

组稿编辑：油俊伟
责任编辑：油俊伟
装帧设计：凡一



「十四五」职业教育国家规划教材



“十四五”职业教育国家规划教材

面向高等职业院校基于工作过程项目式系列教材
企业级卓越人才培养解决方案规划教材

软件测试项目实战

(第2版)

天津滨海迅腾科技集团有限公司◎编著
杨鹏 翟亚峰◎主编

软件测试项目实战 第2版

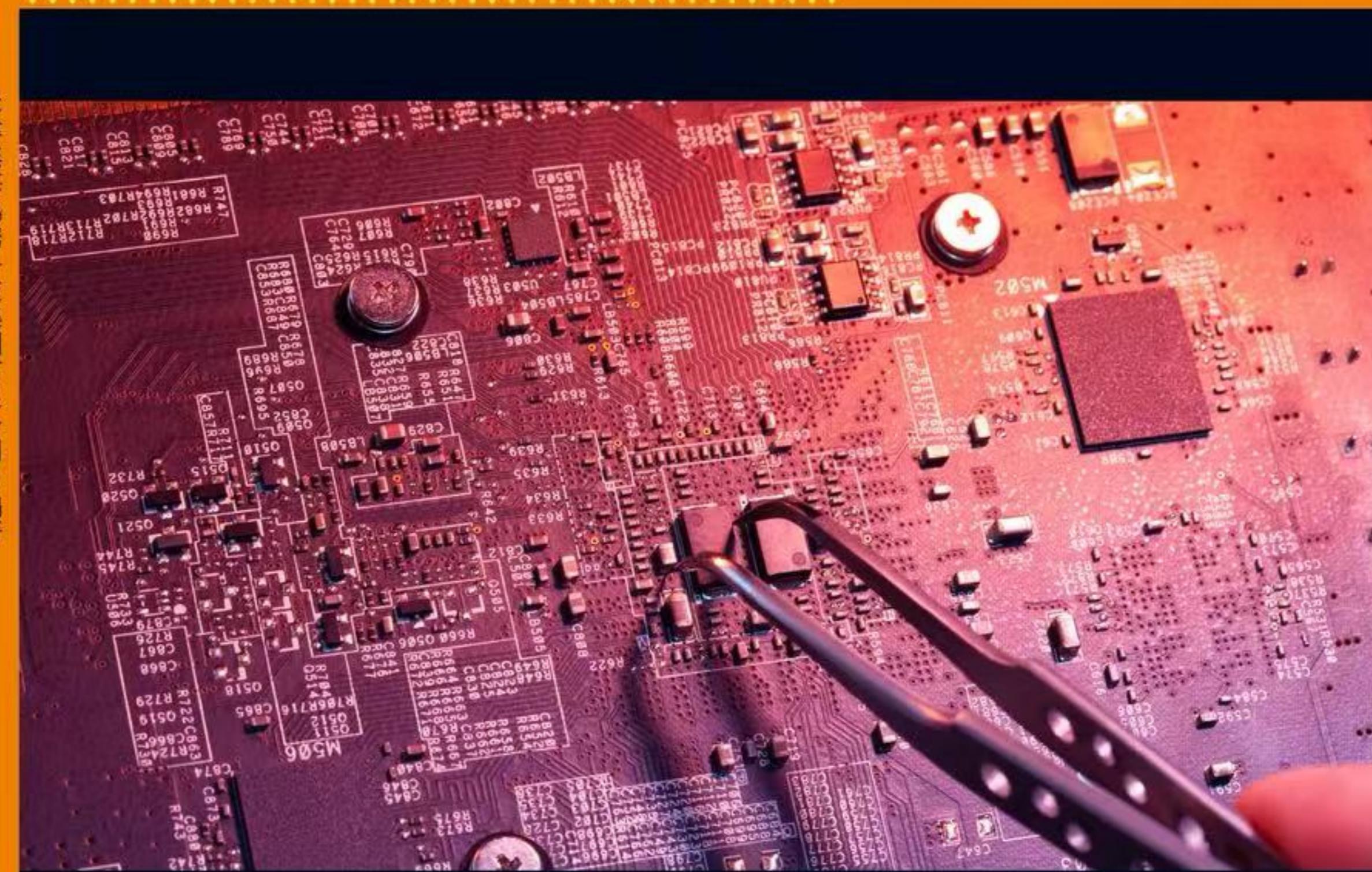
天津滨海迅腾科技集团有限公司

杨鹏

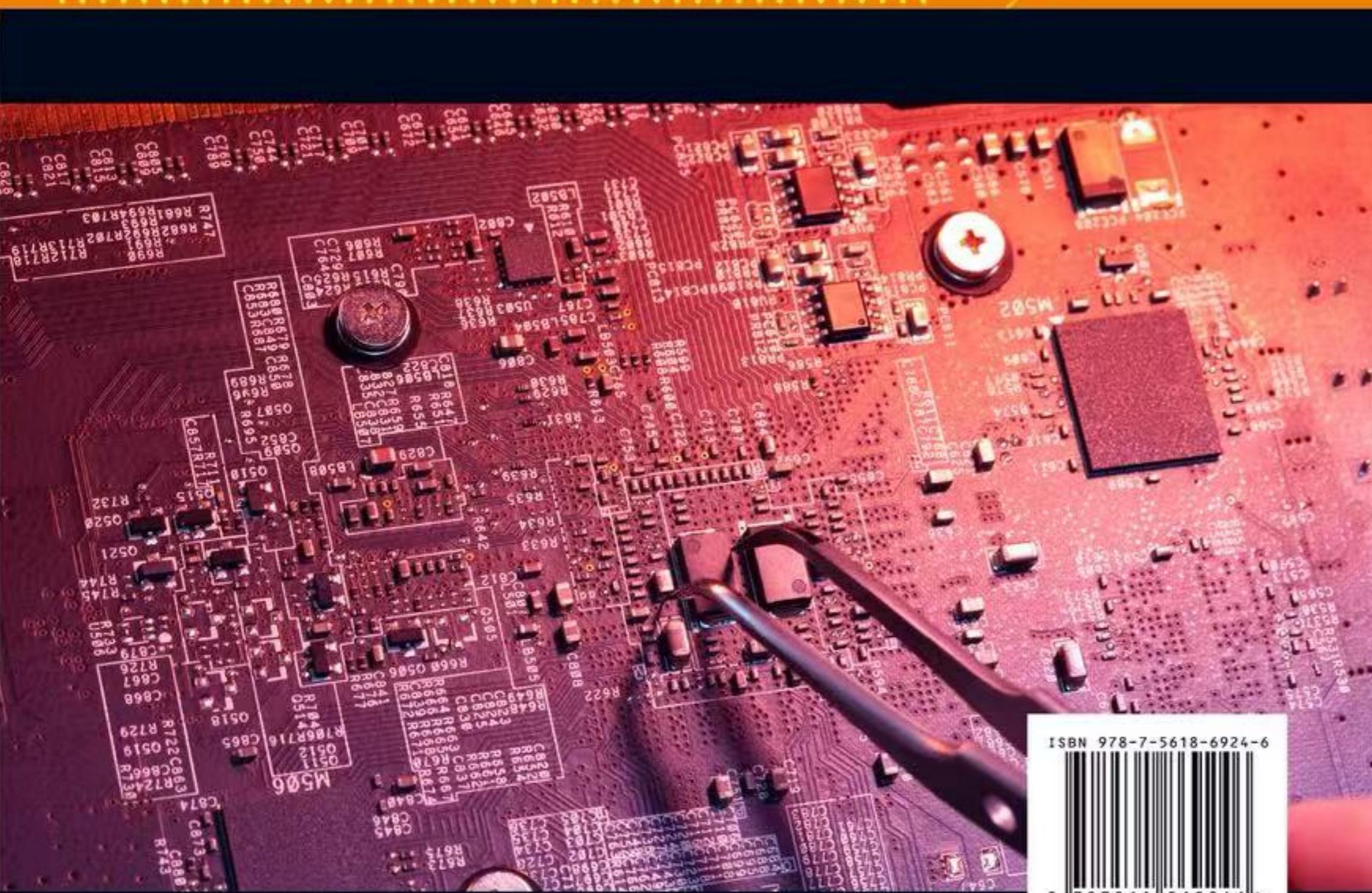
翟亚峰

◎主编

天津大学出版社



天津大学出版社
TIANJIN UNIVERSITY PRESS





“十四五”职业教育国家规划教材

面向高等职业院校基于工作过程项目式系列教材
企业级卓越人才培养解决方案规划教材

软件测试项目实战

(第 2 版)

天津滨海迅腾科技集团有限公司 编著

杨鹏 翟亚峰 主编

图书在版编目（CIP）数据

软件测试项目实战 / 天津滨海迅腾科技集团有限公司编著；杨鹏，翟亚峰主编. —天津：天津大学出版社，2021.5（2026.2重印）
面向高等职业院校基于工作过程项目式系列教材 企
业级卓越人才培养解决方案规划教材

ISBN 978-7-5618-6924-6

I . ①软… II . ①天… ②杨… ③翟… III . ①软件—
测试—高等职业教育—教材 IV . ①TP311.5

中国版本图书馆CIP数据核字(2021)第078950号

RUANJIAN CESHI XIANGMU SHIZHAN

主 编：杨 鹏 翟亚峰
副主编：史玉琢 孟思明 徐书欣
王文立 郭 惠 卜银娜
参 编：窦珍珍 李肖霆

出版发行 天津大学出版社
地 址 天津市卫津路92号天津大学内（邮编：300072）
电 话 发行部：022-27403647
网 址 www.tjupress.com.cn
印 刷 廊坊市海涛印刷有限公司
经 销 全国各地新华书店
开 本 787mm×1092mm 1/16
印 张 17.75
字 数 463千
版 次 2021年5月第1版 2025年1月第2版
印 次 2026年2月第4次
定 价 66.00元

凡购本书，如有缺页、倒页、脱页等质量问题，烦请与我社发行部门联系调换

版权所有 侵权必究

前　　言

党的二十大报告指出：“坚持把发展经济的着力点放在实体经济上，推进新型工业化，加快建设制造强国、质量强国、航天强国、交通强国、网络强国、数字中国。”软件作为新一代信息技术的灵魂，是中国数字经济发展的基础，也是制造强国、网络强国和数字中国等重大战略的支撑。为了提高软件质量，降低软件开发成本，软件测试成为软件与信息服务产业中的共性技术。

本书依据高职软件测试课程教学标准，对接软件测试岗位技能要求、《计算机软件测试规范》国家标准、全国职业院校技能大赛软件测试竞赛规程等，设计覆盖对应知识、技能与素质要求的教材内容体系，有机融入岗位、课程、竞赛和1+X证书的要求，反映软件测试主流技术、发展趋势及相关新技术、新规范和新标准。坚持工学结合、理实一体的编写理念，基于软件测试的工程化特征，根据软件测试岗位典型工作任务，分析对应的知识、技能与素质要求，确立教材知识、技能、素养点组成，甄选、整合、序化内容。采用项目任务驱动编写方式，选取来自企业的真实项目案例，以典型工作任务为载体组织教学单元，将完成任务所需的相关知识、技能和素养构建于项目之中，在完成项目任务的过程中培养软件测试生产过程的关键能力与综合素养。

本书全面贯彻党的二十大精神，坚持立德树人、以生为本，将价值塑造、知识传授和能力培养有机结合。提高软件质量是推进实施国家软件发展战略的重要内容，软件测试是保障软件质量的主要手段。通过深入学习软件测试，培养学习者质量意识、科学思维以及运用所学知识实现科技报国理想的爱国担当。通过引入软件测试职场工作场景，将职业技能、职业理想和职业道德教育融通，注重学思结合、知行统一，引导学习者理解并践行追求品质、精益求精的工匠精神，增强勇于探索的创新精神及善于解决问题的综合能力。

本书划分为初识软件测试、软件开发流程、软件测试的管理、黑盒测试、白盒测试、性能测试、安全测试和自动化测试八个教学项目，以完成项目案例的工作任务为主线，重组知识点与技能点并由此构建相应教学项目的内容，充分满足项目学习、案例学习的要求。本书设计了层级递进的导学路径，每个教学项目按照学习目标、学习路径、任务描述、任务技能、任务实施、任务总结和任务习题进行编排。学习目标即要达成的任务目标，学习路径以思维导图形式给出任务指引，任务描述包含情境导入和功能描述，任务技能讲解完成任务所需的知识技能，任务实施给出完成本项任务的详细过程和具体步骤，任务总结是基于本项任务实施的经验总结，任务习题用于课后复习、拓展和提升。

本书内容理论性与实践性兼具，理论内容阐述简明，实例操作讲解细致、步骤清晰，实现了理实结合。同时配套了教学课件、教学视频、实训源代码包和习题等多样化教学资源，可获得可听、可视、可练、可交互的数字化课程资源支持，可随时随地支持线上线下混合式教学。

本书由6所高职院校7名专业教师、1家科技集团3名技术人员合作开发，充分发挥校企双方经验与资源互补优势。广州番禺职业技术学院杨鹏教授与天津滨海迅腾科技集团有

限公司翟亚峰共同担任主编,天津商务职业技术学院史玉琢、广州铁路职业技术学院孟思明、许昌职业技术学院徐书欣、周口职业技术学院王文立、山西旅游职业学院郭惠、许昌职业技术学院卜银娜担任副主编,天津滨海迅腾科技集团有限公司窦珍珍和李肖霆参编。其中,项目一由史玉琢负责编写,项目二由孟思明负责编写,项目三由王文立负责编写,项目四由郭惠负责编写,项目五由徐书欣负责编写,项目六由卜银娜负责编写,项目七和项目八的编写以及本书的整体编排规划由杨鹏负责,附件一至五以及全书的案例和实训源代码由翟亚峰、窦珍珍和李肖霆负责编写整理。

本书是育训一体、项目式新形态教材,是高职本科、专科院校软件工程、软件技术等相关专业软件测试课程教材,也可作为行业企业的软件测试培训教材。

由于编者水平有限,书中难免出现错误与不足,恳请读者批评指正。

编者

2022年11月

目 录

项目一 初识软件测试.....	1
学习目标.....	1
学习路径.....	1
任务描述.....	2
任务技能.....	2
任务实施.....	17
任务总结.....	26
任务习题.....	26
项目二 软件开发流程.....	28
学习目标.....	28
学习路径.....	28
任务描述.....	29
任务技能.....	29
任务实施.....	47
任务总结.....	57
任务习题.....	57
项目三 软件测试的管理.....	58
学习目标.....	58
学习路径.....	58
任务描述.....	59
任务技能.....	59
任务实施.....	68
任务总结.....	72
任务习题.....	72
项目四 黑盒测试.....	74
学习目标.....	74
学习路径.....	74
任务描述.....	75
任务技能.....	75
任务实施.....	93
任务总结.....	101
任务习题.....	101

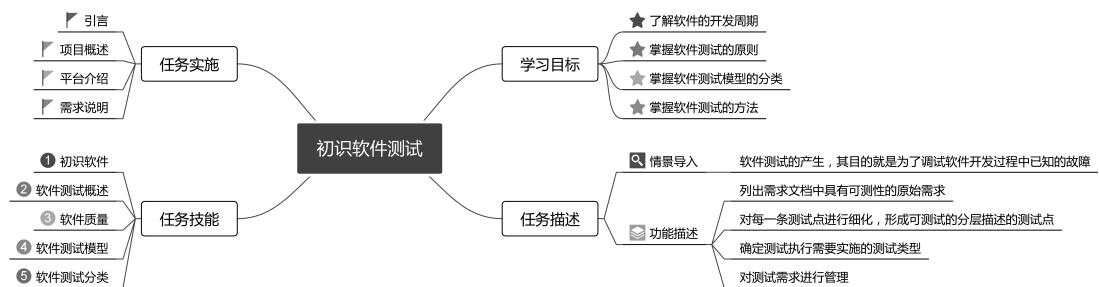
项目五 白盒测试	102
学习目标	102
学习路径	102
任务描述	103
任务技能	103
任务实施	123
任务总结	130
任务习题	130
项目六 性能测试	131
学习目标	131
学习路径	131
任务描述	132
任务技能	132
任务实施	150
任务总结	165
任务习题	166
项目七 安全测试	167
学习目标	167
学习路径	167
任务描述	168
任务技能	168
任务实施	191
任务总结	205
任务习题	205
项目八 自动化测试	206
学习目标	206
学习路径	206
任务描述	207
任务技能	207
任务实施	219
任务总结	228
任务习题	229
附件一 OA 协同办公管理系统需求说明书	230
附件二 OA 协同办公管理系统测试方案	245
附件三 OA 协同办公管理系统测试用例	253
附件四 OA 协同办公管理系统性能	260
附件五 OA 协同办公管理系统 Bug 清单	278

项目一 初识软件测试



通过本项目的学习,了解软件的概念与分类,重点学习软件开发模型、软件测试、软件测试模型以及软件测试分类,具有区分软件测试与质量保证的能力。在学习过程中:

- 了解软件的开发周期。
- 掌握软件测试的原则。
- 掌握软件测试模型的分类。
- 掌握软件测试的方法。





【情境导入】

目前软件系统越来越复杂,一个软件被划分为多个模块,由不同的软件工程师负责编写,一位工程师可能只负责其中一个模块,其对全局设计的理解就会受限制。此时,如果运行软件就容易产生很多的错误。而软件测试的主要目的就是为了调试软件在开发过程中已知的故障。

【功能描述】

- 列出需求文档中具有可测性的原始需求。
- 对每一条测试点进行细化,形成可测试的分层描述的测试点。
- 确定测试执行需要实施的测试类型。
- 对测试需求进行管理。



技能点一 初识软件

1. 软件概念

提起软件,估计大家都不会陌生,比如常用的Office、微信、QQ等都统称为软件,软件是计算机系统中与硬件相互依存的一部分,它是程序、数据及其相关文档的完整集合。其中,程序是按事先设计的功能和性能要求执行的指令序列;数据是使程序能正常操纵信息的数据结构;文档是与程序开发、维护和使用有关的图文材料。

在使用软件过程中总结出软件具有如下特点。

- ①软件是一种逻辑实体,而不是具体的物理实体。因而它具有抽象性。
- ②软件的产生与硬件不同,它没有现实的制造过程。对软件的质量控制,必须着重在软件开发方面下功夫。
- ③任何机械、电子设备在运行和使用中,其失效率大都遵循如图1-1(a)所示的U形曲线。

线(即浴盆曲线)。而软件的情况与此不同,因为它不存在磨损和老化问题。但是它存在退化问题,必须要多次对其进行修改(维护),如图 1-1(b)所示。

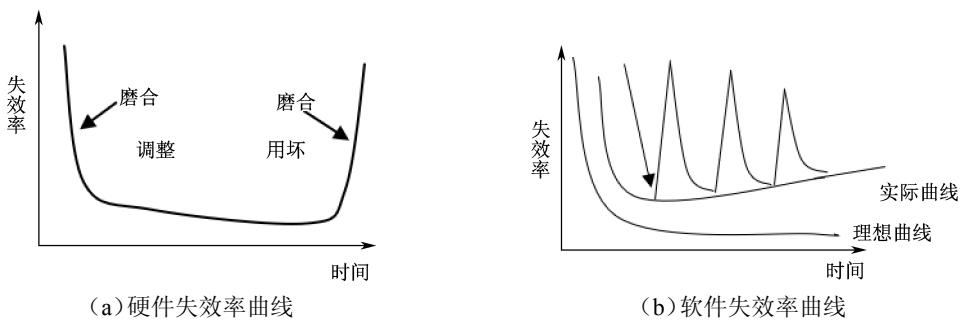


图 1-1 失效率曲线

④软件的开发和运行常常受到计算机系统的限制,对计算机系统有着不同程度的依赖性。为了解除这种依赖性,在软件开发中提出了软件移植的问题。

⑤软件的开发至今尚未完全摆脱手工的开发方式。

⑥软件本身是复杂的。软件的复杂性可能来自它所反映的实际问题的复杂性,也可能来自程序逻辑结构的复杂性。

⑦软件的开发成本相当高。软件的开发工作需要投入大量的、复杂的、高强度的脑力劳动,它的成本是比较高的。

⑧相当多的软件工作涉及社会因素。许多软件的开发和运行涉及机构、体制及管理方式,甚至涉及人的观念和人们的心理等问题。它们直接影响项目的成败。

2. 软件分类

软件是多种多样的,可以根据功能、用途等对软件进行分类,常见的软件一般按照功能、规模、工作方式、服务对象等进行分类。

(1) 按软件功能进行划分

①系统软件:能与计算机硬件紧密配合,使计算机系统各个部件、相关的软件和数据协调、高效地工作的软件。例如,操作系统、数据库管理系统、设备驱动程序以及通信处理程序等。

②支撑软件:是协助用户开发软件的工具性软件,其中包括帮助程序员开发软件产品的工具,也包括帮助管理人员控制开发进程的工具。

③应用软件:是在特定领域内开发,为特定目的服务的一类软件。

(2) 按软件规模进行划分

按开发软件所需的人力、时间以及完成的源程序行数,可确定 6 种不同规模的软件,如表 1-1 所示。

表 1-1 软件规模的分类

类别	参加人员数	研制期限	产品规模(源程序行数)
微型	1	1~4 周	0.5 k

续表

类别	参加人员数	研制期限	产品规模(源程序行数)
小型	1	1~6 月	1 k~2 k
中型	2~5	1~2 年	5 k~50 k
大型	5~20	2~3 年	50 k~100 k
甚大型	100~1 000	4~5 年	1 M (=1 000 k)
极大型	2 000~5 000	5~10 年	1 M~10 M

规模大、时间长、很多人参加的软件项目，其开发工作必须有软件工程的知识做指导。规模小、时间短、参加人员少的软件项目也要了解软件工程的概念，遵循一定的开发规范。其基本原则是一样的，只是对软件工程技术依赖的程度不同而已。

(3) 按软件工作方式划分

①实时处理软件：指在事件或数据产生时，立即予以处理，并及时反馈信号，控制需要监测和控制的过程的软件，主要包括数据采集、分析、输出 3 个部分。

②分时软件：允许多个联机用户同时使用计算机。

③交互式软件：能实现人机通信的软件。

④批处理软件：把一组输入作业或一批数据以成批处理的方式一次运行，按顺序逐个处理的软件。

(4) 按软件服务对象进行划分

①项目软件：也称定制软件，是受某个特定客户（或少数客户）的委托，由一个或多个软件开发机构在合同的约束下开发出来的软件。例如军用防空指挥系统、卫星控制系统。

②产品软件：是由软件开发机构开发出来直接投入市场，或是为大批量用户服务的软件。例如，文字处理软件、文本处理软件、财务处理软件、人事管理软件等。

3. 软件开发周期

软件和其他产品一样，都有一个从“出生”到“消亡”的过程，这个过程称为软件的生命周期。在软件的生命周期中，软件测试是一个非常重要的环节。

软件生命周期分为多个阶段，每个阶段都有明确的任务，这样就使得结构复杂、管理复杂的软件开发变得容易控制和管理。通常情况下，软件周期一般分为六个阶段，分别是制订计划、需求分析、软件设计、程序编写、软件测试、软件运行和维护。

①制订计划：确定待开发软件系统的总目标，在功能、性能、可靠性以及接口等方面提出要求；研究完成该项目软件任务的可行性，探究解决问题的可能方案；制订完成开发任务的实施计划，连同可行性研究报告提交管理部门审查。

②需求分析：对待开发软件提出的需求进行分析并给出详细的定义。编写软件需求说明书及初步的用户手册，提交管理机构评审。

③软件设计：把已确定的各项需求转换成一个相应的体系结构，进而对每个模块要完成的工作进行具体的描述。编写设计说明书，提交评审。

④程序编写：把软件设计转换成计算机可以识别的程序代码。

⑤软件测试：在设计测试用例的基础上检验软件的各个组成部分。

⑥软件运行和维护:已交付的软件正式投入使用,并在运行过程中进行适当的维护。

软件生命周期模型是从软件项目需求定义直至软件经使用后废弃为止,跨越整个生命周期的系统开发、运作和维护实施的全部过程、活动和任务的结构框架。

4. 软件开发模型

软件测试工作与软件开发模型息息相关,在不同的软件开发模型中,测试的任务和作用也不相同,因此测试人员要充分了解软件开发模型,以便找准自己在其中的定位和任务。

(1) 瀑布模型

瀑布模型规定了各项软件工程活动,包括制订开发计划、进行需求分析和说明、软件设计、程序编码、测试及运行维护,如图 1-2 所示。并且瀑布模型规定了它们自上而下,互相衔接的固定次序,如同瀑布流水,逐级下落。

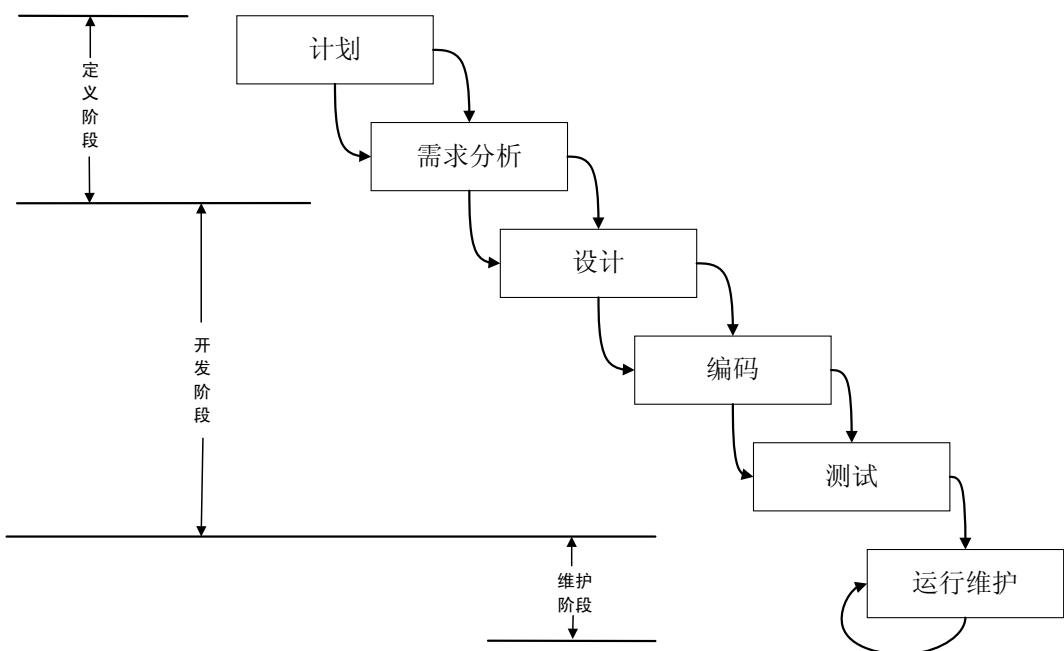


图 1-2 瀑布模型

然而软件开发的实践表明,上述各项活动之间并非完全自上而下,呈线性图式。实际情况是,每项开发活动均处于一个质量环(输入—处理—输出—评审)中。只有当其工作得到确认,才能继续进行下一项活动(在图 1-2 中用向下的箭头表示);否则返工(在图 1-2 中用向上的箭头表示)。

瀑布模型是一个很好的想法,但不切实际。即使能确定开发需要的时间,在没有考虑问题的细节时,是不可能预知在开发过程中会遇到什么困难的,比如设计缺陷、技术问题等。所以,任何阶段都可能比预期的时间长。另外,工作也可能会扩张,以充分利用可用的时间,这样,某个问题之前的各个阶段的修改很可能会浪费大量的时间,最终结果是整个项目都得延迟交付。

(2)螺旋模型

对于复杂的大型软件，开发一个原型往往达不到要求。螺旋模型将瀑布模型与演化模型结合起来，并且加入两种模型均忽略了的风险分析。螺旋模型沿着螺线旋转，如图 1-3 所示，在笛卡儿坐标系的四个象限上分别表达了四个方面的活动，即：

- ①制订计划——确定软件目标，选定实施方案，弄清项目开发的限制条件；
- ②风险分析——分析所选方案，考虑如何识别和消除风险；
- ③实施工程——实施软件开发；
- ④客户评估——评价开发工作，提出修正建议。

沿螺线自内向外每旋转一圈便开发出一个更为完美的新的软件版本。

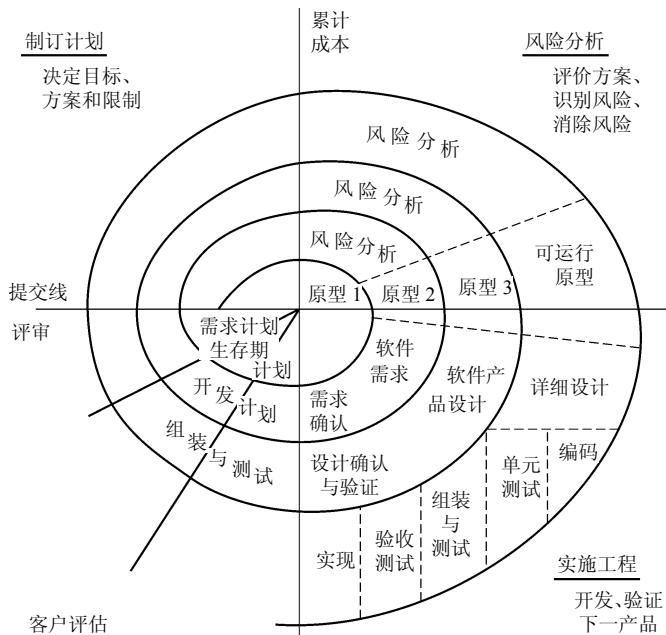


图 1-3 螺旋模型

在完成一次循环后，就增加了对问题域和解决方案的理解，还可以使用户参与进来，这样用户可以对最终系统中包含的事务或功能提出修正建议。有了新的知识库，就可以再次执行一遍开发过程。现在我们更新了需求，更深地理解或更正了分析，加强系统设计，给子系统添加些界面，再编写更多的代码，更多地满足需求，经过三四遍的开发过程，完成系统后，就可以全面测试和部署系统。

与瀑布模型比较，似乎螺旋模型的活动比较少。它使用户参与了整个生命周期，每个人都可以看正在进行的工作；它调整改动的次数和每次改动所花的时间较少。

但是螺旋模型的开发方式也不是完美的，只是把瀑布模型的开发过程重复进行了三四次，如果发现错误，就必须在下一遍开发过程中才能更正它们。因此，螺旋模型本身不是非常有用，需要和其他的模式结合起来一起使用。

(3)喷泉模型

喷泉模型对软件复用和生命周期中多项开发活动的集成提供了支持，主要支持面向对

象的开发方法,“喷泉”一词本身体现了迭代和无间隙特性。系统某个部分常常重复工作多次,相关功能在每次迭代中随之加入演进的系统。所谓无间隙是指在开发活动,即分析、设计和编码之间不存在明显的边界。喷泉模型如图 1-4 所示。

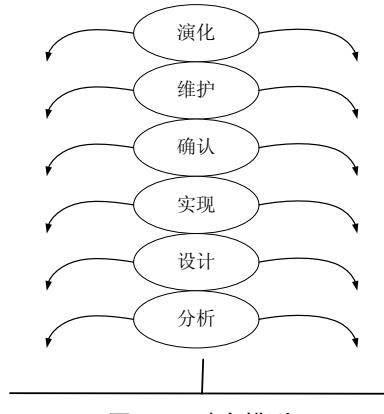


图 1-4 喷泉模型

在喷泉模型中,允许重复开发中的各个阶段根据需要前后移动或者来回移动。

技能点二 软件测试概述



课程思政

1. 什么是软件测试

软件测试是指在规定条件下对程序进行操作的过程,在此过程中测试员可以发现程序的错误、软件的质量以及软件是否满足用户需求等。在软件开发到软件发布这一过程中,软件测试是一个很重要的环节,若测试没有做好,会对公司造成很大的损失。如软件即将销售时发现具有严重问题,从而推延发布日期,失去市场机会;或者软件发布后,用户发现了不能容忍的错误,引起索赔、法律纠纷等。软件测试不仅能提高软件质量,还可以预防问题,减少程序中未发现的缺陷等。软件发布前期工程如图 1-5 所示。

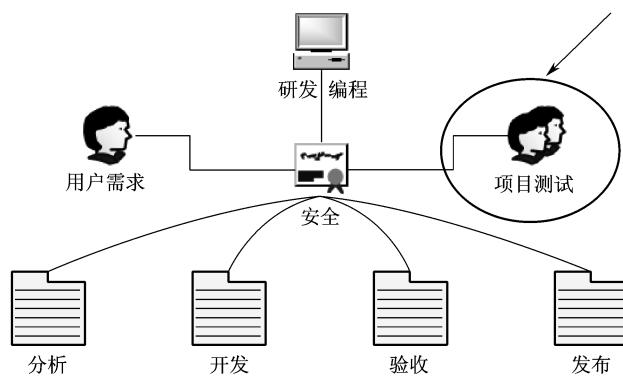


图 1-5 软件发布前期工程

一款软件在开发过程中往往大错误不多,最多的就是隐藏的 Bug(漏洞),测试员必须心细,才能找出软件中的 Bug。对一个好的软件测试员来说,软件有很多地方需要测试,除了程序的问题,影响客户体验的问题也需要通过测试找出,避免软件发布后用户无法使用,造成损失。

软件测试的作用就是发现并指出问题,其只能证明软件存在错误,但不能证明软件没有错误。软件公司对软件开发组的要求是在指定的时间、合理的预算下,提交一个可以交付的软件。测试员的工作就是把软件的错误控制在可以进行产品交付的范围内,但该软件并不是没有错误,软件测试不可能无休止地进行下去,由于不同软件测试的成本不同,因此要把错误控制在一个合理的范围之内。这也要求在项目计划时,给测试留出足够的时间和经费,仓促地测试或者由于项目提交计划的压力而终止测试,会对整个项目造成无法估量的损失。

2. 软件测试的发展

(1) 历史

软件测试起源于 20 世纪 70 年代,随着计算机的发展而产生。早期的软件测试是软件开发过程中的一个阶段,常常由开发人员自己来完成这部分工作。直到 1957 年软件测试与调试区别开来,软件测试成为发现软件缺陷的活动。软件测试的发展如表 1-2 所示。

表 1-2 软件测试的发展

时间	内容
1972 年	Bill Hetzel 博士在北卡罗来纳大学举行了首届软件测试正式会议
1979 年	《软件测试的艺术》(“The Art of Software Testing”)是软件测试领域第一本重要的专著,书中提出软件测试的定义“测试是为了发现错误而执行的一个程序或者系统的过程”
20 世纪 80 年代早期	软件行业开始关注软件产品质量并在公司设立软件质量保证部门 QA, 软件质量保证部门的职能转变为流程监控
20 世纪 90 年代	在各地成立了软件测试机构并提供相应的测试服务
2001 年后	国内兴起了一大批软件测试、软件外包服务公司

(2) 趋势

纵观国内外软件测试的发展现状,可以看到软件测试有以下的发展趋势。

①测试工作将贯穿整个开发体系。测试人员应尽早融入整个软件开发工程,在软件需求阶段就应开发相应的测试方法,使得每一个需求都是可以测试的。

②测试职业分量加重。虽然测试人员和项目开发工程师是矛盾体,但也是相互协调的整体。以前大部分开发人员认为如果开发能力不够,就去做测试,而现在却是厉害的开发者,才能胜任测试工作。

③成立独立的测试部门。越来越多的企业将软件测试和开发等部门一样作为重要的独立部门。

④外包服务增长迅速。软件测试外包将成为全球化的一种趋势,可以利用职业测试专家队伍与机构为自己的产品进行测试,而且可以节省测试费用。

3. 软件测试的原则

(1) 测试应基于用户需求

所有的测试标准应建立在满足客户需求的基础上,从用户角度来看,最严重的错误是导致程序无法满足需求的错误。应依照用户的需求配置环境并且依照用户的使用习惯进行测试并评价结果。假如系统不能满足客户的需求和期望,那么,这个系统的研发是失败的,同时在这个系统中发现和修改缺陷也是没有任何意义的。开发过程中用户早期介入和接触原型系统就是为了避免发生这类问题。有时,站在开发者角度看起来完美的产品,可能并不是用户真正想要的产品。

(2) 做好软件测试计划是做好软件测试工作的关键

软件测试是有组织、有计划、有步骤的活动,因此必须提前做好测试计划,并且要严格执行测试计划,避免测试的随意性。测试计划应包括所测软件的功能、输入和输出、测试内容、各项测试的进度安排、资源要求、测试资料、测试工具、测试用例的选择、测试的控制方法和过程、系统的配置方式、跟踪规则、调试规则、回归测试以及评价标准等。另外,回归测试的关联性一定要引起充分的注意,因修改一个错误而引起更多错误的现象并不少见。

与软件测试相关的活动依赖于测试对象的内容。对于每个软件系统,其测试策略、测试技术、测试工具、测试阶段以及测试出口准则等的选择,都是不一样的。同时,测试活动必须与应用程序的运行环境和使用中可能存在的风险相关联。因此,没有两个系统可以以完全相同的方式进行测试。比如,对关注安全的电子商务系统进行测试,与对一般的商业软件进行测试的重点是不一样的,它更多关注的是安全测试和性能测试。

(3) 应尽早开始软件测试并不断进行软件测试

由于软件的复杂性和抽象性,在软件生命周期各阶段都可能产生错误,所以不应把软件测试仅仅看作是软件开发的一个独立阶段,而应当把它贯穿于软件开发的各个阶段。在需求分析和设计阶段就应开始进行测试工作,编写相应的测试计划及测试设计文档,同时坚持在开发的各阶段进行技术评审和验证,这样才能尽早发现和预防错误,杜绝某些缺陷和错误,提高软件质量。尽早开展测试准备工作使测试人员能够在早期了解到测试的难度,预测测试的风险,有利于制定出完善的计划和方案,提高软件测试及开发的效率,规避测试中存在的风险。尽早开展测试工作,有利于测试人员尽早发现软件中的缺陷,大大降低错误修复的成本。测试工作进行得越早,越有利于提高软件的质量,这是预防性测试的基本原则。

(4) 测试前必须明确定义好产品的质量标准

只有建立了质量标准,才能根据测试的结果,对产品的质量进行分析和评估。同样,测试用例应该确定期望输出结果。如果无法确定期望输出结果,则无法进行检验。必须用预先确定对应的输入数据和输出结果来对照检查当前的输出结果是否正确,做到有的放矢。系统的质量特征不仅仅是功能性要求,还包括了很多其他方面的要求,比如稳定性、可用性和兼容性等。

(5) 避免测试自己的软件

由于心理因素的影响或者程序员本身对用户需求或者规范理解得不正确导致程序中存在错误,应避免程序员或者编写软件的组织测试自己的软件。一般要求由专门的测试人员进行测试,并且要求用户参与,特别是验收测试阶段,用户是主要的参与者。

(6) 应充分注意测试中的集群现象

一般来说,一段程序中已发现的错误数越多,其中存在的错误概率也就越大。错误集中发生的现象,可能和程序员的编程水平和习惯有很大的关系。因此,对发现错误较多的程序段,应进行更深入的测试。

(7) 必须检查每个实际输出结果

应当彻底检查每个测试的执行结果,避免因为疏忽或者对执行结果与预期结果的一致性主观臆断造成错误遗漏。

(8) 穷举测试是不可能实现的

由于时间和资源有限,穷举测试是不可能实现的,软件测试也不能无限地进行下去,应适时终止。此外,应避免冗余测试。

(9) 测试设计决定了测试的有效性和效率

测试设计决定测试的有效性和效率,测试工具只能提高测试效率而非万能。根据测试的目的,采用相应的方法设计测试用例,从而提高测试的效率,更多地发现错误,提高程序的可靠性。除了检查程序是否做了应该做的事,还要检查程序是否做了不该做的事;另外,测试用例的编写不仅应当根据有效和预料到的输入情况,而且应当根据无效和未预料的输入情况。

(10) 注意保留测试设计和说明文档,并注意测试设计的可重用性

妥善保存测试计划、测试用例、出错统计和最终分析报告,为维护等提供方便。

(11) 杀虫剂悖论

杀虫剂用得多了,就会使害虫产生免疫力,杀虫剂就发挥不了效力。在测试中,同样的测试用例被一遍一遍反复使用时,发现缺陷的能力就会越来越差。这种现象的主要原因在于测试人员没有及时更新测试用例,同时对测试用例及测试对象过于熟悉,形成思维定式。

为避免这种现象,测试用例需要经常评审和修改,不断增加新的不同的测试用例来测试软件或系统的不同部分,保证测试用例永远是最新的,即包含最后一次程序代码或说明文档的更新信息。这样软件中未被测试过的部分或者先前没有被使用过的输入组合就会重新执行,从而发现更多的缺陷。同时,作为专业的测试人员,要具有探索性思维和逆向思维,而不仅仅是做输出与期望结果的比较。

(12) 计划测试工作时不应该默认假定不会发现错误

具体内容略。

4. 软件测试的目标

①以最少的人力、物力、时间找出软件中潜在的各种错误和缺陷。

②修正各种错误和缺陷提高软件质量,避免软件发布后潜在的软件错误和缺陷造成隐患所带来的商业风险。

③将测试过程中得到的测试结果和测试信息,作为后续项目开发和测试过程改进的参考;避免出现重复的问题。

④采用高效的测试管理手段,提高软件测试的效率和软件产品质量。