PHP : construire son propre framework

Développement ▸ Web avancé

Présentation

Un framework (ou cadriciel, ou infrastructure logicielle) n'est pas qu'un ensemble de fonctions disponibles : il structure le développement. Cette structure s'obtient en imposant l'ordre dans lequel s'exécute votre code, en contraignant les endroits dans lesquels votre code peut être rédigé ou encore en imposant des conventions de nommage.
Que vous souhaitiez créer votre propre framework, contribuer aux frameworks existants, mieux comprendre comment ceux-ci fonctionnent ou tout simplement améliorer vos compétences dans la création d'interfaces de programmation, cette formation vous propose de découvrir les composantes essentielles des frameworks : l'inversion de contrôle, les design patterns (patrons de conception) utilisés et les bonnes pratiques à respecter.

Objectifs

- Savoir expliquer et utiliser l'inversion de contrôle
- Savoir intégrer des bibliothèques existantes
- Savoir utiliser à son avantage les patrons de conception (design patterns)
- Apprendre à développer en laissant des espaces de configuration
- S'initier à déterminer une syntaxe adaptée pour l'interface de programmation

Moyens pédagogiques

Formation alternant la théorie et la pratique, avec un intervenant présentant la formation en direct. Possibilité de réaliser cette formation en face-à-face dans nos locaux, dans vos locaux d'entreprise ou à distance via visioconférence.
Une fois terminée, vous disposerez d'une attestation de suivi de la formation, ainsi que des contenus de cours relatifs au contenu présenté.

Evaluation des acquis

Questionnaires à choix multiples (QCM) en fin de formation.
L'intervenant formulera également une appréciation sur vos acquis pour l'entièreté de la formation.

Durée : 21 heures (3 jours)
Référence : DE-086
Niveau : Avancé
Formation inter-entreprises :

Formation en présentiel et distanciel
Tarif : 1490 € HT

Formation intra-entreprises :

Formation disponible et adaptable sur demande
Tarif : sur devis

Contact

Anthony DUPUIS
Conseiller formation
07 68 53 96 11
anthony@ls-a.fr

Public :

Toute personne souhaitant améliorer ses compétences en développement informatique.

Avez-vous les prérequis ?

Une connaissance de PHP équivalente à la formation DE-083 (PHP : Initiation à Laravel) est nécessaire
Une connaissance de la programmation orientée objet équivalente à la formation DE-088 (POO et Design Patterns) est nécessaire
Une connaissance des fonctionnalités avancées de PHP équivalente à la formation DE-084 (PHP : créer un environnement robuste et résilient) est un plus

Dernière modification :

11/01/2023

Document non contractuel. ls-a Formation est une marque déposée de ls-a. Tous droits réservés.

PHP : construire son propre framework

Développement ▸ Web avancé

Programme de la formation

1- Cadrer le développement

Ce qu'est et ce que n'est pas un framework
Notions d'inversion de contrôle
Quelle philosophie ?
Déterminer les objectifs du framework
Les pièges à éviter

2- Inverser le contrôle

Offrir des espaces de développement en verrouillant le fonctionnement général
Définir un cycle de vie adapté : démarrage, routage, gestion d'erreurs, ...
Création de systèmes d'injection de dépendances
Permettre la surcharge d'un comportement
Système de « hooks » (grappins), système événementiel, système de fichiers

3- Séparer la définition de l'implémentation

Notion d'interface au sens général
Différents modèles de pilotes (drivers) : avantages et inconvénients
Algorithme par chaînage et design pattern « Chaîne de responsabilité »
Verrouiller le typage sans coupler du code : analyse et solutions

4- Offrir des éléments de configuration

Quelle syntaxe pour quels usages ? Avantages et inconvénients
Choisir entre dotenv, JSON, YAML, XML, base de données ...
Interpréter la configuration : bonnes pratiques et conventions
Adapter le cycle de vie du framework à la configuration

5- Intégrer et organiser des bibliothèques externes

Composants conseillés : tests, linter, HTTP, appels système, documentation
Respect des PSR (PHP Standard Recommandations)
Transformer son framework en module composer
Définir une structure de dossiers adaptée
Créer un package privé via composer