

<<软件需求>>

图书基本信息

书名：<<软件需求>>

13位ISBN编号：9787302098348

10位ISBN编号：7302098344

出版时间：2004-11

出版时间：清华大学出版社

作者：（美）Karl E.Wiegers

页数：357

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件需求>>

内容概要

《软件需求》(第2版)(Software Requirements)是有关软件需求的经典教材,本书全面而深入地讲述了软件开发中一个至关重要的问题——软件需求问题。

软件开发人员及用户往往容易忽略沟通的重要性,导致软件开发出来后,不能很好地满足用户的需要。

返工不仅在技术上给开发人员带来巨大的麻烦,并且会造成人力、物力和资源的浪费,还使软件性能深受影响,所以在开发早期提高项目需求分析的质量,减少重复劳动,通过控制项目范围的扩大及需求变更来达到按计划完成预定目标,是当前软件业急需解决的问题,也是本书讨论的主要内容。

《软件需求》(第2版)(Software Requirements)介绍了贯穿整个开发周期的管理需求工程的实用技术,包括多种可以促进用户、开发人员和管理层之间有效沟通的方法。

这一版对第一版进行了扩充,提供了新的实例,及作者在实际工作中遇到的各种实际案例和解决方案。

此外,还添加了新的章节、需求示例文档以及故障诊断指南等。

本书对第1版的内容进行了扩展,不仅对原有的知识点进行了补充,还引入了一些新知识,以求与时代发展同步。

<<软件需求>>

作者简介

Karl E. Wieggers是需求工程和软件过程改进领域内的顾问专家。作为Process Impact公司的首席顾问，他曾举办过许多培训讲习班，并多次在行业大会上发表演讲。Karl曾两次荣获Software Development Productivity Award，这一奖项是专门为奖励有助于提高生产率的产品和著作而设立的。

<<软件需求>>

书籍目录

第一部分什么是软件需求？

为什么要实现软件需求？

哪些人应参与软件需求第1章 软件需求基础知识1.1 软件需求的定义1.1.1 对需求的不同解释1.1.2 需求的层次1.1.3 不属于需求的内容1.2 需求的开发与管理1.2.1 需求开发1.2.2 需求管理1.3 所有项目都有需求1.4 优秀的团队遇到糟糕的需求1.4.1 用户参与不足1.4.2 用户需求扩展1.4.3 有歧义的需求1.4.4 镀金问题1.4.5 过于抽象的需求1.4.6 忽略了某类用户1.4.7 不准确的计划1.5 优质需求过程的好处1.6 优秀需求的特点1.6.1 需求陈述的特点1.6.2 需求规格说明的特点第2章 客户眼中的需求2.1 客户2.2 客户与开发人员的合作伙伴关系2.2.1 软件客户的权利法案2.2.2 软件客户的义务法案2.3 关于“签字”第3章 需求工程的推荐方法3.1 知识技能3.2 需求获取3.3 需求分析3.4 规格说明3.5 需求验证3.6 需求管理3.7 项目管理3.8 开始新实践3.9 需求开发过程第4章 需求分析员4.1 需求分析员的职责4.1.1 需求分析员的工作4.1.2 需求分析员必备的技能4.1.3 需求分析员必备的知识4.2 如何培养需求分析员4.2.1 从用户转为分析员4.2.2 从开发人员转为分析员4.2.3 主题专家4.3 营造合作的氛围

第二部分软件需求开发第5章 确定产品前景与项目范围5.1 通过业务需求定义前景5.1.1 相互矛盾的业务需求5.1.2 业务需求与用例5.2 前景与范围文档5.3 关联图5.4 保持范围的适度第6章 获取客户的需求6.1 需求的来源6.2 用户类6.3 寻找用户代表6.4 用户代言人6.4.1 外部的用户代言人6.4.2 对用户代言人的要求6.4.3 设置多位用户代言人6.4.4 如何让人接受用户代言人的概念6.4.5 用户代高言人应避免的陷阱6.5 谁来做出决策第7章 聆听客户的需求7.1 需求获取7.2 需求获取讨论会7.3 将客户的意见归类7.4 需求获取中的注意事项7.5 寻找遗漏的需求7.6 如何判断需求获取是否已完成第8章 理解用户需求8.1 用例法8.1.1 用例与使用场景8.1.2 确定用例8.1.3 编写用例8.1.4 用例与功能性需求8.1.5 用例的好处8.1.6 使用用例时应避免的问题8.2 事件—响应表第9章 遵守规则9.1 业务的规则9.1.1 事实9.1.2 约束9.1.3 动作触发规则9.1.4 推论9.1.5 计算9.2 在文档中记录业务规则9.3 业务规则和需求第10章 编写需求文档10.1 软件需求规格说明10.1.1 需求的标识10.1.2 处理不完整10.1.3 用户界面和软件需求规格说明10.2 软件需求规格说明模板10.3 编写需求文档的原则10.4 改进前后的需求示例10.5 数据字典第11章 一图胜千言11.1 需求建模11.2 从客户需求到分析模型11.3 数据流图11.4 实体—关系图11.5 状态转换图11.6 对话图11.7 类图11.8 判定表和判定树11.9 最后的提醒第12章 软件质量属性12.1 质量属性12.2 定义质量属性12.2.1 对用户重要的属性12.2.2 对开发人员重要的属性12.3 性能需求12.4 用Planguage定义非功能性需求12.5 属性的折中方案12.6 实现非功能性需求第13章 通过制作原型减少项目风险13.1 什么是原型和为什么要建立原型13.2 水半原型13.3 垂直原型13.4 废弃型原型13.5 演化型原型13.6 书面原型和电子原型13.7 原型评估13.8 创建原型所带来的风险13.9 原型法成功的因素第14章 设定需求优先级14.1 为什么要设定需求优先级14.2 优先级规则14.3 优先级的等级14.4 根据价值、成本和风险来设定优先级第15章 需求确认15.1 需求评审15.1.1 审查过程15.1.2 需求评审面临的困难15.2 测试需求15.3 制定验收标准第16章 需求开发面临的特殊难题16.1 维护项目的需求16.1.1 开始捕获信息16.1.2 亲身实践一下新的需求技术16.1.3 遵循跟踪链16.2 软件包解决方案的需求16.2.1 开发用例16.2.2 考虑业务规则16.2.3 定义质量需求16.3 外包项目的需求16.4 突发性项H的需求16.4.1 非正式用户需求规格说明16.4.2 现场客户16.4.3 尽早地而且要经常地设定优先级16.4.4 简单的变更管理第17章 超越需求开发17.1 从需求到项目规划17.1.1 需求和预估17.1.2 需求和进度安排17.2 从需求到设计和编码17.3 从需求到测试17.4 从需求到成功

第三部分软件需求管理第18章 需求管理的原则和实践18.1 需求基线18.2 需求管理过程18.3 需求版本控制18.4 需求属性18.5 跟踪需求状态18.6 评估需求管理的工作量第19章 变更管理19.1 管理范围蔓延19.2 变更控制过程19.2.1 变更控制策略19.2.2 变更控制过程描述19.3 变更控制委员会19.3.1 CCB的组成19.3.2 CCB规章19.4 变更控制工具19.5 测量变更活动19.6 变更需要付出代价：影响分析19.6.1 影响分析的过程19.6.2 影响分析报告模板第20章 需求链中的联系链20.1 需求跟踪20.2 需求跟踪动机20.3 需求跟踪矩阵20.4 需求跟踪工具20.5 需求跟踪过程20.6 需求跟踪可行吗？

必要吗？

第21章 需求管理工具21.1 使用需求管理工具的益处21.2 需求管理工具的功能21.3 实现需求管理自动化21.3.1 选择适当的工具21.3.2 改变文化21.3.3 使需求管理工具服务于自己

第四部分实现需求工程第22章 改进需求过程22.1 需求与其他项目过程的联系22.2 需求和各涉众组22.3 软件过程改进的基本原则22.4

<<软件需求>>

过程改进周期22.4.1 评估当前采用的方法22.4.2 规划改进活动22.4.3 建立、实验并实现新过程22.4.4 评估结果22.5 需求工程过程资产22.5.1 需求开发过程资产22.5.2 需求管理过程资产22.6 需求过程改进路线图

第23章 软件需求与风险管理23.1 软件风险管理基本原理23.1.1 风险管理的要素23.1.2 编写项目风险文档23.1.3 制定风险管理计划23.2 与需求相关的风险23.2.1 需求获取23.2.2 需求分析23.2.3 编写需求规格说明23.2.4 需求确认23.2.5 需求管理23.3 风险管理是我们的好帮手附录A 当前需求实践的自我评估附录B 需求和过程改进模型B.1 软件能力成熟度模型B.2 CMMI-SE/SWB.2.1 需求管理过程域B.2.2 需求开发过程域附录C 需求错误诊断指南C.1 根本原因分析C.2 需求问题的常见现象C.3 实现解决方案常常会遇到的障碍附录D 需求文档范例D.1 前景和范围文档D.1.1 业务需求D.1.2 解决方案的前景D.1.3 范围和局限性D.1.4 业务上下文D.2 用例D.3 软件需求规格说明D.3.1 介绍D.3.2 总体描述D.3.3 系统特性D.3.4 外部接口需求D.3.5 其他非功能性需求D.3.6 附录A 数据字典和数据模型D.3.7 附录B 分析模型D.4 业务规则术语表结语

<<软件需求>>

章节摘录

第1章 软件需求基础知识 “您好。
是Phil么？
我是人力资源部的Maria。
我们使用您做的人事管理系统时遇到点问题。
有位女职员想把名字改成Sparkle Starlight，可我们在系统里怎么都改不过来。
能帮帮忙吗？
” “她嫁了一个姓Starlight的人么？
” “没有，她没结婚，只是改了名字。
” Maria答道，“所以才有这样的麻烦。
好像只有在婚姻状况改变时才能改名字。
” “是的。
我从来没想到谁会无缘无故地改名字。
我们讨论系统的时候您可没跟我提过这种可能。
所以只能从修改婚姻状况的对话框进入修改名字的对话框。
” “谁都可以改名字。
只要他愿意，随时都行，这是合法的。
我以为您知道呢。
” Maria说，“星期五之前必须搞定，不然Sparkle就兑换不了支票。
您能在那之前把这个错误改过来么？
” “这根本就不是错误！
” Phil反驳道，“我从来不知道您需要这个功能。
我正忙着做一个新的性能评估系统。
而且我还要对人事管理系统进行一些修改，”（话筒里传来翻纸的声音），“对，这就有一个。
月底没准能改好，这周肯定不行，抱歉。
下回早点儿告诉我，麻烦把问题写下来。
” “我怎么跟Sparkle说？
” Maria问，“兑不了支票她就得赊账。
” “搞清楚，Maria，这可不是我的错。
” Phil抗议了，“如果您当时告诉我要能随时修改姓名，就不会有今天的事。
您不能怪我没猜到您的想法。
” Maria很生气却无可奈何，只好气冲冲地说：“好了。
就是这种事让我恨透了计算机。
改好了马上通知我，这总可以吧。
” 如果您曾经有过这种客户经历，您肯定明白这种连最基本的操作都完不成的软件多么让人烦恼。
即便开发人员最终可能会帮您改好，您通常也不愿总求助于他。
然而，站在开发人员的立场，如果系统完成后才从用户那里得知需要什么功能，也的确很难接受。
已经完全按最初的要求实现了系统，却不得不停下手头的项目去修改系统以便满足用户的新需求，这也是件很讨厌的事。
许多软件问题都源于收集、记录、协商和修改产品需求过程中的方式不当。
前面Phil和Maria的例子中就有这些方面的问题，包括信息收集方式不正规，没有明确提出想要的功能，假设是未经沟通的错误假设，需求的定义不够充分，以及未经仔细考虑进行需求变更等。

<<软件需求>>

后记

结语 软件项目要取得成功，最重要的莫过于要了解需要解决哪些问题，需求为项目的成功奠定了基础。

如果开发团队及其客户没有就产品功能和特性达成共识，那么其结果很可能会令人很不愉快，而这是我们都不愿意看到，且应该尽量避免的。

如果当前的需求实践并不能使您得到满意的结果，那么可以仔细考虑一下本书中提出的哪些技术可能会对您有所帮助，并有选择地应用这些技术。

有效需求工程的重要原则包括： 客户代表尽早介入需求工程，并且要有足够的客户参与。

迭代地或增量地开发需求。

以各种方式来表示需求，以确保每个人都能理解。

确保需求对所有涉众的完整性和正确性。

控制需求变更方式。

要改变软件开发组织的工作方式并不是一件容易的事，因为我们很难证实当前的工作方式不如我们喜欢的方式好，也很难断定下一次应该尝试哪种方法。

我们很难抽出时间学习新技术、开发改进的软件过程、试验并调整过程、以及将它们传达到组织的其他部门。

使涉众各方确信必须进行变更也是一件很困难的事。

但是，如果不改变工作方式，我们就没有理由相信当前项目将比上一个项目更好 软件过程改进的成功取决于下面几个因素： 清楚地描述组织的痛苦所在。

一次只集中改进少许领域。

目标明确，定义改进活动的计划。

描述与组织变更相关的人为因素和文化因素。

说服高级经理将过程改进视为业务成功的战略投资。

当我们定义图来改进需求工程实践并开始付诸于行动时，要牢记上述过程改进原则。

保留那些适合于组织和团队的实践方法。

如果我们积极地应用已知的良好实践，并依靠常识，就可以显著地改进处理项目需求的方式，并获得由此带来的全部优点和益处。

另外还应记住，没有优秀的需求，软件就像是一个巧克力盒子：我们决不会知道我们将得到什么样的产品。

<<软件需求>>

编辑推荐

需求工程的最佳实践，更多的示例，新主题，以及需求文档范例如果没有正式的可验证的软件需求及有效管理需求的系统，开发人员开发出来的程序通常会与客户需要的程序不一致。

在本书中，Karl Wiegiers对其获奖文章中的最佳实践进行了整理和扩充，这些实践是所有软件开发参与者的重要参考依据。

《软件需求》(第2版)(Software Requirements)可以作为计算机专业及软件工程专业学生的教材使用，也非常适合作为项目经理、软件开发人员的指导性参考书。

<<软件需求>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>