

# ToDo&Co

Rapport de qualité et de performances

# SOMMAIRE

- 01 Contexte et besoin
- 02 PHP, Symfony et leurs versions
- 03 Qualité du code
- 04 Performances
- 05 Tests PHPUnit
- 06 Propositions d'améliorations

# CONTEXTE ET BESOIN

ToDo & Co est une startup qui 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.

Des levées de fonds ont été validées, et vont donc permettre à l'entreprise de se développer pleinement, et surtout l'application.

Ici, le choix a été fait de travailler avec le framework Symfony. Cependant, l'application n'a pas été optimisée à son développement, il a donc été demandé d'effectuer un audit, afin de pouvoir jauger la quantité de travail à venir.

---

## PHP, SYMFONY ET LEURS VERSIONS

L'application a été montée suivant les versions suivantes :

PHP 5.5.9

Symfony 3.1

Comme l'application est légère pour le moment, nous avons fait le choix ( intéressant sur le plan financier particulièrement ) de recommencer un projet à zéro dans les dernières versions en date, et de réimplémenter les fonctionnalités précédentes.

La nouvelle version de l'application a donc été montée suivant les versions suivantes :

PHP 8.1

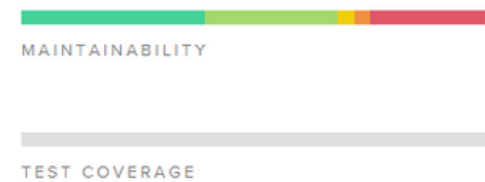
Symfony 6.3

# QUALITE DU CODE

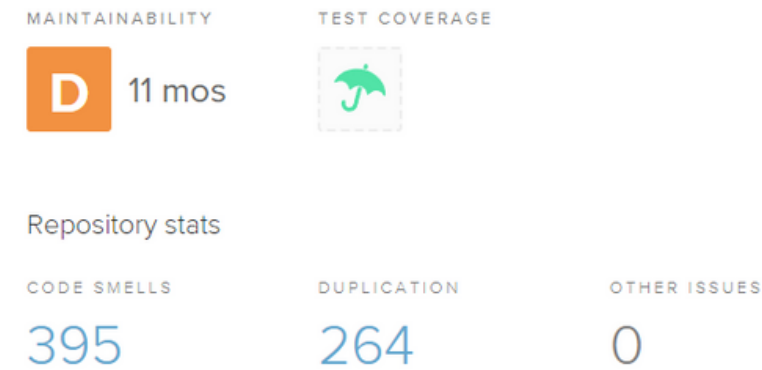
## Ancienne version

### Breakdown

597 FILES



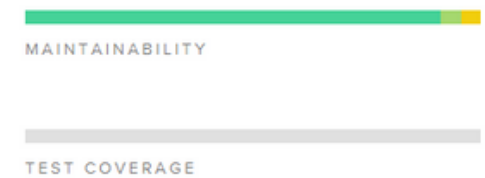
### Codebase summary



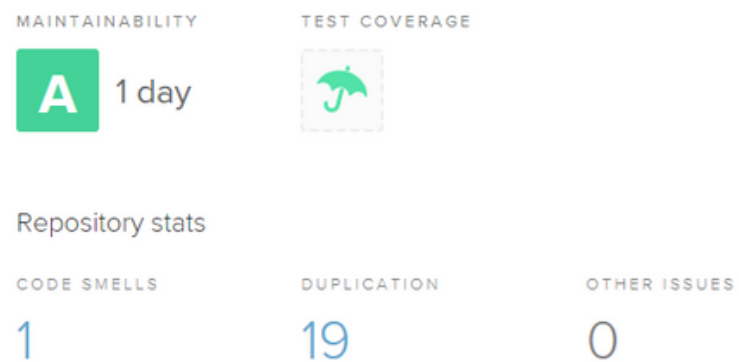
## Nouvelle version

### Breakdown

92 FILES



### Codebase summary

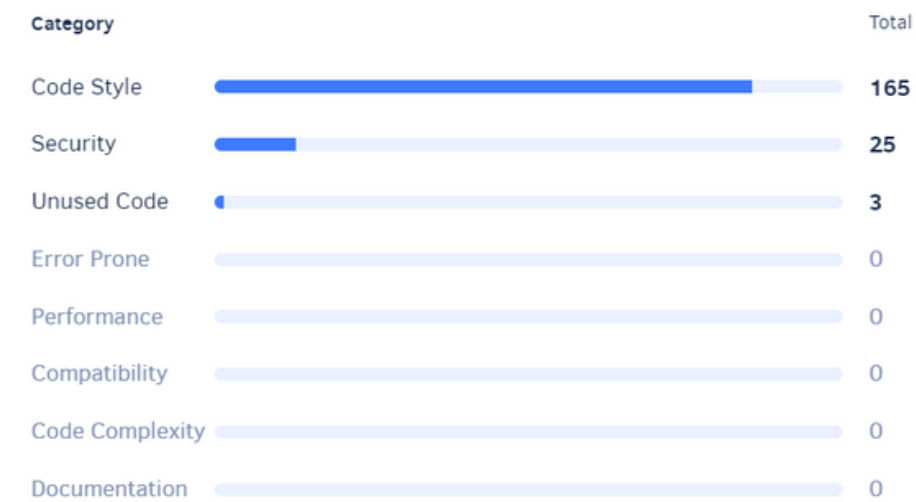


Sur CodeClimate, un tri a été fait ( minime ) sur les fichiers sur la nouvelle version, mais on peut tout de même constater un bien meilleur résultat concernant la maintenance, les duplications, et également le nombre de fichiers, ce qui joue énormément sur la qualité du code.

## Ancienne version

193 total issues

code quality **C**



## Nouvelle version

326 total issues

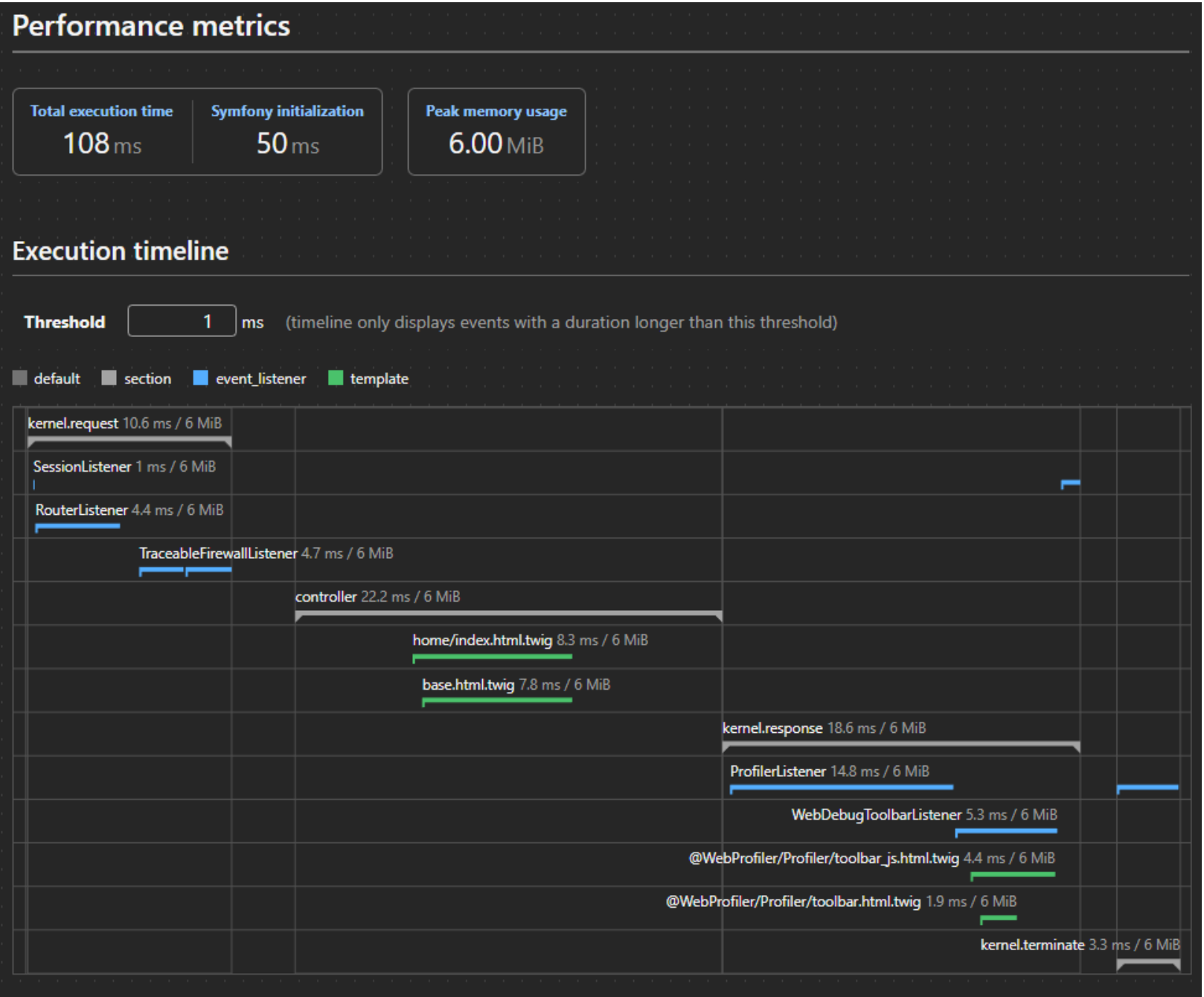
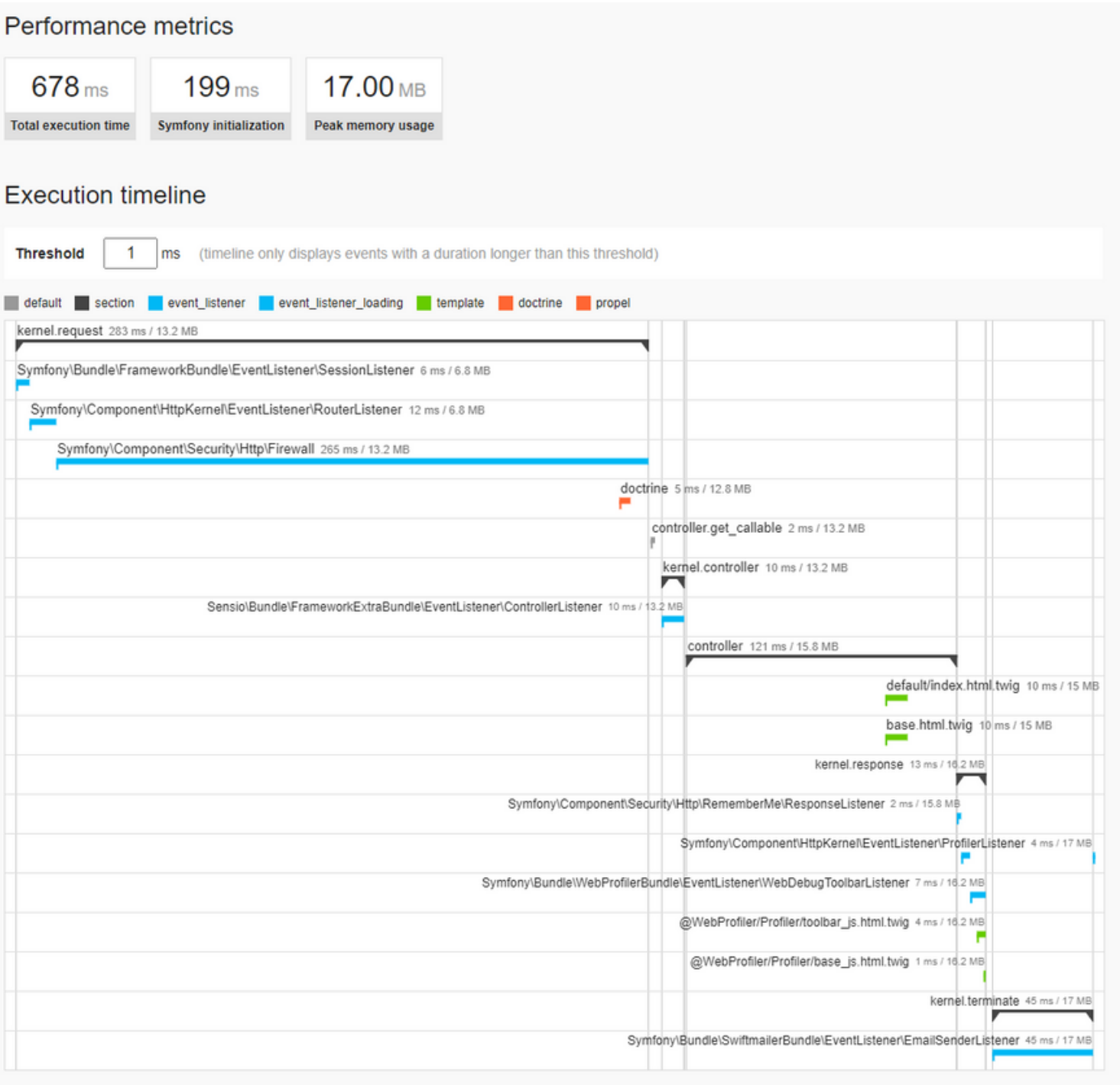
code quality **B**



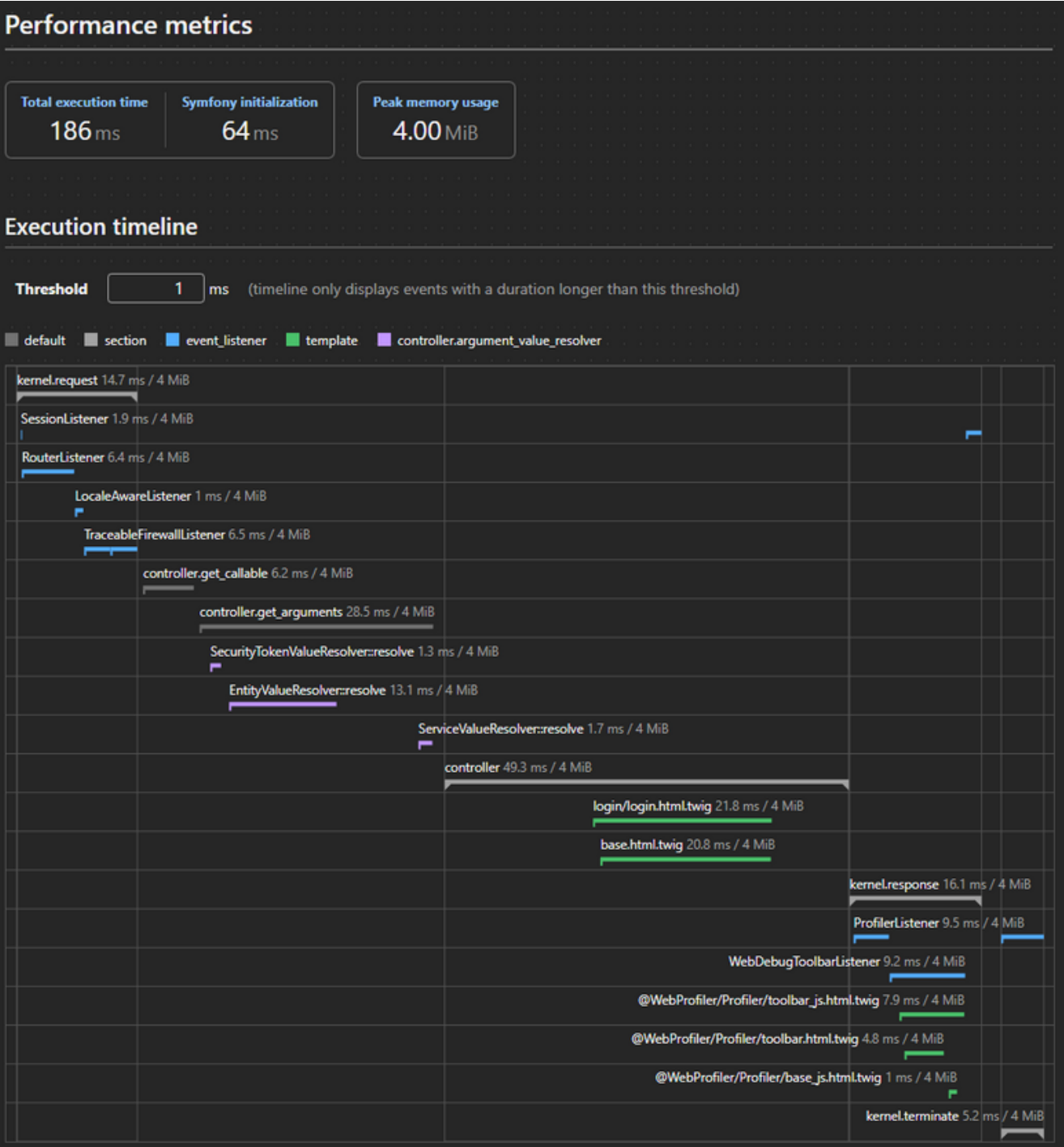
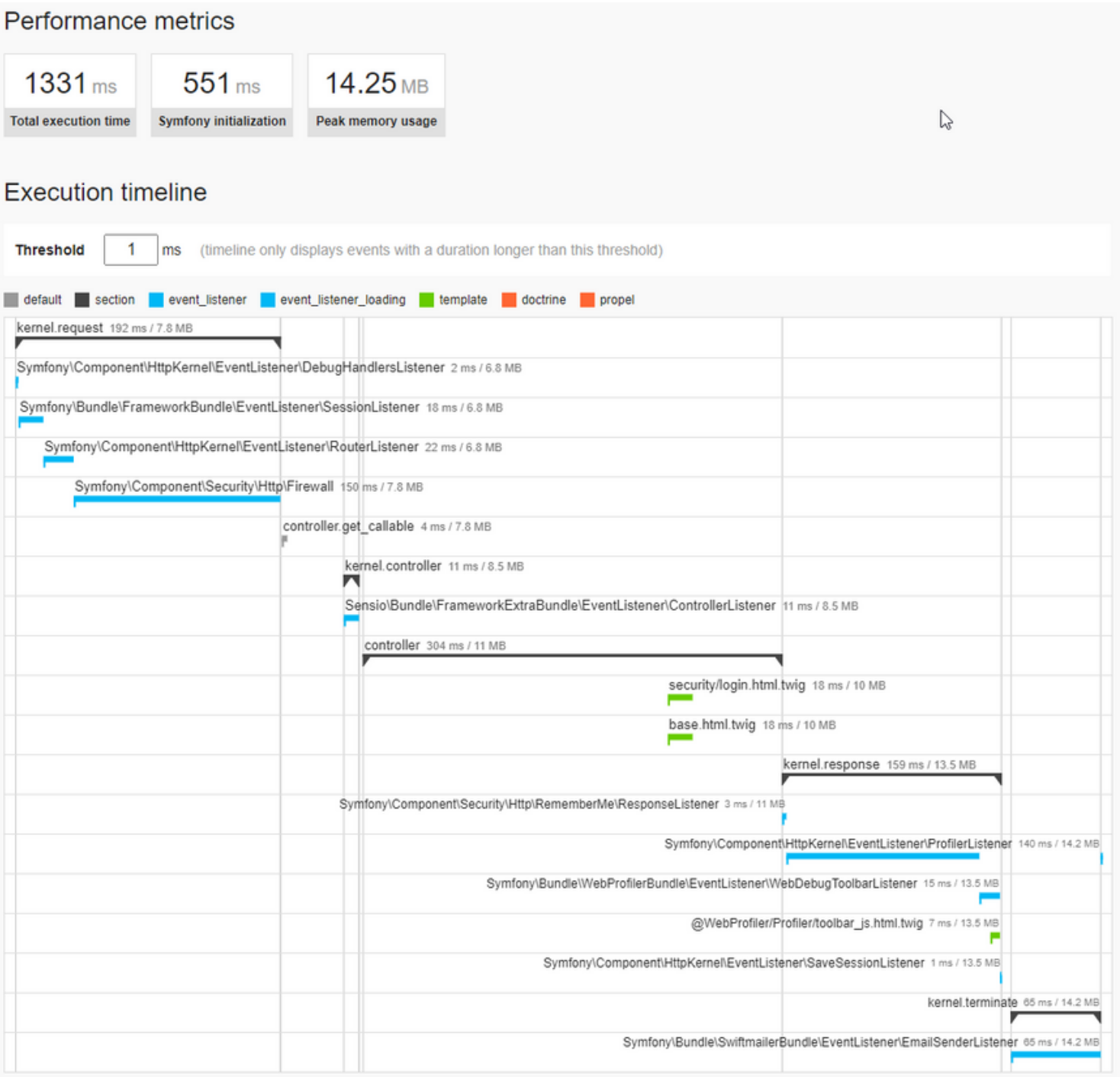
Sur Codacy, on peut remarquer plus d'erreurs au total, mais une note globale meilleure : en effet, il y a plus d'erreurs de style ( par exemple des espaces en trop dans le code ), mais il n'y a plus d'erreurs de sécurité.

Les erreurs de style seront corrigeables sans souci par la suite, moyennant un peu de temps, mais n'impactent pas le bon fonctionnement de l'application.

# PERFORMANCES : HOME PAGE

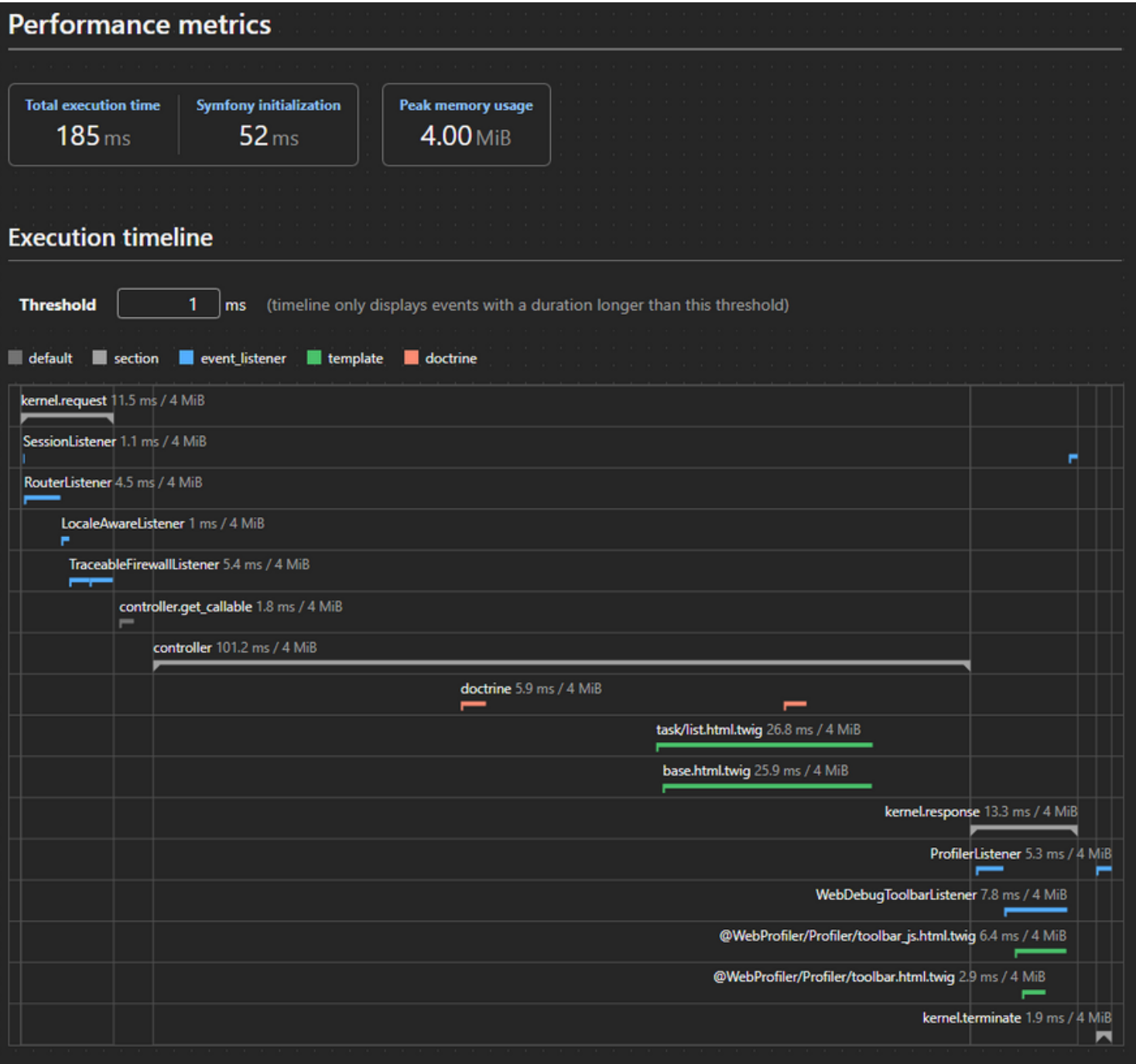
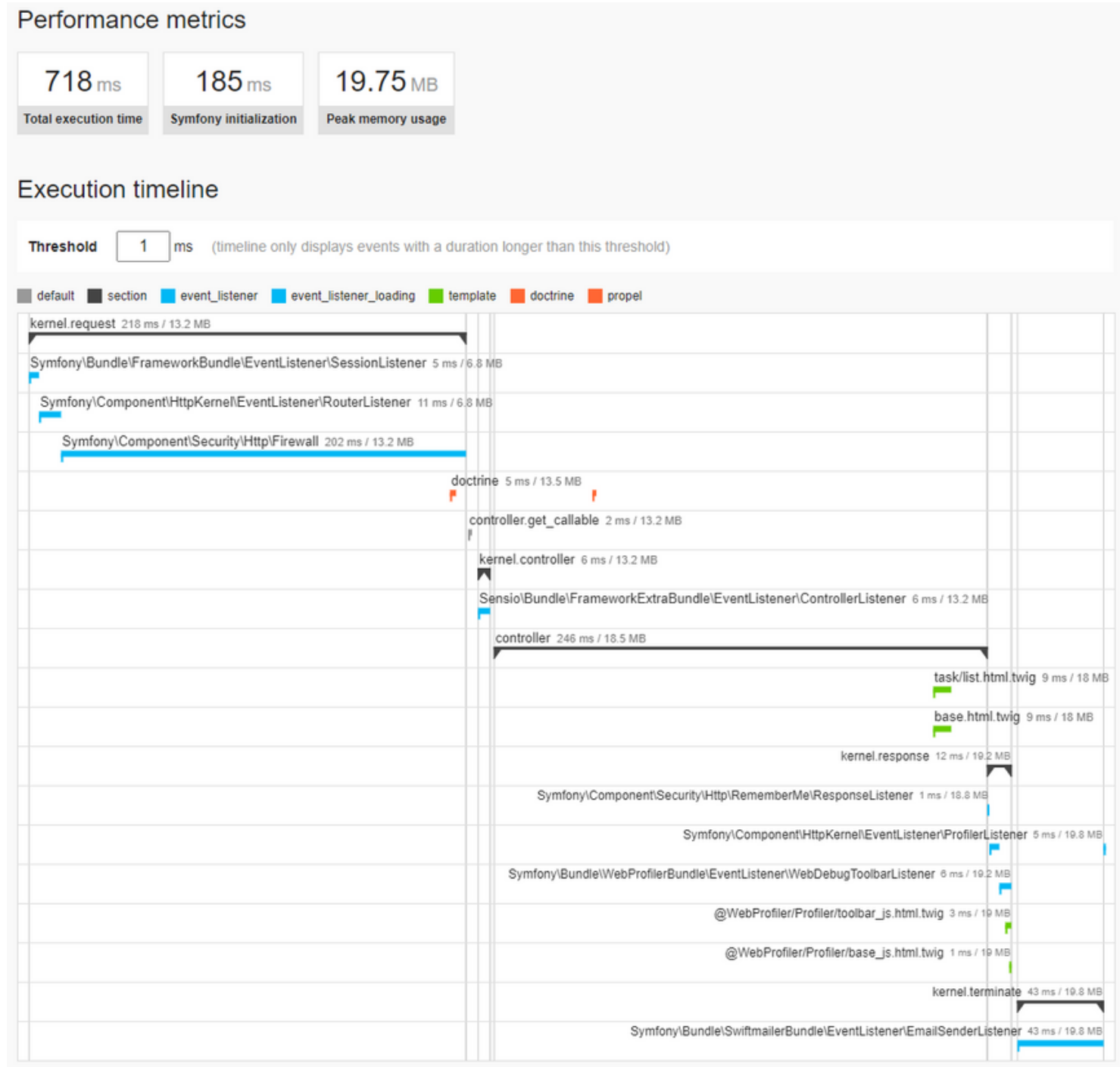


# PERFORMANCES : LOGIN PAGE

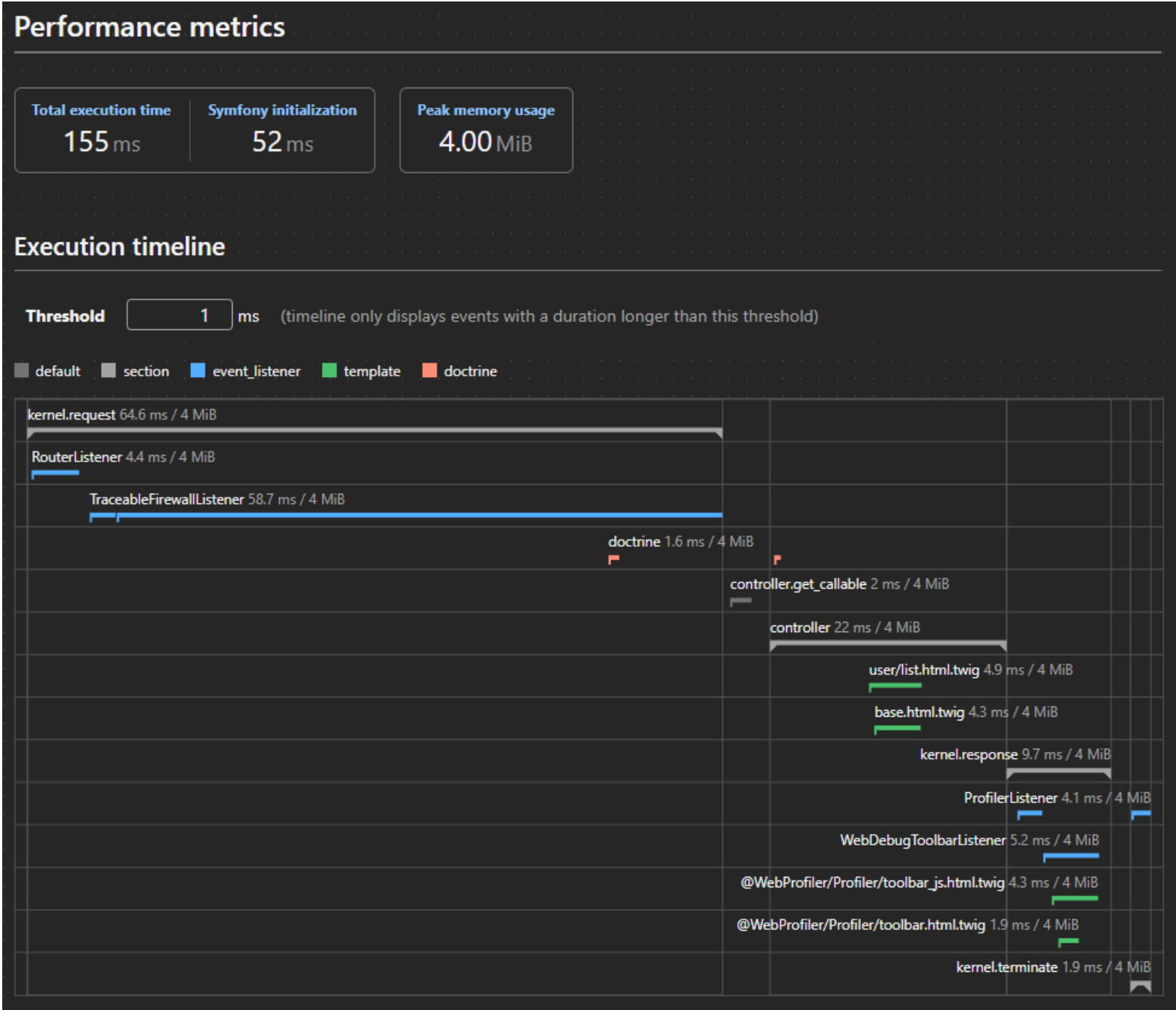
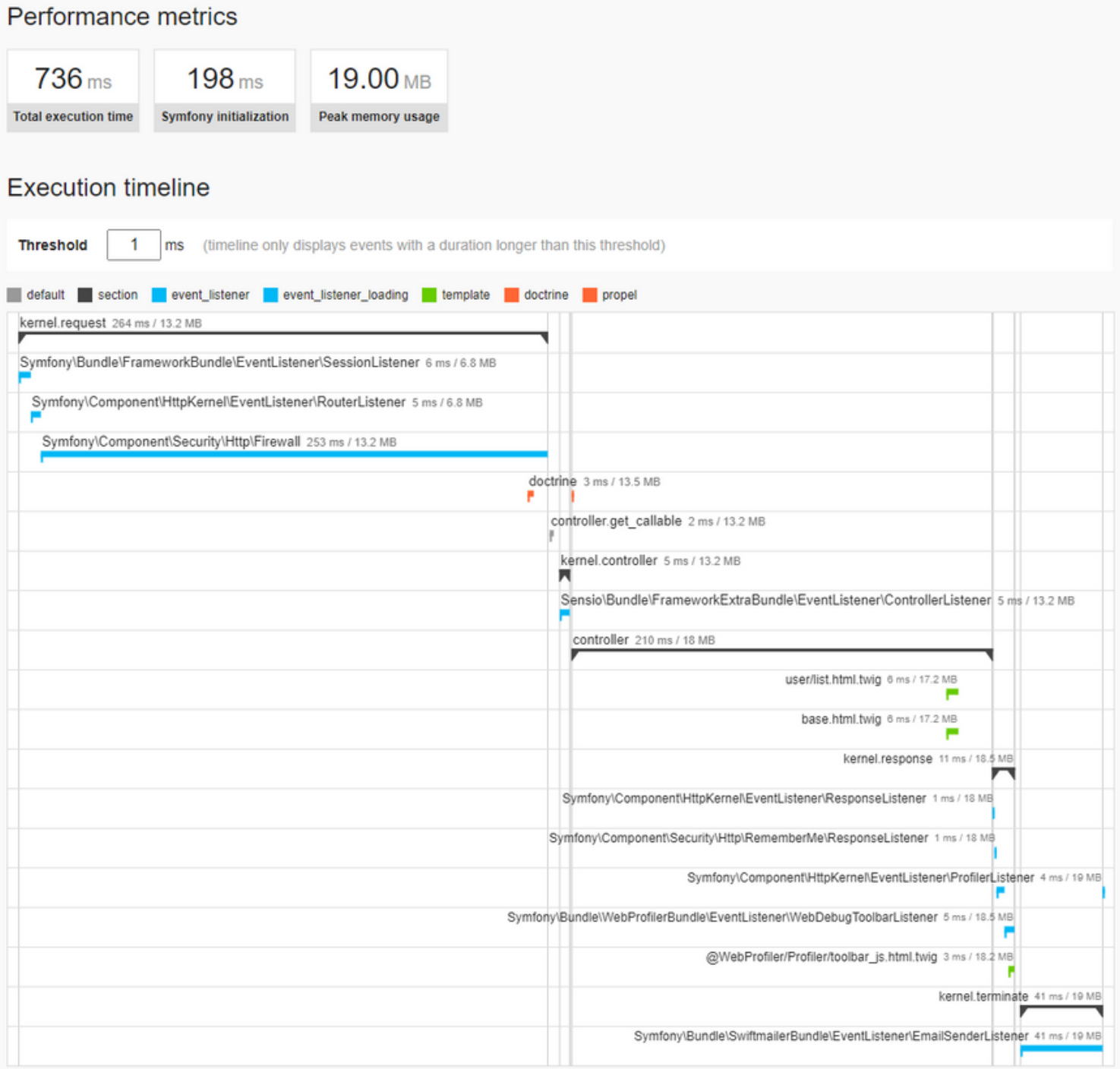




# PERFORMANCES : TASKS LIST



# PERFORMANCES : USERS LIST





# TESTS PHPUNIT

		Lines	
Total	<div><div></div></div>	81.87%	149 / 182
📁 Command	<div><div></div></div>	0.00%	0 / 15
📁 Controller	<div><div></div></div>	97.47%	77 / 79
📁 DataFixtures		n/a	0 / 0
📁 Entity	<div><div></div></div>	100.00%	44 / 44
📁 Form	<div><div></div></div>	100.00%	26 / 26
📁 Repository	<div><div></div></div>	11.11%	2 / 18
📄 Kernel.php		n/a	0 / 0

## Legend

**Low:** 0% to 50%   **Medium:** 50% to 90%   **High:** 90% to 100%

Nous avons mis en place des tests unitaires et fonctionnels grâce à PHPUnit, sur la nouvelle application.

Nous sommes arrivés rapidement à un taux de couverture des lignes de code de 81.87%.

L'obtention du 100% serait possible avec plus de temps, mais pas forcément utile dans le cas présent.

# PROPOSITIONS D'AMELIORATIONS

Un travail de montée en versions, ainsi que d'amélioration du code ( phpstan, pcp-cs-fixer, renommage des fonctions, mise en place des migrations ... ) a été fait, car gérable dans le temps imparti, néanmoins nous avons plusieurs propositions à faire pour la suite, qui ne pouvaient pas être effectuées sans validation et / ou rapidement :

- Si intéressant : passer sous un système d'inscription avec validation par un administrateur, avec une validation du mail par système de token ( plus sécurisé )
- Revoir les règles de sécurité au niveau du access\_control ( par exemple, la liste des tâches est visible sans être connecté, pareil pour la création de tâches ).
- Ajouter un bouton vers la liste des utilisateurs, uniquement visible aux administrateurs.
- Modifier la route pour le bouton des tâches en cours et des tâches terminées + revoir les methods associées : ici un seul lien fonctionne, et il mène vers toutes les tâches, aucun tri n'est fait.
- Ajouter une option de suppression ou d'archivage des utilisateurs.
- Changer le lien de " To Do List App " ou ajouter un bouton Accueil, car on a pas de lien actif nous permettant de revenir en page d'accueil.
- Créer des pages d'erreur personnalisées pour les erreurs 404, 500 etc.
- Mise en place d'une pipeline d'intégration continue, afin d'automatiser les tests et de s'assurer du respect des normes de la qualité du code. ( Lancement des tests à chaque commit, passage du php-cs-fixer, PHPStan, lint commands ... )
- Passer dès que possible à Symfony 6.4, afin de préparer la migration en 7 ( permettra de corriger en amont les deprecations etc )