

# 测试理论第一天笔记

## 今日目标

- 了解测试相关流程
- 知道测试用例的构成
- 能够用等价类划分法设计测试用例

## 测试回顾

- 软件测试：通过**手工**或者**自动化**的方式运行被测的软件是否正常（看预期结果和实际结果是否一致）
- **测试目的：保障软件的质量（尽可能多的发现系统中的错误，证明软件存在问题）**
- 测试体现形式：通过找出bug的形式验证质量
- 测试人员具备的素质（521）：

应用：简历的自我评价里面可以应用

- 5个心：责任心、细心、耐心、专心、自信心
- 2个能力：沟通能力、表达能力
- 1个精神：团队协作精神

## 软件的质量

### 前置铺垫

- 需求：用户的想法，为了实现某个目的而产生的想法
- 需求规格说明书：将用户的想法转化为技术上可以实现的文档

### 软件质量模型

应用场景：提供对于软件产品从测试角度思考的一种思路

- 定义：实际实现的产品和需求描述是否相一致，相一致程度高说明质量满足需求（好）
- 如何评判软件的质量？
  - 功能：软件产品是否具备某种能力

某手机是否支持5G

- 性能：软件产品对于时间和空间的占用程度高低

速度快、占用空间小

- 兼容性：软件兼容其他类型的软硬件的能力

著名的"3Q"大战

- 易用性：在**一定用户群**的基础上，软件是否好用、容易理解

遵循专业性，例如：财务系统软件（账单、流水等）

- 可靠性：软件是否具备持续无故障运行的能力
- 安全性：软件运行过程中对于数据的传输和存储是否安全

属于专项测试，要求较高

- 可移植性：软件产品从一个环境移植到另一个环境中正常运行的能力
- 可维护性：软件出现故障后，自我修复/恢复的能力

## 软件的生命周期

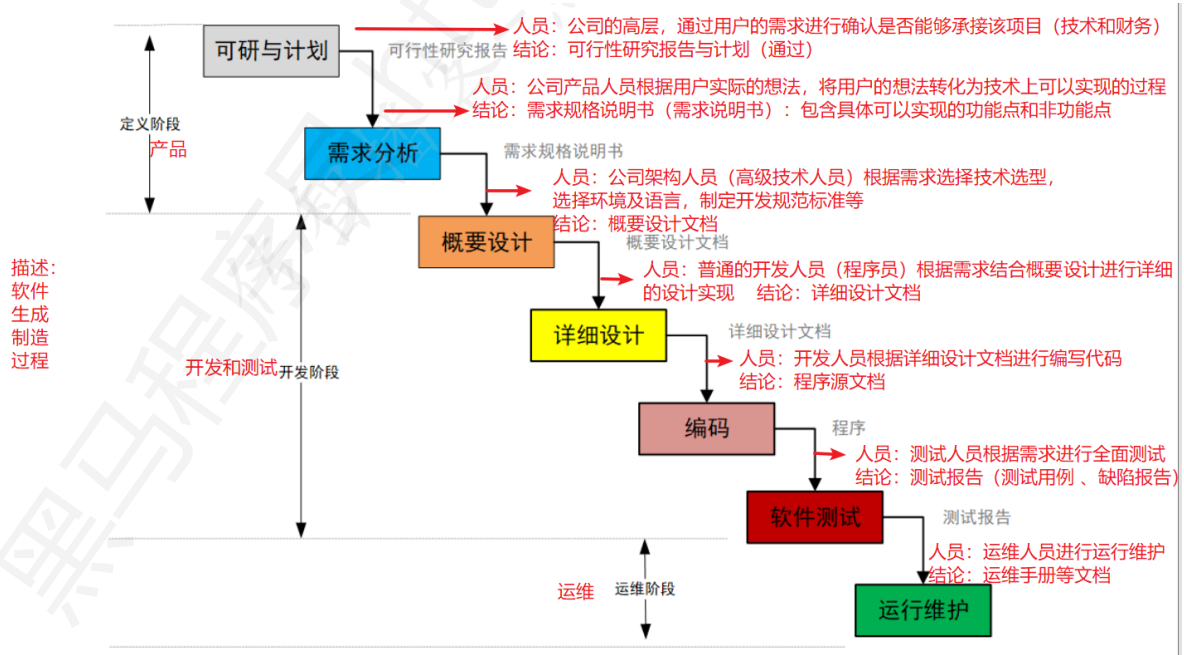
软件生命周期：软件从无到有到消亡的过程

软件生命周期也叫软件开发过程模型、软件生命周期模型

## 瀑布模型

描述软件生成到消亡的过程模型图

- 该模型目前实际工作中已不常用，但是该模型是其他新型模型的“鼻祖”



## 瀑布模型的优点

- 每个阶段比较清楚，并且有对应的文档产生
- 当前一个阶段完成后，才开始后面的阶段（一次性的）

## 瀑布模型的缺点

- 发现问题的时机比较晚，失去提前纠错的机会
- 测试介入比较晚

## 适用场景

- 适用于需求不易发生变化的大项目

## 【扩展】敏捷开发模型

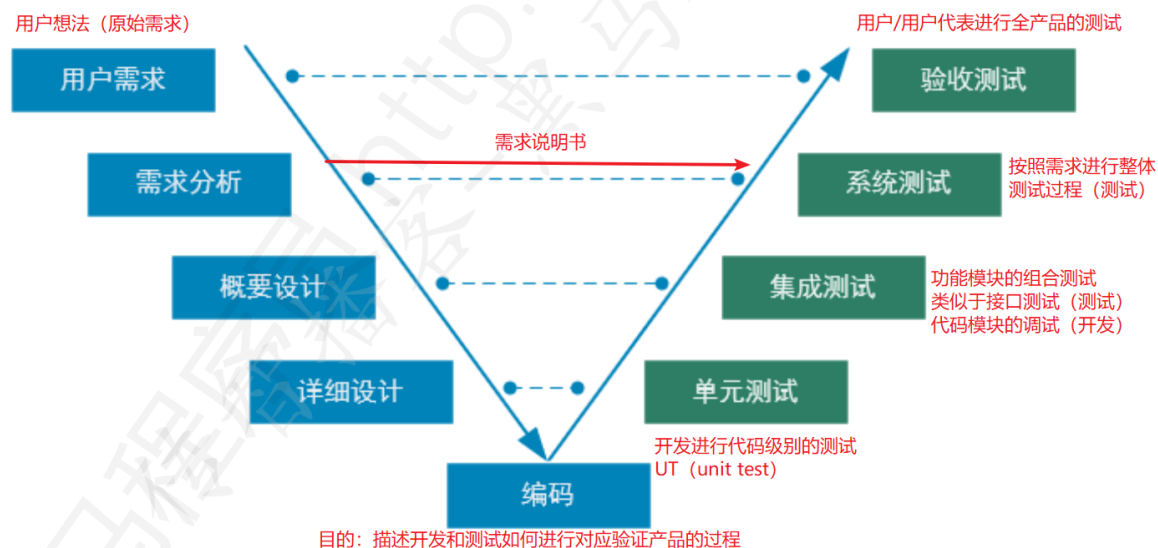
能够适用需求的变化，并且能够给出快速的响应

- 小步快跑
- ACP

## 软件测试模型

### V模型

作用：主要描述测试、开发之间的对应关系



### V模型优点

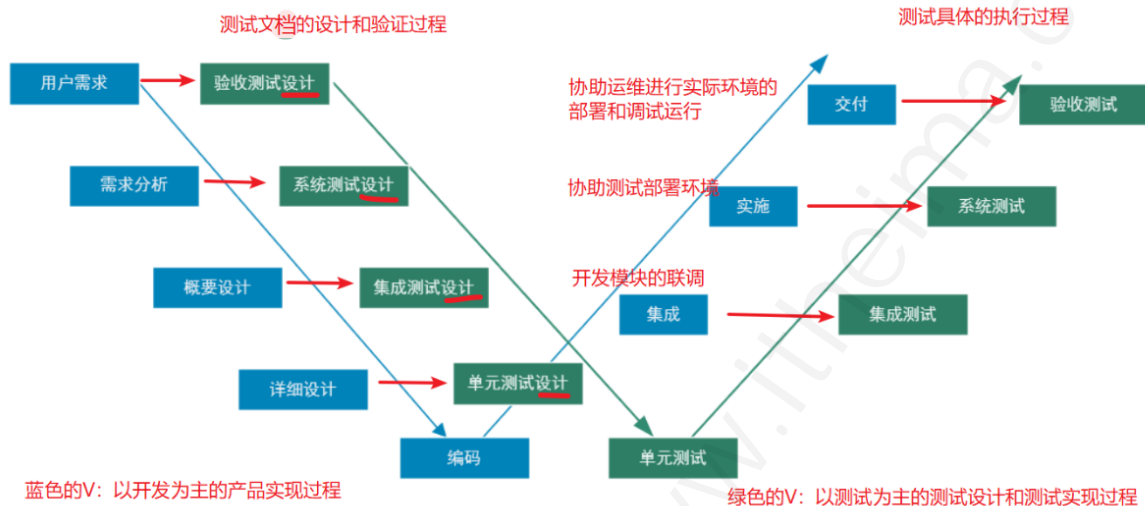
- 每个阶段比较清楚，测试过程由底层（代码）测试到高层（应用）测试过程

## V模型缺点

- 不适用于需求的变更，发现问题的时机比较晚

## W模型

作用：将测试过程更加细化说明，对应测试、开发之间的关系更加清楚



## W模型优点

- 测试介入时间早，能够及时发现问题，降低修复成本
- 测试伴随整个软件生产周期，除了测试软件之外，还需要验证文档

## W模型缺点

- 该模型应用起来复杂度高（具备计算机技能、业务能力、管理能力、测试素质）

## 测试用例

目的：

- 方便测试验证（将需求大量描述拆分为小的测试点）
- 体现测试人员的思路，测试设计的全面性（后续测试直接可以使用）
- 测试的量化体现，能够反应测试进度

## 定义

测试用例，也叫Test Case，为了特定的目的而设计的一组测试输入，执行条件和预期结果构成的文档。

编号	目的	前提条件	操作步骤	希望结果
1	验证开机	设备有电	按下开机键	能够开机
2	验证运行速度	设备有电	运行大软件	流畅不卡顿

## 构成要素

规范的测试用例应该包含哪些内容？

注意：实际工作中，如果企业中有自己的用例模板，则用自己公司的即可，核心内容基本一致

- ✓ 用例编号：表示用例的唯一性，有时也叫用例ID → 变化构成推荐：字母+数字。 eg: login-001 登录-001
- ✓ 用例标题：表示要测试或验证的目的，通常一句话简要描述 → 编写：验证XXXX是否正常 要求：写准确
- 测试项目：当前测试的功能所属范围 → 简单描述：eg:登录模块 或者写项目名称即可
- ✓ 用例级别：表示用例的重要程度或者影响力 → 描述：高 中 低 推荐用英文简称：P1 .... (priority)
- 预置条件：验证该功能需要的前提条件 → 注意：必要的前提条件（如果不写可能会有歧义）
- 测试输入：必要的输入数据 → 有数据输入的测试，需要准备测试数据
- ✚ 执行步骤：验证该功能需要的先后操作步骤 → 具体测试的先后次序，一般通过序号+文字描述：1.xxxx 2.xxx
- ✚ 预期结果：希望得到的结果 → 一般是结论+（现象） eg:能正常开机，进入桌面操作系统

测试用例模板

用例编号	用例标题	测试项目	用例级别	预置条件	测试输入	执行步骤	预期结果

## 测试用例设计方法

如何写好测试用例？

### 等价类划分法

- 等价类定义
  - 在批量的测试数据中，选取具有共同特征的数据子集作为测试的输入
- 等价类的分类
  - 有效等价类：满足需求的测试数据
  - 无效等价类：不满足需求的测试数据

### 等价类划分法设计用例步骤

案例：如何测试两个两位数整数之间的和（即-99到99之间数据求和）没有问题？

- 明确需求
  - 测试目的

eg：验证两位数整数是否能够正常求和

- 测试条件

eg :

- 不超过两位数
- 整数、字母、中文、特殊符号、空格、空

- 长度
- 类型（来源于键盘输入）
- 规则

- 划分等价类

- 有效等价类：满足需求的所有条件

eg : -10 , 30

- 无效等价类：不满足（只要不满**其中一个**条件即可）需求的所有输入数据

eg : 包含字母、中文、符号、空格、空

- 提取数据编写用例
  - 通过测试用模板按照要求填写内容

## 等价类适用场景

- 针对有**批量数据**输入的测试场景，无法穷举测试时适用
- 常见代表：输入框（下拉框、选择框、下拉列表等）

## 编写用例注意事项

- 单模块测试中，用例标题具有唯一性
- 必要步骤尽可能清楚
- 预期结果尽量描述测试结论性的语句及现象（如果有具体现象最好描述）
- 用例编号和测试项目简称对应设置

## 案例练习

- 参见课堂资料Excel文档