

ToDo & Co – TodoList

Contribuer au projet

Cette application est basée sur un des framework PHP les plus populaires qui est le framework Symfony en sa version 5.2 et les affichages se font avec la librairie Twig.

Pour contribuer au projet, il est essentiel de savoir utiliser [git](#) et [github](#) (le projet étant hébergé sur [github](#)).

Pour en savoir plus sur [git](#), vous pouvez vous rendre sur la documentation officielle ou sur la documentation proposée par [github](#). Vous pouvez contribuer à ce projet, de plusieurs manières :

Correction de bugs ;

Proposition de nouvelle fonctionnalité ;

Etc.

Vous pouvez contribuer au projet de l'une des manières suivantes :

- Correction de bug
- Ajout d'une fonctionnalité

Dans tous les cas, il vous faut respecter les étapes indiquées dans le guide de soumission pour contribuer au projet:

Correction de Bug

Si vous remarquez un bug dans le code du projet vous pouvez faire une proposition de correction en créant une issue sur le dépôt [github](#) ou en proposant un [pull request](#) qui contient la correction directement.

Ajout de nouvelle fonctionnalité

Pour proposer une nouvelle fonctionnalité, proposer un [issue](#) d'abord sur [github](#) où vous décrivez clairement la fonctionnalité.

Après validation de l'issue vous pouvez soumettre le [pull request](#) de la fonctionnalité.

Procédure de soumission d'un pull request

#1 Créer une issue

Pour créer une issue, accédez aux problèmes du projet et créez un nouveau problème. Assurez-vous de suivre ces étapes :

- Vérifier que le problème n'a pas été résolu déjà
- Utiliser les étiquettes mises à disposition sur ce projet

- Soyez le plus précis et concis possible dans la description du problème (bug, question, nouvelle fonctionnalité...)
- Mentionner la version sur la quelle vous apportez la contribution
- Indiquer les librairies que vous utilisez

#2 Cloner le projet

La troisième étape consiste à cloner notre projet sur votre espace de travail local si vous ne l'avez pas encore fait. Alors faites la commande suivante:

```
git clone https://github.com/FrancoisNimpagaritse/p8_todolist
```

Suivez ensuite les étapes de configuration du projet.

#3 Créer une nouvelle branche

Pour créer une branche, dans un terminal à la racine du projet :

```
git checkout -b <branch-name>
git push -u origin <branch-name>
```

#4 Commit du code

Valider localement vos modifications de code avec la commande :

```
git commit -m "Message descriptif"
```

#5 Exécuter les tests

Dernière chose avant de soumettre votre travail, veuillez vérifier que votre code ne casse pas notre application. Pour vérifier, exécutez PHPUnit :

```
vendor/bin/phpunit
```

Pour avoir aussi la couverture du code lors de l'exécution du test:

```
vendor/bin/phpunit
```

Si des échecs de tests sont affichés, corrigez-les avant de continuer.

#6 Proposer un pull request

Dans le terminal du dépôt local exécuter la commande :

```
git push origin <branch-name>
```

Rendez-vous ensuite dans la section [Pull Requests](#) de notre projet et créez votre [pull request](#).

Une fois le pull request validé et la fusion de branche faite, récupérez la nouvelle version pour actualiser votre branche master en local

```
git branch -d master
```

```
git checkout -b master
```

Mettez à jour la branche master avec :

```
git fetch -vp
```

Supprimer la branche de l'issue avec :

```
git checkout -d <branch-name>
```

Règles à respecter pour la qualité du code

Développez ce que vous voulez mettre en place. Veuillez noter que votre travail ne sera pas accepté s'il ne respecte pas les règles suivantes :

- Respecter les normes PSR-1, PSR-2, PSR-12 du langage PHP. Ces recommandations font partie des normes de codage à respecter.
- Respecter les bonnes pratiques de Symfony.