Protocol for analysis qualitative

1. Introduction

- **Context:** The prompt engineering approach was recently applied in a unit of analysis - project analyzed during the case study - and, from this, the generated proto-personas were collected, the time of execution of the approach and responses from a form based on the model TAM. After this collection, interviews were conducted with each participant of the analysis unit that carried out the approach, following a certain script. The recordings collected in these interviews are the **inputs** for the qualitative analysis. The analysis, in the adopted model, aims to extract **codes and themes** from the interview transcripts (as if they were, respectively, the essence of what was said in the transcript and the trends between participants).
- **What is this document and what is it for?** This document defines the protocol for carrying out qualitative analysis, in the context of research which involves applying the prompt engineering approach to the generation of proto-personas during Lean Inception. This protocol is based on the procedures defined by *Cruzes and Dyba.* This document is essential to establish the protocol to be used to perform the qualitative analysis from the responses to the interviews carried out during the collection stage.

2. Protocol for qualitative analysis

- The following describes the protocol for conducting the qualitative analysis, adapting **the** procedures of *Cruzes and Dyba and* **the article Investigating the Developer eXperience of LGBTQIAPN+ People in Agile Teams.** The steps of the protocol consist of:
  - **(1) Free Coding:** Perform line-by-line coding of the findings of primary studies, identifying the main relevant codes.
    This step will be done manually.
  - **(2) Organizing Codes:** Grouping these 'free codes' into related areas to form 'descriptive themes'. This step requires constant comparison and reciprocal translation between the data. This step will be done using LLM.

- In the protocol below, an additional step was inserted to extract the transcript, since most of the new data is in recordings of the meet.

2.1. Extract the transcript

- In the [Anonymous] drive folder, under **Interviews,** all recordings of the interviews are stored, organized according to the researcher who conducted them. Before extracting them, it is necessary to transcribe the recording.

- To make the transcription, you need to:

1. Download the recording of the interview in question.
2. Convert to MP3 using online tools.
3. To transcribe audio to text, follow these steps:
   a. Install the necessary dependencies (and make sure you have Python and pip installed on your computer):

```
pip install openai-whisper pip
install docx
```

   b. Access a directory on your machine, create a file called "transcriptor.py” and put the following content:

```
import whisper
from docx import Document

def transcribe_audio(audio_path):
    model = whisper.load_model("base")

    # Transcribe the audio
    result = model.transcribe(audio_path)

    return result["text"]

if __name__ == "__main__":
    audio_file = "p1.mp3"
    transcription = transcribe_audio(audio_file)

    doc = Document()
    doc.add_heading("Audio Transcript", level=1)
    doc.add_paragraph(transcription)
    doc.save("p1.docx")

    print("Transcript saved in transcript.docx")
```

c. Place the mp3 generated in step 2 in the same directory as the "transcriptor.py” script and change the value of the **audio_file variable,** within the conditional expression, with the name of the recording.

d. After that, type the following command in the terminal

```
python3 transcriptor.py
```

e. With this, a docs file should have been generated in your current directory.

f. With the docs in hand, place them in the respective participant's folder in drive, along with the recording and notes.

g. When you access the generated documents, you will see that it is a single continuous text with some spelling errors. The next step is to separate the statements of the researcher and the participant.

h. With this, the transcription ready for extraction of codes is obtained and themes.

NOTE: There are other tools that make this approach easier (most of them are paid). If you have access to any of them, you can use them without any problems. The focus is only on obtaining the final transcription.

2.2. Extract the codes

- This step is done manually.
- A **code** is a summary of a relevant aspect (in relation to the research objective) contained in a transcription unit. It is a label or tag applied to data segments that serves to categorize and summarize relevant information. The transcription unit is a part of the transcript that has a meaning and minimally provides context.

- Given an interview transcript, the following steps should be taken to extract codes:

    - For each transcription block, an analysis must be carried out by the researcher to extract the **essence** of what was said in that segment (code).

        - Notes taken during the interview should be used to help define the code.

        - If necessary, the transcription excerpt can be analyzed by observing it in the original recording, to better support code extraction.

        - Some guidelines:

            - A code must be atomic, that is, it cannot encompass other codes.

            - A transcription block may contain multiple codes. If this is the case, the transcription must be divided into smaller units,

called transcription units, until the codes are completely separated.

- Once the codes have been extracted, they must be evaluated by the other team members.

- Resolving doubts and standardizing codes can be done during the protocol.

- The codes and transcription units (quotations) must be placed on the participant's spreadsheet.

2.3. Extract the themes

- To extract the themes, LLM will be used (standardizing the GPT Chat), following the recommendations of:
    - J. Roberts, M. Baker, and J. Andrew, “Artificial intelligence and qualitative research: The promise and perils of large language model (llm)'assistance',” Critical Perspectives on Accounting, vol. 99, p. 102722, 2024.

    - L. Yan, V. Echeverria, GM Fernandez-Nieto, Y. Jin, Z. Swiecki, L. Zhao, D. Gaÿsevi ´c, and R. Martinez-Maldonado, “Human-ai collaboration in thematic analysis using chatgpt: A user study and design recommendations,” in Extended Abstracts of the CHI Conference on Human Factors in Computing Systems, pp. 1–7, 2024.

- The following prompt, based on the ”structured task description” and ”Input-ProcessOutput (IPO)” patterns, should be applied in GPT Chat:

> I am performing a thematic synthesis process based on the responses to a survey. I will provide the manually extracted codes, then I will ask you to generate possible themes from the codes.
>
> Associated codes - Sequence of manually extracted codes;

- GPT Chat will generate several **themes** and the final theme will be considered based on the following protocol:

- **Who performs the analysis:** researchers, in groups (to reduce the bias of each participant), manually.
- **Criteria for topic selection:**
  1. **Relevance to the RQs:** the topic must help answer the proposed RQ of the research and be relevant to it.
  2. **Uniqueness of the theme:** According to the principles of Braun and Clarke's a theme must be coherent (self-contained) and essentially different from others.
  3. **Theme compliance with codes:** the theme must cover all codes of that semantics.
- **If there is still doubt between one or more themes,** these can be modified or combined, including unique elements from each alternative.

- The following will not be considered:
  - **Verify if the proposed themes are truly representative of the nuances and context present in the original data .** A theme might seem logical based on the codes but might lose important contextual information from the original responses.
  - **Look for supporting evidence in the original data for each proposed theme.** The theme that is most strongly and consistently supported by the raw data is likely the most appropriate.
    - Because, the way the analysis was done, the codes represent the units, and therefore, when the theme represents the codes, it represents the units.
  - **Employ an Iterative Refinement Process:** Thematic synthesis is often an iterative process .... Don't expect ChatGPT's first suggestions to be perfect.
  - **Provide feedback to ChatGPT:** If none of the initial themes are satisfactory, you can refine your prompt or provide more context to guide the model in generating more relevant themes .... You could tell ChatGPT why certain themes are not suitable or suggest connections between the codes that it might have missed.
    - The choice is made manually.

- Once the themes have been extracted, they must be evaluated by at least one other member of the team.
- The topics must be placed on the participant's spreadsheet.