

论软件需求获取技术及应用

【摘要】

2019年3月，我单位联合某高校研发了《程序在线评测比赛考试系统》。系统以程序代码在线提交自动评测功能为核心，分为题库模块、评测机模块、实验作业模块、考试模块、比赛模块、抄袭判定模块、用户管理模块等，支持对接教务平台。在项目中我担任系统架构师，负责架构设计工作。

本文以该系统为例，主要论述了软件需求获取技术在项目中的具体应用。在需求的前期阶段，以用户访谈和调查问卷结合的方式来进行需求获取；在需求的中期阶段，以现场观摩的方式来进行需求获取；在需求的后期阶段，采取构造快速原型的方式，持续迭代，来进行需求细化和系统演进。利用多种技术实施需求获取，有效地降低了项目风险，最终系统顺利上线，获得了用户的一致好评。

【正文】

笔者在一个专为高校建设计算机专业智能教学一体化平台的单位任职，过往成果有《计算机组成原理仿真实验系统》等。2019年3月，我单位联合某大学研发了《程序在线评测比赛考试系统》项目（以下简称“OJ系统”），以取代原有传统的编程上机考试平台。

系统以程序代码的在线提交自动评测功能为核心，主要分为题库模块、评测机模块、实验作业模块、考试模块、比赛模块、抄袭判定模块、用户管理模块等。题库模块主要负责试题和测试用例的管理，用户根据试题要求编写程序代码提交到系统，系统将测试用例与程序代码发送给评测机模块，由评测机自动编译、执行、判分，并将结果发送给其他相关模块进行统计；实验作业模块用于在线布置作业，从题库中选取试题，设置截止日期等要求；考试模块用于学生在线考试，按教师预先设置的参数自动从题库随机抽题生成试卷，以及向教务平台上传考试成绩；比赛模块主要用于ACM竞赛的培训；抄袭判定模块用于鉴定代码与他人代码雷同率；用户管理模块负责用户信息的管理。在这个项目中，我担任了系统架构师的职务，主要负责系统的架构设计相关工作。

常见的需求获取技术有用户访谈、问卷调查、现场观摩、原型化方法等。用户访谈主要对三个之内代表性客户谈话沟通获取需求，优点是灵活性好，适用范围广，缺点是时间难以安排，信息量大记录困难，需要领域知识，对机密话题敏感等，适合简单小范围的需求获取；问卷调查主要通过设计调查表来收集用户需求，优点是可短时间内廉价从大量回答中收集数据，缺点是双方未见面无法澄清提问，反馈信息不全面，无法深入问题细节，适合大范围需求收集；现场观摩主要针对较复杂、难理解的流程、操作，优点是直观清晰，缺点是效率较低，适合复杂需求的获取；原型化方法通过构造一个简易原型系统，根据用户在试用过程中意见重复修改直到满意，优点是允许用户早期交互反馈，缺点是较为费时，会误导用户对未来系统有不切实际的希望，适合需求不明确的情况。

本文主要通过前期、中期、后期三个阶段，分别论述了OJ系统项目在需求获取过程中所采用的具体技术方法。

1. 前期阶段

在此阶段遇到的主要问题就是 OJ 平台需要支撑全校编程课程的教学活动，这就需要有效、快速地全面概括需求。我们采用了用户访谈和调查问卷结合的方式，来进行需求获取。由于涉及编程课程众多，为了能突出重点，我们先选择了在教学方式上有代表性的部分课程，如程序设计基础、算法与数据结构、操作系统等，优先进行，然后再逐渐铺开。我们把需求调研团队分成了几组，分别进行需求收集。课程教学组长负责课程组的教学领导工作，我们采用了访谈的方式进行需求获取，通过与课程教学组长的详细沟通，我们对 OJ 系统的主要业务功能、用户角色等有了整体、全面的了解。由于任课教师具体的教学活动过程较为复杂，如实验作业、考试、比赛培训等，在课程教学组长的配合下，我们制作了调查问卷表格，下发给各位任课教师，经过统计整理后，我们获悉了编程课程的教学活动、过程细节。这种安排主次分明、详略得当，在前期起到了不错的效果，给后续的需求获取活动搭起了良好的基础。

2. 中期阶段

在前面的阶段，虽然在总体需求的获取上已经有了一定的基础，但很多流程仅仅通过访谈和问卷无法直观了解。为了防止前期需求分析的缺陷带到后续阶段，我们决定采取现场观摩的方式来进行需求获取。我们征得单位领导的同意，在课程教学组长的协助和安排下，跟随任课教师，前往学校的计算机实验室，对目前编程课程的实验教学现场进行了观摩，了解了在传统实验教学方式中，学生和教师们的具体操作流程。比如，在传统的环境下，学生按题目要求编写完程序代码，需要手动进行编译、执行，然后输入测试数据，观察是否得到预期的结果。教师批改学生提交的实验作业时，也需要手动将代码逐个拷贝到编译环境中执行、测试。在这样的过程中，大部分时间和精力都在反复地复制粘贴，随意输入的测试用例并不能全面测试出程序的设计问题，且很难进行抄袭、雷同的判定。通过现场观摩的方式，我们更清楚地了解了业务流程，为后续 OJ 系统解决传统实验教学中存在的问题起到了良好的作用。

3. 后期阶段

在此阶段我们基本上已经完成了大部分业务需求的收集，通过快速原型法构造出了一个简易的 OJ 系统，供用户试用与反馈。这个原型只是一个系统框架，很多操作是空动作，目的是向用户说明系统的功能和操作方法，以后再随着开发进程以及需求明确逐步求精。例如程序代码评测和考试功能，暂时不会进行实际的程序评测和累分，而是直接显示一个固定的评测结果和成绩，展示给用户看。整个构建过程，让用户也参与到设计中，提供了工作流程方面、业务领域方面不可或缺的经验，也为以后项目通过验收提供了有力支持。在每一次迭代过程中，通过和课程组交流，在完善需求的基础上，完善对象模型。某次试用中，课程组向我们提出一个需求“用户提交的程序代码，评测应当在较短时间内完成出结果，不能让用户等待”，但无法明确“较短时间”是多少合适。为完成这一需求，采取模拟延迟的方式，让课程组教师现场试用，明确了这个时间应该在 10 秒内，并反复修改原型，完成了这一迭代需求。

【总结】

系统自 2019 年 10 月正式上线已运行一年有余，在学校的日常教学考试和竞赛培训中投入使用，截至目前已有 3000 名以上的学用户、评测了 70000 份以上的程序代码，获得了单位同事领导和学校教师们的一致好评。在使用中系统也

出现了一些问题，比如评测机服务需要执行用户提交的代码，部分用户短时间内提交了大量不安全代码，导致评测机集群中所有实例全部宕机。这是由于早期在可靠性方面的需求分析不够周全，系统没有进行充分的容错设计。我们引入心跳机制、快照回滚机制，以及基于机器学习技术的预判断机制，使评测服务宕机时能够在 10 秒内自动重置恢复运行，最终解决了该问题。

实践证明，OJ 系统项目能够顺利上线，并且稳定运行，与系统采用了合适的需求获取技术密不可分。经过这次需求获取技术的方法和实施的效果后，我也看到了自己身上的不足之处，在未来还会不断地更新知识，完善本系统在各方面的设计，使整个系统能够更加好用，更有效地服务于高校师生。

版权声明：本文为 CSDN 博主「苏若藓」的原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接及本声明。

原文地址：https://blog.csdn.net/sinat_31152963/article/details/111387674