

## Formation Symfony Avancé

<b>Durée :</b>	5 jours
<b>Public :</b>	Développeur PHP confirmé
<b>Pré-requis :</b>	Avoir suivi la formation Symfony initiation et approfondissement ou connaissance équivalente
<b>Objectifs :</b>	Découvrir les fonctions avancés de Symfony - Gérer les événements et l'automatisation de tâches - Tester et valider son application
<b>Sanction :</b>	Attestation de fin de stage mentionnant le résultat des acquis
<b>Taux de retour à l'emploi:</b>	Aucune donnée disponible
<b>Référence:</b>	PHP724-F
<b>Note de satisfaction des participants:</b>	5,00 / 5

### Introduction

Revue de l'architecture du framework  
Évolution suivant les versions

### Les événements et écouteurs

Découpler d'avantage de code métier via le gestionnaire d'événement  
Créer un écouteur d'événement : EventListener  
Créer un souscripteur d'événement : EventSubscriber  
Événement natifs symfony et événements personnalisés

### Altérer un comportement sans héritage via souscripteur d'événement

### Le composant Cache

Présentation du composant Cache  
Cache contract vs PSR-6  
Accéder et sauvegarder des données en cache  
Supprimer, invalider ou programmer l'expiration des données  
Liste des Adapters disponibles

### Atelier : Mise en cache sous Doctrine

### Le composant Messenger

Comprendre les principes de communications inter-application  
Créer le message et le handler  
Diffuser le message

Les transports disponibles : AMQP, Redis, Doctrine, In Memory, ...  
Configurer les transports et le superviseur

## **Gérer des traitements en parallèle via Messenger**

### **Le composant Mail**

Les composant Mail et Swift\_mailer  
Installation et configuration du transport  
Créer un mail, gérer les adresses  
Gérer le format de contenu : text/html, utiliser twig  
Attacher un fichier, embarquer un image

### **Mettre en place les services d'envoi de mail**

### **Le composant Console Commands**

Définir une commande  
Gestion des entrées / sorties  
Accès aux services  
Tester les commandes  
Sortie avancée : style et couleur  
Sortie avancée : barre de progression, section, tableau  
Entrées : distinguer arguments et options  
Créer des questions : confirmation, information, choix

### **Définir des actions automatisables via des commandes**

### **Formulaire avancé**

Créer un type de champs personnalisé avec son thème  
Gérer les données associées modèle-normalisée-vue  
Définir les DataTransformer associés et y injecter des services  
Définir un DataMapper pour les données composé  
Associer son type de champs personnalisé via un FormGuesser  
Modifier dynamiquement le formulaire via les événements de formulaire

### **Mise en place de champs de formulaires avancés**

### **Intégration de WebPack Encore**

Installation et configuration Yarn / Webpack  
Définir les entrées webpack et leur ressources associées  
Inclure les entrées dans Twig  
Gérer les dépendances dynamiquement avec les modules ES6  
Utiliser Sass, Less ou Stylus  
Utiliser Typescript  
Passer des données de Twig au javascript  
Gestion du cache et versionning

### **Mettre en place une interface riche**

### **Les tests unitaires et fonctionnels**

Utilisation de PHPUnit Bridge  
Tests unitaires des services métier  
Tests fonctionnels et gestion des formulaires  
Gestion des dates : ClockMock

**Mise en place de procédures de tests**

**Mise en Application : Mettre en place une API REST avec authentification**

**Passage de la certification (si prévue dans le financement)**