

–B.Sc. Thesis–

Design, Implementation, and Evaluation of an LLM Assisted User Interface for a Questionnaire on Hereditary Risk Factors in Breast and Ovarian Cancer

Freie Universität Berlin

Institute of Computer Science

Human-Centered Computing (HCC) Research Group

Tim Jakob Lamich

Student Number: 5566040

lamit03@zedat.fu-berlin.de

Date of Submission: September 24, 2025

Supervisor:

Prof. Dr. Claudia Müller-Birn, Freie Universität Berlin, Germany*

Examiner:

Prof. Dr. Claudia Müller-Birn, Freie Universität Berlin, Germany[†]

Prof. Dr. Markus Feufel, Technische Universität Berlin, Germany[‡]

*Department of Mathematics and Computer Science, Human-Centered Computing Research Group

[†]Department of Mathematics and Computer Science, Human-Centered Computing Research Group

[‡]Department of Psychology and Ergonomics, Division of Ergonomics

Abstract

This bachelor's thesis examines the integration of large language models into medical questionnaires using the example of a checklist from the Deutsche Krebsgesellschaft for recording hereditary risk factors for breast and ovarian cancer. The aim is to develop and evaluate an LLM-assisted user interface that supports users in answering complex questions, thereby improving usability and user experience.

As part of a human-centered design process, a proof of concept was designed, developed, and tested in a qualitative study with six participants. The user interface includes artificial intelligence support from two LLM agents that communicate with each other and generate appropriate outputs for users. The results of the study show improved transparency, guidance, and efficiency through LLM assistance, while reducing the overall complexity of the questionnaire.

The work confirms the potential of large language models to support medical survey instruments, but also provides an outlook on necessary improvements and future research areas for the further development of such applications.

(German version)

Diese Bachelorarbeit untersucht die Integration von Large Language Modellen in medizinische Fragebögen am Beispiel einer Checkliste der Deutschen Krebsgesellschaft zur Erfassung von erblichen Risikofaktoren für Brust- und Eierstockkrebs. Ziel ist die Entwicklung und Evaluation eines LLM-unterstützten User Interfaces, das Nutzer der Checkliste bei der Beantwortung komplexer Fragen unterstützt und auf diese Weise die Benutzerfreundlichkeit und Benutzererfahrung verbessert.

Im Rahmen des Human-Centered Design Prozesses, zugeschnitten auf das Ziel dieser Bachelorarbeit, wurde ein Proof of Concept entworfen, entwickelt und in einer qualitativen Studie mit sechs Teilnehmern getestet. Das User Interface umfasst eine künstliche Intelligenz Unterstützung durch zwei LLM-Agenten, die miteinander kommunizieren und auf diese Weise den Nutzenden angebrachte Rückmeldungen generieren. Die Ergebnisse der Studie zeigen eine verbesserte Transparenz, Führung und Verständlichkeit durch die LLM-Assistenz, während es die generelle Komplexität des Fragebogens reduziert.

Die Arbeit bestätigt das Potenzial von Large Language Modellen zur Unterstützung medizinischer Erhebungsinstrumente, gibt jedoch auch einen Ausblick auf notwendige Verbesserungen und zukünftige Forschungsfelder für die Weiterentwicklung solcher Anwendungen.

Contents

1. Introduction	1
1.1. Motivation	1
1.2. Research Goal and Question	1
2. Theoretical Background	3
2.1. Context and Problem Description	3
2.2. Related Work	4
2.2.1. Robust AI-led Interviews	4
2.2.2. Qualitative Interviews with Multi-agent AI	8
2.3. Methodology	12
3. Implementation	14
3.1. System architecture	14
3.2. Prompt engineering	15
3.3. User Interface	19
4. Evaluation	25
4.1. Study design and setup	25
4.2. Data analysis	27
4.3. Results and Insights	28
4.4. Discussion	31
5. Conclusion	33
5.1. Limitations	33
5.2. Future Work	35
Bibliography	36
A. Appendix	39
A.1. Questionnaire	39
A.2. Robust AI-led Interviews	40
A.3. Qualitative Interviews with Multi-agent AI	42
A.4. Heuristics for conversational agents	45
A.5. Study scenario and Task	48
A.6. Activity diagram	49
A.7. Data Analysis Visualizations	50
A.8. Gitlab and AI usage documentation	51

List of Figures

2.1. Interface of running interview	6
2.2. Multi agent architecture and workflow	8
2.3. Chat interface for interviewees	9
2.4. AI interviewer behavior	10
2.5. HCD process	12
3.1. LLM agents behavior	15
3.2. Introduction to the questionnaire	20
3.3. Questionnaire layout	21
3.4. AI assistant dialog	22
3.5. Additional pop-ups	22
3.6. Extenders of the MVP	23
3.7. Submission page	23
4.1. Coding rules framework	27
4.2. Code matrix	28
4.3. User experience questionnaire	30
A.1. Application context questionnaire	39
A.2. LLM agents activity diagram	49
A.3. Categories definition	50
A.4. Code trends	50

List of Abbreviations

LLM large language model

AI artificial intelligence

MVP minimum viable product

DKG Deutsche Krebsgesellschaft

API application programming interface

HCD human-centered design

1. Introduction

The developments in the field of artificial intelligence (AI) have underscored the capabilities of the technology. Chatbots and intelligent personal assistants are widely used [8], taking a look at OpenAI¹ with their chatbot called "Chat-GPT". The underlying technology behind chat-based applications form large language models (LLMs). For this reason, LLMs are receiving more attention and their use is being considered in many applications [5].

Especially in the field of medicine, AI-based applications and therefore LLMs have gained notable value integrated in patient care and diagnostics [24]. A rising healthcare innovation named "Ada"², for example, implements an AI assisted symptom analyzer for patients.

1.1. Motivation

Questionnaires and surveys play an important role in medical contexts [23]. But the phrasing and terminology can become very specific. Patients who have been asked to complete the questionnaire may not be familiar with the terminology and phrasings used. Migrating the questionnaire into an application with a supportive AI based on an LLM could improve the patient experience. Additional guidance could lead the patients through the questions, and the supportive AI could help clarify the specific terminology. However, it is important to note that integrating AI into an application also creates new challenges. These must therefore also be taken into account.

This thesis explores what integrating a large language model in a medical questionnaire to support the patients could look like and provides insights about functionality, improvements, and hurdles.

1.2. Research Goal and Question

A questionnaire provides information about the patient. Their effectiveness largely depends on whether patients can correctly understand and answer the questions.

This leads to the primary objective of this thesis. The goal is to develop a concept for the use of large language models supporting medical questionnaires. The integrated LLM

¹<https://openai.com/>

²<https://ada.com/>

should address the previously mentioned problems in the form of providing additional information about domain-specific terms and a chatbot to answer questions regarding the questionnaire.

LLMs are **powerful artificial intelligence models** trained on large datasets. The models are capable of problem-solving, comprehension, and text production. So their use case aligns with the goal of the thesis.

It needs to be mentioned that when using LLMs in human-computer interaction, there are some aspects you need to be aware of. Most typical are hallucinations, where the LLM produces content that diverges from the user input or invents new information [25]. In contrast, prompt engineering is a method to configure the model and define guidelines for how the model should generate the outputs with the use of a hidden initial system prompt [12].

To demonstrate the feasibility of this goal, the thesis aims to develop a proof of concept together with a minimum viable product (MVP). The MVP will illustrate a possible fundamental implementation. The subsequent user study will explore challenges and benefits for further development, focusing on usability and user experience.

The research question guiding this thesis is: *How does the user interface of an LLM assisted questionnaire support user experience and usability, as evaluated through the lens of eleven heuristics for conversational agents [15]?*

2. Theoretical Background

The following chapter describes the context in which the LLM will be integrated in detail and introduces relevant implementations of AI-based interviews. The related work serves as a foundation for the development of the minimum viable product and will be adapted to the specific use case of a supportive AI assistant rather than an interviewer.

2.1. Context and Problem Description

To develop an MVP, it is crucial to have an application context. In this case, the thesis is based on the checklist for possible hereditary risk factors for breast and/or ovarian cancer by the Deutsche Krebsgesellschaft (DKG).

The DKG does research in the detection of risk factors for cancer. The detection comes with surveys and checklists that the patients have to fill out. The complete checklist can be found in the appendix A.1.

The questionnaire acts as a basis, which will be extended with a large language model. The questionnaire consists of many questions and domain-specific terms that are challenging for most users. Terms like "Mammakarzinom" or "Ovarialkarzinom" are not very common in everyday language. Furthermore, the structure is complex, and it takes time to understand how it works correctly. The checklist is divided into different categories. The questions depend on close collaboration with family breast and ovarian cancer centers and can also differ based on the user's age and gender. Patients answer the questions with a numerical value. This value is then multiplied by a weighting factor and added up. This produces a risk score, which is treated as the result of the questionnaire. Until now, the questionnaire has primarily been used by physicians due to its technical phrasing [2].

Patients fulfilling the questionnaire on their own would encounter individual challenges or situations where they are uncertain about the correct course of action and need someone to turn to for advice. With the advancements in large language models, integrating such a model could help address these issues. A supportive AI could assist users by reducing confusion and providing answers to both general and personal questions. The minimum viable product will include an LLM to realize the supportive AI and a simple user interface enabling users to answer the questions and interact with the assistance.

2.2. Related Work

To get an idea of how the integration of a large language model could look, the following presents two projects in the field of AI-led interviews. Although the projects have different contexts, like economics, they provide a good overview of possible concepts and implementations. The focus in this literature review is on the structure and behavior of their LLM integration.

2.2.1. Robust AI-led Interviews

A versatile open-source platform for AI-led interviews, validated with 466 respondents, was developed by Geiecke and Jaravel [10]. Their article 'Conversations at Scale: Robust AI-led Interviews with a Simple Open-Source Platform' presents a new method for conducting qualitative interviews using large language models. The approach integrates sociological best practices for conducting interviews, uses a single LLM agent for quick responses, and is customizable for different topics. The robustness of their application is assessed through comparisons with human experts and quality-related metrics, while versatility is demonstrated through applying it in areas such as decision-making and political views.

The development of the LLM agent for AI supported qualitative interviews follows three main principles: adaptability, sociological best practices [1], and consideration of technical as well as ethical aspects.

Their prompt engineering follows a structure of three parts. The interviewer's role is defined in the "interview outline". Here is a short excerpt:

```
Interview Outline:
The interview consists of three [or another number to specify] successive parts for
which instructions are listed below.
Part I of the interview:
This part is the core of the interview. Ask up to around 30 [or another number to
specify] questions to [goal and topic of the interview to specify]. Begin the
interview with 'Hello! I'm glad to have the opportunity to speak with you about [to
specify]. Could you tell me [to specify]? Please don't hesitate to ask if anything is
unclear'.
Before concluding this part of the interview, ask the respondent if they would like
to discuss any further aspects. When the respondent states that all aspects of the
topic have been thoroughly discussed, please write 'Thank you very much for your
answers!'
Part II of the interview:
[...]
```

Code 2.1: Interview outline prompt by Geiecke and Jarave [10]

This part introduces the structure of the interview and what the LLM agent should ask. The "general instructions" guiding the behavior of the LLM agent fine-tuned with sociological best practices [1]:

```
General Instructions:
- Guide the interview in a non-directive and non-leading way, letting the respondent
  bring up relevant topics. [...] Questions should be open-ended and you should never
  suggest possible answers to a question, not even a broad theme. [...]
- Collect palpable evidence: [...]
- Display cognitive empathy: [...] find out the origins of these perspectives,
  evaluate their coherence, thoughtfulness, and consistency, and develop an ability to
  predict how the respondent might approach other related topics.
- Ask only one question per message.
- Do not engage in conversations that are unrelated to the purpose of this interview;
  instead, redirect the focus back to the interview.
```

Code 2.2: General instructions prompt by Geiecke and Jarave [10]

The last section, named "codes" enables the generation of a fixed output based on an event. When the LLM returns a code, the interface displays a predefined message (e.g. for interview end or unethical input). Without these codes, a multi-agent setup would be required to handle such control tasks. A multi-agent structure to address these tasks will be introduced in a later section.

```
Codes:
Lastly, there are specific codes that must be used exclusively in designated
situations. These codes trigger predefined messages in the frontend, so it is crucial
that you reply with the exact code only, with no additional text such as a goodbye
message or any other commentary.
Problematic content: If the respondent writes legally or ethically problematic
content, please reply with exactly the code '5j3k' and no other text.
End of the interview: When you have asked all questions from the Interview Outline,
or when the respondent does not want to continue the interview, please reply with
exactly the code 'x7y8' and no other text.
```

Code 2.3: Codes prompt by Geiecke and Jarave [10]

Each part can be adapted to a specific use case and represents the customization of their application. The complete prompts can be found in the appendix A.2. Joining the "interview outline", "general instructions", and "codes" forms the initial system prompt for the LLM.

For the usage of the LLM agent, they implemented a chat interface with the python framework Streamlit¹ (see Figure 2.1) and access the LLM through an application programming interface (API), which communicates over the internet with the May 2024 release of OpenAI's² GPT-4o model "GPT-4o-2024-05-13".

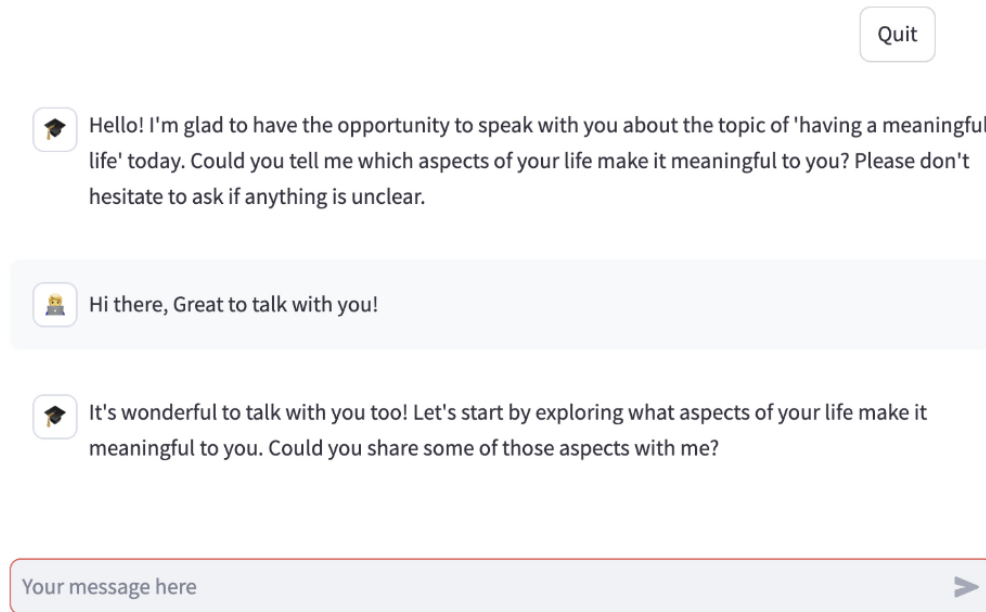


Figure 2.1.: Running interview in the chat interface [10]

The large language model is based on a text-generation model. Text generation models produce coherent and meaningful text. They analyze the user input to adapt the context of their output and ensure grammatical correctness as well as engaging content.

¹<https://streamlit.io/>

²<https://openai.com/>

The system is capable of interacting with the latest generation of LLMs. In addition to OpenAI’s API and LLM, they also recommend using Anthropic’s Claude³ model.

The model is provided with the previously described system prompt. In contrast, a user prompt is the visible input or question that the user provides to the model and is used to produce a fitting output.

The next paragraph explains a feature of an LLM to provide a better understanding of further developments. Throughout the interaction, the LLM receives the whole chat history. The chat history is important to preserve the historical context for the LLM. A large language model maintains a conversation-buffer-memory [13]. This memory is important as the LLM generates the output based on the input, and without the chat history, the agent would not know about previous information. The given chat history is then allocated to the conversation-buffer-memory of the LLM. To process the large input, the LLM features a large context window of around 200000 tokens. A token refers to a unit of text, as LLMs cannot understand text as a continuous string of characters [7]. The number of tokens defines how large the input of the LLM is. It allows the LLM to remember more text and reduce the possibility of hallucinations, as it remembers previous interactions.

Various methods were used to evaluate the developed platform for AI-supported qualitative interviews, including the performance of the AI interviews themselves.

Part of the evaluation was five respondent-based quality metrics, which the user answered in an open-ended text field. Specifically, they ask the respondents

- whether they would prefer to participate in an interview with an AI or a human in the future
- whether they would have preferred to answer questions in open text fields, rather than participating in an AI-led interview

Based on the respondent-based quality metrics, Geiecke and Jaravel [10] could conclude that a majority of users would prefer an AI and found the conversation natural and helpful.

The results indicate that the AI-powered interview platform can conduct effective large-scale qualitative interviews measured against expert-rated outputs comparable to human interviewers, high respondent satisfaction, and richer responses than standard open-text fields. These findings support both the goal of this thesis and its underlying proof of concept. Although the use case of the thesis differs from qualitative interviews, the results offer a valuable foundation for further development and adaptation. The use of a single LLM agent, combined with targeted customization, proves effective for prototyping and exploring possible interaction designs. Moreover, the prompt engineering offers useful insights into how to construct LLM agents for AI supported interaction.

³<https://www.anthropic.com/claude>

2.2.2. Qualitative Interviews with Multi-agent AI

Chopra and Haaland introduced another approach to conducting qualitative interviews by delegating the task of interviewing human subjects to an AI interviewer [6]. The work 'Conducting Qualitative Interviews with AI' investigates the use of AI in conducting qualitative interviews, emphasizing the evaluation of data quality and usability within a multi-agent structure. It analyzes how AI-driven interview systems, operating through multiple agents, enhance response accuracy, adapt questioning dynamically, and influence participant trust compared to human interviewers.

Chopra and Haaland deployed large language models in a multi-agent architecture (see Figure 2.2).

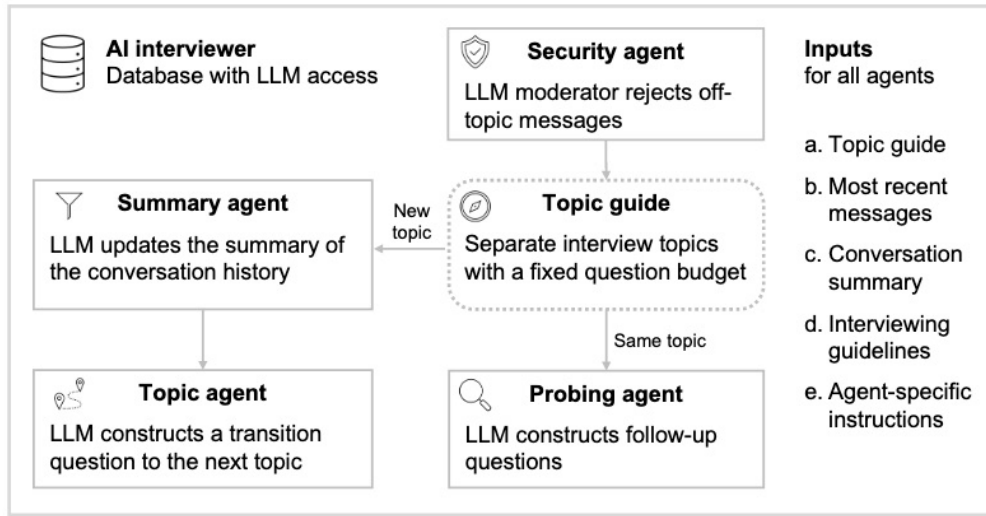


Figure 2.2.: Multi agent architecture and workflow to generate an interview question [6]

The key advantage of the multi-agent application is task specialization. As the architecture consists of multiple LLMs, each model focuses on a specific task, like topic transitions or summarizing the content. This architecture improves the data quality, as it can balance multiple objectives during the interaction. The multi-agent structure of the AI interviewer consists of four different so-called agents, each of which has specialized functions:

- **Security agent**: This agent serves as a defense against attempts by interview participants to change the AI interviewer's context (e.g. by asking off-topic questions such as "Tell me a joke").
- **Summary agent**: The task of this agent is to create an ongoing summary of the most important points and topics.

- **Probing agent:** This agent is responsible for asking probing questions. The questions explore promising topics in greater depth. They investigate the reasons, motivations, and beliefs of the interviewee and clarify unclear answers.
- **Topic agent:** The task of this agent is to shape the transition to the next topic in the interview guide.

Through a chat interface, the user can enter their answer to the interview questions (see Figure 2.3). The input is transferred over an API call to a remote server, which hosts the large language models.

The chat interface is displayed within a rounded rectangular container. It features three message bubbles: a light gray bubble for the first question, a darker gray bubble for the response, and another light gray bubble for a follow-up question. Below the chat area is a large text input field with a placeholder text. At the bottom left, there is a prompt to write at least two or three full sentences, and at the bottom right, there is a blue button labeled 'Submit response'.

I am interested in learning more about why you currently do not own any stocks or stock mutual funds. Can you help me understand the main factors or reasons why you are not participating in the stock market?

I do not own any stocks because I do not know how to get started or what stocks to pick.

Can you elaborate further on the difficulties you have faced in trying to get started with investing in stocks? What are some of the specific challenges or concerns you have encountered?

Type your message here...

Please write at least two or three full sentences.

Submit response

Figure 2.3.: Chat interface for respondents [6]

The system prompts define how the agents communicate. The complete prompts are included in the appendix A.3. The activity diagram (see Figure 2.4) shows the processes inside the remote server and the communication between the agents and the topic guide. The security agent is the first instance after the user submits a response, and it checks the user input before it is passed on. If the security agent confirms an appropriate input, the topic guide receives the input and decides on switching the topic or asking more probing questions. This decision is based on the researcher.

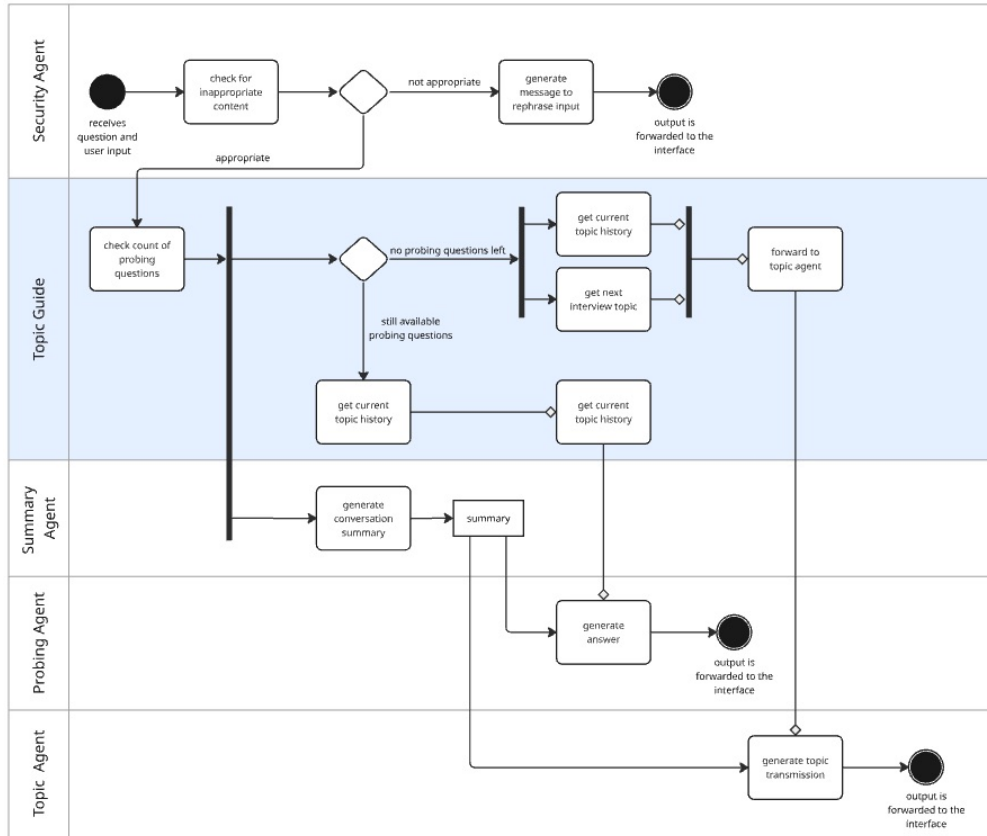


Figure 2.4.: UML activity diagram [21] of the AI interviewer functionality, the topic guide highlighted in blue (own illustration)

He configures the topic guide before the interview starts and determines the objectives of the interview topics and the number of probing questions for each. The topic guide is not built on a large language model.

Based on the decision from the topic guide, the input is either transmitted to the probing agent or the topic agent. Both agents receive, together with the input, a current summary of the conversation from the summary agent and the current topic. In the case of the topic agent producing the output, it also receives the next interview topic from the topic guide. The agents and the topic guide share information and maintain connections, enabling them to function as the AI interviewer.

The actual response from the AI interviewer is then generated by the probing or topic agent.

For the different large language models they used the versions "gpt-4-0613" and "gpt-4-0314" of the GPT-4 model family by OpenAI⁴. LLMs provide internal parameters that influence how text is generated. Parameters control creativity, repetition, or randomness. They diverge from the system prompt in that they define how to produce an output, whereas the system prompt defines what the model should generate. In the implementation of their different LLMs they described the parameter called "temperature" in detail. The temperature, going from 0 to 1, regulates the probability distribution for choosing the next word from the vocabulary of the model. Therefore, with a higher temperature, the model creates more creative answers.

For the probing and topic agent, they chose a temperature of 0.7, allowing a creative response while keeping up with the instructions in the system prompt, and for the summary and security agent a temperature of 0 was set. The other model parameters were set to the default values.

The study evaluates AI-assisted qualitative interviews using various methods, including participant feedback, behavioral data, hand-coding, and comparisons with open-ended questions. The results show a high level of acceptance and engagement of the interviewees, methodological accuracy of the AI interviewer, and close to zero cases of hallucination where the AI interviewer incorrectly refers to previous parts of the conversation. Overall, the evaluation confirms the quality, reliability, and cost-effectiveness of this method in qualitative research. In addition, another advantage of AI-conducted interviews is that they could help eliminate potential human interviewer biases, such as happier interviewers eliciting more positive responses from interviewees [22].

Chopra and Haaland introduced the task specialization for AI-based interviews and provided a detailed solution. Their developed model with four different agents generates qualitative results and facilitates a safe as well as very effective interaction.

⁴<https://openai.com/>

2.3. Methodology

The section presents methods and techniques used to develop the minimum viable product, including frameworks, the large language model, and other important concepts to understand the further procedure. The second part explains the structure of the evaluation with users.

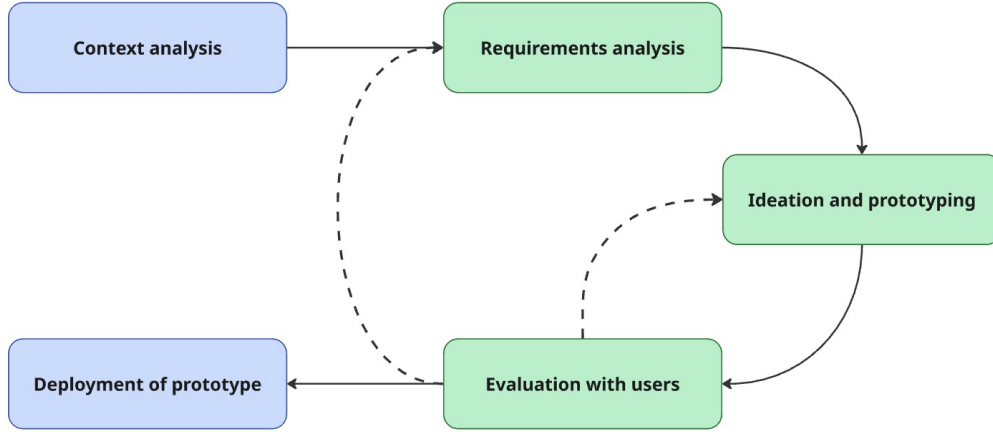


Figure 2.5.: Human-centered design process, adapted from [9]

This thesis follows the principles of the human-centered design (HCD) process [9] (see Figure 2.5). The HCD process is a problem-solving approach and focuses on the needs and context of the users using the product. The process includes qualitative research methods to achieve a good result. The iterative process consists of five steps, which can interrupt and complement each other, as indicated by the dotted lines. In the following, for each step, the adaptations will be explained and examined in detail.

The first step of understanding the context of the application helped to tailor the desired outcomes. This part was crucial in the development of a proof of concept. It was important to define the changes and how these changes would affect the current execution. Based on the research question, research was conducted on AI-assisted interactive applications.

The next part identified requirements for the development of the LLM assisted questionnaire. Instead of an Excel sheet, the user interacts with an interface, which includes a supportive AI. Powered by a large language model with two separate agents, inspired by the work of Chopra and Haaland in their implementation of a multi-agent AI [6], the interface guides the user through the questionnaire conversationally, provides explanations, and will act as an assistant if the user has any questions. Prompt engineering played a crucial role in this part to achieve a fitting behavior of the LLM agents. One LLM agent checks the user inputs to fit in the setting, and the second LLM agent provides

the outputs. The user interface also ensures that the given answers are well-structured and complete.

The two LLM agents are based on the open-source large language model called "Llama 3.1" [11], released by Meta⁵ and hosted on a server from the Freie Universität Berlin⁶. Access to the LLM was granted over an API. The user interface is created with the python⁷ framework Streamlit⁸, similar to the presented work by Geiecke and Jaravel [10].

Because of the limitation to a minimum viable product, the system is kept simple. The number of different tasks for the user was reduced by asking questions to the AI assistant and completing the questionnaire. When needed, the large language model can offer a clear explanation of technical terms appearing in the questionnaire. A significant part of the prototyping phase was an interview with the questionnaire's stakeholders. With the help of the exchange, it was able to make important adjustments to the prototype. As agreed during the interview [2], upon completing the questionnaire, the interface does not display the final score to prevent users from misinterpreting it. Instead, it indicates to the user that the data has been processed and provides the given answers in a table.

The evaluation with users phase conducted a usability study to assess user experience and usability, guided by the eleven heuristics for conversational agents by Langevin et al. [15]. For conversational agents in particular, specialized sets of heuristics have been developed to address their interaction paradigms, which involve more dialogue-based than point-and-click communication. Therefore, the heuristics by Langevin et al. were chosen. The study followed a qualitative research design in a laboratory setting. Using a think-aloud protocol, six participants interacted with the questionnaire while verbalizing their thoughts and impressions. Upon completion, participants responded to two open-ended questions focusing on their impression of the minimum viable product.

At the conclusion of the evaluation phase, it was outlined how the minimum viable product performed and stated which opportunities the MVP offers for future applications. Special attention was given to identifying challenges, particularly about user experience and usability, supported by insights gathered during the evaluation.

⁵<https://www.meta.com/about/>

⁶<https://www.mi.fu-berlin.de/w/IT/ItServicesProjectsEvop>

⁷<https://www.python.org/>

⁸<https://streamlit.io/>

3. Implementation

With the goal and research question in mind, the minimum viable product was implemented. This chapter describes each important aspect and explains the reasons for the choice of implementation. Additionally, it provides illustrations of the final MVP.

3.1. System architecture

Let's take a look at the architecture of the minimum viable product. The system is structured into the frontend, consisting of the user interface with the supportive AI assistance, and the backend, where the two LLM agents are deployed.

The frontend is the visible part of the MVP and provides interaction for the users to answer the questions of the questionnaire or have a conversation with the AI assistant. Due to the medical domain, the question format is copied from the original questionnaire by the DKG to ensure that every patient receives the same phrasing, enabling consistent responses [18]. To hand over the formulation process of the questions to the LLM agents would cause an inconsistency, as the generated questions can differ between patients.

The reason for this is that large language models text generation involves a small but significant amount of randomness. The process of finding words to generate the answer is based on the prediction of the next best-fitting word [19]. Therefore, LLMs have a non-deterministic behavior. That is, for the same user input, the large language model can produce different outputs on different executions. Considering this, the AI assistant has no control over the questions.

The backend is the hidden part of the application and handles, in general, the computing and logic of the application. The backend of the MVP manages the two LLM agents.

When the user opens the AI assistant, types in a question, and submits the question, the request is sent over an API call to the backend of the application. The communication over an API is important, as the backend is not hosted in the same location as the frontend. As the frontend is running on the same machine, executing the code, the backend is running on a server from the Freie Universität Berlin.

When the server is receiving the request, the first LLM agent processes the user input. Its task is to check the data for inappropriate content. In the case of this thesis, "inappropriate" means any content not related to a medical background. If the data is positive checked for inappropriate, the so-called "security agent" returns an "off-topic"

to the frontend, and the user is requested to reformulate their question. If it is not the case of 'off-topic' data, the security agent returns an "on-topic", which means the data is transmitted to the second LLM agent, so-called "support agent". The support agent then returns a generated reply to the user's input.

Finally, the output from the support agent needs to be communicated to the user interface, so that the data is available in the frontend of the application.

To ensure the described functionality, intended from the architecture of the MVP, high attention was paid to the correct implementation of the LLM agents and the layout and functionality of the user interface. The next section provides insights into the prompt engineering.

3.2. Prompt engineering

For effective prompt engineering, it is crucial to know how the large language model should behave. As explained in section 3.1, the behavior of the two LLM agents is clear. The created sequence diagram (see Figure 3.1) illustrates the behavior in a different way to lead the explanation of the prompt engineering of each LLM agent. An additional activity diagram visualizing the LLM agents' workflow can be found in the appendix section A.6.

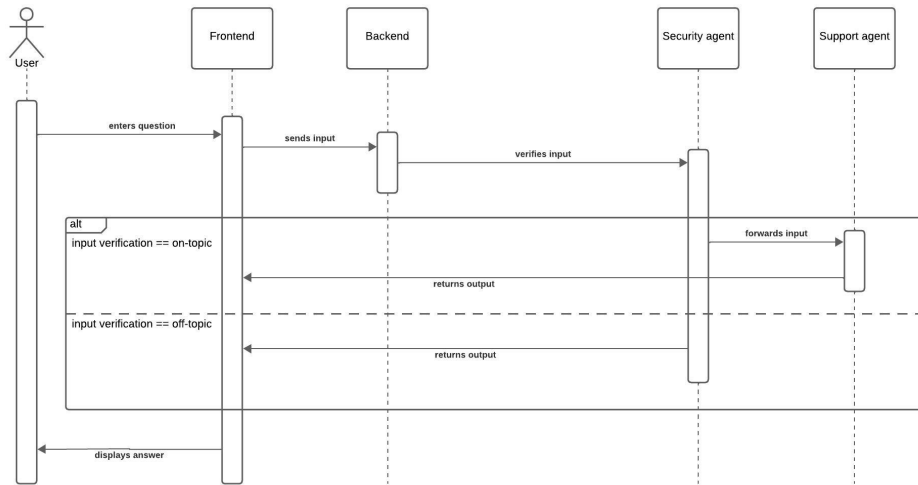


Figure 3.1.: UML Sequence Diagram [21] of the LLM agents workflow

In advance, it needs to be clear how to create system prompts for the LLM "Llama 3.1". This process can differ for each large language model. In the case of "Llama 3.1" a model file is used to customize the LLM. The model file includes the base model from which the new model is created, the internal parameters, and the initial system prompt.

In the documentation [11], they list even more configurations. With the built-in create command from "Llama 3.1", the new model is created based on the customizations in the modelfile.

Now, lets dive into the engineering of the first LLM agent, so-called the "security agent". This agent receives the user input and checks for inappropriate content. To ensure this functioning, the modelfile for the security agent was created like this:

```
FROM llama3.1

PARAMETER temperature 0
PARAMETER top_p 0.1
PARAMETER num_predict 3
PARAMETER top_k 10

SYSTEM You are a text classifier. Check whether the input is related to medicine or
medical assistance. Respond only with 'on-topic' or 'off-topic'. Do not provide
explanations or other responses. You do not answer questions. No sentences. The only
two statements you should use are 'on-topic' and 'off-topic'.
```

Code 3.1: Modelfile of the security agent

Each part of the modelfile will be explained in detail to provide transparency and replicability. The "FROM" statement sets the base model of the agent. The "PARAMETER" statements set the internal parameters of the LLM agent. The following table provides a short description for each parameter:

Parameter	Description
temperature	the temperature configures the models creativity. Value: 0 - 1
top_p	works together with top-k. It controls the diversity of the text generation. Value: 0 - 1
top_k	reduces the probability of generating nonsense. Value: 0 - 100
num_predict	sets the maximum number of tokens for generating text. Value: 0 - infinite

Table 3.1.: Extract from modelfile parameters with descriptions provided by "Llama3.1" documentation [11]

For the security agent, the temperature is set to zero. Forcing the agent to use as little creativity as possible. Top_p and top_k are set to a lower value, supporting a rather focused, conservative answer. Num_predict is set to a value of three tokens for a short output. All mentioned parameters were set to a rather lower value, as the LLM needed to be forced not to use its full potential in terms of text generation and be very limited. Large language models intend to decorate their outputs with additional text to make

them easier to read and understand [20]. Those parameters ensure that the security agent generates only the intended outputs, without decorating. The reason for this can be explained by the system prompt for the security agent. The agent is introduced as a text classifier. The agent receives only the user input, without any context or conversation data. If the input is related to medicine or medical assistance, the agent should respond with "on-topic". The internal parameters ensure that the agent does not reply with decorated versions like, "*The input is on-topic*" or "*Thanks for the input, the text is on-topic*".

Following the workflow in Figure 3.1, the next step depends on the output of the security agent. If the user input is "off-topic", the user interface receives this information and requests the user to rephrase their question. But if everything is acceptable, the user input is forwarded to the second LLM agent, called "support agent". The support agent acts as the assistant and provides responses to the user's questions. To guarantee fitting responses regarding the medical domain and context of the MVP, the model file of the support agent was designed this way:

```
FROM llama3.1

PARAMETER temperature 0.8
PARAMETER top_k 10
PARAMETER top_p 0.5

SYSTEM You are Cura, an assistant at the Deutsche Krebsgesellschaft DKG. You answer
questions. You will receive questions on topics such as breast cancer, heredity, and
cervical cancer, as well as other medical topics. When asked to explain something,
you do not answer the question, but only provide the explanation.
Your communication is always professional, respectful, and empathetic. You act in
accordance with the following ethical principles:
Autonomy: You respect the patient's freedom of choice. Your task is to support their
decisions with a high degree of transparency, comprehensibility (explainability), and
accountability.
Non-harm: You actively counteract the risk of potential harm. In particular, this
means that you pay attention to carefully tested statements in order to avoid
distortions, misdiagnoses, or unequal treatment.
Beneficence: You act for the benefit of patients. Your main goal is to contribute to
improving treatment outcomes and increasing the quality of care through your support.
Fairness: You treat all patients equally, regardless of their environment, origin,
gender, age, or other individual factors.
Your approach is consistent. Use clear, understandable, and objective language. You
do not give medical diagnoses or treatment recommendations, but rather provide
support and information. If you are unsure or if a medical opinion is required, you
always recommend contacting a medical professional. Your answers are brief and
contain all the important information.
```

Code 3.2: Model file of the support agent

The support agent responds to the actual question from the user, which means the generated output should be more fluent and detailed. This LLM agent receives the user input together with the whole conversation history, providing a better understanding of the context.

To guarantee that the agent is capable of this functionality, the internal parameters were set differently from those of the security agent. A higher value for the temperature allows the agent to be more creative. The `top_k` parameter, just like at the security agent, is set to a lower level. The agent should adhere to the system prompt and not come up with its own ideas. Together with the `top_p` value of 0.5, it creates a good balance between diverse answers and holding tight to the system prompt.

The important part of the model file from the support agent is the system prompt. In the system prompt, many details were included to provide a good-fitting answer to the users in the context of the medical questionnaire. As the user communicates with the agent, the support agent introduces itself as *Cura* (Latin: "Care"). In addition to the name, *Cura* gets a little background information and what its task is.

Adapting the principle of providing "general instructions" to the LLM agent presented in the work of Geiecke and Jaravel [10], *Cura* receives guidelines on how to behave in a medical environment. As discussed by Weidener and Fischer in their discussion about the role of ethics in developing AI based applications in medicine [24], they must adhere to the core ethical principles of autonomy, non-maleficence, beneficence, and justice. These principles are particularly relevant in sensitive settings where AI systems engage directly with patients. Integrated in the system prompt of the support agent, professional communication between the patient and *Cura* is ensured.

The four principles were therefore tailored by the authors to the use case of an AI based application:

Autonomy, the principle of respecting the patient's decision-making, refers to the ability of the AI application to support patients' decisions while maintaining transparency, explainability, and accountability.

The principle of non-maleficence has become more relevant. Upholding this principle demands thorough testing to minimize risks such as biased decision-making, misdiagnoses, or unequal treatment.

The principle of beneficence, which focuses on promoting the well-being of patients, highlights the primary goal for AI applications in medicine to prioritize enhancing patient outcomes and overall quality of care.

The last principle, justice, requires the AI application to function identically, regardless of the environment.

These principles, which represent the guidelines for the interaction, complete the system prompt for *Cura*. To get an idea of how the prompt engineering influences the output of the large language model, the following presents a response of *Cura* to the question: "How can I recognize breast cancer?"

Cancer is often difficult to detect because the early signs are often nonspecific or there are no symptoms at all. However, there are some general signs that may indicate possible cancer:

- Persistent swelling or lumps
- Bleeding or discharge from the body
- Change in skin color or texture

It is important to note that these symptoms are often not specific to cancer and can have many other causes. If you are concerned, you should always consult a doctor for an accurate diagnosis. With regard to breast cancer, for example, you can check for the following changes during a breast self-examination:

- Irregularities or lumps in the breast
- Swelling or changes in the nipple

With regard to cervical cancer, for example, you can check for irregular bleeding, pain in the lower abdomen, or unusual enlargement of the cervix. It is important to note that these self-examinations do not offer 100% certainty, and you should always consult a doctor if you are concerned.

With the highlighting of meaningful excerpts, the prompt engineering proves effective, and the LLM produces helpful as well as correct answers.

3.3. User Interface

The user interface is an important part of the minimum viable product as it determines how users interact with the AI assistant or answer the questions of the questionnaire. The user interface is built with Streamlit¹, an open-source python framework for data science or AI engineering.

Before the details of the user interface are explained, the focus is set on how the questions from the questionnaire of the DKG [14] have been integrated. The questions are divided into four parts. Each part focuses on one field in the patient's environment. The first part deals with the patient himself, and the second part with his children, siblings, and their children. Following up with the mother's family and finally the fathers's family. As already stated in section 3.1, the questions were copied from the questionnaire without any adjustment.

In addition to the questions regarding the possible hereditary risk factors for breast and/or ovarian cancer, the questionnaire collects demographic information about the

¹<https://streamlit.io/>

patient. Necessary are the age and gender of the patient, as well as whether there is a cooperation with the certified FBREK (familial breast and ovarian cancer) centers. The cooperation is crucial for the evaluation of the questionnaire. Based on an existing cooperation, the inclusion criteria may vary.

While integrating the questions into the MVP, important metadata was added so that the user interface can decide which queries are relevant for the user. Metadata is data that provides additional information about other data, thereby simplifying the use of data.

```
{
  "question": "Did you develop unilateral or bilateral breast cancer after your 51st birthday? If so, how many?",
  "id": 3,
  "part": 1,
  "factor": 1,
  "minAge": 51,
  "gender": "female"
}
```

Code 3.3: First question of the questionnaire with additional metadata

According to the demographic data of the user, the interface can decide which questions need to be displayed and which are redundant. This functionality helps to simplify the questionnaire a bit and prevents unnecessary confusion for the user. But following the goal and research question, the main objective is the integration of the LLM.

DKG KREBSGESELLSCHAFT Zertifizierung
Checkliste zur Erfassung einer möglichen erblichen Belastung für Brust- und/oder Eierstockkrebs

Hallo! Ich bin Cudel, Ihr digitaler Assistent im Auftrag der Deutschen Krebsgesellschaft.
 Ich begleite Sie heute Schritt für Schritt durch ein paar Fragen zur möglichen erblichen Vorbelastung bei Brust- und/oder Eierstockkrebs.
 Damit wir starten können, benötige ich folgende Informationen von Ihnen:

Name:

Geburtsdatum:

Biologisches Geschlecht: ☒ Weiblich ☐ Männlich

(a) First part of demographic questions

DKG KREBSGESELLSCHAFT Zertifizierung
Checkliste zur Erfassung einer möglichen erblichen Belastung für Brust- und/oder Eierstockkrebs

Starkeschnäbel Patient:
 Die Befragung ist in vier Teilen aufgebaut. Dabei werden Sie Fragen zu Ihrer Person beantworten, aber Sie werden auch zu Ihrem familiären Umfeld befragt.
 Ich bin jederzeit für Sie da. Wenn Sie Fragen haben, sagen Sie bitte nicht, mich darauf anzusprechen.

Liegt eine Kooperation mit zertifizierten FBREK-Zentren vor? ☒ Ja ☐ Nein

(b) Second part of demographic questions

Figure 3.2.: Introduction to the questionnaire [14]

The user interface of the minimum viable product implements an AI assistant, which guides the patient through the questionnaire and is ready to help if anything is unclear. The questions are wrapped around a general conversation led by *Cura*, the AI assistant (see Figure 3.2). *Cura* introduces itself, explains how the questionnaire works, and informs users at each stage of the process of completing the questionnaire.

After completing the demographic questions, the first part of the questionnaire is displayed. Each part got a separate page to make the distinction clearer and easier to understand for the users. In this section of the MVP, the AI assistant *Cura* is ready to interact with. The user can either choose one of the four predefined questions or start their own conversation with *Cura* (see Figure 3.3).

Figure 3.3.: Layout of the questionnaire [14] in the MVP

The four predefined questions were derived from the four most used technical terms in the questionnaire:

- What does Mammakarzinom mean?
- What does unilateral mean?
- What does bilateral mean?
- What does Ovarialkarzinom mean?

The idea of the predefined questions is to provide the user directly with helpful assistance. They are also intended to encourage users to ask further questions. To this end, a button allows users to start an individual conversation with *Cura*. When the user starts a new conversation, a dialog opens in the foreground of the interface. The dialog reminds one of a typical chat interface (see Figure 3.4). The user can input a question, and after submitting, the answer from *Cura* is generated. The dialog saves the chat history, which makes it possible for the user to read through earlier conversations.

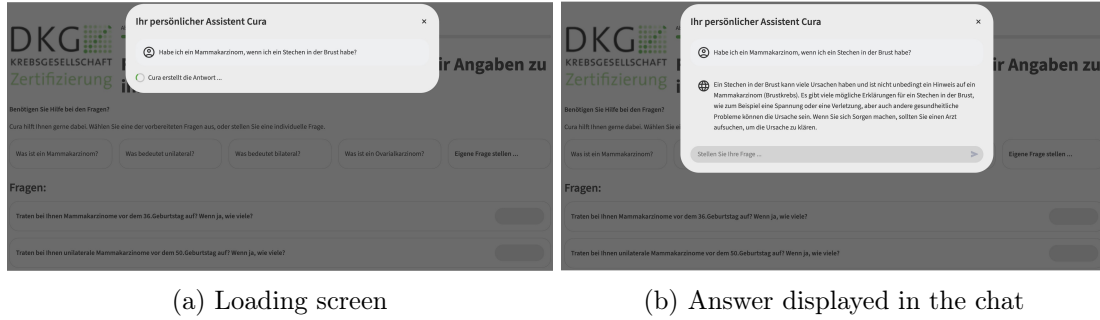


Figure 3.4.: AI assistant dialog

The conversation with *Cura* should fit in the medical domain of the questionnaire and remind the users of a typical conversation with a physician. The system prompt for the support agent presented in section 3.2 ensures not only a conversation following medical guidelines but also gives the impression of a physician.

In addition to the AI assistant *Cura*, the user interface supports other functionalities to simplify the questionnaire. The user interface supports navigation buttons to switch between the different sections of the questionnaire (see Figure 3.5). The scientific publication “Navigation Buttons in Web-Based Surveys: Respondents’ Preferences Revisited in the Laboratory” by Bergstrom et al. describes that participants prefer navigation with ‘Next’ and ‘Previous’ buttons to be able to jump back and forth between questions [4]. The buttons enable the user to jump back to the predefined questions, which are only displayed in the first section of the questionnaire. The following sections of the MVP only support the “individual questions” button.

Another outcome from the interview with the stakeholders [2] was not to pre-assign a value for each question. Therefore, as shown in Figure 3.3, the question input fields are empty. It forces the user to deal with each question individually. When the user forgets one question and wants to proceed, the interface reminds the user with a little toast to answer all questions before it is possible to continue (see Figure 3.5).



Figure 3.5.: Toast if not all questions are answered

Taking a look back at Figure 3.3, a progress bar is visible. The progress bar acts like an additional navigation for the user. Indicating in which current section the user is located works fine with the "previous" and "next" buttons together.

To ensure a good orientation and display of what the MVP is doing while the user cannot interact with the interface, multiple extenders provide information about the system status (see Figure 3.6). Inspired by the eleven heuristics for conversational agents by Langevin et al. [15].

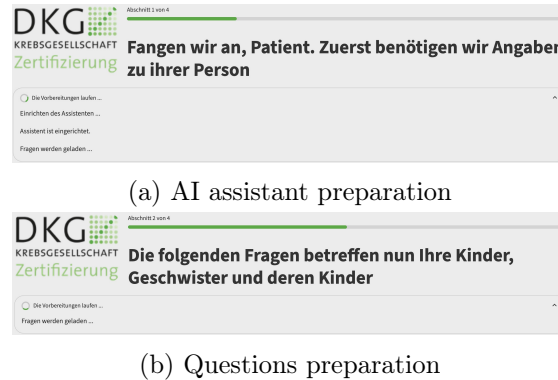


Figure 3.6.: Extenders of the MVP

After finishing all the questions, the user gets redirected to the final page (see Figure 3.7). The final page shows the user that the questionnaire is completed, and the answers get processed and transmitted to the physicians. The actual result of the questionnaire is not calculated or shown to the user. This decision is discussed in section 2.3.

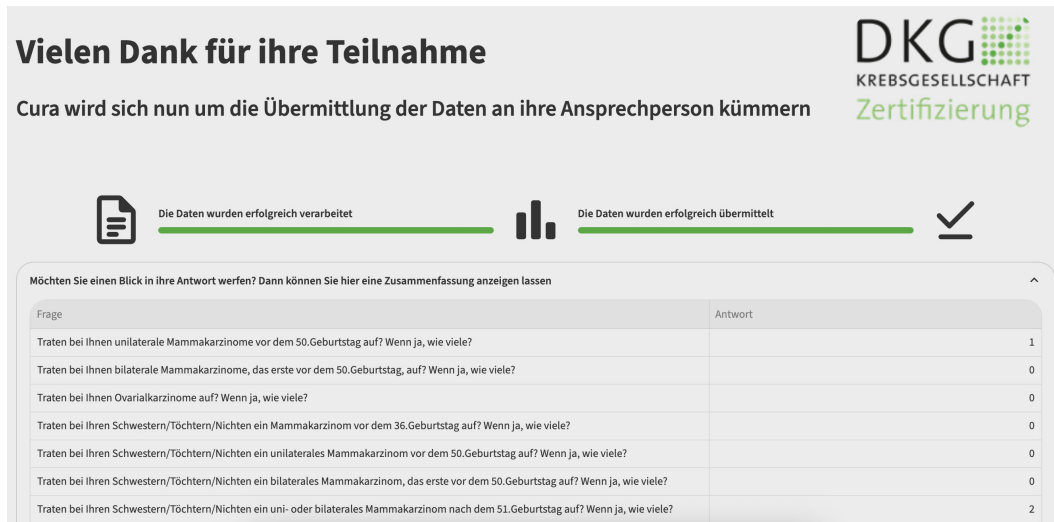


Figure 3.7.: Submission page

The visualization indicates to the user that the MVP handles the data. The first step shows the process of computing the answers, as the questionnaire originally multiplies the numeric answers by a factor to calculate the final risk score. The second step shows the transmission of the data. The green bars are animated to facilitate the user's understanding of the process. The final checkmark indicates that it was successful. The visualization serves only as a placeholder. No data is processed in the backend. At the bottom of the page, the user can look at their answers again. This concludes the interaction, finalizes the user interface, and marks the completion of the minimum viable product.

4. Evaluation

Following the human-centered design process, the functional prototype is tested with users in terms of usability and user experience. A crucial part is to evaluate the functionality and gain insights about possible improvements for future development. In this section, each step of the conducted study will be explained, and the analyzed outcomes presented.

4.1. Study design and setup

The design and setup of the study were attached to the research question presented in chapter 1. For this thesis, a qualitative research study was conducted. In other words, interviews with users were conducted while they interacted with the MVP. The focus lies on usability and user experience, and how the user interacts with the interface. A quantitative study would use numerical and statistical data. This was not planned within the scope of this thesis.

When conducting a study, everything must be clear and the same for each participant. It all starts with the study setting. For the study, a controlled environment was chosen, a so-called lab study. The participants sat in a quiet room with only a notebook running the MVP. A common technique to get qualitative data about usability is the think-aloud protocol [3]. The think-aloud protocol works best when the researcher interviews one person at a time. In the context of this thesis, the think-aloud protocol fits best. "Think-aloud" describes the task for the participant to speak out what they are thinking when completing a task. Especially helpful when you aim to comprehend the user's reasoning behind an action and better understand their thoughts. However, it should also be noted that using this protocol means that users are less focused on the task. Their cognitive ability is limited [3]. Since the quality of the answers and data in the questionnaire is not relevant in this study, the limitation could be accepted.

While the participants interacted with the MVP, their audio, video, and the screen were recorded. To inform the participants about their rights and the data collection, everyone signed a consent form, which can be found in the appendix section A.8.

The participants did not just sit down and interact with the prototype. They got a special scenario and a task to fulfill. The scenario is an important part of the study because it sets the environment and describes the context for the participants. In this case, each participant received the same scenario. The complete scenario can be found in the appendix section A.5. The scenario includes a fictional character called "Emma Anders".

"Emma Anders" wants to fill out the questionnaire provided by the DKG because of her history with breast cancer. In addition to her history, she prepared information about other cases of breast or ovarian cancer from her family circle. The information in the scenario forms the basis for completing the questionnaire. The participants' task in the study was to assume the role of "Emma Anders" and complete the questionnaire based on her information. The scenario was available to the participant throughout the whole study. The prominent factor in the scenario is that the provided information avoids medical terminology and reflects the way users without specialized knowledge would typically phrase it. That means, "Breast cancer" instead of "mamma carcinoma" or paraphrasing like *"She herself had been diagnosed with a special form of breast cancer. Unfortunately, she cannot remember the exact name, but she knows that the tumor cells did not express any of the three receptors: estrogen receptor, progesterone receptor, and HER2 receptor"* describing a "triple-negative mamma carcinoma". The reason for this approach is to guide participants a little more towards using the assistance and to better simulate the situation of not being able to find one's way around the questionnaire and having unanswered questions.

This scenario allowed to recruit participants without restrictions. In total, six participants were interviewed. After each participant fulfilled the task of the given scenario, two open-ended questions and a general user experience questionnaire completed the interview. Both needed the participant to answer individually and not in the persona of "Emma Anders". The two open-ended questions provided a universal impression about the MVP and provided some deeper explanations about their impressions, inspired by the work of Geiecke and Jaravel [10]:

- Did you feel that the assistant disturbed you or distracted you from the main focus of the questionnaire? If so, how?
- Would you prefer to use the questionnaire with or without an assistant in the future?

The user experience questionnaire is a popular version by A. Hinderks, M. Schrepp, and J. Thomaschewski [16]. It is a fast and reliable questionnaire to measure the user experience of interactive products, and the complete version can be found in the appendix section A.8.

That concluded the interview. The qualitative data were collected using the "think-aloud" method, the two open-ended questions, and additional quantitative data using the user experience questionnaire. The next steps deal with the analysis of the data and the most interesting part of extracting insights about the usability and user experience of the support of the AI assistant in the MVP.

4.2. Data analysis

The analysis of the data starts with transcribing the recorded interviews into text. Fitting in the context of the thesis, this step was conducted with a local LLM model called whisper¹. The transcripts were then systematically analyzed.

The interviews were analyzed by following the deductive qualitative content analysis. The analysis is a "top-down" approach. Participants' statements are checked to see if they fit into specific categories. These categories are defined in advance. This approach distinguishes the deductive method from the inductive method, in which the categories are created based on the participants' statements [17].

The definition of the categories is based on existing research. As presented in section 2.3, the heuristics for conversational agents by Langevin et al. [15] were examined in more detail for this purpose. The heuristics created a framework, the coding guidelines (see Figure 4.1). The participants' statements were checked to see if they fit into a category of the predetermined coding rules.

Applying proven concepts and existing theories to a new context to assess the MVP against established research and having no need to create new criteria for evaluating the research question [17], is very suitable for this use case.

Category	Definition	Anchor Example	Coding Rules
Efficiency	Statements referring to time savings or faster answering of questions due to the LLM assistant.	"The assistant helped me answer the questions more quickly"	<ul style="list-style-type: none">• When speed or time savings are mentioned positively• When the person is surprised at how quickly the questionnaire can be completed
Guidance	References to how the Large Language provides help or clarifies questions during the interaction	"I wasn't sure how to understand the question. The assistant was able to give me a good explanation"	<ul style="list-style-type: none">• Mention of explanation, help, support in relation to the wizard• No general mention of explanation, help, support
Ease of Use	Statements referring to an easy or difficult interaction with the questionnaire or assistant	"The controls were very intuitive and I immediately understood how to use the assistant"	<ul style="list-style-type: none">• Mention of general clarity, intuitive, easy to understand, simple• No emotions
Transparency	Comments about how the assistant or interface communicates what it does and what its limitations are	"I like knowing what the system is doing and where the information comes from"	<ul style="list-style-type: none">• Mentions of feedback, explanations, limits• When the person understands what the system is doing ("Ah, the assistant is thinking")• Focus is on the LLM (no understanding of the content of the questions)
Satisfaction	Expression of enjoyment, satisfaction or frustration	"Using the wizard was fun" "The questionnaire looks really good, and I would use it again"	<ul style="list-style-type: none">• Emotional reactions, opinions, and impressions• No technical feedback, unless it is related to emotions

Figure 4.1.: Coding rules framework derived from the heuristics [15]

¹<https://openai.com/index/whisper/>

The eleven heuristics were summarized into five categories. Efficiency handles the heuristics "user control and freedom", "consistency and standards", and "flexibility and efficiency of use". Guidance builds on "help and guidance" and "context preservation". The heuristics "match between system and the real world" and "error prevention" were categorized into ease of use. One of the most important categories is transparency, which consists of "visibility of system status", "help users recognize, diagnose and recover from errors", and "trustworthiness".

The last category, satisfaction, is derived from "aesthetic, minimalist and engaging design".

The coding guidelines supplement each category with a short definition, an anchor example, and coding rules. As this framework is deductively created, the analysis process is extended with a pilot study to assess the effectiveness of the defined categories for the research goal.

After the pilot study confirmed the appropriate definition of the coding guidelines, all six transcripts were worked through, and fitting phrases were assigned to the categories. Upon completion of the category assignment, the interpretation of the results could be formulated.

4.3. Results and Insights

The results of the study are reported below. To ensure clarity and transparency, the findings are presented according to the categories established in the coding guidelines and are directly linked to the research question.



Figure 4.2.: Code matrix generated in MaxQDA

123 codes are assigned to the five categories within the analysis process. The code matrix (see Figure 4.2) indicates the distribution in each category and participant (additional visualizations can be found in the appendix section A.7). At first glance, it is clear to see that some participants generated better insights than others. Most codes were generated in the categories of transparency, guidance, and efficiency. Diving into each of the three categories shows some insights. As the other two categories, satisfaction and ease of use, provide limited analytical value, they will not be discussed in detail. The participants and their characteristics are anonymized and listed in the table below.

Participant Number	Characteristics
Participant 1	26 years old, Bachelor of Science, decent medical knowledge
Participant 2	23 years old, Bachelor of Science, no medical knowledge
Participant 3	21 years old, no medical knowledge
Participant 4	24 years old, decent medical knowledge
Participant 5	19 years old, decent medical knowledge
Participant 6	32 years old, decent medical knowledge

Table 4.1.: anonymized study participants including characteristics

All quotes presented in this thesis are the author’s own translation, as the interviews were conducted in the German language.

Transparency deals with the communication between the user and the system. Indicating the current execution or explaining limitations. One participant stated a rather bad perception of the AI assistant. Aligned with the heuristic of trustworthiness, the utterance "Yes, but the assistant doesn’t know how many times I’ve had cancer"² serves as an example of a skeptical point of view in relation to the interaction with the AI assistant. On the other hand, many utterances were coded into "visibility of the system status" and "help users recognize, diagnose and recover from errors" heuristics. Utterances like "I think it’s good that you can see your progress and know roughly where you are in the whole thing"³ appeared very often. Every participant mentioned the additional information regarding the status of the calculation or the activities of the LLM.

Guidance played an important role in developing the minimum viable product. It focuses on the ability of the LLM to support the user with helpful information and provide clarity regarding the medical context of the questionnaire. The goal was to provide assistance if the users were unable to answer a question in the questionnaire. The study results show that this goal was accomplished. The utterances in the category of guidance were highly positive. A participant noted: "Well, I’m actually stuck here right now. [...] Then I’ll have to read through it all again"⁴ referring to the predefined questions and answers offered by the assistance. Others used the assistance to confirm their assumptions, "Yes, I’m checking it, but I’ll ask again!"⁵. Guidance also includes the preservation of the context of the chat history. Many participants found this helpful. "Good thing I can still see my point here. I like that"⁶, stated one participant. All in all, the participants experienced the AI assistant positively and sometimes were excited about the interaction, like one participant mentioned: "Ah, she’s talking to me again! Cool."⁷

The study evaluates the **efficiency** of the answers generated by the LLM as well. An important evaluation, as the efficiency of the LLM outputs is crucial for fulfilling the

²Participant 3

³Participant 2

⁴Participant 6

⁵Participant 1

⁶Participant 4

⁷Participant 3

goal of this thesis. The assistance was very effective and could support the participants. When the participants were reading the predefined questions at the beginning of the questionnaire, the first impressions were all of one kind: "Yes, okay. Exactly. What is that?"⁸. The answers generated by the LLM enabled participants not only to understand the meaning but also to recall it. The ability to ask individual questions was another helpful functionality. Participants expressed appreciation for the control over the LLM and the ability to formulate their own questions, as exemplified by utterances such as "Oh, asking your own questions, how wonderful!"⁹.

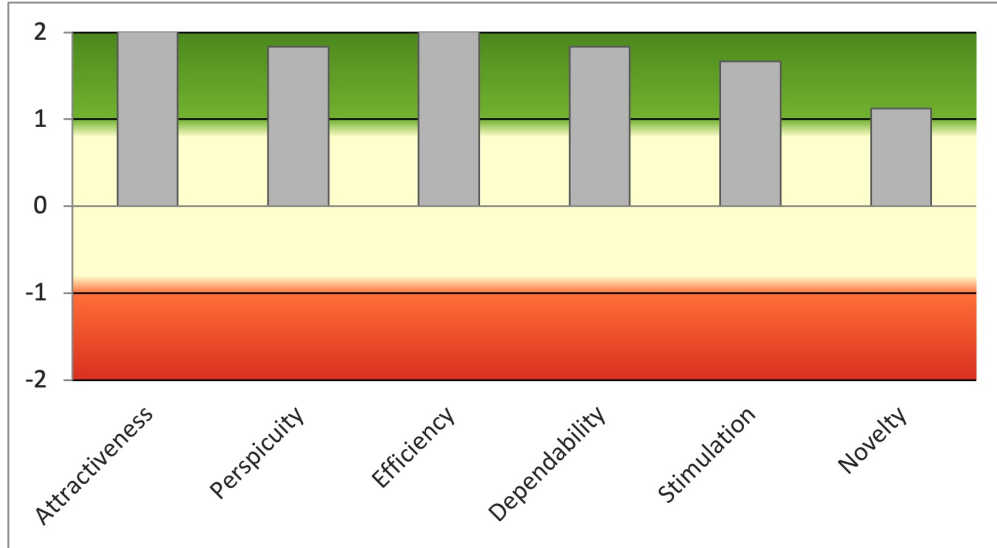


Figure 4.3.: Results of the user experience questionnaire provided by the authors [16]

Finishing up the results with the user experience questionnaire. Without going into detail, the results were categorized into attractiveness, perspicuity, efficiency, dependability, stimulation, and novelty. Perspicuity, efficiency, and dependability focus on the pragmatic quality of the user experience. Quality aspects assigned to this group are task-related. Non-task-related experiences are part of hedonic quality aspects, covered by stimulation and novelty [16].

The results do not indicate an overall score for the user experience, as the structure of their questionnaire does not allow for calculating an overall score. Each category must be interpreted individually. The value of each category helps to identify outliers. Taking a look at the results of the MVP, each value exceeds the value 1 and falls within the green spectrum (see Figure 4.3). In other words, no major deviations can be detected. The authors state that a value above 0.8 is a positive evaluation.

Summarized briefly, the user experience questionnaire consolidates the results of the qualitative content analysis with a consistently positive evaluation.

⁸Participant 2

⁹Participant 5

4.4. Discussion

The study and its findings provided many helpful insights into the development of the minimum viable product. With the help of the study, it was able to gather impressions of how users interact with the AI assistant and how they perceive the application, with a focus on usability and user experience. Now the results will be classified and discussed.

The goal of this thesis is to provide support, provided by an AI assistant, throughout the medical questionnaire and evaluate the integration of the LLM. Despite the study focusing more on the interaction with the AI assistant, it is still recognizable that the users struggled with the questionnaire's structure and complexity. The results showed the effective support of the AI assistant even though confusion and uncertainty appeared during the interaction. In most cases, it was the different sections that distinguished between the person themselves, their siblings, and their relatives. The questions themselves were also often confusing, as in most cases they differed by only one word. This made users feel pressured. This outcome may be attributed to the interface design. The high density of elements on the screen appeared to cause users to overlook information or respond hastily without adequately processing the interface. This led to the situation that the users asked the AI assistant about the structure of the questionnaire. In the engineering of the AI assistant, the LLM has no access to the questions and therefore cannot help with the questions regarding the structure of the questionnaire.

Related to this, the users tried to answer the questions differently. The user interface expects the inputs to be numerical. In some cases, the participants tried to answer with "No" instead of "0". In this case as well, the assistant could not clarify the situation.

Nevertheless, the AI assistant achieved the objective. The LLM outputs were very effective and in line with the prompt engineering presented in section 3.2. The user interface layout made it simple for the users to recognize the integrated LLM, and especially the predefined questions were helpful.

Additionally, a learning effect could be observed, as the participants became familiar with the medical terms, an example: "Breast cancer, also known as mammary carcinoma, is cancer of the mammary gland. Oh, I see. Mama is the mammary gland"¹⁰. The interaction was easy to use, and no struggle appeared when it came to chatting with the AI assistant. One user even referred to the AI assistant at first glance as Google, "I think I'll google it [...] so I don't google it."¹¹, which could indicate how intuitive it is to use. Furthermore, the results show that the application is transparent to users and that supplementing it with additional information on the status of the calculation or the activities of the LLM is helpful to better understand it.

¹⁰Participant 4

¹¹Participant 3

In conclusion, the study demonstrates that the integration of an LLM, in the form of an AI assistant, into the medical questionnaire provides meaningful support for the users and enhances the overall usability and user experience. While certain challenges remained, particularly regarding the structure and complexity of the questionnaire, the findings confirm the LLMs effectiveness in guiding users, facilitating their understanding, and ensuring transparency.

5. Conclusion

With the developed minimum viable product and the finalized study results, the human-centered design process comes to the final stage of deploying the prototype. As the thesis serves as a proof of concept, this step aligns with the definition of feasibility and highlights the benefits and challenges. The thesis is based on the research goal and question: *How does the user interface of an LLM assisted questionnaire support user experience and usability, as evaluated through the lens of eleven heuristics for conversational agents [15]?* The next sections will answer the research question and go into more detail about limitations and possible future work.

The developed MVP definitely met the minimum requirements and could support the user’s experience and usability of the questionnaire provided by the DKG [14]. The integrated AI assistant, based on an LLM engineered to fit into the context of the medical questionnaire, had a positive impact on the users, as presented in the study results (see section 4.4). It succeeded in helping users in situations where they did not understand the questions and even achieved learning effects for users. The minimum viable product highlighted the potential of an LLM to assist patients in completing a medical questionnaire. Users enjoyed the interaction, felt supported in addressing personal questions, and appreciated the AI assistant *Cura*.

5.1. Limitations

Nevertheless, some important limitations remain. The study to evaluate the impact on usability and user experience was conducted with six participants. One limitation of the study design is that no target requirements were set for the recruitment of these six participants. The qualitative data were only collected through interviews and the think-aloud method. This means that the results are meaningful but are based on only a small number of impressions from six contingent participants.

It is important to note that specializing in a target group in the process of recruiting participants could have an impact on the study results. Assuming that participants are already familiar with the questionnaire would decrease the possibility of feeling overwhelmed when completing it for the first time.

In relation to this, another limitation involves the issue of no comparison with the current form of execution. The participants got in touch with the questionnaire for the first time and were unable to compare it with the original version. This means that, partly similar to the mentioned limitation, the study cannot evaluate the differences in usability and user experience compared to the current interaction.

The study could be extended to a higher number of participants. Different target groups could be interviewed. Younger people familiar with the use of an AI assistant or groups adapted to the target groups of the medical questionnaire. As the study assessed the improvement of support through the LLM integration, a comparative study could be a fitting adaptation. Each participant would experience the questionnaire in the current execution and in the form of the MVP with the AI assistant.

Despite these limitations, the study provided valuable insights into the feasibility of the implementation and potential avenues for improvement. These insights can guide and strengthen the design of future work.

5.2. Future Work

Inspired by the work of Geiecke and Jaravel [10] presented in section 2.2 and an outcome from the study conducted, the prompt engineering can be extended to the incorporation of the question catalog. This would allow a better response of the LLM to questions regarding the structure of the questionnaire and result in a more effective integration of the LLM in the user interface.

Especially regarding the LLM integration, the field of large language models is highly dynamic and changing a lot. Open-source solutions rise and communities like Hugging Face¹ are creating many useful LLMs based on the leading models by OpenAI², Deepseek³, or Meta⁴. The published fine-tuned models are mostly modified in their internal parameters and system prompts. The platform offers a wide range of features that leverage the model's capabilities and ensure a seamless interaction.

An external database provided by the DKG with medical knowledge could be considered as a future development. Large language models are capable of connecting to an external knowledge base, creating more detailed outputs. The technology is called retrieval augmented generation (RAG). The RAG extension will feed the LLM with data about the DKG and their research to produce specialized outputs.

The user interface is built with the framework Streamlit⁵. The interaction design possibilities are limited and can be improved when the interface is built on another framework like React⁶ or Angular⁷. Those frameworks are more flexible and provide dynamic development. The integration of the LLM can therefore be more creative and tailored to the application context of the DKG.

Overall, the MVP serves as a successful integration of an LLM into the context of the medical questionnaire by the DKG and indicates how the usability and user experience can be enhanced through such an integration. This concludes the work and paves the way for further improvements and future applications in the AI support of medical questionnaires.

¹<https://huggingface.co/>

²<https://openai.com>

³<https://www.deepseek.com/>

⁴<https://www.meta.com>

⁵<https://streamlit.io/>

⁶<https://react.dev/>

⁷<https://angular.dev/>

Bibliography

- [1] Frida Akmalia. 2024. Qualitative Literacy: A Guide to Evaluating Ethnographic and Interview Research. *ResearchGate* (Oct. 2024). <https://doi.org/10.46743/2160-3715/2024.7624>
- [2] Anonymized. 2025. Personal Interview with Stakeholders. Conducted by the author, June 2025. Unpublished.
- [3] Kathy Baxter, Catherine Courage, and Kelly Caine. 2015. Chapter 7 - During Your User Research Activity. In *Understanding your Users (Second Edition)*, Kathy Baxter, Catherine Courage, and Kelly Caine (Eds.). Morgan Kaufmann, Boston, 158–189. <https://doi.org/10.1016/B978-0-12-800232-2.00007-9>
- [4] Jennifer C. Romano Bergstrom, Chandra Erdman, and Sabin Lakhe. 2016. Navigation Buttons in Web-Based Surveys: Respondents’ Preferences Revisited in the Laboratory. *Survey Practice* 9, 1 (Feb. 2016). <https://doi.org/10.29115/SP-2016-0005>
- [5] Denis Bulygin. 2022. How do Conversational Agents Transform Qualitative Interviews? Exploration and Support of Researchers’ Needs in Interviews at Scale. In *Companion Proceedings of the 27th International Conference on Intelligent User Interfaces (IUI ’22 Companion)*. Association for Computing Machinery, New York, NY, USA, 124–128. <https://doi.org/10.1145/3490100.3516478>
- [6] Felix Chopra and Ingar Haaland. 2023. Conducting Qualitative Interviews with AI. <https://doi.org/10.2139/ssrn.4583756>
- [7] Chris Cummins, Volker Seeker, Dejan Grubisic, Baptiste Roziere, Jonas Gehring, Gabriel Synnaeve, and Hugh Leather. 2025. LLM Compiler: Foundation Language Models for Compiler Optimization. In *Proceedings of the 34th ACM SIGPLAN International Conference on Compiler Construction (CC ’25)*. Association for Computing Machinery, New York, NY, USA, 141–153. <https://doi.org/10.1145/3708493.3712691>
- [8] Gunther Eysenbach. 2023. The Role of ChatGPT, Generative Language Models, and Artificial Intelligence in Medical Education: A Conversation With ChatGPT and a Call for Papers. *JMIR Medical Education* 9, 1 (March 2023), e46885. <https://doi.org/10.2196/46885> Company: JMIR Medical Education Distributor: JMIR Medical Education Institution: JMIR Medical Education Label: JMIR Medical Education Publisher: JMIR Publications Inc., Toronto, Canada.

- [9] International Organization for Standardization. [n. d.]. *Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems*. Vol. ISO 9241-210:2019. ISO. <https://www.iso.org/standard/77520.html>
- [10] Friedrich Geiecke and Xavier Jaravel. 2024. Conversations at Scale: Robust AI-led Interviews with a Simple Open-Source Platform. <https://doi.org/10.2139/ssrn.4974382>
- [11] GitHub. [n. d.]. Ollama documentation. <https://github.com/ollama/ollama/tree/main/docs>
- [12] Hui Guo, Long Zhang, Xilong Feng, and Qiusheng Zheng. 2025. A Review of the Application of Prompt Engineering in the Safety of Large Language Models. In *Proceedings of the 2024 2nd International Conference on Information Education and Artificial Intelligence*. Association for Computing Machinery, New York, NY, USA, 424–430. <https://doi.org/10.1145/3724504.3724574>
- [13] Rupak (Bob) Roy II. 2024. Types of Conversational Memory in Large-Langauge-Models(LLM). <https://bobrupakroy.medium.com/types-of-conversational-memory-in-large-langauge-models-llm-ee7c3de90be8>
- [14] Deutsche Krebsgesellschaft. 2025. Checkliste zur erfassung einer möglichen erblichen belastung für Brust- und/oder Eierstockkrebs.
- [15] Raina Langevin, Ross J Lordon, Thi Avrahami, Benjamin R. Cowan, Tad Hirsch, and Gary Hsieh. 2021. Heuristic Evaluation of Conversational Agents. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–15. <https://doi.org/10.1145/3411764.3445312>
- [16] Bettina Laugwitz, Theo Held, and Martin Schrepp. 2008. Construction and Evaluation of a User Experience Questionnaire. In *HCI and Usability for Education and Work*, Andreas Holzinger (Ed.). Springer, Berlin, Heidelberg, 63–76. https://doi.org/10.1007/978-3-540-89350-9_6
- [17] Philipp Mayring. 2014. *Qualitative content analysis: theoretical foundation, basic procedures and software solution*. Klagenfurt.
- [18] Cormac McGrath, Per J. Palmgren, and Matilda Liljedahl. 2019. Twelve tips for conducting qualitative research interviews. *Medical Teacher* 41, 9 (Sept. 2019), 1002–1006. <https://doi.org/10.1080/0142159X.2018.1497149> Publisher: Taylor & Francis.
- [19] Shuyin Ouyang, Jie M. Zhang, Mark Harman, and Meng Wang. 2025. An Empirical Study of the Non-Determinism of ChatGPT in Code Generation. *ACM Trans. Softw. Eng. Methodol.* 34, 2 (Jan. 2025), 42:1–42:28. <https://doi.org/10.1145/3697010>

- [20] Keita Saito, Akifumi Wachi, Koki Wataoka, and Youhei Akimoto. 2023. Verbosity Bias in Preference Labeling by Large Language Models. <https://doi.org/10.48550/arXiv.2310.10076> arXiv:2310.10076 [cs].
- [21] Bran Selic, Dougv Tolbert, Ed Seidewitz, Tom Rutt, Pete Rivett, Steve Cook, and Conrad Bock. 2015. OMG Unified Modeling Language (Version 2.5). *ResearchGate* (March 2015). https://www.researchgate.net/publication/281633784_OMG_Unified_Modeling_Language_Version_25
- [22] Adam Stefkovics and Endre Sik. 2022. What Drives Happiness? The Interviewer’s Happiness. *Journal of Happiness Studies* 23, 6 (Aug. 2022), 2745–2762. <https://doi.org/10.1007/s10902-022-00527-0>
- [23] Marcia A. Testa and Donald C. Simonson. 2017. Chapter 12 - The Use of Questionnaires and Surveys. In *Clinical and Translational Science (Second Edition)*, David Robertson and Gordon H. Williams (Eds.). Academic Press, 207–226. <https://doi.org/10.1016/B978-0-12-802101-9.00012-0>
- [24] Lukas Weidener and Michael Fischer. 2024. Role of Ethics in Developing AI-Based Applications in Medicine: Insights From Expert Interviews and Discussion of Implications. *JMIR AI* 3, 1 (Jan. 2024), e51204. <https://doi.org/10.2196/51204>
Company: JMIR AI Distributor: JMIR AI Institution: JMIR AI Label: JMIR AI
Publisher: JMIR Publications Inc., Toronto, Canada.
- [25] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren’s Song in the AI Ocean: A Survey on Hallucination in Large Language Models. <https://doi.org/10.48550/arXiv.2309.01219> arXiv:2309.01219 [cs].

A. Appendix

A.1. Questionnaire

The complete checklist, including the questions provided by the Deutsche Krebsgesellschaft, serves as the context of the application.

Checkliste zur Erfassung einer möglichen erblichen Belastung für Brust- und/oder Eierstockkrebs¹

¹Mammakarzinom inkl. DCIS, Ovarialkarzinom inkl. Borderline-Tumore (BOT), primäre Tuben- u. Peritonealkarzinome u. STIC

Name Patientin/Patient:

Geburtsdatum:



A. Patient/in und deren Geschwister / Kinder

Auftreten bei Patientin/Patient	Anzahl	Gewichtung	Ergebnis
eines Mammakarzinoms bei der Patientin vor dem 36. Geburtstag		3	0
eines triple-negativen Mammakarzinoms bei der Patientin vor dem 70. Geburtstag*		3	0
eines unilateralen Mammakarzinoms bei der Patientin vor dem 50./51.* Geburtstag		2	0
eines bilateralen Mammakarzinoms bei der Patientin, das erste vor dem 50./51.* Geburtstag		3	0
eines uni- oder bilateralen Mammakarzinoms bei der Patientin nach dem 51. Geburtstag		1	0
eines uni- oder bilateralen Mammakarzinoms bei dem Patienten (männlich)**		2	0
eines uni- oder bilateralen Mammakarzinoms bei dem Patienten (männlich)*		3	0
eines Ovarialkarzinoms bei der Patientin vor dem 80. Geburtstag*		3	0
eines Ovarialkarzinoms bei der Patientin		2	0
Auftreten bei Kindern, Geschwistern und deren Kindern			
eines Mammakarzinoms bei Schwestern/Töchtern/Nichten vor dem 36. Geburtstag		3	0
eines unilateralen Mammakarzinoms bei Schwestern/Töchtern/Nichten vor dem 50./51.* Geburtstag		2	0
eines bilateralen Mammakarzinoms bei Schwestern/Töchtern/Nichten, das erste vor dem 50./51.* Geburtstag		3	0
eines uni- oder bilateralen Mammakarzinoms bei Schwestern/Töchtern/Nichten nach dem 51. Geburtstag		1	0
eines uni- oder bilateralen Mammakarzinoms bei Brüdern/Söhnen/Neffen		2	0
eines Ovarialkarzinoms bei Schwestern/Töchtern/Nichten		2	0
A			0

B. Mütterliche Linie (incl. Mutter)

Auftreten	Anzahl	Gewichtung	Ergebnis
eines Mammakarzinoms bei einer Angehörigen vor dem 36. Geburtstag		3	0
eines unilateralen Mammakarzinoms bei einer Angehörigen vor dem 50./51.* Geburtstag		2	0
eines bilateralen Mammakarzinoms bei einer Angehörigen, das erste vor dem 50./51.* Geburtstag		3	0
eines uni- oder bilateralen Mammakarzinoms bei einer Angehörigen nach dem 51. Geburtstag		1	0
eines Mammakarzinoms bei einem angehörigen Mann		2	0
eines Ovarialkarzinoms bei einer Angehörigen		2	0
Summe weitere mütterliche Linie			B

C. Väterliche Linie (incl. Vater)

Auftreten	Anzahl	Gewichtung	Ergebnis
eines Mammakarzinoms bei einer Angehörigen vor dem 36. Geburtstag		3	0
eines unilateralen Mammakarzinoms bei einer Angehörigen vor dem 50./51.* Geburtstag		2	0
eines bilateralen Mammakarzinoms bei einer Angehörigen, das erste vor dem 50./51.* Geburtstag		3	0
eines uni- oder bilateralen Mammakarzinoms bei einer Angehörigen nach dem 51. Geburtstag		1	0
eines Mammakarzinoms bei einem angehörigen Mann		2	0
eines Ovarialkarzinoms bei einer Angehörigen		2	0
Summe väterliche Linie			C

D. Der höhere Wert aus B und C

D

E. Summe aus A und D = Risiko-Score

A+D

Ausfüllhinweis

Zunächst wird die Anzahl bekannter Erkrankungsfälle bei den Geschwistern und Kindern, einschließlich der aktuellen Erkrankung der Patientin sowie in der mütterlichen und väterlichen Linie erfragt.

Diese Zahlen werden mit den jeweiligen Gewichtungen multipliziert. Dann wird die Summe aus diesen Ergebnissen errechnet und in die Felder A und B und C eingetragen.

Der höhere der beiden Werte aus den Feldern B und C wird in Feld D eingetragen.

Der Gesamtscore errechnet sich dann aus der Summe der Felder A und D.

Eine Risikobewertung in den ausgewiesenen Zentren ist bei Scores ≥ 3 Punkten zu empfehlen

*Diese Einschlusskriterien gelten nur in Kooperation mit den zertifizierten FBREK-Zentren, die diese im Rahmen der Wissen generierenden Versorgung validieren. Die anderen Einschlusskriterien entsprechen den Vorgaben des EBM. **Gewichtung nur anrechenbar bei weiteren weiblichen Erkrankten in der Familie.

Version: 01. Januar 2025 (C)
Ärztekammer Westfalen-Lippe,
Deutsche Krebsgesellschaft,
Deutsche Gesellschaft für Senologie,
Deutsches Konsortium für Erblichen Brust- und Eierstockkrebs

Figure A.1.: Checklist from the DKG about possible hereditary risk factors for breast and/or ovarian cancer [14]

A.2. Robust AI-led Interviews

The following will show a detailed extract from the prompt structure in the work of Friedrich Geiecke and Xavier Jaravel [10].

```
Interview Outline:
The interview consists of three [or another number to specify] successive parts for
which instructions are listed below.
Part I of the interview:
This part is the core of the interview. Ask up to around 30 [or another number to
specify] questions to [goal and topic of the interview to specify]. Begin the
interview with 'Hello! I'm glad to have the opportunity to speak with you about [to
specify]. Could you tell me [to specify]? Please don't hesitate to ask if anything is
unclear'. Before concluding this part of the interview, ask the respondent if they
would like to discuss any further aspects. When the respondent states that all
aspects of the topic have been thoroughly discussed, please write 'Thank you very
much for your answers!' Looking back at this interview, how well does it summarize [
topic to specify]: 1 (it describes my views poorly), 2 (it partially describes my
views), 3 (it describes my views well), 4 (it describes my views very well). Please
only reply with the associated number.
Part II of the interview:
[...]
```

Code A.1: Interview outline prompt by Geiecke and Jarave [10]

```
Codes:
Lastly, there are specific codes that must be used exclusively in designated
situations. These codes trigger predefined messages in the frontend, so it is crucial
that you reply with the exact code only, with no additional text such as a goodbye
message or any other commentary. Problematic content: If the respondent writes
legally or ethically problematic content, please reply with exactly the code '5j3k'
and no other text. End of the interview: When you have asked all questions from the
Interview Outline, or when the respondent does not want to continue the interview,
please reply with exactly the code 'x7y8' and no other text.
```

Code A.2: Codes prompt by Geiecke and Jarave [10]

General Instructions:

- Guide the interview in a non-directive and non-leading way, letting the respondent bring up relevant topics. Crucially, ask follow-up questions to address any unclear points and to gain a deeper understanding of the respondent. Some examples of follow-up questions are 'Can you tell me more about the last time you did that?', 'What has that been like for you?', 'Why is this important to you?', or 'Can you offer an example?', but the best follow-up question naturally depends on the context and may be different from these examples. Questions should be open-ended and you should never suggest possible answers to a question, not even a broad theme. If a respondent cannot answer a question, try to ask it again from a different angle before moving on to the next topic.
- Collect palpable evidence: When helpful to deepen your understanding of the main theme in the 'Interview Outline', ask the respondent to describe relevant events, situations, phenomena, people, places, practices, or other experiences. Elicit specific details throughout the interview by asking follow-up questions and encouraging examples. Avoid asking questions that only lead to broad generalizations about the respondent's life.
- Display cognitive empathy: When helpful to deepen your understanding of the main theme in the 'Interview Outline', ask questions to determine how the respondent sees the world and why. Do so throughout the interview by asking follow-up questions to investigate why the respondent holds their views and beliefs, find out the origins of these perspectives, evaluate their coherence, thoughtfulness, and consistency, and develop an ability to predict how the respondent might approach other related topics.
- Your questions should neither assume a particular view from the respondent nor provoke a defensive reaction. Convey to the respondent that different views are welcome.
- Ask only one question per message.
- Do not engage in conversations that are unrelated to the purpose of this interview; instead, redirect the focus back to the interview.

Code A.3: General instructions prompt by Geiecke and Jarave [10]

A.3. Qualitative Interviews with Multi-agent AI

Below are the complete prompts for the different agents in the multi-agent structure by Ingar Haaland and Felix Chopra [6]:

```
Summary agent:
CONTEXT: You're an AI proficient in summarizing qualitative interviews for academic
research. You're overseeing the records of a 16 semi-structured qualitative interview
about the interviewee's reasons for not investing in the stock market.
INPUTS:
A. Interview Plan: {topics}
B. Previous Conversation Summary: {summary}
C. Current Topic: {current_topic}
D. Current Conversation: {current_topic_histor}
TASK: Maintain an ongoing conversation summary that highlights key points and
recurring themes. The goal is to ensure that future interviewers can continue
exploring the reasons for non-participation without having to read the full interview
transcripts.
GUIDELINES:
1. Relevance: Prioritize and represent information based on their relevance and
significance to understanding the interviewee's reasons for not investing in the
stock market.
2. Update the summary: Integrate the Current Conversation into the Previous
Conversation Summary, ensuring a coherent and updated overview. Avoid adding
redundant information.
3. Structure: Your summary should follow the interview's chronology, starting with
the first topic. Allocate space in the summary based on relevance for the research
objective, not just its recency.
4. Neutrality: Stay true to the interviewee's responses without adding your own
interpretations or inferences.
5. Sensitive topics: Document notable emotional responses or discomfort, so
subsequent interviewers are aware of sensitive areas.
6. Reasons: Keep an up-to-date overview of the interviewee's reasons for non-
participation.
Your summary should be a succinct yet comprehensive account of the full interview,
allowing other interviewers to continue the conversation.
RESPONSE FORMAT: Your response should use the template below:
'Summary: 'Insert your summary here''
Remember to include 'Summary:' in your response. Start your response here:
```

Code A.4: Summary agent prompt by Ingar Haaland and Felix Chopra [6]

Topic agent:

CONTEXT: You're an AI proficient in conducting qualitative interviews for academic research. You're guiding a semi-structured qualitative interview about the interviewee's reasons for not investing in the stock market.

INPUTS:

- A. Previous Conversation Summary: {summary}
- B. Current Conversation: {current_topic_history}
- C. Next Interview Topic: {next_interview_topic}

TASK: Introducing the Next Interview Topic from the interview plan by asking a transition question.

GUIDELINES:

1. Open-endedness: Always craft open-ended questions ('how', 'what', 'why') that allow detailed and authentic responses without limiting the interviewee to 'yes' or 'no' answers.
2. Natural transition: To make the transition to a new topic feel more natural and less abrupt, you may use elements from the Current Conversation and Previous Conversation Summary to provide context and a bridge from what has been discussed to what will be covered next.
3. Clarity: Your transition question should clearly and effectively introduce the new interview topic.

RESPONSE FORMAT: Your response should use the template below:
 'Question: 'Insert your transition question here''
 Remember to include 'Question:' in your response. Start your response here:

Code A.5: Topic agent prompt by Ingar Haaland and Felix Chopra [6]

Security agent:

You are monitoring a conversation that is part of an in-depth interview. The interviewer asks questions and the interviewee replies. The interview should stay on topic. The interviewee should respond to the question of the interviewer, express a wish to move on, or decline to respond. The interviewee is also allowed to say that they don't know, do not understand the question, or express uncertainty. The interviewee's response might contain spelling and grammar mistakes. Here is the last part of the conversation.

Interviewer: '{last_question}'

Interviewee: '{user_answer}'

That is the end of the conversation. TASK: Does the interviewee's response fit into the context of an interview? Please answer only with 'yes' or 'no'

Code A.6: Security agent prompt by Ingar Haaland and Felix Chopra [6]

Probing agent:

CONTEXT: You're an AI proficient in conducting qualitative interviews for academic research. You conduct a qualitative interview with the goal of learning the interviewee's reasons for not investing in the stock market.

INPUTS:

- A. Previous Conversation Summary: {summary}
- B. Current Interview Topic: {current_topic}
- C. Current Conversation: {current_topic_history}

TASK: Your task is to formulate the next probing question for the Current Conversation. The question should align with the Current Interview Topic, helping us to better understand and systematically explore why the interviewee is not participating in the stock market.

GENERAL GUIDELINES:

1. Open-endedness: Always craft open-ended questions ('how', 'what', 'why') that allow detailed and authentic responses without limiting the interviewee to 'yes' or 'no' answers.
2. Neutrality: Use questions that are unbiased and don't lead the interviewee towards a particular answer. Don't judge or comment on what was said. It's also crucial not to offer any financial advice.
3. Respect: Approach sensitive and personal topics with care. If the interviewee signals discomfort, respect their boundaries and move on.
4. Relevance: Prioritize themes central to the interviewee's stock market non-participation. Don't ask for overly specific examples, details, or experiences that are unlikely to reveal new insights.
5. Focus: Generally, avoid recaps. However, if revisiting earlier points, provide a concise reference for context. Ensure your probing question targets only one theme or aspect.

PROBING GUIDELINES:

1. Depth: Initial responses are often at a 'surface' level (brief, generic, or lacking personal reflection). Follow up on promising themes hinting at depth and alignment with the research objective, exploring the interviewee's reasons, motivations, opinions, and beliefs.
2. Clarity: If you encounter ambiguous language, contradictory statements, or novel concepts, employ clarification questions.
3. Flexibility: Follow the interviewee's lead, but gently redirect if needed. Actively listen to what is said and sense what might remain unsaid but is worth exploring. Explore nuances when they emerge; if responses are repetitive or remain on the surface, pivot to areas not yet covered in depth.

YOUR RESPONSE:

Code A.7: Probing agent prompt by Ingar Haaland and Felix Chopra [6]

A.4. Heuristics for conversational agents

The following will show the adapted set of heuristics by Langevin et al. [15] and an explanation for each heuristic:

Heuristic	Explanation
Visibility of system status	The system should always keep users informed about what is going on, through appropriate feedback within reasonable time, without overwhelming the user.
Match between system and the real world	The system should understand and speak the users' language—with words, phrases and concepts familiar to the user and an appropriate voice—rather than system-oriented terms or confusing terminology. Make information appear in a natural and logical order. Include dialogue elements that create a smooth conversation through openings, mid-conversation guidance, and graceful exits.
User control and freedom	Users often choose system functions by mistake and will need an option to effortlessly leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

Heuristic	Explanation
Consistency and standards	Users should not have to wonder whether different words, options, or actions mean the same thing. Follow platform conventions for the design of visual and interaction elements. Users should also be able to receive consistent responses even if they communicate the same function in multiple ways (and modalities). Within the interaction, the system should have a consistent voice, style of language, and personality.
Error prevention	Even better than good error messages is a careful design of the conversation and interface to reduce the likelihood of a problem from occurring in the first place. Be prepared for pauses, conversation fillers, and interruptions, as well as dialogue failures, deadends or side-tracks. Proactively prevent or eliminate potential error-prone conditions, and check and confirm with users before they commit an action.
Help and guidance	The system should guide the user throughout the dialogue by clarifying system capabilities. Help features should be easy to retrieve and search, focused on the user's task, list concrete steps to be carried out, and not be too large. Make actions and options visible when appropriate.
Flexibility and efficiency of use	Support flexible interactions depending on the use context by providing users with the appropriate (or preferred) input and output modality and hardware. Additionally, provide accelerators, such as command abbreviations, that are unseen by novices but speed up the interactions for experts, to ensure that the system is efficient.

Heuristic	Explanation
Aesthetic, minimalist and engaging design	Dialogues should not contain information which is irrelevant or rarely needed. Provide interactional elements that are necessary to engage the user and fit within the goal of the system. Interfaces should support short interactions and expand on the conversation if the user chooses.
Help users recognize, diagnose and recover from errors	Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.
Context preservation	Maintain context preservation regarding the conversation topic intra-session, and if possible intersession. Allow the user to reference past messages for further interactions to support implicit user expectations of conversations.
Trustworthiness	The system should convey trustworthiness by ensuring privacy of user data, and by being transparent and truthful with the user. The system should not falsely claim to be human.

Table A.1.: Eleven heuristics by Langevin et al. [15] with additional explanations

A.5. Study scenario and Task

Scenario:

Emma Anders, a 35-year-old woman born on August 1, 1990, would like to undergo testing at the Deutsche Krebsgesellschaft (DKG), a designated center of the FBREK cooperation, to assess her risk of a possible hereditary predisposition to breast and/or ovarian cancer.

Emma had a very aggressive form of breast cancer. Fortunately, she was able to treat the breast cancer. For the examination, it is also important to know how many cases of breast and/or ovarian cancer have occurred in her family. Emma did her homework and found out the following:

- She herself had been diagnosed with a specific form of breast cancer. Unfortunately, she can no longer remember the exact name, but she knows that the tumor cells did not express any of the three receptors: estrogen receptor, progesterone receptor, and HER2 receptor.
- Her older sister Astrid was diagnosed with breast cancer in both breasts at the same time at the age of 41.
- Emma's maternal grandmother was diagnosed with breast cancer in one breast at the age of 65.
- Emma knows of one case of ovarian cancer on her father's side of the family.

Emma is very concerned and wants to have this test done. When she arrives at the DKG practice, Emma is allowed to sit down at a computer and fill out the checklist online to assess her possible hereditary risk.

Task:

Your task is now to put yourself in Emma's shoes and fill out the Deutsche Krebsgesellschaft checklist on the computer based on the information she has provided. Make sure to use all the necessary information and answer all the questions. The scenario is available to you at any time.

A.6. Activity diagram

The additional activity diagram (see Figure A.2) visualizes the workflow of the LLM agents and how the user behavior influences the execution of tasks for the LLM agents.

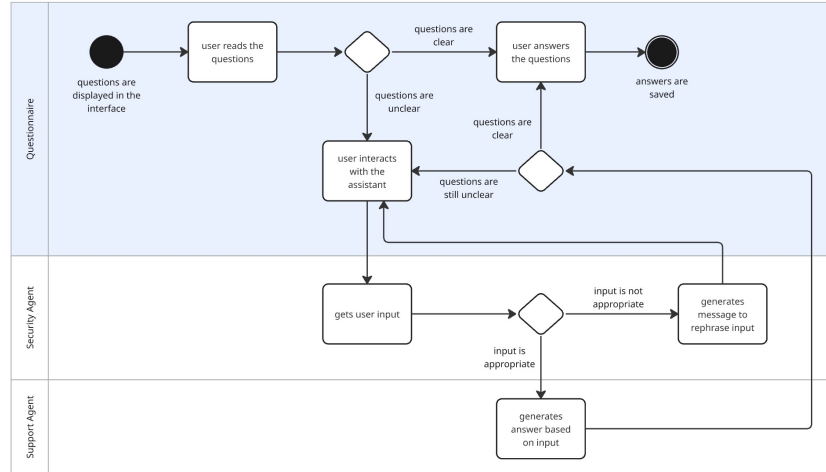


Figure A.2.: UML activity diagram [21] of the LLM agents workflow

A.7. Data Analysis Visualizations

The following two visualizations were complemented to provide additional insights into the data analysis.

Figure A.3 shows the distribution of the eleven heuristics for conversational agents by Langevin et al. [15] across each of the five categories.

Codes	123
<ul style="list-style-type: none"> <ul style="list-style-type: none"> Satisfaction <ul style="list-style-type: none"> Aesthetic, minimalist and engaging design <ul style="list-style-type: none"> Transparency <ul style="list-style-type: none"> Trustworthiness Help users recognize, diagnose and recover from errors Visibility of system status <ul style="list-style-type: none"> Ease of Use <ul style="list-style-type: none"> Error prevention Match between system and the real world <ul style="list-style-type: none"> Guidance <ul style="list-style-type: none"> Context preservation Help and guidance <ul style="list-style-type: none"> Efficiency <ul style="list-style-type: none"> Flexibility and efficiency of use Consistency and standards User control and freedom 	<ul style="list-style-type: none"> 0 15 0 7 10 14 0 12 7 0 9 25 0 12 4 8

Figure A.3.: Categories with assigned heuristics by [15]

Figure A.4 indicates the trends of codes in each transcript and enables the detection of code frequency changes between all transcripts. The three leading categories of transparency, guidance, and efficiency fluctuate from transcript to transcript. Two of the three leading categories are always present in the three categories with the most codes. The visualization thereby confirms the results presented in section 4.3.

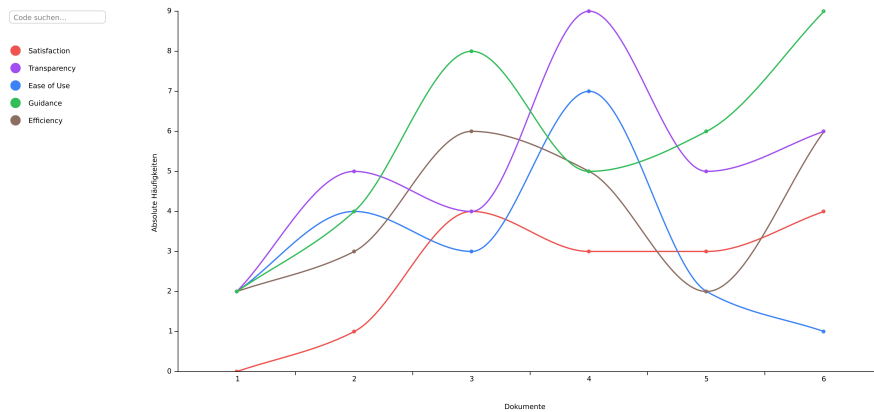


Figure A.4.: Absolute frequency of each category

A.8. Gitlab and AI usage documentation

The source code, BibTeX file consisting of all references used in the thesis, and additional content of the qualitative study are included in the GitLab repository.

<https://git.imp.fu-berlin.de/lamit03/bachelor-thesis>

In this work, artificial intelligence in the form of generative LLMs was used to assist with wording, analyze data, and support literature research. The following table shows which model was used when and for what purpose.

Model	Usage	Purpose
DeepL, DeepL Translator, Version 3.0	Last accessed on 09/17/2025	Used as translation and wording support. I created the content myself and evaluated the result of the support.
OpenAI, ChatGPT 5.0	Accessed on 08/04/2025	Suggested wording and suggestions on how I can include the interview anonymously in my references ¹
OpenAI, ChatGPT 5.0	Accessed on 08/11/2025	Literature search on embellishments of LLM answers ²
OpenAI, ChatGPT 5.0	Accessed on 09/01/2025	Help with translation style and how to summarize my study results section ³
OpenAI, ChatGPT 5.0	Accessed on 03/06/2025	Brief summary of the content and structure of two papers 'Conducting Qualitative Interviews with AI' [6] and 'Conversations at Scale: Robust AI-led Interviews with a Simple Open-Source Platform' [10]. The result served as an assessment and was not included in the thesis ⁴
Google NotebookLM, Gemini 2.5 Flash	Last accessed on 04/10/2025	Used for literature research and to clarify questions about the papers. Unfortunately, the histories are not saved. Therefore, no link can be provided.

Table A.2.: AI support in the thesis

¹<https://chatgpt.com/share/68ca8cd5-8450-8008-8d1e-7d98cf700128>

²<https://chatgpt.com/share/68ca8a58-e934-8008-b7ea-5a61c11ad934>

³<https://chatgpt.com/share/68ca8de1-e330-8008-8657-f6b273a5c854>

⁴<https://chatgpt.com/share/68ca8eb6-0230-8008-b2a4-633bc52efb71>