



第三部分 软件需求工程

- 3.1 软件需求与需求工程
- 3.2 需求获取与建模
- 3.3 需求规格说明



需求规格说明书(SRS)

□ 软件需求规格说明书 **SRS (Software Requirements Specification):**

- 需求开发的结果，精确的阐述一个软件系统必须提供的功能和性能，以及它所要考虑的限制条件。
- 为了使用户和软件开发者双方对该软件的初始规定有一个共同的理解，使之成为整个开发工作的基础。





需求规格说明书(SRS)

- 是具有一定法律效力的合同文档
- 清楚地描述软件在什么情况下，需要做什么，以及不能做什么
- 以输入/输出、输入到输出之间的转换方式来描述功能性需求
- 描述经过干系人磋商达成共识的非功能性需求
- 一般参考需求定义模板，覆盖标准模板中定义的所有条目
- 作为后续的软件评估依据和变更的基准



需求规格说明书(SRS)

■ SRS作为软件开发各类人员之间进行理解和交流的手段:

- 用户：通过**SRS**指定需求，检查需求描述是否满足期望；
- 设计人员：了解软件需要开发的内容，将其作为软件设计的基本出发点；
- 测试人员：制定测试计划、测试用例和测试过程；
- 产品发布人员：编写用户手册和帮助信息；
- 项目管理人员：规划软件开发过程、准确估计开发进度和成本、控制需求变更过程。



SRS应包含的内容

□ 功能(Functionality):

- 该软件系统能够向用户提供何种服务?

□ 外部接口(External interfaces):

- 该软件系统如何与用户、操作系统、硬件、其他软件系统进行交互?

□ 性能(Performance):

- 软件系统在运行速度、可用性、响应时间、恢复时间等方面有哪些要求?

□ 非功能属性(Non-Functional Attributes):

- 软件系统在可移植性、安全性、可靠性、可维护性等方面有哪些要求?

□ 约束条件(Design constraints imposed on an implementation):

- 是否存在必要的标准、编程语言、运行环境、资源约束等因素?



SRS不应包含的内容

□ 项目开发计划

— 诸如成本、人员、进度、工具、方法等

□ 产品保证计划

— 诸如配置管理、验证与测试、质量保证等

□ 软件设计细节

— 需求通常用于表达“做什么”，而不描述“如何做”





选择合适的规格说明方式

- 考虑以下两个具体项目场景:

1) 小型项目, 1名程序员, 2个月的开发周期

- 程序员直接和用户对话, 写了几页纸的备忘录

2) 大型项目, 50 名程序员, 2年的开发周期

- 专门的团队进行需求建模与分析, 撰写了500页的需求规格说明

	项目 A	项目 B
需求撰写的目的	凝练工程师对问题的理解; 获取用户反馈	文档本身就是交付物; 为研发人员提供丰富细节
管理方面的用途	资源的分配与规格说明书无关; 相关资源已经就位	基于规格说明进行估算和任务进行规划
预期读者	主用户: 作者本人; 从用户: 客户	主用户: 程序员, 测试人员, 项目经理; 从用户: 客户



好的SRS应具备的特点

- 正确性(Correct)
- 无二义性(Unambiguous)
- 完整性(Complete)
- 一致性(Consistent)
- 按重要度/稳定性排序(Ranked for importance and/or stability)
- 可验证性(Verifiable)
- 可修改性(Modifiable)
- 可跟踪性(Traceable)



软件需求问题：不一致

编号	标识	利益相关者	需求描述	杀手功能	重要性	优先性	工作量
1	Browse Course	学生 教务人员 老师	根据范围浏览可供选修的所有课程，这里的范围包括开设课程的学院、职称、课程类别（选修/必修），所浏览的课程信息包括课程名称、上课老师信息、开设学院、上课时间/地点、 课程类别 、学时/学分数、课程描述	否	重要	优先	10
			对同一内容大家理解和认识不一样，需要达成一致				

课程类别：有些人认为需要，有些人认为不需要



软件需求问题：模糊不准确

编号	标识	利益相关者	需求描述	杀手功能	重要性	优先性	工作量
1	Browse Course	学生 教务人员 老师	根据范围浏览可供选修的所有课程，这里的范围包括开设课程的学院、职称、课程类别（选修/必修），所浏览的课程信息包括课程名称、上课老师、开设学院、上课时间/地点、课程性质、学时/学分数、课程描述	否	重要	优先	10
			某项内容没有清晰的描述，需要细化或者准确化 学院类别：按照学院编号还是学院的名称				



软件需求问题：不完整

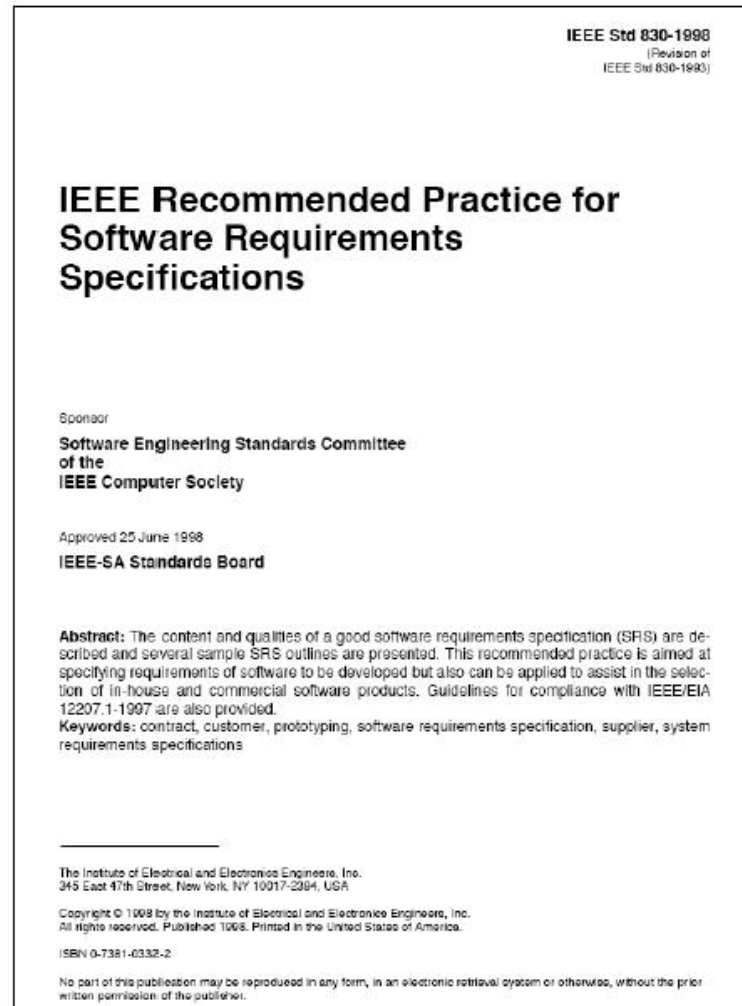
编号	标识	利益相关者	需求描述	杀手功能	重要性	优先性	工作量
1	Browse Course	学生 教务人员 老师	根据范围浏览可供选修的所有课程，这里的范围包括开设课程的学院、职称、课程类别（选修/必修），所浏览的课程信息包括课程名称、上课老师、开设学院、上课时间/地点、课程性质、学时/学分数、课程描述	否	重要	优先	10

漏掉了某些重要的软件需求，需要补充完整

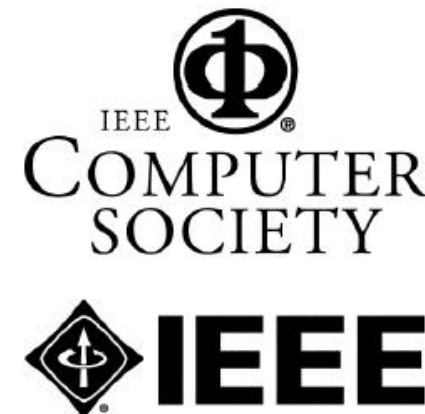
学期：按照学期来浏览可选修的课程（暑期/秋季）



SRS模板



- 采用**IEEE标准830-1998**所给出的模板进行**SRS**书写:
- **IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications**





SRS的三大组成部分

1. 引言(introduction)

概要描述本SRS，便于读者理解文档如何编写、如何阅读和解释

2. 整体描述(overall description)

概述正在定义的软件产品以及它所运行的环境、使用产品的用户、已知的限制/假设/依赖等

3. 需求描述(specific requirements)

每一项需求的详细定义(输入、输出、业务逻辑等)，这也是SRS的主体。



SRS的三大部分：“1. 引言”

1. 引言(introduction)

1. 目的(purpose)
2. 范围(scope)
3. 术语表(definitions, acronyms, and abbreviations)
4. 参考文献(references)
5. 整体结构(overview)

2. 整体描述(overall description)

3. 需求描述(specific requirements)



SRS的三大部分：“2. 整体描述”

1. 引言(introduction)

2. 整体描述(overall description)

1. 产品上下文环境(product perspective)
2. 产品外部接口(external interface)
3. 产品功能(product functions)
4. 用户类和特征(user characteristics)
5. 设计和实现上的约束(constraints)
6. 运行环境(environment)
7. 假设和依赖(assumptions and dependencies)
8. 未来的需求(future requirements)

3. 需求描述(specific requirements)



SRS的三大部分：“3. 需求描述”

1. 引言(introduction)
2. 整体描述(overall description)
3. 需求描述(specific requirements)
 1. 各项功能的详细描述
(根据各项功能来组织章节)
 2. 非功能需求



随时准备迎接需求变化

- 这是一种态度
- 越多的干系人参与，将获得越多的需求特征
- 但不能通过减少干系人的方法来解决这个问题
- 干系人有改变他们想法的权利
- 不要问：“这是你最终的需求吗？”
- 请将变化看成机会，而非威胁