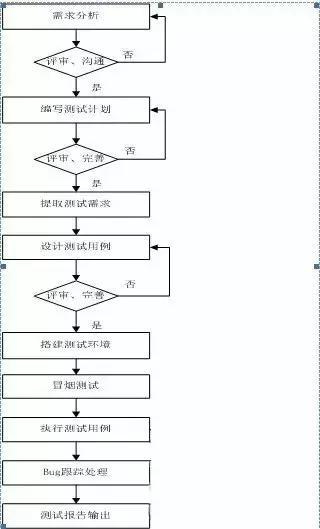
# **软件测试基本流程与要求**

软件测试的基本流程是希望通过规范化、标准化的流程，让软件测试可以变得高效，下面我们来看一下测试流程图。



**释义：测试需求分析**

测试需求是整个测试过程的基础；确定测试对象以及测试 工作 的范围和作用。用来确定整个测试工作（如安排时间表、测试设计等）并作为测试覆盖的基础。而且被确定的测试需求项必须是可核实的。

即，它们必须有一个可观察、可评测的结果。无法核实的需求不是测试需求。所以我现在的理解是测试需求是一个比较大的概念，它是在整个测试计划文档中体现出来的，不是类似的一个用例或者其他.

·测试需求是制订测试计划的基本依据，确定了测试需求能够为测试计划提供客观依据；

·测试需求是设计 测试用例 的指导，确定了要测什么、测哪些方面后才能有针对性的设计测试用例；

·测试需求是计算测试覆盖的分母，没有测试需求就无法有效地进行测试覆盖；

**释义：测试方法与规范**

1、测试方法

随着软件技术发展，项目类型越来越多样化。根据项目类型应选用针对性强的测试方法，合适的测试方法可以让我们事半功倍。

以下是针对目前项目工程可以参考的测试方法：

**--β测试（beta测试）**--非程序员、测试人员

β测试，英文是Betatesting。又称Beta测试，用户验收测试（UAT）。

β测试是软件的多个用户在一个或多个用户的实际使用环境下进行的测试。开发者通常不在测试现场，Beta测试不能由程序员或测试员完成。

当开发和测试根本完成时所做的测试，而最终的错误和问题需要在最终发行前找到。这种测试一般由最终用户或其他人员完成，不能由程序员或测试员完成。

**--α测试（Alpha测试）**--非程序员、测试人员

α测试，英文是Alphatesting。又称Alpha测试.

Alpha测试是由一个用户在开发环境下进行的测试，也可以是公司内部的用户在模拟实际操作环境下进行的受控测试，Alpha测试不能由该系统的程序员或测试员完成。

在系统开发接近完成时对应用系统的测试；测试后，仍然会有少量的设计变更。这种测试一般由最终用户或其他人员来完成，不能由程序员或测试员完成。

**--兼容性测试**--测试人员

兼容性测试是指测试软件是否可以成功移植到指定的硬件或者软件环境中，例如在B/S项目中各个不同浏览器之间的测试。

**--用户界面测试-UI测试** --测试人员

用户界面测试，英文是User interface testing。又称UI测试。

用户界面，英文是User interface。是指软件中的可见外观及其底层与用户交互的部分（菜单、对话框、窗口和其它控件）。

用户界面测试是指测试用户界面的风格是否满足客户要求，文字是否正确，页面是否美观，文字，图 片组合是否完美，操作是否友好等等。UI 测试的目标是确保用户界面会通过测试对象的功能来为用户提供相应的访问或浏览功能。确保用户界面符合公司或行业的标准。包括用户**友好性、人性化、易操作性** 测试。

用户界面测试用户分析软件用户界面的设计是否合乎用户期望或要求。它常常包括菜单，对话框及对 话框上所有按钮，文字，出错提示，帮助信息 (Menu 和Help content)等方面的测试。比如，测试Microsoft Excel中插入符号功能所用的对话框的大小，所有按钮是否对齐，字符串字体大小，出错信息内容和字体大小，工具栏位置/图标等等。

**--冒烟测试**--版本编译者

冒烟测试，英文是Smoketesting。

冒烟测试的名称可以理解为该种测试耗时短，仅用一袋烟功夫足够了。也有人认为是形象地类比新电路板功基本功能检查。任何新电路板焊好后，先通电检查，如果存在设计缺陷，电路板可能会短路，板子冒烟了。

冒烟测试的对象是每一个新编译的需要正式测试的软件版本，目的是确认软件基本功能正常，可以进行后续的正式测试工作。**冒烟测试的执行者是版本编译人员。**

**--随机测试**--测试人员

随机测试，英文是Adhoctesting。

随机测试没有书面测试用例、记录期望结果、检查列表、脚本或指令的测试。主要是根据测试者的经验对软件进行功能和性能抽查。随机测试是根据测试说明书执行用例测试的重要补充手段，是保证测试覆盖完整性的有效方式和过程。

随机测试主要是对被测软件的一些重要功能进行复测，也包括测试那些当前的测试样例(TestCase)没有覆盖到的部分。另外，对于软件更新和新增加的功能要重点测试。重点对一些特殊点情况点、特殊的使用环境、并发性、进行检查。尤其对以前测试发现的重大Bug，进行再次测试，可以结合回归测试(Regressivetesting)一起进行。

**--黑盒测试（功能测试）**--测试人员

黑盒测试，英文是BlackBoxTesting。又称功能测试或者数据驱动测试。

黑盒测试是根据软件的规格对软件进行的测试，这类测试不考虑软件内部的运作原理，因此软件对用户来说就像一个黑盒子。

软件测试人员以用户的角度，通过各种输入和观察软件的各种输出结果来发现软件存在的缺陷，而不关心程序具体如何实现的一种软件测试方法。

**--性能测试**

性能测试，英文是PerformanceTesting。

性能测试是在交替进行负荷和强迫测试时常用的术语。理想的"性能测试"(和其他类型的测试)应在需求文档或质量保证、测试计划中定义。性能测试一般包括负载测试和压力测试。

通常验证软件的性能在正常环境和系统条件下重复使用是否还能满足性能指标。或者执行同样任务时新版本不比旧版本慢。一般还检查系统记忆容量在运行程序时会不会流失(memoryleak)。比如，验证程序保存一个巨大的文件新版本不比旧版本慢。

**释义： 页面原型（demo）**

页面原型是项目人员快速熟悉项目的最佳路径。在需求不够明确，设计说明书不够全面的情况下，页面原型也是后期测试用例编写思想的重要根据。

**释义： 测试过程设计**

明确测试目的，最终达成目的并验证结果是测试要做的事情。包括：

1.测试范围：描述本次测试中的测试范围，如：测试软件功能范围、测试种类等。

2.简单的描述如何搭建测试平台以及测试的潜在的风险。

3.项目信息：说明要测试的项目的相关资料，如：输入输出文档，产品描述，软件主要功能。

4.人力资源的分配。

5.测试需求：笼统说，就是测试中的所有设计和需求文档。作为本次测试的依据

**释义： 测试策略制定**

这一阶段在于需求、详细设计、测试计划完成之后，主要是本次测试的策略阶段。很多公司少这个一个阶段，需要有计划性的分出产品的功能扣出测试的功能点，现阶段大多公司都是直接拿着文档就开始做用例设计。

对需求进行分析，列出具体的功能列表。（一般根据功能交互文档就能明确出此功能的大体功能，一层层的分下去，一直到每个功能表单。然后考虑到使用那些测试方法？工作一旦做到执行阶段，我们可以更好的根据这些功能表一点一点的覆盖。也能让我们在用例评审时，充分的证实我们的工作是有效的能够保证产品的质量。）一般在此之前，一些业务培训和需求评审是有必要是听一下的。这样能够更早更熟练的理解需求，也能保证产品设计中出现的一些误区。

对于一个个测试该如何进行测试？如下：

**a)功能测试**

---功能范围（划分出各自负责的功能模块）

---使用测试方法（等价类、边界值等测试方法方法）

---测试标准（符合设计、需求和规范文档对该功能的描述）

**b)界面测试**

**c)兼容性测试**

**释义： 测试计划**

要充分考虑测试计划的实用性，即测试计划与实际之间的接近程度和可操作性。编写测试计划的目的在于充分考虑执行测试时的各种资源，包括测试内容、测试标准、时间资源、人力资源等等，准确地说是要分析执行时所能够调用的一切资源以及受各种条件限制，可能受到的各种影响。

* **测试内容**：对一个软件来说测试计划中会明确本次测试做哪些测试？

如：系统测试：在整个系统测试中会有（界面测试、功能测试、性能测试、兼容性测试、安装卸载测试、可靠性测试等测试）。

* **测试目的**：一般多为保证产品质量是否达到预期的指标。这个指标也就是在测试中定义的结束标准。
* **测试标准**：需要考虑本次测试需要输入那些文档，该项目结束标准定义、测试结束标准的定义？bug级别定义、优先级定义、bug管理流程定义。这个都需要在执行测试事明确。计划中应该包含这些内容。
* **资源分配**：这里分为人力资源、软硬件资源等划分。一般会把人力资源的利用写入一个测试人员任务分配表里，按照不同的阶段，每个阶段提交相应的成果（难度很大）。软硬件资源中主要是在做计划时考虑到需要多少电脑或别的工具，列出清单。
* **测试风险**：大多考虑到的就是项目开发延期、测试人员不足用例无法全面覆盖测试点、时间不足用例无法全部执行、bug无法及时修改导致无法验证、测试人员技能不足导致测试进度拉长。
* 软件测试策略一般都是分开来做相关测试方案。

**释义： 测试附件**

用例模板、缺陷报告模板

测试环境的搭建

缺陷管理流程和缺陷级别定义

缺陷状态一般分为：新建、打开、已分配、已修复、关闭、重新打开

中间会有：延期、重复、拒绝等状态

释义： 缺陷管理流程：



1.测试人员或开发人员发现bug后，判断输入哪个模块的问题，填写bug报告后，系统会自动通过Email通知开发组长和该模块开发者。

2.开发组长根据具体情况，重新reassigned分配给bug所属的开发者。

3.开发者收到email信息后，判断是否为自己的修改范围。

若不是，重新reassigned分配给开发组长或应该分配的开发者。

若是，进行处理，resolved并给出解决方法。（可创建补丁附件及补充说明）

4.测试人员查询开发者已修改的bug，进行回归测试。

经验证无误后，修改状态为verified。待整个产品发布后，修改为closed。

还有问题，reopened，状态重新变为"new"，并发送邮件通知。

5.如果这个bug一周内一致没被处理过。Bugzilla就会一直用email骚扰它的属主，直接采取行动。管理员可以设定最迟采取行动的期限，比如3天，系统默认7天。

释义： 缺陷等级划分：

释义： 测试实施与执行

开发就会转版本给我们测试部门进行系统测试了。拿到版本我们首先搭建测试环境

做一个预测试，目的是来评断这个版本是不是可测试的。如果预测试不通过，打回开发部返工，如果通过了，就开始我们第一轮的系统测试。

第一轮系统测试我们会执行我们所编写的所有测试用例，做好测试结果的记录，发现缺陷了提交缺陷报告。当第一轮测试结束后，我们把所有的bug单提交给开发人员，由他们进行修改。

在他们修复bug期间，我们会对第一轮系统测试做一个测试评估，出一个测试报告。还要根据实际情况，对我们写的测试用例进行修改和增加。

开发改bug结束，提交一个新的版本给我们，我们重新搭建测试环境开始第二轮系统测试。首先是回归我们提交的缺陷报告，然后会在用例中挑选一些优先级别比较高的用例来进行测试，发现问题了继续提交缺陷报告，只到缺陷率低于用户要求了，我们就进行最后一轮的回归测试，结束系统测试。