

# 需求文档化与验证

刘钦

# 主要内容

- 为什么文档化需求
- 用例文档
- 软件需求规格说明文档
- 文档化需求的注意事项
- 验证需求
- 度量需求功能点

# Why documents?

- Teamwork and communication
  - Project manager
  - Architect
  - Software Designer
  - Programmer
  - Maintainer
  - ...

# Tips of documents

- Why standard?
- Can we solve communication problem with other methods?

# 主要内容

- 为什么文档化需求
- 用例文档
- 软件需求规格说明文档
- 文档化需求的注意事项
- 验证需求
- 度量需求功能点

# 用例文档

- 在用户的角度以用例文本为主描述软件系统与外界的交互
- 基本职责是把问题域信息和需求传达给软件系统解决方案的设计者

# 一、文档的信息

- 1、对文档本身特征的描述信息，例如文档的标题、作者、更新历史等；
- 2、为了方便读者阅读的导读性信息，例如写作的目的、主要内容概述、组织结构、文档约定、参考文献等。

# 二、用例图或者用例列表

使用一个和几个用例图来概括文档中出现的所有用例及用例间的关系。在文档内用例比较多的情况下，也可能使用一个列表来代替用例图，列表内逐一列出文档内所有用例的 ID、名称和其他需要的概括性信息。

# 三、用例描述

## 用例 1

对用例 1 的详细描述，描述的方式如第 6 章所述。

...

## 用例 n

示例

# 主要内容

- 为什么文档化需求
- 用例文档
- 软件需求规格说明文档
- 文档化需求的注意事项
- 验证需求
- 度量需求功能点



# 软件规格说明文档

- 在软件产品的角度以系统级需求列表的方式描述软件系统解决方案

用例

系统规格


侧重于交互流程

侧重于独立需求

以一次交互为基础

以一次交互中的软件系统处理细节为基础

# Excises

- 在一栋大楼中有一个电梯，现在有人打算从1楼使用电梯到达x楼。
- 请写出其用例的场景描述 
- 根据用例场景编写SRS片段

# 示例

1. 引言

1.1 目的

1.2 范围

1.3 定义、首字母缩写和缩略语

1.4 参考文献

1.5 文档组织

2. 总体描述

2.1 产品前景

2.2 产品功能

2.3 用户特征

2.4 约束

2.5 假设和依赖

3. 详细需求描述<sup>\*</sup>

3.1 对外接口需求

3.1.1 用户界面

3.1.2 硬件接口

3.1.3 软件接口

3.1.4 通信接口

3.2 功能需求

3.2.1 系统特性1

3.2.1.1 特性描述

3.2.1.2 刺激/响应序列

3.2.1.3 相关功能需求

3.2.1.3.1 功能需求1.1

...

3.2.1.3.*n* 功能需求1.*n*

3.2.2 系统特性2

...

3.2.*m* 系统特性*m*

3.3 性能需求

3.4 约束

3.5 质量属性

3.6 其他需求

附录

索引

<sup>\*</sup> [IEEE1998]为此处提供了 8 种不同的格式，分别适用于不同的应用情景。模版中的格式仅仅是其中之一。

# 主要内容

- 为什么文档化需求
- 用例文档
- 软件需求规格说明文档
- 文档化需求的注意事项
- 验证需求
- 度量需求功能点

# 技术文档写作要点

- 简洁
- 精确
- 易读（查询）
  - 1、有效使用引言、目录、索引等能够增强文档易读性的方法。
  - 2、使用系统化的方式组织内容信息，提供文档内容的可读性。
- 易修改

# 系统化的方式

- 使用相同的语句格式来描述相似、关联的信息。
- 使用列表或者表格来组织独立、并列的信息。
- 使用编号来表达繁杂信息之间的关系,包括顺序关系、嵌套关系和层次关系。
  - 对图、表进行编号
  - 对文档的章节进行编号
  - 对信息内容进行标识和编


歧义词汇	改进方法
可接受的、足够的	具体定义可接受的内容，说明系统怎样判断“可接受”或“足够”
依赖	描述依赖的原因，数据依赖？服务依赖？还是资源依赖？等等
有效的	明确“有效”所意味的具体实际情况
快的、迅速的	明确指定系统在时间或速度上可接受的最小值
灵活的	描述系统为了响应条件变化或需求变化而可能发生的变更方式
改进的、更好的、优越的	定量说明在一个专门的功能领域内，充分改进的程度和效果
包括但不限于、等等、诸如	应该列举所有的可能性，否则就无法进行设计和测试
最大化、最小化、最优	说明对某些参数所能接受的最大值和最小值
一般情况下、理想情况下	需要增加描述系统在异常和非理想情况下的行为
可选择地	具体说明是系统选择、用户选择还是开发人员选择
合理的、必要的、适当	明确怎样判断合理、必要和适当
健壮的	显式定义系统如何处理异常和如何响应预料之外的操作
不应该	试着以肯定的方式陈述需求，描述系统应该做什么
最新技术水平的	定义其具体含义，即“最新技术水平”意味什么
充分的	说明“充分”具体包括哪些内容
支持、允许	精确地定义系统的功能，这些功能组合起来支持某些能力
用户友好的、简单的、容易的	描述系统特性，用这些特性说明词汇所代表的用户期望的实质

# 可修改性


- Independent
- ID
  - $1, 2, \dots, n$
  - $1, (1.1, (1.1.1, \dots)), 2, (2.1, (2.1.1, \dots))$
  - `Function.task.step`
- Reference, not repeat



# 需求书写要点

- 需求书写要点
  - 使用用户术语
  - 可验证 
  - 可行性
- 需求规格说明文档书写要点
  - 充分利用标准的文档模版，保持所有内容位置得当
  - 保持文档内的需求集具有完备性和一致性。
  - 为需求划分优先级

# Requirements Review

- “After the payment process is complete, the relevant information should be appended to a log file.”
- a. This requirement should be rewritten; it is incorrect.
- b. This requirement should be rewritten; it is ambiguous or inconsistent.
- c. This requirement should be rewritten; it is unrealistic.
- d. This requirement should be rewritten; it is unverifiable.
- e. This requirement is fine.

# Requirements Review

- “The system should be constructed so that it will be easy to add new functionality in the future.”
  - a. This requirement should be rewritten; it is incorrect.
  - b. This requirement should be rewritten; it is ambiguous or inconsistent.
  - c. This requirement should be rewritten; it is unrealistic.
  - d. This requirement should be rewritten; it is unverifiable.
  - e. This requirement is fine.

# Requirements Review

- “The price of a gasoline purchase is computed as the price per gallon for the type of gas purchased, multiplied by the number of gallons purchased (use two decimal points for representing fractions of gallons).”
  - a. This requirement should be rewritten; it is incorrect.
  - b. This requirement should be rewritten; it is ambiguous or inconsistent.
  - c. This requirement should be rewritten; it is unrealistic.
  - d. This requirement should be rewritten; it is unverifiable.
  - e. This requirement is fine.

# Requirements Review

- “The system should be available 24 hours a day, 7 days a week.”
  - a. This requirement should be rewritten; it is incorrect.
  - b. This requirement should be rewritten; it is ambiguous or inconsistent.
  - c. This requirement should be rewritten; it is unrealistic.
  - d. This requirement should be rewritten; it is unverifiable.
  - e. This requirement is fine.

# 主要内容

- 为什么文档化需求
- 用例文档
- 软件需求规格说明文档
- 文档化需求的注意事项
- 验证需求文档
- 度量需求功能点

# 验证需求的方法

- 评审
- 开发系统测试用例
- 度量

# 评审的注意事项

- 重视需求评审
- 保证用户与客户参与
- 用户对场景与线索表现出了最大的兴趣
- 使用检查列表



# 开发系统测试用例

- (1) 以需求为线索，开发测试用例套件；
- (2) 使用测试技术确定输入/输出数据，开发测试用例。

# 测试用例套件

- 基于用例描述，可以为销售处理确定测试用例套件

测试用例套件		覆盖流程			
TUS1	正常流程	3a	3b	5-8a	
TUS2	正常流程	1a			9a 11a
TUS3	正常流程			5-8b	

-

表 7-2 测试用例套件对需求的覆盖情况

	测试用例套件 3	测试用例套件 2	测试用例套件 3
Sale.Input	TUS1	TUS2	TUS3
Sale.Input.Member		TUS2	
Sale.Input.Payment	TUS1	TUS2	
Sale.Input.Cancle			TUS3
Sale.Input.Del	TUS1		
Sale.Input.Goods	TUS1	TUS2	TUS3
Sale.Input.Invalid	TUS1	TUS2	TUS3
Sale.Member.Start		TUS2	
Sale.Member.Notstart		TUS2	
Sale.Member.Cancle		TUS2	
Sale.Member.Valid		TUS2	
Sale.Member.Valid.List		TUS2	
Sale.Member.Invalid		TUS2	
Sale.End.Null	TUS1	TUS2	
Sale.Paynent.Goods	TUS1	TUS2	
Sale.Paynent.Goods.Gift	TUS1	TUS2	
Sale.Paynent.Goods.Check	TUS1	TUS2	
Sale.Payment.Goods.End	TUS1	TUS2	
Sale.Del.Null	TUS1		
Sale.Del.Goods	TUS1		
Sale.Goods	TUS1	TUS2	TUS3
Sale.Goods.Num	TUS1	TUS2	TUS3
Sale.Goods.Subtotal.Special	TUS1	TUS2	TUS3
Sale.Goods.Subtotal.Common	TUS1	TUS2	TUS3

# 测试用例对需求的覆盖情况

# 建立测试用例

- 主要是基于规格的技术，设计测试场景的输入与输出数据

ID	输入				预期输出
	商品信息	特价	赠品	支付	
TUS1-1	无商品	无	无	无	不做任何处理，关闭销售任务
TUS1-2	商品1 (1、1(双)、35) 商品2 (2、1(双)、50)	无	无		
TUS1-3	商品1 (1、1(双)、35) 商品2 (2、1(双)、50)	无	无	85	无找零，系统行为满足后置条件
TUS1-4	商品1 (1、1(双)、35) 商品2 (2、1(双)、50)	商品1特价20		100	找零30，系统行为满足后置条件
TUS1-5	商品1 (1、1(双)、35) 商品2 (2、1(双)、50)	总额特价50以上0.8折		100	找零31，系统行为满足后置条件
TUS1-6	商品1 (1、1(双)、35) 商品2 (2、1(双)、60)	商品1特价20 总额特价50以上0.8折		100	找零32，系统行为满足后置条件
			...		
			...		

# 主要内容

- 为什么文档化需求
- 用例文档
- 软件需求规格说明文档
- 文档化需求的注意事项
- 验证需求文档
- 度量需求功能点

# 度量需求

- 用例的数量
  - 平均每个用例中的场景数量
  - 平均用例行数
- 软件需求数量
- 非功能需求数量
- 功能点数量

# 度量的意义

- 如果平均的用例场景数量过低，那么就可能存在对异常流程考虑不周的可能。
- 如果平均用例行数过大或者过小，那么可能对用例的细分粒度过大或者过小。
- 用例数量、软件需求数量和功能点数量应该是相对比例均衡的，如果三者之间有着非常大的差距，那么可能会有需求的遗漏。



# 功能点度量

- 用于估算和度量软件系统规模与复杂度的抽象单位。
- 在需求开发阶段，估计代码行数误差较大，使用功能点来估算和度量软件系统的规模与复杂度则有较好的效果。

# 功能点度量

- 输入数量：一次有意义的输入，需要程序员进行一次编程处理
- 输出数量：系统需要对外展示的内容组，表现为屏幕界面、打印输出、错误提示等等。
- 查询数量：用户的“命令”输入，通常表现为鼠标点击和键盘输入（热键）
- 逻辑文件数量：系统内部的持久化数据，包括文件、数据表等
- 对外接口数量：与外部系统交换数据的软硬件通信接口。

Sale.Input	系统应该允许收银员在销售任务（输出：销售任务主界面）中进行键盘输入
Sale.Input.Member	在收银员请求输入（查询）会员客户编号时（输入），系统要标记会员，参见 Sale.Member
Sale.Input.Payment	在收银员输入结束商品输入命令（查询）时，系统要执行结账任务，参见 Sale.Payment
Sale.Input.Cancle	在收银员输入取消命令（查询）时，系统关闭当前销售任务（输出），开始一个新的销售任务
Sale.Input.Del	在收银员输入删除已输入商品命令时（查询），执行删除已输入商品命令，参见 Sale.Del
Sale.Input.Goods	在收银员输入商品目录中存在的商品标识时（输入），系统执行商品输入任务，参见 Sale.Goods
Sale.Input.Invalid	在收银员输入其他标识时，系统显示输入无效
Sale.Member.Start	在销售任务最开始时请求标记会员，系统要允许收银员进行输入
Sale.Member.Notstart	不是在销售任务最开始时请求标记会员，系统不予处理
Sale.Member.Cancle	在收银员取消会员输入（查询）时，系统关闭会员输入任务，返回销售任务，参见 Sale.Input
Sale.Member.Valid	在收银员输入已有会员的客户编号时（逻辑文件），系统显示该会员的信息（输出）

# 需求的度量

# 销售任务

- 输入： 3
- 输出： 11
- 查询： 7
- 逻辑文件： 12
- 对外接口： 0

# 功能点测度总数

功能点测度总数 =  $\sum_{j=1}^5 (C_j \times f_j)$

功能点测度		加权因子fj		
		简单系统	中等系统	复杂系统
C1	输入数量	3	4	6
C2	输出数量	4	5	7
C3	查询数量	3	4	6
C4	逻辑文件数量	7	10	15
C5	对外接口数量	5	7	10

功能点测度总数=3×4+11×5+7×4+12×10=215

i	描述	MSCS估计值 (Fi)
1	系统需要备份和恢复吗?	0
2	需要专门的网络数据通信吗?	0
3	存在分布式处理功能吗?	5 (RMI)
4	性能关键吗?	2
5	系统将运行在一个现有的、使用困难的操作环境吗?	3 (收银员)
6	系统需要在线数据项吗?	5 (领域数据都是在线的)
7	在线数据项目需要对多个屏幕或操作建立输入事务吗?	4 (多客户端)
8	逻辑文件在线更新吗?	5
9	输入、输出、文件或查询是复杂的吗?	2 (都较为独立)
10	内部处理是复杂的吗?	3 (特价与赠品计算)
11	所设计的代码要求可复用吗?	4 (课程要求)
12	设计要求包括移交与安装吗?	3
13	系统需要设计为多个安装以适应不同组织吗?	0
14	系统设计要求易于修改和易于使用吗?	4 (课程要求)



# 销售用例的功能点

$$FP = \text{功能点测度总数} \times [0.65 + 0.01 \times \sum_{i=1}^{14} F_i]$$

- $FP(\text{Sale}) = 215 * (0.65 + 0.01 * 40)$
- $= 225.75$

# Q&A