

# Symfony 4

## Introduction

---

AYMEN SELLAOUTI

## C'est quoi symfony ?

---

*« Symfony is a set of PHP Components, a Web Application framework, a Philosophy, and a Community — all working together in harmony. » [site Officiel Symfony]*

- ✓ Ensemble de composants PHP
  - ✓ Framework pour les applications web
    - ✓ Basée sur des composants
    - ✓ Structuration du code
    - ✓ Maintenabilité
  - ✓ Une philosophie
    - ✓ Les bonnes pratiques
    - ✓ standardisation
  - ✓ Une très grande communauté
- FrameWork : Cadre De Travail (Boite à outils)
- Ensemble de composants servant à créer :
- Fondation
  - Architecture

# Pourquoi utiliser un Framework

---

**Productivité** : ensemble de composants déjà prêt à l'emploi

**Propreté du code** => maintenabilité et évolutivité

**Basée sur une architecture** => facilite le travail en équipe

**Communauté et documentation**

# Installation de Symfony (1)

---

Afin d'installer Symfony, vous devez disposer de Composer qui est un PHP Package Manager.

<https://getcomposer.org/download/>

**Pré requis :**

PHP 7.1.3 ou plus (7.2 de préférence)

Lancer la commande : `composer create-project symfony/website-skeleton nomProjet` pour une version d'un projet web qui contient les bibliothèques de bases dédiées à un projet web.

Lancer la commande : `composer create-project symfony/skeleton nomProjet` pour une version orientée vers les microservices ou les API.

## Installation de Symfony (2)

---

Vous pouvez aussi utiliser le symfony client : <https://symfony.com/download>

Un binaire symfony sera alors installé et défini dans les variables d'environnements.

Vous n'avez qu'à utiliser donc la commande `symfony new nomProjet` pour avoir votre projet

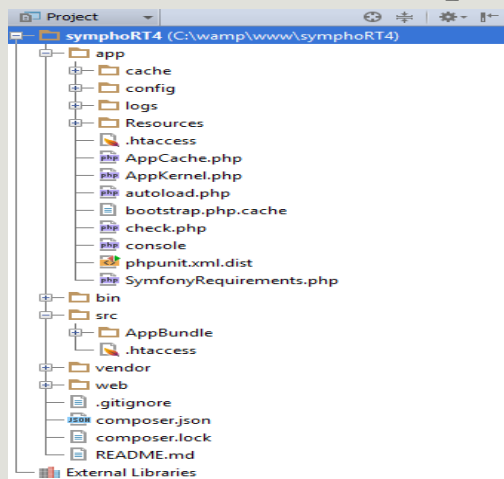
## Symfony Roadmap

---

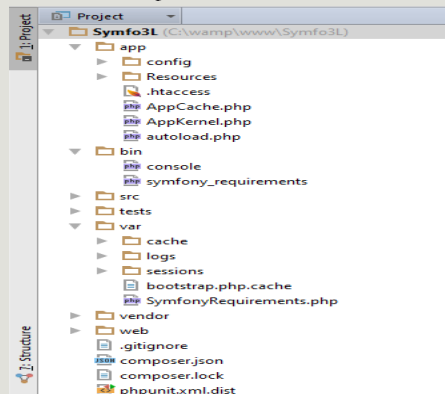
Vous pouvez vérifier les différentes roadmap des versions symfony. Choisissez les versions stables LTS (Long Term Support)

<https://symfony.com/roadmap>

## Architecture d'un projet Symfony 2.8 et 3.\*



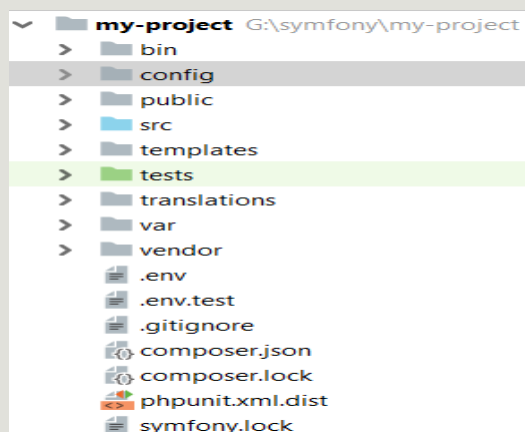
- **app/** : La configuration de l'application,
- **src/** : Le code PHP du projet,
- **vendor/** : Les bibliothèques tierces,
- **web/** : Le répertoire Web racine.



Symfony 3

Identifier les différences entre symfony2.8 et 3

## Architecture d'un projet Symfony 4.\*



# Le contrôleur frontal (1)

---

public/[index.php](#)

Il joue le rôle de dispatcheur :

- Intercepte les requêtes
- Appelle le noyau de symfony (AppKernel.php)
- Le noyau prépare la réponse à rendre

[https://symfony.com/doc/4.2/create\\_framework/front\\_controller.html](https://symfony.com/doc/4.2/create_framework/front_controller.html)

# Le contrôleur frontal (2)

---

Le contrôleur principal s'occupe de gérer les requêtes, mais cela signifie plus que simplement déterminer une action à exécuter. En fait, il exécute le code qui est commun à chaque action, soit les étapes suivantes :

- Définir les constantes.
- Déterminer les chemins des bibliothèques Symfony.
- Charger et initialiser les classes du cœur du framework.
- Charger la configuration.
- Décoder la requête URL afin d'identifier les actions à effectuer et les paramètres de la requête.
- Si l'action n'existe pas, faire une redirection sur l'action 404 error.
- Activer les filtres (par exemple, si les requêtes nécessitent une authentification).
- Exécuter les filtres une première fois.
- Exécuter l'action et générer la présentation.
- Exécuter les filtres une seconde fois.
- Renvoyer la réponse.

# Architecture MVC

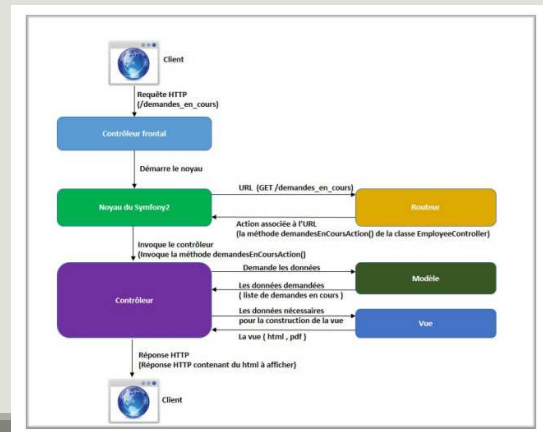
Symfony n'est pas une architecture MVC (Model View Controller).

SensioLab définit Symfony comme un HTTP

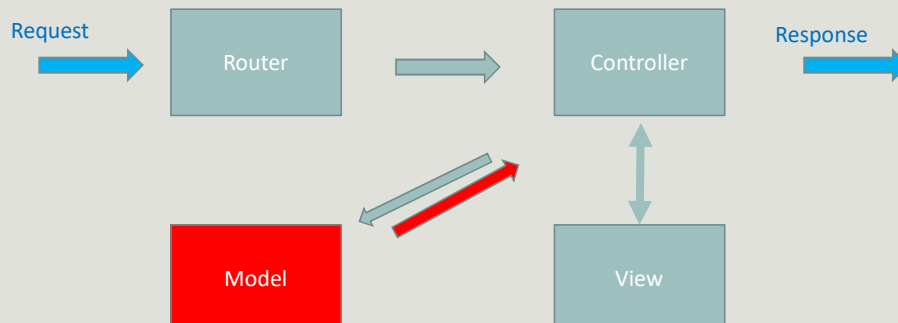
Framework basé sur l'échange HTTP Request/Response.

Cependant on peut l'implémenter.

La raison pour laquelle Symfony n'est pas défini comme telle est que la couche Model n'est pas fournie



# Architecture MVC



Cette partie est indépendante de Symfony vous pouvez utiliser ce que vous voulez Doctrine par exemple

# Symfony un Framework HTTP (Request/Response)

## La classe Request (HttpFoundation)

---

Le composant `HttpFoundationRequest` permet d'encapsuler les propriétés d'une requête http.

Dans Symfony, la création de cet objet se fait au niveau du contrôleur frontal.

```
$request = Request::createFromGlobals();
```

Ceci revient à appeler le constructeur de la classe et de lui passer les variables globales nécessaires et qui concerne la requête.

```
$request = new Request (
    $_GET,
    $_POST,
    [],
    $_COOKIE,
    $_FILES,
    $_SERVER
);
```

# Symfony un Framework HTTP (Request/Response)

## La classe Request (HttpFoundation)

---

L'objet request contient des informations concernant la requête du client. Ces informations sont récupérables via quelques propriétés public :

- `request`: équivaut à `$_POST`;
- `query`: équivaut à `$_GET ($request->query->get('name'))`;
- `cookies`: équivaut à `$_COOKIE`;
- `files`: équivaut à `$_FILES`;
- `server`: équivaut à `$_SERVER`;
- `headers`: équivaut à une partie de `$_SERVER ($request->headers->get('User-Agent'))`.

[https://symfony.com/doc/master/components/http\\_foundation.htm](https://symfony.com/doc/master/components/http_foundation.htm)

# Symfony un Framework HTTP (Request/Response)

## La classe Response (HttpFoundation)

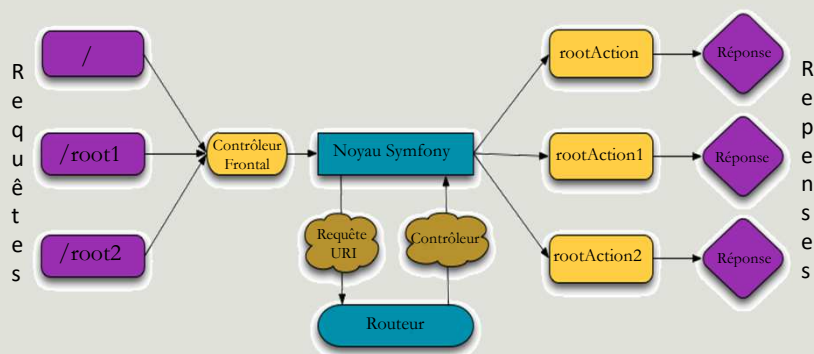
Le composant HttpFoundationResponse permet d'encapsuler les propriétés de la réponse à envoyer à l'utilisateur.

Englobe particulièrement :

- Le contenu de la réponse
- Le code de la réponse
- Un tableau Headers HTTP

[https://symfony.com/doc/master/components/http\\_foundation.html#response](https://symfony.com/doc/master/components/http_foundation.html#response)

## Traitement d'une requête au sein de Symfony





## Récapitulatif du flux Requête/Réponse

---

1. Un client envoie une requête HTTP.
2. Chaque requête va exécuter le Front Controller
3. Le front Controller lance Symfony et lui passe les informations de la requête afin qu'elle soit traitée.
4. Symfony utilise les routes et les contrôleurs afin de créer une réponse
5. Symfony retourne une réponse à l'utilisateur.

[https://symfony.com/doc/4.2/introduction/http\\_fundamentals.html](https://symfony.com/doc/4.2/introduction/http_fundamentals.html)

---

aymen.sellaouti@gmail.com