International Journal of Engineering Education Vol. 28, No. 1, pp. 17–25, 2012
Printed in Great Britain

0949-149X/91 $3.00+0.00

Application of Project-Based Learning in a Theoretical Course: Process, Difficulties and Recommendations*

K. BURAK CODUR[1], SERÇİN KARATAŞ[2] and ALI H. DOGRU[3]
[1] Turkish Military Academy, Defense Sciences Institute, 06654 Bakanliklar, Ankara, Turkey. E-mail: kbcodur@yahoo.com
[2] Gazi University, Gazi Faculty of Education, Department of Computer Education & Instructional Sciences, L Block Room No: 306 06500 Teknikokullar Besevler, Ankara, Turkey. E-mail: sercin@gazi.edu.tr
[3] Middle East Technical University, Computer Engineering Department, 06531 Ankara, Turkey. E-mail: dogru@ceng.metu.edu.tr

This paper presents a case study about the application of a project-based learning approach. In this case study, software development projects are performed by the students, using historical software development methods in order to demonstrate evolution of the subject. The presented case study differs from others reported in the literature in its utilization of historical methods for project execution. Getting feedback and reaction of students and assessing the success of the project-based learning implementation employed are the main objectives of the case study. Concluding the case study, critical points concerning this project-based learning implementation are identified and recommendations for similar implementations are mentioned.

Keywords: project-based learning; software engineering; software development standard

1. Introduction

The beginning of the 1900s was when project-based learning (PBL) came into being. The educational philosophy proposed by John Dewey in these years was based on learning through the experiences that students have gained. Unlike classical education that was based on the transfer of the past experiences of the teacher, the new approach proposed by Dewey was regarded as progressive. In this approach the students gain experience about the problems of their era with the participation of the teacher. Continuity and interaction are important and critical features of this educational process [1]. Studies conducted in the topic of PBL have been influenced by this philosophy that has a general framework drawn up by Dewey.

In recent years, PBL approaches have increasingly drawn great interest. In particular this interest is based on principles such as conducting complex tasks, aiming to achieve real products, learning about and facing real problems in modern education [2]. PBL can also be explained as a model organizing learning via projects that generally comprise complex tasks and students have to persist until they solve the problems they face [3, 4].

In PBL, students usually work in small groups for projects. Tasks can involve examination or research on a particular topic. The topic can involve more than one field of study. Students in the same group try to come to a common conclusion by cooperating with one another in a specific period of time. In PBL, students in the group ask questions, refine these questions, discuss ideas, make predictions, collect and analyse data, depict the conclusion and share what they have found with the other group members in order to reach a conclusion [5].

The basic and common features relating to PBL are as follows [6]:

- *Introduction* aiming at forming the work environment,
- *Task* describing the essence of the work, to force the students, and also to have the quality of feasibility,
- *Sources*,
- *Process* to be followed by the students in order to fulfill the task,
- *Guidance* to support the students,
- *Cooperative* learning as a result of the group work of the students,
- *Reflections* where the results are summarized, discussed, and the problems are solved.

It is possible to see many project-based learning applications in engineering education. For example, after 20 years of experience, it was revealed that lessons taught with PBL in the senior years of the undergraduate programmes at the Technical University of Madrid (UPM) were effective in facilitating the teaching of technical, personal and contextual skills and solving real problems which are likely to be encountered in the professional field [7]. Similarly, PBL was used in a course where database design was taught to 501 students from 2004 to 2008, and it was seen that students learning via PBL had lower rates of dropping the course, better passing grades in the course, and higher rates of participating in the course when compared to the students learning via conventional methods [8].

In another study, students were required to per-

* Accepted 18 August 2011.

form four projects, each of which lasted three weeks. Then, a qualitative assessment was made by conducting interviews with the students [9]. In this study, it was concluded that, due to the PBL approach, students could acquire more rapidly and in a better way the skills in engineering design, problem solving, accessing information, engineering perspectives, laboratory skills, manufacturing a product and recognizing the relationships between the parts of the product.

In a study on computer and electrical engineering education, it was observed that students thoroughly learnt the concepts required for their casework in the computer architecture course and they developed cooperative skills [10]. The curriculum of Department of Electronic Engineering in the School of Telecommunication Engineering (ETSIT) UPM comprised four theoretical and four PBL (including a Postgraduate thesis) courses in the five years before publication of the research. The researchers examined this programme for four years; from the answers given by students to questionnaires, it was observed that students' interest in the PBL courses increased by 71 per cent and student achievement in PBL courses increased by 74 per cent. In addition, it was also found that students were able to develop more complex electronic systems [11].

In another engineering example, in 2002, at the American University of Beirut, PBL was applied to integrate Mechatronics in the existing curriculum of the Mechanical Engineering department. To this end, students were offered laboratory practices focusing on open-ended projects instead of just structured laboratory experiences during a term. At the end of the course, a questionnaire relating to the general features of the course was administered to the students. The results showed that 34 of 54 students participating in the course considered the experience to be perfect while 18 considered the course to be good for them; 36 students thought the group project approach was perfect while 15 consider this approach to be good [12].

This article presents a case study, which is used to assess the feedback and reactions of the students to a PBL implementation in a theoretical software engineering course. The PBL implementation differs from others reported in the literature in its utilization of historical methods for project execution. Section 2 discusses objectives of the work, Section 3 explains the method used, Section 4 summarizes the results and Section 5 concludes the article.

2. Objectives of the work

As in all the departments of engineering schools, the students' strength and depth of core knowledge, their learning how to learn, their ability to constantly adapt to new situations, and being inquisitive are the basic principles. Within the frame of these basic principles, in the departments of software engineering, 'process modelling' is taught as a course. In this way, it is intended that students will be able to understand and apply the software development processes used in the industry.

As demonstrated by the examples in the literature, in order to improve learning experience of the students, PBL methods can be utilized in the process modelling course. It is the objective of this study to perform a PBL implementation in a process modelling course and present feedback and reactions of the students. Thus, this case study is a contemporary and functional example of PBL implementation.

The PBL implementation in the process modelling course could have been conducted using projects convenient in the existing time frame. Another approach could utilize the evolution of process model concepts. This evolutionary approach requires performance of projects about each of the older process models. The latter method is used in this case study.

The processes applied during software development have gone through changes over the years. As emphasized by Boehm [13], these changes generally took place within a cycle between the thought that software is a product of engineering and that it is close to the craft, and certain syntheses were reached. As demonstrated in Fig. 1, the original thesis about software is to engineer it like hardware and the original anti-thesis is to approach the software as a craft. Some syntheses were reached as software engineering advanced in time. The advantage of the evolutionary approach employed in this case study is that the transitions mentioned in Fig. 1 are seen by the students, and concepts relating to the essence of software engineering are better comprehended. Thus, another significant contribution of this study is the implementation of the aforementioned evolutionary approach. This contribution also has the potential to increase interest in the case study.

3. Method

This article presents a case study. Case study is a method that is frequently referenced in the literature and which deals with phenomena in their natural environment. It is especially utilized where the boundaries between the phenomenon and the environment are not clear. Case studies focus on a particular unit of interest, such as an individual, a group, an establishment or a program. The purpose is to perform a better inspection of the phenomenon and reach a deeper understanding of it [14]. In order

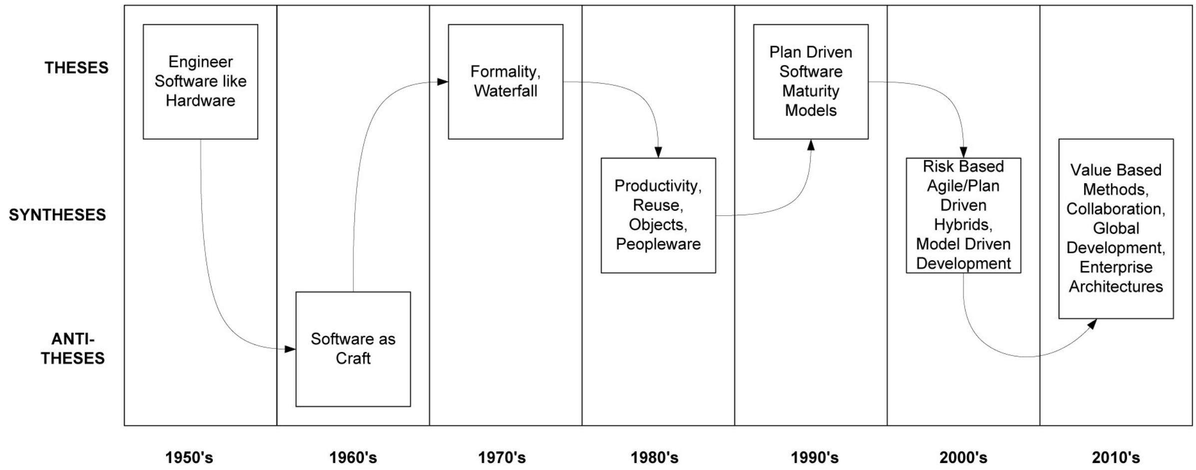

Fig. 1. Transitions in the software engineering community between theses, anti-theses stating and the syntheses, presented by Boehm [13].

to perform a case study, activities such as data collection, interviews, observations and archival research can be utilized [15]. However, case studies cannot be generalized, and this attribute is among the limitations of this method. In order to decrease the effects of these limitations, the literature suggests that case studies should be conducted in accordance with a theory [14].

The study introduced in this paper examines the application of PBL to a course given to a group of students in a software engineering programme. It can be classified as a case study. It is also compatible with PBL studies mentioned in the literature.

3.1 Design of the PBL Implementation

Considering the historical development relating to software processes, software development projects in the military domain and software development standards used in realizing them make up a special case study [16]. Military needs involve all the applications in the field of software, but address just one user (armed forces), thus the military domain is a monopsony. This single user has the power to have an impact on the development process of the products it buys. In the topic of software development, this power is used in terms of software development standards. As from the year 1978, the MIL-STD-1679 (1978–1985), DOD-STD-2167 (1985–1994), MIL-STD-498 (1994–1998), and IEEE 12207 (1998–present) standards were compulsory for military applications, i.e. all software development projects were developed in accordance with these standards, unless a waiver is issued through a formal process. These standards also define software development processes. Therefore it is possible to claim that these standards include the evolution of software development processes in their contents.

Utilizing this special case for the military domain, the PBL implementation is designed so that each student group conducts a set of projects, each in accordance with a different military standard, in the temporal order of the standards' establishments (i.e. starting with MIL-STD-1679). As the course is planned for one academic semester (14 weeks), the number of projects was limited to three. Thus, each group implements a project in accordance to MIL-STD-1679, another one using DOD-STD-2167 and final one with MIL-STD-498.

Implementations made by the students in each and every project are planning, requirement analysis, and preliminary design. At the end of these implementations, related documents are prepared by the students. The work flow diagram pertaining to this process is presented in Fig. 2. In the planning stage of the projects, students prepare the Software Development Plan document and make effort and schedule predictions relating to this project. In the stage of the Requirement Analysis, students define software requirements relating to the project and prepare a Software Requirements Specification document. In the Preliminary Design stage, students prepare a Software Design document and make prediction for the number of source code lines.

The templates of the documents to be prepared by students are conveyed to them in advance. Explanations about what they should write under each title are provided in the templates. These explanations aim to help the students understand the requirements of standards to be applied. For example, in the Software Development Plan template, there are orientations such as activities that must be planned. In the planning stage, students are expected to use these orientations while making schedule and effort predictions.

An Operational Concept Description document (describing the needs of the project's customer), a System-Subsystem Specification document (involving the requirements of the project's customer),

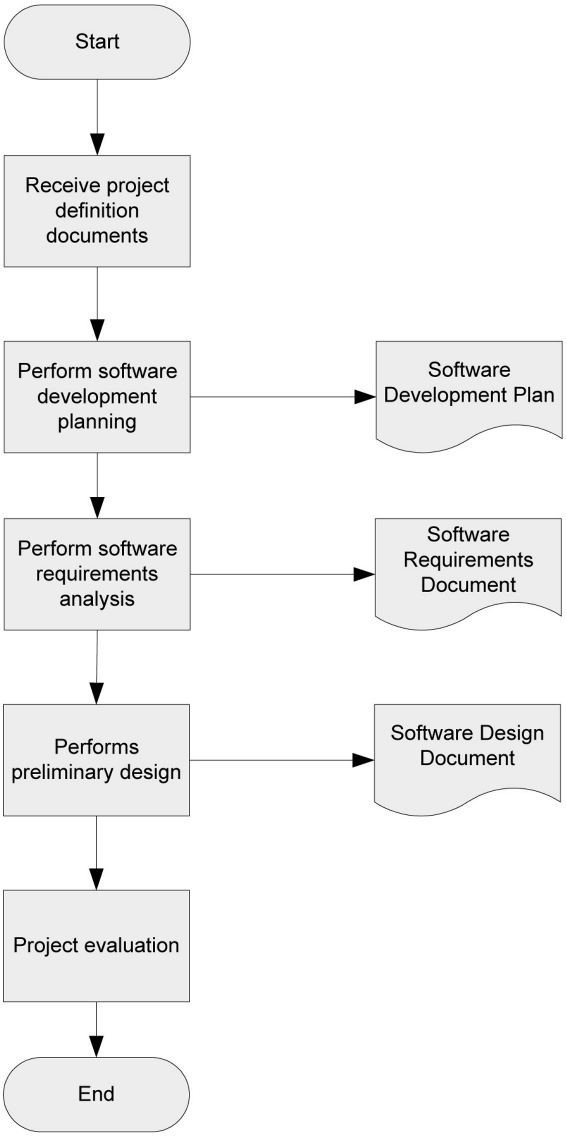

Fig. 2. Work flow diagram demonstrating steps for execution of projects by students. Repeated for each project.

and a System-Subsystem Design document are prepared for each project (these document are used also in the processes defined by the standards). In this way, an environment is created, where the projects' subjects, their requirements, and their system design are explained and a direct transition can be made to the software development activities. The project topics are chosen from military areas.

The number of projects and application calendar of these projects are determined according to various criteria. Within the scope of educational purposes, it is determined to conduct three projects so that each student group would work with each standard at least once. Based on the fact that when a project is repeated by the same group more than once, the second and third application would be affected by the first application, groups are required to work on a different project at every turn. The need to work on three standards handling a different project creates the need to determine the number of projects and groups in such a way that there will be three or multiples of three projects and project groups. An example for Group—Standard—Project distribution according to six projects and six groups is presented in Table 1. As can be seen, the PBL implementation is conducted in three stages in such a way that one standard corresponds to any given period.

As mentioned in the introduction, there are basic and common features related to PBL. This PBL implementation is also designed in the context of these features. The basic and common features of PBL application conducted in this study are explained below:

Introduction: Students are told that they will perform small-scale projects according to different standards and that each standard will involve different processes (Process Modelling course).

Task: Students are told that they will perform projects according to different standards; each of these projects will reflect historical development. They will observe how software processes have been developed via these projects; the parameters they collect will be used as a numerical analysis tool. Within this scope, it is mentioned that there is no success criterion such as 'answer key' to compare the work of the students and that what is expected from them is to reflect the requirements of the standards.

Sources: Texts of the standards to be applied by the students, project definitions and templates of the documents to be produced by the students are given to the students at the beginning of the course.

Process: Dates of performing of the projects and

Table 1. Example of project distribution according to periods (or standards) and groups

		Project					
Period	**Standard**	**1**	**2**	**3**	**4**	**5**	**6**
1	MIL-STD-1679	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
2	DOD-STD-2167	Group 3	Group 1	Group 2	Group 6	Group 4	Group 5
3	MIL-STD-498	Group 2	Group 3	Group 1	Group 5	Group 6	Group 4

calendar of each stage are determined in the context of projects. The process to be used in the projects is dependent on the standard used.

Guidance: Guidance is provided to the students in the course hours and during the assistant hours arranged on Saturdays during the term, and via e-mail at any time.

Cooperative learning: Students work in groups of three or four.

Reflections: Deficiencies seen in the projects and the data collected are evaluated together with the students in the class.

3.2 Data collection instruments

At the end of the application, a Course Evaluation Survey is planned. It consists of 10 items pertaining to student levels of learning during the course and what they have learned in connection with their working life.

The survey is designed as a voluntary online survey, which is filled over the Internet. This method provides advantages in two respects: the possibility of accessing the survey independently of time and environment and preserving the confidentiality of the participants of the survey.

3.3 Limitations

This case study is limited to a graduate course in software engineering, which was attended by 21 students for 14 weeks. Also, out of these 21 students, 16 attended and completed the Course Evaluation Survey.

4. Results and discussions

4.1 Implementation and results

The study was conducted in the Process Modelling course taught in the Software Engineering Master's Programme conducted in Middle East Technical University Computer Engineering Department in the academic year 2007–2008. A significant majority of those taking the course were software engineers working in the companies operating in METU Technology Park. 22 students attended the course and 21 took part in the study. Demographic information is given in Fig. 3, 4 and 5 in accordance with the responses given by 16 out of 21 students who completed the Course Evaluation Survey at the end of the course.

The employers of 9 out of 16 students (56 per cent) participating in the Course Evaluation Survey only worked on civilian projects, 4 (25 per cent) on just military projects and 3 (19 per cent) worked on both military and civilian projects. While domain of the employers of 12 students out of 16 participating in the Course Evaluation Survey (75 per cent) was only software, the domains of the employers of 4 (25 per cent) of the students were both hardware and software.

As shown in Fig. 4, 81 per cent of 16 students taking part in the Course Evaluation Survey had two to four years of experience. Even though the stages of development of projects can last for up to 10 years for complex system projects, a period of two years is generally sufficient for the development of software components in most of the projects. Accordingly, it can be concluded that the great majority of the students have at least one software development experience.

Fig. 5 gives an idea about the tasks conducted during their work experience. In the projects involved within this study, students were required to realize software planning, software requirement analysis and software preliminary design. As shown in Fig. 4, approximately 70 per cent of the students have worked in the fields of software requirement

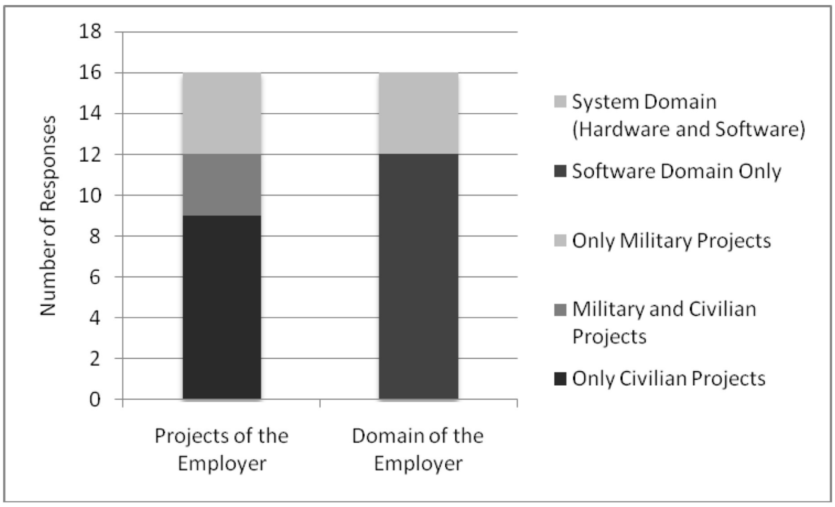

Fig. 3. Projects performed by students' employers and their domains.

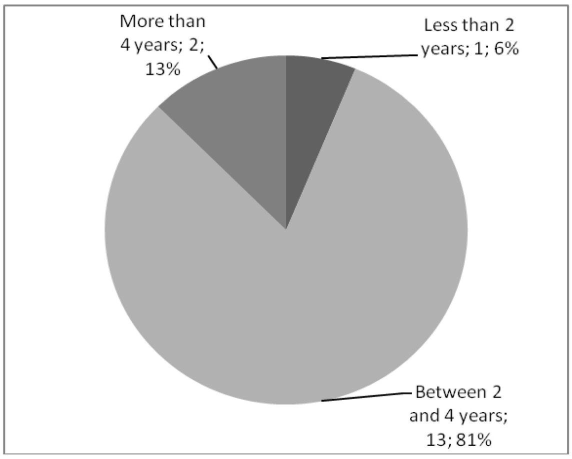

Fig. 4. Work experience of students (as duration).

analysis and software design with just a small number of students having worked in software planning.

Considering the total number of students, six groups of three or four people were determined and consequently six projects were identified. The same Group—Standard—Project distribution of Table 1 was employed.

During the implementation, the course teacher and assistants assumed the roles of the 'procurement authority' in the projects and examined the documents submitted. Students had the opportunity to ask questions about the projects and standards during the guidance hours of three hours a week and via e-mail.

Implementation of projects was conducted as homework during the term. The grades of the projects were used as the final grades of the course. In this way, it was possible to continue covering the concepts related to the course (especially the modern concepts not included in the projects) through lectures in the course hours. Data were also collected (at the end of each project phase, i.e. planning, requirement analysis and preliminary design) from the students about how many hours they worked at home while conducting the projects (Table 2). These data were reviewed throughout the course to monitor the work load of the students and allowing intervention in the event of any unanticipated increases.

The responses given by students to Course Evaluation Survey with regard to the course operations are summarized in Table 3. 14 out of 16 students who completed the questionnaire (87.5 per cent) mentioned that they allocated adequate time for the projects; 11 (68.75 per cent) stated that they were able to understand the projects by reading the provided documents; 12 (75 per cent) stated that they were able to understand the requirements and differences of three standards; 10 (62.5 per cent) stated that projects turned out to be very beneficial for their academic and business careers.

4.2 Discussion

The PBL implementation was performed with a relevant set of students, as per the demographic information presented in Fig. 3, 4 and 5. Results

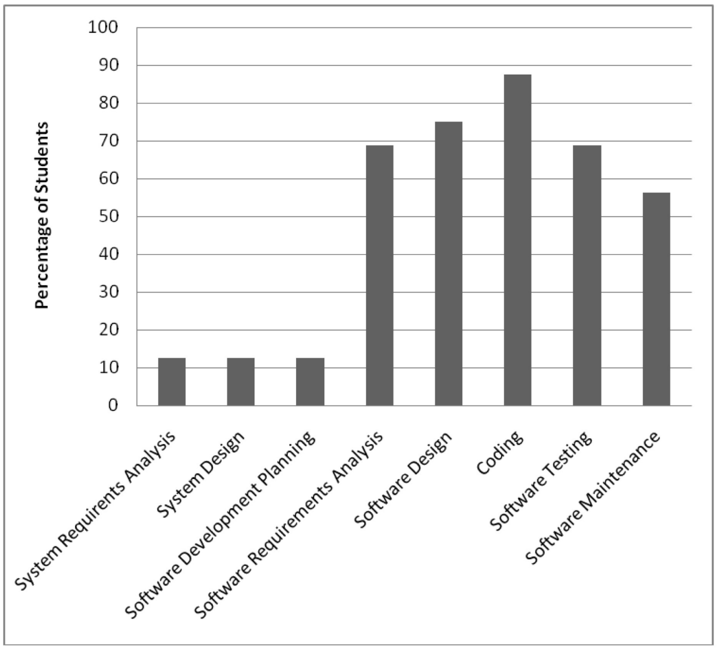

Fig. 5. Software engineering activities performed by students during careers before the course.

Table 2. Effort spent by students for different phases of the projects (in hours)

Phase:	Average group effort	Average student effort	Standard deviation	Daily time spent per student	Standard deviation
Planning	18	5.3	2.1	0.5	0.2
Requirements Analysis	14.5	4.2	1.9	0.7	0.6
Preliminary Design	14.2	4.1	1.3	0.5	0.1

Table 3. Statistical analysis of student responses to the Course Evaluation Survey

	Question	Average Score	For 3 and more	
			n	%
1	I was able to allocate adequate time to the projects (Scale 1-5. 1: I do not agree at all, 5: I completely agree)	3.8	14	87.50
2	I was able to understand the projects by reading the documents provided (Scale 0-5. 0: I could not understand at all, 5: I completely understood)	2.7	11	68.75
3	I was able to understand the requirements and differences of the three standards (Scale 0-5. 0: I could not understand at all, 5: I completely understood)	2.9	12	75.0
4	Projects were very beneficial for my academic and business career (Scale 1-5. 1: I do not agree at all, 5: I completely agree)	3.5	10	62.50

of the PBL implementation demonstrated that students gained new skills about software processes and they were able to follow their evolution.

As shown in Fig. 5, 12.5 per cent of the students taking the course and completing the questionnaire had conducted software development planning in their previous work, 68.75 per cent had worked in software requirement analysis, and 75 per cent had designed software. From this point of view, it is thought that the course brought knowledge and experience to the students especially in the topic of software development planning.

Examining the content of the documents submitted in the projects (Software Development Plans, Software Requirements, Software Design Description documents), it was found that students were able to handle projects according to the content of the standards and they were able to use the choices offered by the standards within this framework. The software development models used in the projects are an example which can be evaluated within this scope. The software development models chosen to be used in the projects are shown in Table 4. It can be seen that students suggested the use of different models in the projects developed with MIL-STD-498 which allows a wider variety of software development process models.

However, it was found that students had difficulty in two main points especially when working with the old standards. The first difficulty was working with concepts that are not being used today. Some concepts in the standards are not currently used. Accordingly, students had not learnt about how to apply them in their previous educational life. The impact of this knowledge deficiency became apparent when lecturers needed to intervene with the submitted documents. For example, lecturers demanded revising of the software planning documents in the projects performed for the MIL-STD-1679 which was the first standard to be applied in the projects and also the first standard in the historical order. This was because it was observed that topics such as use of formal methods (which can be related to the expression "Engineer software Like Hardware" of Fig. 1), which are not mentioned in the current standards, were not included in the planning. On the other hand, application of these concepts is necessary to understand the processes in the past and the change occurring in these processes. In fact, these concepts are expressed in texts of the standards which had also been given to the students.

The second difficulty was the application of current concepts to old standards. For example, while risk analysis exists in modern applications, this activity is not present in some standards examined in the course. What is expected during the projects is for students to perform the activities mentioned in the standards and not to do more

Table 4. Software development models used in the projects by the students.

	Project 1	Project 2	Project 3	Project 4	Project 5	Project 6
MIL-STD-1679	Waterfall	Waterfall	Waterfall	Waterfall	Waterfall	Waterfall
DOD-STD-2167	Waterfall	Iterative	Waterfall	Waterfall	Waterfall	Waterfall
MIL-STD-498	Spiral	Waterfall	Incremental	Waterfall	Spiral	Waterfall

than that. Otherwise, it will not be possible to understand the development of the processes and differences between them. It was observed that while working with old standards, modern concepts that were not in the scope of the standards were used by some students.

According to the literature, handicaps of the PBL process are as follows:

- working in groups in order to perform project activities;
- interaction of students with other group members;
- communication with project shareholders;
- timely sharing of the activities [17].

In this study, the difficulty of communication with project shareholders attracts attention. It led to the formation of the idea during the discussions with the students that they were unwilling to perform the projects. Even though complaints were made about comprehensibility of the projects and standards from time to time, just two groups asked questions during the guidance hours and only seven questions were asked by all the groups via e-mail during the whole term. Other examples in the literature also emphasize that cooperation of students is a crucial source of motivation [10]. Unfortunately, it is thought that students missed this point in this particular example.

The impacts of handling PBL implementation as homework are given in Table 2 as student working hours. Considering the most negative project data, a student had to work on the average, 0.7 of an hour per day in the requirement analysis phase of a project. The work pace revealed in Table 2 was evaluated as sustainable by the lecturers. However, the total hours spent for each project indicates that it would be necessary to work more intensively as the submission deadline approached unless the work was distributed over the given time in a disciplined way. This is also compatible with the projects in real life.

Examining Table 3 it can be concluded that students were mostly satisfied with the course; they understood it and it was beneficial for them. However, it is worth noting that more than a quarter of the students gave negative responses to statements other than the statistics of allocating time and five students did not even complete the questionnaire. When these statistics are combined with the findings given above about the tendency of using guidance opportunities related to the projects, it is considered that the course was not as popular as the PBL applications described in the literature. This may result from the fact that projects presented in this study were theory-based. Engineering students may expect project products to be concrete.

As a result, the PBL application case presented in this paper reached its educational goals, providing desired knowledge and experience for the students. But this single application proved to be unpopular when compared with other successful PBL application examples.

5. Conclusion and recommendations

This article presented a case study where a process modelling course in software engineering was conducted using PBL. In the course, the topic was explained through the evolution of processes over a period of time. Students were required to perform projects by applying the standards of previous decades in a special domain (military projects). In this way, this PBL application differed from other applications mainly in the sense that projects were performed with methods used in the past.

Although the PBL application satisfied the teaching goals (since the students gained new skills about software processes and they were able to follow the evolution of software processes), the course was not popular among the students, as demonstrated by their enthusiasm and motivation for participation. Also, there were some implementation difficulties.

As mentioned in the 'Results and Discussions' section, students had some difficulties in applying methods used in the past. They also tended to include methods in their projects that were currently in existence but were not used at the time of the standards utilized in the course. Therefore this study has given examples of the difficulties likely to be experienced when PBL is applied to projects that have an historical dimension.

The observations from this study indicate that it is crucial that students adapt to the virtual context (and the environment, technology and rules related to these contexts) in PBL applications including the repetition of methods that were used in the past. It is necessary to discuss this topic specifically within the scope of Introduction and Guidance among the basic and common features related to PBL.

The adaptation of students to the virtual context formed within the scope of PBL can also be associated with the motivation they feel for the course. The fact that students did not participate in the coaching activities for this PBL implementation in the expected ratio can be seen as lack of motivation.

An additional important point is that this PBL implementation was applied in the context of the projects being undertaken as homework. When the work load of students is kept at a reasonable level through frequently obtained feedback during the course, PBL can be implemented as homework. The advantage of this situation is that it allows the use of course hours for the purposes of lectures and reviewing the projects.

The case study presented in this paper provides an introductory example to potential future PBL implementations with historical dimension. The conclusions drawn from the study sets the path that should be considered for similar future applications. If additional and diverse applications are conducted in the future, the education community will certainly benefit from the experience.

References

1. J. Dewey, *Experience and Education*, Kappa Delta Pi Publications, 60 Anniversary Edition, 1998.
2. J. Ravitz, J. Mergendoller, T. Markham, C. Thorsen, K. Rice, C. Snelson and S. Reberry, Online professional development for project-based learning: Pathway to systematic improvement, *The Association for Educational Communications and Technology,* October 21, 2004. Chicago, IL.
3. J. Thomas, A review of research on project-based learning, 2004 http://bobpearlman.org/BestPractices/PBL_Research.pdf (Accessed 13 September, 2010).
4. M. Tuncer, Proje Tabanli Ögrenme ile Problem Tabanli Ögrenmenin Fark ve Benzerlikleri [The similarity and differences of project based learning and problem based learning], *e-Journal of New World Sciences Academy,* **4**(2), 2009.
5. S-F. Lam, R.W. Cheng and H.Choy, School support and teacher motivation to implement project-based learning, *Learning and Instruction,* **20**(6), 2010, 487–497.
6. M. M. Grant, Getting a grip on project-based learning: Theory, cases and recommendations. *Meridian: A Middle School Computer Technologies Journal,* **5**, 2002.
7. I. Ríos, A. Cazorla, J. M. Díaz-Puente and J. L. Yagüe, Project–based learning in engineering higher education: two decades of teaching competences in real environments. WCES Procedia. *Social and Behavioural Sciences* **2**, 2010, 1368–1378.
8. C. Domínguez and A. Jaime, Database design learning: A project-based approach organized through a course management system, *Computers & Education,* **55**(3), 2010, 1312–1320.
9. M. Frank, I. Lavy and D. Elata, Implementing the Project-Based Learning Approach in an Academic Engineering Course, *International Journal of Technology and Design Education* **13,** 2003, 273–288.
10. A. Martínez-Monés, E. Gómez-Sánchez, Y. Dimitriadis, I. M. Jorrín-Abellán, B. Rubia-Avi, and Vega- G. Gorgojo, Multiple Case Studies to Enhance Project-Based Learning in a Computer Architecture Course. *IEEE Transactions on Education,* **48**(3), 2005, 482–489.
11. J. Macías-Guarasa, J. M. Montero, R. San-Segundo, A. Araujo and O. Nieto-Taladriz, A Project-Based Learning Approach to Design Electronic Systems Curricula, *IEEE Transactions on Education,* **49**(3), 2006, 389–397.
12. A. Smaili and S. Chehade. Effective Integration of Mechatronics into the Mechanical Engineering Curriculum: A Cooperative, Project-Based Learning Model with Seamless LabLecture Implementation, *International Journal of Engineering Education,* **21**(4), 2005, 739–743.
13. B. Boehm, "A View of 20th and 21st Century Software Engineering", *ICSE 2006 Keynote Address*, May 25, 2006.
14. R. K. Yin, Case study research (Design and Methods), 3rd ed., Sage Publication, 2003.
15. D. Ary, L. C. Jacobs, A. Razavieh and C. Sorensen. Introduction to research in education, 8th ed., Thomson Wadsworth, 2010.
16. K. B. Codur and A. H. Dogru. Evolution of software development standards in the military domain and effects on software applications, *in: Proc.IWPSE-EVOL'09 (Amsterdam, The Netherlands, August 24–25, 2009)*, pp. 41–45.
17. H. Keser and D. Karahoca, Designing a project management e-course by using project based learning. WCES-2010, World Conference on Educational Sciences, Bahcesehir University, Istanbul—Turkey, 04–08 February. *Procedia–Social and Behavioural Sciences Journal, 2.* 5744–5754

K. Burak Codur received his Ph.D. in Technology Management in 2009 from Defense Sciences Institute of Turkish Military Academy. His research interests are software engineering and management of defense projects.

Serçin Karataş received her Ph.D. in Education Technologies in 2005 from Education Sciences Institute of Ankara University in Turkey. She is currently an associate professor at Gazi University, Department of Computer Education and Instructional Technologies. Her research interests are theoretical foundations and applications of e-learning and distance learning, management of distance learning, instructional design, storyboard and course content development for e-learning, principles of e-learning media, interactivity application in e-learning, instructional material development and instructional graphics and animations.

Ali H. Dogru is currently with the Computer Engineering Department, METU as an associate professor. He received his Ph.D. degree in Computer science from the Southern Methodist University in 1992. Having founded and coordinated the Software Engineering Master's programme, he has been involved with research related to Software Engineering and especially Software Product Lines and Software Architecture related fields. He has many publications in these fields including books and also has led many masters and doctorate level thesis work. He has conducted joint project work with the industry, mostly as a consultant, besides taking part in the training activities for the industry and various organizations.