

Usability Quality Aspects embedded in the Business Model

Juan Carlos Moreno¹, Marcelo Martín Marciszack¹, Mario Alberto Groppo²,

Centro de Investigación, Desarrollo y Transferencia de Sistemas de Información (CIDS),
Universidad Tecnológica Nacional – Facultad Regional Córdoba,
Maestro Lopez esq. Cruz Roja Argentina S/N, Ciudad Universitaria,
Córdoba, República Argentina
{jmoreno33, marciszack}@gmail.com ¹, sistemas@groppo.com.ar ²

Abstract. This work is a proposal for the identification Non Functional Usability Requirements Specification, at an early stage. The procedure starts in a business processes modeling domain, where essential specifications are captured. Then they are mapped to a structure called Baseline Requirements using the concept of paradigm transformation from Model Driven Software Design. In this context, usability specifications are introduced inside the scenarios using transformations, and take part in these scenarios by using Lexicon Extended Language (LEL). The use of LEL allows to describe the usability specifications and the construction of a dictionary, which will later be needed for integrating the usability specifications in the software programs developed. This also provides the possibility to identify, define and maintain the traceability of usability specifications defined at early stages.

Keywords: Usability, Business Processes Model, Scenarios, Lexicon Extended Language, Model Transformation

1 Introduction

Developing applications efficiently has been one of the main worries in Information Systems Engineering, without omitting to consider the quality of those applications. This could be a consequence of several causes, such as: the advance of nanotechnology in microprocessor development leading to continuously smaller devices, the great technological progress in telecommunication networks and the expansion of Internet, increasing the need of web applications all over the world. Its impact can be seen not only in interactive applications through WEB 2.0 [1], but also in Social Networks [2]. Functional Requirements are usually given more relevance, independently from the software development methodology used. But product quality not only depends on these latter (FR), but also on Non Functional Requirements (NFR). What has been exposed raises the question if it is feasible to develop a methodological proposal that allows the introduction of usability requirements at early stages in software development, always starting from Software Engineering point of view.

Thus, the main goal was to develop a systematized procedure that allows, introducing usability aspects at early stages of software development life cycle, the analyst to rely on usability specifications before the software development is finished.

This work is structured as follows: the first part is an introduction of the state of the art, elements and concepts used to develop the methodological proposal are introduced later. After that, the process that allows to carry out the proposal and the tools used are explained, including an example and some observations on the whole experience. Finally, a conclusion on the final proposal is elaborated.

2 State of the Art

Information systems development in a web platform is a process that requires great knowledge of methodologies, security and diverse technologies, in order to build a useful and correct application for any user. In this way, the goal of Systems Engineering, besides the construction of functionally correct applications, is to construct quality applications through different methods and principles [3]. Often, when building an application or program, the focus tends to be on aspects related to the architecture, persistence and functionality of processes related to functional requirements (FR), not taking into account non-functional requirements (NFR). Additionally, the software quality is frequently measured based in common sense and developers experiences [4].

One of the relevant Non-Functional Requirements, in web application development, is Usability. The concept of usability has been defined by several International Organizations, which establishes rules on Quality Standards (ISO, IEEE). Such rules name Usability as a software attribute, and is related with the quality of such software.

In ISO/IEC 9126-1 [5], usability is considered as a software quality parameter and is one of the relevant characteristics of software. Usability is defined as “the capacity in which a software product can be understood, learnt and used by certain users under certain conditions in a specific use context”. It considers external, internal and in use quality of a software product [6]. Usability is decomposed in sub-attributes, such as ease of learning, comprehension, operativeness and usability compliance [7].

ISO 25000 (Square) [8] considers Usability as a quality aspect under two different points of view: one is software, as a product itself, and the other would be the use capacity, from user perspective in a specific context. The goal is to provide certain criteria to help the analyst to build an integral and usable software product. This implies taking into account certain features in the construction of a product, such as: ease of understanding, learn curve, ease of use, help, technical support, attractiveness and compliance to rules.

It can be seen that Usability concept is evaluated from different perspectives in the different definitions mentioned, considering it as a quality feature inherent and intrinsic to software. Thus, the study of Usability has to be considered in the different stages of the software construction life cycle. Even if it is taken into account, it is important to remember that Usability is considered at a final stage in software construction. At this stage, any modification affects the system architecture, since interfaces are already designed and the cost of any modification is very high [9], [10]. One of the possible

solutions to this problem is to include the analysis and introduction of Usability criteria at early stages, during the requirements elicitation stage.

There are a variety of methodologies for software construction, but a new paradigm of development, called Model Driven Software Development (MDA) has awoken special interest in the last decade, considering the features and advantages that provides for software architecture. Model Driven Software Development (MDA) has standardized the model transformation stages, to develop and build systems that are consistent to the original designed model, created during the requirements elicitation stage. In this transformations process, the traceability of requirements gains momentum since it is necessary to measure the magnitude of the impact of changes, in an update or system modification, and, at the same time, be able to introduce such changes in an automatic and immediate manner. In this proposal, essential requirements will arise from the business model, capturing both functional and non-functional requirements, and applying transformations that will allow the analyst to understand the domain of the problem. In this context and with the scenarios designed, Usability Non Functional Requirements will be worked through the analysis of their attributes.

3. Proposal Description

Elements and concepts used in the development of the proposal method are described in this section. It is described as a process, carried out in two stages.

3.1 First Stage

The first stage consists in the development of a business model, using BPMN [11], considering the following statements in the design of the model:

- Usability specifications, that are modeled, will form part in the non-functional requirements set to be satisfied. BPMN has a stereotype called “business rule”, used for modelling this type of activities.
- Business rules are defined one time and are applied to every activity where non-functional requirements are detected. Therefore, an activity with the stereotype “Business Rule” will be generated for every usability specification and will then be associated with the activities where that requirement is detected. Later, these Business Rules will be mapped to a structure called Requirements Baseline [12], [13]. The Requirements Baseline structure uses scenarios to model the behavior of the system and Lexicon Extended Language (LEL) to represent the domain of the system. The activities of “Business Rule” type will be transformed to LEL entries and scenario restrictions respectively.
- Every activity will be mapped as scenarios, with the exception of those that the analyst defines with “Manual” stereotype, for being considered as non-automatable.

- Activities of “Business Rule” type that are associated with an activity defined with “Manual” stereotype, will neither be mapped as LEL entries nor scenario restrictions.

A model exchange file with XPDL format [14] will be later generated with the modelling tool, containing all model definitions.

3.2 Second Stage

The second stage consists in making the necessary transformations to introduce every definition generated in the previous stage and contained in the XPDL file, in the Requirements Baseline structure.

Baseline Mentor Workbench [15] (BMW) tool is used as mainframe, adding the following functionalities:

- “Restrictions” element is incorporated in scenario definition, containing the association to LEL entries, related to usability specifications.
- Necessary rules for scenario generation and LEL entries are created from the definitions captures in the XPDL file.
- Scenarios and LEL entries have to be manually completed. Scenario descriptions will be used in the system user interfaces generation. The usability requirements for the user interface to be built will be obtained from the captured restrictions and completed in every scenario.

Process and activities carried out in each stage can be seen in Fig. 1.

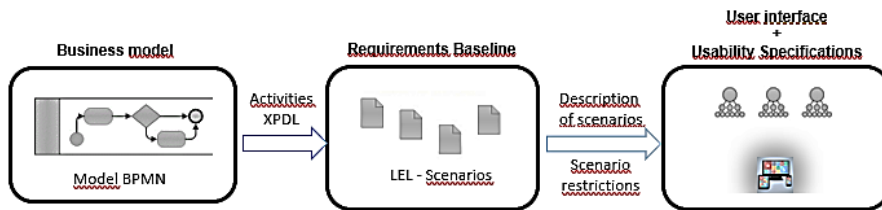


Fig. 1. Usability Requirements Elicitation Process Schema, starting from the BPMN Model.

4. Procedure applied example

A students system, modeled in BPMN, will be used as an example domain to analyze this proposal, more specifically the “Manage Teachers” process indicated in Fig 2.

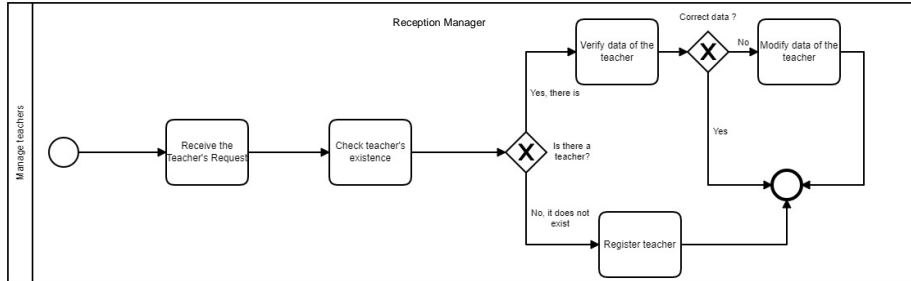


Fig. 2. “Manage Teachers” BPMN model example for a student’s system.

4.1 Usability Criteria Specification

Usability specifications will be modeled as activities using the “Business Rule” stereotype, and will be associated to the activity that has to satisfy the specifications, as shown in Fig. 3.

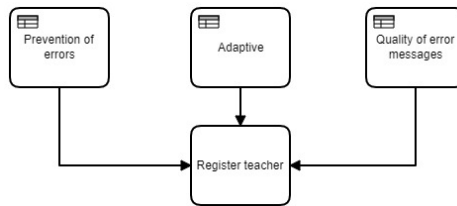


Fig. 3. Usability Criteria Specification starting from the business model.

Every activity will be mapped to the Requirements Baseline structure, with the exception of those selected with the “Manual” stereotype.

Activities selected as “Business Rule” that are associated with “Manual” type activities will not be mapped.

The complete result of the process can be seen in Fig. 4.

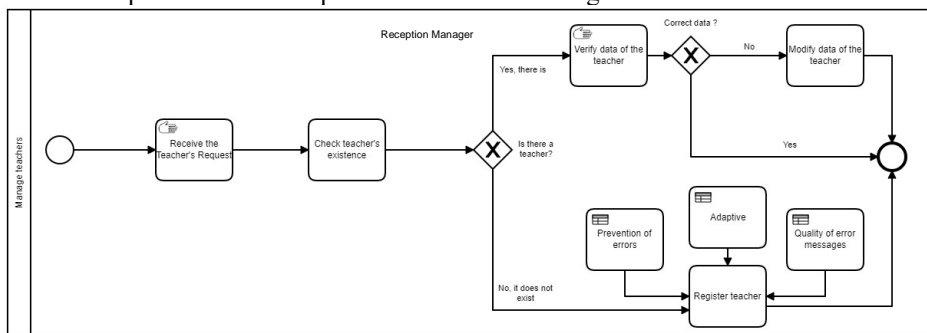


Fig. 4. Result after applying the Usability Criteria Specification starting from BPMN Model process.

Once the BPMN model is complete, the model has to be exported from the tool in a XPD file. This file will be introduced in the BMW tool, and the needed functionalities will be added, in order to process the XPD file with the BPMN model definitions.

The processing of the XPD file will consist in the creation of a scenario for every activity, except in those of “Business Rule” type, which will be introduced inside of LEL as a symbol. LEL symbols are described using a notion corresponding to the meaning of the symbol, and an impact that indicates the effects of the symbol in the system. Each symbol has to be classified according to its function in subject, object, verb or state, and will have different notions and impacts depending on the classification in which they are [17], [18], as indicated in Table 1.

Table 1. Symbol Definition Heuristics.

Symbol	Notion	Impact
Subject	Describes who is the subject	Registers actions carried out by the subject.
Object	Defines the object and identifies other terms which have some relation to the object.	Describes actions that may be applied to the object.
Verb	Describes who executes the action, when it happens and which are the procedures involved.	Describes the restrictions on the action, which actions are started by the environment and the new situations that appear as a result of this action.
State	Describes what it means and which actions can be executed as a consequence of this state.	Describes other related situations and actions.

- Verbs represent the actions carried out in the system. Actions are applied to objects or subjects.
- Subjects are the ones that carry out the actions indicated in verbs.
- Objects represent passive elements that receive the actions indicated in verbs, executed by subjects.
- States are used to describe specific conditions of objects or subjects.
- Usability specifications can be classified inside objects category, since they will be applied or evaluated in a specific moment through an action started by a subject.

LEL symbols corresponding to usability specifications will be described in a way that they comply to two rules simultaneously [16][17]:

-Circularity Principle: limiting language in function of the domain through the maximization of LEL symbols, which is accomplished using symbols already described inside the LEL in the definition of notion and impact.

-Minimal vocabulary Principle: where the task is to minimize the use of symbols external to the application domain.

This will allow to maintain a data dictionary with all the definitions and the hierarchy of usability specifications that need to be satisfied.

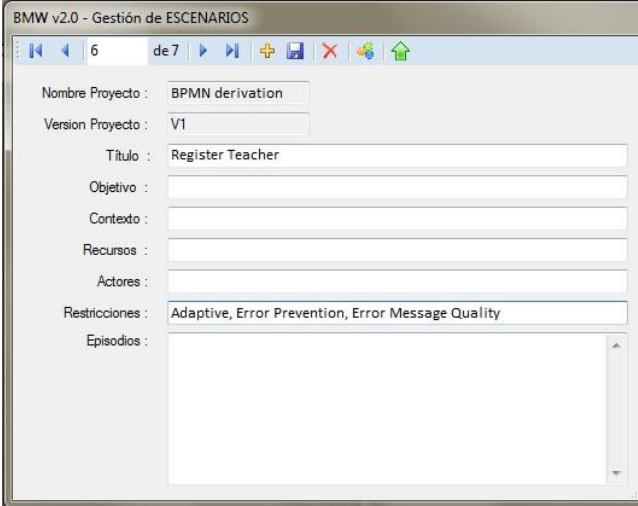
Finally, the analyst will have to complete the description of the scenarios in a similar way as he would do with UML Use Cases [19]. The following will have to be described for each scenario:

- Title: needed to identify the scenario.
- Goal: main goal of the scenario, has to be coherent to the title.
- Context: it is used to describe the initial state, place and moment of execution of the scenario.
- Resources: LEL symbols of object type available for the execution of the scenario.
- Actors: LEL symbols of subject type that carry out actions in the scenario.
- Episodes: They represent the set of actions carried out by actors to execute the scenarios. An episode can appear in different scenarios.

“Check Teacher Existence”, “Register Teacher” and “Modify Teacher” scenarios will be the ones derived directly from activities that do not have “Manual” stereotype.

LEL entries corresponding to usability specification will be automatically mapped inside the scenario under a new element called “Restrictions”.

“Register Teacher” scenario can be seen in Fig. 5, with its corresponding restrictions mapped to the BPMN model. Remaining descriptions must be made by the analyst.



The screenshot shows a software window titled "BMW v2.0 - Gestión de ESCENARIOS". The window has a toolbar with navigation and editing icons. Below the toolbar, there are several input fields for scenario details:

- Nombre Proyecto : BPMN derivation
- Version Proyecto : V1
- Título : Register Teacher
- Objetivo : (empty field)
- Contexto : (empty field)
- Recursos : (empty field)
- Actores : (empty field)
- Restricciones : Adaptive, Error Prevention, Error Message Quality
- Episodios : (empty list area)

Fig. 5. Usability Criteria introduced in the Scenario Management tool.

The resulting scenarios are later used to generate the user interfaces of the system to be built.

This process will allow to identify and define usability specifications at early stages in the development process. On the other hand, the use of LEL allows to generate and maintain a complete data dictionary, both in the definitions and in the hierarchies of usability specifications.

Finally, specifications defined in the BPMN modelling process that, after being mapped to LEL as its vocabulary, form a dictionary of data, will allow the analyst to

maintain traceability of specifications from the start of BPMN modelling to the scenarios used for user interface generation.

5. Results

It is possible to capture requirements at early stages of modelling, starting with the business model. Usability specifications could be modelled inside BPMN, which were later introduced through transformations as LEL symbols in the Object category, taking into account circularity and minimal vocabulary principles. Additionally, the automatic mapping of activities can be done, by using XPDL files, for creation and definition of scenarios, allowing to associate them with usability specifications defined in the LEL vocabulary, in the scenario “Restrictions”. This methodology also provides the possibility to generate a LEL dictionary of data complete with the definitions of usability specifications of transformed models.

6. Discussion

The obtained results confirm that it is possible to carry out the mapping of usability specifications at early stages of software development by applying a combination of methodologies, that will allow to identify, define and maintain the specifications that will have to be taken into account in the process of software generation, that include the user interfaces for the system. There is space for a debate on the use of patterns not only for the definitions of interfaces with usability aspects, but also the possibility to use metrics associated with such interfaces in an early manner in order to measure quality aspects (usability). These prototypes generated from patterns, that comply to usability requirements already predefined, will guarantee certain quality at an early stage and could provide the possibility to introduce certain predefined metrics to evaluate the presence of usability early on.

7. Conclusions

This process allow to identify, define, maintain and improve the traceability of usability specifications at early stages of software development, allowing to know, from the business model, the usability specifications that must be satisfied by user interfaces of the system to be built. The proposed methodology uses Scenarios and Lexicon Extended Language, and by means of transformations it allows to formulate Conceptual Models with Usability criteria specified at an early stage. Obtained results will be used for Pattern Study, starting from Business Models, for the design and construction of Conceptual Models.

References

1. J. Martins, R. Gonçalves, J. Pereira, M. Pérez Cota. "Iberia 2.0: A way to leverage Web 2.0 in Organizations". Information Systems and technologies (CISTI), 2011 7th. Iberican conference.
2. J. Martins, R. Gonçalves, J. Pereira, T. Oliveira, M. Pérez Cota, "Social networks sites adoption at firm level: a literatura review". Information Systems and technologies (CISTI), 2014 8th. Iberican conference.
3. R. S. Pressman, What a tangled Web we weave, vol. 17, IEEE Software, 2000, pp. 18-21.
4. Abrahao S., Condori-Fernandez N., Olsina L., and Pastor O., "Defining and validating metrics for navigational models," IEEE, Ed., Australia: Software Metrics Symposium, 2003. Proceedings. Ninth International, 2003, p. 200-210.
5. Norma ISO/IEC ISO9126-1, "Software Engineering -Product Quality - Part 1," Quality Model, 2001.
6. Nigel Bevan, "Quality and usability: A new framework," Achieving software product quality, 1997.
7. Mario G. Piattini, Felix O. Garcia, and Ismael Caballero, "Calidad de Sistemas Informáticos", México, ISBN 978-970-15-1267-8, 2007.
8. ISO/IEC 25000, Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE).
9. L Bass and B John, "Linking usability to software architecture patterns through general scenarios," The journal of systems and software, no. 66, pp. 187 - 197, 2003.
10. Eelke Folmer and Jan Bosh, "Architecting for usability: A survey. ," Journal of Systems and Software, pp. 61 - 78, 2004.
11. Object Management Group. Business Process Model and Notation (BPMN)," Agosto 2011. [En línea]. Available: <http://www.omg.org/spec/BPMN/>.
12. Leite J.C.S.P., Rossi G., et al. Enhancing a Requirements Baseline with Scenarios. Proceedings of RE 97': International Symposium on Requeriments Engineering, IEEE. Enero 1997.
13. Leite J.C.S.P, Albuquerque Oliveira, A P. A Client Oriented Requirements Baseline. Proceedings of RE 95': Second IEEE International Symposium on Requeriments Engineering. Inglaterra, Marzo 1995.
14. Xpdl.org. (2016). Welcome to XPDL.org. [online]. Disponible en: <http://www.xpdl.org/index.html>.
15. Antonelli, R. (2004). Traceability en la elicitación y especificación de requerimientos. [online] Hdl.handle.net. Disponible en: <http://hdl.handle.net/10915/4061>.
16. Leite, J.C.S.P., "Eliciting Requirements Using a Natural Language Based Approach: The Case of the Meeting Scheduler Problem", March 1993.
17. Hadad, G., Kaplan, G., Maiorana, V., Balaguer, F., Oliveros, A., Leite, J.C.S.P., Rossi, G. Informe Técnico: "Léxico Extendido del Lenguaje y Escenarios del Sistema Nacional para la Obtención de Pasaportes". Proyecto de Investigación, Departamento de Investigación, Universidad de Belgrano, Buenos Aires, 1996.
18. Leonardi, C., Leite, J.C.S., Rossi, Gustavo. "Una estrategia de Modelado Conceptual de Objetos, basada en Modelos de requisitos en lenguaje natural". Tesis de Maestría Universidad Nacional de la Plata. <http://postgrado.info.unlp.edu.ar/Carrera/Magister/Ingenieria%20de%20Software/Tesis/Leonardi.pdf>.
19. OMG. Unified Modelling Language: Superstructure Version 2.0 (online), Julio 2005, Disponible en: <http://www.omg.org/>