Enhancing Introductory Programming with Kodu Game Lab in a High School classroom

Allan Fowler, Waiariki Institute of Technology, Rotorua, New Zealand, allan.fowler@waiariki.ac.nz

Abstract: Kodu Game Lab is a tile based visual programming tool that engages students through playing and developing computer games to learning programming concepts. A series of exploratory investigative studies was administered to establish if student engagement with programming was enhanced through using Kodu Game Lab.

Introduction

Engaging mainstream students in introductory programming lessons is a great challenge for Information and Communication Technologies (ICT) teachers (Guzdial and Soloway, 2002; Wiedenbeck, 2005). Guzdial and Soloway (2002) suggest that engaging students is critical to deep learning. With the advent of object-orientated programming languages like Alice (Dann, et al., 2009), Scratch (Meerbaum-Salant, et al., 2010), Game Maker (Overmars, 2004) and Kodu Game Lab (MacLaurin, 2009), there has been an increased interest in the value of these tools to see if and how these tools can improve student engagement. Moreover, these tools also represent the potential to increase student interest in computer science (Guzdial and Soloway, 2002). A computer game development tool presents students the opportunity to develop their own worlds rich in visual and auditory interactive content, thus inspiring and motivating students to create their own designs (Lawhead, et al., 2003).

The Study

To evaluate the effectiveness of using Kodu Game Lab to introduce the programming concepts, an experiment involving students was conducted on a Year 9 ICT Literacy class in a New Zealand High School. The goal of this study was to measure the levels of engagement, enjoyment, and how much fun students felt they had while using this tool. The study was integrated into the existing class timetable and therefore the students had no choice in learning the materials (but could elect not to take part in the study). The students would receive one one-hour class per week for four weeks. As exploratory research this study was limited to 'what people said' and the option of developing and standardizing additional metrics was left open.

The Participants

The study involved 19 participants aged between 14 and 15 years old. Of the total population, 68% were male and 32% were female. The majority (95%) of the class identified as being European/Pakeha (Caucasian) and the remainder identified as being Maori.

Methods

To understand the students' perceptions about programming, programmers and obtain demographic data students were asked to participate in a pre-exposure survey. To collect the perceptual and demographic data a five-point Likert scale was used. The Likert scale provided a very positive (5) response, a very positive response (5), a positive response (4), a neutral response (3), a negative response (2), and a very negative response (1).

Throughout the study the students underwent continuous structured observation by the class teacher and levels of engagement, collaboration, and peer teaching were observed—as were levels of boredom and frustration during each lesson. To ensure that the observed behavior was a result of the lesson and not other mitigating factors, the class teacher also provided a rating of any external factors. To collect the data the class teachers were given guidance and a class observation form with a three point Likert scale to measure each observed behavior for each student. The observation sheet included the definition and attributes of each behavior for observation.

Data Analysis

The students (n=19) reported moderate to high levels of enjoyment with 12% of students indicating high levels of enjoyment and 12% indicating they enjoyed the lessons. This represented a collective 32% of the students reporting an enjoyable experience.

The cumulative levels of observed enjoyment and engagement behaviors were moderately high. Table 1 shows the results of the observed enjoyment and engagement behaviors. Conversely, the levels of observed boredom and frustration by the class teacher were not high. Table 2 shows the results of the observed boredom and observed frustration behaviors.

Rating	Observed enjoyment %	Observed engagement %
1 (low)	3	6
3	29	9
5 (high)	71	89

***Table 1*: Observed Student Behaviors – Enjoyment and Engagement.**

Rating	Observed boredom %	Observed frustration %
1 (low)	75	69
3	22	25
5 (high)	3	6

***Table 2*: Observed Student Behaviors – Boredom and Frustration.**

The levels of collaboration and peer teaching were also observed. Table 3 demonstrates that both the amount of collaboration was high and peer teaching was moderately high.

Rating	Observed Collaboration %	Observed peer teaching %
1 (low)	3	6
3	57	60
5 (high)	40	34

***Table 3*: Observed Student Behaviors – Collaboration and Peer Teaching.**

While acknowledging the limitations imposed by the design and execution of the exploratory study the measureable changes suggest that an object orientated education tool to introduce programming concepts to High School students provides motivational value. The concept fun was used by students to express satisfying learning experiences in a routine school class.

References

Dann, W., Cooper, S. and Pausch, R. (2009). Learning to program with Alice, Second Edition, Upper Saddle River, NJ: Pearson.

Guzdial, M. and Soloway, E. (2002). Teaching the Nintendo generation to program. *Communications of the ACM*, 45, 17-21.

Lawhead, P. B., Duncan, M. E., Bland C. G., Goldweber, M., Schep, M., Barnes D. J. and Hollingsworth, R. G. (2003). A road map for teaching introductory programming using LEGO(c) Mindstorms robots, inroads, The SIGCSE Bulletin, 35, 191-201.

MacLaurin, M. (2009). Kodu: end-user programming and design for games. *Proceedings of the 4th International Conference on Foundations of Digital Games*, ACM New York, NY.

Meerbaum-Salant, O., Armoni, M. and Ben-Ari, M. (2010). Learning computer science concepts with scratch. *Proceedings of the Sixth international workshop on computing education research*. ACM, New York, NY, USA, 69-76.

Overmars, M. (2004). Teaching computer science through game design, *Computer*, 37, 81-83.

Wiedenbeck, S. (2005). Factors affecting the success of non-majors in learning to program. *Proceedings of the first international workshop on computing education research*, ACM, New York, NY, USA, 13-24.

Acknowledgements

This research was conducted under the auspices of the Waiariki Institute of Technology Research Committee ethical standards. The author would like to acknowledge the support and assistance of Microsoft Research's FUSE Labs. This research was partially funded and supported by Waiariki Institute of Technology.