



国家示范性软件学院
软件工程实训系列教材

软件工程 实训项目案例Ⅲ

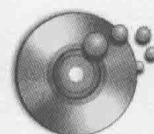
——C++程序设计篇

RUANJIAN GONGCHENG SHIXUN XIANGMU ANLI Ⅲ
C++ CHENGXU SHEJI PIAN

主 编 熊庆宇 文俊浩 雷跃明 谭会辛 曾 骏 杨正益



重庆大学出版社
<http://www.cqup.com.cn>



国家示范性软件学院
软件工程实训系列教材

软件工程 实训项目案例 Ⅲ

——C++程序设计篇

主 编 熊庆宇 文俊浩 雷跃明 谭会辛 曾 骏 杨正益

重庆大学出版社

内 容 提 要

在重庆大学软件学院开展软酷实训过程中,积累了大量的优秀 C++ 项目案例,对 C++ 方面的开发人才的培养具有重要价值。本书从中精选了一些有代表性的案例,通过项目案例产出物,完整展示项目的实践过程,以帮助读者掌握 C++ 开发技术,更深入地理解软件工程的理论知识,并更好地开展项目实训。

本书共 6 章,第 1 章介绍了实训过程,第 2 章介绍了 MFC 基本的开发技术,第 3—6 章分别介绍了五子江湖、酷 Down 下载系统、邮件分发系统、计费管理系统 4 个创新项目案例,供实训过程中参考和学习。

图书在版编目(CIP)数据

软件工程实训项目案例.3, C++ 程序设计篇/熊庆宇等主编. —重庆:重庆大学出版社,2016.3

ISBN 978-7-5624-9682-3

I. ①软… II. ①熊… III. ①软件工程—案例②C 语言—程序设计 IV. ①TP311.5②TP312

中国版本图书馆 CIP 数据核字(2016)第 032753 号

软件工程实训项目案例Ⅲ——C++ 程序设计篇

主 编 熊庆宇 文俊浩 雷跃明

谭会辛 曾 骏 杨正益

策划编辑:彭 宁 何 梅

责任编辑:陈 力 版式设计:彭 宁 何 梅

责任校对:邹 忌 责任印制:赵 晟

*

重庆大学出版社出版发行

出版人:易树平

社址:重庆市沙坪坝区大学城西路 21 号

邮编:401331

电话:(023) 88617190 88617185(中小学)

传真:(023) 88617186 88617166

网址:<http://www.cqup.com.cn>

邮箱:fxk@cqup.com.cn (营销中心)

全国新华书店经销

万州日报印刷厂印刷

*

开本:787mm×1092mm 1/16 印张:16.5 字数:466千

2016 年 4 月第 1 版 2016 年 4 月第 1 次印刷

印数:1—1 000

ISBN 978-7-5624-9682-3 定价:39.00 元

本书如有印刷、装订等质量问题,本社负责调换

版权所有,请勿擅自翻印和用本书

制作各类出版物及配套用书,违者必究

前言

软件产业是国家战略性新兴产业之一,是国民经济和社会信息化的重要基础。近年来,国家大力支持和发展软件产业,软件产业在国民经济中越来越起到举足轻重的作用。软件产业的发展需要大量兼具软件技术和软件工程实践经验的软件人才。因此,为了实现面向产业、面向领域培养实用的软件专业人才的目标,软件专业人才的培养需要突破传统的软件技术人才培养的方式,学生除了要学习软件工程专业的的基础理论和软件开发技术外,更加强调软件工程实践能力的培养,以适应我国软件产业对人才培养的需求,实现软件人才培养的跨越式发展。传统的软件专业教学按照软件工程知识体系设置课程,重点培养学生掌握扎实的软件基础理论和专业技术,教学模式主要是以知识点课堂教学为主,实验教学为辅。

目前大多数高等院校的课堂教学中都采用传统的讲授型教学方法,以知识点为主线讲解概念、原理和技术方法,期间会通过实例的讲解来加深对知识点的理解,最后会围绕知识点布置作业、实验或项目。这种以教师为中心的灌输式教学模式能较好地保证知识的系统性,但是实践性不强、教学枯燥、互动性较差、学生的积极性不高,不适宜学生的软件工程实践能力的培养。实验教学作为辅助教学方式,尽管能够在一定程度上加深学生对知识点的理解,但实验内容多是对课堂内容进行验证或实现,学生机械地运行程序,对知识的理解浮于表面,这种实验方式也不能完全达到培养学生软件工程实践能力的目标。因此,在软件专业人才的培养过程中,作为对知识点课堂教学和实验教学模式补充,有必要引入全新的软件工程实践教学模式——软件案例驱动教学模式。

案例教学模式源自哈佛商学院的“案例式教学”。案例是由一个或几个问题组成的内容完整、情节具体详细、具有一定代表性的典型实例,代表着某一类事物或现象的本质属性。所谓案例教学就是在教师的指导下,根据教学目的和要求,组织学生通过对案例的调查、阅读、思考、分析、讨论和交流等活动,交给学生分析问题和解决问题的方式和方法,进而提高他们分析问题和解决问题的能力,加深他们对基本概念和原理的理解。在软件教学中应用软件案例驱动的教学模式,是以教师为主导,以学生为主体,通过对一个或几个软件案例的剖析、讨论和实践,深入理解和掌握案例本身所反映的软件工程相关的基本原理、技术和方法,进而提高分析问题和解决问题的能力,实现

软件开发全过程的软件工程实践方法的建立和实践能力的提高。

重庆大学是教育部直属的全国重点大学,是国家 211 工程和 985 工程重点建设的大学。重庆大学软件学院成立于 2001 年,是国家发改委和教育部批准成立的 35 所国家示范性软件学院之一。几年来,学院积极探索新型办学理念和办学模式,秉承“质量是命脉、创新是动力、求实是关键、团队是保障”的办学宗旨,以培养多层次、实用型、复合型、国际性的软件工程人才为目标,注重办学特色,严格培养质量,在人才培养、队伍建设、学科建设、产学研合作、科学研究等方面取得了长足发展。学院在 2004 年获得重庆市教学成果一等奖等奖项,2005 年获得国家教学成果二等奖,2006 年通过教育部国家示范性软件学院验收。

重庆大学软件学院在软件案例驱动教学模式培养软件产业所急需的实践型软件专业人才方面进行了大量有益探索。重庆大学软件学院与深圳市软酷网络科技有限公司合作,在长期的软件工程实践教学过程中积累了丰富的、面向不同领域的教学软件案例,并不断研究和提炼,形成项目实训案例,可供软件工程实践教学使用。深圳市软酷网络科技有限公司多年来致力于软件案例教学,开发实用的案例库教学管理平台,与国内多所软件学院合作开设软件案例教学方面的课程,并面向社会培训不同级别的软件开发人才,为培养实践型的软件工程人才进行了有益的尝试。重庆大学软件学院与深圳市软酷网络科技有限公司在项目实训与案例驱动教学方面进行了多年的合作,取得了较好的成效,也获得了学生的高度认可。特别是在 C++ 项目实践过程中,积累了较多的优秀项目案例。为配合软件案例驱动的教学,合作编写了项目实训案例系列教程。

软件工程实践案例系列教程,为高校的软件工程教学提供了软件案例及教学指导。其目标是促进教学与工程实践相结合,不断沉淀教学成果,完善软件工程教学方法和课程体系。

本系列教程中的案例是由 C++ 应用项目案例中精选出来的,具有典型性和代表性,符合 CMMI 过程标准和案例编写规范,易于使用和方便学习。可用于高等院校软件工程专业的案例教学或实践教学,支持高校应用型、工程型的人才培养。同时,也可作为软件行业或不同应用领域中的软件项目实训教材,支持软件产业人才的继续教育和培养。

案例的选择是案例教材编写的关键。案例的选取应以激发学生的学习兴趣、提高学生分析解决问题的能力 and 软件工程实践能力为出发点,根据知识点教学内容的需要,选取典型行业的典型应用的软件案例。案例选择的原则如下:

(1) 生动实际

案例教材的案例来源于实际需求,贴近生活实际,生动有趣,

可以激发学生的学习兴趣。

(2) 领域背景

案例的选择尽量贴近软件行业的不同领域,具有典型性和代表性,这样更能贴近软件工程实践,使案例教学更好地满足实践教学的目标。

(3) 难易适中

案例的选择要考虑学生的知识背景,难易适中的案例才会调动学生的学习兴趣和积极性,有利于学生进行深入学习,调动学习的主动性和积极性。

(4) 覆盖面广

案例要能覆盖多个知识点,以便提高学生综合运用知识的能力,达到整合知识的目的。

案例教材的内容不是单纯的案例介绍,而是以案例教学为核心的整个教学过程的设计。每个案例的内容都按照软件工程过程进行组织,包括:项目立项、项目计划、软件需求、软件设计、软件实现、软件测试等环节。

因篇幅所限,本书仅收录了4个有代表性的优秀C++项目案例,其并不能完全代表各个类别,但基本上满足上述原则,可在后续的教程中增加和更新案例内容。由于作者水平有限,书中难免存在疏漏之处,欢迎广大读者提出宝贵的意见。

编 者

2016年1月

目 录

第1章 实训简介	1
1.1 实训简介	1
1.2 实训过程	2
第2章 C++ 基础	12
2.1 安装 C++ 开发环境	12
2.2 MFC 的项目结构	18
第3章 软件工程实训项目案例一:五子江湖	47
3.1 项目立项报告	48
3.2 软件项目计划	49
3.3 软件需求规格说明书	51
3.4 软件设计说明书	62
3.5 软件测试计划	73
3.6 软件测试报告(示例)	75
3.7 项目验收报告(示例)	78
3.8 项目关闭总结报告(示例)	81
第4章 软件工程实训项目案例二:酷 Down 下载	83
4.1 项目立项报告	83
4.2 软件项目计划	84
4.3 软件需求规格说明书	89
4.4 软件设计说明书	105
4.5 软件测试计划	127
4.6 软件测试报告(示例)	130
4.7 项目验收报告(示例)	135
4.8 项目关闭总结报告(示例)	138
第5章 软件工程实训项目案例三:邮件分发系统	140
5.1 系统项目立项报告	141
5.2 软件项目计划	142
5.3 软件需求规格说明书	144
5.4 软件设计说明书	155

5.5	系统测试计划	184
5.6	系统测试报告(示例)	187
5.7	系统验收报告(示例)	192
5.8	项目关闭报告(示例)	196

第6章 软件工程实训项目案例四:计费管理系统 198

6.1	系统项目立项报告	199
6.2	软件项目计划	200
6.3	软件需求规格说明书	202
6.4	软件设计说明书	213
6.5	系统测试计划	240
6.6	系统测试报告(示例)	242
6.7	系统验收报告(示例)	247
6.8	项目关闭报告(示例)	251

附录 253

词汇表 255

参考文献 256

第 1 章

实训简介

1.1 实训简介

实训教学是训练学生运用理论知识解决实际问题、提升已有技能和实践经验的重要过程,是学校教学工作的重要组成部分,相对于理论教学更具有直观性、综合性和实践性,在强化学生的素质教育和培养创新能力方面有着不可替代的作用。2010年6月,作为中国教育部落实《国家中长期教育改革和规划纲要(2010—2020年)》和《国家中长期人才发展规划纲要(2010—2020年)》的重大改革项目的“卓越工程师教育培养计划”正式制订,此计划的目标就是培养造就一大批创新能力强、适应经济社会发展需要的高质量各类型工程技术人才。在此背景下,工程项目实训更显示出其重要性。而对“以市场为导向,以培养具有国际竞争能力的多层次实用型软件人才为目标”的软件工程专业人才培养,实训环节显得尤为重要。

重庆大学软件学院软酷工程实践,由重庆大学软件学院和软酷网络科技有限公司联合实施,在校园里共建工程实践基地,采用国际化软件开发方式和企业化管理模式,由软酷网络公司负责管理项目的研发过程,让学生体验企业软件项目开发的全过程,加强理论知识的综合运用,锻炼学生的实践能力,提升软件工程素养。

软酷工程实践,将软件研发的专业课程结合到项目实践的过程中,以实际软件开发项目和企业规范的软件开发过程为主线,以项目开发和交付为目标,以技术方向和研究兴趣为导向,让学员参与到实际的软件项目开发中来,加深对需求分析、架构设计、编码测试、项目管理等方面的知识运用,巩固软件工程课程群的理论知识并应用于实践,加深学生对理论知识的理解和实践动手能力,提高技术水平和创新能力,并积累一定的实际项目开发和项目管理经验,最终帮助学生达到重庆大学软件学院的人才培养目标。

软酷工程实践,采用以学生实际开发体验为主、企业导师重点讲授并全程指导为辅的CDIO(做中学、学中做)形式,使不同知识结构、不同软件开发动手实践能力、不同职业发展目标的学生,都能够按照自己的基础和职业规划目标,在自己合适的软件工程角色中,获得学习、体验、实践、提高的机会,实现与企业人才需求的无缝对接。

1.2 实训过程

软酷工程实践按照 CMMI 3 建立项目软件过程,以让学员能在规范的项目过程下开展实训,并熟悉项目研发生命周期,如图 1.1 所示。

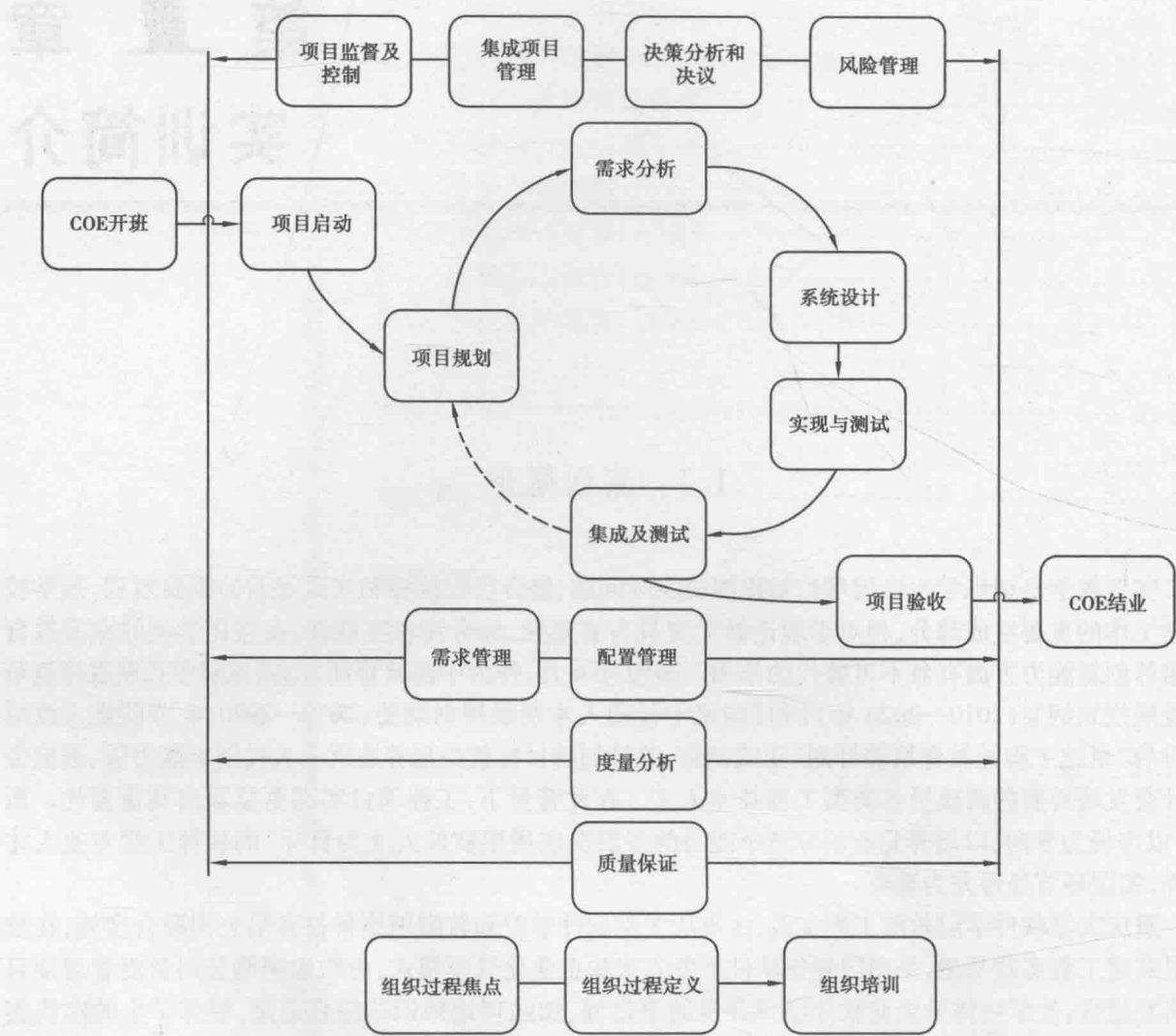


图 1.1 项目软件过程

在项目开发小组中,一般不固定区分需求分析、系统设计、程序编码、测试、配置管理等角色,采用轮流和交叉的方式,让学员都有机会担任这些角色,获得多种角色的开发经验。

- ①项目经理:负责项目的组织实施,制订项目计划,并进行跟踪管理。
- ②开发人员:对项目经理及项目负责。
- ③需求分析员:负责系统的需求获取和分析,并协助设计人员进行系统设计。
- ④系统设计、架构设计:负责系统设计工作,并指导程序员进行系统的开发工作。
- ⑤程序员:一般模块的详细设计、编码测试,并交叉进行模块的白盒测试。
- ⑥数据库管理员:负责数据库的建立和数据库的维护工作。
- ⑦测试人员:进行项目各阶段的测试工作,包括模块测试(白盒测试)、系统的需求测试、集成测

试、系统测试等工作,对用户需求负责。

⑧配置管理员:负责项目的配置管理。

⑨质量保证人员:由独立的小组进行。

1.2.1 需求分析及原型设计

项目需求分析是一个项目的开端,也是项目建设的基石。在以往建设失败的项目中,80%是由于需求分析的不明确造成的。因此,对用户需求的把握程度,是项目成功的关键因素之一。

需求是指必须实现什么的规格说明。它描述了系统的行为、特性和属性,是在开发过程中对系统的约束。需求包括业务需求(反映了组织机构或客户对系统、产品高层次的目标要求)、用户需求(描述了用户使用产品必须要完成的任务)、功能需求(定义开发人员必须实现的软件功能,使用户利用系统能够完成他们的任务,从而满足了业务需求)、非功能性需求(描述系统展现给用户的行为和执行的操作等,它包括产品必须遵从的标准、规范和约束,操作界面的具体细节和构造上的限制)。

需求分析阶段可分为获取需求→分析需求→编写需求文档3个步骤。

(1) 获取需求

①先了解项目所有用户类型以及潜在的类型。然后根据用户的要求来确定系统的整体目标和系统的工作范围。

②将需求细分为功能需求、非功能需求(如响应时间、平均无故障工作时间、自动恢复时间等)、环境限制、设计约束等类型。

③确认需求获取的结果是否真实地反映了用户的意图。

(2) 分析需求

①以图形表示的方式描述系统的整体结构,包括系统的边界与接口。

②通过原型、页面流或其他方式向用户提供可视化的界面,用户可以对需求作出自己的评价。

③系统可行性分析,需求实现的技术可行性、环境分析、费用分析、时间分析等。

④以模型描述系统的功能项、数据实体、外部实体、实体之间的关系、实体之间的状态转换等方面的内容。

(3) 编写需求文档

①使用自然语言或形式化语言来描述。

②添加图形的表述方式和模型表征的方式。

③需包括用户的所有需求(功能性需求和非功能性需求)。

在通常情形下,分析需求是与获取需求并行的,主要通过建立模型的方式来描述需求,为客户、用户、开发方等不同参与方提供一个交流的渠道。这些模型是对需求的抽象,以可视化的方式提供一个易于沟通的桥梁。

用于需求建模的方法有很多种,最常用的包括用例图(Use Case)、实体关系图(ERD)和数据流图(DFD)3种方式。在面向对象分析的方法中通常使用 Use Case 来获取软件的需求。Use Case 通过描述“系统”和“活动者”之间的交互来描述系统的行为。通过分解系统目标,Use Case 描述活动者为了实现这些目标而执行的所有步骤。Use Case 方法最主要的优点在于它是用户导向的,用户可以根据自己所对应的 Use Case 来不断细化自己的需求。此外,使用 Use Case 还可以方便地得到系统功能的测试用例。ERD 方法用于描述系统实体间的对应关系,需求分析阶段使用 ERD 描述系统中实体的逻辑关系,在设计阶段则使用 ERD 描述物理表之间的关系。需求分析阶段使用 ERD 来描述现实世界中的对象。ERD 只关注系统中数据间的关系,而缺乏对系统功能的描述。DFD 作为结构化系统分析与

设计的主要方法,尤其适用于 MIS 系统的表述。DFD 使用 4 种基本元素来描述系统的行为,过程、实体、数据流和数据存储。DFD 方法直观易懂,使用者可以方便地得到系统的逻辑模型和物理模型,但是从 DFD 图中无法判断活动的时序关系。

在需求分析阶段通常使用原型分析方法来帮助开发方进一步获取用户需求或让用户确认需求。开发方往往先向用户提供一个可视界面作为原型,并在界面上布置必要的元素以演示用户所需要的功能。可以使用 DreamWare 等网页制作工具、HTML 语言、Axure-RP 原型开发工具等快速形成用户界面,生成用户可视的页面流。原型的目的是获取需求。有时也使用原型的方式来验证关键技术或技术难点。对于技术原型,界面则往往被忽略掉。

对于 C++ 项目而言,原型设计的重要性更为突出。甚至可以说,界面(美观+易用性)是 C++ 项目的灵魂。

原型设计,绝不仅仅只是画几个界面,设计思路应遵循用户导向+简易操作原则:

- ①要形成对人们希望的产品使用方式,以及人们为什么想用这种产品等问题的见解。
- ②尊重用户知识水平、文化背景和生活习惯。
- ③通过界面设计,让用户明白功能操作,并将作品本身的信息更加顺畅地传递给使用者。
- ④通过界面给用户一种情感传递,使用户在接触作品时产生情感共鸣。
- ⑤展望未来,要看到产品可能的样子,它们并不必然就像当前这样。

在需求分析和原型设计阶段,离不开各种各样功能强大的工具。常用需求分析和原型设计工具包括:Axure RP Pro、StarUML、Visio、FreeMind 思维导图软件。

①Axure RP Pro。Axure RP 能帮助网站需求设计者快捷而简便地创建基于目录组织的原型文档、功能说明、交互界面以及带注释的 wireframe 网页,并可自动生成用于演示的网页文件和 Word 文档,以提供演示与开发。

Axure RP 的特点是:快速创建带注释的 wireframe 文件,并可根据所设置的时间周期,软件自动保存文档,确保文件安全。在不写任何一条 html 与 JavaScript 语句的情况下,通过创建的文档以及相关条件和注释,一键生成 html prototype 演示。根据设计稿一键生成一致而专业的 Word 版本的原型设计文档。

②StarUML。StarUML 可绘制 9 款 UML 图:用例图、类图、序列图、状态图、活动图、通信图、模块图、部署图以及复合结构图等。

完全免费:StarUML 是一套开放源码的软件,不仅提供免费自由下载,连代码都免费开放。

多种格式影像文件:可导出 JPG、JPEG、BMP、EMF 和 WMF 等格式的影像文件。

语法检验:StarUML 遵守 UML 的语法规则,不支持违反语法的动作。

正反向工程:StarUML 可以依据类图的内容生成 Java、C++、C# 代码,也能够读取 Java、C++、C# 代码反向生成类图。

③Visio。Microsoft Visio 可以建立流程图、组织图、时间表、营销图和其他更多图表,将特定的图表加入文件,让商业沟通变得更加清晰,令演示更加有趣。

④FreeMind 思维导图软件。FreeMind 是一款实用的开源思维导图/心智(MindMap)软件,其可用来作为管理项目(包括子任务的管理,子任务的状态,时间记录,资源链接管理),笔记或知识库,文章写作或者头脑风暴,结构化地存储小型数据库,绘制思维导图,整理软件流程思路。

在需求分析阶段,有下述几点注意事项:

- 需求分析阶段关注的目标是“做什么”,而不是“怎么做”。
- 识别隐含需求(有可能是实现显式需求的前提条件)。

- 需求符合系统的整体目标。
- 保证需求项之间的一致性,解决需求项之间可能存在的冲突。

1.2.2 需求及原型评审

需求文档完成后,需要经过正式评审,以便作为下一阶段工作的基础。评审的目的是在缺陷遗漏到开发的下一阶段之前将其探查和标识出来,这样有助于在问题扩大化、变得复杂难以处理之前将其纠正。需求评审通过对需求规格说明书进行技术评审来减少缺陷和提高质量。需求评审可以通过以下两种方式进行:用户评审和同行评审。用户和开发方对于软件项目内容的描述,是以需求规格说明书作为基础的;用户验收的标准则是依据需求规格说明书中的内容来制订,所以评审需求文档时用户的意见是第一位的。而同行评审的目的,是在软件项目初期发现那些潜在的缺陷或错误,避免这些错误和缺陷遗漏到项目的后续阶段。

评审(不仅限于需求评审,也包括设计和其他类型的评审)的基本目的是:

- ①在开发的较早阶段将缺陷探查出来。
- ②验证工作产品符合预先设定的准则。
- ③提供产品和评审过程的相关数据,包括对评审中能发现的缺陷数的预测能力。

评审(不仅限于需求评审,也包括设计和其他类型的评审)须遵循以下的基本原则:

①评审是一个结构化的正式过程,由系统化的一系列检查单来帮助工作,并且参与者分别有不同的角色。

②评审人员事先要经过准备工作,并在小组评审进行之前要明确他们自己工作的重点,以及个人已经发现的问题。

③评审的工作重点是发现问题,而不是解决问题。

技术人员进行小组评审,项目负责人通常不参与软件工作产品的小组评审,但对评审结果要了解。但是对于项目管理文档,有经验的项目负责人要参与小组评审。

小组评审数据要记录下来,以供监控小组评审过程是否有效。

需求评审的重点包括:

以下基本问题是否得到解决?

- 功能:本软件有什么用途?
- 外部接口:此软件如何与人员、系统硬件、其他硬件及其他软件进行交互?
- 性能:不同软件功能都有什么样的速度、可用性、响应时间、恢复时间等?
- 属性:在正确性、可维护性、安全性等方面都有哪些事项要考虑?
- 是否指定了在需求规格说明书范围之外的任何需求?
- 不应说明任何设计或实施细节。
- 不应该对软件附加更多约束。
- 需求规格说明书是否合理地限制了有效设计的范围而不指定任何特定的设计?
- 需求规格说明书是否显示以下特征?

(1) 正确性

需求规格说明书规定的所有需求是否都是软件应该满足的?

(2) 明确性

- ①每个需求是否都有且只有一种解释?
- ②是否已使用客户的语言?

③是否已使用图来补充自然语言说明?

(3) 完全性

①需求规格说明书是否包括所有的重要需求(无论其与功能、性能设计约束、属性有关还是与外部接口有关)?

②是否已确定并指出所有可能情况输入值的预期范围?

③响应是否已同时包括在有效输入值和无效输入值中?

④所有的图、表和图表是否都包括所有评测术语和评测单元的完整标注、引用和定义?

⑤是否已解决或处理所有的未确定因素?

(4) 一致性

①此需求规格说明书是否与前景文档、用例模型和补充规约相一致?

②它是否与更高层的规约相一致?

③它是否保持内部一致,其中说明的个别需求的任何部分都不发生冲突?

(5) 排列需求的能力

①每个需求是否都已通过标识符来标注,以表明该特定需求的重要性或稳定性?

②是否已标识出正确确定优先级的其他重要属性?

(6) 可核实性

①在需求规格说明书中说明的所有需求是否可被核实?

②是否存在一定数量可节省成本的流程可供人员或机器用来检查软件产品是否满足需求?

(7) 可修改性

①需求规格说明书的结构和样式是否允许在保留结构和样式不变的情况下方便地对需求进行全面而统一的更改?

②是否确定和最大限度地减少了冗余,并对其进行交叉引用?

(8) 可追踪性

①每个需求是否都有明确的标识符?

②每个需求的来源是否确定?

③是否通过显式引用早期的工件来维护向后可追踪性?

④需求规格说明书产生的工件是否具有相当大的向前可追踪性?

1.2.3 概要设计及数据库详细设计

系统设计是在软件需求与编码之间架起一座桥梁,重点解决系统结构和需求向实现平坦过渡的问题。系统设计的主要任务是把需求分析得到的 DFD 转换为软件结构和数据结构,其包括:计算和配置设计、系统模块结构设计、数据库和文件设计、代码设计以及系统可靠性与内部控制设计等。设计软件结构的具体任务是:将一个复杂系统按功能进行模块划分、建立模块的层次结构及调用关系、确定模块间的接口及人机界面等。数据结构设计包括数据特征的描述、确定数据的结构特性、数据库的设计。

一个完整的系统设计应包含下述内容:

①任务:目标、环境、需求、局限。

②总体设计:处理流程、总体结构与模块、功能与模块的关系。

③接口设计:总体说明外部用户,软、硬件接口;内部模块间接口。

④数据结构:逻辑结构、物理结构,与程序结构的关系。

⑤模块设计:每个模块“做什么”、简要说明“怎么做”(输入、输出、处理逻辑、与其他模块的接口,与其他系统或硬件的接口),处在什么逻辑位置、物理位置。

⑥运行设计:运行模块组合、控制、时间。

⑦出错设计:出错信息、出错处理。

⑧其他设计:安全性设计、可维护性设计、可扩展性设计。

详细阅读需求规格说明书,理解系统建设目标、业务现状、现有系统、客户需求的各功能说明是进行系统设计的前提。常规上,系统设计方法可分为:结构化软件设计方法和面向对象软件设计方法。在此,我们重点介绍面向对象软件设计方法(OO设计方法)。

第一步是抽取建立领域的概念模型,在UML中表现为建立对象类图、活动图和交互图。对象类就是从对象中经过“察同”找出某组对象之间的共同特征而形成类:

①对象与类的属性:数据结构。

②对象与类的服务操作:操作的实现算法。

③对象与类的各外部联系的实现结构。

④设计策略:充分利用现有的类。

⑤方法:继承、复用、演化。

活动图用于定义工作流,主要说明工作流的5W(Do What、Who Do、When Do、Where Do、Why Do)等问题,交互图把人员和业务联系在一起是为了理解交互过程,发现业务工作流程中相互交互的各种角色。

第二步是构建完善系统结构:对系统进行分解,将大系统分解为若干子系统,子系统分解为若干软件组件,并说明子系统之间的静态和动态接口,每个子系统可以由用例模型、分析模型、设计模型、测试模型表示。软件系统结构的两种方式为层次、块状。

①层次结构:系统、子系统、模块、组件(同一层之间具有独立性)。

②块状结构:相互之间弱耦合。

系统的组成部分:问题论域(业务相关类和对象)、人机界面(窗口、菜单、按钮、命令等)、数据管理(数据管理方法、逻辑物理结构、操作对象类)、任务管理(任务协调和管理进程)。

第三步是利用“4+1”视图描述系统架构:用例视图及剧本;说明体系结构的设计视图;以模块形式组成包和层包含概要实现模型的实现视图;说明进程与线程及其架构、分配和相互交互关系的过程视图;说明系统在操作平台上的物理节点和其上的任务分配的配置视图。在RUP中还有可选的数据视图。

第四步是性能优化(速度、资源、内存)、模型清晰化、简单化。

数据库设计是系统设计中的重要环节,对于信息系统而言,数据库设计的好坏直接决定了系统的好坏。数据库设计又称数据库建模,指对于一个给定的应用环境,构造最优的数据库模式,建立数据库及其应用系统,使之能够有效地存储数据,满足各种用户的应用需求(信息要求和处理要求);它主要包括两部分内容:确定最基本的数据结构以及对约束建模。

(1) 建立概念模型

根据应用的需求,画出能反映每个应用需求的E-R图,其中包括确定实体、属性和联系的类型。然后优化初始的E-R图,消除冗余和可能存在的矛盾。概念模型是对用户需求的客观反映,并不涉及具体的计算机软、硬件环境。因此,在这一阶段中设计者必须将注意力集中在怎样表达出用户对信息的需求,而不考虑具体实现问题。

(2) 建立数据模型

将 E-R 图转换成关系数据模型,实际上就是要将实体、实体的属性和实体之间的联系转换为关系模式。

(3) 实施与维护数据库

完成数据模型的建立后,对字段进行命名,确定字段的类型和宽度,并利用数据库管理系统或数据库语言创建数据库结构、输入数据和运行等。

数据库设计应遵循如下原则:

- 标准化和规范化(如遵循 3NF)。
- 数据驱动(采用数据驱动而非硬编码的方式)。
- 考虑各种变化(考虑哪些数据字段将来可能会发生变更)。

设计阶段常用工具如下所述。

① Rational Rose。Rational Rose 太过庞大,其优势可能在于强大的功能,包括代码生成,它能够直观地表现出需求分析和功能设计阶段的思路。

② EA(Enterprise Architect)。小巧、界面美观、操作方便是 EA 的优点,在功能上其包含了大部分设计上能够用的功能,是一款很不错的设计软件。缺点是代码生成功能,有了设计图之后,再按照设计图重构代码的确是一件挺头痛的事情。

③ PowerDesigner。不可否认 PowerDesigner 是一款数据库设计必不可少的功能齐全的设计软件。PowerDesigner 是 Sybase 公司的 CASE 工具集,使用它可以方便地对管理信息系统进行分析设计,其几乎包括了数据库模型设计的全过程。利用 PowerDesigner 可以制作数据流程图、概念数据模型、物理数据模型,可以生成多种客户端开发工具的应用程序,还可为数据仓库制作结构模型,也能对团队设备模型进行控制。

1.2.4 设计评审

设计文档完成后,需要经过正式评审,以便作为下一阶段工作的基础。评审的目的是在缺陷泄漏到开发的下一阶段之前将其探查和标识出来,这有助于在问题扩大化、变得复杂难以处理之前将其纠正。设计评审通过对系统设计说明书进行技术评审以减少缺陷和提高质量。设计评审通常采用同行评审,目的是在软件项目初期发现那些潜在的缺陷或错误,避免这些错误和缺陷遗漏到项目的后续阶段。

系统设计评审的重点包括:

- 系统设计是否正确描述了预期的系统行为和特征。
- 系统设计是否完全反映了需求。
- 系统设计是否完整。
- 系统设计是否为继续进行构造和测试提供了足够的基础。

数据库设计评审重点包括:

- 满足需求。
- 整体结构。
- 命名规范。
- 存储过程。
- 注释。
- 性能。

➤可移植性。

➤安全性。

1.2.5 编码

根据开发语言、开发模型、开发框架的不同,编码规范细则不尽相同,甚至不同软件公司也有着各自不同等级、不同层次要求的编码规范;但是在各式各样的编码规范之间,存在差异的通常是执行细节,在总体标准上仍然是统一的。

执行编码规范的目的是提升代码的可读性和可维护性,减少出错概率。

标准意义上的编码规范应包含下述几个方面:

➤排版。

➤注释。

➤标识符命名。

➤变量与结构。

➤函数与过程。

➤程序效率。

➤质量保证。

➤代码编辑、编译、审查。

1.2.6 测试

软件测试是在将软件交付给客户之前所必须完成的重要步骤,是目前用来验证软件是否能够完成所期望的功能的唯一有效的方法。软件测试的目的是验证软件是否满足软件开发合同或项目开发计划、系统/子系统设计文档、SRS、软件设计说明和软件产品说明等规定的软件质量要求。软件测试是一种以受控的方式执行被测试的软件,以验证或者证明被测试的软件的行为或者功能符合设计该软件的目的或者说明规范。所谓受控的方式,应该包括正常条件和非正常条件,即故意去促使错误的发生,也就是事情在不该出现的时候出现或者在应该出现的时候没有出现。

测试工作的起点,是从需求阶段开始的。在需求阶段,就需要明确测试范围、测试内容、测试策略和测试通过准则,并根据项目周期和项目计划制订测试计划。测试计划完成后,根据测试策略和测试内容,进行测试用例的设计,以便系统实现后进行全面测试。

测试用例设计的原则有基于测试需求的原则、基于测试方法的原则、兼顾测试充分性和效率的原则、测试执行的可再现性原则;每个测试用例应包括名称和标识、测试追踪、用例说明、测试的初始化要求、测试的输入、期望的测试结果、评价测试结果的准则、操作过程、前提和约束、测试终止条件。

在开发人员将所开发的程序提交测试人员后,由测试人员组织测试,项目内部测试一般可分为3个阶段:

(1) 单元测试

单元测试集中在检查软件设计的最小单位——模块上,通过测试发现实现该模块的实际功能与定义该模块的功能说明不符合的情况,以及编码的错误。由于模块规模小、功能单一、逻辑简单,测试人员有可能通过模块说明书和源程序,清楚地了解该模块的 I/O 条件和模块的逻辑结构,采用结构测试(白盒法)的用例,尽可能达到彻底测试,然后辅之以功能测试(黑盒法)的用例,使之对任何合理和不合理的输入都能鉴别和响应。高可靠性的模块是组成可靠系统的坚实基础。