Applying Interdisciplinarity and Agile Methods in the Development of a Smart Grids System


Computer and Electronic Engineering Graduate Program
Brazilian Aeronautics Institute of Technology
Sao Jose dos Campos, Sao Paulo, Brazil
marcelopaivaramos@gmail.com, gmatuck@gmail.com, ciromatrigrani@gmail.com, samoel@gmail.com, eliezer.segeti@gmail.com, mca.leite@gmail.com, cunha@ita.br, vdias@ita.br.

Abstract — This paper describes an experiment of integrating undergraduate and graduate courses at the Brazilian Aeronautics Institute of Technology, within the Computer Engineering Undergraduate Program and the Electronic and Computing Engineering Graduate Program, during the 2nd semester of 2011. A Smart Grids System case study was used as a Problem Based Learning strategy together with agile best practices within a project development. This experiment has involved four courses: Embedded Systems (undergraduate); Embedded and Real-Time Systems; Software Quality, Reliability, and Safety; and Advanced Topics in Software Testing. During this experiment, the students of all these courses had to interact and collaborate in order to produce a prototype of a Smart Grids System, simulating a real project development.

KeyWords – Interdisciplinarity; Agile Methods; Problem Based Learning; Smart Grids Systems; Software Engineering Integration.

I. INTRODUCTION

The development of software projects is a complex process, especially when developed with distinct functions and by different teams [1].

During the 2nd semester of 2011, at the Brazilian Aeronautics Institute of Technology (Instituto Tecnologico de Aeronautica - ITA), students from the four different courses of: CES-63 Embedded Systems; CE-235 Embedded and Real-Time Systems; CE-230 Software Quality, Reliability, and Safety; and CE-237 Advanced Topics in Software Testing have participated in the development of an academic experiment, involving Problem Based Learning (PBL).

This article describes the application of interdisciplinarity in a PBL, involving the agile development on a Smart Grids System.

The development process was based on the Scrum Agile Method and their best practices. The communication and collaboration among teams have represented one of its main challenges [2].

Besides the agile development, only the use of collaborative tools and an Integrated Computer Aided Software Engineering Environment (I-CASE-E) were used in this experiment.

This article represents a natural continuation of the previous work done by [2], during the 1st semester of 2011.

Next sections describe: the Smart Grids contextualization; the integration among four courses; the adopted Agile Method; an analysis and discussion of some obtained main results; and also some conclusions, recommendations, and suggestions for future works.

II. SMART GRIDS CONTEXTUALIZATION

Nowadays, economic growth and Information and Communication Technologies (ICT) have been demanding investments and researches for technology solutions that meet the growing needs of the modern world.

To provide sustainability to global growth, the energetic sector emerges as a base of infrastructure, to avoid problems of energy outages and blackouts, causing significant impact on the economy of involved countries [3] [4] [5].

The energetic sector in every country is facing difficulties to appropriately ensure the production and distribution of energy.

Smart Grids Systems represent a set of technologies to provide an efficient and integrated systems management from electric power generation, communications, and infrastructure of authorized network [3] [4] [5].

Through the use of sensors and real-time embedded systems, Smart Grids should mainly provide appropriate management, communication and traceability distribution, and electricity consumption.

Smart Grids technologies have been disseminated and used on a large scale, by developed countries, and its benefits have been already proven [3] [4] [5].

Fig. 1 shows an overview of a Smart Grids System, with its main components.

The technologies that guide Smart Grids have several different advantages like: providing lower energy consumption; and allowing concessionaries to improve their distributions, while providing better service for clients.
On this scenario, the application of Smart Grids technologies has been reducing energy distribution costs and helping to reduce carbon emissions [6]. With its application, it becomes possible, to offer services with higher quality, reliability, safety, and even security, regarding the distribution of power, performance, monitoring, and the identification of possible system failures [3] [4] [5].

There are the following core roles in Scrum: Product Owner, Scrum Master, and Scrum Team [8] [9] [13].

The Product Owner represents the customer, being primarily responsible for defining product features, prioritizing them according to the market value, performing the accepted results; and adding value and quality to the process [8] [9] [13].

The Scrum Master is the facilitator. He has the main task of ensuring the implementation of the Scrum process, according to recommendations; removing obstacles and distractions; and ensuring the functionality and productivity of the team [8] [9] [13].

Finally, the Scrum Team is responsible for delivering the product, usually consisting of five to nine multifunctional members [8] [9] [13].

The Scrum method is divided into small time boxes, called Sprints, typically with 2 to 4 weeks duration.

At the end of each Sprint, some meetings are held like: the Sprint Review Meeting, focusing on “what” was developed or was pending; and the Sprint Retrospective, doing a reflection on the last Sprint process [8] [9] [11] [13]. Fig. 3 shows the Scrum Process Cycle.

Fig. 4 illustrates the Burndown Chart, an artifact of Scrum responsible for displaying the development status. It shows that it is possible to identify the velocity of each team on a project, and whether the product will be delivered on time [11].

There are the following core roles in Scrum: Product Owner, Scrum Master, and Scrum Team [8] [9] [13].

The Product Owner represents the customer, being primarily responsible for defining product features, prioritizing them according to the market value, performing the accepted results; and adding value and quality to the process [8] [9] [13].

The Scrum Master is the facilitator. He has the main task of ensuring the implementation of the Scrum process, according to recommendations; removing obstacles and distractions; and ensuring the functionality and productivity of the team [8] [9] [13].

Finally, the Scrum Team is responsible for delivering the product, usually consisting of five to nine multifunctional members [8] [9] [13].

The Scrum method is divided into small time boxes, called Sprints, typically with 2 to 4 weeks duration.

At the end of each Sprint, some meetings are held like: the Sprint Review Meeting, focusing on “what” was developed or was pending; and the Sprint Retrospective, doing a reflection on the last Sprint process [8] [9] [11] [13]. Fig. 3 shows the Scrum Process Cycle.

Fig. 4 illustrates the Burndown Chart, an artifact of Scrum responsible for displaying the development status. It shows that it is possible to identify the velocity of each team on a project, and whether the product will be delivered on time [11].

IV. INTERDISCIPLINARY INTEGRATION

This section describes the four courses involved in the development of the Smart Grids on a PBL: CES-63, CE-235, CE-230, and CE-237.
The CES-63 and CE-235, Embedded Systems and Embedded Real-Time Systems, have the same syllabus. So, in this article, all definitions related to CES-63 are still valid to CE-235, and will be referred as Embedded and Real-Time Systems.

The interdisciplinarity in the PBL was promoted through Exercises Lists (ELs). During the 2nd semester of 2011, there were a total of five ELs for each course, as shown on Table I.

On that semester, the use of collaborative tools was essential to carry out the project activities, especially when dealing with multidisciplinary and geographically distributed teams. It also contributed to the compliance schedule of 17 weeks.

A. CES-63 and CE-235 - Embedded and Real-Time Systems

The CES-63 and CE-235 courses aim to introduce the ITA students to the theoretical and practical concepts of embedded and real-time systems development.

Their syllabi include the application of the main methods, techniques, tools, and procedures through troubleshooting tasks supported by the Agile Methods, Model Driven Architecture (MDA), and I-CASE-E tools.

<table>
<thead>
<tr>
<th>ELs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Understanding the minimum documentation necessary and sufficient for the design, prioritization, and planning of initial requirements (Story Cards) for an Embedded and Real-Time System being developed within the standards of Agile Methods.</td>
</tr>
<tr>
<td>2</td>
<td>Practical training for the development of embedded systems, using MDA and I-CASE-E Tools, and updating the minimum documentation, in accordance with the recommendations of the Agile Development.</td>
</tr>
<tr>
<td>3</td>
<td>Practical applications of the knowledge acquired in EL2 in the 1st interaction among the CES-63 and CE-235 courses and the CE-230 and CE-237 courses through the implementation of the requirements (1st Sprint Backlog) designed from EL1.</td>
</tr>
<tr>
<td>4</td>
<td>Adaptation of the documentation and requirements prioritization for the 2nd Sprint Backlog to new requirements for the integration of the six modules of the PBL, imposed by the Clients (Professors), verifying the adaptability of the open scope of the Agile Method Development; and implementation of the 2nd Sprint adapting and adding tests and measurements of quality on the 1st interaction among the CES-63, CE-235, CE-230, and CE-237 courses.</td>
</tr>
<tr>
<td>5</td>
<td>Integration of smart electric meters modules preserving the minimum requirements initially conceived and valued by the results of tests and measurements of quality on the 2nd interaction among the CES-63, CE-235, CE-236, and CE-237 courses.</td>
</tr>
</tbody>
</table>

TABLE I. THE CES-63 AND CE-235 EXERCISES LISTS (ELs)

At the end of the CES-63 and CE-235 courses, an embedded and real-time system including a smart meter for a Smart Grids network was successfully produced, by applying the main theoretical concepts and practical development of embedded and real-time system, supported by the new paradigms of Agile Method, MDA, and I-CASE-E Tools.

The use of Interdisciplinarity has provided students with the development of real requirements for an embedded and real-time system, that was tested and qualified through systems integration and teamwork, combined with an Agile Method Development.

B. CE-230 - Software Quality, Reliability, and Safety

The CE-230 course aims to introduce the ITA graduate students to concepts of Software Quality, Reliability, and Safety, considering the main norms, standards, and procedures.

Its course grid includes topics about new processes for building computer systems with quality on I-CASE-E platforms and the best practices of agile software development. The Exercises Lists (ELs) of the CE-230 course are shown on Table II.

<table>
<thead>
<tr>
<th>ELs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 and 2</td>
<td>Study, preparation, publication, and individual students presentations of: norms of International Organization for Standardization / International Electro Technical Commission (ISO/IEC), and chapters of the Book “Integrating CMMI and Agile Development” [13]. This EL has intended to train students in using the main quality standards, understanding a case study, and integrating a hybrid methodology.</td>
</tr>
<tr>
<td>3</td>
<td>Application of the following techniques: Burndown Chart, Planning Poker, and Harness Test. Building a Quality Assurance Plan in the 1st interaction with the CE-235 course. Measuring quality in the 1st Sprint Development of the CE-235 course through a Reliability and Safety questionnaire.</td>
</tr>
<tr>
<td>4</td>
<td>Updating the developed documentation from the 3rd EL and a new interdisciplinary interaction involving measurements of software quality, reliability, and safety from the CE-235 course on the 2nd Sprint Development.</td>
</tr>
<tr>
<td>5</td>
<td>Atmeasurement of quality on the 3rd Sprint Development (final integration) with the CE-235 course and presentation of results, where students from CE-230 and CE-235 courses came together to perform, in a symbotic way, the delivery of the completed project to the client.</td>
</tr>
</tbody>
</table>

TABLE II. THE CE-230 EXERCISES LISTS (ELs)

At the end of the CE-230 course, the software quality, reliability, and security was measured over an embedded and real-time system, on an integrated network of smart meter for the Smart Grids system, developed by the students from the CE-63 and CE-235 courses. At that time, it was also evaluated their applications of the Agile Method Development and the best practices and techniques used.

The application of Interdisciplinarity provided students to use their learning in the measurement of software quality, reliability, and security in a real system through teamwork combined with best practices of an Agile Method.

C. CE-237 - Advanced Topics in Software Testing

The CE-237 course aims to introduce the ITA graduate students with the understanding of the main techniques involved in the process of advanced topics in software testing. These applications have primarily focused on systems development with an Agile Method, in order to improve the efficiency of their interest in solving engineering problems, by
using the Software Test Technologies to reduce costs and also the involved resources too. The Exercises Lists (ELs) of the CE-237 course are shown on Table III.

<table>
<thead>
<tr>
<th>ELs</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fill the work papers to answer the questions (yes or no) for a hypothetical Company and some Testers who would interact with the students/developers from the CES-63 and CE-235 courses; draw graphs footprints of type Kiviat; analyze the footprints; suggest measures to correct the flaws; and propose training for their Testers [15].</td>
</tr>
<tr>
<td>2</td>
<td>Fill work papers 2-1 and 2-2 from Perry’s book, 2006 [15]; prioritize Test Factors (Risks) in numerical order (1 to 15) or (high, medium, or low); and justify each choice.</td>
</tr>
<tr>
<td>3</td>
<td>Perform the 1st interdisciplinary interaction by analyzing the strengths and weaknesses of the software testing process that has been done within the development process of the integrated real-time embedded system, involving the smart meter network for the Smart Grids system within the context of the CE-235 course.</td>
</tr>
<tr>
<td>4</td>
<td>Fill Test Plans within the Agile Method applying to run it in an interdisciplinary way for each requirement (User Story) developed within the context of the CE-235 course.</td>
</tr>
<tr>
<td>5</td>
<td>After listing all User Stories requirements of a developed module, from the CES-63 and CE-235 courses, find out on which one the functional and non-functional tests are applied.</td>
</tr>
</tbody>
</table>

At the end of the CE-237 course, an embedded and real-time system, involving smart meters for a Smart Grids network was partially implemented and tested, by using the main techniques of advanced topics in software testing and an Agile Method [16].

The interdisciplinarity has provided the CE-237 students with actual requirements for a real problem where the main concepts learned from the advanced topics on software testing could be applied.

The high realism of the proposed environment brought some difficulties to justify the reduction of test scopes like: the high volume of the required theoretical knowledge; the delay in publication of the model; and the pioneering initiative and effort to develop an academic and critical embedded real-time system in an interdisciplinary way, using Agile Methods.

What has minimized these difficulties was the fact that many students of the four involved courses have attended, on the previous semester, the CE-229, a Software Testing Fundamentals Course. Therefore, the developers were also able to apply, in an interdisciplinary way, the main techniques learned from CE-229 course. Google has been using lately this multifunctional concept [17].

D. Interactions among Courses from Different Exercises Lists (ELs)

Fig. 5 illustrates the interdisciplinary interactions occurred among Exercises Lists 1 to 5 (ELs 1 to 5), developed by teams from the different courses of CE-235/CES63, CE-230, and CE-237, during the Smart Grids system project.

On ELs 1 and 2 teams have worked in planning, research, and study. ELs 3, 4, and 5 held three Sprints, which have resulted in the Prototype of the Smart Grids systems embedded software. It was implemented in a Scrum framework with quality, reliability, and safety together with the use of techniques of Software Testing, thereby reducing risks of development.

During the development stages, the teams have met several times in classroom and virtually, via the Internet, which enabled interactions among courses, resulting in collaborative works from students residing from various cities of the country.

V. CASE STUDY

As an interdisciplinary topic of the Problem Based Learning (PBL), Professors (as Client players) drafted a preliminary set of requirements, with minimum specification for the development of a prototype system involving smart electric meters components (Smart Meters).

A conceptual model was initially presented to the students, to be developed and verified for quality and testability and benchmarked, by using Agile Methods on the timeframe of a course of only 17 academic weeks.

The initial conceptual model was divided into six modules to be developed in parallel by six teams and then integrated to form a single system, involving smart electric meters.

Each Smart Meter module distinguished mainly according to their applications:

- Smart Meter Central (SM-CEN) - responsible to control, monitor, manage, and centralize information from all other Smart Meters;
- Smart Meter Network (SM-RED) - responsible for managing communication between remote Smart Meters;
- Smart Meter Residential (SM-RES) - responsible for controlling, monitoring, and managing information from Smart Meters residential and their interactions with Smart Meters Network and Central;
- Smart Meter Industrial (SM-IND) - responsible for controlling, monitoring, and managing information from Smart Meters for industrial departments and their interactions with the Smart Meters Network and Central;
- Smart Meter Outlet or Point of Sale (SM-PVE) - responsible to manage or monitor information from Smart Meters Outlets and their interactions with the...
Smart Meters Network and the Smart Meter Central, involving transactions from purchases and/or sales of energy, coming from Smart Meters Mobiles; and

- Smart Meter Mobile (SM-MOV) - responsible to control, monitor, and manage information from Smart Meters Mobile, their interactions with the Smart Meters Outlet, and eventually with Smart Meters Network and Central.

After the conceptualization of modules, for the beginning of agile development, it was prepared a minimum of requirements to be developed by students from each module (using 5 plus or minus 2 User Stories). Fig. 6 shows the initial problem conceptualization. The gray color sticky notes represent the minimum amount of User Stories prepared for each module.

![Figure 6. The Problem Based Learning (PBL) Initial Conceptualization.](image)

Thus, the Smart Meter modules and their User Stories were divided among six student teams distributed in four different courses (CE-235/CES-63; CE-230; and CE-237 courses). A schema of this division is shown in Fig. 7.

![Figure 7. Teams from CE-235/CES63, CE-230, and CE-237 courses.](image)

The leveling period has involved all members of Smart Grids PBL system and made possible to start a strategic planning and an initial rapprochement among teams, since they were set randomly at the beginning of the courses. Fig. 8 shows the period of leveling knowledge among courses, during the project life cycle.

Throughout the course it was made a projection that this prototype should not spend more than three development cycles (sprints) and each one having at least one User Story to be developed, tested, and audited by each student.

Within each Sprint development, the PBL has increased its level of integration, ending in a single, tested, and audited integrated model, involving the smart electric meters of a Smart Grids System.

![Figure 8. The knowledge-leveling period among courses.](image)

Within each integration phase of the project, some adaptations have been emerged. These would be problematic for a Waterfall Methodology, but normal for an Agile Method (open scope). Some positive effects of these allowed changes and transformations, were:

- After these interdisciplinary interactions, it was possible to obtain a richer and fuller architecture from a parallel development, involving 68 students divided into 6 teams; and
- At the end of the project development, it was noticed that due to scope changes during this agile and open process, new valuable items took shape and were incorporated to the final product.

Fig. 9 shows the final architecture adopted for the Smart Grids System prototype.

![Figure 9. Final architecture adopted for the Smart Grids System prototype.](image)
VI. MODEL EVOLUTION

The student teams involved with CE-235/CES-63, CE-230, and CE-237 courses have designed and implemented in parallel six system prototype modules starting from low level of detailed requirements specification.

The main obtained result was a different prototype from the expected one, with still more complexity and refinements, able to implement some nice features that were not even mentioned or taught at the beginning of the Smart Grids System prototype.

As an example, the Smart Grids system prototype ended up with some additional features to solve integration problems raised from parallel development such as simulators stubs.

From this experience, it was possible to model and build a richer system prototype involving electric smart meters within the knowledge domain of real-time embedded systems.

The final model has consolidated the experiences and ideas developed by six student teams from four different courses at ITA, during the 2nd semester of 2011.

In order to consolidate these ideas into a final product based upon a PBL approach, the following 2 steps had to be taken:

Step One – Selection: On this step, a selection considered only the User Stories from the six student groups that really have added business values and directly related to the initial proposal of an electric smart meter for the Smart Grids System prototype. Some extra features or functionalities arisen from technical adjustments on the integration stages like stubs simulators and other features were not taken into account.

Fig. 10 shows an example of the selection step considering the User Stories with better business value. User Stories in gray color represent lower business value and those in dark color represent the ones that were excluded from selection.

Step Two – Grouping: After selection, a grouping step was performed to identify the usefulness of the work done by the student teams. At this point, parallel teamwork developments have forced the grouping of those common features already implemented more than once. Fig. 11 shows sticky notes of the same color, as redundant features that fulfilled the same needs, which provided some adjustments on the system architecture.

First Step: Selection.

Second Step: Grouping.

Figure 10. The Selection Step involving Prioritization and Classification.

Figure 11. Trials for redundancy elimination.

VII. THE MAIN RESULTS

To build the Smart Grids System prototype, the students from CES-63 and CE-235 courses have implemented 6 Smart Meters, complying with 127 system requirements specified in the initial phase of the project. Some of them have been modified and/or even have arisen, during the agile development process, characterizing new needs.

From a total of 28,770 lines of code, only 6.8% were hand programmed. These results were achieved mainly due to the adoption of Model Driven Architecture (MDA), I-CASE-E tools, and the Scrum Agile Method for managing software development.

From each stage of software development (Sprint), student teams from CES-63 and CE-235 courses have implemented some parts of requirements.

In parallel, the student teams from CE-230 course have checked the product considering quality (compliance with requirements), reliability (quality on time), and safety (reliability on use).

The student teams from CE-237 course have tested the product, in order to reduce its risks and better the cost-benefit ratios. This was done, by adopting Scrum and its best practices. Tests were applied and reviewed, during each sprint, following the concepts of continuous improvement by using the Plan-Do-Check-Act (PDCA) technique [11][19].

In addition to reducing risks and improving software quality, reliability, and safety on final product, the use of previous mentioned models, methods, tools, and techniques also provided frequent deliveries and intermediate features fully developed, and better development speed and business
value for the entire software, hardware, and end product system.

The development teams (from CES-63 and CE-235 courses) were composed of 41 students. The quality auditor teams had 15 students and the tester teams 12 students. By working together, they were able to deliver a product prototype specified by the customer, within a time frame of only 17 academic weeks.

Table IV illustrates some of the achieved results during the construction of this academic Smart Grids System prototype.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Meters</td>
<td>6</td>
</tr>
<tr>
<td>Initial requirements</td>
<td>127</td>
</tr>
<tr>
<td>RRRT-coded lines</td>
<td>26,811</td>
</tr>
<tr>
<td>Hand-coded lines</td>
<td>1,950</td>
</tr>
<tr>
<td>Total lines of code</td>
<td>28,770</td>
</tr>
<tr>
<td>Files</td>
<td>104</td>
</tr>
<tr>
<td>Developers</td>
<td>41</td>
</tr>
<tr>
<td>Quality Auditors</td>
<td>15</td>
</tr>
<tr>
<td>Testers</td>
<td>12</td>
</tr>
<tr>
<td>Development Time (in weeks)</td>
<td>17</td>
</tr>
</tbody>
</table>

Unlike the traditional method of waterfall, where time, resources, and scope are set in advance right at the beginning of the project and cannot be changed, the use of Scrum Agile Method and its best practices has opened the scope where changes were allowed.

Exploring this novelty, the project scope had numerous changes, from simple changes in User Stories to complex ones, remodeling its architecture, but all of them always aligned with customer needs and business values.

This result can be seen in Table V, which highlights the information in chronological order and relates the evolution of prototype with iterations and integrations performed.

<table>
<thead>
<tr>
<th>Steps</th>
<th>Description</th>
<th>Number of User Stories</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>User Stories</td>
<td>127</td>
</tr>
<tr>
<td>Step One</td>
<td>Selection: Technical and Market Needs</td>
<td>144</td>
</tr>
<tr>
<td>Step Two</td>
<td>Grouping: Business Value Filtering and Redundancy</td>
<td>58</td>
</tr>
</tbody>
</table>

VIII. CONCLUSION

Students from the Brazilian Aeronautics Institute of Technology (Instituto Tecnologico de Aeronautica - ITA) taking the Computer Engineering Undergraduate Program and the Electronic and Computer Engineering Graduate Program, on the 2nd semester of 2011, were submitted to the experience of integrating the different courses of: CES-63/CE-235 Embedded and Real-Time Systems; CE-230 Software Quality, Reliability, and Security; and CE-237 Advanced Topics in Software Testing. The practical exercises of these courses have provided the students with high level of understanding and problem solving capabilities.

The major finding of this project was the successful involvement of 68 students (41 developers, 15 quality auditors, and 12 testers) from different locations and courses, through the use of I-CASE-E and collaboration tools, implementing an embedded and real-time system, tested and checked for quality within 17 weeks.

The thematic of Smart Grids Systems as a Problem Based Learning (PBL) was important to motivate students who had produced documents management, modeling diagrams, systems using I-CASE-E tools, quality statistics, and test cases for a real project. The experiment was conducted in an academic environment, but has resulted in a functional prototype, making it possible to stimulate future work in a real and effective implementation of a Smart Grid System.

The integration of courses and teams has demanded different syllabi planning and collaborations among teachers, students, and academy. Unfortunately, this type of initiative has been rarely reported in educational environments around the world. The teachers evolved in these courses played an important role, teach and broadcast the knowledge between the teams, providing the basis for the Smart Grids development. Once each student chosen a team during the courses, this group can’t until the development finishes.

The main advantages observed about applying agile methods against traditional methods in the Smart Grids development were: (a) better commitment between teams and the objectives; (b) project development process more suitable to deal with uncertain and adjustments; (c) more deliverables for the customers; (d) and better productivity.

In order to improve the Smart Grids development, managing, tests and communications, a set of programs were widely used. For example, the computational programs Skype, Google Docs, Rational Tools, Office, and Virtual Machines.

Aiming to improve the learning process in the Computer Engineering Program at undergraduate level and the Electronic and Computer Engineering Program at graduate level, the authors of this paper recommend the use of integrated courses and troubleshooting combined with practical exercises, such as the ones performed on the project reported in this paper.

ACKNOWLEDGMENT

The authors of this paper thank the Brazilian Aeronautics Institute of Technology and its Computer Engineering Undergraduate Program and Electronic and Computer Engineering Graduate Program for supporting the development of this work and accepting the challenges of innovation and interaction within the proposed courses of CES-63/CE-235, CE-230, and CE-237, during the 2nd semester of 2011.
REFERENCES


