

软件测试分类

XMind
试用模式



- 1. 性能测试 (Performance Testing)
 - a. 检查系统是否满足需求规格说明书中规定的性能
 - b. 通常表现为以下几个方面:
 - 稳定性: 软件不崩溃, 无论是否高发节点。
 - 响应时间: 2/8原则。2秒体验很好, 5秒可接受, 8秒不可忍受。
 - 吞吐量 (TPS): 客户端与服务器之间的信息交互。
- 2. 安全测试 (Safety Testing)
 - 安全测试是一个相对独立的领域, 需要如: WEB安全测试、网络协议、防火墙、CDN、熟悉各种操作系统的漏洞、熟悉路由器等。(学习困难)
- 3. 兼容性测试 (Compatibility Testing)
 - 主要至软件之间能否很好运作, 会不会有影响、软件和硬件之间能否发挥很好的效率工作, 会不会影响导致操作系统的崩溃。
 - 各个平台兼容性测试
 - 浏览器兼容性测试
 - 软件向前或向后兼容
 - 测试软件是否与其他相关软件兼容
 - 数据兼容性测试

- 1. 静态测试 (Static Testing)
 - 静态方法指不运行被测程序本身, 仅通过分析或检查程序的语法、结构、过程、接口等来检查程序的正确性, 对需求规格说明书、软件设计说明书、源程序做结构分析、流程图分析、符号执行来找错。
 - 注意: 静态测试属于白盒测试, 白盒测试不等于静态测试。
- 2. 动态测试 (Dynamic Testing)
 - 动态测试是指通过运行软件被测程序, 检查运行结果与预期结果的差异。

- 1. 黑盒测试 (Black-Box Testing)
 - 也称为功能测试。测试中把软件当作黑盒子, 不关心内部结构, 只看软件的输入和输出数据。
 - 等价类、边界值、场景法、判定表、因果图、错误推断法、正交实验法
- 2. 白盒测试 (White-Box Testing)
 - 也称为结构测试、逻辑驱动测试或基于代码的测试。白盒测试指打开盒子研究源代码和程序结果。
 - 语句覆盖、判定覆盖、条件覆盖、判定条件覆盖、路径覆盖
- 3. 灰盒测试 (Gray-Box Testing)
 - 介于白盒测试和黑盒测试之间的一种, 会和测试多用于集成测试阶段, 不仅关注输入、输出的正确性, 同时也关注程序内部的情况。

- 1. 手工测试 (Manual Testing)
 - 概念: 手工测试是由人一个一个的输入用例, 然后观察结果, 和机器测试相对应, 属于比较原始但是必须的一种。
 - 不可被完全替代
- 2. 自动化测试 (Automation Testing)
 - 概念: 在预设条件下运行系统或应用程序, 评估运行结果。(预先条件包括: 正常条件和异常条件)。简单来说, 自动化测试就是把人为驱动测试行为, 转化为机器执行的一种过程。

- 1. 单元测试 (Unit Testing)
 - 概念: 也称作模块测试。对软件的组成单位进行测试, 其目的是检验软件基本组成单位的正确性。测试对象为软件测试最小单位: 模块
- 2. 集成测试 (Integration Testing)
 - 概念: 也称作联合测试、组装测试; 将程序模块采用适当的继承策略组装起来, 对系统的接口及继承后的功能进行正确性检验的测试工作。
 - 目的: 检查软件单位之间的接口是否正确
- 3. 系统测试 (System Testing)
 - 概念: 将软件系统看成一个系统的测试。包括功能、性能以及软件运行的软硬件环境进行测试。时间大部分在系统执行阶段, 包括回归测试和冒烟测试
- 4. 验收测试 (Acceptance Testing):
 - 概念: 部署软件之前的最后一个测试操作。它是技术测试的最后一个阶段, 也成为交付测试。向软件购买者展示软件系统满足原始需求

- 1. α 测试 (Alpha Testing)
- 2. β 测试 (Beta Testing)
- Alpha与Beta测试的区别:
 - a. 测试场所的不同: Alpha测试是指把用户请到开发方的场所来测试, Beta测试是指在一个或多个用户的场所进行的测试。
 - b. Alpha测试的环境是受开发方控制的, 用户的数量相对较少, 时间比较集中。Beta测试的环境是不受开发方控制的, 用户数量相对比较多, 时间不集中。
 - Alpha测试优先于Beta测试执行。通用的软件产品需要较大规模的Beta测试, 测试周期比较长。

- 1. 国际化测试 (International Testing)
 - 软件的国际化 and 软件的本地化是开发面向全球不同地区用户使用的软件系统的两个过程。而本地化测试和国际化测试则是针对这类软件产品进行的测试。由于软件的全球化普及, 还有软件外包行业的兴起, 软件的本地化和国际化测试俨然成为了一个独特的测试专门领域
- 2. 本地化测试 (Localization Testing)

- 1. 文档测试 (Document Testing)
 - 开发文件
 - 可行性研究报告
 - 软件需求说明
 - 数据要求说明书
 - 概要设计书
 - 详细设计说明书
 - 数据库设计说明书
 - 模块开发卷宗: 每个模块之间的说明
 - 用户文件
 - 1. 用户手册
 - 2. 操作手册
 - 用户文档的作用:
 - 改善易安装性
 - 改善软件的易学性与易用性
 - 降低技术支持成本
 - 3. 管理文件
 - 项目开发计划
 - 测试计划
 - 测试分析报告
 - 开发进度月报
 - 项目开发总结报告
 - 注: 在实际测试中, 最常见的就是用户文件的测试, 例如: 用户操作说明书的测试等。
 - 文档测试关注的点:
 - 文档的术语
 - 文档的正确性: 语句语病、文字错误, 描述错误, 描述不清等
 - 文档的完整性: 基本流的完整性, 闭环。
 - 文档的一致性: 文档与软件本身一致。
 - 文档的易用性: 易操作

- 5. 易用性 (用户体验性测试) (User Ability Testing)
 - 易用性是交互的适应性、功能性和有效性的集中体现。又叫用户体验测试
- 6. 界面测试 (User Interface Testing)
 - 界面测试 (UI测试), 测试用户界面的功能模块的布局是否合理、整体风格是否一致、各个控件的放置位置是否符合客户使用习惯, 此外还要测试界面操作便携性、导航简单易懂性, 页面元素的可用性, 界面中文字是否正确, 命名是否统一, 页面是否美观, 文字、图片组合是否完美等
- 7. 安装测试 (Installation Testing)
 - 测试程序的安装、卸载。最典型的就是APP的安装和卸载

- 一、非功能测试
- 二、功能测试