A Study of the Scrumage Teaching Approach: Student Learning and Attitude Changes

Shannon Duvall
Computer Science Department
Elon University
Elon, NC, USA
sduvall2@elon.edu

Dugald R. Hutchings
Computer Science Department
Elon University
Elon, NC, USA
dhutchings@elon.edu

Scott Spurlock
Computer Science Department
Elon University
Elon, NC, USA
sspurlock@elon.edu

Robert C. Duvall
Computer Science Department
Duke University
Durham, NC, USA
rcd@cs.duke.edu

***Abstract*— Best practices for teaching a diverse population include incorporating several modes of instruction, providing a variety of resources, and encouraging collaboration. However, traditional classroom management does not necessarily support these best practices. Scrumage (SCRUM for AGile Education) is a recently proposed classroom management technique in which students are given autonomy to choose individually from a variety of pedagogies (e.g., traditional lectures, active learning, a flipped-based approach, etc.), resulting in multiple simultaneous pedagogical methods in a single course. In addition, students in a Scrumage classroom must frequently reflect on the effects of their choices. In this work, we compare Scrumage and traditional sections of an introductory programming course. Scrumage students showed improved attitudes about learning, especially in the areas of Effort Regulation (perseverance in problem solving) and Control of Learning (taking responsibility for learning success). We believe that by promoting positive attitude changes and better content learning, Scrumage has potential for widening the retention of students in Computer Science.**

Keywords—Computer Science Education, Pedagogy, Inclusion, Classroom Management, Scrum

I. Introduction and Related Work

Professors who want to teach in a way that is inclusive to diverse populations adopt course goals beyond content learning objectives, such as creating a welcoming environment, celebrating individual differences, and fostering a growth mindset. Research shows that students are more successful when they have positive attitudes about learning, confidence in their abilities, and an accurate view of their own progress [1], [2].

It is not always clear how to practically achieve this type of goal in a typical classroom, however. For example, using peer instruction or team learning may be helpful for promoting persistence in underrepresented racial groups, but these practices can be disengaging or overwhelming for people with autism spectrum disorders. As Rose et al. concludes, "There is no one means of engaging students that will be optimal across the diversity that exists" [3].

The diverse needs of diverse learners are recognized by the Universal Design for Learning (UDL) theoretical framework [4]. UDL enumerates three key principles: that learning should utilize multiple means of representing new material, multiple means of expressing knowledge, and multiple means of engagement [3]. However, UDL is a set of *principles*, not a set of *practices*. In other words, how individual educators go about the implementation of the principles is neither prescribed nor obvious.

II. Scrumage Course Management

We propose implementing UDL principles using a novel class management system called Scrumage [5]. The aim of Scrumage (SCRUM for AGile Education) is to give students the power to choose their own pedagogical approach, despite the fact that other students in the same course may be choosing differently. Rather than the professor choosing, e.g., traditional lectures, flipped classroom, or game-based class, and then imposing the choice on all the students, instead the Scrumage technique presents each of these as options for the learner.

Allowing a wide variety of choices requires simultaneously conducting differing methods of teaching and learning in a single course. To manage the classroom in a structured and organized way, Scrumage borrows techniques from the Scrum project management system widely used in industry by allowing self-regulating teams to plan and schedule the completion of work by collaborating in short bursts called sprints. In Scrumage, the "work" being undertaken is learning and the fulfillment of course objectives; the client is the instructor. The completion of specific deliverables (i.e., assignments) are a byproduct of the main "project" of mastering the material at hand. To mirror the values of both Scrum and UDL, Scrumage emphasizes giving students as much control over course management choices as possible and encouraging them to try new methods as the course progresses. Students are generally allowed to work in teams whose size and processes are largely unregulated by the professor. Importantly, students get to choose individually how they use class time by making requests from the professor. Learning methods like reading the textbook, listening to a lecture, watching a video, or completing a worksheet are not generally required activities but rather comprise a menu of options from which students can choose. In this way, students avoid learning activities they believe to be ineffective and thus are more committed and engaged in the activities they do choose. At the end of the unit, or sprint, students are assessed in usual ways with assignments, quizzes, or tests. They then complete a retrospective in which they reflect on their success – both in how they performed (as reflected by

978-1-7281-7172-2/20/$31.00 ©2020 IEEE

grades) and also as to which learning activities were most effective for them, allowing for more informed decisions on how to learn in the next sprint. The Scrumage approach includes a systematic way of implementing several best practices for inclusion and diversity: presenting multiple choices for learning, encouraging (but not requiring) social interactions among learners, providing a rapid feedback loop with several small assessments, and offering opportunity for reflection and change.

III. Methodology

We administered pre- and post-surveys to examine 6 sections of an introductory computer science course (CS1) at a liberal arts institution in the southeastern United States. Of the 6 sections, 3 were delivered using Scrumage and 3 were delivered using a more typical approach that blended lecture, live demos, and in-class labs, which we refer to as a *traditional* approach. We determined changes in student learning attitudes by adapting the "Motivated Strategies for Learning Questionnaire" [6] to examine 4 categories: Effort Regulation, Metacognitive Self-Regulation, Help Seeking, and Control of Learning Beliefs. To determine student content acquisition, we wrote a 12-question survey based on the *SCS1* assessment [7].

IV. Results

Across all sections, concepts test scores increased from the start to the end of the semester, with an average starting score of 29% and an average ending score of 42%. Overall, students in Scrumage sections (N=18) outperformed traditional-section students (N=24), with an ending average score of 51% versus 36%, suggesting better content learning. Fig. 1 shows how the distribution of scores changed from start to end using a kernel density estimate (KDE) for Scrumage (left) and Traditional (right) sections, with the solid curves showing starting scores and the dotted curves showing ending scores. In both pedagogies, improvement is evident, i.e., the dotted curves are shifted to the right of the solid curves. However, improvement among Scrumage students was greater on average and more uniformly distributed than among traditionally-taught students.

The learning attitudes survey results show several key differences between the pedagogy methods as well. Fig. 2 shows the average change in score from pre- to post-test for each attitude category for 32 Scrumage students and 25 Traditional students. Generally, changes were not dramatic but typically moved in different directions, other than Metacognitive Self-Regulation (strategies for planning, monitoring, and modifying a student's own cognition), where students from both pedagogies saw a slight average increase.

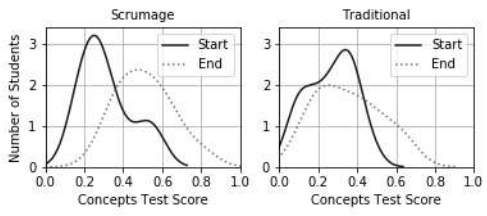

Fig. 1. The distribution of scores from pre- and post-test (solid and dotted lines, respectively) is shown for Scrumage (left) and Traditional (right) using a Kernel Density Estimate.

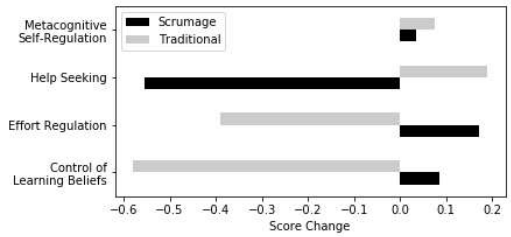

Fig. 2. Within each Attitudes Survey category, the mean score change from pre- to post-test is shown for Scrumage (black) and Traditional (gray).

For questions in the Help Seeking category, relating to student efforts to get help with the material, Scrumage student scores decreased from the start to the end of the course, while traditional student scores increased. We hypothesize that Scrumage students may have been less inclined to seek help due to the focus in Scrumage of providing students with a variety of resources in each sprint, empowering them to find answers on their own. Also, because students typically worked in teams, which offer a built-in mechanism for getting help, they may not have viewed working within their team as "seeking help." For questions in the Effort Regulation category (relating to students' perseverance and work ethic) and Control of Learning Beliefs (addressing students' feelings of being capable of learning the course material), Scrumage scores mildly improved on average, while traditional scores dropped more precipitously. It appears that many students in traditional computer science courses come to doubt their own ability to learn the material, irrespective of the amount of effort expended, while Scrumage students did not encounter the same frustrations. We conjecture that Scrumage students feel a greater sense of agency in their own learning. These results lead us to believe that using the Scrumage framework may help beginner programmers develop resilience and persistence in problem solving rather than adopting a negative attitude.

References

[1] J. Ehrlinger and E. A. Shain, "How accuracy in students' self perceptions relates to success in learning," *Applying Science of Learning in Education*, p. 142-151, 2014.

[2] Susan A. Ambrose and Marsha C. Lovett, "Prior Knowledge is More Than Content: Skills and Beliefs Also Impact Learning," *Applying Science of Learning in Education*, pp. 7–19, 2014.

[3] D. H. Rose, W. S. Harbour, C. S. Johnston, S. G. Daley, and L. Abarbanell, "Universal design for learning in postsecondary education: Reflections on principles and their application," *Journal of postsecondary education and disability*, vol. 19, no. 2, pp. 135–151, 2006.

[4] D. Rose, "Universal Design for Learning," *J Spec Educ Technol*, vol. 15, no. 3, pp. 45–49, Jun. 2000.

[5] S. Duvall, D. R. Hutchings, and R. C. Duvall, "Scrumage: A Method for Incorporating Multiple, Simultaneous Pedagogical Styles in the Classroom," in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, Baltimore, Maryland, USA, 2018, pp. 928–933.

[6] P. R. Pintrich, T. Garcia, W. J. McKeachie, and D. A. F. Smith, *Motivated strategies for learning questionnaire*. Regents of the University of Michigan, 1991.

[7] M. C. Parker, M. Guzdial, and S. Engleman, "Replication, Validation, and Use of a Language Independent CS1 Knowledge Assessment," in *Proceedings of the 2016 ACM Conference on International Computing Education Research*, Melbourne, VIC, Australia, 2016, pp. 93–101.