Applying multiplayer role based learning in engineering education: Three case studies to analyze the impact on students' performance

Pilar Sancho, Ruben Fuentes, Pedro Pablo Gomez-Martin, Baltasar Fernandez-Manjon

Dpt. of Software Engineering and Artificial Intelligence at Complutense University of Madrid

pilar@sip.ucm.es, *ruben@fdi.ucm.es*, *pedrop@fdi.ucm.es*, *balta@fdi.ucm.es*

ABSTRACT

In this paper we present some results obtained in an ongoing research project aimed at developing a collaborative 3D fantasy virtual learning scenario (which we have given the name Nucleo) for teaching subjects related to computer programming in engineering education and technical schools. Our system has three basic objectives. The first is to induce a change in the students' attitude towards study by placing them at the center of the learning experience. The second is to increase students' motivation by bringing multimedia formats, content interactivity and the aesthetics currently preferred by engineering students into the learning scenario, while at the same time, maintaining a clear focus on the curriculum's learning objectives. The third is to provide a learning environment that simulates the way people work in teams in a real-life professional context, focusing on developing basic teamwork abilities and important soft and social skills. The system is currently being tested as several engineering educational contexts to prove some basic hypothesis. Here we describe the main features of our system, the results obtained in three different cases of study and a discussion about what conclusions can be drawn from them.

1. Introduction

Computer programming disciplines are part of the curricula in several different Engineering Schools and also in many Spanish Technical Schools. However, we have detected deficiencies and problems with teaching these disciplines in the Computer Science School and in the Electrical Engineering School at the Complutense University of Madrid:

- In the last five years drop-out rates have increased and a more and more passive attitude on the part of students in the classroom has been observed. There is a recent research tendency that insinuates the negative influence of students' early exposure to multimedia and technological devices in this phenomenon [18] [32] [33] [31]. According to this school of thought, traditional learning formats are no longer appealing to young people.
- Traditionally, very little attention has been paid to developing students' teamwork abilities, such as conflict management, handling of process coordination, communication among team members, task division and planning or leadership. Although the acquisition of these sorts of abilities is always desirable, they are of key importance in the domain of software engineering and information systems. The development of software is usually the result of the coordinated efforts of members of a team in which everyone plays a concrete role. Nevertheless, the computer programming curricula in our faculties is mainly focused on developing theoretical concepts and technical abilities.

The Nucleo system uses a problem based learning (PBL) framework that is staged inside a virtual world of games and role plays, conceived to be applied in a blended learning setting. PBL has proven its efficiency over the years in helping to develop not only students' technical abilities but also teamwork skills [14]. Nevertheless, it is not easy to implement effective PBL, especially in non face-to-face settings, one of the main obstacles to overcome being the capacity to create effective group dynamics among virtual team members. Research literature on virtual learning teams indicates that there are a number of characteristics for successful, dynamic and high performance virtual teams. These include genuine interdependence, rotating leadership, high levels of trust in other members of the team, social communication, ability to work on a common group goal, and tools to support project management [14]. According to these works, it also seems clear that the feelings of belonging to a community lead to greater commitment, greater cooperation, and greater satisfaction. Nucleo applies four combined strategies in order to increase the effectiveness of the collaborative learning experience:

- The learning takes place in the context of a multiplayer role game staged in a virtual fantasy world, with a twofold objective: first to enhance students' motivation, forcing them to abandon their passive listening role (games and virtual worlds are increasingly being used in education [9] to engage students in active learning processes); and second, to create a propitious atmosphere that may lead to the creation of social and affective bonds among players, which leads to the formation of a community of practice [6].

- Formation of heterogeneous teams, as an implicit assumption of collaborative learning is that students teach and learn from one another. The teams are formed around those students whose learning strategies are effective by using Vermunt's framework for learning styles [44].
- Assignation of functional roles. Roles appear to be most relevant when a group pursues a shared goal requiring a certain level of task division, coordination and integration of individual activities [42], particularly in software engineering projects. On the other hand, roles are used in professional life (in fact, in the software industry they are of crucial importance), so this is also a way of training our future engineers in what it will be their required professional skills.
- Dynamic reconfiguration of teams and rotation of roles. During the course students have the opportunity to belong to different teams and perform different roles in order to promote the acquisition of different abilities, and to confront them with the paradox of having to collaborate sometimes and to compete other times with the same people.

The Nucleo system is used in a blended learning context for teaching computer programming disciplines and it is plugged to the pre-existing Learning Management System (LMS) in order to enhance both tools with complementary functionalities. The project is being developed following an iterative and incremental development process as a way of gradually verifying different sets of hypothesis. In Phase 1 several features concerning the instructional framework were tested and improved through the experimentation carried out in three different computing programming courses, two of which took place at the previously mentioned Engineering Schools at the Complutense University and one at a Technical School. This paper is mainly devoted to presenting the results obtained and analyzing the efficiency of the framework. This analysis considers how the framework influences the drop-out rate and motivation while maintaining the curriculum's learning objectives, whether it really produces a change in the students' attitude toward their own learning process, and whether the students' team work abilities are improved.

The rest of the paper is structured as follows. First, in section 2, some related work is reviewed. Section 3 presents a discussion over general issues concerning the approach taken. In Section 4 the Nucleo framework is described from three different points of view: conceptual, instructional and technological. Then the incremental project development plan is sketched along with the main hypothesis to be proved. In section 6, the three case studies and the experimental data obtained are described. Finally, some conclusions are presented, as well as the future work involved in the following project development phases.

2. Related work

The framework proposed combines several different existing approaches, specifically dPBL (distributed Problem Based Learning), CSCL (Computer Supported Collaborative Learning), and learning in virtual worlds or MUVEs (Multi-User Virtual Environments). There are a number of different applications belonging to these approaches that share some features with the Nucleo system. In this section we present some of the similarities, although as far as we know, Nucleo is a unique combination of all these approaches.

PBL has been implemented several times in virtual settings for different domains, including for subjects related to software programming; just to mention a few that share some features with the Nucleo system: CROCODILE [26] is a multi-interface PBL system in which groups interact through shared virtual rooms. The interaction among the Nucleo community also takes place in a virtual world and the members of the same team interact within the limits of the virtual ship while the whole class interaction occurs on a virtual island (which may be comparable to CROCODILE virtual rooms, but addressing the two different levels of social interaction our system requires); STEP [41] is a dPBL environment that uses specialized tools to facilitate the execution of a set of individual and group tasks in which the resolution process of the proposed problems is divided. In Nucleo there are different tools assigned to different roles and the resolution of the problem is the underlying objective of different activities; Alien Rescue [2], like Nucleo, stages a problem based learning strategy in a fantasy world within the context of a game.

Also, several CSCL applications have studied the impact of team composition on the efficiency of collaborative learning: I-Help [19] identifies the four standard patterns of behaviour in collaborative learning (tutor, student, expert and fellow learner) using taxonomies; GRACILE [5] contains students tasked with the execution of specific learning activities based on the intentions of the group and the its common knowledge; In the work of [27] the complementary knowledge of individuals is taken into account when forming groups; In **¡Error! No se encuentra el origen de la referencia.** and [10] considerations about the students' learning styles are considered in group formation using Felder-Silverman's learning style model; Finally in work [37] a model based on student personality and intelligence is proposed to make up the groups. As is described in section 3.1, Nucleo uses Vermunt's framework both to form heterogeneous but complementary groups and to assign roles to individuals, by means of an adaptation process and a collaborative student model.

Recently, the interest of virtual immersive worlds or MUVEs (Multi-User Virtual Learning Environments) to stage learning is rapidly increasing. A very similar pedagogical approach to the one taken in Nucleo is implemented in the Harvard University River City Project [35], a MUVE designed around topics that are central to biological and epidemiological subject matter. As visitors to River City, students travel back in time, bringing their 21st-century knowledge and technology to address 19th-century problems. River City is a town besieged with health problems, and students work together in small research teams to help the town understand why residents are becoming ill [12]. In the Massachusetts Technological Institute Revolution project, students experience history and the American Revolution by participating in a virtual community set in Williamsburg, VA on the eve of the American Revolution [34]. Other examples of the use of MUVE in education are: Atlantis Quest, a project developed by the University of Indiana for children between 9 and 12 [4], TappeIN [43] for teachers' online professional development, or AquaMoose 3D [3], one of the first applications of MUVEs in education. There is now emerging a whole branch of applications based on the use of the Second Life environment [38] for educational purposes; over 400 universities and 4,500 educators participate in the Second Life Educators List (SLED) [39]. Another related project is Sloodle [40], which uses Moodle services and database through the 3D interface of Second Life. The Nucleo system is also an application that has an underlying LMS that provides tools, data and services.

3. Discussion of the approach

The work presented in this paper is a field study about the application of our Nucleo framework in programming courses. Nucleo is an innovative contribution to the field of virtual collaborative blended learning that can be applied in real and complex domains, with limited budgets, and while considering aspects such as the integration with pre-existing ICT infrastructure. This paper presents the results of three experiments conducted in three different real contexts, involving more than 250 students, who were tracked for a whole semester. Several assessment methods were applied in order to extract the conclusions presented, including two different ways of evaluating the students' knowledge (exams and deliverables), periodical satisfaction questionnaires and teachers' informal perceptions. From this data, our group draws some conclusions about the influence of Nucleo in the students' motivation and their acquisition of technical and soft skills.

It can be argued that an accurate measurement supporting such conclusions would require wider experimentation. For instance, evaluating the efficacy of games in terms of acquiring knowledge is a task that will engage the whole educational community for many years (studies such as [20], or [21]). In fact, this is one of the most polemical and difficult to achieve issues in the game based learning domain (see for example [20] [15] [45]). Although it is certainly beyond the reach of our possibilities to offer a categorical response to some of these key issues, we would like to support, with some discussion, the scope of the study concerns, the relevance of the samples, and the validity of the evaluation methods performed.

At least in the Spanish higher education context, measurements of students' knowledge gain are done through the results obtained from exams. Even though the effectiveness of this method may be debatable, it can be considered the most common mechanism to assess knowledge, and thus we have adopted it to evaluate this gain. As section 6 shows, more students that followed our experimental framework passed the final exam, and they obtained better marks on average than their colleagues who followed the traditional method.

Another controversial issue about game based learning approaches is how to categorize them: is the game a carrier of the learning or is it a place where the learning takes place? According to several authors [13][11] the main difference between "edutainment" and game based learning is that, while edutainment games follow a format in which players either practice repetitive skills or rehearse memorized facts, educational video games "require strategizing, hypothesis testing, or problem-solving, usually with higher order thinking rather than rote memorization or simple comprehension. Characteristics of such games include a system of rewards and goals which motivate players, a narrative context which situates activity and establishes rules of engagement, learning content that is relevant to the narrative plot, and interactive cues that prompt learning and provide feedback." ([13] pp: 21). We certainly think that our approach presents all the characteristics necessary to be included in the second typology. Several features of the system presented in this paper support this claim:

1. The game is set around challenging problems that require developing higher order thinking skills rather than memorizing concepts.

2. The problems are conceived in such a way that require the development of a team strategy. For this strategy, the team members are provided with explicit guidelines that depend on the role they are assigned.

3. The whole framework is staged in an immersive atmosphere and follows a fantasy narrative that situates the learning activity, in order to promote the development of social bonds that may lead to an improvement of the collaboration process. Several recent research works have defended the effectiveness of this strategy. Moreover, the educational community is beginning to consider MUVE's potential to create immersion atmosphere as tools worth testing, to the extent that the use of this sort of environments has multiplied in the last three years (see for example [15][16][21]).

4. The learning content is relevant to the narrative plot. The students are provided with hints and guidelines to obtain the content that is relevant for the development of the missions. Therefore, knowledge is acquired in situation and is applied in context.

5. The game includes a system of rewards and goals that motivate the players. In the Nucleo system, students' avatars get different distinctive physical features linked to their intellectual achievements in order to promote motivation by social recognition. The atmosphere of competition is enhanced by publishing individual and team rankings.

6. Interactive cues and feedback are constantly provided throughout the development of the missions. The tracking system that the interconnection between the game and the LMS supports, helps in providing students with constant feedback and clues to solve the game challenges.

4. The Nucleo project

4.1 Nucleo conceptual framework

PBL was proposed by Neufeld and Barrows [29] in the 1970s for university courses and since then, has gradually spread to many other knowledge domains [8]. According to Barrows [7], PBL can be explained as “the learning that results from the process of working towards the understanding or resolution of a problem.” This approach is usually case based, small group, self-directed learning, in which a group is given a specific problem to solve. Instead of an expert, the group has a tutorial leader or facilitator who shares information and guides the group through the learning process. In sum, PBL learning is a process of building on prior knowledge, problem solving, the use of critical thinking approaches and reflection **[25].** This self-directed, collective approach is a very different way to teach compared to traditional lecture-based approaches.

Educational literature has shown the benefits of using PBL and other approaches that promote active collaborative learning to improve students’ thinking skills [23]. It has been demonstrated that it leads to deeper levels of learning, critical thinking, shared understanding, and long-term retention of the learning material. Furthermore, collaborative learning also provides opportunities for developing social and communication skills, acquiring positive attitudes towards co-members and learning material, and building social relationships and group cohesion [22]. Despite all these advantages, it is not easy to implement a PBL approach effectively, especially in virtual learning environments or even in blended learning environments, where the richness of personal interaction among members is more limited than in a face to face setting. Two of the reasons are:

- It relies very heavily on group dynamics for its success. Group cooperation and cohesiveness have been identified as key factors [6].
- Social interaction appears to be the key to collaboration [17]. Nevertheless, just placing students into groups and giving support to some kind of communication among them does not guarantee the emergence of the social interactions that lead to effective collaboration.

In Nucleo, as is done in most PBL environments, learning is structured around the resolution of "real world", complex and ill-structured problems, which have to be solved by collaboration among the members of small teams. The main differences between Nucleo and a classical PBL approach are the following:

The first important difference is that in Nucleo the "real world" is a fantasy world. Therefore the practical programming cases around which the learning is structured follow the narrative of the game’s back-story. The game metaphor takes the student to Nucleo, an artificial universe in peril of extinction. The survival of this entire world lays on the students' shoulders. In order to confront the menace, they are trained in the weapons of knowledge. The training simulates a real attack from the enemy (in the form of a mission), which the aspirants must repel by clustering in small combat units (usually composed of 3 or 4 members). Students’ avatars play the role of these champions and their type of participation, duties, and skills in the crew are conditioned by their role. The different crews compete to obtain the best solution. At the end of the training period, only the best will reach the grade of Paladins. We have found that presenting this scenario at the beginning of the course certainly intrigues the students, thus inducing a change in their attitude (they turn from passive listeners to active warriors within the game atmosphere). Also the structure of the game provides the motivation and the urge to solve the problems and the characterization makes the player's role in the narrative believable, therefore facilitating the learner's immersion [36]. The idea of embedding a PBL approach in a game to create a immersive fantasy atmosphere has also been used in several learning applications, such as Alien Rescue [2], River City [35] or Atlantis Quest [4].

The second important difference is that Nucleo seeks to increase the effectiveness of collaboration by improving group dynamics. The system includes an adaptation module in order to form heterogeneous teams and assign internal roles. Similar approaches have been taken in [1], [10] and [37] but using different conceptual models to group the students. In addition, in Nucleo, teams and roles are reassigned dynamically after every mission, in order to enrich the social interaction and give the student the opportunity to handle different responsibilities and duties, while they get different perspectives for solving the problem.

The rationale for organizing teams according to heterogeneity in ability is defended in several research works [30]. The classification of students in Nucleo is performed by means of Vermunt's Inventory of Learning Styles [44], a 100 question inventory which students must complete before starting the course. Vermunt's framework was specially conceived for university students and is really more of a classification of students according to the strategies they usually employ when learning, rather than a learning style classification as it is usually understood. Vermunt classifies students into four types depending on the attitudes they adopt in five different areas of learning. These four learning styles are: meaning-directed (MD), application-directed (AD), reproduction-directed (RD), and undirected (U). Those students who are able to self-regulate their learning processes would benefit from a loose teacher strategy. This is usually highly correlated with MD and AD patterns. On the other hand, students without this ability would need stronger teacher control and guidance. This corresponds to the RD and the U patterns. Therefore, we try to group the more autonomous students together with those who require stronger leadership throughout the learning process, implicitly assuming that this way of grouping will be of advantage to every member of the team and that the students will learn from and teach one another.

We also have correlated Vermunt's resulting profiles with the three Nucleo roles and embedded them in the metaphor atmosphere (i.e. *Captain of the Crew*, *Knowledge Integrator* and *Chief/Head of communications*). The *Captain* is in charge of project planning and monitoring project progress, the *Knowledge Integrator* is in charge of supervising and making sure that all team members acquire the knowledge required, and the *Chief/Head of Communications* is in charge of managing the communication between team members and the tutors as well as managing conflicts among team members. The design of the specific responsibilities linked to these roles makes them inter-dependent as a means to encourage collaboration in pursuing a common objective, thus fostering group cohesion and responsibility. In this sense, roles appear to be most relevant when a group pursues a shared goal requiring a certain level of task division, coordination and integration of individual activities [42]. On the other hand, roles are used in professional life (in the software industry they are of key importance), so this is also a way of training our future engineers in what it will be their required professional skills.

4.2 Nucleo's instructional approach

As has been said in the previous subsection, in Nucleo the learning process is structured according to the classical PBL schema. Knowledge is gained through collaboration procedures designed to solve ill-structured, open-ended problems. The difference is that problems in Nucleo are embedded in the game narrative (they are called missions) and solving them is part of the game.

Thus, within the game context, a mission (which is really a complex practical case immersed in the game narrative) is an event in the competition to become a Paladin. It simulates a real risk situation which Paladins must solve in the fight against the enemy, and teams compete among themselves to obtain the best solution.

The learning process follows a cyclical structure:

- A course is made up of several missions determined by the tutor. Each mission represents a learning objective included in the curriculum.
- Each mission is composed of several activities. Usually, the end of each activity is marked by the production of a certain result (usually in the form of a conceptual model or schema for the solution, a report, a document, etc.) which has to be delivered to the "Sages" for evaluation. The resulting products of one activity often work as the starting point for the next one to be performed.
- Before starting a new mission, the teams are reviewed and, if necessary, re-configured. The same process is applied to the roles the individuals hold within the team. Re-configuration of the teams and re-assignation of the roles depend on the results obtained in the previous missions by means of an adaptation cycle and a user modelling process. Belonging to different teams and performing different roles enriches social interaction and gives the students the opportunity to see the solution from different perspectives.
- At the end of each mission, the solutions are evaluated. In the evaluation not only the technical quality of the solution is considered, but also how the members of a team have perceived the performance of each individual regarding the fulfilment of his/her duties within the team. In order to increase the competitive atmosphere, individual and group rankings are published at the end of every mission. Also, students' avatars are rewarded with physical distinctions linked to their intellectual achievements, as social recognition seems to be a very powerful motivation in multi player environments [6].

The Nucleo system is used in combination with on-site classes (blended learning approach) as a way to manage distant interaction and collaboration among team members. The process starts from the delivery of each new mission. The students receive a notification about the new mission, their new team and their new role. The environment provides for two virtual scenarios to address the two levels of social interaction; the interaction among the whole class (including the tutor) and intra-group interaction. They both provide social tools (forums, notice boards, chats, file sharing, blogs and wikis) in order to facilitate collaboration, and the different roles and teams have restricted access to areas and services. Figure 1 shows the ship where intra-group interaction takes place, and the whole island.

Figure 1. Ship and island of Nucleo.

4.3 Nucleo architecture

The Nucleo system is designed as the fusion of three main elements: a Learning Management System (LMS), a Multi-User Virtual Environment (MUVE) and an adaptation module (see Figure 2).

In the last few years, in spite of the high costs of deploying e-learning systems, LMSes have been widely adopted as virtual education tools at universities and training centres. This is probably due to the fact that LMSes offer a series of significant advantages when managing an ample training context: they are flexible enough to allow the implementation of different teaching strategies; they support a very rich level of interaction among tutors, students and student teams; they allow the reuse of learning resources in the shape of fragments (i.e., learning objects) or even complete courses; they implement administrative tools able to communicate with other applications (i.e. ERP), and the path to interoperability is open. But even though most LMS platforms for higher education now offer multiple functionalities for implementing different educational approaches –like forums, chats, file sharing, etcetera–, it is quite frequent to use them exclusively as repositories of contents in which the subject matter is basically static (MSword files or PowerPoint slides) [24], their user interfaces are not very appealing and they offer very few opportunities for adaptation. On the other hand, it is also true that learning environments using MUVE technology, in which social interaction modes and contents are very rich, seem to be very restrictive as repositories of contents and almost completely inoperative when managing some typical processes in an integral management of learning, such as maintaining students' historical records or performing administrative tasks. The Nucleo system uses Moodle services, tools and contents, covered by a virtual reality "skin", with direct access to its database. Figure 3 shows how the forum and learning activities stored and managed through Moodle are displayed and used through the virtual learning space. This means that the data generated during the learning process is stored directly in the centralized system, and is also managed in a centralized way, with all its subsequent benefits. We are using the Multiverse platform [28] facilities to generate the MUVE.

The adaptation model in Nucleo aims at improving teams' efficiency, which would produce a comprehensive improvement in the learning process and a lighter teaching load for the tutor. With this objective in mind, we are using two combined strategies: the formation of heterogeneous teams and the assignment of functional roles. The groups and internal roles are determined by means of the student profile.

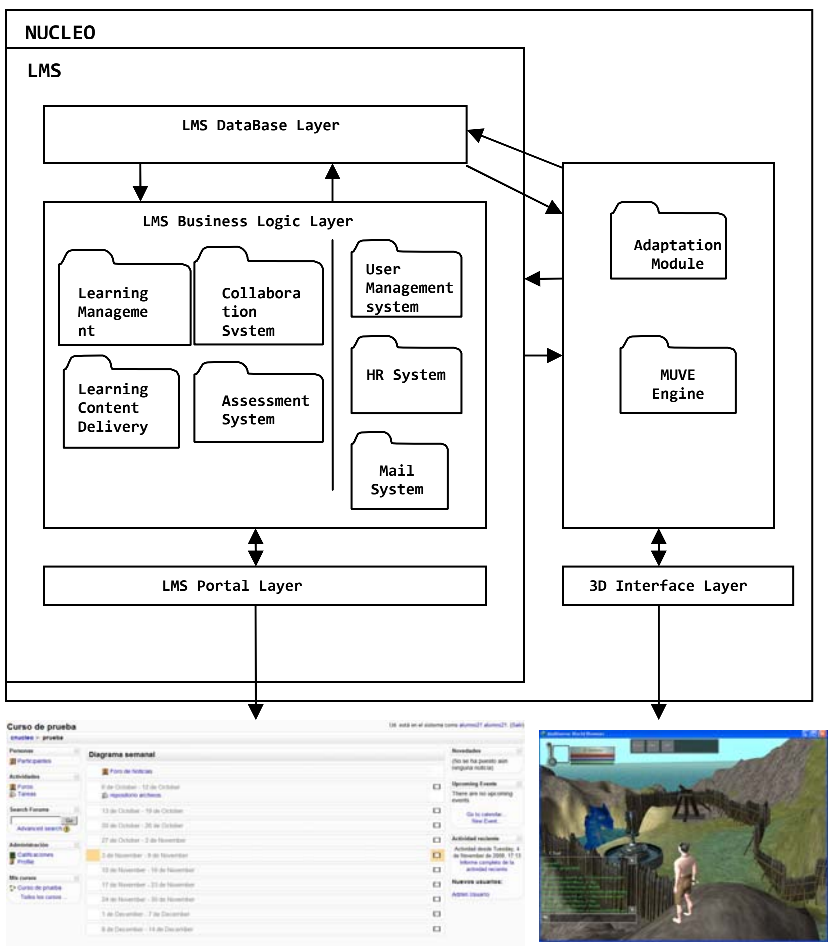

Figure 2. Reference architecture for the Nucleo system.

In the construction of the student model the result of the Individual Learning Style (ILS) [44] is considered: a questionnaire filled out by the student at the beginning of the course, which is distributed and evaluated through the LMS, and is maintained through a user-modelling process that takes into account three different data inputs: the results obtained on the missions, the peer-evaluation the students get and how often the students make use of the system tools. The adaptation cycle follows the same cycle the learning strategy does: in every mission, teams are reconfigured, and roles are reassigned.

Figure 3. Moodle forum and event panel displayed in the Nucleo environment.

5. The project development plan

To increase the cost efficiency and optimize the educational value of our system, the development project follows an iterative incremental software process. We have designed four different phases in order to gradually prove different sets of hypotheses. Therefore, every new phase relies on the proven set of hypotheses from the previous one:

- Phase 0: Documentation and research in order to define the pedagogical strategy and the instructional design of the system. This was a purely documental phase with zero investment in software development. It was aimed at researching the different pedagogical and technological strategies for e-learning systems in order to determine the instructional design of our system as it is presented in sections 3.1 and 3.2. In this phase the basic pedagogical hypotheses were also outlined as well as the back-story and the game narrative.
- Phase 1: Proof of concept. This phase required minimal investment in software development as it made use of free collaborative software to support the learning environment (we used Google social tools). It was aimed at verifying the main hypothesis on which the system relies. The experimental results obtained in this phase are presented in section 5.
- Phase 2: Proof of the effectiveness of the virtual scenario. The main objective of this phase is to verify the impact of several aspects of the 3D virtual scenario and the avatars on students' motivation. In order to find a lower budget compatible with our defined objectives, we decided to use a free game engine with plenty of graphical resources (Multiverse [28]) and to use these resources in prototype construction.
- Phase 3. Development of the beta system. In this phase a complete system will be developed and distributed for beta testing in different learning contexts. Due to its great requisites in development effort and investment, we want to assure and refine certain key functionality features in the previous phases.

6. Case studies

With the aim of obtaining information about the effectiveness of the instructional framework we designed an experiment that involves three different case studies concerning computer programming subjects in engineering education in three different educational contexts: two at the university level and one at professional training. The two at the university level used the programming language C++ and involved over 60 and 110 students respectively; the one at professional training used Java and had just 15 students. All of them took part in a four-month learning period, although the students in two of the experiences had a previous period of four months with a standard learning approach.

In this section we describe the three different case studies in the following terms. The first sub-section describes the common goals for all the cases. Afterwards, the case studies themselves are described separately, together with the results obtained in each study.

6.1 Goals of the experiment

The main goal of the experiment is to gather information about the effectiveness of the instructional framework developed in the following terms:

1. In order to evaluate the effect of the Nucleo approach on student's attitude towards study, we measure different issues related to student participation.
2. In order to evaluate Nucleo efficiency, we compare the marks obtained in the same exams by students who have participated in the experiment with the ones obtained by students who followed a classical instructional approach.
3. To measure the impact of the proposed learning framework on student motivation and drop-out rates, we compare the drop-out rates for the classical teaching approach with those for Nucleo.
4. In order to verify the accuracy of Vermunt's model for assigning the functional roles, we measure students' satisfaction with their peers, in terms of how their teammates perceive the fulfillment of their responsibilities. Team formation relies primarily on the assumption that there is a student with good learning practices who plays the role of the captain of the crew, and who is responsible for distributing and planning the teamwork. This figure is of key importance for the team's performance, so we measure whether Vermunt's ILS really identifies strong students accurately by

verifying if MD and AD students get better marks than the other profiles.

5. We evaluate the perception that students have towards the effect of the framework on the development of soft and teamwork skills by means of a questionnaire the students are asked to fill out.

6.2 First case study: teaching programming fundamentals at the Electrical Engineering School

The course "Programming Fundamentals" (PF) is offered in the first semester as an optional subject during the second cycle in the Electrical Engineering School at the Complutense University. It is aimed at teaching some computer programming basics (i.e. algorithms, program design, coding) to electrical engineers, using C++ as the programming language.

The traditional approach is structured in three hours per week of non-practical lecture sessions in a classroom where no computers are available for the students' use. Nevertheless, the students may optionally take another complementary subject called "Programming Laboratory" which consists exclusively of practical work with a computer on a two hour per week basis. At the beginning of the course, the lecturers highly recommend that the students take both courses in parallel in order to complement the theoretical and the practical work. Although this is a non-compulsory recommendation, it is usually followed by around 60% of the students.

The evaluation of the students in the traditional approach compiles the results obtained from the final exam and class work. The class work takes into account the participation in common discussions and the marks obtained from solving several practical cases proposed throughout the course. These practical cases are usually solved individually. The exam makes up 60% of the final grade, while all the rest of the work makes up 40%.

Academic year	Students enrolled	Sitting the exam	Course passed	Average mark (over 10)
2005-06	115	43	30	5,1
2006-07	110	33	24	5,0

Table I. Statistics about the last two years of PF courses.

Table I shows the statistical results obtained the past two years in the PF course. In 2005-2006 only 26% of the students enrolled passed the course, and in 2006-2007 this rate decreased to 21%. The drop-out rates were also very high: 62% of the students abandoned in 2005-2006 and 70% did the same in 2006-2007. The teacher, the syllabus and the examination method were the same both years.

In 2007-2008, the Nucleo experiment was performed in the PF course with 60 students. The new learning scenario was presented at the beginning of the course. The students were initially puzzled and reluctant to participate even if enrollment was only for volunteer students. As a result of this, the class was divided into two groups: around 63% chose the traditional learning approach while 37% participated in the Nucleo experiment. The Nucleo students were told that the results of the work done in the experiment would mean a 40% share of their final mark, the same as the class work in the traditional group. This data is reflected in the first column of Table II.

Group	Enrolled students	Sitting the exam	Drop-out over enrolled
Traditional	38	13	65,79%
Nucleo	22	20	9,09%

Table II. Statistics of students getting to the final exam in the 2007-08 PF course.

The Nucleo students had only a two hour on-site session every two weeks and they were excused from attending the theoretical lectures. These sessions were aimed at coordinating the different groups and having a shared discussion about the requisites of the missions proposed. Contact among the members of the teams mainly happened through a common virtual space provided with social software and collaborative tools (such as document share facilities, post, discussion forums and blogs). We used the free software facilities provided by Google groups for this purpose. All the Nucleo participants shared a common space that simulated the Nucleo Training Academy. All the public information was published in this space and each team had its own private group where the teacher, i.e. the Sage, was invited. The goal of these coordination spaces was to boost the feeling of community and therefore to enhance collaboration within the groups, between the groups and with the teacher.

The Nucleo teams were organized according to the rules expounded in section 3.2.2 and the team members were assigned a role with specific responsibilities (see section 3.2.3). All the teams had a leader (a member of the Evian tribe) with a high MD punctuation according to Vermunt's ILS.

The experimental course was made up of four missions. All the missions were highly demanding for novice programmers, considering that no theoretical lectures were given to group members. At the end of every mission, the groups were ranked according to the quality of the solutions delivered and so were the individuals in order to promote motivation through competition and social recognition. The individual rates were calculated by taking into account two inputs: the score obtained by the team and an individual score that every member obtained from their teammates' evaluation. In this peer-evaluation the responsibilities for the fulfillment of every role were assessed.

Table II and Table III reflect the comparative results obtained by students following both approaches.

Group	Pass exam	Average mark (over 10)	% pass over enrolled
Traditional	7	4,67	18,42%
Nucleo	13	5,73	59,09%

Table III. Statistics of students passing the 2007-08 of PF course.

As explained in sections 3.2.2 and 3.2.3, students were categorized into three different tribes (Evians, Ruks and Exters) depending on the results they obtained from the initial learning style questionnaire. Every member had to perform different duties inside the team depending on the tribe he was assigned to. At the end of each mission, students were asked to evaluate their teammates, taking into account several issues concerning the fulfillment of their assigned duties. Table IV reflects the marks obtained in the final exam by the Nucleo participants according to the tribe they belonged to and the average punctuation obtained by each tribe according to the perception of their teammates.

	Evians	Ruks	Exters
Average exam mark	6,66	5,36	5,53
Peer-evaluation	9,375	7,954	8,24

Table IV. Final exam marks (out of 10) and peer-evaluation results obtained by the different tribes (out of 10).

6.3 Second case study: teaching C++ programming at the Computer Science Engineering Faculty

The annual course "Laboratory of Programming II" (LP2) is offered in the second year of the three-year Computer Science Engineering degree. The laboratory is held in conjunction with two more theoretically-oriented courses: "Object-Oriented Programming" and "Data Structures". Thus, one of its main goals is to provide students with the practical knowledge related to these courses. In addition, the course is also oriented to providing a way of improving students' capabilities in terms of carrying out a complete development project in small teams.

The traditional approach adopted in the LP2 course includes a 1-hour lecture session per week in the classroom and 2 hours per week of supervised work in the laboratory. The work in the laboratory consists of several practical programming cases that apply the theoretical lessons. It is estimated that in order to pass the course, students need at least five additional hours per week of work on their own.

The evaluation of students in the traditional approach includes a final exam and class exercises. Laboratory work is mandatory and makes up an additional 10% over their score in the final grade.

Over the last years, this course has suffered from worrying drop-out rates and very low marks. Table V summarizes some statistics about recent courses in LP2. The data cast average figures of 77,33% for the drop-out rate and 5,02 on a scale of 0-10 for the mark of the students that **attended** the exam. It needs to be remarked that group B is the one preferred by students who combine their studies with a part-time job and by those repeating the course. For this reason, its drop-out rate is usually higher (78,83%) and the marks lower than in the corresponding A group.

Academic year	Enrolled students	Sitting the exam	Passing the course	Average mark
2005-06 A	93	22	6	4,53
2005-06 B	57	13	8	5,02
2006-07 A	106	24	14	5,28
2006-07 B	65	14	7	4,90

Table V. Statistics about traditional LP2 courses.

Given these facts, the lecturers of the 2007-08 course decided to try the Nucleo approach in both groups, offering the students an optional participation in the experiment.

While in the traditional approach, students organized themselves into groups of two members where they chose their team partner and organized their own work, the Nucleo students were organized in groups of three students where roles and responsibilities were assigned by the lecturers.

As in the previous case studies, the Nucleo groups shared a central coordination space (i.e. a Google group) with the lecturer, as well as another space dedicated to their internal coordination (where the Sages where also included as guest members). No group spaces were provided for traditional students (only the usual support of the Learning Management System provided by the University for all students and subjects).

Five practical assignments were proposed to the students during the course (both to the traditional and to the Nucleo students). At the end of each of them, the students had to deliver the software code and its related documentation in order to get an evaluation from the lecturers. For the traditional students the mark obtained by the group was the same mark assigned to the individuals. The Nucleo individual mark was scaled by a factor that took into account the way their teammates evaluated the fulfillment of their duties in the team as performed in the previous case study. The rankings obtained by the teams and the individuals were published in the common space.

With this setup, Table VII summarizes the results of the experiment at the end of the course. This table makes a distinction regarding the real number of students that initially attended classes versus the number of students that were officially enrolled in the course (see Table VI). It has been observed that some students enroll in the course but never attend classes at all. This can be due to several reasons: they enroll to complete bureaucratic requirements, find a job, or prefer to go directly to the final exam in September. These figures correspond to the "Initial drop-out". The "Traditional students" and "Nucleo students" consider only those students that actually enroll in practice groups.

Group	Enrolled students	Traditional students	NUCLEO students	Initial drop-out
A	101	42	27	31,68%
B	74	26	17	41,89%

Table VI. Statistics about enrollment in the 2007-08 course of LP2. Students attending the course are in the columns "Traditional students" and "Nucleo students".

Group	Drop-out over enrolled	Drop-out over attending	Sitting the exam	Passing the exam	Average mark	% pass over attending
A traditional	77,03%	59,53%	17	13	5,26	30,95%
A Nucleo	14,81%	14,81%	23	16	5,46	59,26%
B traditional	92,73%	85,00%	4	4	6,13	15,38%
B Nucleo	10,52%	0,00%	17	17	4,96	52,94%

Table VII. Statistics of students passing the 2007-08 course of LP2.

	Evians	Ruks	Exters
average exam mark A	5,87	4,9	5,7
average exam mark B	5,3	4,3	5,5
peer-evaluation A	6,9	7,4	7,1
peer-evaluation B	9,5	8,2	7,68

Table VIII. Final exam marks (over 10) and peer-evaluation results obtained by the different tribes (out of 10).

Table VIII reflects the average figures obtained by the three different tribes, both in the final exam and in peer-evaluation.

To finish, we would like to make some remarks concerning this case study. The first one is that we found a strong resistance from the students to change to the new method because they perceived that Nucleo would mean more personal effort and would make the subject more difficult to pass. Our first intention was to apply the Nucleo method without offering the possibility of following the traditional approach. Nevertheless, we abandoned this idea after receiving plenty of angry complaints, because we found that true commitment was significant for the success of the experiment. The second issue worth mentioning is the motivation coming from social recognition. Although in the traditional approach the best practices got extra points, students were usually not motivated to work for them. On the contrary, the Nucleo students competed among themselves just to be the best in the course, which seemed to have a fairly positive influence on their motivation.

6.4 Third case study: teaching Java programming at a Technical School

The third case study was carried out at an official Spanish Technical School, during the second year of Network and System Administration. During this second year the students, who had already acquired some programming fundamentals in C during their first year, learn Java language for system programming and study *low level* aspects such as threads, monitors (semaphores) and sockets.

Technical School (TS) education in Spain has a different objective from university studies and students choosing this course of study show remarkable differences from those attending university as well. TS is conceived as more practical training, and the number of students per class is significantly lower than at the University. The average student is not usually conscious of the importance of active working. In fact, the students who typically enroll in TS have less initiative and a no-effort mentality. They assume that instructors should teach them everything without having to put in much of an effort themselves, and they usually feel frustrated if this does not happen. This means that students demand a lot of help from the teacher while they are solving their class assignments, and a lot of time is wasted while they are waiting for attention.

Students in TS have more supervised classes per week than at the University. Specifically, the subject where the experiment was carried out consists of 8 hours per week and students usually do not work too much at home. Therefore, only 2 or 3 of those hours are used for theoretical lectures. The rest of the time is used by students to solve the proposed tasks and exercises in class. Student interest, assignment solutions and work attitude make up 30 or 40% of the final mark, which it is obtained by means of sitting a written exam.

Since practical aspects of the subjects are so important in TS, and require so many class hours, student motivation is crucial. It should be noted that the students have access to their computers throughout the class, so during the theoretical sessions, the teachers must fight against more attractive distractions such as web browsing or videogames. Only when they have a clear interest in the subject will they be able to resist these *temptations*; otherwise the teacher's efforts are useless and the student will pay no attention to the explanations.

The third case study was carried out in this context, with only 15 students. During the first 4 months, a traditional class approach was used. Two or three hours were used for theoretical aspects of programming, and a set of exercises were proposed to be solved in class. Although no restrictions about cooperation or external help where imposed, students tended to get lost rather quickly and more often than not they just waited to copy down the teacher's solution, playing an absolutely passive role in their own education. Only two students (13%) demonstrated enough interest to work out the assignments. The others argued that they were too difficult and did not even try to solve them. It is important to notice that a great deal of effort was made to make exercises motivating by including simple games to interest them, such as tic-tac-toe, four in a line, Tetris and Sudoku. At the end of this 4-month period, six students did not attend classes regularly (40%), and sometimes only five students (33%) went to class. Finally, only four students passed the exam after this period, although the exam exercises were very similar to the assignments proposed and solved in class.

The next four months only the Nucleo approach was used, giving the students no other alternative method. All students were categorised, and four teams were created, one with three members, and the others with four. The decision for this irregular distribution was the previous knowledge of attendance: to avoid faulty groups formed by absent students, those who did not usually go to class were assigned to the groups of 4 members. As in the previous case studies, the social software offered by Google was used as a way to promote cooperation and facilitate communication. Nevertheless, being a full-time classroom course, these tools were used as a document repository (including the log files required) instead of a way of keeping the students in contact.

The classroom dynamics experienced an incredibly positive change once the Nucleo approach started. The previous passive attitude became active participation in every mission. *All* teams delivered the exercises on time. It is worth pointing out that, during the first four months, the students usually complained that exercises were too complex to be solved all alone. However, in the PBL setting proposed with Nucleo, *no theoretical classes* were given and, even so, the students searched for information and reached successful solutions for missions that were quite complex. This shows that the students were *more involved* in their own learning and Nucleo had a positive effect on their participation.

Regarding class attendance, the week after Nucleo started, *every* student attended class, which was something that had never happened in the previous semester. Although later on some of the students started to miss class again, statistically the high rates that were common during the first four months were never reached again (we averaged 12 students per session during the second semester). This fact may be explained due to the *social pressure* of belonging to a *team that needs you*. In fact, a student who finally decided not to take the examination, continued going to class to take advantage for the next year. Given these facts, it seems clear that the Nucleo set improved the students' involvement, attitude and motivation.

Unfortunately, motivation and participation do not always mean better scores. Only five students (one more than in the first four month period) passed the first exam, although in the second exam the number increased to nine. Once again, the difficulty of the two exams was quite similar to the missions proposed.

Concerning the accuracy of Vermunt’s classification, this case study is quite different from the previous ones because the number of students was small, and we already had had contact with all of them for four months, eight hours per week. Vermunt’s test is supposed to be useful as a way to pseudo-automatically identify functional roles, but this is not needed when you already know the people to be classified. Despite this, we also used Vermunt’s classification in this case study in order to confirm our preconceived ideas, showing that in this case it was quite accurate (the MD students were in fact the strongest students with a correlation of 80%).

7. Discussion and future work

The first two case studies were conducted in similar conditions and contexts. The target groups were university-level engineering students, the courses had had similar drop-out rates and low marks over the last two years, and the experiment was conducted in a similar way, with some of the students following a traditional teaching approach and some voluntarily participating in the Nucleo approach.

The conclusions that can be drawn from the experiment are also similar. The statistical data shows with a significant difference level of .95, that the drop-out levels are highly reduced among the participants in both experiments while the rate of students passing the exam rises. In the first case, 65,8% drop-outs in the traditional approach versus 9,1% obtained in the Nucleo approach, and 72,3% versus 7,04% for the second case (the average rate of the two groups A and B analyzed). Also, in the first case, 59,1% of the students who followed the Nucleo approach passed the exam while only 18,4% of the traditional set did. Similarly in the second case these rates are 56,1% vs 23,2%. In terms of the average marks obtained by the students who followed the two different approaches, the statistical data does not show a clear difference. Although the marks obtained in the Nucleo set were slightly higher in both case studies, there is no significant statistical difference because dispersion levels are also very high. No clear conclusions can be drawn either, in terms of the accuracy of Vermunt’s classification. Although in the first case study the Evian tribe (MD and AD students) obtained better marks and better peer punctuation, the difference was not significant. In the second case study, the data is even more confusing as none of the tribes obtain clearly higher values than the others. Nevertheless, the students appear to be happy with the roles assigned and the clear distribution of responsibilities, and found it was positive for completing the work.

The third case study presented was conducted in a very different context, with pretty different targeted students. Although the experiment was less statistically rigorous, it can be considered more "personal". The group was less numerous (only 15 students participated) but the teacher had considerable personal contact with them. In this third case the participation in the Nucleo set was compulsory during the second semester while everybody followed the traditional teaching approach during the first semester. In this case, the teacher's perception matches up with the data of the two previous ones: the drop-out rates (measured in terms of class attendance) were significantly reduced and students’ interest increased (less time wasted on videogaming and browsing, and everybody turning in assignments on time). Nevertheless, only a slight difference in the rate of students passing the exam was observed (26,7% vs 33,3%). The teacher also observed that Vermunt’s classification was quite accurate regarding his previous conception of the students' attitude (he had dealt with the 15 students during a whole semester on an 8-hour per week basis).

Concerning the evaluation of the improvement in the acquisition of soft and teamwork skills, according to the registered data, virtual collaborative learning tools are used three times more frequently, if we compare the game setting to the use of the previous LMS setting without any sort of gaming strategy behind it. Of course, this only means that students interacted more (which is a way of developing social bonds and practicing communication skills) but does not necessarily imply the development of soft skills. Nevertheless in all three cases a final questionnaire was filled in by the Nucleo students at the end of the experiment. This questionnaire sought to gauge how the experience was perceived by the students compared with a traditional learning scenario. 95% of the students found the experience "very positive", remarking that it helped them to acquire a more active attitude towards their study and that it had helped them to practice teamwork abilities and communication skills. In addition, most of them found this was a more motivating way of learning and, even though it had involved a considerably higher amount of effort than a traditional approach, they would repeat the experience.

Finally, there are also some drawbacks in applying this method that are worth being mentioned. In the first case study we observed that many of the students focused mainly on learning the concepts that they were interested in, so their final acquisition of knowledge was quite irregular. They were experts at managing some advanced programming features, while they had not acquired some of the basic concepts. We tried to correct this problem in the next two experiments (cases 2 and 3) by giving basic theoretical lectures once the mission was solved, thus obtaining better results.

Although cases 1 and 2 have shown the positive influence of the method on drop-out rates and on the number of students passing the final exam, it can be argued that there are many factors that may have influenced it besides the pedagogical approach. A key fact can be that the experiment was conducted with volunteer students, and thus it may have attracted the more motivated students. Also, the teacher's attitude might have been different and more enthusiastic with the new experiment setting than with the traditional scenario. Nevertheless, we can now provide new evidence from recent experiments that would contradict these hypotheses. Concerning the influence of the experiment attracting the already more motivated students, this year we are performing a new experiment for the subject “Fundamentals of Programming”. First of all, although the general tendency in the last five years has been a gradual decrease in the enrolment (and neither the teacher nor the course curriculum has changed during this period), this year (2008-09) enrolment has increased by 35%. Also, the Nucleo methodology has become compulsory and thus followed by *all* the students in the class. So far and with more than 75% of the course completed, only 9% of the students have withdrawn, while the average rates in previous years for the same period reached an average of 45%. Concerning the issue that the teacher's attitude may have been different in the two learning settings presented (traditional and Nucleo), it is more difficult to offer any objective measurements. Even so, the same teacher has been in charge of the FP course during the last six years and she has permanently been worried and concerned about improving high drop-out rates. Some other motivation techniques such as more practical works or plain PBL were applied in previous years without any significant results. In the last two years, she has only been involved in the Nucleo experiment. Over these six years, she has also participated in several official Spanish Innovative Programs for teaching and learning, and she has always obtained excellent results in the students' annual evaluation.

It is very difficult to evaluate the pedagogical effectiveness of videogames and immersive worlds, even more if complex topics with real students in a pre-existing setting are considered (as happens in the Nucleo context) Nevertheless, as previously mentioned, this is a generally open issue for this type of systems (as confirmed by the extensive study on the research literature on the subject that can be found in [20]).

In addition to the previous argument, the third experiment showed a similar tendency, even though in this case, participation in the experiment was compulsory. Finally, we have also observed that the internal organization of groups into students who are not close mates makes for a neat division of responsibilities but also leads to a certain lack of communication. Some students focus only on their assigned issues and largely ignore what their mates have done. This also shows that in some cases the student with the role of knowledge integrator did not really fulfill the duties of the role.

To sum up, all three case studies where the Nucleo approach was applied show very promising results in terms of the objectives pursued, particularly in increasing the students' motivation, shifting their attitude towards a more active role and improving communication skills and teamwork capabilities. Nevertheless, the experiments also drew the conclusion that several issues should be improved and reviewed. Firstly, Vermunt's learning style classification framework for assigning functional roles in a team does not show enough reliability. Therefore in the next phase, we are considering including individual personality traits as proposed in [37] to improve the accuracy of the model. In addition, the students pointed out that they would rather choose the physical appearance of their avatars instead of being assigned to a fixed tribe. In the current experiments the tribe and the physical configuration were linked to the functional roles assigned by means of Vermunt’s categorization. For the next implementation, this will also be taken into account since students will be able to select their own avatars even though their responsibilities will still be assigned by the system. This is expected to improve the students’ immersion in the metaphor. Finally, concerning the functional roles, we will try to fix the weaknesses in the KI role through the inclusion of certain activities designed to monitor the fulfillment of this role's responsibilities.

In the next semester the 3D world will be introduced to frame the instructional design and new experiments will be conducted in the same contexts, with the improvements that we have detected in Phase 1. Also, our intention is to determine whether it is worth it to fully develop a 3D scenario that contributes to the learning immersion, thus improving the motivation for social recognition through the avatar's physical change.

Summarizing, even though we are aware of the limited scope of our experiment (it cannot be compared in budget or in the number of people that may constitute the target audience, with the Virtual Leader project [45], for example) we are convinced, based on the arguments provided, that it has brought a small light to a very difficult problem we have been facing in our software engineering schools for the last years. After testing our framework in three different real context situations, we have obtained promising results that induce us to think that this method may bring an overall benefit for our students' education, which has led us to keep on researching and enhancing our system. Like Galileo, we say, "Eppur si muove" , and yet it moves...

8. Acknowledgments

We must thank our collaborators Albert Meco and Daniel Leiva from Totemcat, and especially to Abraham López Guerrero for his support and imagination. The Spanish Committee of Science and Technology (projects TSI-020301-2008-19 and TIN2007-68125-C02-01) has partially supported this work, as well as the Complutense University of Madrid (research group 921340) and by the EU Alfa project CID (II-0511-A).

9. References

[1] E. Alfonseca, R. M. Carro, E. Martín, A. Ortigosa, and P. Paredes, The impact of learning styles on student grouping for collaborative learning: a case study. J. User Modeling and User-Adapted Interaction 16(3-4), 377-401 (2006).

[2] Alien Rescue: A Problem Based Learning Environment for Middle School Science. Computer Software, Version 2.0 (2002).

[3] AquaMoose 3D, Georgia Institute of Technology http://www.cc.gatech.edu/elc/aquamoose (last access: 15/01/2009).

[4] Atlantis Quest, Indiana University. http://atlantis.crlt.indiana.edu (last access: 15/01/2009).

[5] G. Ayala, and Y. Yano, Collaborative learning environment based on intelligent agents. J. Expert Systems with Applications 14(1), 129-137 (1998).

[6] J. Baron, Glory and shame: powerful psychology in multiplayer games. Proceedings of the Game Developers Conference (GDC99), San Francisco, CA. Gamasutra (1999). http://www.gamasutra.com/features/19991110/Baron_01.htm (last access: 10/05/2008).

[7] H. S. Barrows and R. M. Tamblyn, Problem-based learning: an approach to medical education. Springer Series on Medical Education, Springer Publishing Company (1980).

[8] H. Barrows, Problem based learning in medicine and beyond: a brief overview. J. New Directions for Teaching and Learning 68, 3-12 (1996).

[9] K. Corti, Games-based learning; a serious business application. PIXELearning Limited (2006). http://www.pixelearning.com/docs/seriousgamesbusinessapplications.pdf (last access: 15/01/2009).

[10] K. Deibel, Team formation methods for increasing interaction during in-class group work. Annual Joint Conference Integrating Technology into Computer Science Education, Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITiCSE 2005), 291-295, Mote de Caparica, Portugal, ACM Press (2005).

[11] G. Denis and P. Jouvelot, Motivation-driven educational game design: applying best practices to music education. Proceedings of the 2005 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE05), 462-465, Valencia, Spain, Curran Associates Inc. (2005).

[12] E. Dieterle and J. Clarke, Multi-user virtual environments for teaching and learning. In M. Pagani (Ed.), Encyclopedia of multimedia technology and networking (2nd ed.), pp. 1033-1041, Idea Group, Inc., Hershey (2005).

[13] M. J. Dondlinger, Educational Video Game Desing: A review of the Literature. J. Applied Educational Technology 4(1), 21-31 (2007).

[14] B. J. Duch, S. E. Groh, and D. E. Allen, Why problem based learning? A case study of institutional change in undergraduate education. In B. J. Duch, S. E. Groh, and D. E. Allen (Eds.), The power of problem based learning, pp. 3-11, Stylus Publishing (2001).

[15] S. Freitas, Learning in Immersive Worlds: A Review of Game Based Learning. Technical Report prepared for JISC e-learning programme (2006).

[16] S. Freitas, Serious Virtual Worlds: A Scoping Study. Technical report prepared for JISC e-learning programme (2008).

[17] D. R. Garrison, Quality and theory in distance education: theoretical considerations. In D. Keegan (Ed.), Theoretical principles of distance education, pp. 9-21, Routledge (1993).

[18] J. P. Gee, What video games have to teach us about learning and literacy. Palgrave Macmillan (2003).

[19] C. Hansen and G. McCalla, Active open learner modelling. Proceedings of the Workshop on Learner Modelling for Reflection in AIED2003, Sydney, Australia (2003).

[20] R. T. Hays, The effectiveness of Instructional Games: A Literature Review and Discussion. Technical Report A539144, Naval Air Warfare Center Training Systems Division (2005).

[21] The Horizon Report, 2007 Edition. The New media consortium and the EDUCAUSE Learning Initiative (2007).

[22] D. W. Johnson and R. T. Johnson, Learning together and alone: cooperative, competitive and individualistic learning, 5th ed. Allyn & Bacon (1994)

[23] K. Kreijns, P. A. Kirschner, and W. Jochems, Identifying the pitfalls for social interaction in computer-supported collaborative learning environments: a review of the research. J. Computers in Human Behavior 19, 335-353 (2003).

[24] D. Livingstone and J. Kemp, Massively multi-learner: recent advances in 3D social environments. Computing and Information Systems Journal 10(2), (2006).

[25] G. Maudsley, Do we all mean the same thing by “problem –based learning”? A review of the concepts and a formulation of the ground rules. J. Academic Medicine 74(2), 178-185 (1999).

[26] Y. Miao, S. J. Holst, J. M. Haake, and R. Steinmetz, PBL-Protocols: Guiding and controlling problem based learning processes in virtual learning environments. In Proceedings of the 4th International Conference on the Learning Sciences (ICLS 2000), 232-237, Ann Arbor, MI, USA, Erlbaum (2000).

[27] M. Muehlenbrock, Learning group formation based on learning profile and context. International Journal on e-learning 5(1), 19-24 (2006).

[28] Multiverse, The Multiverse Network Inc. http://www.multiverse.net/index.html (last access: 15/01/2009).

[29] V. R. Neufeld and H. S. Barrows, The "The McMaster Philosophy": An Approach to Medical Education. Journal of Medical Education 49(11), 1040-1050 (1974).

[30] B. Oakley, R. M. Felder, R. Brent, and I. Elhajj, Turning student groups into effective teams. Journal of Student Centered Learning 2(1), 9-34 (2004).

[31] D. Oblinger and J. Oblinger, Is it age or IT: first steps towards understanding the net generation. In D. Oblinger and J. Oblinger, (Eds.), Educating the Net Generation surveys, Educause Publication (2005).

[32] S. Papert, The children's machine: rethinking school in the age of the computer. Basic Books (1993).

[33] M. Prensky, Digital natives digital immigrants. On the Horizon 9(5), 2001.

[34] Revolution. Project, Massachussetts Technological Institute, http://educationarcade.org/node/357 (last access: 15/01/2009).

[35] River City Project Harvard University: http://muve.gse.harvard.edu/rivercityproject (last access: 15/01/2009).

[36] K. Royle, Games-Based learning a different perspective. Innovate, Journal of Online Education 4(4), 2008

[37] V. Sánchez-Hórreo and R. M. Carro, Studying the impact of personality and group formation on learner performance. Proceedings of the 13th International Workshop on Groupware: Design, Implementation and Use (CRIWG 2007), Lecture Notes in Computer Science 4715, 287-294 (2007).

[38] Second Life, Linden Research Inc. http://secondlife.com/ (last access: 15/01/2009)

[39] SLED, Second Life Educators: https://lists.secondlife.com/cgi-bin/mailman/listinfo/educators (last access: 15/01/2009).

[40] Sloodle, Sloodle Project Team, http://www.sloodle.org (last access: 15/01/2009).

[41] C. A. Steinkuehler, S. J. Derry, D. K. Woods, and C. E. Hmelo-Silver, The STEP environment for distributed problem-based learning on the world wide web. Proceedings of Computer Support for Collaborative Learning (CSCL '02), 217-226, Bouler, CO, USA, Erlbaum (2002).

[42] J. W. Strijbos, The effect of roles on computer-supported collaborative learning. Unpublished doctoral dissertation. Heerlen, The Netherlands: Open University of the Netherlands (2004). Available online: http://www.ou.nl/Docs/Expertise/OTEC/Publicaties/jan-willem%20strijbos/Dissertation_Strijbos__Online_rev_1-11-04.pdf (last access: 15/01/2009).

[43] Tapped In, SRI, http://tappedin.org (last access: 15/01/2009).

[44] J. D. Vermunt, Inventory of Learning Styles (ILS) in higher education. University of Tilburg, Tilburg, Germany (1994).

[45] Virtual Leader Project: http://www.simulearn.net/pdf/practiceware_works.pdf (last access: 15/01/2009)

Caption for figures and tables

Figure 1. Ship and island of Nucleo.

Figure 2. Reference architecture for the Nucleo system.

Figure 3. Moodle forum and event panel displayed in the Nucleo environment.

Table I. Statistics about the last two years of PF courses.

Table II. Statistics of students getting to the final exam in the 2007-08 PF course.

Table III. Statistics of students passing the 2007-08 of PF course.

Table IV. Final exam marks (out of 10) and peer-evaluation results obtained by the different tribes (out of 10).

Table V. Statistics about traditional LP2 courses.

Table VI. Statistics about enrollment in the 2007-08 course of LP2. Students attending the course are in the columns “Traditional students” and “Nucleo students”.

Table VII. Statistics of students passing the 2007-08 course of LP2.

Table VIII. Final exam marks (over 10) and peer-evaluation results obtained by the different tribes (out of 10).

Biographies

Pilar Sancho Thomas received the MS in physics science from the Universidad Complutense of Madrid in 1996. She worked as an RD consultant for four years. Since 1999, she is a lecturer at the Department of Software Engineering and Artificial Intelligence of the Complutense Univeristy in Madrid, Spain, and a member of the <e-UCM> research group.

Rubén Fuentes-Fernández received the MSc and PhD in computer science from the Universidad Complutense of Madrid in 1997 and 2004 respectively. He worked as consultant and project leader in database systems for four years. He is currently a PhD Professor in the Department of Software Engineering and Artificial Intelligence at the Universidad Complutense of Madrid. His research interests are in the application of social sciences to the development of software systems.

Pedro Pablo Gómez-Martín received the MS and PhD in Computer Science in 2000 and 2008 from the Universidad Complutense of Madrid. He has taught in the Master of Videogame Programming in the same University since its first edition in 2004. His research involves the Artificial Intelligence and Software Engineering aspects of the development of educational video games, searching new ways to improve the way this software guide the student learning while keeping the development costs under control.

Baltasar Fernández-Manjón received the MS in physics science in 1991 and PhD in Computer Science in 1996 from the Universidad Complutense of Madrid. He is an associate professor at the Department of Software Engineering and Artificial Intelligence at the Complutense University in Madrid, Spain, and the director of the <e-UCM> research group.