

STUDENTS' PERCEPTION TOWARDS PROGRAM VISUALIZATION ON SMARTPHONE - CASE OF SUNLAB INITIAL INVESTIGATION

Elhard James Kumaliya¹, Ymran Fatih² and Yi Sun²

¹*Dodoma University, Dodoma, Tanzania*

²*Kobe Institute of Computing, Kobe, Japan*

ABSTRACT

Effectiveness of program visualization tools in teaching and learning programming is well documented. However most of program visualization tools are desktop applications, this study aims at investigating students' perception of program visualization on Android smartphones. Study result shows that program visualization using smartphones can arouse students' enthusiasm in learning programming especially when they can write program to interact with their environment. Students were interested in writing small programs to interact with smartphone sensors and perform trivial functions like sounding an alarm whenever a phone enters a school football pitch.

KEYWORDS

m-Learning, e-Learning, Program Visualization, Computational Skills, Teaching Programming

1. INTRODUCTION

Computational thinking is nowadays a vital skill in our daily life. In addition to reading, writing, and arithmetic skills computational thinking is seen as necessary and important skills in this digital age. Computer programming is an effective and most applicable way of teaching computational thinking. However, computer programming is difficult to master. Therefore, different teaching methods and tools have been developed to help students master computer programming concepts. According to Serafini's vision learning programming on an adequate level of abstraction is a very effective didactic approach to Computational Thinking (Serafini, 2011). Program Visualization (PV) tools are promising programming teaching tools in early stages of the learning path of a programmer. PV tools can engage students in viewing animation, creating animations, responding to questions and changing inputs.

Urquiza-Fuentes and Velázquez-Iturbide surveyed 24 experiments of evaluating program visualization and algorithm animation systems. Their results show that learning can be improved with PVs in different ways depending on engagement level. Improvements in knowledge acquisition were detected in 75% of the experiments at any engagement level (Urquiza-Fuentes, et. al., 2009). Moreover, improvements in attitude towards either material or subject matters were detected. Finally, it was found that programming skills of students were improved. Despite of the importance of PV in teaching and learning computational skills there are few studies on the effectiveness of PV using smartphones. In this paper, we present finding on the students' perception of program visualization on smartphones using SunLab and make several contributions to future works on PV on smartphones. This paper is the extended work of the paper "dynamic program visualization on Android smartphones for novice Java programmers" (Kumaliya, E. et. Al. 2018).

2. RELATED WORK

There are different approaches that have been adopted to utilize the potential of mobile phone to students studying programming. These approaches include delivering education content, introduction to programming where students learn directly by developing smartphone applications and using the smartphone to write

programs that will be run on smartphones (John M.S. & Ran, M. 2015). Moreover, there are mobile platforms for learning programming on smartphones like mobProg a mobile-based application that provides students with a smartphone-based platform for learning Java programming (Hashim, A. 2007) and Microsoft TouhDevelop a programming environment intended to enable anyone to use a phone to program the phone using scripts for their windows-based smartphones (Athreya, B. Et al. 2012). Despite wide availability of smartphones, there are few studies that attempts to include PV in mobile devices for teaching programming.

In teaching programming there are several studies on smartphones to visualize program or algorithms in web-based format. Example, a collection of algorithm visualizations of various classical and non-classical algorithms presented in a unified and interactive web-based visualization (Halim et. al. ,2012). mJeliot provides support for students to interact with Jeliot via their mobile phones (Pears et.al., 2011). These tools do not support students to create their own programs and visualize them. Improvement is made in SunLab to enable students to write their own programs, compile, and execute their programs using smartphone. In addition to that students can interact with their environment using the sensors embedded on their mobile phones.

3. SUNLAB, A PROGRAM VISUALIZATION TOOL ON ANDROID SMARTPHONE

SunLab is a dynamic program visualization tool developed to help novice programmers to learn programming concepts using Android smartphones. The initial concept was a dynamic program visualization tool for novice Java programmers (Kumalija, E. et. Al. 2018). This tool was changed from Java programming to pseudo programs so that to focus on learning the programming concepts, not the language syntax.

SunLab supports dynamic visualization for sequential instruction execution, variable declaration, and assignment, Expression Evaluation, data input, message output, selection, loop and function calls. SunLab key features are language independent, visualization control, visual tracing of statement execution and ability to interact with the environment through smartphone embedded sensors.

3.1 SunLab Program Visualization

The diagrams below show SunLab editor panel and program visualization panel.

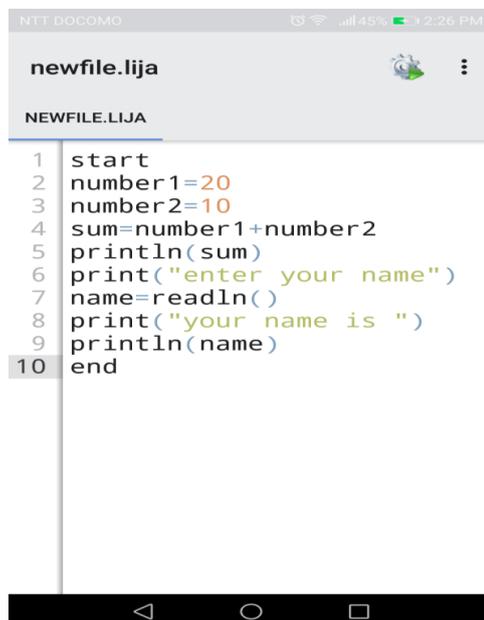


Figure 1. SunLab editor panel

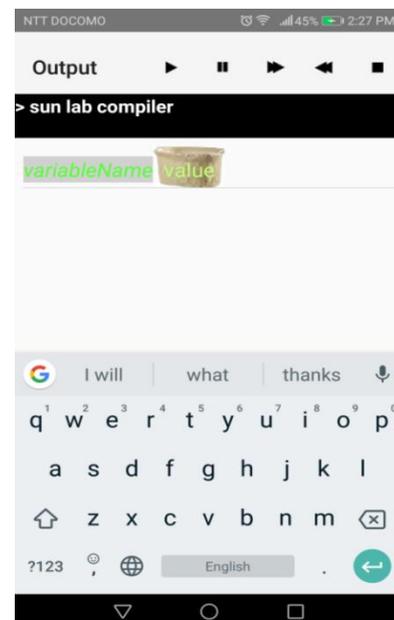


Figure 2. SunLab editor panel

In every statement execution a graphical presentation of variable changes and output is generated. Students can control the animation speed by using control buttons on top of the animation window.

Sequence of instruction execution is animated by highlighting with pale dark blue color the instruction number of the current instruction (see instruction number 10 in figure 1 above). The console of traditional compilers is retained where user can enter input to the program and output of the program as shown in figure 3 below. Understanding variables and role of variables is the most important part in understanding programming. SunLab provides variables visualization.

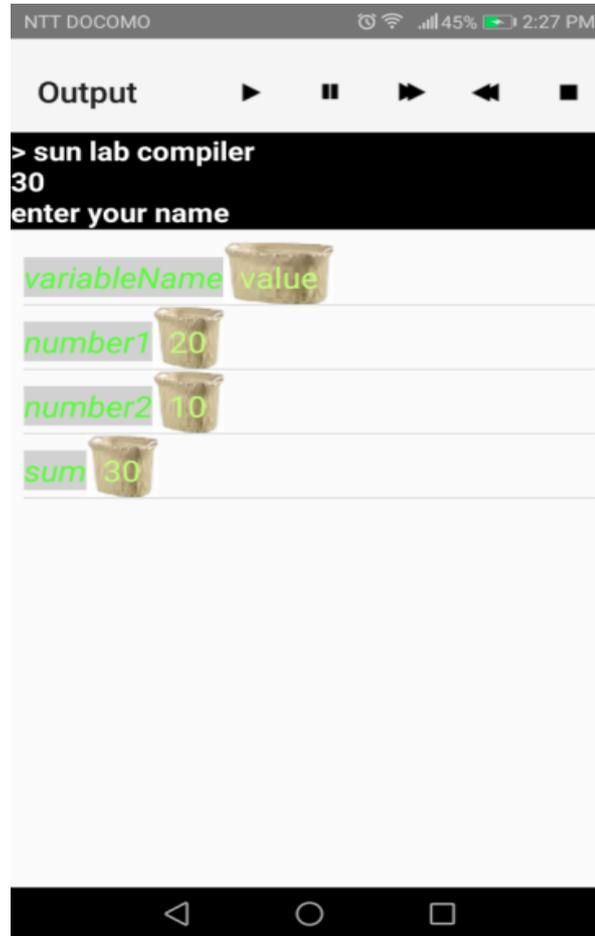


Figure 3. Variables and Role of Variables Visualization

In computer programming, a **variable** is a storage location (identified by a memory address) paired with an associated symbolic name (an *identifier*), which contains a data item that may take on more than one value during the runtime of a program. In SunLab a variable is visualized as a basket containing a value as shown in figure 3 above. The value on changes as the variable value changes during runtime of the program. More variables appear as they are referenced and disappear as they are no longer referenced for students to easily understand the role of variables.

Every programming language has built in functions to help programmers with common tasks like data input, data output, and mathematical functions. SunLab has built-in functions to help users create applications with less effort. These functions support user input, printing output to console, taking a picture using a camera, playing music and getting values from sensors embedded in a smartphone. Moreover, SunLab has functions for putting a flash on and for performing basic mathematical functions.

4. RESEARCH DESIGN

Lower secondary school students at Kokirie secondary school in Tanzania were recruited for initial investigation. The criteria for recruitment were: - the student should have taken a class on introduction to computer programming, student should have permission to use a smartphone and bring a smartphone to school from home. Thirty-one students in lower secondary school volunteered for the study, among these 31 students 18 were girls and 13 were boys. The age distribution of participants was between 14 years old to 17 years old. Students used different versions of Android OS and smartphones.

Recruited students were introduced to SunLab application features for 2 hours every day in one week. This introduction took place at school, then after they were given one week to use the SunLab out of school time. After one week of using the application, students' feedback on usage of SunLab application was collected through interview with participants. The focus of data collected was frequency of use, ease of use of the editor, visibility of animations, preference of built in functions and if it helps to understand the programming concepts. Furthermore, students were asked to showcase applications they have written.

5. RESULTS AND DISCUSSION

5.1 Continuance Intention to Use

When a user uses the application once and does not use it again, it is the sign that the user did not like the application. Students were asked how many days in the week they used the application at least once in a day. 19% used the application in less than days, 32% used the application in less than 4 days, while 49% used the application in 5 to 7 days.

The high number of students who used and reused the application signifies that students were interested in using the application.

5.2 Usability of the Text Editor

The text editor for programming is an important component of the SunLab system. It is important to know how users perceive how easy or difficult is to use the text editor.



Figure 4. Editor Usability

Generally, users were satisfied with the usage of the editor, out of 31 participants 18 participants found that it is neither difficult nor easy to use the editor. However, 16 participants expressed their concerns on the difficult on using special keys on the virtual keyboard.

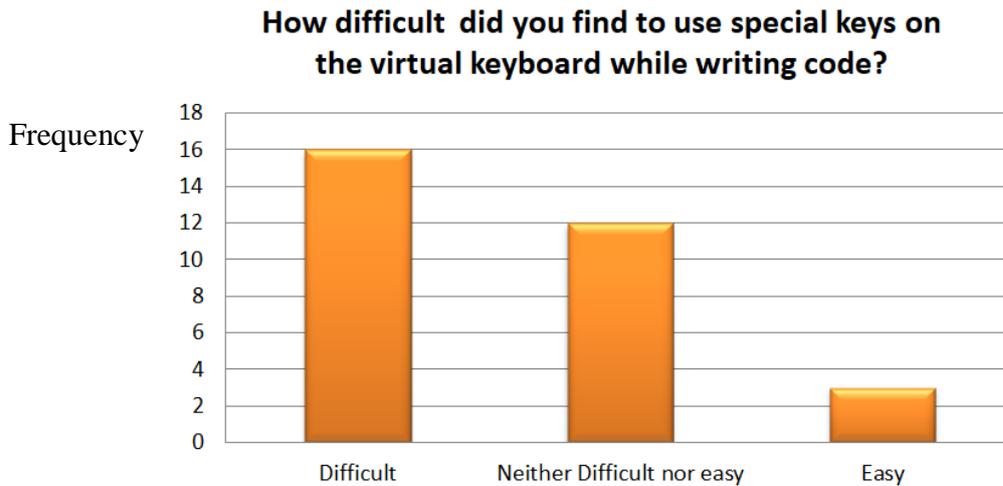


Figure 5. Response on Usability of Special Keys

Using special key was found difficult because it is not common to use special keys in the android users, it is very rare for someone to use special key like { or >> in daily usage of smartphone. Sometimes students were required to go three levels down to get the key. The key suggestion bar was helpful but it was new to students.

5.3 Animation Visibility

Students were asked on the visibility of animations. Visibility of animations was intended to measure how students perceived the easiness to see and understand the graphical execution of the program.

How did you find the visibility of animations?

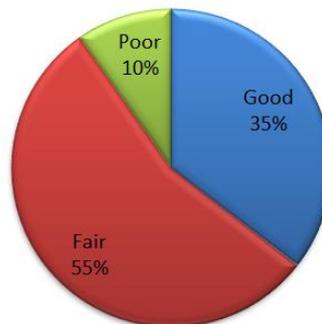


Figure 6. Animation Visibility

In this question, 35% responded positively that the visibility of animations is good and 55% response towards visibility of animations was neutral. These are generally promising results considering the screen size of the smartphones. However this area needs improvement.

5.4 Function Preferences

In any programming language exists pre-defined functions for common programming tasks like input, output and mathematical functions that users can use without the need to re-invent the wheel. SunLab is no exception. It contains pre-defined functions (inline functions) that users can use. These pre-defined are to help users with common programming tasks. Moreover, in Sunlab these functions help users to program and interact with different smartphone embedded devices like camera, GPS etc.

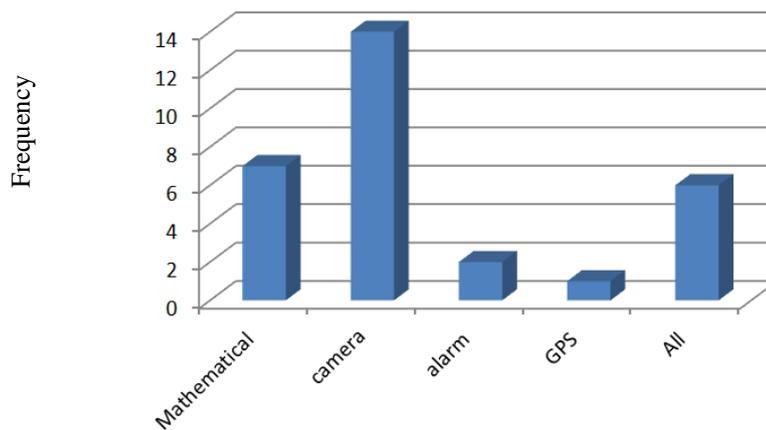


Figure 7. Built-in Functions Preferences

The histogram above shows the preferences of students on SunLab inline function. Most students expressed interest in camera functions, mathematical functions being the second in preference. The camera preference is attributed to daily usage of camera by these students so they were curious how they can program and control the camera from their code. The GPS functions were less preferred, the reason being complicated to make any useful information from the data received from GPS functions and the mathematics involved in the processing of data to get desired result which was a heavy cognitive load for this level of students.

5.5 Help in Understanding Programming Concepts

The core objective of developing SunLab application is to help students understand programming concepts easily and acquire computational thinking skills. Students were asked if using SunLab helped them to understand programming concepts. Students were asked if this application helped them to understand expression evaluation, variables and arrays, if-else, while loop and functions. Program visualization was provided for these programming constructs.

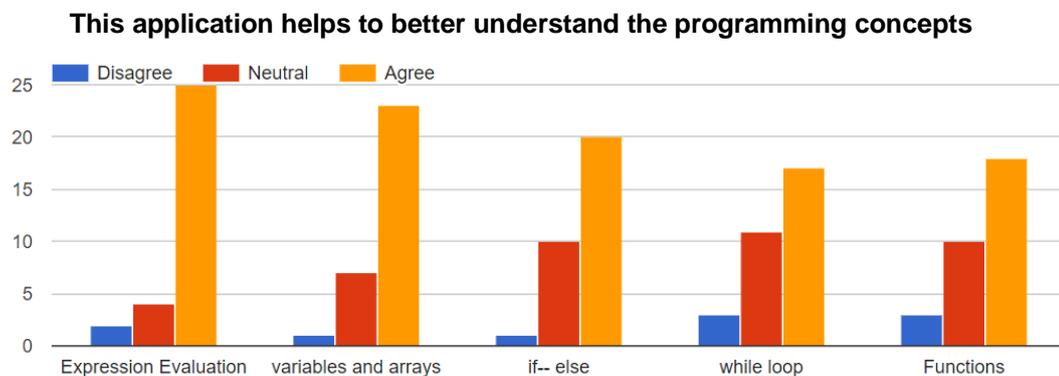


Figure 8. Help in Understanding Programming Concepts

The above histogram shows the response of students. Most of the students agreed that this application helped them to understand programming concepts better than before. However very few were neutral and some students said it did not help them.

5.6 Applications Developed Using SunLab

Every student was told to write any application he/she can write using SunLab and showcase that application at the end of one week. Students wrote very trivial applications like application to add two numbers, application

to show usage of looping function. Using built-in functions students wrote programs to take picture using camera and sounding alarm.

One interesting application was developed by two students in collaboration. They developed an application that sounds an alarm whenever they are out of the school's football pitch. During presentation they said their application can be used to alert them when they go to play so that they will not go very far from their village, the issue that their parents insists every day.

6. CONCLUSION

In this initial investigation, it was observed that mobile devices are potential teaching and learning tool in lower secondary schools. Mobile phones are widely available at home. Hence they can be used as tools for students for after school study activities.

Students participated in this study learned mathematical and computational ideas (such as variables, conditionals, loops and functions), they are also learnt strategies for solving problems, designing projects, and communicating ideas. Using SunLab application also stimulated their creativity thinking, by thinking what they can do more with smartphones and computers in general. These skills are useful not just for computer scientists but for everyone, regardless of age, background, interests, or occupation.

Programming using mobile phones can be a viable solution to schools with financial and environment constraints for setting a computer laboratory. There is no upfront cost for schools to set a computer lab and the school does not need to be connected to electricity supply or have electricity generator. Collaboration between school and parents can enable students to learn computational and programming skills during after school hours using parents' smartphones.

Currently, SunLab is designed for writing basic programs. Later it can be extended to support programming Android components like views so that to enable students to write Graphical user Interface android applications using their mobile phones. Moreover, the inline function to interact with devices like camera, accelerometer, GPS, and flashlight can be improved.

This research proved the concept that smartphone can be used in classrooms to teach programming. However, there are challenges like editor usability. It is okay to write small programs using mobile editor but the task becomes tedious when the program has many lines of code. It is challenging to visualize variables when users declare long variable names due to small screen size of smartphones. Long-term evaluation of the Sunlab application is recommended to assess its' effectiveness in improving the performance of students in semester by comparing it with students taught using a traditional computer laboratory.

REFERENCES

- Athreya, B. et al. 2012. End-user programmers on the loose: A study of programming on the phone for the phone. In *Visual Languages and Human-Centric Computing (VL/HCC), 2012 IEEE Symposium*, pp. 75-82.
- Halim, S. et al. 2012. Learning Algorithms with Unified and Interactive Web-Based Visualization. *Olympiads in Informatics*, 6.
- Hashim, A. 2007. *Mobile Technology for Learning Java Programming-Design and Implementation of a Programming Tool for VISCOS Mobile*. Unpublished Master's thesis, University of Joensuu, Finland.
- John, M. S. and Rani, M. S. 2015. Teaching Java Programming on Smartphone-pedagogy and Innovation; Proposal of its Ontology Oriented Implementation. *Procedia-Social and Behavioral Sciences*, 176, 787-794.
- Kumalija E. et al. 2018. Dynamic Program Visualization on Android Smartphones for Novice Java Programmers. In *IADIS International Conference Mobile Learning*, pp. 173-177.
- Pears, Arnold. and Moritz Rogalli. 2011. mJeliot: a tool for enhanced interactivity in programming instruction. In *Proceedings of the 11th Koli Calling International Conference on Computing Education Research*, pp. 16-22. ACM.
- Serafini, G., 2011, October. Teaching programming at primary schools: visions, experiences, and long-term research prospects. In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives* (pp. 143-154). Springer, Berlin, Heidelberg.
- Urquiza-Fuentes, J. and Velázquez-Iturbide, J. 2009. A survey of successful evaluations of program visualization and algorithm animation systems. *ACM Transactions on Computing Education (TOCE)*, 9(2), 9.