



Progression sur la quête **Générer le CRUD**

(/tasks/7172)

(/tasks/7173)

(/tasks/7174)

0%

Générer le CRUD | Générer le bundle Coav

Pour le moment Symfony a créé un bundle dans le dossier src pour qu'une page s'affiche quand tu lances l'appli pour la première fois. Il s'appelle AppBundle et il ne nous servira à rien. Tu peux donc simplement le supprimer afin qu'il ne crée pas de confusion chez les développeurs qui reprendraient ton code.

Il est très important de ne pas garder de "code mort" dans tes projets. Le code mort est du code qui n'est jamais exécuté par l'application et c'est toujours très frustrant de faire l'effort de comprendre un bout de code puis de s'apercevoir qu'il ne sert à rien et que le développeur précédent a juste oublié de l'enlever.

Un fois le dossier supprimé pense à supprimer la ligne qui le concerne dans app/AppKernel.php. Ce fichier indique à Symfony tous les bundles qu'il doit charger pour fonctionner. On y trouve à la fois les bundles vendors (qui sont dans vendor) et les bundles propres au projet (qui sont dans src).

Tu peux maintenant créer ton bundle en tapant la commande suivante dans un terminal. Pense à sélectionner le dossier flyaround comme dossier courant (`cd /var/www/html/flyaround`).

```
php app/console generate:bundle
```

Là, Symfony te pose plusieurs questions qui vont lui permettre de générer le bundle qu'il te faut. Réponds-y comme moi puis nous commenterons ces réponses.

élèves connectés

```
(https://github.com/WildCodeSchool/feedback/issues/new?
romain@romain-ThinkPad-T420s: /var/www/html/flyaround$ php app/console generate:bundle
PHP Warning:  Module 'xdebug' already loaded in Unknown on line 0

Welcome to the Symfony bundle generator!

Are you planning on sharing this bundle across multiple applications? [no]:

Your application code must be written in bundles. This command helps
you generate them easily.

Give your bundle a descriptive name, like BlogBundle.
Bundle name: WCS/CoavBundle

In your code, a bundle is often referenced by its name. It can be the
concatenation of all namespace parts but it's really up to you to come
up with a unique name (a good practice is to start with the vendor name).
Based on the namespace, we suggest WSCoavBundle.

Bundle name [WSCoavBundle]:

Bundles are usually generated into the src/ directory. Unless you're
doing something custom, hit enter to keep this default!

Target Directory [src/]:

What format do you want to use for your generated configuration?

Configuration format (annotation, yml, xml, php) [annotation]: yml

Bundle generation

> Generating a sample bundle skeleton into src/WCS/CoavBundle OK!
> Checking that the bundle is autoloading: OK
> Enabling the bundle inside app/AppKernel.php: OK
> Importing the bundle's routes from the app/config/routing.yml file: OK
> Importing the bundle's services.yml from the app/config/config.yml file: OK

Everything is OK! Now get to work :).

romain@romain-ThinkPad-T420s: /var/www/html/flyaround$ 
[0] 0:bash* "romain-ThinkPad-T420s" 15:30 21-Apr-16
```

Nom du bundle

La norme de Symfony impose que le nom d'un bundle commence par une majuscule et finisse par Bundle. La bonne pratique en vigueur dans de nombreuses entreprises (dont la Wild Code School) veut que les bundles soient stockés dans le sous-dossier src/WCS et que leur nom soit écrit en CamelCase (Vois la ressource 1 pour plus d'info).

Le format de la configuration

La configuration du bundle peut être écrite de plusieurs façons. Le YAML (abrégé yml) permet d'avoir une configuration isolée dans un fichier clair et lisible. De plus nous réutiliserons le YAML pour d'autres configurations plus tard dans le projet.

Génération du bundle

Ici, Symfony ne nous demande rien mais nous informe de ce qu'il fait. Va voir dans src/WCS/CoavBundle pour admirer tout son travail. Tu remarqueras notamment la présence d'un contrôleur et d'une vue (src/WCS/CoavBundle/Resources/views/Default/index.html.twig) mais pas de modèle pour l'instant. Le bundle est activé automatiquement dans AppKernel.php et le chargement des routes est aussi automatique. Tu le verras dans le fichier app/config/config.yml. Tu peux d'ailleurs supprimer les routes concernant le bundle AppBundle pendant qu'on y est.

Bien sûr, tout ça peut aussi être fait à la main. Le seul fichier indispensable dans notre bundle est WCSCoavBundle.php. Il contient une classe WCSCoavBundle qui elle-même étend la classe Bundle. La norme de Symfony impose que chaque fichier contienne une seule classe qui a le même nom que le fichier.

1. Wikipédia CamelCase (<https://fr.wikipedia.org/wiki/CamelCase>)

Le CamelCase est une convention de nommage couramment utilisée en programmation

2. Création d'un bundle avec Openclassrooms

(<https://openclassrooms.com/courses/developpez-votre-site-web-avec-le-framework-symfony2/utilisons-la-console-pour-creer-un-bundle>)

Le tutoriel d'openclassrooms donne plus de détails

[Retour à la quête \(/quests/49\)](/quests/49)

[Valider + Suivante](#)

>

[\(/tasks/7172/validate?redirect=%2Ftasks%2F71](/tasks/7172/validate?redirect=%2Ftasks%2F71)