Transforming Privacy Artifacts into Accessible Reports for Non-Technical Stakeholders

Preprint, compiled June 10, 2026

Zoe Pfister[1*], **Clemens Sauerwein**[1], **Benedikt Dornauer**[1], **Tina Mersch**[2], **Christian Wolf**[2], **Ruth Breu**[1], **and Michael Vierhauser**[1]

[1]University of Innsbruck, Department of Computer Science, Austria
[2]EKS InTec, Germany

Abstract

The transition toward Industry 5.0 is reshaping industrial work environments with an emphasis on human-centricity, enabling close collaboration between humans and machines to enhance productivity and flexibility. However, such systems typically require monitoring of human workers and operators, often involving sensitive data, raising significant privacy concerns. As a result, affected workers and unions frequently reject human-machine collaboration features due to a lack of transparency regarding privacy threats and implemented mitigation strategies. To enable early stakeholder involvement, establish trust, and support informed decision-making, privacy implications must be communicated in a way understandable to non-technical stakeholders. Yet, current Requirements Engineering (RE) practices provide limited methodological support for making privacy threats and mitigations accessible to non-technical stakeholders (e.g., individual workers or their representative unions). In this paper, we propose a conceptual framework that guides software design from human monitoring-related use cases and requirements to informed decision-making guidance focusing on non-technical stakeholders. Building on principles such as Privacy by Design, the framework leverages Large Language Models (LLMs) to transform technical artifacts into accessible privacy reports. We share initial insights from two industry use cases, evaluate the quality of the generated reports, and outline future research directions toward integrating privacy transparency into RE processes for human-centric industrial systems.

1 Introduction

The transition toward Industry 5.0 emphasizes human-centricity [1] and promotes close collaboration between humans and machines. In particular, these interaction points require stakeholder-centered Cyber-Physical Systems (CPS) engineering and operation [2]. The overarching objective is to enhance productivity while simultaneously increasing flexibility [3]. To ensure correct and safe operation, as well as enable self-adaptation during collaborative tasks, systems must implement monitors to collect and analyse diverse runtime information [4, 5, 6]. In scenarios involving human collaboration, monitored data is not solely limited to pure machine-related data, but also includes information about human operators and workers [7, 8]. For example, consider a mobile robot carrying heavy tooling while following a worker [9]. To operate safely, the robot must track both the operator and nearby humans to prevent collisions. We refer to requirements that involve the collection and processing of data about human stakeholder activities as *human monitoring requirements*.

However, monitoring humans at runtime inevitably raises the issue of collecting potentially sensitive information, posing significant privacy concerns, potentially hampering, or even preventing the acceptance and implementation of collaborative features altogether. In the past, several cases of extensive workplace surveillance and misuse have resulted in pushback, public backlash, and regulatory scrutiny. For example, Amazon France Logistique was fined 32 million euros in 2023 for intrusive employee monitoring that violated GDPR principles [10]. In a similar case in 2020, H&M was fined 35 million euros for monitoring several hundred employees at a service centre in Germany [11]. These and many other examples highlight how privacy concerns can hinder the acceptance of monitoring-enabled systems and fuel opposition from workers and unions.

From a requirements perspective, human monitoring requirements often face significant criticism and resistance from worker unions, employee representatives, and privacy advocates. Monitoring practices are frequently regarded as intrusive and ethically problematic, with workers and their representatives viewing such technologies with suspicion due to concerns about privacy violations, job quality, reduced autonomy, and employee dignity [12, 13]. In many cases, collective bodies and unions frequently argue that such systems should not be implemented without transparent safeguards, informed consent, clear documentation of purpose, and effective mitigation strategies to address privacy risks [14, 15]. In addition, compliance with regulatory frameworks, such as the GDPR, introduces further constraints during system design. While several researchers have investigated how the GDPR and similar regulations can be integrated and aligned with Requirements Engineering (RE) processes or how privacy-related requirements can be extracted [16, 17, 18, 19], current RE practices still lack structured processes that bridge privacy design artifacts and informed decision-making by non-technical stakeholders. As a result, privacy threats, trade-offs, and mitigation strategies must be communi-

Accepted for publication at the 34th IEEE International Requirements Engineering Conference (RE@Next'26).
correspondence: Zoe.Pfister@uibk.ac.at

cated in a way accessible to non-technical stakeholders, including workers and their representatives, to increase transparency and support technology acceptance.

In this paper, we address this challenge by integrating privacy transparency considerations into Software Engineering (SE) and RE processes. We develop a first iteration of a framework that guides software design from human monitoring requirements to informed decision-making guidance by workers and their representatives. Specifically, we employ established security and privacy analysis techniques, such as Data Flow Diagrams (DFD) and the STRIDE framework [20], and provide these artifacts as input to Large Language Model (LLM) agents. More specifically, several agents transform the technical artifacts – together with the use cases and requirements – into accessible privacy reports. With this approach, we intend to increase involvement and trust of non-technical stakeholders.

In this early research, we aim to pursue the following research objectives (ROs):

RO1: Develop a structured RE process that supports privacy analysis and communication of privacy threats and mitigation tactics related to human monitoring requirements, thereby facilitating broader understanding and technology acceptance among stakeholders.

RO2: Investigate how LLM-assisted transformations of technical privacy artifacts can improve the accessibility and understandability of privacy threats and mitigation strategies for non-technical stakeholders.

Our contributions in this paper are threefold: First, we present a structured process that links human monitoring requirements, privacy design artifacts, and use cases to decision-making processes in worker unions. Second, we propose an approach to increase technology acceptance in Industry 5.0 by making privacy threats and their mitigations more accessible through LLM-assisted reporting. This includes generating human-readable reports that explain the privacy implications of human monitoring requirements and planned mitigations. Third, we report on a preliminary validation, gaining insights through a questionnaire and feedback session with our industry collaborator, and outline directions for future research areas.

The remainder of this paper is laid out as follows: Section 2 presents background, existing research, and our motivating example. In Section 3, we introduce our framework and process steps. We apply the framework using the motivating example in Section 4. Section 5 presents a proof-of-concept validation using two industry use cases. Finally, Sections 6 and 7 outline our research roadmap and conclusions.

Data availability: We provide all supplementary material as part of a repository: https://github.com/Ethical-Human-Machine-Interaction/PrivacyArtifacts2Report.

2 BACKGROUND & RELATED WORK

Privacy in industrial CPS concerns how workers' personal data is collected, processed, and shared. In monitoring-enabled environments, such data may include behavioural, positional, or performance-related information about workers. Regulatory frameworks such as the GDPR treat privacy as a design concern by requiring data protection measures and transparent, accessible information for data subjects [21].

Security and Privacy Analysis Techniques: In Software Engineering, Data Flow Diagrams are commonly used to visualize how data moves through a system by modelling processes, data stores, external entities, data flows, and trust boundaries [22]. DFDs also form the basis for STRIDE-based threat modelling. STRIDE is a widely used approach for systematically identifying security threats during system design [23, 24]. It categorises threats into six classes: **S**poofing, **T**ampering, **R**epudiation, **I**nformation Disclosure, **D**enial of Service, and **E**levation of Privilege. STRIDE analyses each element of a DFD for potential threats within these categories. This structured approach helps developers identify security risks early in the design phase and derive appropriate mitigations before implementation. Since STRIDE explicitly addresses information disclosure threats, which are particularly relevant in privacy-sensitive systems [23], DFDs combined with STRIDE provide a link between GDPR-oriented privacy analysis and structured threat identification.

LLMs/Agents supporting RE Activities: LLMs are transformer models [25] trained on large text corpora and designed to generate plausible continuations of textual prompts [26]. One key aspect is their ability to perform "in-context learning", i.e., adopting new tasks at runtime based on natural language instructions (prompts) [26]. This flexibility has led to increasing interest in using LLMs to support various activities in the SE lifecycle [27, 28, 29]. However, LLMs often act as black boxes, lacking transparency (e.g., through hallucinations) and controllability, especially in complex tasks [30]. This limitation is particularly relevant when generating privacy-related artifacts for stakeholders, where each artifact must be rigorously validated and, if necessary, edited by a human to ensure accuracy and correctness. To address this problem, Wu et al. [26] propose *Chaining LLM steps together*, where a complex problem is decomposed into multiple sub-problems executed sequentially. Each step receives inputs from previous steps of the chain. Naturally, humans can analyse the intermediate outputs and edit them should they detect incorrect statements.

Another important factor influencing output quality is prompt formulation. For example, Brown et al. [31] have shown that few-shot prompting, where the prompt includes examples of the task to be completed, can significantly improve model performance compared to zero-shot prompts. A more recent approach is *chain-of-thought prompting* [32], where the input prompt contains chain-of-thought examples of the task to solve. This increased performance on arithmetic and complex symbolic reasoning tasks, among others. Additional work has also explored template-based prompting approaches for structuring LLM interactions in specific domains [33].

To better understand what monitoring data is collected, and how privacy may be affected, we enrich traditional use case templates (e.g., goals, scenarios) with monitoring aspects. The *"monitoring use cases"* [34] describe what data is collected during monitoring, what equipment is used for collection, and who the stakeholders being monitored are. These monitoring use cases can then be further enriched with information for a GDPR Data Privacy Impact Assessment (DPIA) [21, 35].

Motivating Example: Our motivating example is drawn from a use case defined by our industry collaborators EKS InTec

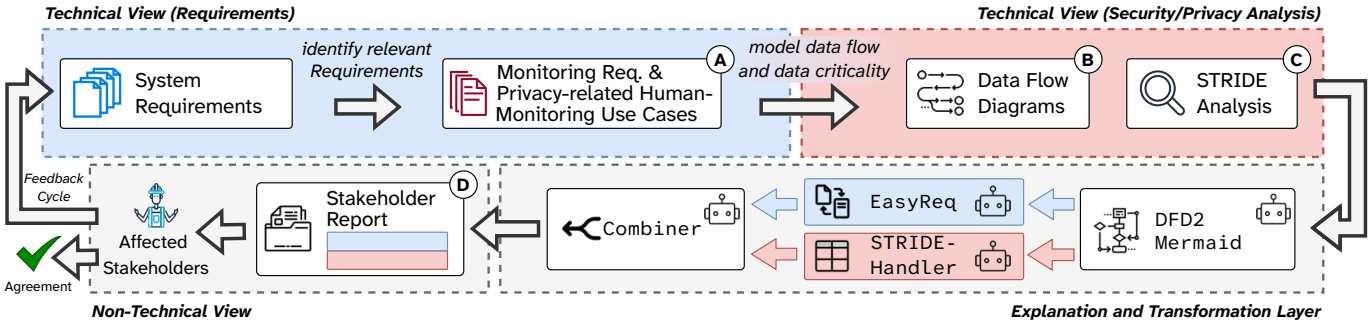

Figure 1: Overview of our proposed framework, transforming human monitoring requirements and privacy analysis artifacts into stakeholder-oriented privacy reports.

(*InLine Control of Product Assembly*). The goal is to improve product quality during assembly. To achieve this, the system uses camera-based tracking to monitor the assembly process and identify deviations or faulty components. When such deviations are detected, the system notifies shop-floor workers in real time, temporarily pausing the assembly process until the error is corrected. While the primary functionality of the use case revolves around quality assurance in the assembly process, it necessitates continuous monitoring of workers. This introduces several privacy concerns related to the collection of video footage and behavioural information of shop-floor workers. Without systematic and rigorous privacy modelling and the translation of these technical specifications into transparent privacy reports, the system risks both non-compliance with GDPR and opposition from worker unions due to the sensitive nature of the captured video and behavioural data.

3 Towards a Privacy Transparency Framework for Human Monitoring Use Cases

Our framework (cf. Fig. 1) builds upon established software engineering artifacts. System engineers identify use cases and requirements that involve monitoring of human stakeholders. These (A) *human monitoring requirements* and *use cases* serve as the starting point for subsequent privacy analysis and stakeholder communication activities. In the following, we describe the framework and its core components.

3.1 Framework Overview

After identifying relevant monitoring use cases and requirements, the framework applies established security and privacy analysis techniques. Specifically, this involves constructing (B) a Data Flow Diagram and (C) performing a STRIDE analysis. These artifacts support the systematic identification of potential privacy risks affecting stakeholders and the development of planned mitigation strategies to address them. Together with the human monitoring requirements and use cases, these artifacts can also support the preparation of a GDPR-mandated Data Protection Impact Assessment (DPIA) [21]. For example, the artifacts align well with the *DPIA Data Wheel* [35], which maps questions about data processing activities to GDPR principles, facilitating regulatory compliance.

The use case, associated requirements, Data Flow Diagram, and STRIDE analysis then serve as input to the LLM Agents within the *Explanation and Transformation Layer*. Within this layer, the complex technical artifacts are transformed into a (D) *privacy report tailored to non-technical audiences*. The resulting reports are presented to stakeholders involved in the decision-making process of accepting or rejecting the proposed privacy-related requirements (e.g., shop-floor workers or worker unions). The goal is to facilitate informed stakeholder discussions and support consensus-building regarding the acceptability of the proposed solutions. Based on feedback, the process either results in formal agreement on privacy requirements or triggers an iterative feedback cycle in which the use case and requirements can be revised and adapted.

3.2 Explanation and Transformation Layer

The core component of the framework is the *Explanation and Transformation Layer*, which bridges the gap between technical artifacts and the perspectives of non-technical stakeholders. It receives the monitoring use case, its associated requirements, DFD, and the STRIDE analysis, and transforms them into a structured privacy report. This transformation is performed by a workflow of four specialized LLM agents:

First, the `DFD2Mermaid Agent` converts a DFD image into a machine-readable Mermaid[2] diagram and generates a textual summary describing its components, data flows, and trust boundaries. This serves two purposes: reducing the number of tokens required to process the graph, and providing a human-readable representation of the model, allowing a human reviewer to validate the LLM's interpretation of the data flow and correct errors early in the process.

The conversion of technical artifacts into stakeholder-oriented explanations is handled by the `EasyReq Agent` and the `STRIDE-Handler Agent`. The latter receives the DFD Mermaid diagram and summary of the `DFD2Mermaid Agent`, along with the original requirement and the STRIDE analysis, as input. It iterates over the STRIDE entries, summarizing each identified threat in language comprehensible to non-technical stakeholders. The agent also enriches the summary with concrete privacy threat examples relevant to the given use case and explains how each identified threat is intended to be mitigated. In parallel, the `EasyReq Agent` rewrites the original use case and requirements into a simplified version suitable for non-technical stakeholders. Further, it generates a short rationale explaining the purpose, intended benefit, and implementation motivation of each requirement (cf. Fig. 3).

[2]See https://mermaid.ai.

Both agents follow explicit output guidelines, as specified in their respective input prompts. They are explicitly instructed to avoid technical terminology and instead explain concepts using simple terms, while keeping the output concise. Further, the `EasyReq Agent` is instructed to focus on the *what* and *why* of the use case rather than the *how*, while the `STRIDE-Handler Agent` illustrates threats using concrete examples and addresses privacy and surveillance-related concerns. Finally, the `Combiner Agent` combines the outputs of all preceding agents into a structured HTML privacy report.

4 PROTOTYPE IMPLEMENTATION AND USAGE EXAMPLE

We implemented an initial research prototype of the Explanation and Transformation Layer workflow using *n8n* [36], a workflow automation platform for orchestrating agents. In future iterations, we plan to further extend and implement the workflow using LangChain[3] to allow for more flexible agent coordination and integration with additional LLMs.

The `DFD2Mermaid Agent` uses Gemini 2.5 Pro, which consistently produced valid conversions from DFD images into Mermaid representations during preliminary experiments, in contrast to other models. The remaining agents use Claude Sonnet 4.5. Extended reasoning ("Thinking Mode") is only enabled for the `STRIDE-Handler Agent`, where more complex reasoning is required to interpret and summarize STRIDE analysis results. Prompts were initially designed manually using Claude Console following the RICE template (Role, Instructions, Context, Constraints, Examples) [28]. We then iteratively refined these prompts using Claude Sonnet 4.5 to conform to the Claude Prompting Best Practices [37, 38], e.g., structuring inputs via XML tags. Additionally, for both the `EasyReq Agent` and `DFD2Mermaid Agent`, we included a scratchpad section to the prompt, outlining a list of steps to follow during text generation (cf. Figure 2), facilitating the model's chain-of-thought reasoning capabilities [32].

As this implementation serves as an initial end-to-end proof of concept, we intend to compare the current solution with other LLM providers, such as ChatGPT, or Le Chat[4] as part of future work (cf. Section 6).

Using the Framework: We demonstrate our framework using the motivating example *InLine Control of Product Assembly* introduced in Section 2. The workflow begins when a system engineer uploads design artifacts: (A) the use case (including goals, scenarios, and monitoring properties) and associated requirements; (B) a Data Flow Diagram; and (C) the results of a STRIDE analysis. A partial example is shown on the left side of Fig. 3. In this example, the DFD models how camera data flows from the sensor to an Edge Device Data Processor, which analyses the footage for assembly errors and forwards detected violations to a Cloud Platform (annotated as (1) in Fig. 3). Applying STRIDE to this data flow reveals a potential information disclosure threat, where unauthorized individuals could gain access to raw camera data if the data flow is insufficiently protected. A possible mitigation in this scenario is encryption. Since the final report focuses on privacy aspects, we scoped the STRIDE analysis to the most relevant threat categories for privacy, namely *Information Disclosure*, *Spoofing*, and *Tampering*.

[3] See https://www.langchain.com.

[4] See https://chatgpt.com and https://chat.mistral.ai/chat respectively.

> You are a security engineer with expertise in translating technical security information for non-technical stakeholders.
> Your task is to take a technical software security requirement and its use case, then rewrite the requirement in clear, accessible language that non-technical stakeholders, especially worker unions, can understand.
> You should also explain why this security requirement is important and planned.
>
> [input-documents]
>
> Your goal is to:
>
> 1. Rewrite the requirement in plain, non-technical language that avoids jargon, acronyms, and technical terminology
> 2. Explain the value and rationale for why this requirement is planned
>
> [guidelines for non-technical communication]
>
> Before writing your final output, use the scratchpad to:
>
> 1. Identify the core purpose of the technical requirement
> 2. Note any technical terms that need to be simplified or explained
> 3. Think about what business risks this requirement mitigates
> 4. Consider what stakeholders care about most
>
> <scratchpad>[Your analysis here]</scratchpad>
> [examples]

Figure 2: Single-Shot Prompt Template with Chain-of-Thought Scratchpad used for Simplifying Requirements and Use Cases for Non-technical Stakeholders. Used by the `EasyReq Agent`.

Based on these artifacts, the workflow transforms the data for non-technical stakeholders. The DFD is first converted to Mermaid syntax and summarized in natural language. The resulting output, together with the STRIDE analysis, the use case, and the requirements, is then provided as input to the `STRIDE-Handler Agent`, which generates structured threat explanations. An example is shown on the bottom-right of Fig. 3, where the information disclosure threat is converted into structured prose:

1. *Plain-language threat description (what):* The threat is described in accessible language, supported by illustrative examples for better clarification. In the example, the agent explains that unauthorized individuals could access raw camera footage.
2. *Explanation of stakeholder impact (why):* The potential consequences are described from the perspective of the stakeholders (e.g., shop-floor workers in the motivating example). The explanation highlights that unauthorized access to camera footage is a clear privacy violation and could be misused for surveillance beyond the intended purpose of quality control.
3. *Mitigation explanation (how):* The proposed mitigation strategy is discussed. In the example, encryption is described as making the data unreadable to anyone without proper authorization.

Each STRIDE threat is transformed following this structure.

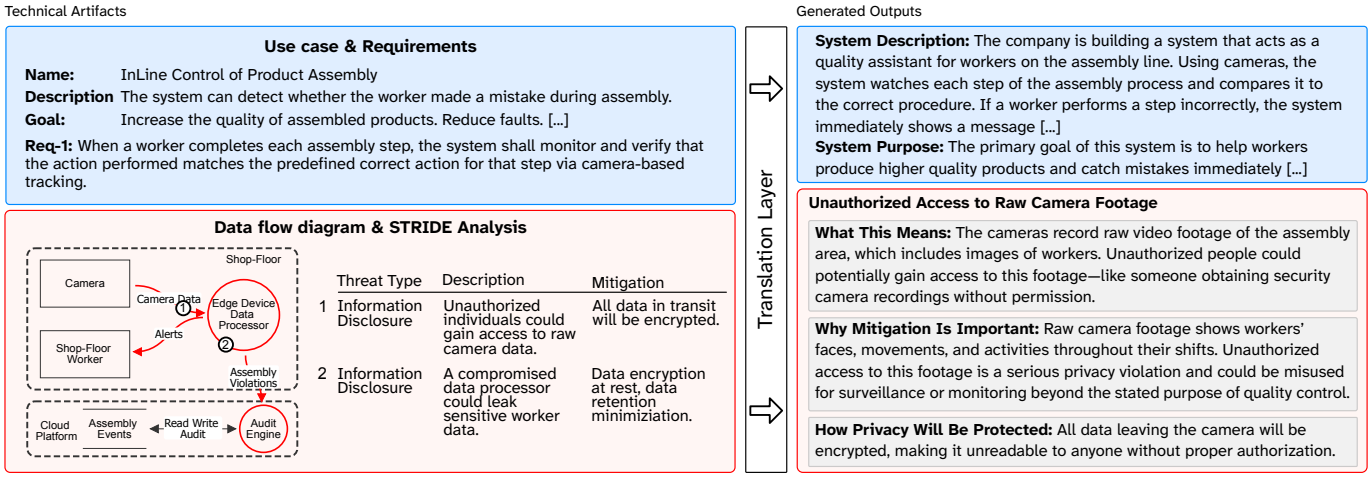

Figure 3: Example of Inputs and Generated Outputs of the Workflow based on the example (UC1) introduced in Section 2.

Next, the *EasyReq Agent* simplifies the original use case and requirements, producing a natural-language version and a rationale explaining the intended purpose/benefit of each requirement (cf. top-right side of Fig. 3).

Finally, the *Combiner Agent* integrates both outputs, generating the final report (D) in HTML format. The report begins with an executive summary, followed by a simplified system description and purpose statement produced by the EasyReq Agent. Next, the report lists the privacy threat explanations, each containing the three-part structure described above. This ordering ensures that a non-technical reader first understands what the system does and why it is being built before being introduced to potential privacy risks and mitigation strategies.

5 PRELIMINARY VALIDATION

The goals of our approach are to develop a structured RE process that supports privacy analysis and communication to non-technical stakeholders (RO1) and to automatically generate an easily understandable privacy report based on a set of (technical) requirements and relevant standards and guidelines (RO2). Through this preliminary validation, we aim to obtain initial insights into the quality of our generated reports and the feasibility of our proposed process and workflow. We, therefore, focus on two aspects: First, we demonstrate the feasibility of our proposed workflow using our end-to-end prototype applied to two real-world use cases (cf. Fig. 3) from our industry collaboration (cf. Section 5.2). Second, we assess the quality of the generated reports using a combination of Software Requirements Specification (SRS) metrics and qualitative expert feedback through a survey and semi-structured interview with a domain expert from our industry collaborators (cf. Section 5.3). The full set of interview questions and results can be found in the supplementary material.

5.1 *Validation Setup*

We were provided with two use cases by our industry collaborators: (UC1) *InLine Control of Product Assembly*, monitoring workers to detect assembly errors and sequence violations, and (UC2) *Cycle Time Monitoring for Lean Production*, tracking worker assembly durations to identify process inefficiencies. In both cases, requirements include camera-based monitoring of shop-floor workers, making them suitable candidates for privacy threat analysis and subsequent stakeholder-oriented reporting.

To assess report quality, we designed a questionnaire based on established document quality criteria. Since we are not aware of existing frameworks to evaluate non-technical privacy reports, we base our validation criteria on those proposed by Krishna et al. [27], who evaluate LLM-generated Software Requirements Specifications (SRS). Specifically, we collected six metrics: (1) internal consistency, (2) redundancy, (3) completeness with respect to technical artifacts, (4) conciseness, (5) correctness, and (6) understandability on 5-point Likert scales. These criteria capture general aspects of document quality and provide a suitable baseline for evaluating structured privacy reports. After completing the questionnaire, the expert participated in a semi-structured interview to provide qualitative feedback on report structure and understandability.

The domain expert was provided with two reports: one manually created report – created by one author, and reviewed by two additional authors – and one automatically generated using our prototype implementation. The reports were presented in a blinded manner, i.e., the expert was not informed which report was generated by the framework. After answering the questions concerning the 6 quality metrics, the domain expert participated in a follow-up feedback session, where we conducted a semi-structured interview to gain more information on how to improve the report's structure and understandability in future work.

5.2 *End-To-End Report Generation*

For each of the two use cases and their requirements, we manually created a Data Flow Diagram and conducted a STRIDE analysis, focusing on the privacy-relevant threat categories of Information Disclosure, Spoofing, and Tampering. Based on these artifacts, we generated reports using our workflow and manually created baseline reports for comparison.

Repeated executions of our workflow revealed minor inconsistencies in the generated output, confirming the non-deterministic nature of LLM-based workflows. For example, section titles for privacy violations were sometimes derived directly from threat titles of the STRIDE analysis, which might be unclear to

non-technical stakeholders. We also observed minor formatting variations (bullet points vs. prose) across several workflow runs. To reduce visual bias, both manual and generated reports were formatted uniformly before being shared with our expert.

5.3 *Expert Feedback*

Quantitative Results: Initial feedback from our expert indicates that the generated reports matched or outperformed the manually created baseline reports across all evaluated quality dimensions. The largest differences were observed in the completeness, conciseness, and understandability measures for UC1, where the generated report scored two points higher (on a 5-point scale). The internal consistency varied between the individual UCs, with the generated report scoring a 3 for UC1 and 5 for UC2. Related to understandability, the expert perceived the generated reports as more accessible from a language standpoint compared to the manually written ones. However, the generated reports introduced other issues, such as undefined abbreviations (e.g., LTPE for Lean Time Processing Engine), highlighting the need for human validation of generated outputs.

Qualitative Results: The follow-up interview provided additional insights into how the reports' understandability and structure could be further improved with respect to RO2. Currently, each STRIDE threat is presented as a separate section in the report. However, the expert pointed out that this structure can result in perceived redundancy, making it difficult to understand that the sections actually describe different parts of the data flow. For example, multiple threats in UC1 used encryption as their primary mitigation tactic (e.g., camera to data processor to cloud platform). Grouping such related threats and explaining them along the data flow could improve readability and reduce redundancy, making the report more concise. The expert indicated that restructuring reports to reflect data flow dependencies and shared mitigation strategies complemented by a (simplified) visualization of the DFD (cf. Fig. 3), could further increase both conciseness and stakeholder comprehension. However, they highlighted that there is a trade-off, and care must be taken, ensuring that the report does not become too technical. These findings provide initial evidence for RO2, while also highlighting areas for improvement in report structuring and presentation.

The expert further suggested enhancing the report with a more comprehensive introduction, describing the system in greater detail, as well as a concluding section summarizing key insights, open issues, and areas requiring stakeholder discussion. Additionally, introducing prioritization, e.g., using the MoSCoW principle (must, should, could, won't) [39], may serve as a basis for stakeholder deliberation.

5.4 *Threats to Validity*

Our initial proof-of-concept demonstrates the feasibility and successful application of our framework and the generated reports. However, while the preliminary validation is based on a limited number of use cases, they represent real-world scenarios from our industry collaborators. To address this limitation, we are currently collaborating with industry partners to extend the set of use cases and apply the framework to more complex and diverse scenarios. In the initial prototype implementation and preliminary validation, we primarily used Claude Sonnet 4.5 (due to availability) for our agents. In future work, we plan to evaluate our framework using diverse LLMs (cf. Section 6). Despite this limitation, Claude Sonnet 4.5 produced promising results in the generated reports.

Finally, the interview revealed that inconsistencies within the input artifacts themselves propagate into the generated reports. For example, one artifact proposed encryption as a mitigation against threats where the adversary potentially holds the decryption keys. Since the LLM cannot produce internally consistent output from contradictory inputs, such issues carry over into the final reports. This highlights a fundamental limitation of the approach: the quality of the generated privacy report is bound by the quality and consistency of the initial technical artifacts, making rigorous validation of input documents a prerequisite for meaningful report generation.

6 DISCUSSION & RESEARCH ROADMAP

Our initial validation, preliminary results, and expert feedback indicate that we are able to partially achieve our research objectives: our early version of the framework demonstrates how privacy analysis can be systematically integrated into early requirements engineering processes (RO1), and how structured, stakeholder-oriented privacy reports can be generated from technical artifacts (RO2). Building on these results, we identify five research directions for our future work.

Integration of diverse Privacy Analysis Frameworks: In our initial prototype, we employed the STRIDE framework [20] for systematically collecting and documenting privacy-relevant threats and corresponding mitigations. However, several alternative frameworks exist, and security and privacy researchers have developed various approaches for specifically analysing privacy risks, such as LINDDUN [40], the DPIA Data Wheel framework [35], or STPA-Priv [41]. Given our modular architecture, these can be easily integrated as new agents. As part of RO1 (cf. Section 1) – extending our structured process – we will analyse other assessment frameworks, investigate the strengths and limitations, and evaluate their suitability in early requirements elicitation processes to improve technology acceptance. Furthermore, we plan to develop an ontology, relating concepts across frameworks (threats, assets, controls, privacy goals). From this, we aim to derive recommendations and guidelines for selecting a framework based on project characteristics and the specific nature of human monitoring involved.

Mapping to Regulatory Texts: Currently, input artifacts do not explicitly link identified threats and mitigations to relevant regulatory provisions, such as the GDPR [21]. This lack of traceability limits the applicability of the generated reports in compliance verification (e.g., GDPR Data Protection Impact Assessments). To address this limitation, we plan to enrich privacy threats and mitigation information with explicit mappings to relevant regulatory texts, including, for example, specific articles and recitals of the GDPR. This could be achieved through the integration of Retrieval Augmented Generation (RAG) [42] context pipelines supplying the agents with relevant regulatory passages when processing STRIDE artifacts, or through the integration of external knowledge sources via the Model Context Protocol (MCP)[5]. For example, mitigation tactics, such as data minimisation (cf. Fig. 3), could be linked to Article 5(c) of the

[5] See https://modelcontextprotocol.io.

GDPR [21]. Such additional augmentations would enable the generation of specialized reports tailored to legal stakeholders and support compliance verification, further extending RO2.

Agent-Augmented Support: Currently, our workflow relies on manually created design artifacts as input for report generation. In our ongoing work, we intend to extend the framework towards increased automation through LLM-supported artifact generation. This includes using custom agents that generate requirements from use cases, or conduct an automatic STRIDE analysis with tools such as STRIDE GPT[6]. Another challenging topic for future work is the automated generation of DFDs from software requirements artifacts, where the LLM must make sound architectural decisions to satisfy all system requirements. While such automation has the potential to reduce modelling effort, it also shifts the burden towards validation. This is especially critical for privacy and security aspects of SE, where incorrect threat assessments or missing mitigations may lead to non-compliance with the law, making rigorous human-in-the-loop validation essential. As part of this, we will investigate perceived accuracy and completeness of generated artifacts from the perspective of security professionals, as well as the associated cognitive load, for example, using established methods such as NASA-TLX.

Persona-specific Reports: Stakeholders involved in the decision-making process differ substantially in their technical expertise and informational needs. For example, worker representatives may primarily be interested in understanding what data is collected and the privacy implications, whereas legal experts require detailed mappings to regulatory provisions to evaluate compliance. Currently, our framework generates a single report targeting non-technical stakeholders. Future work will introduce stakeholder persona-specific report generation to tailor reports to the specific concerns and expectations of different stakeholder groups, supporting RO2. Persona-based customisation can be achieved by adapting LLM agent roles and prompt configurations. A key research challenge, therefore, is identifying which information is relevant for each persona, ensuring that simplification of the content does not compromise its completeness or correctness (cf. Section 5).

Comprehensive evaluation: Our preliminary validation provides initial evidence that the LLM-generated privacy reports created via our framework accurately reflect relevant requirements and privacy analysis artifacts while improving accessibility to non-technical stakeholders. However, a more comprehensive evaluation is required to fully address RO1 and RO2. First, we plan to conduct an interview study with non-technical stakeholders and legal experts to better understand their expectations regarding privacy reports. These insights will then inform the second iteration of our privacy report generation process. Additionally, we plan to conduct a comparative evaluation of LLM-generated and manually created reports with both technical and non-technical stakeholders. Based on the feedback received from our expert interview, we will also explore automated validation methods (e.g., round-trip reconstruction of reports and technical artifacts, LLM-as-a-Judge [43]) to identify information loss or inconsistencies in generated reports and reduce manual validation efforts.

[6] See https://github.com/mrwadams/stride-gpt.

7 CONCLUSION

In this paper, we presented a Requirements Engineering framework that integrates privacy threat analysis with automated, non-technical stakeholder-oriented report generation. By combining established security analysis techniques, such as STRIDE and Data Flow Diagrams, with a modular LLM-based transformation pipeline, the approach enables the systematic derivation of structured, non-technical privacy reports from technical Software and Requirements Engineering artifacts. Our preliminary validation demonstrates both the feasibility of the approach and its potential in future work. At the same time, the results highlight important challenges, including the consistency of input artifacts, the need for improved report structuring, and the importance of human validation in the loop. Overall, the findings suggest that LLM-supported transformations can serve as an effective mechanism to operationalize privacy-by-design principles within requirements engineering processes. As part of our research roadmap and ongoing work, we will extend the framework to incorporate additional privacy analysis methods, support artifact creation with LLMs, and conduct an empirical study to further assess the effectiveness and generalizability of our framework.

ACKNOWLEDGMENT

This work was supported by the ITEA 4 project GENIUS, with funding from the Austrian Research Promotion Agency (FFG; grants 931318 and 921454) and the German Federal Ministry of Research, Technology and Space (BMFTR; grant 16IS24069D).

REFERENCES

[1] J. Leng, W. Sha, B. Wang, P. Zheng, C. Zhuang, Q. Liu, T. Wuest, D. Mourtzis, and L. Wang, "Industry 5.0: Prospect and retrospect," *Journal of Manufacturing Systems*, 2022.

[2] K. Michailidis, R. Strazdina, and M. Kirikova, "Continuous requirements engineering for digital transformation.," in *BIR Workshops, CEUR Proc.*, 2021.

[3] T. Kopp, M. Baumgartner, and S. Kinkel, "Success factors for introducing industrial human-robot interaction in practice: An empirically driven framework," *The International Journal of Advanced Manufacturing Technology*, 2021.

[4] X. Zheng, C. Julien, R. Podorozhny, F. Cassez, and T. Rakotoarivelo, "Efficient and scalable runtime monitoring for Cyber-Physical System," *IEEE Systems Journal*, 2016.

[5] A. Stojadinović, N. Stojanović, and L. Stojanović, "Dynamic monitoring for improving worker safety at the workplace: use case from a manufacturing shop floor," in *Proc. of the 9th ACM Int'l Conf. on Distributed Event-Based Systems*, 2015.

[6] M. Vierhauser, J. Cleland-Huang, S. Bayley, T. Krismayer, R. Rabiser, and P. Grünbacher, "Monitoring CPS at runtime-a case study in the UAV domain," in *Proc. of the 44th Euromicro Conf. on Software Engineering and Advanced Applications*, 2018.

[7] A. Buerkle, H. Matharu, A. Al-Yacoub, N. Lohse, T. Bamber, and P. Ferreira, "An adaptive human sensor framework for human–robot collaboration," *The International Journal of Advanced Manufacturing Technology*, 2022.

[8] H. Liu and L. Wang, "An ar-based worker support system for human-robot collaboration," *Procedia Manufacturing*, 2017.

[9] Alex Greenberg – Siemens, "The future of bridging humans, robots, and humanoids with process simulate software." https://blogs.sw.siemens.com/tecnomatix/the-future-of-bridging-humans-robots-and-humanoids-with-process-simulate-software-video, 2025.

[10] European Data Protection Board, "Employee monitoring: French sa fined amazon france logistique eur 32 million." https://www.edpb.europa.eu/news/national-news/2024/employee-monitoring-french-sa-fined-amazon-france-logistique-eu32-million_en, 2024.

[11] European Data Protection Board, "Hamburg commissioner fines h&m 35.3 million euro for data protection violations in service centre." https://www.edpb.europa.eu/news/national-news/2020/hamburg-commissioner-fines-hm-353-million-euro-data-protection-violations_en, 2020.

[12] T. Singh and A. Johnston, "How much is too much: employee monitoring, surveillance, and strain," in *Proc. of the 15th Int'l Conf. on Computational Intelligence and Security*, 2019.

[13] E. Oz, R. Glass, and R. Behling, "Electronic workplace monitoring: what employees think," *Omega*, 1999.

[14] Eurofound, *Employee monitoring and surveillance – The challenges of digitalisation*. Publications Office of the European Union, 2020.

[15] R. Connolly and C. McParland, "Dataveillance: Employee monitoring & information privacy concerns in the workplace," *Journal of Information Technology Research*, 2012.

[16] O. Kosenkov, E. Zabardast, D. Fucci, D. Mendez, and M. Unterkalmsteiner, "Privacy by design: Aligning gdpr and software engineering specifications with a requirements engineering approach," *Information and Software Technology*, 2025.

[17] C. Negri-Ribalta, M. Lombard-Platet, and C. Salinesi, "Understanding the gdpr from a requirements engineering perspective—a systematic mapping study on regulatory data protection requirements," *Requirements Engineering*, 2024.

[18] S. Abualhaija, M. Ceci, N. Sannier, D. Bianculli, S. Lannier, M. Siclari, O. Voordeckers, and S. Tosza, "Llm-assisted extraction of regulatory requirements: A case study on the gdpr," in *Proc. of the IEEE 33rd Int'l Requirements Engineering Conf.*, 2025.

[19] N. Manasreh, P. Spoletini, M. Valero, V. Nino, and I. Sanchez-Cardona, "Designing age-friendly apps: Mining functional, usability, and privacy requirements from existing mobile applications," in *Proc. of the IEEE 33rd Int'l Requirements Engineering Conf. Workshops*, 2025.

[20] L. Conklin, "Threat Modeling Process | OWASP Foundation." https://owasp.org/www-community/Threat_Modeling_Process, 2025.

[21] "Regulation (EU) 2016/679 of the European Parliament and of the Council on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)," 2016.

[22] L. Sion, K. Yskout, D. Van Landuyt, and W. Joosen, "Solution-aware data flow diagrams for security threat modeling," in *Proc. of the 33rd Annual ACM Symp. on Applied Computing*, 2018.

[23] S. Hernan, S. Lambert, T. Ostwald, and A. Shostack, "Threat modeling-uncover security design flaws using the stride approach," *MSDN Magazine-Louisville*, 2006.

[24] A. Shostack, *Threat modeling: Designing for security*. John Wiley & Sons, 2014.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. ukasz Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, 2017.

[26] T. Wu, M. Terry, and C. J. Cai, "AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts," in *Proc. of the 2022 CHI Conf. on Human Factors in Computing Systems*, 2022.

[27] M. Krishna, B. Gaur, A. Verma, and P. Jalote, "Using LLMs in Software Requirements Specifications: An Empirical Evaluation," in *Proc. of the 32nd Int'l Requirements Engineering Conf.*, 2024.

[28] A. Vogelsang, "From Specifications to Prompts: On the Future of Generative Large Language Models in Requirements Engineering," *IEEE Software*, 2024.

[29] X. Hou, Y. Zhao, Y. Liu, Z. Yang, K. Wang, L. Li, X. Luo, D. Lo, J. Grundy, and H. Wang, "Large Language Models for Software Engineering: A Systematic Literature Review," *ACM Trans. Softw. Eng. Methodol.*, 2024.

[30] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu, "A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions," *ACM Transactions on Information Systems*, 2025.

[31] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, and et al., "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems*, 2020.

[32] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. V. Le, and D. Zhou, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," *Advances in Neural Information Processing Systems*, 2022.

[33] Y.-E. Tian, Y.-C. Tang, K.-D. Wang, A.-Z. Yen, and W.-C. Peng, "Template-Based Financial Report Generation in Agentic and Decomposed Information Retrieval," in *Proc. of the 48th Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2025.

[34] Z. Pfister, M. Vierhauser, R. Wohlrab, and R. Breu, "Towards a Value-Complemented Framework for Enabling Human Monitoring in Cyber-Physical Systems," in *Requirements Engineering: Foundation for Software Quality*, 2025.

[35] J. Henriksen-Bulmer, S. Faily, and S. Jeary, "DPIA in Context: Applying DPIA to Assess Privacy Risks of Cyber Physical Systems," *Future Internet*, 2020.

[36] n8n, "AI workflow automation tool." https://n8n.io/, 2026.

[37] ANTHROPIC PBC, "Improve your prompts in the developer console." https://claude.com/blog/prompt-improver, 2024.

[38] ANTHROPIC PBC, "Prompting best practices." https://platform.claude.com/docs/en/build-with-claude/prompt-engineering/claude-prompting-best-practices, 2026.

[39] D. Clegg and R. Barker, *Case Method Fast-Track: A Rad Approach*. Addison-Wesley Longman Publishing Co., Inc., 1994.

[40] K. Wuyts, L. Sion, and W. Joosen, "Linddun go: A lightweight approach to privacy threat modeling," in *Proc. of the 2020 IEEE European Symp. on Security and Privacy Workshops*, 2020.

[41] S. S. Shapiro, "Privacy Risk Analysis Based on System Control Structures: Adapting System-Theoretic Process Analysis for Privacy Engineering," in *2016 IEEE Security and Privacy Workshops*, 2016.

[42] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Advances in Neural Information Processing Systems*, 2020.

[43] D. Li, B. Jiang, L. Huang, A. Beigi, C. Zhao, Z. Tan, A. Bhattacharjee, Y. Jiang, C. Chen, T. Wu, *et al.*, "From generation to judgment: Opportunities and challenges of llm-as-a-judge," in *Proc. of the 2025 Conf. on Empirical Methods in NLP*, 2025.