Exploring the Effects of Prototyping on Requirements Elicitation Interviews

by

Samantha Leigh Warmerdam

6595189

Master's Thesis

Submitted in Fulfilment of the Requirements for a Master's Degree in Business Informatics at Utrecht University

2024

First supervisor: Fabiano Dalpiaz

Second supervisor: Sjaak Brinkkemper

Abstract

This thesis examines the effects of prototyping on requirements elicitation interviews. The primary goal hereof is to determine the extent to which the need to develop a prototype affects the qualities of requirements elicitation interviews when compared with a given framework. The results were intended to be used to improve current requirements engineering practices and provide insights into how interviews and prototyping interact so that the interview process may be refined.

Using content analysis alongside deductive coding, basic interviews were compared with interviews conducted alongside the creation of a prototype. Several findings emerged as a result. Firstly, the type of interview – namely initial or follow-up interviews – has an impact on interview qualities. Secondly, the use of a prototype caused the group creating prototypes to use less hypotheticals and examples during their interviews. This is speculated to relate to an existing concern about prototype usage causing bias towards existing solutions.

This research highlights the need for further research into more robust guidelines and frameworks for effective interviewing, as well as general research regarding the interactions between elicitation techniques in general. Future work should aim to validate the results of this thesis and explore other elicitation techniques in this manner, with the goal of reaching a level of understanding of requirements elicitation techniques that positively impacts both experts and novices alike.

Acknowledgements

I would like start off by thanking my supervisor, Dr. Fabiano Dalpiaz. You showed me boundless patience and understanding throughout this whole process. Your support, advice, and many tireless hours spent guiding and assisting me for this thesis will forever be with me, and I thank you for everything.

Thank you to Alessio Ferrari. Your contributions to the first phases of this thesis were invaluable, and I am grateful for all of the time you set aside and guidance you gave.

I am grateful to Merve Yldis, for your contributions to the initial phases of refining the guidelines.

Thank you to all of the participants for your time and commitments, and for graciously allowing me the use of your data in this thesis.

And of course, a huge thank you to all the other staff in the department who made this possible. You do great work, and I am thankful for everything you have done to support myself and other students.

To my study advisor Sara O'Keeffe, for checking in on me and guiding me through the steps to get here. You made sure I got on track and helped me during a very difficult time.

An enormous thank you to my psychologist, Ayelle Martis. Without you, I would not be here today. You helped me in ways I was afraid were no longer possible, and reminded me that I am worth so much more than I thought. Thank you for everything.

To Heleen Hoolwerf-Overduin, for taking care of my physical health and making sure I got the care I needed when I needed it.

To my family, I am grateful for your support, words of encouragement, and efforts to keep me on track to complete this thesis. You all made sure I was working, and helped ease peripheral stresses when times were tough, and for that I am forever grateful.

To Jacques Fooy, thank you for keeping me stable when things were dire, and for being my number one cheerleader every waking moment. I will never forget your boundless support during this thesis.

To Noël Bainathsah, for giving me invaluable emotional support throughout the entire process, and for being an incredible friend and housemate throughout everything. We did it!

To Shannon du Plessis, thank you for always keeping my head on right, being an irreplaceable friend even when you have five hundred other commitments, and for keeping me calm and reasonable when things seemed completely out of control. I could not ask for better, and I am so incredibly thankful I have you in my life.

To my other degenerates: Jordi Ferreira, Riaan Fourie and Daniel Holland. Without you, I doubt I would be here. You all kept me (in)sane and happy in times when I was fighting for a semblance of normalcy.

And finally, thank you to anyone and everyone who helped in some way, treated me with kindness and patience, or even made me an extra-good coffee on days when it was difficult to get out of bed. You have all been incredible, and I am so grateful that I have been able to reach the end of the road with your support.

Table of Contents

List of Abbreviations

Abbreviation	Definition
RE	Requirements Engineering
RE interviews	Requirements elicitation interviews
MVP	Minimum Viable Product
AI	*Explore As-Is System* tag in NVivo
TB	*Explore To-Be System* tag in NVivo
TB (neg)	*Explore To-Be System*, negative usage tag in NVivo
DR	*Develop Rapport with Stakeholders* tag in NVivo
AM	*Avoid Misinterpretations* tag in NVivo
VC	*Verify Conclusions with Stakeholders* tag in NVivo
VC (summary)	*Verify Conclusions with Stakeholders*, summary tag in NVivo
PI	*Present Non-Superficial Introduction* tag in NVivo
FT	*Allocate Stakeholder Feedback Time* tag in NVivo
HE	*Use Hypotheticals and Examples* tag in NVivo
SA (pos)	*Introduce Stakeholder Agency*, positive usage tag in NVivo
SA (neg)	*Introduce Stakeholder Agency*, negative usage tag in NVivo

1 Introduction

It is no secret that information systems, or ISs, have become progressively relevant over the last century. The interpretation of what constitutes ISs has evolved significantly, from **decision-making** or **problem-solving** systems in the context of the individuals making use thereof, to a variety of definitions depending on who one asks and in which industry (Hirschheim & Klein, 2012; Wood-Harper et al., 1985).

As the concept meaning has shifted, it has opened the doors for numerous new streams in which ISs may be created, used, and studied. This, in turn, has led to the changing expectations of the industries and specialists making use thereof. Indeed, many industries have given information systems ever-increasing importance in their various processes, leading to a need for effective creation, usage, and updating of these systems (Hirschheim & Klein, 2012; Sulianta et al., 2019).

Cue: Requirements Engineering. **Requirements Engineering**, or **RE**, is defined by Laplante & Kassab (2022) as follows:

> … the branch of software engineering concerned with the real-world **goals** for, **functions** of, and **constraints** on [software] systems. It is also concerned with the relationship of these factors to precise **specifications** of system [and software] behavior and to their evolution over time and across families of related systems (Laplante & Kassab, 2022; Zave, 1997).

From this definition, we can glean that systems in the context of RE have goals, functions, and constraints, and that these elements form part of system specifications. Put differently, there are **requirements** of a system in a particular context which should be fulfilled by identifying the precise nature of those requirements regarding how they influence the system. Without these requirements, developers may lack the appropriate scaffolding upon which to build an information system (Bormane et al., 2016; Khan et al., 2014). In this regard, identifying, or **eliciting**, these requirements for the software becomes a significant factor in constructing or modifying a system. In fact, today, elicitation is considered one of the most critical parts of IS creation and development as it directly affects project success (Kumari & Pillai, 2014).

Requirements elicitation, therefore, is one of the most important first steps in the development of information systems. It involves extensive communication between stakeholders – or parties

who hold a stake in the system – and analysts – or requirements engineers. During these communications, software requirements are identified, refined, and prepared for the next phase in the IS development timeline. Therefore, elicitation should be conducted in the most effective manner to structure software requirements which meet the relevant standards imposed on such requirements, such as being complete and unambiguous (Bano et al., 2018). However, even RE professionals may experience issues when trying to elicit requirements effectively from stakeholders, and this can have an impact on the developed software (Martin et al., 2019; Spijkman et al., 2022).

Chief among elicitation techniques is the conducting of interviews with stakeholders (Tiwari & Rathore, 2017; Wagner et al., 2019). These interviews serve to facilitate a dialogue between stakeholders and analysts in order to identify the most significant requirements for a given system. The quality of these interviews can vary depending on a variety of factors (such as interviewer experience, interview method, additional methods used, and stakeholder expertise), impacting the resulting requirements and the process of eliciting them (Davis et al., 2006; Zowghi & Coulin, 2005).

In addition, some evidence suggests that developing a **prototype** alongside the interview process can be effectively used to demonstrate available solutions to stakeholders and encourage their active involvement in developing requirements (Zowghi & Coulin, 2005). An investigation is required in order to identify how the creation of a prototype affects interviews throughout the elicitation process, and therefore what the quality is of the resulting requirements. This has the aim of refining existing requirements elicitation techniques. The specific gap to be explored in this regard is elaborated on in the sections that follow.

1.1 Problem Statement

An appropriate execution of the elicitation process is imperative to creating high-quality requirements. Elicitation is also widely known to be a difficult task for professionals and students alike (do Prado Leite & Gilvaz, 1996; Falessi et al., 2018; Mohedas et al., 2022). Many factors can contribute to the quality of elicitation and of the resulting requirements, including stakeholder experience and techniques used (Aranda et al., 2022; Burnay et al., 2020).

Among these techniques are interviewing and prototyping, as mentioned above. Often used

together, these techniques each have their own strengths and weaknesses. As previously stated, interviewing is one of the most common, traditional elicitation techniques used today (Tiwari & Rathore, 2017; Tuunanen, 2003). Prototyping is also widely used, but is speculated to be situation-dependent in its results. Specifically, it is posited that prototyping may have a negative impact on the requirements elicitation process in certain contexts, such as having low domain knowledge or emerging bias towards specific solutions. This can, in turn, negatively affect the resulting requirements (Carrizo & Quintanilla, 2018; Mannio & Nikula, 2001).

Overall, there is currently no scientific evidence supporting the inclusion of a prototype alongside interviews in the requirements elicitation process. Since RE is still growing and developing, there exists a need to address uncertainties like this in order to refine the RE process and determine the impact of certain factors.

Further investigation into the nature of this impact is needed in order to understand the problem domain more thoroughly and prevent inferior elicitation. To evaluate the gap in information, the research conducted explores the influence of the presence of a prototype on the qualities of requirements elicitation interviews. This research aims to determine whether the influence of a prototype on requirements elicitation interviews tends towards specific characteristics in a specified context.

This should contribute to more effective requirements elicitation when using the interview process and deciding which techniques to combine it with, as well as to the creation of more robust, high-quality design specifications (Dar et al., 2018). Furthermore, the research should contribute to a better understanding of requirements elicitation interviews, specifically with regard to prototypes. It should also have the consequence of contributing further insights on how to analyse these interviews in general.

1.2 Research Questions

In order to address the problem defined above, there are some factors which must be considered. Firstly, it would be subjective to declare that an interview possesses certain characteristics without first formulating a means of comparison. There is therefore a need to identify some kind of framework to act as the point of comparison for this research and future work.

Secondly, the concept of a *quality*, in this context, implies attributes or characteristics. Identifying the *qualities* of something enables comparison of that item's characteristics to that of others. In the context of the research, identification of the qualities of requirements elicitation interviews should aid in understanding precisely how these interviews change in the presence of a prototype.

Lastly, requirements elicitation interviews are often conducted more than once, leading to an initial interview and follow-up interview(s) (Hickey & Davis, 2003; Schneider, 2007). Since the creation of a prototype in this context is implied to occur after at least one interview, it is important to use both initial and follow-up interviews in order to derive an appropriate comparison.

In order to address these factors, the following main research question is proposed:

> **MRQ:** To what extent does the need to develop a prototype affect the qualities of requirements elicitation interviews when compared with a given framework?

This research question can be broken down into several sub-questions concerning each of its factors, where $H_{0\text{-}X}$ represents the null hypothesis where relevant:

> **RQ_1:** What frameworks exist for analysing the qualities of requirements elicitation interviews?
>
> **RQ_2:** To what extent does the type of interview affect the qualities of requirements elicitation interviews, regardless of prototype creation?
>
> > **$H_{0\text{-}2}$:** Interview type has no effect on the qualities of requirements elicitation interviews.
>
> **RQ_{2A}:** To what extent does the need to develop a prototype affect the qualities of *initial* requirements elicitation interviews?
>
> > **$H_{0\text{-}2A}$:** The need to develop a prototype has no effect on the qualities of initial requirements elicitation interviews.
>
> **RQ_{2B}:** To what extent does the need to develop a prototype affect the qualities of *follow-up* requirements elicitation interviews?
>
> > **$H_{0\text{-}2B}$:** The need to develop a prototype has no effect on the qualities of follow-up requirements elicitation interviews.

To answer these questions, this thesis has been structured as follows: RQ_1 shall be answered through the conducted literature review described in *§2 Literature Review*; The remaining research

questions shall be addressed according to the research method described in *§3 Research Method;* The refinement of the selected framework from RQ_1 is described in *§4 Refining the Framework*; The discovered results of the research are offered in *§5 Results*; These results are analysed and discussed in the context of the research goals and questions in *§6 Discussion*; Finally, closing thoughts are offered in *§7 Conclusion*.

2 Literature Review

Research into the exact nature of the relationship between elicitation interviews and prototypes is scarce. However, many resources cover these subjects individually or side-by-side. In order to explore the literature as thoroughly as possible, a systematic literature review was conducted. This involved isolating the elements of all research questions and determining what the most important factors were for each of these elements.

By identifying these factors, a scope was formed, allowing for the creation of a suitable method for finding, including, or excluding literature. In the sections that follow, the method and scope are defined. Thereafter the conceptual foundations of the subject matter are explored, as well as the existing frameworks which were considered for the research. **The section on existing framework also serves to answer RQ_1.**

2.1 Method

Since the section that follows will be used in part to answer RQ_1, it is important to determine precisely what method has been followed to answer this question. As previously mentioned, a systematic literature review was used in order to ensure the subject matter was sufficiently explored. This entailed the following steps, according to Khan et al. (2003):

1. Framing the question(s)
2. Identifying relevant publications
3. Assessing study quality
4. Summarizing the evidence
5. Interpreting the findings

Framing the question(s) involves identifying the research questions (*§1.3 Research Questions*) as well as their components. These components approximate to the variables for research. For the identified research questions in this research, the variables were determined to be roughly as follows:

- interview quality
- prototype usage

- interview type
- framework principle

Identifying relevant publications was then performed using these variables. In order to ensure these were covered thoroughly, each variable – or a variant / level thereof – was entered as a search term or keyword, either in isolation or in combination with one of the other keywords. These were entered into popular academic search engines, such as Google Scholar, ScienceDirect, and ResearchGate. Some examples of the keywords used include the following:

- *requirements elicitation*
    - *methods*
    - *interviews*
    - *prototyping*
    - *interviews quality*
    - *framework*
    - *assessment*
- *prototyping in requirements engineering*
- *assessment of requirements elicitation interview quality*
- *how to conduct requirements elicitation interviews*
- *factors influencing requirements elicitation interviews*
- and so on.

Literature which directly concerned any of these topics was added to the Mendeley Reference Manager, where they were organised by category to facilitate evaluation.

Assessing study quality was then performed by ensuring the gathered literature fell within the identified scope (see *§2.2 Scope*) to a satisfactory degree. Studies were also evaluated based on the number of citations they have, as well as which journals they were published in and the type of publication.

Summarising the evidence resulted in the completed literature review presented in *§2.3 Conceptual Foundations* and *§2.4 Existing Frameworks*. In these sections, the overall findings of the literature are explored and summarised.

Finally, **interpreting the findings** was conducted mainly in the context of RQ_1, since this research question relies solely on existing literature. For the rest of the literature, the findings were used to identify the existing understandings, current expectations, and gaps in research surrounding the topic of prototyping in the requirements elicitation interview process.

2.2 Scope

For the most part, initial queries to the various academic search engines were filtered within the last five years, in order to ensure relevance of the subject matter. However, papers falling outside of this range were considered where relevant. In fact, exclusion criteria were quite relaxed, as the literature into the MRQ is scarce – approximately 100 results were found when combining "requirements elicitation interviews" and "prototypes" / "prototyping" specifically, with around 60 of these having been published within the last five years.

To avoid oversaturation of information, texts which contained approximately the same information and citations were explored for their connecting citations where relevant. The cited works were then given priority if they explored the subject more deeply or were considered "classics" – seminal papers which are cited hundreds of times. The general process for deciding whether or not to use a piece of literature is outlined in **Figure 1**. This figure also represents the combination of steps 2 and 3 of the systematic literature review.

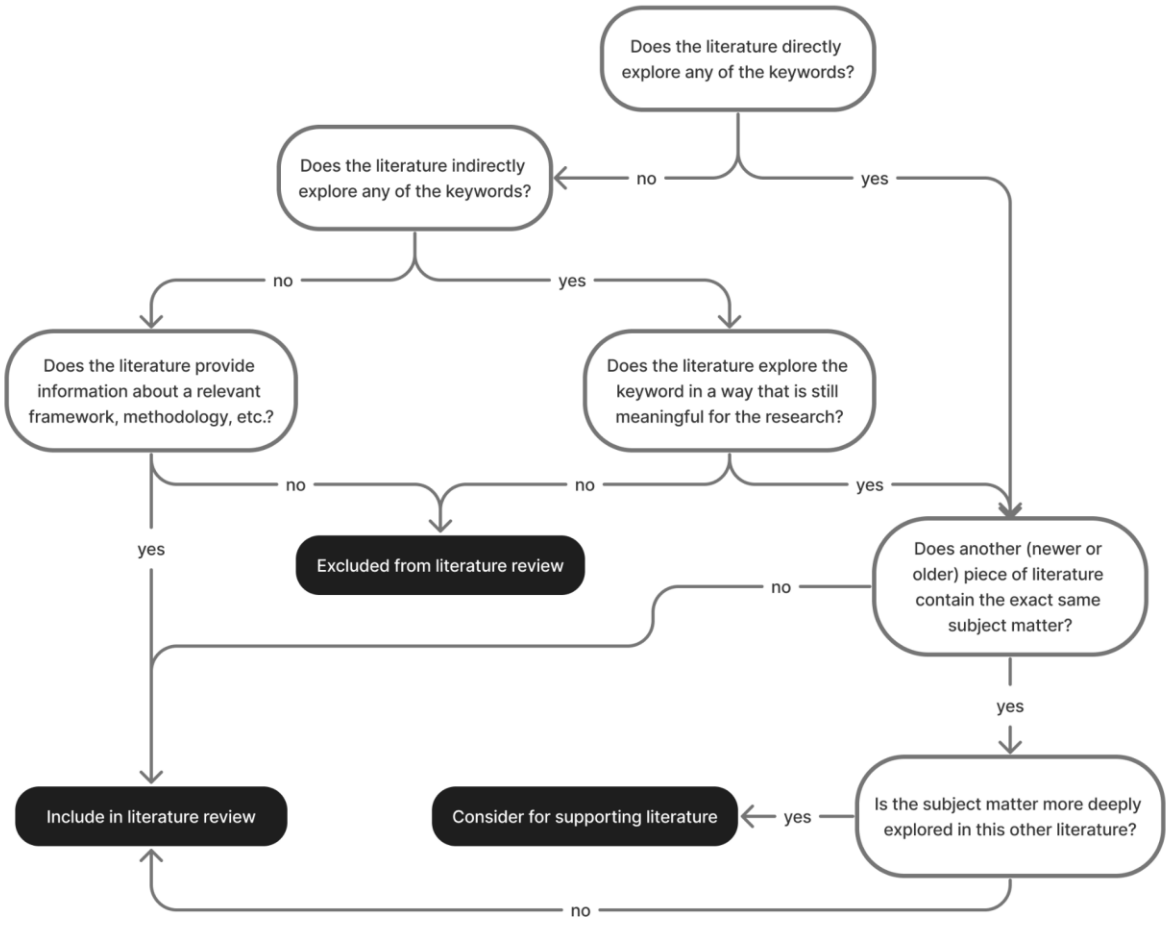


Figure 1. The decision map for deciding whether to include a piece of literature.

2.3 Conceptual Foundations

2.3.1 Requirements Elicitation Interviews

Requirements elicitation interviews (or RE interviews) are interviews aimed at identifying new requirements, or otherwise refining existing ones, and are the most common technique used for elicitation (Wagner et al., 2019). Usually, this takes the format of stakeholders (*interviewees*) being asked questions by the requirements engineer(s) (*interviewers*) in order to identify desired and undesired requirements (Ferrari et al., 2019; Häußer et al., 2022; Rafiq et al., 2017). Interviews are usually tailored to the industry in which they are conducted, such as journalism or research (Bano et al., 2019; Martin, 2017). RE interviews may differ from these in a few ways.

Firstly, they require thorough preparation and preset goals in order to be used effectively. Identifying stakeholders who will be able to answer questions the most suitably in order to achieve one's goals is valuable, as it reduces the chances of acquiring low-quality or low-value requirements (Zhao & Zhao, 2018). Creating some kind of guiding document can also make the interview process smoother, while also making it easier to recover if the interview gets derailed. This can take many forms, including bullet-pointed lists, mind-maps, standard question lists, and so on (Baxter et al., 2015; Portigal, 2013).

Secondly, the characteristics of the interviewer and their questions can make or break the resulting requirements. The phrasing of questions is imperative for getting the most suitable answers to convert into formalized requirements (Zaremba & Liaskos, 2021). In this regard, interviewers should have some knowledge of the problem domain in order to communicate effectively. In turn, they should be able to guide the interview in such a way that their interviewees are able to give relevant, useful, and complete answers without becoming overwhelmed by jargon, control, or other factors (Hadar et al., 2014; Häußer et al., 2022).

Finally, these interviews may include other elicitation techniques, such as storyboarding, simulations, workshops, brainstorming, and prototyping. They are also usually conducted more than once over the course of the requirements formalization process (Ferrari et al., 2022; Lending et al., 2022; Rueda et al., 2020). Generally, the first interview is considered an initial interview, while all interviews thereafter are considered follow-up interviews.

Initial interviews are the first interviews conducted in a formal setting with the stakeholders and the requirements engineer(s). Usually, these are exploratory interviews in a very limited capacity, as it is expected that the interviewer already has some knowledge of the problem domain. The main purpose of these interviews is to gain a more robust understanding of the problem domain, while also gathering data as effectively as possible in order to facilitate adequate requirement construction (Ferrari et al., 2022; Hadar, et al., 2014). While these initial interviews should be fairly thorough and informative, they are usually complemented by follow-up interviews.

Follow-up interviews are interviews performed after the initial interview. These are usually used to address ambiguities, clarify unclear requirements, correct mistakes, address inconsistencies, validate requirements, and uncover remaining requirements. These interviews may tend to take

time and energy, and so it is important that they are concise, cover the relevant topics in appropriate ways, and respect the constraints of both stakeholders and requirements engineers (Schneider, 2007). Depending on the nature of previous interviews, these follow-up interviews can take many forms, and include many other elicitation methods, such as prototyping (Häußer et al., 2022; Spoletini et al., 2018).

2.3.2 Prototyping in RE

Prototyping is the act of creating mock-ups or blueprints of the front end of an application or system (Pacheco et al., 2018; Rueda et al., 2020). Primarily, prototypes are used to validate and elicit requirements. Often used in conjunction with other elicitation techniques, prototyping can take many forms – such as *paper-based prototyping*, where system mock-ups are created on paper, *"Wizard of Oz" prototyping*, where an individual simulates system responses to certain user inputs, and *automated prototyping*, where a rapid development environment is used to create an executable prototype (Paetsch et al., 2003).

Prototyping can be split into various classifications, as seen in **Table 1** (Mannio & Nikula, 2001). In general, prototypes can be classified into two major categories, namely: *low-fidelity* and *high-fidelity*.

Low-fidelity prototypes are prototypes which are constructed quickly, with little to no actual functionality. Their main purpose is to demonstrate the interface of a system through storyboards, wireframes, and so on. They are often used fairly early in the development lifecycle, as they require low investment of resources in order to explore conceptual approaches to the system. These kinds of prototypes include throw-away, non-executable, visual, horizontal and requirements prototypes (Mannio & Nikula, 2001; Rudd et al., 1996).

Table 1. Prototype classifications, according to Mannio and Nikula (2001).

System development model	
Evolutionary - arise early in development and extend to final system - incremental	Throw-away - discarded after final system development - specific case

Operation	
Executable - constructed using high-level programming languages - high-risk and complex	Non-executable - paper prototypes and mock-ups - low-risk and simple
Representation	
Textual - programming - formal languages	Visual - modelling techniques (e.g. entity-relationship-diagrams, storyboards) - mock-ups
Level of detail	
Vertical - high-quality implementation - non-functional requirements	Horizontal - functional requirements - no real functionality - refine unclear requirements
Objective	
Architectural - performance - technical feasibility	Requirements - requirements acquisition - user interface design

High-fidelity prototypes are interactive prototypes which represent core functionality of the product's user interface. They take time to develop, with the trade-off being accurate representation of the final product. High-fidelity prototypes include evolutionary, executable, textual and architectural prototypes, and can include both horizontal and vertical prototypes depending on necessity and scope (Palanque et al., 2009; Rudd et al., 1996).

The preferred technique for students and non-professionals appears to be low-fidelity paper-based (visual non-executable) prototyping, mainly due to its simplicity, ease-of-execution, and low time investment. Students may also not have access to the appropriate software for creating more sophisticated prototypes (Miller, 2021). On the other hand, the literature seems to suggest that professionals demonstrate no preference, instead adjusting the type of prototyping used based on the specific needs of the project, its complexity, stakeholders, timeline, and available resources (Gordon & Bieman, 1995; Mannio & Nikula, 2001; Palanque et al., 2009; Wagner et al., 2019).

According to many sources, prototyping is presumed to contribute positively to the elicitation process, especially when developing new systems (Lending et al., 2022; Zowghi & Coulin, 2005). It can give teams direction and focus, thereby enhancing communication between stakeholders and analysts. The active involvement of stakeholders in the development process can also improve trust and morale of all involved parties. It can also facilitate earlier feedback on refining requirements (Hickey & Dean, 1998; Qurban & Austria, 2009; Yousuf & Asger, 2015). For students, certain types of prototyping – such as paper prototyping – may assist with creative problem-solving without worrying about technological barriers or complex visualizations (Miller, 2021; Vijayan & Raju, 2011).

However, there is also evidence to suggest that prototyping may negatively impact elicitation, leading to poorer quality of elicited requirements and the end product. This can be due to many factors. Firstly, prototyping may consume unnecessary time and costs, especially when systems are more complex (Anwar & Razali, 2014; Mannio & Nikula, 2001). Secondly, once a system is "on paper", stakeholders and analysts alike may become resistant to changes therein, in a phenomenon known as requirements fixation. This can, in turn, have an impact on creativity and originality (Mohanani et al., 2014). Finally, there is a risk that prototyping may lead to incomplete system implementation, especially when conducting evolutionary prototyping (Yousuf & Asger,

2015; Zowghi & Coulin, 2005). A recent study has also suggested the possibility that, without domain familiarity, prototyping may not lead to the elicitation of sufficient requirements. This could be attributed to analysts not knowing what topics are relevant, and therefore which questions to ask (Carrizo & Quintanilla, 2018).

2.3.3 Question Types in (RE) Interviews

Questions in interviews can be formulated and categorized in a multitude of ways. Depending on the need for categorization, the nature of the interviews being conducted, and the granularity required, the types of questions being asked become significant for acquiring the appropriate data in an interview.

However, choosing a typology guide can be daunting, especially when it comes to RE interviews, since very little literature exists specifically for this purpose. Zaremba and Liaskos (2021), for example, provide a thorough way of sorting questions, including categories such as time, content, style, objective, and so on, each containing their own sub-categories (see Appendix A). This typology comes as the result of an extensive literature review into the various ways of categorizing and characterizing RE interview questions. Unfortunately, the paper contains very little explanation into the limits of each category and sub-category, making the lines potentially blurry during analysis. Additionally, this typology may be too densely packed for the current research, as it involves the attachment of several, potentially irrelevant, tags to single questions.

Other researchers have chosen to categorize questions in a different way. Hartwell, Johnson and Posthuma (2019) categorize questions on a matrix, including factors like experience, knowledge, and situation-specificity (see Figure 2). This matrix serves as a conceptual model for the foundations and foci of question types in structured interviews. The model is largely based on works such as that of Klehe and Latham (2006) – who roughly divide questions into those pertaining to behavior description interviews and situational interviews – and Taylor and Small (2002), who categorize questions as either situational or past behavior. Unfortunately, this categorization is slightly too simplistic, as it is constructed for interviews in general, rather than specifically for requirements engineering.

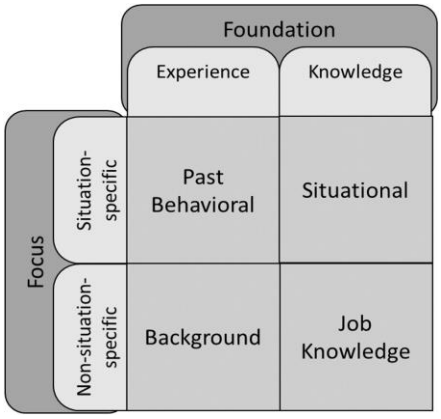


Figure 2. The foundations and foci of structured interview question types, according to Hartwell, Johnson and Posthuma (2019).

Kvale (1994) provides perhaps the most elegant typology, categorizing questions into 10 types, namely: introducing questions, follow-up questions, probing questions, specifying questions, direct questions, indirect questions, structuring questions, silence, interpreting questions, and throw-away questions. These question types are used or considered in existing studies (Sadeghi, 2019; Zaremba & Liaskos, 2021), and seem to demonstrate a manageable level of granularity when it comes to processing the data. While this typology is also catered towards interviews in general, the details of each question type are broad enough to apply them to RE interviews. There is also significant overlap between these question types and the typology presented by Zaremba and Liaskos (2021).

- ***Introducing questions***, or *opening questions*, are questions which begin the interview, or serve to facilitate exploration on a new subject.
- ***Follow-up questions*** are used to clarify or extend what has already been said.
- ***Probing questions***, similarly to *follow-up questions*, extend given answers. They serve to further extend the narrative, over-and-above what might be discovered through follow-up questions.
- ***Specifying questions*** serve to draw more precise information from broad or general statements.
- ***Direct questions***, like *specifying questions*, are aimed at gaining more precise information, but are more explicitly and unambiguously stated. These questions are usually saved for later in interviews, once interviewees have been able to make their own explorations and

descriptions.

- ***Indirect questions*** are open-ended, projective questions which allow the interviewees to give answers that they consider important or relevant. They usually lead to the asking of other question types.
- ***Structuring questions*** serve to close off a part of the interview and move on to another, or to move away from rambling or irrelevant answers.
- ***Silence*** is simply a moment to reflect on and give personal meaning to information. It can be good to have silence in interviews, whether due to the need for a break or to respect for cultural differences.
- ***Interpreting questions*** seek to validate an interviewer's understanding of a concept or answer by rephrasing it. They serve to clarify and more thoroughly interpret information.
- ***Throw-away questions*** are aimed at relaxing the interview or the subject matter, such as in the case of a breach of sensitive areas of discussion.

These categories appear to be clearly defined and relatively simple to identify in practice, while also providing enough complexity to allow for some level of analysis.

2.4 Existing Frameworks

In order to analyse the acquired data appropriately, it is necessary to have a standard for comparison, especially when investigating the quality of the data. Below is explored some of the available frameworks and rubrics for analysing the qualities of RE interviews. This section also serves to answer RQ_1:

> **RQ_1:** What frameworks exist for analysing the qualities of requirements elicitation interviews?

Burnay et al. (2020) claim that statements have different *grounds*. Put simply, they define *grounds* as the foundation, or the "the underlying rules … that were used by stakeholders to share the Statement." These rules are either speculation (*hypothetical statement*) or actual experience (*experiential statement*), and they hypothesize that these will cause variation in information quality. The two concepts are not entirely separate, as they claim experiential statements may lead to extrapolation from related experience, leading to hypothetical statements.

They briefly define a set of qualities they use to explore these statements. These qualities are:

exhaustive, or how much relevant information is provided in a statement made by a shareholder; *steady*, or how subject to change a statement is once made by a stakeholder; *objective*, or whether another person will interpret a statement in the same way as initially intended by a stakeholder; *creative*, or how novel and valuable a statement is relative to common knowledge; and *orderable*, or how a statement can be prioritized or ordered when considering other statements made by a stakeholder (Burnay et al., 2020).

This approach to analysis prioritizes stakeholder responses, and therefore assigns quality to those responses. Despite this approach being relatively simple with regard to interview analysis, it is valuable to consider, as these qualities may be applied to both stakeholders and analysts. However, this approach may be too granular for the nature of the research to be conducted, as it requires the analysis of individual statements according to five qualities each.

Ezell et al. (2019) developed a rubric for assessment of requirements elicitation skills in students. The items in this rubric included *opening, analyze current (as-is) state, design the to-be system, visualization, closing, relationship building, active listening* and *teamwork.* This rubric was later revised by the authors to include *greeting* and elevate *relationship building* to be made up of *active listening* and *teamwork* (Lending et al., 2022). This is summarized in **Table 2**, where (+) indicates a new addition and (-) indicates a removal.

Many of these items are relevant in the context of the research to be conducted. However, while the rubric is impressively robust, some aspects cannot be evaluated using the available data in the research. For example, body language, affect, and positions of individuals cannot be evaluated using an audio file or a transcript (see *§3.1 Data Collection & Tools*). Regardless, this rubric is under consideration for the research.

Table 2. Criteria for Effective Requirements Elicitation Interviews according to Ezell et al. (2019) and Lending et al. (2022).

Criteria Item	Description
Greeting (+)	Greeting appropriately and generating rapport through small talk, team introduction, icebreakers.
Opening	Providing an organizational frame for the client, as well as a schedule, interview goals, and purpose.
Analyze current (as-is) state	Understanding the as-is system, and asking about good and bad aspects of the current situation, system, process, artifacts, etc.
Design the to-be system	Involving the client in the design of the to-be system.
Visualization	Using appropriate visuals (wireframes, storyboards, etc.) in order to aid parts of the meeting, using visuals to determine and evaluate scope, and integrating visuals into the interview discussions.
Closing	Debriefing, recapping, future plans, last questions.
Relationship building (-)	Greeting appropriately (shaking hands, standing up, etc.), attentive affect (eye contact, affirming remarks).
Active listening	Paying attention, giving feedback, attempting to interpret and paraphrase ideas, referencing past answers, requesting appropriate clarification.
Teamwork	Making roles and responsibilities appear normal and natural, having a clear leader, providing different viewpoints from different team members, effective teamwork.

Mohedas et al. (2022) provide, in the opinion of the researcher, the most appropriate method of evaluation, by constructing an in-depth set of recommended practices identified by evaluating a large swathe of appropriate literature. This set contains 12 practices and their descriptions, set out in Table 3. The practices are: *encourage deep thinking, develop a rapport with the interviewee, avoid misinterpretations, be flexible and opportunistic, verify the conclusions drawn from interviews, designer begins and interviewee concludes, use projective questioning techniques, use a co-creative interview strategy, introduce domain knowledge, have the interviewee teach you, explore contradictions,* and *break down expert tasks*.

This is certainly a much larger set of items for analysis, but it is also thorough and concise. Most, if not all, viable strategies employable by novices are included, making for much easier application to the research. Where they are not, it can be assumed these may be errors by the novices. Should this framework be used, adherence to the recommended practices would be evaluated against deviation therefrom in the analysis (see *§3 Research Methodology*).

Bano et al. (2019) explores an approach for analysing interviews from the opposite end of the spectrum. Whereas the previous papers have explored conducting interviews correctly, Bano et al. explore the incidence of mistakes during interviews. These are categorized as *question formulation, question omission, order of interview questions, communication skills, analyst behavior, customer interaction* and *teamwork and planning*. These categories are further divided into 34 sub-categories that provide insight into the exact nature of potential mistakes (see Appendix B).

The incidence of mistakes in the interview process can lead to issues further on in the process of requirements analysis. In this study, mistakes were first discovered through observation of student analysts, and then used as a means for further study of how these mistakes can be improved upon. The end goal was to determine whether students could improve their skills over the course of three interviews, with mistakes being made known to them.

As these mistakes were gathered from a study of novice analysts, this paper is under consideration for its relevance, but has been excluded for now due to the previous framework appealing more to the research needs. However, it is possible that this direction could be explored if time constraints do not become a concern.

Table 3. Recommended practices in requirements elicitation (Mohedas et al., 2022).

Recommended Practice	Description
Encourage deep thinking	Interviewers encourage stakeholders to think analytically and with logical reasoning.
Develop a rapport with the interviewee	Interviewers facilitate a comfortable discussion, and open and honest responses.
Avoid misinterpretations	Interviewers keep stakeholders' exact words or verify their interpretations.
Be flexible and opportunistic	Interviewers allow deviation from predefined topics and questions, especially in favour of exploring different, relevant topics that arise from stakeholder responses.
Verify the conclusions drawn from interviews	The results of interview analysis are verified with stakeholders to ensure they align with their perceptions.
Designer begins and interviewee concludes	Interviewers define goals and purposes of stakeholder interviews at the interview start, and stakeholders are given time at the end of the interview to discuss topics they were not asked about.
Use projective questioning techniques	Interviewer uses stories, metaphors, analogies, etc. to enhance elicited information.
Use a co-creative strategy	Interviewers attempt to shift ownership of the interview goals to stakeholders, so that they feel more of a stake in the outcome of the interview.
Introduce domain knowledge	Interviewers introduce domain knowledge, which can aid eliciting information about topics for which stakeholders may hold expertise.
Have the interviewee teach you	Interviewers feign ignorance in order to encourage stakeholders to elaborate on specific topics or break subjects down.
Explore contradictions	Interviewers explore contradictions between stakeholder statements, both of an individual and of different stakeholders.
Break down expert tasks	Interviewers follow up and probe experts in order to obtain all information, including information regarding physical and cognitive processes towards a task or goal.

3 Research Method

In preparation for this study, several options were considered on how to conduct it effectively. Grounded theory was originally considered an appropriate method for conducting this research. This methodology involves inductive reasoning by generating a hypothesis during the exploration of raw data (Oktay, 2012; Walker & Myrick, 2006). Initially, this was considered due to the exploratory nature of the study as well as comparability with an existing study on a similar topic. This similar study explored the contents of interviews conducted using the fit-gap analysis technique, which – given an existing software product – identifies supported needs as *fits* and unsupported needs as *gaps* (Spijkman et al., 2021).

However, it emerged that this was not the most suitable for the research as more methodologies were considered and the area of research became more apparent. This was, in part, due to the use of a pre-existing framework for coding and analysis, which is not congruent with the principles of grounded theory. Additionally, while the research is aimed at indicating a correlation between prototype usage and interview qualities, the ultimate focus of this research is on determining whether prototype creation may lead to positive or negative qualities, and not necessarily on any defined or discovered correlation as of writing.

Other study types – specifically ethnographic, phenomenological, and action research – were deemed irrelevant, once again due to the nature of the study to be conducted. The study does not concern any ethnographic information, nor does it pertain to the personal experiences of participants. It also does not involve exposing participants to a particular action or artifact. As such, these methodologies would not fully support the research questions.

Ultimately, it was determined that the best way to explore the research questions effectively was to allocate a method to each one insofar as was possible. In this regard, two major research methods were used in order to explore the subject matter. These are outlined in more detail below.

3.1 Research Design

3.1.1 Research Question 1: Systematic Literature Review

In order to begin the research with as clear of an image of the subject landscape as possible, a

systematic literature review was conducted as per the method outlined in *§2.1 Method*. Briefly, this was conducted (1) to gather as much information about the research domain and related studies as possible, and (2) to answer the first research question, namely:

> **RQ_1:** What frameworks exist for analysing the qualities of requirements elicitation interviews?

To achieve both of these purposes, the systematic literature review method outlined in Khan et al. (2003) was used. The findings for this research question have therefore been assumed to be answered in *§2 Literature Review*, where several frameworks were identified in order to analyse the quality of requirements elicitation interviews.

3.1.2 Research Question 2 and Main Research Question: Content Analysis

In order to answer the remaining research questions, including the main research question, it was necessary to determine the precise type of research that would be conducted and what steps would be the most appropriate in the research context. For reference, the remaining research questions are as follows:

> **MRQ:** To what extent does the need to develop a prototype affect the qualities of requirements elicitation interviews when compared with a given framework?
>
> **RQ_2:** To what extent does the type of interview affect the qualities of requirements elicitation interviews, regardless of prototype creation?
>
> > **RQ_{2A}:** To what extent does the need to develop a prototype affect the qualities of *initial* requirements elicitation interviews?
> >
> > **RQ_{2B}:** To what extent does the need to develop a prototype affect the qualities of *follow-up* requirements elicitation interviews?

A variety of methods were considered for the main body of the research. In addition to those previously mentioned, for this particular subsection of research questions, case study research was considered. **Case study research** involves the examination of individuals or groups of people as they conduct tasks or experience certain factors (Starman, 2013). Since the research includes the exploration of student groups as they perform requirements elicitation interviews, this methodology was initially considered appropriate.

However, there is some debate regarding whether or not certain factors disqualify research from

being considered a "case study." Wohlin (2021) defines a case study as follows:

> … an **empirical** investigation of a case, using **multiple** data collection methods, to study a contemporary phenomenon **in its real-life context**, and with the investigator(s) **not taking an active role** in the case investigated.

When one considers these requirements, the current research is disqualified on the grounds that it is within a simulated context instead of a real-life one. It is worth noting that the author himself mentions some differences in the requirements of a case study, with a point of relevance being that the real-world context is not always mentioned as necessary. Despite this, it was the decision of the researcher that this method would not be used in order to be consistent with the literature.

In this regard, since a large number of recorded interviews in a simulated setting were collected, content analysis was finally selected as the suitable method for the research. Briefly, **content analysis** has been defined as:

> "a research method that provides a **systematic** and **objective** means to make valid **inferences** from **verbal, visual, or written data** in order to **describe** and **quantify** specific **phenomena**" (Downe-Wamboldt, 1992).

In the case of the present research, we can identify a few things: Firstly, it is intended to make inferences about the data regarding its quality; Secondly, we are using verbal data (albeit transcribed); and Thirdly, we are attempting to describe and quantify recommended practices. Even when considering other definitions of content analysis, it is found that this method suits the research well.

According to White & Marsh (2006), a researcher uses analytical constructs, also called "rules of inference", to organise the data in such a way that it contributes to the answering of the research question. These analytical constructs are based on either existing theories, expert knowledge and experience, or existing research. In the case of the present research, existing research is used in the form of a variety of frameworks.

Additionally, White & Marsh (2006) choose to use the following definition of content analysis:

> "a research technique for making **replicable** and **valid** inferences from **texts** (or other meaningful matter) to the **contexts** of their use" (Krippendorff, 2004).

They also define a procedure for content analysis. The basic steps thereof are as follows: (1)

Establish hypotheses; (2) Identify appropriate data; (3) Determine sampling method and sampling unit; (4) Draw sample; (5) Establish data collection unit and unit of analysis; (6) Establish coding scheme that allows for testing hypothesis; (7) Code data; (8) Check for reliability of coding and adjust coding process if necessary; (9) Analyse coded data, applying appropriate statistical test(s); and (10) Write up results (White & Marsh, 2006).

For the purposes of this research, the method used for synthesizing and coding data (steps 6 – 9) will be deductive coding. **Deductive coding** is coding performed using a predefined list of codes. This helps to focus research and avoid unnecessary complications – such as perpetually defining new codes as they are discovered – which aids in answering specific research questions (Skjott Linneberg & Korsgaard, 2019). In the case of this research, deductive coding will include the use of a refined version of Mohedas et al. (2022)'s framework, described in detail in *§4 Refining the Framework.*

In this regard, **the proposed steps for research are as follows:**

1. **Generate hypotheses.** This involved the definition and refinement of the research questions, as well as the hypothesis for each, resulting in the questions defined in *§1.2 Research Questions.*
2. **Determine appropriate data**. For this step, the research questions were used to guide decision-making on what data would be the most suitable and reasonable to use for the research. This data included problem description documents, audio recordings of interviews, and completed requirements specifications.
3. **Determine sampling method & unit.** For this research, university students enrolled in a Requirements Engineering course were chosen as the overall sample group.
4. **Draw sample**. The sample group was analysed and refined in order to best represent real-world requirements engineers. This step, as well as step 3 above, are detailed in *§3.2 Sample Group & Setting.* It is prudent to mention here that this step was performed retroactively, just before data were actually processed, which is also explained in the mentioned section.
5. **Establish data collection unit and unit of analysis.** This step involved creating a repository for data collection, as well as establishing the groups for analysis. The creation

of groups was actually performed prior to the execution of the major research process, since the results of a university course project were used and these groups had already been formed.

6. **Establish coding schema that allows for testing hypothesis.** The framework demonstrated in *§4 Refining the Guidelines* was used to generate a suitable coding schema in NVivo.
7. **Code data.** This was performed in NVivo, a program commonly used in research to codify data.
8. **Check for reliability of coding and adjust coding process if necessary.** This process was conducted over approximately four months, during which the used schema was adjusted alongside the guidelines for applying the schema. In short, steps 6 – 8 have been used iteratively, until the level of agreement was approximately 95% across taggers. This is discussed in more detail in *§4 Refining the Guidelines.*
9. **Analyse coded data, applying appropriate statistical test(s).** For the purposes of the research, a Wilcoxon Signed-Rank test was used to evaluate differences between initial and follow-up interviews regardless of prototype usage. A Mann-Whitney U test was performed when looking at differences between the control group and the group making use of a prototype per interview type.
10. **Write up results.** Finally, this step involved the analysis of the data and the creation of the thesis document (White & Marsh, 2006).

3.2 Sample Group & Setting

The sample group for the purposes of the research included two sets of university students at Utrecht University, participating in the Requirements Engineering master's course in the years 2022 and 2023. Certain demographic factors – such as age, gender, ethnicity, nationality, first language and enrolled degree were not considered relevant or disclosed in the study, and therefore did not form part of the identifying factors for the sample group. It can therefore be assumed that members of the sample group were diverse in these factors.

It was expected of the 2022 group to create a prototype from their interviews. The university students from each year were divided into subgroups of approximately four, in which two students

represented the analysts and two students represented the stakeholders. In terms of groups consenting to be part of the research, this amounted to 30 subgroups of students in the 2022 group (the prototype group), and 23 subgroups of students in the 2023 group (the control group). Each subgroup chose a fictional company for which to begin the process of creating or updating an information system. Students then had to conduct two interviews in order to develop a design specification for their chosen company.

The resulting interviews and design specifications were subject to grading according to the Requirements Engineering course grading criteria and then – for the purposes of this research, grouped by the course instructor into four performance categories, namely: *satisfactory, good, very good,* or *excellent*. Subgroups achieving a *very good* or *excellent* result were chosen as suitable for the study (since this research aims to align as closely as possible with professional-level requirements elicitation) and were entered into a random sample generator in order to select 8 subgroups from each year. This resulted in the 16 subgroups used for the research.

3.3 Data Collection & Tools

Data were collected from each of the participating groups. This data includes problem description documents, audio recordings of interviews, and completed requirements specifications. Group B consisted of the 2022 students, who were required to create prototypes during the interview process. These prototypes were created primarily using the Mendix low-code platform, where students had to create high-fidelity, evolutionary prototypes. Group A had no such requirement, hence their use as the control group.

The type of prototype created was not controlled, meaning that participants may have created prototypes with very different characteristics, and not necessarily all conformed to one type of prototyping (for example, one group may have made a simple mock-up, while another may have created a fully-functional MVP). Participant experience in requirements elicitation prior to participation was excluded from the scope, as it is assumed that students participating in the Requirements Engineering course of Utrecht University have participated in the activities thereof, including:

> …the **RE process** and its activities; standards and tools; **agile RE**, user stories; **requirements elicitation**; […] from requirements to architectures; requirements

> **prioritization**; […] (verification of) formal **specifications**… (Utrecht University, n.d.).

The audio files for all groups were transcribed using the Microsoft Office transcription service. This tool allows for the differentiation of speakers, as well as for the inclusion of timestamps and the selection of appropriate dialects (such as American, British, or South African English). It is also a tool which creates no inflated costs to the researcher, and so for these reasons was considered to be appropriate. With regard to analysing these transcripts, NVivo was chosen for tagging, due to its use in research involving coding and the researcher's existing familiarity with the software. NVivo is also a tool that was made available to the researcher by the research institution.

Initially, speaker affect, tone, and body language were not evaluated, since this information was either unavailable or considered irrelevant for the nature of the study. However, the transcriptions obtained from the Microsoft Office transcription service were found to be marginally inaccurate. To remedy this, the audio recordings were used alongside the transcriptions to ensure their accuracy. In this regard, speaker affect, tone, and cadence were considered insofar as they affected the intent or meaning behind particular parts of the interviews.

3.4 Data Analysis

As previously mentioned, the main statistical tests used for the research were the Mann-Whitney U test and the Wilcoxon Signed-Rank test. Briefly, the Wilcoxon Signed-Rank test is a non-parametric statistical test used to compare two sets of scores or data that come from the same participant. In this case, the participant is each subgroup of students, and the two sets of scores are that of the initial and follow-up interviews (Lund & Lund, 2018a). This was used to answer RQ_2.

The Mann-Whitney U test, used to answer RQ_{2A} and RQ_{2B} is also a non-parametric test that is instead used to compare differences between two independent groups where the data gathered are either ordinal or continuous. For this research, the independent groups are the control group and the prototype group, and the data are each category of the coding schema. This same test is performed for both initial (RQ_{2A}) and follow-up interviews (RQ_{2B}). Neither the Mann-Whitney U test nor the Wilcoxon Signed-Rank test assume data normality (Lund & Lund, 2018b).

Something of note is that interviews may differ substantially in terms of their length, the rate of questioning (which can be affected by something as simple as language proficiency), the

proportions of tags, and so on. Therefore, in order to make the data as accurate as possible, the data were normalised. There were two types of normalisation used: Normalisation by time, and normalisation by total tags. It was assumed that normalisation by time was relevant and necessary for all data. However, normalisation by total tags was used to determine whether qualities could potentially be attributed to the rate of questioning, or whether they persisted regardless.

Normalisation by time was performed using this formula:

$$Nt_i = T_i * \frac{a_t}{t_i}$$

where Nt_i *= normalised total for an individual tag i based on average time,* T_i *= total for tag i,* t_i *= interview time,* and a_t *= average time across all interviews*.

Normalisation by total tags was performed similarly, using the formula:

$$NT_i = Nt_i * \frac{a_T}{T_i}$$

where NT_i *= normalised total for an individual tag i based on average total tags,* Nt_i *= normalised total for an individual tag i based on average time,* T_i *= interview total (positive) tags,* and a_T *= average total (positive) tags across all interviews*. Negative tags were not used for the totals, since we assume that the practices included in the total tags are all desirable.

3.5 Ethical Considerations

All documents were anonymised prior to their inclusion in the study. Audio recordings were anonymised as much as possible, and transcriptions were entirely anonymised. Additionally, all participants were given a choice regarding their participation in the study and were requested to complete a consent form.

Briefly, this consent form informed participants of the nature of the study, the implications of their participation, and their right to have their data accessed or erased. They were also given the option of withdrawal at any time. This consent was primarily obtained through the educator involved in the Requirements Engineering course at Utrecht University, and therefore no personal information has been passed on to the researcher. The use of the recorded and transcribed dataset is made

possible thanks to the ethical Science-Geosciences Ethics Review Board of Utrecht University (case S-20339). All participants will have access to the resulting thesis once the research has concluded, through the thesis repository of Utrecht University.

3.6 Threats to Validity

As with all studies, there are several threats to validity to consider when conducting an experiment. No experiment is entirely free of these threats, and it is vital to determine them as early on in the experimental lifecycle as possible. To the extent possible at the time conducted, the threats to validity in the conducted experiment have been addressed or otherwise accepted as a limitation of the research.

3.6.1 Internal Validity

Lack of randomisation was considered as a threat to the study, since the target population was a fixed set of students. This is also a threat to case study research in general, since cases are usually hand-selected to a certain degree. To remedy this insofar as was possible, the participants that were selected from each group were randomised after applying the selection criteria.

Failure to complete tasks is a factor which became relevant, as some of the students either did not provide both recordings or failed to complete the course. However, this particular threat was made irrelevant due to the large number of students in the unsorted sample group, leading to the full 16 subgroups being fulfilled.

Selection bias due to the use of two groups in separate academic years was considered as a threat, since there is always the possibility of minute differences in curriculum, duration, timelines, and so on. However, these differences were not thought of as large enough – at least in the context of the given tasks – to warrant concern. In addition, the selection of only high achievers may bias the results. This was weighed against the desire to have the sample group closely represent professionals. Ultimately, it was decided that representativeness was of a higher priority.

Finally, **Survival bias / mortality** was a treat due to the voluntary nature of participation. Participants were also able to withdraw from the study. This was addressed by gathering a large amount of data from as many participants as possible. In this way, sample mortality was avoided as much as possible, and the desired data were retrieved in the desired amounts.

3.6.2 External Validity

Sampling bias is always a risk when performing studies on students as representatives of professionals. However, since the participants are all masters students taking part in a course which requires a basic knowledge of requirements elicitation and expands on this knowledge, this bias has been slightly reduced. However, it is not possible to remove this bias completely without using professionals as the sample group. In addition, a certain level of sampling bias is likely to be present in the data due to the selection of high-achieving student groups.

Observer bias was also a potential risk of this study, especially since the sample is made up of students in a university course completing tasks for a grade. However, it is assumed that the course lecturer only stressed the importance of good performance insofar as it affected their course grades, rather than the outcome of the study. In this way, observer bias is thought to be minimised, since the data will be analysed using different standards and methods.

Extraneous variables such as year of study, mean age, level of experience, etc. were considered to be trivial for the purposes of the research, since this diversity is considered representative of real-world situations. In addition, both groups are assumed to be similarly diverse, since Utrecht University is an international university that attracts a wide variety of scholars.

4 Refining the Guidelines

As was previously mentioned, a framework was selected, modified, and refined in order to analyse the interviews as effectively as possible. The refinement of the framework and guidelines was a process which originally spanned two months. However, this was extended to five months to ensure that the modifications were sound. This extension was also in part due to unforeseen circumstances which led to a need for the guidelines to be changed dramatically.

Initially, refinement followed a simple iterative process, as demonstrated in Figure 3. In the first two-month period, the researcher collaborated closely with a second researcher. This second researcher was writing their bachelor's thesis based on the application of the framework developed in this thesis. They were expected to not only contribute their ideas to the documentation, but also tag a random sample of the data themselves. In addition to this, the primary supervisor for both parties attended several meetings, monitoring and adding comments where necessary. This was done with the goal of reducing bias as much as possible during the tagging process and achieving high inter-rater reliability.

Meetings occurred approximately once every two weeks to allow parties time for tagging sample interviews, running the agreement analysis in NVivo, and documenting the discrepancies. During these meetings, each party was given the opportunity to explain their thought process regarding a particular tag, after which a discussion was held to determine what the best course of action would be in order to reduce the differences. At the end of each meeting, a small to-do list with deadlines was written and the next meeting was pre-emptively scheduled. The documentation was then updated shortly after each meeting in order to reflect any discussed changes.

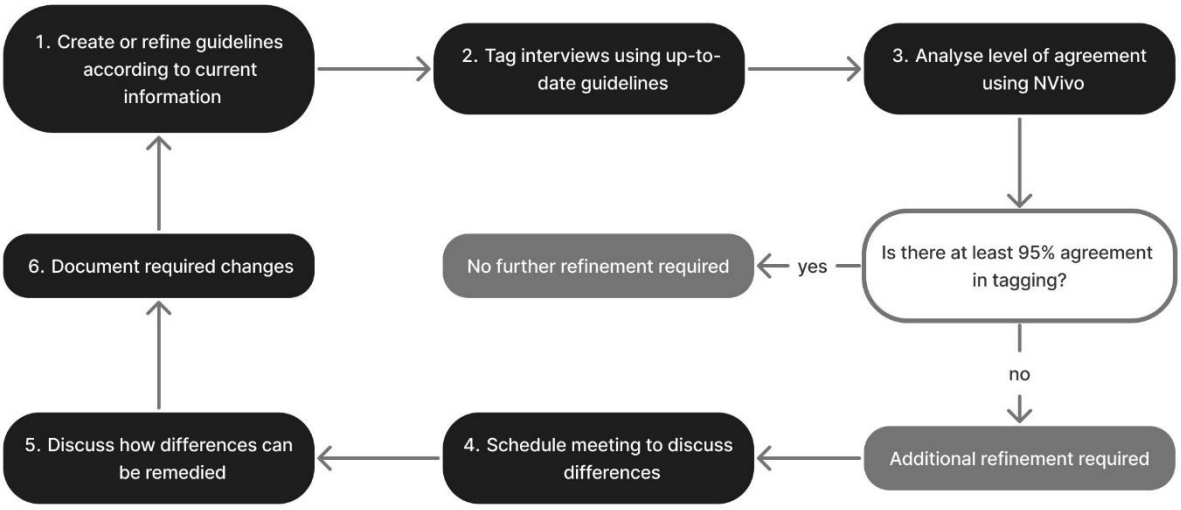


Figure 3. Process followed in order to refine the framework.

Originally, the documentation was intended to be a set of basic guidelines to be executed alongside the practices identified in Mohedas et al. (2022). The practice descriptions, as well as the mistakes identified by Bano et al. (2019), were put into a single spreadsheet matching each practice with its relevant mistakes. This, in turn, was used to create marginally simplified versions of each practice. Finally, the relevant information for each practice was used to create a document of basic guidelines, which were to be used in addition to the spreadsheet (hereafter the "coding schema"). A very brief general description was also added to the document (hereafter the "tagging guidelines" or "coding guidelines"), as seen in Figure 4. The original coding schema explorations, the practice descriptions, the mistakes from Bano, the tag definitions, and the original coding guidelines are seen in appendices D, E, F, G, and H respectively.

Over time, the coding schema was modified to include extended descriptions of each practice and mistake, discovered by consulting a variety of the creators' other works. Definitions for each type of tag (such as *Question*, *Statement*, etc.) were also included so as to avoid misunderstandings and narrow the scope as much as possible. Originally, descriptions of what the absence of, negative use of, or unused opportunity for a tag should look like were also included, as these were going to form a significant part of the analysis of quality for the interviews. Examples for each tag and its respective occurrences (positive, negative, and opportunity) were also described.

FIRST, SOME GENERAL GUIDELINES

Questions should always be tagged in their entirety, including the "?".

Tag the moment of **Opportunity** by tagging the last word of the previous statement (interviewee, usually) and the first word of the following statement (interviewer), for example: "customers. *Speaker 2:* Alright,".

Unless otherwise specified, a **Question** is positive (good quality), and **Opportunities** and **Interview Qualities** are negative (bad quality). A **Statement** can be positive or negative, depending on the given description. No individual tag is both positive and negative!

Figure 4. Original general section of the tagging guidelines.

Eventually, this information was used to modify the tagging guidelines where necessary, filling in blanks and extending instructions or descriptions as much as possible. A –matrix was also occasionally used in order to determine overlaps and differences (see Figure 5).

APPENDIX

The following table is aimed at creating a better understanding of the differences and similarities between some overlapping practices.

	System features	As-is system	To-be system	Presents situation for analysis	Requires in-depth answers	Asks for stakeholder's opinion	Uses stories, metaphors, role-playing, analogies, what-ifs, etc.	Can use domain-specific information	Identifying individual tasks in a business process
Encourage Deep Thinking	x		x	x	x	x			
Use Projective Questioning Techniques				x		x	x		
Use a Co-creative Strategy						x			
Introduce Domain Knowledge								x	
Break Down Expert Tasks	x	x			x			x	x

Figure 5. Matrix used to determine differences between certain tags which often overlapped.

4.1 Refinement per Tag

Each tag in the framework was considered both individually and as a whole alongside all other tags. This was to ensure each tag covered the appropriate content, while not overlapping with any

other tag. Most tags were eventually renamed in order to better suit their intended uses.

4.1.1 Explore As-Is System (formerly *Break Down Expert Tasks*)

In its first iteration, this tag was very simple, containing nothing more than a guideline for how to tag questions in the text (see Figure 6). However, this was found to be extremely difficult to apply, especially considering the significant overlap with the *Introduce Domain Knowledge* practice. It was also found to be too vague in practice, leading to a need for more specific instructions and examples.

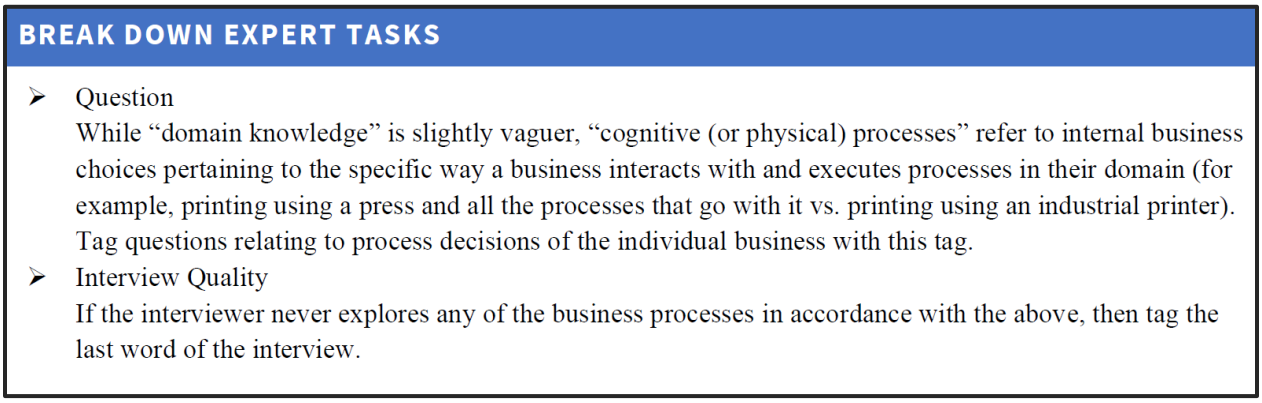


Figure 6. Original version of the *Explore As-Is System* guidelines.

Upon closer inspection, it was discovered that this was the only practice that seemed to handle the as-is system explicitly. It was also the only practice pertaining to existing business tasks or processes. Therefore, this was used to narrow down the scope for this tag. This also led to the name being changed from *Break Down Expert Tasks* to *Explore As-Is System* based on two related assumptions: firstly, that the descriptions of expert tasks tend towards the system as-is; and secondly, that the exploration of the as-is system will automatically include the exploration of current business processes and tasks. This change also made the final tag easier to use, since its intention became significantly easier to identify based on its name alone. The final version of this tag's guidelines is demonstrated in Figure 7.

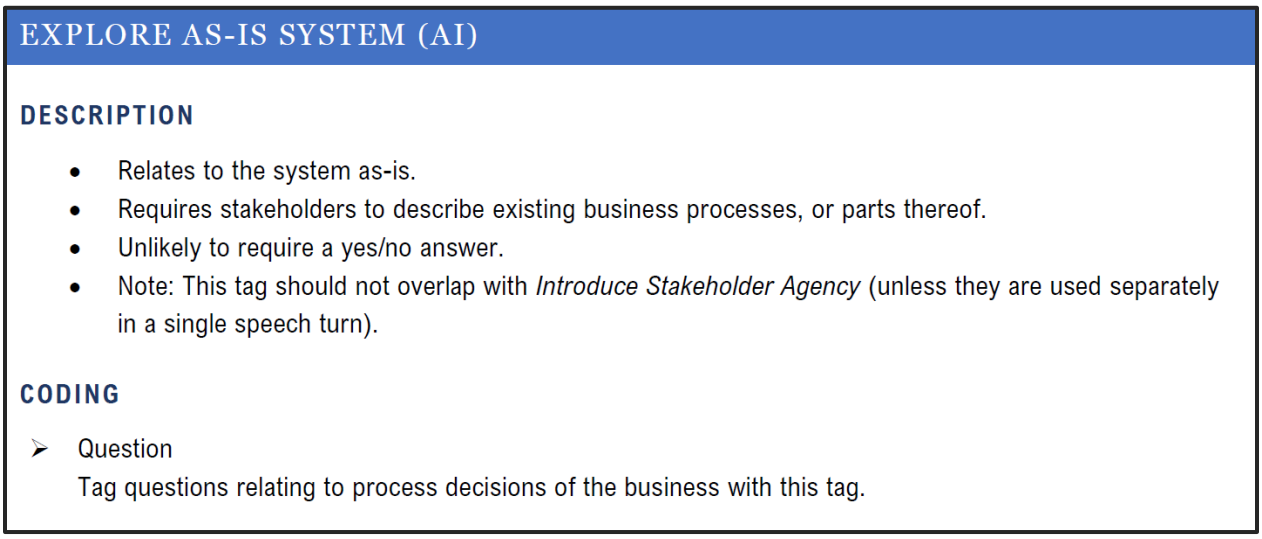


Figure 7. Final version of the *Explore As-Is System* guidelines.

4.1.2 Explore To-Be System (formerly *Encourage Deep Thinking*)

In the original framework, *Encourage Deep Thinking* was defined roughly as making stakeholders think about processes, examples, etc. in a way that yields better, more deep conclusions and uses analytical thinking. Overall, this tag could initially be applied to both the as-is and the to-be situations. However, this caused significant overlap between this tag and the *Break Down Expert Tasks* tag, since both required in-depth answers (i.e. answers not limited to a few words or sentences) about system features. It was also found that most of the description of this practice pertained to the to-be system, with only a small portion of it pertaining to the as-is system. The original guideline description shown in Figure 8 was also severely lacking, since most of the information was contained in the coding schema.

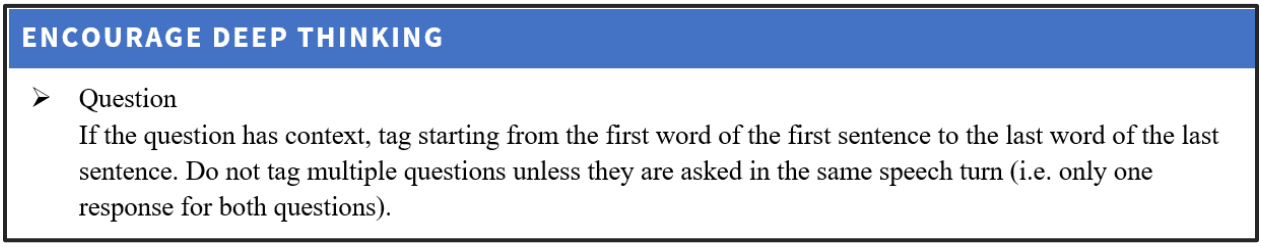


Figure 8. Original version of the *Explore To-Be System* guidelines.

After careful consideration and many iterations, it was decided that the best course of action would

be to overhaul this tag entirely, removing the chance for overlap as much as possible and narrowing the scope to include only the to-be system. It was also decided that this should primarily be used to cover functional requirements, with non-functional requirements only being included insofar as they do not overlap with another tag, *Introduce Stakeholder Agency* (discussed later in this section). To reflect these changes appropriately, and to make the guidelines as consistent and readable as possible, this tag was renamed to *Explore To-Be System* as a counterpart to the *Explore As-Is System* tag above. The final version of these guidelines is demonstrated in Figure 9.

EXPLORE TO-BE SYSTEM (TB)

DESCRIPTION

- Relates to the system to-be (or the refinement of the as-is system).
- Requires the stakeholder to make decisions about the system functions as they pertain to **functional requirements. Non-functional requirements** can also be included in this tag, insofar as they do not expect a stakeholder to be an expert (see *Introduce Stakeholder Agency*). For example, "Do you want the system to be able to run on any device, or just on the computers in the building?" would be acceptable.
- Requires the stakeholder to analyse or assess situations or examples in order to determine the best functionality / requirements.
- Unlikely to require a yes/no answer.

CODING

- Question
  If the question has context that spans multiple speaker turns, tag starting from the first word of the first speaker turn to the last word of the last speaker turn. Otherwise, tag questions pertaining to the to-be system as normal.
- Statement (Negative)
  Interviewer present solutions without any feedback from, or invitation to consider things more deeply to, interviewee.

Figure 9. Final version of the *Explore To-Be System* guidelines.

4.1.3 Develop Rapport with Stakeholders (formerly *Develop a Rapport with the Interviewee*)

In general, *Develop a Rapport with the Interviewee* was applied fairly consistently, with only minor differences in opinion being discovered during the refinement process. This was, in part, due to the simplicity of the descriptions in the coding schema. As a result, the initial guidelines were extremely sparse (see Figure 10).

DEVELOP A RAPPORT WITH THE INTERVIEWEE

- Question
  Tag each question (i.e. with separate answers) individually.
- Interview Quality
  If there are no *Question* tags, tag the first word of the interview to show that the interview does not contain any rapport-building.

Figure 10. Original version of the *Develop Rapport with Stakeholders* guidelines.

Minimal changes were made to the guidelines for this tag. One of the major changes that was made was to ensure that this tag did not overlap with any other tags, since the description heavily implies that any occurrences should not involve "work" (i.e. there should be a deliberate choice to explore the stakeholders' personal lives over the system itself). For consistency, this tag was renamed to *Develop Rapport with Stakeholders* and the guidelines were changed as in Figure 11 to more precisely reflect its scope.

DEVELOP RAPPORT WITH STAKEHOLDERS (DR)

DESCRIPTION

- Analyst asks [personal] questions that do not relate to the business or its processes (**small talk**).
- Analyst makes comments to ease tension, foster a friendly environment, and so on (e.g. compliments).
- Analyst asks questions relating to the stakeholders' background. These can include past and current job experience, **BUT** should not concern their activities within the business and its processes (i.e. something like their job title is fine, but precisely what they do regarding the as-is system should not be considered rapport-building).

CODING

- Question, Statement
  Tag each question (i.e. with separate answers) or statement individually.

Figure 11. Final version of the *Develop Rapport with Stakeholders* guidelines.

4.1.4 Avoid Misinterpretations

The practice *Avoid Misinterpretations* was originally extremely vague and was difficult to separate from other original practices (such as *Explore Contradictions* and *Verify the Conclusions Drawn from Interviews*) when used for tagging interviews. In fact, the tag was one of the most disputed,

being one of the last to achieve a desired level of agreement between raters. The original description also overlapped significantly with the original description of *Verify Conclusions with Stakeholders*, since it calls for the verification of information with stakeholders. As a result, the guidelines (see Figure 12) required major changes, and a huge shift in scope.

AVOID MISINTERPRETATIONS

- Question
  Questions should be confirmatory or clarifying in nature. These will usually take the form of "What do you mean by that?" or "Can you elaborate on that?" – generally non-specific questions relating to clarification.
- Opportunity
  An opportunity of this kind will generally be after an interviewee has said something vague or domain-specific, where an analyst does not appear to comprehend or continue the conversation line.

Figure 12. Original version of the *Avoid Misinterpretations* guidelines.

The resulting guidelines are shown in Figure 13. An explicit difference between *Avoid Misinterpretations* and *Verify Conclusions with Stakeholders* was defined, to reduce the overlap between these as much as was possible (note: since *Explore Contradictions* was removed in the final version of the tagging guidelines, it was not necessary to address any overlap). Many of the ambiguous terms in the original practice description were elaborated on, and a line was drawn between clarifying questions and information-gathering. The tag was not renamed, as it was determined that the issue rested with the practice description.

AVOID MISINTERPRETATIONS (AM)

DESCRIPTION

- Clarifying questions where the analyst is not already confident about a specific answer or solution.
- Analyst is specifically **asking** for an answer, rather than implying or stating that they already have a particular answer in mind.
- Usually takes the form of asking for elaboration, specifying an option or answer between several (i.e. narrowing the scope), etc.
- There should be **grounds for misinterpretation** to occur (i.e. there is information upon which to make a decision) – otherwise it is simply information-gathering.
- **Always** consider whether an AI / TB tag is also relevant.
- The simple difference between *Avoid Misinterpretations* (AM) and *Verify Conclusions with Stakeholders* (VC, below) is that **VC** demonstrates that **a solution is already decided on** to some degree, whereas **AM** implies that **there is no specific solution yet**, or that a mentioned solution is purely for context.
- For example: "You mentioned that students need to input their SSN to get the benefits. Does that mean international students don't get the benefits?" would be considered AM, whereas "So international students are excluded from benefits, since they don't have a SSN," would be considered VC. The wording implies that there is a certain level of **confidence** in the conclusion.
- AM also differs from *Introduce Stakeholder Agency* (SA) in that **AM** presumes that the **stakeholder knows** what the correct option or answer is, whereas **SA** requires the stakeholder to **make a decision**.

CODING

➢ Question
Questions should be clarifying in nature. These will usually take the form of "What do you mean by that?", "Can you elaborate on that?", and so on.

Figure 13. Final version of the *Avoid Misinterpretations* guidelines.

4.1.5 Verify Conclusions with Stakeholders (formerly *Verify the Conclusions Drawn from Interviews*)

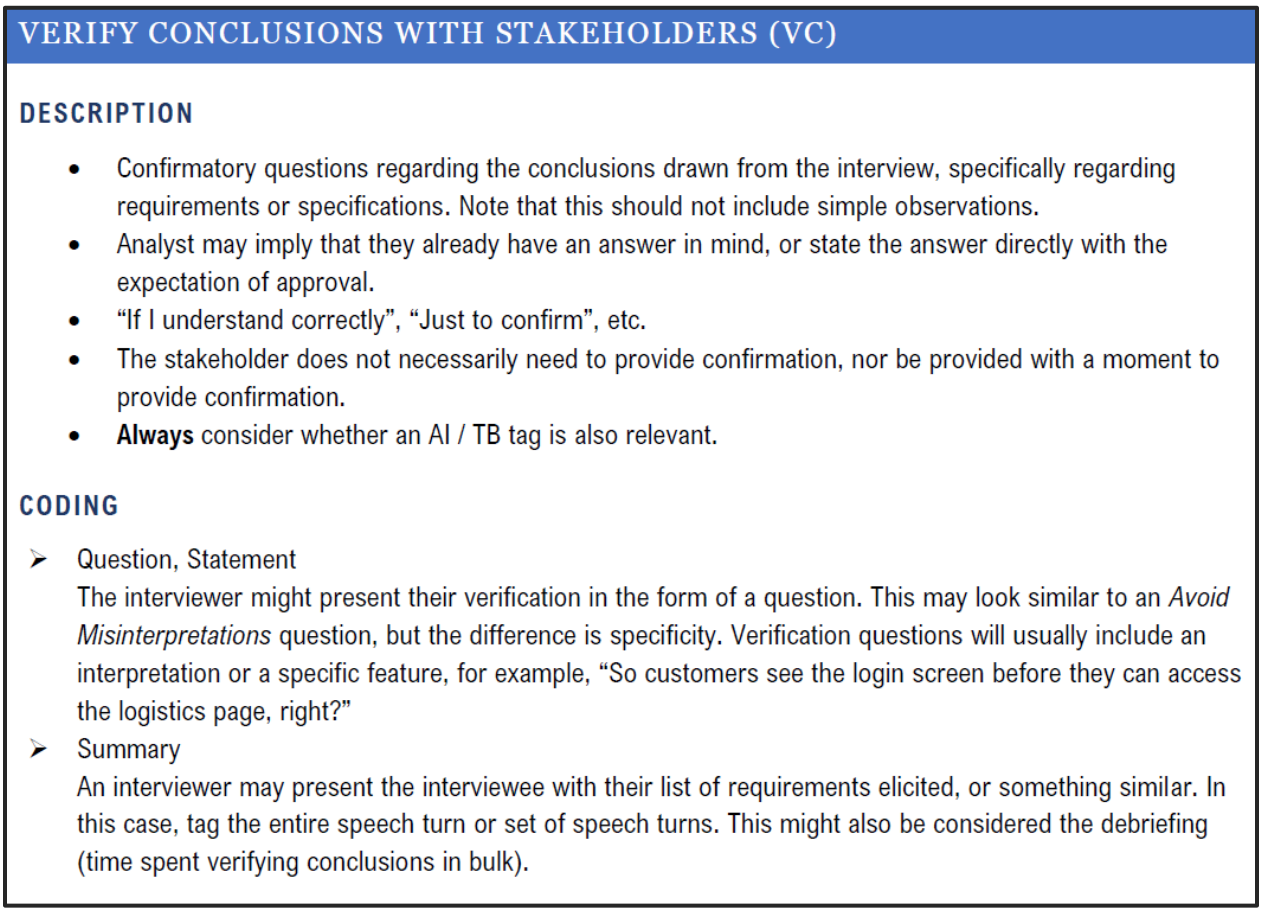
As was previously mentioned, there was significant overlap between *Verify the Conclusions Drawn from Interviews* and the original *Avoid Misinterpretations* practice. As is demonstrated in Figure 14, there was already an attempt to distinguish the tags. However, the guidelines were found to be too vague, even when consulting the practice descriptions in the coding schema. They were also too complicated to apply in their original form, leading to a massive simplification of their use.

VERIFY THE CONCLUSIONS DRAWN FROM INTERVIEWS

- Question
  The interviewer might present their verification in the form of a question. This may look similar to an *Avoid Misinterpretations* question, but the difference is specificity. Verification questions will usually include an interpretation or a specific feature, for example, "So customers see the login screen before they can access the logistics page, right?"
- Statement
  An interviewer may present the interviewee with their list of requirements elicited, or something similar. In this case, tag the entire speech turn. This might also be considered the debriefing.
- Interview Quality
  If there is no debriefing of *this* kind at the end of the interview, tag the last word of the interview. Debriefing here refers to time spent verifying conclusions.

Figure 14. Original version of the *Verify Conclusions with Stakeholders* guidelines.

The final version of the guidelines demonstrated in Figure 15 were found to reduce the overlap adequately, especially when used in conjunction with the final description of the *Avoid Misinterpretations* guidelines. It was also made to be more specific, and the types of tags were simplified so that instances were clearer and easier to analyse.



VERIFY CONCLUSIONS WITH STAKEHOLDERS (VC)

DESCRIPTION

- Confirmatory questions regarding the conclusions drawn from the interview, specifically regarding requirements or specifications. Note that this should not include simple observations.
- Analyst may imply that they already have an answer in mind, or state the answer directly with the expectation of approval.
- "If I understand correctly", "Just to confirm", etc.
- The stakeholder does not necessarily need to provide confirmation, nor be provided with a moment to provide confirmation.
- **Always** consider whether an AI / TB tag is also relevant.

CODING

- Question, Statement
  The interviewer might present their verification in the form of a question. This may look similar to an *Avoid Misinterpretations* question, but the difference is specificity. Verification questions will usually include an interpretation or a specific feature, for example, "So customers see the login screen before they can access the logistics page, right?"
- Summary
  An interviewer may present the interviewee with their list of requirements elicited, or something similar. In this case, tag the entire speech turn or set of speech turns. This might also be considered the debriefing (time spent verifying conclusions in bulk).

Figure 15. Final version of the *Verify Conclusions with Stakeholders* guidelines.

4.1.6 Present Non-Superficial Introduction & Allocate Stakeholder Feedback Time (formerly *Designer Begins and Interviewee Concludes*)

Originally, the two respective tags *Present Non-Superficial Introduction* and *Allocate Stakeholder Feedback Time* formed one singular tag: *Designer Begins and Interviewee Concludes* (demonstrated in Figure 16). The practice description was relatively clear, but some disagreement on what constituted an appropriate introduction and conclusion did occur. This led to the need to define these more explicitly.

As a result, the guidelines were modified to exclude superficial introductions and non-explicit setting aside of feedback time. In addition, it was decided that the introduction and conclusion tags should be kept separate, since they demonstrate slightly different intentions from the analysts. For example, *Present Non-Superficial Introduction* demonstrates a focus on interview structure,

whereas *Allocate Stakeholder Feedback Time* demonstrates a focus on the stakeholders' opinions and feedback. These changes are demonstrated in Figure 17 and Figure 18 respectively.

DESIGNER BEGINS AND INTERVIEWEE CONCLUDES

- Statement
  If the interviewer gives a briefing (or introduction) for the interview at the beginning, tag the entirety of the introductory section.
  If the interviewer gives time at the end of the interview for the interviewee to address anything not covered by the questions, then tag the initiating sentence(s) or speech turn.
- Interview Quality
  If there is no briefing at the beginning of the interview, tag the first word of the interview.
  If there is no debriefing of *this* kind at the end of the interview, tag the last word of the interview.
  Debriefing here refers to allowing time for interviewee concluding thoughts.

Figure 16. Original version of the now-split *Present Non-Superficial Introduction* and *Allocate Stakeholder Feedback Time* guidelines.

PRESENT NON-SUPERFICIAL INTRODUCTION (PI)

DESCRIPTION

- Analyst explains the **purpose** of the interview at the **beginning** (e.g., their goals and the design project). This should include things like the structure of the interview, the designers' names, etc.
- Superficial, unstructured introductions are not considered part of this tag. For example, an introduction where analysts introduce themselves, and then simply state "We will be interviewing you today. Let's start off with …" does not provide sufficient structure regarding the interview timeline.

CODING

- Interview Quality
  If the interviewer gives a briefing (or introduction) for the interview at the beginning, tag the entirety of the introductory section, INCLUDING the first speaker name.

Figure 17. Final version of the *Present Non-Superficial Introduction* guidelines.

ALLOCATE STAKEHOLDER FEEDBACK TIME (FT)

DESCRIPTION

- Analyst leaves time at the **end** of the interview to allow the stakeholder to offer any **concluding thoughts** that were **not covered by questions.**
- This should be an explicit setting aside of time to explore topics, questions, etc. brought up by the stakeholders.
- This is **not** considered a use of SA.

CODING

- Interview Quality
  If the interviewer gives time at the end of the interview for the interviewee to address anything not covered by the questions, then tag the initiating speech turn.

Figure 18. Final version of the *Allocate Stakeholder Feedback Time* guidelines.

4.1.7 Use Hypotheticals and Examples (formerly *Use Projective Questioning Techniques*)

Use Projective Questioning Techniques, demonstrated in Figure 19, was initially very broad, including everything from storyboarding and roleplay to hypothetical questions. This caused a lot of differences in tagging, since (a) it was not always clear when something like an artifact or a paper mock-up was present in the interview room and (b) there was often some confusion as to what was a basic process question and what was a question pertaining to a hypothetical situation. It was also difficult to provide distinguishing examples for this particular tag usage. Overall, the scope for this particular practice needed to be significantly narrowed and made more explicit.

USE PROJECTIVE QUESTIONING TECHNIQUES

- Question
  These questions might follow on from an interviewee's description of a process or feature, for example: "How would you handle a customer ordering something that is out-of-stock?"
- Interview Quality
  If the interviewer never uses hypotheticals or storytelling in their questions, then tag the last word of the interview.

Figure 19. Original version of the *Use Hypotheticals and Examples* guidelines.

It was eventually decided that "projective questioning techniques" was simply too broad, and so

the name was changed to *Use Hypotheticals and Examples*, since this more closely described the actual practice being used. Additionally, cases not specifically including a hypothetical situation or an example were excluded. This was to minimise confusion when applying the tag. Additional guidelines were also added to differentiate this tag from basic as-is and to-be questions (see Figure 20).

USE HYPOTHETICALS AND EXAMPLES (HE)

DESCRIPTION

- Analyst frames questions using hypothetical situations, examples and outliers.
- These mostly include situations which stray from the absolute standard process (i.e. the "perfect world").
- This can also include giving examples to better clarify the details of a process or answer. For example, if an analyst describes a scenario where a particular step in a process has been reached, and then asks for further information (note: the provision of an example or hypothetical is **required** for this tag!).
- If the **answer would change / be less specific** without the use of the example or hypothetical, then do **not** use this tag (for example: "So their information – for example date of birth or ID – gets stored in their patient file?").
- This tag will almost always be used alongside another tag (e.g. **Explore to-be system**).

CODING

➢ Question
These questions might follow on from an interviewee's description of a process or feature, for example: "How would you handle a customer ordering something that is out-of-stock?"

Figure 20. Final version of the *Use Hypotheticals and Examples* guidelines.

4.1.8 Introduce Stakeholder Agency (formerly *Use a Co-Creative Strategy*)

Initially, the *Use a Co-Creative Strategy* tag was rather vague and led to many disagreements as to what constituted co-creativity in the context of the used interviews. With the practice being defined as analysts "increas[ing] the stakeholder's sense of ownership of interview/product requirements," the precise nature of "ownership" becomes an issue. Despite the simplification demonstrated in Figure 21, the application of the tag required a major overhaul.

USE A CO-CREATIVE STRATEGY

- Question
  Tag questions that give the interviewee an opportunity to provide active feedback on the requirements or artifacts, or which give the interviewee an opportunity to control the interview.
- Statement
  Statements which give interviewees a stake in, or a responsibility for, the process or product will use this tag. Statements which share the interview control with the interviewee are also valid here. Be sure to tag the entire sentence or group of sentences, starting with the first word of the first sentence and ending with the last word of the last one. (Positive)

Figure 21. Original version of the *Introduce Stakeholder Agency* guidelines.

This overhaul resulted in the guidelines shown in Figure 22. As demonstrated, many changes were made to the guidelines, and a lot of specific situations were used to identify appropriate tagging opportunities. There are also several distinctions between other tags and scenarios, such as *Allocate Stakeholder Feedback Time*. The intention of the practice was kept much the same, with more elaborate explanations. Additionally, the name was changed to *Introduce Stakeholder Agency*, as it was determined that the practice essentially aimed at promoting stakeholder agency during the interview process. A distinction was also added between positive and negative uses of the tag, since it is possible for analysts to give stakeholders agency over something that should probably be handled by the analysts themselves.

INTRODUCE STAKEHOLDER AGENCY (SA)

DESCRIPTION

- Questions or statements that give stakeholders a more **significant role** in the final system design.
- Questions or statements where analyst and stakeholder might **jointly** come up with a solution.
- Analyst may give stakeholders the opportunity to provide **active feedback** on the requirements elicited thus far.
- Analyst may cede control of the interview to the stakeholders (this excludes *Allocate Stakeholder Feedback Time*).
- **In short,** this requires stakeholder to **make decisions** and **give their opinions** rather than provide **existing information** about the business and its goals, plans, or objectives.
- Giving stakeholders **too much control** should be considered a **negative** use of this principle. For example, asking stakeholders **highly technical questions**, such as questions pertaining to **non-functional requirements** (e.g. "How many milliseconds of delay would you be expecting between clicking *submit* and returning to the home screen?") gives stakeholders responsibility for something that should be the responsibility of developers, and expects them to have technical knowledge not usually possessed by standard stakeholders.
- In general, exchanges where it does not make sense to include a stakeholder in a decision, or where the analyst is expecting the stakeholder to be an expert (in something other than their own business dealings) should be considered **negative**.
- **Note:** This tag should not overlap with *Explore As-Is System* (unless they are used separately in a single speech turn).

CODING

- Question, Statement (Positive)
  Tag **questions** that give the interviewee an opportunity to provide active feedback on the requirements or artifacts, or which give the interviewee an opportunity to control the interview.
  Statements which give interviewees a stake in, or a responsibility for, the process or product will use this tag. Statements which share the interview control with the interviewee are also valid here.
- Question, Statement (Negative)
  If an analyst is asking questions or making statements which demand the stakeholders come up with a solution, or require them to have technical knowledge beyond what should be reasonably expected of a standard stakeholder, use this negative tag.

Figure 22. Final version of the *Introduce Stakeholder Agency* guidelines.

4.1.1 Be Flexible and Opportunistic – REMOVED

At the start of the process, the *Be Flexible and Opportunistic* tag was difficult to apply. In general, the distinction between analysts' prepared questions and the questions arising as a result of discussion was hard to identify, especially without additional context from the analysts themselves

– their prepared questions were not shared with the researchers. As a result, the guidelines were not very specific in their first iteration, as seen in Figure 23 below.

BE FLEXIBLE AND OPPORTUNISTIC

- Question
  Question should follow on from a topic the interviewee arrived at.
- Statement
  Tag statements which are made to shut down conversation: "Let's move on," "Let's get back on track", etc. Tag only the statement which shuts down the conversation, not the entire speech turn. (Negative)
- Opportunity
  An opportunity of this kind will generally be after an interviewee has started discussing a new topic, or a topic not initially asked about by the interviewer. If there is already a *Statement* tag, then this tag is not necessary.

Figure 23. Original version of the *Be Flexible and Opportunistic* guidelines.

Eventually, the guidelines demonstrated in Figure 24 were formulated. As can be seen, there are minimal guidelines, and there are very few differences between the original and the most recent versions. Once again, the lack of context proved to be the major issue when applying the tag, leaving a lot of the process up to the individual opinions of the taggers.

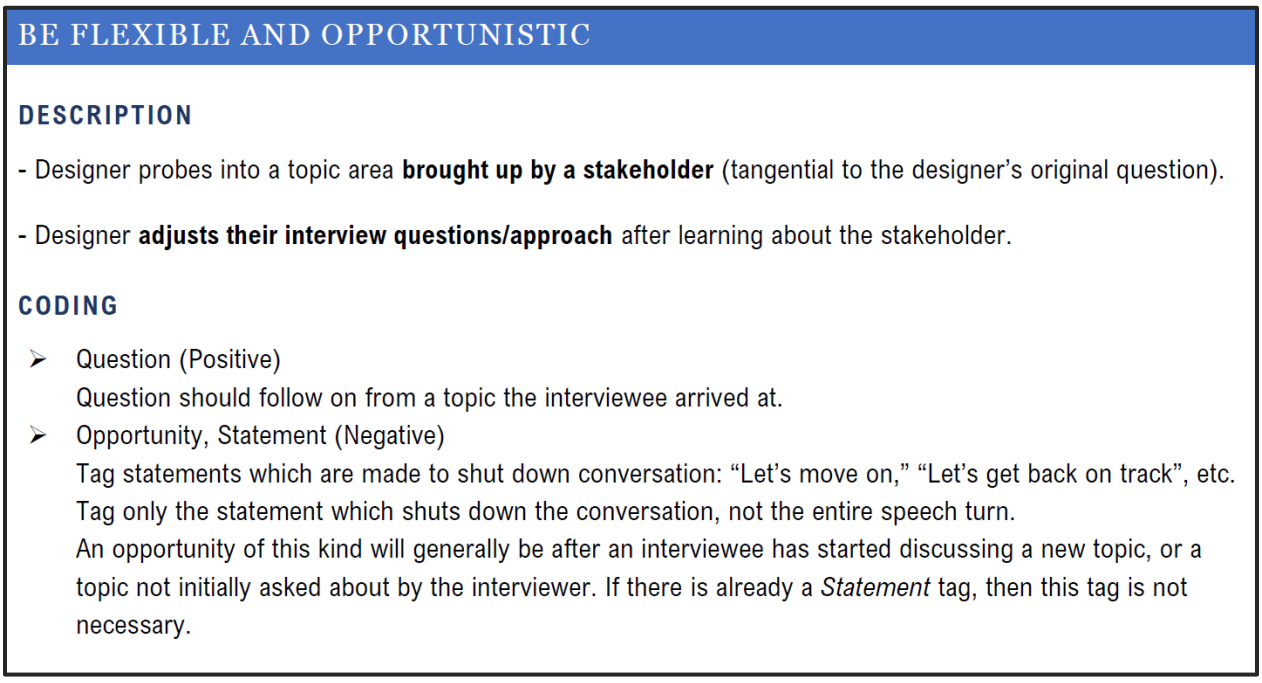


Figure 24. Most recent version of the *Be Flexible and Opportunistic* guidelines.

Ultimately, it was decided that the practice was too ambiguous in nature without a variety of contextual information, and these guidelines were removed from the document.

4.1.2 Introduce Domain Knowledge – REMOVED

Like the previous tag, *Introduce Domain Knowledge* was initially found to be quite difficult to use. Domain knowledge is relatively difficult to distinguish from business information, as analysts may not know precisely where the line is between the two. The first set of guidelines in Figure 25 attempted to make a minor distinction by identifying jargon and domain concepts as markers of this tag.

INTRODUCE DOMAIN KNOWLEDGE

- Question
  Domain knowledge refers to the field in general (for example, paper printing). It is not specific to the business itself. Interviewers might demonstrate this knowledge by asking questions that use jargon or refer to concepts within the domain.
- Opportunity
  If an interviewee brings up something domain-specific and the interviewer does not explore it (where it is clear that exploration would be beneficial), or the interviewer asks a basic question where a domain-specific question could have replaced it, then this tag is used.

Figure 25. Original version of the *Introduce Domain Knowledge* guidelines.

Refining this tag proved difficult, and little to no changes were made between the original guidelines and the most recent version thereof. It was also discovered that there was significant overlap between *Introduce Domain Knowledge* and *Break Down Expert Tasks*, since the previously mentioned distinction between the domain and the business was troublesome to identify, especially without having expertise in the subject matter of each individual interview. Therefore, the guidelines remained much the same, as seen in Figure 26.

INTRODUCE DOMAIN KNOWLEDGE

DESCRIPTION

Designer uses **domain knowledge** from prior interview or observation or literature or any other information source to frame question. Domain knowledge refers to the field in general (for example, paper printing). It is not specific to the business itself.

CODING

- Question, Statement (Positive)
  Interviewers might demonstrate domain knowledge by asking questions or making observations (or some combination of the two) that use jargon or refer to concepts within the domain. In the case where domain knowledge is contained in context rather than a single question, tag the context with the question.
- Opportunity (Negative)
  If an interviewee brings up something domain-specific and the interviewer does not explore it (where it is clear that exploration would be beneficial), or the interviewer asks a basic question where a domain-specific question could have replaced it, then this tag is used.

Figure 26. Most recent version of the *Introduce Domain Knowledge* guidelines.

It was eventually decided that this tag be excluded, since removing the overlap between it and *Explore As-Is System* would be an unnecessarily difficult task. Moreover, it was determined that the distinction was not entirely necessary for the purposes of the research.

4.1.1 Explore Contradictions – REMOVED

One of the more straightforward practices was that of *Explore Contradictions*, since the premise is very simple: if there is a contradiction, it should be addressed. However, as pointed out in the guidelines in Figure 27, it would be extremely time-consuming to identify all contradictions across interviews, and so the scope of this tag was reduced to include only those contradictions occurring in close verbal proximity.

EXPLORE CONTRADICTIONS

- Question
  This should be fairly straightforward to spot.
- Opportunity
  Spotting contradictions that span entire interviews is a bit time-consuming. Therefore, only use this to tag contradictions happening within one to three speech turns of each other. Tag at the end of the speech turn where the second contradicting statement occurred (and the beginning of the next, as usual).

Figure 27. Original version of the *Explore Contradictions* guidelines.

Due to the limited scope of tagging, the guideline definition was extended to include other types of contradictions, as defined in Figure 28. However, no other significant changes were made.

EXPLORE CONTRADICTIONS

DESCRIPTION

Designer asks about **discrepancies** within a stakeholder's own responses or **differences** between different stakeholders' responses.

CODING

- Question (Positive)
  This should be fairly straightforward to spot – interviewer will mention a difference, or ask for clarification as to why different answers were given, etc.
- Opportunity (Negative)
  Spotting contradictions that span entire interviews is a bit time-consuming. Therefore, only use this to tag contradictions happening within one to three speech turns of each other. Tag at the end of the speech turn where the second contradicting statement occurred (and the beginning of the next, as usual).

Figure 28. Most recent version of the *Explore Contradictions* guidelines.

This guideline was also eventually removed, as it was deemed too rare, time-consuming, and irrelevant for the research.

4.1.2 Have the Interviewee Teach You – REMOVED

Finally, there was originally a section in the guidelines intended for the practice *Have the Interviewee Teach You*. However, it was very quickly determined that this would be impossible to tag, as it required analysts to feign ignorance – something that cannot be determined using only

audio. Therefore, this section was removed from the very beginning of the process.

5 Findings

In this chapter shall be presented and discussed the acquired results of the research. Firstly, some general statistics are shown. Then the results comparing the initial interviews of the control group to the initial interviews of the prototype group are considered. Thereafter are given the results comparing the follow-up interviews in the same way. Finally, the differences between overall initial interviews and overall follow-up interviews are demonstrated.

5.1 General Statistics

On average, interview durations were about the same between initial and follow-up interviews, as well as between the control group and prototype group. There was a large difference in total tags between the initial and follow-up interviews for the control group (86.25 vs. 128.13), whereas for the prototype group there was no change (73.63 for both). This also means that the follow-up interviews of the control group demonstrated the most tags per minute, at 2.56 tags per minute.

As a reminder, each interview's tags were adjusted according to the average time for conducting an interview using the formula:

$$Nt_i = T_i * \frac{a_t}{t_i}$$

where Nt_i *= normalised total for an individual tag i based on average time,* T_i *= total for tag i,* t_i *= interview time,* and a_t *= average time across all interviews*.

These results were then further adjusted for a second round of analysis using the following formula:

$$NT_i = Nt_i * \frac{a_T}{T_i}$$

where NT_i *= normalised total for an individual tag i based on average total tags,* Nt_i *= normalised total for an individual tag i based on average time,* T_i *= interview total (positive) tags,* and a_T *= average total (positive) tags across all interviews*.

After this normalisation, the results regarding the means were as represented in Table 4 for the initial interviews, and as represented in Table 5 for the follow-up interviews.

Table 4. Mean statistics for all tags in initial interviews normalised by time and total tags.

Group		AI	TB	DR	AM	VC	VC (summary)	HE	SA (pos)
Control	Mean	24.346	28.099	1.234	7.163	14.433	0.531	6.351	6.819
	N	8	8	8	8	8	8	8	8
	Std. Dev.	15.535	11.641	1.701	3.153	6.551	0.7999	5.002	4.031
Prototype	Mean	29.924	20.946	4.055	8.924	12.683	2.003	1.675	8.768
	N	8	8	8	8	8	8	8	8
	Std. Dev.	10.809	11.131	4.241	5.188	8.596	1.893	2.689	5.257
Total	Mean	27.135	24.523	2.644	8.043	13.558	1.267	4.013	12.099
	N	16	16	16	16	16	16	16	16
	Std. Dev.	13.246	11.606	3.445	4.246	7.480	1.596	4.570	4.782

Table 5. Mean statistics for all tags in follow-up interviews normalised by time and total tags.

Group		AI	TB	DR	AM	VC	VC (summary)	HE	SA (pos)
Control	Mean	6.186	41.350	0.758	6.879	15.336	0.631	5.250	12.583
	N	8	8	8	8	8	8	8	8
	Std. Dev.	7.128	3.305	0.889	5.046	4.718	0.779	5.484	5.919
Prototype	Mean	9.225	37.170	2.740	8.738	14.260	1.816	3.411	11.615
	N	8	8	8	8	8	8	8	8
	Std. Dev.	8.405	11.501	3.617	5.947	5.252	1.433	2.639	3.666
Total	Mean	7.706	39.260	1.749	7.808	14.798	1.224	4.331	12.099
	N	16	16	16	16	16	16	16	16
	Std. Dev.	7.690	8.455	2.743	5.413	4.855	1.271	4.264	4.782

5.2 Overall Interview Qualities

5.2.1 Normalised by Time Only

When looking at the difference between initial and follow-up interviews across all groups, there are several interesting results, as demonstrated in Table 6. The Wilcoxon Signed Ranks test was used to analyse the data. When normalising the data using **time only**, there is a significant difference between the *explore-as-is* value, *explore system to-be* value, the *introduce stakeholder agency (positive)* tag, and the *introduce stakeholder agency (negative)* tag when comparing initial and follow-up interviews. The AI tag was used significantly more in the initial interviews, with $z = -3.362$, $p < 0.001$. Regarding the TB tag, there was a statistically significant increase in the median use thereof, with $z = -2.43$, $p = 0.015$. For the SA (pos) tag, follow-up interviews also demonstrated a significant increase in the median use thereof, with $z = -2.12$, $p = 0.034$. Finally, the SA (neg) tag was used significantly more in follow-up interviews, where $z = -2.578$, $p = 0.01$.

Table 6. Results of the Wilcoxon Signed Ranks test on all interviews normalised by time only.

	AI	**TB**	TB (neg)	DR	AM	VC	VC (summary)	HE	**SA (pos)**	**SA (neg)**
Z	**-3.362**[b]	**-2.43**[c]	-1.604[c]	-.565[b]	-.362[c]	-.879[c]	-1.067[c]	-.341[c]	**-2.12**[c]	**-2.578**[c]
Asymp. Sig. (2-tailed)	**<.001**	**.015**	.109	.572	.717	.379	.286	.733	**.034**	**.010**

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

c. Based on positive ranks.

When we consider what these results might mean, some interesting conclusions emerge. As a reminder, these results were aimed at answering the following research question:

> **RQ_2:** To what extent does the type of interview affect the qualities of requirements elicitation interviews, regardless of prototype creation?

where the hypothesis was as follows:

> **$H_{0\text{-}2}$:** Interview type has no effect on the qualities of requirements elicitation interviews.

As was previously mentioned, the AI tag was used significantly more in initial interviews than in follow-up interviews, with $z = -3.362$, $p < 0.001$. This aligns with the hopes of the researcher, as

it is a general expectation that the initial interview is mainly exploratory in nature while the analysts determine missing information about the system and the stakeholders' desires. In other words, there is more exploration of the as-is system in initial interviews than follow-up interviews (Hickey & Davis, 2003; Schneider, 2007).

Conversely, the TB tag occurred significantly more often in follow-up interviews, with $z = -2.43$, $p = 0.015$. Again, this is in alignment with researcher expectations, as – at least in a context where less than a handful of interviews are conducted – follow-up interviews should be likely to contain more questions regarding the to-be system. This is assumed in part due to the previous result regarding the AI tag, since time not taken up by the as-is system should, in general, be taken up by the to-be system. Another part of the assumption is that, once analysts have begun to gather enough information about the old system or situation, they will begin to use that information to formulate questions about the new system.

Regarding the use of the SA (pos) and SA (neg) tags, it is unclear why there was a significant increase in the use of these tags in follow-up interviews. One explanation is that follow-up interviews might, by nature, include more usage of stakeholder agency in general, since a lot of to-be questions are expected to include brainstorming, asking for opinions, determining solutions as a team, and so on. Another explanation is that this was simply a fluke, as these tags did not occur as often as many of the others, and so it is possible that these tags should be observed in more settings and in greater amounts.

Overall, interview type appears to be correlated with a change in the qualities of requirements elicitation interviews, specifically regarding the use of the practices *Explore As-Is System, Explore To-Be System,* and *Introduce Stakeholder Agency*. Therefore, we are able to reject the null hypothesis $H_{0\text{-}2}$ when normalising by time only.

5.2.2 Normalised by Time and Total Tags

If we normalise these results using **time and total tags,** the above results change as seen in Table 7.

Table 7. Results of the Wilcoxon Signed Ranks test on all interviews normalised by time and total tags.

	AI	**TB**	**DR**	AM	VC	VC (summary)	HE	**SA (pos)**
Z	**-3.156**[b]	**-3.361**[c]	**-2.103**[b]	-.31[b]	-.1.189[c]	-.0[d]	-.426[c]	**-2.534**[c]
Asymp. Sig. (2-tailed)	**<.001**	**<.001**	**.0.35**	.756	.234	1	.67	**.011**

a. Wilcoxon Signed Ranks Test

b. Based on positive ranks.

c. Based on negative ranks.

d. The sum of negative ranks equals the sum of positive ranks.

We now see a significance emerge in the use of the practice *develop rapport with stakeholders.* The DR tag was used more often in follow-up interviews than in initial interviews, with $z = -2.103$, $p = 0.035$. This equates to an effect size where $\eta^2 = 0.276$, $d = 1.236$. In other words, the effect is large enough to be perceived in everyday life. This can be further observed in **Figure 29**.

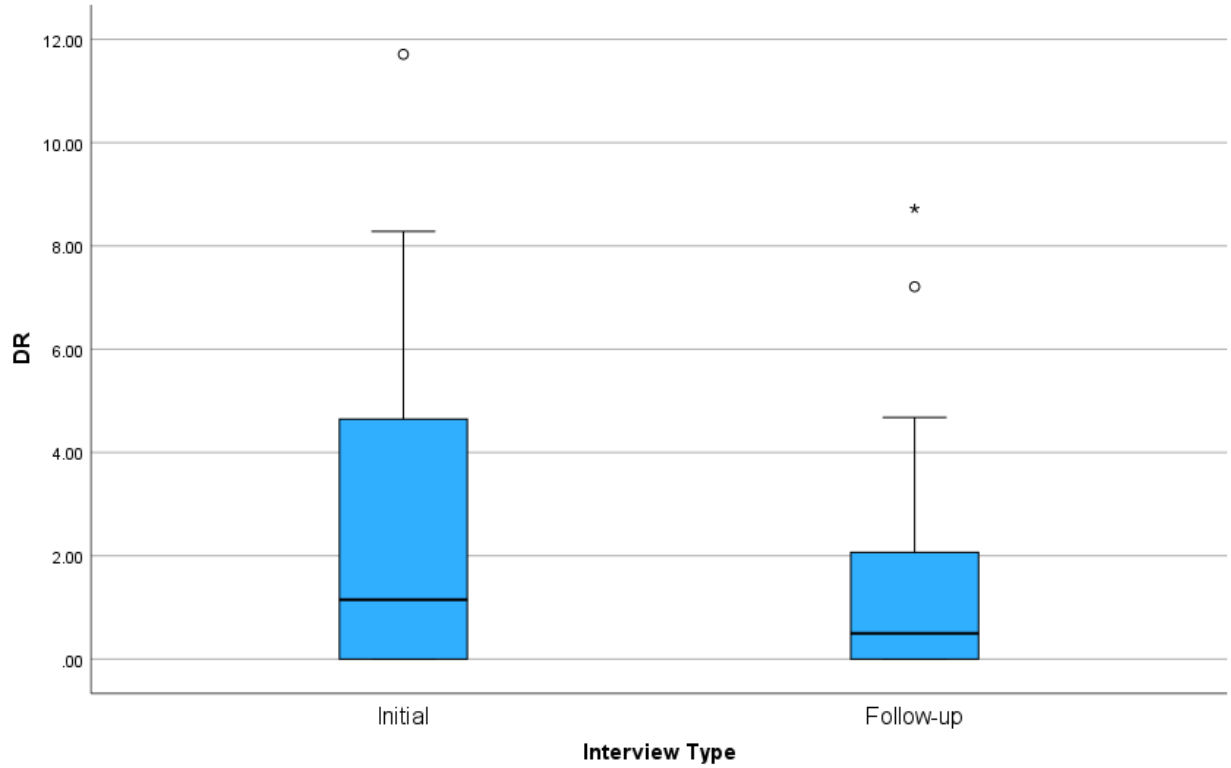


Figure 29. Box-and-whisker plot representing the difference between initial and follow-up interviews regarding the incidence of the HE tag.

It is unclear why this result occurred. It is possible that the follow-up interview is more comfortable for participants, since they are now familiar with each other to a certain extent. It's also possible that, as with the above, it was simply due to something else. For example, it could be possible that the change in the analyst running the interview more often than not led to more rapport being formed with stakeholders than with the analysts in charge in previous interviews. Another possibility is simply that, due to the participants being university students, they are still in the process of learning how to conduct interviews properly. Regardless, with this result, we are able to reject the null hypothesis $H_{0\text{-}2}$ when normalising by time and total tags.

5.3 Initial Interview Qualities

As a reminder, the following section has the goal of answering the following research question, with its null hypothesis:

RQ_{2A}: To what extent does the need to develop a prototype affect the qualities of

initial requirements elicitation interviews?

$H_{0\text{-}2A}$: The need to develop a prototype has no effect on the qualities of initial requirements elicitation interviews.

A Mann-Whitney U test was performed in order to determine whether differences existed between the control group and the prototype group with regard to initial interviews. The results of this test are presented in Table 8. According to the data, none of the median scores for recommended practices were shown to be statistically different between the two groups, with $p > 0.05$ for all measured practices. The practices that came closest were *explore to-be system* and *introduce stakeholder agency*, with $U = 14$, $z = -1.89$, $p = 0.059$ and $U = 18$, $z = -1.47$ and $p = 0.141$ (using the asymptotic p-value) respectively. This supports the null hypothesis for all tags.

Table 8. Results of the Mann-Whitney U test on all initial interviews normalised by time only.

	AI	TB	TB (neg)	DR	AM	VC	VC (summary)	HE	SA (pos)	SA (neg)
Mann-Whitney U	29	14	20	30	21	20	22	24	18	23
Wilcoxon W	65	50	56	66	57	56	58	60	54	59
Z	-.315	-1.89	-1.849	-.219	-1.155	-1.26	-1.078	-.841	-1.47	-.97
Asymp. Sig. (2-tailed)	.753	.059	.064	.826	.248	.208	.281	.4	.141	.332
Exact Sig. [2*(1-tailed Sig.)]	.798[b]	.065[b]	.234[b]	.878[b]	.279[b]	.234[b]	.328[b]	.442[b]	.161[b]	.382[b]

a. Grouping Variable: Year

b. Not corrected for ties.

If we assume that all groups performed the same number of recommended practices and normalise the data in this way, the results change subtly, as per Table 9. Adjusting the data in this manner yields one significant result for the median of HE, where $U = 11$, $z = -2.239$ and $p = 0.025$ using the asymptotic p-value. In other words, there is a significant difference between the amount of *use hypotheticals and examples* tags in initial interviews when comparing the control group and the prototype group, where the control group uses this tag more often. This therefore means the null hypothesis can be rejected concerning the HE tag.

If one calculates the effect size for this tag in this context, the results are $\eta^2 = 0.304$, $d = 1.322$. We can therefore discern that there is a very large or significant effect size concerning the HE tag, meaning the difference between the control and prototype groups is easily observable. This

difference is further illustrated in Figure 30 in the form of a box-and-whisker plot.

Table 9. Results of the Mann-Whitney U test on all initial interviews normalised by time and total tags.

	AI	TB	DR	AM	VC	VC (summary)	**HE**	SA (pos)
Mann-Whitney U	27	20	19	28	27	17.5	**11**	25
Wilcoxon W	63	56	55	64	63	53.5	**47**	61
Z	-.525	-1.26	-1.386	-.42	-.525	-1.627	**-2.239**	-.736
Asymp. Sig. (2-tailed)	.6	.208	.166	.674	.6	.104	**.025**	.462
Exact Sig. [2*(1-tailed Sig.)]	.645[b]	.234[b]	.195[b]	.721[b]	.645[b]	.13[b]	**.028[b]**	.505[b]

a. Grouping Variable: Year

b. Not corrected for ties.

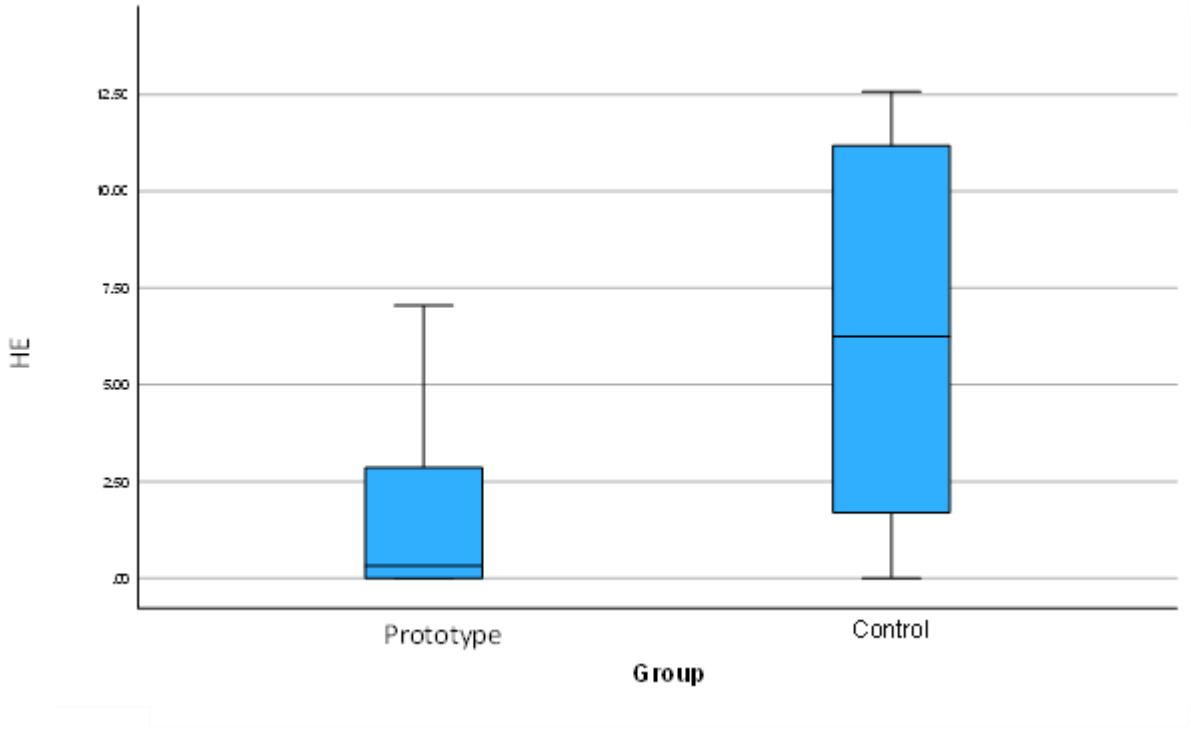


Figure 30. Box-and-whisker plot representing the difference between the control group and the prototype group regarding the incidence of the HE tag.

In terms of what these results mean, HE tag was used proportionally more often in the control group than in the prototype group. This result is in line with researcher expectations, since it is

expected that the control group will engage more in abstractions and hypotheticals. In initial interviews in particular, it is possible that this is due to more specific questions being asked in the prototype group to facilitate better prototype creation. Put differently, analysts may have avoided hypotheticals, and simply ensured they had all the pertinent information for a particular functionality in order to build a mock system rather than ensure they covered all bases. Beyond this, it is unclear why there is such a significant difference here.

5.4 Follow-Up Interview Qualities

Similarly to the previous section, this section is aimed at answering the following research question, with its null hypothesis:

> **RQ_{2B}:** To what extent does the need to develop a prototype affect the qualities of *follow-up* requirements elicitation interviews?
>
> **$H_{0\text{-}2B}$:** The need to develop a prototype has no effect on the qualities of follow-up requirements elicitation interviews.

When performing the Mann-Whitney U test on follow-up interviews using time-only normalised values (see Table 10), it was determined that all but one practice showed no significant difference between median scores between the control and prototype groups with regard to follow-up interviews. The practice *use of hypotheticals and examples* was found to be the only practice that demonstrated a statistically significant difference between the median of the control group and the prototype group, with U = 13, z = -0.315, p = 0.043 (using the asymptotic p-value). We can therefore reject the null hypothesis regarding the HE tag. However, when calculating the effect size for this, it is discovered that $\eta^2 = 0.008$, d = 0.185. Therefore, there is a small or insignificant effect size resulting from the scores of the HE tag, making this difference difficult to observe.

Table 10. Results of the Mann-Whitney U test on all follow-up interviews normalised by time only.

	AI	TB	TB (neg)	DR	AM	VC	VC (summary)	**HE**	SA (pos)	SA (neg)
Mann-Whitney U	31	20	32	20	27	24	18.5	**13**	29	20
Wilcoxon W	67	56	68	56	63	60	54.5	**49**	65	56
Z	-.105	-1.26	0	-1.279	-.525	-.84	-1.514	**-2.025**	-.315	-1.849
Asymp. Sig. (2-tailed)	.916	.208	1	.201	.6	.401	.13	**.043**	.753	.064

Exact Sig. [2*(1-tailed Sig.)]	.959[b]	.234[b]	1[b]	.234[b]	.645[b]	.442[b]	.161[b]	**.050[b]**	.798[b]	.234[b]

Performing the Mann-Whitney U test on data that has been normalised by time and tag total, we obtain the results as demonstrated in Table 11. It was found that there were no significant differences between the control group and the prototype group for the follow-up interviews, with the closest tag being *explore to-be system (negative)*, where U = 20, z = -1.849, and p = 0.064. We can therefore not reject the null hypothesis for any tags using this method.

Table 11. Results of the Mann-Whitney U test on all follow-up interviews normalised by time and total tags.

	AI	TB	DR	AM	VC	VC (summary)	HE	SA (pos)
Mann-Whitney U	25	29	28	25	28	17	28	26
Wilcoxon W	61	65	64	61	64	53	64	62
Z	-.735	-.315	-.439	-.735	-.42	-1.617	-.42	-.63
Asymp. Sig. (2-tailed)	.462	.753	.661	.462	.674	.106	.674	.529
Exact Sig. [2*(1-tailed Sig.)]	.505[b]	.798[b]	.721[b]	.505[b]	.721[b]	.13[b]	.721[b]	.574[b]

When trying to interpret these results, the HE tag was found to be used significantly more often in the control group than in the prototype group. However, the effect size was small, meaning it would be difficult to observe this difference in real life. As with the above, it is expected that the lack of a reference point (i.e. the prototype) means there is a need for more abstractions and examples in certain scenarios. For example, when determining system functions relating to navigation, appearance, etc. it is less likely that those using prototypes would need to use abstractions as opposed to simply demonstrating the area of interest.

In summary, initial interviews compared between the control group and prototype group showed no significant differences when controlling for time only, and showed only a significant difference for HE tag use when controlling for time and total tags. Follow-up interviews showed a significant difference for HE tag use when controlling for time only, and no significant differences when controlling for time and total tags.

6 Conclusion

This thesis has explored the relationship between requirements elicitation interviews and the use of prototypes. As a reminder, this research was necessitated by the lack of scientific evidence regarding the use of a prototype alongside requirements elicitation interviews. Specifically, a need to reduce uncertainty surrounding the RE process for (new) analysts was apparent, especially as this pertains to the refinement of the process. After careful examination of the existing literature, frameworks, and guidelines, and applying the gathered information in the experiment, insights were discovered as to the nature of the relationship between prototyping and requirements elicitation interviews.

To this end, the following research questions were posed, with their null hypotheses where relevant:

> **RQ_1:** What frameworks exist for analysing the qualities of requirements elicitation interviews?
>
> **RQ_2:** To what extent does the type of interview affect the qualities of requirements elicitation interviews, regardless of prototype creation?
>
> > **$H_{0\text{-}2}$:** Interview type has no effect on the qualities of requirements elicitation interviews.
>
> **RQ_{2A}:** To what extent does the need to develop a prototype affect the qualities of *initial* requirements elicitation interviews?
>
> > **$H_{0\text{-}2A}$:** The need to develop a prototype has no effect on the qualities of initial requirements elicitation interviews.
>
> **RQ_{2B}:** To what extent does the need to develop a prototype affect the qualities of *follow-up* requirements elicitation interviews?
>
> > **$H_{0\text{-}2B}$:** The need to develop a prototype has no effect on the qualities of follow-up requirements elicitation interviews.

When seeking the answers to these research questions, several observations came to light. Firstly, it was determined that the type of interview – either initial or follow-up – has an impact on the qualities of requirements elicitation interviews, thereby allowing the rejection of $H_{0\text{-}2}$. Specifically, initial interviews tend to have more usage of the *Explore As-Is System* practice, whereas follow-

up interviews tend to demonstrate increased usage of the *Explore To-Be System, Develop Rapport with Stakeholders,* and *Introduce Stakeholder Agency* practices, meaning the rejection of the null hypotheses $H_{0\text{-}2A}$ and $H_{0\text{-}2B}$ respectively. In other words, analysts are more likely to ask questions relating to the as-is system during initial interviews, and questions relating to the to-be system during follow-up interviews. It is highly suspected that this is due to the nature of these respective interviews, since one would expect analysts to explore the scope of the existing system prior to speculating about the system to be created.

Moreover, it is speculated that the increased drive towards stakeholder agency is due to the nature of follow-up interviews as a way to come up with solutions or brainstorm ideas. The focus on rapport with stakeholders is also speculated to be due to increasing familiarity over the interview timeline, as stakeholders and analysts became more comfortable with each other.

Secondly, when we consider the major aim of the research, the central question addressed in this research was as follows:

> **MRQ:** To what extent does the need to develop a prototype affect the qualities of requirements elicitation interviews when compared with a given framework?

After conducting the research, it was determined that the major impact of prototyping lies in the use of hypotheticals and examples. Specifically, it was found that – when controlling for either time only or for time and total tags – the control group engaged in the *Use Hypotheticals and Examples* practice more often than the prototype group. All other qualities were not changed enough to be considered statistically significant.

What this means in practice is unclear at this juncture. However, it is possible that this reduction in the use of hypotheticals may be contributing to an existing concern regarding prototype usage, namely bias towards existing solutions (Carrizo & Quintanilla, 2018; Mannio & Nikula, 2001). There are many unanswered questions like this regarding the nature of the results, leading to the need for additional research in this regard.

6.1 Future Work

In light of the contributions made throughout this thesis, there are several possibilities when it comes to future iterations of this work. Overall, when conducting the research, many decisions

were made that could have changed the outcome of the research if a different path had been followed. For example, it was decided that only one researcher would perform tagging, which had the potential to cause bias despite the inclusion of a secondary review. In future, it may be a good idea to perform tagging with several different researchers in order to minimise potential bias as much as possible.

Regarding additional research which could be performed, there are several options for expanding on the results or exploring new facets of the experiment. Firstly, future research could expand upon the coding guidelines and the recommended practices in order to create an even more robust framework. This could ensure that many of the missed opportunities of the present research could be addressed.

Secondly, it would greatly improve the quality of research to conduct the same experiment with young professionals instead of university students. This would not only improve the real-world applicability of the work, but also give insights as to what kind of quality is currently to be expected from new analysts. It could also provide insight as to the role of analyst expertise in this particular scenario. Another potential track for future work could be the conducting of a large-scale longitudinal field study regarding the successful and failed practices of RE interviews, especially regarding on-paper vs. in-practice interviews.

Thirdly, it may be worthwhile to modify the research method to be more empirical, so as to allow different statistical analysis methods to be used and explored. It might also be interesting to explore different elicitation methods, like focus groups or observation, to see how these are impacted by or impact the other elicitation techniques.

6.2 Final Thoughts

In conclusion, this thesis has explored the effects of prototyping on the qualities of requirements elicitation interviews. The findings herein underscore a need for more robust guidelines and practice frameworks regarding how to conduct interviews effectively, as well as a need for more research regarding the interplay between various elicitation techniques – in this case, prototyping and interviews.

Future work should focus on expanding on the research by further validating the results, exploring

new facets of the presented relationship, considering other elicitation techniques, and so on. By performing additional research in this way, it should theoretically be possible to achieve a level of understanding of requirements elicitation techniques that positively impacts both experts and novices alike.

References

Anwar, F., & Razali, R. (2014). Requirements Elicitation Techniques Selection Survey. In H. Fujita, A. Selamat, & H. Haron (Eds.), *New Trends in Software Methodologies, Tools and Techniques. Proceedings of the Thirteenth SoMeT* (pp. 280–294). IOS Press BV.

Aranda, A. M., Dieste, O., Panach, J. I., & Juristo, N. (2022). Effect of Requirements Analyst Experience on Elicitation Effectiveness: A Family of Quasi-Experiments. In *IEEE Transactions on Software Engineering*. https://doi.org/10.1109/TSE.2022.3210076

Bano, M., Zowghi, D., Ferrari, A., Spoletini, P., & Donati, B. (2018). Learning from mistakes: An empirical study of elicitation interviews performed by novices. In *Proceedings - 2018 IEEE 26th International Requirements Engineering Conference, RE 2018*, 182–193.

Bano, M., Zowghi, D., Ferrari, A., Spoletini, P., & Donati, B. (2019). Teaching requirements elicitation interviews: an empirical study of learning from mistakes. In *Requirements Engineering*, *24*(3), 259–289.

Baxter, K., Courage, C., & Caine, K. (2015). *Understanding Your Users: A Practical Guide to User Research Methods.* Morgan Kaufmann.

Bormane, L., Gržibovska, J., Bērziša, S., & Grabis, J. (2016). Impact of requirements elicitation processes on success of information system development projects. In *Information Technology and Management Science, 19*(1), 57-64.

Burnay, C., Bouraga, S., Gillain, J., & Jureta, I. J. (2020). What lies behind requirements? A quality assessment of statement grounds in requirements elicitation. In *Software Quality Journal*, *28*(4), 1615–1643.

Carrizo, D., & Quintanilla, I. (2018). Prototyping use as a software requirements elicitation technique: A case study. *Advances in Intelligent Systems and Computing*, *746*, 341–350.

Dar, H., Lali, M. I., Ashraf, H., Ramzan, M., Amjad, T., & Shahzad, B. (2018). A systematic study on software requirements elicitation techniques and its challenges in mobile application development. *IEEE Access*, *6*, 63859–63867.

Davis, A., Dieste, O., Hickey, A., Juristo, N., & Moreno, A. M. (2006). Effectiveness of requirements elicitation techniques: Empirical results derived from a systematic review. In *14th IEEE International Requirements Engineering Conference*, 179-188.

do Prado Leite, J. C. S., & Gilvaz, A. P. P. (1996). Requirements elicitation driven by interviews: The use of viewpoints. *Proceedings of the 8th International Workshop on Software Specification and Design, IWSSD 1996*, 85–94.

Ezell, J. D., Lending, D., Dillon, T. W., May, J., Hurney, C. A., & Fulcher, K. H. (2019). Developing Measurable Cross-Departmental Learning Objectives for Requirements Elicitation in an Information Systems Curriculum. *Journal of Information Systems*

Education, *30*(1), 27–41.

Falessi, D., Juristo, N., Wohlin, C., Turhan, B., Münch, J., Jedlitschka, A., & Oivo, M. (2018). Empirical software engineering experts on the use of students and professionals in experiments. *Empirical Software Engineering*, *23*(1), 452–489.

Ferrari, A., Spoletini, P., Bano, M., & Zowghi, D. (2019). Learning Requirements Elicitation Interviews with Role-playing, Self-Assessment and Peer-review. *Proceedings of the IEEE International Conference on Requirements Engineering*, *2019-September*, 28–39.

Ferrari, A., Spoletini, P., & Debnath, S. (2022). How do requirements evolve during elicitation? An empirical study combining interviews and app store analysis. *Requirements Engineering*. https://doi.org/10.1007/s00766-022-00383-7

Gordon, V. S., & Bieman, J. M. (1995). Rapid Prototyping: Lessons Learned. *IEEE Software*, *12*(1), 85–95.

Hadar, I., Soffer, P., & Kenzi, K. (2014). The role of domain knowledge in requirements elicitation via interviews: An exploratory study. *Requirements Engineering*, *19*(2), 143–159.

Hartwell, C. J., Johnson, C. D., & Posthuma, R. A. (2019). Are we asking the right questions? Predictive validity comparison of four structured interview question types. *Journal of Business Research*, *100*, 122–129.

Häußer, D., Lauenroth, K., van Loenhoud, H., Schwarz, A., & Steiger, P. (2022). *Handbook IREB Certified Professional for Requirements Engineering Advanced Level Requirements Elicitation*. International Requirements Engineering Board.

Hickey, A. M., & Davis, A. M. (2003). Elicitation Technique Selection: How Do Experts Do It? In *Proceedings of the 11th IEEE International Requirements Engineering Conference*, 169-178.

Hickey, A. M., & Dean, D. (1998). Prototyping for Requirements Elicitation and Validation: A Participative Prototype Evaluation Methodology. In *AMCIS 1998 Proceedings*, 798–800.

Hirschheim, R., & Klein, H. K. (2012). A Glorious and Not-So-Short History of the Information Systems Field. *Journal of the Association for Information Abstract Research Article*, *13*(4), 188–235.

Khan, K. S., Kunz, R., Kleijnen, J., & Antes, G. (2003). Five steps to conducting a systematic review. *Journal of the Royal Society of Medicine, 96*(3), 118-21.

Khan, S., Dulloo, A. B., & Verma, M. (2014). Systematic Review of Requirement Elicitation Techniques. In *International Journal of Information and Computation Technology*, 4(2), 133-138.

Klehe, U., & Latham, G. (2006). What Would You Do—Really or Ideally? Constructs Underlying the Behavior Description Interview and the Situational Interview in Predicting Typical Versus Maximum Performance. *Human Performance*, *19*(4), 357–382.

Kumari, S. N. & Pillai, A. S. (2014). A systematic literature review on the software requirements elicitation issues. *Journal of Network and Innovative Computing, 2*, 283-308.

Kvale, S. (1994). InterViews: An introduction to qualitative research interviewing. In *InterViews: An introduction to qualitative research interviewing.* Sage Publications, Inc.

Laplante, P. A., & Kassab, M. H. (2022). *Requirements Engineering for Software and Systems*. CRC Press.

Lending, D., Ezell, J. D., Dillon, T. W., & May, J. (2022). A Rubric to Evaluate and Enhance Requirements Elicitation Interviewing Skills. *Journal of Information Systems Education, 33*(4), 371–387.

Lund, A & Lund, M (2018a). Wilcoxon Signed-Rank Test using SPSS Statistics. *Laerd Statistics*. URL: https://statistics.laerd.com/spss-tutorials/wilcoxon-signed-rank-test-using-spss-statistics.php

Lund, A & Lund, M (2018b). Mann-Whitney U Test using SPSS Statistics. *Laerd Statistics*. URL: https://statistics.laerd.com/spss-tutorials/mann-whitney-u-test-using-spss-statistics.php

Mannio, M. & Nikula, U. (2001). Requirements Elicitation Using a Combination of Prototypes and Scenarios. In *WER*, 283-296.

Martin, J. R. (2017). *Actuality Interviewing and Listening: How to Conduct Successful Interviews for Nonfiction Storytelling, Actuality Documentaries and Other Disciplines Using Mindful Listening.* Real Deal Press, JR Martin Media Incorporated.

Miller, D. (2021). The Best Practice of Teach Computer Science Students to Use Paper Prototyping. In *International Journal of Technology, Innovation and Management*, *1*(2), 42–63.

Mohanani, R., Ralph, P., & Shreeve, B. (2014). Requirements Fixation. *Proceedings of the International Conference on Software Engineering 2014*, *1*, 895–906.

Mohedas, I., Daly, S. R., Loweth, R. P., Huynh, L., Cravens, G. L., & Sienko, K. H. (2022). The use of recommended interviewing practices by novice engineering designers to elicit information during requirements development. *Design Science*, *8*.

Oktay, J. S. (2012). *Grounded Theory*. Oxford University Press.

Utrecht University (n.d.). *Osiris Student Mobile*. Retrieved July 7, 2023, from https://osiris-student.uu.nl/#/onderwijscatalogus/extern/cursus?cursuscode=INFOMRE&collegejaar=huidig

Pacheco, C., Garcia, I., & Reyes, M. (2018). Requirements elicitation Techniques: A systematic literature review based on the maturity of the techniques. *IET Software*, *12*(4), 365–378.

Paetsch, F., Eberlein, A., & Maurer, F. (2003). Requirements Engineering and Agile Software Development. WET ICE 2003. Proceedings of the Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 308–313.

Palanque, P., Ladry, J.-F., Navarre, D., & Barboni, E. (2009). High-Fidelity Prototyping of Interactive Systems Can Be Formal Too. *Lecture Notes in Computer Science*, *5610*, 667–676.

Portigal, S. (2013). *Interviewing Users: How to Uncover Compelling Insights*. Rosenfeld Media.

Rafiq, U., Bajwa, S. S., Wang, X., & Lunesu, I. (2017). Requirements elicitation techniques applied in software startups. *Proceedings - 43rd Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2017*, 141–144.

Rudd, J., Stern, K., & Isensee, S. (1996). Low vs. High-Fidelity Prototyping Debate. *Interactions*, *3*(1), 76–85.

Rueda, S., Panach, J. I., & Distante, D. (2020). Requirements elicitation methods based on interviews in comparison: A family of experiments. *Information and Software Technology*, *126*.

Sadeghi, Nastaran. (2019). Do Oral Placement Tests in English Language Institutes Conform to Interview Principles? -A Survey Conducted in Iran. *Language Teaching Research Quarterly*, *10*, 48–61.

Schneider, K. (2007). Generating Fast Feedback in Requirements Elicitation. In *LNCS*, *4542*, 160-174.

Skjott Linneberg, M., & Korsgaard, S. (2019). Coding qualitative data: a synthesis guiding the novice. In *Qualitative Research Journal*, *19*(3), 259–270.

Spijkman, T., Dalpiaz, F., & Brinkkemper, S. (2021). Requirements Elicitation via Fit-Gap Analysis: A View through the Grounded Theory Lens. *Advanced Information Systems Engineering: 33rd International Conference, CAiSE 2021*, 363–380.

Spijkman, T., Dalpiaz, F., & Brinkkemper, S. (2022). Back to the Roots: Linking User Stories to Requirements Elicitation Conversations. *2022 IEEE 30th International Requirements Engineering Conference (RE)*.

Spoletini, P., Ferrari, A., Bano, M., Zowghi, D., & Gnesi, S. (2018). Interview review: An empirical study on detecting ambiguities in requirements elicitation interviews. In: Kamsties, E., Horkoff, J., Dalpiaz, F. (eds) Requirements Engineering: Foundation for Software Quality. REFSQ 2018. Lecture Notes in Computer Science. *LNCS*, *10753*, 101–118.

Starman, A. B. (2013). The case study as a type of qualitative research. In *Journal of Contemporary Educational Studies*, *1*, 28–43.

Sulianta, F., Rosita, A., Zulpratita, U. S., Heryono, H., Laksana, E. A., & Yuliani, S. (2019). Different Kinds of Modern Technique to Develop Various Information Systems. *Universal Journal of Electrical and Electronic Engineering*, *6*(2), 20–31.

Taylor, P. J., & Small, B. (2002). Asking applicants what they would do versus what they did do:

A meta-analytic comparison of situational and past behaviour employment interview questions. *Journal of Occupational and Organizational Psychology*, *75*(3), 277–294.

Tiwari, S., & Rathore, S. S. (2017). A Methodology for the Selection of Requirement Elicitation Techniques. *ArXiv, abs/1709.08481*.

Tuunanen, T. (2003). A New Perspective on Requirements Elicitation Methods. In *Journal of Information Technology Theory and Application, 5*(3), 45-72.

Vijayan, J., & Raju, G. (2011). A New approach to Requirements Elicitation Using Paper Prototype. In *International Journal of Advanced Science and Technology*, *28*, 9-16.

Wagner, S., Fernández, D. M., Felderer, M., Vetrò, A., Kalinowski, M., Wieringa, R., Pfahl, D., Conte, T., Christiansson, M. T., Greer, D., Lassenius, C., Männistö, T., Nayebi, M., Oivo, M., Penzenstadler, B., Prikladnicki, R., Ruhe, G., Schekelmann, A., Sen, S., … Winkler, D. (2019). Status quo in requirements engineering: A theory and a global family of surveys. *ACM Transactions on Software Engineering and Methodology*, *28*(2), 1-48.

Walker, D., & Myrick, F. (2006). Grounded theory: An exploration of process and procedure. In *Qualitative Health Research, 16*(4), 547–559.

White, M. D. & Marsh, E. E. (2006). Content Analysis: A Flexible Methodology. *Library Trends, 55*(1), 22-45.

Wohlin, C. (2021). Case Study Research in Software Engineering—It is a Case, and it is a Study, but is it a Case Study? *Information and Software Technology*, 133. https://doi.org/10.1016/j.infsof.2021.106514

Wood-Harper, A. T. Antill, L, & Avison, D. E. (1985). *Information Systems Definition: The Multiview Approach*. Blackwell Scientific Publications, Ltd.

Zaremba, O., & Liaskos, S. (2021). Towards a typology of questions for requirements elicitation interviews. In *Proceedings of the IEEE International Conference on Requirements Engineering*, 384–389.

Zave, P. (1997). Classification of research efforts in requirements engineering. *ACM Computing Surveys (CSUR), 29*(4), 315-321.

Zhao, Y., & Zhao, J. (2018). *A Systematic Stakeholder Selection Model in Requirements Elicitation for Software Projects: A Systematic Mapping Study* [Masters Thesis], Blekinge Institute of Technology.

Zowghi, D., & Coulin, C. (2005). Requirements Elicitation: A Survey of Techniques, Approaches, and Tools. In *Engineering and Managing Software Requirements*, 19–46.

Appendix

A. Full Document of Coding Guidelines

HOW TO APPLY THE CODING SCHEMA

The following guidelines have been adapted from the principles proposed by Mohedas et al.[1] They are to be used to tag requirements elicitation interviews, in order to identify the use of particular interviewing practices. This, in turn, is to be used to determine the difference in used practices between the interviews and the significance of these differences.

FIRST, SOME GENERAL GUIDELINES

You are strongly encouraged to use the **audio recordings** alongside the transcript – the text does not matter as much as the **number of tags.** Try to always tag things stated by the analyst, if possible.

When in **doubt** about a particular tag, **do not use it**. This ensures that false positives are avoided.

Unless otherwise specified, tagging should be applied to the **interviewer / analyst's** speech, and not the interviewee / stakeholder.

Unless otherwise specified, all tagging should start from the first **spoken** word (i.e. the tag should **not** start with a speaker name or timestamp). For the sake of consistency, tag the **entire** speech turn, even if the occurrence itself is only one sentence of a longer speech turn. If a speech turn is interrupted by background noise, small affirmations, etc., treat it as if there was no interruption.

In the case of a file where a speaker turn is split up into several lines of dialogue (i.e. split up by timestamp instead of speaker), tag from the first spoken word of that speaker to the last spoken word of that speaker (in cases where the transcription software has recorded the last word of the speaker as the first word of the next speaker, only tag until the last word of the speaker turn itself). For example (yellow indicating the tag):

> "Speaker 1: How would you like the interface to
>
> Speaker 2: look. I guess I want it similar to the Albert Heijn app"

In the case of multiple occurrences of the same tag in one speech turn, split the tags where the next occurrence begins. For example: "So how should the system store new users? || And what permissions should they get?" (where || indicates the split).

If multiple **Questions** are asked which all lead to one answer (i.e. the interviewer asks a question, gives some context etc., then asks a different question still requiring the same / a similar answer), tag the first word of the first question instance until the last word of the last question instance. This includes situations where the interviewer has to repeat a question to a stakeholder who does not understand something.

Unless otherwise specified, the tagging units **Question, Interview Quality, Summary** and **Statement** are used to indicate good (positive) qualities of the interview.

A **single** Question / Statement / etc. can have **multiple tags.**

[1] Mohedas, I., Daly, S. R., Loweth, R. P., Huynh, L., Cravens, G. L., & Sienko, K. H. (2022). The use of recommended interviewing practices by novice engineering designers to elicit information during requirements development. Design Science, 8. https://doi.org/10.1017/dsj.2022.4

EXPLORE AS-IS SYSTEM (AI)

DESCRIPTION

- Relates to the system as-is.
- Requires stakeholders to describe existing business processes, or parts thereof.
- Unlikely to require a yes/no answer.
- Note: This tag should not overlap with *Introduce Stakeholder Agency* (unless they are used separately in a single speech turn).

CODING

- Question
  Tag questions relating to process decisions of the business with this tag.

EXPLORE TO-BE SYSTEM (TB)

DESCRIPTION

- Relates to the system to-be (or the refinement of the as-is system).
- Requires the stakeholder to make decisions about the system functions as they pertain to **functional requirements**. **Non-functional requirements** can also be included in this tag, insofar as they do not expect a stakeholder to be an expert (see *Introduce Stakeholder Agency*). For example, "Do you want the system to be able to run on any device, or just on the computers in the building?" would be acceptable.
- Requires the stakeholder to analyse or assess situations or examples in order to determine the best functionality / requirements.
- Unlikely to require a yes/no answer.

CODING

- Question
  If the question has context that spans multiple speaker turns, tag starting from the first word of the first speaker turn to the last word of the last speaker turn. Otherwise, tag questions pertaining to the to-be system as normal.
- Statement (Negative)
  Interviewer present solutions without any feedback from, or invitation to consider things more deeply to, interviewee.

DEVELOP RAPPORT WITH STAKEHOLDERS (DR)

DESCRIPTION

- Analyst asks [personal] questions that do not relate to the business or its processes (**small talk**).
- Analyst makes comments to ease tension, foster a friendly environment, and so on (e.g. compliments).
- Analyst asks questions relating to the stakeholders' background. These can include past and current job experience, **BUT** should not concern their activities within the business and its processes (i.e. something like their job title is fine, but precisely what they do regarding the as-is system should not be considered rapport-building).

CODING

➢ Question, Statement
Tag each question (i.e. with separate answers) or statement individually.

AVOID MISINTERPRETATIONS (AM)

DESCRIPTION

- Clarifying questions where the analyst is not already confident about a specific answer or solution.
- Analyst is specifically **asking** for an answer, rather than implying or stating that they already have a particular answer in mind.
- Usually takes the form of asking for elaboration, specifying an option or answer between several (i.e. narrowing the scope), etc.
- There should be **grounds for misinterpretation** to occur (i.e. there is information upon which to make a decision) – otherwise it is simply information-gathering.
- **Always** consider whether an AI / TB tag is also relevant.
- The simple difference between *Avoid Misinterpretations* (AM) and *Verify Conclusions with Stakeholders* (VC, below) is that **VC** demonstrates that **a solution is already decided on** to some degree, whereas **AM** implies that **there is no specific solution yet**, or that a mentioned solution is purely for context.
- For example: "You mentioned that students need to input their SSN to get the benefits. Does that mean international students don't get the benefits?" would be considered AM, whereas "So international students are excluded from benefits, since they don't have a SSN," would be considered VC. The wording implies that there is a certain level of **confidence** in the conclusion.
- AM also differs from *Introduce Stakeholder Agency* (SA) in that **AM** presumes that the **stakeholder knows** what the correct option or answer is, whereas **SA** requires the stakeholder to **make a decision**.

CODING

➢ Question
Questions should be clarifying in nature. These will usually take the form of "What do you mean by that?", "Can you elaborate on that?", and so on.

VERIFY CONCLUSIONS WITH STAKEHOLDERS (VC)

DESCRIPTION

- Confirmatory questions regarding the conclusions drawn from the interview, specifically regarding requirements or specifications. Note that this should not include simple observations.
- Analyst may imply that they already have an answer in mind, or state the answer directly with the expectation of approval.
- "If I understand correctly", "Just to confirm", etc.
- The stakeholder does not necessarily need to provide confirmation, nor be provided with a moment to provide confirmation.
- **Always** consider whether an AI / TB tag is also relevant.

CODING

- Question, Statement
  The interviewer might present their verification in the form of a question. This may look similar to an *Avoid Misinterpretations* question, but the difference is specificity. Verification questions will usually include an interpretation or a specific feature, for example, "So customers see the login screen before they can access the logistics page, right?"
- Summary
  An interviewer may present the interviewee with their list of requirements elicited, or something similar. In this case, tag the entire speech turn or set of speech turns. This might also be considered the debriefing (time spent verifying conclusions in bulk).

PRESENT NON-SUPERFICIAL INTRODUCTION (PI)

DESCRIPTION

- Analyst explains the **purpose** of the interview at the **beginning** (e.g., their goals and the design project). This should include things like the structure of the interview, the designers' names, etc.
- Superficial, unstructured introductions are not considered part of this tag. For example, an introduction where analysts introduce themselves, and then simply state "We will be interviewing you today. Let's start off with …" does not provide sufficient structure regarding the interview timeline.

CODING

- Interview Quality
  If the interviewer gives a briefing (or introduction) for the interview at the beginning, tag the entirety of the introductory section, INCLUDING the first speaker name.

ALLOCATE STAKEHOLDER FEEDBACK TIME (FT)

DESCRIPTION

- Analyst leaves time at the **end** of the interview to allow the stakeholder to offer any **concluding thoughts** that were **not covered by questions.**
- This should be an explicit setting aside of time to explore topics, questions, etc. brought up by the stakeholders.
- This is **not** considered a use of SA.

CODING

➢ Interview Quality
If the interviewer gives time at the end of the interview for the interviewee to address anything not covered by the questions, then tag the initiating speech turn.

USE HYPOTHETICALS AND EXAMPLES (HE)

DESCRIPTION

- Analyst frames questions using hypothetical situations, examples and outliers.
- These mostly include situations which stray from the absolute standard process (i.e. the "perfect world").
- This can also include giving examples to better clarify the details of a process or answer. For example, if an analyst describes a scenario where a particular step in a process has been reached, and then asks for further information (note: the provision of an example or hypothetical is **required** for this tag!).
- If the **answer would change / be less specific** without the use of the example or hypothetical, then do **not** use this tag (for example: "So their information – for example date of birth or ID – gets stored in their patient file?").
- This tag will almost always be used alongside another tag (e.g. **Explore to-be system**).

CODING

➢ Question
These questions might follow on from an interviewee's description of a process or feature, for example: "How would you handle a customer ordering something that is out-of-stock?"

INTRODUCE STAKEHOLDER AGENCY (SA)

DESCRIPTION

- Questions or statements that give stakeholders a more **significant role** in the final system design.
- Questions or statements where analyst and stakeholder might **jointly** come up with a solution.
- Analyst may give stakeholders the opportunity to provide **active feedback** on the requirements elicited thus far.
- Analyst may cede control of the interview to the stakeholders (this excludes *Allocate Stakeholder Feedback Time*).
- **In short,** this requires stakeholder to **make decisions** and **give their opinions** rather than provide **existing information** about the business and its goals, plans, or objectives.
- Giving stakeholders **too much control** should be considered a **negative** use of this principle. For example, asking stakeholders **highly technical questions**, such as questions pertaining to **non-functional requirements** (e.g. "How many milliseconds of delay would you be expecting between clicking *submit* and returning to the home screen?") gives stakeholders responsibility for something that should be the responsibility of developers, and expects them to have technical knowledge not usually possessed by standard stakeholders.
- In general, exchanges where it does not make sense to include a stakeholder in a decision, or where the analyst is expecting the stakeholder to be an expert (in something other than their own business dealings) should be considered **negative**.
- **Note:** This tag should not overlap with *Explore As-Is System* (unless they are used separately in a single speech turn).

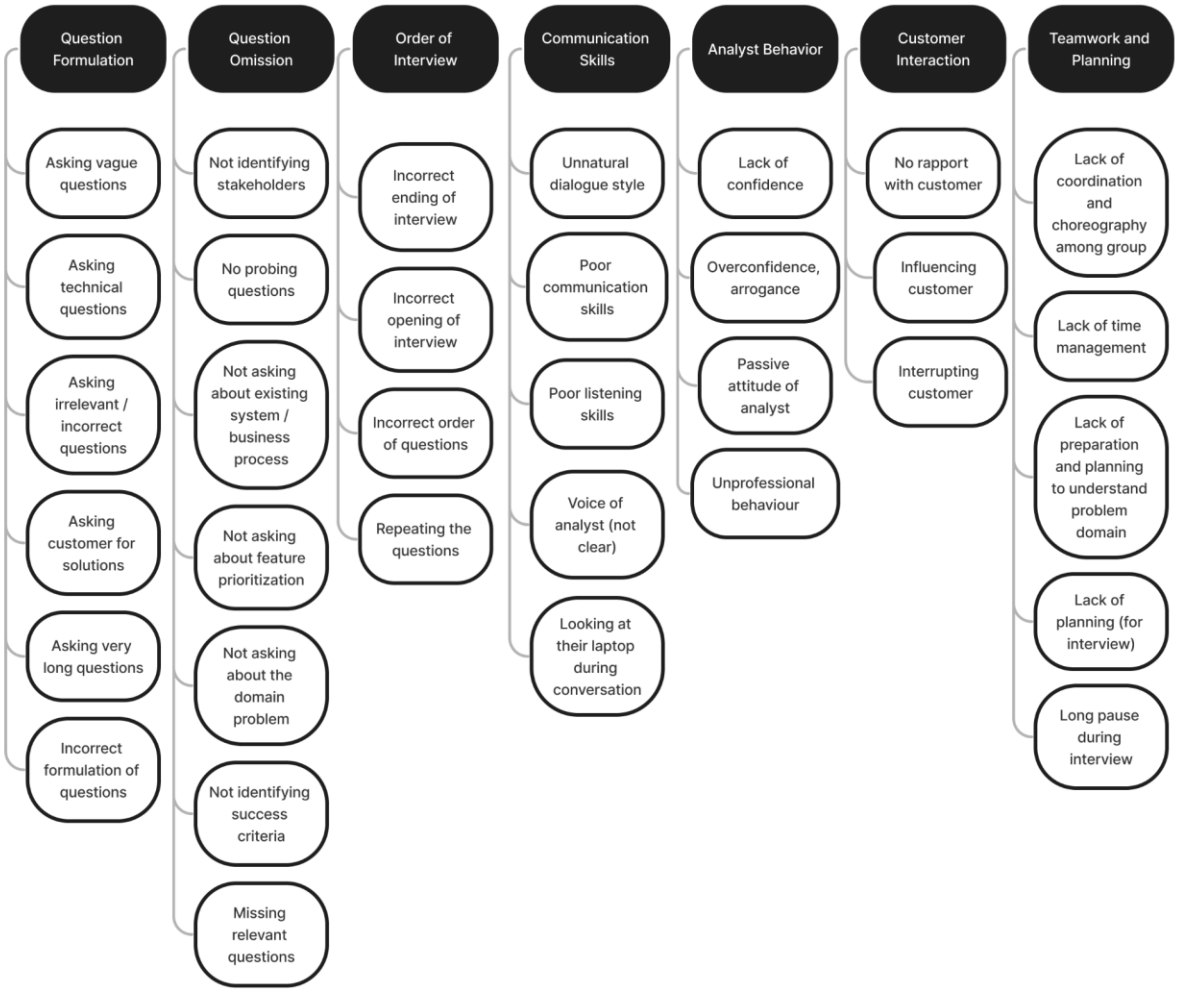
CODING

- Question, Statement (Positive)
  Tag **questions** that give the interviewee an opportunity to provide active feedback on the requirements or artifacts, or which give the interviewee an opportunity to control the interview.
  Statements which give interviewees a stake in, or a responsibility for, the process or product will use this tag. Statements which share the interview control with the interviewee are also valid here.
- Question, Statement (Negative)
  If an analyst is asking questions or making statements which demand the stakeholders come up with a solution, or require them to have technical knowledge beyond what should be reasonably expected of a standard stakeholder, use this negative tag.

B. Zaremba and Liaskos's (2021) Question Categories and Sub-Categories

Time	Content	Form	Style	Probing Style	Objective
Past	Users / Stakeholders	Open	Catch-all	Elaboration / Informational Probe	Forced Choice
Present	Needs / Evaluations	Bipolar	Comparison / Contrast	Reason Seeking Probe	Leading
Future	Processes	Multiple Choice	Declarative	Clearinghouse Probing	Direct
	Objects / Data		Introducing	Consistency Probe	Indirect
	Required Functions		Specifying	Interpreting	Negative Balance
	Technology		Structuring	Check-reflect	
	Pragmatics		Ready Reference	Echo Probing	
			Instructional Ready Reference	Question Reformulation Probe	
			Directional	Restatement Probe	
			Tour	Leading Probe	
			Targeted – Minitour		

C. Classification of interview mistakes by Bano et al. (2019)



D. Original Explorations on Coding, Part 1: Coding Schema

MOHEDAS Practice	BANO Mistakes	Code Description	Simplified description of absence / negative / unused opportunity	Tagging unit(s)
Encourage deep thinking	asking technical questions, asking vague questions, asking customer for solutions, asking very long questions, influencing customer	- Designer asks question that requires stakeholder to (i) **Analyse a situation:** presents a situation (or multiple) and asks stakeholder to analyse, assess, make a determination, and so forth (ii) **Integrate ideas:** asks stakeholder to synthesise multiple ideas to establish broader understanding (iii) **Use logical reasoning:** asks stakeholder to reason out why something is the way it is, or how something occurs, and so forth	Asking questions with complexity that confuses stakeholders; forcing stakeholders to consider concepts they may not be familiar or comfortable with	Question, Statement
Develop a rapport with the interviewee	No rapport with customer, incorrect opening of interview, poor listening skills, overconfidence / arrogance, interrupting customer	- Designer begins the interview with non-design related questions or **small talk.** - Designer uses personal questions (when appropriate) to develop relationship with stakeholder: (i) **Conversation** at beginning or end; **non design related** conversation during interview (ii) Questions about the **stakeholders' background** to better understand them	Lack of icebreakers or introductions; disrespecting customers	Question, Interview Quality
Avoid misinterpretations	asking vague questions, no probing questions, poor communication skills	Designer asks a **clarifying question** to ensure they fully understand a stakeholder's response	Lack of addressing / exploration of ambiguities	Question, Opportunity
Be flexible and opportunistic	missing relevant questions, lack of confidence, passive attitude of analyst, influencing customer	- Designer probes into a **topic area brought up by a stakeholder** (tangential to the designer's original question) - Designer **adjusts their interview questions/approach** after learning about the stakeholder	Inability to deviate from planned questions; dismissal of further discussion on emerging topics	Question, Statement, Opportunity
Verify the conclusions drawn from interviews	no probing questions, incorrect ending of interview	- Designer presents the stakeholder with **their interpretation** of the stakeholder's response for confirmation - Designer presents **conclusions drawn from other interviews** to stakeholders to determine if prior conclusions hold true - Designer **verifies the requirements/specifications** generated through interviews	No debriefing or confirmation of understandings	Question, Statement, Interview Quality
Designer begins and interviewee concludes	incorrect opening of interview, incorrect ending of interview, no rapport with customer, lack of time management, lack of planning (for interview)	- Designer explains to the stakeholder the **purpose** of the interview (e.g., their goals and the design project) - Designer leaves time at the end of interview to allow the stakeholder to offer any **concluding thoughts** that were **not covered by questions**	No outline or rapport at beginning of interview; no debriefing or extra time for loose ends	Statement, Interview Quality
Use projective questioning techniques	asking technical questions, asking customer for solutions, asking very long questions, incorrect formulation of questions, no probing questions, poor communication skills	Designer frames questions using: (i) **Hypothetical:** how would you, what would be your preference if, if you had to choose, and so forth (ii) **Story telling:** if you had to purchase a toy today, how would you go about it	No analogies or reframing techniques used to further analyst understanding, especially with the absence of follow-up questions	Question, Interview Quality
Use a co-creative strategy	not identifying stakeholders, not asking about feature prioritiation, not identifying success criteria, overconfidence / arrogance, influencing customer, interrupting customer	Designer asks questions or makes comments that would increase the stakeholder's **sense of ownership of interview/product requirements**; questions give ownership of the process to the stakeholder	Stakeholders are not treated as knowledgeable; stakeholders are told rather than asked	Question, Statement
Introduce domain knowledge	asking technical questions, not asking about existing system / business process, not asking about the domain problem, lack of preparation and planning to understand problem domain	Designer uses **domain knowledge** from prior interview or observation or literature or any other information source to frame question	Interviewer does not explore relevant domain knowledge or specialise questions for the problem domain	Question, Opportunity
Have the interviewee teach you	asking irrelevant / incorrect questions, asking customer for solutions, incorrect formulation of questions, passive attitude of analyst, lack of confidence, unprofessional behaviour, influencing customer, overconfidence / arrogance	Designer uses **ignorance** to get the stakeholder to teach them about an idea/procedure/concept/etc.	Interviewer comes across as confused or unconfident, or refuses to demonstrate a lack of knowledge	Question, Statement
Explore contradictions	no probing questions, missing relevant questions, poor listening skills	Designer asks about **discrepancies** within a stakeholder's own responses or **differences** between different stakeholders' responses	Interviewer fails to point out and explore a (relevant) contradiction	Question, Opportunity
Break down expert tasks	not identifying stakeholders, not asking about existing system / business process, missing relevant questions	Designer probes stakeholder's responses to break-up **complicated cognitive processes** into more manageable steps	Interviewer does not explore stakeholder interactions with the business process or its tasks	Question, Interview Quality

E. Original Explorations on Coding, Part 2: Practice Descriptions from Mohedas

PRACTICE	DESCRIPTION
Encourage deep thinking	Interviewers should encourage stakeholders to think deeply, defined as analytical thinking, integration of ideas,and use of logical reasoning. This approach encourages stakeholders to move beyond superficial responses and provide more in-depth knowledge on a subject.
Develop a rapport with the interviewee	Interviewers should strive to develop a good rapport with stakeholders. This approach facilitates a more comfortable discussion and more open and honest responses.
Avoid misinterpretations	Misinterpreting a stakeholder's responses can lead to erroneous information being collected. Interviewers should document stakeholder's exact wording or the interviewer should present their interpretation back to the stakeholder for verification.
Be flexible and opportunistic	Interviewers should not be rigidly attached to a predefined list of questions or topics. They must be able to identify relevant topics arising from stakeholders'responses and dig deeper into this newly discovered information.
Verify the conclusions drawn from interviews	If conclusions are drawn during analysis of interview results (or during interviews), interviewers should attempt to verify that these conclusions align with stakeholders'perceptions.
Designer begins and interviewee concludes	At the start of the interview, interviewers should define the purpose and goals of the interview with the stakeholders. At the end, interviewers should allow time for stakeholders to discuss topics of interest that were not asked about.
Use projective questioning techniques	Interviewers should use stories, metaphors, drawings, analogies, role-playing, third party projections, and so forth, to enhance the information elicited.
Use a co-creative strategy	Interviewers should encourage stakeholders to take ownership of the goals of the interview; giving stakeholders a stake in the outcome can result in more useful informationelicitation.
Introduce domain knowledge	Interviewers should seek to introduce domain knowledge (either from other information sources or other interviews). This approach can help elicit information about specific topics for which stakeholders may be experts.
Have the interviewee teach you	Interviewers can profess ignorance to encourage stakeholders to explain a specific topic or break complicated subjects into component parts.
Explore contradictions	Interviewers should seek to understand contradictions (within a stakeholder's own responses) or disagreements between different stakeholders.
Break down expert tasks	Experts may fail to mention all of the cognitive or physical processes required to perform a specific task or reach a goal. Interviewers should follow-up and probe experts to capture all information

F. Original Explorations on Coding, Part 3: Mistake Descriptions from Bano

MISTAKE	DESCRIPTION
Asking vague questions	the type of questions that may yield multiple interpretations, or cases where no reasonable meaning can be inferred from the questions asked
Asking technical questions	it cannot be assumed that the business owner/project sponsor has detailed technical knowledge. The analyst often uses technical language
Asking irrelevant / incorrect questions	asking questions that are not relevant for the development of the system, or are inappropriate for the profile of the customer
Asking customer for solutions	straightforward
Asking very long questions	customer might ask to repeat the question several times
Incorrect formulation of questions	same as above, question is not understandable
Not identifying stakeholders	did not ask about relevant stakeholders
No probing questions	follow-up questions, no exploration (maximum information already reached)
Not asking about existing system / business process	did not ask about the problem, details but no bigger picture
Not asking about feature prioritization	no attempt to prioritie features required by customers
Not asking about the domain problem	questions are generic and domain-independent, analysts unable to elaborate or provide examples in domain context
Not identifying success criteria	analysts do not seek qualifiers from stakeholders
Missing relevant questions	self-explanatory
Incorrect ending of interview	no summary / debriefing, abrupt ending
Incorrect opening of interview	direct questions immediately, no rapport building or general description of project
Incorrect order of questions	Asking questions about concepts or topics that have not been explored yet
Repeating the questions	self-explanatory, does not include clarification
Unnatural dialogue style	linguistic difficulties of analyst, bad question phrasing, stakeholder is confused by dialogue
Poor communication skills	never going off-script (scholastic), many ambiguities
Poor listening skills	analyst does not display attentiveness to given answers, same mistakes repeated despite correction, repeating questions, interrupting
Voice of analyst (not clear)	self-explanatory, also irrelevant?
Looking at their laptop during conversation	self-explanatory, also irrelevant?
Lack of confidence	losing control of the interview, inability to be flexible, etc.
Overconfidence, arrogance	interrogative questioning, lack of a plan, dismissing potential of mistakes
Passive attitude of analyst	low voice, slow attitude, nervousness
Unprofessional behaviour	acting embarrassed, apologizing often, excessive laughter, rushing
No rapport with customer	no introductions, icebreakers, etc. Comes across as an exercise or assignment
Influencing customer	spending too much time promoting analyst ideas over listening to customer criticisms
Interrupting customer	not allowing customer to finish statements or discourse
Lack of coordination and choreography among group	no division of tasks like notetaking, analysts interrupt each other
Lack of time management	cutting the interview short, not getting any planned information, ending interview early
Lack of preparation and planning to understand problem domain	no domain knowledge demonstrated in questions
Lack of planning (for interview)	no coordination over who will speak, or the order of questions
Long pause during interview	self-explanatory

G. Original Explorations on Coding, Part 4: Tag Definitions

TAG	DESCRIPTION
Statement	A non-questioning phrase or set of phrases.
Question	A question which requires an answer from another party.
Opportunity (?)	An instance where an individual could have asked a question or made a statement, but did not.
Interview Quality	An attribute of the interview in its entirety. For example, whether or not the interview includes an introduction and a debriefing.

H. Original Document of Coding Guidelines

HOW TO APPLY THE CODING SCHEMA

FIRST, SOME GENERAL GUIDELINES

Questions should always be tagged in their entirety, including the "?".

Tag the moment of **Opportunity** by tagging the last word of the previous statement (interviewee, usually) and the first word of the following statement (interviewer), for example: "customers. *Speaker 2:* Alright,".

Unless otherwise specified, a **Question** is positive (good quality), and **Opportunities** and **Interview Qualities** are negative (bad quality). A **Statement** can be positive or negative, depending on the given description. No individual tag is both positive and negative!

ENCOURAGE DEEP THINKING

- Question
  If the question has context, tag starting from the first word of the first sentence to the last word of the last sentence. Do not tag multiple questions unless they are asked in the same speech turn (i.e. only one response for both questions).

DEVELOP A RAPPORT WITH THE INTERVIEWEE

- Question
  Tag each question (i.e. with separate answers) individually.
- Interview Quality
  If there are no *Question* tags, tag the first word of the interview to show that the interview does not contain any rapport-building.

AVOID MISINTERPRETATIONS

- Question
  Questions should be confirmatory or clarifying in nature. These will usually take the form of "What do you mean by that?" or "Can you elaborate on that?" – generally non-specific questions relating to clarification.
- Opportunity
  An opportunity of this kind will generally be after an interviewee has said something vague or domain-specific, where an analyst does not appear to comprehend or continue the conversation line.

BE FLEXIBLE AND OPPORTUNISTIC

- Question
  Question should follow on from a topic the interviewee arrived at.
- Statement
  Tag statements which are made to shut down conversation: "Let's move on," "Let's get back on track", etc. Tag only the statement which shuts down the conversation, not the entire speech turn. (Negative)
- Opportunity
  An opportunity of this kind will generally be after an interviewee has started discussing a new topic, or a topic not initially asked about by the interviewer. If there is already a *Statement* tag, then this tag is not necessary.

VERIFY THE CONCLUSIONS DRAWN FROM INTERVIEWS

- Question
  The interviewer might present their verification in the form of a question. This may look similar to an *Avoid Misinterpretations* question, but the difference is specificity. Verification questions will usually include an interpretation or a specific feature, for example, "So customers see the login screen before they can access the logistics page, right?"
- Statement
  An interviewer may present the interviewee with their list of requirements elicited, or something similar. In this case, tag the entire speech turn. This might also be considered the debriefing.
- Interview Quality
  If there is no debriefing of *this* kind at the end of the interview, tag the last word of the interview. Debriefing here refers to time spent verifying conclusions.

DESIGNER BEGINS AND INTERVIEWEE CONCLUDES

- Statement
  If the interviewer gives a briefing (or introduction) for the interview at the beginning, tag the entirety of the introductory section.
  If the interviewer gives time at the end of the interview for the interviewee to address anything not covered by the questions, then tag the initiating sentence(s) or speech turn.
- Interview Quality
  If there is no briefing at the beginning of the interview, tag the first word of the interview.
  If there is no debriefing of *this* kind at the end of the interview, tag the last word of the interview. Debriefing here refers to allowing time for interviewee concluding thoughts.

USE PROJECTIVE QUESTIONING TECHNIQUES

- Question
  These questions might follow on from an interviewee's description of a process or feature, for example: "How would you handle a customer ordering something that is out-of-stock?"
- Interview Quality
  If the interviewer never uses hypotheticals or storytelling in their questions, then tag the last word of the interview.

USE A CO-CREATIVE STRATEGY

- Question
  Tag questions that give the interviewee an opportunity to provide active feedback on the requirements or artifacts, or which give the interviewee an opportunity to control the interview.
- Statement
  Statements which give interviewees a stake in, or a responsibility for, the process or product will use this tag. Statements which share the interview control with the interviewee are also valid here. Be sure to tag the entire sentence or group of sentences, starting with the first word of the first sentence and ending with the last word of the last one. (Positive)

INTRODUCE DOMAIN KNOWLEDGE

- Question
  Domain knowledge refers to the field in general (for example, paper printing). It is not specific to the business itself. Interviewers might demonstrate this knowledge by asking questions that use jargon or refer to concepts within the domain.
- Opportunity

If an interviewee brings up something domain-specific and the interviewer does not explore it (where it is clear that exploration would be beneficial), or the interviewer asks a basic question where a domain-specific question could have replaced it, then this tag is used.

HAVE THE INTERVIEWEE TEACH YOU

- Question
- Statement

EXPLORE CONTRADICTIONS

- Question
  This should be fairly straightforward to spot.
- Opportunity
  Spotting contradictions that span entire interviews is a bit time-consuming. Therefore, only use this to tag contradictions happening within one to three speech turns of each other. Tag at the end of the speech turn where the second contradicting statement occurred (and the beginning of the next, as usual).

BREAK DOWN EXPERT TASKS

- Question
  While “domain knowledge” is slightly vaguer, “cognitive (or physical) processes” refer to internal business choices pertaining to the specific way a business interacts with and executes processes in their domain (for example, printing using a press and all the processes that go with it vs. printing using an industrial printer). Tag questions relating to process decisions of the individual business with this tag.
- Interview Quality
  If the interviewer never explores any of the business processes in accordance with the above, then tag the last word of the interview.