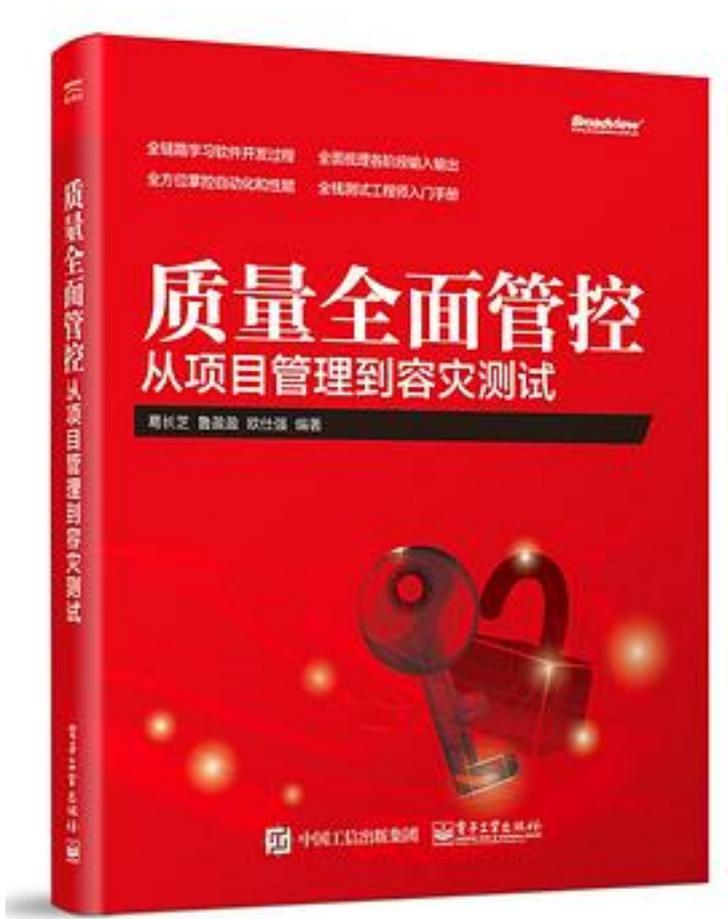


质量全面管控——从项目管理到容灾测试



[质量全面管控——从项目管理到容灾测试 下载链接1](#)

著者:葛长芝

出版者:电子工业出版社

出版时间:2017-2

装帧:平装

isbn:9787121307867

《质量全面管控——从项目管理到容灾测试》专门为有志于软件测试的工程师打开软件测试的大门，笔者结合案例讲解测试实践技术，主要内容有：项目管理、项目需求管理、代码质量控制、自动化部署、软件测试、安全测试与安全管理、自动化测试基础、自

自动化测试框架、性能测试、性能分析、监控平台与故障排查、灾难恢复与容灾测试等。
《质量全面管控——从项目管理到容灾测试》使用了大量的原创图表，提供了基础工具的使用方法和流程。

《质量全面管控——从项目管理到容灾测试》图文并茂，通俗易懂，提供的大量实例可以使读者边学习边实践，深入理解书中的内容，并将所学到的知识应用于实际项目中，对于初中级软件测试工程师来说是不可多得的工具书。

作者介绍:

葛长芝

2001年毕业于东北师范大学，从事软件测试16年，先后就职于益海嘉里集团，任职测试经理，快钱支付公司任职测试架构师，现在就职于K12在线教育公司掌门1对1，任职测经理。具有丰富的软件开发和软件测试经验，设计开发自动化测试框架，精通性能测试、容量规划和容灾测试。对于软件测试始终保持极大的兴趣，对于测试前置和全链路测试，有非常独到的见解。

目录: 第1章 项目管理 1

1.1 项目管理概述 2

1.2 软件项目管理 5

1.3 项目质量管理. 10

1.4 项目管理流程. 11

1.4.1 项目启动. 11

1.4.2 项目计划. 13

1.4.3 软件需求阶段 14

1.4.4 变更管理. 16

1.4.5 设计阶段. 18

1.4.6 构建阶段. 19

1.4.7 测试阶段. 21

1.4.8 部署与试运行 23

1.4.9 项目总结. 25

1.5 项目管理十诫. 26

1.6 项目管理工具对比. 28

1.6.1 Microsoft Project 28

1.6.2 Redmine 29

1.6.3 Feng Office 30

第1章 项目管理 1

1.1 项目管理概述 2

1.2 软件项目管理 5

1.3 项目质量管理. 10

1.4 项目管理流程. 11

1.4.1 项目启动. 11

1.4.2 项目计划. 13

1.4.3 软件需求阶段 14

1.4.4 变更管理. 16

1.4.5 设计阶段. 18

1.4.6 构建阶段. 19

1.4.7 测试阶段. 21

1.4.8 部署与试运行 23

1.4.9 项目总结. 25

1.5 项目管理十诫. 26

1.6 项目管理工具对比. 28	28
1.6.1 Microsoft Project 28	28
1.6.2 Redmine 29	29
1.6.3 Feng Office 30	30
2.5.3 《软件需求规格说明书》. 72	72
2.6 测试需求 73	73
2.7 需求管理工具. 74	74
2.7.1 Rational RequisitePro 75	75
2.7.2 TelelogicDoors 76	76
2.8 Plandora 实战 76	76
2.8.1 搭建Plandora 77	77
2.8.2 管理员配置 78	78
2.8.3 前端用户. 82	82
2.8.4 需求变更实例 88	88
2.9 要点回顾 92	92
第3章 代码质量控制 93	93
3.1 静态代码分析. 94	94
3.2 代码文档规范. 94	94
3.2.1 命名规范. 95	95
3.2.2 编码规范. 96	96
3.2.3 注释规范. 97	97
3.2.4 异常处理规范 102	102
3.2.5 单元测试规范 103	103
3.2.6 文件解析规范 103	103
3.3 Sonar 简介 106	106
3.3.1 Sonar 体系架构. 107	107
3.3.2 Sonar 代码规则. 108	108
3.4 Sonar 服务端. 110	110
3.4.1 环境要求 110	110
3.4.2 Sonar 服务器搭建 110	110
3.4.3 Sonar 配置 112	112
3.4.4 Sonar 插件 114	114
3.5 Sonar 客户端. 116	116
3.5.1 Sonar-Runner 客户端. 116	116
3.5.2 Maven 插件方式 117	117
3.5.3 Ant 插件方式. 118	118
3.5.4 Eclipse 插件方式 119	119
3.6 最佳实践 121	121
3.6.1 项目配置 122	122
3.6.2 持续审查 123	123
3.6.3 结果分析 125	125
3.6.4 集成曲线图. 128	128
3.7 要点回顾 129	129
第4章 自动化部署 130	130
4.1 引入自动化部署. 131	131
4.1.1 复杂的手工部署 131	131
4.1.2 自动化部署方式 132	132
4.1.3 持续集成思想 133	133
4.2 自动化部署的特点 134	134
4.2.1 环境一致性. 134	134
4.2.2 部署系统化. 134	134
4.2.3 配置集中化. 135	135
4.3 版本控制 137	137
4.3.1 Git 简介. 137	137

- 4.3.2 Git 部署. 138
- 4.3.3 Git 客户端使用. 141
- 4.3.4 Git 相关操作 144
- 4.3.5 代码管理 147
- 4.4 JDeploy 平台. 151
- 4.4.1 认识JDeploy 151
- 4.4.2 JDeploy 部署配置 151
- 4.4.3 一键部署项目 152
- 4.5 要点回顾 155
- 第5章 软件测试. 156
- 5.1 软件测试 156
- 5.1.1 软件测试发展史 157
- 5.1.2 软件测试的未来 160
- 5.1.3 测试部门组织架构 161
- 5.1.4 软件测试的基本类型. 163
- 5.2 测试模型 164
- 5.2.1 瀑布模型 164
- 5.2.2 敏捷模型 165
- 5.2.3 敏捷测试与传统测试的区别. 166
- 5.3 系统测试流程 167
- 5.4 根据需求原型设计测试用例 171
- 5.4.1 需求原型规范样式 171
- 5.4.2 设计测试用例 173
- 5.5 缺陷描述 175
- 5.5.1 缺陷属性 175
- 5.5.2 缺陷描述示例 176
- 5.6 测试的策略. 178
- 5.7 测试过程的经验总结. 179
- 5.8 质量保证 180
- 5.9 要点回顾 181
- 第6章 安全测试与安全管理 182
- 6.1 安全测试 182
- 6.1.1 安全测试概述 183
- 6.1.2 安全测试的基本过程. 184
- 6.1.3 安全测试与安全运维. 186
- 6.1.4 安全测试工具 186
- 6.1.5 安全测试用例 187
- 6.2 开发安全规范 189
- 6.2.1 跨站脚本安全规范 189
- 6.2.2 SQL 注入安全规范. 191
- 6.2.3 页面组件和敏感数据的安全规范 193
- 6.2.4 Java 安全规范 196
- 6.2.5 应用集成安全规范 197
- 6.3 代码安全审核工具Checkmarx. 197
- 6.3.1 SQL 注入 198
- 6.3.2 反射型跨站脚本攻击. 199
- 6.3.3 储存型XSS. 201
- 6.3.4 HTTP 响应头分裂 (Http_Response_Splitting) 201
- 6.4 安全漏洞 202
- 6.4.1 信息收集 202
- 6.4.2 口令入侵 204
- 6.4.3 心脏滴血漏洞 207
- 6.5 OSSIM 安全管理平台 209
- 6.5.1 OSSIM 架构 210

- 6.5.2 安装与部署. 211
- 6.5.3 OSSIM 控制台. 211
- 6.5.4 Web 界面配置 212
- 6.5.5 OSSIM 使用实战. 214
- 6.6 要点回顾 219
- 第7章 自动化测试基础 220
 - 7.1 自动化基础. 221
 - 7.1.1 自动化测试流程 222
 - 7.1.2 自动化测试特点 225
 - 7.1.3 自动化测试工具 226
 - 7.1.4 标记语言介绍 228
 - 7.2 Web 自动化测试. 233
 - 7.2.1 元素定位 234
 - 7.2.2 Selenium IDE 238
 - 7.2.3 Selenium 使用 241
 - 7.3 接口自动化测试. 247
 - 7.3.1 接口测试类型 248
 - 7.3.2 接口测试工具 249
 - 7.3.3 Mock 测试 250
 - 7.3.4 HTTP 协议测试 254
 - 7.4 TestNG 框架 258
 - 7.4.1 TestNG 配置 258
 - 7.4.2 TestNG 注解 260
 - 7.4.3 测试套件 260
 - 7.4.4 数据驱动 265
 - 7.4.5 执行测试结果 267
 - 7.4.6 测试集成 268
 - 7.5 要点回顾 270
- 第8章 自动化测试框架 271
 - 8.1 框架分析 272
 - 8.1.1 框架设计目标 272
 - 8.1.2 业务流程层次分析 272
 - 8.1.3 业务流程测试自动化. 273
 - 8.1.4 手工用例自动化 274
 - 8.2 框架设计 274
 - 8.2.1 框架设计思想 274
 - 8.2.2 框架物理架构 276
 - 8.2.3 框架逻辑架构 277
 - 8.2.4 框架工作流程 278
 - 8.3 框架开发 278
 - 8.3.1 创建测试用例 279
 - 8.3.2 创建测试数据 280
 - 8.3.3 创建测试项目 281
 - 8.3.4 开发框架运行类 285
 - 8.3.5 开发公共接口 288
 - 8.3.6 添加日志报告 292
 - 8.4 脚本开发 292
 - 8.4.1 编写测试脚本 292
 - 8.4.2 调试运行脚本 293
 - 8.4.3 上传脚本 294
 - 8.5 持续集成 296
 - 8.5.1 Jenkins 服务器搭建. 296
 - 8.5.2 Jenkins 相关插件 298
 - 8.5.3 部署测试执行机 299

- 8.5.4 分布式测试. 300
- 8.5.5 配置测试任务 303
- 8.5.6 查看运行结果 310
- 8.6 要点回顾 312
- 第9章 性能测试. 313
- 9.1 性能测试基础 313
- 9.1.1 性能术语 314
- 9.1.2 需求分析与策略 317
- 9.2 测试利器之LoadRunner. 321
- 9.2.1 LoadRunner 安装贴士 321
- 9.2.2 脚本与优化. 322
- 9.2.3 设置场景 339
- 9.2.4 运行场景 340
- 9.2.5 收集和分析结果 341
- 9.3 测试利器之JMeter 343
- 9.3.1 JMeter 介绍. 343
- 9.3.2 JMeter 脚本与优化. 344
- 9.3.3 收集监控数据 353
- 9.3.4 运行测试 353
- 9.3.5 JMeter 使用小结 354
- 9.4 性能测试框架搭建 355
- 9.4.1 JMeter 配置监听器. 355
- 9.4.2 InfluxDB 数据库配置 356
- 9.4.3 InfluxDB Graphite Listener 配置. 357
- 9.4.4 查看InfluxDB 结果 357
- 9.4.5 Grafana 配置 358
- 9.5 性能测试实战 359
- 9.5.1 明确测试需求 359
- 9.5.2 选取测试方法和策略. 361
- 9.5.3 准备测试脚本 362
- 9.5.4 执行与分析测试结果. 362
- 9.5.5 提出调优建议 364
- 9.5.6 交付测试报告 364
- 9.6 性能调优 364
- 9.6.1 CPU 使用率过高 364
- 9.6.2 I/O 使用率过高. 365
- 9.6.3 进程数调整. 367
- 9.6.4 线程不安全. 367
- 9.6.5 数据库连接数过少 368
- 9.6.6 数据导入慢. 369
- 9.7 要点回顾 369
- 第10章 性能分析. 371
- 10.1 系统硬件资源监控 372
- 10.1.1 nmon 工具. 373
- 10.1.2 Linux 系统监控命令 378
- 10.2 JVM 监控与分析 386
- 10.2.1 JVM 基础 386
- 10.2.2 JVM 垃圾回收 388
- 10.2.3 常见JVM 命令 390
- 10.2.4 堆分析工具MAT 402
- 10.2.5 JConsole 403
- 10.2.6 JProfiler 406
- 10.3 数据库性能分析 415
- 10.3.1 软解析和硬解析. 415

- 10.3.2 SQL 执行计划分析 417
- 10.3.3 数据库连接数监控 418
- 10.3.4 Oracle 数据库性能诊断报告AWR. 419
- 10.4 要点回顾 427
- 第11章 监控平台与故障排查 428
 - 11.1 监控系统 428
 - 11.1.1 日志监控平台. 429
 - 11.1.2 硬件和应用监控平台 429
 - 11.2 Zabbix 简介 430
 - 11.2.1 系统架构 430
 - 11.2.2 配置Zabbix 432
 - 11.2.3 常见的配置问题. 436
 - 11.2.4 监控主机 438
 - 11.3 美化界面Grafana 441
 - 11.3.1 部署Grafana. 441
 - 11.3.2 使用Grafana. 442
 - 11.4 服务器故障排查 447
 - 11.4.1 清楚故障的前因后果 447
 - 11.4.2 搜寻蛛丝马迹. 447
 - 11.4.3 列出当前运行的进程 449
 - 11.4.4 监听网络服务. 451
 - 11.4.5 查看硬件状态. 451
 - 11.4.6 列出挂载点和文件系统. 457
 - 11.4.7 过滤内核和中断信息 459
 - 11.4.8 定时任务 460
 - 11.4.9 分析系统日志. 461
 - 11.5 要点回顾 461
- 第12章 灾难恢复与容灾测试 462
 - 12.1 灾难恢复 463
 - 12.1.1 灾难恢复的规范. 463
 - 12.1.2 灾难恢复能力等级 466
 - 12.1.3 灾难恢复的关键指标 468
 - 12.2 容灾测试 471
 - 12.2.1 容灾的起源 471
 - 12.2.2 容灾的定义 472
 - 12.2.3 容灾的区别 474
 - 12.3 详解容灾测试 475
 - 12.3.1 容灾测试的目标. 475
 - 12.3.2 职责的划分 476
 - 12.3.3 容灾测试的流程. 477
 - 12.4 容灾测试实战 478
 - 12.4.1 容灾测试计划. 479
 - 12.4.2 容灾用例与Bug. 480
 - 12.4.3 容灾线上演习. 484
 - 12.4.4 容灾长期规划. 485
- 参考文献. 486
- • • • • [\(收起\)](#)

[质量全面管控——从项目管理到容灾测试_下载链接1](#)

标签

测试

全面

评论

了解一下测试的方方面面！

可以作为了解软件质量管理的入门书

覆盖项目全部环节 项目、测试相关部分详细 有实际借鉴意义

[质量全面管控——从项目管理到容灾测试_下载链接1](#)

书评

[质量全面管控——从项目管理到容灾测试_下载链接1](#)