Evaluating Presentation of Requirements Documents: Results of an Experiment

Yu-Cheng Tu, Ewan Tempero, and Clark Thomborson

Department of Computer Science, University of Auckland, Auckland, New Zealand

Abstract. There are diverse stakeholders for requirements documents in many development environments, and yet these requirements documents should be presented in such a way that all stakeholders will be able to engage them successfully. In order to produce effective requirements documents, analysts need guidance when developing new documents. They also need a convenient and accurate way to evaluate the effectiveness of existing documents. We have been exploring whether our three-factor measurement of document "transparency" would be useful in these ways. Our experimental results, presented in this article, support the hypothesis that transparency can be usefully characterised by accessibility, understandability, and relevance.

1 Introduction

There are many stakeholders for requirements documents, meaning presenting these documents in such a way that all will be able to engage them successfully is a challenge. We have been exploring how the concept of "transparency" can be used to evaluate the effectiveness of documents (or any artefacts) for a given set of stakeholders. We have developed a definition of transparency, and identified specific properties of a document that support or interfere with its transparency. We are now evaluating how well these ideas help assess the effectiveness of document. We have performed an experiment to answer two main questions — whether the general concept of transparency is useful for evaluating the effectiveness of documents, and whether our particular characterisation of transparency is useful. The first question has been answered in other discussions [1,2][1]. We present the results regarding the second question in this paper.

Our experiment compared two requirements documents that presented functional requirements to 10 software developers and 48 university students of computer science or software engineering. According to our definition (discussed in section 2.3), one of the documents was more transparent than the other. Our overall goal of the experiment was to determine which document was more effective. Our overall results (the first question above, and discussed in more detail in section 2.4) provided support that the document we had assessed as being more transparent was also the more effective. We also asked participants to comment on properties of transparency that we had identified, and more generally on what

[1] These are available from `http://goo.gl/sWgW2Q`

D. Zowghi and Z. Jin (Eds.): APRES 2014, CCIS 432, pp. 120–134, 2014.

they found helped or hindered engaging with the documents. It is the responses to these questions that we discuss in this paper.

The remainder of the paper is organised as follows. In the next section, we discuss the background and related work. In particular, we discuss issues with communication as they relate to documentation, and provide more details on our view of transparency. In section 3, we present the methodology used in our experiment, including detailing aspects of the requirements documents examined by the participants. Section 4 presents the results of the experiment. Finally section 5 presents our conclusions and discusses future work.

2 Background and Related Work

2.1 Communication through Documents

One challenge in requirements engineering is poor communication between users and developers. According to Bubenko [3], problems in communication are due to users not fully understanding the implications of the requirements presented to them by developers. Bubenko further says that current modelling methods for recording requirements are difficult for users to understand. He also says that developers have problems in analysing and determining the quality of a requirements specification. Similarly, Al-Rawas and Easterbrook [4] say that it is difficult to resolve misunderstanding of requirements between stakeholders using documentation, which is a one-way communication channel. One reason for misunderstanding is the unfamiliarity of the notations used to model requirements. Another reason is the use of terminology to communicate technical matters where one party cannot understand [5].

Poor communication can be a consequence of the presentation of requirements which in turn affects users' ability in comprehending requirements. For example, formal notations are useful for verifying the completeness of requirements, but they are often difficult for non-expert stakeholders to understand. Bubenko [3] mentions in the methodology challenges in requirements engineering that users often sign-off requirements specifications without fully understanding them.

We are interested in understanding how to evaluate presentation of requirements that affect communication in requirements engineering. Much focus in requirements engineering is on formal and semi-formal requirements specifications, however, Bubenko mentions that one of the challenges is "to improve user – developer communication, much more than to expand the use of formal methods". Although the challenge of improving user – developer communication was raised almost 20 years ago, there seems to be little advice on the effect of different requirements presentations on user – developer communication. If we better understand how requirements presentation affected users and developers, then we will be able to produce more effective requirements documents with fewer communication problems.

2.2 Documentation in Requirements Engineering

In software engineering, documents are used as a medium to communicate information, ideas or feedback about a software system to stakeholders. Forward and Lethbridge [6] say that "documentation is an important tool for communication and should always serve a purpose". Likewise, requirements documents such as requirements specifications are also important to successful requirements engineering.

A requirements specification is a basis of communication among all stakeholders [7]. It is being used through the software life cycle, from systems procurement, development, and to implementation of a software system [3]. According to SWEBOK [8], requirements documentation is one key to the success of any requirements process. To be successful, requirements are examined by stakeholders to ensure that software engineers have defined the right system. The requirements for a software system should be understandable and usable by experts and non-expert stakeholders. Therefore, it is important that requirements notations and processes are appropriate for different stakeholders [9].

However, it is not easy to produce effective requirements documents for all stakeholders. According to Al-Rawas and Easterbrook [4], requirements documents are "poor substitute for interpersonal communication". This is mainly due to the fact that there are many stakeholders for requirements documents and they usually prefer using different notations. Not every stakeholder will be familiar with different notations. Al-Rawas and Easterbrook discover in their interviews with various software developers that 86% of their participants said their customers need additional explanation to understand the notations used for the requirements. They also discover that developer's time is wasted in interpreting raw natural language requirements where a diagram or a formal notation could be used to represent the requirements. Similarly, one challenge faced in requirements validation is the review of requirements documentation as stakeholders cannot review the document thoroughly and sign-off due to time pressure [10].

The effectiveness of requirements documents in helping stakeholders achieving their goals can be affected by different factors. For example, according to Davis et al. [11], there are 24 qualities of a quality software requirements specification (SRS). A quality SRS "contributes to successful, cost-effective creation of software that solves real user needs". Example qualities include unambiguous, complete, and correct information. Similarly, an SRS document can be evaluated with indicators such as size, readability, depth and text structure [8]. However, current research focuses on methodology and notations used for representing requirements [9]. There seems to be little advice as to how to evaluate how effective a document is, other than to have potential stakeholders to try to use it, and little advice as to how to create a document that stakeholders will be able to successfully engage with. We believe that the concept of transparency can help with both goals, and we discuss this concept in the next section.

2.3 Transparency in Software Engineering

The term "transparency" in many areas has the notion of information being visible or open to those with a stake in that information. This concept is important, especially, to organisations as it has effects on the success, reputation, and credibility of organisations [12]. Transparency can also be one of the criteria for evaluating the effectiveness of public participation because it enables the public to see the outcome of such participation [13,14]. In business ethics, transparency is an ethical principle which aims to enhance public acceptance as well as to demonstrate fairness of organisations in decision-making [15].

Similarly, transparency refers to the visibility of a product or a development process to stakeholders. Stakeholders can evaluate a software system or to make decisions based on the visible information. This term is used in many areas of software engineering [2]. These areas include information privacy [16,17], computer ethics [18,19,20], and agile development [21,22,23]. However, the different uses appear inconsistent and there is no clear definition.

Based on our literature review [2], we define transparency in software engineering as *the degree to which stakeholders can answer their questions by using the information they obtain about a software system during its life cycle.* In this definition, stakeholders refer to anyone involved in the development of a software system. Example stakeholders are software developers, project managers, clients, and end users.

While our definition explains what transparency means, it does not provide an easy means to establish what transparency there is. To this end, we have identified three characteristics that apply to transparency: to answer stakeholders' questions, information needs to be *accessible*, *understandable*, and *relevant.* Accessibility concerns the ability of stakeholders in obtaining information. We define accessibility as *the degree to which stakeholders can obtain information that they believe is likely to answer their questions easily.*

To decide if the information answers their questions, stakeholders must understand the meaning of the information. This characteristic of transparency, understandability, we define as *the degree to which the information obtained by stakeholders can be comprehended with prior knowledge.*

Relevance is concerned with how well stakeholders can answer their questions using the information. Relevance is defined as *the degree to which the information obtained by stakeholders answers their questions.*

As noted in the introduction, we performed the experiment to help answer the questions as to whether our overall definition of transparency is useful, and whether the characteristics we have identified help determine transparency as we have defined it. In this paper we present the results relating to the second question. First, we summarise the results of the first question.

2.4 Usefulness of Transparency

Our experiment presented two documents that we had assessed as having different levels of transparency to participants, and asked questions relating to participants'

ability to engage with them. Section 3 provides an overview of the experimental design and the materials. Regarding our first question, the usefulness of the concept of transparency, indicated that the document we evaluated as having better transparency was also the document that our participants were better able to engage with them. Participants spent less time and were able to answer more questions correctly with the document that we evaluated to be more transparent. Participants who used the more transparent document were also more confident about their answers. Full details of these results, their analysis and discussion, are available in discussions by Tu et al. [1,2] (see footnote 1).

3 Methodology

3.1 Experimental Design

The main research questions for this paper are:

1. Are the properties of accessibility, understandability, and relevance useful to reasoning about transparency?
   This question directly addresses our main goal of determining whether these characteristics are useful to establishing transparency.
2. What elements in requirements documents affect participants' ability in understanding requirements?
   This question helps us determine whether we have missed anything that may be relevant to determining the transparency of documents.

The experiment involves the use of two different types of requirements documents and a questionnaire (see footnote 1). The first requirements document is an actual requirements document (ReqSpec) which describes an integration of an accommodation management system and an identity management system for a particular organisation. The second document is a use case model (UCM), which we created using the information from the ReqSpec document. A questionnaire is also constructed for participants to answer questions about the documents.

The ReqSpec document was originally the requirements document for a system that has been implemented. The only modifications we have made are to anonymise it. It is written in natural language and in free text format. It does not follow any specific formats or standards.

The UCM document is created by extracting information from the ReqSpec document. We chose the use case model as the second requirements document type because it is used to capture functional requirements of a software system [24]. To construct use cases for our experiment, we follow the template guidelines by Anda et al. [25]. The template guidelines include a template for describing an actor and a template for describing a use case. Information is extracted from the ReqSpec document with minimal changes to the original text. All use cases are based on the original text of the ReqSpec document.

The questionnaire for our experiment contains 23 questions for participants from software industry and 26 questions in total for student participants. Some

of the questions are optional. The questionnaire is divided into three sections: Demographics, Part 1. Reviewing Functionality of a Software System, and Part 2. Overview of the Software Document. In the Demographics section, participants are asked to answer questions about themselves such as their roles in a software project, their degree and major.

The purpose of Part 1 is to help us to compare the effectiveness of the two requirements documents. In this section each of the participants are given one of the two documents described above and asked to answer questions based on the information provided in the document. They are also asked to write down problems if they could not answer the question rather than leave it blank. This section contains eight questions in total. The first question asks participants to write down the type of requirements documents that they receive at the start of the experimental session. Participants are then asked to record the time they start answering this section. The questions are organised in the order of ease in locating answers in the ReqSpec document. All questions, except for one, have specific answers found in the ReqSpec document and the UCM document. The last question asks participants to record the time when they finish answering this section of the questionnaire. There was a 40-minute time limit to this section.

In Part 2, we ask participants nine questions about their opinions on the requirements documents. We ask participants questions relating to the three attributes of transparency. These attributes are not explicitly stated in the questions, but rather are described in general terms.

Experimental Hypotheses. Our assessment of transparency suggests that the UCM document is better than the ReqSpec document [2]. On the basis of our assessment, we have the following hypotheses:

1. Accessibility
   - H_0 : There is no difference between the accessibility of UCM document and the accessibility of ReqSpec document.
   - H_a : UCM document has better accessibility than ReqSpec document.
2. Relevance
   - H_0 : There is no difference between the understandability of UCM document and the understandability of ReqSpec document.
   - H_a : UCM document has better understandability than ReqSpec document.
3. Understandability
   - H_0 : There is no difference between the relevance of UCM document and the relevance of ReqSpec document.
   - H_a : UCM document has better relevance than ReqSpec document.

3.2 Execution

We used convenience sampling for selecting potential participants in our experiment. Our experiment was a between-subject design in which each participant

was subject to only one treatment. Each participant read only one type of the requirements documents, either a ReqSpec document or a UCM document.

Participants' main tasks were to answer the questionnaire and to read the requirements documents given to them at the beginning of the experimental session. Participants were not required to read everything provided in the documents. They need to read only the parts that they think could help them to answer questions in Part 1 of the questionnaire.

Participants in our experiment could choose to participate either in-person or on-line. For the in-person experiment, the researcher handed participants the materials in printed copies. The researcher was present at all times during the experiment session to answer any questions. For the on-line experiment, the questionnaire was self-administered. Participants received the materials in PDF format as well a link to the web-based questionnaire via email. Participants completed the web-based questionnaire on their own.

4 Results

The responses were transcribed into spreadsheets. To perform statistical analysis, Likert scale responses were transformed into numerical values. For example, Likert items such as "Very poor, Poor, Satisfactory, Good, and Very Good" were transformed into 1, 2, 3, 4, and 5 respectively. We used the transformed values in parametric statistical tests such as t-tests, which according to Norman [26], could be used for Likert data without "coming to the wrong conclusion".

We labelled the comments made by participants in the questionnaire with codes. The codes were based on our definition of transparency as well as any interesting points that arose in the comments. We then identified themes from the codes and grouped the codes according to themes. Coding enabled us to identify any common patterns relating to transparency from the experiment.

4.1 Demographics

We recruited 10 software practitioners and 48 university students. There were 29 participants for each type of document. Of the 10 industry participants, four people have zero to four years of experience and six have five to nine years of experience working in the software industry. All industry participants reported that they held the role as developers in a software project at the time of the experiment. Some participants were also architects or requirements engineers.

47 student participants came from the University of Auckland. Of the 48 student participants, there were 21 graduate students and 27 undergraduate students. Of the 27 undergraduate students, 16 were specialising in Software Engineering. All of the undergraduate student participants were in their second year of study or above at the time of the experiment.

4.2 Transparency of Requirements Documents

Accessibility. In question P2Q5a (see footnote 1), we ask our participants about the accessibility of the requirements documents. We ask them how helpful

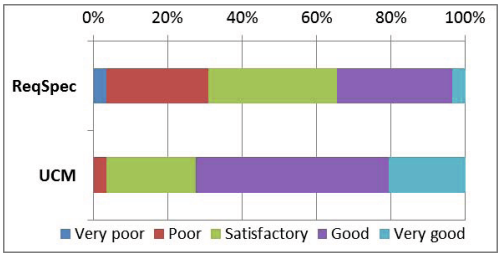

Fig. 1. Participants' assessments on how well the ReqSpec document and the UCM document were in helping participants to identify the desired information to answer questions in Part 1 (Accessibility)

the given document is to identify the information that they might need to answer questions in Part 1. Figure 1 shows participants' assessment of how well the documents were in helping participants to identify information. 10/29 participants using the ReqSpec document rated it good or very good whereas 21/29 participants using the UCM document rated it good or very good. Some participants also commented on how the documents helped them to identify information. For example, one of the comments from participants using the UCM document is "Contents & Use case diagram helped to identify the sections".

The mean values for the two treatments are 3.03 and 3.90 with standard deviations of 0.94 and 0.77 for the ReqSpec document and the UCM document respectively. The mean values suggest that the ReqSpec document and the UCM document were more than satisfactory for our participants on average.

To test whether the difference is statistically significant, we perform an independent-samples t-test. The t-test ($t = -3.81$, $df = 56$) indicates that the difference is statistically significant (0.000) at the 0.05 level of significance. The mean difference is -0.86, and the 95% confidence interval of the difference is -1.32 and -0.41. Further, Cohen's effect size value ($d = 1.01$) suggests a high practical significance.

The analysis shows that there is a difference in the accessibility of information using different requirements documents. Since the UCM document mean is greater than the ReqSpec document mean, the UCM document is better than the ReqSpec document in terms of helping participants to identify the desired information.

Understandability. In P2Q5c, we ask participants how helpful they think that the documents are to understand information and how well they think that they have understood the information in the documents. Figure 2 shows participants' assessments on the ReqSpec document and the UCM document in helping them to understand the functionality of the software system.

More than 60% of our participants reported that both documents were good or very good in helping them to understand the functionality of the software system. Two out of the 58 participants reported that the documents were poor.

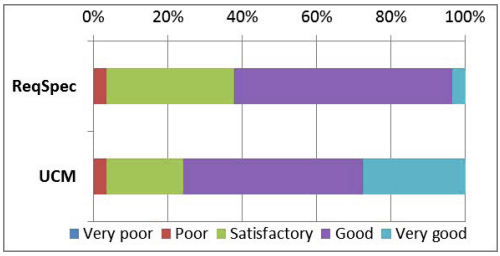

Fig. 2. Participants' assessments of the helpfulness of the ReqSpec document and the UCM document to understand the functionality of the software system (Understandability)

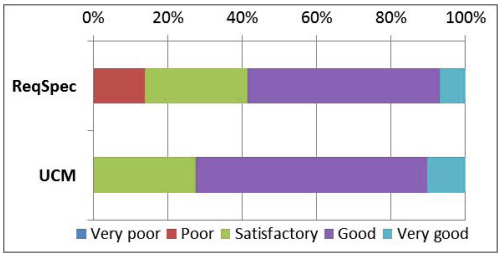

Fig. 3. Participants' self-assessments on how well they have understood the information provided in the ReqSpec document and the UCM document (Understandability)

The mean values for treatment ReqSpec and treatment UCM are 3.62 and 4.00 with standard deviations 0.62 and 0.80 respectively. The t-test gives some evidence against the existence of no difference between the means ($p = 0.049$). The t-value is -2.01 with 56 degrees of freedom. The mean difference is -0.38, and the 95% confidence interval of the difference is -0.76 and -0.002. Moreover, Cohen's effect value ($d = 0.53$) suggests a moderate practical significance. Since the mean for the UCM document is greater than the mean for the ReqSpec document, the UCM document is more helpful than the ReqSpec document in participants' understanding of the functionality of the software system.

In P2Q6a, we ask a similar question about how well participants think that they have understood the information provided in the documents. As shown in Figure 3, more than half of the 58 participants reported that they have a good or very good understanding of the documents. No participants who used the UCM document reported that they understood the information poorly. Four participants who used the ReqSpec document reported that they understood the information poorly.

The means are 3.52 and 3.83 with standard deviations 0.83 and 0.60 for treatment ReqSpec and treatment UCM respectively. The t-test shows a two-tailed p-value of 0.109 which suggests that there is no significant difference at the 0.05 level of significance ($t = -1.63$, $df = 51.09$, mean difference $= -0.31$, 95%

Table 1. Number of participants who either went through different parts of the requirements document to answer P1Q6 or not (Relevance)

	ReqSpec	UCM	Total
Yes	20	11	31
No	9	18	27
Total	29	29	58

confidence interval of difference = $-0.69, 0.07$). Cohen's effect value is 0.43 which suggests a low to moderate practical significance.

The statistical analysis shows that there is some evidence against the null hypothesis. The UCM document seems better than the ReqSpec document for the understandability of functional requirements. However, the statistical analysis for P2Q6a shows no significant difference in the understandability of information using the ReqSpec document and the UCM document by our participants. The mean values from P2Q5c and P2Q6a indicate that both documents were more than satisfactory in helping participants to understand information.

Relevance. In Part 2 of the questionnaire, we ask two questions about the relevance of information. We first ask participants in P2Q4 whether they have to go through different parts of the requirements documents in order to answer P1Q6. P2Q4 enables us to evaluate the sufficiency of the information at a particular location to answer the questions. If the information is insufficient, participants are likely to try and look for another location in the document.

Table 1 shows the number of participants who either went through different parts of the document or not. It appears that there were more participants who went through different parts of the ReqSpec document than participants who went through the UCM document to answer P1Q6. The observed proportion of yes to no for participants using the ReqSpec document is 0.69:0.31, whereas the proportion of yes to no for participants using the UCM document is 0.38:0.62. We compare the two proportions by using Fisher's exact test. The null hypothesis for the test is that there is no difference between the two proportions. We get a two-tailed p-value of 0.03 which is significant at the 0.05 level. Therefore, we reject the null hypothesis. This supports the existence of a difference between participants using different documents to answer P1Q6. In addition, the Phi coefficient of association ($\phi = -0.31$) suggests a weak negative association.

We also ask participants in P2Q5b to rate how helpful they think that the documents are to read only the relevant information to answer questions in Part 1. Figure 4 shows the distribution of participants' assessments on the requirements documents in P2Q5b. Participants using the ReqSpec document seem to have varied opinions about the document. Approximately 80% of participants using the UCM document reported the UCM document was good or very good in reading relevant information.

The means for the responses by participants using the ReqSpec document and participants using the UCM document are 2.97 and 3.76 with standard

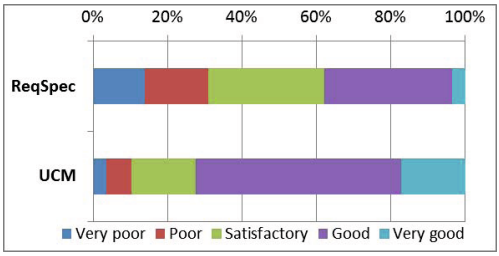

Fig. 4. Participants' assessments on how well the ReqSpec document and the UCM document were in helping participants to read only the relevant information that they needed to answer each question in Part 1 (Relevance)

deviations of 1.12 and 0.95 respectively. We find that the two-tailed p-value is 0.005 from the t-test which is less than the 0.05 level of significance ($t = -2.91$, $df = 56$, mean difference $= -0.79$, 95% confidence interval $= -1.34, -0.25$). Hence, we reject the null hypothesis. This indicates that there is a significant difference in the relevance of information in the ReqSpec document and the UCM document. Furthermore, Cohen's effect value ($d = 0.76$) suggests a moderate to high practical significance.

The analysis shows that the UCM document provides more relevant information than the ReqSpec document. Fewer participants who used the UCM document went through different parts of the document than participants who used the ReqSpec document. Participants who used the UCM document tended to be more satisfied with the relevant information than participants who used the ReqSpec document.

4.3 Elements that Affect Transparency of Requirements Documents

Accessibility. We find several themes that affect the accessibility of requirements documents. We find that the organisations of the ReqSpec document and the UCM document have positive and negative effects on the accessibility of information. For example, participants found the use case diagram, the document structure, and the table of contents in the UCM document helpful in locating information. On the other hand, participants using the ReqSpec document commented that headings and sections of the document needed to be improved. An index and an appendix could be included in the ReqSpec document to improve participants' locating information.

Another theme that arises is the format of the document. 55 participants were given physical copies of the ReqSpec document and the UCM document in the experiment. Participants were required to find information in the document manually, which in turn could take more effort than searching for information electronically. A few of our participants made that observation. One participant also commented that his or her "ability to manually search text has diminished" because he or she became used to finding information on a computer. This

suggests that information in electronic format could help to improve accessibility of information.

We also find different factors that hinder participants in locating information within the ReqSpec document or the UCM document. For example, participants using the ReqSpec document found similar information was distributed throughout the document, and as a result they were confused when trying to locate specific information. Some participants using the ReqSpec document also commented that the table of contents was not helpful for finding information or that the navigation of the document was not easy. Similarly, one participant using the UCM document mentioned that he or she needed to "... refer back and forth..."

Among the comments made by participants using the ReqSpec document, there is a common theme regarding time. Out of all 58 participants, five participants who used the ReqSpec document noted that they could not locate the information after spending 10 minutes or a long time on each question of Part 1 of the questionnaire. However, we did not find any participants who used the UCM document commenting that they spent more than 10 minutes on each question. Similarly, at least 10 of the 29 participants using the ReqSpec document mentioned that they needed to look through the document to answer questions whereas no participants using the UCM document made that comment.

Understandability. We identify two main themes that are related to the understandability. The first is related to how the ReqSpec document and the UCM document affect the understandability of information. Participants who used the UCM document said, the use case diagram was useful in helping them to understand the functionality of the system. However, a few of the participants who used the UCM document suggested that the use case diagram was insufficient. More diagrams such as workflow diagrams could improve understanding of the system's functionality. Participants using the ReqSpec document also suggested including use case diagrams as well as diagrams such as sequence diagrams in the document to help readers understand the system. Similarly, participants using the ReqSpec document and participants using the UCM document suggested that using pictures or illustrations helps in understanding.

The second theme is related to different factors that hinder participants' understanding of the information. A few of our participants commented that they needed more time to understand the information presented, particularly in the ReqSpec document. Similarly, the terminology and abbreviations used in the ReqSpec document and the UCM document were not easy for two of our participants. Another factor that hindered the participants' understanding of information is the confusing nature of the information in the ReqSpec document. Of the 58 participants, 4 participants commented that the information was confusing.

Relevance. We find three main themes that affect how relevant receivers thought the information was to answer the questions. 26 participants commented that they could not answer questions sufficiently using the requirements documents.

Participants commented on problems such as missing detailed information in the documents. Participants also commented on the information in the documents being unclear which also affected their ability in understanding information.

The second theme that we find is related to participants having too much information which might affect the time that they spend on answering their questions. Several participants using the ReqSpec document reported that there was too much text to read in the document. There were also two participants using the UCM document who reported that the use cases were long. Furthermore, there were concerns about over-documentation and long documents which could cause participants to spend too much time on documenting or reading irrelevant information.

Another theme is related to the problem of finding relevant information by our participants. This theme comes mainly from the responses made by participants to P1Q7. There was no clear or specific information in either document to answer P1Q7. 23 participants commented that they could not find the information at the expected location to answer P1Q7. For example, one of the participants who used the ReqSpec document reported that he or she "looked in section 3 page 27 because contents suggested data requirements but did not find relevant information." Similarly, a participant who used the UCM document answered P1Q7 with the comment: "... not seen relevant information on page 17. Neither for the Use Case Diagram on page 4." Based on such comments, we find that the information presented in the documents could be irrelevant for answering questions.

4.4 Limitations

This experiment is limited with respect to generalisation of the results to other types of requirements documents. The experiment only compares two types of requirements documents for presenting functional requirements of a software system. The results might be different if the questions for the requirements documents were different. For example, the UCM document could be less relevant to participants than the ReqSpec document if we asked questions relating to non-functional requirements of the software system.

This experiment is also limited to the type of participants we recruited. For the purpose of the experiment, we limited our participants to software developers and university students who have some experience using different software artefacts. The results might be different if non-expert stakeholders were involved in the experiment. Non-expert stakeholders such as clients might be unfamiliar with notations or language used in the requirements documents. They might also engage with the documents differently to expert stakeholders.

There are also threats to validity of this experiment. For example, the questionnaire might favour the UCM document. The questions might contain keywords that only appeared in the UCM document. To mitigate this threat, we first created the UCM document using the information provided in the ReqSpec document. We then created the questions based on the ReqSpec document. Full details of the threats and mitigations are available in other discussions [1,2].

Although there are limitations in our experiment, we have made progress towards answering our research questions. We have collected evidence to support our hypotheses about the usefulness of accessibility, understandability, and relevance to characterise transparency. We present our conclusions and discuss future work in the following section.

5 Conclusions

From the first part of our experiment we knew that the UCM document was to be the more effective document. In the second part of the experiment which we reported on here, we are interested in establishing why. The results of the experiment suggest that our characterisation of transparency with the properties of accessibility, understandability, and relevance provides a useful means to determine the transparency of requirements documents. We are able to identify the differences in the presentation of requirements documents with the three properties of transparency. In the experiment, we find that the UCM document is better than the ReqSpec document in terms of accessibility and relevance. We find that both documents are more than satisfactory for our participants to understand information about the functional requirements of a software system.

We also discover different elements in the requirements documents that have affected our participants' ability in understanding requirements. Elements such as document organisations, table of contents, and diagrams have both positive and negative effects on the transparency of information in requirements documents.

The results of the experiment give us confidence to continue investigating the concept of transparency, that it is a fruitful area for future research in software engineering. In this paper, we demonstrate the evaluation of two types of requirements documents with software developers and university students. As future work, we can apply the three properties of transparency to compare other types of documents as well as diagramming notations with different types of stakeholders.

References

1. Tu, Y., Tempero, E., Thomborson, C.: Evaluating transparency of requirements documents (March 2014) (unpublished manuscript)
2. Tu, Y.: Transparency in Software Engineering. PhD thesis, University of Auckland, New Zealand, Thesis under examination (2013)
3. Bubenko, J.A.: Challenges in requirements engineering. In: Second IEEE International Symposium on Requirements Engineering (1995)
4. Al-Rawas, A., Easterbrook, S.: Communication problems in requirements engineering: A field study. In: Professional Awareness in Software Engineering (1996)
5. Saiedian, H., Dale, R.: Requirements engineering: Making the connection between the software developer and customer. Inform. Software Tech. 42(6) (2000)
6. Forward, A., Lethbridge, T.C.: The relevance of software documentation, tools and technologies: A survey. In: ACM Symposium on Document Engineering (2002)

7. Leffingwell, D., Widrig, D.: Managing Software Requirements: A Unified Approach. Addison-Wesley Professional (2000)
8. Abran, A., Bourque, P.: SWEBOK: Guide to the Software Engineering Body of Knowledge. IEEE Computer Society (2004)
9. Cheng, B.H.C., Atlee, J.M.: Current and future research directions in requirements engineering. In: Lyytinen, K., Loucopoulos, P., Mylopoulos, J., Robinson, B. (eds.) Design Requirements Engineering. LNBIP, vol. 14, pp. 11–43. Springer, Heidelberg (2009)
10. Hansen, S., Berente, N., Lyytinen, K.: Requirements in the 21st century: Current practice and emerging trends. In: Lyytinen, K., Loucopoulos, P., Mylopoulos, J., Robinson, B. (eds.) Design Requirements Engineering. LNBIP, vol. 14, pp. 44–87. Springer, Heidelberg (2009)
11. Davis, A., Overmyer, S., Jordan, K., Caruso, J., Dandashi, F., Dinh, A., Kincaid, G., Ledeboer, G., Reynolds, P., Sitaram, P., Ta, A., Theofanos, M.: Identifying and measuring quality in a software requirements specification. In: IEEE First International Software Metrics Symposium (1993)
12. Oliver, R.: What is transparency?. McGraw-Hill (2004)
13. Bickerstaff, K., Tolley, R., Walker, G.: Transport planning and participation: The rhetoric and realities of public involvement. J. Transp. Geogr. 10(1) (2002)
14. Rowe, G., Frewer, L.: Public participation methods: A framework for evaluation. Science, Technology, & Human Values 25(1) (2000)
15. Vaccaro, A., Madsen, P.: Transparency in business and society: Introduction to the special issue. Ethics and Information Technology 11(2), 101–103 (2009)
16. Clarke, R.: Internet privacy concerns confirm the case for intervention. Communications of the ACM 42(2) (February 1999)
17. Awad, N., Krishnan, M.: The personalization privacy paradox: An empirical evaluation of information transparency and the willingness to be profiled online for personalization. MIS Quarterly 30(1) (2006)
18. Santana, A., Wood, D.: Transparency and social responsibility issues for wikipedia. Ethics and Information Technology 11 (2009)
19. Fleischmann, K., Wallace, W.: A covenant with transparency: Opening the black box of models. Communications of the ACM 48(5) (May 2005)
20. Fleischmann, K., Wallace, W.: Ensuring transparency in computational modeling. Communications of the ACM 52(3) (March 2009)
21. Ingalls, P., Frever, T.: Growing an agile culture from value seeds. In: Agile Conference, AGILE 2009 (August 2009)
22. Bird, C.: Top 10 tips for better agile. Information Professional 2(6) (2005)
23. Schwaber, K., Sutherland, J.: The scrum guide (July 2012), http://www.scrum.org/Portals/0/Documents/Scrum%20Guides/Scrum_Guide.pdf
24. Fowler, M.: UML distilled: A brief guide to the standard object modeling language. Addison-Wesley Professional (2004)
25. Anda, B., Sjøberg, D., Jørgensen, M.: Quality and understandability of use case models. In: Lindskov Knudsen, J. (ed.) ECOOP 2001. LNCS, vol. 2072, pp. 402–428. Springer, Heidelberg (2001)
26. Norman, G.: Likert scales, levels of measurement and the "laws" of statistics. Advances in Health Sciences Education 15(5) (2010)