

1、用例技术

1.1、用例图

用例图描述软件系统为用户或外部系统提供的服务。

- 用例图最重要的元素是参与者（Actor）和用例（Use Case），以此体现系统能为外部参与者（Actor）提供的功能（Use Case）。
- 参与者是与系统交互的角色或系统，既可以是系统的用户（User），也可以是和系统有直接交互关系的系统（System）。
- 用例的名称应该从参与者的角度进行描述，并以**动词开头**，这样一来通过“读图”可以清晰地获得用例图的语义。

用例图所做的，一是确定与本系统交互的角色或外部系统，二是描述系统必须提供的功能。

1.2、用例简述（用户故事）

通过简短的文字对用例的功能进行描述，一般而言，用例简述都应包含成功场景的简单描述。

- 用例名称
- 用例简述
- 优先级

1.3、用例规约

用例规约的主要目的是界定软件系统的行为需求，行为需求是指软件系统为了提供用户所需的功能而必须执行哪些行为。

1.3.1、格式

1. 用例名称
2. 简要说明
3. （需求背景及可能的变化）--可选
4. 事件流
 - a. 基本事件流
 - b. 扩展事件流
5. 非功能需求
6. 前置条件
7. 后置条件
8. （使用频率）--可选
9. 扩展点
10. 优先级

1.3.2、要点

- 用例规约对系统行为的描述是以用户为中心展开的，便于和用户交流。
- 既关注主事件流所描述的成功场景，也关注备选事件流所描述的异常场景，有利于促进系统化思维、发现异常场景、完善系统功能和提高易用性。
- 实践中可以对用例规约的格式进行剪裁甚至扩充，以满足具体工作需要。
- 不应该包含用户界面原型。
- 后置条件应覆盖所有可能得用例结束后的状态。不仅仅是用例成功结束后的状态，还应该包含用例因发生错误而结束后的状态。

1.4、用例实现（鲁棒图）

即协作，多个对象为了完成某种目标而进行的交互。

1.5、与需求关系

用例图从总体上反映了用户需求。

用例简述是行为需求的简化描述。

用例规约也描述系统的行为需求，但属于“规格级”详述。

鲁棒图属于设计范畴，确切地讲是初步设计。

2、应用场景

2.1、用例与需求分析

4种用例技术，用于3种实践场景：

- 用例图 + 用例简述：用于**需求捕获**，又可用于业务不太复杂的系统**需求分析**。
- 用例规约：用于**需求分析与需求规格定义**。
- 用例实现（鲁棒图）：用于**初步设计**。

需求捕获、需求分析以及系统分析之间相互伴随、交叉进行。

2.2、用例与需求文档

1. 前言

- a. 目的
- b. 范围
- c. 定义、缩写词、略语
- d. 参考资料

2. 需求概述

- a. 用例模型 // 此节，归档“用例图”等。
- b. 限制与假设

3. 具体需求

- a. 用例描述 //此节，归档“用例规约”等。

- b. 外部接口需求
 - i. 用户需求
 - ii. 硬件接口
 - iii. 软件接口
 - iv. 通信接口
- c. 质量属性需求
 - i. 性能
 - ii. 易用性
 - iii. 安全性
 - iv. 可维护性...
- d. 设计和实现约束
 - i. 必须遵循的标准
 - ii. 硬件的限制....

2.3、用例与需求变更

用例图，能稳定反应用户级需求。

用例规约，因为精确所以易变。

- 推后用例细化：不是对软件架构起关键作用的用户，可以推迟到要实现该用例所定义的功能之前才进行细化。
- 激发需求变更：通过不断增加功能、发布小版本、提供给用户试用、接受用户反馈等一系列的活动，让用户一步步明确自己真正想要的功能。