

软件工程 课程设计

李龙澍 郑 诚 主编
杨为民 李学俊 程 凡 姚 晟 编著



第2版



机械工业出版社
China Machine Press

高等院校计算机课程设计指导丛书

软件工程 课程设计

李龙澍 郑 诚 主编
杨为民 李学俊 程 凡 姚 晟 编著



第2版



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

软件工程课程设计 / 李龙澍等编著. —2 版. —北京: 机械工业出版社, 2016.9
(高等院校计算机课程设计指导丛书)

ISBN 978-7-111-54876-8

I. 软… II. 李… III. 软件工程 - 课程设计 - 高等学校 - 教学参考资料 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2016) 第 219740 号

本书遵循软件工程课程设计的基本要求, 独立于具体的软件工程教材, 从实际应用案例出发, 严格按照软件工程设计规范, 逐章给出软件工程课程设计的方法和思路, 重点讲解软件的分析、设计、编码、测试和维护技术, 目的是让读者掌握软件开发的基本过程和核心技能, 加强工程设计能力, 提高自学能力、创造能力和团队协作能力。

本书可以作为高等院校计算机及相关专业软件工程课程设计的教材或教学参考书, 也可以供软件开发人员和有关技术人员阅读使用。

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)	
责任编辑: 迟振春	责任校对: 董纪丽
印刷: 北京诚信伟业印刷有限公司	版次: 2016 年 9 月第 2 版第 1 次印刷
开本: 185mm×260mm 1/16	印张: 16.75
书号: ISBN 978-7-111-54876-8	定 价: 39.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换	
客服热线: (010) 88378991 88361066	投稿热线: (010) 88379604
购书热线: (010) 68326294 88379649 68995259	读者信箱: hzjsj@hzbook.com

版权所有·侵权必究
封底无防伪标均为盗版
本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

高等院校 计算机课程设计指导丛书

专家指导委员会

(以姓氏拼音为序)

- | | |
|-----|------------|
| 陈向群 | (北京大学) |
| 陈鸣 | (解放军理工大学) |
| 戴葵 | (国防科技大学) |
| 何钦铭 | (浙江大学) |
| 廖明宏 | (哈尔滨工业大学) |
| 林闯 | (清华大学) |
| 刘振安 | (中国科技大学) |
| 马殿富 | (北京航空航天大学) |
| 齐勇 | (西安交通大学) |
| 宋方敏 | (南京大学) |
| 汤庸 | (中山大学) |
| 王立福 | (北京大学) |
| 吴功宜 | (南开大学) |
| 赵一鸣 | (复旦大学) |

联络人 温莉芳

第2版前言

现代信息技术的发展日新月异,“互联网+”在改变着人们的思维方式,大数据、云计算、物联网深入人们的工作和生活空间,日益普遍使用计算机来生产、处理、交换和传播各种形式的信息。信息技术体系已经成为一个为实现现代化战略目标而采用的综合技术结构,从最初主要偏向科学与工程应用,发展成为当今科学与工程与大数据信息混合应用的阶段。为了适应大数据信息的应用,需要一个严密的管理控制机制,这就是计算机软件。软件工程是一门研究用工程化方法构建和维护有效的、实用的和高质量的软件的学科。

软件工程课程设计,可以让学生根据课堂讲授内容,做相应的自主练习,消化课堂所讲解的内容,通过调试典型案例初步感受软件开发过程,通过完成课程设计教材中的案例,逐渐培养软件设计能力和解决实际问题的能力。

随着计算机软件的规模和复杂度的不断增大,开发人员增加,开发时间增长,这些都增加了软件工程的难度,也促进了软件工程的日臻完善。本书第1版经过全国各地师生5年教与学的实践,收到了众多的好评与鼓励,同时也得到了一些有益的修改建议。为了使本书更好地满足实践教学的需要,更好地做到思路清晰,通俗易懂,由浅入深,重在实用,更加强调增强学生的工程设计能力,让学生学得会、用得上,我们深感需要对部分内容进行适当的修订。

在教材的修订过程中,作者对软件工程课程设计的知识体系和核心内容再次进行了深入的探讨,综合考虑软件工程课程设计的整体结构和软件工程初学者的接受能力,为了更加适应读者的学习需求,认真调整了案例内容和表述方式,使得软件结构更加满足大数据、云计算、物联网等现代技术的需求,为培养卓越软件工程师打下良好的软件工程实践基础。

安徽大学对本书的修订工作从人力和物力上给予了大力支持。全国各地的读者对本书第1版给予了高度评价,同时也提出了许多宝贵的意见和建议,对读者的厚爱和无私帮助表示衷心的感谢。我们的许多同行和学生,对第2版书稿提出了大量宝贵意见,在此表示衷心的感谢。

一切为了读者,为了一切读者,为了读者一切,是我们的心愿和目标,但是由于作者水平有限,难免出现这样或那样的错误与不足,敬请广大读者不吝赐教。

作者

2016年6月于安徽大学

第1版前言

随着计算机科学技术的迅速发展, 计算机应用范围越来越广, 计算机软件的需求量也越来越大, 软件产业蓬勃兴起。软件产业化的一个重要方面是软件开发工程化, 采用先进的工程化方法进行软件开发是实现软件产业化的关键技术手段。软件工程是将系统性的、规范化的、可量化的方法应用于软件的开发、运行和维护, 其重点在于软件的分析与评价、规格说明、设计和演化, 同时还涉及创新、管理、标准、个人技能、团队协作和专业实践等。它应用工程的概念、原理、技术和方法, 以及科学的开发技术和管理方法来开发软件。软件工程的目标是提高软件产品的质量和软件开发效率, 降低软件维护的难度。

加强实践环节、培养创新人才已经成为全国高校本科生培养的大方向。从计算机学科各相关专业的特点来看, 更强调课程体系整体优化, 立足系统, 软硬结合, 加强实践, 注重创新和发展学生个性。本书是作者结合多年软件工程课程设计的教学经验编写的, 针对学生学习中遇到的问题, 反复修正教学内容, 总结启发式教学方法, 强调软件工程课程设计的系统整体性和实践性, 面向学生、贴近实际, 力争让学生学得会、记得牢、用得上。

本书的主要特点是:

- 思路清晰。以案例为线索, 每个案例都贯穿软件开发的各个阶段, 重点放在训练读者分析问题和解决问题的能力上。
- 通俗易懂。将复杂的概念用读者容易理解的简洁语言描述出来, 不依赖于某一本软件工程教材, 具有通用性。
- 重在实用, 强调亲自动手实践。从需求分析到编码测试, 由浅入深, 让读者做完课程设计方案后, 能够分析、设计和具体实现软件系统。

本书采取案例驱动, 每章都以期刊管理系统、图书管理系统、网上商城管理系统、饭卡管理系统、研究生培养管理系统5个案例为线索, 演示软件开发的全部过程。程序代码采用不同的流行语言, 以满足各种读者的需要。

全书共分6章。第0章概述软件工程的目标和原则, 以及软件工程课程设计的目标、结构和评价标准; 第1章是软件系统分析, 包括可行性研究的任务和步骤, 系统分析的任务, 各案例系统的任务描述、数据流图、数据字典、E-R图、性能要求和运行环境, 面向对象分析, 评价标准; 第2章是软件系统设计, 包括软件的设计原则和方法、总体设计和详细设计、面向对象设计、评价标准; 第3章是软件系统编码, 主要包括编码和评价标准; 第4章是软件测试, 讲述软件测试的目的和步骤、单元测试、集成测试、面向对象测试、评价标准; 第5章是软件维护, 阐述各个案例系统的维护过程和评价标准。

本书由李龙澍主持编写, 参与编写工作的还有郑诚、姚晟、程凡、杨为民、李学俊、徐怡, 具体分工如下: 李龙澍(第0章)、郑诚(期刊管理系统)、程凡(图书管理系统)、杨为民(网上商城管理系统)、姚晟(饭卡管理系统)、李学俊(研究生培养管理系统), 徐怡参加了第2章的编写和修改工作。中国科学技术大学刘振安教授以及安徽大学计算机科学与技术学院的教师、领导和学生对本书的编写工作给予了大力支持, 并提出了许多宝贵意见, 在此表示衷心感谢。

由于作者水平有限, 难免出现一些疏漏和错误, 殷切希望读者提出宝贵的建议和修改意见。

作者

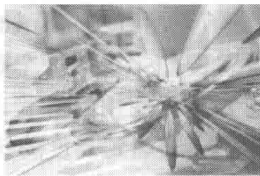
2009年12月于安徽大学

目 录

专家指导委员会	
第2版前言	
第1版前言	
第0章 概述	1
0.1 软件工程的目标和原则	1
0.1.1 软件工程的目标	2
0.1.2 软件工程的原则	2
0.2 软件工程课程设计目标	3
0.3 软件工程课程设计结构	4
0.4 软件工程课程设计的主要任务和评价标准	5
0.5 本教材的主要特色和使用方法	6
0.5.1 本教材的主要特色	6
0.5.2 本教材的使用方法	7
第1章 软件系统分析	8
1.1 概述	8
1.1.1 可行性研究的任务和步骤	8
1.1.2 软件系统分析的任务	9
1.2 期刊管理系统需求分析	11
1.2.1 系统任务概述	11
1.2.2 数据流图	12
1.2.3 数据字典	14
1.2.4 加工说明	14
1.2.5 E-R图	16
1.2.6 性能要求	16
1.2.7 运行环境	17
1.3 图书管理系统需求分析	17
1.3.1 系统任务概述	17
1.3.2 功能需求	17
1.3.3 数据流图	18
1.3.4 数据字典	20
1.3.5 E-R图	22
1.3.6 性能要求	24
1.4 网上商城管理系统需求分析	24
1.4.1 系统任务概述	24
1.4.2 功能需求	24
1.4.3 数据流图	25
1.4.4 数据字典	30
1.4.5 E-R图	32
1.4.6 其他相关要求	32
1.5 饭卡管理系统需求分析	33
1.5.1 系统任务概述	33
1.5.2 工作原理	33
1.5.3 流程图	34
1.5.4 数据流图	35
1.5.5 数据字典	39
1.5.6 性能要求	42
1.5.7 运行环境	42
1.6 面向对象分析	42
1.6.1 概述	42
1.6.2 研究生培养管理系统需求	43
1.6.3 功能模型	44
1.6.4 对象模型	50
1.6.5 动态模型	52
1.6.6 定义服务	54
1.7 评价标准	55
第2章 系统设计	56
2.1 概述	56
2.1.1 总体设计过程	56
2.1.2 总体设计原则	57
2.1.3 详细设计过程	59
2.1.4 系统设计的方法	61
2.2 期刊管理系统设计	64
2.2.1 总体设计	64
2.2.2 详细设计	67
2.3 图书管理系统设计	70
2.3.1 总体设计	70
2.3.2 详细设计	71
2.4 网上商城管理系统设计	83

2.4.1 总体设计	83	4.2.3 测试过程	221
2.4.2 详细设计	83	4.3 图书管理系统测试	223
2.5 饭卡管理系统设计	92	4.3.1 概述	224
2.5.1 总体设计	92	4.3.2 测试方法和步骤	224
2.5.2 详细设计	92	4.3.3 测试过程	224
2.6 面向对象设计	100	4.4 网上商城管理系统测试	227
2.6.1 概述	100	4.4.1 概述	227
2.6.2 研究生培养管理系统结构设计	101	4.4.2 测试方法和步骤	228
2.6.3 模块及人机界面设计	101	4.4.3 测试过程	228
2.6.4 数据设计	108	4.5 饭卡管理系统测试	231
2.6.5 对象设计	111	4.5.1 概述	231
2.7 评价标准	113	4.5.2 测试方法和步骤	231
第3章 系统编码	115	4.5.3 测试过程	232
3.1 概述	115	4.6 面向对象测试	234
3.2 期刊管理系统编码	117	4.6.1 概述	234
3.2.1 系统登录	117	4.6.2 研究生培养管理系统测试	235
3.2.2 读者功能模块	119	4.7 评价标准	238
3.2.3 管理员功能模块	120	第5章 软件维护	241
3.3 图书管理系统编码	128	5.1 概述	241
3.3.1 程序设计语言和数据库系统的 选择	128	5.1.1 软件维护类型	241
3.3.2 系统模块的编码实现	130	5.1.2 软件维护过程	242
3.4 网上商城管理系统编码	149	5.2 期刊管理系统维护	242
3.4.1 程序设计语言和数据库系统的 选择	149	5.2.1 软件维护	242
3.4.2 系统模块的编码实现	151	5.2.2 系统备份	243
3.5 饭卡管理系统编码	173	5.3 图书管理系统维护	244
3.5.1 程序设计语言和数据库系统的 选择	173	5.4 网上商城管理系统维护	245
3.5.2 系统模块的编码实现	174	5.4.1 运营保障	245
3.6 面向对象系统实现	201	5.4.2 维护要素	246
3.6.1 概述	201	5.4.3 维护内容	246
3.6.2 研究生培养管理系统编码	202	5.4.4 信息备份	248
3.7 评价标准	214	5.5 饭卡管理系统维护	250
第4章 软件测试	216	5.5.1 软件维护	250
4.1 概述	216	5.5.2 硬件维护	251
4.1.1 软件测试的目的	216	5.5.3 数据维护	251
4.1.2 软件测试的步骤	217	5.5.4 维护的管理和步骤	251
4.2 期刊管理系统测试	220	5.5.5 系统备份和恢复	252
4.2.1 概述	220	5.6 面向对象维护	254
4.2.2 测试方法和步骤	220	5.6.1 概述	254
		5.6.2 研究生培养管理系统维护	254
		5.7 评价标准	256
		参考文献	257

第 0 章 概 述



随着计算机应用的日益普及和“互联网+”社会发展新形态的推进，人们对软件的需求量急剧增加。但是，计算机软件开发技术却远远没有跟上硬件技术的发展，使得软件开发的成本逐年剧增，更为严重的是，软件的质量没有可靠的保证。软件开发的速度与计算机普及的速度不相适应，软件的质量与应用的要求不相适应，软件开发技术已经成为影响计算机系统发展的“瓶颈”。软件开发人员执行工程规范不严格，团队意识不强，导致软件工程处于相对落后的位置，软件开发过程无法适应工业化、工程化的需求。软件工程的教学普遍存在着以课堂教学为主，理论和实践明显脱节的情况，让学生学习软件工程课程设计是非常必要的，本章将介绍软件工程课程设计的目标、结构和任务，以及本教材的特色和使用方法。

0.1 软件工程的目标和原则

软件工程是一门研究用工程化方法构建和维护有效的、实用的和高质量的软件的学科。软件工程是应用计算机科学、数学及管理科学等原理开发软件的工程。因此，软件工程是一门用科学知识和技术原理来定义、开发、维护软件的学科。它应用工程的概念、原理、技术和方法，以及科学的开发技术和管理方法来开发软件。

软件工程学研究如何应用科学理论和工程技术来指导计算机软件的开发与维护。它的基本目标是制定一套科学的工程方法，设计一套方便、实用的工具系统，提高软件开发的效率和质量。

软件工程学的三个基本要素是：软件定义、开发、维护的方法；软件定义、开发、维护的工具；软件定义、开发、维护中的管理措施。这三个要素简称方法、工具、管理。

软件工程具有以下特性：

1) 研究开发“大”程序的方法。“大”程序和“小”程序是相对概念，所谓“大”程序就是几个人在较长时间内研制完成的程序。软件工程学研究高效率、高质量地建造大程序的方法。

2) 降低软件的复杂性。近年来，软件规模越来越大，软件的逻辑结构越来越复杂，软件工程学把难解决的大问题化解成若干个容易解决的小问题，并且让各个小问题之间保持简单的接口协议。

3) 适应软件的易变性。软件往往模拟现实世界解决问题的方法，如果现实世界解决问题的方法变了，软件也应该做相应的变化。例如，对于学生成绩管理系统，学年制以年级为基本教学管理单位，改为学分制后就要以课程的学分作为基本管理单位，这样管理软件就要做相应的变动。另一方面，随着计算机硬件和操作系统的改变，软件也要做相应的变化。

4) 提高软件的开发效率。自软件诞生以来，每种语言的诞生、每种软件开发方法的提出、每种管理技术的改进，都以提高软件的开发效率作为主要目标之一。目前提高软件的开发效率仍然是软件工程的重要问题。

5) 加强软件开发人员的协作。由于软件规模的庞大，软件开发必须多人协作。应严格规定每个人应该承担的任务，制定一套严格的功能界定规范、接口标准规范和评价管理办法。

6) 有效地支持用户。开发的软件最终是由用户使用的，软件提供的功能应该能完成用户交给的任务。另一方面，应该写用户手册和培训材料，让用户很容易学会软件的使用。

7) 两种文化背景的统一。计算机软件开发人员和使用软件的用户往往具有两种不同的文化背景。例如, 构建一个财务管理系统, 计算机软件开发人员是掌握软件工程、程序设计语言等知识的专家, 使用财务管理系统的人员是掌握会计学知识的专家, 这两种知识背景的人员在对系统的理解上要达到高度的统一, 才能开发出满足用户需求的软件。

0.1.1 软件工程的目标

软件工程的目标是提高软件产品的质量和软件开发效率, 降低软件维护的难度, 最终实现软件生产的自动化。要达到这个目标, 必须注重考虑下面几个方面的问题:

1) 正确性: 运行的软件能够准确无误地执行用户要求的各种功能, 满足用户要求的各种性能指标。

2) 可靠性: 有时也称为健壮性, 就是在运行环境(包括硬件、操作系统等)出现小故障, 或者人为操作不当的情况下, 不会导致软件系统失效。对于一些重要领域的计算机管控系统以及自动控制系统, 可靠性要求会更高。

3) 有效性: 软件系统能在一定的时间资源和空间资源环境下, 完成规定的任务。要求开发人员在固定的计算机硬件和操作系统环境下, 开发出满足用户需求的软件系统。

4) 可修改性: 软件产品投入使用后, 允许对软件系统进行修改, 而不增加系统的复杂性。在软件开发过程中, 可以随时对软件进行调试、修改; 对投入使用的软件, 修正错误、改进性能、增加功能以及适应硬件环境的变化, 都需要修改软件。可修改性是对软件维护的支持。

5) 可理解性: 包括两个方面的内容, 一是软件系统结构清晰、容易理解; 二是程序算法功能清晰、容易读懂。可理解性有助于控制软件系统的复杂性, 提高软件的可维护性。

6) 可重用性: 软件中的某些部分可以在系统的多处重复使用, 或者在多个系统中使用。可重用的部分有的可以直接使用, 有的需要稍微修改后使用。可重用性不仅有助于降低软件的开发和维护费用, 也有助于提高软件的质量和开发效率。

7) 可适应性: 体现软件在不同的硬件和操作系统环境下的适应程度, 适应性强的软件要求在最流行的硬件和操作系统环境下开发运行, 这种软件容易推广应用。

8) 可移植性: 体现了软件从一种计算机环境移动到另一种计算机环境的难易程度。软件开发过程中要尽量使用不依赖于或者很少依赖于计算机硬件和操作系统的计算机语言编写程序。

9) 可跟踪性: 包括两个方面的内容, 一是可以根据软件开发的文档对设计过程进行正向跟踪或逆向跟踪; 二是软件测试和维护过程中对程序的执行进行跟踪, 根据跟踪情况, 分析程序执行的因果关系。

10) 互操作性: 多个软件相互通信, 协作完成任务的能力。软件开发要遵循某种国际标准, 互操作性支持网络环境下分布式软件的开发。

0.1.2 软件工程的原则

为了解决软件危机, 达到软件工程的既定目标, 软件的工程设计、工程支持以及工程管理必须遵循以下四条基本原则:

1) 选取适宜的开发模型。该原则与软件系统设计有关。在系统设计过程中, 软件需求、硬件需求以及其他因素间是相互制约、相互影响的, 经常需要权衡。因此, 必须认识到需求定义的易变性, 采用适当的开发模型, 保证软件产品满足用户的要求。

2) 采用合适的设计方法。在软件系统设计中, 通常需要考虑软件的模块化、抽象与信息隐蔽、局部化、一致性以及适应性等特征。合适的设计方法有助于这些特征的实现, 以达到软件工程的目标。

3) 提供高质量的工具支撑。工欲善其事, 必先利其器。在软件工程中, 软件工具与开发环境对软件过程的支持非常重要。软件工程项目的质量与开销直接取决于对软件工程所提供的支撑质量和效用。

4) 高度重视软件开发过程的管理。软件工程的管理直接影响可用资源的有效利用,只有有效管理了软件过程,才能生产满足目标的软件产品以及提高软件组织的生产能力,实现有效的软件工程。

在软件开发过程中,必须遵循下列软件工程原则:

1) 抽象:抽取反映事物本质的特性、行为和与其他事物之间的关系,忽略其他非本质性的细节问题。抽象可以把庞大的问题分解成具有层次结构的简单小问题。

2) 信息隐蔽:将一个相对独立的功能在软件中设计成一个独立的模块,模块内部的算法和数据是封闭的,其他模块不能直接访问。模块有一个简洁的接口,模块之间只有通过接口相互访问,一个模块不能直接调用另一个模块中的程序,也不能直接使用另一个模块中的数据。

3) 模块化:模块是具有独立命名并且可以独立访问的程序段,如程序中的函数。模块化是把软件划分成独立的模块,每个模块完成一个子功能。模块化有助于实现抽象和信息隐蔽。模块的大小要适中,模块过大会使模块内部结构复杂,软件开发和维护都比较困难,也不利于软件重用;模块过小会使模块之间的联系变得复杂,也增大了开发和维护软件的难度。

4) 局部化:将一个模块内使用的数据和操作集成在这个模块内部,使其他模块不能直接访问,避免产生不必要的错误,这样便于开发、维护和使用。

5) 一致性:包括软件系统的各个模块的概念、符号、术语保持一致;程序内部接口风格一致;软件与硬件接口风格一致;用户界面风格一致;文档与功能、性能保持一致等。

6) 完备性:软件系统完全实现用户需求的功能、性能,在非正常环境下系统一般不失效。例如,突然停电,系统不会造成数据混乱;用户输入错误,系统能给出出错提示信息,并且正常运行。

7) 可验证性:包括在软件开发或维护的过程中,对每个阶段的进展情况进行测评;在软件系统运行过程中,对每个功能进行验证。

要达到软件工程目标,需要解决正确性、可靠性、有效性、可修改性等软件工程目标的10个问题,而遵循以上软件工程的7条基本原则,有助于达到软件工程目标,提高软件产品的质量和软件开发、维护的效率。

0.2 软件工程课程设计目标

软件工程课程设计可以培养学生完整、严格的软件工程观念、意识和能力。一般来讲,课程设计比教学实验更复杂一些,涉及的深度更广一些,并更加注重实际应用,重点培养学生的软件开发能力、软件工程素质和软件项目管理能力,目的是通过课程设计的综合训练,培养学生实际分析问题和解决问题的能力,最终目标是通过软件工程课程设计的形式,帮助学生系统掌握该门课程的主要内容,使学生了解软件开发和软件管理的思维模式和行为方式,更好地完成软件工程课程的教学任务。另外,软件工程课程设计中一般完成较大的综合设计,可以分成几个小项目供学生分工合作,以培养学生的团队协作精神。

软件工程学科已经形成了一套富有成效的软件开发方法、软件开发工具和组织管理措施,要真正掌握并熟练运用软件工程的方法进行软件开发,必须有针对性地进行训练。软件工程课程设计从完整的软件系统开发、维护和管理角度出发,按照软件生命周期的阶段进行划分,将软件工程涉及的理论方法通过一系列的课程设计课题进行综合训练,锻炼学生分析问题、设计模型、编写程序、测试、维护和管理等实际动手能力,培养学生的团队协作精神,使学生不仅具有扎实的基本理论,还具有较强的基本技能和良好的基本素质,从而培养知识、能力、素质三者协调发展的具有创新意识的高科技人才。

软件工程课程设计是综合性的实践活动,其主要目的是使学生通过实践训练进一步掌握软件

工程的方法和技术,提高软件开发的实际能力,培养创造性的工程设计能力和分析问题、解决问题的能力。

通过软件工程课程设计,可以促进學生有针对性地主动学习和查阅有关软件工程的基本教学内容及相关资料,从而实现如下目标:

1) 完成从理论到实践的知识升华过程。学生通过软件开发的实践进一步加深对软件工程方法和技术的了解,将软件工程的理论知识运用于开发的实践,并在实践过程中逐步掌握软件工具的使用。

2) 提高分析实际问题和解决实际问题的能力。软件工程课程设计是软件工程的一次模拟训练,学生通过软件开发的实践积累经验,提高分析问题和解决问题的能力。

3) 培养创新能力。软件工程课程设计提倡和鼓励开发过程中使用新方法和新技术,激发学生实践的积极性和创造性,开拓思路设计新算法,培养创造性的工程设计能力。

4) 培养学生的团队协作精神,建立群体共识。软件工程是一项系统工程,只有靠集体的有效协作才能完成,在软件开发的过程中可以让学生充分体会到团队协作的重要性。

总之,软件工程课程设计的目标就是让学生在实践中学会开发软件。

0.3 软件工程课程设计结构

软件工程课程设计的任务是要求学生针对具体的软件工程项目,完成从软件工程管理、需求分析、总体设计、详细设计、编码、测试、维护等各阶段的工作,使学生进一步理解和掌握软件开发模型、软件生命周期、软件过程等理论在软件项目开发过程中的意义和作用,培养学生按照软件工程的原理、方法、技术、标准和规范进行软件开发的能力,培养学生的合作意识和团队精神,培养学生对技术文档的编写能力,从而使学生提高软件工程的综合能力,提高软件项目的管理能力。

本书选择了5个不同的软件开发项目,其中4个采用传统的软件开发方法,1个采用面向对象的软件开发方法。每个项目都涵盖软件开发过程中的所有核心知识点,项目的可操作性强,力图给学生展现一个清晰的软件开发过程,让学生在训练过程中全面掌握软件的开发技术。

本书采取案例驱动的方法,每章都以5个案例为线索,循序渐进、深入浅出、手把手地教会学生开发软件。其中,采用传统的软件开发技术的4个案例是:案例1“期刊管理系统”,案例2“图书管理系统”,案例3“网上商城管理系统”,案例4“饭卡管理系统”;采用面向对象的软件开发技术的案例5是“研究生培养管理系统”。具体说明如下:

1. 软件系统分析(第1章)

软件生命周期方法学把软件定义时期分为软件定义、可行性研究和需求分析三个阶段。软件系统分析训练软件定义时期的基本技术和基本方法,重点是软件需求分析,关键技术是数据流图和数据字典。软件需求分析是软件定义时期的最后一个阶段,它的基本任务是准确地回答“系统必须做什么”这个问题。需求分析并不是确定系统怎样完成它的工作,而仅仅是确定系统必须完成哪些工作,也就是对目标系统提出完整、准确、清晰、具体的要求。需求分析的结果是系统开发的依据,关系到整个软件工程的成败和软件产品的质量,因此需求分析对于整个软件的开发具有重要意义。软件定义是开发大型软件前用户给出的大概要求,一般不需要形式描述。部分软件工程教材中把软件的可行性研究和系统开发计划与软件的需求分析分开介绍,本书中为了清晰起见,把可行性研究和系统开发计划作为需求分析的准备工作放在一起介绍。在面向对象方法学中,软件开发人员清楚用户的需求同样是十分重要的,其关键技术是对象模型、动态模型、功能模型和服务。

2. 系统设计(第2章)

软件需求分析回答了“系统必须做什么”这个问题,确定了系统必须完成哪些工作,但并没

有回答“系统应该怎么做”这个问题,即系统怎样完成它的工作。系统设计是把用户需求转化为软件系统的最重要的技术开发环节,它把用户的需求转换成明确的系统任务描述,得到一个软件设计的总体思路,解决“系统应该怎么做”这个问题。系统设计的优劣从根本上决定了软件系统的质量。

软件需求(包括功能性需求与非功能性需求)是系统设计的基础。系统设计的目标就是使所设计的系统能够被开发方顺利地实现,并且恰如其分地满足用户的需求,使开发方和用户都能获得最大的利益。开发人员不能为了追求技术的先进性,偏离需求开展系统设计工作。

系统设计包括总体设计、详细设计、用户界面设计、数据库设计。总体设计将软件系统设计成相对独立的模块。对于面向对象技术,模块设计将软件功能结构设计的模块(结构元素)对应到软件的对象和类。详细设计要对目标系统的功能进行精确描述,其中数据结构和算法设计一般来说是设计数据的表示及其详细的算法流程。用户界面设计是开发中的一个重要方面,其设计目标是开发者根据自己对用户需求的理解而制定的。数据库设计是设计数据库的表和对这些表中的数据进行操作。一般将用户界面设计、数据库设计并入详细设计中统筹考虑,并且仍然称为详细设计。

3. 系统编码(第3章)

软件开发的最终目标是产生能在计算机上运行的程序。系统编码也称系统实现,编码的目的就是把软件设计的结果翻译成某种程序设计语言的程序。本书将分别应用VC++、C#、Java、Access等通用语言,编写期刊管理系统、图书管理系统、网上商城管理系统、饭卡管理系统、研究生培养管理系统这5个案例的核心程序,使其能够在计算机上运行。在程序设计中要注意程序的整体层次结构,选用合适的标识符,并加入适当的注释,以增加程序的可读性和可维护性。

4. 软件测试(第4章)

在开发软件系统的整个过程中,面对可能遇到的各种错综复杂的情况,总会出现一些不可避免的 errors 和故障。软件系统测试的基本目的就是在软件产品投入使用之前,尽可能多地发现软件产品中存在的各种错误,即消除故障,保证软件的可靠性。

单元测试又称模块测试。在一个软件系统中,每个模块最好都能够完成一个子功能,并且每个模块和同级的其他模块之间没有相互依赖的关系,即每个模块都能够独立地完成自己的功能,这样就可以对每个模块进行单独测试而不需要考虑模块之间的相互关系。模块测试的目的主要是为了保证每个模块作为一个独立的单元能够正确地运行并完成其功能,在模块测试中发现的往往是编码中出现的错误,所以一般是自己编写程序自己进行模块测试。

集成测试包括子系统测试和系统测试,它是将软件组装起来进行测试的技术。集成测试主要是在把模块按照软件设计的要求组装起来的同时进行测试,其主要目的是发现模块之间的相互影响以及各个模块接口之间可能存在的有关问题。

5. 软件维护(第5章)

软件维护是指在软件系统交付使用之后,软件使用人员为了适应新的要求、满足新的需要或改正软件中存在的错误而对软件系统进行修改的过程。软件系统维护是整个软件生命周期中持续时间最长、代价最大的最后阶段。

软件系统维护通常包括:为了改进原来的软件系统而进行的完善性维护;为了适应新的外部环境而进行的适应性维护;为了改正在实际使用过程中暴露出来的错误而进行的纠错性维护;为了改进软件系统将来的可维护性和可靠性而进行的预见性维护。其中,完善性维护占整个维护工作量的一半以上。

0.4 软件工程课程设计的主要任务和评价标准

软件工程课程设计是软件工程及相关专业的实践教学必修课程,其任务是要求学生针对具体

的软件工程项目,完成软件的整个开发过程,从而提高学生软件工程的综合实践能力。

要有效完成软件工程课程设计的任务,必须做到如下几条:

1) 软件工程课程设计准备。

软件工程课程设计指导书要规范、完整,符合教学大纲要求;课程设计的场所、设备等条件能够充分满足软件工程课程设计教学需要。

2) 软件工程课程设计选题。

软件工程课程设计题目符合专业培养目标和软件工程课程教学要求,设计题目深度、广度、难度适当,符合教学大纲要求;紧密结合生产、科研、管理、社会工作等实际;能够提供多个设计题目供学生选择,充分调动学生的学习主动性和积极性。

3) 软件工程课程设计过程。

首先制定完整科学的工作计划,认真填写任务书,因材施教,鼓励创新,注重学生综合能力培养,保证绝大多数学生按进度要求独立完成全部工作量,并且学习态度积极主动;严格按照每个阶段的评价标准对软件工程课程设计过程进行全程监控评价,认真履行课程设计答辩(口试)程序,严格掌握成绩评定标准,客观、真实地反映学生的课程设计质量,成绩呈正态分布。

4) 软件工程课程设计效果。

大多数学生的设计说明书思路清晰、文字表达能力强、书写工整,图(表)整洁、规范,软件产品符合软件工程行业技术标准;大多数学生能够掌握和运用基本理论知识,设计、实践能力达到教学大纲的基本要求,部分学生在课程设计过程中表现出较好的创新意识和创新能力。

软件工程课程设计的总成绩可以为等级(一般分为优秀、良好、中等、及格、不及格5个等级),也可以为分数。评定成绩时,应依据各个阶段工作量的大小和难度、文档质量、动手实践能力、团结协作能力等因素进行综合考虑,给出合理的课程设计成绩。

各个阶段设计质量高,过程规范,有较好的独特见解,文档正确规范,逻辑结构清晰、语句流畅,可以评为优秀。

各个阶段设计符合质量要求,过程规范,有一定的独特见解,文档正确规范,逻辑结构清晰,可以评为良好。

各个阶段设计基本符合质量要求,过程规范,文档基本正确规范,逻辑结构清晰,可以评为中等。

各个阶段过程规范,设计符合要求,文档基本正确,可以评为及格。

开发阶段过程不规范,设计不符合要求,达不到及格标准,则评为不及格。

0.5 本教材的主要特色和使用方法

软件工程课程设计是一门实践软件开发与维护的普通原理和技术的工程设计和实验课程。与其他计算机专业基础课程相比,它的研究范围不仅涉及技术方法,还强调软件工具的灵活使用和团队人员的协同合作、项目的高效管理等多个方面。随着软件产业成为我国核心的、最具广阔就业前景的信息产业,软件工程课程设计课程的建设在计算机类相关专业的本科教学中占据了重要地位,特色鲜明的教材是学好这门课程的基础。

0.5.1 本教材的主要特色

本教材遵循软件企业的项目管理和软件开发模式,运用项目驱动、案例教学方法,总结了“期刊管理系统”、“图书管理系统”、“网上商城管理系统”、“饭卡管理系统”、“研究生培养管理系统”5个不同项目的开发过程和设计技巧。其主要特色如下:

1) 采用任务驱动的方式组织内容,将软件工程课程设计的内容融入任务教学之中,使学生带着真实的任务在探索中学习。在这个过程中,学生可以不断地获得成就感,更大地激发求知欲望,

逐步形成一个感知心智活动的良性循环,从而培养出独立探索、勇于开拓进取的自学能力。

2) 以软件工程的现代教学理论为指导,案例选择注重学生的熟悉程度,以激发学生的学习兴趣,调动学生学习的积极性,充分体现学生的主体作用。本书以5个实际应用项目为案例,强调“做中学”,具有很强的可操作性,易学易用。

3) 充分体现培养学生学习能力的宗旨。软件工程课程设计不是软件工程理论课程的教材,所以不是在每个案例中面面俱到,而是在每个案例中只使用一种描述方法,使学生集中精力学好并掌握该方法。又因为每个案例分别使用不同的方法,所以本书涵盖了最基本的内容,这就可以使学生在掌握基本方法的基础上,自主探索,举一反三,触类旁通,从而训练学生的科学思维方式和创新性软件开发的能力。

4) 注重让学生掌握完整的软件工程知识体系。每个案例自成体系,并且具有突出的分析、设计、开发、维护特色,通过一个案例的实践,学生对软件开发过程就会有一个较全面的理解,避免学生产生对软件工程各种技术的片面认识。本教材介绍的案例,目的是为了给学生提供软件开发技术的完整知识结构体系,培养学生的科学精神和工程设计能力。

5) 适用性强。本教材既努力反映现代软件工程技术的新成就,将典型的软件分析、设计、编码、测试等技术引入到教材中,又考虑到每个案例的独立特色,并通过多年的教学实践不断更新和完善它们,注重理论联系实际,便于读者理解、掌握与应用所学知识。本教材不仅适合大学生学习,也适合软件开发人员学习参考。

6) 案例自成体系,开发方法各具特色。垂直(自上而下)方向的每个案例自成体系,水平(横向)方向的描述方法在不失流行的前提下,略有差异,方便学生根据给定案例进行各种组合练习,这不仅大大提高了案例的利用率,而且扩大了训练范围,使学生有更多的选择余地,受到更好的锻炼。

0.5.2 本教材的使用方法

本教材与软件工程课程的理论教学同步,吸收了大多数软件工程教材的精华,采用了当前最流行的分析、设计、编码、测试和维护技术。本教材的5个案例分别使用不同的风格,所以能使學生熟悉目前的主要流派,开拓知识的深度和广度。

软件工程课程设计采取项目驱动的方法,按照软件企业项目管理和开发模式,学生分成3~5人一个小组,进行软件系统的合作开发。在整个课程设计的实践过程中,指导教师既充当客户,又充当项目经理,不仅协调团队内部之间的关系,掌握项目开发进度,还担当技术指导,了解学生灵活运用知识的能力。这个过程对学生的创新精神、实践能力和团队协作精神的培养具有显著的效果。

本教材由5个案例组成,共分6章,第0章是每个读者都应该阅读的,对于后面的5章,每个课程设计小组只需要选择1个自己感兴趣的案例进行学习,仿照所选案例的要求进行分析、设计、编码、测试和维护。从第1章到第5章,每章对应软件开发的一个阶段,每章的开始概述该开发阶段的主要任务,最后给出该开发阶段的评价标准。

为了增加学生的软件开发经验,体会不同的开发风格,在课程设计过程中,学生也可以参考其他案例的设计技巧来设计所选案例,从而锻炼自己分析问题和解决问题的能力。



第 1 章

软件系统分析

1.1 概述

软件系统分析（简称软件分析）是软件开发的第一个阶段，是对所要完成问题的定义，其目的是弄清用户需要计算机解决什么问题，写出规范的需求分析文档。软件分析包含两个过程：可行性研究和需求分析。

1.1.1 可行性研究的任务和步骤

一个工程和一个项目是不能盲目开工的，在确定是否开工以前，首先要开展可行性研究。可行性研究是为了弄清所定义的项目是否是可能实现的、是否具有开发价值。下面介绍如何开展可行性研究。

1. 可行性研究的内容

可行性研究的内容如下：

- 1) 经济可行性：研究有无经济效益，多长时间可以收回成本。
- 2) 技术可行性：研究现有的技术是否可行，有哪些技术难点，采用的技术的先进程度。
- 3) 运行可行性：研究为新系统规定的运行模式是否可行。
- 4) 法律可行性：研究新的系统开发会不会引起侵权。

2. 成本效益分析

成本效益分析是估计新系统所需成本和可能产生的效益。系统成本包括开发和运行维护成本、物质消耗、占用操作员和维护人员的数量、培训费用。成本效益分析还要估计系统效益，包括经济效益和社会效益。

3. 可行性论证报告

在可行性研究的后期要形成可行性论证报告，它是可行性研究的标志性成果。可行性论证报告的组成如下：

(1) 系统概述

当前系统及其存在问题描述，新的目标系统及其各个子系统的功能与性能。

(2) 可行性分析

它是报告的主体，新系统在经济上、技术上、法律上的可行性以及对建立新系统的主客观条件分析，如果有多种方案，应该对几种方案进行比较，并指明推荐的方案。

(3) 结论意见

综合上述分析，说明新系统是否可行：1) 可立即执行；2) 推迟执行；3) 不可行或不值得。

4. 项目实施计划

(1) 系统概述

包括项目目标、主要功能、系统特点以及关于开发工作的安排。

(2) 系统资源

包括开发和运行该软件系统所需要的各种资源，如硬件、软件、人员和组织机构。

(3) 费用预算

分阶段的人员费用、机时费用及其他费用。

(4) 进度安排

各阶段的起止时间、需要完成的文档、验证方式及产品清单。

1.1.2 软件系统分析的任务

要完全弄清用户对软件系统的确切要求，需要用需求规格说明书描述。软件系统分析的具体任务是：

- 1) 确定对系统的综合要求。
- 系统功能要求。

• 系统性能要求：可靠性、安全性、响应时间（查询、更新）。

• 系统运行要求：环境（硬件、软件、数据库、网络、通信）。

• 将来可能的要求：与其他系统的连接。

2) 画出系统的逻辑模型。

用数据流图、数据字典和加工（或处理）描述。

3) 修正系统的开发计划。

适当修正计划时期的开发计划中的成本和进度。

对需求规格说明书的要求如下：

- 1) 准确性和一致性。不能含混不清，前后矛盾。
- 2) 无二义性。
- 3) 直观、易读，容易修改。采用简单符号、表格和标准图形表示。

常用的软件分析方法主要包括传统的分析方法和面向对象的分析方法。传统的分析方法的典型代表是结构化分析方法。结构化分析方法是20世纪70年代中期由Yourdon等人倡导的一种面向数据流分析方法。结构化分析就是使用数据流图（Data Flow Diagram, DFD）、数据字典（Data Dictionary, DD）、结构化语言、判定表和判定树等工具，建立一种称为结构化说明书的文档。

1. 数据流图

数据流图是软件系统逻辑模型的一种图形表示。结构化分析方法认为：从根本上来说，任何软件系统都是对数据进行加工或变换。在数据流图中，加工有两种常用的图形表示形式：圆框和圆角方框，如图1-1所示。数据流图中箭头代表数据流动方向，数据流的名称标在箭头边上，图1-1的含义是输入数据流经过加工后转变成输出数据流。



图1-1 数据流与加工的示例

(1) 组成符号

数据流图使用以下四种基本图形符号：

- 圆框或圆角方框代表加工。
- 箭头代表数据的流向，数据名称总是标在箭头边上。
- 直角方框表示数据的源点或终点，如图1-2a所示。
- 单线、双线或带缺口矩形框表示数据存储（数据文件、数据库），如图1-2b所示。



a) 数据流的源点或终点

b) 数据存储的几种表示符号

图1-2 数据流图的其他图形符号表示

在数据流图中，除了标示加工的名字，还要在加工框中加上编号，以便数据流图识别和分层。