Requirements Engineering (2018) 23:401–424
https://doi.org/10.1007/s00766-018-0290-5

RE 2017

Improving the identification of hedonic quality in user requirements: a second controlled experiment

Andreas Maier[1,2] · **Daniel M. Berry**[3]

Received: 21 October 2017 / Accepted: 23 March 2018 / Published online: 12 May 2018

Abstract
Systematically engineering a good user experience (UX) into a computer-based system under development demands that the user requirements of the system reflect all needs, including emotional, of all stakeholders. User requirements address two different types of qualities: pragmatic qualities (PQs), that address system functionality and usability, and hedonic qualities (HQs) that address the stakeholder's psychological well-being. Studies show that users tend to describe such satisfying UXes mainly with PQs and that some users seem to believe that they are describing an HQ when they are actually describing a PQ. The problem is to see if classification of any user requirement as PQ-related or HQ-related is difficult, and if so, why. We conducted two controlled experiments involving the same twelve requirements-engineering and UX professionals, hereinafter called "analysts." The first experiment, which had the twelve analysts classifying each of 105 user requirements as PQ-related or HQ-related, shows that neither (1) an analyst's involvement in the project from which the requirements came nor (2) the analyst's use of a detailed model of the qualities in addition to the standard definitions of "PQ" and "HQ" has a positive effect on the consistency of the analyst's classification with that of others. The second experiment, which had the twelve analysts classifying each of a set of 50 user requirements, derived from the 105 of the first experiment, showed that difficulties seem to be caused both by the analyst's lacking skill in *applying* the definitions of "PQ" and "HQ" and by poorly written user requirement specifications. The first experiment revealed that classification of user requirements is a lot harder than initially assumed. The second experiment provided evidence that the difficulties can be mitigated by the combination of (1) training analysts in applying the definitions of "PQ" and "HQ" and (2) casting user requirement specifications in a new template that forces provision of the information needed for reliable classification. The experiment shows also that neither training analysts nor casting user requirement specifications in the new template, by itself, mitigates the difficulty in classifying user requirements.

Keywords Hedonic quality · Pragmatic quality · Controlled experiment · User experience · Project involvement · Definitions of pragmatic and hedonic qualities · Quality model · Classifier training · User story template

1 Introduction

Since the beginning of the current millennium, researchers have been showing that a software product's quality can be separated into two quality dimensions: pragmatic (a.k.a. ergonomic) quality (PQ) and hedonic quality (HQ) [1, 2]. While a PQ is a product quality that is relevant to achieving a particular task (a.k.a. utility) or to how a user accesses this functionality in an efficient way (a.k.a. usability), an HQ is a product quality that emphasizes the user's psychological well-being. In terms of HQs, a user's psychological well-being can be increased by three product characteristics [1, 3]:

- by the product's provision of stimulation or fun to the user, by the product's challenging and novel character,

✉ Andreas Maier
a.f.maier@googlemail.com

✉ Daniel M. Berry
dberry@uwaterloo.ca

[1] Department of Computer Science, Technical University of Kaiserslautern, Gottlieb-Daimler-Strasse 42, Building 47, 67663 Kaiserslautern, Germany

[2] Fraunhofer Institute for Experimental Software Engineering IESE, Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany

[3] Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada

- by the user's self-identification or enhancement achieved through the product's communication, or conveying, of important personal values[1] to the user, and
- by the symbolic character of the product that reminds the user of personally relevant prior experiences, relationships, or thoughts.

1.1 Definitions of "PQ," "HQ," and "objective quality"

The definitions given here and to the experiment subjects are consolidated from those found in the literature [3–5].

A user requirement of a product is related to PQ if any of the following three cases applies:

1. The requirement addresses the product's relevant functionality to effectively achieving a particular task, i.e., the product's utility, as it is perceived by the user.
2. The requirement addresses the product's ways to efficiently access its functionality to effectively achieve a particular task, i.e., the product's ease of use, as it is perceived by the user.
3. The requirement addresses the product's overall usefulness, e.g., by aiming at product assessments such as structured, practical, predictable, or simple.

An example of a user requirement focusing on PQs, addressing an increase in the product's ease of use, is: "The product must be easy to handle."

A user requirement is related to HQ if any of the following two cases applies:

1. The requirement addresses the user's psychological well-being through the fulfillment of his human psychological needs. The following six human psychological needs can be fulfilled through the interaction with a software product: autonomy, competence, sensory stimulation, relatedness, popularity, and feeling of security.
2. The requirement addresses the product's overall appeal, e.g., by aiming at hedonic product assessments such as amusing, beautiful, captivating, competitive, creative, enjoyable, exciting, exploratory, fun, impressive, innovative, integrating, interesting, inventive, motivating, novel, original, playful, pleasant, premium, presentable, stylish, and thrilling. This overall appeal is manifested through emotional expressions such as experiencing fun, enjoyment, happiness, and pleasure. Such positive emotions are caused mainly by the provision of stimulation by the product's challenging and novel character, or by personal identification by communicating important personal values to relevant others.

This concern with PQ and HQ is situated within the general concern about psychological needs and emotion in RE [6–10]. However, HQ addresses more than just psychological needs and emotions and more than the everyday meaning of "hedonic." An example of a user requirement focusing on HQs, addressing an increase in the user's well-being by having the product enhance the user's self-identification, is: "The smartphone should perfectly fit my personal style by being exciting, to increase my personal image."

If a user requirement appears to be related to *both* PQ *and* HQ, then it is classified as related to HQ, because

1. an HQ is more important for the fulfillment of psychological needs and thus for inducing pleasure [3], and
2. in any case, each HQ is built on a PQ; that is, for each hedonic requirement, there is at least one pragmatic requirement that fulfills the hedonic requirement.

Note that even though HQs are much more important for positive UX than are PQs, HQs are not the only things that matter, because, as noted, PQs are needed to facilitate the provision of HQs. Also, there may be a positive UX even with poor PQ or with poor HQ. With poor PQ, a product, such as an expensive one, that communicates a personal image or a status is appealing, nonetheless. With poor HQ, merely useful products can be appealing when a user has a strong personal bond with the product, for example, when the product is a gift from a good friend, a mate, or another relevant person. So, eventually, what makes a product appealing is its HQ, whose manifestation is facilitated by related PQs.

Other user requirements that are not related to UX address functional features of a product that are based on the product's objective quality, i.e., that are provided by the product itself and do not involve the user's interaction with the product. Such product qualities do not directly fulfill user goals, but are necessary for the eventual fulfillment of user goals. Such a product quality can be referred to as a system function or a system feature that is expressed as a system requirement. So, a requirement that focuses on an objective product quality addresses qualities such as the product's functionality, capacity, interaction style, and portability, without the provision of subjective assessments of those qualities.

An example of a requirement focusing on a product's objective quality is: "As a user, I want to get the app via the Apple App Store to use it on my iOS system." This requirement addresses the technical availability of an iOS app, but the requirement focuses neither on the app's utility or usability nor on the user's well-being. So it would be classified as neither pragmatic nor hedonic.

[1] For example, an expensive product communicates the value of the product's user's being rich to other people.

1.2 Problem description

As mentioned, Hassenzahl et al. found that HQs are important potential sources for increased software quality [11]. Moreover, studies by Hassenzahl et (different) al. [12] showed that the fulfillment of basic human needs by an interactive system's HQs is the source of a system user's positive user experience (UX) that manifests itself in the emotional consequence of feeling pleasure. As a result, HQ-related aspects of UXes are considered to be more important than PQ-related aspects of UXes [1, 3–5, 13–16].

According to Hassenzahl [3], a desired product that induces a positive UX is one with an uncompromising combination of strong PQs and strong HQs. While HQs are important to induce a positive UX, each HQ must be implemented with the help of at least one underlying PQ. Therefore, requirements analysts must be able to reliably identify both PQs and HQs, and not just PQs, in user requirements to lay the foundation for the engineering of a desired software product.

It is important to emphasize that a focus on PQs is not wrong, particularly for software, such as a weather predictor or a factory automator that has complex functionality whose consideration dominates the software's requirements analysis. Probably because of the traditional concern for functional requirements to the neglect of non-functional requirements [17], it appears that, in practice, even for software that is regarded as novel and exciting to a user and that is related to a user's self-identification, such as a mobile phone, a digital camera, an MP3 player, or a computer game, PQs remain the focus of the software's requirements analysis, even to the extent of being the sole focus [18]. When HQs have been ignored, the software's developers end up not knowing what really satisfies the software's users, i.e., what increases their psychological well-being. The developers have to learn what satisfies users by costly trial and error. Avoiding this cost requires eliciting both PQs and HQs in user requirements. Section 2, on related work, documents the tendency to ignore HQs during RE and the effect of this ignoring.

Accordingly, we had begun research to devise methods to assist analysts in eliciting PQs and HQs in user requirements. Specifically, these methods are

- the provision of more detailed definitions of "PQ" and "HQ," and
- more discussion within the project team about PQs and HQs in the user requirements during its analysis of these user requirements

The fact that analysts, and even users, have focused on PQs to the neglect of HQs in the past meant that the focus of any such method would likely be on HQs, to counteract the tendency to focus on PQs. Nevertheless, we were finding that the methods that we were proposing were not working as well as we had hoped. The question was "Why?"

For a successful application of a method, analysts need to understand the underlying concepts. For the proposed methods, he or she must be able to classify each requirement as being related to a PQ, an HQ, or neither. We checked informally and saw that analysts (as well as the second author when this research began) could not *reliably* identify PQs and, especially, HQs, and that they could not *reliably* distinguish between PQs and HQs, perhaps because of HQs' subjective and context-sensitive character. If this inability is not addressed and reversed, systematically engineering HQs into products will not be possible. Achieving a positive UX will be haphazard and inefficient, if at all possible, because the baseline for a good UX is not explicitly available in requirements specifications. We realized that an empirical study was needed to confirm that the classification of user requirements by their kinds of quality was the difficulty, and if so, to begin to understand why.

1.3 Overall research objectives

The overall objective of our research is to learn about the difficulties of determining where a user requirement lies in the PQ-related–HQ-related spectrum, with the aim of making it possible for analysts to reliably, correctly classify user requirements whenever they need to. The conference paper, of which this paper is an extension [19], describes one experiment which answered some but not all questions about the difficulty. The present paper, called "this paper," describes a second, follow-up experiment that answers questions not answered by the first experiment and questions raised by the results of the first experiment.

1.4 Vocabulary

In the experiments, and therefore, in this paper, the following definitions apply:

- analyst: the person doing the classification
- user requirement: a general statement of user goals or business tasks that a user needs to perform or has performed
- user requirement specification: some expression of a user requirement
- user story: an informal specification of a user requirement, describing one particular feature from the perspective of one particular user playing one particular role, and written in a constrained structure that forces provision of the feature, the role, and the perspective. The structural constraint is often provided by the now standard Connextra user story template [20]:

> As a *role*, I want *goal/desire* so that *benefit*.
> or
> As a *role*, I want *goal/desire*.
> That is, the so that *benefit*., the benefit, or rationale, part, is optional.

In addition, in the remainder of the paper, the following phrases, in which "X" stands for a specification of a user requirement in some form, including that of a user story, are used with the given meanings:

- "specification" = "requirements specification"
- "classify X as pragmatic" = "classify X as a PQ-related X"
- "classify X as hedonic" = "classify X as an HQ-related X"
- "classify X" (without a following "as") = "classify X as hedonic or pragmatic"
- "the definitions" = "the definitions of 'PQ' and 'HQ"'

1.5 Remainder of the paper

In the rest of this paper, Sect. 2 provides an overview of related work. Section 3 summarizes the first experiment. Section 4 uses open questions from the first experiment to motivate the second experiment. Section 5 describes the design of the second experiment including the treatment and control variables, the description of the subjects, the randomization, and the hypotheses of the experiment; Sect. 6 describes the analysis procedure and shows the raw data of the experiment. Section 6.4 discusses the qualitative feedback collected from the subjects. Section 7 evaluates the quantitative data to allow assessing support for the hypotheses. Section 8 evaluates the validity of the reported results. Section 9 discusses the results and points to future work. Section 10 discusses the general implications of the results. Section 11 concludes the paper.

2 Related work

Independently of, and soon after Hassenzahl identified hedonic qualities as important for inducing positive UX, others, e.g., Ramos et al. [7] and Thew et al. [8] considered the impact of so-called soft issues such as stakeholder emotions and politics on the requirements for computer-based systems (CBSs). They observed that the introduction of any CBS to an organization transforms the organization and changes the work patterns of the system's users in the organization. These changes interact with the users' values and beliefs and stakeholder politics to trigger emotional responses which are sometimes directed against the CBS and its proponents. The advice given by Ramos et al. and Thew et al. was for requirements analysts to watch for signs of relevant emotions, values, beliefs, and political actions as they are expressed or occur during requirements elicitation and other RE-time interactions with stakeholders.

Besides the related work cited in the first paragraphs of Sect. 1, there has been some empirical work about PQs and HQs, demonstrating the importance of considering HQs during RE. This work also situates the concern about PQs and HQs in the general RE context, and among goals, functional, and non-functional requirements.

In a study with 45 participants, Partala and Kallinen [18] examined the importance of PQ-related aspects for the identification of the participants' most satisfying UXes. They asked each participant to describe the single most satisfying and the single most unsatisfying UX that he or she encountered in the past six months. For the purposes of this study, for a participant, "a UX" was defined as "an experience related to a single event, in which the participant's usage of a technological system formed a substantial part." Then, they tried to learn from each participant the reasons that the chosen events were so satisfying and unsatisfying.

For each of the two UXes that a participant reported, the participant indicated whether technical problems and usability problems were involved in the UX. The participant indicated also the importance of various PQ-related aspects to his or her choices for the most satisfying and the most unsatisfying UX. Finally, the participant gave a qualitative description of each of the two UXes.

For the most satisfying UXes, technical problems and usability problems played only minor roles. However, for the most unsatisfying UXes, technical problems and usability problems played major roles. PQ-related aspects were scored as being not very important for the identification of the most satisfying UXes. Nevertheless, 78% of the 45 qualitative descriptions of the most satisfying UXes and 91% of the 45 qualitative descriptions of the most unsatisfying UXes mentioned PQ-related aspects.

In the same study, Partala and Kallinen show that HQs are more important than PQs in determining satisfaction of a UX; Usability and utility, both PQ-related aspects, can lead to a neutral state of satisfaction, but do not cause emotional reactions that exceed this neutral state

Diefenbach and Hassenzahl [p. 5][21] describe the undesired impact of the dominance of PQs over HQs on requirements analysis and product design. Requirements analysis may reveal solely pragmatic user needs, simply because direct probing of the stakeholders causes the stakeholders to feel that they must justify their feature requests, and PQs are more fashionable to report for technical products than are HQs. Consequentially, study after study points out a pronounced stakeholder requirement

for PQs. And if these studies are taken seriously, the apparent demand for the PQs reported in the studies will be reflected in product design as well. This focus on PQs may then result in overly functional products with only a small potential to create the experiential quality so crucial for emotional attachment.

In addition, Diefenbach and Hassenzahl show that many a user, when asked privately, expresses a preference for a hedonic product over a pragmatic product, *even* when the user understands the product as purely pragmatic, and cites only PQs as reasons for buying the product.

The observation that many a person who has an iPhone *loves* it for being an iPhone and not for its functionality, referring to the device as "my iPhone" rather than "my cellphone" or "my smartphone," might appear to contradict this last finding by Diefenbach and Hassenzahl. In this case, "being an iPhone" would be an HQ that some cellphones have and others do not. There is no contradiction. With any technical product, functionality and thus PQs come into a user's mind much more easily than do emotional aspects and thus HQs. The iPhone is a technical product that also focuses on its HQs, such as its captivating type of interaction, the cool digital assistant, the luxury it communicates, and the innovative technology it often introduces to the market. So, the iPhone is an example of a desired technical product with strong PQs and strong HQs. It is neither merely pragmatic nor merely hedonic product.

In a literature search of 151 HQ-related publications referring to the concept of HQ, Diefenbach, Kolb, and Hassenzahl [2] explored (1) the impact of HQs on human–computer interaction (HCI) research and on the design of interactive products and (2) how the concept of HQ was used and further developed. The research revealed that interest in HQs increases and that focus is shifted to an experiential perspective in technology design, as the frequency of scientific publications on the concept of HQs increases. Additionally, the integration of HQs into HCI enabled a better understanding of human experiences in HCI and allowed for the development of new models of UX with better predictive power of assessment, preference, and acceptance of technology. Moreover, the research confirmed the importance of the separation of PQs and HQs and that products with a high HQ created more positive affect than products with a high PQ. From this study, they say [2, p. 8] "we need new methods to assess and explore the hedonic" and "more elaborated approaches to design for hedonic quality." They conclude their publication with "we should not only use the hedonic as a part of models or as a standard evaluation routine, but engage into further developing, clarifying, empirically substantiating, and critically reexamining the concept and its claims."

3 Summary of the first experiment

The first experiment [19] was designed to test how difficult it is to classify user requirements as PQ-related, HQ-related, or neither. An additional aim of the experiment was to begin to identify causes of the observed difficulties. It was hypothesized that any difficulty in classifying user requirements for a software system is mitigated

- by providing to the subjects very detailed definitions, such as those provided in the quality model described in the online experiment materials package [22].
- by having as subjects those in the system's developing team, who have greater familiarity with the system's goals and user requirements than do others.

These mitigations were chosen for testing because they are parts of a proposed method to support analysts in classifying user requirements.

The research questions that drove the first experiment were:

RQ1: What is the influence of the level of detail of the definitions used for the classification on the difficulty of classifying user requirements?

RQ2: What is the influence of project involvement on the consistency among analysts of the classification of user requirements?

Formulated as a GQM model [23, 24], the first experiment's goal was:

- Study classifications of user requirements,
- for the purpose of characterizing the difficulty of doing the classifications,
- from the viewpoint of analysts who are requirements engineers or UX professionals,
- in the context of

    1. user requirements for software for which UX matters,
    2. specifications of user requirements formulated as user stories, and
    3. definitions of "PQ" and "HQ."

The top part of Fig. 1, consisting of Goal G, Research Questions RQ1 and RQ2, Metrics M1 through M7, and the blue arcs between them, constitute the corresponding GQM tree. This tree gives the research questions, RQ1 and RQ2, that drove the first experiment.

The user requirements classified in the experiment were elicited in a Fraunhofer IESE research project, the Digital

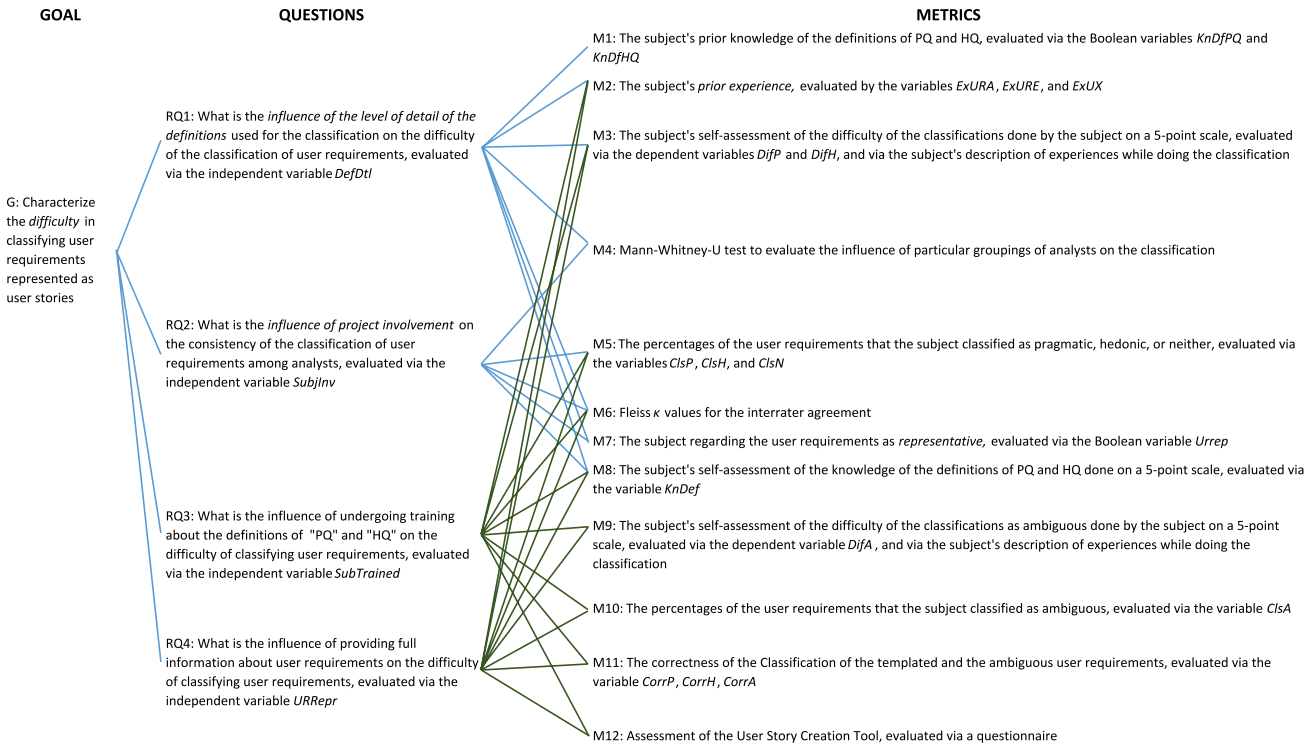

Fig. 1 GQM Tree For First and Second Experiments

Villages project [25], in which an app for logistics, information transmission, and citizen participation in villages was being developed. The app was provided as both a mobile app and a Web app. With the app, a citizen can ask other citizens in the same village to bring a particular parcel to a particular place, e.g., from the post office or from the seller to the citizen's home. Each citizen who agrees to transport a parcel gets a particular reward, e.g., money or credit points that can be used to reduce prices on particular goods in particular shops. Via an information channel, a citizen can trace the app development project itself. Additionally, a citizen can provide ideas for the improvement of the app and for new products and services in the village that are implemented by the citizens themselves. Any other citizen can rate and comment on these improvement ideas. To motivate a citizen for both bringing parcels to another citizen and providing ideas to the project, strong HQs are necessary.

Given that the first author was working at the Fraunhofer IESE, to find subjects, we used a convenience sampling of RE and UX professionals working at the Fraunhofer IESE and of RE and UX professionals who were members of the Digital Villages project, in which the user requirements considered in the experiment were elicited, analyzed, and discussed. It was easy to convince these potential subjects to participate in the experiment since they were all employees at the *IESE (Institute for Experimental Software Engineering)*, in which there is a tradition of employees' participating in experiments.

In designing the first experiment, we identified two orthogonal dichotomous, equal-sized groupings of the twelve subjects that participated in the experiment. The first grouping was actually the method to select the twelve subjects from a pool of potential subjects

1. *PMG* (project member grouping) of six subjects randomly selected from the Digital Villages project team and
2. *NPG* (non-project grouping) of six subjects randomly selected from a pool of other potential subjects, none of whom were in the Digital Villages project team.

The orthogonal grouping was achieved by giving to one random half of each of the above groupings only minimal definitions of "HQ" and "PQ" and giving to the other half these same definitions and an elaborate model of the same qualities:

1. *LDG* (low detail grouping) of six randomly selected subjects that used only the definitions given in Sect. 1.1, and
2. *HDG* (high detail grouping) of six randomly selected subjects that used both these definitions and a quality model, summarized below, that had been developed prior to our even deciding that this experiment was necessary.

The quality model provided to the *HDG* describes the three factors of UX, namely user, system, and HCI, and decomposes each factor into criteria, subcriteria, and metrics. PQ and HQ are criteria of the factor HCI. Each element of the model is described by a definition found in literature. Hence, the quality model describes the relations between the elements of UX, particularly between PQ, HQ, and the other elements, in more detail than do the definitions.

The full quality model is available online as part of the experimental materials package [22] developed for the first experiment.

The research questions led to two pairs of hypotheses[2], alternative and null, to be tested by the first experiment:

HAE: The classification of user stories is significantly easier for subjects in the *HDG*, who used both the definitions and the quality model, than for subjects in the *LDG*, who used only the definitions.

HNE: The difficulty of the classification of user stories by subjects is the same for subjects in the *HDG* and in the *LDG*, and is thus independent of whether or not the subjects used the quality model in addition to the definitions.

HAC: The classification of user stories is significantly more consistent among the subjects in the *PMG*, who were in the Digital Villages project, than among the subjects in the *NPG*, who were not in the project.

HNC: subjects in the *PMG* is the same as among the subjects in the *NPG*, and is thus independent of whether or not the subjects were in the Digital Villages project.

In the evaluation of these hypotheses, consistency is measured by interrater reliability, which is measured by the Fleiss κ value [26]. A pair of subjects' classifications are considered consistent with each other when their classifications achieve the standard *significant interrater agreement*, with $\kappa > 0.6$ [27].

We had considered phrasing the research questions and the hypotheses in terms of correctness of the subjects' classifications, but could not. To do so would require knowing the correct classification of each user story in order to create a gold classification against which to compare any subject's classification. Producing the correct classification of a user story requires *both* (1) understanding the concepts of PQ and HQ thoroughly and (2) knowing what was meant by the user who expressed the story's requirement. The first author, an expert in PQs and HQs, was not a member of the Digital Villages project, and no member of the Digital Villages project was anywhere near as expert in PQs and HQs as was the first author. Thus, there were no readily available persons who could reliably produce the correct classifications for all 105 user stories, and there was not enough time to bring the first author enough up to speed on the Digital Villages project to be able to produce the correct classifications. By the time the second experiment was ready to conduct, the first author had learned enough about the Digital Villages project to rewrite the user requirements in a way that allowed easy production of the gold standard. See Sect. 5.1 for details.

The first experiment showed that neither project involvement nor the addition of a detailed quality model to the definitions mitigates the difficulty of classifying user requirements into pragmatic requirements and hedonic requirements. Specifically, for each null hypothesis, HNE and HNC, the experiment did not show enough evidence that it could be rejected.

We thought that possible causes of the difficulties might be the analysts' insufficient skill in applying the definitions, or poorly written user requirements. We noted that future studies will need to focus on these potential causes. In particular, the impact of the representation of user requirements and of focused training in identifying PQ and HQ in user requirements will need to be evaluated. Although HNE could not be rejected, prior knowledge of the definitions and a self-assessment of knowing these definitions might not be sufficient to promote accurate classification of user requirements. Besides the two causes of difficulties in classifying user requirements, the experiment revealed that classifying user requirements is much harder than initially assumed and that further studies on possible causes of difficulties and to find solutions to these difficulties were necessary.

4 Motivation for second experiment

As we began to design the second experiment, we thought a little bit more about what the first experiment showed, namely that

1. neither the level of detail of the definitions used for the classification (addressing RQ1),
2. nor project involvement and thus problem domain expertise (addressing RQ2)

have a positive influence on the ease of classifying user requirements and on the consistency of these classifications among analysts. We hypothesized that possible causes of the difficulties might be

1. the analysts' insufficient skill in applying the definitions, or
2. poorly written user requirements.

[2] In a hypothesis name, "A" means "alternative," "N" means "null," "E" means "concerning ease of classification," and "C" means "concerning consistency of the classifications."

Although subjects of the first experiment claimed to have knowledge about the definitions, prior knowledge of the definitions and a self-assessment of knowing these definitions might not be sufficient to ensure accurate classification of user requirements. Actual experience in classification of user requirements might be required.

So, the second experiment focuses on identifying the causes of the difficulty in classifying user requirements that we observed in the first experiment. In particular, the second experiment evaluates the impact

1. of focused training in classifying user requirements and
2. of the representation used of user requirements,

with the aim of answering the following two new research questions:

RQ3: What is the influence of undergoing training about the definitions of "PQ" and "HQ" on the difficulty of classifying user requirements?

RQ4: What is the influence of providing full information about user requirements on the difficulty of classifying user requirements?

Recall that the overall objective of the research is to provide a means to improve the correctness of analysts' classifications of user requirements. In the lead up to the first experiment, as mentioned, we realized that there was no reliable way to determine for any user requirement its correct classification, for the simple reason that the specifications of many user requirements do not give enough information to correctly classify them. So, the objective of the research became to provide a means to improve the consistency of analysts' classifications of user requirements.

The first experiment showed that substantial agreement is hard to achieve with the normal user requirement specifications, i.e., with user requirements represented in the Connextra user-story template. Among other things, we decided to try improving the writing of specifications of user requirements by providing a new, more complete template that forces the writer to provide all the details, either missing from or only optionally summarized in the Connextra template, that allow reliable, correct classification. This full information includes purpose, product quality, perspective, need, intention, product quality, related software product, functionality, and implementation. Note that some of this information is the Connextra template's optional benefit information given in more detail.

The new template itself is:

{Purpose} In order to ≺ *Action* ≻ ≺ *Goal/Desire* ≻
[<*concretization of Goal/Desire*>]
{Product Quality} by a better ≺ *Product Quality* ≻,
{Perspective} As <*Role*>,
{Need} I want to be able to <*Need*>,
{Intention} to <*Intention*>.
For example, I have experienced
{Product Quality} a better ≪ *Product Quality*≫
When I interacted
{Related Software Product} with <*Software Product*>
{Functionality} that provided the possibility to
<*Functionality*>
{Implementation} implemented by <*Implementation*>.

In the following brief explanation of the new template, "the user" means "the user that is telling the user story," and "the product" means the "interactive software product whose requirements are expressed in this user story.

- The Purpose denotes either a pragmatic quality or a hedonic quality that the user expects the product to provide as a result of implementing the user story.
- The Product Quality denotes the objective product quality, i.e., the objective feature that is required to implement the given purpose.
- The Perspective denotes the stakeholder role played by the user.
- The Need describes the concrete need of the user that must be satisfied by the product's features.
- The Intention denotes the user's intended interaction with the product, to provide an analyst an idea why the user told this user story.
- The Related Software Product denotes a specific, named interactive software product that the user used in the past; it should be possible for a developer to find information on the product, to be able to understand the user's past experience with the named product.
- The Functionality refers to a feature of the related software product that fulfilled the given purpose in the user's past.
- The Implementation denotes the implementation of the given need by the related software product; it shows a developer what exactly fulfilled the given purpose in a user's past.

With the labels provided in the new template, an analyst finds the information required for classification without having to read the whole user requirement, to interpret the user requirement, or to interrogate the actual users. See the online appendix [28] for a document containing a full explanation of the new template's fields and an example of its use for each of a pragmatic and a hedonic user requirement.

In the lead up to the second experiment, we rewrote all the user requirement specifications using the new template.

At this point, we realized that we had enough information to reliably determine the correct classification for every user requirement in the set. It was now possible to create gold standard classifications, one for each user requirement, against which to measure correctness of any analyst's classifications.

Finally, we developed an online interactive User Requirement Classification Trainer which helps its user to become more reliably correct in classifying user requirements. It runs its users through a number of example user requirements specifications. For each example, the trainer first asks the user to classify it. After the user responds, the trainer indicates whether the user's classification is correct. Then, regardless of the correctness of the user's classification, the trainer gives the correct classification, explains the reasons for that classification, and shows the example's path through the quality model, which was developed for the first experiment [19, 22].

To summarize, the overall objective and goal inherited from the first experiment together with the additional research questions leads to the two-experiment GQM tree shown in the entire Fig. 1. As is common, this tree suggests more experimentation than is reported in this paper.

5 Experiment design

The second experiment followed a 2×2 mixed factorial design [24, 29] with the same twelve subjects who participated in the first experiment.

Using the same subjects for two experiments always raises the question of learning effects *between* the two experiments. However, in this case, we actually *wanted* a learning effect and to measure it, namely whether training the subjects in classification is effective. In addition, we did test whether the subjects spent any of the time between the experiments thinking about the particular requirements being classified and found that they had not. See Sect. 5.1 for details.

5.1 Independent and dependent variables

The independent variable about a subject is

1. whether the subject has undergone training in applying the definitions, *SubjTrained*, with *SubjTrained* = "Tr" meaning "**Tr**ained in applying the definitions" and *SubjTrained* = "Un" meaning "**Un**trained in applying the definitions."

SubjTrained was used to build two equal-sized groupings of the twelve subjects,

1. *TrSG* (trained subject grouping) of six subjects randomly selected from the pool of the twelve subjects, and
2. *UnSG* (untrained subject grouping) of the remaining six subjects from the twelve subjects.

Each subject in *TrSG*, for whom *SubjTrained* was set to "Tr," was asked to undergo training by using the User Requirements Classification Trainer before he or she performed the experiment. A brief description of the trainer is found in a document in the online appendix [28] for this paper. Each subject in *UnSG*, for whom *SubjTrained* was set to "Un," had not undergone this training. Since *TrSG* is disjoint from *UnSG*, *SubjTrained* is a between-subjects variable.

The independent variable about a user requirement specification is

1. the form of the specification, i.e., the representation of the user requirement, *URRepr*, with *URRepr* = "Or," meaning "**Or**iginal user requirement," i.e., that the user requirement from the Digital Villages project is represented in the standard Connextra user story template, and *URRepr* = "Te," meaning "**Te**mplated user requirement," i.e., that the user requirement is represented in the new user story **Te**mplate that forces providing the full information needed for classification.

In the second experiment, each of the twelve subjects was asked to classify 50 user requirements:

- 22 user requirements from the Digital Villages project, in their original representation, for which *URRepr* = "Or";
- the same 22 user requirements, rewritten with additional information as prescribed by the new template, for which *URRepr* = "Te"; and
- an additional 6 user requirements from the Digital Villages project, in their original representation, which happened not to include a benefit part, and which were not rewritten with the new template, for which *URRepr* = "Or."

Thus, all told, among the 50 user requirements classified, 28 had *URRepr* = "Or" and 22 had *URRepr* = "Te."

The reason for having exactly 22 user requirements that are written both in the standard Connextra user story template and in the new template is that there are two criteria[3] of PQ and nine criteria[4] of HQ for a total of eleven criteria. The set of user requirements consists of two user requirements

[3] (1) ease of use and (2) utility.

[4] (1) enablement of personal development, (2) identification, (3) symbolism, (4) attachment, (5) aesthetics, (6) luxuriousness, (7) trust, (8) physical comfort, and (9) freedom from risk.

for each criterion of the respective type of user requirements, i.e., of pragmatic requirements and hedonic requirements. So, we ended up with (11 criteria × 2 user requirements =) 22 user requirements to test. The fact that there are much more criteria for HQ than there are for PQ, leads to an imbalance in the numbers of pragmatic requirements and hedonic requirements within the set of user requirements used in the experiment. However, there is a balance in the number of user requirements for each of the eleven criteria of the two types of user requirements. A set of user requirements balanced in the numbers of hedonic and pragmatic user requirements would actually have been disadvantageous, because this balance would not have taken into account that hedonic user requirements are more diverse than pragmatic user requirements.

The additional 6 user requirements had been written with the Connextra user story template, but without the optional benefit part. It is not possible to decide if they are pragmatic or hedonic. It is, however, certain that none of them is neither pragmatic nor hedonic, because none of them focuses on an objective product quality; each is either pragmatic or hedonic. Therefore, we call these 6 user requirements *ambiguous*.

We identified and developed the concept of an ambiguous user requirement specification when we were analyzing the results of the first experiment and thinking about possible causes of the difficulties the first experiment revealed. Once we had identified the ambiguous user requirement specification, we realized that we had not taken and could not take them into account in the first experiment, and would have to do so in any subsequent experiment.

Recall that in the first experiment, we could not determine for each user requirement its correct classification, and we used consistency of classifications among subjects, as indicated by the Fleiss κ, interrater agreement, as a surrogate for correctness in the first experiment. Each of the newly templated user requirements, however, provides all the information necessary to correctly classify it as pragmatic or hedonic.

A templated user requirement provides a purpose, which must always be the modification of either a PQ or an HQ. The modification of a PQ or an HQ can be either the decrease of, the increase of, the adaptation of, or the enablement of the quality. Therefore, a templated user requirement cannot be neither pragmatic or hedonic. However, no such deduction was possible for the original user requirements from the Digital Villages project.

So, with the new templated requirements, the authors were able to evaluate the correctness of the classification by a gold standard we created with respect to pragmatic and hedonic user requirements. For the gold standard, the authors interpreted 22 of the 105 original user requirements and rewrote them using the new user story template. Afterward, each of them separately classified the 22 templated requirements as pragmatic and hedonic. After that, the authors compared their classifications. They discussed those that they had classified differently in order to agree on one classification. Thus, the gold standard represents a consensus between the authors as experts in RE and in user requirements classification. Appendix A in the online appendix [28] provides the gold standard.

Of the 22 templated user requirements, 4, 18%, are pragmatic and 18, 82%, are hedonic. Note that none of the templated user requirements are classified as neither. We can decide correctness of a classification for only the 22 templated and the 6 ambiguous user requirements. None of these 28 is classified as neither. Therefore, user requirements classified as neither are not considered in any discussion of correctness.

Even though we can measure correctness for 28 of the user requirements, because we cannot measure correctness for the 22 original user requirements, and we need to be able to compare second experiment data with first experiment data, we continue to evaluate *consistency* of classifications among subject. Also, we do not expect to achieve a 100% correct classification for any type of user requirements. Finally, evaluation of the consistency of classifications among the subjects allows identifying templated user requirements that are still difficult to be classified by many subjects.

Four possible confounding variables or threats had to be controlled in the second experiment.

1. The effect of a subject's experience in requirements analysis and in UX analysis, design, evaluation, or engineering was controlled by (1) first ranking the subjects by their average numbers of years of experience in these two areas, and (2) then assigning to the two different training conditions those with similar averages. Table 1 shows for each subject his or her experience and average numbers. The training condition for each subject is encoded in the middle two characters, either "Tr" or "Un," of his or her Subject ID.
2. In the hopes of understanding the between-experiments learning effect, we conducted a brief scripted interview of the 12 subjects before beginning the second experiment:

"After you took part in the last experiment, have you:

- reflected on the distinction of PQ and HQ?
- reflected on the definitions of PQ and HQ? [sic]
- informed yourself about PQ and HQ by reading literature on this topic?

Table 1 Matching of subjects according to their years of experience

Subject ID	Experience in user requirements analysis	Experience in UX analysis, design, evaluation, or engineering	Mean experience
STr1	1.50	1.00	1.25
SUn1	2.00	0.50	1.25
STr2	1.00	3.00	2.00
SUn2	2.00	2.00	2.00
STr3	2.00	2.00	2.00
SUn3	1.00	4.00	2.50
STr4	3.00	3.00	3.00
SUn4	4.00	3.50	3.75
STr5	4.00	4.00	4.00
SUn5	5.00	5.00	5.00
STr6	5.50	8.00	6.75
SUn6	7.00	14.00	10.5

- informed yourself about PQ and HQ by talking to experts on this topic?
- talked to other subjects of the last experiment? If so, have you talked about PQ and HQ in general? Have you talked about your classification of particular user requirements?
- Applied the definitions of PQ and HQ [sic] to classify user requirements? If so, have you applied the definitions in a real project?"

None of the twelve subjects considered, talked about, or applied the definitions between the two experiments. So we felt confident that any observed learning about classification would take place during the second experiment.

3. Recall that the 50 user requirements to be classified in the second experiment were built from 28 user requirements from the 105 user requirements from the Digital Villages project used in the first experiment. Of these 28, 22 were not ambiguous and six were ambiguous. Each of the 22 unambiguous user requirements was used twice in the second experiment, once it its original form from the first experiment and once rewritten in templated form. The potential bias in the selection of the 28 user requirements to be classified in the second experiment from the 105 classified in the first experiment was avoided by having the first author doing the selection from the 105 and having the second author doing the classification by himself. Each selected and classified user requirements was discussed by the two authors to determine its suitability for the second experiment, as described in the next paragraph. The selection, classification, and discussion continued until the two authors were able to agree on a set of 28 user requirements to be used in the second experiment.

   In the discussions, the classification results from the first experiment were used to identify user requirements that were easy to classify, namely those that all subjects classified correctly, and user requirements that were difficult to classify, namely those that no none subject classified correctly. Five easy-to- and 6 difficult-to-classify user requirements were selected from the 105 original user requirements in this way. The other half of the 22 user requirements were user requirements that the subjects had classified differently in the first experiment. Each of the six ambiguous user requirements had no benefit part. Four of the six ambiguous user requirements were classified in the first experiment as pragmatic by exactly half of the subjects and as hedonic by the other half of the subjects. The remaining 2 ambiguous user requirements were classified in the first experiment as pragmatic by all except one subject.
4. The possible within-experiment learning effect, arising from the fact that 22 of the user requirements appear in two different formats among the 50 user requirements, was controlled by having one half of the subjects of each grouping to start with classifying the original user requirements and the other half start with classifying the templated user requirements.

All in all, each of twelve RE and UX professionals was asked to serve as an analyst and to classify as pragmatic, hedonic, neither, or ambiguous, each of 50 real-life app user requirements that were represented as user stories. The task the subjects had to perform in the experiment was described in detail in a two-part questionnaire, described in Sect. 5.3, to be read and filled in by each subject. Half of the questionnaires gave the original user requirements before the templated user requirements and the other half

gave the templated user requirements before the original user requirements.

The dependent variables about a subject are:

- obtained during the first experiment,
  1. *ExURA*, the subject's experience in user requirements analysis; and
  2. *ExUX*, the subject's experience in UX analysis, design, evaluation, or engineering;
- obtained from Part I of the questionnaire,
  3. *ClsP*, the percentage of the 50 user requirements that the subject classified as pragmatic;
  4. *ClsH*, the percentage of the 50 user requirements that the subject classified as hedonic;
  5. *ClsN*, the percentage of the 50 user requirements that the subject classified as neither;
  6. *ClsA*, the percentage of the 50 user requirements that the subject classified as ambiguous;
  7. *CorrP*, the percentage of the pragmatic user requirements correctly classified as pragmatic;
  8. *CorrH*, the percentage of the hedonic user requirements correctly classified as hedonic;
  9. *CorrA*, the percentage of the ambiguous user requirements correctly classified as ambiguous[5]; and
  10. *LEDef*, the subject's level of expertise in terms of the definitions;
- obtained from Part II of the questionnaire,
  11. *DifP*, the subject's self-assessment of the difficulty of classifying templated user requirements as pragmatic;
  12. *DifH*, the subject's self-assessment of the difficulty of classifying templated user requirements as hedonic; and
  13. *DifA*, the subject's self-assessment of the difficulty of classifying original user requirements as ambiguous.

Each experience variable, *Ex...*, is measured in years. The possible values of each of *DifP*, *DifH*, and *DifA* are the values in a five-point scale, with "1," "2," "3," "4," and "5" meaning, respectively, "Very difficult," "Difficult," "Moderate," "Easy," and "Very easy."

[5] There is no *CorrN* variable, because correctness can be evaluated for only templated, pragmatic or hedonic, and ambiguous user requirements.

A number of groupings of subjects are considered. Two, i.e., *TrSG*, and *UnSG*, are based on the between-subjects independent variable *SubjTrained*. Intersecting these two groupings with the grouping of the user requirements imposed by the value of the user requirement independent variable *URRepr* allows building four additional relevant groupings:

1. *TrOrG*, the grouping of trained subjects classifying the original user requirements,
2. *TrTeG*, the grouping of trained subjects classifying the templated user requirements,
3. *UnOrG*, the grouping of untrained subjects classifying the original user requirements, and
4. *UnTeG*, the grouping of untrained subjects classifying the templated requirements.

Other groupings are emergent, based on commonality in the values of the subjects' dependent variables. It proved necessary to compute for each grouping G of subjects and for each classification C, the κ measuring the interrater agreement among the subjects in G for the user requirements with the classification C. Together, these κ values measure the degree to which the classifications of the subjects in G agree with each other.

5.2 Hypotheses

The four pairs of hypotheses, alternative and null, formulated for this experiment are:

HOr:Tr>Un: The classification of original specifications by trained subjects is significantly easier than the classification of original specifications by untrained subjects, thus comparing *TrOrG* with *UnOrG*.
HOr:Tr=Un: The classification of original specifications by trained subjects is of the same difficulty as the classification of original specifications by untrained subjects, thus comparing *TrOrG* with *UnOrG*.
HUn:Te>Or: The classification by untrained subjects of templated specifications is significantly easier than the classification by untrained subjects of original specifications, thus comparing *UnTeG* with *UnOrG*.
HUn:Te=Or: The classification by untrained subjects of templated specifications is of the same difficulty as the classification by untrained subjects of original specifications, thus comparing *UnTeG* with *UnOrG*.
HTe:Tr>Un: The classification of templated specifications by trained subjects is significantly easier than the classification of templated specifications by untrained subjects, thus comparing *TrTeG* with *UnTeG*.
HTe:Tr=Un: The classification of templated specifications by trained subjects is of the same difficulty as the

classification of templated specifications by untrained subjects, thus comparing *TrTeG* with *UnTeG*.
HTr:Te>Or: The classification by trained subjects of templated specifications is significantly easier than the classification by trained subjects of original specifications, thus comparing *TrTeG* with *TrOrG*.
HTr:Te=Or: The classification by trained subjects of templated specifications is of the same difficulty as the classification by trained subjects of original specifications, thus comparing *TrTeG* with *TrOrG*.

Each hypothesis's name is of the form

$$\mathrm{H}V{:}v_1 \square v_2$$

in which

- V is a value of *one* of the two independent variables, *SubjTrained* or *URRepr*,
- v_1 and v_2 are the two values of *the other* of the same two independent variables,
- $\square$ is a relation, "=," meaning "is of the same difficulty as," or ">," meaning "is significantly easier than."

$\mathrm{H}V{:}v_1 \square v_2$ hypothesizes that when condition V is held constant, then classification under condition v_1 is related as $\square$ to classification under condition v_2. Thus, $\mathrm{H}V{:}v_1 > v_2$ and $\mathrm{H}V{:}v_1 = v_2$ are one pair of hypotheses with $\mathrm{H}V{:}v_1 = v_2$ being the null hypothesis and $\mathrm{H}V{:}v_1 > v_2$ being an alternative hypothesis.

We chose these particular hypotheses because we believe that

1. the more a subject is trained in applying the definitions, and
2. the more information provided in the user requirement specification being classified,

the more likely the subject is able to classify the user requirements (1) easily, (2) correctly, and (3) consistently with other subjects.

The hypotheses are evaluated using the independent variables *SubjTrained* and *URRepr*, the dependent variables *LEDef*, *ClsP*, *ClsH*, *ClsN*, *ClsA*, *CorrP*, *CorrH*, *CorrA*, *DifP*, *DifH*, and *DifA*, and the κ values.

- That each of *CorrP*, *CorrH*, and *CorrA* is less than 85%, and that there is nonsignificant interrater agreement, with $\kappa \leq 0.6$, are together taken as an objective sign that classification is difficult for the subjects.
- That each of *CorrP*, *CorrH*, and *CorrA* is greater than or equal to 85%, and there is significant interrater agreement, with $\kappa > 0.6$, 0.6, are together taken as a literal sign that subjects' classifications are consistent with each other.
- That any of the variables *DifP*, *DifH*, and *DifA* is less than 4 is taken as a subjective sign that classification is difficult for the subjects.

A correctness score of 85% is taken as the minimum for substantial correctness because 85% is considered a B grade in the grade scale used by US colleges and universities.

The evaluation of HOr:Tr>Un and HOr:Tr=Un provides an answer to RQ3, while the evaluation of HUn:Te>Or and HUn:Te=Or provides an answer to RQ4. The evaluation of HTe:Tr>Un, HTe:Tr=Un, HTr:Te>Or, and HTr:Te=Or together provide evidence of the interaction of the effect of (1) subjects' training in applying the definitions and (2) the amount of information provided in the user requirements specifications being classified.

5.3 Procedure of the experiment

Before the experiment was conducted, the subjects were informed about the experiment and were asked to formally agree to participate in the experiment. The twelve subjects who agreed to participate were assigned to the *TrSG* and *UnSG*, as described in Sect. 5.1.

The questionnaire that was read and filled in by each subject consists of two parts and is found in the online appendix [22]:

1. In Part I, after an introduction to the topic of the experiment, the subject was asked to classify each of 50 user stories as pragmatic, hedonic, neither, or ambiguous. The subject was provided (1) the definitions, and (2) the instructions when to classify a requirement as pragmatic, hedonic, neither, or ambiguous based on the explicit provision of respective keywords and on the hierarchy of HQ over PQ over product quality; see the questionnaire including the instructions for the classification in a document in the online appendix [28]. A subject of *TrSG* was asked to use the trainer before he or she proceeded with Part I of the questionnaire. Before the subject was asked to begin with the actual classification, he or she was asked to provide his or her level of expertise in terms of the definitions of "PQ" and "HQ" on a five-point scale based on Bloom's taxonomy [30, 31], to provide the value of the *LEDef* variable.

   For the classification, the subject was provided an MS Excel file listing all user stories on two sheets, to install on his or her own notebook. Sheet 1 of this file contains all 28 original user requirements, including six ambiguous user requirements, and Sheet 2 contains all 22 templated requirements. The subject did the classification by modifying one of the classification columns of the

Table 2 Subjects' raw variable values and groupings' aggregate variable values

ID	*ExAvg*	*LEDef*	*ClsPO*	*ClsPT*	*ClsHO*	*ClsHT*	*ClsNO*	*ClsNT*	*ClsAO*	*ClsAT*	*DifP*	*DifH*	*DifA*
STr1	1.25	3	17.86	22.73	78.57	77.27	0	0	3.57	0	5	5	2
STr2	1.25	3	46.43	27.27	28.57	63.64	0	0	25	9.09	3	3	3
STr3	2	1	39.29	27.27	25	72.73	21.43	0	14.29	0	5	5	none
STr4	2	2	71.43	27.27	17.86	72.73	0	0	10.71	0	4	4	4
STr5	2	2	35.71	27.27	60.71	72.73	3.57	0	0	0	4	4	none
STr6	2.5	2	75.00	22.73	21.43	77.27	0	0	3.57	0	5	5	2
SUn1	3	3	53.57	45.45	42.86	54.55	0	0	3.57	0	4	4	3
SUn2	3.75	3	75	54.55	25	40.91	0	0	0	4.55	5	5	3
SUn3	4	2	57.14	31.82	39.29	63.64	3.57	4.55	0	0	4	4	none
SUn4	5	2	64.29	18.18	28.57	81.82	0	0	7.14	0	2	4	1
SUn5	6.75	3	64.29	50	32.14	50	0	0	3.57	0	5	4	3
SUn6	10.5	4	96.43	18.18	3.57	81.82	0	0	0	0	5	5	none
Agg *TTL*	3.67	2.5	58.04	31.06	33.63	67.43	2.38	0.38	5.95	1.14	4.5	4	3
SD *TTL*	2.58	0.76	20.26	11.71	19.09	12.56	5.89	1.26	7.19	2.71	0.92	0.62	0.86
Agg *TrSG*	3.17	2	47.62	25.76	38.69	72.73	4.17	0	9.52	1.52	4.5	4.5	2.5
SD *TrSG*	1.82	0.69	20.06	2.14	22.72	4.54	7.83	0	8.42	3.39	0.75	0.75	0.83
Agg *UnSG*	4.17	3	68.45	36.36	28.57	62.12	0.6	0.76	2.38	0.76	4	4	3
SD *UnSG*	3.08	0.69	14.2	14.61	12.71	15.45	1.33	1.7	2.66	1.7	1.07	0.47	0.87

Legend

ID Subject Identity Number ("STr" = "subject, trained" or "SUn" = "subject, untrained"; subject serial number = 1, 2, ..., 6, among those of his or her training condition)

ExAvg Mean number of years of experience in user requirements analysis and in UX elicitation, analysis, design, evaluation, or engineering

LEDef Subject's level of expertise with the definitions

ClsPO Percentage of original user requirements classified as pragmatic

ClsPT Percentage of templated user requirements classified as pragmatic

ClsHO Percentage of original user requirements classified as hedonic

ClsHT Percentage of templated user requirements classified as hedonic

ClsNO Percentage of original user requirements classified as neither

ClsNT Percentage of templated user requirements classified as neither

ClsAO Percentage of original user requirements classified as ambiguous

ClsAT Percentage of templated user requirements classified as ambiguous

DifP Difficulty of classifying templated user requirements as pragmatic (1 = "very difficult" ...5 = "very easy")

DifH Difficulty of classifying templated user requirements as hedonic (1 = "very difficult" ...5 = "very easy")

DifA Difficulty of classifying original user requirements as ambiguous (1 = "very difficult" ...5 = "very easy")

Agg Aggregate = median for *LEDef*, *DifP*, *DifH*, and *DifA* columns, = mean for all other columns

StD Standard deviation

TTL Total Grouping, i.e., all twelve subjects

TrSG Trained Subject Grouping, i.e., subjects used the trainer

UnSG Untrained Subject Grouping, i.e., subjects did not use the trainer

spread sheet, to provide the values of the *ClsP*, *ClsH*, *ClsN*, and *ClsA* variables. Additionally, the subject was asked to describe in the "comment" column, any issues, i.e., difficulties, he or she encountered during the classification.

2. In Part II, to get qualitative data, the subject was asked to evaluate and describe his or her experiences while doing the classification requested in Part I. Specifically, the subject was asked to rate with the five-point scale introduced in Sect. 5.1, the difficulty of classifying user stories as pragmatic, to rate the difficulty of classifying user stories as hedonic, and to rate the difficulty of classifying user stories as ambiguous, to provide the values of the *DifP*, *DifH*, and *DifA* variables. Table 2 shows the raw data for each variable for each subject and the aggregated value for each variable for each considered grouping of subjects, including the total grouping of all subjects.

Table 3 Correctness of the classifications of templated and the ambiguous user requirements

Subgrouping	*CorrT*		*CorrP*		*CorrH*		*CorrA*	
	Average	*P* value	Average	*P* value	Average	*P* value	Average	*P* value
All Subjects	80.68	0	83.33	0.000012	80.09	0.019	6.94	0.6466
UnSG	72.73	0	75	0.0972	72.22	0.217	5.56	0.856
TrSG	88.64	0	91.67	0.00003	87.96	0.04799	8.33	0.844
Most Experienced of All Subjects	84.85	0	83.33	0.00024	85.19	0.081	8.33	0.7166
Most Experienced *UnSG*	80.3	0.00045	75	0.1977	81.48	0.4469	11.11	0.9165
Most Experienced of *TrSG*	89.39	0	91.67	0.028	88.89	0.232	5.56	0.877
Most Inexperienced of All Subjects	76.52	0	83.33	0.09	75	0.2168	5.56	0.8797
Most Inexperienced of *UnSG*	65.15	0.0784	75	0.7024	62.96	0.618	0	0.982
Most Inexperienced of *TrSG*	87.88	0	91.67	0.0918	87.04	0.3	11.11	0.796

Then, in an attempt to learn the difficulties in learning and applying the definitions, the subject was asked to describe what he or she believed caused the ease or difficulty he or she encountered during the classification task in Part I.

The experiment took about 60 to 90 minutes per subject in the *TrSG*, and about 45 to 60 minutes for a subject in the *UnSG*.

To minimize interference factors, each subject performed the experiment procedure in the same calm office environment and was allowed to ask comprehension questions whenever he or she wanted to. Every subject was allowed to consult the definitions at any time.

No subject was paid for taking part in the experiment.

6 Analysis

After all subjects had completed the experiment, we began to analyze the data that the subjects provided. Free-form answers in the "comments" columns of the spread sheets were consolidated and grouped, i.e., clustered. Aggregate values of numerical dependent variables were determined by calculating the arithmetic mean for the values of every variable except for the variable *LEDef* and the three five-point difficulty ratings, *DifP*, *DifH*, and *DifA*. For the values of each exception variable, the aggregate values were determined by calculating the median.

6.1 Analysis of the classifications of user requirements

A subject's classification of a templated user requirement is *correct* if and only if the classification agrees with that of the gold standard for the user requirement, and a subject's classification of an ambiguous user requirement is *correct* if and only if the classification is "ambiguous." For a subject to be considered to be classifying *substantially correctly*, he or she has to have classified at least 85% of the 28 templated and ambiguous user requirements correctly. The *correctness of a subject's classifications* is the percentage of the 28 templated and ambiguous requirements he or she classified correctly. The *correctness of the classifications of a subgrouping of subjects* is the average of the correctness values of its subjects' classifications. A claim of *substantial correctness* of the classifications of the members of a subgrouping is reported only if the correctness of the classifications of the subgroup is at least 85% with a p value of less than 0.05.

As in the first experiment [19], the consistency of the classifications of a subgrouping of subjects is measured by computing κ on these classifications.

The analyses in this section consider correctness and consistency of the classifications of number of different subject subgroupings, each being defined by its members' sharing one particular configuration of values of independent and dependent variables. Some subgroupings, e.g., *UnSG* and *TrSG*, were introduced during the design of the second experiment; others emerged during the analyses. In these analyses, subgroupings may overlap.

6.1.1 Correctness

Table 3 shows the percentages of user requirements of several categories of user requirements that were classified correctly by the members of several subgroupings of subjects. The categories of user requirements are the templated, pragmatic, hedonic, and ambiguous user requirements, whose averages are reported in the *CorrT*, *CorrP*, *CorrH*, and *CorrA* columns, respectively. *CorrP*, *CorrH*, and *CorrA* are dependent variables described in Sect. 5.1. *CorrT*, the percentage of the templated user requirements classified correctly, is calculated as a weighted average of *CorrP* and *CorrH*, weighted by the ratios of pragmatic and hedonic user requirements among the templated user requirements. For each category *C*, there are three columns, reporting the category's average, the *CorrC* value and the corresponding p value.

The subgroupings, one per row, are (1) all subjects; (2) *UnSG*, the untrained subjects; (3) *TrSG*, the trained subjects; (4, 5, and 6) the most experienced of all subjects, of *UnSG*, and of *TrSG*; and (7, 8, and 9) the most inexperienced of all subjects, of *UnSG*, and of *TrSG*. The most experienced members of any subgrouping are *those* (i.e., possibly more than one) with *the one* highest average experience value, and the most inexperienced members of any subgrouping are *those* with *the one* lowest average experience value.

The 3 most experienced subjects who used the trainer achieved substantial correctness of classifications, and in fact, the highest percentages of correct classifications: 89.39%, among the 22 templated user requirements; 91.67%, among the 4 templated pragmatic user requirements; and 88.89%, among the 18 templated hedonic user requirements. However, even the 3 most inexperienced subjects who used the trainer achieved substantial correctness of classifications, and in fact, the same percentage of correct classifications: 91.67%, among the 4 templated pragmatic user requirements that the most experienced subjects did, and they achieved substantial correctness, 87.04%, among the 18 templated hedonic user requirements.

In general, among the subgroupings of all subjects in Table 3, only the most experienced of them achieved substantial correctness in classifying only hedonic templated user requirements. Among the subgroupings of the most inexperienced subjects, only those that used the trainer achieved substantial correctness in classifying templated user requirements of all three categories. Even among the subgroupings of the most experienced subjects, only those that used the trainer achieved substantial correctness in classifying templated user requirements of all three categories. Nevertheless, experience level by experience level, and templated user requirement category by category, subgroupings of subjects who used the trainer achieved higher correctness than those of subjects who did not use the trainer.

On the other hand, the correctness of the classifications of ambiguous user requirements was below our expectations for each subgrouping, even for subjects that used the trainer. As an aside, for each subgrouping, the subjects' self-assessments of the difficulty of classifying ambiguous requirements are only "moderate." Moreover, the most experienced subjects did not classify any templated user requirement as ambiguous, independently of the use of the trainer. Because none of the templated user requirements is ambiguous, this non-classification is correct. Also, none of the subjects in the *UnSG* and in the subgrouping of the most inexperienced of the *UnSG* classified any of the templated user requirements as ambiguous.

Finally, we consider the user requirements among the 22 original requirements that were classified as neither pragmatic nor hedonic. Because we do not know the correct classifications of any of the 22 original user requirements, we cannot measure the correctness of "neither" classifications. However, we do know that none of the most experienced subjects classified any of templated user requirements as neither, independently of the use of the trainer. Because none of the templated user requirements is classified as neither, this non-classification is correct. In addition, none of the subjects in the subgroupings of the most inexperienced subjects in the *TrSG* and of the most experienced subjects in the *UnSG* classified any of the templated user requirements as neither.

6.1.2 Consistency

With $p < 0.05$, there is substantial agreement among the subjects who used the trainer in the following subgroupings of subjects:

- subjects who categorized the 22 templated user requirements, $\kappa = 0.618$,
- subjects who categorized the 4 templated pragmatic user requirements, $\kappa = 0.636$,
- subjects who categorized the 18 templated hedonic user requirements, $\kappa = 0.649$,
- the most experienced subjects who categorized the 22 templated user requirements, $\kappa = 0.683$,
- the most experienced subjects who categorized the 4 templated pragmatic user requirements, $\kappa = 0.683$, and
- the most experienced subjects who categorized the 18 templated hedonic user requirements, $\kappa = 0.683$.

For no other subgrouping of subjects who used the trainer was substantial agreement achieved. Nevertheless, κ of the categorization of the 22 templated user requirements among the most inexperienced subjects is between 0.5 and 0.6.

The consistency among subjects who did not use the trainer is no better than or is not significantly better than the consistency reported in the first experiment among subjects who categorized the 22 original user requirements, independently of the experience of the subjects.

In addition,

- use of the trainer without templated user requirements does not lead to substantial agreement;
- templated user requirements without use of the trainer does not lead to substantial agreement; and
- only the combination of use of the trainer and categorizing templated user requirements leads to substantial agreement.

In the first two cases, without substantial agreement, the measured inconsistency is even higher for experienced subjects. However, lacking experience is compensated by use of the trainer: the consistency of the classifications of templated user requirements is higher for inexperienced subjects

who used the trainer than for experienced subjects who did not use the trainer.

The correctness of the classification of ambiguous requirements has already been observed to be very poor. In addition, the κ values for all of the examined subgroupings of subjects who classified the 6 ambiguous user requirements were quite low; in fact, no such subgrouping achieved substantial agreement. Among the subgrouping of the subjects who did not use the trainer, κ was even less than 0, which means "worse agreement than by chance."

Finally, we consider the user requirements among the 22 original requirements that were classified as neither pragmatic nor hedonic. The κ values for all of the examined subgroupings of subjects who classified any user requirements as neither were quite low. Here too, no such subgrouping achieved substantial agreement, and among the subgrouping of the subjects who did not use the trainer, κ was less than 0.

6.2 Difficulty of classifying user requirements

Subjects in *TrTeG* assessed the difficulty of classifying the user requirements that they classified as pragmatic or hedonic as slightly easier, namely between "easy" and "very easy," than did subjects in *UnTeG*, who assessed the same difficulty as "easy." In general, subjects in the union of *TrTeG* and *UnTeG* assessed the difficulty of classifying user requirements as easier than did subjects in the union of *TrOrG* and *UnOrG*. Specifically, in the first experiment, the subjects assessed the difficulty of classifying the original user requirements that they classified as pragmatic, as "moderate," but in the second experiment, the subjects assessed the difficulty of classifying the templated user requirements that they classified as pragmatic, as "easy" to "very easy" In the first experiment, the subjects assessed the difficulty of classifying the original user requirements that they classified as hedonic, as "difficult," but in the second experiment, the subjects assessed the difficulty of classifying the templated user requirements that they classified as hedonic, as "easy" to "very easy"

6.3 Analysis of the most difficult templated user requirements

By having analyzed the correctness of each subject's classification of each individual templated user requirement, we found that there was one templated user requirement that was classified incorrectly by each subgrouping, independent of the use of the trainer and of the template. Templated user requirement R1 was classified correctly by at most 66.67% of the subjects in the subgroupings and is thus regarded a difficult user requirement:

R1:
{Purpose} In order to increase the consistency of my shop system
{Product Quality} by a better availability of items,
{Perspective} as a seller,
{Need} I want to be able to be informed that a buyer ordered goods from me,
{Intention} to process the order.
For example, I have experienced
{Product Quality} a better availability of items
{Related Software Product} with a telecommunication system
{Functionality} that provided the possibility to be informed that a caller tried to talk to me,
{Implementation} implemented by a provision of the caller's number, the date of the call, and the possibility for the caller to leave a message.

R1 focuses on consistency, which is an aspect of trust, which is a criterion of HQ. However, ten subjects were of the opinion that the requirement is a basic functionality for using the system. According to these subjects, it is necessary for the seller to have sufficient information to process the order, so the user requirement is necessary to achieve the seller's goal of selling goods. Hence, the subjects classified the user requirement as pragmatic. The information within the main part of the user requirement that persuaded the subjects to a pragmatic classification is:

{Product Quality} by a better availability of items,
{Need} I want to be able to be informed that a buyer ordered goods from me,

This persuasion was strengthened by information from the experience part of the user requirement:

{Functionality} that provided the possibility to be informed that a caller tried to talk to me,
{Implementation} implemented by a provision of the caller's number, the date of the call, and the possibility for the caller to leave a message.

Two of the 10 subjects, after a brief discussion with the first author, admitted that the given purpose is hedonic. Nevertheless, for these 2 subjects the user requirement was rather pragmatic, because the provided purpose,

{Purpose} In order to increase the consistency of my shop system,

of increasing consistency is not hedonic to them.

Another subject commented that R1 simply does not sound like a hedonic user requirement. Yet another subject classified R1 as pragmatic, because it is about the informativeness of the system, and thus R1 increases the system's ease of use.

Subjects in *UnSG* misclassified two other templated user requirements, R2 and R3 below, while subjects in *TrSG* classified them correctly.

R2:
{Purpose} In order to decrease the expenditure of human resources when I order goods
{Product Quality} by a better adaptation to individual preferences,
{Perspective} as a recipient,
{Need} I want to be able to set which notifications I would like to receive,
{Intention} to be disturbed only in matters that are important to me.
For example, I have experienced
{Product Quality} a better adaptation to individual preferences
When I interacted
{Related Software Product} with a social media platform
{Functionality} that provided the possibility to set which notifications I would like to receive,
{Implementation} implemented by the possibility to set individually for each type of notification, which notifications I would like to receive.

R2 focuses on ease of use, which is an aspect of PQ, but 3 subjects indicated that the focus was on the line,

{Product Quality} a better adaptation to individual preferences,

which was regarded as a hedonic aspect. Another subject indicated that the notifications are pure reminders, but are not necessary to achieve the core tasks. According to this subject, R2 increases the user's well-being, because the user feels secure when the system supports him in not forgetting tasks.

Yet another subject concentrated on the given intention of R2,

{Intention} to be disturbed only in matters that are important to me.,

and thus classified R2 as hedonic.

R3:
{Purpose} In order to increase the provocation of memories when I give a textual feedback
{Product Quality} by a better item understandability,
{Perspective} as a user,
{Need} I want to be able to give a textual feedback,
{Intention} to report problems or the like.
For example, I have experienced
{Product Quality} a better item understandability
When I interacted
{Related Software Product} with a chat tool
{Functionality} that provided the possibility to give a textual feedback,
{Implementation} implemented by presenting the textual feedback in speech bubbles that reminded me of comic strips that I liked very much in my youth.

R3 focuses on the product's symbolism, which is an aspect of HQ, but 3 subjects focused on the phrase,

when I give a textual feedback,

which served as a trigger to the 3 subjects to classify R3 as pragmatic. One of these 3 subjects commented that the

{Product Quality} ... item understandability,

is an additional hint for classifying R3 as pragmatic. Two other subjects regarded R3 as a purely functional user requirement.

Another subject did not know why he or she classified R3 as pragmatic.

6.4 Qualitative feedback

The qualitative feedback on the difficulty of classifying user requirements shows that subjects regard the definition of "PQ" as clear and the classification of user requirements as pragmatic as easy, especially for templated user requirements. The decision trees featured in the trainer are considered very helpful for doing the classification, and subjects wished that they could have used them during the actual classification task, as well. On the other hand, some subjects commented that the templated user requirements were hard to read.

The same qualitative feedback shows that subjects regard also the definition of "HQ" as clear and the classification of user requirements as hedonic as easy, but some of the subjects believed that hedonic benefit parts were not very convincing and that HQ was hidden in the non-templated, original user requirements, so that HQ was not easy to recognize. Furthermore, some subjects found a functional necessity to implement an HQ, and then struggled to classify the user requirement because of its apparent pragmatism.

6.5 Influence of the use of the trainer

To test the influence of the use of the trainer, we applied the Mann–Whitney U test [32]. We needed to apply this test because a small sample size does not allow assuming a normal distribution. We used an ordinal scale for the subjective rating of the difficulty in classifying user requirements. Therefore, it was necessary to use the nonparametric Mann–Whitney U test for independent sample sizes.

Table 4 Mann–Whitney U tests of influence of using the trainer on the results

Pair of groupings	Difficulty	*U* value	Critical value at $p < 0.05$	*P* value
UnSG vs. *TrSG*	*DifP*	17.5	5	1
UnSG vs. *TrSG*	*DifH*	17	5	0.9362
UnSG vs. *TrSG*	*DifA*	12	2	1

Table 5 Mann–Whitney U tests of influence of using the templates on the results

Pair of Groupings	Difficulty	*U* value	Critical value at $p < 0.05$	*P* value
Exp. 1 vs. *Exp. 2*	*DifP*	40.5	37	0.0735
Exp. 1 vs. *Exp. 2*	*DifH*	9.5	37	0.00034

First, we applied the Mann–Whitney U test three times, once for the *DifP* variable, once for the *DifH* variable, and once for the *DifA* variable. The null hypothesis tested in the Mann–Whitney U test is: "The two samples come from the same population."

Table 4 shows that each of the Mann–Whitney U tests is not significant when a critical value with $p < 0.05$ is taken. Thus, each of the compared samples comes from the same population. Therefore, the use of the trainer does not influence the difficulty of classifying user requirements.

6.6 Influence of the use of the new template

By performing the Mann–Whitney U test on the first and second experiments' self-assessments of the difficulty in classifying pragmatic and hedonic user requirements, we learn if differences in the self-assessments of the difficulties are caused by the use of the template. For this test, the self-assessments by *TrTeG* and *UnTeG* were evaluated against the self-assessments provided in the first experiment.

Table 5 shows that the Mann–Whitney U test is not significant for the self-assessment of the difficulty in classifying pragmatic user requirements, but it is significant when for the self-assessment of the difficulty of classifying hedonic user requirements when a critical value with $p < 0.05$ is taken. Thus, the compared samples come from the same population with respect to the difficulty of classifying pragmatic user requirements, but the samples come from different populations with respect to the difficulty in classifying hedonic user requirements. Therefore, the use of the template does not influence the difficulty of classifying pragmatic user requirements, but it *does* influence the difficulty of classifying hedonic user requirements.

6.7 Summary of the results

All in all, the experiment reveals that each of having subjects exercise the trainer and of writing user requirements with the help of the template helps to make classifying user requirements easier, more correct, and more consistent. Having trained subjects classify templated user requirements is even more promising. The significant impact of the trainer shows evidence to expect that additional training and more experience in classifying user requirements as pragmatic and hedonic will even increase the easiness, correctness, and consistency of the classification results.

7 Statistical evaluation of the hypotheses

This section first gives criteria for rejecting a hypothesis and then proceeds to decide which hypotheses can be rejected by use of the data reported in Sect. 6.

We reject HOr:Tr=Un if

1. the median of at least one of *DifP*, *DifH*, and *DifA* is at least 4 for at least one of *TrOrG* and *UnOrG*, meaning that at least one difficulty is at least "easy," or if
2. the mean κ for at least one of *TrOrG* and *UnOrG* is greater than 0.6, meaning that there is substantial agreement among the subjects in at least one subgrouping.

We reject HUn:Te=Or if

1. the median of at least one of *DifP* and *DifH* is at least 4 for at least one of *UnTeG* and *UnOrG*, or if
2. the mean κ for at least one of *UnTeG* and *UnOrG* is greater than 0.6.

We reject HTe:Tr=Un if

1. the median of at least one of *DifP* and *DifH* is at least 4 for at least one of *TrTeG* and *UnTeG*, or if
2. the mean κ for at least one of *TrTeG* and *UnTeG* is greater than 0.6, or if
3. the mean of at least one of *CorrP* and *CorrH* for *TrTeG* and for *UnTeG* is at least 85%, meaning that least one subgrouping achieves substantial correctness.

We reject HTr:Te=Or if

1. the median of at least one of *DifP* and *DifH* is at least 4 for at least one of *TrTeG* and *TrOrG*, or if
2. the mean κ for at least one of *TrTeG* and *TrOrG* is greater than 0.6.

Now we apply these criteria to the required data from Tables 2 and 3.

For HOr:Tr=Un:

1. The median of *DifP* is 3, and the median of *DifH* is 2 for *TrOrG*; and
   the median of *DifP* is 4, and the median of *DifH* is 3 for *UnOrG*.
2. The mean κ is 0.181 for *TrOrG*, and
   the mean κ is 0.184 for *UnOrG*.

Therefore, HOr:Tr=Un cannot be rejected.

For HUn:Te=Or:

1. The median of *DifP* is 4.5, and the median of *DifH* is 4 for *UnTeG*; and
   the median of *DifP* is 4, and the median of *DifH* is 4 for *UnOrG*.
2. The mean κ is 0.307 for *UnTeG*, and
   the mean κ is 0.184 for *UnOrG*.

Therefore, HUn:Te=Or cannot be rejected.

For HTe:Tr=Un:

1. The median of *DifP* is 4.5, and the median of *DifH* is 4.5 for *TrTeG*; and
   the median of *DifP* is 4.5, and the median of *DifH* is 4 for *UnTeG*.
2. The mean κ is 0.618 for *TrTeG*, and
   the mean κ is 0.307 for *UnTeG*.
3. The mean of *CorrP* is 91.67%, and the mean of *CorrH* is 87.96% for *TrTeG*; and
   the mean of *CorrP* is 75%, and the mean of *CorrH* is 72.22% for *UnTeG*.

Therefore, HTe:Tr=Un is rejected.

For HTr:Te=Or:

1. The median of *DifP* is 4.5, and the median of *DifH* is 4.5 for *TrTeG*; and
   the median of *DifP* is 3, and the median of *DifH* is 2 for *TrOrG*.
2. The mean κ is 0.618 for *TrTeG*, and
   the mean κ is 0.181 for *TrOrG*.

Therefore, HTr:Te=Or is rejected.

All in all, the statistical analysis of the second experiment's results leads us to not reject HOr:Tr=Un and HUn:Te=Or but to reject HTe:Tr=Un and HTr:Te=Or.

8 Threats to the validity of the results of the experiment

To increase the construct validity of the experiment, real-life, representative user requirements were used in the experiment. So, the results of the experiment are assumed to be representative of what would happen with other user requirements, specified as user stories. In addition, the definitions were written carefully, and the subjects were asked to stick to the definitions as written in order to minimize the tendency for a subject to interpret the definitions in his or her own way.

The two experiments potentially suffer from mono-operation bias[6][32], because, in both experiments, the subjects classified only one set of user requirements, which was limited in size and limited to only one project's context. Nevertheless, we still believe that the results can be generalized because of the use of real user requirements and the representativeness of these user requirements. The experienced RE and UX professional subjects of the first experiment were asked about the representativeness of the 105 user requirements during the first experiment [19]. Not surprisingly, every subject regarded the 105 user requirements as representative, if for no other reason that they were taken from their own real-life project. The 50 user requirements used in the second experiment were a subset of the 105 user requirements used in the first experiment, so the user requirements of the second experiments are regarded as representative, as well.

With respect to internal validity, the different effects that were observed for the *TrSG* and the *UnSG* are regarded as being caused by the use of the trainer, because the provision of the trainer was the only difference between the *TrSG* and the *UnSG*. As described in the discussion of controlling threats in Sect. 5, selection bias was minimized by having each author select adequate user requirements separately and then having the two authors discuss the selected user requirements. The threat of participant error was minimized by having each subject conduct the experiment without any time restriction in a distraction-free office with his or her own notebook and with access to the definitions at any time.

[6] Wohlin et al. [32] define mono-operation bias as "If the experiment includes a single independent variable, case, subject or treatment, the experiment may under-represent the construct and thus not give the full picture of the theory. For example, if an inspection experiment is conducted with a single document as object, the cause construct is underrepresented."

Researcher bias was minimized, because the experiment conductor, who was the first author, was not part of the Digital Villages project, and he provided only minimal assistance to the subjects. In addition, standard measures were used to analyze the data. To exclude a within-experiment learning effect, one half of the subjects classified the original user requirements first, and the other half classified the templated user requirements first.

Language problems might have been introduced. Each of the project's user requirements was written in German, and we translated these user requirements into English for the experiment. Because each of the main translator (the first author) and the subjects is a native German speaker, interpretation of the user requirements might have been faulty. However, each of these people routinely uses English in his or her working environment, which housed the Digital Villages project and in which the experiments were conducted. Also, the second author, a native English speaker, helped to improve the translations.

We could not check the degree to which any subject complied with the guideline of using exactly the provided definitions.

The external validity of the results was helped by forming groups of experienced RE and UX professionals, whose jobs included classifying user requirements as pragmatic or hedonic. Thus, since the years of experience of the subjects ranged from 0.5 to 14 years, we expect that the experiment results can be generalized to other RE and UX professionals, independent of their experience, as long as the professionals have been gaining experience in requirements analysis or in UX analysis, design, evaluation, or engineering, for at least 0.5 years.

External and conclusion validity were enhanced by the assignment of subjects to the subgroupings *TrSG* and *UnSG* according to their years of experience in RE and UX to produce subgroupings with roughly the same mix of years of experience.

Nevertheless, there is a threat to external validity in the complexity of the new template. It is quite detailed and requires very careful application. Currently, we are not sure if it will be usable by real-life analysts when dealing with actual CBSs. Whether the new template will be used is probably a matter of the analyst's motivation. If an analyst *needs* to classify user requirements correctly, he or she will put in the effort to learn even something complex.

Conclusion validity is helped by the application of statistical tests to the data with p value of 0.05, but the small number of data might have prevented us from demonstrating additional relationships in the data.

9 Discussion and future work

The experiment presented in this paper revealed that when both the trainer and the new user story template were used, classifying user requirements as pragmatic and as hedonic became significantly easier. The experiment supports this conclusion by its having shown that two null hypotheses, HTe:Tr=Un and HTr:Te=Or, could be rejected. Therefore, given the direction of the rejection, there is strong evidence for their alternative hypotheses, HTe:Tr>Un and HTr:Te>Or.

Both quantitative data and qualitative feedback showed that the causes of difficulty in classifying user requirements actually are

- the analysts' lacking skill in applying the definitions of "PQ" and "HQ" and
- poorly written user requirements.

In particular, using the trainer seems to be effective for experienced analysts for identifying both PQ and HQ. Inexperienced analysts benefit from the use of the trainer only for identifying PQ, but not for identifying HQ. These results are valid only when templated user requirements were classified. Therefore, both the trainer and the new template have to be used in order to improve classification. In particular, by using both the trainer and the template, analysts classified significantly more user requirements correctly as pragmatic and significantly more user requirements correctly as hedonic than they had done in the first experiment. In addition, using the trainer facilitates the classification of user requirements by leading analysts to not classify user requirements as neither PQ nor HQ. The trainer seems to have no effect on the classification of ambiguous user requirements. Use of the new template leads to a significant decrease in the number of user requirements being classified as ambiguous.

Finally, the experiment showed that only two null hypotheses, HOr:Tr=Un and HUn:Te=Or, could not be rejected. So, we conclude that there is strong evidence that:

> HOr:Tr=Un The classification of original specifications by trained subjects is of the same difficulty as the classification of original specifications by untrained subjects and is thus independent of whether or not the subjects were trained in applying the definitions by using the trainer.
> HUn:Te=Or The classification by untrained subjects of templated specifications is of the same difficulty as the classification by untrained subjects of original specifications and is thus independent of whether or not the user

requirements were represented with the new user story template.

However, as mentioned, using *both* the trainer and the new template leads to a significant reduction in the subjects' assessments of the difficulty of classification, to a significant increase in interrater agreement about the classifications, and to a significant improvement in the correctness of classifications.

The experiment provides evidence that the most experienced subjects classify templated user requirements fairly well, but achieve substantial correctness only with training. These conclusions suggest that experience in RE and UX without training is not enough. Future studies will have to explore the effect of experience in RE and UX on classification more thoroughly.

The misclassifications of R2 and R3 were caused by a lack of skill in applying the definitions of "PQ" and "HQ." The subjects who classified R2 and R3 incorrectly did not focus on the purpose of the user requirements and interpreted them from their own perspectives. By applying the trainer, analysts learn that the expenditure of human resources in R2 is a PQ and that the provocation of memories in R3 is an HQ. Future work will have to evaluate how much training is needed to become skillful in reliably classifying user requirements correctly. Classifications from an analyst's gut feeling, such as that we found for R3, are expected to completely vanish with the use of the trainer.

R1 was found to be the most difficult user requirement independently of the use of the trainer. Increasing the consistency of a system as an aspect of trust seems to be a necessity of a system for users nowadays. That is, consistency has to be provided in order to enable a user to interact with a system. This makes consistency a hygienic factor, i.e., a PQ, and not an HQ anymore. However, consistency is an HQ according to the UX Quality Model, and analysts have to learn such specific difficulties in applying the definitions of "PQ" and "HQ." Future work has to identify other criteria of PQ and HQ that cause difficulties in classifying user requirements and to evaluate whether the trainer has to focus more attention on such difficulties.

We expect that RE and UX professionals that use both the new user story template and the trainer will be able to reliably classify pragmatic and hedonic user requirements correctly. That is, we expect that these RE and UX professionals will substantially agree on their classifications and that each will achieve substantial correctness on his or her classifications. We expect that after an RE or UX professional has been trained in applying the definitions of "PQ" and "HQ" with the trainer, he or she will be able to identify the focus of any templated user requirement. The use of the new template ensures that all information that is necessary for correct classification is included in a user requirement and is thus available to be applied as is taught in the trainer. If most RE and UX professionals classify correctly, they will certainly substantially agree in their classifications.

Nevertheless, an ambiguous, non-templated user requirement, which lacks a rationale, i.e., a benefit part, will still have to be interpreted. Even for an ambiguous user requirement, trained RE and UX professionals should substantially agree in their classifications, if for no other reason than they should end up making similar interpretations, guided by what they have been trained to notice about a user requirement: By using the trainer, an RE or UX professional gets a deeper understanding of the differences between PQ and HQ, so that even without an explicit expression of the purpose of a user requirement, he or she is more likely able to correctly guess the need that is addressed by the user requirement.

Some other future work is suggested by an observation by one of the anonymous reviewers of this paper. This reviewer noticed that the *DifA* values in Table 2 show that participants found that dealing with ambiguous user requirements was more difficult than dealing with the other kinds. However, the reality is that *none* of the templated user requirements and only six of the original user requirements that these participants classified were truly ambiguous. Thus, it appears that participants were classifying any challenging user requirement, which they did not really or fully understand, as *ambiguous*. Future work is needed to determine how prevalent is this kind of thinking. More generally, future work is needed to reveal if the perceived challenge of a classification task influences the classification.

The usability of the new template in practice will have to be evaluated as part of future work. The first author has already begun a usability study of the new template. The first results of this study show that analysts easily learn how to fill the template and that they regard the template as very useful for their daily work. Future work includes the development and evaluation of the effectiveness and usability of a tool for the creation of user requirements according to the new template. This tool is supposed to reduce the effort for learning how to use the new template.

10 General discussion

The results of this paper have three kinds of implications:

- for theory,
- for practice, and
- for teaching.

To use Shirley Gregor's taxonomy of theories [33], there are two types of theories that match our experiment:

Theory for explaining and predicting: Our experiments show that only the combination of the use of the new template and the trainer results in an increase of the goodness of the classification of user requirements, while neither the use of the template and the use of the trainer in isolation nor the use of more detailed definitions and project involvement lead to better classification. So, we *explain* what is needed to positively influence the classification, and since we show that several groups of analysts benefit from the use of both the template and the trainer, we are able to *predict* the same in general.

Theory for design and action: Based on the first theory, we propose a method that explains how to conduct an experiment or a real-life user requirements analysis in order to get reliable results. Not only did we *design* the new template and the trainer and show that the combination of both is necessary to get good classification results, we also showed what analysts have to *actively* learn in order to reliably classify user requirements. In addition, we provide *actionable* rules and a process for the correct classification of user requirements.

Space does not permit a fully detailed discussion of the practical impact of the results. Nevertheless, the first author has already begun work to evaluate the practical impact of the distinction between PQ and HQ on the daily work of RE and UX professionals. The first results of this evaluation reveal that this distinction makes analysts aware of the differences between PQ and HQ in interactive software products, that it helps analysts (1) to get a more detailed understanding of the real needs of the users of an interactive software product, (2) to specify more detailed UX requirements for these products, and (3) to work with UX.

This work has implications for software engineering education. Due to the increase of the importance of HQ for software engineering in recent years, teachers will have to teach both the theory of PQ and HQ and the practice of classification so that they become part of every software engineer's repertoire. Even if an individual software engineer does not classify user requirements and does not use the new template, increased awareness of the differences between PQ and HQ and increased skill in identifying PQ and HQ will enable the software engineer to more reliably build interactive software products that induce pleasure and thus, positive UX.

11 Conclusion

The results reported in this paper have by no means exhausted the data that we have collected. In particular, there are interesting observations about the differences in the distributions of the numbers of user requirements receiving each classifications in the two experiments, among the different subgroupings in these experiments, and in the gold standards. Sections 8 through 10 identify specific work that needs to be done to deal with threats and to answer some questions. These and a deeper study of the qualitative data await future work.

Regardless, the results of our experiments improve our understanding of hedonic user requirements. Perhaps we are on our way to achieving the goal of enabling software engineers to reliably develop products that induce a positive UX.

Acknowledgements The authors thank this paper's anonymous reviewers for RE'17 and for this special issue of *REJ*, Sebastian Adam, Joerg Doerr, and Andreas Jedlitschka for their comments on earlier drafts of this paper. Daniel Berry's work was supported in part by a Canadian NSERC grant NSERC-RGPIN227055-15.

References

1. Hassenzahl M (2004) The interplay of beauty, goodness, and usability in interactive products. Hum Comput Interact 19(4):319–349
2. Diefenbach S, Kolb N, Hassenzahl M (2014) The 'hedonic' in human–computer interaction: history, contributions, and future research directions. In: Proceedings of conference on designing interactive systems (DIS), pp 305–314
3. Hassenzahl M (2003) The thing and I: understanding the relationship between user and product. In: Blythe M, Overbeeke K, Monk A, Wright P (eds) Funology: from usability to enjoyment. Kluwer, Norwell, pp 31–42
4. Jordan P (2000) Designing pleasurable products: an introduction to the new human factors. CRC, London
5. Roto V, Law E, Vermeeren A, Hoonholt J (2011) User experience white paper. In: Demarcating user experience, 2011. http://www.allaboutux.org/uxwhitepaper
6. Ramos I, Berry D (2005) Is emotion relevant to requirements engineering? Requir Eng J 10(3):238–242
7. Ramos I, Berry DM, Carvalho JA (2005) Requirements engineering for organizational transformation. Inf Softw Technol 47(7):479–495
8. Thew S, Sutcliffe A (2008) Investigating the role of 'soft issues' in the RE process. In: Proceedings of IEEE international requirements engineering conference (RE), pp 63–66
9. Milne A, Maiden N (2012) Power and politics in requirements engineering: Embracing the dark side? Requir Eng J 17(2):83–98
10. Sutcliffe A, Rayson P, Bull CN, Sawyer P (2014) Discovering affect-laden requirements to achieve system acceptance. In: Proceedings of IEEE international requirements engineering conference (RE), pp 173–182
11. Hassenzahl M, Beu A, Burmester M (2001) Engineering joy. IEEE Softw 18:70–76
12. Hassenzahl M, Diefenbach S, Göritz A (2010) Needs, affect, and interactive products—facets of user experience. Interact Comput 22(5):353–362
13. Desmet P, Overbeeke C, Tax S (2001) Designing products with added emotional value: development and application of an approach for research through design. Des J 4(1):32–47
14. McCarthy J, Wright P (2004) Technology as experience. Interactions 11(5):42–43

15. Hassenzahl M, Tractinsky N (2006) User experience—a research agenda. Behav Inf Technol 25(2):91–97
16. Hassenzahl M (2008) User experience (UX): towards an experiential perspective on product quality. In: Proceedings of 20th international conference of association for Francophone d'Interaction Homme–Machine (IHM), pp. 11–15
17. Doerr J (2011) Elicitation of a complete set of non-functional requirements, Ph.D. dissertation, University of Kaiserslautern, Kaiserslautern, DE
18. Partala T, Kallinen A (2012) Understanding the most satisfying and unsatisfying user experiences: emotions, psychological needs, and context. Interact Comput 24(1):25–34
19. Maier A, Berry DM (2017) Improving the identification of hedonic quality in user requirements—a controlled experiment. In: Proceedings of IEEE international requirements engineering conference (RE), pp 205–214
20. Alliance A (2018) Glossary: role-feature-reason. https://www.agilealliance.org/glossary/role-feature/. Accessed 2 Apr 2018
21. Diefenbach S, Hassenzahl M (2011) The dilemma of the hedonic—appreciated, but hard to justify. Interact Comput 23(5):461–472
22. Maier A (2017) An experiment package for the evaluation of difficulties in the classification of user requirements that are provided as user stories, Fraunhofer IESE, Tech. Rep. Report No. 002.17/E. https://cs.uwaterloo.ca/~dberry/FTP_SITE/tech.reports/MaierBerryExperimentalMaterials/
23. Basili V, Caldiera G, Rombach H (2001) Goal question metric paradigm. In: Marciniak J (ed) Encyclopedia of software engineering, vol 1. Wiley, New York, pp 528–532
24. Jedlitschka A, Ciolkowski M, Pfahl D (2008) Reporting experiments in software engineering. In: Guide to advanced empirical software engineering. Springer, London, pp 201–228
25. Fraunhofer IESE, Project Digital Villages, 2017. https://www.iese.fraunhofer.de/en/innovation_trends/sra/digital-villages.html
26. Fleiss JL (1971) Measuring nominal scale agreement among many raters. Psycholog Bull 76(5):378–382
27. Landis J, Koch G (1977) The measurement of observer agreement for categorical data. Biometrics 33(1):159–174
28. Maier A, Berry DM (2017) Online appendix for improving the identification of hedonic quality in user requirements—two controlled experiments, University of Waterloo, Tech. Rep. Technical Report. https://cs.uwaterloo.ca/~dberry/FTP_SITE/tech.reports/MaierBerryOnlineAppendix/
29. Keppel G, Wickens T (2004) Design and analysis: a researchers handbook, 4th edn. Prentice Hall, Englewood Cliffs
30. Anderson L, Krathwohl D, Airasian P, Cruikshank K, Mayer R, Pintrich P, Raths J, Wittrock M (2001) A taxonomy for learning, teaching, and assessing: a revision of Bloom's taxonomy of educational objectives. Pearson, Allyn & Bacon, New York
31. Bloom B, Engelhart M, Furst E, Hill W, Krathwohl D (1956) Taxonomy of educational objectives, handbook I: the cognitive domain. Longmans, Green and Co, New York
32. Wohlin C, Runeson P, Höst M, Ohlsson MC, Regnell B, Wesslén A (2012) Experimentation in software engineering. Springer, Heidelberg
33. Gregor S (2002) A theory of theories in information systems. In: Gregor S, Hart D (eds) Information systems foundations: building the theoretical base. Australian National University, Canberra