

1. Record Nr.	UNINA9911035061203321
Autore	Bach James
Titolo	Taking Testing Seriously : The Rapid Software Testing Approach
Pubbl/distr/stampa	Newark : , : John Wiley & Sons, Incorporated, , 2025 ©2026
ISBN	1-394-31974-6 1-394-25320-6
Edizione	[1st ed.]
Descrizione fisica	1 online resource (563 pages)
Altri autori (Persone)	BoltonMichael
Disciplina	005.1/4
Soggetti	Computer software - Testing
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di contenuto	Cover -- Title Page -- Copyright Page -- About the Authors -- Acknowledgments -- Contents at a Glance -- Contents -- Foreword -- Reader Support for This Book -- Part I Introduction -- Chapter 1 Why Another Book About Testing? -- Many Cultures of Testing -- Why Us? -- Why Testing? -- Why Testers? -- By The Way, Now Everyone is a Tester. . . -- Why Not "Traditional Testing?" -- Part II Rapid Software Testing Methodology -- Chapter 2 Foundation -- The Meaning of Testing -- Testing vs. Checking -- Testing vs. Performing a Test -- Deep vs. Shallow Testing -- Narrow vs. Broad Testing -- Our Vision for RST -- Design Features -- Foundational Ideas -- Who We Are -- Practitioners and Clients -- Students and Experts -- What We Study -- Systems and Models -- Products and Users -- Quality and Risk -- How We Manage -- Context and Mission -- Process and Methodology -- Roles and Trading Zones -- Tacit and Explicit Knowledge -- Freedom and Formality -- Spontaneity and Deliberation -- Iteration and Emergence -- Control and Legibility -- Stories and Strategy -- Heuristics and Skills -- Activities and Enoughness -- How We See -- Stance and Distance -- Focusing and Defocusing -- Experience and Instrumentation -- Sensemaking and Unconscious Bias -- Experiments and Demonstrations -- Assessment and Measurement -- Chapter 3 How to Do a Test -- The Process of Testing -- Our View of the Scientific Method -- The Fundamental Method of Testing -- Why Do We

Say "Explore" Instead of "Record" or "Measure"? -- How Is the Reality of the Product Explored? -- The Anatomy of a Check -- The Meaning of a Check -- Fail Is Not an Option (for Good Reporting) -- The Anatomy of a Test -- Coverage -- Intentional Coverage -- Incidental Coverage -- To Cover Is to Sample -- How Not to Think About Equivalence Classes -- A Better Way to Think About Equivalence Classes. A Better Way to Think About Boundaries, Too -- Pushing Limits -- Oracles -- Oracles May Take Many Forms -- Procedures -- Designing a Test Procedure -- Anatomy of "One Step" of a Procedure for a Test -- How To Go Wrong When Performing a Test -- The Art of Bug Reporting -- Normal Bug or Enhancement Request? -- How to Investigate a Bug -- How to Investigate Intermittent Problems -- Be Comforted: The Cause Is Probably Not Evil Spirits -- General Suggestions for Investigating Intermittent Problems -- Considering the Causes of Intermittent Problems -- How to Report a Bug -- Formal vs. Informal Bug Reporting -- Elements of a Basic Formal Bug Report -- Give the Bug Report a Good Focus -- Assessing the Significance of a Bug -- The Bug Pipeline -- Chapter 4 How to Do a Test Strategy -- The Structure of Test Strategy -- Welcome to Your New "Normal" -- Pleasing vs. Disturbing -- Easy vs. Stressful -- Acceptable vs. Disallowed -- Natural vs. Contrived -- Us vs. Them -- Typical vs. Unusual -- Standard vs. Special -- "This Is Fine." -- First of All Things: Center Yourself -- Developing a Strategy Is an Ongoing Exploration -- Twelve Test Strategy Entry Points -- What Are You Here For? -- What Do You Need to Learn? -- What Is Happening Right Now? -- How Is Your Testing Constrained? -- What Testing Has Been Done Already? -- How Is the Product Being Built? -- What Is the Product? -- How Important Is Your Testing? -- How Will People Most Likely Use the Product? -- What Testing Is Easy to Do? -- What Do People Expect You to Do? -- What's Fun to Do? -- Seasons of Testing: Strategy Throughout the Project Cycle -- Regression Testing May Not Be What You Think -- The Challenge of Alignment: Strategy Is Different for Testers and Builders -- Testability Enables Test Strategy -- Good Test Strategy Is More Than Just Following Behind Development -- Practices Worth Practicing. Make a Product Coverage Outline -- Apply the Heuristic Test Strategy Model -- Organize Strategy According to Risk -- Risk Analysis Is Mostly Conversation and Consensus -- Brainstorming Using the Four-Part Risk Story -- Putting It All Together: Lenses of Testing -- Be a Problem-Solver, Not a Task-Doer -- A Strategy Example: Bibliography Generator -- Context -- Strategy -- Strategy Example #2: A More Normalized Version -- Activity Types Mentioned in the Table -- What About Oracles? -- Final Thought -- Chapter 5 How to Account for Your Work -- Telling a Compelling Story -- Note-taking Is a Core Testing Skill -- Jon Bach's PROOF Heuristic -- Safety Language -- Telescoping Reports -- James's Low-Tech Testing Dashboard -- Stories Mediate Software Projects -- High Alignment vs. Low Alignment -- The Temptation to Coerce -- Artifact-based vs. Activity-based Management -- Artifact-Based Management -- Beware of Counting Incommensurable Artifacts -- The Basic Problem of Artifact-Based Management -- Activity-based Management -- Session-Based Test Management -- Thread-Based Test Management -- Please Don't Fake Your Testing -- Chapter 6 How to Apply Automation in Testing -- Robots! Help! -- The Trouble with "Automation" -- In RST, we don't call it test automation . . . -- . . . yet we explore many ways to use tools -- Consider Augmented Experiential Testing -- Beware of Oversimplified Output Checking -- . . . we incorporate tools incrementally and opportunistically . . . -- Consider Applying a Blink Oracle -- Notice the Hidden Costs of Automation -- Consider Recruiting a Dedicated Toolsmith -- How

Testers and Toolsmiths Can Work Together -- . . .and we promote testability, so tools work better -- Classic Traps of Automation in Testing -- Traps of Ignorance -- Scripting Trap -- Trusting Trap -- Atrophy Trap -- Obscurity Trap -- Traps of Economy -- Shallowness Trap.

Testability Trap -- Maintenance Trap -- Sunk Cost Trap -- Learning Curve Trap -- Traps of Alignment -- Rathole Trap -- Legibility Trap -- Harmony Trap -- Considerations for GUI-level Automation -- First Things First -- Programmatic Access to the GUI Can be Difficult to Achieve -- Variation in the GUI Multiplies the Cost and Trouble -- GUI automation means you are simulating users. How good is that simulation? -- Considerations for Adopting a Tool -- Capabilities and Power -- Effort to Operate -- Feasibility of Adoption -- Learning, Troubleshooting, and Support -- The Golden Rule of Tool Adoption -- Chapter 7 How to Approach AI and Testing -- What If We Could Test by Magic? -- How Is Modern AI Special? -- Problems to Look For with AI in Testing -- Transpection: A Basic Skill for Collaborating with AI -- Beware of the Productivity Paradox -- Are People Magic Boxes? Can Be . . . -- Part III Application and Customization -- Chapter 8 Prospective Testing -- Notice Something Before We Say Anything More . . . -- But . . . Why Bother Doing This at All? -- How to Do Prospective Testing -- Who Is Involved in Prospective Testing? -- What Does Prospective Testing Look Like? -- When Does Prospective Testing Happen? -- How Can a BA or Developer Prepare for Prospective Testing? -- How Can You (the Tester) Deal with Resistance to Questions during Meetings? -- "Analysis Paralysis" or "Go Fast and Break Things?" -- A Cheat Sheet for Prospective Testing -- What exactly are we talking about? -- Is this worth discussing here and now? -- What exactly are we trying to achieve? -- What influences must we consider? -- What other features or requirements will be affected? -- What specific data or conditions must this feature be able to process or work with? -- What are the merits of different ways of designing or implementing this feature?.

How will the new feature handle errors or recover from failure? -- How will we test the new feature once it exists? -- Chapter 9 Test Reporting (Without Pretentious Metrics) -- Metrics Are Nothing Without a Story -- Bad Metrics -- Elements of Bad Metrics -- Coverage -- A Simple Story for Coverage -- Release Coverage Outline -- Better Test Reports -- Sample Test Report -- Chapter 10 Working with Quality Characteristics -- Developing the Model -- Quality Characteristics -- Internal Quality Characteristics -- Using the Model -- Epilogue -- Chapter 11 Adventures in Testability -- Testability on the High Seas -- Advocating for Testability -- Step 1: Identify Things That Make It Difficult to Test -- Step 2: Determine Who Can Help You Solve the Problem -- Step 3: Sell It ("What's In It for Them?") -- My Job Is Testability -- Chapter 12 Mindopsy: Dissecting My Thinking as a Tester -- What Is a Mindopsy? -- Conversation with the Business Analyst -- Analysis of the Conversation -- Tree-Type Notes for This Conversation -- Summary -- Chapter 13 Rapid Usability Testing -- Why Do We Fail to Test for Usability? -- Understanding Users and Goals -- Creating Personas and User Goals -- The Electricians and the Rental Company -- "Eric," the Electrician -- "Rob," the Rental Guy -- Living the Persona -- How to Document a Useful Persona -- Rapid Usability Testing Basics -- Recruit Users -- Plan and Prepare -- Run the Test -- I Am Testing an Off-the-Shelf Standard System-What Should I Do? -- Checking Usability -- General Checklist of Usability (Nielsen-Norman) -- Web Content Accessibility Guidelines (WCAG) -- Chapter 14 RST Meets Signals-Based Testing -- Signals for Testing -- How This Looked at Microsoft -- Self-

Taught AI-Generated Workload -- Analyzing the Signals and Looking for Failure -- Feature Coverage -- Crashes and Hangs -- Assertion Failures -- Performance Markers.
Problematic Patterns in Events.

Sommario/riassunto

Elevate your software testing approach with a methodology from industry leaders who dedicated their careers to studying, practicing, and teaching the craft of testing. Dive into the world of expert software testing with Taking Testing Seriously: The Rapid Software Testing Approach.
