

PEARSON  
Education  
培生教育集团

HZ BOOKS  
华章教育

面向计算机科学与技术专业规范系列教材



# 软件需求 工程

毋国庆 梁正平 袁梦霆 李勇华 编著



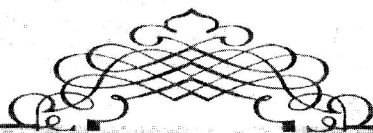
*Software Requirements Engineering*

机械工业出版社  
China Machine Press






面向计算机科学与技术专业规范系列教材



# 软件需求 工程

毋国庆 梁正平 袁梦霆 李勇华 编著

*Software Requirements Engineering*

 机械工业出版社  
China Machine Press

软件需求工程在软件开发中的作用是相当重要的。因此,对于从事和将要从事软件开发工作的人们来说,有必要了解和掌握软件需求工程中的一些内容。本书全面和系统地介绍了软件需求工程的基本概念和原理,以及开发和管理软件需求的方法和技术。此外,本书也介绍了软件需求工程中的一些新方法和技术。为便于读者学习和理解,本书在介绍软件需求工程内容时结合了许多的典型实例。

本书可作为本科生高年级和研究生的教材,也可供从事软件开发工作和研究的专业人员参考和自学。

版权所有,侵权必究。

本书法律顾问 北京市展达律师事务所

## 图书在版编目(CIP)数据

软件需求工程/毋国庆等编著. —北京:机械工业出版社,2008.8  
(面向计算机科学与技术专业规范系列教材)

ISBN 978-7-111-24809-5

I. 软… II. 毋… III. 软件开发—高等学校—教材 IV. TP311.52

中国版本图书馆 CIP 数据核字(2008)第 115307 号

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:王 玉

三河市明辉印装有限公司印刷·新华书店北京发行所发行

2008 年 8 月第 1 版第 1 次印刷

185mm×260mm·12.25 印张

标准书号:ISBN 978-7-111-24809-5

定价:25.00 元

凡购本书,如有倒页、脱页、缺页,由本社发行部调换  
本社购书热线:(010)68326294

# 出版者的话

机械工业出版社华章公司是国内重要的教育出版公司，培生教育集团(拥有 Addison Wesley、Prentice Hall 等品牌)是全球知名的教育出版集团，双方在过去长达十余年的合作中秉承“全球采集内容，服务教育事业”的理念，遴选、移译了国外大量的在计算机科学界享誉盛名的专家名著与名校教材，其中包括 Donald E. Knuth、Alfred V. Aho、Jeffrey D. Ullman、John E. Hopcroft、Dennis Ritchie 等大师名家的经典作品(收录在大理石封面的“计算机科学丛书”中)，这些作品对国内计算机教育及科研事业的发展起到了积极的促进作用。

随着国内计算机科学与技术专业学科建设的不断完善、教学研究的蓬勃发展，以及教材改革的逐渐深化，计算机科学与技术专业的优秀课程及教材不仅仅是“引进来”(版权引进)，而且需要“走出去”(版权输出)了。

近几年以来，教育部计算机科学与技术专业教学指导分委员会根据我国计算机专业教育的现状以及社会对人才的需求，发布了《高等院校计算机科学与技术专业发展战略研究报告暨专业规范(试行)》(以下简称《规范》)。为配合《规范》的实施推广，同时为落实中央“提高高等教育质量”的最新指导思想，在教育部计算机科学与技术专业教学指导分委员会的指导下，在国内知名高校众多教授的帮助下，我们出版了这套“面向计算机科学与技术专业规范系列教材”。

本套教材的作者在长达数十年的科研和教学经历中积累了大量的知识和经验，也奠定了他们在学术和教学领域的地位，教材的内容体现了他们的教学思想和教学理念，本套教材也是传承他们优秀教学成果的最好载体，是中国版的专家名著和名校教材，相信它们的出版对提高计算机科学与技术专业的教育水平和教学质量能够起到积极的作用。

华章与培生作为专业的出版团队，愿与高等院校的老师共同携手，在这套教材的出版上引进国际先进教材出版经验，在教学配套资源的建设上做出新的尝试，为促进中国计算机科学与技术专业教育事业的发展，为增进中国与世界文化的交流而努力。



华章教育



培生教育集团

# 面向计算机科学与技术本科专业规范系列教材

编委会

主任委员：蒋宗礼

副主任委员：王志英 钱乐秋

委员：（以姓氏拼音为序）

陈道蓄	陈 明	何炎祥	傅育熙
黄刘生	贾云得	姜守旭	马殿富
李仁发	李晓明	刘 辰	齐 勇
孙吉贵	孙茂松	吴功宜	吴 跃
谢长生	于 戈	张 钢	周兴社

秘书组：温莉芳 刘立卿 姚 蕾

本书责任编辑：刘 辰

# 序言

近 20 年里，计算机学科有了很大的发展，人们普遍认为，“计算机科学”这个名字已经难以涵盖该学科的内容，因此，改称其为计算学科（Computing Discipline）。在我国本科教育中，1996 年以前曾经有计算机软件专业和计算机及应用专业，之后被合并为计算机科学与技术专业。2004 年以来，教育部计算机科学与技术教学指导委员会根据我国计算机专业教育和计算学科的现状，为更好地满足社会对计算机专业人才需求，发布了《高等学校计算机科学与技术专业发展战略研究报告暨专业规范（试行）》（以下简称《规范》），提出在计算机科学与技术专业名称之下，构建计算机科学、计算机工程、软件工程和信息技术四大专业方向。《规范》中四大专业方向的分类，在于鼓励办学单位根据自己的情况设定不同的培养方案，以培养更具针对性和特色的计算机专业人才。

为配合《规范》的实施，落实中央“提高高等教育质量”的精神，我们规划了“面向计算机科学与技术本科专业规范系列教材”。本系列教材面向全新的计算学科，针对我国高等院校逐步向新的计算机科学与技术专业课程体系过渡的趋势编写，在知识选择、内容组织和教学方法等方面满足《规范》的要求，并与国际接轨。本套教材具有以下几个特点：

(1) 体现《规范》的基本思想，满足其课程要求。为使教材符合我国高等院校的教学实际，编委会根据《规范》的要求规划本套教材，广泛征集在国内知名高校中从事一线教学和科研工作、经验丰富的优秀教师承担编写任务。

(2) 围绕“提高教育质量”的宗旨开发教材。为了确保“精品”，本系列教材的出版不走盲目扩大的路子，每本教材的选题都将由编委会集体论证，并由一名编委担任责任编辑，最大程度地保证这套教材的编写水准和出版质量。

(3) 教材内容的组织科学、合理，体系得当。本套教材的编写注重研究学科的新发展和新成果，能够根据不同类型人才培养需求，合理地进行内容取舍、组织和叙述，还精心设计了配套的实验体系和练习体系。

(4) 教材风格鲜明。本套教材按 4 个专业方向统一规划，分批组织，陆续出版。教材的编写体现了现代教育理念，探讨先进的教学方法。

(5) 开展教材立体化建设。根据需要配合主教材的建设适时开发实验教材、教师参考书、学生参考书、电子参考资料等教辅资源，为教学实现多方位服务。

我们衷心希望本系列教材能够为我国高等院校计算机科学与技术等专业的教学作出贡献，欢迎广大读者广为选用。

“面向计算机科学与技术专业规范系列教材”编委会

# 前 言

随着计算机应用的不断发展和深入，人们需要收集和处理的的信息规模急剧增大，其中计算机软件在信息收集和处理中起着至关重要的作用。由于软件已成为信息基础设施并具备密集型特点，使得软件一方面日益成为人们生活中的一部分，如电子政府、电子商务和手机等，人们也日益依赖软件。另一方面，一些高尖端的技术领域，如航空航天、国防军事等领域，对软件的质量提出了很高的要求。然而，软件开发的开发状况和质量远未达到人们的期望和要求，例如，大部分软件产品不能在预期计划和预算经费内完成，软件的质量低下等。软件质量所导致的软件故障和失效，常常对人们的工作和生活带来诸多不便，甚至造成重大损失。虽然产生上述问题的原因有很多，但软件需求问题可以说是其中的一个最大原因。许多人经过研究发现，当软件开发项目失败时，软件需求问题通常正是核心问题。因此，在软件开发过程中，必须极早和有效地发现和解决与软件需求相关的问题。

在很长一段时间里，人们并没有充分认识到软件需求的作用，软件工程界也一直没有将需求工程作为一个独立的部分进行深入的分析和研究。直到 20 世纪 90 年代中期，随着软件系统开发中出现的诸多问题，人们才逐渐认识到软件需求在整个软件开发中的重要性。通过一系列关于软件需求的重要学术会议进行广泛深入的研究和讨论，才使得需求工程作为一门独立的子学科正式形成。需求工程是指应用工程化的方法、技术和规格来开发和管理软件的需求。需求工程的目标就是要获取高质量的软件需求。与软件工程中传统的需求分析概念相比，需求工程突出了工程化的原则，强调以系统化、条理化和可重用的方法和技术进行与软件需求相关的活动，从而有利于提高所有与软件需求相关的活动及其过程的可管理性，降低需求开发和管理的难度和成本。

本书共分为 12 章。第 1 章概要地介绍了需求工程的历史背景、基本原理和一些基本概念。第 2 章介绍了几种软件的开发过程模型、需求工程在软件工程和软件开发中的重要地位以及软件需求的开发和管理过程。第 3~9 章按需求工程中开发和管理过程的顺序介绍各个阶段的任务、步骤、方法和技术。其中，由于在需求工程中形式化理论的研究是相当重要的内容，故第 7 章重点介绍在需求分析和需求定义中使用的一些基本理论和一些形式化描述方法。第 10 章和第 11 章主要讨论和介绍需求工程中近几年所研究出的新理论和新方法，如面向问题域的需求建模方法和面向多视点的需求工程方法。尽管这些新理论和新方法尚未实用化，但它们促进了需求工程的研究和发展。最后，第 12 章概要地介绍了需求工程与软件管理的关系，以及如何根据需求安排开发进度和估算工作量的方法。

基于需求工程在软件开发中的重要地位，软件开发人员有必要学习和了解需求工程的有关知识。目前，国外有关需求工程的教材和专业书籍已有很多，我国也出版了一些需求工程的翻译书籍。这些都是为了提高人们对需求工程重要性的认识，以及更进一步推动有关需求工程的理论、方法和技术的研究和实践。本书的编者多年来一直从事需求工程方面的研究和教学实践，特别是本书的第一作者在武汉大学软件学院以多届计算机专业和非计算机专业的研究生为对象进行了有关需求工程课程的讲授。本书也是根据该课程的讲义和学生们的建议，以及一些相关的资料编写而成的。在编写过程中，本书也参考了国内出

版的一些教材和翻译书籍。在此，谨向这些书的作者们表示衷心的感谢。编者希望此书有助于从事软件开发的专业人士和计算机专业的学生们的工作和学习，并从中获得一些有益的知识。

本书适合作为计算机专业高年级本科生和研究生的教材，也可供具有一定实践经验的软件开发人员和计算机用户等参考和自学。不过，本书中部分内容是关于形式化理论和新方法方面的介绍，需要较好的理论基础和参考其他方面的资料，故建议在教学中应根据学生的具体情况有选择地讲授本书的内容，如可在讲授中跳过第7章和第10章等。此外，需求工程是一门实践性较强的课程，虽然本书没有给出习题和思考题，但讲授需求工程课程的老师可根据本书的内容设置一些习题和思考题，读者可在本书的基础上结合软件开发中的一些具体实例进行实验和实践。

在本书的写作过程中，除封面署名外，刘小丽、张帆和万黎等参与了部分工作，特别是机械工业出版社华章分社的有关人员给予了极大的支持和帮助。此外，本书也得到国家高技术研究计划(863计划)的资助，借此表示衷心的感谢。

由于需求工程诞生的时间相对较为短暂，还处于发展过程中，加之编者们的水平有限，错误在所难免，恳请多多指教。

作者  
2008年7月  
于武汉



# 教学建议

## 教学目的与要求

软件需求工程是近几年从软件工程学科新发展出来的子学科。软件需求工程主要研究如何开发和管理软件需求,包括技术方法、形式化理论、工具和管理等许多方面。本课程着重从实用角度讲述软件需求工程的基本原理、概念、技术和方法,特别是软件需求的获取过程和软件需求建模的方法等,同时,尽量注意课程内容的全面性和系统性。通过本课程的学习,力争使具有实际软件开发经验的学生得到进一步提高,特别是在理论方面。对于不具有实际软件开发经验的和其他专业的学生,可以从中了解软件的开发过程并认识软件需求对软件开发的重要性,通过理解、学习和掌握软件需求工程中的一些概念和方法,为将来从事软件开发和管理工作打下良好的基础。为了较好地学习和掌握本课程的内容,要求学生应具有一定的计算机软件方面的知识。对于计算机专业的学生,最好学过软件工程课程或具有一定的软件开发经验。此外,学完本课程的内容后,最好将其应用于描述某个实际软件系统,并生成有效的需求规格说明文档。

## 课程教学内容与学时分配

### 第 1 章 需求工程概述(2 学时)

本章大部分内容属于概述性质。可以通过若干个小的示例使学生理解软件需求工程的重要性和任务、软件需求的概念以及分类、需求规格说明书等概念。

### 第 2 章 软件工程与需求工程(3 学时)

本章介绍软件工程与需求工程的相互关系。首先介绍软件工程产生的原因,通过对比的方式介绍几种软件开发过程模型,分析各种模型的优缺点以及适用范围。然后,结合一个示例,介绍需求工程在软件工程中的地位和作用,以及需求工程所面临的困难。最后,介绍软件需求的开发管理过程,使学生掌握软件需求开发的几个基本步骤。

### 第 3 章 需求获取(3 学时)

准确把握需求获取的几个基本步骤,了解软件需求的层次、用户分类的原则、软件需求来源和需求决策者等知识。掌握收集需求信息的步骤、方式和技巧。掌握非功能需求的定义以及获取的方法。最后,通过实例讲授场景技术在需求获取中的运用。

### 第 4 章 需求分析(2 学时)

本章介绍需求分析的具体工作。首先结合示例介绍系统关联图的制作方法和好处,介绍影响需求可行性的几个风险以及评估风险的简单方法,然后介绍如何描述用户接口模型以及如何确定需求的优先级,最后介绍需求建模的意义以及数据字典的相关知识。

### 第 5 章 需求建模方法与技术(8 学时)

本章详细地介绍几种需求建模方法,是课程的重点。首先简要介绍软件模型的定义和分类,结合示例介绍 SA 方法(包含基本思想、描述方法和分析步骤)。然后介绍 OMT 方法。如果学生已经学过面向对象的设计方法,则可以跳过 5.4.1 节,直接介绍 OMT 的建模过程,并通过几个示例介绍 OMT 方法的图形描述工具。最后介绍基于图形的需求建模技术。本章的

重点是 SA 方法和 OMT 方法，建议教学时采用对比的方法，并配合适当的练习。

#### 第 6 章需求定义(2 学时)

掌握一个严格的需求规格说明书应该具有的特性。掌握需求规格说明书的结构、内容和编写要求，了解需求规格说明的描述语言。

#### 第 7 章需求的形式化描述(3 学时)

本章内容有较大难度，本科生可以酌情少讲或直接跳过。首先介绍形式化方法在需求描述中的作用和意义，然后介绍几种形式化规格说明方法，最后结合具体的示例分别介绍 Z 语言、LOTOS 语言和 B 方法。

#### 第 8 章需求验证(3 学时)

了解需求验证的目的和任务。掌握验证的内容和方法。掌握需求评审的分工、过程、内容和困难。了解需求测试的方法。了解编制用户使用手册草案和解释需求模型的意义。

#### 第 9 章需求管理(2 学时)

掌握需求管理的主要内容。准确掌握需求变更控制的方法和步骤。了解需求规格说明文档的版本控制和需求变更状态的跟踪。掌握需求跟踪技术。

#### 第 10 章面向问题域的需求分析方法(3 学时)

本章内容有较大难度，本科生可以酌情少讲或直接跳过。了解问题域的基本概念、问题域划分的方法和步骤、问题框架和问题框架类型等知识点。通过校园通示例掌握 PDOA 方法。了解问题框架实例间的关系及其组合。

#### 第 11 章面向多视点的需求工程(3 学时)

理解视点的概念。了解多视点和需求工程的关系。结合列车控制系统示例，使学生掌握多视点需求工程的需求分析全过程。

#### 第 12 章需求工程与软件开发管理(2 学时)

了解需求与估算之间的相互关系。了解需求对项目进度安排的影响。了解基于需求的软件规模估算和基于需求的工作量估算方法。

## 课程教学重点、难点及注意的问题

**课程教学重点：**(1) 软件需求工程的一些基本概念和基本原理

(2) 软件开发的基本过程

(3) 软件需求获取的过程、方法和技术

(4) 软件需求建模方法和技术

(5) 软件需求定义和验证的方法和技术

(6) 软件需求管理的方法和技术

**难点及注意的问题：**(1) 软件需求工程中各种方法和技术的实际应用

(2) 软件需求工程中某些抽象概念和形式化理论的理解

(3) 需根据教学对象调整授课内容和学时数

## 实验及实践性环节

在完成课内理论学时后，作为教学实践可将已学知识应用于某个具体软件系统的需求分析。

# 目 录

出版者的话	
序 言	
前 言	
教学建议	
<b>第 1 章 需求工程概述</b> .....	1
1.1 需求工程的重要性 .....	1
1.2 什么是软件需求 .....	2
1.3 软件需求的分类 .....	3
1.4 需求规格说明 .....	4
1.5 需求工程定义 .....	5
1.6 其他一些基本概念 .....	6
<b>第 2 章 软件工程与需求工程</b> .....	7
2.1 软件工程 .....	7
2.2 软件开发过程模型 .....	7
2.2.1 瀑布式模型 .....	8
2.2.2 快速原型模型 .....	9
2.2.3 渐进式模型 .....	10
2.2.4 螺旋式模型 .....	11
2.2.5 面向对象的开发模型 .....	12
2.3 需求工程在软件开发中的地位 .....	13
2.3.1 需求工程对软件开发的影响 .....	13
2.3.2 需求工程面临的困难 .....	14
2.4 软件需求的开发和管理过程 .....	15
<b>第 3 章 需求获取</b> .....	17
3.1 确定需求开发计划 .....	17
3.2 确定项目的目标和范围 .....	18
3.3 确定调查对象 .....	19
3.4 实地收集需求信息 .....	22
3.4.1 实地收集需求信息面临的困难 .....	22
3.4.2 实地调查的步骤 .....	23
3.4.3 实地收集需求信息的方式 .....	23
3.4.4 需求信息的分类 .....	24
3.5 确定非功能需求 .....	26
3.6 在收集需求信息中应注意的问题 .....	27
3.7 使用场景技术的需求获取 .....	28
3.7.1 场景的定义及构成 .....	28
3.7.2 场景的表示 .....	29
3.7.3 场景的种类 .....	29
3.7.4 使用用例的需求获取 .....	30
3.7.5 场景技术的特点 .....	31
<b>第 4 章 需求分析</b> .....	32
4.1 建立系统关联图 .....	32
4.2 分析需求的可行性 .....	33
4.3 构建用户接口原型 .....	34
4.4 确定需求的优先级别 .....	35
4.5 需求建模 .....	36
4.6 建立数据词典 .....	37
<b>第 5 章 需求建模方法与技术</b> .....	38
5.1 什么是模型 .....	38
5.2 软件工程中的模型 .....	39
5.3 结构化的需求建模方法 .....	40
5.3.1 SA 方法的基本思想 .....	41
5.3.2 SA 方法的描述手段 .....	41
5.3.3 实例说明 .....	49
5.3.4 SA 方法的分析步骤 .....	52
5.4 面向对象的需求建模方法 .....	54
5.4.1 面向对象方法中的一些基本概念 .....	54
5.4.2 面向对象的需求分析 .....	57

5.4.3	OMT 方法的图形描述 工具 .....	58	7.7.2	B 方法中的数学符号 .....	111
5.4.4	基于 OMT 方法的需求 建模步骤 .....	64	7.7.3	B 方法中的抽象机 .....	112
5.5	基于图形的需求建模技术 .....	78	7.7.4	B 规约的实例 .....	114
5.5.1	UML 概述 .....	78	<b>第 8 章</b>	<b>需求验证</b> .....	116
5.5.2	活动图 .....	79	8.1	需求验证的目的和任务 .....	116
5.5.3	协作图 .....	80	8.2	需求验证的内容和方法 .....	117
5.5.4	实体关联图 .....	81	8.3	需求评审 .....	117
<b>第 6 章</b>	<b>需求定义</b> .....	83	8.3.1	审查人员的确定和 分工 .....	118
6.1	需求规格说明的作用 .....	83	8.3.2	正式的审查过程 .....	119
6.2	需求规格说明的特性 .....	84	8.3.3	审查的内容 .....	120
6.3	需求规格说明的结构和内容 .....	85	8.3.4	需求评审面临的困难 .....	121
6.4	需求规格说明文档的编写 要求 .....	91	8.4	需求测试 .....	121
6.5	需求规格说明的描述语言 .....	93	8.5	编制用户使用手册草案 .....	122
<b>第 7 章</b>	<b>需求的形式化描述</b> .....	96	8.6	解释需求模型 .....	123
7.1	形式化规格说明及其方法 .....	96	<b>第 9 章</b>	<b>需求管理</b> .....	124
7.2	形式化规格说明与软件开发 .....	97	9.1	需求变更控制 .....	124
7.3	基于公理或推理规则的形式 化规格说明 .....	98	9.2	需求规格说明文档的版本 控制 .....	128
7.4	基于代数的形式化规格说明 .....	100	9.3	需求变更状态的跟踪 .....	128
7.5	形式描述语言 Z Notation .....	101	9.4	需求跟踪 .....	129
7.5.1	Z Notation 简介 .....	101	9.4.1	可跟踪信息分类 .....	130
7.5.2	Z Notation 的数学 符号 .....	101	9.4.2	需求跟踪技术 .....	130
7.5.3	Z Notation 中的图表 .....	102	<b>第 10 章</b>	<b>面向问题域的需求 分析方法</b> .....	133
7.5.4	Z Notation 规约的 实例 .....	103	10.1	问题域 .....	133
7.6	形式描述语言 LOTOS .....	106	10.2	问题域的划分 .....	135
7.6.1	LOTOS 简介 .....	106	10.3	问题框架 .....	135
7.6.2	LOTOS 的数据描述 语言 .....	106	10.4	问题框架的类型 .....	136
7.6.3	LOTOS 的进程 .....	107	10.5	PDOA 方法的分析步骤 .....	141
7.6.4	LOTOS 规约的实例 .....	109	10.5.1	问题及问题域的 界定与描述 .....	141
7.7	B 方法 .....	110	10.5.2	基于问题框架的 问题域划分 .....	145
7.7.1	B 方法简介 .....	110	10.6	问题框架实例间的关系及 其组合 .....	150

10.6.1 问题框架实例间的 关系 .....	150	11.3.3 视点的分析 .....	159
10.6.2 问题框架实例的 组合 .....	151	11.3.4 视点的集成 .....	162
<b>第 11 章 面向多视点的需求工程</b> ...	153	11.4 实例 .....	163
11.1 什么是视点 .....	153	<b>第 12 章 需求工程与软件开发管理</b> ...	170
11.2 多视点与需求工程 .....	154	12.1 需求与估算 .....	170
11.3 多视点需求工程的过程模型...	155	12.2 需求与项目进度安排 .....	171
11.3.1 视点的标识 .....	156	12.3 基于需求的软件规模估算 .....	172
11.3.2 视点的表示 .....	157	12.4 基于需求的工作量估算 .....	173
		<b>参考文献</b> .....	175



# 需求工程概述

## 1.1 需求工程的重要性

随着计算机应用的不断发展和深入，软件系统的日益大型化、复杂化，软件的开发成本越来越高，软件开发的風險也越来越大。Standish 集团公司的研究报告称：在美国，每年用于软件开发的费用在一千多亿美元以上，其中，大型公司开发一个软件项目的平均成本为 232.2 万美元，中等大小的公司为 133.1 万美元，小型公司则为 43.4 万美元。调查显示，31% 的项目在完成之前被取消，进一步研究的结果还表明：52.7% 的项目实际所花费的成本为预算成本的 189%<sup>[1]</sup>。根据该公司的另一项分析，项目失败或严重超支的 8 个最重要原因中有 5 个都与需求相关：即需求不完整、缺乏用户的参与、客户期望不实际、需求和需求规格说明的变更和提供许多不必要的功能<sup>[2]</sup>。

一些具体的案例令人触目惊心：伦敦股票交易项目 TAURUS，在花费了数百万英镑之后于 1993 年被取消（项目失败的总损失估计达到几亿英镑）。调查结果显示，许多问题源于未能协调那些不一致的需求<sup>[3]</sup>。Swanick 空中交通控制系统原计划在 1998 年完工，但直到 2001 年尚未交付使用，额外开支高达 1 亿英镑以上。经官方调查，发现其中的一个主要原因在于“缺乏健壮的需求规格说明导致无法继续进行系统实现”<sup>[4]</sup>。

与此同时，另外的一些调查和研究显示：一个与需求相关的错误发现和解决越迟，其修复的代价越昂贵。A. Davis 研究发现，在需求阶段检查和修复一个错误所需的费用只有编码阶段的 1/5 到 1/10，而在维护阶段做同样的工作所需付出的代价却是编码阶段的 20 倍<sup>[5]</sup>。这意味着在维护阶段修复一个错误的代价与需求阶段修复一个同样的错误的代价的比值可高达 200 : 1。

诸如此类的调查研究目前已有很多。虽然项目失败涉及的原因多种多样，但正如 R. Glass 所说，“项目需求无疑是在软件项目前期造成麻烦的一个最大原因。一个又一个的研究已经发现，当项目失败时，需求问题通常正是核心问题”<sup>[6]</sup>。因此，在软件开发过程中，必须极早、有效地发现和解决与需求相关的问题。

在很长一段时间里，人们并没有充分认识到软件需求的作用，软件工程界也一直没有将需求工程作为一个独立的部分进行深入地分析和研究。直到上世纪 90 年代中期，随着软件系统开发中出现的诸多问题，人们才逐渐认识到软件需求在整个软件开发中的重要性。通过一系列关于软件需求的重要学术会议进行广泛而深入的研究和讨论，由 IEEE 创办的专门研究软件需求的国际期刊《Requirement Engineering》的出版发行标志着需求工程作为一门独立的子学科正式形成。

## 1.2 什么是软件需求

需求这一单词在日常生活中经常使用。通常的需求是指人对于客观事务需要的表现，体现为愿望、意向和兴趣，因而成为行动的一种直接原因。例如，当某个顾客向裁缝师傅定做一套服装时，这位裁缝师傅首先要获得这位顾客的一些数据，如身高、胸围、腰围、臂长和样式等等，然后根据这些数据制作服装。这些数据就是该顾客订做服装的具体需求。试想，如果裁缝师傅将顾客的这些具体需求弄错或者根本不知道的情况下，无论其如何精心制作，使用多好的面料，其所做的工作都将是枉然的！因为客户可能根本不能穿，或者穿得不舒适。这个例子说明，需求对最终产品能否适用是至关重要的。同理，对于软件开发来说，软件需求就是软件用户认为其所使用的软件应该具备的功能和性能。

对于软件需求的定义，不同的研究人员有不同的看法。A. Davis 认为，软件需求是从软件外部可见的、软件所具有的、满足于用户的特点、功能及属性等的集合<sup>[5]</sup>。I. Sommerville 认为，需求是问题信息和系统行为、特性、设计和实现约束的描述的集合<sup>[7]</sup>。而 M. Jackson 等人则认为，需求是客户希望在问题域内产生的效果<sup>[8,9]</sup>。在比较正式的文档中，IEEE 软件工程标准词汇表将需求定义为<sup>[10]</sup>：(1) 用户解决问题或达到目标所需的条件或能力；(2) 系统或系统部件要满足合同、标准、规范或其他正式规定文档所需具有的条件或能力。其中(1)是从用户的角度定义的，(2)是从软件系统的角度定义的。

关于软件需求还有其他不同的定义。产生这些不同形式的定义的原因，一是需求工程的发展过程还不太长，人们认识还在不断深入；二是真正的“需求”实际上是在人们的脑海中形成的，很难给予准确的定义。这也是导致需求工程难度很大的原因之一。不过，根据这些定义，我们可以认为软件需求是指软件系统必须满足的所有功能、性质和限制。

对于一个软件系统，不同的人对它应具有的功能和性能会有不同的需求。例如，对于文字处理系统这一软件，A 先生打算将其用于编辑英文论文。于是他希望文字处理系统具有能简单地描述数学公式、检查英语单词和文法等功能。B 先生则打算利用文字处理系统制作贺年卡片。他希望系统具有能处理图片和进行彩色打印的功能等。C 先生则希望文字处理系统具有简单地制作中文文档和快速打印的功能等。于是，即使对于同一个文字处理系统，由于使用者的立场不同，对其应具有的功能和性能也变得有所不同。因此，对于软

件开发者来说,在开发一个软件系统之前,应考虑该系统的利用者有什么样的需求,该软件能解决什么问题等。否则,开发出的软件要么利用者不满意,要么根本不能使用,从而导致软件开发成本和时间的浪费。

### 1.3 软件需求的分类

虽然对软件需求的定义有多种形式,但从软件用户多年来对软件的实际需求来看,软件的需求(或用户需求)通常可以大致分类如下:

- 目标需求:反映组织机构或客户对系统和产品提出的高层次的目标要求,其限定了项目的范围和项目应达到的目标。
- 业务需求:主要描述软件系统必须完成的任务、实际业务或工作流程等。软件开发人员通常可从业务需求进一步细化出具体的功能需求和非功能需求。
- 功能需求:指开发人员必须实现的软件功能或软件系统应具有的外部行为。
- 性能需求:指实现的软件系统功能应达到的技术指标,如:计算效率和精度、可靠性、可维护性和可扩展性等。
- 约束与限制:指软件开发人员在设计和实现软件系统时的限制,如:开发语言、使用的数据库等。

在这些需求中,功能需求描述系统做什么,由性能需求和约束与限制构成的非功能需求则为实现这些功能需求设定约束和限制。软件需求间的关系可分层次地表示,如图 1-1 所示。



图 1-1 软件需求间的层次关系

由以上的这些需求就可构成软件需求规格说明。下面我们通过与文字处理系统相关的部分需求来说明需求的分类。

- 目标需求:用户使用系统能有效纠正文档中的拼写错误,系统能满足用户的业务要求以及提高用户的工作效率。
- 业务需求:当找到文档中的拼写错误时,通过可供选择的单词表,选择单词表中的一个单词后,再替换掉原来的单词。

- 功能需求：查找文档中的单词，并高亮度地显示出错的单词。用对话框显示可供选择的单词表，实现整个文档范围内的替换。
- 非功能需求：检查单词的速度快，准确率要求达到 99%，系统的有效性和可靠性要高等。
- 约束与限制：文件内部格式要与 Word 系统一致。开发平台为 Linux 系统，使用 C 语言等。

## 1.4 需求规格说明

软件需求规格说明亦称软件需求规约或功能规格说明，可以说是需求工程最终产生的结果。所谓需求规格说明是软件所应满足的全部需求，并可用文档的方式完整和精确地陈述这些需求。需求规格说明是项目相关人员对将要开发的软件系统所达成的共识，是进行系统设计、实现、测试和验收的基本依据，也是整个软件开发过程中最重要的文档<sup>[11]</sup>。需求规格说明同时代表了权限的移交点：客户对需求规格说明的说明内容拥有最终发言权，而开发人员则需根据软件需求规格说明实施软件系统的开发。因此，最终开发出的软件系统是否能真实、全面地满足客户的要求，取决于需求规格说明是否真实、完整和一致地反映客户的真正意图。

需求规格说明应精确地描述一个软件系统必须提供的功能和性能，以及所要考虑的约束条件与限制。因此，需求规格说明也可以说是在 1.3 节中所定义的所有软件需求的集成，并使用某种描述语言如自然语言按照规定的书写格式编写的文档。关于需求规格说明的模板和具体内容，将在后面给予详细地说明。

需求规格说明在软件系统开发中起着十分重要的作用，但要把用自然语言表达的需求完整无误地表达出来并不是一件容易的工作。因为各人的理解不同，即便是同一个人，他在不同的时间也可能产生不同的理解。因此，作为一个质量较高的需求规格说明，通常应满足如下的特征<sup>[12]</sup>。

**完整性。**每一项需求必须将所要实现的功能描述清楚，以便开发人员获得设计和实现这些功能所需的必要信息。

**正确性。**每一项需求都必须准确地陈述其要开发的功能。做出正确判断的参考是需求的来源，如用户或高层的系统需求规格说明。如果软件需求与对应的系统需求相抵触，则是不正确的。只有用户才能确定需求的正确性。这就是一定要有用户的积极参与的原因。

**可行性。**每一项需求都必须在已知系统和环境的权能和限制范围内是可以实施的。为避免不可能实现的需求，最好在获取需求(或收集需求)的过程中，始终有一位软件开发小组的成员与需求分析人员或考虑市场的人员在一起，由他负责检查技术的可行性。