

# MEDEDUC: An Educational Medical Serious Game

Vitor Manuel Fragoso Ferreira, Rosa Maria E. Moreira da Costa<sup>(✉)</sup>,  
and Vera M.B. Werneck

Universidade do Estado do Rio de Janeiro – UERJ,  
Instituto de Matemática e Estatística –IME,  
Mestrado em Ciências Computacionais,  
Rua São Francisco Xavier 524-6° andar – Bl. B, Rio de Janeiro  
RJ 20550-013, Brazil  
vitor@fragoso.me,  
{rcosta,vera}@ime.uerj.br

**Abstract.** In general, serious games motivate people from all ages and have been applied in many areas to stimulate the learning processes. However, the adoption of specific methodologies to construct these games is still rare. This paper aims at presenting an experiment that applied an agile methodology to develop the MEDEDUC. It is a multi-agent game to improve learning and clinical performance of students from the health care area. This game allows the student to learn pulmonology by answering questions, in five levels of difficulty and exploring multimedia presentations. MEDEDUC was developed using SCRUM agile methodology and the results stressed some difficulties associated to the adoption of this agile methodology.

**Keywords:** Multi-agents systems · Scrum methodology · Agile methodologies · Medical serious games

## 1 Introduction

Serious games provide a problem situation environment [5] and there is a potentially effective use of games for medicine students; however there are a limited number of studies [2].

Multi-agent systems (MAS) are composed of autonomous modules that perform activities and communicate with each other in order to achieve a common goal. The construction of multi-agent systems involves different characteristics from those employed in traditional software development. According to Jubilson [8], the conception, design and implementation of such systems that operate in distributed environments, must consider new development process perspectives.

Some agent-oriented methods have been proposed for building MAS but there is no consensus about the best one that must be applied in MAS development [12]. In general, these methodologies are complex to use, the time consuming is high and generate a great number of diagrams and documentation. In addition, these processes have not been able to reach the desired flexibility.

To overcome these problems in the MEDEDUC development process we considered concepts of agile methodologies such SCRUM. This methodology improves software production focusing on customers' requirements, frequent deliveries, daily meetings, review meetings and retrospective meetings.

The Agile SCRUM [10] proved to be able to deliver a final product with better requirements; the modules were completed and presented to clients' review with a short iterations period. In addition, the development team is always ready to include all the desired requirements and improve the game.

The MEDEDUC game focuses on the pulmonology area by letting the student answer questions with different levels of difficulty. The questions present images, sounds and videos to enrich the communication and the learning process.

The game starts testing the students' previous knowledge to direct him to the respective game level. If the game identifies that the student is having a high number of errors a basic knowledge module will be active by the intelligent agents. The student is rewarded when each module ends with success. The awards are medical equipments which compose the student virtual office such as stethoscope, x-ray machine, computer or others. At the end of the game, the student will have his office set up. The intelligent interactive interface opens new possibilities to have fun in the learning process and to keep the student involved into the game activities.

The relevance of this study is based on the lack of experiments that integrate, in a practical way, issues such as multi-agent systems, medical educational games and agile methodologies.

## 2 Serious Games

According to Blackman [4], serious games compose a special category of games that complement a fun activity with specific content.

Serious games in Educational area promote a competitive universe with pre-established rules [7]. The playful and challenging approaches motivate users to accomplish tasks.

Specifically, in the health care area, the serious games offer a great potential to improve the clinical performance of medical students. In this case, they can transfer the game situations to real life. Several studies have suggested beneficial effects in the use of educational games in health and education [13]. In this context, we have some examples of success.

Abreu et al. [1] created a multi-agent 3-D serious game to stimulate cognitive functions of people with different disabilities due to trauma or brain accident. Scarle et al. [9] developed the Match-3 serious game that was designed to combat childhood obesity using the Wii-system. Atkinson and Narasimhan [3] created a medical gaming environment for diagnosis and management of Parkinson's disease using a haptic interface.

When these products are created we must consider some aspects related to costs, time and effort required for their development. The adoption of a software development methodology can aid to avoid some of problems related to those items.

Next we present the SCRUM methodology which considers a dynamic process to create software.

### 3 SCRUM Methodology

Scrum [11] is an agile development methodology with incremental and iterative processes that has good management practices set providing quick adjustments, monitoring, constant visibility and realistic plans.

The Scrum development process iterations are called sprints, which should last two to four weeks. This time aims at promoting greater visibility throughout the development process by the possibility to delivery small functionalities system with features being validated with the customer and, if necessary, modified promptly to avoid waste of work. The team, as far as possible, should be kept in the constant size, allowing the collected metric and estimates become efficiently [11].

Scrum is an open model and is not predictable [10]. In general, we can identify some important points: (i) self-organizing teams; (ii) development progress through sprints, sprint being a Scrum development cycle, characterized by a short period, where the team focus to achieve a specific goal; (iii) Requirements of products organized in a list of items called Product Backlog; (iv) Closed concept of time (Timeboxed) that are all tasks within the Scrum having maximum time set for its implementation and must not supplant these times [11].

The customer is close to the process and becomes almost part of the team, indicating how they want their product and how it should be output. Their stories are told and requirements appear spontaneously.

The sprint is divided into stages, the planning meeting (Sprint Planning) is the first stage, where the team (Scrum Team) together with the customer (represented by the product owner), define what will be developed and implemented in the iteration, creating the Product Backlog. The Product Backlog (Sprint Backlog) is a list containing the business features, technical requirements and the errors found in the system, that need to be developed or improved. This list is in constant evolution, because new business requirements can appear and also the software must be maintained or re-factored. Each increment of the Product Backlog must be implemented during the Sprint.

The Product Owner role has the responsibility of maintaining the list of priorities and shows the prospects of the project to the team. A Scrum challenge is to keep an organized Project Backlog, either with the priorities or with respect to the complexity necessary for the stories fulfillment.

After prioritizing the Project Backlog items, the details of the software are described by creating stories and tasks necessary for the Sprint development. The story is the element that describes a feature, an improvement or correction to be implemented, indicating what will be developed and how it will be done.

In Scrum there are tasks follow-up meetings of each Sprint [11]. For this, there must be a plan that should contain the goal, team members, the Sprint itself, the date and time of presentation of the Sprint.

During the Sprint the team performs Daily Meeting, that should not take more than 15 min, and they aims at investigating the development progress using for this the Burn-down Chart which is a graph used for tasks monitoring in progress and finished.

In the whiteboard, the goals and requirements are written on Post-it notes and are placed in order of their priority. They are then allocated to the phase in which its implementation is: not started, in progress or completed. Their team members change the status of each requirement in the frame. In this white board, too, is the Burn-down Chart, which is a graph showing the progress of work.

At the end of Sprint are held two other meetings, one for the delivery validation (Sprint Review) where the customer and others who are interested in the product can verify if the sprint goal was achieved. After, sprint retrospective meeting is performed only by the staff, evaluating the Sprint in the process perspective, team or product, noting the successes and mistakes in order to improve the work process for the next Sprints.

In addition to the Sprint planning meeting, the Scrum addresses other follow-up meetings of the processes, such as: (i) Sprint Review that is the presentation of the increases made at the end of each Sprint Backlog; (ii) Daily Scrum which is tracking and monitoring processes; (iii) Retrospective Meetings which is the reflection of what has passed and planning of actions to be taken to achieve the objectives; (iv) FUP Meeting which is the detailing and elucidations concerning requirements; (v) Re-Estimate Meetings which is an application of techniques such as Poker Planning for estimating requirements; (vi) Sprint Planning 2 which is the discussion of technical aspects, such as product architecture, component reuse, etc.

There are different roles in Scrum: Scrum Master, Product Owner and Team (team working on processes). The Scrum Master is responsible for facilitating the work, by helping the team to solve problems and difficulties during the sprints and by disseminating and applying the process in the organization. The Product Owner has the function of representing the interests of all involved in the process (stakeholders), defining the product features and prioritizing items in the Product Backlog. The Product Owner is responsible for creating a guideline for the Sprint Planning.

The Scrum team should be multidisciplinary, self-organized and small with a maximum of 10 to 15 participants. The team should be also independent and have control over the development process, and the responsibility, at the end of each sprint, to present the results of the work for the customer.

Scrum features are: (i) Customers become part of the development team; (ii) transparency in the planning and development; (iii) frequent meetings with stakeholders (all involved in the process) to monitor progress; (iv) problems are not “left out” and no one is penalized for recognize or describe any unseen problem; (v) working hours should be optimized in the sense that “overtime” does not necessarily mean “produce more”; (vi) Teamwork is critical to the success of the process.

#### **4 MEDEDUC: A Medical Serious Game**

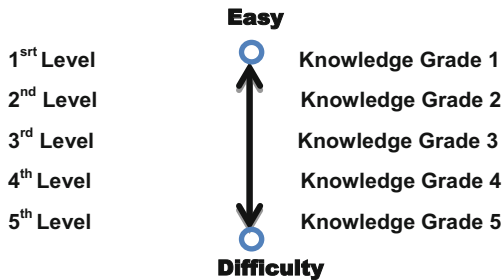
MEDEDUC is a medical educational game developed to support Medical students by teaching concepts of Pulmonology, combining fun and learning.

The students must answer questions organized in difficulty levels and are rewarded at end of each module. This award is medical equipment such as stethoscope, x-ray machine, computer or others, and this award will compose his/her virtual office. By the end of the game (fifth level), the student will have his/her office set up.

A test is performed to verify the student previous knowledge, thus the game can directed the student to the respective level of knowledge. During the game if the student is having a high number of errors, he/she will have to learn basic subjects related to the active module. The interactive, fun and intelligent environment are fundamental to keep the student involved into the game activities.

The MEDEDUC has 110 questions with three answer options for each, divided into 5 knowledge levels (Fig. 1) and the Reinforcement Level.

The basic requirements for the game are: (i) each module will be available to the user gradually, according to his/her progress in the game; (ii) the user should be informed when the result of a module is positive or negative; (iii) the higher the level, the greater the complexity assigned to each question [6].



**Fig. 1.** Knowledge grade by level

The game behavior requirements for Level and Reinforcement modules are: “(i) the questions explore audio, video, images and text items; (ii) Each question has three response options (A, B and C); (iii) When selecting an incorrect option, the user must be informed of the correct option; (iv) When selecting an incorrect option, the user must receive a return with a justification reporting why the chosen option is incorrect. (v) Upon entering the game for the first time, you must perform the Leveling and other features will be unavailable; (vi) During the Leveling, the user must answer 10 questions related to five levels. For each level, there are two questions distributed randomly; (vii) When the user hit a question, a green indicator in the status bar should be shown. For errors, the user sees a red indicator. (viii) By clicking on the indicative position of each question on the status bar, the user should be able to review the question, answer and return. However, the user cannot change the selected option; (ix) The user should perform the Level 5, even if it hits all the questions. When the user finish answering the questions of the Leveling, the user should receive a return stating the level reached.” [6]. Figure 2 presents a Leveling Module interface.



Fig. 2. MEDEDUC level model prototype [6]

## 5 Developing MEDEDUC Using SCRUM

Agile methodologies are well known to emphasize communication and customer engagement. However, for complex problems, modeling methodologies helps the solution design and team communication. This section presents SCRUM development experience applied to a MAS game that was considered a successful operation prototype.

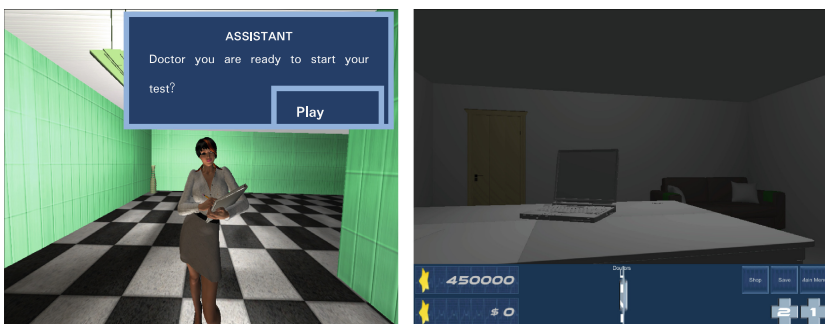
We needed almost four months to construct the MEDEDUC prototype. The team was composed of five undergraduate students in computer science, a master's student and two research professors. The master student is an expert of agile methodologies and performs the Product Owner role with the support professors that are expert in Multi-agent System development and Virtual Reality games in the health care area. At first a Medicine professor supported the definition of the MEDEDUC requirements as defined in Sect. 3 but this professor did not have the contact with the whole team.

The team has experience in software development processes but never developed MAS. Some of them had already worked constructing games and had theoretical knowledge about Scrum. However, they have a few experiences in real projects. So the first two weeks they attend lectures about agents, MAS and Scrum methodology. After that they present a Sprint Planning of the first sprint. Table 1 shows the Sprint Planning of this step.

For each Sprint, the group and the Product Owner held a meeting, where members of the group summed up the work evolution. This meeting was performed in about two weeks. The team members thought that sometimes the Sprint duration was short of time, especially when they had to developed complex algorithms to represent the agents' behaviors. Another point considered was that the several changes resulted of these review meetings required them to understand and integrate these new functionalities in the system in a short time, to be ready for the next meeting. Figures 3 and 4 show the user interface of MEDEDUC.

**Table 1.** Sprint planning report

	Artefact	Preview date	Real data	# Days late
1	Report of Product Delivers Sprint 1	22/10/2014	22/10/2014	0
2	Sprint 1 Revision Meeting	04/11/2014	04/11/2014	0
3	Report of Product Delivers Sprint 2	05/11/2014	06/11/2014	1
4	Sprint 2 Revision Meeting	18/11/2014	18/11/2014	0
5	Report of Product Delivers Sprint 3	19/11/2014	19/11/2014	0
6	Sprint 3Revision Meeting	02/12/2014	02/12/2014	0
7	Report of Product Delivers Sprint 4	03/12/2014	03/12/2014	0
8	Sprint 4 Revision Meeting	16/12/2014	16/12/2014	0
9	Report of Product Delivers Sprint 5	17/12/2014	17/12/2014	0
10	Sprint 5 Revision Meeting	13/01/2015	13/01/2015	0
11	Game deliver	19/01/2015	19/01/2015	0
12	Game presentation	22/01/2015	22/01/2015	0

**Fig. 3.** Main screen**Fig. 4.** Game final screens with a character and the office

At the end of the development process, the owner considered that the final product met the proposed requirements.

In Scrum, the client must be all the time available to the team. However, despite this, as this methodology is not specific to work with intelligent agents, we observed some problems to identify and implement agents' features. The lack of models created difficulties for developing the agent architecture. In general, agile methodologies do not consider a high quantity of documentation. Then, to alleviate these problems, the team documented some agents' features to improve the understanding of its functionalities.

Another point observed was that the excess of meetings during the development process can increase the number of client requirements because the client sees the system working and proposes new features to be incorporated in the system. In this case, the system is never considered finished.

## 6 Conclusions

This paper described the adoption of an agile methodology to develop an agent oriented medical educational game. In this case, we explored the SCRUM agile methodology that is not specific to develop MAS.

By one side, the related experiment tried to verify the potential of this methodology to develop MAS. At the other side, our current researches try to develop serious games to be used by students from health care areas as a learning method.

We identified some fragility in the adoption of this methodology, which can overcome by integrating usual practices imported from other specific methodologies for MAS' development. But, as we had a decrease in the time of development of such complex system, we considered that we can use the SCRUM, which is non specific MAS methodology, to develop agent oriented systems.

## References

1. Abreu, P.F., Carvalho, L.A.V., Werneck, V.M.B., Costa, R.M.E.M.: Employing multi-agents in 3D game for cognitive stimulation. In: Proceedings of the XIII Symposium on Virtual and Augmented Reality, Uberlândia, vol. 1 (2011)
2. Akl, E.A., Sackett, K., Pretorius, R., Erdley, S., Bhoopathi, P.S., Mustafa, R., Schunemann, H.J.: Educational games for health professionals. *Cochrane Database Syst. Rev.* (1), CD006411 (2008)
3. Atkinson, S.D., Narasimhan, V.L.: Design of an introductory medical gaming environment for diagnosis and management of Parkinson's disease. In: Proceedings of the 2nd International Conference on Trends in Information Sciences and Computing (TISC 2010), pp. 94–102, December 2010
4. Blackman, S.: Serious Games... and Less! *Comput. Graph.* **39**(1), 12–16 (2005). ACM
5. Connolly, T.M., Boyle, E.A., MacArthur, E., Hainey, T., Boyle, J.M.: A systematic literature review of empirical evidence on computer games and serious games. *Comput. Educ.* **59**(2), 661–686 (2012)



6. Ferreira, V.M.F., Carvalho, J.C.C., Werneck, V.M.B., Costa, R.M.E.M.: Developing an educational medical game using AgilePASSI multi-agent methodology. In: IEEE 28th International Symposium on Computer-Based Medical Systems, pp. 298–303 (2015)
7. Janarthanan, V.: Serious video games: games for education and health. In: Proceedings of the 9th International Conference on Information Technology (ITNG 2012), pp. 875–878, April 2012
8. Jubilson, E.A., Joe Prathap, P.M., Vimal Khanna, V., Dhanavanthini, P., Vinil Dani, W., Gunasekaran, A.: An empirical analysis of agent oriented methodologies by exploiting the lifecycle phases of each methodology. In: Satapathy, S.C., Govardhan, A., Srujan Raju, K., Mandal, J.K. (eds.) Emerging ICT for Bridging the Future - Proceedings of the 49th Annual Convention of the Computer Society of India (CSI) Volume 1. AISC, vol. 337, pp. 205–214. Springer, Heidelberg (2015)
9. Scarle, S. et al.: Complete motion control of a serious game against obesity in children. In: Proceedings of the 3rd International Conference on Games and Virtual Worlds for Serious Applications (VS-Games 2011), pp. 178–179, May 2011
10. Schwaber, K.: Agile Project Management with Scrum. Microsoft Press, Redmond (2004). ISBN: 978-0-7356-1993-7
11. Schwaber, K.: Scrum Guide (2009). [http://www.training.com.br/download/GUIA\\_DO\\_SCRUM.pdf](http://www.training.com.br/download/GUIA_DO_SCRUM.pdf). visited in 2016 (in Portuguese)
12. Shehory, O., Sturm, A.: A brief introduction to agents. In: Shehory, O., Sturm, A. (eds.) Agent-Oriented Software Engineering, pp. 3–11. Springer, Heidelberg (2014)
13. Wattanasoontorn, V., Hernandez, R.J.G., Sbert, M.: Serious games for e-health care. In: Cai, Y., Goei, S.L. (eds.) Simulations, Serious Games and Their Applications. Gaming Media and Social Effects, pp. 127–146. Springer, Singapore (2014)