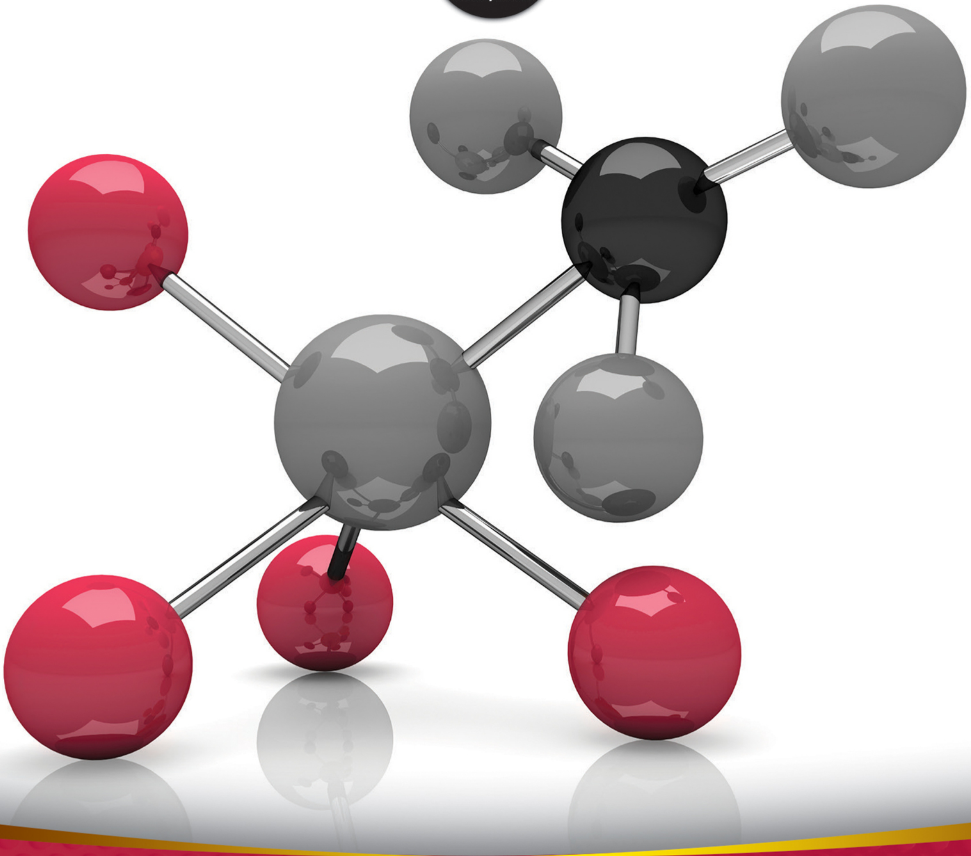

Professional

Java® EE Design Patterns

Murat Yener, Alex Theedom

PROFESSIONAL JAVA® EE DESIGN PATTERNS

PROFESSIONAL

Java® EE Design Patterns

PROFESSIONAL

Java® EE Design Patterns

Murat Yener
Alex Theedom

wrox™
A Wiley Brand

Professional Java® EE Design Patterns

Published by
John Wiley & Sons, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

Published simultaneously in Canada

ISBN: 978-1-118-84341-3
ISBN: 978-1-118-84358-1 (ebk)
ISBN: 978-1-118-84345-1 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Library of Congress Control Number: 2014946684

To Nilay and all my family (Semra and Musfata Yener), for all your support and time I needed to write this book.

—Murat

To Mariu, for all your support and encouragement.

—Alex

ABOUT THE AUTHORS

MURAT YENER is a code geek and open source committer, working at Intel New Devices Group as an Android developer. He has extensive experience with developing Java, web frameworks, JavaEE, and OSGi applications, in addition to teaching courses and mentoring. Murat is an Eclipse committer and one of the initial committers of the Eclipse Libra project, he is currently working on building native and Hybrid mobile apps with HTML5 and mGWT.

Murat has been a user group leader at GDG Istanbul since 2009, organizing, participating, and speaking at events. He is also a regular speaker at JavaOne, EclipseCon, and Devoxx conferences.

Linkedin—`www.linkedin.com/in/muratyener`

twitter—`@yenerm`

blog—`www.devchronicles.com`

ALEX THEEDOM is a Senior Java Developer at Indigo Code Collective `indigocodecollective.com` (part of the E-scape Group) where he played a pivotal role in the architectural design and development of a microservice based, custom built lottery and instant win game platform.

Prior to that, he developed ATM software for an international Spanish bank and code quality analysis software for a software consultancy.

Alex is experienced with Java web application development in a diverse range of fields including finance, e-learning, lottery and software development. His passion for development has taken him to projects throughout Europe and beyond. He is a blogger at `alextheedom.com` and can be found helping fellow problem solvers in online forums.

Linkedin—`www.linkedin.com/in/alextheedom`

Twitter—`@alextheedom`

Blog—`www.alextheedom.com`

ABOUT THE TECHNICAL EDITOR

MOHAMED SANAULLA is a Software Developer with over five years of professional experience developing software. He is currently working for India's largest e-Commerce establishment and is also a moderator on the JavaRanch Forums. When he is not working on his PC, he is busy tending to his cute little daughter. He shares his experiments and thoughts on software development at `http://blog.sanaulla.info`.

CREDITS

ASSOCIATE PUBLISHER
Jim Minatel

PROJECT EDITOR
Adaobi Obi Tulton

TECHNICAL EDITOR
Mohamed Sanaulla

PRODUCTION MANAGER
Kathleen Wisor

COPY EDITOR
Karen A. Gill

MANAGER OF CONTENT DEVELOPMENT AND ASSEMBLY
Mary Beth Wakefield

MARKETING DIRECTOR
David Mayhew

MARKETING MANAGER
Carrie Sherrill

PROFESSIONAL TECHNOLOGY AND STRATEGY DIRECTOR
Barry Pruett

BUSINESS MANAGER
Amy Knies

PROJECT COORDINATOR, COVER
Patrick Redmond

PROOFREADER
Nancy Carrasco

INDEXER
John Sleeva

COVER DESIGNER
Wiley

COVER IMAGE
© iStock.com/pavlen

BACKGROUND
© PhotoAlto Images/Fotosearch

ACKNOWLEDGMENTS

AS MY COAUTHOR Alex always says, we wanted to write a book we would like to own and read for ourselves. To begin with, I want to thank Alex for all his patience, hard work, and great knowledge. Without him, this book wouldn't be nearly as good.

I am grateful to Mary James, our former acquisitions editor, who contacted me about writing a book on Spring but listened to my ideas that formed the basis of this book. Without her support and guidance, this book wouldn't have become a reality. No words would be enough to thank Adaobi Obi Tulton, who patiently worked on all the details while keeping most of the schedule stresses away from us. And thanks, of course, to everyone at Wrox/Wiley who got this book on the shelves. Thanks, also, to Reza Rahman for all his encouragement.

I must thank three important people who had a huge impact on where I am in my professional life in terms of software.

First, thanks to my dad, Mustafa Yener, for buying me my first computer, a C64, at an early age while I was asking for slot cars. That computer is where I wrote my very first codes.

Second, thanks to my thesis advisor, Prof. Mahir Vardar, whom I owe all the early guidance I needed to start my career.

Finally, thanks to my life-time mentor and friend (also my ex-boss) Naci Dai, who taught me almost anything I know about being a professional software developer.

—Murat

WE ARE VERY PROUD of this, our first book, and hope that you will get as much from reading it as we have writing it. We approached writing this with the perspective that it should be the kind of book we would buy if we hadn't written it. We have achieved that.

However, this book would not have been possible without the dedication, patience, and understanding of the many others who have contributed directly and indirectly to its creation. We would like to acknowledge the contributions made by the dedicated and experienced team at Wiley Publishing. They have stuck with us through thick and thin and believed that it was all possible. We would like to give special thanks to Mary James, our acquisitions editor, whose support made this book a reality. Thanks also to Adaobi Obi Tulton, whose patience and gentle nudges kept us on our toes and whose attention to detail saved us from tripping over ourselves. I would like to thank my coauthor, Murat Yener, for his inspiration and sense of humor that makes this book unique; and finally, but not least, I would like to thank my wife, Maria Eugenia García García, for her total support and understanding while writing this book. Thank you.

—Alex

CONTENTS

FOREWORD

Ignorant men raise questions that wise men answered a thousand years ago

—Johann Wolfgang von Goethe

Design patterns are our link to the past and the future. They make up a foundational language that represents well understood solutions to common problems that talented engineers before us have added to our collective knowledge base. Design patterns or blueprints exist in every engineering field in one way or another. Software development is no different. Indeed, design patterns are probably our most tangible link to engineering rather than the more organic and less regimented world of the artisan or craftsman. The art and science of design patterns was brought to the world of software engineering—and more specifically to enterprise Java—by the seminal Gang of Four (GoF) book. They have been with us ever since through our adventures in J2EE, Spring, and now modern lightweight Java EE. This is for very good reasons. Server-side Java developers tend to write the type of mission critical applications that need to stand the test of time and hence benefit the most from the discipline that design patterns represent.

It really takes a special kind of person to write a book on design patterns, let alone a book on how to utilize design patterns in Java EE applications. You require not only basic knowledge of APIs and the patterns themselves, but deep insight that can only come with hard-earned experience, as well as an innate ability to explain complex concepts elegantly. I am glad Java EE now has Murat and Alex to accomplish the mighty feat.

This book fulfills a much needed gap and fills it well. It is also very good that the book is on the cutting edge and covers Java EE 7 and not just Java EE 6 or Java EE 5. In fact many of the design patterns covered, like Singleton, Factory, Model-View-Controller (MVC), Decorator, and Observer, are now incorporated right into the Java EE platform. Others like Facade, Data Access Object (DAO), and Data Transfer Object (DTO) fit elegantly on top. Murat and Alex tackle each pattern, explain its pragmatic motivation, and discuss how it fits into Java EE.

It is an honor and a privilege to write a small opening part of this very important book that I hope will become a very useful part of every good Java EE developer's bookshelf. I hope you enjoy the book, and that it helps you write better, more satisfying enterprise Java applications.

M. Reza Rahman
Java EE/GlassFish Evangelist
Oracle Corporation

INTRODUCTION

THIS BOOK DISCUSSES THE CLASSIC DESIGN PATTERNS that were first mentioned in the famous book by the GoF[1] and updates them specifically for Java EE 6 and 7.

In every chapter we describe the traditional implementation of each pattern and then show how to implement it using Java EE-specific semantics.

We use full code examples to demonstrate both the traditional and Java EE implementations and color each chapter with real-life stories that show the use (or misuse) of the pattern.

We investigate the pros and cons of each pattern and examine their usages. Each chapter finishes with some exercises that challenge your understanding of the pattern in Java EE.

WHO THIS BOOK IS FOR

This book is for everyone with any level of experience. It covers almost everything about a pattern, from how it is referred to in other books, to code on basic Java implementation, to Java EE implementation, and finally real life examples of how and when to use a specific pattern. It also has real life war stories that talk about good and bad practices.

Having some basic knowledge of design patterns and Java EE will aid you as you read this book.

If you are already experienced with patterns and basic Java implementations, you may prefer to jump into Java EE implementations. Refreshing your memory and knowledge of design patterns could prove helpful.

WHAT THIS BOOK COVERS

This book covers all classical design patterns that Java EE offers as part of standard implementation, besides some new patterns. The coverage goes back to Java EE5 and is up to date for the latest version available, which is Java EE 7.

We hope this book will be a reference you will keep on your shelf for a long time.

HOW THIS BOOK IS STRUCTURED

Each chapter focuses on a design pattern. If the pattern is classical, a simple Java implementation is given after the explanation of the pattern. Each chapter offers war stories telling a good or bad real life example about the pattern focused on/in the chapter. The war story is followed by a Java EE implementation, example, and explanation. Each code sample given can be run by itself. Finally, each chapter ends with when and how to use the pattern effectively.