

Practical CakePHP Projects



Kai Chan and John Omokore
with Richard K. Miller

Practical CakePHP Projects

Copyright © 2009 by Kai Chan and John Omokore with Richard K. Miller

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-4302-1578-3

ISBN-13 (electronic): 978-1-4302-1579-0

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Java™ and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc., in the US and other countries. Apress, Inc., is not affiliated with Sun Microsystems, Inc., and this book was written without endorsement from Sun Microsystems, Inc.

Lead Editor: Steve Anglin

Technical Reviewer: David Golding

Editorial Board: Clay Andres, Steve Anglin, Mark Beckner, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Michelle Lowman, Matthew Moodie, Jeffrey Pepper, Frank Pohlmann, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh

Project Manager: Richard Dal Porto

Copy Editor: Marilyn Smith

Associate Production Director: Kari Brooks-Copony

Production Editor: Candace English

Compositor: Patrick Cunningham

Proofreader: Martha Whitt

Indexer: Brenda Miller

Artist: April Milne

Cover Designer: Kurt Krames

Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail orders-ny@springer-sbm.com, or visit <http://www.springeronline.com>.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail info@apress.com, or visit <http://www.apress.com>.

Apress and friends of ED books may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Special Bulk Sales—eBook Licensing web page at <http://www.apress.com/info/bulksales>.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at <http://www.apress.com>.

For Rita
—*Kai Chan*

For Comfort
—*John Omokore*

For Marian
—*Richard K. Miller*

Contents at a Glance

About the Authors	xiii
About the Technical Reviewer	xiv
Acknowledgments	xv
Introduction	xvii
■ CHAPTER 1 Cake Fundamentals	1
■ CHAPTER 2 Blogging	29
■ CHAPTER 3 E-Commerce	47
■ CHAPTER 4 A Message Forum Web Service	85
■ CHAPTER 5 Google Maps and the Traveling Salesman	131
■ CHAPTER 6 Mashing Twitter with the Google Translator	173
■ CHAPTER 7 Unit Testing and Web Testing	213
■ CHAPTER 8 A Cake Control Panel	237
■ CHAPTER 9 Translating Stories	271
■ CHAPTER 10 Adding Automagic Fields	307
■ CHAPTER 11 Cake Tags	317
■ CHAPTER 12 Dynamic Data Fields	329
■ CHAPTER 13 Captcha	359
■ INDEX	371

Contents

About the Authors	xiii
About the Technical Reviewer	xiv
Acknowledgments	xv
Introduction	xvii

CHAPTER 1	Cake Fundamentals	1
	Cake Features	1
	The Ingredients of Cake	2
	The Model-View-Controller Design Pattern	2
	Rapid Application Development	2
	PHP 4+	3
	Object-Oriented Programming	3
	Dissecting Cake	5
	Cake's Directory Structure	5
	The Cake Naming Conventions	6
	Models	8
	Model Creation	8
	Data Validation	13
	Views	15
	Controllers	18
	Cake Components	19
	Helpers	21
	Plugins	22
	Vendors	25
	Summary	27

CHAPTER 2	Blogging	29
	Creating the Database	29
	Reviewing the Application Structure	31
	Creating the Post Model	31

Creating the Posts Controller	32
Listing the Posts	32
Adding a Post	35
Updating a Post	38
Unpublishing a Post	40
Publishing a Post	41
Deleting a Post	42
Creating an RSS Feed	43
Summary	46

CHAPTER 3 **E-Commerce** 47

The Online Shop Layout	47
Two Site Layouts	48
Layout of the Main Content	48
The User Journey	49
Setting Up the Shop Database	50
Interacting with the Online Shop Database	52
The Category Model	53
The Categories Controller	56
The Product Model	59
The Products Controller	59
The Cart Model	63
Handling User Requests	67
The ApplicationController Class	67
The Home Page	69
The Carts Controller	69
The Order Model	75
The Google Checkout Button	79
The PayPal Submit Button	82
Summary	83

CHAPTER 4 **A Message Forum Web Service** 85

Our Take on Web Services	85
Web Service Elements	85
REST and HTTP	87
Result Return Formats	88
Application Requirements	88
Threads and Posts	89
Web Service Requests	90
Layout	91

Application Structure	91
JSON Web Services	92
Our Application Controller	94
Fetch a Message	97
Fetch Several Messages	100
Fetch the Threads	105
Post Messages	110
Process a Message	116
Process a Search Request	120
Writing the API Documentation	125
Summary	129

■ CHAPTER 5 **Google Maps and the Traveling Salesman** 131

Hello Map!	131
Google Maps Explained	133
Geocoding	133
Google Map Events	134
Map Interface Elements	134
Overlays	134
Driving Directions	135
Application Requirements	136
Application Structure	138
Cake Models	140
The Interface	141
The Global Layout	141
Home Page	143
Travel Mappr Manager	146
Finding Locations	148
The Traveling Salesman Algorithm	150
Plotting the Journey	156
Journey Data	160
Saving a Journey	160
Saving Tags	167
Retrieving and Editing a Journey	168
Viewing a Journey	169
Summary	172

CHAPTER 6	Mashing Twitter with the Google Translator	173
	The Twitter API	173
	The Google Ajax Language API	175
	Application Requirements	176
	Application Structure	178
	Cake Models	181
	Internationalization and Localization	184
	Web Services	186
	Caching	188
	Caching Views	189
	Caching Models	189
	Caching Twitter and Google Translations	189
	Caching and the Application Layout	190
	Changing Languages	190
	Changing Locales	191
	The Controllers	192
	The Twittertwister Controller	192
	The TwitterRequest Controller	194
	The TwitterStatus Controller	196
	The ApplicationController	209
	Summary	211
CHAPTER 7	Unit Testing and Web Testing	213
	Getting Programming Done	213
	Our Case Study: An App Like In/Out	214
	Creating the Application	214
	Adding Username Validation	219
	Using Cake's Unit Testing Framework	221
	Installing SimpleTest	222
	Creating Your Own Unit Tests	224
	Using Assert Methods	231
	Testing the Entire MVC System	232
	Web Testing	232
	Creating Web Tests	232
	Web Testing Any Application	234
	Test-Driven Development	235
	Summary	235

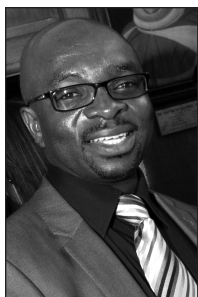
CHAPTER 8	A Cake Control Panel	237
	Application Requirements	237
	The Authentication and ACL Components	238
	The Authentication Component	238
	The Access Control List Component	238
	Component Setup	241
	Control Panel Application Controllers	242
	The Control Panel Controller	242
	The Actions Controller	244
	The Groups Controller	251
	The Users Controller	259
	Testing the Control Panel	266
	Summary	270
 CHAPTER 9	 Translating Stories	 271
	Application Structure	271
	The Translate Behavior	272
	Stories	273
	Baking Cake	273
	Adding Stories	277
	Administering Stories	279
	Translating Stories	279
	Viewing Stories	282
	Deleting Stories	284
	Listing Stories	284
	Translation Pagination	288
	Locale and Language Selection	292
	Setting Locale by Browser	292
	Setting Locale by Language Code	293
	Setting Locale by Hand	293
	User Authentication	293
	Logging In	294
	Logging Out	296
	Baked Code	297
	Summary	306

CHAPTER 10	Adding Automagic Fields	307
	Cake's Built-in Magic Fields	307
	Writing a Custom Behavior	308
	Building Custom Magic Fields	309
	Access Data Field	310
	Record Order Data Field	311
	Locking Data Field	313
	Summary	315
CHAPTER 11	Cake Tags	317
	Content and Data Separation	317
	View Template	319
	Cake View Class Extension	321
	Cake Plugins	323
	Yahoo! Maps	325
	Summary	328
CHAPTER 12	Dynamic Data Fields	329
	Traditional Product Searching	329
	The Dynamic Data Approach	330
	Considerations for Using the Dynamic Data Approach	331
	The Product Database Design	332
	Baking for This Application	337
	Building the Product Search Feature	338
	Creating the Product Search Form	338
	Processing the Search	347
	Adding a Product	353
	Creating Table Entries	353
	Entering Product Data	354
	Summary	358
CHAPTER 13	Captcha	359
	Captcha Implementations	359
	Captcha Types	360
	ASCII Art Captcha	361
	A Captcha Component	362
	The ASCII Art Component Class	362
	The Captcha Controller	365
	Summary	369
INDEX		371

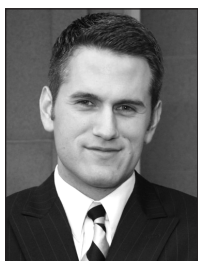
About the Authors



■ **KAI CHAN** started his computing career in the late 1980s. His current interests include programming methodology, the Semantic Web, data visualization, and enterprise systems. Kai holds a Computer Science bachelor's degree and a master's degree in Computer Graphics. He is a cofounder of the Azzian MVC CMS framework. Together with John Omokore and others, he runs a software and training company in London, specializing in various large-scale projects, from SAP to e-commerce web sites. When he has a spare moment, he likes tennis, squash, and long-distance running.



■ **JOHN OMOKORE** is a developer, technical consultant, writer, and trainer. John has programming experience in many technologies, including Linux, PHP, MySQL, and Ajax. He has worked on market research data analysis, database development, and related systems. He received his bachelor's degree in Mathematics and is pursuing a postgraduate degree in software engineering at Oxford University in England. John provides consulting and web development services to corporate organizations around the world. He's a cofounder of AlternativeBrains and the Azzian MVC CMS framework and sits on the board of many companies. John lives outside London with his wife, two children, and some animals. His career interests include open source scripting languages, OOP programming, and the use of SAP in large-scale industries (chiefly oil and gas). When not scripting, he enjoys playing chess and squash, visiting the gym, and a bit of socializing.



■ **RICHARD K. MILLER** graduated from Brigham Young University with a degree in Business Management but has been interested in technology since he began computer programming at age 10. His experience includes web programming, Internet marketing, and new media strategies. He is the developer of several MediaWiki extensions and WordPress plugins, including the widely used What Would Seth Godin Do plugin.

About the Technical Reviewer



■ **DAVID GOLDING** began developing web sites in 1999 and first started using CakePHP on a bet he couldn't complete a web application in five minutes. He is the author of *Beginning CakePHP: From Novice to Professional* (Apress, 2008) and has taught CakePHP even while it was still in early stages of development. David has a degree in European Studies from Brigham Young University and continues work in religious studies and history as a graduate student at Claremont Graduate University. He lives with his wife, Camille, and his son, Kenny, in Southern California.

Acknowledgments

When we first decided to write this book, we really didn't think it would be that difficult a task. After all, we've been coding and writing documentation for years and years. Now having written the book, we can honestly say it has been one of the hardest projects we've done since we wrote our first-ever Hello World program. As such, with tears streaming from our eyes, we would wholeheartedly like to thank all the people involved. It all sounds like a cliché, but it's all true. Thank you to the team at Apress, the Cake Software Foundation, colleagues, friends, families, and neighbors. In no particular order, we would like to thank them individually. They are Steve Anglin, Richard Dal Porto, Matt Wade, Marilyn Smith, Joohn Choe, David Golding, Nancy Wright, Richard K. Miller, Rita Woo, Terry Wells, Dan Jackson, Candace English, and God.

Kai Chan and John Omokore

Thank you to Kai Chan and John Omokore for allowing me to take part in this book. I've enjoyed working with them and the entire Apress team. Thanks to David Golding for getting me involved. I'm thankful for good parents, family, friends, and colleagues, and to God.

Richard K. Miller

Introduction

First off, thank you for picking up this book. Whether you are standing in a bookshop or reading this at home, we assume you probably have a strong interest in developing web sites. In the past few years, the number of web site frameworks has increased dramatically. This is especially true for PHP-based frameworks. Many people have chosen to adopt CakePHP (Cake, for short) for various reasons, such as these:

- PHP programmers are widely available. Most projects have tight deadlines, and you want team members who can quickly pick up a new piece of technology.
- CakePHP is easy to learn. You want a powerful tool that you can easily master.
- CakePHP has good support. Developers frequently post and reply to messages on the Cake forum. And there are always some good discussions happening on the Cake IRC. (To see for yourself, simply download mIRC from <http://www.mirc.com/>, connect to the server `irc.freenode.net`, and join the `#cakephp` channel.)

When you are developing a site using Cake, you often find yourself trawling through tutorials online to see how things are done. We've done that ourselves many times. However, despite the power of the Internet, we still like to look through books. And we think you will find this book a great help in your Cake development endeavors, in addition to all of the material available online.

Most of the applications in this book have been written as a result of some real-world development we have done in the past. We focus on projects that we think are relevant to the future of web development.

Let's take mashups, for example. We should all take an interest in this ever-expanding area of web development. We can honestly say that any successful online web site in the future will need to easily communicate with other applications. Application designers will need to bear this in mind. Matters such as search engine optimization need to be built into the application itself. Cake allows us to think in terms of the high-level architecture instead of the nuts and bolts of a web application.

Who Should Read This Book

Practical CakePHP is a book mainly for developers. To get the most from it, you should be comfortable with a number of web technologies and programming concepts. These include PHP, SQL, HTML, JavaScript, object-oriented programming, and design patterns, as well as the general principles of web development. If you are at the forefront of web development, then this book is for you!

If our book sounds a little too advanced for you, we recommend that you do some preliminary reading. We suggest the following books:

- *Beginning PHP and MySQL: From Novice to Professional, Third Edition*, by W. Jason Gilmore (Apress, 2008)
- *Beginning CakePHP: From Novice to Professional* by David Golding (Apress, 2008)

How This Book Is Organized

Each chapter in this book has been chosen so it will cover the core features in Cake, plus some of the minor features as well. The following is a rough breakdown of what each chapter includes.

- Chapter 1, “Cake Fundamentals,” gives you an introduction to Cake. If you are new to the CakePHP framework, this is the place to start.
- Chapter 2, “Blogging,” provides you with a simple blogging application. It’s perfect for beginners who want to know what a Cake application looks like. If there are two chapters in the book that need to be read in sequence, they are Chapters 1 and 2.
- Chapter 3, “E-Commerce,” gives you greater insight into the way Cake is used in a common application. We walk through implementing an online shop using the Cake framework.
- Chapter 4, “A Message Forum Web Service,” covers the development of a web services API. We guide you through creating a clean API, so any third party can access your application using standard protocols.
- Chapter 5, “Google Maps and the Traveling Salesman,” shows you how the Google Maps API is used with Cake. One of the main features of this chapter’s application relates to the classic traveling salesman problem: a salesman needs to visit a number of cities only once and return to where he started.
- Chapter 6, “Mashing Twitter with the Google Translator,” emphasizes the importance of web services in modern web application development. In true Web 2.0 and Cake fashion, this chapter’s application mashes the Google Ajax Language API with the Twitter API to provide automatic translation of Twitter messages.
- Chapter 7, “Unit Testing and Web Testing,” covers one of the hottest topics among web professionals. Cake 1.2 devotes a large section to testing, and this chapter shows you how to take advantage of Cake’s integrated unit testing features.
- Chapter 8, “A Cake Control Panel,” covers Cake’s access control lists and security features. We develop a web-based front end that allows administrators to manage user security.

- Chapter 9, “Translating Stories,” provides you with the knowledge to tackle Cake’s internationalization and localization features. We develop an application in which news stories are available in other languages, with an administration area where translators can translate stories from a base language to another language.
- Chapter 10, “Adding Automagic Fields,” demonstrates extending Cake’s use of automagic fields like `created`, `modified`, and `title`. We create three new automagic fields.
- Chapter 11, “Cake Tags,” shows you our take on an established technology where XML tags are used as a wrapper to coding logic. Using Cake, we develop our own HTML-based tags to display two Yahoo maps.
- Chapter 12, “Dynamic Data Fields,” extends the e-commerce chapter with a special product-filtering technique. We take a dynamic data approach to product searches.
- Chapter 13, “Captcha,” shows how ASCII Art can be used as a Captcha test. In this chapter’s project, the Captcha test is housed in a Cake component so it can be used by other applications.

How to Contact the Authors

The authors can be contacted as follows:

- Kai Chan can be contacted at kai.chan@edgeable.com.
- John Omokore can be contacted at john@omokore.com.
- Richard K. Miller can be contacted at richard@richardkmiller.com.

