Revista Eletrônica de Sistemas de Informação
ISSN 1677-3071

Vol. 9, No 2
2010

doi:10.5329/RESI.2010.0902

Sumário

Revista hospedada em: http://revistas.facecla.com.br/index.php/reinfo
Forma de avaliação: *double blind review*

Esta revista é (e sempre foi) eletrônica para ajudar a proteger o meio ambiente. Ela voltou a ser diagramada em uma única coluna para facilitar a leitura na tela do computador. Mas, caso deseje imprimir esse artigo, saiba que ele foi editorado com uma fonte mais ecológica, a *Eco Sans*, que gasta menos tinta.

(mapa de palavras com os termos mais frequentes nos artigos desta edição)

INFORMATION SYSTEMS GRADUATE EDUCATION AND RESEARCH IN BRAZIL

(paper submitted in June, 2010)

Renata Mendes de Araujo
Programa de Pós-graduação em Informática
Universidade Federal do Estado do Rio de Janeiro (Unirio)
renata.araujo@uniriotec.br

Márcio de Oliveira Barros
Programa de Pós-graduação em Informática
Universidade Federal do Estado do Rio de Janeiro (Unirio)
marcio.barros@uniriotec.br

ABSTRACT

This paper presents the Graduate Program on Information Systems of the Federal University of the Rio de Janeiro State (UNIRIO) and its strategies to foster IS research in Brazil. These strategies include the design of IS research areas and the curricula for its Master degree course; the organization of graduate software development courses; the students' involvement in extension projects and the structuring of a healthy relationship with industrial partners. The paper concludes with the challenges faced by the Program to enforce its strategies and IS research in Brazil.

Key-words: information systems; graduate IS strategy; graduate IS curricula; information systems education.

RESUMO

Este artigo apresenta o Programa de Pós-graduação em Sistemas de Informação da Universidade Federal do Rio de Janeiro (UNIRIO) e suas estratégias para desenvolver pesquisas na área no Brasil. Essas estratégias incluem o projeto de áreas de pesquisa em Sistemas de Informação e do currículo do seu programa de Mestrado, a organização de cursos de pós-graduação em desenvolvimento de software, o envolvimento dos estudantes em projetos de extensão e a estruturação de um relacionamento saudável com parceiros industriais. O artigo é encerrado com a discussão dos desafios enfrentados pelo Programa para desenvolver suas estratégias e a pesquisa em Sistemas de Informação no Brasil.

Palavras-chave: sistemas de informação; estratégia de Sistemas de Informação para pós-graduação; currículo de pós-graduação em Sistemas de Informação; educação em sistemas de informação.

1 INTRODUCTION

Since 1979, the Federal University of Rio de Janeiro State (UNIRIO) has been continuously recognized for its educational and research results in the areas of Arts, Human Sciences and Health. In 1999, the Information Systems Department (DIA) was set up to support a new undergraduate course on Information Systems (DIA-UNIRIO, s.d.). This department has been part of a strategy to consolidate UNIRIO’s position in the national technology education and research scenario, especially in Computer Science (CS).

In 2006, the department secured accreditation from the Brazilian Education and Culture Ministry (MEC-CAPES) to establish the Information Systems Graduate Program (PPGI-UNIRIO). This Program aims to keep the department’s educational strategy aligned with the IS area, allowing for the continuous formation of IS professionals and strong interaction between undergraduate and graduate courses.

Currently, the PPGI-UNIRIO includes 18 researchers who have received their PhD degree in Computer Science from top quality Brazilian universities. As a graduate program, we focus our research and lecturing efforts on Information Systems, aiming at strengthening the scientific and professional knowledge in the IS field; providing the market with skilled professionals in modeling, development, selection, deployment, and management of IS in organizations; allowing students to develop their skills for teaching and researching in IS; contributing to the continuous improvement of UNIRIO’s academic community; developing state-of-the-art IS research; and contributing to social and economic development of the geographic region where we are located.

After three years of activities, PPGI-UNIRIO can be considered a consolidated research center. 23 MSc dissertations were completed in 2009 with academic research growing each year (see Figure 1). These figures are expected to rise in the ensuing years through the strengthening of our current staff’s work as well as through the hiring of new researchers. The growing number of student applications for the selection process also contributes to that. Additionally, our research group has gained visibility in the Brazilian’s Computer Science research scenario, as ascertained by its participation in both national and international research projects/ consortiums, along with the organization of IS-oriented academic conferences. Finally, the development of joint projects with the industry has been providing benefits both for academic research towards real-world problems as well as for students’ training.

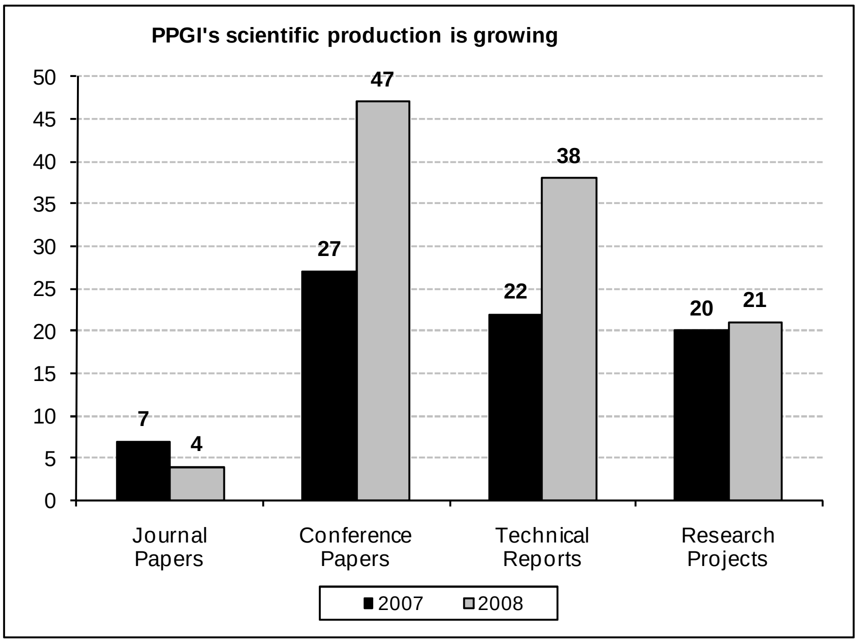

Figure 1 - Summary of academic research per year

Source: the authors

This paper details the strategies being conducted by the Program to foster IS research in Brazil. In the process of implementing both the undergraduate and the Master Degree courses, we have been required to address several IS education-related issues, such as the design of IS research areas, the definition of a curriculum for the IS Master program, the organization of software development courses, the involvement of students in extension projects, and the development of a healthy relationship with industrial partners. This paper describes our experience in dealing with this scenario and the challenges being faced to enforce these strategies.

The paper is structured to present, in the next section, the main strategies conducted within PPGI-UNIRIO, which we believe have helped our consolidation and growth as a graduate course. These strategies comprise curriculum design and involvement in the outline of a country-wide IS research agenda. Following this, we characterize the demand for an IS graduate course in Brazil. Next, we outline our main strategies for conducting industry-aligned research. Finally, the paper presents a case study executed during a software construction course in which students were given work assignments strongly related to a problem that was addressed by many software development organizations at the time, summarizing lessons learned from this experience. The final section concludes the paper by discussing the outcomes and challenges for the future.

2 A RESEARCH STRATEGY ALIGNED TO THE BRAZILIAN IS CURRICULUM

The Brazilian Computer Society (SBC) is a non-profit scientific organization that gathers researchers, students, and professionals who perform scientific research and technology development in Computer Science. Since 1999, SBC's Computer Science Education Working Group has been elaborating a curriculum for undergraduate courses related to the Computing area (SBC, 2003). This curriculum is formally used as a reference by Brazilian educational institutions. In this document, SBC defines two classes of undergraduate courses: (i) those regarding Computer Science theories, problems and fundamental aspects; and (ii) those which view Computer Science as the means to solve non-CS problems. Information Systems Bachelor courses are part of the latter group, while Computers Science and Computing Engineering courses represent the former. The Computer Science Education Working Group also discussed what should be a curriculum for Information Systems courses which has also been used as a reference for structuring and evaluating IS undergraduate courses.

According to the proposed curriculum, the purpose of an IS undergraduate course is to prepare professionals to perform information management, planning, and innovation projects related to system development, process modeling, and IT infrastructure. In summary, such courses should aim to provide the industry with professionals that are able to understand organizational problems in different domains and use computer-related technologies to provide a solution for these problems. Therefore, the IS curriculum encompasses disciplines on computer science and information systems (as the basis for solving problems), but also humanistic knowledge, in order to prepare the professional to properly identify such problems and understand the social environment in which they are embedded. A remarkable recommendation within the curriculum proposal is the need to provide learning through participation in real projects, especially in association with industry partners.

This curriculum is also aligned with the view proposed by international associations such as the Association for Computer Machinery (ACM) (ACM COMPUTING CURRICULA, 2008), the IEEE Computer Society, and the Association for Information Systems (AIS) (COMPUTING CURRICULA, 2005). The Computing Curricula suggests that Information Systems specialists should focus on integrating information technology solutions and business processes to meet organizations' business objectives in an effective and efficient way. The curricula recognizes that the majority of IS programs is located in business schools (specially in USA), emphasizing the organizational and behavioral aspect of IS, whereas those that exist under computer science schools usually have a stronger technology focus (that is how Brazilian courses are usually shaped).

The IS undergraduate course at UNIRIO was established in 1999, following the Ministry's and SBC's directives and recommendations for IS education. Our IS course comprises areas such as Software Engineering, Databases, Data Structures, Algorithms, Management and Administration, Computer Networks, and Operational Systems. Despite considering business subjects, the course still has a strong technology focus, more aligned with a Computer Science view. The course is continuously evaluated by the Ministry, and its students have been quickly absorbed by the market. One remarkable characteristic of the undergraduate course at UNIRIO is that it is offered in the evening, following recommendations by the Brazilian Government to increase the availability of undergraduate opportunities for students who cannot be fully dedicated to their studies due to working needs.

Advancing on our educational strategy, in 2006 we proposed the establishment of a Master Degree course on Information Systems. This Master Degree course is also associated to the Computer Science area, where our core researchers have their DSc degrees. Alignment with Computer Science was also needed for accreditation purposes, since the CS area is responsible for evaluating and following course evolution. The IS Master Degree envisions that IS research comprises three major aspects: (i) modeling, which comprises system conception, information and knowledge representation; (ii) development, which deals with the complexity of developing large scale IS projects; (iii) management, regarding the effective usage of IS into business, organizational, and social contexts. Pursuant to these major aspects, the course currently entails three research areas: Knowledge Representation and Reasoning, Distributed Systems and Networks, and Business Supporting Systems, respectively related to the aspects mentioned above.

The Knowledge Representation and Reasoning area studies alternatives for modeling and representing the knowledge and behavior of human and computational agents interacting through a system. The area discusses solutions to problems related to knowledge models based on metadata and ontologies; definition of heuristics for knowledge retrieval; application of AI and machine learning techniques on IS domain problems; modeling and developing IS with high-level automation, like Semantic Web systems; and developing IS for non-conventional applications, with complex data structures or requirements involving the simulation of virtual agents' behavior. Currently, major research projects residing within this research area address organizational learning; knowledge, information, and data management; information management in distributed environments; operational research and optimization; digital entertainment and TV; and business intelligence management.

The Distributed Systems and Networks area follows the trend of IS to make use of distributed architectures and organizational environments. This research area studies the different contexts for the use of technologies associated to organization, construction, use and performance evaluation of distributed systems and data communication networks. The focus lies on providing solutions to problems related to: distributed system verification, management, and monitoring; IS performance improvement, through distributed and parallel access to stored data in local networks and database clusters; new programming models for distributed applications; environments, tools, techniques, and methodologies for data integration in organizations; new mathematical models for evaluation of computing/ communication systems; study of user behavior in different computing/ communication systems; characterization of data traffic in corporate environments; and the development of multimedia applications. Currently, research projects within this research area address the following topics: multimedia information systems; operational research and optimization; system security; educational models; tool support for distance learning; and network management.

The Business Supporting Systems area discusses the IS management aspect, being concerned about the impact of IS on different business domains and how their use can be continuously improved in organizations. This area discusses solutions to problems related to business process modeling and management; organizational information architectures; IS development and deployment; IS project management; IS usability and accessibility; collaborative and social systems; organizational knowledge management; decision support systems; and the application of Economy, Finance and Management concepts as tools to support decision making on IS projects. Major research projects for the Business Supporting Systems area include digital democracy; collaborative systems; human-computer interface; business process management; and project management for information systems.

Interdisciplinarity is another major pillar of our IS graduate course. Since we view IS as the means to support other disciplines and practices through the use of computer-related artifacts, getting closer to researchers in other main fields developed inside UNIRIO was a natural thing to do. As an example of such effort, a research is being conducted, through the Knowledge Representation and Reasoning area, on music repositories and knowledge representation about Brazilian popular music in partnership with the UNIRIO Arts Department (MOTTA, 2008; SANTIS, 2008). We expect that more examples of interdisciplinary work will arise, with information systems as the means to solve problems in different domains. For instance, an association between IS and Digital Heritage studies is being conducted within UNIRIO's Social Memory Department, aiming at discussing Social Memory and Digital Heritage through the Web.

3 CHARACTERIZING STUDENT DEMAND FOR AN IS GRADUATE COURSE IN BRAZIL

Since our first student selection process (2006), PPGI-UNIRIO researchers were surprised with the high number of candidates – more than 100 students applying for about 30 places to be filled per year. This is even more remarkable if we consider that the number of students applying

doi:10.5329/RESI.2010.0902001

for consolidated Brazilian CS Programs has been decreasing in the recent years. In order to characterize the demand for positions in the different research areas that our Master degree program provides, we have collected non-sensitive and unidentifiable information about applicant students during the selection processes and analyzed this information. This section summarizes our analyses and provides the means to improve our research and development strategies to foster IS role within the Brazilian Computer Science community.

The following chart (Figure 2) characterizes student demand for positions within the three research areas made available by the program. As the chart shows, there is a concentration of demand on the Business Supporting Systems (BSS) research area. We can also observe a growth trend for the Knowledge Representation and Reasoning (KRR) area and a decreasing trend for the Distributed Systems and Networks area (DSN). In average, 62% (sixty two percent) of the applying students select the BSS area as their first option, while 23% (twenty three percent) and 15% (fifteen percent) select the KRR and DSN areas, respectively.

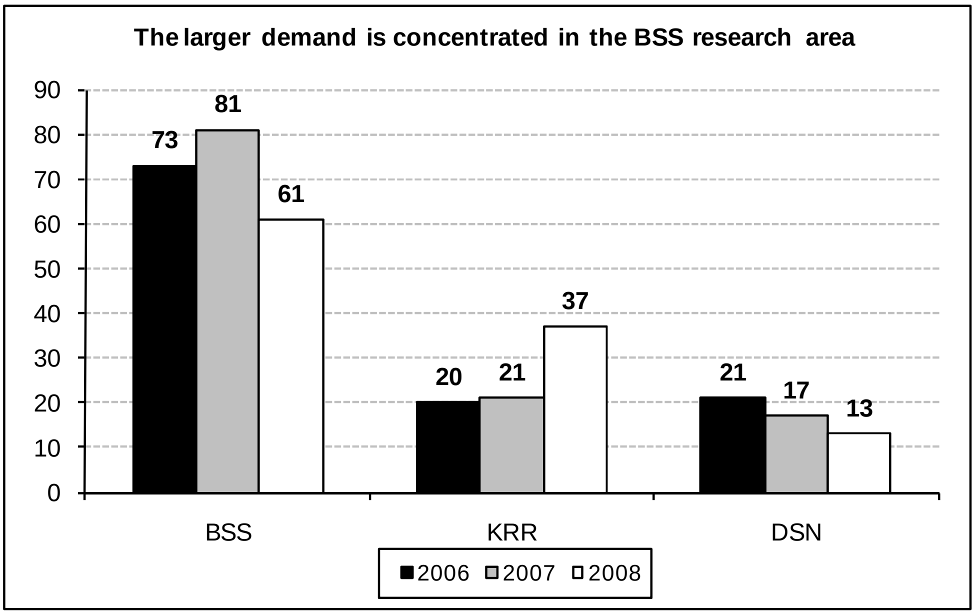

Figure 2 - Number of candidates applying for each research area

Source: the authors

Figure 3 depicts the distribution of applicant students by gender. As can be noted, most of our candidates are male. In average, we have one woman for every three men, that is, 75% of our candidates are male, while only 25% are female.

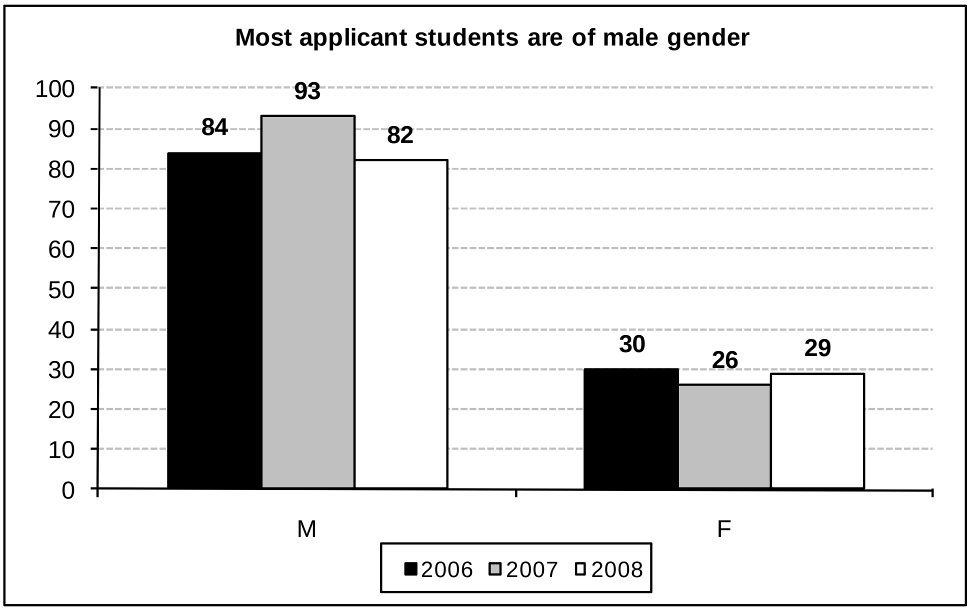

Figure 3 - Number of candidates, according to gender

Source: the authors

Figure 4 presents the distribution of applicant students by age. Collected data shows that most of our candidates are less than forty years old, though there were older candidates participating in the first and second selection processes. We can observe a balanced distribution of recently-graduated candidates (less than 30 years old) and applicants that have worked in the industry for some time before becoming interested in a Master degree course. We believe that this is an interesting student population for an IS graduate course, because it allows for interaction between freshly-graduated students and more experienced ones. Younger students may introduce the more experienced ones to new technologies, with which the later may not have been involved. Experienced students, on the other hand, may expose their practical knowledge in classroom or informal discussions, enriching the younger ones' view of the practical world. According to our view of IS research, this experience exchange is very welcome. It is also noticeable that most CS graduate courses in Brazil attract recently-graduated students, usually bellow 30.

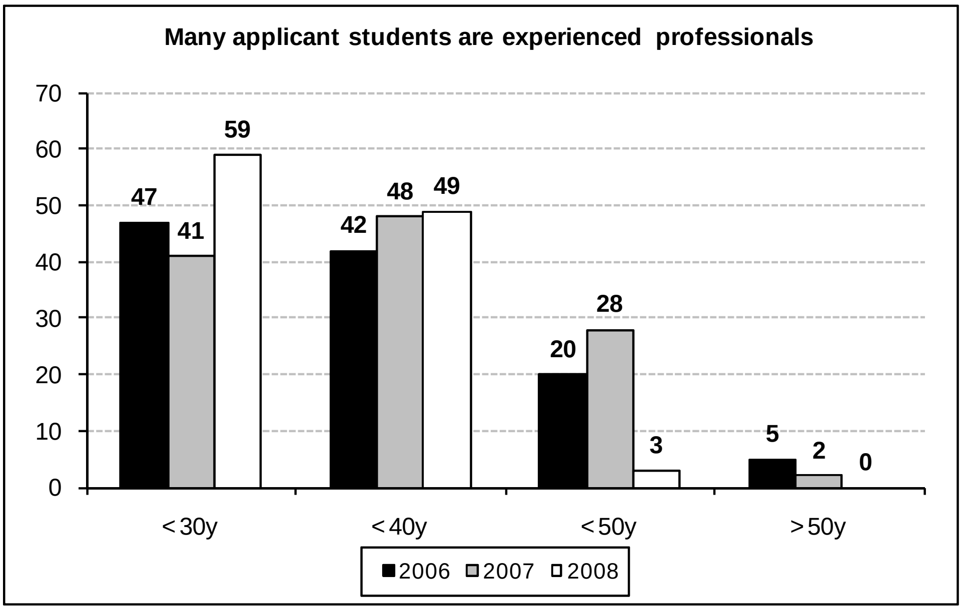

Figure 4 - Number of candidates, according to their age

Source: the authors

The following chart (Figure 5) presents the distribution of applicant students by the type of company in which they work. There is a balance between candidate students coming from privately-held companies and government-held organizations. It is important to notice that Rio de Janeiro (Brazilian state in which PPGI-UNIRIO is located) was our country's capital city up to 1960, when the capital moved to Brasília. Thus, Rio de Janeiro is home for many government-held companies, like Petrobras (petroleum), Eletrobras (electric power distribution), Furnas (electric power distribution), SERPRO (data processing), and BNDES (banking). In average, 49% (forty nine percent) of our candidates work for privately-held companies, while 44% (forty four percent) come from companies held by the government. The remaining 7% (seven percent) come from military institutions or is comprised of students with no former working experience.

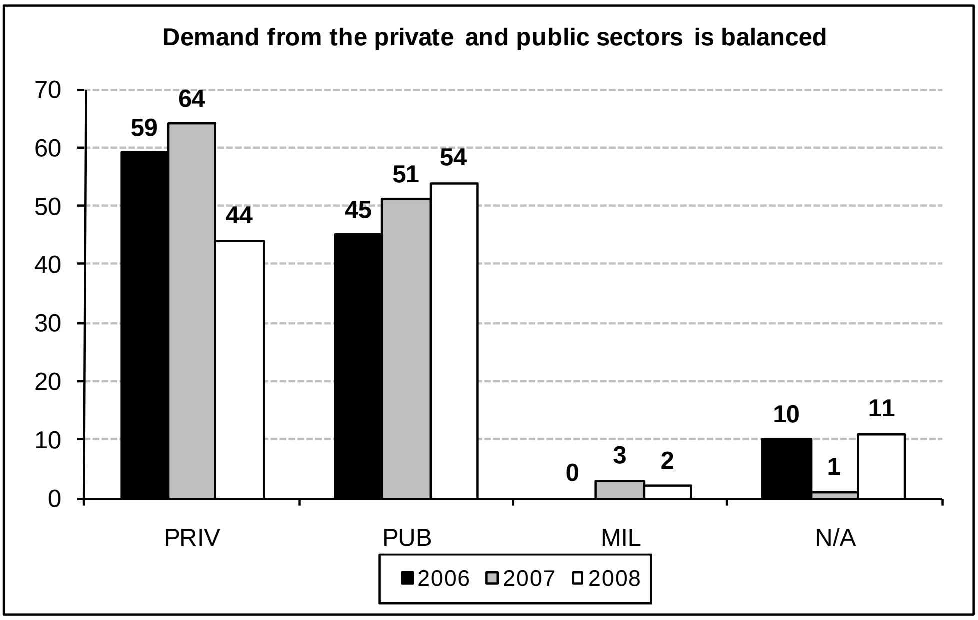

Figure 5 - Number of candidate students according to major industrial sectors

Source: the authors

Figure 6 presents the distribution of applicant students according to their experience domain, that is, the economic sector that encompasses their employer (CS – Computing Services, Energy, Education, Services, Telecommunications, Construction, Financial, other). In average, 25% of our candidates come from computer-related companies, mostly software development organizations. 16% of our applicants are educators, usually part-time lecturers for privately-held universities. Recent governmental rules for education require that a large number of university professors must have developed graduate studies (Master or Doctoral degrees) and the many universities are still adjusting their staff to these rules, thus requiring their educators to seek graduate courses. The energy sector, particularly petroleum and electric companies, is represented by 11% of our candidates. In average, the civil construction sector provides as many candidates as the energy sector, but this is due to a peak in 2007, when we had 25 candidates from construction companies. The regular number of applicants from these field is around 5 to 10.

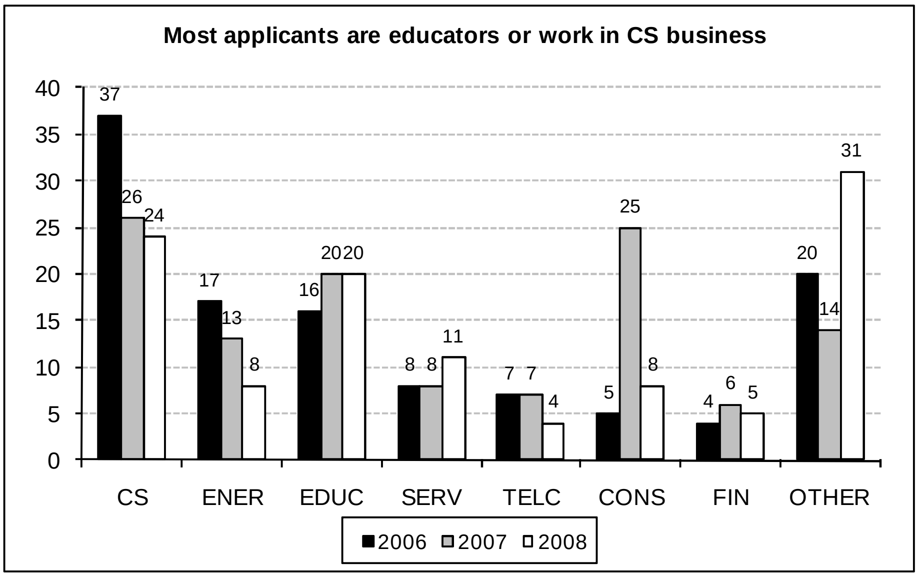

Figure 6 - Number of candidates, according to industry

Source: the authors

Finally, Figure 7 depicts the distribution of applicants according to their academic background, that is, the type of course in which they have obtained their bachelors degree (Management, Computer Science, Math, Engineering, other). In average, 67% of our applicants had their bachelors in Computer Science, either Information Systems, or Computers Science or Computing Engineering. Engineers (electrical, civil, among others) represent the second largest group, comprising about 7% of our applicants. Math and Management courses come next, providing about 5% of our candidates each.

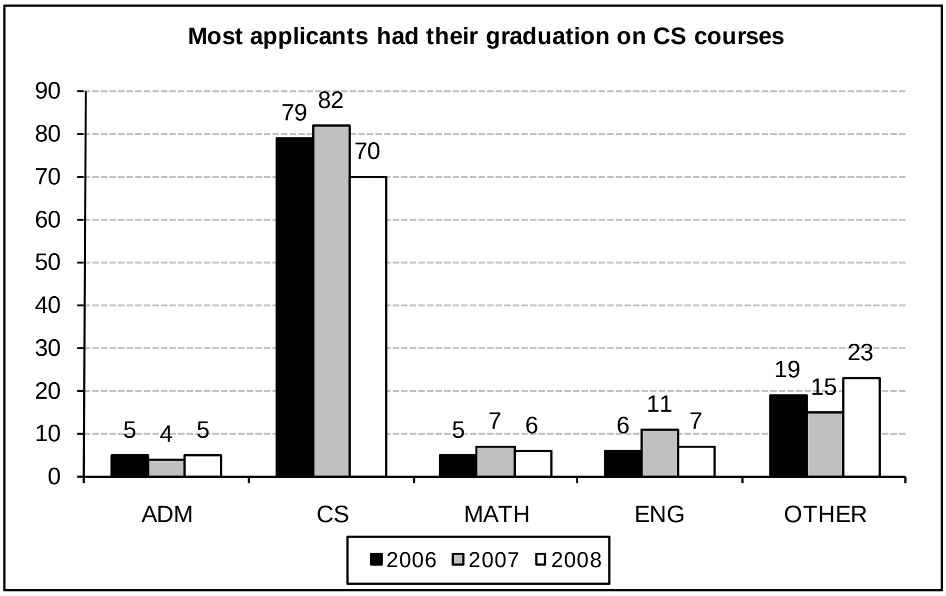

Figure 7 - Number of candidates, according to academic background

Source: the authors

Thus, data about our selection process shows that the usual candidate for an Information System Master's course is a male professional, from 25 to 40 years old, with background in Computer Science, working for a private or government-held computer technology-related company, and interested in conducting research on business support systems.

4 APPLIED RESEARCH AND RELATIONSHIP TO INDUSTRY

We believe that conducting applied research, focused on real problems, may be the reason for the interest on PPGI-UNIRIO. The candidates' basic profile consists of professionals in private or government companies searching for opportunities to conduct research projects aligned to the problems they face in their daily work. Although these students have less availability to perform research activities, their view about domain-specific problems and the feasibility of solutions brings a strong and healthy perspective for interaction between theory and practice.

Some actions performed by researchers at PPGI-UNIRIO enforce its applied characteristic. The Research and Practice in Technology Group (NP2Tec) involves PPGI-UNIRIO's researchers who perform consulting activities and collaborative research with the industry (NP2TEC, s.d.). NP2Tec's projects are focused on areas such as Business Process Modeling and Management, IT Architecture, Information Systems engineering and development.

One of the main tenets of this group is that these projects must be part of the students' education, either in undergraduate or graduate courses. Projects are managed by researchers, who continuously supervise the students taking part therein; results are, thereafter, published in academic conferences and journals.

NP2Tec's interaction with industry partners has proved to be one of the most successful approaches for spreading PPGI-UNIRIO's research products and beliefs. In almost all projects conducted by NP2Tec, both the results of MSc dissertations or research projects bear space to be applied and validated; also industrial partners' problems are continuously generalized to become new research problems to be addressed by MSc students in their dissertations. Approaches for IS requirements' elicitation through business models, methodologies for SOA definition, methods for knowledge formalization using ontologies and business processes, approaches for business process management, and ways to structure business processes' offices are just a few examples of subjects being studied both in research projects and dissertations aligned with and applied to industry partners' real world problems.

5 TRAINING STUDENTS BASED ON REAL-WORLD PROBLEMS: A CASE STUDY

Concerning educational methodologies, IS education, even at graduate levels, must consider the “real world problems” which bring the main motivation for IS research. PPGI-UNIRIO explores the real world view in special courses, such as the one which will be presented in the following sub-sections.

Software construction (McCONNELL, 1993) is the set of disciplines which comprises the middle part of a software development life cycle, conveying detailed design, implementation, and testing activities. It is preceded by requirement elicitation and architectural design, and followed by deployment and maintenance. If analyzed in isolation, software construction encompasses the set of software engineering activities where technical (computer science related) skills are more intensively applied than social and communication (human related) abilities.

Lehman (1996) proposes a classification schema for software systems that divides systems into three classes: S-systems, P-systems, and E-systems. S-systems comprehend the set of systems that can be formally defined by a specification and are derivable from it – that is, the problem is stated in terms of the entire set of circumstances to which it applies, and its solution is well known (PFLEEGER, 1998). P-systems are systems in which the real world cannot be fully described or a complete implementation of a known theoretical solution is not feasible in current computing platforms. Therefore, implementation is usually limited to an approximation of the known solution. Finally, E-systems are systems in which the real world is under constant change: while either S- and P-systems consider a stable real world, an E-system incorporates the changing nature of the real world and has to be continually updated in order to remain useful.

We observe that most real information systems – that is, systems which will be used to support the business processes of a real industry or economical sector – are E-systems, whose requirements may change according to external events (such as new market demands, changes in regulation, availability of new concurrent products, among others) and whose usefulness depends on the ability of the development team to constantly adapt the system and keep it functional as its requirements change.

Thus, the development of real information systems is a complex task in which software construction and requirement elicitation activities are entwined. As requirements are identified or changed during the development life cycle, no clear division among construction and elicitation activities can be precisely established. Developers implementing the components which will compose the system may have to rethink their design assumptions according to requirement changes. Requirement engineers, on the other hand, may have to persuade users to postpone certain nonblocking requirements in order to keep project schedule and planned release dates. This involves negotiation within the development group and users. Negotiation requires preparing arguments, structuring the complex causal relationships involved in the decisions, and passing along this knowledge. Therefore, technical knowledge (such as software design principles or programming language proficiency) and human-related abilities (such as communication skills, pro-activeness, planning, among others) should be experienced in unison, allowing problems observed in one aspect to magnify the other aspect, requiring decisions to be taken under incomplete information, and trade-offs to be carefully evaluated.

However, software system development education usually separates teaching of technical skills from human related activities. In the former, it is common practice to give precise specifications to students (that is, experience the development of S-systems), so that they can apply their technical knowledge independently of any kind of misunderstanding due to faulty communication. In the latter, human-related abilities are practiced without the need to develop a real product and, thus, there is no need to negotiate the introduction of new requirements or changes to the project plan with technical developers. Many technical educators reduce the value of human-related activities ("what does not compile, does not count"), while the second group place technical skills in a secondary role ("there is no sense in writing a system that does what no one needs"). Thus, junior developers are usually unable to use technical knowledge in unison with human-related abilities to the best of the projects under development.

In our Master Degree course on Information Systems, we envision that IS education should strive to present students with assignments and case studies as close as possible to real world problems (that is, E-systems, instead of S-systems). This is a huge distinction between Information Systems and Computer Science, since the latter does not have the need to be applied and thus can use abstract examples to highlight the specific issues under interest.

In the following sub-sections, we present a case study executed during a software construction course that was taught to our Master Degree students on IS. This course represents one of our experiences while trying to put to practice our perspective of confronting the students with real-world problems. We tried to emulate a software development organization scenario in order to transfer the software engineering knowledge (limited to software construction activities, but involving communication abilities) which would be required from professionals in such a scenario.

During the course, students were given work assignments strongly related to a problem addressed by many industrial projects at the time: the development of software applications to support the implementation of the Basel II framework in Brazilian financial institutions. Students were asked to develop an information system to support a set of functions related to this problem domain and to adapt such system according to events happening in the real world (such as changes in the requirements or invalidation of previous assumptions). Thus, the construction of an E-system could be fully experienced, as its specification changed due to real events and the system's internal structure should be adapted to support such changes.

5.1 THE BASEL II FRAMEWORK

In June 2006, the Basel Committee on Banking Supervision (BIS, 2009) published the final version of the Basel II framework. This framework was presented as the result of a seven-year project to secure international convergence on supervisory regulations governing the capital adequacy of internationally-active financial institutions. The Basel framework aims to create an international standard, which countrywide regulatory agencies can use when creating rules about how much capital financial institutions need to deposit to guard their liabilities against the market, credit, and operational risks they face due to their investment strategies.

The development of such framework started in 1988, when a first version of the document was released and adopted by the central regulatory agencies for banks and financial institutions in several countries. Through 2001 and 2003, the framework was improved according to feedback from these regulatory agencies and to the results of quantitative studies on the impact of their adoption. Subsequently, additional proposals were released in this period. Finally, a complete document was released and endorsed by the Central Bank Governors and Heads of Banking Supervision of the Group of Ten countries (BIS, 2006).

The Basel II framework describes a comprehensive measure and minimum standard for capital adequacy that national supervisory authorities should strive to implement through domestic rule-making and adoption procedures (BIS, 2009). It seeks to improve the existing rules by aligning regulatory capital requirements more closely to the underlying risks faced by financial institutions (such as banks, asset managers, insurance companies, among others). In addition, the Basel II framework is intended to promote a forward-looking approach to capital supervision, to encourage financial institutions to identify the risks they take, and to improve their ability to manage these risks. As a result, it is intended to be more flexible and better able to evolve with advances in markets and risk management practices.

On the other hand, the Basel framework is not an international agreement in the legal sense: there is no signature or contract. National regulatory agencies of interested countries may endorse the framework by issuing regulations aligned to its principles.

The Brazilian Central Bank, the national regulatory agency for Brazilian financial institutions, has supported the Basel framework since its first proposal (the Basel I framework was adopted by the Brazilian Central Bank in 1994). However, in 2004, the regulatory agency took a more conservative approach to enforce the Basel II framework in Brazil, by issuing rules which would require that "most financial institutions apply a simplified standard, which consists of several improvements upon the Basel I framework, including elements which, in accordance to financial instruments to mitigate credit risk, allow for better alignment among the required capital and risk exposure" (BRAZILIAN CENTRAL BANK, 2004).

Furthermore, financial institutions should prepare their internal control policies (and the information systems that support these policies) to comply with further regulations in accordance to the entire Basel II framework. In the 2007-2008 period this transition was accelerated and the Brazilian Central Bank issued regulations requiring that financial institutions inform their market risk exposures through a monthly market risk report (or DRM, for "Demonstrativo de Risco de Mercado", in Portuguese).

The DRM report and the events that happened from January to May, 2008, present us with the context for the case study depicted in the following sub-sections, in which a group of students developed a programming class project to handle values from the DRM report, being influenced by events occurring in the real world.

5.2 THE COURSE ON SOFTWARE CONSTRUCTION

The PPGI017 Course on Advanced Techniques for Software Construction aims to teach Master Degree students how to develop high quality software by applying known standards to the coding and design phases of the software development life-cycle. Students learn the importance of code organization and comprehensibility, general software design principles, characteristics of good and bad software design, change to incorporation issues, design principles applied to object-oriented development, class characterization, package design, design patterns, and refactoring. The course also includes lectures on other practical issues related to software construction, such as configuration management, bug reporting and tracking and unit testing.

The course is optional for the Master Degree students at the Federal University of Rio de Janeiro State Information Systems Graduate Program (PPGI-UNIRIO, 2009). The 2008 offer of the PPGI017 Course occurred in the first semester of the year and only six students attended the course (a low, but not unusual, number of graduate students whose interests include software design and programming). Student evaluation within the course is composed of a formal exam (for a weight of one in their final grade) and up to four practical work assignments that add up to a second evaluation (for a weight of two in the final grade). This strategy allows us to determine whether students have built a conceptual background on software design and also if they can put that knowledge to practice in the work assignments.

Our vision of Information Systems as a practical science, which should be aligned to real world needs, has always been a driving force to decide which work assignments should be submitted to students. However, it is difficult to find project opportunities large enough to be relevant, but also small enough to be feasible in a four-month course period. Moreover, it was desirable to present the students with cases that have characteristics of real world projects – such as imprecise requirements, late-changing requirements, competition among solution providers, and strict deadlines – instead of completely-specified, flat horizon projects, which are common practice on programming assignments. It was also desirable to subject the students to “not-so-reliable” development environments, so that they can experience network and communication problems, and unscheduled changes to the platform, compiler, and other development tools. After all, these are scenarios which may happen in the real world, and students should be prepared to handle them accordingly.

We tried to impose some of these restrictions on a former offer of the PPGI017 course (in 2007). In such offer, students were subjected to assignments bearing somewhat fuzzy specifications. They should, by themselves, ask the supervisor for more information and for clear definitions over some concepts involved in the assignments. However, these were abstract projects with little resemblance to the real-world projects they would be involved with the industry. Due to these characteristics, students thought they had the right to question the assignment. So, instead of searching for clear definitions, they sought to reduce the scope or to limit the assignments to a subset of the cases it was intended to handle gracefully. The unavailability of more precise specifications led some of them to justify errors which could have been deeply investigated (under the argument that they were in a software construction course, not on a requirement elicitation one). In the end, students had applied the skills and lessons learned in the discipline, but we were only partially satisfied with the results and, consequently, also with the assignments that had been proposed.

While preparing the 2008 course offer, we sought the opportunity to explore a real world problem still in the open, and the next Basel II framework implementation step in our country presented the perfect scenario. Students would develop a tool to help the Brazilian Central Bank analyze the DRM reports sent by the many financial institutions under its supervision. Each report could have thousands of nominal values and an analysis tool to support the investigations the central agency needed to conduct was desirable. For the purpose of student evaluation, the proposed tool specification would be broken down into three or four distinct assignments.

Specifications of the proposed tool introduced the students to the financial domain – a single one of them had had former experience in this domain – and presented the desired functions which the proposed tool should implement (purposefully, fuzzy specifications). Directions to download a set of documents published by the Central Bank itself, including layouts and formulae specifications to calculate the values to be presented in the DRM report, were given to the students, allowing them the access to documents being used to build industrial applications for the domain. They could e-mail questions directly to responsible persons in the Central Bank, as could any one in a software development organization. Delays to obtain answers could be experienced, as they would in the real world.

Students could read issues related to their assignments in the newspapers. They felt as if they were working for software companies producing banking software or compliance analyzers for the Central Bank, capturing and analyzing the reports – and they took it seriously on their development. Motivation was fantastic, as I had never seen before in a programming course!

Also, a competitive evaluation scheme was set. Students were divided into three groups (A, B, and C), each composed of two students. We decided to divide students into three groups (instead of two) because they suggested that it would be easier to work with a single partner than with two (due to the agenda of their courses and other personal commitments). Moreover, larger groups would require larger assignments, which would be harder to combine into a single reference implementation. Finally, three groups would allow more competition than two groups.

Each group should deliver every assignment and each deliverable was evaluated according to its internal and external properties: code readability, simplicity, class organization, and code and design organization. Draws were allowed and late deliveries were punished by deducting one point for each day after the original delivery date. This enforced competition, as it would happen among software solution providers, and competition was embraced by the groups.

5.3 THE DRM REPORT – CHRONOLOGY OF EVENTS

In the following, we describe the events regarding the definition and regulatory requirements related to the DRM report and how they influenced the PPGI017 course, allowing students to experience the development of an information system subjected to changes in the real world. Events are presented in chronological order:

- Jan/2008 – the Brazilian Banking Institutions Association, an association supporting the Central Bank on the design of the regulations related to the DRM report, released a public notice on the report, establishing most of the formulas to calculate the risk exposures that it should convey and the format in which the financial organizations should deliver it to the Central Bank;
- Mar/2008 – The 2008 offer of the PPGI07 course started and the students were presented to the Basel II framework case. They were given a briefing on how risk exposure information should be provided to the Central Bank, received the public notice from the Brazilian Banking Institutions Association, and the proposed layouts for the DRM report;
- Mar/2008 (a week later) – students were presented to the first work assignment, in which they had to develop (design and code) a set of

doi:10.5329/RESI.2010.0902001

classes to model the information and risk exposures to be presented in the DRM report. As in the assignments to come, no particular software development process or toolset was imposed to the students, which were allowed to use any tool or methodology they found helpful. Java was selected as the programming language for the assignments, just to ensure that all students would code in the same programming language. Unit testing was a requirement for a successful delivery;

- Apr/2008 (two weeks later) – the first assignment was delivered by the students and evaluated by the course supervisor. Group A had the best deliverables, though the internal organization of their code could be improved. They received a grade 9 (nine), while groups B and C made a draw for the second position, earning a grade 7 (seven). A summary of each group's evaluation was made available for all groups, along with the distinct implementations. This allowed them to learn from each other's mistakes and be better prepared for the second assignment. Group A's deliverables were improved to become a reference implementation to be used by all groups in the following assignments;
- Apr/2008 (one week later) – students were presented to the second work assignment, in which they had develop classes to read the risk exposure information from a DRM report on the reference implementation of the class structure they had developed in the first assignment. Examples of DRM reports, generated using a worksheet, were delivered as a test-bed for their implementation (with a known formatting error in one of the report entries);
- Apr/2008 (one week later) – two groups of students (A and B) detected a formatting error and communicated it to the course supervisor. They were asked to manually correct the report examples and inform the third group. One group (B) also found an unexpected error (a real one, non-intentionally introduced by the course supervisor), which was handled in the same fashion. Moreover, some business rules which were not clearly defined in the specification (such as the length and content of some fields of the report) were established by the supervisor;
- May/2008 (one week later) – two groups delivered the assignment on time (B and C), while the third one delivered it two days later (A). The three groups had a draw for the first position, but all of them could improve the class design used to implement the requested functionality. Each group earned a grade of 8 and group A suffered a 2-point penalty for late delivery (ending up with a grade 6). As in the former assignment, a summary of each evaluation and its related deliverables was made public to all groups, and a reference implementation was developed to be used in the following assignments. This time, however, the reference implementation was coded almost from scratch by the course supervisor;
- May/2008 (one week later) – the third assignment was submitted to the students. They would have to develop a tool to provide a graphical

visualization for the risk exposure values within a group of DRM reports received for the same period (that is, the same month and year). The graphical representation should be customizable: it should allow users to select which parts of the DRM report would be presented in the screen. The tool itself was loosely described, and discussion during the presentation of the assignment contributed to cut the edges of its specification;

- May/2008 (same week) – the Brazilian Central Bank officially published a regulatory requirement describing the final version of the DRM report. Such version had several distinct aspects from the one proposed in January. Among such differences, (a) the DRM report should include a new section, aggregating risk exposures from the previous ones; (b) the DRM report, which was supposed to be published in a textual file format, was now to be published in an XML format; (c) an extra vertex was included in every financial curve upon which a financial institution could have risk exposure; and (d) spot values upon which banks could face risk exposure were required to be treated as curves (in spite of not having any market to market future values);
- June/2008 (three weeks later) – the third assignment was delivered by the students. While they still expected a fourth assignment, this was supposed to be a simple extension of the third one, most probably another strategy to visualize the information they already knew how to read and represent in object structures. However, they received the official regulatory requirement released in May, in its original format, and discovered that they would need to update the tool so that it continued to be useful. Two weeks was the deadline to deliver the new version;
- June/2008 (two weeks later) – while course conclusion date was fast approaching, the three groups strived to accomplish the last work assignment. Group B received the higher grade, while Groups A and C had a draw in their evaluation. A complete tool emerged as a result produced by the groups, actually of interest to the compliance agency. However, the major result was the intangible opportunity experienced by the students attending to the course (and the lecturer).

5.4 AFTERMATH AND LESSONS LEARNED

One expected that students would see the value of previous discussions on code organization and design principles while developing the proposed project. When they had to come back to the code written months before, they should be able to appreciate comments and code readability. Thus, it was expected that they saw the value of foreseeing future changes and building the design into separate components that would shield the changes suffered by the software, so that they affected only limited portions of the system.

On the other hand, conducting a project experience like the one described in this paper is not an easy task. We discuss herein some of the

doi:10.5329/RESI.2010.0902001

difficulties found during the PPGI017 course offer presented in the preceding sections:

- It is very difficult to find real problems, like the Basel II framework implementation. Usually, real world problems are more complex than it would be feasible for a short course, like PPGI017. On the other hand, a similar situation may be emulated, or a complex problem can be cut to pieces in the right size for the course requirements and limitations;
- Since the real world is being addressed, the scenario influencing the assignments cannot be controlled by the course supervisor: there is no guarantee that relevant issues and events will arise. Thus, the course supervisor may find the need to suggest development scenarios that differ from the real world itself, including events relevant for the course's purposes. This may affect motivation, since the closeness to the real world and the events that are really happening is broken;
- On the flip side of the scenario, where no events occur, real world events may occur in a different direction than originally intended by the course supervisor. If such events are not aligned with the course objectives, the supervisor may act as a filter for the incoming news from the outside world, presenting only relevant events to the students;
- It is very useful to work with partial deliveries, analyzed by the course supervisor and distilled into reference implementations for the assignments in sequence. A phased evaluation approach allows students to learn from their errors and provides a good starting point for the following deliveries. The reference implementation also contributes to keeping competition up, since all students depart from the same point for every new delivery (no one is left behind due to a bad deliverable). On the other hand, the course supervisor has an extra amount of work to compile several proposed solutions into a single, well-defined reference implementation;
- Students were free to use any methodologies, technique or tool they found useful to complete the assignments. In a course addressing a particular methodology, tool support or development philosophy – such as agile processes, robust processes, usage of version control systems, usage of test coverage tools, applications of static code analysis, among others – restrictions should be imposed to the students in order to constrain their options while proceeding with the development;
- Aligned to the former point, no restriction was directly imposed upon the design proposed by the students, such as limiting the cyclomatic complexity of their code, restricting method size in LOC, measuring cohesion or coupling. Design attributes (such as class dependencies, package organization, cohesion and coupling) were evaluated in an adhoc manner by the course supervisor and discussed with students. Possibly, the usage of software metrics would be helpful for the supervisor and could provide a basis for students to comprehend how such metrics could be useful to them;

- Shared evaluation, in which a student can read the evaluation received by the other students and even analyze their deliverables, allows students to learn from errors committed by the others, thus preventing these errors to be incurred again. However, some students may prefer to keep their evaluation private (the course supervisor may be required to do so). In large groups, a letter of permit for disclosing evaluations may be required (or even a closed evaluation model may need to be used);
- We decided to apply a single written exam during the course. A better strategy would include a written evaluation by the end of the course, to observe whether students had improved their ability to write about the issues addressed during the course. Though a comparison would be feasible, due to the reduced number of subjects, we could hardly draw any definitive conclusion about the possible learning improvement obtained by executing the real-world project. Anyway, should we be able to repeat the procedures, we would include a second written exam;
- The effort to organize a project such as the one proposed in the preceding sections is huge. The course supervisor must have the time to select a proper project, the time required to evaluate the deliveries sent by students (since these will be the basis for further assignments, correction should occur as soon as possible), the time to enhance the best solution into a reference implementation, and the time to identify and filter incoming events related to the problem domain. All these activities add up to class preparation, preparation and correction of exams (in our case, a single exam), preparation of supporting exercises, and so on. So, the supervisor must (a) be prepared to spend more time during course execution than simpler assignments would require; and (b) put the hands on the job to prepare the reference implementations, even in a scenario where none of the teams of students deliver a good solution for a given assignment;
- Due to the aspects presented above, we do not believe that the strategy used in the PPGI017 course escalates to be directly applied in a larger scenario. It is important to notice that we had only 6 students who could be easily organized into 3 groups. It might be difficult to run the same procedures in a course with, for instance, 40 undergraduate students;
- Though the cost to prepare and run the course is huge, the ability to reuse course materials and assignments for a new offer in the future is somewhat limited. Since the real project is temporary (it should be, at least), its related events and news are supposed to end in a given time frame. So, following such a period, the project (along with its related materials) will become a past experience, and, therefore, will lose some of its relation to the real world, though it will remain a concrete problem that was addressed by the industry: it will still be a better assignment than a generic, abstract project.

Finally, due to the success of using the real world example as a course assignment, and trying to deal with the cost of replication, we are now interested in evaluating the feasibility of using a virtual environment, such as Second Life (LINDEN LABS, 2008), to simulate the real world, along with events and news about a simulated industrial problem.

6 CONCLUSIONS

Information Systems is not a consolidated research area in Brazil. Its multidisciplinary nature - involving Management, Computer Science, Production Engineering and Social Sciences - and its constant need for practical application make it hard for it to be recognized as a research area on its own. The PPGI-UNIRIO is associated to the Brazilian Computer Science research committee and subjected to its regulations about research productivity. In this context, national research and graduation in CS is characterized by graduate programs where IS appears as just one among many other CS themes. PPGI-UNIRIO, on the other hand, embraces IS as the essence for its graduate program, drilling down research lines from major aspects and observed problems of IS.

In order to change this scenario and bring IS to attention within the CS community, the following actions have been outlined by the IS Brazilian research community (CIDRAL *et al.*, 2008): a) define an IS research reference, summarizing the main research topics as well as allowing flexibility for researching in different and evolving domains, subjected to technological and social changes; b) be open to multi/inter/ transdisciplinarity, defining different levels of relevance and rigor needed for research methods in the IS area; c) develop competencies in both quantitative and qualitative research methodologies, resulting in relevant contributions to the research area and to the society; d) apply research results to industrial, governmental and society problems, in an integrated way; and e) describe the profile of the usual IS graduated professional, showing its alignment to the needs of the industry and the market.

IS programs should enforce IS research characteristics into their activities, such as applied research, multidisciplinarity and intersections with Management and Social Sciences. However, each one of these activities involves specific challenges. Methodologies for social and applied research, such as action-research are not yet fully understood as scientific research methods, especially in the Computer Science field. Multi-disciplinarity, although providing extremely interesting research results, faces difficulties to have its value acknowledged, especially concerning publications.

Lastly, education on information systems should strive to present students with assignments and case studies close to real world problems. This is a great distinction between information systems and computer science, since the latter does not display the permanent need to be applied and thus can use abstract examples to highlight its specific issues of interest.

The actions outlined above are also challenges embraced by PPGI-UNIRIO to open a path for bringing the CS and IS Brazilian research communities closer together. Establishing a common research language and stimulating publications and participation in both communities are additional strategies to achieve greater collaboration. In PPGI-UNIRIO there is special funding for researchers that publish in important IS conferences such as ICIS, AMCIS, as well as in Brazilian IS forums. Additionally, students in PPGI-UNIRIO must attend a regular course on Scientific Methods/Theory Construction where specific IS research methods are presented and discussed, so as to be applied into their research work. Another desired possibility for improving the interdisciplinarity is to include researchers from different IS sub-areas (Business and Informatics, among others) in the evaluation boards of theses and dissertations produced in the area.

REFERENCES

ACM Computing Curricula. Available at: http://www.acm.org/education/curricula-recommendations. Accessed: 17/Nov/2010

BIS - Bank for International Settlements. Basel II: revised international capital framework. 2009. Available at: http://www.bis.org/publ/bcbsca.htm. Accessed: 15/Oct/2010.

BIS - Bank for International Settlements. International convergence of capital measurement and capital standards: a revised framework (comprehensive version). 2006. Available at: http://www.bis.org/publ/bcbs128a.pdf. Accessed: 15/Oct/2010.

Brazilian Central Bank. Comunicado Nº 12.746. 2004. Available in Portuguese at: https://www3.bcb.gov.br/normativo/detalharNormativo.do?N=104206982&method=detalharNormativo. Accessed: 15/Oct/2010.

CIDRAL. A, RALHA, C., CÁCERES, E., SANTORO, F., AUDY, J., BARROS, M., FURTADO, O., ARAUJO, R., Sistemas de informação no Brasil – desafios e oportunidades [Information systems in Brazil – challenges and opportunities]. Available in Portuguese at: http://sbsi2009.cic.unb.br/atas.html. Accessed: 15/Oct/2010.

Computing Curricula. Computing Curricula – The Overview Report. 2005. Available at: http://www.acm.org/education/curric_vols/CC2005-March06Final.pdf. Accessed: 15/Oct/2010.

DIA-UNIRIO, Information Systems Department – UNIRIO. Website. s.d. Available at: http://www.uniriotec.br. Accessed: 15/Oct/2010.

LEHMAN, M. Laws of software evolution revisited. *Lecture Notes in Computer Science*, v. 1149, Springer Berlin / Heidelberg, 1996.

LINDEN LABS. Second Life: official site of the 3D online virtual world. 2008. Available at: http://www.secondlife.com. Accessed: 15/Oct/2010.

McCONNELL, S. *Code complete: a practical handbook of software construction*. Redmond, WA: Microsoft Press, 1993.

MOTTA, E.N., Construção de uma base de conhecimento da música popular brasileira utilizando mineração em textos [Building a knowledge base on Brazilian popular music using text mining]. In: Simpósio Brasileiro de Sistemas de Informação, 4, Rio de Janeiro, *Proceedings...* SBC, 2008.

NP2TEC - Núcleo de Pesquisa e Prática em Tecnologia. Website. Available at: http://www.uniriotec.br/~np2tec. Accessed: 15/Oct/2010.

PFLEEGER, S.L. *Software engineering*: theory and practice. Upper Saddle River, NJ: Prentice-Hall Inc., 1998.

PPGI-UNIRIO. Graduate program in information systems – UNIRIO. 2009. Available in Portuguese at: http://www.uniriotec.br/ppgi. Accessed: 15/Oct/2010.

SANTIS, R. Sistemas de Informação para a representação de conhecimento e análise semiótica de obras musicais [Information Systems for knowledge representation and semiotic analysis of music artifacts]. In: Simpósio Brasileiro de Sistemas de Informação, 4, Rio de Janeiro, *Proceedings...* SBC, 2008.

SBC, Currículo de referência da SBC para cursos em computação e informática [SBC's reference syllabus for computer and informatics courses]. 2003. Available at: http://www.sbc.org.br (in Portuguese). Access: 15/Oct/2010.