

A Web based Approach for Teaching and Learning Programming Concepts at Middle School Level

Sania Bhatti

Department of Software Engineering, Mehran University of Engineering & Technology, Jamshoro
Email: sania.bhatti@faculty.muett.edu.pk

Amirita Dewani

Department of Software Engineering, Mehran University of Engineering & Technology, Jamshoro
Email: amirita@faculty.muett.edu.pk

Sehrish Maqbool

Govt. polytechnic Institute girls, Latifabad, Hyderabad
Email: memon_0686@hotmail.com

Mohsin Ali Memon

Department of Software Engineering, Mehran University of Engineering & Technology, Jamshoro
Email: memohsin@gmail.com

Received: 28 January 2019; Accepted: 01 March 2019; Published: 08 April 2019

Abstract—One of the major concerns in teaching and learning programming concepts is the complexity of syntax and precision of semantics of programming languages. Traditional teaching methods are static and passive i.e. they do not engage students in an interactive manner thereby making it difficult for students to grasp the contents and instructors to convey the instruction. This obstacle even becomes challenging when programming courses are to be taught to beginners. To cope up with this challenge, this work has proposed and prototyped a system that is aimed to focus on students at their middle level of education. Multimedia technology i.e. videos have been used to plunge the students in an interactive environment where learning JavaScript programming becomes fun instead of a mind-burden. Visualization concepts have been incorporated to provide visual learning for variables, loops, control structures, functions etc. This application is dynamic in nature that is user can not only understand the programming concepts but can also run the codes using code panel. The designed system has been tested to ensure the functionality, performance and feedback from the targeted users as discussed in results section.

Index Terms—Visualization, programming tool, scripting languages, JavaScript, learning, middle school.

I. INTRODUCTION

Programming is the procedure of developing an executable machine level program that accomplishes a

specific task and can be executed without any error. It is the process of formally writing codes based on the human inputs and corresponding machine outputs. Programming makes the creation of computer applications and software's possible. It is something that can help a common person and students to understand and to interact with the technology efficiently. Literature has highlighted that following problems [1] are encountered in learning computer programming skills:

- Starting with theoretical concepts first in coding usually frustrates the beginner.
- Conventional code learning courses start with the console-based applications and this is an uninterested approach for the students.
- The programming complication and ambiguity also lead towards the deficiency of learning programming among the students.

After conducting a survey based on views of 105 Engineering students about difficulties in learning programming, authors [2] suggested that a visualization tool would be useful for assisting students in understanding the employment of the program structure. Graphical programming replaces the console-based output with images, graphics and animation [3]. It is difficult for middle school students to learn the code lines which give colorless text output; they always want a language that can help them in not only learning the programming concepts but also in designing the physical objects which they design using graphical tools like

Adobe Photoshop. A simple application for learning basic programming concepts which gives output as graphical objects is highly demanding. A web-based visualization application is the best direction which would assist the students for entering in the world of programming because after doing this the students will have the clear idea that whether he/she wants to go towards designing, game creation or console application creation. This type of application will help at larger scale in creating the interest in middle school children towards learning programming because it represents the programming language in an effective and interesting way so that the student must not build a fearful imagination about programming languages.

This work mainly focuses on the idea of offering a way to learn the programming through creative graphics and this makes the program creation accessible for the beginners, teachers and designers. New users will start writing programs by learning through graphics rather than console-based outputs. If the students' experience with computer programming during middle school increases, then they may more likely consider enrolment in a high school programming course. The work proposes a web-based learning approach that can help the middle school students to learn the programming concepts using a language that gives a complete graphical output of programming instructions. The designed web application, alias "Codeable", provides an easy to use environment for students to enhance their code learning and teaching them the concepts like variables, functions, control structures in JavaScript language [4]. p5.js library has been used to work with LOC (lines of code). This library provides the facility of creating drawings and animation [5] using programming logics.

The paper is organized as follows. Section II highlights some of the recent works. The Codeable web architecture is depicted in section III. The detailed implementation is shown in section IV. Testing results are analyzed in section V. Finally, section VI states concluding statements with future scope.

II. RELATED WORK

Visualization tools have revealed excessive potential [6] for developing an environment to assist novices for understanding programming concepts. As the students cannot realize what is happening in a computer when a program is executed, a teaching support to assist them in visualizing the programming concepts animatedly is the need of the day to enrich students' intangible understanding regarding programming environment. Scratch is one of the tools that are used for teaching programming through animations [7]. In Scratch environment, a program is consisted of a set of animated images whose location and visibility are controlled with scripts, control structures and drag-and-drop programming tools. A script is simply a combination of blocks or code lines that instructs a sprite about its functionality. The blocks or scripts are dragged and dropped into the scripting area. From there, they are

joined together in particular fashion to give sprite a specific functionality. For students, under the age of 12, it is difficult to use an environment with so many advanced tools. For using scratch environment, a tough training of teachers is also required. This environment does not train the students about writing the lines of code with proper syntax and semantics in any conventional programming language. To scrutinize which programming constructs are used by students in Scratch, authors in study [8] did investigation using quantitative analysis and compared it with Alice environment [9]. The study involved more than 300 students, ages ranging between 11-14, and analyzed three types of project including games, videos and storytelling.

Another study has proposed a code visualization approach in which kids are taught programming through some activities [10]. A gaming theme is used for student assessment. They are given some activities to solve syntactic and semantic problems in code, these activities can be "spot the error", "predict output", "put in the correct order". The gaming activities or course is evaluated by teachers and designers. It does not only teach the programming concepts but also knowledge of writing lines of code in easiest way. One of its limitations is that it does not teach students a complete program flow. It does not teach students that how data moves from one method to another method or where to write the statements i.e., whether a control structure should be written before variable declaration or after variable declaration etc. This application covers multimedia technology to allow students to visualize the lesson contents. It does not provide any text editor to write the code in.

Authors proposed an IDE i.e. Geenfoot in study [11]. This IDE was developed to learn programming and typically focuses on students with age 14 and upwards. Geenfoot is extensively used for teaching courses at college and university, but it currently covers java programming language only. Authors proposed a tool i.e. "Iconic Programmer" in study [12], which allows students to use mouse click menus, options and symbols to write programs. This tool is used extensively in college and university courses but focuses on java programming language.

Challenges in learning and teaching programming have prompted researchers and scholars to contribute in this field. A lot of work has been carried out as discussed previously in this section, each having its own context and are mostly based for learning java programming language. In this research work we have proposed an idea using multimedia technology and visualization to learn JavaScript scripting language.

III. SYSTEM ARCHITECTURE

In this section, the architecture of the Codeable is discussed. In Codeable, HTML5, CSS3 and BOOTSTRAP has been used for the user interface presentation and JavaScript, P5.JS and NodeJS for the business logic implementation at the back end of the

application. Presentation logic is created by HTML5 that is carried out by JavaScript, which supports numerous APIs to load data on the web pages dynamically without the need of refreshing the pages. Bootstrap is used to make this web application responsive. Pawtoon has been used to create video tutorials of the specific programming concepts.

Fig.1 depicts the System diagram and illustrates each functionality of the system starting from the user identification, environment (Nodejs – express and body-

parser libraries) used for sending requests to the server in Codeable; how all the entities are connected to the server and how a client receives the response. The communication among components of the system is also shown in Codeable system diagram. The server is responsible to process requests and acts as a main component; client can interact with application when server response is correct and provide all the presentation and scripting files needed by the user to perform different tasks.

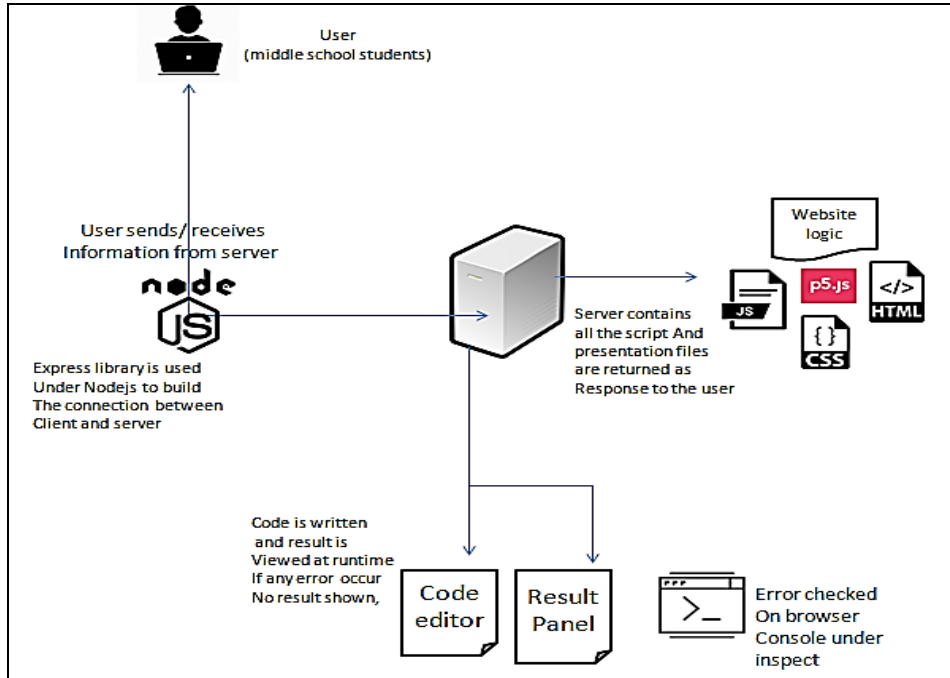


Fig.1. Codeable application architecture

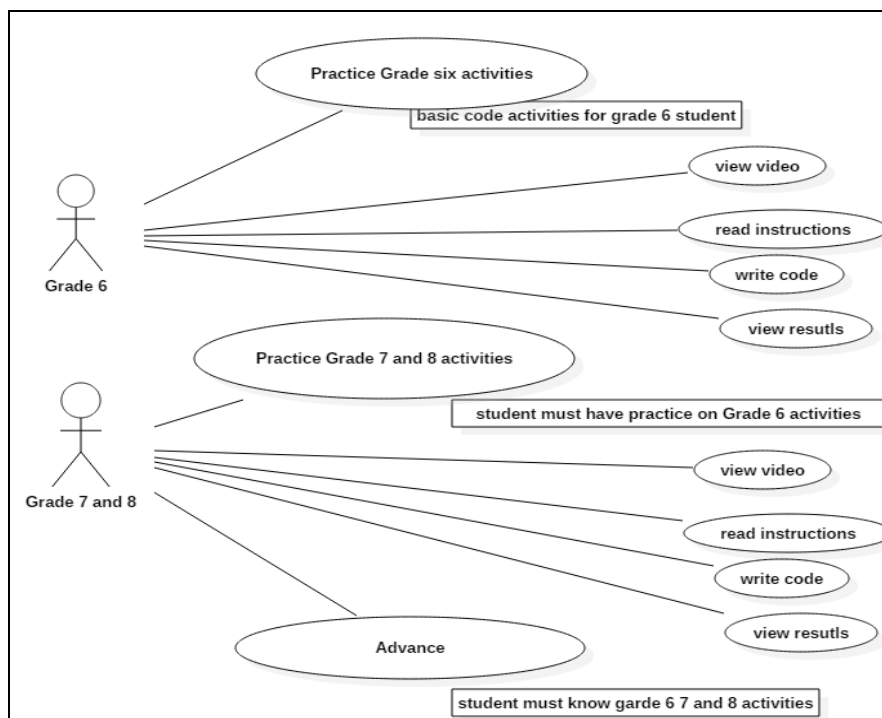


Fig.2. Use case diagram elaborating Codeable actors and activities

As this work is serving the middle school students; so grade six students will practice on very basics to know what the programming is and to perform each activity student is directed to view the video and read the instructions at the very first stage. Afterwards student can be able to read and write the code and view the results. As the student completes this step he/she can jump to the practice of grade 7 and slowly jumps towards the class 8th activities. The use case diagram shown in Fig. 2 elaborates the activities which can be performed by the students.

IV. SYSTEM IMPLEMENTATION

All modules are connected in network-based client server fashion. The application contains two main interfaces i-e the home page that navigates the student towards different levels of code learning and the student’s activity interface. The student’s activity interface contains the code and result panel where student can write and execute the code respectively after learning programming logics from given video tutorials. This application helps the student to build the programming concepts after watching video tutorials at lower level and higher level students can also write the whole program lines using code panel and can see the result in form of text, graphics, colors and animation on results panel. The student can also see the errors in code syntax using the browser console.

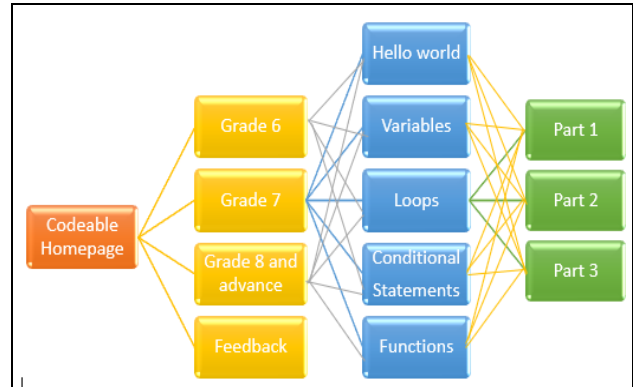


Fig.3. Codeable Layout

The home page of the web application directs the students towards different level of code learning as shown in Fig. 3. It contains the navigation bar with containers that are marked as grade 6, grade 7, grade 8 and advance. The containers facilitate the user for selection of programming concepts based on their grades. After choosing the category for code learning, the user will choose that which programming concept he/she wants to learn from category. This interface is comprised of containers that will navigate the user towards an activity interface for programming concept i.e., functions, variables, control structures, arrays etc. The activity user interface is shown in Fig. 4; consisting of four panels which are: code panel, instruction panel, tutorial panel and result panel.

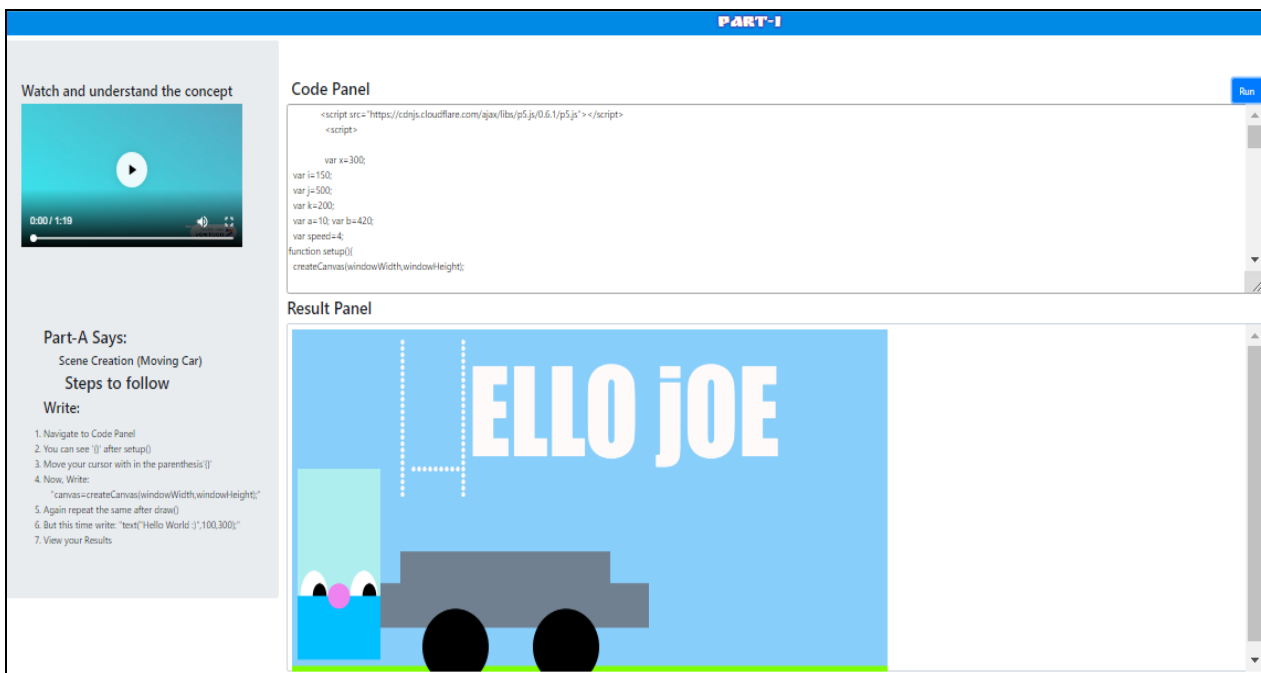


Fig.4. Results Panel

A. Code Panel

This panel is like a code editor. The user can write code in this panel. The code editor can run simple JavaScript as well as p5.js code. The errors that can occur in lines of code can be checked on web browser’s console using inspect function. This panel is depicted in Fig. 5.

B. Instruction Panel

This panel will instruct the user that how to write a code on code panel with proper syntax and semantics.

C. Tutorial Panel

This panel provides a video tutorial to the users. This video tutorial teaches the student about a specific programming concept i.e., loop, if else, etc. and the language that is used to implement it. Before writing code, it is necessary for the students to watch the video.

D. Result Panel

This panel will show the result to the user. After execution of code the result is displayed in this panel. The result panel is shown in Fig. 6.

```
Code Panel
<script src="https://cdnjs.cloudflare.com/ajax/libs/p5.js/0.6.1/p5.js"></script>
<script>
  var x=300;
  var i=150;
  var j=500;
  var k=200;
  var a=10; var b=420;
  var speed=4;
  function setup(){
    createCanvas(windowWidth,windowHeight);
  }
  function draw(){
    background(135,206,250);
    fill(127,255,0);
    rect(0,530>windowWidth,300);
    noStroke();
    fill(255);
```

Fig.5. Code editor for writing code

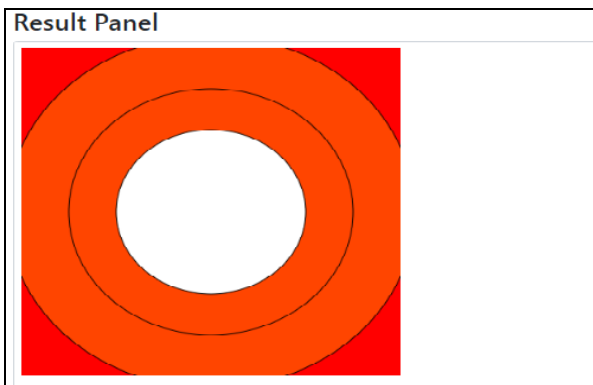


Fig.6. Results Panel

V. RESULTS & DISCUSSION

Three types of testing have been performed to measure the quality of developed system. These are load testing, platform testing and usability testing.

Minigun is a simple but powerful load testing tool and generates reports as easy-to-read JSON files. Detailed performance metrics are observable by generating graphical reports. Test summary is presented in Fig. 7. Test was successfully completed without any network or OS errors.

Test run on 03 Oct 2018 18:05:37			
Summary		Codes	
Test duration	120 sec	404	2400
Scenarios created	1200		
Requests completed	1200		

Fig.7. Test summary generated via Minigun

The html output of a testing report generated after configuring and running Minigun on a page of Codeable on localhost is presented here. All the pages are tested in the similar fashion. Fig. 8 depicts the overall latency distribution along x-axis and time in mill seconds along y-axis. It shows that minimum and median values of latency are less than 5ms for 200 requests.

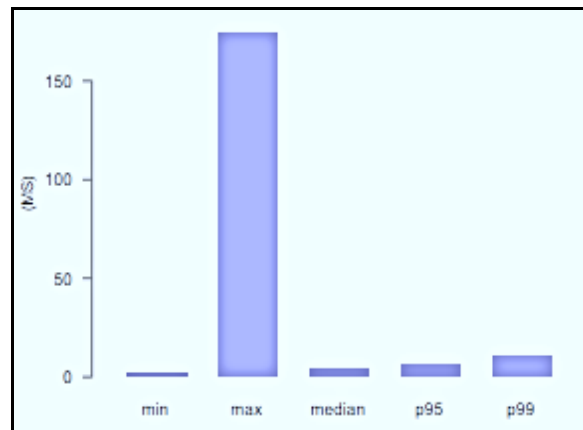


Fig.8. Overall latency distribution generated via Minigun

Fig. 9 portrays the idea to look at the intervals between successive occurrences of the latency values. Latency at intervals shows how much latency occurs at different time periods.

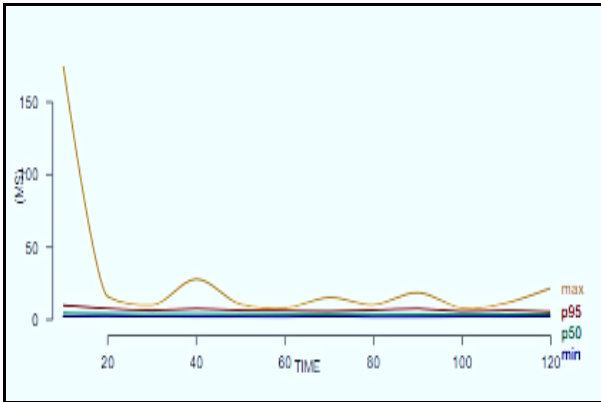


Fig.9. Latency at intervals generated via Minigun

Fig. 10 illustrates the mean requests per second (RPS) which is a scalability measure characterizing the output handled by a system. In this graph, mean RPS shows the sizing and scale of load testing by testing APIs. RPS count specifies how much RPS decreases when user load increases and is shown in figure 11. It shows that the RPS count is constant over the duration of run.

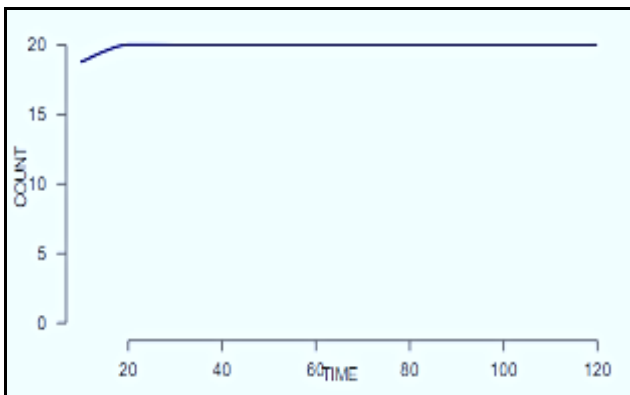


Fig.10. Mean RPS count generated via Minigun

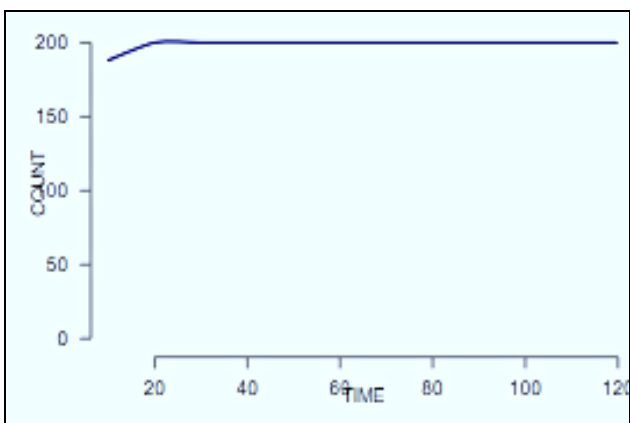


Fig.11. RPS count generated via Minigun

Table 1. Results of platform testing

Platforms	Loading	Design	Control modules & Interface activity	Overall Functionality
Opera	OK	OK	OK	OK
Google chrome	OK	OK	OK	OK
Mozilla Firefox	OK	OK	OK	OK
Microsoft	OK	OK	OK	OK

After running Codeable on different platforms it was observed that control modules, interface activity and its overall functionality is acceptable. All the results are shown in table 1.

As per guidance from literature the quality of a web-based application could be measured by adopting web quality frameworks. For the evaluation of Codeable, we have employed web evaluation framework (WEF) [13] partly according to the users of our web application. WEF is a hierarchical framework based on two levels. The first level is comprised of five quality metrics which are Aesthetics, Ease of Use, Multimedia, Rich Content and Reputation. Each of this quality metric is further partitioned into variety of quantifiable parameters. Codeable usefulness was measured through a likert-type scale. This scale usually comprises of a series of statements with four to seven options. For maintaining simplicity equal weightage was assigned to individual quality attribute. Because Codeable is designed for middle school children, therefore 3-point scale for quality measurement is included in the feedback interface of the application consisting of “Poor, Good, and Excellent” [14]. The feedback was collected from 50 students of Grade 6, among them 23 were boys and 27 were girls. The total quality value was calculated using equation 1 given by authors in [15]. The resulting measures after taking feedback from the sample are shown in Fig. 12. The statistical data reveals expressively positive results and students’ satisfaction regarding the quality measures. Number of students with the good rating of likert-scale is expressively higher than the poor in all the quality measures.

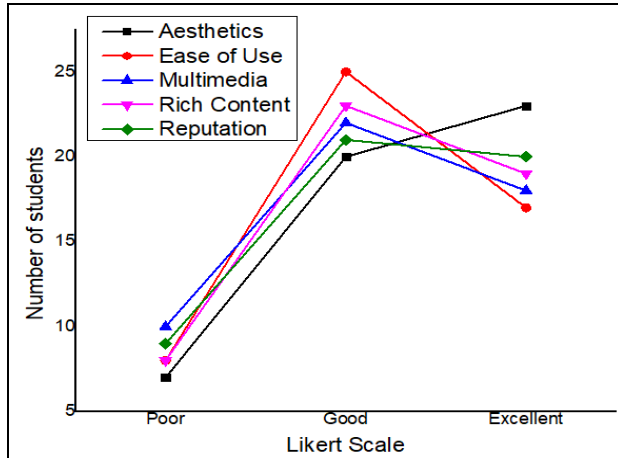


Fig.12. Codeable quality measure via WEF

VI. CONCLUSION

Teaching and learning have been traditionally challenging because of the abstract nature of programming concepts. This paper has presented a web-based learning environment that aids learning of JavaScript concepts. There are two main contributions of this work. First contribution is the development of a web based platform based on multimedia technology for learning Javascript. Second is the performance of three types of testing including load testing via minigun, platform testing and usability testing by deploying web evaluation framework.

Currently, basic contents and constructs have been considered like variables, loops, conditional and control structures etc. The pedagogical goal of this application was to enable students, beginners and novices to learn programming in an interactive way. Moreover, this application does not only provide the information of programming concepts using video tutorials, graphics and instructions but it also provides a code panel that works just like a code editor. Using that code editor student can write both the code given in web application and the self-created code. The system is fully functional and got satisfactory feedback from end users as mentioned in results section. In future, this application can be enhanced to incorporate more advanced programming concepts. This application has used some limited libraries of P5.js, in future some more libraries can be used to work with animations and sounds.

ACKNOWLEDGMENT

The authors are extremely grateful to Mehran University of Engineering & Technology, Department of Software Engineering for all the assistance and technical support to accomplish this work.

REFERENCES

- [1] S. M. Salleh, Z. Shukur, and H. M. Judi, "Analysis of research in programming teaching tools: An initial review," *Procedia-Social and Behavioral Sciences*, vol. 103, pp. 127-135, 2013.
- [2] S. R. M. Derus and A. Z. M. Ali, "Difficulties in learning programming: Views of students," in *1st International conference on current issues in education*, pp. 74-79, 2014.
- [3] V. Stoffova, L. V égh, and K. Siakas, "Using Animations for Improving Learning," in *BCS Quality Specialist Group's Annual 20th International Conference on Software Process Improvement - Research into Education and Training (INSPIRE) conference, 2018: British computer society, London, UK.*
- [4] L. V égh, "Javascript Library for Developing Interactive Micro-Level Animations for Teaching and Learning Algorithms on One-Dimensional Arrays," *Acta Didactica Napocensia*, vol. 9, no. 2, pp. 23-32, 2016.
- [5] T. Linden and R. Lederman, "Creating visualizations from multimedia building blocks: A simple approach to teaching programming concepts," 2011.
- [6] T. Rajala, M.-J. Laakso, E. Kaila, and T. Salakoski, "Effectiveness of Program Visualization: A Case Study with the ViLLE Tool," *Journal of Information Technology Education*, vol. 7, 2008.
- [7] A. Dahotre, Y. Zhang, and C. Scaffidi, "A qualitative study of animation programming in the wild," in *International symposium on Empirical Software Engineering and measurement*, 2010, p. 29: ACM.
- [8] J. C. Adams and A. R. Webster, "What do students learn about programming from game, music video, and storytelling projects?," in *Proceedings of the 43rd ACM technical symposium on Computer Science*, 2012, pp. 643-648: ACM.
- [9] S. Cooper, W. Dann, and R. Pausch, "Alice: a 3-D tool for introductory programming concepts," *Journal of computing sciences in colleges*, vol. 15, no. 5, pp. 107-116, 2000.
- [10] A. Rudder, M. Bernard, and S. Mohammed, "Teaching programming using visualization," presented at the *IASTED International conference on web-based education*, 2007.
- [11] M. Kölling, "The greenfoot programming environment," *ACM Transactions on Computing Education (TOCE)*, vol. 10, no. 4, p. 14, 2010.
- [12] S. Chen and S. Morris, "Iconic programming for flowcharts, java, turing, etc.," in *Annual SIGCSE Conference on innovation and Technology in Computer Science Education*, 2005, vol. 37, pp. 104-107: ACM.
- [13] Z. Zhou, "Evaluating websites using a practical quality model," 2009.
- [14] M. V. Wade, "Likert-type scale response anchors," *Clemson International Institute for Tourism & Research Development, Department of Parks, Recreation and Tourism Management, Clemson University*, 2006.
- [15] A. Dewani, S. Bhatti, M. A. Memon, W. A. Arif, Q. Arain, and S. B. Zehra, "Sign Language e-Learning system for hearing-impaired community of Pakistan," *International Journal of Information Technology*, vol. 10, no. 2, pp. 225-232, 2018.

Authors' Profiles



Dr. **Sania Bhatti** is working with the Department of Software Engineering, Mehran UET, Jamshoro, Sindh, Pakistan. She obtained her PhD degree from the University of Leeds, United Kingdom in 2010 under scholarship of faculty development program. Her research interests include modeling and simulation, communication networks and software engineering. She has published more than 20 journal papers and 12 International conference papers.



Ms. **Amirita Dewani** is currently a faculty member in Department of Software Engineering, Mehran UET, Jamshoro, Sindh Pakistan. She has received her M.E degree from Mehran UET, Jamshoro. Her research interests include Data mining, Machine learning, Software Engineering and Global Software Development. She has written various journal and conference articles at local and international level.



Ms. **Sehrish Maqbool Ahmed** is working with the Government polytechnic institute for girls Latifabad, Hyderabad, Sindh, Pakistan. She obtained her M.S in computer science from Isra university, Hyderabad in 2015. She received her B.S in Computer science from University of Sindh, Pakistan, in 2008. Her research interests include computer networks, operating system and network administration.



Dr. **Mohsin Ali Memon** is working with the department of Software Engineering, Mehran UET, Jamshoro, Sindh, Pakistan. He obtained his PhD degree from the Department of Computer Science, University of Tsukuba in 2014. His research interests include interaction technologies, life logging, privacy control methods and machine learning. He received his B.Eng. in Software Engineering and M.Eng. in Information Technology from Mehran University of Engineering and Technology, Pakistan, in 2006 and 2009, respectively.

How to cite this paper: Sania Bhatti, Amirita Dewani, Sehrish Maqbool, Mohsin Ali Memon, "A Web based Approach for Teaching and Learning Programming Concepts at Middle School Level", International Journal of Modern Education and Computer Science(IJMECS), Vol.11, No.4, pp. 46-53, 2019.DOI: 10.5815/ijmeecs.2019.04.06