Scholarship of Teaching and Learning: Developing a Student-Friendly Mobile Device Programming Editor to Increase Female Student Retention

Investigator: Hacer Varol, Ph.D. Computer Science Department. (936) 294-1075, hxv002@shsu.edu
Budget: $2,000
Course: COSC 1436 Programming Fundemantals – I. Sections 01, 02, 03, 04, and 05. Approximate Enrollment: 100

Executive Summary: The total number of enrolled majored students and awarded degrees in computer science is increasing in the last few years. However, to say the least, only having one female student compared to seven male students in computing area degree plans is a concern. In order to attract more women students and increase the retention in Computer Science, different strategies have been used. The effort put by the faculty varies among the level and style being used while some of those studies target introductory-level computer programming courses. The main rationale behind this is that some female students change their majors after taking the first course from computer science. Therefore, in order to promote the computer science field for women, we can say that the introductory level programming courses need to be more appealing to these students rather than being so difficult that they lose interest in this field of study. Therefore, in this project, a mobile device pictorial programming editor will be designed and implemented for COSC 1436 Programming Fundamentals - I course to replace the traditional desktop computer-based text editor. With the introduction of visualization and an interactive application for coding purposes on a device that one carries most of the time will increase the active engagement and help to reduce the loss of female students in the program.

Project Narrative: The number of student enrollment in Computer Science (CS) programs has been steadily increased [1]. However, there has been great concern about the low percentage of female students [2, 3]. Although the female student population in the CS area needs to be increased to accommodate the needs of the future STEM workforce, it has been reported that a number of CS female students tend to change their major after taking their first CS course [4], which is usually an introductory-level programming course. One of the major components in the introductory programming course is the use of code editors, which has a significant effect on female students' intention to keep pursuing CS as their major [5]. Although instructors have used visual editors that use block-like interfaces, such as MIT's Scratch[1], Greenfoot[2], and Alice[3], they are not only dedicated application for desktop platforms, but they also have other issues, such as the challenging transition from the concept of blocks to text-based programming language [6], complex visualization [7], difficult error tracking [5], and complicated source code viewers [8, 9]. Thus, while using the visualized user interface, students should be able to create, continue, and save their work in a fashion that will allow them to immediately "compile and run" (execute) their program in a way that replicates the conventional programming process. Since, currently, mobile java editor applications are mostly textual based, creating a mobile pictorial programming editor, in which the students being able to write and execute their programs visually on their mobile devices at any time and place will result in increased engagement and ready-to-go active learning.

**Objectives:** The development of a mobile device code editor is proposed to reduce the struggles and frustration experienced by female CS students in their first programming course. It is a dire need to develop a mobile editor that (1) facilitates the understanding of programming concepts, (2) automatically generates programming statements and the related syntax, (3) minimizes the time spent on fixing the errors, and (4) shows the results immediately.

**Methodology:** The procedures of this work include design/development and implementation/evaluation processes. _Design/Development._ The project team will utilize Java development modules for creating a mobile application for cross platform use (iOS and Android) and Pico Compiler API[4] for compiling and executing the written code. The specific four features are as follows: **(1) Minimizing Syntax Errors:** We propose to develop the editor tool to reduce the frustration level by designing it to decrease the number of syntax mistakes while writing a program code. To do this, we will use buttons that prompt the student to enter the coding statement without providing a syntax for it. Then, the editor will generate the complete code including the appropriate and correct syntax rules; thus, removing the chance for syntax errors. This process will also teach the students the syntax by reflecting on what is needed to make the complete statement. Once the student enters the information required by each button, the generated code will be sent to a text window. The students will be shown the source code before they drag it to the text area; **(2) Immediate Responses:** The editor tool will provide immediate responses. The student will be able to click on a button and enter the desired operation. They will immediately see what the code looks like without having

---

[1] https://scratch.mit.edu/

[2] https://www.greenfoot.org

[3] https://www.alice.org

to consult texts or references. After the code is created for the students, they will be able to drag the code directly to the text area as they are viewing the code immediately; **(3) Easy Transition:** The ease of transition is important because as the students' progress into the next level, they will need to be able to utilize the advanced programming environments (i.e., IDE[5]). Therefore, the text area will be similar to a modern programming environment, such as automatic indenting and syntax highlighting; and **(4) Instructor Dashboard:** Through a reporting system, the proposed tool will be designed to support the instructor to analyze the performance of students' coding results.

Implementation/Evaluation. After developing the proposed tool, other instructors who teach introductory level programming courses (Ph.D. students) will also utilize this tool in the 2021-2022 academic year. The student performance on coding, learning outcomes (i.e., final grade), drop-out rates, and satisfaction levels will be analyzed by comparing the courses that use the proposed tool with the control courses that use the conventional coding tools.

Expected Outcomes: The primary outcome of this project is a female student-friendly mobile device programming tool that will be utilized in COSC 1436 starting from the 2021-2022 Academic Year. Additional outcomes can be seen in Table 1. In addition, this project will be used as the foundation for seeking future external research grants.

The Significance of Expected Results: Providing a female student-friendly mobile programming editor is required given the sharp decrease in the female student enrollment in CS courses after taking their first programming language courses. It is expected that this project contributes to the CS field by supporting female students to keep working on their computer programming learning on-the-go with the tool's supportive features described above. Finally, it is expected that the increased female student population in CS will address the shortage of workforce in diverse professional STEM areas.

Budget and Budget Justification: $1,000 stipend is requested for Dr. Varol who will oversee the implementation, testing, and overall evaluation of the application. Another $1,000 stipend is requested for a junior level Software Engineering student who will implement the mobile application with the supervision of the investigator.

Table 1: Timeline and Expected Outcomes of the Project

Phase	Tasks	Spring 2021										Spring 2021	Fall 2022	Expected Outcomes
		Jan 15-31	Feb 01-15	Feb 16-28	Mar 01-15	Mar 16-31	Apr 01-15	Apr 16-30	May 01-15	Summer 1	Summer 2			
1	Interface design	▓	▓											Product design
	Function design		▓	▓										A prototype of the product
2	Interface development				▓	▓								Product development
	Function development						▓	▓						The developed product
3	Product implementation							▓	▓					Product validation
	Performance analysis								▓	▓	▓			Performance report
4	Project report writing										▓			Final report
5	Results Dissemination											▓	▓	Publication/Presentation

Works Cited

[1]   S. Zweben and B. Bizot, "Continued Booming Undergraduate CS Enrollment", 2015 CRA Taulbee Survey, *Computing Research News*. 2016. Available: http://cra.org/wp-content/uploads/2016/05/2015-Taulbee-Survey.pdf

[2]   M. Crawford, "Engineering Still Needs More Women" *The American Society of Mechanical Engineers (ASME)*, 2012. Available: https://www.asme.org/career-education/articles/undergraduate-students/engineering-still-needs-more-women

[3]   D. Beede, T. Julian, D. Langdon, G. McKittrick, B. Khan and M. Doms, M, "Women in STEM: A gender gap to innovation," *Washington, DC: Economics and Statistics Administration*, U.S. Department of Commerce, 2011

[4]   L. Jackson, A. Lawson, M. Diack, E. Khosravi, and R. Vincent-Finley, R, "An Analysis of Pair Programming as a Pedagogical Tool for Student Success in Computer Science at a Community College," *EdMedia: World Conference on Educational Media and Technology,* 2016, pp. 699-703.

[5]   H. Varol, and C. Varol, "Improving Female Student Retention in Computer Science during the First Programming Course*", International Journal of Information and Education Technology*, Volume 4, Issue 5, pp. 394-398, October 2014, DOI:10.7763/IJIET.2014.V4.437

[6]   W. M. DuBow, I. Valkalis, M. A. Perez-Quinones, and J. T. Black, "Growing female undergraduate enrollments in computer science: some successful approaches", *In Proceedings of the 44th ACM technical symposium on Computer Science Education,* March 6-9, 2013, Denver, CO, pp. 377-378.

[7]   W. Dann and S. Cooper, "Education Alice 3: Concrete to Abstract," *Communications of the ACM 52.8,* 2009.

[8]   P. Henriksen and M. Kolling, "Greenfoot: Combining Object Visualization with Interaction," *Conference on Object Oriented Programming Systems Languages and Applications*, 2004.

[9]   S. Kouznetsova, "BlueJ and Blackjack to Teach Object-Oriented Design Concepts in CS1," *Journal of Computing Sciences in Colleges* 22.4, 2007.

4 Pico Compiler (https://www.marcinolawski.pl/pico/)

5 Integrated Development Environment (https://en.wikipedia.org/wiki/Integrated_development_environment)