

Document downloaded from:

<http://hdl.handle.net/10251/179786>

This paper must be cited as:

Ormeño Ayala, YI.; Panach Navarrete, JI. (2013). Mapping Study about Usability Requirements Elicitation. Springer. 672-687. https://doi.org/10.1007/978-3-642-38709-8_43



The final publication is available at

https://doi.org/10.1007/978-3-642-38709-8_43

Copyright Springer

Additional Information

Mapping Study about Usability Requirements Elicitation*

Yeshica Isela Ormeño¹, Jose Ignacio Panach²

¹Centro de Investigación en Métodos de Producción de Software ProS, Universitat Politècnica de València, Camino de Vera s/n, 46022 Valencia, Spain

yormeno@pros.upv.es

²Escola Tècnica Superior d'Enginyeria Departament d'Informàtica, Av. de la Universidad, s/n 46100 Burjassot, Valencia, Spain

joigpana@uv.es

Abstract. The HCI community has developed guidelines and recommendations for improving the usability system that are usually applied at the last stages of the software development process. On the other hand, the SE community has developed sound methods to elicit functional requirements in the early stages, but usability has been relegated to the last stages together with other non-functional requirements. Therefore, there are no methods of usability requirements elicitation to develop software within both communities. An example of this problem arises if we focus on the Model-Driven Development paradigm, where the methods and tools that are used to develop software do not support usability requirements elicitation. In order to study the existing publications that deal with usability requirements from the first steps of the software development process, this work presents a mapping study. Our aim is to compare usability requirements methods and to identify the strong points of each one.

Keywords: Usability, MDD, Non-functional Requirements, Mapping Study

1 Introduction

The goal of developing usable systems has been dealt with by the Human Computer Interaction (HCI) and Software Engineering (SE) fields. In both communities, usability is usually considered in the last stages of the software development process, when the interfaces have already been designed. Including usability characteristics at these last stages could affect the system architecture. To minimize this problem, usability should be included at the requirements elicitation stage [5], [20]. The SE community has broad experience in early requirements elicitation and there are sound methods. However, these methods are mainly focused on functional requirements and Non-Functional Requirements (NFR) have historically been forgotten at this early stage.

*This work has been developed with the support of MICINN (PROS-Req TIN2010-19130-C02-02), UV (UV-INV-PRECOMP12-80627), GVA (ORCA PROMETEO/2009/015), and co-financed with ERDF. We acknowledge the support of the ITEA2 Call 3 UsiXML (20080026) and funding by the MITYC (TSI-020400-2011-20).

According to many authors, fulfilling functional requirements is not enough to create a quality product [49]. Usability is a key factor in obtaining good acceptance rates.

In this study, we aim to identify the existing methods for capturing usability requirements. To do this, we perform a Mapping Study (MS) based on the works performed by Kitchenham [29]. A MS provides an objective procedure for identifying the nature and extent of the research that is available to answer a particular question. These studies are also useful to identify gaps in current research and to suggest areas for further investigation. Of all the software development methods, we focus on the Model-Driven Development (MDD). MDD aims to develop software by means of a conceptual model, which is the input for a model compiler that generates the system code implementation. The SE community has been working with this paradigm, and, nowadays, there are sound methods and tools (e.g. OO-Method [39], WebRatio [2], OOHDM [12]). However, to the authors' knowledge, none of these methods deal with usability. In general, existing MDD methods deal with usability when the models that represent the functional requirements have been defined and the code has been generated. At this stage, if the analyst needs to improve the system usability, the code must be modified manually. Moreover, some changes require the architecture to be re-worked [5], [20]. These are the reasons why more efforts should be made to include usability in MDD methods, and this MS aims to be a step forward this direction.

Our long term target is twofold: (1) to improve current practices of usability requirements elicitation; and (2) to enhance the existing MDD methods to support usability requirements elicitation. The MS can help us to identify the advantages and disadvantages of each existing capture method, as a previous step for our target. However, the MS is not exclusive to MDD; it can analyze in detail any software development method that includes usability requirements elicitation.

This study is structured as follows. Section 2 reviews related works about usability requirements elicitation. Section 3 describes the design process of the MS. Section 4 shows the results obtained from the study. Section 5 presents a discussion about the results. Section 6 presents our conclusions and future work.

2 Related Work

Usability has been studied in several mapping studies and systematic reviews. The MS provides a systematic and objective procedure for identifying all the information that is available to answer a particular research question, topic area, or phenomenon of interest [29]. This section summarizes the different studies on requirements elicitation techniques, NFRs, and development methods based on usability.

First, we focus on studying techniques for capturing requirements that deal with usability. In this area, Dieste [13] updates a Systematic Review (SR) where interview-based techniques seem to be the most effective capture techniques. Carrizo [7] presents a framework to support decision-making, where some capture techniques responds better to certain project features than other capture techniques. Second, we focus on NFRs, since usability is considered by many authors to be a NFR. In the state-of-the-art written by Chung [11], the reviewed works are classified into six are-

as: software variability, requirements analysis, requirements elicitation, requirements reusability, requirements traceability, and aspect-oriented development. Svensson [50] performs a SR to identify: elicitation requirements, metrics, dependencies, cost estimation, and prioritization as important areas for managing quality requirements. Mellado [34] carries out a SR about security requirements engineering in order to summarize evidence regarding security. The precision and reliability of the information are his main contribution. Mehwish [33] reports a SR to collect evidence of software maintainability prediction. The results suggest that there is little evidence for the effectiveness of these predictions. Third, we focus on studies that deal with methods to build usable systems. Folmer [20] performs a survey to explore the feasibility of a framework that can be applied to usability at the architectural level, taking into account design methods for usability design and evaluation tools. He concludes that there are no techniques for dealing with usability at the architectural level. In Fernandez's work [18], the objective of the MS is to summarize the current knowledge of methods in order to evaluate usability in web applications. The results show the need for usability evaluation methods that are specific to the web.

In summary, we state that most of the existing research publications related to usability are focused on: inclusion of usability features at the design stage; usability evaluation at early phases; methods to assess usability at the implementation stage; usability evaluation throughout the web development process; and techniques for usability specification during the software development process. However, we have not found mapping studies or SRs focused on usability requirements elicitation at early phases. We aim to study the existing literature concerning usability requirements elicitation in order to summarize current knowledge. This information will be used in a future work to design a framework for usability requirements elicitation using existing guidelines.

3 Mapping Study Design

The MS provides a wide overview of a research area to identify the quantity and type of research and results available within it. We considered the following elements: research questions, search strategy, selection criteria, quality assessment, data extraction strategy [28]. Next, we apply these elements to our MS.

Our **research question** is: "What are the proposals to elicit usability requirements throughout the software development process?". It includes methods, notations, guidelines, tools, and empirical validations which are related to the usability area. The main goal is divided into six subgoals since the general research question is very abstract and involves many concepts. Each subgoal has been formulated as a research subquestion. These are: *SQ 1.1 Methods to elicit usability requirements*. It aims to study whether or not the proposed methods (including NFR methods) can capture usability requirements at early stages; *SQ1.2 Methods to elicit interaction requirements*. It aims to study the existing methods to elicit interaction requirements related to usability. These methods are included because some authors improve usability by means of visual characteristics; *SQ 1.3 Usability guidelines to elicit usability requirements*. It aims to study the recommendations that help the analyst to identify

usability requirements; *SQ 1.4 Tools to support usability requirements elicitation*. It aims to study the tools or prototypes that support the methods to elicit usability requirements; *SQ1.5 Notations to elicit usability requirements*. It aims to identify the existing representations in which the usability requirements are depicted. The target is to identify which notations are the most frequently employed for capturing usability requirements; *SQ1.6 Empirical validation environment*. It aims to study whether the proposal to elicit requirements was validated in an academic context or in industry.

The **search strategy** is composed of:

Defining the search sources. These sources are based on digital libraries that include peer-reviewed literature, such as: IEEEExplore, ACM Digital Library, Springer Link, and Science Direct. Our main tool for searching in all these libraries was Sciverse Scopus, since it allows searching in all the mentioned digital libraries (among others). The sources explored were the proceedings of conferences, journals, books, and workshops. The search area is restricted to the computer science area. The search period is from 2000 to 2011.

Building and applying the search string. The search string is a set of terms to obtain the publications that answer the research question. Our search string is composed of two substrings: Usability Requirements and Software Engineering. With the first we collect publications related to how to elicit Usability Requirements, including software quality features and works related to requirements elicitation. The second substring is related to Software Engineering concepts based on requirements elicitation.

Search string = (*Usability Requirement*) AND (*Software Engineering*)

Usability Requirement = (*usability requirement OR user requirement OR usability elicitation OR interaction requirement OR non-functional OR usability guidelines*).

Software Engineering = (*MDD OR model-driven OR MDA OR notation OR tool OR interface OR engineering OR test*).

We have included the term “non-functional” into the “Usability Requirements” group since usability is frequently considered as a NFR.

The **selection criteria** contains inclusion criteria (IC): IC1) Does the work define how to extract usability requirements?; IC2) Is the proposal applied to an environment based on MDD conceptual models?; IC3) Does the work define how to represent the requirements of usability? and exclusion criteria (EC): EC1) Publications focused on guidelines, notations, and tools where usability has not been considered or has not been included; EC2) Publications that consider only functional requirements; EC3) Publications written in a language that is not English.

Next, we **select the publications** through a systematic process:

Reading the title and the abstract. A total of 150 publications are returned by the search string, which are divided into three groups (50 publications) to be independently evaluated by three reviewers in order to apply the inclusion and exclusion criteria. The publications whose inclusion is doubtful must be discussed by the three reviewers until they arrive at a consensus. The result of this selection is a total of 65 publications, which are based only on the title and abstract of the publications. This selection is called “potential publications”.

Reading the whole publication. At this time, the whole publication is read. The inclusion and exclusion criteria are applied again for each potential publication, which

are divided into three groups (one group per reviewer). The result of this selection is a total of 27 “initial selected publications”, which are considered to be relevant.

Searching in references. In several cases there may be some relevant publications prior to the year 2000, such as Nielsen’s work [36]. In order to avoid discarding these interesting older publications, we review all the publications referenced in publications from 2000 to 2011. If a publication was written before 2000 and it has not been referenced in the last 12 years, then that work is not relevant for the community, and it is therefore discarded from our study. The process to review the references of publications from initial selected publications obtains 5 publications. 2 publications support inclusion criteria and are added to initial selected publications. Finally, a total of 29 publications are our “selected publications”.

In order to assess the reliability of inclusion, we apply the statistical measure of Fleiss’ Kappa [19]. This statistic assesses the reliability of agreement between a fixed number of rates when classifying items. Its value ranges between 0 (poor agreement) and 1 (full agreement). We take a sample of 20 publications of the 65 potential publications, 10 of which are randomly selected and 10 of which are defined by the reviewers from the 29 selected publications. The Fleiss’ Kappa value is 0.63, which is considered to be a “Considerable level”.

Table 1. Likert-Scale Questionnaire

| Subjective Questions | 1=Yes | 0=Partially | -1=No |
|---|-------------------|-------------|------------------|
| 1. Is the method to elicit the usability requirements clear? | | | |
| 2. Are the guidelines to elicit requirements comprehensible? | | | |
| 3. Are the guidelines to elicit requirements useful in other contexts? | | | |
| 4. Are the publications tools downloadable? | | | |
| 5. Is there a clear case study or example illustrating the proposal? | | | |
| 6. Is the whole proposal empirically validated? | | | |
| 7. Are the results clearly explained? | | | |
| 8. Is the notation to elicit requirements easy to learn? | | | |
| Objective Questions | | | |
| 9. Has the publication been published in journal or conference proceedings? | | | |
| 1=Very important | 0=Important | | -1=Not important |
| 10. Has the publication been cited by other authors? | | | |
| 1= More than 4 | 0=Between 2 and 4 | | -1=Less than 2 |

In order to perform the **quality assessment**, we use the Likert-Scale to be filled out by three reviewers for each selected publication. Table 1 contains closed-questions that are classified into two groups: Subjective Questions and Objective Questions. For question N° 9, we consider conferences at CORE ranking [38]. The publication is “Very important” if the conference is CORE A or B or if it is a book section, “Important” if the conference is CORE C or if it is a Workshop, “Not important” when the conference is not any CORE. For journals, the Journal Citation Report (JCR) [23] classification is used. The publication is considered to be “Very important” when it appears in JCR, “Important” when it does not appear in JCR but is indexed in other lists, and “Not important” when it is not published in any known list. For question N°

10, we use the H factor, which identifies the number of citations that each publication receives from other authors. The Publish or Perish [1] tool was used. In order to identify the quality of each publication, the three reviewers filled out the quality questionnaire. The aggregation of all the reviewers is performed by means of an arithmetic mean. After calculating the arithmetic mean for each question, we add these values, providing a single number between -10 and 10 which is denominated Quality Score. We consider that the Quality Score publication is “Very good” if it is more than 3, “Good” if it is between -2 and 2.99, and “Bad” if it is less than -2 (See Fig. 2b).

The **data extraction strategy** consists of classifying the possible answers for each research subquestion. The classifications are defined to facilitate the answer for our research question. These are:

- SQ1 Methods to elicit usability requirements. a) Yes b) No
- SQ2 Methods to elicit interaction requirements. a) Yes b) No
- SQ3 Guidelines to elicit usability requirements. a) Existing b) New c) Not exist
- SQ4 Tools to support the usability requirements elicitation a) Interface design (assistant to design) b) Model development c) Not Exist
- SQ5 Notations to elicit usability requirements. a) UML b) Natural Language (workshop sessions, checklists, questionnaires, heuristics, brainstorming, or interviews) c) i* framework d) CTT (Concur Task Trees) [40] e) Formal. (logical operators or grammars) f) QOC (Question Option Criteria) [31] g) BPMN h) Not exist
- SQ6 Empirical validation environment. a) Industrial b) Academic c) Not Exist

4 Results

Summary sources from search studies. The selected publications used in our MS are published in different sources. Table 2 shows the 65 potential publications and the 29 selected publications, classified by conference, journal, book, workshop, and other sources. Table 3 shows publications presented in conferences only. They are classified by level of the conference according to the CORE list. Finally, Table 4 shows publications published in journals only. The classification is based on the JCR list.

Table 2. Publications by Source

| Source | Potential | Selected |
|--------------|-----------|-----------|
| Conference | 31 | 14 |
| Journal | 16 | 9 |
| Book | 4 | 3 |
| Workshop | 4 | 1 |
| Other | 10 | 2 |
| Total | 65 | 29 |

Table 3. Publications by Conferences

| CORE | Potential | Selected |
|--------------|-----------|-----------|
| A | 12 | 6 |
| B | 10 | 4 |
| C | 9 | 4 |
| Total | 31 | 14 |

Table 4. Publication by JCR

| JCR | Potential | Selected |
|--------------|-----------|----------|
| Yes | 10 | 8 |
| No | 6 | 1 |
| Total | 16 | 9 |

Selected publication analysis. Table 5 shows the results of the 29 selected publications according to the data extraction strategy. Note that the answer for research subquestion SQ5 is not exclusive, i.e. more than one choice can be the answer.

Table 5. Mapping of selected publication

| SQ1 | | SQ2 | | SQ3 | | | SQ4 | | | SQ5 | | | | | SQ6 | | | Quality Score | ID | | | |
|-----|---|-----|---|-----|---|---|-----|---|---|-----|---|---|---|---|-----|---|---|---------------|----|---|-------|------|
| A | B | A | B | A | B | C | A | B | C | A | B | C | D | E | F | G | H | | | A | B | C |
| | X | X | X | | | | | X | | X | | | | | | | | X | | | 5,00 | [10] |
| | X | X | X | | | | | X | X | | | | | | | | | X | | | 3,67 | [37] |
| | X | X | X | | | | | X | | | | | | | | X | X | | | | 7,00 | [27] |
| | X | X | X | | | | | X | | | | | | | | X | X | | | | 1,00 | [26] |
| | X | X | | | | X | X | | | | | | X | | | | | X | | | -1,00 | [6] |
| | X | X | X | | | | | X | | X | X | | | | | | | X | | | 1,33 | [21] |
| X | | X | | X | | | | X | X | | | | | | | | | X | | | 3,67 | [24] |
| | X | X | | X | | | | X | | X | | | | | | | | X | | | 1,00 | [4] |
| | X | X | X | | | | | X | X | X | | | | | | | | X | | | 0,00 | [51] |
| | X | X | | X | | | | X | X | X | | | | | | | | X | | | -0,33 | [17] |
| | X | X | | X | | | | X | X | X | | | | | | | | X | | | -0,67 | [35] |
| X | X | | | X | X | | | X | | X | | | X | | | | X | X | | | 3,00 | [45] |
| X | | X | X | | | | | X | X | X | X | | | | | | | X | | | 4,67 | [14] |
| | X | X | X | | | X | | | | | | | | | X | | | X | | | -0,33 | [32] |
| X | | X | | X | | | X | | | X | | | | | | | | X | | | -2,00 | [3] |
| | X | X | | X | | | | X | X | | | | | | | | | X | | | 0,33 | [42] |
| | X | X | X | | | | | X | | | | | | | | | | X | | | 0,33 | [44] |
| | X | X | X | | | | | X | X | | | | | | | | | X | | | 0,67 | [8] |
| | X | X | X | X | | | | | | | | | | | | X | X | | | | 4,00 | [22] |
| | X | X | X | | | | | X | | | | | | | | X | X | | | | -2,67 | [43] |
| | X | X | | X | | | | X | X | X | | | | | | | | X | | | 2,67 | [46] |
| | X | X | X | | | X | | | | | | | X | | | | | X | | | 5,00 | [47] |
| | X | X | | X | X | | | X | X | | | | | | | | | X | | | 2,67 | [15] |
| | X | X | | X | X | | | X | X | | | | | | | | | X | | | 0,33 | [16] |
| | X | X | | X | | | | X | X | | X | | | | | | | X | | | 4,00 | [38] |
| | X | X | X | | | | | X | | | | | | | | X | X | | | | 1,33 | [30] |
| X | | X | | X | X | | | | X | | | | X | | | | X | | | | 7,67 | [48] |
| X | | X | X | | | | | X | X | X | X | X | X | | | | | X | | | 6,67 | [9] |
| X | | X | X | | | | | X | X | | | | | | | | | X | | | 4,00 | [25] |

SQ1: A) Yes 24.14% B) No 75.86%; SQ2: A) Yes 17.24% B) No 82.76%; SQ3: A) Existing 31.03% B) New 24.14% C) Not Exist 44.83%; SQ4: A) Interface Design 17.24% B) Model Development 24.14% C) Not Exist 58.62%; SQ5: A) UML 41.38% B) Natural Language 27.59% C) i* 27.59% D) CTT 13.79% E) Formal 6.9% F) QOC 6.9% G) BPMN 3.45% J) Not Exist 17.24% SQ6: A) Industrial 10.34% B) Academic 58.62% C) Not Exist 31.03%.

Next, we summarize the most relevant outcomes for each research subquestion:

SQ1 Methods to elicit usability requirements. There are few methods that propose capturing usability requirements, and usually they are included within NFR methods. In general, the requirements elicitation process uses traditional techniques (e.g. interviews, questionnaires, checklists, workshops) to elicit NFR at the same time the sys-

tem functionality and architecture are defined [45], [14], [25]. The most common goals of the studied NFR methods is to elicit measurable NFRs such a way they can be evaluated [14], [24]. These methods can be customizable for a different context if some settings are applied to a specific context. Therefore, a holistic quality model that fits every context does not exist, and NFR methods only provide basic requirements management by means of extensions [14]. The major benefits are the enhancement of the communication between the stakeholders and an increase in the flexibility of their applications; although some methods [25] tend to use more resources than others. The results indicate a limited number of approaches that deal with usability requirements at early stages.

SQ2 Methods to elicit interaction requirements. Methods to specify interaction requirements are based on the construction of a model and the definition of structural patterns for different design solutions [38], [37], [6]. These models support the systematic analysis of interaction requirements that can be selected from artefacts like a library of interaction attributes [47], [45]. These methods improve usability by means of applying formal modelling to analyze interactive systems systematically [6]. However, further work is needed to deal with dynamic specifications that depend on system functionality.

SQ3 Guidelines to elicit usability requirements. The publications aim to overcome the obstacle of the usability inclusion in the methods to elicit usability requirements and the different interpretations of the guidelines by the stakeholders. The methods that use *existing* guidelines, for instance ISO 9241-11 or ISO 9126, provide guidelines to determine usability requirements according to the definition of usability. They are understandable and can be implemented in a specific context [8], [32], [14], [51]; however their application is not an easy task [25], [21], [47]. The guidelines related to functional usability features are more practical, but they need to specify the usability feature by means of design patterns in the architectural design [37]. On the other hand, the *new* guidelines show a variety of representations (e.g. catalogues, methodologies, styles) [10], [22], [30] that are used to elicit usability requirements in different situations. All these representations allow to reuse its knowledge, to add new knowledge, to combine organizational memory or to combine different requirement scenarios. Other representations are based on patterns, templates, or models [27], [26], [38]. These artefacts can be improved or adapted according to which usability requirements are being captured. Nowadays, the guidelines do not provide precise, practical support to address usability requirements elicitation at the early stages.

SQ4 Tools to support usability requirements elicitation. These publications present tools to support: frameworks [45], structured styles [21], scenarios [48], notations [32], and methods [47]. The interface design tools support the requirements specification and validation through task flows and scenarios. Their main goals are focused on relating design options with functional and non-functional requirements within the design process of interactive systems. In order to reach this goal, it is necessary to incorporate a mechanism of transformation, (for example, from task flow diagrams to formal representations [45], [48]) and to solve traceability problems. The tools that are *model-based* can resolve this inconvenience by means of a global integration approach among notations and tools. However this is not an easy task [4], since most

tools focus specifications on requirements models or requirements metamodels. In order to define a elicitation process, the use of templates that are obtained through interviews [15], [16] or the use of patterns that provide a concise description of the users (detailing every significant characteristic [21]) are common.

SQ5 Notations to elicit usability requirements. The different notations are used in different stages of the software development process, and more than one notation is usually applied to the development method [28], [51]. The user requirements specifications are usually presented to end-users in normal text, even though the analyst works with languages based on models (SysML, UML). These requirements are based on a series of interviews and studies with end-users [46], [25], [14]. Some proposals aim to integrate functional requirements and NFR in the same elicitation process. These works propose a metamodel that combines UML with PLUS [51], [35], [45]. Therefore, *UML* and *Natural Language* are the most widely used notations (41.38% and 27.59%). In *Formal* notation, the specification is structured using hierarchical interfaces components that describe all the actions and visible attributes of the system [6]. In general, the other studied notations are currently supported by patterns, scenarios, and formatted templates in order to visualize and implement usability requirements [6], [38], [48], [25]. These representations help analysts to elicit requirements, even though they are not always easily understood by the end-user.

SQ6 Empirical validation environment. We observe that case studies, experiments and illustrative examples that have been presented in *Industrial* or *Academic* environments do not have explicit metrics to evaluate the usability requirements elicitation. In general, existing validations are focused on quantitative [24], [35], [27] and qualitative usability requirements [25]. The users' usability evaluation is often based on test and usability scenarios [27]. All the studied publications share the same protocol for the empirical validation. First, the publication proposes a method, technique or model to elicit usability requirements. Second, the publication details the results of the validation. Third, there is a discussion where a qualitative analysis is performed in detail and some lessons learned are shown. [47], [27], [48], [15]. Studied publications are focused on evaluating a few usability features; however, the study of a reduced number of features is not enough to consider software as being usable. The patterns [6], [47], [37], [15], scenario management [48], [9], [21], checklists [14], work sessions [25], and templates [6] are the most common artefacts used to evaluate usability and other NFRs.

Graphics of mapping results. We present four graphics of the MS results. Two correspond to comparison between research subquestions and the others correspond to the potential and selected publications and to the Quality Score of the selected publications. The six research subquestions give us an overview of the usability requirements and how they are related. Apart from reinforcing our conclusions of this study, this information can highlight some gaps that should be researched further.

Fig. 1a shows comparisons between research subquestions SQ1, SQ2, SQ3, and SQ4. The most important outcomes are the following: there is not any new guideline to elicit usability requirements or interaction requirements; there is the same number of publications where the tool is a support for interface design and model develop-

ment; there are a large number of publications that do not address methods of usability requirements elicitation or methods of interaction requirements elicitation.

Fig. 1b shows comparisons between the research subquestions SQ4 and SQ5. The most important outcomes are the following: UML, Natural Language, and CTT are notations used by model development tools and by design interface tools; BPMN and QOC are notations that are not used by model development tools; i* and Formal are notations that are not used by interface design tools.

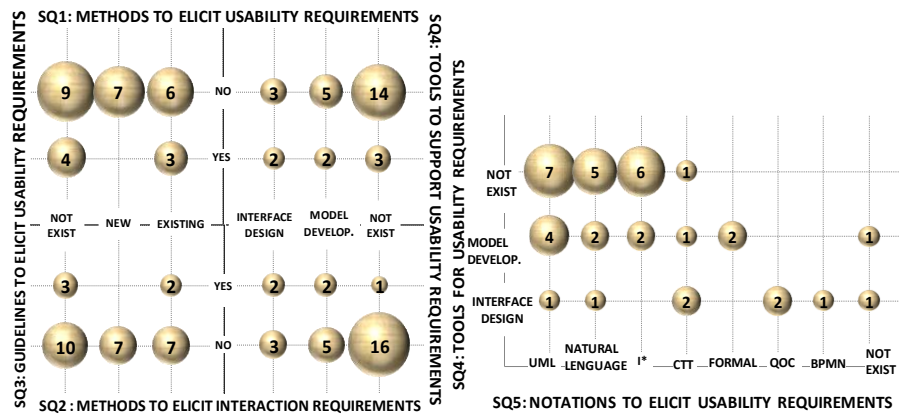


Fig.1a) Mapping results SQ1,SQ2,SQ3,SQ4

Fig.1b) Mapping results SQ4,SQ5

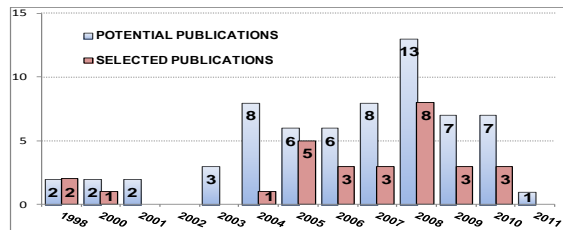


Fig. 2a) Frequency of publications by year

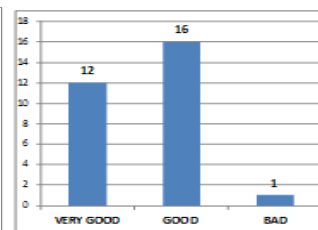


Fig. 2b) Publications by Quality Score

Fig. 2a shows the number of potential publications and selected publications classified by year. It can be observed that there are very few publications published each year. Of the 29 selected publications, 8 of them were published in 2008. This is the year that had the most publications for improving usability requirements elicitation. The year 1998 is included in the graphics because the two publications obtained from the referenced publications were published that year. None of the selected publications were published in 2001, 2002, 2003, and 2011.

Fig. 2b shows a frequency graphic that describes the quality assessment of the selected publications. This graphic is obtained from the Quality Score of selected publications, which can be “Very good”, “Good”, and “Bad”, according to our quality criteria. The graphic shows a high number of publications that are considered to be “Good” publications and “Very good” publications. Both results make up 95% of the total of the selected publications.

5 Discussion

In the selected publications, the usability requirements elicitation is usually performed at the analysis stage [46], [15], i.e., once all functional requirements have been captured. This late capture involves changes in system architecture since some usability requirements are related to functionality [5], [20]. In general, the methods used to elicit usability requirements deal with usability when the functional requirements have been previously captured by means of traditional techniques (e.g. interviews, questionnaires, focus groups, use cases) [35], [3].

The analysis of the results shows that there are very few publications that clearly address how to perform the capture process of usability requirements at early stages. Moreover, existing approaches do not propose a precise and unambiguous notation to represent these requirements, which makes difficult to apply them in real systems. There are some publications where usability requirements elicitation is performed at the design stage together with interaction requirements elicitation [25], [45], [24].

When the usability topic is dealt with at requirements elicitation, the ISO standards are used as guidelines to be applied in software development systems. For instance, the ISO 9241-11 is considered to be a basic reference for some practitioners, researchers, and designers [25], and for any kind of requirements the standard ISO 9126-1 is used [32]. The application of guidelines is necessary, but it is not enough; the main problem is the correct application and complete understanding by the end user. Guidelines are only built up in a general way, but they are not a total support for usability system development. There are some proposals that aim to help the requirements engineers to address usability requirements from the early stages by means of GUIDE rules [22] and a catalogue based on the *i** framework [10]. Both techniques are context-specific, even though GUIDE uses a case-based repository for taking decisions and *i** framework collects a large amount of knowledge to achieve usability goals.

Another aspect that is observed in selected publications is the use of artefacts, such as: patterns, scenarios, and templates, which are frequently used as support for methods to elicit usability requirements and interaction requirements [6], [48], [16]. The methods proposed in the selected publications are inflexible and require considerable effort to be applied in contexts that are different from the contexts where they have been defined [22]. The guidelines, notations, and artefacts used in these methods are closer to elicit interaction characteristics rather than usability characteristics. In general, guidelines for usability requirements elicitation are defined in a very generic way for different abstraction levels [8].

The tools to represent usability requirements which are based on a conceptual model have great possibilities of being useful for building extensions to other models (e.g. finite state machine) [45] or for being used in different contexts with other usability requirements. For large project, these tools are too limited, since the identification of requirements and modularization of the system need more special processes, methods and techniques. Moreover, once these requirements have been structured and gathered in a tool, they could be reused in later projects. Only few approaches include tools to support existing eliciting methods. Most approaches must be applied manual-

ly or they require a tool that is not provided by the authors [17], [42], [38]. This makes difficult the adoption of those approaches in industrial environments. The necessity of a tool is more urgent in those proposals that use several notations and combine the use of different artifacts (e.g. templates, questionnaires, workshops) [30], [14], [47]. Working with all these items manually is a huge effort for the analyst.

Validation methods are another crucial aspect for the evaluation of a proposal. The selected publications present case studies, experiments, and examples that do not show whether or not the inclusion of usability requirements produces a positive impact on the final product. In addition, only a small percentage of proposals have been applied in an industrial context [24].

Many works propose eliciting usability requirements with a graphical notation [10], [9], [6]. This enhances the abstraction for the requirements engineer but sometimes can difficult the end-user participation, who usually cannot understand those notations. Other proposals elicit usability requirements textually [25], [8], [48] facilitating the end-user participation. However, these proposals cannot be used for a development method based on models, since models do not exist.

If we focus our analysis on approaches to capture usability requirements in MDD environments, we notice that there are few proposals [38], [17], [46], [4]. Moreover, usability requirements are not usually considered as a main topic in those proposals. Usability requirements are combined with other NFR or with functional requirements, which makes difficult to focus the elicitation process on usability issues. Moreover, transformations among models are not discussed in those publications even though this is a basic pillar in the MDD paradigm (where transformations can be automated or semi-automated). Another problem of the existing proposals within the MDD paradigm is that there are not evaluations or tools to demonstrate that they can work in real systems. Existing approaches are just theoretical proposals that have not been implemented yet.

Note that our mapping study has some limitations. The first one is that we cannot ensure that all existing publication related to usability requirements have been considered. We have focused our research on Scopus, which is a tool that looks for publications in several digital libraries, such as IEEEExplore, ACM Digital Library, Springer Link, and Science Direct (among others). In order to minimize the loss of some important publications, we have analyzed references from publications retrieved by Scopus. However, publications that have not been published in those libraries or publications that have not been referenced are out of our search. Second, some found publications were not accessible (our university had no license to read them). This happened with 6 publications from 65. If we compare inaccessible publications with the total amount of publications, we notice that the percentage of unread publications is a minimum portion 9.23%. Throughout the whole mapping study we have been guided by an expert at mapping studies and systematic reviews. This expert helped us in the application of the protocol and recommended us some tools. For example, the use of Refworks [41] to eliminate duplicities in our search of publications, since the search string can find the same publication more than once.

6 Conclusions and Future Works

This MS combines usability aspects from both the Software Engineering (SE) community and the Human-Computer Interaction (HCI) community. We have explored the development methods that consider usability as a requirement from the SE community. We have studied the guidelines and heuristics from the HCI community that are used to develop usable applications. The MS aims to review existing studies related to usability requirements in both communities. Our main target is specially focused on proposals to elicit usability requirements from the early stages of the software development process.

The MS has been performed according to Kitchenham's methodology, focusing on the last 12 years. A total of 29 publications were selected from an initial set of 150 publications returned by the search string. The quality assessments of the publications were developed in order to contrast the significance of the selected publications, where 97% is composed of good publications and very good publications.

Using the results of the MS, we can conclude that there is a clear research line in the field of usability requirements in MDD environments. Usually, MDD methods have historically been focused on modelling behaviour and persistency, but relegating interaction (and particularly usability) to a manual implementation. This manual implementation clearly contradicts the MDD paradigm, which advocates that the analyst must work with holistic conceptual models, where every feature of the system (including usability features) could be represented. We plan to develop a framework to elicit usability requirements in such a way that it could be used in any MDD method. The main benefit of embedding usability requirements in a MDD method is that the next steps of the software development process can be derived from the requirements elicitation step. We plan to develop transformation rules from the usability requirements to generate analysis and design models. Furthermore, the MS can also be used as a starting point for future systematic reviews based on usability requirements elicitation.

References

1. Publish or Perish, <http://www.harzing.com>
2. Acerbis, R., Bongio, A., Brambilla, M., Butti, S.: WebRatio 5: An Eclipse-Based CASE Tool for Engineering Web Applications. In 7th International Conference on Web Engineering, Springer-Verlag, Berlin, Heidelberg, 501-505. (2007)
3. Akoumianakis, D., Katsis, A., Vidakis, N.: Non-functional User Interface Requirements Notation (NfRn) for Modeling the Global Execution Context of Tasks. In 5th International Conference on Task Models and Diagrams for Users Interface Design, Springer-Verlag, Hasselt, Belgium, 259-274. (2007)
4. Ameller, D., Franch, X., Cabot, J.: Dealing with Non-Functional Requirements in Model-Driven Development. In 18th IEEE International Conference on Requirements Engineering (RE). Sydney, NSW, 189-198. (2010)
5. Bass, L., John, B.: Linking Usability to Software Architecture Patterns through General Scenarios. Journal of Systems and Software, Vol. 66, No. 3, 187-197. (2003)

6. Campos, J., Harrison, M., Graham, T., Palanque, P.: Systematic Analysis of Control Panel Interfaces Using Formal Tools Interactive Systems. Design, Specification, and Verification. Springer-Verlag, Vol. 5136, Berlin, Heidelberg, 72-85. (2008)
7. Carrizo, D., Dieste, O., Juristo, N.: Study of Elicitation Techniques Adequacy. In 11th Workshop on Requirements Engineering. Spain, Barcelona, 104-114. (2008)
8. Cronholm, S. and Bruno, V.: Do you Need General Principles or Concrete Heuristics?: A Model for Categorizing Usability Criteria. In 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat, ACM, Cairns, Australia. (2008)
9. Cysneiros, L. M., Leite, J.C.S.P.: Nonfunctional Requirements: from Elicitation to Conceptual Models. IEEE Trans. on Softw. Eng., Vol. 30, No. 5, 328-350. (2004)
10. Cysneiros, L.M., Werneck, V. M. Kushniruk, A.: Reusable Knowledge for Satisficing Usability Requirements. In 13th IEEE International Conference on Requirement Engineering, IEEE Computer Society, Washington, DC, USA, 463-464. (2005)
11. Chung, L. Leite, J.C.S.P.: On Non-functional Requirements in Software Engineering. LNCS, Springer, Vol. 5600, Berlin, Heidelberg, 363-379. (2009)
12. Daniel, S., Rita de Almeida, P., Isbela, M.: OOHDM-Web: An Environment for Implementation of Hypermedia Applications in the WWW. SIGWEB NewsL., Vol. 8, No. 2, 18-34. (1999)
13. Dieste, O., Lopez, M., Ramos, F.: Updating a Systematic Review about Selection of Software Requirements Elicitation Techniques In 11th Workshop in Requirements Engineering, Barcelona, Spain. (2008)
14. Doerr, J., Kerkow, D., Koenig, T., Olsson, T., Suzuki, T.: Non-functional Requirements in Industry - Three Case Studies Adopting an Experience-based NFR Method. In 13th IEEE International Conference on Requirements Engineering, Washington, DC, USA, 373-384. (2005)
15. Escalona, M.J., Arag, G.: NDT. A Model-Driven Approach for Web Requirements. IEEE Trans. Softw. Eng., Vol. 34, No. 3, 377-390. (2008)
16. Escalona, M.J., Koch, N., Filipe, J., Cordeiro, J., Pedrosa, V.: Metamodeling the Requirements of Web Systems Web Information Systems and Technologies. Springer-Verlag, Berlin, Heidelberg, Vol 1, 267-280. (2007)
17. Fatwanto, A. and Boughton, C.: Analysis, Specification and Modeling of Non-Functional Requirements for Translative Model-Driven Development. In International Conference on Computational Intelligence and Security, Washington, DC, USA, 405-410. (2008)
18. Fernandez, A., Insfran, E., Abrahão, S.: Usability Evaluation Methods for the Web: A Systematic Mapping Study. Information and Software Technology, Vol. 53, No. 8, 789-817. (2011)
19. Fleiss, J.L.: Statistical Methods for Rates and Proportions. John Wiley & Sons, New York, Ed. (1981)
20. Folmer, E., Bosch, J.: Architecting for usability: A Survey, Journal of Systems and Software, Vol. 70, No. 1, 61-78. (2004)
21. Grosse-Wentrup, D., Stier, A., Hoelscher, U., Dössel, O., Schlegel, W.C.: Supporting Tool for Usability Specifications. In World Congress on Medical Physic and Biomedical Engineering. Springer-Verlag, Munich, Germany, 845-847. (2009)
22. Henninger, S.: A Methodology and Tools for Applying Context-specific Usability Guidelines to Interface Design. Journal Interacting with Computers, Vol. 12, No. 3, 225-243. (2000)
23. Journal Citation Reports, <http://ip-science.thomsonreuters.com>

24. Jokela, T., Koivumaa, J., Pirkola, J., Salminen, P., Kantola, N.: Methods for Quantitative Usability Requirements: A Case Study on the Development of the User Interface of a Mobile Phone. *Personal Ubiquitous Comput.*, Vol. 10, No. 6, 345-355. (2006)
25. Jokela, T., Seffah, A., Gulliksen, J., Desmarais, M.C.: 8 Guiding Designers to the World of Usability: Determining Usability Requirements Through Teamwork. Springer Netherlands, Vol. 8, 127-145. (2005)
26. Juristo, N.: Impact of Usability on Software Requirements and Design. Springer-Verlag, Vol. 55-77. (2009)
27. Juristo, N., Moreno, A. M., Sánchez, M. I.: Guidelines for Eliciting Usability Functionalities, *IEEE Trans. Softw. Eng.*, Vol 33, No. 11, 744-758. (2007)
28. Kitchenham, B.: Procedures for Performing Systematic Reviews, Technical Report TR/SE-0401. (2004)
29. Kitchenham, B. A., Charters, S.: Guidelines for performing Systematic Literature Reviews in Software Engineering, EBSE Technica Report. (2007)
30. Lauesen, S. Younessi, H.: Six styles for usability requirements. In REFSQ'98 (1998)
31. MacLean, A., Young, R. M., Bellotti, V. M. E., Moran, T.P.: Questions, Options, and Criteria: Elements of Design Space Analysis. *Human-Computer Interaction*, Vol. 6, No. 3, 201-250. (1996)
32. Martinie, C., Palanque, P., Winckler, M., Conversy, S., DREAMER: A Design Rationale Environment for Argumentation, Modeling and Engineering Requirements. In 28th International Conference on Design of Communication. São Paulo, Brazil. (2010)
33. Mehwish, R., Emilia, M., Ewan, T.: A Systematic Review of Software Maintainability Prediction and Metrics. *IEEE Computer Society*, Washington, DC, USA, 367-377. (2009)
34. Mellado, D., Blanco, C., Sánchez, L. E., Fernandez, E.: A Systematic Review of Security Requirements Engineering. *Comput. Stand. Interfaces*, Vol. 32, No. 4, 153-165. (2010)
35. Nguyen, Q.L., Non-Functional Requirements Analysis Modeling for Software Product Lines. In ICSE Workshop on Modeling in Software Engineering, Washington, DC, USA, 56-61. (2009)
36. Nielsen, J.: Usability Engineering. Morgan Kaufmann. (1993)
37. Panach, J.I., España, S., Moreno, A. and Pastor, Ó.: Dealing with Usability in Model Transformation Technologies. In ER 2008, Springer LNCS Barcelona, 498-511. (2008)
38. Panach, J.I., España, S., Pederiva, I., Pastor, O.: Capturing Interaction Requirements in a Model Transformation Technology Based on MDA. *Journal of Universal Computer Science (JUCS)*, Vol. 14, No. 9, 1480-1495. (2007)
39. Pastor, O., Molina, J.: Model-Driven Architecture in Practice. Springer, Ed. (2007)
40. Paterno, F.: Model-based Tools for Pervasive Usability. In *Interacting with Computers* 17 (3), Elsevier, 291-315. (2004)
41. Refworks, <http://www.refworks.com/>
42. Röder, H.: Using Interaction Requirements to Operationalize Usability. In ACM Symposium on Applied Computing, Sierre, Switzerland. (2010)
43. Sajedi, A., Mahdavi, M., Pourshirmohammadi, A., Nejad, M. M.: Fundamental Usability Guidelines for User Interface Design. In International Conference on Computational Sciences and Its Applications ICCSA. Washington, DC, USA, 106-113. (2008)
44. Shehata, M., Eberlein, A., Fapojuwo, A., O.: A Taxonomy for Identifying Requirement Interactions in Software Systems. *Comput. Netw.*, Vol. 51, No. 2, 398-425. (2007)
45. Sindhgatta, R. and Srinivas, T. Functional and Non-Functional Requirements Specification for Enterprise Applications. Springer-Verlag, Vol. 3547, Berlin, Heidelberg, 189-201. (2005)

46. Soares, M. S., Vrancken, J. L. M.: Model-Driven User Requirements Specification using SysML. *Journal of Software*, Vol. 3, No. 6, 57-68. (2008)
47. Sutcliffe, A.G., Kurniawan, S., Jae-Eun, S.: A Method and Advisor Tool for Multimedia User Interface Design. *Int. J. Hum.-Comput. Stud.*, Vol. 64, No. 4, 375-392. (2006)
48. Sutcliffe Alistair G., Michele, R.: Experience with SCRAM, a SCenario Requirements Analysis Method. In *3rd International Conference on Requirements Engineering: Putting Requirements Engineering to Practice*, IEEE Computer Society, 164-171. (1998)
49. Svensson, B., Gorschek, T., Regnell, B., Torkar, R., Shahrokni, A., Feldt, R.: Quality Requirements in Industrial Practice - An Extended Interview Study at Eleven Companies. *IEEE Trans. Softw. Eng.*, Vol. 38, No. 4, 923-935. (2012)
50. Svensson, R.B., Höst, M., Regnell, B.: Managing Quality Requirements: A Systematic Review. In *36th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA)*. 261-268. (2010)
51. Yi, L., Zhiyi, M., Weizhong, S.: Integrating Non-functional Requirement Modeling into Model Driven Development Method. In *17th Asia Pacific Software Engineering Conference*, Beijing, China. (2010)