

全国高等院校计算机基础教育研究会
“计算机系统能力培养教学研究与改革课题”立项项目

软件工程 课程设计教程

主 编◎李香菊 孙 丽 谢修娟 操凤萍
副主编◎朱 林

案例教学 | 选择主流案例，进行全生命周期讲解

内容丰富 | 理论知识精炼，案例结合习题，实用性强

效果显著 | 突出实践，以提高工程思维能力为目标



北京邮电大学出版社
www.buptpress.com

全国高等院校计算机基础教育研究会

“计算机系统能力培养教学研究与改革课题”立项项目

软件工程

课程设计教程

主 编◎李香菊 孙 丽 谢修娟 操凤萍

副主编◎朱 林



北京邮电大学出版社
www.buptpress.com

内 容 简 介

本书是软件工程课程设计的指导教材,全书共6章,内容涵盖了软件工程基本概念、结构化软件开发方法介绍、面向对象软件开发方法介绍、软件测试技术介绍、结构化软件开发方法案例文档和面向对象软件开发方法案例文档,附录内容介绍 Visio 2010 和 Rational Rose 2003 两个建模工具的使用方法。本书以增强实践能力为目标,通过实例与理论知识相融合的讲解方式,帮助读者理解软件开发任务,掌握开发方法,参照提供的案例文档完成软件工程课程设计。

本书内容通俗易懂,既可以成为高校软件工程和计算机科学与技术等相关专业本科、专科的软件工程课程设计教材,又可以作为理论课程的辅助教材,也适合作为自学教材,是一本理论联系实际、实践性较强的专业书籍。

图书在版编目(CIP)数据

软件工程课程设计教程 / 李香菊等主编. -- 北京:北京邮电大学出版社, 2016.1

ISBN 978-7-5635-4659-6

I. ①软… II. ①李… III. ①软件工程—课程设计—高等学校—教材 IV. ①TP311.5

中国版本图书馆 CIP 数据核字 (2016) 第 008358 号

书 名: 软件工程课程设计教程

著作责任者: 李香菊 孙 丽 谢修娟 操凤萍 主编

责任编辑: 刘 颖

出版发行: 北京邮电大学出版社

社 址: 北京市海淀区西土城路 10 号 (邮编: 100876)

发 行 部: 电话: 010-62282185 传真: 010-62283578

E-mail: publish@bupt.edu.cn

经 销: 各地新华书店

印 刷: 北京通州皇家印刷厂

开 本: 787 mm×1 092 mm 1/16

印 张: 16.5

字 数: 412 千字

印 数: 1—2 000 册

版 次: 2016 年 1 月第 1 版 2016 年 1 月第 1 次印刷

ISBN 978-7-5635-4659-6

定 价: 35.00 元

· 如有印装质量问题, 请与北京邮电大学出版社发行部联系 ·

前 言

软件工程是一门研究软件开发和管理的工程学科,是高等院校计算机科学与技术、软件工程及相关专业重要的主干课程。软件工程课程设计是在软件工程课程后开设的一个综合性的实践教学环节,其目的在于促进学生复习和巩固计算机软件设计知识,加深对软件设计方法、软件设计技术和设计思想的理解,并能运用所学软件设计知识进行综合软件设计,增强软件开发实践能力。

总结前期多项关于软件工程课程的教学改革成果,结合社会企业对计算机专业学生实践能力的要求,该教材通过实例与理论知识相融合的讲解方式,适当介绍理论知识,突出实践能力的培养,帮助学生理解软件开发任务,掌握开发方法,参照提供的案例文档完成软件工程课程设计,主要包括如下内容。

(1) 软件工程基本概念:介绍软件工程的原理、简单分析软件生命周期、软件开发模型和软件开发方法。

(2) 结构化开发方法介绍:讲解如何采用结构化开发方法进行软件开发,选择学生所熟悉的案例,贯穿软件需求分析、概要设计和详细设计的过程,重点分析各阶段要完成的任务和建模方法,以达到深入理解并掌握结构化开发方法的目的。

(3) 面向对象软件开发方法介绍:讲解如何采用面向对象开发方法进行软件开发,选择学生熟悉的案例,贯穿软件需求分析、系统分析和系统设计的过程,重点分析各阶段要完成的任务和建模方法,以达到深入理解并掌握面向对象开发方法的目的。

(4) 软件测试技术介绍:介绍软件测试分类和测试方法,通过案例讲解软件测试方法。

(5) 结构化软件开发方法案例文档:选取案例,按照软件开发过程,完成各个开发阶段的文档,包括系统可行性分析报告、需求分析报告、概要设计报告、详细设计报告和测试报告。

(6) 面向对象软件开发方法案例文档:选取案例,按照软件开发过程,完成各个开发阶段的文档,包括系统分析报告、系统设计报告和系统实现报告。

(7) 附录内容:介绍建模工具的使用方法,主要是 Visio 2010 和 Rational Rose 2003 两个工具。

相比其他软件工程课程设计教材,本教材具有以下特色:

(1) 结构化开发方法和面向对象开发方法的讲解涵盖软件工程过程的关键阶段,突出实践性,简单介绍理论知识,重点结合实例分析软件开发过程中要完成各项任务的方法,

揭示软件工程理论在实际开发中的应用。

(2) 结合软件工程课程设计的教学目标,对于开发的各个阶段,分别提供了两种开发方法的案例文档,以便读者学习如何编写文档。

(3) 测试内容由教授软件测试的老师负责编写,给出了软件测试计划和测试报告的编写方法,并引入基于场景测试的方法。

(4) 案例中留出部分任务由学生完成,学生边学边做,加深理解。

本书由李香菊、孙丽、谢修娟、操凤萍担任主编,朱林担任副主编,李香菊负责统稿。具体编写分工为:李香菊编写第1~3章,孙丽编写第4章、第5章中的测试部分及附录C和附录D,操凤萍编写第5章和附录A,谢修娟编写第6章和附录B,朱林为该书的编写搜集了大量的资料。本书中的部分案例选自学生的优秀作业,并经老师进一步完善,于祥、叶慧敏和陈鹏等同学参与了案例资料整理工作。在本书编写的过程中东南大学计算机学院的王晓蔚副教授和沈军教授提了很多建议,在此一并表示感谢。

由于时间仓促和编者水平所限,不当和谬误之处敬请广大专家和读者指正。

编 者

2016年1月1日

目 录

第 1 章 软件工程综述	1
1.1 软件的发展与软件工程	1
1.2 软件工程的原理	1
1.3 软件生命周期	2
1.4 软件开发方法	3
第 2 章 结构化系统分析与设计方法	5
2.1 结构化开发方法概述	5
2.2 案例介绍	6
2.3 结构化可行性分析	6
2.3.1 可行性分析任务	6
2.3.2 案例讲解可行性分析过程	6
2.4 结构化系统需求分析	8
2.4.1 结构化需求分析任务	9
2.4.2 案例讲解结构化需求分析过程	10
2.5 结构化概要设计	21
2.5.1 结构化设计原则	22
2.5.2 结构化概要设计任务	23
2.5.3 案例讲解结构化概要设计过程	24
2.6 结构化详细设计	34
2.6.1 详细设计任务	34
2.6.2 案例讲解详细设计过程	34
第 3 章 面向对象系统分析与设计方法	41
3.1 面向对象开发方法概述	41
3.1.1 面向对象的基本概念	41
3.1.2 面向对象的主要特性	42
3.2 统一建模语言 UML	42
3.2.1 模型元素	43
3.2.2 UML 图	44
3.2.3 视图	44

3.3 案例简介	45
3.4 面向对象系统分析	45
3.4.1 面向对象分析任务	45
3.4.2 案例讲解面向对象系统分析过程	46
3.5 面向对象系统设计方法	52
3.5.1 面向对象设计任务	52
3.5.2 案例讲解面向对象设计过程	52
第4章 软件测试	69
4.1 软件缺陷	69
4.2 软件质量	69
4.3 软件测试流程	70
4.4 制订测试计划	71
4.5 设计测试用例	72
4.5.1 白盒技术	72
4.5.2 黑盒技术	77
4.6 测试执行阶段	82
4.6.1 单元测试	82
4.6.2 集成测试	83
4.6.3 确认测试	85
4.6.4 系统测试	85
4.6.5 验收测试	86
第5章 结构化开发案例——书务管理系统	89
5.1 案例介绍	89
5.2 系统可行性分析报告	89
5.2.1 引言	89
5.2.2 可行性研究的前提	90
5.2.3 对现有系统的分析	91
5.2.4 所建议系统技术可行性分析	91
5.2.5 成本效益分析	96
5.2.6 社会可行性分析	97
5.2.7 结论意见	98
5.3 系统需求分析报告	98
5.3.1 引言	98
5.3.2 任务概述	99
5.3.3 系统需求结构分析	100
5.3.4 功能分析	101
5.3.5 性能分析	110

5.3.6 运行环境分析	111
5.4 系统概要设计报告	111
5.4.1 引言	111
5.4.2 小型书店书务系统的体系结构	111
5.4.3 系统数据库设计	115
5.5 系统详细设计报告	119
5.5.1 引言	119
5.5.2 系统主程序流程	119
5.5.3 界面设计	126
5.6 系统集成测试计划书	137
5.6.1 引言	137
5.6.2 测试项目	138
5.6.3 被测特性	138
5.6.4 测试方法	143
5.6.5 测试通过标准	144
5.6.6 测试挂起和恢复条件	145
5.6.7 应提供的测试文件	145
5.6.8 测试任务	145
5.6.9 测试环境需求	146
5.6.10 角色和职责	147
5.6.11 测试进度	147
5.6.12 记录和解决问题	147
5.6.13 重新测试程序	147
5.7 系统验收测试计划书	147
5.7.1 简介	147
5.7.2 任务概述	148
5.7.3 验收项目和验收标准	149
5.7.4 验收测试方法	153
5.7.5 测试策略	156
5.7.6 应提供的测试文件	157
5.7.7 测试任务	157
5.7.8 测试资源	157
第6章 面向对象开发案例——云环境下高校网络教辅系统	160
6.1 案例简介	160
6.2 面向对象分析	161
6.2.1 系统的功能需求分析	161
6.2.2 划分子系统	162
6.2.3 建立用例模型	162

6.2.4	系统的业务流程分析	165
6.2.5	功能用例的详细描述	168
6.3	面向对象设计	181
6.3.1	建立类图	181
6.3.2	对象交互设计	189
6.3.3	对象状态模型设计	191
6.3.4	数据库设计	191
6.4	面向对象实现	195
6.4.1	系统开发软件和运行环境	195
6.4.2	系统的物理实现构件	196
6.4.3	系统主要页面展示	198
6.4.4	重难点模块的程序实现	201
6.4.5	系统的部署	210
附录 A	Visio 2010 简介	211
附录 B	Rational Rose 简介	224
附录 C	书店书务管理系统模块场景法测试	245
附录 D	书店书务管理系统模块基路径测试	253
参考文献		256

第 1 章 软件工程综述

1.1 软件的发展与软件工程

软件是程序设计发展到一定规模,并且逐步商品化的过程中形成的。软件除了能够完成预定功能和性能的可执行的计算机程序外,还包括使程序正常执行所需的数据,以及与程序开发、维护和使用有关的图文材料。随着计算机技术的发展,软件开发经历了程序设计阶段、软件设计阶段和软件工程阶段的演变过程。

1946—1955 年,软件开发处于程序设计阶段,此时还没有软件的概念,程序设计规模很小,主要用于科学计算,侧重节省空间和编程技巧,基本没有开发过程的相关文档。

1956—1970 年,社会各行业对软件的需求量剧增,程序规模和应用范围扩大,出现了“软件作坊”的开发组织形式,将程序商业化,进入软件设计阶段。虽然该阶段计算机技术发展迅速,高级编程语言层出不穷,但是软件产品的质量不高,往往不能满足用户需要,导致了“软件危机”的产生。“软件危机”主要指软件开发过程陷入了不可控的情况,如软件开发成本和周期超过预算,缺少开发文档,导致维护困难,软件质量不可靠等。“软件危机”的出现是因为软件规模扩大,复杂度增高,缺乏正确的开发方法,软件开发过程混乱。

自 1970 年起,软件开发人员开始了解决“软件危机”的征程,借鉴其他行业的生产管理手段,将软件开发进行“工程化”管理,软件开发进入了软件工程阶段。软件工程是将系统化的、严格约束的、量化的工程化方法应用于软件的开发、运行和维护过程,并且随着计算机技术的发展,研究适合软件开发的工程化方法。

目前软件质量主要从适用性、有效性、可修改性、可靠性、可理解性、可维护性、可重用性、可移植性、可追踪性、可互操作性等几个方面衡量,良好的软件工程方法有助于提高软件产品的质量和开发效率,减少维护的困难。

1.2 软件工程的原理

著名软件工程专家 B. Boehm 综合有关专家和学者的意见,总结了多年来开发软件的经验,于 1983 年提出了软件工程的七条基本原理,将工程化方法有效应用到软件开发过程中。

(1) 用分阶段的生存周期计划进行严格的管理

将软件开发与维护的漫长过程划分成若干个阶段,并制订出切实可行的计划,然后严格按照计划对软件的开发与维护工作进行管理。

(2) 坚持进行阶段评审

为软件开发周期的每个阶段制定评审标准,对每个阶段都进行严格的评审,以便尽早发

现在软件开发过程中所犯的错误。

(3) 实行严格的产品控制

当用户改变需求时,必须实行严格的产品控制,主要通过基线配置管理,保持软件各个配置成分的一致性。一切有关修改软件的建议,特别是涉及对基准配置的修改建议,都必须按照严格的规程进行评审,获得批准以后才能实施修改。

(4) 采用现代程序设计技术

采用先进的技术既可提高软件开发的效率,又可提高软件维护的效率。

(5) 软件工程结果应能清楚地审查

软件产品是逻辑产品,软件产品的开发过程的工作进展情况可见性差,难以准确度量。为了更好地进行管理,应该根据软件开发项目的总目标及完成期限,规定开发组织的责任和标准,从而使得所得到的结果能够清楚地审查。

(6) 开发小组的人员应该少而精

开发小组人员的素质和数量是影响软件产品质量和开发效率的重要因素,组成少而精的开发小组是软件工程的一条基本原理。

(7) 承认不断改进软件工程实践的必要性

积极主动地采纳新的软件技术,不断总结经验,保证软件开发与维护的过程能赶上时代前进的步伐,能跟上技术的不断进步。

1.3 软件生命周期

软件生命周期是从提出软件开发需求开始,经历软件开发过程,直到软件投入使用,最终被淘汰为止的整个时间。软件生存周期大致分为软件定义、软件开发、软件维护三个阶段,其中定义阶段包括问题定义、可行性研究和需求分析,开发阶段包括概要设计、详细设计、编程和测试,维护阶段包括运行与维护,如图 1.1 所示。将整个软件生命周期划分为若干阶段,每个阶段都有严格的定义、工作内容和审查标准,开发人员可以按照生命周期按部就班地进行软件开发,使规模大、活动多、管理复杂的软件开发活动变得容易控制和管理,以提高软件的质量。

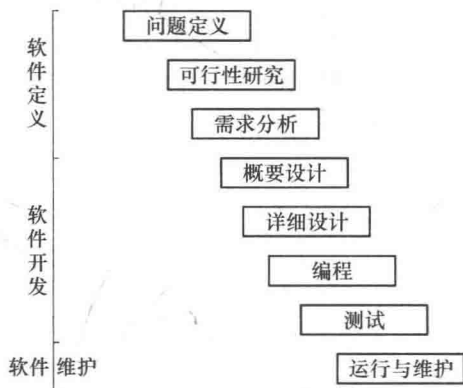


图 1.1 软件生命周期

(1) 问题定义:用户提出软件开发需求以后,经过初步的调查和访问,软件开发方与需求方共同讨论,明确软件的实现目标、规模及类型。

(2) 可行性研究:经过粗略的分析和设计,得到若干个系统方案,对每一个解决方案都可以从技术上、经济上、社会因素上分析可行性,确定软件开发的可行性。

(3) 需求分析:经过详细调研,确定系统的功能、性能和其他方面的要求,建立逻辑模型,回答“系统必须做什么”,并制订系统测试计划。

(4) 概要设计:根据需求分析结果,确定系统的事务处理流程,建立系统总体结构和全局数据

结构,划分功能模块,定义各个功能模块的接口,并制订集成测试计划。

(5) 详细设计:给出软件的每一个构成元素,如程序模块、存储数据结构、输入/输出数据等的实际方案的详细策划,并制订单元测试计划。

(6) 编程:编写程序源代码,进行单元测试和调试,编写用户手册。

(7) 测试:按照测试计划完成集成测试和系统测试,编写测试报告。

(8) 运行与维护:在软件运行过程中,根据用户需求,对软件进行维护,对修改进行配置管理,记录修改记录和故障报告。

在实际软件开发过程中,并不是严格按照生命周期的定义来进行各个阶段的工作的,而是根据软件特点和开发技术等因素,按照某种软件开发模型进行软件开发。软件开发模型是由软件工程师在具体的实践工程活动当中设计并提炼出来,能够覆盖软件生命周期的基本阶段,用一个合理的框架来规范描述。目前典型的软件开发模型如表 1.1 所示。

表 1.1 软件开发模型

开发模型	特 点	适用场合
瀑布模型	线性模型,每一阶段必须完成规定的文档	需求明确的中、小型软件开发
快速原型模型	用户介入早,通过迭代完善用户需求,原型废弃不用	需求模糊的小型软件开发
增量模型	每次迭代完成一个增量,可用于 OO 开发	容易分块的大型软件开发
螺旋模型	典型迭代模型,重视风险分析,可用于 OO 开发	具有不确定性大型软件开发
喷泉模型	以用户需求为动力、以对象为驱动力的模型,支持软件复用及多项开发活动的集成	支持面向对象的开发方法
敏捷模型	一种轻量、高效、低风险、更强调团队协作和沟通的开发方式	中小型开发团队,客户需求模糊或多变
统一过程 RUP	基于构件,用例驱动、以基本架构为中心、迭代和增量,时间上分为四个连续的阶段,即初始阶段、细化阶段、构建阶段和交付阶段	基于构件开发,支持面向对象
构件集成模型	软件开发与构件开发平行进行	领域工程、行业的中型软件开发

1.4 软件开发方法

软件开发方法是认识、理解和描述软件系统结构的一种思维模式,是软件开发所遵循的办法和步骤,以保证所得到的运行系统和支持的文档满足质量要求。软件开发人员在实践过程中,总结了一些有效的开发方法,如 Parnas 方法、结构化开发方法、面向数据结构的软件开发方法、PAM 问题分析法、面向对象的软件开发方法、可视化开发方法、ICASE 方法、软件重用和组件连接方法,本书重点介绍结构化开发方法和面向对象的开发方法。

结构化开发方法是由 E. Yourdon 和 L. L. Constantine 提出的,强调系统结构的合理性以及所开发的软件的结构合理性,主要是面向数据流的,因此也被称为面向功能的软件开

发方法或面向数据流的软件开发方法。针对软件生命周期各个不同的阶段,结构化开发方法有结构化分析(SA)、结构化设计(SD)和结构化程序设计(SP)等组成。结构化设计方法是以自顶向下,逐步求精为基点,以模块化、抽象、逐层分解求精,信息隐蔽化、局部化和保持模块独立为准则,以数据流图、数据字典、结构化语言、判定表、判定树等图形表达为主要手段,强调开发方法的结构合理性和系统的结构合理性的软件分析方法。

面向对象开发方法是一种把面向对象的思想应用于软件开发过程中,指导软件开发活动的系统方法。面向对象开发方法是基于所研究的问题,对问题空间进行自然分割,识别其中的对象及其相关关系,以“对象”为基础对软件进行处理的开发方法。它按照人类自己认识客观世界的一般方法和一般思维去分析和解决问题,是人类认识过程的计算机模拟。面向对象开发方法就是基于对象概念,以对象为中心,以类和继承为构造机制,来认识、理解、刻画客观世界和设计、构建相应的软件系统。

结构化开发方法和面向对象开发方法有如下区别。

(1) 处理问题时的出发点不同

结构化方法强调过程抽象化和模块化,以过程为中心;面向对象方法强调把问题域直接映射到对象及对象之间的接口上,用符合人们通常思维方式来处理客观世界的问题。

(2) 处理问题的基本单位和层次逻辑关系不同

结构化方法把客观世界的问题抽象成计算机可以处理的过程,处理问题的基本单位是能够表达过程的功能模块,用模块的层次结构概括模块或模块间的关系和功能;面向对象方法是用计算机逻辑来模拟客观世界中的物理存在,以对象的集合类作为处理问题的基本单位,尽可能使计算机世界向客观世界靠拢,它用类的层次结构来体现类之间的继承和发展。

(3) 数据处理方式与控制程序方式不同

结构化方法是直接通过数据流来驱动,各个模块程序之间存在着控制与被控制的关系;面向对象方法是通过用例(业务)来驱动,是以人为本的方法,站在客户的角度去考虑问题。

第 2 章 结构化系统分析与设计方法

本章以某企业的采购业务为案例,按照软件生命周期的划分,详细讲解结构化可行性分析、需求分析、概要设计和详细设计四个阶段的任务及开发过程。

2.1 结构化开发方法概述

结构化方法是最早最传统的软件开发方法,也是迄今为止信息系统中应用最普遍、最成熟的一种,它引入了工程思想和结构化思想,使大型软件的开发和编程都得到了极大的改善。针对软件生存周期各个不同的阶段,结构化开发方法由结构化分析(SA)、结构化设计(SD)和结构化程序设计(SP)等过程组成。

结构化方法的基本思想可概括为:自顶向下、逐步求精、模块化技术。自顶向下逐层分解,是指在程序设计时,先考虑问题大的方面,在确定了主要方向后,再由表及里深入到问题具体的细节,由易到难,逐层解决问题。这是一个由模糊到清晰,由概括到具体的过程,如图 2.1 所示。而逐步求精是在遇到复杂问题的时候,先设计一些子目标作为过渡,来逐步细化。结构化方法强调功能抽象和模块化,采取了分块处理问题的方法,可以把一个比较复杂的问题分解为若干个容易处理解决的部分,从而降低了问题处理的难度。

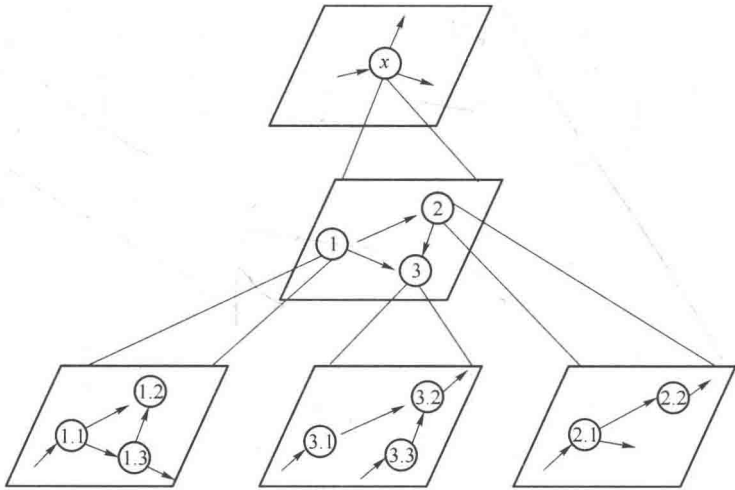


图 2.1 问题的自顶向下逐层分解

2.2 案例介绍

本章主要针对下述案例讲解软件的结构化开发过程。

某企业生产某种商品,组织结构有管理部门、生产部门、采购部门、销售部门、库存部门、财务部门、人事部门等,该案例重点分析采购业务的处理过程。采购部门负责企业生产的原材料的采购,首先,库存部门按照生产需求与库存情况,填写采购申请单,并提交给采购部门;然后,采购部门按照采购申请单,查询原材料信息及供应商信息,制订采购订单,并提交部门审核人进行审核,未通过审核的订单需要重新编订,审核通过的订单交付给指定的供应商,要求供货,同时,采购部门根据合格的采购订单编制采购付款申请单,并提交财务部门;最后,供应商按时供货,并提供供货单,采购部门根据采购订单验证供货单和商品,将符合要求的材料入库,编制采购入库申请单,并提交给库存部门;如果不符合要求,则要求退货,并制定退货单给供应商,同时制定采购退货收款申请单,并提交给财务部门。

2.3 结构化可行性分析

可行性分析是指在软件项目计划阶段,用最小的代价在尽可能短的时间内,研究并确定客户提出的问题的是否有行得通的解决办法。如果问题没有解决方案或者不值得解决,分析员应该建议停止该项目,以避免时间、资源、人力和金钱的浪费;如果问题值得解决,分析员应该提供一个解决方案。

2.3.1 可行性分析任务

项目的可行性分析主要完成如下任务。

- (1) 对当前正在使用的系统或工作方式进行调查研究,进一步确定系统规模和目标。
- (2) 以当前系统为基础,结合用户对问题的描述,导出目标系统的逻辑模型;设计不同的解决方案,然后比较多个备选方案,分析利弊,主要从经济可行性、技术可行性、操作可行性和社会可行性四个方面进行分析,判断原定的系统目标和规模是否现实,系统完成后所能带来的效益是否大到值得投资开发这个系统的程度。
- (3) 最后,编写可行性分析报告,给出结论意见。

2.3.2 案例讲解可行性分析过程

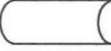
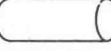
1. 用系统流程图对现有系统进行分析

系统流程图用图形符号以黑盒子形式描绘系统里面的每个部件(程序、文件、数据库、表格、人工过程等),是描绘物理系统的传统工具。系统分析员了解系统业务处理概况的过程后,采用图形的方式描述业务处理过程,它是系统分析员做进一步分析的依据,也是系统分析员、管理员、业务操作员相互交流的工具,利用它可以帮助分析人员找出业务流程中的不合理流向。

系统流程图的绘制是按照系统业务的实际处理步骤和过程进行的,习惯画法是使信息在图中从顶向下或从左向右流动。每个符号代表了系统的一个部分,并没有指明内部处理

细节,箭头表示信息流转的路径或流程。系统流程图常用符号如表 2.1 所示。

表 2.1 系统流程图符号

符号	名称	说 明
	穿孔卡片	表示用穿孔卡片输入或输出,也可表示一个穿孔卡片文件
	文档	通常表示打印输出,也可表示用打印终端输入数据
	磁带	磁带输入/输出,或表示一个磁带文件
	联机存储	表示任何种类的联机存储,包括磁盘、磁鼓、软盘和海量存储器件等
	磁盘	磁盘输入/输出,也可表示存储在磁盘上的文件或数据库
	磁鼓	磁鼓输入/输出,也可表示存储在磁鼓上的文件或数据库
	显示	CRT 终端或类似的显示部件,可用于输入或输出,也可既输入又输出
	人工输入	人工输入数据的脱机处理,例如,填写表格
	人工操作	人工完成的处理,例如,会计在工资支票上签名
	辅助操作	使用设备进行的脱机操作
	通信链路	通过远程通信线路或链路传送数据

在画业务流程图之前,要对现行系统进行详细调查,并写出现行系统业务流程总结。

(1) 库存部门按照生产需求与库存情况,填写采购申请单,并提交给采购部门;然后,采购部门按照采购申请单,查询原材料信息及供应商信息,制订采购订单。

(2) 将采购订单提交部门审核人进行审核,未通过审核的订单需要重新编订,审核通过的订单交付给指定的供应商,要求供货。

(3) 采购部门根据合格的采购订单编制采购付款申请单,并提交财务部门。

(4) 供应商按时供货,并提供供货单,采购部门根据采购订单验证供货单和商品。

(5) 将符合要求的材料入库,编制采购入库申请单,并提交给库存部门;如果不符合要求,则要求退货,并制订退货单给供应商。

(6) 采购部门对不合格的原材料制订采购退货收款申请单,并提交给财务部门。

按照上述流程完成系统流程图,如图 2.2 所示。

2. 推荐方案的可行性分析

对现有系统进行初步分析后,分析人员可以给出不同的解决方案,然后对方案进行可行性分析,主要从经济可行性、技术可行性、操作可行性和社会可行性四个方面进行分析,得到最好的推荐方案。

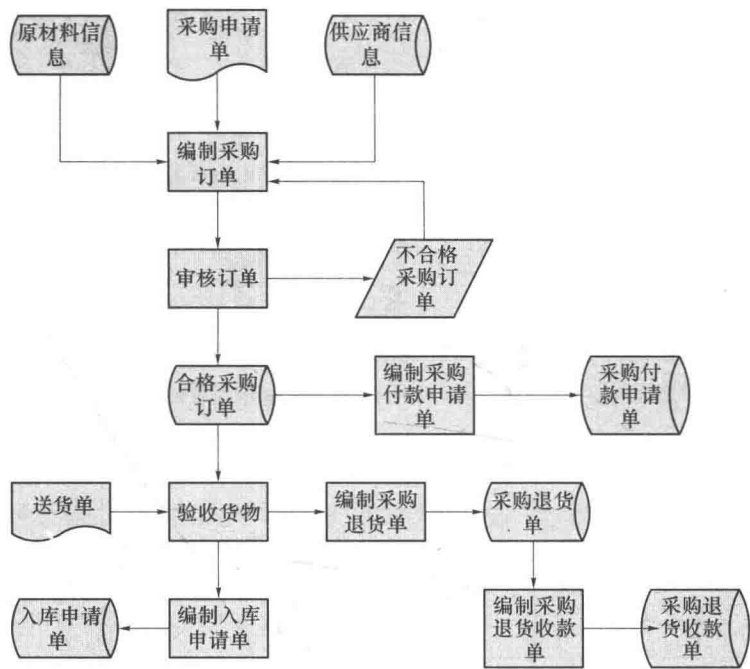


图 2.2 采购系统流程图

经济可行性:分析员需要进行成本/效益分析,从经济角度判断系统开发是否“合算”。所谓成本,包括购置并安装软、硬件及有关设备的费用、系统开发费用、系统安装、运行及维护的费用、人员培训费用。而效益是指系统为用户增加的收入或为用户节省的开支,还有给潜在用户心理上造成的影响,也是间接效益。

技术可行性:进行技术风险评价,通过对备选方案进行一系列的试验、评审和修改,最后由项目管理人员做出是否进行系统开发的决定。主要从开发者的技术实力、以往工作基础、问题的复杂性等出发,判断系统开发在时间、费用等限制条件下成功的可能性。如果开发技术风险很大,或者模型演示表明当前采用的技术和方法不能实现系统预期的功能和性能,或者系统的实现不支持各子系统的集成,则项目管理人员可以做出停止系统开发的决定。

操作可行性:方案要符合用户的操作习惯,能够与用户业务的手工作业部分以及相关系统平滑连接,融合为一个有机的系统。通过软硬件提供给用户的操作方式,既要便于计算机实现,又要符合人机功效学原理。

社会可行性:从法律、社会效益等更广泛的方面研究每种方案的可行性。分析要开发的项目是否存在侵权、妨碍等社会责任问题;系统运行方式在用户组织内是否行得通;现有的管理制度、人员素质、操作方式是否可行等。

2.4 结构化系统需求分析

需求分析的主要任务是确定系统“要做什么”,通过软件开发人员与用户的交流和讨论,进行细致的调查分析,准确理解用户的功能需求、性能需求、运行环境需求和操作界面需求,将用户非形式的需求陈述转化为完整的需求定义,再由需求定义转换到相应的形式功能规