

软件设计和测试的质量保证

软件质量保证（SQA-Software Quality Assurance）是建立一套有计划，有系统的方法，来向管理层保证拟定出的标准、步骤、实践和方法能够正确地被所有项目所采用。软件质量保证的目的是使软件过程对于管理人员来说是可见的。它通过对软件产品和活动进行评审和审计来验证软件是合乎标准的。软件质量保证组在项目开始时就一起参与建立计划、标准和过程。这些将使软件项目满足机构方针的要求。那么到底该如何来管理软件设计和测试的质量呢？

软件在没有发布之前的开发过程主要分为需求分析、设计、编码和验证四个阶段，最终的软件质量与这四个阶段的各自质量之间的关系如果用 C 语言来表达的话应当是：
最终的软件质量 = 需求分析质量 && 设计质量 && 编码质量 && 验证质量
即，最终的质量来自于各阶段质量之“与”，只要其中一个环节质量是差，则产品的整体质量都将是差，千万不要认为是“或”的关系。由此看来每一个阶段的质量都起着决定性的作用。

在需求分析阶段，通过结对分析与需求评审来保证需求分析的质量。需求分析的质量是否优良，决定着后继各工作环节能否按既定方向进行。往往由于客户方生产较为繁忙，计算机应用水平不高等因素，与中层管理人员、基层操作人员的交流有很大困难，参与项目的时间与精力得不到保障，因此，作者制订了结对分析的策略，以提高需求分析的效率与质量。

结对是敏捷软件开发一个最佳编程实践，将其引入到需求分析

阶段。需求分析各个环节(包括需求获取、分析, 与用户沟通)均由两个开发人员来共同完成, 能取得较高的需求分析质量。例如, 这一策略在仓库备料及库存管理这一关键环节的需求分析上发挥了重要作用。仓库管理人员年纪较长, 对计算机概念模糊, 沟通上存在很大障碍, 通过两个需求分析人员共同来完成该项工作, 做到了分析人员的优势互补。通过结对分析不仅得到较高的需求分析质量, 并且节省了用户的时间, 对用户也有一定的推动。只有两个分析人员达成一致, 才将相应需求嵌入基线库中, 保证了需求的科学性。这样做虽然增加了人力成本, 但提高了需求分析效率

强化需求评审。需求评审分为三个层次: ①全局评审。参与人员主要是用户方核心管理层, 业务领域专家, 开发方分析人员。主要任务是评审需求与公司目标、高层想法是否一致。这一评审至关重要, 它把握了软件系统实施方向的正确性。②局部评审。参与人员主要是用户方业务中层及一线操作人员, 开发方业务领域开发工程师。主要任务是就业务层相关业务流程进行再次确认。因为需求过程中相应部门的业务人员站在自己的角度看待问题, 对于整个流程关心不够, 需要纠正各部门用户需求不一致的问题。③内部评审。全体开发人员、分析人员参与。分析人员建立初步的 UML 用例图, 工作流程图, 活动图等供全体讨论。任务是澄清一部分模糊的需求。

在需求阶段, 还采取了其它措施来保证需求分析的质量, 如强调需求分析工具的使用等。

设计与编程质量对于软件产品的质量有着直接的作用。在这一阶段，我们的主要措施是强调纪律与规范，制订相应的约束规范文件，要求设计有可追溯性，强调设计、开发过程标准化。另一主要措施选择适合的工具。选择的合适的开发平台，后台数据库，使用数据库小工具，使用微软企业库。企业库是微软针对共性内容发布的经过实践检验的高质量产品。

制订 NET 编码规范，目的是保证不同开发人员的代码风格基本一致，降低开发人员流动可能导致的项目风险;强调代码的可理解性与“设计先行与强化注释”;制订了 SQL 书写规范。这一系列规范标准的制订从源头上保证了软件设计的质量。

编码人员分成二类角色：业务领域工程师和核心代码工程师。业务领域工程师针对具体的业务功能进行编码。在设计时我们要求业务领域工程师要有这样的意识：变更不可避免，在设计尽可能考虑到将来变更的可能性，尽可能做到模块之间的松耦合，模块内部的高内聚。核心代码工程师完成公共代码的实现与维护。核心代码质量在系统中至关重要。对核心代码工程师加以分组，每组两位工程师实行结对编程。以克服思维局限，共同完成设计、UML 图、实现，保证代码质量。实践证明，采取这些办法后，在用户提出相应功能变更时，软件修改相对较轻松，能较好地适应业务变化。

测试是保证软件质量较为重要的一个环节。针对开发团队部分成员测试意识不强，项目部有针对性地进行培训;强调测试与设计同步，测试用例与功能代码的编制同步;及时进行单元测试，统一用

VS2005 for Tester 作为测试工具;Bug 管理统一存放在服务器中，严格控制 Bug 的开启与关闭，并把 Bug 数据作为工程师考核依据之一，指派兼职文员进行 Bug 管理。

所有测试中出现的问题，经整理后在项目团队例会上进行总结；在团队内部交流并在平台 WSS 中建立一个专门的警示网站，发布已经暴露的 Bug 解决方案，让项目团队共享，避免问题重复出现；构建项目团队自有知识库，作为公司设计重用的一个重要资源。

除了高度重视实现阶段的测试，同时也积极主动地去发现质量问题。团队使用 VS2005 自带工具进行性能优化分析，找出使用频度高与关键代码进行性能优化。利用 VS2005 压力测试工具，找出性能瓶颈，制订相应策略进行优化。

对于测试过程结果的文档建设来说，非常广，非常全的文档当然是有好处的，但是在现实的情况下，并没有如此大的精力和时间去建设非常广，非常全的文档。所以，择其重要性而有选择性的进行文档建设，是一个测试团队所需要的，单纯从文档建设的数量上来评估测试过程结果的质量，是很失真的一种行为，但是如何去选择文档的重要性，不同的人有不同的观点，我觉得，文档的重要性一是对于整个测试过程有帮助的，另一是对当前需求的测试过程的知识传递程度两个方面来评定文档的重要性。就此，可以比较容易的观察整个测试过程结果的文档建设及文档知识传递程度来评估测试过程结果的质量。

当然，我们也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，在这里提出几点建议：

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。

只有在需求分析、设计、开发等各个环节实施质量保证措施，才能保证软件产品的质量。软件项目要做到“有法可依，执法必严”，并要高度重视各个环节辅助工具的使用，这样才能较好地保证最终软件产品的质量。

软件 1403 班

于杰

14126173