Semantically-enabled Project-based Collaborative Learning of Software Patterns

Zoran Jeremić[1], Jelena Jovanović[1], Dragan Gašević[2]
[1]University of Belgrade, Serbia; [2]Athabasca University, Canada
jeremycod@yahoo.com, jeljov@gmail.com, dgasevic@acm.org

Abstract

Teaching and learning software design patterns (DPs) is not an easy task. Apart from learning individual DPs and the principle 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. Following this paradigm, and using active learning techniques, project-based learning (PBL) and collaborative learning (CL), we have developed a learning environment for software DPs which leverages semantic technologies to integrate several existing learning systems and tools.

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 design. As stated by many researchers [1], problem solving in SE is best learned through practice, and taught through examples. Students therefore must be given a significant number of assignments, work on them collaboratively and thus prepare for the work in software development teams. 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].

Apart from learning individual DPs and the principle behind them, students should learn how to understand and apply patterns they have not seen before, how to integrate different DPs, and how to use this knowledge in real-life situations.

All the above mentioned specificities of learning software DPs indicate the need for the social constructivist approach in SE education. In particular, an active learning paradigm is needed which recognizes that student activity is critical to the learning process [3].

Following this paradigm, and using active learning techniques, project-based learning (PBL) and collaborative learning (CL), we have developed an integrated learning environment for software DPs called DEPTHS (Design Patterns Teaching Help System) [4]. DEPTHS integrates an existing Learning Management System (LMS), a software modeling tool, diverse collaboration tools and relevant online repositories of software DPs. Thus, it provides a comprehensive e-learning environment, offering a variety of learning activities combined with the software modeling through the real-world examples and enriched by context-aware educational services intended to improve learning process. To enable the integration of these different learning systems and tools in a common learning environment, we have used the Semantic Web technologies. In this paper, we described pedagogical background that this framework is based on: PBL and CL.

2. Background

We have explored a number of theories and research fields in the area of PBL and computer supported CL and identified the following three as the most important for teaching/learning software DPs: Learning through Design (LTD), PBL and Engagement theory.

In *learning through design*, students develop deep understanding of academic content by creating meaningful products that reflect their knowledge of the subject domain. These projects require that students not only learn the subject matter well enough to represent it in a final design, but also that they master the particular means of production used to create the product itself [5].

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 class-

978-0-7695-3711-5/09 $25.00 © 2009 IEEE
DOI 10.1109/ICALT.2009.106

room [6]. 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 genetic framework called "Genex framework" [7], 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".

3. Project-based Learning in DEPTHS

A typical scenario for learning software DPs with DEPTHS assumes a PBL approach with CL support. 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, submitting, and evaluating solutions etc.

Brainstorming has foundation in two Genex's phases, *collect* and *relate* (see Section 2). First, the student is asked to present his ideas about the possible ways for solving problem, discuss his peers' ideas and rate them. In order to get 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. Moreover, to get some initial directions on the performing task, the student uses semantically-enabled peers finding service to find people who have shared interests and are engaged in similar problems. Afterwards, the student has to find associations between the gained knowledge and the problem to be solved and to propose potential solution strategies. Later consultations are directed at refining the idea to accommodate criticisms.

Genex's phase *create* is found in several DEPTHS activities, namely exploring earlier works (projects, discussions or brainstorming) on similar problems, creating design artifacts using software modeling tool or evaluating peers' solutions.

Previous works on similar problems could be useful for students as they give them opportunities to learn from positive examples; and provide them with new facts and information, and an idea how to apply the same approach (design patterns) in a similar situation. Moreover, exploring previous works provokes critical thinking as it helps student think about alternatives their advantages and disadvantages. DEPTHS context-aware learning services for discovery of relevant learning resources (both external and internal) greatly facilitates this task (see Section 4).

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 project, as well as to perform evaluation of each other's work. Students reflect critically on their own and others' contributions, and acquire knowledge about other possible solutions; this helps them recognize possible improvements in their own solutions. DEPTHS uses ontologies to capture the semantic of the students' evaluations so that they can be used for recommendations as well as feedback provisioning.

Genex's *donate* component in DEPTHS stresses the benefits of having authentic deliverables that will be meaningful and useful to someone else. All students' projects are published and publicly available; they are stored together with contextual semantic-rich metadata which facilitates their discovery and reuse. Moreover, students can learn from each other as their projects became available before the final due date.

4. DEPTHS Architecture

Figure 1 illustrates the architecture of DEPTHS. It integrates existing, proven learning systems, tools and services in order to provide an effective collaborative environment for teaching and learning software DPs. In particular, DEPTHS currently integrates an LMS (Figure 1A), a software modeling tool (Figure 1B), a feedback provisioning tool for teachers (Figure 1C), a collaborative annotation tool (Figure 1D), and online repositories of software DPs (Figure 1E). This integration is achieved through a flexible underlying ontology-based framework called LOCO (Figure 1F).

LOCO (Learning Object Context Ontologies) is a generic framework capable of formally representing diverse kinds of learning contexts [8]. We define learning context as a specific learning situation, determined by the learning activity, the learning content, and the student(s) involved. The LOCO ontologies are used as the basis for storage (Figure 1J) and exchange of data among DEPTHS components.

Since different learning systems, tools and services use different formats for representing and storing interaction data, *Data Mapping Module* (Figure 1G) performs the mapping of the those native data formats into RDF triples compliant with the LOCO's learning context ontology. The resulting (RDF) data is stored in the *Repository of interaction data*. Data mapping is performed throughout each learning session in order to keep the semantic repository updated.

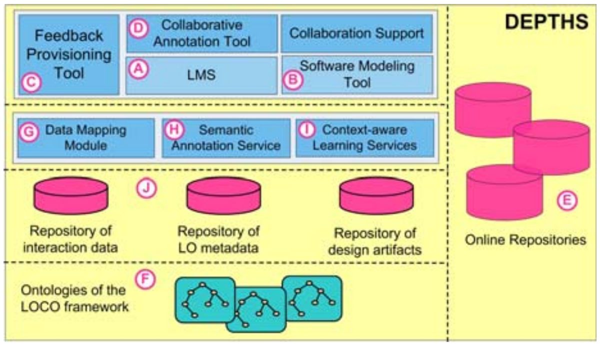

Figure 1. DEPTHS architecture

Semantic Annotation Service (Figure 1H) is used for annotating online resources in publicly accessible repositories of DPs, as well as software models (created by students) stored in the *Repository of design artifacts*. This module automatically extracts metadata based on the LOCO's ontology of software DPs and stores them in the *Repository of LO metadata*.

Context-aware learning services (Figure 1I) 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 LOCO context, DEPTHS suggests Web resources from publicly accessible repositories of software DPs that it finds relevant for the student's given context.
- *Discovery of relevant internally produced resources*. This service suggests internally created resources (e.g., discussion threads, brainstorming notes, and project description) that could be useful for a student to solve a problem at hand in the given LOCO context.
- *Experts, teachers and peers discovery. Based on the* current LOCO context, DEPTHS suggests other students or experts as possible collaborators.
- *Context-based semantic relatedness.* This service is used by other services for computing semantic relatedness between diverse kinds of learning artifacts.

We have implemented DEPTHS by leveraging open-source solutions and extending them with Semantic web technologies. Specifically, we have integrated the 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 [9]. Moreover, we use semantic annotation services of the KIM framework (http://www.ontotext.com/kim) and Sesame server (http://www.openrdf.org) for semantic repositories. In order to provide students with context-aware educational services of the DEPTHS framework, we have extended both Moodle and ArgoUML so that they can make use of these services. Moreover, we have developed a Moodle module that supports project-based collaborative learning, as described in Section 3.

5. Conclusion

Collaborative learning through project-based work promote abstraction from experience, explanation of results, and understanding of conditions of DPs applicability in real world situations; it also provides the experience of working in software development teams. Following this paradigm, we have developed a learning environment for software DPs which leverages semantic technologies to integrate several existing learning systems and tools. Semantic Web technologies provide beneficial educational services that makes search for relevant resources and peers fast and effective. These services also make learning inventive, more productive and enjoyable.

Our preliminary evaluations with the students provided us with very positive feedback. We are now preparing an extensive evaluation with a group of SE students from the University of Belgrade.

6. References

[1] M.Jazayeri, "The Education of a Software Engineer", Proc. of the 19th IEEE Int'l Conf. on Automated Soft. Eng, 2004, pp xviii-xxvii.

[2] E. Gamma, R. Helm, R. Johnson, J. Vlissides, "Design Patterns: Elements of Reusable Object-Oriented Software", Addison-Wesley, Reading, MA, 1995.

[3] I. Warren, "Migrating to a Teaching Style that Facilitates Active Learning", CiLTHE Stage 1 Dissertation, Lancaster University, 2002.

[4] Z. Jeremic, J. Jovanovic, D. Gasevic, "Towards a Semantic-rich Collaborative Environment for Learning Software Patterns", In Proc. of the 3rd Euro. Conf. on Tech. Enhanced Learning, 2008, pp. 155-166.

[5] J.L. Kolodner, D. Crismond, J. Gray, J. Holbrook, & S. Puntembakar, "Learning by Design from Theory to Practice", In Proc. of Int'l Conf. of the Learning Sciences, 1998, pp. 16-22.

[6] G. Kearsley, & B. Schneiderman, "Engagement theory: A framework for technology-based learning and teaching", 1999, http://home.sprynet.com/~gkearsley/engage.htm

[7] B. Shneiderman, "Creating Creativity: User Interfaces for Supporting Innovation", ACM Trans. on Computer-Human Interaction, Vol. 7, No. 1, 2000, pp 114 – 138.

[8] J. Jovanović, D. Gašević, C. Brooks, V. Devedžić, M. Hatala, T. Eap, G. Richards, "Using Semantic Web Technologies for the Analysis of Learning Content", IEEE Internet Comp., Vol. 11, No. 5, 2007.

[9] C. Torniai, J. Jovanovic, D. Gasevic, S. Batemen, M. Hatala, "E-Learning Meets the Social Semantic Web", In Proc. of 8th IEEE Int'l Conf. on Advanced Learning Technologies, 2008, pp. 389-393