Designing Block-Based Programming Language Features to Support Upper Elementary Students in Creating Interactive Science Narratives

Andy Smith[1], Bradford Mott[1], Sandra Taylor[1], Aleata Hubbard Cheuoua[2],
James Minogue[1], Kevin Oliver[1], and Cathy Ringstaff[2]

[1]North Carolina State University, Raleigh, North Carolina
{pmsmith4, bwmott, smtayl23, jminogu, kmoliver}@ncsu.edu

[2]WestEd, Redwood City, California
{ahubbar, cringst}@wested.org

ABSTRACT

Recent years have seen a growing recognition of the importance of enabling K-12 students to engage in computational thinking, particularly in elementary grades where students' dispositions toward STEM are developing. Block-based programming has emerged as an effective tool for engaging these novice learners in computational thinking. At the same time, digital storytelling has emerged as a promising avenue for creating motivating problem-solving scenarios that engage students in science investigations. Although block-based programming and digital storytelling are in many ways synergistic, there is a lingering question of how to design block-based languages at an age-appropriate level to enable effective and engaging storytelling. In this work, we review design principles from prior block-based and digital storytelling systems as well as propose the design of block-based programming language features to enable the creation of rich, interactive science narratives by upper elementary students.

KEYWORDS

Digital storytelling; Block-based programming

1 INTRODUCTION

Digital storytelling, which combines traditional storytelling with rich digital technologies, has emerged as an innovative approach for engaging students in deep, meaningful learning [1, 2]. By supporting students in creating interactive science-focused narratives using block-based programming, students are afforded the opportunity to deepen their conceptual understandings of science, while also developing computational thinking skills.

2 OVERVIEW

Creating rich block-based programming environments to support novices has seen success in bringing computational thinking to K-12 education [3]. To support the creation of computationally-rich digital stories, we present a set of custom blocks appropriate for upper elementary students. These blocks seek to enable the creation of interactive narratives, such as supporting character dialogue and orchestrating story events (Figure 1), while at the same time adhering to design principles identified in prior research on block-based programming for elementary students [4]. This includes avoiding difficult concepts such as initializing variables and properties as well as focusing on sequential programming rather than parallel event-based programming [5].
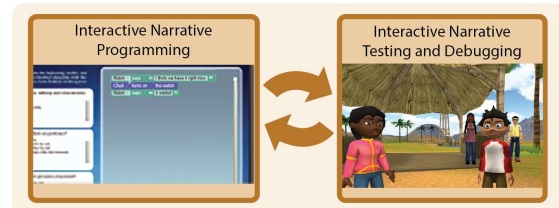

Figure 1: Interactive Narrative Development

3 CONCLUSION

Combining block-based programming and digital storytelling provides significant promise for engaging upper elementary students in computational thinking. In this poster, we present a set of custom blocks and design principles to enable students to produce rich, interactive stories that integrate physical science concepts.

REFERENCES

[1] B. Robin. 2008. Digital storytelling: A powerful technology tool for the 21st century classroom. *Theory into practice*, 47(3), 220–228.

[2] Y. Yang and W. Wu. 2012. Digital storytelling for enhancing student academic achievement, critical thinking, and learning motivation: A year-long experimental study. *Computers & Education*, 59(2), 339–352.

[3] A. Repenning, A. Basawapatna, and N. Escherle. 2016. Computational thinking tools. In *Proceedings of the 2016 IEEE Symposium on Visual Languages and Human-Centric Computing. IEEE*, 218-222.

[4] D. Weintrop, A. K. Hansen, D. B. Harlow, and D. Franklin. 2018. In *Proceedings of the 2018 ACM Conference on International Computing Education Research* (ICER '18). ACM, New York, NY, USA, 21-29.

[5] D. Franklin, G. Skifstad, R. Rolock, I. Mehrotra, V. Ding, A. Hansen, D. Weintrop, and D. Harlow. 2017. Using upper-elementary student performance to understand conceptual sequencing in a blocks-based curriculum. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education. ACM Press*, 231-236.