Deployment of Capstone Projects in Software Engineering Education at Duy Tan University as Part of a University-wide Project-based Learning Effort

Duc Man Nguyen, Tien Vu Truong, Nguyen Bao Le
Duy Tan University (DTU)
Da Nang, Vietnam
mannd@duytan.edu.vn, vudalat@yahoo.com, baole@duytan.edu.vn

***Abstract*- One of the biggest challenges for schools and institutions offering IT-training programs in Vietnam is how to minimize the amount of retraining by IT companies and corporations for new IT graduates due to the gap between college education and real-world practices in IT. In particular, how can our IT graduates acquire the right knowledge, skills and attitudes to fit in our IT labor market?**

There are many solutions to the above problem like restructuring the curricula, integrating the materials of higher-division courses, or adding in practical skill-set training, etc. However, the most effective solution from our experiences is to change our current methodology of teaching and learning IT in Vietnam; specially, enhancing our students' capability through more use of real-world IT projects. This is usually known as the Project-Based Learning (PjBL) approach in which students explore real-world problems or challenges, and simultaneously developing interdisciplinary skills while working in small collaborative groups or teams. Since project-based learning is involved with active and engaged learning, it inspires students to acquire a deeper knowledge of the subjects they are studying. In this paper, we will present some current alarming issues of the IT training and education situation in Vietnam and how the Faculty of Information Technology of Duy Tan University tackles those issues through its deployment of major IT Capstone projects as part of a university-wide Project-based Learning effort.

Keywords: learning outcomes, IT labor force, Project-Based Learning (PjBL), simulation, case study, Capstone project, integrated active learning, experimental learning, learning assessment, Design-Implement

I. INTRODUCTION

Currently, there are over 260,000 people working in the IT sector in Vietnam (including in hardware, software, digital content, etc.) with the growth rate of 13%~18% a year [1][2]. The number of universities and colleges which have IT training programs also increased from 192 in 2002 to 277 in 2010 [1]. The enrollment quota for IT training programs also continues to increase: about 30,000 in 2006 to over 60,000 [1]. On average, the quota of the following year is 6%~8% higher than that of the current year. However, according to the annual report from IT schools, while the quotas for IT training always increase, the actual number of students registering for IT programs in Vietnam is declining 10%~15% per year [1][2].

The annual decline in the number of students registering for IT programs would lead to the decline in the number of workers in IT who help meet the increasing need of the IT labor market. Besides, the quality of workers in IT is another critical and crucial subject. According to recent forecasts, by the year of 2020, the IT labor market of Vietnam will need about 600,000 new workers while the training system can only provide approximately 400,000 [1][3]. In addition, the percentage of graduates who can meet the real-world requirements of the IT industry without any retraining is expected to be low.

Based on reports from many IT businesses, the current IT labor force of Vietnam has many limitations, including: lack of expert knowledge on project deployment or systems implementation, lack of experience with large-scale and/or complex IT projects, lack of professional work practices, poor language ability, weak soft-skills, insufficient knowledgeable about customers and their cultural values, inadequate management and project-management capability. Meanwhile, the majority of new graduates in IT in Vietnam are not able to quickly adapt to the working environment because of the big gap between their training and the real world. Most Vietnamese college students also do not have adequate soft skills, project-management skills, teamwork skills, negotiation and communication skills, etc. As a result, many IT businesses in Vietnam complain that they can hardly recruit the right personnel, and most of the time, they have to spend a couple of months or more training new recruits [3].

978-0-7695-4960-6/13 $26.00 © 2013 IEEE
DOI 10.1109/LaTiCE.2013.27

Being an institution specializing in IT training in Central Vietnam, Duy Tan University recognizes that there should be immediate change to the IT education in Vietnam. In particular, we need to change and reinvent how we assess learning outcomes, how we evaluate IT curricula, how we utilize different technologies for teaching and learning as well as how we raise the public awareness about quality standards in IT training. The purpose is to bridge the gap of IT practices of the academia versus those of industry and of developing countries versus those of developed countries so as to produce a new generation of IT workers in Vietnam, who have the knowledge, skills and professionalism that stakeholders demand. There are a number of solutions to this problem including restructuring IT curricula toward "on-the-job" training approach, integrating and building interdisciplinary knowledge base, utilizing new technologies for teaching and studying, etc. However, our major solution here will be to set up comprehensive Capstone projects which help link up different knowledge as well as different real-world stakeholders together, using the Project-based Learning (PjBL) approach. In fact, according to the ACM curricula for Computer Science and Software Engineering of 2001 and 2004, respectively, the Capstone project is always emphasized as a very important component for students to bring together the skills and knowledge learned in order to resolve some problem [13][14].

II. REINVENTING IT TRAINING WITH PROJECT-BASED LEARNING

Reinventing IT training using the PjBL approach include a series of activities in which the learning process takes place through in-school or real-world development and implementation of a software product, a network system, an information system, etc. Such an approach yields the following positive impacts on the quality of IT training and education:

- Students will develop interpersonal and communication skills through interaction with different stakeholders in their projects [13][14].
- Students will be motivated to acquire new skills and knowledge in software design and development in order to carry out the projects for client businesses in the industry [14].
- Students will effectively learn about theoretical IT knowledge by practicing it in real-world projects.
- Students will benefit from the active learning methodologies included in the PjBL approach.
- Students will become more dynamic and independent in their development, application, and review and/or evaluation of various IT ideas [13].

In short, what students go through will be more of a "learning-by-doing" experience [15]. This learning experience, on the other hand, is one of or a mix of the following learning modes:

(1) Observing - Reflecting: learning through observing the work of others or contemplating on one's own work

(2) Theorizing: learning through concept development, analysis and synthesis based on past observations or experiences

(3) Experiencing: learning through one's own actions and/or activities and/or behaviors

(4) Testing: learning through experimentation or selection amongst a certain number of available options

The above learning modes, in turn, can be arranged for through the following school work or exercise formats:

Simulation

Simulations, which are often used in scientific research, are the processes of model development, object modeling and process iteration. Instead of having to study certain macro-issues which may prove to be impossible or extremely costly, we model and conduct the operation and interaction of similar and/or related objects in the laboratory. The end results will be subsequently verified with real-world measurements. Many simulations can be implemented on computer hardware and software [13]. Based on the results obtained from those simulations, we will be able to draw up the path for actual IT research and development later on. Simulations in school will provide students with certain skills like Modeling skills, Testing & Integrating skills as well as Graphics & Design communication skills.

Case Studies

The key elements of most case studies are established based on actual situations that IT workers and their stakeholders usually run into. The main purpose will be to describe and exchange experiences about typical dilemma situation, and how to analyze and resolve problems involved. Across different situations, students will need to solve many problems within some predefined amount of time and resources. Students will be put in a position, in which they need to make their own decision or to call for support from other stakeholders. A variety of situations in different case studies will not only help students develop their own creativity and initiative but also help them with a sense of comfort even in the face of challenges later on. Case-study exercises in school will definitely help students with their Solution Formulation, Estimation and Quantitative Analysis capabilities [14].

Real-world Projects

Real-world projects in IT are usually complex ones which link up many different tasks for an end result. In order to successfully carry out a real-world project, students are required to possess skills and knowledge from different disciplines as well as the ability to work alone and in teams at different points in time. Typical activities in a real-world IT project include requirement engineering, design thinking, product development/testing/maintenance, and other decision-making activities related. By participating in these activities independently or as a team member, students will be able to develop different technical and interpersonal capabilities in their search for the most practical solution or end result. Successful completion of real-world projects will help students develop skills in Assumption Making, Design & Implementation, Writing, Communication and Presentation [4].

With the purpose of setting up major Capstone projects to provide students with a design-implementation experience, it seems that the structure of real-world projects is most fit for that of Capstone projects. However, in order to design for controlled end results in Capstone projects, the use of simulations and case studies in the framework of a real-world project actually will yield better results in terms of students' understanding of the materials of interest. A mix of simulations, case studies, and real-world projects in a Capstone project will also help accommodate for the capacity of students and faculty mentors in handling complex implementation and testing requirements of IT projects. And for students to effectively developed important IT skills, from our experiences, they must participate in at least two Capstone projects and these projects must be closely tied up with the training curricula.

III. IMPLEMENTATION OF IT CAPSTONE PROJECTS AT DUY TAN UNIVERSITY

In this paper, we will demonstrate how Duy Tan University developed and deployed Capstone projects at the Faculty of Information Technology following the PjBL approach:

A. Capstone Project and its Purpose:

Capstone projects are set up for students to solve some practical problem through the integration of many skills and learned knowledge to develop new ideas, design, and implementation approaches [14]. For Duy Tan University, the main purpose of IT Capstone projects is to provide students with a sense of "real-world", "on-the-job" activities in their study of IT product development. By following the PjBL approach, Capstone projects at Duy Tan are facilitated by different PjBL teaching/learning methodologies like active learning, group learning, experimental learning, etc. Another purpose for the adoption of Capstone projects and PjBL approach at Duy Tan University is to help with the application for ABET of IT programs at the Faculty of Information Technology and the International School of Duy Tan.

Specifically, there are two Capstone projects for students majoring in Software Engineering, Information Systems and Network Security at Duy Tan University: one at the end of the sophomore year and one at the end of senior year. Since the Capstone project at the end of the sophomore year mostly provides an introduction and simulation of the real-world IT industry, we will focus on the Capstone project at the end of the senior year. For this Capstone project, students in teams will have to work with some real-world IT business client to design, develop, and implement an IT product or solution, for a period of 6 to 8 months. Faculty mentors from Duy Tan University will also be assigned to each Capstone project to ensure smooth communication with the business clients and to support the students when needed.

B. Project Structures and Time Frames:

In the senior Capstone project, students will be assigned to teams of 3 to 5 members. The team assignment is made so as to ensure that the skills and capabilities of each team member will supplement for those of others as indicated by most ACM/IEEE curriculum guidelines [13][14]. Each team will have to bid for one of the real-world projects from several IT-business clients of Duy Tan University. IT-business clients of Duy Tan are usually software companies from Da Nang, Hanoi or Ho Chi Minh City. One or two IT faculty mentors of Duy Tan will also assigned to each Capstone project. These faculty mentors are usually the ones who have connected the IT-business clients with Duy Tan University in order to ensure smooth communication [13].

Of the 6 to 8-month period of the senior Capstone project, there will be two stages - each lasts for 3 to 4 months. During the first stage, each student will need to spend a minimum of 16 hours per week for team interaction, regulation and rule formulation, conflict resolution, software process selection, and proposal write-up. It is expected that the first stage will take up to 192 hours (i.e., 16 hours x 12 weeks) per student in every team, but as soon as the project proposal is accepted by the business clients, the team will move on the next stage. In the second stage, students will have to spend time on requirement analysis, product architecture, detailed design, coding, testing, product implementation and maintenance. At least 40 hours of work per week or a total of around 480 hours will be expected from every student in the Capstone project for this stage. In short, students will go through every step of the complete software development life cycle, and they need to understand what to carry out in each step for each role in the team. Of course, there may be some deviations depending

on the software development methodology the students select, for example, it will be quite different between an AGILE approach and a plan-driven structure.

C. Specific Implementation Tasks in a Capstone Project:

- Team Formulation:

One of the first activities in a Capstone project is the skills-review session in which each member will identify how he or she may best contribute to the project. Together, the team must define roles, responsibilities for each team member [14].

While nobody in the team is "the Boss", a project-management role is needed to help coordinate, plan, and track the team's activities and efforts. The Project Manager (PM) keeps the team focused and makes sure that various tasks get done on time. The PM will also review each member's work to make sure that they meet the technical and quality requirements. The PM is the most important role in any IT Capstone project, as a result, the team need to select the most skilful, persuasive and energetic member for this. Sometimes, following the guidelines from PjBL, this role will be rotated amongst team member from one development phase to another so that everyone will have a chance to play it [15].

Beside the PM roles, there are other roles that team members must also identified and assigned such as Software Configuration Manager, Software Quality-Assurance Manager, Testing Manager, etc. Since there are several development phases, team members can take turn switching roles with one another in each phase.

- Creation of Product Ideas and Project Structures:

In this step, IT-business clients will hold a series of presentation to introduce their projects and related problems and/or solution suggestions with all Capstone teams at Duy Tan University. Based on their capacity and capability, each team will bid for the most suitable project. After finishing the bidding process, each team will be assigned one or two mentors for their project. The team will actively plan for meetings to design, develop and implement the project. Even though the project idea may not be original because it is presented by the business clients from the first place, students are encouraged to develop original ideas in their team formation or technology application for the project at hand.

- Design & Implementation:

Before any actual design and implementation activities start, successful team formulation and continuous maintenance of the team effectiveness later on is arguably the biggest challenge to most students. From the beginning, team members do not have the right to select one another. Then, they have to join force together for the most important project before graduation. But because of this, team members need to put aside their differences, their interests, and their ego so as to develop a strong team for the project. It is important for students to understand that the Capstone team is "larger" than themselves and every team member must jointly contribute to the success of the project [13][14]. Faculty mentors and business clients will continuously provide students with inputs about the expectations and goals they have for the project, and students need to adjust themselves and their team accordingly. Of course, all of these come with great challenges, and by overcoming those challenges, students will also meet several requirements of the PjBL approach in terms of their communication, teamwork and professionalism [17].

Once the team and its project are in place, team members need to actively assess their capabilities and the feasibility of the project. Specifically, they need to ask themselves the following questions:

- Is the project likely to be a success?
- Can the team complete the project within the expected schedule?
- What are the risks or difficulties of the project?
- Do we have the right people for the project?
- Do team members have adequate knowledge and skills to complete this project?
- Does the team have a clear vision of the product concept and a good plan to carry out the project?
- Has the team defined its project goals, outcomes and timelines?
- Do team members understand well their roles and responsibilities?
- Do team members hold each other accountable for the project timelines, commitments and results?
- Do team members cooperate to accomplish the goals?
- Do team members communicate clearly and honestly with each other?
- Do team members feel responsible and accountable for the team's achievements?
- ...

If team members are honest to themselves and to each other in answering the above questions and if the answers to those questions are mostly positive, then the team may start its Capstone project. If otherwise, then the team must discuss it with the faculty mentors and business clients for proper guidance and support. Problems left unresolved will often become risks and threaten the success of the project later on [15].

The next step after the capability and feasibility assessment is about how to set up an effective project plan. Even though the project plan is usually put together by the PM, it is important that the plan be discussed and agreed on by all the team members to ensure their commitment for the project [17]. Depending on the software development process selected, the PM will build detailed plans which are consistent with all the criteria and requirements of that software development process. Later on, especially before each development phase of the software development process, the PM needs to share all information about the actual number of hours that each team member will have to spend for their tasks. This will create a sense of fairness and team members can also plan for their next tasks after that [17]. In addition, it is vital to assign a backup person for every task in case any one member may not be able to complete it.

As the design work goes on, the team needs to discuss with its business clients from time to time about the desired scope of the project. When the project scope is finally approved, the team will need to revise the Software Requirements Specification (SRS) to fit the approved scope. They may also have to revise the use cases and use-case diagrams and narratives designed previously as well as to fill in missing details and to indicate all the preconditions, post-conditions and triggers for each use case. Once all of these are done, the Design phase is basically completed. Design work, however, does not stop here: depending on the problems and risks encountered later, the PM may decide to go back and revise the design accordingly.

With the project design ready, the team needs to identify specific tasks to be carried out in each phase of the software development process. This can be done by breaking down each product function and/or component into a number of tasks which are unique by the type of capability required to complete them [15]. These tasks will then be assigned to team members by the PM, and the team members can estimate their effort for these tasks by asking some of the following questions:

- How long does it take to complete a certain task?
- Which tools are needed to complete a certain task?
- Can this task be carried out at the same with another?
- What may happen to other tasks if it takes too long to complete a certain task?
- ...

By incorporating answers to questions like the above, the PM will be able to develop an overall project schedule. Since not all tasks can be carried out in a parallel manner or one task will have to depend on the outputs of another, the PM as well as every team member needs to identify the dependencies among their tasks in order to set up an orderly sequence for task execution. (Note: In setting up the overall project schedule, the team should not look for perfection, but rather, the schedule should be deemed as a tool for tracking and reporting on the team's progress and may be subject to change when needed.)

In the process of project implementation, many tasks of coding, testing and redesigning will be carried out in sequence or at the same time in a complex manner. As a result, the PM needs to monitor every effort of each team member on a weekly basis and the team also needs to meet at least once per week to discuss project progress [15]. The main purpose is to identify any unexpected anomaly in the overall project schedule. Weekly meetings also provide a chance for team members to identify what task or function have been fully completed or partially completed or uncompleted as well as to explain why it is so. By listening to what the team members have to say, the PM will be able to understand why project progress is deviated from the original plan and in what direction [18]. The PM will then document all the changes in different tasks, task requirements, task design, or task implementation plan of each team member, and reflect those changes back to the SRS by adding, dropping or revising certain functional and/or nonfunctional requirements and use cases. Later on, the team may use the documentation on changes to its overall project schedule and SRS to determine if such changes bring about better quality for the final product.

Quality is an important component in the PjBL implementation philosophy [18] at Duy Tan University for its IT majors, specifically. Quality not only has to do with the fineness of the end product but also with the efficiency and effectiveness of the processes used to create that end product [18]. That said, team members must follow a formal process in designing, coding, testing and implementing their products. Any detraction from that process must be identified for team members to learn from their mistakes, if any. Formal process structures are also known to help enhance teamwork and mutual collaboration. However, this does not mean that students always have to follow certain formal process, changes to a process is welcome as long as they help improve the overall project efficiency and are approved by faculty mentors.

Right before the closing day of the senior Capstone project, all the teams will need to schedule a meeting with their faculty mentors to review on different things that they have learned from the project. The following items will have to be prepared before that meeting:

- Evaluation about the quality of the project (including the quality of teamwork and the selected processes)
- List of all the metrics used in the project and evaluation of their usefulness for the project outcomes
- Evaluation about the effectiveness of the team's communication, collaboration, management, and deliverables
- Analysis and synthesis of all measurement data about different criteria at the end of each development phase as well as of the entire project
- Overall project evaluation

The teams must also be ready to provide evidences for their above evaluation or analysis in the final review session with their faculty mentors. In addition, each team member must keep detailed track of their roles and responsibilities throughout the project. They eventually will need to ask themselves the two following questions:

- Have I really learned anything from this Capstone project?
- What advice should I give to future Capstone project teams?

Besides the end product or product prototype, Capstone teams are also required to put together an overall report of their accomplishments, conclusions, recommendations and lessons from the Capstone project. This report must be prepared in the highest quality possible, i.e., it has be clear, complete, well-written, and delivered on time [14]. It will serve as the basis for final assessment as mentioned in the next section.

D. Capstone Project Assessment:

After the final review with faculty mentors, Capstone teams will need to make a presentation to the Board of Capstone Assessment which consists of 5 members including the faculty mentors, the business clients and other DTU instructors. Each team will have a maximum of 30 minutes for their presentation, and the Board of Capstone Assessment will spend at most 1 hour and 30 minutes for Q&A (Questions & Answers) and evaluation of the project outcomes. Questions will be given to specific team members regarding their roles and contribution to the project. Students are expected to handle difficult questions concerning their approach in "selling" their ideas and vision in a real-world business environment [18].

Besides the presentation, Capstone teams are also graded based on the team's final report which reflects all the effort and progress of the Capstone team throughout the project. As mentioned earlier, the final report has to be of good quality. Good reports are the result of great teamwork and process, attention to details, and strong performance of the end product. While students may cheat the audience with a well-written report, it will be difficult for them to disguise ineffective team behavior, lack of progress, and poor quality product in their presentation [18]. The Board of Capstone Assessment will give out the final grades for each team member based on a combined evaluation of the final report, the board presentation and the end product. Specific criteria used in the grading are, as followed:

- Product Quality (40%)
  * New or original ideas, Project Proposal, product features, (perceived) quality of product
- Deployment of Software Development Process (30%)
  * Planning, Requirement Engineering, Architecture & Design, Implementation & Testing
- Teamwork & Team Management (30%)
  * Team organization, meetings, presentations, Project Notebook (for documentation), Weekly Progress Reports, preliminary and final Team and Self Appraisals, Individual Contribution Breakdown

In general, most of the above grading criteria for Capstone projects at Duy Tan adhere to the PjBL standards and requirements in terms of process focus, student centeredness, and required interpersonal and teamwork skills [15][17]. In addition, they also put certain focus on various performance outcomes of the end product. This is to accommodate with the fast changing technical requirements of the IT industry [16].

IV. STUDENT FEEDBACKS & LABOR MARKET RESULTS OF THE PROJECT-BASED LEARNING AND CAPSTONE PROJECT APPLICATION IN IT MAJORS AT DTU

Feedbacks from the students of the Software Engineering, Information Systems, and Network Security programs of Duy Tan University which adopt the new Capstone format together with the PjBL approach were mixed as the students mostly focused on what they had not been able to achieved through the projects [19]. The biggest complaint was that the students had to focus most of their time and effort on the project due to the strict peer-to-peer reviews between the team members [19]. This had a bad effect on the results of other courses they were taking at the

same time. Some suggested that the whole last semester should be for the senior Capstone project only. While it is hard to allow for the registration of only one Capstone course during the last semester given the strict regulations of the Ministry of Education & Training of Vietnam on our curriculum formats, we consider this complaint as a success as the group work format was moving in the right direction of PjBL, requiring full commitment from every student [18][19]. The second major complaint from many Capstone team members was that the PM appeared to be the one who learned the most from the Capstone project because of his or her many interrelated rights and responsibilities [19]. This is indeed a major problem to us as it created a sense of unfairness that the PjBL approach has always tried to promote. On the other hand, for teams, which often rotated the PM role between their members, the project outcomes were usually not very good, possibly due to the extra time and effort spent for the rotation. Some mentors have already suggested that we need more than one major Capstone project so that every student will have the chance to be the PM in at least one project; however, this also will require bigger investment of time, money, and other resources. A better approach would be to restructure our first Capstone project at the end of the second year but this will come at a constraint that our students are not well-equipped with different IT skills and knowledge at that time. The third biggest complaint came from only a number of students, yet, were alarming for those who overrated the benefits of real-world projects. That was the case of students who worked on small components of very large-scale projects in some IT companies: they did not have a big picture of what they were working on, and thus did not realize much benefit of their IT skills and knowledge; most of their efforts were spent on system integration work, instead [19]. As much as we realize how to solve this problem, we also fear that the availability of small- and medium-sized real-world project will become scarce at most IT companies and businesses in the long run.

Besides the complaints and negative feedbacks, we also recognized that the senior Capstone project setting has brought about many positive developments. Our students in Software Engineering and Information Systems have become more aware of the distributed and cloud-computing world around their software [19], and most of their projects had tried to tackle this issue one way or another. As for our Network Security students, even though the nature of their work and thinking mostly has to do with "scripting", they also have become more organized and controlled with their end product [19]. Most security end products or prototypes had attempted to tackle the issues of future extension and platform independence. In addition, there was a huge surge in terms of interest in many projects for wireless and ubiquitous devices [19], which are more than often of small- and medium-sized projects, and thus, can provide an alternative for the mentioned problem of students working on small components of very large-scale projects at real-world IT companies and businesses. Projects on wireless and ubiquitous devices also offer a taste of "start-up" spirit for our students since they can easily sell their products online. Other than that, our students have generally become agile and flexible in their teamwork and interpersonal skills; they also have become very good with their presentation skills, using different graphics and visualization tools; and most importantly, they have become more "open" in their perspective and approach [12][19].

The new Software Engineering/Information Systems/Network Security programs at Duy Tan University, which adopt the PjBL approach with 2 major Capstones, have been well received by the labor market. Compared to the overall placement ratio of 89% of Duy Tan, the ratio of over 95% after only 3 months from graduation for the Software Engineering and Information Systems graduates of DTU is remarkable [12]. Surveys also showed that up to 85% of those graduates were offered jobs by the IT-business clients who hosted their Capstone projects without any need for job application or interview [12]. Most of the IT-business clients have commented that the new batch of graduates from the PjBL programs at DTU is more active and creative in their work.

PLACEMENT RESULTS OF SOFTWARE ENGINEERING/INFORMATION SYSTEMS STUDENTS AT DUY TAN UNIVERSITY

Year	**Traditional Programs**			**PjBL Programs with Capstones**			**Number of Graduates per Year**
	Job Offer before Graduation	*Employment after Graduation (less than 3 month)*	*Employment after Graduation (from 6-12 months)*	*Job Offer before Graduation*	*Employment after Graduation (less than 3 month)*	*Employment after Graduation (from 6-12 months)*	
2009	5.0%	75.4%	87.0%				64
2010	11.3%	88.24%	94.0%				72
2011				38.0%	89.2%	100%	62
2012				59.2%	95.1%	100%	98

(Source: Placement Survey Results, DTU Career Center, 2012)

Note: The restructured Network Security program has not produced any graduates until 3 years from now.

V. CONCLUSION

With the focus placed on learning process improvement and personal skill-set development, the PjBL approach has proved to be a good tool for the restructuring of IT curricula at Duy Tan University, which in turn have their own academic focus on different IT development processes and life cycles. The outcomes, however, would not have been that good without the adoption of large-scale Capstone projects, which concentrate on the learning outcomes and professional practice development. Past adoptions of many small projects in different courses at DTU have appeared to be ineffective for the students' learning curve because students are overloaded with a great amount of schoolwork most of the time. However, there are also certain feedbacks that usually only those who played the PM role benefited the most from this new approach. As a result, future studies should look for ways to design the optimal Design-Implement experiences for every Capstone team member, no matter what role(s) he or she plays.

VI. REFERENCES

[1] National Steering Committee on ICT, Ministry Of Information and Communications, "Information & Communication Technology Vietnam 2011-White book," Vietnam: Information and Communication Publishing, 2011.

[2] VAIP, Vietnam ICT Index 2011, Vietnam: Annual Report of Vietnam Association for Information Processing, 2011

[3] Ngo, T. V., "International experience in using Human Resource Efficiency in Global CyberSoft Vietnam," http://www.globalcybersoft.com: 2011.

[4] Magboo, M. S. A. and Magboo, V. P. C., "Assignment of Real-World Projects: An Economical Method of Building Applications for a University and an Effective Way to Enhance Education of the Students," Journal of Information Technology Education, Volume 2, 2003.

[5] Bédard, D., Lison, C., Dalle, D., Côté, D., and Boutin, N., "Problem-based and Project-based Learning in Engineering and Medicine: Determinants of Students' Engagement and Persistance," Interdisciplinary Journal of Problem-based Learning: Vol. 6: Iss. 2, Art. 8, 2012.

[6] Boondee, V., Kidrakarn, P., Sa-Ngiamvibool, W., "A Learning and Teaching Model using Project-Based Learning on the Web to Promote Cooperative Learning," European Journal of Social Sciences - Volume 21, Number 3 , 2011.

[7] Yam, L. H. S., Rossini, P., "Implementing a Project-Based Learning Approach in an Introductory Property Course," 16th Pacific Rim Real Estate Society Conference, Wellington, New Zealand, January 2010.

[8] Chandrasekaran, S., Stojcevski, A., Littlefair, G., Joordens, M., "Learning through Projects in Engineering Education," SEFI 40th Annual Conference, Greece, September 2012.

[9] So, H-J., Kim, B., "Learning about problem based learning: Student teachers integrating technology, pedagogy and content knowledge," Australasian Journal of Educational Technology 2009, 25(1), 101-116.

[10] Akınoğlu, O. and Tandoğan, R. O., "The Effects of Problem-Based Active Learning in Science Education on Students' Academic Achievement, Attitude and Concept Learning," Eurasia Journal of Mathematics, Science & Technology Education, 2007, 3(1), 71-81

[11] Gülseçen, S. and Kubat, A., "Teaching ICT to Teacher Candidates Using PBL: A Qualitative and Quantitative Evaluation," Educational Technology & Society, 2006, 9 (2), 96-106

[12] DTU Career Center, "Placement Survey Results," DTU Press Release: Da Nang, 2012.

[13] Association for Computing Machinery, "Computing Curricula 2001 - Computer Science," National Science Foundation under Grant No. 0003263: December 2001.

[14] IEEE Computing Society and Association for Computing Machinery, "Software Engineering 2004 – Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering," National Science Foundation under Grant No. 0003263: August 2004.

[15] Ronis, D., "Problem-based Learning for Math & Science," Corwin Press: CA, U.S., 2008.

[16] Marshall, L., "A Comparison of the Core Aspects of the ACM/IEEE Computer Science Curriculum 2013 Strawman Report with the specified Core of CC2011 and CS 2008 Review," Science Education Research Conference, Wroclaw, Poland.

[17] Bender, W. N., "Project-based Learning - Differentiating Instruction for the 21st Century," Corwin Press: CA, U.S., 2012.

[18] Boss, S., Krauss, J., "Reinventing Project-based Learning," International Society for Technology in Education (ISTE): Washington D.C., U.S., 2007.

[19] DTU International School, "Capstone Feedbacks and Evaluations," Da Nang, 2012.