

Exploration of Scrum-based Software Engineering Practice Teaching

Yingan Cui¹, Hui Xia¹, Hui Liu¹, Xue Li²

Xi'an University of Technology, Xi'an 710048, Shanxi, China

Shanxi Normal of University, Xi'an 710062, Shanxi, China

Abstract: The improvement of software engineering practical ability has always been the focus and difficulty in the teaching process of the Software Engineering course. In recent years, the course team has introduced the Scrum agile model to promote students to carry out iterative development, and established the learning mode of Sprint continuous improvement. Students' learning interest, software development ability, and teamwork ability have been significantly improved, which has gained a good application effect in undergraduate teaching and comprehensively improved students' learning output. Our practice shows that undergraduates' software engineering ability can be improved more efficiently only by following the principle of asymptotic approach, linear learning model is not the best practice, and iterative process and continuous improvement have better results.

Keywords: Software engineering; Practical education; Scrum process; Agile process; Iterative development

基于Scrum的软件工程实践教学探索

崔颖安¹, 夏辉¹, 刘辉¹, 李雪²

西安理工大学, 中国·陕西西安 710048

陕西师范大学, 中国·陕西西安 710062

摘要: 软件工程实践能力的提升一直是《软件工程导论》课程教学过程中所关注的重点与难点。近年来, 课程组引入 Scrum 敏捷模型推动学生进行迭代式开发, 通过建立 Sprint 持续改进的学习模式。学生的学习兴趣、软件开发能力以及团队协作能力均有明显提升, 在本科教学中获得良好的应用效果, 全面提升了学生的学习产出。我们的实践表明: 本科生的软件工程能力需要遵循渐近的原则才有可能得到更高效地提升, 线性学习模型并非最佳实践, 迭代过程与持续改进有更好的效果。

关键词: 软件工程; 实践教学; Scrum 过程; 敏捷过程; 迭代式开发

1 引言

《软件工程导论》是信息技术类相关专业的一门非常重要的专业基础课, 主要讲授软件工程的基本概念、基本原理和基本方法。要求学生了解软件项目开发和维护的一般过程, 建立科学的软件设计开发理念, 能将软件工程方法应用于复杂工程问题的分析、设计、编码和测试实践中。为了提高学生的学习产出, 课程组自2015年起对本课程进行了系列改革。首先引入国际通用的CDIO工程教育理念对课程的教学模式进行改革, 形成理论讲授、案例分析、问题讨论相结合的教学模式。其次利用学校云课堂在线学习平台开展线上线下混合式教学, 将课堂延伸到了课外, 拓

展了学生的学习空间, 初步建立了过程性学习机制。

在课程改革取得成绩的同时, 课程组也意识到课程还存在一些突出的矛盾。按照预期, 学生应该能够应用各类软件工程CASE工具, 进行全生命周期的软件工程实践。但是从实际的学习产出来看, 由于绝大多数本科生缺少实际项目的开发经验, 因而很难将课程中所学的知识转化为解决实际复杂工程问题的能力。另外课程评价机制需进一步优化和完善, 特别是项目实践的成绩无法充分的体现在最终成绩中。这样使学生错误的认为本课程只需要多做些题、只要取得较高的考试成绩就算达成了学习目标。为了进一步提高学生的实践能力, 本课程近几年应用 Scrum 敏捷模型进行软件工程实践能力的提升, 在探索过程中, 我们逐

步找到了一些好的做法，本文介绍我们的探索和体会。

2 Scrum敏捷模型介绍

Scrum 属于敏捷模型，图 1给出了scrum的整体框架^[1]。该图表明scrum是由若干个sprint组成的迭代过程。每一个sprint的目的是定义在sprint中可以交付什么，以及如何实现该工作。通常情况下，每一个sprint为项目团队仅规定2-4周的任务，这些任务就是一个迭代周期，在每个迭代周期结束时，scrum团队交付产品进入评审阶段，而后根据评审发布新任务以明确下一个sprint的工作^[2,3]。



图1 scrum模型框架

尽管Scrum模型中每一个sprint的周期是2-4周，但是Scrum仍旧是一种非常高效的敏捷过程，其关键在于它的迭代式需求管理机制。简单地说，在Scrum项目启动时必须初始化一个需求清单（Product Backlog），这些需求需要按照用户要求的迫切性或敏感度排序。这意味着只需了解最关键的用户需求就可以启动第一个Sprint进行开发^[4]。与此同时，可以再去获取次优先级的需求，当第一个Sprint完成后，次优先级的需求以及已实现的系统一起进入评审议程，经过与实际用户需求的匹配分析后再制定新的Sprint backlog。如此循环反复，即可高效率完成软件的敏捷开发。

除了高并发的需求获取机制，Scrum的日常管理秉承轻量级的管理理念。开发团队在每天早上开展全员的“每日站会”，会议时间最多不超过30分钟，但是会议主题高度聚焦于当日的工作，因而通过“每日站会”能够及时的暴露软件开发过程中存在的各种问题，这样就能够避免将小问题累积成重大问题，为低成本、高效解决问题提供了可能^[5]。总的来看，Scrum希望将传统的线性软件生命周期模型转化为闭环的软件迭代过程，通过不断的增量式迭代开发促进软件功能的微循环改进。

3 教学改革的实施

《软件工程导论》是我校计算机学院各专业的必修课，理论教学是从第1周到第10周，实践环节安排在第4周到第18周，在18周时提交实践性作业并完成课程考试，因此整个项目实践总时长为15周。从我校课程的安排来看，与绝大多数院校信息技术类专业的安排都是一致的，因而其他院校的实践经验对我们也具有很强的参考性，根据之前的教学改革经验^[6,7]，我们可成整体实践环节安排如下：

Step-1.项目启动阶段：1周

通常情况下，本课程有近400名选课同学，这些同学分布在不同的专业中，包括计算机科学与技术、物联网工程、软件工程、网络工程等。课程组开发了一个随机分组的程序，把全体同学跨专业分配在不同的组中，每个小组大约6人左右。实践作业的题目由老师确定，题目为《学术讲座web发布系统》，课程组给出基本需求，各Scrum小组可以在此基础上进一步个性化需求。与此同时，课程组在服务器上配置开发环境与项目管理工具，准备若干业务仿真数据，为后期开发做好准备工作。

Step-2.Scrum执行阶段：10周

(1) Sprint初始化：在确认了Scrum小组的分配名单以后，课程组全体教师共同担任Product owner，负责用户需求的管理和指导，组内同学自行确定Scrum master和Scrum developer，根据Scrum规定的职能要求，各类角色开始启动自己的工作并在GitLab中完成初始化。Sprint初始化与需求紧密相关，或者说，需求对Sprint的规划与设计影响至关重要，因为不同规模、不同流程模式的需求会对软件系统的复杂系统与开发难度产生很大的影响。为了达到应有的教学效果，课程组会给出需求中的核心要求，这个要求是基于流程化模式的需求，需求相对比较复杂，能够符合本科生教学的要求，尤其是多功能协作开发的要求。

(2) Sprint设计：

各Scrum小组与Product owner谈论，确定本组的Product Backlog，在Product Backlog达成共识后，确定初步的sprint计划。为了充分体现Scrum敏捷开发的特性，课程组要求每一个Scrum小组sprint迭代周期至多为二个星期，这就意味着在项目开发过程中会执行5个Sprint迭代。根据项目的开发迭代周期，每个小

组根据Product Backlog中需求项的优先级来制定Sprint Backlog。在Sprint的设计上，我们自身的体会一定要在首个Sprint就能创建出多功能协调的复杂应用场景。具体的说就是学术讲座通知信息的管理、学术讲座信息的微信与邮件订阅、学术讲座信息的微信与邮件通知三大模块同时考虑。这样做的目的—一方面是开发小组的每一个同学都有自己的任务，避免浑水摸鱼；另外也是为了在最开始就要求同学紧密协调，如数据接口、方法接口的规范交互。

(3) Sprint迭代

每一个Scrum小组根据当前Sprint Backlog的任务进行软件开发，开发的代码需要提交GitLab。利用GitLab的CI/CD功能，快速完成产品的集成与部署，在完成部署后同步更新个人工作状态，标识工作进度。考虑到学生实施开发的实际情况，我们将Sprint的每日站会（Standup Meeting）与每两周进行的Sprint评审会合并。这样既可以确保会议内容充实，也能在一个会议中对Product Backlog进行修正以规划新的Sprint Backlog。

Sprint迭代是一个非常令人激动的过程，之所以这样说主要是因为每一个小组迭代的系统都有很大的差异。例如有的小组在教师提供的基础需求上增加了学术讲座视频资料管理功能、有的小组则增加了讲座广告导出功能、有的同学甚至增加了学术讲座自动上传功能（将学术会议的视频通过网络爬虫自动化上传到视频网站中）。这些个性化的需求都是各个小组自身的思考与讨论的结果，每一个功能都是对教师给出需求的提升，充分展现了同学的个性与风格，使系统更加别具一格。

Step-3.项目试运营：1周

经过10周的Sprint迭代开发，绝大多数小组能够发布一个较为完整的系统。在确认应用环境与系统的可用性后，教师团队释放系统访问权限供全体同学使用。各组成员可以选择任意方式对其他小组的软件进行试用与测试，考验软件的正确性、可用性、安全性、兼容性等多个方面。各组可以使用项目管理软件中提供的Bug跟踪系统记录不同小组在测试过程中暴露的问题，待一周结束以后，所有待测系统下线，内测结束。

项目试运营这个环境非常必要，一方面它可以促进每个小组提高软件系统的设计、开发、测试质量。

让同学们明确的感到工程化开发与之前课程设计的根本性不同。有的同学认为项目开发与课设一样，都是为了学习某种技术而进行的实践。事实上，工程化软件的质量要求要远远高于课程设计，任何一个项目组都必须提供可用性强的系统，这对于同学的反思与进一步改进都会有很大的促进作用。

Step-4.Scrum小组回顾会议 1周

教师团队将测试结果发布给全体同学，一方面各组根据自己内测暴露的问题进行修改和完善，另外一方面要求各小组反思和总结。反思的重点在于回答如下问题：为什么我们的系统会存在这些缺陷？为什么在项目的构思阶段没有考虑到这些问题？为什么测试阶段没有发现这些问题？其他小组的优势在哪里以及产生这种优势的深层次原因是什么？我们应该从其他小组借鉴哪些有价值的宝贵经验？

Scrum回顾要有重点。尽管授课教师都希望学生能够通过本课程的学习达到综合性的提升，但是我们必须考虑一个非常现实的情况就是毕竟在校本科生的工程实现能力是有限的，另外学生能够投入课程的时间资源也是非常有限的。因而任课教师要能指导学生进行合理的反思。首先还是要鼓励学生，让学生能够树立信心，既要看到软件开发需要掌握大量的技能，也要看到只要努力就有可能取得进步这一基本事实，不能因为开发能力或者经验的不足就对同学未来的学习信息有所打击。另外要引导同学工程技术必须和现实需求结合，技术服务于现实世界才能发挥技术应有的价值，否则就是屠龙之术、毫无意义。此外要有持续改进，不断完善的精神追求，这样才能不断进步。

Step-5.Scrum系统优化 2周

根据Scrum小组回顾会议的反思，每一个Scrum master重新设置改进版Product Backlog,改进任务对应一个时长为2周的Sprint，在这个Sprint中完成本小组需要进一步需要修改与完善的工作。经过全面测试后，各Scrum小组向GitLab提交最终代码，同时在项目管理软件中标识项目结束。待项目结束后，任课教师会对各组的系统进行综合点评，点评主要是从Scrum过程的角度（如表 1）再次梳理整个迭代过程中的关键点和关键措施，帮助同学们体会敏捷过程与传统软件生命周期模型的不同，给同学更多启发，以便在后续的各类学科竞赛中继续应用本课程所学的方法。

Sprint 关键活动			
关键活动	输入	活动	输出
Scrum 每日会议	Product backlog 与业务需求等组合形成的初步需求	每日会议主要目的是组内的沟通,让大家知道当前任务和各自的困难。	每日更新看板 每日确定下一个 sprint 的待办事项
Scrum 评审会议	待交付的软件 (可以是测试环境中能正常运行的软件)	向最终用户展示工作成果,团队成员得到反馈	与用户需求的偏差 修正日志
Scrum 回顾会议	针对当前 Sprint 所有的产出物,包括代码、文档、用户测试结果等	按照 PDCA 环,对当前 Sprint 的全部活动进行反思,为持续改进提供指导	新的 sprint 规划 新的 Product backlog

表1 Scrum过程中关键活动及其产出

4 总结

本课程自启动Scrum敏捷过程进行软件工程项目实践以后,我们观察到学生的学习兴趣、实践能力和团队协作精神都有了比较明显的提升。首先,学生普遍认可敏捷开发的实用价值。与之前课程采用传统的软件生命周期模型相比,不少同学觉得软件生命周期模型尽管在某些应用场景有非常重要的作用,但是在课程实践中难以得到实际应用。Scrum的引入,特别符合学生的认知规律和学习规律,尤其对于参加互联网+大赛、大学生挑战杯等竞赛的同学,他们感觉特别实用,原本无序的团队管理变得有序且能持续改进。

其次,学生感到学习效果有了明显的改进。由于课程采用Scrum进行迭代,这就促使同学不断的进行软件改进。在项目试运营阶段,每一个小组都可以对其他小组进行评价,尤其是好的作品,得到的“点赞”数非常高,这些佳作就会被众多同学下载,对其需求、设计、代码进一步的研究。通过相互学习,不少同学能够从这些作品中学到很多知识,另外,很多同学会从这些优秀作品背后学习到优秀同学思考问题的方法、精益求精的精神,而这些都能带给同学巨大

的触动。

经过近3年的实践,我们看到了Scrum方法给软件工程实践教学带来的效果,也观察到很多同学学习态度的转变。我们深深感到Scrum方法能够有效激发学生的积极性和主动性,全面提高了学习产出。在未来的教学中,我们将进一步优化 scrum在课程中的使用方法,更好的融合项目管理工具与软件工程各环节的CASE工具的信息交互,更加敏捷的推动项目过程的管理与软件质量改进,使学生的实践能力有更大的提升。

参考文献

- [1] <https://blog.csdn.net/johnstrive/article/details/46502283>
- [2] Mahnic, V., and Zabkar, N., 2022. Measurement repository for scrum-based software development process.
- [3] Joskowski, A., Przybylek, A., and Marcinkowski, B., 2023. Scaling scrum with a customized nexus framework: a report from a joint industry-academia research project. *Software: Practice and experience*(7), 53.
- [4] Rodriguez, G., Vidal, S., Marcos, C., and Saucedo, A. C. M., 2022. Evaluating students' perception of scrum through a learning game. *Computer applications in engineering education*.
- [5] 胥康. Scrum 方法在软件项目管理中的应用. *信息系统工程*, 2017(1), 1.
- [6] 尹华, 王志坚. 基于 scrum 敏捷开发的软件实训设计. *计算机教育*, 2016(6), 5.
- [7] 顾家铭. Scrum 和 robocode 在 c# 教学中的应用探索. *科教导刊*, 2016(6), 2.