

Training Project Managers to Acquire GSD Soft Skills: A Serious Game

Rubén Márquez
Alarcos Research Group
University of Castilla-La Mancha
Ciudad Real, Spain
Ruben.Marquez@uclm.es

Aurora Vizcaíno
Alarcos Research Group
University of Castilla-La Mancha
Ciudad Real, Spain
Aurora.Vizcaino@uclm.es

Félix Oscar García
Alarcos Research Group
University of Castilla-La Mancha
Ciudad Real, Spain
Felix.Garcia@uclm.es

Abstract—A good project manager should develop a set of hard and soft skills if they wish to manage any kind of software project. However, when the team members are geographically distributed, several new skills should be taken into account. Aware of this fact, in this paper we describe the skills and competences that a manager of Global Software Development projects should master, and we sketch out a serious game that can be used to train some of these soft skills, which are grouped under the 3Cs (Communication, Coordination and Control). With this approach we place the focus on the project manager role and seek an immersive and engaging training experience.

Keywords—Global Software Development; Project Management; Soft Skills; Serious Game.

I. INTRODUCTION

Software engineering is an important area of computing, which searches for solutions to complex problems; to reach these solutions, however, it is necessary for software engineers to master a set of knowledge and skills [1]. In the field of software engineering, a new development model has emerged, one which is more offshore than the conventional model, due to the growth of globalization [2]. This development model is called Global Software Development (GSD); it is a term which has come to stay, because it is growing quickly and can be considered an increasing trend in work on software projects [3]. GSD can thus be defined as the development of a software project that involves several work teams, which may belong to different cities or countries [2].

In terms of GSD, the project tasks are separated, and delivered as different jobs. This means that managing GSD project is not a simple task; distance, language and cultural barriers affect Coordination, Communication and Control, thereby increasing the complexity of the management [4]. A variety of problems can therefore occur, such as: strategic and cultural issues, inadequate communication and lack of knowledge among workers, as well as project and process management problems. Future engineers need to be aware of all these potential difficulties, if they are to be prepared for the new world of software development [3].

The points above are supported by a number of articles, which point out that the lack of skills and competences in project managers is the main cause of much project failure [5], especially in GSD projects. This is why many companies explain the need to teach the technical and non-technical skills required in GSD or co-located projects to recently-graduated and newly-qualified Project Management (PM) professionals. Those skills will enable them to manage these

kinds of projects effectively, and produce qualified professionals with this expert knowledge [5], [6].

In many domains, games are used to teach and train different aspects of a given field. Educational games simulate an environment, which help the students to feel motivated, thus improving the teaching-learning process [5]. Games can be therefore be justifiably considered to be a way for future project managers to fill in the gaps in the practical information they possess [1]. Some examples of serious games used to teach GSD and PM are [5], [7].

Taking this fact into account, we have been developing Global Skills Game, a serious game designed to help project managers to develop some skills that are advisable to have when managing a GSD project.

The rest of the paper is organized as follows. In Section II a set of useful soft skills for working in GSD project and in Section III for working in PM are specified. In Section IV describe our approach, Global Skills Game along with its characteristics. Finally, in Section V, the conclusion and future work are given.

II. SKILLS AND ABILITIES FOR GSD

Anyone who works on a software project must have a set of skills that work towards the success of the project, since, as previously stated, many of the failures in software projects are due to a lack of skills and competences on the part of those working on the project. In a GSD project in particular, the members really must possess certain specific skills.

First of all, and according to [3], some of the most useful skills for software development are being able to *speak the English language*, *local cooperation* and *decision making*. Nevertheless, those abilities that provide the greatest benefit in distributed environments are, in addition to the very important *speak the English language: remote cooperation*, *intercultural cooperation* and *cooperation with clients*.

Besides, and according to [6] some skills which must be provided in training for a GSD are *to be aware of all possible problems*, mastering *communication protocols* (especially using computers), *oral and written communication through a common language* and *codes of ethics* and *time management*.

In [8], the authors indicate that the important skills that should be taught to students of GSD are *regular communication with distributed team members*, *team dynamics*, *working in culturally diverged teams*, *managing time* and *using collaborative technologies*.

Moreover, a framework to teach several GSD skills is described in [9]; the main skills considered are: *computer*

mediated communication, iterative development in remote client-developer relationships and distributed PM.

The competence model specified in [10] indicates what the general competences for GSD teams are, dividing them into four groups depending on the roles that are involved in a GSD team (software engineer, team leader, project manager and organizational unit manager). The abilities that are pointed out as being necessary in the training of any software engineering who is working in GSD are *synchronous and asynchronous communication, identification and management of requirement, technical problem solution, share knowledge management, advanced techniques for distributed communication, self-learning capacity, international relationship ability, use of communication and information technologies, ability to work in a global setting and oral and written communication in English.*

III. SKILLS AND ABILITIES FOR PM

Once we have carried out an analysis of GSD, listing the skills and competences which are needed in this type of development environment, in this section we are going to focus on the competences for PM.

First of all, the authors in [11] state that risk is one of the most important parts of PM. It is thus highlighted that a distributed software development project manager has to take into account areas of risk such as *time zone, cultural, geographical and language differences*, as well as the processes of *Coordination, Communication and Control*.

In addition to these considerations, the research in [12] offers a range of skills which must be acquired by the project manager in software development if he or she is to successfully achieve the project goals. The skills that are proposed are: *communication skills* (the project manager should listen to the team workers, promote trust relationships and understand the personality differences between its members, with the aim to improve the work process, reduce conflict between workers, and strengthen cooperation); *team building skills* (the project manager must ensure there are strong links formed between other team members and other teams); and *problem-solving skills* (the project manager has to visualize and solve complex problems, making decisions which may have a certain impact on the project cost, quality and productivity).

Finally, the skills that are proposed in [10] as necessary for a project manager in a distributed environment are *decision taking* (the choice between different alternatives in different aspects of PM), *meeting management* (holding meetings with different workers to talk about PM), *establishment of rules to work with shared data* (due to the fact that data is shared between different work teams, rules must be established for its use), *collecting, analysing and interpreting information* (all the necessary project documentation must be processed in order to take the right decisions), *positive attitude and capacity for motivating others* (in this way the project workers also have a positive attitude and their productivity increases), *organization and planning capacity* (the project manager has to carry out the organization and planning of the different tasks of the project), *initiative and leadership* (the project manager is the

key component in a distributed project; hence, he or she must have an attitude of leadership and initiative for decision taking), *interpersonal conflict resolution* (since there are many work teams in a distributed project, it is not uncommon for different workers to have conflicts, so the project manager has to solve these by taking the right decision), *identification of competence and CV* (the project manager should be able to know the skills and competences of workers, so as to assign the workstation correctly), and *requirement estimation and prioritization* (the project manager must know the requirements of the project and carry out the prioritization of each one according to the needs of the client).

To sum up, the above study (an overview of the soft skills needed when working in GSD and PM) has helped us to decide which competences we want to implement in our serious game.

IV. THE GAME

In the previous sections we have presented a study of the current literature on the skills needed to work in a distributed software environment, and on those skills needed to perform the activities of a project manager.

In this section our serious game is presented, indicating in detail what it consists of, what particular skills our player will learn, and how he or she will face with different elements to advance towards the successful development of a distributed software project.

Before we start implementing our game, we should first of all reach a decision about which skills from Section II and III will be taught and improved by playing it. It should be noted that the goal of this game is to train individuals for management of software projects in distributed environments, so both GSD and PM skills should be included. Taking that into account, in Table I a set of soft skills which will be implemented in our game is presented, indicating in each case whether it is a skill that concerns Global Software Development (GSD), or Project Management (PM), or both (BTH); the table also shows how the soft skill will be implemented, and how the player will be able to acquire it.

As we can see in the table below, we are going to introduce several skills into our game; however, there are many other GSD and PM skills that have been left out for some reason. These include such skills as those to do with codes of ethics, since these almost always depend on the particular kind of the project, companies, and even countries involved, so their implementation would be too complex. Another not supported skill is that of establishing rules to work with shared data; due to like the previous skill, it will depend on the particular company(ies) involved in the project. As far as the skills positive attitude, capacity to motivate others, or initiative and leadership are concerned, these are considered as very personal skills that will depend on the disposition of the person involved. Therefore, we are going to implement some strategies to develop those skills, or at least to make the player aware of the importance of having them. However, it can be a challenge for a serious game to evaluate them.

TABLE I. SOFT SKILLS IMPLEMENTED IN GLOBAL SKILLS GAME

Soft Skill	Type	Implementation
Time management	GSD	There will be some information in the game about the evolution of the project being managed; this information will be the current budget, time-to-delivery, and the current progress. The player will thus be able receive training in how to conduct the time management of the project, since he/she will also be able to carry out activities designed to optimize the above information.
Intercultural cooperation	GSD	Some of the events that will occur will have certain characteristics that are specific to a given culture; in this way the player will learn how these features affect intercultural cooperation between globally-divided work teams, and they will be able to try to improve the cultural relationship between the members of the project.
Cooperation with client	GSD	Other events will be directly or indirectly involved with the needs of the client of the project, thus enabling the player to learn how to cooperate with the customer in such a way that the project meets their expectations. Furthermore, the player will be able to communicate with the client and request them to answer any queries that may arise.
Communication skills	PM	One of the types of event that will come up in the game will be communication events, in which the player will learn the characteristics that are involved in the communication between the members of a distributed software project; these events will train the ability to improve that communication.
Coordination skills	PM	Another type of event that will occur in the game will be coordination events, in which the player will learn the characteristics that are involved in the coordination of the project, and this will train their ability to carry out different project coordination activities.
Control skills	PM	Control events will be another kind of event in the game; in these, the player will learn how to control the project, and receive training in how to solve a problem, should one occur.
Decision making and problem-solving	BTH	Throughout the execution of the game different events will appear in which some problem arises; the player must decide what is the best way to maintain the execution of the project and work out how to solve the problem.

Once we decided the skills that players will learn and receive training in by playing our game, we designed and implemented the game’s graphical interfaces. The first interface is the general menu of the serious game.

After choosing the context of the game project, the player will have to configure different game parameters which will affect the simulation of the game. The parameters with which the player will be able to interact will be both the general aspect of the project, such as the number of sites, the country where the client is located, the common language for communication between sites, and the specific aspects of each site. These details include the number of workers, the country where the site is located, the knowledge of the common language, or whether the site is the main one or not. In addition, the player will be able to choose the type of communication between the different sites and the client. While the project is being configured by the player, the game will show a set of important success factors that are involved in, and characterize, the development of a distributed project, as collected by [13]; some of these success factors are “Working time overlap”, “Language difference”, “Cultural difference”, “Communication” or “Sites number”. Finally, and by means of the above factors, the game will calculate a level of difficulty, from low to very high, and will represent the difficulty that the distributed project has to face with, taking into account the current configuration of the player. This screen will enable the player to begin to become familiar with the aspects and factors involved in a distributed project and learn from the feedback, as these factors can have an important influence on the project.

Finally, and once the distributed project that we are going to simulate is configured, the graphic interface of the game

where the simulation of the game is to take place will appear. This screen (Figure 1) is divided into two parts; the bottom strip in which the player, through progress bars, will find information that will evolve throughout the simulation of the game. This includes information such as the level of stress, the remaining budget, the time until the delivery of the project and the progress of the project. There will also be general information on the project, such as its difficulty, initial budget and duration. At the bottom of this screen we can find information on each of the sites, such as the level of motivation, communication with other sites, or workload. Turning now to the upper part of the screen, in which the graphic game will be found, this will consist of the static map of an office, along with the character of the player, who will be able to move about with total freedom.

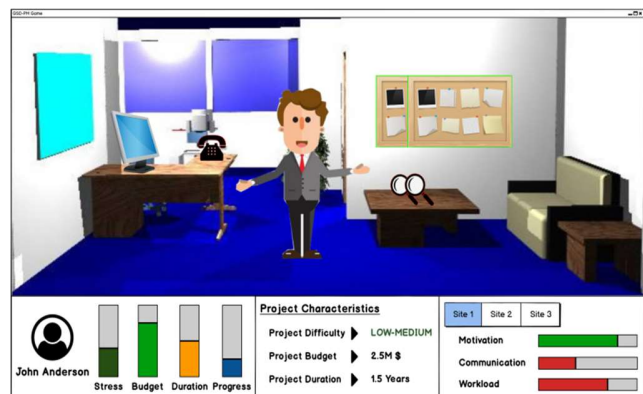


Figure 1. Global Skills Game Graphic Interface

The simulation of the game will consist in the dropping of different objects onto the screen (where the speed of the drop will depend on the difficulty of the project). These objects will be a telephone, a note and a magnifying glass, and each of them will represent an event that has occurred in the simulation of the project, identifying the particular type of event according to the 3Cs, Communication, Coordination and Control, respectively. When an object drops, the player will be able to interact with it so that the description of the event and what has happened appears; in this way the player will be able to train ability in each one of the 3Cs and understand how each one of these affects the evolution of a distributed project. In addition, some of the events will have certain cultural characteristics which will help the player to understand the intercultural cooperation of a distributed project and train in how to deal with this. Other events will have the client of the project as the main actor. This will help the player to understand cooperation with the client and train their skill in this area. The events with which the player interacts will have an impact on the execution of the project, affecting the budget, duration or even the stress of the player; this impact may be felt in any of the different sites, and it may be positive or negative, depending on the particular type of event. Negative events will consist of problems which the player as project manager will have to tackle by taking certain decisions to solve the problems in the best possible way. These decisions can be made in two ways; either by choosing a solution from the whole range that is offered (if offered) or by interacting with the computer and by performing an activity that solves the problem. The user will be able to use the computer to carry out different activities to try to improve the execution of the project; these include tasks such as hiring and firing personnel, giving extra payments or holding a face-to-face meeting.

The project simulation will continue until several possible results occur: the player has overcome the stress level; the project has run out of budget; the project has not met the deadline. In those cases, the player has lost the game. If the project has finished successfully on time and within budget the player has won the game.

V. CONCLUSION AND FUTURE WORK

This paper presents a study carried to find out what soft skills a project manager needs to learn when working on a GSD project. Taking this study as a basis, we then went on to develop a serious game to train future project managers. The fact that it is a serious game offers the advantage of its being much more entertaining than other traditional training methods.

As future work we will focus on testing our serious game on students of software engineering, in order to evaluate it. We will also test it on practitioners, in the quest to make possible improvements.

ACKNOWLEDGMENTS

This work has been funded by the G3SOFT project (Consejería de Educación, Cultura y Deportes de la Junta de Comunidades de Castilla La Mancha, y Fondo Europeo de Desarrollo Regional FEDER, SBPLY/17/180501/000150)

and by the BIZDEVOPS-GLOBAL project (Ministerio de Ciencia, Innovación y Universidades, y Fondo Europeo de Desarrollo Regional FEDER, RTI2018-098309-B-C31).

REFERENCES

- [1] M. Kosa, M. Yilmaz, R. V. O'Connor, and P. M. Clarke, 'Software engineering education and games: A systematic literature review', *Journal of Universal Computer Science*, vol. 22, no. 12, pp. 1558–1574, 2016.
- [2] A. Vizcaíno, F. García, I. G. R. De Guzmán, and M. Á. Moraga, 'Evaluating GSD-aware: A serious game for discovering global software development challenges', *ACM Transactions on Computing Education*, vol. 19, no. 2, pp. 1–23, 2019.
- [3] I. Bosnić, I. Čavrak, and M. Žagar, 'Assessing the impact of the distributed software development course on the careers of young software engineers', *ACM Transactions on Computing Education*, vol. 19, no. 2, pp. 1–27, 2019.
- [4] I. Richardson, S. Moore, D. Paulish, V. Casey, and D. Zage, 'Globalizing software development in the local classroom', presented at the Software Engineering Education Conference, Proceedings, 2007, pp. 64–71.
- [5] J. E. N. Lino, M. A. Paludo, F. V. Binder, S. Reinehr, and A. Malucelli, 'Project management game 2D (PMG-2D): A serious game to assist software project managers training', presented at the Proceedings - Frontiers in Education Conference, FIE, 2015, vol. 2014, pp. 1–8.
- [6] M. J. Monasor, A. Vizcaíno, and M. Piattini, 'Training Global Software Development skills through a simulated environment', presented at the ICSOFT 2010 - Proceedings of the 5th International Conference on Software and Data Technologies, 2010, vol. 2, pp. 271–274.
- [7] J. Noll, A. Butterfield, K. Farrell, T. Mason, M. McGuire, and R. McKinley, 'GSD Sim: A Global Software Development Game', presented at the Proceedings - International Computer Software and Applications Conference, 2014, vol. 18-21-August-2014, pp. 15–20.
- [8] M. Paasivaara, C. Lassenius, D. Damian, P. Rätty, and A. Schröter, 'Teaching students global software engineering skills using distributed Scrum', in *2013 35th International Conference on Software Engineering (ICSE)*, 2013, pp. 1128–1137.
- [9] D. Damian, A. Hadwin, and B. Al-Ani, 'Instructional design and assessment strategies for teaching global software development: A framework', presented at the Proceedings - International Conference on Software Engineering, 2006, vol. 2006, pp. 685–690.
- [10] J. Saldaña-Ramos, A. Sanz-Esteban, J. García, and A. Amescua, 'Skills and abilities for working in a global software development team: A competence model', *Journal of Software: Evolution and Process*, vol. 26, no. 3, pp. 329–338, 2014.
- [11] J. M. Verner, O. P. Brereton, B. A. Kitchenham, M. Turner, and M. Niazi, 'Risks and risk mitigation in global software development: A tertiary study', *Information and Software Technology*, vol. 56, no. 1, pp. 54–78, 2014.
- [12] K. Sutling, Z. Mansor, S. Widarto, S. Lecthmunan, and N. H. Arshad, 'Understanding of Project Manager Competency in Agile Software Development Project: The Taxonomy', in *Information Science and Applications*, Berlin, Heidelberg, 2015, pp. 859–868.
- [13] A. Vizcaíno, F. García, J. C. Villar, M. Piattini, and J. Portillo, 'Applying Q-methodology to analyse the success factors in GSD', *Information and Software Technology*, vol. 55, no. 7, pp. 1200–1211, Jul. 2013.