*Author version, published in:* **Cognition Technology & Work**, *19 (2), pp. 493-515, Springer, 2017.*

# The use of Task Modeling in Interactive System Specification

Taisa Guidini Gonçalves, Káthia Marçal de Oliveira, Christophe Kolski

*LAMIH CNRS UMR 8201, University of Valenciennes and Hainaut-Cambrésis,*
*Le Mont Houy, 59313 Valenciennes cedex 9, France*
*Taisa.Guidinigoncalves@etu.univ-valenciennes.fr, {Kathia.Oliveira, Christophe.Kolski}@univ-valenciennes.fr*

**Abstract** Task modeling is undoubtedly a key step for task analysis during the development of iterative systems since it helps not only in the understanding of what users want but also in how to design for them. As a consequence, it should be considered as indispensable while specifying the requirements of an interactive system. With this in mind, we analyzed the requirement specifications of a typical interactive system which is a compulsory element of the final grade for a Human-Computer Interaction course in two Master's degree programs at the University of Valenciennes. To that end, we decided to use an approach based on project-based learning (PjBL). Sixty-three requirement specifications performed since 2010 were investigated to find out how the task modeling was represented. Moreover, suggestions concerning pedagogical improvement made by the students in an open question on the evaluation form were analyzed using the Ground Theory method. The results showed that since students are not required to perform task modeling by the requirement specification, they do not do so even in the case of complex interactive systems. Criticism, positive and improvement issues raised by the students were also identified. In this paper we present a detailed analysis of this study that opens up questions about HCI education and effective learning of task modeling, and, as a consequence, its potential use in industry.

**Keywords** Task modeling; User interface design; HCI education; Project-based learning; Case study.

## 1. Introduction

The development of interactive systems has been increasingly supported by a considerable number of models (Santoro, 2005). In this context, task models – the result of task modeling – are essential to represent Human-Computer Interaction (HCI) at a high level of abstraction (Diaper and Stanton, 2004). A task model improves understanding of the application domain, supports effective design, as well as usability evaluation, and represents documentation for the system (Courage et al., 2009); (Diaper and Stanton, 2004).

Task modeling is considered to be a more precise refinement and specification of the task analysis, with the main goal of building a model that describes the relationships among the various identified tasks precisely (Santoro, 2005). Task analysis, which may be used at any stage of the software development cycle, is a good opportunity to involve users in the design process (Paradowski and Fletcher, 2004); engineers observe the actions of users in detail in order to understand how tasks are executed to achieve the desired goals (Kolski et al., 2012) and then represent them in task models.

Aware of the importance of task modeling in interactive system design, this paper presents a study performed with the goal of investigating how task modeling is being represented in the requirement specification of interactive systems. To do so, we analyzed sixty-three requirement specifications for a complex interactive system performed by students from two Master's degree programs at the University of Valenciennes, in the context of a HCI course. The same professor has been teaching this course since 1994. Using a Project-Based Learning approach (Markham, 2012), he has given the same industrial problem as the subject for the requirement specification counted as part of the final grade of the HCI course since 2010. Our goal in this study was to answer a main research question: *To what extent has task modeling been applied in requirement specifications of typical interactive systems?* To answer this question, we analyzed how task modeling was actually applied in the requirement specifications (level of detail, coverage of all user profiles and consistency with the functional specification).

This paper is organized into five more sections. In Section 2 we present a background divided into two parts: (i) a brief description of the main concepts and work related to task modeling; and (ii) an overview of the different educational theory trends to order to situate the context in which this study was carried out. In section 3, all the details of the research protocol for this study are described. In sections 4 and 5, we analyze and discuss the findings of this study. Finally, in section 6 we draw our conclusions by raising some open questions.

## 2. Background

### 2.1. Task Modeling for HCI

Task modeling is "an iterative process of identification and description of tasks" (Pribeanu, 2007). The input for this process is an informal list of tasks provided by task analysis and the result is represented in a task model (Santoro, 2005).

The task model improves understanding of the application domain, supports effective design, supports usability evaluation, and represents documentation for the system (Caffiau et al., 2014); (Courage et al., 2009); (Diaper, 2004); (Lepreux et al., 2003); (Macredie and Wild, 2000). According to (Limbourg and Vanderdonckt, 2004), task models present diversity in relation to the formality level, such as: making designers aware of possible problems related to usability; assessing human performance; supporting the design by providing detailed models that describe hierarchical tasks, objects and knowledge structures.

Several formalisms have been proposed for task modeling (Diaper and Stanton, 2004); (Santoro, 2005); (Caffiau et al., 2014). We can quote for instance: Hierarchical Task Analysis (HTA) (Annett and Duncan, 1967), Method for Task Description (*Méthode Analytique de Description des Tâches* - MAD) (Scapin and Pierret-Golbreich, 1989), Task Knowledge Structure (TKS) (Johnson and Johnson, 1991), DIANE+ (Tarby and Barthet, 1996), (van Westrenen, 2011), ConcurTaskTrees (CTT) (Paternò et al., 1997), GroupWare Task Analysis (GTA) (Stary and van der Veer, 1999), Kernel of Model for Activity Description (K-MAD) (Caffiau et al., 2014), Human-centered Assessment and Modeling to Support Task Engineering for Resilient Systems (HAMSTERS) (Martinie and Palanque, 2015). Santoro (2005) also suggests the use of activity diagrams from Unified Modeling Language (UML) (Rumbaugh et al., 1997) for task modeling since they have quite similar purposes in several techniques for developing task models. Moreover, *Modéle Conceptuel des Traitements* (MCT from systemic method Merise (Rochfeld and Tardieu, 1983)), Structured Analysis and Design Technique (SADT) and Petri Nets (Abed et al., 1991); (Palanque and Bastide, 1997); (Kontogiannis, 2005); (Ezzedine and Kolski, 2005) have been also used for the same purpose.

According to Limbourg and Vanderdonckt (2004), most of these task modeling notations make it possible to model two aspects: the task structure specifying the task hierarchy and decomposition of the task into sub-tasks, and the task sequencing, linking tasks with different sequencing relationships, constraints or operators. Task models may be represented at different levels of abstraction (Limbourg and Vanderdonckt, 2004). When designers intend to provide specific guidance for the design, then the activities are represented at a low level of granularity. If they want to specify the requirements of how activities should be performed, they consider only the high-level tasks.

Moreover, the definition of the different user profiles that will interact with the system is usually required. The user profile describes the characteristics of potential users of the system in detail and identification of such profiles presents the first understanding of these users (Courage et al., 2009); (Hackos and Redish, 1998). Once profiles are identified, they can be grouped based on their similarities, such as: age (child, youth, adult, elderly), education level (basic, undergraduate, graduate), etc. So, user profile is an explicit digital representation of a person. Each user profile may correspond to a specific task model. If the tasks are different for several profiles, the designers should produce different corresponding task models.

Different works in literature explore the use of task modeling in different contexts (see for instance, (Li et al., 2016); (Anzalone et al., 2015); (Moussa et al., 2015); (Gharsellaoui et al., 2012); (Molina et al., 2007); (Paradowski and Fletcher, 2004); (Paternò and Santoro, 2002); We highlight the following work: Molina et al. (2007) used task modeling (ConcurTaskTree - CTT) on reengineering processes to obtain a mobile version of the system (desktop metaphor) that is situated in the context of mobile learning user interfaces; Paradowski and Fletcher (2004) use the task analysis method to perform usability analysis on the graphical user interface of a software application with the objective of quantifying shortcomings and inefficiencies. For these authors, the task analysis method illustrates a practical and efficient way for software designers to improve software usability, as well as user satisfaction. Paternò and Santoro (2002) present a study about a systematic inspection-based analysis to improve the usability and safety aspects of an application, in which the system prototype and the related task model are considered in the evaluation.

By searching on electronic databases (e.g. Scopus[1] - that indexes several journals and conferences - and Web of science[2]) we found a large number of studies on task modeling theory. Nevertheless, we found only three works (Caffiau et al., 2014) (Kolski et al., 2012) (Oliveira et al., 2015) that describe case studies of the application of task modeling in the context of HCI education. We are also the authors of two of these works.

In the first study, Caffiau et al. (2014) analyzed task modeling training in bioinformatics for a specific task model method (K-MAD) and its associated tool (K-MADe). This study aims to evaluate the training itself in terms of the benefits of using a specific tool, how students design the task models using the tool, and their learning progress. Analyzing the results of a questionnaire for the students after the experiment, the authors conclude that presence of task model tools is better for

[1] www.scopus.com
[2] http://login.webofknowledge.com/

task modeling learning. The students were explicitly asked to do a task model for the purpose of the study. One can ask what the result would have been if this recommendation had not been clearly established.

In a previous study, Kolski et al. (2012) analyzed HCI specifications (such as models) in a graduate program. In general, they concluded that the HCI specifications were very rich, allowing the identification of different types of design error, the identification of design creativity by the students and a great diversity of the solutions. This study motivated us to perform an in-depth investigation into task modeling in the requirement specification of interactive systems in HCI education.

A first step in this direction (Oliveira et al., 2015) was a work about task analysis in the context of undergraduate and graduate HCI programs. This work presented three different studies in teaching task modeling by four different professors, who used different formalisms of task modeling (CTT, HTA, K-MAD). The general conclusion was that: (i) most students are not convinced of the utility of task analysis in software design; (ii) they do not think that the use of task models add value to the design, even if its benefits are highlighted; (iii) they know how to build syntactically/correct task models, but they do not exploit their semantics or consider how to use them in a new design. However, this study was a short high-level discussion consolidating different studies and not exploring in detail the task modeling in the requirement specification of interactive systems.

### 2.2. Learning theories

In educational psychology, there are four main trends in educational theory (Talon et al., 2012); (Warin et al., 2016); (Warin et al., 2011). The first one, the behaviorist approach, is a form of pedagogy centered on the teacher and necessitates individual work from the learner. It is appropriate for mass education. The second, cognitivist approach, allows modeling and explaining the mental activities that generate a behavior. It assumes that the learner interprets the environment according to its mental schemes to internalize new knowledge. The constructivist approach considers that learning results from the interaction between learners and their environment and thus is not only a result of the influence of the environment. Finally, the socio-constructivist approach suggests that learning is situated in a specific context and embedded within a particular social and physical environment.

Learning methods have been proposed to support these approaches. We can quote two of them: Problem-Based Learning (PBL) and Project-Based Learning (PjBL). PBL is utilized to address students' lack of motivation and low pass rates (Neville, 2009). An essential component of this approach is the presentation of a real-world problem before the content to be studied is introduced. PjBL has a constructivist orientation and focuses on student interest (Markham, 2012). The author (Markham, 2012) defines PjBL as "an extended learning process that uses inquiry and challenge to stimulate the growth and mastery of skills". Warin et al. (2016) summarize some of the main features of PjBL as: it is often multi-disciplinary, it takes weeks or months long, it includes the creation of product, it often involves real-world, and fully authentic tasks and settings. Moreover, it usually takes the form of student projects that are realistic cases. In realistic and complex industrial cases, the students will implement what they have learned.

PjBL has spread widely among disciplines (Dargham and Chin, 2015); (He et al., 2016); (Köse, 2010); (Letouze et al., 2016); (Mistry et al., 2016); (Martín et al., 2014); (Seman et al., 2016). He et al. (2016) used the PjBL approach to introduce touch sensing technologies to electrical and computer engineering technology students. They present that the students showed great interest in PjBL and the adoption of touch devices into embedded system design projects helps to improve user interactions with the computing systems. Studies like Martín et al. (2014) present the application of PjBL in Computer Engineering and Software Engineering degrees. This work presents a model, focused on helping teachers and students in PjBL being supported by a tool. The tool helps teachers in designing the project and also helps students with the project specification and implementation.

## 3. Description of the Study

In this section we show the main elements of our study. First, we present the pedagogical approach that was used and the studied project. Then, we present our research protocol describing the context and subjects of the study, the research questions and threats of validity.

### 3.1. Pedagogical approach

With the initial goal of evaluating interactive system specification learning in two Master's degree programs, the main professor (one of the authors) has been carrying out a study since 2010. To that end, he decided to use an approach based on project-based learning (PjBL) theory due to the nature of computer science courses, where the students are usually faced with a software development project in which knowledge of several disciplines is applied. Moreover, the professor of the courses has been able to use his twenty years of experience in teaching and research in interactive system design to stimulate the learning by answering students' inquiries during the project development. Another feature of PjBL that is adequate for this study is constructivist orientation. We argued that in a Master's degree program, the students are the main actors in the process of learning. The professor presents some subjects, provides additional references and gives some stimulus to make the student look for more details about what was presented in class, in other words, focuses on student interest.

However, considering the particularities of the Master's degree programs in terms of hours per course, we had to deal with the following constraints:

- It was not possible to define a project to be developed in weeks or months; we had to place the project in three weeks, where the students are in class during 3 sessions lasting 2 hours each. The students could therefore work outside the class hours. To address this constraint, the project was limited to the requirement specification using what was taught about interactive system development, without the need to code the user interface defined. The final product was, therefore, the requirement specification, including mockups and models.
- It was not possible either to place the students in a real world with real users to be interviewed and the real context of the project (in this case: industrial process control domain). However, the chosen project subject was really related to a complex problem already verified and mastered by the professor. Moreover, the professor creates a scenario in which he plays the role of a highly skilled user (with in-depth knowledge of three of the seven user profiles concerned, in fact the three most important user profiles in this study) to answer specific questions.

Since PjBL usually takes the form of a student project, we considered these adaptations for the application of PjBL in our study as acceptable and that it could be under our control.

### 3.2. The studied project

The students receive a detailed description of a real problem involving an industrial process with five interconnected mixing stations (see Figure 1) that need a supervisory HCI. The real problem considers seven different profiles of potential user: supervisors, overseers, rounds men, production engineers, experts, maintenance technicians and fire department employees (for more details, see below).

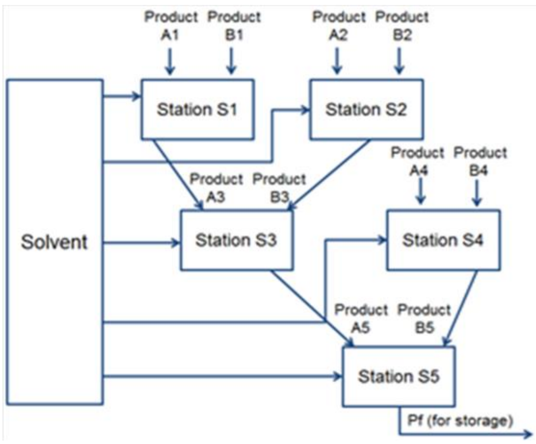


Figure 1. Five interconnected mixing stations

The system must be displayed on a single or several screens in a control room occupied by two human operators, called supervisors, working in rotating shifts (3 x 8h). These **supervisors** remain in contact through support equipment and interactive devices with **rounds men** (novices or more or less experienced) equipped with mopeds.

**Production Engineers** need to perform statistical analyses before, during, or after certain productions, to make the best decisions possible. They are in charge of several processes (for them, this set of stations corresponds to one among other processes).

**Maintenance Technicians** are involved with curative or predictive maintenance objectives (including emergency repairs). Everyone must be informed of their presence in the field and operations concerned.

Several **experts** with over 25 years of experience in many types of processes are constantly reachable (in the country or abroad) to deal with possible questions day and night (if they are on-call) related to an ongoing problem, the progress of which they must be able to observe, irrespective of where they are. They are more or less expert in mechanics, electricity, chemistry, etc.

The **Overseer** is responsible for the supervisors and rounds men and receives comprehensive instructions (objectives, changes,…) from production engineers. He/She needs to have an overall view of all activities and refers to his/her hierarchy. Finally, in the event of a problem, the **fire department employees** must be able to intervene effectively.

Orders cannot originate in the control room, only the supervisor is authorized to decide what actions can be executed on the process. Each action (e.g., start cycle, add solvent) is carried out by the rounds men following the instructions of the supervisors. A rounds man needs about 5 to 10 min to move from one station to another. The mixing process at each of the five stations (see Figure 1), called S1, S2, S3, S4 and S5, lasts 10, 10, 20, 20 and 50 min, respectively. The total process, starting with the basic products (A1, B1, A2, B2, A4 and B4), is composed of the mixtures from these five mixing stations and the result is a finished product (Pf). Several constraints to ensure the security and functionality of the system should be taken into account (e.g. run 24/24, need to exchange specific information while in dangerous situation, etc.). To summarize, different problems of safety, production, and quality have to be taken into account by the different types of user, which shows the different profiles of human-interaction with the system.

### 3.3. The Subjects and Context of the Study

The subjects for our study were the students in the second year of two Master's degree programs at the University of Valenciennes: Automation & Human-Machine Systems (AHMS) and Computer Science (CS), from 2010 to 2014. We recall that following the LMD (Licence, Master, and Doctorate) European System[3], the Licence corresponds to the three first years of study in a university; the Master corresponds to two years after that, and the doctorate 3 years after the Master's degree. Table 1 shows the HCI course in the Master's degree programs to indicate when we performed the study. The number of the semester is considered from the student's admission to the university. That way a student in the seventh semester, for example, is doing the first year of the Master's degree as defined by the European System.

Table 1. Information on Master's degree programs

| | AHMS program | #Credits* | CS (IF and FA) program | #Credits* |
|---|---|---|---|---|
| Third year of licence in engineering | Software Engineering course (basic methods and models) – sixth semester | 2 | Software Engineering course (methods and models) – sixth semester | 2 |
| First year of Master | UML course – seventh semester | 2 | UML course – seventh semester | 2 |
| | | | HCI course including supervised exercises ** – eighth semester | 2 |
| Second year of Master | HCI course including supervised exercises ** & project *** – ninth semester | 2 | Adaptive Interactive System Design and Evaluation course including project *** – ninth semester | 2 |

* European Credits Transfer Scale (ECTS) equivalent
** The exercises about task modeling in both programs are the same.
*** These projects in both programs are the same (it is the one presented in this paper)

In the AHMS program, the study was conducted after 12 hours of classes (9 hours of the HCI course and 3 hours of supervised exercise, including task modeling using CTT, HTA, MAD, and SADT in both cases). In addition, Table 1 shows also that the students have a Software Engineering (methods and models) course in the third year of the Licence in engineering, and a UML course in the first year of the Master's degree.

In the CS program, the study was conducted after 21 hours of classes (in the first year - 9 hours of the HCI course and 3 hours of supervised exercise, including task modeling using CTT, HTA, MAD, and SADT and in the second year - 9 hours of the Adaptive Interactive Systems course). We can see in Table 1 that the students have a Software Engineering course (basic methods and models) in third year of Licence in engineering, and a UML (Unified Modeling Language) course in the first year of the Master's degree. The task modeling exercises and final project are the same in both courses. The HCI course is composed of several lectures (divided into six main parts) and exercises classes covering different subjects, as presented in Figure 2.

The lecture classes are performed with goal of presenting the students with the particularities of interactive system development (specification, design and evaluation). Many illustrations are presented to explain the different concepts and techniques used in interactive system development. In addition, several references to web sites, books and guidelines are provided to support further investigation (resources are also available on an internal system). Concerning task modeling, the most important principles of CTT, MAD, HTA and DIANE are presented and specific references are provided. Moreover, the students from both programs resolve exercises in class using particularly CTT as an example.

[3] The European System aims at unifying European higher education and facilitating equivalence between countries in order to promote student travel within Europe.

The same experienced teacher (more than 20 years as full professor in the university) conducted the classes and the project during the five years using a constructivist approach. He showed the particularities of interactive system specification, design and evaluation and how those subject are integrated in software engineering activities.

***Part 1. Definitions, basic concepts, examples***
*HCI in general. Definition of interactive system with many examples from different domains. Explanation about what are task and activity. Examples of advanced interaction devices…*
***Part 2: Criteria for design and evaluation***
*Heuristics of Shneiderman, Heuristics and maxims of Nielsen, criteria of Scapin and Bastien (1997), International standards (ISO 13407, ISO 16982), definition of User eXperience…*
***Part 3. Analysis and system modeling (for normal and abnormal situations)***
*Recalls concerning Cartesian approach (SADT), fluency graphs, object-oriented analysis (UML), and systemic approach (MERISE), Petri Nets, FMEA (Failure Mode Analysis, Effects and Criticality Analysis), fault trees...*
***Part 4: Software engineering (SE) and HCI: modeling elements***
*SE enriched in terms of HCI (UCD), presentation of several task-modeling formalisms (CTT, MAD, HTA, and DIANE) (with exercises), Mockup and Prototyping are presented as essential in HCI, Interactive system architecture models, presentation modes. Practical exercises with CTT formalism.*
***Part 5: HCI evaluation: Representative methods***
*Global typology, design tests, usability labs, Wizard of oz experiment, heuristic evaluation, evaluation grids, eye-tracking, cognitive walkthrough…*
***Part 6: adaptive UI***
*Basic principles, examples with intelligent, plastic and personalized UI (for the CS program, this last part can be considered as an introduction of the course called "Adaptive Interactive System design and evaluation" (9 hours of lectures focused on different types of so-called adaptive or intelligent user interfaces – The HCI course was considered as a pre-requisite course for this one).*

Figure 2. Content of HCI course

After the classes, the students are required to produce a specification report (20 to 30 pages) about the design of an interactive system using a description of a real problem (see next session) where interactive actions are explicitly defined. The students do not receive any recommendation specification. That means they should provide their specification as they wish, following only a general recommendation of using the subjects presented in the lectures: the teacher explains that they are free to use what they learned in the current course or in other courses (particularly those presented in Table 1 - Software Engineering, UML and Human-Computer Interaction courses).

Figure 3 illustrates the general steps followed in the course to produce the requirement specification. First of all, during three sessions lasting two hours each, in presence of the teacher, the students play the role of design teams, organized in groups of 3-4 members in a competitive context; each group represents a fictive company.

The groups are invited to specify the system for the control room and the HCI intended for the supervision of five inter-connected product mixing stations (presented in the previous section). The students compete to win since the students from the winning group obtain the best marks. Using the PjBL approach, the teacher plays the role of a very skilled worker, since he knows the application domain in-depth: he was first a rounds man in the company; he was then promoted to be a supervisor, and finally promoted to become an overseer. He answers questions asked by each group (without information transmission between the groups for reasons of confidentiality). The students have no or little knowledge about the application domain.

After the first and second sessions the students can (optionally) work outside the sessions. However, after the third session the students must work outside the session to finalize the project. One week after the last session, the students must submit the report (a deadline must be respected, as for a real call for tenders), along with the answers to an evaluation questionnaire about the course (see Appendix A). Finally, in the last step the teacher evaluates the students' reports and communicates the final grade and a ranking of the groups.

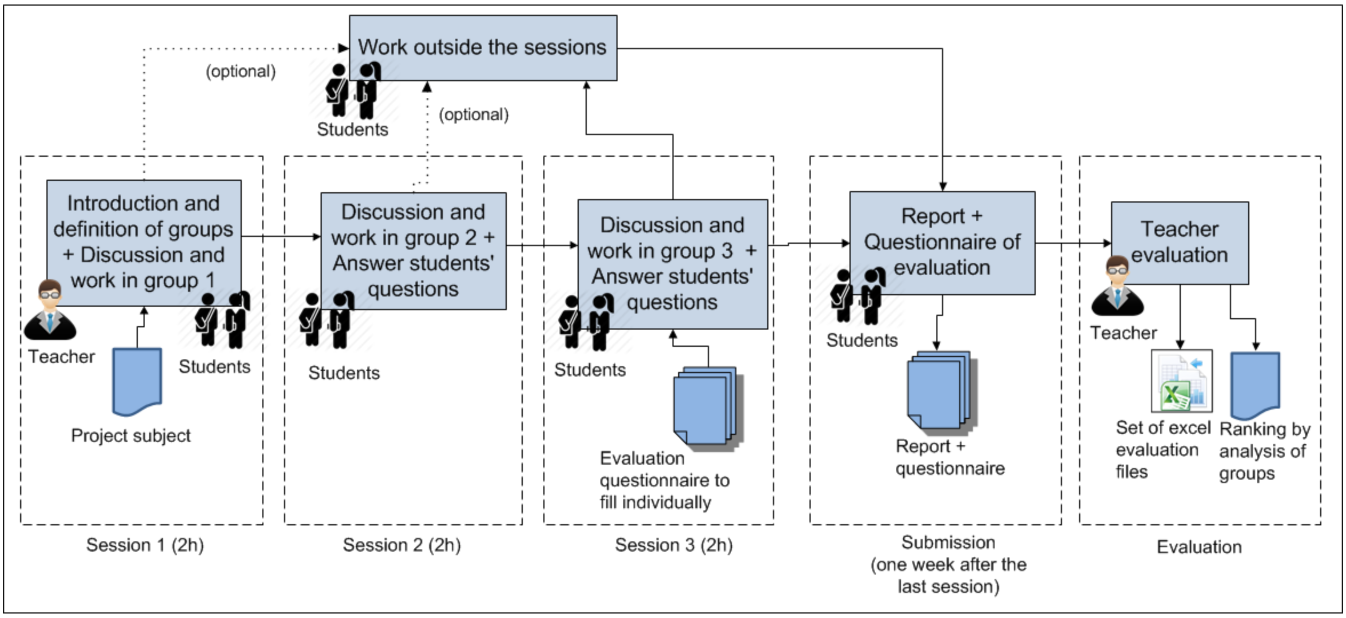


Figure 3. General steps in the development and evaluation of the requirement specification

Before 2012, the Master's degree program in Computer Science was only composed of full-time students (CS-IF). From 2012, the program was divided into two groups: full-time (CS-IF) and block-release apprenticeship (CS-FA), where the students work part-time on the Master's degree and part-time in a company working in industrial projects. We analyzed 63 reports: 20 from the AHMS program and 43 from the CS program (30 from CS-IF and 13 from CS-FA). These reports were produced by 205 students: 55 students from the AHMS program (2010-2014), 106 students from CS-IF (2010-2014) and 44 students from CS-FA (2012-2014) for which we had all the answers of the feedback questionnaire. For the purpose of this study, we only considered complete projects for which we had the report and the answers for the students' evaluation questionnaires.

### 3.4. The Research Questions

To execute this study, we defined one main question and secondary questions. We recall that the modeling task was done in an educational context (not in an industrial context). The main question is:

- **Question 1:** *To what extent has task modeling been applied in requirement specifications of typical interactive systems?*

  Moreover, we defined the following secondary questions:
- **Question 2 -** *Which are the methods used for task modeling?*
- **Question 3 -** *How detailed was the task modeling?*
- **Question 4 -** *Does the task modeling consider all profiles defined in the problem?*
- **Question 5 -** *Are the user profiles described in task modeling also described in use case diagrams?*

Based on the Evaluation questionnaire (Appendix A) answered by students at the end of the project, we defined two more questions:

- **Question 6:** *What is the student's feedback considering the project subject, evaluation and pedagogical issues?*
- **Question 7:** *Does the student propose improvements in the used teaching/learning pedagogy?*

(-1) not considered - "task modeling not considered";
(0) bad - "bad task modeling";
(1) just mentioned but not detailed - "task modeling mentioned but not detailed";
(2) not enough - "task modeling is not enough";
(3) average - "average task modeling";
(4) good - "good task modeling"; and
(5) very good - "task modeling very well defined".

Figure 4. Likert scale

To answer questions 1, 2, 3, 4 and 5, two authors of this paper (first individually and then in a consensual meeting) analyzed the specification reports.

To answer question 1, we used a 7-point Likert scale adapted from the one proposed by (Kolski et al., 2012) in a case study about HCI specification (Figure 4). We defined each scale point for our context (text presented in quotes in Figure 4) when analyzing the reports.

For question 2, we looked for any kind of task specification (informal specification not using a method, or specification using a specific method such as those taught in class - CTT, HTA, MAD, and SADT and Petri Nets). For question 3, we classified the specification as presenting very generic modeling (named global modeling) or detailed modeling. For question 4, we checked if the task modeling considered the seven profiles of the study project (see section 3.2). This question was completed by verifying if each profile was considered in the task model and in the use case model usually used for requirement specification, answering question 5.

To answer questions 6 and 7, we collected the data from the responses to the Evaluation questionnaire (Appendix A). This questionnaire is composed of several closed questions and one open question to be answered by each student. For the closed questions, one of the authors quantified the number of answers for each item being evaluated and generated the graphical results. The answers of the open question were analyzed again by the two authors (first individually and then in a consensual meeting) using the Ground Theory method (Strauss and Corbin, 1998), as presented in section 4.8.

### 3.5. Threats to validity

To analyze the results, we considered the threats to validity proposed by Wohlin et al. (2012): construct validity, internal validity, conclusion validity, and external validity. We analyzed each one of them whilst trying to define some mitigation as described below.

Threats to the **construct validity** illustrate the relation between theory and observation, and the questions of whether the treatment adequately reflects the cause; whether the result adequately reflects the effects. In our case, the main aspect related to this treat is whether the final result (i.e., the requirement specification of the project) used as an evaluation for the course is adequate to evaluate the application of task modeling in requirement specifications. Since the project is used as an evaluation for the course, the students could feel anxious about being evaluated, apply what they know best, and not what they do not feel comfortable using. To minimize this threat, we provide the students with a detailed description of a real problem and the problem is discussed during the three class sections. Moreover, we decide to consider any representation as task modeling; from simple lists of tasks to adequate models using a notation presented in class. This representation would be analyzed in the same way considering the predefined scale (Figure 4). The students are also assured that everything they do will be counted positively for them. In other words, the professor motivates the students to use whatever they want to specify the requirements of the proposed interactive system.

Threats to the **internal validity** draw from influences that can affect the independent variables with respect to causality without the researchers' knowledge. In our study this threat is associated to the students and their interest in learning and using task modeling. As previously presented, in all classes the teacher provides additional references for further study about all techniques, standards and theory presented in the class. However, we cannot guarantee that they will look at this material. To minimize the risk of them not doing so, the teacher defines a competitive environment where the best report will receive a bonus (the best mark). However, we cannot be sure they will do it. We therefore assume this risk.

Threats to the **conclusion validity** are those that affect the ability to draw the correct conclusion about the relation between the treatment and the outcome of our study, in our case the application of task modeling. The results can be considered not to be reliable since we are using only a part (task modeling) of the final product (report specification) of the study. To allow us to draw the best conclusion and mitigate this risk, we consider a seven point Likert scale, where we can really identify to what extent the students perform task modeling. Moreover, we introduce a last open question to get the opinion of the student about the pedagogy used in the course. In this way, we can analyze the results comparing what is identified and the opinion of the students.

Finally, threats to the **external validity** are conditions that limit our ability to generalize the results of our experiment outside the scope of our study. This study was planned in the context of two Master's degree programs and, therefore, cannot be generalized. However, the study could be replicated in many other universities.

## 4. Analysis of Results

In this section we present the results by first overviewing the information on the students' profile and then answering each one of the research questions previously defined. For the analysis of the students' answer to the questionnaire, when the students left a blank answer we consider, for reasons of presentation, the value "Not Answer" as a scale point (this happens for 16 questionnaires – 13 with only one blank answer and 3 with more than one).

### 4.1. General students' profile

Concerning the gender of the respondents, we had 23 females and 182 males. Figure 5 presents the general results about the students' evaluation.

We note that the majority of students consider themselves to be good workers (50%) and methodical (71%). They prefer to work in pairs (51%) rather than individually (19%) and in a team (30%). Moreover, the great majority (46%) prefer to be guided in part of the work (i.e., not all the time or have only general lines of the work), in the beginning (39%) than be guided from the start and throughout the work (14%). To analyze the profile of the group of students that work on the same project, we compute the mode for each item as presented in Table 2 for the 63 reports. In general, the working groups are good workers (35%), methodical (68%), that prefer to be guided only in part of the work (37%), but that would prefer to work in pairs (46%).

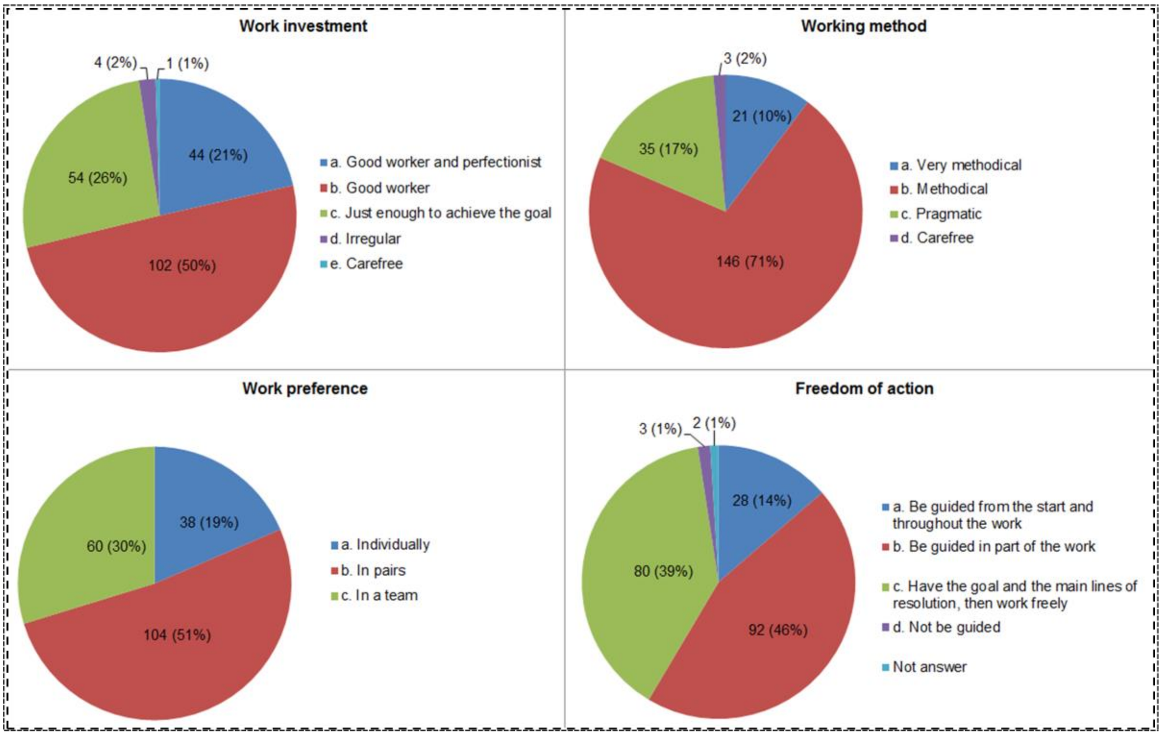


Figure 5. General Profile

### 4.2. Analyzing results of Question 1 - *To what extent has task modeling been applied in requirement specifications of typical interactive systems?*

To answer this question, we considered any specification that describes the final user tasks as a result of task modeling. Figure 6 presents the results of our findings for the 63 reports. We note that 55% (11/20) of the reports from AHMS, 53% (16/30) from CS-IF and 46% (6/13) from CS-FA did not present any result of task modeling.

Considering that the content of task modeling was taught in theoretical and practical classes, we expected to find more meaningful results. We are concerned about the fact that a significant proportion of students did not consider modeling tasks of end users in the specification phase, which is the phase where the task modeling has more emphasis on the system development (Courage et al., 2009); (Santoro, 2005).

Only 48% of the reports (9/20 from AHMS, 14/30 from CS-IF and 7/13 from CS-FA) performed task modeling and only a small part of the reports (12) that considered task modeling was really relevant for us, meaning reports where task models were evaluated from 3 to 5 in the scale.

We found only 3 (approximately 15% of the total) of 20 reports from AHMS program; 6 (approximately 20% of total) of 30 reports from CS program (CS-IF), and 3 (approximately 23% of total) of 13 reports from CS program (CS-FA). These results show that the task modeling activity was not performed well by students in the system specification.

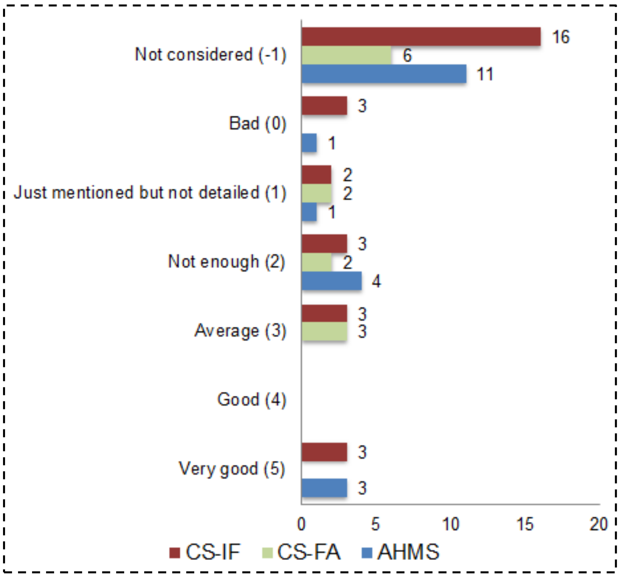


Figure 6. Results of Task modeling

Table 2. Mode of each item of the group profile

| Options in the scale | Work investment | Working method | Work preference | Freedom of action |
|---|---|---|---|---|
| a | Good worker and perfectionist (7) | Very methodical (1) | Individually (6) | Be guided from the start and throughout the work (4) |
| b | **Good worker (22)** | **Methodical (43)** | **In pairs (29)** | **Be guided in part of the work (23)** |
| c | Just enough to achieve the goal (14) | Pragmatic (2) | In a team (11) | Have the goal and the main lines of resolution, then let you do (21) |
| d | Irregular (1) | Carefree (0) | - | Not be guided (0) |
| e | Carefree (0) | - | - | - |
| a & b | Good worker and perfectionist & Good worker (1) | Very methodical & Methodical (3) | Individually & In pairs (2) | Be guided from the start and throughout the work & Be guided in part of the work (0) |
| b & c | Good worker & Just enough to achieve the goal (5) | Methodical & Pragmatic (6) | In pairs & In a team (2) | Be guided in part of the work & Have the goal and the main lines of resolution, then let you do (4) |
| No mode (more than 3 values) | 13 | 8 | 13 | 11 |
| **Total of reports** | **63** | **63** | **63** | **63** |

### 4.3. Analyzing results of Question 2- *Which are the methods used for task modeling?*

The results for this question (see Figure 7) show that for the 63 reports, only 30 present some task modeling where different approaches were used: from informal specifications (not using a method) (14%) to the use of different formalisms for task modeling (33%). We expected to find the use of the taught modeling formalisms: CTT, HTA, MAD, and SADT and Petri Nets. However, after analyzing the first question we identified that the modeling formalisms used were not only those taught in class. For the informal specification, we found: *"simple list of tasks"* (textual description), *"a flow of task execution"*, *"simplified task tree"* (a drawing where they mixed profiles, tasks and sub-tasks), and *"informal design"* (a general drawing of the tasks). For the specifications with the use of formalisms, we found: Activity diagram (from UML), CTT, HTA, MCT (from Merise), SADT, and Petri Nets. Some reports present the use of two combined models, as follows: CTT/Activity diagram, list of tasks/activity diagram, and SADT/Petri Nets. The findings (10/30 = 1 report with HTA, 6 reports with CTT, 3 reports with SADT and Petri Nets) show that the formalisms taught in class were **not systematically** used by the students.

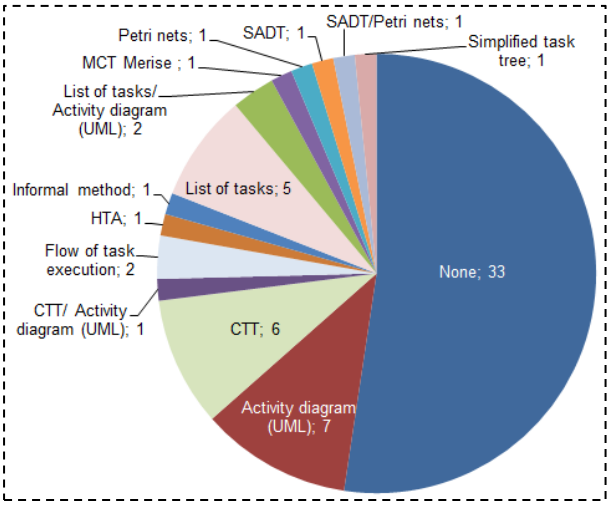


Figure 7. Overall result of task modeling methods

Looking in more detail, Figure 8 shows the methods used for the AHMS program and for the CS program. We were surprised by the fact that there are students from computer science (HCI course during the previous year) who do not use any formalism (22 out of 43 reports) for task modeling. We expected that they would consider the importance of task modeling to specify interactive system and what was taught in class, because they are the future people that will work in the industry. This is especially the case for CS-FA students which already work part-time on industrial projects.

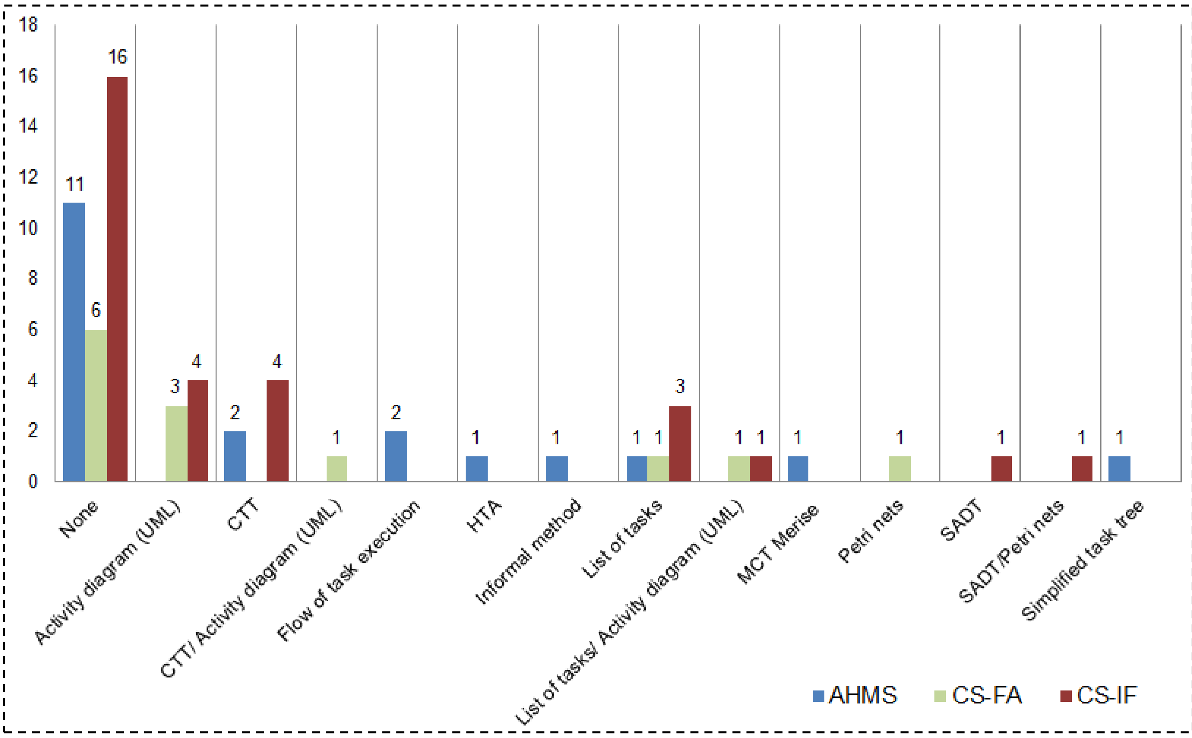


Figure 8. Detailed result of task modeling methods

We were also surprised with the high number (33) of projects that did not present task modeling (more than 50%). By looking at the quality of the requirement specification (considering the grade defined by the professor), we observe that the worst grade, the best and the average for the requirement specification with task modeling was better than those that did not use it (see Table 3). Even the average difference is small (5%), considering that the grade is from 0 to 20, we note that the worst grade for specification with task modeling was much higher than 10 which is the value considered to validate a course. Moreover, one project got the highest grade. Although the task modeling is only one of the elements of the requirement specification, we could say that modeling the task probably helped the students to better understand the problem and, as a consequence, get better specifications.

Table 3. Grade for projects that present and do not present task modeling (grade 0 – 20)

| Specification | # of projects | Worst grade | Best grade | Average |
|---|---|---|---|---|
| Without task modeling | 33 | 6,0 | 16 | 13 |
| With task modeling | 30 | 11,5 | 20 | 14 |

### 4.4. Analyzing results of Question 3- *How detailed was the task modeling?*

As described previously, for this question we classified the result of task modeling as *"global modeling"* or *"detailed modeling"*. For instance, when using CTT we considered *"global modeling"* when the report presents just a high level of task tree (abstract tasks) without defining the primitive tasks. On the other hand, *"detailed modeling"* considers several levels of abstraction in the task tree.

From the 20 reports from the AHMS program, 9 presented a model and 44% of these (4/9) have a "detailed task modeling". For 43 reports from the CS program (30 CS-IF and 13 CS-FA), 21 presented a model and 21% of these (3/14) were detailed (all of them from CS-IF). These results found (see Figure 9) that even when the models are defined, they are not defined in detail, contrary to what we expected.

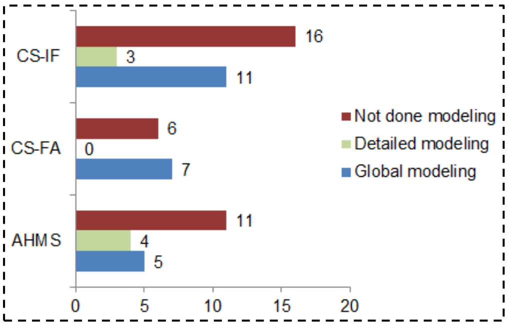


Figure 9. Task modeling details

We argue that task models must provide a level of detail that covers the different levels of user tasks that can contribute to the design of user interfaces (UI) which reflect the reality of the end users. These principles were explained in class[4] and not followed by the students. Figure 10 shows some examples of task models using the CTT method, where we can note: (a) a task model for the supervisor profile and (b) a task model for the rounds men profile, which presents a modeling error because it did not use the temporal operators.

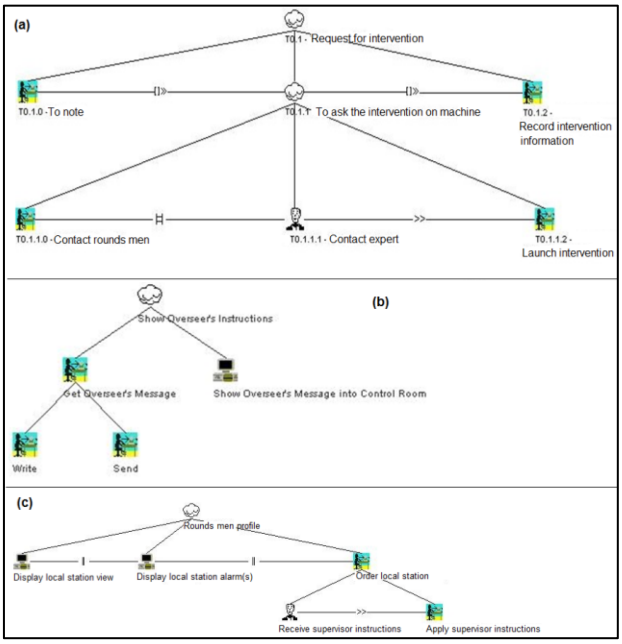


Figure 10. Examples of task models

[4] Recall: Only several weeks ago before the project for the AHMS program, and during the previous year for the CS program.

**4.5. Analyzing results of Question 4-** *Does the task modeling consider all profiles defined in the problem?*

In this question, we found different results for the three groups. Figure 11shows our findings for each user profile defined in the study. We noted that of 63 projects (20 from AHMS, 13 from CS-FA and 30 from CI-IF) only 30 presented task models. For the reports from the AHMS program (9 reports out of 20), the profiles most described are: **rounds men** (9 times) and **overseer** (7 times).

Figure 10 shows two examples (b and c) of task model for the rounds men profile. For the CS-FA program (7 reports out of 13) the profiles most described are: **supervisor** (6 times) and **rounds men** (4 times). Finally, for the CS-IF program (14 reports out of 30), the profiles are: **supervisor** (14 times) and **rounds men** (7 times). Nevertheless, we expected to find all profiles described in each one of the 63 reports, in different task models or in a generic one.

All user profiles were found in the reports by the AHMS and CS-IF program, but only 1/9 and 1/14 reports, respectively from each of these programs, considered all user profiles in task modeling. On the other hand, for CS-FA, six user profiles were found in the reports, but no report (0/7) presents the seven user profiles in task modeling. These results show that the user profiles defined in the project were not identified in most task models. Moreover, usually every user profile should be represented in an associated task model (even if this model is included as part of a complete model for the whole system).

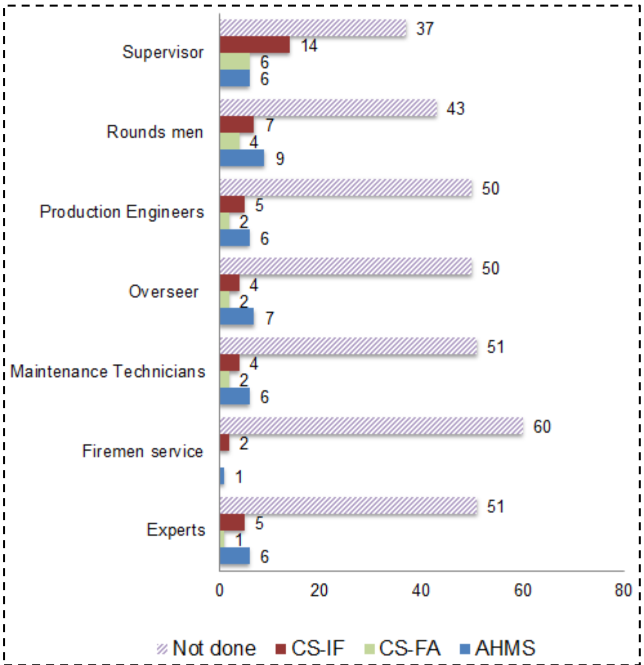


Figure 11. User profiles found in task models

**4.6. Analyzing results of Question 5 -** *Are the user profiles described in task modeling also described in use case diagrams?*

To answer this question, we analyzed the use case diagrams in the requirement specifications. Table 4 summarizes the found results for the three groups. We note that fewer than half of the reports (40%, i.e., 8/20 reports) for the AHMS program presented use case diagrams; of them, only 2 have all user profiles defined. For the CS program, although we found better results concerning the definition of use case diagrams (100% from CS-FA and 86% (26/30) from CS-FI), very few reports present all user profiles (3, i.e., 23% for CS-FA and 10, i.e., 38% for CS-IF). In general, 32% of all the reports considered the seven user profiles in use case diagrams. Figure 12 presents the user profiles found for the three groups. Like in the previous question, we expected that all 63 specifications would present all user profiles defined in the use cases.

Table 4. Results of Use Case Diagrams profiles

| | AHMS | CS-FA | CS-IF | Total |
|---|---|---|---|---|
| # Reports that present Use Case Diagram | 8/20 (40%) | 13/13 (100%) | 26/30 (86%) | 47/63 (74%) |
| # Reports that present all seven profiles | 2/8 (25%) | 3/13 (23%) | 10/26 (38%) | 15/47 (32%) |

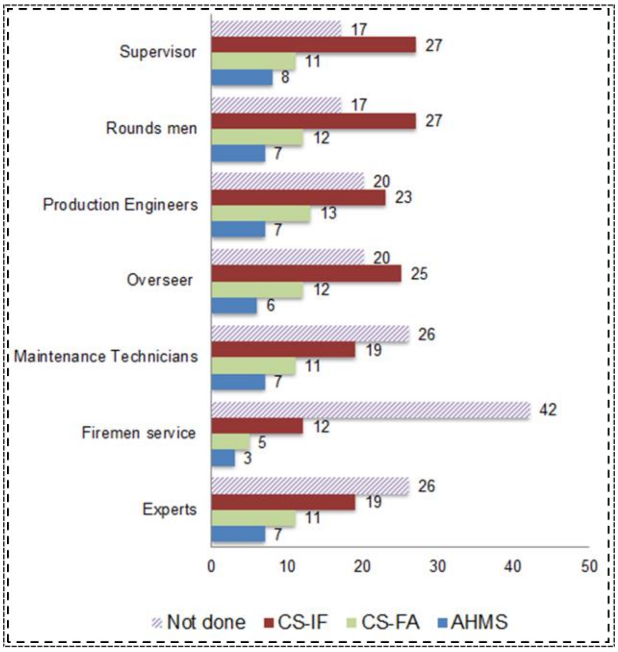


Figure 12. User profiles found in use case diagrams

Figure 13 shows the general results for the user profiles found in use case diagrams and in task models. In both cases we expected to find 63 reports with use case diagrams and task models, but we identified only 47 reports that have use case diagrams and 30 reports that have task models. The user profiles represented the most in use case diagrams are: supervisor, rounds men, production engineers, and overseer. The supervisor was the user profile taken into account the most in task models.

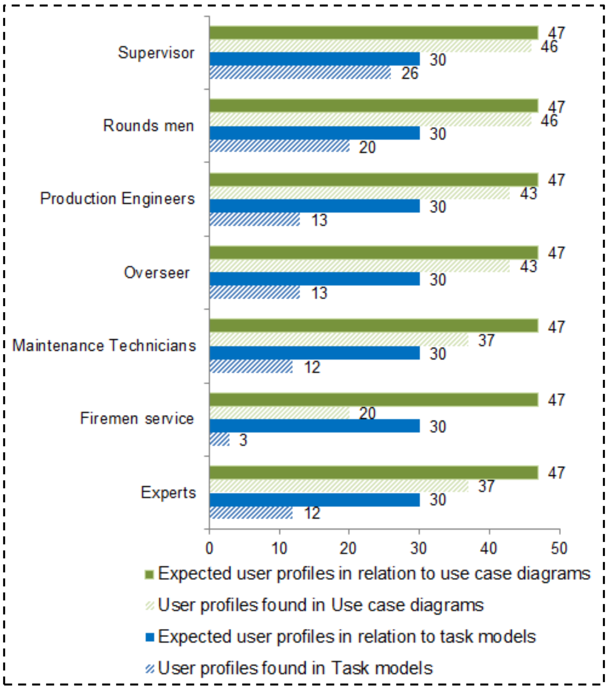


Figure 13. General results for the user profiles found in Use Case diagrams and in Task models (for 63 projects)

**4.7. Analyzing results of Question 6 -** *What is the students' feedback considering the project subject, evaluation and pedagogical issues?*

For this question we analyzed three topics from the 205 responses to the individual evaluation questionnaire. About the mini-project (requirement specification - see Figure 14), most of the students (63,90%) initially felt interested in the subject, considered that the subject was well detailed (54,15%) for their comprehension and that it was "at the right level" (67,32%) for their learning. However, they considered that the time spent in the supervised work classes was not relevant enough (51,22%) or not at all sufficient (16,59%) when compared with the time spent[5] to perform the whole project specification.

The pedagogy was evaluated considering twelve items as presented in Figure 15. We note that:

- more than half of the students (67% - 137) felt interested in the use of a scenario/methodology even though it is obligatory;
- 71% (145) declared they read the subject very carefully;
- 50,73% (104) of the students considered the scenario easy to understand by themselves, but 60,98% (125) consider that is easier to understand in a group;
- 58% (120) of the students are almost certain that the scenario/method makes the supervised classes more motivating and encourages greater participation;
- the students also think the use of scenario/method is relevant (67% - 137 students), easy to apply (53% - 109 students) and that its application (81% - 165 students) has favored the quality of the report;
- 68,79% (141) declare that the acquired knowledge (previously and in the teaching class) was at least largely sufficient to perform the project (52,20% largely and 16,59% absolutely) against 1,46% (3) that considered it not enough.

Finally about the evaluation (see Figure 16), 63% students considered the evaluation by the requirement specification "binding but supportable", 77,56% considered that the evaluation system was "highly" or "absolutely pertinent" to promote learning and the majority (72%) prefer being evaluated with the project instead of only an exam.

In summary, we can conclude with the evaluation of all the questions that the students were quite satisfied with the applied methodology (PjBL) and the assessment of their learning by using the project requirement specification.

**4.8. Analyzing results of Question 7 -** *Do the students propose improvements in the used teaching/learning pedagogy?*

The results for this question were collected from the last open question (*Do you have suggestions for improving the proposed pedagogy?*) of the Evaluation questionnaire (Appendix A). Several students (112) did not answer the open question about the applied pedagogy (i.e., no suggestion form from them). Of 205 students, 93 answered the question.

We analyzed all descriptive answers based on a simplification of the coding process defined by the ground theory method (Strauss and Corbin, 1998). The ground theory is a qualitative method where data collection, analysis and eventual theory stand in close relationship to one another. After collecting the data about a subject, the coding process of ground theory defined by Strauss and Corbin (1998) presupposes the identification of significant data by coding and using a constant comparison analysis method in which, iteratively, we go back and forth in the codes generated from the data, comparing the codes to find adequacy, conformity and coherence among them. The coding process is composed of three main steps: (i) open coding, where the data is broken down and examined to identify concepts, their properties and dimensions; (ii) axial coding, where the performed coding is clustered around the axis of a category; and finally, (iii) selective coding, a process of integrating and refining categories to form a large theoretical scheme so that the research findings take the form of a theory. Our goal was not to form a theory but to analyze the qualitative answers. For this reason, we did not perform step (iii).

We started open coding by marking all excerpts in the data (answers to the open question) related to improvement (the goal of the research question). The excerpts could be one or more sentences in the same answer that have the same meaning. Figure 17 shows an example of a marked excerpt of an answered questionnaire.

As we marked an excerpt, we define a concept or code that identifies it. Whenever we marked another excerpt that seemed to talk about the same concept we grouped it in the same code. To do so, we generalized the code in a way that represents all excerpts. Table 5 presents an example of open coding.

[5] The analysis of the reports shows that the students were motivated since they worked outside of the class to finally produce professional quality reports, sometimes comprising about thirty pages. They considered that the time in class was not enough and all compensated by a considerable work outside of the class.

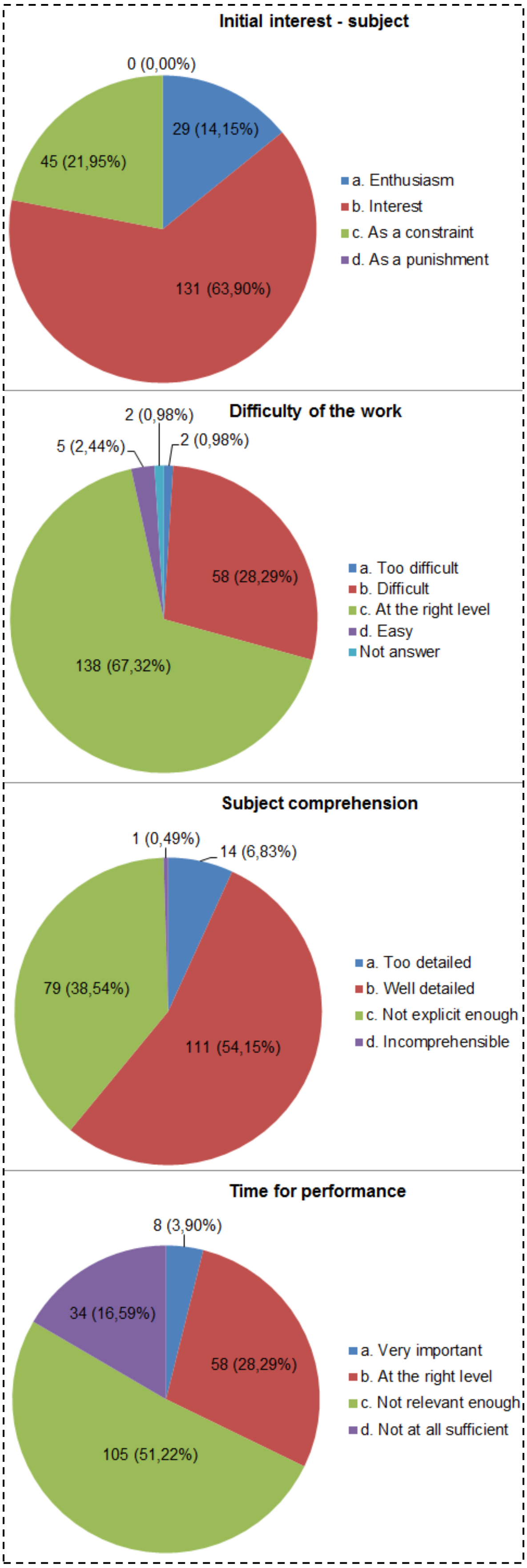


Figure 14. Students' opinion about the studied project

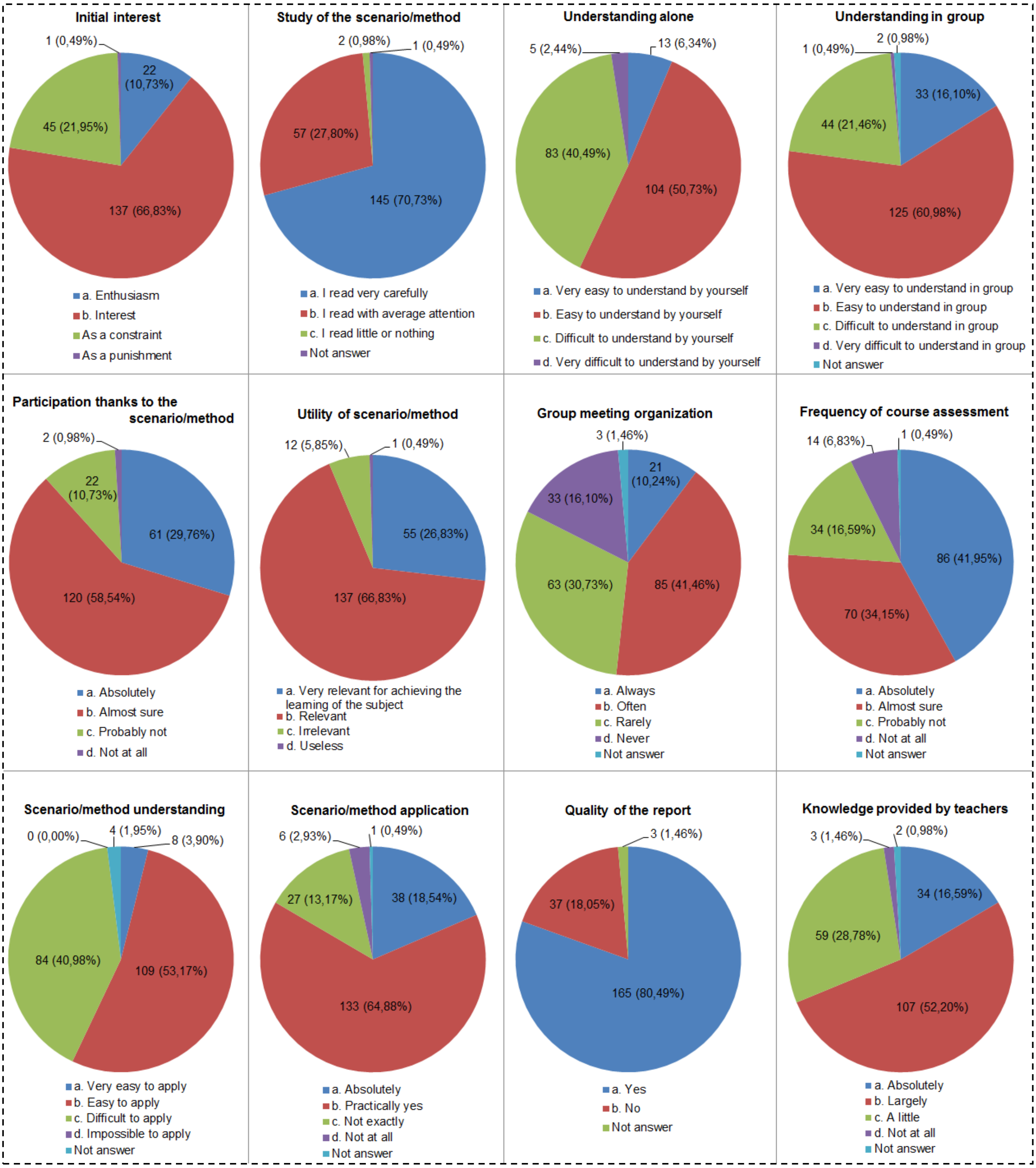


Figure 15. Students' opinion about the pedagogy

| **Workload** | **Relevance** | **Preference of a single exam** |
|---|---|---|
| **a.** Absolutely (21- 10,24%) | **a.** Absolutely (59 - 28,78%) | **a.** Absolutely (9 - 4,39%) |
| **b.** Binding but supportable (128 - 62,44%) | **b.** Highly pertinent (100 - 48,78%) | **b.** Strongly (7 - 3,41%)) |
| **c.** Binding but easy to integrate into your training workloads (37 - 18,05%) | **c.** Not very pertinent (40 - 19,51%) | **c.** A little (40 - 19,51%) |
| **d.** Not at all (17 - 8,29%) | **d.** Not at all (5 - 2,44%) | **d.** Not at all (147 - 71,71%) |
| **Not answer** (2 - 0,98%) | **Not answer** (1- 0,49%) | **Not answer** (2 - 0,98%) |

Figure 16. Students' opinion about the evaluation

Table 5. Example of codes and excerpts

| Code | Excerpts |
|---|---|
| Project in parallel with the course | "I think it would be interesting to put the mini-project at the beginning of the course." "Possibly, to do the supervised work of the project in parallel with the course, in order for students to implement the models explained progressively." "Possibly, to do the (project) in parallel with the course so that the students implement the models explained during the course." |
| Appreciable pedagogy and appreciable methodology | "The pedagogy in this project is satisfactory." "The proposed pedagogy is interesting and effective. These simulation exercises and group projects allow students to directly apply their knowledge and skills by being really active. These methods are, in my opinion, relevant and should be encouraged and increased." "The Teamwork helps to improve the level of knowledge and accelerates learning." "I found the method of evaluation by group projects very relevant and more rewarding in terms of new knowledge, enabling you to express yourself, to share ideas, and assimilate what was taught more than individual exams." |

**Do you have suggestions for improving the proposed pedagogy?**

*I would say that the group work is very interesting, and especially it simulates the professional environment, but it is necessary that the members of the groups express their opinion in front of the others, to organize a debate between groups; in my opinion, it would help those for whom speaking in public is difficult, and also teach them how to be heard and to impose their ideas.*
*I found the method of evaluation by group projects very relevant and more rewarding in terms of new knowledge, enabling you to express yourself, to share ideas, and assimilate what was taught more than individual exams.*

Figure 17. Example of marked excerpts in the questionnaires

After that, all codes were clustered into three main categories (axial coding): (i) criticism issues, (ii) improvement issues and (iii) positive issues. From the 93 answers we identified, 118 excerpts were coded and divided into these groups (see Table 6).

In general, we can quote the following main findings:

- Several students (6/205) think that the course has a large program covering all aspects for interactive system development, making it difficult to retain all the information;
- Some of the students (14/205) argue that it should have more supervised work classes and laboratory classes and fewer lecture classes;
- Several students (10/205) suggest that the project should be performed in parallel with the course;
- Some of them (21/205) mention that they did not really know how to put in place the various models and techniques presented in the course, and they think that it lacked examples of similar projects;
- They suggest some improvements such as the use of a current and commercial problem (6/205); to perform an oral presentation of the project in class (6/205); more guidance for the students with more details of the work that should be done (11/205); add a collective brainstorming phase (2/205);
- Some of the students (19/205) liked the methodology and pedagogy adopted for the project, as well as the type of assessment.

## 5. Discussion

This study involved 205 Master's degree students. These students can all be considered potentially as future designers of interactive systems since they are close to integrating the industry, and therefore working on many projects. These students were placed in a situation close to industrial reality: they were asked to provide a specification document for a complex interactive system; no specific modeling formalism was compulsory, but they could use any modeling method/approach they had learned in the recent years of university education (in HCI classes, software engineering modules, etc.).

The students worked in groups. Sixty-three specification reports were analyzed. It was thus possible to analyze in depth how they exploited the task models that were taught and practiced with several exercises often commented on or mentioned in different courses. Their systematic exploitation was therefore expected in the 63 specifications. It is important to emphasize the following point: if the task models had been imposed *a priori*, they would naturally have been used in each specification, as in any exercise or practical work on modeling tasks, as we have seen in all our HCI courses with different teachers and different levels of students for at least a decade.

We believed that students had understood the relevance or not of task modeling in the specification and whether its use is a constraint or not. The particularity of this study was the complete absence of instructions concerning the modeling of tasks.

Another point to remember is that the HCI module (in computer science, automation or any other domain) is only one module among others, whether students are interested in / passionate about it or not. They must be trained and every year throughout their curriculum they deal with many subjects: programming, database, software engineering, artificial intelligence, operational research, computer architecture, complexity, etc. One trend observed in many universities in the world is the fragmentation of lessons, the reduction of hourly volumes because of budgetary reasons.

Table 6. Categories and codes for the 93 responses on 205 to the question 7

| **Category** | **Code** | **# of excerpts** |
|---|---|---|
| Criticism issues (58 excerpts) | Code 1: Time spent on the project was not enough | 11 |
| | Code 2: Course more convenient (less theory) with concrete examples | 6 |
| | Code 3: More supervised work sciences to devote to mini-project | 14 |
| | Code 4: Lack of orientation to which models to use | 4 |
| | Code 5: More guidance with respect to understanding of the subject and the expectations of mini-project | 11 |
| | Code 6: Examples of previous mini-projects | 6 |
| | Code 7: Proposal of different subjects for the mini-project | 6 |
| Improvement issues (41 excerpts) | Code 1: Other options of work and evaluation outside the mini-project | 7 |
| | Code 2: Oral presentation of the project, integration and participation of groups | 6 |
| | Code 3: Correction of the mini-project | 4 |
| | Code 4: Project in parallel with the course | 10 |
| | Code 5: Proposal of works to make alone and works to make in group | 2 |
| | Code 6: Add a phase of brainstorming and a phase of comparison of the results | 2 |
| | Code 7: Perform throughout the course small and similar practical examples (e.g. Mockup) to the mini-project | 9 |
| | Code 8: Linking this module with a development module | 1 |
| Positive issues (19 excerpts) | Code 1: Appreciable pedagogy and appreciable methodology | 17 |
| | Code 2: Appraisal of the request for opinion from the students | 1 |
| | Code 3: Interesting mini-project subject | 1 |
| | **Total** | 118 |

The results (Figure 6) found for the main question of our study indicate that the level of importance given to task modeling for system specification was not so high. As said above, considering that the content of task modeling was taught in theoretical and practical classes, we expected to find more meaningful results. We are concerned about the fact that a significant proportion of students did not consider the modeling of user tasks in the specification phase, which is the phase where the modeling of tasks has more emphasis on the system development (Courage et al., 2009); (Hackos and Redish, 1998).

As previously shown (Figure 6 and Figure 7) only 48% (30/63) of the reports performed task modeling. A small part (40%, i.e., 12 of 30 reports) of the reports presented a good result of task modeling. However, though analyzing the answers collected for the last question (question 7) together with each one of the other questions (questions 1 to 5 of the section 3.4), we have very rich points of discussion.

First of all, for the first question *(To what extent has task modeling been applied in requirement specifications of typical interactive systems?)* we found that only 48% (30/63) of the reports presented task models. Analyzing the comments of the students, we could suppose that maybe they do not have enough time or they do not feel comfortable in applying the methods since they did not have as many practice exercises as they believed to be necessary (**Criticism issues** - *code 1* - Time spent on the project was not enough and *code 2* - Course more convenient (less theory) with concrete examples). In addition, in Question 6 (**Realization time**), 51% of the students think that the dedicated realization time compared to the work required was *not important enough* and 34 students (17%) think that the time was *not enough*. We believe that the time must have been an important and decisive factor in choosing to perform task models or other models for the project (for instance, use cases). However, this does not explain why, when they have the choice, they prefer not to do task modeling.

One possibility could be the lack of availability of good tools for each formalism. Even the study of (Caffiau et al., 2010b) was focused on the use and evaluation of a specific tool, they concluded the use of tools helps in task modeling learning. We did not exploit this aspect in our study. We note that 74.6% (47/63) of the reports present use case diagrams (from UML). Several tools (commercialized and free tools) are available to support with UML modeling diagrams. This can indicate that the students probably feel more comfortable with the use of UML. However, even though this percentage is not very bad (at least, if we compare it with the task modeling application), we expected to find more reports with use case models considering the high availability of tools and the wide use of UML in industry. This may suggest that students are not very interested in modeling.

For the second question (*Which are the methods used for task modeling?)* we expected to find the use of the taught modeling methods: CTT, HTA, MAD, and SADT and Petri Nets. However, after analyzing the first question we identified that the modeling methods used were not only those taught in class. We found that only 33% (21/63) of the reports really used taught task modeling formalisms. We could suppose that not all the students knew what exactly to apply (**Criticism issues** - *Code 4* - Lack of orientation to which models to use, *Code 5* - More guidance with respect to understanding of the subject and the expectations of mini-project and *Code 6* - Examples of previous mini-projects). However, this does not explain the fact that some project reports did not present any task modeling. In addition, the students had three sessions with the presence of the professor (supervised work classes) where they could ask questions related to the project. For example, some reports used an activity diagram (UML) to model the user tasks. Even if these reports did not use task model formalisms, we appreciated the students who made an effort to model the tasks of the users.

For the third question (*How detailed was the task modeling?*) we found that only 23% (7/30) reports presented "detailed task modeling". Analyzing the comments of the students, we could suppose that maybe they do not retain and put into practice all the information given about the task modeling methods (**Criticism issues** - *code 2* - Course more convenient (less theory) with concrete examples). However the majority of the students (68,78%) answered that they had enough knowledge obtained previously or with the teacher in the course (see Figure 16 - 107 students answered "largely" and 24 "absolutely") to perform the required work.

For the fourth question (*Does the task modeling consider all profiles defined in the problem?*) we found only 6% (2/30) reports modeling the seven profiles defined in the problem description. However we found that the main profiles (rounds men, overseer and supervisor) were considered for several reports. We believe the students chose only the most demanded profiles in relation to the presented scenario.

Finally, for the fifth question (*Are the user profiles described in task modeling also described in use case diagrams?*) we found that only 32% (15/47) gave use case diagrams modeling all user profiles defined. However, we found that the main profiles (rounds men, overseer and supervisor) were considered for several reports. We could suppose that maybe the students thought that not all profiles were important to the system specification.

We also calculated the average (13.78 out of 20) of the final grade of the 63 specification reports. We considered that the specification reports were generally good, but that the task models were not actually developed as expected. This leads us to believe that the Project-Based Learning theory has been generally satisfactory, but that students do not see much interest in using in task models.

Finally, to the best of our knowledge, there is no study in the same direction as ours, investigating the use of modeling by practitioners (in our case students) when it is not compulsory. Nevertheless, a lot of studies related to modeling can be found in literature. For instance to investigate the complementarity of models (e.g. (Siau and Lee, 2004); (Briand et al., 2012)), the comprehensibility of models (e.g. (Nugroho, 2009); (Hadar et al., 2013)), its applicability to support other phases of software development and maintenance (e.g., (Fernández-Sáez et al., 2016); (Nugroho and Chaudron, 2009)) and its application in industry (Anda et al., 2006). Those works differ from ours in their main goal and even if they cannot be compared, open new perspectives of study; for instance: about the comprehensibility of the produced task models, their use to support other activities in software development and their application in industry.

## 6. Conclusion

This paper presents a study about HCI education, with a focus on task modeling. The study was conducted with 205 students from two Master's degree programs at the University of Valenciennes, France. Sixty-three specification reports were analyzed considering different aspects: the definition of task models using methods, the quality of these task models and the specification of profiles with the task models. Furthermore, 205 evaluation questionnaires (Appendix A) were analyzed concerning the project subject, project evaluation, applied pedagogy and suggestions concerning the proposed pedagogy.

In general, this study confirmed what we previously found with some other colleagues (Oliveira et al., 2015): most students do not use task modeling naturally and spontaneously in software design and it seems that they do not think or understand that task models add value to the requirement specification, even if their benefits are highlighted in class. It showed that fewer than 50% of the students' reports presented results of task modeling. Several students complained about the large scope of theory in the course and probably they did not retain the knowledge acquired in previous courses. However, these possible justifications do not explain why, when the students have the choice of performing task modeling or not, they actually do not perform it. But the groups have generally felt very involved and have spent considerable time working outside of the sessions, in order to provide a professional quality report.

If we also consider that the time spent explaining and applying task modeling should not increase too sharply, with the consequent overshadowing of the importance of other essential concepts (dialogue modeling, presentation methods, methodological approaches to design and evaluation, Interactive systems architecture, adaptive/intelligent/customized interaction, context sensitivity, prototyping principles and tools, etc.), it seems urgent: 1) to confirm this study by others of the same type (our study could be used as a benchmark for other researchers); 2) to investigate new methods of teaching task modeling, coming out of the classical, yet natural, and probably the most commonly used in universities, that is the explanation of the basic principles, then application on representative cases and exercises. Moreover, another research study could consider a decision-making support about choosing the models to be applied according to the current stage in the development cycle of the interactive system. Recommendations are available or disseminated on this subject in many HCI works, but relationships should also be advocated with respect to models from other fields of HCI, especially software engineering.

Finally, to summarize, some open questions can be raised from this study: What are the main difficulties of task modeling that impede the students from really performing it? What is the effect of using a tool in the acceptability of one formalism? Why do the students not perform task modeling when they are specifying requirements of an interactive system? How can we effectively and naturally integrate task modeling in the requirement specification of interactive systems? Only by answering these questions for HCI education can we expect the effective learning of task modeling and as a consequence its potential use in industry.

**Acknowledgements** The authors would like to acknowledge the financial support granted by CAPES - Science without Borders Program. They thank warmly the students who participated in this study and Bruno Warin (University of Littoral Côte d'Opale, Calais, France) who proposed the evaluation questionnaire. The authors thank also the anonymous reviewers for their numerous constructive remarks.

## References

Abed, M., Bernard, J.M., Angué, J.C., 1991. Task analysis and modelization by using SADT and Petri Networks, in: Proceedings of Tenth European Annual Conference on Human Decision Making and Manual Control. Liege, Belgium.

Anda, B., Hansen, K., Gullesen, I., Thorsen, H.K., 2006. Experiences from Introducing UML-based Development in a Large Safety-critical Project. Empir. Softw Engg 11, 555–581. https://doi.org/10.1007/s10664-006-9020-6

Annett, J., Duncan, K.D., 1967. Task analysis and training design. Occup. Psychol. 41, 211–221.

Anzalone, D., Manca, M., Paternò, F., Santoro, C., 2015. Responsive Task Modelling, in: Proceedings of the 7th ACM EICS, EICS '15. ACM, New York, NY, USA, pp. 126–131. https://doi.org/10.1145/2774225.2775079

Briand, L., Labiche, Y., Liu, Y., 2012. Combining UML Sequence and State Machine Diagrams for Data-flow Based Integration Testing, in: Proceedings of the 8th European Conference on Modelling Foundations and Applications. Springer-Verlag, Berlin, Heidelberg, pp. 74–89. https://doi.org/10.1007/978-3-642-31491-9_8

Caffiau, S., Scapin, D.L., Sanou, L., Guittet, L., 2014. Using K-MADe for learning task modeling: Interest and Difficulties. J. Interact. Pers.-Système 1, 1–28.

Courage, C., Redish, J.G., Wixon, D., 2009. Task Analysis, in: Sears, A., Jacko, J.A. (Eds.), Human-Computer Interaction: Development Process. CRC Press, NW, pp. 33–53.

Dargham, J.A., Chin, R.K.Y., 2015. A Framework for Integrating Project-Based Learning into the Curriculum for Outcome Based Education, in: Proceedings of the 7th International Conference on Engineering Education (ICEED). Presented at the International Conference on Engineering Education (ICEED), IEEE, pp. 6–9.

Diaper, D., 2004. Understanding Task Analysis for Human-Computer Interaction, in: Diaper, D., Stanton, N.A. (Eds.), The Handbook of Task Analysis for Human-Computer Interaction. Lawrence Erlbaum Associates, Mahwah, NJ, pp. 5–47.

Diaper, D., Stanton, N.A., 2004. The Handbook of Task Analysis for Human-Computer Interaction. Lawrence Erlbaum Associates, Mahwah, NJ.

Ezzedine, H., Kolski, C., 2005. Modelling of cognitive activity during normal and abnormal situations using Object Petri Nets, application to a supervision system. Cogn Technol Work 7, 167–181. https://doi.org/10.1007/s10111-005-0184-4

Fernández-Sáez, A.M., Genero, M., Caivano, D., Chaudron, M.R., 2016. Does the Level of Detail of UML Diagrams Affect the Maintainability of Source Code?: A Family of Experiments. Empir. Softw Engg 21, 212–259. https://doi.org/10.1007/s10664-014-9354-4

Gharsellaoui, A., Bellik, Y., Jacquet, C., 2012. Requirements of Task Modeling in Ambient Intelligent Environment, in: Proceedings of the 2nd International Conference on Ambient Computing, Applications, Services and Technologies. Presented at the International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT'12), pp. 71–78.

Hackos, J.T., Redish, J.C., 1998. User and Task Analysis for Interface Design. John Wiley & Sons, Inc.

Hadar, I., Reinhartz-Berger, I., Kuflik, T., Perini, A., Ricca, F., Susi, A., 2013. Comparing the comprehensibility of requirements models expressed in Use Case and Tropos: Results from a family of experiments. Inf. Softw. Technol. 55, 1823–1843. https://doi.org/10.1016/j.infsof.2013.05.003

He, N., Huang, H., Qian, Y., 2016. Teaching Touch Sensing Technologies through Project-Based Learning, in: Proceedings of IEEE Frontiers in Education Conference. Presented at the IEEE Frontiers in Education Conference (FIE), IEEE.

Johnson, H., Johnson, P., 1991. Task knowledge structures: Psychological basis and integration into system design. Acta Psychol. (Amst.) 78, 3–26.

Kolski, C., Loslever, P., Sagar, M., 2012. The performance of future designers on the specification of supervisory HCI: case study of a simulated work situation. Cogn. Technol. Work 14, 107–128. https://doi.org/10.1007/s10111-010-0169-9

Kontogiannis, T., 2005. Integration of task networks and cognitive user models using coloured Petri nets and its application to job design for safety and productivity. Cogn. Technol. Work 7, 241–261. https://doi.org/10.1007/s10111-005-0010-z

Köse, U., 2010. A web based system for project-based learning activities in "web design and programming" course. Procedia - Soc. Behav. Sci. 2, 1174–1184. http://dx.doi.org/10.1016/j.sbspro.2010.03.168

Lepreux, S., Abed, M., Kolski, C., 2003. A Human-centred Methodology Applied to Decision Support System Design and Evaluation in a Railway Network Context. Cogn Technol Work 5, 248–271. https://doi.org/10.1007/s10111-003-0128-9

Letouze, P., Souza Júnior, J.I.M., Silva, V.M., 2016. Generating Software Engineers by Developing Web Systems: A Project-Based Learning Case Study, in: Proceedings of the 29th International Conference on Software Engineering Education and Training. Presented at the International Conference on Software Engineering Education and Training, IEEE, pp. 194–203.

Li, S., Chen, W., Fu, Y., Wang, C., Tian, Y., Tian, Z., 2016. Modeling Human Behavior in Manual Control Rendezvous and Docking Task. Cogn Technol Work 18, 745–760. https://doi.org/10.1007/s10111-016-0388-9

Limbourg, Q., Vanderdonckt, J., 2004. Comparing Task Model for User Interface Design, in: Diaper, D., Stanton, N.A. (Eds.), The Handbook of Task Analysis for Human-Computer Interaction. Lawrence Erlbaum Associates, Mahwah, NJ, pp. 135–154.

Macredie, R., Wild, P., 2000. An Evaluation of the Potential of Task Analysis in the Evolution of Interactive Work Systems. Cogn Technol Work 2, 7–15. https://doi.org/10.1007/s101110050002

Markham, T., 2012. Project Based Learning Design and Coaching Guide. HeartIQ Press.

Martín, J.C., López, C.L., Martínez, J.E.P., 2014. Supporting the design and development of Project Based Learning courses, in: Proceedings of IEEE Frontiers in Education Conference. Presented at the IEEE Frontiers in Education Conference (FIE), IEEE.

Martinie, C., Palanque, P., 2015. Designing and Assessing Interactive Systems Using Task Models., in: Monteiro, I.T., Silveira, M.S. (Eds.), Tutorials of the 14th Brazilian Symposium on Human Factors in Computing Systems. SBC, pp. 29–58.

Mistry, R.D., Halkude, S.A., Awasekar, D.D., 2016. APIT: Evidences Of Aligning Project Based Learning With Various Instructional Strategies For Enhancing Knowledge In Automobile Engineering, in: Proceedings of International Conference on Learning and Teaching in Computing and Engineering. IEEE, pp. 107–114.

Molina, A.I., Redondo, M.A., Ortega, M., 2007. Applying Task Modeling and Pattern-based techniques in Reengineering Processes for Mobile Learning User Interfaces: A case study. J. Comput. 2, 23–30.

Moussa, F., Ismail, I., Jarraya, M., 2015. Towards a Runtime Evolutionary Model of User-adapted Interaction in a Ubiquitous Environment: The RADEM Formal Model. Cogn Technol Work 17, 391–415. https://doi.org/10.1007/s10111-014-0288-9

Neville, A.J., 2009. Problem-based learning and medical education forty years on. A review of its effects on knowledge and clinical performance. Med. Princ. Pract. 18, 1–9.

Nugroho, A., 2009. Level of detail in UML models and its impact on model comprehension: A controlled experiment. Inf. Softw. Technol. 51, 1670–1685. https://doi.org/10.1016/j.infsof.2009.04.007

Nugroho, A., Chaudron, M.R.V., 2009. Evaluating the Impact of UML Modeling on Software Quality: An Industrial Case Study, in: Schürr, A., Selic, B. (Eds.), Model Driven Engineering Languages and Systems: MODELS 2009. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 181–195. https://doi.org/10.1007/978-3-642-04425-0_14

Oliveira, K.M., Girard, P., Gonçalves, T.G., Lepreux, S., Kolski, C., 2015. Teaching Task Analysis for User Interface Design: Lessons Learned from Three Pilot Studies, in: Proceedings of the 27th Conference on Interaction Homme-Machine, IHM'15. ACM, pp. 31(1–6). https://doi.org/10.1145/2820619.2825011

Palanque, P., Bastide, R., 1997. Synergistic modelling of tasks, users and systems using formal specification techniques. Interact. Comput. 9, 129–153. http://dx.doi.org/10.1016/S0953-5438(97)00013-1

Paradowski, M., Fletcher, A., 2004. Using task analysis to improve usability of fatigue modelling software. Int. J. Hum.-Comput. Stud. 60, 101–115. http://dx.doi.org/10.1016/j.ijhcs.2003.09.004

Paternò, F., Mancini, C., Meniconi, S., 1997. ConcurTaskTrees: A Diagrammatic Notation for Specifying Task Models, in: Howard, S., Hammond, J., Lindgaard, G. (Eds.), International Conference on Human-Computer Interaction. Springer US, Boston, MA, pp. 362–369. https://doi.org/10.1007/978-0-387-35175-9_58

Paternò, F., Santoro, C., 2002. Preventing user errors by systematic analysis of deviations from the system task model. Int. J. Hum.-Comput. Stud. 56, 225–245. http://dx.doi.org/10.1006/ijhc.2001.0523

Pribeanu, C., 2007. An approach to Task Modeling for User Interface Design. Int. J. Comput. Electr. Autom. Control Inf. Eng. 1, 1398–1401.

Rochfeld, A., Tardieu, H., 1983. MERISE: An information system design and development methodology. Inf. Manage. 6, 143–159. http://dx.doi.org/10.1016/0378-7206(83)90032-0

Rumbaugh, J., Jacobson, I., Booch, G., 1997. Unified Modeling Language Reference Manual. Addison-Wesley.

Santoro, C., 2005. A Task Model-Based Approach for the Design and Evaluation of Innovative User Interfaces. Presses Universitaires de Louvain, SIMILAR., Louvain.

Scapin, D.L., Pierret-Golbreich, C., 1989. Towards a Method for Task Description: MAD. Work Disp. Units 89, 371–380.

Seman, L.O., Gomes, G., Hausmann, R., 2016. Statistical Analysis Using PLS of a Project-Based Learning Application in Electrical Engineering. IEEE Lat. Am. Trans. 14, 646–651.

Siau, K., Lee, L., 2004. Are Use Case and Class Diagrams Complementary in Requirements Analysis? An Experimental Study on Use Case and Class Diagrams in UML. Requir Eng 9, 229–237. https://doi.org/10.1007/s00766-004-0203-7

Stary, C., van der Veer, G.C., 1999. Task Analysis Meets Prototyping: Seeking Seamless UI-development, in: CHI ’99 Extended Abstracts on Human Factors in Computing Systems, CHI EA ’99. ACM, New York, NY, USA, pp. 104–105. https://doi.org/10.1145/632716.632783

Strauss, A., Corbin, J., 1998. Basics of qualitative research: Techniques and procedures for developing grounded theory. Sage publications.

Talon, B., Sagar, M., Kolski, C., 2012. Developing Competence in Interactive Systems: The GRASP Tool for the Design or Redesign of Pedagogical ICT Devices. Trans Comput Educ 12, 9:1–9:43. https://doi.org/10.1145/2275597.2275598

Tarby, J.C., Barthet, M.F., 1996. The Diane+ method, in: Proceedings of the Second International Workshop on Computer-Aided Design of User Interfaces. Presented at the Second International Workshop on Computer-Aided Design of User Interfaces (CADUI’96), pp. 95–120.

van Westrenen, F., 2011. Cognitive work analysis and the design of user interfaces. Cogn. Technol. Work 13, 31–42. https://doi.org/10.1007/s10111-010-0153-4

Warin, B., Kolski, C., Sagar, M., 2011. Framework for the evolution of acquiring knowledge modules to integrate the acquisition of high-level cognitive skills and professional competencies: Principles and case studies. Comput. Educ. 57, 1595–1614. http://dx.doi.org/10.1016/j.compedu.2011.02.013

Warin, B., Talbi, O., Kolski, C., Hoogstoël, F., 2016. Multi-Role Project (MRP): a new Project-based Learning Method for STEM. IEEE Trans. Educ. 59, 137–146.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A., 2012. Experimentation in Software Engineering. Springer-Berlin Heidelberg.

## Appendix A. Evaluation questionnaire

**Preamble:** We started with a course that could be described as classic, associated with different supports. Then, in supervised work classes, I proposed an active pedagogy, supported by the performance of a collective mini-project. To improve this pedagogy, I would like to know how you feel about it. With this in mind, I would like you to fill out the following questionnaire. The questionnaire responses will be used only for research purposes and anonymously. Thank you in advance for your help.

***General profile***

**1. Gender:**

**2. Work investment -** About your work investment in the master, you consider yourself as:

a. Good worker and perfectionist
b. Good worker
c. Just enough to achieve the goal (The average in an exam for example)
d. Irregular
e. Carefree

**3. Working method -** You evaluate yourself as:

a. Very methodical
b. Methodical
c. Pragmatic

d. Carefree

**4. Work preference -** When you have the choice, you prefer to work:

a. Individually
b. In pairs
c. In a team

**5. Freedom of action** - When doing the work, you prefer to:

a. Be guided from the start and throughout the work
b. Be guided in part of the work
c. Have the goal and the main lines of resolution, then work freely
d. Not be guided

***The proposed mini-project***

**1. Initial interest -** You can say that the theme of the project initially aroused:

a. Enthusiasm
b. Interest
c. As a constraint
d. As a punishment

**2. Subject comprehension -** About your comprehension, you think the subject was:

a. Too detailed
b. Well detailed
c. Not explicit enough
d. Incomprehensible

**3. Difficulty of the work -** You consider the work to be done:

a. Too difficult
b. Difficult
c. At the right level
d. Easy

**4. Time for performance** - Compared to the work required to complete the mini-project, you consider the time for performance spent in the supervised work classes was:

a. Very important
b. At the right level
c. Not relevant enough
d. Not at all sufficient

***The pedagogy***

**1. Initial interest -** You can say that the obligation to respect a scenario/methodology initially aroused:

a. Enthusiasm
b. Interest
c. As a constraint
d. As a punishment

**2. Study of the scenario/method –** Did you read the scenario/method (in relation to project subject)?

a. I read very carefully
b. I read with average attention
c. I read little or nothing

**3. Understanding alone –** You think the scenario/method (project subject) is:

a. Very easy to understand by yourself
b. Easy to understand by yourself
c. Difficult to understand by yourself
d. Very difficult to understand by yourself

**4. Understanding in group -** You think the scenario/method (project subject) is:

a. Very easy to understand in a group
b. Easy to understand in a group
c. Difficult to understand in a group
d. Very difficult to understand in a group

**5. Participation thanks to the scenario/method -** Compared to sessions where the teacher presents the knowledge to learn on the "blackboard" (video presentation), do you think the scenario/method makes the supervised classes more motivating and encourages greater participation?

a. Absolutely
b. Almost sure
c. Probably not
d. Not at all

**6. Utility of scenario/method** - You think the scenario/method is:

a. Very relevant for achieving the learning of the subject / subjects studied in class

b. Relevant
c. Irrelevant
d. Useless

**7. Group meeting organization -** Were the group meetings organized (designation of a facilitator, a rapporteur, agenda, duration, time of individualized speech, etc.):

a. Always
b. Often
c. Rarely
d. Never

**8. Frequency of course assessment -** Do you think that regular assessments encourage better learning than an overall assessment at the end of the course?

a. Absolutely
b. Almost sure
c. Probably not
d. Not at all

**9. Scenario/method understanding -** You think the scenario/method is:

a. Very easy to apply
b. Easy to apply
c. Difficult to apply
d. Impossible to apply

**10. Scenario/method application -** Did you apply the scenario/method?

a. Absolutely
b. Practically yes
c. Not exactly
d. Not at all

**11. Quality of the report** – Has the application of scenario/method favored the quality of the final product (the report)?

a. Yes
b. No

**12. Knowledge provided by teachers -** Is the knowledge acquired by your group or the course given by the teacher before the project sufficient to do the required work?

a. Absolutely
b. Largely
c. A little
d. Not at all

***The evaluation***

**1. Workload** - Does the system of evaluation by report seem cumbersome?

a. Absolutely
b. Binding but supportable
c. Binding but easy to integrate into your training workload
d. Not at all

**2. Relevance** – Does the evaluation system seems relevant to promote learning?

a. Absolutely
b. Highly pertinent
c. Not very pertinent
d. Not at all

**3. Preference of a single exam** - Would you have preferred a global exam instead of an exam and the project report?

a. Absolutely
b. Strongly
c. A little
d. Not at all

***Open question***

Do you have suggestions for improving the proposed pedagogy?