

# Facing the Challenges of Teaching Requirements Engineering

Roxana Lisette Quintanilla Portugal<sup>1</sup>, Priscila Engiel<sup>1</sup>, Joanna Pivatelli<sup>1</sup>, Julio Cesar Sampaio do Prado Leite<sup>1</sup>

<sup>1</sup>Departamento de Informática, PUC-Rio, Rio de Janeiro, RJ, Brasil

{rportugal, pengiel, jpivotelli}@inf.puc-rio.br, www.inf.puc-rio/~julio

## ABSTRACT

This paper reports on our experience of teaching Requirements Engineering for undergraduate students. It is well known, the obstacles educators have in teaching requirements engineering. These obstacles are related to the very nature of requirements engineering: a multidisciplinary field that deals with both computer science and social sciences concepts. Teaching requirements engineering just with problems descriptions, as a basis for the construction of requirements specifications or requirements models, misses the point. Educators should also provide students with ways of gathering client information. However, to be effective in this regard, there is the need that students interact with clients. Our pedagogical strategy is designed to tackle these challenges. Notwithstanding, we need to have feedback about the strategy, which lead to the design of an assessment to gauge the efficacy of our pedagogical strategy. We, describe the strategy, stress its novelty in facing the challenges, and provide assessment results over 3 semesters.

## Categories and Subject Descriptors

K3.2 [Computer and Information Science Education] D.2.1 [Requirements/Specification]

## General Terms

Requirements elicitation, requirements engineering education, creativity, pedagogic strategy.

## Keywords

Teaching Requirements Engineering, Project-Based Learning, Role-playing Learning, creativity, collaboration, communication, coordination, cooperation.

## 1. INTRODUCTION

Requirements Engineering (RE) has been, since its creation, concerned with its interdisciplinary characteristic; an example is the work of Sommerville et al. [1]. Requirements elicitation strategies are usually derived from social sciences [2], and this

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [Permissions@acm.org](mailto:Permissions@acm.org).

ICSE '16 Companion, May 14-22, 2016, Austin, TX, USA  
© 2016 ACM. ISBN 978-1-4503-4205-6/16/05...\$15.00  
DOI: <http://dx.doi.org/10.1145/2889160.2889200>

can be a barrier for people with a computer science background.

RE courses have the difficult task to design an approach to convey RE concepts, which must encompass practical experience. Literature reports on strategies that are able to pair students with industry [19] [20], but attaining this setting is non-trivial. Role-playing is a collaborative strategy applied in many teaching strategies due to resource, financial, and time constraints [3][4] [28]. In these strategies, different roles are to be assumed by students and instructors in order to enact practical experience.

Another problem discussed in RE courses is concerned with reliance on problem description, as noted by Sikkel and Daneva [5], “in our educational system it is institutionalized that students get a 'perfect' problem description”. This approach hinders important aspect of requirements engineering education, mainly those related to elicitation, which involves collaboration and different types of dialogues. As such, practical experience is key to conveying the theoretical aspects of elicitation. Sedelmaier and Landes [6] also mention this problem: the “on the silver tray” misconception usually held by students regarding requirements.

In this paper, we explore a novel role-playing approach using groups of undergraduate students and instructors (lecturer and graduate students). Instructors, besides lecturing, play the role of consultants. Whereas each group of students plays the role of three types of companies: a startup, a requirements construction, and an auditing one. We describe the overall goals of our strategy at Section 2. Section 3 focus on our role-playing problem-based approach. Section 4 explains the assessment performed using several elicitation techniques. Section 5 situates our strategy with related work. We conclude at Section 6 comparing our assessment with our pedagogical goals.

## 2. GOALS

The goals we want to achieve are summarized below (Table 1).

These goals are set around the 5W2H<sup>1</sup> as to address the issues of elicitation (G1, G2), modeling (G2), analysis (G3), and management (G2, G4) in requirements engineering. In particular, with G1, we address the difficulty that clients have in defining what is needed. This goal is important for students, as they need to understand the intricacies of elicitation (“silver tray” [6]), whereas the completeness fallacy, the existence of different viewpoints, and the lack of well established needs permeates the task of understanding what is to be built. With G4, we address management with an emphasis in collaboration both within the requirements team, but also with clients and consultants.

<sup>1</sup> <http://amazonngg.blogspot.com.br/2006/05/what-is-5w1h-or-5w2h-framework.html>

**Table 1 Pedagogical Goals**

GOALS	
1	Students should learn the difficulties of being a client, that is, having trouble to define <i>what</i> they need as a product.
2	Students should learn to elicit <i>where</i> , <i>when</i> , <i>what</i> and <i>why</i> the clients ( <i>who</i> ) needs, experiencing with real situations, and using requirements techniques to define <i>how</i> and for <i>how much</i> they can build the product (the requirements).
3	Students should learn about quality control, experiencing the verification process using inspection ("auditors").
4	Students should learn how requirements teams work, experiencing how to manage a project, defining <i>when</i> the activities will be developed, <i>who</i> will be responsible for each activity, and <i>how much</i> effort will be demanded. Students should also learn how to interact with consultants.

To reach these goals the undergraduate course has 60 hours in a semester (2 classes of 2 hours per week). We allocate these hours in three parts: traditional lectures with weekly group exercises, a project-based assignment, and assessment. As such, our strategy has to deal with different resources and different pedagogy techniques. We detail these in the following Section.

### 3. THE STRATEGY

Our strategy is influenced by the concept of Pedagogy Transparency [7], which believes that: "... transparency seeks an environment where goals are open and teaching methods aims consensus by focusing on learner participation and feedback arising from his/her participation." The works of Freire [8] and Gal'perin [9] have influenced this concept.

The course has three parts: traditional lectures, project-based learning, and feedback assessments. This course has been applied in three semesters: 14-1 (27 students), 14-2 (10 students) and 15-1 (20 students) with one senior instructor and four assistants. The course is offered to junior/senior undergraduate students, in either computer science, computing engineering, or in information systems. The instructor delivers the classes and handles the exercises. The assistants, graduate students (some with industrial experience), act as consultants during the project and handle the assessment. The instructor also acts as a consultant.

Grading is based on the exercises, on the individual exam after the lecture part and on the final project document. Grading is independent of the assessment part.

### 3.1 LECTURES

Considering project-based approaches limitations, Mills and Treagust [10] say, "many topics must be learned in a certain order, because missing essential parts will result in failure to learn later concepts. This problem will be hard for a student to correct, no matter how good their metacognitive skills". We concur with this consideration. To tackle the problem, our approach gives lectures in the first half of the course (8 weeks)

Lectures use two supporting materials: an on-line book [11] and a blog [12] with the syllabus, the schedule and annotations for each week of class. Becker's [2] book is also used to support the elicitation part. Each week the students, as a group, have to do an exercise based on the material given in class. An example of such exercise is using the tool Atlas-Ti [13], or building i\* models [18]. Lectures cover requirements elicitation; requirements modeling, requirements analysis, and requirements management (see Section 2).

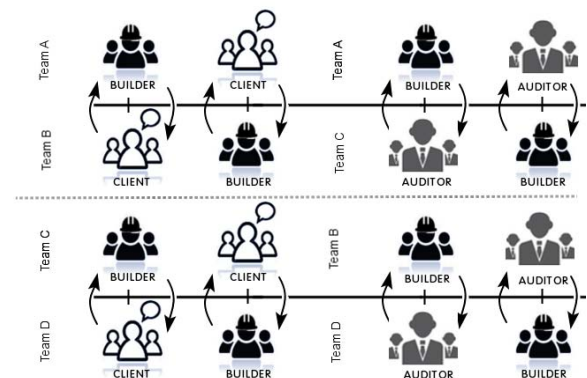
## 3.2 Project-Based Learning

In the second part of the course (8 weeks), lecture time is used by teams to work together and/or with consultants mentoring the project. These sessions start with general directions about what need to be done, including team formation. Students organize their own teams, within a limit (max (6), min (3)) of students per team. Teams will play three roles: client, builder, and auditor. Each team's role corresponds to one company. So, if a classroom has 4 teams (as the 14-1 class), there will be 12 companies. Each team should provide one executive summary per company with name, portfolio and staff. Figure 1 shows the interactions in the case of 4 teams.

Each client team has to come up with an idea that will demand software support. The instructor gives an overview of the concept of Startup [14] and encourages the students to have ideas that could be a future business. These ideas would be the drive for the client team<sup>2</sup>, who will need a software system to support the business. The client team has to produce an initial document (one page) describing the company and the desired product, this document is free format, no templates. The ideal situation is that a team could see this exercise as a real opportunity to have an idea of a killer application. As such, creativity techniques, as noted by Grube and Schmid [17], play a role in this context.

Each builder team has to elicit the needs of the client team and build the requirements for the software that will support the ideas of the client team. Audit teams act in the quality control part of the project by inspecting [15] the models produced by a builder team.

The interactions among the teams (Figure 1) is fundamental for the strategy. Therefore, during a class with 4 teams, one team will have the opportunity to work together with two different teams. In Figure 1, for instance, Team A is the builder for Team B and Team C is the auditor for Team A. As such, there is an emphasis on collaborating with other groups of students, which helps the achievement of G4 (Table 1).



**Figure 1 Interaction among Builders, Clients, and Auditors**

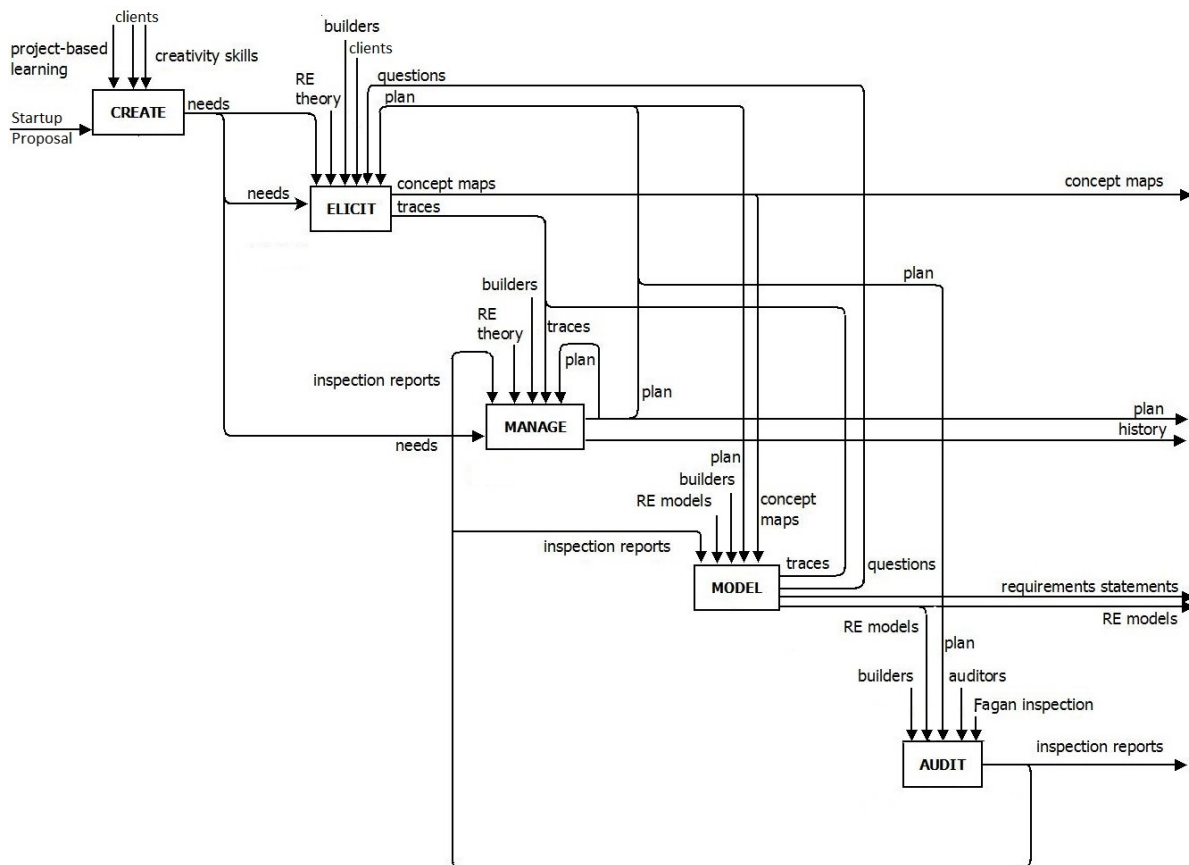
The project-based process is summarized by the SADT (Structured Analysis and Design Technique) model [16] built with the objective of describing the process from the viewpoint of the educators. The SADT diagram (Figure 2) is composed by the activities of CREATE, ELICIT, MANAGE, MODEL and A AUDIT<sup>3</sup>. Stakeholders (clients), constrained by project-based learning and creativity skills CREATE, producing needs. Builders

<sup>2</sup> Defining a Startup business is the core strategy to achieve goal G1.

<sup>3</sup> The activities are aligned with the goals (Table 1)

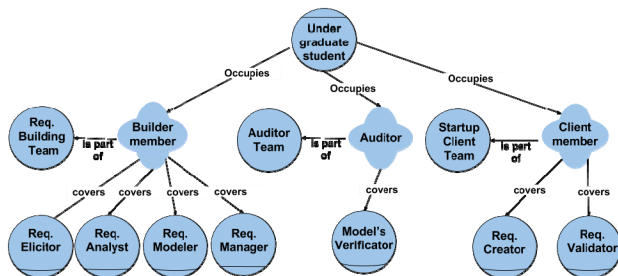
and Stakeholders consuming needs and constrained by RE theory, question and plan ELICIT, producing traces and concept maps. Builders consuming needs and using inspection reports, RE theory, traces and plan MANAGE, producing plan and history. Builders constrained by inspection reports, concept maps, plan

and RE theory MODEL, producing traces, requirements statements and RE models. Auditors and builders constrained by plan, RE models and Fagan inspection AUDIT, producing inspection reports.



**Figure 2 SADT- The Overall Project-Based Learning Process**

To better explain roles and positions that a student (agent) passes through during this part of the semester, we model the situation as an i\* actor diagram [18].



**Figure 3 i\* actor diagram**

There are three main actors in our approach: the instructor, undergraduate students, and graduate students. Undergraduate students' positions (Figure 3) are: Client member who is responsible to idealize the system or mobile app needs; Builder member who is responsible for building the requirements through eliciting, modeling, analyzing (V&V) and managing tasks, with the aim to create the requirements document.; and Auditor who is

responsible to verify models through checklist inspection. The bottom of Figure 3 shows the roles covered by each position. This actor structure is central to achieve the goals (Table 1), since students will need to play those roles, and each role is related to the capabilities needed for a requirements engineer.

Instructor and graduate students play the role of consultants, who provide guidance to builder and auditor companies.

As a customer, the Startup Client Team has to explain, as best as possible, the high-level needs they had for their app or software product to the builders. The Client Team strives for something new and innovative. We, as consultants, perceived that they freely selected the following creative techniques [17]: brainstorming, different perspectives and random input.

On the other hand, the builders company (Req. Building Team) produces a description of its profile, and the expertise of its members, to sell them as a company. Then, they were asked to consult several sources of information for the startup requirements, and to create minute's reports when information was gathered from clients. After an initial contact with the Startup team, the builders proposed an overall plan with milestones and a budget proposal. This planning should contain a task-oriented

plan for the remaining 8 weeks. After the client approval, the project started, following the plan. Builders were also asked to freely choose the techniques taught in the first part of classes. They have to use three elicitation and three modeling languages. Modeling could be performed after the basic facts are elicited, or may be in parallel with elicitation. During the process, they were asked to keep the trace of artifacts and to produce meeting minutes, which will be used for tracking the evolution of knowledge. After models are available, the builder team hires the audit company to check the models. The report produced by the audit team will be used as feedback as to improve the builder's models.

In the analysis activity, the builder teams practiced verification strategies, with revisions. The audit company performed inspection using Fagan inspections [15] with preparation, reading, meetings, rework, and reassessment. Fagan inspection of models was based on pre-defined checklists elaborated by the instructor. Finally, they were asked to produce the list (following the IEEE standard) of functional, non-functional and inverse requirements [11] for the required product.

During the process, consultants provide advice as needed: helping with templates of documents (if asked), revising and proposing improvements in the artifacts, helping with the doubts or to validating the work being done. Each team had a consultant, who besides assisting the team also uses observation techniques to gather facts about the process.

The activities (Figure 2) explain the workflow performed in this part of the course. Elicitation used techniques such as meetings, questionnaires, interviews, observation, documents reading, reverse engineering, and reuse. In order to perform the modeling activity, the students could choose, among others, the following: scenarios, language extended lexicon, entity-relationship model, data flow diagram, SADT, and i\*.

### 3.3 Packing the strategy

To facilitate the reuse of the strategy, we listed the overall activities that actors (students, graduate students, and instructor) have to play in enacting the strategy. A list of 14 items is packed in four parts: plan, do, control, act (PDCA<sup>4</sup>). **Plan:** 1) Define teams (companies), 2) Explain roles, 3) Define and explain the project-based strategy, 4) Pair consultants with companies, 5) Create a business idea (that needs software), 6) Choose elicitation techniques 7) Choose model languages, **Do:** 8) Elaborate schedule with activities, 9) Organize information in a proposal, 10) Execute the schedule, **Control:** 11) Manage the activities in schedule (track), 12) Execute verification (reading, meeting), **Act:** 13) Prepare a list with defects for model rework, 14) Prepare the final document to be delivered to clients.

We believe that this list may help other courses inherit some of our ideas, making the proper adjustments with respect to each particular context.

## 4. ASSESSMENT USING ELICITATION TECHNIQUES

The third part of the course is the assessment part. This is another way of providing feedback [19] [20] to the stakeholders involved,

and improving our pedagogic approach. During the application of the strategy over time, we evolved our assessment incorporating more elicitation techniques, as to improve our fact gathering. Table 2 maps the techniques and the semester they were applied.

For each of the following Subsection, we analyze the results of each of the fact gathering used. The results are shown for each semester. As the questionnaires had open and closed questions, we have used quantitative analysis, and, for further explanation, qualitative ones.

### 4.1 E-mail Questionnaire

The e-mail questionnaire was developed in our first experience with the strategy. During 2014-1, we as graduate students and the instructor built this questionnaire according to our observations as consultants, and applying a collaborative approach 3C [21].

The e-mail questionnaire [24], a set of quantitative and qualitative questions, was provided to students via Google Forms. We consider this method an efficient way to reach the students, easily and quickly as to collect the most answers as possible.

At the end of all semesters, and after the instructor gave the final grade, an e-mail was sent inviting students to give us impressions about the pedagogical approach. The e-mail gently invites students to answer the questionnaire stressing that their participation was not mandatory. We did not track answers; we designed the questionnaire and its application as to be impossible to identify respondents.

At first, it was difficult to collect answers of classes, but we kept sending more e-mails trying to motivate students to collaborate with the consultants. We provided a link to our questionnaire and gave them a deadline to reply. For three times we extended deadline to obtain more answers. In all classes, we got answers for about half of participants, with the following ratios (students/answers): 14-1(27/22), 14-2(10/5), and 15-1 (20/9).

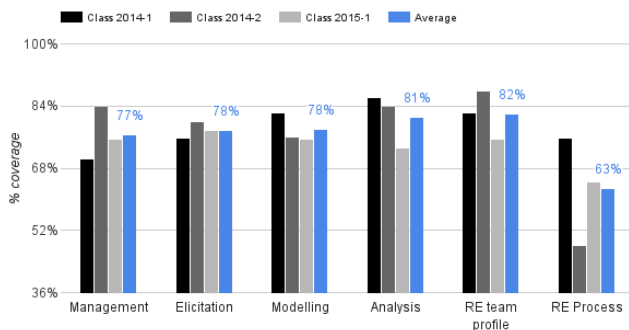
The aim of this questionnaire is to catch the student's point of view about the role-playing approach, issues and obstacles they faced. We designed the questionnaire with 14 quantitative questions related to assuming different roles, and 2 qualitative questions about weakness and strengths of the approach. The questionnaires as well as the results are available at [24]. The questionnaire was designed to be measured using a Likert scale [22]. In a question, each possible response is a statement valued in the range from very poor to excellent. Only one answer is allowed and the respondent has to choose the best choice that fits his comprehension. Quantitative questions are divided in four clusters: (1) 6 questions about the builder role, (2) 4 questions about the client role, (3) 2 question for the auditor role, and (4) 2 questions about management. In the scale from 1 to 5 points, we valued each answer with the values of Likert scale: answer 1 is 1 point; answer 2 is 2 points until answer 5, it is mandatory to choose only one, which the respondent thinks better **satisfices** [23] the question.

<sup>4</sup> <https://en.wikipedia.org/wiki/PDCA>

**Table 2 Map of the Requirements Elicitation techniques used for gathering facts for assessment**

Classes	Self-Assessment as Teams	Assessment of Final Report	Consultants Observations	Consultants Interviews	Self-Assessment as Students		Product Validation Questionnaire
	E-mail Questionnaire	Document Reading			Initial Questionnaire	Final Questionnaire	
2014-1	yes	yes	yes	no	no	no	no
2014-2	yes	yes	yes	yes	no	no	yes
2015-1	yes	yes	yes	no	yes	yes	yes

For the analysis, each question was valued by coverage: if all students marked 5 in a question, the coverage means 100%. On the other hand, the 1 scale means 0%. The Likert scale should be viewed as agreement-disagreement scale<sup>5</sup> and the mapping of coverage depends on how it was applied, we choose a percentage range (100;0), with the intervals 100, 75, 50, 25, and 0. Having the coverage for each question, we created bar charts, one for each cluster, comparing the results of the 3 semesters.



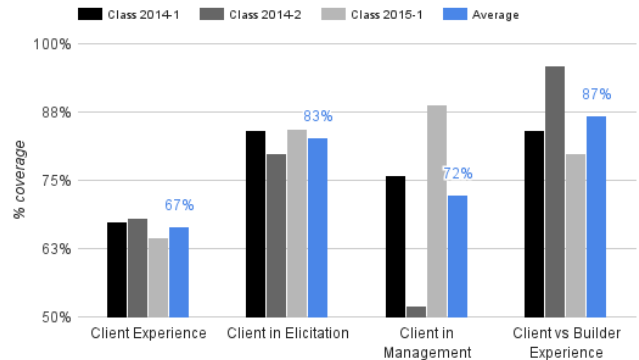
**Figure 4 Self-Assessment as a Requirements Builder Team**

For the first group of questions related to the builder's performance, we got results between 3 to 5 Likert points. However, the lower values for class 2014-2<sup>6</sup> in RE process draws attention. We looked for details in the questionnaire results, and this item was related to the 6th question: "How your group performed the construction of requirements?" an average value of 48%, equivalent to 3 Likert points, pointed out that most students marked the option 3: "1<sup>st</sup> Elicitation, 2<sup>nd</sup> Modelling, 3<sup>rd</sup> Inspection". From this, we get that students did not perceive that "requirements tasks can be overlapped", option 5.

Figure 5 relates to client performance; there, Client Experience and Client in Management draw our attention. In the first case, the corresponding question: "How did you understand the client role?" got answers with an average of 4 Likert points: "It helps to understand the elicitation process". In the second case, for 2014-2, the question: "How do you describe the experience of being a client in the management tasks carried out by the builders?" got an average answer of 3 Likert points: "the meetings were held with the presence of different people, and many questions were repetitive", where the 5 point option was: "all members of our group had an overall clear idea of our demands, such that anyone could answer questions posed by builders in meetings".

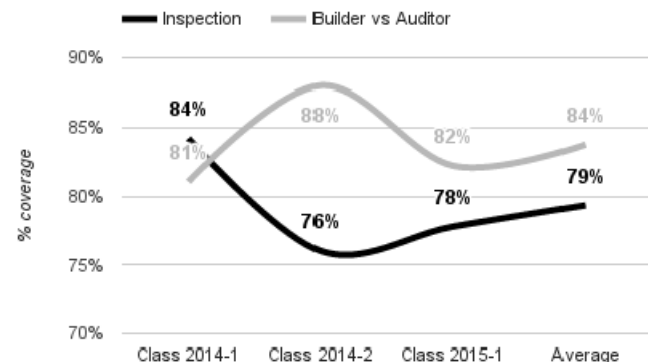
<sup>5</sup> "When responding to a Likert questionnaire item, respondents specify their level of agreement or disagreement on a symmetric agree-disagree scale for a series of statements. Thus, the range captures the intensity of their feelings for a given item" ([https://en.wikipedia.org/wiki/Likert\\_scale](https://en.wikipedia.org/wiki/Likert_scale))

<sup>6</sup> Class of 2014-2 was smaller (10 students, 2 teams) and with less interaction with consultants.



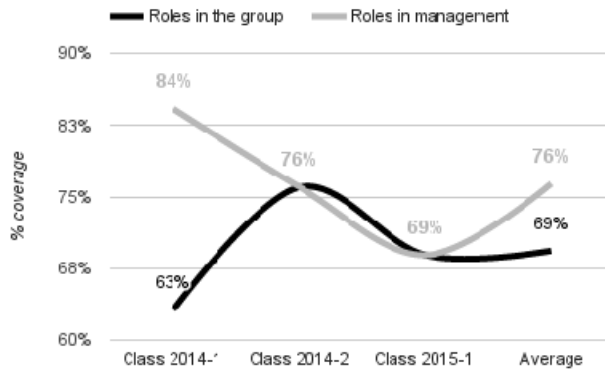
**Figure 5 Self-Assessment as Client Team**

For the third cluster (Figure 6) related to Auditor performance, one value draws our attention, the 76% in Inspection for the class 2014-2. Even with a positive score, the question: "How do you believe the inspection task helped in the understanding of quality?" got an average of just above 3 Likert points: "We understand that the inspection helps in quality". We found this is an acceptable response since our higher expectation, a 5 Likert points option was: "We used the inspection results to improve our models".



**Figure 6 Self-Assessment in Quality Activities**

The last cluster is related to performance of Roles in the group, and Roles in the management activities (Figure 7). A closer look in Roles in management shows the question: "How your group used the management functions?" with 84%, 5 Likert points with the answer: "We followed the planning, upgrading and creating traces between documents", and 69%-76%, 4 Likert points, with the answer: "There was distribution of responsibilities and followed the planning". However, there are no results lower than 3 Likert points, which relates to the answer: "We created minute notes for all meetings".



**Figure 7 Self-Assessment of Roles in Management**

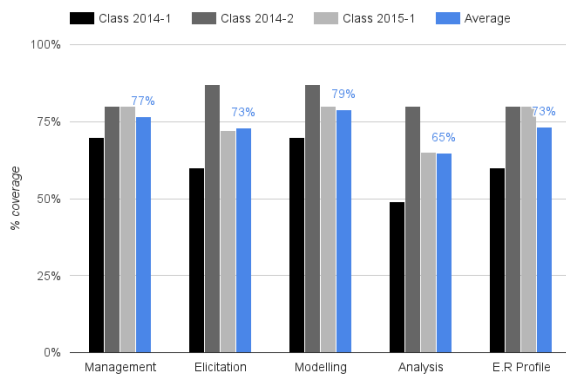
## 4.2 Document Reading

Document reading is an overall technique for elicitation, used when the information source is a document [11]. Different instantiations of the reading technique may be applied depending on the context. In our case, we have a series of documents produced by each group, packaged as the final report each builder team presents. Each report is comprised of minute's notes, documents of planning, models produced, mockups, map minds, questionnaires, and interviews.

We performed the reading guided by a checklist of eleven concerns. This checklist is aligned with the e-mail questionnaire, with the intention to compare those results but from another point of view. That is, of readers of the final report.

At the end of each semester, the instructor and one of the graduate students assess the teams' performance as Builders, Clients, Auditors, and also how well the final document is formatted (presentation). The results are available at the link [24]. The process of reading each report, combined with the assessment process, took, in average, two hours.

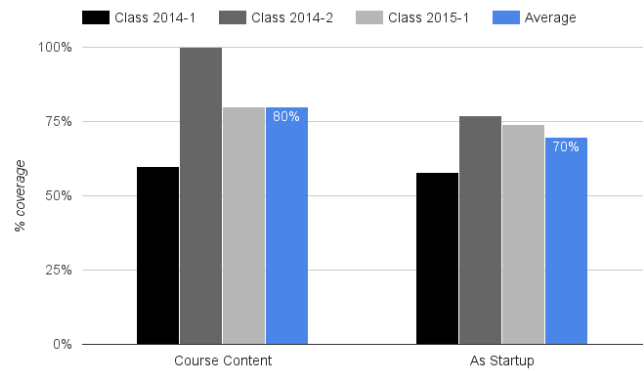
The first cluster of questions (Figure 8) relates to Builder performance: there, the Analysis task is the one with lower average, but increasing from the first data set (2014-1). Class of 2014-2 got better averages, which may be related to being the smallest one.



**Figure 8 Assessment of builders**

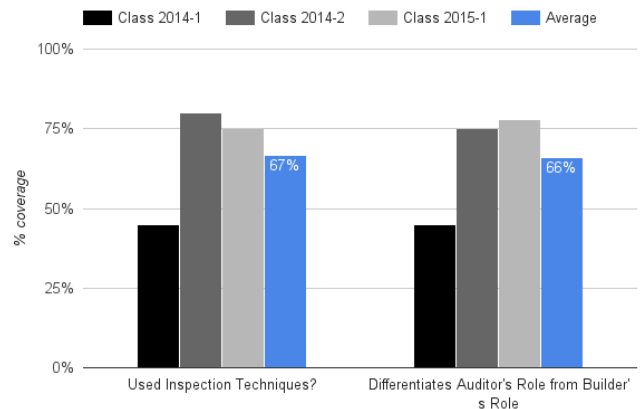
In the cluster of questions for client's assessment (Figure 9), we got lower values for clients acting as startups, but these values improved from the 2014-1 semester. For us, this means that acting

as real clients is a challenge. With respect to course content, the class of 2014-2 delivered a good description of the client companies.



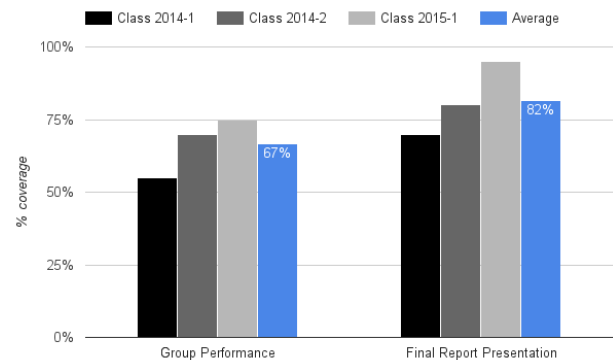
**Figure 9 Assessment of Clients**

In the Auditor's assessment cluster (Figure 10), we see an improvement since the first class, where the inspection checklists were applied just in the last sessions and students did not have proper time to change their models. The improvement reflects the treatment we gave to the feedback, structuring the use of inspection during lecture and in the project-based part.



**Figure 10 Assessment of Auditors**

The last cluster (Figure 11), related to management, also shows that we (instructors) are using the feedback to improve the strategy. One of the ways we achieved this, is making our expectations more transparent [7], and suggesting the teams to make their final projects available on the Web.



**Figure 11 Other aspects Assessment**



Finally, we compared the grading of the eleven concerns for each class with each class average final grade given by the instructor to the project, as a sanity check. There were no noticeable variations (Figure 12).

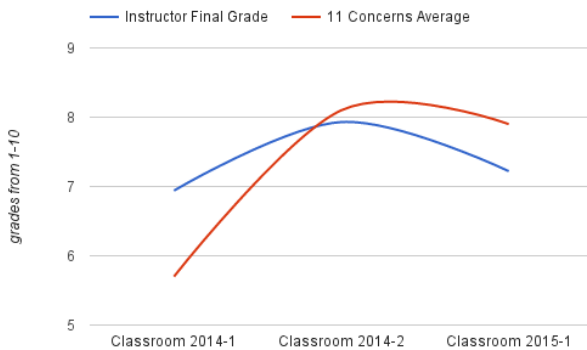


Figure 12 Average Grading

### 4.3 Consultants Observation

Each semester we took a different approach towards observation. In the class of 2014-1, as consultants, we recorded in a private blog our impressions of each team we belonged. For the class of 2014-2, we wrote our observations while performing the interview in one of the last sessions of the semester. For the class of 2015-1, we again participated more actively as consultants but without a group assigned, it was random according to our availability. Consultants were told by the instructor to write our observation every week but without observing what the other consultant wrote; then in our final session, we read all the observations. This last observation activity is part of a tech report, which is in progress while writing this work.

### 4.4 Consultants Interviews

In the class of 2014-2, we designed an interview to cope with some doubts we had from our first experience. From our observations recorded in the blog, we extracted the most common concerns. (1) We wanted to know if not having a defined problem statement was an issue for students; (2) we were concerned if the number of participants in each team was a problem, and (3) how the teams dealt with the workload. In general, the interview answers were positive; they said they felt more freedom proposing their own problem statement. However, about the concern (1) some declared, *“the statement was created as client requested features”*, *“I had to miss a lot of classes, leaving the responsibility to the group, then they had to share the information”*, *“It is difficult, it generates misunderstandings on parts of the work”*. As the interview had open questions, we pretend to go further on the qualitative analysis of the answers.

### 4.5 Self-Assessment of Students

For the 2015-1 semester, the instructor applied a questionnaire at the first and at the last lecture. The questionnaire had to be marked by a personal code that students had to remember until the end of the semester. This was a way of pairing questionnaires without identifying the student. We managed to pair 11 questionnaires.

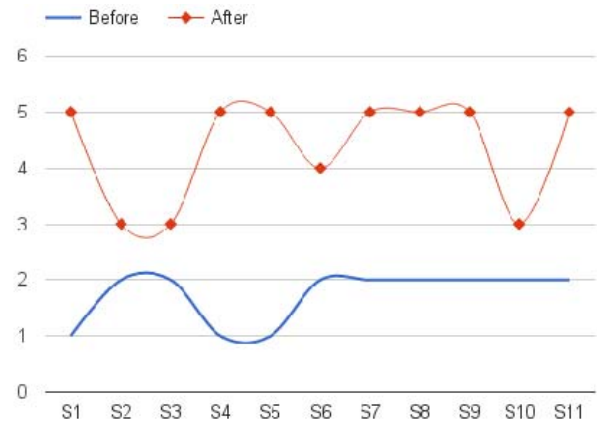


Figure 13 Question: What is your knowledge level in Requirements Engineering?

Based just in the quantitative questions, we see that most of the students improved considerably. However, 3 of them were outliers (Figure 13). Searching in the qualitative answers, we found answers that justify the outliers. Table 3 summarizes the findings.

Table 3 Impression of the students before and after class

Stu- dent	Which are your expectations with the course?	Have your expectations for the course been met?
S2	<i>“To have the necessary theory to deal with the elicitation and modeling of requirements”</i>	<i>“I expected a more intensive teaching style, with more emphasis on the concepts and more extensive and specific material”</i>
S3	<i>“I do not have much in mind”</i>	<i>“So so. I learned something, but I believe that the course lasted beyond the expected”</i>
S10	<i>“learn techniques of requirements engineering”</i>	<i>“Yes”</i>

Regarding the second question (Figure 14), we learned that the majority of students learned requirements elicitation techniques.

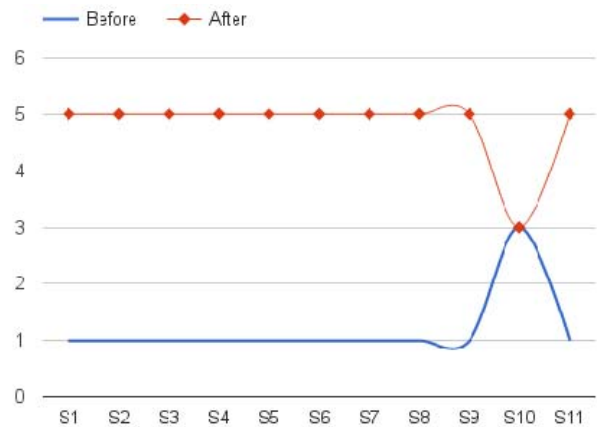
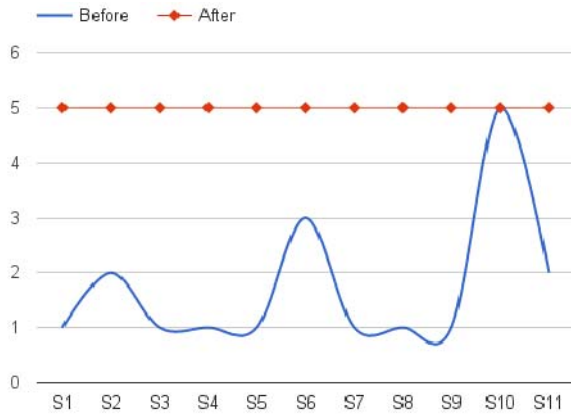


Figure 14 Question: Do you know how to use Requirements Elicitation techniques

To last question (Figure 15) we perceived that for some of the students, which declared knowledge in modeling at the beginning, there was no considerable improvement.

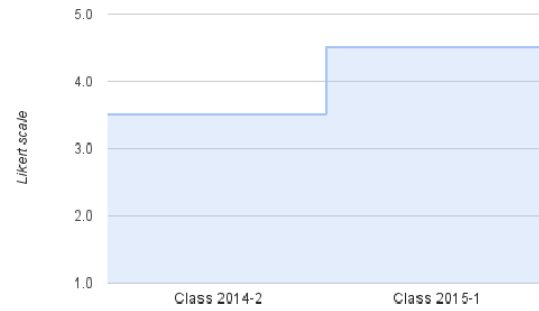


**Figure 15 Question: Do you know how to use modelling languages?**

#### 4.6 Product Validation Questionnaire

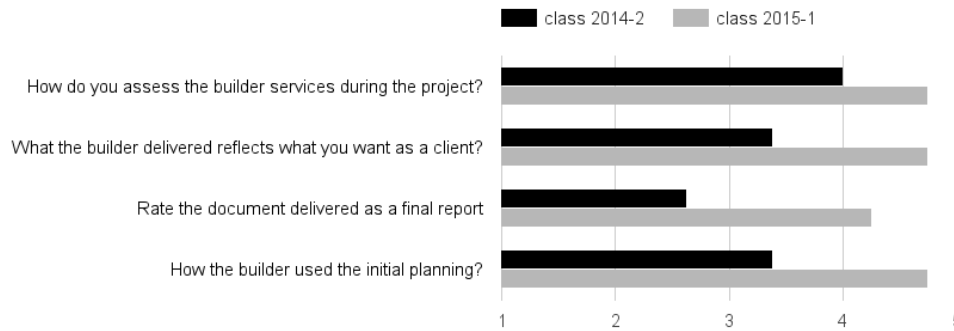
After the first semester experience, we figure out that students end the semester with enough knowledge as to judge the work of the team they hired. Then, we created a questionnaire [24] so students, when receiving the product (requirements document) from the builder could assess their satisfaction as clients.

The result (Figure 16) shows that the group of 2014-2 was more critical, we assume this happened because they were just two groups in this class, so they got more time with the instructor.



**Figure 16 Average of Client Assessment**

We asked more questions to evaluate the individual builder services (Figure 17). This details the assessment and shows an increase on client satisfaction from one semester to the other. We assume that groups of 2015-1 were most satisfied than groups of 2014-2 because classes in 2015-1 had more groups than in 2014-2. With more students in classes, they could interact among themselves, observe other clients groups and improve their position as Client Member (Figure 3).



**Figure 17 Client Validation of Builders Product**

### 5. RELATED WORK

Teaching Requirements Engineering using a non-traditional approach has a relative recent interest in RE, and articles from an industry viewpoint demands a RE profile to fit into the RE practice [23] [26]. As stated before, the main challenge is providing students with a reasonable understanding of what they may face in real projects.

Polajnar and Polajnar [4] cite two ways to create a realistic learning experience in requirements elicitation: with virtual and with real clients. Virtual stakeholder consist in use a non-present person or group who have, or not have, software engineering background and are usually acquainted with the application domain. Real clients consist in bring to course real needs of a real problem.

Zowghi and Paryani [27] were among the first applying role-playing as pedagogical tool for teaching RE through problem

solving. They focused in tasks like elicitation, analysis, modelling, validation, specification and management of requirements in a way to give skills that student will need in RE practice. However, their approach differs from us in their lecture classes (content), which is in parallel to tutorial classes (real-world problem), and with respect to assessment being used as feedback for both students and instructors. In our strategy, we focus the half of the semester in traditional lectures with exercises and the last 8 weeks, we use the role-playing strategy with the Startup projects. We also believe that forming a company and not just a team provides more similitude with real projects.

Liang and Graaf [28] applied role-playing for undergraduate students and their approach is very similar to ours, with builders, clients and auditing tasks. However, they worked with a wiki to construct requirements specifications during classes and collaboratively. In our strategy the requirements specification is built as the final artifact of the role-playing.



From Mills and Treagust Work [10] they ask themselves, if a project-based learning is successful in engineering. This because, they find this kind of approaches are poorly evaluated. From our assessment, we can say that crossing results from the diverse point of views, as consultants, as interviewers, observant, and self-assessments amongst others, permitted to have a rich feedback with positive results.

Before our experience, Serrano et al. [29] performed a similar work, for the same course in 2008, in which a management point of view was taken. Serrano et al. points out that their assessment has a complexity and uncertainty for the different tasks in a role-playing, and then, they state, "It is worth mentioning that the use of different perspectives and points of view contribute for further completion of the assessment". From this suggestion, we created several questionnaires for different moments of our approach.

The specific use of requirements elicitation techniques for assessment was adopted as a manner to get facts for what is taught and to expose students to these techniques. We found a similar approach in the work of Waters and McCracken [30] for assessment in problem-based experiences.

The work in [27] gives a richer understanding of multiple perspectives of RE and the techniques to apply for each task of RE. From our assessment, we found that students usually have a higher perception of their performance, compared of what we really see in assessment of their artifacts.

From the approach of Liang and Graaf [28] it would be useful to incorporate the bidding of projects, with the aim to stimulate the interest or commitment of the builders with the project, and the boosting of creative clients proposing attractive startup projects.

## 6. CONCLUSION

Comparing the pedagogical goals with the results from the e-mail questionnaire and document reading, we can say that goal #1 "*Students should learn the difficulties of being a client, that is, experiencing the difficulty to define what they need as a product*" **was mainly achieved**<sup>7</sup>. From the e-mail questionnaire where students self-assessed as clients (Figure ), the values of client experience, client in elicitation and client in management, between the three get an average 73%. Those values are aligned with the client as startup (Figure ) with an average of 70%. It means that we compared what they said with what we saw in their reports. As all questionnaires are measured with the Likert scale, an average of 70% means almost 4 Likert points, equivalent to mainly achieved.

For the goal #2 "*Students should learn to elicit where, when, what and why the clients (who) needs, experiencing with real situations, and using requirements techniques to define how and for how much they can build the product (the requirements)*", we can say this **was mainly achieved**, since any value are beyond 3 Likert points. Notwithstanding, from the student point of view (Figure ), they ranked their requirements analysis performance with an average of 81%. In contrast, we observed in their reports (Figure ) an average of 65% for analysis.

The goal #3 "*The Students should learn about quality control, experiencing the verification process using inspection*

*("auditors")*" **was partially achieved**<sup>8</sup>. Students ranked their performance on using inspections techniques, with an average of 79% (Figure ). In contrast, we got an average of 66% for what we saw in their reports (Figure ). For the criteria "Builder vs Auditor", they said they got in average 83% performance. However, we got 66%. However, Figures 8 and 10 show that we are managing to improve students' performance with regard to analysis, and particularly to a better use of inspection.

The goal #4 "*Students should learn how requirements teams work, experiencing how to manage a project, defining when the activities will be developed, who will be responsible for each activity, and how much the effort will demand. Students also should learn how to interact with consultants*", **was mainly achieved**. The students consider they get in average 72%, between roles in group and roles in management (Figure ). Concerning what we assessed in the reports, they get an average of 73%. This goal is both assessments (questionnaire and document reading) are very similar. A 72% and 73% value is very near the 4 Likert points.

Despite the positive results, we have also gathered, from the qualitative fact-finding, some concerns. One is related to the difficulty in creating models. Another is the students' perception that requirements management was well performed, which is not exactly the case.

It worth noting that some students complained about the lack of a problem description ("silver tray"[6]) and that the groups were too large. These two facts showed us that the design of the course was effective. First, because to tackle the elicitation challenge, students have to learn that requirements are not there in written form [5], they have to be elicited in the building process. Second, since it brings up the awareness that, in real projects, there are several stakeholders, so large groups are frequent.

Our approach meets some of the recommendation from industry [10] by teaching notions of budgeting, estimating and defining product features. These tasks are part of the management activities we propose. Our attention to project planning, quality control, company structure, client involvement, and budgeting address most of the issues put forward by Peters and Moreno [31].

Future work will be continuing the qualitative assessment from the data that we have gathered, trying to explain some of the oddities we found in quantitative data. This semester (2015.2), we are testing a different lecture schedule, giving more emphasis on modeling, especially intentional modeling [18].

## 7. ACKNOWLEDGMENTS

Portugal, Engiel and Pivatelli acknowledges the support of Capes. Leite acknowledges the support of CNPq. Reviewers: thanks!

## 8. REFERENCES

- [1] Sommerville, I., Rodden, T., Sawyer, P., Bentley, R., & Twidale, M. (1993, January). Integrating ethnography into the requirements engineering process. In *Requirements Engineering, 1993., Proceedings of IEEE International Symposium on* (pp. 165-173). IEEE.

<sup>7</sup> Our coverage map (100;0) was mapped to: **achieved, mainly achieved, partially achieved, barely achieved** and **unachieved**.

<sup>8</sup> Figure 10 shows improvements on the quality of models after inspection. Students (auditors) follow the checklist and do report the problems to builders. Most of the teams used this feedback as to improve their final project.

- [2] Becker, H. S. (2008). *Tricks of the trade: How to think about your research while you're doing it*. University of Chicago Press.
- [3] Callele, D., & Makaroff, D. (2006). Teaching requirements engineering to an unsuspecting audience. *ACM SIGCSE Bulletin*, 38(1), 433-437.
- [4] Polajnar, D., & Polajnar, J. (2004, May). Teaching software engineering through real projects. In *WCCCE 2004 Western Canadian Conference on Computer Education* (pp. 83-90).
- [5] Sikkel, K., & Daneva, M. (2011, August). Getting the client into the loop in information system modelling courses. In *Requirements Engineering Education and Training (REET), 2011 6th International Workshop on* (pp. 1-4). IEEE.
- [6] Yvonne Sedelmaier, Dieter Landes, *Active and Inductive Learning in Software Engineering Education*, 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering
- [7] Monsalve, E. S., Leite, J. C. S. D. P., & Werneck, V. M. B. (2015, May). Transparently teaching in the context of game-based learning: the case of simules-W. In *Software Engineering (ICSE), 2015 IEEE/ACM 37th IEEE International Conference on* (Vol. 2, pp. 343-352). IEEE.
- [8] Freire, P. (1996). *Pedagogia da autonomia: saberes necessários à prática docente*. São Paulo: Paz e Terra.
- [9] Gal'perin, P. Y. (1989). Organization of mental activity and the effectiveness of learning. *Soviet Psychology*, 27(3), 65-82.
- [10] Mills, J. E., & Treagust, D. F. (2003). Engineering education—Is problem-based or project-based learning the answer?. *Australasian Journal of Engineering Education*, 3(2), 2-16.
- [11] J.C.S.P. Leite, *Livro Vivo: Engenharia de Requisitos*, available <http://livrodeengenhariaderequisitos.blogspot.com/>, 2007.
- [12] Notes in Requirements Engineering. <https://inf1377.wordpress.com/>. Accessed: 10/22/2015.
- [13] Atlas TI Qualitative Data Analysis. <http://atlasti.com/>. Accessed: 10/22/2015.
- [14] Ries, E. (2011). *The lean startup: How today's entrepreneurs use continuous innovation to create radically successful businesses*. Random House LLC.
- [15] Fagan, M. E. (2001). Advances in software inspections. In *Pioneers and Their Contributions to Software Engineering* (pp. 335-360). Springer Berlin Heidelberg.
- [16] Chen, P. P. S. (1976). The entity-relationship model—toward a unified view of data. *ACM Transactions on Database Systems (TODS)*, 1(1), 9-36.
- [17] Grube, P. P., & Schmid, K. (2008, September). Selecting creativity techniques for innovative requirements engineering. In *Multimedia and Enjoyable Requirements Engineering-Beyond Mere Descriptions and with More Fun and Games, 2008. MERE'08. Third International Workshop on* (pp. 32-36). IEEE.
- [18] Oliveira, A. D. P. A., & do Prado Leite, J. C. S. (2011). Building Intentional Models Using the ERi\* c Method. *Cadernos do IME-Série Informática*, 32, 46-53
- [19] Penzenstadler, B., Mahaux, M., & Heymans, P. (2013, May). University meets industry: Calling in real stakeholders. In *Software Engineering Education and Training (CSEE&T), 2013 IEEE 26th Conference on* (pp. 1-10). IEEE.
- [20] Santos, S. C. D., & Soares, F. S. (2013, May). Authentic assessment in software engineering education based on PBL principles: a case study in the telecom market. In *Proceedings of the 2013 International Conference on Software Engineering* (pp. 1055-1062). IEEE Press.
- [21] Fuks, H., Raposo, A. B., Gerosa, M. A., & Lucena, C. J. (2005). Applying the 3C model to groupware development. *International Journal of Cooperative Information Systems*, 14(02n03), 299-328.
- [22] R.A. Likert. A technique for the measurement of attitudes Archives.
- [23] Simon, H. A. (1959). Theories of decision-making in economics and behavioral science. *The American economic review*, 253-283.
- [24] Questionnaires for Requirements Assessment. <http://www.inf.puc-rio.br/~rportugal/re-assessment>. Accessed: 10/22/2015
- [25] Nuseibeh, B., & Easterbrook, S. (2000, May). Requirements engineering: a roadmap. In *Proceedings of the Conference on the Future of Software Engineering* (pp. 35-46). ACM.
- [26] Berenbach, B. (2005, August). A hole in the curriculum. In *Proceedings of the 1st International Workshop on Requirements Engineering Education and Training (REET'05), Paris, France*.
- [27] Zowghi, D., & Paryani, S. (2003, September). Teaching requirements engineering through role playing: Lessons learnt. In *Requirements Engineering Conference, 2003. Proceedings. 11th IEEE International* (pp. 233-241). IEEE.
- [28] Liang, P., & De Graaf, O. (2010, September). Experiences of using role playing and wiki in requirements engineering course projects. In *Requirements Engineering Education and Training (REET), 2010 5th International Workshop on* (pp. 1-6). IEEE.
- [29] Serrano, M., Serrano, M., Napolitano, F., Kinder, E., Douglas, M., Loyola, D., ... & do Prado Leite, J. C. S. (2008). Uma Proposta para Avaliação de Equipes de Requisitos. In *Workshop em Engenharia de Requisitos (WER) 2008*.
- [30] Waters, R., & McCracken, M. (1997, November). Assessment and evaluation in problem-based learning. In *Frontiers in Education Conference, 1997. 27th Annual Conference. Teaching and Learning in an Era of Change. Proceedings.* (Vol. 2, pp. 689-693). IEEE
- [31] Lawrence Peters, Ana M. Moreno, *Educating Software Engineering Managers – Revisited*, 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering.