



北京交通大学
Beijing Jiaotong University

软件质量保证结课论文

题目：如何保证软件测试质量

学院：软件学院

年级：1403 班

学号：14126112

姓名：李锦绣

随着软件系统规模和复杂性的增加，其中软件开发成本和由于软件故障而造成的经济损失也正在增加，使得软件质量问题成为制约计算机发展的关键性因素。在软件产业高速发展的过程中，软件质量成为决定企业竞争优势的最重要因素。通常我们可以从三个方面来考虑软件质量：软件结构方面；功能与性能方面；开发标准与文档方面。不同性质和用途的软件会有不同的特征集合和质量要求，不同角色对软件的质量又有不同的观点。若从用户的角度来评价软件质量，他们关心的是所要的功能在软件中是否实现，软件的效率和可靠性如何，软件是否方便使用，改变或添加软件功能是否方便等等。从开发者角度看，虽然和用户的角度关注点相似，但为了保证软件产品的质量，他们更加注重软件的结构和软件的可维护性，以及在开发不同阶段对软件中间结果测试是否方便。而从管理者角度观察软件，他们则更注重软件整体的质量，而不是某个具体质量属性，需要根据软件应用的类型和商务需要为不同的特征分配权值，同时还要考虑成本、资源消耗、开发周期的限制等，最后在这些管理要素和质量改进需求间寻求平衡。软件质量保证是一种有计划的，系统化的行动模式，它是作为一个有系统，有计划地行动集合，为项目或产品符合已有技术需求提供充分信任所必需的。而软件质量保证活动不应局限于功能需求方面，它应该包含在开发、测试与维护整个过程中。

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。所谓软件测试，就是出于正常合理的目的，在特定的时间环境，用事先制定的标准衡量一种软件产品或特性是否符合预期。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。软件测试是为了软件质量足够可靠，不是为了软件没有 BUG，质量完美。由于软件产品本身无形态，是复杂、知识高度密集的逻辑产品，没有一种软件方法可以保证在软件的设计和实现过程中没有错误。

软件测试的价值与成功更多的源于经验和管理，能不能把一个测试项目计划、组织的井井有条和快速高效，把一个庞大的任务科学的细分并在合适的点上进行监督，用丰富的经验预判并规避可能的风险，这才是决定一个测试项目是否成功的关键。在软件开发的每一个阶段，在当前的软件开发过程中，有 30%-40% 的精力被花费在软件生命周期内所有的检查、评审、确认的软件测试上。软件测试的原则需要我们必须遵循以下几点：在整个开发过程中要尽早地和不断地进行软件测试；在开始测试时，不应默认程序中不存在错误；设计测试用例时，要给出测试的预期结果；测试工作应避免由系统开发人员或开发机构本身来承担；对合理的和不合理的输入数据都要进行测试；重点测试错误群集的程序区段；除检查程序功能是否完备外，还要检查程序功能是否有多余；用穷举测试是不可能的；长期完整保留所有的测试用例和测试文件，直至该软件产品被废为止。

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。

首先，测试应尽早介入项目开发中。在以往参加的项目测试中，要求测试人员严格按照 V 模型定义的标准来开展测试工作，结果往往是在项目初期测试人员工作量极度不饱和；而到项目后期，一旦项目经理决定压缩测试时间，测试人员就不得不加班加点赶进度。测试人员叫苦不堪，直接导致测试效果不理想。在项目开发初期，测试人员可以开展很多有价值的工作，例如从需求分析阶段就参与讨论，一方面对业务流程以及关联模块要很清楚，另一方面根据需求分析文档整理出测试需求。在开发设计过程中根据需求和设计文档进行测试用例的详细设计；在测试用例编写完毕后认真考虑每个测试用例的优先级，把好钢花在刀刃上。对于优先级高的功能优先进行充分测试，对于测试优先级低的使用例放到测试后期进行。提前准备测试环境，包括数据库环境，操作系统和 web 应用服务器，以及复杂集群环境。为减轻测试人员的工作量，特别是测试后期需要回归测试，可以考虑使用自动测试工具代替人工测试。做好技术和测试人员的培训工作，组织相关人员对业务需求、编程语言、环境配置、检测内容等进行详细讲解。

编写一份详细的测试用例提高测试质量。详细的测试用例完全覆盖了代码的所有路径。把这样一份测试用例发放到测试人员手中，都能高质量的执行测试过程，测试用例完全覆盖所有需求，测试人员就不会因为不熟悉业务而遗漏需要测试的需求。把测试用例分为场景用例、个别用例、体验用例，通过这三类讨论如何提高测试用例水准。用户使用或操作习惯搭配称为场景，要写好场景用例，需要仔细了解测试目标的业务逻辑，然后根据业务逻辑来筛选出有效场景和无效场景，此类用例贴近用户使用习惯，查不出缺陷则已，一旦查出都是影响比较大的。不同的用户可能有不同的操作习惯，所以要完全模拟出用户的操作是比较难的，但是入口确是我们可以控制的。因此我们要整理出达到测试目标的入口，然后根据入口搭配出各种路径。针对这些路径进行覆盖，这样就可以设计出好的场景用例；个别用例这类用例比较好设计，这类用例可以抽出其共同部分写成通用用例，这样不但可以让用例看起来清楚，还能突出重点。主要是指针对输入或者各类参数进行的针对性的用例，比如上例中的输入框，选择框的针对性用例；体验用例扩展出来比较多，仁者见仁智者见智，但做好这部分测试，产品的黏性就好，竞争力就更强。

但测试用例没有逻辑性会拖慢整个效率，因此我们划分测试用例优先级为四级来提高测试效率。冒烟测试：在程序提交后，对程序的基本功能进行自上而下的简单测试。强调功能覆盖率，可以执行最基本的功能，但不验证功能细

节的正确性。目的是先通过最基本的测试，尽可能早发现影响测试的严重 bug，直接打回开发组，避免浪费测试时间。高优先级：保证功能的稳定性，目标的行为和能力可以正常工作，包括模块功能和流程方面的测试。中优先级：对功能区域的细节进行测试，包括边界、错误和配置方面的测试。低优先级：通常可放在测试后期被执行的测试用例，包括 GUI、负面、压力和性能测试。具体测试过程中，测试组将中、低级测试安排在开发人员对冒烟、高优先级测试发现 bug 后进行代码修改阶段进行，极大地提高测试效率。

当然，仅仅交出一份令人满意的用例是不够的。如果设计有错误，测试的质量就难以保证。即便测试后发现设计的错误，这时修改的代价也很昂贵。测试用例设计得不可能天衣无缝，不可能完全满足软件需求的覆盖要求，测试执行过程里肯定会发现有些测试路径或数据在用例里没有体现，那么事后该将其补充到用例库里，以方便他人和后续版本的测试；如果没有这样做，那么测试部门负责人和每个测试员都难辞其咎，是该重新坐下来思考一下公司的测试用例管理规范 and 测试流程，以严格和良好的机制促使和保证测试执行者按其执行了。一套完善的软件过程测试规范，对软件测试过程进行管控，不仅是对人员沟通问题进行管控，而且是对问题反映流程进行的约定，这样一个流程可以降低沟通成本，而且对软件测试流程进行了统一的管理。

我们要在做好详细的测试管理计划的基础上建立较完善的软件测试管理体系，主要目的是确保软件测试在软件质量保证中发挥应有的关键作用，包括以下工作：（1）软件产品的监视和测量：对软件产品的特性进行监视和测量，主要依据软件需求规格说明书，验证产品是否满足要求。所开发的软件产品是否可以交付，要预先设定质量指标，并进行测试，只有符合预先设定的指标，才可以交付。（2）产品设计和开发的验证：通过设计测试用例对需求分析、软件设计、程序代码进行验证，确保程序代码与软件设计说明书的一致，以及软件设计说明书与需求规格说明书的一致。对于验证中发现的不合格现象，同样要认真记录和处理，并跟踪解决。解决之后，也要再次进行验证。（3）对不符合要求的产品的识别和控制：对于软件测试中发现的软件缺陷，要认真记录它们的属性和处理措施，并进行跟踪，直至最终解决。在排除软件缺陷之后，要再次进行验证。（4）软件过程的监视和测量：从软件测试中可以获取大量关于软件过程及其结果的数据和信息，它们可用于判断这些过程的有效性，为软件过程的正常运行和持续改进提供决策依据。

但是即使拥有了全面的管理规范，应用在一群测试技能底下的测试人员手中，并不会让测试的质量有本质的提高。重点落在了测试人员的能力提升。一个优秀的测试人员不仅可以写出高质量的测试用例，覆盖测试的所有需求，还能在测试过程中发挥个人主管意识，发现可能存在的问题，比如未规定的输入，页面响应的性能问题。提升了测试人员的技能，还可以真正发现测试过程管理中的问题，持续优化已有的测试过程管理。加强测试部门内部人员的培训

教育很重要，包括工作技能与个人素质两方面，可通过国内主要的培训机构，也可是购买工具厂商的直接培训。其次是要学习借鉴国外完善的测试机制，包括丰富的软件测试经验，强大的测试工具，优秀的测试管理水平。真正解决测试手段落后、测试方法单一和测试工具欠缺的问题，在企业内部形成一个严密有效的纠错系统，使国内的测试工作流程、技术水平接近国外先进水平，这样才能提高国内软件开发与测试的整体管理水平，增加软件产品的竞争力。

国内软件测试行业目前仍处在群雄逐鹿、百家争鸣的时期，人云亦云，不如从企业自身出发，找到并确立最适合自我的测试管理解决方案，整顿自身的测试工作流程，提高测试人员素质，将软件测试的力量发挥到最大值。