An approach to identify relevant subjects for supporting the Learning Scheme creation task

Huander Tironi
Post Graduate Program in informatics
(PPGIa) – Polytechnic School
Pontifícia Universidade Católica do
Paraná - PUCPR
Curitiba, Brasil
huander.tironi@gmail.com

André Menolli
Computer Science Departament
Universidade Estadual do Norte do
Paraná - UENP
Bandeirantes, Brazil
menolli@uenp.edu.br

Sheila Reinehr, Andreia Malucelli
Post Graduate Program in Informatics
(PPGIa) - Polytechnic School
Pontifícia Universidade Católica do
Paraná – PUCPR
Curitiba, Brazil
sheila.reinehr@pucpr.br,
malu@ppgia.com.br

***Abstract*—The necessity to improve performance of the processes within organizations, gave rise to many research that apply concepts from educational area in software development companies. Many studies are related to Organizational Learning (OL), an area that helps companies to improve their processes significantly through the reuse of experiences. In recent works, some approaches propose to generate courses in organizations from content produced by employees. The main limitation of these approaches is the high dependence of an expert, who is responsible by the courses. Even a qualified expert, can be unfamiliar with the real need of the team's learning, and mapping the organizational needs requires time and effort. This work presents a mechanism for software development companies, capable of recovery searches performed by employees on the internet, in order to discover the real necessity of the team's learning. From these needs is purposed a learning schema of a unit of learning (the structure of a course), so helping the expert in the course creation task. An initial experiment was conducted and the results indicate that the use of the approach is viable and may help an expert create units of learning, assisting to improve the OL in software development teams.**

***Keywords-** Organizational Learning; Unit of Learning; IMS Learning Design; Learning Scheme; Recommendation System*

I. INTRODUCTION

Nowadays, the impact of knowledge on organizations is so relevant that it is not treated as a strategic factor in potential, available to few privileged, but as a common element essential to the company survival [1]. Knowledge is vital to corporations, especially for intensive knowledge company. The intensive knowledge projects refer to those where most work is said to be of intellectual nature and qualified employees form the bulk of workforce [2].

On the software development, the technical expertise that each employee acquires with the business practices and routines is valuable for the organization. Thus, as time goes by, the experiences and lessons learned gained make the software professionals more valued, becoming them in a source of basic knowledge to the company. However, the high value given to these employees creates an interest by other companies on these professionals. Losing an experienced employee to another company, means losing the acquired knowledge over time [3].

This situation makes organizations look for ways to store and share the knowledge generated. The field that seeks minimize those problems is the Organizational Learning (OL), which deals with the capacity or processes within an organization to maintain or improve performance based on experience [4].

Some recent researches are applying concepts from educational area, such as Learning Objects (LO) and Units of Learning (UOL), to improve OL in software development companies. A LO is defined as any independent digital or non-digital entity that may be reused in several teaching contexts [5]. Furthermore, a UOL can be seen as a general name for a course, a workshop or a lesson that can be instantiated and reused many times by different people and in different settings in an online environment [6].

Based on this context, the work of Menolli, Reinehr and Malucelli [7] proposes a semantic collaborative environment for software development companies. The environment aims to organize the content generated using social tools in learning objects and later, using a learning design defined by an expert, create units of learning, using semantic technologies. However, in this approach, the creation of courses depends directly of an expert, who defines a course structure, using a Learning Scheme (LS). LS is a structure defined on a meta-language, e.g. XML, which the tags form a structure that contains elements from a course such as a process of teaching and learning [8].

This dependency makes indispensable the presence of an expert, who can assume a high cost position. However, even the expert taking a high position; he can be unfamiliar with the real need of the team learning. To map the team needs requires time and effort and can be a barrier to set a Learning Scheme.

Therefore, having exposed these limitations it is necessary to advance, regarding learning, and present an approach that assist to identify relevant subjects and content to the team.

Hence, this paper presents an mechanism for software development companies, capable of recovery searches performed on the internet by the employees, and then, using a clustering algorithm, to group this searches, helping on the Learning Scheme creation task.

The remaining parts of the paper are organized as follows: Section II presents background information on the main

DOI reference number: 10.18293/SEKE2015-141

concepts behind the proposed mechanism, such as Learning Scheme, Clustering, Text Mining and Recommendation Systems. Section III shows some related works. Section IV introduces the proposed mechanism as well as its architecture. Section V presents an experiment and Section VI presents the final considerations about the study.

II. Background

A. Learning Design

The structure of a UOL is defined using some kind of Educational Modeling Language (EML), that are models of semantic information or aggregations, that describe, of a pedagogic point of view, a content as well as educational activities [9].

The EML are organized on units of study in order to allow its reuse and interoperability [10].

One of the main Educational Modeling Language is the IMS Learning Design (LD) [11], which supports the use of different approaches of teaching or learning, such as: behaviorists, cognitive and constructivist. The model describes “Units of Learning”, as elemental units that come learning events for learners, satisfying one or more learning objectives.

The IMS Learning Design specification is a meta-language that describes all the elements of the project of a process of teaching and learning, elaborated by the work group IMS/GLC [11]. The IMS LD describes a method comprising by a series of activities conducted by both the student and the team, in order to reach the learning objectives [9].

However, the IMS LD is much more complex than only organizing knowledge in a course form. Menolli, Reinehr and Malucelli [7], proposed an adaptation of IMS LD to become viable its use on an organizational learning environment. The main differences between the learning design proposed on Menolli, Reinehr and Malucelli [7] and the actual IMS LD, is that on the proposed environment the components related with time and execution control on a UOL were not used. As the purpose of this work is to present an advance on an approach already proposed, it is been used the same concepts.

In this approach, the UOL contains Resources and is organized as a Learning Design, which contains definitions such as Pre Requisites and Learning Objectives. The LD is also bound to an activity which contains a Description and its Structure.

This information is organized in a XML file called Learning Scheme. The Learning Schema follows the IMS LD structure, and contains information about the course, such as objective and prerequisite, beyond the activities of learning as well its hierarchy and sequence.

B. Recommendation Systems

There is an extensive class of Web Applications that involve predicting user responses to option. Such a facility is called a recommendation system [12].

There is a list of applications of recommendation systems that goes from Products to News Recommendation, but there is a few applications aimed to learning. Some [13] proposed a semantic recommendation system for e-learning domain to help the learners find subject they need to learn based on learners knowledge level, learners profile and some learners evaluation. Also is presented [14] a model to improve proactive context-aware recommendations in e-Learning systems to be applied in online e-Learning authoring tools.

There are two basic architectures for a recommendation system: Content-based systems examine properties of the items recommended; and Collaborative filtering systems, that recommend items based on similarity measures between users and/or items [15].

However, the one which fits better to the proposed mechanism is called Content-Based. The Content-Based systems focus on properties of items. Similarity of items is determined by measuring the similarity in their properties [16].

In a content-based system is necessary to build each item a profile, which is a record of collection of records representing important characteristics of that item. In simple cases, the profile consists of some characteristics of the item that are easily discovered. But, there are other classes of items where it is not immediately apparent what the values of features should be [17]. It is considered to this work one of them: words or documents collections.

In order to identify these words, we proceed with some practices of text mining called Filtering, which is a list of words to discard because they represent low-semantic words (prepositions, etc), and Stemming words to achieve a canonical concept representation (e.g. analysis, analyzing, analyser are collapsed to ANALY).

Once the documents are represented by sets of words, is necessary to measure the similarity of two or more documents, and to that there are several natural distance measures can be used, such as Jaccard similarity coefficient [12] between the sets of words, or cosine distance between the sets, treated as vectors.

III. Related Works

In recent years, organizations have begun to place more value on the experience and know-how of their employees, i.e., their knowledge [18]. Therefore, it has become a challenge to develop and implement processes that generate, store, organize, disseminate and apply the knowledge produced and used in a company in such a way that it can be systematically and reliably accessed by the organizational community [7].

In recent years, software companies have used tools and technologies to knowledge management that were not designed for this specific purpose [19]. The arising of the Web 2.0 (blogs, wikis, content sharing sites, social networks, etc.) gives access to a growing need for Recommendation Systems based on social and information network mining methods [20].

More and more companies are interested towards the integration of Recommendation Systems in the Intranet in order to further improve communications [20] and organizational learning.

In the work of Reichling, Veith and Wulf [21] is proposed an expertise recommender system for the specific needs of a major European national industry association. Other studies have been proposing knowledge management systems such as Luo and Cao [22] that presents an architecture to realize knowledge sharing and knowledge recommendation based on user model. Also is presented by Ale et al. [23], an architecture to provide a technological support for knowledge representation and retrieval activities.

The growing number of works related to organizational learning area shows that the search for techniques for improving learning in teams is recurring and current on software factories. The environment proposed in this study is an improvement of the consolidated approach developed by Menolli, Reinehr and Malucelli [7] and is presented in the section the follows.

IV. ENVIRONMENT

The proposed approach aims to generate the Learning Scheme using data from the searches performed by the employees on search engines, such as Google, Yahoo and Bing. The reason of choosing queries typed into search engines as source of information is because more and more, software developers are using search engines to find techniques, coding solved issues and new technologies [24]. According to QuantCast, the online programming forum "StackOverflow", increased from around one thousand accesses to more than three million visits per day since 2009. It shows that many programmers get answers for their needs on the internet, submitting a question or searching search for answers to a question that has already been made.

This approach is divided into two components. The first is called "Themes and Roles Identifier" (TRI) and works as a collector of queries typed on search engines and user's information. This component uses the collected information to suggest to the expert the themes and roles most searched.

The second component defines a course from all the information gathered on the TRI and defines the UOL structure. This component is called the "Course Definer" (CD). As a final task, based on the Learning Design, the CD will set a Learning Scheme, which finally can be used to create a Unit of Learning.

A. *Themes and Roles Identifier*

This component has as main objective to present suggestions of course's themes and users that may participate or teach those courses. To do that, we collect the employee's queries in search engines, e.g., "Java Polymorphism Examples", and the employee's information such as IP addresses, date and time of search. So, it is used text mining practices and a clustering algorithm to group these data and discover what have been the most searched queries by the group.

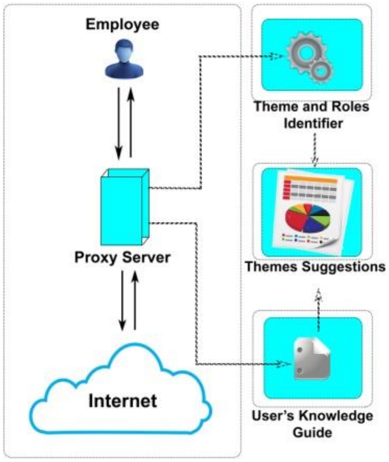

Figure 1. Approach to collect and generate the themes suggestions.

To obtain these topics is necessary to delineate a strategy to present a way of, by the analysis of a Proxy Server's Log, to generate a search engine query report. The approach to collect both the theme and the employee's information is presented on Figure 1.

Thus, with a proxy server mediating the connection between the user and the search engine is possible to capture the log of each query. However, an access log usually is not an easy reading file.

To get useful information from these records is needed to understand a record line. These records are divided into columns. First column refers to a Unix date and time, that after converted, becomes readable, for example:

1413858715.311 = Tue, 21 Oct 2014 02:31:55

The third column refers to the IP address that attempted to connect. Companies usually have a fixed address to all employees or computers, which enables pointing the employee to the connection he attempted. The proposed environment allows the expert to register the employees' information such as name and skills, and then, bind this information to the IP addresses used by the employees in the company network.

The most important information for the proposed environment is the query. To collect this information was necessary to look into each record and learn what splits the typed query, from the rest of the record. As seen in most records, some HTTP parameters on a query request differ from the normal requests. It is crucial to get only queries arising from a search engine, such as Google, then, to split the query from the record we look for parameters that mark the beginning and the end of the queries. After splitting the query, all the gathered information is record on a database.

After recording data, the result that we have is a list of IP addresses, dates and queries searched by the employees. However, even that these queries represent a necessity of an employee, not all words typed into a search engine influences the meaning of the query. As the objective is to provide relevant information to help creating a course, the solution to extract the important words was to use some techniques from text mining like tokenization, filtering and stemming.

After recording the important queries and its details, the next step is to cluster these records. The algorithm chosen to cluster the queries is the Cliques. The reason of using Cliques is because on the proposed environment, we seek cohesion between the elements of the same group, and the clustering algorithm provides it. The final result is a list of searches sorted by the most queried theme.

B. *Course Definer*

The aim of the Course Definer (DC) is to use all information obtained in the previous step, to help an expert to create a course.

The objects that are being treated in this component are texts and they represent a learning necessity of the group. It is an expert's work, to look onto the team's learning necessity and understand what the objectives and methods will be used to help improve the knowledge. Other concepts bound to a course structure still needs the choice of a professional, therefore, the proposed environment can facilitate this process and make it more interactive.

Information such as pre requisites and level of difficulty may be presented in the environment, using employee's information, as well as learners and staff levels. Hierarchy and contents may be suggested to the expert so that he can complete the course structure. The final task of the DC is to write the XML learning scheme file, which could be used by the expert to generate the course with its contents in an environment like proposed by Menolli, Reinehr and Malucelli [7].

C. *Architecture*

This section presents an architecture that gathers and organizes the components of the proposed environment, to create the learning schema.

Figure 2 provides a general overview of the proposed mechanism architecture. The architecture is subdivided into three tiers: application, middleware, and server log and database. The last block presented in this architecture, called Internet Connection, treats of the connections made by the employees and their searches.

The Application Tier is responsible for the user interaction and provides subsidy for the content inclusion, such as employees information, and creation of course structure. This tier is composed of four main structures that may be feed by the information collected and treated by the other tiers. The structures defined by the IMS Learning Design are: themes; roles; contents; and hierarchy.

The Middleware Tier provides a combination of Information Recovery, Text Mining and Clustering techniques. In addition, it makes, as a final task, the Course Structure Transformation, which takes all the data clustered and applies other text mining and clustering algorithm, but now, identifying the kind of course's structure, e.g., "Examples", "Concepts", "Exercises", etc. After rendering these data, the Middleware Tier presents them to the Application Tier, separating the information collected between the structures of the Learning Design, such as "Theme", "Roles", "Contents" and "Hierarchy".

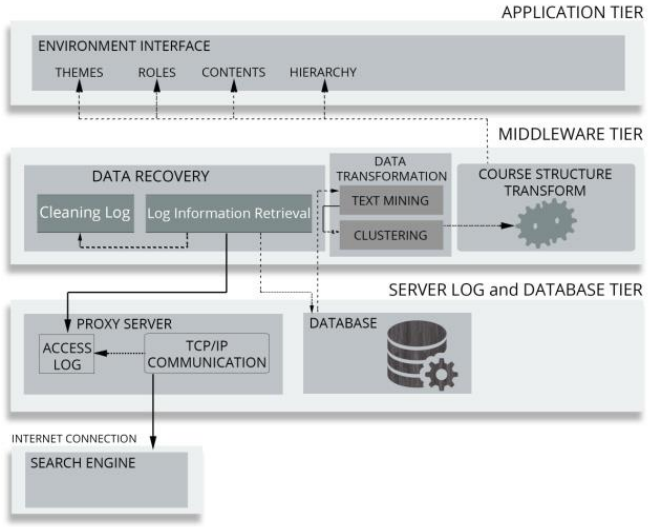

Figure 2. Proposed Architecture.

The Server Log and Database tier is responsible for interoperate between the user and the Internet Connection Tier, gathering and storing the connections attempts into a log file, and also storing the information found, after retrieving this from the server.

Thus, the objective of this architecture is the generation of courses structures, through the recovering of information shared by the team members on the internet search engines.

V. EXPERIMENT

The main objective of this experiment is to analyze if the suggested themes based on the queries performed by the development team can be considered relevant. To achieve this goal were elaborated some specific objectives that must be performed in this experiment:

- Present the environment to a development team;
- Validate the application of the approach in an corporative environment from the perspective of the employees;
- Analyze the suggested themes;

We proposed the use of a development team from a company of the vehicular tracking branch. In this experiment, four employees were selected, among which, all are linked to the development of software with a minimum of two years experience.

The areas of the software development included were interface and business layer programming and database management and analysis.

To manage the information about the employee responsible for a query, it was designed a screen which allows the expert to Create, Read, Update and Delete (CRUD) employees, as well as their specific knowledge, its levels, and the IP address used by this employee on the network. To simulate an organizational environment, a local network was built. Also, it was created four employees with specific skills and its levels. The Employees were named E.1, E.2, E.3 and E.4. Each employee was mapped with an IP address into the local network, so that the mechanism could cross the search with the user.

After created the four employees, it was necessary to collect their searches on a search engine. Each employee was asked to perform at least fifty queries on the search engine. The criteria to perform the queries, was that the subject should be related to both the projects of the company and the developer skills.

Thereafter, the employees' searches were recovered from the proxy server log and stored on the database. Then, the Data Transformation component created from the stored queries, a list of bag of words. An algorithm of stop words removal was used to maintain only the words that represent the developer needs.

In order to present those searches grouped as themes, the Jaccard Similarity Coefficient was calculated to each pair of bag of words. Jaccard Coefficient uses the ratio of the intersecting set to the union set as the measure of similarity. Thus it equals to zero if there are no intersecting elements and equals to one if all elements intersect.

The average Jaccard Index established in previous simulations was 0.7, then, to this experiment, with the purpose of finding the best clusters, the same index was used.

After clustered, queries were presented as themes by the mechanism along with the amount of times that such searches were performed by the employees. Despite the tasks given to each employee were focused on different tasks, some queries performed by the users had the same theme. The main themes researches brought by the mechanism were about the interface framework called knockoutJS. Most of queries pointed to the specific words “knockout bind context”, “nested foreach knockout” and “computed function knockout”. Other queries pointed to themes related to the programming language C#, e.g. “C# MVC partial view”, “C# trend line calc” and “Json serialize into object C#”.

In order to validate the themes suggestions and the approach, a questionnaire was applied to the development team that participated in this experiment.

The results presented that beyond using the search engines to look for answers to their development needs, there is an incentive of the company managers on improving quality by allowing the use of search engines to that end. Also is shown that despite this approach of collect the internet log might seem intrusive, the development team considered important the fact of the company know the learning necessity of them.

Regarding the suggested themes, the developers were questioned about the necessity of learning on the topics that the mechanism presented as the most searched theme. The results presented as expected that the most searched themes were a group necessity. Also, the answers pointed that the words inside the themes were related to each other, showing that the mechanism didn't mix different queries in a single theme.

The experiment proceeded to the course definer component. The expert selected the themes suggested by the mechanism that were related to the technology “knockoutJS”, then the Course Definer crossed the results with the IP Addresses in order to map the employee responsible for each search. The employees pointed for the themes selected where E.1, E.2 and E.4. None of the searches performed by E.3 were present on the selected themes, because this specific developer was not involved on interface interaction, but only data processing.

With the themes selected, the Course Definer suggested a course where the main key words were the ones found in all the selected themes. In this case, the key word suggested was “knockout”.

In order to suggest contents to the course, the course definer searched again in the queries to find other themes that may belong to the main theme. As the word “Knockout” represent a technology, this word appeared in other themes suggested, but combined with other words, such as: “css bind”, “observable array”. Thus, these themes were suggested along with the main theme, as contents to the course.

On the next step, the expert defined that to this course the main skill needed was KnockoutJS. To point and suggest the roles (learner and staff) into the course, the course definer looked into the employee's skills that were related to the main technology defined in the course, i.e., “KnockoutJS”. Employee E.2 was the only having a senior level to the skill needed. Because of this, E.2 was pointed by the CD as the main tutor to the course. E.1 and E.4 were pointed as learners because their skill levels were plenum and junior respectively. The expert stills had the option of changing the roles of the employees and include new employees to the course. We choose to include the E.4 to participate of the course as a learner.

After that, in the Course Structure Transformation component, the expert was able to determine what kind of activities the course would have, such as: concepts, examples, advantages, disadvantages; and when to use the concepts learned. The activities chosen to the course created were concepts, examples, exercises and test.

Finally, the expert informed the objectives of the course and placed the order of the contents. Thus, having the main structures defined in the Course Definer Component, the mechanism was able to generate the XML Learning Scheme. Once the XML Learning Scheme was ready, we uploaded it to the Semantic Collaborative Environment proposed by Menolli, Reinehr and Malucelli [7]. A positive result presented is that the Semantic Collaborative Environment was able to read the generated XML Learning Scheme and look for learning objects related to the themes described on the Course Definer.

VI. Final Considerations

The work presented focuses on the identification of the specific needs inside an organization. The identified themes and roles that are presented to the expert are the basis for the definition of learning schema.

However, even a trained expert, who has sufficient knowledge to generate units of learning to assist the employees, has a complex job when it comes to know that the need for discovery by company employees, not only for growth of this employee, but that this knowledge sustain it on a daily basis in the activities assigned to them.

Towards this direction of the problem, we propose a mechanism supported by concepts of organizational learning in order to contribute to the expert responsible for generating units of learning. This semi-automatic mechanism creates the units from the searches conducted on the Internet by means of search engines.

We estimated two main contributions to the completion of the proposed work. The first, is propose an approach to get the searches performed on search engines, aiming to catalogue these themes that were popular and along with a guiding company's knowledge, suggest topics of courses to be generated to the organization.

The second is to provide a component integrated into the semantic collaborative environment presented by Menolli, Reinehr and Malucelli [7], which is effective to help the expert generates the Units of Learning, in order to make the learning more effective.

Lastly, we identified some gaps that may be supplied on next works. The main gap is to run an experiment on an organizational environment, so other factors that surround the environment could be analyzed and evaluated. It would help identifying unanticipated problems and to adapt the mechanism to solve them. It is also identified the necessity of run another experiment with the objective of evaluate the Course Definer approach from a pedagogical perspective once the definition and application of a course is also related to the educational area.

REFERENCES

[1] Neves, E. O., 2011. Organizational Learning: Considerations about methodologies of development promotions. Administration and Economy University Magazine, 3, 2 – 16.

[2] Alvesson, M., 2000. Social identity and the problem of loyalty in knowledge-intensive companies. Journal of Management Studies, 37, n. 8, 1101-1123.

[3] Menolli, A., Malucelli. A., Reinehr, S., 2011. Towards a Semantic Social Collaborative Environment for Organizational Learning in: International Conference on Information Technology and Applications, 65-70.

[4] Nevis, E. C., Di Bella, A., Gould, J. M., 1995. Understanding organizations as learning systems. Sloan Management Review, 36, n. 2, 73-85.

[5] Polsani, P. R., 2004. Use and abuse of reusable learning objects. Journal of Digital Information, 3, n. 4.

[6] Koper, R., O., B., St.B.D., Ab.B. (2004). Representing the Learning Design of Units of Learning. Educational Technology & Society, 7, 97–111.

[7] Menolli, A. L., Reinehr, S., Malucelli, A., 2013. Improving Organizational Learning: Defining Units of Learning from Social Tools. Informatics in Education. 12, n. 2, 273-290.

[8] IMS Global Learning Consortium, 2003. "IMS Learning Design Information Model", Final Specification, from http://www.imsglobal.org/learningdesign/ldv1p0/imsld_infov1p0.html.

[9] Amorim R. R., Lama M., Sánchez E., Riera A., Vila X. A., 2006. "A Learning Design Ontology based on the IMS Specification: The Need for a Learning Design Ontology," Educational Technology & Society, 38-57.

[10] Rawlings, A., Van Rosmalen P., Koper R., Rodríguez-Artacho M., Lefrere P., 2002. "Survey of Educational Modelling Languages," Learning Technologies Workshop, from http://www.cenorm.be/cenorm/businessdomains/businessdomains/isss /activity/emlsurveyv1.pdf.

[11] Witten, I.H., Frank, E., 2005. Data Mining: Practical Machine Learning Tools and Techniques.

[12] Ullman, J., Rajaraman, A., 2011. Mining of Massive Datasets, 2, 307-341.

[13] Shishehchi, S., Banihashem, S. Y., Zin, N. A. M., 2010. A Proposed Semantic Recommendation System for E-learning, ITSim, 1, 1-5.

[14] Gallego, D., Barra, E., G., A., H., G., 2013. Enhanced recommendation for e-learning authoring tools based on a proactive context-aware recommender, Frontiers in Education Conference, 1393-1395.

[15] Leskovec, J., Rajaraman, A., Ullman, J.D., 2014. Mining of Massive Datasets, 2, 287-319.

[16] Liang, G., Weining, K., Junzhou, L., 2006. Courseware Recommendation in E-learning System, ICWL 2006, LNCS 4181, 10-24.

[17] Ricci F, Rokach L, Shapira B, Kantor PB., 2011. Recommender Systems Handbook. Springer.

[18] Davenport, T. H., Prusak, L., 1998. Working Knowledge: How Organizations Manage What They Know. Boston, MA, USA: Harvard Business School Press.

[19] Menolli, A. L., Cunha, M. A., Reinehr, S., Malucelli, A., 2015. "Old" theories, "New" Technologies: Understanding knowledge sharing and learning in Brazilian software development companies, Information and Software Technology, 58, 289-303.

[20] Stan, J., Muhlenbach, F., Largeron, C., 2014. Recommender Systems using Social Network Analysis: Challenges and Future Trends, Encyclopedia of Social Network Analysis and Mining, 1-22.

[21] Reichling, T., V Veith, M., Wulf, V., 2007. Expert Recommender: Designing for a Network Organization, Computer Supported Cooperative Work, 16, 431-465.

[22] Luo, Y., Cao, F., 2009. Web Knowledge Management System Based on User Model, ICIE '09, 1, 552-556.

[23] Ale, M. A., Toledo, C. M., Chiotti, O., Galli, M. R., 2014. A conceptual model and technological support for organizational knowledge management, Special Issue on Systems Development by Means of Semantic Technologies, 95, 73-92.

[24] Manning, C. D., Raghavan, P., Schütze, H., 2008. An Introduction to Information Retrieval, Cambridge University Press, 421-442.