Evaluating the Effectiveness of Board Game Usage to Teach GSE Dynamics

Rini van Solingen, Kevin Dullemond, Ben van Gameren
Delft University of Technology
{d.m.vansolingen, k.dullemond, b.j.a.vangameren}@tudelft.nl

Abstract

When preparing students for an industrial or academic career in software engineering, educational institutions need to address the globally distributed character of the software engineering profession. As it will be difficult to have students actually participating in a real-life, multi-site, globally dispersed, industrial project and thus acquiring knowledge from experience, there is a challenge to properly prepare students for their future jobs.

One way to teach the dynamics of global development is by letting students participate in a game in which these dynamics are at the core. In this paper a board game is presented that can be used for such a purpose. The paper presents a description of the game and the findings from a controlled experiment to test its learning effectiveness.

1 Introduction

It is getting increasingly common for software teams to no longer conduct their work from a single office building. Developing software products and services is simply no longer an activity that takes place on one geographic location by closely located and informally communicating software engineers. This happens both due to the globalization of business [1] and because people are starting to work from home more and more [2]. This also impacts software engineering education.

Education should prepare students so that they are ready to cope with large-scale software development [3]. Curricula for software engineering should, therefore, be able to teach the impacts of globally dispersed software development. After all, these students will, in some point in time, join the software engineering workforce and as such they need to be prepared to work in globally distributed settings. Industry is often not satisfied with the level of real-world preparedness of university graduates [3]. They prefer to hire candidates that already posses the skills and knowledge to succeed [5]. As a consequence many companies seem to need to teach graduate students the dynamics of global development themselves.

Software professionals must be able to balance and make trade-off decisions in dynamic environments, something which is quite challenging to teach in a class room setting. One way to teach the dynamics of global development is by having students participate in games that incorporate such real-life dynamics. In games students are able to experience some dynamics of global development without having to carry out actual globally distributed software development. The contribution of this paper is to present: a. the design of a board game that enables the teaching of globally distributed software engineering dynamics and b. the findings of a controlled experiment to validate learning effectiveness of playing this game.

The paper is structured as follows. Firstly, we present background material on the dynamics of global software engineering in section 2 and then investigate the usage of games for teaching these GSE dynamics in section 3. Following this, we explain the game and its design in section 4 and show an example of a playing round to show its dynamics in section 5. Subsequently we describe a controlled experiment to test the games learning effectiveness in section 6 and discuss the results from this experiment in section 7. Finally, we discuss our study and draw conclusions in section 8.

2. The dynamics of GSE

Global Software Engineering is dynamic because many variables are involved in selecting the most appropriate course of action. On the one hand the relative importance of potential benefits is involved. If a company has one large or several small customers in a specific region, customer proximity [7][8] could be a motivation to expand its business to that location. Next to this, availability of a sufficiently large [1], skilled [1] or favourably priced workforce [8][9] are also worth considering when deciding whether and how to exercise a GSE strategy. Finally, business centric reasons like reduction in time to market [1][10], global presence [1] and being better suited to handle the increased organization scale [1] are also factors to consider. Next to the potential benefits, expanding a business globally also introduces a number of challenges which should be considered and acted upon when constructing a GSE strategy. These challenges arise from the existence of three distances in GSE: geographical, temporal and socio-cultural [11][12].

Because of the geographical distribution of the company the cost of meeting face-to-face increases, both in money and in time, and therefore these meetings become less common. When working geographically distributed, people have less informal interactions [13][14], experience an increased effort to initiate contact [15] and communicate less effective in general. These challenges can result in a lack of shared understanding [1][14], a lack of teamness [1] and a reduction of trust [16], all of which can cause a degradation of both the efficiency and quality of the work. The existence of the temporal and socio-cultural distances has similar effects. Because of the temporal distance the amount of overlap in working time is reduced [15][17] which can result in delay of communication [12][18] and a decrease of communication in general. Finally, a socio-cultural distance within a team can result in misunderstandings [1] and therefore a decrease of communication effectiveness. An overview of the benefits and challenges of GSE can be found in [19].

3. Teaching dynamics

Teaching standard software engineering without considering interaction with other disciplines fails to introduce students to the total environment experience in which industry products are developed [20]. As such, we advocate that educational institutions should address the globally distributed character of the software engineering profession in their teaching curricula. It is however infeasible to completely teach all dynamics that can be learned in real-life. It is for example often infeasible to let students participate in a real-life, multi-site, globally distributed, industrial software engineering project. Therefore, acquiring knowledge by experience is only partly possible. Software engineering as a profession is rather difficult to learn solely through reading software engineering literature and listening to lectures [3]. This presents a major challenge to properly prepare students for their future jobs in which globally distributed software engineering will play an important role.

Teaching the dynamics of GSE is not straightforward. It is difficult to teach students the consequences of time differences, coordination difficulties and cultural diversity. Explaining the complexities and challenges is possible; however, this will only limitedly impact their thorough understanding. Carrying out real development projects by students is often strongly limited in size and complexity. In experiencing global dynamics in student projects, it can be expected that students experience difficulties due to the inherent academic background of these projects. Because of this it is doubtful whether such experiences contribute to learning how to properly carry out software development projects. Finally, the time and scope constraints inherent in an academic setting prohibit projects to be of sufficient size to exhibit most of the phenomena present in real-world software engineering processes- those that involve large, complex systems, large teams of people, and other factors such as management, workplace issues, and corporate culture [4]. Even some accreditation institutions rule that educational institutions should teach students to function in multidisciplinary teams, communicate effectively and provide broad education in a global and social context [20]. As such, pedagogical challenges for teaching GSE include simulating the industrial environment in which students experience real-life struggles such as uncertainty, change, trade-offs, low-performing teams, cultural diversity, political conflicts and differences in skills and knowledge [20].

One way of teaching real-life dynamics beyond participating in practice is making use of games. Because many students enjoy playing games, it seems logical to combine this play aspect with instruction and learning. Because games require the active participation of students, the material has a greater chance of being integrated into the cognitive structures of the individuals and thus being retained [21]. Subject matter areas where specific content can be targeted are more likely to show beneficial effects from gaming: for areas such as math, physics and language arts, where the specific objectives can be stated, simulation and games can be used [21]. As such it makes sense to investigate and design a game that puts the dynamics of GSE at its core. When playing such a game students experience the dynamics of global development without having to carry out actual software development.

Applying games to the education of software engineers is not a new idea [22]. As an example, Baker et al. present a game called Problems and Programmers to teach project dynamics to students by means of a card game [4]. Problems and Programmers has three qualities that make it effective for teaching, being: (i) competitive, which is proven to encourage collaborative learning, (ii) physical, ensuring that underlying mechanisms become visible, and (iii) fun, which is known to be highly conductive to learning [4]. A game to teach GSE dynamics should build upon similar qualities.

4. Game design

IT Billionaire[1] is a turn based game intended for 2 through 4 players in which players attempt to become

[1] A full description of the rules of the game can be found on http://www.aspic.nl/ITBillionaire/rules.pdf

a billionaire by running a globally operating software engineering company. The game takes place on a world map and every round of play represents a day (the game board is displayed in Figure 1).

Figure 1: IT Billionare Game Board

In such a round the action starts in the east (where the sun rises) and gradually moves to the west. The players select the actions (i.e. buy lab, select work, do work) they wish to perform in a round in advance by choosing a number of cards before the round starts.

Like mentioned before, the goal of developing this game was to help teach GSE dynamics to students. To accomplish this, the game should, on the one hand, possess a number of qualities that contribute to learning effectiveness and, on the other hand, the game should reflect the GSE dynamics we wish to teach. The qualities that contribute to learning effectiveness are the following:

- The game should be competitive because it motivates students to play the game [4] and because it encourages collaborative learning, an educational technique that is known to have significant advantages [23]. IT Billionaire is a competitive game; players compete with each other to make the most profit with their fictional company. In their struggle to do so they make use of the same resources and have to develop a strategy which has a large impact on the outcome of the game.
- The game should relate to (physical) real-world entities. This physical nature further encourages collaborative learning and also helps to visualize concepts [4]. IT Billionaire relates to real world entities by (i) using a world map as the playing board, (ii) allowing the opening of physical labs on this map, (iii) using a physical representation for the projects and products being developed and finally (iv) by using a physical representation for the amount of work done on a particular project or product (an amount of rings on a container – see Figure 2).
- The game should have a fun and engaging nature because this is known to be highly conductive to learning [6][23][24][25]. In IT Billionaire unexpected situations caused by the closed selection of action cards, use of the same resources and selecting the event card after the selection of the action cards, add to this quality.

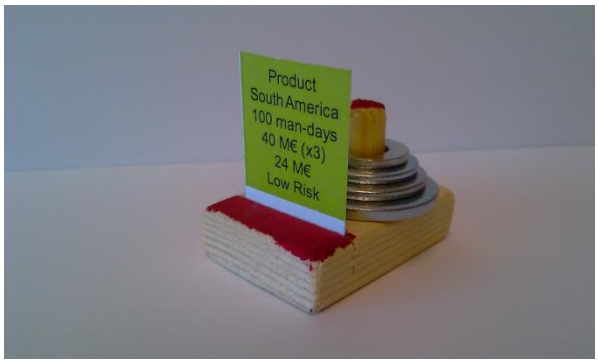

Figure 2: Container Carrying a Low Risk Product

The GSE dynamics are reflected in the game as follows:

- The players can buy labs in different parts of the world representing a specific number of man-days available each round of play. The price of the labs with a similar amount of man-days available differs between the different continents to reflect the differences in cost of labour in the different continents.
- The amount of labs and the capacity of these labs differ between the different continents to indicate the differences in availability of workforce between these continents.
- When players perform work in the same continent as the customer of the work, the work they perform on the project is done more effectively.
- When working on the same project at the same time (so in the same time zone) with two or more labs, the work is done at a suboptimal efficiency. This is to reflect overhead in the communication and collaboration between the dislocated teams.
- When work is transferred between different labs a certain number of man-days is lost due to the effort for handing over the work to the other team.
- In the game it is possible to develop following a follow-the-sun paradigm [1]. So, a player can start a day by working on a project with a lab in the east, subsequently move the work to Europe or Africa when it becomes day there and finish the day by again moving the work, this time to a western continent and carrying out work on the project in a lab there as well.

- Before the start of a round of play (but after the players have selected their actions) an event card is drawn making it impossible to carry out work in one specific continent during that day. These events are meant to reflect risks and regional instability and therefore some continents are more prone to be selected than others.

5. Game play example

To give an impression what playing IT Billionaire is actually like, we will show a scenario in which several elements and actions of the game are demonstrated. This scenario shows a single round of the game which represents a single day in the world of GSE. In this scenario the player already bought a medium sized lab in Europe for 30 million euros in a previous round (shown in Figure 3).

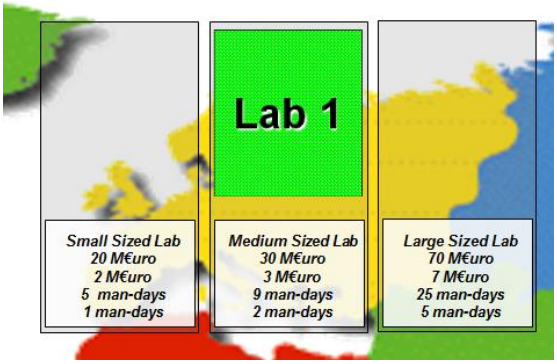

Figure 3: Medium Sized European lab

Before the start of each round every player pays the 'Upkeep' for all of his labs. So the player should pay three million euros since he only owns the European lab. Subsequently all players choose which 'Action Cards' they wish to play in the upcoming round. There are five types of action cards: acquire lab, sell lab, select work, do work and stop work. Because the player owns one lab, he can select two 'Action Cards' (1 + # labs). In this case the player selects a select work and a do work 'Action Card'. Finally, when all players have selected the 'Action Cards' they wish to play in the coming round, an 'Event Card' is drawn from the stack of 'Event Cards'. Such an 'Event Card' makes carrying out work in a specific continent impossible in the upcoming round. In this case an 'Event Card' is drawn which causes that it is not possible to carry out work in Asia this round.

Having completed the Pre-Round actions the round itself begins and the three time zones are played from East to West. In each time zone all players get the option to play zero or more of the 'Action Cards' they selected before the start of the round. At the start of the round each player plays, in a clockwise order, zero or more 'Action Cards' in time zone East. Note that in this round it is not possible to carry out work in Asia by playing a do work 'Action Card' since the drawn 'Event Card' prevents this. Because the player does not have a lab in Asia and Oceania he decides to play the select work 'Action Card' to acquire work. Now the player takes two 'Work Items' from the stack of 'Work Items' and for both of these cards the player has to decide whether he accepts them or not. The player decides to only accept the high risk project shown in Figure 4 and the select work 'Action Card' is discarded.

Figure 4: High Risk Project Card

When all players are done in time zone East we move to time zone Center. Again all players get the option to play zero or more of the 'Action Cards' they selected. The player decides to actually carry out some work and he plays the do work 'Action Card'. To do this, the player selects one of his 'Work Items' and places this on a lab he owns in the current active time zone. By doing so the lab is blocked for the current round and cannot be used again until the next round. So, in this case, the player decides to carry out the high risk project he just selected in his medium sized lab in Europe.

In order to determine how much work is actually performed we apply the following modifications to the available quantity of man-days.

1. **If the 'Work Item' is carried out in the same continent as where the customer is located:** double the available quantity of man-days
2. **If the 'Work Item' is a high risk 'Work' Item, a dice is thrown**
   a. **If a 1 is rolled:** The available quantity of man-days is set to 0
   b. **If a 2 or 3 is rolled:** The available quantity of man-days is divided by 2
   c. **If a 4,5 or 6 is rolled:** The available quantity of man-days stays the same
3. **If the 'Work Item' has been moved to this lab in the current round:** Withdraw the overhead fee of the lab from the available quantity of man-days

When after adding the available quantity of man-days to the 'Work Item' the total quantity of man-days of the 'Work Item' equals or exceeds the duration of the 'Work Item', the 'Work Item' has finished. In this case the high risk project is carried out in the same continent as where the customer is located (Europe), so

we double the capacity of the European lab (available quantity of man-days: 9 x 2 = 18). Because this is a high risk project the player throws a dice and a 4 is rolled, so the available quantity of man-days stays the same. Finally we should withdraw the overhead fee of the lab from the available quantity of man-days since the project has been moved to the Europe lab in the current round (available quantity of man-days: 18 - 2 = 16). Because the quantity of man-days of the current project exceeds the duration of the current project the project is finished. Finally, when all players are done in time zone Center we move to time zone West. Again all players get the option to play zero or more of the 'Action Cards' they selected.

When the round of play finishes because all players have have had the chance to play 'Action Cards' in all time zones, both the projects finished during the current round and the periodic product payments are paid out to the players. So the player receives 30 million euros because he finished his high risk project.

Overview of a round of play:

Pre-Round Actions:

1. Pay the 'Upkeep' for the labs you own
2. Choose the 'Action Cards' to play in the upcoming round
3. Turn over the 'Event Card' on top of the stack of 'Event Cards'

In-Round Actions:

1. All players play zero or more 'Action Cards' in time zone East
2. All players play zero or more 'Action Cards' in time zone Center
3. All players play zero or more 'Action Cards' in time zone West

Post-Round Actions:

1. Receive payment for projects finished during the past round
2. Receive periodic payment for the finished products in the product payment list
3. Reshuffle the 'Event Card' with the rest of the 'Event Cards'

It may seem complex how the game is played from reading the foregoing example flow. The game has already been played quite a number of times and its playing dynamics have evolved. Experience shows that it takes about one single round before participants understand the game flow and rules, which is quite typical and acceptable for board games.

6. Experiment Set-up

In order to evaluate the learning effects of the game; we have set up a controlled experiment [26]. The subjects in the study were students at Delft University of Technology who are in the master phase of their study on computer science. All students follow an elective course (IN4185 on globally distributed software engineering). The experiment was scheduled at the fourth lecture (in a sequence of seven). Lectures one through three addressed GSE in theory but also included guest lectures from industry elaborating on experiences, best-practices and bad-practices.

Students participated voluntarily in the experiment. In total 16 students participated in the experiment. The group was split in two groups of 8 students (Group 1: Test Group, Group 2: Control Group). The distribution was random, but taking into account: their average grade, mother language and work experience as software engineer. The reason we included these three dimensions in the selection was to ensure that (because of the small size of our sample) the two groups would be as similar as possible.

Each group of eight students was then moved to a separate lab, far apart so there was no interaction between them. When arriving in the lab, the group of eight was subdivided into four teams of two students.

We had prepared a case description with questions to ask students for their insights on global software engineering, specifically focused on challenges and benefits (this case is included in Appendix A). This case and questions were intended to measure the insights of the students into the dynamics of GSE.

The test group first played the game and then took the test (read the case and answered the questions), while the control group took the test without playing the game. In order to give the students the same experience, the control group played the game too, but only after taking the test. As such, playing the game did not have impact on their answers, while the test group was assumed to have a learning advantage from playing the game. With the answers to the case questions we measured the learning effects of playing the game.

As to ensure that the students would take the game and test seriously, we made the game into a contest (champagne for the winners) and were present all the time to ensure things were taken seriously. According to our observations the students were and also the answers to the case and questions confirmed this.

The students were given 2.5 hours to play the game. We would indicate three rounds in advance when the final round in the game would be played. For the case and questions, the students were given 45 minutes. All students completed the test within this time frame.

Figure 2: Photo impressions from the experiment

7. Experiment data analysis

Before going into details on the results of the experiment, we first want to emphasize that the sample size (two groups of eight students each) is too small to draw externally valid conclusions. As such, all the findings below are of an indicative nature.

Having, addressed this major limitation, it is still interesting to observe differences between the test group (having played the game) and the control group (that had not played the game). Although, we could have defined 'correct answers' for the questions to the students, we will not analyse the extend in which they actually came up with these answers. The primary reason for not doing that is that we have not objectively defined what the 'correct answers' are. If we would have wanted to do that we would have needed for example an expert panel of significant size, and furthermore it is not even guaranteed that such a panel would come to a common agreement. To look at the learning effectiveness of playing the game, it is not necessary to know the 'correct answers'. After all, comparing the test group with the control group indicates the differences in learning effects between having and not having played the game. As such, we can see what the learning effects were on the test group and whether this learning effect was intended.

Learning effect: Broader scope on GSE benefits

When looking at the understanding of the participants of the benefits of GSE (as answer to question 2), the test group identified a broader set of benefits (Figure 6). The control group identified 'market proximity' and 'time-to-market' as most dominant while the test group also included cost impacts and access to a sufficiently large and talented work force. Also in question 6b, on involving a site in India, the test group showed a broader view on benefits.

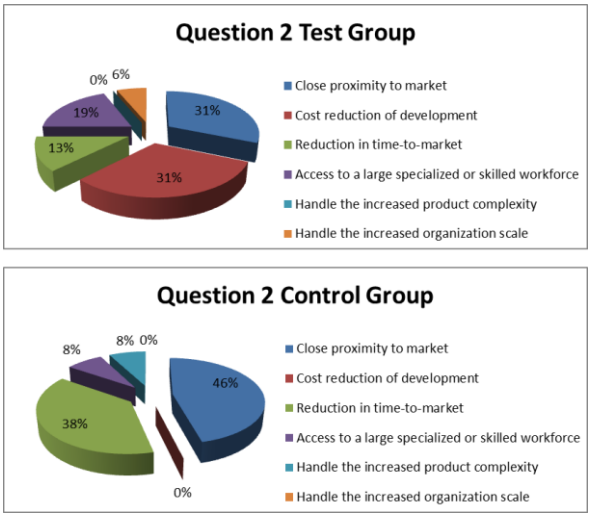

Figure 6: Comparison of groups on GSE benefits

Learning effect: Cost awareness

The test group clearly indicates cost reduction as benefit for GSE (60% indicate it as a benefit to question 2); in contrast to the control group (0%). This is not a surprise from playing the game, as one of the core aspects addressed in the game is labour cost in certain geographical areas. What is surprising is the low percentage of participants mentioning cost benefits in the control group. One reason could be that in the lectures preceding the experiment, several speakers did express that cost reduction alone should never be the main reason to work globally.

Learning effect: Time-to-market dynamics
When looking at the potential time-to-market benefit from working globally, we saw an opposite trend. The test group indicated time-to-market less (25%) than the control group (60%). This is quite remarkable as time-to-market is an important aspect to do well in the game. However, the game does address the complexities involved when pursuing this, namely: distance to the customer and follow-the-sun challenges. As such one could argue that the game might give some insights on the complexity involved with decreasing time-to-market by working globally.

Learning effect: Awareness of cultural impacts
In the answers to the questions we noticed a stronger awareness in the test group on cultural factors. Though the game not specifically addresses cultural factors, it does contain national specific impacts, such as: bank holidays, natural disasters, and political instability. Also actively addressing world-wide locations (using a world map) could have contributed to this awareness. Comparing all answers to the questions, we see that the test group addresses cultural impacts almost twice as much as the control group. However, several of the participants in the test group do not point at cultural factors that strongly, making it doubtful that playing the game directly contributes to awareness of cultural impacts.

Learning effect: Need for transparency
Finally, the test group (60%) identified the need for increased transparency as a factor to pursue when working globally (question 7), compared to the control group (25%). Although, this is not an intended factor addressed in the game itself, it might be caused by the transparency during the game. Participants have a constant overview of the status worldwide and act upon that. This might lead to awareness that in real-life this is quite different, but that is however just a guess.

8. Conclusion and discussion

When preparing students for an industrial or academic career in software engineering, educational institutions should address the globally distributed character of the software engineering profession. As it will be difficult to have students actually participating in a real-life, multi-site, globally dispersed, industrial projects and thus acquiring knowledge from experience, there is a challenge to properly prepare students for their future jobs. One way to teach the dynamics of global development is by having students participate in a game in which these dynamics are at the core.

In this paper we presented a board game that can be used for such a purpose. Furthermore, we performed a student experiment to validate learning effectiveness of board game usage.

Looking at the detailed answers to questions we did see some differences between test group and control group on the way in which benefits of GSE are viewed, the way in which cost reduction is seen as benefit and the extend in which cultural factors are mentioned as impact on GSE.

Overall, the collected data (Figure 7) does not identify a strong distinction between test group and control group. When we add all factors addressed in the game that are mentioned by participants to all questions, we measure only a small difference in favour of the test group. As such we have no clear evidence (yet) that playing this game will directly lead to increased insights in GSE, although on certain specific aspects in some individual answers differences can be observed, which can however, also be caused by the relative small size of the sample.

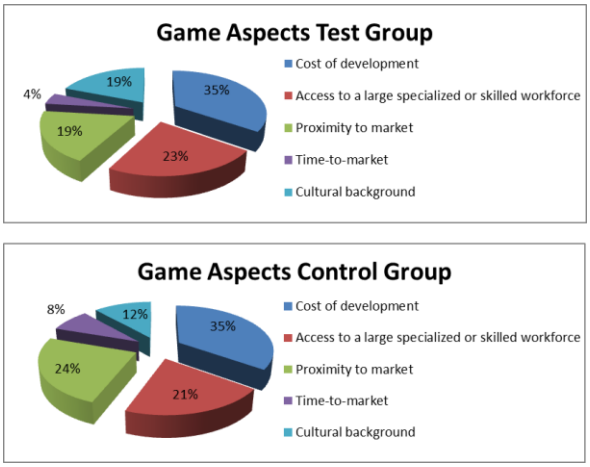

Figure 7: Comparison on overall GSE knowledge

As a last word, it is worth mentioning that the students enjoyed playing the game and gave positive evaluations to the experiment. Following the experiment we collected student feedback on a feedback form, and asked their opinions about using the game in the course. In general the students were positive and indicated they enjoyed playing the game *("Overall very fun game!", "I really enjoyed playing the game", "I really liked the competitive nature of the game")* and playing the game contributed to their understanding of the subject at hand *("It made me think about the various factors that affect decisions for a software engineering company that considers global software development", "it helped me to understand some of the basic aspects of distributed software development").* The average rating for the lectures was 7.4 (stdv 1.1) while the average rating for the game workshop was 8.2. (stdv 1.4).

As such, integrating board games, such as ours, into curricula is likely to be beneficial. Student involvement, fun and motivation are after all also important to facilitate learning processes [6][23][24][25].

9. References

[1] Carmel, E., *Global software teams: collaborating across borders and time zones*, Prentice Hall PTR, Upper Saddle River, 1999.

[2] The Dieringer Research Group Inc. *Telework Trendlines: A Survey Brief by WorldatWork*, 2009, http://www.workingfromanywhere.org/news/Trendlines_2009.pdf.

[3] Wohlin, C., Regnell, B., *Achieving industrial relevance in software engineering education*, In: Proceedings of the 12th conference on software engineering education and training, IEEE Computer Society pp. 16-25, 1999

[4] Baker, A., Navarro, E.O., Hoek, A. van der, *An experimental card game for teaching software engineering processes*, The Journal of Systems and Software, Vol. 75, pp. 3-16, February 2005

[5] Callahan, D., Pedigo, B., *Educating experienced IT professionals by addressing industry needs*, IEEE Software, pp. 57-62, September/October 2002

[6] Ferrari, M., Taylor, R., and VanLehn, K., *Adapting Work Simulations for Schools*, The Journal of Educational Computing Research, vol. 21, no 1, 1999, pp. 25-53.

[7] Grinter, R., Herbsleb, J., and Perry, D., *The geography of coordination: dealing with distance in R&D work*, In: Proceedings of the ACM SIGGROUP 1999 International Conference on Supporting Group Work. ACM Press, 1999, pp. 306–315.

[8] Damian, D. and Moitra, D., *Guest Editors' Introduction: Global Software Development: How Far Have We Come?*, IEEE Software, vol. 23, no. 5, 2006, pp. 17–19.

[9] Carmel, E. and Agarwal, R., *Tactical approaches for alleviating distance in global software development*, IEEE Software, vol. 18, no. 2, 2001, pp. 22–29.

[10] Ebert, C. and De Neve, P., *Surviving global software development*, IEEE Software, vol. 18, no. 2, 2001, pp. 62–69.

[11] Olson, G. and Olson, J., *Distance matters*, Human-Computer Interaction, vol. 15, no. 2/3, 2000, pp. 139–178.

[12] Conchúir, E., Holmström, H., Ågerfalk, P. and Fitzgerald, B., *Exploring the assumed benefits of global software development*, In: Proceedings of the 2006 International Conference on Global Software Engineering, IEEE CS, 2006, pp. 159–168.

[13] Ågerfalk, P.J., Fitzgerald, B., Holmström, H., Lings, B., Lundell, B. and Conchúir, E., *A framework for considering opportunities and threats in distributed software development*. In: Proceedings of the International Workshop on Distributed Software Development, 2005, pp. 47–61.

[14] Herbsleb, J.D. and Moitra, D., *Global software development*, IEEE Software, vol. 18, no. 2, 2001, pp. 16–20.

[15] Ågerfalk, P.J., Fitzgerald, B., Holmström, H., and Conchúir, E., *Benefits of global software development: The known and unknown*, In: Proceedings of the 2008 International Conference on Software Process, 2008, Springer, pp. 1–9.

[16] Moe, N.B. and Smite, D., *Understanding lacking trust in global software teams: A multi-case study*, In: Proceedings of the International Conference on Product-Focused Software Process Improvement, 2007, Springer, pp. 20–34.

[17] Kiel, L., *Experiences in distributed development: a case study*, In: Proceedings of the 2003 International Conference on Software Engineering, 2003, pp. 44–47.

[18] Herbsleb, J., Paulish, D. and Bass, M., *Global software development at siemens: experience from nine projects*, In: Proceedings of the 2005 International Conference on Software Engineering, 2005, ACM Press, pp. 524–533.

[19] Dullemond, K. and Gameren, B. van, *Technological support for distributed agile development*, Master thesis, Delft University of Technology, 2009.

[20] Burnell, L.J., Priest, J.W., Durrett, J.R., *Teaching distributed multidisciplinary software development*, IEEE Software, pp. 86-93, September/October 2002

[21] Randel, J.M., Morris, B.A., Wetzel, C.D., Whitehill, B.V., *The effectiveness of games for educational purposes: A review of recent research*, Simulation & Gaming, Vol. 23 No. 3, pp. 261-276, September 1992

[22] Kraut, R., Fussell, S., Lerch, F., Espinosa, A., *Coordination in Teams: Evidence from a Simulated Management Game*, Human-Computer Interaction Institute, paper 102, (2005), http://repository.cmu.edu/hcii/102.

[23] Bruffee, K., *Collaborative Learning: Higher Education, Interdependence, and the Authority of Knowledge*, John Hopkins University Press, 1983.

[24] Lepper, M.R., & Cordova, D.I. (1992), *A Desire to Be Taught: Instructional Consequences of Intrinsic Motivation*. Motivation & Emotion, 16(3), 187-208.

[25] Malone, T.W., Lepper, M.R., *Making Learning Fun: A Taxonomy of Intrinsic Motivations for Learning*, In: Aptitude, Learning, and Instruction - Volume 3: Conative and Affective Process Analysis. Richard E. Snow, Marshall J. Farr (editors), Lawrence Erlbaum Associates, 1987, pp 223-.

[26] Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A., *Experimentation in Software Engineering: An Introduction*, Kluwer Academic Publishers, 2000.

Appendix A: Case and Questions

The following case and questions were handed out to the participants.

Case information

Consider yourself acting as the CIO of a company called "LogiStrux". You are based in the Detroit Headquarters. Your company delivers software products for complex car manufacturing production processes and logistical processes. Your clients are the major car producers all over the world. You have sales offices close to the headquarters of your major customers (Japan, USA, Germany and France). Your business is growing largely and you do need expansion of your workforce, especially at the engineering level. Your current (and only) software development centre is in Stuttgart (Germany).

For the development of your next generation product (large, highly complex and innovative), you definitely need additional engineers. Your current pool works on the maintenance and servicing of the current product lines, so shifting work is not an option. You need to hire more people. You have, however, many options for ways in doing so, for example you could establish your own development centres globally, continue increasing the staff size in Germany, work with global subcontractors, take-over/buy an existing development centre/company, etc.

Question 1

On what factors would you base your decision regarding how to increase your work force and how would these factors influence this decision?

Please explain why these factors influence your decision in the way you describe.

Question 2

What would be the two most important reasons for you in favour to start working globally at the engineering level?

Please explain why you consider these two reasons most important

Question 3

What would be the two most important reasons for you against working globally at the engineering level?

Please explain why you consider these two reasons most important

Question 4

What would be your financial considerations regarding expanding the engineering workforce? In other words: when you look at the financial side of expanding your workforce, what arguments do you have for your choices?

Please clarify your answer clearly.

Additional Case information

Suppose you have a discussion with an international subcontractor that has a global presence. Working with that subcontractor is one of your options, so you sit down to talk with them, to find out if this could work for you.

This subcontractor has development centres in: Chicago (USA), Frankfurt (Germany), Bangalore (India), Beijing(China).

Question 5

Which of these subcontractor locations would you consider as most interesting options in your discussions with that subcontractor and why would you include this/these?

Please explain your choice clearly

Question 6

This potential subcontractor strongly urges to involve their Indian development centre, in the collaboration with you.

6a. What would be the two most important challenges you expect to arise from involving the Indian site in such a collaboration?

Please elaborate why you think these are the most important ones?

6b. What would be the two largest benefits you expect to arise from involving the Indian site in such a collaboration?

Please elaborate why you think these are the most important ones?

Question 7

What are the most stringent demands you would impose on this subcontractor when involving their Indian site (please focus on operational issues regarding e.g. development tools, working practices, processes and infrastructure)?

Please elaborate why you think that these demands are the most important ones?