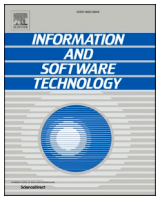
Information and Software Technology 164 (2023) 107324

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof



An empirical experiment of a usability requirements elicitation method to design GUIs based on interviews

Yeshica Isela Ormeño [a], José Ignacio Panach [b,*], Oscar Pastor [c]

[a] Universidad Nacional de San Antonio Abad del Cusco, Perú
[b] Departament d'Informàtica, Escola Tècnica Superior d'Enginyeria, Universitat de València, Avenida de la Universidad s/n, València, Burjassot 46100, Spain
[c] Valencian Research Institute for Artificial Intelligence (VRAIN), Universitat Politècnica de València, Camí de Vera s/n, València, Valencia 46022, Spain

ARTICLE INFO

Keywords:
Usability requirements elicitation
Interviews
Empirical experiment
Guidelines

ABSTRACT

Context: The usability requirements elicitation process is a difficult task that lacks methods to guide and help analysts, who are usually not experts at usability.
Objective: This paper conducts an experiment with two replications to evaluate a method that elicits usability requirements based on structured interviews named UREM versus an unstructured method. The method consists of guided interviews by the analyst using decision trees. The tree is composed of questions and possible answers. Each question appears when there are different possible design alternatives, and each answer represents one of these alternatives. The tree also recommends the alternative that enhances the usability based on existing usability guidelines.
Method: We have conducted an experiment with two replications with 22 and 26 subjects playing two different roles in a within-subjects design. The analysts used a tree to guide the interview and elicit the requirements while the end users had to explain to the analyst the type of system to develop. During the interview, the analyst must design a paper prototype to be validated by the end user. For the analyst, the experiment measures the effectiveness of usability requirements elicitation, the effectiveness of the use of the usability guidelines, the efficiency of the elicitation process, and the satisfaction with the entire elicitation process. For the end user, the experiment measures the satisfaction with the designed prototype at the end of the interview.
Results: UREM yielded significantly better results for the effectiveness in the usability requirements elicitation process and for the effectiveness in the use of usability guidelines when compared to unstructured interviews. The use of UREM did not reduce the analysts' efficiency and both analyst and end user remained the same satisfaction.
Conclusions: Eliciting usability requirements is a difficult task if it is done with unstructured interviews and without usability recommendations.

1. Introduction

Usability is an important quality characteristic of software and is an essential element to be considered in the development of different software systems in order to determine the development's success or failure [1,2]. The ISO 9241–11 [3] standard defines usability requirements as the effectiveness, efficiency, and satisfaction of a user achieving his/her goals in a defined context of use. Similarly, according to the ISO/IEC 25,010 [4] standard, usability is the degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use. Today we live with new and innovative ways of interacting with computers, and this era requires application software that has high usability levels that decrease potential usability difficulties and risks [5]. However, usability requirements are usually ignored during the software development process, especially in the early stages of requirements elicitation. This increases the cost of solving usability problems and affects the quality of final products.

The software engineering and requirements engineering community knows that the process of eliciting the usability requirements of a system is not an easy task and requires a lot of effort. Therefore, methods that help software engineers or systems analysts in the process of eliciting

* Corresponding author.
E-mail address: joigpana@uv.es (J.I. Panach).

https://doi.org/10.1016/j.infsof.2023.107324
Received 16 February 2023; Received in revised form 24 August 2023; Accepted 28 August 2023
Available online 29 August 2023

usability requirements are needed, reducing time and resource costs, and complying with standards or regulations for different domains and platforms. Since usability is a multifaceted concept, there are many usability techniques for performing usability studies. Interviews and prototypes are the most common techniques used to elicit usability requirements, but they must be structured correctly so that they can be defined, measured, and evaluated properly [6]. An analyst that elicits requirements is not usually an expert at usability and needs some guidelines to be able to design usable interfaces.

In order to help analysts design usable systems, in a previous work [7], we proposed the Usability Requirement Elicitation Method (UREM). UREM consists of a decision tree where nodes are questions and answers. The analyst must navigate throughout the tree asking questions to the end user and providing to the end user different answers as possible design alternatives. Questions appear when the analyst has to choose among several design alternatives. Each answer is a design alternative. In order to help in this choice, the tree must also show which alternative optimizes the usability. Each answer of the tree has a description that suggests for which circumstances this design is recommended. Thus, the analyst can recommend a specific option to the end user, but the end user is the one who desires what she/he prefers. The recommendations have been extracted from usability guidelines. The question-answer format of this interview is a way to guide the requirements elicitation process in order to elicit usability requirements. During the interview, the analyst must design a paper prototype with the GUI. The end user must validate this design, proposing any changes that she/he considers optimize usability. Usability requirements is a concept that affects many factors, not only the visible GUI that is the result of the design, but also functionality, learnability, efficiency, etc. [8]. UREM can be used for all the usability requirements whose guidelines can be written in the tree structure as answers or recommendations.

The main contribution of this article is the design and conduction of an empirical experiment to validate UREM with two replications of 22 and 26 subjects respectively. The design includes two treatments: unstructured interviews and UREM. Both treatments are participatory methods to involve the end users and analysts throughout the design process [9]. The experiment is a within-subjects design (repeated measures) where each subject plays the role of analyst or end user in one of both treatments. We defined 24 pairs of subjects from the 48 subjects recruited for both replications. In each of these pairs, roles were swapped during the application of each treatment. The subject that played the role of analyst had to guide the interview in order to elicit the usability requirements and validate these requirements using a paper prototype. The subject that played the role of end user had to explain to the analyst the type of system they needed and the usability requirements that had to be included. We used two different problems in order to avoid the carryover effect between treatments. For the analyst, the response variables were: the effectiveness of the usability requirements successfully elicited; the effectiveness of the usability guidelines properly applied in the prototype; the efficiency in the requirements elicitation process; and the satisfaction during the whole elicitation process. For the end user, the response variable was the satisfaction with the designed GUI.

The results yielded two significant differences between UREM and the unstructured interview: (1) UREM was more effective in the usability requirements elicitation; (2) UREM was more effective in the application of the usability guidelines to improve usability. The lack of significant differences in efficiency using the two elicitation methods means that, even though UREM might be considered more cumbersome at first glance, its use did not increase the time required to design the GUI. The improvement in effectiveness using UREM does not lead to an improvement in the satisfaction of the analyst and the end user. An analysis of these results is discussed in the article.

This article is organized as follows. Section 2 describes the related works. Section 3 explains UREM and the unstructured interview in detail. Section 4 justifies the experimental design. Section 5 presents the statistical results. Section 6 discusses and interprets the results. Finally, Section 7 presents the conclusions and future work.

2. Related works

In this section, we describe works that are related to usability requirements elicitation and their empirical validations. We conducted a Targeted Literature Review (TLR) [10], which is a non-systematic, in-depth, and informative literature review aimed at keeping only the significant references in order to maximize rigorousness while minimizing selection bias. For this purpose, the semantic question about usability requirements elicitation is translated into the following syntactical queries used as a search string: ("usability requirements" AND ("method" OR "methodology" OR "model") AND ("experiment" OR "case study")). This search string was applied to the title, keyword, and abstract of the Scopus digital library, ACM Digital Library, Web of Science, and IEEExplore in May 2023.

As exclusion criteria, we have: (1) tutorial papers; (2) papers that do not deal strictly with usability requirement elicitation; (3) papers that do not report the results of the experiment; (4) papers without methods or models; and (5) paper without any experimental design carried out. As inclusion criteria, we have: (1) papers that describe the developing methodology in usability requirement elicitation; (2) papers that describe how they evaluated or analyzed developing methodology; and (3) papers that include a case study and/or guidelines for the elicitation process. The search string returned 22 papers from the Scopus digital library and 23 papers from the IEEExplore digital library. After applying the exclusion and inclusion criteria to the title and abstract, and gathering the papers from both outlets and search string, we finally analyzed the content of 15 papers, which we describe below. The references resulting from these searches were classified into four categories, which are discussed further in the following subsections. This classification aims to identify the papers that have proposed requirements elicitation methods for both specific contexts and non-specific contexts, papers that use usability guidelines in their proposals of requirements elicitation, and papers that validate empirically a requirements elicitation method. These four types of papers cover the target of our contribution: an empirical validation of a requirements elicitation method of non-functional requirements based on usability guidelines. Table 1 shows a summary of all of these works, comparing the proposed method, metrics, tools, and techniques.

2.1. Usability requirements elicitation for specific contexts

This subsection describes the works whose processes have been developed to be carried out for a specific problem domain, to test the method in an existing application, or to understand/complement it. Gunduz and Pathan [11] describe usability problems found in touchscreen mobile flight-booking applications and suggest solutions to eliminate such problems. A qualitative research approach is used for usability analysis. They considered users' actions and reactions towards the application for their specific context and collected their opinions with regard to efficiency, user satisfaction, and adoption of the application. The case study was carried out on a Turkish Airlines' commercial mobile flight-booking application where 20 interviewees from different countries were randomly selected from novice and advanced users. They use questionnaires and interviews during the practical investigation.

Troyer and Janssens [12] present a Feature Modeling method which is a variability modeling technique used in Software Product Lines. It has a twofold approach: one to unlock available information on requirements elicitation and the other to provide a mechanism for guiding the stakeholders (non-computing people) through the requirements elicitation process. The feature model is supported in a tablet app that provides explanations for different usability issues, possible design options and alternatives, and the impact of the choices. Two case studies based on games and e-shop web applications were conducted using evaluation sessions that focused on the usability of the tool,

Table 1
Overview of state of the related works.

Scope	Authors	Methods	Metrics	Tools	Techniques
Usability Requirement Elicitation from Specific Context over Existing Systems	Gunduz and Pathan [11]	Qualitative research approach	Easiness, efficiency, user satisfaction, and adoption of the application.		Questionnaire Interview sessions Likert scale questions
	Troyer and Janssens [12]	Feature Modeling	Effectiveness of the Guinea maps tool. Completeness of the template. Relevance of the template. Learnability of the app. Easy of use Good overview	Guidemap tool	Usability questionnaire Interview Templates workshops
	Fahey et al. [13]	Business Process Modelling (BPM)	Usability testing Optimize time management of users Facilitate work practice change		Ethnographic analysis Workshop and multi-stage Delphi interview Iterative prototyping Process maps Screenshots
	Temper et al. [14]	Vaguely Quantified Nearest Neighbor Fuzzy model Rough Set Theory (RST)	Feasibility, trust score, Equal Error Rate	Fuzzy-Weka	Particle Swarm Optimization Fuzzy rules
	rocha et al. [15]	behavior-driven development based on user stories	Adherence to a template to include behaviors		User stories
Usability Requirement Elicitation from Others General Methods with Unexisting Systems	De Carvalho et al. [16]	Functional Resonance Analysis Method (FRAM and MacKnight) and BPMN	Average performance, completeness Likert Scale		Ethnography, Questionnaires,
	Nhavoto [17]	Design science research methodology	Functionality Completeness Consistency Accuracy Performance Reliability and Usability	Web client for the Web-SMS tool	Brainstorming Focus group meetings Algorithm
	Elias [18]	Ontology, software agents, SPARQL rules usability methods	Standardization of Pedagogical Usability Standardization of Technical Usability Moodle graphical report		Questionnaires Usability techniques Checklists
	Yuan, X. and X. Zhang [19]	Ontology model	Learnability Efficiency Reliability Satisfaction		Rules Algorithm
	Abad et al. [20]	LPP (Loud Paper Prototyping) Silent Paper Prototyping (SPP) No Paper Prototyping (NPP)	Learnability Navigation helpful Improvements Understandability		Latent Dirichlet Allocation-LDA NVivo [11] tool
Using Guidelines	Márquez and Taramasco [21]	D&I framework	Perceived usefulness Perceived ease of use and user control Health-ITUES questionnaire		Interviews Requirement elicitation guidelines Working sessions
	Abdallah et al. [23]	eXtreme Scenario Based Design Quality in Use Integrated Model Usability Critical Parameters Workshop	Learnability Efficiency Effectiveness Likert scale		Scenarios Workshops (SUS) questionnaires
Empirical validations	Vitiello et al. [24]	The empowerment-driven (UX) Requirements Engineering method	Index of Self Efficacy (ISE), the Index of Knowledge & Skills (IKS), the Index of Personal Control (IPC), and the Index of Motivation (IMOT). Efficacy and efficiency	Sedato prototype	Interview, Questionnaires
	Tanikawa et al. [25]	Process support method	Validity of the output requirements and the effectiveness		Entry form check item in-house guidelines for usability improvement [Hiramatsu]
	Abad et al. [26]	Wizard-of-Oz (WOz) User Reviews	Efficacy Effective in capturing NFR Clarifying existing FR	Statistical methods Saturate web-based coding tool	Storyboarding Low-fidelity prototyping Meeting Github repository Latent Dirichlet Allocation (LDA) algorithm topic models package in R
	Peruzzini and Germani [27]	User-Centered Design (UCD) Delphi methodology Design Structure Matrix (DSM) Quality Functional Deployment	Satisfaction Usable solutions Correlation between users' needs and system funcionalities Positive effect on efficiency		Workshops Focus groups Brainstorming Questionnaires

brainstorming sessions, and templates done by requirements engineering experts.

Fahey et al. [13] describe the value of a design approach to elicit user requirements by performing business process modeling (BPM) and the elicitation and modeling of user requirements through the work of the users. It presents a case study of how an outpatient Electronic Patient Record (EPR) system was successfully implemented in the Epilepsy Unit of Beaumont Hospital, Dublin. The determination of functional (FR) and non-functional user requirements (NFR) was realized through a series of traditional requirements elicitation techniques such as workshops and multi-stage Delphi interviews. Process maps were drawn up and confirmed with end users, and new prototypes were developed on paper and on mock-up screens. They conclude that the more time spent on usability issues in the early stages of system development, the more likely a system will undergo a successful implementation with minimal disruption of the necessary services.

Temper et al. [14] introduce an efficient continuous biometric authentication technique using touchscreen gestures and related posture information that is based on a Vaguely Quantified Nearest Neighbor classifier combined with a scoring model and fuzzy classifier. A bank app prototype implemented on a Google Nexus 4 mobile phone was developed to evaluate the security and usability requirements. The evaluation was conducted with 22 volunteers based on a trust score which was used as an indicator to verify whether or not the person that enters information within the app is a legitimate user. The calculation of the score is based on touchscreen gestures and posture information. The results depicted how the trust score evolves over time. The initial results showed the applicability of behavioral biometrics as an additional security mechanism on mobile phones.

Rocha et al. [15] have defined a method to elicit requirements based on structured interviews using user stories. These user stories are used in a behavior-driven development context with templates for guiding the writing of such stories. The approach can be helpful to ensure that consistent information about the requirements is provided. User stories written using terms of an ontology describing events, behaviors, and user interface elements can be used to promote consistency of requirements. Moreover, user stories can be used for testing the automation of diverse types of artefacts, such as task models, low-fidelity prototypes or final implementation of the interactive system. The approach was validated in a case study with potential product owners in a research institute, where subjects had to write their own user stories to describe a feature they are used to performing.

The above research works were performed for a specific context. the work of Troyer and Janssens [12] is for Software Product Lines, the work of Fahey et al. [13] is for BPM, the work of Temper et al. [14] is for touchscreen gestures, and the work of Rocha et al. [15] is for behavior-driven development. Each method seeks to elicit requirements and to find solutions for usability issues in its own way. The techniques that are most widely used to support the methods are unstructured interviews, brainstorming, focus groups, and questionnaires with Likert scale, but there are also proposals such as the work of Rocha et al. that propose a structured method.

2.2. Usability requirements elicitation for non-specific contexts

This subsection describes the works to elicit requirements that have been performed from a non-specific context, i.e., the method can be applied in different domains. De Carvalho et al. [1,16] evaluate the possibility of discovering usability requirements from information in the Functional Resonance Analysis Method (FRAM) in the health field. The methodology follows these steps: (1) identification of the context; (2) identification of problems and difficulties in the execution of a task; (3) definition of solutions; and (4) definition of software requirements. Two experiments were conducted. The first one was a patient selection process with BPMN notation, and the second one was a patient selection process through a FRAM model. The results showed that the FRAM method used for complex systems yields more requirements, especially usability requirements. There was also superiority in the average performance related to the number of requirements per activity/function, the average in functional requirements, and the quality (availability, understanding, clarity, completeness) of the elicited requirements.

Nhavoto et al. [17] presents an integrated mobile phone text-messaging system that is used to follow up on Human Immunodeficiency Virus (HIV) and Tuberculosis (TB) patients. The study focuses on three key activities: eliciting the requirements, design of the GUIs, and implementation of a prototype named SMSaúde to facilitate communication between patients and the healthcare systems. Testing and evaluation of the SMSaúde system were done using seven quality criteria (functionality, completeness, consistency, accuracy, performance, reliability, and usability) and six different requirements (data collection, telecommunication costs, privacy, data security, the content of text messages, connectivity, and system scalability). The artifact was improved interactively and incrementally. During the design and development process, a broad set of usability requirements was identified in two brainstorming design sessions. They plan to perform an evaluation of the system, including a satisfaction survey of the health professionals and patients.

Elias et al. [18] presents a semiautomatic validation system to improve usability in Computer Support Collaborative Learning (CSCL) environments. It uses an ontology to represent usability knowledge and software agents to automate the process. This system uses usability methods and techniques to create SPARQL rules to deal with usability issues. The rules were performed by the interaction among agents, using questionnaires to know the users' opinion about usability. A case study in a real collaborative learning environment based on Moodle at Federal University of Alagoas - Brazil was described to present the advantages of using the proposed system. As a result, the system provides graphical reports and checklists to help the administrator improve the usability of the CSCL environment.

Yuan and Zhang [19] present an ontology model to represent the knowledge of common and variable software assets for interactive requirements elicitation. The instances of an abstract model help the interactive software customization system to communicate with software clients via dialog in natural language. In order to demonstrate how it works and to provide evidence of its usability, they include a case study of an online book shopping system with experienced and non-experienced software clients. The system retrieves product information from the ontology model and presents software requirements in utterances as slots for users to fill in. Learnability, efficiency, reliability, and satisfaction, along with several other measurements, were evaluated. The proposed approach was capable of not only eliciting requirements but also automatically converting client-picked requirements into service descriptions in Web Ontology Language for the production of customized software systems.

Abad et al. [20] study the impact of Loud Paper Prototyping (LPP) on requirements elicitation. They compare this technique with several variations of Silent Paper Prototyping (SPP) such as traditional Woz, sketching, and storyboard. Furthermore, they present a comparison between LPP and elicitation meetings alone as well as paper prototyping versus No Paper Prototyping (NPP). Two research questions were defined: (1) How does paper prototyping help in capturing mobile App requirements?; and (2) Does LPP affect the type of requirements extracted during requirements elicitation? These questions were analyzed in a case study with two mobile application developments teams. The results showed that (1) SPP is more efficient in capturing NFRs than NPP; and (2) LPP is more useful in adding new NFRs and moving/modifying existing ones. Among the techniques reviewed, most teams found LPP to be the most useful approach for managing mobile application requirements.

All of these research works deal with methods, models, and techniques that are oriented to information management in order to elicit requirements during the design and development process. The elicited

usability requirements were generally obtained from brainstorming sessions, interview sessions, and questionnaires. Some works show a formal analysis of data to improve the elicitation of usability requirements by algorithms. The selected case studies were adapted to methods or models in order to demonstrate their effectiveness. In most of the previous works, the usability requirements are studied together with functional requirements and other NFRs in the elicitation process. In other words, the methods are not exclusive to the elicitation of usability requirements.

2.3. Using guidelines

This subsection describes the papers whose elicitation method depend on usability guidelines. Márquez and Taramasco [21] present a methodology that uses dissemination and implementation (D&I) strategies to recommend requirements elicitation guidelines [22] for eliciting requirements in health systems. The D&I framework considers two phases: The first phase aims to identify the goals of the system. The second phase is about the implementation strategies and requirements elicitation guidelines represented in a model and a multidimensional catalog based on a source of knowledge that generates a set of guidelines for the elicitation of requirements to be evaluated by IT professionals. Working sessions were conducted by IT professionals and clinicians to ensure that each strategy/guideline relationship was fully explained. To assess the impact of using the D&I framework, the authors present a real clinical software case study of the main software component of SIGICAM related to clinical priorities that were developed using the D&I framework. The analyzed variables were: impact, perceived usefulness, perceived ease of use, and user control. The results show an acceptable level of usability with approximately 72% approval.

Abdallah et al. [23] introduced an enhancement of an eXtreme Scenario Based Design (XSBD) process named Quatified eXtreme Scenario Based Design (QXSBD) to quantify usability. QXSBD complements XSBD with a set of usability metrics that need to be assessed in an agile process based on usability guidelines. This framework uses the Usability Critical Parameters Workshop (UCPW) to identify usability scenarios from stakeholders (usability engineers, developers, end users, and customers) and Quality in Use Integrated Model (QUIM) procedures to assign required values. The UCPW provides engineering practices defining the usability requirements and design goals. In order to demonstrate the feasibility of the QXSBD, an interactive system, Customer Request Project, was implemented where efficiency, effectiveness, productivity, and learnability were selected as usability critical parameters. After applying the QXSBD process, the usability defect rate was reduced by 30%. The team questionnaire and end user questionnaire show that UCPW provides practical tactics and guidelines to implement usability scenarios on the process cycle, achieving better user satisfaction.

In the previous frameworks, requirement elicitation guidelines are based on a source of knowledge obtained from workshops sessions conducted by usability experts and the IT team. The carrying out of these workshops increases the need to dedicate more time to the process of eliciting, redefining, and updating usability parameters. In addition, the continuous participation of usability specialists is needed to clarify and explain the reasons and effects of the use of these parameters.

2.4. Empirical validations

This subsection describes the empirical evaluations of requirements elicitation methods. There are proposals where the evaluation of the method is unstructured, i.e., formal mechanisms are not used. Vitiello et al. [24] proposed a methodology to extract UX requirements. It is a transformative process that starts from a contextual investigation in order to understand users, their behavior (decision making, self-management, communication, and engagement), and capacities (self-efficacy, knowledge & skills, personal control, and motivation), which are expressed in terms of human needs. The author tested the methodology on a case study of polypharmacy management Interviews. The questionnaires give an initial measure of user empowerment perception represented with empowerment perception ratings such as the Index of Self Efficacy (ISE), the Index of Knowledge & Skills (IKS), the Index of Personal Control (IPC), and the Index of Motivation (IMOT). The results showed that an improvement in the described capacity indicators was achieved.

Tanikawa et al. [25] present a method that focuses on clarifying the needs related to the customer's usability (clarification of customer needs) and the matching of these needs with the system design (conformity between needs and design). The approach consists of defining the activities (tasks and procedures) that are needed to support those needs. An entry form is used to specify target tasks of a system, identify representative users, and describe the works they are in charge of in each task. They also developed check items for specifying the characteristics of the users and tasks of the target system based on in-house guidelines for usability improvement [28]. As a result, the needs and requirements generated by the support method were almost equivalent to those extracted with the work of the experts. Positive effects on efficiency and quality improvement of activities were reported, including a reduction of man-hours for preparation of customers interviews and requirements elicitation.

Abad et al. [26] conducted two studies to compare the role of early usability requirements specification and app reviews. The evaluation focuses on how Wizard-of-Oz (Woz) technique can be used to elicit usability requirements. The first study was about the role of Woz in requirement elicitation activity with the use of storyboarding, low-fidelity prototyping, and meetings between the development team and the client. The second study was related to comparing the role of user review analysis and Woz in eliciting and defining mobile app requirements. It was conducted using 40 mobile apps that are available on Google Play. The results showed that while user reviews are a powerful tool for capturing FRs, there were reports of bugs in several app categories. The authors conclude that Woz is effective in capturing usability requirements and clarifying existing FRs.

Peruzzini and Germani [27] propose a new model to design assistive ICT-platforms including smart products and services to support active aging for elderly and frail people by adopting a user-centered approach to define an interoperable architecture that integrates different types of smart objects. The approach aims to deal with three limitations of existing ambient assisted living systems: low system usability, poor acceptance by users, and lack of personalization. As a result, they obtained a highly usable and flexible platform that is designed according to the specific needs of their direct users with high user satisfaction, usable solutions, user-friendly products, and services with high-level functions integrating data from completely different contexts. Techniques such as interviews, questionnaires, focus groups, and brainstorming were used to conduct the process. Positive effects on efficiency and quality improvement of activities were reported, including a reduction of man-hours for preparing customers interviews and for extracting evidence-based requirements.

Most related works are based on interviews and questionnaires, but none include usability recommendations to guide the end user in the different GUI designs. Moreover, the proposed techniques based on interviews are usually unstructured, so, in the end, how the interview is conducted depends on the interviewer's skills. UREM was proposed as an attempt to cover this gap, proposing a structured interview that is specific for usability requirements. The contribution of this article is the validation of UREM based on effectiveness, efficiency, and satisfaction. These three metrics are the most commonly used in the previous works to validate requirements elicitation methods.

3. Usability requirements elicitation process

This section describes the two methods used to elicit usability

requirements that we analyze in our experiment. The first method uses unstructured interviews and the second method is UREM [7], which uses structured interviews based on usability guidelines and interface design guidelines by means of a tree structure to minimize the cognitive effort. Note that both methods are participatory methods [9] with the end user. The difference lies in the fact that UREM utilizes a flow for requesting input from the end user and provides usability recommendations. Below, we describe both methods in detail.

3.1. The unstructured requirements elicitation method

The unstructured method [29] consists in eliciting usability requirements in an unstructured way, without any guideline or tool to support the process. These are the steps of the method:

- The process begins with an interview between the analyst and the end user. The analyst must ask to the end user how she/he prefers the GUI. There is no guide for what questions must be asked, what design alternatives are possible, and which design alternative optimizes the usability. The analyst organizes the questions as she/he prefers.
- During the interview, the analyst draws a paper prototype of the GUI described by the end user that best fulfills the elicited requirements.
- During this process, the end user can suggest any changes after seeing the results of the prototype. Thus, the analyst can evolve the prototype during the interview until the end user is completely satisfied with the result and considers that the proposed solution fulfills the GUI requirements.

At the end of the session, we have the paper prototypes of all of the GUI that fulfill the usability requirements from the point of view of the end user.

3.2. The usability requirements elicitation method (UREM)

This section presents a summary of eliciting usability requirements proposed by UREM. UREM is a structured and general purpose method for designing GUIs compliant with usability guidelines, that supports the analyst during usability requirements elicitation. To do this, a tree structure is built by a usability expert based on user interface design guidelines and usability guidelines to be executed in the process of eliciting usability requirements. The tree is composed of four elements: questions, answers, groups of questions, and designs. Fig. 1 shows a general schema of the tree structure used by UREM.

We describe each element of the tree as follows.

- -Question (Qi) is defined based on UI design guidelines that are represented in different design alternatives for GUI components. The design guidelines present diverse design alternatives for GUI components (e.g. menu). In order to ask the end-user which alternative she/he prefers, we have defined a question when alternatives to design appear. For example, when we are designing a selectable task, we can ask about how to show it. A possible question is ""Which UI component is used to show selectable tasks?"
- Answer (Ai) is composed of exclusive alternatives for each question based on GUI design guidelines, where the analyst selects which one best fits the user's requirement. These options are presented to the analyst in such a way that she/he can choose which one best fits user's requirements. For each question, some answers are recommended based on usability guidelines. These recommendations aim to help the end user choose the best answers. They are not mandatory; the end user can accept the recommendations or reject them. When answers are shown to the analyst, we will show which answers are recommended by usability guidelines. Possible answers can be yes/no or the choice of one item from a list. For example, the answers to the question "Which UI component is used to show selectable tasks?" can be: RadioButtons, Textfields, CheckBoxes or Slider. According to usability guidelines, a RadioButton is used for a persistent single-choice list.
- Group of Questions (GQi) are created since some branches of the tree structure are not mutually exclusive (the end user should be asked all of the questions). This type of branch is represented by a group of questions that gathers several questions that are grouped by a design characteristic. For example, the question "Which UI component is used to show selectable tasks?" can be gathered with other questions that ask about Selection Dialogues, such as "Where is the action button located?", "Where is the dialog box located?", and "Where is the positive action on a button located?". All these questions have in common that deal with how selection dialogues are displayed, and all of them are gathered in the same Group of Questions.
- Designs (Di) are the interface designs reached at the end of the tree structure (they are the leaves of the tree). The tree structure is navigated from the root to the leaves. When the analyst reaches a leaf in the tree, a design has been obtained. The final design of the whole system is the set of leaves in the tree that the analyst has reached. More details can be found in [7]. For example, a design can be a selection dialog with radio buttons, where each item shows an enumerated data.
- The tree structure is built by an expert in interface design and usability. This expert must have enough knowledge to specify design alternatives as questions and answers, as well as to specify the usability guidelines as recommended answers. Once the tree is completed, the analysts can use it an unlimited number of times to elicit usability requirements in several projects. The analysts that use the tree structure do not need knowledge of usability or design since all this information is represented in the tree structure. In order to interview the client to elicit usability requirements, the analyst starts to navigate from the root of the tree, and asks the questions to the end user during the interview. The analyst asks the questions according

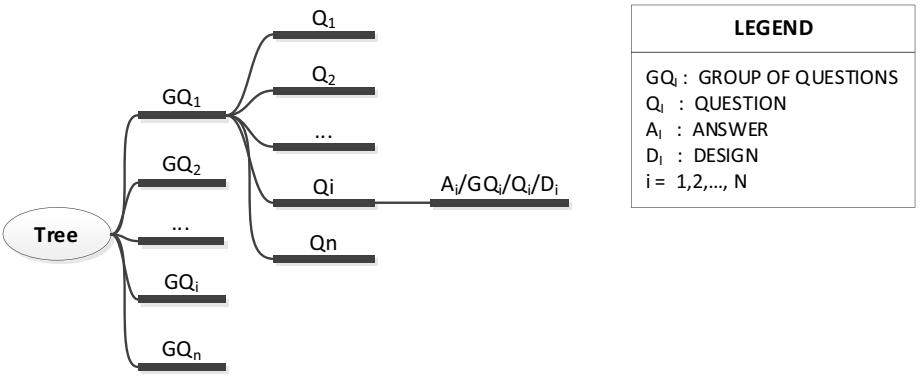


Fig. 1. General representation of the tree structure.

to their sequence in the tree, from the root to the leaves. The analyst only navigates through the branch of the answer selected by the end user. When the analyst reaches a branch with a group of questions, all of the questions must be answered. Only the analyst can continue with the next question if the flow has reached a leaf and, then continues with the next question in the group of questions. The possible navigation between two nodes of the tree structure can be: 1) from a group of questions to a single question or to another group of questions (GQi→ Qi / GQi); 2) from a question to an answer (Qi →Ai); 3) from an answer to a question, to a group of questions, or to a design (Ai → Qi / GQi /Di).

The process of eliciting usability requirements is supported by a tool (hci.dsic.upv.es/urem) that supports the creation and navigation of several trees. The analyst uses the tool to perform the elicitation using interview eliciting. The result after navigating the decision tree with UREM can be seen as a design rationale [30,31]; following the flow of the interview we have the report that explains why a system has been designed the way it is. GUI designs must be manually drawn by the analyst.

3.2.1. An illustrative example of working with UREM

This section presents a short and illustrative example of how to deal with UREM to develop a GUI design for a medical system starting from a set of usability requirements and using the usability guidelines represented in the tree structure. The example focuses on the usability requirements that are related to data entry forms (Fig. 2). All of the entire process is performed in an interview between the end user and the analyst. The first question that the analyst asks the end user is "Should textfields have selectable options"? This question has two possible answers. "yes" or "no". The recommended option is "yes". If the end user opts for "yes", the next question that the analyst asks is "In which component are the options displayed?" There are four possible answers: Dropdown menu (recommended option); Emergent popup, Radiobuttons; Checkboxes. Each one of these options is a leaf in the tree, so it involves a specific design (Table 2). If the end user opts for the recommendation and chooses the answer "Dropdown menu", we have reached design D1. Below, the flow continues with the question "Should textfields have a label?". This question has two possible answers: "yes" or "no". The answer "yes" is recommended based on usability guidelines. If the end user opts for the recommendation and chooses the answer "yes", we have reached design D5 (Table 2). Note that D1 refers to the items that compose the textfield, while D5 refers to the label of the textfield.

4. Experiment definition and planning

In this section, we describe the experiment design according to Juristo and Moreno [32].

4.1. Goal

The main goal of this experiment is to compare the use of a structured method (named UREM) for interviewing the end user in order to elicit usability requirements with the use of unstructured interviews for the purpose of studying the pros and cons of UREM in the GUI design. The experiment is conducted from the perspective of researchers and practitioners who are interested in investigating how useful a structured interview method is compared to an unstructured interview method in eliciting usability requirements.

4.2. Research questions and hypothesis formulation

Our empirical study is based on the concept of quality, which is defined in terms of effectiveness, efficiency, and satisfaction (ISO 25,010) [4]. The concept of quality is different depending on the role of the subjects that participate in the validation (as analyst or end user). From the point of view of the analyst, we aim to study whether the requirements elicitation method affects the elicitation process. This means that we need research questions to analyze the effectiveness, efficiency, and satisfaction of the process of usability requirements elicitation. From the point of view of the end user, quality refers to how satisfied the end user is with the designed GUI. Both perspectives of quality are represented in the research questions. Note that the experiment uses a tree structure previously existing. The role of expert in interface design and usability that builds the tree structure of UREM is played by one experimenter. The study of how the tree is built is out of scope of the current analysis. While the construction of the tree structure is done once, its use is unlimited, which leads to focus the experiment on the use of the tree structure instead of its construction. In the experiment, the construction of the tree structure required two hours, including the time to study the design alternatives to be specified as answers, the usability guidelines to be identified as recommendations, and the specification of all this information in the UREM tool. The experimenter who built the tree is an expert in interface design and usability that has been evaluating usability in systems for more than ten years.

The research questions used in our validation are described as follows:

RQ1: Effectiveness is defined in ISO/IEC-25,010 as "the degree to which specified users can achieve specified goals with accuracy and

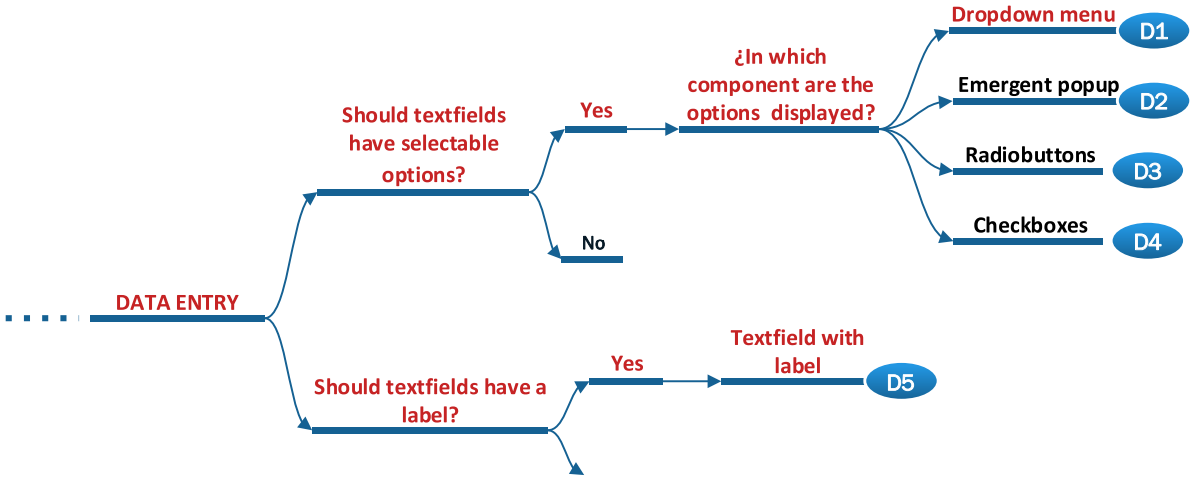


Fig. 2. Illustrative example of usability elicitation.

Table 2
GUI designs for each leaf of the tree.

DESIGNS	GUI DESIGNS
D1	
D2	
D3	
D4	
D5	

completeness in a specified context of use". *Effectiveness in use* is applied in two contexts: elicited usability requirements (RQ1r) and guidelines recommendations (RQ1g).

RQ1r: *Is **analyst effectiveness** to elicit usability requirements affected by the elicitation method?*

We operationalize effectiveness as the percentage of usability requirements satisfied by the analyst. The null hypothesis tested to address this research question is: *H01r: The analyst effectiveness using UREM is similar to that of using unstructured interviews.*

RQ1g: Is **analyst effectiveness** to apply usability guidelines affected by the elicitation method?

We operationalize effectiveness as the percentage of usability recommendations that the designed GUI prototype includes. The null hypothesis tested to address this research question is: *H01g: The analyst effectiveness using usability guidelines in UREM is similar to that of using unstructured interviews.*

RQ2: Efficiency is defined in ISO/IEC-25,010 as "the degree to which specified users expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use". *Efficiency* is studied based on usability requirements (RQ2r).

RQ2r: *Is **analyst efficiency** affected by the usability requirements elicitation method?*

We measure analyst efficiency as the ratio percentage of usability requirements successfully elicited by the time spent to elicit the usability requirements. The null hypothesis tested to address this research question is: *H02r: The analyst efficiency using UREM is similar to that of using unstructured interviews.*

RQ3: Satisfaction is defined in ISO/IEC-25,010 as "the degree to which users are satisfied in a specified context of use". *Satisfaction* is analyzed from two perspectives: analyst satisfaction (RQ3a) and end user satisfaction (RQ3e), since the satisfaction of the analysts who design interfaces may be different from the satisfaction of the end users that will use the interfaces.

RQ3a: *Is **analyst satisfaction** affected by the usability requirements elicitation method?*

We measure analyst satisfaction as the level of contentment of the analysts during the usability requirements elicitation. The null hypothesis tested to address this research question is: *H03a: The analyst satisfaction using UREM is similar to that of using unstructured interviews.*

RQ3e: *Is **end user*** satisfaction *affected by the usability requirements elicitation method?*

We measure end user satisfaction as the level of contentment of the end-user with the designed prototype as a result of the process of requirements elicitation. The null hypothesis tested to address this research question is: *H03e: The end user satisfaction using UREM is similar to that of using unstructured interviews.*

4.3. Factors and treatments

We now define factors and their levels to operationalize the reason for our experiment construct. Factors are variables whose effect on the response variables we want to understand [34]. Treatments are the factor alternatives that help us answer the questions of the research hypotheses.

The experiment studies one factor: the usability requirements elicitation method with unstructured interviews (T1) and UREM (T2), where T1 is referred to as the control treatment. Table 3 shows the description of the factor and its two treatments.

In the first treatment (T1), the analysts conduct the elicitation process using interviews without any structure. This means that the analysts can ask any question regarding the GUI design. Moreover, even though the subjects playing the role of analysts know usability guidelines, there is no recommendation system to suggest a specific design for enhancing usability (as described in Section 3.1).

In the second treatment (T2), the analysts use UREM as a method to elicit usability requirements. The analysts must follow a question-answer format based on the different alternatives specified in a decision tree that is defined in advanced. This decision tree also suggests which design alternative optimizes the usability based on usability guidelines. The details of this treatment are described in Section 3.2

4.4. Response variables and metrics

Response variables are the values that are measured in the experiment in order to study how the factors influence these variables [32]. Below, we define a response variable for each research question (summary in Table 4).

For **RQ1,** Effectiveness is the response variable. This response variable was divided into **RQ1r** to measure the effectiveness of eliciting usability requirements and **RQ1g** to measure the effectiveness of the usability recommendations provided by the guidelines. The metric for RQ1r is calculated as the percentage of usability requirements that are satisfied by the analyst in the GUI prototype built at the end of the interview. For each experimental problem, there is a list of usability requirements that the designed GUI in a prototype must include at the end of the interview. This list is called experimenters' solution since it is defined by the experimenters (in this case, the authors of the article). Possible values for Effectiveness fluctuate from 0% (no usability requirement of the experimenters' solution appears in the designed GUI) to 100% (all of the usability requirements of the experimenters' solution appear in the designed GUI). The metric for RQ1g is calculated as the percentage of designs reached following the tree structure that fits the recommendations provided by the usability guidelines. Possible values fluctuate from 0% (there is no design that agrees with any usability guidelines) to 100% (all of the designs agree with the usability guidelines).

For **RQ2r**, Efficiency is the response variable. This response variable is measured as the ratio percentage of usability requirements successfully elicited by time spent by the analyst eliciting the usability requirements and drawing the GUI prototype. The time is measured in minutes. The larger efficiency, the better the efficiency.

For **RQ3**, *Satisfaction* is the response variable. This response variable was divided into **RQ3a** to measure the analyst's satisfaction and **RQ3e** to measure the end user's satisfaction. RQ3a was measured using the MAM questionnaire developed by Moody [36]. Moody defined a framework (based on the work by Lindland et al. [37].) to measure satisfaction in terms of Perceived Usefulness (PU), Perceived Ease of Use (PEOU), and Intention to Use (ITU). This framework has been previously validated and is widely used [38]. Based on [36], we defined eight questions to measure PU, five questions to measure PEOU, and two questions to measure ITU. The questionnaire is based on a 5-point Likert questionnaire with five possible answers: "Strongly Disagree", "Disagree", "Undecided", "Agree" and "Strongly Agree". RQ3e is based on the Computer

Table 3
Description of the factor and treatments.

Factor	Treatment	Description
Usability Requirements Elicitation Method	T1: unstructured interviews	Experimental subjects elicit usability requirements through unstructured interviews.
	T2: UREM	Experimental subjects elicit usability requirements through UREM

Table 4
Response variables.

Response Variables	Metrics	Definition	Research Questions
Effectiveness for usability requirements elicitation	Percentage of usability requirements successfully elicited.	Percentage (between 0% and 100%) of the usability requirements included in the GUI prototype after the interview that match the usability requirements of the experimenters' solution.	RQ1r
Effectiveness of usability guidelines	Percentage of usability guidelines used correctly on usability requirement elicitation	The number of usability guidelines used correctly divided by the total number of usability guidelines.	RQ1g
Efficiency for usability requirements elicitation	Percentage of usability requirements successfully elicited /Time spent to complete the usability requirement elicitation process	Time is the amount of minutes that the analyst requires to elicit usability requirements and design the GUI prototype.	RQ2r
Analyst's Satisfaction	Perceived usefulness (PU),	The addition of the questions that ask for PU on a Likert scale	RQ3a
	Perceived ease of use (PEOU)	The addition of the questions that ask for PEOU on a Likert scale	
	Intention to use (ITU)	The addition of the questions that ask for ITU on a Likert scale	
End user's Satisfaction	Computer System Usability Questionnaire (CSUQ)	The addition of the questions of the CSUQ on a Likert scale	RQ3e
	Satisfaction with analyst's recommendations	One extra question in the CSUQ to ask about the usefulness of the recommendations	

System Usability Questionnaire (CSUQ) [33], which is a 5-point Likert questionnaire that asks about the satisfaction of the end user with the GUI. We have extended this questionnaire with a specific statement to evaluate whether or not the recommendation system was useful: "Are analyst' recommendations useful to improve the usability of the system?". Table 5 shows a summary of the research questions, hypotheses, response variables, and metrics used to test these hypotheses.

4.5. Experimental subjects

The subjects participating in the experiment were undergraduate students in computer science from the Universidad Nacional de San Antonio Abad del Cusco (UNSAAC, Perú). The computer science students have previously taken software engineering courses with enough knowledge about information systems. We selected 48 computer science students. Replication 1 (R1) was conducted with 22 undergraduate students and Replication 2 (R2) was conducted with 26 Master's students. All of them played the role of analyst and the role of end user. The subjects had previous knowledge of the unstructured requirements elicitation method but none knew anything about UREM. We spent two hours training the subjects in UREM before conducting the experiment. Apart from a theoretical description, the training activity consisted of doing a brief exercise to navigate throughout the decision tree in order to identify the different alternatives. The subjects filled in demographic questionnaires before running the experiment in order to characterize the population. Tables 6–9 summarize the main characteristics of participants and their background.

Table 7 focuses on development experience measured as the number of months or years that the students have developed software in companies. Most of the participants had work experience even though they were students. Table 6 shows the type of job and the (average, minimum, and maximum) time spent on that job. Table 8 shows their previous experience with usability and requirements elicitation methods. Only 8 persons had not heard of user interface design and only 5 persons had not heard of requirements elicitation techniques. Table 9 shows their previous experience with unstructured interviews and structured methods. Most of the subjects had not worked with any structured method before the experiment, and a few subjects had worked with some method. The item "Other" gathers other options with no agreement among the subjects. Our sample is representative of a population of novice developers. Even though the use of students in experiments limits the generalization of results, it is useful, depending on the target of the experiment, as other works such as Falessi et al. [34] claim. For this experiment, our objective is to compare subjects that have knowledge in unstructured interviews with novice subjects who have experience in structured interviews. At first glance, the structured interview is at a disadvantage due to the absence of experience. Therefore if the results are positive for the structured method, we can conclude that the structured interview is better in spite of this disadvantage. Other benefits of recruiting students are that they often come at a lower cost and are more accessible because they are taking courses at a university. Moreover, for the students, the experiment can be viewed as a learning experience of technology or methods to be evaluated.

Table 5
Summary of research questions, hypotheses, response variables, and metrics.

Research Questions	Hypotheses	Response Variables	Metrics
RQ1r	H_{01r}	Effectiveness of usability requirements elicitation	M1: Completeness
RQ1g	H_{01g}	Effectiveness of usability guidelines	M1: Correctness
RQ2r	H_{02r}	Efficiency for usability requirements elicitation	M2: Completeness/ Time
RQ3a	H_{03a}	Analyst Satisfaction	M3A: PU, PEOU, ITU
RQ3e	H_{03e}	End user Satisfaction	M3E: CSUQ

Table 6
Types of jobs performed and the time duration of the job.

None		1 month		1–3 months		More than 3–12 months		More than 12 months	
R1	**R2**	**R1**	**R2**	**R1**	**R2**	**R1**	**R2**	**R1**	**R2**
0	0	0	0	10	4	7	4	5	18

Table 7
Job experience at software development companies.

		Junior Programmer		System Analyst/ Programmer		Lan Technician		System Manager	
		R1	R2	R1	R2	R1	R2	R1	R2
Number of students		8	4	7	4	5	8	2	6
Duration (months)	Avg.	6	6	12	24	18	24	18	24
	Min	3	3	6	12	8	12	12	12
	Max	12	6	36	36	36	36	24	36

Table 8
Experience with software development.

Experience with	I have never heard of it		I have heard of it		I have some knowledge of it		I know it	
	R1	R2	R1	R2	R1	R2	R1	R2
Usability	8	8	7	6	4	7	3	5
User Interfaces design	4	2	11	8	4	11	3	5
Requirements elicitation and requirement analysis	0	0	8	2	7	13	7	11
Requirements elicitation techniques	1	4	5	5	9	9	7	8
Requirements elicitation methods	2	6	4	10	9	5	7	5

Table 9
Experience with elicitation methods.

Methods	Name of method/technique	Number	
		R1	R2
Unstructured	Interview	20	26
	Focus Group	8	12
	Questionnaires	23	25
	User stories	7	13
	Other	5	12
Structured	Eyetracking	0	0
	Remo	0	0
	Reassure	0	0
	Other	2	0

4.6. Experiment design

This section describes the within-subjects design (or repeated measures) where the subjects play two different roles, one for each treatment. We divided the group of subjects into pairs. For each pair, we randomly assigned two roles: analyst and end user. These roles were swapped for each treatment. We used two different problems (one for each treatment) in order to avoid the carryover effect, so this is *paired design blocked by experimental objects* [35]. Table 10 shows the summary of the design that was applied in both replications. In the first session, all of the pairs worked with the unstructured method. Half of the pairs were in a group named G1 and worked with Problem 1 (P1), while the other half were in a group named G2 and worked with Problem 2 (P2). In the second session, the subjects swapped their roles and all of the pairs

Table 10
Within-subjects design of the experiment.

		P1	P2
Session 1	**Unstructured interview**	G1	G2
Session 2	**UREM**	G2	G1

worked with UREM. G1 worked with P2 and G2 worked with P1.

This design has the following advantages: (1) largest sample size possible to analyze the data; (2) we avoid the learning effect; (3) the problem is not confused with the treatments. The expected time required to fulfill the user requirements defined in each treatment was around 30 min. This value was defined taking into account two factors: a previous pilot test, and the problem complexity.

The design avoids most of the threats:

- The experiment findings do not depend exclusively on one problem (since we use two problems).
- The pairs cannot share their GUI prototypes with members of other groups since all of the subjects work at the same time with the same treatment.
- All of the subjects are used in both treatments, avoiding variability among subjects.
- The context of the experiment in Session 1 is the same as in Session 2.

4.7. Experimental object

In order to observe the effects produced by the two treatments (i.e., unstructured interview and UREM), we defined two problems to elicit usability requirements, one for mobile *health center* (P1), and one for *mobile banking* (P2). Both problems are in the context of mobile applications. P1 aims to represent a system where users can login, list the health services, query the schedule for attendance, make a new appointment, and list the previous appointments. P2 aims to implement a bank management application. The end user can log in and access the bank services, such as bank accounts, location of cash dispensers, access news, and language customization. The end user has a personal section where she/he makes bank transfers, list credit cards, and update personal data. Table 11 and Table 12 respectively show the usability requirements that the subjects that play the role of the client must demand in the prototypes designed by the analyst. Even though these lists are not exclusive for each type of problem, using a different list in each problem allows us to validate different branches of the tree structure. These requirements are known by the end user, and the analyst must elicit them with interviews. When clients describe the problem to analysts, they must consider all these requirements shown in Tables 11 and 12. The description of the problems in the same way as they were distributed to the clients is shown in Appendix C.

4.8. Instrumentation

All the instruments used for running the experiment can be accessed in a Zenodo repository [36]. Below, we describe all of them:

- **Demographic questionnaires**: The online questionnaires gather information about the subjects', experience using apps or web applications, as well as their level of experience in developing information systems. This questionnaire is shown in Appendix A.
- **Experimental object**: Two problems make up the experimental objects. We have an experimenters' solution with the usability requirements that the GUI must support. This experimenters' solution is shown in Appendix B. The list of requirements shared with the end users to specify the system required is shown in Appendix C
- **Satisfaction questionnaires**: The questionnaires measure the analysts' satisfaction and the end users' satisfaction. Each questionnaire has 15 questions in a 5-Likert scale format. These questionnaires are shown in Appendix D.
- **Spreadsheets:** The spreadsheet is used to evaluate the metrics of the experiment. These calculations were carried out by two experts in usability engineering and measurement.
- **Tool**: This is the tool that supports UREM (http://hci.dsic.upv.es/urem). This tool can guide the end user through the design alternatives, recommending those alternatives that optimize the usability. The tree with of the all the questions, answers, and recommendations is shown in Appendix E.

Table 11
Mobile health center requirement list.

N°	Usability Requirements of List_Req1
1	The widgets must be self-descriptive to facilitate the understanding of the requested data.
2	To avoid errors in data entry, helpful information should be displayed.
3	If the data entry is mandatory, the user should be notified.
4	To facilitate the data entry, the choices must be shown to the user.

Table 12
Mobile banking requirement list.

N°	Usability Requirements of List_Req2
1	When inserting data, widgets must avoid errors.
2	Mandatory information must be clearly identified.
3	The system must help fix errors when they arise.
4	The system must offer actions to activate/deactivate pre-established options.

4.9. Experiment procedure

This section describes the procedure used to conduct the experiment. This procedure was executed twice, for the two replications R1 and R2). The experimental process consists in interviews within a pair of subjects. The procedure is strictly based on the experiment design configuration shown in Fig. 3. The procedure has been labelled with numbers to explain each step. Before the experiment, we explained the goals of the experiment to the experimental subjects as well as the role they played in it. We also randomly created the two groups of subjects (G1, G2). The diagram in Fig. 3 summarizes the procedure. Each number inside the circle represents the number of step that is represented in the figure.

Below we describe the steps of Session 1, where unstructured interviews is used.

Step 1. The subjects complete the demographic questionnaire. The questions were the same for all of the experimental subjects independently of their group and role.
Step 2. The experimenter divides all of the subjects into two groups (G1 and G2). The subjects play one role in each of the two sessions.
Step 3. The subjects that play the role of end users read the description of the system (P1 or P2) and the list of the usability requirements that the system must support.
Step 4. The subjects that play the role of analysts must use unstructured interviews to elicit the usability requirements by interviewing the subjects that play the role of end users. Through question-answers, the analysts must draw a prototype of GUI that satisfies the usability requirements for the specific problem.
Step 5. Once the analysts finish the GUI prototype, they complete a satisfaction questionnaire to report their level of satisfaction during the unstructured interview to elicit usability requirements. The end users must complete a satisfaction questionnaire about the result of the prototype. This questionnaire is used to determine whether or not the prototype meets the end users expectations.

Below we describe the steps of Session 2, where UREM is used.

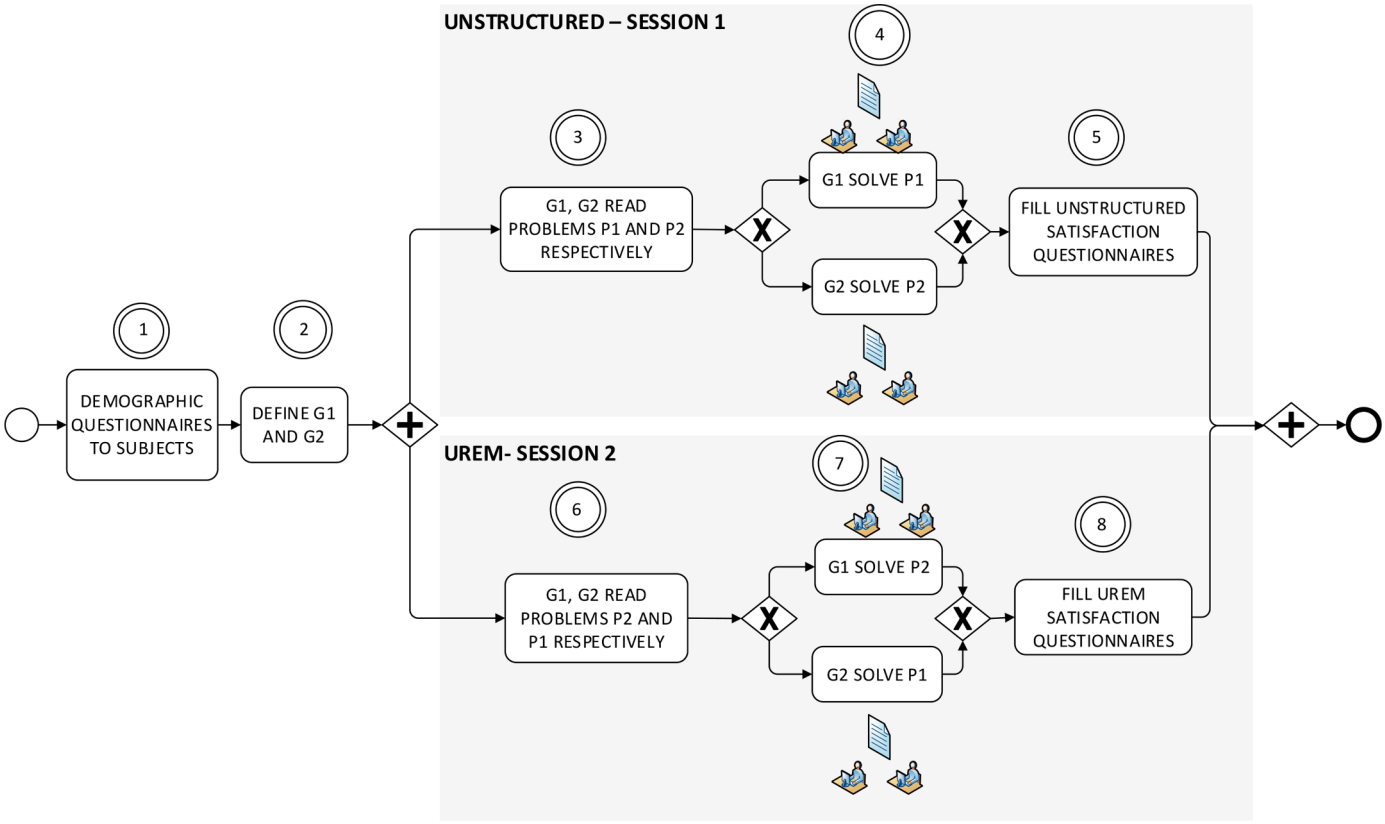


Fig. 3. Summary of the experimental procedure.

Step 6. The subjects that play the role of end users read the description of the system (a different problem from the one used in Step 3) and the list of the usability requirements that the system must support. The experiment continues in the second session with UREM.
Step 7. The subjects that play the role of analysts must use UREM to elicit the usability requirements by interviewing the subjects that play the role of end users. Following the tree structure, the analysts ask each question following the guide of the tree. The analysts must also recommend the option that best optimizes the usability based on suggestions of the tree. Afterwards, the analysts must draw a prototype of a GUI that satisfies the usability requirements for the specific problem.
Step 8.- Both the analysts and the end users complete the satisfaction questionnaire in the same way as in Step 5, but specifically for UREM.

4.10. Data analysis

Replications 1 and 2 respectively have 11 and 13 subjects playing the role of analysts. This sample size is not large enough to apply a parametric test. Therefore, when we analyze the replications separately, we opt for a non-parametric test such as Mann-Withney. We consider differences to be significant when the p-value is less than 0.05. When we analyze Replication 1 and Replication 2 together, we have a large enough sample size (24 subjects playing the role of analysts) to apply the General Linear Model (GLM). There are two requirements for applying a GLM test: homogeneity of the covariance matrices and sphericity. Levene's test is used to check the condition of homogeneity of covariance matrices where the null hypothesis is that the observed covariance matrices of the dependent variables should be equal across groups [37, 38]. All of the Levene's test p-values were greater than 0.05. Therefore, we cannot reject the null hypotheses of homogeneity of covariance, which means that the premises of the statistical tests are met in this regard. Mauchly's test is used to check the sphericity condition. In our case, however, there are only two treatments (unstructured interviews and UREM). This precludes a sphericity violation [37], and the test is unnecessary. We regard the differences between treatments as being significant when the GLM p–value is less than 0.05.

For variables with significant differences according to the GLM, we calculated the degree of these differences using partial eta squared. The partial eta squared results were interpreted as follows: Values of less than 0.3 mean a significant, but weak, effect; values between 0.3 and 0.6 mean a moderate effect, and values greater than 0.6 mean a strong effect. Statistical power is the probability of rejecting a false null hypothesis. Statistical power is inversely related to beta or the probability of making a type II error. In short, power $= 1 - \beta$. Power in software engineering experiments tends to be low, e.g., Dyba et al. [39] reports values of 0.39 for medium effect sizes and 0.63 for large effect sizes. Low values of statistical power mean that non-significant results could imply the acceptance of null hypotheses when they are false. Therefore, we calculated the power to find out whether our results were influenced by this widespread problem in software engineering. Note that effect size and power cannot be calculated in non-parametric tests.

5. Results

First, we analyzed the data of each experiment separately using Mann-Whitney as a non-parametric test. Second, we gathered the results using a moderator variable named "Replication" to look for differences between the two experiments. Replication 1 refers to the 22 undergraduate students and Replication 2 refers to the 26 Master's students (as described in Section 4.5). In the aggregation, apart from analyzing the difference for Method, we looked for differences in the Method*Problem and Method*Replication interactions. This test is based on the GLM. Below, we analyze the results ordered by response variable.

5.1. Effectiveness of usability requirements elicitation

Table 133 shows the statistical results of Replication 1 and Replication 2 separately and both replications together. Replication 1 yielded significant results for the method. The average for effectiveness in the usability requirements elicitation was 78.18 for the unstructured interview and 93.45 for UREM. Therefore, we conclude that UREM yields better effectiveness for Replication 1. Even though Replication 2 did not present statistical differences, the p-value is very close to being less than 0.05 (it is exactly 0.05). When analyzing the averages of Replication 2, the unstructured interview was 71.01 and UREM was 86.61. Thus, there is a clear trend showing that UREM yields better effectiveness in the requirements elicitation process.

Fig. 4 shows the box-plot analyzing the two replications together. The first quartile, the median and the third quartile are clearly better for UREM. When analyzing the data with GLM, we obtained a p-value of 0.000 (Table 13), which means that UREM was statistically better than the unstructured interview. The effect size (0.274) yielded a weak effect, and the power (0.978) was enough to avoid rejecting the null hypothesis for poor sample size. There are no significant differences in the Method*Problem and Method*Replication interactions, which means that the results do not depend on the problem used or the replication where the experiment was conducted.

In conclusion, **we reject H_{01r}** (the analyst effectiveness using UREM is similar that using unstructured interviews.), since UREM yielded better results than the unstructured interview.

5.2. Effectiveness of usability guidelines

Table 14 shows the statistical results after applying the non-parametric test and GLM to each replication alone and both replications together, respectively. Both Replication 1 and Replication 2 yielded significant results (p-value of 0.001 and 0.000[1]). In Replication 1, the average for the effectiveness of the guidelines was 35.36 for the unstructured interview and 62.72 for UREM. Replication 2 also showed a better average for UREM (71.76) than the unstructured interview (33.76). Therefore, we can state that, in both replications, UREM yields a design that better fits the usability guidelines.

Fig. 5 shows the box-plot of both replications together. The first quartile, the median and the third quartile are better for UREM. When analyzing the data with the GLM test, we obtained a *p*-value of 0.000 (Table 14), which means that UREM is statistically better than the unstructured interview. The effect size of 0.571 means a moderate effect and the power of 1 is very high, which ensures having enough sample size to avoid rejecting the null hypothesis for a lack of sample. There were no significant differences in the Method*Problem and Method*Replication interactions, which means that results do not depend on the problem used or the replication where the experiment was conducted.

In conclusion, **we reject H_{01g}** (the analyst effectiveness using usability guidelines in UREM is similar to that of using unstructured interviews) since UREM yields better results than the unstructured interview.

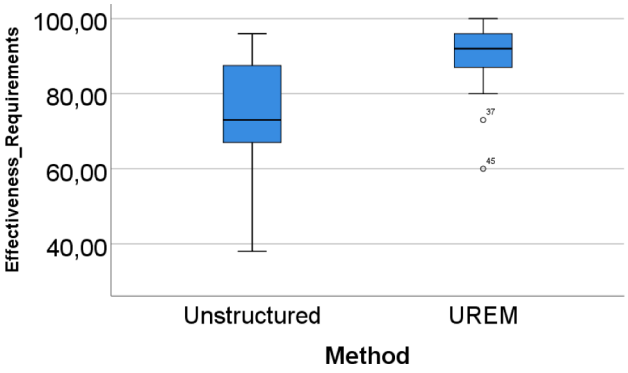


Fig. 4. Box plot of effectiveness for usability requirements elicitation with both replications.

[1] We use only 3 decimals even though the statistical package works with more.

Table 13
Statistical results of effectiveness for usability requirements elicitation.

	Rep. 1	Rep. 2	Both rep.
p-value Method	**.001**	.05	**.000**
p-value Method*Problem	–	–	.195
p-value Method*Replication	–	–	.195
Effect size	–	–	.274
Power	–	–	.978

Table 14
Statistical results of effectiveness for usability guidelines.

	Rep. 1	Rep. 2	Both rep.
p-value Method	**.001**	**.000**	**.000**
p-value Method*Problem	–	–	.05
p-value Method*Replication	–	–	.05
Effect size	–	–	.571
Power	–	–	1

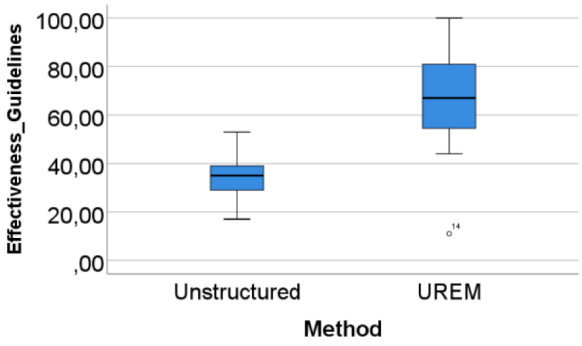


Fig. 5. Box plot of effectiveness for usability guidelines with both replications.

5.3. Efficiency for usability requirements elicitation

Table 15 shows the statistical results of Replication 1 and Replication 2 separately and both replications together. Replication 1 shows a significant result with a p-value of 0.018 while Replication 2 shows no significant results with a p-value of 0.489. In Replication 1 the average was 0.953 for the unstructured interview and 1.34 for UREM. In Replication 2, the average was 0.998 and 0.886 respectively. The results are contradictory in both replications, but the differences are so slight that we cannot draw conclusions.

Fig. 6 shows the box-plot of efficiency aggregating both replications.

Table 15
Statistical results of efficiency.

	Rep. 1	Rep. 2	Both-rep.
p-value Method	**.018**	.489	.220
p-value Method*Problem	–	–	**.021**
p-value Method*Replication	–	–	**.021**
Effect size	–	–	–
Power	–	–	.230

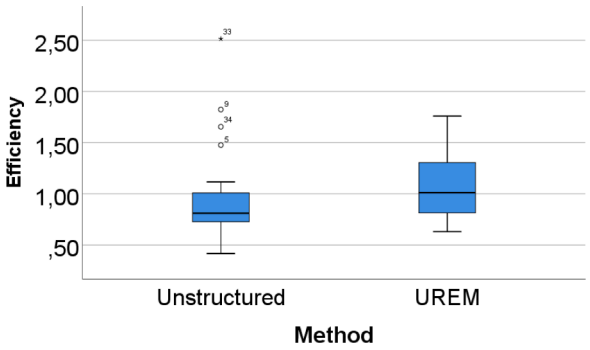


Fig. 6. Box plot of efficiency.

The median, the first quartile, and the third quartile are slightly better for UREM. Although these differences are not strong, UREM shows a trend with a better efficiency. The GLM test showed no significant results (p-value 0.220), with a power of 0.230, which is low. A larger sample size may produce some significant differences between treatments. Both the Method*Problem and Method*Replication replications yielded significant differences. This means that there is a specific problem and a specific replication that affects the result. To analyze this idea, in Fig. 7 we show profile plots of both interactions. Fig. 7(a) shows that the Bank Problem (P2) is better in UREM. Fig. 7(b) shows that Replication 1 is better for UREM.

In conclusion, **we cannot reject H_{02r}** (the analyst efficiency using UREM is similar to that of using unstructured interviews), so there are no differences between the unstructured interview and UREM.

5.4. Analyst satisfaction

Analyst satisfaction was measured using three different metrics: Perceived Usefulness (PU), Perceived Ease of Use (PEOU), and Intention to Use (ITU). When analyzing the p-values of each replication separately (Tables 16–18), only PEOU yielded significant results in Replication 1 (p-value was 0.028). The average in this case was 16 for the unstructured interview and 13.63 for UREM, so the subjects perceived the unstructured interview being as easier to use. The other averages were: PU in Replication 1: 30.18 in the unstructured interview and 25.9 in UREM; ITU in Replication 1: 10.81 in the unstructured interview and 9.81 in UREM; PU in Replication 2: 29.46 in the unstructured interview and 28.76 in UREM; PEOU in Replication 2: 15.07 in the unstructured interview and 14.69 in UREM; ITU in Replication 2: 10.15 in the unstructured interview and 10.23 in UREM. Note that most of the results yielded slightly better satisfaction for the unstructured interview, but this difference was not significant.

Figs. 8–10 show the box plot of the two replications together for PU, PEOU, and ITU, respectively. PU and ITU yielded the same median for both treatments. In the case of PEOU, the median was slightly better for the unstructured interview. For the three metrics (PU, PEOU, and ITU), the third quartile was very similar for both treatments, but the first quartile was better for the unstructured interview. The statistical test of the GLM did not yield significant differences for any metric (all p-values were higher than 0.05) and there were no differences for Method*Problem and Method*Replication interactions. The statistical power was

Table 16
Statistical results of PU.

	Rep. 1	Rep. 2	Both rep.
p-value Method	.065	1	.128
p-value Method*Problem	–	–	.434
p-value Method*Replication	–	–	.434
Effect size	–	–	–
Power	–	–	.330

Table 17
Statistical results of PEOU.

	Rep. 1	Rep. 2	Both rep.
p-value Method	**.028**	1	.141
p-value Method*Problem	–	–	.561
p-value Method*Replication	–	–	.561
Effect size	–	–	–
Power	–	–	.311

Table 18
Statistical results of ITU.

	Rep. 1	Rep. 2	Both rep.
p-value Method	.193	.579	.429
p-value Method*Problem	–	–	.636
p-value Method*Replication	–	–	.636
Effect size	–	–	–
Power	–	–	.122

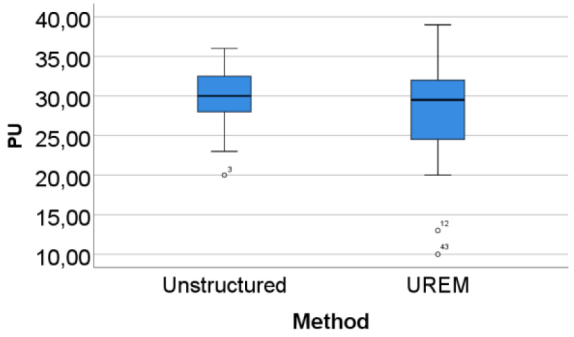


Fig. 8. Box plot of PU.

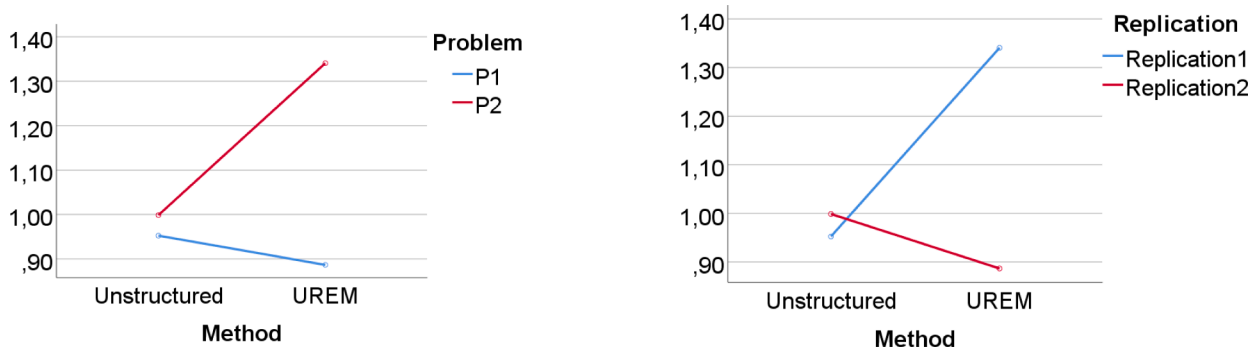


Fig. 7. (a) profile plot of Method*Problem. (b) profile plot of Method*Replication.

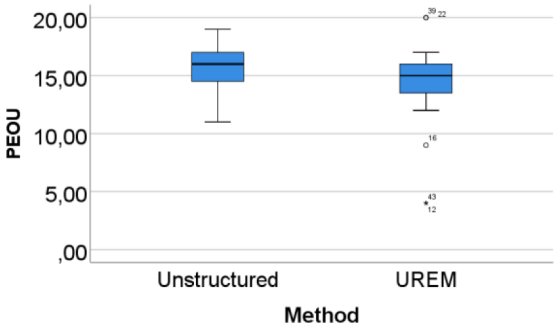


Fig. 9. Box plot of efficiency.

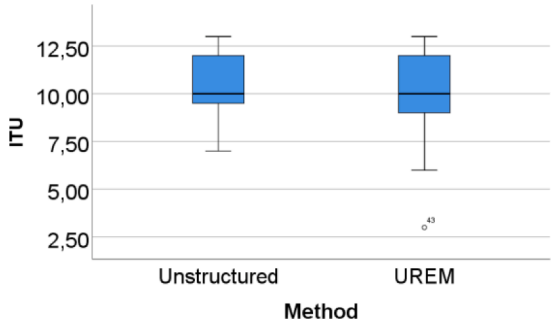


Fig. 10. Box plot of efficiency.

low in the three metrics, so significant differences may appear in a larger sample size.

In conclusion, **we can only reject H_{03a}** (The analyst satisfaction using UREM is similar to that of using unstructured interviews) **for the metric PEOU in Replication 1**, where the unstructured interview yields a better satisfaction level. The other metrics did not present significant differences in each replication separately or together.

5.5. End user satisfaction

End user satisfaction is measured using two metrics: the CSUQ questionnaire and the satisfaction of the end user with the recommendation offered by the analyst to improve usability. The p-values of each replication individually were higher than 0.05 (Tables 19 and 20), so there were no significant differences between treatments in any replication. The average of CSUQ in Replication 1 was 70.72 for the unstructured interview and 75.81 for UREM. In Replication 2 the average was 78.23 for the unstructured interview and 66.46 for UREM. The median of satisfaction with the recommendations to improve the usability in Replication 1 was 4 for both the unstructured interview and UREM. In Replication 2, it was also 4 for both the unstructured interview and UREM. All of this descriptive data does not yield any conclusion in the differences between the two treatments.

Figs. 11 and 12 show the box plot of the two replications together for the CSUQ questionnaire and the end user satisfaction with the recommendations to improve usability. The medians in both plots were similar. The first quartile was slightly better for the unstructured interview in both metrics. The third quartile was better for the unstructured interview in the CSUQ metric, while the third quartile does not present differences in the metric of satisfaction with the recommendations. The statistical test did not yield significant differences for any metric (all p-values were higher than 0.05), and there were no differences for Method*Problem and Method*Replication interactions.

In conclusion, **we cannot reject H_{03e}** (the end user satisfaction using UREM is similar to that of using unstructured interviews), so there were no differences between treatments in terms of satisfaction with the recommendations to improve usability. Table 21 summarizes the results of the statistical tests for all of the hypotheses.

Table 19
Statistical results of CSUQ questionnaire.

	Rep. 1	Rep. 2	Both rep.
p-value Method	.151	.153	.426
p-value Method*Problem	–	–	.136
p-value Method*Replication	–	–	.136
Effect size	–	–	–
Power	–	–	.123

Table 20
Statistical results of end user satisfaction with the recommendations.

	Rep. 1	Rep. 2	Both rep.
p-value Method	.562	.287	.504
p-value Method*Problem	–	–	.396
p-value Method*Replication	–	–	.396
Effect size	–	–	–
Power	–	–	.101

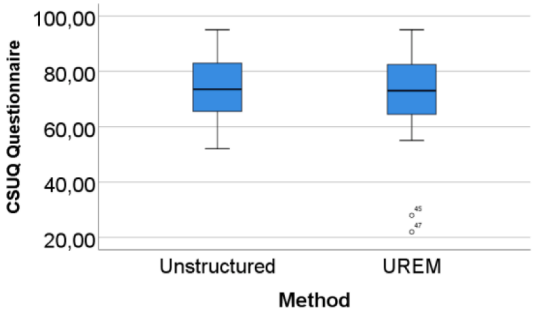


Fig. 11. Box plot of CSUQ questionnaire.

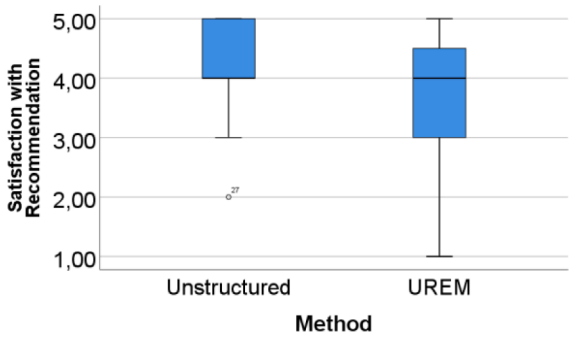


Fig. 12. Box plot of end user satisfaction with the recommendations.

Table 21
Summary of the results.

Hypotheses	Results
H_{01r}	Effectiveness of usability requirements elicitation is **significantly** better for UREM
H_{01g}	Effectiveness of usability guidelines is **significantly** better for UREM
H_{02r}	Efficiency for usability requirements elicitation is **the same** for UREM and the unstructured interview
H_{03a}	Analyst Satisfaction is the **same** for UREM and the unstructured interview
H_{03e}	End user Satisfaction is the same for UREM and the unstructured interview

5.6. Usability requirements problems and usability guidelines compliance

Next, we describe the actual results in terms of usability requirements problems and level of compliance with usability guidelines found during the experimentation. Fig. 13a and b show the percentage of usability requirements used in the experiment that are successfully elicited in P1 and P2 respectively. These requirements were defined in Tables 11 and 12 and used to measure the response variable Effectiveness for usability requirements elicitation. Both plots show that UREM obtains a better percentage than the Unstructured method. If we focus on UREM for P1, the lowest effectiveness is for "Display different choices" since several prototypes did not show all the menu options by default. "Helpful information" is around 85% since most prototypes included helpful information to describe the options and actions that each interface offers. "Notification of mandatory data" and "Self-descriptive widgets" are close to 100%. Almost all interfaces included self-descriptive widgets and identified the mandatory widgets to fill in. If we focus on UREM for P2, the lowest level is for "Avoid errors". A few interfaces did not include a list of enumerated options to avoid errors. "Flexibility to activate/deactivate" is around 85%, which means that most interfaces included options to modify the default options; for example, the date of today, or your current position to look for the closest bank to extract money. "Help to fix errors" and "Notification of mandatory data" are close to 100%. Most interfaces included messages to guide the end-user when an error arises, and mandatory data is clearly identified in the interfaces. Note that, even though the requirements are the same for both P1 and P2, UREM yields better effectiveness in the usability requirements elicitation.

Fig. 14 shows the percentage of usability guidelines that are satisfied in P1. These usability guidelines are the ones used to build the tree structure used in the experiment (Appendix B). The percentage of agreement with usability guidelines is used in the experiment to measure the response variable Effectiveness of usability guidelines. Note that there is a large difference between UREM and Unstructured method for "Use a dialogbox to show error message", "Use asterisk for mandatory fields", "Use alternative text for textfields", and "Use dropdown for a menu with several options". In the Unstructured method, most prototypes did not specify the mechanisms to notify about errors. Moreover, they used the red color or a bold font to highlight the mandatory data (instead of an asterisk). Almost no interface used alternative text for textfields. Menus with several options were designed mainly with a list (instead of a dropdown). The level of agreement with usability guidelines improves when using UREM. All the guidelines are larger than 65% except for "Use dropdown for the menu with several options". Even though the tree structure recommended the use of a dropdown, several clients preferred a design with all the items in the interface without a dropdown.

Fig. 15 shows the percentage of usability guidelines satisfied in P2 both with UREM and with the Unstructured method. Note that there are usability guidelines around 0% with the Unstructured method: "Use text and icon for help actions", "Use a dialogbox to show error message", and "Use alternative text for textfields". Even though many subjects used text to describe actions, a few of them complemented the text with an icon. Moreover, as in P1, a few prototypes included dialogboxes to show errors messages and a few prototypes used alternative text for textfields. The guidelines "Use asterisk for mandatory fields" and "Use dropdown for a menu with several options" show a value of around 20%. This is because mandatory fields are represented in red color or bold and menus with several options are displayed with items without dropdown. On the contrary, some guidelines are very similar between UREM and the Unstructured method: "Use the whole screen to select the different options", and "Use a vertical list". Subjects tend to use all the size of the screen to design the interface, and lists are always shown in vertically. If we analyze the results for UREM, all values of agreement with usability guidelines improve. The only guideline that is below 65% is "Use dropdown for a menu with several options". This shows that even though UREM recommends usability guidelines, the results of the design are not 100% compliant with usability guidelines. The client chooses between applying the usability guidelines or any other alternative she/he prefers.

6. Discussion

This section discusses the results, looking for justifications for the data and comparing the outcomes with previous existing empirical works. We analyze the results for each hypothesis. H_{01r} yields significant differences, where UREM presents better effectiveness in the requirements elicitation process. Since effectiveness is defined as the percentage of usability requirements successfully elicited, this means that working with UREM helps the analyst identify successfully more usability requirements than an unstructured interview does. These differences arise in Replication 1 and when both replications are aggregated, but it does not appear in Replication 2. This may be due to the low sample size if we analyze replications individually. The descriptive data in Replication 2 shows a trend of more effectiveness of UREM than the unstructured interviews. Note that the previous experience of the subjects was mainly in unstructured interviews (Table 7), and only two subjects had experience in structured interviews. Even though the experience in the two treatments is so unbalanced, the effectiveness with UREM (a structured method) is clearly better when a short training is provided before the experiment. This result aligns with previous works in the literature, which state that structured interviews are the most effective elicitation techniques in a wide range of domains and situations [40,41].

H_{01g} also yields significant differences, where UREM shows better effectiveness applying usability guidelines. This means that analysts working with UREM are more compliant with usability guidelines than analysts working with the unstructured interview. Note that the use of

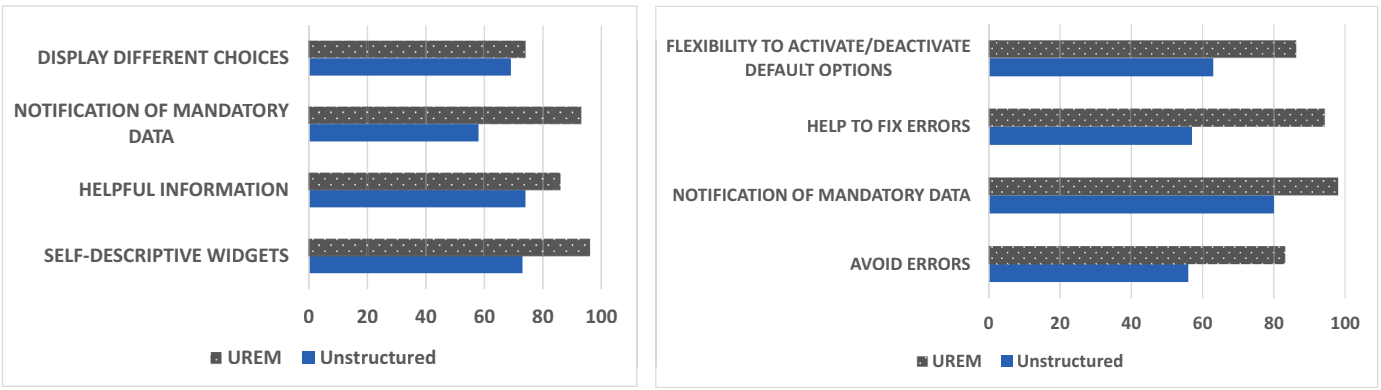


Fig. 13. (a) Percentage of usability requirements correctly elicited in P1. (B) Percentage of usability requirements correctly elicited in P2.

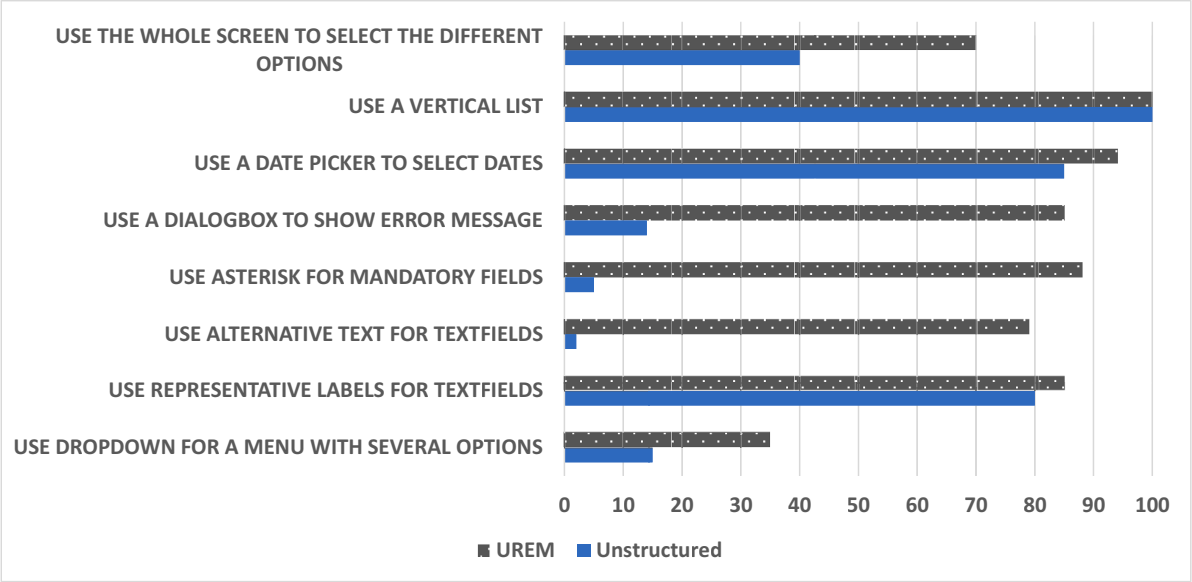


Fig. 14. Percentage of usability guidelines satisfied in P1.

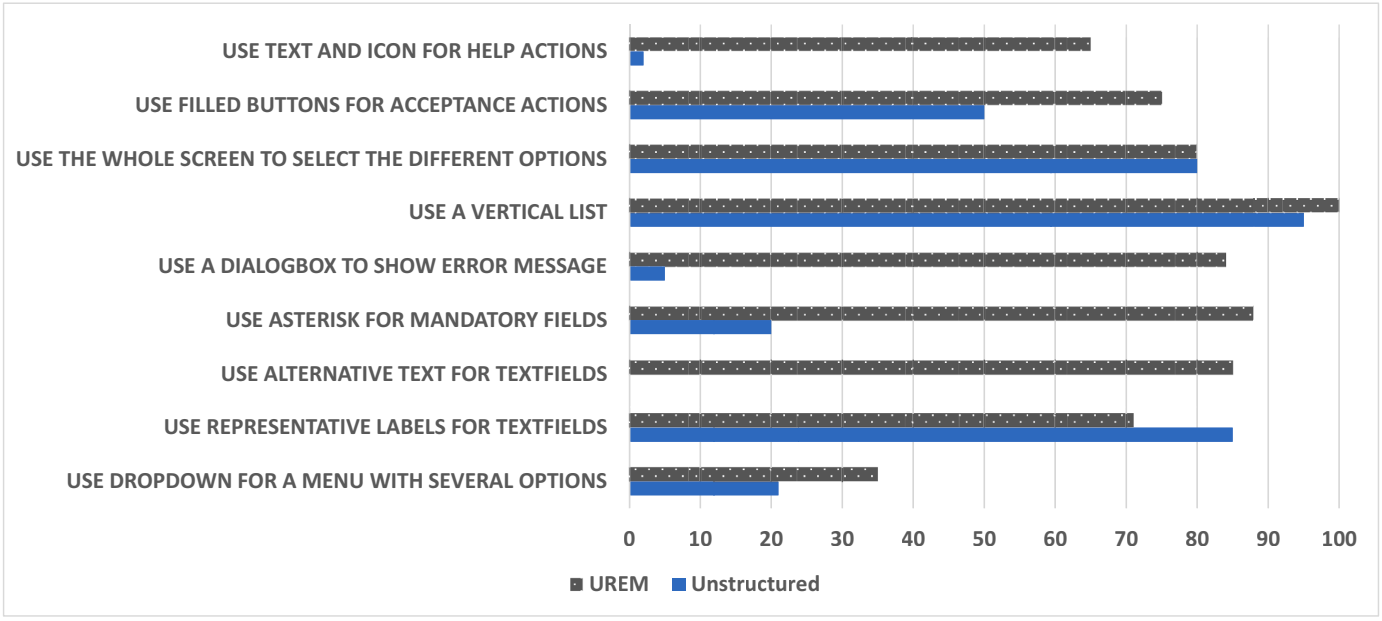


Fig. 15. Percentage of usability guidelines satisfied in P2.

UREM does not ensure the support of usability guidelines in the GUI designs. UREM suggests which design alternative is the one that best fits the usability requirements. However, the choice of the final design depends on the agreement between the analyst and the end user, and this choice may be different from the one suggested by UREM. Based on these results, we can state that most analysts agreed to accept the suggestions of the UREM method to improve usability. Median for the effectiveness of usability guidelines (Fig. 5) is 70%. This means that even using UREM, some subjects did not follow the usability suggestions. Note that the subjects that were recruited in the experiment had experience in the requirements elicitation process but only half of them had experience with usability (Table 8). Even though their experience in usability is not high, the designed GUI are compliant with the usability guidelines. This means that UREM helps design usable interfaces even when the analyst is not an expert in usability guidelines. There are previous works that have classified the different usability guidelines, reporting advantages and describing how to deal with the guidelines [42]. To our knowledge, there are no previous works that structure the information of the guidelines in a tree structure as a helping guide during the requirements elicitation process. UREM provides a clear contribution to the field of usability guidelines assistance.

H_{02r} does not yield significant differences between UREM and the unstructured interview. Differences only appear in Replication 1. Moreover, if we analyze the descriptive data after aggregating both replications, we see that the averages are very similar between UREM and the unstructured interview. This means that, even though the use of UREM could lead to an increase in the required time, the data shows that this increase in time is not real. The efficiency needed to navigate throughout the tree structure is the same as the efficiency needed to conduct an unstructured interview. This conclusion may be biased by the size of the tree, but, in our experiment, we are not working with a small tree. This may reduce the effort required by the analyst for the

navigation. The whole tree is shown in Appendix E. This result contradicts the conclusions of other previous works, which state that structured interviews such as JAD require more effort than unstructured ones such as Brainstorming [43]. The statistical power is low, so to be completely sure that significant differences in terms of efficiency do not arise between the two treatments, we need a larger sample size. In this hypothesis, we identified two interactions as being significant: Method*Problem and Method*Replication. The differences between UREM and the unstructured interview are more evident in P2 (bank) than in P1 (health center). UREM required more time in P1, which reduced the efficiency. The subjects who were recruited for the experiment may have had more experience in interaction with banking systems, so the effort spent for each treatment was low in this problem because the analysts could have had a possible prototype in mind for this type of system. A health center application is usually used with less frequency than a banking application. This may have led to requiring more effort to elicit the requirements, which may highlight the difference in efficiency between the treatments. With regard to the Method*Replication interaction, the difference between treatments is more evident in Replication 1. This could be due to the profile of the subjects of that replication; they are undergraduate students with low experience in software development companies (Table 6). This result together with the significant result for efficiency in Replication 1 leads to thinking that UREM shows a better efficiency in a context with low professional experience.

H_{03a} yields significant differences for the PEOU metric in Replication 1. When analyzing the box plot of the two replications together, there is a trend where the unstructured interview obtains a better satisfaction. The low power may justify that this significant difference is not present when the two replications are aggregated together. Since the significant result focuses only on one replication, general conclusions cannot be drawn. Note that most of the subjects have experience in the area of software development (Table 8), and they have a good background with unstructured interviews (Table 10). Despite this advantage for the unstructured interview compared with UREM, the subjects do not have a clear preference for either method. To the authors knowledge, there are no previous works that have experimentally evaluated how the structured interviews may affect the analysts' satisfaction. This lack of empirical works may be because satisfaction is a broad term with several perspectives. For example, the work of Elrakaiby et al. [44] states that satisfaction depends on motivation, relevance of the realization, and relevance of the statement. All of these characteristics are difficult to control in an empirical evaluation.

H_{03e} does not yield significant differences between UREM and the unstructured interview. This means that from the point of view of the end user, there is no difference between the two treatments. Even though the usability requirements are elicited with more effectiveness using UREM, the end users are no more satisfied with the designed GUI. Previous works in the literature state that there is a relationship between usability features supported by the system and end user satisfaction [45]. Note that the statistical power is very low in both metrics that analyze the hypothesis; it is possible that some significant differences may arise with a larger sample size. Moreover, the designed GUI are only some parts of the system; the analysts did not design the whole system. An experiment involving more types of interfaces with more complexity might help to find differences between the treatments. We plan to replicate the experiment with a larger sample size and with more complex problems in order to analyze in detail how the use of UREM affects the end user's satisfaction.

As conclusions of our analysis, we can state that UREM helps to improve the effectiveness of the usability requirements elicitation process. Moreover, UREM helps the inclusion of usability guidelines in designs even though the analysts that make the design are not experts in usability. These advantages do not involve a loss of efficiency in the requirements elicitation process and GUI design.

7. Threats to validity

We have classified the threats to validity of our experiment based on the classification provided by Wohlin et al. [46]. We described each type of threat as: avoided, incurred, and mitigated.

Conclusion validity. This threat is concerned with issues that affect the ability to draw the correct conclusions about relationships between the treatment and the outcome. Threats of this type are: (1) *Low statistical power:* This appears when the sample size is low. After the aggregation of both replications, we obtain enough statistical power for response variables that are related to effectiveness. However, efficiency, analyst satisfaction and end user satisfaction is affected by this threat due to low power. (2) *Violated assumptions of statistical tests*: GLM has some assumptions that must be satisfied in order to conduct the test. We avoided this threat since the aggregation of both replications satisfies all of these assumptions. (3) *Fishing:* This appears when experimenters are looking for a specific result. Even though one experimenter was the designer of UREM, the other two experimenters that participated in the design and interpretation of the results were not the authors of UREM. Therefore, this threat was mitigated. (4) *Reliability of measures*: This appears when measures have errors due to problems with instruments. We mitigated this threat by conducting a pilot study with two subjects before conducting the real experiment. This helped to check all of the experimental artefacts. (5) *Reliability of treatment implementation:* There is a risk that the implementation is not similar between different replications. We mitigated this threat since the experimenter who described the treatments and conducted the experiment was the same in both replications. It is also possible that end users describe the usability requirements wrongly, and this may affect RQ1r and RQ1g. This is mitigated because both treatments suffer this threat, so it should not affect positively or negatively a specific treatment. (6) *Random heterogeneity of subjects:* This appears when the sample size is too heterogeneous and this variation is larger than the variation produced by the treatment. Subjects of R2 (Master's students) have more job experience than subjects of R1 (undergraduate students). Since we analyze each replication individually, we can analyze whether or not there are differences between both profiles.

Internal validity. This threat is concerned with influences that may affect the dependent variable with respect to a causality which the researchers are unaware of. Threats of this type that may appear are: (1) *History*: This appears when the treatments are applied at different moments. Our experiment was affected since unstructured interviews and UREM are applied in different sessions. Even though we tried to maintain the same context and conditions, we cannot ensure that the different moment of each session did not affect the results. (2) *Maturation*: This appears when the subjects react differently as time pass. We mitigated this threat by conducting each session in a maximum of one hour. This was to avoid boredom and fatigue. (3) *Instrumentation*: This appears when the instruments used in the experiment may affect the results. This threat was mitigated since the satisfaction questionnaires were validated previously. The analyst satisfaction questionnaire is based on the TAM by Davis [47] while the end user satisfaction is based on the CSUQ [33]. (4) *Selection*: How the subjects are recruited may affect the results. In our experiment, the participants participated as part of a course. The participation in the experiment was not mandatory, but it gave the participants extra credit in the course. This may lead to subjects being overmotivated, which may result in a threat. (5) *Mortality*: This appears when the subjects abandon the experiment before finishing. We avoided this threat since no subject left the experiment. (6) *Compensatory rivalry:* This appears when the subjects receive different treatments. We avoided this threat since all of the subjects received both treatments and all of the subjects played both roles (analyst and end user). (7) *Differences between roles*: playing the role of the analyst can be easier than playing the role of the end-user. When subjects play the role of the analyst, they act with the role that their course is preparing for. This may lead to more motivated subjects when they play the role of the analyst. We have mitigated

this threat by swapping the roles between both treatments.

Construct validity. This threat is concerned with generalizing the results of the experiment to the concept or theory behind the experiment. Threats of this type that our family of experiments may be open to are: (1) *Inadequate preoperational explication of constructs*: This appears when the theory behind the treatment has not been sufficiently defined. We avoided this threat since the UREM method had a proper definition before conducting the experiment. (2) *Mono-operation bias*: This appears when experiments with only one factor may under-represent the construct. We mitigated this threat by analyzing the interaction of the method with the problem and the replication. This was to look for differences due to context or problem complexity. (3) *Mono-method bias*: This appears when a simple type of metrics is used. We mitigated this threat since the analyst satisfaction and end user satisfaction depend on more than one metric. However, the effectiveness of usability requirements elicitation, the effectiveness of usability guidelines, and efficiency were affected by this threat. (4) *Problem homogeneity*: This appears when experimental problems are too homogeneous to generalize the results to other problems. We mitigated this threat by choosing problems from different domains.

External validity. This threat is concerned with conditions that limit the ability to generalize the results of experiments to industrial practice. Threats of this type are: (1) *Interaction of selection and treatment*: This appears when the subjects are not representative of the population that we want to generalize. We mitigated this threat since, even though the subjects were students, they had previous experience in real software development projects. (2) *Interaction of setting and treatment*: This appears when the experimental setting or the material are not representative of our target of study. We mitigated this threat since the usability requirements and the problems were aligned with the context where UREM is used. (3) *Interaction of history and treatment*: this appears when the experiment is conducted at a special time that may affect the results. Our experiment was affected by this threat since each replication was conducted on different days. (4) *Interaction between research questions:* this appears when there is a correlation between research questions. The experiment suffers this threat since RQ2r might be somehow correlated to RQ1r. The fewer usability requirements satisfied by the analyst, the shorter the time required to define them.

8. Conclusions

This article presents an empirical experiment that compares structured interviews with unstructured interviews in order to elicit usability requirements. Structured interviews are operationalized as UREM, which is a method based on a decision tree where the analyst guides the interview by navigating throughout the tree structure. Each branch of the tree includes a question for the end user with possible answers. Moreover, the answer that is more compliant with existing usability guidelines is recommended. In the unstructured interview method, the analyst must elicit usability requirements without any guide. In this work, this control treatment is referred to as unstructured interview. The evaluation is conducted to analyze four response variables: effectiveness in the usability requirements elicitation; effectiveness in the application of usability guidelines; efficiency; the analyst's satisfaction; the end user's satisfaction. As significant results, UREM is more effective in the usability requirements elicitation and also more effective in designing interfaces that are compliant with usability guidelines.

Note that even though the recruited subjects are students, a large percentage of them have experience in real software development companies. Therefore, the results could be generalizable to any person with some type of experience in software development, not just students. The experiment was conducted with two different problems so the results are not associated to a single problem. This also facilitates the generalization of results.

Some lessons have been learned during the conduction of the experiment: (1) The effort to build the tree in UREM is high. This is something that was not analyzed in the experiment, but the required effort is not null. Note that this effort can be recovered; the same tree structure is useful for any future development; (2) The recommendations during the tree structure navigation may be different depending on the usability guidelines used to build the tree. Even though most usability guidelines agree on the characteristics that optimize usability, there are some guidelines that may present some contradictions. In the end, the expert at usability that builds the tree structure is the one who chooses the most suitable usability guidelines for the recommendations; (3) Most of the end users accepted the usability recommendations. This value may have been different if the subjects had more experience in usability characteristics. Other experiments can be conducted to determine how the level of experience may affect the results. (4) Due to the structure of questions, UREM may leave no room for discovering designs not included as alternatives in the tree structure.

As future work, we plan to replicate the experiment in order to enhance the sample size. Some response variables such as the analyst' satisfaction and the end user' satisfaction have a low statistical power. With a larger sample size we may be able to identify more significant differences for these response variables. Moreover, we aim to analyze more factors, such as previous experience in usability concepts and the complexity of the problems. In a future validation of UREM, we plan to include other metrics such as creativity when the tree structure is built and when it is used in the interviews; qualitative analysis of how designers perceive the use of UREM; need of training for the method; overall appreciation of the guidance provided; reusability in multiple contexts of use; perception of the time and effort necessary to prepare the tree structure; and flexibility to run the method. We also plan to compare UREM with other structured interview methods.

CRediT authorship contribution statement

Yeshica Isela Ormeño: Investigation, Methodology. **José Ignacio Panach:** Writing – original draft, Formal analysis. **Oscar Pastor:** Supervision, Writing – review & editing.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests

Data availability

Available in https://doi.org/10.5281/zenodo.7646554.

Acknowledgements

This work was developed with the support of the National University of San Antonio Abad of Cusco under the program Yachayninchis Wiñarinanpaq CONCYTEC and FONDECYT, the support of Generalitat Valenciana with CoMoDID (CIPROM/2021/023) and GENI (CIAICO/2022/229), as well as the support of the Spanish Ministry of Science and Innovation co-financed by FEDER in the project SREC (PID2021–123824OB-I00).

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.infsof.2023.107324.

References

[1] M. Rajanen and N. Livari, "Usability cost-benefit analysis: How usability became a curse word?," pp. 511–524, 2007.
[2] D. Quiñones, C. Rusu, V. Rusu, A methodology to develop usability/user experience heuristics, Comput. Stand. Interfaces 59 (2018) 109–129.

[3] ISO, ISO 9241-11: ergonomic requirements for office work with visual display terminals (VDTs): part 11: guidance on usability, 1998.
[4] ISO/IEC, "ISO /IEC 25010: 2011 systems and software engineering@ systems and software quality requirements and evaluation (SQuaRE)@ system and software quality models," 2013.
[5] H.A. Hutahaean, R. Govindaraju, I. Sudirman, Identifying usability risks for mobile application, in: Proceedings of the International Conference on Engineering and Information Technology for Sustainable Industry, Tangerang, Indonesia, 2021, pp. 1–6.
[6] E.M. Rey, V.M. Bonillo, D.A. Ríos, Session details: theme: software design and development: UE - usability engineering track, in: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, Limassol, Cyprus, 2019.
[7] Y.I. Ormeño, J.I. Panach, N. Condori-Fernández, Ó. Pastor, Towards a proposal to capture usability requirements through guidelines, in: Proceedings of the IEEE 7th International Conference on Research Challenges in Information Science (RCIS), 2013, pp. 1–12.
[8] J. Nielsen, Usability Engineering, Morgan Kaufmann, 1993.
[9] M.J. Muller, Participatory design: the third space in HCI. The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, Associates Inc, 2002, pp. 1051–1068. L. Erlbaum.
[10] K. Petersen, R. Feldt, S. Mujtaba, M. Mattsson, Systematic Mapping Studies in Software Engineering, EASE, 2008, pp. 68–77.
[11] F. Gunduz, A.S.K. Pathan, Usability improvements for touch-screen mobile flight booking application: a case study, in: Proceedings of the International Conference on Advanced Computer Science Applications and Technologies, ACSAT, 2012, pp. 49–54.
[12] O.D. Troyer, E. Janssens, A feature modeling approach for domain-specific requirement elicitation, in: Proceedings of the IEEE 4th International Workshop on Requirements Patterns (RePa), 2014, pp. 17–24.
[13] P. Fahey, C. Harney, S. Kesavan, A. McMahon, L. McQuaid, B. Kane, Human computer interaction issues in eliciting user requirements for an electronic patient record with multiple users, in: Proceedings of the 24th International Symposium on Computer-Based Medical Systems (CBMS), 2011, pp. 1–6.
[14] M. Temper, S. Tjoa, M. Kaiser, Touch to authenticate—continuous biometric authentication on mobile devices, in: Proceedings of the 1st International Conference on Software Security and Assurance (ICSSA), 2015, pp. 30–35.
[15] T.R Silva, M. Winckler, C. Bach, Evaluating the usage of predefined interactive behaviors for writing user stories: an empirical study with potential product owners, Cogn. Technol. Work 22 (2020) 437–457.
[16] E.A. De Carvalho, A. Jatobá, P.V.R. De Carvalho, Usability for complex systems?: an experimental evaluation with functional resonance analysis method, in: Proceedings of the 18th Brazilian Symposium on Human Factors in Computing Systems (IHC), 2019, pp. 1–4.
[17] J.A. Nhavoto, Å. Grönlund, W.P. Chaquilla, SMSaúde: design, development, and implementation of a remote/mobile patient management system to improve retention in care for HIV/aids and tuberculosis patients, JMIR Mhealth Uhealth 3 (2015).
[18] E. Elias, D. Miquilino, I.I. Bittencourt, T. Tenório, R. Ferreira, A. Silva, S. Isotani, P. Jaques, Towards an ontology-based system to improve usability in collaborative learning environments, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 7315, LNCS, 2012, pp. 298–303.
[19] X. Yuan, X. Zhang, An ontology-based requirement modeling for interactive software customization, in: Proceedings of the IEEE International Model-Driven Requirements Engineering Workshop (MoDRE), 2015, pp. 1–10.
[20] Z.S.H. Abad, S. Moazzam, C. Lo, T. Lan, E. Frroku, H. Kim, Loud and interactive paper prototyping in requirements elicitation: what is it good for?, in: Proceedings of the IEEE 7th International Workshop on Empirical Requirements Engineering (EmpiRE), 2018, pp. 16–23.
[21] G. Márquez, C. Taramasco, Using dissemination and implementation strategies to evaluate requirement elicitation guidelines: a case study in a bed management system, IEEE Access 8 (2020) 145787–145802.
[22] S. Tiwari, S.S. Rathore, and A. Gupta, "Selecting requirement elicitation techniques for software projects," pp. 1–10, 2012.
[23] A. Abdallah, R. Hassan, M.A. Azim, Quantified extreme scenario based design approach, in: Proceedings of the ACM Symposium on Applied Computing, 2013, pp. 1117–1122.
[24] G. Vitiello, R. Francese, M. Sebillo, G. Tortora, M. Tucci, UX-requirements for patient's empowerment - the case of multiple pharmacological treatments: a case study of it support to chronic disease management, in: Proceedings of the IEEE 25th International Requirements Engineering Conference Workshops, REW, 2017, pp. 139–145.
[25] Y. Tanikawa, R. Okubo, S. Fukuzumi, Process support method for improved user experience, NEC Tech. J. 8 (2014) 28–32.
[26] Z.S.H. Abad, S.D.V. Sims, A. Cheema, M.B. Nasir, P. Harisinghani, Learn more, pay less! lessons learned from applying the wizard-of-oz technique for exploring mobile app requirements, in: Proceedings of the IEEE 25th International Requirements Engineering Conference Workshops (REW), 2017, pp. 132–138.
[27] M. Peruzzini, M. Germani, Designing a user-centred ICT platform for active aging, in: Proceedings of the IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA), 2014, pp. 1–6.
[28] H. Takeshi, F. Shin'ichi, Applying human-centered design process to SystemDirector Enterprise development methodology, NEC Tech. J. 3 (2008) 12–16.
[29] S. Sharma, S. Pandey, Revisiting requirements elicitation techniques, Int. J. Comput. Appl. 75 (2013) 35–39.
[30] T.R. Gruber, C. Baudin, J.H. Boose, J. Webber, Design rationale capture as knowledge acquisition, ML Workshop (1991).
[31] C. Martinie, P. Palanque, M. Winckler, S. Conversy, DREAMER: a design rationale environment for argumentation, modeling and engineering requirements, in: Proceedings of the 28th ACM International Conference on Design of Communication, São Carlos, São Paulo, Brazil, 2010, pp. 73–80.
[32] N. Juristo, A.M. Moreno, Basics of Software Engineering Experimentation, Springer Science & Business Media, 2013.
[33] J.R. Lewis, IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use, Int. J. Hum. Comput. Interact. 7 (1995) 57–78.
[34] D. Falessi, N. Juristo, C. Wohlin, B. Turhan, J. Münch, A. Jedlitschka, M. Oivo, Empirical software engineering experts on the use of students and professionals in experiments, Empir. Softw. Eng. 23 (2018) 452–489.
[35] N. Juristo, A. Moreno, Basics of Software Engineering Experimentation, Springer, 2001.
[36] Y. Ormeño, J.I. Panach, and Ó. Pastor, "Experimental material of the article "an empirical experiment of a usability requirements elicitation method based on interviews"," Z. 10.5281/zenodo.7646554, 2023.
[37] L.S. Meyers, Applied Multivariate Research: Design and Interpretation, Sage Publications, Thousand Oaks, 2006. G. Gamst and A. J. Guarino.
[38] L.S. Meyers, G. Gamst, A.J. Guarino, Applied Multivariate Research: Design and Interpretation, Sage Publications, 2016.
[39] T. Dybå, V.B. Kampenes, D.I. Sjøberg, A systematic review of statistical power in software engineering experiments, Inf. Softw. Technol. 48 (2006) 745–755.
[40] A.M. Davis, Ó.D. Tubío, A.M. Hickey, N.J. Juzgado, A.M. Moreno, Effectiveness of requirements elicitation techniques: empirical results derived from a systematic review, in: Proceedings of the 14th IEEE International Requirements Engineering Conference (RE'06), 2006, pp. 179–188.
[41] N. Bahurmuz, R. Alnajim, R. Al-Mutairi, Z. Al-Shingiti, F. Saleem, B. Fakieh, Requirements elicitation techniques in mobile applications: a systematic literature review, Int. J. Inf. Technol. Proj. Manag. (IJITPM) 12 (2021) 1–18.
[42] M.S. Goundar, B.A. Kumar, A.B.M.S. Ali, Development of usability guidelines: a systematic literature review, Int. J. Hum. Comput. Interact. (2022) 1–19.
[43] O. Okesola, K. Okokpujie, R. Goddy-Worlu, A. Ogunbanwo, O. Iheanetu, Qualitative comparisons of elicitation techniques in requirement engineering, J. Eng. Appl. Sci. 14 (2019) 565–570.
[44] Y. Elrakaiby, A. Ferrari, P. Spoletini, S. Gnesi, B. Nuseibeh, Using argumentation to explain ambiguity in requirements elicitation interviews, in: Proceedings of the IEEE 25th International Requirements Engineering Conference (RE), 2017, pp. 51–60.
[45] J.M. Ferreira, S.T. Acuña, O. Dieste, S. Vegas, A. Santos, F. Rodríguez, N. Juristo, Impact of usability mechanisms: an experiment on efficiency, effectiveness and user satisfaction, Inf. Softw. Technol. 117 (2020), 106195.
[46] C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, A. Wesslén, Experimentation in Software Engineering, Springer Science & Business Media, 2012.
[47] F.D. Davis, User acceptance of information technology: system characteristics, user perceptions and behavioral impacts, Int. J. Man Mach. Stud. 38 (1993) 475–487.