

Audit de qualité et de performance

Nous avons comparé l'application avant et après la mise à jour.

État des lieux avant modification

L'application présente de nombreux problèmes et une dette technique importante. Elle utilise **Symfony 3.0**. Sorti novembre 2015, cette version n'est plus maintenue. Comme on peut le voir sur la page [release](#) de cette version, la correction des bugs a cessé en juillet 2016 et la correction des problèmes de sécurité est terminée depuis janvier 2017. Cette version nécessite une version de PHP 5.5.9 ou supérieur, PHP 5 étant une version obsolète, puisqu'on en est à PHP 8.3 avec lequel l'application n'est pas compatible.

L'application présente en elle-même des problèmes de sécurité et d'ergonomie :

- La gestion des utilisateurs est accessible à n'importe quel utilisateur, comme on peut le voir dans la configuration (fichier security.yml). Cela représente un risque majeur de sécurité.
- Il y a des problèmes d'ergonomie : la navigation dans l'application est compliquée, l'affichage laisse à désirer (par exemple des problèmes d'alignement des boutons).
- Les tâches ne sont pas associées à l'utilisateur qui les a créées.
- L'application est lente à la navigation, ce que l'audit a confirmé

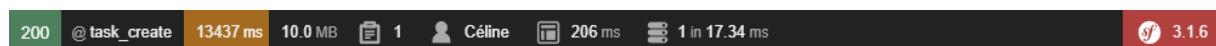
Audit de performance

Pour l'audit de performance, nous avons utilisé le profiler de Symfony qui a l'avantage d'être gratuit. Pour récupérer les données du profileur, sur la version originale du projet, on passe par le fichier **app_dev.php**. L'application étant obsolète, nous l'avons fait tourner sur un container docker basée sur l'image 7.0-apache.

Page d'accueil :



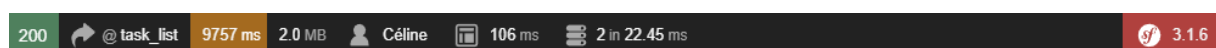
Page de création d'une tâche :



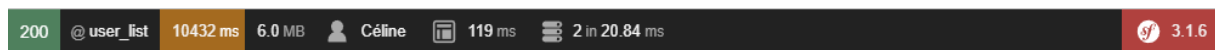
Page d'édition d'une tâche :



Page de listing des tâches :



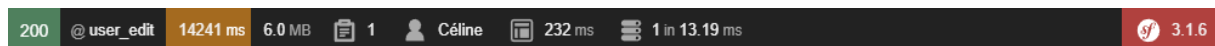
Page de listing des utilisateurs :



Page de création d'un utilisateur :



Page d'édition d'un utilisateur :



Sur l'ensemble des pages, on remarque que le temps de chargement est très long avec une augmentation de ce temps sur les pages de formulaire : l'outil l'indique en orange. Le temps de chargement des templates twig va de 106ms à 232 ms. Le profileur nous indique en outre que la version de Symfony est obsolète : en effet, la 3.1.6 n'est plus maintenue depuis longtemps et présentent donc d'importantes failles de sécurité. Par contre, le nombre de requête par page est bon avec un à deux requêtes.



Si on regarde le détail du chargement sur une page particulièrement lente, par exemple l'édition d'une tâche, on voit que deux tâches prennent beaucoup de temps de chargements : le firewall et le controller. Une mise à jour vers une version plus récente de Symfony permettra de réduire beaucoup ce temps de chargement.

Qualité de code

Pour la qualité de code, nous avons utilisé l'outil Codacy. Voici le lien pour consulter le rapport : <https://app.codacy.com/gh/CelineFoucart/project8-original/dashboard>.

Comme on peut le voir, le projet a une note de C et 285 issues, dont 252 de style de code et 30 de sécurité, ce qui est très problématique. On trouve par exemple :

- Des dépendances qui présentent des problèmes de sécurité : la version 5.4.3 de **swiftmailer/swiftmailer**.
- On a des utilisations des superglobales : **\$_SERVER**.
- On a des utilisations de la fonction **exit**, de **echo**.
- On a des echos de textes non échappés

En problèmes de sécurité medium, on a une utilisation directe de la fonction header.

```
III CRITICAL | Security
Direct use of $_SERVER Superglobal detected.

web/app_dev.php

13  if (isset($_SERVER['HTTP_CLIENT_IP'])
```

```
III CRITICAL | Security
Use of exit language construct is discouraged.

web/app_dev.php

18  exit('You are not allowed to access this file. Check '.__FILE__.' for more information.');
```

État des lieux après modification

La première étape a consisté à migrer l'application vers Symfony 7.1 et à la rendre compatible avec PHP 8.2. Cela a nécessité de lourdes modifications dans l'architecture des dossiers, la création de migrations pour générer les tables user et task.

La deuxième étape a consisté à revoir entièrement le système de création de compte et d'édition des utilisateurs. L'accès à la gestion des utilisateurs a été restreintes aux seuls administrateurs. Les tâches sont accessibles aux utilisateurs authentifiés. Une page profil a été ajoutée : ainsi, l'utilisateur courant ne peut modifier que les informations de son propre compte. Une page d'inscription a été créée. L'application présente désormais un système de rôle et a deux rôles : ROLE_ADMIN et ROLE_USER.

Enfin, la dernière étape a consisté à ajouter une relation entre la tâche et un utilisateur. L'utilisateur ne peut modifier que ses tâches. Les tâches sans auteur sont donc associées à un utilisateur virtuel anonyme et ne peuvent être modifiées que par l'administrateur. L'application ne permet plus de créer une tâche sans auteur.

Une fois ce travail effectué, des tâches unitaires et fonctionnaire ont été ajoutés jusqu'à atteindre une couverture de code de 95%.

Audit de performance

Pour l'audit de performance, nous avons aussi utilisé le profiler de Symfony après la mise à jour de l'application qui tournait avec le serveur de développement de Symfony.

Page d'accueil :



Page de création d'une tâche :



Page d'édition d'une tâche :



Page de listing des tâches :



Page de listing des utilisateurs :



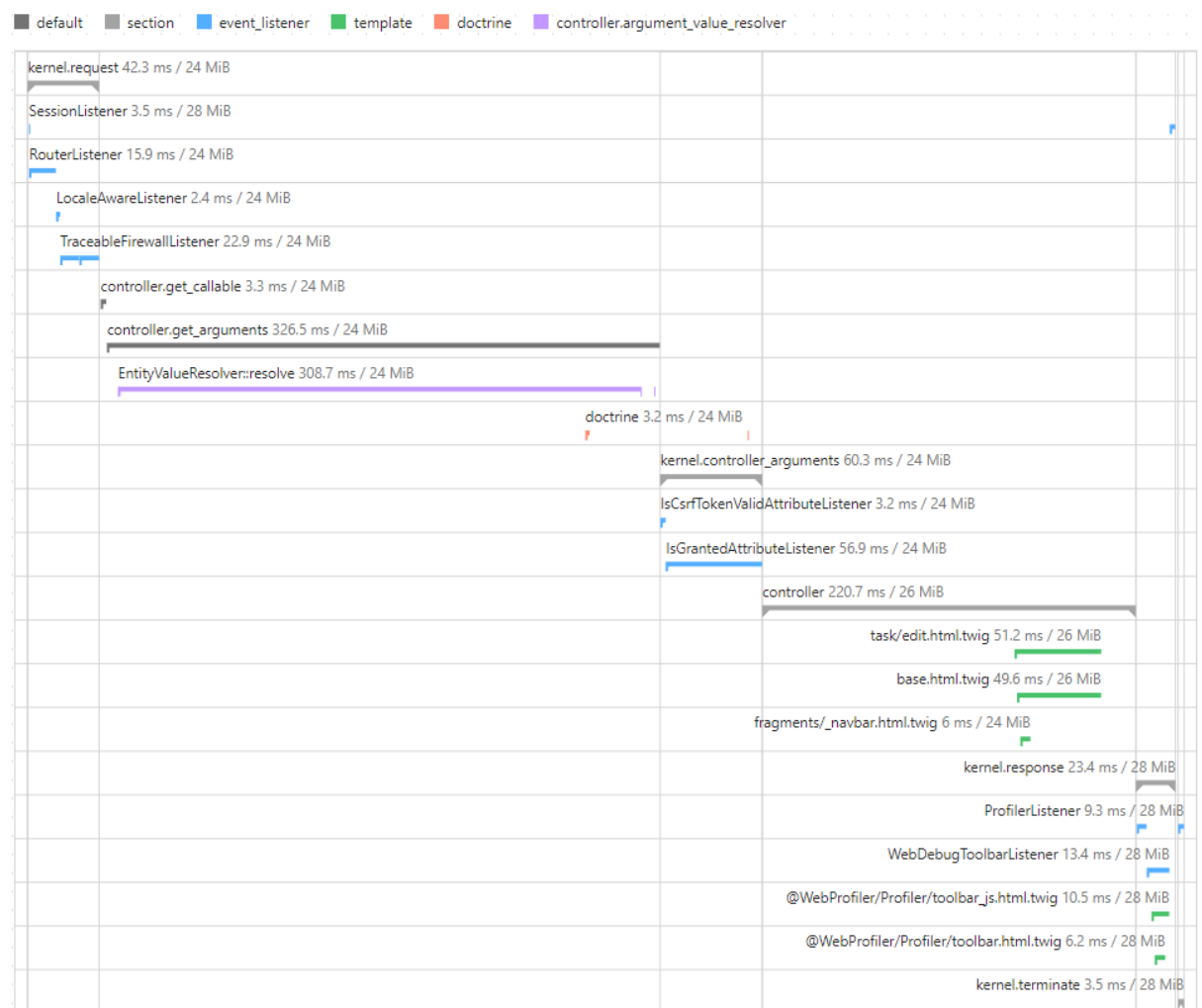
Page de création d'un utilisateur :

Page d'édition d'un utilisateur :

Profil (où l'utilisateur édite ses informations) :

Comme on peut le constater, au terme de la mise à jour, la version de Symfony n'est plus considérée comme obsolète. Le temps de chargement est bien meilleur alors même qu'on se trouve dans un environnement de développement. Si on compare le temps de chargement des templates twig, on est désormais de 19ms à 67ms.

Regardons à nouveau le détail du temps de chargement de la page d'édition d'une tâche, comme avant la mise à jour.



La mise à jour a amélioré grandement les temps de chargement. On voit par exemple que le firewall prend bien moins de temps.

Qualité de code

Un nouvel audit de qualité a été réalisé après les modifications. Voici le lien pour consulter le rapport : <https://app.codacy.com/gh/CelineFoucart/projet8-ToDoList/dashboard>.

Le projet a désormais une note B. Les problèmes de sécurité ont été réduit à leur minimum, grâce à la mise à jour du projet et de ses dépendances. La qualité de code a aussi été améliorée. Avant modification, il y avait 285 issues dont 30 de sécurité. Désormais, il y a 36 issues. Il reste 6 issues de sécurité. Elles concernent l'utilisation de `require` ou `require_once`, dans les fichiers de configuration et d'initialisation, généré par symfony. Il y a aussi l'utilisation de Superglobales dans le fichier `bootstrap.php`, utilisé pour les tests par le framework. Il ne me semblait pas opportun de toucher à ces fichiers.