# Optimize Quality for Business Outcomes

## A Practical Approach to Software Testing

Andreas Golze

Charlie Li

Shel Prince

# Introduction

The word *testing* has many meanings. Teachers test their students to see how much they've learned, small children test the taste and feel of almost anything by putting things in their mouths, and scientists test the strength of materials to see how far they can be stressed before they break. (Some people say teenagers do the same thing to their parents!)

But those kinds of testing are not what we discuss in this book. Software testing is a process where we check a behavior we observe against a specified behavior the business expects.

The trial-and-error behavior that helps us learn and understand the world as children is testing without understanding the expected result.

Both methods of testing have elements in common. In both methods, we want to find out what happens if we're using something in a specific way. The primary difference is that in software testing, the tester should know what behavior to expect as defined by the business requirements. We agree on this defined, expected behavior and any user can observe this behavior.

Today, we see folks in the IT industry still confusing these two concepts of testing. Often, two groups of people are doing software testing:

- The developers or engineers who build the software application want to make sure the final product works as they intended. The testing is done from an IT or technical perspective without relating back to the business.
- The end users who will be using the final product in their jobs, in many cases, do ad-hoc testing based on their day-to-day work experience. They make assumptions of how the application should work that come from their individual needs and ideas. Their approach is more trial and error than systemized testing — it's often ineffective and creates a lot of rework.

If each group is testing based on their needs and ideas, who's testing for the business outcomes? Who validates that the application will meet business requirements? Who mitigates the risk of failure of a software application in production?

Software testing, like software development, is a profession. Testers need to master proven techniques. Organizations like the British Computer Society and the International Software Testing Qualifications Board (ISTQB) have begun building standardized training and granting testing certification levels.

Like developers, software testers are a special breed with unique talents that can bridge the gap between the business and IT. They must have technical skills coupled with a business mentality. To build a professional test team and do testing with rigor, not by trial and error, takes time and money. For most companies using IT as the backbone for their business, time and money are the two things that are not readily available.

We, along with our colleagues, have consulted with thousands of customers on quality and testing and have seen the same issues time after time. In this book, we present a pragmatic solution to help address some of these challenges.

Our goal is not to write a comprehensive book on testing or advanced testing theory. We're showing you practical solutions, based on implementation experience. If we can help organizations solve some of their most important quality challenges quickly, we have succeeded in what we've set out to achieve here.

In this book, we address fundamental questions around testing from a business perspective.

- In Chapter 1, we examine why testing is so important to a successful product launch.
- In Chapter 2, we discuss how we know what to test and how to determine that testing has met its objectives.
- In Chapters 3 and 4, we present efficient, effective methods of software test design called behavioral modeling.
- In Chapter 5, we define practical implementation strategies for optimizing software testing, as well as show how to prioritize testing tasks and which parts of testing should be automated.
- In Chapter 6, we give you a *SUCCESSFUL* approach to measuring and using key performance indicators.
- In Chapter 7, we offer some guidance on how to start implementing the concepts in this book.
- The appendices contain technical details, tips, tricks, and guidelines for those who want to go further with these concepts and begin to implement them.

Throughout this book, we use the phrase *business function(s)* to define a set of discrete end-user steps that constitute a basic business activity such as "log in," "search for a flight," or "register user." In some application environments the terms, *process steps* or *work steps* may be used instead.

We hope you gain a new business perspective on software testing and find some practical guidance for implementing quality improvements within your organization.

We might say, “Enjoy!” But we know what you’ll really enjoy is the confidence that you know how to move through your software development cycle in the most efficient, cost-effective way — and that you know your customers will be satisfied at the end of it. We will share this win-win process with you.

# About the Authors

**Andreas Golze**, Global Practice Director of Mercury's Professional Services, is an expert with many years' experience in the IT industry. He is one of the founders of the Mercury Quality Model (MQM), which is now the standard for many large-scale enterprise organizations around the globe. Before his move to Mercury, Andreas served as the Senior Executive at the TestLab Business Unit of SQS AG, having earlier crafted and built up this unit in the role of Department Manager. He was also responsible for developing and expanding partner business. Andreas Golze studied computer science and minored in business studies at the University of the Armed Forces in Munich. His career began with a position as a programming officer at NATO in Belgium. Today, Mr. Golze lives in Cologne, Germany with his wife, Petra and their four children.

**Charlie Li**, Director of Product Marketing for Application Delivery Solutions at Mercury Interactive, has extensive experience consulting with Fortune 500 customers on large-scale quality management and testing projects in his seven years with Mercury. Previously, he served as the Practice Director of Mercury's Professional Services and is one of the founders of the Mercury Quality Model, which is now the standard for many large-scale enterprise organizations around the globe. Charlie has also built relationships and is working closely with partners like SAP, WIPRO, and Infosys, which are driving quality innovations into new markets. Charlie started his career as an IT consultant and software engineer and holds a computer science degree from the Georgia Institute of Technology. Today, Mr. Li lives in Atlanta, Georgia with his wife, Mandy and their dog Lucy.

**Shel Prince,** Principle Process Consultant of Mercury's Professional Services, has been in the software quality field for almost 40 years, entering the workforce after undergraduate study in Physics and graduate studies in Business and Psychology. As one of the co-founders of the Mercury Quality Model, Shel has leveraged his vast experience in the quality field to continuously innovate the quality model. Prior to joining Mercury, he was a consultant for large-scale enterprise companies and governments in many parts of the world. His broad range of experience helped in forming his theories and approach to the fields of software quality and software testing. Today, Mr. Prince lives in Sammamish, Washington with his wife, Nancy and their dog Beau.