TEST Engineering & Management

November-December 2019
ISSN: 0193-4120 Page No. 4774 - 4778

CheckMyCode: Designing Assignment Submission System with Web-Based Java Compiler

Afiqah M. Azahari*, Arniyati Ahmad, Syarifah Bahiyah Rahayu
Cyber Security Centre, National Defence University of Malaysia (NDUM)
afiqah.azahari@upnm.edu.my

Article Info
Volume 81
Page Number: 4774 - 4778
Publication Issue:
November-December 2019

Article History
Article Received: *5 March 2019*
Revised: *18 May 2019*
Accepted: *24 September 2019*
Publication: *23 December 2019*

Abstract:

Integration of the web-based system into the curriculum of computer science education has significantly increased over the past decade. In response to the growing numbers of students, needs of an electronic method as for receiving and processing students' assignments and examinations is required especially in programming courses. Most of the lecturers still using email as a medium for collecting programming task. To organize, download, compile, run and evaluate student's programming file is challenging and time-consuming. Therefore, to overcome the issues, a well-designed assignment submission system is needed to facilitate collections of the programming task. The study is focusing on developing a visualization tools which compiling, running and submitting the programming task via CheckMyCode system. CheckMyCode equips with cloud compiler that could be run via a browser. Then, a student could write, compile, run and submit their code through the web-enabled compiler system. The functionality of CheckMyCode including creating a task, assignment and task submission, web code editor, grading system, and task feedback. CheckMyCode helps to reduce and minimize human error, assist lecturer in term of processing and controlling the given programming task. It is anticipated that CheckMyCode system will provide an educational web-based application which assists student to gain experience in the programming field and overcome the learning curve issue.

Keywords: *web-based, compiler, assignment submission system*

1. Introduction

Public and privates' universities and colleges are the main provider of knowledge workers. Most of students gained their knowledge from the institutions through various fields like engineering, computer science, applied science, business, etc. Every student in any institution is tightly attached to the course requirements. Every semester student needs to deal with assignments, tutorials, quizzes and test to assess their level of knowledge and understanding contributes to a grade as part of their final achievement. For Computer Science courses, most of the course isstructured in two ways whichare face-to-face learning and tutorial. For example, studentslearn theory and programming language in class, and apply and test in the practical (hands-on) tutorial in computer laboratory. Lecturer will assign programming assignments to be completed individually or in group. At the same time, they might need to complete assignments from other courses as well. There are many ways of completing and sending the assignments as required by the lecturers. Some lecture prefers students to send their assignment in hardcopy, while other lecturer wants them in softcopy. All submitted assignments areassessed. The assessment of the assignments involved programming code is the most challenging task. The lecturer needs to compile and checkthe output whether the algorithm works as required. This really challenging and time-consuming when it involves many students. To overcome the problems, this paper proposeda design and development of a visualization tool named as CheckMyCode. The system provides a centralized solution, which provides submission, assessment and feedback of students' assignment. This web-based system will receive, compile, run and assess students' programming code.

2. Research Background

There are both static and dynamic assessment tools for computer programs. Assessment tools can be used to help the instructor/lecturer in grading task as well as giving automatic feedback directly to student – survey could be found in (Ihantola, Ahoniemi, Karavirta, &

Seppälä, 2010). Tools that support online programming lecturers can be divided into 3 type such as visualization tools (Glassman, Scott, Singh, Guo, & Miller, 2015), automated assessment tools (Staubitz, Klement, Renz, Teusner, & Meinel, 2015) and support tools (Tominaga & Ota, 2017). Literature often focus on source code analysis with developments on semantic and conceptual analysis of code for a better evaluation. The focus here to develop a teaching and learning support tools in programming class. For instance, National Defense University Malaysia (NDUM) offers two main programming languages C++ and Java. Currently, lecturers using medium such as email or e-learning system as for submitting programming task or assignment. To organize, download, compile, run and evaluate student's programming file one by one is challenging and time-consuming. Therefore, the proposed intelligence tool will assistNDUMstudents gain experience in programming field and overcome the learning curve issues. CheckMyCode system could be easily integrated with programming class in NDUM. Students are allowed to write the program using a web browser to get immediate and accurate feedback for personal evaluation. This project aimed to design and develop a tool where students and lecturers could benefitsboth. For lecturers, evaluation of programming task and time management problems couldbe tackled, while the student will be able to spot their difficulty domains and practice even more. In CheckMyCode system, students and lecturers are allowed to compile and runs the program via a web browser without having started a language compiler separately.

3. Literature Review

Formative Feedback

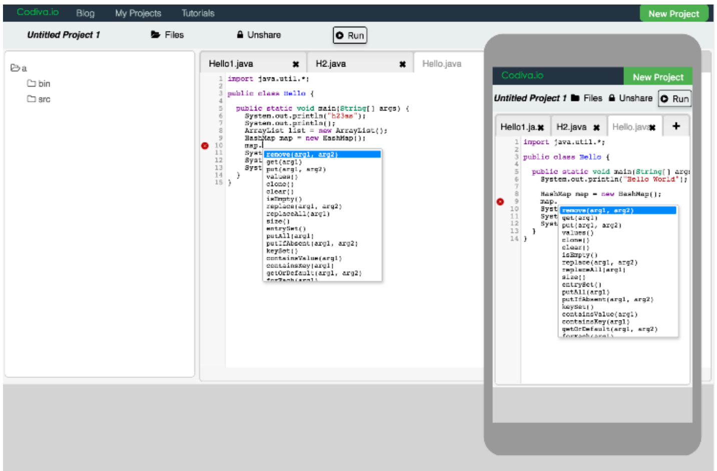
In learning, students need to see immediate results for instructional improvement. Formative feedback is a piece of information given to learners in response to an action that has taken by learners. It comes in a variety of types such as explanation to the correct answers, hints or worked example (Shute, 2008). The course and their learnt will be more valuable to them. Moreover, formative feedback in learning allows an instructor to check on or follow up the student performance throughout the semester. Today, the computer system has been recognizing as an effective tool to support the provision of formative feedback. Formative feedback should be integrated with student's online learning space. Feedback could be share immediately to students and will motivate the student to engage with the feedback given. Moreover, to achieve course objective, empirical evidence suggests that a computer system could address the issues of student engagement and increase student motivation in learning (Hatziapostolou & Paraskakis, 2010).

21st Century Learning Model

One important thing that a country need in order to develop and prosper is education. Education would moulds individual character and individual. Current technology driven world has change the process of acquire of knowledge, which reconceptualise the traditional classroom.It also believe that online education has been one requirement intoday generation education system(Malate, 2011). Collaborative learning, use of ICT equipment either software or hardware and formative feedback was promoted by 21st century learning model (Chai & Kong, 2017). Today modern education is very different from traditional education. The methodology to deliver today’s learning is very interactive. Used of technology is necessary in today’s classroom as an evolution of technology helps student to stay engaged in today’s boundless world.Teachers’ now using multimedia presentation as a ‘mandatory’ teaching aids. Quiz and test were do online. Platform such as online assignment management system were used by most institution in order to helps educators to keep track submission, marking and return on students’ assignment, project or test.

Web based code editor and compiler

Build-in web-based code editor allows students to code directly within the web browser. Students and lecturers only need one program on a browser such as Chrome or Firefox to compile and run their code. Editors will have features specifically designed to simplify and speed up typing of source code, such as syntax highlighting, indentation, autocomplete and brace matching functionality. These editors also provide a convenient way to run a compiler, interpreter, debugger, or other programs relevant for the software development process. Most web-based code editor comes with the multiple numbers of language modes including Java and C++. Most web-based code editor is open sources. Codiva is an example of Java online compiler available in the web. Codiva is open-source Java compiler. It can be reached via the codiva.io link. Codiva features including auto-completion (Figure 1), compiles as users type and parses compilation errors in the code editor section. However, Codiva only supports Java, C and C++. It also does not support sharing of code written and compiled to other users through the web.



```
import java.util.*;

public class Hello {

  public static void main(String[] args) {
    System.out.println("h23ss");
    System.out.println();
    ArrayList list = new ArrayList();
    HashMap map = new HashMap();
    map.
    Syst
    Syst
    Syst
  }
}
```

```
remove(arg1, arg2)
get(arg1)
put(arg1, arg2)
values()
clone()
clear()
isEmpty()
replace(arg1, arg2)
replaceAll(arg1)
size()
entrySet()
putAll(arg1)
putIfAbsent(arg1, arg2)
keySet()
containsValue(arg1)
containsKey(arg1)
getOrDefault(arg1, arg2)
```

```
import java.util.*;

public class Hello {

  public static void main(String[] arg
    System.out.println("Hello World");

    HashMap map = new HashMap();
    map.
    Syst
    Syst
    Syst
  }
}
```

Figure 1 Auto completion and error handling in Codiva

putenv() function

To run a Java code without specific compiler, a specific function to sets the value of server environment variable was needed. putenv() function could be used add setting to environment variable. The function will exist for the duration of the current request. At the end of the request, server's environment variable will be restored to its original state. Moreover, this function will be return 0; on successful completion. Otherwise, it will return non-zero value and set errno as to return the error number. However, putenv() function maybe fail if not sufficient memory was available (Holt & Huang, 2018). For CheckMyCode system, student's Java code submitted to web server need to compile via command prompt. Therefore, CheckMyCode system need to set the path of Java Development Kit (jdk) file in environment variable. Code below shows an example implementation of putenv() to change the path of jdk file using php programming language.

```
<?php

   $newPath = "C:\ProgramFiles\Java\jdk-  10.0.1\bin"
   putenv("PATH=$newPath");

?>
```

4. Methodology

This section outlines the work that the researcher needs to accomplish in developing this system. This project aims to develop an interactive web-based application, which is used by computer science student to submit programming task. The focus here concerns on developing a visualization tools which compiling, running and submitting the programming task could be done in one system. CheckMyCode system will be develop using Agile software development methods. Professional will join in building the continuous prototyping based on end-user requirement (NDUM's student and lecturer). Agile software development is chosen due to its popularity because it emphasized collaboration with customers, communication between the developer, and fast delivery of the product and demand change of the requirement (Soni & Kohli, 2017). Furthermore, Agile could respond to changes quickly. It allowing the developer to go back to a previous stage and carry out necessary changes in terms of refining the system without much delay as compared to the conventional system development methods (Misra & Singh, 2015).

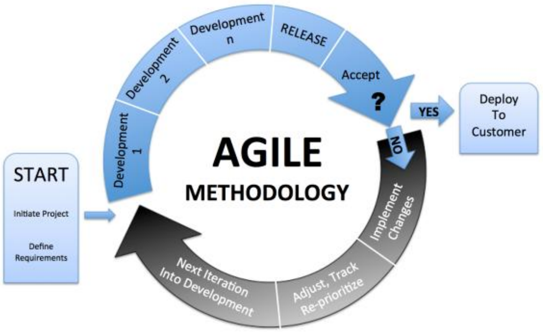


Figure 2 Agile System Development Life Cycle

An agile software development process always starts by defining the users and documenting a vision statement on a scope of problems, opportunities, and values to be addressed as seen in Figure 2. End-users will capture the vision of the system from the team of users and deliver the vision to the developer. Next, multidisciplinary software development teams will work together to get the job done. Not the whole application, firstly, the developers will build a part of the system database, business logic and user interface. To do this the developers will collaborate among themselves on what and how they are developing the system. Then, part of the system that has been developed will be presented and reviewed by end-user. Completed and the accepted system will be deployed for end-user. However, the uncompleted system will give back to the developer for enhancement with acceptance criteria from end-user that define a solution for the accepted system. Implement agile principle in developing web-based system usually give a better quality and faster developing application and better technical practices.

5. Proposed CheckMyCode System Architecture

A system architecture is an abstract design concept of a system. System architecture shows a structure of moving parts and its connection. The diagram below (Diagram 1) shows a system architecture of CheckMyCode system.

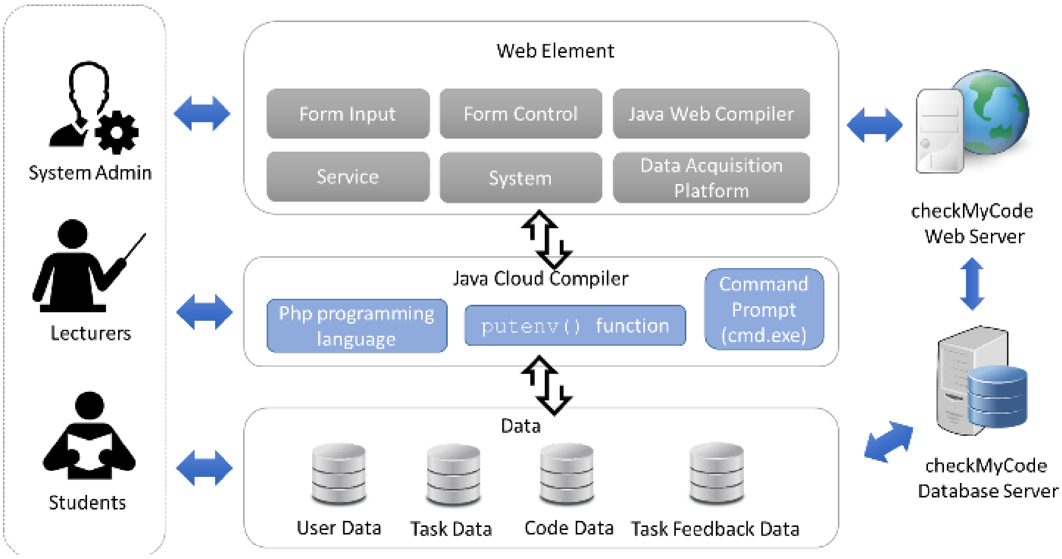


Figure 3CheckMyCode proposed architecture

CheckMyCode system is considered as an enhancement to previous programming task submission and management system. CheckMyCode system should solve the list of current problems as follows:

1. Students often forgetting task submission dates.
2. Students are not notified if their new task need to be submitted via email.
3. Student often miss to put their name when they submit their programming task via email to lecturers.
4. Unrecognized files submitted by a student leading to the cumbersome task of recognizing the owner of code.
5. Lecturers need to give the task's feedback to student one by one due to submission via email.

The proposed CheckMyCode system consists of a web server, a web server, an interactive front end user interface module, online Java compiler module and specific user's site. The web server will host the CheckMyCode system to users. To get and store data from users, a web server need to be connected to a database server. Through the web server, lecturers could do the following:

a. Create programming task
b. Assign task to specific group of students
c. Give student's task feedback through the web.
d. Give a mark to every task created.

On the other hand, students willbe notified if there is any new programming task assigned to them. Moreover, students are also capable to write, test, run and submit the code to the lecturer.

6. Conclusion

The proposed new system will introduce a new approach in learning. CheckMyCode system facilitates the submission of programming task to lecturers. Additional elements like online Java compilers will ease the process of testing, submitting and evaluation of programming task. Furthermore, with the submission management system, a task that submitted to lecturer will become more organized. Formative feedbacks could be sent to students immediately. Besides, the framework is easy to adapt for any programming task such as algorithm development, data structure or object-oriented programming.

References

1. Chai, C. S., & Kong, S.-C. (2017). Professional learning for 21st century education. *Journal of Computers in Education*, *4*(1), 1–4.
2. Glassman, E. L., Scott, J., Singh, R., Guo, P. J., & Miller, R. C. (2015). OverCode: Visualizing variation in student solutions to programming problems at scale. *ACM Transactions on Computer-Human Interaction (TOCHI)*, *22*(2), 7.
3. Hatziapostolou, T., & Paraskakis, I. (2010). Enhancing the impact of formative feedback on student learning through an online feedback system. *Electronic Journal of E-Learning*, *8*(2), 111–122.
4. Holt, A., & Huang, C.-Y. (2018). Overview of GNU/Linux. In *Embedded Operating Systems* (pp. 11–40). Springer.
5. Ihantola, P., Ahoniemi, T., Karavirta, V., & Seppälä, O. (2010). Review of recent systems for automatic assessment of programming assignments. In *Proceedings of the 10th Koli calling international conference on computing education research* (pp. 86–93). ACM.
6. Malate, R. F. (2011). Impact of Corporatization on Higher Education. *Liceo Journal of Higher Education Research*, *7*(1), 225–235. https://doi.org/10.7828/ljher.v7i1.24
7. Misra, S. C., & Singh, V. (2015). Conceptualizing open agile software development life cycle (OASDLC) model. *International Journal of Quality & Reliability Management*, *32*(3), 214–235.
8. Shute, V. J. (2008). Focus on formative feedback. *Review of Educational Research*, *78*(1), 153–189.
9. Soni, D., & Kohli, P. J. (2017). Cost estimation model for web applications using agile software development methodology. *Pertanika Journal Of Science & Technology*, *25*(3), 931–938.
10. Staubitz, T., Klement, H., Renz, J., Teusner, R., & Meinel, C. (2015). Towards practical programming exercises and automated assessment in Massive Open Online Courses. In *Teaching, Assessment, and Learning for Engineering (TALE), 2015 IEEE International Conference on* (pp. 23–30). IEEE.
11. Tominaga, H., & Ota, S. (2017). A Proposal of Assistant Support Tool in Classroom Round and Personal Tutoring with a Seating Chart for Introductory C Programming Exercises. In *2017 21st International Conference Information Visualisation (IV)* (pp. 372–377). IEEE.