

## DEVELOPPEMENT WEB AVEC SYMFONY3

### PROGRAMME DE FORMATION BLENDED LEARNING (présentiel + e-learning)

**Objectifs:**

Découvrez l'essentiel du framework Symfony3 avec les meilleurs experts. Vous apprendrez à maîtriser le modèle MVC ainsi que les formulaires, l'authentification ou bien encore les tests unitaires et la gestion du cache HTTP.

**Durée:**

- 28 heures en présentiel
- 1 mois en e-learning.

**Prix inter:** 1 990€ HT / personne**Code:** SF3C4**Public:** Développeur PHP**Prérequis:** Maîtrise de la programmation orientée objet en PHP**Moyens Pédagogiques**

Sensiolabs university a conçu la meilleure expérience de formation Symfony en associant le présentiel et le e-learning. En complément de la formation en présentiel, la mise en place d'un accès à **notre plateforme de révision en ligne** pendant 1 mois, permet d'associer différentes pratiques d'apprentissage, de valider ses compétences, suivre sa progression et de rester en contact avec son formateur sur toute la durée du programme. Les supports de cours seront consultables dans l'espace personnel de chaque participant. Les formateurs SensioLabs sont certifiés Symfony et pratiquent le contrôle continue pendant les formations en présentiel. Des ordinateurs sont mis à la disposition des participants sur demande.

Formation en présentiel  
Développement web avec Symfony



Formation continue en  
e-learning pendant 1 mois

- Plateforme accessible dès le début de la formation initiale
- Support des cours de la formation initiale directement consultables et téléchargeables
- Ressources complémentaires à la formation initiale : exercices, vidéos, tests, évaluation des compétences...
- Chat en ligne avec le formateur

# **PROGRAMME FORMATION DEVELOPPEMENT WEB**

## **AVEC SYMFONY3**

### **Découvrir SYMFONY 3**

- Présentation générale du framework et de ses outils
- Installation d'une « Distribution Standard Symfony »
- Architecture d'un projet Symfony (application, bundles, librairies, configuration etc.)
- Traitement d'une requête HTTP avec Symfony
- Comprendre la notion d'environnement (développement, production et test)
- Conventions de codage et de nommage des fichiers
- Introduction au concept de « Bundle »

### **Savoir déboguer son code**

- La barre de débogage (« Web Debug Toolbar »)
- La journalisation des erreurs (« logs »)
- L'interface de profilage des requêtes (« Web Profiler »)
- Les traces d'exception et d'erreur (« stack traces »)
- Les outils des composants Debug et VarDumper

### **Maîtriser la configuration**

- Enregistrer des routes (YAML et annotations)
- Contraindre les routes (méthodes HTTP, nom de domaine, paramètres, etc.)
- Configurer un « bundle »
- Définir des paramètres globaux de configuration
- Extraire les paramètres sensibles dans des variables d'environnement
- Comprendre la surcharge de configuration par environnement

### **Répondre à la requête de l'utilisateur**

- Développer des classes de contrôleur et des méthodes d'action
- Transformer un objet de requête en objet de réponse
- Associer des URLs aux contrôleurs grâce aux annotations
- Générer une réponse brute ou à partir d'un gabarit Twig
- Accéder à l'objet de requête pour lire les données du client
- Lire et écrire des données dans la session de l'utilisateur
- Générer une page de type 404
- Savoir rediriger l'utilisateur vers une autre page

### **Générer des pages Web dynamiques**

- Découvrir le moteur de rendu Twig
- Concevoir des gabarits concis et explicites
- Accéder aux variables globales Twig
- Maîtriser la grammaire Twig (tags, filtres, fonctions, testeurs, etc.)
- Construire des pages plus complexes avec l'héritage de gabarits
- Factoriser le code dupliqué avec l'inclusion de gabarits
- Exécuter une action et afficher son résultat depuis une vue
- Générer des URLs à partir de la configuration des routes
- Afficher des formulaires Symfony

## Interagir avec l'utilisateur grâce aux formulaires

- Concevoir et traiter des formulaires simples
- Appliquer des règles de validation sur les données transmises
- Prototyper le rendu d'un formulaire avec Twig
- Récupérer et traiter les données validées d'un formulaire

## Internationaliser et localiser l'interface

- Installer le composant de traduction
- Changer la langue par défaut de l'application
- Définir des clés de traduction dans les vues Twig
- Générer les squelettes de fichiers de traduction (XLIFF, YAML)
- Traduire les clés de traduction
- Gérer les traductions incluant des règles de pluralisation

## Pérenniser l'application avec les tests automatisés

- Découvrir le framework de tests PHPUnit
- Comprendre les enjeux des tests automatisés
- Configurer une suite de tests automatisés
- Créer et exécuter une suite de tests unitaires
- Écrire et exécuter une suite de tests fonctionnels
- Générer des rapports de couverture de code

## Externaliser le métier dans les services

- Comprendre le principe de l'injection de dépendance
- Introduction au conteneur de services de Symfony
- Lister les services existants du framework Symfony
- Enregistrer de nouveaux services dans le conteneur
- Configurer des services avec des paramètres globaux
- Tirer profit de l'enregistrement automatique des services
- Configurer manuellement certains services métiers

## Sécuriser l'accès à certaines pages de l'application

- Introduction aux principes d'authentification et d'autorisation
- Sécuriser certaines parties d'un site avec une authentification par formulaire Web
- Établir une politique de droits d'accès selon les rôles de l'utilisateur courant
- Tester les droits de l'utilisateur courant depuis un contrôleur ou une vue

## Améliorer les performances avec le cache HTTP et ESI

- Introduction au cache HTTP
- Découvrir les différents types de cache
- Comparer les stratégies de cache : expiration versus validation
- Appliquer les règles d'expiration et de validation des pages
- Cacher des fragments dynamiques de page avec les ESIs (« Edge Side Includes »)
- Installer et configurer le proxy inversé natif de Symfony

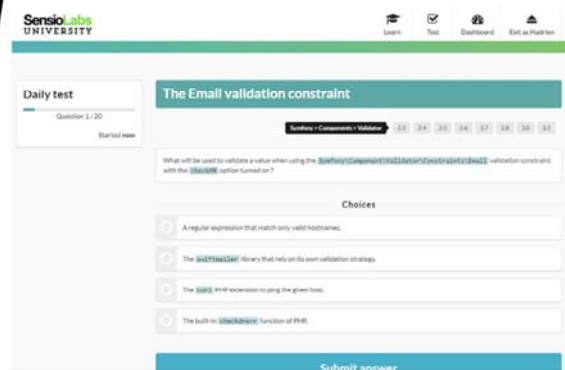
## PLATEFORME DE REVISION EN LIGNE

### TESTS QUOTIDIENS

Chaque jours nous proposons aux développeurs de passer un test d'évaluation

Sur le principe du QCM les tests se composent de 20 questions

Nous avons constitué une base de plus de 1200 questions sur PHP, la programmation orientée objet, Symfony, Doctrine, Twig ...



### GRAPHES DE COMPETENCES

Nous utilisons les résultats de chaque test pour représenter les compétences sous forme de graphes

D'un coup d'œil le développeur peut donc constater de son niveau sur chaque thématique et composant que constitue Symfony

C'est un excellent moyen pour visualiser les sujets sur lesquels il doit encore progresser

