ANALYZING THE FACTORS IMPACTING THE CHOICE OF THE FACT-FINDING TECHNIQUE FOR REQUIREMENTS ELICITATION

Abdullah A H Alzahrani

Computer Science Department, Computing College Alqunguda,
Umm Al Qura University
e-mail: aahzhrani@uqu.edu.sa
Saudi Arabia

Abstract: Requirements Engineering is an important stage of the software engineering development process. Since the 1990s, most IT projects failures have been attached to requirements. This highlights the importance of Requirements Engineering and in particular requirement elicitation. Researchers have investigated a number of aspects with related to requirement elicitation such as factors affecting gathering requirements, the effectiveness of fact-finding techniques, and the ways of representing the requirements. However, an insufficient amount of work has been done on linking the effectiveness of fact-finding techniques with the factors that affect requirements gathering. The purpose of this paper is to study the linkage and investigate the appropriateness of the commonly used fact-finding techniques. The methodology used in this paper is surveying the views of software engineers and analysts in the IT industry.

Key words: Requirement engineering, Fact-finding techniques, requirements analysis, requirements elicitation.

1. INTRODUCTION

Systems engineering is critical and Requirements Engineering is an important stage of that overall process [1]. Requirements engineering (RE) is regarded as the process of defining, documenting and maintaining requirements, which are the needs of stakeholders [2]. Sommerville [3] believes that depending on the type and size of application as well as the culture of the organization, requirement engineering processes might vary. However, some processes are fundamental for all. Figure 1 illustrates the 6 processes and the relations between them.

The process of Requirement Engineering (RE) out of these is the first step of software development and thus according to [43], [46] remains the most important stage of the process of software development life cycle (SDLC).

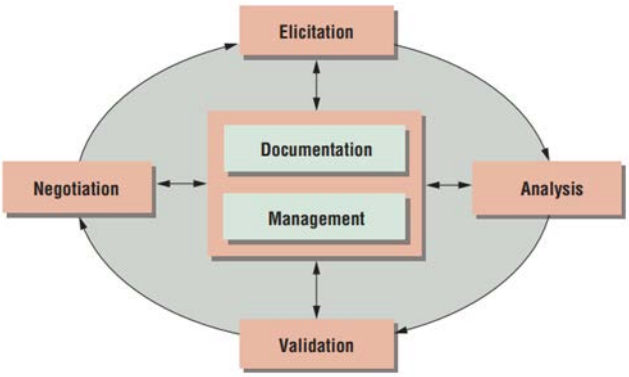

Figure 1. Requirements engineering processes cycle [3]

Requirements elicitation is the process of requirement engineering that converts the problem to concrete well-presented requirement specification by obtaining information from stakeholders and different sources [4 - 6]. Previous literature has studied and named techniques which can be used in requirements elicitation such as; Observation, Interviews, Protocol analysis, Brainstorming, Rapid prototyping, and Scenario [7, 8]. These techniques are referred to be the fact-finding techniques which help analysts and software engineers to obtain the requirements from stakeholders.

Although gathering information is one of the aspects of requirements engineering. There are a number of problems, which are encountered within the stage of requirements engineering. These problems vary from the information gathering to the representation of the requirements. Many researchers have studied the challenges of gathering information and the ways of overcoming these challenges [7-9, 14, 15, 44]. Other researchers investigated the problems of requirements representations such as ambiguity and suggested ways to overcome such problems [16-22].

Requirements stage is important in the development life cycle of any software. Paying more attention to have a clear set of requirements minimizes the possibility of project failure. Therefore, requirements engineering is not a cosmetic process for software development, in fact, it is a measure of success or failure of IT projects. Furthermore, since the 1990s, one of the top three reasons for project failure has been attributed to the inappropriate and unsuitable project requirements taken into account [10-13].

However, it is reasonable to question which technique is the best to conduct requirements elicitation. As it is one of the complex parts of Requirement Engineering phases, and thus often demands careful consideration. Particularly in cases of Global Software Development set-ups, so the probability of the failure can be decreased [39]. This motivated Davis et al. [9] to investigate the techniques in order to check the effectiveness of them on finding the right kind of requirement. The results concluded that taking interviews is the most popular technique, which can gather more information than other techniques. However, interviews are not the

only form which is used for the fact-finding technique. There are a variety of techniques used in the elicitation process of a single software system, making the selection of the effective fact-finding technique a challenging issue [41]. If the engineer, for instance, selects an inappropriate technique, based on inadequate factors such as personal references instead of the basis of stakeholder's characteristics or project characteristics can have negative effects on the quality of elicited requirements, and more so, on the development on the software as a whole [42]. Further, the selection of the fact-finding technique is based on a combination of factors for example; the analyst is only familiar with this technique, this is their favourite technique, a methodology is being followed which requires a specific technique or in some cases, the analysts even trust their instinct to guess the appropriate technique [40]. The decisions of the elicitation process, therefore, are found to be biased in many cases, resulting in a decreased quality of requirements elicited.

It is therefore clear from the above analysis, that an efficient approach towards selecting the factors which will influence the fact-finding tool of the elicitation process is significant for high-quality information collected. The similar notion motivated the study of [45] who observed evident differences between the fact-finding techniques of observations and questionnaires based on the fact that the systems were being made for children.

However, there is a lack of studies which focus their research on the appropriate factors, which impact the choice of fact-finding techniques in requirement elicitation, resulting in the development of successful software systems. Especially, there is insufficient empirical evidence on this topic of interest. These factors, as mentioned above, can be some general or software-specific elements which impact the choice of techniques to be employed [2]. Some researchers advocate that just one elicitation technique or tool is appropriate for all kinds of software and is applicable in all cases [24]. However, others argue that these factors are software-specific or project-specific since they are critical to the success of the elicitation process.

In a similar regard, previous research [26] has explained that the selection of specific elicitation techniques might employ the following factors;

a) The analyst is well familiar with the fact-finding technique being used.
b) Analyst mostly prefers this technique.
c) The technique is well aligned with the methodological approach taken to develop the software system
d) The technique is based on analyst intuition [26].

Clearly, a fact-finding technique in elicit requirement is chosen based on a variety of different factors depending on the analyst, and the software systems. In addition, the existing studies are not adequate enough to decide which set of factors are appropriate for choosing the technique of requirement elicitation.

Consequently, the study aims to investigate which set of factors are most likely to impact the choice of fact-finding technique in requirement elicitation for the successful development of software systems. This would help transfer expertise to

select effective factors, which help choose the right technique to give the right results. Furthermore, an expert system for such expertise could be developed. This research intends to bridge the gap and offer a linkage between the factors and the appropriate fact-finding techniques of requirements elicitation.

2. RELATED STUDIES

As mentioned earlier, software requirements are more commonly elicited through interviews. But software engineers try to use more than one elicitation technique to collect a full range of facts and requirements for effective development of their software systems. Thus, over a period of time, various elicitation techniques have been used, coming from the fields of cognitive psychology, sociology and linguistics. The techniques based on the field of linguistics is important as Noraini Che et al. [15] investigated the communication challenges between developer and customer in requirement elicitation. The authors suggested a model with intervention steps, which could help in overcoming issues such as incorrect information, ambiguous information, and frequently changed requirements. These issues are attributed to the communication challenges between developer and customer. Subsequently, using intervention steps could help in effective techniques used for requirements elicitation so an appropriate set of requirements can be achieved.

However, with the increasing advances and complexities in modern software designs and system usage, the requirements are complicated to understand and explain. That is why many studies [27, 28] have shown evidence to prove that interviews are no more a useful technique.

Carrizo et al. [29] discuss that elicitation techniques are of various forms and can be more effective for some systems than for others. Practically observed, when analysts try to identify the system software requirements, they commonly use only one technique, as discussed earlier which is interviews, although they have a little know-how of other methods [30]. This is probably the case when a software engineer is not fully aware of the benefits that each technique provides. Thus, there is no universal standard or methodology, which can help select the elicitation or fact-finding technique. In a lot of cases, an elicitation technique is selected, not on the grounds of its strengths or weaknesses but on the grounds of its common usage or familiarity with the analyst [31]

Arshad et al. [25] carried out a study to identify and determine the factors which affect the requirements gathering and discovery process and cause hindrance individually or collectively during the fact-finding process. The study was a qualitative study with 15 participants (3 are analysts) from Healthcare Knowledge Management Systems in Pakistan. The authors identified 11 factors affects the requirements elicitation. The authors investigated these factors from the perspective of occurrence and severity.

This research aims to identify the factors and link them with the software engineers' (in private and public sectors) appropriate choice of fact-finding

techniques. After reviewing the literature extensively, it has been affirmed that altogether only a few researchers have focused on the factors of the selection of elicitation techniques. Basically, as Carrizo et al. [32] has cited, that elicitation technique selection has been the major focus of only 10 studies. Based on the research [32] the factors which help to determine the suitable fact-finding technique are categorized into Elicitor, Informant, Problem Domain, Solution Domain and Elicitation Process. Figure 1 shows how each factor has different contextual attributes which can influence the choice of fact-finding technique for elicitation process

Factor	Attributes
Elicitor	Requirements engineering experience (elicitation) Technical knowledge of (training in) elicitation methods Knowledge of (familiarity with) domain Experience with elicitation methods (technique) Cognitive problems
Informant	Number of users (per session) Number of experts User involvement (interest) Location/accessibility Availability of time Expertise Cognitive styles (articulability) Personality variables Cognitive problems (consensus among informants) Cognitive skills
Problem domain	Type of phenomena Type of information (to be elicited) Type of heuristics Level of available information Domain fields Perceived structuredness Problem definedness
	Uncertainty Type of tasks Domain entities Confusedness Size Complexity
SD[a]	Product type Problem-solving methods
Elicitation process	Purpose of requirements Constraints (time) Process time Development methodology

[a] Solution domain.

Figure 2. Contextual attributes of factors. Source Carrizo et al. (2014) [29]

3. RESEARCH QUESTIONS

Based on the works of Carrizo et al. [29], the research intends to study the six attributes of the factors: Informant, Elicitor, Problem Domain, Solution Domain and Elicitation Process. Based on these attributes, which are considered as the proxies of the factors, the study outlines the following research questions:

Q1. Does the type of stakeholders or users of the software have an influence on the software engineers' choice of fact-finding techniques?

Q2. Is the status of availability of documentation of the system a factor, which influences software engineers' choice of fact-finding techniques?

Q3. Do systems meant to be used by the general public has an impact on software engineers' choice of fact-finding techniques?

From the research questions mentioned above, the following are the factors under investigation.

Factor (1). Number of stakeholders
Factor (2). Stakeholders knowledge of IT
Factor (3). Geographical Distance
Factor (4). LESS Availability of documentation
Factor (5). Conflicting Information
Factor (6). Public use

Many fact-finding techniques are introduced in the field of requirement engineering. Consequently, [24] have outlined and discussed a number of techniques which includes; Interviews, Questionnaires, Task Analysis, Domain Analysis, Introspection, Card Sorting, Laddering, Group Work, Brainstorming, JAD, Observations, Prototyping, Protocol Analysis, Goal-Based Approaches, Scenarios and Viewpoints, which help in the elicitation process. Based on the works of Zowghi and Coulin [24], the current research employs 5 out of them to investigate which factors (mentioned above) out of the six, influence the choice of selecting these techniques.in that case, these are:

T1. Sampling of existing documents, forms, databases and source files
T2. Interviews
T3. Observation
T4. Questionnaires
T5. Prototyping.

4. RESEARCH METHODOLOGY

After studying and researching the literature related to the requirement elicitation techniques, the researcher of the current paper outlined a survey questionnaire (See Appendices) to have an insight of the software analyst perspective, since previous researches in the similar field are based on the same methodological approach [32] or a semi-structured interview [25].

Basing the current study on the survey methodology is useful for gaining opinions from an extensive number of software engineers and analysts in the industry. The results of the survey, subsequently, can be generalized since it enables the researcher to employ a large sample size [33]. According to the study of Saunders et al. [34], survey analysis is also useful to quantify a large data set through inferential and descriptive statistics. On the other hand, semi-structured interviews

would have given more in-depth data but would have been hard to quantify. The survey questionnaire is also useful to analyse the results in the form of charts and compare percentages of the results. It is observed that surveys are a suitable method to measure the characteristics and assumption of different groups or opinions and attitudes towards a specific phenomenon [35]. In the context of the current study, the survey methodology is deemed appropriate as it helps to know the percentage of software engineers and analysts whose decisions get impacted by the aforementioned outlined factors. Finally, a survey questionnaire enables the researcher to easily administrate, control and generate data quickly as compared to other similar strategies such as semi-structured interviews.

The questionnaire is divided into three parts. The first section intended to give clear information and instructions to respondents as to what the survey is about and how their privacy is secured. The second section aims to obtain the consent of participation and to gather some general information, including personal information and academic qualification, work sector, Job title, and years of experience. The third part of the survey is a combination of questions' types, including MCQs, rate-scale questions, Likert-scale questions and open-ended questions. These questions are generated/adopted from previous research [2] to ensure the credibility and reliability of the instrument.

Following the methodological design, the study undertook convenience sampling as the sampling strategy for data collection. Convenience sampling, coming under the umbrella of non-probability sampling techniques enables the researcher to select the sample size while considering the factors of accessibility and proximity [34]. Based on this, the convenience sampling enabled the current study to collect data in a cost-effective and time-efficient manner from as many respondents as possible.

Grounded on this technique, the survey questionnaire was distributed online, via emails to 80 respondents out of which the study retained the results of 57 participants who responded with complete and clear answers. The participants of the study are software engineers and analysts, currently working in the industry of IT in both the public and private sector (i.e., 11% are working in the private sector and 29% are working in public sector). Out of these 57 participants, 86% are male and 14% are female, with the majority of 26 analysts who have a PhD qualification, 20 have an MSc qualification and the remaining 11 are BSc qualified. Further, 31 analysts and engineers work for international companies while only 14 are the ones working for local companies. After the educational background, the 57 respondents revealed at which job positions they are working on. Such as; the majority of the respondents are working in the academic field of software engineering, (i.e., 25 respondents), following the frequency of system's analysts (i.e., 11 respondents), and only 5 of the respondents are working as a systems engineer. The remaining 9 and 7 respondents are working as developers and software engineers. Lastly, majority of them have an experience of 1 to 5 years, and 6 to 10 years, both accounting for 21 respondents

each, along with only 4 respondents having an experience of over 20 years. While the remaining 11 shows experience of 11 to 15 years.

Accordingly, the response rate of the survey is measured to be 71.25%. The focus was to utilize expertise in order to gain better results. After collecting data from the participants, data is analysed and presented using statistical software to allow comparison of the results and formulation of visual graphs. More specifically, the analysis techniques including frequency analysis and graphical analysis.

In addition, the researcher aims to enable other researchers to access the collected data in case of any replications and analysis for future studies within a similar field. Further, the survey is conducted keeping ethical boundaries of respondents in mind and thus their identity is kept confidential. Also, this information was stored in a safe, password-protected device so that there is no risk of a privacy breach or data theft to the collected data.

5. RESULTS AND DISCUSSION

In this section, results from the survey and their interpretation are discussed. First, the main outcome of the study is considered in the light of the 6 factors outlined and their degree of influence in choosing the fact-finding techniques in the requirement analysis stage. Second, the results of the most common fact-finding techniques used in different work sectors (i.e., public or private) are compared to discuss the implications of the results. Third, the results of the appropriateness of use for each fact-finding technique with consideration of the influence of each factor is demonstrated through the results extracted. Finally, the main comparison showing the results of the appropriateness of use and the influence of each factor is identified and discussed.

5.1. Impact of factors

In Figure 3, a summary of all factors is shown that impact the system analyst and software engineer's choice of fact-finding technique, based on the respondents' perception. The most important factors, based on the given rating (0-5; not at all-extremely), are Factor (1) Number of stakeholders and Factor (2) Stakeholders knowledge of IT.

In addition, Figure 3 shows that Factor (4) i.e. Availability of documentation and Factor (5) Conflicting Information is also important. These factors can be related to the aforementioned factors (Factor 1 and 2) as information is provided by the stakeholders. This reflects the influence and the important role that stakeholders in the requirement analysis stage of the software system development life cycle.

Furthermore, it is apparent from Figure 3 that Factor (3) Geographical Distance has approximately equal influence as that of Factor (5) Conflicting Information when considering the choice of the fact-finding techniques in the requirement analysis stage. The figure also indicates a moderate influence of Factor (6) i.e. Public use.

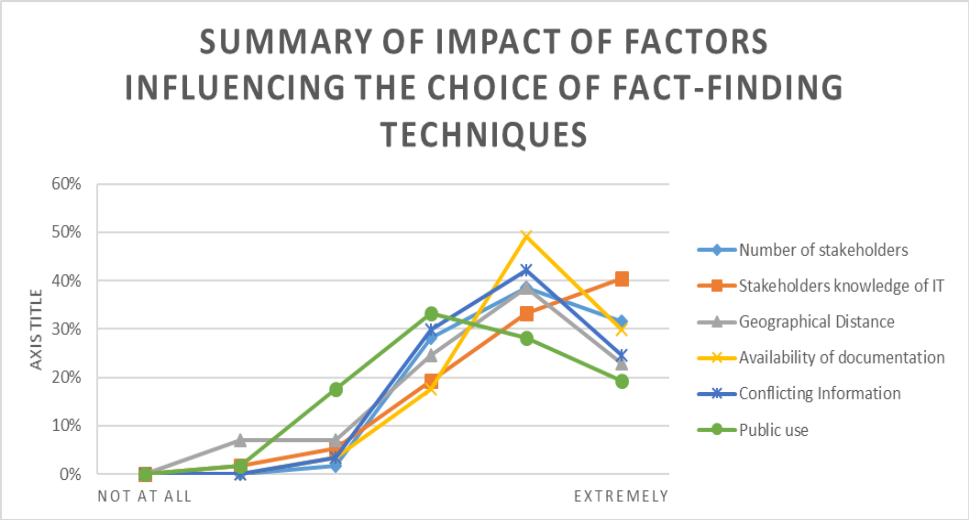

Figure 3. Summary of the impact of Factors influencing the choice of Fact-Finding Techniques

Furthermore, it is apparent from Figure 3 that Factor (3) Geographical Distance has approximately equal influence as that of Factor (5) Conflicting Information when considering the choice of the fact-finding techniques in the requirement analysis stage. The figure also indicates a moderate influence of Factor (6) i.e. Public use.

5.2. Frequency of use of fact-finding techniques

It is important, when investigating the factors of the choice of fact-finding techniques, to investigate the frequency of use of fact-finding techniques. Figure 4 summarizes the frequency of use of fact-finding techniques by software engineers and analyst in general. The fact-finding techniques considered in this study, as mentioned earlier, are: T1) Sampling of existing documents, forms, databases and source files, T2) Interviews, T3) Observation, T4) Questionnaires, and T5) Prototyping.

Figure 4 indicates that, apparently, T1 and T3 are used more often than other fact-finding techniques. Furthermore, T2 stands as the second more frequently technique used in requirement engineering. Finally, T4 and T5 come on the third place of the frequency of use of fact-finding techniques.

However, Figure 4 shows the frequency of use of fact-finding techniques in general. The question is that does Work Sector has an influence on this result. In order to answer this question, separation and re-analysis of data collected for the investigation of the frequency of use of fact-finding techniques are made. This is to check and compare public and private sectors frequency of use of fact-finding techniques.

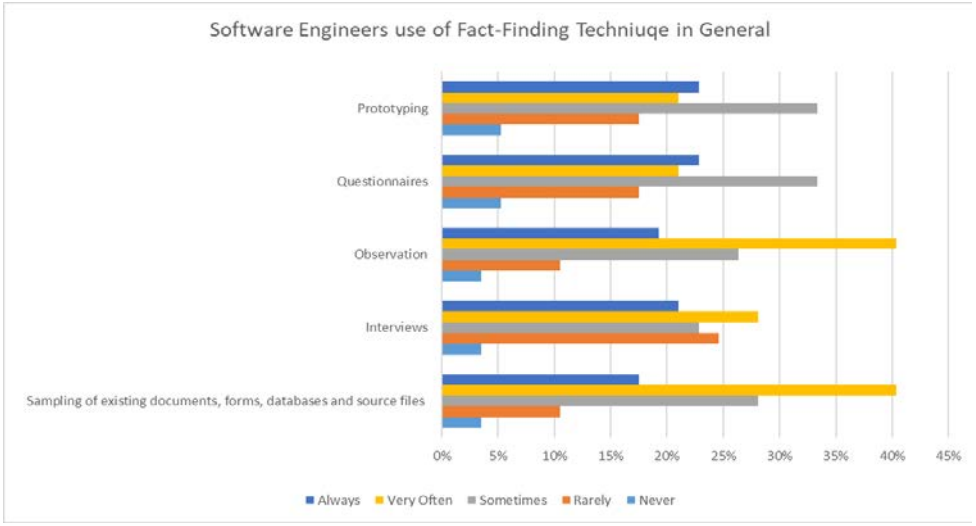

Figure 4. Software Engineers use of Fact-Finding Techniques in General

Surprisingly, as can be seen in Figure 5, it seems that the private sector tends to use T3 more often than any other fact-finding techniques. T1 and T2 come after T3 in the frequency of use. Furthermore, T5 has outstanding use with less frequency. However, this result cannot be generalized for the private sector as the participation from this sector in this study counts only 10%. Moreover, the results shown in Figure 6 highlights some contrast between the private sector and the public sector, public sector, where T1 and T3 appears to be most prominent similar to previous literature [36].

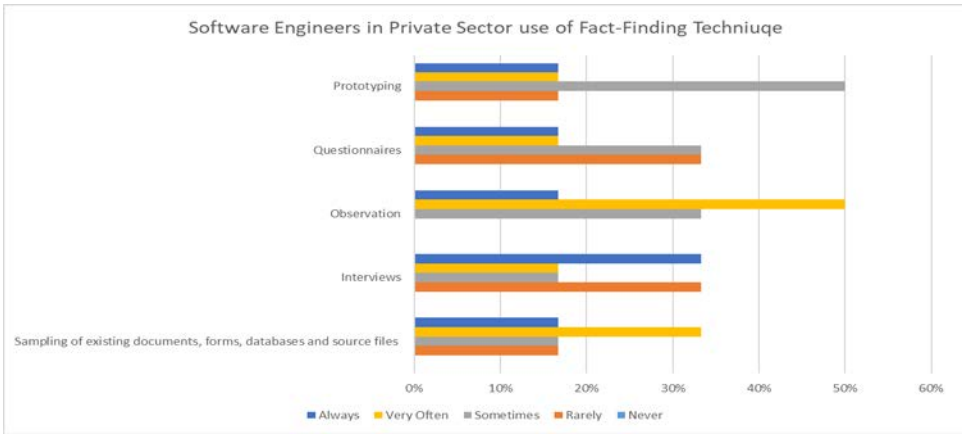

Figure 5. Software Engineers in Private Sector use of Fact-Finding Techniques based on the 5 Techniques investigated; T1) Sampling of existing documents, forms, databases and source files, T2) Interviews, T3) Observation, T4) Questionnaires, and T5) Prototyping

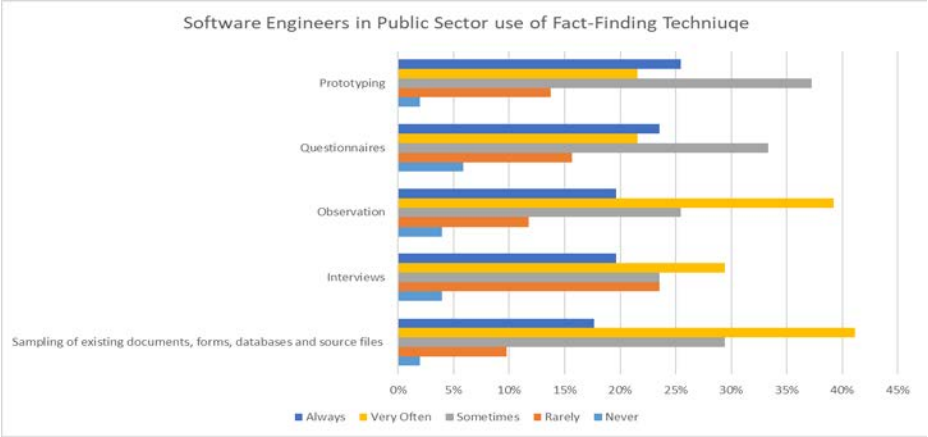

Figure 6. Software Engineers in Public Sector use of Fact-Finding Techniques based on the 5 techniques investigated: T1) Sampling of existing documents, forms, databases and source files, T2) Interviews, T3) Observation, T4) Questionnaires, and T5) Prototyping

5.3. Appropriateness of Fact-finding techniques

The third part of the questionnaire asked respondents, which fact-finding technique according to them is appropriate, based on the different factors. It studies the extent of influence of each factor on the software engineers' choice of fact-finding techniques. Next, each factor's implication accordingly is being discussed and the results are compared.

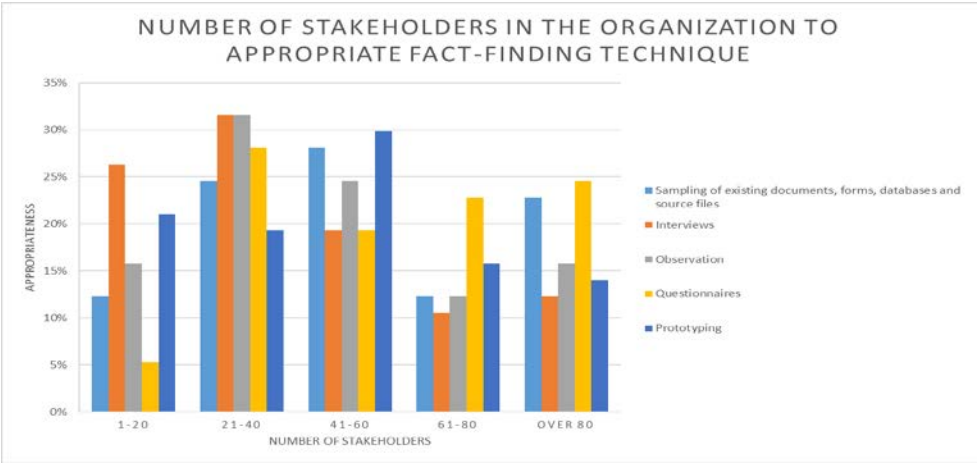

Figure 7. Choice of Engineers to Appropriate Fact-Finding technique based on Number of Stakeholders in the organisation

Figure 7 illustrates the appropriateness of fact-finding techniques (T1 – T5) against each category of a number of stakeholders, which has been divided into 5

groups. The range of each group is illustrated in the figure on the x-axis. This categorization helps in studying what implication does the number of stakeholders (i.e., Factor 1) has on the techniques used by software engineers and analysts. Results in Figure 7 and Table 1 shows differences of software engineers' choices when Factor (1) is considered.

As can be seen, software engineers tend to use T2 and T5 more than other fact-finding techniques when the number of stakeholders is between 1- 20 people. However, when the number of stakeholders grows (21- 40), software engineers prefer to use T2 and T3 in almost the same rate as shown in Table 1.

Table 1. Number of Stakeholders in the organization to appropriate Fact-Finding technique

Fact-Finding technique	Number of Stakeholders				
	1-20	21-40	41-60	61-80	Over 80
Sampling of existing documents, forms, databases and source files	12%	25%	28%	12%	23%
Interviews	26%	32%	19%	11%	12%
Observation	16%	32%	25%	12%	16%
Questionnaires	5%	28%	19%	23%	25%
Prototyping	21%	19%	30%	16%	14%

In addition, software engineers' preferences of techniques when the number of stakeholders is between 41- 60 are given to T5, T1, and T3 respectively. Furthermore, these options of fact-finding techniques seem to be narrowed as the number of stakeholders increases. The outstanding choice when the number of stakeholders is over 60 is T4. This obvious as the intent of this technique is to cover a large number in any field.

The results extracted from these descriptive stats and cross tabs, therefore, affirm that the number of stakeholders is particularly considered an important factor to imply which technique the software engineer should be using. This is determined as it can be observed with the number of stakeholder's range changing, the fact-finding technique is also changing. Hence conforming to previous literature [37], the number of stakeholders has a substantial implication on engineers' choice of elicitation process. Further, it is also exhibited that when the number of stakeholders was few, many engineers preferred interviews. With the increasing numbers, the technique changed to the questionnaire. Thus, validating to [27], who explained that interviews take a lot of time, and depending on the size of the respondent groups it's not always the suitable technique [38].

Factor (2) is Stakeholders knowledge of IT. This factor is directly related to stakeholders and their knowledge. The measurement here is 0 to 5, where 0 is not appropriate and 5 is the most appropriate technique when the stakeholders' IT knowledge is basic. Figure 8 demonstrates that software engineers prefer to conduct

more interviews and sampling of documentation (T2 and T1 respectively) when the stakeholders have no or low knowledge of Information Technology (IT). Through this, it can be implied that there actually exists a communication barrier between the stakeholders who have lack of knowledge about IT and software engineers [15], because of which interviews are seen as a better alternative to take into account for the RE process. Through interviews, intervention steps can, therefore, be taken to make the ambiguous questions, understandable to the stakeholders and get to know their requirements, even if it is in layman terms [24].

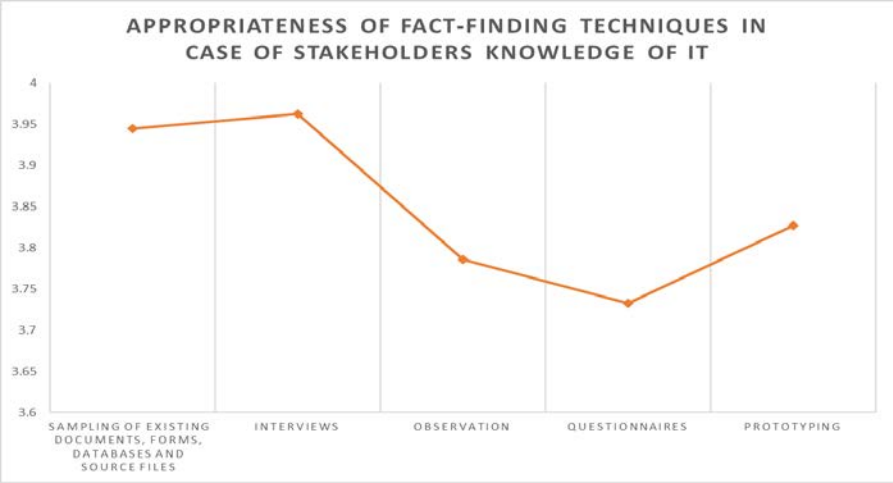

Figure 8. Appropriateness of Fact-finding techniques in case of Stakeholders knowledge of IT

Rendering to the changing RE techniques, the Geographical Distance Factor (3) seems to have different implications on the chosen fact-finding technique of the software engineer. This was observed through asking, that if there is a geographical distance, which technique is most appropriate (i.e., 5) to not appropriate at all (i.e., 0). Software engineers consider T1, T4 and T5 as the most appropriate techniques to be used with approximate 4 out of 5 ratings when the geographical distance is an issue. The geographical distance here is referred to both, between the branches of the organization whose system is to be developed or evolved as well as the geographical distance between stakeholders and software engineer. Again questionnaires, interviews and prototyping are seen to be on the same highest level, similar to [37] that these three techniques are useful for clear user feedback for refinement and a better understanding of the system developed.

Another important factor influencing the software engineers' choices of fact-finding techniques is Factor (4) Availability of documentation. Predictably, as shown in Figure 9, T1 is not gaining focus from software engineers as T1 is directly linked to the availability of documentation. Here the metric used for ratings in the questionnaire was 0 to 5, where 0 is not appropriate while 5 is most appropriate.

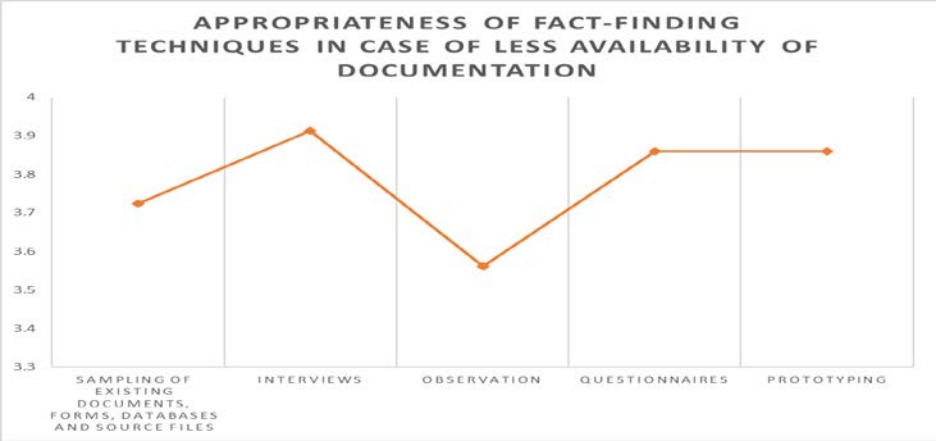

Figure 9. Appropriateness of Fact-finding techniques in case of LESS Availability of documentation

Consequently, software engineers seem to prefer to conduct more interviews in the requirement gathering and analysis stage. However, unexpectedly T3 (observation) seems not to be at a competitor level of T1, T4, and T5.

One of the factors considered in this research is Factor (5) Conflicting Information. In order to validate its importance, software engineers and analysts participating in this research were required to indicate the occurrence of cases where stakeholders providing conflicting information. It is observed that around 53% of software engineers encountered the case of conflicting information either always or very often, out of 5-metric scale ranging from Always to Never. In addition, 42% of software engineers indicate that they come across this case sometimes. However, none of them shows that conflicting information is something that has never occurred.

As a result, appropriateness of use for each fact-finding technique with consideration of the influence of conflicting information was investigated. Software engineers prefer to consider T1 as the most appropriate fact-finding technique in the presence of conflicting information. In addition, more interviews (T2) are to be conducted to overcome conflicting information. However, prototyping (T5) is a less appropriate technique for most of the software engineers and analysts participating in this study. This might be attributed to the time and effort that prototyping needs to overcome the conflicting information as explained in previous research [38]

Figure 10 illustrates the influence of Factor (6) Public Use on choices of software engineers of fact-finding techniques. As it is clear in Figure 10, when the system is intended to be used by the general public, software engineers participating in this study indicate that using T1 (sampling of documentation) is the most appropriate fact-finding technique. Furthermore, T3 (observation) comes as the second preferred technique in terms of appropriateness. This shows that the type of

usage of the software system has somewhat implication on the choice of RE processing of software engineers as investigated by Carrizo et al [29].

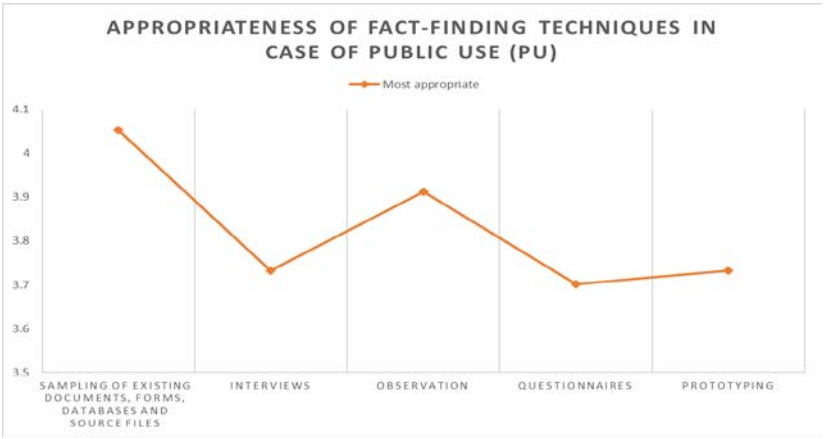

Figure 10. Appropriateness of Fact-finding techniques in case of Public use (PU)

5.4. Comparison of the appropriateness fact-finding techniques against factors

In this section, a summary of the appropriateness of Fact-Finding Techniques against different Factors (Factor 2 – 6) will be shown and discussed. This is done to allow the comparison of the appropriateness of Fact-Finding Techniques.

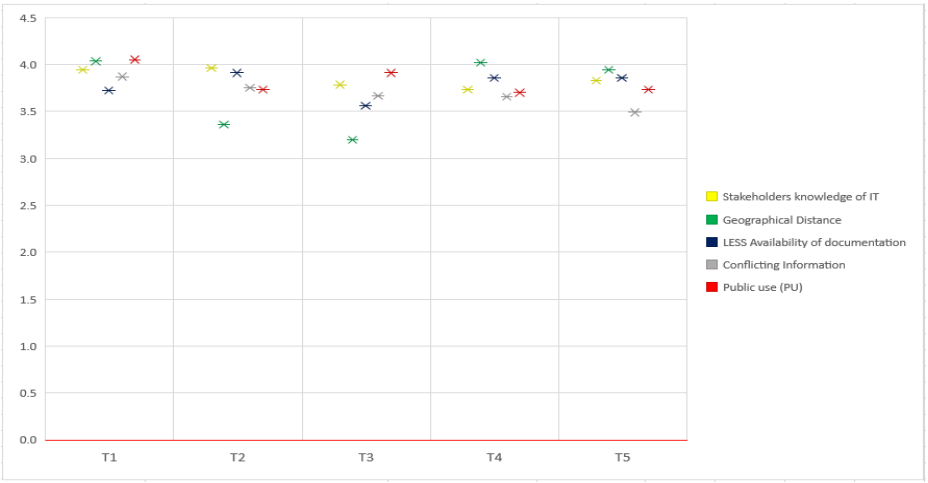

Figure 11. Summary of the appropriateness of Fact-Finding Techniques against different Factors

Figure 11 illustrates the summary of the appropriateness of Fact-Finding Techniques. The considered fact-finding techniques are in the vertical axis, whereas the horizontal axis shows the appropriateness of each of them on the scale of 0-5.

The series of values are the results of the appropriateness according to software engineers participating in this research.

Figure 11 and Table 2 indicate that T1 (Sampling of existing documents, forms, databases and source files) is most appropriate when all factors are existing except with factor 4 (less Availability of documentation). This might be due to the fact that factor 4 is affecting directly the inputs of T1. However, this does not hide the results that most software engineers prefer this fact-finding technique than other techniques.

Table 2. Appropriateness of Fact-Finding Techniques against different Factors

Fact-Finding techniq ue	Stakeholders knowledge of IT (Factor 2)	Geographical Distance (Factor 3)	LESS Availability of documentation (Factor 4)	Conflicting Information (Factor 5)	Public use (PU) (Factor 6)
T1	3.9	4.0	3.7	3.9	4.1
T2	4.0	3.4	3.9	3.8	3.7
T3	3.8	3.2	3.6	3.7	3.9
T4	3.7	4.0	3.9	3.7	3.7
T5	3.8	3.9	3.9	3.5	3.7

With regards to T2 (interviews), is more suitable when all factors are existing except with factor 3 (Geographical Distance). This is obvious as using T2 when factor 3 exists would introduce more expenses and enlarge the time frame especially with the organization has branches located distantly far from each other [23].

T3 (observation) seems to gain considerable attention from software engineers when, respectively, factor 6 (Public use) and factor 2 (Stakeholders knowledge of IT) are present. In addition, T3 is also preferable to be used with factor 4 and 5. However, factor 3 (geographical distance) seems to negatively affect software engineers' choice of T3. Although observations are known to be inexpensive methods to collect accurate and reliable data; it is difficult to conduct active observation with greater distance between the researcher and respondent [24].

However, it seems that factor 3 is positively influencing the software engineers' choice T4 (questionnaires). In addition, software engineers participating in this research indicate that T4 is also suitable when factor 4 exists (i.e., availability of documentation). This might be attributed to the possibility of using questionnaires results as a way to re-documenting the system. Also, it implies that questionnaires are successful with geographical distance, as they can reach to a larger number of people, within a short period of time, makes the research economical. It can also help collect huge amounts of rich data within a short period of time [37]. However, at the same time, at a geographical distance, questions can be misinterpreted, and useful feedback is not always collected [38].

Regarding T5 (prototyping), software engineers seem to prefer using it when factor 3 and 4 are affecting the processes of requirement engineering. However, T5

use is less preferred when factor 5 is present. This might be because of the possible difficulty and loss of effort, which might occur [9].

Finally, Table 3 shows only the highest score each fact-finding technique gained. For the purpose of having a better view of the result and comparing the results, only the highest score is displayed in Table 3. Whereas other scores were omitted. In addition, the score was replaced with an asterisk symbol (*). Consequently, Table 3 indicates that T1 is preferable, based on three factors, which are Factor 3, 5, and 6.

Furthermore, Table 3 demonstrates that three fact-finding techniques (T2, T4, and T5) are identically compatible with factor 4.

Table 3.Number of stakeholders in the organization to appropriate Fact-Finding Techniques

Fact-Finding technique	Stakeholders knowledge of IT (Factor 2)	Geographical Distance (Factor 3)	LESS Availability of documentation (Factor 4)	Conflicting Information (Factor 5)	Public use (PU) (Factor 6)
T1		■		■	■
T2	■		■		
T3					
T4		■	■		
T5			■		

6. CONCLUSION

In this research, factors affecting software engineers' choice of fact-finding techniques have been studied. The main findings are that the factors related to stakeholders and availability of documentation of the system have a considerable impact on the software engineers' choice of fact-finding techniques. However, a factor related to general public use of the system is less effective on the choice of requirements elicitation techniques.

In addition, surprisingly, T1 (Sampling of existing documents, forms, databases and source files) seems to outperform T1 (interviewing) as many software engineers and analysts stated in their answers. The reason behind the high importance is that availability of documentations gained as a factor is impacting the choice of requirements elicitation technique.

Furthermore, 53% of the participants articulated that they have encountered the case of conflicting information either always or very often. Therefore, conflicting information should be considered in further studies on the factors affecting requirement engineering and requirements elicitation.

6.1. Limitations

Considering the study is based on quantitative descriptive analysis, the results are expressed statistically through graphs and percentages. There is a lack of use of inferential statistics, and the study does not actually measure the impact's extent. Further, the results are limited to 57 participants, from the public and private sector, thus a larger sample size can be employed to make the results more generalizable. This is to gain further information and identify the differences between sectors.

Another constraint which limits the scope of the study is the factors and techniques taken into account by the study, as there are around 27 attributes in total based on previous literature [29].

6.2. Future work

In future work, an expert system to convey the expertise to new software engineers and analysts is considered to be developed and tested. This expert system could be the most beneficial to research the impact of a suitable choice of fact-finding techniques when different factors occur. Having such a system can produce qualitative research on factors affecting the choice of fact-finding techniques.

REFERENCES

[1] J. Dick, E. Hull, and K. Jackson, *Requirements engineering*. Springer, 2017.

[2] B. Nuseibeh, S. Easterbrook, Requirements engineering: a roadmap, *Proceedings of the Conference on the Future of Software Engineering*, 2000, pp. 35-46.

[3] I. Sommerville, Integrated requirements engineering: A tutorial, *IEEE Softw.*, vol. 22, no. 1, pp. 16–23, 2005.

[4] H. Van Vliet, *Software engineering: principles and practice*, 3rd ed. John Wiley and Sons, Ltd, 2008.

[5] I. Sommerville, P. Sawyer, Viewpoints: principles, problems and a practical approach to requirements engineering, *Ann. Softw. Eng.*, vol. 3, no. 1, pp. 101–130, 1997.

[6] A. Distanont, H. Haapasalo, M. Vaananen, J. Lehto, The engagement between knowledge transfer and requirements engineering, *Int. J. Manag. Knowl. Learn.*, vol. 1, no. 2, pp. 131–156, 2012.

[7] B. Davey and C. Cope, Requirements Elicitation–What's Missing?, *Issues Informing Sci. Inf. Technol.*, vol. 5, 2008.

[8] N. A. Maiden and G. Rugg, ACRE: selecting methods for requirements acquisition, *Softw. Eng. J.*, vol. 11, no. 3, pp. 183–192, 1996.

[9] A. Davis, O. Dieste, A. Hickey, N. Juristo, A. M. Moreno, Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic

review, *14th IEEE International Requirements Engineering Conference (RE'06)*, 2006, pp. 179–188.

[10] R. Stanley, L. Uden, Why projects fail, from the perspective of service science, *7th International Conference on Knowledge Management in Organizations: Service and Cloud Computing*, 2013, pp. 421–429.

[11] I. Attarzadeh, S. H. Ow, Project management practices: Success versus failure, *2008 International Symposium on Information Technology*, 2008, vol. 1, pp. 1–8.

[12] S. Group, *CHAOS manifesto 2013*, 2013.

[13] G. Stepanek, *Software project secrets*. Springer, 2005.

[14] W. J. Lloyd, M. B. Rosson, J. D. Arthur, Effectiveness of elicitation techniques in distributed requirements engineering, *Proceedings IEEE Joint International Conference on Requirements Engineering*, 2002, pp. 311–318.

[15] N. C. Pa, A. M. Zin, Requirement elicitation: identifying the communication challenges between developer and customer, *Int. J. New Comput. Archit. Their Appl.*, vol. 1, no. 2, pp. 371–383, 2011.

[16] H. Yang, A. De Roeck, V. Gervasi, A. Willis, B. Nuseibeh, Analysing anaphoric ambiguity in natural language requirements, *Requir. Eng.*, vol. 16, no. 3, p. 163, 2011.

[17] H. Yang, A. De Roeck, V. Gervasi, A. Willis, B. Nuseibeh, Extending nocuous ambiguity analysis for anaphora in natural language requirements, *2010 18th IEEE International Requirements Engineering Conference*, 2010, pp. 25–34.

[18] A. Cimatti, M. Roveri, A. Susi, S. Tonetta, Formalizing requirements with object models and temporal constraints, *Softw. Syst. Model.*, vol. 10, no. 2, pp. 147–160, 2011.

[19] F. Chantree, B. Nuseibeh, A. De Roeck, A. Willis, Identifying nocuous ambiguities in natural language requirements, *14th IEEE International Requirements Engineering Conference (RE'06)*, 2006, pp. 59–68.

[20] V. Ambriola, V. Gervasi, On the systematic analysis of natural language requirements with circe, *Autom. Softw. Eng.*, vol. 13, no. 1, pp. 107–167, 2006.

[21] A. Ferrari, G. Lipari, S. Gnesi, G. O. Spagnolo, Pragmatic ambiguity detection in natural language requirements, *2014 IEEE 1st International Workshop on Artificial Intelligence for Requirements Engineering (AIRE)*, 2014, pp. 1–8.

[22] A. Ferrari, P. Spoletini, S. Gnesi, Ambiguity and tacit knowledge in requirements elicitation interviews, *Requir. Eng.*, vol. 21, no. 3, pp. 333–355, 2016.

[23] Y. Zhang, B. M. Wildemuth, Unstructured Interviews. pp. 1-9, 2009.

[24] D. Zowghi, C. Coulin, *Requirements Elicitation: A Survey of Techniques, Approaches, and Tools, Engineering and Managing Software Requirements*, Aurum, A & Wohlin, C (Eds.), Springer, USA. 2005.

[25] A. Arshad, M. F. B. Noordin, R. B. Othman, An Exploratory Qualitative Study to Identify Factors Affecting Requirements Elicitation and Fact Finding While Developing Health Care Knowledge Management Systems in Pakistan, *Sci. Int. Lahore*, vol. 29, no. 3, pp. 633–639, 2017.

[26] A. M. Hickey and A. M. Davis, Elicitation Technique Selection: How Do Experts Do It?, *Proceedings of the Eleventh IEEE International Requirements Engineering Conference*. pp. 169-178, 2003.

[27] H. Beyer, K. Holtzblatt, Apprenticing with the Customer, *Commun. ACM*, vol 38, no. 5, pp. 45–52, 1995.

[28] N. Maiden, G. Rugg, ACRE: selecting methods for requirements acquisition, *Softw. Eng. J*, vol. 11, no. 3, pp. 183–192, 1996.

[29] D. Carrizo, O. Dieste, N. Juristo, N, Systematizing requirements elicitation technique selection, *Information and Software Technology*, vol. 56, no. 6, pp. 644-669, 2014.

[30] H. F. Hofmann, F. Lehner, Requirements engineering as a success factor in software projects, *IEEE Softw.*, vol. 18, no. 4, pp. 58–66, 2001.

[31] P. Chatzoglou, L. Macaulay, Requirements capture and IS methodologies, *Inf. Syst. J.*, vol 6, no. 3, pp. 209–225, 1996.

[32] D. Carrizo, O. Dieste, N. Juristo, N, Contextual attributes impacting the effectiveness of requirements elicitation Techniques: Mapping theoretical and empirical research, *Information and Software Technology*, vol. 92, pp. 194-221, 2017

[33] D. De Vaus, *Surveys in social research*. Routledge, 2013

[34] M. L. Saunders, P. Lewis, A. Thornhill, *Research Methods for Business Students*. Financial Times Prentice Hall Inc., London, 2009

[35] F. J. Fowler Jr, *Survey research methods*. Sage publications, 2013.

[36] N. Martin, S. Gregor, Requirements Engineering: A case of developing and managing quality software systems in the public sector, in *Engineering and Managing Software Requirements* (pp. 353-372). Springer, Berlin, Heidelberg, 2005.

[37] S. Arif, Q. Khan and S. A. K. Gahyyur, Requirements Engineering Processes, Tools/Technologies, & Methodologies, *International Journal of Reviews in Computing*, pp. 42-56, 2009.

[38] M. Yousuf, N. Asger, Comparison of various requirements elicitation techniques, *International Journal of Computer Applications*, vol. 116, no. 4, 2015.

[39] S. N. Kumari, S. A. Pillai, A survey on global requirements elicitation issues and proposed research framework, *2013 IEEE 4th International Conference on Software Engineering and Service Science*. pp. 554-557, 2013.

[40] F. Anwar, R. Razali, Requirements Elicitation Techniques Selection Survey, In *SoMeT*. (pp. 280-294), 2014, August.

[41] N. R. Darwish, A. A. Mohamed, A. S. Abdelghany, A hybrid machine learning model for selecting suitable requirements elicitation techniques, *International Journal of Computer Science and Information Security*. *14*(6), pp.1-12, 2016.

[42] L. C. Ronoh, G. M. Muchiri, F. Wabwoba, *Factors affecting requirements elicitation for heterogeneous users of information systems*. 2015.

[43] J. Okesola, Olatunji. et al. Qualitative Comparisons of Elicitation Techniques in Requirement Engineering. *Qualitative comparison of elicitation techniques in requirements engineering* 14 (2), 2019, pp. 565-570.

[44] A. Ullah, S. Nazir, S. Shahzad, Trade-off Analysis Among Elicitation Techniques Using Simple Additive Weighting Method, in *Advances in Intelligent Systems and Computing*, Springer International Publishing, 2018, pp. 498–514.

[45] J. Horkoff, J. Ersare, J. Kahler, T. D. Jorundsson, I. Hammouda, Efficiency and Effectiveness of Requirements Elicitation Techniques for Children, in *2018 IEEE 26th International Requirements Engineering Conference (RE)*, 2018.

[46] IA. Iqbal, I. A. Khan, S. Jan, A Review and Comparison of the Traditional Collaborative and Online Collaborative Techniques for Software Requirement Elicitation, in *2019 2nd International Conference on Advancements in Computational Sciences (ICACS)*, 2019.

Information about the author:

Abdullah A H Alzahrani He received the BSc degree from KAU University, Jeddah, Saudi Arabia in 2007 and received the MSc and Ph.D. degrees from University of Essex, Colchester, United Kingdom, in 2011 and 2016, respectively. He joined Umm Al Qura University, Makkah, Saudi Arabia in 2008. Since 2008, he has been with the Computing College at Alqunfuda, Umm Al Qura University, where he is currently an Assistant Professor. His main areas of research interest is software engineering. Dr Abdullah A H Alzahrani was a Vice Dean of Computing College at Alqunfuda from 2016 to 2019. Currently, Dr Alzahrani Vice Dean for Development and Entrepreneurship of Computing College at Al Lith, Umm Al Qura University from 2019 to present.

Manuscript received on 03 January 2020

APPENDIX – LIST OF QUESTIONS IN THE SURVEY

General Data

2. Name (optional)
3. Gender
4. Email (optional)
5. Academic qualification
6. The sector in which you work in
7. Job title
8. Years of Experience

Part (1 of 3)

9. Rate your knowledge of: Requirements engineering; Fact-finding techniques

10. How often do you use the following fact-finding techniques? Sampling of existing documents; Interviews; Observation; Questionnaires; Prototyping.

11. In the case that the system is to be used by the general public, to what extent does this affect the choice of the fact-finding technique?

12. If the system is used by the general public, rate the fact-finding techniques as most appropriate to less appropriate. Sampling of existing documents; Interviews; Observation; Questionnaires; Prototyping.

Part (2 of 3)

13. To what extent the number of stakeholders (staff - administrators - system owners - technical staff within the organisation) within the system influences the choice of the most appropriate fact-finding technique

14. If the number of stakeholders within the organisation influence the choice of a fact-finding technique, choose fact-finding techniques based on the most appropriate for each range of number of stakeholders. Sampling of existing documents; Interviews; Observation; Questionnaires; Prototyping.

15. To what extent do stakeholders' knowledge of information technology affect the choice of fact-finding technology?

16. If stakeholders' knowledge of IT not is very Basic knowledge, rate the fact-finding techniques as most appropriate to less appropriate: Sampling of existing documents; Interviews; Observation; Questionnaires; Prototyping.

Part (3 of 3)

17. To what extent does the geographical distance affect the choice of the fact-finding techniques?

18. In the case that the geographical distance affects the choice of fact-finding technique, rate the fact-finding techniques as most appropriate to less appropriate: Sampling of existing documents; Interviews; Observation; Questionnaires; Prototyping.

19. To what extent does the availability previous documentations of the system to be evolved affect the choice of the fact-finding techniques?

20. In the case that the previous documentations of the system to be evolved are unavailable, rate the fact-finding techniques as most appropriate to less appropriate: Sampling of existing documents; Interviews; Observation; Questionnaires; Prototyping.

21. When collecting data in the requirement analysis, how often:

22. To what extent does the occurrence of conflicting information affect the choice of the fact-finding techniques?

23. In the case the occurrence of conflicting information affects the choice of fact-finding technique, rate the fact-finding techniques as most appropriate to less appropriate: Sampling of existing documents; Interviews; Observation; Questionnaires; Prototyping.

24. In this study, we highlighted the most important factors influencing the choice of fact-finding techniques. In your opinion, mention other factors influencing your choice of the most appropriate fact-finding technology