**测试文档**

# 测试要求

1.准确性(Accurate)

Tests what the descrīption says it will test. 测试覆盖了描述部分需要测试的内容。

2.经济性(Economical)

Has only the steps needed for its purpose.测试用例没有冗余的步骤

3.可重复性(Repeatable)

Self standing, same results no matter who tests it.测试用例应该是独立一致的，不管任何人执行，结果都一致。

4.可追踪(Traceable)

To a requirement.测试用例应该追溯到具体需求。

5.自我清理(Self cleaning)

Returns the test environment to clean state.测试结束后，恢复到原有干净的状态，不应该对原有系统造成影响。

6. 结构化和可测试性（Structure and testability）

测试用例应该是结构化。一般可以根据一个横向维度，对测试用例进行功能模块的划分；同时纵向维度上可以根据测试类别对测试用例进行纵向结构的划分。

测试同时应该是可测试性的。对于无法执行的测试用例是没有意义的。

7. 规范性

Has a name and a number. 命名 + 编号

Has a stated purpose that includes what requirement is being tested。

对测试的内容有清晰的目的。

Has a descrīption of the method of testing。测试方法

Specifies setup information - environment, data, prerequisite tests, security access。环境， 数据， 前提，权限。

Has actions and expected results。步骤， 期望结果。

tates if any proofs, such as reports or screen grabs, need to be saved。

Leaves the testing environment clean。清理数据，还原系统。

这里其实包含一个测试用例的组成部分：

1. 命名， 编号（一般会结合功能进行命名）

我们的命名原来有很多问题，这次要准确。

1. 目的描述

这个要有

1. 测试类型（该测试用例属于功能测试，性能测试，单元测试，系统测试等等）
2. 环境

最前面写清楚

1. 测试数据

数据描述清楚一些

1. 前提
2. 步骤
3. 期望结果
4. 实际结果
5. 测试结果（通过还是失败）

一般来说测试用例，不会说明备份系统，还原系统的步骤，这两个步骤一般都会由自动化脚本自动执行。

8. 简洁性

Uses active case language。使用主动语气语句，就是动词搭配名字的语句，比如Put value, Click button. 不使用， please put value, you go to…

Does not exceed 15 steps。不超过15步。

Matrix does not take longer than 20 minutes to test。执行时间不要超过20分钟。这两点其实是希望测试用例的规模比较小，粒度不要太大。这点在大型系统不太适用。

9. 完整性

Automated scrīpt is commented with purpose, inputs, expected results。自动化脚本应该包含必要的注释，包括，目的，输入，预期结果。

Setup offers alternative to prerequisite tests, if possible。 如果可能，提供不同的前置条件下的测试。

测试用例应该尽量完整，包含自动化脚本。

10. 有效性

Is in correct business scenario order with other tests? 测试用例是否符合商业案例？

11. 独立性

测试用例应该保持独立性，一个测试用例最好是能独立运行，不依赖于其他的测试用例的输出结果。出于结构的考虑，有些特殊测试用例设计本身就是作为setup来设计的，这个除外。