Project-Based Collaborative Learning Environment with Context-Aware Educational Services

Zoran Jeremić[1], Jelena Jovanović[1], Dragan Gašević[2], and Marek Hatala[3]

[1] FON-School of Business Administration, University of Belgrade, Serbia
[2] School of Computing and Information Systems, Athabasca University, Canada
[3] School of Interactive Arts and Technology, Simon Fraser University, Canada
jeremycod@yahoo.com, jeljov@gmail.com, dgasevic@acm.org,
mhatala@sfu.ca

Abstract. Teaching and learning software design patterns (DPs) is not an easy task. Apart from learning individual DPs and the principles behind them, students should learn how to apply them in real-life situations. Therefore, to make the learning process of DPs effective, it is necessary to include a project component in which students, usually in small teams, develop a medium-sized software application. Furthermore, it is necessary to provide students with means for easy discovery of relevant learning resources and possible collaborators. In this paper, we propose an extensive project-based collaborative learning environment for learning software DPs that integrates several existing educational systems and tools based on the common ontological foundation. The learning process in the suggested environment is further facilitated and augmented by several context-aware educational services.

Keywords: Semantic web, ontologies, collaborative learning, project-based learning, software patterns, context-awareness.

1 Introduction

The major concern of today's software engineering (SE) education is to provide students with the skills necessary to integrate theory and practice; to have them recognize the importance of modeling and appreciate the value of a good software design; and to provide them with ability to acquire special domain knowledge beyond the computing discipline for the purposes of supporting software development in specific domain areas. Software engineering students should learn how to solve different kinds of software problems both on their own and as members of a development team. As stated by many researchers [1], problem solving in SE is best learned through practice, and taught through examples. In addition, it is essential that students learn how to exploit previous successful experiences and knowledge of other people in solving similar problems. This knowledge about successful solutions to recurring problems in software design is also known as software design patterns (DPs) [2]. However, the increasing number of software DPs, described in several different patterns' forms and stored in many online repositories, makes it difficult for students to find an appropriate pattern that could be used in a specific situation.

All the above mentioned specificities of learning software DPs indicate the need for the social constructivist approach in SE education, as well as educational services that provide students with right in time advices about learning resources and possible collaborators. In particular, an active learning paradigm is needed which recognizes that student activity is critical to the learning process [3]. Following this paradigm, we have developed DEPTHS (Design Patterns Teaching Help System) [4] - a learning environment for project-based and collaborative learning of DPs. DEPTHS integrates an existing Learning Management System (LMS), a software modeling tool, diverse collaboration tools and relevant online repositories of software DPs. The integration of these different learning systems and tools into DEPTHS environment is achieved by using the ontologies. We have also built context-aware educational services that are available throughout the DEPTHS environment. These services enrich and foster learning processes in DEPTHS in two main ways: 1) by recommending appropriate learning content (i.e., Web page(s), lessons or discussion threads), and 2) by recommending peers (students, teachers or experts) that are currently dealing with or have experience in the same or a similar software problem.

In this paper, we describe pedagogical background that this comprehensive learning environment is based on – project-based learning and collaborative learning, as well as context-aware educational support provided in the form of educational services available within each system and tool integrated into DEPTHS.

2   Project-Based Learning in DEPTHS

Having in mind that effective learning of software DPs requires a constructivist approach to be applied in the teaching process, we have identified two most important theories in this field: Project-based learning (PBL) and Engagement theory. PBL is a teaching and learning model that organizes learning around projects. Projects comprise complex tasks and activities that involve students in a constructive investigation that results in knowledge building. *The engagement theory* is based upon the idea of creating successful collaborative teams that work on tasks that are meaningful to someone outside the classroom [5]. Its core principles are summarized as "Relate", which emphasizes characteristics such as communication and social skills that are involved in team effort; "Create", which regards learning as a creative, purposeful activity; and "Donate", which encourages learners to position their learning in terms of wider community involvement. Later research inspired by this approach, suggests a generic framework called "Genex framework" [6], that describes four phases a creative process will most likely pass through, "Collect", which regards searching and browsing digital libraries, visualizing data and processes, "Relate", "Create" and "Donate". Based on the guidelines for teaching SE to students (e.g., [1]), and our own teaching experience, we believe that the presented theories provide a solid base for effective learning of software DPs. Accordingly, we based the DEPTHS framework on them.

A typical scenario for learning software patterns with DEPTHS assumes a PBL approach with collaborative learning support (Figure 1). In particular, a teacher defines a specific software design problem that has to be solved in a workshop-like manner by performing several predefined tasks: brainstorming, creating and submitting solutions, evaluating solutions etc. These activities enable and even foster active learning that has strong foundation in the engagement theory and Genex's framework.

Brainstorming has foundation in two Genex's phases, collect and relate. First, a student is asked to present his ideas about possible ways for solving the problem under study and to discuss and rate his peers' ideas. In order to get enough information to perform this task, he needs to search online repositories about software DPs and other related course content. DEPTHS makes this search more effective by providing semantically-enabled context-aware learning services for finding related online and internally produced resources (Figure 1B). Moreover, to get some initial directions on the performing task, the student uses semantically-enabled peers finding service (Figure 1A) to find people who have shared interests and are engaged in similar problems.
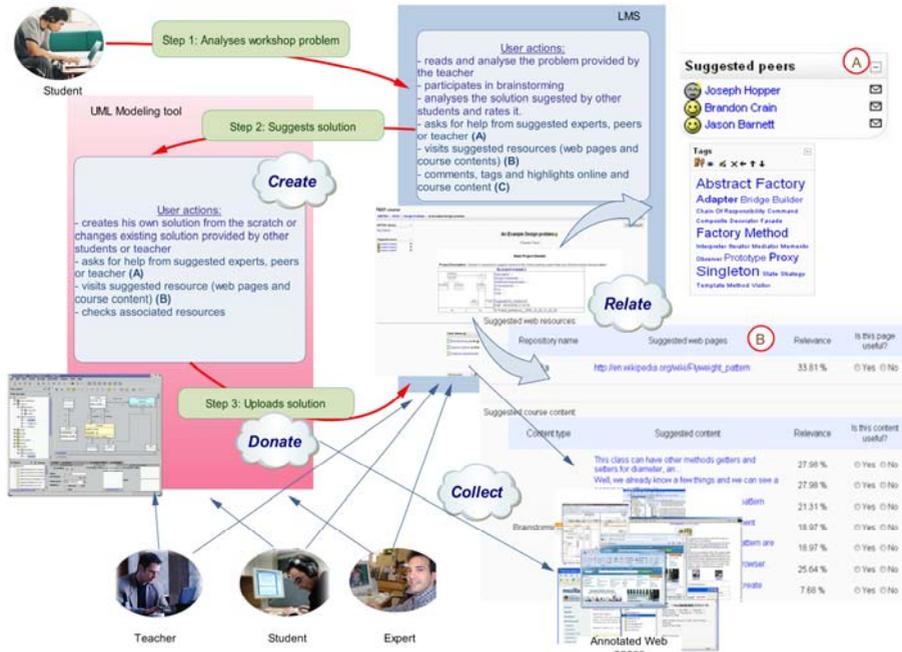

Fig. 1. An example learning scenario with DEPTHS: problem-based learning with collaborative learning support (titles in clouds indicate Genex's framework phases)

Genex's phase *create* is found in several DEPTHS activities, namely exploring earlier works on similar problems, creating design artifacts using software modeling tool or evaluating peers' solutions. Having acquired the required knowledge, students should complete the deliverable using the software modeling tool. This kind of learning activity requires students to externalize their knowledge, to analyze possible solutions and to provide a design rationale. After completing the project, students are asked to evaluate their own and other students' projects. Students reflect critically on their own and others' contributions, and acquire knowledge about other possible solutions. Genex's *donate* component in DEPTHS stresses the benefits of having authentic deliverables that will be meaningful and useful to someone else.

3   Educational Services in DEPTHS

During the last couple of years *context-aware learning* has gained a constantly increasing attention of the e-learning research community. Despite different interpretations of the term 'context' by different authors, researchers seem to agree that a learning context is about the environment, tools, resources, people (in terms of social networking), and learning activities. Being more specific, context in learning systems is mostly characterized by the learners, learning resources and a set of learning activities that are performed in the light of a specific pedagogical approach [7].

In order to provide effective, context-aware learning in DEPTHS we have developed supporting services: Semantic Annotation and Context-aware Learning Services.

Semantic Annotation Service is used for annotating online resources in publicly accessible repositories of DPs, as well as software models created by students. It analyses the text of each resource, recognizes specific domain topics (i.e., software DP's name), creates the semantic-rich metadata (based on the ontology of software DPs) and stores them in the DEPTHS's repository of LO.

Context-aware learning services are accessible to all systems and tools integrated in the DEPTHS framework and are exposed to end users (students) as context-aware learning features. They are based on Semantic web technologies, and include:

  - *Web resource finding.* Based on the student's current learning context, this service generates a list of recommended Web resources from publicly accessible repositories of software DPs. To do this, it computes the relevance of each resource (i.e., Web page) available from these repositories for the student's current learning context and selects the most relevant pages for the student. The computation of relevancy of a Web resource is based on two kinds of semantic metadata: 1) the semantic metadata assigned to the resource by the Semantic Annotation Service and 2) the formal (i.e. ontology-based) representation of the student's current learning context.

  - *Discovery of relevant internally produced resources.* This service suggests internally created resources (e.g., discussion threads, comments…) that could be useful for a student to solve a problem at hand in the given learning context. The computation of relevance is done in a similar manner to the one applied for external, Web resources.

  - *Experts, teachers and peers discovery.* Based on the current learning context, this service suggests other students or experts as possible collaborators. Collaborators are selected and sorted using an algorithm which considers their competences on three different levels: same content (i.e. current software problem), similar or related learning content (i.e. similar software problem) and broader content (i.e. software problem in the same course). Estimation of the peer's competence on each level is performed through assessing three types of competence indicators: 1) participation in learning activities, 2) knowledge level estimated by the teacher and through peers' evaluations and 3) social connections with the peer asking for help.

We have implemented DEPTHS by leveraging open-source solutions and extending them with Semantic web technologies. Specifically, we have integrated Moodle (http://moodle.org) LMS, ArgoUML (http://argouml.tigris.org) software modeling tool, OATS (Open Annotation and Tagging System) tool for collaborative tagging and highlighting (http://ihelp.usask.ca/OATS) and LOCO-Analyst tool to provide

teachers with feedback regarding students' activities [7]. Moreover, we have extended both Moodle and ArgoUML with the above described context-aware educational services and developed a Moodle module for project-based collaborative learning.

4 Evaluation

The evaluation of DEPTHS was conducted in February 2009, in the context of a course on software development taught at the Department of Computer Science of Military Academy in Belgrade, Serbia, with a group of 13 students of the fifth year of the computer science program. The students were divided into 4 groups (3 groups with 3 students and 1 group with 4 students), based on the teacher's subjective opinion about their knowledge and their grades in courses related to SE.

The aim of the evaluation was to determine how effective DEPTHS is for learning DPs. Specifically, we wanted to check if active students involvement in real world problems and the employment of context-aware educational services could ensure more effective way of imparting knowledge in the domain of software development.

We used an interview to collect data about the students' satisfaction with and attitudes towards learning with the DEPTHS system, as well as their perceptions regarding the effectiveness of learning with DEPTHS. The DEPTHS system received high marks from the students. Majority of them (53.85%) reported they have learned as effectively as in traditional way, and 30.77% reported that they have learned more than in traditional way. The students reported it was intuitive and very easy to use (76.92%), but they also have reported some technical issues caused by a software bug in the software modeling tool. The students felt educational services provided in DEPTHS are very helpful: Web resource recommendation – 92.30%; course content recommendation – 84.61%; and peers recommendation – 76.92%. They also thought that the activities provided within the tasks considerably contribute to the learning process (brainstorming – 76.92%, and evaluating each other's works – 100%).

5 Discussion and Conclusions

The work presented in this paper is related to two favored research fields: collaborative learning in the domain of SE and context-aware learning. Even though extensive work has been done in both research fields, to the best of our knowledge there were very few attempts in developing collaborative learning environments that support knowledge creation and sharing through the collaborative learning process based on the active learning principles.

Unlike other learning environments that rely upon a similar PBL-based approach (i.e., students learn collaboratively by solving practical problems), such as the one presented in [8], DEPTHS offers higher learning potential by providing access to relevant learning resources, allowing for sharing and commenting of produced learning artifacts, and facilitating context-aware learning (i.e., context-aware retrieval of peers, formal and informal learning content). In addition, having grounding our approach in pedagogical theories and best practices of collaborative learning, we can expect to provide students with better learning experience than systems (such as the

one presented in [9]) that do consider the learners current context when recommending learning resources but fail to provide adequate pedagogical baseline, or systems (e.g., [10]) that fail to consider students participation in learning activities as one of the relevant factors determining their competence in a given topic.

Aiming to develop a learning environment that would allow for effective learning of software DPs, we leveraged semantic technologies to integrate several existing learning systems and tools, and develop context-aware educational services. Our present implementation and first evaluation results convince us that this environment could significantly contribute to effective teaching and learning of DPs. Semantic Web technologies facilitate development of beneficial educational services that makes search for relevant resources and possible peers fast and effective.

We are encouraged with the results of the initial evaluation study that show very positive students' attitude toward learning in DEPTHS learning environment. Students' perception of system's usefulness is valuable and encouraging for our further research. However, the results we got still do not have a statistical power, as the participants' sample was too small. Further research is required that would include sufficient participants to ensure the general applicability of the findings. In addition, in our future work we intend to do a more precise evaluation of each specific educational service as well as a quantitative evaluation of the students' learning effectiveness.

References

1. Jazayeri, M.: The Education of a Software Engineer. In: Proc. of the 19th IEEE Int'l Conf. on Automated Soft. Eng., pp. xviii-xxvii (2004)
2. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading (1995)
3. Warren, I.: Migrating to a Teaching Style that Facilitates Active Learning. CiLTHE Stage 1 Dissertation, Lancaster University (2002)
4. Jeremic, Z., Jovanovic, J., Gasevic, D.: Towards a Semantic-rich Collaborative Environment for Learning Software Patterns. In: Dillenbourg, P., Specht, M. (eds.) EC-TEL 2008. LNCS, vol. 5192, pp. 155–166. Springer, Heidelberg (2008)
5. Kearsley, G., Schneiderman, B.: Engagement theory: A framework for technology-based learning and teaching (1999),
   http://home.sprynet.com/~gkearsley/engage.htm
6. Shneiderman, B.: Creating Creativity: User Interfaces for Supporting Innovation. ACM Trans. on Computer-Human Interaction 7(1), 114–138 (2000)
7. Jovanović, J., Gašević, D., Brooks, C., Devedžić, V., Hatala, M., Eap, T., Richards, G.: Using Semantic Web Technologies for the Analysis of Learning Content. IEEE Internet Computing 11(5) (2007)
8. Baghaei, N., Mitrovic, A., Irwin, W.: Supporting collaborative learning and problem-solving in a constraint-based CSCL environment for UML class diagrams. International Journal of CSCL 2(2-3), 150–190 (2007)
9. Ghidini, C., Pammer, V., Scheir, P., Serafini, L., Lindstaedt, S.: APOSDLE: Learn@work with semantic web technology. In: I-Know 2007, Graz, Austria (2007)
10. Yanlin, Z., Yoneo, Y.: A Framework of Context Awareness support for peer recommendation in the e-learning context. British Journal of Educ. Techn. 38(2), 197–210 (2007)