

第3章 需求获取

需求获取应该是主动的

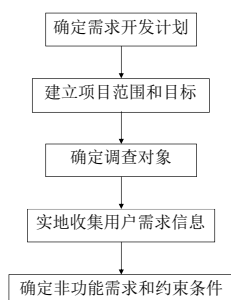
- 需求获取是撒网打鱼，不是休闲钓鱼

比较项	钓鱼模式	网罗模式
心态	被动，愿者上钩	主动，为了生计
目标	没有计划，时间到就结束 ——访谈是走过场	要抓到足够的鱼 ——一定要把计划问题问完
地点	找个环境好的地方 ——对方访谈对象没有预先计划	要找有鱼的地方 ——根据问题选择合适的访谈对象
对象	不管什么鱼，不管大鱼小鱼 ——宏观、细节一把抓	根据要捕的鱼选择不同密度的网 ——根据访谈者类型决定信息层次

2

第3章 需求获取

软件需求获取
(简称需求获取)
阶段的任务简单
的说就是获取用
户的需求信息。
其过程如左图所
示:



3

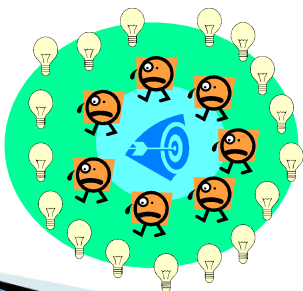
3.2 确定项目的目标和范围

- 此阶段即**需求定义**阶段，基本任务是根据项目目标把项目相关人员定位到一个共同的和明确的方向上，并决定软件系统的**目标**和**范围**
- 严格的说，需求定义不属于需求工程的范畴，而属于立项管理的工作
- 需求定义的产物对于需求工程的工作有着直接的影响

4

需求定义的步骤

- 目标 (Goal) → 问题 (Problem) → 可选方案 (option) → 建议方案 (Answer)



5

可行性分析

- 确定目标：信息系统实现前
- 提出解决方案：分析P，给出O，得出A
- 可行性分析：
 - 效益分析：经济可行性，投资回报
 - 社会可行性
 - 技术可行性

6

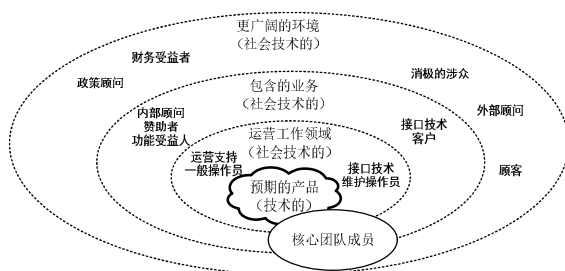
在问题定义上达成共识

- ▶ 把问题写下来，看每个人是否都同意
- ▶ 采用标准化格式：
 - > 问题：描述问题
 - > 影响：确定受问题影响的风险承担人
 - > 结果：确定问题对风险承担人和商业活动的影响
 - > 优点：指出解决方案并列主要优点

理解原因后对问题的陈述

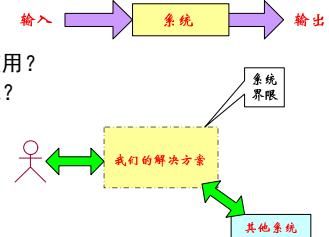
- ▶ 问题：随时提供大学体育赛事的最新报道
- ▶ 影响：移动办公的人群
- ▶ 结果：他们不可能花很多时间来搜索他们感兴趣的新闻，因此无法随时了解到有关他们母校的赛事（或者他们感兴趣的其他大学的体育赛事）。
- ▶ 成功的解决方法：当发生他们感兴趣的新闻时，向他们发出通知，并提供一个地点来为他们提供所请求的新闻。

寻找涉众

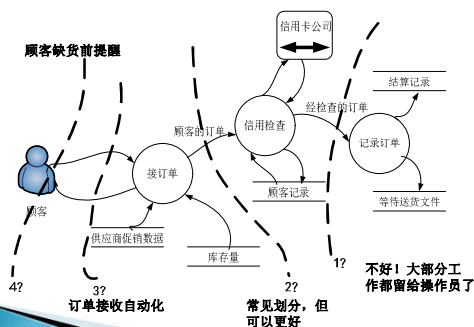


定义系统界限

- ▶ 谁会对系统提供信息？谁会在系统中使用信息？谁会从系统中删除信息？
- ▶ 谁将操作该系统？
- ▶ 谁是系统的维护者？
- ▶ 系统将会在哪儿被使用？
- ▶ 系统从哪儿得到信息？
- ▶ 哪些外部系统要和系统进行交互？



定义解决方案



评估解决方案

- ▶ 经济约束：预算？
- ▶ 行政约束：存在许可问题？潜在内外部政问题？部门间问题？
- ▶ 技术约束：技术选择有何限制？限制在已有平台或技术上？禁止使用新技术？需要购买软件包？
- ▶ 系统约束：建立在现有系统上？需要维护与原系统的兼容性？必须支付什么操作系统？
- ▶ 环境约束：合法吗？安全性要求？其他标准限制？
- ▶ 进度及资源：进度要求？已有资源？外部劳动力可用否？有无扩展资源？

需求定义阶段产物

- ▶ POS项目综述
 - 目标、相关人员和用户、限制条件、关键术语、相关事实与假设、工作范围、费用计划、风险、可行性
- ▶ Vision愿景
 - 业务需求
 - 背景、机遇、目标、客户需求、提供给客户的价值、业务风险
 - 解决方案
 - 愿景说明、主要特征、假设依赖
 - 范围和局限性
 - 首次发行范围、随后发行范围、局限性
 - 业务环节
 - 客户概貌、项目优先级、操作环节
 - 产品成功因素

目标

- ▶ SMART原则
 - 必须是具体的Specific
 - 必须是可度量的Measurable
 - 必须是可达到的Attainable
 - 必须和其他目标具有相关性Relevant
 - 必须具有明确的截止期限Time-bound

项目	内容
目标	对贷款客户的信用风险实现量化管理
业务优势	降低不良贷款率
度量指标	不良贷款率下降5%
合理性	降低不良贷款率带来的效益与构建系统花费成本相比是否有价值
可行性	对信用风险实现量化管理能降低不良贷款率吗？
可达性	该目标能达到吗？

3.1 确定需求开发计划

确定需求开发计划的基本任务是确定需求开发的实施步骤，并给出收集需求活动的具体安排和进度。

需求开发计划需要注意以下几点：

- (1) 只考虑与需求开发相关的工作；
- (2) 应考虑困难性和灵活性；
- (3) 应考虑书写和整理需求规格说明及其文档所花费的时间。

3.3 确定调查对象

本阶段的基本任务是明确地确定来自不同层次的需求来源和用户，并将其分类。

应根据需求的层次来区分不同的用户：

- (1) 提出目标需求的用户；
- (2) 提出业务需求和功能需求的用户；
- (3) 软件开发人员，主要是指系统分析员。

3.3 确定调查对象

- ▶ 软件系统面临的用户众多，而这些用户由于所在的部门、职责和掌握的知识不同而存在差异，为了避免忽视和遗漏某些用户的情况，可以根据用户的某些方面将用户分类。
- ▶ 在将用户分类后，在分类的基础上进一步寻找每类用户的代表或联络人，这些人代表了一个特定的用户类，并可充当该用户类与开发人员之间的“窗口”。
- ▶ 这些人也必须是真正的用户，而不是单纯的代理人。

3.3 确定调查对象

表3.1 用户代表的义务

- 1) 给分析人员讲解业务及说明业务方面的术语等专业问题。
- 2) 抽出时间清楚地说明需求并不断完善。
- 3) 当说明系统需求时，力求准确详细。
- 4) 需要时及时对需求做出决策。
- 5) 要尊重开发人员的成本估算和对需求的可行性分析。
- 6) 对单项需求、系统特性或用例划分优先级。
- 7) 评审需求文档和原型。
- 8) 一旦知道要对项目需求进行变更，要马上与开发人员联系。
- 9) 在要求需求变更时，应遵照开发组织确定的工作过程来处理。
- 10) 尊重需求工程中开发人员采用的流程（过程）。

3.3确定调查对象

软件需求可来自与各个方面，而且用户类也不一定都是指人。有时也可以把其它应用系统或计算机硬件设备和接口等视为附加的用户类成员，这样就可确定软件系统与哪些外部应用系统或计算机硬件相关的需求。这就是说需求信息来源除了来自用户类外，还可来自于其他方面。

25

3.3确定调查对象

几个典型的软件需求来源：

1. 直接和间接使用软件系统的用户；
2. 系统需求规格说明；
3. 市场调查和用户问卷调查；
4. 已开发出的和待开发的同类软件系统的描述和文档；
5. 对人工系统的存在问题的报告和增强要求；
6. 观察正在工作的用户；
7. 用户工作内容的分析。

26

3.3确定调查对象

当确定了用户类及明确了用户需求的主要来源后，这样就可从不同的渠道和不同的人那里收集到大量的需求信息。但这些需求信息既包含了明确的用户需求，也包含了一些不一致和含糊的需求，而且软件开发人员也难以解决。因此，这就需要寻找需求的**决策者**。

在处理有问题的需求信息时，决策者并不是固定不变的，而是根据实际中可能发生的具体问题来确定。

27

3.4实地收集需求信息

在确定了需求的来源和调查对象后，下一步就是实地收集需求信息。实地收集需求信息阶段的任务就是到现场实地调查和与用户交流，收集和用户需求信息。

28

用户需求的冰山模型

- ▶ 意识到的需求
 - 困扰用户的问题、用户自己能够设想到的功能
- ▶ 无意识的需求
 - 用户的实际工作场景
- ▶ 未梦想的需求
 - 用户无法构想出对其工作产生革新性变化的解决方案



29

3.4实地收集需求信息

实地收集需求信息可能面临的困难：

1. 能提出软件需求的用户可能觉得他们没有充分的时间与开发人员进行交流和讨论；
2. 有时用户希望通过简单的方法和说明，或者通过简单回答开发人员的询问后，软件开发人员就能清楚地理解他们的需求，而不需要花费太多的时间进行讨论；

30

3.4实地收集需求信息

3. 用户和开发人员都只考虑自己的利益；如：有些用户由于缺乏使用计算机的经验，导致产生畏难情绪；有些用户认为开发软件系统自己的关系不大，对待需求信息的收集工作采取消极的态度。
4. 用户本身不能提出明确的需求；
5. 开发人员缺乏用户的业务知识，而用户也缺乏计算机方面的知识，导致双方在交流中产生许多的困难，以至收集工作难以进行。

31

阻碍需求获取的心理现象

- ▶ 言过其实心理
 - 通过比较用户代表的表述来识别言过其实，利用差异展现、瓶颈分析法来缓解影响
- ▶ 越俎代庖心理
 - 识别正确的被访谈者
- ▶ 非正式心理
 - 做好访谈计划，避免在办公室访谈
- ▶ 抗拒心理
 - 倾听对方，化敌为友
- ▶ 推卸责任心理
 - 让被访谈者介绍工作场景

32

3.4实地收集需求信息

实地调查的步骤：

1. 向掌握“全局”的负责人调查；
2. 向部门负责人调查；
3. 向业务人员调查。

步骤（2）和步骤（3）是一个反复的过程，而且每次调查之前要制定调查提纲，每次调查要作记录，并交由用户审查核实，以保证需求信息的可靠和准确。

33

3.4实地收集需求信息

实地收集需求信息的方式

1. 用户访谈
2. 问卷调查
3. 文档研究
4. 场景串联
5. 现场观摩
6. 联合开发

34

3.4实地收集需求信息

需求信息可大致分类如下：

1. 目标需求；
2. 用例说明；
3. 业务规则；
4. 功能需求；
5. 非功能需求；
6. 外部接口需求；
7. 限制
8. 数据定义；
9. 解决思想。

35

用户访谈

- ▶ 优点：直接有效、形式灵活、交流深入的宽带通信形式（文字、声音、图像）
- ▶ 缺点：占用时间长、信息存在片面性
- ▶ 用户访谈类型
 - 高层管理人员：探讨系统目标与范围
 - 中层管理人员：理清需求脉络
 - 操作层：填充需求细节
 - 技术团队：讨论解决方案的可行性
- ▶ 参考要点：
 - 时间不超过1小时
 - 注意记录
 - 有适当的计划

36

用户访谈

时间计划

阶段	任务	时间	备注
开场白	陈述预先的理解	5-15分钟	明确访谈的主题、范围和层次
预先计划问题	寻求问题的答案	25-30分钟	主体工作，必须完成
即兴问题	扩大需求信息量	20-30分钟	不宜过度扩展
总结	总结访谈内容	5-10分钟	访谈者向被访谈者陈述

用户特点

- 思维比较发散：先保证不遗漏计划问题，再探讨即兴问题
- 思维比较聚焦：适当穿插即兴问题

记录方式

- 自己做笔记（分神）、专门一同事做笔记（成本高）、录音（失去身体语言）/录像（难操作）

37

用户访谈

制作访谈问卷并且事先发给被访谈人

- 两天之前，一周以内

把握语言节奏

→句式简单，问题简单

→问问题，听取回答，然后反馈理解

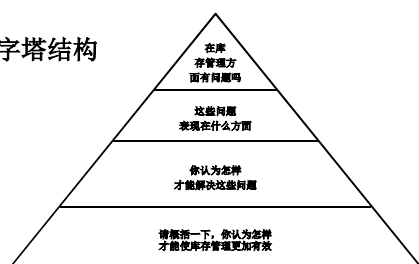
组合不同的问题类型

比较项	封闭式	半封闭式	开放式
信息收集有效性	最弱	较弱	最强
被访谈者回答难度	最简单	较简单	较难
对被访谈者诱导性	较容易	容易	不会
访谈所需时间	节省	节省	可能会浪费
信息的广度和尝试	较窄	较窄	较广

38

用户访谈

金字塔结构

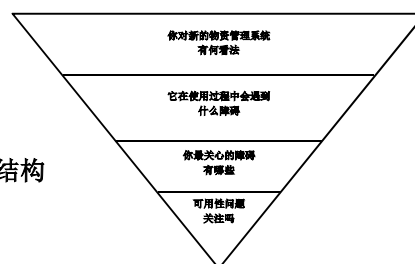


- 一种归纳过程，由封闭式问题开始，逐渐使用开放式问题

39

用户访谈

漏斗结构



- 一种演绎过程，由开放式、通用性问题开始，逐渐使用封闭式问题，缩小可能答复。

40

用户访谈

注意沟通细节

- 使用模型帮助沟通
 - 及时用草图绘制模型，获取用户反馈
- 避免干扰性暗示
 - “我得时间比你的宝贵”
 - “我听不懂你在说什么”
- 消除术语影响
 - 勇于询问术语，不要不懂装懂
 - 术语可能存在同义词

41

用户访谈

善于观察异常

现象	内涵
语速极快、描述过程逻辑性强	1. 传达的信息是用户很熟悉的工作 2. 描述的信息可能是理想化的制度
语速缓慢、中间有很多的停顿	1. 用户对讲述的信息不够熟悉 2. 传达的信息是平时考虑不多的 3. 这些信息很容易产生变化，需要注意
突然改变语气、脸色	1. 所问的问题是对方很敏感的 2. 涉及到公司政治、有利益影响 3. 传递的信息有明显的误解
回答某问题时眼睛经常向左上方转动	对方很可能在回忆，信息相对真实
回答某问题时眼睛经常向右上方转动	对方很可能在“创造”，真实性待确认
坐姿很随意，诸如翘着二郎腿	对访谈内容重视度不高
回答比较强势，甚至有攻击性	对信息系统的建设有反对意见或敌意
身体或腿部晃动十分厉害	对方情绪较不安

不要遗漏问题

- 不要遗漏计划问题
- 使用“你认为我还应该了解什么问题？”等问题

42

用户访谈计划

- 时间、人员
 - 采用模板
 - 访谈主题、访谈要点、期望部门、备注、访谈人、访谈时间

协同办公系统项目调研计划

根据项目进度项目组将在 11 月 8 号进行信息发发布的需求调研，需求调研工作大概需 10 天，请各单位协助做好准备工作以及积极配合与需求调研，下面将说明具体的时间安排：

时间	调研内容	调研单位
第一天	关于项目、前期中心、公文文件、	行政管理部
第二天	流程审批系统、行政处理子系统、	行政管理部
第三天	文件处理、	行政管理部
第四天	培训管理模块、绩效管理、	人力资源部
第五天	人事管理子系统、	人力资源部
第六天	财务系统、	财务部
第七天	QHSE 系统、	QHSE 部
第八天	QHSE 管理子系统、	QHSE 部
第九天	合同采购控制管理系统、	商务部、计划控制部
第十天	888	公司信息办公室

需要各部门提前准备：

1. 相关模块流程、表格等相关纸质文件。
2. 提前梳理各相关模块流程，可描述清楚即可。

43

用户访谈计划

访谈内容

访谈对象	计划要点	备注
高层管理人员	罗列部分问题/机会点 准备相关系统的实施经验案例 列举一些潜在的解决方案	确认已列出的问题/机会点，探讨潜在的问题/机会点，标识出关键 Stakeholder
中层管理人员	罗列相关业务事件列表 收集与特定业务事件相关的资料 准备一些业务事件的关键问题点 准备一些相关的管理控制点	确定每个业务事件的流程、相关实体、相关参与者、明确管理控制点
操作层	罗列出相关的业务活动 针对业务活动的问题点、需求点 罗列出相关的业务实体、说明	从基本情况、功能、数据、非功能、用户环境、开放问题多角度设计问题
技术人员	罗列出要解决的问题 整理出预想的解决方案 标识关键疑点	针对每个解决方案逐一分析，结合针对问题探讨方案的合理性和可行性

44

用户访谈计划示例：

1. 访谈内容概述
 - (1) 业务事件：涉税业务受理
 - (2) 了解涉税业务受理的整个过程，用户特点，输入、输出与处理要点，数据量等。
 - (3)
2. 访谈对象
 - 职位：涉税窗口工作人员
 - 姓名：_____联系方式：_____
 - 其他相关信息
3. 访谈计划
 - 基本情况
 - 每个窗口有多少工作人员，主要负责哪些事情？
 - 请简述一下你们的工作过程。（场景）
 - 用户环境
 - 你们的电脑配置如何？一般还使用哪些应用程序？
 -

45

用户调查

- 优点：调查面比较广，能够获得更多人的反馈，是用户访谈技术的有效补充，能够克服用户访谈的片面性。
- 缺点：不易深入
- 使用方式：
 - 先调查，后访谈：适合市场调查
 - 先访谈，后调查：适合需求获取
- 使用时机
 - 存在大样本用户
 - 存在跨地域用户

46

用户调查问卷示例

- 姓名：_____ 部门：_____
- 职位：_____ 联系电话：_____
1. 目前业务受理过程效率存在问题吗？
 - A. 是 B. 否
2. 你通常一天会处理多少笔业务受理？
 - A. 20 笔以下 B. 20 至 30 笔 C. 40 至 60 笔 D. 60 笔以上
3. 系统出现故障时，你希望采用以下哪种措施？
 - A. 暂停受理，待系统恢复 B. 系统提供应急措施
 - 如果选 B 请提出建议：_____
4. 对于本系统，最希望实现哪些功能或者满足哪些要求？
 - (1) _____
 - (2) _____
 - (3) _____

47

用户调查要点

- 设计要点
 - 合适的篇幅与布局
 - 由易到难
 - 逻辑相关性
 - 控制在 1~3 页内
 - 问题类型的选择
 - 封闭性问题→半封闭性问题
 - 开放性问题：跟随策略+简短
 - 避免 C 现象、D 现象
- 分析要点
 - 剔除无效问卷
 - 对问卷填写人进行分类

48

文档研究

- 通过研究用户文档，捕获需求，特别是数据项和数据流信息
- 优点：
 - 详细、直观地对数据流细节进行了解与分析
- 缺点：
 - 由于文档量通常很大，容易陷入文山书海
- 要点：
 - 注意文档历史性
 - 化被动收集为主动索取

49

情节串联

- 借助原型来加速需求捕获的过程
- 优点
 - 用户友好、交互性强，让用户能够早期参与
- 缺点
 - 费时费力
- 使用要点
 - 以“情节”为线索
 - 强调“串联”本质



50

现场观摩

- 到业务现场观摩以建立更加感性的认识
- 优点：
 - 能够对需求与业务流程建立直观的认识
- 缺点：
 - 消耗时间长、“观摩”容易导致失真
- 要点：
 - 避免失真
 - 避免走马观花
 - 尽量建立可重复观摩场景

51

联合开发

- 用户代表、需求分析人员、开发人员代表齐聚一堂，以头脑风暴的方式进行需求探讨
- 优点：
 - 最理想的需求捕获方式，是击破盲点的关键手段
 - 用于项目启动初期和关键主题的探讨
- 缺点：
 - 使用成本过高
- 要点：
 - 会前有准备
 - 在理念上建立共识
 - 确保真正的Stakeholder参与
 - 会中有控制

52

联合开发

- 中间休息 迟到**

规则：每人先发一张，允许迟到一次而后向罚款箱投入xx元

目的：保持会议的动向

1次自由 廉价攻击

规则：每人先发一张，允许自由攻击或批评任何个人或部门，而后再出现就向罚款箱投入xx元

目的：逗趣并提醒人们注意发言

5分钟发言

规则：每人可以在任何时候使用，让出讲台给他并计时，不得打断

目的：允许结构化的即席发言

好主意！

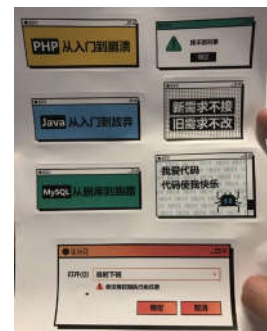
规则：每人先发两张，送给提出好建议的人，要尽快给出

目的：刺激与会者并奖励提出创新者
- 会后有总结

53

可变需求的获取

- 变更可能是需求文档的重要组成部分之一
- 捕获需求时不要忽视对变更可能性的理解
 - 流程变化
 - 业务规则变化
 - 用户界面变化



54

需求协商

- ▶ 在需求捕获的过程中，经常遇到需要协商的地方，协商的结果取决于需求人员的沟通技能。
 - 揭开解决方案后面的问题
 - 共赢性谈判
 - 转换技巧
 - 将未知问题转换为已知问题
 - 相对重要转换为相对次要
 - 关注点转移
 - 隐喻

55

使用场景技术的需求获取

- ▶ 场景的定义
 - 所谓场景是指用户与软件系统实现某个目标而进行交互活动过程的描述。
- ▶ 场景的构成
 1. 执行者（用户）
 2. 进入场景前系统状态的描述
 3. 执行者的目的
 4. 动作和事件系列（包括正常或非正常事件）

56

使用场景技术的需求获取

- ▶ 例：关于切断P C机电源的场景

王某是使用装有Windows98系统的P C机的用户，并且已有一年的经验。他几乎每天使用P C机向朋友等发电子邮件。今天在发送了4封电子邮件后想切断P C机电源。

王某首先按下“屏幕中的开始”按钮，并在显示出来的菜单中选择“关闭计算机”选项。在屏幕中央出现了与关闭计算机相关的对话框，以询问用户是否真正关闭计算机。王某确认并按下了“关闭计算机”的按钮。计算机在使屏幕变黑后，自动切断P C机的电源。

57

使用场景技术的需求获取

- ▶ **执行者（用户）**：王某；
- ▶ **进入场景前系统状态的描述**：使用P C机的经验是1年。几乎每天使用。另外，今日发送电子邮件的工作已结束；
- ▶ **执行者的目的**：退出Windows，并切断P C机的电源；
- ▶ **动作和事件系列**：第2段文字，从按下“开始”按钮的动作开始，到切断P C机电源的事件完成为止。

58

使用场景技术的需求获取

- ▶ 场景应具有的特征
 1. 场景代表某些用户可见的功能，实现一个具体的系统需求；
 2. 场景总是被参与者启动的，并向参与者提供可识别的信息；
 3. 场景必须是完整的。

59

使用场景技术的需求获取

- ▶ 场景的表示

非形式化的表示	形式化的表示
自然语言	状态图
结构化语言	流程图
图形	时序图
动画片等	代数描述图等

60

使用场景技术的需求获取

场景的种类

1. 按执行者的目标能否实现
正常场景和失败场景；
2. 按场景描述的内容
正向场景和逆向场景；
3. 场景之间亦可以建立关系以及精化处理。

61

使用场景技术的需求获取

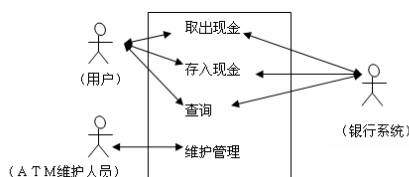
使用用例的需求获取

用例通常用于描述可发生的**所有事件序列**，而场景则是描述其中的一部分。因此，用例也可以说是场景的集合，一个场景是用例的实例。

62

使用场景技术的需求获取

例：自动取款机用例模型



63

使用场景技术的需求获取

例：取出现金用例

正常情况：

- 在显示屏上显示问候信息；
- 顾客将磁卡插入ATM；
- ATM读出磁卡上的代码，并检索该卡能否使用；
- 如果磁卡能使用，ATM要求顾客输入密码。
- 等待密码输入；
- 顾客输入密码；
- 如果密码正确，ATM请求顾客选择事务处理类型。
- 等待输入事务类型；
- 顾客选择取现金事务，并输入取出的数量；
- ATM做好取现金的准备，同时根据指定的银行帐号向银行系统发送请求信息。
- ATM吐出相应的纸币；
- ATM向顾客返还磁卡。
- ATM打印并输出收付款说明书。

64

使用场景技术的需求获取

场景技术的特点

1. 把软件系统的需求信息**文本化**，有助于在实现软件系统前明确用户与软件系统的**相互作用**；
2. 可以把当前系统**存在的问题**作为实例并记录下来；
3. 可以成为项目相关人员间的**共同语言**；
4. 由于场景描述了软件系统的操作，比较具体，其**易理解性**较好；
5. 通过场景使得提出和获得**需求**的**双方**之间能建立起相应的理解。

65

使用场景技术的需求获取

需要注意的问题

1. 场景的**数量**，即一个软件项目应该写多少个场景没有一个限制标准，主要视项目的规模和复杂性而定。但如果场景数量过大，往往也易加大分析和理解的难度。
2. 场景的**冗余**问题。应尽量避免场景描述的内容发生重叠，可根据实际情况合并和去掉一些内容重叠的场景。
3. 应防止场景描述内容的**冗长**。

66

3.5 确定非功能需求

- ▶ 非功能需求是衡量软件能良好运行的**定性指标**。由于缺乏**定量指标**，因此很难根据这些需求来评价软件系统，这也是开发出来的软件系统与用户所满足的软件系统之间存在差异的主要原因。
- ▶ Robert Charette:
 - 真正的现实系统中，在决定系统的成功或失败的因素中，满足非功能需求往往比满足功能需求更为重要。

67

对用户重要的属性

- ▶ 有效性Availability
- ▶ 效率Efficiency
- ▶ 灵活性Flexibility
- ▶ 完整性Integrity
- ▶ 互操作性Interoperability
- ▶ 可靠性Reliability
- ▶ 健壮性Robust
- ▶ 可用性Usability

对开发者重要的属性

- ▶ 可维护性maintainability
- ▶ 可移植性Portability
- ▶ 可重用性Reusability
- ▶ 可测试性Testability

属性的取舍

	Availability	Efficiency	Flexibility	Integrity	Interoperability	Maintainability	Portability	Reliability	Robustness	Testability	Usability
Availability	+	-	-	-	-	+	+	+	+	+	+
Efficiency	-	+	-	-	-	-	-	-	-	-	-
Flexibility	-	-	+	+	+	+	+	+	+	+	+
Integrity	-	-	-	+	+	+	+	+	+	+	+
Interoperability	-	+	-	-	+	+	+	+	+	+	+
Maintainability	+	-	+	-	-	+	+	+	+	+	+
Portability	-	+	-	-	-	+	+	+	+	+	+
Reliability	+	-	+	+	+	+	+	+	+	+	+
Robustness	+	-	+	+	+	+	+	+	+	+	+
Testability	+	-	+	+	+	+	+	+	+	+	+
Usability	-	-	-	-	-	-	-	-	-	-	+

选择的非功能需求属性之间的正负关系

应用举例

- 1、如果软件要在多平台下运行（可重用性），那么就不要对可用性抱有乐观态度
- 2、可重用软件能普遍适用于多种环境中，因此，不能达到特定的容错（可靠性）或完整性目标。
- 3、对于高安全性的系统，很难完全测试其完整性需求；可重用的类组件或其它应用程序的互操作可能会破坏其安全机制。

3.5 确定非功能需求

在收集非功能需求信息时常用的方法：

1. 将不同用户类代表提出的可能很重要的非功能需求进行综合，并根据其中的每个需求设计出许多方法，然后根据用户的回答，使这些需求更明确化；
2. 开发人员与用户一起对每一个非功能需求制定**可测试和可验证**的具体标准；
3. 设计与非功能需求相冲突的假设示例，利用**反例**来提示用户。

72

3.6 在收集需求信息中应注意的问题

1. 应能适当的调整收集范围；
2. 尽量把用户所持的假设解释清楚，特别是发生冲突的部分；
3. 尽量理解用户用于表达他们需求的思维过程，特别是尽量熟悉和掌握用户具有的一些专业知识和术语；
4. 在收集需求信息时，应尽量避免受不熟悉细节的影响；
5. 应尽量避免讨论一些具体的解决方案；
6. 需求信息收集工作的结束

73

需求获取总结

- ▶ 主动性
 - 尽可能多、有方向
- ▶ 计划性
 - 时间、人员、内容
- ▶ 科学性
 - 从宏观到细节
 - 善于使用不同捕获技术

74