

ANNA P. MURRAY

THE COMPLETE
**SOFTWARE
PROJECT
MANAGER**

MASTERING TECHNOLOGY
FROM PLANNING TO
LAUNCH AND BEYOND

WILEY

THE COMPLETE SOFTWARE PROJECT MANAGER

MASTERING TECHNOLOGY FROM
PLANNING TO LAUNCH AND BEYOND

Anna P. Murray

WILEY

Copyright © 2016 by Anna P. Murray. All rights reserved.

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.

Published simultaneously in Canada.

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, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600, or on the Web at www.copyright.com. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Cataloging-in-Publication Data

Names: Murray, Anna, 1966- author.

Title: The complete software project manager : mastering technology from planning to launch and beyond / Anna P. Murray.

Description: Hoboken, New Jersey : John Wiley & Sons, Inc., [2016] | Includes index.

Identifiers: LCCN 2015036771 (print) | LCCN 2015040806 (ebook) | ISBN 9781119161837 (cloth) | ISBN 9781119219910 (ePDF) | ISBN 9781119219903 (ePub)

Subjects: LCSH: Computer software—Development. | Software engineering—Management.

Classification: LCC QA76.76.D47 M877 2016 (print) | LCC QA76.76.D47 (ebook) | DDC 005.1—dc23

LC record available at <http://lccn.loc.gov/2015036771>

Cover Design: Wiley

Cover Image: © shuoshu / Getty Images

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ABOUT THE AUTHOR

Anna P. Murray is a nationally recognized technology consultant and the founder of emedia LLC, a certified women-owned business. She has consulted on and run large-scale software projects for Kellogg, Harvard Business Publishing, Time Out New York, Slate Group, The American Association of Advertising Agencies, National Cancer Institutes, *New York* magazine, and many others.

Murray is a double winner of the Stevie Award for Women in Business, a recipient of a Mobile Marketing Association award for mobile app development, several Kellogg agency partner awards, and Folio's Top Women in Media Award. She has served as President and Vice President of the International Women's Writing Guild.

One of a rare species—a woman who owns a successful software-development company—she loves to combine her two passions: technology and writing. Anna writes on technology from a variety of vantage points, including its humorous impact on daily life, its serious business applications, and its role in changing the way people relate to one another.

After spending several years as a teacher and journalist, Murray founded emedia, one of the earliest web-development firms, in 1996. She holds a bachelor's degree from Yale and a master's in journalism from Columbia.

INTRODUCTION

The history of software development teaches us that between 30 and 40 percent of all software projects fail. A good majority of those are canceled completely and never see the light of day. It seems everyone has a different prescription for how to avoid this fate. There are shelves of books on how to improve the software development process—books on Agile methodology, Scrum, the Waterfall method, rapid application development, extreme programming, top-down versus bottom-up, and even something called the chaos model.

When a business is faced with a software development project, people don't have the time to become experts in software management and development theory. What you need is not theory, but rather a practical, hands-on guide to managing this complex process, written in a language you can understand.

Some undertakings—be it having kids, climbing Mount Everest, or, yes, software development—are just difficult. And sometimes the most helpful thing is to hear from someone who has been there before and walked the path. That person can tell you what you're going to encounter and what difficulties to expect. An experienced guide may not be able to take away the challenges, but she can prepare you for them.

About a year ago, I was working on a large implementation project. About halfway through, a young project manager came to me and said, "Anna, do you have a crystal ball? Because at the beginning of this project you sat us all down and told us what to expect. You gave us a list of many things and every single one of them has happened—and happened on schedule. Do you have that written down anywhere?"

Now I have it written down somewhere.

Software Project Management

This book is specifically about software project management. There are many other types of project management. Project management is needed for digging oil wells, building skyscrapers, and launching rockets.

Examples of software projects include

- Launching websites
- Installing a CRM (customer relationship management) tool
- Implementing a new accounting package
- Building a custom application for your business

Software is truly “its own animal.” People tasked with managing software projects need processes and advice just for them.

A Holistic Approach

Many books on project management focus on what I call the “literal project manager,” the person whose title is “project manager” on his business card. He or she may have a specific certification in the discipline of project management. In Chapter 5, we’ll talk in depth about project teams and team members. For now, it’s important to understand that the literal project manager is not the only one with project-management responsibilities. People might even be surprised learn how narrowly focused the project manager’s job often is (Figure I.1).

Many books on project management concentrate only on the literal project manager. The purpose of many of these books is to help the project manager pass his or her certification tests, such as the PMP exam. That’s great, if you are facing the exam. Unfortunately, this narrow focus leaves much out, not only for the literal project manager but for everyone else.

True project management extends to many people and to the entire business. There is the project sponsor, the program manager, the project manager, and programming teams as well as stakeholders in the broader business. Therefore, this book takes a holistic view of project management. It can be read by businesspeople, programmers, project managers, program managers, and CEOs—anyone who is involved in or affected by a software project (Figure I.2). Of course, it has critical information for people whose job titles are project manager and program manager.

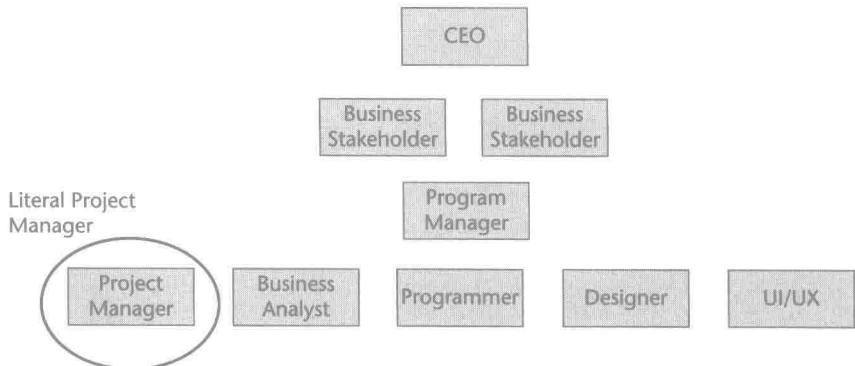


Figure I.1

FOREWORD

A generation ago, Internet pioneer Carl Malamud took issue with people who complained about bulky computers, byzantine interfaces, and buggy software: “A lot of this ‘too hard to use’ stuff will go away. Radio was so messy for the first 20 years, it wasn’t funny. Cars, ditto—you had to be a mechanic to drive one.” Computers, he prophesied, would one day be as easy to use and reliable as automobiles had become, and the electronic frontier would become as simple to navigate as an Interstate with a road atlas.

That day came, of course—including Google Maps to replace the road atlas—but it brought along a paradox. The simplicity, power, and awesomeness that users and executives take for granted disguise the very technical, messy, and difficult work that happens behind the scenes. Gremlins lurk under the lid of your laptop; krakens crouch behind the scrim of your Cloud. A lot of what seems to gleam is the shimmer of WD-40 on duct tape.

Amateurs cheerfully rely on smoothly running software for business and pleasure, but the work of IT is no place for an amateur. Creating and designing systems, maintaining them, and running projects to install, fix, rip out and replace, upgrade, and integrate them—these are complicated tasks. They rarely go according to plan, because they invariably run into some obstacle that could not possibly be foreseen. Even professionals in the industry can be seduced by a project’s progress into thinking, “No problem,” when they should be thinking, “Oh, this is hairy.” And, of course, finger-pointing is the consequence of problems, with IT blaming the business, the business blaming IT, and everybody blaming the consultants and software companies.

Anna Murray’s book is an extraordinary accomplishment: It speaks as clearly to the IT professional as it does to executive civilians like me. (I’m an English major with a Mac.) For the professional, it is an experienced, no-nonsense guide: the mentor you wish you had. Anna tells you how to put an IT project into the strategic framework that is right for your organization, how to scope a project, how to communicate and plan with stakeholders; she tells you how to assemble your tools and team; she explains how to find, select, and manage vendors; and she tells you how to cook and eat crow when you need to—as you almost certainly will.

The strategic point is particularly valuable. IT long ago migrated from the glass house to the desktop, but it is now part of every nerve and sinew of an organization: It is in the skin that touches customers, in the brain that analyzes performance, in the heart that pumps resources where they are needed. Technologists don’t need to be strategists, but they can no longer be strategically

ignorant. Anna shows how to connect strategy and IT, how to ask the right questions, and how to frame the trade-offs and decisions that complex IT projects require in a way that business leaders can understand.

This is, at the same time, the book IT managers should give to their nontechnologist colleagues, bosses, and internal customers. Her distinction among IT projects that are simple, complicated, and complex is worth the price of admission by itself. I can think back to half a dozen cases where the business-IT relationship got into trouble because the nontechnologists did not know or could not understand what their IT colleagues were up against. Any executive who reads this book will ask better questions, get better answers, and have a better understanding of the answers.

Thomas A. Stewart
Executive Director, National Center for the Middle Market
Fisher College of Business
The Ohio State University

ACKNOWLEDGMENTS

This book grew out of a two-decades-long conversation with my colleagues and with the many clients we are privileged to serve. Starting back in the Windows 3.1 days, we have grappled with the complexities of technology and survived the numerous inevitable crises that accompany software development. My work has provided me with incredible opportunities to collaborate with and learn from these exceptional professionals.

I am grateful for Denise Mitchell and Rebecca Harrigan, incomparable technology leaders, partners, and clients, and for the great team at Kellogg Corporation. Also, for the executive group and the great IT, BA, and PM teams at Harvard Business Publishing. I also owe a debt of gratitude to my own team, from whom I am always learning: Frank G. Andrews, Pedro Garrett, Atsushi Tatsuoka, and Steve Vance.

Thanks goes as well to fellow technology writer Mike Barlow, who showed such generosity of spirit and time in championing this book. To Sheila and Gerry Levine, whose wise counsel has supported me through everything from content to contracts. And, to the International Women's Writing Guild, whose sisterhood of writers and teachers filled all kinds of gaps from craft to the business of writing.

What could be more valuable to a writer than an editorial partner with skill, wit, and wisdom? Hilary Poole, writer and editor extraordinaire, was one of the original collaborators on this project. Thank you for all your help from tidied-up commas to slashed technology jargon.

To the incredible team at Wiley: Sheck Cho, Pete Gaughan, Connor O'Brien, Michael Henton, and Caroline Maria Vincent. Your professionalism and collaboration are all any author could hope for.

And finally, to Chris Moschovitis, whose brilliance in technology and so many other things awes me every day. I am lucky to go to work with you each morning and to come home to you each night.

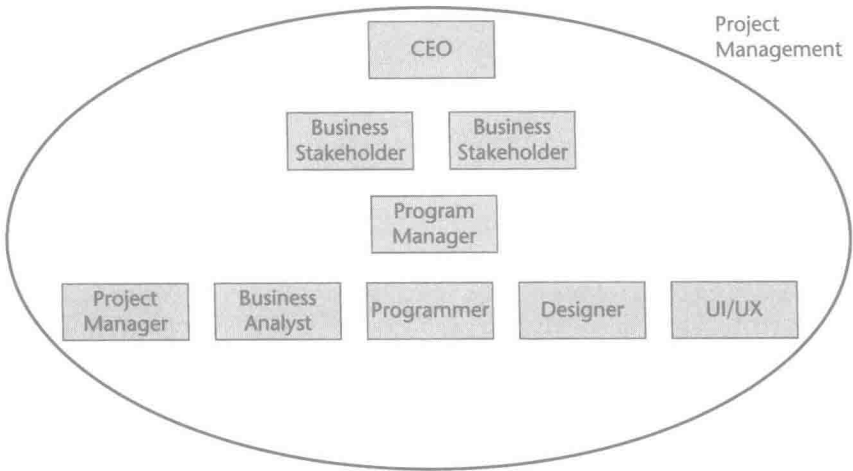


Figure 1.2

For Medium-to-Large Projects

Project management looks different depending on the size of the project. The sort of project management involved in putting up your personal WordPress blog likely happens all in your own head. Inside of businesses most projects that last more than a couple of weeks require some kind of external project management.

For this book, I will be giving processes and advice appropriate for medium-to-large projects. This generally means projects that take at least three months and may last as long as two years. The reason I've chosen to focus on projects of this size is they are the most common. Also, smaller projects, as noted earlier, probably don't need much project management at all. Finally, I have found that once people understand the fundamentals of managing medium-to-large size projects, they are able to scale the processes and recommendations both up and down to suit their needs.

Agile vs. Waterfall

For those of you who don't know these terms, don't worry. We'll talk more about this in Chapter 2. Many people familiar with modern software development want to know right up front: Are you talking about project management practices for Agile development or for Waterfall?

This book assumes the incorporation of key Agile methodologies, while acknowledging that most businesses cannot adhere to a purely Agile style. In

fact, the issue of Agile vs. Waterfall is a major reason I wrote this book. After running hundreds of software projects inside many different types of businesses, I discovered that good project management is a blend. It must be “agile enough.”

Why Listen to Me?

As the CEO of emedia (www.emediaweb.com), I have been developing software and managing software teams since 1994. I am proud to say that we have a superb track record in delivering software development projects, from large database implementations to small business websites, on time and on budget.

I’ve managed website launches, database migrations, finance system replacements, custom application development, content management system implementations, and packaged software deployments. The dollar value of these projects has ranged from the small thousands to the tens of millions. It’s allowed me to see the commonalities among many software development endeavors and to develop strategies to improve the process.

I am also proud to say that the majority of my customers have been with me for years—a decade or longer in some cases. When you have a challenging undertaking, you want a partner you trust.

Who Is This Book For?

If you’re inside an organization undertaking a software development project, this book is for you. If you are a consultant or vendor who rolls out software for customers, this book is for you, too. If you are a business leader, programmer, project manager, or program manager, this book is also for you.

This book has chapters on all the pragmatic stuff you need to know: organizing and staffing a project; the common conflicts that crop up; planning; risk management; and, of course, budgeting. There’s even a chapter for people who are midway through a project and are encountering problems. If you read from start to finish, the book will take you through the life cycle of a software development project. Or, you can dip into the chapter based on your particular interest or area of challenge.

CONTENTS

FOREWORD xvii

ACKNOWLEDGMENTS xix

ABOUT THE AUTHOR xxi

INTRODUCTION xxiii

CHAPTER 1 Software Development Explained: Creativity Meets Complexity 1

A Definition of Software Development 1

Why Is Software Development So Difficult? Hint: It's *Not* Like Building a House 1

The Simple, the Complicated, and the Complex 2

Metaphor #1: Piles of Snow 3

Metaphor #2: The Ikea Desk 4

Metaphor #3: Heart Surgery 5

Using the Three Metaphors in Project Management 6

CHAPTER 2 Agile, Waterfall, and the Key to Modern Project Management 7

Agile and Waterfall 7

Waterfall 7

Waterfall's Problems 8

The Requirements Requirement 9

Inflexibility 9

Loss of Opportunity and Time to Market 9

Customer Dissatisfaction 10

Agile 10

Lack of Up-Front Planning 12

Lack of Up-Front Costs 12

Stakeholder Involvement 13

Extensive Training 13

Where Agile Works Best 14

The Need for Up-Front Requirements in Many Projects	14
The Real World	15
Agile Enough	15
The Software Development Life Cycle	15

CHAPTER 3 Project Approaches; Off-the-Shelf and Custom Development; One Comprehensive Tool and Specialized Tools; Phased Launches and Pilots 17

The Custom vs. Off-the-Shelf Approach	18
History	18
The Benefit of Off-the-Shelf	19
Off-the-Shelf Examples	19
Thinking You're Editing When You're Actually Creating	20
Common Challenges with Off-the-Shelf Software	20
Business Compromise	21
Discovering You Made the Wrong Choice with Packaged Software	21
Breaking the Upgrade Path	21
Locked into a Partnership and the Product Roadmap	22
Expense of Off-the-Shelf	22
Where Packaged Software Works Well	23
Frameworks and the Blurring Worlds of Custom and Packaged Software	23
Integrations vs. One Tool for the Job	24
To Phase or Not to Phase	25
Bigger Is Not Always Better	26
The Pilot Approach	26
Why Not Pilot?	27

CHAPTER 4 Teams and Team Roles and Responsibilities Defined 29

Teams and the Roles on Teams	29
Project Leadership	30
The Key Business Stakeholder	31
The Project Sponsor	31
The Program Manager	32
Project Manager	32
Multiple Project Managers	33

Confusion About the Project Manager Role; It's More Limited than You Think	34
Project Team	34
The Business Analyst	35
User Experience	35
Designer	35
The Programmers	35
Architect	36
Systems Administrator	36
Team Member Choice and Blending Roles	37
Getting All the Roles Covered	37
Real-World Examples for Role-Blending	38
Project Sponsor as Program Manager	38
Program Manager as Business Analyst	39
Front-End Programmer as User Experience	39
Design, UX, and Business Analysis	40
Back-End Programmer as Architect	40
SysAdmin as Architect	40
Professionals and Personalities	40
Programmers	40
Project Managers	41
Business Analysts and User Experience People	42
Architects and Systems Administrators	42
Insource or Outsource: Whether to Staff Roles with Internal People or Get Outside Help	43
The Myth that Insourcing Programming Is Better	43
Inexperience with Projects	44
How Knowledge Goes Stale	44
Outsourced Teams	44
When to Use Internal or External Teams	45
Roles Easiest to Outsource	46
Roles "in the Middle"	46
Roles that Are Usually Internal	47
Vendors and Hiring External Resources	47
Some Tech-Types to Avoid: Dot Communists and Shamans	47
The Shamans	48

Boundaries, Responsibilities, and Driving in Your Lane	49
Techies Who Don't Drive in Their Lane	50
Business Stakeholders Who Shirk Responsibilities	50
Business Stakeholders, Step Up!	51
Have a Trusted Technology Partner	52
How Best (and Worst) to Work with Your Technology Partner	52
Too Many Cooks	53

CHAPTER 5 Project Research and Technology Choice; Conflicts at the Start of Projects; Four Additional Project Delays; Initial Pitfalls 55

Choice of Technology, a Definition	56
The Project's Research Phase	56
Current State	56
Integrations and Current State	57
Data and Current State	57
Business Needs	58
Possible Technology Solutions	58
Demos	59
Comparison Grids	59
Talk to Other People, a Journalistic Exercise	60
How Do You Know When Your Research Is Done?	61
Research Reality Check	62
You Can't Run the Control	62
Religious Wars	63
Passion over Reason	64
Business Stakeholders and Controlling Ego	64
How to Stop a Technology Religious War	65
Not So Easy	65
Preventing a Technology Religious War	65
Being Right	66
Stopping a War in Its Tracks	66
Détente and Finally Ending a Technology Religious War	67
Clarity	67
The Role of the CIO	68
Two Most Important Factors in Core Technology Decisions	69

Budget Constraints	69
The Team	69
Choosing Technology and What <i>NOT</i> to Consider: The Future	70
Other Conflicts that Delay the Start of Projects	71
Business Strategy and Organizational Authority	71
Design	73
Blue Sky	73
Overanalysis	74
The Project Charter, a Key Document	74

CHAPTER 6 Final Discovery; Project Definition, Scope, and Documentation 77

Budgeting and Ongoing Discovery; Discovery Work Is Real Work	78
Budgeting Final Discovery	78
What Discovery Costs	79
What Comes Out of Final Discovery: A Plan	79
Getting to a Plan	80
The Murk	80
Getting Out of the Murk	81
The Plan for the Plan—Company A	82
Hosting	82
Content Entry	82
Search	82
Content Pages and Features	83
Integrations	83
Back-end System	83
Data Migration	84
How Anyone Can Make a Plan for the Plan	84
Different Approaches to Elicit the Plan for the Plan	85
Exception to the Murk	86
Breakout Sessions	87
The Weeds Are Where the Flowers Grow	87
Not All Questions Will Be Answered	88
Agile, Waterfall, and Project Documentation	89
The Scope Document	90
Project Summary	90

Project Deliverables	90
Out of Scope	90
Constraints	91
Assumptions	91
Risks	91
Timeline	92
Budget, Scope, Timelining, and Horse-Trading	93
Metrics	93
What About “the List”?	94
Defining and Visualizing and Project Scope	94
What Usually Happens	95
The Chicken and the Egg	95
Common Questions	97
Where Does Design Fit In?	97
Working with Marketing Stakeholders	98
How You Know You’re On the Wrong Track	98
A Word About Ongoing Discovery	99

CHAPTER 7 Budgeting: The Budgeting Methods: Comparative, Bottom-Up, Top-Down, and Blends; Accurate Estimating 101

An Unpleasant Picture	102
What Goes on Behind the Scenes; a Scene	102
Budgeting Type 1: Comparative Budgeting	103
Gotchas with Comparative Budgeting	104
Budgeting Type 2: Bottom-Up Budgeting	104
The Rub in Bottom-Up Budgeting	105
Budgeting Type 3: Top-Down and Blends	105
Why RFPs Don’t Work	106
Accurate Estimating and Comparison Budgeting	107
Effective Estimating in Top-Down and Bottom-Up Budgeting	108
Establish a Base Budget for Programming, Ongoing Discovery, Unit Testing, Debugging, and Project Management	108
Percentages of Each	108
Programming Hours—Raw and Final	109
The Math Part	109