

# 如何保证软件测试的质量

## 一、软件测试的流程

我们知道软件测试的流程大致如下：

1. 需求：阅读需求，理解需求，与客户、开发、架构多方交流，深入了解需求。
2. 测试计划：根据需求估算测试所需资源（人力、设备等）、所需时间、功能点划分、如何合理分配安排资源等。
3. 用例设计：根据测试计划、任务分配、功能点划分，设计合理的测试用例。
4. 执行测试：根据测试用例的详细步骤，执行测试用例。
5. 执行结果记录和 bug 记录：对每个 case 记录测试的结果，有 bug 的再测试，管理工具中编写 bug 记录。
6. defect tracking：追踪 leader 分配给你追踪的 bug，直到 bug fixed。
7. 测试报告：通过不断测试、追踪，直到被测软件达到测试需求要求，并没有重大 bug。
8. 用户体验、软件发布等。

## 二、软件测试实际情况

软件测试对于软件设计来说是很重要的一个环节，它涵盖了软件的全部细节，比如软件的实用性、操作性、稳定性、安全性、简便性等等。可以说没有经过详细的测试的软件绝对不会成为一个优质的软件，要保证软件质量，必须要保证有严格的软件测试过程。

但是实际情况中往往不尽如人意，常常会遇到时间紧张，任务重的项目，在这种情况下，如何保证测试质量，我这里讲几点见解，希望老师批评指正：

1. 项目成员明确需求，需求按优先级排序，评审之后少做变更。

需求是源头，PD，开发，测试前期评审把好关，越早发现问题，越容易解决，花费的代价越小。要做到需求按优先级排序，把需求分解成具体的最小级别的功能点，先实现高优先级的需求。三方评审通过后，项目中冻结需求，尽量少做变更。

2. 制定合理的测试计划，明确里程碑时间和负责人

测试计划是指导测试行动的总纲领，规划好测试设计，用例编写，测试执行的时间，测试负责人每天关注进展，及时调配资源，将问题解决在萌芽状态。

3. 保证测试设计和用例的质量

资深的测试工程师负责测试设计；按测试组成员能力水平分配任务，完成用例设计。完成之后，进行测试组和项目组的评审，查漏补缺。

#### 4. 提高测试介入的标准

时间紧张，需要开发保证代码质量，测试介入的标准肯定是必须通过冒烟测试。冒烟用例评审时一定找开发确认，开发自己先执行成功冒烟用例，保证测试介入后能顺利走下面的流程。

#### 5. 迭代测试

开发迭代提交模块，测试针对性进行测试。迭代测试增加了测试时间但是并没有延误整个的时间进度，因为在每一个迭代过程中测试过程都是提前开始的。

#### 6. 每天召开晨会，沟通项目进度，解决问题

介入测试时，开发，测试，PM 等团队成员每天花半小时召开晨会，沟通各自的进展，列出项目中的问题，确认解决人和解决时间。问题及时解决，会加深团队伙伴的信任，激发工作热情。

当然，测试工作最终还是基于代码质量的，当我们发现低估了项目复杂度的时候，增加测试时间才是明确的选择。欲速则不达，着急冒进，项目的质量很难得到保障，压缩的时间迟早会补偿回去。

因此制定合适的测试过程规范，制定合理的测试计划，设计合适的测试方案，编写的测试用例覆盖到所有的需求，对测试执行过程进行监控，使用工具管理测试发现的缺陷，对缺陷进行统计分析，指导过程改进。

### 三、接下来浅析如何提升软件测试质量

随着软件测试受关注程度越来越高，如何采用技术手段有效提高软件测试质量就成了软件测试领域的一个重要课题。接下来从软件测试的基本概念开始，对如何以软件测试性设计为中心、合理运用软件测试技术来提升软件测试质量提出了自己的看法。

软件产品的质量取决于软件开发过程，软件测试作为软件生存期中的一个重要阶段，受重视程度越来越高。软件测试是保证软件质量和可靠性的关键步骤，也是用来验证软件是否能够完成所期望功能的唯一有效的方法。测试已不仅仅局限于软件开发中的一个阶段，它已开始贯穿整个软件开发过程，进行测试的时间越早，整个软件开发成本下降就越多。大量统计表明，软件测试的工作量往往占到软件开发总量的 40% 以上，在极端的情况下，甚至可能高达软件工程其它步骤成本总和的三至五倍，其目的是尽可能的提高软件产品的质量和可靠性。

#### 1、软件测试相关概念

1) 软件测试：软件测试是为了发现错误而执行程序的过程。或者说，软件测试是根据软件开发各阶段的规格说明和程序的内部结构而精心设计出一批测

试用例，并利用这些测试用例的运行结果来发现程序错误的过程。

2) 软件测试用例：测试用例实际上是对软件运行过程中所有可能存在的目标、运动、行动、环境和结果的描述。测试用例是测试组织的最小单位，指对一项特定的软件产品进行测试任务的描述，体现测试方案、方法、技术和策略。内容包括测试目标、测试环境、输入数据、测试步骤、预期结果、测试脚本等，并最终形成文档。

软件测试的核心是设计和执行测试用例。而测试用例的选择问题可以看作是从庞大的输入状态组合中，搜寻那些可以发现错误的状态组合。因此需要用抽象的手段来尽量使测试更加有效。

3) 测试用例库：完整的单元测试很少只执行一个测试用例，开发人员通常都需要编写多个测试用例才能对某一软件功能进行比较完整的测试，这些相关的测试用例称为一个测试用例集。将大量的测试用例收集到测试用例库中，合理的分类后供测试人员选择使用，能够极大地提高软件问题的发现率。

## 2、提高测试质量的方法

### 2.1 采用测试性设计技术

软件测试是目前用来验证软件是否能够完成所期望的功能的唯一有效的方法。但是在测试的实施过程中，由于种种原因导致测试的难度相当大，甚至出现了无法测试的情形。为了提高软件的可测试性，我们在软件设计时应当遵循测试性设计原则，通过改变设计或代码、为软件增加专门测试结构等方法来提高软件的可测试性。

1) 测试驱动设计。这种设计就是直接把软件需求变成测试代码。在确定软件测试性能要求的基础上优先编写测试代码。先写验收测试，再写单元测试，并在开发过程中不断修正。

2) 每个操作对应一个方法，使得方法小型化。使用小型化方法说明和重载带有缺省方法参数的方法，使得测试中调用这些方法变的很容易。

3) 显示与控制分离。把代码移到 GUI 视图的外面，各种 GUI 动作就能成了模型上的简单方法调用。这样，在修改程序功能不会影响视图，同时通过方法调用测试功能也比间接地测试功能更容易。

4) 对于可能要作为参数的类，做一个接口。用接口说明外部程序组件或在需要时改变接口形成一个空类，作为参数传入。

### 2.2 选择合适的测试管理模型

模型是系统功能的形式化或半形式化的表示，支持输入状态组合的系统枚举。基于模型的测试主要考虑系统的功能，可以认为是功能测试的一种。测试模型体现了被测试系统的最本质的功能关系。而且要比系统本身更易于开发和分析。一个可测试的模型要能提供足够的信息用来产生测试用例。所以可测试的模型必须

满足以下要求：

- 1) 必须是某种测试实现的完全准确的反映，模型必须表示要检查的所有特征；
- 2) 是对细节的抽象；
- 3) 可以表示所有事件和所有的动作；
- 4) 可以表示系统的各种状态，以便由可知的方法来确定已达到或没有达到什么状态。

### 2.3 采用合理的测试技术

软件测试方法众多，各有优点，但任何一种测试方法都无法覆盖所有测试的需求。这就需要充分掌握软件测试的各种方法，熟悉其优缺点，根据系统需求精心设计组合。我们可以从以下两个方面展开：

#### 2.3.1 从代码的特性角度出发展开测试：

- 1) 单元测试：按照代码的单元组成逐个进行测试。
- 2) 功能测试：按照软件的功能或特性逐个进行测试。
- 3) 系统测试：对完整的代码进行编译和连接，以检查程序的主要功能能否达到预期目标。
- 4) 回归测试：对以前修复过的 **Bug** 重新进行测试，看该 **Bug** 是否会重新出现。值得注意的是，回归测试并不是软件测试的一个独立阶段。

#### 2.3.2 从用户的角度出发展开测试：

- 1) 配置测试：从用户的使用出发进行多方面的测试。
- 2) 兼容性测试：主要考虑软件和操作系统的兼容性问题。
- 3) 压力测试：在各种极限情况下对产品进行测试，以检查产品的长期稳定性。
- 4) 性能测试：测试是保证程序具有良好的性能，能否达到预期的性能指标。
- 5) 文档和帮助文件测试：对文档和帮助文件进行检测，保证用户可以通过学习文档和帮助文件正常使用产品。
- 6) Alpha 和 Beta 测试：在正式发布产品之前将软件测试版发送给用户，让用户在使用中找到能够存在的 **Bug** 或者反馈相关信息，以便在正式版中得到解决。

### 2.4 建立可复用的测试用例库

软件测试过程中产生的大量测试用例对保证软件的质量起到关键作用，测试用例的复用就显得尤其重要。在测试中应当构造出基于复用的测试用例，并采用测试用例库管理的方法来实现，这样可以大大提高软件测试的工作效率。

- 1) 基于复用的目的，对所使用的测试用例进行统一的建模组织，有效地将测试用例收集到测试用例库中，并按照行业项目等进行多级合理的分类、组织、

存储。对采用不同方式描述的测试用例，将分别实现不同程度的复用。

2) 对数据库中存有的测试用例实行有效管理，通过提供有助于复用的多种查询方式，确保测试用例的复用程度，对不同类别的测试用例进行复用频率的统计，为测试人员提供有效的复用信息，在保证软件测试质量的前提下，提高了测试的效率。

3) 对数据库中的测试用例具体实现复用，通过查询出所需的测试用例，得到测试用例信息，针对不同描述方式的测试用例采用不同的复用方式进行复用。

### 3、结语

这小节从技术的角度提出了一种以软件测试性设计为中心，配合合理的软件测试方法来实现提高软件测试质量的方法。

## 四、总结

总的说来，保证软件测试的质量需要从多方面考虑，比如测试的流程、实际项目的特殊情况。另外值得说明的是：软件测试人员的水平高低会严重的影响测试质量，这需要我们在实际工作中特别加以注意。

姓 名： 刘红华  
学 号： 14126117  
联系方式： 13120184513