## Contents

1 **Introduction** ................................................................. 1  
   1.1 Consistency, Complexity, and Change ............................ 1  
   1.2 Synopsis ............................................................... 2  

2 **What Is Software Quality, and Why Does it Matter?** .......... 7  
   2.1 Why Care about Software Quality? ................................. 7  
   2.2 What Drives Software Quality Assurance? ......................... 14  
   2.3 Defining “Software Quality” ........................................ 16  
      2.3.1 The Challenge of Defining Quality ......................... 16  
      2.3.2 Quality Models - a Historical Perspective ............... 18  
   2.4 Key Points ............................................................ 21  

3 **Software Development Processes and Process Improvement** ... 23  
   3.1 Process and Process Improvement in Manufacturing .......... 24  
      3.1.1 The Industrial Revolution ................................... 24  
      3.1.2 Plan Do Check Act ............................................. 26  
      3.1.3 Quality-Driven Manufacturing in Japan .................... 27  
      3.1.4 Total Quality Management .................................... 30  
   3.2 The Software Development Process ............................... 31  
      3.2.1 The Waterfall Model .......................................... 33  
      3.2.2 Iterative and Incremental Software Development ......... 35  
   3.3 Agile Software Development ....................................... 38  
      3.3.1 The Principles of Agile Software Development ........... 38  
      3.3.2 An Example: SCRUM ......................................... 39  
      3.3.3 Relation to Total Quality Management ..................... 42  
      3.3.4 Why Not Always Go Agile? ................................... 44  
   3.4 Software Process Improvement - The Capability Maturity Model 45  
   3.5 Key Points ............................................................ 48
## 4 Managing Requirements and Code

4.1 Managing Requirements

4.1.1 What is a Requirement?

4.1.2 Requirements Elicitation

4.1.3 Requirements Documents

4.1.4 Security Requirements

4.1.5 Tracing Requirements

4.1.6 Prioritisation

4.1.7 Oversight with Kanban boards

4.2 Writing Maintainable Source Code and Handling Change

4.2.1 Coding Conventions and Design / Architecture Patterns

4.2.2 Collaborative Development and Version Repositories

4.3 Key Points

## 5 Planning Activities and Predicting Costs

5.1 Planning

5.1.1 Program Evaluation and Review Technique (PERT)

5.1.2 Gantt Charts

5.2 Predicting Costs

5.2.1 Base Models

5.2.2 Parameter Fitting by Linear Regression

5.2.3 COCOMO

5.2.4 Planning Poker

5.2.5 Uncertainty and Predictive Accuracy

5.2.6 Keeping Track of Progress

5.3 Key Points

## 6 Testing

6.1 The Foundations of Software Testing

6.2 White-Box Testing

6.2.1 Code coverage

6.2.2 White Box Test Generation

6.2.3 The Case(s) Against Code Coverage

6.2.4 Goto Fail: A Case For Code Coverage

6.2.5 An Alternative: Mutation Testing

6.3 Black-Box Testing

6.3.1 Specification-Based Testing

6.3.2 Random Testing

6.3.3 Exposing Security Flaws with Fuzz-Testing

6.4 Key Points

## 7 Software Inspections, Code Reviews, and Safety Arguments

7.1 Formal Inspections

7.2 Modern Code Reviews - Reviewing Code During Development

7.2.1 Tool-Driven Code Review
7.2.2 Pull-Based Development ........................................ 130
7.2.3 The Impact of MCR on Software Development and Quality 131
7.3 Code Reviewing Techniques ........................................ 132
  7.3.1 Tool-Driven Code Review ..................................... 133
  7.3.2 Developer-driven Code Reviews ............................... 134
7.4 Safety Arguments and Inspections of Safety Requirements ........ 136
  7.4.1 Checklists ....................................................... 136
  7.4.2 Safety Argumentation and the Goal Structure Notation .... 138
7.5 Key Points ............................................................ 139

8 Measurement ............................................................ 141
  8.1 Measurement Basics ............................................... 142
  8.2 Metrics ............................................................... 147
    8.2.1 Size and Complexity ......................................... 148
    8.2.2 Modularity Metrics ............................................ 153
    8.2.3 Maintainability Metrics and the Maintainability Index ... 158
  8.3 Validity and the Use of Goal Question Metric .................. 159
    8.3.1 Problems of Validity ......................................... 159
    8.3.2 Goal Question Metric ........................................ 160
  8.4 Key Points ............................................................ 162

9 Conclusions ............................................................. 165
  9.1 Topical and Emerging Quality Concerns ........................ 165
    9.1.1 Autonomy in Socio-Technical Systems ................... 165
    9.1.2 Data-Intensive, Untestable Systems ....................... 167
  9.2 Concluding Remarks ............................................... 169

References ............................................................... 171

Index ................................................................. 179
Software Quality Assurance
Consistency in the Face of Complexity and Change
Walkinshaw, N.
2017, XI, 181 p. 63 illus., 41 illus. in color., Softcover
ISBN: 978-3-319-64821-7