

*Development of an Introductory Programming Education System
by Automatic Generation of Lecture Slides*

Tatsuyuki Takano, Kanto Gakuin University, Japan
Osamu Miyakawa, Tokyo Denki University, Japan
Takashi Kohama, Tokyo Denki University, Japan

The Asian Conference on Society, Education & Technology 2016
Official Conference Proceedings

Abstract

The learning of programming has become easy because the online learning of programming has spread globally. Such e-learning systems execute programs in which students can create and evaluate programming exercises. Coding a program has a number of steps, such as declaration variables and methods. However, e-learning systems for general programming do not evaluate programs in the middle of coding. We developed the system evaluated in the middle of creation of a student's program. Thus, on detecting an error in the program, the system will point it out. In this system, since it had pointed out by the correction videos prepared beforehand, concrete directions cannot be performed. Hence, we developed a system that can automatically generate slides containing contents of advice. This paper describes our system in detail.

Keywords: programming education, software engineering, e-learning

iafor

The International Academic Forum
www.iafor.org

Introduction

The learning of programming has become easy with the global spread of the online learning of programming. Such e-learning systems execute programs in which students can create, as well as evaluate, programming exercises. The coding of a program involves many steps, such as declaration variables and methods. However, an e-learning system for general programming does not evaluate the program in the middle of coding. Hence, we developed a system that can evaluate a student's program in the middle of coding (Tatsuyuki, Kentaro, Takashi, & Osamu, 2015). Therefore, if the system finds an error in the program, it will point out the error. In the system, since it had pointed out by the correction videos prepared beforehand, concrete directions cannot be performed. We developed the system to generate automatically slides of the contents of advice. This paper describes the details of the system.

The flow of learning using the system

Candidates are students of the beginner who learn Java programming. The contents exercised using the system are creating the skeleton of a program from the class diagram of UML.

A student learns the correspondence of a class diagram with a program by creating a skeleton of the program using the system. Figure 1 shows the flow of learning using the system. The main study methods are the same as those used in the previous system. A student creates a program with reference to the class diagram. Further, if a program has errors, the system will point them out and the student will correct the program according to the indication. The indicated error is shown as not only a compilation error but also a class or method name misspelling. There is no major change in the algorithm that judges errors. The differences between the previous system and this one are shown as follows:

1. The lecture video is changed into a lecture slide.
2. When there are two or more errors, the system indicates more than one.
3. When the correction of an error is finished, it is judged immediately.

First, the system can now give directions to a student by specifying a concrete class name in a lecture slide. Second, while only one screen was displayed in the previous system, multiple screens could be displayed simultaneously in this system. Furthermore, this system can point out an error precisely by using the character string of the error parts. Third, the earlier system judged when the correction of an error in a program was over and the cursor moved to the following line. However, this system judged correctly for every character.

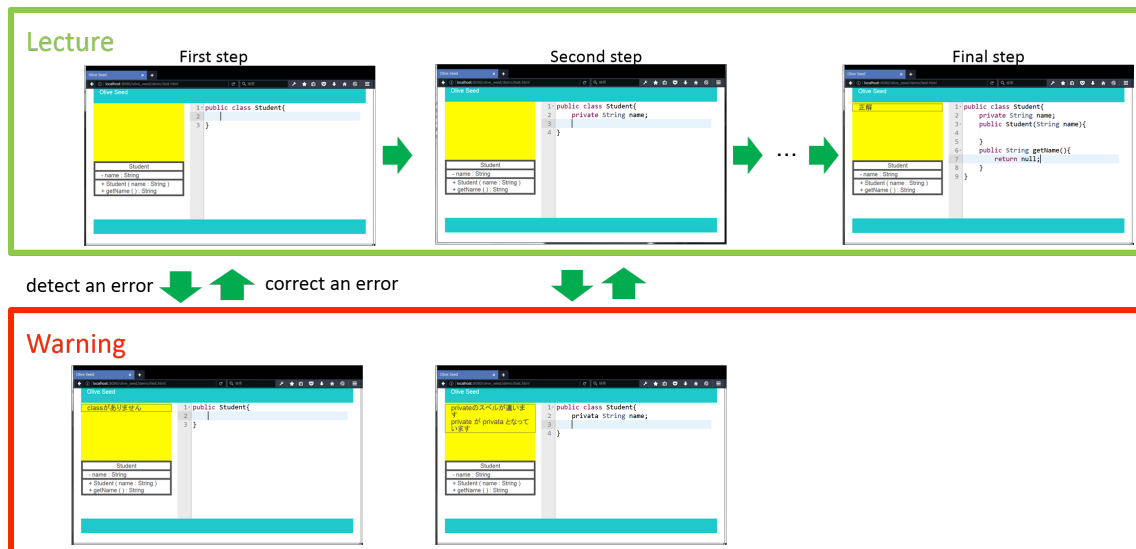


Figure 1: Flow of learning using the system.

The appearance of the system

Figure 2 depicts a general view of the system. It is developed as a web system, and a student uses the system using a web browser. The composition of the screen is shown in Figure 3.

The screen composition of the system is as follows:

1. Lecture slide
2. Class diagram
3. Editor

A lecture slide gives three types of information: Displays the directions of the creating program, indicates errors, and judges the completion of the program. A class diagram shows the declaration of the program to exercise. The editor is equipped with the auto-indent function and syntax highlight of Java.

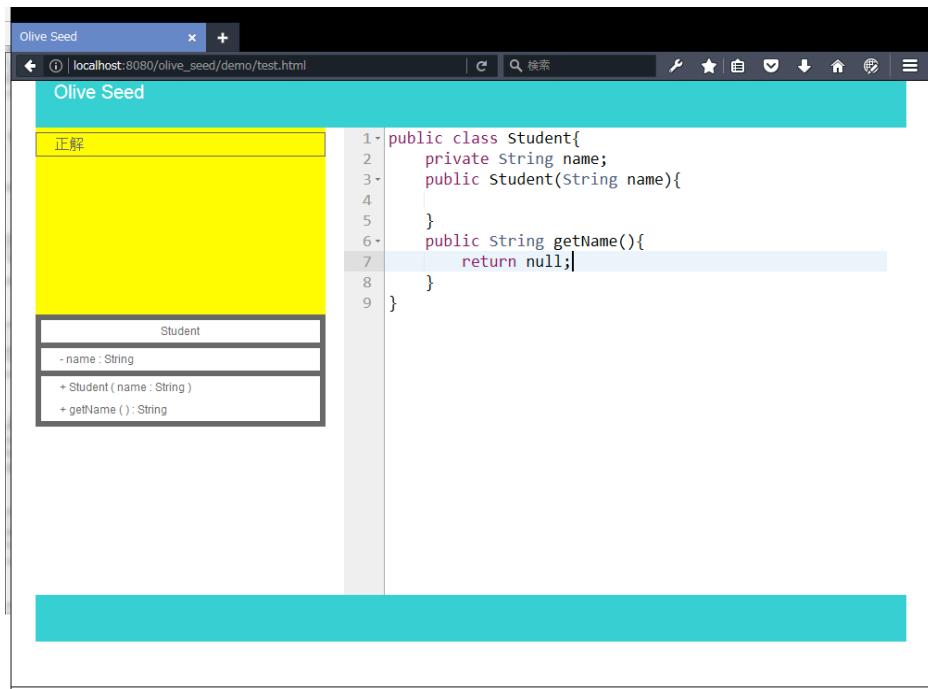


Figure 2: Appearance of the system.

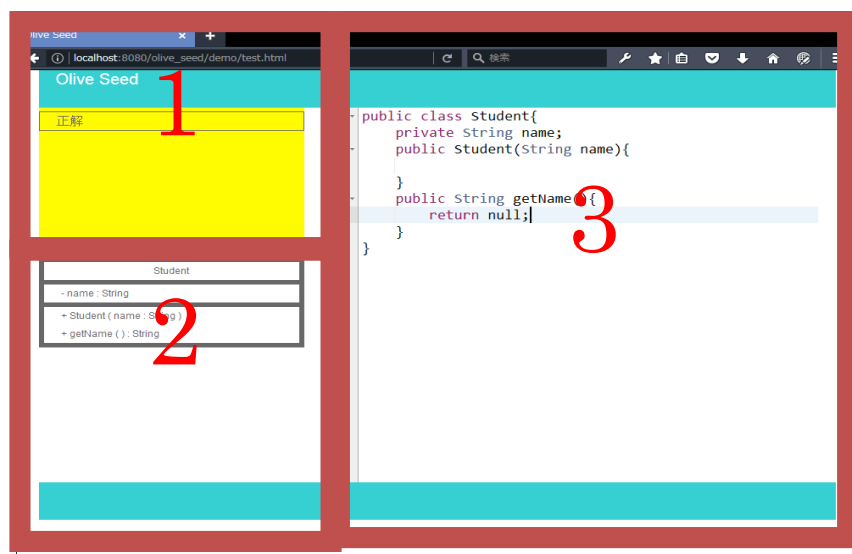


Figure 3: Composition of the screen.

System Integration

The composition of the system is shown in Figure 4. It consists of server client forms. The server is developed using Grails, which is a web framework created by Groovy. Groovy is one of the programs operated by the Java Virtual Machine.

On the server side, the syntax analysis of a correct answer program and collection of a student's data are performed. The client side consists of mainly HTML and JavaScript. JavaScript is performing the generation of a lecture slide, lexical analysis of a student's program, and judgment of errors. An error shows the judgment result of a

spelling error, inserting words, and deleting words for a class name, a method name, or a reserved word in Java. Further, the kind of sum total is 69 patterns. The data of the syntax-analysis result of a correct answer program, and so on, are acquired using a JSON form.

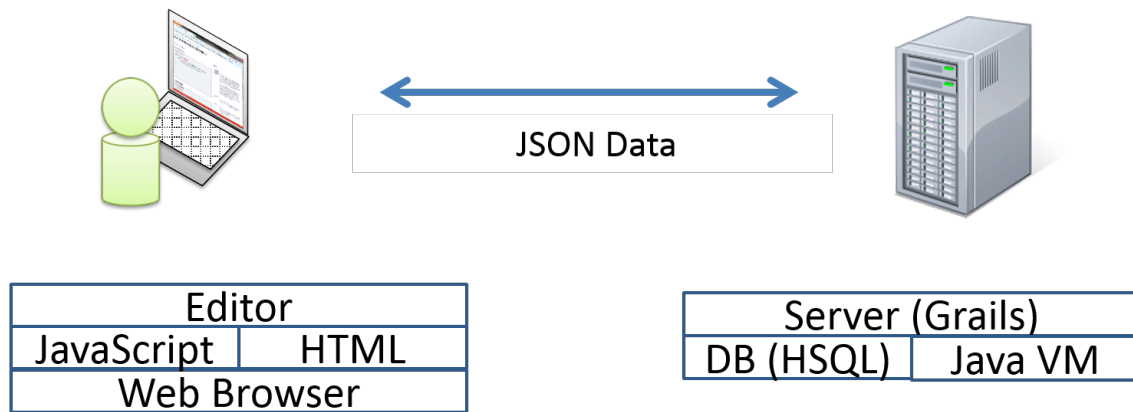


Figure 4: System integration.

Conclusion

In this research, when a student created a source code in accordance with a lecture video and an error was detected in the student's code, the system switched to a slide that urged error correction. The error correction slides of this system support 69 major types of errors made by students. The developed system was able to indicate two or more errors. Moreover, the system is ready for test employment in a real-world lecture setting, to acquire student data, and to be evaluated.

References

Tatsuyuki, T., Kentaro, M., Takashi, K., & Osamu, M. (2016). Development of a system using interactive video for novice programmers. Proceedings from *The Asian Conference on Society, Education & Technology 2015*, 195–200.

The Grails Framework. Retrieved from <https://grails.org/>.

The Groovy programming language. Retrieved from <http://groovy-lang.org/>.