NFR Catalogues for RFID Middleware

Renato Cristiano Torres
AESA Institutional
Jundiaí – Brazil
e-mail: renato.torres@aesapar.com
and
Luiz Eduardo Galvão Martins
Federal University of São Paulo (UNIFESP)
Institute of Science and Technology
São José dos Campos – Brazil
e-mail: legmartins@unifesp.br

ABSTRACT

Non-functional requirements (NFR) are related to the user satisfaction about the quality attributes of the information system. In some cases these requirements are ignored or implemented by the end of the project in a chaotic way. It happens because, in many cases, the user does not have enough contact with the information system to solve these requirements, such as non-functional requirements for Radio-Frequency Identification (RFID) middleware. This study presents the preparation of non-functional requirements catalogues for RFID middleware supported by Non-Functional Requirements Framework (NFR-Framework). Two case studies were performed to evaluate the Requirements Engineering process in the creation of the NFR catalogues and the effectiveness of the reuse of the catalogues. As a result, a set of non-functional requirements are presented and organized into catalogues that work as the foundation for RFID system developers in the identification and validation of non-functional requirements for RFID middleware in information systems context.

Keywords: Requirements Engineering, RFID Middleware, NFR-Framework, Non-Functional Requirements.

1. INTRODUCTION

The need to identify physical objects is essential for companies in the areas like logistics, manufacturing, aviation, security and hospital. The radio frequency identification (RFID) uses a wireless system to identify objects precisely and without the necessity of a line of vision of the object, so that the target-object can be covered by several other objects and even so it can be identified [05].

The RFID identification, in many cases, eliminates the user's need to perform the identification process. Since it does not depend on the alignment with the object, the identification process is automated by the computer systems dedicated to it. This elimination or reduction of human intervention is one of the main advantages presented by the RFID systems, generating savings and efficiency to the companies with deployed RFID systems [16].

These systems usually rely on various quality attributes, e.g.: accuracy in data capture, high availability (ensuring the system do not stop), among others. These are some quality attributes the RFID systems must have to satisfy the users and their businesses [3].

As these quality attributes are often implied, there is the need of understanding in detail quality attributes of the RFID and how developers can elicit them. This article will report a component of the RFID systems known as RFID middleware, which objective is to obtain data from the hardware responsible for the capture and delivery of identification data to the users application, as well as the use of *NFR-Framework* to elicit and manage the quality attributes required for the middleware RFID resulting in the creation of a knowledge base to be reused called non-functional requirements catalogue.

Catalogues form a knowledge base and demonstrate a detailed relationship among non-functional requirements, which are essential to understand the impact of certain actions of prioritization, operation and constraints that might affect the non-functional requirements. Therefore the creation of catalogues to non-functional requirements has a deserved importance, considering the historical cases of projects that were affected by the absence or lack of understanding of such requirements [2][8].

A framework which provides techniques for processing, representing and management of non-functional requirements, was used to prepare the catalogues presented in this paper. Unlike most approaches, the NFR-Framework proposed by Chung [4] uses non-functional requirements to guide the developing process through the construction of Strategic Interdependency Graph (SIG) that records the reasoning of the treatment of non-functional requirements, defined in the catalogues of types, methods and interdependencies.

This article presents the problem of non-functional requirements elicitation in autonomous systems, meaning with little or no user interaction, through the use of catalogues of non-functional requirements in the construction of RFID middleware, application whose user interaction is minimal. The remaining work is organized as follows: section two summarizes the RFID technology basic concepts; section three details the operation and symbols used by NFR-Framework; section four, demonstrates how the NFR catalogues were prepared, section five presents case studies and the conclusion is presented in section six.

2. RFID TECHNOLOGY

The radio frequency identification system was developed with the purpose of automatically capturing data and identifying various goods, such as automotive parts, vehicles and animals and only in early 2000's started being used by retailers as well as being explored by other areas such as logistic, pharmaceutical, hospital and others. It happened due to the miniaturization of chips and to the decreasing of costs of transponders and readers [1]. RFID systems are composed by four elements [4][7]: Transponder, Reader, Antenna and Middleware. The transponder consists of a unit that contains a radio transmitter, receiver and antenna. When the transponder receives a signal from the reader, responds by transmitting a user's unary identification code along with any other data previously stored in its memory. The transponder is also known as RFID tag [6].

Basically the RFID reader radiates a magnetic field through one or more antennas. The transponder picks this signal up through its antenna and changes it into energy by induction. This power is enough to energize the circuit and thus the transponder develops its identity to the reader. The middleware has its particular importance because it is an application layer that communicates with readers and share captured data with other applications. An RFID middleware must have the following features [9][17]:

- Flexibility to change in the business rules;
- Capacity to integrate with other technologies;
- Effective architecture to handle large amounts of data;
- Security access to data;
- Interoperability with various devices;
- High reliability to critical missions.

Oug *et al.* [10] describe an RFID middleware structure, as showed in Figure 1. It can be observed that it is software that will have the rules and parameters of the business in its implementation and it will intercept the data exchange between readers and the users' final application. It will also be able to make decisions upon the received data.

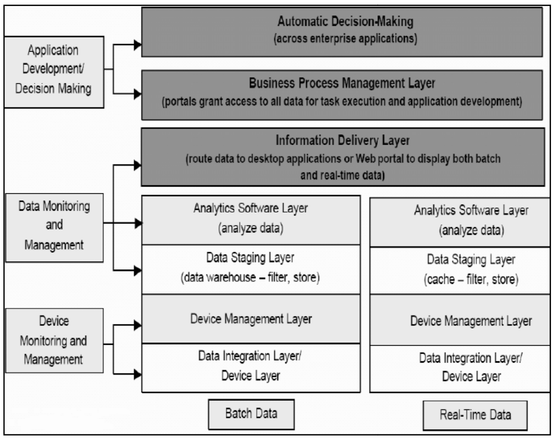

Figure 1- RFID middleware structure proposed by Oug *et al.* [10].

This information could be available in batch or in real time. Each layer interacts with a type of information and makes it available on a consolidated basis to the corporative systems, making the communication with the end user effective. Each reader has a specification driven by the manufacturers to exchange data, which provide communication libraries to their RFID readers, which may be different so the middleware has as its main goal to hide from the end user all the communication complexities, data capture, devices and interaction with other systems. Fulfilling this goal generates the need of the application to have numerous non-functional requirements to satisfy the user.

3. NFR-FRAMEWORK

The NFR-Framework is based on goal specifications that are catalogued and expressed through graphs called SIG (Softgoal Interdependency Graphs). Unlike other goal-oriented approaches, NFR-Framework addresses the development through SIG [4]. *NFR*-Framework allows creating a historical base of knowledge, so catalogues are reused as a starting point for the development of new systems with the same context. This reuse provides benefits related to the form of declaring and treating non-functional requirements. Basically NFR-Framework consists of five components: Softgoals, interdependency graphs, evaluation procedures, refinement methods and correlation rules between requirements. Softgoals are classified into three types:

- **NFR Softgoal** – Represents non-functional requirements;
- **Operationalisation Softgoal** – Represents non-functional requirements operationalisation;
- **Claim Softgoal** – Informs a specific need "to satisfice" a goal.

Figure 2 - Graphical Representation of softgoals proposed by Chung [4].

The refinement of a softgoal is the decomposition into subgoals in order to detail the softgoal. This technique breaks the softgoal into other softgoals with no intent of creating new softgoals but to eliminate any ambiguity. Subgoals can be detailed through a type AND or type OR relationship according to Figure 3.

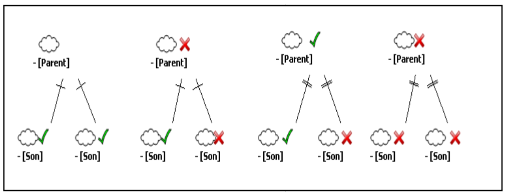

Figure 3 - Types of decompositions and contributions to achieve the softgoal, (figure created with the tool StarUML, with plugin NFR-Framework).

A major goal of refinement is to decompose an NFR softgoal until achieving their operationalisation softgoals in charge of satisfying it. The next step is to assign the satisfaction degree of the softgoal in its interdependency relationship to other softgoals, which may take several states, as in Figure 4.

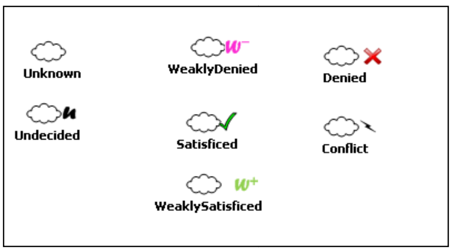

Figure 4 - States of a softgoal proposed by Chung [4].

The meaning of the softgoal states is explained as follows:

- **Satisfied**, the goal was satisfactorily reached;
- **Denied**, refers to the state in which the goal was not achieved;
- **Weakly Denied**, the goal was partially unsatisfied;
- **Weakly Satisfied**, the goal was partially satisfied;
- **Conflict**, this state occurs when a softgoal is satisfactory to a refinement and, at the same time is unsatisfactory to another;
- **Unknown**, refers to a state of a *softgoal* which evaluation is inconclusive;
- **Undecided**, is the state of a softgoal that was not evaluated.

Contributions may combine various types of influence: total or partial, negative or positive. The contribution to partially positively satisfy a parent softgoal is represented by ("+" or HELP), which maintains the son positive contribution sign, but weakens the satisfaction. The negative is represented by ("-" or HURT) used to inform that the son softgoal negatively influences and weakens the satisfaction of the parent goal but maintains the contributions sign. The totally positive represented by ("++" or MAKE) maintains the sign of the son softgoal. Totally negative represented by ("--" or BREAK) informs that the son *softgoals* do not satisfy the parent softgoal.

The purpose of SIG is to represent the interdependencies among softgoals, operationalisations and contributions the softgoals perform. These contributions modify the softgoals states, which are also represented in the graph, assisting decision making by performing or not given operationalisation.

NFR-Framework stores the knowledge expressed in SIG into catalogues with the purpose of structuring and enabling the reuse of the knowledge. To do so, it classifies the catalogues sorting the knowledge by subject. Figure 5 presents a type NFR catalog, which demonstrates NFR softgoals, i.e. non-functional requirements.

Catalogues of methods are responsible for expressing operationalisation routines to satisfy the referred goal. As well as the catalogue of types is expressed in a hierarchical tree, in its root more general methods can be found, and in its leaves, more specific methods. Figure 6 presents the catalog of methods to operationalize the softgoal *confidentiality*, considering this goal within the domain of maintaining the security of a bank account.

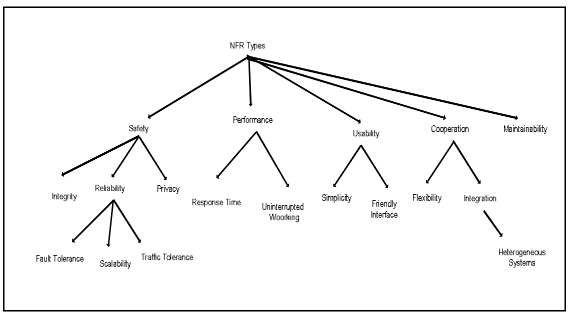

Figure 5 - Example of catalog of types for RFID middleware.

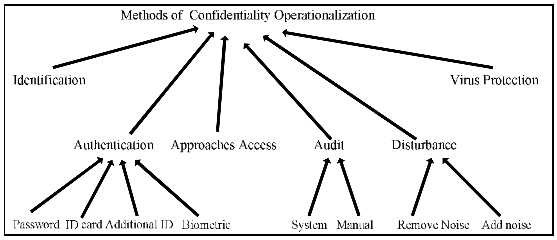

Figure 6 - Catalog of Operationalisation Method to Perform Reliability, adapted from Chung [4].

Correlation catalogues demonstrate softgoals interdependencies and their contributions in the relation. Table 1 presents data demonstrating, for example, that the "Additional ID" operationalisation softgoal is required to enhance confidentiality, but it causes a negative contribution to the softgoal usability. It is due to the fact that the "Additional ID" operationalisation will force the user entering more data when authenticating, thus negatively impacting the way the authentication is used. It is clear that this impact is subjective and is also why the developer must make decisions concerning the form and types of contributions in the relations, ensuring the satisfaction of all possible softgoals.

Table 1 – Catalog of correlation.

Catalog of Correlation					
Sons Operationalisation Softgoals	*NFR Softgoal* NFR (Non-Functional Requirement)				
	Precision	Confidentiality	Team in Charge	Space	Usability
Validation	+	+	-		
Compression			-	+	
Indexation			+		
Authentication		+			
Additional ID		+			-

4. RELATED WORKS

In [10] Oug et al. examined functionality, reliability, usability, efficiency and portability among the quality characteristics of software in international standard ISO/IEC 9126, as well as the quality elements of standard RFID middleware of EPC Global. Based on such analysis they extracted some items for evaluating the quality of RFID middleware in ubiquitous and pervasive computing systems. Using the AHP - Analytic hierarchy process, they evaluated the subjective characteristics of stakeholders and proposed a selection method to evaluate quality.

The proposed selection method is useful to develop RFID middleware in areas such as distribution and logistics to select RFID middleware suitable for their environment [10].

In [18] Koskela et al. presented a framework that enables development of ubiquitous web applications that combine both physical and virtual worlds. They argue that applications based on their framework are well scalable and can be administered remotely via web. As mobile device manufacturers are adopting Near Field Communication (NFC) it is gradually becoming more pervasive. Eventually, this may lead to smart physical spaces where everything can be interacted with by touching.

5. PREPARATION OF NFR CATALOGUES FOR RFID MIDDLEWARE

The preparation of catalogues established a knowledge base supported on *NFR*-Framework to assist RFID developers in the analysis and implementation of solutions involving RFID middleware for enterprise information systems. The preparation process was divided in three groups: selection of non-functional requirements supported on the work of Oug *et al.* [10]; interview with users of RFID based information systems; development of SIG for each non-functional requirement using NFR-Framework. To do so, the type NFR catalogues, catalogues of methods, catalogues of correlation and SIG were developed.

5.1 Selection of Non-Functional Requirements

The proposal of Oug *et al.* [10] is a technique of evaluating the quality of non-functional requirements implemented in an RFID middleware, based on ISO/IEC 9126 rule and presents classes of non-functional requirements desirable in RFID middleware for user satisfaction. During the study, 16 non-functional requirements for RFID middleware were selected and related to the classification, according to Table 2.

Table 2 – Classes of non-functional requirements for RFID middleware proposed by Oug et al. [10].

NFR Softgoals Selected	
Uninterrupted operation	Reliability
Performance	Safety
Scalability	Flexibility
Response time	Tolerance to traffic
Support to Heterogeneous Systems	Simplicity
Friendly Interface	Cooperation
Usability	Privacy
Fault Tolerance	Maintainability

5.2 Identification and Prioritization of NFR Softgoals

NFR softgoals are the system supergoals and are associated to the non-functional requirements class for middleware, based on the work of Oug *et al.* [10]. However, it is noted that only the supergoals would not be enough to achieve a level of detail to implement an RFID middleware. It is necessary to elicit the non-functional requirements with a greater level of detail and the user's point of view, since some constraints are declarative and usually are not explicitly presented, as is the case of temporal constraints that will be represented later as a claim softgoal. Another aspect is that within the class of non-functional requirements there are several decompositions to be performed to detail each NFR softgoal, and within this subset resulting from the decomposition are other operationalisation non-functional requirements, which in turn, are responsible for the NFR softgoal satisfaction with a greater level of detail.

As an example of identification and prioritization of NFR softgoals, interviews with eight different users were performed to elicit non-functional requirements for RFID middleware. Each user was inserted in a different line of business. There were logistic managers (4), administration (2) and production (2), all experts in their field of work, none of them was a computer specialist, though they had reasonable computer knowledge along with experience in working on projects involving RFID systems. The objective was to associate their non-functional needs to the listed NFR softgoals. After this review, the requirements were classified and prioritized as Critical, Important and Helpful (according to Table 3).

Table 3 – Table of quantification of the priorities selected by the users.

NFR Softgoal	Priority
Uninterrupted Operation	Helpful(0) Important(2) Critical(6)
Performance	Helpful(0) Important(5) Critical(3)
Scalability	Helpful(0) Important(1) Critical(7)
Integrity	Helpful(0) Important(3) Critical(5)
Support to Heterogeneous Systems	Helpful(2) Important(1) Critical(5)
Friendly Interface	Helpful(5) Important(2) Critical(1)
Usability	Helpful(6) Important(1) Critical(1)
Fault Tolerance	Helpful(0) Important(1) Critical(7)
Reliability	Helpful(0) Important(0) Critical(8)
Safety	Helpful(2) Important(0) Critical(6)
Flexibility	Helpful(5) Important(1) Critical(2)
Traffic Tolerance	Helpful(0) Important(1) Critical(7)
Simplicity	Helpful(7) Important(1) Critical(0)
Cooperation	Helpful(4) Important(2) Critical(2)
Privacy	Helpful(0) Important(1) Critical(7)
Maintainability	Helpful(3) Important(4) Critical(1)

Users also received a card with NFR softgoals to check one of the three priority options (helpful, important and critical, see Table 4 - the number in parenthesis represents the total amount of choices), along with the definition to these options. After the users were interviewed, a score was rated to each NFR softgoal, then it was possible to choose only the more frequent priorities from their answers. There was not a tie but there were situations when the NFR softgoal was rated as helpful in one business and critical in another. It was due to the fact that there were businesses with different goals. An example is the "Safety" NFR softgoal: in the context of distribution and logistics, it was rated as helpful and as critical in the hospital context since its reliability has to be headed to treat patients' data, medications, exams and others.

Table 4 – Table of NFR Softgoal and its respective priorities.

NFR Softgoal	Priority
Uninterrupted Operation	Critical
Performance	Important
Scalability	Critical
Integrity	Critical
Heterogeneous Systems Support	Important
Friendly Interface	Helpful
Usability	Helpful
Fault Tolerance	Critical
Reliability	Important
Safety	Critical
Flexibility	Helpful
Traffic Tolerance	Helpful
Simplicity	Helpful
Cooperation	Helpful
Privacy	Critical
Maintainability	Important

This prioritization is important to solve possible conflicts and guide decisions making when a goal has a negative influence over another in a way that this one cannot achieve the desired satisfaction. Then one of the priorities is rated as the most important for the system and its specification is negotiated with the user. It makes possible to anticipate the satisfaction problems and eliminate possible errors with partial requirements satisfaction that usually lead to work over the specification and implementation again.

5.3 NFR Catalogues

The classes of non-functional requirements selected were arranged in a tree, presenting their interdependency relationship according to NFR-Framework, as shown in Figure 7.

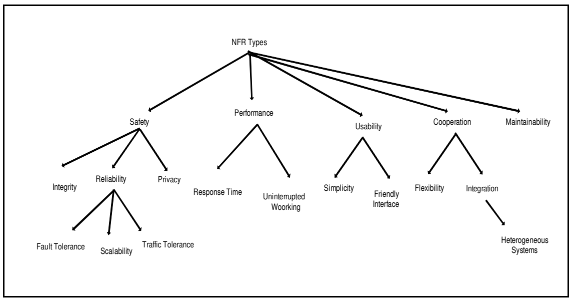

Figure 7: Catalog of NFR Types for RFID middleware.

NFR softgoals decompositions were performed based on their definitions. For example, the safety softgoal is performed through its three NFR softgoals: integrity, reliability and privacy.

The composition of the method catalogues was divided into three phases: NFR Softgoal decomposition when it is needed, identification of operation method(s) and its SIG graph representation. NFR-Framework enables presenting the method catalog through a bottom-up directed graph (Figure 6), or through a SIG graph using an operation softgoal claim when it is necessary. The SIG graph was chosen because the operation interdependencies and their decompositions are not clear when represented by arrows. The method catalogues were created by noting the reuse possibility, so they have generic methods for RFID middleware, as it is shown in Figure 8 with a tree shaped method catalog for fault tolerance NFR softgoal.

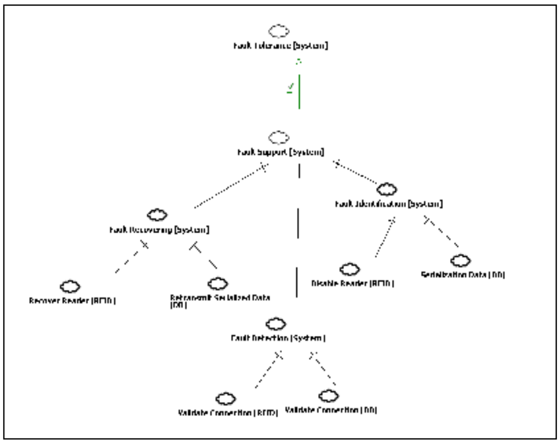

Figure 8 - Method Catalog created for a NFR Fault Tolerance Softgoal represented using SIG.

In Figure 9 a SIG graph of a fault tolerance NFR softgoal for RFID middleware is presented with restrictions shown by the claim softgoals and the negative contributions that affect other NFR softgoals. This process of catalogue elaboration was applied for each NFR type for RFID middleware. Considering the pages limitation in the paper, only the catalog for fault tolerance softgoal was presented (Figures 8 and 9).

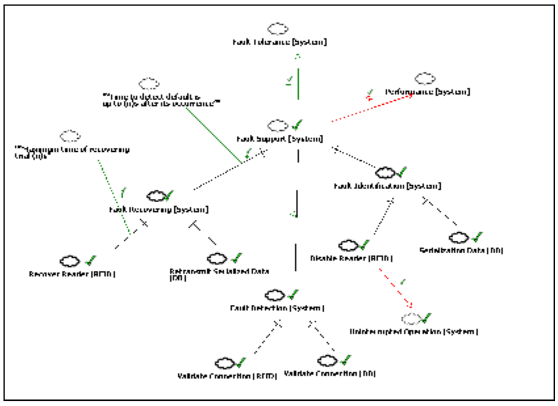

Figure 9: SIG graph of fault tolerance NFR softgoal for RFID middleware with claim softgoals.

The "Disable Reader" operation contributes negatively to the Uninterrupted Operation NFR softgoal not "satisficing" this softgoal if it is performed. The Fault Support is also a negative contribution to the performance, since the mechanisms for recovering, detecting and identifying faults impact the cost of processing, leading to a partial satisfaction of the NFR softgoal performance [15][11]. When they occur SIG graph demonstrates its importance because it makes possible the decision making of which and how the requirements will be operated, considering the contributions and states of the softgoals. All the decisions of contributions are made under the developer's point of view and interpreting the user's needs.

6. NFR CATALOGUES APPLICATION

Two case studies were performed in different businesses context to evaluate the benefits of the NFR-Framework and the catalogues elaborated for RFID middleware. The first case study was performed in a paper and cellulose industry and its goal was to build an RFID middleware to track the loading of raw material, using the traditional method [12][13][14], and to build NFR catalogues along with the requirements engineering process. Then it was possible to compare characteristics of the traditional method and the NFR-framework. The RFID middleware has to obtain captured data from RFID readers, make them available to the management systems and make decisions to certain situations of discharging the material (Figure 10 illustrates the general architecture adopted). The second case study was performed in a context of a chemical industry and the goal was to specify and implement a solution to track loading and downloading of bulk chemical material, providing data to the management systems and to the monitoring central of the production plant, through the reuse of NFR catalogues previously created. The solution consists of RFID middleware and a graphical application to interact with the user (Figure 11 illustrates the general architecture adopted).

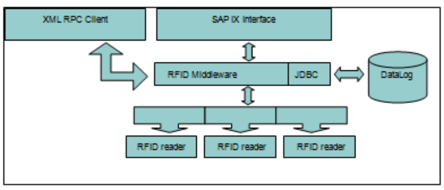

Figure 10 - Middleware architecture to case study I.

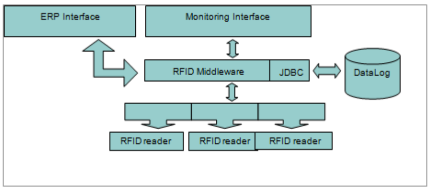

Figure 11: Middleware architecture to case study II.

Users monitor and interact with RFID middleware through a monitoring panel as shown in Figure 12. This cooperative application was developed with the purpose of performing the interface between middleware and user, using an interface enriched by graphics resources.

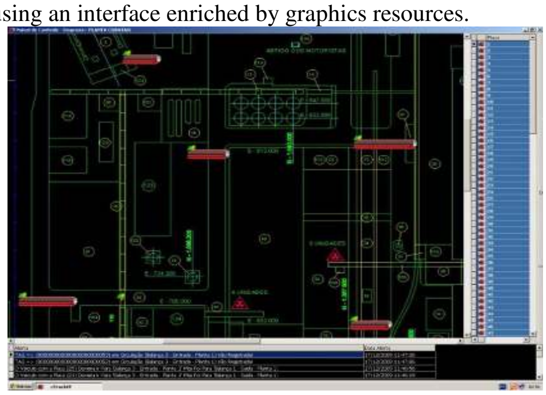

Figure 12: Monitoring Panel of the vehicles tracking systems that uses the RFID middleware in case study II.

6.1 Case Study I: NFR-Framework versus Traditional Model

A comparison of the Traditional Model and NFR-Framework was performed to present the differences, benefits and use of the models to show the RFID middleware specification. In the traditional model the elicitation process begins with an interview with the users to survey the needs that were analyzed, classified and catalogued using use cases models. In the first case study, a second interview with the same users was performed using the previously created catalog of types of NFR-softgoals to conduct the interview. By doing so, a process of acceptance of the non-functional requirements proposed by the users was performed, though the questioning about other possible requirements was not eliminated. The use of the catalog of types of NFR softgoal was total. The use of the catalog of types captured much more requirements (presented in Table 5 and 6) than the traditional model because there were other implied softgoals, which were detailed and presented to the user through an NFR catalog of types. It can be explained because the users have no acknowledgement of such requirements and because of the developer's lack of experience when specifying the RFID middleware.

Detailing NFR softgoals was possible by the use of catalogues of NFR-Framework types which presents the softgoals in a structured way making possible the understanding of its relations to other NFR softgoals, facilitating the composition understanding and its discussion with the user.

Table 5 – Quantity of non-functional requirements elicited in each model.

Non-Functional Requirements	Quantity
NFR-Framework	18
Traditional	8

With the catalog of non-functional NFR softgoals validated by the users, the process of creation of other catalogues and SIG graphs was initiated. It could be noted that the NFR-Framework played its role in the analysis phase, since its use enriched the refinement of non-functional requirements and, as a result, lead to the findings of new functional requirements, as presented in Table 6.

Table 6 - Quantity of functional requirements elicited in each model.

Functional Requirements	Quantity
NFR-Framework	48
Traditional (based on Use Cases)	18

Though the NFR-Framework is focused on non-functional requirements, it contributes a lot to the findings of new functional requirements in the phase of analysis of requirements, when the creation of new catalogs is initiated, as in this case. Most of the functional requirements were linked to non-functional requirements.

On the other hand, it could also be observed that the use of NFR-Framework turned the analysis and validation of requirements slower and expensive. Table 7 presents the relation of time among requirement activities to: elicitation, analysis, specification and validation of requirements in the traditional way and using NFR-Framework. It is due to the fact that in these phases the catalogues and decompositions that makes the activity slower than the traditional process, which only describes the non-functional requirement (NFR softgoal). Of course this cost is acceptable because the requirements finding was highly improved when NFR-Framework methodology was adopted.

Table 7 – Comparison of productivity in each step to non-functional requirements.

Activity	Traditional Method (min/req)*	*NFR-Framework* Method (min/req)**	Dif (%)
Elicitation	15,6	9,3	59,61
Analysis	11,4	24,3	113,10
Specification	4,30	4,5	1,32
Validation	6,10	26,2	329,50

** Time is the average and it were extracted from a base of knowledge in a software house with about 3000 non-functional requirements, 4 analysts with 3 year experience.*
*** Time calculated to perform the activity using the NFR-Framework*

6.2 Case Study II: NFR Catalogues Reuse

Differently from the first case study, in the second study only NFR-Framework was used to specify and implement RFID middleware. The catalogues created were used to guide elicitation, analysis, specification and validation on the non-functional necessities of the users.

Table 8 – Comparison of productivity in each step.

Activity	Traditional Method (min/req)*	NFR-Framework Method (min/req)**	Dif (%)
Elicitation	15,6	9,3	59,61
Analysis	11,4	15,1	32,46
Specification	4,30	4,5	1,32
Validation	6,10	19,0	211,48

** Time is average and was extracted from a basis of knowledge in a software house with a little more than 3000 non-functional requirements.*

*** Time assessed to perform the activity reusing catalogues for RFID middleware.*

As showed in Table 8, comparing with the results in Table 7, the productivity increased relatively using NFR-Framework method, highlighting the analysis of requirements, since the decomposition work and the NFR softgoal selection were done (because the catalogues of RFID middleware requirements prepared were totally reused). However, it is important to stress that validation of requirements has a reasonable influence in the process effectiveness, because when the context of the business changes it is necessary to perform innumerous validations to verify not only if the catalogues prioritization are according to the model of the business, but also if they are sufficient, since new necessities can be included.

It actually happened in the second case study because there was a new NFR softgoal "Space" that was included and directly related to the NFR softgoal "Performance". The context of this business had a specific need to form and guarantee of captured data storage and this specific need lead to new decision makings related to the dependency on the softgoals.

7. CONCLUSIONS

This paper reveals a methodology to treat non-functional requirements to the requirement engineers and developers, along with improving the elicitation, the requirements specification and consequently the quality of the RFID middleware development. Catalogues of types and methods to 16 non-functional requirements were created and could be reused to elicitation, analysis, validation and specification of RFID middleware requirements, as well as possible efforts to use the NFR-Framework to specify the RFID middleware were quantified.

It is possible that, with some efforts, the catalogues can be used to other similar information systems within the RFID context since non-functional requirements are possibly the same or with a few differences because the middleware is a subset of the RFID applications. Catalogues assist developers with little expertise and becomes a guide to develop applications involving RFID middleware. For more experienced developers it becomes a tool of critical analysis, which allows performing numerous verifications on the users' needs, therefore anticipating the decision making about any item that could influence the success of the project. In both cases, it can reduce or even eliminate possible failures on the identification of functional and non-functional requirements. As future work we intend to develop a software tool to facilitate the adoption of the catalogues proposed to reuse RFID middleware requirements.

8. REFERENCES

[1] Al-Amir, Z., Al-Saidi F. A., Abdulkadir, H.: Design and implementation of RFID System; 5th International Multi-Conference on Systems, Signals and Devices, p. 1-6 2008.

[2] Al-Kassab, J., Rumsch, W. C.: Challenges for RFID Cross-Industry Standardization in the Light of Diverging Industry Requirements. IEEE System Journal, v. 2, issue 2. p. 170-177 2008.

[3] Bhattacharjya, A., Pal, R. K.: Distributed design of universal lightweight RFID system for large-scale RFID operation. Proceedings of the IEEE International Summer Conference of Asia Pacific on Business Innovation and Technology, p. 40-44 2011.

[4] Chung, L., Nixon, B., Yu, E., Mylopoulos, J.: Non-Functional Requirements in Software Engineering, 1st edition, Kluwer Academic Publisher 2000.

[5] Floerkmeier, C., Lampe, M.: RFID Middleware Design – Addressing Application Requirements and RFID Constraints. Proceedings of the Joint Conference on Smart Objects and Ambient Intelligence: innovative context-aware services: usages and technologies, New York, 2005. p 219-224 2005.

[6] Glover, B., Bhatt, H.: Fundamentos de RFID, 3.a Edição, Editora Érica, Brazil 2005.

[7] Hind, D. J., Derby, D.: Radio Frequency Identification and Tracking System in Hazarduos Areas. Proceedings of the Fifth International Conference on Electrical Safety in Hazardous Environment, London. p. 215-227 1994.

[8] Ilie-Zudor, E., Kemeny, Z., van Blommenstein, F., Monostori, L., van der Meulen, A.: A Survey of Applications and Requirements of Unique Identification Systems and RFID Techniques. Computers in Industry, Elsevier. p. 227 – 252 2010.

[9] Li, C-M. J.: An Integrated Software Platform for RFID-Enabled Application Development. Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, vol. 1, Taichung. pp 4 2006.

[10] Oug, G., Kim, Y. D., Kim, S., Rhew, Y. S.: A Quality Evaluation Technique of RFID Middleware in Ubiquitous Computing. International Conference Hybrid Information Tecnology, vol.2, p. 730-735 2006.

[11] Pradhan, D. K.: Fault-Tolerant System Design, Prentice Hall, 1st edition, New Jersey 1996.

[12] Sommerville, I., Sawyer, P.: Viewpoints: Principles, Problems and a Practical Approach to Requirements Engineering. ACM Annals of Software Engineering. vol. 3. p. 101-130 1997.

[13] Sommerville, I.: Integrated Requirements Engineering: A Tutorial. IEEE Software, vol. 22, n.1. p. 16-23 2005.

[14] Sommerville, I.: Software Engineering. 9th edition, Addison-Wesley 2010.

[15] Weber, T., Jansch-Porto, I., Weber, R.: Fundamentos de Tolerância a Falhas. Anais da IX JAI - Jornada de Atualização em Informática, X Congresso da Sociedade Brasileira de Computação, Vitória 1990.

[16] Zuo, Y., Pimple, M., Lande, S.: A Framework for RFID Survivability Requirement Analysis and Specification. Innovation in Computing Science and Software Engineering, Springer Netherlands, p. 153-159 2010.

[17] Yang, P., Wu, W., Moniri, M., Chibelushi, C. C.: Efficient Object Localization Using Sparsely Distributed Passive RFID Tags. IEEE Transactions on Industrial Electronics, v. 60, issue 12, p. 5914-5924 2013.

[18] Koskela, M., Ylinen, J., Loula, P.: A Framework for Integration of Radio Frequency Identification and Rich Internet Applications. 29th International Conference on Information Technology Interfaces, Croatia, p. 691-695 2007.

Received: April 2014. Accepted: August 2014.