Author version, published in: Journal of Software: Evolution and Process, 30 (11), pp. 1-22, 2018.

HCI in Practice: an Empirical Study with Software Process Capability Maturity Models Consultants in Brazil

Taisa Guidini Gonçalves, Káthia Marçal de Oliveira, Christophe Kolski
LAMIH CNRS UMR 8201, University of Valenciennes and Hainaut-Cambrésis, Polytechnic University of Hauts-de-France
Le Mont Houy cedex 9 – 59313, Valenciennes, France
taisaguidini@gmail.com, kathia.oliveira@univ-valenciennes.fr, christophe.kolski@univ-valenciennes.fr

Abstract. Human-Computer Interaction (HCI) and Software Engineering (SE) are undoubtedly important domains for the development of interactive systems. The quality of an interactive system is usually considered dependent on the user interface design and evaluation, which implies the use of HCI and SE approaches on the adequate software process. With the wide use of software process capability maturity (SPCM) models in the industry and their role of proposing best practices for software development and maintenance, some questions come up: to what extent are HCI approaches known and used to develop interactive systems? What is their level of use compared to SE approaches? To answer these questions, we conducted an empirical study in the Brazilian industry. Thirty-six official consultants for the two SPCM models used in Brazil (CMMI-DEV and MR-MPS-SW) provided us with their perception of knowledge and use concerning a set of predefined categories of approaches for HCI and SE that support the engineering practices of those SPCM models. By using a paired t-test, we concluded that consultants of those models do not know and do not use HCI approaches as well as they know and use SE approaches. Moreover, they know little about HCI approaches, which may justify their poor use.

Keywords: Human-Computer Interaction; Software process capability maturity models; Software Engineering; Interactive system; Empirical study.

1. Introduction

With the dissemination of the use of portable devices (smartphones, tablets, etc.) and internet, we can say that we are living in the era of interactive systems supporting all our everyday activities. The software industry has been in charge of producing systems that serve a wide population in the most diverse activities of the society. As a consequence, Human-Computer Interaction (HCI) issues have gained great attention in software systems development. Indeed, several studies showed the importance of usability for the quality of use and adoption of software systems [1] [2] [3] [4] [5]. In 2006, Theofanos's report [6] presented real studies of software project failure due to usability problems. Previous studies also proved that a large percentage of the code of the interactive system is dedicated to human-machine communication [7]. This scenario may lead us to conclude that HCI engineering practices are really important for the development of interactive systems.

In parallel to this scenario, software engineering practices have been applied in the industry, in part thanks to the use of software process capability maturity (SPCM) models [8] [9] [10] [11]; both in classical development and in new agile software development [12] [13] [14] [15] [16]. These models are a collection of best practices in software engineering that help organizations to improve their process. They define "what" needs to be implemented but not "how" to do it. A large number of official appraisals using these models indicate that the implementation of SPCM models is a good way to apply software engineering practices in the industry. For instance, more than 10,000 official appraisals using CMMI (Capability Maturity Model Integration), the international model most widely known in the world, are reported from over 80 countries. Other national SPI models (such as the MR-MPS-SW Brazilian model [17] and the MoProSoft Mexican model [18]) are also being largely used in industry; e.g., more than 600 official appraisals on the national model (MR-MPS-SW) created in 2005.

We argued that HCI engineering is related to the software engineering particularly applied to the interactive system projects. Therefore, the use of SPCM models should also help in the application of HCI engineering in industry. Jokela and Lalli [13] said, for example, that several process areas from CMMI have a direct relationship with usability practices, consequently HCI engineering. However, to what extent are HCI approaches known and used to develop interactive systems? What is their level of use compared to SE approaches?

In this paper, we present a study aiming to answer those questions by surveying official Brazilian consultants of the two most used SPCM models in Brazil (Capability Maturity Model Integration for Development - CMMI-DEV [19] and the MPS for Software reference model - MR-MPS-SW [17]). We chose to work with official consultants from these models because they are experts who support many enterprises in their software development strategies. Moreover, they are officially registered in the database of the official institutions that control the appraisals in those models[1]. The goal of this survey was, therefore, to investigate to what extent the HCI and SE approaches are known and used by consultants when applying SPCM in the development/maintenance of a software system. Investigating both areas allows us not only to identify the level of knowledge and use of each one but also to have a reference for analysis and discussion. We considered as an **approach**, any method, technique, pattern or standard from both domains (HCI and SE). To perform the survey, we looked for the consultants' **perception of knowledge** and **use**, meaning the extent to which the consultants recognize they know and they use the specified approaches while applying SPCM models in software project development.

This paper is organized as follows: Section 2 presents the background of this work: briefly introducing some HCI and SE approaches, the SPCM models which are the focus of our study, and some related studies about HCI in practice. Section 3 describes the planning and execution of our survey. Section 4 presents a discussion of the results and threats to validity. Finally, section 5 closes the paper with some final remarks and ongoing works.

2. Background

2.1. HCI and SE approaches: a brief overview

Over the past few years, HCI engineering and SE have proposed models, methods, techniques, and standards (from hereafter in this paper referred as **approaches**) to support the analysis, design, implementation and evaluation of software systems.

On one hand, the Software Engineering community has used concretely in the industry the models, methods, techniques, and standards that have been developed for a long time. Software development life cycles or process models (e.g. the waterfall model [20], the V-model [21]) and ISO standards that provide processes for the development of a system or product have also been defined (e.g. ISO/IEC 12207 [22], IS0 25000 [23]). We can also quote, for example, methods for analysis and design (e.g. UML diagrams [24], design patterns [25], software architecture models [26]), techniques for verification (unit test, acceptance test, formal usability inspection) and validation [27].

[1] For CMMI-DEV, CMMI Institute - http://cmmiinstitute.com/resources/ and for MR-MPS-SW, SOFTEX - http://www.softex.br/mpsbr

On the other hand, Human-Computer Interaction Engineering has also proposed approaches to support interactive system development. For instance, different software development life cycles such as the star model [28], Nielsen's usability engineering life cycle [29] and Mayhew's usability engineering life cycle [30] have been proposed. In addition, the classical life cycles of software engineering (such as the V-model [21], spiral model [31], etc.) have also been enhanced in terms of HCI (see for example [32] [33]). Moreover, the HCI literature offers approaches to support the implementation of all phases of these development life cycles, such as: i) requirement analysis phase (e.g. task analysis methods [34]; techniques to identify user needs and requirements [35]; prototyping techniques [36]); ii) design and implementation phases (e.g. architecture patterns [37] [38]; design patterns [39] [40]); iii) evaluation phase (e.g. techniques for validation and verification: usability tests [41]; standardized usability questionnaires [42] [43]; heuristic evaluation [44]; cognitive walkthrough [45]; automated evaluation [46].

We note that although both communities have worked in parallel, several approaches were used in both communities. Prototyping, techniques to identify user needs and requirements, some verification and validation techniques (such as the use of test with end users) are shared by both communities as good practices for software development.

2.2. Software Process Capability Maturity Models

Software process capability maturity (SPCM) models can be defined as a collection of software engineering best practices, organized in process areas or processes, which help companies to improve their software processes. Several SPCM models are proposed. We focus our study on the international model CMMI-DEV [19] and on the Brazilian model MR-MPS-SW [17]. In fact, although each SPCM model has its own structure, they can be used jointly by organizations wishing to improve their processes thanks to the document published by SOFTEX [47] that presents the equivalences between them.

CMMI-DEV [19] is composed of a set of components; the main ones used for our study are: (i) *process area*, a cluster of related practices in an area that, when implemented collectively, satisfies a set of goals considered important for making a significant improvement in that area; (ii) *specific goal* (SG) that describes characteristics that must be present to satisfy the process area; and, (iii) *specific practice* (SP) that describes a set of specific best practices in a single process area. Generic goals and generic practices are also defined to be applied in all process areas.

The CMMI-DEV version 1.3 [19] has 22 process areas organized into four categories: engineering, support, project and process management. The engineering category is directly related to the development/maintenance and evaluation of software being composed of the following process areas: Requirements Development (RD), Technical Solution (TS), Product Integration (PI), Validation (VAL) and Verification (VER). These process areas encompass 40 specific practices (SP) grouped into 14 specific goals (SG). The five process areas and their specific practices used in this study are presented in the **Appendix**. Figure 1 presents, as an example, the specific goals and practices for the Requirements Development process area [19]. Moreover, CMMI-DEV uses maturity and capability levels to describe an evolutionary path for an organization that wants to improve its processes. Maturity levels are the most applied in the industry [16] [48] organizing the process areas in a staged representation where each level is composed of several process areas. CMMI-DEV proposes five maturity levels: 1-Initial, 2-Managed, 3-Defined, 4-Quantitatively Managed and 5-Optimizing. Engineering process areas are in level 3 (Defined).

SG 1 Develop Customer Requirements

- SP 1.1 Elicit Needs
- SP 1.2 Transform Stakeholder Needs into Customer Requirements

SG 2 Develop Product Requirements

- SP 2.1 Establish Product and Product Component Requirements
- SP 2.2 Allocate Product Component Requirements
- SP 2.3 Identify Interface Requirements

SG 3 Analyze and Validate Requirements

- SP 3.1 Establish Operational Concepts and Scenarios
- SP 3.2 Establish a Definition of Required Functionality and Quality Attributes
- SP 3.3 Analyze Requirements
- SP 3.4 Analyze Requirements to Achieve Balance
- SP 3.5 Validate Requirements

Figure 1. Requirements Development process area [19]

MR-MPS-SW is composed of 19 software processes organized in maturity levels [17]. A process in MR-MPS-SW model is composed of a purpose (the main objective to be expected with the execution of the process) and expected results (the results from the execution of the specific process). A process capability is represented by a set of process attributes. MR-MPS-SW presents seven maturity levels ranging from level G (initial) to A (highest). The processes related to development/maintenance of the software are organized in level D. To obtain a maturity level, the enterprise must meet all the process attributes required for all processes related to that maturity level.

To assure the compatibility of the MR-MPS-SW with CMMI-DEV a technical mapping [47] is established as a guideline for enterprises. This technical mapping associates each expected result of MR-MPS-SW with each specific practice of CMMI-DEV, also defining if it is totally equivalent, jointly equivalent, not equivalent and why or inexistent. Figure 2 presents the equivalence between the components of the models represented by the same colors as follows: a process in MR-MPS-SW is equivalent to a process area in CMMI-DEV; the purpose of the process is equivalent to the set of specific goals of the corresponding process area; and, an expected result in MR-MPS-SW is equivalent to a specific practice in CMMI-DEV. Moreover, an equivalence also exists between the maturity levels of each model. For instance CMMI-DEV level 3 is equivalent to MR-MPS-SW level C.

2.3. Related Work

Several studies have investigated the perception of knowledge about HCI and/or the use of HCI approaches in practice. We found twelve studies in literature using an ad hoc technique [49] [50] [51] [52] [53] [54] [55] [56] [57] [58] [59] [60]. These studies were not developed in the context of software development with SPCM models implementations. They show the practice of HCI, usability and User Experience (UX) in the industry for different countries (Korea [49], Switzerland [51], Germany [56], Italy [57], Nigeria [58], Malaysia [52] [53] [54] [55], Brazil [59] [60], and the United States and European countries [50]), showing the difficulties and benefits of HCI.

Venturini et al. [50] used the phases of a software process to evaluate HCI in practice but did not specify which HCI approaches could be used in each phase. Vukelja et al. [51] asked the participants (134) about the use of software engineering methods in the software development. Regarding the test activity, they said that the system modules' and the systems are tested. The modules are tested in 76.2% of cases, and the systems in 98.1% of cases. Usability tests are only conducted in 37.9% of the cases. The documentation for the end user is written in 34.2% of cases in parallel with the development and at the end; unfortunately, in 65.8% of cases, it is only done at the end.

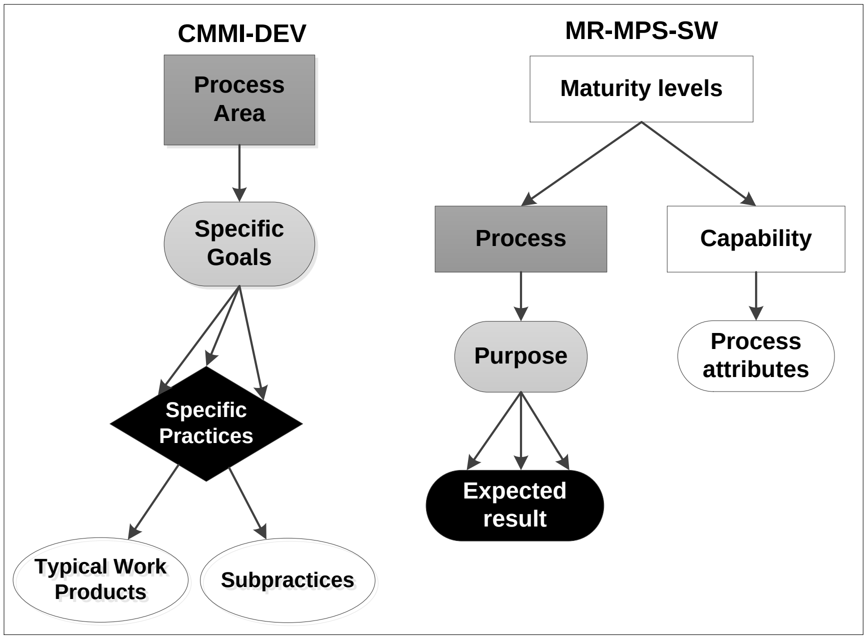


Figure 2. Equivalence between the models

Four studies [50] [56] [59] [60] did not discuss the perception of knowledge on HCI, and four others [52] [53] [55] [57] did not discuss the use of HCI approaches.

Different results were obtained in the studies about the perception of knowledge of HCI. Some studies showed that most of the participants know about HCI [52] and usability [49] [52] [53] [54] [55]. Others show that the majority has a low level of knowledge on HCI [51] or does not know well what usability is [57]. Ogunyemi et al. [58] showed in their study that, although the majority of the organizations claim to be aware of HCI, the responses of the participants (by interview) about the HCI methods applied in their companies do not support this claim. The authors concluded that the perception of knowledge about HCI in those companies is inadequate.

The two studies in Brazil [59] [60] aimed to investigate specifically the use of usability/UX practices. The first one [60] surveyed UX professionals and identified that their main activities performed during their work are: Low Fidelity Prototypes (74.8%), High Fidelity Prototypes (67.3%), User Interviews (61.2%), Heuristic Evaluation (60.4%), and Usability Test (55.1%). We can see that the most frequent activities were related to the development of prototypes. The second study [59] extended the first one focusing on small enterprises that work with interactive system development to investigate activities of Usability/UX in the evaluation of the systems. They found that usability tests and heuristic evaluation were the most widely used evaluation methods.

Analyzing the studies on the use of HCI approaches, we identified that: (i) Task analysis was classified as the most used method in [49] and [54]; (ii) Lo-fi prototypes and Hi-fi prototypes were placed between the three first places of most used for [50] [59] and [60]; (iii) Usability tests was placed in fifth position for [50] [59] and [60], and in ninth and tenth position for [54] and [49], respectively. That means, we found several similarities if we considered the findings of the approaches more used in practice.

3. Investigating the Practice of HCI issues in Brazil

As defined in the introduction section our main assumption is that HCI approaches are not sufficiently known and not sufficiently used as SE approaches in practice when dealing with the development/maintenance of a software system. To investigate this, we planned a survey following the procedures for empirical research as defined by [61]. In this section, we present the planning and the execution of this survey.

3.1. Survey Planning

3.1.1. Definition of Goal and Hypotheses

We formalized the goal of this study according to [62] as follows:

Analyze Human-Computer Interaction **(**HCI) and Software engineering (SE) approaches
for the purpose of identification
with respect to the perception of knowledge and use of methods, techniques, standards, and patterns
from the point of view of SPCM model consultants
in the context of SPCM model implementations.

In other words, the goal is to identify the level of perception of knowledge and use of methods, techniques, standards, and patterns of SE and HCI related to SPCM model implementations. We mean by SPCM implementations, the use of SPCM models in the development/maintenance of software projects in industry. Since we were interested in activities dealing with software development/maintenance, we focus our studies on the practices related to engineering process areas (CMMI-DEV level 3) and their equivalent in MR-MPS-SW (placed in level D).

Enterprises that decide to use SPCM models usually hire a consultant to help them to introduce the practices and to train the staff. This is done because, in general, those enterprises are interested in being officially assessed by the institutes that manage the models (CMMI Institute[2] for CMMI-DEV and SOFTEX[3] for MR-MPS-SW). The consultants are responsible for introducing the approaches to be used by the software developers of the enterprises in the development/maintenance of the software products. Therefore, we considered consultants to be a good source to investigate what has probably been used in industry when applying SPCM models. By focusing on the consultants, we can also control the scope of participants for the survey.

Moreover, the report from SOFTEX [63] shows that those enterprises develop systems for different application domains, such as banking automation, distance education, e-business, electronic commerce, human resource management, school administration, web pages. That means they develop systems that deal directly with final end-users requiring good user interaction to be really used.

To address our goal, we needed to compare the perception of knowledge and use of these approaches with SE approaches. Therefore, we also identified and evaluated SE approaches that are usually applied with SPCM models. From this, we set two hypotheses to be investigated:

- **H1.** SPCM models consultants do not know HCI approaches as they know Software engineering (SE) approaches when applying the same specific practice of the CMMI-DEV engineering process area or its correspondent result of the MR-MPS-SW.

[2] http://partners.cmmiinstitute.com/find-partner-organization/
[3] http://www.softex.br/mpsbr/instituicoes-autorizadas/

- **H2.** SPCM models consultants do not use HCI approaches as they use Software engineering (SE) approaches when applying the same specific practices of the CMMI-DEV engineering process area or its correspondent result of the MR-MPS-SW.

Hereafter when we mention "consultants know or do not know", this means their perception of what they know or do not know, i.e., the consultants recognize that they know and they use. Our hypotheses were formalized as described below. For the first hypothesis (H1) we have:

- **Null hypothesis ($H1_0$):** SPCM models consultants know HCI approaches (KHCI) as they know SE approaches (KSE) when applying the same specific practice of the CMMI-DEV engineering process area or its correspondent result of the MR-MPS-SW[4].
  $H1_0$: $KHCI_i - KSE_i = 0$;
- **Alternative hypothesis ($H1_A$):** SPCM models consultants know HCI approaches less than SE approaches when applying the same specific practice of the CMMI-DEV engineering process area or its correspondent result of the MR-MPS-SW.
  $H1_A$: $KHCI_i - KSE_i < 0$;

Similarly, the second hypothesis (H2) related to the use of HCI approaches was formalized as follows:

- **Null hypothesis ($H2_0$):** SPCM models consultants use HCI approaches (UHCI) as they use SE approaches (USE) when applying the same specific practice of the CMMI-DEV engineering process area or its correspondent result of the MR-MPS-SW.
  $H2_0$: $UHCI_i - USE_i = 0$;
- **Alternative hypothesis ($H2_A$):** SPCM models consultants use HCI approaches less than SE approaches when applying the same specific practice of the CMMI-DEV engineering process area or its correspondent result of the MR-MPS-SW.
  $H2_A$: $UHCI_i - USE_i < 0$;

3.1.2. Instrument

Based on our goal and defined hypothesis, to prepare our survey we first need to identify the set of HCI and SE approaches that may support the consultants while using CMMI-DEV and MR-MPS-SW. With the large use of SPCM models in the industry, we considered that SE approaches (for instance, the use of Unified Modeling Language, design patterns, etc.) are well-known for supporting different activities in the software development. However, we could not have the same assumption for HCI, even more so considering the state of practice presented in section 2.2.

With this in mind, we needed to identify which HCI approaches could support each one of the practices of the two SPCM models, in order to prepare our survey to answer our study hypothesis. To that end, we performed a study to analyze CMMI-DEV engineering practices and HCI literature, and we interviewed twenty experts from HCI with considerable experience (19 years in average) in the academic and industrial fields. By evaluating specific practices of CMMI-DEV and examples of HCI approaches, 14 categories of approaches were defined (see Table 1). Each one of these categories supports one or more practices of CMMI-DEV. For instance, the second category of Table 1 (*Techniques to identify user and organizational requirements*) can be used to support different practices (SP) of the Requirement Development process area. Examples of approaches for each category were indicated without intending

[4] Where *i* represent each category of HCI and SE that supports each CMMI-DEV practice or result of MR-MPS-SW.

to be exhaustive, but rather to exemplify the kind of approaches in that category. A detailed description of this study can be found in [64].

Table 1. HCI and SE categories of approaches

#	Human-Computer Interaction (HCI)		Software Engineering (SE)	
	Category	**Examples of approaches**	**Category**	**Examples of approaches**
1	Techniques to identify user needs	Brainstorming; Interviews; Surveys/Questionnaires; Card Sorting; Focus Groups; Field Studies/Observation.	Techniques to identify needs	Brainstorming; Interviews; Questionnaires; Card Sorting; Focus Groups; Field Studies/Observation; Workshops; Protocol Analysis.
2	Techniques to identify user and organizational requirements	Scenario; User stories; Storyboards; Task Analysis; Persona; Context-of-use analysis; User Profile (Detailed); Requirements specification templates (e.g. VOLERE, IEEE, RESCUE).	Techniques to identify requirements	Scenario; User stories; Storyboards; Task Analysis; Use cases; Quality Function Deployment; FAST (Facilitated Application Specification Techniques): JAD, The Method
3	Task Modeling	HTA (Hierarchical Task Analysis); SADT (Structured Analysis and Design Technique); CTT (Concur Task Tree); K-MAD (Kernel of Model for Activity Description); GTA (Groupware Task Analysis); HAMSTERS notation; Task Model Standard (W3C).	Software Modeling	HTA (Hierarchical Task Analysis); SADT (Structured Analysis and Design Technique); Business case analysis; Suitable UML diagrams: Use case, Activity diagram, Class diagram, Sequence diagram, State machine diagram, Communication diagram, Timing diagram.
4	Standards and Guidelines for HCI design	ISO/IEC 25000 (2014); ISO/IEC 9126-1 (2001); ISO 9241-11 (1998); Ergonomic Criterion (Scapin and Bastien, 1993; Vanderdonckt, 1994); Accessibility standards and guidelines (WAI-W3C); Nielsen's Heuristics; Golden Rules of Interface Design.	Standards and Guidelines for design	ISO/IEC 25000 (2014); ISO/IEC 9126-1 (2001); Accessibility standards and guidelines (WAI-W3C); Domain-Specific Standards (E.g. security, critical systems ...).
5	Prototype for HCI requirements	Paper Prototyping/Sketches; Storyboards; Wireframes; Mockups; Wizard of Oz; Video prototyping.	Prototype for requirements	Paper Prototyping/Sketches; Storyboards; Wireframes; Mockups; Wizard of Oz; Video prototyping.
6	Techniques to validate HCI requirements	Thinking Aloud; Proto Task (K-MAD); Task Model Simulator (CTT); Focus Group for evaluate requirements.	Techniques to validate requirements	Thinking Aloud; Analysis; Simulations; Demonstrations; User Testing (using Prototypes); Perspective base-reading.
7	Architecture patterns for HCI	MVC (Model-View-Controller) Model; Arch Model; Language Model; SEEHEIM Model; PAC (Presentation-Abstraction-Control) Model; PAC-AMODEUS Model; CAMELEON-RT; Frameworks.	Architecture Patterns for SE	MVC (Model-View-Controller) Model; Service-Oriented Architecture (SOA); 3-Tier Model; Pipes and Filters; Suitable UML diagrams: Class diagram, Component diagram, Deployment diagram.
8	Design patterns for HCI	A Pattern Language for Human-Computer Interface Design; A Pattern Approach to Interaction Design; Pattern Languages in Interaction Design: Structure and Organization; Designing interfaces.	Design Patterns for SE	Design Patterns: Elements of Reusable Object-Oriented Software; GRASP - General Responsibility Assignment Software Patterns; Head First Design Patterns; Patterns of Enterprise Application Architecture.
9	Techniques for interaction modeling	MoLIC (Modeling Language for Interaction as Conversation); UAN (User Action Notation); TAG (Task-Action Grammar).	Interaction modeling for SE	Suitable UML diagrams: Component diagram, Interaction overview diagram.

#	Human-Computer Interaction (HCI)		Software Engineering (SE)	
	Category	**Examples of approaches**	**Category**	**Examples of approaches**
10	Techniques for HCI documentation	Style guide; Architecture for help; Training Program.	Techniques for final documentation	Style manual; ISO/IEC 26514 (2008).
11	Iterative and Evolutionary Prototypes	User interface toolkits; User interface builders; User interface development environments.	Prototype (system versions)	User interface toolkits; User interface builders; User interface development environments.
12	Evaluation methods for HCI verification	Unit test; Integration test; System test; Acceptance test; Installation test.	Verification methods	Unit test; Integration test; System test; Acceptance test; Installation test.
13	Evaluation methods for HCI review	Semiotic inspection; Formal usability inspection; Consistency inspection; Cognitive walkthrough; Groupware walkthrough; Guidelines review; Metaphors of human thinking; Heuristic evaluation	Review methods	Inspections; Structured walkthroughs; Guidelines review; Pair programming; Audits.
14	Evaluation methods for HCI validation	Usability testing; Communicability test; Standardized usability questionnaires; Post-experience interviews; User experience evaluation.	Validation methods	Acceptance test with users; Formal review; Tests of products (by end user/stakeholders); Analyses of product; Functional demonstrations.

From the HCI categories previously identified and associated to each practice of CMMI-DEV and expected result of MR-MPS-SW, we defined equivalent[5] categories considering the software engineering point of view. We used as a start point the HCI categories previously identified. These categories are the result of the analysis of CMMI-DEV that is a model well known by the SE community and software development industry that use this model. Then, from an *ad hoc* literature review and considering classical books on software engineering usually used in computer science courses in Brazil [24] [25] [27] [65] [66], we selected several approaches that could support each practice in each category. This proposition was peer-reviewed by five experts (Ph.D., with experience in the industry) from software engineering where they suggest other examples of approaches for each category. Our idea was to get more examples and confirm our propositions. We did not consider a larger number of experts to be necessary since our goal was not to be exhaustive in terms of examples of approaches but to quote some important ones. Moreover, the chosen approaches are classical ones, and some of them are well known by the community. The final result is also presented in Table 1 (two last columns). We can observe that some examples of some categories are overlapped (for instance, three first examples of category 1 and all examples of categories 5, 11 and 12). This overlapping is due to the fact that many approaches have been built by different areas of computer science and used by SE, HCI, among others.

With the categories of approaches defined, the questionnaire to be used for the survey was prepared. A web form composed of two parts was developed. The first part was to collect demographic data. The first

[5] In fact, the categories in each domain represent the same kind of approaches. Their names were only written differently to make the identification easier for each domain (HCI and SE). For instance: "Design patterns for HCI" and "Design patterns for SE". Moreover, while naming the categories we looked for the best way to represent the examples of approaches to that category. As consequence, in some cases the naming of a pair of categories were not following the same style. For instance, "Techniques for interaction modeling" that really represents several techniques from HCI and "Interaction modeling for SE" where the examples were diagrams from UML used for interaction modelling but that are not techniques.

four data fields (respondent identification, e-mail, degree subject, training area) were designed to identify the respondent (see **Appendix**).

The second part was composed of questions about HCI and SE approaches. Figure 3 presents a screenshot with one question about HCI and SE approaches. The final version of the questionnaire can be found in the **Appendix**. For each category (HCI and SE category), participants had to answer to what extent they know and use the approaches when implementing the practices of SPCM models, using a Visual Analogue Scale –VAS [67]. This scale is usually used in psychological studies that allow all arithmetic calculus. It consists of a horizontal line with two anchor points. We used the classical anchor points, from 0 to 10.

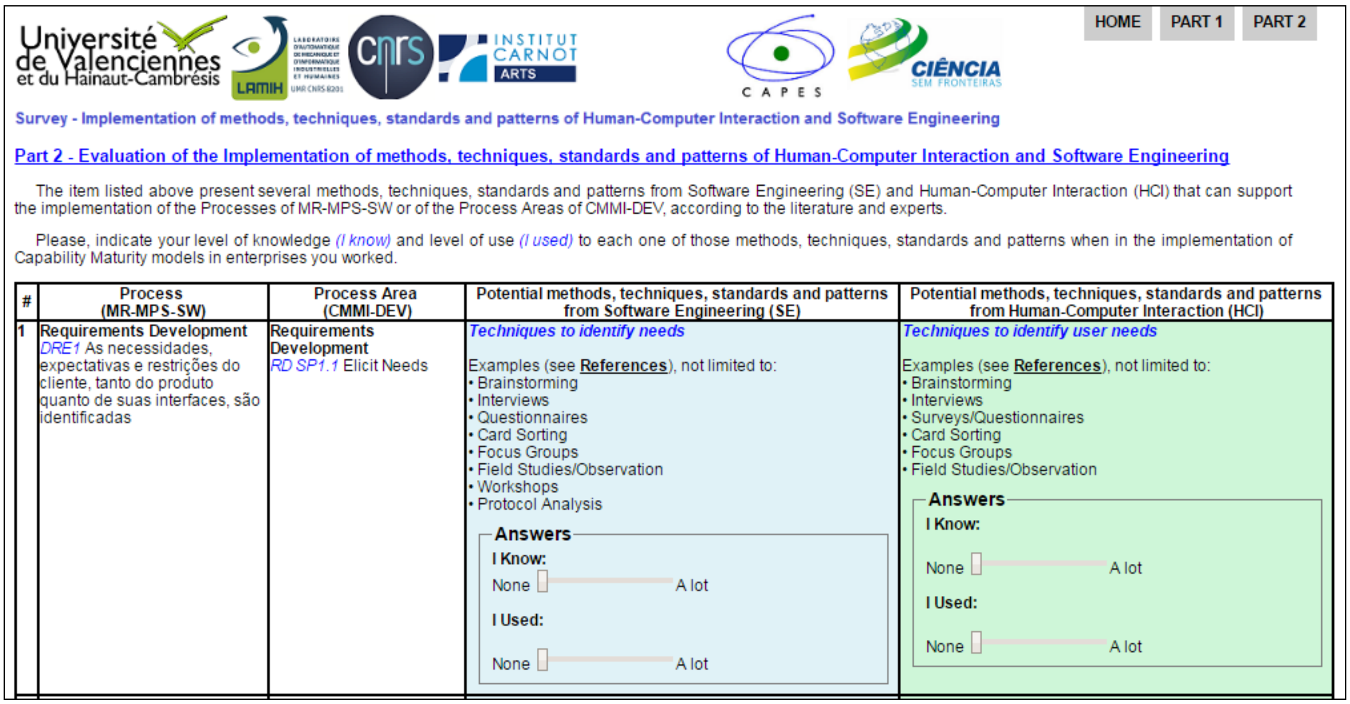


Figure 3. Questionnaire

One could say that it could be better to ask the consultants about each approach in each category and not about their overall perception per category. Indeed, this could give a very precise idea about used and known approaches. However, we had 111 approaches in total considering all categories (HCI and SE) which would result in a long and time-consuming questionnaire requiring. In order to ensure that we could get consultants available to answer the questionnaire, we chose to ask per category, which would imply a shorter time to answer.

3.1.3. Subjects and participation selection

The subjects selected for this study are the Brazilian SPCM model consultants who work in enterprises associated with the CMMI Institute and SOFTEX databases. Three enterprises are associated with the CMMI Institute and SOFTEX. Only SPCM model consultants that have implemented CMMI-DEV maturity level equal to or greater than 3 and MR-MPS-SW maturity level equal to or greater than D could participate in this study since our interest is the engineering process areas. Eleven partner organizations were selected from the CMMI Institute and SOFTEX databases. They regrouped 114 SPCM model consultants but only 40 have the required profile for our study. The report presented in

[63] shows that the majority of appraised enterprises has level G, which indicated that the majority of SPCM model consultants worked in implementations for this level (not for higher levels as required for our study) and justify the small population. Therefore the identified population, even though it is small (40 consultants), is considered representative because they are official consultants of the SPCM models and with the required profile for the study (they have already worked on engineering process areas in their consulting activity).

The sampling of the population was selected for non-probability sampling technique, and it is a quota sampling where the subjects are selected from various elements of a population. Table 2 shows the information about the SPCM models partner enterprises.

Table 2. Partner Organizations

Partner organizations	Number of consultants	Study population	SPCM models	CMMI Partner	SOFTEX Partner
A	17	8	CMMI-DEV and MR-MPS-SW	No	Yes
B	12	2	CMMI-DEV and MR-MPS-SW	No	Yes
C	18	4	CMMI-DEV and MR-MPS-SW	No	Yes
D	9	7	CMMI-DEV and MR-MPS-SW	Yes	Yes
E	11	3	CMMI-DEV and MR-MPS-SW	Yes	Yes
F	6	1	CMMI-DEV and MR-MPS-SW	No	Yes
G	14	6	CMMI-DEV and MR-MPS-SW	No	Yes
H	8	2	CMMI-DEV and MR-MPS-SW	Yes	Yes
I	1	1	CMMI-DEV	Yes	No
J	9	4	CMMI and MR-MPS-SW	No	Yes
K	9	2	MR-MPS-SW	No	Yes
Total	114	40			

After building the questionnaire, we carried out a pilot survey (pre-testing) with two SPCM model consultants to assess the survey instrument. They are Brazilian SPCM model consultants with the same characteristics as the population selected for this study. The two consultants answered the web questionnaire and filled in an evaluation form related to the instrument. The instrument was filled out by the consultants without the help of an instructor and the evaluation form was sent by e-mail.

The study was planned to be conducted off-line. The questionnaire was available on a website[6] in such a way that the SPCM models consultants answered the questionnaire in his/her time and environment, not being monitored. The survey request was sent by email. We contacted each person of the partner enterprises to explain about the survey and asked if they had the profile to answer the questionnaire (have participated in consulting for implementation of CMMI-DEV maturity level equal or greater than 3 and MR-MPS-SW maturity level equal or greater than D). We also confirmed with the coordinator of each partner enterprise whose members had the required profile.

3.2. Survey Execution

The survey request was sent by email on 11th Nov 2016, with four reminders till 30th March 2017. During the activity of operation of the study, we obtained 36 responses (sample size) out of 40 (our population size). Considering the confidence level of 95%, as usually recommended, we have a 5% error margin[7]. The margin of error is a percentage that describes how close the answer our sample gave is to

[6] https://surveytesis.000webhostapp.com/survey2/index1.html

[7] https://www.surveymonkey.com/mp/margin-of-error-calculator/

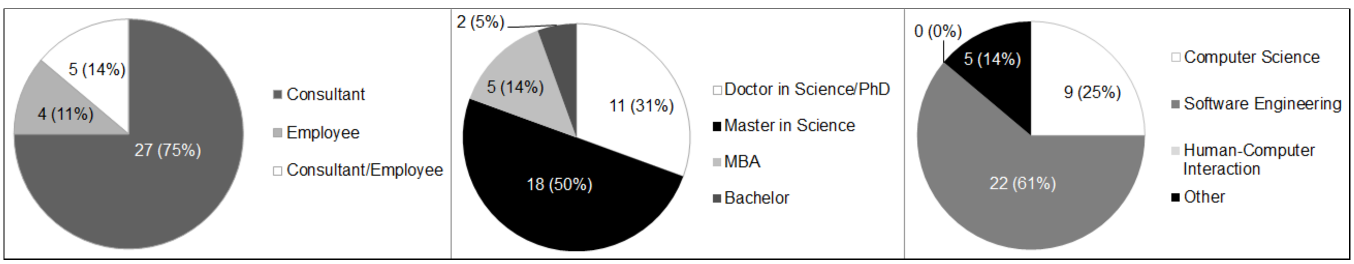
the “true value” in our population. In addition, we had a percentage of 90% as response rate, which was considered a reliable level.

3.2.1. Descriptive Data

The descriptive data was collected from part 1 (Characterization) of our questionnaire to identify the respondent (see all questions in **Appendix**). Figure 4 shows the profile of respondents where we note that:

- 27 participants (75%) (Figure 4 (a)) have worked as a consultant in the enterprises; 5 (14%) have worked as a consultant and were employees, and 4 (11%) were employees of the enterprises where the software was developed;
- regarding the training, Figure 4 (b) shows that most of them (18/36) has a master’s degree, 11 have a Ph.D., 5 participants (approximately 14%) have followed a Master of Business Administration (MBA) course and 2 have bachelor’s degree.
- about the training area Figure 4 (c) shows that 22 respondents (61%) declared Software Engineering as their domain (2 of them correspond to MBA courses, 8 have a doctorate, and 12 correspond to a Master of Science).

Moreover, the work time with SPCM implementation in the industry was between 5 and 25 years, and the mean was 12.85. The majority of respondents were placed between 10 and 16 years of work time.



(a) (b) (c)

Figure 4. Descriptive data

As regards the capability maturity models and levels (see Table 3) supported in the implementations, we can note that: (i) 25% (9/36) of the SPCM model consultants have supported implementations in CMMI-DEV level 3 and MR-MPS-SW level C; (ii) 22% (8/36) declared that they have supported implementations in CMMI-DEV level 3 and MR-MPS-SW levels C, D. One consultant implemented only CMMI-DEV levels 5, 4, 3, and three consultants implemented only MR-MPS-SW level C. In general, MR-MPS-SW level C and CMMI level 3 are the levels which are the most implemented in organizations. We recall that, considering the SOFEX report, the typical systems developed by the appraised enterprise are to support several administrative and commercial activities (such as electronic commerce, bank automation, web pages).

Table 3. Models and levels

Models and levels	# Consultants
CMMI-DEV 5-4-3	1
CMMI-DEV 5-4-3 and MR-MPS-SW A-B-C-D	3
CMMI-DEV 5-4-3 and MR-MPS-SW B-C-D	1
CMMI-DEV 5-4-3 and MR-MPS-SW C-D	3
CMMI-DEV 5-3 and MR-MPS-SW A-C	2
CMMI-DEV 5-3 and MR-MPS-SW C-D	2
CMMI-DEV 5-3 and MR-MPS-SW C	1
CMMI-DEV 5 and MR-MPS-SW A	1
CMMI-DEV 3 and MR-MPS-SW C	9
CMMI-DEV 3 and MR-MPS-SW C-D	8
CMMI-DEV 3 and MR-MPS-SW D	2
MR-MPS-SW C	3

3.2.2. Answering Hypotheses

To answer hypotheses 1 and 2 (H1 and H2) we start by calculating[8] the mean values for the 14 questions/categories of each variable (KHCI – Know of HCI approaches, KSE – Know of SE approaches, UHCI – Use of HCI approaches, and USE – Use of SE approaches) related to the 36 answers collected as presented in Table 4. Using the data from this table, we performed statistical tests that confirmed that the data (of each variable) follows a normal distribution and is homoscedastic.

Table 4. Mean of each question per variable

#	HCI categories	Mean KHCI	Mean UHCI	SE categories	Mean KSE	Mean USE
1	Techniques to identify user needs	7.3	6.2	Techniques to identify needs	8.9	8
2	Techniques to identify user and organizational requirements	6.7	5.2	Techniques to identify requirements	8.94	7.6
3	Task Modeling	3.6	1.9	Software Modeling	7.5	5.8
4	Standards and Guidelines for HCI design	4.2	2.32	Standards and Guidelines for design	6.1	4.3
5	Prototype for HCI requirements	5.7	3.9	Prototype for requirements	7.2	5.6
6	Techniques to validate HCI requirements	3.2	1.8	Techniques to validate requirements	7.3	5.5
7	Architecture patterns for HCI	3.8	2.7	Architecture Patterns for SE	7.6	6.3
8	Design patterns for HCI	2.7	1.6	Design Patterns for SE	5.5	3.8
9	Techniques for interaction modeling	1.4	0.8	Interaction modeling for SE	8	6.4
10	Techniques for HCI documentation	3.6	2.28	Techniques for final documentation	4.6	3
11	Iterative and Evolutionary Prototypes (system versions)	4.8	3.8	Prototype (system versions)	6.2	5.4
12	Evaluation methods for HCI verification	7.2	6.1	Verification methods	9.11	8.2
13	Evaluation methods for HCI review	4.1	2.2	Review methods	8.8	7.2
14	Evaluation methods for HCI validation	5.6	4	Validation methods	9.07	7.9
	Mean for all categories	4.56	3.20	**Mean for all categories**	7.48	6.07

[8] We used Minitab tool version 17 (2016/17) for all statistical analysis. http://www.minitab.com/en-us/products/minitab/

Knowing that the data is normal and homoscedastic, we could perform the paired t-test[9]. To that end, we used the mean values (see Table 4) to test each hypothesis (globally) for all questions/categories. The purpose of this test is to reject the null hypotheses. In the paired t-test KHCI minus KSE (KHCI - KSE) and UHCI minus USE (UCHI - USE) items were computed. As previously mentioned, we considered $\alpha = 0.05$ which allows us to build a confidence interval of 0.95. Table 5 presents the results of the paired t-test (step (iii)). The results allow us to refute the two null hypotheses ($H1_0$: KHCI - KSE = 0; and $H2_0$: UHCI - USE = 0) since $p < 0.0001$ is less than $\alpha = 0.05$ and the T-value (-6.91 for variable *Know*, and -7.57 for variable *Use*) is smaller than the critical value (-1,771)[10]. We accept, therefore, the two alternative hypotheses ($H1_A$: KHCI - KSE < 0; and $H2_A$: UHCI - USE < 0).

Table 5. Results of paired t-test

	N	Mean	Standard deviation	Standard error of the mean	T-value	p-value
KHCI	14	4.56	1.74	0.46	-6.91	0.0001
KSE	14	7.48	1.44	0.38		
Mean difference of Know	14	-2.92	1.58	0.42		
UHCI	14	3.20	1.70	0.45	-7.57	0.0001
USE	14	6.07	1.62	0.43		
Mean difference of Use	14	-2.87	1.41	0.37		

N = the number of answers for each variable.

After that, we performed the paired t-test considering all values (36 answers) for each question/category (14) to test each hypothesis (H1 for *Know*, and H2 for *Use*). The purpose of this test is to reject the null hypotheses (H1 and H2) for each question/category. Table 6 shows the results of the paired t-test considering the 14 items of each variable (*Know* and *Use*) using the 36 responses.

Table 6. Results of paired t-test for each item of the questionnaire

Item	Know (KHCI – KSE) - 36 responses			Item	Use (UHCI – USE) - 36 responses		
	Mean	Standard deviation	T-value		Mean	Standard deviation	T-value
1	-1.556	3.341	-2.79***	1	-1.792	3.360	-3.20***
2	-2.239	3.033	-4.43***	2	-2.356	3.110	-4.54***
3	-3.867	2.751	-8.43***	3	-3.817	2.885	-7.94***
4	-1.919	2.573	-4.48***	4	-1.989	2.653	-4.50***
5	-1.447	3.095	-2.81***	5	-1.647	3.212	-3.08***
6	-4.103	3.356	-7.34***	6	-3.692	3.605	-6.14***
7	-3.711	3.100	-7.18***	7	-3.600	3.178	-6.80***
8	-2.839	3.066	-5.56***	8	-2.186	2.754	-4.76***
9	-6.600	3.126	-12.67***	9	-5.594	3.869	-8.68***
10	-1.022	3.414	-1.80**	10	-0.706	3.705	-1.14*
11	-1.461	2.774	-3.16***	11	-1.536	2.738	-3.37***
12	-1.917	3.768	-3.05***	12	-2.131	3.691	-3.46***
13	-4.633	3.639	-7.64***	13	-4.947	3.218	-9.22***
14	-3.500	4.048	-5.19***	14	-3.869	3.933	-5.90***

* $p = 0.131$; ** $p = 0.041$; *** $0.001 \leq p \leq 0.004$.

[9] The paired t-test procedure is used to compare the mean difference between two populations when one believes that some dependency exists.

[10] https://support.minitab.com/en-us/minitab/18/help-and-how-to/statistics/basic-statistics/how-to/paired-t/interpret-the-results/all-statistics-and-graphs/#t-value

The results allow us to refute the null hypothesis ($H1_0$ – variable *Know*) and to accept the alternative hypothesis ($H1_A$ – variable *Know*) for all items (14 categories). For all categories (see Table 6 – variable *Know*), p-value < 0.05 (α value) and the T-value (see Table 6) is smaller than the critical value (-1,697). The null hypothesis ($H2_0$ – variable *Use*) was also refuted for thirteen items (13 categories). For these categories (see Table 6), p-value < 0.05 (α value) and the T-value < critical value (-1,697), see Table 6. However, the null hypothesis could not be rejected for category 10 (see Table 6 – variable *Use*) because the T-value (-1.14) is greater than the critical value (-1.697) and p-value (0.131) is greater than α (0.05).

4. Discussion

The most import finding of this study was the confirmation of hypotheses H1 and H2.

4.1. Hypothesis H1 (Perception of knowledge about HCI and SE approaches)

This hypothesis was confirmed showing that SPCM model consultants do not know HCI approaches as they know SE approaches when implementing the same specific practices of the SPCM models. Figure 5 shows the mean for each variable related to use (KHCI and KSE)[11].

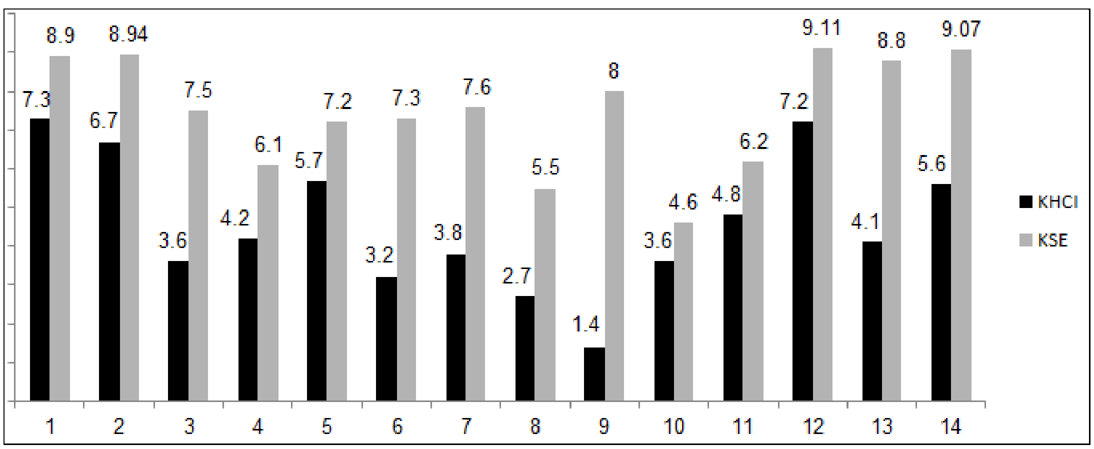


Figure 5. Mean of each question per variable (KHCI and KSE)

Analyzing Table 7 and Figure 5, we note that the category *techniques to identify user needs* (category 1 - KHCI) presented the highest level of **perception of knowledge** (mean = 7.3). The mean of all the categories for KHCI was 4.56, which is a bit lower than the central point (5) of our scale.

Table 7. HCI and SE rank for "perception of knowledge"

Rank	Categories rank for KHCI	Mean KHCI	Categories rank for KSE	Mean KSE
1	Techniques to identify user needs (category 1)	7.3	Verification methods (category 12)	9.11
2	Evaluation methods for HCI verification (category 12)	7.2	Validation methods (category 14)	9.07
3	Techniques to identify user and organizational requirements (category 2)	6.7	Techniques to identify requirements (category 2)	8.94
4	Prototype for HCI requirements (category 5)	5.7	Techniques to identify needs (category 1)	8.9
5	Evaluation methods for HCI validation (category 14)	5.6	Review methods (category 13)	8.8

[11] The numeration 1 to 14 represents each question/category as presented in Table 4.

However, when we analyze the means of each category for KHCI, we note that only five categories have a mean greater than the central point (5). The categories are presented in descending order in Table 7.

When we did the same analysis for each category of KSE (see Figure 5) we note that almost all categories (except the category 10 - *techniques for final documentation*) have the mean greater than the central point (five) of our scale. In Table 7, we show the categories that presented the five highest means and we can note that:

- for the HCI categories (*techniques to identify user needs*, *evaluation methods for HCI verification*, and *techniques to identify user and organizational requirements*) the suggested methods were completely or partially the same as ones proposed for its correspondent SE categories (e.g., Brainstorming, Unit test, Scenario, respectively). We believe that this fact can explain the similarity between the rank of the categories *techniques to identify user and organizational requirements*, a third position for HCI and *techniques to identify requirements*, a third position for SE;
- for the SE category (*validation methods*) the suggested methods for SE imply the participation of the users or end users (such as acceptance test with users and tests of products (by end user/stakeholders));
- for the SE category (*techniques to identify needs*) the suggested methods were completely or partially the same as for the HCI category (*techniques to identify user needs*);
- the categories ranked (for both domains) in the first five positions are associated with the practices of three process areas: requirements development, verification and validation, which are the areas that some examples of approaches are overlapped (see section 3.1.2).

The category *techniques for interaction modeling* (category 9 - KHCI) and the category *techniques for final documentation* (category 10 - KSE) are the categories with the lowest means regarding the **perception of knowledge**. We believe that the result found for the category *techniques for interaction modeling* is due to the fact that these techniques are relatively young in the HCI community and relatively new in Brazil [68] [69] [70] [71]; it is possible that the approaches are not yet being used in practice.

In addition, the survey revealed the little perception of knowledge (mean = 3.6) and even less perception of use (mean = 1.9) for the approaches of the HCI category *Task Modeling*. Approaches of this category support the elicitation of user needs, the establishment of user interface requirements, and the analyses of user interface requirements. A task model is a model-based approach to user interface design where the results are models that describe the activities that should be performed in order to reach users' goals [72]. Task models are useful in different phases of the development of interactive applications: requirements analysis, design of the user interface, usability evaluation, documentation and others [72]. As consequence, not knowing and/or use these approaches may have a bad impact on the whole development of the software system.

Similarly, the little perception of knowledge (mean = 3.2) and of use (mean = 1.8) for the approaches of the HCI category *Techniques to validate HCI requirements* were identified. These approaches aim to analyze user interface requirements in order to balance stakeholder needs with design constraints and minimize the risk of user interface development.

The results found for the categories *Architecture patterns for HCI* and *Design patterns for HCI* were also low. These categories offer approaches to support the architectural decisions to design and implement the user interface. The architecture pattern support functional requirements and quality attributes requirements (such as usability), and is used to create the product architecture [19]; and the design patterns provide solutions to specific usability problems related to interface design and interaction [73]. These patterns can be used in different phases of the software development process (for example, in the interface evaluation), according to the development methodology adopted [74] [75], which shows the importance of being known and used. Moreover, the patterns can be used to improve, by way of example, what is proposed by guidelines and HCI heuristics (which in our work are part of the *Standards and Guidelines for HCI design* category).

According to Furtado et al. [76] HCI is not yet widely known and has not yet been formally adopted in the industry. The authors [76] explains the plans to bring closer industry and academy through the evangelization of HCI research and practice in Brazil. This link between industry and academy is also discussed by Scheiber et al. [56]. They argue that the universities are considered as important sources of knowledge regarding HCI by software producers. We defend that this lack of knowledge on HCI has a relevant impact on the performance of the industry to develop quality systems, as shown by Theofanos [6] and Myers & Rosson [7].

4.2. Hypothesis H2 (Perception of use about HCI and SE approaches)

This hypothesis was confirmed showing that SPCM models consultants do not use HCI approaches as they use SE approaches when implementing the same specific practices of the SPCM models.

Figure 6 shows the mean for each variable related to use (UHCI and USE)[12]. The category *techniques to identify user needs* (category 1 – UHCI) presented the highest level of **perception of use** (mean = 6.2). The global mean of all categories for UHCI was 3.20, which is lower than the central point (five) of our scale.

When we analyzed the means of each category of UHCI (see Figure 6), we noted that only three categories have a mean greater than the central point (5) of our scale. The categories are presented in descending order in Table 8.

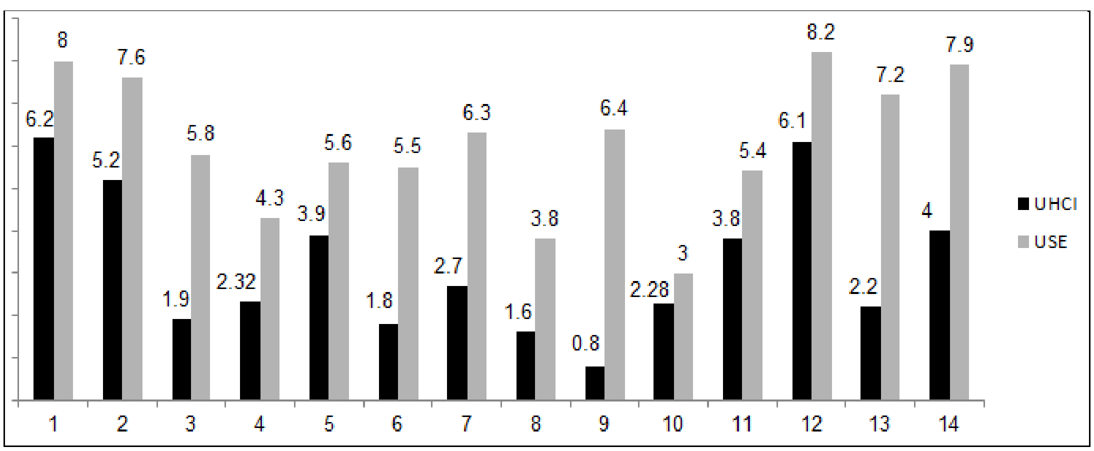


Figure 6. Mean of each question per variable (UHCI and USE)

When we performed the same analysis for each category of USE (see Figure 6) we note that almost all categories (except for three categories: *standards and guidelines for design*, *design patterns*, and *techniques for final documentation*) have a mean greater than the central point (five) of our scale. In Table 8, we show the categories that presented the three highest means.

[12] The numeration 1 to 14 represents each question/category as presented in Table 4.

Table 8. HCI and SE rank for "perception of use"

Rank	Categories rank for UHCI	Mean UHCI	Categories rank for USE	Mean USE
1	Techniques to identify user needs (category 1)	6.2	Verification methods (category 12)	8.2
2	Evaluation methods for HCI verification (category 12)	6.1	Techniques to identify needs (category 1)	8.0
3	Techniques to identify user and organizational requirements (category 2)	5.2	Validation methods (category 14)	7.9

The categories that presented the three first positions for the **perception of use** are the same categories that are placed in the three first positions for the perception of knowledge, in both domains (HCI and SE). As previously explained the suggested methods for these categories (HCI and SE) are completely or partially the same. Similarly to the **perception of knowledge**, the category *techniques for interaction modeling* (category 9 - KHCI, mean = 0.8) and the category *techniques for final documentation* (category 10 - KSE, mean = 3.0) are the categories with the lowest means regarding the **perception of use**.

4.3. Comparing results with literature

As previously discussed in section 2.2, Vukelja et al. [51] investigate the use of SE methods in software development. Regarding modules and systems tests, they found that: (i) the modules are tested in 76.2% of the cases; and (ii) the systems in 98.1% of the cases. In our study, we found that the SE category *verification methods* was the category most used by consultants (mean = 8.2). They also said that the documentation for the end user is written in 34.2% of the cases in parallel with the development and at the end; and in 65.8% of the cases only at the end. Our results show that the SE category *techniques for final documentation* was the least used category by consultants (mean = 3.0).

Analyzing usability tests, Vukelja et al. [51] found that this type of test is conducted in only 37.9% of the cases. In our case, the SE category *validation methods* (that include usability tests) was classified in the third position (see Table 8) of our ranking. The approach *Acceptance test with users* is placed in this category.

We analyzed our top ten HCI approaches against the results presented in the literature. To perform this analysis, we considered the HCI category ranking (Table 9) of each study from literature, and we looked for the same examples of approaches defined in the categories of our research. Analyzing the ranking of the HCI techniques/methods used in practice by UI/UX/UCD/usability practitioners, we found that:

- Venturi et al. [50] and UXPA Curitiba [60] shared three techniques that had the same classification (*hi-fi prototypes, heuristic evaluation*, and *usability test* – classified as second, fourth and fifth techniques in Table 9);
- Ji & Yun [49] and Venturi et al. [50] did not share techniques with the same classification rank.

Similarly, we analyzed the rank of the HCI techniques/methods (Table 9) used in practice cited by three studies and that were performed with software developers. We can note that:

- Salgado et al. [59] and our study shared two techniques that had the same classification (see the techniques classified as first and tenth in Table 9 – these techniques correspond to requirements development and verification process areas, respectively);
- Hussein et al. [54] and our study shared two techniques (techniques placed in Table 9 as second and fourth position – these techniques correspond to verification and validation process areas, respectively);

- Ji & Yun [49] and our study shared two techniques with the same rank classification (see the techniques classified as third and sixth in Table 9 – the third was associated to requirements development process area, and the sixth to technical solution, product integration, verification and validation process areas).

Analyzing the five first HCI categories (techniques/methods) classified by our study (last column of Table 9) we note that the first, third and fifth categories were associated to the requirements development process area; the second and fourth categories correspond to the verification and validation process areas, respectively. These results were corroborated by some studies cited in this research that discussed the frequent use of HCI methods for validation [49] [50] [54] [59] [60] and verification [50] [54] [60], and also the use of techniques to identify user needs in the initial phases of software development [49] [50] [54] [59] [60].

4.4. Threats to Validity of the results

We considered the four threats of validity proposed by [61] (construct, internal, external and conclusion validity) trying to define some mitigations as described below.

Threats to the **construct validity** illustrate the relationship between theory and observation and the questions of whether the treatment adequately reflects the cause; whether the result adequately reflects the effects. In our case, the question is whether the items (HCI and SE approaches) to be evaluated adequately reflect the application for the practices of the SPCM models. To minimize this threat, the web questionnaire was built using original text and examples from the official documentation of CMMI-DEV and MR-MPS-SW models; the HCI approaches were evaluated with twenty domain experts [64] (after a carrying out an *ad hoc* study[13] of literature); the SE approaches were peer-reviewed by five domain experts after an ad hoc study of literature. Moreover, the category names were defined to represent the examples of approaches proposed for each specific domain (HCI and SE) even when there were some approaches similar in both of them. Another threat to the construction validity concerns the interpretation of the practices of the CMMI-DEV and the MR-MPS-SW model for the customization of the instrument. This risk was accepted because we used the equivalence mapping [47] of the models to build the instrument.

Threats to the **internal validity** draw from influences that can affect the independent variables with respect to causality without the researchers' knowledge. In our case, this threat is associated with the subjects involved in the study. The subjects were selected by quota sampling, SPCM model consultants who have implemented maturity levels "A, B, C or D" of the MR-MPS-SW model and/or maturity levels "5, 4 or 3" of the CMMI-DEV model. The subjects should characterize their perception of knowledge and use related to the implementation of HCI and SE approaches in different projects. A potential risk is that a person who has not done implementations for the levels mentioned in the study answers the survey. To minimize this risk, we selected SPCM model consultants who are associated to partner enterprises of the CMMI Institute and SOFTEX. In addition, we explicitly asked them if they have the required experience and we confirmed this with the coordinator of each partner enterprises.

[13] The study was made for a particular purpose; it is not planned before it happens. https://dictionary.cambridge.org/dictionary/english/ad-hoc

Table 9. Rank of the HCI techniques/methods used in the practice

Rank	Salgado et al. [25]	UXPA Curitiba [26]	Hussein et al. [20]	Ji and Yun [15]		Venturi et al. [16]	Gonçalves et al.
	Subject: software developer	*Subject: usability/UX professionals*	*Subject: software developer*	*Subject: development practitioners*	*Subject: UI/usability practitioners*	*Subject: UCD practitioners*	*Subject: software developer*
1	User interviews	Lo-fi Prototypes	Task analysis	Task analysis	Task analysis	User interviews	Techniques to identify user needs - *e.g.: Interviews*
2	Lo-fi prototypes	Hi-fi prototypes	User Acceptance Test	Evaluate existing system	Evaluate existing system	Hi-fi prototyping	Evaluation methods for HCI verification - *e.g.: acceptance test*
3	Hi-fi prototypes	User interviews	User experience	User analysis/profiling	User analysis/ profiling	Lo-fi prototyping	Techniques to identify user and organizational requirements -*e.g.: user Profile (detailed)*
4	Contextual analysis	Heuristic evaluation	Evaluate existing system	Surveys	Surveys	Expert or heuristic evaluation	Evaluation methods for HCI validation
5	Usability tests	Usability test	Surveys	Scenarios of use	Scenarios of use	Qualitative, quick and dirty usability test	Prototype for HCI requirements
6	Heuristic evaluation	Personas	Heuristics evaluation, usability expert evaluation	Screen mock-up test	Heuristics evaluation, usability expert evaluation	Observation of real usage	Iterative and Evolutionary Prototypes (system versions)
7	Personas	Survey	Scenarios of use	Navigation design	Navigation design	Scenarios	Architecture patterns for HCI
8	Survey	Contextual analysis	User analysis/ profiling	Usability checklists	Usability checklists	Style guides	Standards and Guidelines for HCI design
9	Remote usability tests	UX Training	Lab usability testing	Participatory design	Focus group interview	Early human factors analysis	Techniques for HCI documentation
10	Guidelines/ Checklist review	Card sorting	Navigation design	Lab usability testing	Lab usability testing	Competitive analysis	Evaluation methods for HCI review

Threats to the **external validity** are conditions that limit our ability to generalize the results of our experiment outside the scope of our study. This study was performed in the Brazilian context. Although Brazil is one of the top ten countries that apply CMMI-DEV, according to CMMI Institute, we cannot generalize the results of this study to any country.

Threats to the **conclusion validity** are those that affect the ability to draw the correct conclusion about the relation between the treatment and the outcome of our study. Analyzing our hypotheses, we identified the risk of an SPCM model consultant having answered that he/she does not know and/or does not use a category of HCI or SE approach because he/she does not recognize the approach's name. To minimize this threat, we included several examples and their bibliographic references on the web questionnaire. In this way, the SPCM models consultants could consult the list of bibliographic references in case of doubt. Another risk is that the consultants have answered each question regarding only the approach examples rather than the category, which represent the kind of approaches to support each practice. To mitigate this risk, we included the text "not limited to" before presenting the list of examples. Moreover, to analyze this risk we asked the two consultants who participated in the pre-test what the reasoning used to answer the questions was. They answered that they considered the category in general although they had read the list of examples. Therefore, we considered that the risk was potentially weak. We considered that even though consultants could answer based intuitively on an average of their perception of knowledge and use of all examples in the list from a category, they were evaluating both HCI and SE in the same way, and thus we could analyze the results of the evaluation one against the other. Therefore, we decided to accept this risk. Finally, another threat to validity is about the training area of the consultants in the conclusions. We are aware that training (master's degree or Ph.D.) in Human-Computer Interaction is much younger than courses in software engineering; therefore we have the risk that the answers were biased towards the original training of the SPCM model consultants who in majority declared they had been trained in software engineering. Considering that HCI issues are usually integrated as courses in the master's degree and Ph.D. in different disciplines (such as software engineering and computer systems in general), we accepted this risk.

5. Conclusion

This paper presents an empirical study performed with Brazilian SPCM model consultants. These consultants have implemented SPCM models in Brazilian organizations that were evaluated in these models. The objective was to investigate **the perception of knowledge** and **use** of HCI and SE approaches by SPCM model consultants in the context of SPCM model implementations (CMMI-DEV and MR-MPS-SW).

Analyzing the results of the survey, we concluded that the population of this study does not know and does not use HCI approaches as they know and use SE approaches when applying the same engineering specific practices of the SPCM models. We can suppose that if the SPCM model consultants do not know HCI approaches, the industry will hardly use these approaches.

As consequence, this study shows the need for dissemination of HCI approaches in industry. We believe that one way to do that is by augmenting the number of hours of HCI classes in Computer Science undergraduate courses. Moreover, it shows the importance of having training on the integration of HCI and SE to support software development and maintenance.

Furthermore, we are aware that the industry which applies SPCM models usually use software processes that address these models; therefore it is essential to work on the definition of a software process for interactive systems development that explicitly indicates which HCI approaches to be used in order to have them really applied in industry.

This survey is also being applied in a worldwide context for CMMI-DEV model consultants [77]. Our ongoing works include: (i) the definition of a methodological guide to support SPCM model consultants in their choice of the adequate HCI approach at the right moment; and, (ii) the definition of a software development process for interactive systems, which can be classical or innovative, such as interactive applications on tabletop(s) with tangible objects (see for instance [78] [79]), integrating HCI approaches to support each activity. As future work, we plan to perform several studies in the industry using our methodological guide to verify if knowing which approaches to use, the software developers will use them. Future work can also be performed to investigate more in detail which specific approach is (or is not) known/used.

Acknowledgments
The authors thank the consultants who responded the survey and CAPES - Science without Borders Program for the financial support for this work.

References

[1] A. M. Moreno, A. Seffah, R. Capilla, and M. I. Sánchez-Segura, “HCI Practices for Building Usable Software,” *Computer*, vol. 46, no. 4, pp. 100–102, 2013.
[2] S. Rochimah, H. I. Rahmani, and U. L. Yuhana, “Usability characteristic evaluation on administration module of Academic Information System using ISO/IEC 9126 quality model,” in *International Seminar on Intelligent Technology and Its Applications*, 2015, pp. 363–368.
[3] D. Ameller, M. Galster, P. Avgeriou, and X. Franch, “A survey on quality attributes in service-based systems,” *Softw. Qual. J.*, vol. 24, no. 2, pp. 271–299, 2016.
[4] F. Pinciroli, “Improving Software Applications Quality by Considering the Contribution Relationship Among Quality Attributes,” *Procedia Comput. Sci.*, vol. 83, pp. 970–975, 2016.
[5] M. A. Kabir, M. U. Rehman, and S. I. Majumdar, “An analytical and comparative study of software usability quality factors,” in *Proceedings of the 7th IEEE International Conference on Software Engineering and Service Science*, 2016, pp. 800–803.
[6] M. Theofanos, “A Practical Guide to the CIF: Usability Measurements,” *Interactions*, vol. 13, no. 6, pp. 34–37, 2006.
[7] B. A. Myers and M. B. Rosson, “Survey on User Interface Programming,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 1992, pp. 195–202.
[8] C. G. Wangenheim, J. C. R. Hauck, C. F. Salviano, and A. von Wangenheim, “Systematic Literature Review of Software Process Capability/Maturity Models,” in *Proceedings of International Conference SPICE*, 2010.
[9] CMMI Product Team, “Maturity Profile Report,” Jul. 2015.
[10] C. Estorilio, G. R. M. Vaz, F. C. de Lisboa, and L. de O. F. Bessa, “The relationship between industrial process maturity and quality certification,” *Comput. Stand. Interfaces*, vol. 39, pp. 22–33, 2015.
[11] M. Kalinowski, K. Weber, G. Santos, N. Franco, V. Duarte, and G. Travassos, “Software Process Improvement Results in Brazil Based on the MPS-SW Model,” *Software Quality Professional*, vol. 17, no. 4, pp. 15–28, 2015.
[12] D. O’Neill, “The Way Forward: A Strategy for Harmonizing Agile and CMMI,” *CrossTalk Magazine - The Journal of Defense Software Engineering*, vol. 29 (4), p. 40, Aug-2016.
[13] D. Sharma, N. Narula, D. Lee, and T. R. Leishman, “Agile 5 – Using High Maturity CMMI Practices to Improve Agile Processes and Achieve Predictable Results,” *CrossTalk Magazine - The Journal of Defense Software Engineering*, vol. 29 (4), p. 40, Aug-2016.
[14] S. W. Ambler and M. Lines, “The Disciplined Agile Framework: A Pragmatic Approach to Agile Maturity,” *CrossTalk Magazine - The Journal of Defense Software Engineering*, vol. 29 (4), p. 40, Aug-2016.
[15] CMMI Product Team, “A Guide to Scrum and CMMI®: Improving Agile Performance with CMMI,” CMMI Institute, 2016.
[16] J. Garzás and M. C. Paulk, “A case study of software process improvement with CMMI-DEV and Scrum in Spanish companies,” *J. Softw. Evol. Process*, vol. 25, pp. 1325–1333, 2013.
[17] Softex, “MPS.BR - Brazilian Software Process Improvement, MR-MPS-SW - General Guide,” 2016.

[18] H. Oktaba and A. Vázquez, “MoProSoft®: A Software Process Model for Small Enterprises,” in *Software Process Improvement for Small and Medium Enterprises: Techniques and Case Studies*, H. Oktaba and M. Piattini, Eds. Hershey: New York: Information Science Reference, IGI Global, 2008, pp. 170–176.
[19] CMMI Product Team, “CMMI® for Development,” Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, USA, CMU/SEI-2010-TR-033, 2010.
[20] W. W. Royce, “Managing the Development of Large Software Systems: Concepts and Techniques,” in *Proceedings of WESCON*, 1970, vol. 26, pp. 1–9.
[21] R. H. Thayer and A. D. McGettrick, *Software Engineering: A European Perspective*. Los Alamitos, CA, USA: IEEE Computer Society Press, 1993.
[22] ISO/IEC, “Systems and software engineering – Software life cycle processes (ISO/IEC 12207:2008),” Geneva, Switzerland, 2008.
[23] ISO/IEC, “Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE (ISO/IEC 25000:2014),” Geneva, Switzerland, 2014.
[24] C. Larman, *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*. New Jersey: Prentice Hall, 2004.
[25] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1994.
[26] A. Goldberg, *SMALLTALK-80: The Interactive Programming Environment*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1984.
[27] S. L. Pfleeger and J. M. Atlee, *Software Engineering: Theory and Practice*, 4 edition. Upper Saddle River, N.J.; London: Pearson, 2009.
[28] D. Hix and H. R. Hartson, *Developing User Interfaces: Ensuring Usability Through Product and Process*. New York, NY, USA: John Wiley & Sons, Inc., 1993.
[29] J. Nielsen, *Usability Engineering*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.
[30] D. J. Mayhew, *The Usability Engineering Lifecycle: a practicioner's handbook for user interface design*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1999.
[31] B. W. Boehm, “A Spiral Model of Software Development and Enhancement,” *IEEE Comput.*, vol. 21, no. 5, pp. 61–72, 1988.
[32] A. Valentin, G. Valléry, and R. Lugongsang, *L'evaluation ergonomique des logiciels, une démarche itérative de conception*. Montrouge: ANACT, 1993.
[33] C. Kolski and P. Loslever, “An HCI-Enriched Model for Supporting Human-Machine Systems Design and Evaluation,” *IFAC Proc. Vol.*, vol. 31, no. 26, pp. 419–424, 1998.
[34] D. Diaper and N. A. Stanton, *The Handbook of Task Analysis for Human-Computer Interaction*. Mahwah, NJ: Lawrence Erlbaum Associates, 2004.
[35] C. Courage and K. Baxter, *Understanding your users: a practical guide to user requirements, methods, tools, and techniques*, 1st Edition. San Francisco, CA: Morgan Kaufmann Publishers, 2005.
[36] M. Beaudouin-Lafon and W. E. Mackay, “Prototyping tools and techniques.,” in *Human-Computer Interaction: Development Process*, A. Sears and J. A. Jacko, Eds. NW: CRC Press, 2009, pp. 121–143.
[37] J. Coutaz, “PAC, an Object-Oriented Model for Dialog Design,” in *Proceedings of 2nd IFIP International Conference on Human-Computer Interaction (INTERACT 87)*, Stuttgart, Germany, 1987, pp. 431–436.
[38] L. Bass *et al.*, “The Arch model: Seeheim revisited,” in *Proceedings of User Interface Developers Workshop*, Seeheim, 1991.
[39] J. Borchers, *A Pattern Approach to Interaction Design*. New York, NY, USA: John Wiley & Sons, Inc., 2001.
[40] M. van Welie and G. C. van der Veer, “Pattern Languages in Interaction Design: Structure and Organization,” in *Human-Computer Interaction - INTERACT'03*, 2003, pp. 527–534.
[41] B. Shneiderman, C. Plaisant, M. Cohen, and S. Jacobs, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 5th Edition. Pearson, 2009.
[42] J. Sauro and J. R. Lewis, *Quantifying the User Experience - Practical Statistics for User Research*. Morgan Kaufmann, MA., 2012.
[43] A. Assila, K. M. Oliveira, and H. Ezzedine, “Standardized Usability Questionnaires: Features and Quality Focus,” *Electron. J. Comput. Sci. Inf. Technol. EJCSIT*, vol. 6, no. 1, pp. 15–31, 2016.

[44] J. Nielsen, “Usability Inspection Methods,” J. Nielsen and R. L. Mack, Eds. New York, NY, USA: John Wiley & Sons, Inc., 1994, pp. 25–62.
[45] T. Mahatody, M. Sagar, and C. Kolski, “State of the Art on the Cognitive Walkthrough Method, Its Variants and Evolutions,” *Int. J. Human–Computer Interact.*, vol. 26, no. 8, pp. 741–785, 2010.
[46] M. Y. Ivory and M. A. Hearst, “The State of the Art in Automating Usability Evaluation of User Interfaces,” *ACM Comput. Surv.*, vol. 33, no. 4, pp. 470–516, 2001.
[47] Softex, “MPS.BR - Brazilian Software Process Improvement, Implementation Guide - Part 11: Implementation and Evaluation of MR-MPS-SW: 2016 in conjunction with CMMI-DEV v1.3,” 2016.
[48] L. Grossi, J. A. Calvo-Manzano, and T. S. Feliu, “High-maturity levels: achieving CMMI ML-5 in a consultancy company,” *J. Softw. Evol. Process*, vol. 26, no. 9, pp. 808–817, 2014.
[49] Y. G. Ji and M. H. Yun, “Enhancing the Minority Discipline in the IT Industry: A Survey of Usability and User-Centered Design Practice,” *Int. J. Human–Computer Interact.*, vol. 20, no. 2, pp. 117–134, 2006.
[50] G. Venturi, J. Troost, and T. Jokela, “People, Organizations, and Processes: An Inquiry into the Adoption of User-Centered Design in Industry,” *Int. J. Human–Computer Interact.*, vol. 21, no. 2, pp. 219–238, 2006.
[51] L. Vukelja, L. Müller, and K. Opwis, “Are Engineers Condemned to Design? A Survey on Software Engineering and UI Design in Switzerland,” in *Human-Computer Interaction – INTERACT 2007: 11th IFIP TC 13 International Conference*, vol. Part II, C. Baranauskas, P. Palanque, J. Abascal, and S. D. J. Barbosa, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 555–568.
[52] I. Hussein, E. A. A. Seman, and M. Mahmud, “Perceptions on Interaction Design in Malaysia,” in *Internationalization, Design and Global Development: Third International Conference, IDGD 2009, Held as Part of HCI International 2009*, N. Aykin, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 356–365.
[53] I. Hussein, M. Mahmud, and A. W. Yeo, “HCI practices in Malaysia: A reflection of ICT professionals’ perspective,” in *Proceedings of International Symposium on Information Technology*, 2010, pp. 1549–1554.
[54] I. Hussein, M. Mahmud, and A. O. Md. Tap, “User Experience Design (UXD): a survey of user interface development practices in Malaysia,” in *Proceedings of SEANES 2012*, Langkawi, Malaysia, 2012.
[55] H. M. Hao and A. Jaafar, “Usability in Practice: Perception and Practicality of Management and Practitioners,” in *Proceedings of International Conference on Pattern Analysis and Intelligence Robotics*, 2011, vol. 2, pp. 211–216.
[56] F. Scheiber *et al.*, “Software Usability in Small and Medium Sized Enterprises in Germany: An Empirical Study,” in *Software for People: Fundamentals, Trends and Best Practices*, A. Maedche, A. Botzenhardt, and L. Neer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 39–52.
[57] C. Ardito, P. Buono, D. Caivano, M. F. Costabile, and R. Lanzilotti, “Investigating and promoting UX practice in industry: An experimental study,” *Int. J. Hum.-Comput. Stud.*, vol. 72, no. 6, pp. 542–551, 2014.
[58] A. A. Ogunyemi, D. Lamas, E. R. Adagunodo, F. Loizides, and I. B. D. Rosa, “Theory, Practice and Policy: An Inquiry into the Uptake of HCI Practices in the Software Industry of a Developing Country,” *Int. J. Human–Computer Interact.*, vol. 32, no. 9, pp. 665–681, 2016.
[59] A. Salgado, L. A. Amaral, A. P. Freire, and R. P. M. Fortes, “Usability and UX Practices in Small Enterprises: Lessons from a Survey of the Brazilian Context,” in *Proceedings of the 34th ACM International Conference on the Design of Communication*, New York, NY, USA, 2016, pp. 18:1–18:9.
[60] UXPA Curitiba, “Profile of UX professionals in Brazil 2015,” UXPA - User Experience Professionals Association Curitiba, 2016.
[61] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in Software Engineering*. Springer-Berlin Heidelberg, 2012.
[62] V. R. Basili and H. D. Rombach, “The TAME Project: Towards Improvement-oriented Software Environments,” *IEEE Trans Softw Eng*, vol. 14, no. 6, pp. 758–773, Jun. 1988.
[63] G. H. Travassos and M. Kalinowski, “iMPS 2013 : evidências sobre o desempenho das empresas que adotaram o modelo MPS-SW,” SOFTEX, Campinas/SP/Brazil, 2013.
[64] T. G. Gonçalves, K. M. Oliveira, and C. Kolski, “Identifying HCI Approaches to support CMMI-DEV for Interactive System Development,” *Comput. Stand. Interfaces*, vol. 58, pp. 53–86, 2018.
[65] R. S. Pressman and B. Maxim, *Software Engineering: A Practitioner’s Approach*, 8 edition. New York, NY: McGraw-Hill Education, 2014.
[66] I. Sommerville, *Software Engineering*, 10 edition. Harlow: Pearson, 2015.

[67] M. E. Wewers and N. K. Lowe, “A critical review of visual analogue scales in the measurement of clinical phenomena,” *Res. Nurs. Health*, vol. 13, no. 4, pp. 227–236, 1990.
[68] S. D. J. Barbosa and M. G. Paula, “Designing and Evaluating Interaction as Conversation: A Modeling Language Based on Semiotic Engineering,” in *Interactive Systems. Design, Specification, and Verification: 10th International Workshop, DSV-IS 2003, Funchal, Madeira Island, Portugal, June 11-13, 2003.*, J. A. Jorge, N. J. Nunes, and J. F. e Cunha, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 16–33.
[69] R. O. Prates, “HCI Brazilian Community - After 10 Years,” in *Proceedings of the 11th IFIP TC 13 International Conference on Human-computer Interaction - Volume Part II*, Berlin, Heidelberg, 2007, pp. 631–632.
[70] C. S. Souza, M. C. C. Baranauskas, R. O. Prates, and M. S. Pimenta, “HCI in Brazil: Lessons Learned and New Perspectives,” in *Proceedings of the VIII Brazilian Symposium on Human Factors in Computing Systems*, Porto Alegre, Brazil, Brazil, 2008, pp. 358–359.
[71] R. O. Prates *et al.*, “HCI Community in Brazil—sweet 16!,” *Interactions*, vol. 20, no. 6, pp. 80–81, 2013.
[72] F. Paternò, “Towards a UML for Interactive Systems,” in *Proceedings of the 8th IFIP International Conference on Engineering for Human-Computer Interaction*, London, UK, UK, 2001, pp. 7–18.
[73] E. Folmer, M. van Welie, and J. Bosch, “Bridging patterns: An approach to bridge gaps between SE and HCI,” *Inf. Softw. Technol.*, vol. 48, no. 2, pp. 69–89, 2006.
[74] A. C. da Silva, J. C. A. Silva, R. A. D. Penteado, and S. R. P. da Silva, “Aplicabilidade de Padrões de Engenharia de Software e de IHC no Desenvolvimento de Sistemas Interativos,” in *Anais do IV Congresso Brasileiro de Computação*, 2004, pp. 118–123.
[75] A. C. da Silva, J. C. A. Silva, R. A. D. Penteado, and S. R. P. da Silva, “Integrando visões de IHC e de ES por Padrões no desenvolvimento por prototipação descartável,” in *Proceedings of the Latin American Conference on Human-computer Interaction*, New York, NY, USA, 2005, pp. 223–234.
[76] E. S. Furtado, T. Conte, S. D. J. Barbosa, P. Melo, C. Maciel, and C. R. B. Souza, “Evangelizing HCI Research and Practice in Brazil,” *Interactions*, vol. 23, no. 6, pp. 78–78, 2016.
[77] T. G. Gonçalves, K. M. Oliveira, and C. Kolski, “A study about HCI in practice of interactive system development using CMMI-DEV,” in *Proceedings of the 29th Conference on Interaction Homme-Machine*, 2017, pp. 169–177.
[78] S. Kubicki, M. Wolff, S. Lepreux, and C. Kolski, “RFID interactive tabletop application with tangible objects: exploratory study to observe young children’ behaviors,” *Pers. Ubiquitous Comput.*, vol. 19, no. 8, pp. 1259–1274, 2015.
[79] A. Bouabid, S. Lepreux, and C. Kolski, “Design and evaluation of distributed user interfaces between tangible tabletops,” *Univers. Access Inf. Soc.*, pp. 1–19, 2017.

Appendix

***Survey* - Implementation of methods, techniques, standards and patterns of Human-Computer Interaction and Software Engineering**

This survey aims to evaluate to what extent methods, techniques, standards and patterns of Software Engineering and Human-Computer Interaction are used by software developers that have implemented the maturity levels (A, B, C or D) of the MR-MPS-SW model (Reference Model MPS for Software) and/or the maturity levels (5, 4 or 3) of the CMMI-DEV model (Capability Maturity Model Integration for Development).

We would like to highlight that any publication generated from this survey will present only statistical results by summarizing the raw data and treating the answers anonymously. In other words, no information about the respondents or institution will be presented individually.

This research is part of a doctoral thesis which is being developed at University of Valenciennes and Hainaut-Cambresis and financed by the Brazilian government (Program Science without Borders/CAPES).

The survey is divided into 2 parts (described below) and the estimated time to fill it is 40 minutes.

- Part 1 - Characterization
- Part 2 - Evaluation of the Implementation of methods, techniques, standards and patterns of Human-Computer Interaction and Software Engineering

We really appreciate your help and time with this research.

Best regards,

Taísa Guidini Gonçalves
Káthia Marçal de Oliveira
Christophe Kolski

Part 1 - Characterization

For the characterization, please indicate the items listed below:

<table>
<tr><td rowspan="2">Respondent identification:</td><td colspan="6">Enterprise employee</td><td colspan="8">Consultant of software process capability maturity models</td></tr>
<tr><td colspan="6"></td><td colspan="8"></td></tr>
<tr><td>E-mail</td><td colspan="14"></td></tr>
<tr><td rowspan="2">Degree subject:</td><td colspan="3">Doctor of Science (D.Sc.) or PhD</td><td colspan="3">Master of Science (M.Sc.)</td><td colspan="4">Specialist or MBA degree</td><td colspan="4">Bachelor’s degree</td></tr>
<tr><td colspan="3"></td><td colspan="3"></td><td colspan="4"></td><td colspan="4"></td></tr>
<tr><td rowspan="2">Training area:</td><td colspan="3">Software Engineering</td><td colspan="4">Computer Science</td><td colspan="3">Human-Computer Interaction</td><td colspan="4">Other</td></tr>
<tr><td colspan="3"></td><td colspan="4"></td><td colspan="3"></td><td colspan="4"></td></tr>
<tr><td rowspan="2">Are you an official implementer of the MR-MPS-SW model?</td><td colspan="7">Yes</td><td colspan="7">No</td></tr>
<tr><td colspan="7"></td><td colspan="7"></td></tr>
<tr><td rowspan="3">Are you affiliated to an Implementing Institution (II)?</td><td colspan="7">Yes</td><td colspan="7">No</td></tr>
<tr><td colspan="7"></td><td colspan="7"></td></tr>
<tr><td colspan="7">What?</td><td colspan="7"></td></tr>
<tr><td rowspan="2">Did you take the official CMMI introduction course?</td><td colspan="7">Yes</td><td colspan="7">No</td></tr>
<tr><td colspan="7"></td><td colspan="7"></td></tr>
<tr><td>How many years have you worked in Capability Maturity models implementations?</td><td colspan="14"></td></tr>
<tr><td rowspan="2">Capability Maturity model(s) and maturity level(s) that you have supported implementations:</td><td colspan="6">CMMI-DEV</td><td colspan="8">MR-MPS-SW</td></tr>
<tr><td>5</td><td></td><td>4</td><td></td><td>3</td><td></td><td>A</td><td></td><td>B</td><td></td><td>C</td><td></td><td>D</td><td></td></tr>
<tr><td rowspan="2">Approximately, in how many enterprises and projects have you supported implementations? (for the levels previously selected)</td><td colspan="7">Enterprises</td><td colspan="7">Projects</td></tr>
<tr><td colspan="7"></td><td colspan="7"></td></tr>
</table>

Part 2 - Evaluation of the Implementation of methods, techniques, standards and patterns of Human-Computer Interaction and Software Engineering

The item listed above present several methods, techniques, standards and patterns from Software Engineering (SE) and Human-Computer Interaction (HCI) that can support the implementation of the Processes of MR-MPS-SW or of the Process Areas of CMMI-DEV, according to the literature and experts.

Please, indicate your level of knowledge *(I know)* and level of use *(I used)* to each one of those methods, techniques, standards and patterns when in the implementation of Capability Maturity models in enterprises you worked.

Example of scale of answer to each question

Answers
I Know:
None ------------------------------------ A lot
I Used:
None ------------------------------------ A lot

Question	Process (MR-MPS-SW)	Process Area (CMMI-DEV)	Potential methods, techniques, standards and patterns from Software Engineering (SE)	Potential methods, techniques, standards and patterns from Human Computer-Interaction (HCI)
1	**Requirements Development** DRE1 The customer needs, expectations and restrictions from both, the product and its interfaces, are identified	**Requirements Development** RD SP1.1 Elicit Needs	**Techniques to identify needs** Examples (see References), not limited to: • Brainstorming • Interviews • Questionnaires • Card Sorting • Focus Groups • Field Studies/Observation • Workshops • Protocol Analysis	**Techniques to identify user needs** Examples (see References), not limited to: • Brainstorming • Interviews • Surveys/Questionnaires • Card Sorting • Focus Groups • Field Studies/Observation
2	**Requirements Development** DRE1 The customer needs, expectations and restrictions from both, the product and its interfaces, are identified DRE2 A defined set of customer requirements is specified and prioritized from the needs, expectations, and constraints identified DRE6 Operational concepts and scenarios are developed	**Requirements Development** RD SP1.1 Elicit Needs RD SP1.2 Transform Stakeholder Needs into Customer Requirements RD SP3.1 Establish Operational Concepts and Scenarios	**Techniques to identify requirements** Examples (see References), not limited to: • Scenario • User stories • Storyboards • Task Analysis • Use cases • Quality Function Deployment • FAST (Facilitated Application Specification Techniques): JAD, The	**Techniques to identify user and organizational requirements** Examples (see References), not limited to: • Scenario • User stories • Storyboards • Task Analysis • Persona • Context-of-use analysis • User Profile (Detailed) • Requirements specification templates (e.g.

Question	Process (MR-MPS-SW)	Process Area (CMMI-DEV)	Potential methods, techniques, standards and patterns from Software Engineering (SE)	Potential methods, techniques, standards and patterns from Human Computer-Interaction (HCI)
			Method	VOLERE, IEEE, RESCUE)
3	**Requirements Development** DRE1 The customer needs, expectations and restrictions from both, the product and its interfaces, are identified DRE2 A defined set of customer requirements is specified and prioritized from the needs, expectations, and constraints identified DRE3 A set of functional and non-functional requirements of the product and product components that describe the solution to the problem being solved is defined and maintained from the customer's requirements DRE7 Requirements are analyzed, using defined criteria, to balance stakeholder needs with existing constraints	**Requirements Development** RD SP1.1 Elicit Needs RD SP1.2 Transform Stakeholder Needs into Customer Requirements RD SP2.1 Establish Product and Product Component Requirements RD SP3.3 Analyze Requirements	**Software Modeling** Examples (see References), not limited to: • HTA (Hierarchical Task Analysis) • SADT (Structured Analysis and Design Technique) • Business case analysis • Suitable UML diagrams (see UML diagrams)	**Task Modeling** Examples (see References), not limited to: • HTA (Hierarchical Task Analysis) • SADT (Structured Analysis and Design Technique) • CTT (Concur Task Tree) • K-MAD (Kernel of Model for Activity Description) • GTA (Groupware Task Analysis) • HAMSTERS notation • Task Model Standard (W3C)
4	**Requirements Development** DRE2 A defined set of customer requirements is specified and prioritized from the needs, expectations, and constraints identified DRE3 A set of functional and non-functional requirements of the product and product components that describe the solution to the problem being solved is defined and maintained from the customer's requirements DRE4 The functional and non-functional requirements of each product component are refined, elaborated and allocated **Design and Construction of the Product** PCP3 O produto e/ou componente do produto é projetado e documentado **Validation** VAL3 Critérios e procedimentos para validação dos produtos de trabalho a serem validados são identificados e um ambiente	**Requirements Development** RD SP1.2 Transform Stakeholder Needs into Customer Requirements RD SP2.1 Establish Product and Product Component Requirements RD SP3.2 Establish a Definition of Required Functionality and Quality Attributes **Technical Solution** TS SP2.1 Design the Product or Product Component **Validation** VAL SP1.3 Establish Validation Procedures and Criteria **Verification** VER SP1.3 Establish Verification Procedures and Criteria	**Standards and Guidelines for design** Examples (see References), not limited to: • ISO/IEC 25000 (2014) • ISO/IEC 9126-1 (2001) • Accessibility standards and guidelines (WAI-W3C) • Domain-Specific Standards (Eg. security, critical systems, ...)	**Standards and Guidelines for HCI design** Examples (see References), not limited to: • ISO/IEC 25000 (2014) • ISO/IEC 9126-1 (2001) • ISO 9241-11 (1998) • Ergonomic Criterion (Scapin and Bastien, 1993; Vanderdonckt, 1994) • Accessibility standards and guidelines (WAI-W3C) • Nielsen's Heuristics • Golden Rules of Interface Design

Question	Process (MR-MPS-SW)	Process Area (CMMI-DEV)	Potential methods, techniques, standards and patterns from Software Engineering (SE)	Potential methods, techniques, standards and patterns from Human Computer-Interaction (HCI)
	para validação é estabelecido **Verification** VER3 Critérios e procedimentos para verificação dos produtos de trabalho a serem verificados são identificados e um ambiente para verificação é estabelecido			
5	**Requirements Development** DRE2 A defined set of customer requirements is specified and prioritized from the needs, expectations, and constraints identified DRE8 The requirements are validated	**Requirements Development** RD SP1.2 Transform Stakeholder Needs into Customer Requirements RD SP3.5 Validate Requirements	**Prototype for requirements** Examples (see References), not limited to: • Paper Prototyping/Sketches • Storyboards • Wireframes • Mockups • Wizard of Oz • Video prototyping	**Prototype for HCI requirements** Examples (see References), not limited to: • Paper Prototyping/Sketches • Storyboards • Wireframes • Mockups • Wizard of Oz • Video prototyping
6	**Requirements Development** DRE7 Requirements are analyzed, using defined criteria, to balance stakeholder needs with existing constraints DRE8 The requirements are validated	**Requirements Development** RD SP3.4 Analyze Requirements to Achieve Balance RD SP3.5 Validate Requirements	**Techniques to validate requirements** Examples (see References), not limited to: • Thinking Aloud • Analysis • Simulations • Demonstrations • User Testing (using Prototypes) • Perspective base-reading	**Techniques to validate HCI requirements** Examples (see References), not limited to: • Thinking Aloud • Proto Task (K-MAD) • Task Model Simulator (CTT) • Focus Group for evaluate requirements
7	**Design and Construction of the Product** PCP1 Solution alternatives and selection criteria are developed to meet defined requirements of the product and product components PCP3 The product and/or the product component is designed and documented PCP6 The product components are implemented and verified according to what was designed	**Technical Solution** TS SP1.1 Develop Alternative Solutions and Selection Criteria TS SP2.1 Design the Product or Product Component TS SP3.1 Implement the Design	**Architecture Patterns for SE** Examples (see References), not limited to: • MVC (Model-View-Controller) Model • Service-Oriented Architecture (SOA) • 3-Tier Model • Pipes and Filters • Suitable UML diagrams (see UML diagrams)	**Architecture patterns for HCI** Examples (see References), not limited to: • MVC (Model-View-Controller) Model • Arch Model (Bass et al., 1991) • Language Model • SEEHEIM Model (Pfaff, 1985) • PAC (Presentation-Abstraction-Control) Model • PAC-AMODEUS Model • CAMELEON-RT • Frameworks

Question	Process (MR-MPS-SW)	Process Area (CMMI-DEV)	Potential methods, techniques, standards and patterns from Software Engineering (SE)	Potential methods, techniques, standards and patterns from Human Computer-Interaction (HCI)
8	**Design and Construction of the Product** PCP6 The product components are implemented and verified according to what was designed	**Technical Solution** TS SP3.1 Implement the Design	**Design Patterns for SE** Examples (see References), not limited to: • Design Patterns: Elements of Reusable Object-Oriented Software • GRASP - General Responsibility Assignment Software Patterns • Head First Design Patterns • Patterns of Enterprise Application Architecture	**Design patterns for HCI** Examples (see References), not limited to: • A Pattern Language for Human-Computer Interface Design • A Pattern Approach to Interaction Design • Pattern Languages in Interaction Design: Structure and Organization • Designing interfaces
9	**Design and Construction of the Product** PCP1 Solution alternatives and selection criteria are developed to meet defined requirements of the product and product components	**Technical Solution** TS SP1.1 Develop Alternative Solutions and Selection Criteria	**Interaction modeling for SE** Examples (see References), not limited to: • Suitable UML diagrams (see UML diagrams)	**Techniques for interaction modeling** Examples (see References), not limited to: • MoLIC (Modeling Language for Interaction as Conversation) • UAN (User Action Notation) • TAG (Task-Action Grammar)
10	**Design and Construction of the Product** PCP7 The documentation is identified, developed and made available according to the established standards	**Technical Solution** TS SP3.2 Develop Product Support Documentation	**Techniques for final documentation** Examples (see References), not limited to: • Style manual • ISO/IEC 26514 (2008)	**Techniques for HCI documentation** Examples (see References), not limited to: • Style guide • Architecture for help • Training Program
11	**Design and Construction of the Product** PCP3 The product and/or the product component is designed and documented **Product Integration** ITP1 An integration strategy, consistent with the design and product requirements is developed and maintained for product components **Validation** VAL1 Work products to be validated are identified VAL2 A validation strategy is developed and implemented, establishing schedule, participants involved, methods for validation and any material to be used in the validation **Verification** VER1 Work products to be checked are	**Technical Solution** TS SP2.1 Design the Product or Product Component **Product Integration** PI SP1.1 Establish an Integration Strategy **Validation** VAL SP1.1 Select Products for Validation **Verification** VER SP1.1 Select Work Products for Verification	**Prototype (system versions)** Examples (see References), not limited to: • User interface toolkits • User interface builders • User interface development environments	**Iterative and Evolutionary Prototypes (system versions)** Examples (see References), not limited to: • User interface toolkits • User interface builders • User interface development environments

Question	Process (MR-MPS-SW)	Process Area (CMMI-DEV)	Potential methods, techniques, standards and patterns from Software Engineering (SE)	Potential methods, techniques, standards and patterns from Human Computer-Interaction (HCI)
	identified VER2 Uma estratégia de verificação é desenvolvida e implementada, estabelecendo cronograma, revisores envolvidos, métodos para verificação e qualquer material a ser utilizado na verificação			
12	**Verification** VER1 Work products to be checked are identified VER2 A verification strategy is developed and implemented, establishing schedule, reviewers involved, verification methods and any material to be used in the verification VER3 Criteria and procedures for verifying the work products to be checked are identified and an environment for verification is established VER4 Verification activities, including testing and peer review are carried out VER6 Results of verification activities are analyzed and made available to interested parties	**Verification** VER SP1.1 Select Work Products for Verification VER SP1.2 Establish the Verification Environment VER SP3.1 Perform Verification VER SP3.2 Analyze Verification Results	**Verification methods** Examples (see References), not limited to: • Unit test • Integration test • System test • Acceptance test • Installation test	**Evaluation methods for HCI verification** Examples (see References), not limited to: • Unit test • Integration test • System test • Acceptance test • Installation test
13	**Verification** VER2 A verification strategy is developed and implemented, establishing schedule, reviewers involved, verification methods and any material to be used in the verification VER4 Verification activities, including testing and peer review are carried out VER6 Results of verification activities are analyzed and made available to interested parties	**Verification** VER SP2.1 Prepare for Peer Reviews VER SP2.2 Conduct Peer Reviews VER SP2.3 Analyze Peer Review Data	**Review methods** Examples (see References), not limited to: • Inspections • Structured walkthroughs • Guidelines review • Pair programming • Audits	**Evaluation methods for HCI review** Examples (see References), not limited to: • Semiotic inspection • Formal usability inspection • Consistency inspection • Cognitive walkthrough • Groupware walkthrough • Guidelines review • Metaphors of human thinking (MOT) • Heuristic evaluation
14	**Validation**	**Validation**	**Validation methods**	**Evaluation methods for HCI validation**

Question	Process (MR-MPS-SW)	Process Area (CMMI-DEV)	Potential methods, techniques, standards and patterns from Software Engineering (SE)	Potential methods, techniques, standards and patterns from Human Computer-Interaction (HCI)
	VAL1 Work products to be validated are identified VAL2 A validation strategy is developed and implemented, establishing schedule, participants involved, methods for validation and any material to be used in the validation VAL3 Criteria and procedures for validation of the work products to be validated are identified and an environment for validation is established VAL4 Validation activities are performed to ensure that the product is ready for use in the intended operational environment VAL6 Results of validation activities are analyzed and made available to interested parties	VAL SP1.1 Select Products for Validation VAL SP1.2 Establish the Validation Environment VAL SP2.1 Perform Validation VAL SP2.2 Analyze Validation Results	Examples (see References), not limited to: • Acceptance test with users • Formal review • Tests of products (by end user/stakeholders) • Analyses of product • Functional demonstrations	Examples (see References), not limited to: • Usability testing • Communicability test • Standardized usability questionnaires • Post-experience interviews • User experience evaluation