

ZEND Framework IN ACTION

Rob Allen
Nick Lo
Steven Brown



contents

preface xvii
acknowledgments xix
about this book xxi
about the cover illustration xxvi

PART 1 THE ESSENTIALS1

1 *Introducing Zend Framework 3*

1.1 Introducing structure to PHP websites 4

1.2 Why use Zend Framework? 5

Everything is in the box 6 ▪ *Modern design 6* ▪ *Easy to learn 6* ▪ *Full documentation 7* ▪ *Simple development 8*
Rapid development 8 ▪ *Structured code is easy to maintain 8*

1.3 What is Zend Framework? 9

Where did it come from? 9 ▪ *What's in it? 9*

1.4 Zend Framework design philosophy 15

High-quality components 15 ▪ *Pragmatism and Flexibility 15*
Clean IP 16 ▪ *Support from Zend Technologies 16*

- 1.5 Alternative PHP frameworks 16
- 1.6 Summary 17

2 **Hello Zend Framework! 18**

- 2.1 The Model-View-Controller design pattern 19
 - The model* 20 ▪ *The view* 20 ▪ *The controller* 20
- 2.2 The anatomy of a Zend Framework application 21
 - The application directory* 21 ▪ *The library directory* 21
 - The tests directory* 21 ▪ *The public directory* 22
- 2.3 Hello World: file by file 22
 - Bootstrapping* 22 ▪ *Apache .htaccess* 24
 - Index controller* 25 ▪ *View template* 26
- 2.4 How MVC applies to Zend Framework 27
 - Zend Framework's controller* 28 ▪ *Understanding Zend_View* 32 ▪ *The model in MVC* 35
- 2.5 Summary 39

PART 3 A CORE APPLICATION41

3 **Building a website with Zend Framework 43**

- 3.1 Initial planning of a website 44
 - The site's goals* 44 ▪ *Designing the user interface* 45
 - Planning the code* 47
- 3.2 Initial coding 48
 - The directory structure* 48 ▪ *The Bootstrap class* 48
 - Running the application* 53
- 3.3 The home page 53
 - The initial models* 54 ▪ *Testing our models* 56
 - The home-page controller* 59
- 3.4 Summary 63

- ## 4 *Managing the view* 64
- 4.1 Introducing the Two Step View and Composite View patterns 65
 - 4.2 View handling with Zend_Layout 66
 - 4.3 Integrating Zend_Layout into Places 67
 - Setup* 67
 - *Layout scripts* 69
 - *Common actions using placeholders* 72
 - *The homepage view script* 76
 - 4.4 Advanced view helpers 78
 - Controller integration* 78
 - *View script management* 80
 - HTML header helpers* 81
 - 4.5 Summary 86
- ## 5 *Ajax* 87
- 5.1 Introducing Ajax 88
 - Defining Ajax* 88
 - *Using Ajax in web applications* 89
 - 5.2 A simple Ajax example 91
 - 5.3 Using Ajax client libraries 94
 - 5.4 Using Ajax with Zend Framework 95
 - The controller* 96
 - *The view* 97
 - 5.5 Integrating into a Zend Framework application 98
 - The Place controller* 99
 - *Adding review rating HTML to the view script* 101
 - *Adding JavaScript to the view scripts* 102
 - The server code* 104
 - 5.6 Summary 106
- ## 6 *Managing the database* 107
- 6.1 Database abstraction with Zend_Db_Adapter 108
 - Creating a Zend_Db_Adapter* 108
 - *Querying the database* 109
 - Inserting, updating, and deleting* 110
 - *Handling database-specific differences* 111
 - 6.2 Table abstraction with Zend_Db_Table 112
 - What is the Table Data Gateway pattern?* 112
 - *Using Zend_Db_Table* 113
 - *Inserting and updating with Zend_Db_Table* 114
 - *Deleting records with Zend_Db_Table* 116

- 6.3 Using Zend_Db_Table as a model 116
 - Testing the model* 118
 - *Table relationships with Zend_Db_Table* 123
- 6.4 Summary 127

7 *User authentication and access control* 128

- 7.1 Introducing authentication and access control 129
 - What is authentication?* 129
 - *What is access control?* 129
- 7.2 Implementing authentication 130
 - Introducing Zend_Auth* 130
 - *Logging in using HTTP authentication* 131
- 7.3 Using Zend_Auth in a real application 133
 - Logging in* 133
 - *A view helper welcome message* 137
 - Logging out* 138
- 7.4 Implementing access control 139
 - Introducing Zend_Acl* 139
 - *Configuring a Zend_Acl object* 141
 - *Checking the Zend_Acl object* 142
- 7.5 Summary 146

8 *Forms* 147

- 8.1 Introducing Zend_Form 148
 - Integrated data filters and validators* 148
 - *Integrated error handling* 150
 - *Decorators to simplify markup* 151
 - Plug-in loaders for customization* 152
 - Internationalization* 152
 - *Subforms and display groups* 152
- 8.2 Building a login form 153
 - Setting up paths* 153
 - *Our form view script* 153
 - Updating the AuthController controller action* 15
 - The basic login form class* 155
- 8.3 Filtering and validation 156
 - Basic filtering and validation* 156
 - *Custom error messages* 158
 - *Internationalizing the form* 159
 - Adding a custom validator* 160
- 8.4 Decorating our login form 162
 - Zend_Form default decorators* 162
 - Setting our own decorators* 162
- 8.5 Summary 166

9 Searching 167

- 9.1 Benefits of search 168
 - Key usability issue for users* 168
 - Ranking results is important* 168
- 9.2 Introducing Zend_Search_Lucene 168
 - Creating a separate search index for your website* 169
 - Powerful queries* 171 ■ *Best practices* 175
- 9.3 Adding search to Places 176
 - Updating the index as new content is added* 176
 - Creating the search form and displaying the results* 185
- 9.4 Summary 188

10 Email 189

- 10.1 The basics of email 190
 - Email simplified* 190 ■ *Dissecting an email address* 191
- 10.2 Introducing Zend_Mail 191
 - Creating emails with Zend_Mail* 191
 - Sending emails with Zend_Mail* 193
- 10.3 Building a support tracker for Places 195
 - Designing the application* 195 ■ *Integrating Zend_Mail into the application* 199 ■ *Adding headers to the support email* 200
 - Adding attachments to the support email* 202
 - Formatting the email* 203
- 10.4 Reading email 205
 - Collecting and storing email* 205
 - Reading email with our application* 206
- 10.5 Summary 210

11 Deployment 211

- 11.1 Setting up the server 211
 - Designing for different environments* 212
 - Using virtual hosts for development* 214
- 11.2 Version control with Subversion 216
 - Creating the Subversion repository* 217 ■ *Checking out code from the repository* 218 ■ *Committing changes to the repository* 218
 - Updating a local working copy* 219 ■ *Dealing with conflicts* 220
 - Getting a clean copy from the repository* 222 ■ *Using branches* 222 ■ *Externals* 223

- 11.3 Functional testing 223
 - Functional testing with Selenium IDE* 224
 - *Automating Selenium IDE tests* 226
 - *Functional testing with Zend_Http_Client* 227
- 11.4 Scripting the deployment 229
- 11.5 Summary 229

PART 3 MORE POWER TO YOUR APPLICATION231

12 *Talking with other applications* 233

- 12.1 Integrating applications 234
 - Exchanging structured data* 234
 - *Producing and consuming structured data* 235
 - *How web services work* 236
 - *Why we need web services* 237
- 12.2 Producing and consuming feeds with Zend_Feed 237
 - Producing a feed* 237
 - *Consuming a feed* 239
- 12.3 Making RPCs with Zend_XmlRpc 240
 - Using Zend_XmlRpc_Server* 241
 - Using Zend_XmlRpc_Client* 248
- 12.4 Using REST web services with Zend_Rest 250
 - What is REST?* 250
 - *Using Zend_Rest_Client* 251
 - Using Zend_Rest_Server* 254
- 12.5 Summary 256

13 *Mashups with public web services* 257

- 13.1 Accessing public web services 258
 - Zend_Gdata* 258
 - *Zend_Service_Akismet* 260
 - Zend_Service_Amazon* 260
 - *Zend_Service_Audioscrobbler* 261
 - Zend_Service_Delicious* 261
 - *Zend_Service_Flickr* 261
 - Zend_Service_Gravatar* 262
 - *Zend_Service_Nirvanix* 262
 - Zend_Service_RememberTheMilk* 262
 - *Zend_Service_Simpy* 262
 - Zend_Service_SlideShare* 263
 - *Zend_Service_StrikeIron* 263
 - Zend_Service_Technorati* 263
 - *Zend_Service_Yahoo* 264
- 13.2 Displaying ads with Amazon web services 264
 - The Amazon model class* 265
 - *The Amazon ads view helper* 266
 - *Caching the view helper* 267

- 13.3 Displaying pictures from Flickr 270
 - The Flickr model class* 270
 - Using Flickr in an action controller* 271
- 13.4 Using Zend_Gdata for Google access 274
 - The YouTube API in an action controller* 274
 - The video categories page* 275
 - The video list page* 276
 - The video page* 278
- 13.5 Summary 279

14 **Caching: making it faster** 280

- 14.1 Benefits of caching 281
- 14.2 How caching works 281
- 14.3 Implementing Zend_Cache 284
 - Zend_Cache frontends* 284
 - Zend_Cache backends* 295
- 14.4 Caching at different application levels 296
 - Choosing what to cache* 296
 - Optimal cache expiry* 297
- 14.5 Cache tags 298
- 14.6 Summary 298

15 **Internationalization and localization** 299

- 15.1 Translating languages and idioms 300
 - Translating languages* 300
 - Translating idioms* 301
- 15.2 Using Zend_Locale and Zend_Translate 301
 - Setting the locale with Zend_Locale* 301
 - Translating with Zend_Translate* 303
- 15.3 Adding a second language to the Places application 305
 - Selecting the language* 305
 - The LanguageSetup front controller plug-in* 308
 - Translating the view* 310
 - Displaying the correct date with Zend_Locale* 312
- 15.4 Summary 314

16 **Creating PDFs** 316

- 16.1 Zend_Pdf basics 317
 - Creating or loading documents* 317
 - Creating pages in your PDF document* 317
 - Adding document meta-information* 318
 - Saving the PDF document* 319

16.2	Building a PDF report generator	320
	<i>Our report document model</i>	320
	▪ <i>Our report page model</i>	321
16.3	Drawing text on the page	322
	<i>Choosing fonts</i>	322
	▪ <i>Setting the font and adding text</i>	323
	▪ <i>Adding wrapped text</i>	324
16.4	Working with color	324
	<i>Choosing colors</i>	325
	▪ <i>Setting colors</i>	325
16.5	Using Styles	326
16.6	Drawing shapes	326
	<i>Drawing lines</i>	327
	▪ <i>Setting up dashed lines</i>	327
	<i>Drawing rectangles and polygons</i>	328
	▪ <i>Drawing circles and ellipses</i>	331
16.7	Rotating objects	332
16.8	Adding images to the page	332
16.9	Drawing objects within clipping masks	333
16.10	Generating PDF reports	333
16.11	Summary	335
<i>appendix A</i>	<i>A whistle-stop tour of PHP syntax</i>	337
<i>appendix B</i>	<i>Object-Oriented PHP</i>	350
<i>appendix C</i>	<i>Tips and tricks</i>	367
	<i>index</i>	381