accès au système de fichiers ;
- la mémoire utilisée, en mégabytes ;
- l'utilisation du réseau.