

Email :  Mot de passe :  [Se connecter](#)

Copyright © OpenClassrooms

# Audit de qualité du code & performance de l'application

ToDo & Co



Version : 0.1

Date de la dernière mise à jour : 16 février 2022

# Sommaire

<b>Contexte</b>	<b>3</b>
<b>Audit de performance</b>	<b>3</b>
<b>Audit de qualité du code</b>	<b>6</b>

# 1. Contexte

Les fondateurs de ToDo & Co souhaitent pérenniser le développement de l'application. Cela dit, ils souhaitent dans un premier temps faire un état des lieux de la dette technique de l'application.

Dans ce cadre, un audit de code et de performance est réalisé sur l'application où les conclusions et recommandations seront exprimées dans ce document.

## 2. Audit de performance

### 2.1. Etat des lieux

L'application récupérée est sous symfony 3.1.

Pour permettre de tester l'application et de réaliser l'audit dans les meilleures conditions, une mise à jour vers une version LTS de symfony est indispensable.

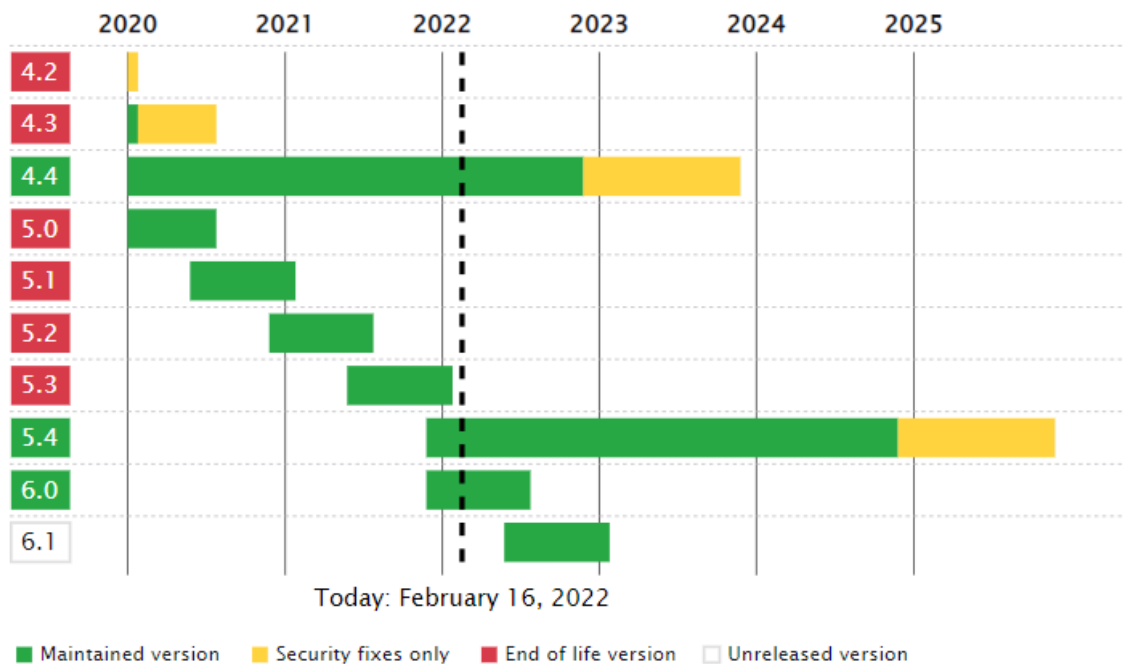
### 2.2. Configurations applicatives et pré-requis

L'audit de performance a été réalisé en local avec la configuration suivante :

- Symfony v5.4.2
- Symfony local server v4.26.6
- PHP v7.4.19
- MariaDB v10.4.19

L'application étant sur une version 3.2 de symfony qui n'est plus maintenue, une montée de version en 5.4 a été effectuée permettant d'être sur une version LTS ayant un support de longue durée ainsi que des corrections contre des failles de sécurité potentielles .

## Symfony Releases Calendar



Cette montée de version a permis de supprimer les dépréciations multiples du projet.

Info. & Errors	1	Deprecations	12	Debug	32	PHP Notices	0	Container	582
----------------	---	--------------	----	-------	----	-------------	---	-----------	-----

Log messages generated by using features marked as deprecated.

Time	Channel	Message
13:56:36	php	User Deprecated: Symfony\Component\HttpKernel\Kernel::loadClassCache() is deprecated since Symfony 3.3, to be removed in 4.0. <a href="#">Show context</a> <a href="#">Show trace</a>
13:56:36	php	User Deprecated: Symfony\Component\HttpKernel\Kernel::doLoadClassCache() is deprecated since Symfony 3.3, to be removed in 4.0. <a href="#">Show context</a> <a href="#">Show trace</a>
13:56:36	php	User Deprecated: Creating Doctrine\ORM\Mapping\UnderscoreNamingStrategy without making it number aware is deprecated and will be removed in Doctrine ORM 3.0. <a href="#">Show context</a> <a href="#">Show trace</a>
13:56:36	php	User Deprecated: Call Form::isValid() with an unsubmitted form is deprecated since Symfony 3.2 and will throw an exception in 4.0. Use Form::isSubmitted() before Form::isValid() instead. <a href="#">Show context</a> <a href="#">Show trace</a>
13:54:56	-	Using the unquoted scalar value "event" is deprecated since Symfony 3.3 and will be considered as a tagged value in 4.0. You must quote it in "D:\Documents\Dev\OPC\P08\todoHS\projets-TodoList\app/config/config_dev.yml" on line 20. <a href="#">Show context</a> <a href="#">Show trace</a>
13:54:56	-	Using the unquoted scalar value "event" is deprecated since Symfony 3.3 and will be considered as a tagged value in 4.0. You must quote it in "D:\Documents\Dev\OPC\P08\todoHS\projets-TodoList\app/config/config_dev.yml" on line 23. <a href="#">Show context</a> <a href="#">Show trace</a>
13:54:56	-	Using the unquoted scalar value "doctrine" is deprecated since Symfony 3.3 and will be considered as a tagged value in 4.0. You must quote it in "D:\Documents\Dev\OPC\P08\todoHS\projets-TodoList\app/config/config_dev.yml" on line 23. <a href="#">Show context</a> <a href="#">Show trace</a>
13:54:56	-	The "framework.trusted_proxies" configuration key has been deprecated in Symfony 3.3. Use the Request::setTrustedProxies() method in your front controller instead. <a href="#">Show context</a> <a href="#">Show trace</a>
13:54:56	-	Not setting "logout_on_user_change" to true on firewall "main" is deprecated as of 3.4, it will always be true in 4.0. <a href="#">Show context</a> <a href="#">Show trace</a>
13:54:56	-(2 times)	Autowiring-types are deprecated since Symfony 3.3 and will be removed in 4.0. Use aliases instead for "Psr\Log\LoggerInterface". <a href="#">Show context</a> <a href="#">Show trace</a>

Cette montée de version a été réalisée grâce à symfony-flex et permet au projet d'avoir une architecture conforme au préconisation d'une application symfony en version 5.4

### 2.3. Etat des lieux et optimisation

Pour réaliser l'audit de performance de l'application, l'outil Blackfire a été utilisé permettant de mettre en avant certains points à optimiser :

Du fait de l'utilisation de l'autoload de composer, celui-ci va vérifier le système de fichier (suivant PSR-4 et PSR-0) pour résoudre le nom de la classe et ceci peut ajouter de la lenteur. Cette utilisation basique de l'autoload de composer est recommandée en environnement de développement mais pas en environnement de production. Une optimisation de celui-ci est donc recommandée.

#### **Optimisations :**

- Une optimisation de l'autoloader de symfony a été réalisée via la commande suivante :

```
composer dump-autoload --no-dev --classmap-authoritative
```

- Une seconde optimisation a été réalisée via OPcache PHP avec la configuration suivante :

```
[opcache]
```

```
opcache.enable=1
```

```
opcache.memory_consumption=256
```

```
opcache.max_accelerated_files=20000
```

```
opcache.validate_timestamps=0
```










```
opcache.revalidate_freq=60
```

- Passage de l'application en "prod"










#### **Résultat :**

Vous trouverez ci-dessous le comparatif des différentes routes de l'application avant et après optimisation.










### Route : **/login**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 /login  312 ms  13 MB	 /login opti  225 ms  2.65 MB	 Comparison  -28%  -79%









### Route : **/**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 homepage  351 ms  15.3 MB	 homepage opti  243 ms  3.28 MB	 Comparison  -31%  -79%










### Route : **/tasks**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 /tasks  355 ms  17 MB	 /tasks opti  251 ms  3.6 MB	 Comparison  -29%  -79%










### Route : **/tasks/create**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 384 ms  20.1 MB	 /tasks/create opti  253 ms  4.64 MB	 Comparison  -34%  -77%










### Route: **/tasks/{id}/edit**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 /task/{id}/edit  385 ms  20.3 MB	 /task/{id}/edit opti  256 ms  4.66 MB	 Comparison  -33%  -77%










### Route : **/users**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 /users  347 ms  16.9 MB	 /users opti  245 ms  3.38 MB	 Comparison  -29%  -80%

Route : **/users/create**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 /users/create  389 ms  20.6 MB	 /users/create opti  259 ms  4.87 MB	 Comparison  -33%  -76%

Route : **/users/{id}/edit**

Résultat sans optimisation	Résultat après optimisation	Comparatif
 /users/{id}/edit  387 ms  20.6 MB	 /users/{id}/edit ...  256 ms  4.87 MB	 Comparison  -34%  -76%

### **Bilan :**

Sur l'ensemble des routes analysées, le temps de chargement a été en moyenne réduit de -31.37% et l'utilisation de la mémoire de -77.87%.

Les différentes pages de l'application se chargent en moyenne en 248 ms.

## **3. Audit de qualité du code**

### **3.1. Etat des lieux et optimisation**

Concernant la qualité du code, vous trouverez ci-dessous l'analyse réalisée par Codacy permettant une analyse des erreurs de code ainsi que du respects des bonnes pratiques, ainsi que l'analyse réalisée via CodeClimate permettant d'analyser la maintenabilité du code de l'application.

Lors de la première analyse Codacy, une seule erreur a été reportée : un paramètre non utilisé dans le SecurityController

## src/AppBundle/Controller/SecurityController.php

Avoid unused parameters such as '\$request'.

Time to fix: 5 minutes 🕒

```
11  /**
12   * @Route("/login", name="login")
13   */
14   public function loginAction(Request $request)
15   {
16       $authenticationUtils = $this->get('security.authentication_utils');
```

Why is this an issue?

Avoid passing parameters to methods or constructors and then not using those parameters.

Ceci est dû à l'utilisation du service 'security.authentication\_utils' utilisé directement dans la fonction loginAction. On privilégiera l'injection de dépendance dans les versions plus récentes de symfony.

Comme indiqué lors de l'audit de performance, une mise à jour de l'application via symfony-flex a été réalisée permettant de passer de la version 3.1 à 5.4.

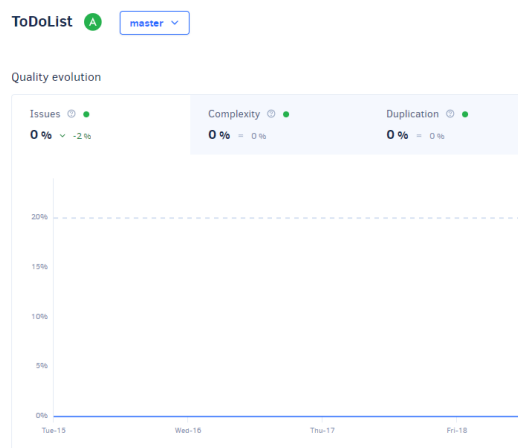
A la suite de cette montée de version nous avons procédé à la correction de l'erreur remontée ci-dessus mais aussi une correction et amélioration de l'application sur les points suivants :

- Amélioration de la sécurité de l'application suivant les recommandations données par le client
- Mise en place d'un système de rôles et de vérification de droit pour les utilisateurs
- Mise en place de la protection CSRF pour le formulaire d'authentification
- Correction des liens HS de l'application
- Correction de l'affichage de la liste des tâches à faire et des tâches terminées
- Étendre les controllers avec le AbstractController ( précédemment étendus de la classe Controller qui est déprécié depuis la version 3.3 de Symfony )
- Mise en place des liens entre les utilisateurs et les tâches suivant les souhaits du client.



A la suite de l'ensemble des modifications, vous trouverez ci-dessous le résultat des analyses de code :

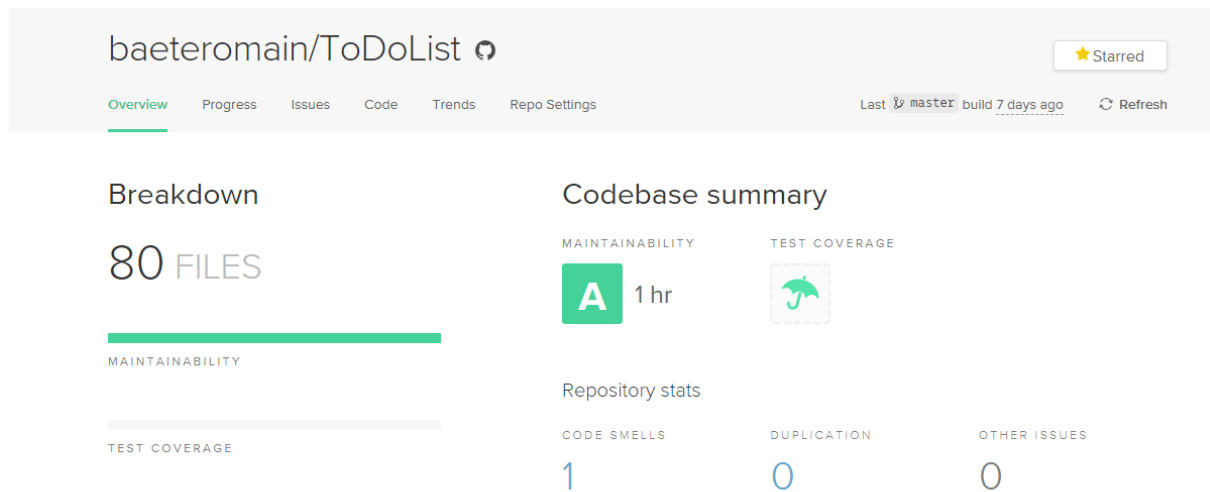
Codacy :



Un badge A ( soit la meilleure note possible ) est obtenu. Ceci confirme une bonne qualité de code de l'application ( les fichiers générés par le framework symfony ont été retirés de l'analyse )

<https://app.codacy.com/gh/baeteromain/ToDoList/dashboard>

CodeClimate :



Un badge A ( soit la meilleure note possible ) est obtenu. Ceci confirme une bonne maintenabilité du code sans mauvaises pratiques, duplications, ou autres problèmes. ( les fichiers générés par le framework symfony ont été retirés de l'analyse )

<https://codeclimate.com/github/baeteromain/ToDoList>