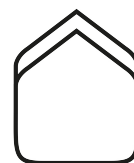


# Formation Symfony Avancée (3 jours)



Accéder à la formation : <https://jolicampus.com/formations/symfony-avancee>

Société JoliCode, 18 Avenue Parmentier 75011 Paris / hello@jolicampus.com / +33 1 43 57 39 11



JoliCampus

## Objectif

Symfony est un outil complexe qui propose un haut niveau d'abstraction et de nombreuses fonctionnalités. Bien que les fonctionnalités de base de Symfony permettent de répondre à la plupart des besoins d'un projet web, il serait dommage de se limiter à celles-ci, et ne pas tirer toute la puissance possible du framework. Avec cette formation "avancée", basée sur la dernière version à ce jour, le participant sera capable d'aller plus loin dans sa compréhension du framework et d'en utiliser les fonctionnalités les plus avancées.

Il sera à même d'étendre le framework pour l'adapter à ses besoins, de déporter des traitements lourds, de mettre en place des tests automatisés complets et robustes. En pleine compréhension du framework, l'apprenant sera capable d'implémenter les exemples sur sa propre machine de développement.

En résumé, cette formation vous permettra :

- ▶ de maîtriser les principaux composants Symfony
- ▶ d'étendre Symfony et Twig pour l'adapter à des besoins spécifiques
- ▶ de créer des formulaires complexes et dynamiques
- ▶ d'assurer la qualité et la non régression en mettant en place des tests
- ▶ de faire des traitements asynchrones
- ▶ s'appuyer sur les composants avancés du framework pour répondre aux besoins métiers

## Pré-requis

- ▶ Bonnes connaissances du framework Symfony
- ▶ Docker
- ▶ Docker compose
- ▶ Ordinateur

## Plan

### *Rappel : les bonnes pratiques*

- ▶ Retour rapide sur les classiques
- ▶ Cycle de release
- ▶ Symfony Flex
- ▶ Structuration des projets Symfony : dossiers, fichiers, convention
- ▶ Documentation
- ▶ Découverte du livre sur les bonnes pratiques Symfony
- ▶ Clean code : lisibilité et maintenabilité

### *Découverte de l'application*

Nous utiliserons une application déjà existante comme fil rouge lors de la formation !

## Durée de la formation :

3 jours

## Tarif :

1800 € HT / personne

## Public

- ▶ Développeurs PHP
- ▶ Chefs de projets techniques
- ▶ Architectes logiciel

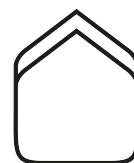
## Pré-requis

- ▶ Bonnes connaissances du framework Symfony
- ▶ Docker
- ▶ Docker compose
- ▶ Ordinateur

[Demander un devis](#)

## ***Bien débogger son application***

- Utilisation des outils natifs de débogage Symfony (Profiler, DebugBar)
- Analyse des traces d'erreurs et logs



JoliCampus

## ***Le composant console, Filesystem***

- Création et utilisation de commandes personnalisées Symfony
- Mise en place de SymfonyStyle ainsi que différent Helper (ProgressBar, Question, Table)
- Interaction avec d'autres services depuis la console
- Découverte du composant Filesystem

## ***Doctrine***

- ***Doctrine Param Converter*** : Utilisation et personnalisation du ParamConverter dans les routes
- ***Types de fetch*** : Comprendre les différences entre fetch, lazy, et extra\_lazy
- ***Types d'hydratation*** : Object, Array, Scalar, SimpleObject
- ***Iterator*** : Optimisation des requêtes en traitant les grands ensembles de données
- ***Cache*** : Utilisation du cache Doctrine pour améliorer les performances des requêtes

## ***MakerBundle***

- Utilisation du MakerBundle pour générer rapidement des éléments standards

## ***Event dispatcher***

- Principe du pattern Mediator avec Symfony
- Créer, écouter et gérer les événements du framework

## ***Sécurité***

- ***Authentication*** : Configuration avancée de systèmes d'authentification
- ***Autorisation*** : Gestion fine des rôles et des permissions
- ***Authentication avec Authenticator*** : Mise en place d'un authenticator personnalisé

## ***Formulaire***

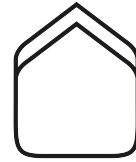
- ***Data Transformer*** : Transformer des données entre la vue et le model
- ***Type Extension*** : Étendre les types de formulaires existants
- ***Form Theme*** : Personnalisation avancée des formulaires via les thèmes Twig
- ***Listener & Extension*** : Ajouter des comportements personnalisés au cycle de vie d'un formulaire

## ***Validation***

- ***Les contraintes natives*** : Comprendre et utiliser les validations fournies par Symfony
- ***Les contraintes inlines*** : Écrire des règles de validation personnalisées
- ***Les contraintes externes*** : Utiliser des validations externes via des services
- ***Les groupes de validation*** : Valider des champs selon des scénarios spécifiques
- ***Les séquences*** : Gestion avancée de la validation séquentielle

## Twig

- Étendre Twig: Créer des filtres, des fonctions et des extensions Twig personnalisées



JoliCampus

## Injection de dépendance

- **Rappels** : Bonnes pratiques d'injection de dépendance
- **Définition de services / Autowire** : Utiliser l'autowiring pour simplifier les définitions de services
- **Services taggués** : Comprendre et utiliser les services taggués dans les extensions de bundle
- **Compiler Pass** : Personnalisation de la compilation du conteneur de services
- **ServiceSubscriber** : Optimisation des dépendances avec le Service Subscriber

## Cache

- **Cache HTTP** : Mise en cache des réponses HTTP et gestion des en-têtes
- **Cache applicatif** : Utiliser des caches au niveau de l'application (ex. Redis, APCu)

## Sérialisation

- **Architecture** : Introduction à la sérialisation / désérialisation de données avec Symfony
- **Utilisation** : Transformer des objets en JSON, XML, YAML et vice-versa
- **Étendre le serializer** : Ajouter des normalizers/dénormalizers personnalisés

## Tests

- **Définition & Principe** : Différents types de tests et leur importance
- **CI/CD** : Intégration des tests dans les pipelines de CI/CD
- **Test Unitaire** : Bonnes pratiques pour tester des services isolés
- **Test Fonctionnel** : Tester les contrôleurs et les intégrations avec les dépendances

## Messenger

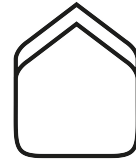
- **L'asynchrone** : Introduction à la gestion des tâches asynchrones avec Messenger
- **Messenger** : Mise en place de bus de messages, gestion des queues, et optimisation des tâches de fond

## Workflow

- **Pourquoi un workflow ?** : Présentation des cas d'usage des workflows dans une application
- **Qu'est-ce qu'un workflow ?** : Concepts de base, état, transition, événements
- **Le Workflow** : Mise en œuvre pratique et configuration d'un workflow dans Symfony
- **Les événements** : Réagir aux événements des workflows pour des actions spécifiques
- **Intégration avec Twig** : Visualisation des workflows dans l'interface utilisateur
- **Les Metadata** : Ajout de métadonnées aux transitions et états
- **Les réseaux de Pétri** : Introduction avancée à la modélisation avec des réseaux de Pétri

## Erreurs & debug avancé

- ▶ *Personnalisation des pages d'erreur* : Adapter les pages d'erreur en fonction des statuts HTTP
- ▶ *Gestion des erreurs avancée* : Gestion des exceptions personnalisées
- ▶ *Logs* : Mise en place d'une stratégie de logging efficace pour la surveillance en production
- ▶ *Composant VarDumper* : Déboguer efficacement avec des dumps intelligents



JoliCampus

## Note

Nous pouvons adapter la formation à vos besoins, en mettant l'accent sur un aspect plus particulier du développement. Avant chaque formation, nos formateurs contactent les participants afin de déterminer vos objectifs et savoir s'il faut mettre l'accent sur un sujet particulier. C'est le bon moment, pour vous, d'échanger avec le formateur pour lui permettre de mieux cerner vos objectifs et les aspects que vous aimeriez voir approfondis.