

## 【测试标准/流程】

- 测试体系树
- 测试开发流程
- 开发-测试流程
- 测试阶段划分
  - 单元测试
  - 模块测试
  - 集成和系统测试
  - 验收测试
- 测试阶段准入准出标准
- BUG级别及定义

### 测试体系树

# 测试体系树

## 认识软件测试

软件测试的意义

软件测试的类别

测试方法划分

黑盒测试、白盒测试、灰盒测试

测试阶段划分

单元测试、冒烟测试、模块测试、集成测试、系统测试、验收测试、回归测试

测试内容划分

功能测试、性能测试、安全测试、兼容性测试、易用性测试等

测试所属方划分

α测试、beta测试

CMMI过程划分

验证测试、确认测试

按是否执行程序

静态测试、动态测试

软件测试生命周期

大致分为：需求分析、计划、方案、用例设计、测试流转、测试实施、测试总结

## 需求分析

了解用户需求（了解用户的真实诉求，更好地代入角色）

软件需求（需求规格说明书、原型）评审

测试需求提取（没需求的情况下，用例的提取）

测试需求分析——分解测试点

## 测试计划

测试时间节点确认（协同设计、开发共同制定）

测试计划制定

时间

资源（人力分配、软硬件、数据集等）

内容范围

测试策略（如执行优先级、执行方法说明等）

风险/问题

测试结束标准

预期产出

测试方案制定（计划补充）

通常复杂的业务、性能测试需要

## 设计

设计评审（概要设计、详细设计、UI效果图、数据库等）

用例设计

用例设计方法选型

边界值分析、等价类划分、业务场景测试、经验探索性测试

用例编写

用例编写规范

用例设计模板

用例评审

提取基础用例集

冒烟用例集

迭代checklist

通用用例集

用例维护

## 测试准备工作

项目涉及技术培训（包括在项目计划中）

测试环境配置

测试数据准备

## 转测

转测内容确认、分析

冒烟测试

## 测试实施

单元测试

通常为：编码规范性测试、代码覆盖率等

集成测试

接口联调测试

测试自动化

功能集成测试

系统测试

功能测试

性能测试

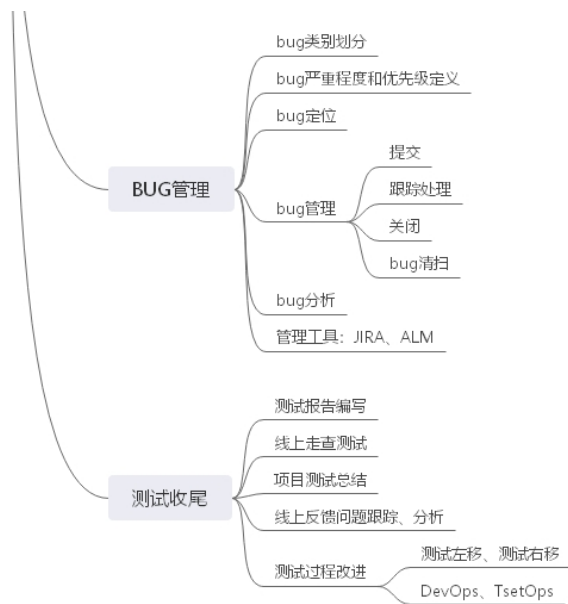
安全测试

兼容性测试

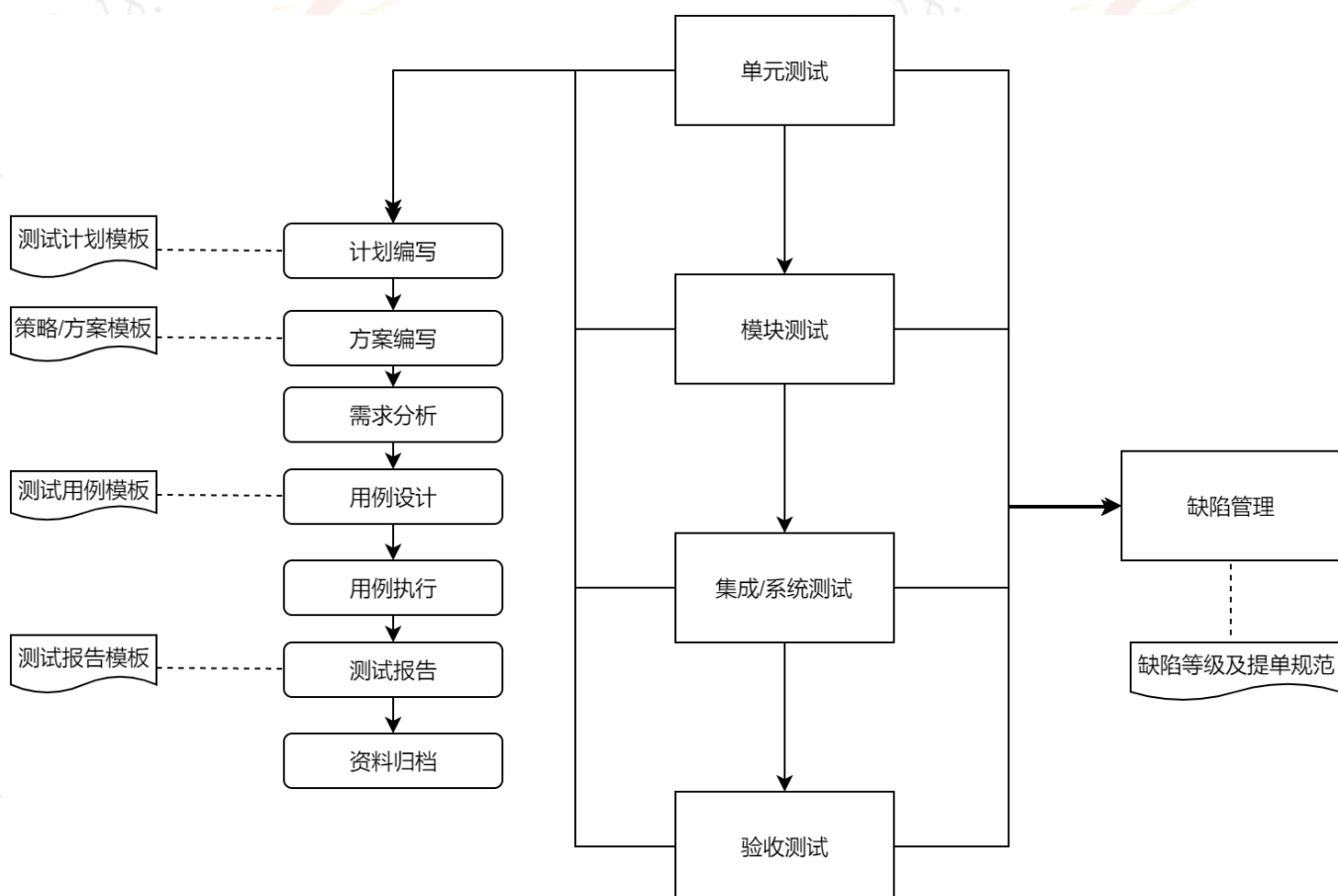
.....

验收测试

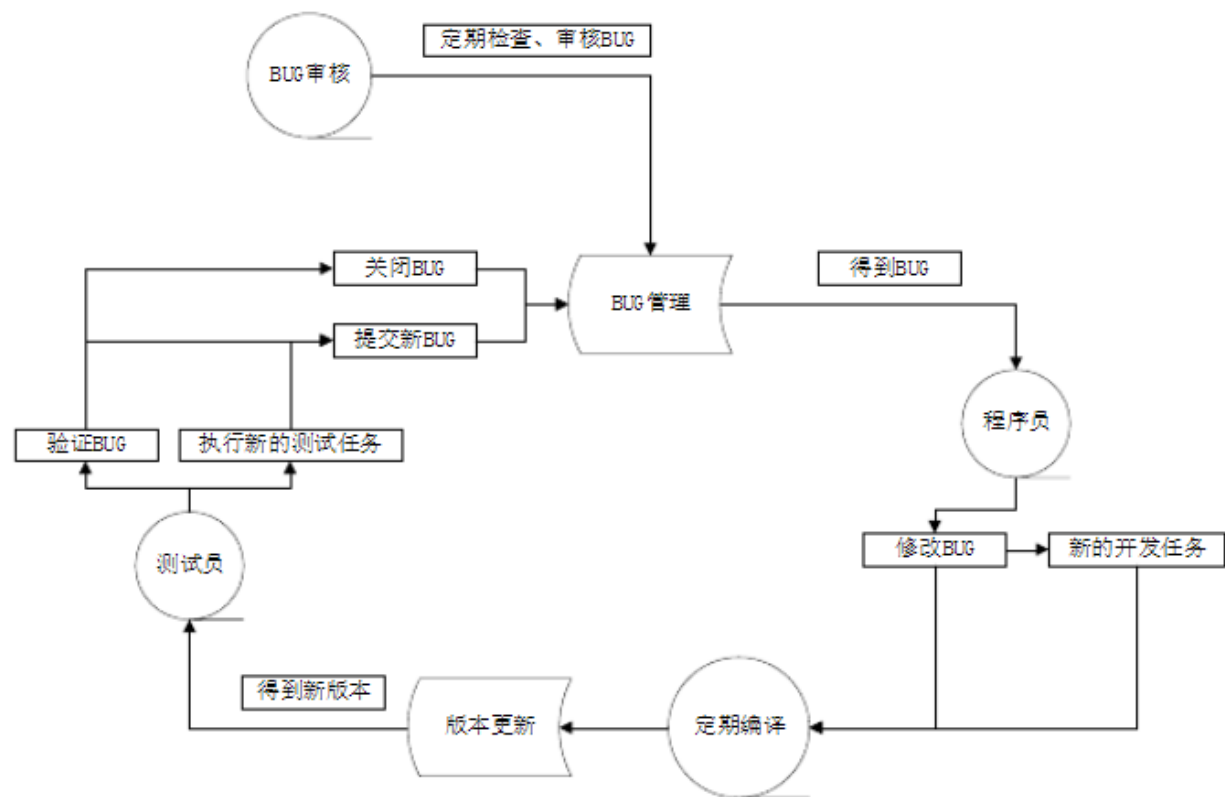
beta测试（灰度测试）



## 测试开发流程



# 开发-测试流程



## 测试阶段划分

### 单元测试

针对程序模块进行正确性检验的测试阶段。程序单元是应用的最小可测试部件，在过程化编程中，一个单元就是单个程序、函数、过程等；对于面向对象编程，最小单元就是方法，包括基类（超类）、抽象类、或者派生类（子类）中的方法。

### 模块测试

针对业务功能模块的正确性验证。

### 集成和系统测试

针对程序单元/模块等组件接口测试，系统功能/性能/压力/稳定性/精度/高可用等全面测试。

### 验收测试

根据产品规格说明书验收，一般包含内部Alpha和外部 Beta测试。

## 测试阶段准入准出标准

准入准出标准	单元测试	模块测试	集成/系统测试	验收测试
--------	------	------	---------	------

准入要求	需求函数/类等代码开发完成	1. 满足 <b>单元测试</b> 准入要求; 2. 测试策略/方案/用例ready;	1. 满足 <b>单元测试</b> 或 <b>模块测试</b> 准入要求; 2. 迭代版本提测说明: 开发自验证新增功能点覆盖率100%, 已修复JIRA问题单验证100%	1. 满足 <b>集成/系统测试</b> 准入要求; 2. 用户测试环境ready; 3. 用户验收指导手册ready;
准出要求	1. 测试结果满足度量指标: - 核心代码100%经过Code Review; - 核心模块代码覆盖率100% - 整体代码覆盖率>=80%; - CI执行通过率100%; 2. 输出单元测试报告	1. 测试结果满足度量指标: - 需求/功能点覆盖率100%; - 用例执行率100%; - 用例通过率(核心功能)100%; - 遗留缺陷指数: $\Sigma \leq 35^*$	1. 测试结果满足度量指标: - 需求/功能点覆盖率100%; - 用例执行率100%; - 用例通过率(核心功能)100%; - 遗留缺陷指数: $\Sigma \leq 35^*$	1. 基于Alpha测试满足度量指标(ad-hoc): - 产品规格覆盖率100% - 遗留缺陷指数: $\Sigma \leq 10^*$ - 满足算法评测等专项测试准入标准
		2. 模块报告(模板)评审通过;	2. 集成/系统测试报告(模板)评审通过	2. 验收测试报告
		3. 模块测试策略和方案归档;	3. 集成测试策略和方案归档;	
		4. 测试用例正式归档;	4. 测试用例正式归档;	

\*遗留缺陷指数:  $\Sigma \leq 35$ 为Major的BUG小于等于5个, Normal的BUG小于等于10个



以上关键数值仅供参考, 各产品线可根据自身产品形态及实际情况对数值进行合理调整。

## BUG级别及定义

序号	级别	说明	划分标准 (考虑平台类和业务类)	期望解决 时效	加权 评估 权重	Beta版本要求	RTM(对外) 发布要求
1	Blocker(阻塞)	阻碍开发和测试工作的问题, 如系统无法进入, 所有工作Blocked	系统不能启动, 或者死循环, 正常操作异常退出或崩溃等问题 重大功能缺陷, 导致主流程中断 系统无法执行, 应用模块无法启动 接口不通, 数据接口返回等错误 其它导致其它核心功能点无法开展相关测试, 比如核心测试项对其有依赖	1 Day (24 Hours)		Blocker=0	Blocker=0
2	Critical(致命)	严重影响功能实现及用户使用的问题	非关键路径程序崩溃(需要一定条件和步骤), 不影响其他测试及路径 严重内存泄露 用户数据丢失或破坏 功能未实现, 有错误 数据错误, 不同步 用户权限问题 安全问题比如高危安全漏洞 严重的数值计算统计错误 核心功能not work, 核心功能实现与需求设计严重不符 性能规格低于设计规格80% 准确率偏离预期百分比大于5% 非常影响用户使用的UI问题, 如屏幕切断, 花屏等	3 Days (TBD)	10	Critical=0	Critical=0
3	Major(严重)	主要功能存在缺陷, 但不影响到系统稳定性, 但用户使用中会存在明显感知	功能实现不完整, 或主要功能点not work 主要功能实现不符合需求, 比如开发实现和产品定义不一致的问题(功能流程实现) 未考虑边界值引起的设计缺陷, 比如接口参数未按说明进行校验且不会导致Critical及以上的问题 关键流程中的错误提示/或者没有提示 性能达到预期80%, 但未达产品设计规格 准确率偏离预期百分比大于2% 需要特殊步骤/条件才能复现的bug	5 Days (TBD)	5	Major<=15	Major<=5

			系统日志系统不完善，导致系统异常后无法获取到完整日志			
			系统实现的界面上，一些可接受输入的控件点击后无作用，对系统功能不能正确实现			
			低概率出现的critical问题，比如较难复现的崩溃问题			
4	Normal (一般)	功能未完全实现但用户使用影响不大，或问题有简单的规避方案	真实场景中边缘、低频使用的功能性问题	版本发布前解决	1	未明确定义，所有Bug需经过产品、测试、交付&售后评审
			重要操作无日志记录			
			边界条件错误			
			重要操作未提示或者提示信息不完整或错误			
			操作页面错误，比如光标跳转定位错误			
			超链接跳转的按钮不明显			
			颜色搭配在某些页面风格突变			
			影响范围有限的兼容性问题等			
5	Trivial (建议)	错别字，易用性及建议性等问题	错别字，文字排列/排序等	版本发布前解决	0.1	未明确定义，所有Bug需经过产品、测试、交付&售后评审
			界面格式不规范			
			辅助说明描述不清楚			
			操作时未给用户友好提示			
			可输入区域和只读区域没有明显区分标注			