The Teacher's Role in Gamification in Software Engineering at Universities (Field Report) - or how geeks can be inspired to sing

Martine Herpers
Computer Science Dept.
Fulda, Germany 36037
martine.herpers@cs.hs-fulda.de

Abstract

New educational methods require new competences from university teachers. In a non-technical seminar with included SE aspects for bachelor computer science student's self-determined learning and gamification could improve student's engagement significantly. Both methods seem to be a good candidates for technical courses in software engineering, too. The article highlights the new role and competences required for university teachers using gamification.

1. Introduction

Computer technology is changing fast and learned know-how is becoming outdated very soon. Industry very often complains about graduates with low key qualifications. The missing skills concern the ability to analyze and reflect independently, to write and communicate research results, to be team minded and able to solve problems. For universities this means that teaching student's common problem solving methods and self-determined learning becomes more and more important.

A special challenge for university teachers is teaching software engineering (SE). At universities of applied sciences some of the students have first experiences with real product development and can at least imaging the importance of working processes. Most of the students, however, have no idea about the challenges in real software development projects. Also university teachers often have only restricted experiences in real life software development. Most universities request that students are getting work experience by performing hands-on trainings. The success of internships in companies, however, depends heavily on the chosen company and its capability level in software development. For teaching software engineering at university it is very important to motivate students and to develop the key qualifications in analyzing, reflecting and communicating.

University education in computer science is changing since several years and in the meantime it is well accepted, that just offering lectures, exercises or software development projects is not enough. With activating mechanisms like online queries, brief teamwork practices, letting students evaluate each other, etc. student's ability to concentrate and learn can be improved [Rac05]. Such mechanisms are very useful, but are more or less just islands of improved learning experience.

A broader mechanism is the **problem based learning** (PBL)[1]. PBL gives students the responsibility for their learning success supported by the teacher's short lectures about the techniques they need to solve the problem. A very similar method with focus on making students wishing to learn is the **gamification**[2] of lectures. Gamification doesn't mean only to add the aspect of fun to the lectures, it can be used to get everyone involved and to let students learn from each other. One goal of a gamification can be that the best students demonstrate and prove their talents in a contest at the end of the course.

For teaching software engineering at universities two different approaches seem to be promising:

- Using PBL on a project level, where some project steps are allowed to fail, if needed working processes have to be ignored (e.g. project feasibility gate fails, because a stakeholder has been ignored)
- Using gamification for strengthening particular capabilities for improving software quality like finding errors, deriving test cases from given specifications or test driven coding.

There might be other approaches for an activating and intensive learning of working processes at universities, but all these new ways of teaching require a change in the teacher's role [Fle07].

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

[1] A good explanation of PBL can be found in Wikipedia: http://en.wikipedia.org/wiki/Problem-based_learning

[2] Gamification is used since several years in industry to motivate employees. A list of gamification projects can be found at [Bol12].

Teachers must develop management skills enabling them to lead the students through such courses. This article shows the challenges teachers are facing by using these new teaching methods. The course "Presentation and Communication" for computer science students of the University of Applied Computer Science in Fulda, Germany, was held in winter semester 2014/2015 and shows all the requirements which are needed to meet for a successful teaching process. Although soft skills are not the main topic of software engineering, they are very important for every project and therefore relevant for all working activities. Because I am convinced, that teaching soft skills requires similar educational capabilities as teaching technical skills, I recommend the experiences I made during the above-mentioned course for teaching all kinds of competences.

As far as my experience goes soft skills are seen as a "not so important" subject for most of the computer science students. Thus there was a high demand of motivation by the teacher to get the students willing to learn it. My experiences show that using activating teaching methods like group work and telling stories along the software development process worked very well. The gamification I used, however, made students enthusiastic learners showing outstanding results at the end.

2. New skills needed: managing instead of lecturing

At the beginning all my colleagues told me, that I would not be able to activate the students for participating on a communication contest (gamification). In particular, that I wanted them – besides to give a 10-minute presentation, to perform an escalator pitch and to draw a visualization on a flip chart – to sing a song as a choir, was considered to be impossible.

I asked my husband and my daughter (student of mechanical engineering at TU Chemnitz), what they thought about the idea to offer such a challenge at the end of the course. My husband thought, that I was very ambiguous, but that the students couldn't be motivated to sing, perhaps for the other contest categories. My daughter's first reaction was: "No, never!". Her second reaction was: "Well, perhaps it might be fun at least."

In literature emotional and enjoyable learning is recommended [Rac07] and after some discussions with a gamification expert, I was convinced, that gamification would result in great learning experiences for the students. So I was not sure, that it will work, but I tried trusting on my managing[3] experience for many years.[4]

In the following the different phases and the corresponding challenges for me as the teacher during the course are explained in more detail. First it is explained, how the students had been motivated to try something unusual. Finally the challenges for teaching it are described in detail.

2.1 The seminar outline

The seminar "presentation and communication" is a required course for bachelor computer science students at the university at Fulda. It takes place in the third semester. So all students attend to seminars in groups of about twenty persons for four hours a week. The winter semester 2014/15 started in October 2014, paused two weeks for Christmas break and finished in February 2015.

Three groups were trained by me, which offered the possibility to organize a contest at the end of the seminar to find out which group got the best communication skills. Two other colleagues, training two other groups, were asked to participate with their groups, but they had no interest in attending the contest.

2.2 How geeks can be motivated

As professor at a university of applied sciences the question for me was: What is motivating for young geeks coming from high school and for those with working experiences? For both groups it is important to understand the relationship of the topic to be learned to the working practice in their future life. So it was easy to motivate the students along the software development process. Telling stories and work experiences from real life as a software manager I got always highest attention by the students. Below you see the topics of the course and how it can be mapped to steps in the software development process.

---

[3] Managing is understood as modern, team-oriented managing. Often coaching competences are required for effective teachers. Coaching, however, would be too week for the tasks of introducing new educational methods like gamification. A clear goal setting capability with the appropriate authority is essential.

[4] Technical team manager in software development, head of department for consulting software quality have been some career stages before my assignment to professor at the university of applied computer science in Fulda.

Table 1: Motivation by references to software development tasks

Learning goal	**Motivation by references to software development tasks**
Presentation / elevator pitch	Presentation of technical ideas/products in the feasibility phase or for customers
Visualization on flip charts	Discussions during all phases
Feedback giving and taking	Performing a review or inspection, Useful for pair programming or in discussions between quality engineer and developer
Conflict resolution	Very important in communication with customers and between developers and testers
Negotiation	Same as conflict resolution
Intercultural communication	Distributed development teams, interdisciplinary teams

Introducing and using up-to-date educational teaching methods needs, however, additional motivation supporting techniques regarding the group processes. In my case it was very important to form the group to become a team ready for fighting against other teams. I trained three groups. All groups had been set together by random, so there were unfortunately only a few already established friendships among the group members. So I started right from the beginning to strengthen trust between the group members by organizing randomly smaller learning groups (about 5 persons) and to allow self-determined learning experiences. In the mid-seminar review all students mentioned that the learning atmosphere allowed them to get in closer contact to all other group members and they want to continue to work using the activating learning methods. Trust among the group members and me as the teacher was well established at this point in time.

The groups were trained well separated from each other. Every group was different in size, in composition of talents and willingness to actively participate in the communication contest at the end:

- **group A**: 23 persons, medium willingness to participate
- **group B**: 16 persons, low willingness to participate
- **group C**: 21 persons, high willingness to participate

It showed that the following aspects were important to form teams accepting the new and challenging learning environment:

- **Openness**: right from the beginning the uncommon combination of learning and setting a goal to win a contest was clearly communicated and explained. Some were really surprised by the expectation to do a contest and in particular to sing in a choir. I got a lot of comments like: "Don't want to sing" and only very few saying "Singing is fun". The contest itself seemed to be mostly accepted right from the beginning, the choir performance wasn't.
- **No doubt about the new format**: The contest was not discussed in the course, neither the choir. As the teacher I outlined the learning effects, generated an officially looking announcement of the contest and organized the eLearning video team to record the contest. In the first half of the seminar the contest was no topic. This time was used to build trust among the students and to learn communication skills. In the middle of the seminar I set up four teams one for each category (presentation, elevator pitch, visualization, choir) of the contest.
- **Creating trust among the group members**: In order to get familiar with each other different team set ups were done. After getting familiar with the new way of working in always new team set ups, the students looked forward to the tasks and the set ups of the next lesson.
- **Supporting the learning activities**: The very active learning style was new for the students. So it was necessary to support them at the beginning by showing them that supporting each other is a wished behavior and nobody is losing his/her dignity in doing unpopular or uncommon things for students like fetching the flip chart or summarizing the results. After the first phase, where I organized the paper for the flip chart or looked for the beamer, all students became familiar with all needed tools and started to support each other.
- **Allowing not to participate at the contest**: the course was organized as a seminar with

compulsory attendance for at least 80% of all lessons. So the students had the possibility of not attending the contest, which was just one of the lessons. At the beginning some of the students planned not to attend. But with growing trust in the other group members and in the teacher, that the contest is not relevant for the grade at the end of the seminar, they accepted the new format.[5]

- **Making it a big event**: The event had to be moderated appropriately with some fun and the without the feeling that somebody could fail. Due to this it was explained, that in the eyes of the teacher contributing was more important than winning and only the fact of contribution would be clearly valued. Furthermore we organized some soft drinks and cookies for our convenience. The importance of the performance war underlined by the fact that all should be recorded. That enabled me also to analyze the activities afterwards and to celebrate the performance with the students.

At the end only two persons didn't attend the contest and two more hadn't been on the stage singing in the choir. Even group B was attending. But the winning team was the one with the highest willingness to participate right from the beginning.

2.3 Challenges for the teacher

As described above the activating teaching methods in particular the gamification need new modern educational approaches of course organizing. Particularly the organization of the contest itself as a big event demands more effort from the teacher as performing a normal lecture. Organizing the team work during the course requires as much time and efforts as developing conventional exercises and correcting them.

In terms of the overall effort the gamification requires perhaps some more preparation as a normal lecture.

The real challenges are others. The teacher's role has to be enhanced by the following two aspects:

- The teacher has to be **really persuaded about the new teaching method**. In case of doubting the success, the student's behavior will probably change immediately and become inactive again.

[5] The importance, that the own learning is supported by the teacher and the peers, is also highlighted by [Pra01].

- Show **veritable management skills**: Setting clearly the goal, guide the team to become clear about the expectations they have to fulfil and believe in the team. In some situations it can become very difficult to still believe that the contest will be a success, because the willing-ness to participate are changing from week to week. The week before the contest seemed to be the most critical. In this week unexpected issues and conflicts were popping up, because everything was becoming more concrete.

At the beginning it is important to allow the denial of all new methods. The following figures show the comments (in German) of one group about what the group wanted to do and what they didn't want. A majority of voices were raised against the very unusual requirement to sing in a choir and one comment says: "no new learning methods".

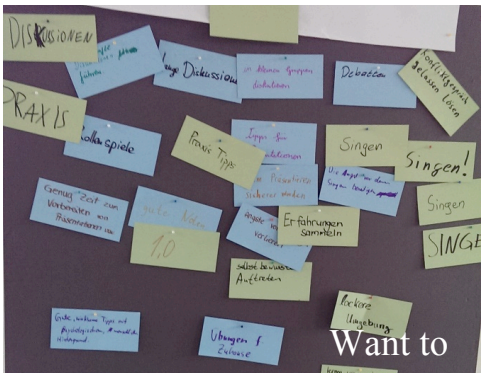

Figure 1: Comments of group A about what they want to be happen in the seminar.

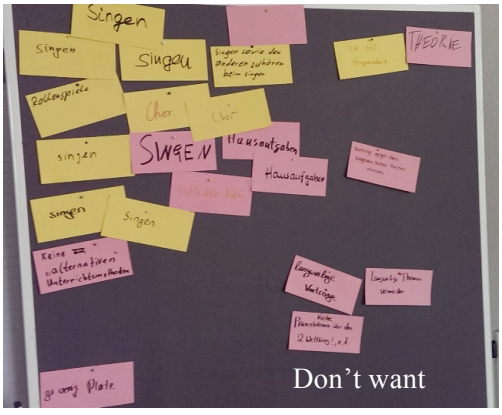

Figure 2: Comments of group A about what they do not want to be happen in the seminar.

During the seminar it was very difficult to judge the probability that the contest at the end will be attended by the students. Some few students seemed to like the idea of a contest, others were still against it and others seemed to wait and see what will happen.

After the contest took place, I asked the students to show their curves of willingness to attend the contest over the time of the seminar.

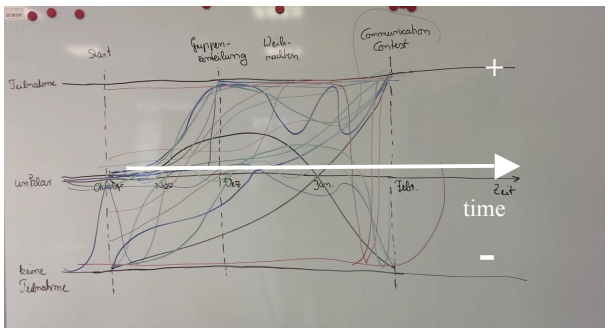

Figure 3: Willingness graphs of the winning group, Group C, 2015

The figure above shows all ups and downs of the willingness of every single team member in attending the contest. The upper bound (+) means that the student wanted to attend the contest, the lower bound (-) stands for not attending the contest. The major events as the start of the course, the sub team's set up, Christmas and the contest were given on the time line. In the lesson after the contest every student was asked to draw a line showing his/her willingness level and its development to attend the contest over the time. Below you see the willingness graphs of the winning team (group C). Only one student drew the line right from the beginning until the end on the Attending bound. All other curves show, that a lot of students underwent massive ups and downs. The teacher must be aware about these ups and downs and he / she has to tackle with it always trusting and supporting the group.

3. Results

Using new educational methods is no end in itself. The learning effects for the students have to be considered at the end. In this course students learned a lot by preparing the contest and showed really great contest acts. For such who had no active part in the contest, the contest has been a good repetition showing how excellent students are performing in different categories.

Most of the students enjoyed the course and learned a lot – not only in the core subject presentation and communication, but also in group dynamics.

4. Conclusion

For software engineering lectures the results of this nontechnical course mean that giving students examples where and how they can make use of the applied techniques is very motivating for them. Furthermore gamification creates a high dynamic environment for learning and at the end the best possible performance in the core subject (whatever the subject is) is shown by the students.

Using activating educational methods works very well, but requires additional soft skills of the teacher. University teachers with low managing experience might make difficulties in guiding the team through the forming and storming phases. In particular for software engineering or quality courses, this shouldn't be any insurmountable barrier. A good solution could be to combine the technic know-how of the university teacher with managing experiences of a practitioner used to guide software teams. Another possibility could be to provide coaching by gamification experts at least for the first time gamification is used.

In other courses, like test-oriented software development, I gained first good experiences with activating educational methods, too. After the success in the nontechnical course I'm convinced that these techniques can also be used in technical courses. PBL and gamification, however, can't be used in all courses in parallel. So the faculty or the university department has to decide how courses can make use of these new educational methods.

5. References

[Bol12] Boller, S. (2012). *100 Great Game Based Learning and Gamification Resources*. (B.-L. P. Inc., Herausgeber) Accessed at 22. March 2015 from

http://www.theknowledgeguru.com/100-great-game-based-learning-and-gamification-resources/

[Fle07] Fleischmann, P., & Geupel, H. L. (2007). Lernteamcoaching. In *Neues Handbuch Hochschullehre* (S. C2.5). Berlin: Raabe.

[Pra01] Pratt, D., & al., e. (2001). Development and Use of The Teaching Perspectives Inventory (TPI). *AERA Annual Meeting 2001.* Seattle, USA: AERA.

[Rac05] Race, P. (2005). *Making Learning Happen.* London: Sage.

[Rac07] Race, P., & Pickford, R. (2007). *Making Teaching Work.* London: Sage.