

# 基于模式的互联网测试设计

高翔(季哥)

# Agenda

- 传统的测试设计
- 三大测试模式
- 基于模式的测试设计

# 测试流程的测试设计阶段



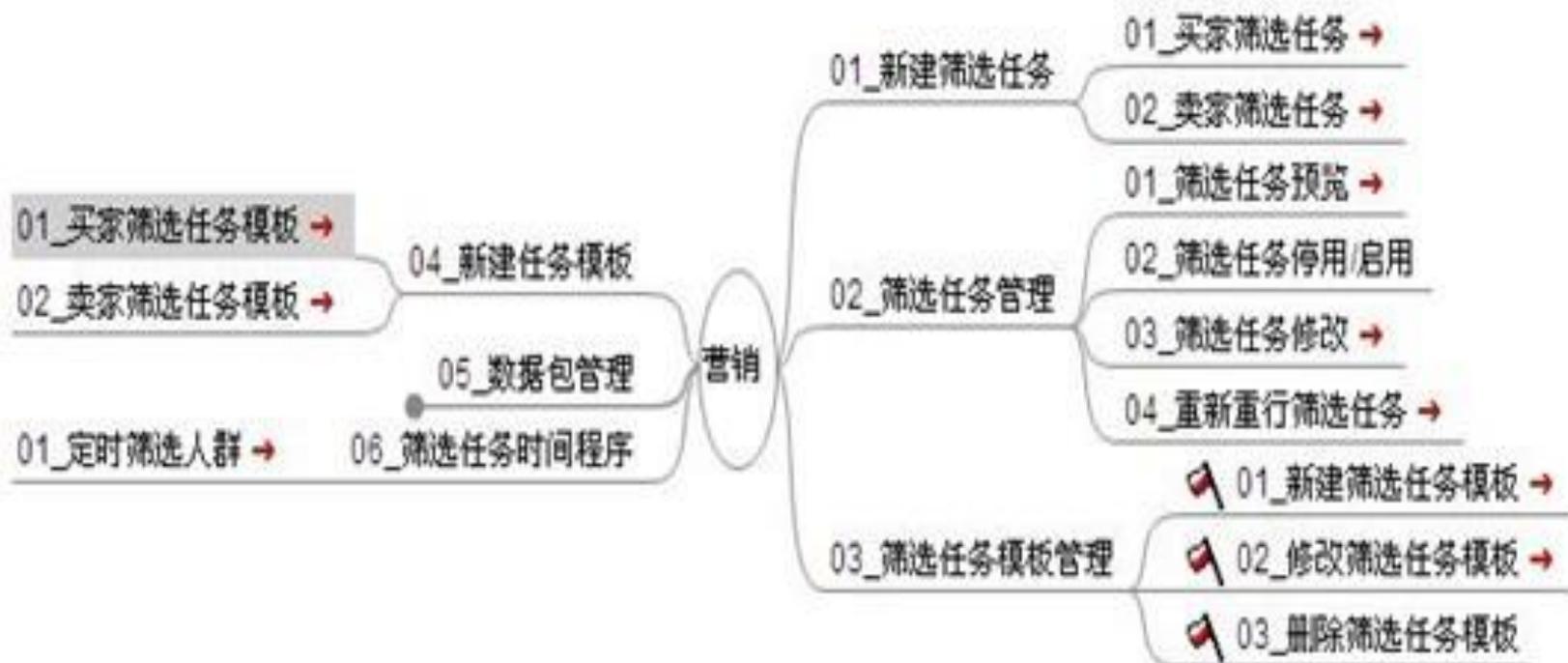
# 测试设计的目的

- 对评审后的需求进行进一步的评审
- 对复杂需求做分解且加深其理解
- 产出具体需求的测试思路和异常分支
- 根本性的指导后续测试用例编写
- 架构整个被测系统的测试需求和用例

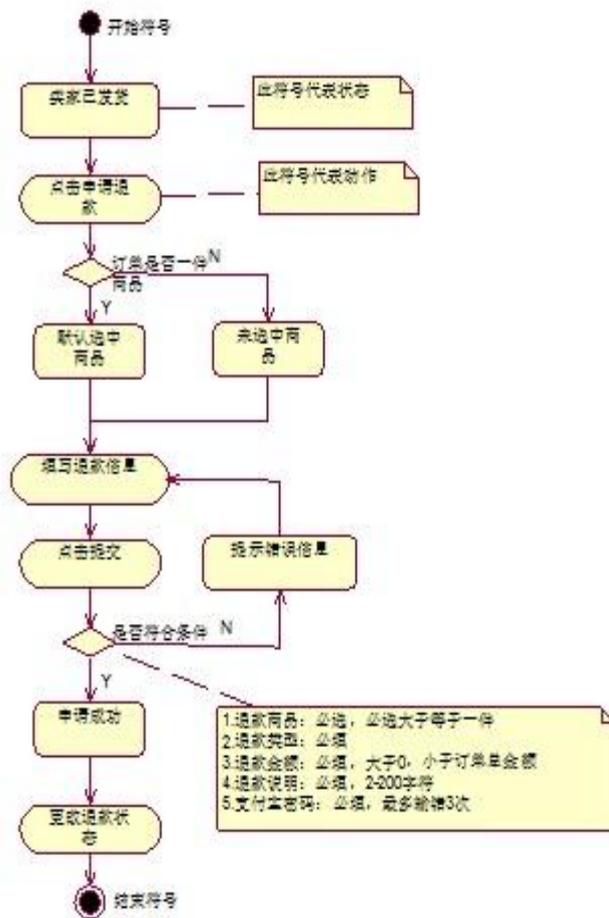
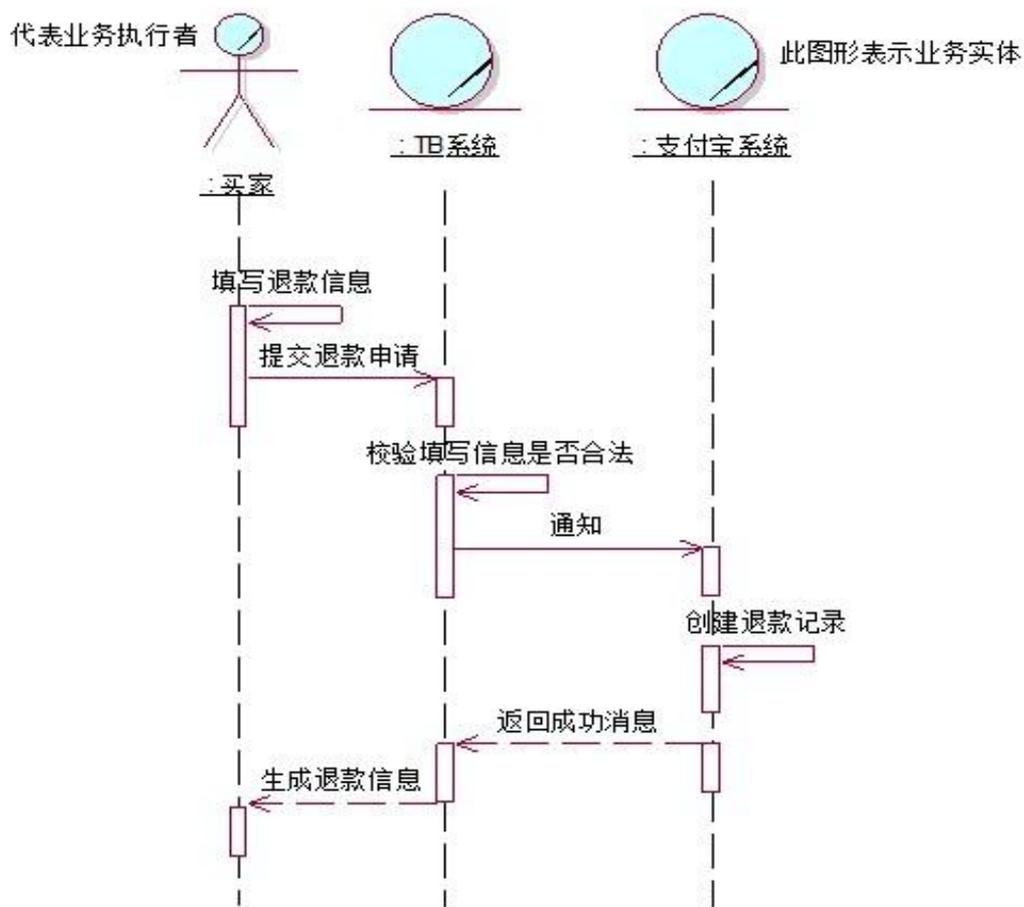
# 测试设计的产出形式

- 功能需求分解MM图： Free Mind, Xmind
- 系统时序图： *Rose Sequence diagram*
- 模块流程图： *Rose Active diagram*
- 用例模块判定表： Excel
- Excel模板： IPO

# 功能需求分解MM图



# 系统时序图和模块流程图



# 用例模块判定表

用例描述	组合描述	条件						结果	
		卖家类型		出售类型		宝贝类型			
不同类型的卖家发布宝贝		B商家	C卖家	发布一口价	发布拍卖	全新	二手	允许发布	不允许发布
	B商家发布全新一口价宝贝	1		1		1		1	
	.....	1		1			1		1
	.....	1			1				1
	.....		1	1		1		1	
	.....		1		1		1	1	
	.....								

# 测试设计的开始和结束

- 产品的需求评审后开始
- 测试设计和测试分析同步
- 啥时候结束呢？
- 测试设计的产出物是什么？分析还是设计？

# 测试设计的核心价值

- 让你清醒地认识被测产品的需求
- 让你分析出被测产品的风险和测试范围
- 得出测试范围内的所有确定的测试思路
- 检验对开发设计文档的理解和评审

# 互联网测试的特点

- 功能层面：
  - 有控件
  - 有组件
  - 有页面
  - 有功能
- 非功能层面：
  - 安全测试、接口测试、性能测试等

# 互联网测试设计的问题

- 测试设计的时间较长，且不能快速对一个系统进行标准和完整的测试设计
- 新项目存在较多的公共功能，未能充分利用公共功能的原有测试设计
- 项目中发现的好的Bug场景思路，未能应用到新项目的测试设计过程中
- 线上的故障场景思路，未能应用到新项目的测试设计过程

# 缺陷模式的结构

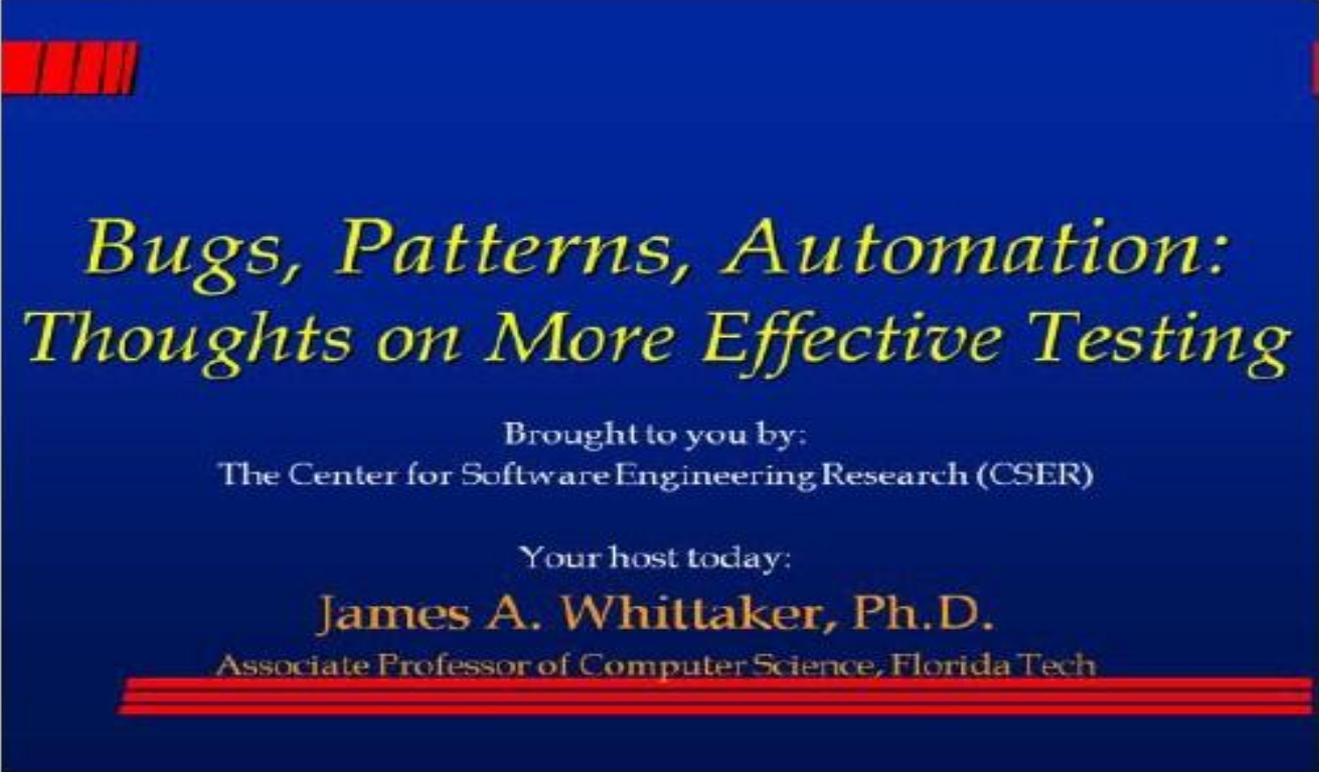
- 一个标准模式的结构：
  - --什么时候应用该模式攻击SUT
  - --什么样的错误出现说明该模式攻击成功了
  - --使用该模式攻击后，如何确定错误
  - --如何指导我们去攻击SUT
  - --例子和分析
- *缺陷模式：默认值输入的应用*
- *<<How to break software>>*

# 缺陷模式的作用和视角

- 测试缺陷模式：一种文档化的过程，且分享从缺陷中获取到的知识
- --- 有利于对新测试人员的培训
- --- 有利于指导自动化测试的实施
- 系统接口、用户接口（输入、输出、存储数据、计算）

# 缺陷模式

James A. Whittaker



*Bugs, Patterns, Automation:  
Thoughts on More Effective Testing*

Brought to you by:  
The Center for Software Engineering Research (CSER)

Your host today:  
**James A. Whittaker, Ph.D.**  
Associate Professor of Computer Science, Florida Tech

# 缺陷->模式->自动化

- 首先，发现一个缺陷
- 然后，确定该缺陷的根本原因，且设计一个模式来抓住其关键的本质
- ---使用该模式去发现更多类似的缺陷
- 最好，确定重复的、麻烦的任务，且编写一个工具来自动化模式的使用

# 测试模式和MBT

- 测试模式
- (1) 一定测试范围内，总结关键测试点，进行简单的归纳和建模
- (2) 得到了能让测试人员快速学习该测试范围内的关键测试点的测试模式
- MBT(基于模型的测试)中的模型
- (1) 对产品需求进行功能分解、逻辑梳理
- (2) UML建模 (TestOptimal、AETG、SpecExplorer)

# 互联网测试三大模式

## 基础功能测试模式

- 重复测试设计严重、效率低、易遗漏

## 线下Bug模式

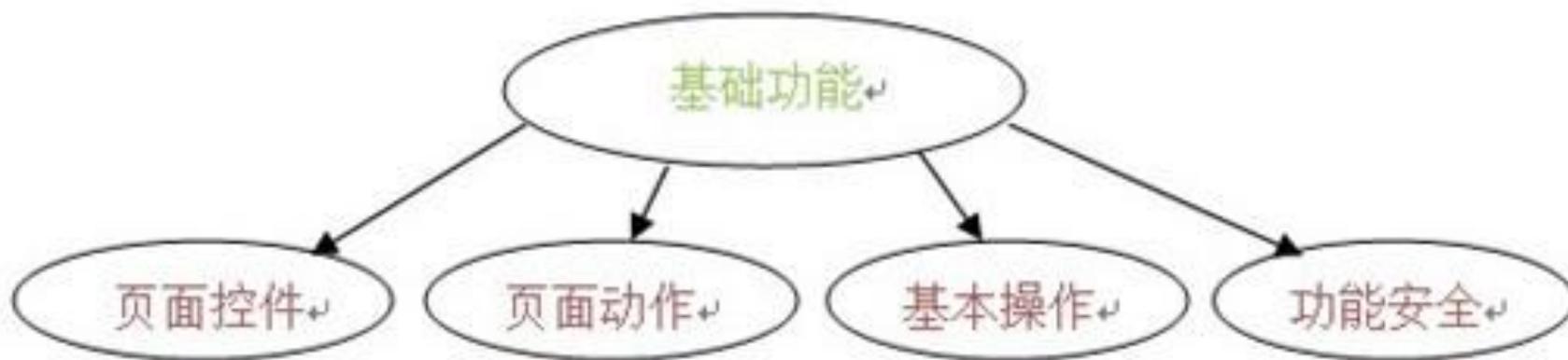
- 经典测试思路、失败模式的共性、分享与成长

## 线上故障模式

- 全面了解线上环境、失败模式的特殊性、Bug预防

# 基础功能测试模式

- 前身：公共用例库
- 四大类型：



# 公共用例中心

**公共用例**

- 00\_页面控件
  - 01\_单选(select)下拉选择框
  - 02\_时间控件
  - 03\_富文本框(textarea)控件
  - 04\_勾选框(checkbox)控件
  - 05\_布尔选择框(radio)控件
  - 06\_表单控件
  - 07\_listbox列表框控件
  - 08\_验证码控件
  - 09\_tab标签页控件
  - 10\_跳转链接控件
  - 11\_功能链接控件
  - 12\_普通文本框控件
- 01\_页面动作
  - 00\_翻页
  - 01\_文件上传
  - 02\_文件下载
  - 03\_表单清空
  - 04\_表单提交
  - 05\_全选 (仅提供测试设计思路, 具体用例可根据自己业务编写)
  - 06\_反选 (仅提供测试设计思路, 具体用例可根据自己业务编写)
  - 07\_重置 (仅提供测试设计思路, 具体用例可根据自己业务编写)
  - 08\_文件保存 (仅提供测试设计思路, 具体用例可根据自己业务编写)
- 02\_基本操作 (仅提供测试设计思路, 具体用例可根据自己业务编写)
  - 00\_单个查询
  - 01\_新增
  - 02\_修改
  - 03\_删除
  - 04\_数据导入
  - 05\_级联查询
- 03\_功能安全
  - 01\_cookies记录和失效
  - 02\_CSRF-免登录(无密码验证)时加token验证
  - 03\_CSRF-数据增删改的操作
  - 04\_CSRF-无check code验证
  - 05\_SQL注入式漏洞
  - 06\_URL REDIRECT安全测试标准用例
  - 07\_不安全的存储
  - 08\_不恰当的异常处理
  - 09\_多系统之间的账户如果单一违背是否有负面影响, 如支付宝账户解绑
  - 10\_检查网页、通知内容, 提示信息

**公共用例提取**

**选择用例类型**

一级类型: 00\_页面控件

二级类型: 03\_富文本框(textarea)控件

**选择您需要的用例**

选择用例:

- 01\_校验为空
- 02\_校验刷新
- 03\_等于最大值长度
- 04\_等于最大值-1长度
- 05\_等于最大值+1长度
- 06\_等于最小值长度
- 07\_等于最小值-1长度
- 08\_校验滚动条
- 09\_校验支持自动换行

**请输入您的参数**

页面名称: 注册页面

控件名称: 用户名

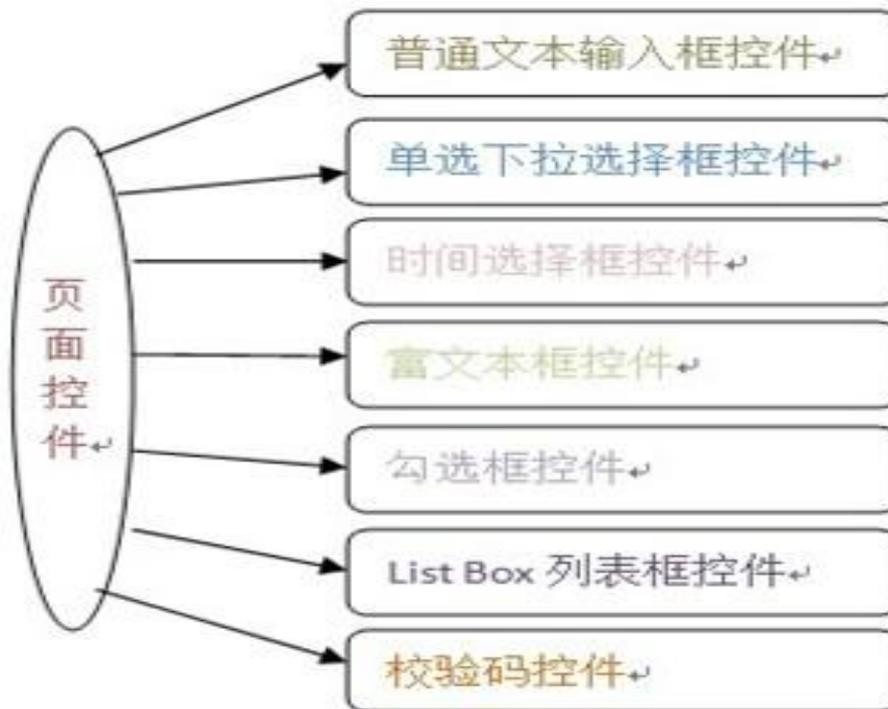
最大值: 225

最小值: 1

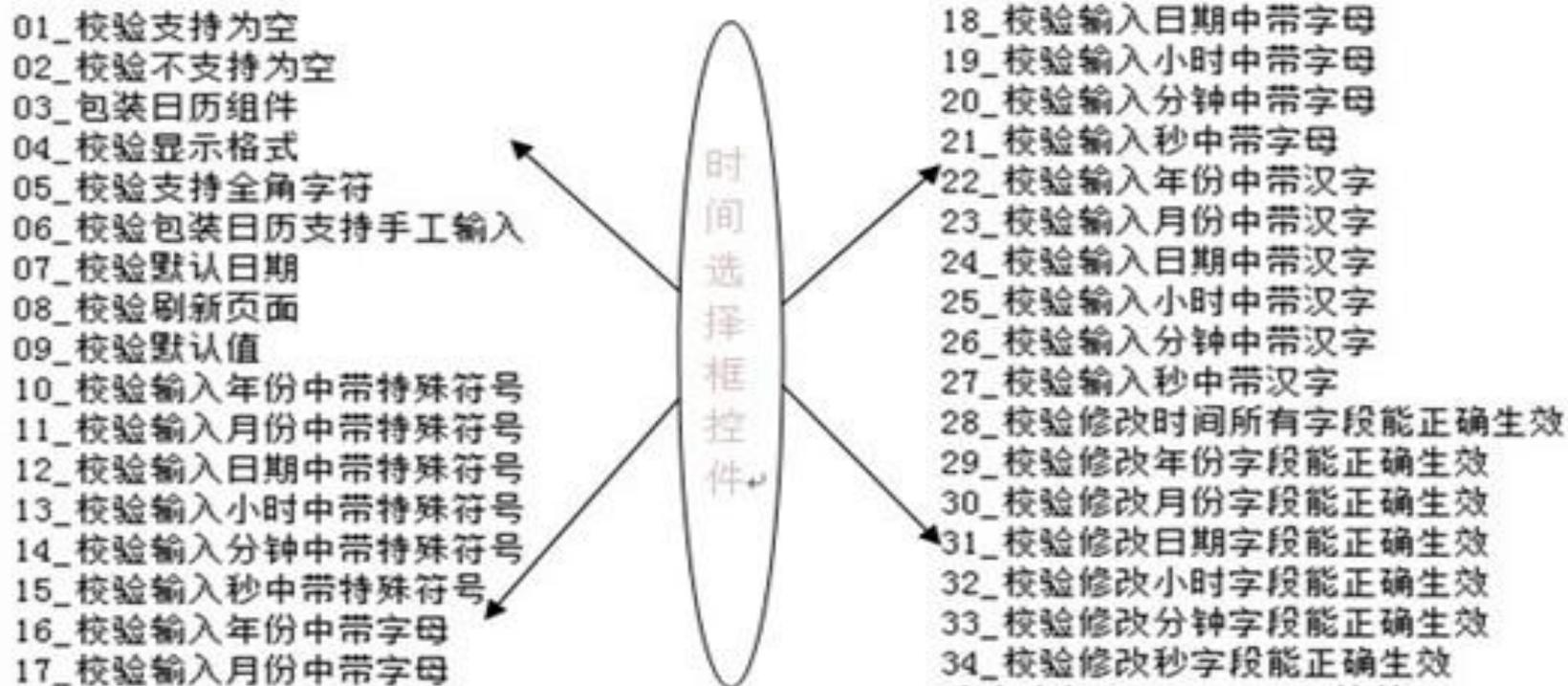
选择空间 导入空间 帮助

# 页面控件模式

- 页面控件的角度出发：
- 可扩展：

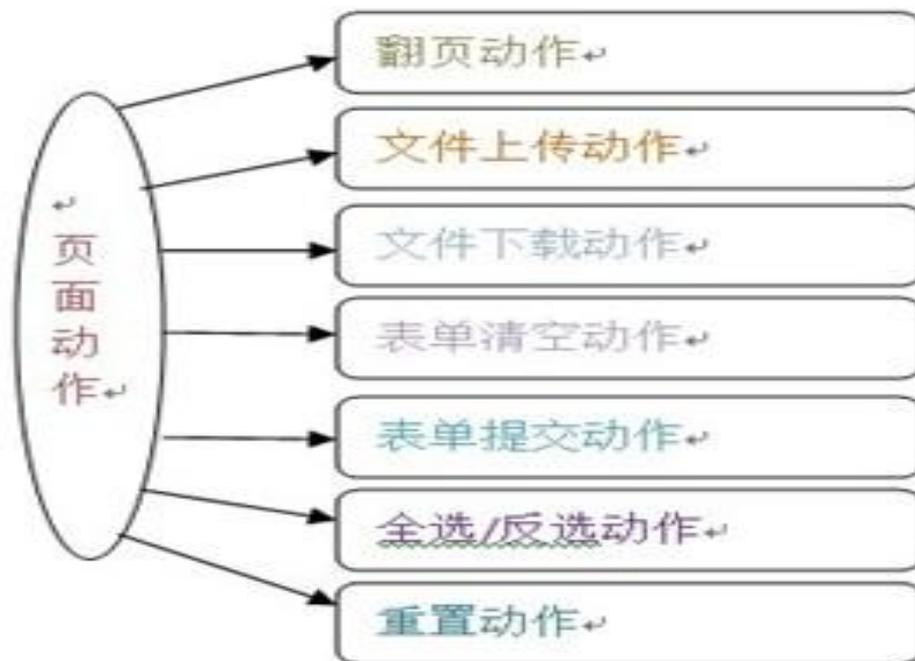


# 时间选择框模式



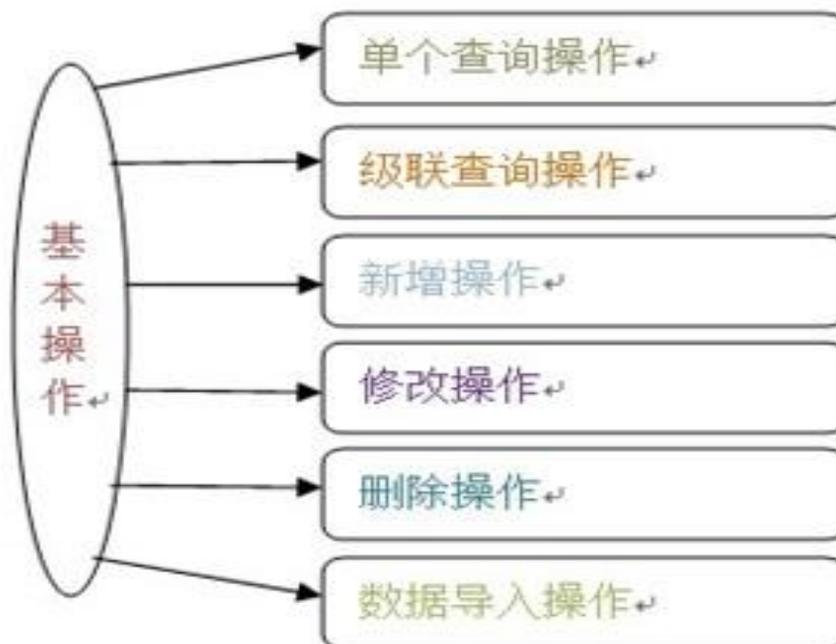
# 页面动作模式

- 页面动作/组件的角度出发:
- 可扩展:



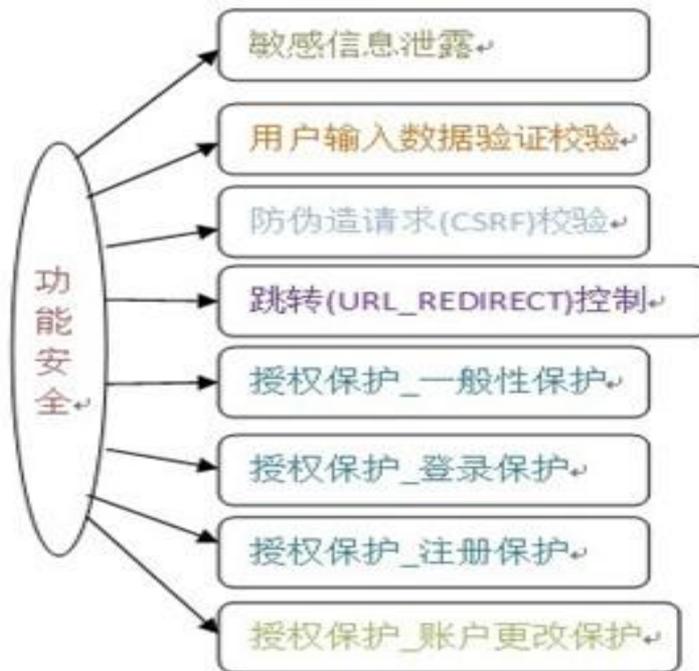
# 基本操作模式

- 页面功能/操作的角度出发:
- 可扩展:



# 功能安全模式

- 页面功能安全的角度出发:
- 可扩展:



# 线下Bug测试模式

- 八大模式



# 多线程并发模式

- 模式解释
- 多线程创建，更新，删除某类数据以多线程方式(同时打开多个页面或浏览器或使用工具模拟)来操作功能数据并校验数据操作的原子性
- 应用场景
- 在涉及到管理操作的一些基础功能的时候考虑此测试模式，比如新增一个会员，编辑一个会员信息，删除一个会员信息等
- Bug案例
- 会员手机注册时；淘宝助理客户端多账号登陆时



# 搜索查询异常模式

## • 模式解释

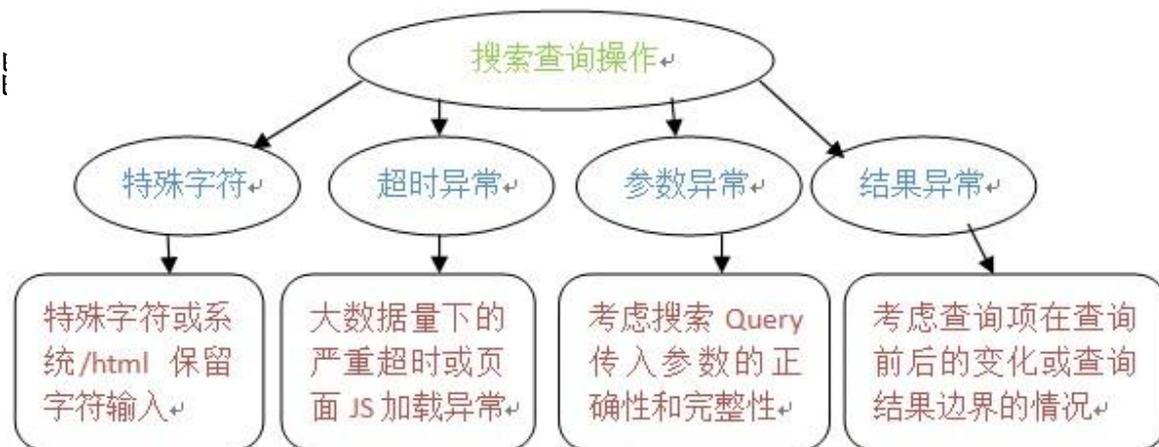
- 搜索或查询功能测试时，多考虑特殊字符进行查询
- 注意一些搜索条件可能出现在大数据量情况下的严重超时问题
- 查看搜索Query传入的参数的正确性和完整性，以及和搜索结果的对应关系

## • 应用场景

- 搜索或查询的管理操作功能

## • Bug案例

- 店铺留言管理页面
- 手机淘宝Wap商城搜索



# 线上故障的处理

- 线上故障怎么处理？
- 故障分析，How？

应用	bug状态	缺陷类型	bug级别	bug描述	bug原因分析	预防措施	后续跟进	遗漏原因分析
----	-------	------	-------	-------	---------	------	------	--------

- 分析后的作用？

# 线上故障的特点

- 环境更复杂
- 数据更全面
- 数据更特殊
- 故障原因更多

# 线上故障的原因

- 核心关注测试人员遗漏点

按遗漏原因分析报告



按缺陷类型分类

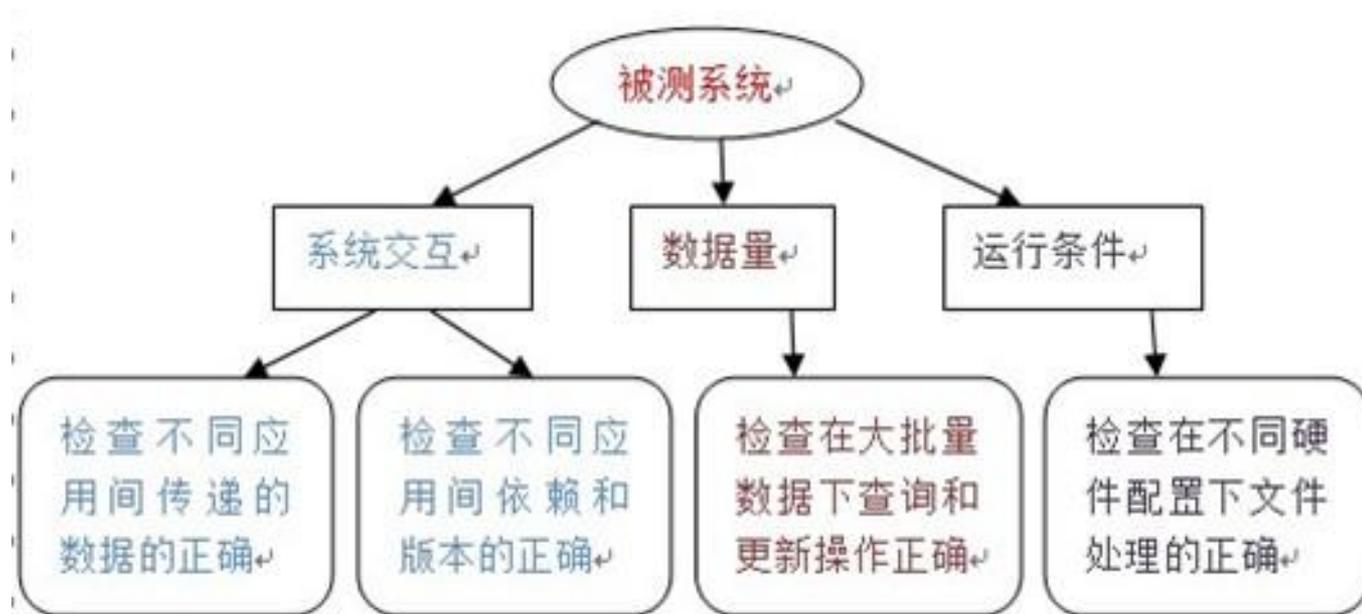
前端问题	11
后端功能错误	14
	25

按测试人员遗漏分类

本应用关联关系未评估到	1
本产品线回归未充分	1
跨产品线回归未充分	2
特殊场景遗漏测试	2
用户体验问题	1
	7

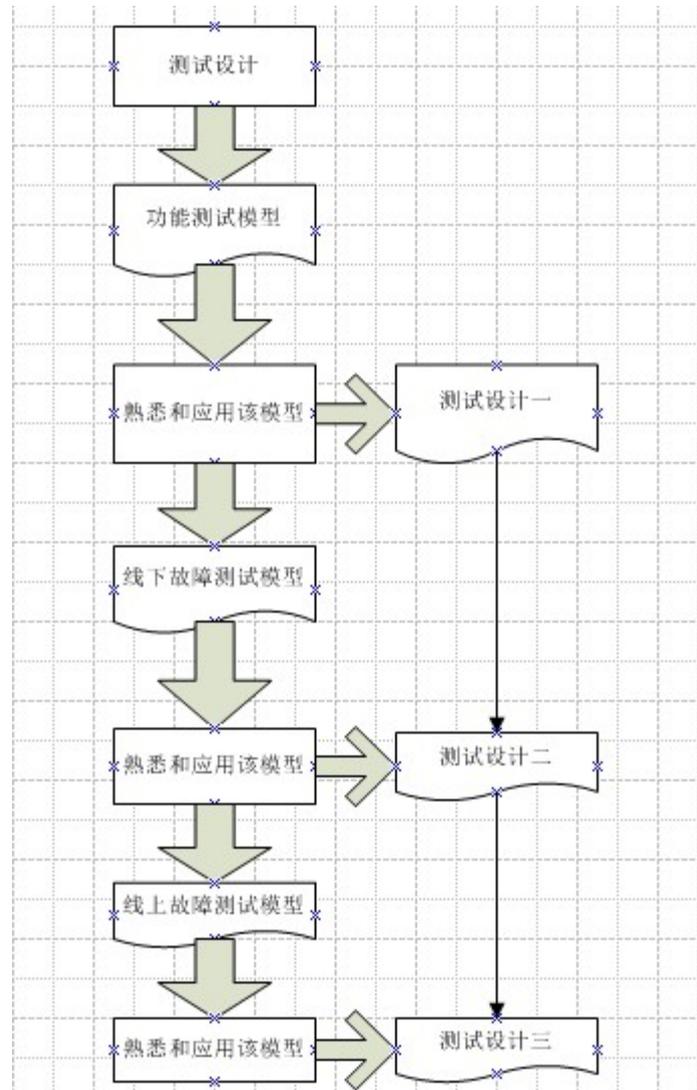
# 线上故障测试模式

- 场景测试的升级版



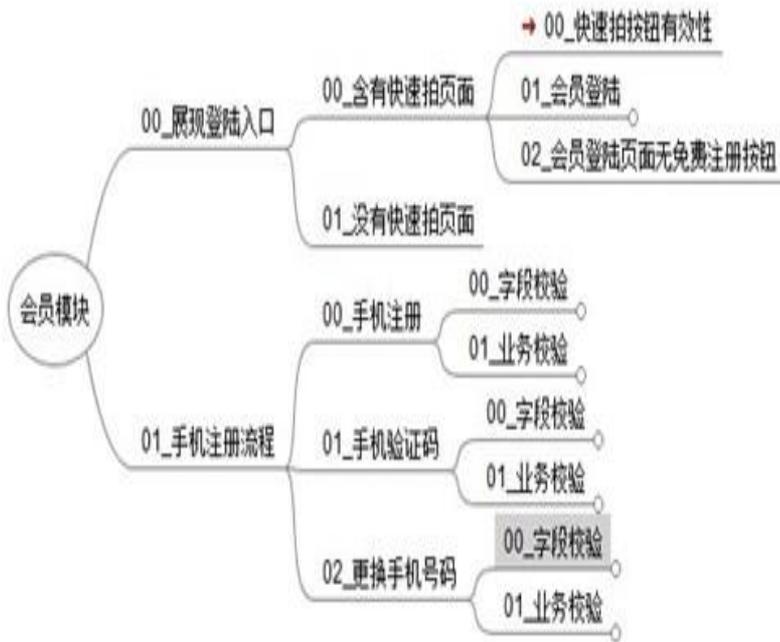
# 基于模式的测试设计

- 测试设计不是分析
- 测试设计是动态、持续的
- 保持怀疑精神
- 测试设计的信息来源：
  - 需求规格说明书
  - 系统概要设计
  - 详细设计和数据库设计
  - 实际产品和测试执行



# 基于模式的测试设计案例

## 用例架构和功能



# 基于模式的测试设计的注意点

- 正确理解测试设计
- 测试模式的建立和理解
- 循序渐进的过程
- 持续完善的过程
- 没有万能的测试模式
- 测试设计的评审：内审 + 外审



# Q&A



Thank You !!

- 联系方式:
- @季哥也是Jerry