From Lines of Code to Lines of Policy? Exploring Software Developers' Perceptions of Their Privacy Policy-Related Activities

Ria Prianka Saha
Technical University of Darmstadt
Darmstadt, Germany
saha@ise.tu-darmstadt.de

Daniel Stäcker
Technical University of Darmstadt
Darmstadt, Germany
staecker@ise.tu-darmstadt.de

Jonas Stromberg
Fraunhofer IGD
Darmstadt, Germany
jonas.stromberg@igd.fraunhofer.de

Steven Lamarr Reynolds
Fraunhofer IGD
Darmstadt, Germany
steven.lamarr.reynolds@igd.fraunhofer.de

Kilian Demuth
Technical University of Darmstadt
Darmstadt, Germany
demuth@peasec.tu-darmstadt.de

Frank Nelles
Technical University of Darmstadt
Darmstadt, Germany
nelles@peasec.tu-darmstadt.de

Christian Reuter
Technical University of Darmstadt
Darmstadt, Germany
reuter@peasec.tu-darmstadt.de

Jörn Kohlhammer
Fraunhofer IGD
Darmstadt, Germany
joern.kohlhammer@igd.fraunhofer.de

Alexander Benlian
Technical University of Darmstadt
Darmstadt, Germany
benlian@ise.tu-darmstadt.de

Abstract

Ensuring that privacy policies accurately reflect the real-world behavior of software systems is a complex socio-technical challenge. Despite the importance of privacy policies for user trust and regulatory compliance, prior research has paid limited attention to how software developers perceive their own activities in achieving alignment between privacy policies and actual software behavior and which organizational factors they identify as leading to misalignment. This perspective is particularly important because developers' implementation practices directly shape how software handles personal data. To address this research gap, we conducted 15 semi-structured interviews with software developers directly involved in implementing privacy-relevant functionality. Our findings identify several alignment activities performed by developers and highlight symbolic compliance as an important organizational factor contributing to gaps between documented privacy commitments and actual software behavior. Our study offers implications for researchers investigating privacy challenges in software development and provides valuable insights for organizations and developers seeking to better align privacy policies with actual software behavior.

Keywords

software developer, privacy policy, software behavior, compliance, data protection

1 Introduction

Software developers' foundational coding efforts make it possible that we use digital products every day. These products often collect, process, and share personal data [19]. When personal data is involved, laws require that data handling practices are communicated transparently and in accordance with regulatory standards (e.g., [29]). Regulations such as the California Consumer Privacy Act (CCPA) or the General Data Protection Regulation (GDPR) impose strict requirements for transparency, accountability, and faithful representation of data practices under the threat of substantial fines [7, 11]. However, many organizations struggle to disclose their data handling practices transparently and truthfully to users and regulators. Nearly all studies investigating whether actual data collection and processing practices align with stated privacy policies have revealed misalignments between policy and practice (e.g., [2, 8, 19, 26, 42, 45]).

In organizations, software developers are often responsible for implementing privacy-related tasks within their coding activities [37, 39]. During development, non-compliance can occur, for example, when new third-party components are integrated or when parts of a privacy-compliant system or workflow are reused [1, 14, 39]. The developer's technical implementation in the code can either protect personal data or unintentionally expose it. For example, in one case in Poland, a technical mistake at the implementation layer during a platform migration led to the exposure of students' personal data on the internet [24]. In another case, developers unintentionally stored passwords in plain text instead of securing them, which resulted in the passwords being stolen [24]. These examples illustrate how privacy protections can fail at the implementation layer, where decisions are made directly in the code base. Developers' decisions and practices at this stage can determine whether software behavior complies with privacy requirements and aligns with stated privacy claims, and whether coding practices introduce risks of privacy breaches [9, 12].

Current research has begun to examine the challenges software developers face when performing privacy-related tasks (e.g., [17, 27, 34, 39]). Studies show that developers often struggle to fulfill privacy requirements, and many technical solutions fail to meet compliance standards such as the GDPR [17, 18]. For example, a

Proceedings on Privacy Enhancing Technologies 2026(3), 426–440

https://doi.org/10.56553/popets-2026-0089

majority of professional software developers produced non-GDPR-compliant solutions even when specifically tasked with privacy-sensitive programming [18]. Similarly, another study highlights that developers are under increased scrutiny due to new regulations but often have limited privacy expertise or access to legal resources, which complicates the implementation of privacy concepts throughout the software development lifecycle [28]. Organizational climate also influences how engineers interpret and prioritize privacy concerns, contributing to differences between legal privacy norms and developers' practical interpretations [15, 41].

Despite this growing body of knowledge, there remains a research gap how software developers themselves understand their role in in shaping the relationship between privacy policies and the behavior of the software they implement. This perspective is important because the implementation layer represents the point at which abstract privacy requirements become executable system behavior. Examining developers' own accounts of their work can reveal which activities they associate with creating and maintaining alignment and how organizational conditions shape these practices. To examine these issues, this study addresses the following research questions:

RQ1: *What activities do software developers perceive in their own work as contributing to the alignment between privacy policies and the software's actual behavior?*

RQ2: *How do software developers describe organizational influences that lead to symbolic rather than substantive alignment between privacy policies and the software's actual behavior?*

To address these research questions, we conducted 15 semi-structured interviews with software developers who are directly involved in implementing software code. The study focuses on developers who implement data-related functionality, as this role translates privacy requirements into executable system behavior and vice versa. While other roles in development teams, such as product owners or release engineers, may oversee requirements or release processes, developers are responsible for implementing the technical mechanisms that determine how software handles personal data.

This study offers two significant contributions. First, it provides an in-depth qualitative exploration of software developers' perceptions of privacy policy-data practice alignment, grounded in their direct coding experiences. Second, it identifies symbolic compliance as a significant factor contributing to misalignment between a software's privacy policy and its actual data practices from the developer's vantage point. Our findings offer practical implications for organizations seeking to bridge the gap between their stated privacy commitments and the actual data handling practices of their software, emphasizing the need for targeted support and resources for developers who are at the heart of this critical implementation process.

2 Background and Related Work

2.1 Misalignment Between Privacy Policies and Software Behavior

Privacy policy–software behavior misalignment are widely documented across software platforms. Research indicates that many software applications do not accurately disclose their actual data collection and data handling practices in their privacy policies (e.g., [6, 31, 42, 46]). Automated analyses reveal broad inconsistencies between declared and implemented data practices across web services (e.g.,[6]), mobile applications (e.g., [8, 42, 46]), and other digital systems (e.g., [3, 26, 45]).

A central source of these mismatches is the involvement of third-party services, such as advertising and analytics providers, which frequently collect user data without being explicitly mentioned in the corresponding privacy policies [25, 31]. In mobile ecosystems, third-party SDKs have been identified as a frequent cause of misalignment, as they often collect privacy-sensitive data beyond what is disclosed in the application's privacy policy [25, 42]. Similar patterns have been documented in the web domain, where many privacy policies fail to accurately disclose third-party data flows [6, 22].

Research further shows that privacy policies may be vague, incomplete, or internally contradictory, which obscures the software's actual data practices [21, 45]. In some cases, policies claim that no personal data is collected while simultaneously permitting third-party services or affiliated entities to collect and share user data [21, 46]. Additionally, some applications do not provide a privacy policy at all, which makes any data collection or sharing practices potentially inconsistent with stated disclosures [6, 46].

In conclusion, while stated privacy policies aim to inform users and ensure compliance, documented privacy disclosures often fail to reflect the software's actual behavior due to a complex interplay of factors centered on software developers.

2.2 Software Developers' Role in Privacy Policy Compliance

Software developers play a central role in privacy compliance because they translate abstract policy requirements into the code that governs data handling. Their interpretations determine whether software behavior aligns with stated privacy commitments. However, many developers lack formal training in core privacy principles such as Privacy by Design, data minimization, or Fair Information Practices [17, 34]. This limited expertise can hinder the correct implementation of privacy requirements and may lead to non-compliant solutions, even when developers receive instructions or expert support [1, 18]. These challenges are particularly evident in smaller development teams, where access to privacy expertise and legal resources is limited [28, 39].

One contributing factor is the "language barrier" between legal and technical domains [17]. Privacy policies and legal requirements are typically written in legal terminology that developers find difficult to translate into concrete technical specifications [5, 17]. Developers report a need for clear, technically oriented guidance, which is often unavailable [10]. As a result, privacy is frequently framed as a legal issue rather than a technical responsibility, which can create gaps in implementation practices [18].

Organizational context further shapes these challenges. When privacy is not embedded in organizational priorities or is secondary to functionality, speed, or innovation, it is often addressed only after core development decisions have been made [18]. Inconsistent managerial signals can reinforce this pattern and lead to discrepancies

between formal privacy expectations and actual implementation practices [4, 15].

Technical complexity also complicates privacy implementation. Modern software systems frequently depend on third-party components, dynamic APIs, and continuous updates, which make it difficult to maintain oversight of data flows [37]. As systems evolve or new services are integrated, developers may struggle to maintain accurate privacy policies and consistent data handling practices [38]. Even when privacy protection is intended, technical mistakes during updates, code reuse, data migration, or third-party integration can unintentionally expose or mishandle personal data [24].

Research also identifies factors that support more consistent privacy practices. Informal discussions within teams, support from management and stakeholders, clear documentation, and practices such as code reviews and privacy training can strengthen privacy awareness during development [5, 10, 39]. Together, these findings indicate that routine development decisions play a critical role in determining whether privacy protections are effectively implemented in practice.

2.3 Research Gap

Software developers are formally responsible for implementing privacy policies in code but often fail to achieve alignment between documented commitments and actual software behavior. Prior research attributes these gaps to organizational deprioritization of privacy, insufficient technical guidance, and conceptual confusion between privacy and security. Studies have examined communication barriers between developers and legal experts [17], organizational privacy strategies [4], or development teams as collective units [28]. Less attention has been given to the individual developers who directly implement privacy requirements in software systems.

This focus is important because developers translate legal and organizational privacy expectations into concrete technical implementations. Through routine coding tasks, integration decisions, and system modifications, they determine how data is collected, processed, and shared within the software. While managerial roles such as product owners or software architects may define priorities or system structures, developers operationalize these requirements at the level of code, where privacy-relevant behavior ultimately emerges.

Studying developers' interpretations therefore provides insight into how abstract privacy requirements are translated into technical actions and how organizational norms influence implementation decisions. This perspective helps explain how alignment between privacy policies and software behavior is produced in practice and why discrepancies between documented commitments and implemented data practices occur.

3 Method

3.1 Study Design and Ethical Considerations

To address our research questions, we conducted 15 semi-structured interviews with software developers. We adopted this qualitative approach in light of the limited body of existing research on how developers perceive and interpret their activities related to the alignment between privacy policies and actual software behavior in the handling of personal data (see section 1 and section 2). Semi-structured interviews enabled us to explore developers' perspectives in a nuanced, open-ended, and context-sensitive manner, while maintaining a clear focus on the research questions [32].

Ethical Considerations. This study was approved by the Institutional Review Board of our university's ethics committee (approval number: EK 12/2025). Prior to participation, we informed participants about the purpose of the study, their rights, and our data handling and storage practices. Participation was entirely voluntary, and participants were free to withdraw at any time without facing any negative consequences. In cases of withdrawal, no data was retained, and any data already collected was immediately discarded. Additionally, participants had the right to request the correction or deletion of their data even after completing the study. All collected data was stored in a secure environment in accordance with our institution's data security policies. Participants were required to read the study information sheet and privacy policy carefully, and informed consent was obtained before data collection started.

3.2 Recruitment of Participants and Sample Characteristics

This study explores software developers' perceptions and interpretations of their own activities as they relate to the completeness and accuracy of the privacy policies of the software they contributed to developing. To this end, purposive sampling was employed to recruit developers who had a basic awareness of the requirement for a privacy policy for the software they worked on. This selection criterion was intended to facilitate the examination of developers' understandings of potential causes of (mis)alignment between privacy policies and the actual privacy-relevant behavior of software systems. In addition, participants were required to have hands-on operational experience in software development and not be exclusively engaged in administrative or managerial roles. This criterion was adopted because developers with substantial operational involvement can directly influence both the software's implementation and its compliance with the stated privacy policy—a perspective that has received limited attention in prior research (see section 1 and section 2).

The first eleven interview participants (P01 to P11 in Table 1) were recruited via the crowdsourcing platform Prolific[1], which is widely used in usable security and privacy research [23, 30], including studies involving software developers [38, 40, 44]. For recruitment via Prolific, a study description was provided to inform potential participants about the core objectives and conditions of the study, including that we explicitly sought software developers to discuss activities related to the completeness and accuracy of software privacy policies. Since Prolific does not support direct filtering for developers with experience in privacy policy–related software development projects, proxy criteria were applied, including current full-time or part-time employment in engineering, IT, information networking, or information security, as well as selecting "coding" as an employment role on Prolific. Furthermore, we restricted recruitment to developers based in the EU or UK in order

[1] https://www.prolific.com/

Table 1: Overview of interview participants. *Participant-ID. **We consider companies as small with < 50, medium with < 250 and large with ≥ 250 employees [36].

P-ID*	Job Role	Work Exp. (Years)	Age	Industry	Software Focus	Country	Company Size**
P01	Sr software developer	15	30 – 34	Real Estate	Web app	Germany	Large
P02	DevOps engineer	5	30 – 34	Information Technology	Embedded system	UK	Small
P03	Software engineer	20	45 – 49	Healthcare	Healthcare system	Germany	Small
P04	Software engineer	10	35 – 39	Media & Entertainment	Web app	UK	Medium
P05	Full-stack developer	8	30 – 34	E-Commerce	E-commerce platform	Germany	Medium
P06	Back-end developer	7	30 – 34	Media & Entertainment	Web app	Portugal	Small
P07	Sr software architect	28	50 – 54	Media & Entertainment	Web app	UK	Large
P08	Jr software developer	< 1	25 – 29	Finance	Cloud-based system	Italy	Large
P09	Software architect	14	30 – 34	Information Technology	HR software	UK	Medium
P10	Software developer	9	25 – 29	Information Technology	Web and mobile app	Poland	Medium
P11	Software developer	7	34 – 39	Information Technology	Enterprise system	UK	Large
P12	Software developer	15	40 – 44	Education	Web app	Germany	Large
P13	Front-end developer	24	50 – 54	Education	Web app	Germany	Large
P14	Application developer	6	25 – 29	Information Technology	Web app	Germany	Large
P15	Software developer	8	35 – 39	Research & Development	Healthcare system	Germany	Large

to delimit and manage in particular the regulatory context of the study. Based on these criteria, 80 users were invited via Prolific to complete a short screening questionnaire administered using SoSciSurvey[2] (see Appendix A). The purpose of the screening survey was to identify interview candidates who met the selection criteria.

As part of the screening survey, participants first completed an informed consent form outlining the study's purpose, duration, withdrawal rights, and data protection procedures. They then provided demographic information (e.g., gender, country, and age) as well as professional background data (e.g., company size, sector, job role, and involvement in projects requiring a privacy policy). Eligibility for participation in the interview study required informed consent, proficiency in English or German, current involvement in a software project subject to a privacy policy, and complete disclosure of country and company size. Based on these criteria, 29 individuals were invited to participate in the interviews, of whom 11 took part. All 11 interviewees indicated in the questionnaire that "software developer" best described their current professional role among the available options.

Participants recruited via Prolific received financial compensation for both the screening survey and the interviews. Compensation was based on Prolific's standard rates and met at least the respective statutory minimum wage in the software developers' countries. Participation in the screening survey, which took approximately one to two minutes, was compensated with a fixed payment of £0.50. Interviews were compensated with a fixed amount of £12.00 regardless of their duration. All participants were informed about the compensation prior to taking part in the screening survey and the interviews.

After coding the first eleven interviews (see Section 3.4 for details on the coding procedure), we recruited four additional participants (P12 to P15) to further explore emerging themes and address remaining open questions. These participants were recruited through referrals within the authors' professional networks and did not receive financial compensation. The same selection criteria applied to the Prolific group were used for these interviewees and were verified through communication prior to and during the interviews. After conducting and coding a total of fifteen interviews, we determined that theoretical saturation had been reached because no new relevant aspects were identified and therefore concluded data collection [16].

Table 1 provides an overview of the final interview sample. The participants' mean age was 30 years, and the average professional experience was 12 years. Two participants (P08 and P12) identified as female, while all others identified as male. The 15 interviewees came from five countries and were employed in organizations ranging from small to large in size and across eight different industries, allowing us to capture software developers' experiences from a broad range of organizational contexts.

3.3 Interview Guide Design and Interview Procedure

Qualitative interviews are characterized by an inherent tension between prior structuring and openness to the phenomenon under investigation [20]. For the interviews conducted in this study, we employed a semi-structured interview guide consisting predominantly of open-ended questions. These questions served as narrative-generating prompts aligned with our research focus. The development of the interview guide began with a brainstorming session within the research team, during which typical activities and procedural steps related to the creation and maintenance of software privacy policies in organizational contexts were identified. Building on the shared understanding established during this session, we compiled an initial set of interview questions. These questions were subsequently reviewed, refined, and thematically

[2] https://www.soscisurvey.de/

organized based on their relevance to the research objectives and their potential to elicit new insights rather than merely reaffirm established knowledge [20]. The guiding principle was to formulate questions that maintained a clear focus on the study's core research aims while allowing for unexpected or emergent perspectives. Example questions included *"Please recall a recent situation where a privacy policy was created or changed: What triggered it?"* and *"How do you keep the privacy policy up to date as the software changes?"*. Appendix B provides an overview of the final interview guide.

The interviews were conducted with the aim of providing participants with a great degree of openness, enabling them to articulate their subjective perspectives and systems of relevance [20]. In this spirit, the prepared interview guide was used as a tool to introduce narrative-generating prompts related to the research topic, rather than as a rigid script to pre-structure the conversations in detail. The interviews followed the principle of allowing the interview participants to guide the flow of the conversation [20]. Accordingly, the sequence of questions was not predetermined but instead adapted to how each participant expressed their views. Interviewees were granted extended monologic speaking time and were encouraged to elaborate on topics beyond the scope of the interview guide.

Fourteen interviews were conducted remotely using videoconferencing software, and one interview was conducted in person. For the remote interviews, participants joined a virtual meeting room at the scheduled time and were informed that activating their camera was optional; some participants nevertheless chose to use video. Prior to all interviews, participants were asked to provide consent for audio recording, and all agreed. The interviews were then conducted and recorded. The average interview duration was approximately 55 minutes; the shortest interview lasted 24 minutes, and the longest 83 minutes.

3.4 Data Analysis

To prepare the data for analysis, all interviews were transcribed. Transcription was carried out using the software noScribe[3], which runs locally on users' machines, thereby maintaining the confidentiality of the interview recordings. The automatically generated transcripts were subsequently reviewed and corrected by members of the author team, using the original recordings to ensure accuracy and address potential quality issues.

An inductive coding approach was employed for the analysis [13]. In the initial phase, interviews were coded on a sentence- or paragraph-level basis. The analysis focused on identifying activities undertaken by software developers related to aligning the privacy policy with the software's actual behavior. In addition, we looked for organizational factors that complicate or undermine these alignment activities. For example, the statement of P03 *"I had to let the others know like what data about the customer I was storing. So, it is like for example, the application allows, especially when I built it on the cloud, I had to actually build the actual registration form for customers to actually register as well. So, I am actually aware of like what data we have collected. So, I did made the relevant people, like, aware, like the rest of my team aware what data I am collecting from the customers."* was assigned the first-order code *"SD informs development team when software task involves collecting or storing new data"*. During the coding process, individual text passages could be associated with multiple first-order codes, while each first-order code could also be derived from multiple text passages.

Similar activities were subsequently grouped into broader patterns of activities (second-order themes). For example, the first-order codes *"Software developer (SD) raises within team when new system is integrated that hands over customer data to a third party"*, *"SD informs development team when software task involved collecting or storing new data"*, and *"SD proactively informs legal team when new system is integrated"* were grouped into the second-order theme *"Informing relevant stakeholders when privacy-relevant substantive software changes have been established"*.

In the final step, the identified second-order themes were aggregated into broader dimensions, representing the highest level of abstraction in our coding hierarchy. For example, the aggregated dimension *"Software developers' privacy-related alignment activities primarily influencing the privacy policy"* encompasses second-order themes such as *"Discussing privacy-relevant behavior of software with relevant stakeholders before deployment and changes"*, *"Asking relevant stakeholders for approval before privacy-relevant substantive software changes have been established"*, and *"Documenting planned or implemented software's data handling practices for relevant stakeholders"*. Figure 1 provides an overview of the resulting data structure [13].

Throughout the entire analysis period, four members of the author team were involved in the coding process, with two researchers working in parallel at all times. They engaged in regular discussions to critically reflect on the emerging findings and to iteratively and consistently refine the coding [33]. The software MAXQDA[4] (version 2025) was used to support and document the coding process.

4 Results

The findings derived from our analysis are structured into three main sections. First, Section 4.1 presents *Software developers' privacy-related alignment activities primarily influencing the privacy policy*. Second, Section 4.2 describes *Software developers' privacy-related alignment activities primarily influencing software behavior*. Third, Section 4.3 presents *Software developers' perceptions of symbolic compliance*. Within each section, second-order themes are presented and elaborated in individual subsections (e.g., 4.3.2, 4.3.3). Exemplary first-order codes are highlighted in **bold**. To illustrate developers' perceived alignment activities as well as perceived organizational challenges and constraints contributing to symbolic compliance, we provide exemplary excerpts from the interview data.

4.1 Software developers' privacy-related alignment activities primarily influencing the privacy policy (RQ1)

Software developers' privacy-related alignment activities that primarily influence the privacy policy occur when changes to the software affect personal data processing and therefore require updates or changes to the privacy policy. Developers describe several activities in their daily work that contribute to this alignment which are presented below. These activities include, for example,

[3]https://github.com/kaixxx/noScribe

[4]https://www.maxqda.com/

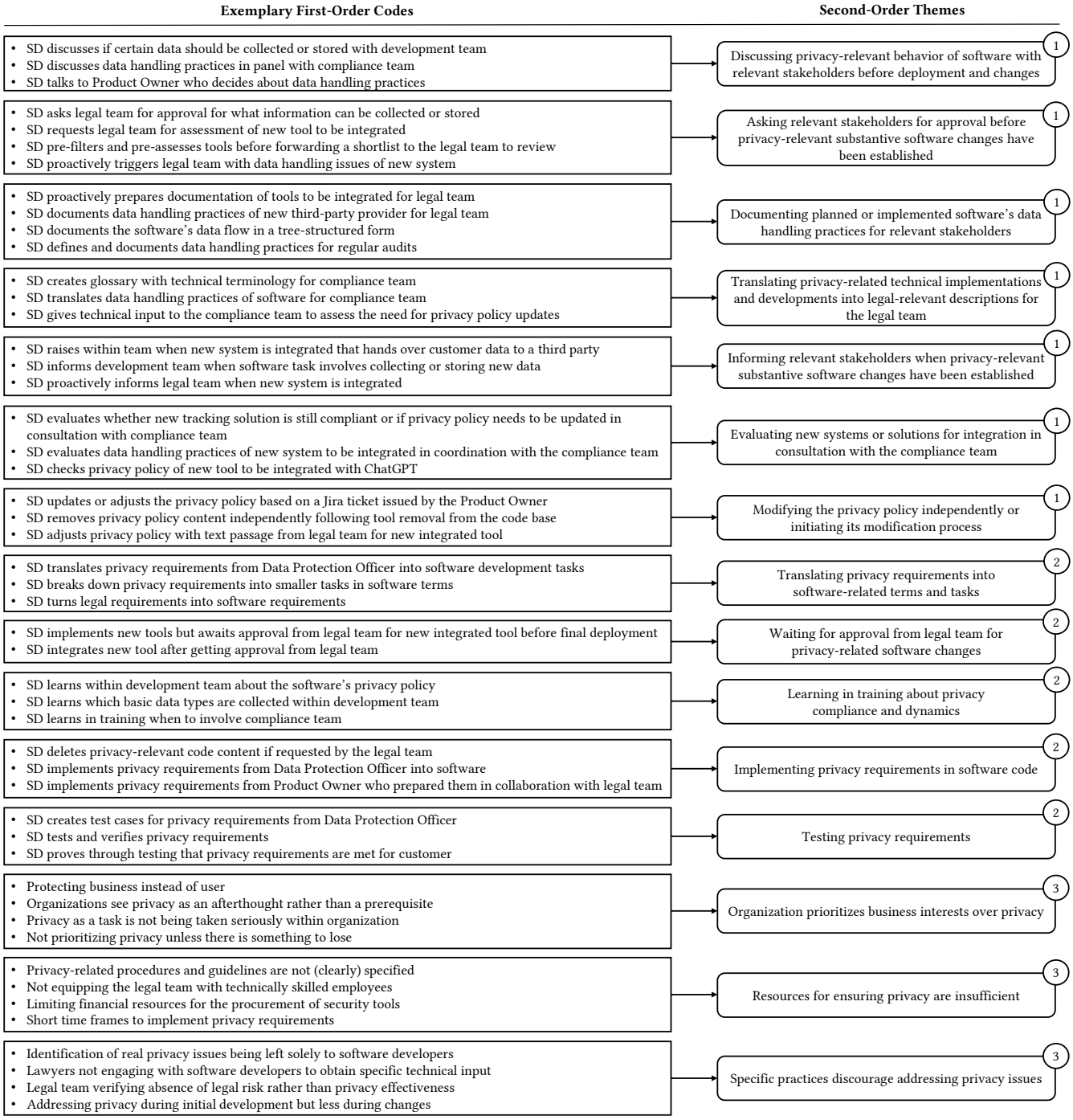

Figure 1: Data Structure.

discussing privacy-relevant behavior of software and requesting approval from relevant stakeholders.

4.1.1 Discussing privacy-relevant behavior of software with relevant stakeholders before deployment and changes. Software developers reported **discussing whether specific data should be collected or stored within the software development team** prior to deployment or the implementation of changes. For example, when collecting or storing additional customer data, it is communicated within the team rather than implemented independently. P02, who works in a tech startup, explains:

> *"If I am going to gather more customer data, I will put it through the others, put it through the team. I wouldn't sort of just do it myself without telling anyone."* (P02)

Developers further reported **discussing data handling practices in consultation with the compliance team or legal team**. In addition, developers **consult the Product Owner**, who is reported to decide on data handling practices of the software they are working on:

> *"We are talking to the product owner and then he or several people will decide what we should provide. It is important to make it understandable how the data is flowing or how the processes go to the product owner. For example, you have a feature that relies on a third-party and then you go back and forth with the third-party, because there are several steps involved. The product owner has to be aware of what we are doing and what we are sending, so he can communicate this then to the lawyers, for example. So that they can check if they have to adjust something or if everything's alright."* (P05)

4.1.2 Asking relevant stakeholders for approval before privacy-relevant substantive software changes have been established. Software developers reported **asking for approval** from relevant stakeholders such as the legal team, compliance team or within their development team prior to implementing substantive software changes that affect current data handling practices. This includes, for example, consulting the legal team and getting approval on what information may be collected or stored:

> *"Whenever we want to collect more information for our telemetry or audience analytics, mostly just those two, then we obviously have to go through [legal team] to get the approvals for what information we can collect."* (P07)

Moreover, developers described activities such as **requesting legal assessment of new tools** considered for integration. In another instance, developers already **pre-filtered and pre-assessed tools for theoretical feasibility before a shortlist of tools is submitted** to the legal team for them to review. Developers also reported **proactively involving the legal team** when data handling issues related to new systems are identified. P01 describes the activity as follows:

> *"In that specific case, we again collected the information about the tool that I mentioned earlier and documented it in the Excel table. We then passed this relevant data on to the [legal team] and asked them to use it for the specific passage in our privacy policy. In our case, this is point 3, Cookies and similar technologies. External tools are also listed there as a subcategory. We needed a new section for this particular tool and, of course, asked for an assessment of whether this was even possible."* (P01)

4.1.3 Documenting planned or implemented software's data handling practices for relevant stakeholders. Developers emphasized the importance of **rigorous documentation as a foundation for privacy-aware development** and saw documentation not only as a compliance requirement but also as a necessary condition for enabling transparency, maintaining consistency between system behavior and privacy statements:

> *"Especially when you process a lot of data, you simply have to document very well: 'What kind of data are we processing? Are new data types being collected? How are they further processed? Are there new processing sources?' That, that has to be an ongoing process and it always has to be kept up-to-date. And really, I think the most important thing is simply to make developers aware of keeping an eye on this documentation and to permanently see it as part of software development and not as 'Ok, this is something we still have to do', but rather that if this documentation does not exist, then the software is not finished and then we are not allowed to put it live."* (P03)

Software developers also described when they need to document planned or implemented software data handling practices for stakeholders such as the legal team. This included **proactively preparing documentation of tools** that are planned for integration and submitting these to the legal team for them to review:

> *"We proactively provided this during development. We weren't asked to do so. We simply created this list and gave it to them, asking them to review it and determine whether it was even possible for us to use these tools. Exactly. It should actually be the other way around, but in our case, it was proactive on our part, simply based on our experience."* (P01)

Developers further reported **documenting the data handling practices of new third-party providers** for the legal team when sensitive data is collected. In addition, developers, for example, **document the software's data flow in a tree-structured form** and **define and document data handling practices for regular audits**.

4.1.4 Translating privacy-related technical implementations and developments into legal-relevant descriptions for the legal team. From the interviews, we observed that software developers are tasked with translating privacy-related technical implementations and developments into legal-relevant descriptions for the legal or compliance team. This includes **creating shared artefacts such as glossaries** that explain technical terminology and data handling practices to support mutual understanding. For example, P03 describes the joint development of a glossary to clarify technical concepts, data flows, associated risks, and potential attack vectors,

as well as challenges arising from differences between compliance requirements and agile software development processes.

> *"You have to somehow find a common language there. ... We then also started, for example, to develop a glossary together, where certain technical terminology is simply explained again: What does that mean in detail? So what does something like encryption at rest or in transfer mean, and then really showing with examples how, data flows, where it is stored, what the risks are here, what possible attack vectors are, what can go wrong? That, there you also have to, I think, simply spend a lot of time together to somehow manage that well in the long run. And you also have to move very close to each other when it comes to certain processes. From a compliance perspective, of course, there is always an attempt to keep the processes watertight, which then, I would say, in agile software development often leads to processes being very slow and very sluggish and that you actually do not work as fast as you would like to work."* (P03)

Developers further report **translating the software's data handling practices** for the compliance team and **providing technical input** to assess whether updates to the privacy policy are required. Developers describe the need to explain technical data-processing practices in simplified terms so that legal or compliance teams can understand how personal data is handled in the system and determine whether the privacy policy requires revision. Developers also report communication difficulties and a lack of shared understanding between legal or compliance roles and software developers, which makes these translation activities necessary.

4.1.5 Informing relevant stakeholders when privacy-relevant substantive software changes have been established. Our data revealed that developers **inform relevant stakeholders** after privacy-relevant substantive software changes have been established. This includes **raising within the software development team** when new systems are integrated that transfer customer data to third parties, such as in the case of new payment systems or inform the software development team when software tasks involve the collection or storage of new customer data and making team members aware of what customer data is being collected. Moreover, developers **inform the legal team** when new systems are integrated and contact legal stakeholders when using new services:

> *"We are using a new service for earning money, like AdSense, or starting new cooperations with direct partners. Or we are using a new service, like a third-party provider. Then we also go to the lawyers, tell them what we are doing, what we are sending, if at all, how recognizable is it on the third-party or not. For example, do we use customer data or do we use more arbitrary data, like numbers instead of names on their side."* (P05)

Developers also noted that legal teams struggled with technical aspects of updating privacy documentation. They outlined that legal teams focused mainly on legal compliance and risk minimization, rather than ensuring the privacy policy accurately reflected the technical handling of user data. According to the developers, this focus limited the extent to which privacy documentation captured technical realities.

4.1.6 Evaluating new systems or solutions for integration in consultation with the compliance team. Our data reveals that the **evaluation of new systems or solutions** for integration happens often in consultation with the compliance team. Software developers also assess the data handling practices of new systems to be integrated and **evaluate whether new tracking solutions collect additional user data**. In addition, developers report evaluating whether the integration of new tracking solutions requires updates to the privacy policy, user consent, or involves a new provider in coordination with the compliance team:

> *"And then we introduced a new tracking solution, and we really had to take a close look: 'Is this a new provider? To what extent are we compliant?'"* (P11)

Moreover, developers describe how they **check the privacy policy of new tools** to be integrated for potential issues using Large Language Models such as ChatGPT:

> *"We sometimes use AI, ChatGPT for example, to summarise things for us. So, now that we have a privacy policy or something like that, I ask ChatGPT, 'Can you summarise this for me briefly in terms of GDPR aspects, whether you see any problems anywhere or anything like that?' Just to save ourselves some work. Yes. That might be a verification process."* (P01)

4.1.7 Modifying the privacy policy independently or initiating its modification process. Software developers report modifying the privacy policy independently or initiating its modification process including **updating or adjusting the privacy policy** based on Jira tickets issued by the Product Owner. Developers further report **removing or adjusting relevant sections of the privacy policy** within the software development team when tools are removed from or added to the code base. For example, P13 reports:

> *"That's usually easier because we simply take a proactive approach: when we no longer use a tool, we simply delete the entire passage from the privacy policy. Yes, we do that proactively and then just send it to Legal and say, for your information, we no longer use the tool, we have deleted the passages, and yes. So I don't think we even wait for approval."* (P13)

In addition, developers highlighted the importance of updating documentation and the privacy policy in tandem for end users. Adjusted privacy policy content is also reported to be based on text passages provided by the in-house legal department and subsequently submitted for legal review.

4.2 Software developers' privacy-related alignment activities primarily influencing software behavior (RQ1)

Software developers' privacy-related alignment activities that primarily influence software behavior capture situations in which changes to the privacy policy necessitate modifications to the software. Revisions to the policy require developers to adapt the system's behavior to ensure consistency with the updated descriptions

of data processing specified in the privacy policy. Typical alignment activities include, for example, translating privacy requirements into software-related terms and development tasks, and waiting for approval from the legal team before implementing privacy-related software changes.

4.2.1 Translating privacy requirements into software related terms and tasks. Privacy requirements are reported to be **translated by software developers into software-related terms and implementation tasks**. This includes, for example, converting privacy requirements received from the Data Protection Officer into development tasks and **breaking these down into smaller, actionable units in software terms**. For instance, P09, who works in a medium-sized company, explains:

> *"They will put requirements into writing, I might have to break those up into, into smaller tasks to make them make more sense into software terms, but that would be then the job of me to turn that into tangible workable things that we can implement in the system, and basically go back and forth to work out if it's possible or, you know, work out the time required for that, the actual effort required, no matter how big or small it looks on the surface."* (P09)

Further, P09 described that when legal stakeholders revised the privacy policies, it was then his responsibility, together with another architect, to develop an implementation plan to operationalize these requirements in the software system and ensure that the specified legal requirements are aligned with the resulting software changes:

> *"They essentially knocked heads together and changed our policies. And that was down to me and another architect to put into a plan of action in order to implement. So I was kind of the second level of that. Although I was required to review certain things and what was possible on a database level, on a backup level, that kind of stuff. So I kind of turned real world legal requirements into software requirements and ensured that they were implemented."* (P09)

4.2.2 Waiting for approval from legal team for privacy-related software changes. As described in Section 4.1.2, software developers request approval from relevant stakeholders for privacy-related software changes and **wait for this approval before implementation**. This includes seeking approval for new tools to be integrated prior to going live and **proceeding with integration only after approval has been granted**. But it can also occur that the legal team advices against a tool, as P01 explains:

> *"Sometimes we also receive feedback from Legal regarding tools where we have not identified any issues, stating, 'Yes, the privacy policy reserves so many rights to do something differently, to process the data elsewhere, to pass on data to third parties over which we have no control.' So then we get a clear message saying, 'We strongly advise against using this tool.' And then we look for alternatives."* (P01)

The interviewed software developers stated that privacy-related feedback from legal teams can occur late in the development process, rather than during implementation. Collaboration was described as infrequent and not regularly integrated into development workflows.

4.2.3 Learning in training about privacy compliance and dynamics. In companies operating in sensitive contexts, such as healthcare, developers reported that privacy compliance receives particular attention. Software developers described that organizations provide trainings and internal guidance to ensure that developers understand how personal data must be handled within the software system and that developers **learn about privacy compliance and related dynamics through training and internal knowledge exchange**. Through these trainings and internal knowledge exchanges, developers learn about the software's privacy policy, the basic types of data collected, and the associated compliance requirements. The trainings also clarify organizational boundaries by indicating in which areas developers can make implementation decisions independently and in which situations **the compliance team must be involved**:

> *"So there were very many trainings with us, so that really every developer understands what, what does such a privacy policy actually mean? What do we promise the users there? What, what are the legal requirements? And of course really the technical responsible persons must then be trained and of course also the product responsible persons, so that from the beginning it is actually very clear, ok, these are the areas in which we can move freely, and here we have to again keep consultation with the compliance team."* (P03)

4.2.4 Implementing privacy requirements in software code. We observed that privacy requirements are implemented by software developers in the software code based on inputs from relevant stakeholders. For example, activities include **deleting privacy-relevant code content upon request from the legal team**:

> *"We only had it more often, that we had to remove content as quickly as possible. So, there we then also get information from Legal. 'Here please remove everything about this project from our website or something like that.' Because they then suspect, that it can come to it soon. So, but concretely I have not experienced that yet. But I also do not know whether, whether they would tell us that. To be honest. So maybe behind these emails please remove everything about it. Maybe there is already a warning letter behind it."* (P01)

Developers further **implement privacy requirements** provided by the Data Protection Officer as well as requirements communicated by the Product Owner that were prepared in collaboration with the legal team:

> *"We have a task where all the main things are provided, so it was prepared beforehand by the product owner in collaboration with the lawyers. And then we as developers implement the changes."* (P05)

4.2.5 Testing privacy requirements. Our interviewed software developers reported testing if privacy requirements are met through the **creation of test cases based on privacy requirements**. P08

describes implementing and verifying access restrictions in software to ensure that specific functionalities are only available to authorized user groups. Moreover, P09 describes testing whether implemented software changes meet specified privacy or data protection requirements. This involves examining how data flows across different system components, to verify that data is processed in accordance with the defined requirements. The **testing** process also includes demonstrating to the client that the implemented changes meet the specified **privacy requirements** before deployment:

> *"Essentially pull that right down to, to the lowest of possible levels where that data is no longer accessible in a way that could identify certain people. And it was a slightly, slightly higher echelon of requirements that our client would typically require. So that was a long, laborious task to, to look at where the data went. Obviously that touches every aspect of the system, be it back end, front end reporting, every sort of microservice, every sort of micro database, every sort of data transposition process. Everything was basically had to be investigated and, pulled apart in order to work out if we could really meet ... [the] requirement. ... And find ways to prove to the client that in testing, I'd actually met those requirements."* (P09)

.

4.3 Software Developers' Perceptions of Symbolic Compliance (RQ2)

The analysis of the interviews revealed various organizational factors that complicate or undermine the alignment activities described in Sections 4.1 and 4.2. The findings suggest that, from the perspective of the interviewed software developers, organizations often place strong emphasis on having a privacy policy that appears externally consistent and compliant, while it may be decoupled to varying degrees from the software's actual behavior. This constellation can be interpreted as a form of *symbolic compliance*. The subcategories of symbolic compliance identified in the analysis are presented below.

4.3.1 Organization prioritizes business interests over privacy. In several interviews, a general perception emerged among software developers that their organizations prioritize business interests over privacy. This includes the impression that organizations tend to see **privacy as an afterthought rather than a prerequisite** and **do not treat it as a serious concern**. Accordingly, **privacy is often prioritized only once serious consequences become likely** or potential harms become more salient. In this context, **reputational risks** resulting from privacy violations were mentioned in particular, as their prevention was described as a key priority because reputational damage could significantly harm business interests. Furthermore, the impression was raised that **teams responsible for privacy lack interest in understanding actual data handling practices**. Instead, the primary focus was perceived to be on ensuring that the privacy policy appears flawless externally, rather than on accurately reflecting data practices. P07 explained that the **business is not interested in having an accurate privacy policy**, as the primary goal is **protecting the business rather than the user**:

> *"I think the main thing would be to get the lawyers and the privacy team to care most about privacy and not just about protecting the business. I feel like their main motivation is protecting the business, not protecting the user. It's all about making sure the business can't get criticized, not about making sure that the user is protected from any privacy risks."* (P07)

P09 expressed a similar perception. Drawing on his interactions with software-as-a-service (SaaS) clients of an HR software, he reported that **clients tend to push back when privacy requirements may reduce software functionality** and that **they do not always perceive value in privacy compliance**:

> *"And it's extremely boring and it's stuff that they [clients] just don't care about. It doesn't make them money. If anything, it saves them from lawsuits, yes, but it doesn't make them money. So it's really hard to put a price tag on them for how important that understanding is and how important it is to be extremely compliant at all times. ... The hardest part is trying to make the client understand the value of this stuff over just profit."* (P09)

4.3.2 Resources for ensuring privacy are insufficient. The analysis of the interviews revealed not only a general perception among some software developers that organizations prioritize business interests over privacy, but also statements indicating that resources for ensuring privacy are insufficient. Among the aspects raised were **limited financial resources for procuring security tools** and **short time frames for implementing privacy requirements**. In addition, several references pointed to **missing expertise**, including low levels of privacy expertise among developers, limited privacy knowledge among B2B clients of an HR SaaS solution, and a lack of technically skilled employees within legal teams. P02 described the situation regarding limited privacy expertise in the startup he works for as follows:

> *"But, I guess, if we do get like an expert in, a specialist, they might just come in and rip everything up telling us that we done it all wrong. I won't be surprised if that's what happens to a lot of startups."* (P02)

In addition, several software developers noted that **privacy-related procedures and guidelines are not clearly specified** within their organizations. These statements referred both to the lack of concrete guidance for the operational implementation work of developers and to missing specifications during the requirements engineering process. In this context, P01 described feeling left alone with the responsibility for addressing privacy-related issues:

> *"Things are a bit unstructured here. You know, the classic flat hierarchies and working with a lot of personal responsibility. So, you often feel a bit left alone and think to yourself: 'well, if I hadn't pointed this out now, we probably would have just gone online with it, even though it's totally vulnerable'. So, I basically have to actively pass this on to Legal. And if I hadn't done that and had just said, 'yeah, the change just slipped through for us and goes online like this,' then—bam—€10,000*

> *warning notice. ... In the end, you'd probably still somehow be held responsible for it. ... The processes are not clearly structured. So, management says, we would like this and that feature, then it kind of passes through middle management like that and eventually ends up with us. Like, 'yeah, just do it.' And then we have to take care that the whole thing basically goes back up the ladder again and that the corresponding approvals are obtained, instead of clarifying something like that beforehand. ... It also just costs time. If at that earlier stage on this ladder from top to bottom it had already been determined, because Legal had already been brought in, that the tool is generally not an option for us, then we wouldn't have had to deal with it in the first place."* (P01)

4.3.3 Specific practices discourage addressing privacy issues. The findings indicate not only a general perception among several software developers that privacy is subordinated to business interests (Section 4.3.1) and that resources for effectively addressing privacy tasks are often insufficient (Section 4.3.2), but also numerous specific practices that discourage addressing privacy issues. One such practice is that the **identification of real privacy issues is often left solely to software developers**, as illustrated by P01's quote at the end of Section 4.3.2. Related statements referred to situations in which **legal teams are not involved early in the development process** or where **no direct contact between developers and lawyers exists**. Furthermore, it was noted that **lawyers do not actively engage with software developers to obtain specific technical input** and that **legal teams tend to focus on verifying the absence of legal risks rather than ensuring privacy effectiveness**, consistent with **privacy policies formulated in deliberately broad and non-specific terms**. In addition, **teams responsible for privacy were described as responding slowly**, which can hinder the development process. P15 noted that software development can be significantly slowed down when privacy issues are raised, **as they may be routed through multiple stakeholders**:

> *"Otherwise it's always difficult, to follow up on it or to find the right people, and then they say, well, there are other people you need to talk to. So, if something comes up regarding data or the use of data, then it's immediately like this, this huge chain, and it just takes time."* (P15)

However, the absence of a direct communication channel between software developers and stakeholders with overarching responsibility for privacy within an organization (e.g., legal or compliance teams) does not necessarily mean that developers are left alone in dealing with privacy-related issues. The findings also include situations in which developers were largely excluded from actively engaging with privacy matters by product owners or responsible managers **who effectively abstracted privacy responsibilities away from software developers**. In these situations, communication with privacy stakeholders was handled by the product owner or other managers, while developers were largely limited to implementing predefined requirements.

Finally, one developer noted that **privacy tends to be addressed primarily during the initial development phase** but receives less attention during later changes. In addition, the following passage of P03 suggests that software developers may actively contribute to **limiting privacy-related requirements for documentation and testing** in order to work more flexibly:

> *"And you also have to move closer to each other when it comes to certain processes. From a compliance perspective, there is of course always an attempt to keep processes watertight, which then, I'd say, in agile software development often leads to processes becoming very slow and rather cumbersome, and you're basically not working as quickly as you would like to. So, in that respect we've gradually moved closer to each other and said, okay, in certain areas we can, for example, relax the required documentation a bit, and we don't have to completely test the entire app just to release a small fix when there was some minor vulnerability somewhere. Instead, we can also find ways to get there faster so that we ultimately deliver value to the users again."* (P03)

5 Discussion

Significant discrepancies between the statements made in a software's privacy policy and its actual data practices are a well-documented phenomenon in prior research (e.g., [2, 8, 42, 45, 46]), yet they have been insufficiently examined from the perspective of software developers. The present study addresses this research gap based on 15 semi-structured interviews with software developers. Figure 2 integrates our findings into an overarching framework. The outer part of Figure 2 relates to RQ1 and highlights the necessity of alignment activities undertaken by software developers to keep the privacy policy and a software system's data practices consistent. Specifically, these alignment activities may primarily involve either adapting the privacy policy to the software's behavior (illustrated by the arrow from "software behavior" to "privacy policy") or, conversely, aligning the software with the privacy policy (illustrated by the arrow from "privacy policy" to "software behavior"). The inner part of the figure relates to RQ2 and identifies symbolic compliance as an important factor that can shape alignment activities and hinder or undermine their implementation and effectiveness. The theoretical contributions, practical implications, and limitations of our work are discussed in the following subsections.

5.1 Theoretical Contributions

Our developers reported that in the absence of formal privacy policy-related processes, responsibilities are often handled informally within development teams. Developers relied on team discussions, personal judgment, and individual initiative to manage tasks such as identifying when a privacy policy might need updating or flagging inconsistencies. This was especially prevalent in small and medium-sized organizations, where flat hierarchies and limited resources make the establishment of structured privacy workflows challenging. These findings align with the study of Tahaei et al. [39] on Privacy Champions, which highlighted that, in the face of organizational and technical barriers, privacy-conscious developers often turn to informal strategies such as internal communication,

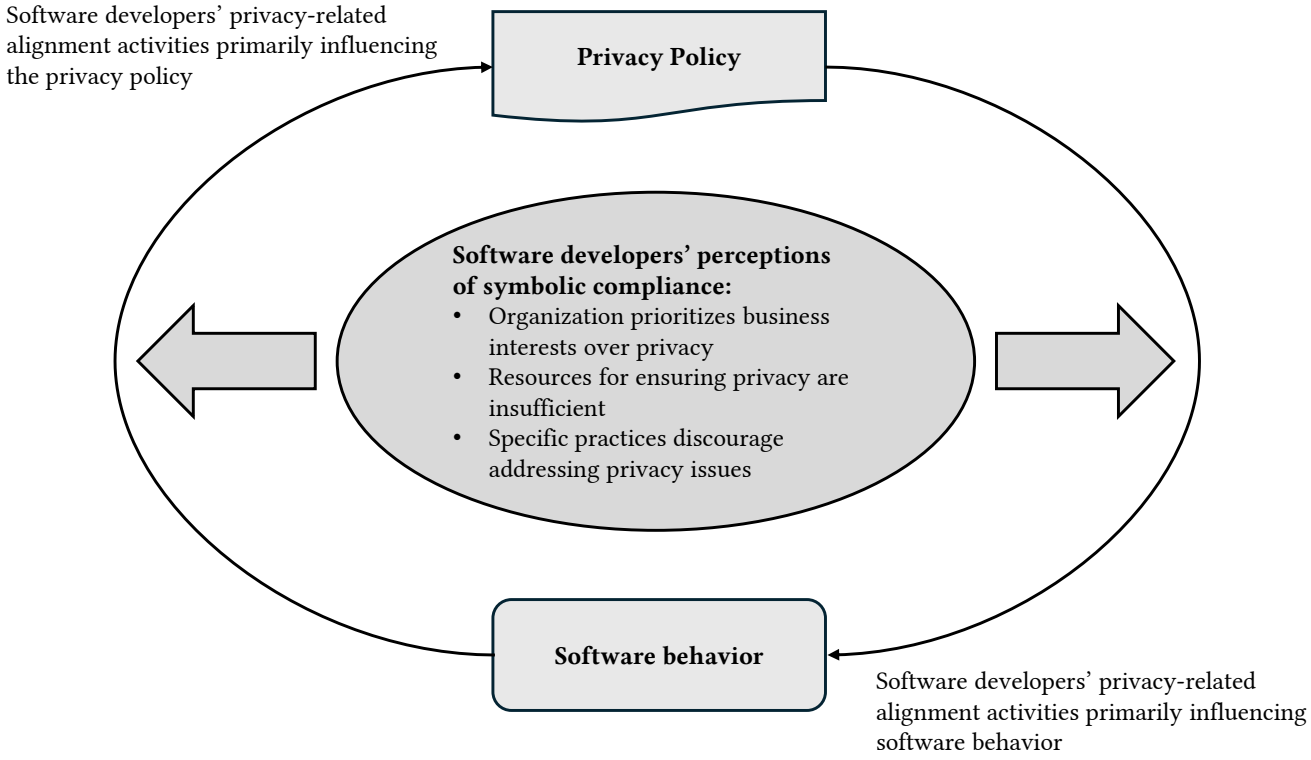

Figure 2: Developers' activities and perceived symbolic compliance in the privacy policy–software alignment cycle.

management support, and peer-driven documentation to foster privacy-respecting practices. While our participants were not formally designated as "Privacy Champions," their reliance on similar informal mechanisms underscores the extent to which developers must navigate privacy concerns without institutionalized support, especially in environments where privacy is not embedded as a first-class priority. Our study extends this understanding by showing how developers compensate for these shortcomings through informal routines and ad hoc practices. While such practices can help detect issues, they also create uneven coverage and leave privacy policy maintenance vulnerable to oversight.

Across interviews, developers reported that once a privacy policy is created—typically with legal input—there is no structured mechanism to ensure it remains accurate over time. Changes in software behavior are not routinely reflected in policy updates unless a developer takes personal initiative or an audit occurs. Even developers who were aware of potential inconsistencies lacked the time, resources, or authority to act on them. The absence of verification practices during day-to-day development is a core reason why these mismatches persist. Our findings also resonate with Bednar et al. [5], who argued that the legal framing of privacy makes it difficult to translate into technical controls. Here, we show that this difficulty is not just conceptual—it is operationalized through missing verification routines in software development.

Our interviewed software developers also described communication with legal or compliance stakeholders as delayed, inconsistent, or entirely absent. Developers often had to translate legal requirements on their own, while legal teams lacked visibility into technical decisions. In some cases, developers were expected to contribute privacy-relevant information but were not provided with structured ways to do so; in others, privacy policies were created without developer input altogether. This finding supports earlier work by Horstmanm et al. [17], who pointed to gaps in privacy knowledge and weak collaboration across roles. Our study refines this by showing how communication breakdowns between developers and legal experts directly result in inaccurate or outdated privacy documentation. It also complements findings by Sirur et al. [35] and von Davier et al. [43], who emphasized the lack of privacy integration within organizational workflows.

Taken together, developers can contribute to both the alignment and misalignment between privacy policies and actual data practices. Their influence is shaped not only by technical implementation choices and decisions but also by their role in communication and documentation. While developers are well positioned to ensure alignment, their efforts are undermined by informal processes, missing verification infrastructure, and weak interdepartmental collaboration.

5.2 Practical Implications

Our findings point to a number of actionable recommendations for both software developers and organizations seeking to reduce misalignment between privacy policies and actual data handling practices.

For *software developers*, the integration of privacy-related tasks into routine development activities is essential. Developers should embed privacy checks into agile workflows, for example, in sprint planning and code reviews. Using tagging systems or checklists can help flag privacy-relevant tasks early in the development process. In addition, maintaining up-to-date documentation that maps data handling logic to legal obligations is critical. Developers may benefit from tools that support inline annotations or automated tracking of data flows. To reduce gaps in legal-technical coordination, developers should also initiate communication with legal or compliance teams when working on features involving personal data. Structured communication templates can help standardize information exchange and ensure critical details are not lost. Finally, developers can actively contribute to verification practices by writing or maintaining test cases that validate privacy-related constraints, such as access controls or logging behavior, and ensure the system remains aligned with what is declared in privacy policies.

For *organizations*, our findings suggest the need to move beyond informal responsibility sharing. Assigning clear privacy roles within development teams—such as Privacy Champions or Liaisons—can support continuous compliance and foster accountability. Organizations should also implement structured workflows and review mechanisms, such as formal privacy reviews integrated into development pipelines or sign-off procedures from legal stakeholders before deploying features that affect data processing. Improving collaboration between legal and technical roles is also crucial. This can be achieved through co-developed documentation frameworks and joint training sessions that help translate legal language into actionable technical terms. Encouraging legal teams to consult developers during the drafting or revision of privacy policies ensures that policies better reflect system realities. In addition, fostering a privacy-positive culture within the organization—by promoting awareness, recognizing developer initiative, and framing privacy as a core element of product quality—can help institutionalize privacy as an ongoing shared responsibility, not merely a compliance task.

5.3 Limitations & Future Work

Our study is based on a purposive sample of 15 software developers, selected to cover a range of organizational contexts, domains, and experience levels. This sampling approach supports depth and diversity of insights but does not aim to produce a statistically representative sample of the broader software developer population. Therefore, the findings should be interpreted within the studied regulatory context.

The qualitative design of our study prioritizes understanding over measurement, which is suitable for exploring software developers' individual perceptions and practices. However, this also means that the conclusions reflect the experiences of a limited number of developers and may not capture all relevant perspectives. Future studies could incorporate larger samples from different regulatory contexts and cultures, including quantitative approaches, to further assess generalizability and validate the identified perceived alignment activities of software developers and their perception of symbolic compliance.

6 Conclusion

This study investigated what activities software developers perceived in their own work as contributing to the alignment between privacy policies and the software's actual behavior and how they described organizational influences that lead to symbolic rather than substantive alignment be- tween privacy policies and the software's actual behavior. Through 15 semi-structured interviews with software developers, who are directly involved in implementing privacy-relevant functionality, we aimed to uncover their perceived alignment activities and their perception of symbolic compliance. Our findings reveal several alignment activities and identify organizational decoupling as an important factor leading to symbolic compliance. By centering the experiences of individual software developers, this study offers a deeper understanding of the sociotechnical challenge that underlie privacy policy compliance.

Acknowledgments

This research work was supported by the National Research Center for Applied Cybersecurity ATHENE. ATHENE is funded jointly by the German Federal Ministry of Research, Technology and Space and the Hessian Ministry of Science and Research, Arts and Culture.

References

[1] Abdulrahman Alhazmi and Nalin Asanka Gamagedara Arachchilage. 2021. I'm all ears! Listening to software developers on putting GDPR principles into software development practice. *Personal and Ubiquitous Computing* 25, 5 (2021), 879–892.

[2] Benjamin Andow, Samin Yaseer Mahmud, Justin Whitaker, William Enck, Bradley Reaves, Kapil Singh, and Serge Egelman. 2020. Actions speak louder than words: {Entity-Sensitive} privacy policy and data flow analysis with {PoliCheck}. In *29th USENIX Security Symposium (USENIX Security 20)*. 985–1002.

[3] Pauline Anthonysamy, Awais Rashid, and Phil Greenwood. 2011. Do the privacy policies reflect the privacy controls on social networks?. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*. IEEE, 1155–1158.

[4] Renana Arizon-Peretz, Irit Hadar, Gil Luria, and Sofia Sherman. 2021. Understanding developers' privacy and security mindsets via climate theory. *Empirical Software Engineering* 26, 6 (2021), 123.

[5] Kathrin Bednar, Sarah Spiekermann, and Marc Langheinrich. 2019. Engineering Privacy by Design: Are engineers ready to live up to the challenge? *The Information Society* 35, 3 (2019), 122–142.

[6] Duc Bui, Brian Tang, and Kang G Shin. 2023. Detection of inconsistencies in privacy practices of browser extensions. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2780–2798.

[7] California Consumer Privacy Act of 2018 (CCPA) 2024. *California Consumer Privacy Act of 2018 (CCPA)*. Retrieved November 30, 2025 from https://oag.ca.gov/privacy/ccpa

[8] Jacob Erickson, Jewel Y. Yuzon, and Tamara Bonaci. 2022. What You Do Not Expect When You Are Expecting: Privacy Analysis of Femtech. *IEEE Transactions on Technology and Society* 3, 2 (2022), 121–131. https://doi.org/10.1109/TTS.2022.3160928

[9] Sascha Fahl, Marian Harbach, Henning Perl, Markus Koetter, and Matthew Smith. 2013. Rethinking SSL development in an appified world. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. 49–60.

[10] Maximilian Josef Frank, Stephen Meisenbacher, Alexandra Klymenko, and Florian Matthes. 2025. Investigating Developer Preferences for the Documentation of Privacy Requirements: Perspectives from Germany. (2025).

[11] GDPR Enforcement Tracker 2025. *Fines statistics*. Retrieved May 28, 2025 from https://www.enforcementtracker.com/?insights

[12] Martin Georgiev, Subodh Iyengar, Suman Jana, Rishita Anubhai, Dan Boneh, and Vitaly Shmatikov. 2012. The most dangerous code in the world: validating SSL certificates in non-browser software. In *Proceedings of the 2012 ACM conference on Computer and communications security*. 38–49.

[13] Dennis A Gioia, Kevin G Corley, and Aimee L Hamilton. 2013. Seeking qualitative rigor in inductive research: Notes on the Gioia methodology. *Organizational research methods* 16, 1 (2013), 15–31.
[14] Jenny Guber and Iris Reinhartz-Berger. 2024. Privacy-compliant software reuse in early development phases: A systematic literature review. *Information and Software Technology* 167 (2024), 107351.
[15] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. 2018. Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering* 23, 1 (2018), 259–289.
[16] Monique Hennink and Bonnie N Kaiser. 2022. Sample sizes for saturation in qualitative research: A systematic review of empirical tests. *Social science & medicine* 292 (2022), 114523.
[17] Stefan Albert Horstmann, Samuel Domiks, Marco Gutfleisch, Mindy Tran, Yasemin Acar, Veelasha Moonsamy, and Alena Naiakshina. 2024. "Those things are written by lawyers, and programmers are reading that." Mapping the Communication Gap Between Software Developers and Privacy Experts. *Proceedings on Privacy Enhancing Technologies* (2024).
[18] Stefan Albert Horstmann, Sandy Hong, David Klein, Raphael Serafini, Martin Degeling, Martin Johns, Veelasha Moonsamy, and Alena Naiakshina. 2025. "Sorry for Bugging you so much." Exploring Developers' Behavior Towards Privacy-Compliant Implementation. In *2025 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1215–1233.
[19] Yousra Javed and Ayesha Sajid. 2024. A Systematic Review of Privacy Policy Literature. *Comput. Surveys* 2 (2024), 1–43.
[20] Jan Kruse and Christian Schmieder. 2015. Qualitative interviewforschung [Qualitative interview research].
[21] Tianshi Li, Elizabeth Louie, Laura Dabbish, and Jason I Hong. 2021. How developers talk about personal data and what it means for user privacy: A case study of a developer forum on reddit. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW3 (2021), 1–28.
[22] Timothy Libert. 2018. An automated approach to auditing disclosure of third-party data collection in website privacy policies. In *Proceedings of the 2018 World Wide Web Conference*. 207–216.
[23] Yanzi Lin, Jaideep Juneja, Eleanor Birrell, and Lorrie Faith Cranor. 2023. Data safety vs. app privacy: Comparing the usability of android and ios privacy labels. *arXiv preprint arXiv:2312.03918* (2023).
[24] Tina Marjanov, Maria Konstantinou, Magdalena Jóźwiak, and Dayana Spagnuelo. 2023. Data security on the ground: investigating technical and legal requirements under the GDPR. *Proceedings on Privacy Enhancing Technologies* (2023).
[25] Mark Huasong Meng, Chuan Yan, Yun Hao, Qing Zhang, Zeyu Wang, Kailong Wang, Sin Gee Teo, Guangdong Bai, and Jin Song Dong. 2024. A large-scale privacy assessment of Android third-party sdks. *arXiv preprint arXiv:2409.10411* (2024).
[26] Amarachi Chinwendu Nwaeze, Pavol Zavarsky, and Ron Ruhl. 2017. Compliance evaluation of information privacy protection in e-government systems in Anglophone West Africa using ISO/IEC 29100: 2011. In *2017 Twelfth International Conference on Digital Information Management (ICDIM)*. IEEE, 98–102.
[27] Mariana Peixoto, Dayse Ferreira, Mateus Cavalcanti, Carla Silva, Jéssyka Vilela, João Araújo, and Tony Gorschek. 2020. On understanding how developers perceive and interpret privacy requirements research preview. In *Requirements Engineering: Foundation for Software Quality: 26th International Working Conference, REFSQ 2020, Pisa, Italy, March 24–27, 2020, Proceedings 26*. Springer, 116–123.
[28] Maxwell Prybylo, Sara Haghighi, Sai Teja Peddinti, and Sepideh Ghanavati. 2024. Evaluating privacy perceptions, experience, and behavior of software development teams. In *Twentieth Symposium on Usable Privacy and Security (SOUPS 2024)*. 101–120.
[29] Regulation (EU) 2016/679 of the European Parliament 2018. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 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). http://data.europa.eu/eli/reg/2016/679/2016-05-04. Accessed: 2025-05-28.
[30] Nathan Reitinger, Bruce Wen, Michelle Mazurek, and Blase Ur. 2024. What does it mean to be creepy? Responses to visualizations of personal browsing activity, online tracking, and targeted ads. *Proceedings on Privacy Enhancing Technologies* (2024).
[31] David Rodriguez, Jose M Del Alamo, Celia Fernández-Aller, and Norman Sadeh. 2024. Sharing is not always caring: Delving into personal data transfer compliance in android apps. *IEEE Access* 12 (2024), 5256–5269.
[32] Ruslin Ruslin, Saepudin Mashuri, Muhammad Sarib Abdul Rasak, Firdiansyah Alhabsyi, and Hijrah Syam. 2022. Semi-structured Interview: A methodological reflection on the development of a qualitative research instrument in educational studies. *IOSR Journal of Research & Method in Education (IOSR-JRME)* 12, 1 (2022), 22–29.
[33] Johnny Saldaña. 2021. The coding manual for qualitative researchers. (2021).
[34] Awanthika Senarath and Nalin AG Arachchilage. 2018. Why developers cannot embed privacy into software systems? An empirical investigation. In *Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering 2018*. 211–216.
[35] Sean Sirur, Jason RC Nurse, and Helena Webb. 2018. Are we there yet? Understanding the challenges faced in complying with the General Data Protection Regulation (GDPR). In *Proceedings of the 2nd international workshop on multimedia privacy and security*. 88–95.
[36] Small and medium-sized enterprises (SME) 2018. Small and medium-sized enterprises (SME). https://www.destatis.de/EN/Themes/Economic-Sectors-Enterprises/Enterprises/Small-Sized-Enterprises-Medium-Sized-Enterprises/ExplanatorySME.html. Accessed: 2025-05-30.
[37] Sarah Spiekermann and Lorrie Faith Cranor. 2009. Engineering Privacy. *IEEE Transactions on Software Engineering* 35, 1 (2009), 67–82. https://doi.org/10.1109/TSE.2008.88
[38] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Deciding on personalized ads: Nudging developers about user privacy. In *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*. 573–596.
[39] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Privacy champions in software teams: Understanding their motivations, strategies, and challenges. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.
[40] Mohammad Tahaei, Kami Vaniea, Konstantin Beznosov, and Maria K Wolters. 2021. Security notifications in static analysis tools: Developers' attitudes, comprehension, and ability to act on them. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–17.
[41] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. 2020. Understanding privacy-related questions on stack overflow. In *Proceedings of the 2020 CHI conference on human factors in computing systems*. 1–14.
[42] Zeya Tan and Wei Song. 2023. PTPDroid: Detecting violated user privacy disclosures to third-parties of Android apps. In *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 473–485.
[43] Thomas Şerban von Davier, Konrad Kollnig, Reuben Binns, Max Van Kleek, and Nigel Shadbolt. 2023. We are not there yet: The implications of insufficient knowledge management for organisational compliance. *arXiv preprint arXiv:2305.04061* (2023).
[44] Daniel Votipka, Desiree Abrokwa, and Michelle L Mazurek. 2020. Building and validating a scale for secure software development self-efficacy. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–20.
[45] Xiao Yu, Yiyu Yang, Wenjie Wang, and Yuqing Zhang. 2021. Whether the sensitive information statement of the IoT privacy policy is consistent with the actual behavior. In *2021 51st annual IEEE/IFIP international conference on dependable systems and networks workshops (DSN-W)*. IEEE, 85–92.
[46] Sebastian Zimmeck, Peter Story, Daniel Smullen, Abhilasha Ravichander, Ziqi Wang, Joel Reidenberg, N Cameron Russell, and Norman Sadeh. 2019. Maps: Scaling privacy compliance analysis to a million apps. *Proceedings on Privacy Enhancing Technologies* (2019).

A Screening Survey

We are looking for software developer, who are developing software on a daily basis. It does not matter, whether their software is sold or provided to the end user by their company or by a third party. With this questionnaire, we are screening for participants who would like to take part in a remote semi-structured interview. The interview's topic will be the creation, maintenance, and verification processes of the privacy policies of the software product you are working on or have worked on in your company.

Q1 What gender do you identify most with?

- Female
- Male
- Non-binary
- Other
- Prefer not to say

Q2 Which country are you currently living in? [Dropdown-list]

Q3 How old are you?

- < 20 years
- 20 – 24 years
- 25 – 29 years
- 30 – 34 years
- 35 – 39 years

- 40 – 44 years
- 45 – 49 years
- 50 – 54 years
- 55 – 59 years
- >= 60 years
- Prefer not to say

Q4 Which professional orientations fit your company best? - You may select more than one answer. [Dropdown-list]
Q5 How many employees are working in your company?

- < 5 employees
- 5 – 9 employees
- 10 – 49 employees
- 50 – 99 employees
- 100 - 249 employees
- >= 250 employees
- I am not sure
- Prefer not to say

Q6 Does your company provide Business-to-Business (B2B) or Business-to-Consumer (B2C) solutions?

- Business-to-Business (B2B) solutions
- Business-to-Consumer (B2C) solutions
- I am not sure

Q7 Are you currently working on a software project, which needs a privacy policy?

- Yes
- No

Q8 Which job description fits your current job position best? [Dropdown-list]

B Interview Guide

Introduction
My name is XY and I am from the Research Group of XY. We want to understand how privacy policies are managed in software development. We want to gain insights into your experiences and practices in creating, verifying, and maintaining privacy policies. What roles are involved, what information is needed, and what challenges arise when ensuring that privacy policies align with the source code and actual behavior of the software. I will ask you some open questions. Take as much time as you want for each. I will not interrupt you. There are no right or wrong answers. I'm interested in your daily work experiences and practices. With your permission, I will start the audio recording.
Warm up

(1) Please describe your current role and typical responsibilities in software development?
(2) What kind of software do you currently work on?
(3) How have you been involved with privacy-related tasks or questions?

Creation of Privacy Policies

(1) In your current or past projects, how were you involved in writing/ reviewing/ updating privacy policies?
(2) When do privacy policies usually come into play in your development process?
(3) Who is typically responsible for creating/ updating privacy policies in your team/ company?
(4) Please recall a recent situation where a privacy policy was created or changed: What triggered it?
(5) A rigid privacy policy describes exactly the behavior of the code, with little room for deviation. A flexible policy may contain placeholder phrases for future development. What is your impression of how flexible or rigid your company's privacy policy is?

Maintenance of Privacy Policies

(1) What steps are taken in your company to keep the privacy policy up to date?
(2) How do you keep the privacy policy up to date as the software evolves?
(3) How do you communicate necessary updates to the policy?
(4) Please describe a recent situation where the policy needed to be revised due to software changes?
(5) Who is responsible for reviewing the privacy policy after code changes?
(6) Who is responsible for updating the privacy policy after code changes?
(7) How is the need to update the privacy policy usually identified?
(8) What kinds of changes typically trigger a policy update (e.g., feature changes, data flows, third-party services)?
(9) What information is needed to assess whether the privacy policy still matches the software?

Verification of Privacy Policies

(1) How do you check that your privacy policy accurately reflects what the software does?
(2) What steps do you follow to verify the match between the policy and the software?
(3) What aspects of the software do you compare against the privacy policy (e.g., data collection, storage, third-party sharing)?
(4) Who is responsible for verifying policy accuracy?
(5) How do you document that the policy has been verified?
(6) In what ways, if any, does your team keep track of verification steps or decisions?
(7) How do you ensure nothing is missed during verification?
(8) Please recall a situation where the privacy policy did not reflect software behavior? How was this identified and fixed?
(9) What makes verification difficult/ time-consuming?
(10) Who is involved in this verification process (e.g., QA, legal, product)?
(11) What challenges do you face in verifying policy accuracy?

Closing Questions

(1) What do you see as the biggest challenges when dealing with privacy policies as a software developer?
(2) What do you think would help you better manage the alignment of privacy policies with software behavior in your work?
(3) Who else do you think we should speak with to better understand these processes?
(4) What else would you like to share regarding the alignment between privacy policies and software behavior in software development?