A Web-Based Environment to Improve Teaching and Learning of Computer Programming in Distance Education

S.C. Ng, S.O. Choy, R. Kwan, and S.F. Chan

School of Science and Technology, The Open University of Hong Kong
scng@ouhk.edu.hk

Abstract. Learning computer programming is not an easy task. Students need to spend hours doing practical activities in order to comprehend the techniques of writing computer programs and beginners usually face a number of obstacles associated with installing and using a compiler or integrated development environment. This paper introduces an online web-based system that provides an interactive integrated environment for students doing programming activities and coursework in a distance learning institution. The interactive system provides students with timely and effective feedback about programming activities without the need to have instructors and students meet at the same time and the same place. The web-based system provides students with an editing, compiling, testing and debugging environment for learning computer programming on the web. Instructors can monitor the learning progress of students, compile the student's program and view the error messages through the student's workplace in the online system.

1 Introduction

Computer programming courses are core components of many university programmes in Computer Science, Engineering, Information Technology, and other related fields. The Open University of Hong Kong (OUHK) offers three courses on computer programming at different levels for undergraduate programmes. The population of students taking programming courses is also large. In OUHK, a first year programming course enrolled over 400 students in the past academic year.

Teaching computer programming has never been a straightforward process. Programming is a skill that requires practice. In order to strengthen students' programming knowledge, students need to do a lot of programming practice. Therefore, practical activities play an important role in the learning process [1]. Traditional web based learning systems have not made full use of the advanced Internet computing technology to improve the quality of teaching and learning. Many people use technology simply to replicate the existing ways of teaching and learning at a distance, such as posting online teaching materials and providing electronic submission for the students' assignments [2]. These kinds of learning systems do not

R.W.H. Lau et al. (Eds.): ICWL 2005, LNCS 3583, pp. 279 – 290, 2005.

provide added value in improving the quality of teaching and learning computer programming in distance education.

With the increase of students taking fundamental programming courses and increasingly limited teaching resources, the tutoring time spent on each student is reduced. As a result, students may suffer from the lack of timely feedback from their instructors. It may lead to poor results in learning computer programming. In a traditional university education setting where students can often meet their fellow students and instructors, the teaching process is significantly improved by conducting practical workshops in which teachers are present to serve the learning needs. Students can also interact with their fellow students in a computer laboratory to enhance the learning process.

However, in a distance learning setting, the education process is distributed. For most of the time, students and instructors are geographically remote from each other. Students are given a set of study materials that consists of traditional textbooks and printed study units. To enable students to carry out programming activities at home, a set of software including a compiler and program development environment is also provided. However, it is both time consuming and frustrating for beginners to learn how to use the integrated development environment (IDE), set it up, and become familiar with the programming environment [3]. According to Proulx [4], the first course in programming is a major stumbling block for novice programmers. Most beginners find problems in setting the environment variables, such as PATH and CLASSPATH, during the installation of Java Development Kit (JDK) in their machines. Some may fail to manage a suitable Java integrated development environment (IDE). It was not uncommon to find that some students are still unable to install the Java JDK properly even after ten weeks of study in an introductory Java programming course in OUHK.

Besides, distance learners spend most of the time working alone. They usually have little or no contact with their fellow students. When they find a problem during their learning, they usually ask their instructors through the telephone or email. It is difficult for the students to express their programming problem clearly in a textual message or vocal communication with their instructors. This kind of debugging process could be prolonged and ineffective. From an instructor's perspective, it is not easy to keep track of students' study progress and give timely feedback to them in a distance learning setting.

This paper describes how the web-based system adopts the Internet technology in a more efficient way to help both instructors and students to teach and learn computer programming respectively in a distance learning environment. Our web-based system aims to improve the effectiveness of teaching elementary Java programming in a distance learning environment. The system provides a virtual workplace for performing programming activities. It provides an online and integrated environment for writing, compiling and testing programs. Students just need to use an Internet-connected computer to access the system for doing programming activities. They don't need to prepare and set up the whole programming environment and IDE in their own machines. The system's file server stores all the student's intermediate and submitted work. The system allows students to do practical exercises by filling in a

template-like programming editing area. When a student wants to compile and run the program, the system will generate the resulting output in Java Archive format (JAR) and return it to the student. The student can run the program in his or her own machine. In case compilation errors occur in the program, the system will generate useful and helpful hints for students to debug the program. The system also includes a communication tool between students and tutors. Whenever a student needs further help, the instructors can directly access the student's workplace, compile the student's program and view the error messages through the online system. All student activities will be recorded in the system so that instructors can use the log data to keep track of the student's progress. In addition, the system provides syntax highlighting in editing program code, which could help novice programmers understand and remember the program code more easily. In short, the new system makes use of the current Internet computing technology to provide an online environment for practical programming activities.

Related work has been found in the literature. CourseMaster [5] was developed as a computer-based tutoring system, which mainly provides features for automatic assessment handling of students' work. Since it is not a web-based system, students still need to learn how to use the IDE first, and that could hinder the students' learning process. ELP [6] provides a web-based interface for students to do programming activities. However, ELP provides limited support for students to edit their programs and get help from the instructors.

The organization of this paper is as follows. Section 2 of this paper shows the system architecture. Section 3 describes the system overview with the functions for both students and instructors. Section 4 discusses the system workflow with the descriptions of user interface generation and compilation result generation respectively. Section 5 discusses the pilot run of the system as a programming learning environment in the presentation of an OUHK course. Finally, conclusions are drawn in section 6.

2 System Architecture

Fig. 1 shows the overview of the architecture of our system. A Content Management System (CMS) runs on the Apache Web Server with MySQL database and provides basic management functions of the website. The system consists of four main sub-modules written by PHP scripts — Activity Parser, Activity Compiler, Student Activity Logger and Activity Log Analyzer — to perform the online programming activities and student performance monitoring service on the web.

There are several interactions between client and server sides. When the client browses an activity, a request will be issued to the server to access a particular programming activity. By using the Activity Parser module, the server parses the Activity XML and loads all the necessary files from the file server. If the request is not a student's first attempt at a particular activity, the Activity Parser module will load the previous work from the Student Activity file. After the parsing and loading

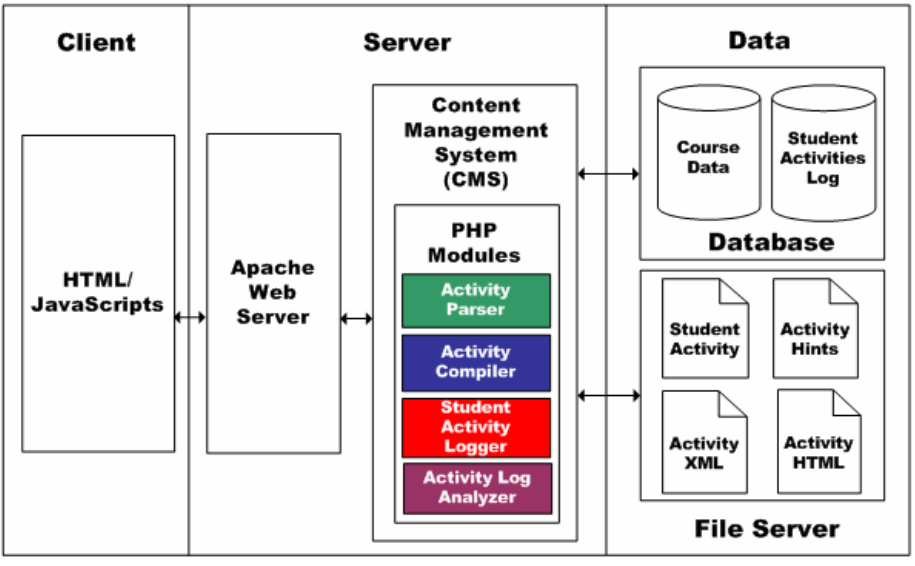

Fig. 1. System Architecture

process, the server will send the activity content in the form of HTML format back to the client. Other than loading the activity, the server also allows the client to compile their programs online. After receiving the client side compilation request, the server compiles the program from the student's working directory. If no error is found, the server will generate a Java Archive (JAR) file for the student to download. Otherwise, an error log will be generated in the current activity, and the error message will be passed back to the client for debugging.

All of the above interactions will be captured by the Student Activity Logger and stored in the Student Activities Log database. By using the Activity Log Analyzer, the tutors are allowed to access the student's performance. The Activity Log Analyzer will make use of the Course Data and Student Activities Log Data to perform those analyses for monitoring the student's progress.

3 System Overview

This section describes how students and instructors interact with our system. Our online system provides both students and instructors with an environment for learning and teaching programming on the web. Instructors can create and upload the programming activities materials to the system through the web browser. The instructors can upload the activities description file, activities template file and the hint files. Activities description file provides the basic description and information about the activities; activities template file provides students with a template to work

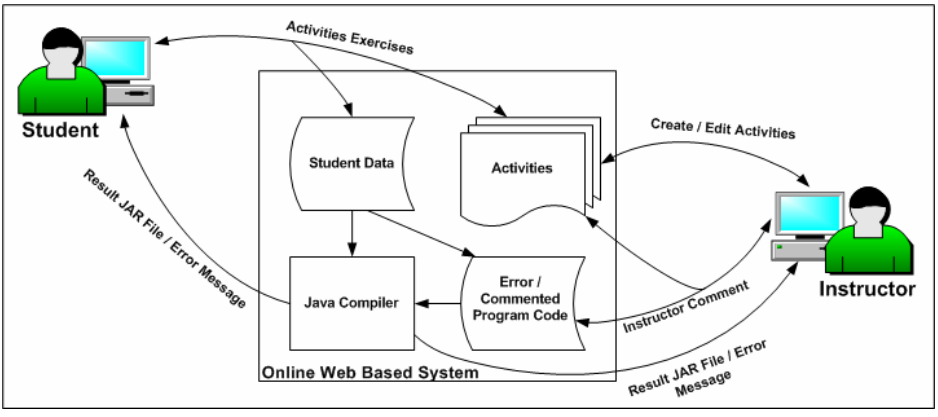

Fig. 2. The Web Based System Overview

on an activity; and hint file provides students with a sample output of the required program for reference.

Fig. 2 illustrates the interaction between the students and instructors on the web-based system. Students can navigate through the activities and attempt each of them. Students are allowed to compile the program at the server side, and the server will send back the resulting ".class" files of the activity packed with other libraries in a Java Archive (JAR) file. In case of syntax errors in the program, the corresponding error messages will be sent back to the student. If a student finds any problem in handling the activities, he or she can send a help message to the respective instructor for help. The system will then copy the student's program code to a temporary directory. When the respective instructor logs into the system, he or she can view the error code directly from the student's temporary workplace. The instructor can compile the program and see the error messages from the online compiler. After fixing the program errors, the instructor can just reply to the student's message and give comments in the program code. The student can view the instructor's comments and suggestions in the reply.

3.1 Student View of the System

Fig. 3 shows how the students walk through and perform the programming activities online. Students can navigate the programming activities by using the web browser. After selecting the programming activity, a text editor is provided for the student to edit the program code. After the student finishes editing the program with the given template, he or she can compile the program by clicking the compile link on the web page. The program code will be sent to the server side for compilation. If there is no error during the compilation process, the system will automatically generate the executable Java Archive (JAR) file back to the student. Otherwise, the system will generate an error message back to the student for debugging. If the student has

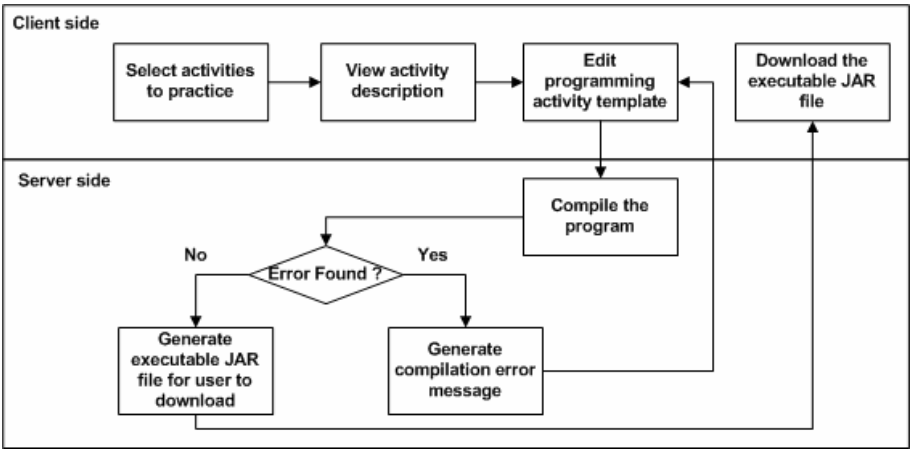

Fig. 3. The flow of performing online programming activities

Fig. 4. Activity Description

installed the Java Runtime Environment (JRE) in his or her own machine, he or she can simply click the “Open” button to run the program immediately. In addition, a

reset function is also provided for students to reset the activities back to the original programming template.

Fig. 4 shows the "Activity Description" interface — students are provided with detailed information about an activity. Fig. 5 shows the "Program Source" interface —

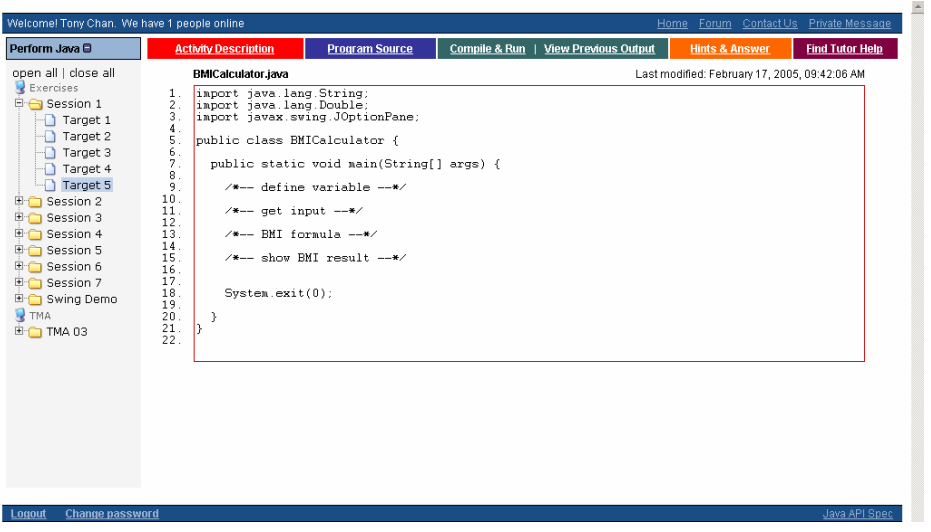

Fig. 5. Program Source Area

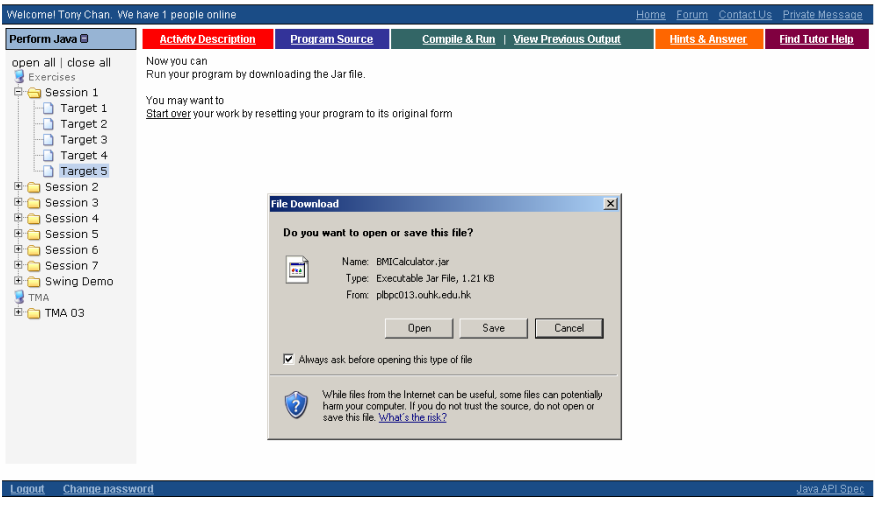

Fig. 6. Compile and Run the program

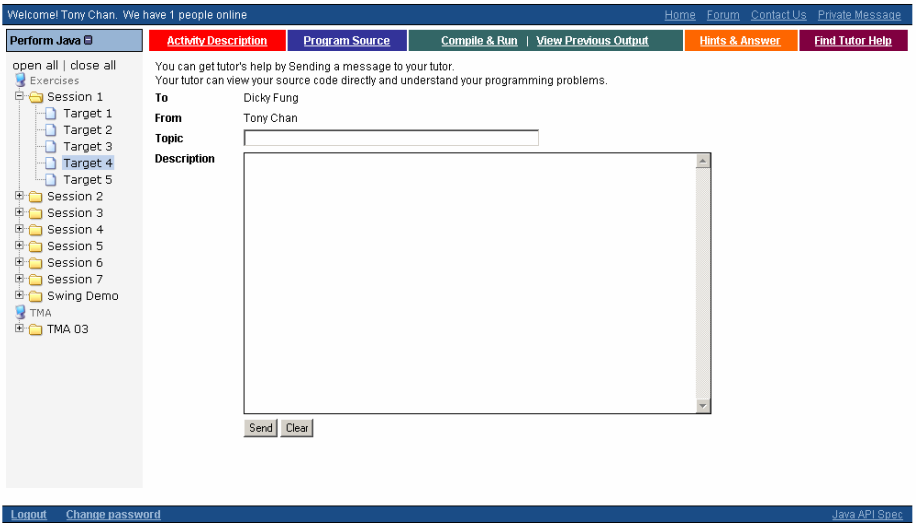

Fig. 7. Find Tutor Help

this area provides a simple editing area for students to write and develop their programs. Fig. 6 shows the "Compile & Run" interface — this area allows students to compile and run their programs directly. The "Hints & Answer" interface provides a sample program output for students' reference.

When a student encounters problems in doing the programming activities, he or she can request help directly through the system. By using the help function, instructors can view the student's program code directly. The student only needs to send a help message to their instructors by clicking the "Find Tutor Help" link in the system. The interface for "Find Tutor Help" is shown in Fig. 7. His or her program will then be copied to a temporary folder inside the server. Once the instructor logs into the system, he or she can directly view the help message from the student with the program codes. The instructor can then compile the program in the system and view the errors. The instructor can provide comments in the program code and reply to the student's message. The student will then receive a reply message from his or her instructor. The message contains a hyperlink that allows student to view the instructor's comments in the program code.

3.2 Instructor View of the System

The system provides a set of functions for instructors to manage the online programming activities. The instructors can create, edit or delete the online activities through our system. It provides a simple interface for instructors to upload the online programming activities files to the server (see Fig. 8). An activity contains three types of file; they are activity description files, activity programming code template files

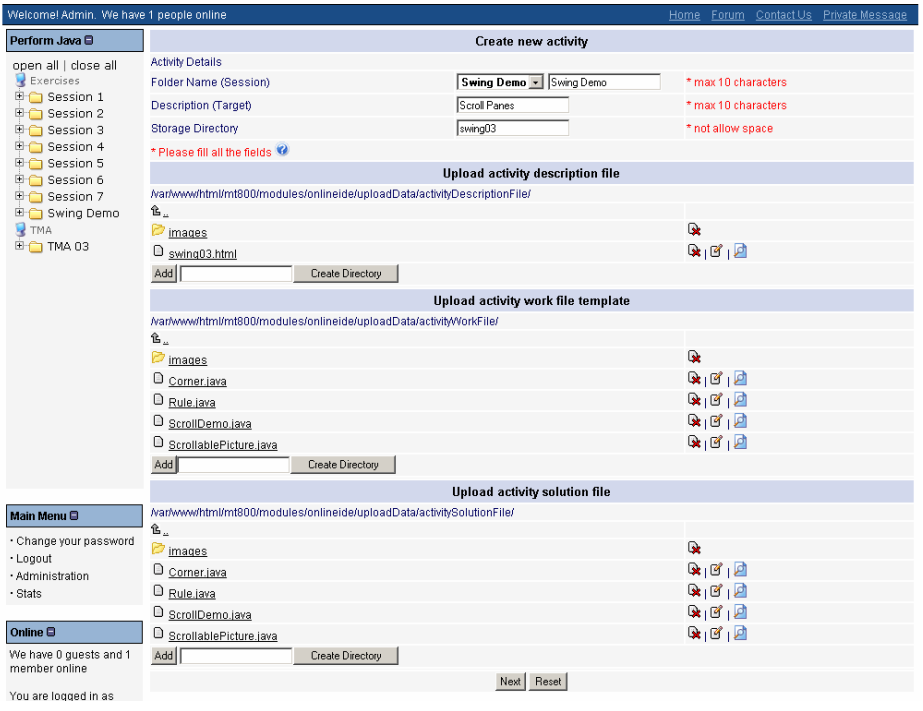

Fig. 8. Interface for instructor to create new activity

and activity hint files. Activity description files are mainly composed of HTML code and related image files. The system also provides a simple HTML text editor for instructors to edit the activity description files online. The activity programming code template files provide students with programming templates to work on as they practice. The files will be loaded into the student data folder when the students first attempts or resets the activity. In order to provide students with a sample output for reference, instructors can upload the activity hints file to the server so that the student can view the activities sample outputs. The hint files then will compile into Java class files and pack into an executable JAR file for students to download.

4 System Workflow

In our system, each activity is described and controlled by an eXtensible Markup Language (XML) file. The Activity Parser module in Fig. 1 is responsible for generating exercises from the XML files and related activities files to be converted into HTML format. Fig. 9 shows the flow of the activity interface that can be generated from the server.

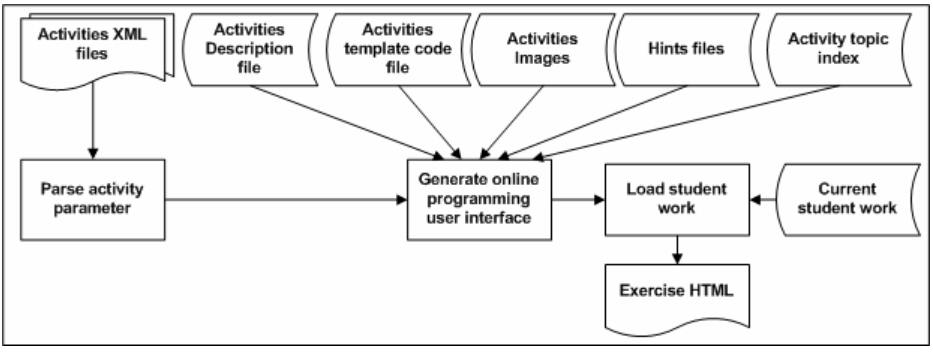

Fig. 9. Steps to generate online programming activity

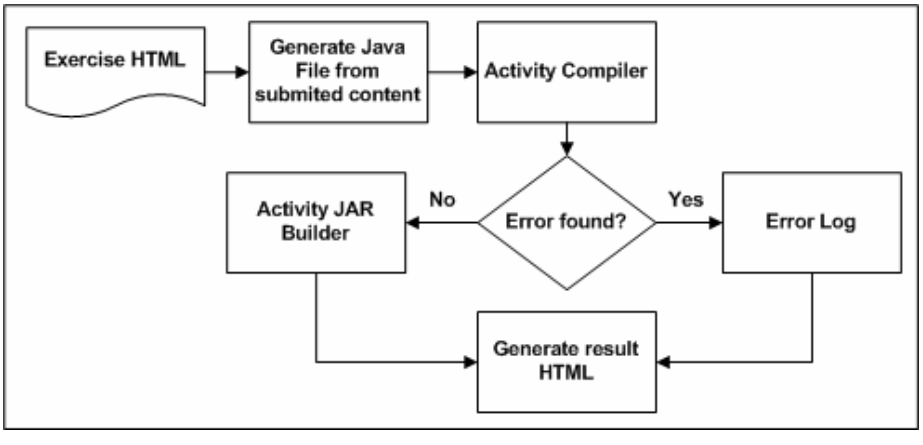

Fig. 10. Generate compilation result

After the students log into the system, there is a tree menu for students to choose which activities to practice. The activities tree menu is generated from the activities XML files. The system will parse the entire activities XML files and then generate the tree menu dynamically.

When the student selects a particular activity to practice, the system will generate the user interface according to the activity XML file information. The activity XML file will provide all the information about the activity such as activity description file name, activity template code file name and main class name. If the student has attempted the activity before, the system will load the student program code back into the simple editor. Finally, the activity HTML will be sent to the student.

The Activity Compiler in Fig. 1 is responsible for compiling, building and making a JAR file from a student's program code. Fig. 10 shows the workflow of how the server generates compilation results. After the student finishes editing the program code, he or she can then click the "Compile and Run" link to send the program to the

server side for compilation. When the server receives the student's program code, the server will generate the Java files first. Then the system will rely on the activity XML file instruction to compile the Java files. If no error is found, the system will execute the jar command to make an executable JAR file for the student to download. In order to let the client side computer run the JAR files easily, the executable JAR file, which has been sent back to the student is described by the Multipurpose Internet Mail Extension (MIME) type content type "application/java-archive". The student can then double click the JAR file to run the program and view the output result directly.

5 Discussion

We have piloted the use of this web-based system as a learning tool for the course *MT800 Information Technology and Software Development* in OUHK. Most students in this course do not have prior Java programming experience. We used the system as an online, interactive learning environment for students to learn Java programming. The students were given a set of online activities to work on at their own pace. Since the materials can be assessed easily through the Internet, students can learn programming with greater flexibility, anytime and anywhere. Students are not required to set the environment variables such as PATH and CLASSPATH when installing JDK in their machines. The only requirement is to have the Java Runtime Environment (JRE) — all the programs can be run immediately in students' machines. Ninety percent of students in the course *MT800* had tried the online practical activities and all agreed that the system makes writing and compiling Java programs much easier. It was encouraging to see 100% submission of Java programming assignments, which indicates all students were confident in handling Java programming assessment. In the past, the normal submission rate of Java programming assignments was around 80%.

6 Conclusions and Future Work

The proposed system provides an online and interactive environment for students to learn programming at a distance. Using this online system, students are not required to install Java Integrated Development Environment (IDE) and set up suitable environment variables for program compilation. Students can get hints and extra help easily through the new system. Once the students can gain access to the Internet, they can use this online system to develop their programs and learn programming anytime and anywhere.

This paper reports the progress of the development status of the online and interactive system for teaching and learning programming. Currently, our system provides basic functions for students to do programming activities. In the future, more advanced features will be added to the system. Further investigation will be concentrated on the use of an intelligent software agent [7] that is able to monitor and collect information about students' performance. The software agent works for the

instructors and tutors to aid early detection of problems in the course delivery and give suitable advice to students in learning how to program.

Acknowledgments

The authors would like to thank the OUHK and Education and Manpower Bureau (EMB) for the Earmarked Research Grant (Ref: 9004/03H) that allowed that the system to be developed.

References

1. Thomas, P. G. and Paine, C. B. (2000). "How student learn to program: Observation of practical work based on tasks completed", Research Report 2000/03, Department of Computing, The Open University, United Kingdom.
2. Syed, M. R. (2001). "Diminishing the Distance in Distance Education", IEEE Multimedia, July-Sept. 2001, pp. 18-20.
3. Lewis, S. F., & Watkins, M. (2001, 22nd January). "Using Java tools to teach Java, the integration of Bluej and CourseMaster for delivery over the Internet", proceedings of the 5th Java in the Computing Curriculum Conference (JICC 5), South Bank University, UK.
4. Proulx V. K. (2000). "Programming Patterns and Design Patterns in the Introductory Computer Science Course", in SIGCSE Bulletin Conference Proceedings of the Thirty First SIGSCE Symposium on Computer Science Education, Vol. 32, No. 1, pg. 80-84, ACM Press, 2000.
5. Foxley, E., Higgins, C., Tsintsifas, A. and Symeonidis, P. (1999). The Ceilidh-CourseMaster System, The University of Nottingham, UK., available online: http://www.cs.nott.ac.uk/~cmp/more_info/html/CMIntro.htm
6. Nghi Truong, Peter Bancroft & Paul Roe (2003). "ELP – A WEB ENVIRONMENT FOR LEARNING TO PROGRAM", ACM International Conference Proceeding Series archive, Proceedings of the twenty-sixth Australasian computer science conference on Conference in research and practice in information technology - Volume 16 Pages: 255 – 264.
7. Caglayan, A. and Harrison C. (1997). Agent Sourcebook: A Complete Guide to Desktop, Internet, and Intranet Agent. John Wiley & Sons.