Shih, JL. et al. (Eds.) (2023). Proceedings of the 31st International Conference on Computers in Education. Asia-Pacific Society for Computers in Education

Gamified Learning Objects for Inclusive Programming and Science Education

Saumay GARG[a], Seema MITTAL[b] & Mukta GOYAL[a*]
[a]Department of Computer Science and Information Technology, Jaypee Institute of Information Technology, India
[b]Department of Electronics and Communication Engineering, Galgotias University, India
*mukta.goyal@jiit.ac.in

Abstract: Children from low-income communities face additional barriers to science and computing education, including limited computer access, language barriers, and a lack of resources for experimentation, often while being out of school. Computer-based learning resources that address these challenges in their design have the potential to make STEM education accessible to a whole new section of society. This paper presents a gamified learning object and a supporting visual programming language (VPL) that allows users to program and execute science experiments in the form of a narrative-driven problem-solving exercise. This tool is primarily intended for children from economically disadvantaged backgrounds.

Keywords: STEM education, Introductory programming, Social-emotional learning

1. Introduction

Gamified computer applications have been extensively explored for enhancing the quality of instruction and engagement of students in introductory programming and science education. Unfortunately, children from underserved communities face unique barriers to accessing science, technology, engineering, and mathematics (STEM) education and are often excluded from such initiatives. Children from low-income families in developing countries may not attend school and get forced into physical labor, street vending, or begging, which leaves no time for learning. And even if they do attend school, they often lack proper equipment and competent faculty which prevents them from meaningfully engaging in STEM education interventions. In absence of government support, individual instruction via computing devices is an option, but is limited by access to network and devices. Furthermore, unstable economic conditions and social stigma may lead to persistent stress which hinders the development of a healthy social-emotional skill-set.

In light of the above limitations, we present a gamified inquiry-driven learning object to teach science concepts, basic coding, and social-emotional learning (SEL). Combining these goals into one app reduces intervention duration. Science inquiry games help lay the foundations for scientific thinking, and SEL is emphasized because it helps improve student performance and emotional well-being (McClelland et al, 2017). Also, block-based programming languages like Scratch have been shown to help young children develop problem-solving and computational thinking skills with minimal cognitive load (Resnick et al, 2009; Pelánek et al, 2022). We address the following research gaps in this paper: (1) Though a game unifying SEL and Physics education goals was proposed by Aleven et al (2013), tools that teach science, SEL, and programming skills through a singular interface don’t exist; (2) Programming platforms like CODE.org (Kalelioğlu et al, 2015) focus on small puzzles that exist as one-off activities that are not adaptable to the local curriculum and lack the scalability to encompass much of the existing STEM curriculum; and (3) Educational interfaces for introductory programming rarely consider the unique challenges of children from low-income communities in their design.

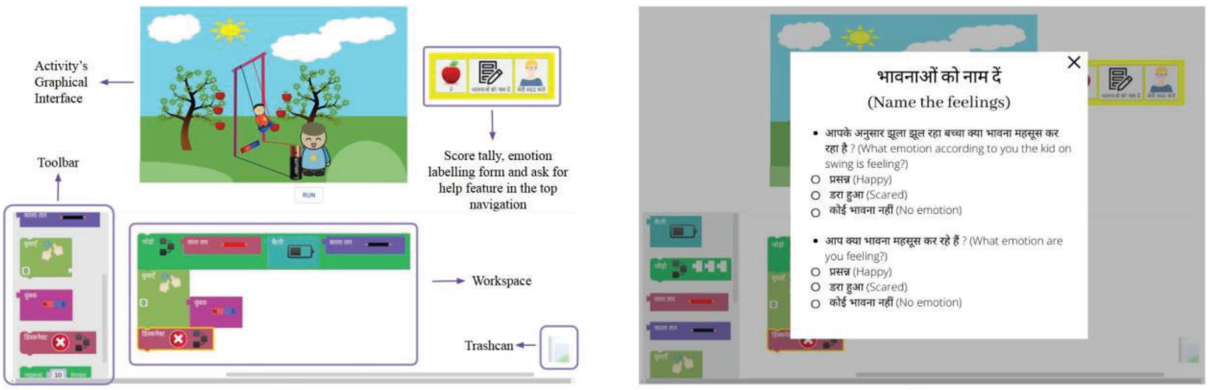
2. Learning Object: Design and Implementation

The learning object includes a toolbar containing the activity blocks, a workspace and an SEL support toolbar; see Figure 1a for the outline. It can also be replicated offline with minimal resources and supports the following objectives:

Support for scientific inquiry: Developing scientific inquiry skills requires repetitive experimentation and real-time feedback. The activity's design is guided by the National Science Education Standards (Aleven et al, 2015) and encourages children to predict, test hypotheses, explain observations, and revise theories based on feedback from the interface.

Support for block programming: Blocks in the activity support the text-free design paradigm with supporting text in a local language (Hindi). The use of icons helps make the exercise more accessible to children who lack adequate literacy skills and extend its accessibility for younger students who might rely solely on visual cues. Some blocks, as shown in Table 1, use parameters to reduce ambiguity by limiting options to a reasonable set.

Support for social-emotional learning: The activity targets three primary SEL goals for elementary students: cooperation and empathy, self-awareness, and help-seeking behavior. It involves a human player and a non-player character (NPC); the player must respect the emotions of the NPC in order to successfully complete the game and earn maximum points. The NPC would express discomfort when under undesirable predicament and leave the game leading to a two-point deduction from the user score. This serves to reinforce the need for cooperation and showcasing empathy towards the NPC. Based on this understanding, the child would be prompted with a short quiz for labeling emotions of self and the NPC (Figure 1b). Additionally, help-seeking behavior is encouraged through tooltip (dialogue box appears on the hover event on each block), a help button (plays a short video explaining the activity setup), and trashcan feature that maintains a history of blocks used for reference during gameplay.



(a) (b)

Figure 1. a) Layout of the proposed interface b) Self-reflection form for SEL support.

The game is set in a playground with a swing and an electromagnetic setup inspired by Gupta (2015). The NPC sits on an iron swing which moves due to electromagnetic forces according to the Fleming's left-hand rule. The player manipulates the swing by altering elements in the virtual setup (like polarity and orientation of magnetic field and electric current). One end of the wires remains connected to the swing while the other can be periodically attached to the battery, this induces a magnetic field around the swing. A magnet beneath it interacts with this field, rocking the swing so the NPC can collect apples from the trees. Being swung too high upsets the NPC which prompts them to leave the game. Therefore, the player would have to balance self-interest and regard for the NPC's emotions in order to gain maximum points. Players learn the Fleming's left-hand rule by continually developing and refining hypotheses about the virtual environment through experimentation as part of the scientific enquiry process. The blocks are designed to teach some basic concepts of programming like sequential execution, functions with parameters (*rotate*, *increase* and *connect* blocks), and loops. It also illustrates semantic errors and type compatibility, e.g., a battery can't substitute for a wire in the connect block. See Table 1 for a detailed description of blocks in the VPL.

We propose the deployment of this activity as a web application so it could be accessed on a wide range of devices and operating systems without the need to install any new software. Additionally, web applications can be conveniently packaged for offline usage. Our implementation used the React library while the graphical interface was made up of Scalable Vector Graphics (SVG) with Cascading Style Sheet (CSS) transitions for animations. The VPL and coding workspace were developed using the Blockly library.

Table 1. *Custom blocks in the visual programming language*

Block Image	***Block Name***	***Description***	***Block Image***	***Block Name***	***Description***
	Battery	Block representing the battery object.		Rotate (battery)	Takes the battery object as argument and rotates it with the angle provided by the player. The values accepted are 90, 180 and 270 degrees.
	Connect	Used to connect wires to battery, the arguments for the connect block are type sensitive (wires can only occupy the extreme positions).		Increase (value)	Increases the current or magnetic field of the object attached. Only allows the battery or magnet blocks to be connected.
	Black Wire	Block representing the wire (black) object. Used to describe the direction of the current when connected to a battery object using the connect block.		Red Wire	Block representing the wire (red) object.
	Magnet	Block representing the magnet object. It can be passed to the 'rotate' or 'increase' block as an argument.		Disconnect	The current flow stops when the wires and battery are disconnected, returning the swing to its initial position.

3. Conclusion and Future Work

This paper presented design considerations, implementation and deployment suggestions for a novel interdisciplinary educational activity supported by a small visual programming language. Though the presented approach could aid in making STEM education more accessible to children from low-income families, there are several questions that need further investigation. The proposed approach must be tested with participants in order to verify its efficacy; questions about cognitive load, perceptions formed, and complexity of code generated should be evaluated. Furthermore, it is essential to fully explore the implications of such intervention on the community's perception of formal schooling and ethics of child employment. Future research could also explore unsupervised joint media engagement (JME) and the potential of parental collaboration for temporarily replacing actual teachers during times of crisis when access to trained educators might be hampered.

References

Aleven, V., Dow, S., Christel, M., Stevens, S., Rosé, C., Koedinger, K., & Zhang, X. (2013). Supporting social-emotional development in collaborative inquiry games for K-3 science learning. In *Proceedings of the 9th Games+ Learning+ Society Conference-GLS* (Vol. 9).

Gupta, A. (2015). *Wonders from Waste*. Scholastic.

Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code. org. *Computers in Human Behavior*, 52, 200-210.

McClelland, M. M., Tominey, S. L., Schmitt, S. A., & Duncan, R. (2017). SEL interventions in early childhood. *The Future of Children*, 33-47.

Pelánek, R., & Effenberger, T. (2022). Design and analysis of microworlds and puzzles for block-based programming. *Computer Science Education*, 32(1), 66-104.

Perez-Escolar, M., & Canet, F. (2023). Research on vulnerable people and digital inclusion: toward a consolidated taxonomical framework. *Universal Access in the Information Society*, 22(3), 1059-1072.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., ... & Kafai, Y. (2009). Scratch: programming for all. *Communications of the ACM*, 52(11), 60-67.