7J’ai récemment réalisé une formation en ligne sur le Framework Symfony, autour des langages PHP et Twig. Excellente et très complète créée par Lior Chamla.

Formation de grande qualité que je recommanderai à qui souhaite découvrir Symfony.

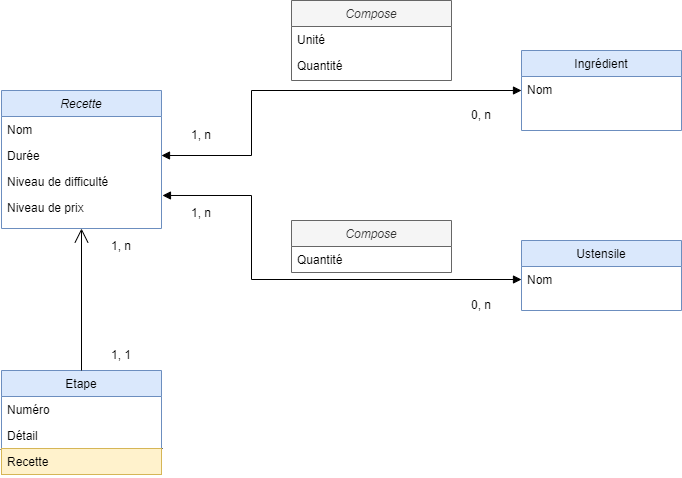
[Symfony 5 : Le guide complet (web-develop.me)](https://learn.web-develop.me/view/courses/symfony-5-le-guide-complet-debutants-et-intermediaires)

Afin de consolider et d’ancrer les compétences acquises par cette formation, je souhaite réaliser un projet et un document de synthèse permettant de faire le tour de toutes les pratiques et tous les enseignements de la formation.

L’idée est également de donner les éléments de base et les étapes à la construction d’un projet, les outils et les dépendances nécessaires à chaque étape.

Je souhaite créer un projet de gestion de recettes de cuisine avec une architecture permettant de gérer les recettes de cuisines, les ingrédients et ustensiles nécessaires à la réalisation de la recette, ainsi que les différentes étapes de fabrication.

Le projet devrait permettre de consulter une liste de recette, créer ou modifier une recette, administrer la liste des ingrédients et ustensiles nécessaire à la réalisation d’une recette.



Ce projet pourrait à terme également évoluer en ajoutant le chargement de photos de la recette, un système de notation de recette ou la possibilité de laisser des commentaires.

**Étape 1 - Les outils nécessaires à la réalisation du projet**

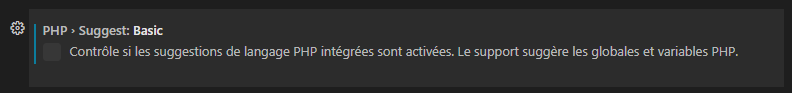
**Éditeur de texte VS Code**

**Extensions à installer :**

* Intelephense
* Namespace Resolver
* Twig Langage2
* MySQL

**Paramétrage de VS Code**

* Désactiver PHP>Suggest:Basic

****

* Activer le formatage lors de la sauvegarde

**Une image contenant texte

Description générée automatiquement**

* Définir le langage Twig

**Serveur Apache**

**PHP** (version minimale PHP7)

Connaitre sa version en ligne de commande : php -v

**Base de données MySQL**

**Composer** (outil de gestionnaires des dépendances)

Connaitre sa version en ligne de commande : composer -V

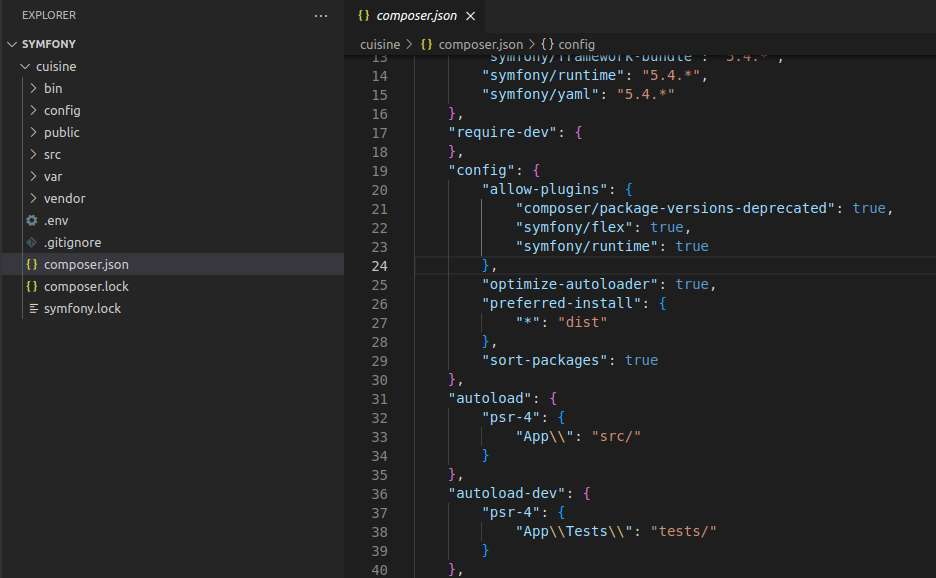
**Git** (outil de versionnage du code source)

Connaitre sa version en ligne de commande : git --version

**Étape 2 - Initialisation du projet**

**Initialisation du projet**





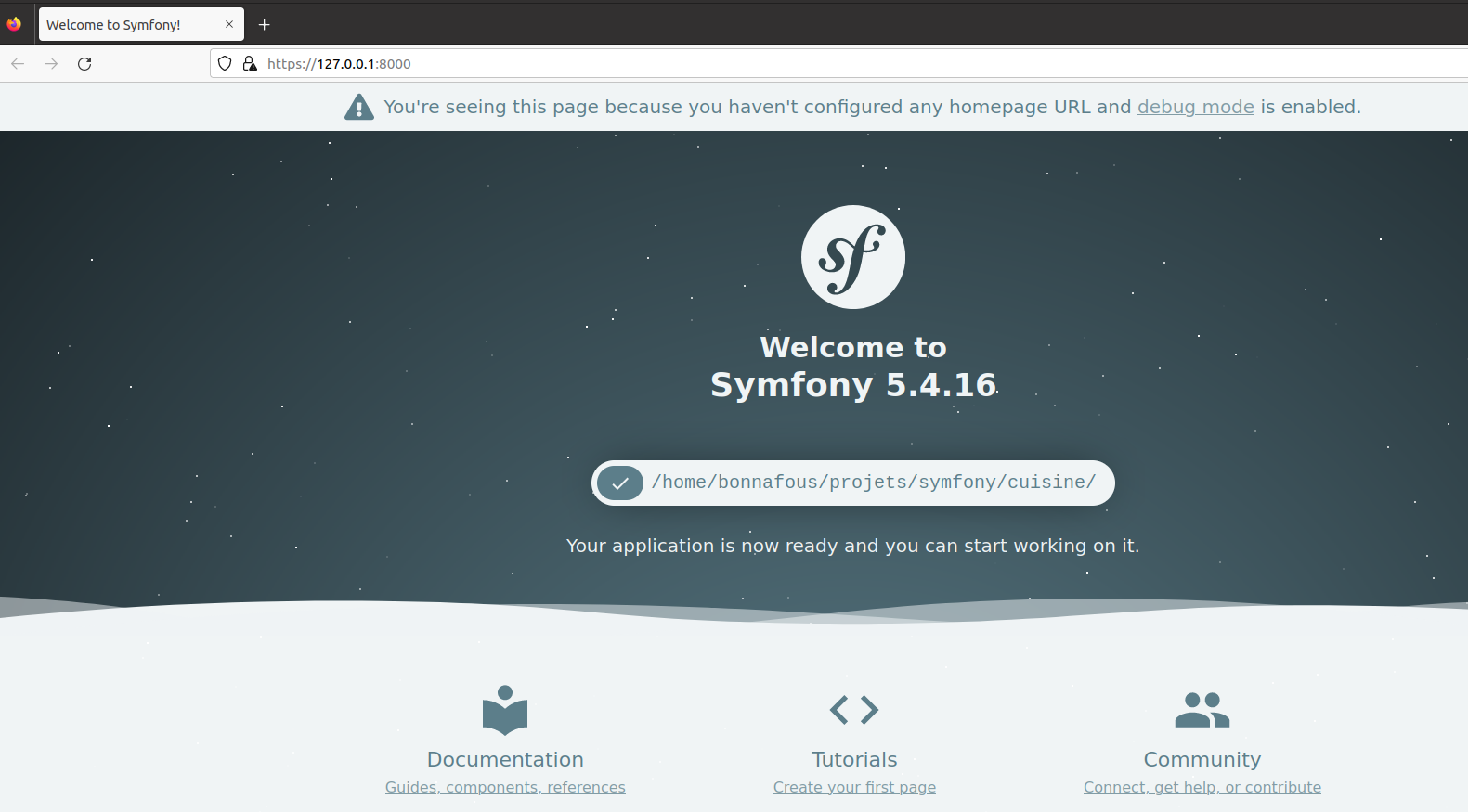
**Création d’un dépôt Git**



**Lancer le serveur**

**Une image contenant texte

Description générée automatiquement**

****

**Étape 3 - Les routes**

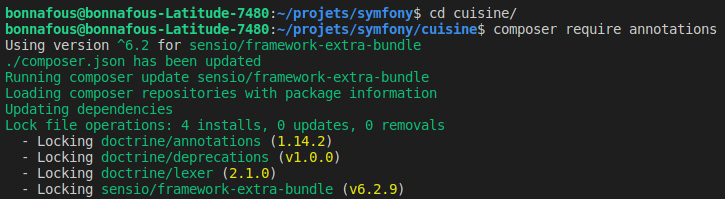
**Utilisation de Symfony Flex pour charger les services d’annotations**

[Using Symfony Flex to Manage Symfony Applications (Symfony 3.4 Docs)](https://symfony.com/doc/3.4/setup/flex.html)

Symfony Flex est une fonction permettant d’installer et gérer les applications Symfony. Il permet d’automatiser les tâches les plus communes et notamment la commande de chargement des services en récupérant les services correspondant à partir de mots clés (mailer, annotations, …) sans avoir à connaître le véritable nom des services.

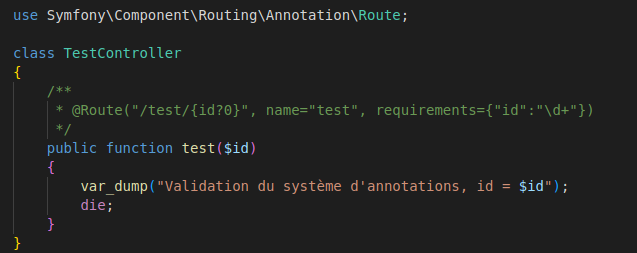
Par exemple lorsque je demande à composer de charger les services pour la gestion des annotations, Symfony Flex va installer automatiquement les bons services par défaut (doctrine/annotations, doctrine/deprecations, doctrine/lexer et sensio/framework-extra-bundle).

**Attention** à bien être positionné dans le répertoire du projet (sinon la commande de chargement ne fonctionne pas)

****

**Test du système d’annotations**

Nécessite l’utilisation de la bibliothèque de Route du service d’annotations de Symfony

****

**A compléter avec quelques routes du projet …**

**Étape 4 - Les objets de la base de données**

**Création de la base de données**

**Installation du système de gestion des objets de base de données Symfony**

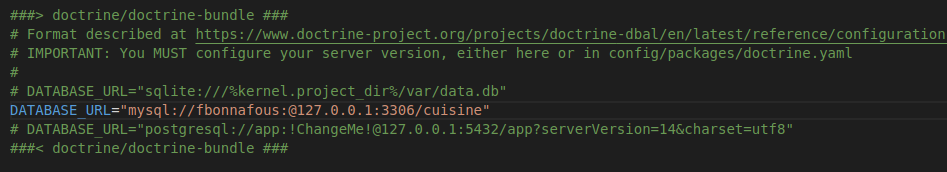
**Une image contenant texte

Description générée automatiquement**

**Une image contenant texte

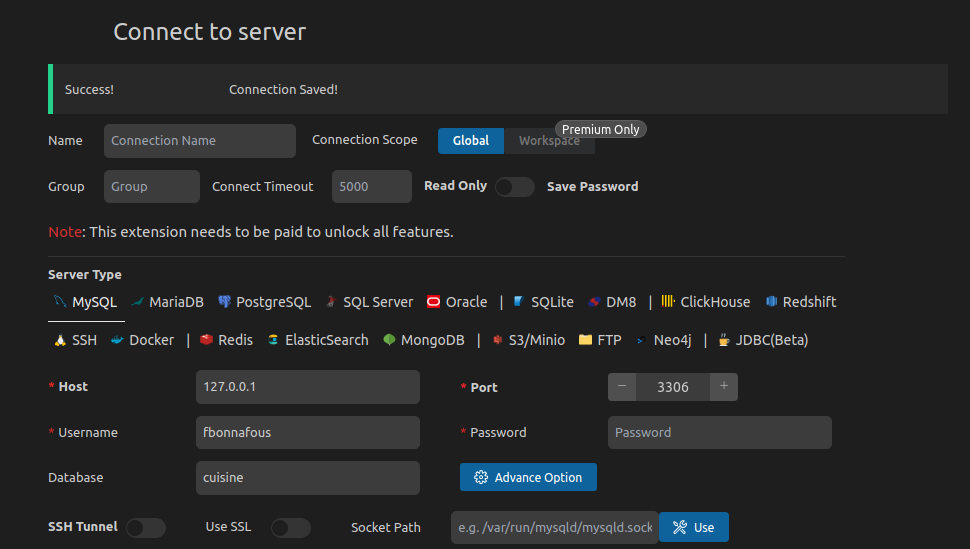
Description générée automatiquement**

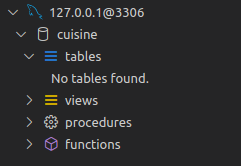
Symfony propose de mettre à jour le fichier d’environnement dans lequel je vais configurer ma base de données. Dans mon cas j’utilise une base de données MySQL pour l’environnement de développement.

****

**Création de la base de données pour le projet**

**Connexion à la base de données via l’extension MySQL de VS Code**

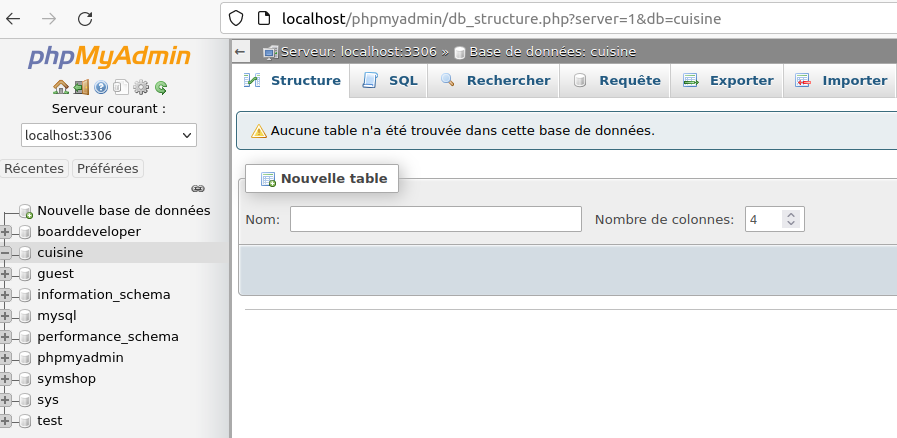
****

****

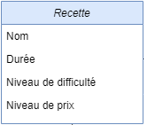
**Accès à la base de données via PHPMyAdmin**

PHPMyAdmin propose des options plus avancées que l’extension MySQL de VS Code. Il peut être intéressant d’y avoir accès.

Personnellement j’utilise l’extension MySQL de VS Code pour la visualisation de données (SELECT) et leurs manipulations de base (INSERT, UPDATE et DELETE). Et PHPMyAdmin pour des fonctionnalités plus avancées, telles que l’administration, la gestion d’import ou d’export.

****

**Création d’un objet simple**

****

**Installation des composant permettant la gestion des objets (maker)**

Maker va me permettre de créer mes entités en ligne de commande

**Une image contenant texte

Description générée automatiquement**

**Liste des commandes disponibles en lignes de commandes via maker**

**Une image contenant texte

Description générée automatiquement**

**Création de l’entité recette et de ses attributs**

**Une image contenant texte

Description générée automatiquementUne image contenant texte

Description générée automatiquement**

**Symfony crée une entité Recette.php ainsi qu’un Repository**

**Une image contenant texte

Description générée automatiquement**

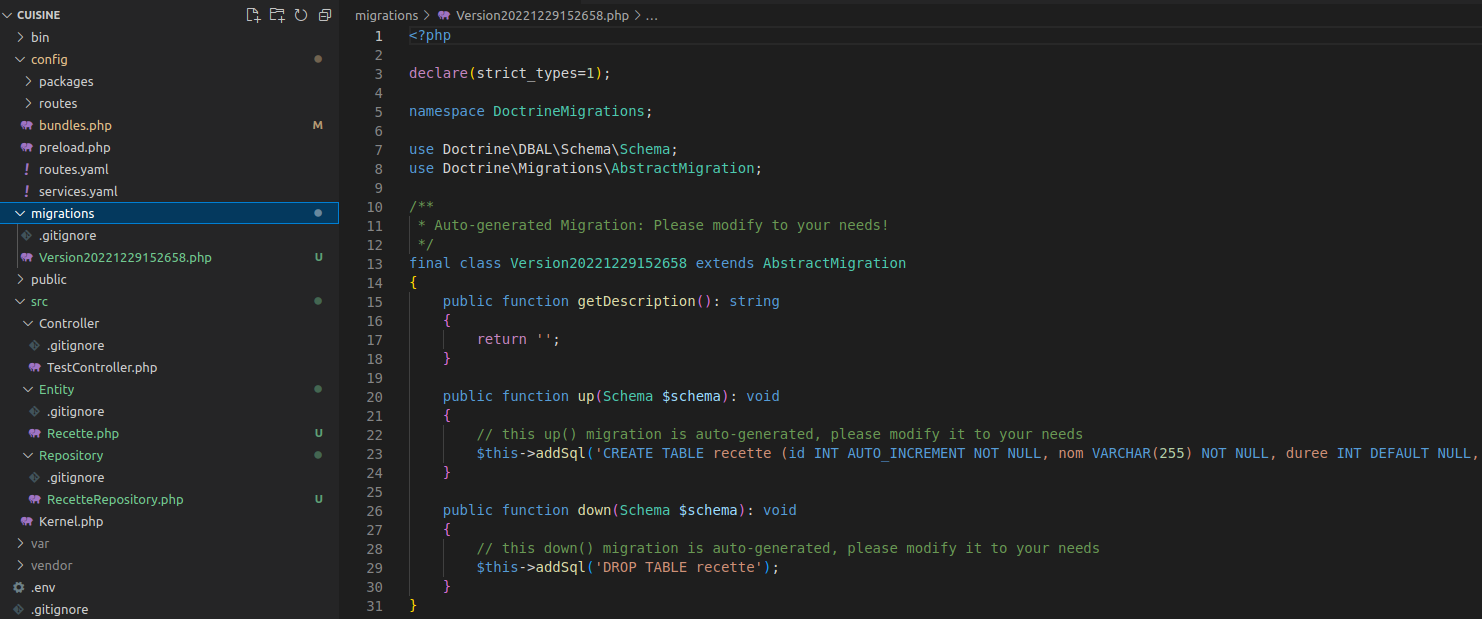
**Une image contenant texte, capture d’écran, moniteur, équipement électronique

Description générée automatiquement**

Il faut ensuite réaliser une migration. La migration va générer un script contenant les instructions SQL nécessaires à la création de l’objet dans la base de données.

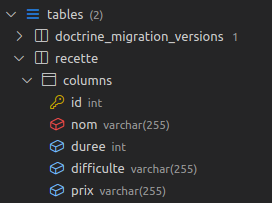
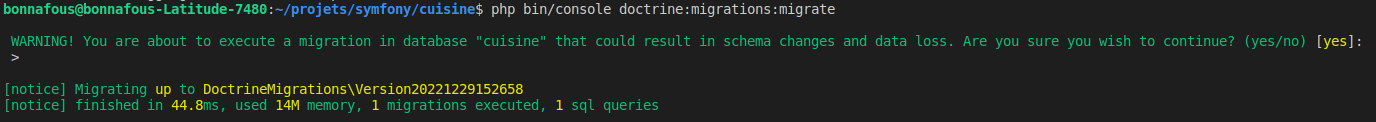
**Une image contenant texte

Description générée automatiquement**

****

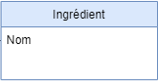
Une fois le script de migration en base, il reste à l’exécuter.

L’exécution du script de migration va créer l’entité dans la base de données.

****

Il reste à créer les objets ingrédients et ustensiles qui vont stocker les éléments nécessaires à nos recettes de cuisine.

Créer les entités, créer les scripts de migration et exécuter les scripts de migration pour mettre à jour notre base de données.



Une image contenant texte

Description générée automatiquement

**Création d’un objet avec un relation de type 1-n**

**Création d’un objet avec un relation de type n-n**

**Étape 5 - Alimenter la base de données (les fixtures)**

Les fixtures sont des jeux de données. Symfony permet d’alimenter la base de données via l’utilisation de jeux de données.

L’installation du gestionnaire de fixtures va créer un répertoire de gestion de fixture dans lequel il va être possible de développer un script (AppFixtures.php) permettant de mettre à jour la base de données.

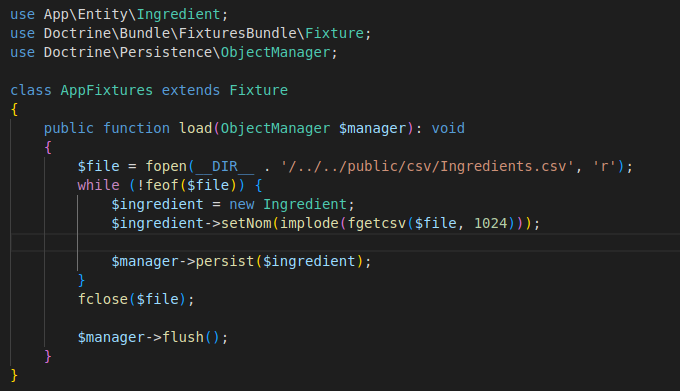
**Installation du système de gestion des fixtures**

**Une image contenant texte

Description générée automatiquement**

****

Afin d’alimenter la base de données j’ai créé deux jeux de données au format CSV qui contiennent une liste d’ingrédients et une liste d’ustensiles que je vais vouloir importer dans la base de données.

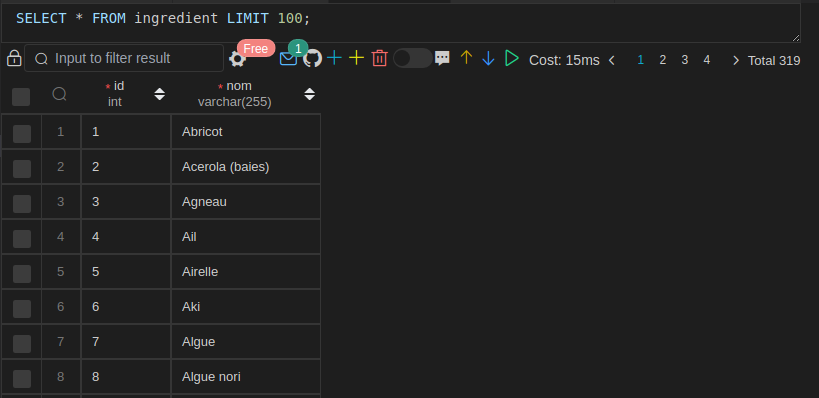
****

Une fois le script crééil suffit de l’exécuter en ligne de commande.

**Une image contenant texte

Description générée automatiquement**

L’exécution du script permet d’alimenter les tables de la base de données.

****

**Étape 6 - Le langage d’affichage (Twig)**

Twig est un moteur de template pour PHP utilisé avec Symfony.

**Installation du moteur de template Twig**

Une image contenant texte

Description générée automatiquement

L’installation met à jour les fichiers de configuration et crée les répertoires dédiés aux templates.



Création de la page d’accueil

Une image contenant texte, équipement électronique, capture d’écran

Description générée automatiquement

**Création du contrôleur de la page d’accueil**

La classe HomeController hérite d’une classe AbstractController qui permet d’hériter de fonctionnalités avancés plus facile à écrire.

Dans cet exemple l’ AbstractController permet d’utiliser une fonction render qui redirige sur la page home.html.twig.

[symfony/AbstractController.php at 6.2 · symfony/symfony · GitHub](https://github.com/symfony/symfony/blob/6.2/src/Symfony/Bundle/FrameworkBundle/Controller/AbstractController.php)

****

**Création du template**

