Changing education through ICT in developing countries

M. Georgsen & PO Zander

AALBORG UNIVERSITY PRESS

Changing Education Through ICT in Developing Countries

Marianne Georgsen

Pär-Ola Zander

AALBORG UNIVERSITY PRESS

Changing education through ICT in developing countries
Edited by Marianne Georgsen and Pär-Ola Zander

1. Open Access Edition

Cover layout: Skinstad
Text layout: Søren Cort Hansen
ISBN: 978-87-7112-124-7

Published by Aalborg University Press | forlag.aau.dk

This book is financially funded by eLearning Lab at the department of Communication and Psychology, Aalborg University.

PEER REVIEWED

Part I

ICT in Higher Education

Developing the Qualifications of the ICT Workforce Through Problem-based Learning

Mayela Coto, Universidad Nacional
mcoto@una.ac.cr

Sonia Mora, Universidad Nacional
smora@una.ac.cr

Marianne Lykke, Aalborg University
mlykke@hum.aau.dk

Abstract

The ICT sector has an important role in the development of Costa Rica, and consists of a mix of multinational and local companies. Such development requires a large workforce related to computer science and informatics, with skills in the areas of problem solving, team work, mathematics, business administration, and foreign languages.

While computer engineering have a tradition of deductive teaching which is strong in Costa Rica, studies show that problem-based, inductive teaching is a preferable alternative. This chapter presents a case study where the School of Informatics at the National University of Costa Rica analyzed and tried out how a problem-based learning approach (PBL) may contribute in developing student skills to real-life problem solving. The project used a design-based research methodology and gradually introduced PBL in a sequence of five programming courses with the purpose to examine and evaluate how PBL can be used to develop the desired real-life problem-solving skills.

The results show that the process of implementing PBL is not straightforward. The alignment of current curricula with deductive teaching methods is very strong, and inductive approaches require significant changes in the mindset of faculty and students. The results also show that the PBL approach has the potential to reduce failure rates, but is needed to consider carefully some key issues if we want to help develop in computer engineering students the skills the country needs

Introduction

When we started writing this chapter, we considered the concept of "development". What does it mean, and in terms of what indicators should we describe Costa Rica as a developing country? Because of our backgrounds, values and culture, our general understanding of development is based on a humanistic approach. For us, it is not possible to understand a human on the basis of economics alone. In that sense, our concept of development focuses on people and on providing them with acceptable minimum living conditions.

The Human Development Index (HDI) has the explicit purpose of changing the focus of development from national income accounting to people-centered policies. In this sense, development is considered more than the GNP growth; it is "a process of enlarging people's choices, to live a long and healthy life, to be educated and to have access to resources needed for a decent standard of living" (Nielsen, 2011; UNDP, 1990). Due to this multivariate approach, we have selected the HDI system in order to discuss and reflect on the concept of development in this chapter.

In terms of the HDI, Costa Rica is a developing country, belonging to the group of High Human Development Countries with a HDI of 0.773. The country ranks in the 62th position out of 187 worldwide. In regional terms, this index is above the average of 0.758 for countries in the high human development group and also above the average of 0.741 for countries in Latin America and the Caribbean (UNDP, 2012).

In the last decades the ICT sector has had an important role in the development of the country. At present Costa Rica is the fourth largest exporter of technology worldwide, according to data from the Costa Rican Investment Promotion Agency (CINDE, 2011b). The exports of high-technology products represented 41% of its total exports in 2009. For the period of 2011-2012, Costa Rica ranked #1 in Latin America for the Foreign Direct Investment (FDI) and Technology Transfer components of the Global Competitiveness Index (GCI). Multinational companies such as Intel, Hewlett Packard, Microsoft, Fujitsu, Sykes, Amazon, Oracle, Cisco, AvVenta, Amazon, Convergys, Dell, and IBM have operations and offices in Costa Rica (CINDE Costa Rica, 2011a). In addition, a growing number of local companies have been established to provide ICT products and services worldwide (Mata, Matarrita, & Pinto, 2012).

There are many reasons for the increasing interest in investing in Costa Rica; one of them is the investment that the country has made in education as part of its development strategy, believing firmly in the key role played by this indicator on competitiveness and development (Villalobos & Monge-González, 2011). The Costa Rican government is required to allocate at least 8% of the country's GDP to education, which has brought significant benefits to the country. Through the years, it has allowed the creation of a skilled workforce. Other factors are a reduction of taxes and trade barriers of technological products and a solid foreign trade policy. Both of them have created very good conditions for attracting high-tech foreign investment (FDI) (Villalobos & Monge-González, 2011).

It is clear that the growing number of multinational and local companies in the ICT sector required a large qualified workforce. Costa Rican universities produce highly qualified individuals as well as entrepreneurs especially in high technology areas (CINDE, 2011c). However, currently the country is facing challenges responding to these new demands. These challenges are in terms of the supply of ICT workers (Mata, Matarrita, & Pinto, 2012), and in terms of skills development (Monge& González, 2007; Villalobos &Monge-González, 2011).

According to Mata, Matarrita and Pinto (2012), the growing number of multinational and local companies in the ICT sector has provoked a deficit of ICT skilled workers in particular in careers related to computer science and informatics. Their study on the demand and supply of ICT workers in the period 2007-2009 reveals a tight labor market for this sector. Regarding the qualification of the workforce, Monge and González (2007), relying on a study conducted by Cespedes and González (2002), reveal a low level of satisfaction of the companies with employees' skills in the areas of problem solving, science and mathematics, business administration, and foreign languages. The authors conclude that a better way to respond to the demands of the ICT sector is creating more dynamic curricula and strengthening relationships between universities and companies to jointly establish the priorities for professional education.

The Costa Rican government continues to establish aggressive policies to attract foreign direct investment and to demand to the universities to react quickly and demonstrate that their curricula contribute to graduate professionals with abilities to meet the needs of this growing ICT sector. In particular, they are focusing on careers within computer science, computer engineering, informatics, systems engineering and related areas.

The School of Informatics at the Universidad Nacional has decided to address this challenge by a project that analyzes and tries out how a problem and project-based learning approach (PBL) can contribute in developing student skills in real-life problem solving. This question is investigated together with the challenge of reducing dropout and failure rates, which are, in particular, very high in programming courses. By reducing these rates the school aims at increasing the number of ICT graduates per year. These assumptions about the feasibility and usefulness of problem-based and project-based learning are examined in a case study with the following research questions:

1. How can a problem-based and project-based pedagogical (PBL) approach contribute in computer engineering curricula with the purpose of developing student skills in real-life problem solving, reducing dropout and failures rates?
2. How can this pedagogical approach be implemented in computer engineering curricula with the purpose of developing student skills in real-life problem solving and reducing dropout and failures rates, and does it work?

The case study involves an educational intervention in five programming courses, and it takes place through four semesters. The project has the support of the Office of the Vice-president for Academic Affairs as a pilot project at UNA, and it is expected that its results can be disseminated to other disciplinary areas.

This work presents a summary of the findings of the first two cycles of this intervention. The chapter is organized in eight sections. Section 2 presents a brief description of problem-based and project-based learning and some experiences at the university level. Section 3 introduces the methodology of the study. The analysis of the context and the design of the educational intervention are discussed in section 4, and the results from two cycles of intervention are presented in section 5 and section 6. Section 7 contains reflections on lessons learned and outlines future work, and section 8 presents final conclusions. An early version of the context analysis and initial design solution has previously been published in Spanish in (Coto, Mora & Lykke, 2012).

Problem-based and Project-based Learning: An Inductive Approach to Teaching

In order to effectively develop the new set of skills demanded by the ICT sector, we need to change the teaching. In computer engineering, teaching has traditionally been deductive (Prince & Felder, 2006), meaning that faculty presents a theme through a lecture which sets the general principles, shows illustrative examples, offers students practical work on similar applications, and then tests their ability to do the same kind of reasoning through exams. This approach is very common at Costa Rican universities. However, studies establish that a preferable alternative for the learning process is to begin with real-life questions to be answered, a realistic case study or a complex problem that must be resolved in the real world. When students analyze data and try to solve a problem, they create and see a need to know principles, facts, rules and procedures. This need has a strong impact on motivation for learning, because students can understand the purpose of what they are learning. This approach is known as inductive teaching and learning (Prince & Felder, 2006).

According to Prince and Felder (2007; 2006), problem-based learning and project-based learning, among others, are inductive methods of teaching and learning. They are student-centered, have constructivist principles, involve active learning and promote collaborative learning. Both approaches achieve a higher motivation and greater responsibility in the learning process.

Problem-based learning is promoted when students are faced with a real problem as a starting point for the acquisition and integration of new knowledge. Prieto (2006) points out that this approach can improve the quality of learning in universities in different ways as it encourages skills such as: problem solving, decision-making, teamwork and communication skills. It also fosters the development of information searching competences and research management capacity, because students are motivated to understand and investigate a problem before proposing a solution. On the other hand, a project-based approach is based on the idea that students can develop projects that can be applied in real-life as a starting point for their learning process. This means setting aside the traditional teaching and following an approach in which interdisciplinary activities promote and stimulate collaborative and student-centered learning.

The effectiveness of PBL (problem-based learning) approach has been analyzed in different studies in the university context. Dochy, Segres, Van den Bossche and Gijbels (2003) reviewed 43 empirical studies of the effects of PBL in the acquisition of knowledge and the development of problem solving skills. With regard to the first element, the results showed that students acquire more knowledge in short time if they are using a traditional teaching approach.

However, students who have experienced the PBL approach are capable of retaining the knowledge through a longer period of time. In addition, with regard to the acquisition of skills, the effect of the PBL approach was strongly positive, according to the authors.

In the same vein, Prince (2004) concludes that faculty who adopt PBL does not perceive a significant improvement in the academic performance of students, but possibly obtain a positive effect on their attitude towards learning, habits of study, the retention of knowledge and the capacity to apply it, as well as the development of critical thinking and skills to problem solving, particularly if the experience is accompanied by explicit training on these processes.

Nuutila, Törmä and Malmi (2005) observed a significant decrease in the dropout rate in a study on the application of PBL in initial programming courses. The authors state that, in addition to programming, students acquired skills related to collaborative work, independent study and communication. The research study followed the students in advanced programming courses, and data show that the average score of those coming from the PBL course were slightly better than the others. Similarly, Hamalainen(2004)reports findings from a course of theoretical concepts from computer science and concludes that the dropout and failure rates decrease when students follow a PBL approach compared to a conventional one. According to the author, students generate a greater commitment to a PBL course than to a traditional one.

In this way, research in pedagogical approaches identifies PBL as a helpful approach to solve partially some of the problems that students in computer engineering face. According to Mills and Treagust (2003) other inductive approaches as project-based learning may be more appropriate, because it better reflects the professional behavior of a computer engineer. Kolmos(2004)mentions that the latter approach facilitates the development of competences on project management and collaboration, both of them essential for computer engineers.

The use of projects as part of teaching and learning is well known in the area of computer engineering. Usually, they focus on the application of knowledge and, possibly, on the integration of previously acquired knowledge (Mills & Treagust, 2003). However, we should consider that project-based learning presupposes an integral educational strategy and not just a complement in the learning process. When a curriculum is organized in projects, most of the courses are linked with the development of a specific project. There are few examples of study programmes where the entire curriculum is organized in projects. The best example of a curriculum almost completely based on the PBL approach is Aalborg University in Denmark, where projects and courses related to these make up 75% of the curriculum (Mills & Treagust, 2003). Other universities using projects as the main focus of their engineering curricula include Roskilde University in Denmark; TU Berlin, Dortmund, Bremen and Oldenburg in Germany; TU Delft and Wageningen University in the Netherlands; Monash University and Central Queensland University in Australia; and Olin College in the United States (Prince & Felder, 2006).

To obtain the advantages of both the project-based and problem-based approach, they are sometimes used jointly, meaning that a problem is presented as the starting point of the learning process, while problem solving takes place through a project. This is the approach of Aalborg University in Denmark where students have an active role in the construction of knowledge in order to make them feel more committed to their own learning and to ensure high quality standards. This form is known as the PBL model of Aalborg University (Barge, 2010) or Project-Oriented Problem Pedagogy (POPP). Its basic principles are: problem solving as the point of departure of the learning process; projects as the way to address the problem; and integration of theory and practice. The students have a level of influence on the selection of the problem, and how to organize the project. Students collaborate in groups of three or more students with feedback from peers and faculty (Barge, 2010; Kolmos, 2004).

These characteristics are particular useful for computer related careers. The ability to solve problems is vital in computer engineering, and many of the activities of professionals in computer engineering are framed in the development of projects. Furthermore, ACM (2011) identifies a set of skills that future graduates must have, among them, problem solving, effective communication, effective group work, professional responsibility and the capacity of lifelong learning. In the same vein, Hissey (2000) and Mills and Treagust(2003) point out that professionals in computer engineering should be able to cope with continuing technological changes, work in uncertain situations, respond to diverse and sometimes contradictory needs, and understand the social, legal, ethical and cultural aspects involved in the exercise of their professional practice.

In the present research project, we have designed an educational intervention with the goal to investigate how principles from PBL may contribute, to some extent, to the demands of the Costa Rican ICT sector. We take as our point of departure the premise that PBL can help prepare students for real-life problem-solving, increase intrinsic motivation in students, reduce dropout rates, improve professional qualifications, and, in this extent, contribute to increase the number of graduates who are needed in the ICT sector.

Methodology for Implementing the PBL Principles in the Programming Area

In the project, we have chosen to introduce gradually the PBL principles in a sequence of five programming courses (Introduction to Programming, Programming I, Programming II, Programming III and Programming IV).With this objective in mind, a design-based research methodology was selected because it facilitates a framework for the study, design and evaluation of tools and educational strategies (Design-Based Research Collective, 2003). According to Wang and Hannafin (2005), the goal of the methodology is to improve and evaluate educational practices in real contexts. It is a flexible but systematic process guided by theories and design principles sensitive to the context, and it is based on a participatory process of collaboration between researchers and practitioners.

Design-based research can be seen as a model of five phases (Coto, 2010; Reeves, 2006) where participants are part of a joint and iterative process of analysis, design, development and implementation. The first phase "analysis of the problem" requires researchers and faculty to undertake an analysis of the context, in this case the curriculum programming courses. The second phase "design solutions with a theoretical framework" involves the creation of prototypes of solutions based on the principles of PBL, the learning objectives of the programming courses, student and faculty perspectives, and the review of the literature. The third phase "evaluation and implementation of the proposed solution in practice" refers to an iterative process of implementation and refinement of the initial design principles and the prototype solution. The fourth phase "documentation and reflection on the design principles" refers to the process of retrospective reflection on the design and its results. This process of reflection allows for iteratively refining the design principles as well as documenting the strengths and weaknesses. The last phase "dissemination and adoption in broader contexts" concerns the adoption and sustainability of the innovation process, as well as the dissemination to other contexts at the Universidad Nacional.

The process is guided by theoretical principles and knowledge of the "culture" of the academic context, and it is based on a participatory process of collaboration between researchers from Aalborg University and the School of Informatics, as well as faculty and students of the latter. The intervention is carried out in the area of programming. One of the advantages of design-based research is that it allows continuous monitoring of the changes, analyzing any discrepancies between the design and participant responses, and from this we are able to propose changes to achieve a more efficient approach of faculty and students.

Below, we will present some of the results obtained in the first two phases: analysis of the context, and design of the initial solution. Furthermore, we will present preliminary results of the evaluation and implementation phase.

Phase I: Analysis of the Context

It is considered fundamental to the project to design prototype solutions with probability of success. As Guskey (2002) suggests, a significant change in attitudes and practices of faculty mainly occurs once they have obtained evidence of the improvement of learning in their students. This model supports change as a process based on positive experiences for faculty.

To achieve the design of "successful" solutions, it was necessary to have a good understanding of the context. This entails to analyze the UNA pedagogical model that is the institutional framework for this process, to analyze faculty and students perspectives and their readiness to confront a process of change, and to analyze the actual conditions of the programming courses.

The Pedagogical Model of UNA

Since 2008, the Universidad Nacional has employed a pedagogical model which is based on an understanding of teaching and learning as a social, historical and cultural process that goes beyond a mere transmission of knowledge. Teaching and learning are thus based on (1) the analysis and questioning of the reality; (2) research and practical work concerning the context in which a student and his career develops; (3) the development of skills for innovation and problem solving, (4) the negotiation of conflicts; (5) interdisciplinary group work; and (6) making decisions based on reliable and timely information (Universidad Nacional, 2007).

The UNA pedagogical model does not point out any specific didactic strategy. UNA expects that its application will be enacted through diverse strategies of teaching and learning in accordance with the object of study, its nature and its practice. These general and flexible principles entail some practical difficulties, especially for faculty without pedagogical background. In the daily practice, the interpretation and reification of the pedagogical model rest heavily on the faculty understanding of its principles; on their own experiences of education; on their conceptions of teaching and learning; on their pedagogical knowledge, experience and skills to implement different didactic strategies; on the influence of their disciplinary context and on the negotiation that takes place in classrooms among faculty, students and practice.

In the project, PBL is understood as one possible concretization of the UNA pedagogical model. Both models are aimed at student centered teaching and learning, where the main role of the faculty is the facilitation of the learning process. In both, teaching is understood as a complex and multidirectional process through which knowledge is constructed and shared.

Faculty Perspective

In order to learn the faculty perspective, a workshop was conducted with the goal to identify the challenges of working with PBL as pedagogical approach. Eight academics from different disciplinary areas (programming, databases and information systems) participated in the workshop. The academics were divided into three groups. Each group was asked to brainstorm on goals that faculty wants to achieve, barriers that can obstruct goals, and on actions that should be implemented to reduce the gap between goals and barriers. The results were grouped by similarities and are shown in Table 1.

Table 1. Faculty perspectives about change towards POPP

BARRIERS
A rigid curriculum administration
Little possibility of collaborative work among professors
Students have low motivation
Low motivation at work
The pedagogy is not conceptualized as a fundamental vehicle to improve students' learning.
GOALS
Improve the organizational climate
Training professors in formulating and solving problems
Have a curriculum aligned with society's needs

Promote a working environment more geared to common goals, collaborative work and the pursuit of excellence
Improve the link University - industry
ACTIONS
A gradual strategy for change that will be able to anticipate future requirements
Design a continuous pedagogical training that allows the development of professor relevant teaching skills
Improve the curriculum

From a faculty perspective, the intervention should consider a rigid curricular administration and the existence of a group of academics who do not fully believe in pedagogy as a vehicle to improve faculty performance and student learning. This leads to promoting a gradual change with successful experiences to motivate professors to introduce small changes and to share their knowledge and experience, in such a way that contributes to improving the communication and learning processes among them. It is also important to monitor students' learning in order to obtain information that leads to an improvement in the curriculum.

As a complement to the previous information, a questionnaire was designed and administered to 14 professors from all the disciplinary areas and curricula levels. The questionnaire assesses professors' perception on the decision level of the students, project development, group work, assessment and motivation, which are all fundamental principles of the POPP approach. The following percentages are calculated on the basis of the number of professors who responded to each question. The response rate is indicated individually for each question below:

a. *With respect to the decision level of students*: 100% of professors provide in their courses a detailed definition of problems and projects. 62% of them argue that students are not mature enough to choose or define a well-working problem to solve. As such, students have almost no level of influence on this matter. On the other hand, 100% of professors say that the problems/projects they provide in classroom, always require research from the students. This can be a bit contradictory when 90% at the same time asserts that they provide everything students need to solve the project in their courses.

b. *With regard to project development*: 64% of professors consider that they offer real-life projects to the students. For 46% of them, the most important element in the selection of a project is the amount of content that it covers. However, 100% indicates that they seek interesting projects for students. In the time frame of a course, 43% of them prefer to assign several short projects instead of just one longer project. 69% of professors claim that the most important component of the project evaluation is that the programming works efficiently. However 93% of them consider the student's learning process as a component of the assessment. 79% of the professors give students feedback on their technical mistakes, and 43% of the professors express that students usually show little interest and commitment to solving projects.

c. *With regard to group work*: 100% of the professors prefer to work with small groups (maximum three students). 79% of the professors state that the working groups resolve conflicts that might arise by themselves.
d. *With regard to assessment*: only 7% of professors think that exams are the best way to evaluate student learning, and 79% uses self- and co-evaluation strategies to assess projects.

e. *With regard to motivation*: 86% of professors state that students need motivation to learn. 93% also believes that teaching strategies or pedagogical approaches can contribute to increase motivation and reduce failure and dropout rates. This point seems contradictory regarding the issue of not considering pedagogy as an important means to improve students' learning, which some professors expressed in the previous workshop.

The previous results show that there are many aspects where professors need to change their mindset in order to be ready to face an approach as POPP. They need to give students more autonomy and a high level of influence on the process of developing problems and projects. They also need to nurtue the research component of the learning process. These aspects may in turn contribute to increase student commitment. Professors also need to focus more on the learning process and provide feedback on this aspect and not only on the technical part and the content of projects.

Student Perspective

In order to learn the student perspective, a questionnaire composed of open and closed questions was designed. The goal was to assess students' preparation towards a curriculum organized under a POPP approach. The basic principles of the selected learning approach were taken into account when designing the questionnaire. The total population was 118 students (five groups of the EIF206 Programming III course), and the response rate was 58%. The main results are presented below:

a. Regarding problem solving: 76% of the students confirm that they recognize that the use of problems is a learning strategy in the curricula. 79% state that they like to solve problems or projects with known conditions, while only 57% say that they like the challenge of exploring unknown situations to them. As regard the extent of the authenticity of the problems only 55 students answered. Of these, 84% consider problems as good examples of real-life problems, 4% think they are completely fictitious, and 12% see them as authentic real problems. For design purposes, this suggests that we should provide students with experiences that foster a more positive attitude towards unknown situations that demand a more complex cognitive activity. The use of real-life situations closer to their future work and professional activity should also be reinforced.

b. With respect to the level of influence desired and exercised by students: Out of 52 students, 98% say that they would like to have an influence on the problems they solve. Additionally, only 58% state to have some level of influence on the problems posed by the professors, while 42% say they have no power in this regard. Furthermore, 75% feel able to decide which problems and projects can be solved. These data suggest that students are more ready to be autonomous compared to what the professors believe. This students' attitude is a key aspect that we need to foster, no matter they are or are not prepared. In this regard, we should not provide students with fully defined projects, but offer some level of infleunec on problem development, type of project, how to present the results, and so on.

c. With respect to experiences in real-life projects and the preparation obtained for approaching them: In the programming courses, students have to complete two to three projects a year. When asked whether they feel able to conduct real projects in a company or institution, 65% of the students express that they feel unable to deal with this type of work. They feel that they lack skills. In addition, they express that the kind of hypothetical projects carried-out in the classroom have not prepared them to face real ones. Whereas solving real-life situations is not only a motivating factor for students, but also trains them for future employment, it seems necessary to provide exemplary projects that enable them to exercise what will be demanded of them in their professional life.

d. With regard to group work: 72% of the students believe that teamwork is an appropriate learning strategy. However, 28% of them do not consider it suitable due to the conflicts that arise, such as inequality in task division and incompatibility of schedules and personalities. When conflicts arise, 58% of students say they are able to solve the conflicts by themselves, 36% require professor assistance, and 6% never solve the problems. Regarding supervision, 63% say they receive little or no professor supervision in the development of the project. The data suggest that it is important to keep group work as a strategy for knowledge sharing, but that students should be prepared to deal appropriately with conflicts that may arise, while professors need to be trained to provide the necessary and sufficient supervision and support.

e. With regard to responsibility for the learning process: 28% of students think that professors should take responsibility for the learning process, and just 41% believe that it is a shared responsibility. These data suggest that we should provide opportunities where students exercise greater control and learn to take responsibility of their learning process and to be more autonomous individuals.

In general, the above findings point to the relevance of moving the processes of teaching and learning towards a POPP approach, because it can offer students greater possibilities to make decisions when working with problem solving, and how to address the problem, and as consequence increase their motivation in the learning process and take responsibility for this process. Furthermore, the approach allows the integration of knowledge from diverse discipline areas and more contact with real-life problems. All of these are desired competences for computer graduates in order to contribute to the society development in an effective way.

Furthermore, the analysis reveals several key issues that are important for the success and contribution of PBL: a. Problems that are closely related to future work activities; b. Open problems that the students develop through the project work; c. Exemplary projects that allow students to exercise; d. Preparation of students to conflict-solving, and e. Shared responsibility between faculty and students for the learning process. All of these are desired skills for computer graduates in order to contribute to the Costa Rican development in an effective way.

Considerations of Programming Courses

The five courses in the area of programming are an essential part of the curriculum, in part because of the skills that students develop and the interest that programming generates. This is particularly true of the course Introduction to Programming, which is the first contact that students have with the disciplinary area. In turn, programming together with the mathematics courses are the courses with the highest failure and dropout rates.

According to data of the Informatics School, the average failure rate (including dropouts, because no individual records exist) of programming courses for the period between 2006 and 2011, can be seen in Table 2.

Table 2. Failure rates of programming courses

Programming course	Failure rate
Introduction to Programming	51,33 %
Programming I	35,93 %
Programming II	30,24 %
Programming III	20,35 %
Programming IV	16,72 %

As it can be seen from these results, the failure rate is higher at the initial and earlier stage of the education. The professors have made a similar observation concerning dropouts. This points to multiple explanations; one of them is that some of the students who start studying Computer Engineering discover after a while that they have made the wrong choice of study. Another explanation relates to the change of logic reasoning that is required for students to face the algorithmic resolution of problems. These indicators pose an additional challenge for the project. It is expected that the POPP approach can contribute to reduce failure and dropout rates, especially in the early courses.

The next section will try out the identified PBL principles: Problems related to future work activities, open problems, exemplary projects, conflict solving, and shared responsibility between faculty and students, and evaluated how they work.

Phase II: An Initial Design Solution

The initial phase of the project, proposes an educational intervention that takes into account two important aspects: (1) a process of faculty training and (2) the gradual introduction of some of the principles of the POPP in the programming courses through the definition and solution of problems. At this point, it is very important to clarify and explain that professors in the programming area have their own mindset about programming and teaching practices, and that the research project does not have any mandatory approach. It means that the professors are not required to make changes, but the project seeks to convince them to implement the proposed changes. As such, not all professors in a specific course adopt the approach in the same manner and to the same extent. This is a great challenges of the project.

Faculty Professional Development

In accordance with the results of the previous stage, it is clear that the curriculum innovation process needs to be linked to a strategy for faculty professional development that allows them to: (1) strengthen their skills in designing new learning environments, and (2) develop the necessary skills to manage a curriculum focused on problem solving and organized in projects.

This process of teacher training aims to strengthen the group of programming professors, and has a goal to establish gradually the basis for the formation of a community of practice as suggested by Wenger (1998). In this sense, teacher learning is conceptualized as result of a collaborative effort in which they receive support from their colleagues, external experts, researchers and administration (Rhodes & Beneicke, 2002; Schlager & Fusco, 2004). The pedagogical intervention aims for a process of collective reflection and continuous refinement in order to solve the challenges of teaching practice and enable the positive transfer of faculty learning into the classroom (Wing Lai, Pratt, Anderson, & Stigter, 2006).

The professional training involves formal and informal learning activities, where professors "learn by sharing". Short training processes each semester are suggested, with the goal of gradually developing skills that allow professors to design learning environments in accordance with the POPP principles, and provide them with opportunities to experiment in classroom. This opportunity of "experimenting" by introducing small changes in the classroom gives professor's the possibility to perceive the classroom as a research area.

Consequently, introducing the POPP principles in programming courses is a process adjusted to the needs and possibilities of professors, the students' maturity and the nature of each of the courses. This will be discussed in the next section.

Introducing the Principles of POPP

POPP incorporates a series of didactic principles as a basis for the design of learning environments (Dirckinck-Holmfeld, 2002; Graaff & Kolmos, 2003; Kolmos, 2004):

- Formulation and investigation of exemplary problems. In order to understand and solve a problem, the students have to go through systematic stages: preliminary investigation, formulation, theoretical and methodological considerations, and experimentation and reflection. The problem is thus considered the starting point and learning is organized around this.
- Participants control. The learning process is led by students in collaboration with the supervisor who guides and supports them in the process of defining and formulating the problem. The goal is to link the problem to their experiences and interests, hereby generating a higher motivation towards learning.
- Interdisciplinary learning. Problems may extend beyond the known disciplinary framework, which should be considered carefully by the professors when planning the teaching process.
- Joint learning and action projects. Learning is conceived as a social act that is carried out through dialogue, communication and collaboration in mixed groups. Often, the projects are carried out in collaboration with companies and public institutions.

For simplicity of understanding and application, the above principles are located in a continuum where the student is gradually gaining a higher level of decision-making and the professors are taking the role of facilitator (see table 3).

Table 3. Introduction levels of POPP in programming courses

Principles	Level 1	Level 2	Level 3
Problem solving	Emphasis on reproduction of methods Professor provides problem statement	Exemplary problems Professor provides problem statement	Exemplary and ill-structured problems Student identifies problems or need
Learner autonomy	Professor provides the specific knowledge and skills to learn	Professor identifies learning needs and learning objectives	Student identifies and establishes content acquisition and goals based on the defined need
Group work	Task divisions between students	Collaborative learning Strategies for conflict management	Collaborative learning Effective leadership

Principles	Level 1	Level 2	Level 3
Integration of theory / practice	Knowledge of theories and methods	Identification and selection of theories and methods	Research on theories and methods
Scaffolding and feedback	At the beginning and end of the project	At each stage of the project	Continuous and integrated in the classroom
Relationship with industry	Hypothetical problems based on real-life solutions	Contextualized industry real problems	Coordination with industry and its needs
Reflection	Minimal self-reflection on learning Reflection on final products	Learner reflection on theories and methods Self-reflection on learning process	Ongoing learner self-reflection Learner reflection on decisions, theories and methods, relationship with industry and the learning process
Evaluation	Professor defines assessments Students provide justification of solutions	Self-assessment, peer assessment Some flexibility in assessments based on student directions	Self-assessment on performance Peer assessment Internalization of feedback and achievement level

We have proposed the table above in response to the analysis of the context, which show evidences that students need to accept the responsibility of learning gradually, and that they also need to be trained in how to deal with learning and conflicts in group work. In a similar way, professors need to accept and learn how to effectively apply the POPP principles in their courses. Many of the courses in the programming area are currently at level 1, where disciplinary contents play a central role, and the professor almost has complete control of the learning process. In addition, reflection is not a common activity in programming courses, and learning is focused and evaluated by final products (programmed projects). At this level, the learning process is not an explicit goal. Thus, a strategy arises that allows to build and evaluate successful experiences in two years (Guskey, 2002), and in addition to move from the current level 1 towards the desired level 3 in the proposed courses.

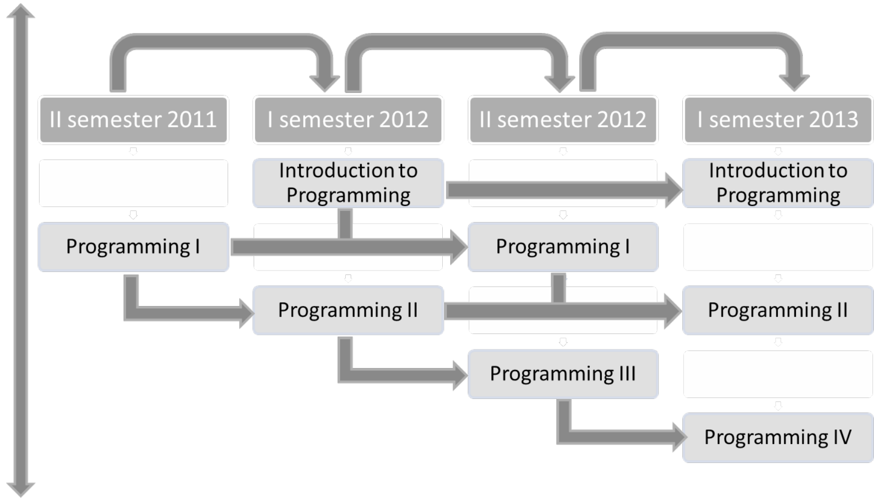

Figure 1. Gradual inclusion process of the POPP principles

Figure 1 schematizes the gradual process of incorporating the PBL principles in the programming courses. As can be seen in Figure 1, it is a continuous process of refinement where the results obtained at the level of a cycle have an impact on the next cycle, and these results have implications for the design carried out in the next course in the sequence. In this paper, we document the first two periods. In 2011, the intervention, in its exploratory approach, took place only in the Programming I course. In the first semester of 2012, the intervention was carried-out in the Introduction to Programming and Programming II courses.

The Problem: The Core of the Design

In inductive teaching, the problem serves as the stimulus for learning. It provides the context and the opportunity to learn, to investigate theories and concepts, and to apply them to real-life situations. In the case of computer engineering, and in particular in the area of programming, problem solving is operationalized through the formulation, implementation and presentation of a programming project, where the final product in most cases is a computer program that to some extent addresses the identified problems. In this way "a good problem" can articulate all the other principles of the POPP approach, and the project becomes the vehicle through which students seek viable solutions to solve them.

In this context, the design of this educational intervention focuses on generating, a new culture towards the resolution of problems, among students as well as professors. This is accomplished through working with the group of programming professors in the development and implementation of learning objectives, types of problems and disciplinary content.

- Learning objectives: The problem is a vehicle through which students gain knowledge and acquire the desired skills in the course (Prieto, 2006), and as such it should have a close relationship with the learning objectives of the programming courses and lead students to investigate and apply the desired knowledge about paradigms, languages, methods, programming best practices, and so on. According to the principle of interdisciplinary learning, the problem must include content from other courses pushing the students to explore links between different areas of computer engineering.

- Types of problems: Problems should be ill-structured, open, and complex; have ambiguous meaning and unknown elements; be open for various solutions; and integrate more than one discipline. Students ought to have the need to analyze and investigate. In this regard, the programming professors ought to move towards projects that involve problems with these characteristics, and thus promote the active construction of knowledge (Graaff & Kolmos, 2003). This leads to promoting a change in culture, for both professors and students, because under the current conditions, an "open project" can be perceived by students as lack of knowledge or effort from professors, and as such it might have a negative impact on the assessment of faculty performance.
- Content of the problems: Programming projects should formulate problems on real situations covering the topics of the courses and exercise what students have to do in their professional practice. This implies that problems should be exemplary and also reflect the level of knowledge and interests of students.
- Way of working: On programming projects, it is common that students divide the project work and at the end put all the parts together. Hence, it is necessary to design projects that create positive interdependencies between students, meaning that they must depend on each other to achieve the common goal. It requires a training process that allows students to develop the skills necessary for the group to function effectively, such as leadership, critical thinking and conflict resolution.

A key factor of the proposed intervention is the ability to develop or select problems with the above stated characteristics. Ideally this process should be a team effort that gives unity to the sequence of five programming courses and ensures incremental growth in the learning objectives. It is intended that as a group, professors can develop a set of projects for each of the courses, and students can select from a pool of these. This will ensure that the project meets the objectives and content of the courses and develops the desired skills of students. In addition, students will be more motivated because they may choose a problem that is meaningful for them. We also expect that this process of collaborative construction of projects among professors will contribute to group cohesion and will be the starting point for creating a community of practice.

In the next section, the preliminary results of two cycles of intervention are presented.

Phase III: Implementation, Evaluation and Refinement

The implementation, evaluation and refinement phase is an ongoing process that began in July 2011 and is planned to finish in July 2013. Until now, two interventions have been carried out. The first one, that took place in the second semester of 2011, is considered as an exploratory intervention because at the same time professors were learning about the approach. In that sense, the obtained data helped us to refine the proposed design. The second intervention took place in the first semester 2012 and was carried out in two courses (Introduction to Programming and Programming II). The professional development of professors was a parallel and complementary activity. This section briefly reports on this process and on the first intervention, and focuses on the data obtained from the second intervention.

Faculty Professional Development

As part of faculty professional development, several formal and informal learning activities have been carried out with the purpose to prepare them to face a POPP approach. The main formal activity was an 8-week online course taught by Aalborg University. Throughout the course, the participant professors analyzed and discussed different approaches to curriculum design with emphasis on the AAU PBL approach. The experience was rich in ideas and reflections on personal and organizational barriers and challenges, but it also illustrated the scarce time that professors have for this kind of processes. In spite of the will to learn, only 8 out of 1 professors who began the process participated regularly in the learning activities.

In addition to the online course, two lectures were given by specialists from AAU, and a number of other activities such as workshops and meetings were organized. These activities aimed to create awareness among participants about the importance of incorporating into the curricula some of the principles of POPP. They were also an opportunity for professors to share their perspective on these principles and how to implement them in their courses. Moreover, they were vital to foster collaboration and build partnerships among the programming professors. From this respect, a group of enthusiastic and engaged professors participated in weekly sessions, where experienced professors presented disciplinary topics to be discussed, solved problems, discussed general teaching strategies, while novice professors improved their disciplinary knowledge in general. In addition, the novice professors shared their willingness to make changes in teaching practice with their experienced colleagues. So far, about 50% of the professors belonging to the programming professors are participating, but the goal is to increase this number, and gradually form a solid and stable community of practice.

The First Intervention

As shown in Figure 1, the first intervention took place in the Programming I course in the second period of 2011. It was carried out in 6 groups of students, for a total of 150 students and 5 professors. Because it is considered as an exploratory intervention, the changes were mainly focused on how to address the projects. In previous years, students were to develop two to three small projects, but it was considered that a single project consisting of three parts was a better alternative to integrate all of the course content and to facilitate student learning. We found that:

- Students feel more comfortable developing a single project. They feel that it is easier for them to understand and acquire the skills needed in the course.
- Students argue that they do not receive timely feedback from faculty, because when they get the comments they no longer have enough time to correct their mistakes. In this sense, faculty also recognized the need to improve the feedback process. However, because the groups have about 25 students, and they are used to work in pairs, it takes long time to give feedback to 13-15 groups.
- When conflicts arise, students tend to dissolve the groups. They are not well prepared to resolve conflicts.

Based on these findings, on the next semester we encouraged faculty to make the following changes:

1. In order to improve the feedback process: (a) to split the project into smaller parts, thus students will have more frequent and timely comments; and (b) to increase the number of group members, thus in groups of four members, faculty will only attend between 5 to 6 groups instead of 13 to 15.
2. In order to promote autonomy: to present open and exemplary problems, that allows students to investigate and develop them through the project work.
3. In order to develop team work skills, increase the number of group members, thus students have more opportunities to learn how to share responsibilities and to solve conflicts.
4. In order to develop the shared responsibility in the learning process, to implement self-assessment and peer-assessment strategies in the group work.

The Second Intervention

The second intervention took place in the first semester of 2012. It was carried out in two courses: Introduction to Programming and Programming II, and about 350 students and 8 faculty members participated. The interventions in both courses are briefly described, and the results are presented below.

The course Introduction to Programming is the first course in the discipline (prior to the Programming I course). It introduces the principles of object-oriented programming with an emphasis on problem solving through the development of algorithms. In general, the course does not involve a project. The proposed change in this course was the introduction of a small project with an open definition to force students to do research and make decisions. In addition, students were asked to form groups of 4 to 5 members and to use self-assessment and peer-assessment strategies in order to promote collaborative work. The group also had to make an oral presentation of their solution in order to help them to develop oral communication skills.

Because the student's knowledge about programming was very limited at this stage, the faculty had defined the problem (about bacteria) as the basis of the project, which was to be modeled from the point of view of the object-oriented paradigm. In order to solve the problem, students had to choose one particular bacteria, research the behavior of it, , and model it under the paradigm. The students could decide themselves whether to implement the solution in a programming language.

The Programming II course is the third one in the sequence of programming courses. To some extent, this course is a prolongation of the Programming I course, and as such the proposed changes were related to the use of one single programming project in five smaller stages with one weekly feedback from faculty. To address the project, the students were formed in pairs. Even though we tried to convince the faculty to form larger groups, they continued to argue that the only way to check students' learning and ensure equal and fair task divisions, is by working in pairs. Also, in this course, the faculty decided to describe the project in a very detailed way, and it was related with a fictitious problem based on a real-life situation.

In order to evaluate and understand the experience, a focus group was conducted with the faculty members, and a questionnaire was designed and administered to the students. In the following we discuss the findings and results concerning how the implemented PBL principles worked in practice.

There were nine participants in the focus group, six (out of nine) from the Introduction to Programming course, and three (out of five) from the Programming II course. In both courses, faculty was satisfied with the changes and stated that the results exceeded their expectations.

In the case of the Introduction to Programming course, they claimed that students demonstrated an application of knowledge about the object-oriented paradigm in modeling the problem, and additionally investigated and managed to adequately justify its selection. This showed how students could apply theory to real-life problem-solving. With regard to group work, the students in some cases considered task divisions to be unbalanced, but overall the groups reached agreement on their performance. The self-assessment and peer-assessment strategies were an adequate tool to increase students' engagement with the group work.

In the case of the Programming II course, they expressed that overall the experience was positive, and that single projects allowed students to achieve the course learning objectives better. They reported an improvement in the course approval rate and in motivation of students. However, they expressed difficulties in giving continuous and timely feedback to group work, mainly due to the larger number of project stages. In addition, faculty claimed not to have sufficient time in class for feedback, forcing them to use a lot of extra-class time for this process. They also mentioned problems of coordination between the faculty who taught the course, which caused that not all groups achieved the learning objectives in the same way.

To understand the student perspective of the Introduction to Programming (IP) course, a questionnaire with 15 questions was developed and applied in six groups. We obtained 124 complete questionnaires out of 150 possible. For the Programming II (PII) course, the questionnaire comprised 19 questions, and was administered to five out of eight groups; 98 complete questionnaires were collected out of 125 possible. Both questionnaires were administered at the end of the semester. Both questionnaires were analyzed from the point of view of the PBL principles that the project aims to foster.

Table 4 presents these data. The percentage indicates the rate of students who agree with the statement. It is important to note that there is no intention to compare the course data in each of the aspects evaluated, because the modifications implemented in each of them were of a different nature, as well as were the learning objectives of each course. This also entails that in the course of Programming II more aspects were studied compared to the Introduction to Programming course.

TABLE 4. Data from the second intervention, 1st semester 2012

Principles	Introduction to Programming (IP) 124 students	Programming II (PII) 98 students
Project formulation		
The project is closely related to course learning objectives	88%	93%
The project managed to integrate theoretical knowledge with practice	85%	90%
The project description was sufficient	68%	76%
I prefer to make several shorter projects than a single large project	------	52%
To split the project in smaller stages is better for my learning process	------	86%
Preparation to face real-life problems and projects		
The project allows the integration of course content with real-life situations	85%	90%
Developing the project prepares me to face a real project in a company	------	85%
Professor feedback		
Professors give timely feedback at each stage	------	71%
The feedback given by professors allows to improve the project solution	------	73%
Level of influence		
I would like to have a higher level of decision-making on the type of problem to be solved in the project	58%	67%
Group work		

Principles	Introduction to Programming (IP) 124 students	Programming II (PII) 98 students
It is better to work in smaller groups (maximum 3) than in larger groups (4-5 people)	43%	79%
Group work is always contentious and difficult to manage	17%	35%
I like group work more than individual work	48%	------
Motivation		
Developing a project is an activity that motivates me	71&	88%
I like to work in projects in the courses	51%	63%
I prefer to work in projects rather than present exams	67%	71%
Assessment		
The criteria used to evaluate the project are adequate	75%	69%
The weight of the project within the course evaluation scheme is proportional to the time and effort I invested in it	56%	56%
Overall perspective		
The course met my learning expectations	81%	85%

From the above information, we can highlight some results:

1. Concerning the formulation of the project, the results show that students believe that both projects required integration of theoretical knowledge with practice, and that they were closely related to the objectives of course. Students from the IP course expressed they had to research more, and in a lower percentage they agree that the project description was enough. These data are consistent with the fact that they received an open problem in contrast with the project specification level in the PII course.
2. Regarding readiness and preparation to face real-life projects, a high percentage (85%) of PII students affirm that the development of the project is good training for them with regards to their future professional work. There was no intention to achieve this goal in the IP course, but it is interesting to compare this data with the data obtained in the questionnaire administered to students from the Programming III course (see Phase I: Analysis of the context). In that case, 65% of students expressed that they do not have the skills to face real projects in a company. One plausible explanation is that while the more "real" the problem that students face in the courses, the more prepared they feel for their future professional practice. This reaffirms the need to offer exemplary projects that enable the students

to exercise what they will be doing frequently in their professional life.

3. With respect to the feedback from faculty, data still show some problems in the way they address this process. As we have seen, faculty acknowledged this problem, but do not have the organizational structure and institutional support to spend more time on supporting group work.
4. With regard to the decision level, it seems that the more advanced the students are, the higher level of influence they want. Data from the questionnaire administered to students in the course of Programming III (see Phase I: Analysis of the context) support this assertion, where 98% of students said they would like to have more influence on the problems they solve. However, it is also important to note the high percentage of faculty who think students are not prepared for a higher level of decision-making. This is another challenge for this research project. It takes time and effort to convince faculty that they must gradually provide more autonomy to students.
5. With regard to group work, it is interesting to note the difference between the two courses. While just 43% of IP students prefer to work in smaller groups, this percentage is much higher in PII. An explanation could be that the IP students have had the opportunity to experience working in larger groups while those of PII have not. Moreover, although IP students worked in larger groups, these students reported less conflict in group work. This supports that working in large groups could be a productive way to balance group work and faculty staff efforts to attend the groups.
6. Regarding motivation, the values of the PII course are higher than those of the IP course. It needs to be considered that the first students have more experience working on projects. In addition, the problem that they solved in the course could be considered more interesting from a programming point of view than the problem assigned to IP students. With regard to assessment, is important to note than just 56% of students from both courses considered that the weight of the project within the course evaluation scheme is proportional to the time and effort they invested in it. This aspect is something that faculty have to reflect on.

Lessons Learned and Future Work

As it was mentioned before, after the exploratory phase in the second semester of 2011, the intervention took place in two courses during the first semester of 2012. From the preliminary results, the data show that in both courses, the failure rate was lower with respect to the average failure rate from 2006 to 2010 (see Table 2). In the case of the Introduction to Programming (IP) course the failure rate decreased from 51.33% to 46.9%; and, in the case of the Programming II (PI) course, it decreased from 30.24% to 17.8%. However, after just one cycle of intervention, it is not possible to assume that the positive result respond completely to the introduced changes. It is necessary to follow with the intervention during a longer period in order to have more clear results.

Regarding the research question 1, the study reveals several key issues that are important for the success and contribution of PBL to computer engineering curricula, mainly in the programming area which is the focus of this study:

a. Problems to solve in the programming courses should be closely related to future professional work activities.
b. Problems should be exemplary and open in order to force students to formulate, investigate and develop them through the project work.
c. To work effectively in group work, students must be prepared to take and share responsibilities and to deal with conflicts.
d. The shared responsibility between faculty and students for the learning process needs to be promoted since the first courses.

All of these are desired skills for computer graduates in order to contribute to the Costa Rican development in an effective way.

In relation with the research question 2, the study revealed the following important key points concerning implementation and use of PBL:

Student perspective:

a. Clear integration of theory and practice, students should investigate about real-life situations and propose solutions to them using the skills acquired through the courses.
b. Student should lead the development of projects.
c. A close collaboration with and frequent supervision from faculty.
d. Autonomy should be gradually fostered.
e. Group size could be larger (than just pairs) to promote students skills in group work and to balance faculty effort in giving feedback.

Faculty perspective:

a. Importance of an iterative approach to curriculum development
b. Preparation of faculty to principles of PBL: Supervision and feedback, student autonomy, new learning strategies, and development and use of realistic problems
c. Group work among faculty members is important for the development of the programming area.

The results also show that the process of developing and implementing PBL teaching methodologies is very demanding, time consuming and sometimes frustrating. While most faculty participants expressed they are open to change, they still maintain control in classroom, and many of them are concerned about giving students more control over some aspects of their learning process. In addition, they still believe that covering a vast amount of content is paramount, and as such, the projects are often aimed at covering as much content as possible. This prevents them to propose more real-life projects. To some extent, students seem to be better prepared than faculty to cope with the changes; they ask for greater autonomy, want to work with real problems, and expect continuous and timely feedback from faculty.

As consequence of those results, we aim for the next cycles of the intervention, to continue moving programming courses from level 1 to 2 and 3 (Table 3). We will foster changes aligned with the PBL principles. In particular we aim to:

- Improve faculty' mechanism of feedback: it is necessary to find means of providing frequent and timely feedback, and also to provide more formative feedback.
- Promote students' autonomy: increasing the students' level of influence on assignments and activities. This may contribute to improve student motivation, self-confidence, responsibility for own decisions, and enthusiasm for learning.
- Focus on skills development and not only on content: faculty should be aware of learning strategies that promote deep learning, lifelong learning, and self-confident autonomous learners.
- Solving real problems: projects should focus on solving exemplary problems that allow students to do the work they will need to do in their professional practice.

It is also considered very important to the success of the project to continue fostering the formation of a community of practice among the faculty of the programming area. We envision that the community will encourage knowledge sharing, learning and change of teaching practices. This is an important issue, since Costa Rica is a small country. Many of the faculty at the School of Informatics works in other universities as well. Working as a community of practice, they will be able to create a common identity and engage with the PBL approach.

Conclusions

Costa Rica as developing country aspires to create the conditions to move towards a developed country, and one of the key elements to achieve this is the human resource. The long tradition of investment in education has produced valuable results. One indicator is the growing number of ICT companies that have established themselves in Costa Rica in the last 10 years. The ICT sector is increasing its impact on the development of the country. If Costa Rica decides to continue promoting policies to attract foreign direct investment, it is clear that universities are the ones that need to provide, at least partially, the needed qualified workforce.

The multinational and national companies in the Costa Rican ICT sector have specific needs, of which many are related to soft skills such analytical skills, reflection skills, and collaboration skills. All these skills are similar to the skills this research project aims to promote in students. In this sense, promoting these skills will contribute to making UNA's graduates better prepared to face industry demands, and in turn contribute to the development of the country.

This project, in spite of its local scope, can be considered a small step in the direction of providing more and better human resources in the Costa Rican ICT sector. In this respect, it is a contribution to the development of the country. The results obtained so far from the present study show that the change process and use of PBL principles is not straightforward. The current alignment of the curricula with deductive teaching methods is very strong, and inductive approaches require significant changes in the mindset of faculty and students. However, it is clear that universities can generate initiatives leading to promote changes, as is the case of the ICT graduates and their impact on the development of the ICT sector.

From the results of this study, we learned that PBL can contribute to develop student skills in real-life problem solving through key elements such as: a. Problems that are closely related to future work activities; b. Open problems that the students develop through the project work; c. Exemplary projects that allow students to exercise; d. Preparation of students to conflict-solving, and e. Shared responsibility between faculty and students for the learning process.

We also learned that the implementation of the approach has the potential to reducing failures rates, but it needs to consider students and faculty perspective. The approach has to clearly integrate theory and practice; to promote students contact with real-life situations and to promote autonomy in a gradual way. It also needs an iterative approach to curriculum development and has to prepare faculty to effectively integrate the PBL principles in the curricula. Only considering these elements the approach could contribute to develop in the computer engineering students the skills that the country requires.

References

ACM. (2011). *Computer Science Curricula 2013.* http://ai.stanford.edu/users/sahami/CS2013/.

Barge, S. (2010). *Principles of problem and project basic learning. The Aalborg PBL Model.* Aalborg: Aalborg University.

CINDE. (2011a). *CINDE Costa Rica*. http://www.cinde.org/

CINDE. (2011b). *CINDE Costa Rica*. http://www.cinde.org/en/services-sector/232-digital-technologies

CINDE. (2011c). *Education overview.* http://www.cinde.org/en/workforce.

Coto, M. (2010). *Designing for Change in University Teaching Practices. The case of UNAgora: A community of Practice Approach to Facilitate University Teacher Professional Development in ICT and Project-Oriented Problem Pedagogy.* Aalborg Universitet, Institut for Kommunikation, Aalborg.

Coto, M., Mora, S. & Lykke, M. (2012). Design considerations for introducing PBL in computing engineering. In *Informatica (CLEI), 2012 XXXVIII Conferencia Latinamericana En.* 1-10.

Design-Based Research Collective. (2003). Design-Based Research: An Emerging Paradigm for Educational Inquiry. *Educational Researcher*, *32*(1), 5-8.

Dirckinck-Holmfeld, L. (2002). Designing Virtual Learning Environments Based on Problem Oriented Project Pedagogy. In L. Dirckinck-Holmfeld, & B. Fibiger, *Learning in Virtual Environments* (págs. 31-54). Frederiksberg: Samfundslitteratur Press.

Dochy, F., Segres, M., Van den Bossche, P., & Gijbels, D. (2003). Effects of Problem-Based Learning: A Meta-Analysis. *Learning and Instruction*, *3*, 533-568.

Graaff, E., & Kolmos, A. (2003). Characteristic of problem-based learning. *International Journal of Engineering Education*, *19*(5).

Guskey, T. (2002). Professional Development and Teacher Change. *Teachers and Teaching: theory and practice*, *8*(3/4), 381-391.

Hamalainen, W. (2004). Problem-based learning of theoretical computer science. *Proceedings of the 34th Annual Conference on Frontiers in Education*, *3*, p. 20-23.

Hissey, T. W. (2000). Enhanced Skills for Engineers. *Proceedings of the IEEE*, *88*, *8.*

Kolmos. (2004). Estrategias para desarrollar currículos basados en la formulación de problemas y organizados en base a proyectos. *Educar*, *33*, 77-96.

Mata, F., Matarrita, R., & Pinto, C. (2012). Assessing Computer Education in Costa Rica: Results of a Supply and Demand Study of ICT Human Resources. *CLEI Electronical Journal*, *15*(1).

Mills, J., & Treagust, D. (2003). Engineering Education - Is Problem-Based or Project-Based Learning the Answer? *Australasian Journal of Engineering Education, 4*.

Monge, R., & González, C. (2007). *The role and impact of MNCs in Costa Rica on skills development and training: The case of Intel, Microsoft and Cisco.* CAATEC.

Nielsen, L. (2011). *Classifications of Countries Based on Their Level of Development: How it is Done and How it Could be Done.* International Monetary Fund.

Nuutila, E., Törmä, S., & Malmi, L. (2005). PBL and computer programming - the seven steps method with adaptations. *Computer Science Education*, *15*(2), 123-142.

Prieto, L. (2006). Aprendizaje activo en el aula universitaria: el caso del aprendizaje basado en problemas. *Miscelánea Comillas: Revista de Ciencias Humanas y Sociales Vol.64. Núm.124. Págs. 173-196*, *64*(124), 173-196.

Prince, M. (2004). Does Active Learning Work? A Review of the Research. *Journal of Engineering Education*, *93*(3), 223-231.

Prince, M., & Felder, R. (2006). Inductive teaching and learning methods: Definitions, comparisons, and research bases. *Journal of Engineering Education*, *95*, 123-138.

Prince, M., & Felder, R. (2007). The Many Faces of Inductive Teaching and Learning. *Journal of College Science Teaching*, *36*(5).

Reeves, T. C. (2006). Design Research from a Technology Perspective. In J. Van den Akker, K. Gravemeijer, S. McKenney, & N. Nieveen, *Educational Design Research* (p. 52-66). London: Routledge.

Rhodes, C., & Beneicke, S. (2002). Coaching, mentoring and peer-networking: Challenges for the management of teacher professional development in schools. *Journal of In-Service Education*, *28*(2), 297-310.

Schlager, M. S., & Fusco, J. (2004). Teacher professional development, technology, and communities of practice: Are we putting the cart before the horse? In S. Barab, R. Klin, & J. Gray, *Designing for Virtual Communities in the Service of Learning.* Cambridge MA: Cambridge University Press.

UNDP. (1990). *Human Development Report.* New York: Oxford University Press.

UNDP. (2011). *Human Development Report 2011. Costa Rica.* hdrstats.undp.org/images/explanations/CRI.pdf.

Universidad Nacional. (2007). *Modelo Pedagógico.* Heredia: Universidad Nacional.

Villalobos, V., & Monge-González, R. (2011). Costa Rican Efforts towards and Innovation-Driven Economy: The Role of the ICT Sector. *The Global Information Technology Report 2010-2011*, 119-126.

Wang, F., & Hannafin, M. J. (2005). Design-based research and technology-enhanced learning environments. *Educational Technology Research and Development, 53*(4), 5-23.

Wenger, E. (1998). *Communities of practice: learning, meaning, and identity.* Cambridge: Cambridge University Press.

Wing Lai, K., Pratt, K., Anderson, M., & Stigter, J. (2006). *Literature Review and Synthesis: Online Communities of Practice.* University of Otago, Faculty of Education, Dunedin, New Zealand.