需求复习整理

*• 案例+思考题*

*– 7~9*

*• 分值分布*

* *需求获取及前期分析:第2、5~9章 40~50分*
* *需求分析:第5、11、12、13、14章 40~50分*
* *其他(15~17章):10~20分*

第2章 需求基础

*• 名词解释*

* *基本概念*

*• 案例*

*– 判定需求的类型,发现错误并修正(示例:ppt 思考题)*

*– 书写需求(示例:第2章案例题4~6)*

1. 名词解释（基本概念）
2. IEEE是怎样定义需求的？从中你可以得到什么认识？
   1. IEEE对需求的定义：
      1. 用户为了解决问题或达到某些目标所需要的条件或能力
      2. 系统或系统部件为了满足合同、标准、规范或其他正式文档所规定的要求而需要具备的条件或能力
      3. 对(i)或(ii)中的一个条件或一种能力的一种文档化表述
   2. 认识：

为了融合不同群体的看法，IEEE的定义当中同时包括了用户的观点（第一种条件和能力）和开发者的观点（第二种条件和能力），但是即便如此，不同群体的人们也很难就IEEE的定义进行一直和准确的解读，因为需求概念的内涵和外延都非常丰富。

1. 解释下列名词：问题域、解系统和共享现象，并结合它们的含义说明软件系统是如何与现实世界形成互动的？
   1. 问题域：

当现实的状况与人们期望的状况产生差距时，就产生了问题。要解决问题，就需要改变现实当中某些实体的状态或改变实体状态变化的演进顺序，使其达到期望的状态或演进顺序。这些实体和状态构成了问题解决的基本范围，称为该问题的问题域

* 1. 解系统：

软件系统通过影响问题域，能够帮助人们解决问题，称为解系统。

* 1. 共享现象：

软件系统能够与问题域进行交互和相互影响的原因在于，软件系统中的某些部分对问题域中的某些部分的具有模拟特性。换句话说，软件系统当中含有问题域某些部分的模型（或模拟），常见的模型包括数据模型、对象模型、处理模型等。问题域中的某些信息能够和模型中的信息建立映射关系。这些通过映射建立的共同知识，就是问题域和解系统之间的共享现象。

共享现象是问题域和解系统实现交互和互相影响的途径与接口，问题域和解系统都通过改变这些共同知识来影响对方，或者通过认同这些共同知识的改变来接受对方的影响。

1. 解释下列名词：需求、规格说明、问题域特性和约束，并结合它们的含义说明需求工程的主要任务是什么？
   1. 需求：

需求是用户对问题域当中的实体状态或事件的期望描述。

\*直接需求（和解系统直接相关）是可以通过更改共享现象被满足的需求;

\*间接需求（和解系统间接相关）是需要修改共享现象，同时连锁影响问题域才能满足的需求。

* 1. 规格说明：

规格说明是解系统为满足用户需求而提供的解决方案，规定了解系统的行为特征。

\*解决方案只能通过改变共享知识，影响问题域的运行，进而满足用户的需求，所以规格说明主要包括两个部分：对共享现象（模型）的描述；系统对共享现象所施加的操作的描述。

\*\*\*需求关注的是现实世界中的部分，软件关注的是解系统，而规格说明关注的是共享现象

* 1. 问题域特性：

在和解系统互相影响的同时，问题域是自治的，它有自己的运行规律，而且这些规律不会因解系统的引入而发生改变。这种自治的规律性称为问题域特性，包括结构特性和行为特性。

\*需要关注的问题域特性：间接特性、约束和假设、社会性因素

* 1. 约束：

问题域当中有些特性完全不受共享现象的影响，即完全不受解系统的影响，同时却可能很大程度上影响共享现象，影响解系统，甚至关乎解系统的成败。这些特性被认为是解系统对环境的依赖特性。当这些特性非常明确时，称之为约束；不明确时，需要限定特性的变化范围，称之为假设。

* 1. 需求工程的主要任务：
     1. 进行需求开发，确定用户的期望效果R
     2. 研究问题背景，描述问题域特性E
     3. 构建解系统，描述解系统行为S，使得E和S的联合作用效果符合需求R，即：E,S |-> R

1. 需求有哪些常见的类别？
   1. 需求的分类1（IEEE）：
      1. 功能需求（必备,价值的来源）：和系统主要工作相关的需求，即在不考虑物理约束的情况下，用户希望系统所能够执行的活动，这些活动可以帮助用户完成任务。功能需求主要表现为系统和环境之间的行为交互。
      2. 性能需求（动态性需要专门模拟）：系统整体或系统组成部分应该拥有的性能特征，例如CPU使用率、内存使用率等。

\*速度(Speed),系统的响应时间。

*e.g.PR1:所有的用户查询都必须在10秒内完成。*

\*容量(Capacity),系统所能存储的数据量。

*e.g.PR2:系统应该能够存储至少10万条销售记录。*

\*吞吐量(Throughput),系统在连续的时间内完成的事务数量

*e.g.PR3:解释器每分钟应该至少解析5000条没有错误的语句。*

\*负载(Load),系统可以承载的并发工作量。

*e.g.PR4:系统应该允许200个用户同时进行正常的工作。*

\*实时性(Time-Critical),严格的实时要求。

*e.g.PR5:监测到病人异常后,监控器必须在0.5秒内发出警报。*

* + 1. 质量属性（从设计［尤其是体系结构设计］的角度来讲）：系统完成工作的质量，即系统需要在一个“好的程度”上实现功能需求，例如可靠性程度、可维护性程度等。

\*可靠性(Reliability):在规格时间间隔内和规定条件下,系统或部件执行所要求能力的能力。

*e.g.QA1:在进行数据的下载和上传中,如果网络故障 ,系统不能出现故障。*

\*可用性(Availability):软件系统在投入使用时可操作和可访问的程度或能实现其指定系统功能的概率

*e.g. QA2:系统的可用性要达到98%。*

\*安全性(Security):软件阻止对其程序和数据进行未授权访 问的能力,未授权的访问可能是有意,也可能是无意的

*e.g. QA3:VIP顾客只能查看自己的个人信息和购买记录;*

\*可维护性(Maintainability):软件系统或部件能修改以排除故障、改进性能或其他属性或适应变更了的环境的容易程度 ,包括可修改性(Modifiability)和可扩展性(Extensibility)

*e.g. QA4:如果系统要增加新的特价类型,要能够在2个人月内完成。*

\*易用性(Usability):与用户使用软件所花费的努力及其对 使用的评价相关的特性。

*e.g. QA5:使用系统1个月的收银员进行销售处理的效率要达到10件商品/ 分钟*

* + 1. 对外接口（从封装和信息隐藏的角度讲）：系统和环境中其他系统之间需要建立的接口，包括硬件接口、软件接口、数据库接口等等。
    2. 约束（容易遗漏）：进行系统构造时需要遵守的约束，例如编程语言、硬件设施等。
  1. 需求的分类2:
     1. 硬件需求
     2. 软件需求
     3. 其他需求：软件、硬件、人力、基础设施等相关因素之间协同与互动的需求

1. 描述业务需求、用户需求和系统（级）需求的区别与联系。
   1. 业务需求：

抽象层次最高的需求。是系统建立的战略出发点，表现为高层次的目标，它描述了组织为什么要开发系统。

*e.g. BR1：实现车辆的统一管理和有效使用。*

为了满足用户的业务需求，需求工程师需要描述系统高层次的解决方案（建立共同前景），定义系统应该具备的特性（限定系统范围）。

*e.g. SS1（解决方案）：实现一个申请子系统，实现一个调度子系统，让二者互相配合实现统一管理*

*SF1（系统特性）：（i）工作人员需要用车时需要提出申请（ii）安排专人对所有的车辆申请进行统一的调度与安排*

* 1. 用户需求（用户角度）：

执行实际工作的用户对系统所能完成的具体任务的期望，描述了系统能够助用户做些什么。

*e.g. UR1：在需要使用车辆时，用户需要填写一个车辆使用申请单，然后等待申请单的反馈信息，并根据反馈信息使用车辆。*

对所有的用户需求，都应该有充分的问题域知识作为背景支持。

*e.g. 对UR1需要补充问题域知识PD1：申请单的内容包括：申请人、对车辆的要求、预计的用途……*

* 1. 系统需求（用户需求进一步明确和细化，是用户与系统的交互）：

用户对系统行为的期望，一系列的系统行为联系在一起可以帮助用户完成任务，达成用户需求，进而满足业务需求；系统需求可以直接映射为系统行为，定义了系统中需要实现的功能，描述了开发人员需要实现什么。

*e.g. UR2：对于由其他工具所创建的外部文件，软件应该提供一种表现和访问的方法*

*SR1：（i）用户应该能够定义外部文件的类型。（ii）每一种外部文件类型都有一个与之相关联的可对其进行处理的工具*

\*\*\*软件解决方案：

问题->目标->用户需求->系统级需求

背景->系统特性->问题域知识->分析模型

1. 优秀的需求哪些特性？试为每一个特性都举出一个不符合的示例。

优秀的需求特性：

* 1. 完整性：不需要做更多的扩展就可以充分的说明用户所需要的系统功能。每一个需求的描述都应该包含开发人员设计和实现这项功能需要的所有信息

*e.g.R6（不完整）：系统应该允许被扩展*

*e.g.R7（完整、较R6精确）:系统的调度算法应该允许被扩展*

* 1. 正确性：真实的反映用户的意图；必须请需求的提出者予以确认
  2. 3) 精确性：描述仅包含必要的信息；简洁、清晰

*e.g.R8（不精确）：在实现之后，系统的调度算法应该允许被扩展。*

* 1. 可行性：由开发人员进行检查需要进行一定的分析和研究，而不是单纯的凭借经验和直觉,必要的时候要通过开发原型来加以验证

*e.g. R9:保证系统核心功能可以7×24小时连续运行*

* 1. 必要性：满足用户的业务需求所必需的
  2. 无歧义：每一项需求都应该有而且只能有一种解释,定义一个可以共同理解的词汇表（Glossary）
  3. 可验证：通过分析、检查、模拟或者测试等方法能够判断需求是否被满足

e.g.R10(不可验证):实现各部门的公文流转无纸化、文档一体化、业务管理的规范化、自动化和网络化；实现工作流程合理化、高效化，决策支持科学化、准确化；统一办公流程、规范公文格式，加强信息交流和共享，提高工作效率。

\*让需求具体化、小心形容词和副词的使用、避免程度词的使用

1. 案例

* 判定需求的类型,发现错误并修正(示例:ppt 思考题)
* 书写需求(示例:第2章案例题4~6)

1. 案例4:设想你自己就是ATM机的唯一用户，写出你对ATM机系统的用户需求

UR1：使用户可以根据系统的明确操作提示做出正确的反应；

UR2：用户插入银行卡后需要输入密码，得到验证后才可进行有效的具体操作；

UR3：在用户进入系统后，可以选择使用查询金额、存取现金、转账的功能；

UR4：用户能够正确、安全地退出系统。

1. 案例5:对上一题，再设想你是需求工程师，尝试将用户需求转换为系统（级）需求

SR1：

（1） 系统显示用户插入磁卡的动态图像，正确标明插卡位置；

（2）用户根据提示，正确插入磁卡；

（3）系统读取磁卡卡号，界面显示输入密码的提示；

SR2：

（1）对用户输入的密码，系统自动进行字符匹配；

（2）匹配正确的话，进入具体操作界面；

（3）匹配不正确的话，警告密码不正确，并提示再次输入；

SR3：

（1）若用户选择查询金额图标和查询金额币种，系统读取银行数据库中用户对应的信息，反馈在用户界面上；

（2）若用户选择取款图标和金额币种及输入金额数目，系统读取用户请求，接受金额，修改数据库中该用户对应的信息，并提示成功与否；

（3）若用户选择存款图标和金额币种，系统弹出存款框，用户放入现金，系统接收现金并辨认真伪，并反馈存入金额数目，得到用户确认后，修改数据库中该用户对应的信息，并提示成功与否；

（4）若用户选择转账图标和金额币种并输入对方账号和转账金额数目，系统读取用户请求，修改数据库中所涉及到的用户的信息，并提示成功与否；

SR4：

（1）用户选择退出图标；

（2）系统提示拔卡信息。

1. 案例6:对ATM机系统，除了功能性需求之外，还有哪些需求需要定义？试一一写出这些需求。

性能需求：速度、容量、吞吐量、负载、实时性

质量属性：可靠性、可用性、安全性、易用性、可维护性

对外接口：用户界面、通信接口（通信协议,寻址,数据及其格式）

约束：系统开发及运行的环境（包括目标机器、操作系统、网络环境、编程语言、数据库管理系统等）、问题域内的相关标准（包括法律法规、行业协定、企业规章等）、商业规则