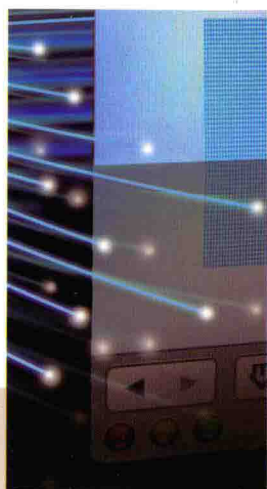


21世纪高等职业教育 计算机系列规划教材

软件项目开发 与管理案例教程

◆ 牛德雄 龙立功 主编
◆ 扶卿妮 熊君丽 杨叶芬 副主编



SOFTWARE



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

21 世纪高等职业教育计算机系列规划教材

软件项目开发与管理案例教程

牛德雄 龙立功 主 编

扶卿妮 熊君丽 杨叶芬 副主编

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书以项目为载体,以任务驱动的方式介绍软件开发中各阶段所需的知识、技术、方法、工具、开发过程,直至项目完成。

本书围绕软件开发能力的培养组织内容,全书共8章,内容如下:第1章,介绍软件、软件开发、软件项目管理等相关概念,重点介绍软件开发中容易混淆的一些概念并为后续学习做知识准备;第2和第3章,介绍传统的软件开发方法,适合软件开发初学者学习;第4和第5章,介绍面向对象的开发方法,利于读者掌握目前流行的面向对象软件开发方法与工具;第6章,介绍软件的实现,包括编码与测试,只有通过测试的软件,其编码才能告一段落;第7章,介绍软件的维护;第8章,介绍本书软件项目完整的开发过程,可作为软件设计文档范本。

本书可作为高职院校项目管理课程的教材,同时也适于软件项目管理人员、软件开发人员阅读,或作为项目管理人员的培训教材使用。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

软件项目开发与管理案例教程 / 牛德雄, 龙立功主编. —北京: 电子工业出版社, 2014.2

(21世纪高等职业教育计算机系列规划教材)

ISBN 978-7-121-22364-8

I. ①软… II. ①牛… ②龙… III. ①软件开发—项目管理—高等职业教育—教材 IV. ①TP311.52

中国版本图书馆CIP数据核字(2014)第010130号

策划编辑: 徐建军 (xujj@phei.com.cn)

责任编辑: 徐建军 特约编辑: 俞凌娣

印 刷: 北京市李史山胶印厂

装 订: 北京市李史山胶印厂

出版发行: 电子工业出版社

北京市海淀区万寿路173信箱 邮编 100036

开 本: 787×1092 1/16 印张: 11.5 字数: 294.4千字

印 次: 2014年2月第1次印刷

印 数: 3000册 定价: 28.00元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010)88258888。

前言

Preface

教育部十六号文件《关于全面提高高等职业教育教学质量的若干意见》明确指出要“加强素质教育，增强学生的职业能力，加大课程建设与改革的力度”。本书尝试摒弃传统软件工程的学科性教学，围绕软件开发需要的职业能力培养，组织与设计教学内容。

传统的软件教学重点在知识与技术教学，其不足主要表现在：概念与理论知识多、力求知识理论体系的完善；软件开发技术、方法的教学针对性不强；每个软件的开发过程中，各阶段间的过渡技术显得脱节；教学内容对程序编码的指导作用不明显等。所以，学生难以理解与领悟其中的内容，特别在实际操作上难以做到学以致用。另外，软件开发与管理模式灵活多样，知识体系庞大，要完整掌握难度大。

针对上述问题，笔者在进行教学内容设计时，分析高职学生的特点，以软件开发专业人才应知应会的知识、能力作为培养目标，并围绕这些目标进行教学内容的设计。本书教学内容的设计具有以下特点：

- 内容的取舍以实用为原则，为职业能力培养目标服务。
- 以软件开发过程典型工作任务为重点，力争在技术上连贯。
- 教学情境的设计深入浅出，以任务驱动的形式组织教学，使学生容易消化。
- 加强各阶段教学内容在技术层面的衔接，突出了软件设计到编码技术上的过渡。
- 内容组织中，体现了“项目导向、任务驱动”的教学。突出展现了软件开发引导案例的技术连贯及其实现，再通过它突出体现软件开发的重要理论知识。
- 整个教学内容以一个项目（物流系统）为载体，将软件开发的理论、方法、工具、开发过程等融为一体。最后一章还提供这些案例的完整文档，让读者（特别是软件专业的学生）能掌握软件开发报告的编写。

软件开发过程主要有需求分析、软件设计、编码与测试、软件维护等活动，它们构成了软件开发过程的主要任务阶段。但是这些活动之间的组织不是简单线性的，它们之间的组织关系非常灵活，有多种经典模式。这就是为什么这些内容在实际开发中难以被掌握的原因。

软件开发各重要阶段的教学，就是本课程的教学重点内容，其实它们又是教学难点内容。这些教学内容及它们在技术层面的过渡往往被以前的教材忽略，或连贯性不强。本书以项目为导向，通过任务驱动组织这些内容，从而解决了上述重点与难点的教学内容及衔接问题。

另外，本书的组织利于软件开发人员的学习。比如第1~3章，目的是培养学生了解软件开发过程，让初学者知道软件是如何开发的，使学生掌握软件编码前的一些概念，并掌握传统需求分析、软件设计的过程与基本方法。而后续第4、5章是面向对象的方法，学生在已学习

需求分析和软件设计的基础上,采用流行的面向对象的方法进行需求、设计与建模,并体现了面向对象的设计向面向对象编码技术上的过渡。第6章介绍了软件的编码与实现,在软件的编码过程中,测试非常重要,只有通过测试才能得到实用软件。第7章介绍了软件使用中的维护。第8章则通过一个软件开发案例,综合应用前面介绍的面向对象软件开发方法,介绍软件开发的过程及文档的编写。

本书条理清晰、内容实用、技术连贯;书中内容的表述力争做到深入浅出,使易混淆的概念、方法等容易被理解与掌握。本书内容的过渡在技术上连贯,体现了“项目导向、任务驱动”的教学,利于培养学生的软件开发能力。本书内容有助于软件开发其他课程的学习与借鉴。

本书由广东科学技术职业学院的牛德雄、龙立功担任主编,其中,第1章、第4章、第5章由牛德雄、龙立功编写;第2章、第3章由杨叶芬编写;第6章由熊君丽编写;第7章由扶卿妮编写;第8章由牛德雄、施茂航编写。另外,魏云柯设计了本书所用到的图,移动中心教学企业顶峰公司参与了教材内容的设计。在此一并表示感谢。

为了方便教师教学,本书配有电子教学课件及相关资源,请有此需要的教师登录<http://61.145.231.44:8080/skills/solver/classView.do?classKey=5560098>教材网站,在“课程资源”的教材资源、教学课件、教学案例处下载;或到华信教育资源网(www.hxedu.com.cn)免费注册后下载,如有问题,可在网站留言板留言或与电子工业出版社联系(E-mail: hxedu@phei.com.cn)。

目前,国内外关于软件工程、软件开发与管理方面的资料非常多,新理论、新技术层出不穷。如何更好地取舍与组织适合高职学生教学的内容需要不断探索。由于时间仓促,书中难免存在疏漏和不足,恳请同行专家和读者能给予批评和指正。

编者

目录

Contents

第 1 章 软件开发与管理概述	(1)
1.1 软件与软件开发概述.....	(1)
1.1.1 程序、软件与系统.....	(1)
1.1.2 软件开发的特征.....	(3)
1.2 软件开发.....	(5)
1.2.1 软件开发过程.....	(5)
1.2.2 软件开发方法学和软件工程.....	(9)
1.3 软件项目管理内容.....	(10)
1.3.1 软件项目管理.....	(11)
1.3.2 软件质量管理.....	(13)
1.3.3 软件过程质量保证体系.....	(15)
小结.....	(17)
习题.....	(18)
实训：制定某项目进度计划.....	(19)
第 2 章 进行需求分析了解用户需求	(20)
2.1 软件需求概述.....	(20)
2.1.1 需求分析任务.....	(21)
2.1.2 需求分析过程.....	(21)
2.1.3 需求管理.....	(23)
2.1.4 软件需求分析从问题定义开始.....	(24)
2.2 需求分析方法与分析模型.....	(25)
2.2.1 需求分析与建模方法.....	(25)
2.2.2 需求分析模型.....	(25)
2.3 用传统方法建立系统数据模型.....	(26)
2.3.1 数据模型 (E-R 图) 的建立.....	(26)
2.3.2 E-R 图简述.....	(29)
2.3.3 数据字典.....	(30)

2.4	用传统方法建立系统功能模型	(30)
2.4.1	系统功能模型(数据流图)的建立	(31)
2.4.2	数据流图简述	(31)
2.4.3	功能模型简述	(32)
2.4.4	功能模型的文字描述	(33)
2.5	建立系统动态模型	(33)
2.6	需求说明书及书写重点	(35)
	小结	(36)
	习题	(36)
	实训:对“学生管理系统”进行需求分析与建模	(37)
第3章	基于需求分析进行软件设计	(40)
3.1	软件设计	(40)
3.1.1	软件设计概述	(40)
3.1.2	软件设计与编码	(42)
3.1.3	软件设计的内容	(43)
3.2	项目设计案例	(44)
3.2.1	从E-R模型进行数据库设计	(44)
3.2.2	概要设计(软件结构设计)	(45)
3.3	软件模块的详细设计	(48)
3.3.1	详细设计的任务	(49)
3.3.2	详细设计的工具	(49)
3.4	软件设计原则	(54)
3.4.1	模块与模块独立性	(54)
3.4.2	模块的耦合性	(55)
3.4.3	模块的内聚性	(55)
3.4.4	模块的规模、大小适中原则	(56)
3.4.5	信息隐藏和局部化的原则	(57)
3.4.6	抽象性的原则	(57)
3.5	软件设计说明书	(58)
3.5.1	软件概要设计说明书大纲	(58)
3.5.2	软件详细设计说明书大纲	(58)
	小结	(59)
	习题	(59)
	实训一:对“学生管理系统”进行软件概要设计	(60)
	实训二:对“学生管理系统”进行详细设计	(61)
第4章	用对象的观点与方法进行分析建模	(64)
4.1	概述	(64)
4.2	从业务描述出发建立系统的用例模型	(65)
4.2.1	建立系统用例模型	(66)
4.2.2	确定系统的用例	(67)

4.2.3	用例之间的关系	(68)
4.2.4	描述用例	(69)
4.3	从用例的交互中识别实体建立对象模型	(71)
4.3.1	识别类与对象	(71)
4.3.2	确定类的属性	(73)
4.3.3	确定类之间的联系	(73)
4.4	类与类之间的“关系”	(74)
4.4.1	关联关系	(74)
4.4.2	泛化关系	(75)
4.4.3	依赖和细化关系	(76)
4.4.4	确定关联	(77)
4.5	从用例的交互中识别交互细节建立动态模型	(77)
4.5.1	建立顺序图	(78)
4.5.2	通过顺序图描述实体间的动态行为	(79)
4.5.3	通过顺序图等动态模型分析与确定类的行为	(79)
小结		(81)
习题		(81)
实训：用面向对象的方法进行需求分析与建模		(81)
第 5 章	面向对象的观点和方法设计软件	(84)
5.1	面向对象软件设计概述	(84)
5.1.1	包图	(85)
5.1.2	数据库设计	(85)
5.1.3	软件架构和类两个层面的设计	(85)
5.2	高层软件体系结构设计	(86)
5.2.1	软件体系结构设计概况	(86)
5.2.2	软件体系结构设计内容	(86)
5.3	底层类的设计	(87)
5.3.1	系统各部件类的设计及其层次	(87)
5.3.2	业务领域类的设计	(88)
5.3.3	类设计	(89)
5.3.4	使用 MVC 设计模式	(90)
5.4	面向对象软件开发过程及最佳做法	(91)
小结		(93)
习题		(93)
实训：用面向对象的方法进行软件设计		(93)
第 6 章	按照软件设计进行编程实现并测试	(95)
6.1	软件编码概述	(95)
6.1.1	从软件设计过渡到软件编码	(95)
6.1.2	程序设计方法	(96)
6.1.3	常见计算机程序设计语言	(97)

6.1.4	常用软件开发工具	(100)
6.1.5	选择计算机程序设计语言并建立软件开发环境	(102)
6.1.6	制定程序编码规范	(102)
6.1.7	制定编程的过程标准	(103)
6.2	程序编码过程	(104)
6.2.1	迭代、增量的开发过程	(104)
6.2.2	选择某种 MVC 开发模式的工具编码	(104)
6.2.3	通过迭代开发直至满足用户需求	(105)
6.3	按照用户需求和软件设计测试软件	(106)
6.3.1	软件测试概述	(106)
6.3.2	软件测试阶段	(108)
6.3.3	软件测试用例设计案例	(110)
	小结	(116)
	习题	(116)
	实训:对“学生管理系统”进行实现并测试	(117)
第7章	通过软件维护不断满足用户的需求	(118)
7.1	软件维护概述	(118)
7.1.1	软件维护原因	(118)
7.1.2	软件维护类型	(119)
7.1.3	软件维护中的常见问题	(120)
7.1.4	软件维护策略	(120)
7.2	软件维护的过程	(121)
7.2.1	维护组织	(122)
7.2.2	维护报告	(122)
7.2.3	维护过程	(122)
7.2.4	维护记录	(123)
7.2.5	维护评价	(124)
7.3	提高软件的可维护性	(124)
7.3.1	软件的可维护性	(124)
7.3.2	软件可维护性的量化	(124)
7.3.3	提高可维护性的方法	(125)
7.3.4	软件维护的副作用	(126)
7.4	软件维护相关文档	(126)
7.4.1	软件问题报告	(127)
7.4.2	软件维护申请报告	(127)
7.4.3	软件维护报告	(129)
7.4.4	软件用户意见反馈及满意度调查	(129)
7.5	实施软件维护	(130)
7.5.1	维护机构	(130)
7.5.2	维护流程	(131)

7.5.3 维护评价	(131)
7.5.4 维护成本	(132)
小结	(132)
习题	(132)
实训: 设计一套结构化维护过程及文档格式	(133)
第 8 章 物流系统开发分析与设计案例	(134)
8.1 引言	(134)
8.2 项目概述	(135)
8.3 需求分析	(135)
8.3.1 业务描述	(135)
8.3.2 用例建模	(135)
8.3.3 用例交互实体建模——对象模型(实体类图)	(136)
8.3.4 功能分析	(137)
8.3.5 交互细节的建模——动态模型	(138)
8.4 软件设计	(138)
8.4.1 功能模块设计	(138)
8.4.2 软件架构设计(基于 JavaEE)	(139)
8.4.3 实体类的设计	(141)
8.4.4 数据库设计	(141)
8.5 软件模块设计与实现	(144)
8.5.1 送货员信息管理模块	(144)
8.5.2 接单管理模块	(146)
8.5.3 接送货管理模块	(147)
8.5.4 仓储与分拣管理模块	(148)
8.5.5 结算管理模块	(152)
小结	(153)
附录 A 计算机软件开发文件编制指南(GB/T8567—1988)	(154)
附录 B UML 简介	(171)
术语表	(173)
参考文献	(174)

第1章

软件开发与管理概述

学习目标

[知识目标]

- 理解程序、软件、系统3个层次概念及其区别。
- 理解程序设计、软件开发、系统应用3个层次的活动。
- 了解软件开发过程主要阶段的概念与任务，包括需求分析、软件设计、编码实现、软件测试和软件维护等阶段。
- 了解几种常见的软件开发模型。
- 了解软件的结构化开发方法、面向对象开发方法。
- 了解软件开发辅助工具的相关概念。
- 了解软件项目管理内容和制作项目进度计划。
- 了解软件质量、质量管理及质量保证体系相关内容。

[能力目标]

- 能将软件项目按软件开发过程分解任务，并进行任务管理安排。
- 能用 Microsoft Project 制作项目开发进度计划，并打印出进度表。

1.1 软件与软件开发概述

1.1.1 程序、软件与系统

刚开始学习软件时，一般先是学“程序”的设计。但程序与软件是一回事吗？

程序是计算机执行代码组成的指令集。读者在刚开始学习程序设计时，往往是先学程序设计基础知识，了解程序设计思想及逻辑算法的概念。但随着学习的深入，逐步到了具有一定“实用”价值的软件开发阶段，这时需要进一步学习软件开发的理论、过程规范等。

从学习阶段来讲，程序设计是软件设计学习的初期阶段；从软件组成来讲，程序设计是软件开发的部分。软件中的各个程序，是整个软件系统的有机组成部分，在软件的运行过程中，它们之间可能需要进行复杂的交互。这也说明了软件的结构复杂，软件开发与管理难度大。

1. 软件相关概念

计算机软件又称“软件”，是相对于计算机“硬件”的概念，它是具有可用性的逻辑“物品”，也是一种“产品”。计算机软件是具有使用价值的计算机（软件）产品的一个完整配置。

从产品构成上来说，软件由下面 3 个部分组成：

- (1) 程序代码。
- (2) 软件文档。
- (3) 数据结构和数据。

这 3 个部分的组成，构成了完整的软件，程序代码只是软件的部分组成。而程序代码又是由各个部分的程序代码组成的一个整体，人们很容易将这个程序代码与软件概念等同。程序代码和软件的关系如图 1.1 所示。

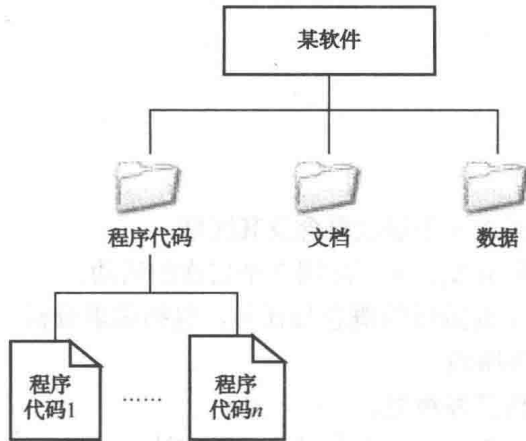


图 1.1 软件与程序代码的关系

这些程序代码整体，再加上软件文档、数据结构与数据等构成了软件的配置，也就是该软件。软件的开发，需要基于这些配置才能有效顺利地进行。软件开发的理论与方法是人们经历了长期的实践与探索才形成的一个较完整的体系。即从早期的程序设计的五花八门，到现在的软件工程思想、技术方法，再到软件开发标准规范，逐步形成了现代的软件开发与管理理论。

系统是指为某个目标而有机地结合的一个较完整的整体，如硬件系统、软件系统、应用软件系统、财务软件系统等。系统具有“生命”与“边界”；系统的结构是由各相互合作的子系统组成的。系统的运行需要硬件条件与软件条件，是在人的操作下进行的。软件系统是计算机系统的一个必要组成部分。

程序、软件、系统（软件）是既有区别又有联系的概念，有着各自不同的范畴与知识领域，且理解角度不同。在学习软件开发时，要立足软件的范畴与体系，围绕“软件产品”的结构、功能、设计与实现过程等各方面知识与技能。

2. 软件的特点

软件相对于硬件来说，具有以下特点：

- (1) 软件是逻辑的，而不是物理的产品。因为逻辑实际只存在于人的头脑当中，所以软件

的开发过程极难控制。

(2) 软件是由人开发形成的，没有明显的制造过程。软件成本集中于“开”上，因而软件项目不能完全像硬件制造项目那样来管理。

(3) 软件由人的“开发”劳动而成，到目前为止，软件开发尚未完全摆脱人的手工方式。所以，个人因素在软件开发过程中所占的重要性比重很大。

(4) 软件成本相当高，具体包括开发费用和维护费用。

(5) 软件本身是复杂的，维护困难且维护成本高。

软件的特点决定软件的开发与管理复杂。

3. 程序设计、软件开发与系统应用与支持

软件设计开发的学习是从程序设计开始的。这时程序设计的规模较小，实现的目标少、实用性不高。随着软件开发知识的深入，进入软件设计与开发领域知识与技能的学习。

程序设计是程序级代码的设计与实现，主要体现在程序处理过程中逻辑的设计与实现上，其相当于软件设计中的“详细设计”。而软件设计不是简单的程序设计的叠加，各个程序之间是有机联系的。由于软件设计复杂，规模大，一般软件设计需要一个宏观蓝图的规划，即所谓的“总体设计”。通过总体设计将复杂的大问题分解成小问题，直到软件中程序代码的设计（即详细的设计）。软件开发还包括需求分析、编码实现与软件测试等阶段。

所以，程序设计是在软件的“整体”结构下的各有机组成部分的程序代码的设计。

例如：一个财务管理软件，它包括该软件的程序代码、使用文档、数据与数据库等；而程序则是该软件中以可执行的程序文件、源程序文件等；而安装使用后，便与计算机硬件、网络等构成一个财务管理系统，而运行的财务软件部分则称为财务系统软件。

软件开发的最终产品是可供应用的软件系统。软件系统能为用户提供一定的应用服务。系统运行与应用对应的工作岗位有：软件系统实施、用户培训、系统操作、系统维护、系统技术支持等。

表 1.1 为软件开发各阶段特征及工作任务类型。

表 1.1 程序设计、软件开发和系统支持的区别

类 型	阶段特征及工作任务类型
程序设计	程序过程级的算法分析与设计、程序设计与实现
软件开发	包括软件的总体结构与程序过程级。任务有需求分析、软件设计、数据库设计、编码实现、软件测试等
系统支持	包括计算机硬件系统与软件系统的运行与支持。对应的任务有以系统的观点进行的系统认知、系统设计、系统开发、系统应用

1.1.2 软件开发的特征

1. 软件的设计与开发的复杂性特征

如前所述，软件本身是个复杂体，软件的设计与开发也是个复杂的过程。

软件设计与开发的复杂性特征：

(1) 软件要满足用户的使用需求，软件的设计要从用户需求开始。

(2) 软件设计开发常常是多人组成的团队进行的复杂工作，需要团队成员之间进行交流与合作。

(3) 软件开发“文档”在项目的开发与管理中是一个基础性内容，文档是团队成员交流与

合作的基础，也是管理与规范的基础。

(4) 软件的开发需要通过项目的管理与控制，使各方面工作秩序化。

(5) 某个软件开发团队的模式相对稳定，是由该团队的成员特点共同决定的，但团队的管理水平和能力有不同的层次，需要不断地改进。

(6) 软件的质量是软件设计与开发的生命。

2. 软件文档及其作用

软件开发知识：软件开发文档

在软件开发中，团队成员之间进行技术交流、分析与设计的表达、管理计划与过程控制、技术资料归档等，均需要软件文档。软件文档在软件开发过程中起着非常重要的作用。初学者开始时感觉不到软件文档的重要性，觉得它是可有可无的。但随着软件开发的深入，文档的重要性就会逐步体现出来。

软件文档包括开发文档、产品文档、管理文档3类。

(1) 开发文档是描述开发过程本身的文档，如需求分析文档、软件设计文档、软件测试文档等。

(2) 产品文档是描述开发过程的产物，如培训手册、用户指南、产品手册、产品宣传册或广告等。

(3) 管理文档是记录项目管理的过程信息，如开发过程的每个阶段的进度和进度变更的记录、软件变更情况的记录、开发阶段评审记录、职责定义等。

软件文档具有以下作用：

(1) 软件文档是软件项目管理的依据。

(2) 软件文档是软件开发过程中各任务之间联系的凭证。

(3) 软件文档是软件质量的保证。

(4) 软件文档是用户手册、使用手册的参考。

(5) 软件文档是软件维护的重要支持。

(6) 软件文档是重要的历史档案。

软件开发所涉及的文档主要有以下几种。

(1) 可行性研究报告。

可行性研究报告说明该软件在技术上、应用上、经济上是否可行，即是否值得开发、是否开发得出、是否违背社会法律和人们的道德规范，如果可行则应该怎样开发等，供决策者参考。

(2) 项目开发计划。

项目开发计划是指为软件项目实施制定出具体的计划，该计划包括各部分工作任务的负责人员、开发进度、开发经费预算、所需要的硬件和软件资源等。

(3) 软件需求说明书。

软件需求说明书也称软件规格说明书，是对所开发软件的功能、性能、用户界面、运行环境等作出的详细说明。它是用户与开发人员双方对软件需求取得共同理解和协议，通过该文档确定下来并作为今后开发工作的基础。

(4) 数据要求说明书。

数据要求说明书对数据的逻辑结构及各数据项的描述，以及对数据采集、数据约束的各项要求的说明，为今后生成和维护数据库、数据文档做准备。

(5) 概要设计说明书。

概要设计说明书是软件概要设计阶段的工作成果,是软件总体设计内容,它说明功能分配、模块划分、程序总体结构、输入/输出以及接口设计、运行设计、数据结构设计、出错处理设计等,为详细设计奠定基础。

(6) 详细设计说明书。

详细设计说明书描述每一模块内部是如何实现的,包括实现算法、逻辑处理流程等。

(7) 用户手册。

用户手册详细描述使软件正常运行的步骤,以及软件的功能、性能、用户操作界面,使用户了解如何使用该软件。

(8) 操作手册。

操作手册为操作人员提供了该软件各种运行情况的有关知识,特别是操作方法的具体细节、注意事项等。

(9) 测试计划。

为做好各种测试,需要为如何组织测试制定实施计划。计划应包括测试的内容、进度、条件、人员、测试用例等,并且包括各测试用例的执行步骤及预期结果和运行的偏差范围等。

(10) 测试分析报告。

测试用例执行完后,通过编写测试计划执行情况的说明,对测试结果予以分析,并提出测试结论意见,形成测试分析报告。

(11) 项目开发总结报告。

软件项目开发完成之后,通过总结软件开发执行情况,如进度、成果、资源利用、成本和投入的人力等进行总结,为今后的开发提供借鉴。同时还对整个开发工作进行评价,总结出经验和教训,以不断提高团队的能力。

1.2 软件开发

软件开发是个复杂的工作任务,且其中包括多个领域的知识,主要有以下几个方面:软件开发过程、软件开发方法、软件开发工具、软件项目管理等。

1.2.1 软件开发过程

软件的开发不同于程序阶段的开发,它是个复杂的系统工程。虽然软件开发是逻辑产品的创造过程,没有一个明显制造过程的特点,但在开发工作过程中的主要工作任务是明确的,即包括:需求分析、软件设计、编码实现、测试、项目管理等。这些任务有着各自明确的内容,它们之间内容虽不同但却相互联系与衔接,形成复杂的工作流程,这个复杂的工作流程就是所谓的软件开发过程。软件开发过程一般不是一个固定的模式。不同的相互衔接模式,构成了软件开发的工作过程的不同模型。

软件开发过程从问题定义,再到需求分析、软件设计、编码实现、软件测试,直到交付使用,最简单的过程如图 1.2 所示。

软件开发过程中的各个任务阶段之间的关系,不是一种简单的流程,如有线性的瀑布式、有螺旋上升的迭代式等。

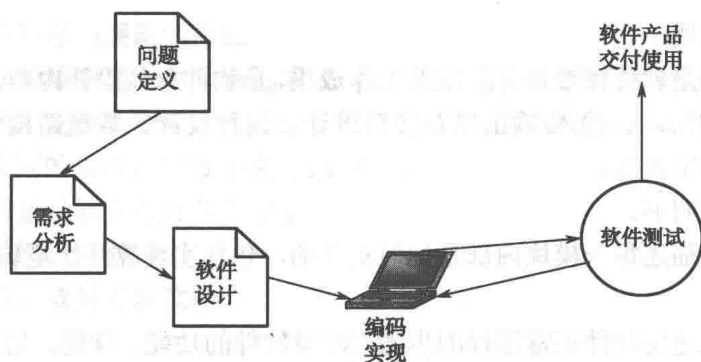


图 1.2 简单的软件开发过程示意

1. 软件开发过程的主要任务活动

软件开发知识：软件开发的工作任务

软件开发不只是程序编码工作，它还包括需求分析、软件设计、编码、软件测试、软件维护等工作任务。

软件开发过程一般包括的任务活动有：问题定义、需求分析、软件设计、编码实现、软件测试、软件维护等。

(1) 问题定义。

从高层次了解“用户要计算机和软件做什么”，确定软件的功能及边界。只有了解了计算机和软件要干什么，才能安排好下一步的工作。该工作一般由系统分析师根据调研现实情况，通过精确的文字陈述出来。

(2) 需求分析。

需求分析的任务是精确地描述软件系统必须“做什么”，确定系统具有哪些功能。该任务是由需求分析师通过分析得到软件的需求，并通过需求分析文档精确地表达出来。该文档是下一步软件设计的基础。

(3) 软件设计。

软件设计的任务是软件设计师将软件要做的功能，即上一步的软件需求转化为要做的内容与规划蓝图，以设计文档的形式表现出来。软件设计回答“怎么做”的问题，包括宏观与结构层面的设计，以及各程序内部处理过程的设计，即所谓的概要设计和详细设计。

(4) 编码实现。

编码实现是程序员将上一步的软件设计蓝图，通过某种程序语言一个一个地完成软件的程序代码，然后将它们集成起来形成一个可使用的完整软件。

(5) 软件测试。

软件测试是测试员将已经编写好的软件进行操作以发现问题。测试员测试的主要依据是软件需求、软件设计等。软件测试是软件质量保证的重要手段，测试要尽可能地发现软件中的问题，这是个非常复杂的工作，测试一般包括单元测试、集成测试、验收测试等。

(6) 软件维护。

软件维护是软件交付使用后，为了保证软件正常的使用，以及满足用户使用时的各项要求而进行的维护工作。一般来说，如果软件在没被废弃前，都需要进行软件维护。

其实，上述各阶段任务是对要开发的“软件”的一个层次的抽象与描述。每个阶段需要上一阶段作为输入，再加上本阶段任务的“工作”，便是后续阶段的输入。软件开发各个阶段的

任务，需要一直做到用户使用满意为止。

软件开发各阶段中的任务和要完成的工作内容如表 1.2 所示。

表 1.2 软件开发任务对应的角色、职责

任 务	职 责	开发人员角色
需求分析	了解、分析用户需求，编写需求分析说明书	需求分析师
软件设计	根据系统要求设计软件总体架构；根据功能需求分析说明书设计软件功能模块并细化，编写软件设计说明书	软件设计师
编码实现	根据软件设计说明书及编码规范编写代码，并进行单元测试	程序员
软件测试	根据软件需求、设计文档设计软件测试用例、制定测试计划，并对软件进行集成测试和系统测试，编写测试报告。配合程序员修改完善代码并进行回归测试	测试员
软件维护	负责维护用户使用过程中出现的各种问题	维护人员
项目管理	制定开发计划，组织团队并制定开发规则，领导各开发人员顺利工作并进行开发过程控制（监督、检查成员工作情况，控制质量等），成员工作评估及激励等	项目经理

2. 软件开发过程模型

软件开发知识：软件开发过程模型

不同的软件开发企业可能采取不同的开发过程方式，并形成了相对固定的开发模式，这些相对固定的开发模式称为软件开发过程模型。这些过程模型有：瀑布模型、快速原型模型、增量模型、螺旋模型、统一过程模型等。

软件开发各个阶段任务，是从产品的提出到开发完成并进行应用。但各工作阶段任务之间的连接不是完全相同的，有线性的、递增的、迭代的、螺旋上升等不同情况，即所谓的不同软件开发过程模型，如图 1.3~图 1.6 所示。



图 1.3 线性的瀑布模型

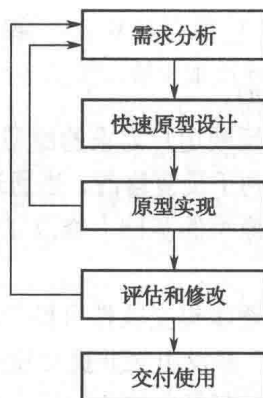


图 1.4 快速原型模型

(1) 瀑布模型。

瀑布模型是将软件开发过程的各个阶段任务规定为自上向下的线性过程，按照这种线性顺序连接的软件开发过程模型。该模型利于传统的结构化开发方法，但缺乏灵活性，用于解决由于需求的不确定性和不准确性带来的问题非常困难。