A 3D Learning Environment for Teaching Computational Thinking

José Mario Ríos Félix[1], Ramón Zatarain Cabada[1], María Lucia Barrón Estrada[1],
Yasmín Hernández Pérez[2]

[1] Tecnológico Nacional de Mexico,
Mexico

[2] Instituto Nacional de Electricidad y Energías Limpias,
Mexico

{mario_rios, rzatarain, lbarron}@itculiacan.edu.mx,
myhp@iie.org.mx

Abstract. Mexico is in the process of integrating Computational Thinking into the public educational curriculum. This paper presents a learning environment that encompasses the four main elements of computational thinking and, in turn, integrates emotional recognition and gamification techniques, with the purpose of optimizing interaction with students. A preliminary test was performed using a technology acceptance model, the result indicates a favorably perceived enjoyment with the application by the students. Nevertheless, experiments focused on analyzing cognitive learning will be carried out.

Keywords: computational thinking, emotional recognition, gamification, learning environment, problem solving, algorithms.

1 Introduction

In our current era, digital skills are necessary for most of the activities that people do every day. Whether you work as a teacher, engineer, doctor, artist, or entrepreneur, you need to interact with various technological means to be able to carry on the day to day. Information and communication technologies (ICTs) have a great impact on most areas of human activity, so companies require a trained workforce in the use of ICTs. It is essential that new generations possess the skills and knowledge that are essential to carry out their daily activities efficiently.

In different countries, initiatives have been created to include Computational Thinking (CT) topics in the basic education curriculum. CT is about formulating solutions to problems with sufficient clarity and, systematically enough, to tell a computer how to solve a specific problem [1, 2]. To integrate the teaching of digital skills into a wide range of educational levels, CT teaching initiatives have been channeled primarily by teaching children to program.

José Mario Ríos-Félix, Ramón Zatarain-Cabada, Maria Lucia Barron-Estrada, Yasmin Hernández

In Mexico, "@prende 2.0" was implemented at the end of 2016, which is a Digital Strategy in Education, prepared by the Ministry of Public Education [3]. This strategy seeks to encourage the use of Information and Communication Technologies to stimulate the development of digital and CT skills, necessary in the social and economic context of the 21st century. Currently this educational platform, covers different subjects (e.g. natural sciences, arts education, physical education, Spanish, civic and ethical training, geography, history, and mathematics) for fifth and sixth grade of primary school. This platform provides audios, thematic diagrams, documents, images, interactive resources and videos. However, these resources do not yet address CT issues. This represents an educational lag compared to other countries.

Currently, Mexico is in the process of implementing public policies for the integration of Computational Thinking in the public educational curriculum [4]. The proposal presented in this document is a learning tool for computational thinking, which relies on affective computing, to recognize emotions focused on learning and applies gamification techniques to motivate its use.

Affective Computing is a branch of Artificial Intelligence whose objective is to allow intelligent systems to recognize, feel, infer and interpret human emotions [5]. The recognition of emotions implies great challenges, due to the difficulties for the definition, and classification of emotional expressions for different people, without contextual or psychological information [6].

This paper is organized as follows. Section 2 deals with related works, emphasizing current initiatives that seek to teach computational thinking. Next, in section 3, the structure and operation of EasyLogic 3D (our learning tool) are explained. Then, in section 4, the experiments and results found are described. Finally, section 5 discusses about the conclusions and future work of our project.

2 Related Work

In many countries, various initiatives have emerged that seek to encourage the incursion of programming and CT in early stages of education. Some of the most important initiatives are mentioned below.

Code.org is a non-profit organization dedicated to expanding participation in computer science by making it available in more schools and increasing the participation of women and students of color with little representation [7]. This organization promotes "The Hour of Code", which is an introduction to one hour sessions of computer science education, designed to show that everyone can learn to program and thus understand the fundamentals of the discipline [8]. One of the great qualities of Code.org is the use of a learning management system, so it has a very well-structured scope and sequence of the puzzles within the platform.

Computational Thinking Initiatives [9], is a program of the Wolfram Foundation, whose motto is "Empowering computer generation." This initiative offers programs and resources to improve CT skills among students around the world. Its mission is to quickly train a broad spectrum of young people today, to become cutting-edge computational thinkers.

European Coding Initiative [10], with its slogan "All you need is to code", promotes coding and CT at all educational levels. It was created in June 2014 sponsored by the European Commission and involves non-profit organizations such as: CoderDojo, Code.org and European Schoolnet.

In general, the teaching of CT is dominated by an approach, in which students learn programming concepts (instruction structure, variables, cycles, data, among others) and the use of abstractions to formally represent the relationships between computing and real-world objects [11].

One of the global most used tools for learning programming concepts in general is Scratch [12]. Scratch is currently described as an authoring tool, since it allows the creation of interactive stories, games, musical compositions, cartoons and simulations. One of its great capabilities is the ability to import and edit images to integrate them into your project.

There are also proposals that focus on construction-based robotics activities [13]–[16], ranging from simulation of robotic environments, to direct robot programming. While the acquisition of robotics equipment can be expensive, they also have many advantages, as they provide users with immediate and concrete information about the effectiveness of their programs, which promotes self-analysis and facilitates debugging.

3   EasyLogic 3D: A Learning Environment

The objective of this research project is to develop an intelligent learning environment, aimed at teaching computational thinking, using cutting-edge techniques such as the recognition of emotions focused on learning and motivational techniques used in education. In general terms, the learning environment is a web platform. To make the tool interesting for students, a 3D graphic engine was integrated, with the purpose of involving the students, so that they had the impression that it is a game, when they are really learning.

The research method used consists of 4 stages: Analysis, Design, Implementation, and Testing. The hypothesis that is sought to clarify is: the students who learn using our tool show greater commitment and motivation in their interaction, compared to students who learn the same concepts through other traditional methods.

3.1   Architecture

The general structure of the system uses a layered architectural pattern known as Model-View-Controller (MVC). This pattern separates the data, the business logic and the way in which the information will be represented to the user within the application. Figure 1 shows the architectural design for the application: in view section, multiple modules are observed, each one for a specific task within the application. In Controller section, each element interacts directly with assistance and *gamification module*. The *assistance module* brings aid to students while using the application depending on user usage and emotion data, likewise, the *gamification module* seeks to keep students
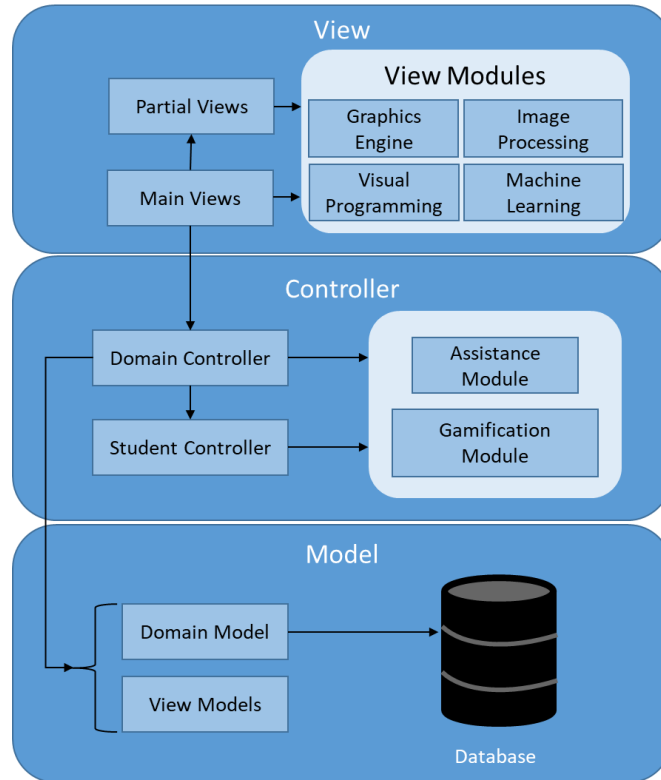
José Mario Ríos-Félix, Ramón Zatarain-Cabada, Maria Lucia Barron-Estrada, Yasmin Hernández

Fig. 1. Architectural design of the application.

motivated to continue using the application, through motivational messages, progress information, among others. Detailed explanation is presented below.

3.2   Assistance Module

The Assistance module is an independent module that analyses user usage data, in this way, determines if they require an intervention during their learning process. The data used by the algorithm consist of three elements: Time spent in the current exercise, number of executions, and student emotion.

Time spent and number of executions, correspond to real-time information, which is obtained while the user solves some exercise in the application. The emotion of the user is historical information of each of the students, previously stored in the database.

The last n emotions (configurable value) registered by the user are used to be processed by the classification algorithm. The current version of this module stablishes a minimum time of 30 seconds, and a minimum between 0 to 2 executions (depending the exercise), before being able to display an intervention. Figure 2 shows the algorithm implemented to decide the type of intervention that will be shown.
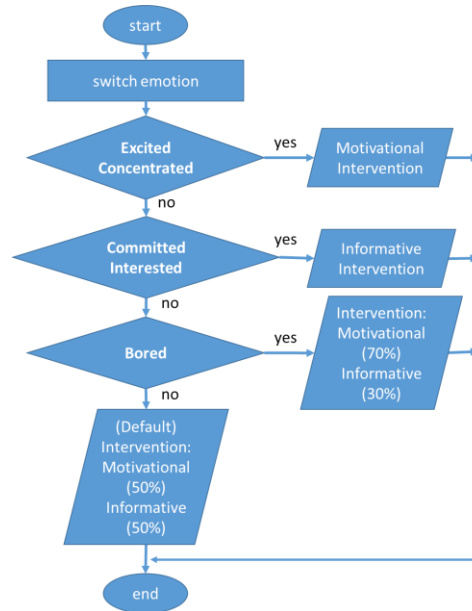
Fig. 2. Logic used to choose the type of intervention to be displayed.

3.3 Gamification Module

The gamification module integrates elements that are commonly found in games to the teaching tool, with the purpose of motivating students to continue using the tool. The following elements have been integrated: points, trophies, levels and progress bar:

- − Points. For each exercise performed, the points obtained by the student are calculated, using an algorithm that calculates based on the base value of each exercise (depending on its difficulty), number of errors made, and the number of executions required to solve the current exercise.
- − Trophies. A series of trophies are included, which are acquired by accumulating points within the tool.
- − Levels. Exercises are classified depending on the level of difficulty of these, which implies a variation in the base value of points obtained for each exercise.
- − Progress bar. An interface to be able to visualize the progress that the user has obtained.

The gamification module is used to obtain information about the current score of each user. In addition, it is responsible for the generation of motivational interventions, based on information about trophies that a user is about to reach, or through motivational phrases.
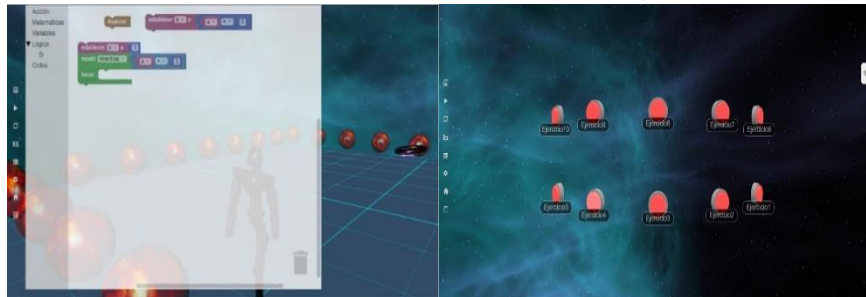
José Mario Ríos-Félix, Ramón Zatarain-Cabada, Maria Lucia Barron-Estrada, Yasmin Hernández

Fig. 3. Graphic interface of the educational environment.

3.4 Image Processing Module

For image processing OpenCV.js [17] was used, a link to a subset of OpenCV functions, suitable for the web platform. Face images from students were extracted, through a face recognition classifier, using a pretrained Haar cascade model, which is one of the most effective methods for this task.

3.5 Machine Learning Module

For the recognition of educational related emotions, a machine learning model was used, which is a variant of a previous work [18], with an accuracy of 82%. This model can recognize 5 different classes: Boring, Committed, Excited, Concentrated, and Interested. The model was trained using a corpus of spontaneous facial expressions, which were obtained from the observation of electroencephalographic (EEG) signals [19]. The five emotions that are recognized by this model are focused on learning, which allows us to use this information, to manipulate in real time the interaction of the application with the user.

3.6 Visual Programming Module

Blockly [20] was used as the visual programming block editor, to ease the creation of algorithms within the application. Blockly is an extensible and configurable JavaScript library created by Google.

3.7 Graphical Application Interface

As mentioned earlier, the system uses various components that interact with each other, which together form a CT teaching environment. Fig. 3 presents the graphic interface of the learning environment, at the left side, an interface for selection of the exercises is presented. On the right side of Fig. 3, a 3D world composed of a platform (grid of positions), the main character (Cyborg), barriers to limit movements, and a goal (objective) is presented for solving algorithm learning exercises.

Fig. 4. Testing session with public school students.

The exercises gradually increase their difficulty, where the following programming concepts are covered: sequential structures, variables, conditional structures and finally repetitive structures.

4 Experiments and Results

Preliminary tests were applied to validate the correct operation of the application, as well as other aspects related to its usability.

4.1 Usability Testing with Experts

These tests were applied first to postgraduate students to get feedback and thus improve the application. To carry out these tests, 14 graduate students in the area of computer science were asked to use the tool. The process consisted of performing 10 exercises within the application, and then answering a brief survey, so that it could serve as feedback.

4.2 Usability Testing with Students

Because the tool is aimed to K5 and K9 students, usability tests were performed with representative users of sixth year of elementary school and first year of high school. The objective was to identify usability problems and collect qualitative and quantitative data, and in this way determine areas for performance improvement and user experience. Figure 4 shows at the right side, students from ETI 75 high school in our city (Culiacán, Sinaloa), who used the application from desktop computers. On the left side, students from the Natalio Landeros Ramos elementary school, also located in Culiacán, who used the application from portable tablets.

After using the tool, students were asked to answer a brief survey to understand their perception of the system. Using the Technological Acceptance Model (TAM) [21], survey results were analyzed to measure perceived usability, perceived ease of use, intended use, among others. The results obtained from this survey were favorable in the aspect perceived enjoyment with the application ($\alpha > 0.6$), and not so favorable in the ease of use, intention of use and attitude towards the use of the tool ($\alpha < 0.6$).

5 Conclusions and Future Work

Computational Thinking is a cognitive process, which implies logical reasoning. The various initiatives related to the CT that exist globally seek to train new generations, so that they obtain the ability to formulate solutions to the problems that will be presented to them, and that will have to be solved using future technologies. Mexico has an educational delay compared to other countries, in relation to the incursion of CT, which must be addressed in the coming years.

In this work, a modern learning environment that integrates emotion recognition and gamification has been presented. This work represents an educational proposal aimed to basic education in Mexico, which would seek to update the national curriculum and thus be integrated into the curricula established by the educational institutions of Mexico.

Preliminary tests carried out indicate favorable results with respect to the perceived enjoyment with the application by the students, which indicates that, in general, the integration of all the elements included in the tool was successful. However, these same preliminary tests indicated some areas of opportunity to continue improving the learning tool.

As future work, multiple tests to compare our tool against other learning methods will be performed, and thus obtain information on the effectiveness of our tool to evaluate commitment and motivation presented during the interaction with the learning environment. It is also expected to develop different machine learning models, seeking to obtain greater precision in the emotional recognition of students.

References

1. Csizmadia, A., Dorling, M., Curzon, P., Humphreys, S.: Computational thinking - A guide for teachers. Research Gate (2015).
2. Zapata-Ros, M.: Pensamiento computacional: Una nueva alfabetización digital. Revista de Educación a Distancia, 46 (2015). DOI: 10.6018/red/46/4.
3. Gómez-Navarro, D.A., Alvarado-López, R.A., Martínez-Domínguez, M., Díaz de León-Castañeda, C.: La brecha digital: una revisión conceptual y aportaciones metodológicas para su estudio de México, Entreciencias Diálogos en la Sociedad del Conocimiento, 6(16) (2018).
4. Cárdenas-Peralta, M.C.: Pensamiento Computacional educación básica. Aprende 2.0, pp. 1–56, (2018).
5. Poria, S., Cambria, E., Bajpai, R., Hussain, A.: A review of affective computing: From unimodal analysis to multimodal fusion. Fusion, 37, pp. 98–125 (2017).
6. Fan,Y., Lu, X., Li, D., Liu, Y.: Video-Based Emotion Recognition using CNN-RNN and C3D Hybrid Networks (2016).
7. Code.org: What will you create? [Online] https://code.org/ (2019).
8. Vogel, S.: Power, Discourse, and Knowledge in Computer Science Education Advocacy: An Analysis of Popular Code.org Videos. pp. 32–117, (2016). DOI: 10.31235/osf.io/rz9gw.
9. Wolfram Foundation: Computational Thinking Initiatives. [Online] https://www.computationinitiative.org/ (2019).
10. Balanskat, A., Engelhardt, K.: Computing our future. pp. 87 (2015).
11. Aho, A.V.: Computation and computational thinking. Comput. J., vol. 55, no. 7, pp. 833–835 (2011). DOI: 10.1093/comjnl/bxs074.

12. Brennan, K., Resnick, M.: New frameworks for studying and assessing the development of computational thinking. Annu. Am. Educ. Res. Assoc. Meet. Vancouver, Canada, pp. 1–25 (2012).
13. Sullivan, F.R., Heffernan, J.: Robotic construction kits as computational manipulatives for learning in the STEM disciplines. J. Res. Technol. Educ., 48(2), pp. 105–128 (2016).
14. Falk, G., Mannock, K.L.: First steps in creative computational thinking with natural language programming and Lego M INDSTORMS. pp. 10–16 (2018).
15. Bers, M.U., Flannery, L., Kazakoff, E.R., Sullivan, A.: Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. Comput. Educ., 72, pp. 145–157 (2014).
16. Rode, J.A., von Rekowski, T., Weibert, A., Booker J.: From computational thinking to computational making. Proc. ACM Int. Jt. Conf. Pervasive Ubiquitous Comput., pp. 401–402 (2015).
17. Taheri, S., Veidenbaum, A., Nicolau, A., Hu, N.: OpenCV: Computer Vision Processing for the Web (2018). DOI: 10.1145/3204949.3208126.
18. González-Hernández, F., Zatarain-Cabada, R., Barrón-Estrada, M.L., Rodríguez-Rangel, H.: Recognition of learning-centered emotions using a convolutional neural network. J. Intell. Fuzzy Syst., 34(5), pp. 3325–3336 (2018).
19. Zatarain-Cabada, R., Barron-Estrada, M.L., Leal-Hernandez, D., Rios-Felix, J.M.: Creation of a Facial Expression Corpus from EEG Signals for Learning Centered Emotions. Proc. - IEEE 17th Int. Conf. Adv. Learn. Technol. (ICALT´17), pp. 387–390 (2017).
20. Fraser, N.: Google blockly-a visual programming editor. [Online] https://developers.google.com/blockly/ (2016).
21. Granic, A., Marangunic, N.: Technology acceptance model : a literature review from 1986 to 2013. pp. 81–95 (2015).