Virtual Learning Environment to Support Object Oriented Programming Learning

Adriana Maritza Angarita Cala
Universidad Autónoma de Bucaramanga, Bucaramanga, Santander, Colombia aangarit@gmail.com

César Darío Guerrero Santander
Universidad Autónoma de Bucaramanga, Bucaramanga, Santander, Colombia cguerrer@unab.edu.co

ABSTRACT

In this paper, a Virtual Learning Environment (VLE) is used to overcome difficulties identified from the student and professors perspective when participating in an Object Oriented Programming (OOP) class. A Blended Learning strategy is used for the students to self-evaluate their concepts and skills gained in class, and from there, perform on-line activities to improve or solve theoretical or practical gaps in a particular OOP subject. The VLE is implemented in DOKEOS and evaluated by first-year engineering students in *Universidad Autónoma de Bucaramanga*. It is shown that by implementing additional learning strategies such as student-centered learning supported by technology, an effective complement is achieved when teaching OOP.

Keywords: E-Learning, Object oriented programming, Dokeos.

RESUMEN

En este artículo, un Ambiente Virtual de Aprendizaje (AVA) es usado para superar dificultades identificadas desde la perspectiva de estudiantes y profesores, cuando participan en una clase de Programación Orientada a Objetos (POO). Una estrategia de aprendizaje mixta es usada por los estudiantes para auto-evaluar los conceptos y habilidades adquiridas en clase, y desde allí, realizar actividades en línea para mejorar o resolver vacíos teóricos o prácticos en un tema concreto de la Programación Orientada a Objetos. El AVA es implementado en DOKEOS y evaluado por estudiantes de primer año de ingeniería de la Universidad Autónoma de Bucaramanga. Se muestra que mediante la implementación de estrategias adicionales de aprendizaje tales como las de educación soportada en tecnología y centrada en el alumno, se logra un efectivo complemento para el aprendizaje de la POO.

Palabras claves: E-learning, Programación orientada a objetos, Dokeos

INTRODUCTION

In recent years, there has been a significant worldwide reduction in the student enrollment to engineering programs. There are several well studied reasons to explain that trend: from low levels in mathematics, the reduction of investment in science and mathematics by some governments, to the perception of young people who think that working in this field will be exhausted. This trend is contrary to what employers will offer in this field in the immediate future: by 2020 there will be 20 million jobs (Jon S, 2009).

Mathematics is not the only reason why students do not enter or leave engineering programs. The difficulty in learning programming, as an essential element in the formation of an engineer, furthermore, it aggravates the problem. Introductory courses in computer programming require pedagogical and technological strategies to attract and not reject students while helping them to get foundations of good programming; "The correct programming does not fall from the sky" (Jadud Matthew, 2005). Object-oriented programming is one of the most influential programming paradigms in recent years (Kölling Michael, 1999) and demands learning strategies to help students to clearly understand the concept and the implementation in computer programs.

This paper proposes an e-learning strategy to help students who are taking an introductory course in computer programming on the subject of OOP. The strategy is developed as a complement of the physical learning environments and for that reason can be considered as a blended learning environment strategy.

STATE OF THE ART

Computer programming education is taking advantage of e-learning implementations from different authors and methodologies. Some of the main approaches in the field are the following:

Misoko (Heo Misoko, 2003) intends to implement Web-based teaching mainly by the personalization of learning as a means to advance the study of programming using the method of "watching others." For this purpose tools were created where students create a code and get feedback from the teacher using the same tool. This allows students to produce code, identify errors and receive guidance and support from their teachers. Misoko aims an improvement of the student learning. This study shows a need for computer programming e-learning support.

In Queensland University (Truong, Bancroft and Roe, 2003) there is a work referred to the Environment for Programming Learning (EPL), which provides a web-based interactive environment to teach computer programming. The University has made it easier to work collaboratively with benefits for both teachers and students by encouraging feedback, practice and allowing students to progress at their own pace. EPL allows easy integration of notes, tutorials and practical exercises, with the possibility of coaching at anytime and anywhere. This study shows the strength of the web-based education in solving various problems that often occur when teaching programming in Java in the classroom.

Taiwan universities have joined efforts to formalize a study presented by Yuin Wu (Hwang Wu-Yuin et al, 2008) where a Web-based solution called "DRW" offers five programming activities with multiple difficulty levels based on the Bloom's taxonomy. DRW provides online code, execution and annotation tools to conduct training and assignments via Web-based programming. It addresses not only a tool but a complete methodology through activities designed for the Web.

In the Department of Computer Science at the University of Warwick (Sitthiworachart and Joy, 2003) a study shows a collaborative e-learning and co-evaluation strategy performed by peers (students). It has been found that constructive criticism against the work of others, encourages students to make best efforts to correct your mistakes by seeing them in other jobs. It also presents some features to improve, as the possibility of affecting students subjectivism in assessing the work of others.

According to Bruce (Bruce Christine et al, 2004), students can learn to program through 5 different ways: Following, where students learn by experience; Building Code, where the learning of programming is through coding; Understanding and integrating, where the act of programming is experienced through the understanding and integration of concepts; Problem Solving, which is experienced through learning what it takes to solve a problem; and Participating, which is learning by experience when being a programmer in the field.

When looking at Object Oriented Programming (OOP), there are several documented difficulties and learning strategies such as developing a strategy to decompose a problem, understand the mechanics of programming, passing parameters, return values of methods and variable declarations, abstract programming techniques used in different situations and finally understand and use concepts of object-oriented programming such as inheritance, abstraction, specific classes and interfaces (Keefe Karen et al, 2006). Problems in teaching OOP are due to bad choice of languages and wrong teaching aids (Kölling Michael, 1999). A suitable environment must have seven key areas: ease of use, integrated tools, support objects, support for reuse of code, learning support, support groups, availability (Kölling Michael, 1999).

There are different OOP strategies and methodologies that have been implemented over the years. One of these is called "Object-first" which is to introduce students to the concepts, classes and instances before procedural elements (Wee fang et al, 2005). Additionally, a simplistic view of a holistic development front leads to include ideas and concepts gradually instead of having an integrated perspective from the outset (Börstler and Sharp, 2003).

It is important to note that there are conditions for effective use of new methodologies. Imparters need to prepare in an advanced level of ICT use and the design of teaching skills and training projects. It is also desired a good learner motivation, dominance by the teacher of what to teach, teaching tools and proximity to the reality as a means of connection between other experiential learning (Vargas and Gamboa Sarmiento, 2008).

A very interesting study developed in Colombia to analyze and design a First Course in Computer Programming is the CUPI2 project (Villalobos Jorge et al, 2005) (Villalobos Jorge, 2007). This project consisted of an initial survey of all experiences in the University in programming courses. Findings address the strategies implemented so far and the study of the possible reasons to explain why some groups may be more successful than in others.

PROBLEM IDENTIFICATION

In order to identify the problem of teaching and learning OOP in the context of a Colombian university, an on'line survey was conducted to gather anonymous feedback from engineering students who have completed their first semester and therefore their first contact with OOP. The survey was implemented in *LimeSurvey* an applied to students from four Colombian universities: *Universidad Autónoma de Bucaramanga, Universidad de Santander, Universidad Pontificia Bolivar and Universidad Industrial de Santander.* The survey was mainly focused on the following topics:

- Prior experience in programming courses
- Work or personal programming experiences
- Degree of complexity of major programming issues
- Importance of the strategies employed when learning programming
- Evaluation of topics in programming fundamentals.

Tables 1 and 2 show the main results of the survey.

Table 1: Results of one question of the survey

Topic	Very Low	Low	Medium	High	Very High
Indicate the degree of complexity that you did each of the following items of course "Fundamentals of Programming.					
Algorithm Concept	28.57%	21.43%	30.95%	7.14%	11.90%
Build algorithms or flow charts	21.43%	16.67%	33.33%	14.29%	14.29%
Relational operators, arithmetic and logical	14.29%	33.33%	35.71%	7.14%	9.52%
Selection Structure Single, Double and Nesting	11.90%	21.43%	26.19%	23.81%	16.67%
Repeat structures	16.67%	16.67%	35.71%	11.90%	19.05%
Numbering systems and conversion	7.14%	23.81%	30.95%	14.29%	23.81%
Introduction to Java	9.52%	11.90%	35.71%	14.29%	28.57%
Builders and Definition	11.90%	11.90%	42.86%	4.76%	28.57%
Concept and application of methods	7.14%	11.90%	42.86%	9.52%	28.57%
Basic operations on arrays	9.52%	9.52%	35.71%	16.67%	28.57%
Object Oriented	9.52%	11.90%	26.19%	19.05%	33.33%
Arrange step methods	7.14%	11.90%	28.57%	16.67%	35.71%

The survey was open for about 20 days to the students and received the answer of 50 people, those who filled out completely, offering results that allowed a glimpse more clearly the points of greatest need.

Table 2: Results of another question of the survey

Topic	Very Low	Low	Medium	High	Very High
Indicate the degree of importance of the following strategies to improve the teaching of computer programming					
Exercises on board	4.76%	4.76%	30.95%	16.67%	42.86%
Practice Exams	0.00%	9.52%	26.19%	23.81%	40.48%
Online quizzes (online)	9.52%	16.67%	30.95%	21.43%	21.43%
Online Tutorials (online)	0.00%	28.57%	9.52%	30.95%	30.95%
Programming in teams of 2 students	14.29%	19.05%	16.67%	21.43%	28.57%
Tutoring with a student of higher semesters	9.52%	9.52%	26.19%	26.19%	28.57%
Teacher's office hours	2.38%	4.76%	9.52%	35.71%	47.62%
Alice	48.78%	12.20%	14.63%	12.20%	12.20%
Programming Videos	4.76%	26.19%	26.19%	14.29%	28.57%
Support through monitors	2.38%	9.52%	23.81%	26.19%	38.10%
Practical work with classroom projects throughout the course	0.00%	4.76%	19.05%	14.29%	61.90%
Analysis and study of software already made	0.00%	2.38%	23.81%	21.43%	52.38%

The results allow us to confirm the great acceptation of TIC and the participation of any Technology in education, which means a very important approach to obtain positive results in the objectives of this paper.

Teachers from the universities mentioned above and others located outside the area where interviewed to share their experience when teaching OOP. Seven of them explained the major concerns in to two specific subjects: the aspects that make teaching programming a very complex task and the teaching strategies used by them.

From students' surveys and teachers' interviews, several elements were collected as a diagnostic and as a starting point for the development of the e-learning strategy. Some of them are the following:

1. Need for alternative learning strategies
2. Need of project-based learning
3. List of subjects to include in the strategy since they were the most difficult for the students.
4. Las competencias iniciales del estudiante tanto en lógica, pensamiento algorítmico, análisis y abstracción, así como en comprensión lectora y matemáticas, son de gran importancia para su aprendizaje.
5. Es substancial hacer énfasis en la metodología para la resolución de problemas y el diseño del algoritmo dentro de los cursos de programación.

VIRTUAL OOP LEARNING ENVIRONMENT

One of the most important considerations in the production of a teaching tool is the clear identification of the end users. The VLE developed in this paper is expected to be used by students who are digital natives and find it very easy to use technological tools. They are Engineers students with a very high chance to know the use of Web 2.0 tools. The design of the tool considers the following assumptions about the students:

- Establishes an autonomous and independent way and time to study the various technological tools that support will be provided within the solution.
- Understands concepts on Object Oriented Programming.
- Develop a project application where he is author and builder through the application of concepts learned.
- listening comprehension and English language
- Uses and uses the tools of Web 2.0
- Works and leverages collaborative work with peers

To help achieve the responsibilities mentioned, the proposal provides the following instructional strategies:

1. Segment on the main themes of Object Oriented Programming for better assimilation of the content to be viewed sequentially and orderly.
2. Use a free online platform that will allow students to review as many times as needed throughout the theme you want, with a communicative approach to collaborative work groups.
3. Perform an initial test to enable students to observe and get a qualitative view of itself and identify the issues that will strengthen as well as a means of validation against the knowledge of input compared to output knowledge.
4. Provide multimedia materials to cover the topics making and different forms of learning and the student repeats necessary.
5. Propose practical exercises mainly focused on the teaching needed to make a simple relationship between reality and layout of this
6. Learning divide in 5 times, each consisting of a learning object containing material, learning activities and two tests to validate learning, one will be presented at the beginning to have an assessment and to make a comparison that leads us to corroborate effectiveness of our AVA and continue improving and a final test that will let you check your learning effectively compared with the initial test

Given these assumptions described above, it is proposed to start with a segmentation approach themes in 5 major topics that will more effectively track the student's learning process, offering a bit more control by the teacher

address strategies to implement to finally work focused on learning objects for each of the issues, drawing on the reuse of course materials and other proposals later. These issues would be handled as follows:

- Abstraction: Software Engineering Analysis and Design Phase.
- Implementation: Declaration of attributes, methods, hiding information, access to builders, packet handling and encapsulation.
- Class Design: Inheritance, over-writing methods, access control methods, overloading methods and data type conversion.
- Advanced: class attributes, class methods, variables final static statements, abstract classes and interfaces.
- Java: Development in the language (code).

The proposal has its main justification in collaborative work, especially cooperative, based on autonomous learning and reflective educates the student in their learning and contextualized through a classroom project. The latter has its foundation in the response of the surveys where students give high marks to the use of development projects within the classroom as an effective way of learning.

Subsequently choose an LMS that has the characteristics consistent with our methodology and where we can carry out your proposal designed. To this end we chose DOKEOS platform on which we proceeded to the implementation of Virtual Learning Environment, which was distributed in 5 main themes identified above. We propose a learning sequence or schedule a learning object, the learner will continue and will be controlled via the accompanying face and control of reports submitted by the tool. This itinerary will have different times that have some prerequisites that will control each step and ensure a path with clear goals and objectives.

In compliance with the qualitative goal starts the first phase of the process with the Abstraction theme through the presentation of a test of 5 who self-closed questions in the tool and may submit as many times as you like, giving a clear and accurate feedback the error and the reasons why it is wrong that response. Such work will allow students learn from their mistakes to find itself the solution to the test presented.

Following the test we propose a learning activity consisting in the implementation of the first phase of work concerning the issue of abstraction, where the student through an approach will place the data context to propose a solution developed.

To carry out this learning activity the student will have the collaborative tools that DOKEOS available to students, giving way to the plurality of voices and polyphony of the symphony. This use of the tools must be coordinated in two ways, one of them is the proposed use of didactic and pedagogic each case, these tools are forums, Wikis, Blogs and chat.

Once you develop all the exercises in the subject, the student presents a similar test to the initial 5 closed questions where you can see a final diagnosis that will demonstrate the knowledge gained through this proposal and compare it to provide initial validation statistics compared to the effectiveness of the proposal.

Five moments arise establishing consistent with the five sub-themes which has divided the main theme to be established in line with the approach of the curriculum and the moments in which students direct the issues proposed in this tool. Each one of them with learning pathways and duration according to class provides attendance and progress in subject matter.

Figura 1 A LVE Lesson page

Figure 2 A LVE Wiki

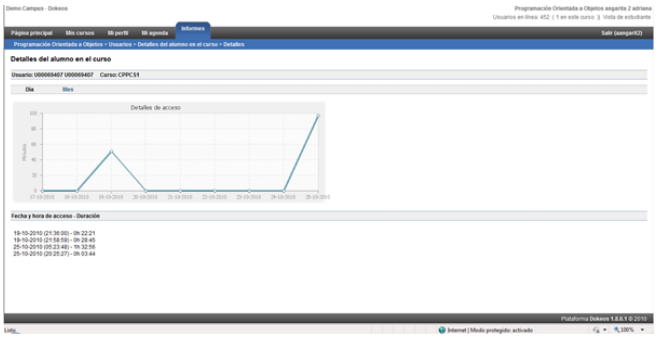

Figure 3 LVE Alumn Control Access

STUDENTS PERSEPTION OF THE DESIGNED VLE

Finally the pilot test was conducted with a group of 30 engineering students the fundamentals of the subject UNAB Programming first half, which according to their level could develop the theme of abstraction throughout the process of this object learning.

The group presented the initial test has mainly focused on the conceptual and score low in most failed. Then proceed to perform the learning activity and send it through the tool tasks, this activity only made 20% (6 students) throughout the group.

It continued with the presentation of a video where an expert discipline, which in turn is a teacher, gave an introduction to OOP and usefulness of the abstraction, and an explanation and clarification of how it could address the proposed exercise for a total understanding and ownership of the item.

Thereafter, in the face when was the last test or final test where students corroborated their learning, unlike the initial test, this test was more focused on the application of knowledge in the conceptual part. This verification throw some low scores indicating that they have greater clarity on the concept that the implementation of these concepts.

Face session was completed with the signing of the survey and concept values the opinion of the students in front of the tool and passing through it, and got a very good rating of the majority even though in many cases can be observed virtual education reserve to itself by the students.

This final survey instrument allows us to see any improvement to launch another test again taking into account the findings and the conclusions drawn from this first stage.

We can say that our exploration phase of the existing developments, from the perspective that teachers and students of the subject of planning, implementation and testing, some points are concluded:

Students expect and need more practical education, focusing on solving problems based on independent learning, but guided by both their teachers and monitors higher semesters will surely know the problems and difficulties that students are going and easy way overcome. As shown by the results to the question of evaluation of the tool:

- *Do you think that virtual education is a useful alternative to reinforce what they learned in class?*

	Grand Total	Percentage	Graph
Yes	22	100 %	
No		0 %	
Total	**22**		

Pilot test was conducted with students at the basic cycle of first-year Engineering Fundamentals of Programming, according to the results and the development of the test, we conclude that the proposed use is necessary for students to advance knowledge address the concepts in Object Oriented Programming so that the top (Abstract) be a reinforcement, ie according to the study group aims to start with students in second semester.

Since the thematic approach spans the full learning issues OOP is proposed for future work to implement the proposed environment through several semesters so that it can be addressed fully and obtain a validation of all tools, besides the acceptance of students for these initiatives reflected in the final survey:

- *Do you think the development of a project (developed in the first part in this stage: Abstraction), it facilitates understanding of each stage of full development?*

	Grand Total	Percentage	Graph
Yes	19	90.48 %	
No	2	9.52 %	
Total	**21**		

Collaborative tools like wikis, chat, blog and forums designed to provide lasting support and collaborative construction must be accompanied by a strong motivation from the classroom by the teacher for the use and exploitation of these, proposing discussions and promoting their value in the academic period, through the investigation of external material and answers to questions that students can observe. This conclusion is supported by the utility that these students find in them a reflection of its response to the survey:

- *Do you think that collaborative or cooperative work proposed by the tools forum, blog and wiki contributors to their learning?*

	Grand Total	Percentage	Graph
Yes	20	90.91 %	
No	2	9.09 %	
Total			

It attempts to update and modify the questions presented in the test periodically, so that students do not pass this information through the following groups and so these tests may not be valid. It also proposes a constant search for material that can be updated documentary proposal, taking into account the characteristics that this type of material should be about teaching and multimedia, as reflecting the views of students:

- *Does the environment easy to use tool (the platform)?*

	Grand Total	Percentage	Graph
Yes	22	100 %	
No		0 %	
Total	**22**		

A possible factor to consider is the difficulty that occurs in the interdisciplinary students, and students from areas other systems have high motivation and denial to the proposal, given its low and interest in this subject.

CONCLUSIONS AND FUTURE WORK

Definitely teaching corresponds in two-way and especially it's very important the knowledge and acceptance of the characteristics of those who is directed, therefore and aware that new generations students feel acceptance and connection with technology, education and who provided should be part of this environment.

The teacher with simple, friendly, adequate and teaching tools, to publish and monitoring of their students guarantees a high success probability in its work.

The allocation of responsibilities with goals, framed in a independence and individuality process and supported by media and related tools, allows to change the student psychology with which perceives the learning process. Get tunning with what he considers his environment and taking advantage for this purpose new features of his personality, allows to get results beyond what you expected.

Approach different learning strategies from the learning concepts, guarantees the coverage of a wider possibilities of the receiver range, as well as the possibility of review study material when he autonomous and not depended learning, it considers, it complements the features that make this proposal, an interesting means of achieving closer to students and ensure their understanding of the subjects related to programming Object-oriented.

REFERENCES

Börstler Jürgen and Sharp Helen (2003). Learning and Teaching Objetc Technology. Special Issue, Journal of Computer Science Education, 13 (4).

Keefe Karen et al (2006). Adopting XP Practices for Teaching Object Oriented Programming. Proceedings of the Eighth Australasian Computing Education Conference, Hobart, Australia.

Kölling Michael (1999). The Problem of teaching object-oriented programming Part I: Languages. Journal of Object-Oriented Programming, 11 (8), pp. 8-15.

Kölling Michael (1999). The Problem of teaching object-oriented programming Part II: Environments, Journal of Object-Oriented Programming, 11(9), pp.6-12.

Sitthiworachart Jirarat y Joy Mike (2003). Web-based Peer Assessment in Learning Computer Programming. ICALT: 180-184.

Truong Nghi, Bancroft Peter y Roe Paul (2003) A Web Based Environment for Learning to Program. 26 Australasian Computer Science Conference, Vol 25, No 1.

Vargas Germán, Gamboa Sarmiento Sonia Cristina (2008) Didáctica en la condición postmoderna. De las competencias a la cooperación. http://dialnet.unirioja.es/servlet/dcfichero_articulo?codigo=2754106&orden=0, 13/02/2010.

Villalobos Jorge (2007) Cómo enseñar a programar: un enfoque efectivo. http://cupi2.uniandes.edu.co/proyecto/presentaciones/Cupi2-2007-04-18-CCC07.PDF , 18/01/2010.

Villalobos Jorge, Casallas Ruby, Marcos Katalina (2005) El reto de diseñar un Curso de Programación de Computadores. XIII Congreso Iberoamericano de Educación Superior en Computación, Pontificia Universidad Javeriana , p.48 - 55 , v. <, fasc.

Villalobos Jorge et al (2005) Looking for a new approach to teach/learn a first computer-programming course. International Conference on Engineering and Computer Education (ICECE´05).

Bruce Christine, Buckingham Lawrence, Hynd John, McMahon Camille, Roggenkamp Mike, Stoodley Ian (2004) Ways of Experiencing the Act of Learning to Program: A Phenomenographic Study of Introductory. Transforming IT education: Promoting a culture of excellence. Informing Science Press, United States, pp. 301-325.

AUTHORIZATION AND DISCLAIMER

Authors authorize LACCEI to publish the papers in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.