Educational Inclusiveness through Ludic Engagement and Digital Creativity

Rachel McCrindle

School of Systems Engineering, University of Reading, Reading, UK
r.j.mccrindle@reading.ac.uk

Abstract. This paper describes an approach to teaching and learning that combines elements of ludic engagement, gamification and digital creativity in order to make the learning of a serious subject a fun, interactive and inclusive experience for students regardless of their gender, age, culture, experience or any disabilities that they may have. This approach has been successfully used to teach software engineering to first year students but could in principle be transferred to any subject or discipline.

Keywords: Ludic engagement, games, digital creativity, group work, soft skills, subject knowledge.

1 Introduction

Ludic engagement may be considered as a way of blurring the distinctions between work and play [1] such that participants are motivated by curiosity, exploration, and reflection rather than the externally-defined tasks [2]. Although frequently associated with computer interfaces ludic engagement can also be more generally focused on the development of artistic prototypes and projects that bring together the disciplines of art, design, entertainment and product development, creating products that combine artistic expression with science and technology principles [3]. Additionally, by taking concepts from computer games, art, interactive media, social networks and modding cultures, ludic interfaces and approaches can create environments which are sensitive to cultural, gender-related, age-related and ethnic specificities [4] and which can be made more inclusive for people with special learning needs or physical disabilities [5].

Within an educational context, ludic interfaces and engagement can be used to blur the traditionally separate academic theories of learning and theories of play into creative, immersive and engaging environments for learning [6]. In the extreme, Quest to Learn, a public school in New York City is offering a ludic model for student engagement whereby their approach to learning mimics the action and design principles of games in the classroom. In this model students encounter a series of increasingly complex, narrative challenges, games or quests, as they move through the curriculum such that learning, knowledge sharing, feedback, reflection and next steps emerge as a natural function of play [7].

C. Stephanidis and M. Antona (Eds.): UAHCI/HCII 2013, Part III, LNCS 8011, pp. 195–202, 2013.

In this paper we describe how concepts of gamification and ludic engagement [8] have been combined with more traditional teaching practices of experiential learning [9] and activity led learning [10] and application to real world considerations, in order to engage students in the learning process of a subject discipline. In doing so, this approach to learning shares concepts with the problem and project based-learning model adopted by Aalborg University [11].

2 Approach to Learning

Game design and digital creativity have been used for the past eight years as an innovative way of teaching software engineering to a large class (100-200 students) of first year undergraduate students taking degrees ranging from Management and IT to Computer Science to Electronics and Cybernetics. As well as differences between degrees studied, variability due to gender, culture, age, internationalization, physical and social disabilities, is evident amongst each cohort, alongside many preferred learning styles. Added to this, students frequently find it difficult to comprehend and appreciate this subject at the point where it is taught, that is, prior to utilising it in the real world either on placement or in graduate employment. In order to increase engagement and understanding of the subject matter, transgress different learning styles and encourage team based working, gamification and ludic approaches are superimposed upon a structured engineering development process.

In this 20 week long assignment students work in teams to design and develop a ‘physical’ board game that will teach novice software engineers or 1st year students 'software engineering'.

In order to realize deeper learning and understanding of the technical and managerial processes involved in the development, delivery and support of software-based systems, students go through an engineering process themselves in order to develop board games that incorporate the principles of software engineering such that if someone else plays their game, they too learn about software engineering – thereby reinforcing learning on several levels. Deliberately, they do not program the game because programming skills are taught in a different module, they should concentrate on the whole lifecycle rather than just the programming aspect, and the level of programming required to produce a comprehensive game is far too complex for a first year module assignment. They do however use digital media tools to help them create their artwork and presentations and to encourage creativity.

The intention of the assignment is to embed the activities of the software engineering lifecycle into the students learning process and encourage deeper learning of the material presented in the lectures. Working in teams, students consider how to develop the rule-based system that drives the nature of their game, create worlds in which they actively participate, use strategic thinking to make choices, seek content knowledge, enhance understanding of complex concepts, and consider how to transfer their knowledge across to others.

In addition to learning software engineering concepts (e.g. software lifecycle stages, management of projects, risks and contingency, resources, application to industry,

consequences of failure), students develop their creative skills (e.g. through team branding, novel ideas and application of concepts, 2-D and 3-D design skills, graphic design, presentation of information and software/tools to delivery creativity). The students also develop 'softer' skills so valued by today's employers such as the ability to work as a team, time management skills, communication skills, design and development skills, problem solving, leadership, problem resolution, responsibility and consequence of actions.

One key intention of the work is that it is 'fun' and achieved via an interactive process rather than a more conventional type of assignment with a key aim being that the learning process of the student becomes embedded in the 'play' aspects of the game. Although some structure is imposed on the process (a series of 11 staged deliverables are created over the 20 week period) the students are encouraged to unleash their creativity and develop their own style and genre of games as evidenced by titles such as the Llamageddon, Killer Programmers and the Quest for the Governmental Contract, the IT Crown, Software Jungle and Alex's Adventures in Software Engineering, to name just five of the 170+ games developed over the past eight years. Some of these games are illustrated in Figure 1.

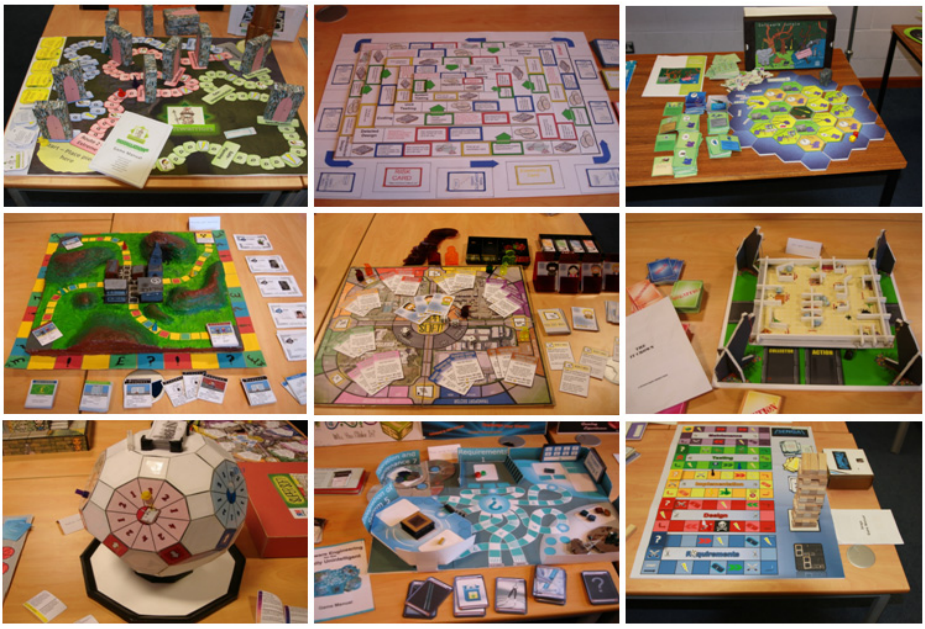

Fig. 1. Examples of games developed to teach software engineering

Although the students are given identical initial specifications, the work that they produce encompasses many different designs and approaches to game play – just as in the 'real' world where different software engineering teams would produce distinct architectural designs, use different algorithms and produce different graphical interfaces to their designs. Further ludic engagement is evident through the posters and

presentations produced by the students to promote their games, with almost all of the work being of a high standard and some of it exceptional. Examples of several posters are shown in Figure 2.

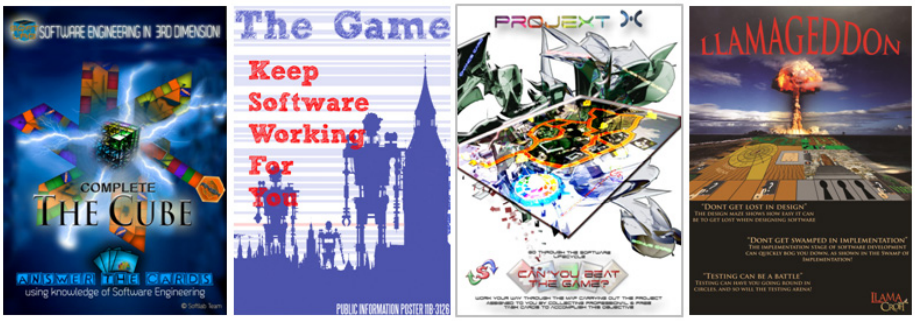

Fig. 2. Posters advertising four different games

3 Making Learning Inclusive

One of the key aims for using games and ludic engagement as a focus for teaching and learning is to encourage students to work together to complete a significant piece of work. Whilst one key goal of this approach is for the students to develop a deeper knowledge and understanding of the subject being taught, an equally important goal is for the students to develop their softer skills, and in particular those related to communication and team work. A cohort of students typically has a mix of degree disciplines, gender, culture, age, special education needs and physical disabilities. Whilst some students naturally find group work and communication easy and intuitive, others find it a highly challenging and uncomfortable process.

In order to ensure that the learning experience for this module is as inclusive as possible we have incorporated a number of ways to encourage and facilitate team formation, bonding and working, that involves preparation prior to the module, monitoring during the module, and reflection after the module. These include:

- Liaising with the disability officer for our School so that we are aware of any students with particular learning needs or disabilities.
- Ensuring that we have rooms booked that facilitate group work rather than traditional classroom style activity.
- Gradually introducing students to group work through a developing series of team tasks in the first few weeks of the module and prior to the start of the main assignment – enhanced by sweets and chocolate!
- Observing the behavior of the trial teams to look for poorly functioning teams, students who appear uncomfortable in their teams, and teams lacking any form of self-direction.

- Having students assess their skills and background experience against the range of activities in the project specification and promoting an ethos that everyone is good at something and has a team role to play.
- Including in the 11 deliverables a wide range of activities and learning outcomes encompassing ideas creation, planning and management, report writing, development activities and creative presentations such that everyone is likely to 'shine' in one or more activities.
- Administering a light touch Belbin test [12, 13] to the students backed-up by a discussion session to make them consider the type of team player they are, as well as their likely strengths and weaknesses when working in a team.
- Logging attendance at all group sessions and following up absenteeism before teams become too fragmented.
- Appointing a project manager for the team and encouraging self-management of the team, but working closely with a team and its individuals if relationships start to break down.
- Including lectures on group dynamics and having industry promote the benefits of working in diverse and global teams, thereby encouraging students to embrace and celebrate diversity rather than trying to avoid it.
- Allowing the students to choose their team name, game genre, presentation style etc. in order to facilitate the blending of the serious side of the assignment with the game dynamics and thereby blurring the distinctions between work, play and learning.

4 Engagement in the Learning Process

Some excellent and highly creative work has been produced and of the 1200+ students that have participated in this activity only one student has been unable to be placed in a team. At the end of the 20 week process students have to critically review their approach to the assignment, assess what they have personally learnt from working in a team and suggest improvements for how it might run even more effectively in subsequent years. Several comments from students are given below:

"...many lessons were learnt about working in a team that would not have been acquired if I just had to write an essay on 'group work' or 'management', this experience will certainly be helpful in the future, as in this field, team work is essential."

"...another enjoyable aspect of the project was overcoming the challenge we had set ourselves by coming up with a complicated design. Once it was finished I felt a great sense of accomplishment because we had met our own specifications"

"...at first I was skeptical about undertaking a non-computer related project for software engineering, and about the same for the usefulness of working in a group. I can proudly say that the idea of getting us used to group work by this method is a good one, the idea that we are to produce a non technically involved project made it easier for us to establish how to work in a team before also having to learn more on

top of that. I found that working in a team was less stressful than having to go through a project on my own and it was better in terms of productivity and efficiency and shared expertise." (Aspergers Student).

Fig. 3. Awards Day

In addition to the more formal assessment of the work and in order to celebrate the successes of the students as well as adding another fun, yet competitive, edge to the work, the Annual Software Engineering Brilliance Awards or SEBAs are organized whereby the students present their work to a panel of judges from industry and academia, competing for prizes in categories related to their game design and pedagogy, posters, presentations and documentation in order to win monetary or technology related prizes as well as the coveted SEBA. See Figures 3, 4 and 5.

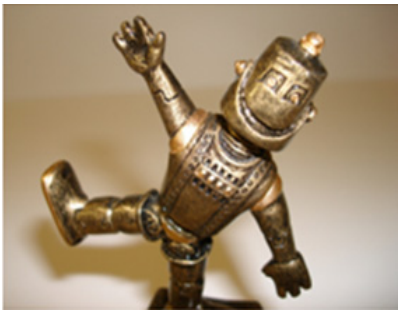

Fig. 4. and **Fig. 5.** Anticipation prior to the SEBA announcements and a SEBA

5 Summary

This approach to teaching software engineering can be measured by various parameters ranging from the visual and pedagogic quality of the work produced; overall performance of students taking the module; positive comments by the students about how fun and engaging the learning process of a subject like software engineering can be; the amount of use of the processes, knowledge and soft skills learnt in subsequent modules, on placement, and upon graduation; winning the UK's Higher Education Academy Engineering Subject Centre's Teaching Award; and by the activity being recognized as an example of best teaching practice by our professional accrediting institutions and the global corporations with which we interact.

This is not to say however that the process is an easy one, there are challenges for both staff and students and the activity requires a significant time investment for it to run smoothly as well some resource for games production. The activity has been running for eight years now with 1200+ students taking part. Although the overall premise of the assignment has remained consistent, tweaks have been made to the process each year based on staff experience and student comments.

Additionally, although these games are focused on teaching software engineering, the ways in which subject knowledge can be embedded into the games, the development processes undertaken, the digital creativity activities, soft skills elements and the overall ludic nature of the games based activity are relevant to, and easily transferrable to, all disciplines whether science or humanities based.

Acknowledgements. Thanks are due to all the students who have participated in this activity and the demonstrators/staff who have helped me run it over the past 8 years.

References

1. Lindley, C.A.: Ludic Engagement and Immersion as a Generic Paradigm for Human-Computer Interaction Design. In: Rauterberg, M. (ed.) ICEC 2004. LNCS, vol. 3166, pp. 3–13. Springer, Heidelberg (2004)
2. Gaver, W., Bowers, J., Boucher, A., Gellerson, H., Pennington, S., Schmidt, A., Steed, A., Villars, N., Walker, B.: The Drift Table: Designing for Ludic Engagement. In: CHI 21004, Design Expo. ACM Press, New York (2004)
3. Sommerer, C., King, D., Mignonneau, L.: Interface Cultures: Artistic Aspects of Interaction. Transcript Verlag (2008)
4. Fuchs, M.: Ludic Interfaces. Driver and Product of Gamification. GAME Italian Journal of Game Studies (2012), http://www.gamejournal.it/ludic-interfaces-driver-and-product-of-gamification/
5. Peterson Brooks, E., Brooks, A.L.: Ludic Engagement Designs for All (LEDA): Non-formal Learning and Rehabilitation. In: Designs for Learning, 3rd International Conference Exploring Learning Environments, pp. 78–80 (2012)
6. Selander, S.: Designs for Learning and Ludic Engagement. Digital Creativity 19(3), 145–152 (2008)

7. Quest to Learn, Institute of Play (2012), http://www.instituteofplay.org/work/projects/quest-schools/quest-to-learn/
8. Deterding, S., Dixon, D., Khaled, R., Nacke, L.: From Game Design Elements to Gamefulness: Defining "Gamification". In: MindTrex 2011. Tampere, Finland (2011)
9. Kolb, A.Y., Kolb, D.A.: The Kolb Learning Style Inventory - Version 3.1 Technical Specifications, Experience Based Learning Systems (2005), http://www.whitewater-rescue.com/support/pagepics/lsitechmanual.pdf
10. Wilson-Medhurst, S., Dunn, I., White, P., Farmer, R., Lawson, D.: Developing Activity Led Learning in the Faculty of Engineering and Computing at Coventry University through a Continuous Improvement Change Process. In: Research Symposium on Problem Based Learning in Engineering and Science Education, Aalborg, Denmark (2008)
11. Aalborg University, Principles of Problem and Project Based Learning, the Aalborg PBL Model (2010), http://www.pbl.aau.dk/digitalAssets/33/33124_pbl_aalborg_modellen.pdf
12. Belbin, Belbin Team Role Summary Descriptions, http://www.belbin.com/content/page/5002/BELBIN(uk)-2012-TeamRoleSummaryDescriptions.pdf
13. Belbin, Team Roles in a Nutshell, http://www.belbin.com/content/page/5664/Belbin(uk)-2011-TeamRolesInANutshell.pdf