

# How To Organize UIauto Testcases

## OverView

本文旨在说明如何筛选可自动化用例,并且规范Case的基本内容.

## Think

100%自动化只是梦想,自动化+人工才是落地的方向.抛开已有的测试用例,思索一下什么是每个版本你重复的工作?什么是最消耗时间的工作?

### 1 写出你重复的工作,具体到每一步你是如何执行的.

打开一个页面,请求一个接口 ...如果你能成功的写出来,那么恭喜你这不仅能做成自动化且你已成功了一半.反之,则放弃.

### 2 提前最消耗时间工作中,重复的部分,然后参考1.

### 3 你的自动化case是否能减少工作时间是唯一的校验标准.

请在开始写用例之前,给出他节省了那部分时间,你是如何保证他的安全性.

### 4 不要一味追求覆盖率和case数量

无用的case只会让时间变得更长且毫无意义.

## Priority

当你思索完后开始编写用例,你应该清晰的了解到每个case该在什么阶段执行.

Case优先级决定了RF中Case的Tag

你应当优先去编写高优Case

## 成本

维护成本是一个高昂的成本,请在编写Case时确认,自己能在短时间内维护好此Case.

短时间指下版本之前.

## Case Detail

### 1. 前置条件

- 测试账号
- 视频av号
- 跳转的scheme
- 测试环境
- Mock数据&api
- 是否首次启动出现
- 设置

### 2. 用例步骤

- 操作步骤

### 3. 预期结果

- 截图
- 断言事项

## CaseTitle

1. 模块-页面-具体内容-断言
2. 评论-视频详情页-评论楼中楼-点击可跳转(此方式可直接复制作为RF的Case)
3. 不要用中文的[]和逗号
4. 以上均为建议,不做要求,只要求同一模块页面的Case能连续展示,不会被其他Case打乱顺序.

## 原子性

一定要保证每条case的独立性!请严格的考虑,各种设置对你Case的影响!请不要仅在此模块的第一条Case中说明设置条件,账号等信息.

## Assert

你的断言应当确保,当你用例执行成功时结果是可靠的.截图,埋点,日志,接口,UI.请结合多种来保证结果的可靠.

可靠的断言:获取到正确的断言条件,不然则Fail

不可靠的断言:没有获取到错误的断言条件,则Pass

错误是无穷集合,你无法通过说明某个错误不存在来证明它是正确的.请选择唯一的正确条件.