

用例驱动技术在需求获取过程中的应用

刘旭勇^{1,2}

(1. 西北大学, 陕西 西安 710069; 2. 陕西省行政学院, 陕西 西安 710068)

摘 要:需求获取是软件开发过程中非常重要的环节,它贯穿于整个软件生命周期。针对系统客户和开发人员由于认识问题角度和知识的差异而存在的需求获取障碍,提出了一种基于用例的需求获取方法。该方法以用户和任务为中心,设计了一个用户需求描述模板,引导用户填写需求描述信息,开发人员利用用例驱动技术对需求描述信息进行分析 and 验证,最终获取全面和准确的参与者和用例信息。该构建过程不仅为构造用例图和类图奠定了基础,而且也使软件工程活动更加规范。

关键词:需求获取;用例驱动;模板;类图

中图分类号:TP39

文献标识码:A

文章编号:1673-629X(2012)03-0153-04

Use Cases Drive Technology Application in Requirements Elicitation Process

LIU Xu-yong^{1,2}

(1. Northwest University, Xi'an 710069, China;

2. Shaanxi Academy of Governance, Xi'an 710068, China)

Abstract: Requirements elicitation is a very important part in the software development process. It runs through the entire software life cycle. According to the system customers and developers existing communication barrier due to understanding differences and perspective of knowledge, put forward a kind of requirements elicitation method based on use case. The method takes the user and task as the center, design a user requirement description template, guiding the user fill requirement description information, developers use the use case driven technique to analyse and verify the information, finally obtain comprehensive and accurate information including actors and use cases. The construction process not only laies a foundation for constructing use case diagrams and class diagrams, but also makes the software engineering activities more normative.

Key words: requirements elicitation; use case driven; template; class diagram

0 引言

在软件系统的开发过程中,最困难和最重要的部分是如何获取用户需求,只要不符合用户的需求,无论软件产品采用多么先进的方法和技术,开发的软件产品都是失败的。依据 leffingwell 的统计,软件项目中 50% 左右的问题都来自于需求的获取和分析阶段^[1]。在传统的结构化软件开发方法中,业务流程或业务数据是需求阶段关注的重点,但是开发出来的产品往往存在维护困难、结构复杂度高、可移植性差等缺点,实践证明必须将两者结合起来统筹考虑才可能开发出用

户满意的软件产品^[2]。虽然面向对象技术解决了结构化方法存在的问题,它将数据和处理过程集合到类中,但由于用户在系统产品的开发过程中参与度较低,使用户想法在最终产品中没有得到充分反映,导致系统无法令用户满意。

文中的软件需求获取方法采用用例驱动技术,从用户的角度出发,通过交互式表格界面,以用户填表的方法,围绕任务层层推进,逐步引导用户表达需求,依据需求信息获取参与者、用例等信息,获取的结果为建立用例图和类图奠定了基础。由于交互方式比较自然和表达方式的图形化,为用户充分参与需求获取活动奠定了良好的基础^[3]。

1 用例驱动分析技术

1992 年,被誉为现代软件开发之父的 Ivar Jacobson 博士将在爱立信公司研发 AXE 交换机时使用的

收稿日期:2011-07-07;修回日期:2011-10-15

基金项目:国家自然科学基金项目(60871097);陕西省科技攻关项目(2009K08-11)

作者简介:刘旭勇(1975-),男,陕西西安人,硕士研究生,工程师,研究方向为软件工程、网络安全等;导师:鱼 滨,教授,研究方向为软件工程、基于 Internet 的软件技术。

技术进行总结并形成了用例驱动技术。目前用例的概念和用例驱动的分析与设计已经成为当今面向对象的系统分析方法不可缺少的部分^[4]。

用例分析技术是以用户的观点为出发点,由用户主动地表达对系统的期望,也为软件需求规格化提供了一个可度量、可验证的基本元素。用例就是为达到一定的目标而与系统之间发生的一段对话,这些对话表示了参与者是如何使用系统的,一个用例通过一组用例实例来定义^[5]。用例不仅是开发人员和用户之间沟通的桥梁,也是项目计划、进度控制、项目测试等环节的基础。

2 基于用例驱动的需求获取过程

软件需求获取是通过与客户、系统用户和其它与系统开发相关的人员彼此交流来获取系统需求的过程。这项工作要求工作人员必须具备应用领域知识、组织知识和特定问题知识。因此,根据开发案例经验,为了获取正确的用例模型,提出如下方法:

2.1 用户功能需求信息的获取

要获得用户的正确需求,必须从用户出发,能够将用户在领域知识方面的认识转换成为软件开发分析人员可以理解的信息,同时能在软件开发中进行正确的分析和编码。

首先需求获取要定义问题范围^[6]。由于技术实现的细节让用户理解有一定难度,造成对系统目标的认识有分歧,所以要明确系统的边界。

其次是对问题的理解。系统中存在着各种类型的多个用户,用户只能从自己工作的角度去理解系统,缺乏或没有能力去了解系统的整体情况,诸如如何提高系统的工作效率、不同用户所面对的系统模块之间有哪些关联、哪些工作可以由软件来完成等等,尽管开发人员和用户之间难免产生沟通的障碍,使得显而易见的信息被忽略,但开发人员必须提供帮助和指导。

最后是确认需求。随着时间的推移,需求的不稳定性将逐步显现,要通过有组织的需求获取活动来克服以上问题。

需求获取、分析和验证的活动是一个相互隔离的、反复和增量的非线性过程^[7]。首先在软件分析人员的引导下,用户以表格的方式填写相关功能需求的信息。其次,分析人员将对客户需求信息并结合相关联的软件需求进行分析,通过系统加工将它们分成不同的类别。再其次,将客户信息编写成文档和示意图。最后,让用户代表对结构化信息进行评审和修改可能存在的错误。这四个过程会贯穿需求分析的整个阶段。

2.2 寻找参与者

参与者是为完成某个任务而与系统进行交互的

人、其他系统、时间和地点等事物,它存在于系统外部,包括主要参与者和辅助参与者两类^[8]。

2.2.1 参与者的定义和分类

参与者分为人、其他系统和时间等三种角色,对于这三种参与者类型可以做不同的分析。

第一种角色是用户,即实际的人,是每个系统中最常用的参与者。由于用户对系统有多种作用,对参与者的命名通常按作用命名而不是按位置命名,避免在每次增加新位置或改变位置时更新模型,以便得到更稳定的参与者图形。

第二种角色是另一个系统。比如超市内的结算系统与银行进行结算时,与银行系统必然存在交互行为,银行系统可以被看成一个参与者^[9]。

第三种常见参与者是时间。时间就是触发某个系统工作的事件。比如防杀病毒系统可能在每日零时进行病毒库升级。由于时间不受用户的控制,因此可确定为一个参与者。

参与者中还有一种是没有实例的抽象参与者。比如某用户可能有行政人员、后勤人员和技术人员三种身份,虽然单位中每个员工都至少有一种身份,但他们都属于员工,它没有实例是个抽象参与者,反映了行政人员、后勤人员和技术人员的一些共性。

2.2.2 获取参与者的方法

获取参与者是一个迭代增量的复杂过程,起初不可能全面或准确地获取参与者,但通过对任务的不断分析和提炼,获取的参与者将逐步清晰并可能不断增加新的参与者^[10]。

一般可以通过以下 3 种途径获取系统参与者:

(1)将域模型中较为重要的域对象定义为系统参与者^[11]。

(2)将商务模型中的人定义为系统参与者。

(3)从用户与专家的沟通或用户的需求信息报告中获取系统参与者^[12]。

寻找参与者不能遵循固定模式,系统分析人员的经验和素质是该项工作成败与否的关键。系统参与者可以依据以下问题确定:

(1)系统数据来源于哪些人或其他系统?

(2)哪些人是系统开发完成后的使用者?

(3)谁是系统的维护和管理人员?

(4)系统数据需要提供给哪些人或其他系统?

要注意的是,在用例建模过程中,系统的组成结构不能抽象为参与者^[3]。比如在计算机硬件系统组成中,计算机硬件系统包括 CPU,将 CPU 抽象成一个独立的参与者是不合适的。

参与者之间也可能存在继承关系,它表示了一个一般性的参与者和其他特殊活动者之间的联系,必须

通过抽象处理调整参与者关系,减少用例图中相关联的通讯的个数,简化用例模型,增强可理解性。

2.3 用户需求用例的获取

依照用户功能需求信息,通过迭代增量过程得到的各种角色的参与者将替代系统用户,也将用户需求的获取转换成参与者用例的获取。用例是参与者使用系统需要达到的目标,它是由系统执行的参与者与系统进行交互的动作序列,其中包括各种备选事件流^[13]。

在用例技术中,“价值”是识别用例的要点,也是用例技术与之前的技术的重要区别^[14]。为保证每个参与者的目标得以实现,构造用例可通过以下步骤:

- (1)确定系统归属的范畴,同时对系统的上下文环境充分了解。
- (2)列举出系统的所有参与者。
- (3)确定参与者所需要完成的目标、输入和输出信息。
- (4)调整参与者的目标,主要用例的名称要依据概要层和任务层的目标来命名。
- (5)确定每个用例的特征,包括用例的范围、前提条件、后置条件等特征。
- (6)确定用例的基本事件流。
- (7)对用例的描述进行复审,将具有共性的描述确定为共用用例。
- (8)重复步骤 2~7 识别出所有用例。

在获取了所有用例后,还需要通过可读性和完整性检验调整用例。获取的用例应具有以下特点:

- (1)用例是实现一个具体的用户目标,并产生一定价值结果;
- (2)用例由系统执行,通过参与者激活并给参与者提供确切的值;
- (3)用例必须通过业务语言和用户观点来描述,并且是参与可见的;
- (4)用例是实现用户目标的完整描述。

根据用例的以上特点,认真分析用户需求信息,通过与用户的不断沟通和交流,充分了解详细的用户需求,最终为每一个参与者确定用例^[13]。

需求阶段获取的用例仅仅反映了参与者需要达到的目标,还没有涉及系统应提供的具体功能细节,必须对用例进行求精,通过对用例事件流的进一步细化,将用例的所有场景所涉及的环节作为功能需求的来源,具体应遵循以下过程:

- (1)在所获取的所有用例中选择一个优先级别最高的用例^[14]。
- (2)场景分析。依据参与者需要达到的目标确定用例的主要场景。然后,通过对主要场景的基本步骤

的分析确定次要场景,依次逐层分解直到得到所有场景。

(3)用例判定。根据场景中下级用例的粒度大小进行判定,将粒度小的用例看做是系统要实现的功能需求,对于粒度大的下级用例应继续当作待求精用例。对待求精用例重复(2)~(3)步获取其功能需求。

重复(2)~(3)步对剩余用例进行求精。当无法继续对所有用例求精时,系统的功能需求就是所有用例的行为集合。

2.4 用户需求描述模板及格式

用例驱动的重点是提供准确的参与者和用户功能需求信息。因此为用户设计了需求描述模板,该模板最大的特点是在以往用例格式的基础上增加了前置条件、后置条件、成功后置条件和失败后置条件,方便用户全面填写它所关心的功能需求,主要是获取用例驱动相关信息。可以往需求获取表格中添加所需内容,同样也无法避免对需求获取信息进行反复的修改,包括对表格中的内容进行修改、删除等操作^[3]。

在获取满意的需求信息后,可以对表格进行保存,以便查询需求信息。结合系统开发的情况,用户需求描述模板的格式以及要素如表 1 所示。

表 1 用户需求描述模板格式及要素

要素名称	简要说明
功能需求描述	用户通过系统做什么,达到什么结果
用例名	每个用例都包括了达到某个目的的名称,名称通常都是动词短语,执行用例的条件必须隐含其中。例如:购买商品
用例描述	每个用例都包含一个此用例的业务需求目标描述和让执行者获取的目标需求,需求的发源地必须在该要素中列举
主要参与者	将与该用例相关的主要参与者进行列举
前置条件	指用例执行前系统所存在的状态或条件。系统对状态或条件必须是可测试的。有些用例中可以没有前置条件
成功后置条件	用例能正常终止并保证交互正常进行时系统所处的状态
失败后置条件	用例不能正常终止并可能影响交互的正常进行时系统所处的状态
关联用例	列举出与该用例的相关用例

将获得的所有信息保存为 XML 文档,为进一步获得用例模型做准备。

2.5 用户需求的用例模型

在获取参与者和用例后,依据用户需求描述经转换得到一个反映用户需求的高层模型——用例模型,尽管它无法反映系统的具体实现细节,但从系统外部用户的角度勾勒出了系统的边界和功能,在此基础上进一步对用例模型不断地修改和完善,检查可能存在的错误和遗漏部分,检查主要从以下几个方面:

- (1)是否有完备的功能需求。用例模型建模工作结束的标志是全面描述系统功能,可以对用户目标的

实现情况进行验证,确保用户目标都被用例实现^[3]。

(2)模型是否容易理解。能够被不同的对象所理解是用例模型的最大优点,因而用例的个数、粒度和模型元素间复杂度都应以可理解为基本原则。

(3)是否存在不一致性。用例模型包括多个组件并由多个工程师合作完成,对模型中多个组件之间存在的矛盾或冲突的地方要及时修正,确保内部一致而保证需求定义的准确。

(4)语义是否存在二义性。不同用户对需求的理解应该是一致的,要确保用例规约的需求含义清晰、准确、无歧义。

3 结束语

在软件开发周期中,早期解决问题往往比把问题留到晚期去解决更为有利^[15],因此使用用例驱动方法,该方法让用户清楚地认识到新系统允许他们做什么,为系统分析和开发人员理解用户的业务和应用领域提供帮助,然后通过用例驱动的过程在后续的分析和设计中进行转化,指导项目开发组的每一个人员协调一致地开展工作,最终开发出让用户满意的软件产品。

参考文献:

- [1] Dean L, Widrig D. Managing Software Requirements—A Unified Approach[M]. [s. l.]: Addison Wesley, 2000.
- [2] Booch G, Rumbaugh J, Jacobson I. Uml 用户指南[M]. 北京:人民邮电出版社, 2006.

(上接第 152 页)

性强等优点^[12,13]。文中所设计的误码测试仪在向虚拟仪器方向发展有很大的扩展空间,可以利用 PC 机的串口与误码仪中单片机的串口进行通信,只要开发出优秀的上位机控制软件,操作人员就能通过 PC 机直接控制误码仪设置并优化各种参数,减轻单片机的任务量。同时控制软件还可以调取测量结果、将测量数据与其他分析软件进行综合分析,以便更好地优化通信系统。

参考文献:

- [1] Newcombe E A, Pasupathy S. Error rate monitoring for digital communications[J]. Proc IEEE, 1982, 70(8): 805-825.
- [2] Fan Y Q, Zilic Z. BER testing of communication interfaces[J]. IEEE Trans on Instrument and Measurement, 2008, 57(5): 897-904.
- [3] 郝云芳. 基于 FPGA 的智能误码测试仪[J]. 西安邮电学院学报, 2006, 11(5): 36-39.
- [4] Clark G, Cain H. Error Correction Coding for Digital Commu-

- [3] 刘旭勇. 用例驱动交互式需求获取技术及支持工具[D]. 西安: 西北大学, 2008.
- [4] 李欣. 基于用例建模的需求获取[J]. 计算机与现代化, 2003, 6(5): 27-29.
- [5] Jacobson I, Booch G, Rumbaugh J. The Unified Software Development Process[M]. 周伯生, 冯学民, 樊东平, 译. 北京: 机械工业出版社, 2002.
- [6] Young R R. 有效需求实践[M]. 韩柯, 耿民, 译. 北京: 机械工业出版社, 2002.
- [7] Rosenberg D, Scott K. 用例驱动 UML 对象建模应用—范例分析[M]. 袁国忠, 管斌, 译. 北京: 人民邮电出版社, 2009.
- [8] 袁红超, 郝克刚, 葛玮. 用例技术的研究和应用[J]. 计算机技术与发展, 2006, 16(2): 96-97.
- [9] 邵天增, 尚冬娟. RGFF 用例构造框架研究[J]. 现代计算机(专业版), 2008(9): 14-17.
- [10] Cockburn A. Structuring Use Case with Goals[J]. Journal of Object-Oriented Programming, 1997, 9(10): 35-40.
- [11] Jacobson I, Pan-Wei Ng. Aspect-Oriented Software Development with Use Cases[M]. [s. l.]: Addison Wesley Professional, 2004.
- [12] 任胜兵, 陈松乔, 喻寿益. 基于事件序列的用例描述形式化及分析[J]. 计算机工程与应用, 2004, 40(23): 61-64.
- [13] 胡树玮, 张修如. 使用用例分析技术捕获需求[J]. 微机发展(现更名: 计算机技术与发展), 2005, 15(7): 4-7.
- [14] 向劲松. 用例驱动方法在软件需求获取方面的研究及应用[D]. 成都: 四川大学, 2005.
- [15] 谈俊峰. 用例分析技术进行需求分析和构架建模[J]. 计算机工程与设计, 2004, 25(2): 252-254.

nications[M]. New York: Plenum Press, 1988.

- [5] 杨飞, 张尔扬. 基于 FPGA 的高速误码仪接收端设计方案的探讨[J]. 电子工程师, 2006, 32(4): 3-4.
- [6] 白驹野, 雷晓平. 单片计算机及其应用[M]. 成都: 电子科技大学出版社, 1997.
- [7] 卢艳萍, 陶成. 基于 FPGA 的误码测试仪[J]. 电子产品世界, 2004(4A): 79-86.
- [8] 樊昌信. 通信原理[M]. 第 5 版. 北京: 国防工业出版社, 2001.
- [9] 李斌, 吴友宇. 基于 FPGA 的误码仪内核设计[J]. 武汉理工大学学报, 2005, 27(6): 12-15.
- [10] 江国强. 数字电子技术[M]. 北京: 电子工业出版社, 2002.
- [11] 卢毅. VHDL 与数字电路设计[M]. 北京: 科学技术出版社, 2001.
- [12] 曹军义, 刘曙光. 虚拟仪器技术的发展与展望[J]. 自动化与仪表, 2003, 18(1): 79-86.
- [13] 敖传宝. 虚拟仪器技术及其发展趋势[J]. 山西电子技术, 2007(6): 79-86.