An Adaptive, Competence based, Approach to Serious Games Sequencing in Technology Enhanced Learning

Luca Cuoco[1], Andrea Sterbini[2] and Marco Temperini[1]

[1]*Department of Computer, Control and Management Engineering, Sapienza University, Via Ariosto 25, Roma, Italy*

[2]*Department of Computer Science, Sapienza University, Via Salaria 113, Roma, Italy*

Keywords: Personalized e-Learning, Game based e-Learning, Adaptivity, Student Model.

Abstract: We present a platform for Technology Enhanced Learning, allowing the learner to follow a path of game applications towards a learning objective. The path is determined by the learner, by selecting each time the next game of her choosing. The games are defined by teachers and experts, or even imported as web resources: they are associated to the system through suitable metadata, that express their pedagogical meaning. The system's interface allows the student to navigate the repository of learning games (organized as a graph) and see among them those that she can select to undertake, and those that are not yet affordable. Whether a game is affordable, at a given moment, is determined by comparing the Student Model with the game's specification. In conclusion, the path of learning activities followed by the learner is built interactively, by the learner, according to learner's choice and the system's pedagogical guidance. The system has not yet been experimented in a real class: we report about its design and implementation, and provide the reader with some simulated applications showing the system's behaviour.

1 INTRODUCTION

In the area of Technology Enhanced Learning (TEL) game based learning is founded on the use of serious games, i.e. (digital) games designed and implemented in order to allow teaching/learning rather than (pure, exclusive) entertainment. Rather than discarding or dulling entertainment, the serious game aims to couple "meaningful play" (Salen and Zimmermann 2004) with a learning outcome.

Certain characteristics of games (such as the need to make choices, and perform an action, rather than explaining what to do) can facilitate learning and increase learning performance in applicative fields (Coller and Scott, 2009; Pasin and Giroux, 2011). In time the use of games and simulations has become one of the most significant approaches to assisted learning (Wu et al. 2012). There are studies that might reduce the enthusiasm (Kebritchi et al. 2012), yet this is one of the hot topics for research in TEL (Van Eck, 2006; Metawaa and Berkling, 2016).

Game Based Learning can also be accompanied by (other) playful aspects of the learning environment, referred to by the term "gamification" (Deterding et al., 2011; Vassileva, 2012). They are in general incorporating game-inspired elements in a non-game environment; examples are the support to leaderboard and badges.

Similarly to what happens in traditional adaptive learning systems (Liang et al., 2012; De Marsico et al., 2013), in an adaptive game based learning system the personalization of the learning offer is based on a Student Model, i.e. a representation of a set of personal traits of the individual student. Examples of such personal traits are, among others, the competence (that is the amount of knowledge possessed at that time by the learner, in relation to the subject matter at hand), or the student's learning style. In such an adaptive system, the student is proposed with different games, or with different (adapted) versions of the same game, according to the current state of her student model.

In this paper we present an approach to personalized game based learning, by the web system DEV, in which the learner is offered to build her personal learning (gaming) path, by selecting games among those available, provided that they are "affordable" according to her Student Model. The system provides an interface to allow teachers to build their course's game repository; the inclusion of a game is implemented by means of a "specification step", in which the teacher defines a "learning

Cuoco, L., Sterbini, A. and Temperini, M.
An Adaptive, Competence based, Approach to Serious Games Sequencing in Technology Enhanced Learning.
DOI: 10.5220/0006338305890596
In *Proceedings of the 9th International Conference on Computer Supported Education (CSEDU 2017) - Volume 1*, pages 589-596
ISBN: 978-989-758-239-4

(gameful) experience", by selecting the game from a general repository and assigning to it the pedagogical characteristics the game is going to have in the course, namely the competence needed in order to play the game fruitfully (i.e. with possibilities of success), and the competence that a success in the game witnesses. The student interface allows the learner to select a game for play. After "game over", the learner's Student Model is updated, hopefully by adding in it the competencies gained/witnessed through the learning game experience completion.

2 RELATED WORK

Hays (2005) reviewed 48 empirical studies on the effectiveness of instructional games published between 1973 and 2005. Results revealed that K-12 learners might find games useful for learning math, social sciences, vocabulary, but that no information existed about game-related learning for other disciplines (e.g., health and geography). In addition, games were found to be useful in teaching social sciences, physics, electronics and engineering principles to college students and, in the workplace, games showed positive learning effects on teaching attention, periscope skills, technical skills and so on. However, no evidence indicates that games are the preferred instructional method in all situations.

Based on the meta-analysis approach, Vogel et al. (2006) investigated the relative strengths of games and interactive simulations against traditional teaching methods. Results indicated that, across populations and situations, games and interactive simulations produce better cognitive gain outcomes.

Applicative fields in which Game based learning is found are many. In (Morelli et al. 2011; MacLean and Robertson, 2012) serious games are applied to provide guide and support to physical exercise. In (de Jong and van Joolingen, 1998) the development of a game simulation deemed "to teach about collisions in physics" is presented.

In the system proposed by the present paper there are two significant aspects that characterize the approach to game based learning: the adaptive student interface, and the possible function of games as assessment means; the former characteristics allows for the construction of a personalized learning (gaming) path, while the second support the updating of the student model after every gaming experience. So in the following we recall some research related to the above mentioned aspects.

Magerko et al. (2008) tackles the problem of delivering "adaptive games", i.e. game applications that can present the learner with differentiated interface and diverse basic game mechanics, depending on the player's gaming type. Significant factors for the personalization are the structure of the interface, and the way the learner is presented with knowledge (such as giving importance to high score, or visualizing texts about the subject matter, or having a time limit to frame the gaming activity). So a single game might come to represent a "space of possible games". Gamer types are taken from an analysis of literature; they are intrinsically motivated Explorers, extrinsically motivated Achievers, and extrinsically motivated "Winners". A mini-game prototype (S.C.R.U.B.) is presented, ranging over microbiology concepts.

In the system presented in this paper, the approach is cruder with respect to the games (we have different games helping pursuing different skills, yet only one version of each game), On the other hand, the personalization is obtained by helping the learner to build her own path of games, by choosing at each time the preferred game among those that are pedagogically affordable at the moment. By "affordable" we mean that a game can be met by the learner, according to the current state of her student model. In (Metawaa et al. 2016) an approach to personalization in game based learning is shown, for students with little or no access to teachers, whereas games (the learning activities available in the platform) are suggested basing on a lightweight modelization of the learner. The student model is focused on previous choices of the learner, and on a game rating derived by the preference accorded by learners to the game.

Another experience possibly related to the personalization in a game based learning environment is in (Lindberg and Laine, 2016), where foundational blocks for the provision of adaptivity in a game based learning system are studied: learner's *learning style*, and *gaming style*. Results shed some light on the play and learning styles among South Korean elementary school students. In this experience the provision for adaptation allowed to change the learning activity settings, according to the personally preferable style of the moment, and provided for an important degree of versatility in the experimented framework.

Regarding the nature of games as assessment means, Loh (2007) treats the problem of designing assessment in a game and uses the concept of "information trail" as a means to track, from a pedagogical point of view, the learner's in-game behaviour ("avatar tracking system") and assess her accomplishments.

3 THE DEV SYSTEM

The DEV system presents the learner with an environment in which several games are available and the learner student model is managed. In the present version of the system the games are all related to topics in Basic Physics, and are designed to allow the learner for experimenting with concepts (such as principles, equations, and computation of the motion of a body in two and three dimensions) and build the answer to questions (such as composing the right equation to use, or computing some values). At the end of each game it is possible to appreciate whether the play has been successful and to what extent (and the system is passed information suitable to update the student model).

We have designed the DEV system around the main goal of allowing the learner to follow a personal path of experiences to reach a goal knowledge (a set of target skills defined by the teacher). The definition and management of the learner's Student Model is material to such aim: the system points out for the learner, among the whole set of available games, those that are at the moment "affordable" for her, so the learner can choose her path, although according to a pedagogical guidance. In this we have taken inspiration by the concept of "zone of proximal development" known in the educational theory of Vygotsky (Chaiklin, 2003; Vygotskij, 1981).

DEV is comprised of a repository where all the available games are collected. Different courses can be built by associating games from the repository to the course area, according to the pedagogical specifications described in the next section.

4 GAME DEFINITION AND STUDENT MODEL

The games presently available in DEV are implemented by different technologies: so far they are either interactive questionnaires or game applications built through the Unity3D framework (Unity3D, 2016). Besides its implementation, each game, say G, is specified by a set of metadata, declaring the skills that are necessary in order to play the game (Required Skills – G.RS) and those that can be considered owned by the learner once she has been successful in the game (Acquired Skills – G.AS).

A *skill* is defined as a pair $<k, c>$. In it k is a "Knowledge Item" (KI), that is an identifier (a name) for a concept or ability (which is further described, in the DEV system with a Glossary topic), while c is the "certainty" that is associated to the possession of k. The certainty factor is a real value in $[0,1]$. The skill's certainty factor, basically describes how much we can be sure that the student does actually possess the related KI. The Student Model (SM) contains the current list of skills possessed by the learner. When a game, G, has been experienced by a learner, l, then she is supposed to have acquired the skills in G.AS, and her student model, l.SM, can be updated accordingly. In particular, if G is specified as follows, where the r_i and a_i are KIs

- $G.RS=\{<r_1,rc_1>, \ldots, r_n,rc_n>\}$
- $G.AS=\{<a_1,ac_1>, \ldots, r_m,ac_m>\}$

The basic updating rule for the SM is as follows:

for all <k,c> in G.AS, <k,c> is added to l.SM

In other words, each skill acquired by the game, is added to the learner's SM, with a certainty equal to that specified in the game definition.

The basic SM updating rule, though, has to be enhanced: it is apparent that, for any given skill, the system could provide (or actually must provide) several games acknowledging it. Moreover, it is possible that a learner will select and complete the same game several times. This, from the teacher's perspective, can be interpreted as useful practice: repeating a game, or, possibly, playing a game that awards skills that are already in one's student model, is a strengthening practice for such skills. On the other hand, while the certainty assigned to a skill that is just being added to the Student Model after a game (a newly acquired skill) is a remarkable fraction of 1 (such as 0.6, the default we are currently using in fact), the increase in certainty awarded to a skill already in *l.SM* ought to be quite lower. This is to avoid reaching a certainty of 1 without a considerable effort, and yet allows for a perceivable increase (fostering a sense of accomplishment in the learner).

For a game specified as above, the cases to be managed during the post-game student model update are as follows (with l a learner, and G a game)

- *for all* $<k,c> \in$ G.AS, with $<k,c> \notin l$.SM, $<k,c>$ *is added to l.SM*
- *for all* $<k,c> \in$ G.AS, with $<k,C> \in l$.SM, i.e. with the skill already present in the student model, with an associated certainty C, the skill's certainty is updated by just a fraction of c: $C \rightarrow \phi(C, c, n_G)$, where ϕ computes the new

value for *C*, according to the value *c* and to the number of times the game *G* has been already played n_G.

So the basic updating rule, valid only for the acquired skills $<k,*> \notin l.SM$, is enhanced by applying the principle that we cannot grant the whole amount of certainty that would be granted the first time the game is solved, or the first time the skill is placed in *l*.SM. This is done by applying an algorithm that reduces *c* according to the cases (a repeated game will acknowledge just a small increase of certainty, progressively decreasing to none_at_all; on the other hand, already possessed skills have a similar, yet less drastic, treatment granting that strengthening competence through different games is more valuable than in the case of game replay.

5 PEDAGOGICAL ASPECTS OF THE SYSTEM

The games occur in a general learning activity: a course, or a cycle of (game-)experiences flanking a course. The objectives of such an activity (Target Skills - TS) are defined as a set of skills as well. The abovementioned general learning activity can be considered completed once all the elements of TS are "covered" by the learner's model: Namely, when the TS is satisfied by the student's SM.

Definition 1. (Skill coverage)

Given two skills, $<k_a,c_a>$ and $<k_b,c_b>$ we say that $<k_a,c_a>$ *covers* $<k_b,c_b>$, and write $<k_a,c_a> \rightarrow <k_b,c_b>$

$$\text{iff } k_a=k_b \text{ AND } c_a \geq c_b$$

Definition 2. (Skill set Satisfaction)

Given a leaner *l*, and a set of skills,

$$S = \{<K_1,C_1>, <K_2,C_2>, \ldots, <K_q,C_q>\},$$

we say that the learner's student model

$$l.SM = \{<k_1,c_1>, \ldots, k_p,c_p>\}$$

satisfies S, and write $S \subseteq SM$

iff $\forall\, i \in [1\ldots,q], \exists\, j \in [1,\ldots,p]$ such that $<k_j,c_j>$ covers $<K_i,C_i>$.

So, the course, or cycle of experiences, with target skills TS, is considered concluded by a learner *l*, once $TS \subseteq SM$.

In DEV, while all games can be inspected by learners at will (to see their definitions), they are not always available for playing (*accessible* henceforth). In particular, only the games, G, whose requirement set, G.RS, is empty, are accessible by everybody. Otherwise, G is accessible for *l*, if and only if *l.SM* satisfies G.RS: $G.RS \subseteq l.SM$.

Definition 3. (Game accessibility)

Given a learner *l*, and a game G, we say G is *accessible* by *l*, and write $G \in ZPD(l)$, iff

$$G.RS \subseteq SM$$

Namely, the system's interface shows to the learner all the available games, let her inspect them, yet let her selects for use only those that are accessible to that particular learner.

The interface is shown in Fig.1; further description of the interface is in the next section.

The AS/RS annotation and the current student model SM cooperate to keep the student inside the set of activities that she is able to try without frustration. In this we get inspiration from the Vygotsky's Zone of Proximal Development (Chaiklin, 2003; Vygotskij, 1981). Notice that the certainty factors are not to be interpreted as a description of knowledge levels 'a la Bloom (Bloom, 1964; Anderson and Krathwohl, 2000), because cognitive levels are not additive. To represent different knowledge levels, we use different KIs.

Finally, we describe an additional feature of the system, although it is not exclusively connected to game experience. As it is imaginable, the games in the system are to be implemented in such a way to offer both gaming activity and communication with the system itself. The communication part is crucial, as it is providing the system with evidences to use for the personal student model updates. We call such a game a *formal resource*. Also a questionnaire (as the system allows the construction of multiple choice questionnaires) can function as a formal resource in the above "communication" sense (although it is not properly a game).

On the other hand, the possibility to join into the system "informal resources" is valuable, and very likely to be requested.

Such informal resources are all those activities that 1) can be easily retrieved on the web, and 2) are (by consequence) not specifically programmed to communicate with the system, and can hardly provide the system with feedback: A YouTube clip, an swf game, taken "as is", and actually any other resource that is available on the web (through a uniform resource indicator, for instance).

So the system provides for the possibility to import an informal resource, by associating to it a questionnaire, so to make of it an overall formal resource.

6 SYSTEM'S FEATURES AND FUNCTIONALITIES

From the point of view of a course designer, the system provides a dedicated area to upload, add and modify the contents which compose the experiences, an editor for their distribution on the navigation graph, and an area for learners' management.

From the point of view of the student, the system provides the environment to explore the course graph and to face the experiences that are accessible, depending on her SM. In addition, a glossary is provided to explain the meaning of the Knowledge Items. Fig. 1 shows a sample student's view of the course repository: games accessible to the learner are circled in yellow; games already played (still accessible) are circled in violet, and the lock represents non-accessibility. Placing the mouse arrow on a node-game highlights the connections of the selected node. A connection is represented by a directed arc in the graph: an arc from G1 to G2 says that part of, or whole, G1.AS could be used to cover (part of) G2.RS, towards making G2 accessible.

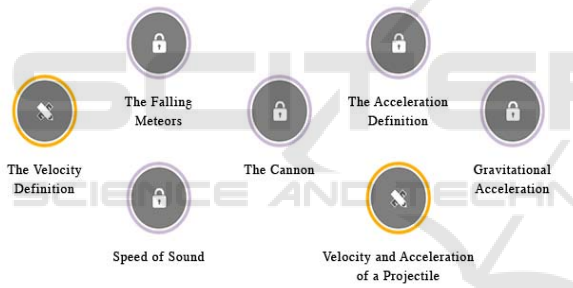

Figure 1: Navigation Graph Tool. Initial state of the games when SM = {} – only some initial games are accessible.

So the student can plan her next experience also basing of what it would unlock (let acquire). This allows, in turn, to build one's own path towards the TS satisfaction. Fig. 2 shows a visualization of the learner's Student Model.

The system provides a questionnaire editor to compose questions and evaluate answers. The questions can be enhanced with Javascript code (e.g. to generate different random parametric instances of the question each time the experience is visited) through an integrated scripting environment.

On the side of permissions management, the system provides ways to freely associate privileges to the users. One user can view a specific area of the system in read-only mode, while another can have more privileges to edit and delete contents. In a second time, the first user can obtain the access to modification. This way, work-flows can be defined to manage different developers working on the same activities (e.g. learning designer, web designer, multimedia/graphic designer, Javascript developer).

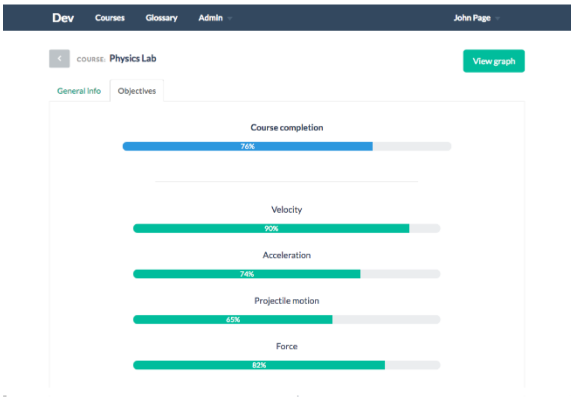

Figure 2: Visualization of the student model. For each element in TS, possession and certainty are shown. The upper bar gives a comparative degree of completion for the whole course (i.e. a degree of current coverage for TS).

6.1 The System in Action

In the following we show a very simple example of how the content definition produces the student view of the course.

The first step is the identification of the Knowledge Items to match the learning needs. We suppose to build a virtual physics laboratory where students can interact with the laws of physics in an immersive environment that simulates everyday life problems. The first KIs defined are: Vectors, Force, Velocity, Acceleration, Linear motion, Uniformly accelerated linear motion, Projectile motion. For each KI a description of the meaning is added to the glossary. Having KIs, we can then define the course learning objectives, TS.

As previously described, each skill obtained by a student has a certainty value between 0.0 and 1.0. We want the student to let develop her Student Model to reach TS satisfaction. Suppose that

TS = {<Vectors, 0.6>, <Velocity, 0.6>, <Acceleration, 0.6>, <Linear motion, 0.8>, <Uniformly accelerated linear motion, 0.8>, <Projectile motion, 0.9>}

To cover TS the student needs to enhance her SM, by executing some of the available experiences. Supposing that SM is empty at start, the course must contain a group of initial experiences with empty requirements. When new skills are acquired, the SM grows, new experiences become accessible, and the course can proceed. An example of starting experience, from Fig. 1, follows:

Name: The Velocity Definition
Informal resource: A video with real life examples and formulas definition
Formal resource: A game application that let the learner do experience with the concept of velocity and also verifies the comprehension of the velocity concept
Req. Skills: { } empty set
Acq. Skills: {<Vectors, 0.6>, <Velocity, 0.6> }

Let's assume that the learner undertakes this first experience, successfully; now Fig. 3 shows the outcome: the experience becomes "played" in the interface, and new experiences are unlocked (become *accessible*).

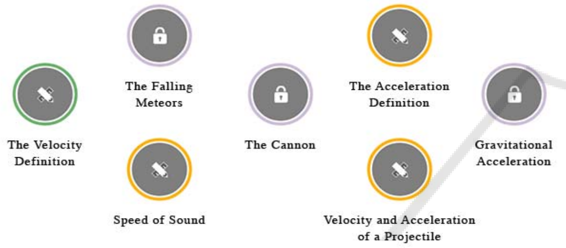

Figure 3: State after completion of the first game. Now SM = { <Vectors, 0.6>, <Velocity, 0.6> } and the experiences "The Acceleration Definition" and "Speed of Sound" became accessible.

Let's also assume that the student undertook the following experience:

Name: The Acceleration Definition
Informal resource: An interactive animation that presents the concept and the formulas
Formal resource: A intelligent questionnaire powered by the scripting environment
Req. Skills: { <Vector, 0.6>, <Velocity, 0.6> }
Acq. Skills: { <Acceleration, 0.4> }

Once the above experience has been completed, the student still needs another experience providing at least other 0.2 certainty value of Acceleration to fulfil the course Target Skills (where a 0.6 certainty is required for the Acceleration KI). Well, the following experience is available, and now accessible, to help covering the gap (Fig. 4 shows the consequences on the student model of undertaking this experience):

Name: Gravitational acceleration
Informal resource: An animation of the newton's apple with some mathematical reflections
Formal resource: A 3D application with inputs for the replication of the falling apple
Req. Skills: { <Acceleration, 0.4> }
Acq. Skills: { <Acceleration, 0.6> }

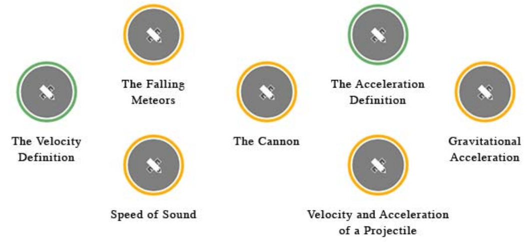

Figure 4. State of the Navigation Graph Tool after completion of two games. Now SM = { <Vectors, 0.6>, <Velocity, 0.6>, <Acceleration, 0.4> } and the experience "Gravitational acceleration" became accessible.

The idea of this example is to show how the course designer could make a whole graph of experiences available (games, closed answer questionnaires, informal resources made formal by questionnaires), and how the learner can select a personal path to let her SM grow towards TS satisfaction. A course would have to provide several alternative starting nodes; moreover, several different resources ought to have intersecting AS, and grant the same skills. It is this redundancy the feature that would provide learners with a true opportunity to build her own personal learning path.

An example of DEV physics games is shown in Fig. 5, where a 3D physics soccer simulator asks the student to solve the problem of passing the ball to a team-mate.

Figure 5: Soccer simulator.

The solution needs stating 1) the right trajectory, as resulting from a correct definition of the motion equations (as shown in Fig. 6), and 2) a correct shooting angle.

Figure 6: Soccer simulator. Construction of the motion equation, through a puzzle game.

7 CONCLUSIONS

DEV is not yet being experimented in a real class, mainly due to the necessity of providing the system with a wealth of good games, to obtain the aforementioned redundancy. So the main activity related to DEV in this time regards the development of games, and make them available into the system. We are currently working of the definition of an API allowing to produce different scenarios of a same game. Moreover, editors for the rapid production of small games, such as puzzle games, ruzzle based games and Tower games, are under design. This should allow us to nourish the system and proceed beyond the simulation we have presented in this paper.

When the presented approach will be more mature, we plan to extend it towards collaborative (game based) learning, by means of multi-player game activities (Sterbini and Temperini, 2009; Nebel et al. 2016). A parallel line of development is towards peer-based assessment and learning (Cho and MacArthur, 2010; Sterbini and Temperini, 2012; Sterbini and Temperini, 2013; Isabwe, 2013; Tenorio et al. 2016). By involving new teachers, we also plan to extend the learning domain to mathematics and formal languages (Hancock, 1995; Formisano et al. 2000; Formisano et al. 2001; Isabwe, 2013). In a near future we want to enhance DEV by gamification applied to the overall system interface. Examples of interventions are as follows:

- making the games more immersive, by showing a personal Avatar actively interacting with the graph of experiences and within the experiences themselves;
- engaging learners in competitions, as it can be supported by showing a leaderboard or awarding badges;
- allowing a different visualization of the student model, under the form of the avatar inventory of abilities, with "power-ups" related to the measure of certainty. In some sense the Student Model already can be thought as an inventory of items, yet the games should be modified to behave differently depending on the "items" carried by the player.
- collecting an overall measure of experience (XP points, which could be thought as a measure of reputation or of learning involvement) describing how well the student has played the whole course.

REFERENCES

Anderson, L. W., Krathwohl, D. R. (eds.), 2000. A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives. Allyn and Bacon.

Bloom, B.S. (ed.), 1964. *Taxonomy of Educational Objectives*. David McKay Co. Inc., New York.

Chaiklin, S., 2003. The zone of proximal development in Vygotsky's analysis of learning and instruction. In: Kozulin, A., Gindis, B., Ageyev, V., Miller, S. (eds.) *Vygotsky's Educational Theory in Cultural Context*, pp. 39–64. Cambridge University Press.

Cho, K., MacArthur, C., 2010. Student Revision with Peer and Expert Reviewing. *Learning and Instruction* 20(4).

Coller, B.D., Scott, M.J., 2009. Effectiveness of using a video game to teach a course in mechanical engineering. *Computers & Education*, 53, 900â€"912.

de Jong, T., van Joolingen, W.R., 1998. Scientific discovery learning with computers simulations of conceptual domains. *Review of Educational Research*, 68, pp. 179-202]

De Marsico, M., Sterbini, A., Temperini, M., 2013. A strategy to join adaptive and reputation-based social-collaborative e-learning, through the Zone of Proximal Development. *Int. J. of Dist. Ed. Tech., IJDET*, 11(3).

De Marsico, M., Sterbini, A., Temperini, M., 2014. Experimental Evaluation of OpenAnswer, a Bayesian Framework Modeling Peer Assessment. In Proc. *14th IEEE ICALT*.

Deterding, S., Khaled, R., Nacke, L., Dixon, D., 2011. Gamification: Toward a Definition, Proc. *CHI 2011 Gamification Workshop.*

Formisano, A., Omodeo, E.G., Temperini. M., 2000. Goals and benchmarks for automated map reasoning. *J. of Symb. Comp*., 29/2, pp.259-297, Academic Press.

Formisano, A., Omodeo, E.G., Temperini. M., 2001. Layered map reasoning: An experimental approach put to trial on sets. *Electronic Notes in Theor. Comp*. Sci., 48, pp. 1-28. Elsevier, Amsterdam.

Hancock, C.L., 1995. Enhancing Mathematics Learning with Open-Ended Questions. *The Mathematics Teacher*, 88/6. ProQuest SciTech Coll.
Hays, R.T., 2005. The Effectiveness of Instructional Games: a Literature Review and Discussion. Tech.rep. NAWC Training Systems Div., Orlando, FL, USA.
Isabwe, G.M.N., 2013. Enhancing Mathematics Learning Through Peer Assessment Using Mobile Tablet Based Solutions. *Doctoral Dissertations at the University of Agder* 64. ISBN: 978-82-7117-731-7.
Kebritchi, M., Hirumi, A., Bai, H., 2010. The effects of modern mathematics computer games on mathematics achievement and class motivation, *Computers & Education* , 55(2), 427-443.
Liang, M., Guerra, J., Marai, G.E., Brusilovsky, P., 2012. Collaborative e-learning through open social student modeling and Progressive Zoom navigation, Proc. *8th Int. Conf. on Collaborative Computing: Networking, Applications and Worksharing*, pp. 252–261.
Lindberg, R.T.S., Laine, T.H., 2016. Detecting Play and Learning Styles for Adaptive Educational Games. In Proc. *8th Int. Conf. on Computer Supported Education (CSEDU)*, Vol. 1, pp. 181-189, ScitePress.
Loh, C.S., 2007. Designing Online Games Assessment as Information Trails. In D. Gibson, C. Aldrich, M. Prensky (eds.) *Games and Simulations in Online Learning: Research and Development Frameworks*, InfoSci, Hershey.
Macvean, A. and Robertson, J., 2012. iFitQuest. In Proc. *14th Int. Conf. on Human computer interaction with mobile devices and services - MobileHCI'12*.
Magerko, B. Heeter, C., Fitzgerald, J., Medler, B., 2008. Intelligent adaptation of digital game-based learning, Proc. *Conf. on Future Play: Research, Play, Share*.
Metawaa, M., Berkling, K., 2016. Personalizing Game Selection for Mobile Learning - With a View Towards Creating an Off-line Learning Environment for Children. In Proc. *8th Int. Conf. on Computer Supported Education, CSEDU*, Vol.2, pp. 306-313.
Morelli, T., Foley, J., Lieberman, L., 2011. Pet-NPunch: Upper Body Tactile / Audio Exergame to Engage Children with Visual Impairments into Physical Activity. In Proc. *Graph. Interface*, pp. 223–230.
Nebel, S., Schneider, S., Rey, G.D., 2016. From duels to classroom competition: Social competition and learning in educational videogames within different group sizes. *Computers in Human Behavior*, 55.
Pasin, F., Giroux, H., 2011. The impact of a simulation game on operations management education. *Computers & Education*, 57, 1240-1254.
Salen, K., Zimmermann, E., 2004. *Rules of play: Game design Fundamental*s. MIT Press.
Sterbini, A., Temperini, M., 2009. Collaborative Projects and Self Evaluation within a Social Reputation-Based Exercise-Sharing System. In Proc. *IEEE/WIC/ACM Int. Joint Conf. on Web Intelligence and Intelligent Agent Technologies, WI-IAT'09,* Vol.3, pp.243-246.
Sterbini, A., Temperini, M., 2012. Supporting Assessment of Open Answers in a Didactic Setting. In Proc. *IEEE 12th ICALT.*
Sterbini, A., Temperini, M., 2013. OpenAnswer, a framework to support teacher's management of open answers through peer assessment. In Proc. *43rd IEEE Frontiers in Education, FIE,* pp. 164—170.
Tenorio, T., Bittencourt, I.I., Isotani, S., Pedro, A., Ospina, P., 2016. A gamified peer assessment model for on-line learning environments in a competitive context. *Computers in Human Behavior* 64 pp.247-263.
Unity3D accessed 2016.12.19. Unity 3D Manual web site: https://docs.unity3d.com/Manual/UnityManual.html.
Vassileva, J., 2012. Motivating participation in social computing applications: a user modeling perspective, *User Modeling and User-Adapted Interaction*, 22, 1–2, pp. 177–201.
Van Eck, R., 2006. Digital game-based learning: It's not just the digital natives who are restless, *EDUCAUSE review*, 41(2), 16.
Vogel, J.J., Vogel, D.S., Cannon-Bowers, J., Bowers, C.A., Muse, K., Wright, M., 2006. Computer Gaming and Interactive Simulations for Learning: a Meta-Analysis. *J. Educational Computing Research*, 34(3).
Vygotskij, L.S., 1981. The development of higher forms of attention in childhood. In: Wertsch, J.V. (ed.) *The Concept of Activity in Soviet Psychology*. Sharpe.
Wu, W.H., Chiou, W.B., Kao, H.Y., Hu, C.H.A., Huang, S. H., 2012. Re-exploring game-assisted learning research: The perspective of learning theoretical bases, *Computers & Education*, 59(4), 1153-1161.