Programme de formation

Magento 2 : Back End

Concepts et bonnes pratiques de développement Back Office avec Magento 2

Durée

5 jours / 35 heures

Description

La formation Magento 2 Back End est une formation technique qui permet d'acquérir les compétences nécessaires pour pouvoir comprendre et étendre les possibilités fonctionnelles de Magento 2. Orienté travaux pratiques, vous aborderez l'ensemble des aspects du développement sous Magento 2

Objectifs pédagogiques

- Installer Magento 2
- Créer un module Front et BackOffice Magento 2 en respectant les bonnes pratiques de conception, codage et de test
- Mettre en place ou enrichir les APIs Magento 2

Public

Profils à dominante technique : Lead Dev, Ingénieur de développement, Développeur junior et senior souhaitant découvrir le framework Magento 2

Pré-requis

Expérience en développement objet PHP et de préférence connaissant Magento 1

Méthodes pédagogiques

50 % théorie / 50 % pratique

Profil intervenant

L'ensemble de nos formations sont animées par des formateurs expérimentés possédant une expérience terrain éprouvée.

Modalités d'évaluation

L'évaluation des acquis se fait tout au long de la session au travers d'ateliers de mise en pratique des notions et concepts abordés pendant la formation.

Programme

Environnement

- Environnement Magento
- Environnement de développement
- Installation de Magento 2
- Configuration de l'environnement de développement
- Les modes : Default, Developer, Production, Maintenance
- Arborescence des fichiers

Instanciation des objets

- Injection de dépendance (Dependency Injection)
- Object Manager
- Compilateur

Modules Magento

- Présentation générale
- Qu'est-ce qui constitue un module
- Contenu minimum d'un module
- Classes d'un module : PSR-2, autoloader, dependency injection, organisation…
- Fichiers XML d'un module : validation obligatoire, cloisonnement par area
- Traductions
- Les lignes de commande (Magento CLI)
- **TP1**
    - création d'un module Magento 2
    - création d'une ligne de commande

Routage et Controllers

- Présentation générale
- Les différents Routeurs disponibles
- Principe de fonctionnement d'un Routeur
- Controllers (alias Action Classes)
- Interprétation d'une URL module/action-path/action
- Configuration du routage pour chaque module
- Résulats possibles: page, JSON, redirection…
- Instanciation à l'intérieur d'un controller (Factory)
- **TP2**
    - création d'un module Magento 2
    - création de 2 controllers
    - traitement des paramètres
    - mettre en place des redirections
    - création d'un routeur spécifique

Base de données et ORM

- **Models**
  - Définitions: ORM, Models, Resource Models, Collections, Resource Adapters
  - Généralités pour le CRUD
  - Liaison Model / Resource Model
  - Liaison Model / BDD
  - Liaison Collection / Model
  - Model Type Interface
  - Vue détaillée : Model, Resource Model, Collection
  - **TP3**
    - création d'un module Magento 2
    - creation de controllers:
      - recuperer des paramètres
      - charger correctement les données
      - afficher un simple retour (JSON ou autre)

- **Mise à jour BDD**
  - Ancien système: Scripts d'install et d'upgrade
    - Install et Upgrade, Schema et Data)
  - Nouveau système:
    - schéma déclaratif (db_schema.xml)
    - data patches
  - Installation d'un module
  - Exemples de scripts
  - **TP4**
    - création d'un module Magento 2
    - creation de nouvelles tables de données via le système déclaratif
    - création des classes modèles/resource/collections associé à ses nouvelles tables
    - ajout de fonctionnalités dans le resource model et dans la resource collection
    - création de controllers permettant d'afficher et de filtrer les entités des nouvelles tables
    - factorisation du code des controllers

- **Modèle EAV (Entity-Attribute-Value)**
  - Concept
  - Méta-tables
  - Models utilisés
  - Chargement et sauvegarde (Différences dans les implémentations des classes ORM, Process de chargement, Source Models : utilisation des valeurs ou labels pour les attributs de type "liste déroulante")
  - Gestion des attributs (Deux aspects de l'EAV : Méta-information et Contenu, Types des valeurs, Classe Setup spécifique, Création d'un nouvel attribut, Modèles des attributs: Backend, Source et Frontend)
  - Entity Increment Model
  - **TP5**
    - création d'un module Magento 2
    - création d'un attribut EAV « series » et son frontend model

Layouts

- Définition
- Les trois types de layout (page layout, configuration de page, layout générique)
- Utilisation des fichiers de layout : Conventions de nommage
- Contenus d'un layout:
    - Containers,
    - Blocks : Les différents types de Blocks prédéfinis
- Templates et leur utilisation
- UiComponents :
    - Cas d'utilisation
    - Définition
    - UiComponents basiques et secondaires
    - Templates des UiComponents
    - Utilisation des UiComponents
    - Exemples
- Références vers les Containers et Blocks existants
    - Actions sur les Blocks
    - Déplacement et suppression des Blocks et Containers
- Le Full Page Cache (FPC)
- **TP6**
    - création de plusieurs modules Magento 2 avec dépendances
    - création de controllers
    - création de layout
    - création et placement de blocks
    - internationalisation (traductions)

Évènements et Observers, Crons & Plugins

- Évènements **et Observers**
    - Déclencher un évènement
    - Déclarer un Observer
- Crons
    - Déclencher de nouveaux crons
    - Les groupes
    - Lignes de commande (CLI)
- Plugins **Magento**
    - Concept et limitations
    - Déclaration
    - Convention de nommage
- **TP7**
    - création d'un module avec dépendance
    - création d'un observeur
    - création de plusieurs plugins
    - création d'un controller avec paramètres qui utilise les plugins

Tests unitaires (PHPUnit)

- Présentation
- Commandes: installation, exécution
- Création d'un TU
- Méthodes de tests, assertions, mocks
- **TP8**
    - création d'un module avec dépendance
    - création d'une classe de test pour l'observer du TP7
    - création de 3 tests unitaires

Service Contracts, API et Services Web

- **Service Contracts**
    - Définition, avantages, inconvénients
    - Composition d'un Service Contract : Data API, Service API
    - Tags dans les commentaires nécessaires pour la génération des Web Services
    - Implémentation des DTO (Data Transfer Objects) : "Custom Attributs" et "Extension Attributes": déclaration et injection
- **Service API**
    - Business Logic API
    - Repositories
- **Service Web**
    - Publication d'un WS REST ou SOAP, D
    - Déclaration d'une méthode de webservice et mapping avec une interface Magento,
    - API REST : paramètres et requêtes via POST et PUT,
    - API SOAP: WSDL auto-généré et son URL,
    - Gestion des ACL (permissions) et authentification, Paramétrage en BO
- **TP9**
    - création d'un module avec dépendance
    - création de plusieurs controllers
    - utiliser un Repository Standard magento pour récupérer les informations
    - créer un Repository/interface sur les entités spécifiques créés dans les précédents TP
    - implémenter des Data Object
    - Créer une API REST et la tester

Back-Office

- Présentation
- Implémentation d'un Controller BO en tenant compte des ACL
- Éléments récurrents: grilles et formulaires: UiComponents
- Présentation des UiComponents dédiés: "listing" (grille) et "form" (formulaire) : Définition des composants enfants, Création des sources de données (composant dataSource et classe dataProvider), Exemples
- Pour les grilles : Définitions des filtres et des colonnes, Autres composants moins complexes: actions en masse, paginations etc
- Pour les formulaires : Déclaration des champs et leur organisation en fieldsets., Déclaration des boutons, Validation

- System Configuration
- Présentation de la configuration BO
- Déclaration d'un point de config
- Inclusion d'un Source Model et d'un Frontend Model
- Menu : Configuration du menu de navigation en BO
- ACL
- Définition
- Organisation des ACL par rôles
- Exemples de contrôle des ACL: _isAllowed()
- Configuration dans la BO
- Création d'ACL

- **TP10**
    - création d'un module avec dépendance
    - création d'ACL
    - ajout d'entrée dans menu
    - création de grid et de formulaire