M-learning Programming Platform: Evaluation in Elementary Schools

Efthimios Alepis and Christos Troussas
Department of Informatics, University of Piraeus, Piraeus, Greece
E-mail:talepis@unipi.gr, ctrouss@unipi.gr

Keywords: affective interaction, evaluation, intelligent tutoring system, mobile learning, programming

Received: January 19, 2017

Mobile learning has invaded the everyday life and therefore evaluation methodology and reporting should be used so that mobile technologies are tested out in a variety of learning contexts. To this direction, a mobile learning platform, named m-AFOL, was designed for elementary school students so that they are taught basic programming principles. The platform is a sophisticated learning tool that incorporates affective interaction among learners and tutoring agents in order to maximize educational gains. The paper presents a two-level evaluation study of m-AFOL, estimating its effectiveness and acceptance rate. The evaluation concludes that the mobile facilities of the resulting intelligent tutoring system are highly acceptable (over 80%) from young learners, being keen on using mobile devices while learning. Concluding. the mobile platform is evaluated in comparison with a desktop version of resulting system.

Povzetek: Razvita je bila inovativna mobilna platforma za učenje, imenova m-AFOL.

1 Introduction

As innovation proceeds with its quickly developing pace, individuals are increasingly involved in technological issues, including both software and hardware. Hence in recent years, the rapid development of high and new technology has opened new horizons in computer-assisted instruction (Troussas et al., 2013). At the same time, there has been an increasing focus on mobile software applications as a result of the rapid development of mobile networks (Troussas et al., 2014 a). Particularly in learning, individuals of all ages use such technology to bolster their knowledge through instruction. The ever increasing mobile population can assist mobile learning which focuses on the mobility of learners and instructors, interacting with portable technologies (Troussas et al., 2014 b). In our days, computer science is being taught in schools, even in lower school grades (Virvou et al. 2013). Young people figure out how to utilize computers, basic algorithmic principles and even pseudo code. In any case, most computer programming languages are far complicated to be taught as they require knowledge from many domains as prerequisites. Moreover, programming used to be quite incomprehensible in its infancy and even today it continues evolving to more "natural" languages for human beings. A characteristic example in this direction is the well-known programming paradigm of object oriented programming (Shieh et al. 1996), (Pastor et al. 2001).

(Abdul et al. 2013) investigated how children view technology according to their perspectives. This study concludes that children actually want technology which is ubiquitous, wearable, natural in interaction and child-centered. Towards this direction, an interesting study is of McKenney & Voogt (2010), who have examined 167 young children access, perceptions and use of technology within and outside of school settings. Their findings, among other, include the fact that children's attitudes toward computers are positive and also inform the debate on the desirability of young children's exposure to computers at home as well as in educational settings. The research presented in (Jackson et al. 2012) examined relationships between children's information technology (IT) use and their creativity, including 491 12-year-old participants. Regardless of race or gender, analyzing the participants' results indicated that using technological means and specifically videogame playing was associated with greater creativity. Over the last decade, parents from all over the world have been positively surprised, watching their young children use and "consume" technology related to their learning. Even very young children have shown comfort and confidence in using software and they can follow pictorial directions and use situational and visual cues to understand and think about their activities (Clements & Nastasi, 1993).

Teaching programming to young children, even in elementary schools is becoming quite popular over the last years. The term "programming" was probably unknown to the majority of people worldwide a few decades ago. However, as technology becomes an integral part of our everyday life, "programming" provides us, in a sense, with a way to communicate with computers. Younger children have a flair of learning easier foreign languages and this may also be the case with programming languages. The authors of (Fessakis et al. 2013) present a case study concerning the dimensions of problem solving using computer programming by 5-6 years old kindergarten children. The authors' research evidence supports the view that children enjoyed the engaging learning activities and also had opportunities to develop mathematical concepts, problem solving and social skills. Kordaki (Kordaki, 2010) has presented and also evaluated a computer-based problem-solving environment named LECGO, designed for the learning of computer programming using Turbo C by beginners. The resulting learning environment has been evaluated by 12th grade students. The authors' findings from the evaluation study indicated that students gain better results within LECGO than in both the paper and pencil environment and the typical programming environment of Turbo C, while performing similar activities. Another interesting and quite relevant study is that of (Werner et al. 2012). In this paper, the authors describe a semester-long game-programming course where 325 middle school students used Alice, a programming environment. Their results of the analysis of 231 programming games show that many games exhibit successful uses of high level computer science concepts.

However, it is inferential that modern programming languages are not "designed" to be used by children, since their principal purpose was to be handled by grown-up programmers in order to produce effective and robust software applications. This comes in contrast with the aforementioned realization that children have a very efficient way of understanding and "communicating" with all kinds of technological means. In view of the above, the authors of this paper have proposed and designed a new programming language, incorporated in a mobile platform named m-AFOL (Alepis & Virvou, 2014). This system is targeted to children as end users and is highly user friendly and interactive since it is able to handle affect through a sophisticated tutoring agent module. This paper focuses on the evaluation of this platform in order to test its effectiveness as an educational tool and also to test its acceptance from elementary school students.

The remaining sections of this paper are organized as follows. In section 2 the authors present briefly an overview of the mobile learning platform with corresponding user interaction paradigms. In section 3 the settings of the evaluation study are shown, accompanied with its preliminary results. Section 4 shows a discussion about the significance and the importance of the evaluation study's findings. Finally, in section 5, the conclusions of this paper are presented.

2 Overview of the mobile learning platform

"Logo" programming language was introduced early in 1967 (Frazier, 1967). The programming language developers' objective was to take the best practices and ideas from computer science and computer programming and produce a user friendly interface that would be suitable for the education of young children. Logo has been used mainly in the past as a teaching language for children but its list handling facilities made it also useful for producing useful scripts. A detailed review on the "Logo" programming language from its early stages can be found in (Feurzeig, 2010).

Modern programming languages try to provide as much user-friendliness as possible while retaining their full programming functionality. Learning a programming language is a complex cognitive process, while it is argued that how people feel may play an important role on their cognitive processes as well (Goleman, 1981). At the same time, there is a growing number of researchers who acknowledge that affect has been overlooked by the computer community (Picard & Klein, 2002). Perhaps a remedy towards the problem of effectively teaching children through intelligent tutoring systems and educational applications may be found in rendering computer assisted and mobile-learning systems more human-like and thus more affective. Hence, the incorporation of emotion recognition modules as well as the interaction with animated tutoring agents in a tutoring system can be quite useful and profitable (Elliott et al., 1999). The presence of animated, speaking agents has already been considered beneficial for educational software (Johnson et al., 2000).

However, after a thorough investigation in the related scientific literature we come up with the conclusions that there is a shortage of educational systems that incorporate multi-modal emotion recognition, while we did not find any existing programming languages that incorporate emotion recognition and/or emotion generation modules in mobile devices. A relevant work in the personal computer dimension is that of Kahn (Kahn, 1996), where an animated programming environment for children is described, named ToonTalk.

The resulting platform named m-AFOL is the acronym for "mobile-Affective Object Oriented Logo Language", as a modernization of the past "Logo" programming language additionally including affective interaction and interaction with mobile portable devices. In the implementation of the m-AFOL language, the prevailing programming dimension is that of object oriented programming (Alepis & Virvou, 2014). Through the language's object-oriented architecture, children get familiar with the object oriented paradigm. Hence, an m-AFOL program may be viewed as a set of interacting objects, with their attributes and their methods, as opposed to the conventional Logo model, in which a program is seen as a list of tasks. Furthermore, the initial goal to create a programming language enjoyable by children is further improved through the m-AFOL language, by a highly user-friendly user interface, designed for affective interaction between high-tech mobile phones (incorporating their own Operating System) and young children.

A major concept in the presented educational platform is the mobile interaction. However, this kind of interaction has some prerequisites, regarding mobile hardware and software specification. More specifically, a smartphone with a multi-touch screen 3.7 inches or higher is required, while at this moment the application is available for the Android OS (Android version 4.0 and higher). Internet wireless connection is also required, since each mobile device acts as a client to a main web server who is responsible for handling and manipulating user models and also for the system's affective interaction capability. This interaction is illustrated in figure 1.

Young students may use the educational application from a smartphone, having the opportunity to learn programming through m-learning courses. The theory is presented in graphical form while an animated tutoring agent is optionally but desirably present and may alternatively read the theory out loud using a speech engine. Students are prompted to write simple programming commands producing lines and basic shapes as first steps towards getting familiar with the programming environment. More advanced level students are given graphical illustrations as examples and are prompted to write complete and sometimes quite complicated programs in the m-AFOL language in order to produce drawings, shapes and particular graphical illustrations. The mobile application is installed locally in each student's smartphone device, while an active internet connection is required for the communication with the main server. As it is already mentioned, the resulting application is targeted to the Android OS platform, while there are future plans for the integration of the resulting ITS to other mobile operation systems as well. Examples of interaction with the m-AFOL programming platform are illustrated in figures 2 and 3. It is noteworthy that in these examples the tutoring agent would also try to sympathize with the aims of each student trying to elicit human emotions. More specifically, figure 2 illustrates the programming mobile frame, where students may write programming commands. Special buttons help them store or load predefined "objects". Observing the m-AFOL's programming code, its Object Oriented "nature" is easily noticeable. Figure 3 illustrates the output of a student's programming code that results in the creation of a "sun" object and a "house" object. The animated agent (of the form of a cartoon computer) is congratulating the student for successfully completing a programming exercise.

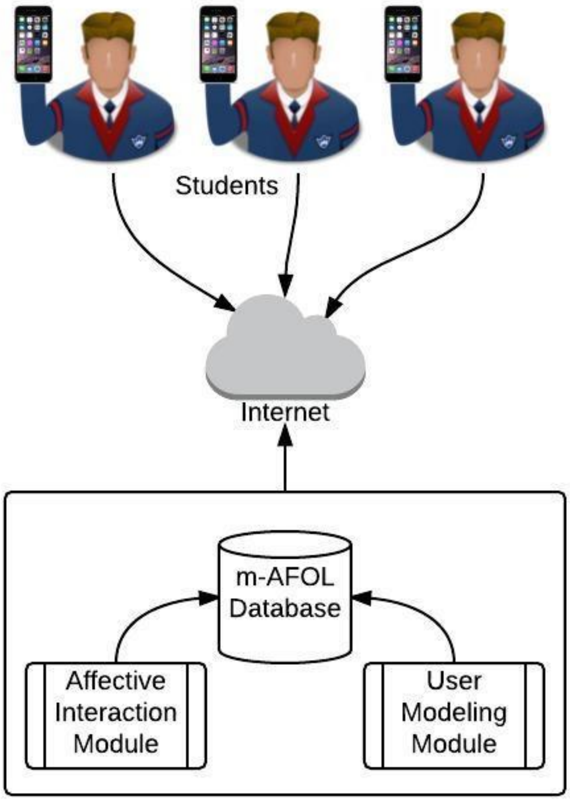

Figure 1: m-AFOL Architecture.

3 Evaluation study

In the evaluation study, 40 elementary school students from two different classrooms participated (20 students from each class). The two classes were from the same

Figure 2: Programming in the m-AFOL mobile environment.

Figure 3: Programming result and interaction with the tutoring agent.

school and of the same grade. The school, that was chosen, is a public school, located in Athens, the capital city of the country. Hence, the school can be seen as a representative sample, since it adequately replicates the larger statistical population in terms of students' characteristics. School teachers also provided very valuable help in the whole evaluation study since they also participated both in the presentation of the ITS to the students and also provided assistance to their students while they interacted with the educational platform. The first class evaluated m-AFOL through mobile devices that were given to all students, while the second class evaluated the educational system through its conventional desktop interface that functions through personal computers. This division was very crucial in order to compare the mobile platforms effectiveness with the traditional way of tutoring through computers. As a result, both classes had given the appropriate hardware and as a next step they were given a brief presentation on how to use the educational platform. Consequently, each class had enough time to spend interacting and completing a whole lesson with m-AFOL, while all students had a small break in the middle of the session. After the completion of their interaction (class A with mobile devices and class B with personal computers), all students were given questionnaires to complete with guidance from the evaluators and also their teachers.

The evaluation study was conducted with the use of self-supplemented scale questionnaires incorporating closed questions for the students. For our research, we have used 28 questions regarding students:

- three (3) exploratory questions
- eight (8) questions regarding navigation in the platform
- six (6) questions regarding the user interface
- five (5) questions regarding hardware and software quality
- four (6) questions regarding evaluation of learning

N	Questions
1	Rate the application's user interface (1-10)
2	Rate your learning experience (1-10)
3	Did you like the interaction with the educational agent (1-10)?
4	Did the agent respond to your questions (1-10)?
5	Did the agent respond to your feelings/mood (1-10)?
6	Could you rate the affective interaction (1-10)?
7	Would you like to use this platform in your school (1-10)?
8	Did you find the tutoring system simple in use (1-10)?
9	Rate the overall quality of the ITS (Device and Software) (1-10)
10	Did you find the ITS helpful for your lesson (1-10)?
11	Would you suggest the ITS to your friends to use it (1-10)?
12	Rate the easiness in handling the device your were given (1-10)

Table 1: Sample of questions of the evaluation study.

It was observed that students became familiar easily and very quickly with the educational software, its features and its functionalities. Their interest was undiminished during the whole period of their interaction with the educational application.

Table 1 summarizes a basic set of questions that were asked to the students after their interaction with the application. These questions follow a 1 to 10 ranking (lower is negative, higher is positive) model.

Finally, figure 6 illustrates the statistical significance of questions 2, 7, 8 and 12. For the Null hypothesis:

"There is no difference between the two groups of students" the t-Test rejects the hypothesis for the fore mentioned 4 questions, while the remaining 8 questions seem to not have statistical differences between the two groups of students.

4 Discussion on evaluation results

The desktop version of m-AFOL, named AFOL has been presented in (Alepis, 2011). Providing a mobile interface for the educational platform was a great challenge since integrating all the system's functionalities in a mobile OS was a quite demanding task. This operation included supporting user modelling and also affective interaction in the new mobile interface. Both for building user models and also for the reasoning mechanisms used in affective interaction, sophisticated techniques including machine learning (Support Vector Machines) and stereotypic reasoning modules had to be used. Both the amount of data and also the processing mechanisms required hardware specifications that far exceed the hardware of the state of the art in modern smartphones. To address this problem, the authors of this paper have re-designed the system's architecture in order to function through a client-server web-service architecture. Thus, all "heavy" and resource demanding operations were processed by the main server, while the mobile devices were handled as user interfaces for the educational interaction, under the precondition of a stable internet connection between clients and the server.

It was expected that younger students with an inherent tend towards new technology would welcome mobile learning adapted to their needs, supporting their learning. The findings of this preliminary study are rewarding the authors' attempts towards moving education to the fast growing field of mobile computing and specifically to mobile intelligent tutoring systems.

Analyzing the results of the evaluation study there is considerable evidence that new mobile technology is quite welcome from young learners and could be incorporated in schools supporting the educational process.

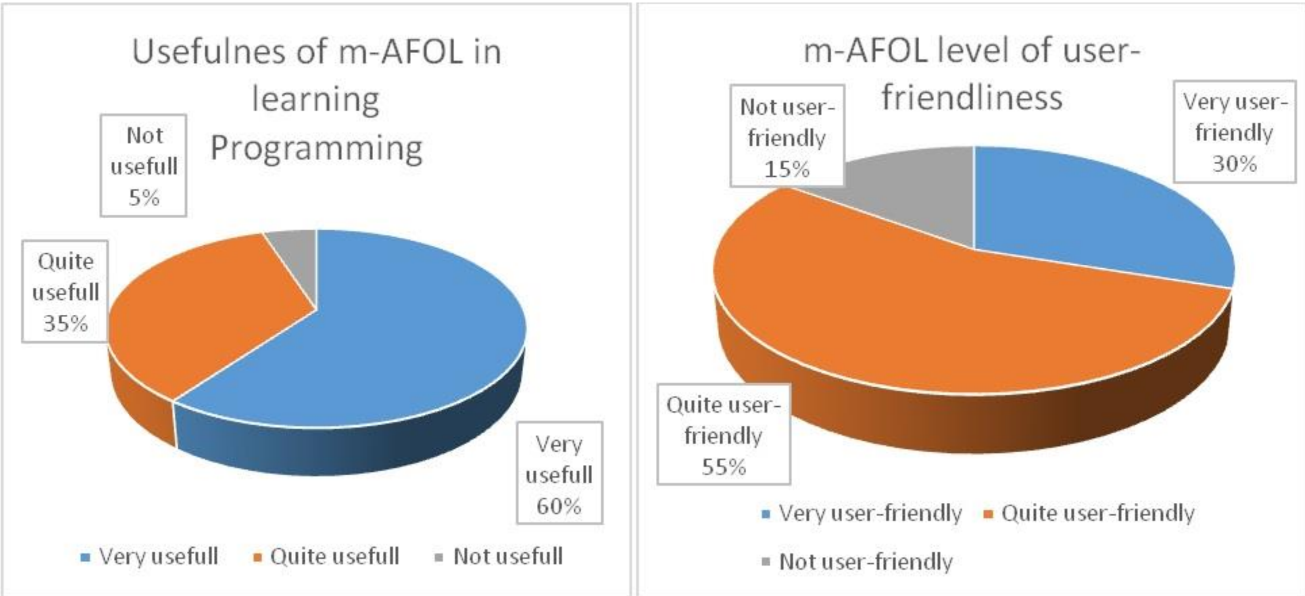

Figure 4: Usefulness and user-friendliness for the m-AFOL platform.

Figure 5: Comparative ranking between the two instances of the educational platform.

The evaluation results are illustrated in figures 4 and 5 respectively. More specifically, figure 4 illustrates the usefulness and user-friendliness of m-AFOL, while figure 5 provides a comparative illustration between the mobile and the desktop version of the intelligent tutoring system, concerning the questions of table 1.

Figure 7 illustrates the whole procedure from the use of both applications (mobile and desktop) until the experimental results. However, there are design and user interface issues that should be addressed carefully, since mobile devices do not seem to have all the advantages a desktop computer could provide. For example in "easiness in use" of the system, the desktop version of the platform was quite prevailing. The same is the case of navigation where the traditional keyboard and mouse seem quicker options in human-computer interaction. However, mobile applications are far from reaching maturity and their domain is enriched on a daily basis. Perhaps, designing applications that do not only extend desktop versions but rather utilize all mobile functionalities could produce

t-Test: Two-Sample Assuming Equal Variances					
Question 2			**Question 7**		
	Variable 1	*Variable 2*		*Variable 1*	*Variable 2*
Mean	8.5	7.3	**Mean**	8.7	7.5
Variance	2.055556	2.233333	**Variance**	0.9	3.388889
Observations	20	20	**Observations**	20	20
Pooled Variance	2.144444		**Pooled Variance**	2.144444	
Hypothesized Mean Difference	0		**Hypothesized Mean Difference**	0	
df	18		**df**	18	
t Stat	1.832352		**t Stat**	1.832352	
P(T<=t) one-tail	0.041748		**P(T<=t) one-tail**	0.041748	
t Critical one-tail	1.734064		**t Critical one-tail**	1.734064	
P(T<=t) two-tail	0.083497		**P(T<=t) two-tail**	0.083497	
t Critical two-tail	2.100922		**t Critical two-tail**	2.100922	
Question 8			**Question 12**		
	Variable 1	*Variable 2*		*Variable 1*	*Variable 2*
Mean	7.4	8.4	**Mean**	6.9	8.5
Variance	0.711111	0.488889	**Variance**	3.211111	1.388889
Observations	20	20	**Observations**	20	20
Pooled Variance	0.6		**Pooled Variance**	2.3	
Hypothesized Mean Difference	0		**Hypothesized Mean Difference**	0	
df	18		**df**	18	
t Stat	-2.88675		**t Stat**	-2.35907	
P(T<=t) one-tail	0.004911		**P(T<=t) one-tail**	0.014912	
t Critical one-tail	1.734064		**t Critical one-tail**	1.734064	
P(T<=t) two-tail	0.009822		**P(T<=t) two-tail**	0.029824	
t Critical two-tail	2.100922		**t Critical two-tail**	2.100922	

Figure 6: Statistical significance in a student's t-Test.

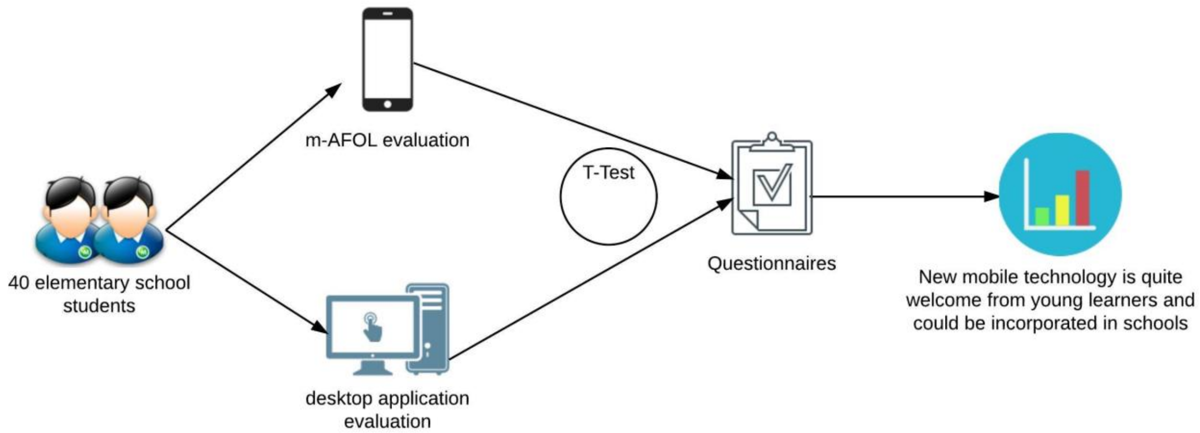

Figure 7: Experimental procedure.

more robust and sophisticated mobile systems. Modern mobile learning applications combined with their inherent portability and independence of time and place could be a great advance towards better and more natural computer assisted learning.

5 Conclusions and future work

In this paper, a mobile educational application for programming courses targeted to elementary school students has been presented and evaluated. The aim of this evaluation was twofold. First, we tried to measure the effectiveness of m-AFOL as an educational tool in a quite complicated domain of knowledge for young children (that of Object Oriented Programming). Secondly, we tried to compare the mobile platform with the same framework in a desktop computer interface. Analyzing the system's evaluation results gave us strong evidence that both students and also their instructors appreciated its functionalities and also approved it as an educational tool.

Mobile content in learning courses has been found quite friendly and attractive to younger students and also profitable for all potential learners.

Summarizing, the objectives of this paper is the presentation of the design of a mobile learning platform, named m-AFOL, for elementary school students so that they are taught basic programming principles and its evaluation. Furthermore, the primary goal of this study was the evaluation of the mobile programming platform in younger ages, namely children in elementary schools, given that students of this age are prone to new technologies. A contribution of the paper are the architecture of this platform being a sophisticated learning tool and incorporating affective interaction among learners and tutoring agents in order to maximize educational gains. Another contribution is that the paper presents a two-level evaluation study of m-AFOL, estimating its effectiveness and acceptance rate. The evaluation concludes that the mobile facilities of the resulting intelligent tutoring system are highly acceptable (over 80%) from young learners, being keen on using mobile devices while learning. Also, a quite interesting finding is that the majority of young children possessed mobile phones (2016-2017 findings) and know to use them.

Limitations and shortcomings can be the following. First of all, the population of the experimental group could include more than 40 elementary school students. Moreover, the evaluation stage may not let students interacting freely with the system and their behavior may not be the actual one. Finally, the mobile programming platform is targeted to the Android mobile operating system, given that it holds the greatest share of the global smartphone market; however, the mobile programming platform could be also targeted to the IOS mobile operating system and tablets or even smart televisions.

It is in our future plans to extend the system’s functionalities by incorporating an authoring module that will give teachers the opportunity to edit the lessons remotely and also monitor the progress of their students. In this way, all users will further benefit from time and place independence during interacting with m-AFOL. Furthermore, our future plans include the evaluation of this mobile authoring tool; as such, the mobile application will be fully functional.

References

[1] Alepis, E., Virvou, M. (2014). *Object-Oriented User Interfaces for Personalized Mobile Learning*, Intelligent Systems Reference Library Volume 64, Springer Berlin Heidelberg, pp. 109-125.

[2] Abdul Razak, F.H. , Salleh, K. , Azmi, N.H. (2013). *Children's technology: How do children want it?*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Volume 8237 LNCS, pp. 275-284.

[3] McKenney, S., Voogt, J. (2010). *Technology and young children: How 4-7 year olds perceive their own use of computers*, Computers in Human Behavior Volume 26, Issue 4, pp. 656-664.

[4] Jackson, L.A., Witt, E.A., Games, A.I. , Fitzgerald, H.E. , Von Eye, A. , Zhao, Y. (2012). *Information technology use and creativity: Findings from the children and technology project*, Computers in Human Behavior Volume 28, Issue 2, pp. 370-376.

[5] Clements, D., Nastasi, B. (1993). *Electronic media and early childhood education*, B. Spodek, editor, Handbook of research on the education of young children, Macmillan, New York, pp. 251–275.

[6] Fessakis, G., Gouli, E. , Mavroudi, E. (2013). Problem solving by 5-6 years old kindergarten children in a computer programming environment: A case study, Computers and Education Volume 63, pp. 87-97.

[7] Kordaki, M. (2010). A drawing and multi-representational computer environment for beginners' learning of programming using C: Design and pilot formative evaluation, Computers and Education Volume 54, Issue 1, pp. 69-87 (2010)

[8] Werner, L., Campe, S., Denner, J. (2012). *Children learning computer science concepts via Alice game-programming*, SIGCSE'12 - Proceedings of the 43rd ACM Technical Symposium on Computer Science Education, pp. 427-432.

[9] Virvou, M., Alepis, E., Mpalasis, K. (2013). *Evaluation of a multimedia educational tool for geography in elementary schools*, Proceedings of International Conference on Information Communication Technologies in Education, ICICTE, pp. 364-374.

[10] Shieh, C.-K., Mac, S.-C., Chang, T.-C., Lai, C.-M. (1996). *An object-oriented approach to develop software fault-tolerant mechanisms for parallel programming systems*, Journal of Systems and Software, Volume 32, Issue 3, pp. 215-225.

[11] Pastor, O., Gómez, J., Insfrán, E., Pelechano, V. (2001). The OO-Method approach for information systems modeling: From object-oriented conceptual modeling to automated programming, Information Systems, Vol 26, Issue 7, pp. 507-534.

[12] Frazier, F. (1967). *The logo system: Preliminary manual*, BBN Technical Report. Cambridge, MA: BBN Technologies.

[13] Feurzeig, W. (2010). *Toward a Culture of Creativity: A Personal Perspective on Logo's Early Years and Ongoing Potential*, International Journal of Computers for Mathematical Learning , Volume 15, Issue 3, pp 257-265.

[14] Goleman, D. (1995). *Emotional Intelligence*, Bantam Books, New York.

[15] Picard, R.,W., Klein, J. (2002). *Computers that Recognise and Respond to User Emotion: Theoretical and Practical Implications*, Interacting with Computers, Volume 14, Issue 2, pp. 141-169.

[16] Elliott, C., Rickel, J., Lester, J. (1999). Lifelike pedagogical agents and affective computing: an exploratory synthesis. In Wooldridge MJ, Veloso M (eds) Artificial intelligence today, LNCS 1600. Springer, Berlin, pp. 195–212.

[17] Johnson, W.L., Rickel, J., Lester, J. (2000). *Animated pedagogical agents: face-to-face interaction in interactive learning environments*, International Journal of Artificial Intelligence 11, pp. 47–78.

[18] Kahn, K. (1996). *ToonTalk - An Animated Programming Environment for Children*, Journal of Visual Languages and Computing, Vol. 7, No. 2, pp. 197-217.

[19] Alepis, E. (2011). *AFOL: Towards a new intelligent interactive programming language for children*, Springer, Smart Innovation, Systems and Technologies Volume 11, pp. 199-208.

[20] Troussas, C., Virvou, M., Alepis, E. (2014). Collaborative learning: Group interaction in an intelligent mobile-assisted multiple language learning system, Informatics in Education, 13 (2), 279-292. (a)

[21] Troussas, C., Alepis, E., Virvou, M. (2014). *Mobile authoring in a multiple language learning environment*, IISA 2014 - 5th International Conference on Information, Intelligence, Systems and Applications, art. no. 6878819, pp. 405-410. (b)

[22] Troussas, C., Virvou, M., Alepis, E. (2013). Comulang: Towards a collaborative e-learning system that supports student group modeling, SpringerPlus, vol. 2, no. 1, pp. 1-9.