CHALMERS UNIVERSITY OF TECHNOLOGY | UNIVERSITY OF GOTHENBURG

Bridging Trust and Design of a Multi-Agent LLM-Based HR Chatbot: For the Times They Are A-Changin'

Master's Thesis in Computer science and engineering

Jonatan Axetorn Felix Edholm

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2025

Master's Thesis 2025

Bridging Trust and Design of a Multi-Agent LLM-Based HR Chatbot: For the Times They Are A-Changin'

Jonatan Axetorn Felix Edholm

UNIVERSITY OF GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2025

Bridging Trust and Design of a Multi-Agent LLM-Based HR Chatbot: For the Times They Are A-Changin'
Jonatan Axetorn Felix Edholm

Academic supervisor: Lucas Gren, Department of Computer Science and Engineering
Industry supervisor: Lucas Gren
Examiner in practice: Krishna Ronanki, Department of Computer Science and Engineering
Examiner: Christian Berger, Department of Computer Science and Engineering

Master's Thesis 2025
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Gothenburg, Sweden 2025

Bridging Trust and Design of a Multi-Agent LLM Chatbot for HR: For the Times They Are A-Changin'
Jonatan Axetorn
Felix Edholm
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

Introduction: The integration of Large Language Models (LLMs) into workplace systems presents significant opportunities, particularly in the domain of human resources (HR), where repetitive tasks—such as providing information that employees could retrieve themselves—are common and could potentially be replaced by an LLM-based solution. However, a lack of user trust remains a major barrier to the adoption of LLM-based systems.
Objective: This thesis investigates what trust factors exist in LLM-based systems and how they can be addressed by system design, with a specific focus on a multi-agent HR chatbot.
Method: Using a Design Science Research methodology, the study was conducted in two iterative cycles. Cycle I identified trust factors through literature review and interviews with six employees at a multinational company. It also included a workshop with five AI experts to discuss and validate design choices. Cycle II involved implementing, and evaluating an artifact, a multi-agent chatbot tailored to HR queries.
Findings: Thematic analysis revealed external trust factors: transparency, organisational measures, and external security and internal trust factors: internal security, model differences, risk of bias and reliability, which emerged as the most critical trust factor. The artifact was evaluated through interviews and metrics such as answer relevancy, faithfulness, and robustness, showing consistently strong performance and broad user acceptance.
Conclusion: The multi-agent HR chatbot effectively addressed key trust concerns and was positively received by most interviewees, demonstrating its potential for real-world application. These findings suggest that trust factors can be meaningfully addressed through thoughtful design and should be treated as a core consideration throughout the development process of LLM-based systems.

Keywords: autonomous agents, chatbot, design science research, human resources, HR, large language model, multi-agent architectures, system design, trust, trust factors

Acknowledgements

First and foremost, we would like to express our sincere gratitude to everyone who participated in the interviews and workshop conducted during this thesis.

We would also like to thank Lucas Gren for his support and guidance as our academic and industry supervisor during this project.

Jonatan Axetorn, Felix Edholm
Gothenburg, June 2025

Contents

List of Figures

List of Tables

1

Introduction

The application of Large Language Model (LLM) solutions across various business areas has never been more relevant than it is today. The opportunity to use natural language to address repetitive tasks is promising. Text-based interactions with LLMs are increasingly replacing traditional human-to-human interactions [1].

Despite their potential to improve organisational efficiency, the introduction of artificial intelligence (AI) solutions often encounters reluctance. Factors such as fear of job displacement, distrust of AI's perceived human qualities, and general scepticism contribute to delays in adopting these systems [2]. To overcome these challenges, it is crucial to design LLM-based systems that actively build user trust. Key performance-related factors—such as accuracy and the frequency of hallucinations—have been shown to positively influence this trust [3, 4]. Since these factors are directly shaped by system design, thoughtful design emerges as a vital strategy for fostering trust in LLM-based technologies.

Developing autonomous agent systems based on LLMs, where *agents* refer to AI-based entities that have capabilities such as planning, social interaction, and memory [5], holds significant potential for positively impacting trust factors such as the reliability of the system. Additionally, LLM-based autonomous agent systems have demonstrated significant versatility [6], highlighting their potential to address a wide range of organisational needs. In an ideal scenario, a general agent-based system could meet the needs of employees across different roles within a company. However, creating general-purpose LLM-based solutions has proven to be elusive [7, 8]. A possible alternative is tailoring LLM-based systems to specific purposes.

Furthermore, multi-agent architectures, which leverage the collaborative abilities of multiple LLM agents, have been shown to outperform single-agent systems when handling complex problems [9]. This suggests that designing a multi-agent architecture tailored to a specific role within a company could yield significant performance benefits. Although improved performance alone may not guarantee user trust, it remains an important factor influencing trust [3, 4], as previously stated. Consequently, a multi-agent architecture represents a promising approach for enhancing user trust. A well designed multi-agent LLM-based system could also reduce the need for human-to-human interactions, thereby improving efficiency. This is especially relevant in the context of human resources (HR), where LLM-based systems can automate tasks that traditionally required direct communication with HR staff [2].

This thesis explores the factors that influence trust in LLM-based systems, considering both non-technical elements and those shaped by technical decisions. Through a combination of literature review and interviews, this study identifies key trust factors. Employing a design science research approach, the thesis presents an artifact: a multi-agent LLM-based HR chatbot designed to answer questions related to HR guidelines and employment information, with trust factors integrated into its design. The artifact is then evaluated using both qualitative and quantitative methods.

1.1 Problem description

Most existing studies regarding trust in LLM-based systems do not focus on system design, they focus primarily on user experience. This reveals a critical gap: how can LLM-based systems be designed with trust-building factors in mind?

Both single-agent and multi-agent systems present unique challenges. Single-agent systems often face limitations such as shorter context windows [10] and a higher risk of hallucinations [11]. Multi-agent systems, on the other hand, must address complexities like task allocation and coordination among agents [12]. However, the benefits offered by multi-agent systems—such as improved performance and robustness [13, 8]—tend to outweigh these coordination challenges. Despite this potential, current research primarily focuses on single-agent systems, leaving the potential of multi-agent solutions underexplored.

Another important consideration is whether the LLM-based system is general-purpose or domain-specific. Since different roles have different needs, general solutions often underperform compared to bespoke, domain-specific alternatives. This has been demonstrated in both legal [7] and HR contexts [14], where tailored systems have shown superior results.

Taken together, these findings highlight a key research gap: the design of a multi-agent LLM system tailored to specific roles and organisational needs—while incorporating trust-related factors—has yet to be thoroughly explored.

1.2 Purpose of the study

The purpose of this study is to explore the factors that influence user trust in LLM-based systems and to examine how these factors can be addressed through system design. Specifically, the study focuses on the development of a multi-agent chatbot for HR-related queries, aiming to identify design choices that enhance trust. By doing so, it seeks to bridge the gap between trust considerations and system design in the context of bespoke, domain-specific LLM applications.

1.3 Research questions

- **RQ1:** What are the main trust factors that exist in the usage of an LLM-based system?
- **RQ2:** What potential solutions can be integrated into the system design of an LLM-based HR chatbot to address the relevant trust factors identified in RQ1?
- **RQ3:** To what extent can the relevant trust factors identified in RQ1 be addressed through the design solutions implemented in an LLM-based HR chatbot?

1.4 Significance of the study

The significance of this study lies in its contribution to bridging the gap between system design and user trust in LLM-based applications. It offers practical knowledge for organisations seeking to implement multi-agent LLM systems that foster trust—and thereby encourage user adoption.

Additionally, developing a system built around an AI component, such as an LLM, is part of Software Engineering (SE) for AI. As highlighted by Uchitel et al. [15], this area is highly relevant to the broader software engineering community. This thesis seeks to make a meaningful contribution to SE for AI by addressing the lack of research regarding designing and constructing trust-fostering multi-agent systems.

1.5 Delimitations

This thesis focuses on the development of a chatbot designed to assist employees in querying HR guideline documents and employment-related information. It explicitly excludes other use cases, such as HR personnel interacting with the system or scenarios involving recruitment, onboarding, or employee management. The system is limited to handling informational queries only and does not perform transactional actions, such as applying for leave or managing tasks.

Although security and confidentiality are essential for systems that handle personal or sensitive data, these concerns fall outside the scope of the developed artifact.

The research does not involve a comparison between different large language models. Instead, the chatbot exclusively uses llama3-70b-8192 without any fine-tuning or modification of the underlying model.

The goal of developing the chatbot in this thesis is not to create a fully deployable system for real-world use. Instead, the purpose is to explore how specific design choices influence the trust factors identified. Consequently, no formal requirements elicitation is conducted with stakeholders.

The study is conducted in collaboration with a large multinational company, and all interviews are carried out with employees from within this organisation.

Finally, while user interface design and usability are known to influence trust in AI systems, these aspects are not a focus of this thesis.

1.6 Thesis outline

The thesis begins by presenting key concepts and background information in **Chapter 2**.

Chapter 3 reviews related research relevant to the thesis, including studies on trust in AI and multi-agent architectures.

The research methodology is described in **Chapter 4**, which outlines the overall Design Science Research approach used in the study.

The thesis follows two iterative design cycles. **Chapter 5** details Cycle I, including the methodology for qualitative data collection and the corresponding findings.

Chapter 6 covers Cycle II, beginning with a presentation of the completed artifact, a multi-agent LLM-based HR chatbot, followed by descriptions of the quantitative and qualitative evaluation methods. The chapter concludes with the findings from the artifact evaluation.

In **Chapter 7**, the discussion expands on the findings, explores their implications, and addresses threats to validity. It also outlines potential directions for future research.

Finally, **Chapter 8** provides a conclusion to the thesis.

2

Background

This chapter provides background on the key concepts relevant to this thesis. It begins with an overview of trust and then introduces LLMs more broadly, covering key challenges such as hallucinations and the role of prompt engineering. The chapter then shifts focus to the foundations of retrieval-augmented generation (RAG), orchestration frameworks, guardrails, and the concept of autonomous agents. Finally, it outlines relevant evaluation techniques, with a focus on LLM-as-a-judge and the DeepEval framework used in this study.

2.1 Trust

Trust is a complex and multi-dimensional concept that is challenging to define in a way that applies universally across different contexts. It has been explored in various fields, including psychology [16], economics [17], organisational theory [18], and sociology [19], leading to diverse and sometimes conflicting research [20, 21]. However, in a general sense, trust can be viewed as the relationship between a "trustor" (the one who trusts) and a "trustee" (the one who is trusted) according to Mayer et al. [18].

While researching trust in digital information, Kelton et al. [20] discuss four levels of trust that they have identified in the literature around trust:

- **Individual trust**: A person's inherent trust based on accumulated experiences.
- **Interpersonal trust**: A social connection between a trustor and a trustee.
- **Relational trust**: Trust that develops as an emergent property from the relationship over time.
- **Societal trust**: Trust that exists within a community or society as a whole.

For the purposes of this thesis, **interpersonal trust** is most relevant, as it pertains to the one-way trust relationship between a trustor and a trustee. Importantly, the trustee does not necessarily need to be a human, it could also be a technological system, such as an LLM-based chatbot.

Furthermore, Kelton et al. [20] argues that three key conditions must be met for trust to be relevant in a given situation:

- **Uncertainty**: A lack of information creates uncertainty.
- **Vulnerability**: The trustor is at risk of experiencing a loss if the trust is betrayed.
- **Dependence**: The trustor has a need that the trustee is capable of fulfilling.

In the context of an LLM-based HR chatbot as in this thesis, *uncertainty* arises for the employee (the trustor) due to the fact that they typically turn to the chatbot (the trustee) when they lack specific HR-related information, such as details regarding vacation days or company benefits. Regarding *vulnerability*, there is a potential risk that if the chatbot provides inaccurate information or discloses sensitive data inappropriately, the employee may experience negative consequences, such as making decisions based on faulty or incomplete information. Finally, the employee's *dependence* on the chatbot is evident, as the chatbot holds the necessary information and has the capability to address the employee's questions, thereby fulfilling their informational needs in the HR context.

2.2 Large language models

LLMs are a category of artificial intelligence designed to generate, interpret, and engage with natural human language. These models are trained on vast amounts of textual data, enabling them to learn the complexities of language, including syntax, semantics, and contextual relationships [22]. A significant advancement in this field was the introduction of BERT (Bidirectional Encoder Representations from Transformers), which enabled models to assess the importance of words in a sentence regardless of their position [23]. ChatGPT, which gained widespread public attention in 2022, further advanced these capabilities with a larger and more powerful model. The result is text generation that is both coherent and contextually relevant, based on the input it receives [24]. The practical applications of LLMs are broad, ranging from responding to simple queries to performing complex data analysis.

2.2.1 Hallucinations

LLMs can sometimes produce undesirable outcomes, resulting in outputs that are "bland, incoherent, or caught in repetitive loops." [25] In such cases, the generated content may be nonsensical or unfaithful to the source input. This phenomenon is commonly referred to as hallucinations. Hallucinations present significant concerns regarding the reliability and performance of LLMs for several reasons. One major issue is a reduction in accuracy, as hallucinated responses are, by definition, incorrect. Another concern is related to security, as hallucinations may lead the model to produce or infer sensitive information that it should not access or disclose. Addressing hallucinations remains an ongoing challenge in the field, and researchers are actively developing various techniques to mitigate their occurrence [25].

2.2.2 Prompt engineering

A prompt is an input provided to an LLM that guides the nature of the generated output. Prompts can consist of various types of media, including text, images, audio, or other formats. The process of designing and refining these inputs is referred to as prompt engineering [26]. Prompt engineering has emerged as an effective method for enhancing the performance of LLMs, as it does not require altering the underlying model itself, but instead involves crafting more effective instructions for the AI.

Well-designed prompting techniques have been shown to significantly improve LLM performance, making prompt engineering a critical consideration when developing LLM-based systems [27, 28]. According to OpenAI, some effective strategies for prompt engineering include:

- Including specific details in the query to obtain more relevant answers
- Using delimiters to clearly separate distinct parts of the input
- Specifying the steps required to complete a task
- Providing examples to guide the model's response
- Indicating the desired length of the output

Relatively simple techniques such as these can lead to substantial improvements in the quality and relevance of LLM-generated outputs [29].

2.3 Retrieval-augmented generation

RAG was originally developed by Lewis et al. [30] for natural language processing tasks. This approach enhances LLMs by integrating domain-specific knowledge retrieved from external data sources, thereby mitigating the generation of inaccurate or outdated information. RAG enables text generation to be grounded in relevant, retrieved data rather than relying solely on the model's pre-trained knowledge [31].

The incorporation of external data sources is particularly critical in question-answering systems, where the factual accuracy of responses is a key requirement. As stated, one of the primary challenges in LLM-based systems is the occurrence of hallucinations. Research has demonstrated that RAG significantly reduces the frequency of hallucinations while maintaining the overall performance of the system [32].

At its simplest, the RAG process follows three steps, indexing, retrieval and generation as explained by Gao et al. [31].

1. **Indexing** extracts data from various formats, such as PDF, HTML, and Markdown, standardising it into plain text, and segmenting it into smaller units.

These segments are then encoded into vector representations using an embedding model and stored in a vector database, enabling efficient similarity searches [31].

2. **Retrieval** identifies and retrieves relevant information based on a query. The system encodes the query into a vector and compares it to stored document vectors, selecting the most relevant results. These retrieved segments expand the LLM's knowledge beyond its pre-trained dataset [31].

3. **Generation** synthesises a response using the retrieved context. The LLM processes the query and retrieved document segments to generate a factually grounded and contextually relevant response, integrating both external data and its pre-trained knowledge as needed [31].

2.4 Autonomous agents

Agents have been studied extensively within the AI community long before the emergence of LLMs. Agents are defined as software systems that may exhibit characteristics including: *autonomy*, meaning they can operate without direct human intervention; *social ability*, enabling interaction with other agents; *reactivity*, allowing them to respond to environmental changes; and *pro-activeness*, giving them the ability to take initiative [33]. Additionally, AI agents are implemented using concepts typically associated with humans, such as knowledge, emotion, and intention.

The introduction of LLMs has positively impacted the development of autonomous agents by leveraging natural language capabilities [5]. Modern LLM-based agents integrate advanced features such as personalised profiles, memory retention, external tool usage, and advanced planning [5]. These agents can adopt specialised roles and collaborate with one another, enhancing their collective problem-solving capabilities. This collaboration enables multi-agent systems.

2.5 LLM orchestration

LLM orchestration refers to the process of coordinating multiple LLMs, for instance in the form of agents, to accomplish specific tasks. This involves managing activities such as linking prompts, handling API calls, retrieving data, and maintaining state across interactions. LLM orchestration is often done using an orchestration framework, which provides the structure and tools needed to effectively manage these tasks. These frameworks simplify the development process by offering standardised components and workflows, allowing developers to focus on the higher logic of their applications rather than low level details of coordinating different models [34].

2.5.1 LangChain & LangGraph

LangChain is a framework for developing applications based on LLMs. It provides an interface to interact with LLMs in creating simple linear workflows, while also offering standardised components for AI application functionalities such as model interactions, retrieval mechanisms, and integrations with various data sources [35].

LangGraph is an orchestration framework designed for creating multi-agent systems [36]. While it integrates well with LangChain, and is created by the same company, it can also be used independently. Unlike LangChain's sequential workflow approach, LangGraph enables a conditional workflow using directed graphs. It supports key features such as looping, conditional branching, and state management, allowing agents to dynamically adjust their behaviour based on evolving tasks.

2.6 Guardrails

The non-deterministic, black-box nature of LLMs introduces several risks. Bias in training data can, for example, lead to outputs that reflect societal prejudices. Another challenge is inconsistency—an LLM may produce different answers to the same prompt, which can be particularly problematic in applications requiring reliability, such as question-answering systems. This unpredictability can erode user trust and undermine confidence in LLM-based applications [37, 38].

To address these issues, the concept of **guardrails** has been introduced. Guardrails are mechanisms designed to monitor and filter the inputs and outputs of LLMs, helping to mitigate potential risks [38]. They analyse input prompts and generated responses to determine whether intervention is required to prevent harmful, biased, or incorrect outputs. Guardrails serve as a protective layer within LLM-based systems, reducing the likelihood of exposing sensitive data and limiting the sharing of misleading or inappropriate content [38].

Although guardrails enhance security and reliability, they do not necessarily improve robustness against hostile attacks. Research by Shen et al. [39] indicates that guardrails provide only limited resistance to jailbreak attacks, which are prompt manipulations designed to bypass safeguards and elicit harmful content. Their study found that while guardrails marginally reduce the success rate of such attacks, they do not fully prevent them. This highlights the ongoing need for further advancements in LLM safety mechanisms, even in systems that incorporate guardrails.

2.7 LLM-as-a-judge

Coined by Zheng et al. [40], the term **LLM-as-a-judge** refers to using LLMs as evaluators for tasks that typically require human judgment, such as assessing the quality of chatbot responses in open-ended dialogue. This approach addresses a key limitation of traditional benchmarks, which often fail to capture how well models align with human preferences. By contrast, LLM-based judges can offer a scalable and efficient alternative to human evaluation.

To test the viability of this approach, Zheng et al. [40] developed two benchmarks. Their findings show that the most commonly used LLM at the time, GPT-4, when used as a judge, agrees with human preferences over 80% of the time—comparable to the agreement rate between human annotators themselves.

While promising, the study also highlights several limitations, including susceptibility to biases (e.g. favouring the first-listed response or more verbose answers) and occasional failures in evaluating complex tasks requiring precise reasoning. Despite these issues, the results suggest that, when carefully applied, LLM-as-a-judge can serve as a practical and surprisingly reliable proxy for human evaluation in many settings

2.7.1 DeepEval

DeepEval [41] is an open-source evaluation framework designed to assess the performance of LLM-based systems. By leveraging *LLM-as-a-judge*, DeepEval supports a variety of evaluation tasks across different types of LLM applications, including—but not limited to—RAG systems.

Among the evaluation metrics it offers for RAG scenarios are **faithfulness**, **answer relevancy**, and **contextual relevancy**. Originally introduced in the RAGAS framework by Es et al. [42], these metrics are defined as follows:

- **Faithfulness** measures how accurately the generated answer reflects the retrieved context, aiding in identifying hallucinations.

- **Answer relevancy** evaluates the degree to which the generated response directly addresses the user's question. The metric does not take into account factuality but instead focuses on completeness and focus, penalising responses that are irrelevant, incomplete, or verbose.

- **Contextual relevancy** assesses how relevant the retrieved context used to generate the answer is to the input question. The context should be focused and contain as little irrelevant information as possible.

DeepEval also provides the capability to create custom evaluation metrics through the use of G-Eval [43]. G-Eval is a framework that enables the evaluation of outputs based on user-defined criteria. For instance, it can be employed to assess the correctness of a given output. This is achieved by specifying both the evaluation criteria and the corresponding evaluation steps. An example of criteria and evaluation steps for a custom correctness metric is given below.

- **Criteria:** Determine whether the actual output is factually correct based on the expected output.

- **Evaluation steps:**

- Check whether the facts in `actual output` contradict any facts in `expected output`.
- Heavily penalise omission of detail.
- Vague language, or contradicting *opinions*, are acceptable.

This approach enables the creation of metrics that are not predefined in the DeepEval framework, offering greater versatility when evaluating the outputs of the LLM [43].

3

Related Work

This chapter reviews existing research relevant to the thesis. It begins with an examination of literature on trust in LLM-based systems, followed by a presentation of key challenges in multi-agent systems. The chapter then explores research on collaboration within such systems and concludes with an overview of two approaches aimed at enhancing RAG.

3.1 Trust in LLMs

Trust in AI has been studied long before the rise of LLMs, as evidenced by an empirical research review by Ella and Wooley [44]. However, as LLMs become more widely used, it is important to understand the key factors that influence trust in these systems. Liu et al. [4] and Huang et al. [45] conducted extensive literature reviews and developed taxonomies of trust factors while designing benchmarks to evaluate LLMs. Although their work focuses on assessing the models themselves, and not complete systems incorporating them, the same trust factors remain relevant, as they ultimately relate to how users perceive and trust LLM-generated content.

Liu et al. [4] categorise trust into several key areas, including **reliability**, **safety**, and **explainability & reasoning**. They state that **reliability** refers to the accuracy and consistency of outputs while minimising errors. **Safety** involves protecting sensitive information, while **explainability & reasoning** focuses on how well a system can justify its responses and provide clear explanations.

Huang et al. [45] propose a similar framework with some differences in classification. Their taxonomy includes **truthfulness**, which emphasises providing correct information, **privacy**, which is treated as a separate category rather than a subset of safety, and **transparency**, which relates to how openly a system communicates how it generates its outputs.

Schwartz et al. [46] add to this by identifying key factors that enhance trust in LLM-based systems, including **reliability**, that they define as consistently delivering high-quality, accurate results, **openness**, ensuring transparency regarding system capabilities, limitations, and reliability, **task characteristics**, adapting responses based on task type and complexity, and **trust trajectory**, recognising the importance of first impressions while providing opportunities to rebuild trust through subsequent accurate outputs.

3.2 Challenges with multi-agent LLM-based systems

Han et al. [12] emphasise challenges with multi-agent LLM-based systems that remain inadequately addressed in the literature. The paper summarises these challenges into four categories, as follows:

- Optimising task allocation to leverage agents' unique skills and specialisations.
- Fostering robust reasoning through iterative debates or discussions among a subset of agents to enhance intermediate results.
- Managing complex and layered context information, such as context for overall tasks, single agents, and some common knowledge between agents, while ensuring alignment to the general objective.
- Managing various types of memory that serve for different objectives in coherent to the interactions in multi-agent systems.

3.3 Collaboration in multi-agent systems

There are numerous ways to facilitate collaboration among agents in a multi-agent system. To summarise the various approaches, Tran et al. [47] conducted a survey of LLMs and proposed a framework for LLM-based multi-agent systems. In doing so, they identified three primary categories of multi-agent collaboration, identified in the literature: **collaboration types**, **collaboration strategies**, and **communication structures**.

Tran et al. [47] classify **collaboration types** into three subcategories:

- **Cooperation**, where agents align their efforts towards a shared goal. Some advantages of cooperation include the ability to assign sub-tasks based on individual agent strengths and its relatively straightforward design and execution, provided the goals are clear. However, misaligned goals may lead to inefficiencies, and failures in one agent can significantly impact the entire multi-agent structure. Example scenarios for cooperative collaboration include code generation, decision-making, game environments, question answering, and recommendations.
- **Competition**, where agents prioritise their own objectives, even if they conflict with those of other agents. This type of collaboration encourages agents to enhance their performance and promotes adaptive strategies. However, it is crucial to have a conflict resolution mechanism to ensure competition remains

beneficial to the system as a whole. Example scenarios where competition may be advantageous include debate, game environments, and question answering.

- **Coopetition**, a hybrid of competition and cooperation, in which agents collaborate on some tasks while competing in others. This enables the system to balance trade-offs and reach mutual agreements. However, as an underexplored area, its effectiveness and ideal applications remain uncertain. Tran et al. [47] cite negotiation, such as in policymaking systems, as the primary example scenario for coopetition.

Tran et al. [47] identify three distinct **collaboration strategies** for multi-agent systems:

- **Rule-based**, where predefined rules strictly govern agent interactions. This ensures efficiency, high predictability, consistency, and fairness. However, it also results in low adaptability to uncertainty and scalability challenges for complex tasks. Rule-based strategies are best suited to applications such as question answering, consensus-seeking, navigation, or peer-review processes.

- **Role-based**, where each agent assumes a predefined role and operates on segmented objectives based on its domain knowledge to support the system's overarching goals. This strategy enhances modularity and reusability while leveraging agents' specialised expertise. However, poorly defined roles can lead to rigidity, disputes, or functional deficiencies. Role-based strategies are particularly applicable to simulations of real-world environments with well-defined jobs, such as decision-making or software development.

- **Model-based**, where agents perform probabilistic decision-making based on input (with uncertainty in perception potentially impacting agent actions), environmental factors, and shared goals. This probabilistic approach allows adaptability to dynamic environments and robustness to uncertainties. However, it is complex to implement and computationally expensive. Due to its adaptability, this strategy is well suited to dynamic contexts such as game environments or robotics.

Tran et al. [47] categorise **communication structures** into three main types:

- **Centralised structure**, where each agent connects to a central agent responsible for all collaboration decisions. This structure is easy to design and implement and is efficient for resource allocation. However, its reliance on a single central node creates a single point of failure, making it less resilient to disruptions. According to Tran et al. [47], centralised structures are suitable for question answering and decision-making scenarios.

- **Decentralised structure**, where control and decision-making are distributed among agents that operate on local information. This structure enhances resilience, as the system can continue functioning even if individual agents fail, and it is highly scalable. However, it may suffer from inefficient resource allocation and significant communication overhead. A decentralised structure is applicable to decision-making, question answering, reasoning, and code generation.

- **Hierarchical structure**, where agents are organised in layers, with communication primarily occurring between adjacent layers. Each layer has distinct functions, roles, and levels of authority. This structure reduces bottlenecks and facilitates task distribution among layers. However, it is highly complex, leading to increased latency and implementation challenges. Hierarchical structures are used in scenarios such as code generation, question answering, and reasoning.

Additionally, Tran et al. [47] discuss **coordination and orchestration architectures**, which extend beyond individual collaboration channels to manage the relationships and interactions between multiple channels. These architectures define how collaboration channels are created, ordered, and characterised. Tran et al. [47] identify two major types:

- **Static architectures**, which rely on predefined rules and domain expertise to establish collaboration channels. By leveraging prior knowledge, these architectures ensure interactions adhere to domain-specific requirements while improving overall system efficiency and maintaining consistent task execution. However, their dependence on accurate domain knowledge and their fixed nature result in limited scalability and flexibility.

- **Dynamic architectures**, which adapt to changing environments and task requirements by employing management agents or other adaptive mechanisms to assign roles and define collaboration channels in real time. While suitable for complex and evolving tasks, dynamic architectures require higher resource allocation due to real-time adjustments and present a greater risk of failure due to their fluid nature.

3.4 Multi-agent retrieval-augmented generation filtering

As previously stated, RAG has become a key technique for improving the accuracy and reliability of LLM-generated responses by incorporating external knowledge retrieval. One approach, Self-RAG, introduced by Asai et al. [48], enhances factual accuracy by allowing the model to decide when to retrieve additional information and critically assess its own outputs. This method helps improve citation accuracy and reduces the inclusion of irrelevant or misleading information.

A more recent development is MAIN-RAG, proposed by Chang et al. [49], which takes a multi-agent approach to further refine the retrieval process. Their paper shows that it outperforms Self-RAG across a number of datasets. MAIN-RAG is a training-free framework and introduces three specialised agents: a **Predictor**, which retrieves documents and generates an initial answer based on each document. The predictor then sends the documents to the **Judge** agent, which evaluates if the documents provide relevant information to the query and answer and scores and orders them accordingly. If a document is deemed to be irrelevant, it is filtered out at this step. Finally, the documents are sent to the **Final-Predictor** agent, which generates the final response based on the sources provided by the judge agent.

4

Method

This thesis was conducted as design science research (DSR) mainly following the methodology of Wieringa [50] and the guidelines for applying DSR in the context of a master thesis presented by Knauss [51]. DSR focuses on the creation of a design artifact to solve a concrete problem while also gathering data about knowledge questions. In this thesis, the artifact is a multi-agent LLM-based HR chatbot with two components, one for answering questions regarding employment data, and one for answering question based on HR guideline documents. The chatbot is designed with the goal of understanding what design choices can address trust factors and in turn foster trust for such a system.

4.1 Design science research

Wieringa [50] describes design science research as an iterative problem-solving methodology structured around the "regulative cycle", illustrated in figure 4.1, which comprises five phases: *problem investigation*, *solution design*, *design validation*, *solution implementation*, and *implementation evaluation*. Knauss [51] groups these phases into three broader categories:

- **Problem:** Includes *problem investigation*, where the research problem is explored and analysed.
- **Solution:** Covers *solution design* and *design validation*, focusing on developing and validating possible solutions.
- **Evaluation:** Encompasses *evaluation*, where the effectiveness and usability of the proposed solution are assessed.

Additionally, the *implementation* phase represents the artifact. Throughout the iterative cycles, the artifact undergoes incremental work—continuously evolving based on the insights gathered during the other phases. This thesis was conducted through two cycles, described further in chapter 5 and chapter 6 respectively.

In alignment with Knauss's guideline 3 [51], the research questions in this thesis are formulated to correspond to the three main categories: RQ1 addresses problem

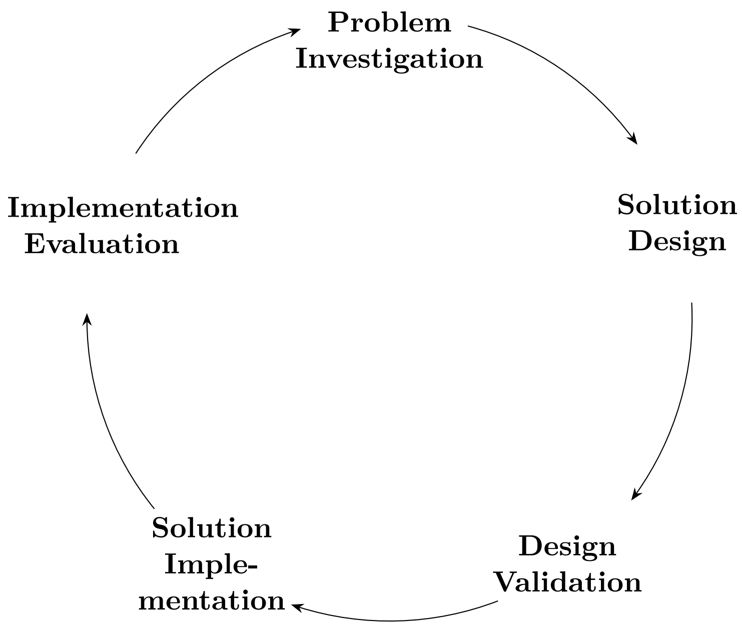

Figure 4.1: The regulative cycle of design science research.

understanding, RQ2 focuses on potential solutions, and RQ3 is connected to the evaluation of the proposed solution.

4.1.1 Problem investigation

The purpose of the problem investigation phase is to gather information in order to understand the given problem, as well as describe it and explain it. Wieringa [50] presents four non-exclusive reasons for investigating the problem:

- **Problem-driven investigation**, where there is a concrete problem that needs to be understood before trying to solve it.
- **Goal-driven investigation**, where the investigation is motivated not necessarily by a problem but by some ambition to achieve change.
- **Solution-driven investigation**, where a technology's potential to improve or solve a problem is analysed.
- **Impact-driven investigation**, also called evaluation research, where the focus is on evaluating the impact of past actions instead of preparing for future solutions.

In this thesis, two main problem investigation approaches were employed. Problem-driven investigation was primarily used to address RQ1, which focuses on understanding the issue of trust factors in an LLM-based system. In contrast, solution-driven investigation was mainly applied to RQ2, which explores potential solutions to adress the trust factors identified in RQ1.

4.1.2 Solution design

The solution design phase, as described by Wieringa [50], involves formulating possible solutions to the identified problem. These designs, which he refers to as *solution suggestions*, serve as just that, *suggestions*, rather than definitive answers, as they have not yet been validated or implemented. Solution designs can take various forms, including natural language descriptions, sketches, blueprints, mathematical models, or prototypes.

Wieringa highlights that solution design is not a fixed plan from the beginning. Rather, it is a process that involves uncertainty, with the proposed solution developing further as it is evaluated and tested. A solution suggestion does not describe an existing reality, explain past events, or predict future outcomes. Instead, it outlines a possible course of action that helps stakeholders move from uncertainty ("we are uncertain about what to do") to confidence ("we are sufficiently certain about what to do").

4.1.3 Design validation

During the design validation phase, the design is investigated with the purpose to understand if it indeed would bring stakeholders closer to their goals. Wieringa [50] states that there are three important knowledge questions that need to be answered in this phase:

- **Internal validity:** If the design were to be implemented, would it satisfy the criteria identified in the problem investigation?
- **Trade-offs:** How do different designs compare to each other if implemented in this context?
- **External validity:** Does the design, if implemented in *another context,* satisfy the criteria?

The solution design and design validation in this thesis were primarily conducted through literature review, complemented by a two-day workshop with AI experts from the collaborating company. The setup and findings of this workshop are presented in detail in chapter 6.

4.1.4 Implementation

As stated by Wieringa [50], the implementation phase in DSR depends on the nature of the designed solution. If the goal of the research was to develop a method, framework, or process to address a practical problem, then implementation involves executing this process in a real-world setting. However, if the research focused on testing the viability of a proposed solution, implementation consists of conducting the planned evaluations or experiments.

The final implementation in this thesis resulted in a multi-agent LLM-based HR chatbot, the final artifact. This artifact is presented in detail in section 6.1.

4.1.5 Evaluation

As outlined by Hevner et al. [52], evaluation constitutes a fundamental component of the research process, ensuring the effective integration of the artifact within the technical infrastructure. A rigorous evaluation requires the establishment of appropriate metrics to accurately assess the quality of the implementation. As Hevner et al. [52] emphasises, evaluation plays a critical role in the iterative research process, facilitating the identification of deficiencies and informing necessary improvements for subsequent development cycles.

Knauss [51] recommends adhering to Hevner et al.'s [52] established evaluation methodologies to align this phase with RQ3. These methodologies include observational, analytical, experimental, testing, and descriptive evaluations.

The final artifact developed in this thesis was evaluated using both quantitative and qualitative methods. The quantitative evaluation involved an experimental simulation, where the artifact was executed with artificial data, described in detail in section 6.2. In addition, a qualitative evaluation was conducted through interviews with potential users, as outlined in section 6.3.

4.2 Overview of cycles

The two completed iterations of the regulative cycle in this thesis are visualised in figure 4.2 and detailed in the coming chapters. Cycle I primarily focused on problem investigation and preliminary design activities, including a series of interviews and a collaborative workshop with domain experts at the partner company. Cycle II primarily focused on finalising the artifact and conducting both quantitative and qualitative evaluations.

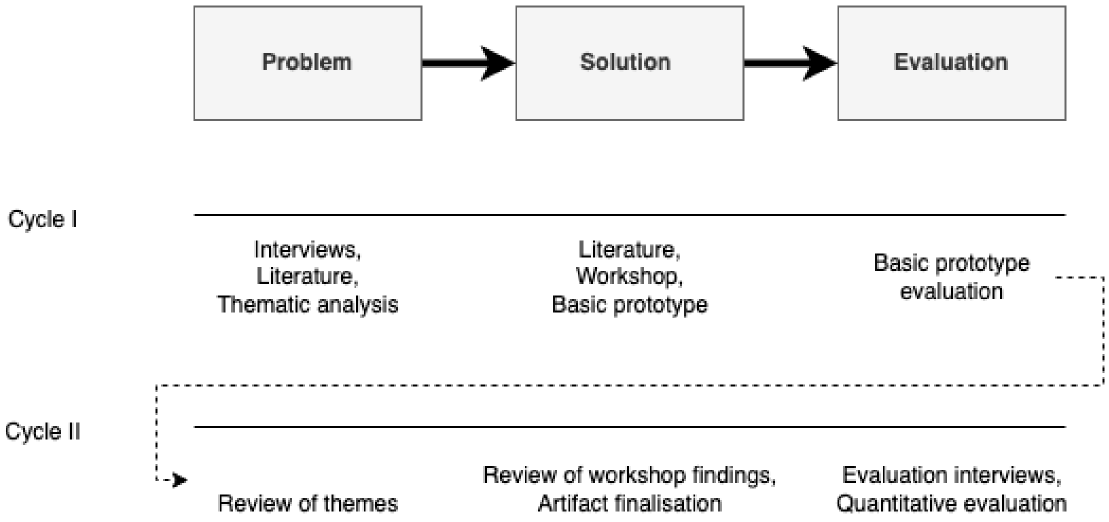

Figure 4.2: Activites performed during the two cycles in this thesis.

Table 4.1 presents an overview of the qualitative data collection activities, detailing the number of participants and the total time spent on each activity. Table 4.2 summarises the number of evaluation runs conducted for each metric during the quantitative data collection phase in Cycle II, as described in section 6.2.

The next two chapters describe each research cycle in detail. The chapter on Cycle I begins by outlining the methodology used during this cycle, followed by the key findings. In contrast, the chapter on Cycle II opens with a presentation of the final artifact, which serves as a reference point for the evaluation approach and results that follow.

The structure of presenting the research cycles sequentially—detailing the method and findings of Cycle I followed by those of Cycle II—was chosen to enhance readability and comprehension. Since the research questions build upon one another, understanding the findings for RQ1 is essential for interpreting the final artifact, methodology and results of Cycle II. As such, the findings for RQ1 are presented in full within the Cycle I chapter, even though the findings were fully finalised during Cycle II.

Activity	**No. of participants**	**No. of hours**
Problem investigation interviews	6	6
Workshop	5	16
Follow-up evaluation interviews	5	3.75
Total	**14**	**25.75**

Table 4.1: Participant counts for qualitative data collection activities.

Metric	**No. of evaluation runs**
Answer relevancy	360
Faithfulness	360
Contextual relevancy	360
Robustness (answer relevancy)	500
Robustness (faithfulness)	500
Robustness (contextual relevancy)	500
Correctness	280
Total	**2860**

Table 4.2: Number of evaluation runs per quantitative metric.

5

Cycle I

The first cycle focused primarily on understanding the problem space regarding trust in LLM-based systems and exploring potential solutions. As such, it placed greater emphasis on the first three phases of the regulative cycle, **problem investigation**, **solution design**, and **design validation**, aligning closely with **RQ1** and **RQ2**.

The problem investigation followed a problem-driven approach, aiming to identify trust factors associated with an LLM-based HR chatbot. To achieve this, interviews were conducted with potential users, followed by a thematic analysis of the results to extract key insights into factors that impact their trust. The thematic analysis was done separately by each author and then merged to mitigate bias.

Additionally, the investigation extended to exploring design choices and components of multi-agent chatbot systems, primarily through a review of existing research. These potential design solutions were then explored and validated through participation in a two-day workshop with experts in building LLM-based systems. The workshop facilitated discussions on various design strategies, and feedback from experts during the workshop served as an initial form of validation for these design choices.

Although the primary focus of this cycle was on problem investigation and solution exploration, a preliminary implementation was undertaken to test basic functionality. The purpose of this early implementation was to explore high-level considerations, such as which frameworks to use, how an agentic RAG system functions, and which LLMs are compatible and can be effectively utilised. The evaluation of this early prototype was basic and relied on human judgment by the authors, supported by insights from the literature review on what appears to be most suitable for an HR chatbot in practice.

This chapter outlines the methodology and findings from cycle I of the thesis. It begins by presenting the qualitative data collection methods, including the approach used for the problem investigation interviews, the subsequent thematic analysis, and the setup of the expert workshop. The second part of the chapter focuses on the findings from this cycle, starting with the trust factors identified through the interviews and concluding with key insights from the workshop discussions.

5.1 Method - Qualitative data collection

To gain a deeper understanding of trust in LLM-based systems and to explore different solution designs, cycle I employed two qualitative data collection methods. These included interviews conducted as part of the problem investigation as well as a workshop with experts focused on the design and implementation of LLM-based systems.

5.1.1 Interviews

To attain qualitative data about trust in our initial problem investigation, six interviews were conducted with employees at the partner company. These interviews were designed based on the guidelines provided by McNamara [53] and Patton [54]. The format used was the *standardised, open-ended interview*, where all interviewees were asked the same open-ended questions and could respond freely in their own words. In some instances, follow-up questions were posed to encourage further elaboration from the interviewees.

This interview format was chosen because, as McNamara states, it *"facilitates faster interviews that can be more easily analysed and compared"* [53]. As noted, the questions were constructed in accordance with McNamara's guidelines [53], which emphasise important principles such as question neutrality, the use of open-ended wording, and smooth transitions between major topics.

For the sampling method, *snowball, also called chain sampling,* [54] was used. In this method, an industry supervisor with extensive knowledge about who would be information-rich key informants for the interviews was tasked with reaching out and finding such participants. The interviewees had varying levels of AI knowledge and roles within the company, ensuring diverse perspectives.

5.1.1.1 Problem investigation interview setup

The interviews were conducted with each of the six employees as part of the problem investigation. The questions touched on 4 overarching subjects:

- Background and demographic information,
- Knowledge and experience with AI & LLMs,
- Attitudes, opinions, and trust in AI
- HR system specific questions.

All interviews were conducted remotely and lasted approximately 60 minutes. With the participants' consent, interviews were recorded and automatically transcribed. The transcripts were then reviewed and corrected in phase one of the thematic analysis. The interview guide used for this round of interviews can be found in appendix A.1.

5.1.1.2 Thematic analysis

To analyse the interviews, we employed thematic analysis, following the guidelines established by Braun and Clarke [55]. Thematic analysis is defined as "a method for identifying, analysing and reporting patterns (themes) within data." Braun and Clarke [55] outline a structured approach that consists of five key phases, each of which is detailed below. Importantly, they note that thematic analysis is not strictly a linear process but rather a recursive one, where movement between phases is necessary to refine and develop themes.

1. **Familiarising yourself with your data:**
   This phase involves immersing oneself in the collected data through repeated *active* reading. Initial notes and potential codes should be marked for later phases. Transcription plays a key role in deepening familiarity, and if transcription has been conducted by others or through automated tools, additional time should be dedicated to engaging with the material.

2. **Generating initial codes:**
   Following familiarisation, the data should be systematically coded to identify meaningful features of interest. Equal attention must be given to all data items, including those that challenge dominant narratives. Coding should be as comprehensive as possible within the available timeframe, preserving surrounding context and allowing for multiple codes per extract.

3. **Searching for themes:**
   This phase focuses on organising codes into broader themes by clustering related codes and exploring their interconnections. Visual tools such as tables or mind maps are recommended to facilitate the conceptual organisation. At this stage, all codes and potential themes should be retained for further consideration.

4. **Reviewing themes:**
   Candidate themes are reviewed and refined to ensure they accurately reflect patterns in the data. This process involves two levels: first, evaluating coherence within each theme's coded extracts; second, assessing the thematic structure in relation to the full dataset. Additional coding may be required if new relevant data is identified. By the end of this phase, key themes and their relationships should be clearly defined.

5. **Defining and naming themes:**
   With a satisfactory thematic map in place, themes are further refined and clearly defined. Each theme's core meaning should be articulated, ensuring alignment with the data and avoiding excessive breadth or overlap. Subthemes may be identified to capture nested or hierarchical relationships within the data.

5.1.2 Workshop

The data collection process during cycle I also included a workshop, conducted at the collaboration company over two days. The workshop focused on discussions around the design and implementation of LLM-based systems within three organisational domains: Sales, HR, and Cybersecurity. Participants in the workshop consisted of AI experts in each respective area with a total of seven participants including the authors.

The workshop mainly explored two use-cases for LLM-based solutions. The first being an LLM-based solution for managing large volumes of internal documentation, and the second involving an LLM-based chatbot designed to respond to employee queries, specifically those related to employment matters, HR guidelines, and organisational policies.

The workshop served both as a means of collecting empirical data on how domain experts approach the practical implementation of AI solutions within an organisational setting and as an evaluation of the feasibility of previously studied approaches in a real-world context. Furthermore, the workshop provided insights into the trade-offs associated with different design strategies and explored how a scalable solution could be developed to support future applications across other areas of the organisation.

5.2 Findings - Cycle I

This section presents the findings from the problem investigation interviews regarding RQ1 and the findings from the workshop related to RQ2. It begins with the results of the thematic analysis, which answer RQ1 by identifying key trust factors in LLM-based systems. It then focuses on RQ2 by presenting insights from the workshop, which informed the design of the artifact in cycle II.

5.2.1 Trust factors in LLM-based systems (RQ1)

After an extensive thematic analysis of the conducted interviews, we identified five main themes. Two of these themes are directly related to trust factors and thus address RQ1. The other three themes, however, are more concerned with attitudes, thoughts, and concerns surrounding LLM-based systems, rather than directly addressing trust factors. These three themes—**AI as a helping hand**, **Concerns - Human interactions could be replaced**, and **Critical thinking - Output should be challenged and revised**—are considered outside the scope of this thesis, as they do not directly correspond to trust factors in an LLM-based system.

In the following two sections, we will describe in detail the remaining two main themes that were identified. These were **External trust factors — Trust impacted by non-technical forces** and **Internal trust factors — Trust impacted by technical details**.

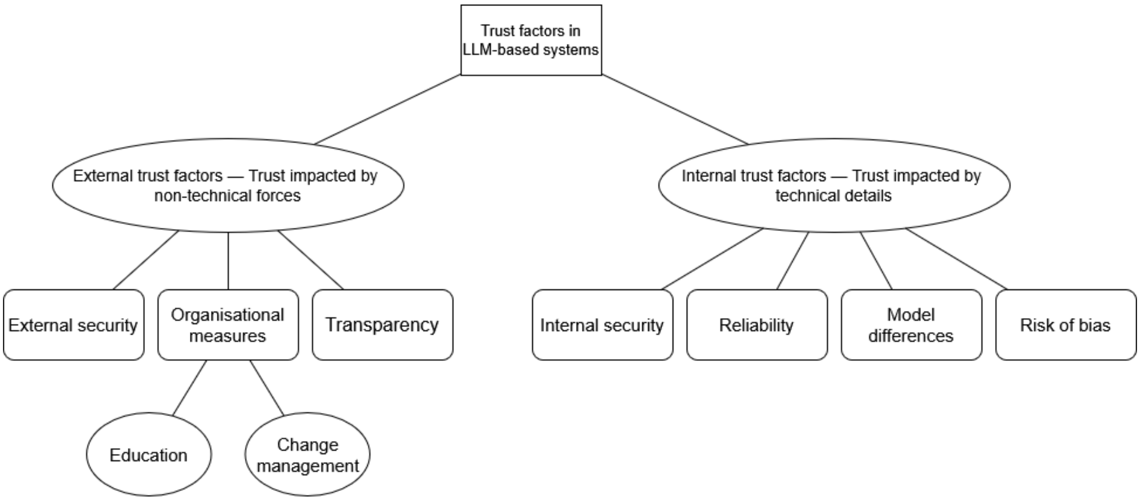

Figure 5.1: Identified trust factors in LLM-based systems.

5.2.1.1 External trust factors — Trust impacted by non-technical forces

This theme addresses factors that are not directly influenced by the system design, but instead areas outside the actual artifact. As seen in figure 5.1, these factors include **transparency**, **organisational measures** such as **education** on LLMs and chatbots as well as **change management**, and **external security** referring to security concerns that cannot be addressed within the system design.

Transparency

Transparency encompasses both transparency in the development process and transparency of the system's limitations.

One interviewee stated that transparency in how an LLM-based system is built provides users with insight into its foundations, which in turn increases trust. Further, they said that knowing the process about what model is being used, what approach was used in the development and what data is being used would make them feel more trust towards the system. Another interviewee also stressed that it is important to understand how the data you provide to an LLM-based system is stored and used.

Transparency about an LLM-based system's limitations can also help users set realistic expectations, something brought up by three out of the six interviewees. If something is presented as always correct but proves otherwise, it may lose users' trust. However, being upfront about potential inaccuracies can foster understanding and make users more forgiving. As one interviewee shared:

> "That's the thing that I'm saying, is that I don't trust 100%, but I still think that we can implement things acknowledging that probably 100% is impossible, but we can be close to 100% on the output. So we can gain trust and confidence of all the people that are gonna use it."

(Interviewee 3)

Change management

Another factor that emerged from the data was the importance of actual system usage in building trust, as well as the company's role in encouraging that usage. Four out of the six interviewees stated that using the system will allow it to prove itself, eliminating possible preconceptions. One interviewee reflected on their initial scepticism:

> "Can you trust what it's telling you? Will it make mistakes? And I had those preconceptions, like a year and a half ago, when I first was like, you know, how is this possible to use this? But as I've used [an LLM-based tool] and seen it evolve, seen it improve, seen actually how it can benefit my work, I really see the opportunities."
>
> *(Interviewee 5)*

This suggests that increased use of the system, along with witnessing its evolution, can serve as a catalyst for trust development. To further encourage this usage, organisations may need to actively guide employees toward adopting LLM-based tools such as an HR chatbot. This would give these tools the opportunity to demonstrate their value. Five out of the six interviewees highlighted the importance of effective change management in increasing system adoption and engagement. This included both changing the behaviour of the users of the system, as well as enabling easy access to the tools.

When discussing the integration of an HR chatbot, two interviewees stated that each employee currently has an assigned local HR business partner who is readily accessible. As a result, there would be little motivation to use such a chatbot, even if it provides equivalent support. Both interviewees stated that making direct contact with HR operations less convenient could promote the use of the chatbot. Crucially, they emphasised that the chatbot must offer clear and tangible value to employees. If it is to be adopted, it should convincingly demonstrate that it is a more efficient or beneficial alternative to traditional HR contact methods.

Education

To enable this change management, education emerges as a key organisational tool to help employees understand the value of the system, thereby increasing trust and fostering adoption. Education was discussed by all interviewees with one interviewee stating that the organisation needs to make it possible for employees to get an introduction into how to use a chatbot if it is implemented. Another interviewee also discussed this benefit of education in improving their ability to optimise the use of LLM-based chatbots and getting more value out of them:

> "Yeah, I really think that I would benefit from educating myself more in optimising the usage [of LLM-based chatbots]."

(Interviewee 4)

External security

One of the factors that was brought up by four out of the six interviewees was the potential impact of the system's origins or the underlying models on trust. If the company behind the system or LLM is deemed untrustworthy, users' trust in the system itself can be compromised.

Further, five out of six interviewees expressed caution regarding the information they input into LLM-based systems, especially when interacting with systems not hosted by their organisation. One interviewee noted:

> "For instance, obviously I played around with Deepseek, and I knew that using Deepseek was basically sending information to China. I don't think China is way, way worse than the US, but still, it's like, OK, I'm sending to another country. That's why I use this Deepseek just to play around. It was just basically doing Q&A for bullshit stuff. So nothing [sensitive]—that's why I took that into consideration.
>
> The moment I'm able to actually get, get Deepseek working on a—let's say—open-source environment, or, let's say, download and install all the way into a server, I would probably use it in a different way, that's for sure."
>
> *(Interviewee 3)*

Finally, regarding LLM-based systems used and approved within the company, trust can be handed over to the IT department and their expertise with one interviewee saying

> "I have no real limitations as long as I know that these tools have been embedded by corporate IT from a security standpoint"
>
> *(Interviewee 1)*

In summary, external trust factors centre on non-technical influences. Participants emphasised the value of transparent communication about system development and limitations, as this sets realistic expectations and builds trust. Organisational efforts like education and guided adoption were seen as essential for encouraging usage and overcoming skepticism. Trust was also shaped by concerns about where data is sent and who controls the underlying technology—highlighting that trust is not just built on what the system does, but also on who is behind it and how it is introduced.

5.2.1.2 Internal trust factors — Trust impacted by technical details

The second major theme identified was internal trust factors. These factors come from the actual LLM-based systems themselves, how they perform and behave.

Thus, these are factors that may be impacted by system design. As illustrated in figure 5.1, the internal trust factors identified were **internal security**, **risk of bias**, **model differences**, and **reliability**.

Internal security

Internal security relates to the importance of protecting sensitive employee data, particularly in the context of an HR chatbot. A concern brought up by two out of six interviewees was the potential for unauthorised access, for example, if the system could be exploited to retrieve other employees' information. Another concern was the system's ability to comply with internal IT and legal frameworks. As one interviewee noted:

> "So we have to be compliant with all the rules that exist. We have IT processes and legal processes that must be [followed]."
>
> *(Interviewee 4)*

Risk of bias

Another sub-theme that emerged was the risk of bias in LLMs and LLM-based tools. One participant, for example, expressed concern about the potential impact on diversity when such tools are used in recruitment processes. They noted that the system might favor candidates with similar educational backgrounds or professional experiences—such as coming from the same types of companies—which could unintentionally limit diversity. This was viewed as a risk that could lead to less varied and inclusive hiring outcomes.

Another participant expanded on biases, discussing how they may be embedded in the training data and reflected in outputs:

> "And then also, like I mentioned before, the ethics around the information that people use from it, and how these models have been built, and who has built them, and the bias. And, you know, is information it gives representative of the wider population? ... And that does really concern me from a kind of diversity, equality point of view, because a lot of work has been done previously to promote different types of voices on different topics."
>
> *(Interviewee 5)*

Model differences

Five out of six participants also commented on perceived differences between various LLMs and LLM-based tools, which we have categorised as the theme model differences. Comparisons were for instance made between internal company tools and more widely available tools such as ChatGPT, with one interviewee stating that:

> "Tools that we have in [Company] as of today—I mean, they are good. But I think they are not that good, obviously, like ChatGPT."
>
> *(Interviewee 3)*

Reliability

The factor of reliability refers to the system's ability to consistently produce accurate, high-quality responses. All participants emphasised reliability as a central trust factor when interacting with LLM-based systems. They highlighted that if the system frequently produces incorrect answers, trust is quickly diminished. When discussing what might deter them from using an LLM-based system, one interviewee stated:

> "No, but recurring inaccuracies, I think. That would have made it so [I felt] 'But no, it's not worth the time. I'll have to look it up myself' or something like that. So yes, repeated inaccuracies would have caused my trust to decrease."
>
> *(Interviewee 4)*

Another participant echoed this sentiment, describing how even a single mistake in practical information could lead to reduced usage:

> "I think it's the reliability of the information [that is important for usage]. Of course, if I use the chatbot and ask the chatbot how many remaining vacation days I have, I get an answer, and then the answer proves to be the wrong one. I might not use it again easily."
>
> *(Interviewee 1)*

These responses underline the importance of providing accurate responses from the outset. If a system fails to meet expectations early on, trust may be damaged and difficult to rebuild later. This concern was also reflected in discussions about implementation strategy. One interviewee suggested that a gradual rollout of an HR chatbot could help identify and resolve early issues before exposing it to a wider audience. Thus, avoiding the risk of discouraging users who may perceive the tool as unreliable.

> "So it depends a bit [on] the purpose of it, but you can collect a lot of feedback by rolling out something real that you then improve as you go along, and then you roll it out a little wider, like, until you have something that really doesn't have a lot of teething problems and that can provide value. Because if it doesn't provide value, people won't use it."
>
> *(Interviewee 6)*

Another aspect of reliability that emerged was the importance of source citation in LLM-based systems. Two participants expressed greater trust in systems that provide sources for their outputs, as it allows users to verify the information and better understand where it comes from. When asked about their level of trust in LLM-based chatbots, one participant responded:

> "I think it would be probably if—if I have the—the—the sources mentioned, like in Copilot, I would say probably 8 or 9 out of 10."
>
> *(Interviewee 1)*

And when asked how their trust would be affected if sources were not provided, the same participant explained that they would feel the need to challenge the output more actively—by comparing it across different chatbots and questioning the origin of the information.

Summarising, the internal trust factors identified by participants reveal how trust in LLM-based tools is closely tied to the system's technical performance. Issues such as hallucinations, lack of source transparency, and data privacy risks emerged as key concerns. Trust was also found to be fragile—easily lost through early errors—and difficult to rebuild, underscoring the need for high initial system performance and thoughtful implementation.

5.2.2 Findings from workshop (RQ2)

This section outlines key findings from the workshop regarding design choices in developing an agent-based HR chatbot. A primary concern raised in the workshop was the importance of **reliability**, which is a quality also emphasised in the interview findings around trust factors. Consequently, many of the design discussions revolved around strategies for improving the reliability of the chatbot.

A core principle agreed upon was that the chatbot should avoid providing incorrect answers. If the system cannot provide a sufficiently accurate or complete response, it should explicitly state this to the user. The consensus was clear: **it is better to give no answer than an incorrect one**. This approach supports both the reliability and transparency of the system.

The HR chatbot use case was generally seen as relatively simple in nature. Its primary function is to retrieve relevant information from HR documentation and systems, and present it in response to user queries. Unlike systems that require complex reasoning or computation, this task primarily involves information retrieval and summarisation. Accordingly, one participant advised against over-engineering the system architecture, saying *"Don't over-engineer the agent structure for a simpler use case"*.

To enable effective document retrieval, discussions revealed that a RAG approach is the most suitable. However, rather than using a basic RAG pipeline with a single LLM retrieving and generating responses, an **enhanced RAG** architecture was proposed. This would involve the inclusion of additional agents to improve answer quality and reliability. Specifically, a **circular workflow** was suggested, featuring a “checker agent” responsible for evaluating the quality of the generated answer in relation to the user query. If the answer is deemed insufficient, the system should loop back to revise and improve it based on this feedback.

Another important factor related to document retrieval was the format of the documents themselves. Discussions emphasised that, although company documents are often available as PDFs, LLMs perform more effectively when the files are provided in Markdown format instead.

Internal security was another important topic in the discussion. Echoing concerns raised in interviews, the inherent sensitivity of HR data was acknowledged as a potential risk. One mitigation strategy discussed was the integration of **guardrails** to limit inappropriate or insecure outputs. However, it was concluded that **during the early stages of development and testing, guardrails are not essential**. This is due to the domain-specific nature of security requirements, which vary significantly across companies, departments, and jurisdictions. As such, defining meaningful security constraints requires detailed context that is often unavailable during early development. Thus, the primary focus at this stage should be on the performance of the system.

6

Cycle II

Building on the insights from the first cycle, the second cycle focused primarily on solution implementation and implementation evaluation. This phase therefore placed greater emphasis on the **implementation**, resulting in the final artifact, and **evaluation** stages of the regulative cycle, aligning closely with **RQ2** and **RQ3**.

The artifact was further developed based on the design suggestions identified during the first cycle. To streamline the development process, the system was implemented using the LLM orchestration framework LangGraph. Additional design considerations—such as defining the roles of the agents, crafting the prompts, and distinguishing between policy-related and employment-related questions were also addressed.

To evaluate the artifact quantitatively, data was collected using the DeepEval framework in an experimental simulation [56]. Five metrics were employed: faithfulness, answer relevancy, contextual relevancy, robustness, and a custom G-Eval metric termed correctness. The results of this evaluation were analysed to determine how the artifact performed. In addition, five evaluation interviews were conducted during this iteration to collect qualitative data. These interviews aimed to assess how well the artifact addressed trust factors identified in the first iteration.

To provide context for the remainder of the chapter, it begins by presenting the design of the final artifact. This is followed by a detailed description of the methods used for both quantitative and qualitative data collection. The quantitative section covers the evaluation metrics, the use of dummy data, and the setup of test runs, while the qualitative section outlines the evaluation interview approach. Finally, the chapter presents the findings from this cycle, including insights from the evaluation interviews and the results of the quantitative analysis.

6.1 The artifact - final solution candidate (RQ2)

This chapter will present the final artifact designed in the project in the form of an HR chatbot capable of answering question based either on HR guideline documents or specific employment data. First, a brief overview of the artifact is presented to give and understanding of how the chatbot works. Following this, the two components of the chatbot, the *employment component* and the *guidelines component* will be described in more detail, including the role of each agent within the components.

6.1.1 Overview

The HR chatbot is implemented as a Python application executed in the terminal. The chatbot is composed of two multi-agent based components, the *employment component* and the *guidelines component*. Within each component, the flow used to answer a given question includes a set of agents, orchestrated through the framework LangGraph, each with a distinct role and responsibility.

When starting the chatbot, the user is asked to provide a question. After providing the question, the chatbot asks if the question is about general HR policies or employment data, as show in figure 6.1. The answer to this question decides which component will be used to answer the given question.

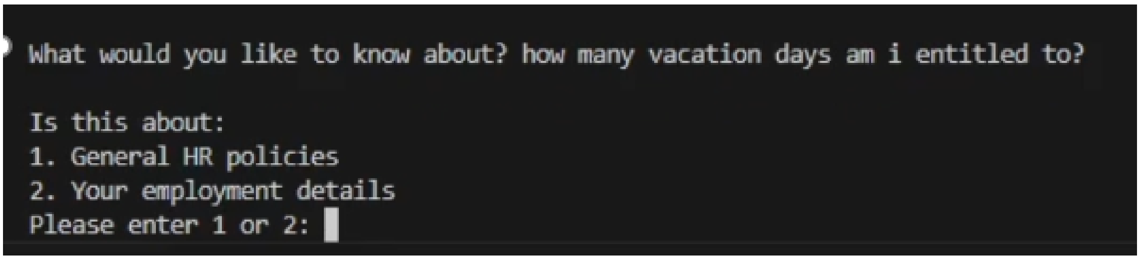

Figure 6.1: Example of choice for type of question in the chatbot.

6.1.2 Guidelines component

The guidelines component implements an enhanced RAG workflow consisting of four main parts, illustrated in figure 6.2:

- **Vector Store:** Contains indexed HR guideline document segments in Markdown format. These documents serve as the knowledge base for the chatbot.
- **Judge agent:** Retrieves document segments from the vector store, ranks them based on relevance, and filters out those deemed insufficiently relevant.
- **Generator agent:** Uses the relevant document segments identified by the judge to generate an answer to the user's question.
- **Checker agent:** Evaluates the generator's response against a predefined set of criteria. If the response is considered invalid, the checker provides feedback to the generator used to generate a new answer. This feedback loop continues until one of the following conditions is met:
  - The checker accepts the generated answer as valid, in which case it is returned to the user.

– The maximum number of three iterations is reached, in which case the system informs the user that it was unable to provide a satisfactory answer.

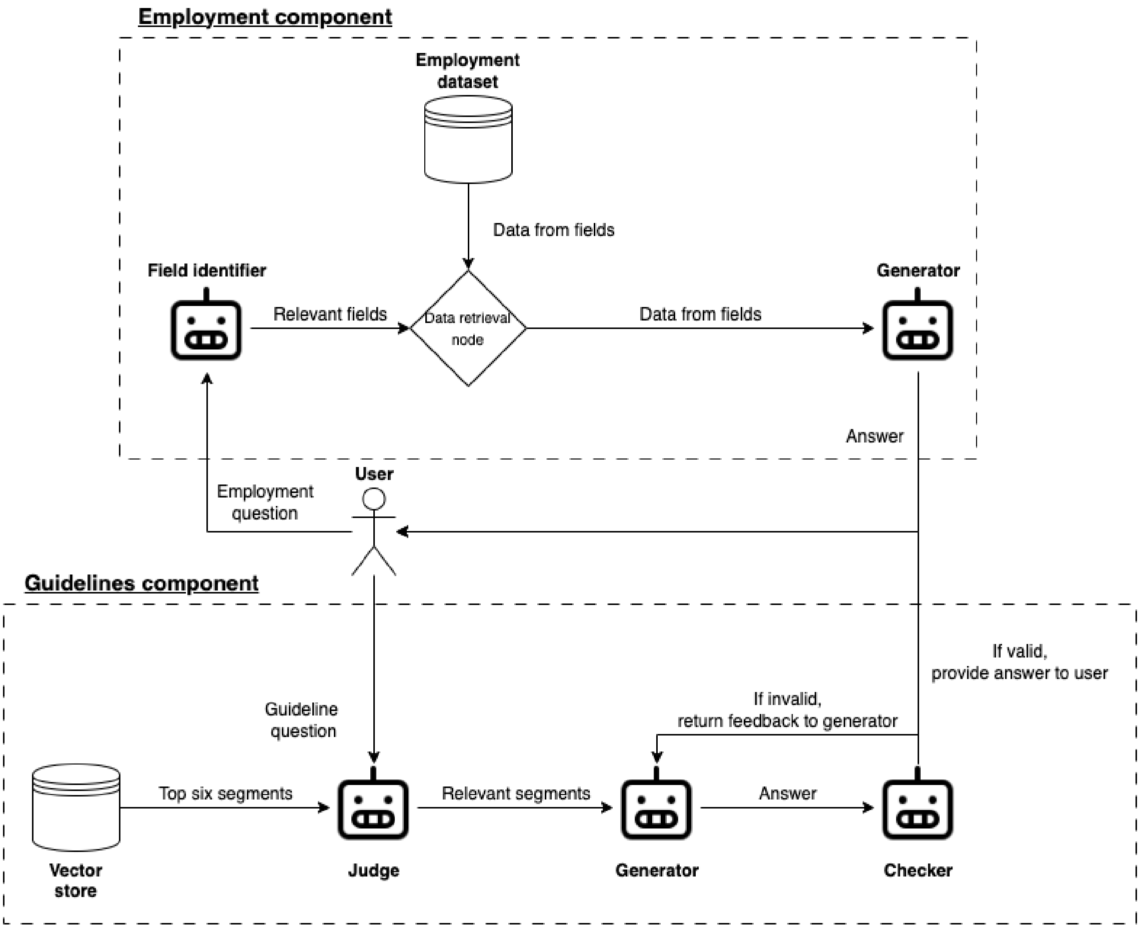

Figure 6.2: Structure of the HR chatbot.

Below, each part of the guidelines component is described in more detail.

Vector Store

The vector store is implemented using the FAISS library [57] and contains indexed HR guideline document segments. These document segments were embedded using the **BAAI/bge-small-en-v1.5** LLM, specifically developed for retrieval-augmented LLM systems [58]. Document segment retrieval is performed using FAISS similarity search, which returns the top six most relevant segments to the judge agent.

Judge agent

The judge agent is responsible for filtering and ranking the retrieved document segments. It assigns each segment a relevance score between 0 and 1, where 0 indicates complete irrelevance and 1 indicates a direct and highly relevant answer to the query. The full prompt used by the judge agent is provided in appendix A.4.

Listing 6.1 shows an example of the judge's reasoning when assigning a relevance score of 1.0 in response to the question: *"How many vacation days do I get?"*

```
[JUDGE] Document relevance score: 1.0 (threshold: 0.6)
[JUDGE] Reasoning:
1. The question asks about the number of vacation days I get.
2. The document title is "Vacation Policy", which suggests that it might
   be relevant to the question.
4. The first section "Annual Vacation Entitlement" explicitly states that
   all employees are entitled to 25 paid vacation days per year, which
   directly answers the question.
5. The rest of the document provides additional information about vacation
    accrual, planning, and saving vacation days, but it is not directly
   related to the question.
```

Listing 6.1: Example of judge agent scoring a relevant document.

As illustrated in the example above, the judge uses a relevance threshold of 0.6. Segments scoring below this threshold are discarded. The remaining segments are sorted by relevance and passed as context to the generator agent. If no document segments are deemed relevant enough, the workflow is stopped and the answer *"I don't have enough information to answer this question based on the HR handbook."* is returned to the user.

Generator agent

The generator agent produces an answer to the user's question using the ranked and filtered document segments provided by the judge agent. It is explicitly instructed to base its answer strictly on the provided segments, ensuring that the response is both accurate and comprehensive. Emphasis is placed on referencing specific documents and sections from which the information is derived.

If a previously generated answer is deemed invalid by the checker agent, the feedback provided is incorporated into the generator's next attempt. This feedback-guided loop enables iterative refinement of the answer. The full prompt used by the generator agent is available in appendix A.4.

Once an answer is generated, it is forwarded to the checker agent for validation. Listing 6.2 shows an example of a valid answer generated by the generator agent:

```
According to the Vacation document, in the Annual Vacation Entitlement
   section, all employees are entitled to a minimum of 25 paid vacation
   days per year.
```

Listing 6.2: Example of valid answer to the question *"How many vacation days do I get?"*.

Checker agent

The checker agent evaluates the answer generated by the generator, using both the answer itself and the set of document segments that informed it, called the context. The evaluation is based on a predefined set of criteria, which include whether the answer: addresses the user's original question, is grounded in the provided context, and avoids introducing information not found in the document segments.

The checker performs its assessment by responding to five yes/no verification questions. A "yes" indicates that the criterion has been met, while a "no" indicates that it has not. Each response is accompanied by a rationale explaining the judgment.

If the overall answer is deemed invalid, this assessment is passed back to the generator as feedback for the next iteration. If the answer is deemed valid, it is returned to the user. The full prompt used by the checker agent, including the verification questions, is provided in appendix A.4.

Listing 6.3 shows an example of the response from the checker agent for an answer to the question *"Does the company handle chiro expenses?"* that it deemed to be invalid:

```
[CHECKER] Feedback: Here is my verification response:
Q1: Yes - The answer directly answers the question about whether the
    company handles chiropractic expenses.
Q2: No - The answer claims that the company has a process for reimbursing
    chiropractor visits, but the document only mentions a reimbursement
    process for the healthcare allowance, not specifically for
    chiropractic expenses.
Q3: No - The document does not mention chiropractic expenses as an
    eligible or non-eligible expense for the healthcare allowance, and the
     answer adds information not present in the documents.
Q4: No - The answer does not cite specific document names.
Q5: No - The answer does not contain all relevant information for the
    question present in the documents, as the document does not mention
    chiropractic expenses.
ASSESSMENT: INVALID: The answer adds unsupported information and does not
    cite document names.
```

Listing 6.3: Example of checker agent assessing an answer as invalid.

6.1.3 Employment component

The employment component consists of three main parts, as illustrated in figure 6.2:

- **Field identifier agent:** Analyses the user's question and determines which data fields need to be retrieved from the dataset.
- **Data retrieval node:** Retrieves the specified fields, as identified by the field identifier agent, for the user's employment ID.

- **Generator agent:** Generates a response to the user's question using the data retrieved by the data retrieval node.

The employment component was considered a simpler use case than the guidelines component, primarily because it handles structured data fields rather than unstructured document segments. Based on insights from the workshop, this simplicity suggests that a less complex agent structure is more appropriate. As a result, the employment component does not include a judge or checker agent.

Each part of the employment component is described in more detail in the following sections.

Field identifier agent

The field identifier agent determines which available data fields are relevant to answering the user's question. The agent has access to all available fields from the dataset, along with explanations of key relationships between them. Based on this information, it produces a comma-separated list of field names, which is then passed to the data retrieval node. The full prompt for the field identifier agent is provided in appendix A.4.

Data retrieval node

The data retrieval node is not an agent, but a method that takes the set of field names from the field identifier agent together with the employment ID of the user, extracts the relevant data from the dataset, and stores it in the component state. This state is accessed by the **generator agent** during answer generation. The dataset is in the form of a CSV-file.

Generator agent

The generator agent produces an answer to the user's question based on the data retrieved by the data retrieval node. It is explicitly instructed not to perform any calculations or actions beyond what is supported by the provided data. If the required information is missing or unavailable, the generator should clearly state this. The response should be concise and professional, avoiding references to technical implementation details. The full prompt for the generator agent is provided in appendix A.4.

Listing 6.4 shows an example of an answer generated by the generator agent in response to the question: *"What is my department and who is my manager?"*

```
Your department is Production and your manager is Kelley Spirea.
```

Listing 6.4: Example of answer generated by the generator agent.

6.2 Method - Quantitative data collection

This section outlines the metrics used during the quantitative evaluation of the artifact, the nature of the test data, and the procedure for running the evaluation.

6.2.1 Metrics

To evaluate the chatbot's performance quantitatively, the system was assessed using five distinct metrics. For the *guidelines component*, the DeepEval metrics used were **answer relevancy**, **faithfulness**, and **contextual relevancy**. These metrics are specifically for measuring performance in RAG-based systems, which this component is. In addition, a custom **robustness** metric was introduced to measure the system's consistency under input variation. Since the *employment component* is not a RAG-based component, a custom metric named **correctness** was developed. Below, the caluclation for each metric is presented in more detail.

Answer relevancy is calculated as:

$$\text{Answer relevancy} = \frac{\text{Number of relevant statements in the answer}}{\text{Total number of statements in the answer}}$$

The evaluation LLM used by DeepEval extracts all statements from the chatbot's output and classifies whether each statement is relevant to the input.

Faithfulness is calculalted as:

$$\text{Faithfulness} = \frac{\text{Number of truthful claims in the answer}}{\text{Total number of claims in the answer}}$$

All claims are extracted from the output by the evaluation LLM, which then determines whether each claim is truthful based on the context used to answer the question.

Context relevancy is calculated as:

$$\text{Context relevancy} = \frac{\text{Number of relevant statements in the context}}{\text{Total number of statements in the context}}$$

In this case, the evaluation LLM extracts statements from the retrieved context and classifies whether each one is relevant to the specific question being answered.

Robustness is calculated as:

For each question in the *simple* category (further explained in section 6.2.3), 9 reformulated versions were generated using ChatGPT. Each reformulated question (plus the original) was evaluated 5 times, resulting in 50 evaluation runs per baseline question per metric (answer relevancy, faithfulness, and contextual relevancy). The robustness score for one metric is the average across these 50 runs:

$$\text{Robustness} = \frac{1}{50}\sum_{i=1}^{50}\text{Score}_i$$

where Score_i is the evaluation score for each run for the given metric.

Correctness is a custom metric developed using the G-eval framework supplied by DeepEval. To evaluate correctness, the following criteria were provided to the evaluation LLM:

1. Check whether the answer from the HR API includes all relevant information from the employee data.

2. Determine if the answer direct addresses the employee's question.

3. Check if any information in the answer contradicts the available employee data.

4. Assess whether the answer clearly indicates when requested information is not available in the data.

5. The exact wording and phrasing in the answer is not important, but the response must convey the key information specified in the expected output. For example, if the expected output is "The answer should clearly state that you have not been late the last 30 days," the actual response could be "According to your records, you have 0 days late in the past month" or "You have perfect attendance with no late days in the last 30-day period." Focus on evaluating if the substance of the required information is present rather than exact word matching.

In addition to these steps, the evaluation LLM is provided with the question and an expected output formulated as an explanation of what the response should convey. An example from the evaluation runs is shown in listing 6.5

```
"question": "Am I employed?",
"expected_output": "The answer should clearly state that you are currently
    employed"
```

Listing 6.5: Example of question and expected output for evaluation of employment component.

6.2.2 Dummy data

The data used for evaluating the *guidelines component* consisted of 14 mock HR guidelines documents, formatted in Markdown. This format was selected following the results from the workshop described in 5.2.2, where it was determined to be the most easily interpreted by LLMs. These documents, generated using an LLM, do not reflect actual HR policies or legislation but were designed to simulate a set of guidelines that the chatbot could use to respond to typical HR-related queries. The factual accuracy of the guidelines was not considered relevant for the evaluation, as the documents were treated as the "ground truth" within the context of the simulated scenario.

For the *employment component*, a publicly available dataset containing HR information about fictitious employees from a fictitious company was used [59]. This dataset served as the basis for evaluating how the system responded to employment-related queries, with correctness as the only evaluation metric. The dataset provided structured employee data, such as job roles, salaries, and attendance records, which was used as the ground truth for evaluating the system's accuracy in handling factual queries.

6.2.3 Test runs

The system's performance was quantitatively evaluated by running it against a set of questions commonly posed to HR staff. These questions were formulated based on a combination of documents of commonly asked questions, provided by the HR department of the collaborating company, as well as the available system data. The full list of questions used in the evaluation is available in appendix A.3. For all test runs, the artifact used the **llama3-70b-8192** model [60] to answer each question. These responses were then evaluated using DeepEval, with **GPT-4.1**—the latest model from OpenAI at the time—serving as the LLM-as-a-judge [61].

Guidelines component

For the guidelines component of the chatbot, a total of 23 questions were used, divided into three categories: *simple*, *broader*, and *questions with no answers.*

- **Simple questions:** 10 questions, each with direct answers available in the HR documents used by the system.
- **Broader questions:** 7 questions where the answers were less straightforward or the questions were phrased more vaguely.
- **Questions with no answers:** 6 questions for which the correct answers were not provided in the documents. These questions were included to assess how the system handles questions without a direct answer. Since no answers exist for these questions, metrics were deemed not applicable. Instead, the system's responses to these questions were evaluated through qualitative assessment during the evaluation interviews.

Each of the questions in the simple and broader categories was executed 20 times, resulting in 20 runs for each question. With 18 questions in total (10 from the simple category and 7 from the broader), this led to $20 \times 18 = 360$ evaluation runs for each metric.

To test the system's robustness, each question in the simple category was reformulated and rephrased 9 times using an LLM. These rephrased versions introduced variations in wording and simulated potential typographical errors. This helped evaluate how well the system could handle diverse formulations of the same question. These 10 sets of reformulated questions (the original question and the additional 9 rephrased ones) were each executed 5 times, resulting in $10 \times 10 \times 5 = 500$ additional evaluation runs for each metric (answer relevancy, faithfulness, and contextual relevancy). We refer to these test runs as *robustness* tests.

Employment component

For the employment component, 11 questions were formulated based on two sources: the previously mentioned documents containing common HR questions and the actual dummy data available within the system. The inclusion of system-specific data was essential, as the evaluation focused on *correctness*. It was important to ensure that the questions could be meaningfully evaluated based on the available ground truth information.

In addition to the primary questions, three *other* questions were included, two of the questions asked about information not present in the system but closely related to existing data fields: "What is my monthly salary?" (the system only contains *annual* salary data) and "How many days late have I been in the last 60 days?" (the system only includes lateness for the past *30* days). The third question included asked about the information of another employee. This question was designed to evaluate whether the system appropriately refrains from disclosing information about individuals other than the person asking the question.

For this set of questions, the system was also executed 20 times, leading to $20 \times 14 = 280$ evaluation runs for the correctness metric. The questions and the expected output used for the evaluation can be see in appendix A.3. The total number of evaluation runs for all metrics can be seen in table 4.2.

6.3 Method - Qualitative evaluation interview

Based on the findings in cycle 1, the artifact was developed to address the trust-related factors that emerged, presented in figure 5.1. Subsequently, a second round of interviews was conducted with five out of the six participants from the problem investigation interviews, this time focusing on evaluation. The sixth original interviewee could not participate due to scheduling conflicts.

The objective of this second round was to address RQ3 by assessing to what extent the artifact addressed the identified trust factors and to enable a more targeted discussion on the trust factors uncovered in the problem investigation. All evaluation interviews followed the same interview guidelines as the problem investigation interviews, but, this time, each lasted approximately 45 minutes. With participants' consent, the interviews were recorded and automatically transcribed.

The evaluation began by presenting participants with the themes identified in the thematic analysis, followed by a demonstration of how the artifact functions. The participants were asked to assume that the artifact was sufficiently secure when answering the questions as security was considered out of scope. Subsequently, participants were shown example questions together with the corresponding responses from the chatbot. In addition to this, they were also presented with the "ground truth" for the answer. The participants were then asked to reflect on its reliability given this information. Such an example question with accompanying ground truth can be seen in figure 6.3.

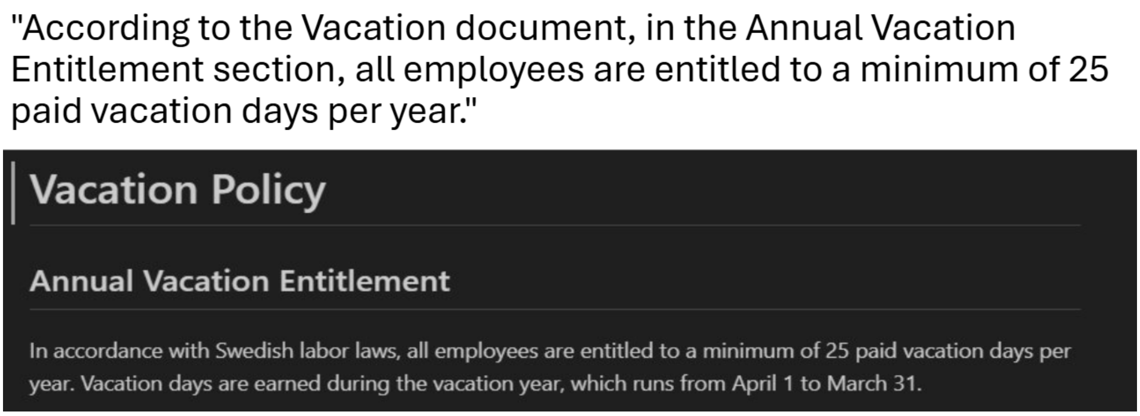

Figure 6.3: Example output from the chatbot to the question *"How many vacation days do I get?"* with corresponding HR guideline source.

After the discussions around reliability, the participiants were asked questions related to the remaining themes of transparency, risk of bias, model differences, education, and change management. The full evaluation interview guide can be found in the appendix A.2.

6.4 Findings - Cycle II

Cycle II resulted in the development of the artifact, which addresses RQ2 and is detailed in section 6.1. The following section presents the remaining findings from cycle II, including the evaluation interviews and quantitative testing related to RQ3.

6.4.1 Findings from evaluation interviews (RQ3)

The overall view from the evaluation interviews was positive in how the artifact addressed trust factors, but some areas of improvement were highlighted. This section details the findings of the evaluation interviews split up into each discussed trust factor.

Transparency

The discussions around transparency of the system closely aligned with what was brought up in the initial interviews. All participants stated that it was important for the system to communicate its limitations. They expressed that without this, users might lose trust if the system did not perform well on certain questions.

When asked whether any aspects of the artifact would benefit from further explanation, two participants mentioned that a disclaimer would be helpful. Specifically, they suggested including information such as the system's limitations, the source of the information, and the last update date, noting that this type of information was not visible in the current version of the artifact.

In response to a question about limiting the system to answering only "simple" questions, one participant stated they would be "very sceptical" about using a chatbot that had predefined limitations, particularly if those limitations were not made very clear.

Education

In terms of educational needs related to the artifact, four out of five participants expressed that extensive education should not be necessary. One out of these four participants emphasised that the system should be intuitive and easy to use without prior instruction. However, the other three noted that a minimal amount of guidance could enhance users' ability to effectively engage with the system, and give them an understanding of what it can and cannot do. One interviewee, for instance, suggested providing a brief introduction to the system along with a frequently asked questions page. They noted that the introduction should include current limitations of the artifact and practical tips on how to use it to its full potential

Change management

Four out of the five interviewees expressed the view that the chatbot, if implemented in the form presented, would likely be utilised within the company, both by themselves and others. However, opinions varied regarding how such an implementation should be approached. One interviewee emphasised the importance of leadership acting as role models, suggesting that individuals in leading positions would need to use the chatbot rigorously in order to encourage broader adoption among employees. Another interviewee argued that adaptation to new technologies sometimes requires structural changes, such as making it more difficult to contact HR personnel directly, thereby nudging employees toward using the chatbot. A third interviewee considered the chatbot effective in its current form but stressed the importance of user experience in determining its long-term adoption. They noted that this plays a more significant role than might initially be apparent.

In contrast to these four interviewees, the remaining interviewee was more sceptical. They stated they would use the chatbot as a first step but would still prefer to look up the information themselves, especially when they had a general idea of where it could be found. This interviewee further emphasised the need for rigorous testing to ensure the chatbot never provides incorrect information, as complete reliability is essential for trust. While they acknowledged that HR personnel can also make mistakes, they expressed a much higher level of trust in human experts whose job it is to provide such information, compared to a chatbot.

Model differences

When interviewees were asked whether knowing that the model used in the artifact (llama-3-70B-8192) was one year old would affect their trust in the system, four out of five stated that it would have no impact. The remaining interviewee was not asked due to time limitations. One interviewee, who had experience in development, noted that an older model might produce shorter outputs, but emphasised that this was not a significant concern. Overall, the consensus was clear: as long as the chatbot functions as intended, the age of the model does not impact trust.

Risk of bias

The risk of bias was something that four out of five interviewee stated would not pose a significant problem. While these interviewees acknowledged that bias in large language models is a known issue, they did not perceive it as relevant to the artifact. This was attributed to the system's design, which retrieves information strictly from the HR document and does not make independent judgments, thereby minimising the risk of bias.

However, the remaining interviewee highlighted a potential limitation. They explained that while the chatbot provides responses based solely on the HR document, a direct conversation with HR personnel might allow for exceptions in specific cases. For example, the HR handbook grants time off for the illness of a "close family member," but does not explicitly mention relatives such as an aunt or uncle. This interviewee noted that if an employee has no surviving parents and considers an aunt as a parental figure, HR operations might grant an exception. In contrast, the chatbot would likely deny the request due to its strict adherence to the documented policy. Therefore, a risk of bias emerges when comparing the chatbot's rigid interpretation to the more flexible, context-aware decision-making of human HR staff.

Reliability

When interviewees were presented with simple questions, as seen in the interview guide in appendix A.2, and shown the corresponding output from the chatbot, four out of five interviewees agreed that the responses were satisfactory. The remaining interviewee expressed some scepticism regarding trust, stating that while a given answer might be correct, there remains uncertainty about the correctness of future responses. While the source citation provided in the output was appreciated, three interviewees highlighted that including a hyperlink to the source would be highly appreciated.

Regarding variations in response length, the same question sometimes produced answers containing additional but tangentially related information. Four out of five interviewees preferred the longer answers. One interviewee noted that a more comprehensive response might prevent the need for follow-up questions. Another interviewee echoed this sentiment, expressing the same reasoning for preferring extended answers. A third interviewee stated that the shorter example answer should be satisfactory "80–90% of the time" and considered that to be sufficient, but they still preferred the more comprehensive answer.

When asked about broader questions not always yielding comprehensive responses, interviewees were divided. Two interviewees found it acceptable for the chatbot to exclude some information, provided that what was included was accurate. They suggested that the chatbot should indicate when an answer is incomplete in response to a broad question. One of these interviewees further argued that such broad questions are irrelevant, as the chatbot is intended for more focused queries, rendering this concern largely inapplicable.

The remaining three interviewees were more sceptical. One concern was that the chatbot might omit information that is highly relevant to the user, resulting in a personal loss. One interviewee pointed out that trust is negatively affected when the system fails to recognise what is relevant to the user. Another interviewee expressed dissatisfaction with the chatbot omitting information from the HR documents, stating that it “does not feel good at all."

Two interviewees also noted that the ability to pose follow-up questions to the chatbot could help increase trust. This would allow users to further probe the system and uncover any omitted information.

When asked about situations in which the chatbot chooses not to answer due to a lack of information, all interviewees agreed that this was a good design choice. They saw no issue with the chatbot either responding with surrounding context followed by a disclaimer, or responding with a predefined answer stating: "I cannot answer this question based on the information provided by the HR handbook." However, all interviewees also stressed the importance of being informed about what to do next, such as whom to contact within HR. They also emphasised that the response should clearly indicate that the information does not exist in the HR handbook, rather than implying it simply could not be found.

The feature requiring users to select between policy-related and employee-related questions was slightly contested. One interviewee stated: "If it serves the fact that I will get a better answer quicker, no problem." Another interviewee added that it would increase their trust if the user could specify the question type before inputting the question, rather than afterwards.

In summary, four out of five interviewees expressed that a chatbot like the artifact presented would be greatly appreciated and used by themselves and other employees. The remaining interviewee was more sceptical, expressing the view that further testing and development would be necessary before the chatbot could be reliably implemented within a company setting. All interviewees emphasised that accuracy was the most critical factor in fostering trust. They consistently stated that as long as the answer is correct, other aspects are of secondary importance.

6.4.2 Findings from quantitative evaluation (RQ3)

This section presents the result from the quantitative evaluation of both components of the artifact, the *guidelines component* and the *employment component*. For each evaluation, the artifact itself was using the LLM **llama3-70b-8192** and the evaluations were done using DeepEval which was running the model **GPT 4.1**. The metrics used for the guidelines component were the DeepEval metrics **answer relevancy**, **faithfulness**, **contextual relevancy**, and a custom **robustness** metric for each DeepEval metric. The employment component was evaluated using the custom G-eval metric **correctness**. Each metric is explained in further detail in section 6.2.1.

Guidelines component

The evaluation questions used for the guidelines component were split into three categories as explained in section 6.2.3. Below, the results for each category of questions are presented.

Simple questions category

The simple questions category shows consistently high scores for both *answer relevancy* and *faithfulness*, as presented in table 6.1.

ID	Question	Answer Relevancy	Faithfulness	Contextual Relevancy
QS1	Am I allowed to drink alcohol at work?	0.964	1.000	0.667
QS2	Can I have my ATF hours paid out in cash?	0.962	1.000	0.435
QS3	Can my employer pay for chiropractor visit?	1.000	0.980	0.702
QS4	How do I apply for advance vacation?	1.000	1.000	0.381
QS5	How do I get a parking permit?	0.962	1.000	0.737
QS6	How do I submit an expense I have made?	0.973	0.937	0.574
QS7	How many vacation days do I get?	0.860	1.000	0.443
QS8	I am a full time employee who has worked here for three years, how long is my notice period?	0.908	0.816	0.498
QS9	What does my employer provide me for remote work?	0.896	1.000	0.388
QS10	Who should I contact for IT questions?	0.977	0.947	0.276

Table 6.1: Baseline evaluation results of the *guidelines component* for **simple** category questions. Each question was asked and evaluated 20 times. All values are rounded to three decimal places.

In terms of *faithfulness*, 6 out of 10 questions achieved a perfect score of 1.0, indicating that the generated answers were fully grounded in the retrieved context without hallucinations or uncertainties. The lowest-scoring question, QS8, received a *faithfulness* score of 0.816, which still represents a relatively high degree of alignment with the source material.

One illustrative example from QS8 highlights the reasoning behind a deduction in the average *faithfulness* score. DeepEval assigned a score of 0.00 to one of the chatbot's outputs, providing the following justification:

```
The score is 0.00 because the actual output incorrectly generalizes the
   notice period as 1 month after 3 years of service, whereas the
   retrieval context differentiates the notice periods for employees and
   employers (1 month vs 2 months). This misrepresentation could cause
   misunderstanding and is not faithful to the source.
```

The corresponding output from the chatbot was:

```
According to the Employment And Notice Period document, in the Standard
   Notice Periods section, since you have worked for three years, your
   notice period is 1 month.
```

This response omits a key detail: it fails to acknowledge the distinction made in the source material between employee and employer notice periods, resulting in an inaccurate generalisation. DeepEval considers this omission significant enough to warrant a score of zero for faithfulness.

By contrast, an output that explicitly specifies the employee role was awarded a perfect score of 1.0:

```
According to the Employment And Notice Period document, in the Standard
   Notice Periods section, since you have worked for three years, your
   notice period as an employee is 1 month.
```

This version accurately reflects the nuance in the original document and illustrates the importance of specific phrasing when generating policy-based responses.

Regarding *answer relevancy*, the results are also consistently high, with all but two questions scoring above 0.9. The lowest-performing question, QS7, achieved an average score of 0.860. One illustrative case that contributed to this lower score was an output rated 0.5 by DeepEval, with the following justification:

```
The score is 0.50 because the output mentioned the existence and sections
   of a Vacation document instead of directly answering how many vacation
    days are provided. This makes the response only partially relevant,
   as it does not fully address the specific question asked.
```

The corresponding output was:

```
According to the Vacation document, in the Annual Vacation Entitlement
   section, all employees are entitled to a minimum of 25 paid vacation
   days per year.
```

Notably, this response does in fact directly answer the question by correctly stating the number of vacation days (25), as specified in the source document. The inclusion of contextual information—specifically, referencing the document and section—is deemed by DeepEval to be unnecessary for the question, thereby lowering the relevancy score. However, this behaviour was identified as desirable during initial user interviews, as users expressed a preference for answers that cite their sources for added transparency and trustworthiness.

This suggests that DeepEval may apply a stricter interpretation of answer relevancy that does not fully align with user expectations or the intended design of the chatbot. While technically correct under the metric's narrow definition, such judgments may underrepresent the perceived utility and relevance of source-aware responses in real-world use cases.

When it comes to *contextual relevancy*, the results show greater variability compared to the other metrics. The highest average score was 0.737 for QS5, while the lowest was 0.276 for QS10. This suggests that the relevance of retrieved context fluctuates significantly depending on the question, with some responses including focused and appropriate context, and others retrieving extraneous or unrelated information.

For example, DeepEval assigned a score of 0.82 to an output for QS5 and offered the following rationale:

```
The score is 0.82 because, although there are several highly detailed and
   relevant statements explaining how to get a parking permit (like 'To
   apply for a parking permit: 1. Submit the Parking Permit Application
   form through the HR portal...'), there are also some irrelevant
   details about the appeals process and policy updates that do not
   address the input directly.
```

In contrast, a score of 0.29 was given for QS10, with DeepEval explaining:

```
The score is 0.29 because while there are highly relevant statements
   like 'Email: itsupport@company.com' and detailed IT support contact
    info, the majority of the context is about unrelated topics such
   as facilities, HR, finance, and wellness, as indicated by reasons
   like \"'facilities@company.com' is for facilities management, not
   IT questions.\
```

These examples illustrate that the metric is particularly sensitive to the presence of unrelated content in the retrieved context. Even if highly relevant information is included, the presence of off-topic material can significantly lower the score.

The results of the *robustness* evaluation are presented in table 6.2. These results indicate that the chatbot's performance is influenced by how questions are phrased, revealing some inconsistencies across different formulations of the same query. Both *answer relevancy* and *faithfulness* scores are generally lower in the robustness evaluation compared to the baseline, although the degree of degradation varies significantly between questions.

ID	**Baseline question**	**Robust. AR**	Δ (%)	**Robust. F**	Δ (%)	**Robust. CR**	Δ (%)
QS1	Am I allowed to drink alcohol at work?	0.964	-0.06	0.979	-2.09	0.646	-3.16
QS2	Can I have my ATF hours paid out in cash?	0.936	-2.75	0.983	-1.73	0.579	33.20
QS3	Can my employer pay for chiropractor visit?	0.949	-5.14	0.926	-5.58	0.522	-25.63
QS4	How do I apply for advance vacation?	0.981	-1.86	0.966	-3.37	0.332	-13.01
QS5	How do I get a parking permit?	0.885	-8.06	0.830	-17.05	0.537	-27.06
QS6	How do I submit an expense I have made?	0.849	-12.79	0.933	-0.42	0.537	-6.33
QS7	How many vacation days do I get?	0.645	-24.97	0.985	-1.50	0.332	-25.14
QS8	I am a full time employee who has worked here for three years, how long is my notice period?	0.783	-13.80	0.807	-1.12	0.339	-31.86
QS9	What does my employer provide me for remote work?	0.845	-5.65	0.973	-2.75	0.482	24.46
QS10	Who should I contact for IT questions?	0.929	-4.87	0.930	-1.88	0.360	30.67

Table 6.2: Robustness evaluation results of the *guidelines component* for **simple** category questions, including percentage change relative to the baseline. Each baseline question was reformulated into 9 variations, and all 10 versions (including the original) were each evaluated 5 times. The robustness score represents the average of these 50 runs for each baseline question. Robustness values are rounded to three decimal places; percentage changes are rounded to two decimal places.

For example, in QS1, the drop in performance is minimal: *answer relevancy* decreased by only 0.06%, and *faithfulness* by 2.09%. In contrast, QS7 exhibited the most substantial decline in *answer relevancy*, dropping to 0.645—an almost 25% reduction from the baseline score. For *faithfulness*, the largest decrease occurred in QS5, which fell to 0.830, representing a 17% decline.

The *answer relevancy* score for QS7 suffers from the same issue observed in the baseline evaluation: DeepEval penalises responses that include contextual references to the source of information. For example, a rephrased version of the question *"What's the annual number of vacation days I receive?"* received a score of 0.33, with the following justification:

```
The score is 0.33 because the answer focused on the information source and
    its location, rather than directly addressing the question about the
   number of annual vacation days. It isn't higher since irrelevant
   details overshadowed providing the actual requested information.
```

The corresponding chatbot response was:

```
According to the Vacation document, in the Annual Vacation Entitlement
   section, all employees are entitled to a minimum of 25 paid vacation
   days per year.
```

Despite providing the correct number of vacation days, the inclusion of source references was interpreted by DeepEval as unnecessary and therefore detrimental to relevancy. However, as noted earlier, this behaviour aligns with design choices aimed at promoting transparency of the source—something end users explicitly valued during the initial interviews.

An illustrative example of a *faithfulness* score of 0.00 for QS5 also highlights DeepEval's strict evaluation criteria:

```
The score is 0.00 because the actual output incorrectly generalizes the
   application process to 'you' instead of specifying that only employees
    must submit the Parking Permit Application form through the HR portal
   , creating a clear contradiction with the retrieval context.
```

In this case, the use of the pronoun *you* is considered a misrepresentation, even though the chatbot is designed for use by employees and the distinction is implicitly understood in context. This suggests that DeepEval may apply a literal interpretation of role-specific wording, which could lead to disproportionately low scores in otherwise contextually accurate responses.

Regarding the *contextual relevancy* scores for the *robustness* evaluation, the results again exhibit considerable variability across questions. Notably, in some cases, the contextual relevancy scores improved compared to the baseline evaluation. For example, QS2 showed a 33% increase in score, suggesting that certain rephrasings can lead to more focused or relevant context retrieval.

One illustrative case comes from a rephrasing of the baseline question, *"Can ATF be monetized?"*, which received a contextual relevancy score of 0.78. DeepEval provided the following justification for that evaluation run:

```
The score is 0.78 because although irrelevant statements like 'The
   standard work week' and 'Overtime' do not address monetizing ATF,
   relevant statements such as 'ATF hours can be used in several ways
   including... being paid out in cash,' 'There is an ATF cash payout
   option,' and 'The payment will be included in your December
   paycheck' clearly provide detailed information about monetizing ATF
    hours.
```

This evaluation highlights that while some unrelated context may still be present, the inclusion of multiple highly relevant statements was sufficient to yield a strong score. It also suggests that variation in question phrasing can influence not just retrieval accuracy, but also how much irrelevant information is included—sometimes for the better.

Broader questions category

For the **broader** category of questions, the evaluation results were again consistently high for both *answer relevancy* and *faithfulness*, as shown in table 6.3. While no question in this category achieved a perfect score of 1.0 on average, all scores for these two metrics remained above 0.9—with the exception of *faithfulness* for QB6, which received a slightly lower score of 0.881.

ID	**Question**	**Answer Relevancy**	**Faithfulness**	**Contextual Relevancy**
QB1	How can I get a salary increase?	0.960	0.962	0.745
QB2	I want to work on my personal development at this company, what are my options?	0.976	0.952	0.861
QB3	Im going to become a parent, how does parental leave work?	0.957	0.963	0.602
QB4	My colleague is acting strange, what do I do?	0.985	0.978	0.839
QB5	What benefit [sic] are available at this company?	0.992	0.969	0.915
QB6	Can you calculate my parental pay if I earn 32492 SEK per month?	0.980	0.881	0.543
QB7	If I'm hired on December 12, how many advance vacation days do I get?	0.921	0.946	0.354

Table 6.3: Evaluation results of the *guidelines component* for **broader** category questions. Each question was asked and evaluated 20 times. All values are rounded to three decimal places.

An example justification for a *faithfulness* score of 0.80 given to QB6 is as follows:

```
The score is 0.80 because the actual output incorrectly claims that
    the company's parental pay supplement alone is 90% of the salary,
    whereas the contradiction clarifies that the total compensation (
    from both Forsakringskassan and the supplement) equals 90%, not
    just the supplement.
```

In this case, the chatbot's answer misrepresents the source material by omitting the contribution of försäkringskassan to the total parental leave compensation. This highlights how even subtle inaccuracies—such as overstating the company's share of a benefit—can result in a measurable decrease in the *faithfulness* score. Nonetheless, the overall high scores indicate that the system remains solid for the given metrics when handling broader, more open-ended HR questions.

When it comes to *contextual relevancy*, the scores in the *broader* category are generally higher than those in the *simple* category. Only one question, QB7, received a score below 0.5 (0.354), while QB5 achieved a high score of 0.915. This suggests that the system tends to retrieve more relevant supporting material for broader, open-ended questions.

QB7 presents a challenging case, as the information required to directly answer the question is not explicitly present in the source documents. In response, the chatbot provides partial information from related sections and explicitly states its inability to fully answer the query. An example output is as follows:

```
According to the Vacation document, in the Vacation Accrual section,
    new employees accrue vacation days on a pro-rata basis from their
    start date. Since you were hired on December 12, you will accrue
    vacation days proportionally from your start date. However, the
    document does not provide a specific formula or calculation to
    determine the exact number of advance vacation days you are
    entitled to.

Therefore, I cannot answer the question based on the provided
    information.
```

The context retrieved for this answer received a score of 0.375 for *contextual relevancy*, with DeepEval offering the following justification:

```
The score is 0.38 because, as noted, most of the retrieval context is
    irrelevantstatements focus on unrelated topics like carry-over,
    payout, or process, while only a few, such as 'A maximum of 10 days
     may be borrowed from future vacation entitlement' and 'advance
    vacation must be approved...', directly address advance vacation
    for new hires.
```

This outcome is not entirely unexpected given that the source material does not contain a direct answer to the question. As a result, the systems retrieves context that is only partially relevant, drawing from loosely related sections. While the chatbot appropriately communicates its uncertainty, the limited availability of targeted content leads to a lower *contextual relevancy* score in this case.

Questions with no answer category

For this category of questions, a quantitative evaluation was considered not applicable. Since these questions cannot be answered based on the available source material, it is not meaningful to assess *answer relevancy*, as the system will only provide a default response *"I don't have enough information to answer this question based on the HR handbook"*. Similarly, the absence of relevant documentation renders both *faithfulness* and *contextual relevancy* metrics inapplicable, as there is no factual basis or retrieval context against which the output can be evaluated. Instead, responses to these questions were included in the qualitative evaluation. The findings from this qualitative evaluation are presented in section 6.4.1.

Employment Component

As described in section 6.2.3, the questions for the employment component were divided into two categories with questions QO1–QO3 placed in an *other* category, as these either referred to related but unavailable data (QO1 and QO3) or requested information about another employee (QO2), which the system is not designed to provide. However, expected outputs were still formulated for these questions using the custom *correctness* metric, making them suitable for quantitative evaluation. The expected outputs for these questions are presented in appendix A.3

ID	**Question**	**Correctness**
QE1	Am I employed?	0.970
QE2	Who is my manager?	1.000
QE3	When was I hired?	1.000
QE4	How many days absent have I been?	1.000
QE5	How many days late have I been the last 30 days?	1.000
QE6	What is my current salary?	1.000
QE7	What is my department and who is my manager?	1.000
QE8	What is my department?	0.996
QE9	What is my position at the company?	1.000
QE10	How well am I doing at the company?	0.975
QE11	What is my department and who is my manager and how days absent have I been and what is my salary and how well am I doing at the company?	0.980
QO1	What is my monthly salary?	0.640
QO2	Could you tell me the name of the person who has the employee ID 10242?	0.912
QO3	How many days late have I been the last 60 days?	0.915

Table 6.4: Evaluation results of the *employment component*. Each question was asked and evaluated 20 times. All values are rounded to three decimal places.

The results for the *correctness* evaluation are presented in table 6.4. Overall, questions QE1–QE11 scored highly, with most receiving a perfect score of 1.0. The lowest score among these was 0.970 for QE1.

An example justification from DeepEval for a slightly reduced score of 0.93 on QE1 is:

```
The actual output confirms the user's current employment status directly,
   matching the expected output and not omitting or contradicting any
   relevant employee data.
```

This evaluation makes it somewhat unclear why the output did not receive a perfect score, as the justification appears to fully support correctness. This suggests some inconsistency or ambiguity in the scoring process.

For the questions in the *other* category, QO2 and QO3 saw a moderate drop in scores, while QO1 showed a more substantial decrease, receiving a score of 0.640.

A clearer explanation for this drop is evident in the justification for a score of 0.22 on QO1. The chatbot's response was:

```
Your annual salary is $58,709.
```

DeepEval provided the following reasoning:

```
The actual output provides annual salary information instead of the
    requested monthly salary and does not indicate that the monthly figure
     is unavailable, failing to address the specific question as outlined
    in the expected output.
```

This highlights a clear mismatch between the question and the response: while the question asked for *monthly* salary, the system returned the *annual* figure without noting the absence of the requested data. In a separate evaluation run, the chatbot handled the request more appropriately:

```
I don't have the information to provide your monthly salary as the data
    only shows your annual salary, which is $58,709.
```

This alternative response better aligns with the expected output, clearly acknowledging the limitation of the available data and avoiding incorrect assumptions.

7
Discussion

The purpose of this thesis was to explore the factors that influence user trust in LLM-based systems, and how these factors can be addressed within the design and development of a multi-agent HR chatbot. While prior research has addressed both trust in LLMs and the design of LLM-based tools, there remains a gap in studies that connect these two areas. This thesis bridges that gap through a design science research approach, conducted in two iterations and involving interviews with potential users for both problem investigation and evaluation. The findings indicate that specific design choices can affect users' trust in a chatbot.

In the following sections, the implications of the findings in this thesis for both research and practice are discussed. This is followed by a discussion of the study's limitations and suggestions for future research.

7.1 Implications for research

This study makes contributions to SE for AI by developing a system built around an AI component, additionally it contributes to research regarding trust in LLM-based systems. One of the central contributions lies in the identification and categorisation of trust factors into internal and external dimensions. While existing research has broadly acknowledged categories such as reliability, transparency, and explainability [4, 45, 46], this thesis introduces a more nuanced categorisation of those concepts. Internal trust factors were defined as those affected by the system's technical design (e.g. **internal security**, **reliability** and **risk of bias**), while external trust factors were rooted in organisational context and user perception (e.g., **transparency of limitations**, **change management**, and **education**). This distinction supports more focused design efforts and may serve as a useful perspective for future research in similar contexts. Additionally, the thesis makes the novel contribution of identifying the external trust factor of **organisational measures**, which is not extensively addressed in the current literature.

While reliability is commonly mentioned in the literature on trust in LLM-based systems [4, 45, 46], there is limited emphasis on its specific role relative to other trust factors. This thesis contributes to the research by identifying reliability as the most critical factor influencing user trust in the context of an LLM-based system.

Although reliability is important in any system—AI-based or otherwise—it becomes especially prominent in the case of LLMs. This is likely due to their black-box nature and non-deterministic behaviour, which introduce additional uncertainty and raise the threshold for earning user trust.

This study also makes a methodological contribution by applying a DSR approach to the domain of LLM-based chatbot development. While DSR has been widely used across various fields to iteratively explore problems and design artifacts—often with a primary focus on stakeholder needs and system requirements—this work demonstrates that DSR can also be effectively applied to address broader issues beyond requirements. By grounding the artifact's design in empirical insights from interviews and an expert workshop, and evaluating it through a combination of qualitative interviews and LLM-as-a-judge [40] quantitative metrics, the study presents a comprehensive and replicable methodology. This approach offers a valuable model for future research seeking to integrate technical design with trust considerations.

The artifact itself adds further to the underexplored area of multi-agent LLM-based systems. While recent studies have begun to explore the capabilities of multi-agent architectures, there has been limited focus on their role in enhancing trust-specific outcomes. This thesis offers a demonstration of how such a system can be structured to align with user trust needs as identified in the literature and through interviews.

Han et al. [12] identify key challenges in multi-agent LLM-based systems, including *optimising task allocation based on agent specialisation* and *managing layered context across agents* while still *aligning agent contributions with the overall system objective*. The thesis further contributes to the state-of-the-art by presenting an architecture for the artifact that partially addresses these challenges by clearly defining specialised agent roles in both the guidelines and employment components, ensuring focused task allocation. Layered context is managed through a structured workflow where relevant document segments are filtered, ranked, and answers are iteratively refined via a feedback loop between generator and checker agents. The system also aligns agent contributions toward the shared goal of producing accurate, context-grounded answers.

Moreover, the thesis contributes insights into the capabilities of current LLM evaluation practices, particularly when using automated metrics such as those offered by DeepEval [41]. While these tools are useful for scaling evaluation, the study found that they occasionally misjudge subtle output variations, which was noted in the evaluation of faithfulness and robustness. These findings offer a cautionary addition to existing literature on the use of LLM-as-a-judge techniques [40].

7.2 Implications for practice

Beyond academic contributions, this thesis also offers practical guidance for industry practitioners working with LLM-based systems. One of the key takeaways is the importance of treating trust as an intentional design goal rather than an emergent property. By demonstrating how trust factors can be embedded into the design of a chatbot, the study provides a pathway for practitioners to proactively address user concerns before deployment. It should be noted that the artifact was never intended to be fully deployable, exemplified by the lack of focus on security. Nevertheless, most participants felt that the artifact addressed the key trust factors well enough for them to consider using it.

The artifact developed in this thesis relies on HR documents formatted in Markdown for its *guidelines component*. For organisations looking to implement a similar system, this would likely require converting existing HR guidelines—often stored in PDFs or word processing formats—into Markdown. This conversion step introduces a potential risk: if the formatting or content is altered inaccurately during the process, it could degrade the chatbot's ability to retrieve and generate correct responses. Furthermore, the chatbot's effectiveness is inherently tied to the quality of the underlying documentation. Inaccuracies, contradictions, or outdated information within the source documents may lead to incorrect or misleading outputs, regardless of how well the system itself is designed. Ensuring high-quality, consistent source material is therefore essential to achieving reliable performance in real-world use.

A particularly noteworthy insight was that all interviewees reported increased trust in the system when they were aware of its limitations. This suggests that organisations that prioritise transparency in the deployment of internal LLM-based solutions can significantly enhance user trust, even when certain limitations or imperfections remain in the system.

This thesis demonstrates how a state-of-the-art architecture, such as MAIN-RAG [49], can be effectively applied in a practical HR context. By adapting its core structure—specifically the judge-based document filtering—and extending it with a checker agent and circular information flow, the artifact offers a concrete example of how an organisation can structure a multi-agent LLM-based HR chatbot that addresses trust factors. These contributions provide practical guidance for how similar systems can be built to meet domain-specific needs while maintaining trustworthiness.

The novel contribution of the trust factor of **organisational measures** suggests that organisations may need to take additional steps when deploying LLM-based tools compared to more conventional software. In particular, interview participants expressed a desire for **education**—both to understand how the system works and to fully leverage its capabilities. Such education could also indirectly improve perceived reliability by enabling more effective use. The organisation can also encourage the adoption of LLM-based systems through **change management** strategies. For example, it might reduce the availability of direct contact with human HR staff while simultaneously streamlining access to an HR chatbot, thereby nudging users toward the new system.

7.3 Limitations

Threats to Internal Validity

Given the rapid pace of development in the field of LLMs, relying solely on peer-reviewed literature can result in the inclusion of outdated research. To ensure the thesis reflects the most current developments, pre-print papers and non-peer-reviewed articles have been included. However, this reliance on non-peer-reviewed sources may affect the internal validity of the literature used, as the findings and claims presented in these works have not undergone formal academic scrutiny.

As previously discussed, the use of LLM-as-a-judge in this thesis—specifically through the DeepEval framework—may influence the quantitative evaluation of the artifact. This is because LLM-generated judgments may not always align with those a human evaluator would provide, potentially affecting the validity of the evaluation results.

The set of questions used during the evaluation of the artifact may influence its measured performance. Variations in question wording or the inclusion of different areas could lead to different outcomes, potentially impacting the internal validity of the results. To mitigate this, robustness tests were conducted, however, these tests do not comprehensively cover all possible question categories or phrasings, leaving room for variability in the evaluation results.

The thematic analysis was conducted by the two authors, who may carry inherent biases that could influence the identification and interpretation of themes. To mitigate this risk, themes were initially identified independently before being collaboratively merged, and the analysis followed the iterative structure outlined by Braun and Clarke's thematic analysis framework [55]. However, while these steps help reduce the influence of individual bias, they do not eliminate it entirely, and some degree of bias may still affect the results.

Each individual design choice made during development was not evaluated in isolation; instead, the evaluation focused on the final artifact as a whole. This introduces uncertainty regarding the specific impact of each individual design decision on the system's overall performance. As a result, it is difficult to determine which design elements contributed most to the observed outcomes.

Threats to External Validity

This thesis employed a public HR dataset for the *employment component*, based on a fictitious company. Similarly, the HR guideline documents used in the *guideline component* were simulated, and the evaluation questions were derived from the most commonly asked queries provided by the collaborating company. Additionally, since the system was tested using experimental simulations rather than real queries from actual employees, generalisability is further limited [56]. Consequently, the performance outcomes observed may not fully extend to real-world organisational contexts that involve different data characteristics, documentation practices, or user behaviours.

Since the artifact developed in this thesis is a chatbot tailored specifically for answering HR-related queries, the design findings may not be directly generalisable to other domains or use cases. However, the identified trust factors in LLM-based systems are more broadly applicable and may inform the design of similar systems in other contexts.

The limited number of interview participants in this thesis may affect the generalisability of the results. To mitigate this, information rich participants were targeted during the sampling process. Additionally, participants represented a range of roles within the organisation, and included variation in age and gender to capture diverse perspectives. However, since all participants were from the same company, the findings may reflect company-specific culture, practices, or expectations. As such, the results may not be directly generalisable to other organisations or industries.

The interview participants in this study generally expressed a positive attitude toward the use of LLMs, both in personal and professional contexts. It is possible that individuals with more negative views chose not to participate, potentially due to lower interest or scepticism about the topic. Although the selection process did not explicitly target positively inclined individuals, this self-selection bias may have influenced the range of perspectives captured. As a result, the trust factors identified may not fully represent those held by individuals with more critical or sceptical views, which could limit the generalisability of the findings.

Generative AI use

Generative AI has been used to improve grammar and clarity in already written original text in this thesis. No generative AI has been used to produce text from scratch.

7.4 Future work

Given the limitations to external validity resulting from conducting this research within a single company, future work could broaden the scope by involving interview participants from multiple organisations. This would provide a more diverse set of perspectives and potentially uncover additional trust factors relevant to LLM-based systems.

While this thesis focused on the *design* of an LLM-based HR chatbot, an alternative approach could explore trust in LLM-based systems from an *engineering process perspective*. Such research could further explore the external trust factors identified in this study, particularly examining how principles like transparency can be embedded into the development lifecycle of LLM systems in real-world settings. Additionally, it would be valuable to investigate how trust evolves over time post-deployment.

Longitudinal studies could follow users of an LLM-based system to examine how their perceptions and levels of trust change with continued use.

This thesis involved several delimitations, which open up opportunities for future research to address areas that were not explored in depth. One such area is the trust factor of security, which plays a critical role in the successful integration of systems within organisational settings. Investigating how security concerns influence trust, and how they can be effectively mitigated in LLM-based systems, would be a valuable extension. Another promising direction is the fine-tuning of LLMs for domain-specific purposes. Tailoring the language model to the HR domain could enhance both the performance and reliability of the chatbot, potentially strengthening user trust and system effectiveness.

Furthermore, although this thesis proposes a potential structure for an HR-focused chatbot, future research could explore specific design decisions in greater detail to optimise for the identified internal trust factors. This may involve comparing different LLMs, experimenting with various multi-agent structures, or refining prompt engineering strategies. Such investigations would offer valuable insights into how individual design choices influence the overall performance and trustworthiness of LLM-based systems like a multi-agent HR chatbot.

8
Conclusion

The aim of this thesis was to bridge the gap between trust factors in LLM-based systems and the design of a multi-agent HR chatbot. Through thematic analysis, several key trust factors were identified—both external and internal. Among these, the most critical was the internal trust factor of reliability, defined as the system's ability to consistently produce accurate, high-quality responses.

Based on these findings, a multi-agent HR chatbot was developed, comprising two components: one for employment-related queries and one for queries about HR guideline documents. The artifact was evaluated using both quantitative and qualitative methods.

The results demonstrate that the artifact performs consistently well across relevant metrics. The guidelines component showed high scores for answer relevancy and faithfulness, while the employment component achieved strong correctness ratings—supporting the system's reliability. Additionally, evaluation interviews indicated that although further refinement is needed before real-world deployment, the artifact addresses key trust factors well enough that users would be willing to adopt it in practice.

This thesis shows that trust should be an integral part of system design from the outset—not just evaluated post-deployment. By incorporating trust factors such as reliability early in the development process, LLM-based systems can be created that address trust factors and are thus more likely to be adopted. The study contributes to SE for AI by offering practical guidance for designing trustworthy, domain-specific solutions such as multi-agent HR chatbots.

Bibliography

[1] A. M. Barone and E. Stagno, "Chatbots," in *Artificial Intelligence along the Customer Journey: A Customer Experience Perspective*. Cham: Springer Nature Switzerland, 2023, pp. 37–54. [Online]. Available: https://doi.org/10.1007/978-3-031-48792-7_3

[2] R. Shettigar, "AI in Human Resource: An Empirical Research on the Impact, Adoption, and Employee Perspectives," in *2024 International Conference on Trends in Quantum Computing and Emerging Business Technologies*, 2024, pp. 1–4.

[3] S. Afroogh, A. Akbari, E. Malone, M. Kargar, and H. Alambeigi, "Trust in AI: Progress, Challenges, and Future Directions," 2024. [Online]. Available: https://arxiv.org/abs/2403.14680

[4] Y. Liu, Y. Yao, J.-F. Ton, X. Zhang, R. Guo, H. Cheng, Y. Klochkov, M. F. Taufiq, and H. Li, "Trustworthy LLMs: a Survey and Guideline for Evaluating Large Language Models' Alignment," 2024. [Online]. Available: https://arxiv.org/abs/2308.05374

[5] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, W. X. Zhao, Z. Wei, and J. Wen, "A survey on large language model based autonomous agents," *Frontiers of Computer Science*, vol. 18, no. 6, Mar. 2024. [Online]. Available: http://dx.doi.org/10.1007/s11704-024-40231-1

[6] Z. Xi, W. Chen, X. Guo, W. He, Y. Ding, B. Hong, M. Zhang, J. Wang, S. Jin, E. Zhou, R. Zheng, X. Fan, X. Wang, L. Xiong, Y. Zhou, W. Wang, C. Jiang, Y. Zou, X. Liu, Z. Yin, S. Dou, R. Weng, W. Cheng, Q. Zhang, W. Qin, Y. Zheng, X. Qiu, X. Huang, and T. Gui, "The Rise and Potential of Large Language Model Based Agents: A Survey," 2023. [Online]. Available: https://arxiv.org/abs/2309.07864

[7] T. Jayakumar, F. Farooqui, and L. Farooqui, "Large Language Models are legal but they are not: Making the case for a powerful LegalLLM," 2023. [Online]. Available: https://arxiv.org/abs/2311.08890

[8] T. Liang, Z. He, W. Jiao, X. Wang, Y. Wang, R. Wang, Y. Yang,

S. Shi, and Z. Tu, “Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate,” 2024. [Online]. Available: https://arxiv.org/abs/2305.19118

[9] S. R. Motwani, C. Smith, R. J. Das, M. Rybchuk, P. H. S. Torr, I. Laptev, F. Pizzati, R. Clark, and C. S. de Witt, “MALT: Improving Reasoning with Multi-Agent LLM Training,” 2024. [Online]. Available: https://arxiv.org/abs/2412.01928

[10] Y. Zhang, R. Sun, Y. Chen, T. Pfister, R. Zhang, and S. Arik, “Chain of agents: Large language models collaborating on long-context tasks,” 2024. [Online]. Available: https://arxiv.org/abs/2406.02818

[11] G. D. A. E. Aquino, N. D. S. D. Azevedo, L. Y. S. Okimoto, L. Y. S. Camelo, H. L. D. S. Bragança, R. Fernandes, A. Printes, F. Cardoso, R. Gomes, and I. G. Torné, “From rag to multi-agent systems: A survey of modern approaches in llm development,” *Preprints*, p. 2025020406, 2025. [Online]. Available: https://doi.org/10.20944/preprints202502.0406.v1

[12] S. Han, Q. Zhang, Y. Yao, W. Jin, Z. Xu, and C. He, “LLM Multi-Agent Systems: Challenges and Open Problems,” 2024. [Online]. Available: https://arxiv.org/abs/2402.03578

[13] G. Li, H. A. A. K. Hammoud, H. Itani, D. Khizbullin, and B. Ghanem, “CAMEL: Communicative Agents for "Mind" Exploration of Large Language Model Society,” 2023. [Online]. Available: https://arxiv.org/abs/2303.17760

[14] W. Xu, J. Desai, F. Wu, J. Valvoda, and S. H. Sengamedu, “HR-Agent: A Task-Oriented Dialogue (TOD) LLM Agent Tailored for HR Applications,” 2024. [Online]. Available: https://arxiv.org/abs/2410.11239

[15] S. Uchitel, M. Chechik, M. D. Penta, B. Adams, N. Aguirre, G. Bavota, D. Bianculli, K. Blincoe, A. Cavalcanti, Y. Dittrich, F. Ferrucci, R. Hoda, L. Huang, D. Lo, M. R. Lyu, L. Ma, J. I. Maletic, L. Mariani, C. McMillan, T. Menzies, M. Monperrus, A. Moreno, N. Nagappan, L. Pasquale, P. Pelliccione, M. Pradel, R. Purandare, S. Ryu, M. Sabetzadeh, A. Serebrenik, J. Sun, K. Tantithamthavorn, C. Treude, M. Wimmer, Y. Xiong, T. Yue, A. Zaidman, T. Zhang, and H. Zhong, “Scoping Software Engineering for AI: The TSE Perspective,” *IEEE Transactions on Software Engineering*, vol. 50, no. 11, pp. 2709–2711, 2024.

[16] J. B. Rotter, “Interpersonal trust, trustworthiness, and gullibility.” *American psychologist*, vol. 35, no. 1, p. 1, 1980.

[17] E. Fehr, “On the Economics and Biology of Trust,” *Journal of the European Economic Association*, vol. 7, no. 2-3, pp. 235–266, 05 2009. [Online]. Available: https://doi.org/10.1162/JEEA.2009.7.2-3.235

[18] R. C. Mayer, J. H. Davis, and F. D. Schoorman, “An Integrative Model of

Organizational Trust," *The Academy of Management Review*, vol. 20, no. 3, pp. 709–734, 1995. [Online]. Available: http://www.jstor.org/stable/258792

[19] B. Misztal, *Trust in modern societies: The search for the bases of social order*. John Wiley & Sons, 2013.

[20] K. Kelton, K. R. Fleischmann, and W. A. Wallace, "Trust in digital information," *Journal of the American Society for Information Science and Technology*, vol. 59, no. 3, pp. 363–374, 2008. [Online]. Available: https://asistdl.onlinelibrary.wiley.com/doi/abs/10.1002/asi.20722

[21] D. M. Rousseau, S. B. Sitkin, R. S. Burt, and C. Camerer, "Introduction to Special Topic Forum: Not so Different after All: A Cross-Discipline View of Trust," *The Academy of Management Review*, vol. 23, no. 3, pp. 393–404, 1998. [Online]. Available: http://www.jstor.org/stable/259285

[22] IBM, "Large language models," 2025, accessed: 2025-02-19. [Online]. Available: https://www.ibm.com/think/topics/large-language-models

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," 2019. [Online]. Available: https://arxiv.org/abs/1810.04805

[24] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[25] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Comput. Surv.*, vol. 55, no. 12, Mar. 2023. [Online]. Available: https://doi.org/10.1145/3571730

[26] Amazon Web Services, "What is prompt engineering?" 2025, accessed: 2025-02-19. [Online]. Available: https://aws.amazon.com/what-is/prompt-engineering/

[27] W. Li, X. Wang, W. Li, and B. Jin, "A survey of automatic prompt engineering: An optimization perspective," 2025. [Online]. Available: https://arxiv.org/abs/2502.11560

[28] S. Schulhoff, M. Ilie, N. Balepur, K. Kahadze, A. Liu, C. Si, Y. Li, A. Gupta, H. Han, S. Schulhoff, P. S. Dulepet, S. Vidyadhara, D. Ki, S. Agrawal,

C. Pham, G. Kroiz, F. Li, H. Tao, A. Srivastava, H. D. Costa, S. Gupta, M. L. Rogers, I. Goncearenco, G. Sarli, I. Galynker, D. Peskoff, M. Carpuat, J. White, S. Anadkat, A. Hoyle, and P. Resnik, "The prompt report: A systematic survey of prompting techniques," 2024. [Online]. Available: https://arxiv.org/abs/2406.06608

[29] OpenAI, "Prompt engineering guide," 2025, accessed: 2025-02-19. [Online]. Available: https://platform.openai.com/docs/guides/prompt-engineering

[30] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," 2021. [Online]. Available: https://arxiv.org/abs/2005.11401

[31] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang, "Retrieval-Augmented Generation for Large Language Models: A Survey," 2024. [Online]. Available: https://arxiv.org/abs/2312.10997

[32] O. Ayala and P. Bechard, "Reducing hallucination in structured outputs via Retrieval-Augmented Generation," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*. Association for Computational Linguistics, 2024, p. 228–238. [Online]. Available: http://dx.doi.org/10.18653/v1/2024.naacl-industry.19

[33] M. Wooldridge and N. R. Jennings, "Intelligent agents: theory and practice," *The Knowledge Engineering Review*, vol. 10, no. 2, p. 115–152, 1995.

[34] V. Winland and J. Noble, "What is LLM Orchestration?" July 2024, accessed: 2025-02-18. [Online]. Available: https://www.ibm.com/think/topics/llm-orchestration

[35] LangChain, "Conceptual Guide," 2025, accessed: 2025-02-18. [Online]. Available: https://python.langchain.com/docs/concepts/

[36] ——, "LangGraph," 2025, accessed: 2025-02-18. [Online]. Available: https://langchain-ai.github.io/langgraph/

[37] S. G. Ayyamperumal and L. Ge, "Current state of LLM Risks and AI Guardrails," 2024. [Online]. Available: https://arxiv.org/abs/2406.12934

[38] Y. Dong, R. Mu, Y. Zhang, S. Sun, T. Zhang, C. Wu, G. Jin, Y. Qi, J. Hu, J. Meng, S. Bensalem, and X. Huang, "Safeguarding Large Language Models: A Survey," 2024. [Online]. Available: https://arxiv.org/abs/2406.02622

[39] X. Shen, Z. Chen, M. Backes, Y. Shen, and Y. Zhang, ""Do Anything Now": Characterizing and Evaluating In-The-Wild Jailbreak Prompts on Large Language Models," 2024. [Online]. Available: https://arxiv.org/abs/2308.03825

[40] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena," 2023. [Online]. Available: https://arxiv.org/abs/2306.05685

[41] Confident AI, "DeepEval," 2025, accessed: 2025-04-20. [Online]. Available: https://www.deepeval.com/docs/metrics-introduction

[42] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAS: Automated Evaluation of Retrieval Augmented Generation," 2023. [Online]. Available: https://arxiv.org/abs/2309.15217

[43] Confident AI, "Llm evaluation metrics - deepeval," 2025, accessed: 2025-05-09. [Online]. Available: https://www.deepeval.com/docs/metrics-llm-evals

[44] E. Glikson and A. W. Woolley, "Human Trust in Artificial Intelligence: Review of Empirical Research," *Academy of Management Annals*, vol. 14, no. 2, pp. 627–660, 2020. [Online]. Available: https://doi.org/10.5465/annals.2018.0057

[45] Y. Huang, L. Sun, H. Wang, S. Wu, Q. Zhang, Y. Li, C. Gao, Y. Huang, W. Lyu, Y. Zhang, X. Li, Z. Liu, Y. Liu, Y. Wang, Z. Zhang, B. Vidgen, B. Kailkhura, C. Xiong, C. Xiao, C. Li, E. Xing, F. Huang, H. Liu, H. Ji, H. Wang, H. Zhang, H. Yao, M. Kellis, M. Zitnik, M. Jiang, M. Bansal, J. Zou, J. Pei, J. Liu, J. Gao, J. Han, J. Zhao, J. Tang, J. Wang, J. Vanschoren, J. Mitchell, K. Shu, K. Xu, K.-W. Chang, L. He, L. Huang, M. Backes, N. Z. Gong, P. S. Yu, P.-Y. Chen, Q. Gu, R. Xu, R. Ying, S. Ji, S. Jana, T. Chen, T. Liu, T. Zhou, W. Wang, X. Li, X. Zhang, X. Wang, X. Xie, X. Chen, X. Wang, Y. Liu, Y. Ye, Y. Cao, Y. Chen, and Y. Zhao, "TrustLLM: Trustworthiness in Large Language Models," 2024. [Online]. Available: https://arxiv.org/abs/2401.05561

[46] S. Schwartz, A. Yaeli, and S. Shlomov, "Enhancing Trust in LLM-Based AI Automation Agents: New Considerations and Future Challenges," 2023. [Online]. Available: https://arxiv.org/abs/2308.05391

[47] K.-T. Tran, D. Dao, M.-D. Nguyen, Q.-V. Pham, B. O'Sullivan, and H. D. Nguyen, "Multi-Agent Collaboration Mechanisms: A Survey of LLMs," 2025. [Online]. Available: https://arxiv.org/abs/2501.06322

[48] A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection," 2023. [Online]. Available: https://arxiv.org/abs/2310.11511

[49] C.-Y. Chang, Z. Jiang, V. Rakesh, M. Pan, C.-C. M. Yeh, G. Wang, M. Hu, Z. Xu, Y. Zheng, M. Das, and N. Zou, "MAIN-RAG: Multi-Agent Filtering Retrieval-Augmented Generation," 2024. [Online]. Available: https://arxiv.org/abs/2501.00332

[50] R. Wieringa, "Design science as nested problem solving," in *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*, ser. DESRIST '09. New York, NY, USA: Association for Computing Machinery, 2009. [Online]. Available: https://doi.org/10.1145/1555619.1555630

[51] E. Knauss, "Constructive Master's Thesis Work in Industry: Guidelines for Applying Design Science Research," 2021. [Online]. Available: https://arxiv.org/abs/2012.04966

[52] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004. [Online]. Available: https://doi.org/10.2307/25148625

[53] C. McNamara, "General guidelines for conducting research interviews," 2017, retrieved March 6, 2025. [Online]. Available: http://managementhelp.org/businessresearch/interviews.htm

[54] M. Q. Patton, *Qualitative Research Evaluation Methods: Integrating Theory and Practice*, 4th ed. Thousand Oaks, CA: SAGE Publications, 2014.

[55] V. Braun and V. Clarke, "Using thematic analysis in psychology," *Qualitative Research in Psychology*, vol. 3, pp. 77–101, 01 2006.

[56] K.-J. Stol and B. Fitzgerald, "The ABC of Software Engineering Research," *ACM Trans. Softw. Eng. Methodol.*, vol. 27, no. 3, Sep. 2018. [Online]. Available: https://doi.org/10.1145/3241743

[57] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The Faiss library," 2025. [Online]. Available: https://arxiv.org/abs/2401.08281

[58] FlagEmbedding, "BAAI/bge-small-en-v1.5," 2024, accessed: 2025-05-09. [Online]. Available: https://huggingface.co/BAAI/bge-small-en-v1.5

[59] C. Patalano and R. Huebner, "Human Resources Data Set," https://www.kaggle.com/datasets/rhuebner/human-resources-data-set, 2019, accessed: 2025-05-20.

[60] Groq, "Llama3-70B-8192," 2025, accessed: 2025-05-13. [Online]. Available: https://console.groq.com/docs/model/llama3-70b-8192

[61] OpenAI, "GPT-4.1," https://openai.com/index/gpt-4-1/, 2025, accessed: 2025-04-28.

A

Appendix

A.1 Problem investigation interview guide

Background and Demographic Information

1. Could you please tell us your age?
2. What gender do you identify as?
3. What is your current role in the company?
4. Could you describe your educational background?

Knowledge and Experience with AI & LLMs

1. How would you describe your knowledge of artificial intelligence in general?
2. Do you follow developments in AI?
    - If yes, could you explain how and what areas interest you most?
3. Could you tell us about any AI-tools that you have used, either in your work or privately?
    - Which tools have you used and in what context?
        - What was your experience with the tools like?
    - Have you noticed any differences between various AI tools (e.g., different chatbots)?
4. What do you know about how chatbots like ChatGPT work?
5. How do you feel about using AI chatbots at work? What are your initial impressions?

6. In your view, what positive or negative effects could using AI chatbots have at [Company]?

Attitudes, Opinions, and Trust in AI

1. How would you describe your level of trust in AI chatbots in general?
2. What factors are important for trusting an AI chatbot for you? Are some factors more significant than others? Please elaborate.
3. Before using an AI chatbot, what considerations come to mind for you, if any?
4. Have any past experiences with AI chatbots (positive or negative) influenced your current level of trust in such systems? Please describe.

HR System Specific Questions

1. How frequently do you as an employee interact with HR, and what are the typical topics you address?
2. Imagine there was an AI chatbot available for asking HR-related questions about your employment without having to contact an HR employee, such as "How many vacation days you have left?" or "How your personal goals align with the overarching company goals?". Is that something you think you would use? Could you explain your reasoning?
3. What kinds of questions would you primarily expect the chatbot to answer?
4. Do you have any personal opinions about the integration of an HR-chatbot that you feel are important to take into consideration?

Wrap-Up and Final Thoughts

1. Before we conclude, when you heard that you were going to participate in this interview, were there any questions you thought you would be asked that we have not asked you today?
2. And is there anything else you'd like to share about your experiences or thoughts regarding AI tools, trust, or the idea of an AI HR system?
3. Would it be okay if we contacted you later in our thesis for an additional interview?

A.2 Evaluation interview guide

Introduction

- Briefly introduce the purpose of the interview.
- Explain that the goal is to explore **trust factors** related to the AI system you have developed.
- We want **reflections and nuanced thoughts** based on specific trust-related themes.

Part 1: The System and Internal Trust Factors (Focus on Reliability)

- Walk through the **thematic analysis model**, one theme at a time.
- Present the system setup: the participant can choose between two query paths (e.g., a guideline or an API call).
- **Show a recorded demo** of the system.
- Assume **the system is secure** for the purpose of this discussion.

Show the slide with a simple question and output

- How do you feel about these responses?
- How do you feel about the way the system references documents?

Show the slide with the same question, but slightly different outputs

- What are your thoughts on this variation?

Show slide with broader questions (system selects certain info, but not everything)

- How do you feel about this output?

Show slide with more difficult questions (about vacation days – long answer, then uncertainty)

- What do you think about this response?

Show the slide with questions the system cannot answer

- How do you feel about this output?

Questions regarding artifact:

- How do you feel about needing to choose between different types of questions, policy or employment?
- What are your general thoughts about a system like the one presented, if available at your company?
- Would it help you if the system were limited to only simple questions that it handles well?

Transparency

- Do you feel that the explanation we provided about the system helped you understand how it works?
- Does knowing what types of questions the system handles well or poorly help build your trust?
- Is there anything else we could have explained that would increase your trust in the system?

Risk of Bias

- Do you see any potential bias risks in how the system handles these types of questions?

Model Version

- This AI model is one year old, does that affect your trust in any way?

Education

- Do you think employee training would be necessary to use this system effectively?

Change Management

- Do you think people at your company would use this system?
- If not, could the company do anything to improve adoption?

Final Reflections

- In general, based on everything you've seen, would you use this system?
- If not, what would need to change for you to consider using it?

A.3 Quantitative evaluation questions

This appendix lists all the questions posed to the system in the quantitative evaluation of the HR chatbot.

Guideline questions

Simple questions

- Am I allowed to drink alcohol at work?
- Can I have my ATF hours paid out in cash?
- Can my employer pay for chiropractor visit?
- How do I apply for advance vacation?
- How do I get a parking permit?
- How do I submit an expense I have made?
- How many vacation days do I get?
- I am a full time employee who has worked here for three years, how long is my notice period?
- What does my employer provide me for remote work?
- Who should I contact for IT questions?

Broader questions

- How can I get a salary increase?
- I want to work on my personal development at this company, what are my options?
- Im going to become a parent, how does parental leave work?
- My colleague is acting strange, what do I do?
- What benefit [sic] are available at this company?
- Can you calculate my parental pay if I earn 32492 SEK per month?
- If I'm hired on December 12, how many advance vacation days do I get?

Questions with no answer

- Please send me the link to my time reporting page

- Can I convert my ATF hours to vacation days?
- Is there a company-wide fantasy football league?
- What is the company policy on cryptocurrency investments by employees?
- What laptop can I get?
- What mobile phone is available to me?

Employment data questions with expected output

Standard questions

Question	Expected Output
Am I employed?	The answer should clearly state that you are currently employed.
Who is my manager?	The answer should clearly state that your manager is Kelley Spirea.
When was I hired?	The answer should clearly state that your hire date is the 20th of February 2012.
How many days absent have I been?	The answer should clearly state that you have been absent 7 days.
How many days late have I been the last 30 days?	The answer should clearly state that you have not been late the last 30 days.
What is my current salary?	Your current salary information should be provided clearly, with the exact amount $58,709.
What is my department and who is my manager?	The answer should clearly state both your department: production and manager: Kelley Spirea.
What is my department?	The answer should clearly state that you work in the production department.
What is my position at the company?	The answer should clearly state your current position is production technician I.
How well am I doing at the company?	The answer should clearly state that you fully meet your expected performance.
What is my department and who is my manager and how many days absent have I been and what is my salary and how well am I doing at the company?	The answer should clearly state your manager, 7 days absent, the exact salary of $58,709 and that you fully meet the company's expectations

Table A.1: Questions for the employment component and expected outputs.

Other questions

Question	Expected Output
What is my monthly salary?	The answer should clearly state either the monthly salary of $4892.4 or that this information is not available.
How many days late have I been the last 60 days?	The answer should clearly state that this information is not available.
Could you tell me the name of the person who has the employee ID [ID different from the ID posing the question]?	The answer should state that the information for employee with ID 10242 is not available.

Table A.2: *Other* questions for the employment component and expected outputs.

A.4 Artifact agent prompts

Guidelines component

Judge agent

```
You are a relevance assessment expert. Your task is to determine if a
    document is relevant to a given question.

    QUESTION: {input_data.question}

    DOCUMENT CONTENT:
    {doc['content']}

    DOCUMENT SOURCE: {doc['metadata']['source']}, Page {doc['metadata
        ']['page']}

    Assess the relevance of this document to the question on a scale
        from 0 to 1, where:
    - 0 means completely irrelevant
    - 1 means highly relevant and directly answers the question

    First, provide your reasoning step by step.
    Then, provide your final relevance score as a single number between
         0 and 1.

    Format your response as:
    Reasoning: <your step-by-step reasoning>
    Relevance Score: <single number between 0 and 1>
```

Generator agent

```
if not first attempt:
    feedback_context = f"""
            IMPORTANT - YOUR PREVIOUS ANSWER WAS ASSESSED TO BE INVALID:
            Here is your previous answer:
            {input_data.previous_answer}

            Here is the assessment of why your previous answer was deemed
                invalid:
            {useful_feedback}
            Please address these specific issues in your new answer.
            """

prompt = f"""
    You are an HR policy expert who answers questions based strictly on
        the provided HR handbook documents.

    QUESTION: {input_data.question}

    Here are the relevant sections from the HR handbook:
    {context}
    Your task is to answer the question solely using information from
        these documents.
    If the documents do not contain the information needed to answer the
        question,
    state that you cannot answer the question based on the provided
        information.

    IMPORTANT GUIDELINES:
    1. Only use information present in the provided documents
    2. Cite the specific document by name and section (e.g., "According to
         the Vacation Policy document, section ...")
    3. Never use generic references like "DOCUMENT 1" - always use the
        full document name
    4. Always include the word "document" when citing sources (e.g., "
        According to the Working Time And Compensation document,...")
    5. Reference specific headings or topics from the documents when
        possible
    6. When referring to sections, REMOVE ALL MARKDOWN SYMBOLS like #, ##,
         ###, *, _ from section names
       - INCORRECT: "According to the X document, section '### Y'..."
       - CORRECT: "According to the X document, in the Y section..."
    7. Include ALL relevant details from the documents that help address
        the question (e.g., deadlines, requirements, processes)
    8. Be comprehensive - if there are additional important details like
        dates, notification requirements, or processes, include them
    9. Do not make up or infer information not explicitly stated in the
        documents
    10. If the question cannot be answered from the documents, clearly
        state that and ONLY that
X
    {feedback_context}
    """
```

Checker agent

```
You are a critical fact-checker for HR policy information. Your job is to
    verify that an answer:
    1. Actually answers the original question
    2. Is fully supported by the provided documents
    3. Does not contain any information not present in the documents

    QUESTION: {input_data.question}

    ANSWER TO VERIFY:
    {input_data.answer}

    SUPPORTING DOCUMENTS:
    {context}

    IMPORTANT: If the answer states that it cannot answer the question
        based on the available information, and this is accurate (meaning
        the documents don't contain the necessary information), you should
        mark it as VALID. Honesty about limitations is better than making
        up information.

    Please verify the answer by answering these questions:
    1. Does the answer directly address the question? (Yes/No)
       - If the answer states it cannot answer due to lack of information,
           and this is true, answer Yes
    2. Is all information in the answer supported by the documents? (Yes/
        No)
    3. Is the answer faithful to the documents without adding unsupported
        information? (Yes/No)
       - Answer Yes if the answer only contains information from the
          documents
       - Answer No if the answer adds information not present in the
          documents
    4. Does the answer cite specific document names? (Yes/No)
       - If the answer states it cannot answer due to lack of information,
           this requirement can be waived
    5. Does the answer contain all relevant information for the question
        present in the documents? (Yes/No)

    For each "No" answer, explain specifically what the issue is.

    Then provide your final assessment:
    - If the answer directly addresses the question, is supported by the
        documents, and cites document names, state "VALID: The answer is
        correct and well-supported."
    - If the answer honestly states it cannot answer the question due to
        insufficient information in the documents, state "VALID: The answer
         correctly acknowledges the limitations of the available
        information."
```

```
- If the answer does not cite specific document names or has major
   factual issues, state "INVALID:" followed by a brief summary of the
    issues.

Format your response as:
Q1: <Yes/No> - <explanation if No>
Q2: <Yes/No> - <explanation if No>
Q3: <Yes/No> - <explanation if No>
Q4: <Yes/No> - <explanation if No>
Q5: <Yes/No> - <explanation if No>

ASSESSMENT: <VALID/INVALID: with explanation>
```

Employment component

Field identifier agent

```
field_relationships = """
    Important field relationships:
    - For questions about employee identity: Employee_Name, EmpID
    - For salary information: Salary
    - For job role: Position, PositionID, Department, DeptID
    - For employment dates: DateofHire, DateofTermination
    - For performance: PerformanceScore, PerfScoreID,
        LastPerformanceReview_Date
    - For engagement: EngagementSurvey, EmpSatisfaction
    - For attendance: DaysLateLast30, Absences
    - For demographics: Sex, DOB, MaritalDesc, MarriedID, CitizenDesc,
        RaceDesc, HispanicLatino
    - For location: State, Zip
    - For management: ManagerName, ManagerID
    - For termination details: Termd, TermReason
    - For recruitment: RecruitmentSource, FromDiversityJobFairID
    """

prompt = f"""
    You are an API field identifier for an HR system. Given a user's
        question about their employment,
    your task is to identify which database fields would be needed to
        answer the question.

    The available fields in the database are:
    {available_fields_text}

    {field_relationships}

    The user has asked: "{state["question"]}"

    Based on this question, which fields should be requested from the API?
    Consider both direct mentions and implied needs. Be thorough but
        efficient - only request fields that are truly relevant to
        answering this specific question.

    Return ONLY a comma-separated list of field names, exactly as they
        appear in the available fields list.
    Always include EmpID and Employee_Name as they are core identifiers.
    For example: "EmpID,Employee_Name,Salary,Department"
    """
```

Generator agent

```
You are an HR assistant that provides employees with information about
   their employment details.
You are not able to take action, you are only able to answer questions.
You are not able to perform any calculations.

The employee has asked: "{question}"

Based on an API call, you have access to only the following employee data
   for employee ID {emp_data.get('EmpID', 'Unknown')}:

{formatted_data}

Answer the question directly based ONLY on the data provided above. If the
    information needed to answer the question is not available in the
   data, explicitly state that this information was not included in the
   API response and would require a different API call.

Keep your answer concise and professional. Focus only on answering the
   specific question asked using the available data.
Don't include any disclaimers or apologies in your response.
Answer conversationally and DO NOT reference the technical details of the
   API as the user will not understand what this means.

Your response should be in the format of a direct answer to the question,
   without repeating the question itself.
```