A “Development Web Environment” for Learning Programming Languages

Juan Ramón Pérez Pérez, M^a del Puerto Paule Ruiz, and
Martín González Rodríguez

HCI Research Group – LabTOO – Department of Computer Science. University of Oviedo
C/ Calvo Sotelo s/n Facultad de Ciencias. 33007 Oviedo. Spain
{ jrpp, paule, martin}@pinon.ccu.uniovi.es

Abstract. This paper proposes an environment for the teaching of program languages in a web environment. We consider that the student has to do exercises when he is learning a programming language; these tasks cause extra difficulties in conventional web environments. To remove these difficulties we propose an application that we have called DWE (Development Web Environment); the core is a language compiler integrated in the web application. Along with this compiler there are other modules to help the student while he’s developing the programs; to obtain this we try to facilitate the communication teacher – student and student – student and we try to use the experience from all users in a group.

1 Introduction

The learning of paradigm of programming and its application in a specific language is not a trivial task. There are many authors that approach the cognitive difficulties that the learning of new paradigm or the change from other paradigms presents [4].

The assimilation and setting in practice of concepts that involves a paradigm of programming implies an important effort to achieve its assimilation by students. On the other hand, for a correct understanding it is essential to combine the work of theoretical concepts with the practical tasks utilizing a programming language. It helps the student to strengthen those things learned in theory and to improve his level of knowledge. If we notice the taxonomy of Bloom's knowledge, is when the student is practicing when he uses more objectives of application and analysis and where he can discover errors of understanding.

All this implies a constant interaction between student – teacher that will provide the student a suitable knowledge of this programming language and of the tool of compilation, and that will permit the student to apply techniques to avoid errors and to solve them if they happen. In a classic learning environment where the teacher is present, this interaction is produced automatically. It is even necessary to take into account the factor of communication student – student. This model of programming learning is hardly transferable to the classical virtual learning environments. These environments are not enough when the student is doing practices of programming languages, because expressing the difficulties that he has with a fragment of code turns out to be difficult for the student. This make difficult the solution of problems.

At present, there are tools that allow the automatic correction [1,2] and proposals [3] are being developed so that these tools may make the student receive information about the type of errors that he has committed and the concepts he has to review. But there aren't tools that allow to advise the student while is accomplishing the practices of programming and before the student obtains the final result.

2 Overview of DWE and Conclusions

DWE allows the users to write, compile and correct errors of their programs written on java. The interface of compilation allows the user to go to the principal functions of the manager of files. It has an area of edition of the source files. When the user compiles one file, the system captures errors that are produced and it gives the user information about the errors and he keeps the information in a database (Fig. 1).

Fig. 1. Modules that constitute DWE

We have designed and developed an environment for the teaching of programming languages that provided several new things on other now existent environments: The compiler integrated in a web environment, avoids the installation and configuration by the user of environment and permits the centralization of maintenance. Help contribution for the user while he's writing the code, not only when errors happen. The user can make good use of the experience of others in the solution of happening errors. The teacher can do a students' tracking individually looking up the errors that the student has committed.

References

1. Dawson-Howe, K.M. Automatic submission of programming assignments. SIGCSE Bulletin 27, 4 (1995) 51–53.
2. Huizinga, Dorota M. Identifying Topics for Instructional Improvement Through On-line Tracking of Programming Assignments. SIGCSE Bulletin 33, 3 (Sep. 2001) 129–132.
3. Marco Galindo, Ma Jesús; Prieto Blázquez, Joseph Necesidades específicas para la docencias de programación en un entorno virtual. Actas JENUI 2002, 5–12.
4. Whitelaw, M. and Weckert, J. The Humanness of Object-Oriented Programming. I Cognitive Technology Conference. Hong Kong (1995).