

# PHP Symfony Initiation + Approfondissement

## INFORMATIONS GÉNÉRALES

**Type de formation :** Formation continue

**Éligible au CPF :** Non

**Domaine :** Développement

**Action collective :** Non

**Filière :** Open Source - LAMP : Linux Apache PHP

**Rubrique :** PHP

**Code de formation :** LEDN208

## € Tarifs

**Prix public :** 2495 €

### Tarif & financement :

Nous vous accompagnons pour trouver la meilleure solution de financement parmi les suivantes :

- **Le plan de développement des compétences** de votre entreprise : rapprochez-vous de votre service RH.
- **Le dispositif FNE-Formation.**
- **L'OPCO** (opérateurs de compétences) de votre entreprise.
- **Pôle Emploi** sous réserve de l'acceptation de votre dossier par votre conseiller Pôle Emploi.
- **CPF -MonCompteFormation**

Contactez nous pour plus d'information

## PRÉSENTATION

### Objectifs & compétences

A l'issue de la formation, le stagiaire sera capable :

- Maîtriser les fondamentaux pour réaliser un projet Symfony

### Public visé

Tout

### Pré-requis

Avoir suivi le cours " PHP Avancé" ou disposer des connaissances équivalentes, en particulier la POO

## 📍 Lieux & Horaires

**Durée :** 35 heures

**Délai d'accès :**  
**8 jours avant le début de la formation**

## PROGRAMME

### Introduction

Framework PHP : définition et revue du marché

Principes d'architecture : centrage HTTP, orientation objet, orientation service

Anatomie du framework : le kernel, les composants et les bundles

### Installer et configurer un projet

Configurer l'environnement, gérer les pré-requis, l'exécutable symfony

Démarrer le projet avec symfony ou composer, choisir son squelette de départ

Ajouter des paquets avec Symfony Flex pour composer

Les répertoires du projet

Configurer le projet via les fichiers de config

Paramétrer son environnement via DotEnv

Choisir et utiliser son serveur de développement

**Atelier:** Installer, configurer et lancer un projet Symfony

### Créer des pages : les contrôleurs

Reconnaître les URLs demandées via les routes

Produire la réponse HTTP par les classes de contrôleur

Utiliser les annotations de routes

Affiner la gestion des routes : paramètres, valeurs par défaut, contraintes, méthodes

Convertir automatiquement les paramètres

Maîtriser le dialogue HTTP via les objets du composant HttpFoundation

Débugger les routes

Profiler les pages

**Atelier:** Créer ses premières pages

### Templating

Principe d'un moteur de template et présentation de Twig

Configurer Twig : chemin, échappement automatique, variables globales, ...

Revue de la syntaxe : les tags et les interpolations

Référencer les pages, créer des liens hypertexte

## 📅 Prochaines sessions

Consultez-nous pour les prochaines sessions.

Référencer des ressources JS, CSS et images  
Factoriser les templates : l'héritage, l'inclusion et la sous-requête

### **Améliorer le rendu des pages, mettre en place un système de navigation**

#### **Base de données (Doctrine)**

Comprendre le rôle d'un ORM  
Installation et configuration de Doctrine  
Créer une classe d'entité  
Utiliser les migrations : créer le schéma  
Enregistrer un objet en base  
Charger des objets depuis la base  
Charger automatiquement depuis la route (ParamConverter)  
Mis en place de relations/associations

**Atelier :** lier le contenu des pages à la base de données

#### **Les formulaires**

Construire le formulaire et l'association à une classe de données  
Rendre le formulaire en HTML  
Réceptionner et valider les données  
Créer ses propres validateurs  
Les classes de formulaire  
Personnaliser le rendu (theming)

### **Créer les formulaires de saisie**

#### **La sécurité**

Principe de fonctionnement : authentification et autorisation  
Créer sa classe d'utilisateur, le fournisseur et gérer l'encodage des mot-de-passes  
Authentifier l'utilisateur sur un contexte de pare-feu  
Contrôler l'accès sur des rôles au niveau du pare-feu, des routes ou dans le code  
Créer des règles personnalisées avec des voters

### **Authentifier les utilisateurs et restreindre l'accès aux pages**

#### **Internationalisation**

Configurer la langue, utiliser des catalogues  
Rédiger les catalogues de messages  
Traduire les validations, les entités, les urls, ...  
Gérer la locale utilisateur

### **Mettre en place des traductions**

Les services  
Utilisation des services du ServiceContainer  
Injection de services et de configuration  
Les services publics et privés  
Renommer les services à l'aide des alias  
Paramétrer les services et gérer les arguments  
Paramétrer le câblage auto et la configuration auto  
Lier des arguments par leur nom ou leur type  
**Atelier :** Déporter les traitements dans des services

### **Gestion de performances**

Revoir les fondamentaux du cache HTTP  
Comparer les deux modèles de gestion de cache Expiration et Validation  
Mettre en place la stratégie de cache sur les routes et dans les contrôleurs  
Exploiter des fragments de pages mis en cache via les Edge Side Includes  
**Atelier :** Réduire les temps de rendus des pages en s'appuyant sur le cache HTTP et les ESI

## **MODALITÉS**

### **Modalités**

**Modalités :** en présentiel, distanciel ou mixte – Horaires de 9H à 12H30 et de 14H à 17H30 soit 7H – Intra et Inter entreprise

**Pédagogie :** essentiellement participative et ludique, centrée sur l'expérience, l'immersion et la mise en pratique. Alternance d'apports théoriques et d'outils pratiques.

**Ressources techniques et pédagogiques :** Support de formation au format PDF ou PPT Ordinateur, vidéoprojecteur, Tableau blanc, Visioconférence : Cisco Webex / Teams / Zoom

**Pendant la formation :** mises en situation, autodiagnostic, travail individuel ou en sous-groupe sur des cas réels

**Méthode**

**Fin de formation** : entretien individuel

**Satisfaction des participants** : questionnaire de satisfaction réalisé en fin de formation

**Assiduité** : certificat de réalisation (validation des acquis)