Redesign of a Gamified Software Engineering Course

Step 2 Scaffolding: Bridging the Motivation Gap

Christoph Thomas (student), Kay Berkling
Department of Computer Science
Cooperative State University Baden Württemberg
Karlsruhe, Germany
thomas-christoph@gmx.de, berkling@dhbw-karlsruhe.de

***Abstract*—This paper builds on work previously published as best paper at CSEDU 2013 [1], which describes the motivation gap between the teacher's view of student motivation and their actual motivation. As a result of this mismatch, the gamified Software Engineering course under observation [2], did not appeal to the students in the expected way. Our findings give rise to a number of design criteria for e-Learning and blended course that address the motivation gap in order to increase the number students that are able to self-regulate their studies and stay or become motivated by the course as defined by Dan Pink's motivational factors: autonomy, mastery and purpose [3].**

Keywords-component; motivation, e-learning, blended learning, gamification, software engineering, education

I. Introduction

Software Engineering is offered during the second of three years towards earning a Bachelor Degree. As the University is based on a quarter system, where students earn salaries and combine work and study in alternating quarters, our student body exhibits typical student behavior in an exaggerated form. Therefore, lending itself to the study, yet allowing the findings to generalize across institutions. A traditionally taught class was gamified to increase student participation and motivation. Findings regarding the motivation gap that lead to the failure of this approach lead to the conclusion and the redesign that is described in this work.

The author of this paper is a Bachelor student and therefore in a unique position of understanding both the researching point of view as well as the student point of view. In this paper, the reader will find approaches to leading students that have been spoon-fed for over 12 years towards becoming independent, responsible students in an open, blended learning environment. It is easy to motivate the good students. The goal of this paper and the presented course design is to not only approach the good students but additionally, to reach out to those students who will have a really hard time with making this transition.

The rest of this paper briefly reviews the current setup of the course and the findings regarding motivation. The Design criteria for scaffolding towards self-regulated learning, which holds true both for blended as well as e-learning courses are then developed in order to evaluate three different common platforms with respect to these criteria. Finally, a platform is chosen and proposed within a specifically designed environment to enhance student learning for the next round of Software Engineering starting this Fall 2013.

II. Previous Work and Status Quo

Students at our University sit in the same room for an average of 5 hours a day, listening to frontal lectures in cohorts of about 30 students. This passive monotony leads to little engagement from the student body during class time. To improve engagement, the five hours of Software Engineering class were redesigned. The vast differences regarding background knowledge and the spread of interest was also addressed with the current design. In order to achieve the goal of engagement, fundamental ideas, shown to be successful motivators in the area of gamification, were included.

In order to integrate gamification ideas into the classroom, the redesign of the course concentrated on the assumption that students are motivated by three human key motivators, also used in game design: Autonomy, mastery and purpose [3]. These motivators are implemented by organizing course content along levels of achievement and giving responsibility back to the students by allowing students to level up autonomously at their own speed and in their own order after achieving mastery of a level. Another important component in certain games is that of collaboration, which was supported through a point system. These points, mistakenly, as it turns out, did not however connect to their grades because they were supposed to reflect the "hero" status. The integrated ideas had to be supported with both technology and important common time in the classroom with the intent of promoting individual interaction and positive peer pressure.

Table 1 lists the gamification components that were implemented into the classroom, a more detailed description is given in another paper, submitted to ICL 2013 [2].

TABLE I. GAME MECHANICS AND DYNAMICS

Game Dynamics	Game Mechanics
Reward	Points
Status	Levels, Paths and Progress
Achievements	Challenges
Competition	Leaderboards
Altruism	Gifts & Sharing
Processes: Interaction with other people Clear path and progress bars Autonomy, Purpose, Mastery	

During the latter part of the course a student survey was conducted. Out of 90 students, 59 answered. As a general rule, students did not receive the gamification ideas in a positive light. A detailed analysis of motivational factors for this student body [1] illuminates the misconceptions regarding student motivation in setting up the gamified classroom. Students' view of autonomy meant that they had the right to not listen during a frontal lecture, mastery was not perceived to be relevant in the framework of a course with grades and the purpose of getting ready to be an excellent member of a project team seemed to be unattainable. Despite being excellently prepared for the exam, as was proven by the high grades, students panicked and felt especially badly prepared. This might be attributed to the Dunning-Kruger effect [4]. Based on our analysis on what went wrong regarding the gamification factors, this paper proposes changes to the course design with the goal of capturing an increasing number of students to be self-motivated, engaged, self-regulating learners.

III. THEORETICAL BACKGROUND

This paragraph briefly introduces a variety of pedagogical aspects that are important foundations of the proposed course design. They are explained in isolation but will be integrated later on in the paper.

A. Course Content for Software Engineering

Topics covered in this Software Engineering course are structured into three pillars of three topics each: Software writing (Design Patterns, Metrics, Testing), Communication (Documentation, Estimation, Reverse Engineering) and Project Management (Processes, Configuration Management, Lifecycle Management). Mastery of the topics culminates in a team-based project experience. This approach gives students enough time to learn all aspects of Software Engineering before applying the collective know-how in a project. Each of the topics was further structured internally into cognitive levels according to **Bloom's taxonomy**, building towards higher level thinking skills. This approach has been used effectively in Computer Science in the past [5][6].

B. Learning Environments

In the **traditional classroom**, twenty or more students are expected to listen to a lecturer. Some disadvantages of a traditional classroom can be the following: The lecturer cannot adapt the teaching style to the students' individual learning needs, the students have to follow the teacher's speed and cannot choose their own learning speed, most of the time I used to hold the lecture and involvement of students s often non-existent.

In the **flipped classroom**, internet technology is used to reduce the amount of input time in the classroom and gain more time for interaction with students [7]. This means that students watch learning videos outside of class and then come prepared to the classroom for a discussion or another form of interaction, such as project work in groups with the teacher as a mentor.

Blended learning refers to all forms of teaching and learning supported by the computer and communication technology [8]. It can be used in-classroom and out-of-classroom and be guided by a teacher or self-paced. It includes features like classroom communication, assignments and wikis. A popular example of an e-learning platform is Moodle. Moodle is open source and can be downloaded and configured for free. With Moodle it is possible to create classrooms, upload learning materials, give assignments and communicate with students.

Social networks can facilitate interaction, communication, and collaboration. Therefore, they are believed to have a high potential as a learning environment [9]. Features of social networks often include user profile sites, private messaging, commenting on other users' contents and adding other users as friends. So users can share sites and other resources they find interesting, funny or helpful and their friends can look at the new content and comment on it or share it with their friends. Classroom time is often neglected as a potential environment to be leveraged as a powerful face-face social network.

Massive Open Online Courses ("MOOCs") are lately offered by universities for students who are unable (in time, location or funding) to study at an elite university. The students can take different online courses, hand in materials via E-Mail or file upload and thus are achieve able to achieve a free education, so far without attaining a degree such a Bachelor. Problems of MOOCs are the very high drop-out rate of 80-90%. Self- regulation and motivational issues that play a role in the regular classroom may be enhanced in this setting and play a major role in finishing a course as there is less pressure and no external motivation of a diploma compared to a real university course. This is called the funnel of participation [10].

C. Learning Theories

There are a number of learning theories that come into play in the current design of the course.

Collaborative learning describes a learning activity of two or more people [11]. It is opposed to individual learning and suggests that learners communicate and work with each other to use each other's skills and resources [12]. The collaboration can take place sitting face-to-face in a classroom or chatting and discussing together in a virtual classroom over the internet. Using clickers in the classroom as part of the frontal lecture, for example, has shown that it is possible to promote collaborative learning even in more restrictive frontal classroom [13].

Constructivism is a learning theory that describes how knowledge is constructed. Having its roots in philosophy, psychology and cybernetics [14], it explains the construction of new knowledge as a product of information and existing knowledge created by experience. This construction is reinforced if the learner is actively involved. The teacher has to become a facilitator by providing tasks and materials for the students instead of just giving a lecture [15]. This part comes into play for the project based course work in the classroom under discussion.

Connectivism regards learning within a social network in that it explores knowledge with self-organizing networks. Learning is as such not controlled either by the learner or the teacher. Learning comes from the people and materials around us. Connectivism relies on rapidly changing environment such as is the case in modern software engineering. Thus, new information is continually acquired. It is important to gain knowledge in this network, distinguish what is important and how this impacts the world [16].

Scaffolding refers to a learning process with an amount of support given by the teacher to the student intending to help the student to achieve the learning goals [17]. The amount of support decreases proportionally to the increasing development of the student's autonomous learning capabilities. One of the key factors that was neglected during the original course design, leaving behind the weaker students while letting the strong ones excel even more.

D. Learning Styles

According to Susan A. Santo [18], there is no generally accepted definition for learning styles despite the fact that many different learning style models exist. For the purpose of this paper, Grasha's definition of a learning style as somebody's preferred way of learning [19] is sufficient. In the following, two different learning style models will be briefly described. According to the Grasha-Riechmann Student Learning Style Scales [20], there are six styles that can be differentiated amongst learners:

The participant learner is very interested in the course content and asks questions.

The avoidant learner works as little as possible or only shortly before a dead-line.

The independent learner works on his/her own and rarely asks for help.

The dependent learner needs lots of support and detailed instruction.

The collaborative learner prefers working in a team.

The competitive learner wants to do better than other course participants.

The second learning style model is Kolb's Learning Style Inventory. It defines learning as a cycle and differentiates students based on their preference for one of the following four parts of the learning cycle [21]:

In the Concrete Experience part, students learn from feeling and practical experience.

In the Reflective Observation part, students learn by observation e.g. by watching and listening.

In the Abstract Conceptualization part, students learn by thinking e.g. by analyzing problems and developing solutions.

In the Active Experimentation part, students learn by doing e.g. by implementing a solution.

IV. REALITIES OF A RESTRICTED ECO-SYSTEM

In the following paragraph, the constraints of the present "eco-system" under which the University and the students operate will be described in more detail.

A. Student Background and Motivation

The Software Engineering course takes place during the second year of study and most participants have had at least a little programming experience with one semester Java programming and web engineering as well as having completed their first two sessions at work. Software Engineering is a key discipline of their studies and students see an opportunity to apply this subject professionally, learn from best practice and to do interesting project work, coding and project management. Catering to such a large spread in know-how of the students has an impact on the design of the course.

B. Learning Environment

Software Engineering has a workload of 270 hours, split into 96 hours of attendance and 174 hours of homework. There are two lectures per week (5 hours) for two quarters, with a quarter in between that is spent at the work place. Given the experience of the author this workload seems to assume that a student is able to work 24/7 and it can be said that the workload is perceived as extremely high by the students. This leads to problems concerning necessary strategies of efficiency on the part of the student that affect the classroom and need to be taken into account for the course design. All courses at this University are taught in the traditional frontal style. Next year it will be redesigned based on the findings after testing out a gamified version of the course last year.

C. e-Learning Platform

Most classes are taught without any e-Platform as support. Moodle is used as a content management platform in some few cases. For Software Engineering, a combination of online quiz taking tools, Moodle as a "content management system" and yet another platform for gamification aspects were used in the classroom [2]. This multi-platform approach proved to be very difficult for the students. As a result, it is important for the redesign to prefer one tool (and compromise on functionality) over a combination of expert tools. This experience is taken into consideration when evaluating the software platform.

D. Learning Theories

In frontal lecture, there really is no underlying learning theory. In the gamified version Bloom's taxonomy was used in combination with a form of Connectivism. It was assumed that learners are able to move up the Bloom's hierarchy from simple problem sets to more abstract higher level thinking skills simply by being provided with the relevant problems. Without scaffolding or explicit awareness that learning opens insecurities (Dunning-Kruger effect), the massive change to constructivist or collaborative setting was too overwhelming for most.

E. Learning Styles

It was assumed that all learners are happy to work independently. Support consisted of providing learning paths, material and individual feedback on quizzes as well as lectures on demand. This assumes that the learners were competitive and happy to collaborate if needed and ready to reach the learning goals set out for them. This did not account for the large variety of learning styles outlined in the previous section.

V. Design Criteria

Based on the experience of teaching the Software Engineering course and the above outline of realities, this section will derive the design criteria for the redesigned course, in order to bridge the motivation gap. The following changes will be made, taking into account the various learning styles and learning theories combined with the previous experience and feedback from students. This will allow for small changes in a student dependent way [22].

A. e-Learning Platform

1. Material has to be of the **highest quality** and located in one platform only to minimize confusion.

B. Learning Theories

2. **Project work and teams** have to be established from the start to help keep people on track and keep up the motivation by seeing that everyone is working. Team-level competitiveness is easier to manage and draws in more students.

3. **Higher-level thinking skills** that do not have one answer to be memorized need to be clearly marked. Dealing with these is a learned skill that takes time.

4. Due to the well-known **Dunning-Kruger effect** (CITE), a cognitive bias that gives students the illusory feeling of superiority, student feelings towards a course may be more negative once they have gained the metacognitive ability to recognize their own limitations. Making these feeling explicit by raising awareness may alleviate their struggle as actual competence weakens their self-confidence and reflects negatively on their view of the learning experience.

C. Learning Styles

5. While few students welcome the new, open learning environment, the majority is overwhelmed, feel left alone and **need scaffolding**. This is done mainly by providing grading for self-regulating activities and granting the right to independence while supporting a fixed lecture series. Students practice planning and sign short-term independent study unit contracts.

6. **Motivators** (autonomy, mastery, purpose) that are considered generic human characteristics need to be **reconsidered** in a constrained eco-system of student life. A sort of “un-schooling” has to take place that takes time.

7. **Gamification elements** (motivators in point 5 above, levels, badges, collaboration and “heroes”) have to be **subtle** as their overt use carries connotation of non-seriousness for students and is therefore deemed inappropriate.

The goal of the course design is to provide the necessary path in order to carry a larger number of students from a passive member of the classroom over to an active student taking responsibility in their own learning that takes less emphasis on grade and higher importance on skills and achievement. Thereby bridging motivation gap and preparing the students for life-long learning and release much-needed independent thinkers into the workforce. Based on the analysis above, the derived requirements of the tool include support for the following:

TABLE II. Design Criteria

Support collaborative aspects: Forum, peer review, teams
Visualize Blooms’ taxonomy: Coding problems by level, peer reviews for higher level problems
Cater to learner types: High-level material in various forms, independent speed, strong scaffolding support
Scaffold towards motivation: Support for autonomy, purpose, mastery
Reward: Positive feedback for achievers in some form, marking progress, visible performance, immediate feedback on lower
Onboarding: immediate successes, immediate feedback for Bloom’s lower-level problems

Finally, a very important criterion from the teacher point of view with a high course load must be the ease of installing and maintaining the platform.

VI. e-Learning Platform

In this section, three platforms for the implementation of a Software Engineering flipped classroom are briefly described and evaluated with respect to past experience and design criteria. Each platform is described within the intended blended learning approach.

A. Moodle

Moodle is a well-known, widely used online learning platform that allows teachers to create virtual class-rooms, often used simply as a content management tool. It is available at the University without any further effort and students are used to working with it in a limited way. Usually, students receive lecture content such as slides and upload assignments for correction by the teacher. Moodle provides a wiki and other features like surveys, communication between students or teachers and collaboration for project work that is only rarely used at our institution. Table 2 shows how the design criteria are supported within Moodle.

Moodle supports “workshops”, submissions of homework that can then be peer evaluated. A lot of recent research has gone into peer evaluations and making these valuable [23].

TABLE III. CRITERIA MOODLE

Collaborative aspects:	- forum, workshops, some support for teams
Bloom's taxonomy:	- peer reviews, lower level automatic quiz grading
Learner Types:	- independent speed, scaffolding not inherently supported but can be integrated into material through color coding. Movies are not easily integrated other than through links.
Motivation:	- through material preparation by teacher.
Reward:	- difficult to demonstrate progress, grade can be seen by students, immediate feedback on automatically graded quizzes, somewhat delayed feedback on higher level assignments through peer-grading and teacher feedback.
Onboarding:	- up to teacher to prepare material appropriately

Regardless of how valuable the peer review is to those being evaluated, the simple act of evaluating someone else has been shown to engage the student in a deeper understanding of the criteria of completing good work. Using Moodle fully, including workshops, forums as well as automatically graded quizzes supports individual learning speed (all information is available, online quizzes are open and repeatable at any time), collaborative work (forum) and higher-order thinking skills (workshops). In this respect, Moodle supports most design criteria for this course.

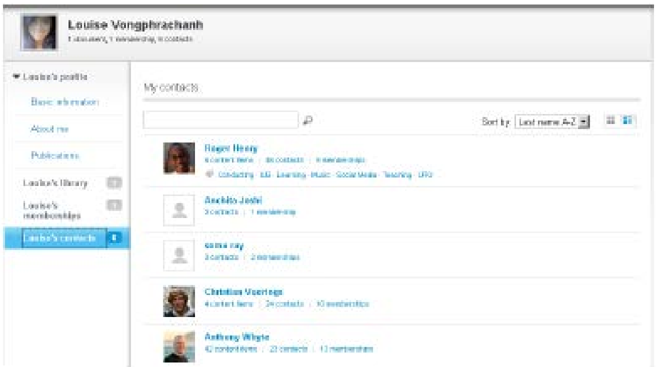

Figure 1: Sakai User Profile

B. *Sakai Project*

While Moodle is available and installed at the University and students are used to it, the Sakai Project [24] is an open-source software that supports teaching, learning and research and would need to be installed and maintained by the teacher of the course.

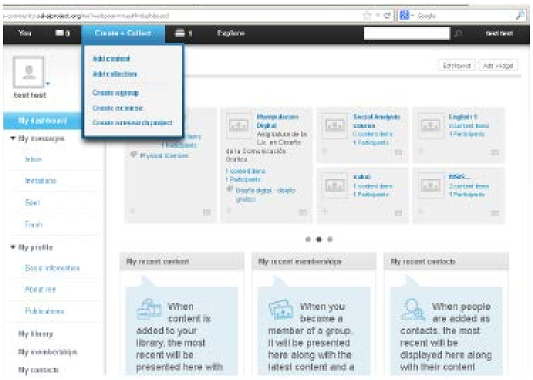

Figure 2: Sakai Dashboard

One of its products, Sakai Open Academic Environment (in the following "Sakai OAE") is aimed to facilitate social academic collaboration and is therefore based on a social network. Sakai OAE also provides an online demonstration site where the following screenshots are taken [25]. Using Sakai OAE, every user has his/her own profile where he/she can add personal information and find his contacts to other users as shown in Figure 1. A user sees a dashboard after logging in. This dashboard displays interesting content from other users and provides information about recent activities, see Figure 2. In this figure, one can also see a messaging function and the possibility to add content or to create a group. Even if the demonstration site looks already very promising, Sakai OAE is still under development. Additionally, its current setup procedure is quite complex and the documentation does not help that much. Table 3 shows the evaluation criteria for Sakai.

TABLE IV. CRITERIA SAKAI

Collaborative aspects:	- messaging, many team functions, peer reviews by commenting, sharing
Bloom's taxonomy:	-scaffolding not inherently supported but can be integrated into material through color coding.
Learner Types:	- independent speed, movies are easily integrated embedding YouTube videos.
Motivation:	- through material preparation by teacher, through comments from course mates.
Reward:	- difficult to demonstrate progress, no grading function, featuring student content, delayed feedback on higher level assignments through peer-grading and teacher feedback.
Onboarding:	- up to teacher to prepare material appropriately

C. *CourseSites*

CourseSites is an online platform that allows teachers to create online courses to support their lectures or to create a MOOC for free. One teacher can have up to five courses at one time with total file storage limit of 500 MB per course. The course setup for teachers is very easy. It starts with a quick setup guide, see Figure 3 where one can choose the course title, provide a course description and choose a course structure which defines the menu structure and activities for the students. In Fig. 4, the chosen course structure is Constructivism.

After completing the quick setup guide, a teacher can easily add content and assessments and customize his/her course using a comfortable user interface. When a student joins a course on CourseSites, he/she finds a dashboard with elements like a weekly planner.

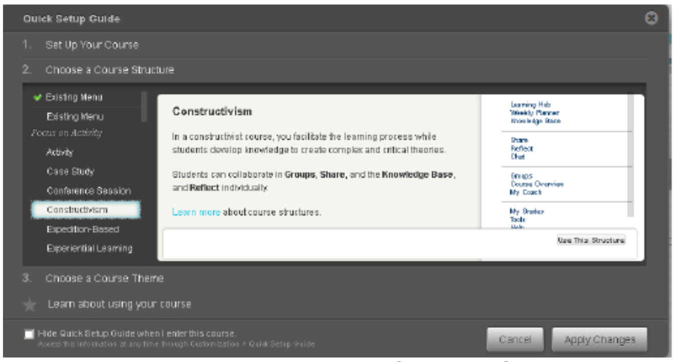

Figure 3: CourseSites Quick Setup Guide

This weekly planner provides an overview about the tasks, deadlines and announcements for the course. Students can share their course experience on a blog or found a group to discuss and work with other students.

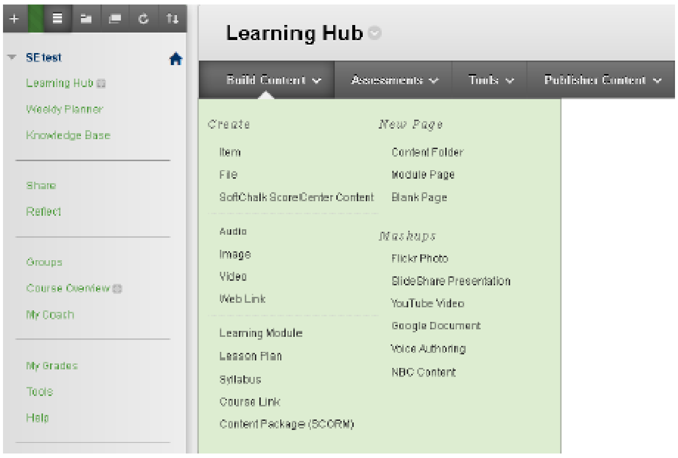

Figure 4: CourseSites content creation

The platform provides lots of group functions including file exchange, a group blog, a discussion board, group tasks and a group wiki. Table 4 shows the evaluation criteria for CourseSites.

TABLE V. CRITERIA COURSESITES

Collaborative aspects:	- forum, wiki, messaging, many team functions
Bloom's taxonomy:	- peer reviews, lower level automatic quiz grading, blog
Learner Types:	- independent speed, scaffolding not inherently supported but can be integrated into material through color coding. Movies are easily integrated embedding YouTube videos.
Motivation:	- through material preparation by teacher, difficult to demonstrate progress, grade can be seen by students, immediate feedback on automatically graded quizzes, somewhat delayed feedback on higher level assignments through peer-grading and teacher feedback.
Reward:	- up to teacher to prepare material appropriately
Onboarding:	- platform offers an introduction course to learn more about the platform's use

VII. IMPLEMENTATION

The chosen platform should fit most of the requirements outlined in Section V, it should be easy to set up for the teacher and it should support class time to be spent on the higher levels of blooms taxonomy with diverse learning styles to support deep learning.

A. *Learning Environment*

The flipped classroom is implemented with collaborative learning in the classroom to leverage in-class time with real contact, one of the benefits of non-online education while still using student time beneficially. The chosen platform serves in several ways, allowing everyone to create and share content, create teams and define goals from the beginning in order to provide enough guidance and scaffolding to the majority of students who need this. Without this platform, personalization and independence is not possible. Without the in-class time and schedule, explained below, scaffolding students is more difficult.

B. *Schedule*

The intended schedule for a learning unit includes the following five steps inspired by Bloom's taxonomy, learning theories, learner types, creating well-paced changes to the traditional eco-system.

1. Students are provided with the learning materials for the Flipped Classroom watching a video or reading a paper as an introduction to the learning unit.
2. Students answer a short quiz about the content of the learning material. The answers are prepared in every team for about ten minutes. Afterwards, the teams share their answers in a common document which is taken for the exam preparation.
3. Students work on a team assignment to apply the new knowledge from the learning material and hand in the outcome for peer review.
4. Students analyze and reflect the project progress of the other teams by peer reviewing the submitted documents from step 3.
5. Students evaluate the learning profit thinking about problems, benefits and the findings from the peer review. This evaluation takes place in the teams and is documented by a blog entry and is guided by a catalogue of questions that every blog entry should cover.

TABLE VI. DESIGN CRITERIA

Learning theory/style	Supported by method
Collaborative learning	Team work for blog writing and project work
Constructivism	Blog writing and project work
Connectivism	Peer reviews and communication
Scaffolding	By preparation of the materials to give clear instructions and by requiring students to form teams
Critical Grasha-Riechmann Student Learning Style Scales	Avoidant and dependent learner supported by regular deadlines, team work and lecturer as a mentor
Kolb's Learning Style Inventory	- Concrete experience by project work - Reflective observation by learning material - Abstract Conceptualization by peer review and writing blog entry - Active Experimentation by project work

Students are responsible for the preparation (step one) outside of class. This should require no more than twenty minutes, otherwise the students' acceptance will suffer. The second step takes place at the beginning of class and is limited to a dura-tion of thirty minutes (the team preparation time of about ten minutes included). Steps three to five follow in class so that students can benefit from the teacher's presence as the so called "guide on the side".

Students can additionally build up a portfolio of useful tools and skills, e.g. setting up a bug tracker and document how to use it. Considering the theory of scaffolding, towards the end of the course, there will be more open and individual project work requiring the students to become more independent.

The grade consists of the following parts:

- The team project work is graded by the docent (40 %).
- The peer review grades from step four (10 %).
- The team blog entries from step five are graded by the docent (20 %).
- An exam is written based on the quiz questions from step two (30 %).
- A good portfolio can earn up to 10 % bonus.

Table maps how the chosen method onto the different learning theories and styles.

C. *Chosen Software Platform*

Table 6, due to its size at the end of this paper, compares all three tools according to the guidelines. It can be seen that CourseSites is the best choice to implement the SE Flipped Classroom with no blockers (red text) and no warnings (yellow text).

The Software Engineering Flipped Classroom Prototype has been designed as a blended learning environment with team work based on Bloom's Taxonomy and a mixture of Connectivism and Constructivism. For the software support of the implementation, the free hosted online platform CourseSites was selected. The teacher can now create a course on this platform and create content and assessments for the Software Engineering courses starting in October. Key to choosing a tool is to reassure that it supports the design criteria and necessary processes in the classroom delineated in the schedule above. In that sense, CourseSites is replacable by any MOOC platform that supports scaffolding towards creating motivated, self-regulating mature learners.

D. *Extendability towards Gamification*

The extendibility towards gamification elements is currently limited to badges. CourseSites provides an integration of Mozilla Open Badge. Using this integration, students can be assigned badges for completing tasks. Whether or not these badges are important to students remains to be seen for this new context.

VIII. CONCLUSION

In this paper we built on previous experience with students in a very constrained classroom setting. It was shown that students' motivations are different from those naturally expected in other environments like sports or hobbies. They do not naturally pursue autonomy, self-regulated learning, or purpose, such as completing the best project, or mastery as it is independent of the grade or takes up extra time that is better spent on other courses. Students are quite aware that grade and mastery are not equivalent. Due to years of conditioning and the realization that a bad grade has a real influence on their future the student is motivated by doing as little as necessary to obtain the best possible grade, an intelligent survival strategy. As a result it has become clear that the design for the course has to include very clear guidelines for scaffolding towards "un-schooling" through the use of excellent material and endorsing self-regulating work. This has to take into account the Dunning Kruger effect that increases students' dissatisfaction as they become conscious about all the material that they have not yet mastered.

Supporting independence requires the use of underlying technology. However, this technology is only fulfilling its purpose of supporting the required learning environment if it supports scaffolding students through supporting teams, easing into higher-order thinking skills, supporting a variety of learner types, rewarding autonomy, mastery and purpose by relating these directly to grading and thus rewarding hard work, providing justice for all by disallowing students to disengage. Easy onboarding for a single platform with excellent material is essential. Support for individual plans and goals is important. These design criteria have been developed out of the experience of gamifying the software engineering course in 2012. These findings are equally important in large online class settings, where it is important to keep the drop-out rate as low as possible, catering not only to the strong students but paving the way for increasing the number of students who are comfortable with self-regulated learning with awareness of the Dunning-Kruger effect for their own reflection.

This paper forms the basis for a redesign of the course. While maintaining gamification elements, these are not emphasized. Rather than simply looking at grades, evaluation criteria for success will depend to a large extend on the percentage of students who will be able to say that they feel comfortable with the new learning environment. If this is the case, then we have succeeded in creating graduates that are not degree seekers but knowledge seekers.

TABLE VII. COMPARISON OF TOOLS ALONG DESIGN CRITERIA

Criteria/ Product	Moodle	Sakai OAE	CourseSites
Group support	Groups have to be enabled by the do-cent	Everyone can create groups	Groups have to be enabled by the do-cent
Profile/ dashboard	Yes	yes	yes
Grade overview	Yes, grading has to be configured for every assessment	Not included by default	Yes, grading has to be configured for every assessment
Content sharing	Within groups or in the forum	On personal site or in groups	Within groups or in the forum
Forum	Yes	Discussion within a group	yes
Usability/ Design	Getting better compared to older versions	Good	Good
Availability	free, already hosted by the DH	free, currently still under development	free, hosted by coursesites.com
Collaborative aspects	forum, workshops, some support for teams	messaging, many team functions, peer reviews by commenting, sharing	forum, wiki, messaging, many team functions
Bloom's taxonomy	peer reviews, lower level automatic quiz grading	Analytic comments, posts	peer reviews, lower level automatic quiz grading, blog
Learner Types:	independent speed, scaffolding not inherently supported but can be integrated into material through color coding. Movies are not easily integrated other than through links.	independent speed, scaffolding not inherently supported but can be integrated into material through color coding. Movies are easily integrated embedding YouTube videos.	independent speed, scaffolding not inherently supported but can be integrated into material through color coding. Movies are easily integrated embedding YouTube videos.
Motivation	through material preparation by teacher	through material preparation by teacher, through comments from course mates.	through material preparation by teacher
Reward	difficult to demonstrate progress, grade can be seen by students, immediate feedback on automatically graded quizzes, somewhat delayed feedback on higher level assignments through peer-grading and teacher feedback	difficult to demonstrate progress, no grading function, featuring student content, delayed feedback on higher level assignments through peer-grading and teacher feedback.	difficult to demonstrate progress, grade can be seen by students, immediate feedback on automatically graded quizzes, somewhat delayed feedback on higher level assignments through peer-grading and teacher feedback
Onboarding	up to teacher to prepare material appropriately	up to teacher to prepare material appropriately	up to teacher to prepare material appropriately, platform offers an introduction course to learn more about the platform's use

IX. Bibliography

[1] K. Berkling and Zundel, A, Understanding the Challenges of Introducing Self-driven Blended Learning in a Restrictive Ecosystem – Step 1 for Change Management: Understanding Student Motivation, CSEDU 2013, 5th International Conference on Computer Supported Education, SciTePress, to appear 2013.

[2] K. Berkling and Ch. Thomas, Gamification of a Software Engineering Course -- and a detailed analysis of the factors that lead to it's failure. Submitted to ICL 2013, 16th International Conference on Interactive Collaborative Learning and 42 International Conference on Engineering Pedagogy, 2013.

[3] D. H. Pink, Drive: The surprising truth about what motivates us. Canongate, 2010.

[4] J. Kruger and D. Dunning, Unskilled and unaware of it: How difficulties in recognizing one's own incompetence lead to inflated self-assessments. *Journal of personality and social psychology* 77.6, 1999, pp. 1121-1134.

[5] D.R.Krathwohl, A revision of Bloom's taxonomy: An overview. Theory into practice, 41(4), 2002, 212-218.

[6] E. Thompson, A. Luxton-Reilly, J.L. Whalley, M. Hu, and P. Robbins, Bloom's taxonomy for CS assessment, *Proceedings of the tenth conference on Australasian computing education-Volume 78* (pp. 155-161). Australian Computer Society, January 2008.

[7] Available http://www.flippedclassroom.com/, retrieved: 16.04.2013.

[8] H. Singh, Building effective blended learning programs, Educational Technology-Saddle Brook Then Englewood Cliffs, *NJ*- 43.6, 2003, 51-54.

[9] G. Veletsianos and C. Navarrete. "Online social networks as formal learning environments: Learner experiences and activities.", *The International Review of Research in Open and Distance Learning* , 13.1, 2012, pp. 144-166.

[10] Clow, Doug (2013). MOOCs and the funnel of participation. In: Third Conference on Learning Analytics and Knowledge (LAK 2013), Leuven, Belgium, 2013, pp. 8-12.

[11] P. Dillenbourg, Collaborative Learning: Cognitive and Computational Approaches. Advances in Learning and Instruction Series. Elsevier Science, Inc., PO Box 945, Madison Square Station, New York, NY 10160-0757, 1999.

[12] M. M. Chiu, Group Problem-Solving Processes: Social Interactions andIndividual Actions.,Journal for the theory of social behaviour 30.1, 2000, pp. 26-49.

[13] Schmidt, B. Learning Strategy and Students' Perception of Different Learning Options in a Blended Learning Environment, CSEDU 2013, 5th International Conference on Computer Supported Education, SciTePress, to appear 2013.

[14] E. Von Glasersfeld, Constructivism in education, Oxford, England: Pergamon Press. , 1989, p. 162.

[15] H. Bauersfeld, The structuring of the structures: Development and function of mathematizing as a social practice. Universität Bielefeld, Institut für Didaktik der Mathematik, 1990.

[16] G. Siemens, Connectivism: Learning and knowledge today, *The International Review of Research in Open and Distance Learning* 9.3, 2006, pp. 1-13.

[17] Sawyer, R. Keith, ed. *The Cambridge handbook of the learning sciences*. Vol. 2. No. 5. Cambridge: Cambridge University Press, 2006.

[18] Santo, Susan A. "Relationships between learning styles and online learning." *Performance Improvement Quarterly* 19.3, 2006, pp. 73-88.

[19] Grasha, Anthony F. "A matter of style: The teacher as expert, formal authority, personal model, facilitator, and delegator." *College teaching* 42.4, 1994, pp. 142-149.

[20] Fuhrmann, B. Schneider and A. F. Grasha. A practical handbook for college teachers. Boston: Little, Brown, 1983.

[21] D. A. Kolb, Experiential learning theory and the learning style inventory: A reply to Freedman and Stumpf. Academy of Management Review, 1981, pp. 289-296.

[22] J. Tagg, *The Learning Paradigm College*. Anker Publishing Company, Inc., 176 Ballville Road, PO Box 249, Bolton, MA 01740-0249, 2003.

[23] K. van Overveld and T. Verhoeff, Self-consistent Peer Ranking for Assessing Student Work - Dealing with Large Populations, CSEDU 2013, 5th International Conference on Computer Supported Education, SciTePress, to appear 2013.

[24] Available online: http://www.sakaiproject.org/sakai-products, retrieved: 24.04.2013.

[25] Available online: https://oae-community.sakaiproject.org/, retrieved: 25.04.2013.