



工业和信息化“十三五”
人才培养规划教材



软件测试 管理与实践

Software Testing Management and Practice

赵聚雪 杨鹏 ◎ 主编



内容简洁、**结构严谨**、逻辑分明、**条理清晰**

以**理论讲解**、**工具实训**、**项目实践**三条线贯穿

通过理论与实践的结合，达到**学以致用**的目的



中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化部“
人才培养规划教材



软件测试 管理与实践

Software Testing Management and Practice

赵聚雪 杨鹏 ◎ 主编

人民邮电出版社

北京

图书在版编目 (C I P) 数据

软件测试管理与实践 / 赵聚雪, 杨鹏主编. — 北京:
人民邮电出版社, 2018.5
工业和信息化“十三五”人才培养规划教材
ISBN 978-7-115-47769-9

I. ①软… II. ①赵… ②杨… III. ①软件开发—程
序测试—高等学校—教材 IV. ①TP311.55

中国版本图书馆CIP数据核字(2018)第061165号

内 容 提 要

本书较为全面地介绍了软件测试的相关理论和工具, 内容翔实, 可操作性强, 简明易懂。本书从实用角度出发, 重点培养学生解决实际问题的能力。

全书共 11 章, 主要包括测试流程管理、测试需求分析、测试计划制订、测试用例编写、测试执行、测试缺陷提交、测试报告编写、测试团队建立、测试管理工具应用等内容。

本书以理论讲解、工具实训、项目实践三条线贯穿。读者通过学习可以理解相关的理论方法; 通过工具实训掌握具体的操作方法; 通过项目实践会应用所学的理论和工具。书中设计了 6 个工具实训任务、8 个项目实践任务、若干理论练习题来辅助读者掌握软件测试管理的具体理论、方法和工具。除此之外, 理论和工具实训都引用教学项目“在线课程作业管理系统”, 并在第 11 章给出了该项目实践的关键文档, 同时在附录中给出了软件测试项目开展过程中经常使用的典型模板, 供读者参考。

本书可以作为高校计算机相关专业软件测试相关课程的教材, 也可以作为想从事软件测试工作的自学者的参考书。建议阅读本书之前先了解与软件测试基础相关的测试概念、用例设计方法等内容。

-
- ◆ 主 编 赵聚雪 杨 鹏
责任编辑 左仲海
责任印制 马振武
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市君旺印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 12.75 2018 年 5 月第 1 版
字数: 307 千字 2018 年 5 月河北第 1 次印刷
-

定价: 39.80 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147 号



前言

FOREWORD

软件测试项目实践涉及软件测试过程中的相关工作，是软件测试工程师、软件项目经理、软件测试经理的典型工作任务，是软件测试工程师必须要具备的技能，也是计算机相关专业的重要课程。在软件测试项目开展过程中，测试任务的划分往往是按照功能模块或者测试类型进行的。测试工程师要管理所负责部分的测试需求、方案、用例、执行以及缺陷提交和总结报告，因此，测试工程师的日常工作包含了软件测试流程中的大部分工作。

建议阅读本书之前先了解与软件测试基础相关的测试概念、用例设计方法等内容。本书以训练读者的软件测试项目实践能力为目标，全面地介绍了软件测试的相关理论和工具，内容翔实，可操作性强，简明易懂。本书主要包括测试流程管理、测试需求分析、测试计划制订、测试用例编写、测试执行、测试缺陷提交、测试报告编写、测试团队建立、测试管理工具应用等内容。全书从实用角度出发，重点培养学生利用理论和工具动手解决实际问题的能力。

通过理论学习、工具实训、项目实践，读者不仅能够掌握软件测试相关的理论，还要掌握相关工具以及操作方法，更重要的是能够通过项目实践获取将理论和工具应用到具体的项目中的能力，最终具备解决实际问题的能力。

本书的参考学时为 46~60 学时，建议采用理论实践一体化教学模式，各章的参考学时见下面的学时分配表。

学时分配表

章	课 程 内 容	学 时
第 1 章	软件测试管理概述	2
第 2 章	软件测试流程管理	2~4
第 3 章	测试需求分析	8~10
第 4 章	测试计划	8~10
第 5 章	测试用例设计和管理	6~8

续表

章	课 程 内 容	学 时
第 6 章	测试缺陷管理及分析	8 ~ 10
第 7 章	测试执行和报告	6 ~ 8
第 8 章	测试组织管理	2
第 9 章	测试相关的其他过程	2 ~ 4
第 10 章	ALM 实践应用	2
课时总计		46 ~ 60

本书关于软件质量模型、软件测试模型部分的介绍引用了相应理论的经典图片，此部分只用于教学使用，不做商业用途使用。

本书由赵聚雪、杨鹏主编，赵聚雪编写了理论部分的第 1、2、3、4、6、8、9 章，以及第 10 章的 ALM 工具实训部分；杨鹏编写了理论部分的第 5、7 章，以及第 11 章的项目实例部分、附录的项目模板部分。

由于编者水平和经验有限，书中难免有欠妥和错误之处，恳请读者批评指正。

编 者

2017 年 11 月

CONTENTS

目 录

第 1 章 软件测试管理概述	1	第 3 章 测试需求分析	20
1.1 软件质量	1	3.1 测试需求	20
1.1.1 软件质量的概念	1	3.1.1 认识软件需求	20
1.1.2 软件质量模型	2	3.1.2 认识软件测试需求	21
1.1.3 软件测试与软件质量	3	3.1.3 测试需求分析知识准备	22
1.2 项目管理与软件测试管理	4	3.2 测试需求分析的步骤	28
1.2.1 项目管理	4	3.2.1 原始需求收集	29
1.2.2 软件项目管理与软件测试	5	3.2.2 原始需求整理	29
1.2.3 软件测试管理的要素	5	3.2.3 需求项分析	31
1.3 测试管理工具	5	3.2.4 建立测试需求跟踪矩阵	32
1.3.1 测试工具与测试管理	5	3.3 项目实践任务三：项目测试需求分析	32
1.3.2 测试管理工具的关系	5	第 4 章 测试计划	34
1.3.3 测试管理工具的基本功能	6	4.1 什么是测试计划	34
1.3.4 测试管理工具的选择	7	4.2 测试计划的主要内容	35
1.3.5 测试管理工具发展趋势	7	4.3 测试计划的典型模板	38
1.4 项目实践任务一：分组和项目选择	7	4.4 组织及编写测试计划	38
第 2 章 软件测试流程管理	8	4.4.1 主要任务	38
2.1 软件研发中的测试	8	4.4.2 一个有用的辅助方法：5W1H 分析法	39
2.2 软件测试模型	9	4.4.3 测试计划编制注意事项	40
2.3 软件测试流程	11	4.5 测试计划评审	40
2.4 软件测试过程资产	14	4.6 测试计划的执行和监控	41
2.5 软件测试流程的建立	16	4.7 项目实践任务四：制订项目测试计划	41
2.6 测试流程中的度量分析	17	第 5 章 测试用例设计和管理	42
2.7 项目实践任务二：实践环境准备	18	5.1 测试用例的概念和设计方法	42
		5.1.1 测试用例的概念	42
		5.1.2 测试用例的重要性	42

5.1.3 测试用例设计方法	43	7.4 测试执行的结束	80
5.2 组织及编写测试用例	45	7.5 项目实践任务六：执行测试并提交缺陷报告	81
5.2.1 测试用例的属性	45	7.6 测试报告	81
5.2.2 测试用例的详细程度	46	7.6.1 测试报告的目的及其种类	81
5.2.3 测试用例编写模板	47	7.6.2 日常型测试报告	82
5.2.4 测试用例编写指南	49	7.6.3 总结型测试报告	85
5.3 测试用例的评审	52	7.6.4 总结型测试报告典型模板	86
5.4 测试用例的管理	53	7.7 项目实践任务七：完成测试报告	86
5.4.1 测试用例的组织和维护	53	7.8 Alpha 测试与 Beta 测试的执行	86
5.4.2 测试用例的统计分析	53	7.8.1 Alpha 测试与 Beta 测试的目的	86
5.4.3 设置测试用例执行顺序	54	7.8.2 Alpha/Beta 测试过程	87
5.5 测试用例管理工具	55	第 8 章 测试组织管理	90
5.6 实践举例：手机闹钟功能测试用例	56	8.1 测试团队的建立	90
5.7 项目实践任务五：编写并管理项目测试用例	61	8.2 测试团队的组织形式	91
第 6 章 测试缺陷管理及分析	62	8.3 软件测试团队的角色配置	93
6.1 软件缺陷的概念	62	8.4 选择合适的测试人员	93
6.1.1 软件缺陷	62	8.5 测试组织管理的主要内容	94
6.1.2 软件缺陷产生的原因	63	第 9 章 测试相关的其他过程	95
6.2 软件缺陷的生命周期	63	9.1 软件配置管理	95
6.3 软件缺陷的报告	65	9.1.1 软件配置管理及其目标	95
6.3.1 软件缺陷的属性	65	9.1.2 配置管理的活动	96
6.3.2 缺陷编写典型模板	68	9.1.3 配置管理的目录结构	97
6.3.3 如何撰写一个好的缺陷报告	69	9.1.4 配置管理的工具	98
6.3.4 软件缺陷管理指南	72	9.1.5 软件测试活动涉及的配置项	98
6.4 软件缺陷的统计分析	72	9.2 软件质量保证	99
6.5 软件缺陷管理工具	75	9.2.1 质量保证的意义	99
第 7 章 测试执行和报告	76	9.2.2 质量保证的相关活动	99
7.1 测试执行	76	9.3 评审	100
7.2 测试执行的任务	76	9.3.1 评审概述	100
7.2.1 测试执行的主要任务	76	9.3.2 同行评审的活动过程	101
7.2.2 测试启动评估	77	9.4 项目实践任务八：测试项目答辩	103
7.2.3 测试用例分配	78		
7.2.4 测试用例的执行	78		
7.3 测试执行监控	79		

第 10 章 ALM 实践应用	104	10.5 ALM 测试实验室	137
10.1 ALM 安装准备	104	10.5.1 测试集的创建和维护	137
10.1.1 ALM 服务器操作系统准备	104	10.5.2 设置测试集中测试用例的执行流	138
10.1.2 实训一 ALM 安装准备	105	10.5.3 执行测试用例, 记录测试结果	140
10.2 ALM 安装和配置	106	10.5.4 分析测试执行情况并生成统计报告	142
10.2.1 安装 ALM 服务器	106	10.5.5 实训五 ALM 测试实验室	142
10.2.2 ALM 服务器部署	108	10.6 ALM 缺陷管理	143
10.2.3 ALM 站点配置	110	10.6.1 添加新的缺陷以及缺陷浏览	143
10.2.4 客户端浏览器登录服务器	116	10.6.2 修改缺陷	145
10.2.5 项目自定义配置	118	10.6.3 关联缺陷和测试用例	146
10.2.6 实训二 ALM 安装和配置	121	10.6.4 分析缺陷并生成统计报告	147
10.3 ALM 测试需求管理	122	10.6.5 实训六 ALM 缺陷管理	148
10.3.1 创建需求	122	第 11 章 项目实战样例	150
10.3.2 需求的维护	123	11.1 项目测试方案	150
10.3.3 用不同视图浏览需求	123	11.2 项目测试用例	162
10.3.4 需求分析	125	11.3 项目缺陷报告	166
10.3.5 实训三 ALM 测试需求管理	126	11.4 项目测试报告	171
10.4 ALM 测试计划管理	127	附录	180
10.4.1 测试计划树的生成	127	附录 1 某企业测试方案模板	180
10.4.2 通过视图查看测试计划树	131	附录 2 测试用例模板	186
10.4.3 测试计划树的查找、维护	132	附录 3 测试缺陷模板	187
10.4.4 测试用例和测试需求的相互关联	132	附录 4 某企业测试报告模板	188
10.4.5 构建测试用例	134	练习题	190
10.4.6 分析测试计划	135	参考文献	195
10.4.7 实训四 测试计划 (测试用例)	136		



第 1 章 软件测试管理概述

随着计算机应用的普及和深入,各行各业都已经离不开计算机和软件,人们的日常生活和工作对计算机和软件的依赖性也越来越大。除了不断追求硬件的更新换代,人们也越来越关注软件的质量问题。这一方面是因为软件产品的质量不好可能引起严重的经济损失,甚至直接伤害到生命或社会安全,比如一些加工控制软件和金融软件;另一方面是用户越来越重视使用体验,不仅要求软件产品“能工作”,而且要求产品能方便、快速地工作。

本章主要讲述什么是软件质量、软件项目管理和软件测试管理的关系,以及测试管理工具的选择。



本章学习目标

- 理解软件质量的含义以及软件测试与软件质量的关系。
- 了解软件项目管理与软件测试管理的关系。
- 理解软件测试管理的要素。
- 了解软件测试管理工具的种类、基本功能及选择依据。

1.1 软件质量

1.1.1 软件质量的概念

从现代质量管理的角度讲,质量是客户要求或者期望的有关产品或者服务的一组特性,落实到软件上,这些特性是软件的功能、性能和安全性等。

ANSI/IEEE Std 729-1983《软件工程术语的 IEEE 标准术语表》(Glossary of Software Engineering Terminology)定义软件质量为“与软件产品满足规定的和隐含的与需求能力有关的特征或特性的全体”。软件质量反映在以下 3 个方面。

- 软件需求是度量软件质量的基础,不符合需求的软件质量不合格。
 - 软件需求开发过程中,往往会有一些隐含的需求没有显式地提出来。如软件应该具备良好的可维护性。
 - 软件研发的流程定义了一组开发准则和最佳实践,用来指导软件人员用工程化的方法来开发软件。如果不遵守这些开发准则,软件质量可能就得不到保证。
- 也就是说,为满足软件各项精确定义的功能、性能需求,需要相应地给出或设计一些质量特性及其组合,作为在软件开发与维护中的重要考虑因素。如果这些质量特性及其组合都能在产品中得到满足,则这个软件产品质量就是高的。特性是软件质量的反映,软件属性可用作评价准则,度量软件属性便可知软件质量的优劣。

1.1.2 软件质量模型

软件质量是各种特性的复杂组合，它随着应用的不同而不同，随着用户提出的质量要求不同而不同。软件产品各种质量特性的组合就叫作软件质量模型。

常见的软件质量模型有 3 种：Boehm 质量模型（1976 年）、McCALL 质量模型（1978 年）、ISO 9126 质量模型（1993 年）。

1. Boehm 质量模型

Boehm 质量模型是 1976 年由 Boehm 等人提出的分层方案，将软件的质量特性定义成分层模型（见图 1-1）。

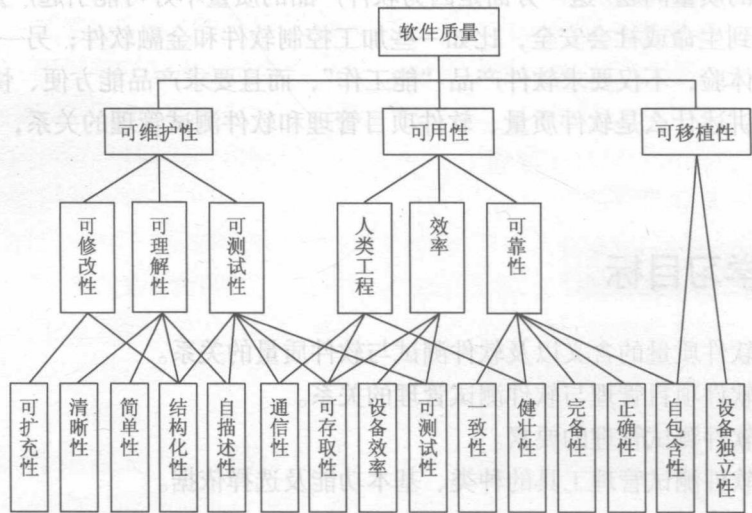


图 1-1 Boehm 质量模型

2. McCALL 质量模型

McCALL 质量模型（见图 1-2）是 1979 年由 McCall 等人提出的软件质量模型。它将软件质量的概念建立在 11 个质量特性之上，而这些质量特性分别是面向软件产品的运行、修正和转移的。



图 1-2 McCALL 质量模型

3. ISO 9126 质量模型

按照 ISO/IEC 9126-1: 2001，软件质量模型可以分为内部质量模型、外部质量模型、使用质量模型。而在质量模型中又将内部质量和外部质量分成 6 个质量特性，将使用质量分成 4 个质量特性，具体见图 1-3。

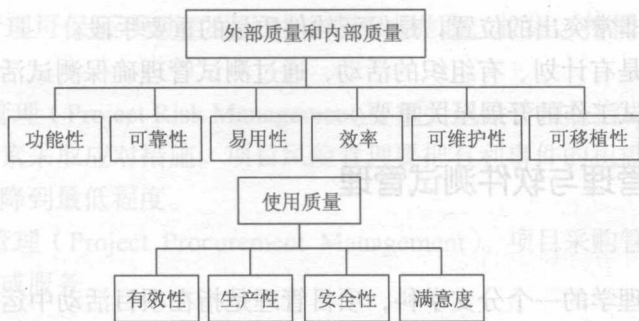


图 1-3 ISO 9126 软件质量模型

不同的软件质量模型提出了不同的软件质量特性，为了更好地理解软件质量与软件质量模型，就要弄清楚这些质量特性的含义。常见的软件质量特性及含义见表 1-1。

表 1-1 常见的软件质量特性及含义

编 号	属性名	属性含义
1	性能（Performance）	系统的响应能力，即要经过多长时间才能对某个事件做出响应，或者在某段时间内系统所能处理的事件个数
2	可用性（Availability）	系统能够正常运行的时间比例
3	可靠性（Reliability）	系统在应用或者错误面前，在意外或者错误使用的情况下维持软件系统功能特性的能力
4	健壮性（Robustness）	软件对于规范要求以外的输入情况的处理能力；健壮的系统能够对规范要求以外的输入做出判断，并且有合理的处理方式
5	安全性（Security）	系统向合法用户提供服务的同时能够阻止非授权用户使用的企图或者拒绝服务的能力
6	可修改性（Modification）	能够快速地对较高的性能价格比对系统进行变更的能力
7	可变性（Changeability）	体系结构扩充或者变更而成为新体系结构的能力
8	易用性（Usability）	衡量用户使用软件产品完成指定任务的难易程度
9	可测试性（Testability）	软件发现故障并隔离定位其故障的能力特性，以及在一定的时间或者成本前提下进行测试设计、测试执行的能力
10	功能性（Function ability）	系统所能完成期望工作的能力
11	互操作性（Inter-Operation）	系统与外界或系统与系统之间的相互作用能力

1.1.3 软件测试与软件质量

要保证软件质量，一方面要用规范化的方法和开发准则指导研发人员用工程化的方法开发软件，另一方面就是对软件进行充分的测试。

软件测试是软件质量控制中的关键活动，是软件质量保证的关键步骤。软件测试在软

件生存周期中占有非常突出的位置，是保证软件质量的重要手段。

软件测试活动是有计划、有组织的活动，通过测试管理确保测试活动的顺利开展，项目的测试管理对测试工作的开展至关重要。

1.2 项目管理与软件测试管理

1.2.1 项目管理

项目管理是管理学的一个分支学科，项目管理是指在项目活动中运用专门的知识、技能、工具和方法，使项目能够在有限资源限定的条件下，实现或超过设定的需求和期望的过程。

项目管理知识体系（Project Management the Body of Knowledge, PMBOK）把项目管理分为 5 个过程及 9 个知识领域。管理过程包括输入、输出、所需工具和技术。各个过程通过各自的输入和输出相互联系，构成整个项目管理活动。

1. 项目管理的 5 个过程

① 启动。成立项目组，开始项目或进入项目的新阶段。启动是一种认可过程，用来正式认可一个新项目或新阶段的存在。

② 计划。定义和评估项目目标，选择实现项目目标的最佳策略，制订项目计划。

③ 执行。调动资源，执行项目计划。

④ 控制。监控和评估项目偏差，必要时采取纠正行动，保证项目计划的执行，实现项目目标。

⑤ 结束。正式验收项目阶段，使其按程序结束。

2. 项目管理的 9 个知识领域

① 项目集成管理（Project Integration Management）。项目集成管理是为了正确地协调项目各组成部分而进行的各个过程的集成，是一个综合性过程。其核心就是在多个互相冲突的目标和方案之间做出权衡，以便满足项目利害关系者的要求。

② 项目范围管理（Project Scope Management）。项目范围管理就是确保项目完成全部规定要做的工作，最终成功地达到项目的目的。基本内容是定义和控制列入或未列入项目的事项。

③ 项目时间管理（Project Time Management）。其作用是保证在规定时间内完成项目。

④ 项目费用管理（Project Cost Management）。项目费用管理，是为了保证在批准的预算内完成项目所必需的诸过程的全体。

⑤ 项目质量管理（Project Quality Management）。项目质量管理，是为了保证项目能够满足原来设定的各种要求。

⑥ 项目人力资源管理（Project Human Resource Management）。项目人力资源管理，是为了保证最有效地使用参加项目者的个别能力。

⑦ 项目沟通管理（Project Communications Management）。项目沟通管理，是在人、思想和信息之间建立联系，这些联系对于取得成功是必不可少的。参与项目的每一个人都必须用项目“语言”进行沟通，并且要明白个人所参与的沟通将会如何影响到项目的

整体。项目沟通管理可保证项目信息及时并准确地提取、收集、传播、存储以及最终进行处置。

⑧ 项目风险管理 (Project Risk Management)。项目风险管理是指识别、分析不确定的因素,并对这些因素采取应对措施。项目风险管理要把有利事件的积极结果尽量扩大,而把不利事件的后果降到最低程度。

⑨ 项目采购管理 (Project Procurement Management)。项目采购管理,是为了从项目组织外部获取货物或服务。

1.2.2 软件项目管理与软件测试项目管理

软件项目是指软件工程类的项目,具体来说,软件项目管理的根本目的是为了让软件项目尤其是大型项目的整个软件生命周期(从分析、设计、编码到测试、维护的全过程)都能在管理者的控制之下,以预定成本按期、按质地完成软件并交付用户使用。

软件测试项目是软件工程项目中的一种,是以软件测试为主要任务的项目。软件项目管理和软件测试项目管理与一般的工程项目管理有共性,但是在实际开展项目管理时因任务特点不同又有其特殊性。

1.2.3 软件测试管理的要素

测试活动贯穿于软件产品的整个生命周期,测试管理贯穿于测试的全过程。软件测试管理着眼于对软件测试的流程进行策划组织,对测试实施中的所有元素进行管理控制,确保测试活动按时保质开展。测试管理主要涉及以下内容:

- 测试过程和资产管理;
- 测试团队管理;
- 测试需求管理;
- 测试计划管理(测试规划);
- 测试用例管理(测试设计);
- 测试缺陷管理;
- 测试工具选择和使用;
- 测试执行和汇报管理。

1.3 测试管理工具

1.3.1 测试工具与测试管理工具的关系

软件测试工具可提高软件测试工作的效率。软件测试工具分为自动化软件测试工具和测试管理工具,所以测试管理工具是测试工具的一种。

自动化软件测试工具存在的价值是为了提高测试效率,用软件来代替一些人工输入,提高测试用例的复用率。常见的自动化软件测试工具主要有单元测试工具、性能测试工具和功能测试工具。比如功能测试工具 QTP (Quick Test Professional)、性能测试工具 Jmeter 和 LoadRunner。

1.3.2 测试管理工具的基本功能

完整的测试管理工具应该能对测试整个流程的各个环节进行管理。对于测试人员来说，测试管理工具能够管理测试过程中测试人员的日常活动，其主要功能包括以下几种：

- 用户及权限管理；
- 测试项目的创建；
- 测试项目需求管理；
- 测试任务分配和实施；
- 测试项目缺陷管理；
- 测试数据收集；
- 测试项目数据分析及统计和报告生成；
- 测试项目用例管理；
- 测试执行管理；
- 测试文档管理。

1.3.3 测试管理工具的来源和分类

测试管理工具有开源的，可以自主开发，也可以购买。测试管理工具的分类如下。

1. 专项测试管理工具

这类工具可管理软件测试中的某个内容，如缺陷管理工具、用例管理工具，比如 BUGzilla 和 BUGfree。BUGzilla 是一个开源的缺陷跟踪系统（BUG-Tracking System），它可以管理软件开发中缺陷的提交（new）、修复（resolve）、关闭（close）等整个生命周期。BUGfree 是一款简单、实用、免费并且开放源代码的一个 Bug 管理系统，不过目前已经不提供更新和技术支持了。

2. 专门的测试管理工具

这类工具对测试的整个流程进行管理，比如 IBM Rational Quality Manager、HP ALM（Application Lifecycle Management）等。ALM 软件是惠普公司的一款高端商业软件，提供了需求管理、缺陷管理、测试用例管理、测试执行管理和各种分析报告管理。

3. 开发和测试都包含的项目管理工具

专门的测试管理工具主要用于第三方软件测试机构，以及软件开发部门和测试部门相对独立的公司。大部分时候，开发和测试属于同一个团队，此时研发团队会使用覆盖整个研发周期的软件产品。这些产品或者是公司自己开发的，或者是从市场购买的，也可能是开源的软件。

比如，禅道项目管理软件是一款国产的开源软件。该软件集产品管理、项目管理、质量管理、文档管理、组织管理和事务管理于一体，是一款功能完备的项目管理软件。

4. 其他可用于测试管理的工具

小型项目团队、初创的团队、学生课程设计团队往往直接利用 Office 办公软件完成软件测试的管理，包括 Office Project、Word、Excel。对于软件测试中的文档管理，则可以

借助于一些文件管理软件，比如 TortoiseSVN、TortoiseHg。

1.3.4 测试管理工具的选择

在进行测试管理工具的选择时，要综合考虑项目大小、团队规模、团队性质、成本预算等因素。工具是一个载体，重要的是按照流程开展工作。

对于新创业项目、学生项目等团队规模比较小的情况，可以选择开源工具或 Office 办公软件，节省成本，简单易用；对于第三方测试团队，则可以选用专门的测试工具；对于研发测试一体化的团队，则可以采用完整的项目管理工具。

1.3.5 测试管理工具发展趋势

软件测试管理工具发展的趋势如下。

① 与其他软件自动化测试工具集成，比如在软件测试用例的管理中，用例可能是 LoadRunner 的性能测试脚本，也可能是 QTP 的功能测试脚本，也可能是需要手工测试的用例。目前的测试管理工具倾向于能直接启动测试用例并执行，这就要求测试管理工具与 LoadRunner、QTP 等自动化测试工具有很好的衔接。

② 与软件开发其他环节的集成越来越紧密，比如软件测试管理工具中的集成版本管理功能。

③ 基于云计算的测试管理工具，比如，QASymphony 开发的 QTest 是基于云计算的测试管理工具，有各种典型的关键特性。QTest 在连接器的帮助下，可以集成 JIRA 整个端到端质量的解决方案。它还集成了其他工具，比如 BUGzilla、FogBUGz、Rally 等。

1.4 项目实践任务一：分组和项目选择

实践任务：

- 所有学员自由组合成测试小组（3~4 人），给出小组名单，并指定组长。
- 选择被测试的软件项目。
- 将分组名单和选定的项目提交给学习委员汇总。

实践指导：

选择的被测试项目可以是企业项目、教学实训项目、学生开发的参赛和课程设计项目、网络开源项目等，软件架构不限。

1. V 模型

V 模型（见图 2-2）是软件测试模型中的一类模型，它强调开发阶段与测试阶段的对应关系。模型明确标识了测试活动同开发阶段的关系，即开发阶段从需求分析开始，经过系统分析、详细设计、编码，然后是测试活动。测试活动对应开发的 4 个阶段，即需求分析、系统分析、详细设计和验收测试。

在这个模型中，把测试作为编码之后的最后一个活动，需求分析等前期产生的错误直到后期的验收测试才能发现。测试活动在编码之后，并且只对代码测试，未能体现尽早测试。



第②章 软件测试流程管理

要开展软件测试管理,首先要建立软件测试的流程。本章主要讲述软件开发过程和测试过程是如何交互的,典型的测试模型有哪些,软件测试的一般流程是怎样的,如何建立软件测试的流程,以及软件测试流程中涉及的资产、度量分析。



本章学习目标

- 理解研发过程和测试过程的交互过程。
- 理解典型的软件测试模型。
- 理解软件测试的一般流程。
- 了解建立一个测试流程应该包含的内容。
- 了解测试中涉及的文档资产及度量分析。

2.1 软件研发中的测试

软件测试和软件开发一样,都遵循软件工程的原理。测试和开发是密切相关的。测试活动是贯穿于软件项目开发的全过程,和开发活动交互开展。

图 2-1 描述了软件项目开发中的测试环节及相应的测试活动。

① 软件需求完成后,需要进行需求的评审,此时测试人员可以参与需求的评审。当需求确定后,测试人员可以开始进行系统测试方案及计划的制订。

② 软件项目总体设计方案完成后,测试人员可以开始进行集成测试方案及计划的制订。

③ 详细设计完成后,测试方可以开始进行模块测试方案及计划的制订。

④ 单元测试和编码一般是同步的,由开发人员自己完成。

⑤ 整个模块开发完成后,测试人员开始进行模块测试。当然在这之前,所有的模块测试用例已经准备完毕。

⑥ 模块测试后是集成测试和系统测试。

⑦ 软件运行维护期间则要对运行期间发现的问题进行回归测试。

从图 2-1 可以看出,在软件项目开发过程中不能把测试理解为开发后期的一个活动,它是贯穿于整个开发过程的。

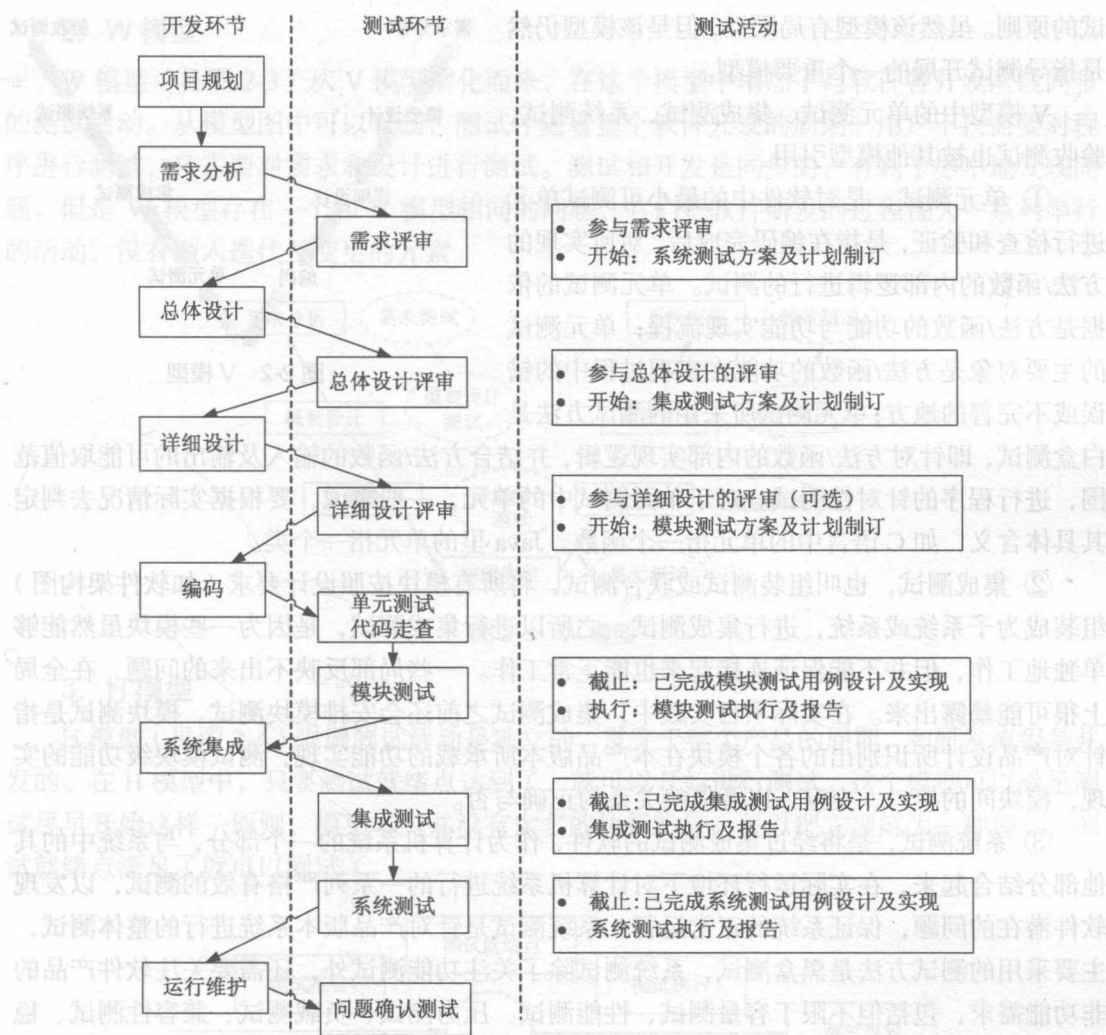


图 2-1 软件开发中的测试环节及测试活动

2.2 软件测试模型

在实践中产生了很多的测试模型。这些测试模型明确了测试和开发之间的关系。主要的测试模型有 V 模型、W 模型和 H 模型。

1. V 模型

V 模型（见图 2-2）是软件测试模型中的一个经典模型，它发展自软件研发的瀑布模型，模型明确标识了测试过程中存在的测试阶段，以及测试阶段与开发阶段之间的关系。从图 2-2 可以看到，项目研发中的开发活动是从需求分析到概要设计，之后到详细设计，再到编码，然后是测试活动。测试活动对应开发的 4 个阶段，分别是单元测试、集成测试、系统测试和验收测试。

在这个模型中，把测试作为编码之后的最后一个活动，需求分析等前期产生的错误直到后期的验收测试才能发现。测试活动在编码之后，并且只对代码测试，未能体现尽早测