A supporting framework for the creation of digital stories and learning programming by the students within Kodu, Scratch and Storytelling Alice

Panagiotis Psomos
Department of Cultural Technology & Communication
University of the Aegean, Mytilene, Greece
panagiotis.psomos@aegean.gr

Maria Kordaki
Department of Cultural Technology & Communication
University of the Aegean, Mytilene, Greece
m.kordaki@aegean.gr

Abstract: Introduction to programming is one of the main challenges of the didactics of computer science since programming is usually difficult for novice users. Different teaching approaches are studied aiming to stimulate the interest of novice programmers as well as to release students from the mental burden of memorizing syntax and grammatical rules. A teaching approach in this direction is learning basic programming structures through the creation of digital stories. Kodu, Scratch and Storytelling Alice are digital environments which support the creation of digital stories for the learning of programming. This paper presents a supporting framework for the creation of digital stories and learning of programming within the aforementioned digital storytelling environments. The aims of this framework are twofold: (a) to help students in their attempts to form the appropriate programs resulting these narratives and (b) to encourage the development of basic cognitive and metacognitive skills which may be developed through the creation of digital narratives by the students.

1. Introduction

Developments in computer science have led to rapid changes and progress in many different scientific and technical fields such as medicine, education and engineering (ACM, 2003). Moreover, given the broad impact of computer science in social and economic life, it is critical to ensure that bright minds will continue to be attracted by computer science education programs. To this end, learning programming is recognized that should be an essential part of general education of all students (Kelleher, 2006). However, novice programmers have difficulties in understanding the primary structures of programming, such as selection and repetition structures, and other basic concepts like variables (Robins, Rountree and Rountree, 2003; Eckerdal, 2009). At this point, it is worth noting that according to modern social and constructivist theories of learning (Piaget, 1952; Bruner, 1960; Vygotsky, 1978; Jonassen, 1999) the active participation of students in the learning process is an important positive factor. Students must be the protagonists of the educational process, while building their knowledge should be achieved through authentic activities that are inherently interesting to them (Waren et al., 2009). It is also important to establish a link between education and everyday experiences of students to bridge the gap between school and extracurricular world of students; informal and formal learning (McGiveney, 1999; Gee 2003). One way to achieve these conditions is the teaching of programming through digital storytelling.

Storytelling is treated as a cultural achievement of mankind (Spaniol, Klamma, Sharda and Jarke, 2006). It is a simple but yet powerful way with which students can be supported to understand the complex and unclassified world of experience (Bruner, 1990). According to Pedersen (1995) narrative is a genuine form of teaching. In recent years however, the narrative was enhanced with the help of digital technology giving traditional narrative a new dimension; digital storytelling. Thus, digital storytelling is the modern expression of the ancient art of storytelling and draws its power from the harmony between image, music, narration and voice (Lowenthal, 2008). When digital storytelling is used for educational purposes the result is Educational Digital

Storytelling (Figure 1). Through properly designed Educational Digital Storytelling activities the 6 cognitive objectives of the revised taxonomy of Bloom (Bloom, Mesia and Krathwohl, 1964): Recall, Comprehension, Application, Analysis, Evaluation, Synthesis and Creation can be achieved (Meerbaum-Salant, Armoni, & Ben-Ari, 2010). Moreover, according to Robin (2006), through digital storytelling students can grasp the following skills:

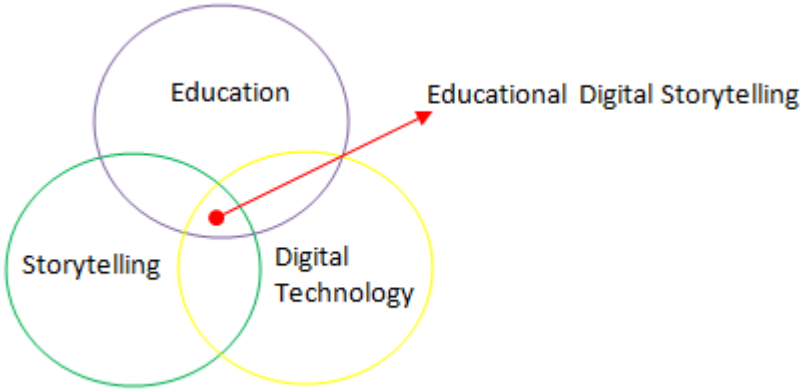

Figure 1: Educational Digital Storytelling

- *Research Skills*: Documenting the story, finding and analyzing pertinent information.
- *Writing Skills*: Formulating a point of view and developing a script;
- *Organization Skills*: Managing the scope of the project, the materials used and the time it takes to complete the task at hand;
- *Technology Skills*: learning to use a variety of tools, such as digital cameras, scanners, microphones and multimedia authoring software;
- *Presentation Skills*: Deciding how to best present the story to an audience;
- *Interview Skills*: Finding sources to interview and determining questions to ask;
- *Interpersonal Skills*: Working within a group and determining individual roles for group members;
- *Problem-Solving Skills*: Learning to make decisions and overcome obstacles at all stages of the project, from inception to completion; and
- *Assessment Skills*: Gaining expertise critiquing their own and others' work.

Furthermore, the creation of digital stories helps the smooth integration of students with physical disabilities or learning difficulties in the learning process, as they are encouraged to take an active role in their learning process (Di Blas & Boretti, 2009; Di Blas et.al., , 2010). Moreover, according to Kelleher (2007) the learning of programming through digital storytelling can attract girls aged between 11 and 15 years, thus potentially increasing the number and diversity of students interested in computer science.

In the next section, there will be a brief description of educational software Kodu, Scratch and Storytelling Alice with the help of which programming can be taught through digital storytelling. Finally, a framework which supports students in the creation of digital stories and learning programming using the aforementioned educational environments is presented. This is the contribution of this article.

2. Few words about Kodu, Scratch and Storytelling Alice

Kodu (http://research.microsoft.com/en-us/projects/kodu/), is a visual programming language which is used for the creation of digital games. It is easy to use and includes tools for creating three-dimensional worlds. Kodu is a multi-dimensional tool for digital storytelling with a variety of possibilities for creating digital stories. It is designed to be user friendly and accessible for children aged between 8-18 years. The creation of digital stories is made through the selection of appropriate characters and objects (e.g. character Kodu, trees, clouds, rocks etc.) that can be used in specific situations. Kodu helps children build a sound programming literacy without complicated programming concepts.

Storytelling Alice (Kelleher, et.al, 2006), is a programming environment that introduces students to computer programming through the construction of 3D animated stories. Its main age target group is between 10 and 17 year old children. It's a variant of Alice which is an object-oriented educational programming language. Its emphasis on storytelling is based on the following three differences from Alice: i) Social

interactions between the characters are possible through the programming of high-level animations. ii) Users are introduced to programming through building a story with the help of a story-based tutorial, and iii) A library with 3D characters and scenery is existent so as to stretch users' imagination.

Scratch (http://scratch.mit.edu/) is an educational programming environment designed from MIT in which novice programmers can express their creativity while promoting their computational thinking. Storytelling is a common use for Scratch as a method of personal expression. Students can create autobiographies, fairytales, family and vacation stories that give a new dimension to their interests and talents. Its target age group is between 6 to 16 year old children, but people of all ages can use Scratch.

3. A supporting framework for the creation of digital stories and learning programming within Kodu, Scratch and Storytelling Alice

Kodu, Scratch and Storytelling Alice can be used in several ways, however, an important category of activities is to ask students to create their own original digital stories. In this case, the goal is twofold: On one hand, students are enabled to experiment and use a variety of programming structures and on the other hand, students have the opportunity to foster multiple skills such as those reported in the previous section (Section, 1; Robin, 2006). Moreover, in the creation of a new digital story the 6 cognitive objectives of the revised taxonomy of Bloom (Recall, Comprehension, Application, Analysis, Evaluation and Creation) can be achieved. A supporting framework for the creation of digital stories and learning programming which aims to support students to develop the skills mentioned above will be presented below:

1. *Creation of the storyboard*:

As a first step students are asked to create a storyboard filling in the appropriate cells of the table shown in Figure 2. Storyboarding can be an effective tool for primary and middle school children to outline and refine their narratives. Storyboards consist of visual representations that help students to create digital stories and constitute a sequential arrangement of images that represents the flow of history. They may also include technical aspects and explanations of the design (Robin, 2006).

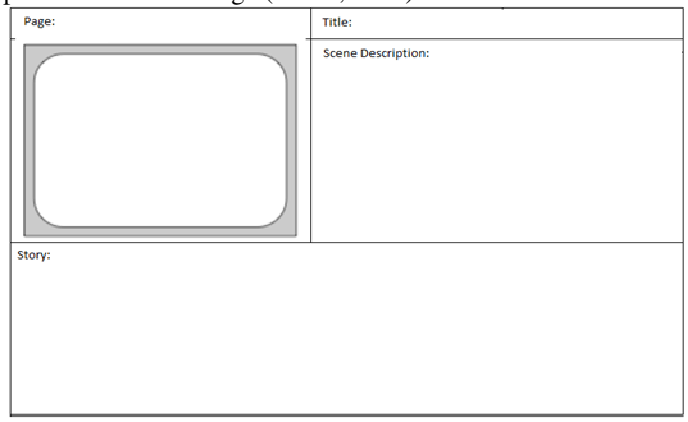

Figure 2: Storyboard

As it is shown in Figure 2, the student is required for each stage of the narrative to:
• draw a sketch
• list the steps that will be performed by the heroes of the story
• write in natural language the narration of each scene.
The storyboard can be used as a basis for the students' reflection in order to move smoothly to the next step which is the programming of the digital story.

2. *Programming of the digital story in either Kodu or Scratch or Storytelling Alice*:

Once the previous step is completed, the students should program in either Kodu or Scratch or Storytelling Alice the digital story described in the storyboard.

3. *Reflection and improvement of the program*:

During the programming of the digital story in the selected educational environment, it is helpful to

give students a framework in the form of self-assessment questions shown in Table 1 - with which they can reflect to their source code and improve their digital story. The questions intent: (a) to help students in their attempts to form the appropriate programs resulting these narratives, (b) to create a digital story in compliance with the basic principles of creating a narrative, and (c) to help students develop skills associated with narrative (Robin, 2006) and reported in the previous section (Section, 1).

4. *Presentation of the digital stories in class*:

After the creation of the digital stories, students are asked to present them in class. Afterwards there will be a discussion on the stories created which will be followed by the evaluation of the digital stories.

Table 1: A supporting framework for the creation of digital stories and learning programming within Kodu, Scratch and Storytelling Alice

	Yes	**Partially**	**I need help**
Organizational Skills:			
Structure: My story has a beginning, middle and end.			
Resource utilization and meeting of deadlines: I created my story making the most of the tools provided on time.			
***Writing Skills*:**			
Grammar: I confirm that the words, grammar, spelling and punctuation of dialogues are correct.			
Flow of the story: the story flows pleasantly, clearly and continuously highlighting important points.			
***Research Skills*:**			
Documentation: I looked at several sources to compose a documented story.			
Originality: My story is original compared to existing stories.			
Problem-Solving Skills:			
Analytical skills and decision making skills:			
Characters: The characters have distinct personalities, as demonstrated by what they say and what they do. *Analysis of the story*: My story is composed of discrete parts, each of which contributes with a specific way to the overall story			
Composition skills:			
Programming Structures: I used a variety of programming structures. *Composition of the story:* My story includes over three scenes.			
Contribution of the story in problem solving: My story can give solutions to the following problems ...			
***Technology Skills*:**			
Technology tools knowledge: I know to use the diversity of tools provided by the software (e.g. microphones).			
***Assessment Skills*:**			

Arguments supporting the quality of my story: The story that I made is very good because ...			
Evaluation of other students' stories: I settled on criteria for the assessment of my classmates´ stories.			
Interpersonal Skills:			
Role: I had a specific and pioneering role in the group			
Quality of Cooperation: I cooperated smoothly and contributed the best possible			
Presentation Skills:			
Structure: My presentation has a structure including: introduction, main body and a concluding summary.			
Content: I think that my presentation is complete, clear and interesting to the listener			

Finally, the same framework can be used with minor grammatical changes (changing the first person to third person) in the assessment of students' stories by the teacher. The proposed evaluation model can follow the 4-point Likert scale (1 - Very Good 2 - Good, 3 - Moderate, 4 - Troubleshooting). Of course, the reported supporting framework is not unique as it is based on a specific theoretical framework. Alternative theoretical learning approaches could also be used so that to enrich the set of the proposed self-assessment questions.

4. Summary and future plans

This study presented a supporting, theoretically based framework for the creation of digital stories and learning programming that can be used within Scratch, Kodu and Storytelling Alice which are digital environments that support the learning of programming by novices. This framework was created to support students in the planning, reflection and improvement of their digital stories towards the fulfillment of the following specific objectives: (a) to help students in their attempts to form the appropriate programs resulting these narratives, and (b) to encourage the development of basic cognitive and metacognitive skills which may be developed through the creation of digital narratives by the students.

In fact, the framework consists of 4 phases: (a) Creation of a storyboard, (b) Programming of the storyboard in either Kodu or Scratch or Storytelling Alice, (c) Reflection and improvement of the program, and (d) Presentation of the digital stories in front of the whole class. Appropriate, theoretically based tools supporting the creation of the storyboard were created, as well as reflection tools aiming at the improvement of the digital stories. These tools can be used not only by students for self assessment and improvement of their digital stories, but also by the teacher to assess students' digital stories. Despite the fact that, the tools created are theoretically based, research on their impact in students' attempts to create digital stories and learning programming is necessary to be performed within real classroom settings. Through such research field-studies, the enrichment of the proposed tools is also possible. This is part of our future research plans.

References

Association for Computing Machinery (2003). *A Model Curriculum for K-12 Computer Science: Final Report of the ACM K-12 Task Force Curriculum Committee*, http://www1.acm.org/education/k12/k1final1022.pdf

Bloom, B. S., Mesia, B. B., and Krathwohl, D. R. (1964). *Taxonomy of Educational Objectives* (2 volumes: The affective domain; The cognitive domain). David McKay.

Bruner, J. (1960). *The Process of Education*. Cambridge, MA: Harvard Univ. Press.

Bruner, J. (1990). *Acts of meaning*. Cambridge, MA: Harvard Univ. Press.

Brusilovsky, P., Calabrese, E., Hvorecky, J., Kouchnirenko, A., & Miller, P. (1997). Mini-languages: A way to learn programming principles. Education and Information Technologies, 2(1), 65-83.

Di Blas, N. and Boretti, B. (2009). Interactive storytelling in pre-school: a case-study. In *Proceedings of IDC 2009*, ACM, New York, NY (2009), pp. 44-51.

Di Blas, N., Paolini, P. and Sabiescu, A. (2010). Collective Digital Storytelling at School as a Whole-Class Interaction. In Proceedings of The *9th International Conference on Interaction Design and Children,* Barcelona, Spain, June 9-11, 2010, pp. 11-19.

Dorner, R., Grimm, P., & Abawi, D. (2002). Synergies between interactive training simulations and digital storytelling: a component-based framework. *Computers & Graphics*, 26, pp. 45–55.

Eckerdal, A. (2009). *Novice Programming Students' Learning of Concepts and Practise*. Dissertation presented at Mathematics and Computer Science, Dept of Information Technology, Upsalla University, Sweden, http://uu.diva-ortal.org/smash/record.jsf?pid=diva2: 173221

Gee, J. P. (2003). *What video games have to teach us about learning and literacy*. New York: Palgrave/Macmillan

Jonassen, D. H. (1999). Designing constructivist learning environments. *Instructional design theories and models*, 2, pp. 215-239.

McGivney, V. (1999). *Informal Learning in the Community: a trigger for change and development*. Leicester NIACE

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M (2010). Learning Computer Science Concepts with Scratch. *ICER 2010*, Aarhus, Denmark, pp. 69-76.

NetShelter. NetShelter (2004). Reaches Enterprise Decision Makers and IS/IT Professionals, Technology Enthusiasts, Consumers, and Hardcore Gamers. Thornhill, ON, Canada, 2004; www.netshelter.net/media_kit/audience/.

Pedersen, E. (1995). Storytelling and the art of teaching. *FORUM*, 33(1). http://eca.state.gov/forum/vols/vol33/no1/P2.htm, Retrieved 20/102011.

Piaget, J. P. (1952). *The origins of intelligence in children*. International Universities Press, New York.

Robin, B. R. (2006) *The Educational Uses of Digital Storytelling*. http://www.coe.uh.edu/digitalstorytelling/evaluation.htm

Robins, A., Rountree, J., and Rountree, N. (2003). Learning and Teaching Programming: A Review and Discussion. *Computer Science Education*, 13, 2, 137-172.

Spaniol, M., Klamma, R., Sharda, N., Jarke, M. (2006). Web-Based Learning with Non-linear Multimedia Stories. *ICWL 2006*: pp.249-263

Warren, S., Dondlinger, M. J., Stein, R., & Barab, S. (2009). Educational game as supplemental learning tool: Benefits, challenges, and tensions arising from use in an elementary school classroom. *JILR*, 20(4), 487–505.

Wing, J. (2006). "Computational Thinking.", *ACM*, 49(3): 33-35.

Vygotsky, L.S. (1978). *Mind in Society*. Cambridge, MA: Harvard University Press.