

ToDo & Co - Todolist

Audit de performance



Nicolas Halberstadt

Table Des Matières

ToDo & Co - Todolist	1
Audit de performance	1
1. Contexte	3
2. Qualité de code	4
3. Audit de performance	5
3.1.Audit	5
3.2.Optimisation	6

1. Contexte

L'application a été développée a dû être développée à toute vitesse pour permettre de montrer à de potentiels investisseurs que le concept est viable.

Lors de la mise en place du MVP, l'environnement de développement était le suivant:

- `Symfony v.3.1.*`
- `PHP v.>=5.5.9`
- `Doctrine bundle v.^1.6`

Plusieurs version de dépendances de l'application étaient dépassées voire archivées, l'environnement a donc été mis à jour lors de l'amélioration de l'application vers l'environnement ci-après:

- `Symfony v.4.4.*`
- `PHP v.7.4.16`
- `Doctrine bundle v.2.4.2`
- Ainsi que les autres éléments nécessaires au bon développement de la nouvelle version de l'application

La version 4.4 de Symfony a été choisie puisqu'étant la dernière version de support à long terme la plus proche de l'installation de base, cela paraissait être le meilleur compromis.

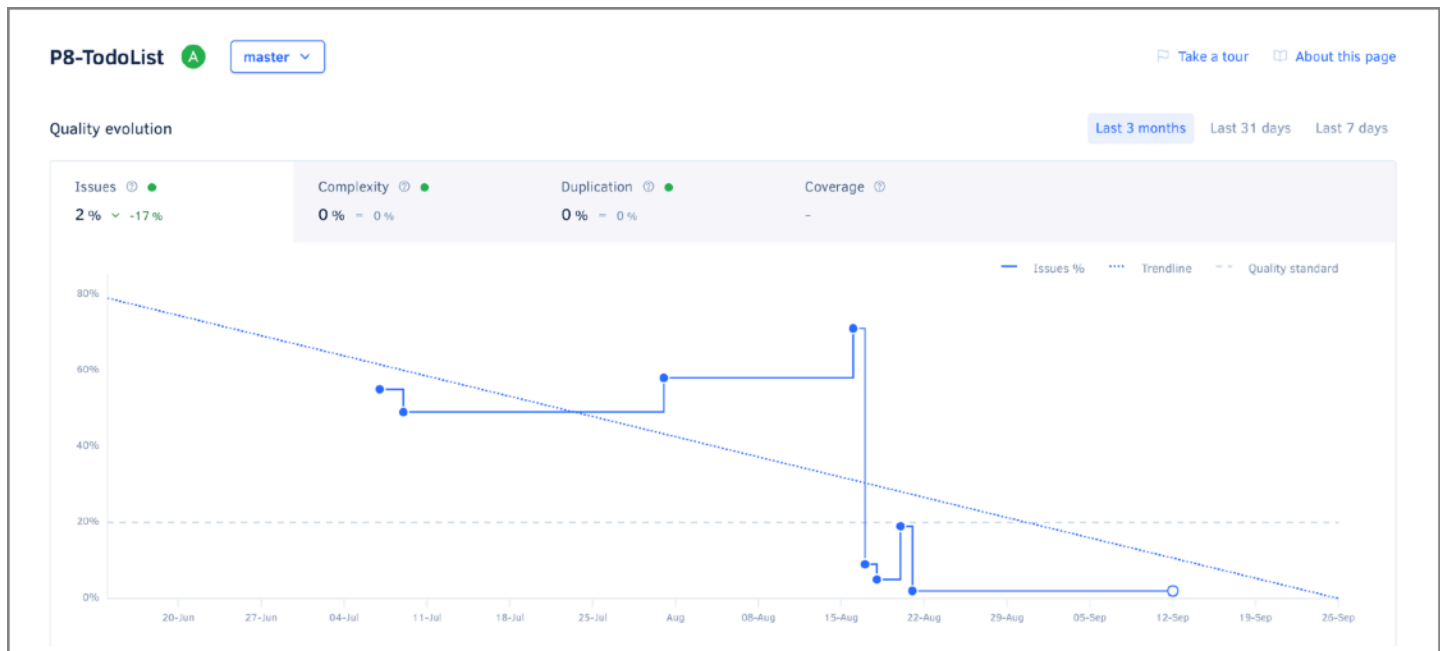
Tous les résultats suivant ont été obtenus après la mise à jour des dépendances du projet.

2. Qualité de code

Les analyses suivantes proviennent de l'outil Codacy:



Lors de la première analyse, Codacy a remonté un taux d'erreur de 55%. Depuis, plusieurs corrections ont été faites et un tri des fichiers à analyser a été mis en place.



Dans l'analyse la plus récente, l'outil remonte un taux de 2% d'issues, 0% de complexité et 0% de duplication de code. Codacy donne donc une note de A au projet pour 990 lignes de code analysées, qui est la meilleure note atteignable par l'outil Codacy.

Lien vers le repository Codacy:

<https://app.codacy.com/gh/NicolasHalberstadt/P8-ToDoList/dashboard?branch=master>

3. Audit de performance

3.1. Audit

L'audit de performance a été fait avec Blackfire. Lors de cette analyse, l'outil nous donne plusieurs indications comme l'I/O wait (temps d'attente du système pour l'écriture ou la lecture de données), le nombre de requête SQL utilisées, la mémoire consommée ou encore le temps d'exécution.



Voici les différentes analyse et leur résultat composant l'audit de performances.

Aucun amelioration n'a été effectuée en amont de l'analyse.

- La page d'accueil:

<https://blackfire.io/profiles/501ae437-77e0-4f6a-83a2-19dc5d8af92c/graph>



200 GET https://127.0.0.1:8000/
 Created 2 minutes ago by halberstadtnicolas@gmail.com



 ↳ 0 rq ⌚ 944 ms ↕ 109 ms 💎 835 ms 📄 21.3 MB 🗂 1.12 kB 🌐 0 µs / 0 rq ⌚ 2.22 ms / 3 rq 📅 Timeline

- La soumission du formulaire de connexion:

<https://blackfire.io/profiles/046ebe97-3fbe-4439-b9be-4e1a6e29e624/graph>



302 POST https://127.0.0.1:8000/login_check
 Created 1 hour ago by halberstadtnicolas@gmail.com



 ↳ 0 rq ⌚ 2.35 s ↕ 231 ms 💎 2.12 s 📄 23.3 MB 🗂 1.72 kB 🌐 0 µs / 0 rq ⌚ 83.2 ms / 4 rq 📅 Timeline

- La page des taches a faire:

<https://blackfire.io/profiles/3607e2f5-3f90-4224-94af-282a6a2e349e/graph>



200 GET https://127.0.0.1:8000/tasks/todo
 Created 1 hour ago by halberstadtnicolas@gmail.com



 ↳ 0 rq ⌚ 1.31 s ↕ 33.8 ms 💎 1.27 s 📄 24.6 MB 🗂 3.67 kB 🌐 0 µs / 0 rq ⌚ 2.95 ms / 5 rq 📅 Timeline

- La soumission du formulaire de creation d'une tâche:

<https://blackfire.io/profiles/ecf78cf3-4e69-4d46-90d9-b7c57e72f9ff/graph>

302 POST https://127.0.0.1:8000/tasks/create
 Created 1 hour ago by halberstadtnicolas@gmail.com






 ↳ 0 rq ⌚ 1.84 s ↕ 376 ms 💎 1.47 s 📄 27.8 MB 🗂 1.9 kB 🌐 0 µs / 0 rq ⌚ 10 ms / 5 rq 📅 Timeline

- La suppression d'une tâche:

<https://blackfire.io/profiles/43b42ee8-f40a-461e-9298-753bc9cc459d/graph>

302 GET https://127.0.0.1:8000/tasks/71/delete

Created 1 hour ago by halberstadtnicolas@gmail.com







↳ 0 rq
⌚ 1.16 s
↕ 88.2 ms
💎 1.07 s
📄 23.7 MB
📦 2.47 kB
🌐 0 µs / 0 rq
🕒 49.8 ms / 6 rq
📅 Timeline

- Marquer une tache comme faite:

<https://blackfire.io/profiles/4cc79fa3-d11e-4653-b5fc-43cb3a5ba4bf/graph>

302 GET https://127.0.0.1:8000/tasks/72/toggle

Created less than a minute ago by halberstadtnicolas@gmail.com




↳ 0 rq
⌚ 1.01 s
↕ 68.8 ms
💎 938 ms
📄 23.6 MB
📦 2.54 kB
🌐 0 µs / 0 rq
🕒 41.7 ms / 6 rq
📅 Timeline

3.2.Optimisation

Afin de tenter d'optimiser les différentes valeurs de l'audit, plusieurs pistes peuvent être envisagées :

3.2.1.APCu cache

Cette extension PHP fournit une association de clé / valeur dans PHP lui-même. Il peut être utilisé pour le cache d'une application (voir [Symfony Cache Component](#)) ou pour optimiser l'autoload de Composer.

3.2.2.L'utilisation d'OPCache

Cette extension php optimise et met en cache le opcode généré par PHP.