

## Avant-propos

<b>1. Lectorat</b> .....	<b>15</b>
<b>2. Contenu du livre</b> .....	<b>15</b>
<b>3. Pré-requis</b> .....	<b>17</b>

## Introduction

<b>1. Un peu d'histoire</b> .....	<b>19</b>
1.1 Petite histoire de l'informatique .....	19
1.2 Petite histoire des langages informatiques .....	20
1.3 Petite histoire de PHP .....	22
1.4 Petite histoire de Zend Framework .....	24
1.5 Regard vers le futur .....	24
<b>2. Choix</b> .....	<b>25</b>
2.1 Choix du langage .....	25
2.2 Choix du framework .....	28
<b>3. Installations</b> .....	<b>30</b>
3.1 Installer un serveur Apache - PHP .....	30
3.1.1 Sous une distribution Linux en utilisant le gestionnaire de paquets .....	30
3.1.2 Par la compilation .....	30
3.1.3 Sous Windows .....	34
3.2 Installer des serveurs PostgreSQL, MySQL et SQLite .....	35
3.3 Installation de PEAR, PECL et d'autres packages .....	36
<b>4. Logiciels utiles</b> .....	<b>38</b>
4.1 Pour Apache .....	38
4.2 Pour MySQL .....	38

4.3 Pour PostgreSQL .....	39
4.4 Pour SQLite .....	39
4.5 Environnement de développement .....	40
4.6 Intégration continue .....	40

## Un peu de théorie

<b>1. Génie logiciel .....</b>	<b>41</b>
1.1 IHM .....	41
1.2 Client/Serveur .....	42
1.3 3-tier .....	43
1.4 MVC en général .....	44
1.5 MVC appliqué au Web .....	45
<b>2. Application à PHP .....</b>	<b>45</b>
2.1 Présentation des serveurs web .....	45
2.2 Grands principes de PHP .....	46
2.3 Points forts et points faibles de PHP .....	47
2.4 La place de PHP .....	50
2.5 Industrialisation des développements .....	50
<b>3. Bases de données .....</b>	<b>52</b>
3.1 Qu'est-ce qu'une base de données relationnelle ? .....	52
3.2 Les bases relationnelles courantes .....	53
3.3 Présentation rapide de PDO .....	53
<b>4. La solution Zend Framework .....</b>	<b>54</b>
4.1 Pourquoi choisir Zend Framework .....	54
4.2 Structurer ses développements .....	55
4.3 Répartir les tâches .....	57
4.4 Maîtriser les coûts .....	57
<b>5. Qualité logicielle .....</b>	<b>58</b>

5.1 Intégration continue .....	58
5.2 Tests unitaires .....	58
5.3 Métriques .....	59
5.4 Détection d'erreurs .....	63
5.5 Duplication de code .....	64
5.6 Lisibilité .....	65

## Application blanche

<b>1. Introduction .....</b>	<b>67</b>
1.1 Application blanche .....	67
1.2 Squelette d'application .....	68
<b>2. Squelette de l'application .....</b>	<b>69</b>
2.1 Installation du ZendSkeletonApplication .....	69
2.2 Déclaration des autres dépendances .....	71
2.3 Mettre à jour les dépendances .....	73
2.4 Analyse succincte du code obtenu .....	73
2.5 Recommandations générales .....	76
<b>3. Configuration du serveur .....</b>	<b>77</b>
3.1 Création d'un hôte virtuel .....	77
3.2 Création d'un alias .....	79
3.3 Recommandations générales .....	81
3.3.1 Apache .....	81
3.3.2 PHP .....	82
<b>4. Création de l'application blanche .....</b>	<b>83</b>
4.1 Bibliothèques personnalisées .....	83
4.2 Jeu de templates .....	84
4.3 Gestion de l'authentification .....	85
4.4 Modules réutilisables ou fréquemment utilisés .....	86
4.5 Recommandations générales .....	87

## Le module, notions de base

<b>1. Introduction et installation</b> .....	<b>89</b>
1.1 Notion de module .....	89
1.2 Initialisation du dépôt de l'application galerie .....	90
1.3 Création d'un hôte virtuel .....	92
1.4 Création d'un alias .....	93
<b>2. Initialisation d'un module</b> .....	<b>94</b>
2.1 Création de l'arborescence d'un module .....	94
2.2 Création du fichier Module.php .....	96
2.3 Gestion du chargement automatique .....	96
2.3.1 AutoloaderProviderInterface .....	96
2.3.2 StandardAutoloader .....	97
2.3.3 ClassMapAutoloader .....	97
2.4 Gestion des sources .....	99
2.4.1 ConfigProviderInterface .....	99
2.4.2 Configuration minimaliste .....	100
2.4.3 Déclaration de la template_map .....	100
2.5 Gestion des routes .....	102
2.6 Création du contrôleur et des vues .....	107
2.7 Configuration de l'internationalisation .....	109
2.7.1 Fichier de configuration .....	109
2.7.2 Module.php .....	111
2.7.3 Installation et configuration de Poedit .....	112
2.7.4 Création du projet Poedit .....	114
2.7.5 Modification des vues .....	116
2.8 Configuration de l'accès aux modèles .....	118
2.9 Configuration d'aide de vue .....	120
<b>3. Modèles Zend\Db - Notions de base</b> .....	<b>122</b>
3.1 Introduction .....	122
3.2 Classe Entité (Entity) .....	125
3.2.1 Introduction .....	125
3.2.2 Utilisation des accesseurs et des mutateurs .....	130

3.2.3 Utilisation des tableaux .....	134
3.3 Classe Gestionnaire (EntityManager) .....	139
3.3.1 Premier jet .....	139
3.3.2 Création d'un gestionnaire abstrait .....	143
3.3.3 Extension du modèle à des fonctionnalités plus larges .....	149
3.4 Création de requêtes SQL .....	151
3.4.1 Se donner les moyens de voir ce que l'on fait .....	151
3.4.2 Possibilités offertes par select .....	153
3.4.3 Possibilités offertes par insert .....	159
3.4.4 Possibilités offertes par update .....	162
3.4.5 Possibilités offertes par delete .....	163
3.5 Entité et gestionnaire pour plusieurs tables .....	163
3.6 Utilisation du modèle .....	168
<b>4. Contrôleurs - Notions de base .....</b>	<b>169</b>
4.1 Introduction .....	169
4.2 Gestion de l'accès aux données .....	171
4.3 Affichage d'une liste de données .....	173
4.4 Affichage d'une donnée unique .....	174
4.5 Traitement de la requête .....	176
4.6 Traitement de la réponse .....	179
<b>5. Vues - Notions de base .....</b>	<b>181</b>
5.1 Introduction .....	181
5.2 Organisation de sa présentation .....	182
5.3 Utilisation des aides de vues .....	184
<b>6. Formulaires - Notions de base .....</b>	<b>185</b>
6.1 Introduction .....	185
6.2 Partage des responsabilités .....	186
6.3 Création d'un formulaire .....	188
6.4 Impacts sur un modèle Zend\Db .....	190
6.5 Affichage du formulaire dans la vue .....	194
6.6 Utilisation d'un formulaire .....	195
6.7 Internationalisation .....	199

6.7.1 Labels, version simple .....	199
6.7.2 Labels, version élaborée .....	201
6.7.3 Messages d'erreurs .....	206

## Le module, aspects avancés

<b>1. Les formulaires - Notions avancées .....</b>	<b>209</b>
1.1 Réalisation d'un composant générique .....	209
1.1.1 Objectifs pour le formulaire .....	209
1.1.2 Objectifs pour l'entité .....	212
1.1.3 Classes permettant d'atteindre ces objectifs .....	214
1.2 Présentation des différents éléments de formulaire .....	219
1.2.1 Présentation générale .....	219
1.2.2 Champs simples .....	220
1.2.3 Label .....	222
1.2.4 Champs temporels .....	224
1.2.5 Champs nécessitant des données .....	224
1.2.6 Champs multiples .....	226
1.3 Utiliser Ajax .....	227
1.4 Personnaliser le rendu graphique .....	231
<b>2. Les contrôleurs - Notions avancées .....</b>	<b>232</b>
2.1 Créer un contrôleur Ajax .....	232
2.2 Créer des interfaces plus conviviales .....	234
2.3 Aides d'action .....	240
2.3.1 Présentation générale .....	240
2.3.2 Redirection .....	241
2.3.3 Params .....	242
2.3.4 FlashMessenger .....	243
2.3.5 Forward .....	246
2.4 Créer une aide d'action .....	247
<b>3. Les vues - Notions avancées .....</b>	<b>249</b>
3.1 Configuration avancée des vues .....	249

3.2 Création d'un layout .....	251
3.3 Relations contrôleur/layout/vues .....	254
3.4 Création d'aides de vues .....	255
3.5 Utiliser les possibilités du langage HTML5 .....	259
3.6 Utiliser des frameworks ou des bibliothèques JavaScript .....	259
3.7 Rajouter des frameworks CSS .....	260

## Plus loin avec les modèles

<b>1. Modèles Zend\Db - Notions avancées .....</b>	<b>261</b>
1.1 Créer une requête pour une liste de sélection .....	261
1.1.1 Contexte .....	261
1.1.2 Composant générique .....	262
1.1.3 Utilisation du composant .....	267
1.2 Hydrator .....	269
1.2.1 Utilisation des accesseurs et mutateurs .....	269
1.2.2 Utilisation des propriétés .....	272
1.2.3 Tableau .....	274
1.2.4 Hydrating .....	275
1.2.5 Stratégies .....	275
1.2.6 Lien avec les formulaires .....	275
<b>2. Modèles Doctrine .....</b>	<b>276</b>
2.1 Introduction .....	276
2.2 Classe modèle .....	276
2.3 Utilisation du modèle .....	279

## Outils transverses

<b>1. Session .....</b>	<b>283</b>
1.1 Stocker un élément de la session .....	283
1.1.1 Conteneur .....	283
1.1.2 Configurations .....	285

<b>2. Système de cache</b> .....	<b>287</b>
2.1 APC .....	287
2.2 Autres systèmes de cache .....	288
<b>3. Logging</b> .....	<b>289</b>
3.1 Écrire dans un fichier de log .....	289
3.2 Logging interactif .....	291
<b>4. Gestion des identités et des permissions</b> .....	<b>292</b>
4.1 Outils proposés par ZF2 .....	292
4.1.1 Authentification .....	292
4.1.2 Permissions .....	293
4.2 ZfcUser .....	294
4.2.1 Introduction .....	294
4.2.2 Installation du projet .....	295
4.2.3 Modification de la base de données .....	296

## Génération de contenu tiers

<b>1. CSV</b> .....	<b>297</b>
1.1 Pré-requis .....	297
1.1.1 Désactiver la partie vue .....	297
1.1.2 Gérer la forme de la réponse .....	297
1.2 Réalisation de l'export CSV .....	298
1.2.1 Gérer les données .....	298
1.2.2 Envoyer le résultat .....	299
<b>2. Excel</b> .....	<b>300</b>
2.1 Pré-requis .....	300
2.1.1 Installation d'un module PEAR .....	300
2.1.2 Chargement à la demande du module .....	301
2.2 Réalisation de l'export Excel .....	302



2.2.1 Création d'un composant générique .....	302
2.2.2 Réalisation d'un exemple concret .....	311
<b>3. Courriels .....</b>	<b>315</b>
3.1 Envoyer un courriel .....	315
3.2 Gérer l'envoi aux formats Texte et HTML .....	318
3.3 Utiliser des templates .....	319
<b>4. Diagrammes .....</b>	<b>321</b>
4.1 Création d'un composant générique .....	321
4.2 Création d'un composant concret .....	324
4.3 Création du contrôleur .....	326
4.4 Affichage du graphique .....	327

## Communication inter-applicative

<b>1. XML-RPC .....</b>	<b>329</b>
1.1 Introduction .....	329
1.2 Serveur .....	329
1.3 Client .....	330
<b>2. SOAP .....</b>	<b>331</b>
2.1 Introduction .....	331
2.2 Serveur .....	331
2.3 Client .....	331
<b>3. RSS .....</b>	<b>332</b>
3.1 Writer .....	332
3.2 Reader .....	336
<b>4. REST .....</b>	<b>338</b>
4.1 Introduction .....	338
4.2 Utilisation .....	338

## Sécurité

<b>1. Considérations générales sur la sécurité</b> .....	<b>341</b>
<b>2. Consignes élémentaires</b> .....	<b>343</b>
2.1 Votre application vit dans un écosystème .....	343
2.2 Suivre des règles de codage claires .....	344
2.3 Toujours vérifier les données .....	345
2.4 Bien utiliser la session .....	347
2.5 Ne pas donner d'indices .....	348
<b>3. Mettre en place une politique de sécurité</b> .....	<b>348</b>
3.1 Paramétrage des serveurs .....	348
3.2 La session .....	351
3.3 Utilisation du cryptage .....	353
3.4 Processus de création d'un compte utilisateur .....	353
3.5 Stockage du mot de passe .....	355
3.6 Processus d'authentification .....	355
3.7 Utilisation des courriels .....	356
3.8 Conclusion .....	357
<b>4. Points faibles usuels</b> .....	<b>357</b>
4.1 XSS .....	357
4.2 CSRF .....	358
4.3 Fixation de session .....	358
4.4 Injections de code SQL .....	359
4.5 Conclusion .....	360

## Concepts clés utilisés par ZF2

<b>1. Nouvelles pratiques</b> .....	<b>361</b>
1.1 Espaces de nommage .....	361
1.2 Fonctions anonymes .....	362

1.3 Closures .....	363
1.4 Inclusion de tableaux .....	364
1.5 Utilisation d'interfaces .....	365
<b>2. Motifs de conception usuels .....</b>	<b>366</b>
2.1 Singleton .....	366
2.2 Fabrique .....	368
2.3 Proxy .....	369
2.4 Template .....	370
2.5 Décorateur .....	371
2.6 Prototype .....	372
2.7 Adaptateur .....	372
2.8 Itérateur .....	373
2.9 Chaîne de responsabilité .....	376
2.10 Fonction de rappel .....	378
<b>3. Injection de dépendance .....</b>	<b>380</b>
3.1 Explication de la notion .....	380
3.2 Injecteur de dépendance .....	383
3.2.1 Introduction .....	383
3.2.2 Définir les dépendances précisément .....	384
<b>4. Annotations .....</b>	<b>386</b>
4.1 Explication de la notion .....	386
4.2 Gestionnaire d'annotations .....	388
<b>5. Programmation événementielle .....</b>	<b>389</b>
5.1 Explication de la notion .....	389
5.2 Gestionnaire d'événements .....	390
5.2.1 Description des composants ZF2 .....	390
5.2.2 Utilisation du gestionnaire d'événements .....	399

## Réponses aux problématiques de performance

<b>1. Modèles</b>	<b>401</b>
1.1 Règles générales	401
1.2 Optimisations côté base de données	405
1.2.1 Mieux vaut faire travailler SQL que PHP	405
1.2.2 Recommandation de conception pour les clés primaires	405
1.2.3 Utilisation des index	406
1.2.4 Utilisation des déclencheurs	407
1.3 Optimisations côté ZF2 pour l'accès aux données	412
1.3.1 Mieux vaut faire travailler SQL que PHP	412
1.3.2 Recommandation d'utilisation pour les clés primaires	412
1.3.3 Utilisation des vues	413
1.3.4 Le point de vue extrême	415
1.4 Optimisations côté ZF2 pour l'écriture de données	416
1.4.1 Mieux vaut faire travailler SQL que PHP	416
1.4.2 Cas particulier du clonage de données	416
1.4.3 Modifier plusieurs enregistrements en une requête	417
1.4.4 Le point de vue extrême	417
1.5 Cache de données	417
1.6 Outils tiers	418
<b>2. Architecture</b>	<b>419</b>
2.1 Les bases	419
2.1.1 Chargement automatique	419
2.1.2 Correction d'erreurs	419
2.1.3 Trop grand nombre de classes	420
2.1.4 Trop grand nombre de méthodes	421
2.2 Bonne utilisation de l'architecture de ZF2	422
2.3 Utiliser un cache d'opcode	422
<b>3. Évaluer son propre code</b>	<b>423</b>
3.1 Évaluation générale	423
3.2 Évaluation fine	423
Index	425