INTED 2010

International Technology, Education and Development Conference

Valencia (Spain)
8th-10th of March, 2010

CONFERENCE PROCEEDINGS

www.inted2010.org

INTED 2010

International Technology, Education and Development Conference

Valencia (Spain)
8th-10th of March, 2010

CONFERENCE PROCEEDINGS

www.inted2010.org

Published by
International Association of Technology, Education and Development (IATED)
www.iated.org

INTED2010 Proceedings CD
Edited by
L. Gómez Chova, D. Martí Belenguer, I. Candel Torres
International Association of Technology, Education and Development
IATED, Valencia, Spain

ISBN: 978-84-613-5538-9
Depósito Legal: V-845-2010

Book cover designed by
J.L. Bernat Tomás

WELCOME INTRODUCTION

Dear INTED2010 participants,

It is a great honour to welcome you to this forth annual edition of INTED2010 (International Technology, Education and Development Conference).

The main aim of this conference is to provide an international forum, counting with experts in different fields and disciplines from more than 60 countries who will present and discuss the latest innovations in education, technology and development.

With the presence of more than 400 attendants, INTED2010 also aims to be a social platform and a great opportunity for networking, which makes this experience more interesting for its international and multicultural atmosphere.

Valencia, venue of this conference, will provide you with the opportunity to discover a city with impressive architecture, interesting museums, lovely beaches and a varied cultural offer that will make your stay unforgettable.

Thank you very much for coming to INTED2010 and for contributing to the improvement of Education with your projects and experiences. We wish you a fruitful conference!

INTED Organising Committee

SCIENTIFIC COMMITTEE AND ADVISORY BOARD

Agustín López	SPAIN	Luciana Oliveira	PORTUGAL
Alexander Schmoelz	AUSTRIA	Luis Gómez	SPAIN
Amparo Girós	SPAIN	Lyudmila Smirnova	UNITED STATES
Andrei Achimas Cadariu	ROMANIA	Mª Jesús Suesta	SPAIN
Anna Mazzaro	UNITED STATES	Marc Seifert	GERMANY
Antonio García	SPAIN	Margarida Lucas	PORTUGAL
Ari-Matti Auvinen	FINLAND	Maria Fojk	IRELAND
Artis Ivanovs	LATVIA	Maria Porcel	SPAIN
Barbara Schroettner	AUSTRIA	Mariane Gazaille	CANADA
Becky Kwan	HONG KONG	Michael Cant	SOUTH AFRICA
Brian McKay-Epp	UNITED STATES	Miriam Schcolnik	ISRAEL
Claudia Kummer	AUSTRIA	Mohamed Elammari	LIBYAN ARAB JAMAHIRIYA
Damien Shortt	UNITED KINGDOM	Niki Frantzeskaki	NETHERLANDS
David Martí	SPAIN	Norma Barrachina	SPAIN
David Nielsen	AUSTRALIA	Norrizan Razali	MALAYSIA
David Santandreu Calonge	HONG KONG	Oge Marques	UNITED STATES
Deirdre Kelleher	IRELAND	Rachid Benlamri	CANADA
Don Cyr	CANADA	Relja Dereta	SERBIA
Elena Ors	SPAIN	Robert Heath	UNITED KINGDOM
Fran Cornelius	UNITED STATES	Roger Bateman	NEW ZEALAND
Giuseppe Fiorentino	ITALY	Sergio Pérez	SPAIN
Helen Keegan	UNITED KINGDOM	Shaibu Bala Garba	OMAN
Ignacio Ballester	SPAIN	Sharon Jumper	UNITED STATES
Ignacio Candel	SPAIN	Silvia Ferraris	ITALY
Ismael Serrano	SPAIN	Siobhan O' Sullivan	IRELAND
Javier Domenech	SPAIN	Susana Raya	SPAIN
Javier Martí	SPAIN	Sven Tuzovic	UNITED STATES
José Antonio Arrueta	SWEDEN	Theresa Fay-Hillier	UNITED STATES
Jose F. Cabeza	SPAIN	Thomas Baaken	GERMANY
Jose Luis Bernat	SPAIN	Victor Fester	NEW ZEALAND
Lasse Ziska	GREENLAND	Xavier Lefranc	FRANCE

CONFERENCE SESSIONS

ORAL SESSIONS, 8th March 2010.

e-learning & Blended Learning (1)
Educational Software and Serious Games (1)
University-Industry Collaboration
Curriculum Design in Engineering Education
Architecture & Urban Planning: Pedagogical & Didactical Innovations
e-learning & Blended Learning (2)
Educational Software and Serious Games (2)
Enhancing Learning and the Undergraduate Experience (1)
Engineering Education: Pedagogical & Didactical Innovations
Architecture & Urban Planning: International Projects & Research
Pedagogical & Didactical Innovations (1)
Computer Supported Collaborative Work. Web 2.0 and Social Networking
Enhancing Learning and the Undergraduate Experience (2)
Experiences in Engineering Education
Arts & Humanities: New Experiences and Pedagogical & Didactical Innovations
Pedagogical & Didactical Innovations (2)
Computer Supported Collaborative Work
Quality Assurance in Education
Learning Experiences in Primary and Secondary School
Bus. Adm. & Mgmt.: Experiences in Education and Pedagogical & Didactical Innovations

POSTER SESSIONS, 8th March 2010.

Poster Session1. Technological Issues & Computer Supported Collaborative Work
Poster Session2. Educational Software and Serious Games & Pedagogical & Didactical Innovations

ORAL SESSIONS, 9th March 2010.

Technology-Enhanced Learning (1)
Curriculum Design and Innovation
Foreign Languages: Experiences in Education
Experiences in Education. New projects and innovations (1)
Teacher and Pre-service Teacher Education Experiences
Experiences in Education. New projects and innovations (2)
Technology-Enhanced Learning (2)
International Projects
Curriculum Design and Innovation. Strategies, Principles and Challenges
Foreign Languages: Pedagogical & Didactical Innovations
Experiences in Education
Technological Issues in Education
Barriers to Learning & Diversity Issues in Education
New Experiences for Curriculum Design
Health Sciences: Experiences and Pedagogical & Didactical Innovations
New Trends in the Higher Education Area
Research on Technology in Education
General Issues. Education and Globalization
Research in Education
General Issues. Education & Development

POSTER SESSIONS, 9th March 2010.

Poster Session1. Experiences in Education and Research & International Projects
Poster Session2. Curriculum Design, University-Industry Collaboration, Quality Assurance & Higher Education Area

VIRTUAL SESSIONS

Computer Supported Collaborative Work
Curriculum Design and Innovation
E-content Management and Development
Educational Software and Serious Games
Experiences in Education
Experiences in Education. Competence Evaluation
Experiences in Education. Enhancing learning and the undergraduate experience
Experiences in Education. Learning Experiences in Primary and Secondary School
Experiences in Education. New projects and innovations
General Issues. Barriers to Learning
General Issues. Education, Globalization and Developmnet
General Issues. Organizational, legal and financial issues
International Projects
New Trends in the Higher Education Area. ETCS experiences and Joint degrees programmes
New Trends in the Higher Education Area. New challenges for the Higher Education Area
Pedagogical & Didactical Innovations. Collaborative and Problem-based Learning
Pedagogical & Didactical Innovations. Evaluation and Assessment of Student Learning
Pedagogical & Didactical Innovations. Learning and Teaching Methodologies
Quality assurance in Education
Research in Education. Academic Research Projects
Research in Education. Experiences in Research in Education
Research in Education. Research on Technology in Education
Technological Issues in Education. E-learning and Blended Learning
Technological Issues in Education. Technology-Enhanced Learning
University-Industry Collaboration
Virtual Universities. Distance education

ABOUT INTED2010 Proceedings CD

HTML Interface: Navigating with the Web browser

This CD includes all presented papers at INTED 2010 conference. It has been formatted similarly to the conference Web site in order to keep a familiar environment and to provide access to the papers trough your default Web browser (open the file named "INTED2010.html").

An Author Index, a Session Index, and the Technical Program are included in HTML format on this disk to aid you in finding particular conference papers. Using these HTML files as a starting point, you can access other useful information relating to the conference.

The links in the Session List jump to the corresponding location in the Technical Program. The links in the Technical Program and the Author Index open the individual paper in a new window. These links are located on the titles of the papers and the Technical Program or Author Index window remains open.

Full Text Search: Searching INTED2010 index file of cataloged PDFs

If you have Adobe Acrobat Reader version 6 or later (www.adobe.com), you can perform a full-text search for terms found in INTED2010 proceedings papers.

Important: To search the PDF index, you must open Acrobat as a stand-alone application, not within your web browser, i.e. you should open directly the file "INTED2010.pdf" in the CD with your Adobe Acrobat or Acrobat Reader application.

This PDF file is attached to an Adobe PDF index that allows text search in all PDF papers by using the Acrobat search tool (not the same as the find tool). The full-text index is an alphabetized list of all the words used in the collection of conference papers. Searching an index is much faster than searching all the text in the documents.

To search the INTED Proceedings index:

1. Open the Search PDF pane through the menu "Edit > Search" or click in the PDF bookmark titled "SEARCH INTED2010 PAPERS CONTENT".
2. The "INTED_index.pdx" should be the currently selected index in the Search window (if the index is not listed, click Add, locate the index file .pdx on the CD, and then click Open).
3. Type the search text, click Search button, and then proceed with your query.

For Acrobat 9:

1. In the “Edit” menu, choose “Search”. You may receive a message from Acrobat asking if it is safe to load the Catalog Index. Click “Load”.
2. A new window will appear with search options. Enter your search terms and proceed with your search as usual.

For Acrobat 8:

1. Open the Search window, type the words you want to find, and then click Use Advanced Search Options (near the bottom of the window).
2. For Look In, choose Select Index.
3. In the Index Selection dialog box, select an index, if the one you want to search is available, or click Add and then locate and select the index to be searched, and click Open. Repeat as needed until all the indexes you want to search are selected.
4. Click OK to close the Index Selection dialog box, and then choose Currently Selected Indexes on the Look In pop-up menu.
5. Proceed with your search as usual, selecting other options you want to apply, and click Search.

For Acrobat 7 and earlier:

1. In the “Edit” menu, choose “Full Text Search”.
2. A new window will appear with search options. Enter your search terms and proceed with your search as usual.

EXPERIENCES IN THE STUDENTS' ASSESMENT IN A COMPUTER SCIENCE APPLIED TO ENGINEERING SUBJECT

D. Úbeda, A. Gil, L. Payá, Ó. Reinoso, L. Fernández

Departamento de Ingeniería de Sistemas Industriales, Miguel Hernández University
Elche / Spain
{*ubeda, arturo.gil, lpaya, o.reinoso, l.fernandez*}*@umh.es*

Abstract

In this paper, the "Engineering Applied Computer Science subject" case is introduced. It is a first course subject of the Industrial Engineering at Miguel Hernandez University. With the imminent arrival of Bologna plan and the ECTS, we have to rate the student performance out of the classroom, and it is more convenient monitoring the individualistic work of each, in this non-presencial part of the credits. In our case, individualistic work of each student is translated in a set of C/C++ practical sessions that must be sent to the teachers in order to be evaluated. In previous courses, there was a big problem with the old evaluation system: students used to send plagiarized source code from other companions, and the practices correction was a hard work because of the high number of matriculations at this subject. In order to solve this problem, we have developed some years ago a Moodle pseudo-module that it is able to detect plagiarized source code in the C/C++ practices after they have been sent to the platform. This module returns to the teacher a complete statistic about the number of detected copies, similar parts of the code, increase or decrease of the copies according to old practices, etc. The system is able to save a source code historical of the last years practices which are compared with the new ones, in order to find some similarities.

Keywords - Plagiarism, Innovation, Moss, Moodle, C++.

1 INTRODUCTION

This paper will not discuss any of the existing plagiarism detection algorithms. The aim is to find a optimal solution to the high rate of plagiarism detected in the first course subject, Engineering Applied Computer Science of the Industrial Engineering at Miguel Hernandez University.

The rate of enrollment in this subject is about 140 students per course, average over the last ten promotions; this is a large number of enrolled students because it is a first-year course signature, and the registration is obligatory for new students. We should add repeater students to this number, which are often numerous, mainly because the rate of abandonment of the subject, perhaps because they have not received similar subjects at high school.

Because of the large number of students enrolled, we decided to do a starting poll the first day of class on the reason about the election of Industrial Engineering degree for preparation as a professional dedication. The discovery was very disappointed, since between 70-80% of the students, had chosen the Industrial Engineering degree as a second option in their choice of matriculation, or had chosen at random, or even by the name of the degree without even knowing the contents of the subjects. The remaining students, between 20 and 30%, said they had chosen this option because they know some people who is working in some professional competence derived from the degree, or they have read the content of the subjects identifying certain skills and finding that they wish to develop or learn for their professional dedication.

Given these overwhelming numbers, we realize that we will meet with a motivation or disinterest problem in some subjects by the students, and it will not be complex to think about a high level of plagiarism.

With this situation, we started to work and decided to devise a plan to try to reduce this forecast of plagiarism. At first, we decided to grant the approval of the doubt and proceeded to deliver the first two practices. Once we have corrected both practices and verified the similarity of code through JPlag [6], we decided to find and integrate a tool that disturbed the students because of the detection efficiency,

and provide teachers a tool for more objective assessment and with more precise details of the plagiarism. We wanted to integrate this tool with Moodle CMS, since it was the portal through which students had access granted to deposit the practices carried out.

2 CURRENT PROBLEMS WITH THE SOURCE CODE EVALUATION

Plagiarism in teaching practices is a subject widely covered in numerous conferences and seminars for teachers responsible of the most varied subjects, from anthropology to the degree of Engineering. It is a complicated problem to solve because on one hand hurts the pride of the teacher, because any teacher likes to have plagiarized practices in their subjects and on the other hand shows a lack of responsibility of the student who first comes to college, where they may not yet have insight the reason because he or she has to study, or the reason because he has to perform a task of a subject, regardless of attaining approval. We are speaking about the first courses of degrees, where there are the greatest problems, since it is found that the number of copies is much higher than in the completion of studies courses.

From our point of view, the solution not only comes down to a warning early in the course on the detection of plagiarism and its consequences, it should show at least once every two or three courses, since it seems that what works better coinciding with Mr Clement [1], is the fear about consequences, and thus the transmission of these consequences through repeater students and new access students. The fear about consequences seems very effective in all cases and with all the tools used.

In most cases, the student believes the teacher is not going to realize about of plagiarism, and perhaps, in subjects where the number of students is high, they are right, because without a computer as we will discuss, it is almost impossible to identify the practices copied, since it is impossible to remember all practices to verify its authenticity with the rest.

As we mentioned at the introduction, we conducted an experiment where we proposed a couple of initial practices, and we don't warned about it would be a plagiarism control. This example was made during the year 2007/08 in the subject of Applied Computer Science and we found that the number of students with fully or partially plagiarized practices (Fig. 1) in one group was higher than the number of detected plagiarism in 5 groups during the next year. This was not the only thing we discovered: as a result of students knew that we were controlling the plagiarism, the level of copies during that course was drastically reduced for the rest of practices, but unfortunately the number of pupils assisting to the subject had also declined equally. It is evident that these students would not make an effort to do the practices and most probably intended to plagiarize from their peers.

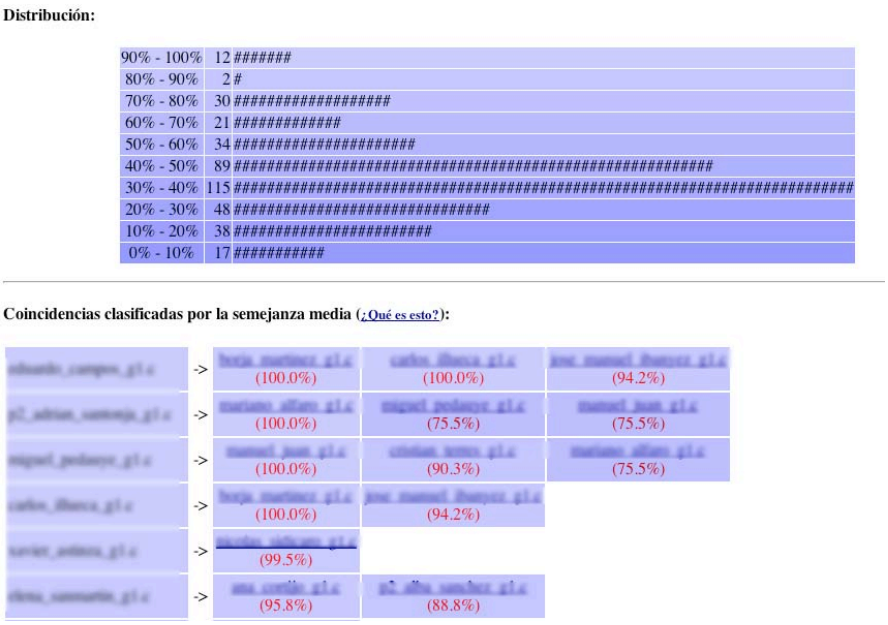

Fig. 1. Distribution of copied students from a practical group during course 2007/08

3 STATE OF THE ART ABOUT PLAGIARISM DETECTION SOFTWARE

Since it was not intended at any time generate new software from the numerous algorithms that exist today, on the detection of plagiarism, we decided to go to the free software community and look for a certain application which suits our needs. Specifically, we were looking for two fundamental characteristics: first, the highest ability to integrate with our system of collection practices, which was Moodle [7], and secondly, it was an external server which implements the algorithm and return the results.

It was necessary any dependence on a client application that would be installed in our collection practices server and that further communication were in a language fully integrated with our website, type shell scripting, C / C + + or Perl [9].

If we consider the current classification of Goel [4] as to the most relevant applications to detect plagiarism of source code, we discard all those that use the system through a non-TPC/IP application, ie application running on a server. In addition, we needed that the application uses a web interface, since otherwise, the compatibility that was intended would be impracticable or extremely difficult to follow, compared to the fast integration necessary for the academic year that he would impart. For this reason, we dismiss the use of JPlag, Sherlock [11] CodeMatch [2] and DPC [3], but first we tried them all looking for the best results regarding the characteristics required.

JPlag has impressed us most positively because of the speed response from the server, the metric derived from the results, the suspect code was highlighted and the graphical interface the web which the results were showed were agile and intuitive (Fig. 1).

This application was based on a Greedy String Tiling algorithm [5]. In fact, we decided to conduct the first tests through this platform. The problem was when several teachers wanted to use the same application or that wanted to use the application on different computers. The client application is programmed in Java, and therefore it was necessary to download the application each time you wanted to use. Although very effective, we seemed that it was not the most optimized application to use with our CMS system.

Therefore, we decided to continue searching for the application that best suits our needs, and observing the fundamental characteristics of the application that was developed at Stanford University [12], MOSS [8], we decided that, we had found it, since it provides us a practical method of sending practices via sockets, through command line, programmed with a simple Perl script it was very quick to integrate into Moodle. The collected information returned by the HTTP petition, was a Stanford University web address where they stored the results of the verification of plagiarism. The algorithm used with MOSS to compare the code is the Winnowing algorithm [13], which it is less powerful than the algorithm mentioned above, but it was sufficiently efficient for our practices.

Fig. 2. Generating a new plagiarism resource

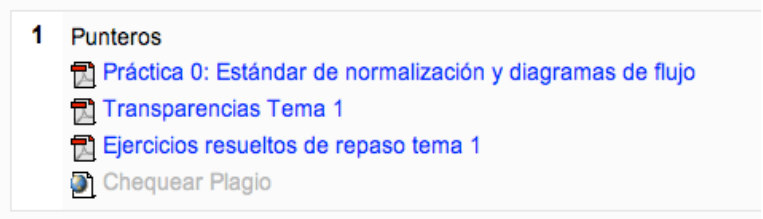

Fig. 3. Plagiarism checking application access

For the implementation was adapted to our needs, we needed one last feature that MOSS does not support: to store the results of the verification of plagiarism on our server. It was not a big problem to implement this functionality through automated whole directory downloading and its 2 levels of subdirectories, where images are stored, etc.., And include it in our application, so that Moodle can store in all the results of plagiarism detection for further consultation, without fear of Stanford University delete these temporary files.

4 MOODLE MODULE INTERFACE

The decision to use MOSS to integrate with Moodle was successful since it was relatively simple, creating a kind of pseudo-module for the CMS. We call pseudo-module because it really is not a module installed in any version of Moodle, but is a standalone application that uses the structure of files and MySQL database to create Moodle.

On one hand, we decided to use the MySQL database for Moodle for not giving students a new user and password, which in practice is not a good service for students to have different keys to enter the subjects. In this way, using the directory structure that generates Moodle for the delivery of individualized tasks, we conducted a series of scripts in PHP [10] to enable us a link to the MOSS server at the University of Stanford, and let us store the link with the results of checking for plagiarism, in a variable in our PHP script, which we would use to display it in our application.

Fig. 4. Sending practices

5 GENERATING A NEW PLAGIARISM RESOURCE

The performance of the pseudo-module is simple, as shown in Fig. 2. Initially it will be necessary to edit the subject or practice in Moodle, and add a resource link. The most important topic is to add the PHP variable for the practice concerned, to the address where our script will send the practices to MOSS server. Another important issue to accomplish the successful operation of the application will be to pass the Moodle identifier course, or 'id course'. Through this identifier, the application will know the directory from which it must extract the files, because of the number of the indicated task in the resource we've created earlier.

At this moment, we have built our transparent application access control plagiarism (Fig. 3).

To query the number of students plagiarized, we could enter with a teacher account of any of the teachers of the subject Moodle, and then click on "Check Plagiarism" access, and a pop-up window will open showing us the results of plagiarism of the actual practice verification (Fig. 4 and 5).

6 CONCLUSION

As a conclusion to the work presented in this paper, we could say that we have obtained a versatile and perfectly adaptable module to any type of Moodle CMS educational system. This project has led us to understand how varies our subject respect to the number of students with plagiarism in their practices, through the showed application that, it is not going to end with plagiarized practices, at least it will serve as a lesson to those that want to try with a obviously plagiarism, not so much by fear of the consequences of plagiarism, which most often ends in suspense, but not by the fear of ridicule to the plagiarism detection.

As a result, we confirmed what we already imagined: if some students don't copy, they would abandon the subject. This has made us to improve our teaching quality surveys respect to the number of submitted and passed students, but has made us to think about the main question, and in which we are currently working: how can we motivate students of programming subjects from first courses?

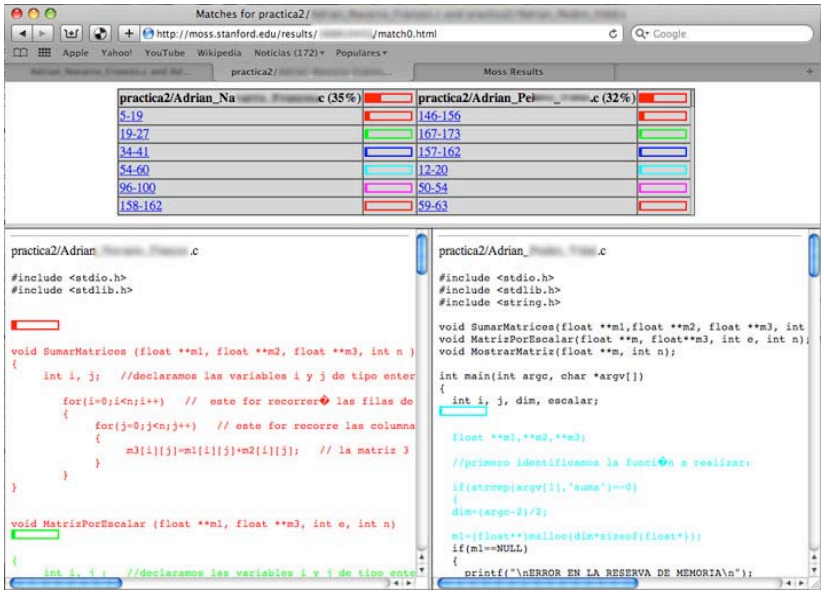

Fig. 5. Plagiarized practices detected

References

[1] Pedro J. Clemente, Alberto Gómez, Julia González, *La copia de prácticas de programación: el problema y su detección*, Dpto. de Informática, Universidad de Extremadura, JENUI'04

[2] Codematch, http://www.zeidmanconsulting.com/CodeSuite.htm

[3] Copy/Paste Dectector (CPD), http://pmd.sourceforge.net/cpd.html

[4] Sanjay Goel, Deepak Rao et. al., *Plagiarism and its Detection in Programming Languajes*, Department of Computer Science and Information Technology, JIITU May, 2008

[5] Greedy String Tiling Algorithm, Lutz Prechelt, Guido Malpohl and Michael Philippsen*, "JPlag: Finding Plagiarisms among a Set of Programs",* 2000. http://page.mi.fuberlin.de/prechelt/Biblio/jplagTR.pdf.

[6] JPlag, https://www.ipd.uni-karlsruhe.de/jplag/

[7] Moodle, http://www.moodle.org

[8] Alex Aiken. MOSS (Measure Of Software Similarity) plagiarism detection system. http://www.cs.berkeley.edu/~moss/ (as of April 2000) and personal communication, 1998. University of Berkeley

[9] Perl scripting Language, http://www.perl.org

[10] PHP Scripting Language, http://www.php.net

[11] The Sherlock Plagiarism Tool, http://www.cs.usyd.edu.au/~scilect/sherlock/

[12] Stanford University, http://www.stanford.edu

[13] Saul Schleimer, Daniel S. Wilkerson, Alex Aiken, *Winnowing: Local Algorithms for Document Fingerprinting*, SIGMOD 2003, June 9-12, 2003, San Diego, CA.

[14] J.C. Sánchez and R.L. Taylor, "La innovación educativa en la universidad española", *Revista de Educación*, Vol. 48, pp. 101-118, (2005).