

Extracted from:

Design It!

From Programmer to Software Architect

This PDF file contains pages extracted from *Design It!*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragprog.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printed versions; the content is otherwise identical.

Copyright © 2017 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The Pragmatic Bookshelf

Raleigh, North Carolina

The
Pragmatic
Programmers

Design It!

From Programmer
to Software Architect



Michael Keeling
edited by Susannah Pflazer

Design It!

From Programmer to Software Architect

Michael Keeling

The Pragmatic Bookshelf

Raleigh, North Carolina



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf, PragProg and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic books, screencasts, and audio books can help you and your team create better software and have more fun. Visit us at <https://pragprog.com>.

The team that produced this book includes:

Publisher: Andy Hunt

VP of Operations: Janet Furlow

Development Editor: Susannah Davidson Pfalzer

Indexing: Potomac Indexing, LLC

Copy Editor: Liz Welch

Layout: Gilson Graphics

For sales, volume licensing, and support, please contact support@pragprog.com.

For international rights, please contact rights@pragprog.com.

Copyright © 2017 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Printed in the United States of America.

ISBN-13: 978-1-68050-209-1

Encoded using the finest acid-free high-entropy binary digits.

Book version: P1.0—October 2017

Welcome!

Software architecture is the foundation on which awesome software is built. A great architecture alone isn't enough to guarantee your software will be a smashing success, but the wrong architecture almost guarantees failure. Software architecture is so important that every software developer should know how to design it.

In this book, you'll learn how to design great software architectures. Just to be clear, this isn't a lesson in Ivory Tower, high-abstraction software design. You also won't find any *magic bean* solutions—frameworks and technologies that magically solve any problem. You will learn how to apply essential design principles and practices, which will make you a stronger programmer, architect, and technical leader.

Designing great software requires more than mastery of principles and practices. How you go about designing a software system is just as important as the final result. In this book, you'll learn how to use design thinking and human-centered methods to design software architectures collaboratively with your team. This approach to architecture design helps you forge a stronger connection between the design decisions you make and the humans affected by those decisions. Putting people first allows you to make better design decisions and, as a result, better software.

Who Should Read This Book?

This book is for anyone who has ever stood at a whiteboard and sketched boxes and lines while trying to answer tough questions.

If you're completely new to software architecture design, then this book is the perfect introduction. We'll start with the basics and work our way through the core fundamentals you need to know to be an amazing software architect.

If you're a programmer who already knows a thing or two about architecture, then this book will help you organize your thoughts. As you read, you may find concepts you intuited on your own but didn't know the name, or perhaps

you'll find gaps you didn't know you were missing. After reading this book, you'll be able to explain why you do what you do, which puts you in a better position to lead others.

If you're already a software architect and this isn't your first rodeo, then this book will give you a fresh perspective on how to lead your team. The junior programmers of today expect to have a greater say in the software they build. The focus on fundamentals in this book will prepare you to teach and mentor today's programmers—the architects of tomorrow—so they can fully participate in the design process. The collaborative design approaches described throughout this book will give you new techniques for safely and productively collaborating with less experienced teammates as you design a software system together.

How to Read This Book

This book is divided into three parts. Parts I and II are designed to be read start to finish. Part III is designed for easy reference.

In Part I you'll learn the basics of software architecture and design thinking needed to become an architect.

In Part II you'll learn the essential skills and knowledge all software architects possess.

Part III includes a set of practical architecture design methods. There are no silver bullets, but every software engineer has a silver toolbox filled with practices, methods, and techniques that together allow them to ship amazing software. The methods in Part III come from my silver toolbox and it is my privilege to share them with you.

Each chapter in Parts II and III focuses on a different design mindset, which you'll learn more about in [Chapter 2, *Design Thinking Fundamentals*, on page ?](#). Design mindsets are a way of thinking about the world to help us focus our attention on the right details at the right time. There are four design mindsets: understand, explore, make, and evaluate. Look for the icon at the start of each chapter to tell you which mindset you'll be learning about next.

Community Tips and Advice

When you opened this book, you joined a community of software architects who help one another by sharing advice, tips, and good practices. To officially welcome you to the community, I've asked some fellow software architects to share tips and advice they think you should know. You'll find their stories and advice in sidebars throughout the book.

Our extraordinary community contributors are Len Bass, Bett Bollhoefer, Simon Brown, George Fairbanks, Thijmen de Gooijer, Patrick Kua, and Ipek Ozkaya. You can learn more about them in [Appendix 1, *Community Contributor Bios*, on page ?](#).

Case Study

When talking about abstract things, it's all too easy to stay abstract. To prevent that from happening I've included a case study—Project Lionheart—based on real systems I've worked on in the past. The case study is introduced in Chapter 1. You'll see examples from the case study as the book progresses.



Get Your Hands Dirty Exercises

Great software architects have dirty hands. To become a great software architect, you have to practice design, not just talk about it. Anytime you see this icon it's time to think critically and put the theory into practice. Like architecture design in the real world, *Get Your Hands Dirty* exercises have many right answers. How you arrive at an answer—the journey—is as important as the solution itself.

Online Resources

This book has its own web page¹ where you can find details about this book, post to the discussion forums, and report errata such as typos and content suggestions. The discussion forums are the perfect place to talk shop with other readers and share your answers to the exercises.

Welcome, thank you for joining me, and let's get started!

1. <https://pragprog.com/book/mkdsa/design-it>