

高职高专计算机系列规划教材

软件行业岗位参考指导与实训丛书

丛书主编 孔 敏

软件需求分析师

岗位指导教程

主 编 桂 超
副主编 蔡 洁



南京出版

南京大学出版社

高职高专计算机系列规划教材

软件行业岗位参考指南与实训丛书

丛书主编 孔 敏

软件需求分析师 岗位指导教程

主 编 桂 超
副主编 蔡 洁
编 者 桂 超 蔡 洁 夏孝云
井 辉 张 月 贾 云



图书在版编目(CIP)数据

软件需求分析师岗位指导教程 / 桂超主编. — 南京:
南京大学出版社, 2017. 1

(软件行业岗位参考指南与实训丛书 / 孔敏主编)

ISBN 978 - 7 - 305 - 14515 - 5

I. ①软… II. ①桂… III. ①软件需求分析—教材
IV. ①TP311.52

中国版本图书馆 CIP 数据核字(2014)第 309926 号

出版发行 南京大学出版社

社 址 南京市汉口路 22 号 邮 编 210093

出版人 金鑫荣

丛 书 名 软件行业岗位参考指南与实训丛书

丛书主编 孔 敏

书 名 软件需求分析师岗位指导教程

主 编 桂 超

责任编辑 徐 鹏 吴 汀 编辑热线 025 - 83686531

照 排 南京理工大学资产经营有限公司

印 刷 南京人文印务有限公司

开 本 787×1092 1/16 印张 9.5 字数 231 千

版 次 2017 年 1 月第 1 版 2017 年 1 月第 1 次印刷

ISBN 978 - 7 - 305 - 14515 - 5

定 价 28.00 元

网 址: <http://www.njupco.com>

官方微博: <http://weibo.com/njupco>

官方微信号: njupress

销售咨询热线: (025)83594756

* 版权所有, 侵权必究

* 凡购买南大版图书, 如有印装质量问题, 请与所购
图书销售部门联系调换

前言

软件行业发展到今天,岗位分工越来越详细。软件公司对各岗位也逐渐有了明确的职责定义和业务范围。通常,一个软件系统开发生命周期内,可行性分析、需求分析、系统设计、编码、调试和测试、验收与运行、维护升级到废弃等阶段,这些阶段中需要不同角色的人员来参与,从需求分析师、软件架构师、系统分析师、软件开发工程师、测试工程师到维护工程师等。但是,这些角色往往会担负在1个或者2~3个人的肩上。一个较小型的项目,也许软件的需求分析、架构设计和分析以及程序的开发都由一个人来完成。所以,一个软件开发工程师必须具备一定的软件架构的知识和技能,才能在不同级别的软件项目中胜任自己的岗位和任务要求,这也是目前软件公司为节约成本、提高附加值而对软件开发工程师提出的更高的要求。

本书为谁而写?

本书为软件开发工程师和学习软件需求分析的人员而写,是入门级培养。为做软件需求分析工作时提供一些方法、参考和借鉴。通过本书的学习之后,基本上对软件需求分析的内容和过程有一个比较清晰的思路,可以通过书中的输入、输出成果物的模板进行基本的软件需求分析工作,以后可以通过阅读更高级的软件需求分析书籍提高在此领域的水平。

写给软件开发工程师:

作为软件开发工程师,你一定掌握了一门或者几门编程语言,有一定的项目开发实践经验,或者正从事着软件项目的开发,你也一直乐在其中。可是有一天,项目经理告诉你,这个软件系统将由你负责完成需求分析的工作,怎么办?到底该从哪里入手呢?本书以软件需求分析作业指导的形式,从软件需求分析工作中必须涉及的各个环节入手,逐一带领读者学习软件需求分析各个阶段的任务、工作流程和输入、输出成果,以及必须要了解的一些知识。软件开发工程师在使用本教程时,不一定要从第一章开始读起,可以根据实际项目需要,有针对性地挑选合适的章节有的放矢地学习。教程最后提供的各个输出物的模板也会更清楚地帮助你梳理软件需求分析的各个环节,告知你在各个阶段需要什么产出物。

写给教师:

作为高校教师,如果您的学校开设了软件开发类似的课程,或者软件开发综合实训课程,那么您一定会选择理论体系更为完善,叙述更为系统化的教程。那么在软件需求分析实践教学环节,您需要一本能够指导学生实践的教程。本书从软件需求分析师岗位入手,以工作流程为轴线,逐一介绍了软件需求分析中涉及的工作任务,如何开展这些工作任务,以及典型的标准文档模板。这些内容对于指导您的学生来说,将是非常便利的。您还可以把该教程作为毕业实训或项目实训的一本参考教程,让学生在项目开展的过程中参考本教程。

写给在校学生:

学习计算机专业的同学都知道,如果想把一门编程语言、一门技术学好,一本教材是不

够的,必须去图书馆借阅与课程相关的书籍进行深入学习,从不同视角和不同层次进行二次学习,这样才是一个学习的好方法。当您在进行课程实训时,实训的内容通常是与一门课程的内容紧密结合的,例如:C#语言、Java语言、数据库等,内容比较单一,基本上重点都是在编码和程序逻辑的功能实现上,不会考虑需求分析。但是,当进入毕业实训和项目综合实训阶段时,不但实训的项目功能复杂了,业务逻辑也多了,不仅要考虑编码,还要考虑软件的需求分析、架构、人员的分配、风险的识别等。本教程可以在软件需求分析领域提供操作级别的指导,在软件需求分析的主要环节中告诉你现在应该做什么、需要什么、怎么做、谁来做、下面要做什么、做出什么成果。

如何使用本教程?

本教程突破传统的软件需求分析书籍知识体系化的编写方式,注重软件需求分析岗位工作过程的描述和内容分解,适合作为软件架构岗位的工作指导教程。

第1章是对软件需求分析师岗位的作业进行整体的指导,读者将会对本书的写作风格有一个清晰的认识。

第2章到第7章是针对软件需求分析师岗位的工作任务进行逐一的作业指导,通过流程图,读者将会对具体的工作流有一个了解,再通过文中工作步骤的分解和指导,读者将会逐渐了解具体工作任务的内容。

第8章是软件需求分析师岗位的评审作业指导。评审在软件项目周期中是一个非常常见和重要的内容,通过这个章节的介绍,读者将学习到在软件需求分析过程中应该何时评审、评审什么内容、评审的流程是什么、什么人员参与评审等。

文章最后的附录部分提供了软件需求分析的文档模板,借助模板,读者可以很快了解相应任务的内容,再通过参考教程前12个章节的过程指导和第13章的实例指导,将最终形成一个格式规范、内容充实的项目文档。

本书作者介绍:

本书是集体智慧的结晶,参加本书编写和制作的人员有:

桂超:负责教程第1章、第3章、第4章、第5章、第6章,第9章以及全书统稿、写作思路和风格设计。

蔡洁:负责教程第2章、第7章的编写。

夏孝云:负责教程第8章,附录部分的编写,以及全书各章节案例的设计和总体风格的把握。

由于作者水平有限,加之创作时间比较仓促,本书不足之处在所难免,欢迎广大读者批评指正。我们的邮箱是1051970037@qq.com。

作者

2016年11月

目 录

前 言	1
第 1 章 软件需求分析概述	1
1.1 软件需求分析	1
1.1.1 软件需求分类	2
1.1.2 需求说明的特征	3
1.1.3 软件需求的重要性	4
1.1.4 软件需求规格说明书	5
1.1.5 需求规格说明的特点	6
1.1.6 正确的需求与有缺陷的需求	6
1.1.7 需求的演进	8
1.2 软件需求管理	8
1.2.1 需求管理目的	9
1.2.2 软件工程方法	9
第 2 章 软件需求分析师岗位作业指导概述	15
2.1 需求分析师岗位概述	15
2.2 工作指导概述	16
2.3 需求分析工作流程概述	16
2.3.1 输入概述	16
2.3.2 工作流程概述	17
2.3.3 需求分析子工作概述	17
2.3.4 需求分析工作步骤说明	21
2.3.5 完成 SRS 工作流程简述	21
2.3.6 输出	23
第 3 章 需求获取作业指导	24
3.1 概述	24

3.2	输入	24
3.3	workflow概述	25
3.4	需求获取工作流程图	25
3.5	需求获取作业指导说明	27
3.5.1	准备	27
3.5.2	需求采集	29
3.5.3	用户访谈	29
3.5.4	小组会议	31
3.5.5	问卷调查	32
3.6	头脑风暴会议	34
3.7	交流的能力与方法	36
3.8	需求的分析技术	39
3.9	需求的确认	40
3.10	输出	42
3.11	需求获取问题总结	42
第4章	业务建模作业指导	48
4.1	概述	48
4.2	输入	48
4.3	workflow	48
4.4	业务建模 workflow图	49
4.5	业务建模 workflow步骤描述	50
4.6	业务建模步骤指导说明	50
4.6.1	业务建模规模	50
4.6.2	确定业务范围	51
4.6.3	确定业务主角	51
4.6.4	术语表	52
4.6.5	识别业务流程	53
4.6.6	分析业务流程	54
4.6.7	业务规则	56
4.6.8	业务用例研讨会	57
4.7	输出	58
第5章	需求分型作业指导	67
5.1	概述	67
5.2	需求的两难问题	68
5.3	输入	69

5.4 workflow	70
5.4.1 workflow图	70
5.4.2 需求分型工作步骤描述	70
5.5 需求分型工作步骤说明	70
5.5.1 功能性需求分型标准	70
5.5.2 非功能性需求分型标准	70
5.5.3 用户需求分型列表说明	74
5.6 发现和归纳功能性需求	74
5.7 发现和归纳非功能性需求	78
5.7.1 用例与非功能性需求	79
5.7.2 非功能性需求类型与软件质量模型	79
5.7.3 定义质量属性	83
5.8 输出	86
第6章 用例建模作业指导	87
6.1 概述	87
6.2 输入	89
6.3 workflow	89
6.3.1 workflow图	89
6.3.2 workflow步骤描述	91
6.4 步骤说明	92
6.4.1 选择系统边界	92
6.4.2 识别系统参与者	92
6.4.3 识别用例	93
6.4.4 找出参与者之间的关系	95
6.4.5 找出用例之间的关系	96
6.4.6 找出参与者和用例之间的关系	102
6.4.7 用例图	102
6.4.8 术语表	103
6.4.9 用例规格说明	104
6.5 输出	111
第7章 《软件需求规格说明书》编写参考指南	112
软件需求规格说明书	112
第8章 需求评审作业指导	121
8.1 概述	121

8.2	需求评审方法	122
8.3	审查参与者	123
8.4	审查流程	124
8.5	审查内容	125
8.6	进入和退出审查的标准	129
8.7	审查输出	129
第9章	需求分析师岗位实训任务与操作案例	132
9.1	需求分析师岗位实训任务	132
9.1.1	实训任务名称	132
9.1.2	实训任务场景	132
9.1.3	实训目标任务	133
9.1.4	实训所需环境	133
9.1.5	实训主要步骤	133
9.1.6	实训总结要求	134
9.1.7	实训操作参考	134
9.2	软件需求分析师岗位作业操作案例	134
9.2.1	任务场景	134
9.2.2	任务目标	135
9.2.3	任务实施步骤	135
附 录	139
	软件行业岗位实训报告	139
	实训报告目录	140
	实训报告内容	141

第 1 章 软件需求分析概述

1.1 软件需求分析

软件需求分析就是把软件计划期间建立的软件可行性分析进行求精和细化,分析各种可能的解法,并且分配给各个软件元素。需求分析是软件定义阶段中的最后一步,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。

1. 需求分析的定义

IEEE(1997)软件工程标准词汇表中定义需求为:

- (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。
- (3) 一种反映上面第一或第二项描述的条件或能力的文档说明。

2. 需求分析的任务

深入描述软件的功能和性能,确定软件设计的约束和软件同其他系统元素的接口细节,定义软件的其他有效性需求,借助于当前系统的逻辑模型导出目标系统逻辑模型,解决目标系统“做什么”的问题。

需求分析工作的基本任务是需求分析师解决“做什么”的问题,需求分析师与用户沟通,能全面了解用户对软件系统的要求。最终需求分析师能够编写《软件需求分析说明书》定义软件系统,并与客户确认需求。

3. 需求分析的作用

开发软件系统最困难的部分就是要准确说明开发什么,最困难的概念性工作便是要编写出详细的技术需求,这包括所有面向用户、面向机器和其他软件系统的接口。如果做错,这将是会最终给系统带来极大损害的一部分,并且以后再对它进行修改也极为困难。目前,国内各类软件产品十分庞杂,一个企业内部可能有几个系统并立运行,它们之间的接口是系统开发人员最头痛的问题之一。对于商业最终用户应用程序,企业信息系统和软件作为一个大系统的一部分的产品是显而易见的。但对于软件开发人员来说,并没有编写出客户认可的需求文档。我们如何知道项目于何时结束?而如果我们不知道什么对客户来说是重要

的,那我们又如何能使客户感到满意呢?

即便并非出于商业目的的软件需求也是必需的。如库、组件和工具这些供开发小组内部使用的软件。当然你可能偶尔无须文档说明就能与其他人意见较为一致,但更常见的是出现重复返工这种不可避免的后果,而重新编制代码的代价则远远超过重写一份需求文档的代价,这些血的教训正在国内的软件开发者身上发生。

1.1.1 软件需求分类

一个好的需求应该具备很强的层次感。从软件用户对软件的实际需求来看,软件需求各部分关系如图 1-1 所示。

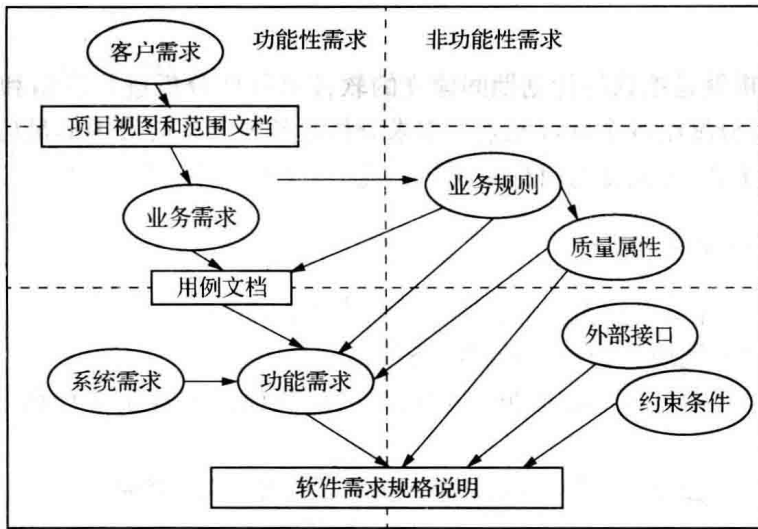


图 1-1 软件需求分层及关系

1. 客户需求

客户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,客户需求限定了项目的范围和项目应达到的目标。客户需求在项目视图与范围文档中予以说明,有时候客户需求是以任务书的方式表达的。客户需求一般表述比较抽象,缺乏很多必要的细节。

2. 业务需求

业务需求也叫产品需求,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。这种需求需要考虑产品的运行方案,在用例(use case)文档或方案脚本(scenario)说明中予以说明。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。

3. 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,使客户能完成他们的任务,从而满足了客户需求。

4. 非功能需求

非功能需求指那些不直接与系统具体工作相关的一类需求。描述了软件产品必须遵从的标准、规范和合业务规则,包括企业方针、政府条例、工业标准、会计准则和计算方法等。

(1) 业务规则

业务规则本身并非软件需求,因为它不属于任何特定软件系统的范围。然而,业务规则常常会限制谁能够执行某些特定用例,或者规定系统为符合相关规则必须实现某些特定功能。有时,功能中特定的质量属性(通过功能实现)也源于业务规则。所以,对某些功能需求进行追溯时,会发现其来源正是一条特定的业务规则。

(2) 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。

(3) 约束

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。

(4) 质量属性

质量属性通过多种角度对产品的特点进行描述,从而反映产品功能。

开发人员在关注软件功能和行为的同时,也要考虑软件产品的易用性、运行速度、可靠性、如何进行异常处理等特性。因此,一个好的软件还必须满足一系列非功能需求。

1.1.2 需求说明的特征

我们可以归纳出好的需求说明应该具备如下特征,这也是需求分析的目标:

1. 完整性

每一项需求都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所有必要信息。

2. 正确性

每一项需求都必须准确陈述其要开发的功能。做出正确判断的参考是需求的来源,如客户或高层的系统需求规格说明。若软件需求与对应的系统需求相抵触,则是不正确的。

3. 可行性

每一项需求都必须是在已知系统和环境的限制范围内可以实施的。为避免不可行的需求,最好在获取需求过程中始终有一位软件工程小组的组员与需求分析人员或考虑市场的人员在一起工作,由他负责检查技术可行性。

4. 必要性

每一项需求都应把客户真正所需要的和最终系统所需遵从的标准记录下来。“必要性”也可以理解为每项需求都是用来授权编写文档的“根源”。

5. 划分优先级

给每项需求、特性或用例分配一个实施优先级,以指明它在特定产品中所占的分量。如果把所有的需求都看作同样重要,那么项目管理者在开发或节省预算或调度中就丧失控制自由度。

6. 无二义性

对所有阅读需求说明的读者,都只能有一个明确统一的解释,避免二义性的有效方法包括对需求文档的正规审查,编写测试用例,开发原型等。

7. 可验证性

检查一下每项需求是否能够通过设计测试用例,来确定产品是否确实按需求实现了。如果需求不可验证,则确定其是主观臆断,而非客观分析。一份前后矛盾、不可行或有二义性的需求也是不可验证的。

1.1.3 软件需求的重要性

1. 需求是产生软件缺陷的最大原因

随着经济全球化进程的不断推进,要增强产品的国际竞争力,产品质量作为经济发展的战略问题变得越来越重要。软件质量问题相当程度上是以软件缺陷的形式出现的。软件缺陷是由很多原因造成的,但从整个开发周期的统计结果来看,我们会意外地发现需求规格说明书是软件缺陷出现最多的地方。需求分析不到位,是产生软件缺陷的最大原因。产生这种情况的原因如下:

- 客户一般是非计算机专业,软件开发人员与客户沟通存在着比较大的困难,对要开发的产品功能理解也不一致。
- 由于软件产品还没有设计、开发,完全靠想象去描述系统的实现结果,所以有些特性还不够清晰。
- 客户的需求总是在不断变化的,容易引起前后文、上下文的矛盾和需求描述的不一致。
- 需求分析没有得到足够的重视,在规格说明书的设计和写作上投入的人力、时间都不足。
- 没有在整个开发队伍中进行充分的沟通,有时只有架构师得到比较多的信息,造成开发团队对问题理解的不一致。

2. 软件开发最困难的工作是编写需求

需求开发在软件项目中扮演的角色非常重要:开发软件系统最困难的部分就是准确说明开发什么。最困难的概念性工作便是编写出详细技术需求,这包括所有面向客户、面向机器和其他软件系统的接口。同时,这也是一旦做错将最终会给系统带来极大损害的部分,并

且以后再对它进行修改也极为困难。

3. 需求开发是定义产品的投资

花费在了解客户需求上的时间,是促使项目成功的一种高层次投资。这对于用户应用程序、企业信息系统和作为大系统的一部分的软件产品是显而易见的。对于开发人员来说,没有编写出客户认可的需求文档,我们如何知道项目于何时结束?如果我们不知道什么对客户来说是重要的,那我们又如何能使客户感到满意呢?

4. 需求有利于内部理解的一致

有些并非出于商业目的的软件,如软件库、组件和工具这些供开发小组内部使用的软件,偶尔不需要文档说明就能与其他人意见较为一致。但更常见的还是出现重复返工这种不可避免的后果,而重新编制代码的代价则远远超过重写一份需求文档的代价。

5. 正确的需求是测试的基础

假如我们为一个要集成到“商业错误跟踪系统”中的简单电子邮件界面写了一页需求说明,而 Unix 系统管理员在为处理电子邮件写脚本时,就会发现简单的一张需求清单竟是如此有用。我们还会发现,当依据需求对系统进行测试时,此系统不仅非常清晰地实现了所有必需功能,而且还可以轻易发现任何最终的错误。正确的需求是测试的基础,在最普遍意义上都是正确的。

1.1.4 软件需求规格说明书

软件需求分析是软件项目中一项十分重要的工作,需求分析工作的结果是否成功有效,直接影响了整个软件工程项目的最终成败。需求分析会产生项目文档软件需求分析规格说明书(SRS)。该文档是用户和系统开发者双方对该软件系统的功能规定与性能规定的共同理解,使之成为整个开发工作的基础。软件需求分析规格说明书主要内容包括该软件的功能、性能、接口、界面及数据要求说明等。该文档完成后,要经过适当的评审,评审通过后,双方要签字确认,形成项目的第一个里程碑。

在软件需求规格说明书中,说明的功能需求充分描述了软件系统所应具有的外部行为。软件需求规格说明在开发、测试、质量保证、项目管理以及相关项目功能中都起了重要的作用。对一个大型系统来说,软件功能需求也许只是系统需求的一个子集,因为另外一些可能属于子系统(或软件部件)。作为功能需求的补充,软件需求规格说明还应包括非功能需求,它描述了系统展现给用户的行为和执行的操作等。它包括产品必须遵从的标准、规范和合约;外部界面的具体细节;性能要求;设计或实现的约束条件及质量属性。所谓约束是指对开发人员在软件产品设计和构造上的限制。质量属性是通过多种角度对产品的特点进行描述,从而反映产品功能。多角度描述产品对用户和开发人员都极为重要。

下面以一个字处理程序为例,来说明需求的不同种类。业务需求可能是:“用户能有效地纠正文档中的拼写错误”,该产品的包装盒封面上可能会标明这是个满足业务需求的拼写检查器。而对应的用户需求可能是“找出文档中的拼写错误并通过一个提供的替换项列表

来供选择替换拼错的词”。同时,该拼写检查器还有许多功能需求,如找到并高亮度提示错词的操作;显示提供替换词的对话框以及实现整个文档范围的替换。从以上定义可以发现,需求并未包括设计细节、实现细节、项目计划信息或测试信息。需求与这些没有关系,它关注的是充分说明你究竟想开发什么。项目也有其他方面的需求,如开发环境需求或发布产品及移植到支撑环境的需求。尽管这些需求对项目成功也至关重要,但它们并非本书所要重点讨论的。

1.1.5 需求规格说明的特点

需求分析的交付物是“需求规格说明”,好的需求文档应该具备如下特点:

1. 完整性

不能遗漏任何必要的需求信息。注重用户的任务而不是系统的功能,将有助于避免不完整性。

2. 一致性

一致性是指与其他软件需求或高层(系统,业务)需求不相矛盾。在开发前,必须解决所有需求间的不一致部分。

3. 可修改性

在必要时或为维护每一需求变更历史记录时,应该修订 SRS。这就要求每项需求要独立标出,并与别的需求区别开来,从而无二义性。每项需求只应在 SRS 中出现一次。这样更改时,易于保持一致性。

4. 可跟踪性

应能在每项软件需求与它的业务和设计元素、源代码、测试用例之间建立起跟踪链,这种可跟踪性要求每项需求以一种结构化的、粒度好的方式编写并单独标明,而不是大段的叙述。

上述对于需求开发的宏观描述非常重要,有了这些宏观理解,我们就可为自己下一步工作设定一个目标和界限,在出现困惑的时候,就不至于迷茫或者无从下手,思路也会更加清晰。一个好的需求分析人员所具备的素质,包括理念、方法论、性格以及正确的哲学观念,对这些问题的融会贯通,是我们成功之本。

1.1.6 正确的需求与有缺陷的需求

需求获取的方法很多,那么什么样的需求获取方法才是正确的呢?

1. 关键是关注规律而不仅仅是记录

需求过程之难,并不在于某些书写标准或规范,也不是仅仅了解客户需要什么,而是需

要深入地从各个角度探究客户需求,总结出客户需求的本质和将来变化的规律,这就对需求分析师提出了很高的要求。需求分析非常重要,如果事情说不清楚,就没有办法做好。

2. 实行有效的通力合作

正确的需求过程强调产品开发中的通力合作,包括在整个项目过程中多方风险承担者的积极努力。收集需求能使开发小组更好地了解市场,而市场因素是任何项目成功的一个关键因素。在产品开发前了解这些比在遭到客户批评后才意识到要节约很多成本。让客户和用户积极参与需求收集过程能使产品更富有吸引力,而且能拥有忠实的客户关系。

3. 全方位考虑需求问题

将选定系统的需求明确地分配到各软件子系统,强调采用产品工程的系统方法。这样能简化硬软件的集成,也能确保软硬件系统功能匹配适当。有效地变更控制和影响分析过程也能降低需求变更带来的负面影响。最后,将需求编写成清晰、无二义性的文档将会极大地有利于系统测试,确保产品质量,以使所有风险承担者感到满意。

4. 有缺陷的需求

但是,现实中我们会遇到太多的、有缺陷的需求收集方法,归纳起来,原因不外是下面几点:

(1) 无足够用户参与

客户经常不明白为什么收集需求和确保需求质量需花费那么多功夫,开发人员可能也不重视客户的参与。究其原因:一是因为与客户合作不如编写代码有意义;二是因为开发人员觉得已经明白客户的需求了。在某些情况下,与实际使用产品的用户直接接触很困难,而客户也不太明白自己的真正需求。

(2) 客户需求的不断增加

在开发中若不断地补充需求,项目就越变越庞大,以致超过其计划及预算范围。计划并不总是与项目需求规模与复杂性、风险、开发生产率及需求变更实际情况相一致,这使得问题更难解决。实际上,问题根源在于客户需求的改变和开发者对新需求所做的修改。

产品开发中不断延续的变更会使其整体结构日渐紊乱,补丁代码也使得整个程序难以理解和维护。插入补丁代码使模块违背强内聚、松耦合的设计原则,特别是如果项目配置管理工作不完善的话,收回变更和删除特性会带来问题。

如果能够尽早发现可能带来变更的特性,我们就能开发一个更为健壮的结构,并能更好地适应它。这样设计阶段需求变更不会直接导致补丁代码,同时也有利于减少因变更导致的质量下降。

(3) 模棱两可的需求

模棱两可是需求规格说明中最为可怕的问题。它的一层含义是指诸多读者对需求说明产生了不同的理解;另一层含义是指单个读者能用不止一个方式来解释某个需求说明。

模棱两可的需求会使不同的风险承担者产生不同的期望,它会使开发人员为错误问题而浪费时间,并且使测试者与开发者所期望的不一致。一位系统测试人员曾坦言,她所在的

测试组经常对需求理解有误,以致不得不重写许多测试用例并重做许多测试。

(4) 不必要的特性

“画蛇添足”是指开发人员试图增加一些“用户欣赏”但需求规格说明中并未涉及的新功能。经常发生的情况是客户并不认为这些功能性很有用,以致在其上耗费的努力“白搭”了。

开发人员应当为客户构思方案并为他们提供一些具有创新意识的思路,具体提供哪些功能要在客户所需与开发人员在允许时限内的技术可行性之间求得平衡,开发人员应努力使功能简单易用,而不要未经客户同意,擅自脱离客户要求,自作主张。同样,客户有时也可能要求一些看上去很“酷”,但缺乏实用价值的功能,而实现这些功能只能徒耗时间和成本。

(5) 过于精简的规格说明

有时,客户并不明白需求分析有如此重要,于是只做一份简略之至的规格说明,仅涉及产品概念上的内容,然后让开发人员在项目进展中完善,结果很可能出现的是开发人员先建立产品的结构之后再完成需求说明。这种方法可能适合于尖端研究性的产品或需求本身就十分灵活的情况。但在大多数情况下,这会给开发人员带来挫折,也会给客户带来烦恼,因为他们无法得到他们所设想的产品。

1.1.7 需求的演进

需求开发是一个演进的过程,也是对问题理解不断深化的过程。在项目开始的时候,需求分析师关注的是产品将要支持的业务(或者称之为工作),在这个阶段,分析师研究场景及其他模型,与客户讨论在业务上应该做什么?要解决什么问题?在这些问题上达成一致意见,这时候的需求可以称之为“客户需求”。随着对业务理解的加深,风险承担者对有助于自己工作的最佳产品做出决定,现在需求分析师开始确定产品的详细功能,并且编写“产品需求”。非功能性需求几乎是在同一个时间导出来的,包括质量属性和限制条件一起被记录下来。此时,需求使用一种与技术无关的方式写下来,它规定了产品应该为业务做些什么,但没有规定应该用什么技术方法来实现。

1.2 软件需求管理

无数事实告诉我们,如果没有很好的需求管理,就不可能很好地完成需求开发工作,而且需求开发也无法寻找到正确的方法。因此,我们首先给出需求管理的推荐实践,然后逐步推进到有组织的方式与顾客共同开发需求,这个顺序是合理的。

很多专业人员都相信需求管理对于任何项目管理者来说都是最重要的工作。因为人们普遍面对着范围蔓延、偏离方向的期望、放弃、迷茫的顾客,这些情况都使项目的前景变得暗淡。因此,我们应该认识到需求管理是一项最基本、最基础的管理职责。

需求不断变更是我们常常遇到的情况,由于人们认识事物的规律,是不可能早期把一切都描述清楚的,需求变更是不可避免的。在一个健全的需求管理方法之下,我们和顾客就可以使用一种机制来同步,共同运作,并且不会产生分歧。