Using a Serious Game Development Approach in the Learning Experience of System Engineering Design

Marco Blokhuis, Nick Szirbik

Using a Serious Game Development Approach in the Learning Experience of System Engineering Design

Marco Blokhuis[1] and Nick Szirbik[2]
[1]University of Groningen, Faculty of Science and Engineering
[2]University of Groningen, Faculty of Economics and Business
M.P.Blokhuis@rug.nl and N.B.Szirbik@rug.nl

Abstract. This is a position paper, presenting also some preliminary results of an experiment in teaching and learning systems engineering design. Students were asked to develop games in conjunction with a system design project. We have measured the effects of the game development on the learned skills, via feedback questionnaires. The preliminary results indicate that students find the game development difficult but also like this aspect the most from the whole coursework. In terms of skills, the ability to adopt a holistic view is considered the most valuable by the students.

Keywords: Serious games, Functional Architectures, Learning experiment.

1 Introduction

At the University of Groningen, graduates, alumni, industry partners, and accreditation organizations of the study Industrial Engineering and Management in the Industrial Engineering and Management master (IEM), acknowledge positively the experience acquired via integrative courses present in the curriculum. The System Engineering course is one of them. This course applies both problem-based and project-centric learning. Within this course students are working in a close-knit group to develop a complex system design. The group, named a triad, and it is formed by three teams of three students and a student who acts as coordinator and integrator. Each triad models a different system, these being complex socio-technical systems, like for example a nation-wide bike theft prevention system, a large scale electric scooter sharing system, locally connected wind farms and consumers, an internationally integrated hydro-power storage system, and other large-scale multi-stakeholder systems. During the last four years (2014-2017) emphasis has been put on the use of serious games in the course, as both a learning experience enabler and knowledge acquisition enhancer. The novel approach presented in this paper encompasses not only the development of the triad's system but also the playing of a serious game that mimics the growth of the system.

The advantages of playing sessions of a specific serious game (GasBoard, developed by university researchers together with industry partners) to support the motivation and learning experience of students during the Systems Engineering course work have been reported previously (Szirbik et al., 2015). To enhance further the students' experience in their system design, the development of a serious game that is specifically related to their triad's system design project was experimented with. The basic feature of this serious game is that the game has to reflect to some extent the system to be designed. In this paper, we are describing an experiment we run through the System Engineering course this year, to test if the development of a serious board game connected to the system to be designed can support positively the learning of systems design. A secondary goal was to find out how the game development can influence and improve the system's architecture.

The rest of the paper is organized as follows. First the context in which the experiment is taking place is described. Second the research question is operationalized. Third the experimental setting is described. As the data related to the experiment has been just gathered and it is under analysis and interpretation, only some preliminary results are presented. Finally, conclusions are then drawn.

2 The system engineering design course as context for research and the main question of this research

The system engineering design course is one of the oldest integrative course in the IEM master program at the University of Groningen. It started in 2004. The project that the students have to undertake is quite equivalent (albeit much shorter in time and smaller in scope) to the pre-inception phase in a real system development process (Buede, 2009). The main output of the coursework is a design of a complex socio-technical system. This consists of a context-placed operational architecture of the system, comprising a functional architecture and a (mostly) generic physical architecture. There are two deliverables for each triad. The first is the Originating Requirements Document (ORD), which is a text augmented with functional and physical diagrams, and a System Requirements Document (SRD), which is a digital database for system design (developed in CORE, a computer aided systems engineering tool, by Vitech Corp.). These two deliverables have to be consistent with each other; the ORD is intended for system's stakeholders use and the SRD for the developers use. The system modelling in both deliverables is relying mostly on IDEF0 (CORE supports this graphical language) functional models – using hierarchical and interaction diagrams to depict the functional architecture. For the quality and detail of this functional architecture the students get more than half of their grade.

Each triad of students has to develop a system design of a complex socio-technical system. To prevent mimicking, no two triads have the same system to design, and from one year to the other, the systems are not repeated, to prevent inter-generational plagiarism. The choice for the socio-technical system to be designed is let to the members of the triad (i.e. they are playing the role of stakeholder and designer in the same time). To allow concurrent design, each team in a triad is responsible to develop a part of the architecture – roughly a third. These parts are delimited by the students themselves. In order to insure coherence between parts and organize the cooperation and the communication between the teams, a separate, tenth student in the triad is playing the role of system integrator. The coursework takes 10 weeks to complete.

During this period, the individual teams and the triads have to deliver 3 intermediary ORDs and SRDs on which they receive feedback from tutors. The timing of these deliverables is illustrated in figure 1.

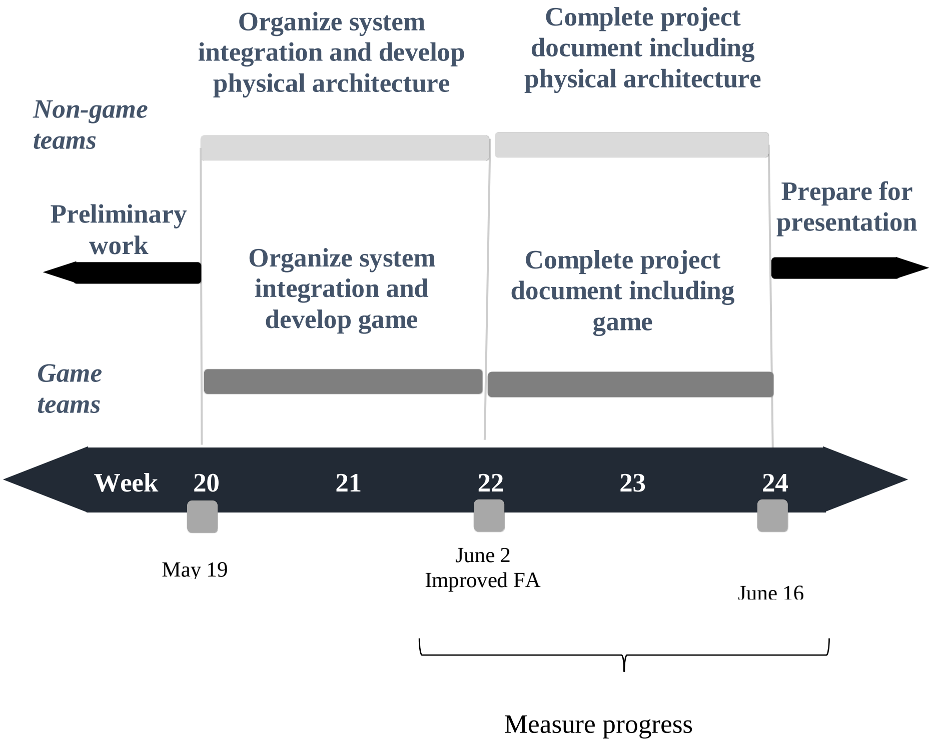

Fig. 1**.** The timeline and the three deliverables related to the experiment

In 2015, serious game playing sessions in GasBoard have been introduced into the course to address observed motivational and communication problems at team and individual levels. It successfully improved the cognitive engagement between the teams and individual students who were doing the designs. It also enhanced the communication and alignment in the integrative process, heightening the motivation to finish with an exciting result. Moreover, it challenged to explore and find new design ideas (Szirbik et al., 2015). Despite the observed improvements, the students who played this digital game reported that the game was “too restrictive and impossible to adapt to their own ideas” (Velthuizen, 2016). They suggested a non-digital version in the form of a board game and argued that “it would have been more interesting and useful to have a flexible board game that was easy to change according to the changes they made in their own design” (Velthuizen, 2016). Letting students to develop their own non-digital board game during their system design development has both material and educational advantages. Considering the material advantage, it saves the resources needed for the development and implementation of a digital game. In fact, it is possible to begin with a clean slate, developing the board game from scratch. The educational advantage is that it offers the flexibility required to try new ideas for the game, as it is easy to implement, change and adapt during a project development. These are features that are particularly important when exploring solutions for the blockage encountered in the preliminary development of socio-technical system of 'chicken-and-egg' development conundrum type (Szirbik et al., 2014; Ittoo et al., 2013a; Ittoo et al., 2013b). This type of system is characteristically marred by the investment blockage phenomenon. Typically, there are stakeholders who are interested to develop a technology-heavy infrastructure for specific clients, but the development funding would be available only if the usage of the infrastructure would enable the users to pay for it (Wene, 1996). This type of systems has been often encountered in multiple past and recent infrastructural projects such as renewable energy networks (Veeningen, 2016) and liquefied natural gas infrastructure (Thunnissen, Van de Bunt and Vis, 2016) and are often chosen by students. The triads that have developed these kind of systems were the ones encouraged to undertake the supplementary task to develop a board game mimicking the growth of the system to be designed.

We expected that developing a board game and playing it can have multiple effects and yield various research results. First, the learning experience of the students is expected to be better than without the game development – we name this the *learning* effect. Second, the generic framework to develop games can be continuously validated, and if necessary, refined – we name this the *knowledge* effect. Third, the final system design in the project can be affected by the game development activities that take place in the same project – we name this the *design* effect. Though we pursue all three lines of investigation in our research, we focus here in this paper only on the learning effect. Therefore, this paper specifically describes the experimental setting that has been developed to answer the question: *How does the game development by students during the coursework support/influence/hinder their learning of systems engineering design.*

3 The Experimental Setting

In order to answer the research question, the study is focusing on the experience of the students who were part of teams designing systems having a chicken-and-egg characteristics. Seventeen out of the 27 teams of the class of 2017 had to design such a system. Six out of these 17 teams have agreed before the first deliverable date to develop a board game – and three of these games were specifically linked to an entire triad's system. Six other teams (actually two whole triads) out of the initial set of 17 attempted later to develop a game to support their final presentation. To keep the workload fair with the other teams, the initial six game teams did not have to finish the physical architecture part. However, this is not really a change, because the game itself contains in the end most of the physical elements of the system. The playing off the game is intended to mimic the growth of the system and it is supposed to help the stakeholders with the discovery of bootstrapping scenarios for the system that is designed by each team. These six teams were provided with a generic framework (architecture and guidelines) for developing chicken-and-egg problem serious games and a manual of how to apply it (Veeningen, 2016). Both the game teams and the non-game teams were pro-actively supported by the instructors during tutorials and received feedback after each deliverable.

We investigated the merits and value of using a game development complementary to the system design that the students have to do for their team project. We focused on the learning effect, that is, the development of the game and its continuous refining and testing will affect in turn the depth of understanding of the students about how systems are designed. In the end, our hypothesis is that the skills and knowledge necessary to design a complex multi-stakeholder system design can be improved by complementing the design process with a serious game development that explores bootstrapping scenarios for its gradual development.

The experiment takes into account the learning results of all the 17 teams that chose a system design that have the "chicken and egg" characteristic. The hypothesis of this research is that these game development and play activities have some positive effects on how the students grasp the basics of systems engineering design. Data is gathered via the two last team's deliverables to assess the design effect of game on the output, and via general course feedback questionnaires that were handed out to all students of System Engineering to evaluate the effect of developing a game on the . The questionnaires inquire about the skills the students think they have learned, what they liked, what they found difficult, their proposal for course content change, participation in post-mortem workshops, continuing to be involved in System Engineering and any other comments about the course. The questionnaire did not contain any specific question about the experience with the game. For the purpose of this paper, we only looked at the skills the students reported to have learnt during the course, and the most liked and disliked aspects in the coursework. For each of the section, they were allowed to give multiple answers. In the next section, the answers provided by the students involved in teams designing chicken-and-egg type of system is presented and analyzed.

4 Some Preliminary results

The original goal was to assess the effect of the game development on the quality/completeness of the final functional architecture. The analysis of difference between the improved Functional Architecture and the final one (the difference between the next to the last and the last deliverable) had proved to be inconclusive, as the student teams did not found the time to include any modification.

Out of the 52 students working on a chicken-and-egg type of system, 30 completed the course feedback questionnaires, which contained a number of 11 open questions – 3 of which are reported here. Fourteen of students were those out of 18 of the students who were part of teams committed to develop games, and 16 of students were those out of

34 the students who did choose not to develop a game. From these completed questionnaires, the information presented in tables 1, 2, and 3 was compiled. Table 1 presents the answers collected that relate to the question about the skills the students openly reported to have learnt (without any suggestion for the response). Table 2 is about the aspects of the course that the students liked the most, and table 3 about the one they liked the least (again, without any suggestion for the response in the questionnaire).

Table 1: Most important learned skills for different types of groups

Skill	*Original game*	*Not originally game*
Group work	4	13
Functional thinking	5	11
Holistic view	6	0
Abstract thinking	3	1
Creativity	1	2
Switching from FA to PA	0	2

The skills reported by the students in the course feedback questionnaires are: group work, functional thinking, holistic view, abstract thinking, creativity, switching from functional architecture (FA) to physical architecture (PA) (cf: Table 1). The skills the most often mentioned are group work (17), functional thinking (16) and learning to acquire a holistic view (7). The acquisition of the two first skills, group working and functional thinking, are in large majority acknowledged by the members of teams that did not develop a game. However, holistic views skills are mentioned **only** by the students that were member of teams that originally decided to develop a game.

Table 2: Most liked aspect about the course

Most liked aspect	*Original game*	*Not originally game*
Game development	7	1
Group work	0	6
Related to course form	3	0
Creativity	2	2
Functional thinking	0	2
Creation of a realistic, tangible system	1	0
Holistic approach	1	1
Developing system	1	1
Freedom for own project	0	1
Playing the game	1	0

The most liked aspect about the course are the aspects related to the game development (8), the group work (6), aspects related to course form (3), creativity (4), functional thinking (2), creation of a realistic, tangible system (1), holistic approach (2), developing the system (2), the freedom for own project (1) and playing the game (1) (cf: Table 2). The game development was mentioned eight times by eight different students. One of these eight students was a student of a late game development triad. Seven students mentioned the group work as their favorite aspect of the course; all of them were in teams that did not originally choose to develop a game. If we consider the students who chose originally to develop a game, the development of the game is chosen most often as most liked aspect, followed by aspects related to the course form. It is interesting to notice that none of them mention group work as most liked aspect of the course. For the students not originally choosing to develop a game, group work is the most popular aspect, followed by creativity and functional thinking.

Table 3: Most difficult aspects about the course

Most difficult aspect	*Original game*	*Not originally game*
Functional thinking	4	3
Game development	6	0
High workload	2	2
Working on large system	1	2
Group work	1	3
Functional architecture	2	1
Defining the initial system	0	1
Managing deliverables	0	1
Gathering real-world data to make system feasible and real	1	0

The most difficult aspects about the course as reported by the students are functional thinking (7), game development (6), high workload (4), working on a large system (3), group work (4), functional architecture development (3), defining the initial system (1), managing the deliverables (1) and gathering real-world data to make the system feasible and real (1) (cf: Table 3). Depending if the students were making a game or not, the aspects found the most difficult differed from answer to answer. For the students making a game, the development of the game came first followed by functional thinking. For the students not developing a game, functional thinking came first as the most difficult aspect, with group work coming second.

5 Conclusions

The students from the groups initially choosing to develop a game reported most often the ability to adopt a holistic view as most valuable skill, followed by functional thinking. In addition, **even if developing the game has been considered as the most difficult aspect they also liked this aspect the most**. Group work is only mentioned marginally by these students in each of the three questions. The students who did not choose initially to develop a game found the group working skills the most valuable skills learned during the course. It is also the most liked aspect and also considered the most difficult aspect of the course. Functional thinking was also considered to be one of the most difficult aspects of the course. From these observations we can conclude that developing a game indicates a perceived increase of the ability to adopt a holistic view as the most valuable, and it is noticeable that this skill is not mentioned at all by the students not initially choosing to develop a game.

These results pertain only to a quantitative interpretation of a small part of the data collected during the experiment. However, the games were filmed, hundreds of notes were taken, post-mortem focus-groups were organized, and the game themselves can be analyzed and compared. We still have to sift through this trove of qualitative data and find the right interpretations. For example, one triad remarked: *"with the creation of the game, we had to work completely together, everybody's ideas were discussed and considered, we agreed, we disagreed, we argued, and improved together our game. More collaboration than ever during the course came up with the game"*. Many more interesting remarks of this kind wait to be properly analyzed and interpreted.

References

Buede, D. M.: The engineering design of systems: models and methods. 2nd edn. Wiley, Hoboken, New Jersey (2009).

Ittoo, A., Szirbik, N. B., Huitema, G., Wortmann, H.: Simulation Gaming and Natural Language Processing for Modelling Stakeholder Behavior in Energy Investments. In: ICS'2013 Proceedings, the 11th Industrial Simulation Conference, pp. 23-27. EUROSIS, Ghent, Belgium (2013a).

Ittoo A., Szirbik N. B., Huitema G., Wortmann H.: Serious gaming augmented with natural language processing for learning on energy infrastructure investment decisions. In: 43rd International Simulation and Gaming Association Annual Conference, Cluj-Napoca, (2012).

Szirbik, N. B., Huitema, G. B., van der Burg, R. H., Wortmann, J. C.: Interactive simulation for discovering investment scenarios in energy systems. Proceedings of the Industrial Simulation Conference, ISC 2014, pp. 53–56, EUROSIS (2014).

Szirbik, N. B., Pelletier, C., Velthuizen, V.: Enhancing an Integrative Course in Industrial Engineering and Management via Realistic Socio-technical Problems and Serious Game Development. In: Umeda, S., Nakano, M., Mizuyama, H., Hibino, N., Kiritsis, D., von Cieminski, G. (eds) Advances in Production Management Systems: Innovative Production Management Towards Sustainable Growth, pp. 541-548. APMS 2015. IFIP Advances in Information and Communication Technology, vol 459. Springer, Cham (2015).

Thunnissen, S. K., Van de Bunt, L.G. and Vis, I.F.A.: Sustainable Fuels for the Transport and Maritime Sector: A Blueprint of the LNG Distribution Network. Logistics and Supply Chain Innovation, pp. 85-103. Springer International Publishing (2016).

Veeningen, J. W.: Design science research issues in developing and applying serious games in the uncertain world of long term energy investments, MSc Thesis, University of Groningen (2016).

Velthuizen, V.: Introducing FASD: A Functional Architecture for Serious game development, BSc Thesis, University of Groningen (2016).

Wene, C. O.: Energy-economy analysis: linking the macroeconomic and systems engineering approaches. Energy, 21(9), pp. 809-824 (1996).