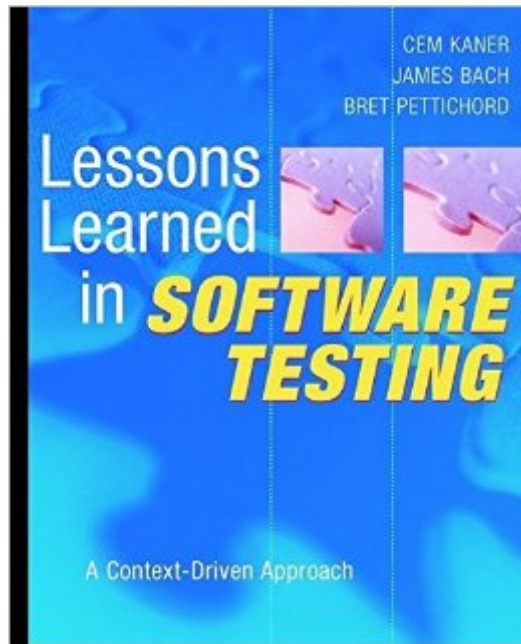


The book was found

# Lessons Learned In Software Testing: A Context-Driven Approach



## Synopsis

Decades of software testing experience condensed into the most important lessons learned. The world's leading software testing experts lend you their wisdom and years of experience to help you avoid the most common mistakes in testing software. Each lesson is an assertion related to software testing, followed by an explanation or example that shows you the how, when, and why of the testing lesson. More than just tips, tricks, and pitfalls to avoid, *Lessons Learned in Software Testing* speeds you through the critical testing phase of the software development project without the extensive trial and error it normally takes to do so. The ultimate resource for software testers and developers at every level of expertise, this guidebook features:

- \* Over 200 lessons gleaned from over 30 years of combined testing experience
- \* Tips, tricks, and common pitfalls to avoid by simply reading the book rather than finding out the hard way
- \* Lessons for all key topic areas, including test design, test management, testing strategies, and bug reporting
- \* Explanations and examples of each testing trouble spot help illustrate each lesson's assertion

## Book Information

Paperback: 320 pages

Publisher: Wiley; 1 edition (December 31, 2001)

Language: English

ISBN-10: 0471081124

ISBN-13: 978-0471081128

Product Dimensions: 7.4 x 0.6 x 9.3 inches

Shipping Weight: 1.3 pounds (View shipping rates and policies)

Average Customer Review: 4.6 out of 5 stars [See all reviews](#) (67 customer reviews)

Best Sellers Rank: #162,753 in Books (See Top 100 in Books) #67 in [Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Testing](#) #244 in [Books > Textbooks > Computer Science > Software Design & Engineering](#) #293 in [Books > Computers & Technology > Networking & Cloud Computing > Networks, Protocols & APIs](#)

## Customer Reviews

This book contains 293 "Lessons". Each seems to be meant for people with certain experiences and certain problems; some very broadly defined, others more tightly. So, how do I grade 293 lessons? One way would be to average them, another to pick on the worst (from my point of view). I choose to pick out the ones that hit me the hardest; the best from my point of view. I've been a developer, a tester, a test manager, and am now a grad student studying testing with Dr. Kaner.

This book was the proximate cause of the last. If I had had this book a couple of years ago, I believe I would have done a much better job as test manager, and my project would have succeeded better with our customer. This is the second best book on testing that I've ever read. By the time I saw Lesson 31, I had already learned it the hard way. "A Requirement is a quality or condition that matters to someone who matters." It doesn't matter what the requirements document says; you ignore the opinion of someone who matters at your peril. I did. Lesson 57: "Make your bug report an effective sales tool." My bug reports developed a pretty good reputation with most of the developers, so I quit paying as much attention to putting convincing arguments in them. Then, we got some new senior developers. I was back at square one without quite realizing how I got there. Don't do that. Lesson 235: "Staff the testing team with diverse backgrounds." When I became test manager, I looked for people like me: computer science degree with developer experience. Well, such people don't work as testers, especially for the location and money we offered. I first hired a young woman with Army training. Later, I figured out how lucky I had been; she was one of the two best testers who worked for me.

[Download to continue reading...](#)

Lessons Learned in Software Testing: A Context-Driven Approach Daniels and Worthingham's Muscle Testing: Techniques of Manual Examination and Performance Testing, 9e (Daniels & Worthington's Muscle Testing (Hislop)) 1000 French Verbs in Context: A Self-Study Guide for French Language Learners: 1000 Verb Lists in Context, Book 2 Software Testing: A Craftsman's Approach, Fourth Edition The Basics of Hacking and Penetration Testing, Second Edition: Ethical Hacking and Penetration Testing Made Easy Penetration Testing: Communication Media Testing (EC-Council Press) ACSM's Resource Manual for Guidelines for Exercise Testing and Prescription (Ascms Resource Manual for Guidlies for Exercise Testing and Prescription) Ruppel's Manual of Pulmonary Function Testing, 10e (Manual of Pulmonary Function Testing (Ruppel)) Manual of Pulmonary Function Testing, 9e (Manual of Pulmonary Function Testing (Ruppel)) Testing Women, Testing the Fetus: The Social Impact of Amniocentesis in America (The Anthropology of Everyday Life) Just Enough Software Architecture: A Risk-Driven Approach Shipping Greatness: Practical lessons on building and launching outstanding software, learned on the job at Google and What Customers Want: Using Outcome-Driven Innovation to Create Breakthrough Products and Services: Using Outcome-Driven Innovation to Create Breakthrough Products and Services Everything I Learned about Life, I Learned in Dance Class Software Engineering Classics: Software Project Survival Guide/ Debugging the Development Process/ Dynamics of Software Development (Programming/General) Surreptitious Software: Obfuscation, Watermarking, and Tamperproofing for

Software Protection: Obfuscation, Watermarking, and Tamperproofing for Software Protection  
Software Reuse: A Holistic Approach (Wiley Series in Software-Based Systems) Developer Testing:  
Building Quality into Software (Addison-Wesley Signature Series (Cohn)) Advanced Software  
Testing - Vol. 3, 2nd Edition: Guide to the ISTQB Advanced Certification as an Advanced Technical  
Test Analyst Testing Computer Software, 2nd Edition

[Dmca](#)