Organisme partenaire Sensiolabs

CODE RHAPSODIE - Organisme de Formation
N° de déclaration : 84691556469
12 bis avenue Joannès Masset
69009 LYON - **04 28 29 77 56**
www.code-rhapsodie.fr
contact@code-rhapsodie.fr

SF4C4 : Découvrir et maîtriser Symfony 4

Informations

Durée : 4 jours soit 28 heures
Code : SF4C4

Niveau : Débutant
Public : Développeurs·euses

Prérequis

Bonnes connaissances de PHP :

- PHP et de la programmation orientée objet.

Conditions matérielles

PHP 7+ installé et opérationnel

Evaluation des compétences

L'évaluation des compétences sera effectuée tout au long de la formation par des cas pratiques qui permettent de valider les blocs de compétence. Une évaluation finale est effectuée en fin de stage.

Moyens pédagogiques

- Formation en petit comité (max 8 personnes)
- Alternance de théorie et d'ateliers pratiques
- **Accès à la plateforme de révision e-learning Sensiolabs pendant 1 mois**
- Support de cours consultables dans l'espace personnel Sensiolabs de chaque participant.
- Formateurs certifiés Symfony
- Evaluation en contrôle continue pendant la formation en présentiel.

Objectifs

Cette formation vous permettra :

- de découvrir l'essentiel du framework Symfony 4,
- d'apprendre à maîtriser le modèle MVC ainsi que les formulaires, l'authentification ou bien encore les tests unitaires et la gestion du cache HTTP.

Programme de la formation

Programme et support élaborés par

SensioLabs

Découvrir Symfony 4

- Présentation générale du projet Symfony 4 (code source, licence, documentation, communauté, philosophie, écosystème, etc.)
- Comprendre le cycle de publication des versions Symfony
- Initier un nouveau projet Symfony 4 avec Composer et Flex
- Apprivoiser l'organisation des fichiers du projet
- Installer des modules complémentaires (bundles, librairies) avec Flex
- Trouver de l'aide et des ressources documentaires
- Comprendre la mécanique de traitement d’une requête HTTP avec Symfony
- Découvrir le concept d'environnement (développement, production et test)
- Développer en suivant les conventions de codage et de nommage des fichiers
- Prendre en main l'utilitaire CLI et ses commandes

Savoir Déboguer son Code

- Installer le paquet « Web Profiler » avec Symfony Flex
- Activer la journalisation des erreurs et des incidents
- Afficher la barre de débogage (« Web Debug Toolbar ») sur les pages Web
- Naviguer dans les différents panneaux du « Web Profiler »
- Analyser les traces d'exception et les messages d'erreur
- Introspecter le contenu d'une variable

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
- Récolter 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 automatiques 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